Big Data Analysis of Bacterial Inhibitors in Parallelized Cellomics

- A Machine Learning Approach

by

Robert Trevino

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved July 2016 by the
Graduate Supervisory Committee:

Huan Liu, Chair
Thomas Lamkin
Jingrui He
Joohyung Lee

ARIZONA STATE UNIVERSITY

December 2016

ABSTRACT

Identifying chemical compounds that inhibit bacterial infection has recently gained a considerable amount of attention given the increased number of highly resistant bacteria and the serious health threat it poses around the world. With the development of automated microscopy and image analysis systems, the process of identifying novel therapeutic drugs can generate an immense amount of data - easily reaching terabytes worth of information. Despite increasing the vast amount of data that is currently generated, traditional analytical methods have not increased the overall success rate of identifying active chemical compounds that eventually become novel therapeutic drugs. Moreover, multispectral imaging has become ubiquitous in drug discovery due to its ability to provide valuable information on cellular and sub-cellular processes using florescent reagents. These reagents are often costly and toxic to cells over an extended period of time causing limitations in experimental design. Thus, there is a significant need to develop a more efficient process of identifying active chemical compounds.

This dissertation introduces novel machine learning methods based on parallelized cellomics to analyze interactions between cells, bacteria, and chemical compounds while reducing the use of fluorescent reagents. Machine learning analysis using image-based high-content screening (HCS) data is compartmentalized into three primary components: (1) *Image Analytics*, (2) *Phenotypic Analytics*, and (3) *Compound Analytics*. A novel software analytics tool called the Insights project is also introduced. The Insights project fully incorporates distributed processing, high performance computing, and database management that can rapidly and effectively utilize and store massive amounts of data generated using HCS biological assessments (bioassays). It is ideally suited for parallelized cellomics in high dimensional space.

Results demonstrate that a parallelized cellomics approach increases the quality of a bioassay while vastly decreasing the need for control data. The reduction in control data leads to less fluorescent reagent consumption. Furthermore, a novel proposed method that

uses single-cell data points is proven to identify known active chemical compounds with a high degree of accuracy, despite traditional quality control measurements indicating the bioassay to be of poor quality. This, ultimately, decreases the time and resources needed in optimizing bioassays while still accurately identifying active compounds.

DEDICATION

This is dedicated to Yvette and Roberto Trevino. There is not a single day where your loss isn't felt. It is my sincerest hope that this work continues to honor the legacy of your lives.

ACKNOWLEDGMENTS

Learning to learn is, by far, one of the most difficult tasks one can undertake. The road to accomplishing such an important goal has required time, patience, and persistence. It has been a long journey filled with joy, disappointment, sleepless nights, and endless days. Though it may sound impressive to proclaim accomplishing this feat through my own self-reliance and hard work - that is far from the truth. There have been too many people in my life that deserve much more than an acknowledgement. Although I'll never truly be able to repay those who have had such a profound impact in my life, I hope that this acknowledgment will serve as a small token of appreciation.

I'd like to begin by acknowledging my brothers-in-arms that I had the distinct privilege and pleasure of serving with in the Air Force. It is no exaggeration to state that some of America's finest men and women have proudly worn the uniform. If it wasn't for the mentoring of my dear friends Norm Kaiser and Michael Barker, I would not have had the opportunity of growing as an individual and airman. I would also like to acknowledge the SMART scholars program, who have provided the funding to pursue my degree while allowing me the opportunity to continue to work with the Air Force.

I would also like to acknowledge Dr. Huan Liu, who has mentored me with patience and wisdom that I could only hope to obtain one day. He has believed in me, often, when I didn't believe in myself and has provided invaluable guidance on my research. He has supported my free spirit by providing the freedom to pursue the research that I found interesting while simultaneously pushing me to be a better scientist.

I would like to thank Dr. Thomas Lamkin, who provided the motivation for this important research. His support has motivated me to continue to strive to be the best research scientist I can be. He is a true visionary with ideas that will undoubtedly leave an indelible mark in society. I would also like to thank the research group that I collaborated with extensively and made this dissertation and the Insights project possible: Steve Kawamoto, Kevin Schoen, Rhonda Vickery, Jack Harris, Ross Smith, Eric Bardes, Scott Tabar, Heather

Pangburn, and Bruce Aronow.

I'd like to acknowledge my family, who have supported me even in the most difficult of days. My parents, Silvia and Roberto, and my siblings, Elizabeth, Sylvia, Yvette, and Jovani, have all played a monumental role in the person that I am and the person I strive to become each and everyday. The bond that is shared between us can never be broken, even in death, and is something that I cherish each and everyday of my life. I'd also like to acknowledge my nephews Alfred, Lawrence, and Daniel, who have assumed the mantle of defenders of this great nation and continue to make me proud of the men they have become.

Finally, I'd like to thank God for all the blessing that have been bestowed upon me. Often times, I've stumbled and faltered, but never once have I felt alone.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Chapter 1

INTRODUCTION

With the rise of antibiotic resistant bacteria, identifying novel drugs that can inhibit bacterial infections has become a top priority in the medical field. The World Health Organization recently released a global report that stated: "A post-antibiotic era–in which common infections and minor injuries can kill–far from being an apocalyptic fantasy, is instead a very real possibility for the 21st century" [97]. Moreover, many different strains of deadly bacteria can be easily weaponized posing a serious global threat given the rise of asymmetric warfare around the world [95]. Unfortunately, the pharmaceutical industry average for the success rate of novel antibiotic drugs is $\approx 4.5\%$ using traditional analysis methods [15]. As of 2013, the capital cost expenditure for a single New Molecular Entity (NME) to become available is approximately 2.558 billion dollars over an eleven year period [28].

Figure 1.1 provides a high level description of the process for identifying novel therapeutic drugs. According to the FDA's Center for Drug Evaluation and Research (CDER), the new drug development and review process has four major stages, post identifying hits in the drug discovery stage, which include pre-clinical research, patient-based clinical trials, new drug application, and FDA review. The initial drug discovery stage is, therefore, vital for successful identification of novel drugs in an efficient manner. The drug discovery process has incorporated automated high throughput screening (HTS), which allows for hundreds of thousands of chemical compounds to be interrogated in a short period of time using optimized biological assessments (bioassays) for efficacy. Though increasing the number of chemical compounds that can be analyzed relatively quickly using automated HTS for efficacy, the success rate remains low. In fact, as a result of the increase in the number of compounds being analyzed, many false positive compounds identified as effective, demand more resources to be expended on in depth analysis without fruitful results [89, 88, 70].

Clearly, too many false positives causes a substantial increase in the cost and time used to eliminate these compounds in subsequent stages. On the other hand, a high number of false negatives that appear to not be effective may cause the complete elimination of potential novel drugs from further screening with unknown duration as to when the discarded compound may be considered again.

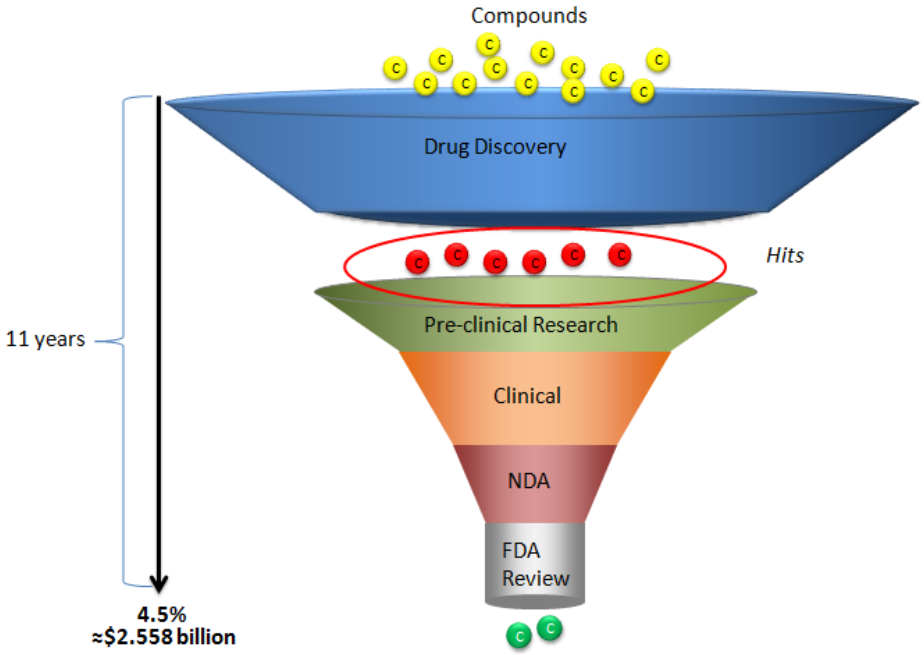


Figure 1.1: The Drug Discovery stage is the initial screening to identify the most promising potential drug candidates (Hits) for further testing. Subsequent stages provide more in depth testing to ensure safety and efficacy before FDA approval for use by the general public. This process takes billions of dollars over a decade time span to accomplish.

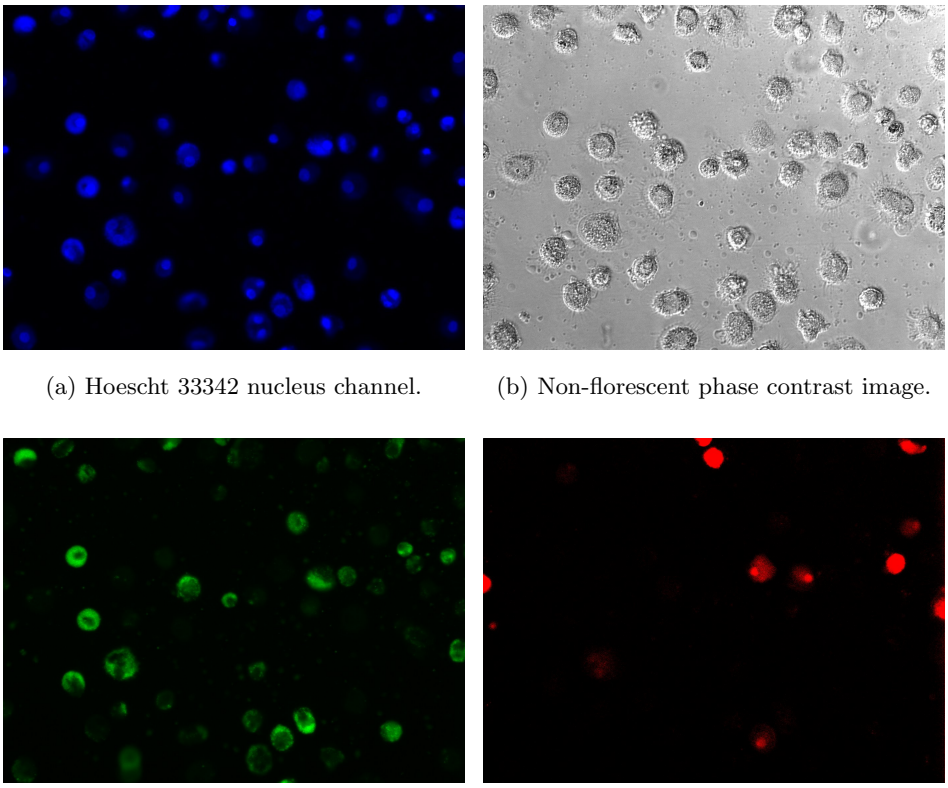
With advancements in automated microscopy, a comprehensive image-based platform called high content screening (HCS) was developed to assist in large scale HTS campaigns to better identify effective chemical compounds. Automated HCS is capable of generating a massive amount of data due to two primary factors: (1) HCS was designed for single cell analysis. The high number of cells typically produced in a bioassay increase the amount of data to be analyzed. (2) Florescent reagents have provided the ability of multispectral imaging of cells. In fact, multispectral imaging has become a ubiquitous component of HCS data that produces an immense amount of biological information across cellular and sub-

cellular data domains [87]. These domains are generally defined by the different spectral wavelength created by fluorescent reagents or by different microscopy technologies such as bright-field, dark-field, phase contrast, differential interference contrast, *etc.*

Fluorescent reagents can be grouped into three primary categories of covalent labels, non-covalent labels, and florescent indicator dyes [88]. A covalent label such as the green fluorescent protein (GFP) shown in Figure 1.2c is often used to monitor a specific protein of interest or identify and track labeled microbes. This label provides crucial information on microbe-cell interaction when analyzing infection or protein-cell interaction in siRNA endeavors. A non-covalent label such as Hoescht 33342 shown in Figure 1.2a is used to stain DNA, thereby marking the nucleus and providing valuable sub-cellular information. In addition to providing cell cycle information, nucleus staining provides critical information often used in cell segmentation. An indicator dye such as Live/Dead[1] far red, is shown in Figure 1.2d is employed to determine the viability of cells after the introduction of bacteria or other lethal treatments. A compromised cell membrane will result in the red viability dye permeating into the cell indicating cell death. Viability information is crucial in determining the safety and efficacy in potential drug candidates.

In contrast to fluorescent reagents, certain microscopy technologies utilize the visible light spectrum in conjunction with a magnifying lens to better visualize objects that are too small to be viewed by the human eye. Bright-field microscopy is the most elementary of the different technologies; it simply passes white light through a magnifying lens and object capturing the amplitude change of the light in the final image. Although bright field microscopy has been used extensively, phase contrast microscopy has become increasingly popular given its ability to produce better contrast in cells that would otherwise be translucent using bright-field microscopy. Phase contrast microscopy not only detects amplitude changes but also phase changes of light by converting shifts in the phase to amplitude changes. The phase contrast image shown in Figure 1.2b provides a more nuanced description of a cell membranes than bright-field, otherwise, could provide.

[1]This dye is produced by Life Technologies https://www.lifetechnologies.com/us/en/home.html



(a) Hoescht 33342 nucleus channel.

(b) Non-florescent phase contrast image.

(c) GFP bacterial image.

(d) Live/Dead Far Red viability image.

Figure 1.2: Multi-wavelength imaging of infected hMDMs. hMDMs were infected with F. tularensis SCHU4 for 30 hours and then stained, fixed and imaged. (a) Nuclear image showing Hoechst 33342 fluorescence. (b) Phase Contrast whole cell image. (c) Bacterial image showing GFP expression. (d) Viability image showing dead cells stained with Live/Dead fixable viability stain.

1.1 The Era of Big Data

Automated HCS has ushered in an era of "big data" analytics in drug discovery endeavors by utilizing multispectral imaging of individual cells. This has provided a mosaic of in depth information of cell and sub-cellular processes and perturbations. Although, multispectral imaging has produced an immense amount of valuable information, it has also produced several challenges. The most obvious challenge is how to properly utilize, store, and retrieve the massive amounts of data generated. Traditional methods have become obsolete in this new age of big data analysis in drug discovery. For instance, generating a single readout dataset for analysis of activation or inhibition of a specific protein has

been replaced by generating a high dimensional dataset requiring multivariate analysis. A medium scale screen of 25,000 compounds in duplicate has been shown to generate upwards of $\approx$ 600 GB of raw image data alone, while a large scale screen of a million compounds can produce over 25 TB of raw image data [17]. Assuming a modest one hundred cells treated per compound and the control data needed to ensure quality of a bioassay, a screen is capable of producing anywhere between 2.5 billion to well over 100 billion individual data points for medium to large scale screening. In addition, each data point can be defined in a feature space ranging from a single feature to tens of thousands of features. The amount of data generated provides a daunting task of identifying useful information that could buttress compound hit selection. Interactive applications such as CellProfiler and CellClassifer have provided researchers with powerful tools to more efficiently incorporate multivariate data in phenotype analysis of cells [16, 74]. Unfortunately, despite these powerful tools, the majority of automated HCS analysis in drug discovery endeavors is still performed using single readout analysis [82]. This lag in use of multivariate data is partially due to the lack of knowledge on how to properly mine the data and implement multivariate analytics in an automated manner.

Another challenge created by automated HCS originates from the cost of using florescent reagents due to the sheer number of cells being generated for multispectral analysis. This is due to not only the cost of the reagents themselves, but also other associated operational costs such as the expertise and the specific "wet lab" environment required to administer, handle, and store the reagents. Moreover, the reagents are toxic to cells over a prolonged period of time. This, unfortunately, limits the analysis of cell populations to a brief period of time. Limiting temporal information in compound analysis does not allow for optimal vetting of cell response to chemical compounds over extended periods of time. Increasing temporal analysis may provide additional information to fully understand the perturbations caused by chemical compounds on cell phenotypes as well as cell microenvinronmet interactions.

1.2 Motivation

"Parallelized cellomics" attempts overcome the challenges previously described by integrating the latest machine learning algorithms using parallel processing in a high performance computing (HPC) environment in the following three major components of image-based HCS:

- ***Image Analytics***

  Image Analytics is concerned with the quality of the images and segmentation of individual cell sub-images within a single image. Computer vision has made significant gains in recent years where this component is certain to benefit from.

- ***Phenotypic Analytics***

  Phenotype Analytics pertains to the measurable perturbations that occur as a result of cell exposure to bacteria or chemical compounds. Traditionally, phenotype analysis has been conducted based on biological hypotheses of the type of perturbation one would expect to observe given a cell's interaction with a biological target and chemical compounds. A single readout or a hand full of measurements were then taken of these phenotypes changes for further analysis creating a "bottom-up" approach. Another more robust approach that is demonstrated is the top-down approach, which measures a significant number of different image properties and allows for feature selection algorithms to identify those that measure significant phenotype perturbations.

- ***Compound Analytics***

  Compound Analytics is the process of identifying active compounds of interest that activate or inhibit a target protein, gene, or microbe. Traditional methods used in compound hit selection have relied upon single readout activity measurements derived from cell population distribution. These measurements are generally represented using a single value such as the mean, median, quartile, *etc.* of a cell population treated with a compound. Single cell analytics, also known as cellomics, has provided the

ability to move away from univariate distribution analysis into multivariate single cell analysis. Cellomics allows for incorporating powerful machine learning algorithms that are capable of handling large quantities of data instances in high dimensional space.

These components form what is called the I^3 paradigm defined as:

- Identify individual cells.
- Identify pertinent phenotypes.
- Identify active compounds.

In this paradigm, Image-base HCS data acts as the mediator or link between the "wet lab" and computational analysis. The primary question investigated is: *To what extent parallelized cellomics can have in improving analysis in the I^3 paradigm to facilitate successful screening of active chemical compounds that inhibit bacterial infection?* In addition, investigation is focused on the impact of using more sophisticated machine learning algorithms in analyzing optical microscopy data while limiting florescent reagent utilization. The primary focus of this dissertation is, therefore, to demonstrate the extent to which machine learning can be utilized to overcome big data analytical problems and reduce dependency of florescent reagents in each component while maintaining competitive results in the I^3 paradigm.

1.3 Roadmap

A roadmap is provided that describes how the rest of this dissertation is organized. The *Image Analytics* component as it pertains to cell segmentation will be thoroughly discussed in chapter 2. Chapter 3 will cover the *Phenotypic Analytics* component and how feature selection can play an pivotal role in phenotype perturbation analysis. Chapter 4 will discuss the *Compound Analytics* component and how integration of cellomics coupled with machine learning provide a robust and powerful alternative to traditional single readout

analysis. Chapter 5 describes the "Insights" project- a powerful parallel processing pipeline designed specifically for big data analysis of image-based HCS data. Chapter 6 will discuss the trajectory of future research on such an important area from which humanity will undoubtedly benefit.

Chapter 2

IMAGE ANALYTICS

The *Image Analytics* component deals with image acquisition and subsequent segmentation of individual cells from the acquired images. The acquisition of images is outside of the scope of this dissertation. However, it goes without saying that image acquisition follows the old adage of "garbage in, garbage out." It is necessary to acquire quality images to increase the probability of successful HCS campaigns.

The identification of cells in an image, often referred to as "segmentation", plays a significant role in the quality of the results obtained in screening campaigns. For instance, Hill *et al.* demonstrated that SK-BR-3 cells that were well segmented increased their ability to resolve specific changes in perturbed cells [40]. A common obstacle to overcome in segmentation of cells is the transparency of many different cell lines, which also limits the amount of biological information that can be subsequently extracted when using bright-field microscopy. Phase contrast microscopy has provided a viable alternative to bright-field microscopy by converting phases shifts in light to visible amplitude changes. Unfortunately, in addition to random noise caused by factors such as irregular lighting and external artifacts, phase contrast is also prone to systematic errors. Certain types of noise, such as "halos" and "shade-offs", are quite prevalent in phase contrast microscopy. Figure 2.1 provides a view of this inherent noise in phase contrast images. There has been extensive research in attempting to overcome these issues to better identify individual cells for use in HCS analysis using different methods that are further described in subsequent sections. First, a problem statement is provided, followed by describing a number of different methods that attempt to segment individual cells.

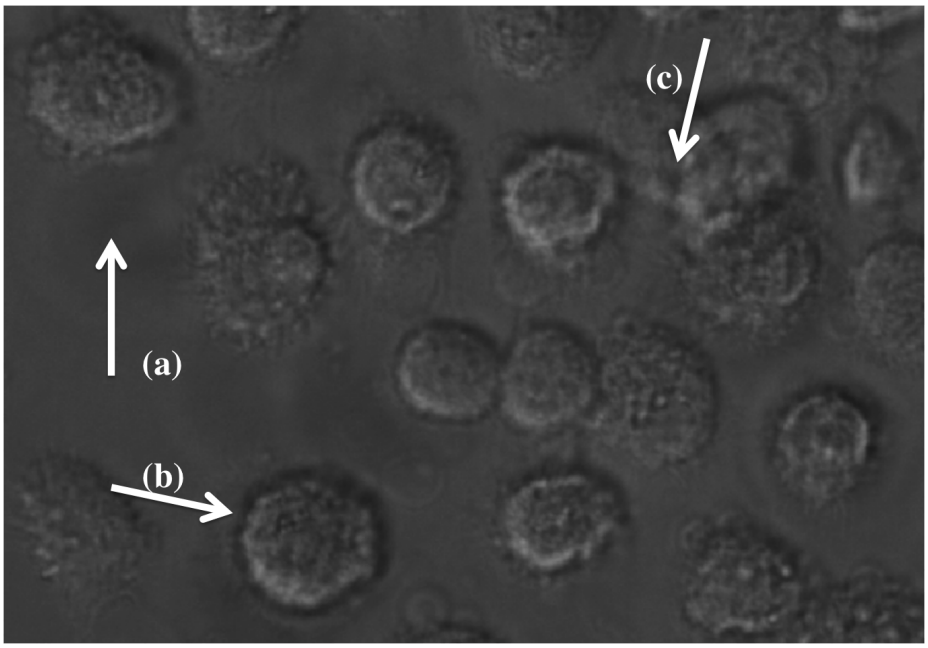


Figure 2.1: The phase contrast channel presents various issues when trying to segment individual cells. (a) Shade offs. (b) Halos. (c) Closely clustered cells.

2.1 Problem Statement

In the simplest terms, the problem statement for cell segmentation can be described at a high level as identifying the region of pixels pertaining to individual cells in an image with high accuracy. Let $\mathbf{I}_{pc}$ be a phase contrast image where (r, c) represents the row and column of each pixel. Each pixel is assigned a label $y(r, c)_{pc} = \{1, 0\}$ for cell and background region, respectively. The objective function to minimize the error of pixel classification is formally defined as:

$$\underset{\mathbf{w}_{pc}}{\text{minimize}} \quad \sum_{r=1}^{R}\sum_{c=1}^{C}(y_{pc}(r,c) - \theta(\mathbf{I}_{pc}(r,c)) * \mathbf{w}_{pc})^2, \tag{2.1}$$

where θ represents the feature space of a pixel $\mathbf{I}_{pc}(r, c)$ and $\mathbf{w}_{pc}$ represents the weights assigned to each feature. There should be a distinct boundary that allows for the clear delineation of each cell in an image. Since $\mathbf{I}_{pc}$ is a gray scale image, finding the appropriate feature space θ that can best delineate between different cells is non-trivial. This is because the edge information may often be compromised when cells cluster together or due to random and systematic noise inherent in phase contrast images. Therefore, defining θ plays

just as an important role as weighting θ using $\mathbf{w}_{pc}$ to properly segment cell and nucleus regions. SIFT [57], Haralic [36], and HoG [24] are examples of well known feature spaces often used in image segmentation. Unfortunately, often times these features spaces are not well suited for automated HCS campaigns where large number of images are generated.

There are two primary measurements that are used in determining the quality of cell segmentation: (1) enumeration and (2) pixel area overlap. Enumeration compares the true number of cells that are in an image to the number identified by a chosen segmentation method. Pixel area overlap is a measure that quantifies how well the appropriate pixel region and boundary of the cells are identified.

2.2 Cell Segmentation

Cell segmentation has been thoroughly investigated within the confines of computer vision analysis. This type of segmentation is unique from object segmentation in an image in that it is not a complex object to be segmented but rather millions of simple objects in a complex environment. Several methods are subsequently described that attempt to accomplish this task on phase contrast microscopy images specifically for use in automated HCS campaigns. Figure 2.2 demonstrates the efficacy of the different described methods for cell segmentation on a dense cluster of cells, which often occurs in automated HCS images.

2.2.1 Thresholding Methods

The first to be described is the thresholding methods, which tend to be the simplest and fastest methods. In theory, using phase contrast images, cellular regions should be identifiable through global thresholding techniques such as Otsu or Kurita *et al.*'s methods [67, 49]. These methods are extremely fast and efficient and are well suited for large-scale HCS campaigns. Unfortunately, there are two significant drawbacks using these thresholding methods. First, they are quite susceptible to the systematic noise previously described in phase contrast images which often hinder the use of such thresholding methods. Second, they are quite susceptible to lighting issues as demonstrated in Figures 2.2b and 2.2c.

In light of these drawbacks, the obvious solution would be to find a method that removes the systematic noise prior to using the thresholding methods. Previous methods have attempted this task by minimizing the presence and effects of systematic noise using a variety of distinct approaches. For instance, Yin et al. focused on a pure phase contrast image restoration by removing the "halo" and "shade-off" artifacts using phase contrast microscopy properties coupled with an iterative optimization algorithm that approximates the restored artifact-free image [101]. A simple thresholding algorithm could then be used to determine cell pixels from background pixels. Su et al. followed up with this work by proposing a phase contrast image restoration method based on the dictionary representation of diffracting patters [85]. Unfortunately, both methods are not well suited for large-scale data sets and are still susceptible to irregularities in microscope lighting using traditional global threshold methods. They also require microscopy information that is often difficult to obtain.

