



# Core Life Analytics

## StratoVieweR, connecting High Content Screen images with feature data

The development and validation of StratoVieweR as a module to the StratoMineR platform

### BAFSTU Report

Supervisor at Core Life Analytics: Wienand Omta PhD

Supervisor UAS Leiden: Marc Besseling PhD

Start of internship: 01-September-2021

End of internship: 31-August-2022

Version 1

Pieter Schreurs

10-05-2022

Hogeschool Leiden, s1105750



## Abstract

The fields of pharmacology, biotechnology and chemistry are becoming increasingly reliant on high throughput screening (HTS). HTS is a method to perform systematic large numbers of automated biochemical experiments. A newer development of HTS is high content screening (HCS). In HCS experiments, images are taken from live or fixed cells to see phenotypic changes after admission of a biochemical reagent.<sup>2,4</sup> The scale in which images are created using HCS makes it implausible to manually review all these images.<sup>6</sup> Using a method named cellpainting, numeric datasets are generated describing the information of the images.<sup>7</sup> Big data analyses platforms like StratoMineR, Tibco Spotfire and Dotmatics exist to analyse these datasets.<sup>8</sup> Whilst numeric data analysis is proven to create credible results, we hypothesise that higher quality results can be achieved by reuniting the images with numeric data.<sup>18</sup> To test this hypothesis we have asked the following questions: (MRQ) *“What is the value of uniting HCS image data with numeric data?”* (sq1) *“Does this connection aid in curating labels, eliminating extreme outliers thus increasing the quality of training data?”* (sq2) *“Can this connection add value to the verification and confirmation process of promising hits?”*

To create a union between the numeric and image data, the module StratoVieweR is developed as an extension of StratoMineR. This module enables users to view images selected through other modules in the StratoMineR platform. It also features automatic channel scaling for comparison, complete plate overviews with thumbnails and control over individual channels. StratoVieweR is compared to its closest competitor, Iviewer.<sup>10</sup> The comparison shows that StratoVieweR has similar features as Iviewer but is superior for the comparison of images. StratoVieweR is also shown to benefit from the integration into the StratoMineR platform which enables the selection of images to view from the results of a data analysis.

To validate StratoVieweR, an experiment is executed in which a supervised clustering analysis is performed 30 times over a full dataset and a dataset that has been curated with the aid of StratoVieweR. The model is instructed to predict microtubule stabilisers in the Caie dataset.<sup>9</sup> Analysing the variance over the 30 runs through a one-way anova analysis indicates a growth in accuracy when training a model using curated data. The prediction model classifies items as microtubule stabilisers where the original class is actually DNA damage. Comparing images of the focus class with images of DNA damage, it became evident that wells treated with epothilone B (Microtubule stabilisers) showed a secondary target of DNA-damage. This observation is reaffirmed by Rogalska et al. (2015) and Poruchynsky et al. (2015).<sup>25-27</sup> To conclude it is confirmed that uniting HCS image data with numeric data, aids in the curation of labels and eliminating extreme outliers, resulting in better predictive models. By evaluating the results of those models using the image data and features of StratoVieweR, more insight is given into the MOA of promising hits.

# Table of Contents

<b>1 Document Overview</b>	<b>6</b>
1.1 Abbreviations	6
1.2 List of images	7
1.3 List of tables	8
<b>2 Introduction</b>	<b>9</b>
2.1 Core Life Analytics	9
2.2 High Throughput Screening	9
2.2.1 Controls	10
2.2.2 Replicates	10
2.2.3 Screening	11
2.3 High Content Screening	11
2.3.1 HCS workflow	11
2.3.2 Image data	12
2.3.3 Numeric data	13
2.3.4 Analysing HCS data	13
2.3.4.1 Extracting features from HCS images	13
2.3.4.2 Data Mining	14
2.3.4.3 Result validation and identification of errors	15
2.4 Caie dataset	18
2.5 Amazon Web Services	18
2.6 Shiny	19
2.7 Project aim	20
2.8 Project overview	20
<b>3 Materials and Methods</b>	<b>22</b>
3.1.1 Materials for StratoVieweR	23
3.1.2 Structure of StratoVieweR module	24
3.1.3 Connecting StratoVieweR	24
3.1.3.1 Database	24
3.1.3.2 MetaData image-connection	25
3.1.3.2.1 FST-file type	25
3.1.3.3 MetaData module-connection	25
3.1.3.3.1 RDS-file type	25
3.1.4 Interface of StratoVieweR	25
3.1.4.1 Buckets	25
3.1.4.2 Image settings	27
3.1.5 Benchmarking of image downsampling	28
3.2 Data analysis aided by StratoVieweR	29
3.2.1 Experimental Design	29
3.2.1.1 Defined variables	30

3.2.1.2 Training and Test -set	31
3.2.1.3 Stratified sampling	32
3.2.2 Materials for validation	32
3.2.2.1 Numeric Features	32
3.2.2.2 Ground truth	34
3.2.3 Data Analysis	36
3.2.3.1 Feature Selection	36
3.2.3.2 Select controls	36
3.2.3.3 Plate normalisation	37
3.2.3.4 Data transformation	37
3.2.3.5 Feature scaling	38
3.2.3.6 Missing data	38
3.2.3.7 Image Curation using StratoVieweR	38
3.2.3.8 Dimensionality reduction	40
3.2.3.9 Hit selection	41
3.2.3.9.1 Basic AI settings	41
3.2.3.9.2 Select predictors	43
3.2.3.9.3 Advanced AI settings	44
<b>4 Results</b>	<b>45</b>
4.1 StratoVieweR	45
4.1.1 Database	45
4.1.2 MetaData files	46
4.1.3 User interaction design	47
4.1.3.1 Plate wise	47
4.1.3.2 Iterative data mining	48
4.1.3.3 Bucket comparison	49
4.1.4 Benchmarking	51
4.2 Data Analysis	54
4.2.1 Plate normalisation	54
4.2.2 Data transformation	55
4.2.3 Scaling	56
4.2.4 Missing data	56
4.2.5 Image curation	57
4.2.6 Dimensional reduction	58
4.2.7 Hit Selection	59
4.2.8 Model	63
<b>5 Discussion and conclusion</b>	<b>65</b>
5.1 Experimental design	65
5.2 StratoVieweR	65
5.2.2 Comparison to Omero Iviewer	65
5.2.2.1 Platemap	66

5.2.2.2 Image Viewer	68
5.2.2.3 Comparing images	70
5.2.3 Reviewing outliers	71
5.3 Data Analysis	73
5.3.1 Accuracy scores, ANOVA	73
5.3.2 Specificity	75
5.3.3 Overfitting	76
5.3.4 Agreement score	77
5.3.5 Predicted classes	77
5.4 Conclusion	79
5.5 Future research	79
<b>References</b>	<b>81</b>

# 1 Document Overview

The domain of High-content screening, methods and tools for HCS data analysis are introduced in the introduction. At the end of the introduction the project aim is defined. The materials and methods chapter, results and discussion of this report shall be split between the development of StratoVieweR and a data analysis experiment designed to validate StratoVieweR. The section development of StratoVieweR should be read as a consideration of methods to achieve the goals and not as a guide to replicate the development. The method described for the data analysis is encouraged to be replicated on other datasets.

## 1.1 Abbreviations

AWS	Amazon Web Services
AZ	Astrazeneca
CLA	Core Life Analytics
CNN	Convolutional Neural Network
CSS	Cascading Style Sheet
ERD	Entity relations diagram
HCS	High Content Screening
HTS	High Throughput Screening
MOA	Mechanism of Action
NGS	Next Generation Sequencing
PCA	Principal Component Analysis
QC	Quality Control
S3	Simple Storage Solution (AWS)
SAAS	Software as a Service
SD	Standard Deviation

## 1.2 List of images

### Introduction

Figure 1.	Image of microplates	10
Figure 2.	Triplicate experiment	11
Figure 3.	Channels in cell imaging	12
Figure 4.	Cellpainting process	13
Figure 5.	HCS data analysis workflow	14
Figure 6.	Plate normalisation example	15
Figure 7.	Examples of image artefacts	16
Figure 8.	384-well Microplate in Iviewer	17
Figure 9.	Iviewer	17
Figure 10.	Reactive programming schematic	19
Figure 11.	Project overview	20

### Materials and methods

Figure 12.	Workflow for StratoVieweR	22
Figure 13.	File structure of StratoVieweR	24
Figure 14.	Design of StratoVieweR, plate view	26
Figure 15.	Design of StratoVieweR, comparison view	27
Figure 16.	Workflow of experiment	30
Figure 17.	Relations between variables	31
Figure 18.	Stratified sampling	32
Figure 19.	Define controls, settings	37
Figure 20.	Plate normalisation, settings	37
Figure 21.	Data Transformation, settings	38
Figure 22.	Feature scaling, settings	38
Figure 23.	QC, settings and plot	39
Figure 24.	StratoVieweR platemap	39
Figure 25.	StratoVieweR comparison	40
Figure 26.	Dimensional Reduction, settings	41
Figure 27.	Hit Selection, supervised settings	42
Figure 28.	Hit Selection, predictor settings	43
Figure 29.	Hit Selection, advanced settings	44

### Results

Figure 30.	ERD of StratoMineR database	45
Figure 31.	Added tables to the StratoMineR database	46
Figure 32.	Platemap Settings, StratoVieweR	48
Figure 33.	Platemap with and without thumbnails, StratoVieweR	48
Figure 34.	Iterative data mining, StratoVieweR workflow	49
Figure 35.	Comparisons of wells, StratoVieweR	50
Figure 36.	Downsampling methods	51
Figure 37.	Downsampling benchmark	52
Figure 38.	Downsampling and plotting benchmark	53
Figure 39.	Plate normalisation, before normalisation	54
Figure 40.	Plate Normalisation, after normalisation	55

Figure 41.	Skewness of features, before transformation	55
Figure 42.	Skewness of features, after transformation	55
Figure 43.	Features, before scaling	56
Figure 44.	Features, after scaling	56
Figure 45.	Missing data	56
Figure 46.	Curated images	57
Figure 47.	Scree plot of principal components	58
Figure 48.	Correlation matrix of features	59
Figure 49.	Sampling distributions	59
Figure 50.	Contour plot of predicted classes	60
Figure 51.	Bar plot of relative importance of phenotypic distance	61
Figure 52.	Replicate outliers	62
Figure 53.	Phenotypic distance	63
Figure 54.	Quality scores of models	64

### **Discussion and conclusion**

Figure 55.	Platemap in Iviewer	66
Figure 56.	Loading times of Iviewer and StratoVieweR	67
Figure 57.	Settings and info in Iviewer	68
Figure 58.	Settings and info in StratoVieweR	69
Figure 59.	Image comparison in StratoVieweR	71
Figure 60.	Systematic outlier plot	72
Figure 61.	Images of outliers	72
Figure 62.	Accuracy of model, boxplot	73
Figure 63.	Accuracy of models, Histogram	74
Figure 64.	Specificity of models, boxplot	76
Figure 65.	95% confidence interval of models	77
Figure 66.	Comparison of Microtubule Stabilisers and DNA damage	78

## 1.3 List of tables

### **Materials and methods**

Table 1.	Resources for StratoVieweR	23
Table 2.	Variables in experiment	31
Table 3.	Resources for experiment	32
Table 4.	Settings for CellProfiler	34
Table 5.	Compounds and MOA Caie experiment	35-36

### **Results**

Table 6.	Image metadata file	47
Table 7.	Communication between modules file	47
Table 8.	Mean accuracy of models and SD	64
Table 9.	Predicted compounds	64

### **Discussion and conclusion**

Table 10.	Shapiro-Wilk normality test	74
Table 11.	One way ANOVA test	75

## 2 Introduction

### 2.1 Core Life Analytics

'Core Life Analytics B.V. is a spin-off of the University Medical Centre Utrecht, founded at the end of 2016. Over the course of 5 years in a screening lab, software was developed designed for biologists to support them with the analysis of High Content Screening data. At the end of 2016, an exclusive licence deal was signed with the university for offering the software platform called StratoMineR in a commercial fashion.

Core Life Analytics (CLA) is a profitable data analytics company based in Den Bosch, The Netherlands. CLA currently sells software as a service (SaaS) subscriptions for the StratoMineR data analytics platform used by pharmaceutical companies, biotech's and academic life sciences groups. Customers include Pfizer, Galapagos, University of Oxford, University of Cambridge, and King's College London.' <sup>1</sup>

### 2.2 High Throughput Screening

High Throughput Screening (HTS) is a method to perform systematic large amounts of automated biochemical experiments. Many different subtypes of HTS experiments are performed in the fields of pharmacology, biotechnology and chemistry. These subtypes of HTS include but are not limited to experiments such as, High throughput sequencing, DNA microarrays and metabolomics. In High throughput sequencing, also referred to as Next generation sequencing (NGS), the composition of nucleotides in the DNA are 'read' and converted to an electric signal that is readable by a computer. The conversion from a chemical process to an electronic signal is often done via bioluminescent probes. A bioluminescent probe is selected that highlights during a specific part of the chemical process. A camera registers the light emitted by the probe and creates a signal then measured by the computer. This principle can be applied to all different types of chemical processes. In drug discovery libraries of potential drugs are screened. These libraries of compounds can consist of chemicals, antibodies, or any reagent expected to give a bioactive response. Well microplates are prepared and scanned using automated microscopy. Well microplates are available with different well counts, 96, 384 and 1536 are the most common, see figure 1.<sup>2-5</sup>

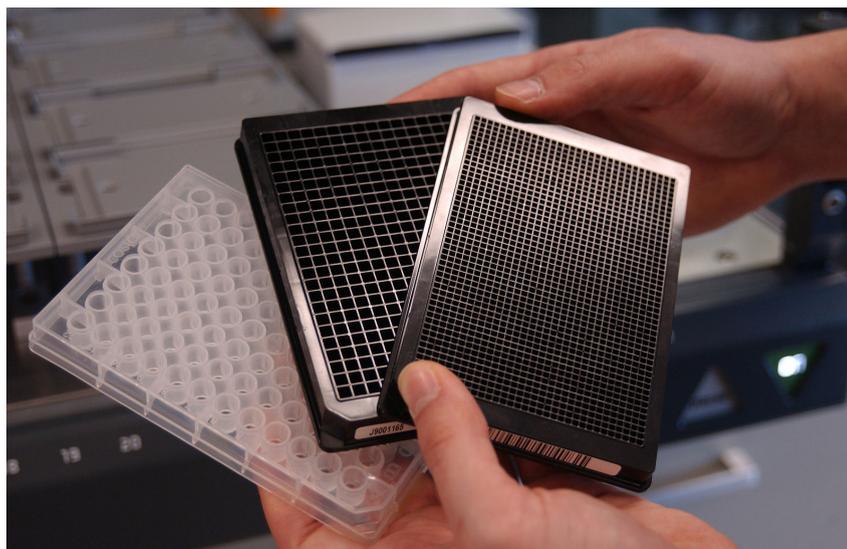


Figure 1. Picture of three well microplates with well counts of, 96, 384 and 1536. The microplates are of the same size, the well size changes depending on the amount.<sup>3</sup>

HTS gives biologists the possibility to screen up to millions of reagents. By implementing HTS it becomes faster to find or develop compounds that affect biological processes. This information is used in drug discovery to quickly filter a library containing millions of compounds, and find the compounds in that library for further research or clinical trials. The concepts introduced below will explain the methods used to generate good quality results in an HTS experiment.

### 2.2.1 Controls

To increase the likelihood of meaningful results from an experiment, controls are implemented. In a microwell plate experiment not all wells are used for screening reagents from a library. Frequently, tens of wells are selected to implement controls in the assay. The controls are well-known reagents that have been studied in the past and are more likely to respond in a certain way. Controls are used to perform quality control (QC), normalisation or hit picking. By using the controls in each plate, a higher confidence in the results can be achieved. When performing QC, one can signify the difference between the positive and negative controls and look for variation between plates if the experiment comprises multiple plates. The control experiments are not only used for QC but can also be used to train a classification model or for calculating distance scores and similarity. QC should be performed on processed data to avoid overlooking artefacts generated by the processing of the results.<sup>4,5</sup>

### 2.2.2 Replicates

Technical or biological replicates are used to increase the trustworthiness and reproducibility of the experiment. This redundancy is called duplicate assay, triplicate assay, et cetera. Replicates can be created in two ways: a technical replicate in which the complete microwell plate is multiplied, or a biological replicate in which the experiment is multiplied over wells of the same plate. A technical or a combination of technical and biological replication is preferred over just a biological replicate. Technical replicates are usually more costly. Figure 2 depicts how technical triplicate assay is prepared. Important to note is that reagents are unlike the schematic display in figure 2, randomised over the plate to avoid bleeding into other wells. As an example, if an assay is conducted using four cell lines containing 100 reagents at eight different concentrations screened in triplicate, excluding the control wells, 9600 wells are needed.<sup>4,5</sup>

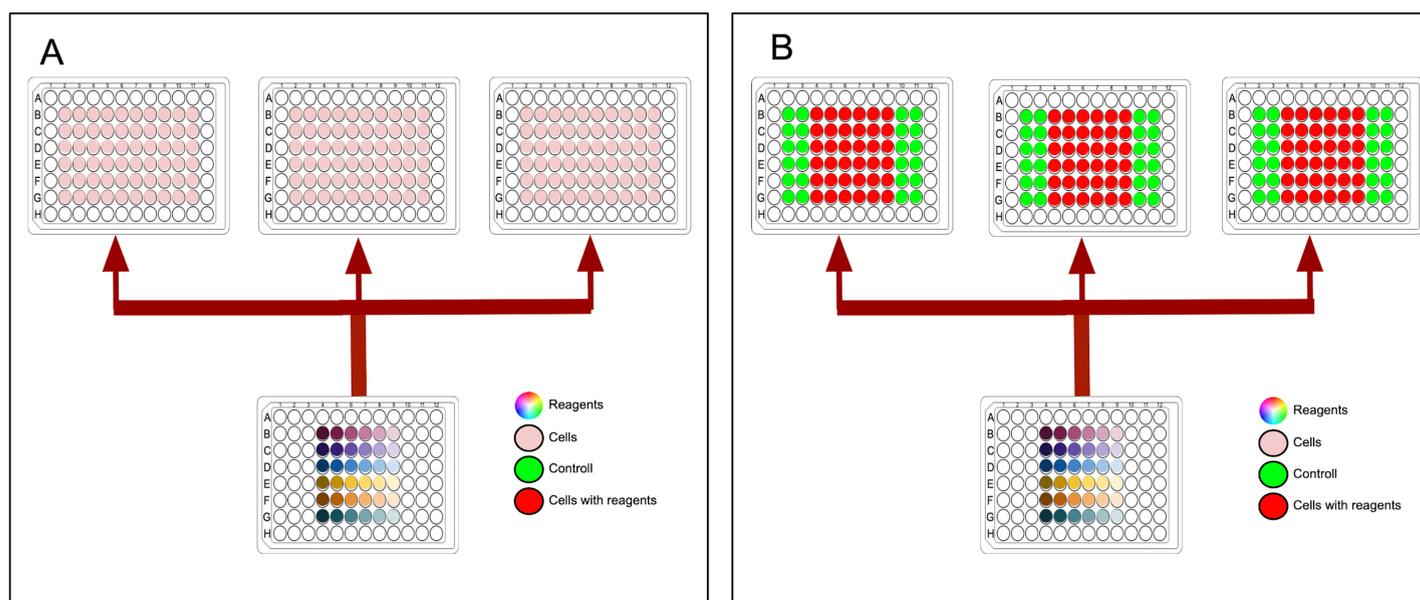


Figure 2. A) Displays step one of a triplicate experiment, reagents are placed in one well microplate and three plates are primed with just cells. B) Reagents are divided into the three well microplates that were primed with the cell line.

### 2.2.3 Screening

In drug development multiple HTS are done. Drug development comprises four steps: drug discovery, pre-clinical development, clinical development and drug agency approval. The first step, 'drug discovery' starts off with target identification and validation. An HTS experiment is performed where a vast library of reagents are tested against a library of proteins. This enables the drug researcher to make a fast estimation on the effectiveness of the drug on its target protein. Pharmacophore modelling and an excellent knowledge of genomics has enabled drug researchers to create many targets. This way one can generate 100.000 assays per day, but can not screen for the compounding interactions in a life or fixed cell. A High Content Screening (HCS) can show us these interactions but at a far lower throughput. The results of the HTS are used to filter the large library in preparation of a lower throughput HCS experiment.<sup>5,6</sup>

## 2.3 High Content Screening

HCS is an HTS method using imaging to see phenotypic changes in live or fixed cells after admission of a compound. More complex responses to compounds and phenotypic changes in complex biological systems such as cells or even complete organisms with very specific traits eg. zebrafish ie. transparent when embryonic, can be measured. Cell biologists used to look at phenotypic changes in these systems per each individual experiment via a microscope. By applying HCS techniques to experiments on complex biological systems a vast amount of images are generated. It thus becomes impractical and uneconomical for biologists to manually assess every image.<sup>5</sup>

### 2.3.1 HCS workflow

An HCS experiment consists of multiple steps. Starting with robots that can perform the automatic liquid handling and pipetting of liquids into micro-well plates. Robots are more precise and consistent than even the best trained lab technicians. The robots can also mix fluids to achieve wells with different

concentrations of the compounds. All cells are marked with luminescent probes. These probes help identify structures of the cells. A wide variety of probes is available to highlight all unique structures of the cells, most common is a probe for the cell's nuclei. This probe is used to detect and count cells that contain a nucleus, most often this is the basis of any automated images analysis protocol. After the libraries have been prepared, an automated microscope takes images of the well using a digital camera. <sup>5</sup>

### 2.3.2 Image data

For each experiment in a unique well, multiple images are taken, resulting in multiple images of different areas of the well. By increasing the magnification of the objective of the camera, a smaller part of the well is captured but in more detail. To still cover the entire area of the well more images are taken. The higher the number of fields per well the higher the resolution. The cameras used typically produce images with a pixel depth of 16 bit. Giving more pixel depth than traditional 8-bit images. In 8-bit imaging there is a pixel depth of  $2^8 = 256$  values whilst 16 bits gives  $2^{16} = 65536$  values for each pixel. Per luminescent probe used in the experiment an image is made for the specific wavelength, this image is called a channel. In an experiment with six different probes this results in six channels per field. Three of these six channels can be combined by assigning red, green and blue to the values and creating a single red green and blue (RGB) image. Figure 3 shows images from an HCS including three channels displayed separately and combined.<sup>4</sup>

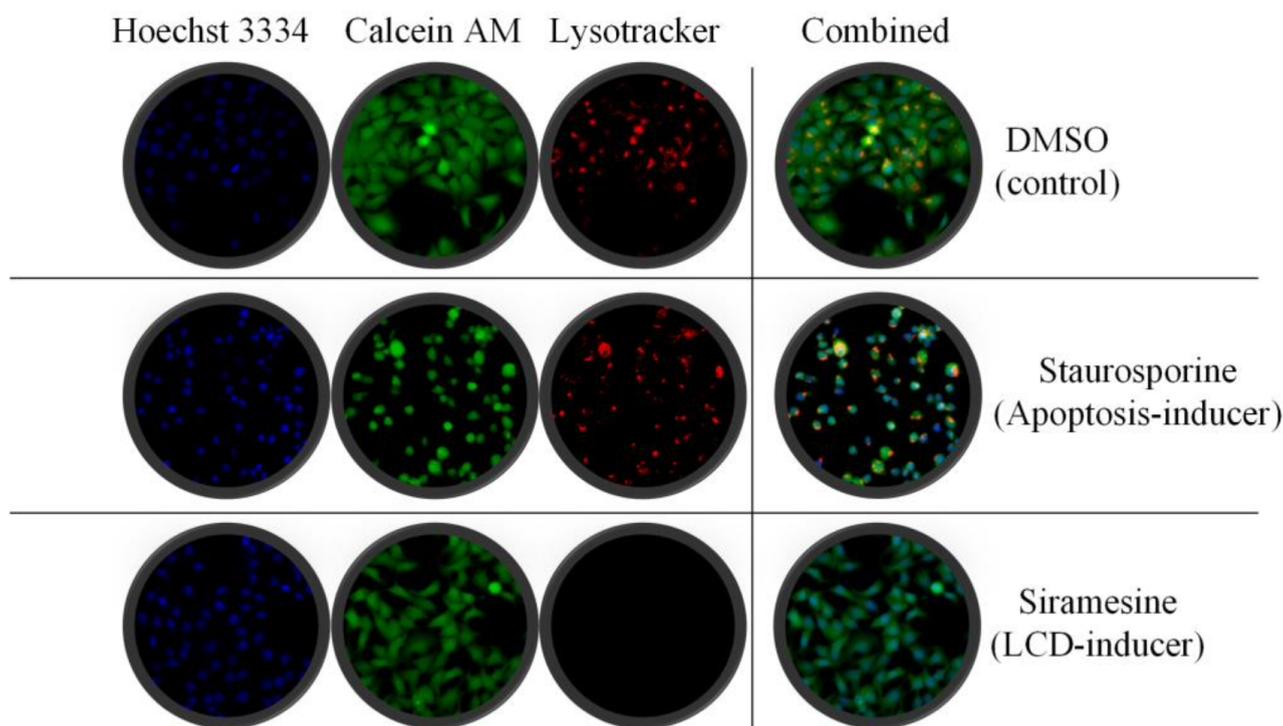


Figure 3. This figure shows images of three wells, the three images are shown as three separated channels and a combined image. The three channels represent the cell nucleus in blue, calcein in green and lysosomes in red, from left to right respectively. The three wells are control in row one, row two where an apoptosis inducer is added, the cells are smaller and brighter. Siramesine (Row three) kills cells via a different pathway and results in no lysosomes (red) and ellipse shaped cells. figure by Omta et al <sup>5</sup>

### 2.3.3 Numeric data

Measuring intensity of different wavelengths reflected by a luminescent probe, an image analysis software package creates an image layer to visualise unique structures in cells. During image analysis, numeric features can be extracted describing the size, area, intensity and texture properties of individual cells, inferring statistics of the phenotype. These properties or features can be calculated from each fluorescent dye or even combinations, i.e. the number of objects A in object B and can result in a very rich data set containing thousands of features.<sup>5</sup>

### 2.3.4 Analysing HCS data

The automation in biological research creates vast amounts of data. Where it was still a possibility to use basic statistics for analysing the results of classic HTS experiments, HCS experiments generate an even larger amount of data that require more processing to create meaningful results. To analyse the HCS results, images have to be converted to a numerical dataset consisting of meaningful features. These features are then mined and analysed. Results are visualised for biologists using different plots and graphics. Looking at the original images based on the results of the data analysis can help the biologist better understand the results of the analysis.

#### 2.3.4.1 Extracting features from HCS images

Features are extracted from HCS images using a method called cellpainting. Image analysis is frequently carried out by commercial tools supplied as proprietary software by the manufacturer of the automated microscope. Some open-source tools are also available such as cellProfiler. To derive features from images an image analysis server needs to identify the structures in the image. Using the different channels, masks are generated. These masks imply borders of the objects that were found using the algorithm that the image analysis software is executing. Figure 4 demonstrates the result of this process. The user has to configure on what intensity the program uses to decide if a pixel is part of the structure or the background. The masks are then used to generate features. These features are stored in a grid where each row is a well or field and each column represents a feature.<sup>5,7</sup>

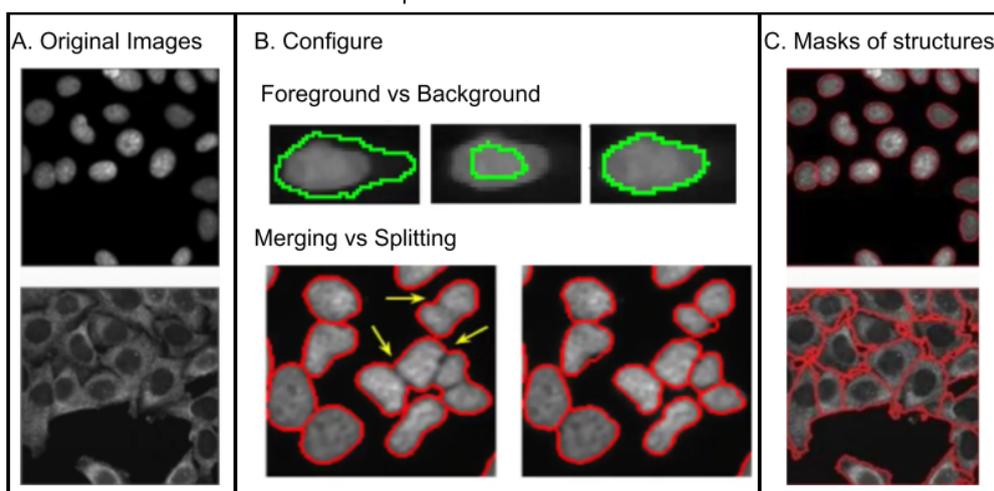


Figure 4. Overview of cellpainting process. A) original channels used for profiling. B) settings that the user has to set. By selecting the correct values for foreground and background, one can avoid multiple structures merging, and structures being ignored. C) results of the cellpainting process, a mask is drawn on top of the images.

### 2.3.4.2 Data Mining

Creating insight into the large and unstructured dataset created by image analysis of an HCS experiment is done via data mining. A data mining pipeline can be developed by a bioinformatician for each project. There are also multiple software packages available for this purpose. Data analysis tools like Spotfire, HC StratoMineR and Gene Data are developed to analyse HCS datasets. These packages are all available commercially. Explaining the process of data mining an HCS dataset will be done on the basis of HC StratoMineR developed by CLA. In figure 5 the basic workflow of HC StratoMineR is shown.<sup>8</sup>

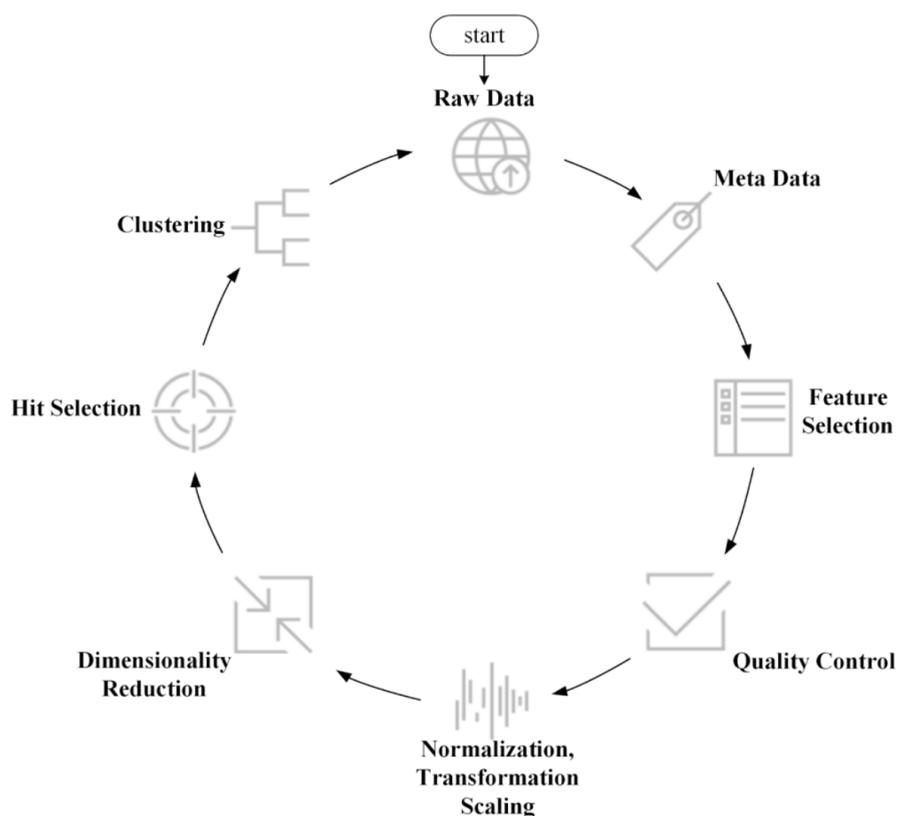


Figure 5. Workflow of the steps taken in an HCS data analysis process, figure by Omta et al.<sup>5</sup>

HC StratoMineR is a web-based data analysis platform which enables users to mine data from an HCS experiment. The StratoMineR platform comprises different applications that are connected within an easy-to-use interface. The following steps are taken to execute the data analyses of an HCS experiment using the StratoMineR platform. Uploading of data, the raw images generated by the automated microscopy are uploaded to the web server. These are stored in an Amazon Web Services (AWS) S3 bucket. CellProfiler Ultra takes batches of the images from the S3 bucket to use the process of cellpainting to identify the structures within the cell and generate features. It is also possible to apply a Convolutional Neural Network (CNN) to generate features using StratoNet. The generated features are then filtered based on variance and counts of missing data within the feature. Multiple visualisations are shown to the user as a means of QC. After the user has checked the plots for the QC, plate normalisation is executed, see figure 6.<sup>8</sup>

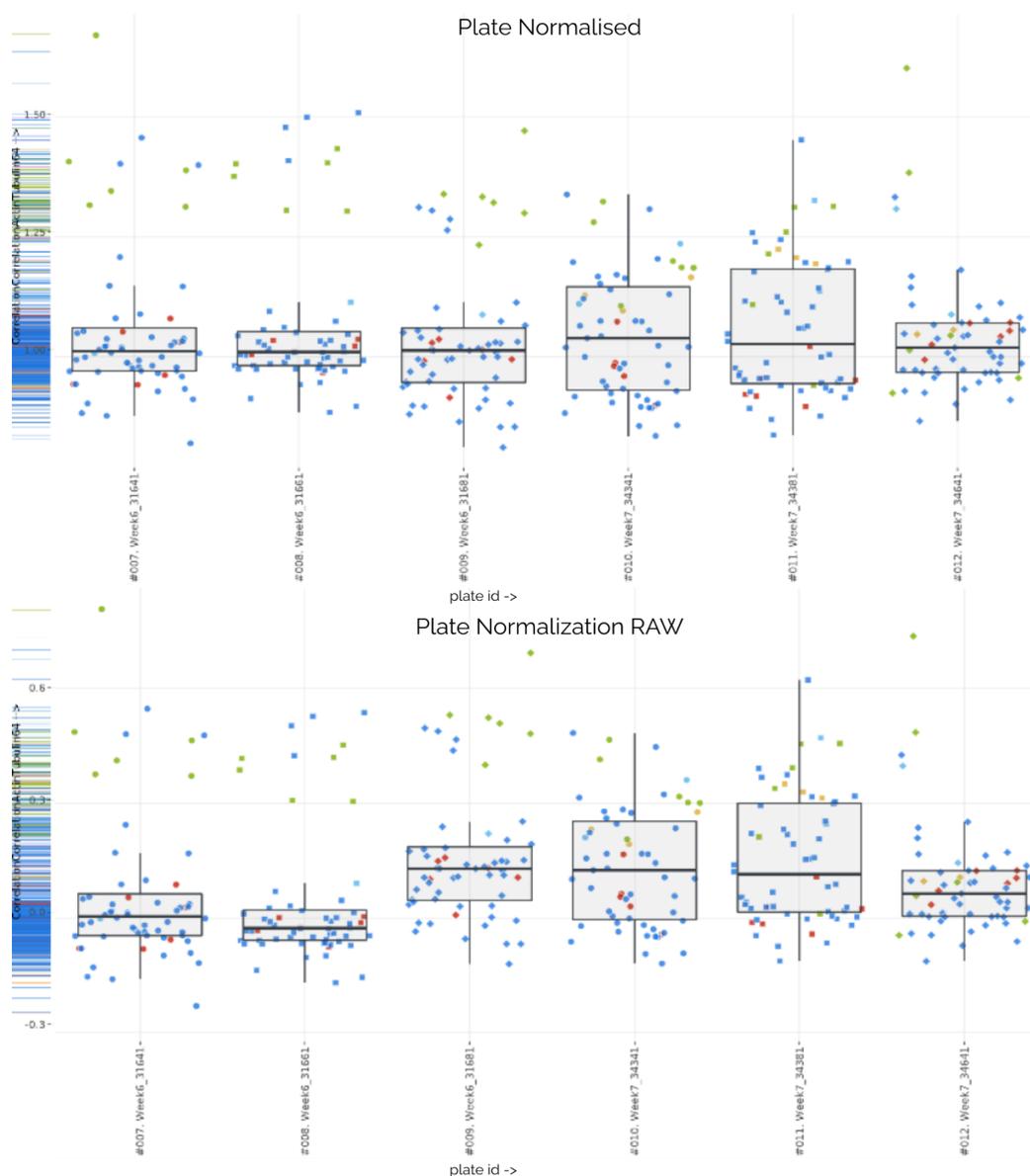


Figure 6. Normalisation of plates in HC StratoMineR. Variation between plates is shown in the lower plot. This has been normalised against the median of the negative control. Normalised plates are shown in the upper plot.

