Knowledge Discovery in High Content Screening

Kennis Ontdekken met High Content Screening

(met een samenvatting in het Nederlands)

Proefschrift

ter verkrijging van de graad van doctor aan de
Universiteit Utrecht
op gezag van de
rector magnificus, prof. dr. H.R.B.M. Kummeling,
ingevolge het besluit van het college voor promoties
in het openbaar te verdedigen op

woensdag 14 oktober 2020 des middags te 2.30 uur

door

Wiert Aldert Omta

geboren op 10 december 1983
te Emmeloord

Knowledge Discovery in High Content Screening
Wienand Omta
Doctoral thesis, Utrecht University, 2020
Cover image: AGM Data - IDR Open Microscopy
ISBN number: 978-90-393-7283-8

Printed by Boekdrukken.com

Promotoren:

Prof. dr. S. Brinkkemper
Prof. dr. J. Klumperman

Copromotor:

Dr. M.R. Spruit

Acknowledgements

My PhD journey started after finishing my master's degree at the end of 2011. I started working in two offices. One of my offices was in the Cell Screening Center (CSC) at the department of Cell Biology where I was managed by dr. David Egan. This is the person who I should thank the most. David did a great job in helping to support and guide me for all these years, thank you David! I also want to thank Judith Klumperman from the department of Cell Biology for her support!

The other office where I worked regularly was at the department of Information and Computing Sciences (ICS) where I was supervised by Sjaak Brinkkemper and Marco Spruit. They both made this project possible, thank you!

There are some colleagues at the Cell Biology department I want to thank. Lieke helped me out a lot in explaining biological terms, protocols and wet-lab stuff. The same goes for Daphne, Despina, Romina and Elisabeth. Thanks to them, I learned a lot about biology and automation, related to this topic.

There are also some colleagues from the ICS department I would like to thank: Amir, it is stunning how smart you are and what you have explained to me, I really enjoyed the nice discussions that we had! Also, Michiel Meulendijk was a huge support for me, also thanks for being my paranimf! I enjoyed all the drinks I had together with Kevin, Jaap, Erik, Ian, Garm and last but not least, my paranimf-buddy Vincent who recently finished his PhD!

Also, thanks to Ger Strous for all the interesting discussions that we had. Furthermore, a special thanks to Bertjan Goorkate from the IT department. I probably drove you nuts with all the IT requests and discussions we had, but thanks to you, I was able to do this!

A big thanks to my interns Marley, Maik, Mehdi, Jacob, Desmond, Andrie, Michael, Bob, Soad, Nils and of course Job. Job now works as employee #1 in our company Core Life Analytics! This was so valuable for me (I hope to them as well). Teaching you statistics, programming and designing and conducting experiments was a lot of fun, extremely useful and sometimes even resulted in a nice publication!

I would like to thank my friends Aad, Bas, Alex, Auke and Vincent for their support. They stressed the importance to relax, go to the movies and of course McDonald's after a nice bike ride! I especially want to thank my parents for their patience, support and showing interest in a topic they probably have a hard time understanding. Finally, I want to thank my fiancée Maaike for her support and patience, thank you love.

Contents

Chapter 1 – Introduction

1.1 High Throughput Screening (HTS)

All areas of life science research have become increasingly data driven (Leonelli, 2012). This was especially true in pharmacy and biotechnology as efforts began in the 1990's to industrialize the drug discovery process (Williams, 2011). It was greatly driven by the implementation of High Throughput Screening (HTS) methods for the identification of active candidate molecules in large libraries (Inglese & Auld, 2007).

HTS involves the use of robotics for carrying out large scale miniaturized biological experiments in an automated fashion (Mayr & Fuerst, 2008) in standardized microplates (Figure 1). It merges the use of robotic plate handling, automated liquid handling and automated assay reading. It is used in functional genomic screening for the identification of drug targets, preclinical hit-to-lead drug discovery, cell-based toxicity assessment, and high throughput mechanism of action screening. In HTS, reagent libraries can be screened against biological assays using fluorescent proteins, chemical substrates, antibodies or cells. Various other high throughput technologies that are not discussed here were introduced in this period. These included DNA microarrays, high throughput sequencing, and mass spectroscopy for lipidomics, proteomics and metabolomics. (Kraljevic, Stambrook & Pavelic, 2004).

High Throughput Screening (HTS) allows for thousands to millions of small biological or pharmacological experiments. Using HTS, one can quickly find active molecules, genes or antibodies that affect relevant biological processes. This is input for the understanding, design and development of therapeutic drugs (Kraljevic, Stambrook & Pavelic, 2004).

The biological systems used in HTS tend to be robust, relatively simple while the HTS campaigns run could involve up to a million or more individual tests, each one might only generate one or two data points (Macarron, 2011). This resulted in a data management problem which was solved by the implementation of enterprise-scale SQL databases. The data analytics was relatively straightforward, however.

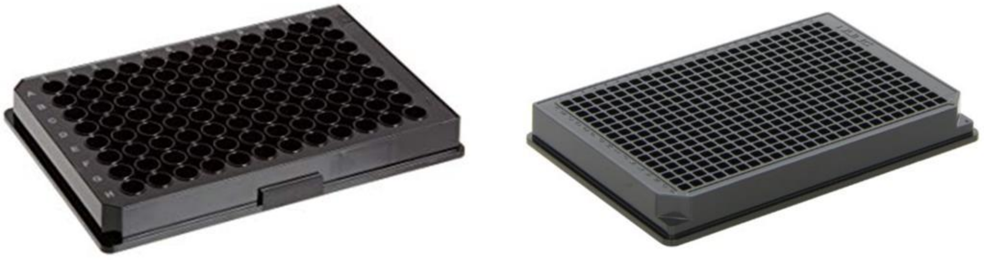
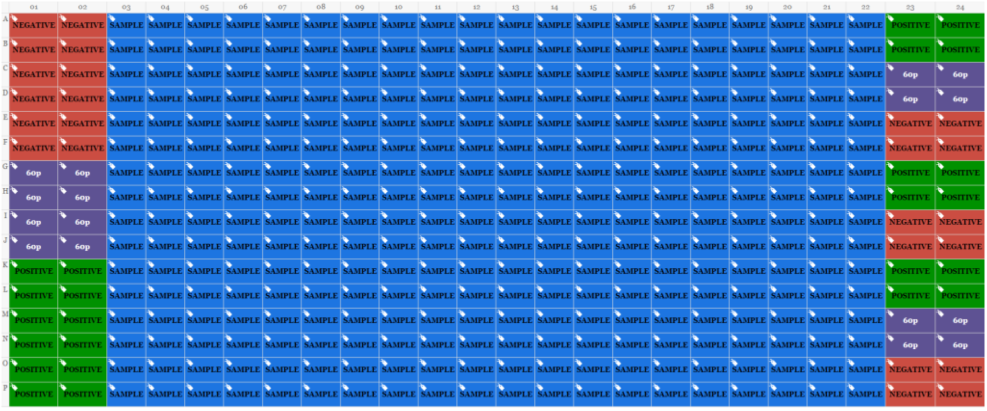


Figure 1. *This figure represents two different types of microplates. The left plate is a 96-well plate and the right one is a 384-well plate with a much higher density. Both plates are exactly 12.7 * 8.5 cm, but the 384-well plate has four times the number of wells.*

In order to fully appreciate the complexities of HTS, it is important to understand important concepts which are introduced below.

Controls

A typical experiment includes controls that give an expected known positive or negative result. In a microplate experiment, a certain number of wells is set aside for these controls (Bray & Carpenter, 2017). An example is shown in figure 2. Controls are added for multiple reasons. One of them is to verify that the assay works as expected. Can we be sure that labeled negative wells are negative and the positive wells are positive? Another reason to include sufficient controls is to perform solid Quality Control (QC), using QC metrics such as Z` and SSMD (Birmingham et al., 2009; Martin et al., 2013), which can signify the difference between a positive control and a negative control. These QC metrics are introduced for the verification of batch or plate effects, undesired biological errors, or side effects due to environmental settings such as temperature, lights and moisture that can result in disrupting the normal function of the assay by e.g. excessive cell death. (Magidson & Khodjakov, 2013). Figure 2 is an example of a typical 384-well plate map which reflects the configuration of controls on both left and right sides of the microplate. Columns 1, 2, 23 and 24 are used for built-in controls. These controls are particularly important in the data analysis part of the HCS workflow. Apart from calculating QC metrics, controls can be used to train a classification model or calculating similarity or distance scores, more about this in section 1.4. A thorough analysis of the controls should always be performed after image analysis to verify that the screen indeed worked as intended and to determine the quality of the screen. Specifically, designed HCS software can demonstrate the required information within minutes to determine if the desired quality level is achieved.



	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A	NEGATIVE	NEGATIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	POSITIVE	POSITIVE
B	NEGATIVE	NEGATIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	POSITIVE	POSITIVE
C	NEGATIVE	NEGATIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	60p	60p
D	NEGATIVE	NEGATIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	60p	60p
E	NEGATIVE	NEGATIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	NEGATIVE	NEGATIVE
F	NEGATIVE	NEGATIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	NEGATIVE	NEGATIVE
G	60p	60p	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	POSITIVE	POSITIVE
H	60p	60p	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	POSITIVE	POSITIVE
I	60p	60p	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	NEGATIVE	NEGATIVE
J	60p	60p	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	NEGATIVE	NEGATIVE
K	POSITIVE	POSITIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	POSITIVE	POSITIVE
L	POSITIVE	POSITIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	POSITIVE	POSITIVE
M	POSITIVE	POSITIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	60p	60p
N	POSITIVE	POSITIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	60p	60p
O	POSITIVE	POSITIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	NEGATIVE	NEGATIVE
P	POSITIVE	POSITIVE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	SAMPLE	NEGATIVE	NEGATIVE

Figure 2. *Defining controls and samples*
In this figure, three different controls are defined: NEGATIVE (in red), POSITIVE (in green) and 60p (in purple), the latter being the POSITIVE control at 60 percent of its concentration. The samples in blue are the wells in this screen that still have an unknown phenotype and are the focus area of the screen. These wells are typically investigated using proven HCS analysis methods in order to find the wells that show interesting phenotypes for further investigation.

Replicates

Replication is used in many scientific areas to ensure the reproducibility of an experiment. This is especially important in cell-based assays. Although cells within a microplate well receive the same treatment, the phenotypic variation between individual cells can still be extremely high. This is quite common in biology and could be due to the cell cycle state, technical factors such as the uneven addition of the reagent or environmental differences. Therefore, screening is regularly carried out in replicates to increase the reproducibility. The replication can take place in two ways: (i) the replication of a well on the same plate, this is called a biological replicate. Replication carried out on a separate plate is called a technical replicate. Technical replicates are usually more expensive but due to batch and plate effects, it is better to perform replicates on separate plates or a combination of technical and biological replicates (Murie, 2015).

Wet Lab Protocols

Wet-lab work basically entails everything that goes on in a biochemical lab and requires a white lab coat. Dry-lab work is anything related to managing and analyzing biological data (derived from wet-lab work) that does not require a biochemical lab. A typical example of wet-lab work is pipetting, which is manually moving liquid from one to another plate or well using a pipet (Scholl, Wille & Van Laerhoven, 2015). Automated liquid-handling uses robotics to carry out the pipetting work in an automated fashion. This is faster, more precise and less error prone. Protocols in HTS involve the process of automation with everything that is required to replicate the manual (wet-lab) activities of a biochemical researcher in the lab (Burbaum, 1998). The protocol includes (i) the concentration of a chemical that will be screened as well as (ii) the media, (iii) the concentration of that media, (iv) the fluorescent dyes, (v) the incubation time and (vi) the cell line(s) being used for the screen.

Data Analytics in HTS

Data gathered from HTS experiments generally involve one or two measurements or extracted features per well. The analysis generally implies the following: First, quality control is performed for the inspection of batch and plate effects. Data can be normalized by a negative control or corrected using a B-score, (to account for systematic edge effects), or loess regression method and is frequently scaled by a z-score (Pelz et al., 2010). Hit Selection is carried out by for example, a ranking based on p-values or three standard deviations from the SAMPLE mean. A more sophisticated approach is the use of SSMD for controlling the number of false positive hits. Most biologists, however, do not find this intuitive to read and the method is not available through most software implementations (Birmingham et al., 2009).

Primary Screen

A primary screen is typically screening a full library in a single dose. This could be a compound library or a full genome-wide siRNA library. When a primary screen is large (>10k), one can decide to screen in singletons (no replication) to decrease costs (Harper & Pickett, 2006). The purpose of the screen is to narrow down many reagents to a manageable list of suspected active compounds, “hits”, that can then further be characterized in secondary or validation screens.

Secondary & Validation Screen

These are screens in which hits from a primary screen are characterized. After a primary screen, a certain cut-off is defined and the reagents inside the subsection are included in the validation screen. This is frequently based on a hit selection method. A validation screen can involve other technologies, such as proteomics or next generation sequencing.

Reagents

Reagents are biological or chemical substances added to a screen causing a biological response that can lead to a phenotypic change in a cell, i.e. the composition of cells' observable characteristics. Examples of reagents are antibodies, compounds, ligands, natural products, FDA approved drugs, siRNAs, shRNAs or miRNAs. RNAi (siRNAs, shRNAs or miRNAs) is a technology used to decrease gene function in a cell.

1.2 High Content Screening (HCS)

In the late 2000s it became clear that the vast investment in the development of fully integrated industrialized drug discovery companies did not deliver the anticipated benefits. This led to a sea-change in the industry (Williams, 2011). In recent years the large pharmaceutical companies have greatly reduced their efforts in the earliest stages of drug discovery, especially target identification and validation, and new leads discovery, and now focus more on attractive molecules and platforms from academia and biotech's that can be optimized and taken more rapidly into advanced animal models and clinical trials.

Phenotypic Drug Discovery

In all sectors of the industry there is a greater emphasis on more complex, physiologically relevant biological systems. This includes the use of patient-derived cell cultures, 3D cell systems and critically, methods that give a more holistic insight into the response or phenotype of these more complex biological systems to a candidate drug molecule (Hughes et al., 2011). These phenotypic methods revolve around the extraction of many numeric results from any individual experiment. These constitute a profile that can be used to classify the response using advanced data analytics methods including artificial intelligence (Ljosa et al., 2013).

One example of these phenotypic technologies is High Content Screening (HCS), in which automated microscopy is used to image cells that are being screened against libraries of chemicals, antibodies or other potentially bioactive reagents (figure 3). Automated image analysis is used to generate a multiparametric numeric profile that can be used to characterize cellular changes. In recent years, it has become a critical technology in virtually all drug discovery programs in the pharmaceutical industry (Hughes et al., 2011).

Data Analytics Bottleneck

The increased complexity of the data being generated in drug discovery means that biologists have to increasingly turn to specialist data scientists in order to analyze their data. This introduces delays in the workflow which are compounded by the complexity of the biological systems being used. The data scientists need an in-depth understanding of the biology in order to be able to mine the data.

Figure 3. *A Thermo Scientific, Cell Insight CS7 Automated Microscope. This device can handle microplates, shown in figure 1 to capture large numbers of microscopic images in an automated manner.*

High Content Screening (HCS) is a subset of High Throughput Screening (HTS), a screening technique that acquires images using automated microscopy (figure 3) using 96, 384 or sometimes even 1536 microplates (figure 1). Traditional HTS does not require image acquisition and image analysis because no images are taken. The measurement used is the amount of light in a well. This dissertation is specifically focused on the data analysis of High Content Screening (HCS) data. HCS differs from HTS in that a true multiparametric or multivariate data set is acquired, containing tens up to thousands of phenotypic features (Singh, Carpenter & Genovesio, 2014). In HCS, acquired screening images using automated microscopy are subsequently used in automated image analysis to generate numeric multivariate data sets. In short, HCS results in extraordinarily rich data that can be used to identify hits or outliers. HCS has been used in various disciplines such as drug discovery, toxicology, and functional genomics e.g. RNAi. Knowledge gained from these screens can be input for pre-clinical and clinical trials for the development of future drugs in Pharma. HCS involves a combination of robotics, liquid handling, image acquisition, image analysis and data analysis, as shown in figure 4.

1.3 HCS Workflow

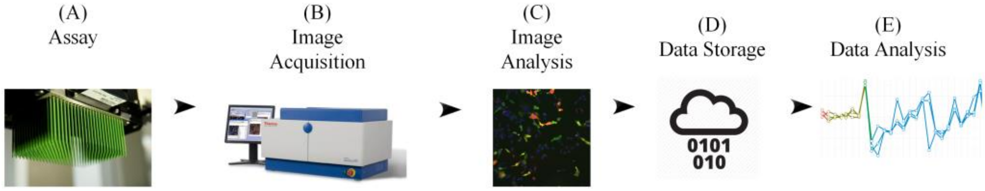


Figure 4. *The High Content Screening (HCS) workflow*

A. *Is a robot that can move and dilute liquids from one plate to another completely automated.*

B. *Is a Thermo Fisher automated microscope box. It can handle 96 and 384 well microplates and capture up to 80X images with a maximum of 6 Channels.*

C. *Represents the process of object recognition, image segmentation and image analysis. In this process a data matrix is generated that represents all images in a numeric fashion.*

D. *Represents the process of storing all the generated data in B and C.*

E. *Represents the process of analyzing the data generated in C and stored in D. This process can reveal new insights and knowledge about phenotypes.*

The first step in the High Content Screening (HCS) workflow is the use of robots that make it possible to move around liquids from master to assay plates, fixing, staining and washing (figure 4A). Frequently, a compound library or siRNA library contains master plates with high concentrations of liquids. These can be diluted and moved to plates used for screening. Therefore, the concentration can be managed for a screen. After all the required preparation for all microplates is done such as adding media, fixing cells and adding fluorescent dyes to capture the phenotype of the cell. Images are captured using an automated microscope (figure 4B). The phenotype of the cells may be captured using multiple fluorescent channels (figure 5). Each taken image represents a part of the well and is called a field. A well may contain tens of fields depending on the resolution of the lens: the higher the resolution, the more fields.

This process is repeated for the number of fluorescent dyes used in the screen. A fluorescent dye is a chemical that binds to a specific protein. In this way the presence or absence of the protein can be perceived (figure 5). For example, an assay contains 24 384-well microplates containing 2 channels and 12 fields (images) per well then 24 (microplates) * 384 (wells) * 12 (fields) * 2 (channels) = 212,184 images of e.g. 512x512 pixels are captured. After the images are captured and stored, image analysis is used to extract numeric data from these images (figure 4C).

The extracted numeric data sets can contain thousands of metrics called features that describe the phenotype of each cell. Typical examples of numeric features that can be extracted are size, area, intensity and texture features. When many features are generated, the resulting data matrix may contain highly redundant features. The features are deeply dependent on the used fluorescent channels in a screen and can highlight extremely specific information related to one or more channels. These features are the main input for the data analysis. Using multiple features in a multivariate analysis is also called multi-parametric data analysis (Tsiper, et al., 2012). The numeric data sets need to be stored in a structured way in order to query them for data analysis. The size of the numeric data sets per individual plate is usually up to tens of gigabytes per microplate.

The final step in figure 4 is the data analysis part of the numeric data, generated and stored in figure 4C & 4D. Omta et al (2012) found that image analysis software is mature but problems arise from that point on forwards in the HCS workflow. Therefore, the following overarching research objective of this dissertation is to design a data analysis system that improves the high content screening workflow to uncover new biomedical knowledge.

Numeric HCS data extracted from cell-based experiments are usually metrics derived from one or more fluorescent dyes. Fluorescent channels can be interpreted as color channels focused on a certain spectrum of the visual. For example, the DAPI or Hoechst dyes visualize the nucleus (figure 5). The nucleus is captured by DAPI, the cytoplasm is captured by Calcein AM and lysosomes are captured by Lysotracker. Other channels can light-up other parts or organelles of the cell, such as mitochondria, endosomes, lysosomes or neurites. Within HCS screening, we use up to five channels combined in one screen (see Bray et al., 2016). Adding more channels will introduce overlap in their light spectrum and increases redundancy. Each of these channels is produced in grayscale images but for human interpretation, each channel is depicted using a distinct color where the nuclei are frequently depicted in blue (see Hoechst, figure 5).

Features are extracted from these channels such as the intensity, area or simply by counting objects. HCS software helps to identify objects in the cell using segmentation. One way to achieve this is by first identifying the nucleus. Using this method, it can be important to determine what the background is and what a valid nucleus is using manual thresholds. Setting these thresholds can have a great impact on the outcome, i.e. the quality and validity of the image analysis. The numeric features that are derived from the microscopic images can then be used as input for statistical analysis. In this step, analytical methods need to be carefully selected or designed.

Data Resolution

The size of a numeric HCS data set depends on two factors: (i) the number of features and (ii) the data resolution. The number of features is directly dependent on the number of fluorescent channels used in the assay and the extensiveness of calculating different types of features such as intensity, area and texture features. The resolution of the data has a strong relation with the number of records in a data set. The number of records in the data set is related to what a record represents. A record in a data set can represent four different things: (i) it represents a well of which e.g. the mean is calculated based on all the cells in a well for measurement x, or (ii) it represents a field with again e.g. a mean calculated based on all the cells in a field or (iii) it represents a cell with its original measurement or the mean calculated based on all the organelles (objects) in a cell. Recently, software packages have become available that can output data (iv) in which each record represents a measurement of an organelle, also called an object.

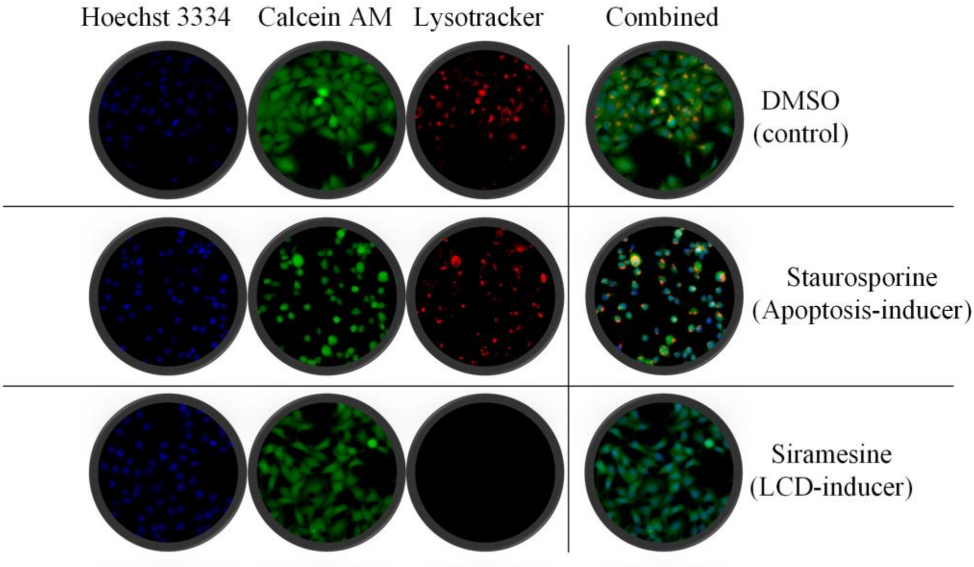


Figure 5. *High Content Screening*
This figure shows a matrix of 4 x 3 images. The first three columns represent the fluorescent label or dye used and the fourth column shows an image of all three labels combined, also called an overlay. The rows represent different chemicals that were added to the cells captured in the images. Hoechst labels the nuclei of the cell in blue, Calcein labels the calcein in green and Lysotracker labels the lysosomes in red. Each of the first three rows represent a different chemical, the first row represents DMSO which is the used medium for chemical screening; in the first row only DMSO is added and represents the negative control. The second row represents the phenotype of cells when Staurosporine is added. Staurosporine kills cells through apoptosis, a very standard cell death pathway. Row two shows cells that are brighter and smaller compared to row one and the lysosomes (in red) in row one and two are still present, compared to the phenotype of the cells when Siramesine is added, which is shown in the third row. Siramesine is a drug that also kills cells but through the lysosomal cell death pathway (LCD). LCD is a quite different way of killing cells compared to row two (using Staurosporine) and demonstrates ellipse shaped cells (in green) and the lysosomes have disappeared (in red) because they died.

1.4 Data Science in HCS

Data science is the field in which professionals turn large and unstructured data into actionable information using methods from the fields of statistics and data mining to uncover insights (Davenport & Patil, 2012). Data science is a multidisciplinary field with applications in business, finance, biology, physics, astronomy and many more. The field of data science requires many skills such as transforming data, writing code, displaying information in a visual way and the ability to communicate to product managers for critical decision-making information (Davenport & Patil, 2012). Here, we describe three major skills to master: computer and network architecture, programming and data mining (Ren et al., 2017; Wu et al., 2018; Horton, Baumer & Wickham, 2015; Wickham, 2019). Most data scientists in the field of life sciences or HCS are called bioinformaticians or computational biologists. That is the field in which computer science and biology are merged. A trained bioinformatician will have a better understanding and more sophisticated knowledge and skills that are required within the domain of life sciences or HCS.

First, computer and network architecture skills aim at the optimization of selecting the right hardware, the right software that manages the hardware and the communication between the devices within the network, often collaborating in parallel to solve computations faster (Almasi & Gottlieb 1988). These settings can take place within a local network, e.g. on a High-Performance Computing (HPC) cluster, or on a public or hybrid cloud solution.

Second, programming is a particularly important skill in data science. Frequently used programming or scripting languages in data science are Java, Python, R, Scala and SAS. These can be used to manage the logic inside the architecture of a computer cluster, serving as an interface or the back-end of a data science platform where statistics is used. Programming languages like R and SAS are well-known as statistical programming languages for which many libraries for a specific field such as Bioinformatics are available. Most programming languages are open source with a highly active community.

The third important skill in data science is data mining, arguably the main skill in data science. This skill involves the ability to preprocess data and to use statistical methods to find patterns in the data that serve as input for finding new and relevant knowledge in the application domain also known as knowledge discovery. Next to that, an important capability embedded within this skill is to visualize the data. Today, this is possible in many ways from quite simple scatterplots, line plots and bar graphs to multivariate heatmaps or contour plots, even in an interactive manner. Many packages in Python, R and Java such as plotly or ggplot2 are being used to visualize data in various (interactive) ways (Galili et al., 2017).

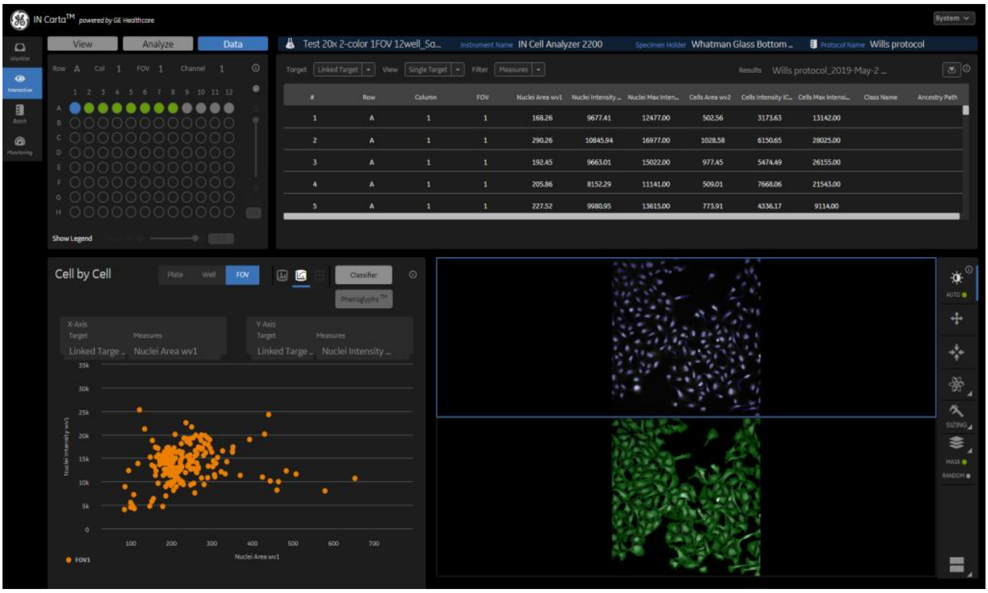
Moreover, when data science research focuses on the development of advanced analytical applications which encompass the entire knowledge discovery process, then Spruit & Jagesar (2016) define this research field of study as Applied Data Science (ADS). “In contrast to theoretical data science, applied data science research primarily focuses on applying machine learning techniques to solve stringent issues in application domains such as HCS, by developing advanced software systems that provide an effective and efficient end-user environment to embed the machine learning or statistical techniques into. However, if applying existent machine learning techniques do not suffice, then this will trigger more theoretical data science research to develop the necessary techniques that do solve the problem at hand" (Spruit & Lytras, 2018).

Software in HCS

Within the HCS workflow, multiple software packages are designed for a specific process in the HCS workflow (figure 4). The first process in the HCS workflow is the robotics phase (figure 4A). This process involves moving and diluting liquids used for screening and is outside the scope of this research. The second phase in the HCS workflow is image acquisition (figure 4B). This is where the images are captured using an automated microscope. Once the images are taken, image analysis (figure 4C) is carried out. Multiple vendors such as GE Healthcare, Molecular Devices, Perkin Elmer and Yokogawa have developed automated microscopes with their own commercial software that manages the device.

This software is taking care of the image acquisition and (on-the-fly) image analysis, where the images are in the process of segmentation and feature extraction. In other words, objects in an image are recognized and multiple statistical measurements can be derived from each object. The resulting numeric data and the related images need to be stored (figure 4D). There are standard and custom solutions for storing the images and numeric data. This is usually done on-premise, using a database or using a bulk storage and storing images in high quality, i.e. TIFF. The storage device requires a large capacity for storing all the data over time. The numeric matrices that are stored can be used in the data analysis process (figure 4E). A few examples of software packages that can manage the image acquisition and image analysis and storage are Cell Insight, Store, INCARTA and Columbus. Figure 6 depicts INCARTA, a software platform that can capture live cells using a 96-well microplate. The images are stored on-the-fly in a connected database. Images can be queried later for the image analysis process.

The image analysis can be executed by commercial packages from vendors like GE Healthcare, PerkinElmer and Thermo Fisher of which the software package that manages the image acquisition can usually also run standard image analysis protocols. For specific protocols one can move to specialized commercial or open source software packages. One of the most cited open source image analysis packages currently is called CellProfiler (Carpenter et al., 2006). CellProfiler was introduced at the Broad Institute in 2006 by Ann Carpenter and is currently a package that can be installed using a small guideline. It can be optionally deployed using cloud computing in AWS using grid computing, to enhance the speed of the software package.



Row A Col 1 FOV 1 Channel 1

Test 20x 2-color 1FOV 12well_So... Instrument Name IN Cell Analyzer 2200 Specimen Holder Whatman Glass Bottom ... Protocol Name Wills protocol

Target Linked Target View Single Target Filter Measures Results Wills protocol_2019-May-2 ...

#	Row	Column	FOV	Nuclei Area wv1	Nuclei Intensity ...	Nuclei Max Inten...	Cells Area wv2	Cells Intensity IC...	Cells Max Intensi...	Class Name	Ancestry Path
1	A	1	1	168.26	9677.41	12477.00	502.56	3173.63	13142.00		
2	A	1	1	290.26	10845.94	16977.00	1028.58	6150.65	28025.00		
3	A	1	1	192.45	9663.01	15022.00	977.45	5474.49	26155.00		
4	A	1	1	205.86	8152.29	11141.00	509.01	7668.06	21543.00		
5	A	1	1	227.52	9980.95	13615.00	773.91	4336.17	9114.00		

X-Axis Target Linked Targe... Measures Nuclei Area wv1

Y-Axis Target Linked Targe... Measures Nuclei Intensity ...

Figure 6. *A screenshot of the Image Analysis Software INCARTA*
This figure represents the user interface of INCARTA (GE Healthcare). It captures and manages the images and calculates features that can be used in the DAP HCS workflow (figure 4E). On the right side of this figure, two channels; DAPI and Calcein are depicted. In the lower left corner, the calculated features nuclei area and nuclei intensity are visualized in a scatterplot.

The fifth phase (figure 4E) in the HCS workflow is data analysis. The data is usually stored in a matrix-oriented fashion in which rows represent either organelles (objects), cells, fields or wells and columns represent features that describe phenotypic properties of the respective rows. Software packages for data analysis comprise Spotfire, HC StratoMineR and Gene Data. These software packages are all commercially available.

The focus of this dissertation is the Data Analysis part of the HCS workflow (figure 4E). Nevertheless, the data analysis has tight connections with the data storage (figure 4D) and Image Analysis (figure 4C). The robotics process (figure 4A) is conducted by biologists who prepare microplates, chemicals, cells and dyes and is part of the wet-lab work. Images are captured using the wells in the microplates at the acquisition stage (figure 4B). Then the image analysis of the resulting images is performed that outputs large and complex numeric data sets. These data sets contain millions of cells and describe each individual screened cell in hundreds or thousands of phenotypic properties.

Knowledge Discovery in Databases Framework

For the fifth HCS phase (figure 4E) the Knowledge Discovery in Databases (KDD) framework (figure 7) is used as a basis for outlining this HCS phase, from now on called the Data Analysis Process (DAP, figure 8). KDD was introduced by Fayyad et al. (1996). The framework starts with a bulk of data that is referred to as the raw data or in other words, unprocessed data. From that point onwards, a selection can be made focused to data that is most relevant, filtered or randomly selected. The first selection is made and that is from this point on the 'target data' or the data that will be used in all future steps within the KDD model. The target data is then preprocessed. This can involve any form of processing so that the data can be used in a form that is accessible and can be used in a format that is useful for further processing of the data, also called Extract, Transform & Load (ETL). The next phase is transforming which can involve any algorithm or transformation of the data that makes the data more useful for steps further downstream in the workflow. Examples here can be normalization for certain assumptions that are required in steps during the data mining process. The data mining process is one of the most important steps in the KDD model in which patterns can be found using e.g. supervised or unsupervised learning. From these patterns, e.g. on average we find one black swan for every 10,000 swans, one can derive knowledge, i.e. black swans are in a severe minority but do exist. This knowledge can be interpreted and evaluated and of course interventions can be designed and implemented.

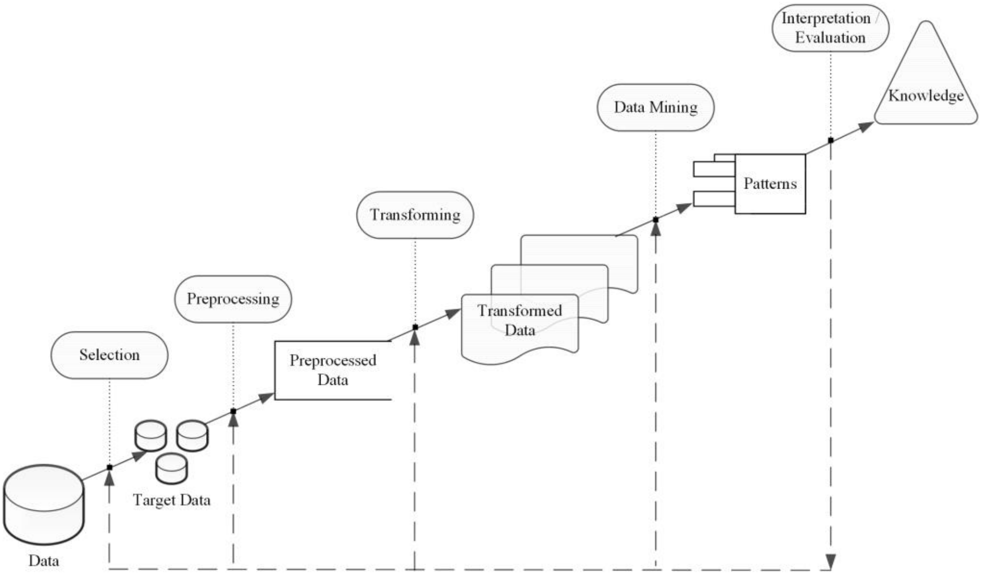


Figure 7. *Knowledge Discovery in Databases (KDD) Framework*
This is the KDD framework introduced by Fayyad et al., (1996) and has been a popular framework for data mining ever since. The phases that are described in this framework can be mapped one by one to the HCS workflow depicted in figure 8.

Data Analysis Process (DAP)

The Data Analysis Process (DAP) (figure 4E & 8) operationalizes the fifth phase of the HCS workflow and is outlined by Young et al., (2008), Caseido et al. (2017) and Omta et al. (2016), and can be compared with the KDD model by Fayyad et al. (1996). The term DAP is introduced to easily refer to the main topic of this thesis which concerns the numeric data analysis process from the numeric data that is extracted from the image data, generated by an automated microscope.
The workflow depicted in figure 8 describes the HCS DAP, of which the Raw Data section refers to the Data section of KDD (figure 7). The Meta Data and ETL (Larson & Chang, 2016) (figure 8) refers to the Selection section in the KDD process (figure 7). A part of the process Variable Selection and Quality Control (figure 8) refers to the KDD section Preprocessing (figure 7), then Normalization, Transformation & Scaling and Dimensionality Reduction (figure 8) refers to the KDD section Transforming (figure 7) and finally Hit Selection and Clustering (figure 8) refers to the KDD section Data Mining (figure 7) to find patterns. The Interpretation/Evaluation section in the HCS domain is a manual process of which the results of the Data Mining phase need to be investigated with the naked eye.

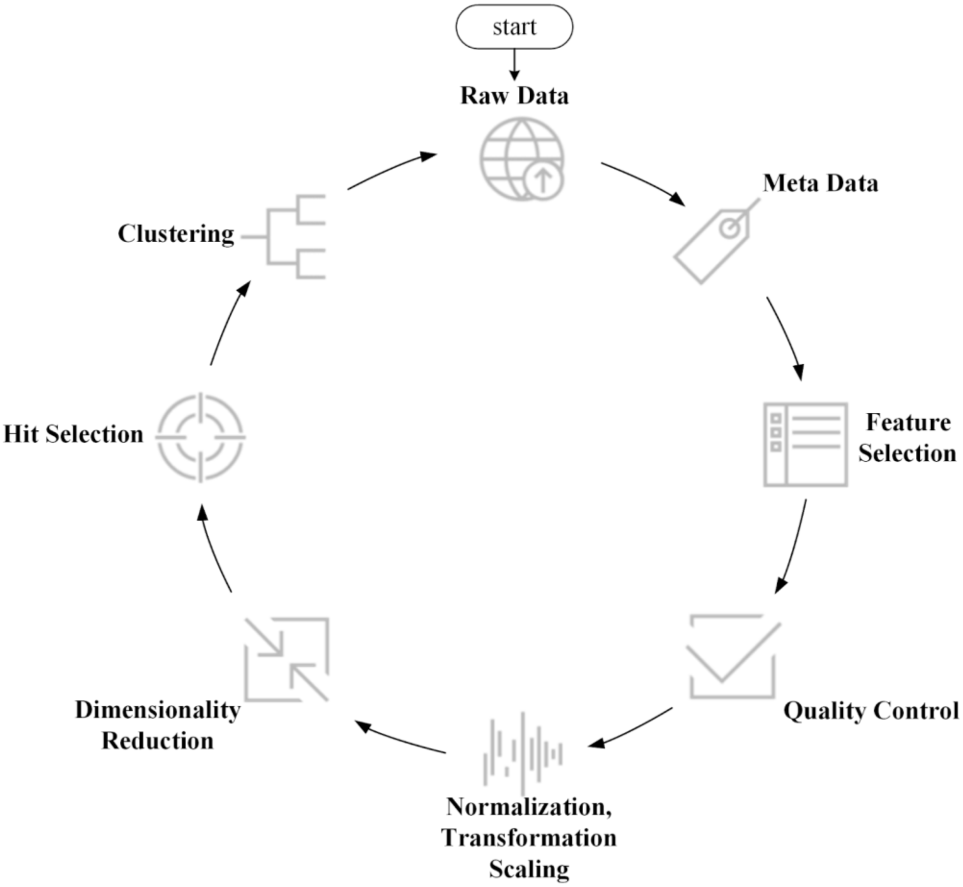


Figure 8. *The basic workflow phases of the Data Analysis Process (DAP, figure 4E) of HCS, as described by Omta et al. (2016), Young et al. (2008) and Caseido et al. (2017).*

Here each phase can have various settings and hyper parameters that define the workflow to carry out an analysis. Typical examples of HCS data analytics workflows can be defined in two categories: unsupervised and supervised learning. Unsupervised clustering is a commonly used technique in HCS in which a set of features is used to create groups or clusters of reagents that are highly similar within the cluster, where individual clusters are distinct from each other (figure 9). Also, within hit picking, unsupervised techniques such as distance-based techniques allow for a multivariate cut-off to determine what is a hit and what is not.

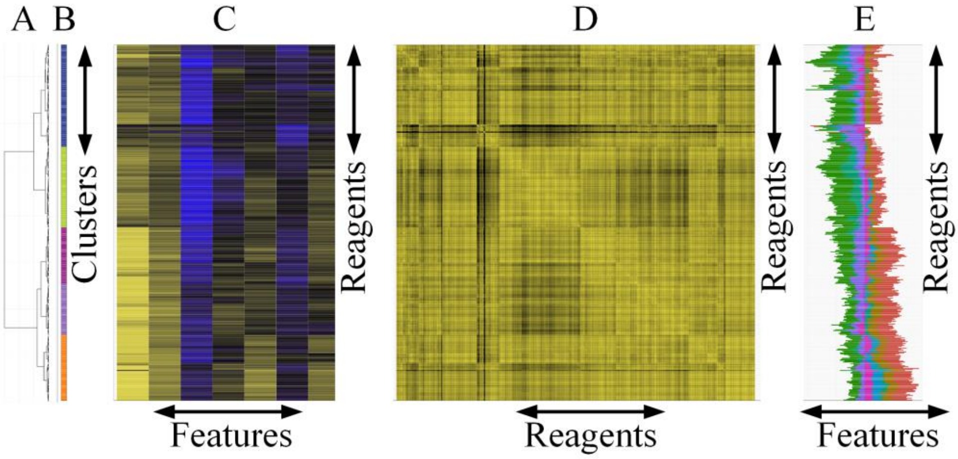


Figure 9. *Unsupervised clustering of HCS data*
This figure represents the same data five times horizontally aligned. At the left of the figure, (A) a dendrogram represents the data in a hierarchical way. Just next to it is (B) a K-means bar plotted containing 5 colors. Each color represents a K-means cluster meaning the reagents within the cluster are very similar and can be interpreted as a group. Next to it, (C) a heatmap containing 7 features/columns and is scaled and visualized, in which yellow means high and blue low, these are the input to order the data in a specific way, e.g. agglomerative. (D) The similarity heatmap represents a comparison of each possible combination of reagents using a cosine vector similarity score that results into a symmetrical matrix. (E) The last visualization is a scaled version of the 7 features (in 7 colors) also used in the heatmap but now stacked as bar graphs.

Finally, a typical example of a supervised approach in HCS is the use of (non-linear) classification. A set of features is used for training labeled data (figure 10) expressing specific phenotypes (classes). Then the classes are being trained to distinguish them as much as possible using the feature set (figure 11). This is usually carried out by splitting the labeled data into two sections; a training set used for training, and a test set used for evaluating the model created from the training set. After evaluating the trained model, all data without a label (the unlabeled data) can then be classified into one of the trained classes using the model. This process can add valuable information whether a part of the data is closer to Class A or Class B.

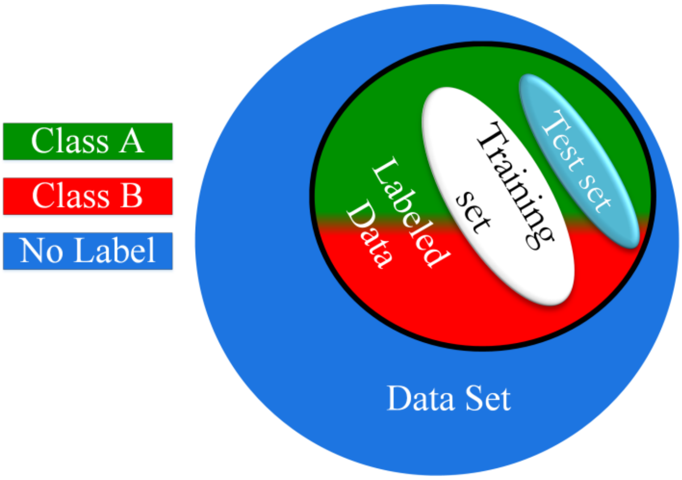


Figure 10. *Data Sampling*
This figure explains the basics of supervised learning. The circle represents the complete data set. The blue part represents the unlabeled data, meaning the data has no annotation. The red and green parts are known but are both different. The Training set is a randomly selected part of the data that will be used for building a model. The Test set is a randomly selected part of the data that will be used to evaluate the model.

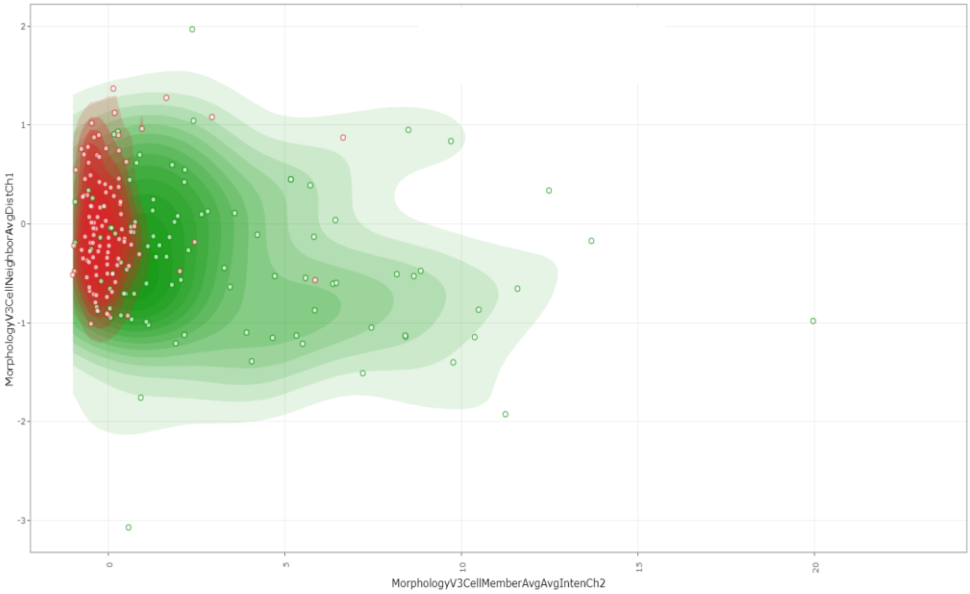


Figure 11. *Results of supervised learning*
This is a contour plot that is showing the coverage of two classes (Class A and Class B). Observed items are depicted as dots and the expected areas are depicted using contours. Class A is depicted in red and is a negative control. Class B is represented in green and is a positive control. The greener the dots are positioned within the green area and similar for red, the higher the observed measurements are inside the expected area thus, the higher the accuracy of the model is.

1.5 Research methods

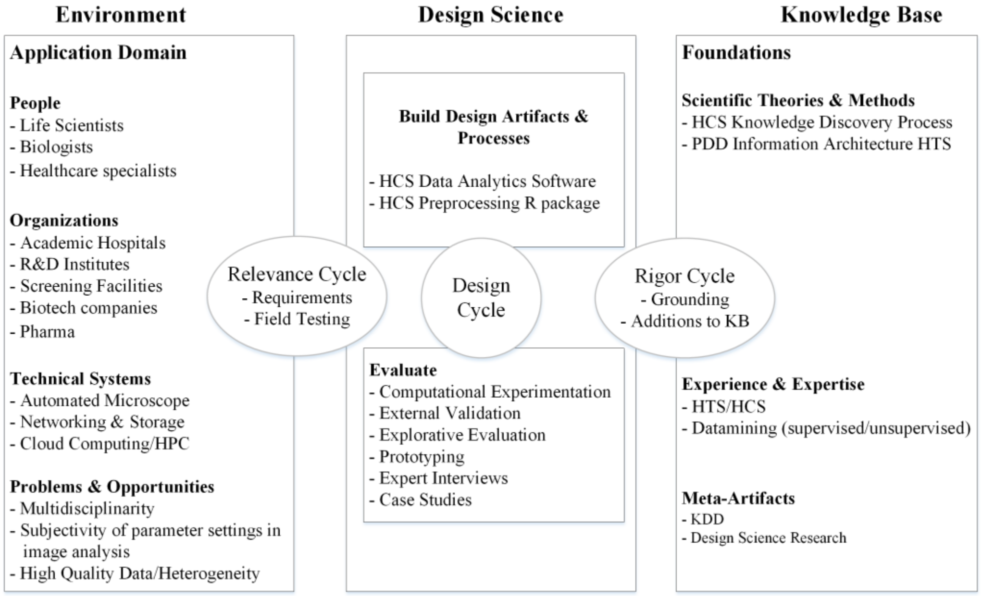


Figure 12. *The Design Science Research Cycle (Hevner et al., 2004), applied to this thesis*
This figure shows all the stakeholders in the domain of HCS. First the people that work with the technology and the software, the organizations and institutes in which these people are employed, the technology they use (Technical Systems), the methods and theories they use, the expertise these people have and the underlying frameworks that are a basis to apply their work. The outcome of their work can finally be evaluated using several methods such as computational experimentation or external validation.

This thesis is in alignment with the empirical cycle of the Design Science Research (Hevner et al., 2004) (figure 12). The theory is designed around technology development and includes seven guidelines; (i) Design as an artifact; it must produce a viable artifact such as a model or a method. (ii) Problem relevance; the objective is to build a technology-based product relevant to the business. (iii) Design Evaluation; The evaluation of the utility, quality and efficacy of the artifact must be demonstrated using well-executed evaluation methods. (iv) Research Contributions; research must produce clear and verifiable contributions. (v) Research Rigor; The research is based on rigorous methods both for the construction as well as the evaluation of the artifact. (vi) Design as a Search Process; An effective artifact is aligned with satisfaction in the problem environment. (vii) Communication of Research; Research must be presented to technology- as well as management-oriented audiences.
This thesis operationalizes the Design Cycle within Design Science Research in the environment of High Content Screening (HCS) as follows. Examples of

artifacts in this research are models e.g. the HCS Knowledge Discovery Process and a software platform, i.e. HCS Data Analytics Software. The artifacts are relevant in the business field of HCS and are technology driven. The evaluation of these artifacts was done using computational experimentation, case studies, expert interviews and prototyping. Research contributions were published in peer-reviewed journals, verified and evaluated by many experts in the HCS field. The data including a detailed description, a Standard Operating Procedure (SOP), the resources and version numbers, were provided in the supplementary data when applicable for a verification of the research output and findings. The construction of the artifact was carried out using a prototyping approach. The evaluation methods were carried out by case studies, expert interviews, exploratory evaluation and external validation e.g. the use of external ontologies to verify the outcome. The satisfaction of an artifact can be decided upon acceptance of a peer-reviewed journal article that discusses the artifact. Communication of research was done towards technology-oriented audiences by means of journals and conferences. The business or management-oriented audiences were also approached during conferences and by means of other channels such as social media, workshops, collaborations and lectures.

In Design Science Research, the framework is applied in an iterative way. The cyclic approach makes it possible to work on an artifact, evaluate it by getting feedback from its users and then incorporate this feedback to incrementally improve the artifact.

1.6 Research Questions and Dissertation Outline

The overarching objective within HCS research is to uncover new knowledge that helps scientists and entrepreneurs to steer and identify phenotypes that are designed for the curation of diseases and fundamental understanding of biology. Here, we identified two major problems in the field of HCS:

I How to derive and capture knowledge from HTS and HCS?
II How to deal with the wealth of data to capture knowledge from HCS data sets?

First, the overall process of HCS needs to be investigated. In this way, gaps can be identified in the process that require either adjustments that can be improved by algorithms or by the support of specific software.

Second, the focus should be on the wealth of rich data that is created during the HCS workflow: how to deal with this wealth of data and how can this data be used as efficiently to generate knowledge that can be used by experts, scientists and entrepreneurs in the field. In addition, the HCS software should ideally be targeted to domain experts within the HCS field, whether these are biologists, statisticians, bioinformaticians, data scientists or computer scientists. In other words, the software must be tuned to the target audience. This implies that it should be clear for whom it should be designed. Finally, the accessibility of the software should be investigated, in order to enable scientists to access the software in the right place and at the right time.

The identification of the two problems above leads us to the following overarching research question.

MRQ: How can multi-parametric data analysis contribute to effective knowledge discovery in High Content Screening?

RQ1: What is an effective information architecture for High Throughput Screening?
RQ2a: What are the required workflow and software components for analyzing HCS data?
RQ2b: What are the implications of single- vs. multi-parametric data analysis of HCS data?
RQ3: How can supervised learning approaches contribute to the analysis of HCS data?
RQ4: What are the effects of using interactive data visualizations in HCS data analysis?
RQ5: How can preprocessing in numeric data analysis be automated?
RQ6: To what extent can the HCS data analysis workflow be applied to low content data?

This thesis consists of eight chapters starting with an introduction, then six chapters representing articles which are all published in renowned peer-reviewed journals in the surrounding field of bioinformatics and finishes with a conclusion chapter. Here, chapter 2 - 7 are briefly described.

Chapter 2: HTS-IA: High Throughput Screening Information Architecture for Genomics

Chapter 2 presents the ten-mile-high view of the HTS/HCS process, *i.e.* from hypothesis creation to knowledge creation and sharing. The chapter describes the step-by-step actions to be undertaken to perform the screening process. Through a case study that was conducted at several Screening Facilities, several expert interviews were carried out. The interviewees include technicians, postdocs and PIs. The main output of this chapter is an extensive overview modeled using a Process Deliverable Diagram (PDD) to structure the many identified aspects to improve the screening process. Example improvements include the lack of an architecture supporting functionality, manual data enrichment, and deficient software for successfully gathering data from various external sources.

Chapter 3: HC StratoMineR: A Web-Based Tool for the Rapid Analysis of High-Content data sets

Chapter 3 describes a comprehensive study that includes an extensive HCS Data Analytics process tested, evaluated, implemented, and validated. The implementation is called HC StratoMineR. This is a software platform in which biologists can analyze their own data without the help of a data scientist or bioinformatician. The software is a web-based tool for the analysis of High Content data derived from plate-based experiments. The software is evaluated using a genome-wide siRNA screen and a compound screen (including small molecules). The results of the siRNA screen are evaluated by enriching the hits through an external ontology. The results of the compound screen were evaluated by (i) looking at the similarity of the phenotypic profile using a cosine vector score and (ii) a chemical profile using a Tanimoto score. Also, (iii) the images were inspected for expected profiles and finally (iv) the clusters were inspected for chemicals with similar structures and similar mechanism of action. Note that research questions 2a and 2b are both answered in Chapter 3.

Chapter 4: Improving Comprehension Efficiency of High Content Screening Data Through Interactive Visualizations

Chapter 4 is a study that investigates the use of interactive visualizations. Here a prototype is built to visualize data in an interactive and non-interactive fashion. An experiment was conducted among 79 students divided into two groups. Participants in both groups were instructed to search for specific facts in the data. Time and accuracy were measured for both groups. The results show that the interactive group performed better in both time and accuracy.

Chapter 5: Combining supervised and unsupervised data analytics methods for phenotypic screening

Chapter 5 introduces a supervised approach for the analysis of HCS data. First unsupervised techniques are used to annotate parts of the data. Then, supervised methods are used to enhance the results compared to only using unsupervised techniques. The approach is demonstrated using an evaluation study with results that are more relevant and useful for finding novelty using supervised methods. A study for finding genes that are involved in the process of mitotic cell cycle was conducted. Using supervised techniques, it was possible to specify groups of genes that are extremely close to the profile of the Mitotic Cell Cycle (MCC) class. This was verified by two external ontologies, i.e. String-DB and GOrilla.

Chapter 6: PurifyR: An R Package for Highly Automated Reproducible Variable Extraction and Standardization

Chapter 6 describes a developed R package to automatically preprocess numeric data. Most of the data preprocessing work must be done by hand. Preprocessing data is unfortunately not the sexiest part of data science. Therefore, it is decided to come up with a workflow including default options that can be adapted but more importantly, implemented in an R package that can automatically preprocess a complete data set. Several data sets were evaluated, and the results demonstrate that a complete data set up to a few gigabytes is preprocessed and ready to be analyzed in seconds.

Chapter 7: The glucocorticoid mometasone furoate is a novel FXR ligand that decreases inflammatory but not metabolic gene expression

Chapter 7 demonstrates a showcase of the entire Data Analysis Process of HC StratoMineR by means of screening small molecules in a low content fashion. The screen in this chapter investigates an FDA approved library of 1200 drugs. The main goal in this study is to find drugs that activate NF-κB that do not show signs of toxic effects. The screening data contains two features and the study includes a similar approach for hit picking as the approach described by Young et al., 2008, Omta et al., 2016 and Caicedo et al., 2017. The results show that drugs were found that significantly inhibit NF-κB transcriptional activity. Drugs that demonstrate low levels of Renilla were excluded from the initial hit list because this is a strong predictor for drug cytotoxicity. The remaining drugs were included for follow-up studies.

From a process perspective

To further clarify the chapter relationships within this dissertation, this section describes the structure from a KDD perspective, as shown in figure 13.