A method named PHANTAST by Jaccard et al. provides a much more robust and faster method of identifying cell regions in phase contrast images without requiring the removal of noise beforehand [42] . This method is quite powerful and efficient in estimating the boundary regions of cells in a phase contrast image. The method analyzes the local contrast within a predefined window. The window is a soft-edge Gaussian kernel with standard deviation of σ also defined by the user. Formally, the local contrast can be defined as

$$C = \frac{\sqrt{(w * I^2 - (w * I)^2}}{(w * I)}, \tag{2.2}$$

where I is the image of interest, w is the Guassian kernel window and $*$ is the convolution operator. This computes the local contrast which is defined as the standard deviation in image I within window w divided the mean within the same window. The center pixel of window w stores the local contrast value for window w. A binary image G is then derived using a global local contrast threshold ϵ applied to matrix C.

$$G(x,y) = \begin{cases} 1 \text{ If } \text{C(x,y)} > \epsilon \\ 0 \text{ If } \text{C(x,y)} \leq \epsilon \end{cases} \tag{2.3}$$

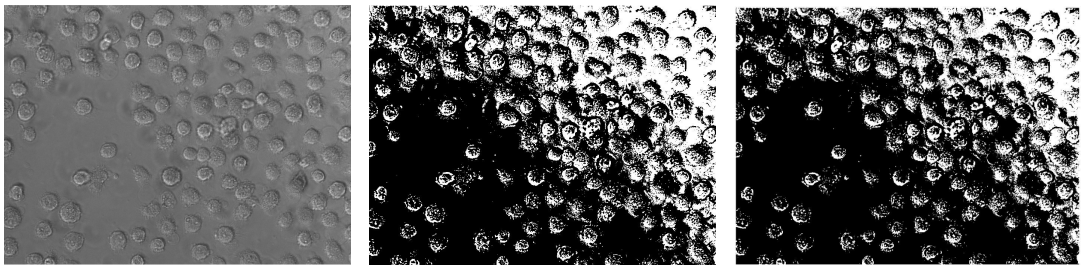
The binary image G provides a reliable approximation of cell pixels since these pixel are located in regions of high intensity variation. The ϵ value is set at 0.03 while the σ parameter is set to 1.4 in the original paper and yielding results in Figure 2.2e.

2.2.2 Superpixel Methods

Unfortunately, none of the thresholding methods discussed using phase contrast restoration or not perform well on images containing high density cell clusters. In an effort to overcome segmentation of clustered cells, superpixel methods were designed. Superpixels are contiguous pixel regions that pertain to the foreground or background of objects of interest. They generally detect boundary regions but also over segment objects requiring that they be incorporated with some merging algorithm. The Simple Linear Iterative Clustering (SLIC) algorithm is a well known efficient and fast superpixel algorithm that has demonstrated the ability to identify superpixels at comparable and in some cases better results than most superpixel algorithms that currently exist [1]. The effectiveness of the use of superpixels in the segmentation of images was demonstrated by [46] using SLIC in conjunction with DBScan. Unfortunately, this method did not perform well placing boundaries within clearly distinguishable cells in phase contrast images. This is due to the amount of noise inherent and actual variance within the image and cell regions.

2.2.3 Watershed Variant Methods

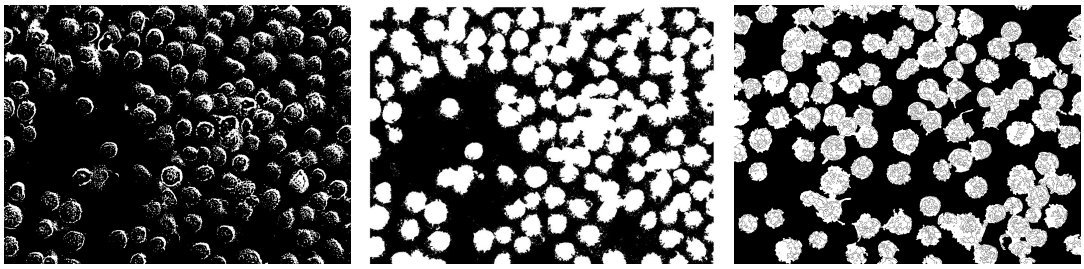
The watershed variant methods are extremely powerful and are well known in computer vision tasks [93, 61]. The name is derived from the way the methods intuitively behave like a catchment basin. In order to mitigate well known over segmentation issue, watershed variant methods that used either local minima or predefined markers to initially grow a region were developed. The most reliable methods of segmenting individual cells rely on nucleus staining



(a) Original Phase Contrast Image.

(b) Otsu thresholding

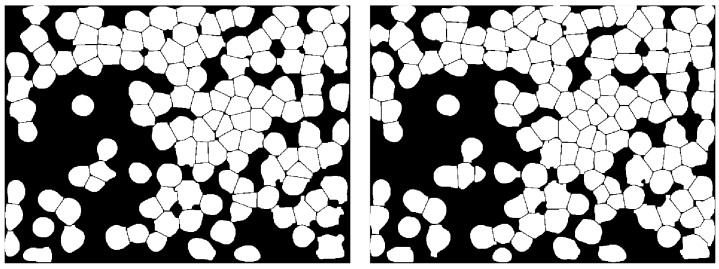
(c) Kurita thresholding



(d) Artifact removal algorithm by Yin et al.

(e) PHANTAST.

(f) SLIC with DB Scan.



(g) Proposed nucleus-based cell segmentation.

(h) Watershed variant with fluorescent staining.

Figure 2.2: Different thresholding methods used as baselines in this work.

to give initial markers that can be subsequently used in watershed variant cell segmentation methods [4, 64, 10]. This has caused fluorescent reagents to be directly relied upon for cell segmentation. This reliance requires the additional step and corresponding resources to ensure that all cells have their respective nuclei stained with florescent reagents in a bioassay. This becomes a burdensome but necessary step when huge campaigns of tens of millions of cells are conducted.

2.3 Proposed Nucleus-based Cell Segmentation

The watershed variant methods that relied on a seed point used fluorescent reagents to identify the nucleus region of the cell and provide that seed point. As such, research was conducted to determine whether the nucleus region of a cell could be identified in a phase contrast image without using fluorescent reagents. Accomplishing this task would reduce cost and complexity of identifying seed points while sustaining the rapid, accurate segment of cells using watershed variant methods. Two novel concepts called nucleus protrusion and spatial variance are introduced as key measurements required for automated nucleus segmentation to occur.

Nucleus protrusion is the extent to which the nucleus protrudes from the cell causing a natural or induced visual cue for detection. The more nucleus protrusion that exists, the more visible the nucleus becomes. Figure 2.3 demonstrates a range of nucleus protrusion in synthetic data at 10% intervals in an 8-bit grayscale image with values ranging from 0 to 255. At 10%, the nucleus is completely invisible to the human eye. This provides very little visual cues for any learning algorithm to exploit. At 100%, the nucleus is clearly protruding from the cell causing a clear visual cue that any well trained learning algorithm could utilize. Using florescent reagents artificially increases the nucleus protrusion to a much higher range than other wise would be achieved naturally while minimizing noise from non-nucleus pixels. This allows for rapid and highly accurate segmentation of nuclei using simple thresholding methods.

Spatial variance is the extent to which the nucleus will arbitrarily be located within the confines of a cell. The nucleus is generally located near the center of the cell. However, this can change depending on a number of factors such as clustering or the health of a cell. Thus, four primary regions are defined within a cell, where each larger region encompasses the previous smaller one as shown in 2.4. If a nucleus is consistently located in a single location, then the dataset is considered to have no spatial variance. If, however, the nucleus varies extensively in the different regions, then a high spatial variance is given to the dataset.

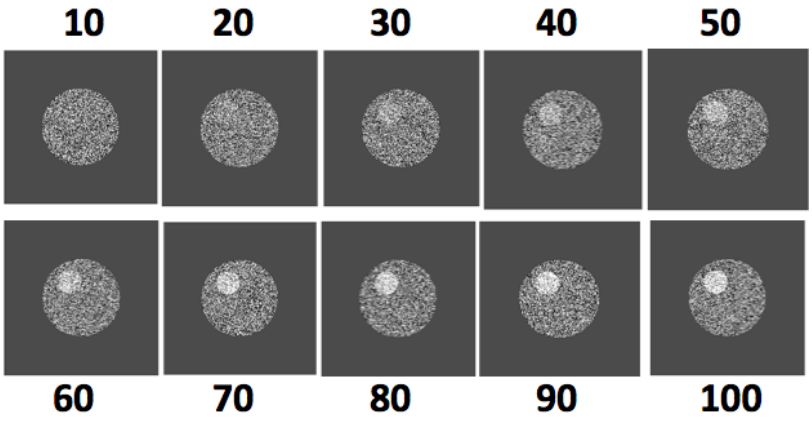


Figure 2.3: The nucleus protrusion scale ranges from no visibility to complete visibility.

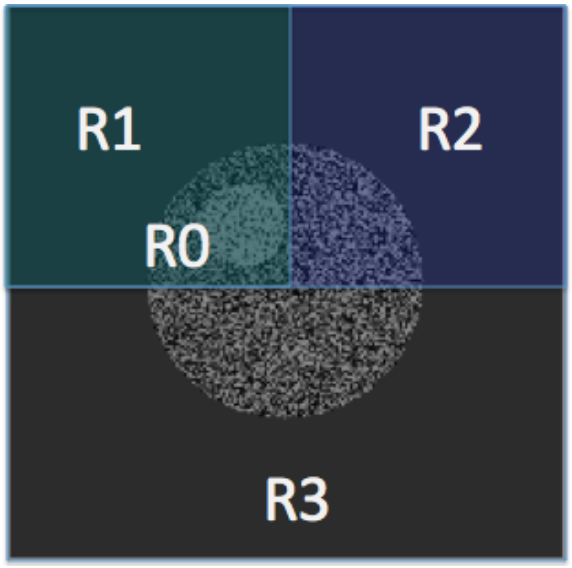


Figure 2.4: The local variance demonstrates the region of a cell the nucleus can be located in. The regions overlap each other with $R0$ representing a stationary point in the cell.

Ideally one would like a dataset with either no spatial variance or high nucleus protrusion. As previously described, the more robust and accurate cell segmentation methods rely on nucleus staining using fluorescent reagents to yield data that enumerates the number cells in an image as well providing a seed point for cell segmentation. This is why florescent staining data is considered the gold standard in image-based HCS analysis for obtaining nucleus information. Unfortunately, the process is massively disruptive to biological processes eventually leading to cell death. In addition, the staining processes affect environmental and phenotype characteristics such as spatial distribution and morphology [42]. As a result,

although highly accurate cell segmentation is achieved, the fluorescent staining limits an assay to an incomplete and often flawed "snapshot" of important biological processes of a cell and the intricate microenvironment. Therefore, a nucleus segmentation with minimal use of florescent reagents would be ideal to achieve accurate cell segmentation, provide DNA information, and ensure minimal disruption of biological processes.

Unfortunately, none of the algorithms previously described for cell segmentation are capable of identifying the nucleus region of cells and, furthermore, make no attempt to do so. There are only a few methods that have attempted to identify the nucleus regions of cells in phase contrast channels. Kazmar *et al.* proposed one of those few works that has used phase contrast microscopy to identify the nucleus region of cell without the use of florescent reagents or staining [44]. Dewan *et al.* demonstrated the feasibility of segmenting nuclei in phase contrast images using features based on intensity, convexity, and texture [27]. They utilized the more popular feature spaces for use in gray scale images defined by the Haarlick features [36]. These features are derived by the use of a gray level cooccurence matrix (GLCM) and measure different textures of the image. A convolutional neural network architecture was also implemented by Song *et al.* to segment stained nuclei in cervical cancer cells [83]. However, their nuclei had high visual cue due to the staining process.

These methods, unfortunately, fail unless the nuclei have a nucleus protrusion value and are clearly visible to the human eye. As previously stated, visibility of the nuclei in a phase contrast image is not always certain and is a function of the cell, the microscopy technology, and the magnification being employed. Therefore, investigation is done into whether spatial variance is low enough that it can supplement visual cues.

2.3.1 Proposed Methods

A multi-layered convolutional neural network (CNN) was implemented to utilize visual cues and spatial information in phase contrast images to identify the nucleus region of a cell as shown in Figure 2.5. Convolutional neural networks (CNN) are currently at the forefront of computer vision tasks. They have demonstrated the capacity to outperform state-of-

the-art methods remarkably well in image segmentation and classification for a variety of datasets [47, 22, 21]. Biological image analytics has especially benefited from advancements in implementation of convolutional neural networks. Ciresan et al. demonstrated the superior capabilities of CNN in biological image analytics when segmenting neuronal structures [19]. In addition, Ciresan et al. utilized deep CNNs to identify mitosis in cervical breast cancer cells [20]. Recently, Hou et al. have demonstrated the utility of CNNs in classifying three of the most common sub-types of Low-Grade Glioma (LGG) using multi-gigapixel images [41].

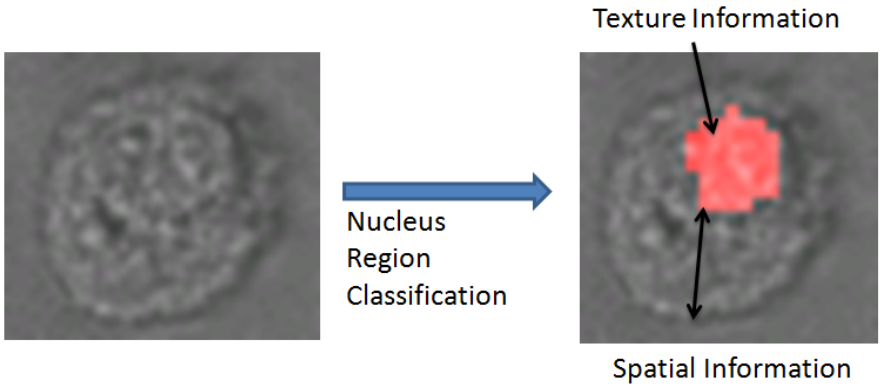


Figure 2.5: The spatial and texture information is researched to determine whether it can be used to identify the nucleus in a cell with low visual cues.

A convolutional neural network is a hierarchical neural network architecture consisting of a series of convolutional layers alternating with sub-sampling layers. This is followed by a number of fully connected layers that are subsequently passed into a classification layer. Convolutional layers use filters or two dimensional kernels that are convolved with the previous layer's output to extract feature mappings containing pertinent image information. Convolutional layers contain three parameters: 1) size of kernel K, 2) Number of kernels N, and 3) size of strides S used for convolution. These parameters determine the number and the size of its output feature mapping. The larger the number of convolutional layers, the more complex the CNN architecture becomes. The proposed CNN architecture shown in Figure 2.6 had the following parameters, which will be described in more detail: Layer 0-$K = 6$, $N = 50$, $S = 1$. Layer 1- $K = 4$, $N = 50$, $S = 1$. Layer 2-$K = 4$, $N = 100$, $S = 1$.

The CNN architecture consisted of three convolutional layers with 50 feature maps in the first two layers and 100 feature maps in the final layer. The feature maps are designed to look for visual cues that discriminate between nucleus pixels and non-nucleus pixels. In addition, the spatial information is utilized to determine whether feature maps can be produced to substitute for visual cues.

Sub-sampling is done to minimize the data as it traverses through the CNN while maintaining pertinent information. Previous research has demonstrated improved performance using max pooling versus other sub-sampling methods such as averaging [75, 76]. Max pooling splits feature mapping into non overlapping regions where the max value is used to represent a given region. Max pooling size was set to 2 which means that each layer a 2x2 section was converted to the max value of that section.

The CNN architecture provided a single hidden layer with 250 fully connected neurons. In order to mitigate over-fitting of the training data, the dropout method was implemented to create thinned neural networks [84]. Srivasta *et al.* demonstrated that the dropout method reduced over-fitting and provided lower training error rates. In addition, rectified linear unit $ReLU$ were implemented as activation functions as they have demonstrated the ability to improve deep learning performance in object recognition [63, 43].

Logistic regression with soft max was used as the classification layer. Elastic net was also implemented into the weights of the neurons to mitigate over-fitting. The L1-norm and L2-squared norm were assigned parameter values of 0.01. These values were obtained from the recommendations of Theano deep learning tutorial [8, 9].

Initial feature maps were generated from two primary images containing spatial and surface information. Figure 2.7 shows the two images are the distance transform and the original image with contrast enhancement using adaptive histogram equalization of the approximated cell region. The distance transform image provides spatial information to detect location patterns and consistency of those patterns of the nuclei in a cell. The original image with contrast enhancement using adaptive histogram equalization provides

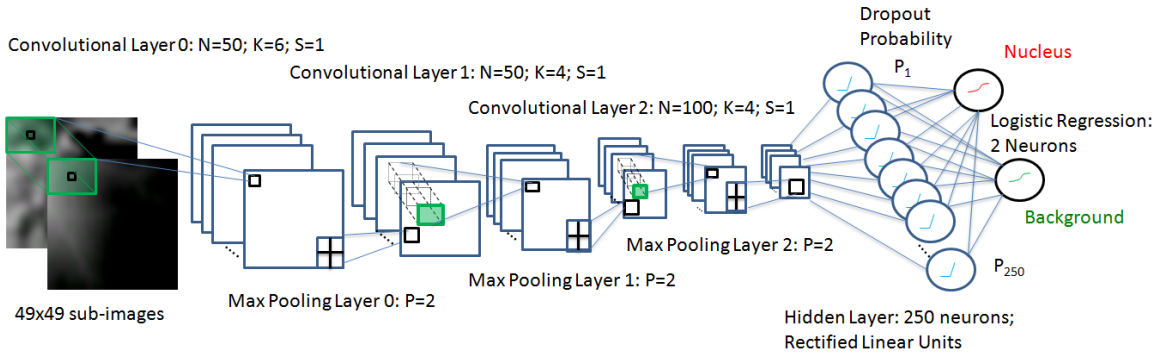
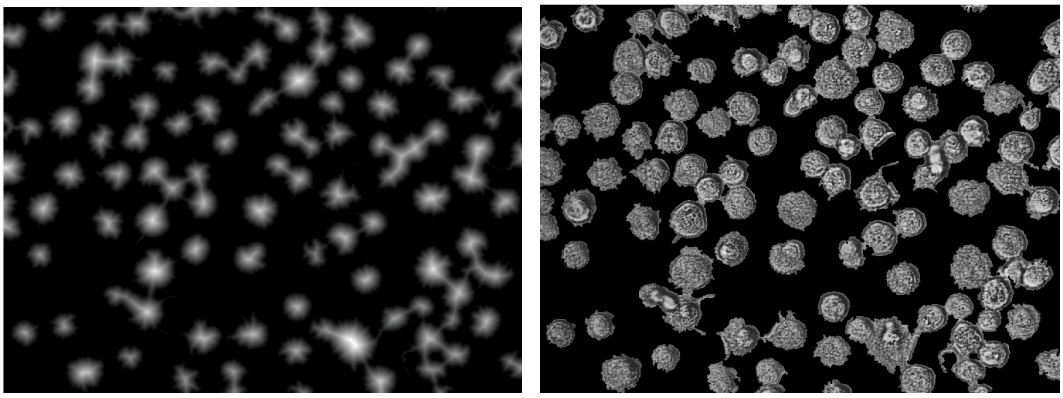


Figure 2.6: The convolutional neural network architecture implemented for pixel classification of nuclei regions uses both spatial and texture information.



(a) Distance transform of cell region provides useful spatial information.

(b) Contrast enhanced imaging provides more nuanced information of boundaries.

Figure 2.7: Information used in nuclei analysis is displayed in the following two images: (A) Distance transformation provides a spatial dimension to nucleus analysis. (B) Contrast enhancement is done to assist in identifying texture changes between cell and nuclei regions.

enhanced texture information to capture nuanced changes in the membrane. Adaptive histogram equalization effectively redistributes image intensity at the local level allowing for more crisp phase contrast image. Since phase contrast images are quite noisy, adaptive histogram equalization is much more effective since it operates on local regions of an image.

Approximated cell regions were used to reduce the number of pixels to be analyzed. The images that were generated consisted of 1040 x 1392 pixels each. A single phase contrast image, therefore, contains 1,447,680 pixels, which leads to over 10 million pixels to be analyzed in a bioassay plate for each well sectioned into 8 sites. However, the majority these pixels are not nuclei pixels. Since a nucleus pixel must necessarily be a cell pixel,

limiting the search space to cell pixels only will reduce the number of pixels to analyze. Song *et al.* demonstrated the utility in reduction of pixels space in preprocessing stages of CNNs used to identify nucleus regions of cells [83]. The method was reliant upon staining to identify cell regions. The method proposed here relies on the unique characteristics of phase contrast microscopy to reduce the pixel space and create approximated cell regions. In order to identify cell pixels, the PHANTAST method previously described was used [42]. This method tends to overestimate cell regions of clustered cells and seldom underestimates cell regions in phase contrast images making it ideal for use in approximation. Hole-filling and artifact removal are needed due to the noise inherent within phase contrast images and limit the false negative and false positive errors, respectively. In order to minimize false positive cell region estimation, artifact removal size is set to 900. It is assumed conservatively that the cell regions will be no less than a 30x30 pixel area. During preprocessing, this removes any items in the original image that contain high variance but are, in fact, too small to accurately reflect a cell. During post-processing, PHANTAST checks the foreground labeled pixels in the image and removes any items in the binary image that have a total area less than 900 in order to remove any anomalies not found in the preprocessing stage of PHANTAST.

Hole filling is used to reduce the false negative cell region estimation. The hole filling size was set to 128 and is derived from the estimation of the space between three perfectly circular cells touching each other.

$$HF = \frac{1}{2} r(2\sqrt{3} - \pi r), \tag{2.4}$$

where r is the radius of each cell. The cell radii is used to create a triangular window, where the base of the triangle is the diameter of the bottom-most cell, and the tip of the triangle stops at the midpoint of the top most cell. Therefore, the triangle's height is found using Pythagorean theorem. Assuming a cell size of 50 pixels, if the area is smaller than 128, it is filled in and considered to be part of the cell region.

Since the original image is 1040x1392, the approximated cell region images are a cascaded

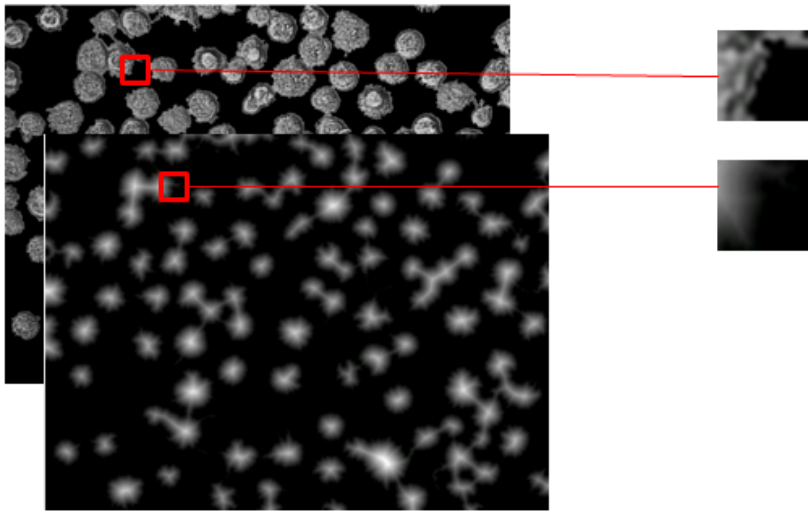


Figure 2.8: The preprocessing stage provides two initial feature maps that are split into sub-images for each pixel.

set with dimensions 1040x1392x2. Training pixels are extracted from the converted set into an $SxSx2$ sub-images where S is the sub-image size determined to best capture nucleus information. As previously notes, the parameter S was set to 49 to yield a sub-image that is larger than a nucleus region but smaller than a cell region. These sub-images is used in training the proposed convolutional neural network to determine nucleus regions.

Post-processing was done on the results of the CNN classifier using a constrained exact cover variant approach. The ground elements $P = \{p_1, p_2, ...p_n\}$ represent the pixels classified as nuclei. The subsets $S_1, S_2, ...S_k \subseteq P$ are defined by all nuclei that fall within a predefined window W of size 35. The window size was selected to encompass the average nucleus size. Since all pixels are given the same weight, an integer linear programming can be formally defined as:

$$\underset{x}{\text{minimize}} \sum_{i=1}^{k} x_i$$

subject to

$$\sum_{i:p \in S_i} x_i \geq 1, \ \forall p \in P$$

$$x_i \leq 1, \ \forall i \in \{1, ..., k\}$$

$$x_i \in \mathbb{N}, \ \forall i \in \{1, ..., k\}$$

$$S_i \cap S_j = \{\}, \ \forall (x_i, x_j = 1)$$

The constraint for all pixels to be selected is removed and replaced with most pixels to be selected. Additional constraints were placed on the selected subsets to ensure that the subsets were within a cell and the distance between the center C of two subsets was greater than some predefined value τ. The τ was set to 45 to reflect the distance between two nuclei centers. The ILP can now be formulated as:

$$\underset{x}{\text{maximize}} \sum_{i=1}^{k} x_i * |S_i|$$

subject to

$$x_i \leq 1, \ \forall i \in \{1, ..., k\}$$

$$x_i \in \mathbb{N}, \ \forall i \in \{1, ..., k\}$$

$$S_i \cap S_j = \{\}, \ \forall (x_i, x_j = 1)$$

$$C_i - C_j \geq \tau, \ \forall (x_i, x_j = 1).$$

An approximated simple greedy solution is to rank all subsets $S \subseteq P$ in descending order by the number of pixels that they cover and then select the top subsets with no overlapping pixels where the center is at least τ units away.

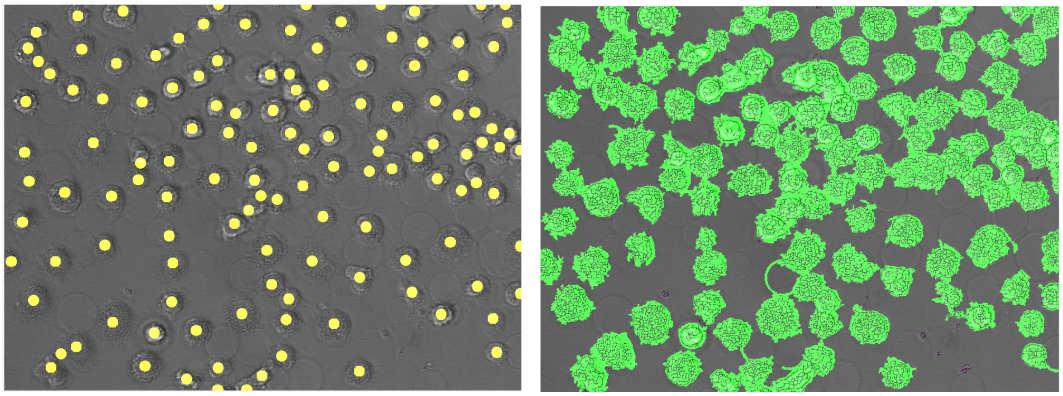
2.4 Results

In order to determine the effectiveness of the proposed methods, they were compared to known nucleus segmentation algorithms previously discussed. Approximately 140,000 pixels were used from different densely clustered cell sub-images. Since the number of nuclei can become extremely large dependent on the data set, we obtain training and testing labeled data using Adiga et al.'s proposed method for generating nucleus masks with florescent reagents [3]. The use of florescent reagents to determine the nucleus regions assures the training and testing data are an accurate estimation of where the nucleus region of a cell is located. The CNN demonstrated the ability to classify nucleus pixels with high accuracy and a relatively small number of training data.