To reduce redundancy, computational load and highlighting relevant features, a dimensionality reduction method can be applied such as Principal Component Analysis (PCA). The software takes the normalised and reduced data in dimensionality and tries to identify hits by means of applying a Euclidean distance for each reagent against an implemented control. Clustering the hits gives insight into different groups of compounds. HC StratoMineR can then also be used to calculate the dose response per hit. All these steps can be exported using the automated report generator.<sup>8</sup>

### 2.3.4.3 Result validation and identification of errors

To validate the results of a carried out data analysis using HCS data, a biologist might want to review the images captured by the automated microscope used to generate the dataset. To compare the images truthfully, intensity levels of the different channels require to be adapted. By looking at the images, a biologist can identify or verify systematic errors, inconsistency of replicates, extreme outliers, interaction

of reagent with probe, and plate or batch effects. Figure 7 shows common errors that are easily identified but might trigger data analysis software to identify that image as a hit.<sup>4</sup>



Figure 7. Examples of image artefacts in the Caie dataset.<sup>9</sup> From left to right: Image one shows a bright red streak, this error can result in wrongly classified features. Image two shows a technical error where the cells were not placed correctly in the microwell, this has resulted in the field being empty, resulting in a far overexposed image. Image three shows a red haze in the right bottom corner, this is the result of a lens flare, the light has reflected inside the lens in such a way that the image is not a representation of what happened inside the well.

Systematic errors are errors resulting from experiment conditions and procedures. Systematic errors appear in all stages of an HCS experiment such as library preparation and can manifest on batch, plate and well level. Sources of systematic errors can be: reagent, either interaction with the target compound or changes in concentration due to reagent evaporation; liquid handling, not optimal robotic operation or pipette malfunction. Variations in external factors can also result in systematic errors (i.e., variation in lighting, airflow, temperature, et cetera).<sup>4</sup>

A software suite that enables biologists to review images captured by the automated microscope is OMERO by openmicroscopy. OMERO displays an overview of a complete well microplate showing the first field of the wells, figure 8. Through an index, the user can change which frame is displayed as the thumbnail for each well. This view makes it effortless for the user to spot obvious problems visible in the images, clusters of wells of the plate as a whole. In figure 8, one can spot a large blue smear in well M1 field #8.

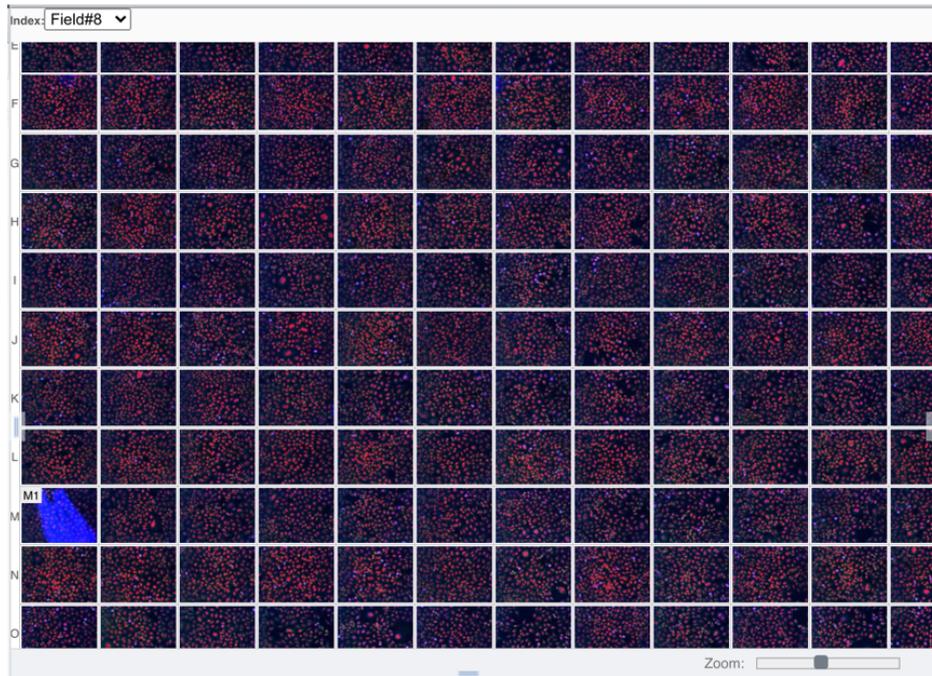


Figure 8. Overview of 384-well microplates in Iviewer, the grid displays a single field per well. The coordinate system shown left M,N,O et cetera display the locations of the images. dataset: idr0056-stojic-Incrnas/screenB, tray 1977

When a well is selected, a second view depicts all frames of that well. In the view, the user has control over the images and can query additional information on the images, see figure 9. In this view, the user can select each frame to be enlarged. It is not possible to view all the frames merged together. In the information panel, the well identifier and channels are given. The settings panel gives users control over the different layers. Specific layers can be switched off, or the intensity of the layer can be changed. When, via the browser the source image is requested, one can see that the image is stored directly as base64 in the database.<sup>8</sup>

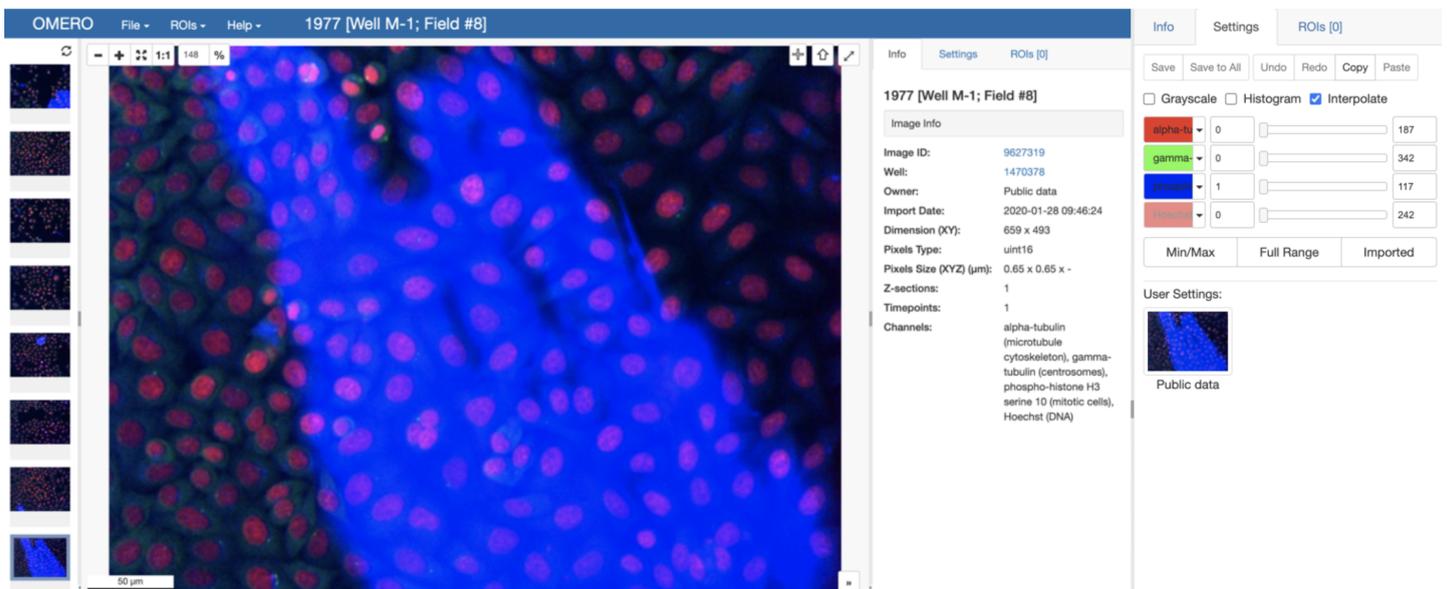


Figure 9. Overview of HCS well in Iviewer. The vertically stacked images on the left display the different fields. The large image shows the selected field. The settings on the right only affect the main field. The sliders in the settings tab change the opacity of the different channels. dataset: idr0056-stojic-Incrnas/screenB, tray 1977, Field #8

Omero offers its user a suite of tools for reviewing HCS images. Omero is an application that due to its universality and some design flaws responds slowly. Displaying the overview of all the images takes a considerable amount of time. This is because the images are all stored as binary format in a postgres database. This is not considered good practice because of the sizable amount of data or the potential cumulative data that will be stored in the future, slowing down the database. It also limits the expandability of the tool.

Currently, Omero does not allow integration with other data analysis platforms because it lacks functionality towards interoperability such as an API. Because of the disconnectedness with other software, it would be difficult to integrate external information such as the class of the well which is not held by the software package Omero. Additionally, Omero does not support the ability to easily display multiple images side by side, this makes it difficult for a cell biologist to validate patterns in the images found by the data analysis software.

A system that would connect the inspection of image data and numeric data in the field of HCS does not currently exist. A tool that would offer this functionality could drastically improve the efficiency in which biologists can identify or verify systematic errors, inconsistency of replicates, extreme outliers, interactions of reagents with probes, and plate or batch effects. It could potentially also help biologists confirm promising hits from their data analysis platform.

## 2.4 Caie dataset

To show the validity of HCS in pharmacological research, researchers at the University of Edinburgh have executed an HCS of 102 well documented drugs and inhibitors in four cancer cell-lines. The targets were added in eight half-log doses, in 96-well assay plates screened in triplicates. Fluorescent probes were added to illuminate the actin cytoskeleton, microtubule and DNA. For this project, a subset of these images comprising a single cell-line have been extracted. These images are stored in an AWS Simple Storage Solution (S3) bucket. The name of the file describes all the necessary metadata; following this pattern: "Week#\_Plate#\_Well#\_Field#\_Channel#". These images can be directly accessed via R using the library "aws.s3".<sup>9</sup>

The images of the Caie dataset show treated breast cancer cells. Each micro-well of cells was treated for 24 hours with one of 113 molecules at eight concentrations. The cells were imaged by fluorescent microscopy using an automated ImageXpress 5000A by Molecular Devices. A 16-bit camera was used to image four fields per well. The images are stored in the tif format. The tiff image format supports higher bit depth than image formats such as PNG, BMP without compression. The images are of 3300 wells in grey scale, this multiplied by the three channels, multiplied by 4 fields results in a dataset of 39,600 images. Each image is ~2.5 Megabytes resulting in a dataset of ~100 Gigabytes.<sup>10</sup>

## 2.5 Amazon Web Services

AWS is a cloud platform. It offers unique features, including cloud storage and computing. The StratoMineR platform uses the computing features and the storage solutions offered by AWS. AWS makes it possible for all users of HC StratoMineR to work in a private cloud environment. This ensures security for the users and offers the benefit that customers do not need to invest in expensive IT infrastructure to execute HCS analyses.<sup>11</sup>

## 2.6 Shiny

For the development of HC StratoMineR the CLA team has chosen Shiny. Shiny is a package for the R programming language that is used to develop interactive applications using reactive programming methodology. Shiny changes how you program an R script. Commonly, R scripts are linear in fashion, whilst Shiny offers the possibility to execute the server logic with reactive programming. In reactive programming, logic is not executed based on the order that the programmer requests it but by order of a designed dependency after the input data changes. If the example shown in figure 10 was programmed in a script-like way, every time one input value changes, all four calculations have to be executed. When the example is programmed reactively, only the first time a result is requested will those four calculations be executed. If input #2 changes, this is a call to recalculate the result, thus only executing calculation 3 and 4. If input #3 where to change, this would only re-execute calculation 4. This elimination of duplicate calculations generates an efficiency bonus. A reactive program loads data into memory only once. The program can then manipulate and visualise this data indefinitely. Thus also limiting potential loading time.

12

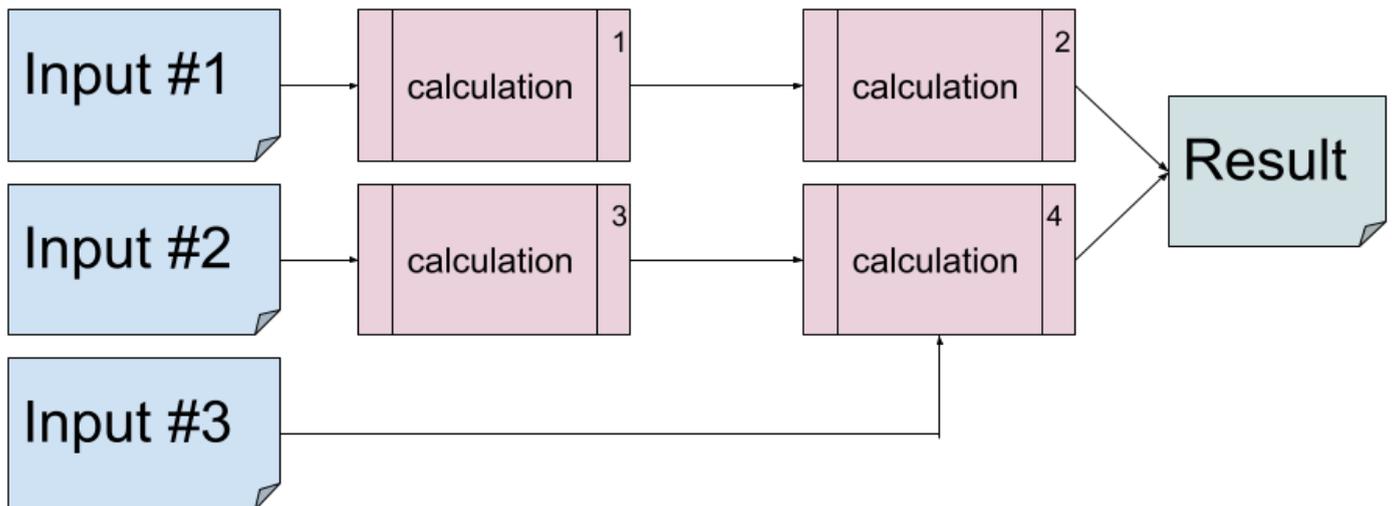


Figure 10. Schematic of an example program to demonstrate order of operations in a reactive program versus a script like program. In a script like program steps are executed sequentially only shown as input #1 and calculation 1 and 2. A reactive program responds to inputs to only perform the calculations that are influenced by the changed inputs, if input #3 changes only calculation 4 is executed whereas in a script all calculations have to be re-executed.

When run, shiny generates a html page. Shiny is built on JavaScript and R. The design of the interface can easily be edited via a Cascading Style Sheet (CSS). Extra functionality not available in R can be added via JavaScript.

## 2.7 Project aim

The aim of this project is to develop a module for the StratoMineR platform to connect HCS image and HCS numeric data. To validate the effectiveness of the StratoVieweR module, research will be conducted using a single cell line of the Caie dataset. The experiment shall compare the accuracy of clusters created using only numerical data analysis with the accuracy of clusters that are manually validated via StratoVieweR. This research implies the comparison of the AS-IS versus the TO-BE situation, given this context, the following research question is derived:

MRQ: What is the value of uniting HCS image data with numeric data?

SQ1: Does this connection aid in curating labels, eliminating extreme outliers thus increasing the quality of training data?

SQ2: Can this connection add value to the verification and confirmation process of promising hits?

## 2.8 Project overview

The development of StratoVieweR and the validation of its effectiveness was done via the workflow as shown in figure 11. A reiterative process of development was chosen so that the module could be easily evaluated as a complete system during the process of development.

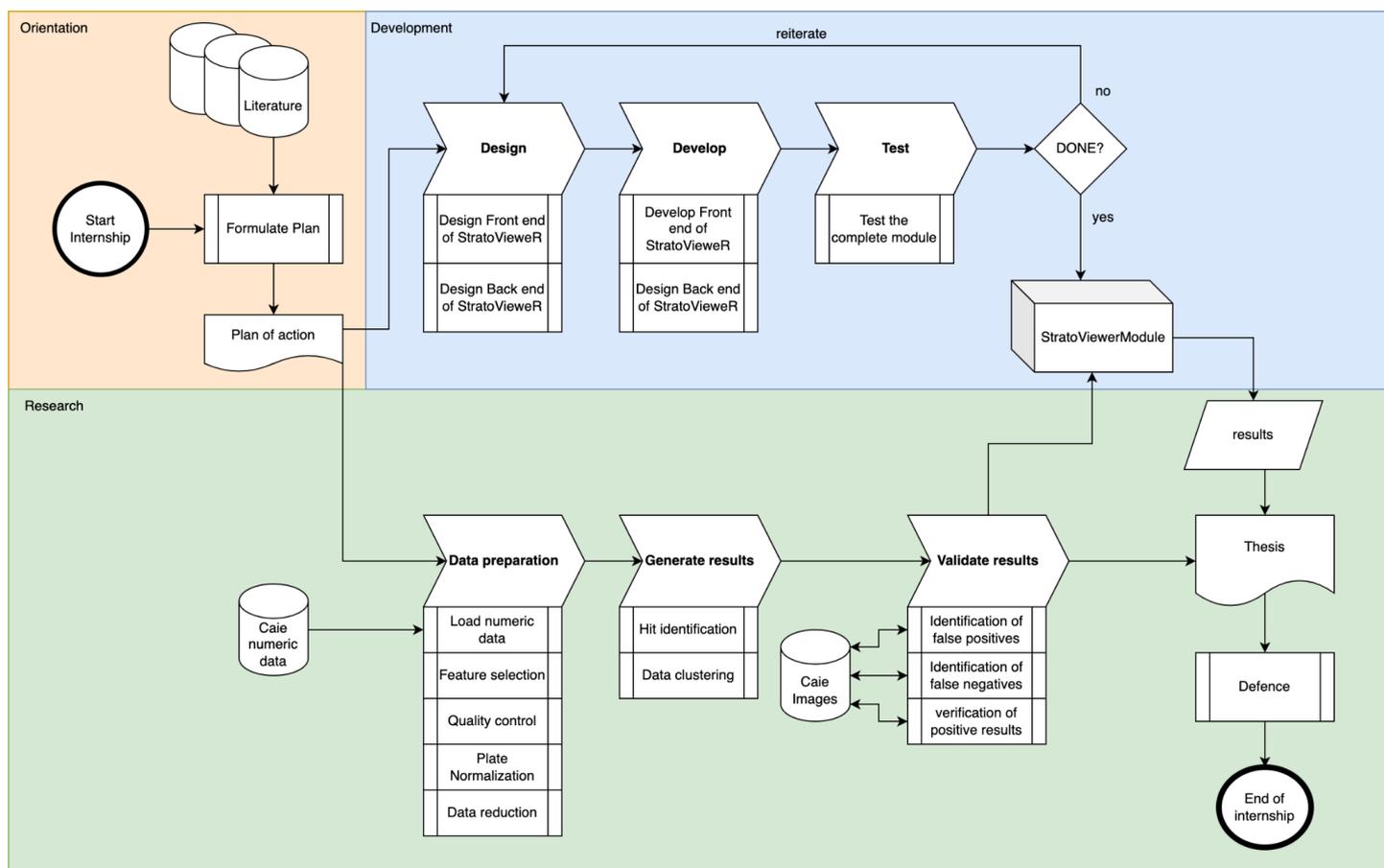


Figure 11. Overview of steps and processes for developing the StratoVieweR software and conducting research. The project consists of three main parts highlighted in: Orange, Orientation; Blue, Development and Green, Research.

The following list of features was drafted to describe the functionality needed to achieve the goals as stated in the project aim.

- Interactive overview of microplates that allows for well selection. This overview will be both in a schematic and thumbnail.
- Ability to change exposure of probe luminescence in the fluorescent dye.
- Ability to export visualisation, with settings used.
- Single Pane view with plate, settings and frames.
- The ability to support up to 6 channels.
- The creation of thumbnails by means of compressing single channel views to minimise loading time.
- The ability to import images and extract meta data and store them in a structured way, this feature is to be developed by the CLA team, and is beyond the scope of this project.

### 3 Materials and Methods

To unite HCS images with numeric data, a workflow has been drafted that describes how a user would interact with a software module enabling this goal. This workflow is shown in figure 12. This module will be referred to as StratoVieweR in the rest of the document. The chapter Materials and Methods describes the development of StratoVieweR and de validation of StratoVieweR as two separate entities and will be reunited in the conclusion.

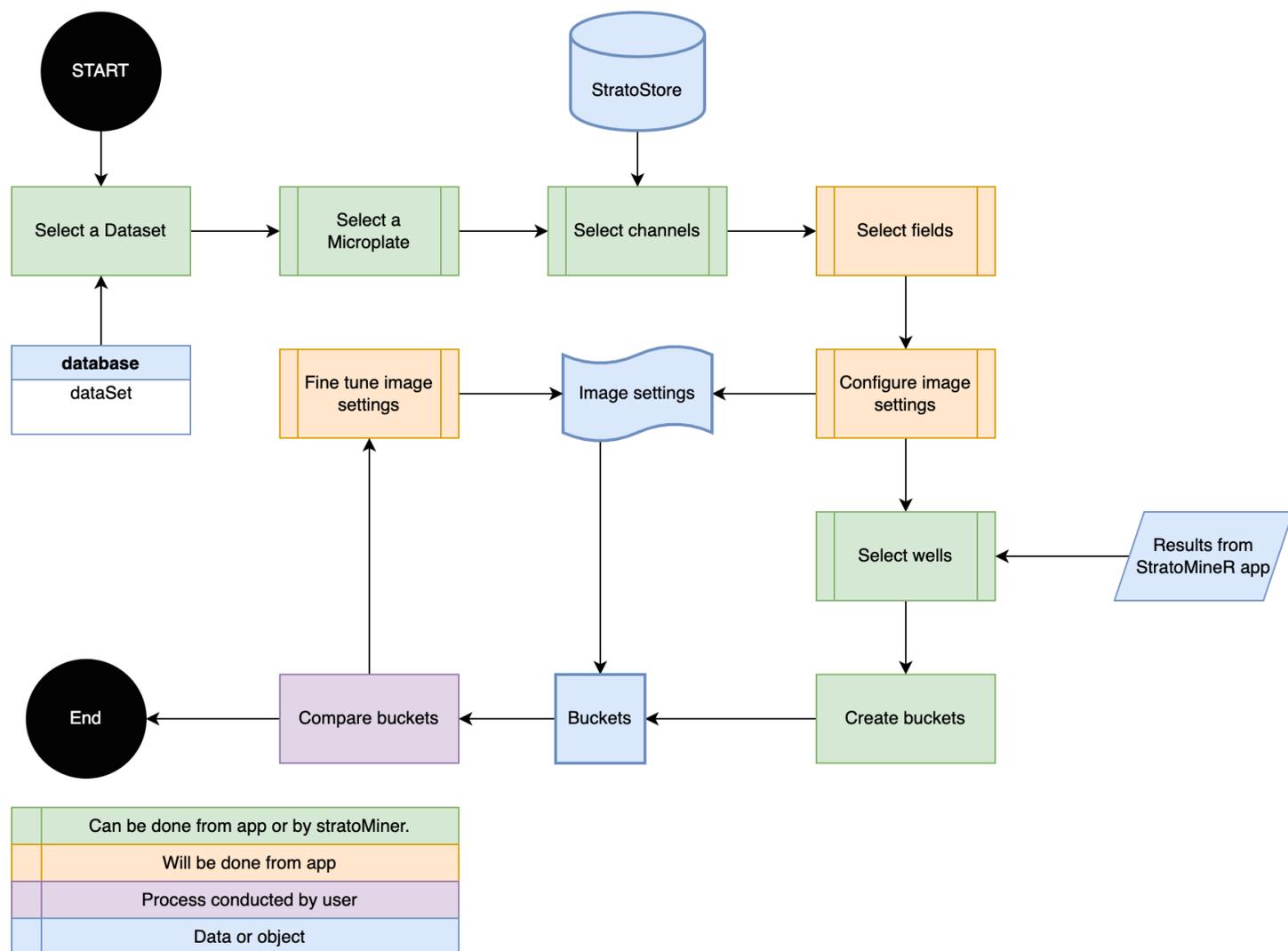


Figure 12. Workflow of the StratoVieweR module. This workflow describes how a user would interact with the module, either completely via StratoVieweR or as an extension of the StratoMineR platform. Objects in green are steps that can be performed via StratoVieweR or skipped when loading data via StratoMineR. Objects in orange are tasks always performed within StratoVieweR. Objects in purple are steps users perform manually. Objects in blue are external inputs or data objects. A user starts by selecting a dataset from StratoStore, then selecting Microplate, Channel, field and wells. The selected wells will be added to a bucket, settings to the image can be changed at any step in the process. The final step is to compare the images between the buckets.

### 3.1.1 Materials for StratoVieweR

The materials, resources and tools used to develop the StratoMineR module are summarised and described in table 1.

*Table 1. Resources used for the development of StratoVieweR, resources written in italic are used for both the development and research stage of this project.*

Category	Description	Details
Programming languages	R 4	Programming language designed for statistical computation
	JavaScript	Programming language designed for interactivity on websites
	HTML 5	Hyper Text Markup Language (HTML), programming language for scripting web pages
	PHP	General purpose scripting language for web development
	Bash/shell scripting	Commands for simple operations on a linux system
Data management	MySQL 5.7	Managed cloud database service
	AWS S3 Buckets	Storage medium on the amazon cloud
Development environments	Rstudio	Integrated development environment (Ide) for the R programming language
	VisualStudio	Universal development environment with ssh connectivity for remote deployment
	AWS compute instance	Cloud computing instance where programs are hosted.
Libraries	Jquery	Library for event handling in JavaScript
	Imager <sup>13</sup>	Library for using Cimg functions in R
	Shiny <sup>12</sup>	Library for creating interactive web applications in R
	AWS CLI 2	Amazon command line interface for the cloud
	Microbenchmark <sup>14</sup>	Tool for benchmarking functions in R
	FST <sup>15</sup>	Package for Lightning fast serialisation of data frames
Software	Omero Iviewer	Software for reviewing HCS image data
	<i>HC StratoMineR</i>	Software platform for HCS data analysis
Data	Caie Numeric Data	Dataset of feature data
	Caie Image Data	Dataset of image data

### 3.1.2 Structure of StratoVieweR module

All R shiny apps have a basic structure consisting of a Server and a UI. This has been expanded by a functions file, a variable file, a PHP file and a CSS file. The CSS file is used to add and modify graphical elements to the module. The functions file is used to separate the functions for the app from the server logic. The PHP file is used to pass information into the url of the page that can be sourced in StratoVieweR. The variables file is used to declare variables based on queries from the database. Some of these queries are executed by scripts already existent in the StratoMineR platform. An overview of files is given in figure 13.

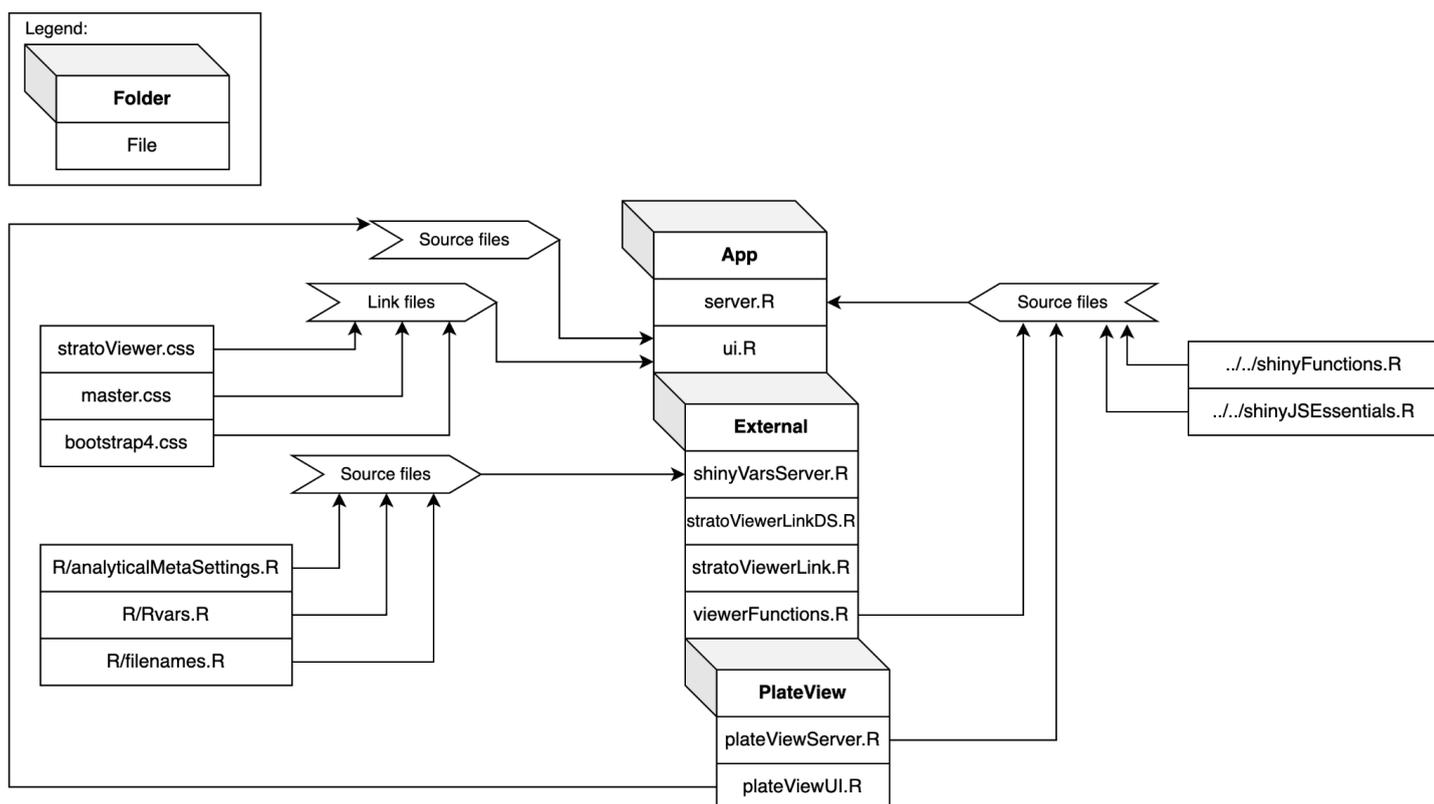


Figure 13. Overview of the files and folders comprising the StratoVieweR module. The files in the centre stack e.g. folders App, External and PlateView, are unique for the StratoVieweR module, all other files are shared between all the modules on the StratoMineR platform.

### 3.1.3 Connecting StratoVieweR

Connecting to the StratoMineR platform is done via three ways. A metaData file is used to facilitate the communication between StratoVieweR and the images. A different metaData file is created to directly communicate between StratoVieweR and other StratoMineR modules. Thirdly StratoVieweR connects to the database to retrieve experiment specific information such as reagent class, channel names. The database connection also lets StratoVieweR extract metadata, this is used to verify if images are available and where the metadata file is stored.

#### 3.1.3.1 Database

StratoMineR used to have the database structured around experiments. To expand the complete StratoMineR platform and to incorporate image data, two new tables are added. `db_imageDataSets` and `db_numericDataSets`. These two tables enable image specific data to be stored in the database. A

connection to `db_experiments` remains to maintain backwards compatibility to experiments initiated before the change. By adding these two tables multiple experiments can use the same `imageDataSet` to avoid redundancy.