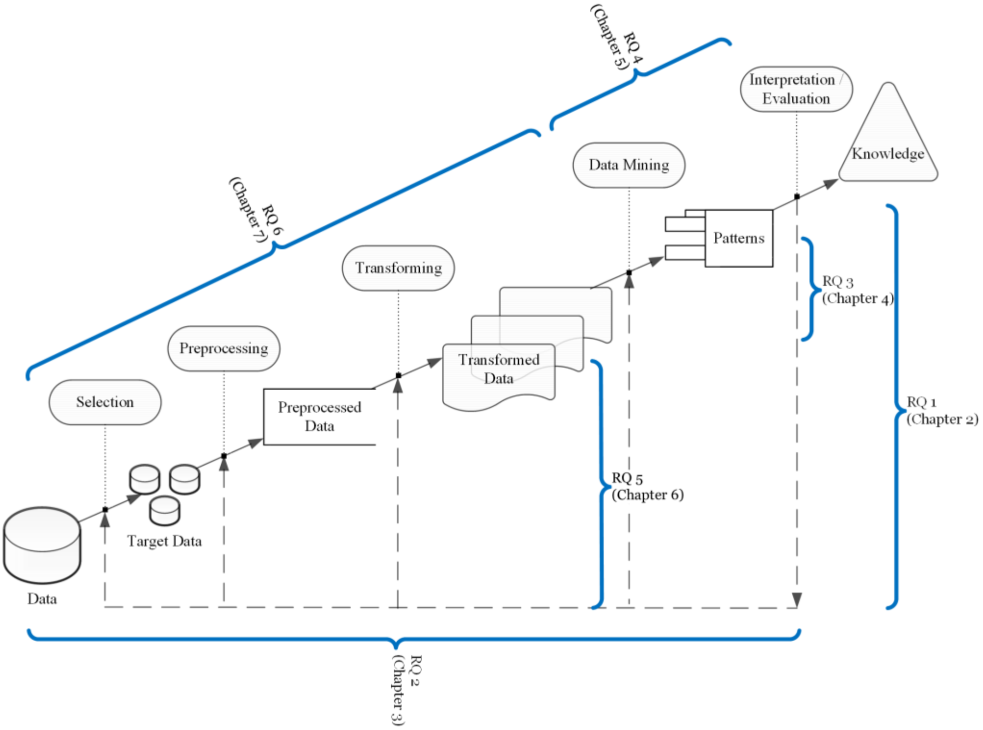


Figure 13. *The focus of the chapters from a KDD perspective*

Chapter 2 is a broader research question regarding the workflow of HCS and goes even further than the KDD model because it also covers aspects that are directly related to the processes involved before the data was created.
Chapter 3 is also a broad study that covers all the KDD phases. This chapter discusses the design and implementation of the data analytics process for HCS using unsupervised methods. The chapter also includes two validation studies, i.e. a chemical screen and a genome-wide siRNA screen in which the data sets are analyzed using the implementation. Furthermore, the evaluation and knowledge that can be extracted from the results of the two analyses are

discussed in a biochemical context that points specifically towards the Knowledge node shown in figure 13.

Chapter 4 focuses on a smaller section of the KDD framework. This chapter is focused on the transformation and interpretation of using interactive visualizations and the effectiveness and performance when interactive visualizations are implemented.

Chapter 5 aims at a subsection of the KDD process where the effectiveness of using supervised techniques in combination with unsupervised techniques is compared to using unsupervised techniques alone. This chapter also includes a validation study. The results and interpretation are discussed in a biological context.

Chapter 6 aims at the beginning and the most important part of the KDD framework, the preprocessing of numeric data in data science. A standard protocol is provided, in order to apply the preprocessing of numeric data in an automatic fashion.

Chapter 7 almost covers the complete KDD framework but is limited to a single parameter approach (low content). The study is a showcase for the methods introduced in chapter 3 on an FDA approved drug screen library of 1200 drugs.

From a data perspective

Data sets used in this thesis are image sets and numeric data created by automated microscopes. The data sets are used for validation to demonstrate various steps, methods and techniques as proof of concept for a viable way of working within the domain of HTS and HCS. All data sets used in this thesis are described in table 1.

From a methodological perspective

Concluding, the overview in figure 14 visualizes this thesis from a methodological perspective. The chapters are described in three facets: system components, research methods and publications. The system components describe the cognitive artifacts from the Information Systems (IS) viewpoint. The research methods are roadmaps, ways of working, libraries or tools and techniques that can be used to design, validate and evaluate cognitive artifacts, prototypes, theories and scientific results. The third concept, publications, can be considered one of the most important facets described in this section, because this part of the research remains available till the end of time. Therefore, a scientific publication should be written in a transparent manner so that experienced researchers in the field can reproduce the experiment described. There is a vertical axis describing a definition study (a ten-miles-high-view study), then Design and implementation describing the core of this thesis, followed by two studies devoted to optimization. Finally, a generalization study, which is not limited to HCS but focused on data science in general.

This means that supplementary data including SOPs, experimental data, methods, tools, techniques including version numbers are supplied together with the publication, preferably in an open source manner. Furthermore, the publication should be accessible for everyone regardless of income, social class, origin or race meaning open access. In this way, knowledge can be shared with those who seek to enrich themselves with knowledge and expertise financed by public money.

Table 1. *This table demonstrates the size of the HCS data sets that were used in this research*

Chapter	RQ	Numeric Data Size	Image Data Size	# Records	# Features
2	RQ1	NA	NA	NA	NA
3	RQ 2a RQ 2b	35 GB 1GB	438 GB 4 GB	57 mln 1.2 mln	74 58
4	RQ 3	77.4 GB	1.28 TB	96.8 mln	158
5	RQ 4	70 GB	876 GB	114 mln	74
6	RQ 5	280 MB 454 MB 6.9 GB	3.5 GB 5.3 GB 8.8 GB	0.4 mln 3.5 mln 0.251 mln	74 233 1787
7	RQ 6	10 KB	NA	13824	4

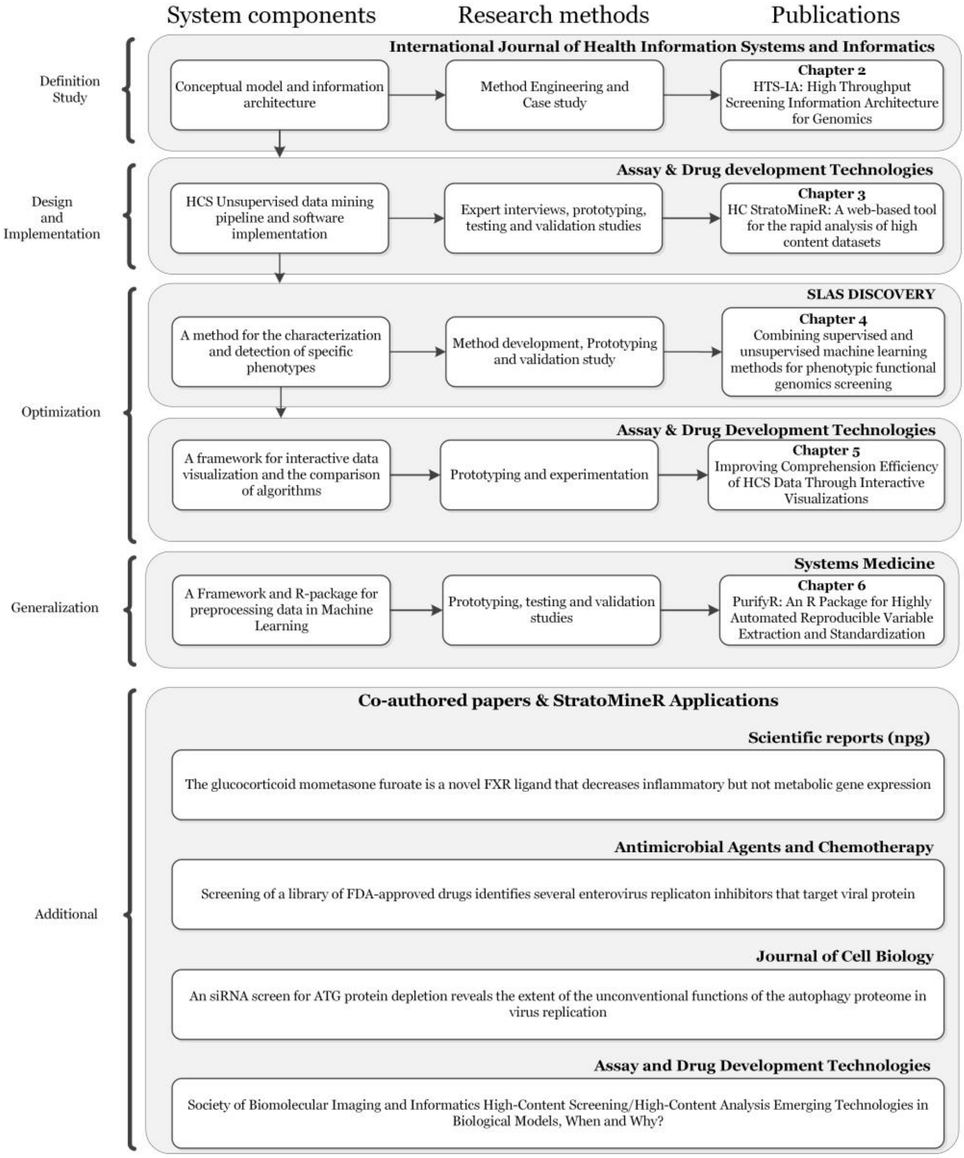

System components
Research methods
Publications
Definition Study
Design and Implementation
Optimization
Generalization
Additional
International Journal of Health Information Systems and Informatics
Conceptual model and information architecture
Method Engineering and Case study
Chapter 2 HTS-IA: High Throughput Screening Information Architecture for Genomics
Assay & Drug development Technologies
HCS Unsupervised data mining pipeline and software implementation
Expert interviews, prototyping, testing and validation studies
Chapter 3 HC StratoMineR: A web-based tool for the rapid analysis of high content datasets
SLAS DISCOVERY
A method for the characterization and detection of specific phenotypes
Method development, Prototyping and validation study
Chapter 4 Combining supervised and unsupervised machine learning methods for phenotypic functional genomics screening
Assay & Drug Development Technologies
A framework for interactive data visualization and the comparison of algorithms
Prototyping and experimentation
Chapter 5 Improving Comprehension Efficiency of HCS Data Through Interactive Visualizations
Systems Medicine
A Framework and R-package for preprocessing data in Machine Learning
Prototyping, testing and validation studies
Chapter 6 PurifyR: An R Package for Highly Automated Reproducible Variable Extraction and Standardization
Co-authored papers & StratoMineR Applications
Scientific reports (npg)
The glucocorticoid mometasone furoate is a novel FXR ligand that decreases inflammatory but not metabolic gene expression
Antimicrobial Agents and Chemotherapy
Screening of a library of FDA-approved drugs identifies several enterovirus replicaton inhibitors that target viral protein
Journal of Cell Biology
An siRNA screen for ATG protein depletion reveals the extent of the unconventional functions of the autophagy proteome in virus replication
Assay and Drug Development Technologies
Society of Biomolecular Imaging and Informatics High-Content Screening/High-Content Analysis Emerging Technologies in Biological Models, When and Why?

Figure 14. *Methodological view of this dissertation*

This figure provides a methodological view on the process and output of this research by providing insight into how the created artifacts are related to the used research methods and how the research results into a journal publication.

Chapter 2 - HTS-IA: high throughput screening information architecture for genomics

This paper describes a high throughput screening architecture for functional genomics screens that use high content methods. Case studies were performed using the Yin case study approach. Additionally, a detailed process model is provided using a Method Engineering approach. This study shows that current information architecture lacks interchangeability and functionality. Data enrichment is carried out manually, and software is still deficient in terms of interoperability to be able to successfully gather data from various external sources. This begs for the growing need of a real integrated laboratory information management system both in academia as well as small-to-medium-sized commercial organizations. Current solutions are designed primarily for clinical samples and lack functionality for larger libraries. A solution should give users the ability to create data pipelines that allow processes to be easily reflected in a relational database.

Adapted From:

Omta, W. A., Egan, D. A., Klumperman, J., Spruit, M. R., & Brinkkemper, S. (2013). HTS-IA: high throughput screening information architecture for genomics. *International Journal of Healthcare Information Systems and Informatics* (IJHISI), 8(4), 17-31.

INTRODUCTION

It simply is not practical, and would be error-prone, to investigate a large quantity of reagents manually (Persides, 1998; Allan et al., 2012). High Throughput Screening (HTS) is a process in which large libraries of chemical or biological reagents can be tested for activity in assays using automated methods (Allan et al., 2012). Certainly, an intensifying domain for bioinformaticians, HTS may now be conceived as an essential experimental instrument for the analysis of many biological processes. Experiments are conducted using 96, 384 or 1536-well microplates where cells or proteins and libraries of reagents are added according to a specified protocol. Reagents are stored in microplates or in individual 2d-barcoded mini-tubes.

High content analysis is a subset of HTS in which images of cells are acquired using automated microscopy. Subsequently, automated image analysis can be used to generate multi-parameter numerical data. Thus, screens generate large amounts of captured data, which require further analysis for the identification of germane outliers, or hits (Pelz, Gilsdorf & Boutros, 2010). HTS is widely used within multiple disciplines including drug discovery, functional genomics and toxicology (Johan et al., 2004; Bajorath, 2002; Maurer, 2000). Functional genomics screens frequently make use of RNA interference technology (RNAi) that allows for the specific reduction or “knock down” of the function of individual genes. Insights can often be gained where domain data sets are innovatively structured and the integration of various data sets is instigated (Johan, 2004). For the current presentation, we will focus on the use of HTS for RNAi screens combined with high content analysis.

Figure 1 depicts a generic model for reagent processing leading to the creation of assay plates that are used for RNAi screens. Oligo stock implies 96 well-plates containing siRNA oligonucleotides bought by a research institute and delivered by a manufacturer such as Thermo Scientific or Ambion. These plates must be processed in order to be useful for screening. In the pooling step, four oligonucleotides that target the same gene are combined in one well. Briefly, one well actually tests four different oligonucleotides. At the same time the single oligonucleotides are also individually stored in oligo pick plates.

A working master plate contains 384 wells and has the same dimensions as a pooled stock or oligo stock. The use of 384-well plates greatly eases the handling of libraries as it reduces the number of plates 16-fold. As well, pooled oligonucleotides are stored in 2D barcoded mini tubes. These can be cherry picked for individual hit confirmation and the generation of smaller focused screens. siRNA experiments are generally performed at nM concentrations (10−9 mol/dm3). The pooled stock is stored at µM concentrations (10−6 mol/dm3). Thus, an intermediate dilution plate is required to decrease the concentration to 100nM before the creation of assay plates. An assay plate is used for experiments and once the data is derived and analyzed, significant outliers or "hits" are selected. As aforementioned, hits are confirmed using the mini tubes. After the confirmation screen, the individual oligonucleotides are picked from the oligo pick plates to verify the activity associated with each of the four. This is called deconvolution and it provides essential information for the evaluation of hits.

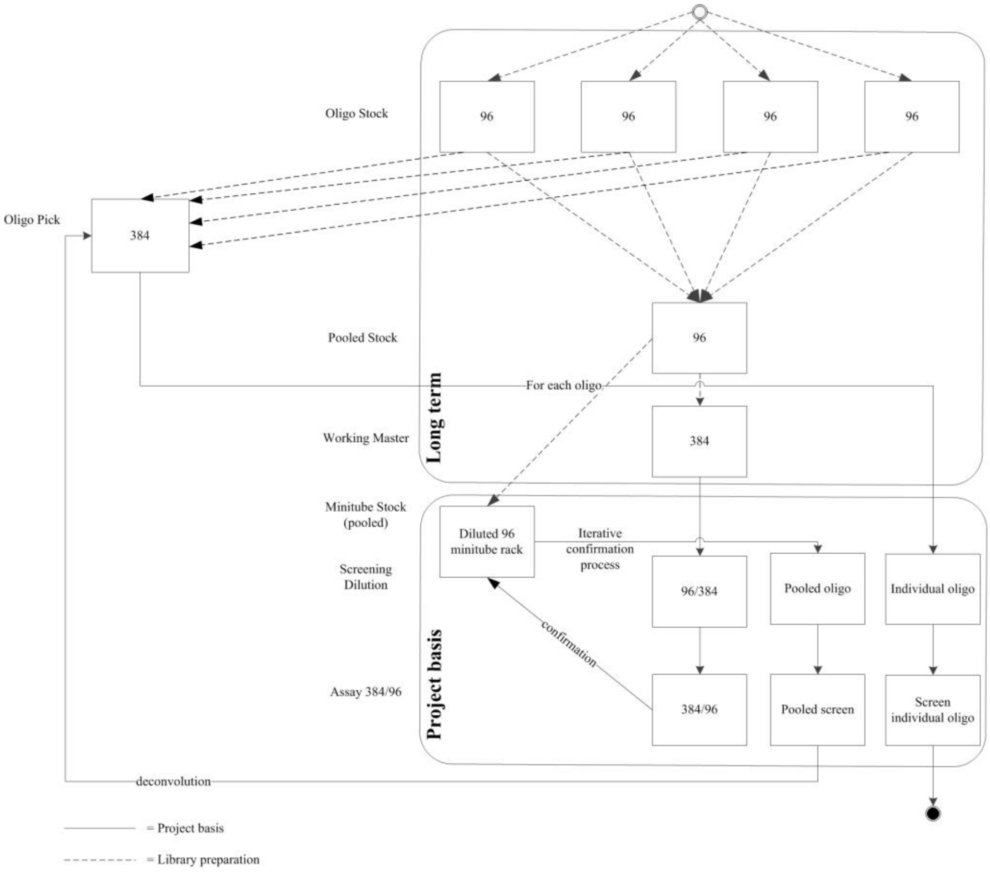


Figure 1. *Generic model of reagent processing.*

Problem description

Scant literature exists on how knowledge may be captured efficiently using flexible IT in an academic setting to support high throughput processes for drug discovery and functional genomics. Currently, there is a lack of knowledge on information flows and how the HTS system architecture can be optimized. Questions include: Are the software packages fulfilling the needs of an HTS facility? To what extent is current software interoperable or ready for seamless communication? Finally, how is the application of software efficient in terms of automation, consistency, redundancy, time utilization and usability? Put together, this research contributes to the functional and technical knowledge gap by investigating the following overarching question: 'What is the most efficient information architecture (IA) for HTS?'

In term of paper organization following this Introduction, the Background Section will discuss previous work conducted on Laboratory Information Management Systems (LIMS), normalization tools, and screen data management systems with an emphasis on the need for a proper IA. After this, the Yin's (2003) case study method will be heighted and adopted to describe how the research question was investigated. Then, the case study results are presented. Finally, we conclude with general remarks and several directions for future research in this field.

Background

Nowadays, for the purpose of efficient research collaboration and cost control, processes such as quantification, normalization and laboratory information management or statistical analysis will usually be supported with the use of "open source" software. A limited body of literature has provided an overview of those commonly used applications within HTS facilities as will be discussed below. A key problem for laboratories engaged in a wide range of high throughput activities is the shortage of cohesive and easy-to-use solutions (Pelz, Gilsdorf & Boutros, 2010). Van Rossum, Tripp & Daley (2010) argued that a high throughput facility needs a sophisticated informatics infrastructure to support high volumes of interleaved screening projects. Additionally, it is argued that technologies and design techniques that are anticipated to support rapid adjustments of software are also important supporting factors (Tolopko et al., 2010).

Laboratory Information Management Systems (LIMS) are required for the collection, viewing and editing of experimental information. Genetic studies, for example, require high throughput and large size sample sets. These samples are usually gathered from different sources and time points (Bajorath, 2002). This calls for appropriate and sensitive handling; otherwise it can lead to amplified errors, decreased accessibility and low efficiency of data. A LIMS is a solution for these issues. SLIMS, an open source LIMS, will be able to cope with information capturing patient samples (van Rossum et al., 2010). However, selected software contains functionality that may be useful and can be supportive in analyzing biological samples and the plates, boxes or mini tubes that are used for physical storage of the samples. These are tools for managing what is where. Biology-Related Information Storage Kit (BRISK) contains tools that support researchers to consolidate their data (Tan, Tripp & Daley, 2011). Furthermore, BRISK allows users to test and discover data in advanced analysis. It operates as a hub for accessing their data. The significance of this problem is growing due to the collaborative work philosophy and the need to share data in academic science. Therefore, demands in databases, storage, retrieval and communication are changing rapidly (Tan, Tripp & Daley, 2011). Problems arise where access to certain data sets needs to be restricted. When depositing data, users as well as data administrators should be able to decide who will be authorized to access what data. Data integrity can be safeguarded by using a multi-tier login system with well-defined user rights (Tan, Tripp & Daley, 2011).

Unlike SLIMS, Screensaver, another “open source” LIMS (Tolopko et al., 2010), were designed for the management of patient sample collections and specifically designed for the management of HTS. Screensaver serves different users, including students, post-doc’s and managers. The software offers a Library Information Management System (LibIMS) and stores experimental results, screening workflows and meta-data from screens. This allows for cross-screen comparisons, including support for reagent cherry picking and heat maps for data visualization through a web-based interface.

Web CellHTS2 is a web-based application for analyzing high-throughput screening data from RNAi and compound screens (Pelz et al., 2010), implemented in R/bioconductor (Ihaka & Gentleman, 1996; Gentleman et al., 2004). Quantified data can be uploaded using the graphical user interface (GUI). Both sample and control-based normalization and predictive analysis (e.g. B-score or Loess regression) are saved as HTML and text files are zipped and then sent by e-mail. The application runs on a JAVA webserver using AJAX technology. R-serve is used to interact among R and the JAVA application.

Omero, another open-source software, is used for managing large-scale image storage, meta-data and non-image data developed in Python, JAVA and C++ (Allan et al., 2012). Omero comes with its own application programming interface (API) and offers a web-based interface. Data can be uploaded using Bio-format, which enables data transformation for >100 different file formats to a common data model. Then, Omero stores the data mapped onto the Open Microscopy Environment (OME) data model (Goldberg et al., 2005). Omero supports interactions with third-party software e.g. Mat-Lab and CellProfiler using a JAVA gateway or the API.

CellProfiler, an application for extracting numerical data from microscopy images, applies automated image analysis (Jones et al., 2008; Kamentsky et al., 2011; Lamprecht, Sabatini & Carpenter, 2007). Cell profiler 2.0, also an “open source” object-oriented stand-alone application, is built for Windows, Linux and MacOS. A plugin is integrated to support a pipeline for ImageJ. Furthermore, CellProfiler supports a large extent of image formats via the Open Microscopy Environment (OME).

Method

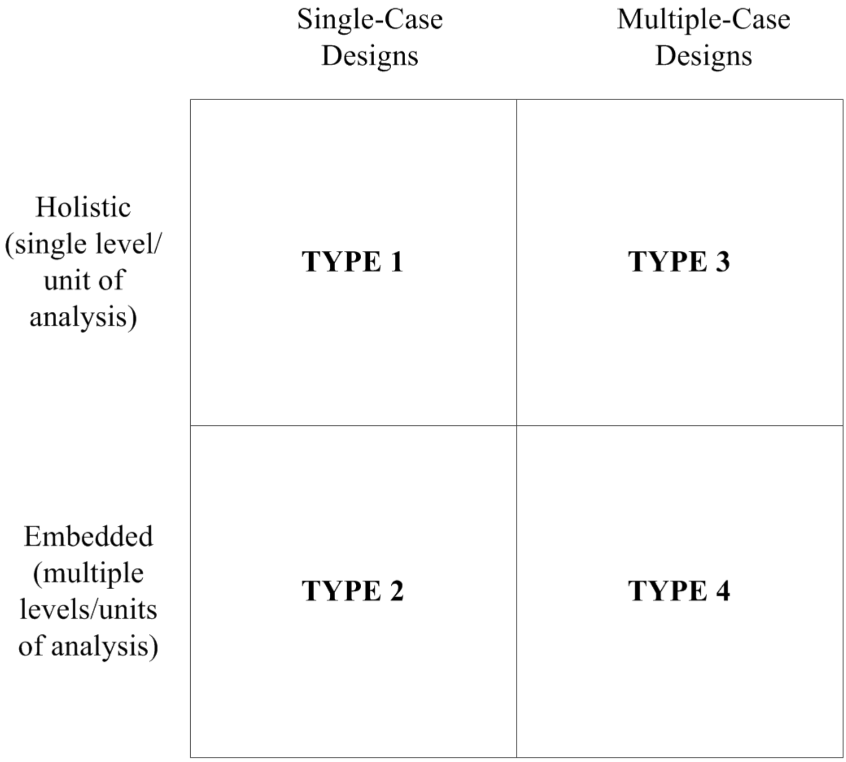
Yin (2003) discussed three types of case study methods: explanatory, descriptive and/or exploratory. Explanatory studies may be used in causal investigations, exploratory studies are most suitable within social sciences and information technology. The nature of this case study is exploratory, a methodology with four iterative stages: designing a case study, conducting the case, analyzing the case study evidence and developing the conclusions, recommendations and implications. In addition, Yin (2003) described four different categories of case studies in a 2 x 2 matrix (figure 2). The horizontal dimension outlines a single or multiple-case design. The vertical line sketches a holistic or embedded approach. In this sense, the HTS-IA study would best fit into type three, which implies a holistic or single unit of analysis and infers multiple case designs.

Additionally, Yin (2003) recommends four validity criteria for empirical research. Construct validity, which means that the measured concept is measuring what should be measured in the right manner (Jansen & Brinkkemper, 2008). Internal validity is defined as forming fundamental associations and avoiding false associations. External validity is the establishment of the field where the study findings can be generalized. Finally, the empirical reliability implies that the same results can be found by repeating the study.

Construct validity is tackled by using multiple sources of evidence, which may be achieved by re-routing the case report back to the interviewees. Feedback would then be provided, including reports and presentations on the IA and process workflows. In this way, the content as well as validity of captured constructs were validated. Internal validity is used only in causal or explanatory case studies. In this exploratory case study, this sort of "internal validity" checking is not needed or applicable. As well, given that the cases here are found to be representative for performing RNAi screens, automated microscopy and robotic liquid handling, the external validity is therefore covered. Taking empirical reliability into account, we would expect that the results we found would be consistent when repeating this study even though, as time goes by, enhanced technology might increase the efficiency of an HTS facility.

Importantly, our attempt here is to focus on the IT architecture of academic HTS facilities. A qualitative analysis is performed where semi-structured interviews at the Dutch Cancer Institute, Leiden University Medical Center, the German Cancer Research Center, the European Molecular Biology Laboratory and the University Medical Center Utrecht were conducted to investigate the needs of a high throughput screening facility in terms of informatics architecture. The interviewees were mostly experienced managers, IT personnel, researchers or Ph.D. students. For the semi-structured interviews, a schematic overview of the HTS architecture (figure 3) was provided to the participants so that they can see and tell how their HTS-IA differed from the envisioned architecture, which was modeled in advance.

Essentially, we asked the heads of HTS facilities to provide information on the IA and workflow of HTS. They pointed out a researcher with the right prerequisite knowledge. Again, this researcher was contacted by email with the same question. An appointment was made for a conference call via Skype or visiting them for a face-to-face interview. Notes were made during the interview and the semi-structured interview schedule provided as guidance. Additional questions were prompted when and if something was unclear. After the interview, the notes were sent by email to the respective participant for validation. Finally, upon receiving feedback, it was promptly processed.



	Single-Case Designs	Multiple-Case Designs
Holistic (single level/ unit of analysis)	**TYPE 1**	**TYPE 3**
Embedded (multiple levels/units of analysis)	**TYPE 2**	**TYPE 4**

Figure 2. *Basic Types of Designs for Case Studies. (Adapted from Yin, 2003).*

Results

Table 1 shows the development stage of the HTS facilities that were studied. The reason for the selected factors is provided in Table 1 is because, according to the interviewees, these variables seem to be the most important aspects of a LIMS. It should be noted that all the interviewed facilities carry out high throughput RNAi screens using high content assays.

Table 1.
- means working with flat files e.g. plain text or Excel sheets
+/- means semi-structured, working with an application that is supportive but still requires adjustments
++ means structured which implies working with integrated software that supports the communication of information in an interoperable and compatible fashion

	Image storage	Reagent Management	Screening Management	Data Analysis	Multi-parameter data
Facility 1	++	+/-	-	+/-	-
Facility 2	++	++	+/-	++	++
Facility 3	++	++	+/-	++	++
Facility 4	++	-	-	-	NA
Facility 5	++	-	+/-	+/-	+/-

A standard architecture for HTS was derived and is depicted in Figure 3. Performing an RNAi screen, images are generated by automated microscopy. Then, a transformation from images to raw data is required and carried out accordingly. Basically, this involves feature extraction from the images such as intracellular spot number, nuclear area and the distribution of intensity. In some facilities, quantification is embedded in the microscope, which is served by built-in software. Some facilities manage quantification using external software because they desire bespoke analysis algorithms. The quantified data can be managed in a screen data management system. This system provides meta-data such as technical information and information concerning the screen for example when, who and what.

A LibIMS provides information concerning the reagent library. In the case of an RNAi screen, it will link a gene name with a RNAi reagent and provides the location of this reagent in various microplates. Volumes, concentrations and other information are also tracked. All facilities perform multi-step physical library management processes on their reagents before performing an assay (an example is depicted in Figure 1). The system needs to deal with plate pooling, replication, dilution and compression from a 96-well format to 384-well or 1536-well format. In the case of screening with pooled reagents, it needs to assist in deconvolution of multiplexed experiments.

As a more advanced system, the LIMS not only has to deal with managing reagents, but also must play a more centered and integrated role. A true LIMS stores not just library information (reagents), but also raw and normalized data-associated project information concerning personnel and assay protocols. Such an advanced LIMS is often connected to the image database and usually includes an advanced user management system. Nowadays, heat maps can be applied to show where the normalized screening results are projected on the associated well and reagents. As well, the related raw data and images are linked. The last step in the interviewed facilities was normalization or statistical analysis, which was generally done by using R or CellHTS2. The application can provide predefined normalization techniques, including median, B-score, loess regression or robust local fit regression (Boutros, Bras & Huber, 2006; Malo, Hanley, Cerquozzi, Pelletier & Nadon, 2006; Birmingham et al., 2009).

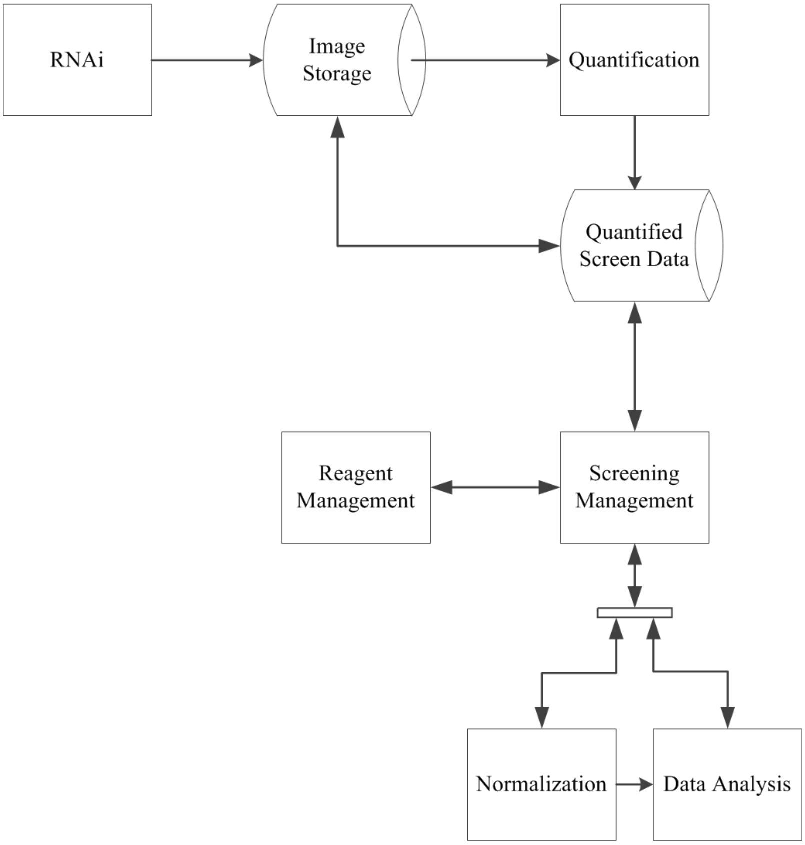
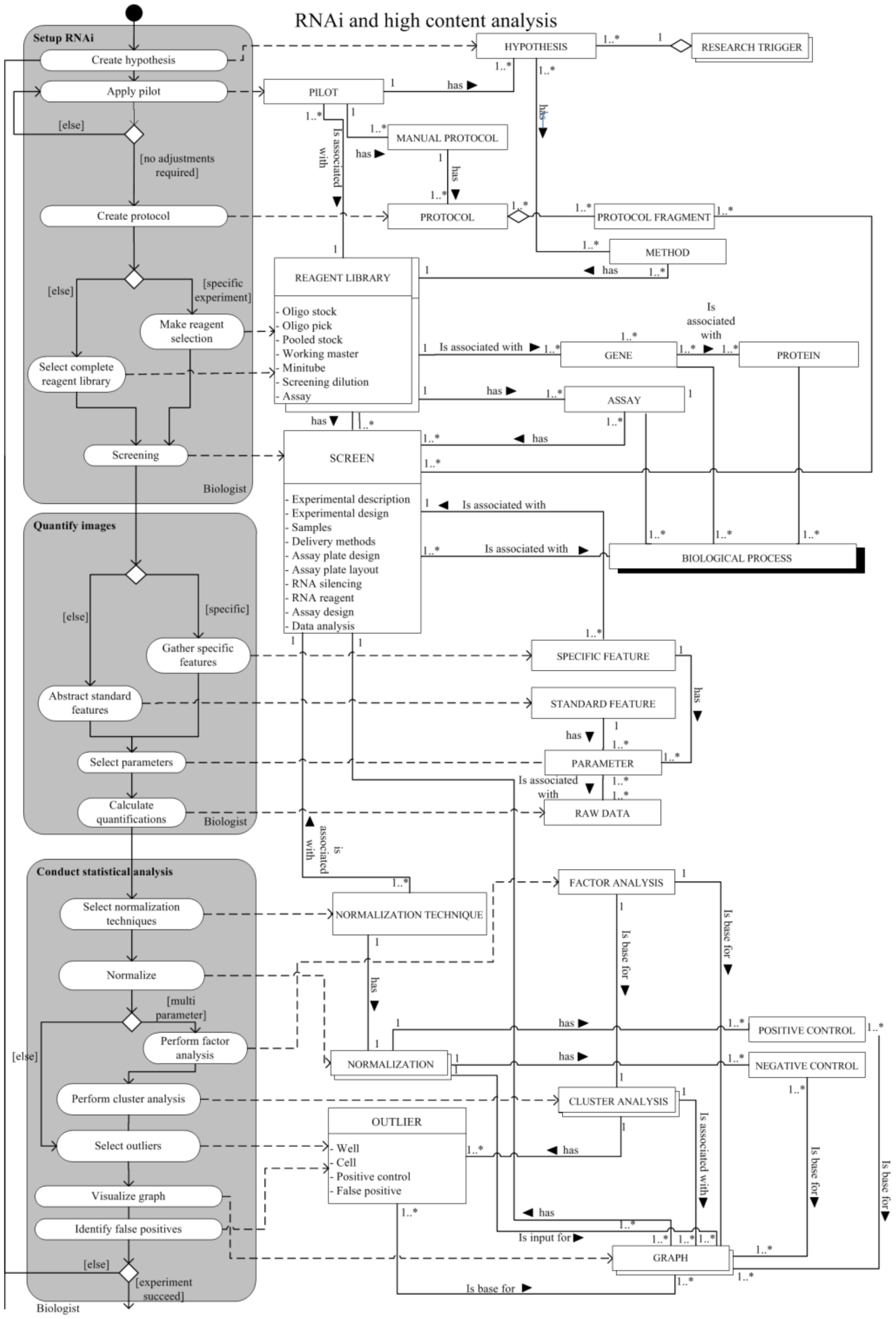


Figure 3. *High throughput screening architecture.*

Otherwise, R or Mat-Lab can be useful tools for non-standard normalization techniques or data analysis. CellHTS2 lacks an option for normalizing multi-parameter data at this time. R is competent to manage multi-parameter data, but it is a command-line application that would require strong prerequisite knowledge in the R-language. In addition, the results of CellHTS2 are offered in HTML format but lack in structure to make further analysis easy. Therefore, a database in which the results are stored in a regulated manner is required for the enrichment of data or additional data analysis. Figure 4 outlines a more detailed process analysis using the Method Engineering modelling approach (van de Weerd & Brinkkemper, 2008).

Following the statistical analysis, the results can often be further enriched using public chemical and biological repositories (figure 4). New knowledge can also be extracted using other methods, for example, pathway analysis. Currently this sort of follow-up analysis is only done manually. Even so, automated enrichment requires very structured data, stored in a semantic manner. This presents a substantial challenge in current bio-informatics practices. The last step of the HTS-IA case study is a reporting phase where tools should be used to clearly visualize findings. This simplifies the sharing of knowledge and provides input for publications. It was observed during the interviews that business processes are not often clear to key users, much less so to the managers. The Process Deliverable Diagram (PDD), work processes bifurcate in a number of places. Hence, it is important that both IA and software should be adapted in order to take these facts into account.



RNAi and high content analysis

)

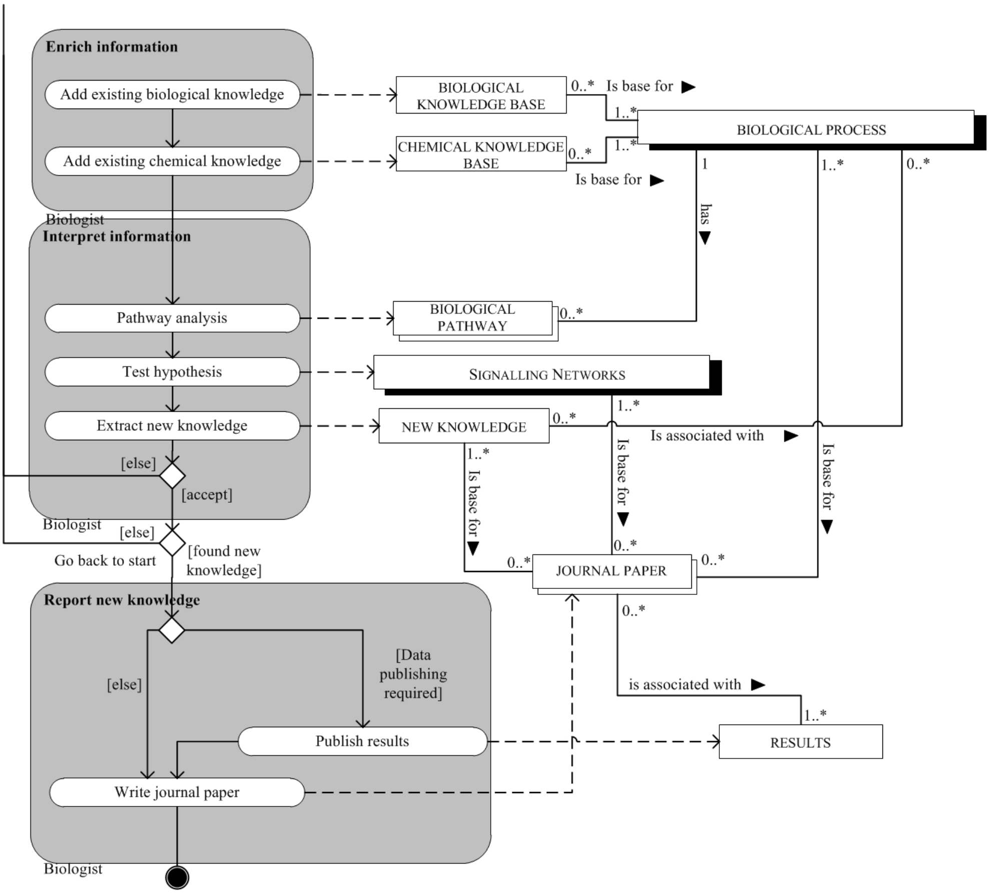


Figure 4. *Process Deliverable Diagram (van de Weerd and Brinkkemper, 2008).*

Conclusion

From an information science perspective, there is potential for improvements for the HTS-IA that is studied in this research. Advances in HTS architecture can be identified by process modelling of workflows. A great variation in the quality of architecture has been found in the different case instances investigated. Indeed, emerging facilities that are served by bioinformaticians have built tools in-house for their own needs. Many of these facilities have implemented LIMS along a central and integrated approach. There are, however, other facilities that have or do not possess documented business processes. Additionally, they do not have a solid HTS-IA and are struggling with the implementation of "open source" software.

There is a need for software with integrated and holistic approach in academia and small- to medium-sized commercial organizations. Currently, most available solutions have been designed for the management of clinical samples. Obviously, it is lacking much of the functionality that is needed in screening facilities that deal with libraries in larger scale operations. Activities such as reagent pooling, and 96 to 384-well compression are unavailable in many commercial solutions. The ideal method would be to integrate important processes into a single application. Within the application, each process may then be represented by a different module.

An important differentiating feature of new software would be the ability to create pipelines that allow processes to be easily reflected in a relational database. Another key differentiator would be the ability of adopted software to easily and efficiently handle mini-tube collections for hit-picking and custom library generation. Such software should also support high content data analysis.

Discussion

The ultimate goal within HTS-IA is that every piece of information from the output of a particular screen should be rapidly organized in an appropriate biological and/or chemical context so as to enable decision-making and the generation of new hypotheses. Traditionally, this has been done manually by bioinformaticians, but if the data is integrated with the appropriate ontology, it can be efficiently and effectively automated. Additionally, external data is easily added to the existing knowledge. In so doing, it will further enrich the processed data.

In a sense, all applications should be connected so that they can automatically export processed data from one application to another. Applications and steps in this process include automated microscopy, assay quantification, statistical analysis, Library Information Management (LibIM), data mining, merging of external knowledge, automated reporting as well as biological and chemical understanding. Furthermore, it is essential for scientists that the associated images or animated visualization are available in an on-demand fashion whether it concerns kinetic, confocal, multi-parameter, individual cell-level or movie experiments. Moreover, informatics workflows should be simple to build and accessible for re-use and modification. Historical insight is also desirable, so scientists can have a vision on what they have been doing, in which order and how it was performed. In a nutshell, aspects such as interoperability, flexibility, consistency as well as semantics are enormously important for enhancing High Throughput Screening in a more developed and matured stage as it is right now.

Finally, "open source" applications exist that can meet certain needs although none of these may be consecutive. Each application is designed with its own purpose, but the applications cannot be reciprocally interlinked. The consequence is working with flat files and using import and export functionality. This even might require adjustments before uploading, for instance, Perl scripting or recalculations by using R or Mat-Lab. Screening facilities are conducting statistical analysis using R, Excel or cellHTS2 on their data. However, none of the screening facilities in this study were performing data mining.

Most of the interviews revealed that reporting was done through publishing a journal paper. Some facilities used tools for reporting results such as Excel, Image J, Adobe Photoshop and R. Facilities at different stages of development were analyzed during this case study. This may have influenced the architecture or PDD that is offered. Also, in the data gathering, face-to-face conversations and Skype meetings were used. These might have affected on how answers were interpreted. The number of HTS facilities that were investigated in this research was limited to five. This analysis was projected primarily towards RNAi screening and therefore cannot be generalized to compound screening. However, many of the concepts are remarkably similar. Modelling of the IA and processes at a high conceptual level was found to be very useful as it makes it easier to identify gaps. In addition, interviews with vendors such as Thermo or FluidX could have provided valuable sights as well. Notwithstanding, security is a critical and sensitive issue, especially within research facilities that have been excluded from this study.

Solutions and Recommendations

i. As mentioned in the results, most HTS facilities do not possess a well-developed workflow of the HTS processes. Key users and decision-makers do not fully understand how information flows. Therefore, it is recommended for every HTS facility that there should be an up-to-date modeled business process available.

ii. Gene names, features and plate identifiers need to be consistent and compatible so that each software package (e.g. LibIMS, LIMS, Screening management and data analysis) can handle the captured data.

iii. A workflow management system is desirable. Such a system should be allowed to change flexibly. Also, the cognitive load on certain processes such as mini-tube handling may be overly extensive that human errors can be very easily made. In such instances, a system that guides users through the process would be helpful.

iv. LIMS should be the central mediator in managing all the information concerning a screen. LibIMS should be subset of LIMS.

v. We suggest a project ID to associate every piece of information including researchers, used reagents, plates, gene names, antibodies or compounds and images.

vi. Finally, as shown in Figure 4, we suggest a phase ID to identify which step is currently applicable within a project.

Future Research Directions

Table 1 provides a preliminary assessment in the development stage of academic HTS facilities in Europe. Future research should elaborate on the development stage according to the model of Steenbergen, Bos, Brinkkemper, Weerd and Bekkers (2010) and Steenbergen and Brinkkemper (2007). Then, a maturity model can be used to understand the situation on several facilities in a more detailed fashion. The model shows the current maturity stage of a facility, focused on important aspects such as enrichment, data mining or screening. A maturity study could provide a tangible path to incremental process improvement, all of which will aid to determine the current maturity level and which level is most desired (Bekkers, van de Weerd, Spruit & Brinkkemper, 2010).

HTS is used in several other fields besides functional genomics. Drug discovery, where large libraries of small molecule compounds or natural product mixtures are screened, entails applications with similar workflows. The types of biological systems in the assays can range from the use of purified proteins to microorganisms such as pathogenic bacteria. A recent innovation is the development of assays for the screening of organoid cultures. HTS can also be applied in genetic screens using model organisms such as yeast and zebrafish. All these applications share certain aspects of the workflow described here, as shown in Figure 4. Therefore, improvements in software supporting HTS processes are broadly supported by other disciplines as well due to commonalities in workflow and data management. Cooperation between the various disciplines could lead to further enhancement.

Appendix

1. Is your current workflow like that of the figure 3?
2. What is the number of applications used in the workflow from the start (automated microscopy) until the end (data mining) or interpretation of the data?
3. Are the various used applications compatible?
4. Are the applications web-based?
5. Is it possible to merge data from outside e.g. gene ontologies with the results?
6. What software is used for getting data out the automated microscopy?
7. What software is used for reagent management?
8. What software is used for data mining
9. What data-mining techniques are used?
10. What software is used for reporting?
11. At what point within the workflow could the process be more efficient, e.g. saving time?

Key Terms and Definitions

Compound Screening: The use of chemical libraries in HTS for drug discovery.

Functional genomics: The large-scale analysis of gene function using techniques such as RNAi, microarray analysis and deep sequencing.

High Content Screening: High Content Screening is a subset of HTS in which analysis methods are used that generate multiple numeric parameters for every sample tested.

High Throughput Screening: In High Throughput Screening biological assays are used in testing large numbers of chemical samples, or biological reagents using robotic systems. HTS is applied in many fields e.g. pharmaceuticals, cosmetic development and academic settings.

Laboratory Information Management System: A Laboratory Information Management System provides support in key elements of the biologist e.g. workflow management, data tracking, and reagent tracking.

Mechanism of action: The specific molecular mechanism that is used phenomenon. It frequently refers to the mechanism of drug function.

Pathway: In biology, a pathway is a series of biochemical interactions mediations frequently by proteins and protein complexes that control various processes such as signal transduction from the cell membrane to the nucleus.

RNAi: The use of small double-stranded RNA oligonucleotides for the specific knocks down of gene function.

Chapter 3 - HC StratoMineR: a web-based tool for the rapid analysis of high-content datasets

High content screening (HCS) can generate large multidimensional data sets and when aligned with the appropriate data mining tools, it can yield valuable insights into the mechanism of action of bioactive molecules. Easy to use data mining tools are not widely available however, with the result that these data sets are frequently underutilized. Here we present HC StratoMineR, a web-based tool for high content data analysis. It is a decision-supportive platform that guides even non-expert users through a high content data analysis workflow.

HC StratoMineR is built using MySQL for storage and querying, PHP as the main programming language and jQuery for additional user interface functionality. R is used for statistical calculations, logic and data visualizations. Furthermore, C++ and GPU power are diffusely embedded in R using the rcpp and rpud libraries for operations that are computationally highly intensive.

We show that we can use HC StratoMineR for the analysis of multivariate data from a high content siRNA knock-down screen and a small molecule screen. It can be used to rapidly filter out undesirable data, to select relevant data, to perform quality control, data reduction, data exploration, morphological hit picking and data clustering. Our results demonstrate that HC StratoMineR can be used to functionally categorize high content screening hits and thus provide valuable information for hit prioritization.

Adapted From:

Omta, W. A., van Heesbeen, R. G., Pagliero, R. J., van der Velden, L. M., Lelieveld, D., Nellen, M., Kramer, M., Y eoung, M., Saeidi, A.M., Medema, R.H., Spruit, M.R., Brinkkemper, S., Klumperman, J. & Egan, D.A. (2016). HC StratoMineR: a web-based tool for the rapid analysis of high-content datasets. *Assay and drug development technologies*, 14(8), 439-452.

Abbreviations

MySQL = My Structured Query Language
PHP = PHP: Hypertext Preprocessor
HPC = High Performance Computing
OGE = Open Grid Engine
GPU = Graphic Processor Unit
CPU = Central Processor Unit
IQR = Interquartile Range
MAD = Median Absolute Deviation
QC = Quality Control
SSMD = Strictly Standardized Mean Difference
NA = Not Available
NaN = Not a number
INF = Infinitive number
PAM = Partitioning Around Medoids
DNA = Deoxyribonucleic acid
CAD = cationic amphiphilic drugs
GB = Gigabyte
CFA = Common Factor Analysis
PCA = Principal Component Analysis
MDS = Multidimensional Scaling

Introduction

Life science researchers are increasingly drowning in their own data (Singh S, et al., 2013). Formerly only specialized groups were in a position to generate large complex data sets, but in today's highly collaborative and distributed research environment, far larger numbers of researchers can gain access to technologies that generate large volumes of data (Marx V, 2013). A major challenge for scientists is the handling and analysis of large data sets such that the data can be efficiently used to generate new knowledge. Almost invariably the development of analysis tools lags behind the technology that is generating the data. The result is that the biologist needs to collaborate with a specialist in order to analyze their data.

The problem is frequently compounded by the fact that analysis methods are often not addressed until after the data set has been generated. The lack of advanced data mining methods means that the amount of new knowledge generated is limited. In large scale screening experiments this is often reflected in scientists only identifying previously known hits, the so called “low-hanging fruit”.

High Content Screening (HCS), an innovative technology combines the use of automated liquid handling, automated fluorescence microscopy and automated image analysis, is a good example of a field that suffers from these problems. Multiple numerical descriptors of cellular morphology, (parameters or variables), are extracted during image analysis. The resultant multivariate numerical data sets can be mined to generate phenotypic profiles or fingerprints for each tested reagent.

It has recently been reported that 60-80% of high content screens use only one or two extracted parameters, despite the wealth of publications that have demonstrated the power of more advanced multi-parameter approaches (Singh S, et al., 2013). This suggests that while many groups are carrying out image-based screens, relatively few are doing real high content analysis and so are not taking advantage of the power of HCS.

There are several tools available to assist with HC data mining. HC Profiler which is marketed by Perkin Elmer is an adapter that allows for the porting of HC data to Tibco Spotfire, a well-established data visualization tool. Tableau is a commercial business intelligence software package that allows interactive plotting and on the fly calculations in a drag and drop fashion on various data sets. Dotmatics is an enterprise scale business intelligence tool for the analysis and visualization for life sciences research data and has a module designed to assist with HC data. Cell Profiler Analyst is an open source software package. It is an extension of the Cell Profiler open source image analysis platform and offers multiparameter data visualization.

At the Cell Screening Core, we needed a tool that biologists can use to mine their own high content data sets. We deemed the tools that were then available unsuitable due to cost, the requirement for a local installation, (not useful for off-site clients), or the fact that they required extensive training to use independently. For this reason, we have developed HC StratoMineR, the package described here. Like the widely used Web CellHTS2 platform, it gives users access to a web-based easy-to-use tool for data analysis (Pelz et al., 2010). Our workflow is partly based on a previously published method for high content data analysis (Young et al., 2008; Omta et al., 2013)

Materials and Methods

High content chemical screen to measure the disruption of lysosomal function

MCF7 cells were cultured in RPMI (GIBCO), supplemented with 5% fetal calf serum, 100 U/ml penicillin and 100 µg/ml streptomycin. 6000 cells per well were plated in 96-well plates. Four hours after compound addition Hoechst, LysoTracker-Red DND-99 (Molecular Probes Invitrogen Corporation) and 6µM Calcein-AM were added. Cells were incubated for 30 minutes and then imaged on a Thermo ArrayScan VTi. Image analysis was carried out using the Cellomics Morphology Explorer Bioapplication, (Thermo Scientific).

High content siRNA knock-down screen to identify novel regulators of mitosis

HeLa cells were cultured in DMEM (GIBCO), supplemented with 6% fetal calf serum, 100 U/ml penicillin and 100 µg/ml streptomycin. siRNAs were transfected using RNAiMax (Invitrogen) according to the manufacturer's guidelines. The human ON-TARGETplus siRNA SMARTpool library (Dharmacon) was used for the genome-wide siRNA screen which was performed in duplicate.

SiRNA libraries were aliquoted in 384-well plates. 1500 cells were added to the wells after incubation of the transfection reagents. After 48 hours of culturing, the cells were fixed using formaldehyde. After staining of the wells with primary and secondary antibody, the mitotic index of the wells was analyzed using a Cellomics Arrayscan VTi (Thermo Scientific). Image analysis was performed using Cellomics Morphology Explorer Bioapplication (Thermo Scientific).
Data was exported using HCS Explorer (Thermo Scientific). Data at cellular level for all available channels was exported in flat text files, one file per assay plate.

Implementation and Architecture

HC StratoMineR is a web-based platform that guides the user through a HC data analysis workflow. It is built using MySQL for data storage and querying, PHP for the front and back-end, and BOOTSTRAP with extra jQuery libraries for additional user interface functionality. Statistical calculations, the logic and data visualizations are managed by R. Specific computationally intensive steps are calculated with the use of the rcpp, rpud and compiler libraries, so that C++

code, GPU power or compiled code can be used to accelerate these processes (FasteR! HigheR! StrongeR! Retrieved from http://www.noamross.net/blog/2013/4/25/faster-talk.html). Multiple instances can be run in tandem. The implementation is designed to handle processes in parallel using the snow package. Using the Linux bash scripting language, HC StratoMineR was designed for the submission of very large high content data sets to a high-performance computing (HPC) cluster running the Open Grid Engine (OGE) so that data can be more efficiently processed. The OGE works most efficiently when each iteration is submitted as an independent job. A list of jobs is distributed by a queuing server (member of the HPC) which distributes the jobs to available threads, (server cores). In each thread a specific part of the data is queried using MySQL and independently processed in R. The web application waits until the last thread has finished and then refreshes the page to present the results. A technical visualization of the architecture is included in figure 1 as well as detailed step-by-step visualization of the workflow, (Supplementary Figure S1). A standard operating procedure (SOP) is provided in the Supplementary Materials to guide a user through the analysis of an example high content data set, (also provided, see 'Links' Supplementary Materials). A detailed description of the HC StratoMineR workflow is given below.

Data Upload

The user logs in to an SSL secured website https://cla.stratominer.com with a username and password while accepting the Terms and Conditions. A new experiment is then created, or the user can go directly to a step in the workflow of a previous experiment. The number of assay plates and the number of replicates i.e. duplicate, triplicate etc. is defined. The system supports 96- and 384-well microplate formats. Data files can be uploaded individually, in ZIP files or a link to a ZIP file can be provided, (see Links for test data in the Supplementary Materials). The data format requires the data from one cell to be in one row (record) in the case of cell level data; one field per row in field-level data; or one well per row in well-level data. One text file per microplate is required. The file structures can now be reviewed to check that they are in the correct format. HC StratoMineR can accept, tab, semicolon or comma separated files. The files are grouped by replicate. HC StratoMineR assumes that reagent replicates are placed on separate microplates. More details on the 'Data Format' can be found in Supplementary Materials.

Meta Data

Parameters/columns that contain metadata such as well location and barcodes are identified. Parameters that contain a unique numerical plate identifier and a human readable identifier are chosen. Additional information is also required i.e. plate format, default parameter and data resolution. The default parameter will be visualized first by default in subsequent steps.

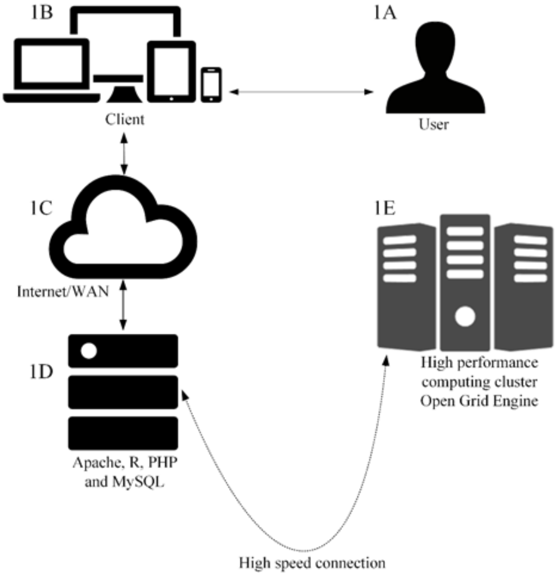


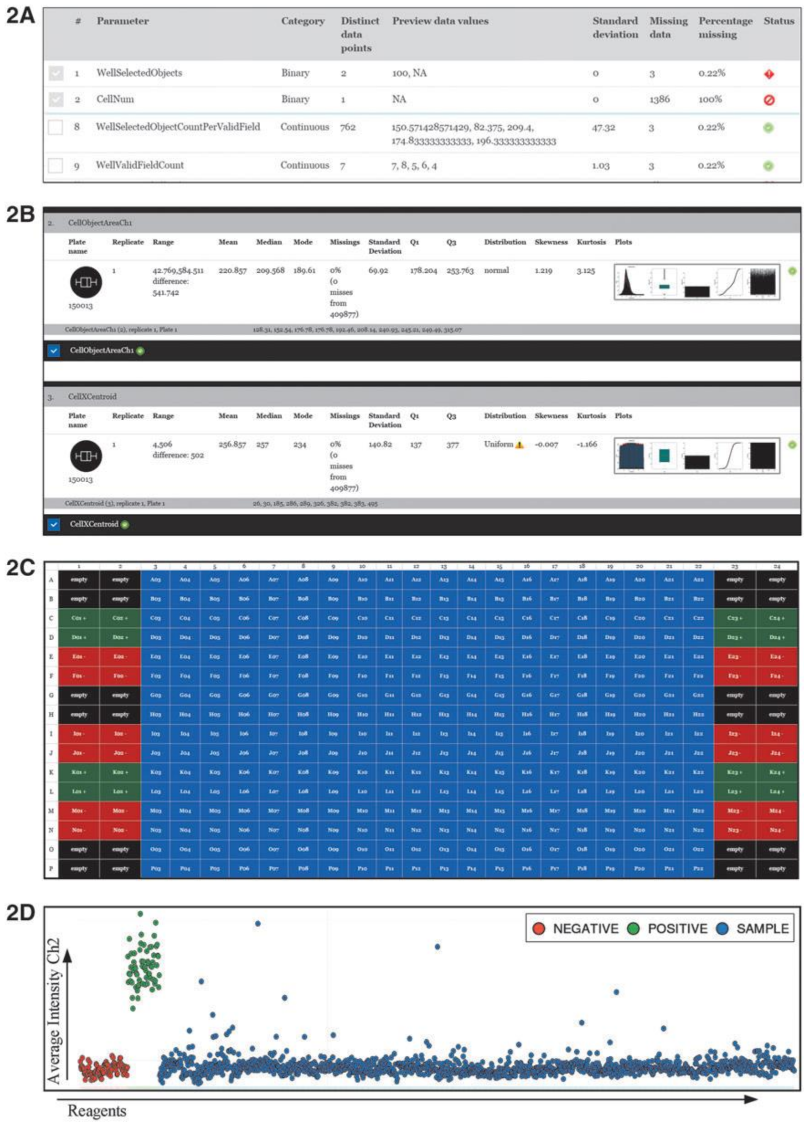
Figure 1. *The architecture of HC StratoMineR*
(A) The user, retrieving and sending information from HC StratoMineR. (B) The device (laptop, workstation, tablet, or phone) accessing HC StratoMineR via a browser. (C) The WAN, or internet connection that provides an SSL secure connection to the server running HC StratoMineR. (D) The webserver running HC StratoMineR by using Apache, R, PHP: Hypertext Preprocessor, and MySQL. (E) A local high-performance computing cluster using the open grid engine connected with the webserver by a high-speed fiber connection. SSL, Secure Sockets Layer; MySQL, My Structured Query Language; WAN, Wide Area Network.