As demonstrated in Figure 2.9, the results indicate that the use of spatial and texture information allows comparable performance to the nucleus staining method and better performance than the phase contrast method. In addition, subsequent cell segmentation using the identified nucleus regions as seed points produces comparable results to those generated by fluorescent reagents as shown in Figure 2.2g.

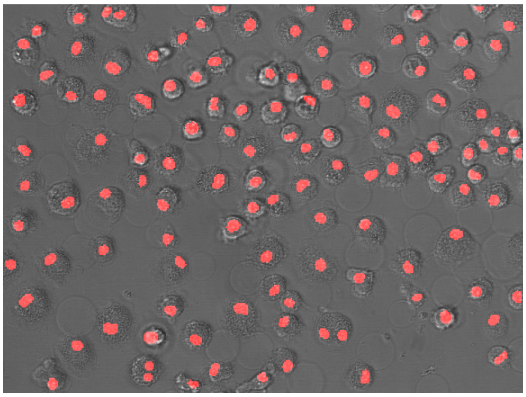
To what extent the spatial information improved the CNN model was also investigated. First, a comparison was done using CNNs trained with contrast-enhanced images versus those trained with contrast-enhanced images as well as spatial information. Table 2.1 demonstrates the significant impact that spatial information has when using spatial information across over forty different images in 6 different wells. Since the majority of pixels generally tend to be background pixels that neither pertain to the cell or nucleus region, the statistical analysis can be misleading providing overly accurate results. Thus, analysis is limited to the more difficult challenge of separating nucleus pixels from cell pixels.

CNNs trained with contrast enhanced images and spatial information out perform those that do not use spatial information. However, the CNNs still overestimate nucleus regions, which is in large part due to low visual cues and the existence of spatial variance. Using the previously described set cover algorithm, the overestimation error can be mitigated quite



(a) Proposed CNN method for identifying nucleus regions of cells.

(b) Results of image segmentation using an artifact removal algorithm by [27].



(c) [3] method using florescent reagents to stain nuclei.

Figure 2.9: Nuclei segmentation methods are compared to each other to show the difference between segmentation using florescent reagents and phase contrast microscopy.

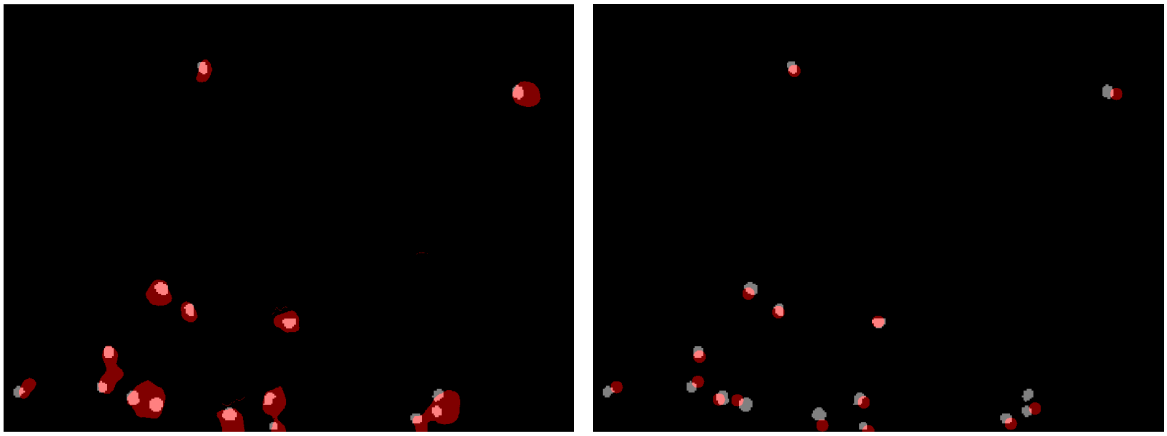
effectively as shown in Figure 2.10.

2.5 Summary

Learning algorithms have demonstrated the ability to identify and classify obvious and clearly distinguishable objects in an image. Nucleus protrusion and spatial variance are two measurements introduced to describe the visual cues and spatial location of nuclei in an image. Two methods were proposed that incorporate spatial location in addition to nucleus protrusion to identify nucleus regions that are not clearly visible or distinguishable. This

Table 2.1: 2D CNN results compared to 1D CNN.

	Contrast Enhanced		Contrast Enhanced And Spatial Information	
	Mean	Standard Deviation	Mean	Standard Devation
Accuracy	0.8679	0.013862	0.86737	0.013614
Precision	0.92099	0.033026	0.97984	0.0071874
Specificity	0.74262	0.13144	0.92671	0.028584
Sensitivity	0.44109	0.060217	0.53557	0.0456
F1 score	0.29677	0.025221	0.34569	0.01909



(a) No set cover analysis.

(b) Using set cover analysis.

Figure 2.10: Nucleus overestimation can be mitigated using set cover analysis in conjunction with CNN trained models.

is information, although available, is not apparent to the human eye. However, a multi-layered convolutional neural network or random forest using a predefined feature space is able to detect these patterns and reveal the location of nuclei in macrophage cells using phase contrast microscopy.

The results of the nucleus segmentation can then be utilized by the watershed variant method proposed by [3]. As previously shown in Figure 2.2g, results demonstrate that if the nucleus is properly estimated using a CNN, cell segmentation can be done rapidly yielding good approximations of cell regions. Unfortunately, CNNs tend to not cope well with high cell density regions despite using spatial information. This is an open ended problem that will need to be addressed in the future.

Chapter 3

PHENOTYPIC ANALYTICS

The *Phenotypic Analytics* component focuses on properly identifying phenotypes that accurately reflect perturbations that have been induced in a cell due to exposure to a biological target, chemical compound, or both. Feature selection has become a critical component of phenotype perturbation analysis when using high dimensional HCS data. This is due to phenotype measurements of multispectral images that may be redundant or noisy providing no useful information while increasing complexity. Phenotype measurements are considered features of the cells they originate from and are produced using different channels in an effort to capture cell and sub-cellular phenotype perturbations. These phenotype perturbations shed light on important biological process changes or disruptions. The visualization of these perturbations has caused image-based HCS to become more prevalent in cell biology and drug discovery endeavors. Feature selection is an essential component in phenotype analysis of cells and their corresponding biological processes by identifying the phenotype perturbations that best distinguish between different controls.

3.1 Defining the Domain

We analyzed three assay plates each with a total of 384 wells. Sixty four of these wells were reserved for control data of healthy cells and those infected with a virulent microbe while the other 320 were treated with different compounds. Two particular inquiries were made with respect to the feature domains and the information provided. The first inquiry was pertaining to how well each domain was able to separate the infected cells from the healthy cells. The second inquiry pertained to how many wells were required to get optimal classification accuracy.

The feature spaces that were defined for the plates were based on the optical microscopy

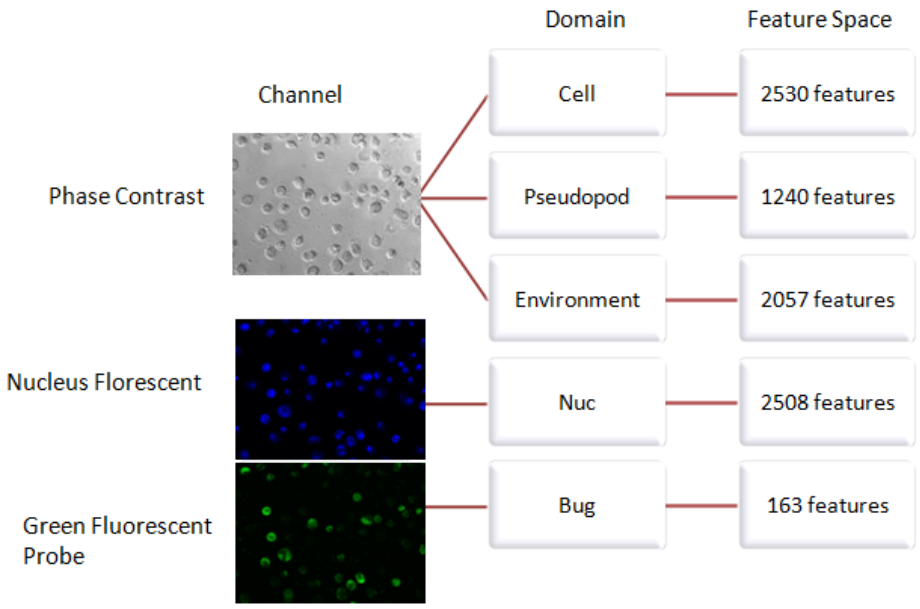
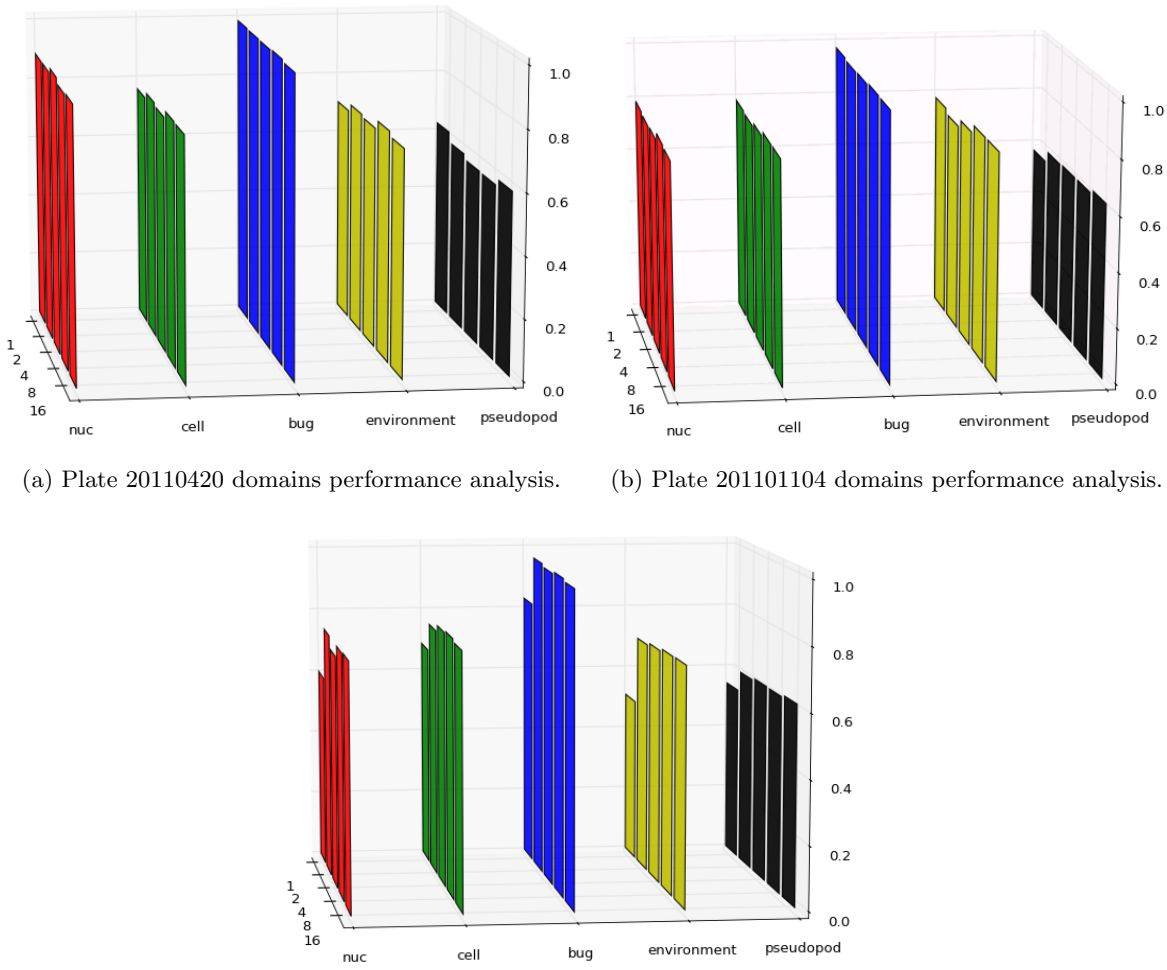


Figure 3.1: Multispectral image channels define the different feature space domains.

images that were obtained. Figure 3.1 demonstrates the composition of the feature space for each image channel. Since the number of features was significantly large, the mRMR algorithms [72] was incorporated to identify the most discriminative features. The Minimum Redundancy Maximum Relevance (mRmR) algorithm is a state-of-the-art feature selection method that not only selects the most discriminative features, but also mitigates the redundancy inherent between the features using the correlation-based measurements on continuous and discrete data. A support vector machine (SVM) model using a radial basis function (RBF) kernel with cost parameter C = 8 was trained in conjunction with features found by mRMR. The number of features was varied from five to one hundred with increments of five. The cost parameter was estimated using a cross-validated grid search. The accuracy across the different domains with using the optimal number of features identified by 10-fold cross-validation is shown in Figure 3.2.

Since the bug domain is defined by the channel corresponding to green florescent protein-labeled bacteria (GFP), it is not surprising that it contains the most informative features. Unfortunately, the domains corresponding to the the phase contrast channel are not nearly as informative or discriminative as those corresponding to the bacteria GFP channel. The



(a) Plate 20110420 domains performance analysis.

(b) Plate 201101104 domains performance analysis.

(c) Plate 201101097 domains performance analysis.

Figure 3.2: The performance of the domains was measured using control data with different algorithms and varying number of training wells.

least descriptive was the pseudopod domain of the phase contrast channel. This domain obtains feature measurements from the identified boundary region of each cell. The explanatory power, or lack there of, may be due to no biological difference between healthy and infected cell boundary regions. Or, and this is more likely, the lack of explanatory power is due to the algorithm's identification of the boundary region, which seems much smoother than actual boundary regions.

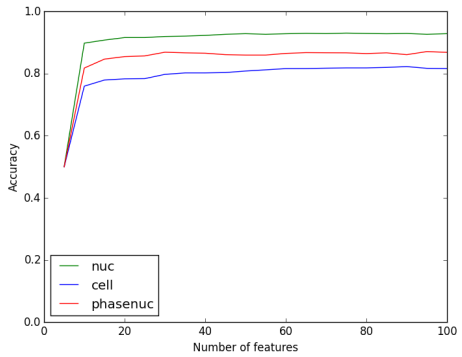
3.1.1 Data Size Analysis

We additionally measured how many wells were needed to obtain a trained classification model in each domain also shown in Figure 3.2. The bug domain demonstrated the need for 1-2 wells to reach optimal classification accuracy. Moreover, the number of features needed to obtain optimal classification accuracy was fairly small - approximately 5-10 features or so. The features identified were indicative of what would biologically be expected with respect to infection. The cell domain behaved a bit more erratically across different plates. In some plates, the number of wells needed to reach optimal classification accuracy was 2, while in others the optimal classification accuracy was achieved using16 wells. Moreover, the number of features needed to reach optimal classification ranged between 25 to 30 features. However, the difference between 2 wells and 16 wells was generally not significant.

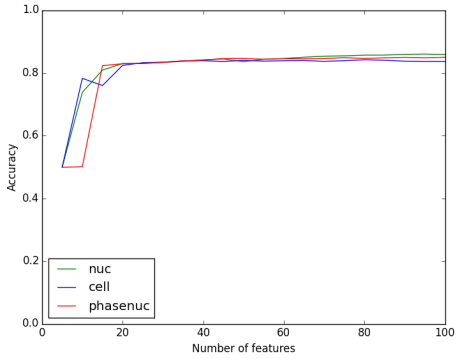
What has been clearly established is that the features in the bug domain defined by the GFP image channel tend to provide the most discriminative information of bacterial infection. Moreover, only a small fraction of these features are needed using only 1-2 control wells of infected and uninfected cells to properly train a highly accurate classifier in the bug domain. The nuc domain provided less descriptive features, although they reached slightly better classification rates than those in the cell domain across the different plates. The features in the cell domain were the third most descriptive followed by the features in the environment and pseudopod domains.

3.2 Domain Information Transfer

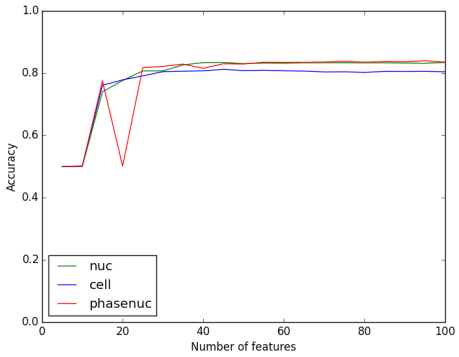
Since current state-of-the-art machine learning methods of feature selection and classification have demonstrated the lack of information in phase contrast channel compared to florescent channels, research was done to determine whether transferring information into the phase contrast channel is a viable option. In order to properly utilize transferred information, there are two primary challenges that must be overcome. First, since the different domains contain heterogeneous data, a connection from one domain to the other is needed



(a) Plate 20110420 phasenuc performance.



(b) Plate 201101104 phasenuc performance.



(c) Plate 201101097 phasenuc performance.

Figure 3.3: The transferring of information from the Hoescht nucleus channel to the phase contrast channel improves accuracy across three different plates.

to transfer information in a manner that improves classification. The second challenge is how to appropriately utilize the transferred information.

3.2.1 Nucleus Transfer

The transferring of information from the Hoescht florescent channel into the phase contrast channel was investigated to determine the subsequent impact on classification accuracy. The simplest way to accomplish this was to take the nucleus masks identified in the Hoescht florescent channel using [3] and use it in the phase contrast channel. A hybrid domain is therefore, defined by the phase contrast channel and nucleus mask and is referred to as the *phasenuc* domain.

3.2.2 Results

Across three different plates, the phasenuc domain provided more informative information than the cell domain. The phasenuc domain classification accuracy was also comparable to the nuc domain classification accuracy in Figure, 3.3b. This indicates that the transferring of information from the Hoescht florescent channel into the phase contrast channel is effective at improving classification accuracy using nucleus mask information. Unfortunately, the information that is transferred corresponds to an individual cell. This means that a nucleus mask in the Hoescht nucleus channel corresponds to a specific cell in the phase contrast channel and, therefore, the cell must exist in the Hoescht nucleus channel in order to use it in the phase contrast channel. Unless, of course, the nucleus can be identified with relying on fluorescent reagents, which was demonstrated as a viable endeavor in the previous chapter.

3.3 Simultaneous Heterogeneous Feature Augmentation and Feature Selection

Research was also done whether it was possible to transfer information from the bug domain to the cell domain in a Heterogeneous framework. Heterogeneous data such as the bug and cell domain data is prevalent in the field of biological image analytics. Since state-of-the-art feature selection algorithms are reliant upon data that share a common feature space, useful information that is shared between domains may not be fully utilized when down-selecting in each domain. Recent transfer learning and domain adaptation algorithms are designed to perform specific classification and pattern recognition tasks but provide limited information on the importance of features within their respective domains. This feature information is vital for the analysis of individual cells. A novel algorithm is proposed that addresses these issues by allowing the transfer of knowledge between domains in order to select the most discriminative features for classification analysis. The algorithm demonstrates its ability to utilize information in different domains to select the features that reach higher discriminative accuracy.

Performance of feature selection algorithms in identifying the most discriminative features generally tend to degrade significantly when small sample sizes are used [80, 81]. Although previously demonstrating that the bug domain using GFP attached to bacteria is capable of identifying the most discriminative features using a single well, all cell populations treated with compounds must still use GFP reagents. As previously noted, the number of cells used in large scale HCS campaigns can reach into the tens of millions. We investigate the use of domain adaptation to transfer information from the bug domain which we consider the source to the cell domain which is considered the target using a small number of cells. This information is subsequently leveraged to assist in feature selection. Previous research in heterogeneous domain adaptation has demonstrated the ability to transfer information between source and target domains [37, 48, 23].

In order to properly utilize transferred information, there are two primary challenges that must be overcome. First, since the heterogeneous data are in different feature spaces, a connection from one domain to the other is needed to transfer information in a manner that facilitates feature selection. The second challenge is how to appropriately utilize the transferred information to improve feature selection. In an attempt to overcome these challenges, a novel feature selection algorithm is proposed based on $\ell_{2,1}$-norm minimization called Simultaneous Feature Augmentation and Feature Selection (SHFAFS) that is capable of transferring knowledge between heterogeneous domains to assist in supervised feature selection. The $\ell_{2,1}$-norm regularization parameter has been extensively used in feature selection endeavors [65, 100, 39, 105]. In addition, Argyriou et al. demonstrated the ability to select features in data sets with heterogeneous tasks and homogeneous domains using $\ell_{2,1}$-norm regularization [6]. The proposed method builds upon these methods by allowing for feature selection in data sets with heterogeneous domains. The main contributions of this method are as follows:

- A principled approach to investigate heterogeneous domain adaptation for feature selection that is able to effectively leverage source domain information in a target

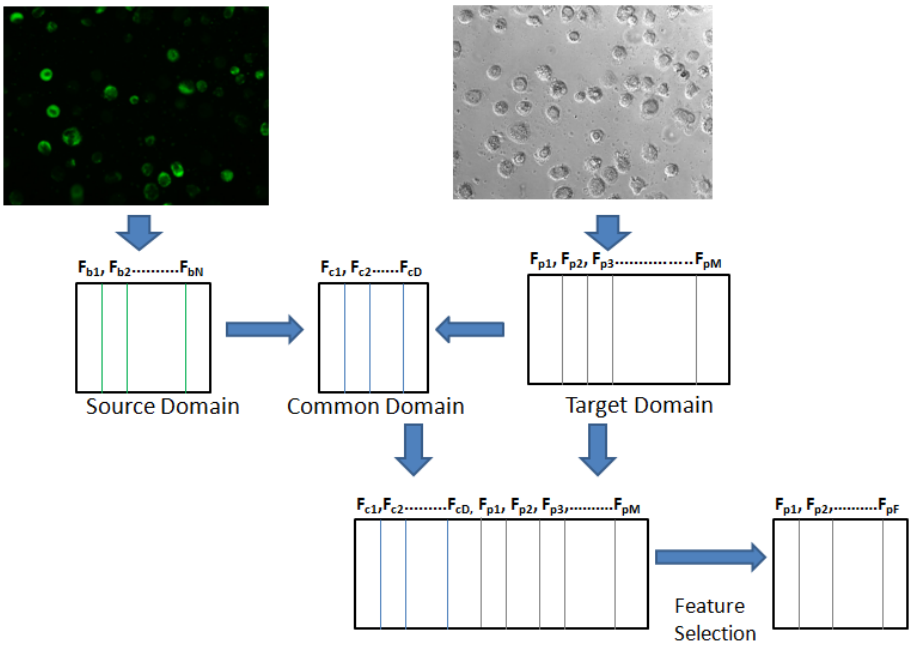


Figure 3.4: SHFAFS transfers information from one domain to the other using a common domain space. Feature selection is subsequently applied to the common domain and target domain combined.

domain is proposed.

- A novel framework, SHFAFS, that uses feature augmentation and sparse learning to accomplish feature selection by utilizing heterogeneous datasets is proposed.

- Experiments are conducted on real world datasets to demonstrate the effectiveness of the proposed framework.

The performance of the algorithm is compared to well known, state-of-the-art supervised feature selection algorithms. Results demonstrate comparable if not better results indicating that data from heterogeneous domains can be used to select pertinent features that improve accuracy in the bright-field domain.

3.3.1 Problem Description

In practice, one very common situation in image-based high content screening is that obtaining the labeled data in some domains is expensive. For example, the use of fluorescent

reagents for nuclear staining of cells incurs costs associated with the reagents themselves as well as the instruments and expertise needed to fully utilize its capabilities. In other words, if $\mathbf{X}_1 \in \mathbb{R}^{n_1 \times d_1}$ is the data matrix for that domain, the number of instances, n_1, is very small while number of features d_1 is usually very large, which usually results in poor performance for the task at hand because a small dataset cannot represent the distribution of the samples in the domain very well. Though obtaining labeled data in one domain is expensive, it is relatively cheap to get labeled data from a closely related domain[1]. For example, bright-field microscopy is much less costly then fluorescent reagents for cells. Let $\mathbf{X}_2 \in \mathbb{R}^{n_2 \times d_2}$ be the data from such a domain. Since $\mathbf{X}_1$ and $\mathbf{X}_2$ are closely related, we would like to transfer some information from $\mathbf{X}_1$ to $\mathbf{X}_2$ to alleviate the small data size problem. However, one problem is that number of features in these two domains are different. Also, $\mathbf{X}_1$ and $\mathbf{X}_2$ lie in two different high-dimensional spaces where information in $\mathbf{X}_2$ cannot be simply transferred to $\mathbf{X}_1$. Thus, we need to make a connection between $\mathbf{X}_1$ and $\mathbf{X}_2$ first.

To make the connection and effectively utilize the heterogeneous features from the two domains, we want to project the two datasets into a common feature space so that these features lie in the same space. Motivated by [25, 68], we augment the the features using projection matrices $\mathbf{P} \in \mathbb{R}^{d_1 \times d_c}$ and $\mathbf{Q} \in \mathbb{R}^{d_2 \times d_c}$ as follows:

$$\begin{aligned} \phi_1(\mathbf{X}_1) &= [\mathbf{X}_1\mathbf{P}, \mathbf{X}_1, \mathbf{0}], \\ \phi_2(\mathbf{X}_2) &= [\mathbf{X}_2\mathbf{Q}, \mathbf{0}, \mathbf{X}_2] \end{aligned} \tag{3.1}$$

In Eq.(3.1), $\mathbf{P}$ is the projection matrix which projects $\mathbf{X}_1$ to $\mathbf{X}_1\mathbf{P} \in \mathbb{R}^{n_1 \times d_c}$ and $\mathbf{Q}$ is another projection matrix to project $\mathbf{X}_2$ to $\mathbf{X}_2\mathbf{Q} \in \mathbb{R}^{n_s \times d_c}$. $\phi_1(\mathbf{X}_1)$ and $\phi_2(\mathbf{X}_2)$ are considered to be in a common space. In addition, $\phi_1(\mathbf{X}_1)$ and $\phi_2(\mathbf{X}_2)$ also contain the original features. The common space feature makes the connection for transfering information and the original features provides discriminative information. Now $\phi_1(\mathbf{X}_1)$ and $\phi_1(\mathbf{X}_2)$ can be think of as data points from the same domain.