### 3.1.3.2 MetaData image-connection

A metadata file in `fst` format is generated when images are uploaded via the `StratoStore` module. `StratoVieweR` reads this file as a `dataTable` and can generate a `plateMap` based on this file if no `plateMap` is available for this experiment. `StratoVieweR` also uses this file to find the location of images on the S3 server. To access files stored on the S3 server, a connection is made using the packages `S3-fs`.

#### 3.1.3.2.1 FST-file type

`Fst` a binary file type created using the `fst` package. `Fst` files are random accessible and are especially useful if large sums of data need to be stored but only part of the data needs to be retrieved. Using modern compression algorithms enables multicore scaling when writing and reading from this file type. The disadvantage of this file type is that it is stored in binary and not human readable.<sup>15</sup>

### 3.1.3.3 MetaData module-connection

When accessing `StratoVieweR` from a different module information on specific wells can be passed through to `StratoVieweR`. A function is written that takes `plateID`, `WellLocation`, and `ReagentCategories` as input and creates a `metaDataFile` in RDS format that `StratoVieweR` uses to create buckets. This function is part of the `StratoVieweR` module but can be used by all other modules on the `StratoMineR` platform.

#### 3.1.3.3.1 RDS-file type

`Rds` is the most common method for storing R data. `Rds` files are smaller than a text file containing the same data. `Rds` also has the advantage of storing data types, this means that after storing and loading data to and from an `rds` file there is no need to redefine data types. Using `rds` files for the metadata module-connection ensures that between modules no information is lost.

## 3.1.4 Interface of StratoVieweR

The interface of the module was designed to both enable users to intuitively access all images corresponding to an experiment as well as give users control on how these images are displayed. To give users an overview of the locations of all the wells two modes are developed. In `StratoVieweR` a schematic plate map is developed as well as a thumbnail version including information on the reagent class. At the top of the page a legend is added that shows which colour corresponds with which reagent class, these colours will also be used as borders for the preview images and as borders for the thumbnail images in the buckets. To switch the channel and field for the thumbnails, dropdown inputs were added. To select which plate to preview another dropdown input was added. A button to remove the reagent class from a well has also been added.

### 3.1.4.1 Buckets

When selecting wells from other modules on the `StratoMineR` platform and then connecting to `StratoVieweR`, those wells are divided into three buckets. The first bucket contains all wells from the most prevalent reagent categorie in the selection. The second bucket contains the second most prevalent reagent categorie and the third bucket contains all remaining wells. Users can add or remove any well to

any bucket via the plateView, a design for the layout can be found in figure 14. When users select a well from a bucket the corresponding plate map is loaded and the image is shown beside the plate map. That well is then also highlighted on the plate map.

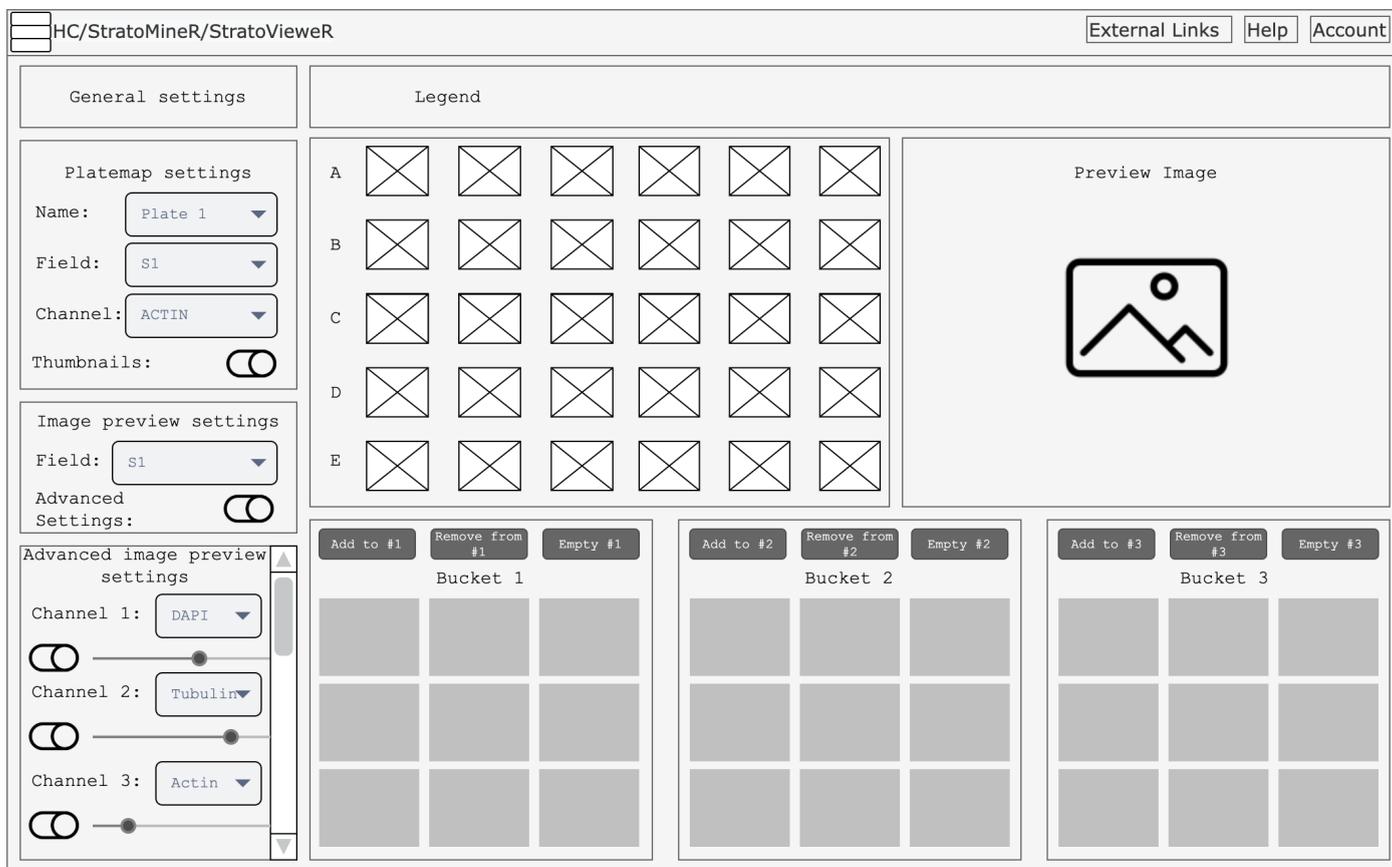


Figure 14. Schematic design of the StratoVieweR module, in plateview mode. The top row of information and inputs are default for the complete StratoMineR platform. The page is split in two columns, the left column shows all the settings and inputs users can select to influence what is shown on the right column. The right column is the content panel and displays the legend, a plate map, preview image and three buckets with controls over those buckets. The settings column (left) consists of four panels, the top panel gives users control over general settings e.g. remove well, refresh page, save and continue, platemap/ bucket comparison. The second panel in the settings column gives the user control over settings that influence the plate map e.g. plate name, field for thumbnail, channel for thumbnails and thumbnails on/off. The third panel in the settings column gives the user control over the preview image, the user can select the field to preview and enable advanced settings. The fourth panel only appears when the user selects advanced control over the preview image, in this panel the user can enable and disable channels, change the colour for each channel, change the scaling of the channels and control over the downsampling of the preview image. The advanced image preview settings panel is designed to be scrollable if the screen is too small to show all the options.

To compare images, a view is made where the three buckets are shown side by side, a design for this layout can be found in figure 15. On this page the user can then compare the images from the three buckets. To aid in the comparison of images a feature is added that can automatically set the channel levels to match one chosen image.

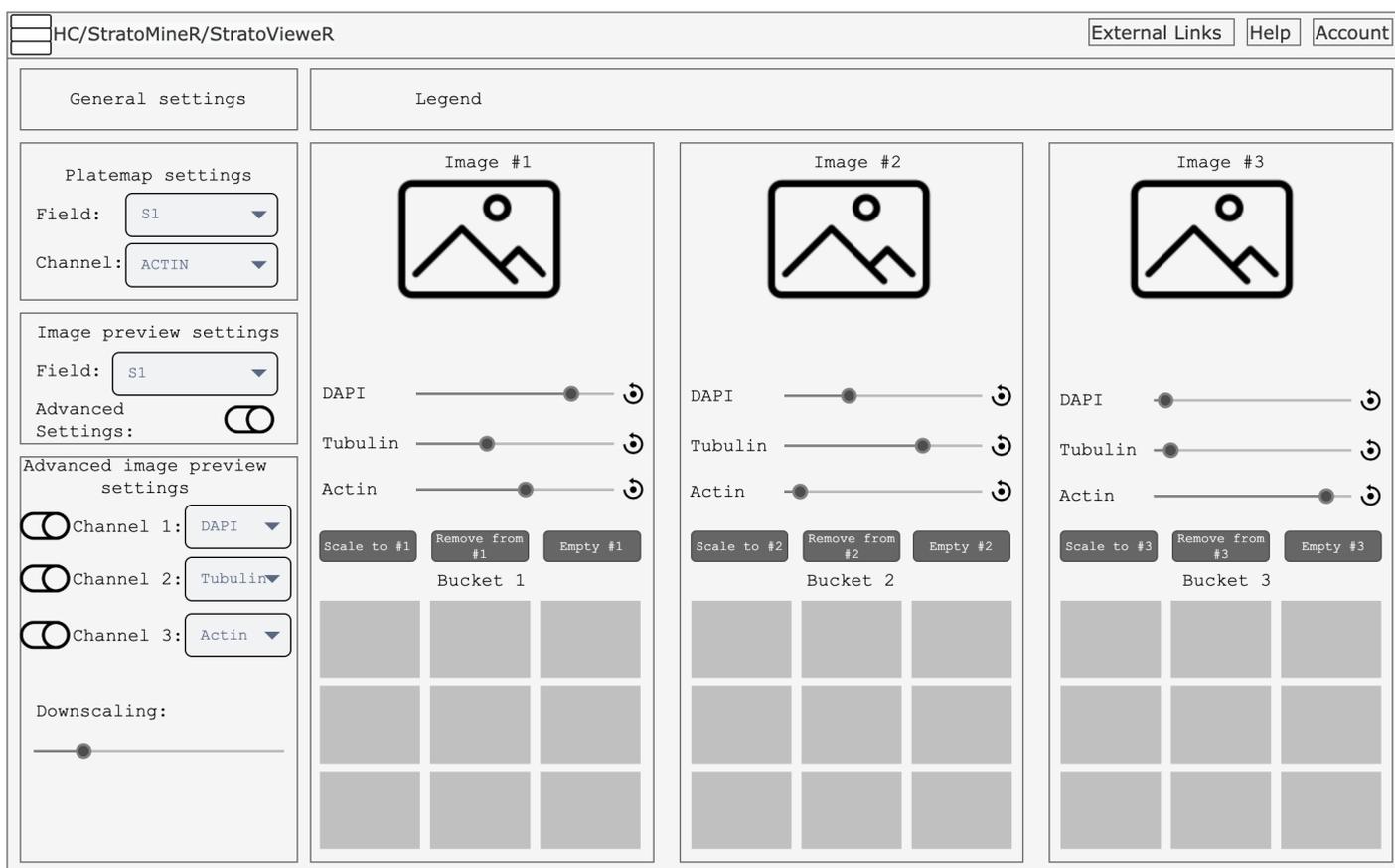


Figure 15. Schematic design of the StratoVieweR module, in bucket comparison mode. The top row of information and inputs are default for the complete StratoMineR platform. The page is split in two columns, the left column shows all the settings and inputs users can select to influence what is shown on the right column. The right column is the content panel. The content panel consists of a single row showing the legend and three columns each showing a single bucket. Each bucket column consists of a preview image with information on the image above. The bucket panel also has control on the intensity levels of each channel and a button to reset the channel to the original image, right of the slider. The settings column (left) consists of four panels, the top panel gives users control over general settings e.g. remove well, refresh page, save and continue and platemap/ bucket comparison. The second panel in the settings column gives the user control over settings that influence the thumbnail images in the buckets e.g. field for thumbnail and channel for thumbnails. The third panel in the settings column gives the user control over the preview image, the user can select the field to preview and enable advanced settings. The fourth panel only appears when the user selects advanced control over the preview image, in this panel the user can enable and disable channels, change the colour for each channel, and control over the downsampling of the preview image. The advanced image preview settings panel is designed to be scrollable if the screen is too small to show all the options.

### 3.1.4.2 Image settings

To select which field is shown in the image preview, a dropdown input was added. To give control on how the images are displayed, a toggle switch was made that enables an advanced image settings tab. In this tab toggle switches were added to select which channels to show. In the same panel dropdown, inputs were made available to select which colour is used to show a channel as well as a slider to change the range of intensity to display. One to three channels can be shown in red, green and blue. But it has been made possible to load up to six channels. To aid in loading times of the images, a slider was implemented that changes the resolution of the image.

### 3.1.5 Benchmarking of image downsampling

Benchmarking R functions is performed using the package `Microbenchmark`. `Microbenchmark` is a package for R that gives detailed benchmarking information on a run expression. `Microbenchmark` is written in C to minimise overhead. The benchmark function takes multiple functions to test.<sup>14</sup>

One option `microbenchmark` gives is 'order', when benchmarking the functions the execution order can be either randomised, in order or block. If option `randomised` is given the functions to test are executed in a randomised order. If option `in order` is given the functions are executed in order e.g. if an iteration of two is chosen for three functions `f1,f2,f3,f1,f2,f3`. The option `block` executes the functions in blocks e.g. if an iteration of two is chosen for three functions `f1,f1,f2,f2,f3,f3`. The default order for benchmarking is set to `random`.<sup>14</sup>

Secondly users can specify a number of warm up iterations. The warm up iterations are run before the real benchmark. This is done to compensate for startup latencies like the hard disc having to spin up or ram loading etcetera. The default warm up iterations are set to two.<sup>14</sup>

Finally users can specify the number of iterations the functions are executed. The default iterations are set to 100.<sup>14</sup>

To enable fast loading of the `StratoVieweR` module, and especially the plate map images can be downsampled. To measure the effect of downsampling different interpolation methods and scaling factors are tested. The benchmark is performed on two situations only downsampling and downsampling plus a modification of the image and plotting the image. Below the code for the two functions that are tested is shown. It also shows how the images were loaded. This code was written only for benchmarking and is not part of the `StratoVieweR` module.

```
# Sourcing three grayscale images using the load.image function:
channel1grey <- load.image('channel1_grey2.tif') # 1.6 MB
channel2grey <- load.image('channel2_grey2.tif') # 1.6 MB
channel3grey <- load.image('channel3_grey2.tif') # 1.6 MB

# Function that performs downsampling on three channels, returns NULL:
compression <- function(scale, interpolation_type) {
  compChannel1gr <- imresize(channel1grey,
                             scale       = scale,
                             interpolation = interpolation_type)
  compChannel2gr <- imresize(channel2grey,
                             scale       = scale,
                             interpolation = interpolation_type)
  compChannel3gr <- imresize(channel3grey,
                             scale       = scale,
                             interpolation = interpolation_type)

  return(NULL)
}
```

```
# Function that performs downscaling using the function imresize, all pixel
values are multiplied by 0.99 and appended to a single three color image.
This image is plotted without rescaling. The plot is not drawn to the screen
and the function returns NULL.
compressionAndPlotting <- function(scale, interpolation_type) {
  compChannel1gr <- imresize(channel1grey,
                             scale      = scale,
                             interpolation = interpolation_type)
  compChannel2gr <- imresize(channel2grey,
                             scale      = scale,
                             interpolation = interpolation_type)
  compChannel3gr <- imresize(channel3grey,
                             scale      = scale,
                             interpolation = interpolation_type)

  red      <- R(compChannel1gr) * 0.99
  green    <- G(compChannel2gr) * 0.99
  blue     <- B(compChannel3gr) * 0.99

  RGB_channels <- imappend(list(red,green,blue), 'c')
  plot(RGB_channels, rescale = FALSE)
  return(NULL)
}
```

The downsampling is performed using the function `imresize`. `Imresize` has the option to choose between seven methods of interpolation. These methods differ in how they determine the value for the pixel that replaces the pixels lost due to the downscaling. The specific techniques are not relevant to the project, but the resulting images could show strong artefacts that could influence a biologist's assessment of the image. The different methods also differentiate in speed.

## 3.2 Data analysis aided by StratoVieweR

### 3.2.1 Experimental Design

To test if uniting HCS images with numeric data has an effect on a classification model, and show the scale of that effect, an experiment as shown in figure 16 has been conducted. This figure shows a method of comparing the AS-IS situation with the TO-BE situation. Performing supervised clustering in `StratoMineR` on the `Caie` dataset represents the AS-IS situation. Performing supervised clustering in `StratoMineR` on the `Caie` dataset accompanied with the possible identification of artefacts through `StratoVieweR` will be the TO-BE situation.

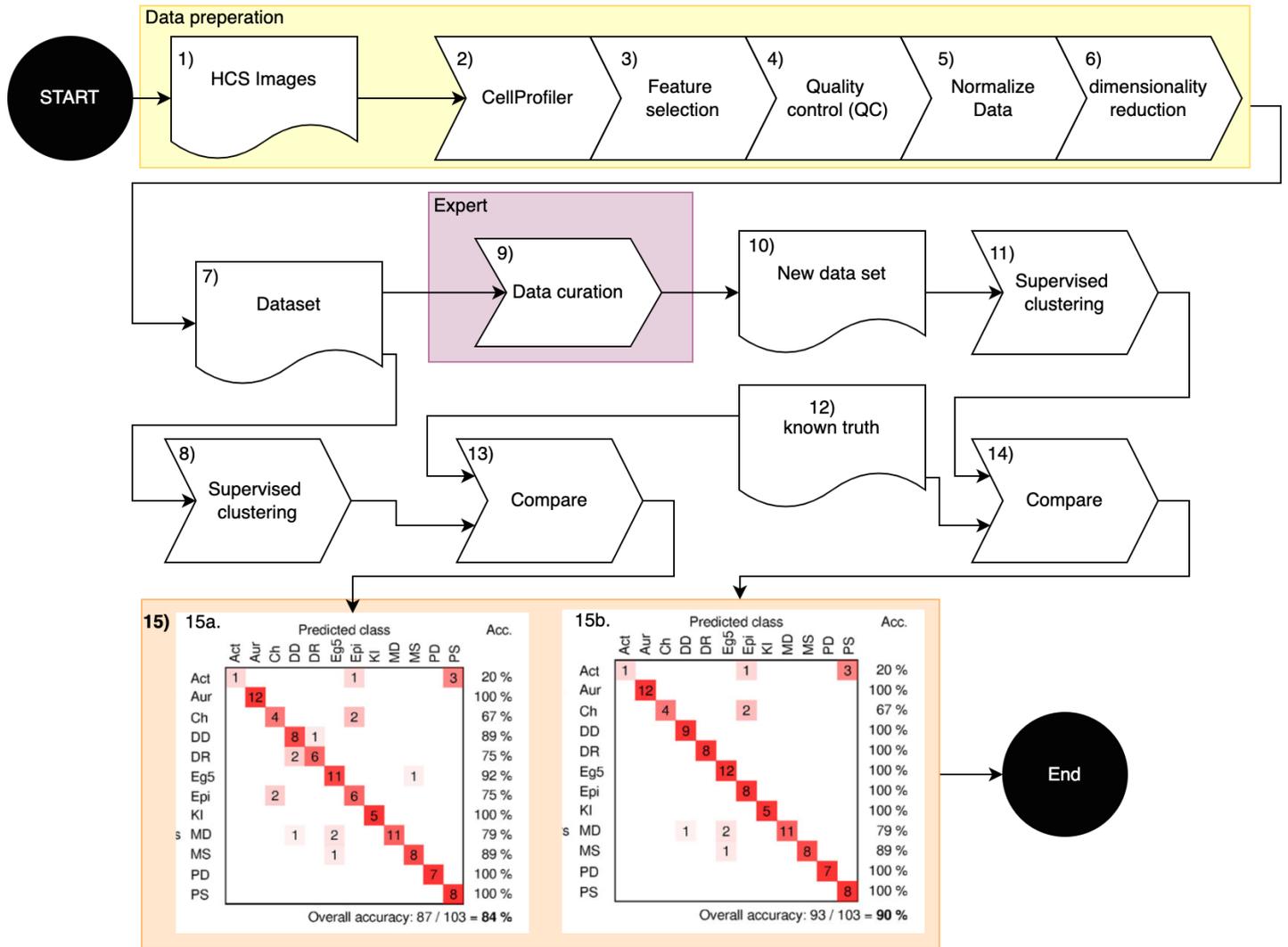


Figure 16. Workflow of the experiment. Step 1, 7, 9, 12 and 14 are data and do not have an action. Step 2 to 6 describes the creation of the data set based on the HCS images, this is highlighted in yellow. These steps are executed once. Gathering HCS images (1), executing in CellProfiler and feature selection was done by researchers at Edinburgh University. Step 8 and 11 are repeated 30 times. Step 13 and 14 are thus also run in multiple, depending on the number of new data sets, resulting in the same number of confusion matrices (step 15a & 15b). The confusion matrices in step 15 are exemplary only and do not hold relative information for this experiment.

### 3.2.1.1 Defined variables

Variables that exist in the dataset and the experiment are categorised in either experimental independent or response variables. The experimental independent variables are those that are not influenced by other variables in this study. Response variables are those that are influenced by one or more of the experimental independent variables. Table 2 gives an overview of the variables present in this experiment.

Table 2. Overview of experimental independent and response variables. The experimental independent variables are not influenced by any other variable, these can be seen as a constant. The response variables are all variables that can be measured that are influenced by either the experimental independent variables or the other response variables.

Experimental independent variables:	Response variables:
Feature data	Percentage of false positives
	Percentage of false negatives
	Percentage classified in similitude of ground truth
	Training set
	Test set
	Model
	Classification
	Starting confusion matrix

The relations between the variables as described in table 2 are shown in Figure 17.

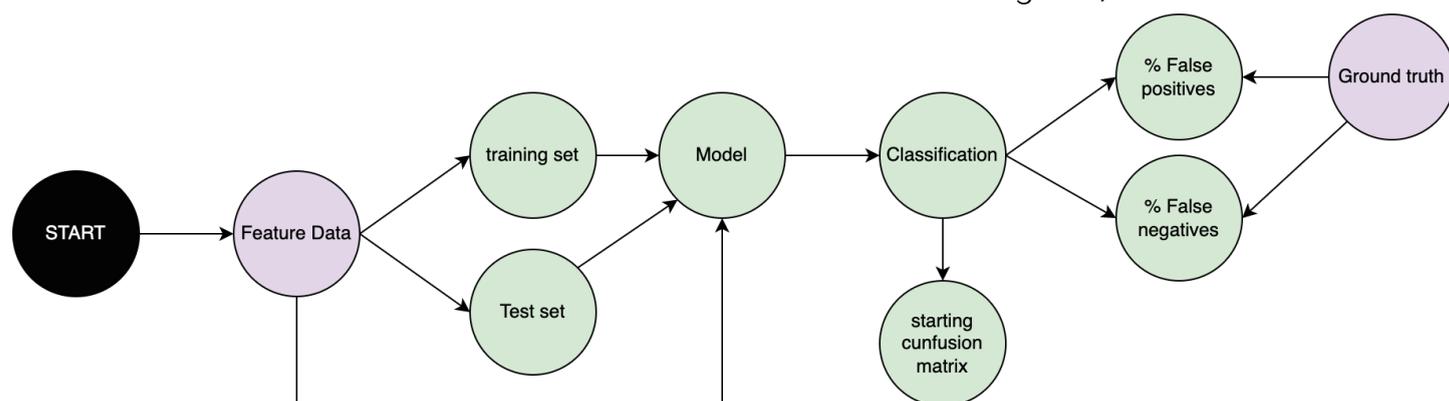


Figure 17. Relations between the variables. Experimental independent variables are highlighted in purple and the response variables are highlighted in green.

### 3.2.1.2 Training and Test -set

Of part of the data, the classification, Mechanisms of Action (MOA), is known, this subset will be used as training and testing data to form a model. The subset will be subdivided in two groups via an 80-20 split. The first group (~80%) will be used to train a model, the second group (~20%) will be used to test the training set. To determine the model's skill cross-fold validation will be applied when creating the test and training set. Cross-validation encompasses the following steps:

1. Randomise the order of the dataset.
2. Split the dataset into k-groups.
3. For each group:
  - a. Set the group as a test data set.
  - b. Set remaining groups as a training data set.
  - c. Fit the model to the training set and evaluate with the test set.
  - d. Keep the evaluation score.
  - e. Discard model.
4. Summarise the skill of the model based on all the evaluation scores.

### 3.2.1.3 Stratified sampling

Stratified sampling is a method to ensure equal representation of classes in the training and test sets. In stratified sampling the population (dataset) is defined by class (Strata). By random sampling the strata this will result in a sample that includes equal amounts of all classes that were present in the population. In Figure 18 an example is given visualising the process and result of stratified sampling. The dataset used in this experiment has a high variance in class size and thus stratified sampling will be used.

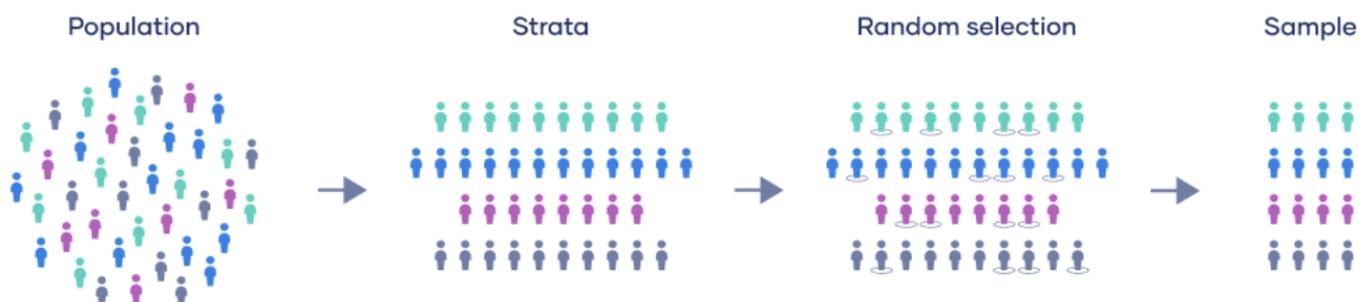


Figure 18. Example of stratified sampling. The population consists of samples with four unique characteristics, these are split into four groups of a single characteristic also called strata. A random selection is done over every strata resulting in the sample where every characteristic is represented equally. Image from scribbr.com<sup>16</sup>

## 3.2.2 Materials for validation

The materials, resources and tools used to validate the StratoMineR module are summarised and described in table 3.

Table 3. Resources used for the validation of StratoVieweR, resources written in *italic* are used for both the development and research stage of this project.

Resources for the research to validate the functionality of StratoVieweR		
Analysis tools	HC StratoMineR	Software platform for HCS data analysis
	StratoVieweR	Module for reviewing HCS image data integrated with HC StratoMineR
	R	Tool for statistical testing and plotting
Data	<i>Caie Numeric data plate 1-5 in triplicate</i>	Dataset of feature data
	<i>Caie Image data plate 1-5 in triplicate</i>	Dataset of image data

### 3.2.2.1 Numeric Features

Researchers at the University of Edinburgh have executed an HCS experiment of 102 well documented drugs and inhibitors in four cancer cell-lines. The targets were added in eight half-log doses, in 96-well assay plates screened in triplicates. Fluorescent probes were added to illuminate the actin cytoskeleton, microtubule and DNA. This dataset will be referred to as the Caie dataset.<sup>9</sup>

Using the CellProfiler tool metadata was extracted from the image files and the folder names as generated by the molecular device microscopy machine. These metadata are extracted according to the following parameters:

The well location, site and ChannelNumber was extracted by the following regular expression:

```
^(?P<Plate>.*)(?P<Well>\x5BA-P\x5D\x5B0-9\x5D{2})s(?P<Site>\x5B0-9\x5D)w(?P<ChannelNumber>\x5B0-9\x5D)
```

The plate number was extracted using the following regular expression:

```
.*(\\\\\\\\\\\\\\\\\x7C/)(?P<folder>.*)
```

The image type was set to be 'grayscale image' with a maximum intensity of 255.0. The relative pixel spacing for X,Y & Z was set to 1.0, this means that for every image a specific pixel describes information about that location irrelevant of the channel. Channel 1 received the image name DAPI, channel 2 received the image name Actin and channel 4 received the image name Tubulin. All images are grouped into categories, plates and wells. Thus every well has 4 field times 3 channels equalling 12 images and every plate has N wells consisting of the aforementioned images. After metadata extraction and assertion, cellProfiler identifies objects. Primary objects are identified based on the raw images. Secondary objects are identified based on images and the primary objects. Tertiary objects are based solely on objects.

The primary object to identify is the nuclei, this was done based on the DAPI images, the DAPI illuminates the DNA thus showing hotspots where the nuclei is. The typical diameter of objects are within 15 pixels and 115 pixels, all objects outside of this range are discarded. Objects touching the border of the image are not automatically discarded. When clumping occurs between nuclei Cellprofiler tries to distinguish between Nuclei by shape, dividing lines between nuclei are also drawn by shape. The maximum number of objects per image is 500.

The secondary object to identify is Cells, cellprofiler looks at the Actinimages, and overlays the nuclei as targets. By 10 pixels the primary objects are expanded to identify cells. The objects touching the border of the image are not discarded. Also the associated primary-object are not discarded. Discarding associated primary objects can be useful when only identifying objects as an intermediate step to identify a non-primary object.

The tertiary object to identify is the cytoplasm. By setting the larger identified objects as cells and the smaller identified objects as nuclei, the cytoplasm can be identified.

With the Cells Cytoplasm and Nuclei identified in all the images, numerical data was extracted describing these objects. Table 4 shows the measurements that were made by CellProfiler on the objects.

Table 4. Measurements made by CellProfiler on the identified objects of the Caie image dataSet, where applicable the method and settings are given for that measurement.<sup>9</sup>

Measurement:	Target Objects:			Method:	Settings:
Object intensity	Cells	Cytoplasm	Nuclei	based on corresponding image	
Object intensity distribution	Cells	Cytoplasm	Nuclei	based on corresponding image	Maximum radius: 100px
Object Size	Cells	Cytoplasm	Nuclei		
Object Texture	Cells			based on Actin image	Texture scale: 5
Object Texture		Cytoplasm		based on DAPI image	Texture scale: 10
Object Texture			Nuclei	based on Tubulin image	Texture scale: 20
Granularity	Cells	Cytoplasm	Nuclei	based on Actin image	
Granularity	Cells	Cytoplasm	Nuclei	based on DAPI image	
Granularity	Cells	Cytoplasm	Nuclei	based on Tubulin image	
Cell Neighbours	Cells			within a specified distance	Distance: 10
Cell Neighbours	Cells			adjacent	Distance: 10
Nuclei neighbours			Nuclei	within a specified distance	Distance: 2

### 3.2.2.2 Ground truth

The ground truth is a file that describes the primary MOA of a subset of the compound concentrations for the Caie dataset. There are 12 different primary MOA present in the tested compounds, of which 6 were identified visually (Actin disruptors, Aurora kinase inhibitors, Eg5 inhibitors, Microtubule destabilizers, and Epithelial). The other 6 were defined via literature research. Not all concentrations of compounds have shown results in this experiment, these were not included in the ground truth. Reasons for this is one of the following: the compound is overly toxic, the compound was inactive, the image did not pass the QC. A complete list of compounds can be found in table 5.

Table 5. Overview of compounds used in the Caie experiment. The compounds are grouped based on documented primary effects. (table continues on next page)<sup>9</sup>

<b>Actin cytoskeleton</b>	<b>Cholesterol lowering</b>	<b>DNA replication/DNA damage</b>	
Cytochalasin D	Lovastatin	Doxorubicin	Chlorambucil
Cytochalasin B	Simvastatin	Hydroxyurea	Mitoxantrone
Jasplakinolide		Methotrexate	Camptothecin
Latrunculin B		Aphidicoline	Cisplatin
Taurocholate		Mitomycin C	5-Fluorouracil
Sodium fluoride		Floxuridine	Bleomycin
		Arabinofuranosylcytosine	Cyclophosphamide monohydrate
		Etoposide	Temozolomide
		Acyclovir	Methoxylamine
<b>Estrogen antagoni</b>	<b>HDAC</b>	<b>Kinases (broad)</b>	
ICI 182,780	Valproic Acid	Genestein	H-7
	Trichostatin A	Staurosporine	Bryostatin
	Sodium Butyrate	Forskolin	PP2
		Okadaic acid	AG 1478
		Herbimycin A	Alosine A
<b>Kinases (CDK)</b>	<b>Kinases (MAPK)</b>	<b>Microtubule</b>	<b>Protein degradation</b>
Olomoucine	UO126	Colchicine	Calpain inhibitor 1 (ALLN)
Roscovite	PD98059	Monastrol	Lactacystin
alsterpauillone	SB202190	Nocodazole	MG132
Bohemine	SB203580	Paclitaxel (taxol)	Leupeptin
CDK 1 inhibitor III	PD169316	Vinblastine	Caspase inhibitor 1 (ZVAD)
CDK 1/2 inhibitor	SP600125	Epothilone B	Chloramphenicol
Indirubin monoxime (CDK/GSK-3)		Vincristine	Calpain inhibitor 2 (ALLM)
		Podophyllotoxin	Calpeptin
		Demecolcine	Cathepsin inhibitor 1
		Docetaxel	Proteasome inhibitor 1

<b>Protein Synthesis</b>	<b>Vesicle trafficking / glycosylation</b>
Anisomycin	Brefeldin A
Cyclohexamide	Deoxymannojirimycin
Emetine	Deoxynojirimycin
Puromycin	Tunicamycin
Rapamycin	Diaminobenzidine
Quercetin	Neomycin
Atropine	Nystatin
	Filipin

Seven of the compounds in the ground truth dataset have compounds with the prefix AZ, these compounds are not found in table 5. These compounds are classified compounds from astrazeneca and have limited to no information available to the public.<sup>9</sup>

### 3.2.3 Data Analysis

This paragraph describes the steps taken to determine the AS-IS situation and the TO-BE situation. Select controls, plate normalisation, data transformation, feature scaling, missing data, data reduction and hit selection were the steps taken to determine the AS-IS. QC-plot and image reviewing are the extra steps added to determine the TO-BE situation. An overview of the order of these procedures can be found in figure 16. All procedures shown in this paragraph are executed on the StratoMineR platform version 1.2.7.