Data Preprocessing

HC StratoMineR automatically highlights parameters that should be omitted. Also, the data category, for example, binary, discrete, and continuous is inspected. If a parameter is binary/discrete, has a standard deviation of zero, or contains ‡95% empty values (NULL), it is marked for removal. Each individual parameter is now set to the right data type (e.g., DOUBLE, INT, and TEXT), and the data are indexed for faster access. The data are checked for the correct number of replicates that was given during data upload. Then, the data are checked for empty fields and inconsistencies across replicates (Figure 2A).

Parameter Selection

Basic metrics such as the range, median, mean, and standard deviation are calculated for each parameter across every replicate and plate. A visualization provides a scatterplot, an error bar, a QQ-plot, a boxplot, and a histogram of each parameter (Figure 2B). HC StratoMineR calculates several parameter scores (estimators) for each well, based on the cells in a well, for example, inter quartile range (IQR), mean, trimmed mean, modus, median, and median absolute deviation (MAD).



	#	Parameter	Category	Distinct data points	Preview data values	Standard deviation	Missing data	Percentage missing	Status
☑	1	WellSelectedObjects	Binary	2	100, NA	0	3	0.22%	♦
☑	2	CellNum	Binary	1	NA	0	1386	100%	⊘
□	8	WellSelectedObjectCountPerValidField	Continuous	762	150.571428571429, 82.375, 209.4, 174.833333333333, 196.333333333333	47.32	3	0.22%	●
□	9	WellValidFieldCount	Continuous	7	7, 8, 5, 6, 4	1.03	3	0.22%	●

2. CellObjectAreaCh1

Plate name	Replicate	Range	Mean	Median	Mode	Missings	Standard Deviation	Q1	Q3	Distribution	Skewness	Kurtosis	Plots
150013	1	42.769,584.511 difference: 541.742	220.857	209.568	189.61	0% (0 misses from 409877)	69.92	178.204	253.763	normal	1.219	3.125	

CellObjectAreaCh1 (2), replicate 1, Plate 1 128.31, 152.54, 176.78, 176.78, 192.46, 208.14, 240.93, 245.21, 249.49, 315.07

☑ CellObjectAreaCh1

3. CellXCentroid

Plate name	Replicate	Range	Mean	Median	Mode	Missings	Standard Deviation	Q1	Q3	Distribution	Skewness	Kurtosis	Plots
150013	1	4,506 difference: 502	256.857	257	234	0% (0 misses from 409877)	140.82	137	377	Uniform ⚠	-0.007	-1.166	

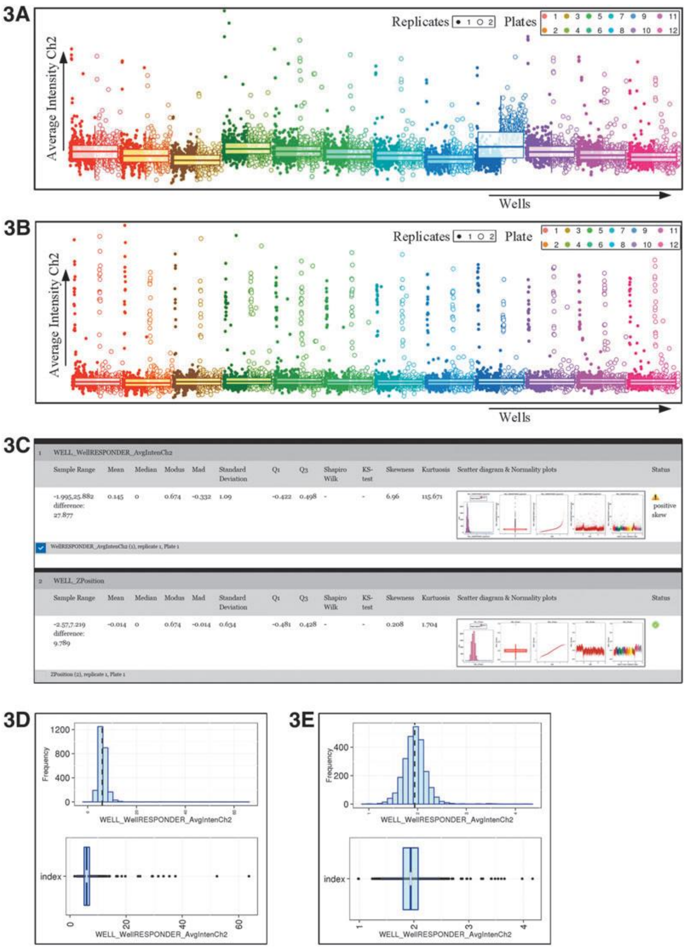
CellXCentroid (3), replicate 1, Plate 1 26, 30, 185, 286, 289, 326, 382, 382, 383, 495

☑ CellXCentroid

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A	empty	empty	A03	A04	A05	A06	A07	A08	A09	A10	A11	A12	A13	A14	A15	A16	A17	A18	A19	A20	A21	A22	empty	empty
B	empty	empty	B03	B04	B05	B06	B07	B08	B09	B10	B11	B12	B13	B14	B15	B16	B17	B18	B19	B20	B21	B22	empty	empty
C	C01 +	C02 +	C03	C04	C05	C06	C07	C08	C09	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	C21	C22	C23 +	C24 +
D	D01 +	D02 +	D03	D04	D05	D06	D07	D08	D09	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23 +	D24 +
E	E01 -	E02 -	E03	E04	E05	E06	E07	E08	E09	E10	E11	E12	E13	E14	E15	E16	E17	E18	E19	E20	E21	E22	E23 -	E24 -
F	F01 -	F02 -	F03	F04	F05	F06	F07	F08	F09	F10	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20	F21	F22	F23 -	F24 -
G	empty	empty	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	G16	G17	G18	G19	G20	G21	G22	empty	empty
H	empty	empty	H03	H04	H05	H06	H07	H08	H09	H10	H11	H12	H13	H14	H15	H16	H17	H18	H19	H20	H21	H22	empty	empty
I	I01 -	I02 -	I03	I04	I05	I06	I07	I08	I09	I10	I11	I12	I13	I14	I15	I16	I17	I18	I19	I20	I21	I22	I23 -	I24 -
J	J01 -	J02 -	J03	J04	J05	J06	J07	J08	J09	J10	J11	J12	J13	J14	J15	J16	J17	J18	J19	J20	J21	J22	J23 -	J24 -
K	K01 +	K02 +	K03	K04	K05	K06	K07	K08	K09	K10	K11	K12	K13	K14	K15	K16	K17	K18	K19	K20	K21	K22	K23 +	K24 +
L	L01 +	L02 +	L03	L04	L05	L06	L07	L08	L09	L10	L11	L12	L13	L14	L15	L16	L17	L18	L19	L20	L21	L22	L23 +	L24 +
M	M01 -	M02 -	M03	M04	M05	M06	M07	M08	M09	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20	M21	M22	M23 -	M24 -
N	N01 -	N02 -	N03	N04	N05	N06	N07	N08	N09	N10	N11	N12	N13	N14	N15	N16	N17	N18	N19	N20	N21	N22	N23 -	N24 -
O	empty	empty	O03	O04	O05	O06	O07	O08	O09	O10	O11	O12	O13	O14	O15	O16	O17	O18	O19	O20	O21	O22	empty	empty
P	empty	empty	P03	P04	P05	P06	P07	P08	P09	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	empty	empty

Figure 2. *Parameter selection and quality control*

(A) A screenshot of data preprocessing where all parameters are inspected for >95% missing data, binary, discriminant, and/or a standard deviation of 0. All parameters that meet one of these aspects are checked for removal, which results in a list of parameters to continue with. (B) A screenshot of Parameter Selection, where the scientist can analyze the distribution in more detail, based on visualizations and additional statistical metrics. HC StratoMineR will suggest the exclusion of parameters with a uniform distribution. (C) A screenshot of the plate map configuration at the QC step. Colors represent the various controls. Here, red represents NEGATIVE, green is POSITIVE, black is EMPTY, and blue is SAMPLE. Once a plate map has been defined, QC plots can be created. (D) An example of a scatter plot from QC on the whole screen. On the x-axis, all the reagents of the screen are plotted and ordered according to the controls defined in the plate map. The y-axis represents the median of the replicates for each reagent of the selected variable. The plot is created based on the configured plate map 2C.



1 WELL_WellRESPONDER_AvgIntenCh2

Sample Range	Mean	Median	Modus	Mad	Standard Deviation	Q1	Q3	Shapiro Wilk	KS-test	Skewness	Kurtuosis	Scatter diagram & Normality plots	Status
-1.995,25.882 difference: 27.877	0.145	0	0.674	-0.332	1.09	-0.422	0.498	-	-	6.96	115.671		⚠ positive skew

☑ WellRESPONDER_AvgIntenCh2 (1), replicate 1, Plate 1

2 WELL_ZPosition

Sample Range	Mean	Median	Modus	Mad	Standard Deviation	Q1	Q3	Shapiro Wilk	KS-test	Skewness	Kurtuosis	Scatter diagram & Normality plots	Status
-2.57,7.219 difference: 9.789	-0.014	0	0.674	-0.014	0.634	-0.481	0.428	-	-	0.208	1.704		●

ZPosition (2), replicate 1, Plate 1

Figure 3. *Data normalization and data transformation*

(A) Raw data plotted per plate with a boxplot. The colors represent the plates, the dots represent the wells, and the shape of the dot represents the related replicate number. The x-axis represents the wells, and the y-axis represents the raw value of the selected parameter. (B) Represents a similar visualization to (A) but here, the data are normalized against the NEGATIVE control, which is done on a plate-to-plate basis. In HC StratoMineR, the two conditions can be compared by hovering over one visualization. (C) A screenshot of data transformation. An overview of statistical metrics and plots required for the decision of whether to transform a parameter. In data transformation, we try to get an approximate multivariate normality. Therefore, an overview is provided on every parameter in the data. Two parameters are shown. The user is recommended as to whether a parameter requires transformation. The user can select the parameters that he/she wants to transform. (D) Untransformed parameter plot. It shows a histogram and (rotated) boxplot; the x-axis shows the (transformed) selected variable for both the histogram and the boxplot. The histogram shows a binned frequency of occurring values of the selected variable on the y-axis. The skewness of the parameter will be shown in an iteration for each variable that was selected where a transformation method is recommended. (E) Transformed parameter plot. The figure is similar to (B) but here, the data are log transformed (recommended by HC StratoMineR). The user can preview a transformation and apply that transformation when satisfied with the selected transformation.

Plate Configuration and Quality Control

In this step the user can carry out quality control based on the performance of the screened

Plate Configuration and Quality Control In this step, the user can carry out quality control (QC) based on the performance of the screened reagents and controls. The user defines the plate map for the experiment. Each well is defined as a positive, negative, empty, or sample well or can be given a specific name (figure 2C). The user can choose various plots for each parameter across the whole data set (figure 2D) and on a plate-to-plate basis to have more insight into data quality. Controls and samples are plotted separately. Also, plate wise strictly standardized mean difference (SSMD), and Z′ are calculated (Birmingham et al., 2009).

Plate Normalization

Next, the data set can be normalized against its mean/median of the samples or against its negative or positive controls to account for plate-to-plate variation in assay performance (Figure 3A, B). Plate effects can be addressed by using the B-score normalization method. A consistent normalization will be applied to all the parameters in the data set in each plate. The user can preview the before and after effect before deciding to apply a normalization method (figure 3A, B).

Data Transformation

For the best results in subsequent analysis steps such as multiple imputation, common factor analysis (CFA), k-means, and cluster analysis, data should approximate a normal distribution (Costello & Osborne, 2005). For this reason, the parameters are checked for the requirement for transformation (Figure 3C). This is done by measuring a significant skewness ($p < 0.0001$). Also, Shapiro–Wilk and Kolmogorov–Smirnov test results are provided (Royston, 1982A; Royston, 1982B; Marsaglia, Tsang & Wang 2003). Information regarding each parameter is given, and HC StratoMineR provides suggestions as to whether and what transformation a parameter requires (Figure. 3D, E).

Data Standardization

Data standardization, or feature scaling, prevents a bias toward a parameter that has a larger range. Many scaling methods have been proposed, including min-max, z-score, and robust z-score (Marsaglia, Tsang & Wang, 2002). Depending on the nature of the screen, there could be a bias in variance and mean per plate or per replicate. That is why the user can choose to preview and apply a standardization method at plate level, replicate level, or if there is no bias, screen level (Figure. 4C, D).

Multiple Imputation

Missing data constitute a major complication in data mining, as many methods such as regression, factor analysis, k-means, and hierarchical cluster analysis cannot handle data sets containing missing data. In HCS, image analysis software may fail to generate measurements, leaving a missing (NULL/not available [NA]) value in the data set. Also, in some rare cases, normalization or transformation methods can generate NA, not a number, -infinitive number (INF) or INF values (Figure 4A). HC StratoMineR highlights parameters that contain a significant number of missing data points compared with the default parameter. Parameters containing missing data can be excluded from further steps. This does not involve omitting any reagents in the data set. Row-wise deletion is not implemented, because this could involve excluding most reagents from the data set. Also, the user can leave out parameters based on the percentage of missing data or the fact that parameters do not show a difference between controls (Figure. 4B).

There is a built-in order of functions that are run by HC StratoMineR to account for missing data. First, the median of the data from the other replicates for that same plate and location is used for imputation. If there are no data available from other replicates for the same parameter within the same reagent, there is no value to be imputed. In this case, Amelia II, (Honaker, King & Blackwell 2011) a package that can provide a solution for up to 50 parameters, is used. An expectation-maximization (EM) method in combination with bootstrapping and Bayesian hierarchical classification is applied. The method is an iteration and can be run in parallel. The method starts if the data matrix contains missing data. The data are bootstrapped; then, the EMmodel is built and used to impute the missing data. Finally, an analysis is done on the created imputed data sets in which they are merged into one solution, which is the final imputed data matrix (Honaker, King & Blackwell 2011). If Amelia II fails (or the number of parameters exceeds the capacity of Amelia II), Mice, a multiple imputation technique based on regression, is applied to the data (Buuren & Groothuis-Oudshoorn, 2010). A column-wise median is imputed on any remaining parameters that contain missing data, which results in a matrix without any missing data.

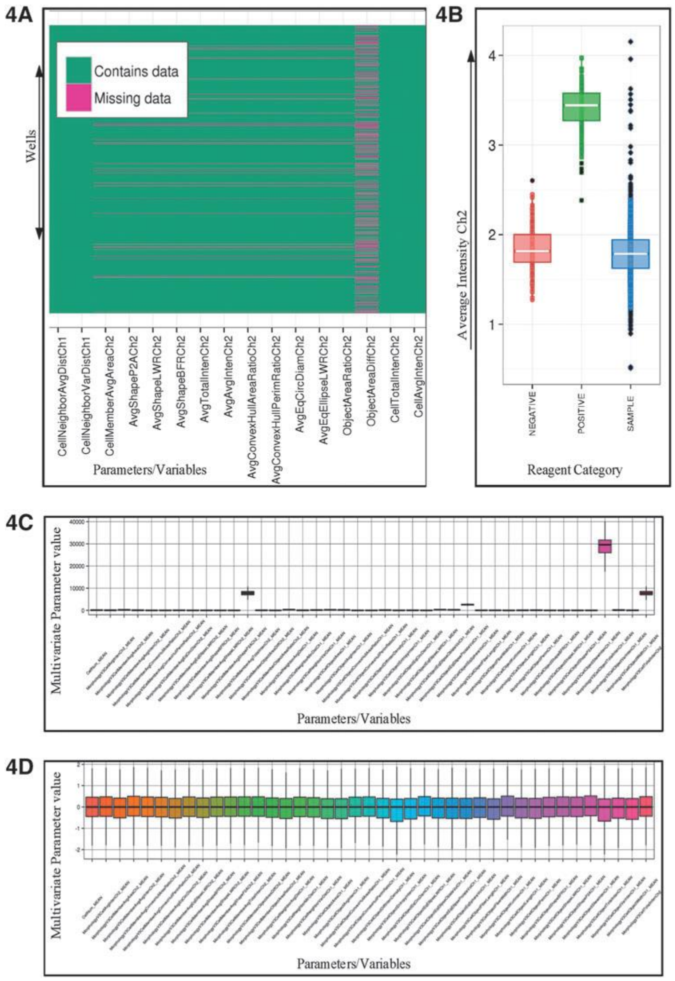


Figure 4. *Missing data and data standardization*

*(A) Missing data heat map. On the x-axis, the parameters are provided and shown in each column; on the y-axis, the wells are presented. The red color represents missing data, and green represents that data are present. One can see the trend and amount of missing data for each parameter. (B) Response to controls. The x-axis shows the controls, and the y-axis shows the normalized value of a parameter. For each variable, a collection of figures is shown as in (B). The user can decide based on this figure whether this parameter gives enough difference between the controls and together with the amount of missing data to include the parameter in further steps or to discard it. (C) Non-standardized data plot. The x-axis represents the individual parameters, and the y-axis represents the multivariate values (value range of all plotted parameters). All selected parameters are visualized to get the parameters across the data set in a similar range. Multiple standardization methods are available. HC StratoMineR provides a non-standardized data visualization and the user can choose a standardization method that will result in a similar range, mean, and standard deviation per individual parameter. (D) Standardized data plot. Represents a similar visualization to (A), but now the parameters are standardized by using a robust z-score where all the parameters have a mean of *0, a standard deviation of 1 and all are in a similar range.*

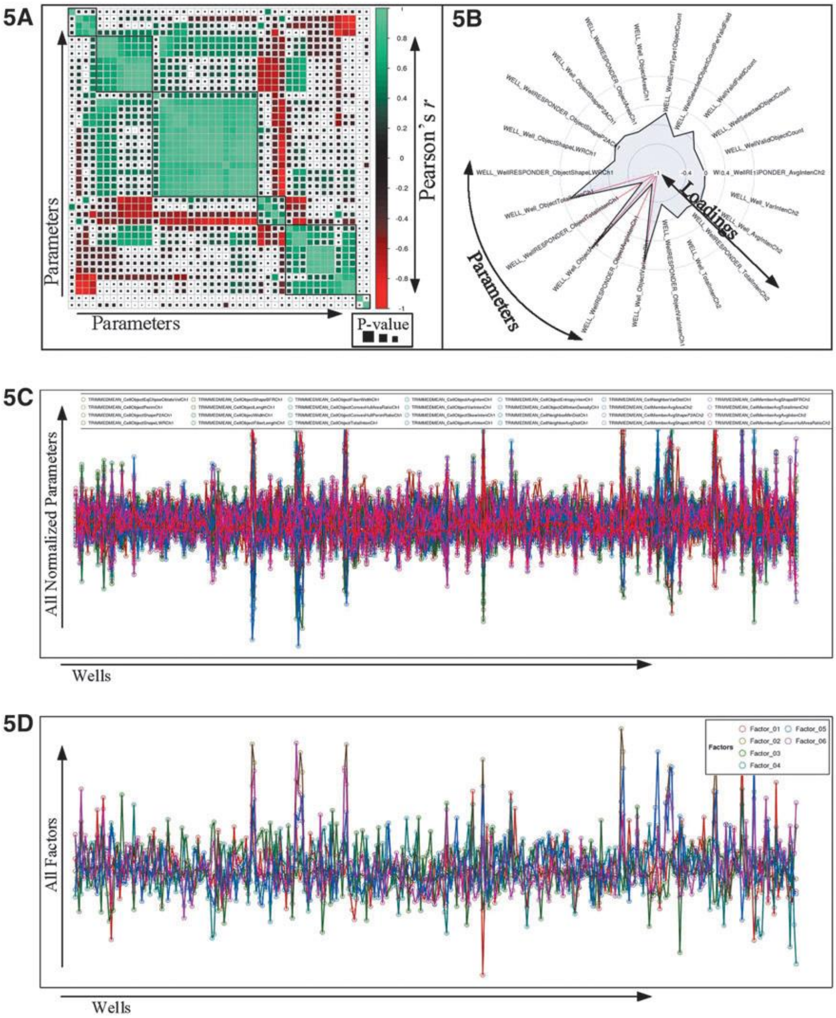


Figure 5. *Data reduction*

(A) Hierarchically clustered correlation matrix. The x-axis and y-axis represent the included parameters. The color of each small square represents a Pearson's correlation coefficient between -1 and 1. The size of each small square represents the P value of the Pearson's r. The number of factors is indicated in the correlation matrix as larger squares around one or more parameters that have a high internal covariation. (B) Polar plot. The polar angles represent the parameter names, and the radius represents the factor loading of the parameter. A significant contribution is considered if >0.4 or <-0.4 and is indicated with a red line. Every factor is visualized in this manner. (C) Multivariate parameter plot. The x-axis shows the wells of one plate. The y-axis shows the multivariate values of the included parameters for factor analysis. The different colors represent the parameters. (D) Multivariate factor plot. This plot represents a similar visualization to (C), but the calculated factor scores are plotted on the y-axis instead of the parameter scores. A comparison of (C) with (D) gives an insight into the effect of the chosen data reduction method.

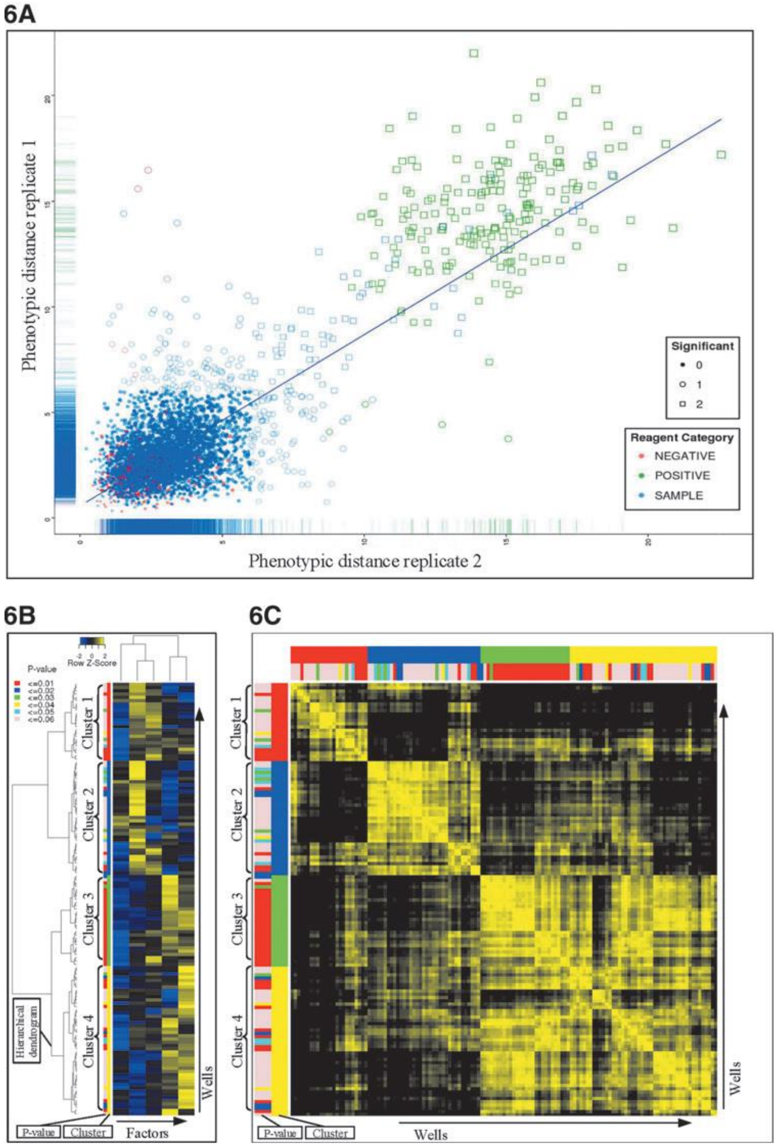


Figure. 6. *Hit selection and cluster analysis*

(A) Phenotypic hit selection plot. The y-axis represents replicate 1 of the screen, and the x-axis represents replicate 2 of the screen. The plot shows the distance score from the NEGATIVE control. The controls are shown in colors. The calculation of the distance is based on included factors or parameters. (B) Cluster analysis. The columns represent the included factors, and the rows represent the reagents. A dendrogram that represents the hierarchical relationships is provided. Color bars indicate which k-means cluster a well belongs to, and a second color bar indicates the P value based on the distance score, calculated in the previous step hit selection. (C) Similarity matrix. The columns and rows represent reagents. The intensity of the color represents the similarity based on a cosine vector score calculated from the included factors or parameters, as in (A). Color bars again represent the clusters and P values.

Data Reduction

In HC StratoMineR, data reduction can be achieved by performing CFA or principal component analysis (PCA) based on a correlation matrix (Figure. 5A). The user can choose an orthogonal rotation where the resulting factors are independent, or an oblique rotation where the resulting factors can have an overlap. A major difference between PCA and CFA is that PCA is computationally less intensive, its goal is just to reduce the dimensionality in the data, and it does not require such strict assumptions (Costello & Osborne, 2005). Each factor or component is visualized in a separate polar plot, indicating the loadings for each parameter in that factor (Figure. 5B). At the end of the data reduction phase, the loadings are used to calculate the factor scores for every sample and control well. The effect of data reduction is shown in comparison to the original parameters (figure 5C, D).

Hit Picking

Hit selection uses a subset of the individual parameters or calculated factor scores generated in the data reduction step to identify significant outlier samples, which are different from one of the selected controls. The Manhattan distance (another preferred distance metric) of all the vectors to the controls is calculated on a plate-to-plate basis for each replicate (figure 6A). This reduces the data for each record to just one distance score, regardless of the number of dimensions. This distance is a measure of the phenotypic effect of the sample on the cells in that well (Young et al., 2008). The Manhattan distances are then transformed to P values by using a Poisson probability distribution with the Lambda of the sample distribution. Statistically significant hits can then be identified for each replicate based on the desired P value (e.g., ≤0.05). Hit selection is carried out for each plate separately due to plate-to-plate variation. Optionally, the mean/median of the Manhattan distances for each well based on its replicates is calculated. The result of the chosen strategy is visualized in a scatter plot, which indicates the selection of hits (figure 6A).

Clustering

Hierarchical cluster analysis can then be performed on the statistically significant hits based on the selected parameters or factors created in the data reduction phase (Young et al., 2008) (Figure. 6B). The clustering method can be chosen and is set on Ward's linkage criteria by default, as this creates compact clusters that share their similarities with their neighbors (Mooi & Sarstedt, 2011).

The final visualization of the data is a similarity matrix (Figure. 6C). For each combination of phenotypic vectors, a cosine vector score is calculated. This is a measure of the angle between the vectors, which can be considered as the similarity of the effect of samples. A generated matrix contains the scores for the possible combinations of samples. These are again clustered according to the same order as the hierarchical cluster analysis. These clustering results are visualized in heat maps with dendrograms for the records (reagents) and a K-means visualization stitched to the heatmap to identify clusters or groups of similar reagents and controls (figure 6B, C). The number of clusters can be automatically detected by partitioning around medoids using the fpc package or can be set manually (Reynolds & Richards, 2006). A less complex visualization, a multidimensional scaling (MDS) plot is also provided, which shows the first and second dimensions of MDS in a scatterplot.

Results

HC StratoMineR uses numeric parameters that are extracted from the images of individual cells to first identify outliers and then cluster the outliers according to the similarity of the resulting profiles. It does this in an unbiased fashion, and the result should be that cells from wells that cluster apart in the data mining step display different morphologies. It should also allow the user to functionally categorize hits. We validated these in two high-content screens: a chemical screen and a genome-wide siRNA knock-down screen, respectively.

The chemical screen was used to characterize 51 compounds that had previously been identified in a screen for small molecules that can kill cells that are highly resistant to apoptotic stimuli (Pagliero et al., 2016). Our assay included three fluorescent labels: Hoechst dye for staining DNA; Lysotracker Red that accumulates and is fluorescent in acidic organelles (in mammalian cells, these are predominantly lysosomes); and Calcein-AM, a marker for cell viability and overall cell morphology. The multiparameter data from the screen were processed with HC StratoMineR generating four factors. All 51 compounds were clustered (figure 7A), and we identified four strong clusters. Clusters 1 and 4 included Fenretinide and Siramesine, respectively, compounds that had been included as controls, since they had been previously described to accumulate in lysosomes and induce cell death. Pimozide and Clomiphene were found in Cluster 4 with Siramesine (figure 7A) and, indeed, showed greatly decreased Lysotracker staining (figure 7D), which was consistent with the phenotype of lysosomal accumulation. Astemizole was in Cluster 1 with Fenretinide (figure

7D), showing a similar lysosomal phenotype. The similarity in the profiles of Siramesine and Fenretinide suggests that these compounds may kill cancer cells via a mechanism of action that involves lysosomal dysfunction, which is consistent with previous studies on Siramesine (Ostenfeld et al., 2005). Auranofin clustered with Staurosporine in Cluster 3 and did not show loss of Lysotracker staining (figure 7D). Therefore, it is probable that compounds in Cluster 3 can kill cancer cells by a different mechanism of action that does not involve the lysosome. Cluster 2 consisted of just one compound Mitoxantrone, which gave a very strong distinct phenotype (figure 7D), with greatly decreased Calcein-AM staining.

Investigation of the phenotypic similarity matrix (figure 7B) highlighted the individual clusters and also showed similarity between Clusters 1 and 3 and between Clusters 1 and 4. Cluster 4, however, is clearly morphologically very distinct from Cluster 3. One possibility is that compounds in Cluster 4 (such as Siramesine) may trigger a mechanism of action that depends solely on lysosomal accumulation; whereas those in Cluster 1 (like Astemizole) may also have contributions from other biological phenomena to its final cytotoxic mechanism of action. More characterization and possibly the combination of different high-content assays would be necessary to confirm and define the biological phenomena involved.

A compound similarity matrix was also generated based on the clustering data from HC StratoMineR. This highlights many similar structures in Clusters 1 and 4 (figure 7C, 7E). These compounds are cationic amphiphilic drugs (CADs), and intriguingly, this class of compounds has previously been shown to accumulate in the lysosomes where they get trapped after protonation in the acidic lysosomal lumen. Because of these physicochemical characteristics, cationic amphiphilic compounds would be expected to interfere with the lysosomal pH gradient, altering the lysosomal enzymes’ activity (Nadanaciva et al., 2011).

The results from this experiment clearly validate the ability of HC StratoMineR to successfully categorize the phenotypic responses of cells to small molecules. The results also highlight the power of this form of high-content analysis in that one can see structure–activity relationships in the mined data.

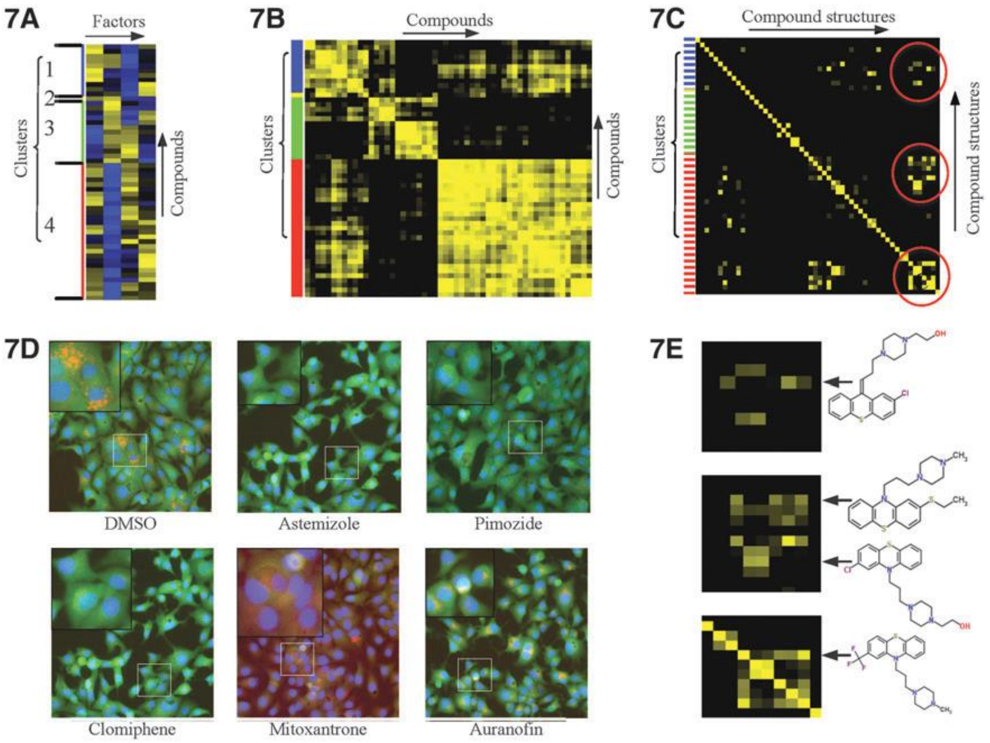


Figure 7. *High-content chemical screen*
(A) Hierarchical clustering analysis with compounds on the y-axis grouped by k-means, Clusters 1–4. Factors are represented on the x-axis, where each column represents one factor. (B) Compound phenotype similarity matrix with compounds on the x-axis and y-axis. Colored squares represent the phenotypic similarity of the compounds based on a cosine vector score of the factors represented in (A), clustered by k-means. Yellow indicates higher phenotypic similarity. (C) Compound structure similarity matrix, with compounds on the x-axis and y-axis. Colored squares represent the Tanimoto similarity score of the structures, clustered by the k-means clusters presented in (A, B, D). Images of cells treated with the named compounds. Images are an overlay of Hoechst (blue), Calcein-AM (green), and Lysotracker Red (red) staining. (E) Details of compound similarities circled in (C).

To demonstrate the utility of HC StratoMineR for the analysis of larger data sets, we used it to process data from a high-content, genome-wide siRNA knock-down screen. The goal of the screen was to identify genes involved in the regulation of mitosis (Neumann et al., 2010). After siRNA knock-down, cells were labeled with 40,6-diamidino-2-phenylindole (DAPI) to identify the nucleus and an antibody against phosphorylated histone H3 to identify cells in mitosis. The screen was carried out in duplicate in one hundred twenty-four 384-well assay plates. Numeric data were generated at the cellular level, resulting in a data set of 45,586,699 data records (~3.3 billion data points and ~35 GB) across 74 parameters, of which 57 were analytical. Of these, 41 were found to be useful for factor analysis.

See “Analysis Steps” in Supplementary Data for more details on the actions taken in each step. We initially used this data set to compare various analysis strategies within HC StratoMineR. Biologists who do not have access to bioinformatics services or advanced data handling tools are often limited to simple analyses, for example, only using one parameter from a high-content data set. To demonstrate how HC StratoMineR could be used to get more value from a data set, we analyzed our genome-wide data set with increasing levels of analytical rigor. The output was evaluated by looking at the number of statistically significant hits (Table 1).

Table 1. *Comparison of Data Analysis Methods*

Analysis Strategy	Estimator	Dimensions	Significant Hits	Cell Level Data
1	MEAN	One Parameter	213	FALSE
2	MEAN	One Factor	247	FALSE
3	MAD	One Factor	338	TRUE
4	MAD	All Factors	586	TRUE
5	IQR	All Factors	614	TRUE
6	MEAN	All Factors	615	FALSE
7	Trimmed MEAN	All Factors	634	TRUE

Our most simple analysis strategy (see Strategy 1 in Table 1) was similar to what is frequently used by screeners who do not have access to advanced bioinformatics tools. This is a single parameter well-level analysis based on the percentage of cells positive for anti-phospho histone H3 staining. This was determined based on a cut-off defined by the screener and gave 213 statistically significant hits. The use of one common factor (see Strategy 2 in Table 1) that contained multiple parameters linked to anti-phosphohistone H3 staining gave 247 statistically significant hits. Switching to cell-level data, however, gave us the opportunity to use robust identifiers such as the MAD, IQR, or trimmed mean. The use of MAD gave an increase of 338 hits. This along with the use of multiple factors allowed us to increase the number of statistically significant hits almost three-fold (Strategy 4–7) to a maximum of 634. The increased number of statistically significant hits detected does lead to a decrease in the percentage of Mitotic hits, 19.24% (41) in Strategy 1 versus 10.72% (68) (see Strategy 7, Table 1). This is due to the fact that by including all of our factors, we are asking a broader question of our data and so we are picking up more phenotypic hits that are not necessarily directly related to mitotic arrest. The increase of statistically significant hits is not, by itself, necessarily desirable as it, no doubt, introduces larger numbers of false positives or hits that are not relevant to the goal of the screen. Indeed, it is known that functional genomic screens using readouts that are related to the cell cycle or viability frequently produce hits from the proteasome, various protein synthesis-related complexes such as the ribosome, and RNA splicing machinery (van Heesbeen, 2015). To determine whether we could separate the various biological processes represented in the statistically significant hits, we performed cluster analysis that generated seven clusters by using four factors with high factor loadings based on anti-phospho histone H3 parameters (figure 8A). We took the lists of hits from each cluster and based on network analysis in STRING DB, (Szklarczyk et al., 2011) and a literature search, we assigned them to the following categories: splicing, mitosis, ribosome, and proteasome. It was immediately clear that there was functional separation in the clusters (figure 8B).

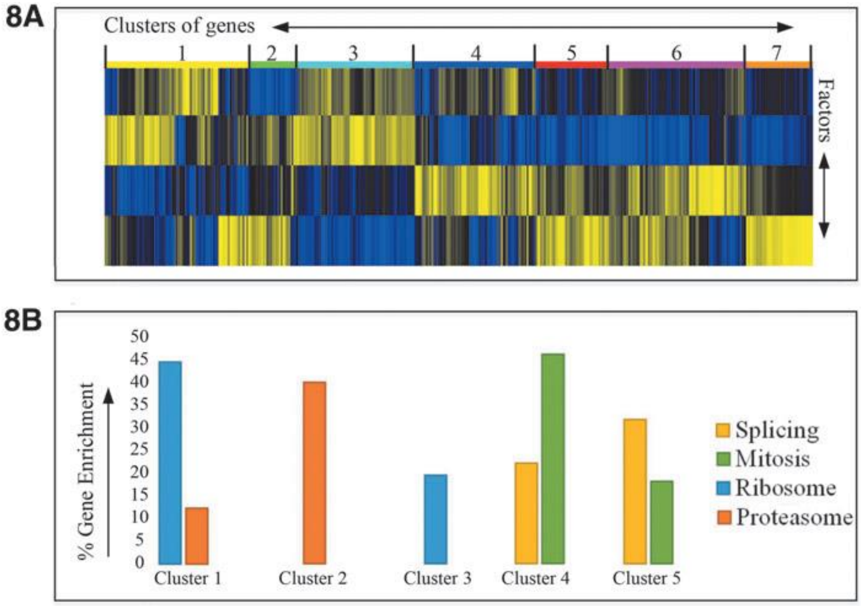


Figure 8. *High-content analysis of a genome-wide siRNA knock-down screen*
(A) Cluster analysis of hits from a genome-wide siRNA knock-down screen. The (rotated) figure shows the selected factors represented in the rows and the genes in the columns grouped by k-means as Clusters 1–7. (B) Enrichment in percentage of the clusters from (A). Using String-DB shows that Clusters 1 and 3 are highly enriched for Ribosome genes, Clusters 1 and 2 are enriched for Proteasome, and Clusters 4 and 5 are enriched for splicing and mitosis. Splicing, yellow bars; Mitosis, green; Ribosome, blue; and Proteasome, red.

Clusters 4 and 5 were the most highly enriched for mitotic genes, whereas ribosome and proteasome genes were enriched in Clusters 1, 2, and 3 (Figure 8B). The splicing genes, however, did not, in this analysis, separate from the mitotic genes. This would suggest that the phenotypes are very similar in this assay. Clusters 6 and 7 did not contain significantly enriched genes from the four assigned categories. These results suggest that HC StratoMineR can not only help researchers identify more hits but also help them identify sections of the data set that are highly enriched for the hits that are relevant to their biology of interest. Our analysis would suggest that novel genes involved in mitosis are more likely to be found in Clusters 4 or 5 than in Clusters 1, 2, or 3. The use of the more advanced method in HC StratoMineR, thus, increased the percentage of relevant mitotic genes by 66% compared with the simplest analysis (see Strategy 1, Table 1).

One important goal in the development of HC StratoMineR was to have a tool that could rapidly analyze large data sets. In our workflow, the computations involved in the Parameter Selection step are rate limiting when it comes to completing a particular analysis. To benchmark the performance of HC StratoMineR, we carried out analyses of varying complexities and with different hardware. With the use of HPC and multithreading, we could complete the rate-limiting step for our genome-wide screen in 7 min (Table 2).

Table 2. *Analysis Time*

# Plates	# Parameters	# Cores	Data Resolution	Format	Hardware	Time (min.)
6	41	6	Cell	96	Single Server	<1
8	41	32	Cell	384	Single Server	~4
124	41	32	Cell	384	Single Server	~14
124	41	156	Cell	384	HPC	~7
8	180	4	Well	384	Single Server	<1

Discussion

HCS technology was first developed in the 1990s; however, although the use of HCS instrumentation has become widespread, many users are not making full use of the power of multiparameter data analysis at cellular resolution. Our experience at the Cell Screening Core leads us to believe that this is due to a lack of access to the appropriate bioinformatics and biostatistics skills, tools, and hardware required to mine complex data sets. Indeed, this is a problem throughout the wider area of life sciences, as it becomes increasingly easy for biologists to generate larger and more complex data sets by using various omics technologies. Our goal in the development of HC StratoMineR was to develop a tool that would allow biologists to independently mine high-content data sets. The need for such tools is also being driven by the resurgence of interest in phenotypic screens for drug discovery. There is an emerging strategy in the pharmaceutical industry based on the idea that phenotypic screens are more promising for first in-class molecules. The molecules from such screens can be used to help identify the (often multiple) useful drug targets for a disease. These can then be addressed by using more targeted approaches (Swinney 2011).

Our validation of the tool has already highlighted many useful features of the software. The QC functionality makes it very easy to get a good overview of the quality of the data at an early stage while the screen is in progress. The CFA functionality has proved to be invaluable. Even though the factors are generated in an unbiased fashion, users can see the underlying biology being represented in the factors that are generated. If a DNA label such as Hoechst or DAPI is included in the assay, we almost invariably see a factor appearing that is related to cellular toxicity. Users can then choose the factors that are related to their question of interest or if they are interested in a broader question, they can include all factors.

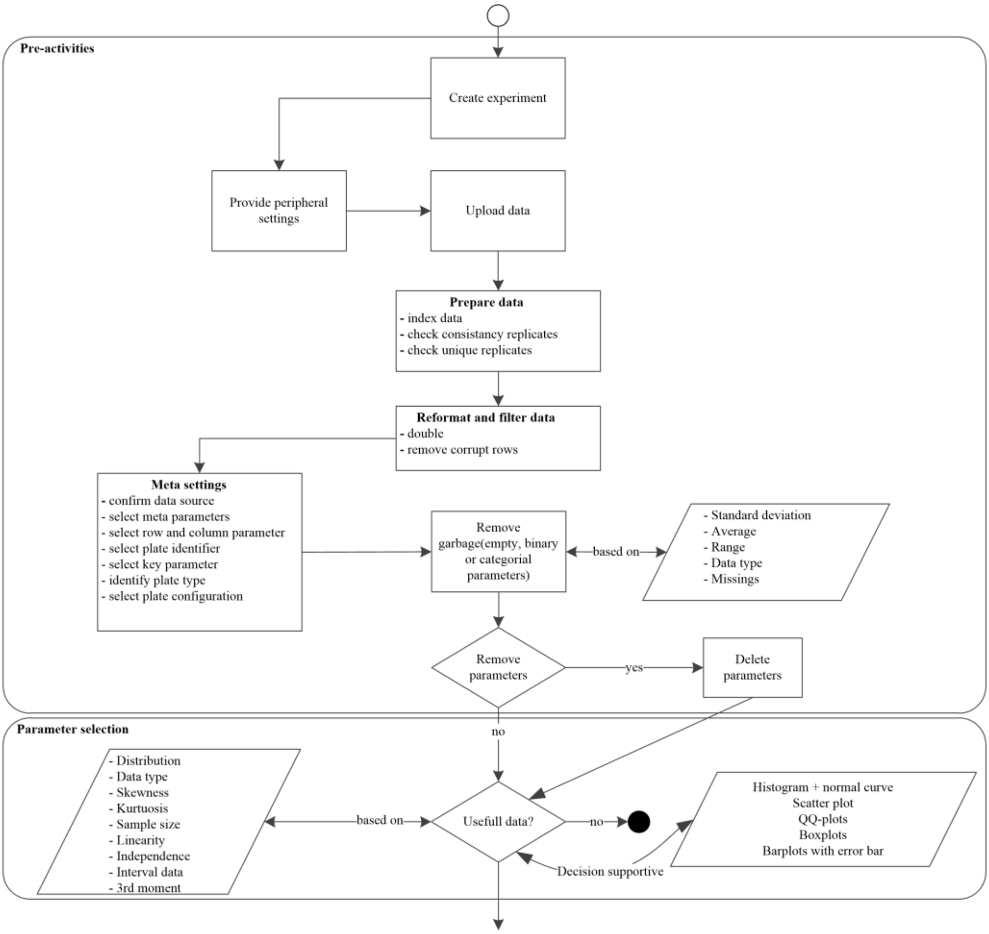
One of the most valuable features of HC StratoMineR has proved to be the speed with which large data sets can be analyzed. If a researcher must work with a bioinformatician to develop scripts for the analysis of data, there are invariably delays, as the biologist needs to educate the bioinformatician about the biological problem. This slows the iterations of data analysis, and the result is that fewer strategies for data analysis are tested. In the case of the genome-wide data set, we presented HC StratoMineR and reduced the analysis time from months to hours. Also, the data analysis method is frequently not addressed until after the data acquisition has finished. HC StratoMineR can be used at all stages of the project, and this gives the user the ability to develop the analysis method at the piloting stage. This can inform the way the screen is run.

Our high-content chemical screen demonstrated the value of HC StratoMineR for phenotypic drug discovery. We were able to correctly identify the CADs as disrupters of lysosomal function and the phenotypical separation of these from apoptosis inducing compounds such as Staurosporine and Auranofin.

The data from our siRNA knock-down screen demonstrated how HC StratoMineR can be used to improve the mining of phenotypic functional genomics data sets. The ability to identify more phenotypic hits and then use the clustering functionality to identify clusters that are enriched for hits of interest will potentially allow screeners to get past the “low hanging fruit” problem and identify weaker hits that are functionally relevant. Critically, HC StratoMineR allows analyses to be done more quickly, thus relieving an analysis bottleneck in HCS.

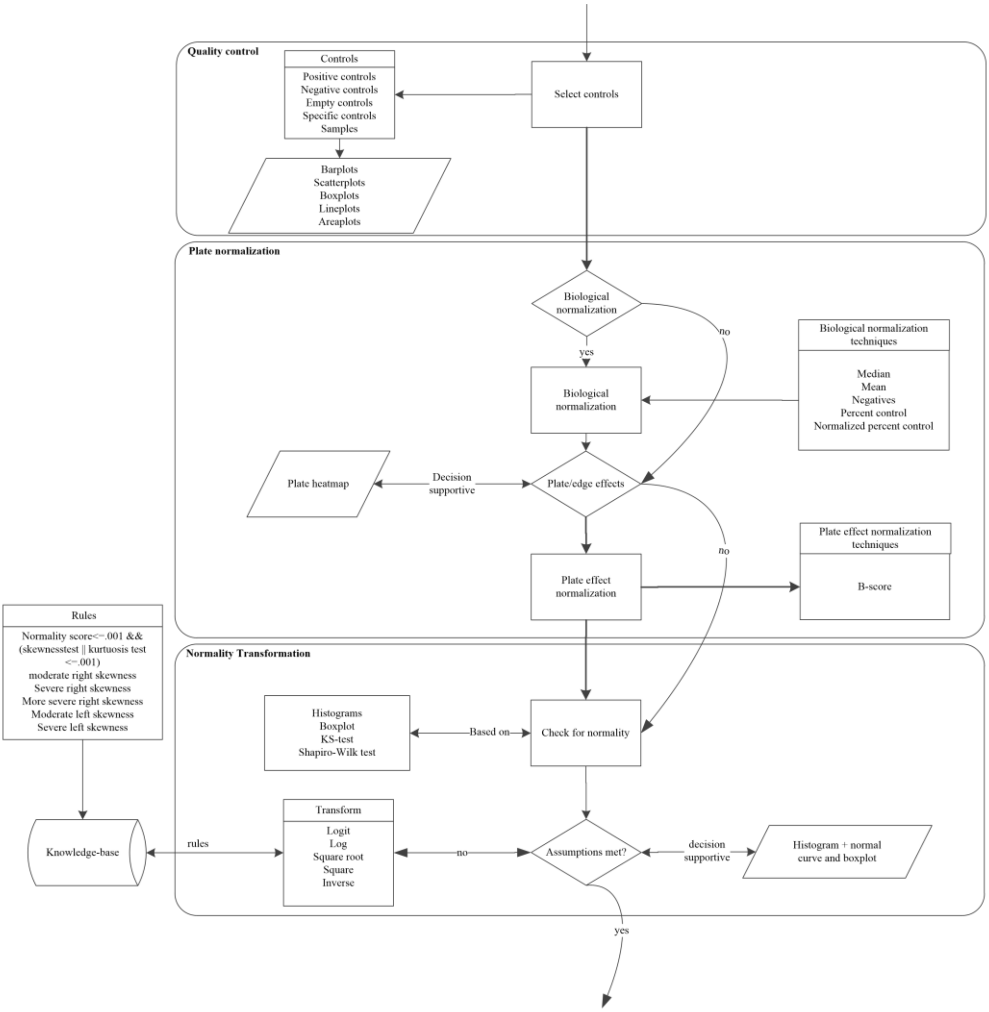
Supplementary Materials

detailed step-by-step visualization of the workflow, Figure 1

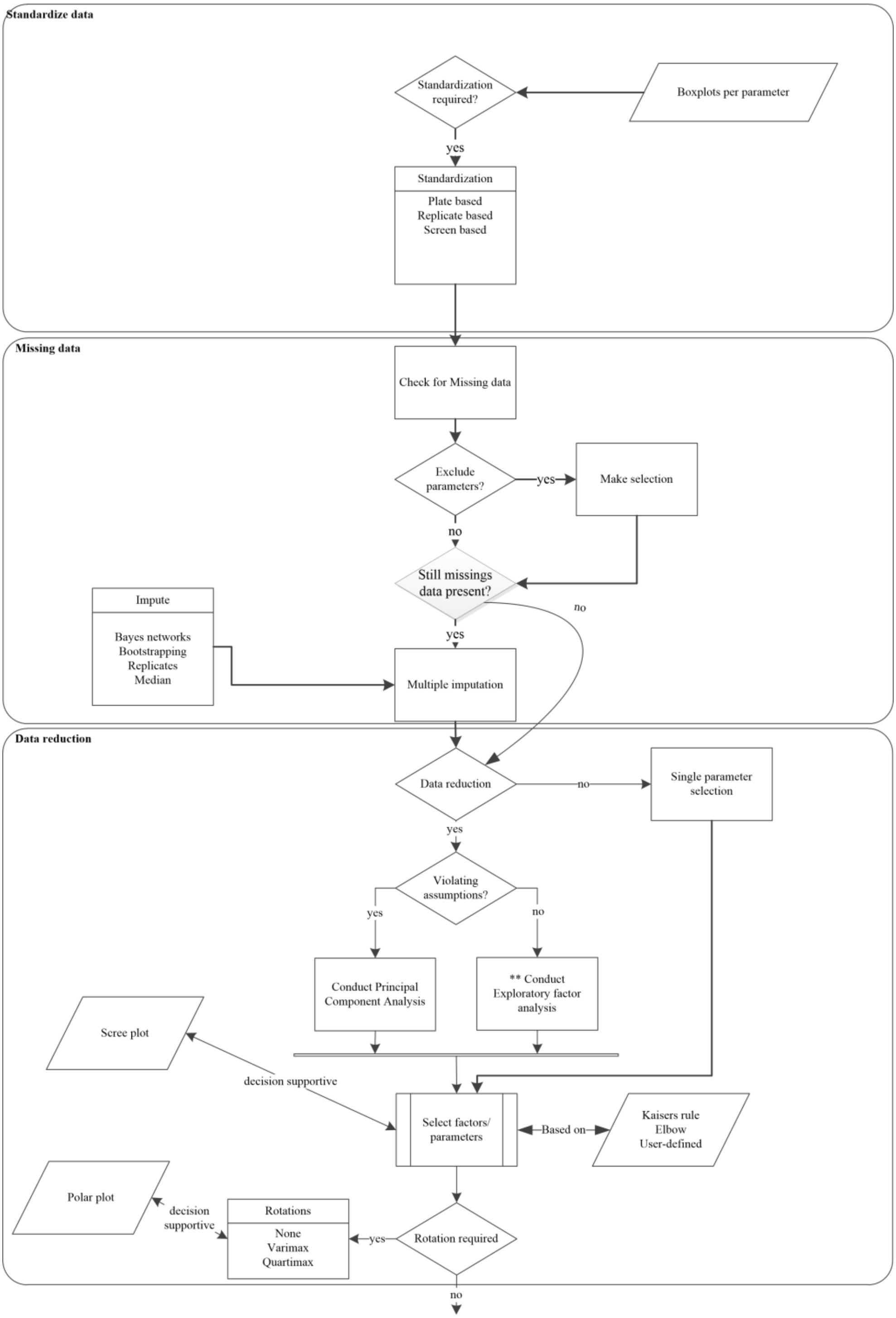


Supplementary Extended Workflow Figure

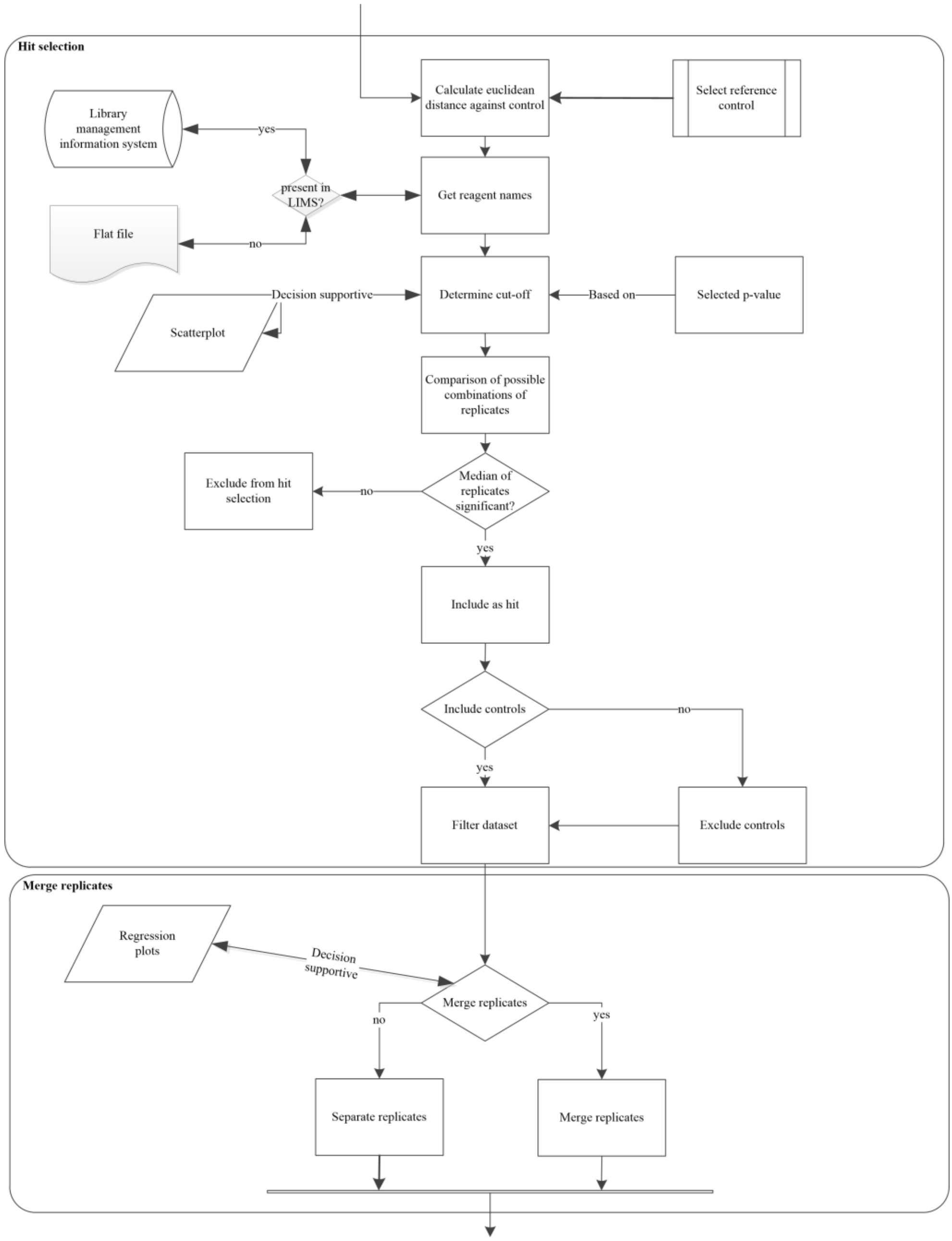
This figure Describes in detail the process from uploading the data to preprocessing, parameter (feature) selection, quality control (QC), plate normalization, data transformation, standardization (scaling), handling missing data, data reduction (dimensionality reduction), hit picking, clustering and the enrichment of data and publishing the analysis as a journal publication.