With the augmented features defined in Eq.(3.1), we are able to perform feature selec-

[1]Two domains are closely related if they are from the same objects or they share certain properties

tions by utilizing both domains. Since we can treat $\phi_1(\mathbf{X}_1)$ and $\phi_1(\mathbf{X}_2)$ as data points from the same domain, we can apply popular $\ell_{2,1}$-norm based feature selection algorithm on it as follows:

$$\begin{aligned}\min_{\mathbf{W},\mathbf{P},\mathbf{Q}} &(1-\alpha)||\mathbf{Y}_1 - \phi_1(\mathbf{X}_1)\mathbf{W}||_F^2 \\ &+ \alpha||\mathbf{Y}_2 - \phi_2(\mathbf{X}_2)\mathbf{W}||_F^2 + \gamma\|\mathbf{W}\|_{2,1},\end{aligned} \tag{3.2}$$

where $\mathbf{W}$ are the corresponding weight of the features in the source, target, and common domains and can be written as:

$$\mathbf{W} = [\mathbf{W}_a; \mathbf{W}_b; \mathbf{W}_c]\,. \tag{3.3}$$

The augmented feature space in the objective function now contains the original and transformed features from each domain. Feature selection on the augmented feature space is accomplished on the original domains by using the $\ell_{2,1}$-norm regularization parameter on the weights $\mathbf{W}$ of the features in conjunction with the projection matrices $\mathbf{P}$ and $\mathbf{Q}$.

$$\begin{aligned}\min_{\mathbf{W},\mathbf{P},\mathbf{Q}} &(1-\alpha)||\mathbf{Y}_1 - \phi_1(\mathbf{X}_1)\mathbf{W}||_F^2 \\ &+ \alpha||\mathbf{Y}_2 - \phi_2(\mathbf{X}_2)\mathbf{W}||_F^2 \\ &+ \beta(||\mathbf{P}\mathbf{W}_a||_{2,1} + ||\mathbf{Q}\mathbf{W}_a||_{2,1}) \\ &+ \gamma(||\mathbf{W}_a||_{2,1} + ||\mathbf{W}_b||_{2,1} + ||\mathbf{W}_c||_{2,1}) \\ s.t.\quad &\mathbf{P}^T\mathbf{P} = \mathbf{I}, \mathbf{Q}^T\mathbf{Q} = \mathbf{I}\end{aligned} \tag{3.4}$$

An additional orthogonal constraint is given to the projection matrices $\mathbf{P}$ an $\mathbf{Q}$ to ensure no redundancy between features and avoid all zero elements in each matrix. Previous research has demonstrated the utility in optimizing by constraining the projection matrices to contain orthogonal columns [35]. In addition, the $\ell_{2,1}$-norm parameter is also used to ensure sparsity on the weights defined by $\mathbf{P}\mathbf{W}_a$ and $\mathbf{Q}\mathbf{W}_a$ of the source and target domains, respectively. This ensures that the latent space does not over fit the data in the source and target domains.

It is easy to see that this algorithm is a generalized form of the least squares with $\ell_{2,1}$-norm minimization algorithm for two different domains. Simply setting parameter $\mathbf{W}_a$ to

0 yields

$$\min_{\mathbf{W}_b,\mathbf{W}_c} \alpha||\mathbf{Y}_1 - \mathbf{X}_1\mathbf{W}_b||_F^2 + (1-\alpha)||\mathbf{Y}_2 - \mathbf{X}_2\mathbf{W}_c||_F^2 + \gamma||\mathbf{W}_b||_{2,1} + \gamma||\mathbf{W}_c||_{2,1}. \tag{3.5}$$

The process of optimizing the objective function is accomplished in an iterative manner by holding $\mathbf{P}$ and $\mathbf{Q}$ constant when updating the weights $\mathbf{W}$ and holding $\mathbf{W}$ constant when updating $\mathbf{P}$ and $\mathbf{Q}$. This ensures that a global optimum is achieved.

The subproblem of updating $\mathbf{P}$ is given by

$$\min_{\mathbf{P}^T\mathbf{P}=\mathbf{I}} \mathcal{L}(\mathbf{P}) = ||\mathbf{X}_1\mathbf{P}\mathbf{W}_a - \mathbf{A}||_F^2 + ||\mathbf{P}\mathbf{W}\mathbf{a}||_{2,1}, \tag{3.6}$$

where $\mathbf{A} = \mathbf{Y}_1 - \mathbf{X}_1\mathbf{W}_b$. Following [35], to solve this orthogonal constraint problem, we use the gradient descent optimization procedure with curvilinear search [96]. First, we calculate the derivative of $\mathcal{L}(\mathbf{P})$ with respect to $\mathbf{P}$

$$\mathbf{G}_1 = \frac{\partial \mathcal{L}(\mathbf{P})}{\partial \mathbf{P}} = 2\mathbf{X}_1^T\mathbf{X}_1\mathbf{P}\mathbf{W}_a\mathbf{W}_a^T - 2\mathbf{X}_1^T\mathbf{A}\mathbf{W}_a^T + 2\mathbf{D}_{ap}\mathbf{P}\mathbf{W}_a\mathbf{W}_a^T \tag{3.7}$$

where $\mathbf{D}_{ap}$ is a diagonal matrix with $\mathbf{D}_{ap}(i,i) = \frac{1}{2||(\mathbf{P}\mathbf{W}_a)(i,:)||_2}$ [2].

G1 is then used to compute the skew-symmetric matrix

$$\mathbf{F}_1 = \mathbf{G}_1\mathbf{P}^T - \mathbf{P}\mathbf{G}_1^T \tag{3.8}$$

A potential solution for updating $\mathbf{P}$ is then computed using **F1** and a parameter τ.

$$\mathbf{P}_k(\tau) = (\mathbf{I} + \frac{\tau}{2}\mathbf{F_1})^{-1}(\mathbf{I} - \frac{\tau}{2}\mathbf{F_1})\mathbf{P} \tag{3.9}$$

[2] In practice, $\mathbf{D}_{ap}(i,i) = \max(\frac{1}{2||(\mathbf{P}\mathbf{W}_a)(i,:)||_2}, \epsilon)$, where ϵ is a small value such as 10^{-16} to prevent $\mathbf{D}_{ap}(i,i)$ from being too close to zero

The parameter τ controls the step size of the curvilinear search function that derives $\mathbf{P}_k$. The parameter τ is updated using an iterative process that decreases its values using a another parameter μ such that

$$\tau = \tau * \mu \tag{3.10}$$

where $0 \leq \mu \leq 1$. The value of τ that solves the subproblem of updating $\mathbf{P}$ is found when the Armijo-Wolfe conditions as defined in [35] are met. If the conditions are not met within a specified number of iterations, a local minimum has been found for $\mathbf{P}$ and $\mathbf{Q}$ and the search is completed.

The subproblem of updating $\mathbf{Q}$ is

$$\min_{\mathbf{Q}^T\mathbf{Q}=\mathbf{I}} \mathcal{L}(\mathbf{Q}) = ||\mathbf{X}_2\mathbf{Q}\mathbf{W}_a - \mathbf{C}||_F^2 + ||\mathbf{Q}\mathbf{W}\mathbf{a}||_{2,1}, \tag{3.11}$$

where in this case $\mathbf{C} = \mathbf{Y}_2 - \mathbf{X}_2\mathbf{W}_c$. The update rule of $\mathbf{Q}$ is found using the same method described for the update rule of $\mathbf{P}$. The derivative of $\mathbf{Q}$ is first found.

$$\begin{aligned}\mathbf{G}_2 = \frac{\partial \mathcal{L}(\mathbf{Q})}{\partial \mathbf{Q}} &= 2\mathbf{X}_2^T\mathbf{X}_2\mathbf{Q}\mathbf{W}_a\mathbf{W}_a^T - 2\mathbf{X}_2^T\mathbf{A}\mathbf{W}_a^T \\ &+ 2\mathbf{D}_{aq}\mathbf{Q}\mathbf{W}_a\mathbf{W}_a^T.\end{aligned} \tag{3.12}$$

The skew-symmetric matrix is then computed

$$\mathbf{F}_2 = \mathbf{G}_2\mathbf{Q}^T - \mathbf{Q}\mathbf{G}_2^T \tag{3.13}$$

A potential solution for updating $\mathbf{Q}$ is then given by

$$\mathbf{Q}_{pot}(\tau) = (\mathbf{I} + \frac{\tau}{2}\mathbf{F_2})^{-1}(\mathbf{I} - \frac{\tau}{2}\mathbf{F_2})\mathbf{Q} \tag{3.14}$$

The optimal parameter value for τ is then found using the iterative approach that was described for finding the updated $\mathbf{P}$.

The subproblem of updating $\mathbf{W}_a$ is

$$\begin{aligned}\min_{\mathbf{W}_a} \mathcal{L}(\mathbf{W}_a) = &||\mathbf{X}_1\mathbf{P}\mathbf{W}_a - \mathbf{A}||_F^2 + ||\mathbf{X}_2\mathbf{Q}\mathbf{W}_a - \mathbf{C}||_F^2 \\ &+ ||\mathbf{P}\mathbf{W}_a||_{2,1} + ||\mathbf{Q}\mathbf{W}_a||_{2,1} + ||\mathbf{W}_a||_{2,1}\end{aligned} \tag{3.15}$$

The derivative of Eq.(3.15) is

$$\begin{aligned}\frac{\partial \mathcal{L}(\mathbf{W}_a)}{\partial \mathbf{W}_a} = &2\mathbf{P}^T\mathbf{X}_1^T\mathbf{X}_1\mathbf{P}\mathbf{W}_a - 2\mathbf{P}^T\mathbf{X}_1^T\mathbf{A} \\ &+ 2\mathbf{Q}^T\mathbf{X}_2^T\mathbf{X}_2\mathbf{Q}\mathbf{W}_a - 2\alpha\mathbf{Q}^T\mathbf{X}_2^T\mathbf{C} \\ &+ 2\mathbf{P}^T\mathbf{D}_{ap}\mathbf{P}\mathbf{W}_a + 2\mathbf{Q}^T\mathbf{D}_{aq}\mathbf{Q}\mathbf{W}_a \\ &+ 2\mathbf{D}_a\mathbf{W}_a,\end{aligned} \tag{3.16}$$

where $\mathbf{D}_a$, $\mathbf{D}_{ap}$, and $\mathbf{D}_{aq}$ are diagonal matrices with $\mathbf{D}_a(i,i) = \frac{1}{2||\mathbf{W}_a(i,:)||_2}$, $\mathbf{D}_{ap}(i,i) = \frac{1}{2||(\mathbf{P}\mathbf{W}_a)(i,:)||_2}$ and $\mathbf{D}_{aq}(i,i) = \frac{1}{2||(\mathbf{Q}\mathbf{W}_a)(i,:)||_2}$, respectively. The update rule of $\mathbf{W}_a$ is

$$\begin{aligned}\mathbf{W}_a = &((1-\alpha)\mathbf{P}^T\mathbf{X}_1^T\mathbf{X}_1\mathbf{P} + \alpha\mathbf{Q}^T\mathbf{X}_2^T\mathbf{X}_2\mathbf{Q} \\ &+ \beta(\mathbf{P}^T\mathbf{D}_{ap}\mathbf{P} + \mathbf{Q}^T\mathbf{D}_{aq}\mathbf{Q}) + \gamma\mathbf{D}_a)^{-1} \\ &* ((1-\alpha)\mathbf{P}^T\mathbf{X}_1^T\mathbf{A} + \alpha\mathbf{Q}^T\mathbf{X}_2^T\mathbf{C})\end{aligned} \tag{3.17}$$

The subproblem of updating $\mathbf{W}_b$ is

$$\min_{\mathbf{W}_b} \mathcal{L}(\mathbf{W}_b) = ||\mathbf{X}_1\mathbf{W}_b - \mathbf{E}||_F^2 + \frac{\gamma}{(1-\alpha)}||\mathbf{W}_b||_{2,1} \tag{3.18}$$

where $\mathbf{E} = \mathbf{Y}_1 - \mathbf{X}_1\mathbf{P}\mathbf{W}_a$. The derivative is

$$\frac{\partial \mathcal{L}(\mathbf{W}_b)}{\partial \mathbf{W}_b} = 2\mathbf{X}_1^T\mathbf{X}_1\mathbf{W}_b - 2\mathbf{X}_1^T\mathbf{E} + 2\gamma\mathbf{D}_b\mathbf{W}_b \tag{3.19}$$

where $\mathbf{D}_b$ is a diagonal matrix with $\mathbf{D}_b(i,i) = \frac{1}{2||\mathbf{W}_b(i,:)||_2}$. Thus, the updating rule of $\mathbf{W}_b$ is

$$\mathbf{W}_b = (\mathbf{X}_1^T\mathbf{X}_1 + \frac{\gamma}{(1-\alpha)}\mathbf{D}_b)^{-1}(\mathbf{X}_1^T\mathbf{E}) \tag{3.20}$$

The subproblem of updating $\mathbf{W}_c$ is

$$\min_{\mathbf{W}_c} \mathcal{L}(\mathbf{W}_c) = ||\mathbf{X}_2\mathbf{W}_c - \mathbf{F}||_F^2 + \frac{\gamma}{\alpha}||\mathbf{W}_c||_{2,1} \tag{3.21}$$

where $\mathbf{F} = \mathbf{Y}_2 - \mathbf{X}_2\mathbf{Q}\mathbf{W}_a$. The derivative is

$$\frac{\partial \mathcal{L}(\mathbf{W}_c)}{\partial \mathbf{W}_c} = 2\mathbf{X}_2^T\mathbf{X}_2\mathbf{W}_c - 2\mathbf{X}_2^T\mathbf{F} + 2\frac{\gamma}{\alpha}\mathbf{D}_c\mathbf{W}_c \tag{3.22}$$

where $\mathbf{D}_c$ is a diagonal matrix with $\mathbf{D}_c(i,i) = \frac{1}{2||\mathbf{W}_c(i,:)||_2}$. Thus, the updating rule of $\mathbf{W}_c$ is

$$\mathbf{W}_c = (\mathbf{X}_2^T\mathbf{X}_2 + \frac{\gamma}{\alpha}\mathbf{D}_c)^{-1}(\mathbf{X}_2^T\mathbf{F}) \tag{3.23}$$

Krylov subspaces are the initial building blocks for the projection matrices of the SHFAFS algorithm. Krylov subspaces comprise of the data X and corresponding label space Y where their relationship is

$$\mathbf{Z}_k = span\{\mathbf{X}^T\mathbf{Y}, (\mathbf{X}^T\mathbf{X})\mathbf{X}^T\mathbf{Y}, ...(\mathbf{X}^T\mathbf{X})^{k-1}\mathbf{X}^T\mathbf{Y}\}, \tag{3.24}$$

where $\mathbf{Z}_k$ is the k_{th} column in the projection matrix that is part of the optimal solution for

$$\min_{\mathbf{W}} ||\mathbf{Y} - \mathbf{X}\mathbf{Z}\mathbf{W}||_2. \tag{3.25}$$

Projecting into Krylov subspace has demonstrated the ability to obtain an approximate optimal solution that significantly reduces the residual error with a small number of columns in $\mathbf{Z}$ [32]. This is quite suitable for transferring domain information since the label space is common to both domains in training data. Thus the domain space of the labeled data is shared between both domains of interest.

In order to satisfy the orthogonal constraint, the Lanczos-Golub-Kahan (LGK) bidiagnolization method was use to find the initial $\mathbf{P}$ and $\mathbf{Q}$ matrices that solved the following optimization problems [32].

$$\min_{\mathbf{W}}(1-\alpha)||\mathbf{Y}_1 - \mathbf{X}_1\mathbf{Z}_1\mathbf{W}||_2 \tag{3.26}$$

$$\min_{\mathbf{W}}\alpha||\mathbf{Y}_2 - \mathbf{X}_2\mathbf{Z}_2\mathbf{W}||_2 \tag{3.27}$$

We set $\mathbf{P} = \mathbf{Z}_1$ and $\mathbf{Q} = \mathbf{Z}_2$. Multiplying $\mathbf{X_1}$ by $\mathbf{P}$ and $\mathbf{X_2}$ by $\mathbf{Q}$ columns projects $\mathbf{X_1}$ and $\mathbf{X_2}$ into a Krylov space. In both cases, $\mathbf{P}$ and $\mathbf{Q}$ are orthogonal and the objective value is minimized with a fraction of the total feature space for both domains.

The weights $\mathbf{W}_a$,$\mathbf{W}_b$ and $\mathbf{W}_c$ are all derived by letting the respective diagonal matrices equal to an identity matrix.

$$\mathbf{W}_c = (\mathbf{X}_2^T\mathbf{X}_2 + \frac{\gamma}{\alpha}\mathbf{I}_c)^{-1}(\mathbf{X}_2^T\mathbf{Y_2}) \tag{3.28}$$

$$\mathbf{W}_b = (\mathbf{X}_1^T\mathbf{X}_1 + \frac{\gamma}{(1-\alpha)}\mathbf{I}_b)^{-1}(\mathbf{X}_1^T\mathbf{Y_1}) \tag{3.29}$$

Once $\mathbf{P}$, $\mathbf{Q}$, $\mathbf{W}_b$, and $\mathbf{W}_c$ were found, $\mathbf{W}_a$ is initialized as

$$\begin{aligned}\mathbf{W}_a = ((1-\alpha)\mathbf{P}^T\mathbf{X}_1^T\mathbf{X}_1\mathbf{P} + \alpha\mathbf{Q}^T\mathbf{X}_2^T\mathbf{X}_2\mathbf{Q} \\ + \beta(\mathbf{P}^T\mathbf{I}_{ap}\mathbf{P} + \mathbf{Q}^T\mathbf{I}_{aq}\mathbf{Q}) + \gamma\mathbf{I}_a)^{-1} \\ * ((1-\alpha)\mathbf{P}^T\mathbf{X}_1^T\mathbf{A} + \alpha\mathbf{Q}^T\mathbf{X}_2^T\mathbf{C}).\end{aligned} \tag{3.30}$$

Algorithm 1 Simultaneous Heterogeneous Feature Augmentation and Feature Selection

Require: $\mathbf{X}_1 \in \mathbb{R}^{n_1 \times d_1}, \mathbf{Y}_1 \in \mathbb{R}^{n_1 \times c}, \mathbf{X}_2 \in \mathbb{R}^{n_2 \times d_2}, \mathbf{Y}_2 \in \mathbb{R}^{n_2 \times c}, \alpha, \beta, \gamma, m$

Ensure: The rank of features in descending order

1: Initialize $\mathbf{P}, \mathbf{Q}, \mathbf{W}_a, \mathbf{W}_b, \mathbf{W}_c$

2: **repeat**

3: Update $\mathbf{W}_a$ using Equation 3.17

4: Update $\mathbf{P}, \mathbf{Q}$ using [35]

5: Update $\mathbf{W}_b$ using Equation 3.20

6: Update $\mathbf{W}_c$ using Equation 3.23

7: calculate objective function

8: **until** stopping criteria is reached

9: return feature index of $\mathbf{X}_1$ in descending order according to $||\mathbf{W}_b(i,:)||_2$ and feature index of $\mathbf{X}_2$ in descending order accodring to $||\mathbf{W}_c(i,:)||_2$

After applying Algorithm 1, we get the feature scores in descending order. These scores describe the amount of variation that a feature explains in the response variable, suggesting

its contribution in reaching the optimal objective function value. The top k features can subsequently be selected by the user for training using any given classifier. Moreover, phenotype analysis can be done using the score ranked features to further understand biological properties of the underlying domains.

The SHFAFS algorithm has four primary parameters, α, γ, β, and m that influence the SHFAFS performance. The α parameter defines the influence of the two domains relative to each other on the objective function. The value ranges from 0 to 1. The α value was measured at values $[0.25, 0.5, 0.75]$. The gamma parameter is associated with the $\ell_{2,1}$-norm cost function for the feature weights W. This ensures small values across different classes and sparsity of the weights. The values of γ were set to $[0.1, 0.5, 1, 2, 10]$. The β parameter controls the sparsity of the weights defined by the projection matrices in conjunction with $\mathbf{W}_a$. The values of β were set to $[0.1, 0.5, 1, 2, 10]$. The m parameter is the size of the common domain space. The m parameter was set to 15 given that there was little change in the residual of the model with any larger value. Figure 3.5 demonstrates the effects that the parameters have on the features selected and their corresponding accuracy. In this case, $\alpha = 0.5$ to analyze the behavior of the algorithm w.r.t. the β and γ parameters. The lower the β and γ parameters, the higher the accuracy from the resultant features. Since, the β and γ parameters control sparsity, this dataset required less sparsity and, therefore, more information from source and target domains to select the best performing feature subset. We did a similar grid search across all parameter values for each training data set to fully optimize the parameters.

3.3.2 Related Works

Feature reduction in heterogeneous domain adaptation (HDA) is generally accomplished through common latent space projection of source and target domains. The common latent space projection occurs when two heterogeneous domains are projected into the same reduced feature space. For instance, Shi et al. proposed using a linear transformation objective function called Heterogeneous Spectral Mapping (HeMap) to define projection

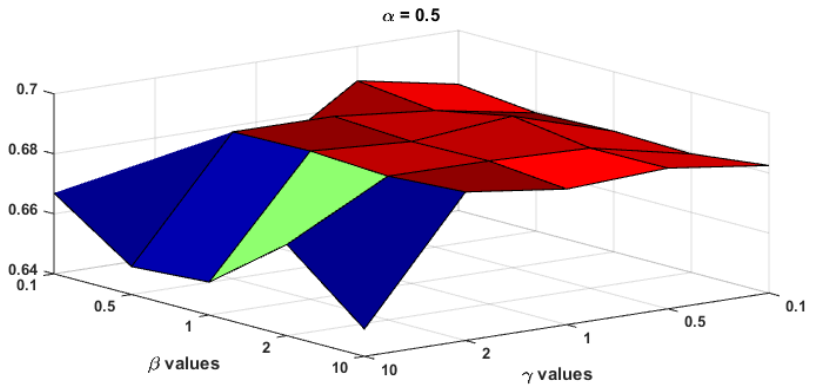


Figure 3.5: A grid search was performed to find the best parameter values for β and γ.

matrices for the source and target domain. The projection matrices were defined by the top eigenvalues and vectors of a matrix A which is found through a linear combination of the source and target domains [79]. Wang et al. proposed the use manifold alignment to create projection matrices that map source and target domains into a common feature space. This method aligns the manifolds by matching corresponding instances while preserving local geometry [94]. Duan et al. proposed a heterogeneous feature augment (HFA) method that used the standard SVM with hinge loss to find source and target projection matrices [30]. Unfortunately, reduction of feature space using projection matrices makes analysis of original target feature space much more difficult. As a result, phenotype analysis becomes much less intuitive in high content screening and cell biology. We, therefore, expand upon HDA analysis by making the feature selection in the target domain of primary importance. This not only identifies descriptive phenotypes but also improves performance of classification.

3.3.3 Results

In order to test the feasibility of the proposed algorithm, a biological assay was designed using image-based high content screening (HCS) data. The assay was designed using images from primary human monocyte-derived macrophages (hMDMs) cell. The data consisted of two classes defined by healthy hMDMs cells and those infected with a green fluorescent protein (GFP)-tagged strain of virulent Francisella tularensis. The objective was to determine

the ability of the SHFAFS to utilize fluorescent domain information in feature selection of bright-field domain features using a small number of data points in each domain. A combination of public and proprietary algorithms measured 6,000 features for each cell across the 2 different domains of interest. Approximately 3,000 of these features were obtained from WNDCharm [66]. A portion of the proprietary feature measurements generated have been described previously in [3]. There were a total of 5,827 features in the phase contrast domain and 162 features in the bacterial fluorescence domain. Therefore, we leveraged the data in the fluorescent domain to reduce the feature space of the phase contrast domain.

The SHFAFS algorithm was compared to four well known feature selection algorithms:

- Relieff [45] is a multi-class feature selection algorithm that weights features by selecting random data points and computing the distance to their the closest **k** neighbors in the same and different classes.

- The Minimum Redundancy Maximum Relevance (mRmR) [72] not only selects the most discriminative features, but also mitigates the redundancy inherent between the features using the the F-statistic and correlation measurements.

- The Fast Correlation Based Filter (FCBF) [103] compares feature-class and feature-feature correlation using symmetrical uncertainty to select the most discriminative features and remove redundant features.

- The least squares with $\ell_{2,1}$-norm minimization algorithm is the SHFAFS algorithm on a single domain setting $\mathbf{W}_a = 0$.

Each algorithm was ran on four different size training data sets of 20, 50, and 75, 100 data points. Testing was then done by randomly selecting 5 different wells on an assay plate. The total number of data points for each testing data set was ≈ 1500 cells. Sampling without replacement for both training and testing data was accomplished using balanced data sets. The algorithms were implemented using matlab and a feature selection package provided and maintained by Arizona State University [109]. The feature selection algorithms were

Table 3.1: Comparison of SHFAFS method to other well known feature selection algorithms.

Algorithm	25	50	75	100
SHFAFS	**74.03**	**75.14**	**75.23**	**77.28**
$\ell_{2,1}$-norm	74.01	74.90	74.72	76.47
Relieff	70.48	74.90	74.87	73.83
MRMR	74.01	72.79	73.68	75.02
FCBF	73.03	71.02	72.07	74.92
All Features	54.67	69.50	70.66	72.54

compared using a random forest classifier. The classifier was implemented using $N = 50$ trees. The random forest classifier was trained on the dataset using the top k features of each of the feature selection algorithms. The value of k features was iterated from 1 to 100 in increments of five. The average maximum accuracy from 10 iterations of a random forest classifier is subsequently reported in Table 3.1 for the different training sizes used.

The SHFAFS algorithm reached accuracy rates comparable or better than the other feature selection algorithms employing the use of heterogeneous domains. Although the gains in accuracy were relatively small, the importance lies in that information can be transferred from one heterogeneous domain to the other to improve selecting the most discriminative features. This is especially beneficial to phenotype analysis of cells where a small number features is much more reasonable for investigation then large numbers of redundant features. Moreover, our method demonstrated as good as or better performance than simply utilizing a more specific form of our algorithm in the least squares with $\ell_{2,1}$-norm minimization.