#### 3.2.3.1 Feature Selection

To ensure all features to be non-redundant a filter-based method of feature selection was performed. This method scores features based on their dependence and correlation between the other variables. Based on a 0.99 correlation cut-off value 96 of the 466 analytical features were automatically selected for elimination. Via the StratoMineR platform no other features were manually selected for removal relying solely on the automated process as described.

#### 3.2.3.2 Select controls

Via the menu 'QC and Controls' the app 'define controls' was selected. Here a plate map was shown that depending on the selected controls wells were highlighted, see figure 19. In this experiment on all plates well B02, C02, D02, E11, F11 and G11 where selected as negative control, well B11, C11, D11, E02, F02 and G02 were selected as being the positive control.



Figure 19. Simplified overview of the module 'Define controls' on the StratoMineR platform version 1.2.7. NEGATIVE and POSITIVE are selected and highlighted on the plate map in red and green respectively. The button 'Save & Continue' progressed the user to the next step in the workflow.

### 3.2.3.3 Plate normalisation

To be able to perform a multivariate analysis on wells from different plates, plate normalisation is performed based on the median sample of each plate. To assess the effectiveness of the plate normalisation a plot is generated based on the correlation-correlation actin tubulin 64 feature. The plate normalisation was available via the menu 'Normalise Transform & Scale'. Here the module 'Plate Normalisation' was selected. Seven options for plate normalisation were available: No Normalisation, B-score, Normalised Percent Control, Percent Inhibition, Median NEGATIVE, Median SAMPLE, Median POSITIVE. To generate plots and validate the normalisation effect a preview feature was selected. See figure 20. This option does not have an influence on the normalisation but only on the generated plots.

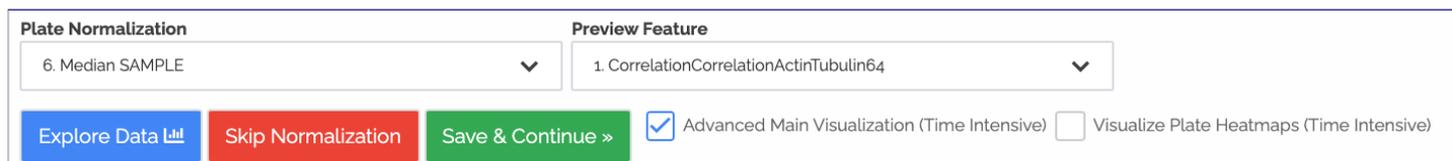


Figure 20. Screen capture of the module 'Plate Normalisation' on the StratoMineR platform version 1.2.7. Plate Normalisation set to Median Sample. Preview Feature selected is 'CorrelationCorrelationActinTubulin64' to generate plot. The checkbox Advanced Main Visualization is selected to generate plots when the button Explore Data is selected. The button 'Save & Continue' progressed the user to the next step in the workflow.

### 3.2.3.4 Data transformation

Though the random-forest model can handle skewed data, the principal component analysis applied in a later step relies on a linear and normally distributed dataset. Skewness was only transformed on features showing significant skewness above 0.0001. The transformation is automatically performed via the menu 'Normalise Transform & Scale'. The module 'Data Transformation' was chosen. A dropdown box was present where the default skewness significance was selected, see figure 21.

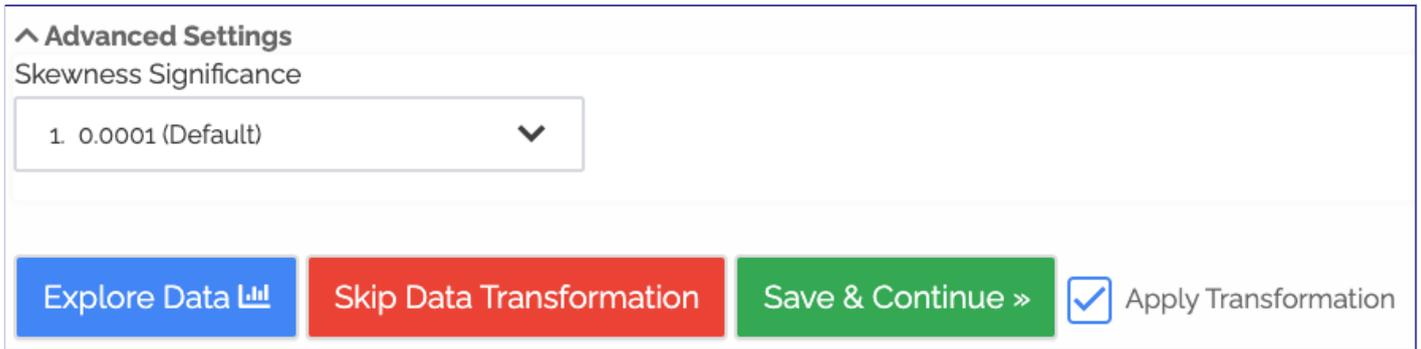


Figure 21. Screen capture of the module 'Data Transformation' on the StratoMineR platform version 1.2.7. Skewness significance set to 0.0001 (Default). The button Explore Data generates plots and statistics to review the effects of the transformation. The button 'Save & Continue' progressed the user to the next step in the workflow.

### 3.2.3.5 Feature scaling

By scaling all feature ranges to an equal scale the machine learning algorithm can calculate distances between the data. The equal range ensures that all features contribute proportionately to the calculated distances. The features were scaled based on the robust Z-score calculated per plate. The module 'Feature Scaling' via the menu 'Normalise Transform & Scale' on the StratoMineR platform was used to perform the feature scaling. Figure 22 shows the options for feature scaling as described prior.

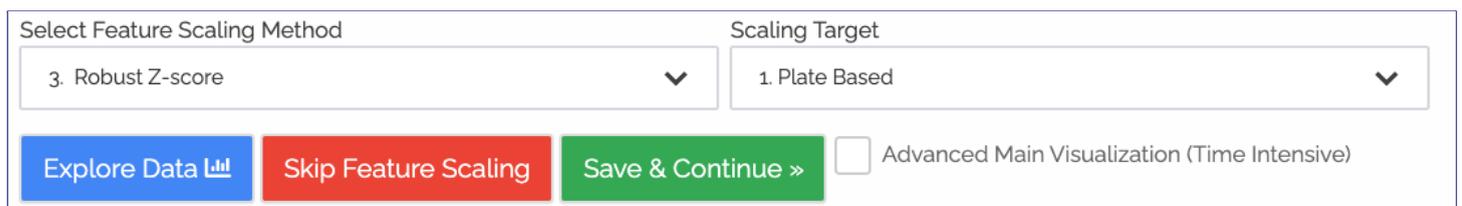


Figure 22. Screen capture of the module 'Feature Scaling' on the StratoMineR platform version 1.2.7. Feature scaling method has five options: None, Z-score, Robust-Z-score, Robust-Z-score (based on control, Min-Max (0-1)). These methods can be targeted on Plate Based or Screen. The button Explore Data generates plots and statistics to review the effects of the feature scaling. The button 'Save & Continue' progressed the user to the next step in the workflow.

### 3.2.3.6 Missing data

Missing data can have an adverse effect on almost all the steps in the data analysis. Therefore missing data was identified and imputed. To impute missing data the median of the feature, in which the missing datapoint is present, is used to replace the missing value.

### 3.2.3.7 Image Curation using StratoVieweR

Manually curating the images to generate the curated dataset was done by selecting wells that are interesting. Figure 23, shows a plot in the 'QC' module that has PCA01 on the X axis and cell counts on the Y axis. A group of wells is selected that cross the two groups. Via the blue button (StratoVieweR), the module StratoVieweR is opened.

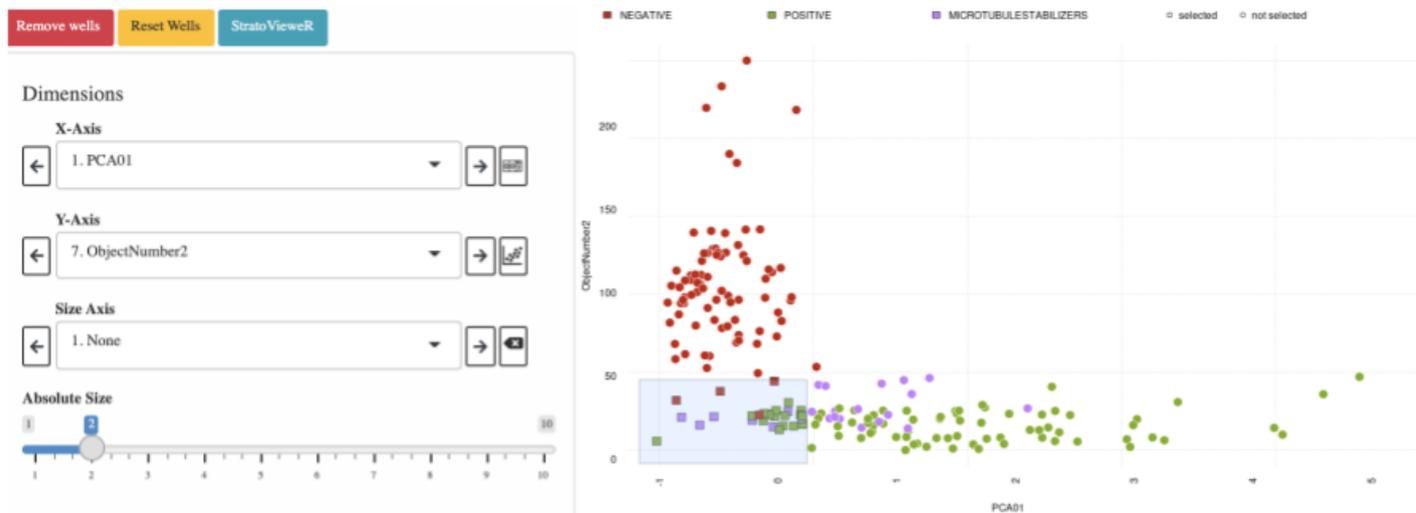


Figure 23. Screen capture of the module QC on the StratoMineR platform version 1.2.7. Showing reagent categories POSITIVE, NEGATIVE, and MICROTUBULE STABILISERS, of the Caie Dataset.

Via the plateMap page (figure 24) of StratoVieweR the selected wells are compared to neighbouring wells and wells from the same reagent categories.

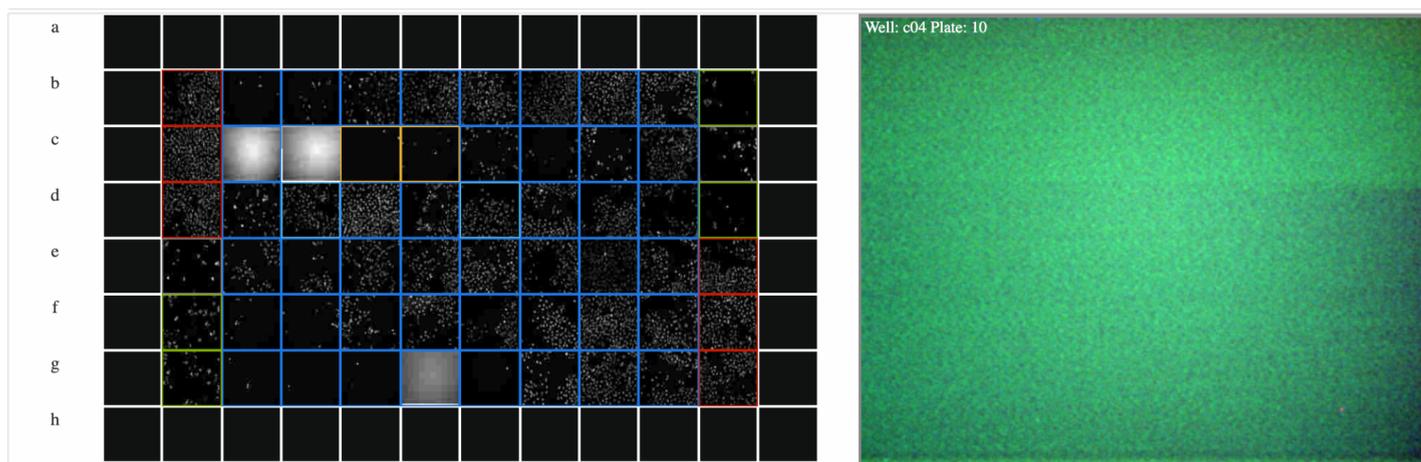


Figure 24. Simplified screen capture of the StratoVieweR module. On the left is the plate map of plate 10 with thumbnails enabled of the Caie Dataset. On the right is a preview of well c04 on plate 10.

After comparing the selected wells to its surrounding wells and or equal reagent categories. The bucket panel of the StratoVieweR module was opened, see figure 25. In this pane wells were compared and the feature 'Scale to image' was used to check if errors persisted when levels were made equal. When wells with errors were identified the button 'Remove Well' removes the reagent class and the well will no longer be part of the dataset.

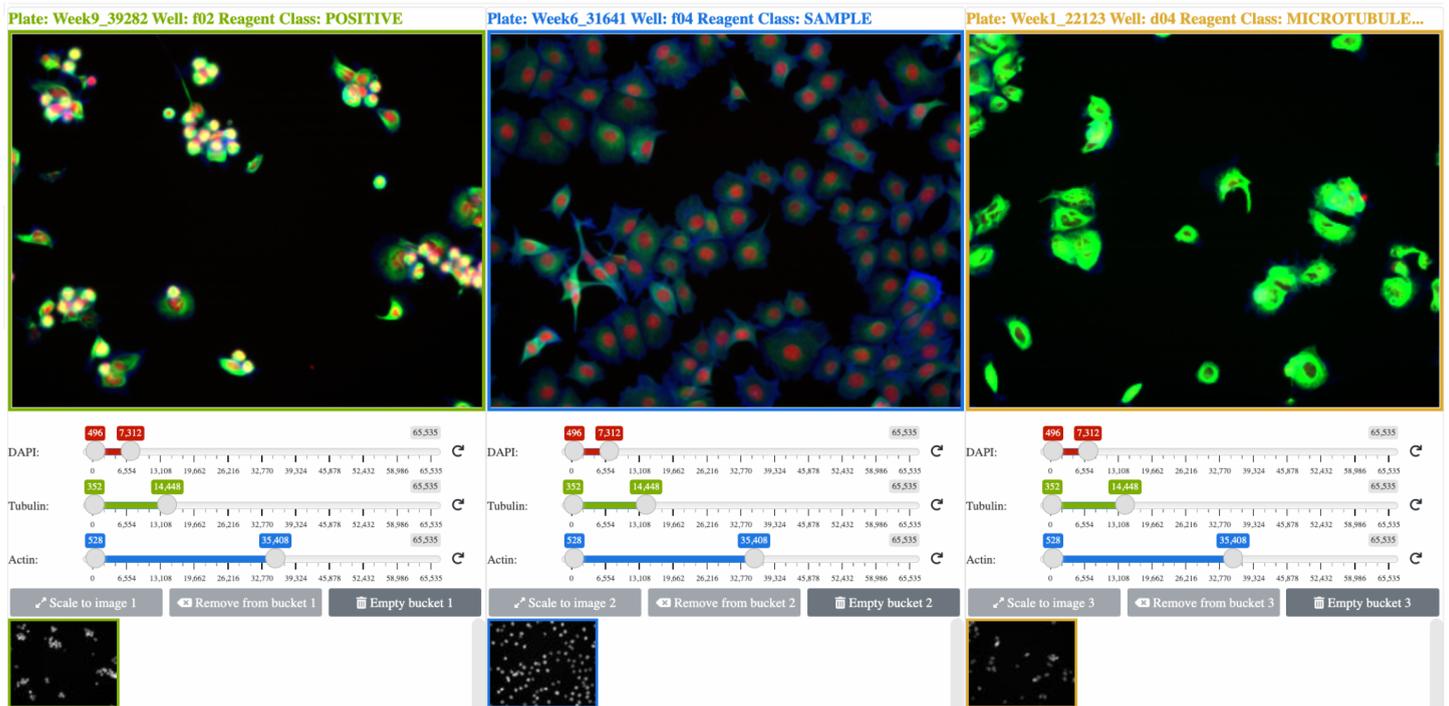
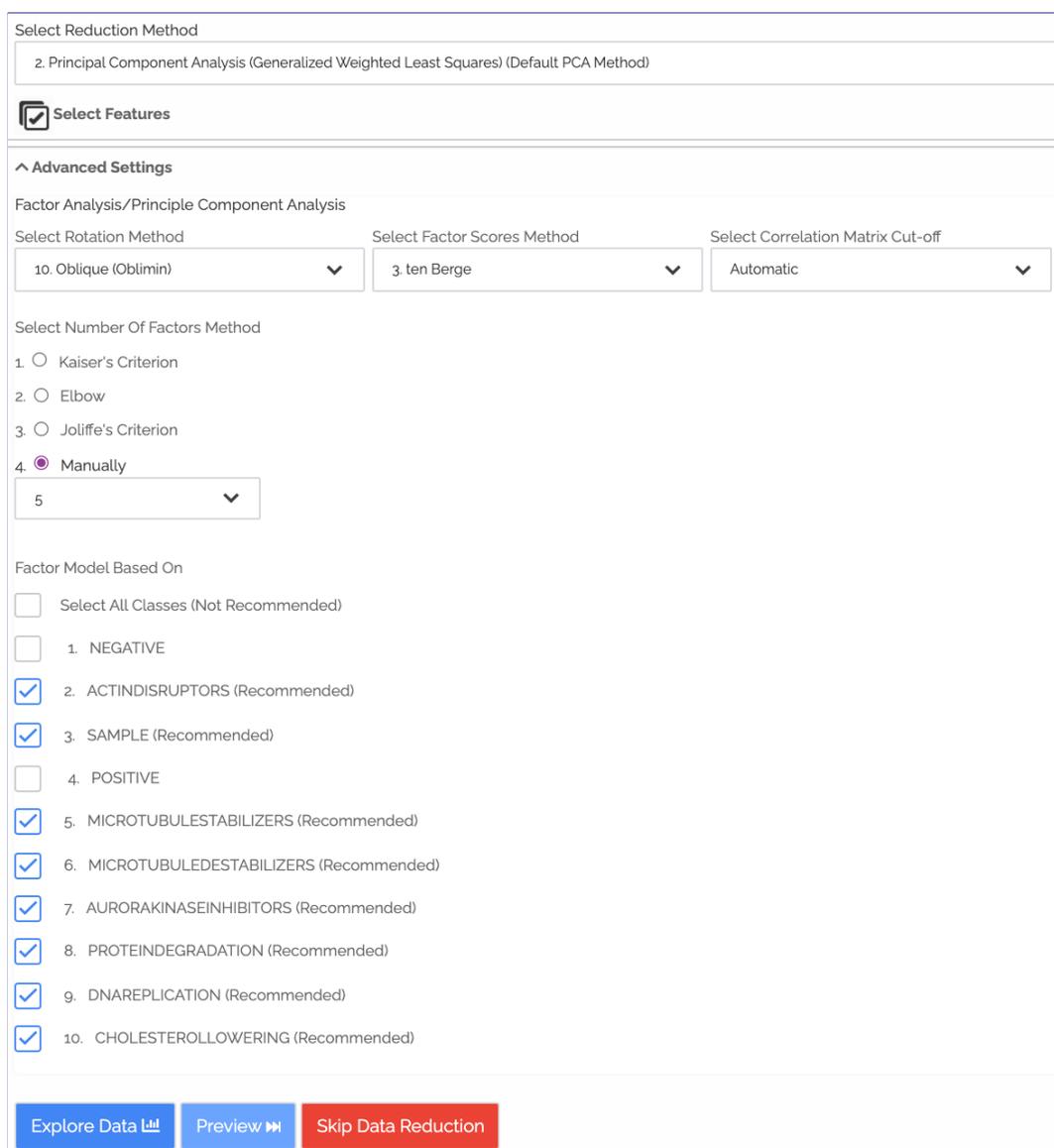


Figure 25. Simplified screen capture of the bucket panel of the StratoVieweR module. Images are scaled to the second image. The channel DAPI is represented in red, Tubulin in green and Actin in blue. From left to right the reagent classes are Positive, Sample and Microtubule stabilisers.

### 3.2.3.8 Dimensionality reduction

Via the menu 'Data Reduction' the module 'Dimension Reduction' was selected. Figure 26 shows a screen capture of the then presented page. To reduce the dimensions the option Principal Component Analysis (PCA) was selected. Under advanced settings, the options Rotation Method: oblique, Factors scores Method: ten Berge and the Correlation Matrix cut-off: Automatic were selected. The number of Factors was manually set to five based on the scree plot. The factor model was based on all Classes except NEGATIVE and POSITIVE. Dimensional reduction was run for both the curated and full dataset.



Select Reduction Method

2. Principal Component Analysis (Generalized Weighted Least Squares) (Default PCA Method)

Select Features

^ Advanced Settings

Factor Analysis/Principle Component Analysis

Select Rotation Method: 10. Oblique (Oblimin) ▼

Select Factor Scores Method: 3. ten Berge ▼

Select Correlation Matrix Cut-off: Automatic ▼

Select Number Of Factors Method

1.  Kaiser's Criterion

2.  Elbow

3.  Joliffe's Criterion

4.  Manually

5 ▼

Factor Model Based On

Select All Classes (Not Recommended)

1. NEGATIVE

2. ACTINDISRUPTORS (Recommended)

3. SAMPLE (Recommended)

4. POSITIVE

5. MICROTUBULESTABILIZERS (Recommended)

6. MICROTUBULEDESTABILIZERS (Recommended)

7. AURORAKINASEINHIBITORS (Recommended)

8. PROTEINDEGRADATION (Recommended)

9. DNAREPLICATION (Recommended)

10. CHOLESTEROLLOWERUNG (Recommended)

Explore Data  Preview  Skip Data Reduction

Figure 26. Screen capture of the module 'Dimensional Reduction' on the StratoMineR platform version 1.2.7. The number of factors was set manually to 5 based on the Scree plot, (see Results, Data analysis, Dimensional reduction). The factors are made based on all reagent classes excluding the negative and positive control classes.

### 3.2.3.9 Hit selection

The final step performed in the analysis is the hit selection. In this step the large sum of reagents is reduced to a smaller set that can be used for further screening. Via the menu 'Hit Selection' the module 'Hit Selection' was selected. Figures 27, 28 and 29 show screen captures of the presented page. Via this module two methods are available to determine hits. Either via Unsupervised clustering or via Supervised classification referred to as Artificial Intelligence (AI) in StratoMineR, see figure 27. For this experiment we had selected AI as the method to select hits.

#### 3.2.3.9.1 Basic AI settings

The basic AI settings give the options to determine what to target and what data to use to generate the model. The training classes were set to NEGATIVE, POSITIVE and MICROTUBULE STABILISERS. Sampling was set to 100.000 records, this is the highest option and can result in oversampling if less than 100.000 records are available. A choice of a 20/80 percent split on the testing vs training size was chosen by

default. The focus class, or the class we want to predict, was set to MICROTUBULE STABILISERS. And the option was chosen to find cells that are similar to MICROTUBULE STABILISERS.

Hit Selection Method

1. Unsupervised

2. Artificial Intelligence

---

Basic Settings

Select Predictors

P-Value Cut-off

0.05 (Default) ▼

---

Supervised Settings

Training Classes

1. NEGATIVE

2. ACTINDISRUPTORS

3. POSITIVE

4. MICROTUBULESTABILIZERS

5. MICROTUBULEDESTABILIZERS

6. AURORAKINASEINHIBITORS

7. PROTEINDEGRADATION

8. DNAREPLICATION

9. CHOLESTEROLLOWERUNG

Sampling: 3. High (100K Records, Slowest) ▼

Testing vs. Training Size: 20% vs. 80% (recommended) ▼

Focus Class: 4. MICROTUBULESTABILIZERS ▼

Find Similar Or Dissimilar Data To Focus Class: Similar ▼

Figure 27. Part of a screen capture of the module 'Hit Selection' on the StratoMineR platform version 1.2.7. This page is only available if the HitSelection Method is set to Artificial Intelligence, and shows the basic settings defining what data to use for the training and testing of the model and what target to train for.

3.2.3.9.2 Select predictors

Selecting predictors is made visible after pressing the button Select Predictors (figure 27). In the panel Select predictors (figure 28) all principal components were selected and all other features were deselected.

**Select Predictors**

Select Components  Select Features  Select all

Feature/Factor/Component	Visualization
<input checked="" type="checkbox"/> 1. PCA01	
<input checked="" type="checkbox"/> 2. PCA02	
<input checked="" type="checkbox"/> 3. PCA03	
<input checked="" type="checkbox"/> 4. PCA04	
<input checked="" type="checkbox"/> 5. PCA05	

Figure 28. Part of a screen capture of the module 'Hit Selection' on the StratoMineR platform version 1.2.7. This panel becomes visible if the button 'Select Predictors' in the module 'Hit Selection' is pressed. In this panel the predictors for the AI model can be chosen.

### 3.2.3.9.3 Advanced AI settings

By selecting Advanced Settings in the 'Hit Selection' module the setting as shown in figure 29 became available. These settings enable advanced control over the AI method. Here Random Forest is chosen as the supervised classification method, with 10-fold Cross validation (CV)

^ Advanced Settings

Advanced Settings

AI Method: Random Forest (Recommended) ▼

Trees: Automatic ▼

Multiple Test Correction: FDR (Default) ▼

X-fold Cross Validation: 4. 10-Fold CV (Extremely Slow) ▼

Explore Data  Clear Settings

Include Reagent Class In Hitlist

- Select All Classes
- 1. NEGATIVE
- 2. ACTINDISRUPTORS
- 3. SAMPLE
- 4. POSITIVE
- 5. MICROTUBULESTABILIZERS
- 6. MICROTUBULEDESTABILIZERS
- 7. AURORAKINASEINHIBITORS
- 8. PROTEINDEGRADATION
- 9. DNAREPLICATION
- 10. CHOLESTEROLLOWERUNG

Figure 29. Part of a screen capture of the module 'Hit Selection' on the StratoMineR platform version 1.2.7. This panel becomes visible if the button 'Advanced Settings' in the module 'Hit Selection' is pressed. In this panel the settings for the AI model can be chosen.

## 4 Results

### 4.1 StratoVieweR

#### 4.1.1 Database

The StratoVieweR module connects with the StratoMineR platform via a database connection. Figure 30 shows an overview entity relation diagram (ERD). In this figure only the different tables, primary and foreign keys are shown. The original database is expanded to include the tables `db_numericDataSets` and `db_imageDataSets`. The table `db_experiments` was expanded with the extra attribute `numericDataSetID` as a foreign key to create a connection with the new tables. The attribute `numericDataSetID` can be left empty if an experiment does not have image data stored on the StratoMineR platform to maintain backwards compatibility with older experiments that have not been initiated via the new `stratoStore` method of uploading data. A record in table `db_numericDataSets` can be used with different experiments workflows on the same numeric dataset. The same holds true for the records in the table `db_imageDataSets`. If a user generates different numeric dataSets based on the same images these can all be used to perform experiments without having to duplicate the data.

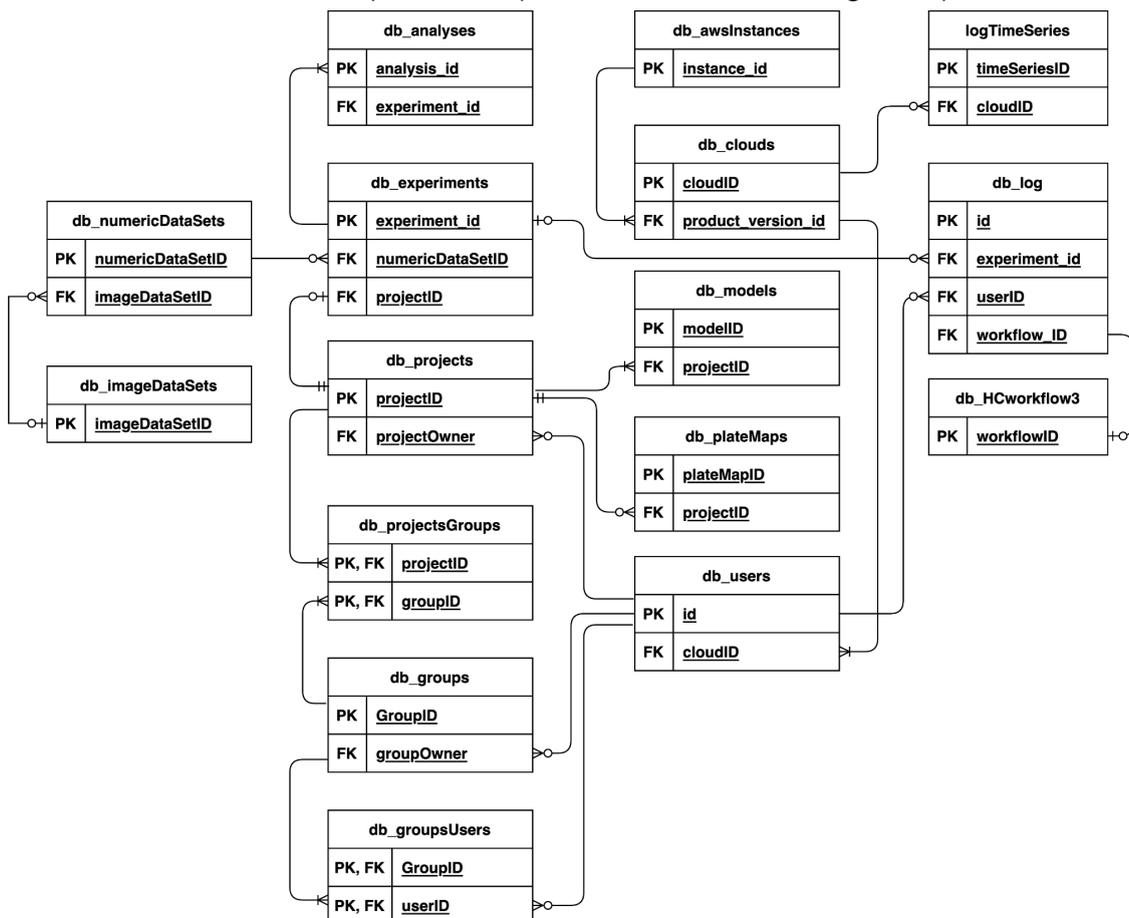


Figure 30. Entity relation diagram (ERD) of the StratoMineR database (SMNR) showing only primary and foreign keys, all other attributes of the tables are not displayed for readability. Tables `db_numericDataSets` and `db_imageDataSets` are added for this project.

All the records of the tables `db_numericDataSets` and `db_imageDataSets` are shown in figure 31. In this figure it shows that there is overlap between the two tables. This is done because both need to be able to exist without the other table existing. And to give users of the StratoMineR platform the possibility to, for example, rename the channel names separately per numeric data set without changing the original imagedata set. It also aids in the possibility to only upload numeric or only upload image data sets. There is also overlap between the two new tables and the original `db_experiments` to maintain backwards compatibility with existing experiments.

<b>db_imageDataSets</b>	<b>db_numericDataSets</b>
<u>imageDataSetID (PK)</u>	<u>numericDataSetID (PK)</u>
dataSetName	<u>imageDataSetID (FK)</u>
cloudID	dataSetName
projectID	cloudID
dataSetOwner	projectID
hasNumericData	dataSetOwner
isValid	isValid
assayPlateCount	assayPlateCount
reagentPlateCount	reagentPlateCount
ReplicatesCount	replicatesCount
plateType	plateType
fileCount	fieldCount
fieldCount	channelCount
fieldCodes	ch1Name
channelCount	ch2Name
ch1Name	ch3Name
ch1Code	ch4Name
ch2Name	ch5Name
ch2Code	ch6Name
ch3Name	nDSComments
ch3Code	protocolFileName
ch4Name	dataSetCreated
ch4Code	lastUpdated
ch5Name	
ch5Code	
ch6Name	
ch6Code	
iDSComments	
plateWise	
dataSetCreated	
lastUpdated	

Figure 31. Detailed overview of the tables `db_imageDataSets` and `db_numericDataSets`. These tables are added to the StratoMineR database (SMNR) see figure 30.

#### 4.1.2 MetaData files

For StratoVieweR and other modules within the StratoMineR platform to find the image files on the s3 server a FST file is created. The first five lines of the fst file for the caie dataset are shown in table 6. This file is created by the StratoStore module. It once again creates some overlap with the information in the database (figure 30), this is done so that based on the information the module fetches from the server can be used to filter the correct filename and other relevant information from the fst File.

Table 6. First five lines from the fst file that connects the filelocations of the images with the rest of the StratoMineR platform. This file has a row for each image in a dataset, wells that do not have images uploaded do not appear in this file. The file gives information on the PlateName, WellLocation, field, channel, fileName, fileExtensions, fileID and plateID.

PlateName	WellLocation	field	channel	fileName	fileExtensions	fileID	plateID
150607	B02	s1	w1	Week1_150607_B02_s1_w107447158-AC76-4844-8431-E6A954 BD1174	tif	1	1
150607	B02	s1	w2	Week1_150607_B02_s1_w23FDB0AC4-EA74-4D33-A7D4-8FFC 4C9ED7C8	tif	2	1
150607	B02	s1	w4	Week1_150607_B02_s1_w429636E34-C663-4E49-84B5-3EA42 9CAB4CE	tif	3	1
150607	B02	s2	w1	Week1_150607_B02_s2_w14AB22A28-7EF5-461A-8956-A7F6A 013F412	tif	4	1
150607	B02	s2	w2	Week1_150607_B02_s2_w261187024-FDFF-4974-A9AC-B09821 E1FoBo	tif	5	1

A second metaData file is the rds file. This file is used to communicate between a well selection from a module to StratoVieweR. The file is named using a random number which is passed into the url of the StratoVieweR module. This file gives the microPlateID, wellLocation and reagentCategories of the selected wells (first five lines shown of a random selection in table 7). By sorting the table and dividing into groups a modified rds file is created with an extra column bucket.