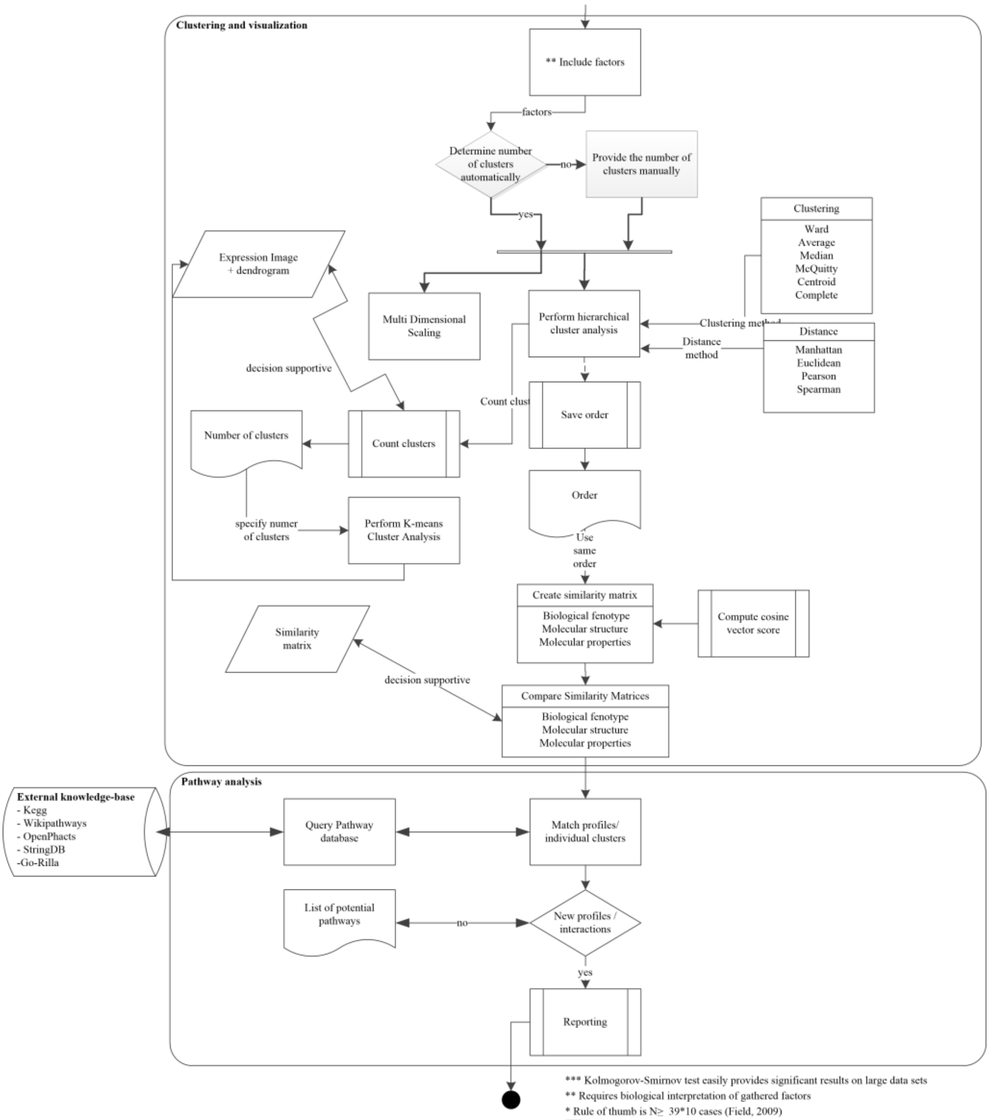
Supplementary Extended Workflow Figure Continued



Supplementary Extended Workflow Figure Continued



Supplementary Extended Workflow Figure Continued



*** Kolmogorov-Smirnov test easily provides significant results on large data sets

** Requires biological interpretation of gathered factors

* Rule of thumb is N≥ 39*10 cases (Field, 2009)

▱ Decision supportive

Supplementary Extended Workflow Figure Continued

Standard Operating Procedure (SOP)

1. (Open browser and link) Using Firefox, Chrome, Safari or Edge open https://cla.stratominer.com
2. (Log on) Log in with your credentials
3. Click the button “Create new experiment”
   a. Provide a name in the text box
   b. Click “Create experiment”
4. Choose the number of plates (4) and replicates (2) and click “Save and continue”
5. Click the “Upload” button
   a. Select the provided files (four zip files, approximately 800 MB)
      i. The files will be copied to the server, after copying and extraction, a green “Save and continue” button appears
      ii. Click the green button “Save and continue”
6. (File format) A preview of the uploaded files show if the default settings are correct. By default, the settings are correct for the test data set, check the files you want to include or click “Check all” and click “Save and continue”
7. (True replicates) Sort the files in the right order (true replicates) using the black arrows. In this case the files are in the correct order. Click “Save and continue”
8. (Select meta parameters) Select the meta parameters only (the non-analytical parameters). By default, the required parameters are already selected, click “Save and continue”
9. (Prepare parameters) HC StratoMineR will now prepare the parameters set the parameters to the right data type and convert empty strings to NULL values, it shows the progress of the processing, this step goes fully automatically.
10. (Meta settings) Provide critical information to HC StratoMineR
    a. Unique Plate id: select the parameter “PlateID”
    b. Plate name: select the parameter “BarCode”
    c. Row coordinate: select the parameter “Row”
    d. Column coordinate: select the parameter “Col”
    e. Plate type: select 384 well plate
    f. Level of detail: select Field/Cell
    g. Field: select the parameter “FieldID”
    h. Click “Save and continue”
11. (Check replicates) Now HC StratoMineR will check your replicates and optimize data access, this goes fully automatically

12. (Pre-selection of parameters) multiple parameters are listed and are advised or even required for removal. The parameters are automatically selected, please click “Delete parameter(s)”
13. (Default parameters)
    a. Select the parameter that describes the cells per well: “CellNum”
    b. Select the parameter that is the most important read-out: “CellAvgIntCh2”
    c. Now the coordinate configurator appears, do not change anything and click “Save and continue” [0,0]
14. (Basic statistics menu) Select your default estimator e.g. “median”, check the plots that you want, NOTE: most options are only available at cell level data. More options check means increasing waiting time!
15. (Select parameters) Select the useful parameters, at least one parameter is required to continue the pipeline) (The wrong parameters are automatically deselected). Select “Save and continue”
16. Please define controls in this page, configure them manually or select the template “ReviewersDataSet” and click “Apply template”. At least negative controls and samples are required
    a. (Explore data) Pick a plot type e.g. “scatter” or “barplot” after everything is defined and click “Explore data” for plots. Click “Save and continue” to move on
17. (Plate normalization) Pick a normalization method e.g. “negative” to normalize every sample against the negative controls, first a preview is shown. Choose another normalization or when satisfied click “Save and continue”
18. (Parameter transformation) For this data set it is not necessary to transform any parameters, click “Proceed” or “Skip data transformation”. When you do want to transform, select the parameter for transformation and click “Transform”. All parameters will pass by one by one to transform them in their specific way e.g. “Log” or “square root” (sqrt), click preview and when satisfied click “Transform” until every selected parameter is processed
19. (Include parameters for multivariate analysis) The parameters you have selected at the Parameter selection step (21) are selected, also the preferred estimator is selected. Select parameters with the status ok (green icon) and choose “apply data reduction”, then click “Include parameters”
    a. Now the missing data will be handled, the amount of time for this process is depending on the selected number of parameters with missing data

20. (Data standardization) The parameters can be standardized so they will be all in the same scale. Select the parameter or your preference and click "Explore" how the data looks. A selected standardization method will be applied to all included parameters. Click "Save and continue" after a selected normalization or click "Skip data standardization".
21. (Upload reagent identifiers) Upload reagent identifiers provide names to the unit of analysis. When there are no reagent names available, you can click "No reagent identifiers". When you do, please upload a tab/comma/semicolon separated file with in the first column the barcode, the second the location [D08], and the third column the reagent name. The following column names are required; plate, well and reagent. Upload the file by clicking "Upload", browse and select your file. When the upload succeeded, click "Continue".
    a. In the next page you can change the separator to tab, comma or semicolon. When you select an option, it will automatically preview those settings.
    b. When satisfied, click "Update and continue".
    c. Now you can change the barcode names so they will match your uploaded reagents file. If they already match, you do not have to change anything, click "Save and continue".
    d. The reagent identifiers/names will be updated. the amount of time for this process is dependent on the size of your screen.
22. (Data reduction) Depending on weather you selected "apply-" or "skip data reduction" in step 25, you will arrive at "Data reduction" or the selection of relevant parameters for hit selection, (described in step 30). The parameters that were included for further (multivariate) analysis in step 25 are now used to apply data reduction. The recommended settings are selected by default i.e. factor analysis, Kaiser criterion, oblique rotation and a correlation cut-off of 1. You can click "Apply" to start the process.
    a. It could be that your correlation cut-off is too high and it fails (it shows a red box in the right upper corner, it will also tell you why it failed) lower the correlation cut-off each time by 0.01 and try again until it succeeds.
    b. When the process is done, it will show results and plots.
    c. Lots of information and plots are shown like scree plots (to check the number of factors and eigenvalues), polar plots and distribution plots but the most important information is the communality scores, shown in the "Factor report". Click continue to proceed.

23. (Provide factor names) Give each factor a name or leave the textbox empty and click “save and continue”.
24. To explore each factor/parameter, click the factors/parameters at the bottom of the page in the dropdown box and hit “Explore”. Select all factors/parameters at the upper part of the page to include them.
25. (Hit selection) Click preview on the default settings to generate a hit list using a Manhattan distance against the negative controls with an α of .05. The hit list is shown below together with some plots. Click “” to download a complete list. Click “Continue” to move on.
26. (Merge replicates) Choose “yes” and click “Save and continue”. The graph shows the (in)consistency of the replicates.
27. (Define number of clusters) Choose default options i.e. Euclidean distance, ward and automatic, click “Save and continue”. These are the parameters that can be set for clustering.
28. (Cluster analysis) Explore the Phenotype and patterns for each reagent in a cluster heat map or multidimensional scaling method. Click “Continue” to proceed.
29. (Report) Here all the decisions and steps are visualized once more. Also, the hit list can be downloaded.

Links

- https://cla.stratominer.com
- **Data set well level:** https://cla.stratominer.com/DATA/demoData/Data_defaultDemoData.zip
- **Data set cell level:** https://cla.stratominer.com/DATA/demoData/cellLevelTest.zip
- **Reagent file:** https://cla.stratominer.com/DATA/demoData/reagentFile.csv
- **Link to video:** https://youtu.be/oAE19iTzi7o

Data Format

All requirements listed below:

- Flat files (.TXT, .CSV)
- Tab, comma or semicolon separated
- No thousand-separator
- Each row in a flat file represents a well, field or a cell
- One file per assay plate
- Replicates in the same location in separate assay plates
- Parameters required:
    - plateID parameter (must be unique)
    - well location parameter (e.g. B07)
    - at least 1 analytical parameter
- HC StratoMineR assumes biological replicates
- Consistency (each file has exactly the same number of parameters with exactly the same name)
- Multivariate analysis requires at least 2 parameters
- Data reduction requires at least 7 parameters
- Knowledge about the control locations (replicates need to be exactly similar)

If you wish to include reagent names in the analysis these can be included in a .CSV file with three columns; plate, well, reagent. Plate should correspond to plate name.

Analysis Steps

- Analytical parameters selected (74-17=54)
- Parameters with no variation or binary and categorical parameters excluded (54-10=44)
- Parameters with a uniform distribution excluded (44-3=41)
- Trimmed mean (95% CI) for each well computed (and other estimators)
- Controls set
- Plate normalized against negative controls
- Transformations applied to parameters with a significantly skewed histogram,
    - positive skewness is treated with a log transformation
    - negative skewness is treated with a square transformation
- Missing values imputed by applying bootstrapping with replacement and Bayesian network analysis. The left missing values are imputed using replicates (no parameters excluded based on # missing values)

- Robust z-score per replicate standardization applied
- Factor analysis applied using 40 parameters
    - No parameters excluded based on non-invertible (singular) matrix
    - Oblique rotation (Oblimin)
    - 7 factors retained
    - Maximum likelihood
    - Bartlett scores are computed
- 4 factors are selected based on their enrichment in mitotic (related) GO-terms
    - List of reagent names is ordered based on one specific factor in a descending fashion, this order is used to enrich each individual factor using Gene Ontology enRIchment anaLysis and visuaLizAtion tool (GO-rilla)
- Hit selection applied using 4 factors
    - Manhattan distance (plate based)
    - Using the negative controls
    - Median of distance scores from replicates for each reagent
    - P-value ≤.05 selected based on Gaussian distribution of median distance scores
    - Extreme inconsistent outliers removed (1.5 * IQR)
- Clustering
    - Hits clustered using Manhattan distance (563 hits excluding controls selected)
    - Ward's agglomerative method
    - Hierarchical dendrogram added
    - K-means applied with 5 clusters manually explored
- Enrichment (563 out of 622 reagent names known by String-db)
    - Each cluster enriched for pathways
    - 2 out of 5 clusters found with (related) mitosis pathways (341 reagents selected) (evidence view, text mining disabled)

Chapter 4 - Combining supervised and unsupervised machine learning methods for phenotypic functional genomics screening

There has been an increase in the use of machine learning and artificial intelligence (AI) for the analysis of image-based cellular screens. The accuracy of these analyses, however, is greatly dependent on the quality of the training sets used for building the machine learning models. We propose that unsupervised exploratory methods should first be applied to the data set to gain a better insight into the quality of the data. This improves the selection and labelling of data for creating training sets before the application of machine learning. We demonstrate this using a high content genome-wide siRNA screen. We perform an unsupervised exploratory data analysis to facilitate the identification of four robust phenotypes, which we subsequently use as a training set for building a high-quality random forest machine learning model to differentiate four phenotypes with an accuracy of 91.1% and a kappa of 0.85. Our approach enhanced our ability to extract new knowledge from the screen, when compared with the use of unsupervised methods alone.

Adapted From:

Omta, W.A., van Heesbeen, R. G., Shen, I., Nobel, J., Robers, D., van der Velden, L.M., Medema, R.M., Siebes, A.P.J.M., Feelders, A. J., Brinkkemper, S., Klumperman, J., Spruit, M.R., Brinkhuis, M.J.S. & Egan, D. A. (2020). Combining Supervised and Unsupervised Machine Learning Methods for Phenotypic Functional Genomics Screening. SLAS DISCOVERY: Advancing the Science of Drug Discovery, 2472555220919345.

Abbreviations

GPU = Graphics Processing Unit
CPU = Central Processing Unit
RF = Random Forest
NN = Neural Network
DCNN = Deep Convolutional Neural Network
KS = Kolmogorov Smirnov
SVM = Support Vector Machine
HCS = High Content Screening
MCAR = Missing Completely at Random
FDR = False Discovery Rate
ML = Machine Learning
MB = Megabytes
GB = Gigabytes

Introduction

After a period in which the pharmaceuticals industry focused greatly on highly reductionist target-based drug discovery, the industry has now in some respects returned to its roots with a much greater emphasis on phenotypic methods (Moffat et al., 2017). The methods are often employed in more physiologically relevant cell systems such as three-dimensional patient-derived organoids (Yang et al., 2019). The major drawback of reliance on a purely phenotypic drug leads discovery approach, however, is that the target remains unknown, and the lack of a defined target makes lead optimization more difficult.

High content screening methods, when combined with multivariate data analytics methods, can give insight into mechanism of action. This is reflected in the recent interest for Cell Painting, a target agnostic phenotypic profiling method. In order to be useful for target identification, these methods are often combined with functional genomics (Bray et al., 2016).

Previously this mostly involved siRNA or shRNA gene knock-down screens, but as the limitations inherent in these methods, such as off-target effects, became apparent, they fell out of favor (Seok et al., 2018). Recently there has been more interest in CRISPR-based gene knock-out screens but as with any other new technology, these methods have their own drawbacks that are now becoming apparent (Munoz et al., 2016).

There are substantial data analytics challenges associated with leveraging the full power of high content screens. Currently the most common approach is to use image analysis software to extract numeric descriptors of cellular phenotype at either well or object, (cell or organoid), level. This need can generally be met with commercial image analysis platforms that are delivered with automated high content imagers or the open source CellProfiler platform (Carpenter et al., 2006). Other options that require more specialist expertise include Image J (Rueden et al., 2017) and image analysis functionality that is available in the KNIME data pipelining platform (Dietz & Berthold 2016).

The mining of the resultant numeric data sets has been more problematic. Perkin Elmer provide an adapter for the Tibco Spotfire data visualization tool called High Content Profiler. Genedata support high content in their Screener platform. In this study the data is mined using the HC StratoMineR platform (Omta et al., 2016). We have previously shown how this platform can be used to mine high content data sets using an exploratory, unsupervised data analytics workflow, in which data reduction followed by the calculation of a multidimensional distance score, allowed for the detection of phenotypic outliers. These outliers could then be subjected to hierarchical clustering to identify groups with similar phenotypes (Young et al., 2008; Omta et al., 2016; Caicedo et al., 2017).

One drawback of an unsupervised approach was that it was difficult to connect the structure within the clustering with the underlying biology. Here we seek to connect the biologist better with the reasons why certain outliers are clustering together.

An alternative approach to the unsupervised method (Young et al., 2008; Omta et al., 2016; Caicedo et al., 2017), would be to use supervised machine learning approaches (Scheeder, Heigwer & Boutros, 2018). These are popularly referred to as artificial intelligence (AI). A training set is used to build a multi-class model that can subsequently be used to classify reagents in a high content screen according to similarity to one or more interesting phenotypes.

A number of studies in the area of HCS have applied machine learning to high resolution data. Neumann et al., (2010) took 190 thousand time-lapse movies from 19 million cell divisions. Approximately 200 features were extracted using segmentation and 3000 nuclei were manually annotated. The set was used to train a Support Vector Machine (SVM) classifier with an accuracy of 87%. The 1.9 billion nuclei were classified into 1 of 16 morphological classes. Phenotypic profiles were used to classify deviations from control groups and identify relevant changes. Genome wide scores were used to flag mitotic hits.

Fuchs et al., (2010) conducted an siRNA screen in HeLa cells to generate automated HCS images stained for DNA, Tubulin and Actin. The cell body and nuclei were segmented, and features were extracted from all three channels. Finally, cells were classified using an SVM model based on eight cellular phenotype classes. This was built on a training set of 1740 cells. The measured accuracy ranged from 96.9 to 100%.

Ljosa et al., (2013) used CellProfiler to generate 453 features for 2.2 million cells from MCF-7 breast cancer cells treated with 113 different compounds at 8 concentrations. Cell data was standardized and normalized prior to analysis. Mean values, KS statistics, SVM and factor analysis were used to calculate profile values for each treatment. Then a mechanism of action score was generated from the calculated statistics.

Advanced Cell Classifier and the Analyst module of CellProfiler allow for the annotation of machine learning classes by directly selecting cell images (Pivvinini et al., 2017; Dao et al., 2016). No matter what populations are chosen, the quality of the analysis is heavily dependent on the quality of the training set used. We hypothesized that the unsupervised data analytics pathway (Young et al., 2008; Omta et al., 2016; Caicedo et al., 2017), would be useful for the generation of a high-quality training set that could then be successfully used to build an effective machine learning model. In this study, the approach of unsupervised analysis followed by a supervised analysis is carried out on a data set that was previously analyzed (Omta et al. 2016; van Heesbeen et al., 2016). We show that the combination of unsupervised and supervised data analytics methods has the potential to enhance the ability to identify new knowledge in functional genomics screens.

Materials and Methods

Wet-lab protocol & data set

The data set used in this study is a genome-wide High Content siRNA Screen that was performed to identify novel regulators in mitosis. In short, a

Dharmacon (Lafayette, CO, USA) genome-wide ON-TARGETplus siRNA SMARTpool library was transfected in HeLa cells in 384-well microplates, (1500 cells per well). After fixation, the cells were stained with diamidino-2-phenylindole (DAPI) after siRNA knock-down for the identification of the nucleus and an antibody against phosphorylated histone H3 to identify cells in mitosis.

Images were acquired using a Thermo (Waltham MA, USA) Array Scan VTi and numeric features for each cell were extracted using the Cellomics Target Activation/Morphology Explorer image analysis software. The methods used in this screen are described in much greater detail in van Heesbeen, et al. (2016) and Omta et al. (2016). The data set contains ~46 million records, each record represents a single cell, consisting of 74 features (Omta et al.,2016; van Heesbeen et al., 2016). In order to carry out data analysis, the data were exported to flat files, one per microplate. The data was available in two resolutions; low (well averages, each record is a well) and high (object level, each record is a cell) (and in two cell lines).

Data preprocessing

Preprocessing data is a very tedious but important part of a data analysis process (Omta et al.,2016; van Heesbeen et al., 2016; Omta et al., 2020). First, data is divided into meta features, (information about the data) and analytical features, (used for data analysis). Examples of analytical features are intensity, area, shape and texture features on various channels (Thermo Scientific, 2010, retrieved from http://www.med.cam.ac.uk/wp-content/uploads/2016/02/MorphologyExplorer_V4_LC06170800.pdf). Features with a standard deviation of 0 or containing ≥95% missing data are omitted. Those with a correlation coefficient ≥.99 are inspected for missing data and only the feature with the lowest number of missing data points is retained.

An additional feature selection is then performed by omitting features containing an equal uniform distribution across the different classes. The z-distribution of the skewness is inspected for significance (p <.001). Here, the Kolmogorov-Smirnov and Shapiro-Wilk tests are too sensitive (Royston, 1982a; Royston, 1982b; Marsaglia, 2003). Features are log transformed in cases of positive skewness and transformed using a square root in cases of negative skewness. Features are then normalized on a plate-by-plate basis by dividing each feature by the median of the negative control (scrambled siRNA). Finally, features are scaled using a robust z-score (Birmingham et al., 2009).

A significant MCAR (Missing Completely at Random) outcome results in case-wise deletion for missing data, while an insignificant MCAR outcome is handled by imputation methods, e.g. regression, random forest or predictive mean matching (Buuren & Groothuis-Oudhoorn, 2010; Little, 1988). The method described by Young et al., (2008), Omta et al., (2016) and Caicedo et al., (2017) focuses on the numeric data analysis after preprocessing, as described above to identify hits. The number of features that are left over after preprocessing can subsequently be included for carrying out further analysis i.e. exploratory, descriptive- or predictive analysis. All the analysis results were generated using R and HC StratoMineR (Ripley, 2001; Omta et al., 2016). All data analyses were carried out on an AWS EC2 r5.xlarge with an Intel Xeon Platinum 8000 series, 4 cores and 32 GB of RAM. This hardware was used because this can be compared to a standard modern laptop.

Results

In this study the siRNA data set was reanalyzed with a similar strategy to that used in the original study (Omta et al., 2016; van Heesbeen, et al. 2016), followed by a supervised machine learning approach. The complete data analysis workflow in this paper was carried out in four stages; stage A (Exploratory Data Analysis) is an unsupervised approach (figure 1A), stage B (Annotation) involves the annotation of the data in preparation for stage C (figure 1B). Stage C (Predictive Data Analysis) is a supervised machine learning stage (figure 1C). In stage D (Evaluation) the results of stage C are evaluated (figure 1D). The data set used in these stages contains 41 useful features that were extracted from the DAPI and pS10-H3 channels.

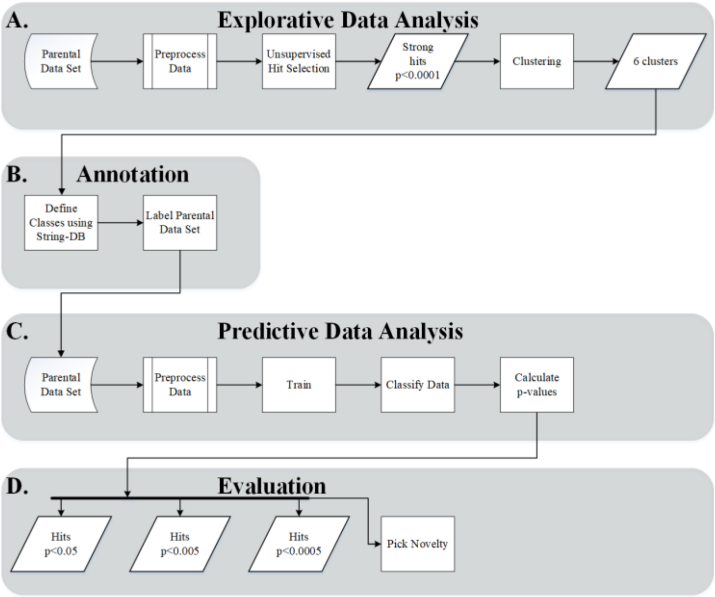


Figure 1. *The process of combining unsupervised with supervised analysis*

A. Describes the exploratory data analysis using well-resolution data carried out in an unsupervised fashion.

B. Describes the annotation process where classes are being labeled using the results of A and using a GO-term analysis from String-DB

C. Describes the predictive analysis stage. A supervised machine learning model is trained based on the data that was annotated in stage B

D. Describes the aggregation of the results and the evaluation of the four hit lists that are generated

Stage A: Exploratory Data Analysis

The first stage in the data analysis workflow is an exploratory data analysis stage or unsupervised approach, (figure 1A) using well-level resolution data, that was carried out as described in Omta et. al. (2016). The data in this stage only contains ~47 thousand records per cell line (two cell lines). After preprocessing, data reduction was carried out using common factor analysis (CFA). CFA generated 5 common factors which were then used to calculate a Euclidean distance score from the median of the negative controls. Also, p-values were calculated and are based on the negative controls and corrected with FDR in order to avoid type II errors. For each screened siRNA pool, the difference between the distance scores from the Parental and EIC cell-lines was calculated. Those that came from wells that had a p-value <.05 in the Parental cell line were chosen as hits. The genes targeted by these 154 siRNA pools were analyzed in String DB (Szklarczyk et al., 2015). As we have seen previously in our analysis, the hit list was enriched for genes involved in mitosis, but also ribosomal genes,

genes related to proteasomal degradation and splicing genes. In this and subsequent String-DB analyses we followed 4 Biological Process GO-terms, (i) GO:1903047 or mitotic cell cycle process which we refer to as “Mitosis” (red dots in figure 2), (ii) GO:0006413 or translational initiation, referred to as “Ribosome” (green dots in figure 2), (iii) GO:0000398 mRNA splicing, via spliceosome, referred to as “Splicing” (purple dots in figure 2) and (iv) GO:0016579 protein deubiquitination, referred to as “Proteasome” (yellow dots in figure 2). Within our 154 hits it was clear that mitosis genes were poorly enriched, (15 genes, FDR = 0.001) compared to Splicing (30 genes, FDR = 7.21E-22), Proteasome (16, FDR = 3.92E-08), and Ribosome (25 genes, FDR = 1.12E-22). This would suggest that looking for novel mitosis genes in the unconnected or gray nodes would be far more likely to deliver genes involved in one of the other processes (see figure 2).

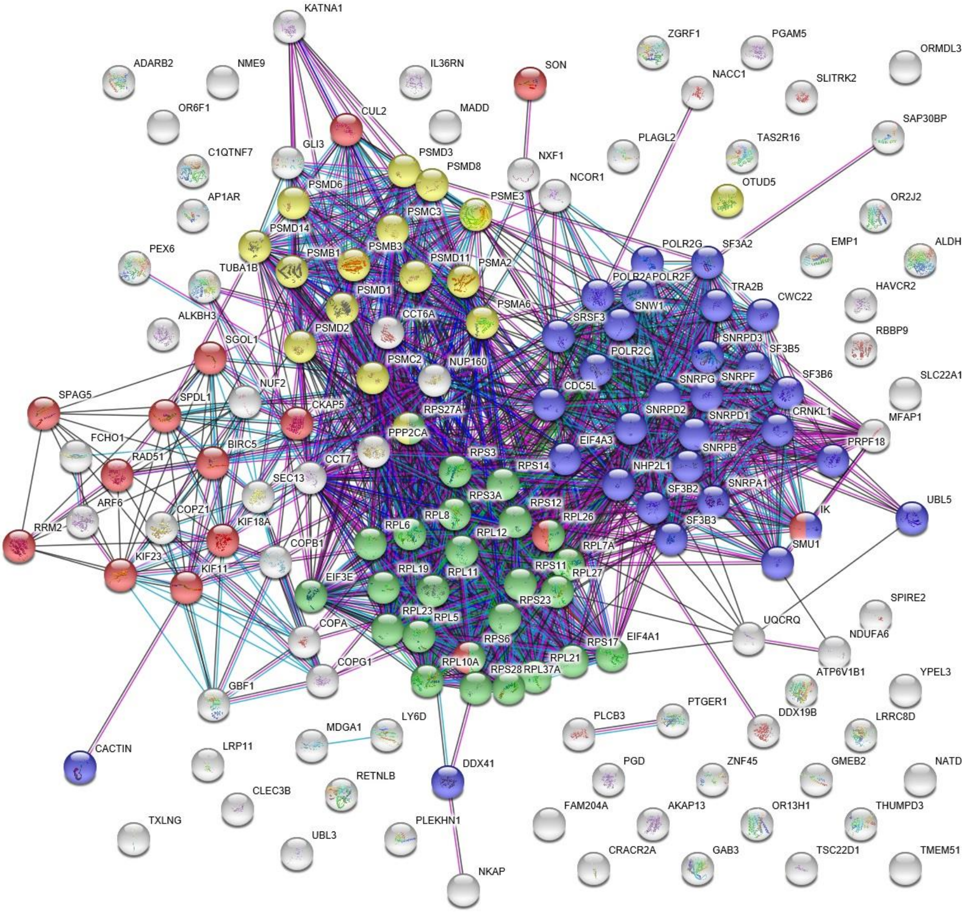


Figure 2. *Interaction of a set of genes from the analysis of the parental vs. the EIC cell line*
The interaction of a set of 154 genes that resulted from the analysis of comparing the parental cell line to the EIC cell line was visualized using String-DB. Four pathways in this set were found significantly

enriched; (i) mitotic cell cycle process (Mitosis, GO:1903047) (15/154 genes, FDR = 0.001) visualized in red, (ii) translational initiation (Ribosome, GO:0006413) (25/154 genes, FDR = 1.12E-22) visualized in green, (iii) mRNA splicing, via spliceosome (Splicing, GO:0000398) (30/154 genes, FDR = 7.21E-22) visualized in purple and (iv) protein deubiquitination (Proteasome, GO:0016579) (16/154 genes, FDR = 3.92E-08) visualized in yellow. Gray items are unknown to the ontology.

In a previous study, hierarchical clustering was used to identify groups of genes that were highly enriched for mitotic cell cycle genes (Omta et al., 2016). The addition of supervised machine learning functionality now gave the opportunity to use a new strategy which could combine the unsupervised and supervised data analytics approaches to address the problem.

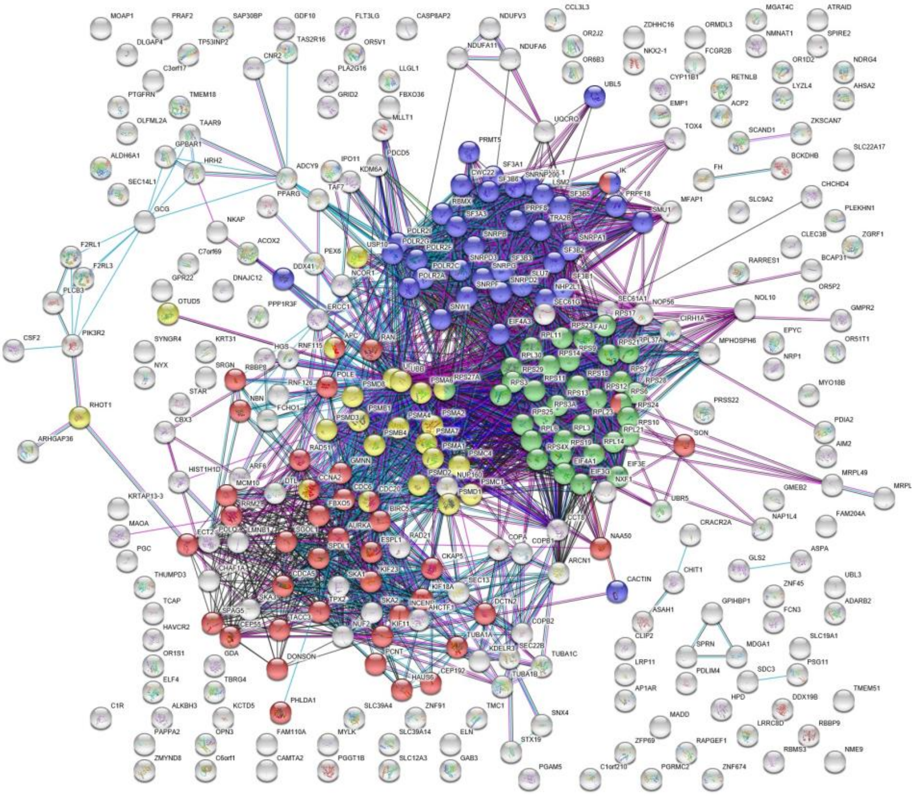


Figure 3. *Interaction of a set of genes from the results of the unsupervised analysis of the parental cell line*

The interaction of a set of 344 resulting genes from the unsupervised analysis of the parental cell line was visualized using String-DB. Four pathways in this set were found significantly enriched; (i) mitotic cell cycle process (Mitosis, GO:1903047) visualized in red, (ii) translational initiation (Ribosome, GO:0006413) visualized in green, (iii) mRNA splicing, via spliceosome (Splicing, GO:0000398) visualized in purple and (iv) protein deubiquitination (Proteasome, GO:0016579) visualized in yellow. Gray items are unknown to the ontology.

A higher resolution Parental data set was available at cell resolution as opposed to well-averaged data. We used the distance based unsupervised method, to identify 344 very strong hits. siRNAs that showed a significant difference (*p*< 0.0001) from the negative control (scrambled siRNA) using a multiparametric distance score (Caicedo et al., 2017; Young, 2008; Omta, 2016) (see figure 1A). These were subjected to hierarchical clustering in combination with K-means clustering in which 6 clusters were generated (see figure 1A). Analysis of these in String-DB highlighted 4 clusters that were highly enriched for Mitosis, Splicing, Proteasome & Ribosome genes (see figure 3). From the hierarchical cluster analysis and K-means clustering, each cluster was submitted to String-DB separately (see Supplementary Data S1, S2, S3 & S4). Cluster 2 (Supplementary Data S2) was clearly most enriched for Mitosis genes, (14 of 48 genes, FDR = 4.39E-09) and Cluster 5 (Supplementary Data S4) for Ribosomal genes, (30 of 95 genes, FDR = 9.25E-36). Interestingly the Splicing and Proteasome genes proved more difficult to separate but were both distributed across Cluster 3 (Supplementary Data S3) (Splicing, 11 of 69 genes, FDR = 3.37E-12; Proteasome, 11 of 69, FDR = 0.0011 and Mitosis 13 of 69 genes, FDR = 1.08E-05). Cluster 1 (Supplementary Data S1) (Splicing, 15 of 55, FDR = 1.76E-12; Proteasome, 7 of 55 genes, FDR = 0.00089). It was notable that Cluster 1 was centered around UBC, (ubiquitin-C) (see supplementary data S1), whereas, Cluster 3 was centered around UBB, (ubiquitin-B) (see supplementary data S3). Cluster 1 also had significant enrichment for Mitotic genes (7 of 55, FDR =0.0233) (see Supplementary Data S1). This information was taken from the String-DB Homo Sapiens Process ontology. The enrichment and annotation were verified using Gorilla (Eden et al., 2009) and confirms our findings with an FDR of 4.64E-02 of Mitosis in Cluster 2, an FDR of 1.22E-5 of Ribosome in cluster 5 and an FDR of 4.99E-02 of Splicing in cluster 1.

Stage B: Annotation

We randomly chose a total of 52 genes (~124,000 cells) from the hit list containing 344 genes that was generated in stage A to label the data and to create a training set for training a four-class classifier model at single cell level in stage C (see figure 1C). The resulting GO-Terms from String-DB were used to annotate the siRNAs that showed significant involvement in the four identified pathways in stage A (see figure 1A). We chose 15 Mitosis genes (GO:1903047) from Cluster 2 for the Mitotic class (~36,000 cells) and 13 Ribosome genes (GO:0006413) from Cluster 5 for the Ribosome class (~31,000 cells). Because Splicing (GO:0000398) and Proteasome (GO:0016579) genes are both across

Clusters 1 and 3 we decided to build a ProteaSplice class using 24 Proteasome & Splicing genes (~57,000 cells). The fourth class is a rest class and is introduced with the ability to capture cells belonging to neither of the three pathway classes. The rest class is a scrambled siRNA and labelled as NEGATIVE that was originally already present in the data set.

Stage C: Predictive Data Analysis

Using the results of the unsupervised approach to label the object-level resolution data with three additional training classes as described in stage B is then followed by a supervised machine learning approach (see figure 1C). In the predictive data analysis or supervised approach (Stage C), data at the object-level is used and contains ~57 million records. Instead of calculating a distance score, as previously has been done with well resolution data (Caicedo et al., 2017; Young, 2008; Omta, 2016), a classification algorithm is used in stage C, using data at object-level.

To explore the possible classification and feature driven approaches, a preliminary analysis was conducted including three classification algorithms. In addition, PCA, ICA or the original feature set was used for building classification models (see column Features/PCA/ICA in Table 1). PCA and ICA are both methods to reduce the dimensionality. These methods support the reduction of redundancy, bias, required computational power and attempt to avoid the curse of dimensionality (Pechenizkiy, Puuronen & Tsymbal, 2006). The results of the preliminary analysis can be seen in Table 1.

The features are simply the original features available in the data set that were treated as described in the data preprocessing section. The Principal Component Analysis (PCA) approach implies the same preprocessing approach and the creation of 7 principal components (based on the elbow method) using a generalized least squares approach (Clavel, Escarguel & Merceron, 2015). The Independent Component Analysis (ICA) approach also implies the same preprocessing treatment and the creation of 7 components using a non-linear method (Hyvärinen & Oja, 2000). For this preliminary analysis, binary classification models were trained (Parmigiani, 2001) using built-in NEGATIVE and POSITIVE controls to explore the options of the original feature space or dimensionality reduction in combination with three classification algorithms.
In all 9 scenarios, 50k records were randomly sampled with replacement out of the total set of ~325k records (Arabatzis & Burkhart, 1992) of data containing the

label NEGATIVE or POSITIVE. The 50k records were then split into an 80% train- and a 20% test set. For the optimization of the hyper-parameters, a 4-fold cross-validation was applied to the train set (Forman & Scholz, 2010).

Table 1. Exploration of Algorithms and the Data Feature Space

Scenario	Classification Algorithm	Features/PCA/ICA	Accuracy	Kappa	Time in Sec.
1	Random Forest (mtry = 20)	Features	96.01%	0.917	106.616
2	Random Forest (mtry = 4)	PCA	86.44%	0.729	35.302
3	Random Forest (mtry = 4)	ICA	85.91%	0.7183	36.219
4	Support Vector Machine (sigma = 0.03768469, C = 43.53546)	Features	96.02%	0.920	1517.326
5	Support Vector Machine (sigma = 0.0657829, C = 477.636)	PCA	86.82%	0.736	3460.237
6	Support Vector Machine (sigma = 0.1616871, C = 43.53546)	ICA	86.63%	0.7327	2491.188
7	Neural Networks (size = 20, decay = 1.044708e-03)	Features	96.57%	0.931	96.339
8	Neural Networks (size = 20, decay = 1.044708e-03)	PCA	86.43%	0.729	49.987
9	Neural Networks (size = 20, decay = 1.044708e-03)	ICA	86.03%	0.7207	53.248

A random search grid was created to find the optimal hyper-parameter settings for all 9 scenarios (Bergstra & Bengio, 2012) (see Table 1). For Random Forests (Breiman, 2001), trees were constantly kept at 128 trees (Oshiro, Perez, et al, 2012) and the hyper-parameter mtry was tuned (see Table 1). For SVM (Schölkopf et al., 2000), a radial kernel was chosen (Chang & Lin, 2011) and the hyper-parameters sigma and C were tuned to optimize the performance of SVM. For Neural Networks (Ripley, 1996), an architecture of one layer was used. The hyper-parameters size and decay were tuned. The hyper-parameter size implies the number of nodes in the hidden layer. The hyper-parameter decay implies the penalty for the size of the weights. The output of tuning the NN hyper-parameters size and decay were the same for all three cases (features, PCA and ICA).

The hyper-parameters for Support Vector Machine (SVM) and Neural Network (NN) are hard to tune and to understand compared to the hyper-parameter of Random Forest (RF). SVM performs well but is very slow. Our final decision is RF because of the ease of use (minimal hyper-parameter tuning), the speed of the algorithm (see Table 1) and the relative low chance of overfitting (Cutler, Cutler & Stevens, 2012). We choose single features over a dimensionality reduction method because RF builds trees based on a random set of features (mtry). RF is not very sensitive to the number of random features (mtry), the total number of features and the tree size with respect to overfitting (Cutler, Cutler & Stevens, 2012). This work was carried out in R using the packages nnet, e1071 and randomForest.

For the classification and identification of our siRNAs and targets, a supervised multiclass machine learning model using Random Forest (Datta and Pihur, 2010) is built based on the four classes described in stage B. Each single cell can be classified in one of these classes according the model using the feature space of the data set. Stratified data sampling was carried out because of variation in class size (see stage B).
The multinomial Random Forest model is applied in classification mode and trained using 128 trees (Oshiro, Perez, et al, 2012). The labeled data, containing ~430k cells is split into 80% training and 20% for testing (Giacomelli, 2013). The model was trained using ~345k data points in 4-fold cross validation and shows a substantial Kappa agreement score, which corrects for agreement expected by chance (Cohen, 1960), between the observed and predicted classes of 0.8517, and an accuracy of 91.1% (García et al., 2009; Cohen, 1960). A random search grid was created to tune the Random Forest Model to find the ideal hyper-parameter setting. The hyper-parameter setting mtry=10 was finally found to be the best option.

The Kappa score can be explicitly important to data sets with imbalanced or skewed classes (see stage B). Table 2 shows the resulting confusion matrix of the classification model. A confusion matrix allows for an indication of the model's accuracy in classifying the test data set. Each row represents the actual class of the cells and each column represents the predicted class of the cells. Each number represents the percentage of cells predicted within the grid. The diagonal in bold shows the percentages that are predicted correctly.

Table 2. *Cellular resolution multiclass classification*

The rows represent the actual class of the cells, the columns represent predicted class of the cells. The diagonal shows the cells and percentages of correctly classified cells. This table is the result of the classification model applied to the test set containing a total of 86372 cells.

		Predicted Class (cellular resolution)				
		Mitosis	**NEGATIVE**	**ProteaSplice**	**Ribosome**	**Total**
Actual Class (cellular resolution)	**Mitosis**	**92.65%** **(10427)**	4.18% (470)	2.11% (238)	1.06% (119)	**100%** (11254)
	NEGATIVE	0.87% (437)	**92.50%** **(46729)**	4.72% (2384)	1.92% (969)	**100%** (50519)
	ProteaSplice	1.09% (178)	11.50% (1877)	**86.06%** **(14047)**	1.35% (220)	**100%** (16322)
	Ribosome	0.68% (56)	7.21% (597)	1.73% (143)	**90.38%** **(7481)**	**100%** (8277)

Stage D: Evaluation

We applied this model to the whole Parental data set. Each cell in every well across the data set was classified into one of the 4 classes. This information was then aggregated using medians and standard errors as an estimator per well to generate the probabilities of each well being a member of the four classes. We ranked all the wells according to the probability that they were in the Mitotic class and generated four hit lists based on p-value cut-offs of 0.05, 0.005, 0.0005, & 0.00005 with 16%, 23%, 29% & 38% of Mitosis genes respectively see figure 4A. Figure 4B demonstrates the same data but in absolute numbers.

In order to determine the likelihood that our approach would make it easier to identify novel regulators of mitosis we carried out a simple search in Pubmed for [Gene Name] AND Mitosis for the genes in the p< 0.0005 list, not assigned to any of our key GO groups. Four, FAM110 (Hauge, Patzke & Aasheim 2007), Rec114 (Stanzione, 2016), UBR5 (Jiang, 2015) and NKAP (Li et al., 2016) were found to have been recently reported to be involved in mitosis or meiosis.

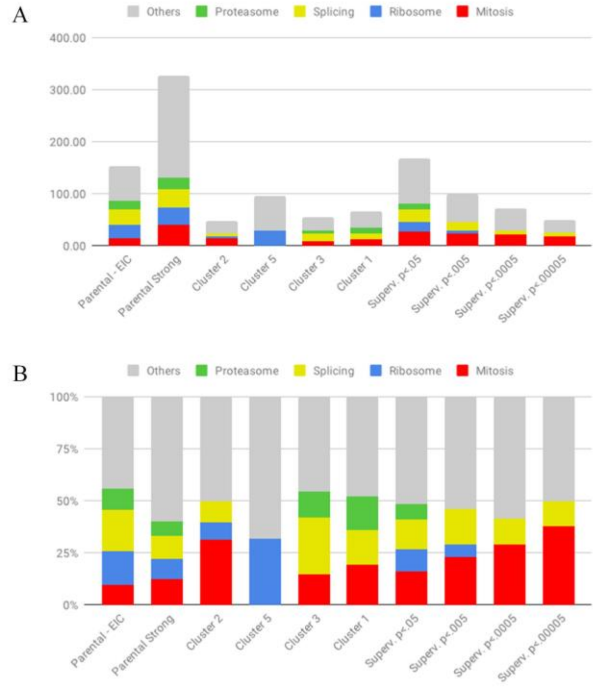


Figure 4. *Results classification*
A. The x-axis represents the data sets, the y-axis represents the absolute number of genes in the set. The colors represent the four identified pathways, annotated as classes in the data
B. The x-axis represents the data sets, the y-axis represents the percentage of genes in the set. The colors represent the four identified pathways, annotated as classes in the data

Discussion

Our study has demonstrated how the combination of unsupervised and supervised machine learning can greatly enhance the efficiency with which new knowledge can be extracted from functional genomics screens. Our original analysis, relying solely on unsupervised methods resulted in a hitlist that was overwhelmed with hits that were of little interest since they were already known to be involved in the core machinery of protein translation, degradation and RNA splicing. It could well have been that interesting hits with novel mechanisms of action could have been found in this, but these would have been difficult to identify.

The unsupervised analysis did prove to be very useful however, identifying hits using a multi-class random forest model allowed for the generation of hit lists that were far more heavily enriched in genes that were centrally involved in mitosis.

In this case this approach could potentially have been used to generate more information from a genome-wide screen that only generated a single publication on three genes that were already known to be involved in mitotic spindle assembly. Confirming that there are novel regulators of mitosis in the supervised machine learning hit lists is unfortunately beyond the scope of this study but we believe that our approach gives biologists the opportunity to be able to deal better with the challenges of validating and characterizing hits from functional genomics screens.

Results of unsupervised machine learning can be used as input for rich data visualizations. In-depth data exploration using these visualizations allows for identifying patterns, systematic errors, false positives and outliers in order to add labels to subpopulations and add value to the data set. These manual annotations are invaluable for training a supervised machine learning model. The supervised model can be trained using the annotations and can be applied to classify new (unseen) data.

The classification result contains a probability score for each class in each record. Each probability score represents the likelihood that a record belongs to a class. The sum of the probabilities of the classes for each record is equal to 1. The set of probabilities represents a matrix *pm* and contains as many records as the classified data set. The number of columns of matrix *pm* is equal to the number of classes that are included in the supervised classification model.

Supplementary Data S5 demonstrates a hypothetical example of the output of a three-class classification model in a probability matrix (*pm*). The matrix *pm* is visualized using a heatmap and is clustered using an unsupervised method to organize the data according to similarity. When a cluster contains records with equally distributed probabilities among the classes (~0.33), we hypothesize that the cluster belongs to a fourth class. This approach of combining unsupervised and supervised machine learning can potentially be used to generate new classes and identify new phenotypes.

One major challenge in phenotypic screening is how to get an insight into what distinguishes different phenotypes. Using the unsupervised fashion, we can create groups of phenotypes using a combination of hierarchical clustering and common factor analysis or principal component analysis profiles. In extreme cases, looking at the images is enough to be able to define what is different e.g. a cluster of toxic reagents. In many cases however, the differences are subtle and hard to define. Tracking back through principal components to identify extracted features that are contributing to phenotypic differences is also not efficient.

However, through the feature importance plot from a supervised machine learning model, one can observe what the key differentiating features are for a set of classes, (see supplementary Data S6). This gives an immediate insight into the biology, especially in a two-class model where one of the classes is based on the negative controls. For screens where the goal is to identify reagents that give one phenotype, this would allow a screener to simplify feature extraction, by limiting it to the critical features and reducing the abundance of redundant features that can be extracted nowadays. This could be especially useful for screens based on the Cell Painting method (Bray et al., 2016).

One possible improvement to the proposed method in this paper would be to allow users to build supervised machine learning models on subpopulations of cells within wells and not just on all-cells-in-well populations. In commercial platforms this is currently possible at the image level, for example in PE Columbus by clicking on individual cells and assigning them to individual classes. The recently introduced Phenoglyphs functionality in GE's IN Carta platform allows a user to define the classes in a more iterative fashion. The Classifier function in the open source CellProfiler Analyst (Dao et al., 2016) offers similar functionality as does the open source Advanced Cell Classifier (Piccinini et al., 2017) platform.

Machine learning methods have long been applied in the analysis to high content data sets, but this has almost exclusively been in a post-hoc analysis, by data scientists writing project-specific scripts. The availability of AI functionality in the StratoMineR platform and the other tools described above gives the screener the ability to leverage the power of AI, but it is critical that the screener can validate the quality of the generated model.

The use of convolutional neural networks to classify high content images directly can be done using Deep Learning. It allows for an alternative approach without the intermediate step of conducting feature extraction and is attracting much attention (Kraus et al., 2017). The success of these approaches however will also highly depend on the quality of the training sets used and the available quantity of data in each training class. We believe that this will require an analogous method to the one we describe here.

Conflict of interest: WAO and DAE are founders and shareholders of Core LIfe Analytics, a company that develops and markets data analytics software including HC StratoMineR, which has been used in this work.

Supplementary Data

Supplementary Data S1

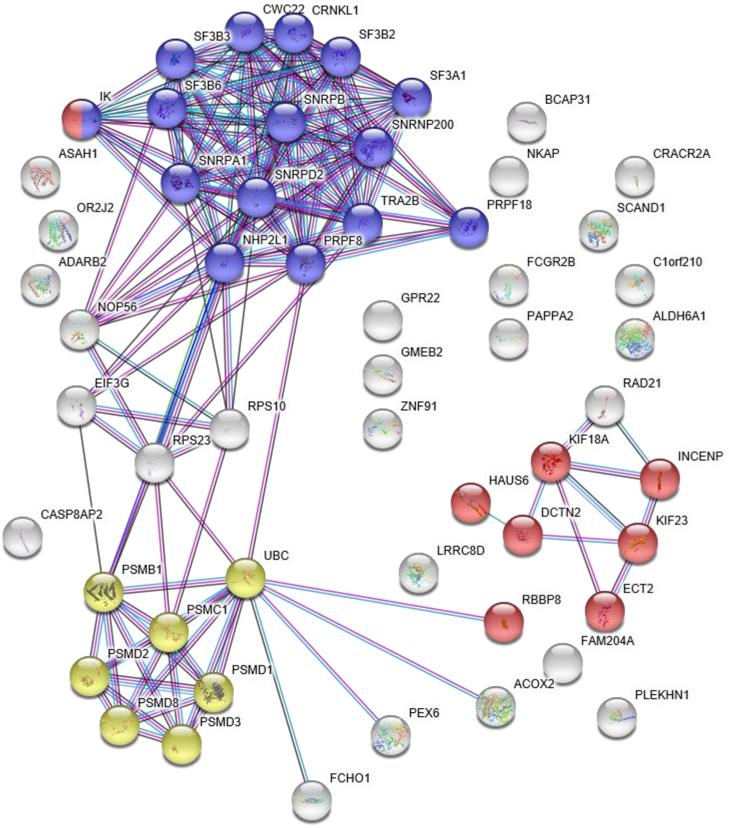


Figure S1. *Cluster 1*
The interaction of a set of 55 genes representing cluster 1, centered around UBC. Out of 55 genes, 7 have significant enrichment for Mitosis (FDR = 0.0233). Mitotic cell cycle process (Mitosis, GO:1903047) is visualized in red. Out of 55 genes 15 have significant enrichment for Splicing (FDR = 1.76E-12), mRNA splicing, via spliceosome (Splicing, GO:0000398) is visualized in purple. Out of 55 genes, 7 have significant enrichment for Proteasome (FDR = 0.00089), protein deubiquitination (Proteasome, GO:0016579) is visualized in yellow. Gray items are unknown to the ontology.

Supplementary Data S2

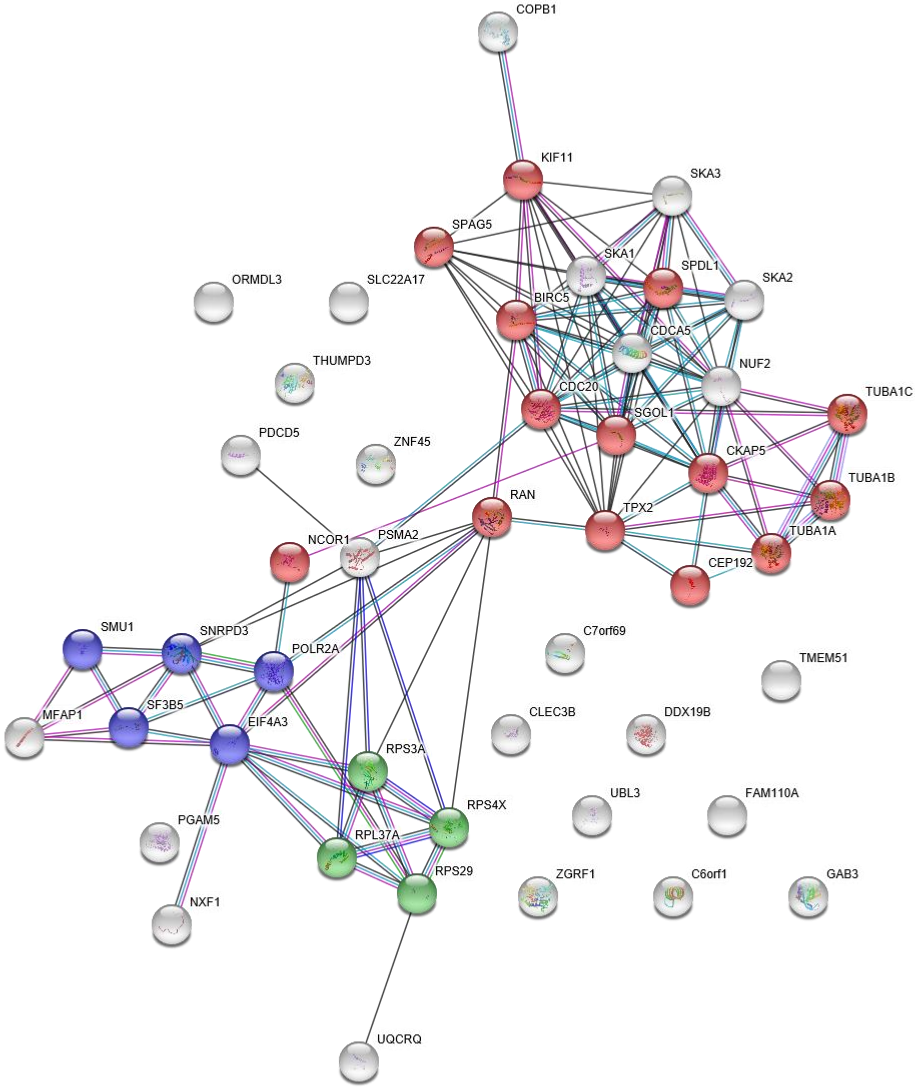


Figure S2. *Cluster 2*

The interaction of a set of 48 genes representing cluster 2. Out of 48 genes, 14 have significant enrichment for Mitosis (FDR = 4.39E-09). Mitotic cell cycle process (Mitosis, GO:1903047) is visualized in red, mRNA splicing, via spliceosome (Splicing, GO:0000398) is visualized in purple and translational initiation (Ribosome, GO:0006413) is visualized in green. Gray items are unknown to the ontology.