The rate of change in accuracy improvement can be seen in Figure 3.6. It demonstrates as the number of source data points increases, the improvement change in classification

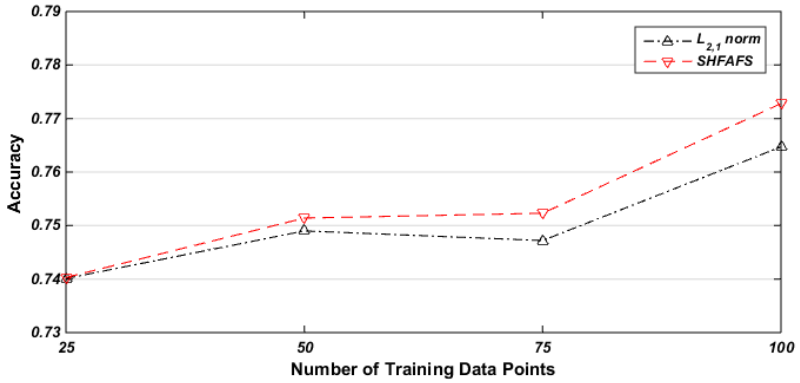


Figure 3.6: SHFAFS improvement in accuracy becomes larger as the number of data points in the source domain increases.

accuracy also increases. The most improvement in accuracy is, seen, when the number of source data points is 100.

Unfortunately, the extent to which SHFAFS can contribute to selecting better features is reliant upon the amount of beneficial information in the source domain. In addition, there are more parameters that need to be tuned adding complexity to the traditional least squares with $\ell_{2,1}$-norm minimization method. Finding these parameters requires more computation time as well. In the following, section we will describe how the parameters were tuned. The ability of SHFAFS to transfer information from one domain to the other to select the most discriminative features using a relatively small number of data points, provides a promising alternative to traditional feature selection algorithms relegated to the information contained in a single domain.

3.3.4 Summary

A novel feature selection algorithm called Simultaneous Heterogeneous Feature Augmentation and Feature Selection (SHFAFS) has been proposed. The algorithm initially finds projection matrices in Krylov space for source and target domains. The projection matrices assist in selecting the best features in the target domain through the use of a least squares with $\ell_{2,1}$-norm minimization function. The method proposes using alternating optimization where the projection matrices are optimized using a curvilinear search method.

The algorithm was compared to state-of-the-art feature selection algorithms demonstrating generally comparable or better performance. We further demonstrated that this algorithm is a generalization of the least squares with $\ell_{2,1}$-norm minimization method. We applied this algorithm to image-based HCS data focusing on fluorescent and phase contrast domains. The results demonstrate that florescent domains are capable of assisting bright-field domains using a small number of data points relative to the data set.

The SHFAFS algorithm can be extended in a number of different directions for future research. Specifically, future work will investigate different heterogeneous data sets and tasks such as tumor samples with genetic expressions. In addition, future work will focus on identifying the parameters in a non-exhaustive search manner. We will investigate whether domain correlations can better determine the parameters to use. Additionally, we plan on investigating the extent to which a source or target domain size are imbalanced and how that affects the amount of information that is transferred.

3.4 Maximum Distance Minimum Error

Another challenge that was investigated was weather the distribution of the data in each of the domains could negatively impact the selection of features that were the most discriminative. The features that measure phenotype perturbations in high dimensional space are often continuous values. Discretization is often employed when continuous data is utilized requiring additional resources and testing to determine how many bins the data should be split into [53]. Methods, such as minimum description language [33], are discretization processes that try to find the most optimal number of bins that reduce the error within each bin. Nevertheless, the discretization process may lose crucial data, require more resources to be utilized, and may not provide an optimal binning of the data. As such, statistically-based feature selection methods are often preferred to those that require an additional discretization step. Methods such as F-test, T-test, Fisher's method, and Maximum Relevance Minimum Redundancy (mRMR) are all powerful well known feature selection algorithms that utilize continuous data when selecting the optimal feature

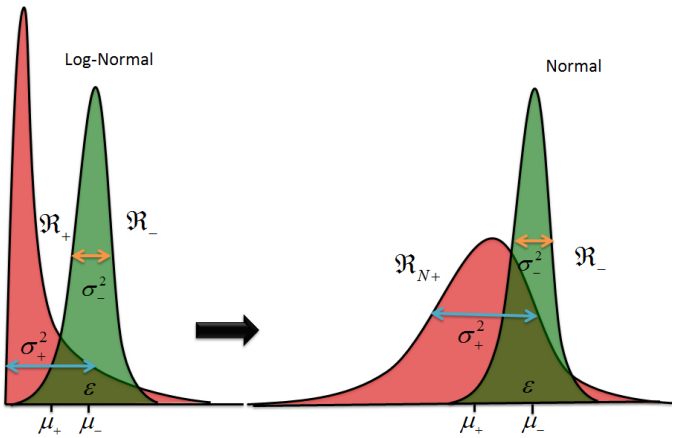


Figure 3.7: Log-normal distribution is not ideally suited for parametric analysis. The mean and standard deviation are both adversely influenced causing the error region to be over estimated within a certain region of the log-normal distribution.

set. However, these algorithms utilize mean and standard deviation parameters to select features that can best distinguish between classes.

Relying on parametric-based feature selection algorithms has its inherent limitations. For instance, image-based HCS data focuses on measuring cell phenotype perturbations caused by the introduction of a biological target and chemical compounds. Generally, as previously described, the HCS data is categorized into two groups of healthy, unperturbed cells and those perturbed due to a biological target of interest called negative and positive control data, respectively. Using differentiated cell lines, there should be similarity and minimal phenotype perturbations in negative control data, reflecting feature measurements that follow a normal distribution with small variance. Differentiated cell line populations tend to be uniform where very little difference exists between one individual cell and the next. The same assumptions cannot be made about positive control data. For instance, cells infected with a bacterial biological target may be impacted differently depending on the infection rate and latent influences on biological processes caused by different bacteria. Generally, a Gaussian distribution is assumed using HCS datasets justifying this assumption by the enormous amount of data generated and the central limit theorem [82].

The variance of the phenotype perturbation and its corresponding feature measurement in positive control data may be much greater than feature measurements of unperturbed

cells. In addition, the phenotype perturbations may not follow a normal distribution. Several studies have found that real world datasets, such as biological data, often resemble more of a log-normal distribution than normal distribution [52, 77]. Figure 3.7 demonstrates how positive and negative control distributions may very depending on the effects that a bacteria has on a cell line. If mean and standard deviation are computed for each distribution, the infected cells distribution is capable of providing rather dubious results by assuming normal distributions. This has severe implications with respect to parametric feature selection algorithms where error regions may be over estimated with phenotype perturbation measurements that follow a log-normal distribution. As demonstrated in Figure 3.7, if the error region is over estimated, parametric feature selection algorithms may inaccurately consider a highly discriminative measurement on par with a low discriminative measurement that demonstrates a normal distribution. Image-based HCS high-dimension data may also be inherently noisy and redundant as has been previously demonstrated [92]. Well known issues such as plate effects and microscopy systematic noise such as those encountered using phase contrast technology, may yield datasets that are not normally distributed [59]. Therefore, this systematic noise may change the data distribution and pose a challenge that normal parametric assumptions may be ill equipped to handle.

A novel method based on a nonparametric approach that is better suited to identifying pertinent features in image-based HCS datasets is proposed. This approach is based on the well known Kolmogorov-Smirnov (K-S) test that is often utilized to test the similarity between two distributions when no assumption can be made about the distribution. Our contributions are as follows:

- Overestimation of error region when assuming normal distribution. In section 3.4.1, we demonstrate that a log normal distribution will always provide a normal distribution counterpart that will increase the error region of a feature within a specific interval.

- The K-S test provides a robust non-parametric alternative to feature selection. In 3.4.2, we demonstrate how the K-S test better discerns between two classes without

assuming normal distribution.

- Data sets where one class follows a normal distribution and the other a log-normal distribution will yield sub-optimal feature selection results when used with parametric-based approaches. In section 3.4.3, noisy synthetic data is generated where the most discriminative features are those that follow a log-normal distribution in one class and show how our non-parameteric based approach outperforms parametric based approaches.

There are two primary reasons for utilizing synthetic data: (1) Synthetic data is easy to control with clearly known properties, and (2) demonstrates the necessity to pursue a non-parametric approach. In addition to utilizing synthetic data, we also utilize well known datasets previously used to test feature selection, and real world image-based HCS data.

3.4.1 Problem Description

Let $X \in \mathbb{R}^{nxd}$ be dataset where n is the number of data points in a dimensional space of size d. Let $Y \in \{0, 1\}$ represent the class labels of positive control data and negative control data, respectively. Furthermore, we assign the positive class distribution $(+)$ to having a log-normal distribution and negative distribution $(-)$ to having a normal distribution. This is more reflective of the measurements of the biological processes that are perturbed by bacterial targets of interest.

The corresponding normal distribution will produce an error region of a log-normal distribution that is over estimated within a certain interval. Parametric feature selection algorithms will give lower results when the majority of other normal distributions intersect this interval.

Theorem 1 (Normal - Log-Normal error relationship). *Let $f_{LN} \sim \mathcal{N}(x, \mu, \sigma)$ be a log-normal distribution and $f_N \sim \mathcal{N}(x, m, s)$ be the corresponding normal distribution, then f_N will produce an interval region $\mathcal{R}_\epsilon$ that over estimates the error regions in parametric feature selection.*

Proof. The mean m and variance s^2 of the corresponding normal distribution to the log-normal distribution is

$$\begin{aligned} m =& e^{(\mu+\frac{\sigma^2}{2})} = e^{\mu} \cdot e^{\frac{\sigma^2}{2}}, \\ s^2 =& e^{(2\mu+\sigma^2)}(e^{\sigma^2}-1) = e^{2\mu}e^{\sigma^2}(e^{\sigma^2}-1). \end{aligned}$$

It is easy to show that $m \geq \mu + 1$. Since the variance σ^2 must necessarily be positive

$$e^{\frac{\sigma^2}{2}} \geq 1,$$

and

$$e^{\mu} \cdot 1 \geq \mu.$$

If $\mu \geq 0$ than $m \geq 1$. If $\mu \leq 0$ than m will approach 0 but remain positive.

This shift of distribution creates an interval region between two values, a and b, where the area of the probability density function (pdf) of the log-normal distribution is less than the area of the corresponding normal pdf.

$$\frac{1}{2s\sqrt{(2\pi)}} \int_a^b e^{\frac{-(x-m)^2}{(2s)}} > \frac{1}{2x\sigma\sqrt{(2\pi)}} \int_a^b e^{\frac{-(ln(x)-\mu)^2}{2\sigma^2}}.$$

Using Bowling *et al.*'s logistic approximation of the cumulative distribution function [12], we demonstrate that the interval spanned from the mean to one standard deviation away from the mean of the corresponding normal distribution $f_{\mathcal{N}}$ will produce an over estimated error region of the true log-normal distribution.

$$\phi(z) = \frac{1}{1 + e^{-(\alpha z^3 + \beta z)}},$$

$$s.t.$$

$$\alpha = 0.07056$$

$$\beta = 1.5976,$$

where the z value is the given centered and standardized value of x.

$$z = \frac{x - \mu}{\sigma}.$$

We can approximate the probability between two values z_1 and z_2 by

$$\phi(z_2) - \phi(z_1) = \frac{1}{1 + e^{-(\alpha z_2^3 + \beta z_2)}} - \frac{1}{1 + e^{-(\alpha z_1^3 + \beta z_1)}}.$$

In a normal distribution, the area between the mean and two standard deviations away is a constant $C \approx 0.47724$. However, in a log-normal distribution, the corresponding region tends to shift given the mean and variance.

Allowing $x_1 = m$ and $x_2 = m + 2 * s$ to represent the region between a and b of the corresponding normal distribution, the z values for a log-normal distribution become

$$z_1 = \frac{ln(x_1) - \mu}{\sigma}$$
$$z_2 = \frac{ln(x_2) - \mu}{\sigma}.$$

z_1 can be reduced to the following

$$\begin{aligned} z_1 &= \frac{ln(x_1) - \mu}{\sigma} \\ &= \frac{ln(e^{(\mu + \frac{\sigma^2}{2})}) - \mu}{\sigma} \\ &= \frac{(\mu + \frac{\sigma^2}{2}) - \mu}{\sigma} \\ &= \frac{\sigma}{2} \end{aligned}$$

Since σ must be positive, the starting point of the log-normal distribution cannot be less than the mean of the corresponding normal distribution.

The z_2 can also be reduced

$$\begin{aligned} z_2 &= \frac{ln(x_2) - \mu}{\sigma} \\ &= \frac{ln(e^{(\mu + \frac{\sigma^2}{2})} + 2 * ((e^{(2\mu+\sigma^2)}(e^{\sigma^2} - 1))^{1/2}) - \mu}{\sigma} \\ &= \frac{\sigma}{2} + \frac{ln(1 + 2 * (e^{\sigma^2} - 1)^{1/2})}{\sigma}. \end{aligned} \tag{3.31}$$

Since the normal distribution area is constant, the following is shown

$$\phi(z_2) - \phi(z_1) < C. \tag{3.32}$$

Setting

$$A = e^{\alpha((z_2)^3 + \beta z_2)}$$

$$B = e^{\alpha((z_1)^3 + \beta z_1)},$$

we can rewrite Equation 3.32 as

$$\begin{aligned} &\frac{1}{1 + \frac{1}{A}} - \frac{1}{1 + \frac{1}{B}} < C \\ &\frac{A - B}{(A+1)(B+1)} < C \\ &1 < C + (C * B) + \frac{C}{A} + \frac{C * B}{A} + \frac{B}{A} \end{aligned} \tag{3.33}$$

$$1 < C + (C * B) + \frac{C}{A} + (C + 1) * \frac{B}{A}$$

Since $\sigma > 0$ then B is lower bounded by 1.

Since Equation 3.31 contains σ in the denominator, we take the limit of σ as it approaches 0.

$$\lim_{\sigma \to 0} \frac{ln(1 + 2 * (e^{\sigma^2} - 1)^{1/2})}{\sigma}$$

Lemma 2.

$$\lim_{x\to 0} \frac{e^x - 1}{x} = 1$$

$$\lim_{x\to 0} e^x - 1 = x$$

Allowing $x = \sigma^2$, we can make the following substitution

$$\lim_{\sigma\to 0} \frac{ln(1 + 2 * (e^x - 1)^{1/2})}{\sigma}$$
$$\lim_{\sigma\to 0} \frac{ln(1 + 2 * (x)^{1/2})}{\sigma}$$
$$\lim_{\sigma\to 0} \frac{ln(1 + 2 * (\sigma^2)^{1/2})}{\sigma}$$
$$\lim_{\sigma\to 0} \frac{ln(1 + 2 * \sigma)}{\sigma}$$

Lemma 3.

$$\lim_{x\to 0} \frac{ln(x + 1)}{x} = 1$$

$$\lim_{x\to 0} ln(x + 1) = x$$

Now we set $x = 2 * \sigma$, we can rewrite the

$$\lim_{\sigma\to 0} \frac{ln(1 + x)}{\sigma}$$
$$\lim_{\sigma\to 0} \frac{x}{\sigma}$$
$$\lim_{\sigma\to 0} \frac{2 * \sigma}{\sigma}$$
$$\lim_{\sigma\to 0} 2$$

Therefore, as σ approaches 0, z_2 approaches 2. Moreover, taking the derivative of z_2,

$$\frac{d}{d\sigma}\left[\frac{\sigma}{2} + \frac{ln(1 + 2 * (e^{\sigma^2} - 1)^{1/2})}{\sigma}\right]$$
$$= \frac{1}{2} + \frac{2\sigma * e^{\sigma^2}}{e^{(\sigma^2-1)^{1/2}} + 2e^{\sigma^2} + 2} - \frac{ln(1 + 2(e^{\sigma^2} - 1)^{1/2})}{\sigma},$$

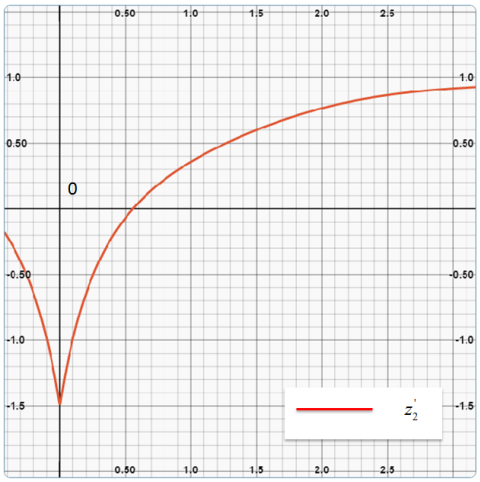


Figure 3.8: The derivative of z_2 demonstrates that its corresponding slope tends to be negative below $\sigma \approx 0.5$ indicating that z_2 decreases as σ initially moves away from 0.

we can determine the behavior of z_2 as it approaches 0. Plotting the derivative as shown in Figure 3.8, we observe that the slope remains negative until around $\sigma \approx 0.5$. Therefore, z_2 increase in value as it approaches 0 starting at $\sigma \approx 0.5$

Since, Equation 3.33 holds when $\sigma \geq 0.5$ due to $C * B$, it is sufficient to show that at $\sigma = 0$ that the following inequality holds,

$$1 \leq 2 * C + \frac{C}{e^{\alpha * 8} * e^{\beta * 2}} + \frac{(C+1)}{e^{\alpha * 8} * e^{\beta * 2}}.$$

By necessity, at $\sigma > 0$, the strict inequality of Equation 3.33 holds since B gets larger and A gets smaller. Substituting $z_1 = 0$ and $z_2 = 2$ back into B and A, respectively, we can express the inequality as

$$1 - 2 * C \leq \frac{C}{e^{\alpha*1} * e^{\beta*1}} + \frac{(C+1)}{e^{\alpha*1} * e^{\beta*1}}$$

$$0.0455 \leq \frac{0.47724}{e^{1.7585*24.4151}} + \frac{1.47724}{e^{1.7585*24.4151}}$$

$$0.0455 \leq 0.0455$$

Since the inequality holds, the proof is completed demonstrating that a log-normal distribution will always produce a corresponding normal distribution with an over estimated error region between the mean and two standard deviations greater than the mean on the normal distribution. □

This has negative effects when using parametric-based feature selection algorithms as we demonstrate using synthetic and real-world data. Selecting the most optimal subset S of features is an NP-hard problem with exponential number of feature subsets as potential solutions. This becomes intractable as the dimensionality of the dataset becomes extremely large. Parametric feature selection algorithms may not provide the most optimal subset S given the distribution and the error region that is subsequently formed when log-normal distributions are the true representative of positive class distribution. We, therefore, propose a non-parametric feature selection method based on the K-S test to overcome these challenges.

3.4.2 Non-parametric Feature Selection

In order to mitigate the parametric shortcomings during feature selection, we propose the Maximum Distance Minimum Error (MDME) method, which is based on the Kolmogrov-Smirnov (K-S) test; this test is a non-parametric statistical method that compares the cumulative distribution functions (CDF) of two sample to determine whether they are

statistically equivalent. The K-S test has previously proven its utility in HCS analysis of phenotype perturbation descriptors and immunofluorescence analysis of cells [34, 73, 50, 7]. Previous feature selection algorithms have also utilized the K-S test in removing redundant features [11, 86]. We demonstrate the ability of the K-S test to, instead, provide a robust score to gage the importance of each feature.

Quantifying the distance between the the respective CDFs of two distributions is accomplished using what's known as the D value.

$$D_i = \sup_{\mathbf{x}} |f_i^+(x) - f_i^-(x)|, \tag{3.34}$$

where the function $f(\cdot)$ is given as

$$f(x) = \frac{1}{n} \sum_{i=1}^{n} I_{[-\infty,x]}(\mathbf{x}_i) \tag{3.35}$$

and the indicator function $I(\cdot)$ is defined as

$$I_{[-\infty,x]}(\mathbf{x}_i) = \begin{cases} 1 & \text{if } \mathbf{x}_i \leq x \\ 0 & \text{if } \mathbf{x}_i > x. \end{cases} \tag{3.36}$$

Figure 3.9 demonstrates how the CDF of two sample distributions is utilized to obtain a quantifiable distance measurement D. The distance is tested to determine whether a null hypothesis of that the distributions come from the same distribution is rejected or accepted. The distance measurement, D, provides a non-parametric alternative to scoring features for importance and facilitate the selection of the optimal subset of features that can best distinguish between two classes.

In feature selection, we can allow the $D \in [0, 1]$ value to represents how distinct one class distribution is from the other where 0 implies the two distributions are identical and therefore inseparable and 1 implies optimal difference in population distributions and completely separable. If a feature $\mathbf{x}_i$ has a D value of 1, then the distribution of the classes is optimally separated. The D value can be thought of as essentially measuring the amount

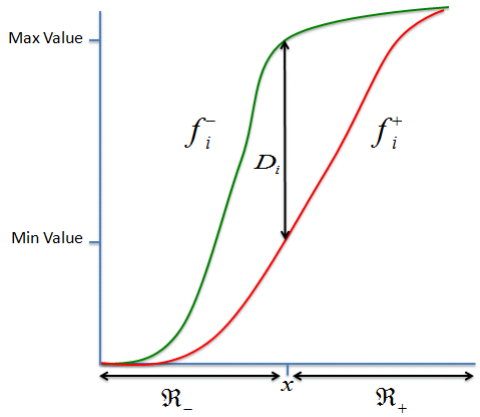


Figure 3.9: The D value provides a decision boundary that minimizes the error regions between two distributions.

of overlap between two classes; the closer to 1 the D value is, the less overlap exists. Using the D-value allows for incorporating decision theory to select those features where there is minimal overlap, and therefore minimum error regions, between the two classes.

The decision boundary is defined by the x value that produces the superum for D. Decision regions $\mathbb{R}$ are naturally derived using the decision boundary such that the error region for each class is minimized. For instance in Figure 3.9, the D_i value defines a maximum and minimum value. These values naturally reduce the error region between two classes for a single variable. In this case, the maximum value produces a decision boundary for the negative class and the min value produces a decision boundary for the positive class. Any value greater than the maximum value is considered an error region for the negative class and, as a corollary, any value less than the min value is considered an error region for the positive class. For each region, two bins are defined as follows

$$B^f_{\epsilon_+} = \forall(x^f_{i_-} \in \mathbb{R}^f_+)$$
$$B^f_{\epsilon_-} = \forall(x^f_{i_+} \in \mathbb{R}^f_-).$$

These bins identify inherent error that exists when the D value of feature f is used. Therefore, we propose a score that takes into account both the D value and the error regions created by the D value. Each feature is given a M_{score} based on the D value the K-S test and an error minimization function $E(\cdot)$.

$$M_{score}(X_f) = D_f - \frac{1}{|S|}\sum_{s \in S} E(X_f, X_s) \tag{3.37}$$

In many filter-based feature selection approaches, a common method of expanding the selected subset of features has been accomplished by either selecting the top scoring features or focusing on minimizing the amount or redundancy that exists within the selected subset using correlation measurements such Pearson's correlation or K-S test. We demonstrate how error minimization in conjunction with the D value produces a model that, instead, reduces residual error rather than redundancy. Thus, in selecting a subset of features, the goal is to reduce the area of the error region that is left uncovered by each of the subsequent features selected as the optimal subset.

The adding of a feature to the subset S is done by the function $E(\cdot)$, which attempts to explain the error left uncovered by features $s \in S$ while minimizing the amount of error introduced by the new feature.

$$\begin{aligned} E(X_f, X_s) = \alpha * (1 - J^+(f, s)) \\ + (1 - \alpha) * (1 - J^-(f, s)), \end{aligned} \tag{3.38}$$

where $J(\cdot)$ is

$$J^+(f, s) = \frac{|B^f_{\epsilon_+}| + |B^s_{\epsilon_+}| + 2 * |B^f_{\epsilon_+} \cap B^s_{\epsilon_+}|}{n^+} \tag{3.39}$$

$$J^-(f, s) = \frac{|B^f_{\epsilon_-}| + |B^s_{\epsilon_-}| + 2 * |B^f_{\epsilon_-} \cap B^s_{\epsilon_-}|}{n^-}, \tag{3.40}$$

where n^+ and n^- are the total number of data points in the positive and negative classes, respectively. The MDME method provides an additional parameter α that allows the user to tune the method based on the importance of error minimization in one class, proving useful when a user has *a priori* knowledge of class importance and data distribution. Since MDME uses CDF, it is less prone to outliers as the raw value does not influence the score

directly. In addition, normalizing the values is not necessary as the cumulative distribution function will not change when the data is normalized.

Algorithm 2 Maximum Distance Minimum Error

Input: $X \in \mathbb{R}^{nxd}$, $\alpha = 0.5$, k

Output: S

1: **for** $i = 1$ to d **do**

2: $\quad D(X_{:i}) = \sup_{\mathbf{x}} |f_i^+(x) - f_i^-(x)|$

3: **end for**

4: $\hat{D} = sort(D)$ in descending order

5: Add $\hat{D}(1) \rightarrow S$

6: **for** $j = 2$ to K **do**

7: $\quad$ **for** $m \in \{X \setminus S\}$ **do**

8: $\quad\quad M_{score}(X_{:m}) = D_{scores}(m) - \frac{1}{|S|} \sum_{s \in S} E(X_m, X_s)$

9: $\quad$ **end for**

10: $\quad max(M_{score}) \rightarrow S$

11: **end for**

12: **return** S

In order to determine the effectiveness of the proposed method, we analyzed synthetic datasets, well known feature selection datasets from ASU feature selection repository, and real world world high content screening datasets. Table 3.2 provides specific information of the sixteen different datasets that were analyzed. Sampling was determined based on the number of instances in the data. If there were less than 100 total instances in the ASU repository feature selection datasets, a leave one out (LOO) approach was implemented. Otherwise, 10-fold cross validation was implemented. For synthetic data and real world image-based HCS data, a bootstrapping approach was implemented since there are tens of thousands of data sets and and is more reflective of real world processes.

We implemented four different classifiers to determine how well the features that were

selected from each statistical algorithm performed.

- *Support Vector Machine (SVM)*: A linear support vector machine classifier was trained where the cost parameter C was varied from 0.5 to 8 by doubling the cost at each increment. A grid search approach was done to find the best C value for each feature selection method.

- *Gaussian Naive Bayes*: A Gaussian Naive Bayes classifier was trained on the data.

- *Random forest*: A random forest classifier was trained on the data where the number of trees T was varied from 10 to 50 with increments of 10. A grid search approach was done to find the best T value for each feature selection method.

- *Decision Tree*: A decision tree finds the best splits in data that reduce entropy.

Using Scikit-learn [71], each of the classifiers were trained using normalized data for the real-world and feature selections datasets and unnormalized data for the synthetic dataset.