Table 7. First five lines of the rds file generated by another module to communicate a selection of wells to the StratoVieweR module. This file gives information about the selected wells, it shows its microPlateID, wellLocation and reagentCategories.

microPlateID	wellLocation	reagentCategories
1	f03	SAMPLE
1	g03	AURORAKINASEINHIBITORS
2	c05	SAMPLE
2	g05	AURORAKINASEINHIBITORS
3	g05	AURORAKINASEINHIBITORS

### 4.1.3 User interaction design

The StratoVieweR module is designed to be fully integrated into the StratoMineR Platform. Three different methods are available to review the images of a dataset via StratoVieweR: Plateview, iterative data mining and bucket comparison. In this paragraph these three examples are demonstrated.

#### 4.1.3.1 Plate wise

Users can view plates with thumbnails in the StratoVieweR module to identify and verificate inconsistency in replicates, plate or batch effects, and extreme outliers. This view also can be used for an extra verification step to see if a correct platemap is selected.

The plateview can be controlled via the menu shown in figure 32. This menu gives the user control over the plate to preview (Plate Id), what field to show as thumbnail (Select field), which channel to use as preview (Select channel), and whether or not thumbnails should be shown (toggle switch). The plateview is automatically updated when changes are made to the input. If via other methods the plate shown is changed this is also changed in the input buttons, this way the information shown in the plateMap Settings always corresponds with the plate shown. The inputs for plateId, field and channel are searchable via text input, browsable via the dropdown. It is also an option to toggle through the options via the arrow buttons placed left and right of the dropdown.

plateMap Settings

Plate Id:

Select field:

Select channel:

Thumbnails

Figure 32. PlateMap Settings from the StratoVieweR module, Three searchable dropdowns to select the Plate Id, Field and channel used to create the plateMap. The thumbnails toggle switch enables or disables thumbnails on the plateMap.

A comparison of a plateview with thumbnails and a schematic plateview is shown in figure 33. Both plateviews of figure 33 are created via the settings of figure 32 but on the right the option thumbnails is disabled.

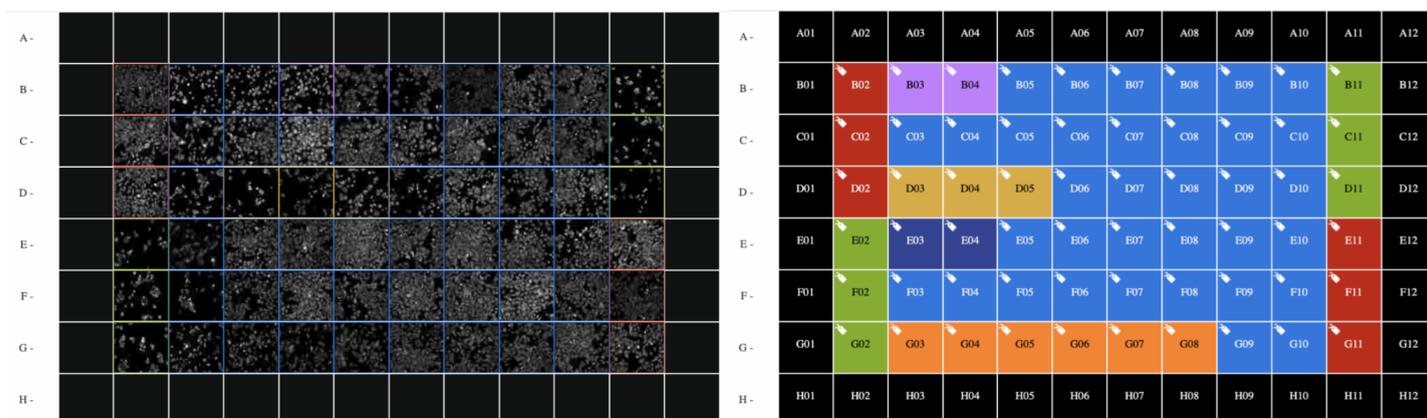


Figure 33. Screenshots of a platemap with (left) and without (right) thumbnails in StratoVieweR, legend is enlarged for clarity. The colour borders and fill correspond to a reagent class. The first and last columns and rows of this plate do not contain images.

#### 4.1.3.2 Iterative data mining

Users can create buckets with wells in the StratoVieweR module by creating a selection in different apps on the StratoMineR platform. When a selection is made developers can use a function developed for StratoVieweR. The function generates a StratoVieweR button and creates a rds file based on the selection. Via this module, developers working on StratoMineR, can with just a few lines of code, link to StratoVieweR. Figure 34 shows how a user would use these features.

Figure 34A shows a screenshot from the QC app where the StratoVieweR function is implemented. When the user created a group of wells (figure 34 A 1) the button StratoVieweR appears (figure 34 B 2). Pressing this button opens the view shown in figure 34C in a new tab of the browser. In the bottom of this page the

buckets are shown with the selection sorted based on the reagent categories of the selection. Pressing the images in the bucket (figure 34 C 3) highlights that well on the corresponding plate map and shows the image to the right. If the user decides that that image has errors pressing the button shown in figure 34 D 4 removes the well from the experiment. The data is retained thus via other apps the removed well can be reintroduced to the dataset. This method can be used to curate a dataset based on a combination of the numeric data (figure 34 A) and the image data (figure 34 C).

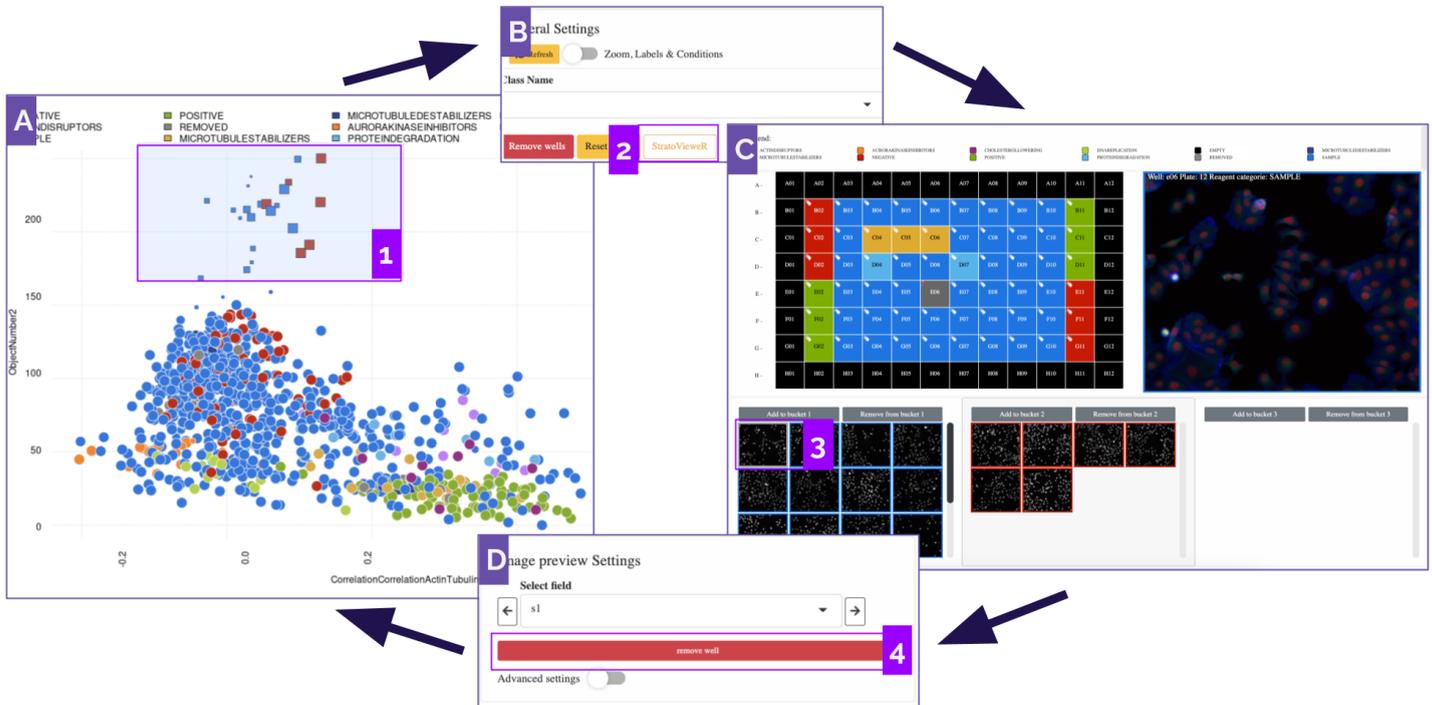


Figure 34. Workflow consists of screenshots on how the StratoVieweR application can be used to aid in an iterative data mining experiment on HCS numeric and image data. Part A shows a screenshot of the QC module with a selection made (1), when this selection is made a button appears (B2) this button links to the StratoVieweR, opening in a separate tab of the browser (C). In StratoVieweR the selected wells are shown in two buckets sorted based on reagent class. The first well (3) is selected, when selected the user can press the button (4) in image preview settings (D) to remove the reagent class. After which the reagent class label is removed from this well and will no longer be part of the data analysis.

#### 4.1.3.3 Bucket comparison

The final proposed method in which StratoVieweR can be used is in the direct comparison of images. Figure 35 shows, from left to right, images from plate 12 well d02 (Negative control) plate 2 well g02 (Positive control) and plate 2 well d04 (Microtubule Stabiliser). Figure 35A and 35B only have the channel Tubulin enabled. In 33A all three images look similar, this means that there seems to be very little to no effect of the different treatments. Using the automatic scaling feature with as target the Negative control image figure 35B is created. The scaling works by applying the native image settings as a template for the other images. Using this scaling it becomes obvious that the labels had a stronger connection with the tubulin in the positive and the microtubule stabiliser treatment. This same process has also been applied with three channels enabled (figure 35C and 35D) this highlights the strong effect on the tubulin compared to the actin and DAPI.

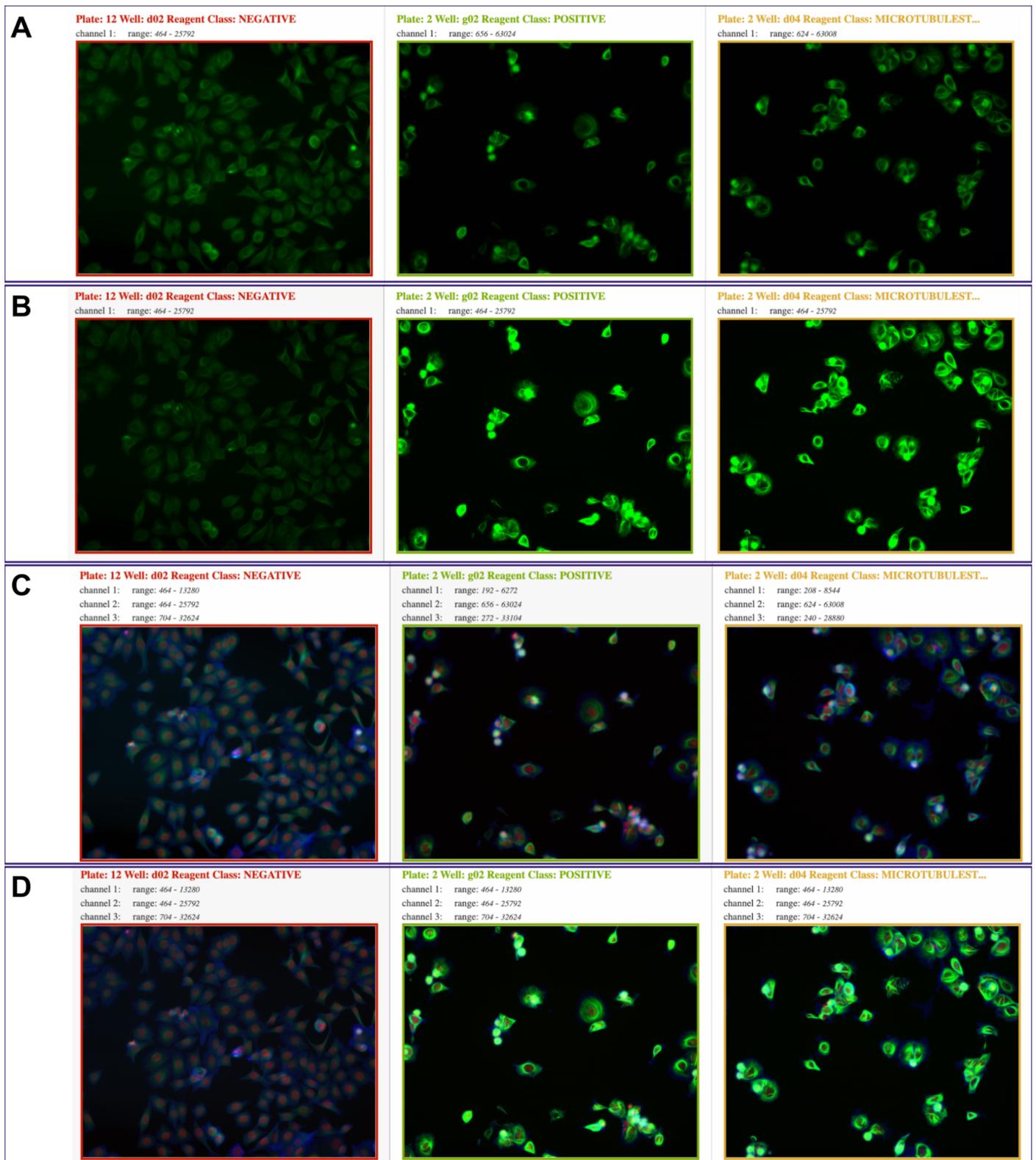


Figure 35. Screenshots from the preview windows in the image comparison view of the StratoVieweR module. The images used for all 4 subfigures are from left to right, plate 12 well d02 (negative control), plate 2 well g02 (positive control) and plate 2 well d04 (microtubule stabilisers) of the Caie dataset. Subfigure A shows unscaled channel 1 (green) tubulin. Subfigure B shows channel 1 (green) tubulin scaled to the negative control image. Subfigure C shows unscaled all channels. Subfigure D shows all channels scaled to the negative control image.

### 4.1.4 Benchmarking

The StratoVieweR module is designed with a large focus on efficiency. Downscaling is applied to try and drastically speed up the loading and displaying of images. To validate the appropriate level of downscaling that results in the highest level of speed increase and the least amount of quality loss a benchmark was run using the R package microbenchmark. The results of the benchmarking can be found in figure 37 and 38. The images generated via the seven different methods of compression are shown in figure 36. Looking at these images, near to no difference can be seen. Only 36G seems to be sharper than the others. The difference is so minimal that the decision was made to continue benchmarking using all the different methods to determine the best level settings.

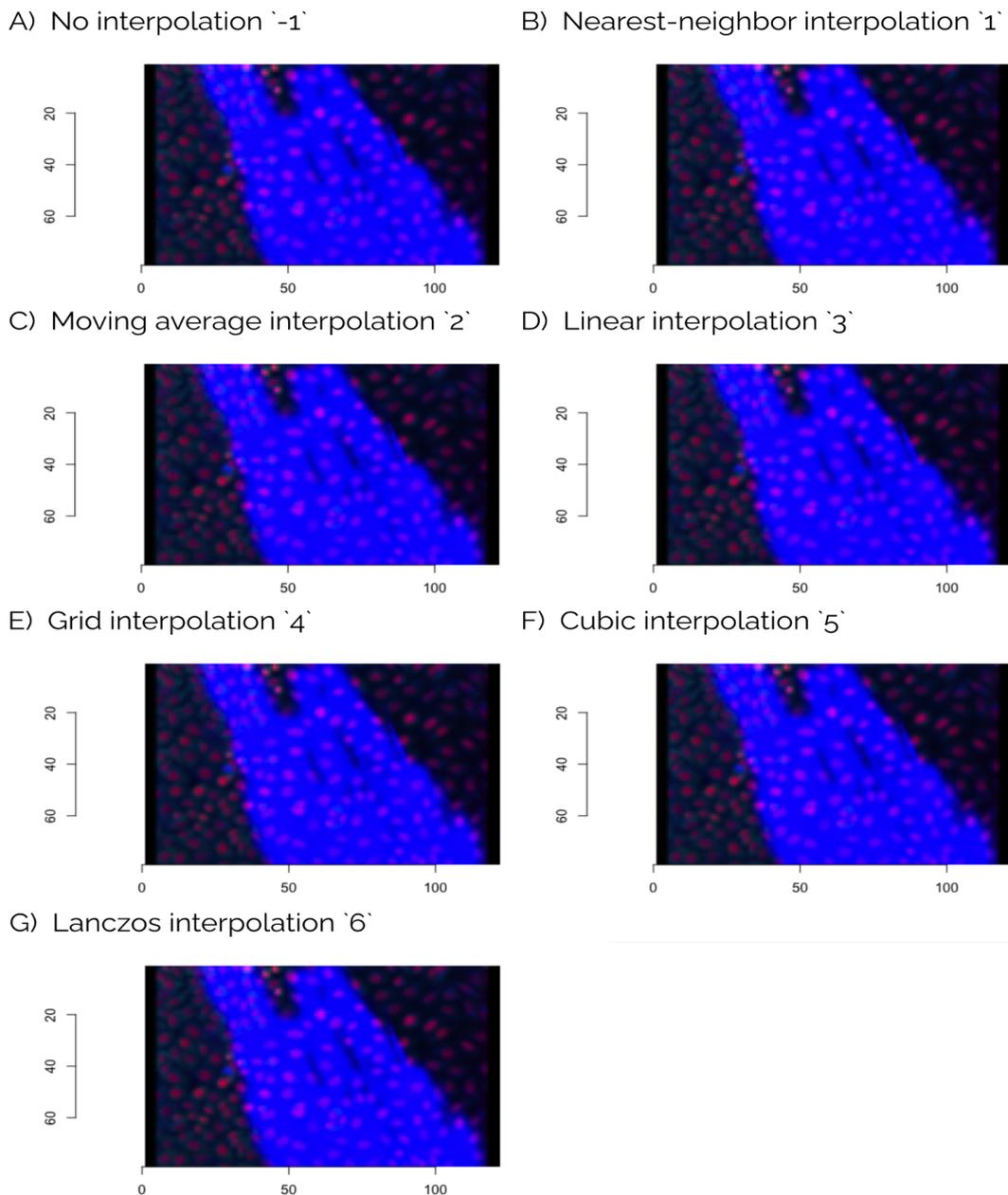


Figure 36. This figure shows the results of applying an 8 fold downsampling to three channels using the seven interpolation methods provided by imager `imresize` option. For each image, after the method, a number is written that corresponds to the option passed into `imresize` to create the image.

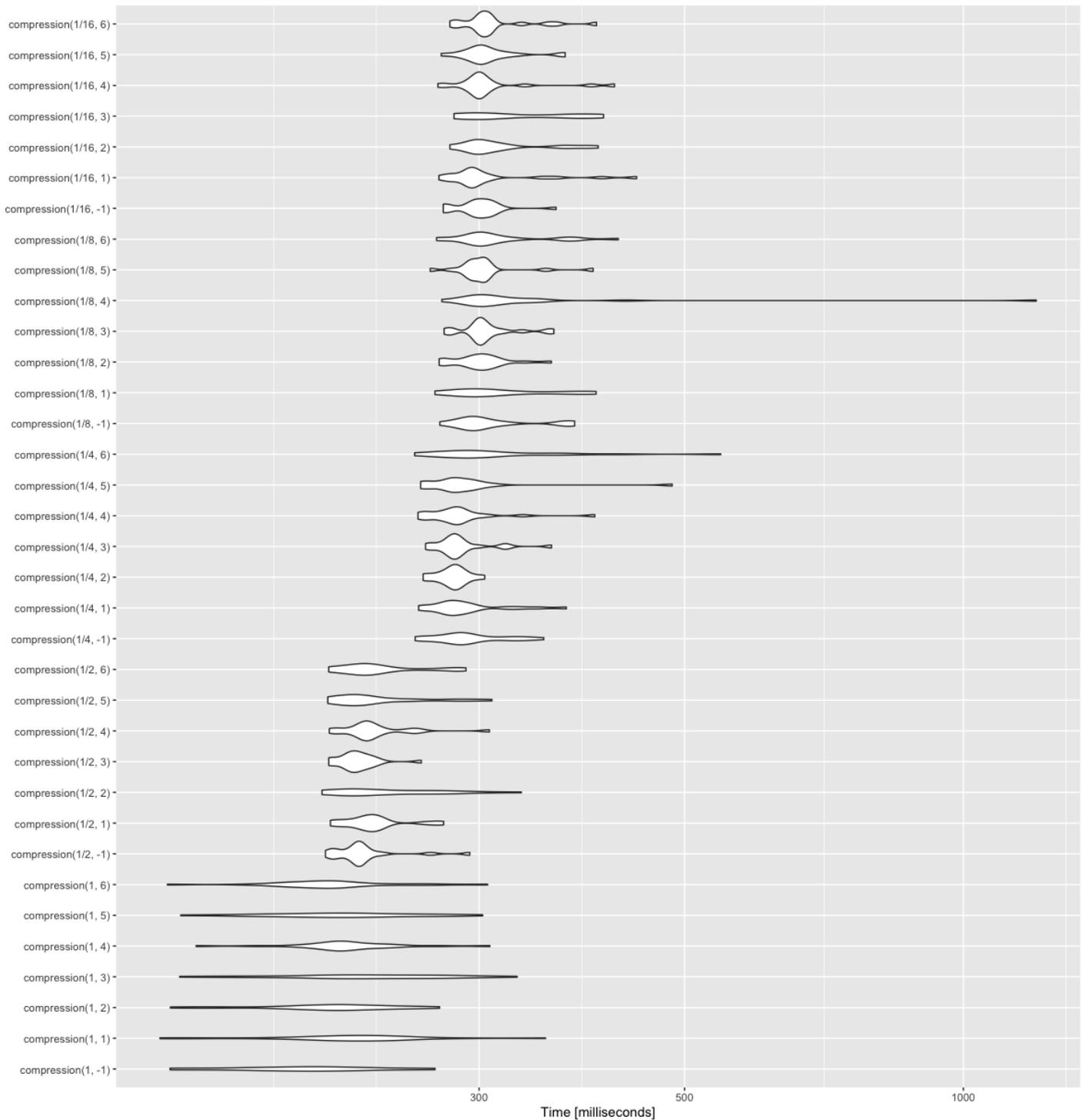


Figure 37. Violin plot of all microbenchmark results of the function `compression` being run 100 times on three channels of 1.6 Mb in size. The x-axis shows the time in milliseconds each iteration of the function took. The line thickness corresponds to how often that time is measured. It is important to note that the x-axis is not linear.

First of, figure 37, displays two notable effects of the downscaling based on scaling factor alone. Excluding downscaling by a factor of one which results in the same image and is thus not relevant for any type of efficiency improvements, the variance and distribution between the different scaling factors does not seem to differentiate strongly. The second noteworthy observation that can be made based on the scaling factors is that the time it takes to perform the downsampling does not scale linearly between the factors. It shows that there is a far larger gap between a downsampling factor of 2 and 4 than between 4 and 8. When

reviewing the different scaling methods it does not appear to consistently influence the speed and variance of the downscaling.

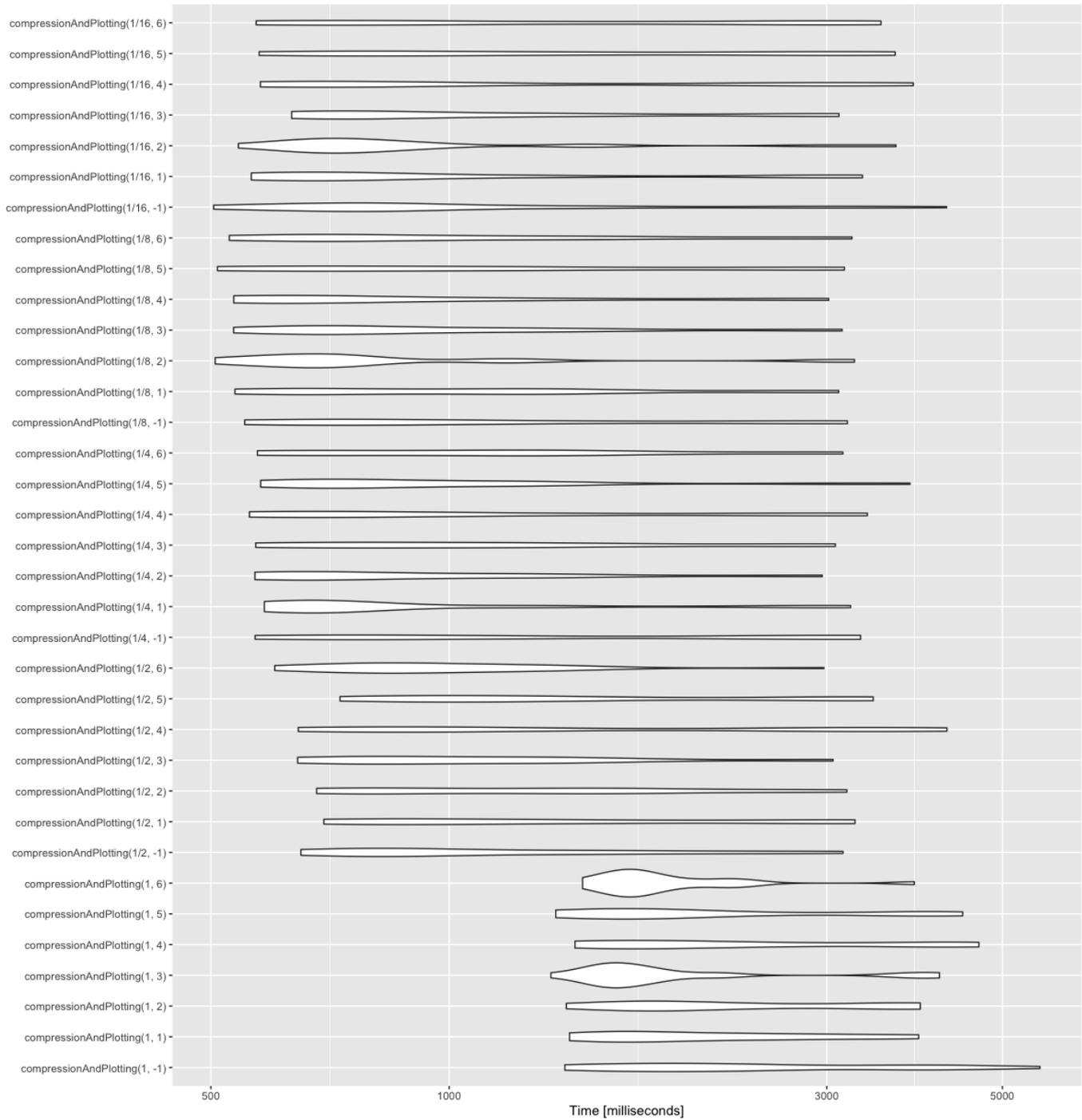


Figure 38. Violin plot of all microbenchmark results of the function `compressionAndPlotting` being run 100 times on three channels of 1.6 Mb in size. The x-axis shows the time in milliseconds each iteration of the function took. The line thickness corresponds to how often that time is measured. It is important to note that the x-axis is not linear.

From figure 38 similar conclusions can be made as those from figure 37. It appears that the increase in speed for compression and plotting also does not scale linearly. There is hardly any benefit in increasing the downsampling factor above 8. However even though there seems to be a bigger concentration on the lower end of time for images downsampled by method option 2 (moving average), this effect becomes only

prevalent above a scaling factor of 4. Below four, method -1 (no interpolation) appears to be most efficient. For the scaling of preview images which require higher resolutions method -1 is chosen and for the platemap method 2 is chosen.

One variable not taken into consideration for these benchmarks is the user's internet connection. The benchmarks were all run on a local system, whilst the implementation is in a cloud environment. Especially for the thumbnail images downscaling to a factor 16 drastically increases the loading time of the plateMap.

## 4.2 Data Analysis

### 4.2.1 Plate normalisation

Figure 39 shows the means of the analytical feature 'correlation Actin-Tubulin 64' plotted per plate before plate normalisation. In this figure we can see that the means of all plates are not in line. The plot also shows that there is a variance in the size of the interquartile range between plates. It is also noteworthy that a difference can be seen between the different reagent classes on the Y-axis, in particular positive versus sample.

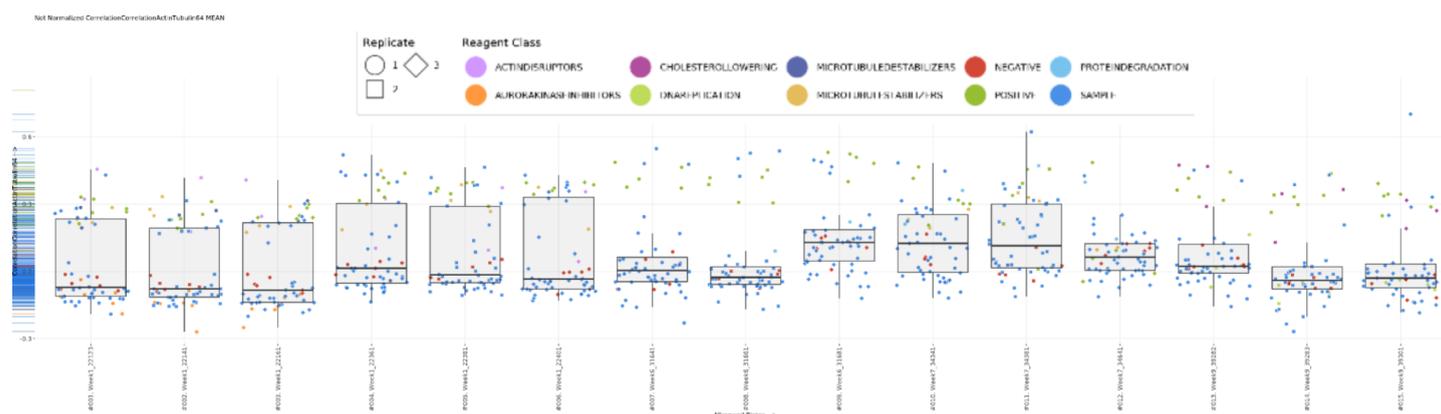


Figure 39. Analytical feature 'correlation correlation Actin-Tubulin 64 MEAN' boxplot before plate normalisation. The median of the feature 'correlation correlation Actin-Tubulin 64 Mean' differs strongly between plates.

Figure 40 shows the means of the analytical feature 'correlation Actin-Tubulin 64' plotted per plate after plate normalisation. In this figure we can see that the means of all plates are closer to zero compared to before as in figure 39. The interquartile range has also been reduced. The divergence between the reagent classes also seems to be reduced compared to before plate normalisation.

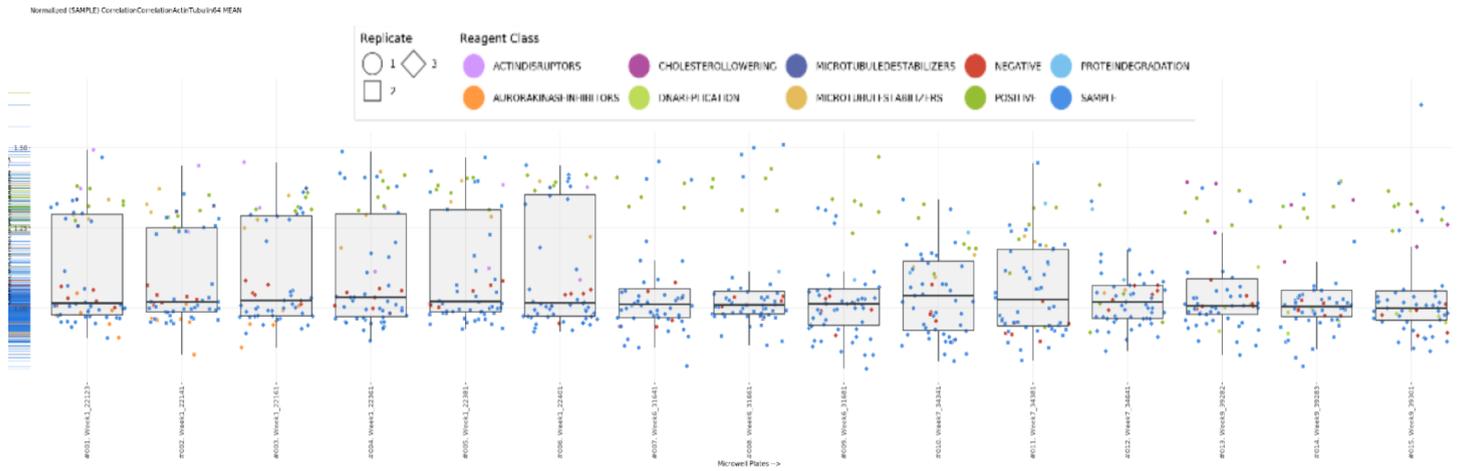


Figure 40. Analytical feature 'correlation correlation Actin-Tubulin 64 MEAN' boxplot after plate normalisation. The median of the feature 'correlation correlation Actin-Tubulin 64 MEAN' differs very little between plates, Q1 and Q3 do show variation between plates.

### 4.2.2 Data transformation

The plot in figure 41 shows the skewness of 100 random analytical features. In this figure, 18 features are highlighted to be corrected. In total the dataset required 59 analytical features to be corrected for skewness. Figure 42 shows the same analytical features as figure 41. Here we can see that after transformation of the features no significant skewness is remaining.