Supplementary Data S3

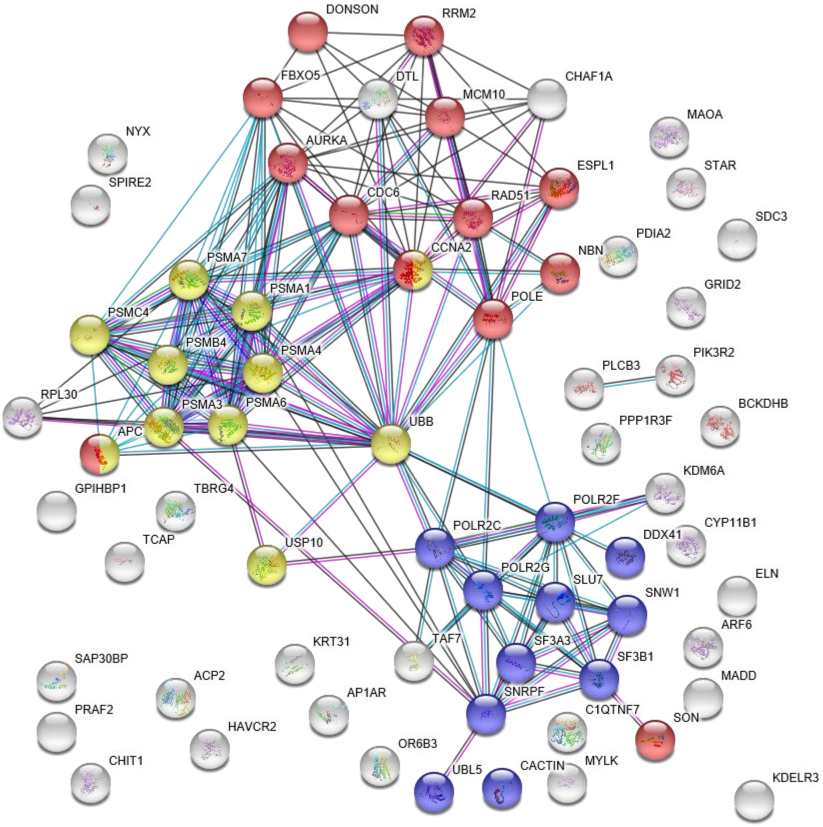


Figure S3. *Cluster 3*

The interaction of a set of 69 genes representing cluster 3, centered around UBB. Splicing and Proteasome genes were both distributed across Cluster 3. Out of 69 genes, 11 have significant enrichment for Splicing (FDR = 3.37E-12) and 11 out of 69 genes have significant enrichment for Proteasome (FDR = 0.0011). Out of 69 genes, 13 are enriched for Mitosis (FDR 1.08E-05) Mitotic cell cycle process (Mitosis, GO:1903047) is visualized in red, mRNA splicing, via spliceosome (Splicing, GO:0000398) is visualized in purple and protein deubiquitination (Proteasome, GO:0016579) is visualized in yellow. Gray items are unknown to the ontology.

Supplementary Data S4

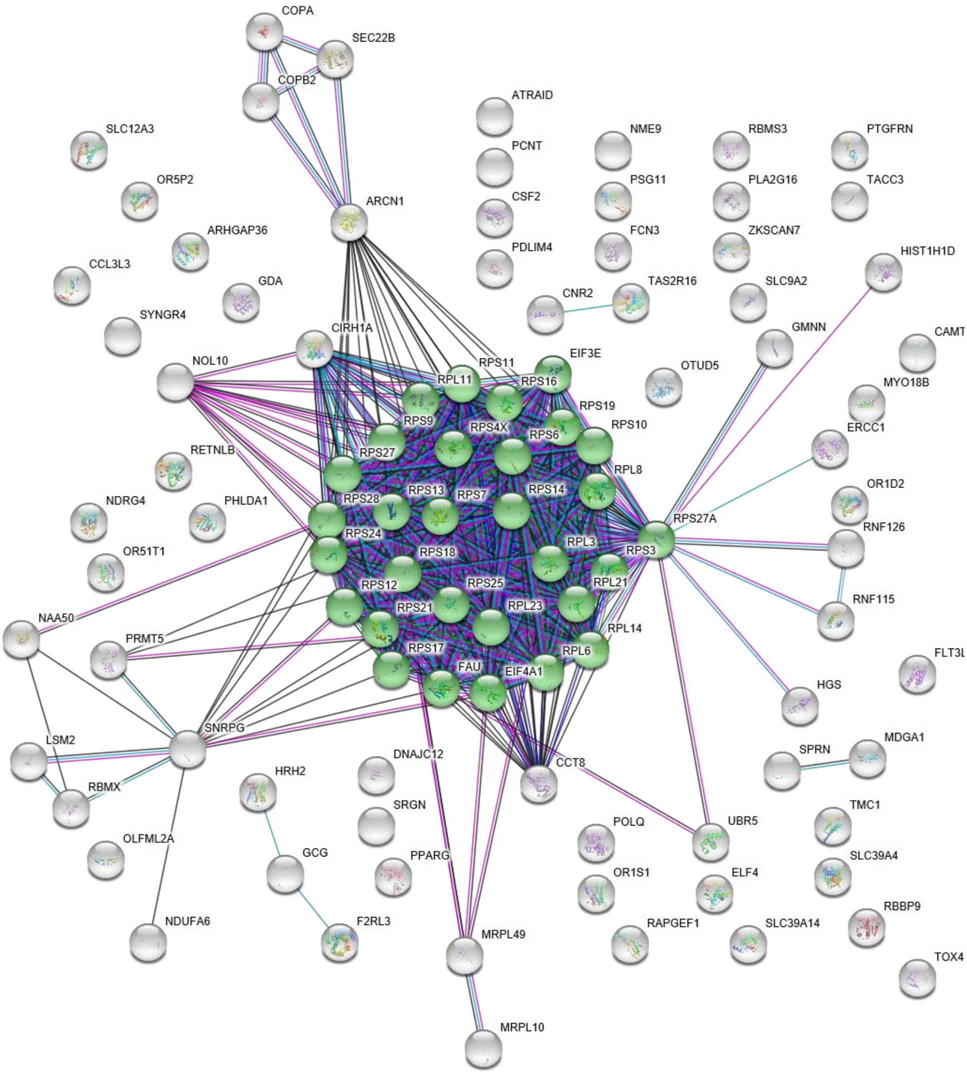
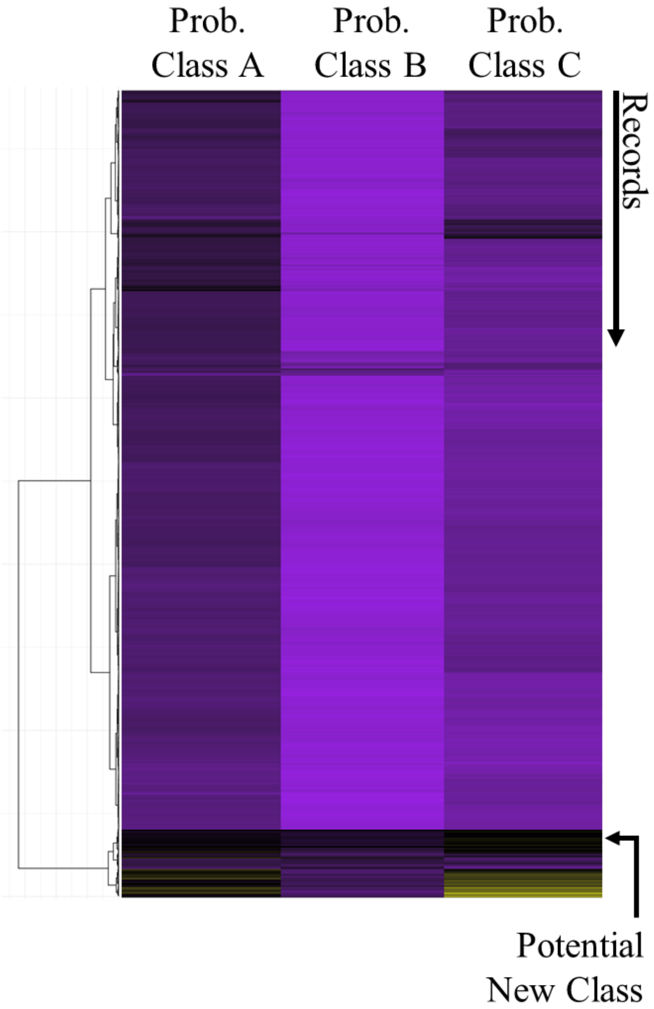


Figure S4. *Cluster 5*

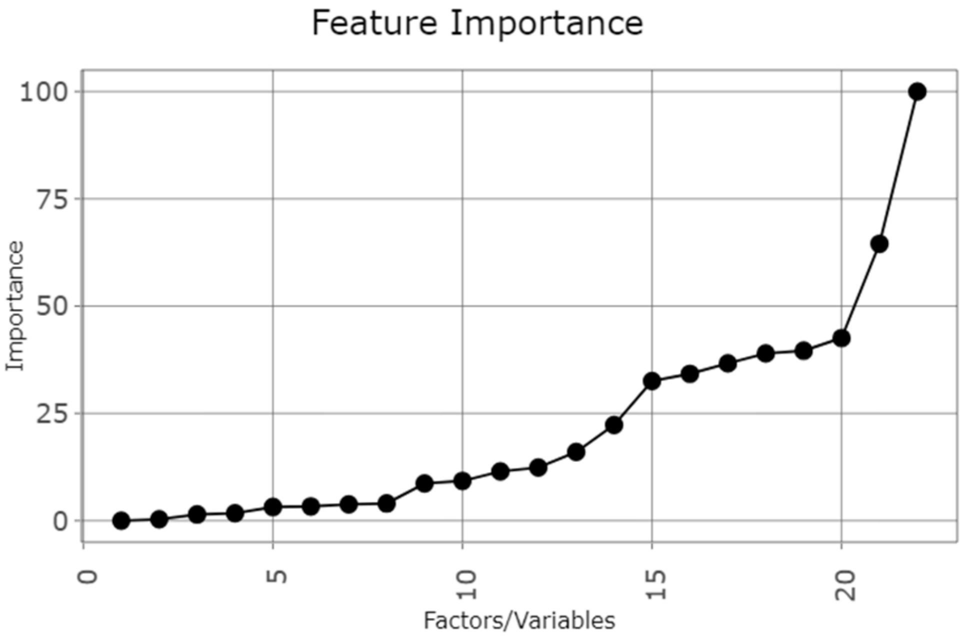
The interaction of a set of 95 genes representing cluster 5. Out of 95 genes, 30 have significant enrichment for Ribosome (FDR = 9.25E-36). Translational initiation (Ribosome, GO:0006413) is visualized in green. Gray items are unknown to the ontology.

Supplementary Data S5



Supplementary Data S5. *Heatmap containing probabilities*
This visualization contains three columns and they represent hypothetical probabilities of a classification; Class A, Class B and Class C, each horizontal line represents a classified record. Purple means low, black means moderate and yellow means high. The New Class is a cluster of records that potentially belongs to a fourth class that was not included in the classification model.

Supplementary Data S6



Supplementary Data S6. *Feature Importance Plot*

This visualization is the feature importance plot showing the usefulness of a feature for training these 4 classes; Mitosis, NEGATIVE, ProteaSplice and Ribosome and separating them successfully.

Top 10 features:

1. *MorphologyV3CellTotalIntenCh2*
2. *MorphologyV3CellObjectVarIntenCh1*
3. *MorphologyV3CellMemberObjectAreaDiffCh2*
4. *MorphologyV3CellNeighborAvgDistCh1*
5. *MorphologyV3CellObjectSkewIntenCh1*
6. *MorphologyV3CellObjectTotalIntenCh1*
7. *MorphologyV3CellNeighborVarDistCh1*
8. *MorphologyV3CellNeighborMinDistCh1*
9. *MorphologyV3CellObjectKurtIntenCh1*
10. *MorphologyV3CellObjectFiberWidthCh1*

Chapter 5 - Improving Comprehension Efficiency of High Content Screening Data Through Interactive Visualizations

In this study, an experiment is conducted to measure the performance in speed and accuracy of interactive visualizations. A platform for interactive data visualizations was implemented using Django, D3, and Angular. Using this platform, a questionnaire was designed to measure a difference in performance between interactive and noninteractive data visualizations. In this questionnaire consisting of 12 questions, participants were given tasks in which they had to identify trends or patterns. Other tasks were directed at comparing and selecting algorithms with a certain outcome based on visualizations. All tasks were performed on high content screening data sets with the help of visualizations. The difference in time to carry out tasks and accuracy of performance was measured between a group viewing interactive visualizations and a group viewing non-interactive visualizations. The study shows a significant advantage in time and accuracy in the group that used interactive visualizations over the group that used non-interactive visualizations. In tasks comparing results of different algorithms, a significant decrease in time was observed in using interactive visualizations over non-interactive visualizations.

Keywords: interactive visualizations, VDM, data mining

Adapted From:

Omta, W. A., Nobel, J. D., Klumperman, J., Egan, D. A., Spruit, M. R., & Brinkhuis, M. J. (2017). Improving Comprehension Efficiency of High Content Screening Data Through Interactive Visualizations. *Assay and drug development technologies*, 15(6), 247-256.

Abbreviations

MySQL = My Structured Query Language
HCS = High Content Screening
HTS = High Throughput Screening
ML = Machine Learning
PCA = Principal Component Analysis
VDM = Visual Data Mining
DM = Direct Manipulation
GUI = Graphical User Interface
OoI = Object of Interest
M = Mean
SE = Standard Error

Introduction

The enormous increase in data generation which has occurred in the last few decades has greatly challenged researchers. “Never before in history, data has been generated at such high volumes as it is today. Exploring and analyzing the vast volumes of data becomes increasingly difficult” (Keim, 2002). This statement is as true today as it was in 2002; more data has been created in the last two years than in the entire history of humanity. The rate at which data creation is happening is ever increasing and it is estimated that by the year 2020, per second, about 51 TB of data will be created every year for every person (Marr, 2015). This may be truer for biology than for any other branch of science, as with the advent of Next Generation Sequencing, High Content Screening and other high throughput technologies, life scientists are producing more and more data.

High Content Screening (HCS) is a technology that combines automated fluorescence microscopy and image analysis to measure phenotypic response of cells to bioactive molecules. Using image analysis, changes in cell morphology are detected. Because multiple features are measured at the same time, this technique can be used for complex tasks such as drug candidate target prediction. It has recently been documented, however, that most HCS experiments do not exploit their full potential, as 60-80% of all HCS screens only use one or two measured variables, even though over ten variables per fluorescent dye can be measured (Singh et al. 2014). The fact that the larger part of the data generated using high throughput technologies (HTS) is being produced in an automated fashion, makes the analysis of the data even more important. Usually many features are recorded, leaving the researchers with highly dimensional data. The data produced in these experiments is often highly heterogeneous, which adds to their complexity. Data mining of high throughput data is therefore much more difficult because prior knowledge is required to understand the patterns in the data (Marx, 2013). There is a shift in the analysis from classical statistics to Machine Learning (ML) in high throughput data, because effectively analyzing all available data though directed statistical analysis is undoable (Greene, 2014). However, the advanced data mining knowledge required to analyze these large amounts of data, is often lacking in the toolkit of most life scientists (Omta, 2016). This pushes the analysis of experimental results away from the life scientist and into the domain of the data scientist, who often lacks the expertise of the life scientist. Therefore, a solution should be sought that allows life scientists to analyze their own experiments. HC StratoMineR, a tool for the analysis of true multivariate high content data was recently published. It is a web-based tool for the analysis of HCS data (Omta, 2016) which allows the user to make use of the full potential of HC data. Within the workflow of HC StratoMineR, consisting of several steps including data preprocessing, data reduction and clustering, there are various opportunities for the user to visualize their data. The visualizations are carried out by R, using the ggplot2 library, generating non-interactive visualizations (Wickham, 2010).

Data visualization

When data is being presented in a textual or tabular form, the amount of data that can be interpreted by a person is limited to a couple of hundred records. Beyond that, some way of visualizing the data is required to understand what information is hidden in the data (Tufte, 1985). Visualizing data is an immensely powerful way to show activity or artefacts within data. The visualization of the data is not only important to achieve clear insight into what the data looks like in its raw state but can also trigger new discoveries and insights (Friendly, 2008; Menger, 2016). For example, to summarize data using standard deviations, means, medians and ranges. As interesting as these summary statistics are, they still do not tell the researcher if the distribution of the data is normal, or if there are any outliers in the data. In contrast, a visualization will provide the researcher with an answer to these questions (Shneiderman, 2001). Visualizations can thus give the user a better understanding of how the data is composed and provide a clear overview of its distribution.

Card et al. (1999) subdivides visualizations into two categories, scientific visualizations, and information visualizations. Scientific visualizations, or “the use of computer-supported (interactive), visual representations of scientific data, typically physically based, to amplify cognition” (Card, 1999) is used in confirmatory data analysis. When there is a clear hypothesis that is to be accepted or rejected, scientific visualizations of data can be used to provide the scientist with the insight required to do so.
Information visualizations, or “the use of computer-supported (interactive), visual representations of abstract data to amplify cognition” (Card, 1999), is a powerful method in exploratory data analysis because it leverages the immense capabilities of the human visual system. Humans can detect and recognize features in a visualization extremely fast (<200 ms), even when analyzing over a million items (Healey, 1995). When providing the human cognitive system with an external aid in the form of a visualization, it allows for the creation of new hypotheses.

However powerful, there are certain limitations to the capability of the human visual system. There is a physical limitation to a maximum of three axes in a visualization. Also, there is a limit to the number of preattentive features, such as hue, orientation, intensity, and size that can be combined freely (Healey, 1995). If three or more of such features are used in the same visualization, they can greatly reduce the comprehensibility of that visualization. Furthermore, when dealing with large sets of data, misinterpretation, disorientation, and occlusion of parts of the data is prone to happen (Shneiderman, 2001). Therefore, to optimally use the potential that information visualization has to offer, methods are needed to communicate the main trends of the data effectively. For instance, visualizing a random subset of the data to increase the readability of the distribution and the variance. Simplified visualizations such as a boxplot can reduce the number of elements to five when visualizing a vector, where a scatterplot can contain thousands of elements (dots) to visualize the same vector. These simplified visualization methods can support the construction of visualizations that can be interpreted efficiently by the user, even when dealing with large amounts of data.

Visual Data Mining

In data mining, statistical methods and algorithms, Naïve Bayes, Principal Component Analysis (PCA), square root transformation, normalization and other methods are used to analyze large sets of data. Despite the effectiveness of these techniques, their complex nature makes the data mining process difficult to comprehend for non-data scientists. Because specific skills are required to properly configure these algorithms and interpret their results, the amount of control a researcher has over the analysis is diminished (Costello, 2005).

It is important to include the flexibility and creativity of the human mind in the data exploration process to make use of the cognitive capacity of the human brain. Visual data mining (VDM) focuses on integrating the user directly into the data exploration process, by presenting the data in some visual form (Keim, 2002). Including information visualization into the data exploration process, enables users to explore large volumes of data without having to understand complex statistical or mathematical methods and algorithms. VDM can still be used with noisy and heterogeneous data through the direct involvement of the user. These aspects of VDM allow for faster data exploration, and it regularly produces better results than automated methods (Keim, 2002). For example, a visualization may reveal distinct clusters in a data set, while the automated analysis of the same data set may not be able to detect these clusters, due to the noisiness of the data. In addition to the detection of clusters, VDM is useful for many other data analysis tasks, such as: outlier detection, feature importance assessment and the detection of patterns (Shneiderman, 2001).

Even though VDM can be performed without the use of data mining algorithms, the combination of both VDM and data mining allows for an even better data analysis solution. For instance, data mining can be used to provide visualizations with the simplification needed to make them more comprehensible, e.g. reducing the number of dimensions in the data to be visualized by factor analysis. Also, VDM is useful in exploring the (intermediate) results of data mining methods, or to make the process of a method clearer (Shneiderman, 2001) e.g. a layer by layer visualization of k-means clustering. Usually, data analysis is performed by a workflow consisting of multiple data mining methods. VDM can be used to explore the results of the entire workflow, or to inspect the (relative) effect of a single method in the workflow (Shneiderman, 2001; Fayyad et al., 1996).

Interactive Visualizations

VDM follows the paradigm of: Overview first, zoom & filter, and then details on demand (Keim, 2002). This makes the interactivity of the visualization an important aspect of VDM because it allows users to directly interact with the visualization. A theory on Direct Manipulation (DM) was described by Shneiderman (1982) within the context of computer applications and graphical user interfaces (GUI). DM endeavors towards an interface such that the object manipulated by the interface is part of the interface itself (Shneiderman, 1982). Typical examples of DM are zooming in on a picture using your fingertips, swiping on a tablet to the next page of a document, or uploading a file using drag and drop functionality by selecting and dragging the file to an upload form. Even though DM was devised in the early 80's, it remains one of the standard doctrines in interface design (Javed, 2011). According to Shneiderman (1982), a system which implements DM has the following principles:

1. Continuous representation of the object of interest (OoI).
2. Physical actions on the OoI or labeled button presses instead of complex syntax.
3. Rapid, incremental, and reversible operations whose impact on the OoI is immediately visible.
4. A learning approach that permits usage with minimal knowledge.

These principles make interacting with systems implementing DM more predictable and intuitive and make them easy to learn and use.

A good way to combine information visualization with automated statistics and data mining is to provide users with an application that implements direct manipulation. Users can explore their data by direct interaction with the visualization as the OoI. When interesting patterns are detected, data mining can be used to further investigate the phenomenon. Because the visualization itself is used as an interface and is constantly being displayed, users can optimally interact with their data (Shneiderman, 2001).

By equipping a visualization with direct manipulation, a user can directly manipulate the visualization, aiding in effective data exploration by focusing on interesting sections. Keim (2002) divides the methods of interactivity of visualization into five categories:

- **Dynamic Projection:** Through dynamic projection of a multidimensional data set, multiple dimensions can be viewed in one visualization. This method allows users to dynamically change the aesthetics of a visualization, such as the ability to change the variables that are on the axis of a visualization
- **Filtering:** Filtering the data to be visualized can also be extremely helpful in data exploration, by dividing the data into interesting subsets. This

can be done by querying a data set for interesting records, or by browsing the data through different subsets. The visualization will be dynamically updated according the filtered data

- **Zooming:** By zooming, very large amounts of data can be condensed for an overview of the data, while interesting parts can be magnified for a more detailed inspection
- **Distortion:** Interactive distortion techniques may be used to focus on a specific section of the data while preserving the overview of the complete data set, e.g. fisheye view. An example of interactive distortion is the blurring of records that are below a certain threshold, showing only the records that are above that record. An example of distortion in a heatmap is shown in the Supplementary Data S1 A & B. In this example, interactive distortion techniques are used to 'find' the top 10% of the data
- **Linking and brushing:** Brushing over a visualization allows for the highlighting of sections, aiding in finding interesting patterns in the data. Linking visualizations to other visualizations may also provide the user with new insights. For example, linking a scatter plot to a histogram (see supplementary data, S1 C & D) combines two different outlooks over the data in one view (Keim, 2002). By brushing over the histogram, the data on the scatter graph is updated

Application of Interactive Methods

There are numerous data cleansing, preparation and manipulation methods (algorithms) available that can be applied to (a part of) the data (Chapman, et al., 2000; Fayyad et al., 1996). Examples are normalization, transformation and scaling algorithms but also clustering and classification algorithms. Some of them are very general while others are designed for very specific purposes e.g. the B-score normalization method, a row and column polish for the correction of plate effects in the analysis of HCS (Birmingham, 2009). Most of these algorithms lack a description in terms of heuristics or best practices. Moreover, the best practices that are available are not always fitting for each situation (Osborne & Costello, 2009). Consequently, it is not always clear which of these methods yields the best results. As previously mentioned, a visualization of the results can provide the user with information about the effect of an algorithm on the data.

In a visualization, the interactive categories of Keim; filtering, zooming, sorting & distorting of the data or parts thereof can reveal patterns in the data that are not easily visible without using other methods. When using non-interactive data visualization, the application of these methods is not possible. In information processing, such as VDM, both speed and accuracy, play an important role (Wickelgren, 1977). In the context we investigate, we require a high accuracy degree and we expect a main difference in speed. In other words, we primarily expect a difference in speed.

Hypothesis 1a: Patterns in the data can be detected faster using interactive data visualizations over non-interactive visualizations.

Hypothesis 1b: Patterns in the data can be detected more accurately using interactive data visualizations over non-interactive visualizations.

Hypothesis 2a: A faster interpretation can be made about the output of different algorithms using interactive visualizations over non-interactive visualizations.

Hypothesis 2b: A more accurate interpretation can be made about the output of different algorithms using interactive visualizations over non-interactive visualizations.

Materials and Methods

Because we are seeking for a way to measure the effects of multiple data manipulation methods (algorithms) and simultaneously provide them with the insight of interactive data visualization, we developed a questionnaire which includes the ability to create interactive visualizations and non-interactive visualizations. Each question is supported with one or more visualizations. To test our hypotheses, we are conducting the questionnaire.

Setup of the Experiment

The questionnaire contains 12 questions (Supplementary Data S2) with two additional test questions. The test questions are used to train the participant in the look and feel of the interface and the format of the questionnaire. The questionnaire is carried out with a control and an experimental group. In the control group, non-interactive visualizations are shown, and interactive visualizations are shown in the experimental group. The same questions and visualizations are presented to both groups. See the visualizations offered with the questions in Figure 1. The questions are offered in a random order to take the user's attention and learning curve into account. This is to avoid the possibility that certain questions are always first or last, which might affect the representativeness of the given answers. The two measured constructs are accuracy and the time required to answer the questions. For each question, the time is measured independently, and the time measured starts when the visualization is fully loaded; so, the loading time and the speed of each user's computer do not affect the time to answer the question. A classroom was prepared for participation. There were 33 students instructed to bring a laptop with an external mouse and headphones to watch the instruction video carefully. They were asked to fill in the questionnaire as accurately and as quickly as possible, with a reward of €50 for the fastest student with the least number of wrong answers. The focus is concentrated on the accuracy over speed to avoid participants clicking through the questionnaire as quickly as possible and have a few correct answers by chance. The participants were instructed not to ask any questions during the questionnaire. In total, 79 students, including computer science, information science, and bioinformatics students, from Leiden and Utrecht participated, whereof 46 people with various backgrounds participated over the Web.

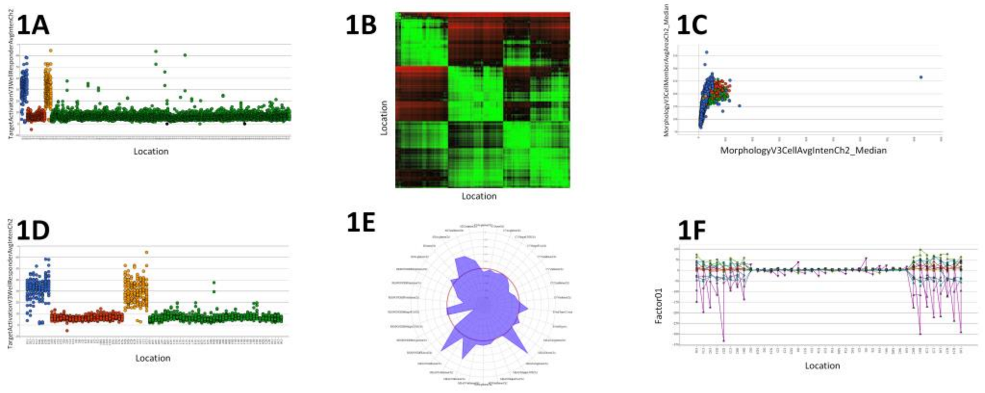


Figure. 1. *Visualizations of the questionnaire*

(A) A scatter plot with the well locations (discrete variable) on the x-axis and the variable TargetActivationV3WellRESPONDERAvgIntenCh2 on the y-axis. Question 1: What is the number of records where TargetActivationV3WellRESPONDERAvgIntenCh2 is 50 or higher? (B) A correlation matrix showing the similarity of the well locations on the x-axis and y-axis in green. Question 2: See the correlation matrix below. Select the pair that correlates between 0.999999 (99.999%) and 1 (100%). (C) A scatter plot showing MorphologyV3CellAvgIntenCh2_MEDIAN on the x-axis and MorphologyV3CellAvgAreaCh2_MEDIAN on the y-axis. Question 3: In the scatter plot below, MorphologyV3CellAvgIntenCh2_MEDIAN(X-axis) contains an outlier of 825. What is the plateName of this outlier? (D) A scatter plot showing the well location (discrete variable) on the x-axis and TargetActivationV3WELLRESPONDERAvgIntenCh2_MEDIAN on the yaxis. Question 4: What happens to the variance of the data when it is log2 transformed? (E) A polar plot showing the variables at the x-axis and the factor loading on the radius in a range from -1 to 1. Question 5: Select the variable that shows a loading of 0.82 on Factor01. (F) A line plot showing the well locations on the x-axis (discrete variable) and Factor 01 on the y-axis. Question 6: In the line plot below, 12 lines are shown. Each line represents a different microplate. Select the microplate that is most similar to microplate 2.

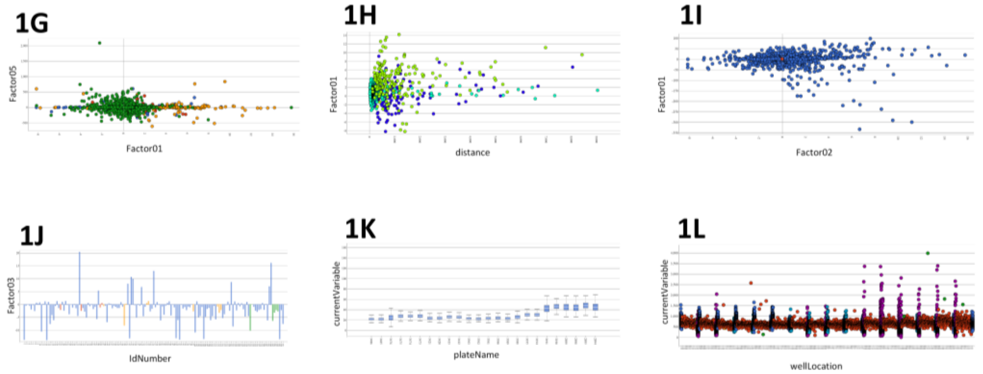


Figure 1. (Continued)

(G) A scatter plot showing Factor01 on the x-axis and Factor05 on the y-axis. Question 7: In the data set provided, there are two variables that result in a cross using a scatter plot. What is the combination of variables that produces a cross? (H) A scatter plot showing a Euclidean distance score on the x-axis and Factor01 on the yaxis. Question 8: In this data set, there is a record that contains a distance value of 9039.15 and a Factor01 value of 1.33. On what wellLocation is this record located? (I) A scatter plot showing Factor02 on the x-axis and Factor01 on the y-axis. Question 9: The scatter plot below contains one red dot. Select the Factor01 value of the record closest to the red dot. (J) A bar plot showing an id number of the x-axis and Factor03 on the yaxis. Question 10: Select the IdNumber of the record with the lowest value for Factor03. (K) A box plot showing the plate name on the x-axis and a variable named currentVariable on the y-axis. Question 11: Select the two normalization methods that align the medians of the box plots on the Y-axis. (L) A scatter plot showing the well locations on the x-axis and a variable called currentVariable on the y-axis. Question 12: Select the group that contains the record with the highest value.

Materials

The questionnaire was conducted in a web-based manner. The interactive and non-interactive versions were performed similarly. The participants were asked to visit a web address to participate in the questionnaire (see Supplementary data S3). Participants were randomly assigned to the experimental or control group. The implementation of the questionnaire consisted of two parts, the frontend, and the backend. The backend of the questionnaire was built using the Python Django framework, that used MySQL for data storage. The Django framework was responsible for the random allocation of participants to a group, but also for the random ordering of questions in the questionnaire. In order to communicate with the frontend of the questionnaire, Django exposed a RESTful api. The front end of the questionnaire was built using the Angular framework, that was responsible for the rendering of the interface. The Angular framework had an asynchronous connection with the Django backend via observables. The (interactive) visualizations are all generated using D3.js, except for the 3D scatter graphs, that is constructed by the Vis.js library. The answering time was measured by the frontend of the application. The answering time was measured from the moment that visualizations were loaded, so loading time of the visualization does not affect the answering time. However, in the non-interactive version, the switching of visualizations is artificially delayed, to simulate the time it takes to render a non-interactive visualization in ggplot2. Both the interactive as the non-interactive group is built using the same engine and framework to avoid other external factors of influence.

Data Analysis Methods

The data set contains 79 records. Each record contains metadata about the participant's age, gender, and educational level. Also, computer literacy, Excel, data mining and English proficiency are measured in a Likert Scale from 1 - 5, in order to discover any relations to these proficiencies or demographics properties. To measure the construct accuracy, the total number of incorrect answers per participant were measured. To measure the construct time, the sum of the time for each individual question per participant was measured. To measure the construct time to compare different algorithms, the total time of the questions related to the construct were measured (Table 1). To measure the construct accuracy to compare different algorithms, the total number of incorrect answers of the questions related to the construct were measured (Table 1).

Table 1. *The Statistical Summary of Time Comparing Methods, Total Time and Wrong Answers in Both Conditions*

Metric	Condition	N	Mean	SD	SE	P-value one-tailed
Total Time	Interactive	39	867.01	372.79	59.69	<0.001
	non-interactive	29	1338.58	662.91	123.10	
Total wrong answers	Interactive	39	2.38	2.68	0.43	<0.001
	non-interactive	29	4.41	2.24	0.42	
Time comparing algorithms (s)	Interactive	39	138.56	75.86	12.15	<0.05
	non-interactive	29	176.57	90.95	16.89	
Wrong answers comparing algorithms	Interactive	39	0.67	0.701	0.148	>0.05
	non-interactive	29	0.93	0.799	0.112	

Students who did not finish the questionnaire completely were left out, which results in a data set of 68 students. To measure a difference in time and accuracy between the control and experimental group, a one-tailed independent sample t-test is used. To measure a difference in the accuracy of detection of patterns in the data between the control and experimental group, a chi-square test was used.

Due to the complexity of the questionnaire and the fact that participants would be using the Web-based environment for the first time, introductory material was provided to the participants in advance. Each group was provided with their own relevant introductory material. The introductory material consists of two test questions, each one accommodated by an explanatory video. Also, the questionnaire was provided with question-specific textual information and information to use the interface.

Results

Table 1 shows the results of the questionnaire performed in this study. The first result is the time in seconds (total time) to complete the questionnaire. On average, it took students more time to complete the questionnaire in the noninteractive group (M= 1338.58, SD = 662.91) than the interactive group (M= 867.01, SD = 372.79); t(66) = 3.726, $P < 0.001$ (figure 2A). A second result is the number of wrong answers given in the questionnaire (Total Wrong Answers).

The questionnaire in total contains 12 questions (excluding test questions). On average, students had more wrong answers in the noninteractive group (M= 4.41, SD = 2.24) than the interactive group (M= 2.38, SD = 2.68); t(66) = 3.303, $P < 0.001$ (figure 2B). A third result is the time in seconds that was required to compare multiple algorithms (time comparing algorithms). On average, it took students more time to compare algorithms in the noninteractive group (M= 176.57, SD = 90.95) than the interactive group (M= 138.56, SD = 75.86); t(66) = 1.877, $P < 0.05$ (figure 2C). The result is the number of wrong answers given in the questionnaire related to the comparison of algorithms. There was no significant association between the visualization group and the accuracy of the comparison of algorithms, $P > 0.05$ (figure 2D).

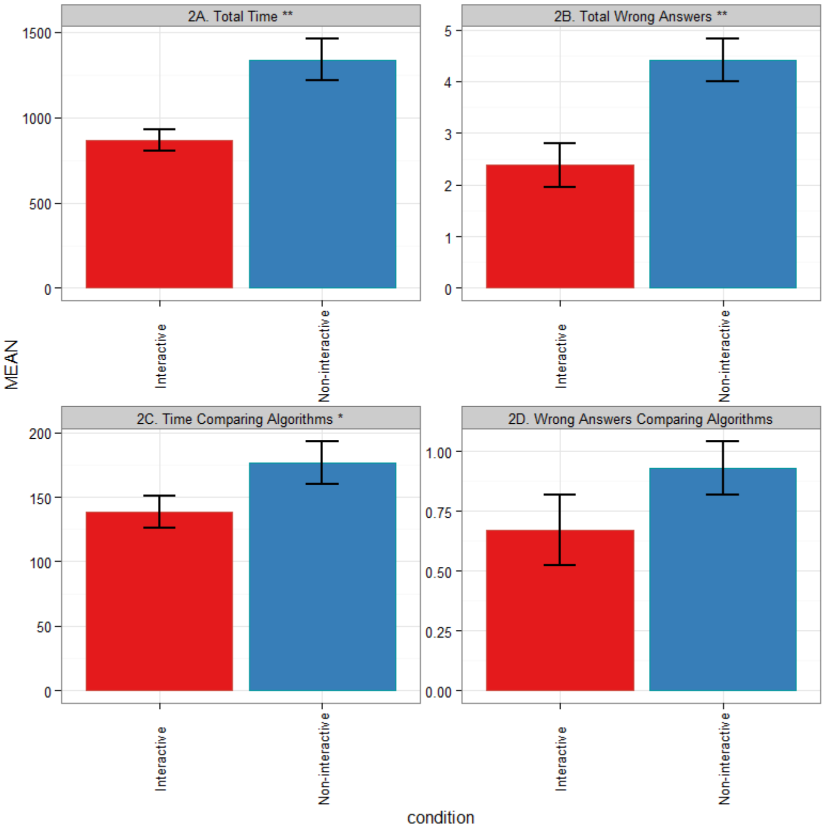


Figure 2. *Bar charts, including error bars, showing differences between interactive and noninteractive visualizations*

*(A) Total time indicates the time to complete the questionnaire, excluding the loading time of the visualizations (significant difference, one-tailed P < 0.001**). (B) Total wrong answers indicate the number of wrong answers of the questionnaire (significant difference, one-tailed P < 0.001**). (C) Time comparing algorithms indicates the time to compare algorithms in seconds (significant difference, one-tailed P < 0.05*). (D) Wrong answers comparing algorithms (result not significant).*

The results contain relatively high standard deviations because some of the students were done extremely fast, for example, 91 s, while others took over 59min to complete the questionnaire.

Learning Effect

The questions built in the questionnaire were presented in a randomized manner to take the user's attention and learning curve into account. This is to avoid that certain questions are always first or last, which might affect the representativeness of the given answers. Although there is a difference in time

one question requires to be answered, there is still a learning curve present that can be explained by the fact that the interface is completely new and possibly a new way to visualize data. We took the questions in the real order and took the median of time across the two conditions: interactive and noninteractive.
Figure 3 shows a curve for both conditions, summarizing the median of time required to answer the first until the last question in the order the questions were offered to the participants, hence allowing to study this learning effect directly. For example, it is expected that the time to answer the questions decreases after the first few questions. As can be seen in Figure 3, we clearly see that there is an initial peak in answering times for the first few questions, after which it seems to decrease to some plateau. We see that, despite the graph being quite noisy due to the smaller sample size and outliers, the curves both demonstrate a reduction in answering time throughout the test and indeed become more stable after about four items. Although the gap reduces, the interactive condition seems to remain somewhat faster. The last questions seem somewhat slower, which might be an indication of testing tiredness of the group of students.

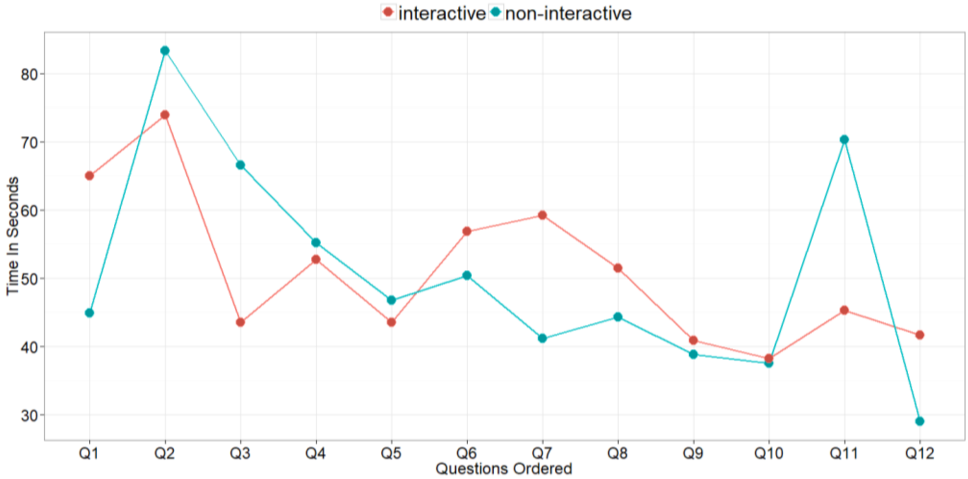


Figure 3. *The learning curve of the interactive and noninteractive versions by time taken per question The x-axis shows the question in the order they were shown to the participants. The y-axis shows the median of the time of the ith question. We regard only the ordering of questions. The first question clearly takes more time, later questions less time.*

Expertise

In the questionnaire, data mining expertise was measured in a Likert scale, by asking participants to rate themselves from 1 to 5 in data mining expertise, as shown in Figure 4. To show if there is a relationship with time/accuracy, we calculated Spearman correlations. Data mining expertise against the total time that was required to carry out the questionnaire shows a Spearman's rho coefficient of r=-0.082, NS. Data mining expertise against the total number of wrong answers shows a Spearman's rho coefficient of r = -0.293, NS. Hence, no significant monotone relationship was found between self-reported data mining expertise and response time, nor between data mining expertise and accuracy.

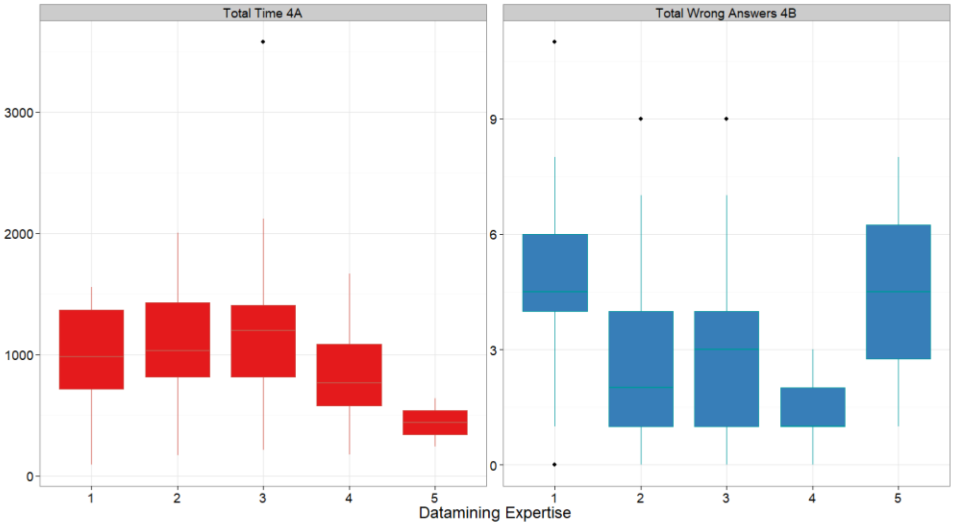


Figure 4. *Box plots showing data mining expertise in relationship to total time and total wrong answers (A) The x-axis shows the data mining Likert scale from 1 to 5. The y-axis shows the box plot of the total time in seconds required to complete the questionnaire. (B) The x-axis shows the data mining Likert scale from 1 to 5. The y-axis of the box plots demonstrates the total wrong answers.*

Discussion

Visualizations with interactive methods are shown to provide better comprehensibility than non-interactive data visualizations. The overall time to perform 12 assignments was significantly decreased in the interactive group. The number of wrong answers given by the interactive group also showed a significantly lower number. The time that was required for comparing the output of algorithms was significantly decreased in the interactive group, although the number of wrong answers given by the interactive group to compare the output of algorithms did not show a significant result. Because we stressed the importance of accuracy over time toward the participants, we influenced their speed and accuracy trade-off described by Wickelgren (1977) and hence mainly expected a reduction in time for the interactive condition. However, when using the right visualization tools, it was demonstrated that both accuracy and speed can be improved at the same time. From the 79 participants, 46 participants who performed the questionnaire over the Web did not have an affinity with data analytics necessarily. The students who conducted the questionnaire in a classroom were students in either bioinformatics, information science, or computer science. So, one might expect some diversity in knowledge of data analytics among the participants. We noticed, during the experimental setup, that random people joined and tried to fulfil the questionnaire. Most of those people gave us feedback in that they had absolutely no idea what they were doing and were quitting the questionnaire after one or two questions. To keep our results relevant and to reduce noise, we tried to include participants in this study through a classroom session that do have more knowledge about fields related to data analytics but have less domain knowledge as life scientists have.

The questionnaire contained 12 questions, including two test questions, to get familiar with the visualization platform. The questions that we designed cover all five categories of Keim (2002) and are questions relevant to the analysis of HCS data, for example, "what is the number of data points" or "what is the plate name of this outlier." The 12 questions also include two questions covering the comparison of the output of algorithms. When we would have implemented questions that were always immediately clear, one would not perform better using interactive visualizations and there would not be a true incentive to use interactive visualizations. We believe that questions or challenges regarding data analysis can be efficiently supported by the right visualizations. Interactive visualizations can add extra value by speeding up the analysis process because of its flexible nature and decrease the user's cognitive load because of the lower burden of recalling visualization objects.

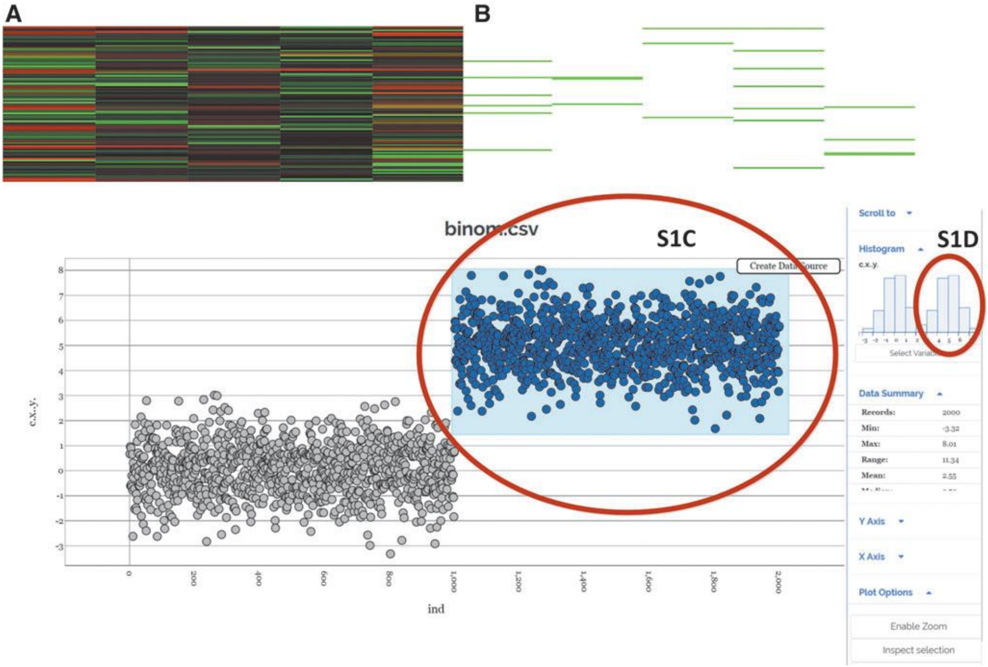
There are certain problems associated with data visualization in general. Visualizations have their limitations when dealing with large data sets, since occlusion of (parts of) the data, disorientation, and misinterpretation can occur (Shneiderman 2001). The visualization of large data sets can also lead to the problem of overplotting, producing a visualization in where individual data points are coerced into a single solid object.Minor differences between data points are not observable in these situations, and only the trend (e.g., linearity) of the data can be derived from these visualizations. With interactive data visualization, it is still possible to view these minor relationships between data points, as a result of a zoom event. The visualization of large data sets also leads to the problem that visualizations take a long time to render. Through the survey, we found an expected result that the visualization of a random sample of a large data set is informative enough to observe the trend (e.g., the distribution) of the complete data set. With a sample, the rendering time of the visualization can be reduced, improving the responsiveness of the visualization process. The cognitive load can thus be reduced when the waiting time is decreased between the inspection of different data visualizations. This leads to the next problem in data visualization: the comparison of the output of algorithms. Usually when a researcher wants to visually compare the output of multiple algorithms, there is a delay between different visualizations that represent the output of different algorithms. The data need to be manipulated by another algorithm and a new visualization needs to be created for each comparison. In this project, a platform was designed that optimizes the comparison of visualizations. Because the delay between viewing different visualizations is minimized, different visualizations can be compared faster than using regular visualization platforms. Keim (2002) reported five categories of interactivity. We propose the addition of a sixth category: the interactive comparison of the output of algorithms (data manipulations). A possible side effect of this interactive method is the bias that may be introduced as researchers will set out to “find” the data algorithm that makes their data look best. At the same time, we also stress that the proposed sixth category could be part of the other methods described by Keim. For example, when linkage is used, two visualizations are interactively connected; thus, they can be compared. A side note here is that there should be a possibility to compare them side by side instead of swiping through the various visualizations that one would compare.

Author disclosure statement

WAO and DAE are both co-founders of Core Life Analytics B.V.

Supplementary Data

Supplementary Data S1



Records:	2000
Min:	-3.32
Max:	8.01
Range:	11.34
Mean:	2.55

Supplementary Figure S1. *Distortion*

A & B Distortion is visualized in a heatmap where the top 10% is visualized, the rest of the data set (not belonging to the top 10%) is filtered out while the position of the items in the top 10% is preserved.

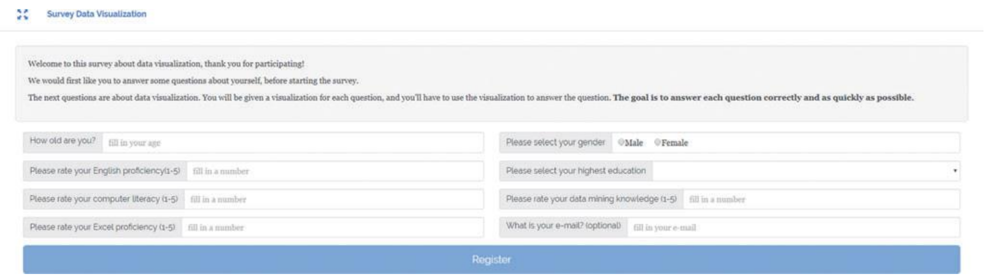
C. highlights a cluster in the data containing two peaks using a scatterplot. The data that is brushed is highlighted with a squared box while the unselected data has a lower opacity and is gray.

D. Corresponds to the data visualized in C and demonstrates two peaks (as in C). The visualizations C & D are linked and the highlighted data in C is also highlighted in D and combines a scatterplot and a histogram in one view

Supplementary S2

1. What is the number of records where TargetActivationV3WellRESPONDERAvgIntenCh2 is 50 or higher?
2. See the correlation matrix below. Select the pair that correlates between 0.999999 (99.999%) and 1 (100%).
3. In the scatter plot below, MorphologyV3CellAvgIntenCh2_MEDIAN(X-axis) contains an outlier of 825. What is the plateName of this outlier?
4. What happens to the variance of data when it is log2 transformed?
5. Select the variable that shows a loading of 0.82 on Factor01.
6. In the line plot below, 12 lines are shown. Each line represents a different microplate. Select the microplate that is most similar to microplate 2.
7. In the data set provided, there are two variables that result in a cross using a scatter plot. What is the combination of variables that produces a cross?
8. In this data set, there is a record that contains a distance value of 9039.15 and a Factor01 value of 1.33. On what wellLocation is this record located?
9. The scatter plot below contains one red dot. Select the Factor01 value of the record closest to the red dot.
10. Select the IdNumber of the record with the lowest value for Factor03.
11. Select the two normalization methods that align the medians of the box plots on the Y-axis.
12. Select the group that contains the record with the highest value.

Supplementary S3



Survey Data Visualization

Welcome to this survey about data visualization, thank you for participating!

We would first like you to answer some questions about yourself, before starting the survey.

The next questions are about data visualization. You will be given a visualization for each question, and you'll have to use the visualization to answer the question. **The goal is to answer each question correctly and as quickly as possible.**

How old are you?

Please select your gender ○ Male ○ Female

Please rate your English proficiency(1-5)

Please select your highest education

Please rate your computer literacy (1-5)

Please rate your data mining knowledge (1-5)

Please rate your Excel proficiency (1-5)

What is your e-mail? (optional)

Supplementary Figure S3. *A screenshot of the registration form of the questionnaire*

Supplementary Data S4

Each Question of the Questionnaire and Its Interactivity Category

Question	Associated Category of Interactivity
1	Brushing
2	Distortion
3	Sorting
4	Comparison of algorithms
5	Distortion
6	Filtering
7	Dynamic projection
8	Dynamic projection
9	Zooming
10	Sorting
11	Comparison of algorithms
12	Filtering

Chapter 6 - PurifyR: An R Package for Highly Automated Reproducible Variable Extraction and Standardization

Motivation: Life sciences experiments that employ automated technologies, such as high content screens (HCS), frequently produce large data sets that require substantial amounts of preprocessing before analysis can be carried out. The standardization of this preprocessing becomes impossible as the data set size increases, if there are manual steps involved. Virtually no standards for preprocessing currently exist and few user-friendly tools are available which allow the cleaning of data files in a simple and transparent manner, while also allowing for reproducibility.

Results: We demonstrate in a publicly available R package, PurifyR, how preprocessing steps can be streamlined and automated. PuriyR supports multithreading and the standardization of large matrix preprocessing. These steps provide transparent and reproducible preprocessing for matrix-oriented data sets.

Adapted From:

Omta, W. A., van Heesbeen, R. G., Shen, I., Feelders, A.J., Brinkhuis, M.J.S., Egan, D.A., Spruit, M.R. (2020). PurifyR: An R Package for Highly Automated Reproducible Variable Extraction and Standardization. *Systems Medicine* 3(1), Integrative Data Analysis in Systems Medicine, 1–7.

1 Introduction

Machine generated data sets, such as those from HCS, are increasingly unavoidable in life sciences and bioinformatics experiments (Macarron et al., 2011; Sundberg 2000) and given their size and complexity, are challenging to both maintain and process without automated preprocessing toolkits (Bergström & Ivarsson 2015). Big data analysis using machine generated (sensor, image and robotics) data sets involves ever increasing time to clean, maintain and summarize (Billingsley et al., 2018). Time spent cleaning data sets continues to increase, consuming valuable time which could be better used for interpretation in later analyses (Bergström & Ivarsson 2015). Looking forward, manual approaches to big data analysis are unsustainable, given how big data and the number of separate tools continue to increase in size and complexity (Bajcsy et al., 2015). Currently, thousands of variables and millions of observations are generated for every high content screening experiment, requiring big data analysis approaches (Sullivan et al., 2015). Continuous improvements to sensor software and broad advancements in hardware provide increased opportunities to capture big data in nearly every experiment (Banks et al., 2017), hinting that big data sets will become more common and orders of magnitude larger in the near future (Esner, Meyenhofer & Bickle, 2018). For example, data collected during high content experiments capture consistently higher resolutions, dimensions (3D) (Kriston-Vizi & Flotow, 2017) and detail as the technology continues to develop (Buchberger et al., 2017). Big data analyses require automation beyond that of pipelines and profiling tools (Kraus et al., 2017).

Cell screening experiments assisted by robotics, such as Luminex bead technology (Labuda et al., 1999) are assisted by robots for automating repetitive and tedious cell screening experimental tasks that generate large amounts of information at cell or even sub-cellular object level. Although groundbreaking, these experimental methods can have unintended consequences for researchers (Nickischer et al., 2018). Up to 80% of the analysis time of large experimental data sets can be consumed (Wickham, 2014) by repetitive and non-value adding tasks (Stodder, 2017) such as removing undesirable experiment-specific artifacts, identifying context-specific outliers, excluding systematic machine-biased errors from results, interpreting manufacturer-specific machine codes, correcting missing data and repairing inconsistent classification methods. Increases in the complexity and the amount of data creates ever increasing work for experimenters to aggregate experimental results and conclusions. Each observation and every value must be screened to validate methodological standards, to be useful for later statistical analyses, and ultimately publication (Li et al., 2016). Often, researchers could benefit from the ability to identify these types of data problems early in experimentation to validate the research procedures that are being followed and to ensure that data quality is sufficient for significant testing and publication (Hoffman et al., 2018). These types of problems go undetected for the duration of experimentation or during piloting stage which could easily have been addressed and avoided if detected earlier. Ideally, experimenters could check these statistics repeatedly during the research process, to identify issues and correct them before the experiment is complete and it is too late to address any collection issues. For instance, data sets must be clean for analysis and prepared before the use of any later machine learning methods (Omta et al., 2017). Time consuming cleansing processes require focus, which can become a distraction from the original purpose of the experiment. When these steps are not documented, the results can be nearly impossible to reproduce and can lead to experimental results which cannot be reused or compared with subsequent projects and findings (Thomas, 2010).

Many packages currently exist for preprocessing data sets (Bellomo et al., 20117) to meet the assumptions of statistical testing methods and machine learning models. However, few methods exist which can comprehensively automate the majority of preprocessing steps in both documentation and reproducibility in a manner sufficient to meet the necessary fundamental assumptions conceivable for most statistical methods such as outlier handling, missing data imputation and feature selection (Dinkla et al., 2017).