Synthetic data was developed by randomly generating a set or $1,500$ features. The features representing the negative control class were generated by randomly selecting a variance value $\sigma_N^2 = [1, 10]$ and mean value $\mu_n = [10, 50]$. The positive control distribution was created to ensure a discriminative log-normal distribution F_{LN}^+ with its corresponding normal distribution F_N^+ significantly overlapping the negative control distribution for no more than 30 features. The rest of the features were randomly generated with the same mean and variance for both classes to represent noisy data. This process was accomplished 50 times for two different versions and the the maximum average accuracy was computed. Version 1 allowed the mean of the corresponding positive normal distribution to the log-distribution and negative distribution range to one standard deviation away from each other. Version 2 only allowed the mean from each distribution to be a half of a standard deviation away. Thus, the version 2 synthetic data represented a much nosier dataset than data 1.

There were 5 well known feature selection algorithms that were implemented using scikit-feature [51] in python. We chose four statistical parametric-based methods that assumed a

Gaussian distribution and one statistical non-parametric method based on Chi-square test. The Chi-square test is a powerful non-parametric tool for determining dependency between groups [60]. It has previously demonstrated its effectiveness in phenotype perturbation analysis [58] and selecting features for text categorization [99]. We briefly describe each of the 5 that were implemented.

- *maximum Relevance Minimum Redundancy (mRMR)* [72]: The mRMR algorithm is a well known and powerful method that attempts to select those methods that are most discriminative while minimizing the amount or redundancy.

- *F-score*[98]: The F-score was proposed by Sewall Wright in 1965 as a correlation coefficient in biological applications. The F-score can handle multiple classes.

- *T-score*[26]: The T-score is similar to the F-score, except that it can only handle binary classes when determining the correlation coefficients.

- *Fisher score*[31]: The Fisher Score selects a subset of features such that the sample variance within the same class is small while the variance of the samples from different classes are large.

- *Chi-Square*[54]: The Chi-square algorithm discretized numeric data while ascertaining which features best reduce error.

3.4.3 Results

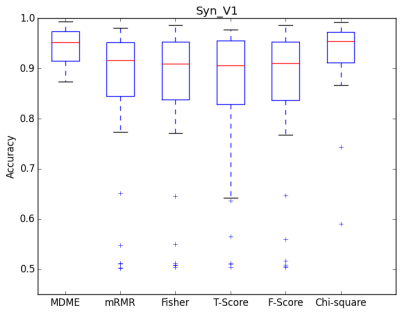
The number features was varied using $f = \{1, 5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$ to determine how well each feature selection method performed with each classifier previously described. We separate the synthetic dataset results from the other datasets because the synthetic datasets were specifically generated to demonstrate the shortcomings of the other statistical parametric-based features selection methods. Figures 3.10 and 3.11 provide accuracy information for the different feature selection methds using each of the classifiers

Table 3.2: Dataset information used in testing MDME.

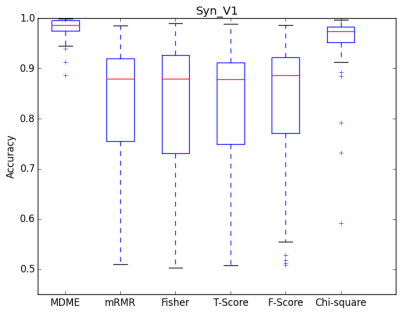
	# Instances	#Features	Type	Sampling	#Test	#Train
Syn_V1	4000	1500	Synthetic Data	Bootstrapping	3900	100
Syn_V2	4000	1500	Synthetic Data	Bootstrapping	3900	100
ALLAML	72	7129	Biological Data	LOO	1	71
arcene	200	10000	Mass Spectrometry	10-fold	20	180
gisette	7000	5000	Digit recognition	10-fold	700	6300
madelon	2600	500	Artificial	10-fold	260	2340
SMK_CAN_187	187	19993	Biological Data	10-fold	19	168
Prostate_GE	102	5966	Biological Data	10-fold	10	92
201100812 1 Well	$\approx 6874 \pm 541$	2530	HCS	Bootstrapping	$\approx 6211 \pm 597$	$\approx 663 \pm 167$
201100812 2 Well	$\approx 6956 \pm 505$	2530	HCS	Bootstrapping	$\approx 5567 \pm 459$	$\approx 1389 \pm 301$
201104270 1 Well	$\approx 2490 \pm 264$	2530	HCS	Bootstrapping	$\approx 2239 \pm 253$	$\approx 251 \pm 116$
201104270 2 Well	$\approx 2352 \pm 257$	2530	HCS	Bootstrapping	$\approx 1857 \pm 257$	$\approx 495 \pm 173$
201104288 1 Well	$\approx 1812 \pm 151$	2530	HCS	Bootstrapping	$\approx 1619 \pm 146$	$\approx 193 \pm 57$
201104288 2 Well	$\approx 1874 \pm 101$	2530	HCS	Bootstrapping	$\approx 1500 \pm 107$	$\approx 374 \pm 89$
201101095 1 Well	$\approx 3718 \pm 368$	2530	HCS	Bootstrapping	$\approx 3347 \pm 347$	$\approx 371 \pm 147$
201101095 2 Well	$\approx 3552 \pm 386$	2530	HCS	Bootstrapping	$\approx 2819 \pm 424$	$\approx 733 \pm 227$
201101097 1 Well	$\approx 3716 \pm 377$	2530	HCS	Bootstrapping	$\approx 3305 \pm 336$	$\approx 411 \pm 186$
201101097 2 Well	$\approx 3679 \pm 361$	2530	HCS	Bootstrapping	$\approx 2964 \pm 353$	$\approx 715 \pm 201$

previously described. Version 2 demonstrates much more variance corresponding with the noisier synthetic data. For both versions of the synthetic data, the proposed MDME method outperformed the other feature selection methods and in the case of version 1 had less variance in the accuracies. The complete set of results is provided in the Appendix at the end of the dissertation.

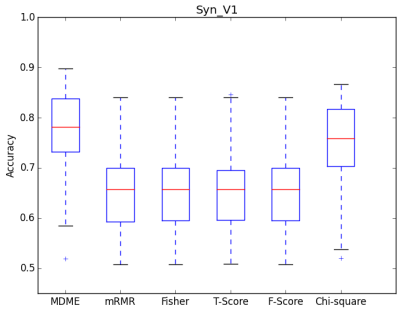
Table 3.3 provide information on performance of each feature selection algorithm using the different classifiers across the different datasets. The results demonstrate that the MDME method performed comparable to, and often times better than the other feature selection methods across the different datasets by different amounts. The MDME method performed consistently better than the other methods when using the HCS datasets consistent with log-normal distribution hypothesis of the data. Traditional parametric feature



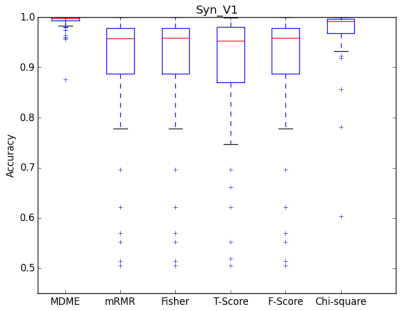
(a) Decision Tree.



(b) Random Forest



(c) Linear SVM.



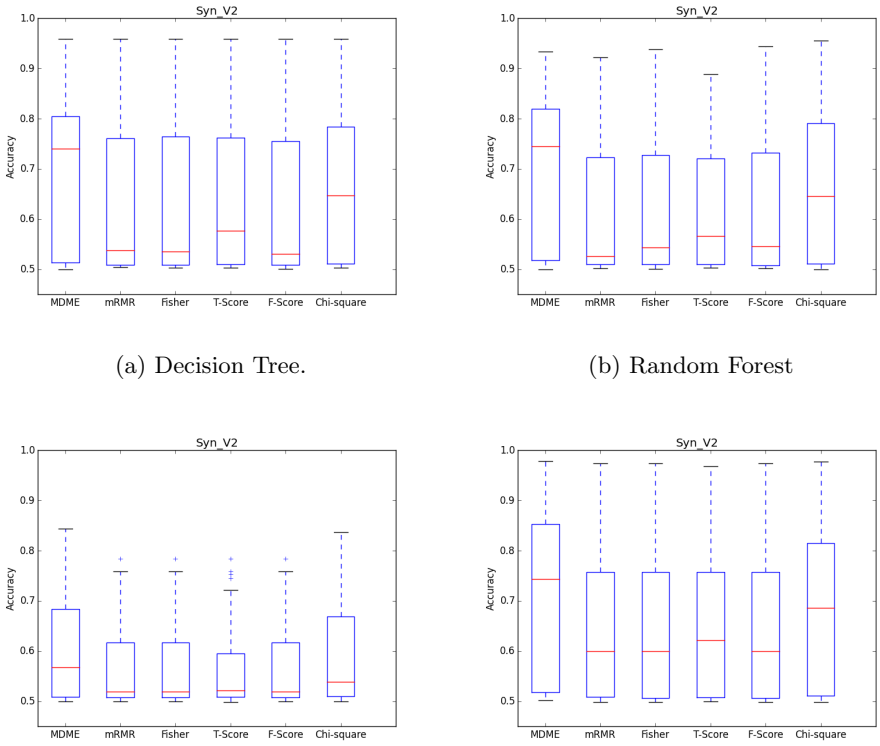
(d) Naive Bayes.

Figure 3.10: Version 1 of the synthetic data generated demonstrated that MDME and chi-squared were both better able to handle datasets where one class had a log-normal distribution.

Table 3.3: Number of datasets where maximum accuracy is achieved.

	MDME	MRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	10/16	6/16	6/16	5/16	6/16	1/16
Gaussian NB	13/16	2/16	2/16	3/16	2/16	3/16
Random Forest	14/16	0/16	0/16	3/16	0/16	0/16
Decision Tree	10/16	1/16	1/16	5/16	0/16	0/16

selection methods rely on normal distribution assumptions to identify the most relevant features capable of distinguishing between different classes. Unfortunately, many real world datasets such as image-based HCS data often tends to follow a log-normal distribution instead. We demonstrated how parametric-based feature selections methods do not perform



(a) Decision Tree.

(b) Random Forest

(c) Linear SVM.

(d) Naive Bayes.

Figure 3.11: Version 2 of the synthetic data was much noisier and demonstrated that parametric-based models performed below non-parametric methods.

well when one of the classes of a given dataset has a log-normal distribution. Synthetic datasets demonstrated how well known parametric-based feature selection methods identified sub-optimal feature sets. We proposed a novel non-parametric approach using the well known K-S test. This method has several advantages including being less prone to outlier influence and not requiring normalizing the data. Across 16 data sets, the proposed MDME method performed comparable to and often times better than the parametric-based feature selection methods with varying number of training instances. It demonstrated the ability to consistently perform better than the other algorithms on real world HCS data. The MDME can, therefore, be considered a viable option for feature selection on real world data where a normal distribution cannot be assumed.

3.4.4 Summary

Traditional parametric feature selection methods rely on normal distribution assumptions to identify the most relevant features capable of distinguishing between different classes. Unfortunately, many real world datasets such as image-based HCS data often tends to follow a log-normal distribution instead. We demonstrated how parametric-based feature selections methods do not perform well when one of the classes of a given dataset has a log-normal distribution. Synthetic datasets demonstrated how well known parametric-based feature selection methods identified sub-optimal feature sets. We proposed a novel non-parametric approach using the well known K-S test. This method has several advantages including being less prone to outlier influence and not requiring normalizing the data. Across 16 data sets, the proposed MDME method performed comparable to and often times better than the parametric-based feature selection methods with varying number of training instances. It demonstrated the ability to consistently perform better than the other algorithms on real world HCS data. The MDME can, therefore, be considered a viable option for feature selection on real world data where a normal distribution cannot be assumed.

Chapter 4

COMPOUND ANALYTICS

With the development of automated microscopy and image analysis systems, the process of identifying novel therapeutic drugs generates an immense amount of data - easily reaching terabytes of information, as previously noted. Despite increasing the amount of data generated during drug discovery endeavors, traditional analysis methods have not increased the overall success rate. The *Compound Analytics* section introduces a novel method based on parallelized cellomics that uses a small number of individual cells in high dimensional space to analyze interactions between cells, bacteria, and chemical compounds. The novel method demonstrates the capacity to distinguish between bacterially infected and uninfected control data using a small number of cells at comparable accuracy levels as using large control datasets, reducing the amount of data needed for quality control. Results further indicate that the proposed method can identify chemical compounds that inhibit bacterial infection using a fraction of the control data generated, allowing for more in depth interrogation of chemical compounds.

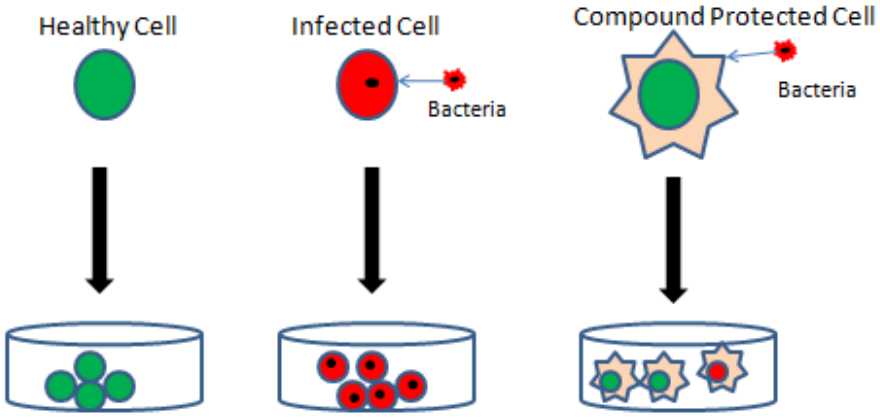


Figure 4.1: Active chemical compounds will preserve a healthy cell profile while protecting against infection.

The goal is to integrate multivariate image-based HCS data with compound hit selection to competitively identify effective chemical compounds that inhibit bacterial infection. A massively parallel method is proposed that uses high dimensional image-based HCS data to identify compounds that inhibit infection while preserving cellular health. Identifying compounds that inhibit bacterial infection in human cells provides unique biological properties to exploit. For instance, compounds that inhibit bacterial infection of a healthy cell should have minimal impact on that cell such that the phenotype signature or profile of cells treated with a chemical compound should closely resemble the profile of healthy, untreated cells. Motivated by Loo *et al.* where a support vector machine (SVM) classifier was used to determine a compound profile [56], a "healthy cell" profile defined by a random forest classifier is analyzed. This simplifies the compound hit selection process significantly by allowing a comparison of healthy, untreated cells to treated cells exposed to bacteria. Additionally, unless using undifferentiated cells, the majority of healthy cells should demonstrate a similar profile even if the cell cycle has not been arrested. This implies that the number of healthy, untreated cells needed to create a reliable profile can be small, reducing the need for large control data sets. Figure 4.1 demonstrates the intuition behind these biological properties. The proposed novel method takes advantage of these assumptions when identifying chemical compounds that inhibit bacterial infection. Contributions of the proposed method are summarized below:

- Demonstrating the ability to differentiate between control infected and uninfected data with high accuracy using high dimensional cell measurements from HCS bioassays. In section 4.4.5, the proposed method demonstrate that even with a low Z' factor given to a bioassay plate using traditional measurements, it achieves high accuracy separating the control data.

- Reducing control data needed to measure differentiation has minimal impact on accuracy. Section 4.4.6 provides results that demonstrate that the control data can be reduced to a single positive and negative control well using a hundred individual cells

from each while still maintaining high accuracy.

- Compound hit selection that inhibits bacterial infection can be accomplished using a small amount of control data. We demonstrate in section 4.4.7 that known, active compounds can be identified using a fraction of the data traditionally required.

These contributions are reliant upon the quality of automated image analytics to properly segment individual cell images and to subsequently produce high dimensional measurements for each cell. Although the proposed method demonstrates the ability to handle noisy data, improving the quality of image analytics is outside the scope of this paper.

4.1 Problem Statement

Let $C = \{c_1, c_2, ...c_n\}$ be the n compounds to be interrogated in an HTS campaign. Let $T = \{t_1, t_2, ...t_n\}$ be the biological targets of interest such as protein or, in this specific case, bacteria tested against the respective n compounds where, generally, $t_1 = t_2... = t_n$. Let $X = \{B_+, B_-, B_1, B_2, ..., B_n\}$ be the complete biological entity data set, which in this case are healthy cells. $B_i = \{b_{i1}, b_{i2}, ..., b_{ik}\}$ are the k cells treated with c_i where t_i is the biological target of interest. B_+ and B_- are the control data sets with biological targets of interest introduced in one and absent in the other, respectively. A healthy cell is defined as $b_{ik} \in \mathbb{R}^d$ where d is the number of features measured for each cell in phenotype analysis.

A perturbation function $G(\cdot)$ is defined that assesses the interaction between a healthy cell, a chemical compound, and a biological target.

$$G(b_{ij}, c_i, t_i) = b_{ij} + H(c_i, t_i), \tag{4.1}$$

where $H(\cdot)$ is compound-target interaction function that introduces the perturbation that a compound and target have on a healthy cell. An optimal chemical compound would cancel out any bacterial effect while contributing no effect of its own on a cell. However, both the compound and bacteria generally introduce some perturbation or error $\nu_i \in \mathbb{R}^d$ to the

healthy cell such that

$$H(c_i, t_i) = \begin{cases} 0, \textit{Optimal} \\ \nu_i, \textit{Otherwise.} \end{cases} \tag{4.2}$$

A compound activity function $A_{eff}(\cdot)$ provides a protection score defined as

$$A_{eff}(B_i, c_i, t_i) = \frac{1}{K} \sum_{k=1}^{K} ||G(b_{ik}, c_i, t_i) - b^*||_2^2, \tag{4.3}$$

where $b^* \in \mathbb{R}^d$ is an optimal healthy cell profile. This can be rewritten as

$$A_{eff}(B_i, c_i, t_i) = \frac{1}{K} \sum_{k=1}^{K} ||\nu_i||_2^2, \tag{4.4}$$

where the compound and bacterial effects on healthy cells is measured using Euclidean distance via the sum of the $l2$-norm squared. The lower the activity protection score, the less error is introduced to a healthy cell indicating more protection from the compound. A compound activity indicator function $f(\cdot)$ is defined as

$$f(c_i, \varepsilon, CE) = \begin{cases} 1, CE \leq \varepsilon, \\ 0, \textit{Otherwise} \end{cases}, \tag{4.5}$$

where CE is a "compound effectiveness" score defined by an estimated $A_{eff}(\cdot)$ and the probability of finding that level of protection randomly. The function returns 1 if a compound is active and 0 otherwise. An active compound will sustain a healthy cell profile for treated cells to not exceed an acceptable user-defined ε threshold level. We propose a method that limits the amount of data in C, T, and B required to obtain an accurate $A_{eff}(\cdot)$ for $f(\cdot)$ to determine compound effectiveness. The method is composed of two primary components: (1) Phenotypic Analyis and (3) Compound Analysis. The *Phenotypic Analysis* component consists of feature measurements and feature selection while the *Compound Analysis* consists of quality control and compound hit selection. We examine implementation of each of these components and how to overcome big data challenges by exploiting parallel processing using HPC systems.

4.1.1 *Parallel Processing in HPC Environment*

To effectively use an HPC system, the proposed method was designed with the HPC architecture in mind. Modern HPC systems are built up from layers of parallelization, shown in Figure 4.2. The lowest level of parallelization is the vector unit, which can evaluate multiple floating point operations (FLOP) concurrently, as long as the operation on each element of the vector is the same. A core is able to execute a single thread at a time and can have multiple vector units within it. Multiple cores are present on each individual processor chip, often referred to as a socket, after the location where it is placed within the motherboard. There can be multiple sockets on a single node. As illustrated in Figure 4.2, each node has a memory bank that all of the sockets are able to access. A HPC system can then be built from multiple nodes that are connected with a network.

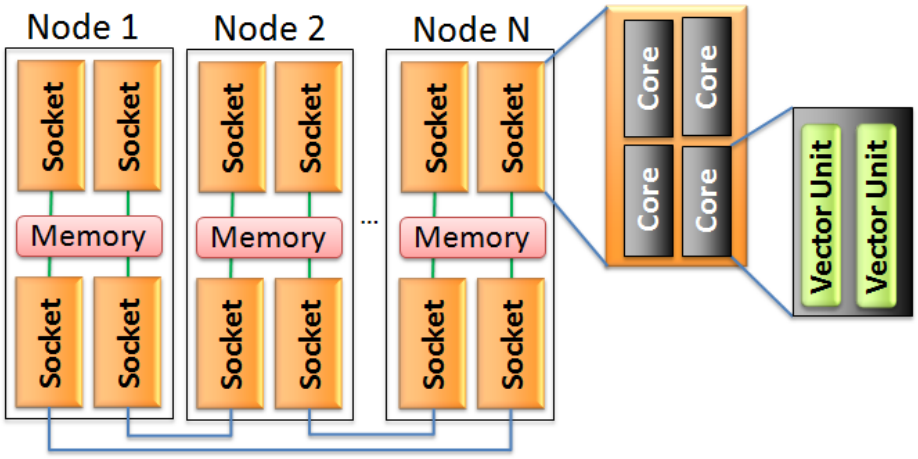


Figure 4.2: Simplified diagram of modern computer. A network connects multiple nodes, and each node has shared memory and multiple sockets where a chip can be inserted. Each socket has multiple cores, and each core has multiple vector units.

Software implementation of the proposed method was also designed to exploit parallel processing. A given sequence of instructions is a thread and can run on a single core. It is possible for a single process to have multiple threads, and thus run on multiple cores. However, the memory associated with a process must occupy a single virtual address space specified by the operating system, and thus must be located on a single node. Thus, threads in a single process may share memory between them, but may not (in general) share memory

with other nodes. There are multiple paradigms for writing multi-threaded programs, such as pThreads, C++11 threads, Java threads, and OpenMP. For the implemented code, all threaded programs have been written using OpenMP.

The HPC industry has long coalesced around the message passing interface (MPI) standard for communication between processes. An MPI program executes multiple processes, referred to as ranks, to complete its computation. Each rank has its own address space and, in general, that address space is not shared between ranks. MPI ranks that exist on different nodes can send messages to each other over a network. MPI ranks may also coexist on the same node. The ability to expand beyond a single node is beneficial for both increasing available cores and available memory. The caveat is that communication over a network is much slower than communication through system memory.

There is a general HPC paradigm referred to as "MPI + X" where X can be pThreads, OpenMP, accelerator offload (e.g. Intel Xeon Phi), or other methods to try to get performance out of a system. In MPI + X, there are multiple MPI ranks in a program, but the ranks may then be broken up into threads or other subunits. The threads for a single rank may share memory amongst themselves and use MPI messages to communicate with other groups of threads. A common approach is to use 1 MPI rank per node (or per socket) and then use OpenMP to have that rank use all of the cores available to it.

The ability to exploit HPC in a given problem is reliant upon how well both functional domain decomposition (concurrent analysis of different algorithms) and data domain decomposition (concurrent analysis of different cells, wells, etc) can be achieved. The ease of exploiting parallel processing in this case is largely reliant on the computational independence of individual or small sets of cell, feature, and compound measurements. The more measurement inter-dependency, the more difficult it is to implement parallel processing due to the increased communication burden. In some situations where there is no required communication, completely separate processes can be launched to make use of the available computational resources. In these situations MPI is not needed at all and OpenMP may

still be used to parallelize common structures such as for loops. Discussion follows as to how to implement parallel processing in the *Phenotypic Analysis* and *Compound Analysis* components and when to use OpenMP and MPI.

4.2 From Phenotypes To Features

The *Phenotypic Analtyics* component is responsible for identifying the most pertinent cellular phenotype measurements capable of distinguishing between infected cells and healthy cells. The proposed approach breaks from traditional methods of measuring different phenotypes perturbations of a cell and selecting the most pertinent measurements. Although the proposed approach tends to be more sophisticated, it is also more computationally expensive since it operates in a high dimension feature space. Distributed processing in a HPC environment allows for rapid execution of the proposed approach and completely mitigates many of the prohibitive time constraint issues that would be faced with individual cells in a high dimensional feature space.

4.2.1 Feature Measurements

The standard method of identifying pertinent phenotype measurements was based on *a priori* biological knowledge. This approach required that the biologist determine which phenotype measurements were capable of detecting the potential changes that would occur between infected and healthy cells. Two ubiquitous phenotype measurements using image-based HCS were cell density and cell infection index, as measured by green florescent probes (GFP) attached or genetically engineered into bacteria. Unfortunately, this approach is limited to what is previously known by biologists pertaining to cell-bacteria interaction and specific florescent signals that may be noisy.

In contrast, no *a priori* assumption of knowledge of phenotype changes is made in this work, instead producing $\approx 11,000$ different image measurements for each cell using a combination of public and proprietary algorithms. Approximately 3,000 of these algorithms were obtained from WNDCharm [66]. A large portion of the algorithms have been described

previously in [3] as well. These cell features are derived across the multiplexed channels defined by the Hoescht (Figure 1.2a), GFP-labeled bacteria (Figure 1.2c), and phase contrast (Figure 1.2b) images.

Depending on the size of the data, hundreds of thousands, if not millions, of individual cells will require that $\approx 11,000$ different image measurements be taken for each cell. Parallel processing in a distributed environment can reduce the amount of real time necessary to generate the measurements making, it necessary for high-dimensional big data analysis. In this particular case, the vast majority of the algorithms generate small sets of feature measurements and are computationally independent allowing for functional domain decomposition. Additionally, the vast majority of algorithms are performed on one cell at a time and therefore allow for data domain decomposition. The combination of two easily decomposed domains makes this problem ideally suited for massive parallelization, using a single stand-alone OpenMP process for each site. A site is a group of cells captured in a set of images that is assigned to a node. Within each site, a list of $< algorithm, cell >$ combinations is created and OpenMP is used to evaluate each element of that list in parallel. Figure 4.3 demonstrates how parallel processing is implemented in a HPC cluster in a distributed manner. Depending on the number of cores available and the number of cells that must be analyzed, the real computational time needed to complete this task can be greatly reduced, often by orders of magnitude.