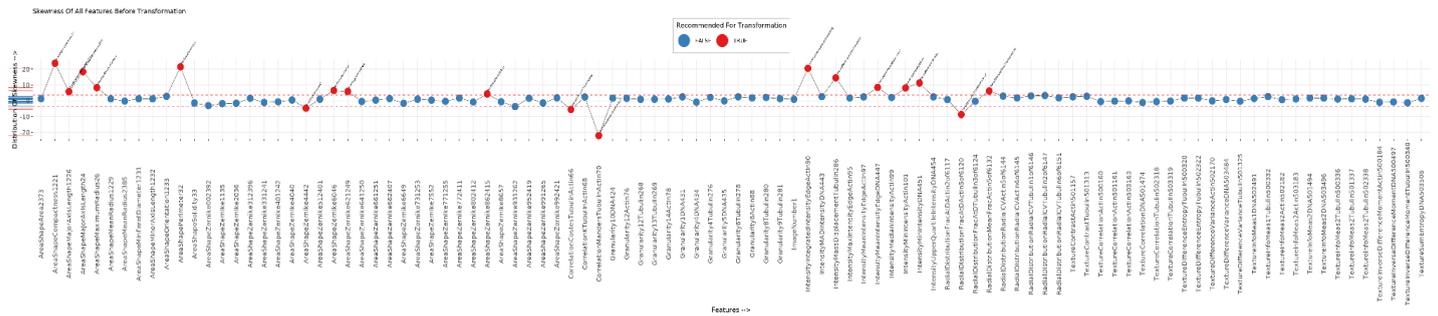


Figure 41. Skewness of all features before transformation. The x-axis shows 100 random analytical features, the y-axis shows the distribution of skewness of a scale of -20 to 20. The red features have a distribution of skewness outside the range of -4 to 4, and are selected for transformation. The -4 to 4 distribution of skewness range is displayed via the two red dotted lines.

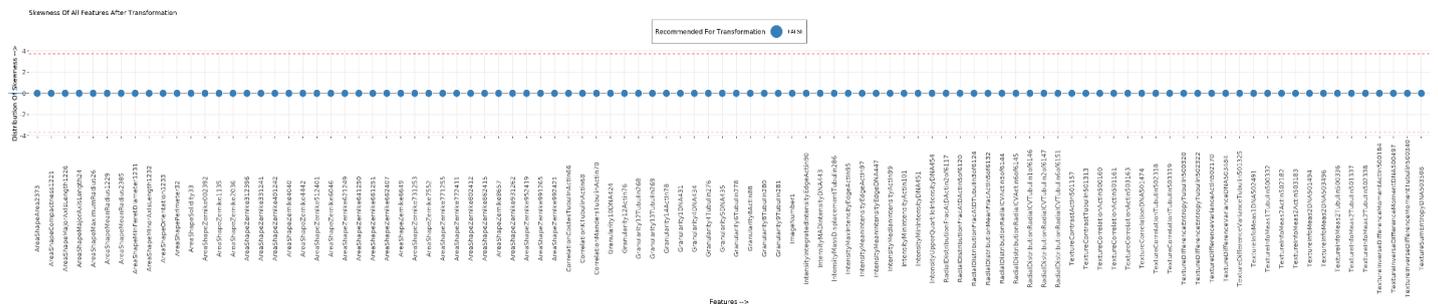


Figure 42. Skewness of all features after transformation. The x-axis shows the same 100 random analytical features as figure 37. The y-axis shows the distribution of skewness on a scale of -4 to 4. After the transformation zero features remain with a distribution of skewness outside the range of -4 to 4. The -4 to 4 distribution of skewness range is displayed via the two red dotted lines.

## 4.2.3 Scaling

The boxplot in figure 43 shows 100 random analytical features before feature scaling. In this figure we can see that there is variance in the robust Z-score means of the different features. Figure 44 shows a boxplot of the same 100 random analytical features of figure 43. In this figure the robust Z-score means are all scaled to zero.

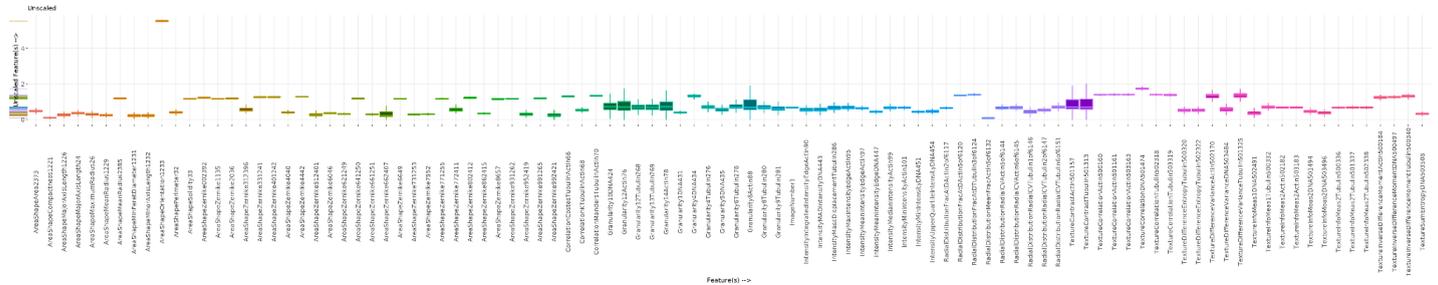


Figure 43. Boxplot of 100 random features before feature scaling based on plate by means of robust z-score. X- axis shows the features, the Y-axis shows the robust z-score of the features.

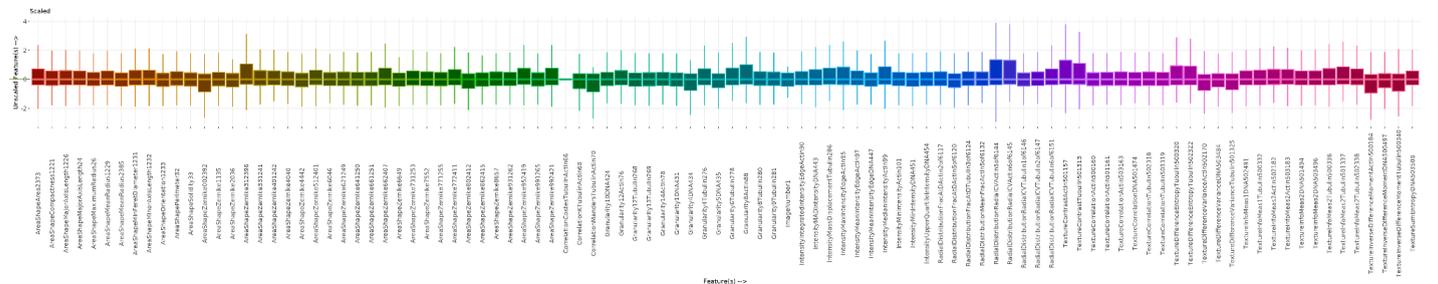


Figure 44. Boxplot of 100 random features after feature scaling based on plate by means of robust z-score. X- axis shows the features, the Y-axis shows the robust z-score of the features.

## 4.2.4 Missing data

Figure 45 shows a plot of 100 random analytical features with in red the amount of missing data per feature plotted, in Blue the total amount of useful data is plotted. Of the 396 features none are identified as features that might contain problems. Also zero features are selected for removal.



Figure 45. Plot of 100 random features. The X-axis shows the features, the Y-axis shows the counts of useful and missing data. Data summary of missing data analysis shown below plot showing that no features might contain problems and thus zero features are selected for removal.

## 4.2.5 Image curation

In the experiment ten images are removed from the training and test data that were identified as having one of the following errors: unequal distribution of cells, contamination of the well, imaging/focusing, high pipetting force, overly toxic reagent or other error resulting in no cells in image. Figure 46 shows the removed images.

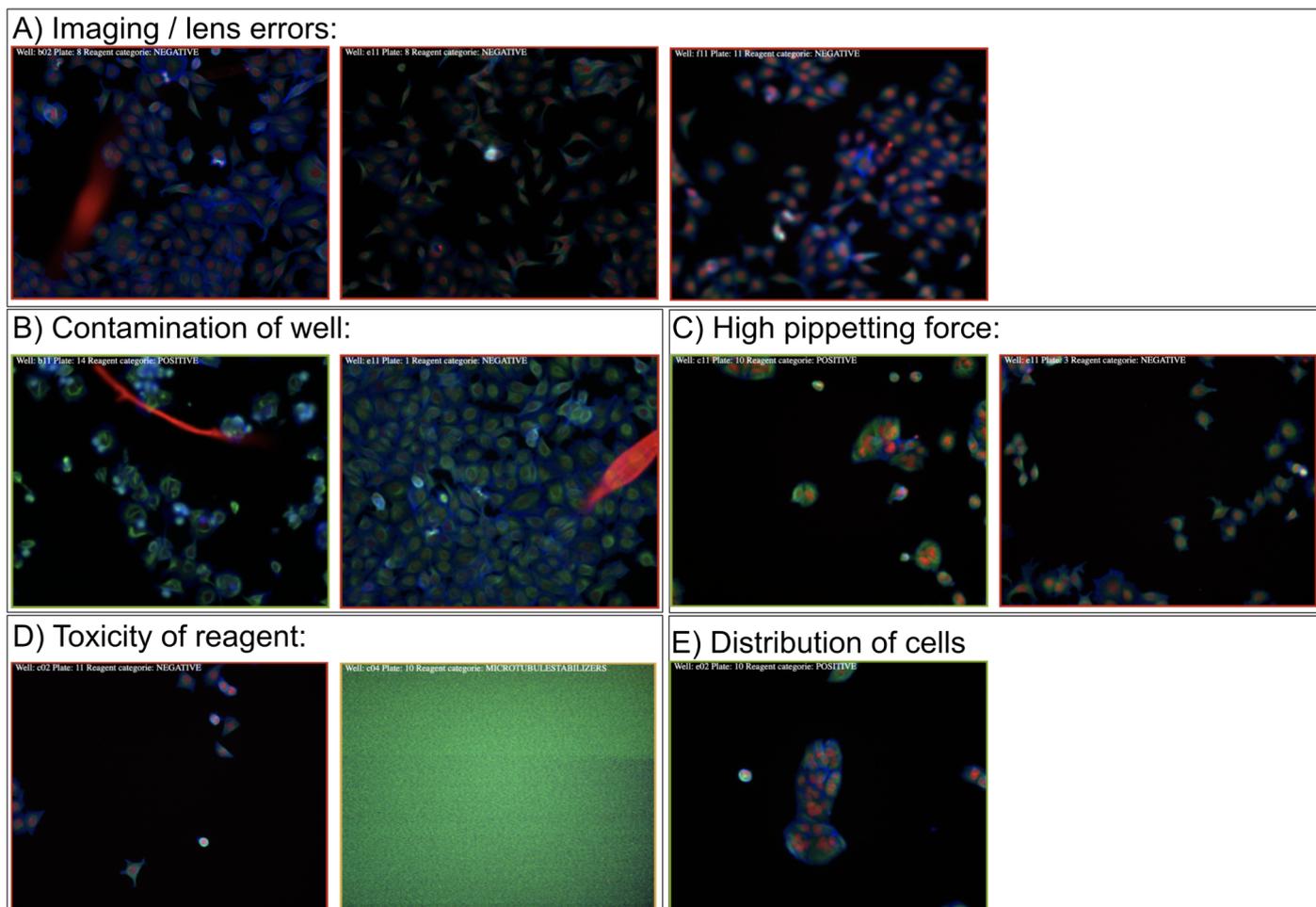


Figure 46. Overview of the removed images from the caie dataset. Subfigure A shows the images with lens or imaging errors. Subfigure B shows the images with contamination on the wells. Subfigure C shows images with a faulty distribution of wells due to a too high pipetting injection force. Subfigure D shows the images where the reagent was overly toxic. Subfigure E shows the image with a bad distribution of cells.

The image of well b02 plate 8 has a lens flare. Well e11 plate 8 shows a bright spot. Well f11 plate 11 is out of focus (blurry). Well b11 plate 14 and well e11 plate 1 have contamination making the numeric values related to the DAPI label unusable. Well c11 plate 10 and well e11 plate 3 only show cells on the borders of the frame, this artefact suggests pipetting in the centre of the well was done with too much force blowing the cells to the edges of the well. Well c02 plate 11 and c04 plate 10 have very little cells suggesting that the reagent was overly toxic. The cells of well e02 plate 10 are all clumped to one spot.

## 4.2.6 Dimensional reduction

Figure 47 shows a scree plot of the principal components made. This scree plot is of the curated dataset. Because the features stayed equal before and after image curation the dimensional reduction remained similar. In the plot a line is plotted for the elbow's method, on eigenvalue 18 and manually a line is added on eigenvalue 16. The elbow's method shows the point where the plot starts to level off. According to this line PCA 1 to 4 should be retained. Scree plots are mostly subjective and PCA 5 could also be retrained.

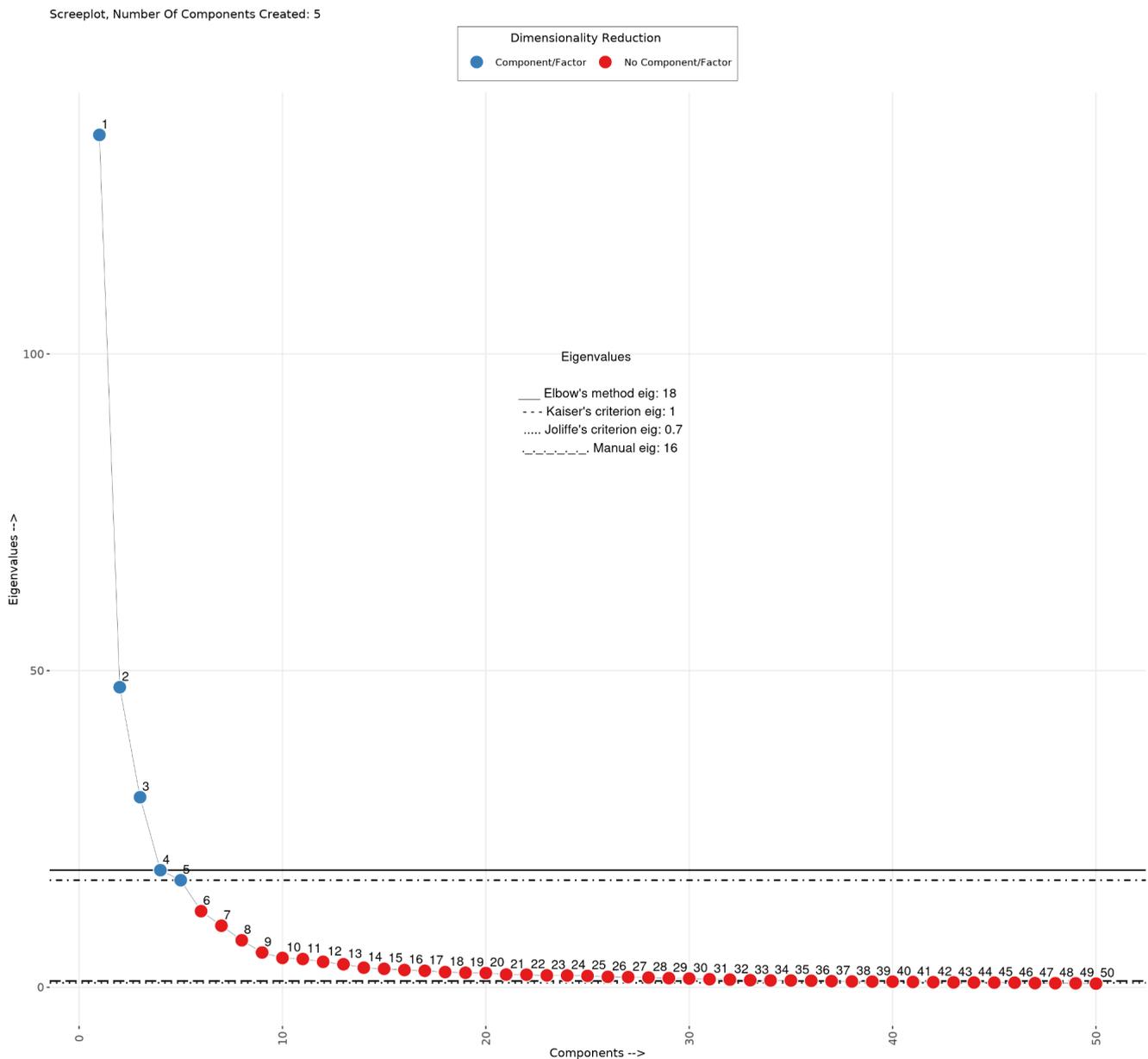


Figure 47. Scree plot of principal components based on the curated Caie dataset. PCA 1 to 50 is plotted, with the chosen PCAs, 1 to 5, highlighted in blue and the remaining in red. On the X-axis the components are numbered, on the Y-axis the eigenvalues are shown.

A correlation matrix of all features and the factor contributions of all features are shown in figure 48. The features shown in 48B are aligned with the correlation plot in 48A. In this figure we can see that PCA2 has a strong correlation (bottom right of 48A). PCA1 also shows groups of strong correlation but is more spread out over the feature categories (y-axis).

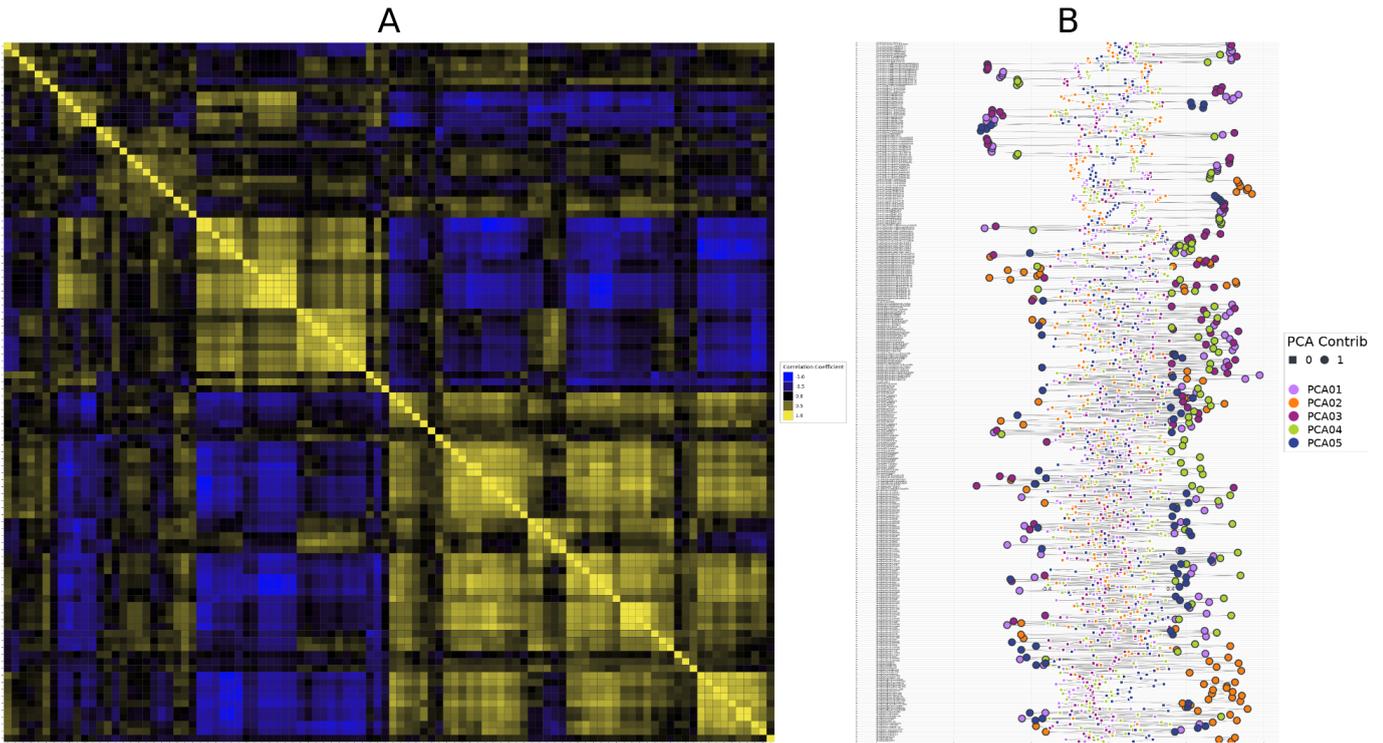


Figure 48. A) Correlation matrix of all features. B) Factor contributions of all features per PCA.

### 4.2.7 Hit Selection

The bar plots in figure 49 show the sampling distributions as the result of the 80/20 split for training and testing, the 10000 records sampling size and the 10 fold-cross validation. Figure 49A shows the sampling distributions for run 30 of the full dataset, and figure 49B shows the sampling distributions for run 30 of the curated dataset. In the full dataset the labelled data consists of 80170 records resulting in an oversampling of ~107.6%. In the curated dataset there are 75200 records resulting in an oversampling of ~114.2%. The training data for the positive and negative controls are almost equal in both the full and the curated dataset. And the training data for the microtubule stabilisers represent a smaller percentage than those aforementioned groups.

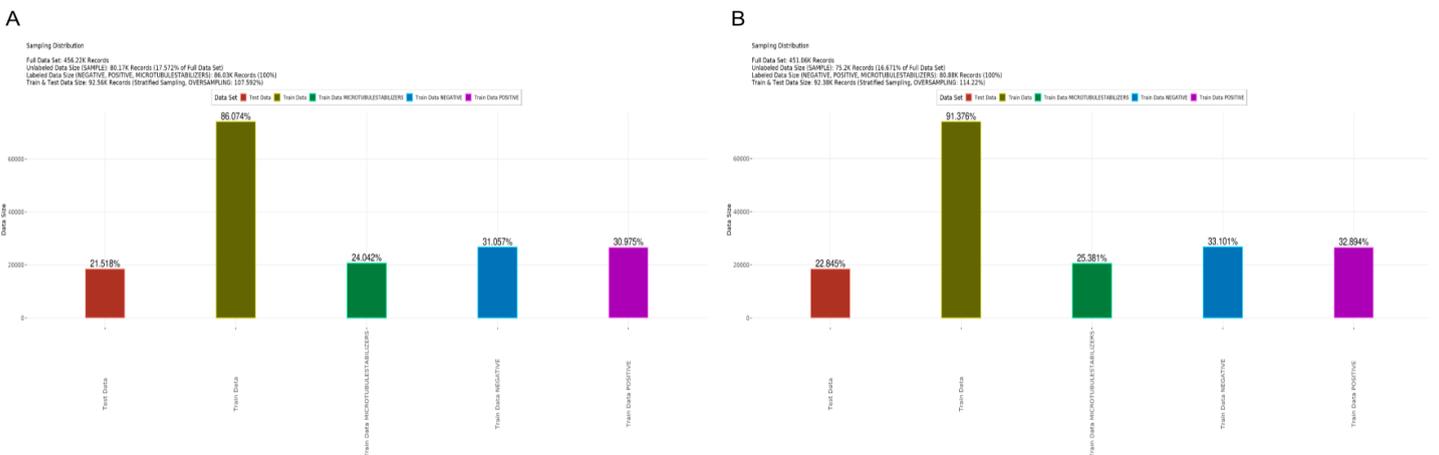


Figure 49. A) Sampling distributions for hit-selection on the full dataset. B) Sampling distribution for hit-selection on the curated dataset. Red is the test data, Yellow is the sum train data, green is the train data of the class microtubule stabilisers, blue is train data of the class negative control and in pink the train data of class positive control is given.

Two contour plots of the predicted classes based on full dataset and curated dataset are shown in figure 50. In both subfigures a contour is drawn that shows that the negative control is separated from the positive and microtubule classes. The contour plot of the curated dataset shows more separation of classes between positive and microtubule than is seen in the full dataset.

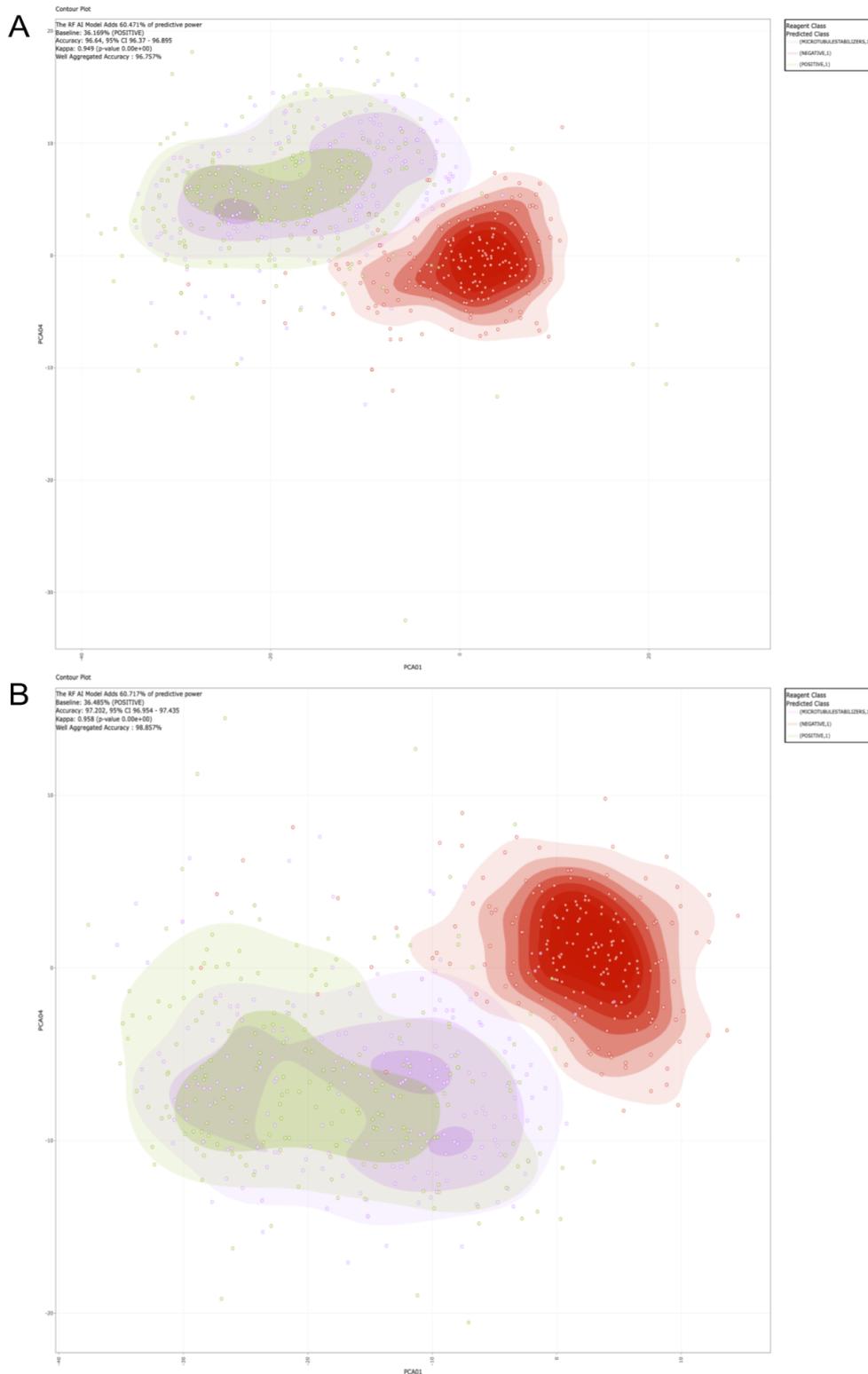


Figure 50. A) Contour plot of the predicted classes of the full dataset. The X-axis shows PCA1, the Y-axis shows PCA4. Purple is predicted class microtubule stabilisers, red is predicted class Negative and green is predicted class positive. B) Contour plot of the predicted classes of the curated dataset. The X-axis shows PCA1, the Y-axis shows PCA4. Purple is predicted class microtubule stabilisers, red is predicted class Negative and green is predicted class positive.

Figure 51 shows a ranked bar plot of the relative importance of the phenotypic distance per well. Part A of the plot is based on the full dataset run 30 and part B is of the curated dataset run 30. This plot indicates how strong the phenotypic change is between the different classes. Due to the plot being ranked based on phenotypic distance, the difference between microtubule stabilisers (yellow), negative control (red) and positive control (green) is highlighted. The X-axis only shows 300 wells because the results of all three replicates are combined per well.

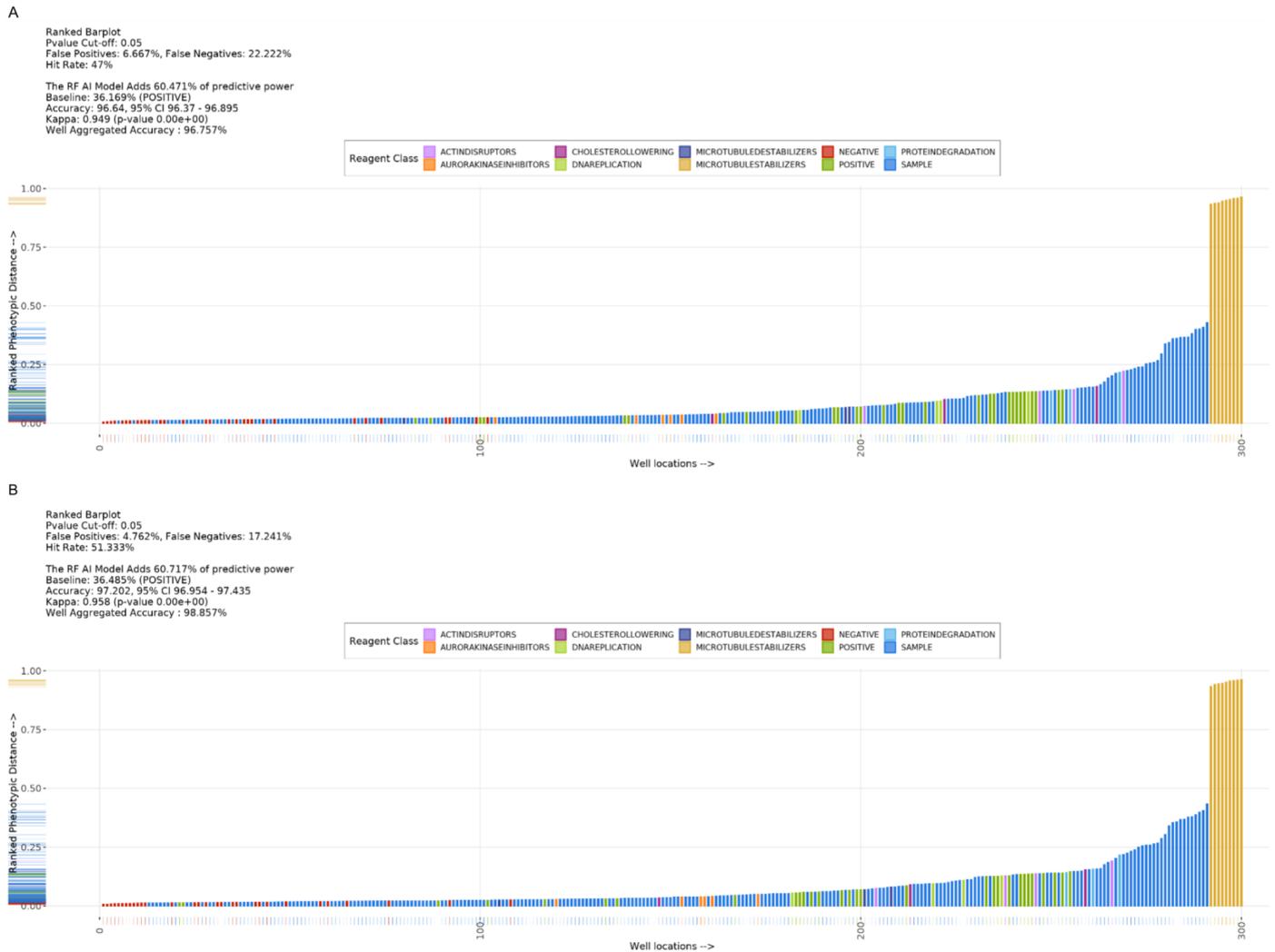


Figure 51. Ranked bar plot displaying the relative importance of phenotypic distance per well. A) Result of random-forest classification off the full dataset, B) Result of random-forest classification off the curated dataset.

Figure 52 shows the replicate outlier plot. Part A of the plot is based on the full dataset run 30 and part B is of the curated dataset run 30. In this plot per well the three different replicates are separately plotted based on their phenotypic distance. This plot indicates that the reagent class sample is the least concentrated along its distance.



Figure 52. Replicate outliers plot. All 900 wells are plotted along the X-axis with distance/similarity plotted along the Y-axis. Visualising the difference in distance amongst replicates of the same well. A) Result of random-forest classification off the full dataset, B) Result of random-forest classification off the curated dataset.

The log 10 corrected P-value of the phenotypic distance of the three replicates combined just as in figure 52 but grouped per reagent categorie instead of ranked based on relative importance is seen in figure 53.

Part A of the plot is based on the full dataset run 30 and part B is of the curated dataset run 30. It is to be noted that the Y-axis on figure 53B runs from 0 till 250 and the Y-axis on figure 53A is ranged 0 till 300.



Figure 53. Scatterplots showing the log<sub>10</sub> corrected P-value of phenotypic distance of the three replicates combined per well. A) Result of random-forest classification off the full dataset, B) Result of random-forest classification off the curated dataset.

### 4.2.8 Model

Figure 54 shows the balanced accuracy of class microtubule stabilisers, the mean of the curated and non curated lines are also drawn. The kappa scores and specificity are also plotted for both models. It is important to note that the plot only shows the range of the accuracy to better visualise the effects. The plot shows that higher scores are achieved by using the curated dataset however some overlap between the two models is visible.