Researchers are therefore required to repeat these tedious and standardizable preprocessing steps manually in every case, which includes the modelling of data sets, building data pipelines and testing code. There is a need for robust and automated preprocessing frameworks for detecting inconsistencies in data and repairing or removing and reporting them in a transparent and autonomous manner. Furthermore, these repetitive steps can be difficult to log and can be impossible to replicate by future researchers. The exact steps followed can easily be forgotten or repeated in different sequences for various reasons, making the cleaning operations impossible to repeat during later research. Often, these issues create irreproducible data sets which leave later researchers confused and unable to verify values or update the results with recent findings.

There is a need to automate the preprocessing of research data sets. Previously published packages have demonstrated the ability to reduce the time necessary to automate preprocessing and produce generally usable data sets for analysis (Piccinini et al., 2017) while ensuring assumptions for various machine learning and statistical methods are met without user intervention or extensive programming expertise (Echeverri & Perrimon, 2006). When experimental data is processed using an automated and documented approach, results can be consistently reproduced in large workflows, and used to validate data sets during experimentation to prevent using bad data (records and features) and to compare results with later studies (Joslin et al., 2018). Preprocessing pipelines themselves can also be published and reused for later research and improve over time (Sommer et al., 2017). Furthermore, the automation of data preprocessing could result in a substantial reduction of the necessary effort required to make use of big data sets; creating a need to automate the preprocessing of big data (Bray & Carpenter, 2017).

2 PurifyR Package Components

Here, we present PurifyR, an R package for big data preprocessing, specifically for preparing high dimensional data sets in a dynamic, repeatable and autonomous manner in order to avoid reinventing the wheel. Experimental data sources such as automated cell screening data are primarily machine generated data sets with thousands of columns and millions of records. Often tens or hundreds of these matrices are generated during the course of experiments, requiring an almost impossible amount of work if aggregated and processed manually using spreadsheet software such as MS Excel, or statistical software such as SPSS (Li, 2017) The PurifyR package facilitates three preprocessing steps; ScanR, ScrubR and SmashR (figure 1). Writing preprocessing scripts are tedious, time-consuming, and error-prone (Wickham, 2014). Re-used scripts mainly contain hard-coded column names and references to values, lists and file names. These scripts cannot be reused in follow-up experiments and require manual rewriting for subsequent analysis and making inadvertent errors or mistranslations and lead to irreproducible results.

2.1 ScanR

ScanR generates common meta-data (Danovi et al., 2012) for each column such as uniqueness and percentage missingness, variance, outliers, and distribution type. The meta-data generated in this step has been built specifically to determine which columns are considered useful and in a 'healthy' state based on well-defined and analysis-specific assumptions of data quality (Bougen-Zhukov et al., 2017) Users are able to review the meta-data of the original data set if desired, before producing a cleaned data set.

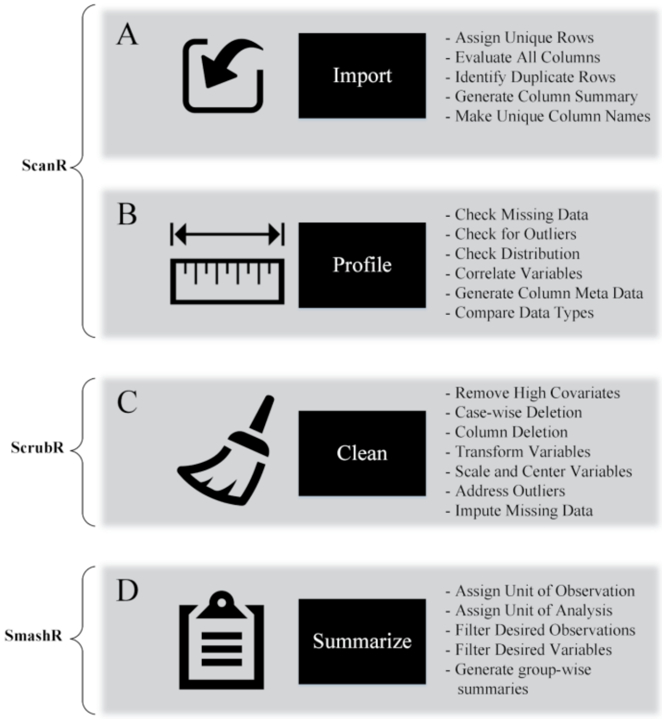


Figure 1. *The Workflow of PurifyR*

A. *Makes sure all variables have unique column names without special characters that can be handled in R*
B. *Carries out a data scan including missing data, distributions and data types*
C. *Allows for removal of outliers and handling transformation, scaling and missing data*
D. *Allows for the aggregation of data based on a defined unit of analysis*

ScanR requires a data source for input and is a data frame or data table. This data table is sampled based on the sampling percentage, to ensure enough records for summary statistics but not an excessive number, if the data table is very large. Each variable in the input data set is summarized for simple statistics such as mean, median and mode (see Supplementary Data A). This is completed in parallel, according to the number of processors available on the computer.

Variable names are cleaned of spaces and special characters and assigned a unique ID to ensure duplicate values are not confused and can be used in later steps. Variable meta-data is generated and provides the ability to compare each column with others by providing a simple list of example values found in each column as well as its highest correlated column and the maximum Pearson correlation coefficient. Variables are tested against rules to identify which contain unique information and certain types of data (categorical, continuous, etc). Each variable receives a test against other similar variables to ensure enough unique information exists to warrant inclusion in later machine learning applications.

Finally, a Mahalanobis distance and covariance matrix is calculated using the resulting healthy variables. If these functions fail, highly correlated variables are iteratively removed one by one until a final list of healthy predictors allows for the calculation of a non-singular matrix. The list provides information about why certain features are excluded e.g. because of high covariation. A final list of healthy predictors is created and assigned to the most clean and unique variables. This list of healthy variables can be passed onto the next step, scrubR.

2.2 ScrubR

The ScrubR step applies rules to each variable and then analyzes each included variable row-wise. It automatically produces appropriate and method-specific transformations, standardizations and imputes outliers and missing values given the results of the meta-data generated above. These configurations have default values and can be configured to meet the specific requirements and assumptions of later analysis methods (see Supplementary Data B) (Boutros, Heigwer & Laufer, 2015). Examples of subsequent analyses are PCA, linear regression, and neural networks (Kraus et al., 2017). Output data sets of the scanR function can be used without further manipulation for later analysis, feature engineering and prediction steps.

The scrubR function requires a data set and the list of healthy predictors calculated in the ScanR function. It will output a cleaned version of the original data. The user can select from a few options, including the row wise missing percent allowed in the final data set, the threshold standard deviation allowed for outliers, the proper transformation method for variables, the intended scaling method for variables and the desired imputation method for addressing missing data.

The function begins by removing records which exceed the missingness threshold i.e. the percentage of columns per record containing missing data. It then selects any outlying points which, based on the settings define an outlying datapoint. Variables which have failed the skewness tests during the ScanR function, will receive a recommended transformation, to adjust skewness towards normality. Each variable is then scaled and replaced with a z-value, or other desired scaling calculation. Missing values are finally case-wise deleted or imputed using a standardized approach, or using a package for imputation, such as MICE (Van Buuren, 2014). for random value replacement, multivariate imputation by regression, or random forest (figure 2).

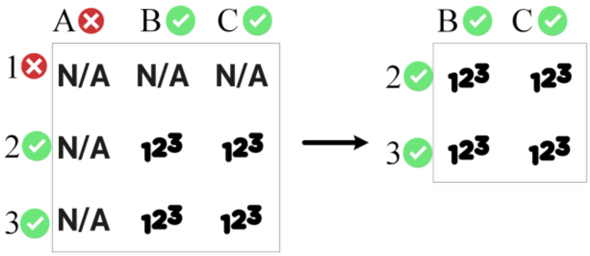


Figure 2. *Visual representation of the PurifyR input and output*
The input and out that is checked for columns that contain missing data i.e. column A or checking missing data on the records e.g. case wise percentage deletion, e.g. record #1.

2.3 SmashR

The SmashR function calculates estimators representing the unit of analysis. It requires a clean data set, such as the data set output by the previous step, ScrubR, as well as a list of healthy predictors, calculated by ScanR or provided manually. This can be used to easily represent the original identity of the data (figure 3). The Unit of Analysis input can be provided by one or more variables in the data, usually a categorical variable. The SmashR function aggregates and groups the originally observed data by the analysis variable. Any missing or N/A values within the unit of analysis are excluded from the analysis. For every value of the unit of analysis variable, a list of summary statistics is generated for every variable in the data set, such as mean, median, maximum, and minimum value. This summarized data is extremely useful for analyzing the original data set and creating visualizations given the summary calculations have been pre-prepared and are quickly available to slice and compare the data set. This step is useful for interpretation and comparison and exploration of the data at a high level.

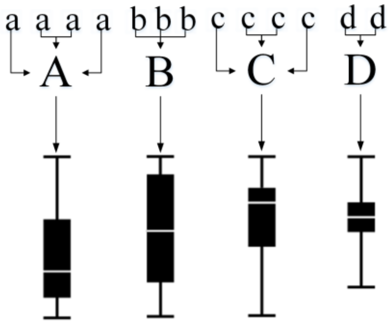


Figure 3. *This figure visually represents the function of SmashR*
The lower cases represent the unit of observation which is input for the calculation of the summary statistics representing the unit of analysis, the capital cases in this figure. Let the small letters be measured cases and the capital cases a set of estimators e.g. a minimum, Q1, median, Q3 and maximum estimator for input for visualization, interpretation and understanding the data.

3 Results

Rule-based preprocessing frameworks, such as PurifyR, as well as additional scripts can be assembled to standardize and automate a great deal of work normally left for bioinformaticians which is also very error-prone and time consuming to understand. Once automated, bioinformaticians can focus on more substantial work such as interpreting the semantics of the data, improving used methods and interpreting the outcome. A suitable preprocessing layer can be reused, and data sets reprocessed repeatedly in a consistent manner. Bioinformaticians can create a data preprocessing pipeline to first begin exploring data without a great deal of exploratory effort initially in removing data which can be described and completed by a rule-based standardized tool. PurifyR focuses on providing this standardized and reproducible context for preprocessing workflows.

The PurifyR package can be installed from Github, (see Supplementary Data F) or a live Shiny implementation can be seen at https://purifyr.stratominer.com/Shiny. Users can call three specific functions to automate preprocessing steps. Predefined configuration values are prepared to ensure data sets meet the assumptions of the following machine learning methods, for use by other packages, PCA, regression, and others. Input data is an existing R data frame or data.table object, from a file or other source. The package can be used to profile data and perform column health checks to recommend only useful features for the use of downstream analysis steps e.g.

machine learning. Second, the package scrubs the healthy columns, from the previous step and performs row-specific processing to ensure only high-quality records are included and missing or out of range values are repaired. Finally, data are transformed and scaled to meet the requirements of matrix-based methods, such as PCA. Finally, the package performs post-analysis profiling to display per-column statistics such as intra-variable variance and correlation metrics, useful for evaluation prior to performing additional machine learning steps.

We tested this on five public data sets, (see Table 1. And Supplementary Data D) and three HCS data sets (see Supplementary Data C) (van Heesbeen, et al., 2016). The data set with over 400K records and >200 features completes in ~0.5 minutes generated on AWS EC2 R5 instance with 4 cores and 32GB RAM. PurifyR operates completely using the data.table package for optimized computation and minimized space required in memory and is using the package parallel for multi-core usage. The data.table approach requires significant additional development effort but demonstrates huge performance improvements, as is shown in Figure 4.

Table 1. *This table represents the results of ScanR*
The results for the following data sets; mtcars, iris, diamonds (ggplot2 package), baseball (plyr package) and flights (nycflights13 package). Also, three HCS data sets were used, one of them available through the supplementary data. The column Data set describes the data set, the column Rows describes the number of rows of the data set and the column Columns describes the number of columns in the data set. The column Calculation Time describes the amount of time required to process the data set with ScanR in PurifyR. Results are generated using an AWS R5 xlarge EC2 instance (4 cores 32 GB RAM).

Data set	Rows	Columns	Calculation Time
mtcars	32	11	0.285 s.
Iris	150	5	0292. s.
diamonds	~53K	10	0.144 s.
baseball	~21K	22	0.339 s.
flights	~336K	19	0.522 s.
HCS Data set I	~400K	49	2.647 s.
HCS Data set II	~3.5mln	233	21.174 s.
HCS Data set III	~251K	1787	771.472 s.

4 Discussion

The PurifyR package demonstrates the ability to reliably carry out on-demand preprocessing of large data sets without the creation of multiple copies thanks to the package data.table. This allows researchers to confidently produce statistics and analyze imperfect data sets directly by running data through PurifyR. Researchers can build data processing pipelines during research and during quantification. PurifyR will clean data sets and highlight records and columns with missing data or outliers, which are not able to be used in downstream statistics and reports. A few public data sets from very small to moderate size such as mtcars, iris, baseball and flights plus three large data sets were subjected to ScanR for feature selection. Supplementary Data D provides the code and processing outcome of processing the data sets using PurifyR. Table 1 reports the size and speed of processing them using the PurifyR package.

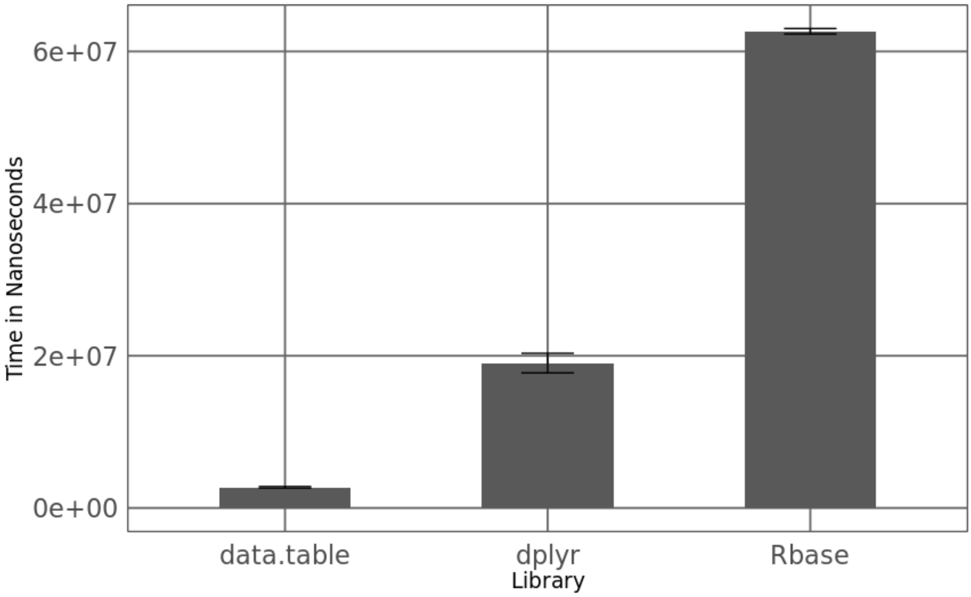


Figure 4. *Performance of three R frameworks*
This figure demonstrates the speed of a standard procedure, calculating a column-wise mean, on the same public dataset baseball. The calculation is carried out by the package data.table, dplyr and Rbase respectively (see x-axis). The y-axis represents the time in nanoseconds. The calculation is measured 100 times using the package microbenchmark. The bars visualize the mean and standard error of the measurements. The performance of data.table shows a 20 time speed-up compared to Rbase and >3 time speed-up compared to dplyr. See Supplementary Data E for the implementation. Results are generated using an AWS R5 xlarge EC2 instance (4 cores 32 GB RAM).

Complex data sets can require substantial time to compare results from previous experiments and represent potential roadblocks for further experimentation due to the excessive amount of time required for aggregating simple statistics on large and disparate data sets. Ideally, experimenters could check these statistics repeatedly during the research process, to identify issues and correct them before the experiment is complete or it becomes too late to address any collection issues.

Large data sets ideally require a rule-based approach to review the large number of automatically generated records and columns. Without automation, the results are difficult to reproduce and frequently prone to reporting errors. Additionally, manual curation of these types of data sets often require a great deal of time for cleansing and to standardize the values (Stodder, 2017). For example, to standardize data to common ranges in preparation for later statistical processing and machine learning steps. Missing data and outlier handling often prove to be complicated and subject to interpretation. This can consume up to 80% of the total time for analysis and reporting of results

(Wickham, 2014). The PurifyR package aims to automate a great part of this time by applying feature synthesis and engineering methods to automate data preparation steps. The PurifyR package automates common preprocessing steps to ensure high quality features are used and each row is prepared to meet the assumptions of later machine learning processing steps. Not only does the package reduces manual effort required and time to process data but will also improve transparency and the reproducibility of the results.

Ultimately, a framework for automating data preprocessing steps would remove much need for repetitive efforts and complex code for each analysis. Unfortunately, there is no golden standard, but there are a few statistical rules of thumb and recommendations in specific domains and methods (Birmingham et al., 2009; Costello & Osborne, 2005). It would simplify analysis and comparison with previous research and ensure a simple explanation that can be seen by all researchers. Moreover, it would help reproducibility move a step forward (Weigt et al., 2018; Szymańska et al., 2015). Many researchers do not need or desire to be involved in the intricate details of cleaning data and would prefer a more autonomous approach, where best practice standards are applied without a great deal of intervention. Ideally, researchers could quickly compute and re-analyze data sets without manual cleaning effort between each iteration.

Conflict of Interest: none declared.
DAE and WAO are both co-founders of Core Life Analytics B.V. a company in data analytics software.

Supplementary Data

Supplementary Data A

List of simple statistics measured in ScanR:

- MIN (minimum value)
- Q1 (1st quartile)
- Median
- Mean
- IQR (Inter Quartile Range)
- Q3 (3rd quartile)
- Max (maximum value)
- SD (Standard Deviation)
- MAD (Median Absolute Deviation)
- CV (Coefficient of Variation)
- SE (Standard Error of the Mean)
- NAs (not availables)
- COUNT
- PercentUnique (percentage of unique values)

Supplementary Data B

Configuration metrics:

maxNumberOfCores:
Specifies the number of cores used for multicore processing

SamplingPercentage:
Specifies the percentage sampling used when data sets contain over 10K data points

numberOfValuesDiscreteVariable:
Specifies the definition of a categorical variable. In other words, the number of unique values in a feature to be classified as a categorical feature

percentUniqueCutoff:
Specifies the percentage cutoff that is allowed to be unique without excluding the feature.

minSamplingSize:
Specifies the minimum number of cases to give a warning that the data set is small.

minSamplingSize:
Specifies the minimum number of cases to give a warning that the data set is large.

selectedTransformationBoundary:
Specifies the p-value for the decision boundary that a z-value of skewness of a variable determines that the feature requires a transformation

ClassVar:
Specifies the features that describes various classes in the data.

NormalizeVar:
Specifies the feature that is used for splitting the data to normalize the data per individual chunk

rowWiseMissingPercentage:
Specifies the percentage that is allowed for inclusion of data, (row wise).

imputationMethod:
Specifies the used imputation method applied to missing data

Supplementary Data C

HCS dataset with ~50 features and ~400k records

https://systemsmedicine.s3-eu-west-1.amazonaws.com/data/Bioinformatics/150001.fst

Supplementary Data D

Mtcars dataset

```
> DT = mtcars
> UoA = c('am', 'vs')
> DT = LoadR(DT)
> correlationCutoff = .99
> SamplingPercentage = 15
> percentUniqueCutoff = 5
> system.time(rs <- scanr(DT, correlationCutoff, SamplingPercentage,
percentUniqueCutoff))
  user  system elapsed
  0.976   0.064   0.285
```

Iris data set

```
> DT = datasets::iris
> UoA = c('Species')
> DT = LoadR(DT)
> correlationCutoff = .99
> SamplingPercentage = 15
> percentUniqueCutoff = 5
> system.time(rs <- scanr(DT, correlationCutoff, SamplingPercentage,
percentUniqueCutoff))
  user  system elapsed
  0.432   0.312   0.292
```

Diamonds dataset

```
> DT = ggplot2::diamonds
> UoA = c('color')
> DT = LoadR(DT)
> correlationCutoff = .99
> SamplingPercentage = 15
> percentUniqueCutoff = 5
> system.time(rs <- scanr(DT, correlationCutoff, SamplingPercentage,
percentUniqueCutoff))
  user  system elapsed
  0.416   0.084   0.144
```

Baseball dataset

```
> DT = plyr::baseball
> UoA = c('team')
> DT = LoadR(DT)
> correlationCutoff = .99
> SamplingPercentage = 15
> percentUniqueCutoff = 5
> system.time(rs <- scanr(DT, correlationCutoff, SamplingPercentage,
percentUniqueCutoff))
  user  system elapsed
  0.640   0.124   0.339
```

Flights dataset

```
> DT = nycflights13::flights
> UoA = c('flight')
> DT = LoadR(DT)
> correlationCutoff = .99
> SamplingPercentage = 15
> percentUniqueCutoff = 5
> system.time(rs <- scanr(DT, correlationCutoff, SamplingPercentage,
percentUniqueCutoff))
  user  system elapsed
  1.044   0.220   0.522
```

HCS Dataset I

```
>DT=data.table::fread('https://cla.stratominer.com/DATA/demoData/FullMit
oticCell_MatthieuNEGATIVEvsPOSITIVE.txt')
> UoA = c('wellLocation')
> initializeCPUS(4)
> DT = LoadR(DT)
> correlationCutoff = .99
> SamplingPercentage = 15
> percentUniqueCutoff = 5
> system.time(rs <- scanr(DT, correlationCutoff, SamplingPercentage,
percentUniqueCutoff))
  user  system elapsed
  4.840   1.256   2.647
```

HCS Dataset II

```
> UoA = c('Row', 'Column')
> DT = data.table::fread('MX985.txt')
|==================================================|
> DT = LoadR(DT)
> correlationCutoff = .99
> SamplingPercentage = 15
> percentUniqueCutoff = 5
> system.time(rs <- scanr(DT, correlationCutoff, SamplingPercentage,
percentUniqueCutoff))
  user  system elapsed
  35.240   6.756  21.174
```

HCS Dataset III

```
> UoA = c('Row', 'Column')
> DT = data.table::fread('41744.csv')
|==================================================|
> DT = LoadR(DT)
> correlationCutoff = .99
> SamplingPercentage = 15
> percentUniqueCutoff = 5
> system.time(rs <- scanr(DT, correlationCutoff, SamplingPercentage,
percentUniqueCutoff))
  user  system elapsed
  921.289  11.594 771.472
```

Supplementary Data E

Rbase, dplyr and data.table performance comparison Rscript:

https://systemsmedicine.s3-eu-west-1.amazonaws.com/data/Bioinformatics/RbaseDplyrdt.R

Supplementary Data F

Live Shiny Implementation:

https://purifyr.stratominer.com/Shiny

Github page:

https://github.com/womta/PurifyR

Install Devtools from CRAN:

```
install.packages("devtools")
```

Install PurifyR from Github:

```
devtools::install_github("womta/PurifyR")
```

Chapter 7 - The glucocorticoid mometasone furoate is a novel FXR ligand that decreases inflammatory but not metabolic gene expression

The Farnesoid X receptor (FXR) regulates bile salt, glucose, and cholesterol homeostasis by binding to DNA response elements, thereby activating gene expression (direct transactivation). FXR also inhibits the immune response via tethering to NF-κB (tethering transrepression). FXR activation therefore has therapeutic potential for liver and intestinal inflammatory diseases. We aim to identify and develop gene selective FXR modulators, which repress inflammation, but do not interfere with its metabolic capacity. In a high-throughput reporter-based screen, mometasone furoate (MF) was identified as a compound that reduced NF-κB reporter activity in an FXR-dependent manner. MF reduced mRNA expression of pro-inflammatory cytokines, and induction of direct FXR target genes in HepG2-GFP-FXR cells and intestinal organoids was minor. Computational studies disclosed three putative binding modes of the compound within the ligand binding domain of the receptor. Interestingly, mutation of W469A residue within the FXR ligand binding domain abrogated the decrease in NF-κB activity. Finally, we show that MF-bound FXR inhibits NF-κB subunit p65 recruitment to the DNA of pro-inflammatory genes CXCL2 and IL8. Although MF is not suitable as selective anti-inflammatory FXR ligand due to nanomolar affinity for the glucocorticoid receptor, we show that separation between metabolic and anti-inflammatory functions of FXR can be achieved.

Adapted From:

Bijsmans, I. T., Guercini, C., Pittol, J. M. R., Omta, W., Milona, A., Lelieveld, D., Egan, D.A., Pellicciari, R., Gioiello, A. & Van Mil, S. W. (2015). The glucocorticoid mometasone furoate is a novel FXR ligand that decreases inflammatory but not metabolic gene expression. *Scientific reports - Nat.*, 5, 14086.

Introduction

Nuclear receptors (NRs) are ligand-activated transcription factors regulating a large variety of target genes (Burris, et al., 2013) A variety of molecular mechanisms by which NRs regulate transcription have been identified (Hollman et al., 2012; Glass & Saijo, 2010). Classically, NRs directly bind to consensus response elements in target genes, thereby either activating or repressing transcription. In addition, NRs function independently of DNA binding, by tethering to other transcription factors (e.g. NF-κB, AP-1, or other NRs2). Since NR activity is regulated by specific ligands which can easily pass cell membranes, NRs are ideal drug targets. However, serious side-effects of the current NR drugs limit their utility due to activation of all transcriptional NR actions4. Therefore, many studies are undertaken to develop selective NR ligands5.

Farnesoid X receptor (FXR, also known as NR1H4) is a nuclear receptor activated by endogenous bile acids (BAs). Upon activation, FXR heterodimerizes with retinoid X receptor (RXR, also known as NR2B1) and binds FXR responsive elements in the promoters of target genes, leading to dissociation of the co-repressor complex, recruitment of co-activators, and transcription initiation. In this manner, FXR regulates bile salt concentrations in liver and intestinal cells by regulating bile salt transport systems, bile salt metabolism, and de novo synthesis (via SHP/FGF19 upregulation) from cholesterol in the liver (Lefebre et al., 2009). FXR also regulates glucose and fat homeostasis via direct binding to target genes (Matsukuma et al., 2006; Chong et al. 2010). We and others have shown that FXR also functions in a DNA-independent manner, by binding to NF-κB, thereby inhibiting NF-κB activity. This results in decreased pro-inflammatory cytokine expression in both liver and intestine (Gadaleta el al., 2011; Vavassori et al., 2009; Wang et al., 2008). FXR activation by a full agonist, obeticholic acid (OCA, 6-ECDCA), strongly improved clinical symptoms and histology of dextran sodium sulphate (DSS)- and trinitrobenzene sulphonic acid (TNBS)-induced colitis in wild type (WT), but not in FXR−/− mice. In addition, intestinal epithelial permeability was decreased, and pro-inflammatory cytokine mRNA expression was inhibited upon FXR activation (Gadaleta el al., 2011). This provides a clear rationale for further exploration of the use of FXR agonists as novel therapeutics for chronic inflammation of liver and intestine. In this context, the development of gene selective FXR modulators, referred to as SBARMs (selective bile acid receptor modulators), is particularly sought in view of their ability to modulate specific genes without affecting others, thus limiting potential side-effects of full FXR agonism upon chronic treatment. We therefore aim to develop selective anti-inflammatory FXR agonists, able to selectively interfere with the two molecular mechanisms by which FXR regulates metabolism and inflammation.

The previously reported high-throughput screening methods to identify FXR agonists are not suitable to detect anti-inflammatory FXR ligands. Co-factor recruitment assays, or comparable assays monitor the recruitment of a co-activator peptide upon ligand binding (Rouleau et al., 2003; Glickman et al., 2002), but this is not anticipated to happen when FXR tethers to NF-κB to inhibit its activity. It is expected that FXR recruits a co-repressor complex in this situation, as has been shown recently for FXR (Kim et al., 2015) and for other nuclear receptors (Saijo et al., 2009). For that reason, we developed an automated high-throughput Luciferase reporter assay to screen chemical libraries to identify compounds that decrease NF-κB activity in an FXR-dependent manner. Mometasone furoate (MF) was identified to regulate NF-κB activity, but not metabolic target genes, in the presence, but not in the absence of FXR, suggesting that separation of FXR anti-inflammatory actions from its metabolic actions is achievable by selective agonists.

Results

Luciferase reporter screen identifies five compounds decreasing TNFα-induced NF-κB activity

In figure 1, we schematically depict the current knowledge on the mechanisms by which FXR regulates metabolism (figure 1A) and anti-inflammatory (figure 1B) effects. We hypothesize that ligands might be able to separate FXR metabolic from anti-inflammatory functions because of the different mechanisms of direct versus tethered DNA binding. To identify FXR-dependent anti-inflammatory ligands, we set up an automated high-throughput Luciferase reporter assay to monitor NF-κB activity and used it to screen the Prestwick Chemical Library®. Ideally, the ligands repress NF-κB activity (figure 1C, left panel), but do not induce SHP or IBABP transcriptional activity (figure 1C, right panel). HEK293T cells transfected with a NF-κB reporter construct in combination with FXR and RXR expression plasmids were incubated with TNFα to activate NF-κB. Of the 1,200 tested drugs (figure 1D; depicted in purple), 34 drugs inhibited TNFα-induced transcriptional activity of the NF-κB reporter (figure 1E, depicted in black). Drugs showing low Renilla activity (suggesting compound cytotoxicity and/or low transfection efficiency) were excluded, yielding 4 candidate drugs significantly reducing NF-κB transcriptional activity (figure 1E, orange circles).

Although nicardipine hydrochloride did not significantly reduce NF-κB activity in the primary screen, we analyzed this compound further since it is structurally related to cilnidipine (a statistically significant hit), and nicardipine hydrochloride was recently shown to function as an FXR agonist (Hsu et al., 2014). Chemical structures of the candidate compounds are depicted in figure 1F. Taken together, five compounds decreased TNFα-induced NF-κB activity (figure 1G).

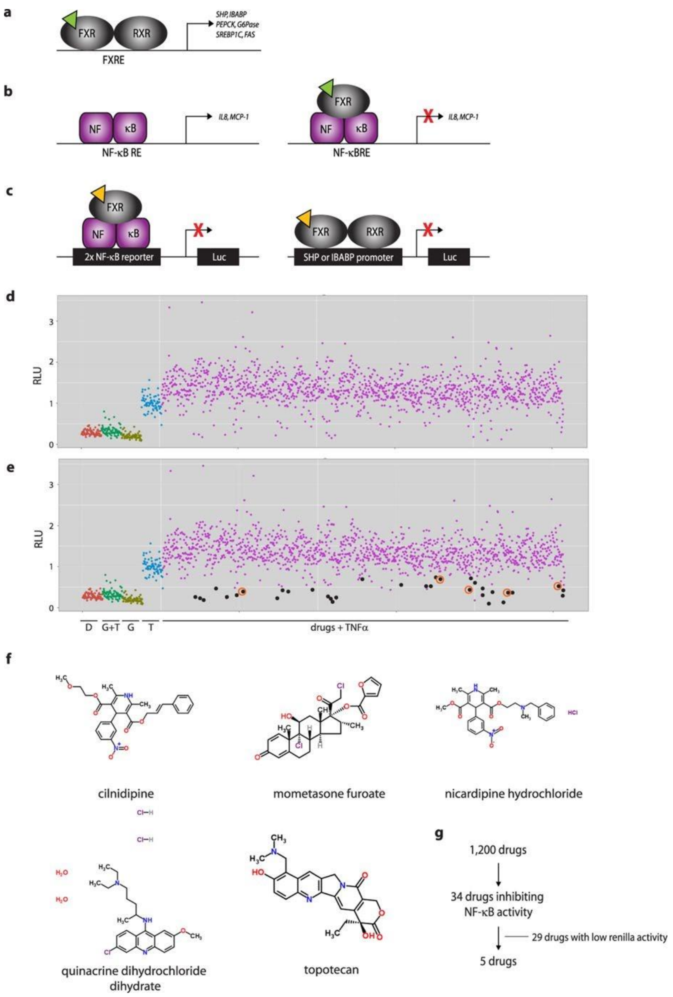


Figure 1. *Schematic representation of the molecular FXR actions in regulation of bile salt Glucose and fat metabolism via direct DNA binding (A) and ameliorating inflammation via tethering transrepression of NF-κB (B). Ligand (green triangle) activated FXR binds to FXR responsive elements (FXREs), thereby activating target genes involved in bile salt homeostasis (SHP, IBABP), glucose (PEPCK, G6Pase), and fat metabolism (SREBP1C, FAS). Binding of NF-κB to its responsive element (NF-κB RE) results in expression of pro-inflammatory cytokines, such as IL8 and MCP-1. FXR binding to NF-κB inhibits this activity, thereby decreasing pro-inflammatory cytokine expression. We have set up an automated high-throughput NF-κB Luciferase reporter assay to test FXR-dependent reduction of NF-κB activity (C). We screened the Prestwick library containing 1,200 FDA approved drugs (yellow triangle) using this assay (left panel). Candidate drugs were subsequently tested for IBABP and SHP transactivation. Figures D-G depict hit selection. Figures D and E show the overall view of the screen. Indicated with black dots are the 34 drugs reducing TNFα-induced NF-κB transcriptional activity significantly ($p < 0.05$) (E). Low Renilla values were considered to reflect poor transfection efficiency or cytotoxicity and were therefore eliminated, leaving 5 compounds significantly reducing NF-κB activity (indicated with black dots surrounded by orange circles. D: DMSO; G+T: GW4064 + TNFα; G: GW4064; T: TNFα (E). Figure F depicts the chemical structures of the five candidate compounds. Flowchart of hit selection is shown in (G).*

The glucocorticoid mometasone furoate is a novel FXR ligand that decreases...

Mometasone furoate is an FXR modulator with predominant anti-inflammatory activity

In figure 2A we validated that the five compounds identified in our screen reduced NF-κB activity in a separate reporter assay. To investigate whether these 5 compounds repress NF-κB activity in an FXR-dependent manner, reporter assays were repeated comparing empty vector (pcDNA3.1) with FXR transfected cells. Three compounds, cilnidipine (C), mometasone furoate (MF), and nicardipine hydrochloride (NH) significantly decreased NF-κB activity in FXR transfected cells. Quinacrine dihydrochloride dihydrate (QDD) and topotecan (T) decreased NF-κB activity independent of FXR, possibly via binding to other nuclear receptors (figure 2B). C and NH were recently described to have transactivation activity (Hsu et al., 2014), and would thus also affect FXR metabolic function, therefore, upcoming experiments were performed for MF only. Figure 2C shows that NF-κB activity was reduced in a dose-dependent manner upon MF treatment. In conclusion, our screen revealed MF as a compound that has FXR-dependent anti-inflammatory properties.

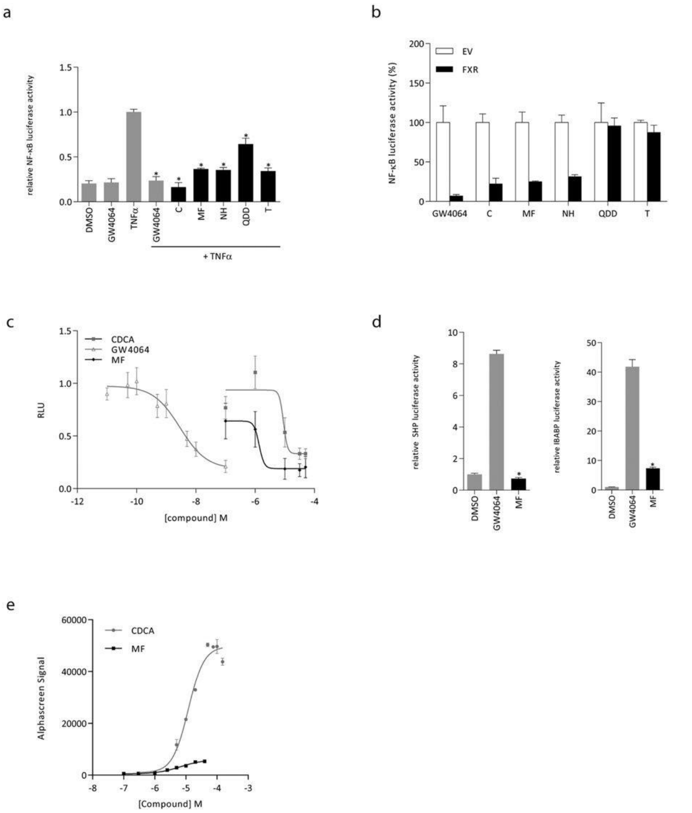


Figure 2.
*(A) Validation of NF-κB Luciferase reporter assay. HEK293T cells transfected with NF-κB reporter, expression plasmids for FXR and RXR, and pTK-Renilla construct were treated with DMSO, GW4064 (1 µM), TNFα (5 ng/ml), GW4064 plus TNFα, or the indicated compounds (10 µM) in the presence of TNFα, for 24 hours. Cilnidipine (C), mometasone furoate (MF), nicardipine hydrochloride (NH), quinacrine dihydrochloride dehydrate (QDD), and topotecan (T) significantly reduced TNFα-induced NF-κB activity. The reporter assay was performed in quadruplicate in three independent experiments. Each bar represents the mean ± SD of one representative experiment. *p < 0.001 as compared to TNFα treated cells. (B) FXR-dependent reduction of NF-κB transcriptional activity. The assay in (A) was repeated with empty vector (EV; white bars) and FXR overexpressing cells (black bars). Data are normalized to the EV activity for each compound. (C) Dose-response curve. MF treatment reduced TNFα-induced NF-κB activity in a dose-dependent manner, with an IC50 value of 1.4 µM. IC50 values of CDCA and GW4064 are 7.9 µM and 3.5 nM respectively. (D) Transactivation reporter assay SHP (left panel) and IBABP promoters (right panel). HEK293T cells transfected with SHP or IBABP promoter constructs, FXR and RXR, and Renilla, were treated with DMSO, 1 µM GW4064, or 10 µM MF for 24 hours. Data presented show one representative experiment of 4 performed experiments. Each bar represents the mean ± SD. *p < 0.001 compared to GW4064 treated cells. (E) FXR coactivator recruitment assay (AlphaScreen). Ligand binding domain of FXR (FXR-LBD) was incubated with increasing amounts of MF or CDCA to examine SRC-1 recruitment. Assay performed in triplicate. One representative experiment is shown.*

Since we aim to develop selective anti-inflammatory FXR agonists, we next determined the capacity of MF to induce transcription of target genes via direct DNA binding to SHP and IBABP promoters. HEK293T cells were transfected with FXR, RXR, and either a SHP or IBABP promoter reporter construct. GW4064 induced a strong response for both promoters (9 and 42-fold respectively). MF showed no SHP, and a strongly reduced IBABP promoter activity compared to GW4064 (figure 2D). In addition, we characterized the binding potency of MF as ligand for FXR by performing an FXR-coactivator recruitment assay. In this assay, ligand binding induces the recruitment of the co-activator SRC-1 to the FXR ligand binding domain (LBD). MF appears a partial agonist with an EC50 of 10.9 ± 3.8 μM and efficacy of 12% compared to CDCA (figure 2E). In summary, GW4064 both activates transcription of SHP and IBABP promoters and inhibits transcription of the NF-κB promoter. In contrast, MF inhibits NF-κB activity comparable to GW4064 and CDCA, with low or absent activity on SHP and IBABP promoters.

Mometasone furoate reduces endogenous pro-inflammatory gene expression in HepG2 cells and intestinal organoids in an FXR-dependent manner. To extend the finding that MF reduced the NF-κB activity in an FXR dependent manner, we analyzed endogenous FXR and NF-κB target gene expression in HepG2 cells stably overexpressing GFP (HepG2-GFP) or GFP-FXR (HepG2-GFP-FXR). Cells were stimulated with TNFα to induce NF-κB activity. Indeed, NF-κB target genes IL8, MCP-1, and CXCL2 increased upon TNFα stimulation. Co-stimulation with MF or GW4064 abolished this effect in HepG2-GFP-FXR cells but not in HepG2-GFP cells (figure 3A), indicating that FXR activation by GW4064 or MF blocks NF-κB activity. Direct FXR target genes SHP, FGF19, KNG1, SDC1 and ICAM16 mRNA expression was induced upon GW4064 treatment, however, only a minor increase was detected in HepG2-GFP-FXR cells treated with MF (figure 3B). To test whether MF also selectively affects pro-inflammatory gene expression in a model system closer to the in vivo situation, we have derived organoids from small intestines from WT and FXR−/− mice, as described in Sato et al. (2009). We show that GW4064 and MF reduced Tnfα and Cxcl2 expression only in WT but not in FXR−/− organoids (figure 3C). Notably, MF also showed FXR-independent decreases in Tnfα and Cxcl2 expression, presumably via activating other NRs such as glucocorticoid receptor (GR). Also in concurrence with the HepG2 model system, in WT but not in FXR−/− organoids treated with GW4064, direct FXR target gene expression of Shp, Fgf (Saijo et al., 2009) and Ibabp is increased, which is absent or decreased upon MF stimulation (figure 3D), indicating that the effect is mediated by FXR. These data independently confirm that compared to GW4064, MF has equal capacity to inhibit NF-κB target gene expression, but not in regulating direct FXR target genes. This suggests that MF is a gene selective FXR modulator.

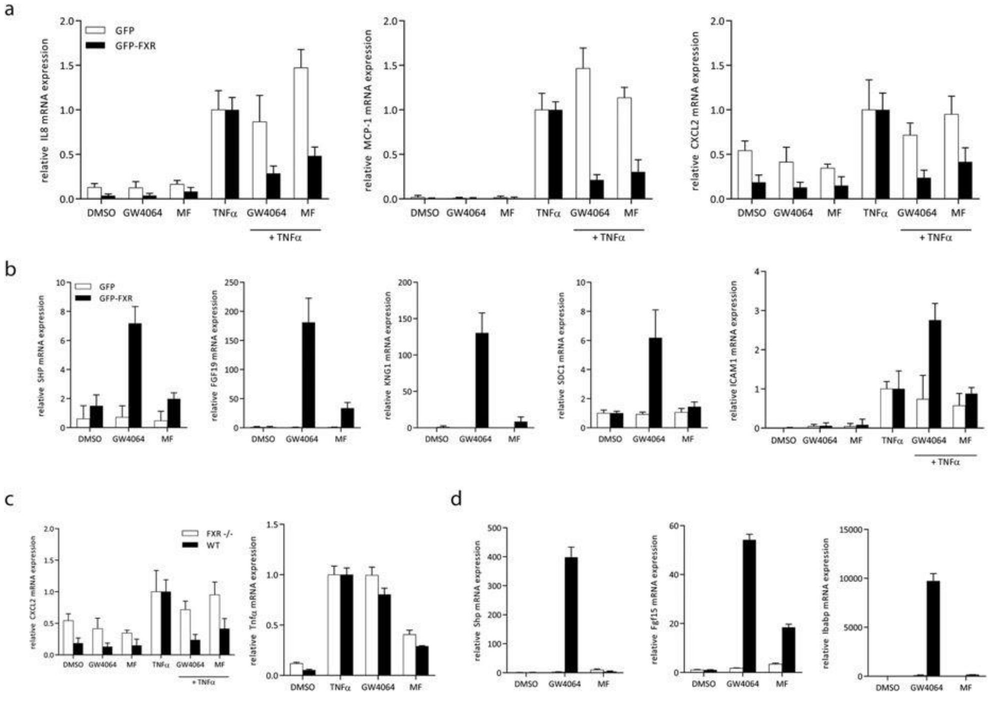


Figure 3. *Endogenous FXR target gene expression in HepG2 cells stably overexpressing GFP (HepG2-GFP; white bars) or GFP-FXR (HepG2-GFP-FXR; black bars). (A) Cells were treated in triplicate with DMSO, 1 μM GW4064, 10 μM MF, 5 ng/ml TNFα, GW4064 plus TNFα, or TNFα plus MF for 24 hours. IL8, MCP-1, and CXCL2 mRNA expression was analyzed by qRT-PCR in duplicate. (B) HepG2-GFP and HepG2-GFP-FXR cells were treated with DMSO, 1 μM GW4064 or 10 μM MF in triplicate for 24 hours. SHP, FGF19, KNG1, SDC1 and ICAM1 mRNA expression was analyzed by qRT-PCR in duplicate. Each bar represents mean ± SD. (C, D) Small intestine derived organoids from 3WT and FXR−/− mice were treated with DMSO, 1 μM GW4064, 10 μM MF, 5 ng/ml TNFα, GW4064 plus TNFα, or TNFα plus MF for 24 hours. mRNA expression of each organoid line was analyzed by qRT-PCR in duplicate. Each bar represents mean ± SEM.*

Computational studies reveal three putative binding modes of mometasone furoate to FXR

Docking calculations and molecular dynamic simulations were carried out to explore the putative binding mode of MF to FXR. Since twenty-five agonist bound FXR LBD co-crystal structures are currently available on RCSB Protein Data Bank (Berman et al., 2000), we decided to use Phase Shape Screening to identify the FXR crystal structure with the suitable conformation able to accommodate MF. Therefore, all the available crystallographic structures of FXR agonists were screened using MF as query structure to determine the highest pharmacophore-based shape similarity between the compound and co-crystallized ligands. OCA (Pellicciari et al., 2002) was found to be the ligand endowed with the greater shape similarity. Accordingly, the co-crystal structure of FXR LBD in complex with OCA (pdb code: 1OSV) (Mi et al., 2003) was selected for the docking calculations. MF was flexibly docked into FXR LBD using Glide software. Ten docking solutions were retrieved and clustered into three different binding modes (clustering criterion: RMSD < 2 Å). Three poses, namely "binding mode" (Burris et al., 2013; Hollman et al., 2012), representative for each cluster, were selected based on the best docking score (XP Gscore; see Table 1). Binding mode 1 represents the energetically favored docking pose (XP Gscore −10.02) and the most represented binding mode, characterized by the furoate group buried inside the FXR binding site establishing positive hydrophobic contacts with helix 7 (figure 4A). Binding mode 2 is like binding mode 1. In this case, MF orientation in the FXR binding cavity is rotated approximately 90° pointing the furoate group towards a region between helix 11 and helix 12 (figure 4A). The less represented binding mode 3 has a low binding score compared to binding modes 1 and 2 (Table 1). The docking pose is head-to-tail flipped so that the MF steroid A ring points towards the core of the FXR LBD with the furoate group oriented towards the solvent in a region between helix 5 and helix 6 (figure 4A). It is interesting to note that the orientation of MF proposed by binding modes 1 and 2 is like that experimentally observed for MF when it binds to GR (He et al., 2014) and progesterone receptor (PR) (Madauss et al., 2004).

Table 1.

Cluster of docking solutions	No of poses in cluster	Top XP Gscore (kcal/mol)	RMSD (Å) versus top ranked solution
Binding mode 1	6	−10.02	0
Binding mode 2	3	−8.61	4.41
Binding mode 3	1	−8.46	8.50

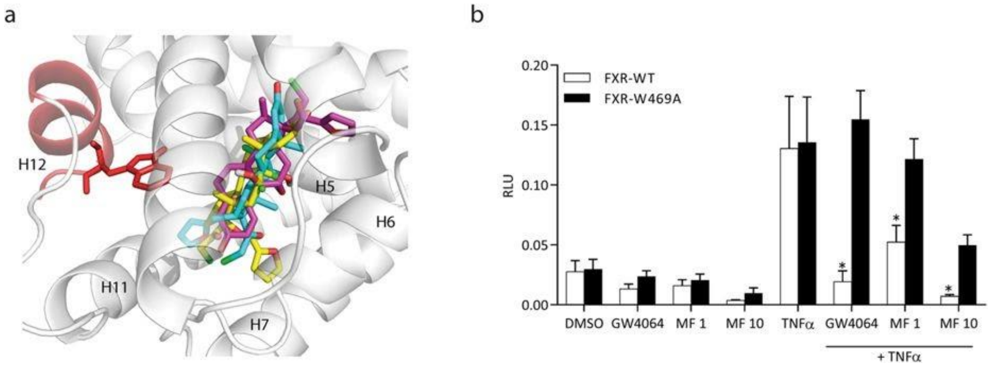


Figure 4. *Three different binding modes of MF to FXR, as determined by docking studies, are depicted (A) Binding mode 1 suggests that the furoate group of MF is buried into the FXR binding site pointing towards helix 7 (yellow carbons); binding mode 2 is characterized by the furoate group of MF oriented toward helix 11 and helix 12 (blue carbons); binding mode 3 is head-to-tail flipped with respect to binding modes 1 and 2 with the furoate group of MF oriented toward the helix 5 and 6 (magenta carbons). Helix 12 is shown in red. (B) HEK293T cells transfected with NF-κB reporter, expression plasmids for wild type FXR (FXR-WT) or mutant FXR (FXR-W469A) and RXR, and pTK-Renilla construct were treated with DMSO, 1 μM GW4064, 1 or 10 μM MF as indicated, or TNFα, for 24 hours. The NF-κB Luciferase reporter assay was performed in quadruplicate. *p < 0.001 compared to FXR-W469A cells treated with GW4064 or MF plus TNFα.*

To support the computational studies, we performed a Luciferase assay in which we compared NF-κB transcriptional activity of wild type FXR (FXR-WT) with mutant FXR (FXR-W469A). The mutated amino acid is located in helix 12, a crucial portion of the FXR LBD involved in the ligand-dependent cofactor recruitment. Upon GW4064 and MF stimulation, FXR-W469A showed significantly reduced transrepression activity compared to FXR-WT, suggesting that MF action is dependent upon the ligand binding domain of FXR (figure 4B).

Next, we assessed the stability of the observed binding modes performing 100 ns of molecular dynamic simulation for the three selected MF/FXR docking complexes. Binding mode 1 was found to be maintained over the simulation time suggesting a high stabilization (MF average RMSD with respect to the starting conformation of 1.59 ± 0.30 Å; figure 5A). Moreover, we observed that i) the MF steroid core establishes good hydrophobic contacts with the side chains of Trp451 and Phe326, ii) the furoate group is oriented towards helix 7 thus positively interacting with side chains of Tyr358, Phe363 and Ile349, iii) the carbonyl group at C3 position engages a hydrogen bond with Arg328 side chain (figure 5A). A greater stabilization was observed during the molecular dynamic simulation of binding mode 2. Here, MF undergoes a slight conformational adjustment during the first picoseconds of the simulation until reaching a more stable conformation that is maintained throughout the simulation time (MF average RMSD with respect to the starting conformation of 2.26 Å ± 0.20 Å) (figure 5B). Accordingly, binding mode 2 is stabilized by both hydrophobic and polar interactions defined as follows: i) particularly stable hydrogen bonds between the hydroxyl group at the C11 β-position and the side chains of Ser329 and His291, ii) hydrogen bonds between the carbonyl groups of C17 substituents and Tyr358 and His444 side chains, iii) π-π and hydrophobic interactions of the furoate group with the aromatic side chains of Phe281, Trp451 and Trp466 (figure 5B). Therefore, giving the high number of the established favorable contacts, the interaction energy between MF and FXR during the molecular dynamic simulation of binding mode 2 was slightly greater than those determined for the simulation of binding mode 1 (binding mode 1 interaction energy = −73.16 ± 6 kcal/mol; binding mode 2 interaction energy = −81.83 ± 3.86 kcal/mol). Analysis of the molecular dynamic simulation of MF binding mode 3 revealed marked MF conformational changes for the first picoseconds of simulation during which MF is shifted deeper into the binding site. This movement is reflected in an average RMSD from the starting conformation of 5.71 Å ± 0.50 Å (figure 5C). Moreover, MF undergoes further conformational adjustments for the whole simulation time, suggesting a lower degree of stabilization of this complex despite the favorable protein-ligand interaction energy (−75.94 ± 5.48 kcal/mol). During the simulation, the C11 β-OH group of MF was found to interact stably with the polar side chain of Tyr366 through a hydrogen bond while the furoate carbonyl moiety is involved in an additional hydrogen bond interaction with the side chain of His341. In addition, the ring A of the steroid scaffold engages Trp466 side chain in stable hydrophobic contacts and the furoate group establishes good hydrophobic interactions with the side chains of Leu284 and Leu345.

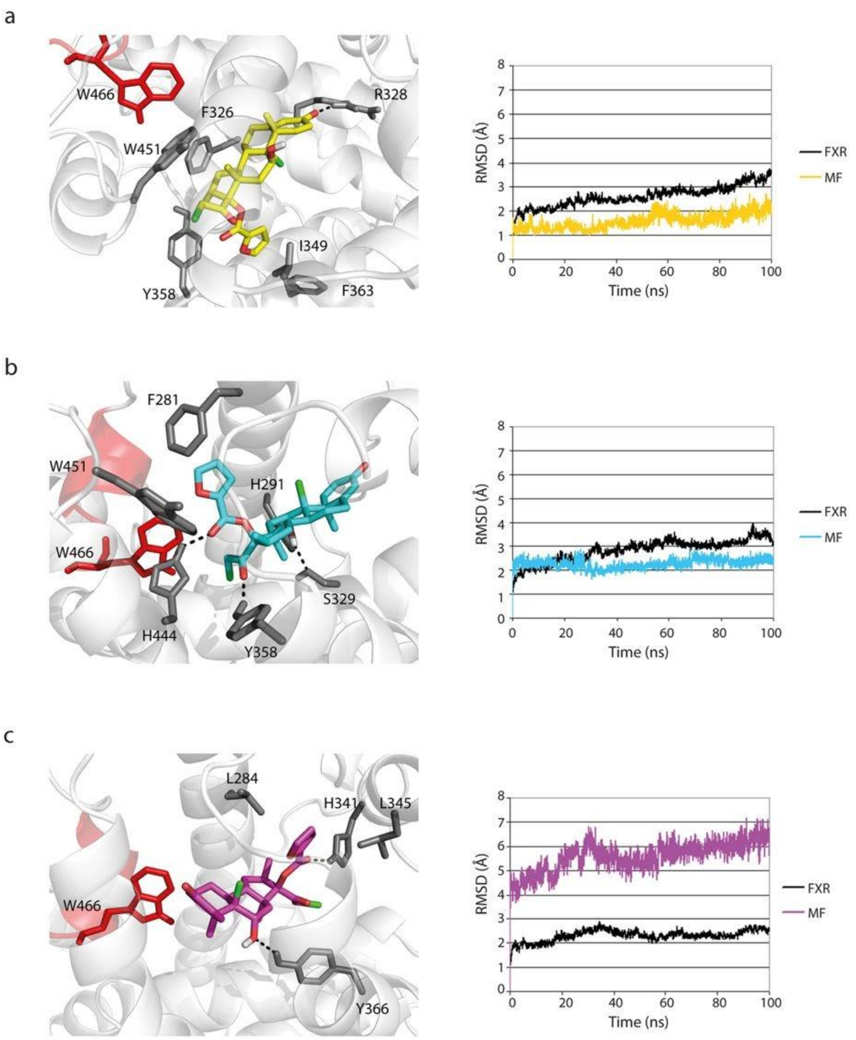


Figure 5. *Graphical representation of the MF-FXR interactions and RMSD*
Calculated during 100 ns of molecular dynamic simulations of the three binding modes suggested by docking studies. (A) binding mode 1 is stabilized mainly by hydrophobic interactions; (B) binding mode 2 is the most stabilized complex due to the high number of both hydrophobic and hydrophilic contacts conserved; (C) binding mode 3 is the less stable binding mode despite conserved interactions are established during the simulation.

GW4064 and MF reduce p65 recruitment to pro-inflammatory gene promoters

Upon an inflammatory stimulus, NF-κB subunits translocate to the nucleus and bind to target regions in the genome, leading to activation of target genes. To investigate whether recruitment of NF-κB to promoters is altered upon MF and GW4064 stimulation, we performed chromatin IPs for p65, the main activating NF-κB subunit, in HepG2-GFP and HepG2-GFP-FXR cells (figure 6). We show that at 4hours after GW4064 and MF stimulation, p65 binding to the CXCL2 (figure 6A) and IL8 (figure 6B) promoters is reduced in HepG2-GFP-FXR cells, but not in HepG2-GFP cells. Control regions (SHP and β-hemo promoters) showed no p65 recruitment (data not shown). We have previously shown that FXR binds p65 in GST-pull down experiments (Gadaleta et al., 2011). We propose that this interaction with FXR prevents p65 binding to the IL8 and CXCL2 promoters, leading to the observed reduction in transcriptional activation.

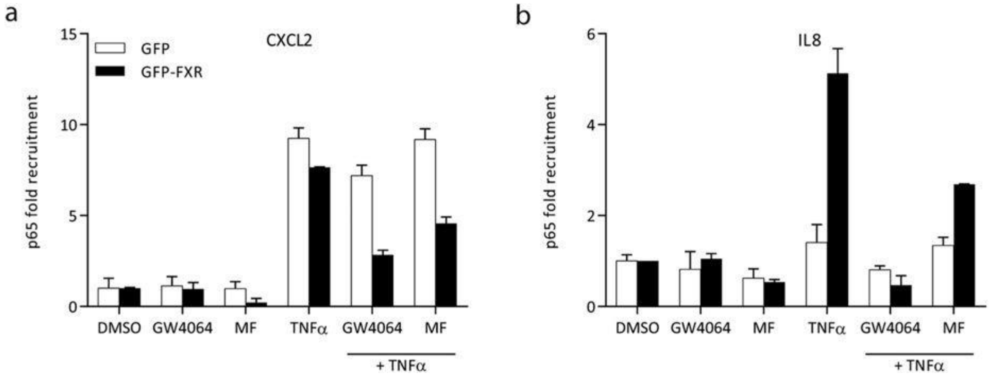


Figure 6.

HepG2-GFP and HepG2-GFP-FXR cells were treated with DMSO, GW4064, MF, and TNFα, as indicated. Cross-linked chromatin from these cell lysates was precipitated using an anti-p65 antibody and analyzed by qPCR with primers specific to the known p65 binding regions of CXCL2 and IL8.