4.2.2 Feature Selection

Since high dimensional cell data is generated for each cell, an additional preprocessing step is required to remove redundant and irrelevant features. The most discriminative features of healthy cells are selected using Minimum Redundancy Maximum Relevance (mRMR) feature selection algorithm [72]. This is a powerful method that not only selects the most discriminative features, but also mitigates the redundancy between the features using correlation between different features versus the correlation of a feature and the different classes [72, 109]. In establishing both maximum relevance and minimum redundancy,

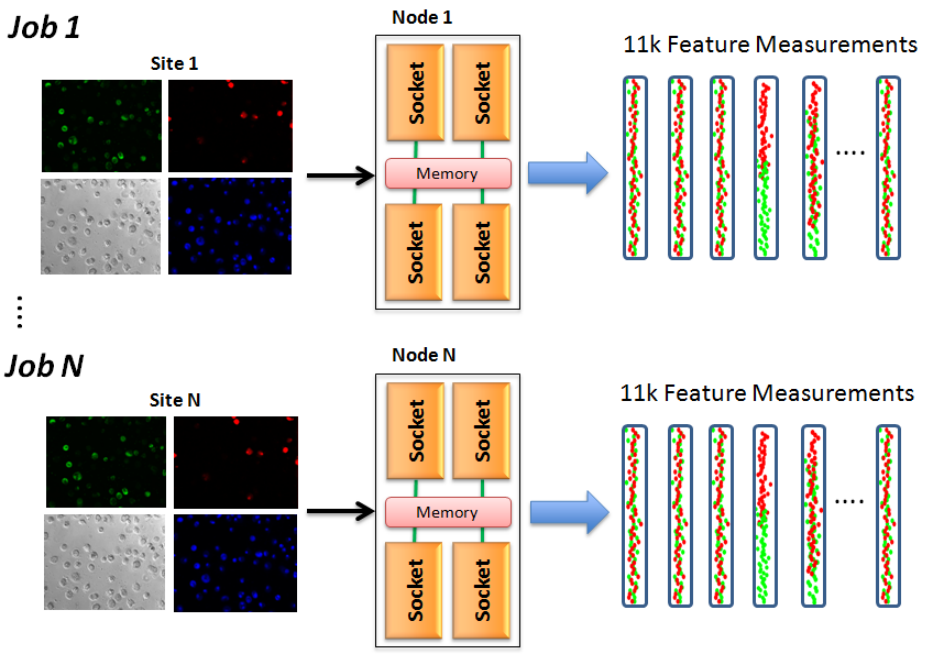


Figure 4.3: Using OpenMP, a node is assigned to a group of cells obtaining feature measurements of both compound-treated and control cells. This parallelization is critical due to the extremely large quantities of cells that must be analyzed.

mutual information is used for categorical data.

$$min(W_I, W_I) = \frac{1}{|S|^2} \sum_{i,j \in S} I(i,j) \tag{4.6}$$

$$max(V_I, V_I) = \frac{1}{|S|} \sum_{i \in S} I(h,i), \tag{4.7}$$

where h is class of interest for the i^{th} feature and S is the set of features that have already been selected. For continuous data, Pearson's correlation is used for feature-feature correlation and the F-test is used for feature-class correlation. The use of the F-test provides the basis for the following proof.

Theorem 4 (mRMR-Z'-factor relationship). *Let f_z^* be the feature with the highest Z'-factor in the complete feature space D, then $f_z^* \in S$ where S is the mRMR reduced feature space.*

Proof. Contradiction- The first feature f_i that is selected by the mRMR algorithm has the

highest F-test value defined as

$$F(f_i, K) = \frac{\sum_{k=1}^{K} n_k(\mu_{ki} - \mu_i)^2}{(K-1)} \cdot \frac{n-K}{\sum_{k=1}^{2}(n_k - 1)\sigma_{ki}^2}, \tag{4.8}$$

where K is number of classes and n is the total number of data points. When $K = 2$, the F-test can be rewritten as

$$\begin{aligned} F(f_i, k) &= \frac{(n_1(B(\mu_{2i} - \mu_{1i}))^2 + n_2(A(\mu_{2i} - \mu_{1i}))^2)(n-2)}{(n_1 - 1)\sigma_{1i}^2 + (n_2 - 1)\sigma_{2i}^2} \\ s.t.\ A &= \frac{n_1}{n_1 + n_2} \\ B &= \frac{n_2}{n_1 + n_2}. \end{aligned} \tag{4.9}$$

A, B, n_1, n_2, and n are all constant values shared between features. Let us assume that there is another feature f_j that has a higher Z' factor than f_i, $Z_{f_j} > Z_{f_i}$, such that

$$\begin{aligned} Z_{f_j} &= 1 - \frac{3\sigma_{1j} + 3\sigma_{2j}}{|\mu_{1j} - \mu_{2j}|} \\ Z_{f_i} &= 1 - \frac{3\sigma_{1i} + 3\sigma_{2i}}{|\mu_{1i} - \mu_{2i}|}. \end{aligned}$$

The inequality can be written as

$$\begin{aligned} \frac{|\mu_{1j} - \mu_{2j}|}{|\mu_{1i} - \mu_{2i}|} &> \frac{\sigma_{1j} + \sigma_{2j}}{\sigma_{1i} + \sigma_{2i}} \\ s.t.\ \sigma_{1j} &\leq \sigma_{1i} \\ \sigma_{2j} &\leq \sigma_{2i} \\ |\mu_{1j} - \mu_{2j}| &> |\mu_{1i} - \mu_{2i}| \end{aligned} \tag{4.10}$$

with the given constraints to ensure that $Z_{f_j} > Z_{f_i}$. The absolute value of the mean difference $|\mu_{ij} - \mu_{ij}|$ can be used in Equation 4.9 without changing the result. Since the only non-constant change to f_i and f_j the squaring of the σ and μ values for each class then

$$\frac{|\mu_{1j} - \mu_{2j}|}{|\mu_{1i} - \mu_{2i}|} = \sqrt{\frac{(n_1(B(|\mu_{2j} - \mu_{1j}|))^2 + n_2(A(|\mu_{2j} - \mu_{1j}|))^2)}{(n_1(B(|\mu_{2i} - \mu_{1i}|))^2 + n_2(A(|\mu_{2i} - \mu_{1i}|))^2)}}$$

and

$$\frac{|\mu_{1j} - \mu_{2j}|}{|\mu_{1i} - \mu_{2i}|} > \frac{\sqrt{\sigma_{1j}^2 + \sigma_{2j}^2}}{\sqrt{\sigma_{1i}^2 + \sigma_{2i}^2}}.$$

Changes to the mean and standard deviation of f_j and f_i do not change the inequality. This implies that if $Z_{f_j} > Z_{f_i}$ then $F(f_j, k) > F(f_i, k)$. However, this is a contradiction since f_i has the highest F-test value. Therefore, f_i must also have the highest Z' factor. □

The proof demonstrates the ability of the mRMR method to initially identify the feature with the highest Z' factor on a balanced dataset with the given constraints. Subsequent features added to the subset provide more explanatory power. If the constraints are removed, the mRMR method chooses the features with the best strictly standardized mean difference (SSMD) [108]. This QC metric has demonstrated to be more robust and statistically rigorous in determining the quality of an RNAi-based assay than the Z-score QC metric. Thus, using mRMR, a properly tuned classification algorithm can perform as well as or better than using a single feature with optimal Z'-factor in distinguishing control data. Moreover, using discrete data demonstrated better results, in part, due to the reduction of noise inherent in continuous data [72]. In order to fully maximize mRMR, the bioassay data is discretized using the minimum description language program (MDLP) algorithm [33].

The mRMR feature selection algorithm is well designed for big data analysis given its relatively low computational complexity for identifying a relevant set of descriptive features. Its greedy search algorithm allows for implementation in a parallel processing environment, demonstrating the ability to reduce real time spent on computation from hours to seconds depending on the size of the data. Both main components of the mRMR algorithm as shown in Equations 4.6 and 4.7 can be parallelized. Figure 4.4 demonstrates how parallel processing is implemented in computing the F-test and Pearson's correlation for a feature set and class label. In the case of the F-test, data domain decomposition can be easily achieved as each feature is tested independently against the class label. For the Pearsons correlation, the data domain decomposition is that each calculation is based on a pair of

features and each pair is computed independently.

In situations such as the computing of F-test and Pearsons coefficient, OpenMP works well by using shared memory to hold the class labels and all of the feature measurements. Since the feature measurements are read only at this point, shared memory can safely be used to avoid unnecessary memory copies. A list of features for the F-test, or pairs of features for the Pearsons coefficient, to evaluate is created and then OpenMP is used to evaluate that list in parallel.

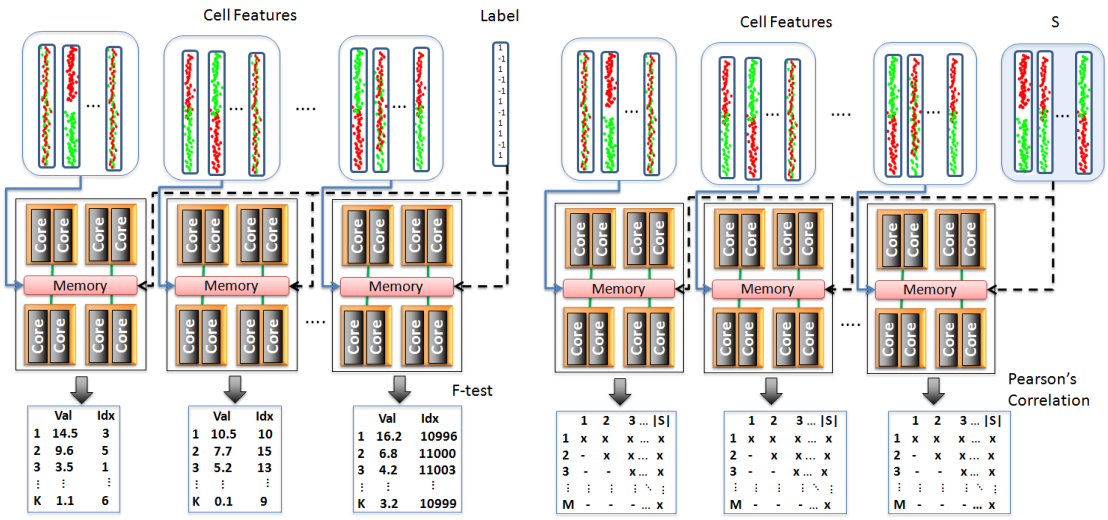
Additionally, the MPI+OpenMP architecture works well here if it is found that a single node is too slow. The same list that is generated for the OpenMP method is created, but it is now distributed amongst the different ranks. Each rank uses OpenMP to parallize evaluation of a smaller set of features, and MPI is used to gather results at the end. For the Pearson's coefficient, compared to the F-test, the calculations lead to far more occurances of the same data being read into memory on multiple ranks. Intelligent task distribution algorithms can reduce, but not eliminate, this extraneous memory usage, though this would still show a significant benefit because of the additional cores in use.

4.3 Compound Analytics

Compound inhibition of bacterial infection intuitively leads to a simple phenotype profile, which we exploit to reduce the number data points needed for quality control, to train a classifier, and, ultimately, identify active compounds. We discuss briefly a traditional method of measuring quality control using control data of a plate and how classification analysis is capable of fulfilling the same function with significantly less control data needed. We also propose a simple efficacy measurement that provides a value describing the activity of a compound, eliminating the reliance on traditional single readout analysis.

4.3.1 Quality Control

This single readout analysis was dependent on the quality of the readout to determine compound activity. This lead to the development of Z'-factor score used as a quality control



(a) Parallelized F-test computation. (b) Parallelized Pearson's correlation.

Figure 4.4: The mRMR algorithm has two primary components that are ideally suited for parallel processing using using MPI + OpenMP and capable of reducing real time for computation. (a) Computation of the F-test in the mRMR algorithms requires the class label with each of the K features. (b) Computing the Pearson's correlation can also be parallelized since previously selected subset of features S is used with each remaining M features not yet selected.

(QC) measure to identify the dynamic range of a single readout [106]. The dynamic range provides information on the extent or the separation between measured activity and inactivity of control data. The Z'-factor uses the mean and variance of each control population to determine the control separation. Subsequent compound analysis will not occur until an acceptable Z'-factor has been achieved ensuring reliable quality.

Using multivariate image-based HCS, the quality of data is no longer reliant on a single readout. We demonstrate that classification analysis is much more powerful at discerning plate quality using control data. The selection of an active compound C is reliant upon the accuracy of the interaction function I_a given in Equation 4.1. In order to properly asses a compound's activity, a threshold value needs to be specified on the acceptable amount of change introduced by compound and target into I_a to change the status of a healthy cell. We propose that allowing a classifier to determine the acceptable threshold will produce a powerful healthy cell profile.

Since we assume no *a priori* knowledge of the effects of bacteria on healthy cells, we

approximate all possible phenotype perturbations as measured by feature changes caused by the target bacteria as $\nu_T \in \mathbb{R}^d$. We let $E_{b_{+j},t_i}[H(b_{+j},t_i)] = \frac{1}{N_i}\sum_{j=1}^{N_i} H(b_{+j},t_+)$ be the expected feature measurements across all N_i control infected cells. We use $\hat{b}$ to approximate the original b_{+j} healthy cell profile. The average negative change to a healthy cell profile by a target bacteria is, therefore, given by

$$\nu_{Tave} = E_{b_{+j},t_+}[H(b_{+j},t_+)] - \hat{b}. \tag{4.11}$$

We can subsequently approximate the activity of a compound by

$$\hat{A}_{eff}(B_i,c_i,t_i) = \frac{1}{K}\sum_{k=1}^{K} ||l(b_{ik},c_i,t_i) * \nu_{Tave}||_2^2, \tag{4.12}$$

where $l(\cdot)$ is a cell classifier trained on control data and the two possible class values returned are defined as

$$0 : Healthy$$
$$1 : Infected.$$

The approximate healthy cell profile contains a certain amount of noise $\hat{b} - b_{ij} = \epsilon_b$. The larger ϵ_b is, the more it affects A_{eff}. Therefore, a classifier that is robust to noise and variance is ideal for determining $\hat{b}$. A random forest classifier is well suited for image-based HCS data given its computational efficiency and its robustness to noise. A random forest classifier is a collection of tree-structured classifiers $\{l(x,\theta_r), r = 1,...R)$, where R is the number of trees and $\{\theta_r\}$ is a random vector [13]. The injected randomness of $\{\theta_k\}$ allows the classifier to be more robust to noise and variance than other ensemble methods when measuring compound activity. The approximated activity using random forest classifier is given by

$$\hat{A}_{eff}(B_i,c_i,t_i) = \frac{1}{K}\sum_{k=1}^{K} ||avrI(l(b_{ik},c_i,t_i,\theta_r)) * \nu_{Tave}||_2^2, \tag{4.13}$$

where $avrI(\cdot)$ is the average prediction across R trees using indicator function $I(\cdot)$ to return majority class prediction. For simplicity, we let $avrI(l(b_k,c_i,t_i,\theta_r)) = avrI(l(b_k))$. It can

be further reduced to

$$\begin{aligned}\hat{A}_{eff}(B_i, c_i, t_i) &= \frac{1}{K}\sum_{k=1}^{K} ||avrI(l(b_{ik})) * \nu_{Tave}||_2^2 \\ &= \frac{1}{K}\sum_{k=1}^{K} avrI(l(b_{ik})) * (\nu_{Tave})^T(\nu_{Tave}) \\ &= \frac{(\nu_{Tave})^T(\nu_{Tave})}{K}\sum_{k=1}^{K} avrI(l(b_{ik}))\end{aligned}$$

Since $(\nu_{Tave})^T(\nu_{Tave})$ is constant across all compounds, it can be dropped so that the effectiveness of a compound is based on the effectiveness of the random forest classifier to properly identify healthy and infected cells.

$$\hat{A}_{eff}(B_i, c_i, t_i) = \frac{1}{K}\sum_{k=1}^{K} avrI(l(b_{ik})) \tag{4.14}$$

This allows for the rapid classification of healthy cells treated with a specific compound. An activity score can then be computed for each compound.

Algorithm 3 Compound Activity

Input: $X = \{B_1, B_2, ...B_n, B_+, B_-\}$, $C = \{c_1, c_2, ...c_n\}$, $T = \{t_1, t_2, ...t_n\}$

Output: AC

$b_{ik} \in X$ *is initialized with* 11k *features;* $AC = [\,]$

1: **for** $i = 1$ to n **do**

2: $B_{N_+}, B_{N_-}, B_{N_i} = Normalize(B_+, B_-, B_i)$

3: $\tilde{B}_{N_+}, \tilde{B}_{N_-} = features(B_{N_+}, B_{N_-})$: mRMR

4: $l = train(\tilde{B}_{N_+}, \tilde{B}_{N_-})$: Random forest classifier

5: Compute: $AC(i) = \hat{A}_{eff}(\tilde{B}_{N_i}, c_i, t_i)$

6: **end for**

7: **return** AC

Given that the computational complexity is approximately $O(M*(m*n*log_e(n)))$ where M is the total number of trees, m is the number of features, and n is the number of data points, the initial training of a random forest classifier is relatively fast. Moreover, once

the random forest algorithm has been trained, classification of individual cells is typically computationally inexpensive.

The random forest algorithm, as demonstrated, is most effective on a specific plate when using matching control data. Training a random forest classifier for each plate can be done in a HPC environment using distributed processing, exploiting data domain decomposition because each plate is computationally independent. MPI once again is not necessary as no communication is necessary between plates. OpenMP can be used by iterating over a given list of trees. Figure 4.5 demonstrates how plate analysis can be parallelized for training a classifier and, subsequently, identifying active chemical compounds. Depending on the number of plates M that are being analyzed, the parallelized method can save a significant amount of real time in hit selection analysis. This differs from traditional analysis that relied on combining compound data across different plates using single variate analysis. The more nuanced approach of splitting the analysis at the plate level is possible due to the robustness of the proposed method with plate level noise.

4.3.2 Hit Selection

Traditional single readout analysis has used two prominent methods called the "top K" and "outliers" method for hit selection. The top K method simply ranks all compounds based on desired single readout results and chooses the first K compounds. The outliers method selects those compounds that have desired activity levels two standard deviations away from the activity level of all other compounds tested. Since both of these methods rely on a relative ranking of compounds, they are susceptible to two primary shortcomings: (1) Identifying inactive compounds in a set of other inactive compounds. (2) Miss relevant compounds in a set of active compounds.

Using $\hat{A}_{eff}$ in conjunction with $f(c_i, \varepsilon, CE)$, a simple compound effectiveness score is derived to replace the single readout methods, mitigating their shortcomings.

$$CE = \alpha * P + (1 - \alpha) * U, \tag{4.15}$$

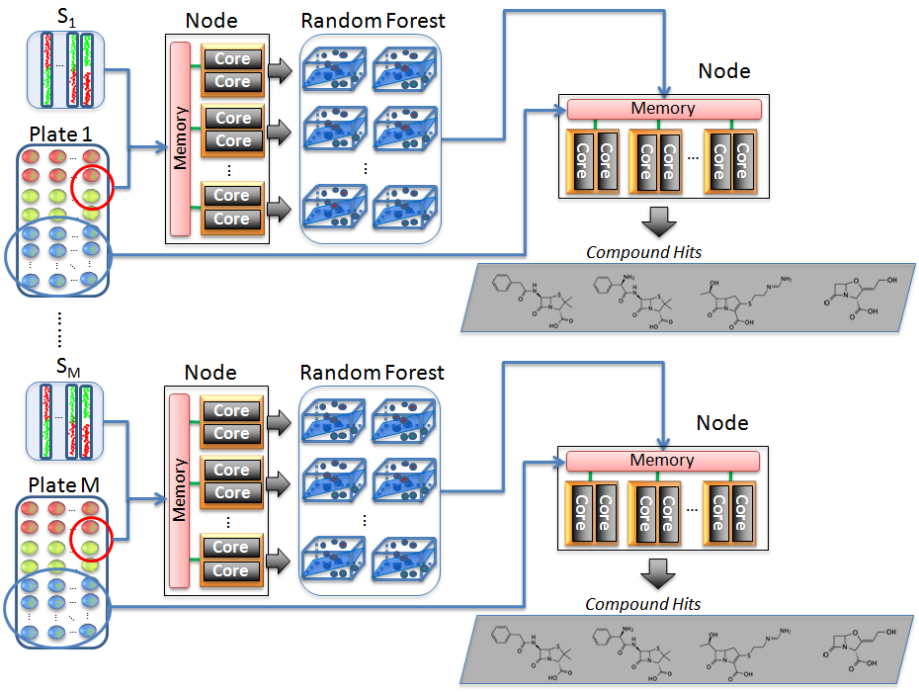


Figure 4.5: Using OpenMP allows compound analysis to be done at the plate level in a distributed environment for training a random forest classifier and subsequently using the classifier in hit selection analysis.

where P is the protection score and U is the uniqueness score assigned to a compound. The α parameter allows for the weighting of the importance of uniqueness of activity versus protection.

The protection score is simply

$$P = 1 - \hat{A}_{eff}. \tag{4.16}$$

If a compound has a low number of cells classified as infected than that compound is demonstrating high protection.

The uniqueness score is the probability of finding a compound's protection activity by random chance on a given bioassay plate. The probability of finding k number of uninfected cells in a compound well C_i of interest is represented by the p-value P_{C_i} of the hypergeometric distribution

$$P_{C_i}(X = k) = \frac{\binom{K}{k}\binom{N-K}{n-k}}{\binom{N}{n}}, \tag{4.17}$$

where K is the number of uninfected cells across the compound wells in a bioassay plate, N is the total number treated cells across the bioassay plate, and n is the number of cells treated with compound C_i. The hypergeometric p-values of some compounds, especially the most effective, tend to get extremely small. Therefore, the uniqueness score is given in a normalized log-based score.

$$U = \frac{-log(P_{C_i})}{max(-log(P_C))}. \tag{4.18}$$

If there are many compounds that have high efficacy in a plate then the uniqueness score will be lower.

Algorithm 4 Cell analytics compound hit selection

Input: AC, ε

Output: hit_list

$hit\_list = [\ ]$

for $i = 1$ to n **do**
  Compute P using $AC(i)$
  Compute U using P and $AC(i)$
  Compute CE using P and U
  $hit = f(c_i, \varepsilon, CE)$
  if $(hit = 1)$ **then**
    $hit\_list = [hit\_list, c_i]$;
  end if
end for
return hit_list

The proposed method relies on only fraction of the control data to not only implement effective QC but also train a powerful learning algorithm, thus, creating a cell profile.

4.4 Results

Evaluation was accomplished by selecting five bioassay plates to test the proposed method. Three main questions were answered - (1) How well can the random forest classifier discriminate between healthy and infected cells? (2) How much dataset reduction can be accomplished while retaining high discriminative power? (3) How well does the method identify known, active chemical compounds?

4.4.1 Experiment

Bioassays were designed using primary human monocyte-derived macrophages (hMDMs) on 384-well plates. The control dataset consisted of two classes defined by healthy hMDMs and those infected with a green fluorescent protein (GFP)-tagged strain of a virulent bacteria species. Infection was allowed to proceed for 30 hours and then cells were stained, fixed and imaged. There were 32 positive and 32 negative control wells per plate with total control population sizes ranging from $20,000$ to over $30,000$ cells. The remaining 320 wells per plate were reserved for hit selection analysis, yielding over $1,500$ total chemical compounds to interrogate.

4.4.2 Image Aquistion and Analysis

Images were acquired using MD Image Xpress Micro fluorescent/phase contrast microscopes for each bioassay plate. Eight sites per well were imaged excluding well edges and corners in order to capture at least 100 cells per well. As previously described in Chapter 1, four Images were captured for every site defining four distinct channels: Hoechst 33342 nuclear staining (377/477 nm excitation/emission), phase contrast cell images (no filter), GFP bacteria fluorescence (485/524 nm excitation/emission), and Live/Dead Far Red viability staining (628/692 nm excitation/emission). All images were collected as 12 bits in a 16 bit short integer data type. Individual cell image segmentation was accomplished as described in [2, 3] resulting in the creation of a nuclear and cell mask that identified the

boundaries of the nuclear and membrane regions of each individual cell, respectively. These masks were used in conjunction with the images in each channel to derive all phenotype measurements.

4.4.3 Preprocessing Data

Imputation was implemented by discarding any data point that had missing feature values. In addition, dead cells identified by Live/Dead Far Red viability were removed from all wells. In order to avoid any bias from either control population, the data was split so that training and testing datasets were balanced to contain the same number of positive and negative data points during control well analysis.

4.4.4 Parameter Implementation

Random forest classifier, mRMR feature selection, and hit selection implementation was accomplished in Matlab. The mRMR algorithm was implemented using the feature selection package distributed by Arizona State University's Data Mining and Machine Learning Laboratory (DMML) [109]. The following parameters were used:

- Random forest: 30 decision trees with 10 features randomly sampled for each tree.
- mRMR: Top 100 features; Mutual information on discretized data.

4.4.5 Multi-Well Classification Analysis

Five iterations of k-fold cross validation was performed using 10%, 20%, 50%, 80% and 90% of the control wells for training. The average of each of the five iterations is shown in figure 4.6. The accuracy of a random forest classifier using different numbers of training wells contained little variation. This demonstrated that the classifier remained consistent whether 30 or 3 control well pairs were selected. Moreover, the accuracy of different plates remained consistently over 95% even though the plates contained very low Z' factor values for the traditional univariate measurements.

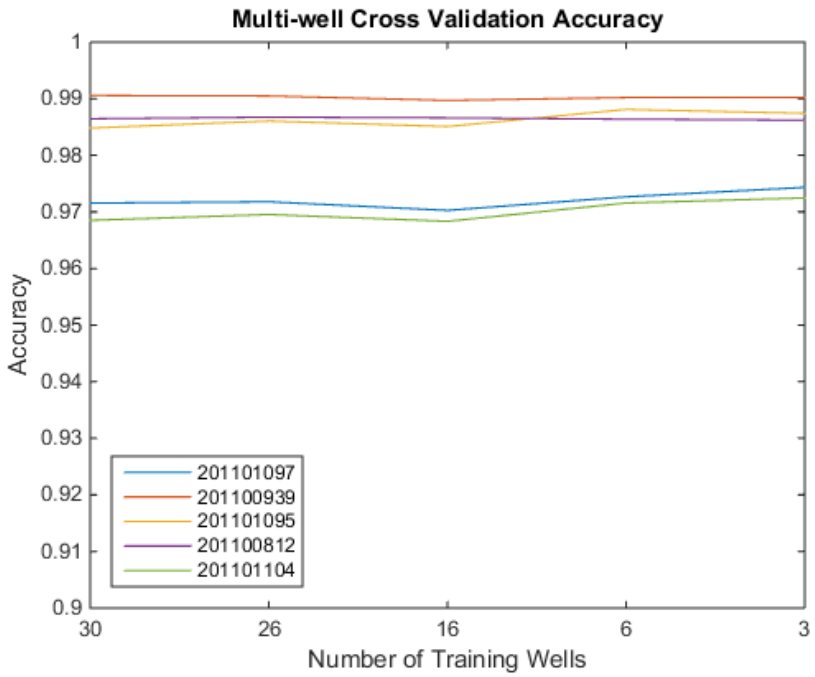
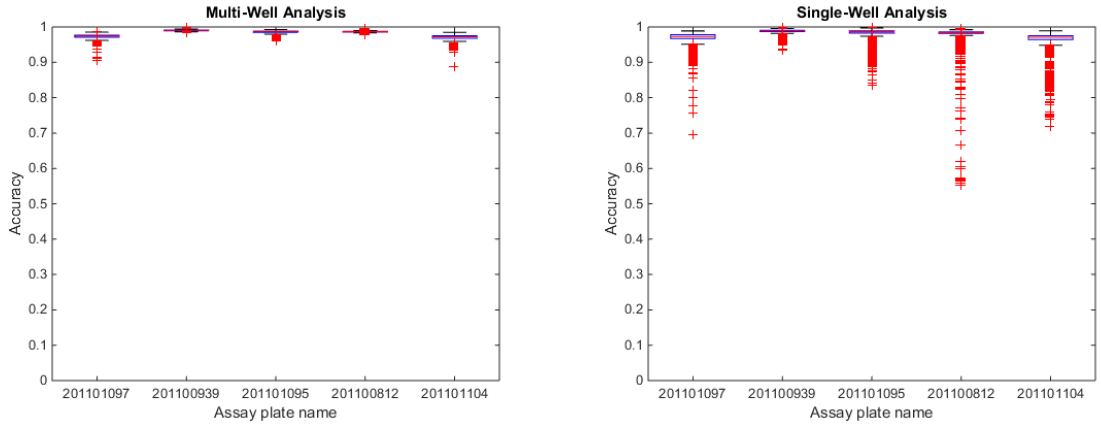


Figure 4.6: Cell classification accuracy using different number of training wells.