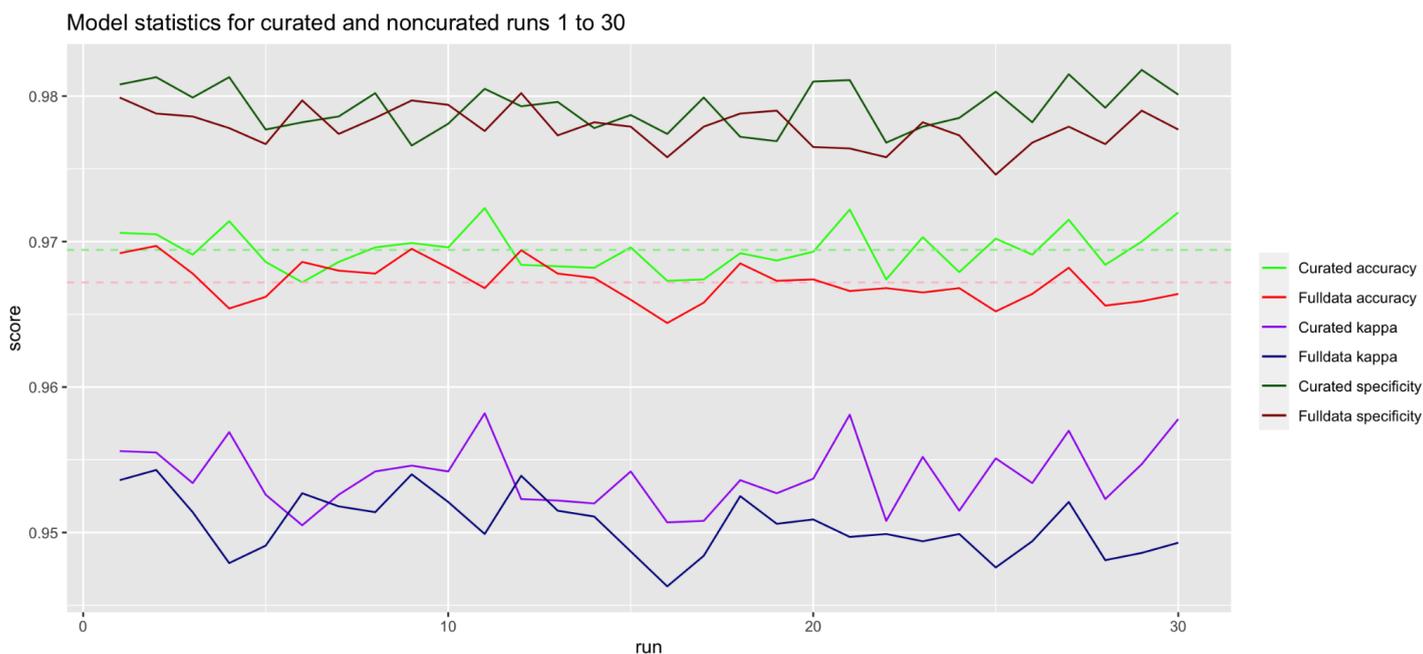


Figure 54. Accuracy scores of curated dataset (green) and full dataset (red) along 30 runs. Solid horizontal lines show the mean, interrupted lines show the min and max. X-axis shows the run number, the y-axis shows the accuracy score.

The mean and standard deviation (sd) of the curated and full set are shown in table 8. The sd is higher in the curated set suggesting that the results are slightly less reliable.

Table 8. Mean accuracy and standard deviation of the 30 runs for the curated and full dataset.

Dataset	Mean accuracy	Standard deviation
curated	0.9799867	0.001422173
full	0.9785233	0.001240879

The results as described in table 9 show that the predictions made based on the curated and full set of training data has no difference in the predicted compounds. How often different compounds are predicted do differ by small amounts.

Table 9. Sum of the compound occurrence counts in top ten predictions for full dataset and the curated dataset 30 runs.

Compound:	MOA:	Full dataset:	Curated dataset:
AZ-N	Unknown	106	105
Bleomycin	DNA-replication/damage	112	116
Docetaxel	Microtubule	82	79

## 5 Discussion and conclusion

The results achieved in the experiment are believed to be promising, however it must be recognised that in the experiment several limitations were not properly taken into account. The limitations of the experiment and choices made are discussed in paragraph 5.1. The stratoViewer module is discussed in 5.2 and the quality of the models are discussed in 5.3.

### 5.1 Experimental design

The experiment is designed to test the effects of curating an HCS dataset based on the images, by measuring these effects, conclusions can be drawn on the added benefit of the StratoVieweR module. Table 2 and figure 17 displayed the variables and their interactions in the experiment. One variable not considered in the experimental design is the impact the expert has on the experiment. The curation of the images was done by a single person. Not properly compensating for this has one of three effects on the final results of the experiment. Either the curation was done perfectly, resulting in an overestimation of the effects on image curation. The curation was done poorly, resulting in an underestimation or even a negative effect. Or finally the curation was done in between poor and perfect, resulting in a measured effect between those aforementioned.

The experiment has been performed on a single dataset. The Caie dataset is believed to be of good quality, see paragraphs 4.2.1, 4.2.2 & 4.2.4. Seeing the dataset used in multiple published papers for validation of methods in HCS data analysis, sets a precedent for this dataset as an exemplified dataset<sup>17-20</sup>. Supervised clustering on this dataset has already proven to achieve great results achieving an accuracy of 97%<sup>21</sup>. Improvements on the accuracy of predictions achieved by supervised clustering of this dataset are expected to be minimal due to the high accuracy of predictions achievable without curating. Based on the results gathered in this experiment it is not possible to blatantly assert the conclusion to datasets other than those tested here. However, given the assumption that lower quality data has more benefit from manual curation, it can be said with confidence, given the stated criteria, the increase in accuracy should manifest in other experiments using similar methods on different datasets. If those datasets are of lower quality eg. systematic errors, inconsistency of replicates, extreme outliers, interaction of reagent with probe, and plate or batch effects; it is expected to see a higher increase in overall accuracy.

Predicting the MOA of reagents using HCS experiments is commonly done by automated clustering techniques.<sup>18, 22</sup> The choice for a random forest clustering method reflects a common method for HCS experiments. The relatively small size of the Caie dataset does introduce the chance of overfitting giving an overestimation of accuracy scores for our models. Overfitting could even be extra prevalent in a curated dataset due to its smaller size skewing the results even more.

## 5.2 StratoVieweR

### 5.2.2 Comparison to Omero Iviewer

StratoVieweR is designed and developed to be a fully fledged image browser and viewer for HCS images. Main benefit of StratoVieweR is the full integration into the StratoMineR platform. To review the benefits and disadvantages of this connection and assess the functionalities as an HCS image browser and viewer, a comparison is made to Omero Iviewer. The comparison will incorporate the platemap overview, image preview and image comparison features of both applications.

#### 5.2.2.1 Platemap

In both Iviewer and StratoVieweR wells are selected via a grid that represents a microwell plate. The plate view from Iviewer only offers a view with images, see figure 55. StratoVieweR has the option to view a schematic plate or an overview with images just as Iviewer, see figure 33. An advantage that Iviewer has over StratoVieweR is that full colour images are used as the thumbnails instead of only single channel grayscale images. However Iviewer is limited to the full scale image, this can be a disadvantage if more than three channels are included in the dataset. If more than three channels are available, it is impossible to use the fourth channel as a thumbnail. It is also not possible to change the colours used for each channel making some channels very hard to see in the thumbnails. StratoVieweR does give full control over the thumbnails being shown, excluding the option for RGB thumbnails.

The schematic platemap shown on the right side of figure 33 is a feature unique to StratoVieweR. The schematic overview gives a fast overview of the reagent classes. The reagent classes are also visible as a thin coloured border around the thumbnails. The colours and reagent classes are shown in the legend. Another advantage of Iviewer versus StratoVieweR is the quick preview of all the fields of a well, as shown in the bottom of figure 55. StratoVieweR does not have this feature and requires the user to select a field to view.

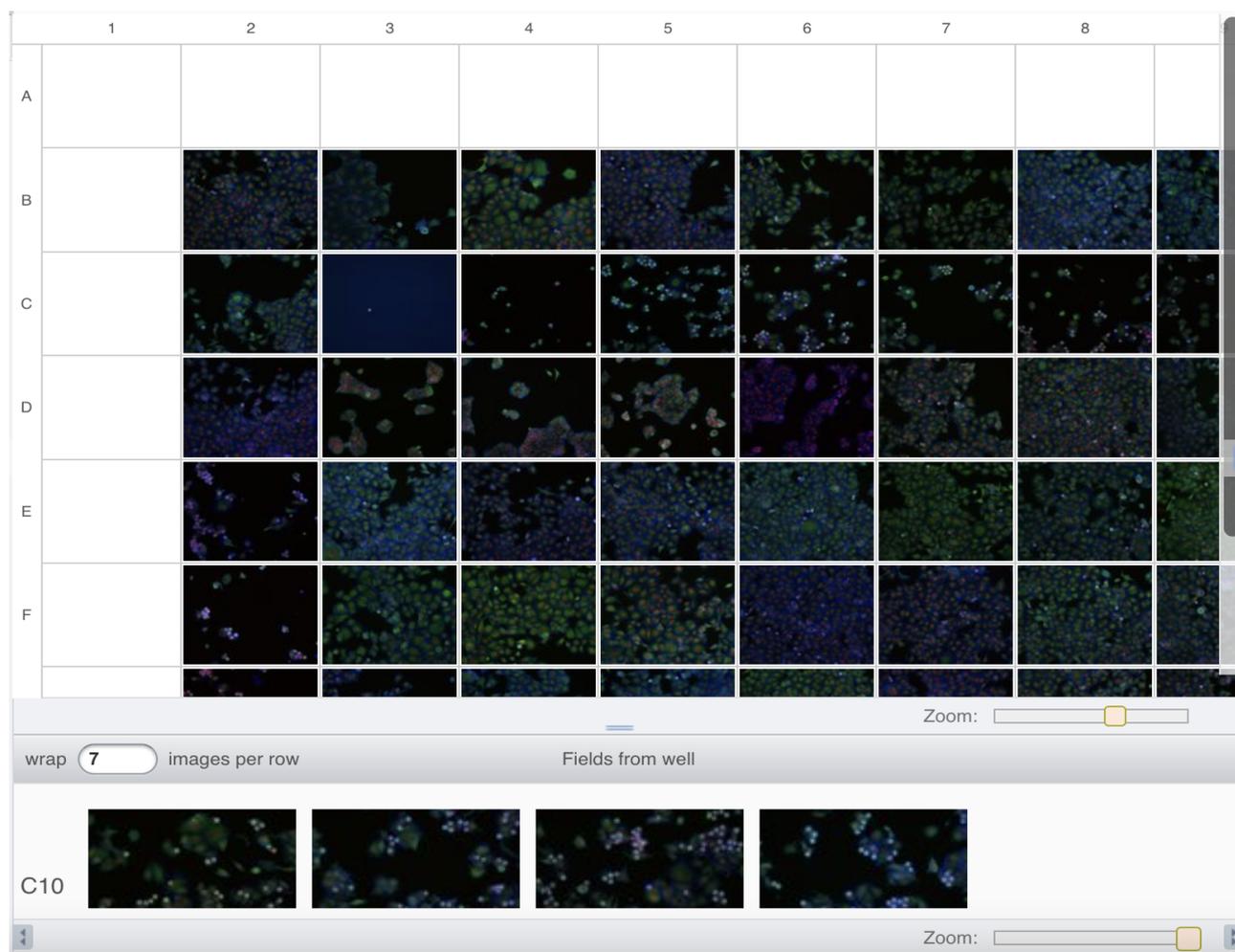


Figure 55. Screenshot of a platemap with full colour thumbnails in Iviewer. Increasing and decreasing the size of the platemap can be done via the slider labelled 'Zoom:'. The first and last columns and rows of this plate do not contain images. A second set of thumbnails is shown below the platemap displaying the fields of a selected well.

As can be seen in figure 55 the Iviewer platemap is zoomable, this makes it easier for users to review a small part of the plate, especially in large plates e.g. 384 well and 1536 well plates. This however does introduce horizontal and vertical scrollbars that are often seen as an anucance. If a 1536 well plate is loaded into StratoVieweR this results in an horizontal scrollbar.

Being able to access all the images in an intuitive overview can be very beneficial, however if loading the interface takes too long the added benefit of the preview will be omitted. In Iviewer rendering ten plates sequentially resulted in a mean loading time of 2.01 seconds. StratoVieweR has more options for loading a platemap thus different metrics are needed for the platemap loading times. StratoVieweR also requires less time to load a platemap after it has been rendered a first time, because of the implementation of reactive programming.

In StratoVieweR a plateview can be loaded without thumbnail images, on startup a mean loading time of 1.11 seconds is achieved. Re-accessing a platemap via StratoVieweR is instant. Loading a platemap with thumbnail images for the first time results in a mean loading time of 13.03 seconds. Re-accessing the platemaps with thumbnails takes a mean of 1.27 seconds. Figure 56 shows a boxplot of the measurements

of the loading times of the described scenarios. The variance in loading times for the Iviewer platemap is larger than for StratoVieweR.

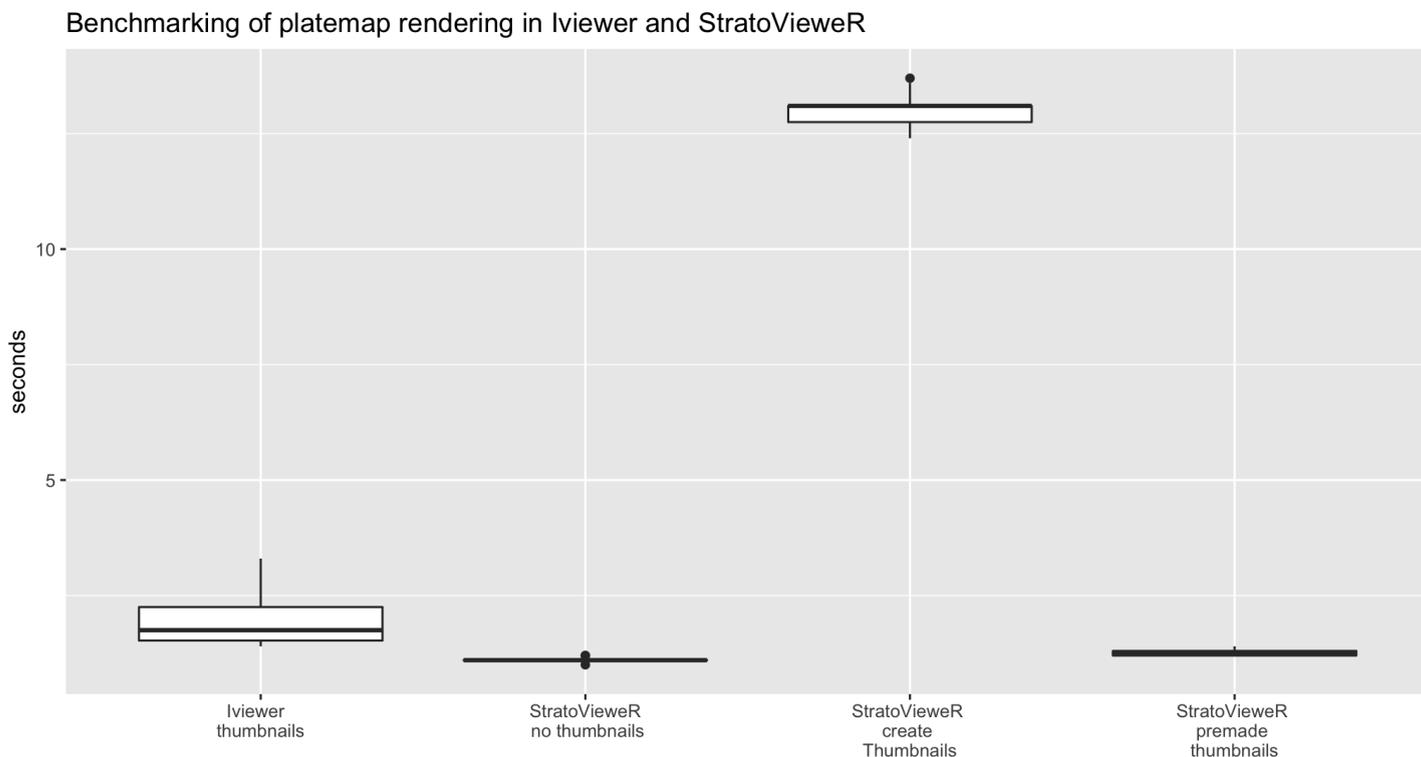


Figure 56. Benchmarking of platemap loading times for Iviewer and StratoVieweR. The loading times are shown as a boxplot. The plot shows that the Iviewer loading times show the largest amount of variance. Creating thumbnails on startup (third column) takes the longest. For all four methods ten measurements were taken.

### 5.2.2.2 Image Viewer

After the platemap has been used to select wells of interest, or in the case of StratoVieweR wells are selected based on feature information through the QC-module. Iviewer and StratoVieweR have interactive image viewing options. Screenshots of the image viewers in Iviewer and StratoVieweR can be seen in figures 57 and 58 respectively.

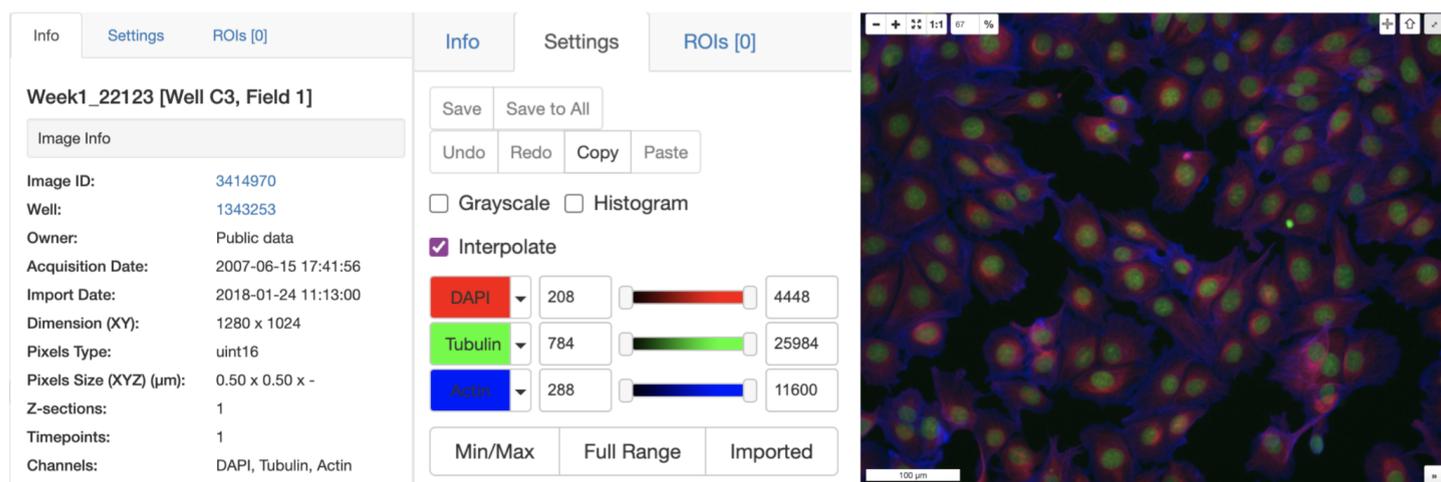


Figure 57. From left to right screenshots of: image info, image settings and image, of the Iviewer program. Image info shows image id, well id, acquisition date, import data, dimensions, pixel type pixel size, Z-sections, timepoints and channels of the image selected. The settings tab gives users control over the different channels and how they are represented in the image. The image is zoomable, can be increased to full screen and has a convenient size scale.

An advantage of Iviewer is the technical information given of the image. StratoVieweR does not provide users with this information. However StratoVieweR does give users information about the reagent class, in extension feature data can be retrieved via the connection to other modules in the StratoMineR platform. Control over the channels is handled almost equally between Iviewer and StratoVieweR. Iviewer offers a grayscale mode where this is not an option in StratoVieweR, the grayscale mode does require a full page refresh after use and can otherwise not be disabled. There is little to no noticeable difference in loading times of images on full resolution between Iviewer and StratoVieweR, this does however change when downsampling is enabled by the user, this decreases the loading times by at least 50 percent. The feature is lacking from Iviewer.

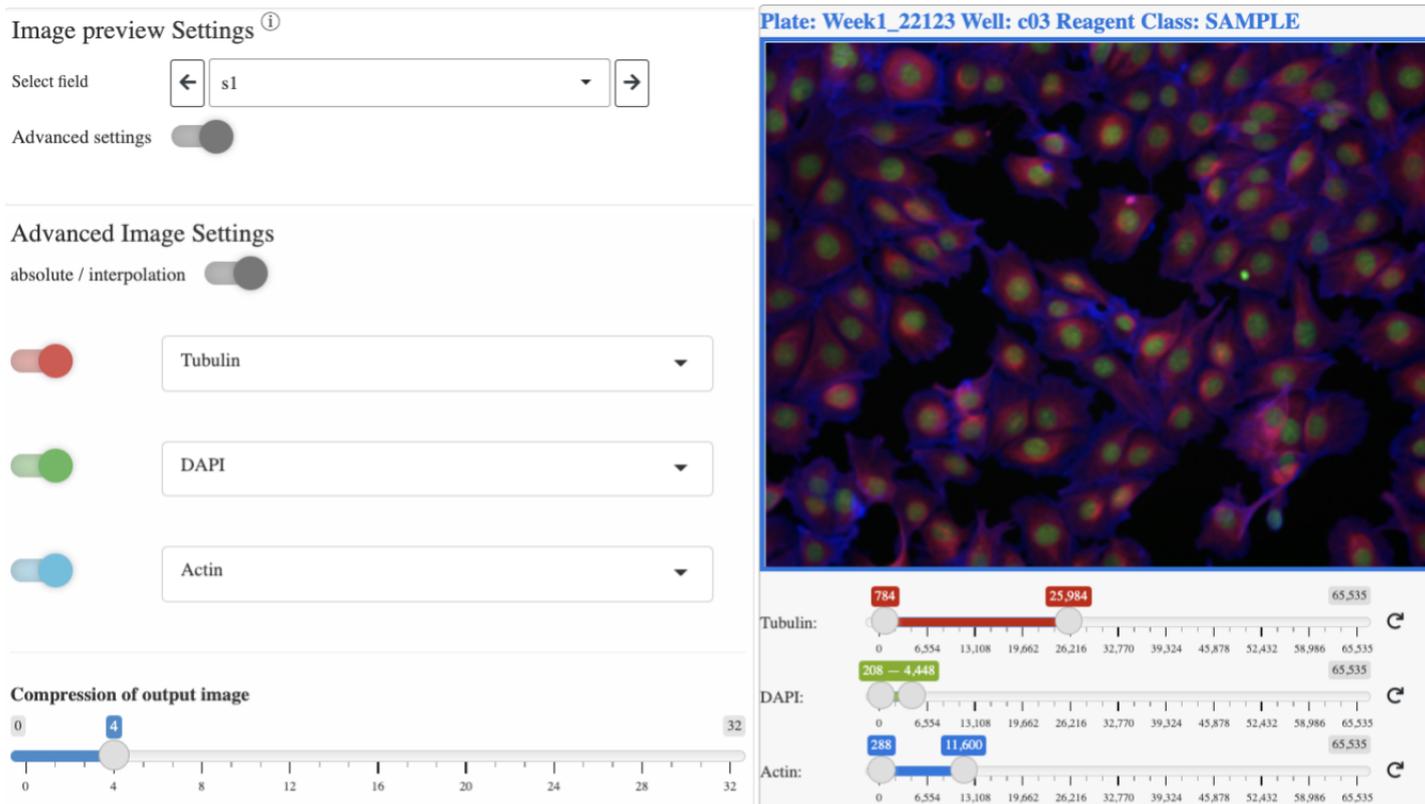


Figure 58. Left; general image settings are shown, these settings apply to all images on screen, and remain when new images are selected. Information of the well and image are shown as a single line of text above the image, the colour of the text and border of the image correspond to the reagent class. The image can be enlarged by clicking on it.

### 5.2.2.3 Comparing images

For image viewing both Iviewer and StratoVieweR offer similar features and the usability will depend on the user's preference. The differentiating feature of StratoVieweR apart from the integration with a data analysis platform is the ease of which images can be compared. Figure 59 shows a screenshot where three images are compared using StratoVieweR. Iviewer opens every selected image in a separate tab of the browser whereas StratoVieweR can show up to three images side by side. This makes it easier for the user to compare images and also enables automatic scaling of images in advantage of the comparison. The scaling infers a form of normalisation for image comparison. Iviewer mimics the behaviour of automatic scaling by letting users copy and paste image settings between images. The scaling feature results in a reduction of a five-fold mouse clicks to be performed by the users. While testing the feature of copy and pasting the image settings in Iviewer only worked after manually adjusting image channels, this seems to be a bug.

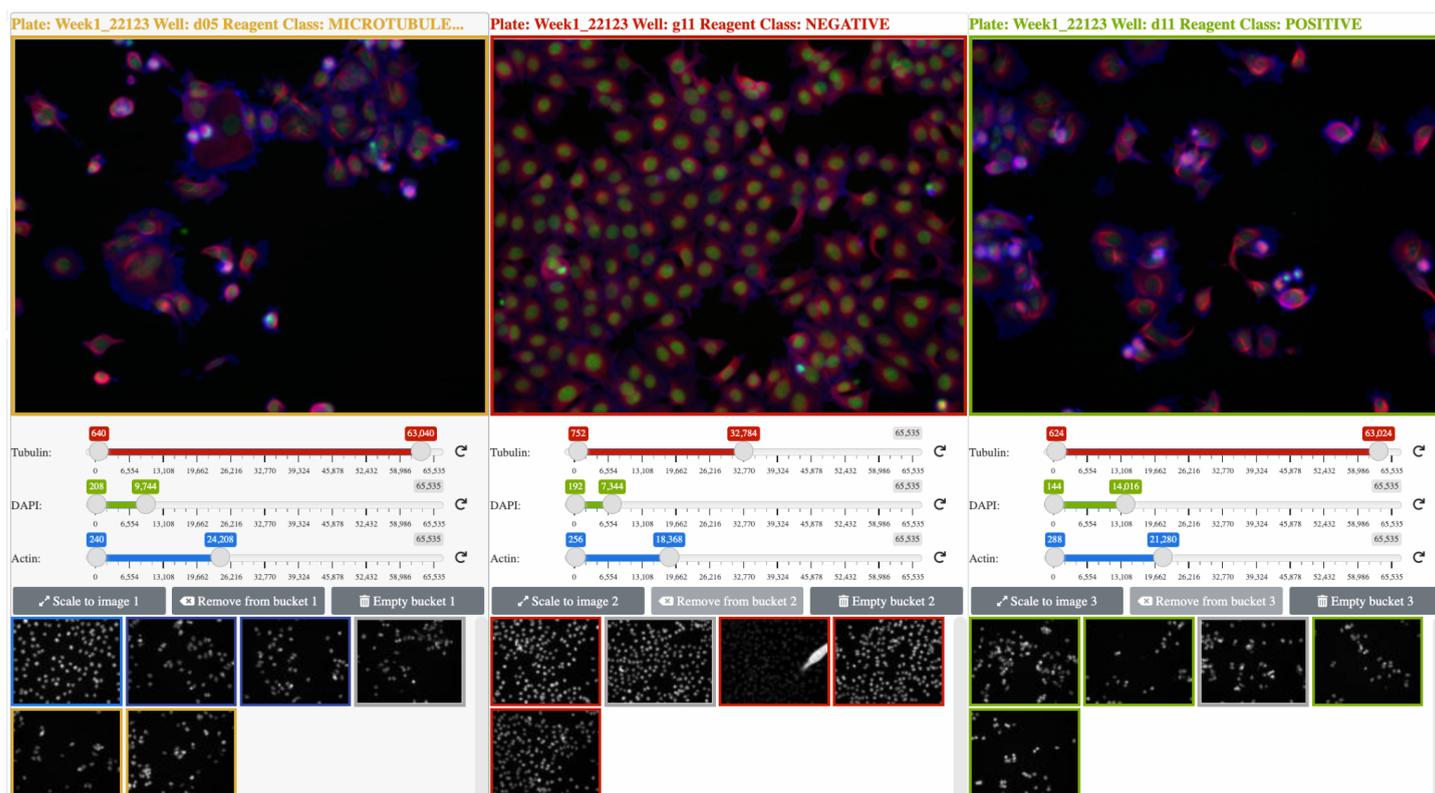


Figure 59. Screenshot of the image comparison mode in StratoVieweR, three images are selected. Above each image the plate and well locations as well as the reagent class is written. Clicking on an image enlarges it in a popup window. Below the images, sliders give users individual control over channels.

### 5.2.3 Reviewing outliers

The experiment has focused on curating outliers to review if curation impacts the quality of supervised classification models. Whilst this method alone has shown to improve classification it does not prove all of the benefits created by the connection of HCS image with numeric data. When an outlier is identified in the numeric data, the vast majority of these outliers is the result of an error e.g. systematic errors, inconsistency of replicates, interaction of reagent with probe, and plate or batch effects. Due to the vast majority of these outliers being caused by errors, it has become standard practice, in numeric data only driven analysis, to filter out all these data points. Assuming all outliers to be errors can however limit the effectivity of an HCS experiment, by ignoring outliers that are in actuality caused by compounds that show an exceptionally strong positive effect.

Visualising this situation is done by selecting two sets of outliers, see figure 60. The plots and images are based on a single plate with five replicates of a dataset by Rhoban et al. (2017).<sup>24</sup> The boxplot in figure 60A shows wells that consistently differentiate over multiple plates in a single well; this plot is used to identify systematic errors. The plot shows that well A21 has a consistently high edge cell intensity on the AGP channel. Removing this outlier from the dataset based on the plot is justifiable. When reviewing the images of well A21 in figure 61A and comparing them to the positive and negative controls no errors are visible. It can not be concluded that the well is in fact a positive outlier, but the images do give a cell biologist the opportunity to make a better assessment of the data.

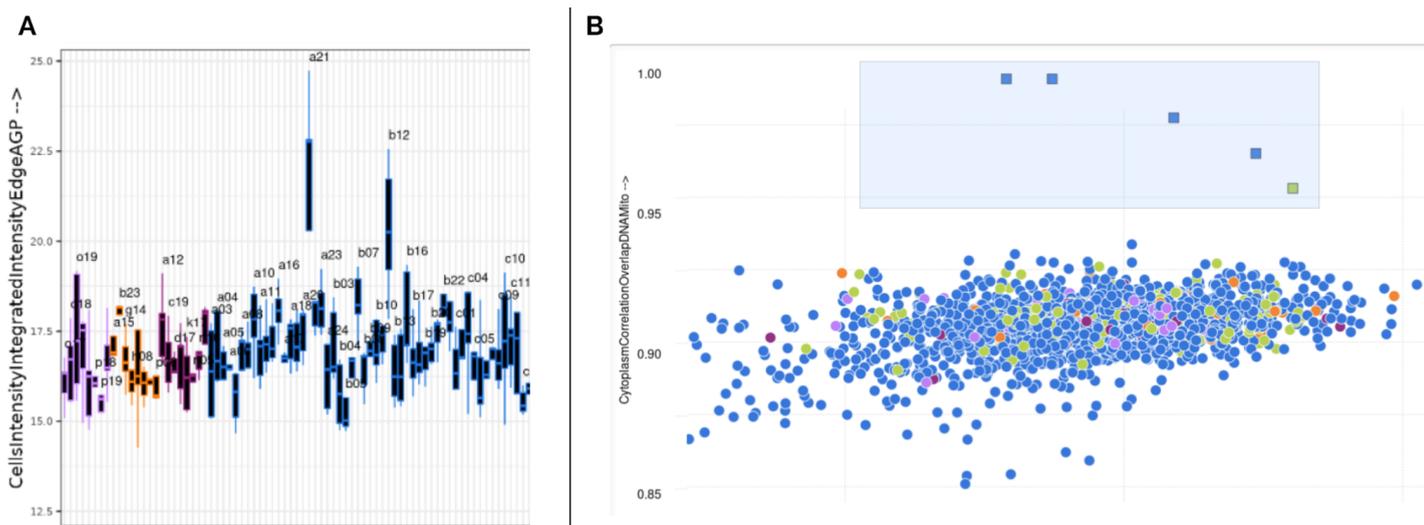


Figure 60. A) Boxplot showing for visualising the systemic outliers. This plot highlights wells that differentiate consistently over all plates. Well A21 and B12 seem to differentiate strongly for this feature. B) Scatterplot with a selection made over five wells. Colours correspond to the reagent categorie, blue for sample and green for positive control. Both plots are based on a dataset by Rhoban et al. (2017)<sup>24</sup>.

The scatterplot shown in figure 60B is also based on the dataset by Rhoban et al. (2017)<sup>24</sup>. A selection is made of five wells, along the x-axis from left to right, well B01, C01, D01, E01 and F01 from replicate 2. A selection of outliers corresponding strongly with a specific part of a single plate should trigger the curiosity of the data analyst. Figure 61B shows the images of the wells, comparing these images, wells C01 and D01 show a similar error, and B01, E01 and F01 appear free of this error. Given this extra information the decision can be made to only remove C01 and D01 from the dataset due to obvious errors and leaving B01, E01 and F01.