Discussion

FXR is regarded as a promising drug target for many liver and gastrointestinal disorders. At present, FXR agonists are in Phase II and III clinical trials for nonalcoholic fatty liver disease (NAFLD), nonalcoholic steatohepatitis (NASH), primary sclerosing cholangitis (PBC), and primary biliary cirrhosis (PSC) (www.clinicaltrial.gov). We know that FXR regulates transcription of many target genes involved in BA, glucose, and fat metabolism. To bypass potential side-effects from treatment, selective FXR modulation might be advantageous over full agonism. We hypothesized that gene-selective ligands may separate two different molecular mechanisms by which FXR functions, i.e. by direct DNA binding or by tethering to another transcription factor. Current knowledge suggests that FXR regulation of BA, glucose and fat metabolism involves direct FXR binding to promoter sequences. In contrast, FXR anti-inflammatory actions have been shown to occur via binding of FXR to NF-κB, thereby inhibiting its activity (Gadaleta el al., 2011; Vavassori et al., 2009; Kim et al., 2015). In order to identify FXR ligands that can inhibit NF-κB activity, we have set up an automated Luciferase reporter assay. We tested 1,200 drugs using a high-throughput Luciferase screen to determine their capacity to inhibit NF-κB. Of the 5 compounds which significantly reduced TNFα-induced NF-κB transcriptional activity, 3 compounds operated in an FXR-dependent manner: nicardipine hydrochloride, cilnidipine, and MF. Transactivation of SHP and IBABP promoters by MF was minimal compared to the full agonist GW4064. Our data confirm the recent study by Hsu et al., who also identified nicardipine hydrochloride and cilnidipine as FXR modulators (Hsu et al., 2014). Since Hsu et al. identified these compounds by a coactivator recruitment assay, which is corresponding to the direct DNA binding of FXR, we did not pursue them further in this paper. Instead, our attention was directed towards MF because of its steroid-like structure, a scaffold that particularly fits with the LBD of FXR (Gioiello et al., 2014). MF is a nanomolar glucocorticosteroid characterized by chlorine substituents at the C9α- and C21-position, endowed with antipruritic, anti-inflammatory, and vasoconstrictive properties. MF has not only high affinity for GR, but also for other NRs, including PR and mineralocorticoid receptor (MR) (Madauss et al., 2004; Austin et al., 2002). Like other corticosteroids, the anti-inflammatory efficacy of MF is mediated by the repression of inflammatory gene transcription either directly (via transrepression) or by activating transcription of anti-inflammatory/repressive factors (transactivation) (King et al., 2013). Here, we show that MF binds to and inhibits NF-κB activity in a dose responsive and FXR-dependent manner, both in reporter assays and on endogenous pro-inflammatory genes in HepG2 cells stably overexpressing FXR and intestinal organoids derived from WT and FXR-/- mice. These results are comparable to the full FXR agonist GW4064. However, FXR target genes which are activated upon direct FXR binding to their promoters did show no or reduced expression in all model systems upon MF treatment. We therefore conclude that it is possible to separate FXR direct transactivation from FXR mediated tethered transrepression.

The FXR-mediated decrease in expression of pro-inflammatory genes by MF and GW4064 presumably does not involve the recruitment of co-activators, but rather involves binding of FXR to p65, thereby prohibiting its binding to promoters (see figure 6) (Wang et al., 2008; Gadaleta et al., 2011). The results of the AlphaScreen assay (figure 2E) concur with this hypothesis, because incubation with MF results in low efficacy for the recruitment of the coactivator peptide of SRC-1 to the FXR-LBD as compared to CDCA (12%), while the effect on mRNA expression of pro-inflammatory genes is similar. Docking analysis and molecular dynamic simulations suggested three putative binding modes of MF to FXR: binding mode 1 is characterized by the furoate group buried inside the FXR binding site towards helix 7; binding mode 2, similar to binding mode 1, but rotated approximately 90°, where the furoate group is oriented in a region between helix 11 and helix 12; and binding mode 3, head-to-tail flipped pose with respect to binding modes 1 and 2, where the steroid A ring points towards the core of the FXR LBD and the furoate group is oriented towards the solvent in a region between helix 5 and helix 6. Molecular dynamics simulations of the three proposed MF binding modes revealed that binding mode 1 is stable over the simulation time as the result of several interactions, mainly hydrophobic. Binding mode 2 was found to be even more stable, indeed, the lower standard deviation in the RMSD values accounts for a greater stabilization of the ligand inside the binding site. Binding mode 2 was also found to be the slightly favored MF binding mode from an energetically point of view because of both hydrophobic and hydrogen bond interactions. Binding mode 3 showed slightly less stabilization in terms of MF conformation despite a good energy content due to the conserved polar and hydrophobic contacts established during the molecular dynamic simulation. Although not definitive, all together these results suggest that binding mode 2 may represent the most probable binding mode of MF to FXR, though we cannot exclude binding mode 1 and 3 as plausible alternative solutions.

Considering that MF is a clinical drug against persistent asthma as well as a well-established treatment for a variety of inflammatory corticosteroid-responsive dermatoses, such as chronic hand eczema, atopic dermatitis (AD), seborrhoeic dermatitis, and psorias (Bourke, Coulson & English, 2009), our findings reveal an alternative template for design of FXR ligands with therapeutic potential to rapid clinical applications by providing a safe lead compound. Structural modifications on the MF scaffold are therefore particularly sought to increase FXR potency and selectivity. In conclusion, we show that MF selectively activates FXR anti-inflammatory actions. Although MF itself is not suitable to pursue as an FXR selective ligand, this opens an exciting new avenue for selective FXR agonism and future opportunities for treatment of chronic inflammation of the liver and gut.

Methods

Materials

The Prestwick Chemical Library® containing 1,200 FDA/EMA approved drugs as 10 mM stock solutions in 96 wells plates was purchased from Prestwick (Prestwick Chemical, Illkirch, France). Cilnidipine (C), mometasone furoate (MF), nicardipine hydrochloride (NH), quinacrine dihydrochloride dehydrate (QDD), topotecan (T), GW4064, and CDCA were purchased from Sigma-Aldrich, TNFα and Prot A beads from Roche. Di(N-succinimidyl) glutarate (DSG), 97% was obtained from Synchem UG & Co. Antibody used for p65 (C-20) ChIP was bought from Santa Cruz.

Cell and organoid culture

HEK293T cells were grown in Dulbecco's modified Eagle's medium (DMEM, Sigma-Aldrich), supplemented with 10% FCS and 1% penicillin/streptomycin (Sigma). HepG2 cells were cultured in DMEM supplemented with 10% FCS, 1% penicillin/streptomycin, and 1% L-glutamine (Lonza; Verviers, Belgium). Medium of HepG2 cells stably expressing pLenti-CMV-neo-GFP (HepG2-GFP) or pLenti-CMV-neo-GFP-FXR (HepG2-GFP-FXR) constructs was supplemented with 100 μg/μl G418. GFP-FXR was cloned in pLenti CMV Neo DEST (705-1), a gift from Eric Campeau (Addgene plasmid #17392) (Campeau et al., 2009). Virus production and transduction of HepG2 cells was done following standard procedures. Small intestines (SI) were isolated from 3WT and 3FXR−/− mice, washed with cold PBS, the epithelial layer gently scraped. The remaining tissue with intact crypts was washed several times with cold PBS and subsequently incubated with 5 mM EDTA/PBS for 1 hour to isolate crypt cells as previously described (Sato et al., 2009). The crypt cells were embedded in matrigel and seeded in 24 well plates containing advanced DMEM/F12 supplemented with penicillin/streptomycin, HEPES, Glutamax, Wnt, n-Acetyl-cysteine, growth factors (Noggin, R-spondin, mEGF), B27, Y-27632, A83-01 and p38 inhibitor.

Automated high-throughput Luciferase reporter assay

We screened the Prestwick Chemical Library® using a Luciferase reporter assay. HEK293T cells were bulk transfected with a NF-κB responsive element reporter construct (pGL2-2κB), pcDNA3.1-FXRα2 or pcDNA3.1-FXRα2-W469A, and pcDNA3.1-RXRα expression plasmids, and pTK-Renilla as an internal transfection control. The next day, cells were transferred to poly-l-lysine

(Sigma-Aldrich) coated 384-well plates (Corning) using Multidrop™ Combi Reagent Dispenser (Thermo Scientific). Cells were stimulated in triplicate with vehicle (DMSO), GW4064 (1 µM), TNFα (5 ng/ml), or with the Prestwick Chemical Library® (10 µM) in presence of TNFα for 24 hours using the Caliper Sciclone liquid handling robot, followed by cell lysis, and firefly and Renilla Luciferase measurement (Dual-Luciferase®Reporter AssaySystem; Promega) using a SpectraMax® M5e Multi-Mode Microplate Reader (Molecular Devices).

To determine transactivation, HEK293T cells were transfected with either pGL2-SHP or pGL3-IBABP promoter constructs in combination with pcDNA3.1-FXRα2, pcDNA3.1-RXRα, and pRL-CMV Renilla plasmids. Cells were stimulated with vehicle (DMSO), GW4064 or MF for 24 hours. Subsequently, cells were lysed, and Luciferase activity was determined.

Data normalization and hit selection

The Prestwick library consisted of four 384-well plates. Each library plate was assayed in triplicate. Data were normalized to the average TNFα value of the corresponding library plate. To perform hit selection, the Manhattan distance score for each drug and control value for each individual library plate was calculated against the average score of the TNFα values. This calculation was based on RLU values, i.e. the Luciferase/Renilla ratio. The distance scores were transformed into p-values using the Cumulative Distribution Function. Hits were considered statistically significant if $p \leq 0.05$. Only drugs that significantly inhibited NF-κB transcriptional activity were considered for further analysis. Subsequently, drugs showing low Renilla activity, suggesting drug cytotoxicity and/or low transfection efficiency, were eliminated.

mRNA isolation and quantitative RT-PCR

HepG2-GFP and HepG2-GFP-FXR cells and SI organoids were treated with vehicle (DMSO), 1 µM GW4064, 5 ng/ml TNFα, TNFα plus GW4064, or 10 µM MF in the presence or absence of TNFα for 24 hours. RNA was isolated using Trizol® reagent (Ambion/Life Technologies). RNA was reverse transcribed using Superscript II (Invitrogen) according to manufacturer's protocol. Quantitative RT-PCR was performed using Fast start Universal SYBR Green Master mix (Roche) and primers for FXR target genes on CFX384™ Real-Time system (Biorad). Target gene expression was normalized to housekeeping gene β2-microglobulin (HepG2) or cyclophilin A (organoids). Primers are listed in Table 2. Data are presented as fold change.

Table 2. Primers

Gene	Forward primer (5′-3′)	Reverse primer (5′-3′)
hB2M	GGCTATCCAGCGTACTCCAAA	CGGCAGGCATACTCATCTTTTT
hCXCL2	CCCATGGTTAAGAAAATCATCG	CTTCAGGAACAGCCACCAAT
hICAM1	CCTTCCTCACCGTGTACTGG	AGCGTAGGGTAAGGTTCTTGC
hFGF19	CGTGCGGTACCTCTGCAT	TCTCCTCCTCGAAAGCACA
hIL8	GGAAGGAACCATCTCACTGTG	GGGTGGAAAGGTTTGGAGTA
hKNG1	AGTAAAACGGGCCCAAAGAC	TCGTTTGCACAATTGAGTAGGT
hMCP-1	CAGCCAGATGCAATCAATGCC	TGGAATCCTGAACCCACTTCT
hSDC1	AGGATGGAGGTCCTTCTGC	CCGAGGTTTCAAAGGTGAAGT
hSHP	AGGGACCATCCTCTTCAACC	TTCACACAGCACCCAGTGAG
mCxcl2	AAAATCATCCAAAAGATACTGAACAA	CTTTGGTTCTTCCGTTGAGG
mCyclophilin A	GGAGATGGCACAGGAGGAA	GCCCGTAGTGCTTCAGCTT
mFgf15	AAAACGAACGAAATTTGTTGGAA	ACGTCCTTGATGGCAATCG
mIbabp	TTGAGAGTGAGAAGAATTACGATGAGT	TTTCAATCACGTCTCCCTGGAA
mShp	CGATCCTCTTCAACCCAGATG	AGGGCTCCAAGACTTCACACA
mTnfα	ACGGCATGGATCTCAAAGAC	AGATAGCAAATCGGCTGACG
hCXCL2 promoter	ATGGTTGGGGCTGGAAAG	CGCCTTCCTTCCGAACTC
hIL8 promoter	CATCAGTTGCAAATCGTGGA	AGAACTTATGCACCCTCATCTTTT

Co-factor recruitment assay

FXR co-factor recruitment was assayed using the AlphaScreen technology according to manufacturer’s instructions. In brief, GST-tagged FXR-LBD was coupled to anti-GST-acceptor beads, and biotinylated-SRC-1 peptide (co-activator) to streptavidin donor beads. Presence of ligand induces a conformational change of the LBD, followed by co-activator binding. Upon illumination at 680 nm, energy is transferred from the donor to acceptor beads, generating a luminescent signal. Biotinylated SRC-1, GST-FXR and ligand were incubated for 1 hour. Detection mix (donor and acceptor beads) was added, followed by 4-hour incubation. Reading was performed using Envision®Multilabel Reader (Perkin Elmer). Dose response curves were performed in triplicate and EC50 values were determined.

Docking studies and molecular dynamics simulations

Chemical structure of MF was drawn using Maestro building fragment tool (Maestro, version 9.3, Schrödinger, LLC, New York, NY, 2012). LigPrep software (LigPrep, version 2.5, Schrödinger, LLC, New York, NY, 2012) was used to generate the three-dimensional structure. The correct chirality of the compound was assessed and the ionization states at pH 7 ± 2 were calculated. The twenty-five deposited FXR co-crystal structures were downloaded from RCSB Protein Data Bank (Berman et al., 2000) and the co-crystalized ligands were used to perform a Phase Shape Screening (Phase, version 3.4, Schrödinger, LLC, New York, NY, 2012) against MF. Phase Shape Screening was run in a pharmacophore-based mode indicating OCA (Pellicciari et al., 2002) as the most similar compound to MF with respect to the shape. The co-crystal structure of FXR in complex with OCA (pdb code: 1OSV) (Mi et al., 2003) was therefore selected for docking studies and submitted to the Protein Preparation Wizard workflow (Maestro, version 9.3, Schrödinger, LLC, New York, NY, 2012). The receptor grid was then calculated using Glide (Glide, version 5.8, Schrödinger, LLC, New York, NY, 2012) software. The centroid of the co-crystalized ligand was taken as the center of the grid and the docking space was set as 29 Å cubic box. MF was then docked in a stepwise manner: MF was flexibly docked into the prepared grid using the Glide Standard Precision (SP) algorithm. Ten poses were collected and subsequently refined with the more accurate Extra Precision (XP) algorithm. The retrieved docking solutions were then clustered into three distinct binding modes (clustering criterion: root-mean-square deviation (RMSD) <2 Å). Three poses representative for the three predicted binding modes, were selected relying on the best Glide XP Gscore and submitted to molecular dynamic simulations using Desmond (Desmond Molecular Dynamics System, version 3.6, D. E. Shaw Research, New York, NY, 2013, Maestro-Desmond Interoperability Tools, version 3.6, Schrödinger, New York, NY, 2013). The systems were built by solvating each complex with SPC water solvent and neutralized by adding sodium ions. The periodic boundary conditions have been set defining a 10 Å width orthorhombic simulation box. The temperature of the system was set to 300 K. The simulations were performed in NPT ensemble using Nose-Hoover chain thermostat and Martyna-Tobias-Klein barostat. The force field used was OPLS2005. Systems were relaxed first and subsequently submitted to 100 ns of trajectory production. Molecular dynamics trajectories were analyzed with the Simulation Event Analysis and the Simulation Interaction Diagram implemented in Desmond. Average RMSD and interaction energies were calculated excluding the first 10 ns of stabilization of the systems. Figures were prepared using PyMOL (http://www.pymol.org).

ChIP-qPCR

To determine p65 recruitment to target promoters, HepG2-GFP and HepG2-GFP-FXR cells were treated with vehicle (DMSO), 1 µM GW4064, 10 µM MF or 5 ng/ml TNFα, as indicated for 4 hours. ChIP was performed according to Saccani et al. (2002) with a two-step crosslinking using DSG and formaldehyde, as described in Nowak et al. (2005). qPCR was performed as described above. Data are presented as fold change relative to DMSO. Primers are listed in Table 2.

Data analysis

GraphPad Prism software version 6.02 (GraphPad Software, Inc.) was used for figure preparation, determining EC50 and IC50 values, and statistical analysis. Unpaired t-test, two-tailed, was used for Luciferase assay. For Luciferase assay and qRT-PCR results, each bar represents mean ± SD, or mean ± SEM (organoids). P-values ≤ 0.05 were considered statistically significant.

Additional Information

How to cite this article: Bijsmans, I. T. G. W. *et al.* The glucocorticoid mometasone furoate is a novel FXR ligand that decreases inflammatory but not metabolic gene expression. *Sci. Rep.* 5, 14086; doi: 10.1038/srep14086 (2015).

Author information

Author notes

Antimo Gioiello & Saskia W. C. van Mil
These authors contributed equally to this work.

Affiliations

Center for Molecular Medicine, UMC Utrecht, Utrecht, the Netherlands
Ingrid T. G. W. Bijsmans, José M. Ramos Pittol, Alexandra Milona & Saskia W. C. van Mil

TES Pharma, Loc. Taverne, Corciano (Perugia), Italy
Chiara Guercini & Roberto Pellicciari

Cell Screening Core, Department of Cell Biology, UMC Utrecht, Utrecht, the Netherlands
Wienand Omta, Daphne Lelieveld & David A. Egan

Department of Pharmaceutical Sciences, University of Perugia, Perugia, Italy
Antimo Gioiello

Contributions

I.T.G.W.B., J.M.R.P., A.M., D.A.E., R.P., A.G. and S.W.C.v.M. designed the study. I.T.G.W.B., J.M.R.P., C.G. and D.L. performed the experiments. I.T.G.W.B., J.M.R.P., C.G. and W.O. analyzed the data. I.T.G.W.B., C.G., A.G. and S.W.C.v.M. wrote the manuscript.

Competing interests

The authors declare no competing financial interests.

Corresponding author

Correspondence to Saskia W. C. van Mil.

Chapter 8 - General Discussion

Conclusion

The incentive for conducting this research has been described in the introduction of this thesis. The domains of High Content Screening (HCS), Biology, Bioinformatics, Robotics & Automation, Information & Computer Sciences and Data Science are intertwined to accelerate the data-to-knowledge process for HCS. The overarching objective of this research is to design a data analysis system that improves the high content screening workflow for uncovering new biomedical knowledge, i.e. identifying phenotypes that are related to the curation of diseases and the fundamental understanding of biology using HCS. The aim of this research is formalized in the following research question:

MRQ: How can multi-parametric data analysis contribute to effective knowledge discovery in High Content Screening?

For the guidance and support of this research, and to provide an answer to this question, this research was embedded within the design science research framework of Hevner et al (2004), as illustrated in Figure 11 within Section 1.4. The research cycle endeavors to design an artifact. First, a high content screening Knowledge Discovery Process was designed and implemented into HC StratoMineR, a platform for the analysis of high content data (Chapter 3). Also, PurifyR, an artifact for the automatic preprocessing of numeric data, was developed (Chapter 6). For the development of these artifacts, a variety of evaluation methods were used. Individual screening facilities were explored in detail using case studies. Expert interviews were conducted for in-depth questions and exploratory information gathering, which was documented and modeled using a Method Engineering approach (Chapter 2). Prototyping was used to develop an artifact that could be used by screeners (Chapter 3). Extensive testing of the artifact was carried out by multiple potential end-users in order to improve the system iteratively. Computational experimentation was executed for the optimization and measurement of the performance (Chapters 3, 4 and 6). Validation studies were conducted to verify the produced results by the artifact (Chapter 3, 4 and 7). Internal validation was conducted by aligning the results with the original microscopy images (Chapter 3). Also, validation using external sources was conducted to verify the produced results, i.e. published papers and external ontologies (Chapter 3 & 4). Finally, experimentation was conducted and combined with the prototyping of an artifact to measure differences in using interactive and non-interactive visualizations (Chapter 5).

The implementations are relevant in the field of high content screening and are based on scientific publications in the domain of high content screening and data analytics. Verification of the artifacts are conducted in the form of peer-reviewed articles, mainly in the field of bioinformatics. Supplementary data is packaged together with the articles when applicable and allows the experiments to be repeated and maximize the transparency and reproducibility. Supplementary data can include a SOP, data, additional results, visualizations, scripts, and links.

Summary of Findings

RQ1: What is an effective information architecture for High Throughput Screening?

The first guideline in the Design Science Research Framework is to design a viable artifact such as a model or a method. First, a case study was designed with a focus on the quality of the Information Architecture (IA) used within a Screening Facility. A qualitative analysis was performed using semi-structured interviews. The interviews revealed information such as the variety of used software packages, the compatibility of the software and software packages that were specifically used for data analysis. This gave insight into the actual need of additional software solutions and which software is lacking in functionality, accessibility, speed or ease of use.

In order to design principles, rules of thumb and guidelines for the development of an artifact that will contribute to effective knowledge discovery in multi-parametric data analysis, the entire process of High Throughput Screening (HTS) was modeled which was derived from the interviews. This started at hypothesis creation and ended with reporting results and conclusions of the experiment. This was modeled using Process Deliverable Diagrams (PDD) in order to understand the process of High Throughput Screening at a conceptual level.

Conducting the interviews revealed that there are great differences in important aspects between screening facilities. Image storage and quantification seem to be the most mature well-developed processes. The microscope is usually equipped with hardware and software supporting image storage and quantification. Reagents can be managed by Advanced Library Information Management Systems (LibIMS) whereas other Screening Facilities are working with flat files or Excel sheets. No Screening Facility used a Laboratory Information Management System (LIMS) that connected the image database, the reagent management, the raw data, the data analysis and project information such as assay protocols and screeners. Data analysis was carried out using an open source platform called CellHTS2 or R scripts. This means that the data analyses performed in these facilities are either all scripted separately by hand for each project or analyzed by means of a platform that does not allow for multi-parametric data analysis. R is a statistical programming language that requires skilled personnel for writing scripts. The data analysis platform, that was used in screening facilities, offers its results in HTML format, and does not allow for downloading the data in a structured manner. It became apparent that most facilities are coping to implement open source software for image storage, quantification, reagent management, screening management and data analysis in one information architecture such as BriskSLIMS, Screensaver, Omero, Cellprofiler and WebCellHTS2.

In conclusion, there is a high demand for a statistical software package that allows and supports the ability to handle and store structured data in order to easily manipulate, visualize and export data. Then end-users can easily enrich results with external databases and ontologies. There is also a need for a comprehensive software solution in academia and small to medium-sized commercial organizations. Most available software packages are designed to manage clinical samples. Current academic and small to medium-sized organizations lack much of the required functionality in their information architecture for effective screening libraries at a larger scale.

RQ2a: What are the required workflow and software components for analyzing HCS data?

Easy to use data mining software that allows for the analysis of multi-parameter HCS data, is not yet available. Web CellHTS2 is a web-based software tool that allows screeners to analyze their data, but it lacks multi-parameter functionality. Therefore, a web-based software package called HC StratoMineR was designed and built. The development was inspired by Web CellHTS2, the multi-parameter approach of Young et al. (2008) and the KDD framework. Web CellHTS2 was an inspiration because of its linear workflow and web-based accessibility. The multi-parameter approach of Young et al. (2008) contains a proven way of working with multi-parameter data based on dimensionality reduction methods but lacks in data preprocessing, which is complemented by the KDD framework.

The workflow is built in a linear fashion because most steps are dependent on previous steps. First, the raw data needs to be uploaded to HC StratoMineR, where the number of plates, replicates and the file format can be configured. Meta settings such as meta variables, a plateID, the data resolution and the well location column need to be defined. Then, data preprocessing can be applied in order to eliminate features that show significant missing data, a standard deviation of 0 or a Pearson correlation coefficient of 1 with another feature in the data set. Next, feature selection is executed including suggestions to eliminate features with uniform distributions regardless of the class. The data is organized in potential reagent classes that can be defined by the user in the plate map configuration. Subsequently, Quality Control (QC) can be conducted in order to verify the quality of the controls (reagent classes), plates and features. Data can be normalized in a plate-by-plate fashion such as a Sample Median or B-score to correct for plate and batch effects. Data transformation is performed on features that show a significant skewness in order to assume multivariate normality. Data can be scaled in such a way that all used features operate in the same range to avoid biased results. Missing data is handled by a simple median-column imputation or more sophisticated multiple-imputation techniques to avoid empty fields in the final data-matrix. Subsequently, the data is ready for dimensionality reduction such as PCA or factor analysis in order to decrease the redundancy of the data and save computational power. Hit selection or hit picking can now be performed to calculate the reagents (hits) that significantly differ from a defined control or reagent class. Finally, the identified hits can be organized in groups of similar phenotypes and mechanisms of action using clustering techniques.

In order to implement the above described workflow in a software package, the following software components were employed. It was decided to create a web-based platform to increase the accessibility. Therefore PHP, a server-sided web-language was selected for easy prototyping. For the storage of meta-data and large numerical data sets, MySQL was chosen due to its tight integration with PHP. The standard front-end output in this type of web-based application is HTML, enriched with flexible JQuery scripts, packaged in the powerful framework called Bootstrap. For handling numeric data, dealing with the logic, applying data mining techniques and visualizing the data, R was selected. This was chosen because of the large active online community supporting R and the powerful R package ggplot2. To increase the performance of the software package, parts of the preprocessing and variable selection phases were running in C++ compiled code using the Rcpp package to efficiently compute large vectors. Other parts of the clustering phase required the calculation of distance matrices that were computed using a package called rpud, which in turn initiates CUDA to perform these calculations using GPU acceleration. Almost throughout the complete workflow a large number of processes iterate over plates or features across the data set, which allow for "*embarrassingly parallel computing*" to speed up the process. Processing can either be distributed over multiple cores on a local machine or one can decide to submit a large screen to a High-Performance Computing (HPC) cluster to extend the power of the local machine running the software application. Finally, the software package components for analyzing HCS data require a secure connection to encrypt the data in a client-server architecture. Therefore, Apache has been used to run the web server and to secure the connection using a 256-bit ssl encryption.

RQ2b: What are the implications of single- vs. multi-parametric data analysis of HCS data?

In HCS, numerous features of cellular morphology can be extracted from multidimensional images (channels), captured using automated microscopy. These features are statistical measurements and together they can show a multidimensional phenotypic profile for each treated and untreated cell, which constitutes a much richer profile than a single feature. In most single parameter approaches, an intensity feature is chosen which basically measures the amount of fluorescent light. In multi-parameter approaches, shape, size, area and texture features in addition to intensity features can be included in the analysis. Furthermore, restricting the data analysis to include one feature can also reduce the focus to a single channel, e.g. only Hoechst (to label DNA) or only Lysotracker (to label lysosomes).

However, multi-parameter methods require more bioinformatics and data analytics skills and challenges to overcome, such as the preprocessing stage which includes the handling of multicollinearity, singularity, missing data and dimensionality reduction techniques such as factor analysis, especially when using single cell data. Nevertheless, these methods allow the researcher to use all potentially useful data and combine it using factor analysis, ruling out redundancy, exploring the data in an unbiased fashion and making use of powerful multidimensional methods such as hierarchical cluster analysis and visualizing it in rich heatmaps. Therefore, the potential reward for employing multi-parameter methods can significantly outweigh the additional complexities. In addition, performing multi-parameter data analysis allows the biologist to explore and discover beyond the low-hanging fruit.

In conclusion, single parameter analysis limits the user in the amount of data and detail that can be included. It is a biased way of performing data analysis because one needs to decide which feature(s) to include. Factor analysis is a very powerful technique to explore multi-parametric HCS data because a correlation matrix and the factor loadings provide the necessary information to correlation, redundancy, noise, and biological relevance. The new latent features (factors) that can be extracted from factor analysis allow the biologist to replace all the original measured features when factor scores are computed and extracted from the analysis. The latent features as a replacement of the original feature space decreases redundancy and bias in the results of the analysis. Also, using the latent features instead of the original features saves computational power due to the reduction of dimensions being processed.

The inability to analyze complex multi-parameter (multivariate) data sets is often due to the lack of access to the appropriate bioinformatics and biostatistics skills, tools, and hardware required to process complex data sets. If a researcher must work with a bioinformatician to develop scripts for the analysis of data, there are invariably delays, as the biologist needs to educate the bioinformatician about the biological problem. This slows down the iterations within data analyses process. The result is that fewer strategies for data analysis are tested. In the case of the genome-wide data set, we presented HC StratoMineR and reduced the analysis lead time from months to hours.

The ability to identify more phenotypic hits and then use the clustering functionality to identify clusters that are enriched for hits of interest will potentially allow screeners to get past the “low hanging fruit” problem and identify weaker hits that are nevertheless functionally relevant.

RQ3: How can supervised learning approaches contribute to the analysis of HCS data?

Chapter four demonstrates how combining unsupervised and supervised machine learning can greatly enhance the efficiency with which new knowledge can be extracted from functional genomics screens. Our original analysis, relying solely on unsupervised methods resulted in a hitlist that was overwhelmed with hits that were of little interest since they were already known to be involved in the core machinery of protein translation, degradation and RNA splicing. It might have been possible to find interesting hits with novel mechanisms of action, but these would have been difficult to identify.

During the last few years, there has been an increase of interest in machine learning for the analysis of image-based cellular screens. The accuracy of the analysis is heavily dependent on the quality of the data and the annotation of labeled data. We used exploratory methods to gain a better insight into the quality of the data and selected representative data that we used for labelling four robust phenotypes. This data was subsequently used for training a random forest classification model to distinguish the four phenotypes with an accuracy of 91.1% and a kappa of 0.85 (four-fold cross-validation).

This approach did prove to be very useful to identify hits. The classification model allowed us to generate hit lists that were far more enriched in genes that were centrally involved in mitosis compared to the possible outcome with unsupervised methods alone.

We generated four hit lists using the p-values .05, .005, .0005 and .00005 respectively and enriched these in the protein-protein interactions database String-DB. The number of hits decreased whereas the level of certainty that the genes in this hitlist involved in mitosis, increased. Thus, this analysis process allows for selecting hits more successfully for a follow-up study or secondary screen which aims to show the highest probability of being involved in mitosis. Thus, increasing the chance of finding novel hits.

RQ4: What are the effects of using interactive data visualizations in HCS data analysis?

In HCS, many features can be measured resulting in a high-dimensional data set. HCS data is often heterogeneous and can be difficult to interpret. That makes mining biological high-throughput data harder. The amount of data that someone can interpret is very limited when data is presented in a textual or tabular fashion. Visualizing data is a way to extend this as a very powerful way for humans to detect activity, patterns, and data artefacts. Visual Data Mining (VDM) concentrates on the integration of the data exploration process by presenting the data in a visual form directly to the user. VDM takes the flexibility and creativity of a human into account by using its cognitive advantage. VDM is described in four steps: (i) overview, (ii) zoom, (iii) filter and (iv) details on demand. Combining VDM with the ability to explore data by direct interaction with the visualization, also called direct manipulation is an interactive visualization. Interactive visualizations allow for visualizing multiple dimensions simultaneously (i) (dynamic projection), dividing the data into subsets (ii) (filtering), magnifying interesting parts (iii) (zooming), focusing on a specific section while preserving the complete overview (iv) (distortion), linking visualizations to other visualization objects (v) (linking) and highlighting sections of data (vi) (brushing).

An experiment was set up to measure the accuracy and efficiency of using interactive visualizations. Therefore, a web-based platform was developed using

Django, D3 and Angular of which data can be visualized using various visualizations, e.g. histograms, line plots, scatterplots, heatmaps and polar plots. HCS data was used for the experiment in which a set of tasks was designed to be performed.

A total of 79 computer science, information science, and bioinformatics students participated in the experiment. Students were randomly assigned to the interactive condition (the condition with the interactive visualizations) or the non-interactive condition (the condition with only non-interactive visualizations).

The accuracy was measured by the number of correct answers per participant. The efficiency was measured by the time in seconds it took to complete all twelve assignments. We can conclude that using interactive visualizations increase the comprehensibility compared to using non-interactive visualizations. Also, the accuracy increases when using interactive visualizations over non-interactive visualizations. The results showed that the interactive group presented a significant lower number of wrong answers compared to the non-interactive group. Also, the interactive group performed significantly faster compared to the non-interactive group.

RQ5: How can preprocessing in numeric data analysis be automated?

Data preprocessing is one of the most time-consuming and tedious data analysis tasks which can consume up to 80% of the time of a data scientist for analyzing large data sets. This work is devoted to repetitive tasks. Preprocessing consists of removing undesired outliers, systematic errors, dealing with missing data or handling assumptions such as linearity, normality, homoscedasticity, multicollinearity and singularity.

A workflow of actions organized in modules within an R package called PurifyR was developed. The first module is called ScanR that scans and fixes the very basics such as duplicate rows and columns and makes sure that all columns have unique names. It analyzes the amount of missing data, extreme outliers, distribution types, highly correlated variables and data types. The next module is called ScrubR and removes highly correlated variables, transforms highly skewed variables, scales the variables and solves missing data by the selected method, e.g. case-wise deletion or median imputation. Finally, it detects extreme outliers based on a set cut-off of standard deviations from the mean. The last module is called SmashR. In this module, the user needs to assign one or more variables that define the unit of analysis. The unit of observation is assumed to be a single record in the data set. Then, for each unit of analysis, a series of estimators such as mean, median, standard deviation and standard error are being calculated. This aggregation allows the user to easily create a visualization that is easy to interpret or to use for a follow up analysis such as regression, classification, or dimensionality reduction.

The R package PurifyR, which implements these modules, has been made available through Github. Published defaults or rules of thumb were gathered and implemented as defaults that can be easily changed, visualized, and explored. Examples use a correlation cut-off of .99 by default and iteratively decrease the cut-off until the result is a non-singular matrix. Another example is using a log or square root transformation to transform features to meet the assumption of multivariate normality.

Nowadays, most available packages are implemented using either the R *base* or using the *dplyr* package. PurifyR was fully developed using the *data.table* approach which is a C compiled package that allows up to 20 times speed-up. Another PurifyR functionality to accelerate the data preparation process is PurifyR's support for parallel processing. Finally, the module ScanR can be executed based on a percentage e.g. 15% randomly selected data to detect the skewness, normality, distribution types and other properties. Because of its random nature, the results provide a very accurate estimation and a great speed-up.

<u>RQ6: To what extent can the HCS data analysis workflow also work for low content data?</u>

In this study, we used HC StratoMineR to identify drugs that activate the FXR mediated suppression of inflammation, but do not activate the expression of genes that are normally directly activated by FXR. A high-throughput Luciferase screen was conducted to monitor NF-κB activity to screen the Chemical Prestwick Library®, a library of 1200 FDA approved drugs. The screen contains four 384-well plates and was screened in triplicates (12 microplates in total). The ideal candidate suppresses NF-κB activity but does not induce SHP or IBABP transcriptional activity. The screen contains two important features that were exported using a microplate reader; (i) Renilla, that was measured for cytotoxicity and transfection efficiency whereas (ii) Luciferase measures the NF-κB activity. TNFα was implemented as a negative control and was used to normalize the data, i.e. all drugs were normalized to the average TNFα value of the corresponding microplate. Then, a feature RLU was calculated which represents the Renilla/Luciferase ratio. The detection of hits should ideally result in candidates that are NF-κB repressors and simultaneously do not show interference with its metabolic capacity.

For hit picking, the Manhattan distance score for each well was calculated against the average score of the TNFα values in a plate-by-plate manner. The distance scores were used for the calculation of p-values of which a cut-off of p<.05 was set. The resulting 34 screened hits showed a TNFα-induced transcriptional activity of the NF-κB reporter. Then, drugs that showed low Renilla values were excluded because low values imply cytotoxicity. This finally resulted in a list of five candidates that reduces NF-κB transcriptional activity significantly.

From the 1200 drugs that were screened, five candidates were identified, and three candidates showed FXR-dependent behavior after quadruplicate dose-response curve experiments. The results of these candidates were confirmed by a recent study of Hsu et al. (2014). Therefore, we can conclude that using the HCS data analysis workflow that is designed for multi-parameter high content data sets, also works for this screen using low content data.

In this workflow, the phases transformation, feature scaling, missing data, dimensionality reduction and clustering are omitted because they are not applicable to low content data. To elaborate, transformation in multivariate analysis is introduced to assume approximate multivariate normality, recommended for a data analysis step such as factor analysis which is depending on a linear model. When processing one or two features, the transformation phase becomes unnecessary. Feature scaling is introduced to eliminate extreme variance in the range of the feature set. This becomes relevant when using more than one feature. There is only one valid option when dealing with missing data for a data set that contains one feature, omitting the data that is missing, i.e. case-wise deletion. Dimensionality reduction is simply not an option using one feature. Clustering one feature becomes redundant to hit picking techniques that are applied to the data which results in a ranked list of importance.

Contributions and Implications

Scientific Contributions

As shown in the Design Science Research Framework in Section 1.5, Figure 12, this thesis contributes to two main theories and methods in the knowledge base: An Information Architecture for HCS and a Data Analysis Process for HCS. In addition, this research contributes to two implemented artifacts. Therefore, this section first describes the contributed theories and methods, followed by the implemented artifacts.

Information Architecture

The information architecture describes an abstract view of the HTS process which starts at hypothesis creation and ends with reporting and publishing a study as a publication. The abstract view has been documented as a Process Deliverable Diagram (PDD) in order to understand the HTS process that includes reagent management, screening management, image analysis/quantification, feature selection, data analysis, data visualization, data enrichment and reporting. The PDD provides insight into the variety and complexity of the HTS process and therefore the need for fit-for purpose software solutions.

HCS Knowledge Discovery Process

The knowledge discovery process for HCS data analysis (or "HCS-tailored Data Analysis Process") which is presented in this thesis, consists of the following workflow phases: Raw data, Meta Data & Extraction, Transformation and Load (ETL), Feature Selection, Quality Control, Data Normalization, Data Transformation Feature Scaling, Missing Data, Dimensionality Reduction, Hit Selection and Clustering. The Data Analysis Process (DAP) starts with Raw Data, which is uploaded by the user. Then meta-data columns are identified that provide crucial information such as a well location, a plate id, the name of a chemical or the concentration of a chemical. The remaining phases are more data oriented.

At first, data preparation is carried out, also called ETL. The data format is prepared, in order to assure the data structure and compatibility. This is dependent on the computer language that is used. In this research, the back end of the developed artifacts is using R and therefore the data handling is conducted using R. This step involves checking and correcting for valid file names, valid column names, consistency in data types, removing empty lines, pairing replicates, adding indexes, and storing the data in an efficient way in order to query subsets later.

Further downstream, processes are executed that involve checking and correcting the data to assure the required assumptions are met for the next data analysis processes: dimensionality reduction, hit picking and clustering. The basic assumptions and requirements in preprocessing the data are checking and eliminating non-discrete variables, empty variables, no variation, extreme high covariation and singularity. Then, feature selection process should be carried out by visual inspection of features using scatter plots and histograms. Plate-by-plate inspections are carried out to the same requirements such as no variation and empty variables. Features showing a uniform distribution among all classes are suggested for removal.

Quality Control (QC) is extremely important within HCS and can be carried out for each screen, replicate, plate, and feature and well (figure 1). The first important aspect is; does the screen work properly and do controls show the expected results. First at plate level, the Z-prime and Strictly Standardized Mean Difference (SSMD) metrics can be calculated, represented in scores that provide information about the robustness of the screen. Visualizations are very important to detect patterns such as systematic errors, batch effects, inconsistency of replicates, extreme outliers, and plate or batch effects (figure 1). A researcher can decide to repeat (a part of) the experiment when the data is showing dissatisfactory results.

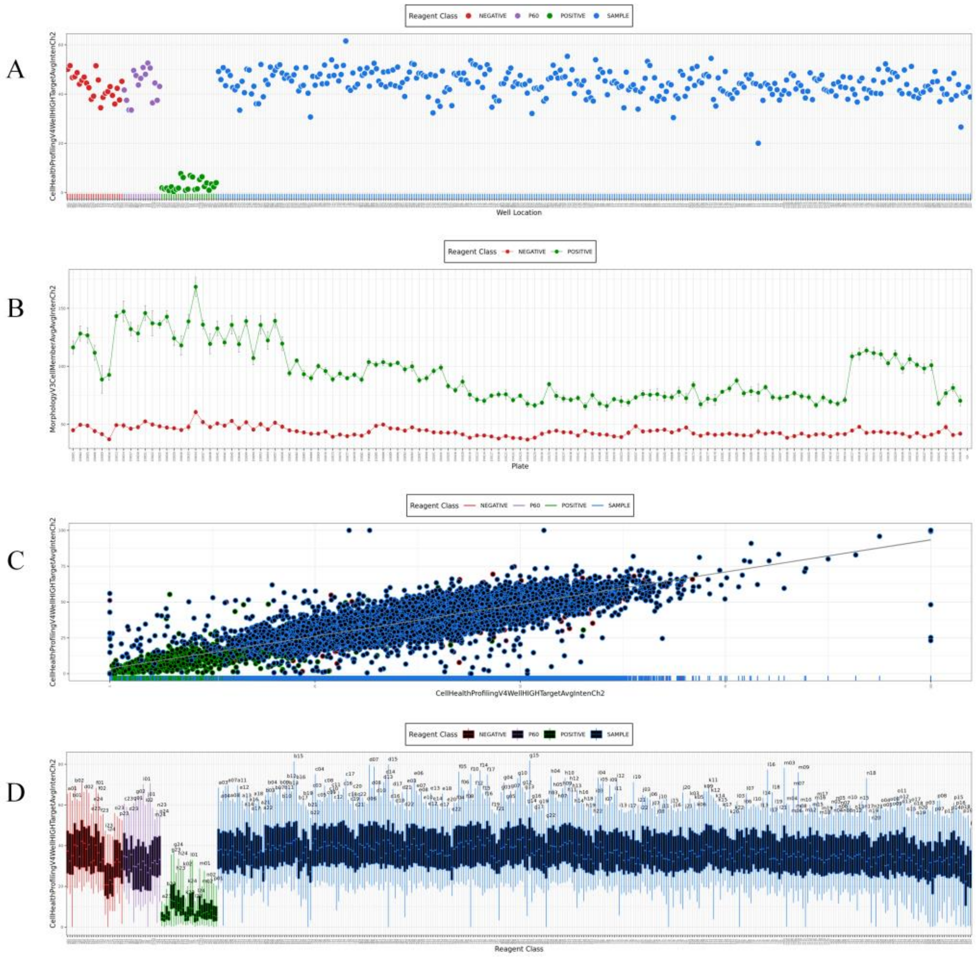


Figure 1. *Quality Control*

1A *represents QC at plate level of which the wells of the positive control (in green) and the wells of the negative control (in red) are nicely separated.*

1B *represents QC at the control level for the whole screen. Here each x-axis position represents a microplate and each dot in the visualization the median of all negative or positive controls of a plate, visualized in red and green, respectively.*

1C *represents QC at the replicate level of which the consistency of the replicates is shown.*

1D *represents QC for checking systematic errors. Each x-axis position represents a well position of a microplate. Each boxplot represents all measurements of a well position of the whole screen. When a boxplot shows a significant change compared to other boxes of the same class, this might indicate that there is a systematic change in that well position throughout the whole screen.*

Data normalization is necessary to allow the comparison of data from one plate to another plate, even when they represent technical replicates. In other words, data normalization aims to rule out plate-to-plate variation. The QC phase can provide input to change the normalization strategy to use an algorithm that corrects the data for plate effects such as a B-Score algorithm that applies a row and column polish.

A transformation phase is mainly focused on the assumption of multivariate normality. This is required for linear methods that rely on the gaussian distribution and will perform better when the assumption of multivariate normality is not violated. Usually the Kolmogorov–Smirnov or Shapiro-Wilk test are limited or too sensitive. Therefore, this is measured by the skewness of each variable.

Feature scaling is like data normalization but is now focused on features, not microplates. Features can be measured in various ranges and therefore, can differ in the impact of the result of distance-based algorithms such as the Euclidean Distance. In order to solve this problem, features are scaled to z-scores or other scaling methods of which all features have e.g. a similar mean and standard deviation which assures similar weights of the features at the input of an algorithm and supports an unbiased analysis approach.

Missing data is a problem that can be challenging in high content data, i.e. multiple features. The use of replicates within screening can be used to estimate a missing data point. When no valid replicate values are available, simple methods such as a column-wise median can be imputed. Other advanced multiple imputation methods such as regression use other features to create a model to predict the missing value. It is important that the missing completely at random (MCAR) assumption is met before applying these methods.

Dimensionality reduction has multiple purposes and one of them is exploration. Dimensionality reduction provides insight into covariation and noise. It can reduce the number of features to a smaller number of latent variables that can be used instead. This helps to reduce the required computational power and helps to overcome the curse of dimensionality. The most important effect of dimensionality reduction is that it helps to overcome a bias in the data analysis approach.

The goal in screening is to identify a set of reagents that initiate a specific phenotype. This process is carried out at the hit selection phase. We have studied two approaches: supervised and unsupervised hit selection. In supervised hit selection, labeled classes are used as input together with the numeric data. A learning algorithm such as random forest can train, learn, and distinguish the classes using the feature space and can classify new data into one of the classes that was used for training. The unsupervised approach is distance-based and calculates a multivariate distance from a reference point such as a negative control. In both the supervised and unsupervised approach, a distance or similarity measurement is calculated that can be used for setting a cut-off.

Clustering can be used after hit picking is performed. The items within the selected set using a cut-off can be included for clustering. A disadvantage of a distance-based unsupervised hit selection approach is that no information is given in which direction a reagent differs from its reference point. Therefore, clustering generates groups of reagents showing similar phenotypes based on the feature space. Hierarchical clustering and k-means clustering are combined to maximize the information given. Because these methods are not capable of defining the number of clusters to generate, it is combined with the gap-statistic, a method to automatically calculate the number of clusters.

Implemented artifacts

The implemented artifacts, described in this dissertation (figure 12, section 1.5), are described here. HC StratoMineR is a web-based tool for the analysis of high content data. This tool is designed for web usage to increase the accessibility of the tool. The only requirements are a device with a proper internet connection and an up-to-date browser. The layer for securing the web application include a password protected area, an encrypted session and an SSL connection. The architecture used is a client/server approach that includes the usage of a high-performance computing cluster (HPC) to be able to accelerate computationally intensive processes.

Other ways to increase the performance of the artifact that are implemented are (i) multithreading, (ii) the usage of compiled C++ code and (iii) using GPU power. The user interface is based on the Bootstrap package that contains a set of elegant and useful widgets. The user is guided through a linear workflow in order to keep the analysis process simple. HC StratoMineR provides the user with recommendations and restrictions such as removing, selecting, and transforming features or selecting hits. All visualizations in the workflow are created using the ggplot2 package such as scatter plots, bar plots, line plots and heatmaps. HC StratoMineR is available at https://cla.stratominer.com/.

The other implemented artifact of this research is the PurifyR R package that was developed for the automated preprocessing of numeric data. In this package, defaults and rules of thumb are implemented that can be modified using attributes. The package is fully developed using the data.table approach and can use multithreading in order to maximize the performance. The PurifyR R package is available at https://github.com/womta/PurifyR.

Societal Implications

In daily practice, life scientists and health care specialists join forces in the domain of HCS to conduct biochemical experiments using microplates. These screens are conducted at research labs, research facility centers, biotech, NGO's, and pharma. There are five major vendors that can provide automated microscopes, including Thermo Scientific and Molecular Devices. These automated microscopes are usually equipped with image storage and image analysis software. Omero and CellProfiler are the most used available open source solutions. However, image analysis is still a very tedious and manual process that is error-prone and time consuming when extracting many features e.g. the Cell-Painting method. These challenges are in line with the two major problems in the field of HCS which are identified in Section 1.6 and are discussed below:

I How to derive and capture knowledge from HTS and HCS?
II How to deal with the wealth of data to capture knowledge from HCS data sets?

The first problem is how to derive and capture knowledge in the field of HTS and HCS. This thesis has extensively described how knowledge in the domain of HTS and HCS should be derived, with skilled personnel in multiple fields. This includes Biologists for the domain understanding, Chemists for the chemical understanding and engineering, automation specialists who understand the automation of biology, technicians that can help manage the project and bioinformaticians that understand the screening process and have sufficient knowledge of multivariate data analysis. Most of all, it is very important that the screener (the creator of the data) is involved in the data analysis process in order to align the knowledge discovery strategies with the data analysis strategies. The problem is frequently compounded by the fact that analysis methods are often not addressed until after the data set has been generated. Therefore, it is particularly important that the data scientist or bioinformatician is involved in the set up and the piloting stage of the screen. The data scientist can provide recommendations to the stakeholders at various stages that support consistent and high-quality data output such as plate map configurations, control positions and the implementation of replicates.

Many software packages support different stages of a screening project, such as Prism, a software package for concentration curves, R for data analysis, or CellProfiler for the image analysis. Frequently a non-fit for purpose software package is used such as Excel for data analysis or managing a database. Also, software packages that are incompatible, but require interoperability between different stages of the screening project, are often used. Therefore, careful package selection should be carried out before integrating it into a screening project. A final remark is that screening facilities should have their business processes very well documented. An example is the process of pooling, deconvolution, or oligo picks. These processes can be very complex and error-prone and would benefit from a well-documented knowledge base that includes flowcharts supported by text.

The second stated problem which still hinders HCS' potential social impact, focuses on how to properly capture knowledge from rich HCS data sets. This thesis has elaborately reported on how to capture knowledge efficiently and which tools are required, preferably a user-friendly, fast, and flexible information system. Also, the right infrastructure is required to handle the data efficiently, i.e. a High-Performance Computing Cluster or sufficient cloud computing for processing power, a network up to 10Gb/s and fast storage up to 100 TB.

For the analysis of the data, detailed Quality Control (QC) is required for checking systematic errors, plate effects, data artifacts, extreme outliers, and z-prime analysis. The use of visualizations is extremely important because of the patterns that can be easily recognized by human beings. The use of interactive visualizations allows for linking, filtering, brushing, zooming and distortion and therefore provides a better focus on the data space that people want to explore. An extra internal validation is the use of microscopic images and aligning them with the data analysis results. In this way, false positives, hits, or extreme outliers can be easily verified such as a florescent compound, a hair or moisture at the back of a microplate. Rich HCS data sets often contain missing data. It is important to investigate the frequency, the spread, and the reason that missing data occurs among the features that were extracted using image analysis.

There are many reasons why a field contains a NULL, NA, INF, -INF or NaN value, e.g. a human error, a systematic error, a threshold problem or simply because the calculation is impossible. After conducting a missing data analysis, a missing data report should be written in order to improve future feature extraction methods. A wealthy data set that contains tens to thousands of features benefits from dimensionality reduction techniques. Reducing the number of features (i) decreases the redundancy or overlap in features and therefore the analysis bias, (ii) reduces the noise in the data, (iii) helps to explore the data e.g. using factor loadings, (iv) it helps to overcome the curse of dimensionality in follow-up data analysis strategies e.g. classification and finally, (v) reduces the required computational power for carrying out data analysis. The result of a dimensionality reduction technique can be saved in e.g. factors or components that may be a replacement for the original features in the data. For the detection of hits, it is important that the controls that are involved in the hit selection method are robust. The robustness of controls can be detected easily using a z-prime, SSMD or other quality control metrics. Using unsupervised techniques is fast and easy to use. Supervised methods are more complex, data intensive, require labeled data, require sufficient data for every class and require data at higher resolution, e.g. field, cell or data captured at object level, possibly in combination with z-stacks, time-series or dose. Unsupervised techniques can be very useful to explore and annotate the data. Then, supervised techniques can be applied using the labeled data to focus or eliminate phenotypes of interest. It is important that various data analysis strategies are being used and explored. There is no data analysis strategy that fits all sizes.

The results can be enriched with external gene ontologies and biochemical libraries such as Uniprot, GOrilla, string-DB or NCBI. Data from these external sources can when combined, be highly informative for the elimination or focus of reagents in a follow-up or secondary screen. Finally, it is important to stress that as many people as possible can access the studies in this field. Therefore, it is recommended to publish in open access journals as much as possible and attach a detailed description of the workflow that was carried out supported with an SOP, the raw data, the results and the used data analysis scripts. These additional materials should be offered in the supplementary data of the publication to increase the transparency and reproducibility of the experiment.

Limitations

This dissertation contains six studies that are described in chapter two until chapter seven. In this section, several additional limitations are addressed that have not been raised in the respective chapters.

In chapter five, an experiment was conducted to collect data for measuring the accuracy and speed of two conditions: non-interactive visualizations and interactive visualizations. Participants were randomly distributed among the two conditions by a prototype that was developed to conduct assignments using non-interactive and interactive visualizations. A group of 33 students participated in one classroom. At the end of the experiment, it was noticed that there was a skewed distribution of the two conditions and that students were more motivated and interested in the interactive condition and therefore tried to login a few times until they had entered the interactive condition. A solution would be a random function that is based on the users IP address.

In Chapter three, a web-based software application was built for the analysis of HCS data. The web application included an option for running the most computationally intensive step using a high-performance computing (HPC) cluster or running it on a local server. The data storage was managed using MySQL on a local server that was running the web application. Submitting a large job to the HPC cluster speeds up the analysis because jobs can be spread over 124 cores simultaneously whereas the local server only has 32 cores. However, submitting the jobs to the HPC cluster still required the local server to perform because the HPC nodes query the required data that needs to be processed from the local server. This resulted in a situation in which 124 large queries were simultaneously carried out on the local server. Therefore, this only resulted in a speedup that was just twice as fast as completely running it locally. An improvement would be to save the required data as binaries on a bulk storage equipped with multiple network interfaces (NIC) to provide the necessary data throughput for simultaneous data transfer without latency or using a load balancer for distributing the jobs within the HPC.

Chapter three contains two validations studies. One of the data sets is a functional genomics screen that seeks for Mitotic hits. The image analysis that was carried out, prior to the data analysis, was based on a cut-off of nine fields. This is usually implemented to save time. Image analysis is carried out from well to well and per well, and it moves from one field to the next one. In this setting the well contains 16 fields but if the image analysis software was not able to find 50 cells in field nine, it stops and moves on to extract data from the next well. This might have implications on the outcome of the analysis. To verify the error that is introduced because of this, the image analysis should be repeated using the same protocol without the built-in limit. The resulting data should then be used again, applying the same data analysis strategy.

Future Work

For future work, this section highlights three research topics that are of high interest to improve data analysis strategies, data quality and performance.

The first aspect of great importance on the data quality is the way in which image analysis is carried out. This is a very complex and tedious process; however, it also comes with great subjectivity of the biologist. He or she determines for example, what defines the background and what are valid nuclei based on thresholds that need to be parameterized. This could lead to false positive results and could be conducted by defaults, standards or at least by rules of thumb which are currently lacking. An opportunity would be to train a machine learning model that supports the search for the right hyperparameter settings e.g. a model based on thousands of examples in which the right balance for background-to-noise ratio is labeled by experts.

Most of the research within this dissertation is focused and uses linear methods except for chapter four. Therefore, it is useful to investigate the quality of the outcome in which linear and non-linear or non-parametric methods are compared. Next to that, there are other important factors why one would prefer a linear over a non-linear method e.g. speed, complexity (i.e. the number of attributes) and trustworthiness. This can be studied using an experiment in which participants carry out multiple data sets using different methods in order to measure the speed, complexity and trustworthiness.

A third research topic that deserves attention in future work, is the knowhow for building a cloud architecture including a batch submission system that conducts the analysis of data sets in the life science field. Various OMICS technologies share similar problems, i.e. large data sets that first require ETL towards a structured data set. Then, large chunks of data that need to be processed through a pipeline of data analysis steps can run efficiently using specific hardware requirements. Requirements can include GPU power, high memory, fast storage, multithreading, and a fast network connection. These requirements can support creating aggregated data sets, reports, or dashboards efficiently. However, it would be very expensive and inefficient to have racks of powerful servers running that meet all these specifications. Therefore, a cloud architecture that is capable to initiate one or more nodes for running data analysis jobs that are aligned with the recommended hardware specifications would greatly enhance the cost-performance trade-off.

Reflection & Valorization

This section is devoted to the process of acquiring a doctorate degree in the field of Bioinformatics. It starts in 2012 where I was dedicated full-time on the improvement of the data analysis processes in High Content Screening. These are the findings and recommendations for future PhD students, and specifically for those that are operating in a technical and multi-disciplinary field.

Domain Knowledge

It is incredibly important to understand all related processes in the domain. To collect all the required information and to understand a specialized domain, interviews were conducted, and a lot of documents were read. In addition, I investigated multiple layers within the domain and the organization (UMC Utrecht), including other screening facilities. Within these organizations, I investigated processes, habits, ways of working, information flows, enterprise software, and hardware architectures. This information was extremely useful to get a feeling for the need in a specific context for a specific audience in a specialized domain.

AS-IS and TO-BE

Once I had the basic understanding of what people were doing and what artifacts were used to make this easier, I had an idea how this could be improved. Semi-structured interviews were conducted for in-depth knowledge on how high-throughput screening could be improved. Afterwards, I started to model AS-IS and TO-BE situations and validated these iteratively using expert-opinions. The TO-BE situations were explored using scripting languages starting with simple statistics.

Building a Prototype

At a certain point, loose scripts required a different approach. This enforced decision making for the required programming languages, frameworks, platforms, and hardware, which I consulted by experts in the field. This has changed over time and is dependent on quality attributes such as security, stability speed or intensity of usage. In this regard, it was important to receive sufficient feedback from real users, and moreover, inform them about the development of the prototype. Potential end-users were able to use and implement the prototype in their production environment. Some users were completely dependent on the prototype. In general, prototypes should be frequently presented at meetings and conferences in order to receive tips and tricks to improve the architecture, back-end or front-end. It is especially useful to receive feedback from various technical people as well as end users with lower computer literacy. From this point on, it might be a good idea to separate the people from these two groups into separate meetings.

During the development of the prototype, I have supervised many students. Most students come from a computer science or bioinformatics background and were able to carry out small-to-medium sized tasks. This was very useful for the students, as they were learning while doing. Furthermore, it was also beneficial for the project and for me personally, learning to guide these students through supervision. The student projects were full-time and very intense, took place at the department and in the same room where I was. This in order to increase the potential for learning, help and discussions.

Teaching

During my PhD project, I was appointed as lecturer and thought courses at bachelor and master level for 18 months, full-time. Most courses were relevant to my project such as statistics, business intelligence, data analytics and advanced research methods. Although it was difficult at the start, I found out that teaching is an important skill that every PhD student should master. At a certain point, it gave me confidence and pleasure to transfer knowledge to a large group of students, sometimes as large as 220! I found it lovely to see that students are learning and developing their skill set. It is important and sometimes challenging to keep their attention and create content without introducing an information-overload. To retain their attention, I have used many real-world examples and included rich visualizations that tell stories.

Publications, publications, publications...

There is an excessive focus on publishing articles in academia. In the area of Bioinformatics and Biomedical Genetics, the area of my PhD studies, articles are mainly published in peer-reviewed journals. The area of computer science also allows for publishing book chapters and conference papers. Recently, the UMC Utrecht published an online article about the number of papers that should be accepted before a graduate student can finish his/her PhD. I believe, in academia we should prioritize quality over quantity, which is in line with the article. This because it requires quite some time, and in our field, sometimes years, to conduct well-designed research, experiments, analyze the collected empirical data and write the article.

Bureaucracy

The academic hospital and university are both large organizations with thousands of employees. To achieve something, one needs patience and perseverance. It took quite a while, before the required hardware was ordered to test and validate the prototype we built. Although the department that I was working for was a research department, there still were many regulations applied in the research side that were originally designed for the clinical side, causing significant delays. For future projects, it might be beneficial to consider the applicability of these regulations towards the research departments.

Meetings and Conferences

In my opinion, it is wise to attend a conference once or twice a year. You can meet new people, discuss, and present your work. Keep in mind however, it is important to attend the right conferences with the right focus.
For the last decade, I have attended many meetings, especially in the first years of my PhD project. I believe, meetings are very likely to have a low potential in knowledge transfer, and they are therefore often inefficient, especially when things are discussed that do not concern most of the audience. Therefore, a PhD student, should be very selective what meetings he/she should attend.

Networking

At the start of a PhD project, the people you probably know best are your colleagues, your daily supervisor, and/or your promotor/professor. During the course of your project, you will be confronted with challenges where potentially other people can provide very useful feedback. I have consulted many experts from many different fields and disciplines. I believe this is joyful, because you can share your vision and opinion about the project, have in-depth discussions and receive extremely valuable input to improve your prototype, your paper in progress or your design of experiment.

From Prototype to Commercial Product

Developing the prototype (StratoMineR) became at some point addictive. I really enjoyed adding features, improving the data processing efficiency and making the prototype faster. This supported the valorization phase of the artifact, but also helped us to carry out real-world analyses to test and validate the prototype and transform it into a more mature product. The best feeling in the world, in my opinion is to receive compliments for the product/prototype that you built. To commercialize the product, we had to close a deal with the organization that manages the intellectual property of the university. This was a tedious process that took a long time. It is extremely important what the term-sheet for such a deal includes, to protect yourself, your co-founder(s), and your own business. I am proud of the deal we have agreed upon together with the university. I would like to specifically mention all the efforts that were made by dr. David Egan in this regard.

As a result of all the efforts over the years, the business is now in operation for almost four years, with multiple top-pharmaceutical companies (including: Galapagos, Pfizer, Janssen Pharmaceuticals and AstraZeneca) as clients and a bright future ahead.

Summary

English Summary

This thesis investigates the data analysis process and techniques in High Content Screening (HCS). HCS technology is a subset of High Throughput Screening (HTS) that applies robotics for screening biochemical experiments in a semi-automated manner. HTS combines the use of robotics, automated liquid handling and microplates. HTS is used in functional genomics, screening drugs and hit to lead drug discovery. In HTS, large libraries of reagents can be screened against morphological assays using fluorescent proteins, chemicals, or antibodies. An important aspect of HTS technology is that numeric data can be extracted per well that reflects the biochemical activity.

HCS is a screening technology where images are acquired using automated microscopy of microplates. Traditional HTS does not support image acquisition nor image analysis. HCS involves a combination of robotics, automated fluorescence microscopy, liquid handling, image acquisition, image analysis and data analysis. Multiparametric data in HCS is calculated using image analysis and stored in external numeric matrices. These data sets are a richer phenotypic reflection of the biochemical activity inside a well compared to traditional HTS methods. The measurements are calculated using fluorescent dyes that bind to proteins in the cell for labelling cellular structures or substances such as DNA. Example measurements such as intensity, area, shape and texture features can be calculated from each fluorescent label and can result into data sets containing thousands of features that can provide a rich phenotypic profile for each individual cell. The data allows biologists to analyze and perceive the phenotypic effect of adding small chemicals to a well. The HCS technology generates a large amount of data that is rich in features and describes the profile of the resulting phenotype of the cells and provide insight into mechanisms of action. HCS results can be used to identify hits or outliers. HCS has been used in various disciplines such as drug discovery, toxicology, and functional genomics e.g. RNAi. Knowledge gained from these screens can be input for pre-clinical and clinical trials for the development of future drugs in Pharma.

The first focus of this thesis is the higher-level workflow of HCS. It is noticeable that most of the screening facilities do not use the right tools and more important, they do not use all relevant data. A closer look into the data analysis reveals that the HCS methods available prove to be very powerful. These methods, however, are not offered through available software. This results in methods that are inaccessible for most end-users. Unfortunately, most screening facilities still hire a bioinformatician for each single project, which requires multiple iterations of teaching sessions to explain the bioinformatician the way the screen was executed. In addition, the goal of the analysis needs to be explained. Frequently, static scripts are developed for a single project that cannot easily be reused and are inflexible because of hard-coded settings.

Specifically, the preparation of data, also called data preprocessing, is a burden for many bioinformaticians and data scientists. That is why a standard data analysis protocol was designed to optimize certain facets of carrying out HCS data analysis. The data set is by default split into consecutive sections in order to allow for parallel processing on a large scale. Inconsistencies in column names or data types across a data set are changed so that column names are unique and data types are consistent across the whole data set. This is required in order to automatically avoid problems in preprocessing the data. Further in the workflow a standard protocol is used for the elimination of features that can cause problems, such as uniform distributions, multicollinearity, and singularity. The solution is an easy to use software package that allows the biologist to analyze his/her own data and is designed for biologists, not for bioinformaticians. HC StratoMineR was designed, implemented, tested, and validated using two case studies to demonstrate the power of these available methods. All methods used were purely unsupervised.

In 2012, the interest in data science and machine learning was growing and so grew the interest in machine learning and supervised learning within the domain of High Content Screening. Supervised learning is a subset of machine learning where parts of a data set are labeled. A label, also called a class, represents a subpopulation. A machine learning model can be built to distinguish differences between the classes and can recognize if new data is likely to belong to a class. Unsupervised learning does not use data labels but purely uses the values of the data to show differences and similarities. This thesis pays attention to unsupervised and supervised learning in the domain of HCS. The results of this study show that Random Forest is a very fast and easy way of using supervised learning effectively when enough high-quality training data is available. For maximizing the level of exploration, combining unsupervised methods with supervised methods is highly recommended. Using unsupervised methods, suitable parts of the data can be identified that can be used for the annotation of classes used in training machine learning models.

Data visualization for the analysis of HCS data is an especially important tool that can be extremely useful for the detection of patterns in the data such as outliers, systematic errors, correlations, and clusters of data. In HCS, many features can be measured resulting in to high-dimensional data. HCS data is often heterogeneous and can be hard to interpret. That makes mining biological high-throughput data harder. In this thesis, attention was paid to the use of interactive visualizations vs. the use of static visualizations. Together with using interactive visualizations, the theory of Visual Data Mining (VDM) was considered. VDM is described in four steps: (i) overview, (ii) zoom, (iii) filter and (iv) details on demand. Our primary hypothesis was that users could work faster and more efficient when they would apply this theory with the ability to perform manipulations on the way the data is visualized such as zooming, filtering, linking and brushing. An experiment was set up to measure the accuracy and efficiency of using interactive visualizations under 79 students. The results demonstrate that the group that was using interactive visualizations was faster and made significantly less errors in the tasks that they had to perform compared to the students that were using non-interactive visualizations.

Data preprocessing is a challenging, time-consuming, and error-prone task in the domain of life sciences and HCS. Bioinformaticians are frequently working with scripts that are designed for specific experiments that cannot easily be reused and are reinventing the wheel, while preprocessing requires up to 80% of the time in data science. Therefore an R Package PurifyR was developed that offers built-in functionality to automatically preprocess a data matrix or data frame including ETL, feature selection, normalization, transformation, scaling and handling missing data in order to fully prepare the data set for machine learning such as PCA, regression and classification. The package has built-in default settings based on rules of thumb and best practices that can easily be adapted.

A validation of the methods we have proposed in this thesis is also applied in a low content screen to identify drug candidates for suppressing NF-κB activity. In this study, the workflow proposed for high content screening was applied to this data. In this analysis, the phases transformation, feature scaling, missing data, dimensionality reduction and clustering have been omitted because they are not applicable for low content data analysis. The data contained two important features: (i) Renilla, a measurement for cytotoxicity and transfection efficiency and (ii) Luciferase, a measurement for the NF-κB activity. The hit picking section resulted in 34 screened hits demonstrating a TNFα-induced transcriptional activity of the NF-κB reporter. The drugs demonstrating cytotoxicity were excluded and the final hit list resulted in five candidates reducing NF-κB transcriptional activity significantly.

Dutch Summary

In dit proefschrift worden het proces en de technieken van data-analyse bij High Content Screening (HCS) onderzocht. HCS, een onderdeel van High Throughput Screening (HTS), past robotica toe voor het analyseren van biochemische experimenten, op een deels geautomatiseerde manier. HTS combineert het gebruik van robotica, het automatisch verwerken van vloeistoffen en microplaten. HTS wordt toegepast binnen de biomedische genetica en bij vooronderzoek van geneesmiddelen. Met behulp van HTS kan de morfologie van grote hoeveelheden biochemische stoffen gemeten worden. Dit wordt gedaan met behulp van fluorescerende eiwitten, chemicaliën en antilichamen. Een belangrijke eigenschap van HTS-technieken is dat numerieke data kan worden afgeleid die de biochemische activiteit beschrijft.

HCS is een onderzoekstechniek waarbij afbeeldingen worden gegenereerd door middel van automatische microscopie. Van oudsher ondersteunt HTS geen afbeelding analyse. HCS combineert het gebruik van robotica, automatische fluorescentiemicroscopie, het verwerken van vloeistoffen, het genereren van afbeeldingen en het analyseren van afbeeldingen en numerieke data. Binnen HCS wordt multivariate data berekend en opgeslagen. Deze data worden verkregen door het segmenteren en analyseren van afbeeldingen. Deze datasets, verkregen door HCS, geven meer informatie over de fenotypische eigenschappen van de biochemische activiteit dan traditionele HTS-methoden. Door fluorescerende stoffen, die binden met eiwitten, worden metingen verkregen die informatie verschaffen over het fenotype van cellen. Metingen kunnen bijvoorbeeld iets zeggen over de intensiteit, vorm, afmeting en textuur. Deze metingen geven informatie over duizenden fenotypische eigenschappen van individuele cellen. Biologen kunnen met behulp van deze data het fenotypische effect van biochemische stoffen op cellen onderzoeken. Door HCS-technologieën kunnen grote hoeveelheden data worden gegenereerd. Deze data kan inzicht verschaffen in de werking van biochemische stoffen. Met de resultaten van HCS, kunnen opvallende metingen worden gedetecteerd. HCS wordt toegepast binnen verschillende gebieden, zoals geneesmiddelenonderzoek en toxicologie. Met de resultaten van HCS-onderzoek kunnen klinische onderzoeken worden uitgevoerd ten behoeve van de ontwikkeling van nieuwe geneesmiddelen.

In dit onderzoek is eerst gekeken naar het globale proces dat nodig is bij HCS. Wat opviel is dat een groot deel van de onderzoeksinstellingen geen gebruik maakt van de juiste software en niet alle bruikbare data in het data-analyse proces wordt gebruikt. Er zijn zeer krachtige HCS-methoden voorhanden, deze methoden ontbreken echter in de beschikbare software. Hierdoor zijn deze methoden voor eindgebruikers niet toegankelijk. Onderzoeksinstellingen huren daarom vaak, op projectbasis, een bio-informaticus in. Dit heeft een nadelig effect op de efficiëntie, omdat de ingehuurde krachten steeds opnieuw ingewerkt moeten worden. De programmacode die wordt ontwikkeld kan vaak slecht worden hergebruikt.

Met name het voorbereiden van data brengt vele uitdagingen met zich mee. Tijdens dit onderzoek is daarom een standaard data-analyse protocol ontworpen. Dit protocol optimaliseert het HCS-data analyse proces. Zo wordt data gesplitst om parallel verwerkt te kunnen worden. Het protocol draagt zorg voor de consistentie van de data. Hierdoor kan verdere verwerking van de data probleemloos verlopen. Daarnaast worden met het protocol ook andere problemen voorkomen, zoals non-normaliteit, singulariteit en multicollineariteit. De oplossing is een gebruiksvriendelijk softwarepakket dat biologen in staat stelt hun eigen data te analyseren. HC StratoMineR is ontworpen, ontwikkeld, getest en gevalideerd aan de hand van twee casestudies. Dit toont de waarde van deze, niet eerder geïmplementeerde, methodes aan.

In 2012 groeide de wereldwijde interesse in data science en machine learning en daarmee de interesse in supervised learning in het HCS-domein. Supervised learning is een onderdeel van machine learning dat delen van een dataset labelt. Een label, ook wel een klasse, representeert een gedeelte van een populatie. Met behulp van een model kan onderscheid worden gemaakt tussen verschillende klassen. Nieuwe data (data zonder label) kan zo geclassificeerd worden. Unsupervised learning maakt alleen gebruik van de waarden van data om deze te onderscheiden. In dit proefschrift wordt het toepassen van supervised learning binnen HCS onderzocht. De resultaten tonen aan dat het gebruik van Random Forest een snelle en eenvoudige manier is om supervised learning toe te passen wanneer er voldoende gelabelde data beschikbaar is. Om zo effectief mogelijk te onderzoeken wordt aangeraden om unsupervised learning te combineren met supervised learning. Met behulp van unsupervised learning kunnen bruikbare delen van de data worden geïdentificeerd om data van labels te voorzien.

Het visualiseren van data is een belangrijke tool om patronen te herkennen. Zo kunnen uitschieters, systematische patronen, correlaties en groepen in data worden geïdentificeerd. HCS levert veel metingen op, dit resulteert in data met veel variabelen. Deze data zijn vaak heterogeen en lastig te interpreteren. Dit maakt het onderzoeken van HCS-data uitdagend. In dit proefschrift is aandacht besteed aan het inzetten van interactieve visualisaties tegenover het gebruik van statische visualisaties. Door het gebruik van interactieve visualisaties is Visual Data Mining binnen handbereik. Visual Data Mining bestaat uit vier stappen: overzicht, inzoomen, filteren en details op verzoek. De primaire hypothese was dat gebruikers sneller en efficiënter konden werken door het gebruik van interactieve visualisaties. Deze hypothese is getoetst door middel van een experiment onder 79 studenten. De resultaten laten zien dat de groep die gebruik maakte van interactieve visualisaties sneller en accurater de analyse opdrachten konden voltooien.

Het voorbereiden van data is een uitdagende, tijdrovende en foutgevoelige taak. Bioinformatici werken vaak met specifiek inzetbare scripts of software, ontwikkeld voor specifieke experimenten en het voorbereiden van data. Deze software kan lastig worden hergebruikt, terwijl het voorbereiden van data tot 80% van de tijd in beslag neemt. Om het voorbereiden van data te vereenvoudigen is er, als onderdeel van dit proefschrift, software ontwikkeld in de vorm van een R-package. Deze software biedt de gebruiker de mogelijkheid om automatisch data voor te bereiden. Het ondersteunt de gebruiker bij ETL, variabele selectie, transformatie, normalisatie, standaardiseren en het verwerken van missende gegevens. Zo kan een dataset volledig worden voorbereid om machine learning toe te passen. De software heeft ingebouwde standaardwaarden die eenvoudig aangepast kunnen worden.

Het in dit proefschrift voorgestelde proces is tevens toegepast bij een low-content onderzoek naar geneesmiddelen. De voorgestelde methoden binnen het proces zijn gevalideerd door middel van een onderzoek naar geneesmiddelen die NF-κB activiteit onderdrukken. Bij dit onderzoek zijn een aantal fasen uit het voorgestelde proces achterwege gelaten, vanwege het lage aantal variabelen (low content) in dit onderzoek. De data bevatten twee belangrijke variabelen: Renilla en Luciferase. Er zijn 34 potentiële kandidaten gevonden die NF-κB activiteit onderdrukken. Geneesmiddelen die cytotoxische activiteit laten zien zijn buiten beschouwing gelaten. De uiteindelijke lijst resulteerde daardoor in vijf kandidaten die NF-κB activiteit significant onderdrukken.

Bibliography

1. Allan, C., Burel, J.-M., Moore, J., Blackburn, C., Linkert, M., Loynton, S., et al. (2012). OMERO: flexible, model-driven data management for experimental biology. [10.1038/nmeth.1896]. *Nat Meth*, 9(3), 245-253.
2. Almasi, G. S., & Gottlieb, A. (1988). Highly parallel computing.
3. Arabatzis, A. A., & Burkhart, H. E. (1992). An evaluation of sampling methods and model forms for estimating height-diameter relationships in loblolly pine plantations. *Forest science*, 38(1), 192-198.
4. Austin, R. J. et al. Mometasone furoate is a less specific glucocorticoid than fluticasone propionate. *Eur Respir J* 20, 1386–1392 (2002).
5. Bajcsy, P., Cardone, A., Chalfoun, J., Halter, M., Juba, D., Kociolek, M., ... & Vandecreme, A. (2015). Survey statistics of automated segmentations applied to optical imaging of mammalian cells. *BMC bioinformatics*, 16(1), 330.
6. Bajorath, J. (2002). Integration of virtual and high-throughput screening. [10.1038/nrd941]. *Nat Rev Drug Discov*, 1(11), 882-894.
7. Banks, P., Appledorn, D., Shumate, C., Schneider, P., Boettcher, K. (2017, August). Roundup: Lights! Camera! Live-Cell Imaging! GEN Genetic Engineering & Biotechnology News, 37(14).
8. Bekkers, W., Weerd, I., Spruit, M., & Brinkkemper, S. (2010). A Framework for Process Improvement in Software Product Management. Systems, Software and Services Process Improvement. In A. Riel, R. O'Connor, S. Tichkiewitch & R. Messnarz (Eds.), (Vol. 99, pp. 1-12): Springer Berlin Heidelberg.
9. Bellomo, F., Medina, DL, De Leo, E., Panarella, A., & Emma, F. (2017). High-content drug screening for rare diseases. *Journal of Inherited Metabolic Disease*, 40 (4), 601-607.
10. Bergström, S., & Ivarsson, O. (2015). Automation of a Data Analysis Pipeline for High-content Screening Data.
11. Berman, H. M. et al. The Protein Data Bank. *Nucleic Acids Res* 28, 235–242 (2000).
12. Billingsley, K. J., Bandres-Ciga, S., Saez-Atienzar, S., & Singleton, A. B. (2018). Genetic risk factors in Parkinson's disease. *Cell and tissue research*, 1-12.
13. Birmingham, A., Selfors, L. M., Forster, T., Wrobel, D., Kennedy, C. J., Shanks, E., ... & Smith, Q. (2009). Statistical methods for analysis of high-throughput RNA interference screens. *Nature methods*, 6(8), 569-575.

14. Birmingham, A., Selfors, L. M., Forster, T., Wrobel, D., Kennedy, C. J., Shanks, E., ... & Smith, Q. (2009). Statistical methods for analysis of high-throughput RNA interference screens. *Nature methods*, 6(8), 569.
15. Bougen-Zhukov, N., Loh, S. Y., Lee, H. K., & Loo, L. H. (2017). Large-scale image-based screening and profiling of cellular phenotypes. *Cytometry Part* A, 91(2), 115-125.
16. Bourke, J., Coulson, I., English, J., British Association of Dermatologists Therapy, G. & Audit, S. Guidelines for the management of contact dermatitis: an update. *Br J Dermatol* 160, 946–954 (2009).
17. Boutros, M., Heigwer, F., & Laufer, C. (2015). Microscopy-based high-content screening. *Cell*, 163(6), 1314-1325.
18. Boutros, M., Brás, L. P., & Huber, W. (2006). Analysis of cell-based RNAi screens: BioMed Central Ltd.
19. Bray, M. A., Singh, S., Han, H., Davis, C. T., Borgeson, B., Hartland, C., ... & Carpenter, A. E. (2016). Cell Painting, a high-content image-based assay for morphological profiling using multiplexed fluorescent dyes. *Nature protocols*, 11(9), 1757.
20. Bray, M. A., & Carpenter, A. (2017). Advanced assay development guidelines for image-based high content screening and analysis. In *Assay Guidance Manual*. Eli Lilly & Company and the National Center for Advancing Translational Sciences.
21. Breiman, L. (2001), *Random Forests*, Machine Learning 45(1), 5-32.
22. Buchberger, A. R., DeLaney, K., Johnson, J., & Li, L. (2017). Mass spectrometry imaging: a review of emerging advancements and future insights. *Analytical chemistry*, 90(1), 240-265.
23. Burbaum, J. J. (1998). Miniaturization technologies in HTS: how fast, how small, how soon? Drug Discovery Today, 3(7), 313-322.
24. Burris, T. P. et al. Nuclear receptors and their selective pharmacologic modulators. *Pharmacol Rev* 65, 710–778 (2013).
25. Caicedo, J. C., Cooper, S., Heigwer, F., Warchal, S., Qiu, P., Molnar, C., ... & Wawer, M. (2017). Data-analysis strategies for image-based cell profiling. *Nature methods*, 14(9), 849.
26. Campeau, E. et al. A versatile viral system for expression and depletion of proteins in mammalian cells. *PLoS One* 4, e6529 (2009).
27. Card SK., Mackinlay JD, Shneiderman B: *Readings in Information Visualization: Using Vision to Think*. San Francisco, Calif.: Morgan Kaufmann, 2007.

28. Carpenter, A., Jones, T., Lamprecht, M., Clarke, C., Kang, I., Friman, O., et al. (2006). CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biology*, 7(10), R100.
29. Chang, C. C., & Lin, C. J. (2011). LIBSVM: A library for support vector machines. ACM *transactions on intelligent systems and technology* (TIST), 2(3), 1-27.
30. Chapman P, Clinton J, Kerber R, et al: CRISP-DM 1.0 *Step-by-step data mining guide*. 2000.
31. Chong, H. K. et al. Genome-wide interrogation of hepatic FXR reveals an asymmetric IR-1 motif and synergy with LRH-1. *Nucleic Acids Res* 38, 6007–6017 (2010).
32. Clavel, J., Escarguel, G., & Merceron, G. (2015). mvMORPH: an R package for fitting multivariate evolutionary models to morphometric data. *Methods in Ecology and Evolution*, 6(11), 1311-1319.
33. Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1), 37-46.
34. Costello AB, Osborne JW: Best practices in exploratory factor analysis: Four recommendations for getting the most from your analysis. *Practical Assessment, Research & Evaluation* 2005;10(7): 1-9.
35. Cutler, A., Cutler, D. R., & Stevens, J. R. (2012). Random forests. Ensemble Machine Learning: Methods and Applications.
36. Danovi, D., Folarin, A. A., Baranowski, B., & Pollard, S. M. (2012). High content screening of defined chemical libraries using normal and glioma-derived neural stem cell lines. In *Methods in enzymology* (Vol. 506, pp. 311-329). Academic Press.
37. Dao, D., Fraser, A. N., Hung, J., Ljosa, V., Singh, S., & Carpenter, A. E. (2016). CellProfiler Analyst: interactive data exploration, analysis, and classification of large biological image sets. *Bioinformatics*, 32(20), 3210-3212.
38. Datta, S. et al. (2010) An adaptive optimal ensemble classifier via bagging and rank aggregation with applications to high dimensional data. BMC *bioinformatics*, 11(1), 427.
39. Davenport, T. H., & Patil, D. J. (2012). Data scientist. *Harvard business review*, 90(5), 70-76.
40. De Bosscher, K., Haegeman, G. & Elewaut, D. Targeting inflammation using selective glucocorticoid receptor modulators. *Curr Opin Pharmacol* 10, 497–504 (2010).
41. Dietz, C., & Berthold, M. R. (2016). KNIME for open source bioimage analysis: a tutorial. In *Focus on Bio-Image Informatics* (pp. 179-197). Springer, Cham.

42. Dinkla, K., Strobelt, H., Genest, B., Reiling, S., Borowsky, M., & Pfister, H. (2017). Screenit: Visual Analysis of Cellular Screens. *IEEE transactions on visualization and computer graphics*, 23(1), 591-600.
43. Echeverri, C. J., & Perrimon, N. (2006). High-throughput RNAi screening in cultured cells: a user's guide. *Nature Reviews Genetics*, 7(5), 373.Esner, M., Meyenhofer, F., & Bickle, M. (2018). Live-Cell High Content Screening in Drug Development. In *High Content Screening* (pp. 149-164). Humana Press, New York, NY.
44. Eden, E., Navon, R., Steinfeld, I., Lipson, D., & Yakhini, Z. (2009). GOrilla: a tool for discovery and visualization of enriched GO terms in ranked gene lists. *BMC bioinformatics*, 10(1), 48.
45. FasteR! HigheR! StrongeR! - A Guide to Speeding Up R Code for Busy People. Available at: http://www.noamross.net/blog/2013/4/25/faster-talk.html. Last accessed February 29, 2016.
46. Fayyad, U. M., Piatetsky-Shapiro, G., & Smyth, P. (1996, August). Knowledge Discovery and Data Mining:Towards a Unifying Framework. In *KDD* (Vol. 96, pp. 82-88).
47. Fayyad U, Piatetsky-Shapiro G, Smyth P: The KDD Process for Extracting Useful Knowledge from Volumes of Data. *Communications of the ACM* 1996;39(11) 27-34.
48. Forman, G., & Scholz, M. (2010). Apples-to-apples in cross-validation studies: pitfalls in classifier performance measurement. *Acm Sigkdd Explorations Newsletter*, 12(1), 49-57.
49. Fuchs, F., Pau, G., Kranz, D., Sklyar, O., Budjan, C., Steinbrink, S., ... & Boutros, M. (2010). Clustering phenotype populations by genome-wide RNAi and multiparametric imaging. *Molecular systems biology*, 6(1), 370.
50. Friendly M: A Brief History of Data Visualization. *Springer Handbooks Comp.Statistics Handbook of Data Visualization* 2008;15-56.
51. Gadaleta, R. M. et al. Activation of bile salt nuclear receptor FXR is repressed by pro-inflammatory cytokines activating NF-kappaB signaling in the intestine. *Biochim Biophys Acta* 1812, 851–858 (2011).
52. Gadaleta, R. M. et al. Farnesoid X receptor activation inhibits inflammation and preserves the intestinal barrier in inflammatory bowel disease. *Gut* 60, 463–472 (2011).
53. Galili, T., O'Callaghan, A., Sidi, J., & Sievert, C. (2017). heatmaply: an R package for creating interactive cluster heatmaps for online publishing. *Bioinformatics*, 34(9), 1600-1602.

54. García, S., Fernández, A., Luengo, J., & Herrera, F. (2009). A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability. *Soft Computing*, 13(10), 959.
55. Gentleman, R., Carey, V., Bates, D., Bolstad, B., Dettling, M., Dudoit, S., et al. (2004). Bioconductor: open software development for computational biology and bioinformatics. *Genome Biology*, 5(10), R80.
56. Giacomelli, P. (2013). *Apache mahout cookbook*. Packt Publishing Ltd.
57. Gioiello, A. et al. Bile acid derivatives as ligands of the farnesoid x receptor: molecular determinants for bile acid binding and receptor modulation. *Curr Top Med Chem* 14, 2159–2174 (2014).
58. Glass, C. K. & Saijo, K. Nuclear receptor transrepression pathways that regulate inflammation in macrophages and T cells. *Nat Rev Immunol* 10, 365–376 (2010).
59. Glickman, J. F. et al. A comparison of ALPHAScreen, TR-FRET, and TRF as assay methods for FXR nuclear receptors. *J Biomol Screen* 7, 3–10 (2002).
60. Goldberg, I., Allan, C., Burel, J.-M., Creager, D., Falconi, A., Hochheiser, H., et al. (2005). The Open Microscopy Environment (OME) Data Model and XML file: open tools for informatics and quantitative analysis in biological imaging. *Genome Biology*, 6(5), R47.
61. Greene CS, Tan J, Ung M, Moore JH, Cheng C: Big data bioinformatics. *Journal of cellular physiology* 2014;229(12), 1896-1900.
62. Harper, G., & Pickett, S. D. (2006). Methods for mining HTS data. *Drug Discovery Today*, 11(15-16), 694-699.
63. Hauge, H., Patzke, S., & Aasheim, H. C. (2007). Characterization of the FAM110 gene family. *Genomics*, 90(1), 14-27.
64. He, Y. et al. Structures and mechanism for the design of highly potent glucocorticoids. *Cell Res* 24, 713–726 (2014).
65. Healey CG, Booth KS, Enns JT: Visualizing Real-time Multivariate Data Using Preattentive Processing. ACM *Transactions on Modeling and Computer Simulation* 1995;5(3) 190-221.Hevner, A.R., March, S.T., Park, J., & Ram, S. (2004). Design science in information systems research. MIS *quarterly*, 28(1), 75-105.
66. Hoffman, A. F., Nolan, J., Gebhard, D. F., Nickischer, D., Omta, W., Cooper, S., ... & Fennell, M. (2018). Society of Biomolecular Imaging and Informatics High-Content Screening/High-Content Analysis Emerging Technologies in Biological Models, When and Why? *Assay and drug development technologies*, 16(1), 1-6.

67. Hollman, D. A., Milona, A., van Erpecum, K. J. & van Mil, S. W. Anti-inflammatory, and metabolic actions of FXR: insights into molecular mechanisms. *Biochim Biophys Acta* 1821, 1443–1452 (2012).
68. Honaker J, King G, Blackwell M: Amelia II: A program for missing data. *Journal of statistical software* 2011;45(7), 1-47.
69. Horton, N. J., Baumer, B. S., & Wickham, H. (2015). Setting the stage for data science: integration of data management skills in introductory and second courses in statistics. *arXiv preprint arXiv*:1502.00318.
70. Hsu, C. W. et al. Quantitative High-Throughput Profiling of Environmental Chemicals and Drugs that Modulate Farnesoid X Receptor. *Sci Rep* 4, 6437 (2014).
71. Hughes, J. P., Rees, S., Kalindjian, S. B., & Philpott, K. L. (2011). Principles of early drug discovery. *British journal of pharmacology*, 162(6), 1239-1249.
72. Hyvärinen, A., & Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5), 411-430.
73. Inglese, J., & Auld, D. S. (2007). High Throughput Screening (HTS) techniques: applications in chemical biology. *Wiley Encyclopedia of Chemical Biology*, 1-15.
74. Jansen, S. (2009). Applied Multi-Case Research in a Mixed-Method Research Project: Customer Configuration Updating Improvement *Information Systems Research Methods, Epistemology, and Applications* (pp. 120-139): IGI Global.
75. Javed W, Elmqvist N, Yi JS: Direct manipulation through surrogate objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* ACM 2011;627-636.
76. Jiang, H., He, X., Feng, D., Zhu, X., & Zheng, Y. (2015). RanGTP aids anaphase entry through Ubr5-mediated protein turnover. J Cell Biol, 211(1), 7-18.
77. Jones, T., Kang, I., Wheeler, D., Lindquist, R., Papallo, A., Sabatini, D., et al. (2008). CellProfiler Analyst: data exploration and analysis software for complex image-based screens. *BMC Bioinformatics*, 9(1), 482.
78. Joslin, J., Gilligan, J., Anderson, P., Garcia, C., Sharif, O., Hampton, J., ... & Trussell, C. (2018). A Fully Automated High-Throughput Flow Cytometry Screening System Enabling Phenotypic Drug Discovery. *SLAS DISCOVERY: Advancing Life Sciences R&D*, 2472555218773086.
79. Kamentsky, L., Jones, T. R., Fraser, A., Bray, M.-A., Logan, D. J., Madden, K. L., et al. (2011). Improved structure, function, and compatibility for CellProfiler: modular high-throughput image analysis software. *Bioinformatics*, 27(8), 1179-1180.

80. Keim DA: Information Visualization and Visual Data Mining. *IEEE Transactions on Visualization and Computer Graphics* 2002;8(1): 1-8.
81. Kim, D. H. et al. A dysregulated acetyl/SUMO switch of FXR promotes hepatic inflammation in obesity. *EMBO J* 34, 184–199 (2015).
82. King, E. M. et al. Glucocorticoid repression of inflammatory gene expression shows differential responsiveness by transactivation- and transrepression-dependent mechanisms. *PLoS One* 8, e53936 (2013).
83. Kraljevic, S., Stambrook, P. J., & Pavelic, K. (2004). Accelerating drug discovery: Although the evolution of '-omics' methodologies is still in its infancy, both the pharmaceutical industry and patients could benefit from their implementation in the drug development process. *EMBO reports*, 5(9), 837-842.
84. Kraus, O. Z., Grys, B. T., Ba, J., Chong, Y., Frey, B. J., Boone, C., & Andrews, B. J. (2017). Automated analysis of high-content microscopy data with deep learning. *Molecular systems biology*, 13(4), 924.
85. Kriston-Vizi, J., & Flotow, H. (2017). Getting the whole picture: High content screening using three-dimensional cellular model systems and whole animal assays. *Cytometry Part* A, 91(2), 152-159.
86. Labuda, D., Krajinovic, M., Richer, C., Skoll, A., Sinnett, H., Yotova, V., & Sinnett, D. (1999). Rapid detection of CYP1A1, CYP2D6, and NAT variants by multiplex polymerase chain reaction and allele-specific oligonucleotide assay. *Analytical biochemistry*, 275(1), 84-92.
87. Lamprecht, M. R., Sabatini, D. M., & Carpenter, A. E. (2007). CellProfiler: free, versatile software for automated biological image analysis. *BioTechniques*, 42(1), 71-75.
88. Larson, D., & Chang, V. (2016). A review and future direction of agile, business intelligence, analytics and data science. *International Journal of Information Management*, 36(5), 700-710.
89. Leonelli, S. (2012). Introduction: Making sense of data-driven research in the biological and biomedical sciences.
90. Li, S., Besson, S., Blackburn, C., Carroll, M., Ferguson, R. K., Flynn, H., ... & Moore, W. J. (2016). Metadata management for high content screening in OMERO. *Methods*, 96, 27-32.
91. Li, S. (2017). *Evaluation and Improvement of Current Computational Tools for Metabolomics Data Analysis* (Doctoral dissertation, Auckland University of Technology).
92. Li, T., Chen, L., Cheng, J., Dai, J., Huang, Y., Zhang, J., ... & Yin, X. (2016). SUMOylated NKAP is essential for chromosome alignment by anchoring CENP-E to kinetochores. *Nature communications*, 7, 12969.

93. Little, R. J. (1988). A test of missing completely at random for multivariate data with missing values. *Journal of the American Statistical Association*, 83(404), 1198-1202.

94. Ljosa, V., Caie, P. D., Ter Horst, R., Sokolnicki, K. L., Jenkins, E. L., Daya, S., ... & Clemons, P. A. (2013). Comparison of methods for image-based profiling of cellular morphological responses to small-molecule treatment. *Journal of biomolecular screening*, 18(10), 1321-1329.

95. Macarron, R., Banks, M. N., Bojanic, D., Burns, D. J., Cirovic, D. A., Garyantes, T., ... & Schopfer, U. (2011). Impact of high-throughput screening in biomedical research. *Nature reviews Drug discovery*, 10(3), 188.

96. Madauss, K. P. et al. Progesterone receptor ligand binding pocket flexibility: crystal structures of the norethindrone and mometasone furoate complexes. *J Med Chem* 47, 3381–3387 (2004).

97. Magidson, V., & Khodjakov, A. (2013). Circumventing photodamage in live-cell microscopy. In *Methods in cell biology* (Vol. 114, pp. 545-560). Academic Press.

98. Malo, N., Hanley, J. A., Cerquozzi, S., Pelletier, J., & Nadon, R. (2006). Statistical practice in high-throughput screening data analysis. [10.1038/nbt1186]. *Nat Biotech*, 24(2), 167-175.

99. Maurer, H. H. (2000). Screening Procedures for Simultaneous Detection of Several Drug Classes Used for High Throughput Toxicological Analyses and Doping Control. A Review. *Combinatorial Chemistry & High Throughput Screening*, 3(6), 467-480.

100. Marr B: Big Data: 20 Mind-Boggling Facts Everyone Must Read. *Forbes Magazine*, 2015

101. Martin, S., Buehler, G., Ang, K. L., Feroze, F., Ganji, G., & Li, Y. (2013). Cell-based RNAi assay development for HTS. In *Assay Guidance Manual [Internet]*. Eli Lilly & Company and the National Center for Advancing Translational Sciences.

102. Matsukuma, K. E. et al. Coordinated control of bile acids and lipogenesis through FXR-dependent regulation of fatty acid synthase. *J Lipid Res* 47, 2754–2761 (2006).

103. Marsaglia G, Tsang WW, Wang J: Evaluating Kolmogorov's distribution. *Journal of Statistical Software* 2003;8(18).

104. Marx V: Biology: The big challenges of big data. *Nature* 2013;498(7453), 255-260.

105. Menger V, Spruit MR, Hagoort K, et al: Transitioning to a Data Driven Mental Health Practice: Collaborative Expert Sessions for Knowledge and

Hypothesis Finding. *Computational and mathematical methods in medicine* 2016.

106. Mi, L. Z. et al. Structural basis for bile acid binding and activation of the nuclear receptor FXR. *Mol Cell* 11, 1093–1100 (2003).

107. Moffat, J. G., Vincent, F., Lee, J. A., Eder, J., & Prunotto, M. (2017). Opportunities and challenges in phenotypic drug discovery: an industry perspective. *Nature reviews Drug discovery*, 16(8), 531.

108. Mooi E, Sarstedt M: A Concise Guide to Market Research. In: *Cluster analysis*, pp. 237-284, Springer Berlin Heidelberg, 2011.

109. Munoz, D. M., Cassiani, P. J., Li, L., Billy, E., Korn, J. M., Jones, M. D., ... & DeWeck, A. (2016). CRISPR screens provide a comprehensive assessment of cancer vulnerabilities but generate false-positive hits for highly amplified genomic regions. *Cancer discovery*, 6(8), 900-913.

110. Murie, C., Barette, C., Button, J., Lafanechère, L., & Nadon, R. (2015). Improving detection of rare biological events in high-throughput screens. *Journal of biomolecular screening*, 20(2), 230-241.

111. Nadanaciva S, et al: A high content screening assay for identifying lysosomotropic compounds. *Toxicol In Vitro* 2011;25(3): 715-723.

112. Netzer, R., Bischoff, U., & Ebneth, A. (2003). HTS techniques to investigate the potential effects of compounds on cardiac ion channels at early-stages of drug discovery. *Current opinion in drug discovery & development*, 6(4), 462-469.

113. Neumann, B., Walter, T., Hériché, J. K., Bulkescher, J., Erfle, H., Conrad, C., ... & Cetin, C. (2010). Phenotypic profiling of the human genome by time-lapse microscopy reveals cell division genes. *Nature*, 464(7289), 721.

114. Nickischer, D., Elkin, L., Cloutier, N., O'Connell, J., Banks, M., & Weston, A. (2018). Challenges and opportunities in enabling high throughput, miniaturized high content screening. In *High Content Screening* (pp. 165-191). Humana Press, New York, NY.

115. Nowak, D. E., Tian, B. & Brasier, A. R. Two-step cross-linking method for identification of NF-kappaB gene network by chromatin immunoprecipitation. *Biotechniques* 39, 715–725 (2005).

116. Omta W.A., Egan D.A., Klumperman J., Spruit M.R., Brinkkemper S. (2013). HTS-IA: High Throughput Screening Information Architecture for Genomics. *International Journal of Healthcare Information Systems and Informatics*, 8(4), 17-31.

117. Omta, W. A., van Heesbeen, R. G., Pagliero, R. J., van der Velden, L. M., Lelieveld, D., Nellen, M., ... & Spruit, M. (2016). HC StratoMineR: a web-based

tool for the rapid analysis of high-content data sets. *Assay and drug development technologies*, 14(8), 439-452.

118. Omta, W. A., Nobel, J. D., Klumperman, J., Egan, D. A., Spruit, M. R., & Brinkhuis, M. J. (2017). Improving Comprehension Efficiency of High Content Screening Data Through Interactive Visualizations. *Assay and drug development technologies*, 15(6), 247-256.
119. Oshiro, T. M. et al. (2012) How many trees in a random forest? In *International Workshop on Machine Learning and Data Mining in Pattern Recognition* (pp. 154-168). Springer Berlin Heidelberg.
120. Ostenfeld MS, Fehrenbacher N, Høyer-Hansen M, Thomsen C, Farkas T, and Jäättelä M: Effective tumor cell death by σ-2 receptor ligand siramesine involves lysosomal leakage and oxidative stress. *Cancer research* 2005;65(19), 8975-8983.
121. Pagliero, R. J., D'Astolfo, D. S., Lelieveld, D., Pratiwi, R. D., Aits, S., Jaattela, M., ... & Egan, D. A. (2016). Discovery of small molecules that induce lysosomal cell death in cancer cell lines using an image-based screening platform. *Assay and drug development technologies*, 14(8), 489-510.
122. Parmigiani, G. (2001). Decision theory: Bayesian.
123. Pechenizkiy, M., Puuronen, S., & Tsymbal, A. (2006, April). The impact of sample reduction on PCA-based feature extraction for supervised learning. In *Proceedings of the 2006 ACM symposium on Applied computing* (pp. 553-558).
124. Pellicciari, R. et al. 6alpha-ethyl-chenodeoxycholic acid (6-ECDCA), a potent and selective FXR agonist endowed with anticholestatic activity. *J Med Chem* 45, 3569–3572 (2002).
125. Pelz, O., Gilsdorf, M., & Boutros, M. (2010). web cellHTS2: A web-application for the analysis of high-throughput screening data. *BMC Bioinformatics*, 11(1), 185.
126. Persidis, A. (1998). High-throughput screening. [10.1038/nbt0598-488]. *Nat Biotech*, 16(5), 488-489.
127. Piccinini, F., Balassa, T., Szkalisity, A., Molnar, C., Paavolainen, L., Kujala, K., ... & Smith, K. (2017). Advanced cell classifier: user-friendly machine-learning-based software for discovering phenotypes in high-content imaging data. *Cell systems*, 4(6), 651-655.
128. Ripley, B. D. (1996). Pattern classification and neural networks.
129. Ripley, B. D. (2001). The R project in statistical computing. *MSOR Connections. The newsletter of the LTSN Maths, Stats & OR Network*, 1(1), 23-25.

130. Ren, J., Guo, H., Xu, C., & Zhang, Y. (2017). Serving at the edge: A scalable IoT architecture based on transparent computing. IEEE Network, 31(5), 96-105.

131. Reynolds AP, Richards G, de la Iglesia B, Rayward-Smith VJ: Clustering rules: a comparison of partitioning and hierarchical clustering algorithms. *Journal of Mathematical Modelling and Algorithms* 2006;5(4), 475-504.

132. Rouleau, N., Turcotte, S., Mondou, M. H., Roby, P. & Bosse, R.Development of a versatile platform for nuclear receptor screening using AlphaScreen. *J Biomol Screen* 8, 191–197 (2003).

133. Royston P: Algorithm AS 181: The W test for Normality. *Applied Statistics* 1982a;31, 176–180.

134. Royston P: An extension of Shapiro and Wilk's W test for normality to large samples. *Applied Statistics* 1982b;31, 115–124.

135. Rueden, C. T., Schindelin, J., Hiner, M. C., DeZonia, B. E., Walter, A. E., Arena, E. T., & Eliceiri, K. W. (2017). ImageJ2: ImageJ for the next generation of scientific image data. *BMC bioinformatics*, *18*(1), 529.

136. Saccani, S., Pantano, S. & Natoli, G. p38-Dependent marking of inflammatory genes for increased NF-kappa B recruitment. *Nat Immunol* 3, 69–75 (2002).

137. Saijo, K. et al. A Nurr1/CoREST pathway in microglia and astrocytes protects dopaminergic neurons from inflammation-induced death. *Cell* 137, 47–59 (2009).

138. Sato, T. et al. Single Lgr5 stem cells build crypt-villus structures in vitro without a mesenchymal niche. *Nature* 459, 262–265 (2009).

139. Schacke, H., Docke, W. D. & Asadullah, K. Mechanisms involved in the side effects of glucocorticoids. *Pharmacol Ther* 96, 23–43 (2002).

140. Scholl, P. M., Wille, M., & Van Laerhoven, K. (2015, September). Wearables in the wet lab: a laboratory system for capturing and guiding experiments. *In Proceedings of the* 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (pp. 589-599). ACM.

141. Scheeder, C., Heigwer, F., & Boutros, M. (2018). Machine learning and image-based profiling in drug discovery. *Current opinion in systems biology*, 10, 43-52.

142. Schölkopf, B., Smola, A. J., Williamson, R. C., & Bartlett, P. L. (2000). New support vector algorithms. *Neural computation*, *12*(5), 1207-1245.

143. Shneiderman, B. (1982). The future of interactive systems and the emergence of direct manipulation. *Behaviour & Information Technology*, 1(3), 237-256.

144. Shneiderman B: Inventing Discovery Tools: Combining Information Visualization with Data Mining. *Discovery Science Lecture Notes in Computer Science* 2001;17-28.

145. Seok, H., Lee, H., Jang, E. S., & Chi, S. W. (2018). Evaluation and control of miRNA-like off-target repression for RNA interference. *Cellular and molecular life sciences*, 75(5), 797-814.
146. Singh, S., Carpenter, A. E., & Genovesio, A. (2014). Increasing the content of high-content screening: an overview. *Journal of biomolecular screening*, 19(5), 640-650.
147. Sommer, C., Hoefler, R., Samwer, M., & Gerlich, D. W. (2017). A deep learning and novelty detection framework for rapid phenotyping in high-content screening. *Molecular biology of the cell*, 28(23), 3428-3436.
148. Spruit,M., & Jagesar,R. (2016). Power to the People! Meta-algorithmic modelling in applied data science. In Fred,A. et al. (Ed.), *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management* (pp. 400–406). KDIR 2016, November 11-13, 2016, Porto, Portugal: ScitePress.
149. Spruit,M., & Lytras,M. (2018). Applied Data Science in Patient-centric Healthcare: Adaptive Analytic Systems for Empowering Physicians and Patients. *Telematics and Informatics*, 35(4), Special Issue: Patient Centric Healthcare, 643–653.
150. Stanzione, M., Baumann, M., Papanikos, F., Dereli, I., Lange, J., Ramlal, A., ... & Jasin, M. (2016). Meiotic DNA break formation requires the unsynapsed chromosome axis-binding protein IHO1 (CCDC36) in mice. *Nature cell biology*, 18(11), 1208.
151. Steenbergen, M., & Brinkkemper, S. (2007). An instrument for the Development of the Enterprise Architecture Practice. *Proceedings of the 9th International Conference on Enterprise Information Systems*, (pp. 14-22).
152. Steenbergen, M., Bos, R., Brinkkemper, S., Weerd, I., & Bekkers, W. (2010). The Design of Focus Area Maturity Models. In R. Winter, J. L. Zhao & S. Aier (Eds.), Global Perspectives on Design Science Research (Vol. 6105, pp. 317-332): Springer Berlin Heidelberg.
153. Stein, R. L. (2003). High-throughput screening in academia: the Harvard experience. *Journal of biomolecular screening*, 8(6), 615-619.
154. Stodder, D. (2017). Why data preparation matters. In Dooley, B., Data preparation challenges facing every enterprise (pp. 5-11). Retrieved from TDWI e-book database.
155. Sullivan, D. P., Faeder, J. R., Rhode, G. K., & Sbalzarini, I. (2015). Image-derived generative modeling of complex cellular organization in both space and time.

156. Sundberg, S. A. (2000). High-throughput and ultra-high-throughput screening: solution-and cell-based approaches. *Current opinion in biotechnology*, 11(1), 47-53.
157. Swinney DC: Phenotypic vs. target-based drug discovery for first-in-class medicines. *Clin Pharmacol Ther* 2013;93(4), 299-301.
158. Szklarczyk D, et al: The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic acids research* 39(1), 561-568.
159. Szymańska, E., Brown, P. A., Ziere, A., Martins, S., Batenburg, M., Harren, F. J., & Buydens, L. M. (2015). Comprehensive data scientific procedure for enhanced analysis and interpretation of real-time breath measurements in in vivo aroma-release studies. *Analytical chemistry*, 87(20), 10338-10345.
160. Tan, A., Tripp, B., & Daley, D. (2011). BRISK—research-oriented storage kit for biology-related data. Bioinformatics, 27(17), 2422-2425.
161. Thomas, N. (2010). High-content screening: a decade of evolution. *Journal of biomolecular screening*, 15 (1), 1-9.
162. Tolopko, A., Sullivan, J., Erickson, S., Wrobel, D., Chiang, S., Rudnicki, K., et al. (2010). Screensaver: an open source lab information management system (LIMS) for high throughput screening facilities. *BMC Bioinformatics*, 11(1), 260.
163. Tsiper, M. V., Sturgis, J., Avramova, L. V., Parakh, S., Fatig, R., Juan-García, A., ... & Robinson, J. P. (2012). Differential mitochondrial toxicity screening and multi-parametric data analysis. *PLoS One*, 7(10), e45226.
164. Tufte ER: The Visual Display Of Quantitative Information. *Journal For Healthcare Quality* 1985;7 3-15.
165. van de Weerd, I., & Brinkkemper, S. (2009). Meta-Modeling for Situational Analysis and Design Methods *Handbook of Research on Modern Systems Analysis and Design Technologies and Applications* (pp. 35-54): IGI Global.
166. van Heesbeen RGHP: *Mitotic spindle assembly: May the force be with you*. Utrecht University, Utrecht, the Netherlands, 2015; ISBN: 978-94-6233-037-5.
167. van Heesbeen, R. G. et al. (2016) Aurora A, MCAK, and Kif18b promote Eg5-independent spindle formation. *Chromosoma*, 1-14.
168. van Buuren, S. V., & Groothuis-Oudshoorn, K. (2010). mice: Multivariate imputation by chained equations in R. *Journal of statistical software*, 1-68.
169. van Buuren, S. (2014). MICE: Multiple Imputation by Chained Equations. R Package Version 2.21.

170. van Rossum, T., Tripp, B., & Daley, D. (2010). SLIMS—a user-friendly sample operations and inventory management system for genotyping labs. *Bioinformatics*, 26(14), 1808-1810.

171. Vavassori, P., Mencarelli, A., Renga, B., Distrutti, E. & Fiorucci, S.The bile acid receptor FXR is a modulator of intestinal innate immunity. *J Immunol* 183, 6251–6261 (2009).

172. Wang, Y. D. et al. Farnesoid X receptor antagonizes nuclear factor kappaB in hepatic inflammatory response. *Hepatology* 48, 1632–1643 (2008).

173. Weigt, D., Sammour, D. A., Ulrich, T., Munteanu, B., & Hopf, C. (2018). Automated analysis of lipid drug-response markers by combined fast and high-resolution whole cell MALDI mass spectrometry biotyping. *Scientific reports*, 8(1), 11260.

174. Wickelgren WA: Speed-accuracy tradeoff and information processing dynamics. *Acta psychologica* 1977;41(1) 67-85.

175. Wickham H: A Layered Grammar of Graphics. *Journal of Computational and Graphical Statistics* 2010;19(1), 3-28.

176. Wickham, H. (2014). Tidy data. *Journal of Statistical Software*, 59(10), 1-23.

177. Wickham, H. (2019). Data science: how is it different to statistics? *IMS Bulletin*, 48.

178. Williams, M. (2011). Productivity shortfalls in drug discovery: contributions from the preclinical sciences? *Journal of Pharmacology and Experimental Therapeutics*, 336(1), 3-8.

179. Wu, J., Dong, M., Ota, K., Li, J., & Guan, Z. (2018). Big data analysis-based secure cluster management for optimized control plane in software-defined networks. IEEE Transactions on Network and Service Management, 15(1), 27-38.

180. www.clinicaltrial.gov A service of the U.S. National Institutes of Health. Date of access 09/01/2015. ClinicalTrial.gov Identifier NCT01999101 (NAFLD), NCT01265498 (NASH), NCT02308111 (PBC), NCT02177136 (PSC).

181. Yang, L., Yang, S., Li, X., Li, B., Li, Y., Zhang, X., ... & Wei, S. (2019). Tumor organoids: From inception to future in cancer research. *Cancer letters*.

182. Yin, R. (2003). *Case study research: design and methods*: Sage Publications.

183. Young, D. W., Bender, A., Hoyt, J., McWhinnie, E., Chirn, G. W., Tao, C. Y., ... & Feng, Y. (2008). Integrating high-content screening and ligand-target prediction to identify mechanism of action. *Nature chemical biology*, 4(1), 59.

List of Publications

1. Omta, W.A., van Heesbeen, R. G., Shen, I., Nobel, J., Robers, D., van der Velden, L.M., Medema, R.M., Siebes, A.P.J.M., Feelders, A. J., Brinkkemper, S., Klumperman, J., Spruit, M.R., Brinkhuis, M.J.S. & Egan, D. A. (2020). Combining Supervised and Unsupervised Machine Learning Methods for Phenotypic Functional Genomics Screening. SLAS DISCOVERY: Advancing the Science of Drug Discovery, 2472555220919345.
2. Omta, W. A., van Heesbeen, R. G., Shen, I., Feelders, A. J., Brinkhuis, M. J. S., Egan, D. A., & Spruit, M. R. (2020). PurifyR: An R Package for Highly Automated, Reproducible Variable Extraction and Standardization. *Systems Medicine*, 3(1), 1-7.
3. Hoffman, A. F., Nolan, J., Gebhard, D. F., Nickischer, D., Omta, W., Cooper, S., ... & Fennell, M. (2018). Society of Biomolecular Imaging and Informatics High-Content Screening/High-Content Analysis Emerging Technologies in Biological Models, When and Why? *Assay and Drug Development Technologies*, 16(1), 1-6.
4. Omta, W.A., Nobel, J., Klumperman, J., Egan, D.A., Spruit, M.R., & Brinkhuis, J.S. (2017). Improving Comprehension Efficiency of High Content Screening Data Through Interactive Visualizations. *Assay and Drug Development Technologies*, 15(6), 247-256.
5. Omta, W.A., Heesbeen, R.G., Pagliero, R.J., Velden, L.M., Lelieveld, D., Nellen, M., Kramer, M., Yeong, M., Saedi, A.M., Medema, R.H., Spruit, M., Brinkkemper, S., Klumperman, J. & Egan, D.A. (2016). HC StratoMineR: A web-based tool for the rapid analysis of high content data sets. *Assay and Drug Development Technologies*, 14(8), 439-452.
6. Mauthe, M., Langereis, M., Jung, J., Zhou, X., Jones, A., Omta, W. A., Tooze, S., Stork, B., Paludan, S. R., Ahola, T., Egan, D.A., Behrends, C., Mokry, M., de Haan, C., van Kuppeveld, F. & Reggiori F. (2016). An ATG protein-specific siRNA screen reveals the extent of unconventional functions of the autophagy proteome in virus replication. *Journal of Cell Biology*, 214 (5): 619 – 635
7. Ulferts, R., Boer, M., van der Linden, L., Bauer, L., Lyoo, H., Mate, M., Lichiere, J., Canard, B., Lelieveld, D., Omta, W., Egan, D., Coutard, B. & van Kuppeveld F. (2016). Screening of a library of FDA-approved drugs identifies several

enterovirus replicaton inhibitors that target viral protein 2C. *Antimicrobial Agents and Chemotherapy*, AAC-02182.

8. Bijsmans, I. T., Guercini, C., Ramos, P. J., Omta, W., Milona, A., Lelieveld, D., ... & van Mil, S. W. (2015). The glucocorticoid mometasone furoate is a novel FXR ligand that decreases inflammatory but not metabolic gene expression. *Scientific reports - Nat.*, 5, 14086.
9. Lefebvre, A., Spruit, M., & Omta, W. (2015). Towards reusability of computational experiments: Capturing and sharing Research Objects from knowledge discovery processes. Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (pp. 456–462). KDIR 2015, November 12-14, Lisbon, Portugal: ScitePress.
10. Omta, W. A., Egan, D. A., Klumperman, J., Spruit, M. R., & Brinkkemper, S. (2013). HTS-IA: High Throughput Screening Information Architecture for Genomics. *International Journal of Healthcare Information Systems and Informatics* (IJHISI), 8(4), 17-31.
11. Omta, W. A., Egan, D. A., Spruit, M. R. & Brinkkemper, S. (2012). Information Architecture in High Throughput Screening., *Procedia Technology*, 5, 696-705.
12. Omta, W.A. (2011). Improving the usefulness and efficiency of a web-based prevention program for screening chronic diseases in the general population., *Utrecht University*, *Master (doctoraal) thesis*, IKU-3481824.

Curriculum Vitae

Wienand Omta was born in 1983. After finishing the MAVO, he did an MBO, HBO premaster and Master of Science in System Engineering, Computer Science, Information Science and Medical Informatics, respectively. He started his PhD research at the University Medical Center Utrecht under supervision of Sjaak Brinkkemper, Judith Kumperman and Marco Spruit. He worked in the lab of dr. David Egan to optimize the efficiency of High Content Screening. During his PhD work, they developed a product called HC StratoMineR which was commercialized in their software company Core Life Analytics B.V. in 2016 and is now being used by multiple top pharmaceutical companies worldwide. Wienand will continue his work as the CTO of Core Life Analytics. You can find more information at www.wienand.nl