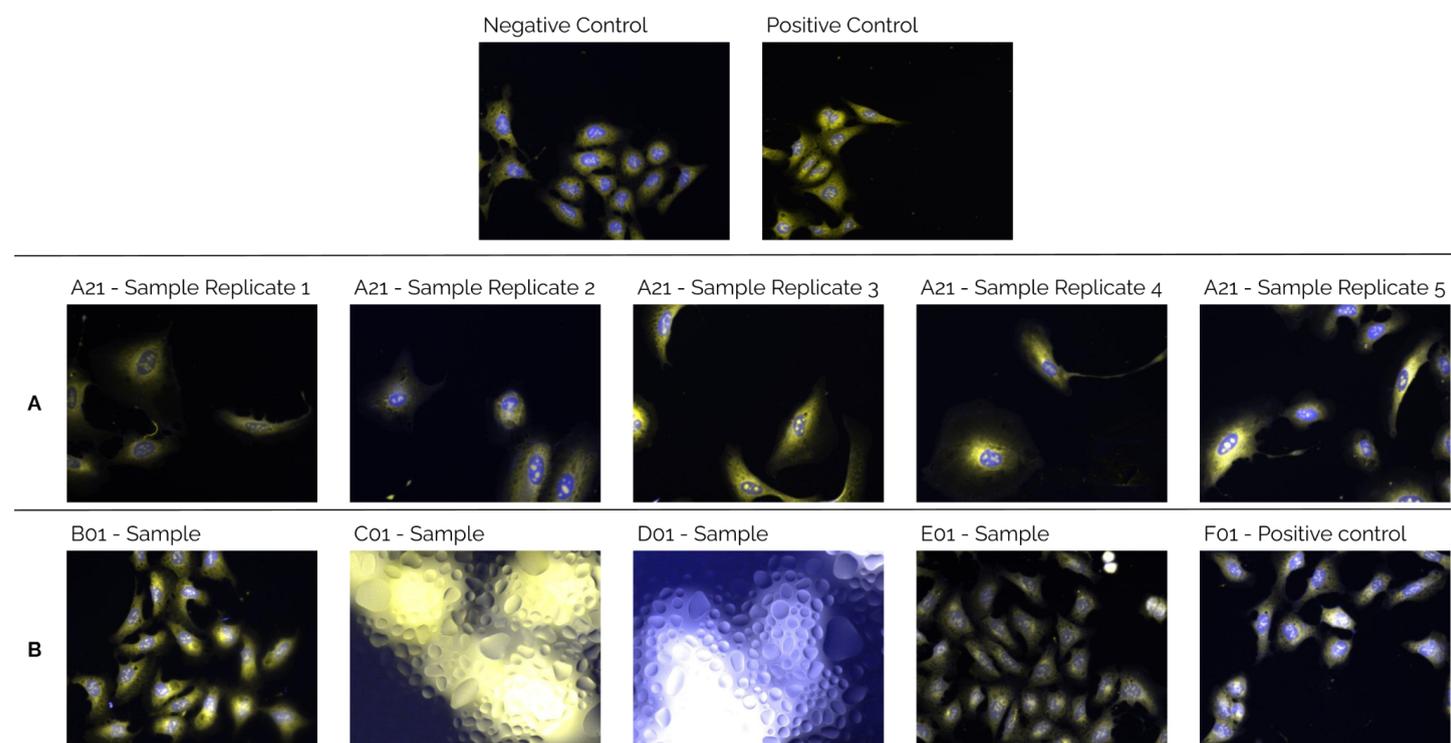


Figure 61. Images of outliers highlighted from the plot in Figure 60 from a dataset by Rhoban et al. (2017)<sup>24</sup>.

## 5.3 Data Analysis

The aim of the project is to determine the impact of uniting HCS image data with the numeric feature data. It is expected that manually curating images of a dataset using StratoVieweR increases the accuracy of the predicted classes. Testing this hypothesis is done by comparing the hit-selection models of a curated and a non curated dataset. In figure 54 of results some overlap between the results of the curated and the non curated scores can be seen. This means that even though the means differ between the two methods there is some overlap for specific runs. In figure 60 a boxplot of the overall accuracy is shown. The boxplot shows that the curated dataset, with its complete interquartile range (IQR) higher than that of the full set, has an overall better accuracy than the full dataset. To test this observation a one way anova will be run.

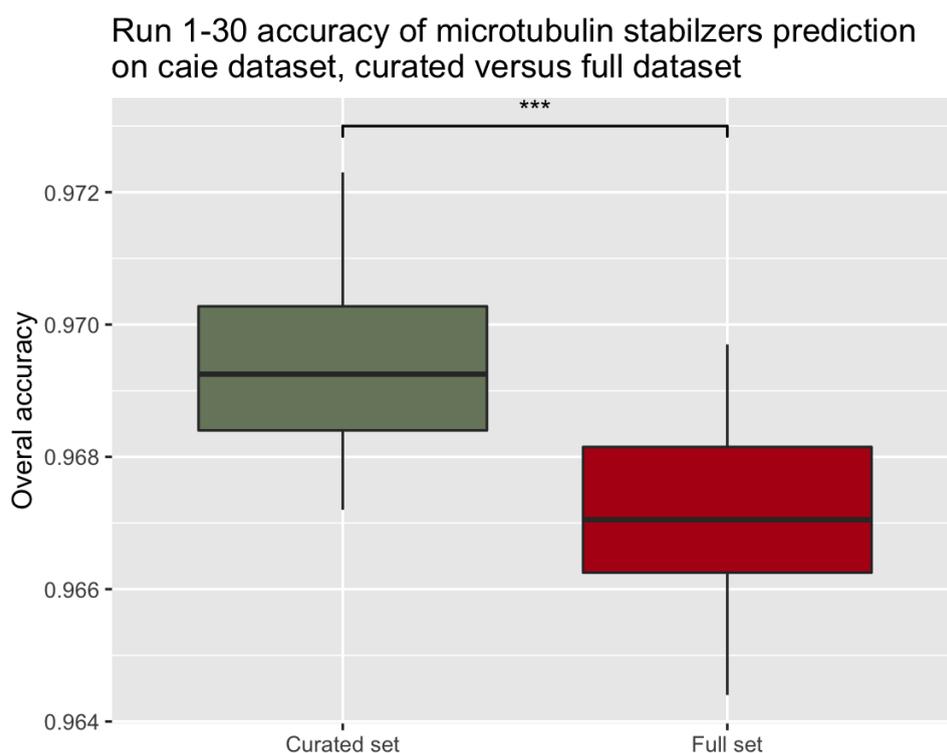


Figure 62. Boxplot of the accuracy scores of run 1 to 30. The y-axis shows the overall accuracy. The three asterisk represent the significance between the groups of a P-value smaller than 0.001.

### 5.3.1 Accuracy scores, ANOVA

To test if there is a significant difference between the accuracy achieved by curating the training dataset an ANOVA test will be executed. To ensure the validity of the ANOVA test four assumptions must be met.

Independence of the observations. All observations (accuracy scores of the model) can only belong to one group. By design of the experiment this assumption is met, there is no relation between the observations made for the curated as for the non curated models.

No significant outliers. The data for both groups are not allowed to have significant outliers. This is validated using the boxplot in figure 60. In the boxplot a significant outlier will be shown if outside of the interval calculated by the following formula:

$$I = [q0.25 + 1.5 \cdot IQR; q0.75 + 1.5 \cdot IQR]$$

Figure 60 shows none of these occurrences of outliers.

Homogeneity of variance. Variance should be close to equal between the two groups, if the group size is equal the results of the ANOVA are generally robust enough. In this experiment the groups are of equal size, and looking at the variance in table 8 very little difference between the groups can be seen. To ensure that this assumption is met a Levene test is performed. This test resulted in a P-value of 0.861, this being above 0.05 homogeneity of variance is assumed.

Normality. The two groups must follow a normal distribution. The first step to review the distribution of the groups is a visual assessment of the histograms. If the data is normally distributed the histogram should follow a bell curve, as can be seen in figure 61. To further test if the data follows the normal distribution a normality test is done. Due to the small sample size, the Shapiro-Wilk test is allowed. The results of this test are shown in table 9. For both the full and curated dataset the W-value is high, a high W-value suggests a normal distribution. This suggestion is confirmed by the P-values of both tests being above 0.05 thus not rejecting the H<sub>0</sub> that the data is normally distributed.

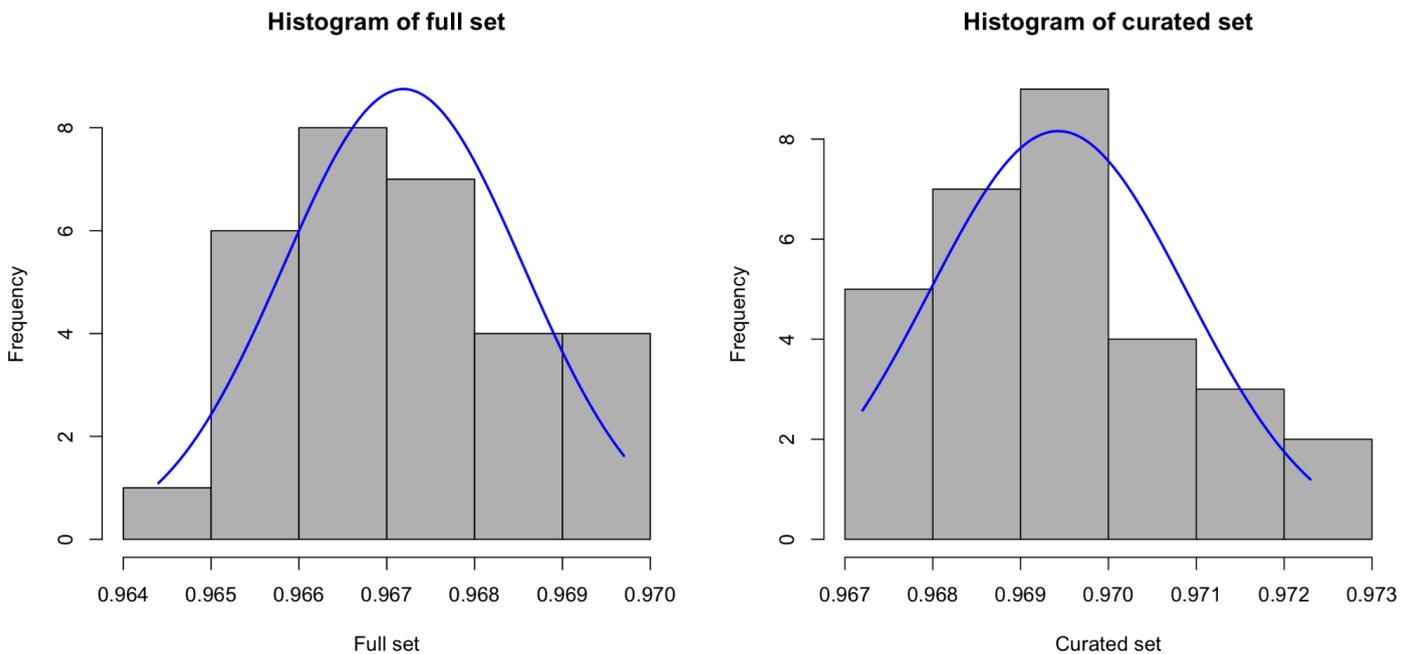


Figure 63. Histogram of the frequencies (y-axis) of the accuracy scores (x-axis) of the full set and curated set. Both distributions appear to follow a bell-shape suggesting it conforms to a normal distribution. A normal distribution is plotted in blue.

Table 10. Shapiro-Wilk normality test results, on the accuracy frequencies for the full dataset and curated dataset.

Data:	W-value:	p-value:
full data set	0.9792	0.8038
curated data set	0.95563	0.2384

Now that all the assumptions are met to execute an ANOVA test the H<sub>0</sub> and H<sub>a</sub> are defined as follows: H<sub>0</sub>, there is no significant difference between the accuracy of the model trained on the full dataset and that of the model trained on the curated data. H<sub>a</sub>, there is a significant difference between the accuracy of the

model trained on the full dataset and that of the model trained on the curated data. Table 11 shows the results of the ANOVA test. The test shows gives a P-value of  $8.99 \cdot 10^{-8}$  this value is smaller than 0.05 thus  $H_0$  is rejected and we can conclude that there is a significant difference in accuracy. The boxplot in figure 60 has shown that the means of the accuracy for the model trained on a curated set is higher than that of the full dataset, combining the results of the ANOVA, it can be concluded that there can be gained a significant increase of accuracy when curating the training set using StratoVieweR based on images over just the numeric data preparation.

Table 11. One way ANOVA on overall accuracy. The three stars show the significance.

	<b>Df</b>	<b>Sum Sq</b>	<b>mean Sq F</b>	<b>value</b>	<b>Pr(&gt;F)</b>
condition	1	7,50E-02	7,50E-02	37.33	8.99e-08 ***
Residuals	58	1,17E-01	2,01E-03		

### 5.3.2 Specificity

Thus far we have discussed the accuracy scores of the models. The accuracy scores give a good estimation of the overall quality of the models but are not conclusive. Specificity is a metric that gives the ratio of true negative predictions over the number of all negative predictions. In this experiment it is thus the ratio of true not microtubule stabilisers over all wells predicted to be something else than microtubule stabilisers. With the goal to predict microtubule stabilisers, this metric gives a good measure of the quality of the model. When analysing a fast library of unknown reagent receiving a high accuracy but a low specificity can result in too many promising compounds needing to undergo further research just to be discovered not to perform the targeted MOA. Figure 54 of results shows that the specificity is slightly higher over most runs for the model trained on the curated dataset, figure 62 shows a boxplot reaffirming that observation. The increase in specificity with a mean of 0.1% is minimal. However this does show that the curation has not had a negative impact on specificity. The high specificity scores, ~0.979 on curated data and ~0.977 on non curated data, shows the excellent quality of the predictions.

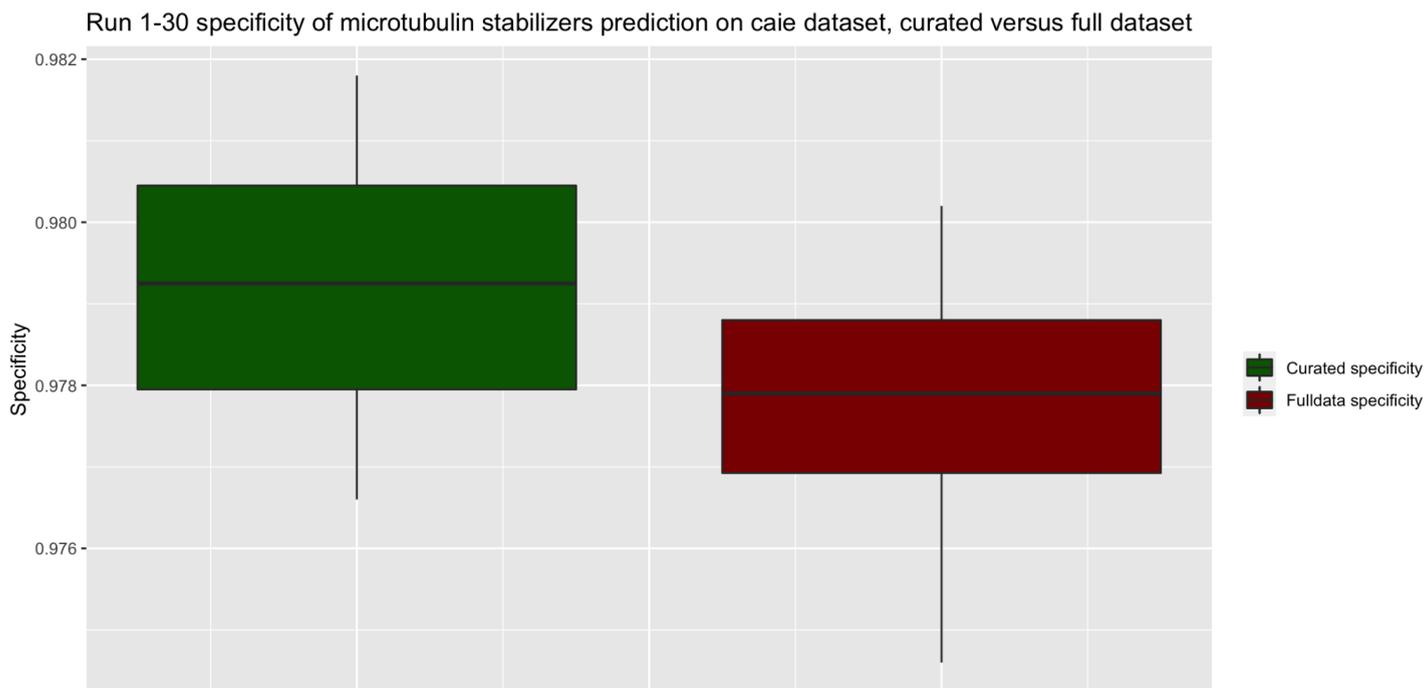


Figure 64. Boxplot showing the specificity of the 30 models trained on the curated dataset and the full dataset. The boxplot shows that the IQR of the model trained on the curated data is above the median of that of the full data set models.

### 5.3.3 Overfitting

In discussing the experimental design the problem and consequences of overfitting were explored. Overfitting is the result of a model that is trained on a dataset to such a degree that the model will only work on the training data and is not able to predict future observations. To create the final model, ten fold cross validation was performed prior, the measurements of those models resulted in a confidence interval for the accuracy. Those intervals are plotted in figure 63. As can be seen in the plot, the range of the confidence interval is narrow and consistent, this suggests that the model performed similarly on the training and the test data.

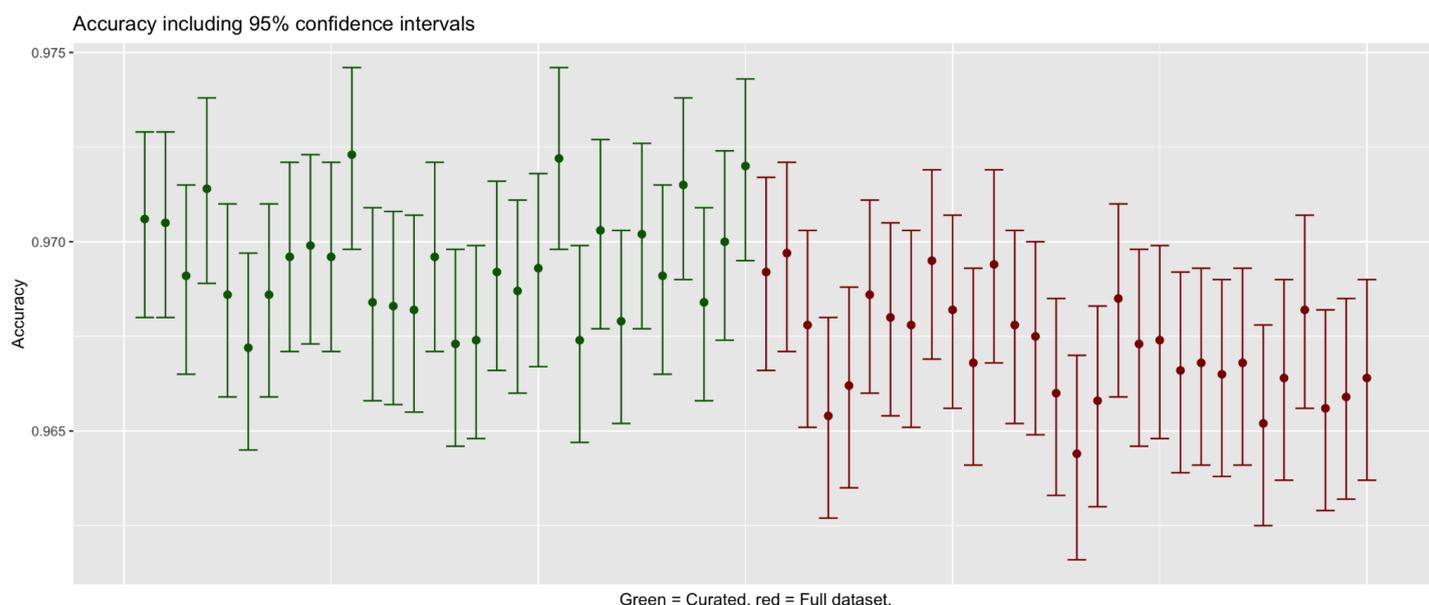


Figure 65. Accuracy scores with the 95% confidence interval based on the ten fold cross validation of 30 models trained on curated data (green) and the 30 models trained on the full dataset (red). The 95% confidence intervals give similar ranges and are of low width for all runs suggesting that there is almost no difference in accuracy based on the cross validation.

### 5.3.4 Agreement score

The class microtubule stabilisers is underrepresented in the dataset i.e. more wells are treated with other compounds. These skewed classes can result in high scores for the model due to many classes correctly not being labelled as microtubule stabilisers by chance. The Kappa score is a metric that measures the agreement between observed and predicted classes. The Kappa score also corrects for chance. Figure 54 shows Kappa scores between 0.9463 and 0.9543 for the models trained on the full dataset and 0.9505 to 0.9582 for the model trained on the curated data. A Kappa score above 0.81 and below 0.99 corresponds to an almost perfect agreement.<sup>26</sup> The high Kappa scores for all the models shows that the results are reliable even though the imbalanced presence of the target class.

### 5.3.5 Predicted classes

Thus far it is concluded that a significant increase of accuracy can be achieved, however, it must also be acknowledged that just relatively small increases in accuracy of a model might not have a meaningful impact on the overall results of the hitselection. To review the actual impact of these improvements a closer look must be given to the predicted classes by the model. Table 9 shows the occurrence of different reagent categories predicted to have a similar MOA as the training class of microtubule destabilizers in the top ten hits (highest P-value predictions) of both the 30 runs of the curated and full dataset as training data.

Based only on the MOA (table 5) the predictions based on the full data set are better. However the target for training the models was chosen to be the microtubule stabilisers that is completely represented by the compound epothilone B, and whilst it is true that the MOA of this compound is microtubule stabilising, multiple reports have found that there is a secondary effect of DNA-damage associated with epothilone B.<sup>25-27</sup> To conclude that the predictions are worse based on the primary MOA alone seems unfair. Figure 66 shows a comparison of a well of cells treated with a negative control and a well of cells treated with

epothilone B to display the effects of the DNA damage and Microtubule stabilising. The images show how the tubulin label is brighter in the epothilone B treated cells than the negative control, this is expected based on the primary MOA. The images also show the DAPI channel for the same wells; here it shows that the epothilone B treated cells have far less DAPI bound to the chromatin than the negative control, reaffirming a supposed correlation between DNA damage and treatment with epothilone B.

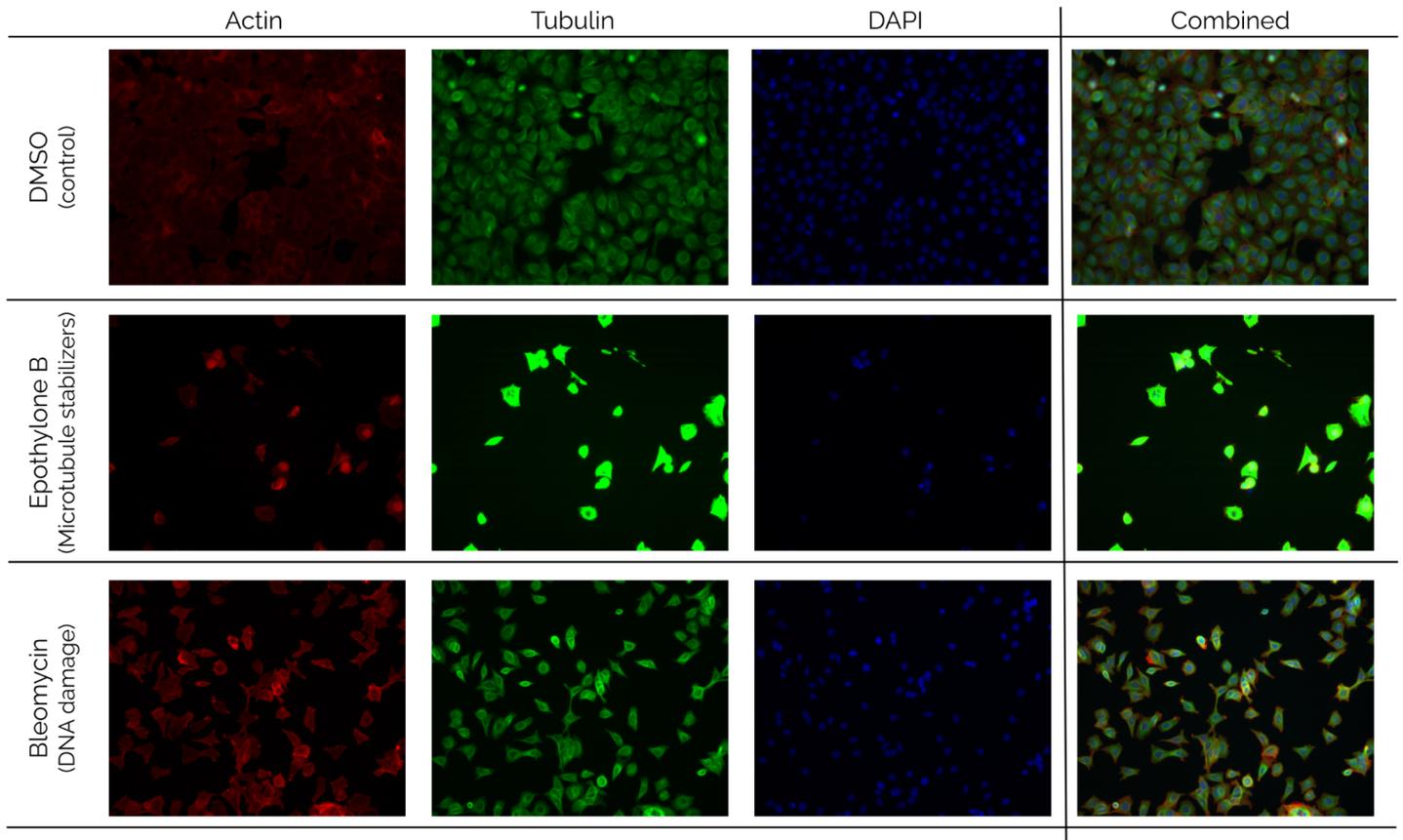


Figure 66. Comparison between Negative class well d02 plate Week7\_34341, Microtubule stabilisers well d08 plate Week1\_22381 and DNA damage well e09 plate Week6\_31641. All images scaled to negative class.

## 5.4 Conclusion

The project aim defines the research questions as following:

- MRQ: What is the value of uniting HCS image data with numeric data?
- SQ1: Does this connection aid in curating labels, eliminating extreme outliers thus increasing the quality of training data?
- SQ2: Can this connection add value to the verification and confirmation process of promising hits?

Sub question one can be answered by reviewing the accuracy scores, supported by the other given metrics, achieved by training on the curated data. Performing the one-way ANOVA statistical test resulted in a significant increase in accuracy, table 11. The connection of HCS image data and numeric data aids in curating labels, eliminating extreme outliers and has increased the quality of training data.

Sub question two is answered by the example shown in the discussion in figure 64, where reviewing predictions made by the model are explained using the images and the scaling feature of StratoVieweR showing secondary mechanisms of actions not present in the MOA list drafted by the creators of the dataset. Being enabled by StratoVieweR to review images gave more insight into the effects of the treatment and helped confirm the quality of the predictions.

Combining the answers of the sub questions answers the main research question. Uniting HCS image data with numeric data aids in the curating of labels, eliminating extreme outliers resulting in better predictive models. By evaluating the results of those models using the image data and features of StratoVieweR more insight is given into the MOA of the promising hits.

## 5.5 Future research

The observations and conclusions made only represent a fraction of the different methods and datasets available in the field of HCS. To fully grasp the scope of possibilities enabled by the unity of numeric and image data and to also find the limits of the current implementation further research is required.

Increasing the weight and statistical significance of the drawn conclusion can be achieved by performing the experiment on more datasets, preferably, datasets of differing quality so that the scale of the effect can be better measured. In these experiments it is advised to have the curation of the dataset be performed by a group of cell biologists, this will enable the results to be corrected for biases introduced during the image curation step. Introducing a third group with randomly removed images in addition to the full dataset and the curated dataset will enable better measuring of the true impact of the image curation.

The numeric data used in the experiment was generated via cell painting. The features created using cellpainting are relatively abstract but still have a semantic meaning. A new development in the domain of HCS is generating the numeric features using a convoluted neural network (CNN).<sup>28</sup> This unsupervised method for creating features is a promising method for extracting even more useful data from an image dataset. The features generated using CNN are fully non semantic i.e. the features have no meaning to humans. It can be hypothesised that the information added to the user by StratoVieweR increases in value

the more abstract the feature data is. To test this hypothesis an experiment can be conducted that reviews the results from supervised versus unsupervised feature generation with and without the connection to StratoVieweR.

Comparing StratoVieweR to Omero Iviewer has shown some features lacking from StratoVieweR. Future development of the application is advised to better the position of StratoVieweR to the competition. StratoVieweR only offers rudimentary zooming of the images, they can be either viewed regularly or full size in a popup. Switching to a more advanced image library or developing a custom solution can add this feature to StratoVieweR. Iviewer also has the thumbnail images preprocessed resulting in faster loading times for the plate on initial startup. Whilst the loading time of 13 seconds in StratoVieweR seems reasonable, improving this to a possible sub two seconds loading time by preprocessing the thumbnails will increase user engagement. StratoVieweR has been designed to work with preprocessed thumbnails and this feature will be implemented after other in development features of the StratoMineR platform are finalised. Preprocessing the images will also enable the addition of multi channel thumbnails.

In the introduction paragraph 2.3.4.1 figure 4 shows masks generated by cellprofiler. Being able to access and view these masks via StratoVieweR will enable users to review the results of their cellpainting and validate if outliers in their dataset are created due to errors during the image processing, the hyper-parameters set for the image analysis protocol or due to the images supplied.

## References

1. Omta WA. *Afstudeerplaats Aanvraagformulier Core Life Analytics Pieter V1.1.*; 01-june-2021.
2. Zanella F, Lorens JB, Link W. High content screening: seeing is believing. *Trends Biotechnol.* 2010;28(5):237-245. doi:10.1016/j.tibtech.2010.02.005
3. Hamilton SD. Microplates.jpg. Published online 28-November-2008.  
<https://en.wikipedia.org/wiki/File:Microplates.jpg>
4. Mark-Anthony Bray, Ph.D., Anne Carpenter. *Advanced Assay Development Guidelines for Image-Based High Content Screening and Analysis.*; 08-juli-2017.  
<https://www.ncbi.nlm.nih.gov/books/NBK126174/>
5. Omta WA. *Knowledge Discovery in High Content Screening.* Doctoral dissertation, Utrecht University; 2020.
6. Burnett N, ed. *Drug Discovery: Innovations in the 21st Century.* Foster Academics; 2019.
7. Bray MA, Singh S, Han H, et al. Cell Painting, a high-content image-based assay for morphological profiling using multiplexed fluorescent dyes. *Nat Protoc.* 2016;11(9):1757-1774. doi:10.1038/nprot.2016.105
8. Omta WA, van Heesbeen RG, Pagliero RJ, et al. HC StratoMineR: A web-based tool for the rapid analysis of high-content datasets. *Assay Drug Dev Technol.* 2016;14(8):439-452. doi:10.1089/adt.2016.726
9. Caie PD, Walls RE, Ingleston-Orme A, et al. High-content phenotypic profiling of drug response signatures across distinct cancer cells. *Mol Cancer Ther.* 2010;9(6):1913-1926. doi:10.1158/1535-7163.MCT-09-1148
10. Williams E, Moore J, Li SW, et al. The Image Data Resource: A bioimage data integration and publication platform. *Nat Methods.* 2017;14(8):775-781. doi:10.1038/nmeth.4326
11. Amazon.com. Accessed 23-September-2021.  
[https://aws.amazon.com/s3/?hp=tile&so-exp=below&ct=fs&refid=ps\\_a134p000003yhxaai&trkcampaign=acq\\_paid\\_search\\_brand](https://aws.amazon.com/s3/?hp=tile&so-exp=below&ct=fs&refid=ps_a134p000003yhxaai&trkcampaign=acq_paid_search_brand)
12. Winston Chang, Joe Cheng, JJ Allaire, Carson Sievert, Barret Schloerke, Yihui Xie, Jeff Allen, Jonathan McPherson, Alan Dipert and Barbara Borges. *Shiny: Web Application Framework for R. R Package Version 1.7.0.*; 2021. <https://CRAN.R-project.org/package=shiny>

13. Barthelme S. Image Processing Library Based on "CImg" [R package imager version 0.42.13]. Published online 2022.
14. Package "Microbenchmark."; 2021.  
<https://cran.r-project.org/web/packages/microbenchmark/microbenchmark.pdf>
15. Lightning Fast Serialization of Data Frames. <https://www.fstpackage.org/>
16. Thomas L. Stratified sampling. Scribbr. Published September 18, 2020.  
<https://www.scribbr.com/methodology/stratified-sampling/>
17. Ando DM, McLean CY, Berndl M. Improving phenotypic measurements in high-content imaging screens. *bioRxiv*. Published online 2017. doi:10.1101/161422
18. Ljosa V, Caie PD, Ter Horst R, et al. Comparison of methods for image-based profiling of cellular morphological responses to small-molecule treatment. *J Biomol Screen*. 2013;18(10):1321-1329. doi:10.1177/1087057113503553
19. Pawlowski N, Caicedo JC, Singh S, Carpenter AE, Storkey A. Automating morphological profiling with generic deep convolutional networks. *bioRxiv*. Published online 2016. doi:10.1101/085118
20. Singh S, Bray MA, Jones TR, Carpenter AE. Pipeline for illumination correction of images for high-throughput microscopy: ILLUMINATION CORRECTION FOR HIGH-THROUGHPUT IMAGES. *J Microsc*. 2014;256(3):231-236. doi:10.1111/jmi.12178
21. Kraus OZ, Ba JL, Frey BJ. Classifying and segmenting microscopy images with deep multiple instance learning. *Bioinformatics*. 2016;32(12):i52-i59. doi:10.1093/bioinformatics/btw252
22. Perlman ZE, Slack MD, Feng Y, Mitchison TJ, Wu LF, Altschuler SJ. Multidimensional drug profiling by automated microscopy. *Science*. 2004;306(5699):1194-1198. doi:10.1126/science.1100709
23. McHugh ML. Interrater reliability: the kappa statistic. *Biochem Med (Zagreb)*. 2012;22(3):276-282. doi:10.11613/bm.2012.031
24. Rohban MH, Fuller AM, Tan C, et al. Virtual screening for small molecule pathway regulators by image profile matching. *bioRxiv*. Published online 2021. doi:10.1101/2021.07.29.454377
25. Rogalska A, Marczak A. Nuclear DNA damage and repair in normal ovarian cells caused by Etoposilone B. *Asian Pac J Cancer Prev*. 2015;16(15):6535-6539. doi:10.7314/apjcp.2015.16.15.6535

26. Poruchynsky MS, Komlodi-Pasztor E, Trostel S, et al. Microtubule-targeting agents augment the toxicity of DNA-damaging agents by disrupting intracellular trafficking of DNA repair proteins. *Proc Natl Acad Sci U S A*. 2015;112(5):1571-1576. doi:10.1073/pnas.1416418112
27. Chen JG, Yang CPH, Cammer M, Horwitz SB. Gene expression and mitotic exit induced by microtubule-stabilizing drugs. *Cancer Res*. 2003;63(22):7891-7899.
28. Steigele S, Siegismund D, Fassler M, et al. Deep learning-based HCS image analysis for the enterprise. *SLAS Discov*. 2020;25(7):812-821. doi:10.1177/2472555220918837