4.4.6 Multi-Well Versus Single-Well Classification Analysis

We further reduced the dataset to 200 data points randomly selected from a random pair of positive and negative control wells in each iteration of the k-fold cross validation. The 200 data point analysis was accomplished up to 10 times for each iteration, depending on the number of training wells. This produced over a thousand iterations across all k-fold cross validation runs for 200 randomly selected data points. The single-well control analysis demonstrated more variation then the multi-well control analysis with respect to accuracy. Figure 4.7 demonstrates the variation in accuracy when 200 data points from a single well were used.

However, the variance was not sufficient to significantly impact the average accuracy. Table 4.1 shows the average accuracy across all iterations for both single well and muti-well training. The results demonstrate that 200 data points from randomly selected single control well pairs with a reduced feature space was sufficient to achieve comparable results to using 90% of control wells. In fact, plate 201101104 demonstrated the most change of $\approx 1\%$.



(a) Multi-well analysis (b) Single-well analysis

Figure 4.7: Analysis of accuracy for multi-well and single-well selection.

Table 4.1: Single-well versus multi-well average across all iterations of cross validation.

	201101097		201100939		201101095		201100812		201101104	
	Multi-well	Single-well	Multi-well	Single-well	Multi-well	Single-well	Multi-well	Single-well	Multi-well	Single-well
Accuracy	0.972	0.969	0.990	0.988	0.986	0.983	0.986	0.979	0.970	0.961
Sensitivity	0.979	0.976	0.992	0.987	0.989	0.982	0.993	0.991	0.977	0.969
Specificity	0.966	0.963	0.989	0.990	0.984	0.983	0.980	0.968	0.963	0.954
Precision	0.966	0.964	0.989	0.990	0.984	0.983	0.980	0.971	0.964	0.955
Recall	0.979	0.976	0.992	0.987	0.989	0.982	0.993	0.991	0.977	0.969
F1 Score	0.972	0.969	0.990	0.988	0.986	0.982	0.987	0.980	0.970	0.962

This accuracy was accomplished despite all five plates receiving very low Z' factor scores using traditional univariate measurements of infection and cell density as demonstrated in Table 4.2.

4.4.7 Hit Selection Classification Analysis

The five different bioassay plates contained known, confirmed compounds that inhibit bacterial infection. The α value was set at 0.5 to give equal weight to the efficacy and uniqueness of compound activity. A standard Bonferroni correction was applied to the hypergeometric distribution's p-value using the number of compounds tested per plate.

Table 4.2: Z' factor analysis of the individual plates using traditional readout measurement for bacteria analysis.

	201101097		201100939		201101095		201100812		201101104	
	Cell Density	Infectivity	Cell Density	Infectivity	Cell Density	Infectivity	Cell Density	Infectivity	Cell Density	Infectivity
PC Mean	269.22	5626.73	277.38	3465.44	263.47	4906.28	448.75	3399.10	280.28	5339.22
PC SD	98.60	994.22	49.29	409.46	90.44	964.15	134.44	271.03	117.25	1080.72
NC Mean	544.66	2583.64	471.71	2023.61	517.78	2378.99	756.68	2053.10	517.13	2669.15
NC SD	123.61	92.51	117.94	149.16	90.57	83.33	136.80	99.28	131.05	108.31
Z' Factor	-1.42	-0.07	-1.58	-0.16	-1.14	-0.24	-1.64	0.17	-2.15	-0.34

Table 4.3: Compound hit selection using single-well analysis.

Bioassay Plate	Active	Known	Jaccard	Enrichment
201101097	23	25	0.0870	0.9130
201100939	8	4	0.5	0.5
201101095	23	26	0.0870	0.9130
201100812	1	1	0	1
201101104	11	14	0.0909	0.9091

The ε threshold was derived by:

$$\varepsilon = \alpha * (0.7) + (1 - \alpha) * \frac{-log(0.005)}{max(-log(P_C))},$$

where P_C is the lowest p-value in the plate of interest. The threshold placed equal importance on those compounds that had healthy cell populations above 70% and those that had a hypergeometric p-value below 0.005.

4.4.8 Feature Selection Variance Analysis

Using the Jaccard and the Hamming distance measures two important feature properties respectively: (1) Overlap of features rankings and (2) Alignment of feature rankings. The

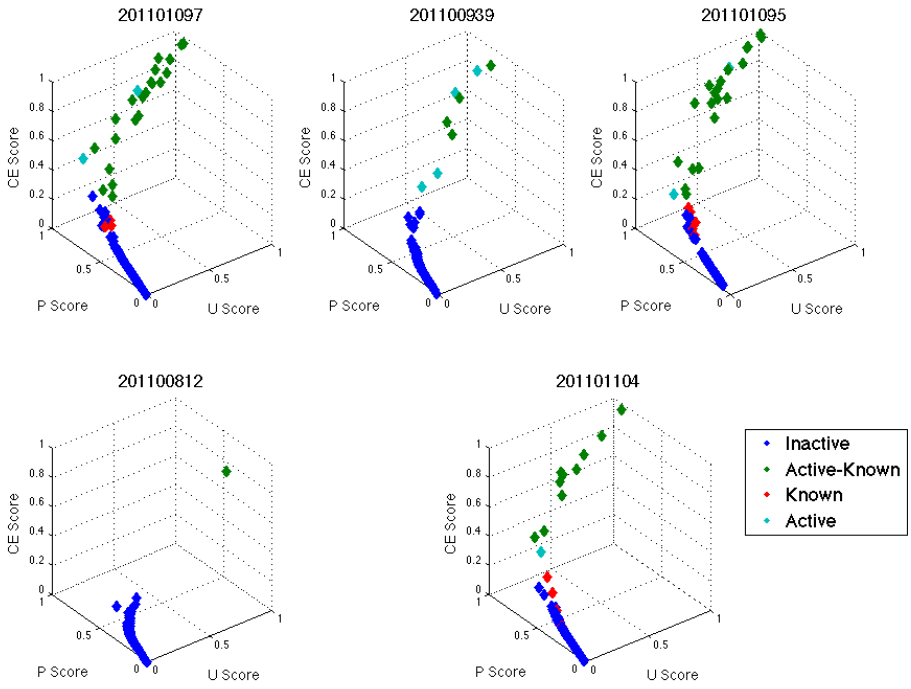


Figure 4.8: Compound hit selection for each plate where active compounds have high overlap with known compounds.

distance values for both the Hamming and Jaccard range from 0 to 1 with optimal overlap and alignment being 0 and no overlap or alignment being 1. Table 4.4 demonstrates these results for the top 105 features identified through the mRMR algorithm for each plate using different numbers of training wells ranging from a single pair to ten pairs.

4.4.9 Plate Variance Analysis

We analyze how the variance between 47 different plates tends to impact the effectiveness of the profile to discern between infected and healthy cells. One, two, and four microtiter plates were randomly selected for training a random forest model. Letting W_i be the set of controls wells used to train a random forest model, $avrI_i$, where i specifies the number of well pairs used, then $W_1 \subseteq W_2 \subseteq W_4$. This provides a view to the extent that additional wells from other plates will improve classification accuracy. Using random sampling with replacement, the process sampling the training and testing plates was repeated 30 times.

Table 4.4: Feature selection intra-plate variance analysis.

	One Well		Two Wells		Five Wells		Ten Wells	
	Jaccard	Hamming	Jaccard	Hamming	Jaccard	Hamming	Jaccard	Hamming
201101097	0.7675	0.9940	0.6964	0.9913	0.6561	0.9913	0.6703	0.9901
201100939	0.7151	0.9916	0.6548	0.9901	0.6416	0.9900	0.6420	0.9888
201101095	0.8015	0.9942	0.6857	0.9910	0.6622	0.9916	0.449	0.9882
201100812	0.7501	0.9916	0.7253	0.9900	0.6903	0.9884	0.6914	0.9879
201101104	0.7838	0.9950	0.7627	0.9930	0.7202	0.9925	0.7129	0.9911

Table 4.5: Inter-plate classification variance analysis.

	Mean	Std	Runs with Improved Accuracy
1 Plate	0.8607	0.028	-
2 Plates	0.8840	0.028	17%
4 Plates	0.8756	0.029	6 %

Figure 4.9 demonstrates the average classification accuracy for $avrI_i$ models that had the same number of training control wells for all 30 runs. Using the Wilcoxon rank sum test, we also compared the accuracy of each $avrI_i$ model in a specific run to determine whether using more wells caused a statistically significant shift ($\alpha = 0.005$) toward higher accuracy. For instance, Table 4.5 demonstrates that $avrI_4$ model had a statically significant higher accuracy than $avrI_2$ model in 16% of the 30 separate runs. The first two columns of Table 4.5 represent the average accuracy across all 30 runs fore each of the $avrI_i$ models.

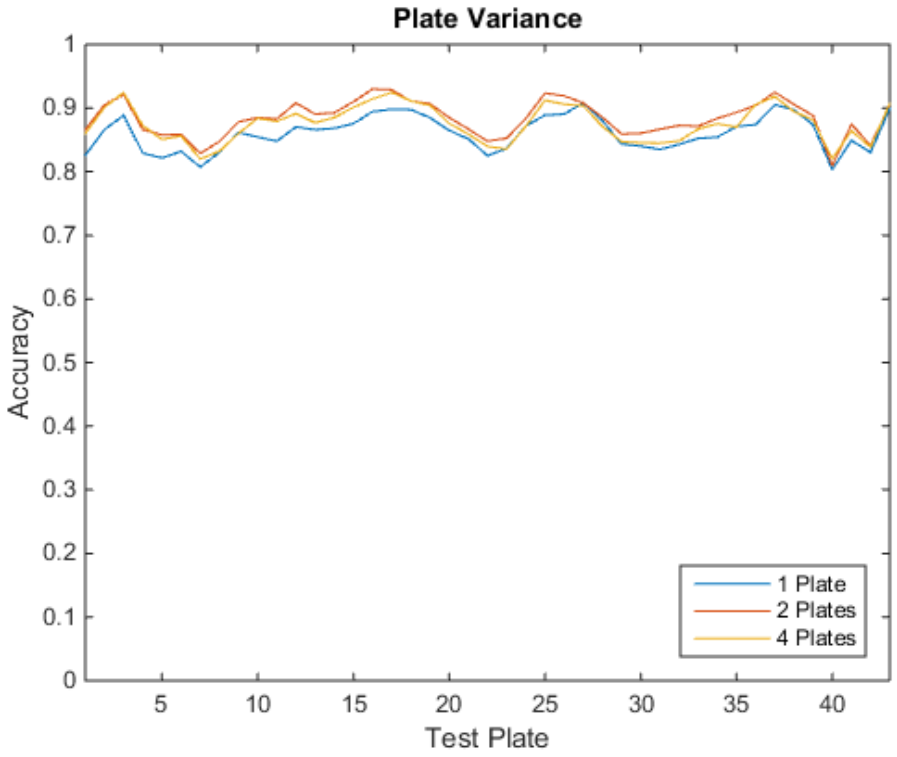


Figure 4.9: The accuracy of a random forests classifier in inter-plate control data analysis.

4.5 Discussion

The features selected by mRMR demonstrated that control data could be separated with a high degree of accuracy even with low $Z\prime-$ factors from traditional single readouts. The number of features was determined based on the fact that a random forest classifier's effectiveness is reliant upon the strength of the individual tree classifiers and the correlation between the features used in each tree. Therefore, to reduce feature correlation, we provided a larger pool from which to select descriptive features. Training a highly accurate random forest classifier was subsequently accomplished using $\leq$ 1% of total control population. However, as figure 4.7b shows limiting the number of training points increases the variance of accuracy; this was especially true in plate 201100812. The origins of the variance are more than likely a product of noise from image analytics and data collection *i.e.* environmental effects or handler error. In addition, smaller numbers of data points tend to be affected more by outliers. Although quality control is still necessary to reduce the effects of noise,

the process becomes simplified with the use of k-fold cross validation and a substantially smaller data set. The use of 200 data points to create a healthy cell profile tends to buttress the biological assumption of little variance in healthy cell phenotypes.

The healthy cell profile is limited to per plate analysis due to the amount of noise that exists in the data. This is why quality control is needed using traditional methods. The mRMR algorithm has a significant amount of variance for features selected even within a plate regardless of the number of wells used. This underscores the importance of having high dimensional feature space to select from as opposed to *a priori* assumptions of phenotype perturbation measurements that may not yield fruitful results. In a low dimensional space, this noise may be too severe for accurate measurement of phenotype perturbations. Allowing a feature selection algorithm to determine the best features provides a robust method for overcoming this noise.

Variance analysis across different plates also demonstrates that enough noise exists to require a random forest classifier to be trained per plate. If a random forest classifier is trained using a single plate for use across different plates, a sub-optimal profile will be produced that may impact the ability to properly identify active chemical compounds. Moreover, combining data from different plates demonstrates a rate of diminishing returns. This is a departure from traditionally combining data from different plates to analyzing individual plates and provides a more nuanced approach taking into consideration noise and data variance. It also provides considerable amount of parallel processing to be exploited for compound analysis. Parallelized cellomics becomes critical for high and ultra-high dimensional analysis given the amount of data generated in image-based HCS bioassays.

Table 4.3 demonstrates the proposed method's ability to identify known compounds with a high degree of accuracy. The enrichment score is the percentage of active compounds that are known compounds. The enrichment score and Jaccard distance confirms that active compounds and known compounds share a high overlapping area across different plates except in plate 201100939; this plate had a total of four known compounds and the

proposed method identified all four. However, it also identified four other compounds that have not been confirmed. These could be false-positive compounds or simply unconfirmed compounds. In plate 201100812, there was only one known compound and it was identified by the proposed method as the top scoring compound. Figure 4.8 shows the enrichment of all plates consistently identifying known compounds at the top of the active compounds list. These results buttress the healthy cell profile hypothesis for identifying novel compounds that inhibit bacterial infection.

4.6 Related Work

One of the earliest analyses of phenotype profiling using multivariate imaged-based HCS was performed by Perlman *et al.* [73]. The study investigated the dose-dependent phenotype changes induced by different drugs in human cell cultures. Young *et al.* profiled phenotype changes by compounds that affect cell proliferation using factor analysis on 36 different cytological features [102]. The application of factor analysis yielded six highly descriptive factors in projected space that were capable of describing the biological response of all cells to the chemical compounds. Christophe et al. used an image-based HCS platform to discover novel compounds that were effective at treating macrophage cells infected with different *Mycobacterium tuberculosis* strains using principle component analysis to project multivariate data into a univariate feature space [18]. These methods differ from the proposed method in that the feature space is much smaller, projection was used, and parallel processing was not utilized.

Liu *et al.* discussed using HPC systems in computational drug discovery and design (CDDD) for personalized medicine using virtual screening, molecular dynamics simulation, and protein folding [55]. Zhang *et al.* also implemented an automated massively parallel virtual screening pipeline for drug discovery endeavors [107]. Virtual screening and molecular simulation show promising results but are not substitutes for actual screening, which the proposed method accomplishes.

4.7 Summary

A novel, massively parallel method has been proposed that identifies chemical compounds capable of inhibiting bacterial infection. Implementation in a parallel processing environment allows it to work in high dimensional space with as little as 200 data points. Results indicate that biological properties such as minimal phenotype variance in healthy cells can be exploited to reduce the number of data points needed to train a classifier. This results in significantly fewer resources needed to identify active compounds through hit selection analysis. We further demonstrate that using mRMR feature selection algorithm ensures that features with optimal SSMD and constrained Z' factor will always be selected for analysis. Given the current process of identifying novel drug therapies, the reduction in data needed to identify compounds that inhibit bacterial infection improves the overall process while reducing the associated cost and time.

Investigation into optimizing single well analysis is needed to minimize performance variance. Factors such as plate location of control wells and cell density will be analyzed to determine the impact on the quality of control data. Investigating the reduction of noise in inter-plate single control well analysis to assist in the identification of effective compounds may further reduce the use of costly resources as well. Analyzing compound mechanisms of action using the novel activity measurements that have been defined in the proposed method in conjunction with random forest probability estimations is also planned. Finally, any hit selection endeavor such as antiviral vaccine identification where a healthy cell profile is ideal will benefit from this analysis. Therefore, investigation will extend to the application of the proposed method to identifying novel therapeutic drugs across a wide range of diseases.

Chapter 5

INSIGHTS PROJECT

The Insights project is an evolving software analytics platform that currently incorporates many of the proposed novel methods presented in previous chapters. It represents the next generation of image-based cellomics providing a parallelized architecture for fast and powerful analysis. An increased throughput of data acquisition, coupled with increased analytics from massive algorithm libraries, has created new big data handling problems within the novel drug discovery process of antimicrobial chemical compounds. The Insights project fully incorporates distributed processing, high performance computing, and database management that can rapidly and effectively utilize and store massive amounts of biological data generated using image-based high content screening (HCS) assays.

The first stage of the novel drug discovery process is responsible for identifying the most "active" chemical compounds that influence a specific biological outcome. This target is usually a protein structure of interest or a specific type of microbe. The result of this stage is the identification of compounds of interest or "hits" to further investigate. One key factor in identifying hits is defining the chemical compound search space. Estimates of the theoretical chemical compound space range from 10^{80} to 10^{180} with the number of those already discovered and commercially available at over $68,000,000$ [29]. Methods such as combinatorial chemistry have allowed for the rapid synthesis of large numbers of chemical compounds in a relatively short period of time [69]. Dependent on the bioassay being conducted, a large number of chemical compounds can either be selectively developed based on the biological target of interest or obtained via commercially available libraries. The vast number of chemical compounds, bacteria, and cell types to investigate produce the challenge for biologists and chemists of narrowing down the search space to those compounds that potentially have therapeutic properties. Narrowing down this search space is compu-

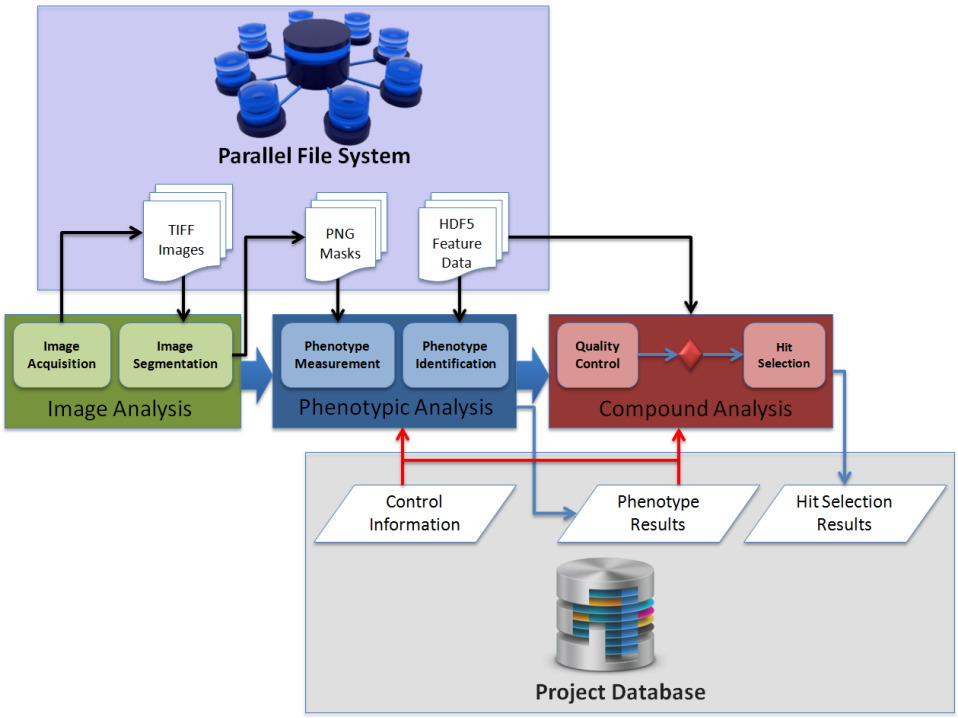


Figure 5.1: The data flow of Insights project virtual pipeline for compound analysis.

tationally expensive and generates large data sets. The Insights project was developed to assist in automated large-scale HTS campaigns by providing a more advanced method of interrogating large quantities of compounds in a relatively short period of time.

The Insights project follows the I^3 paradigm connecting the different components into a seamless, continuous virtual pipeline. Figure 5.1 demonstrates how this virtual pipeline manages information from one component to another utilizing different data management technologies to store and retrieve pertinent information. In addition, the Insights project is flexible enough to allow for the implementation of different pipelines by modularizing and varying the algorithms used within each component dependent on the biological assessment being conducted.

In this chapter, the critical role that computer vision, database management, high performance computing (HPC), and machine learning play in handling the extremely large amount of data generated by the Insights project when analyzing compounds for antibiotic properties is investigated. The Insights project handles the unique challenges that each of

the previously described HCS components present in not only generating but also analyzing and storing the data. An example of a common bioassay for bacterial inhibitors centered on the host-pathogen interaction with human macrophage cells as described in [92, 3] is provided to demonstrate how the Insights project moves from one component to the other.

5.1 Image Analytics Implementation

HCS assays generate massive amounts of data, due in large part to the host-pathogen and pathogen-only HCS assays being based on images of cell populations where each population is treated with a different compound. HCS instrumentation can vary depending on the size and requirements of the assay (A thorough list of available automated microscopy instrumentation can be found in [104]). The number of fields analyzed per microplate well also contributes to the amount of data generated. The fields, or sites, of a well refer to non-overlapping regions unless otherwise specified. The number of sites a well is split into depends on the magnifying objective, which typically ranges from $10x$ to $67x$ depending on the instrumentation used [14]. Images are subsequently acquired from the different sites of a well.

There are a number of different file formats used for microscopy imaging such as TIFF, ICS\ICS2, DIB, JPEG, *etc.* Since the uncompressed TIFF format is lossless by default and preserves cell imaging data obtained from the HCS instrumentation device, it is a safe option for use in storing cell imaging data. The instrumentation used in the Insights pipeline generates uncompressed 12-bit grayscale TIFF images, stored using 16 bits per pixel; each pixel is capable of holding 1 of 4096 different values. The bit depth describes how many gray levels there are in a gray scale image produced by different imaging microscopy. Opaque gray scale images have bit depths of 8, 12, or 16, for available numerical ranges of 0 to 255, 4095, or 65,535, respectively. Typically the pixel values are visually represented as a smooth transition from black (0) to white (the maximum value), though the reverse is possible. Since uncompressed TIFF files are only available as 8 or 16 bit files, 12-bit gray scale images are stored in 16-bit TIFF files with only the first 12 bits being utilized.

5.1.1 Image Acquisition

The MD ImageXpress XLS system can obtain phase contrast and a number of different fluorescent reagent-based images at different spectral channels. Using eight predefined sites within each well, and 4 spectral channels per site, the Insights project pipeline can produce a total of 32 12-bit TIFF images, resulting in $\approx$ 88 MB worth of data per well. Increasing the sites per well, or channels per site, will increase the data size for a single well beyond 88 MB.

In addition to the large space requirement, the sheer number of files can have detrimental effects, especially when they are all stored in the same directory, as the MD Image Xpress microscopy system does. A 384-well microtiter plate with 32 images per well yields over 12,000 images all placed within the same directory. Popular high performance file systems, such as Lustre [78], are designed to handle a small number of large files extremely well, rather than a large number of small files. Thus, the Insights project uses archiving programs with lossless compression, such as those that generate .zip files to address both problems simultaneously. Given this arrangement, placing the images for a single well into an archive file reduces the file count by a factor of 32 and, from experience, yields compression of between 50% and 60%.

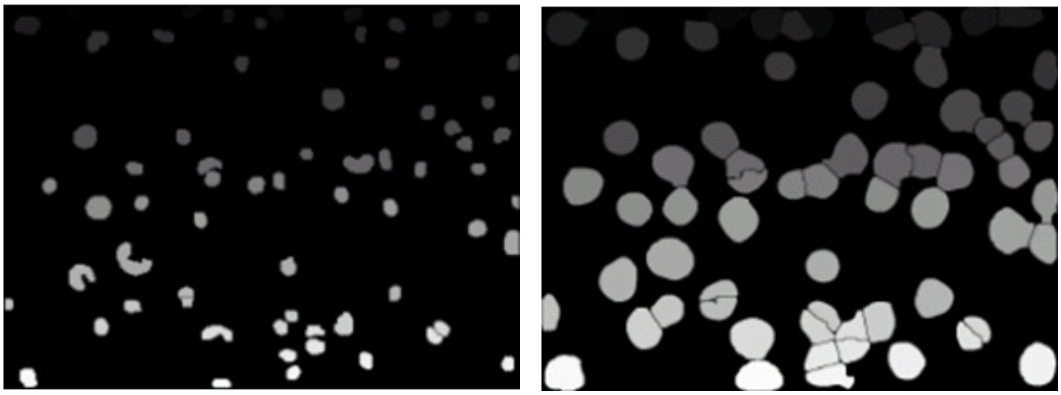
Image acquisition is the cornerstone of any HCS platform. It can have a profound effect on the quality of the bioassay and impact all subsequent stages. The quality of the images taken has a significant impact on the overall HCS process. Systematic noise, such as microscopy lighting, image focusing, and optical errors - such as phase contrast halo and shadeoffs, *etc.* - are propagated through the rest of the analysis and exacerbate known issues such as plate effects. Therefore, image preprocessing is often needed to correct noise. Since it is impossible to manually analyze every image produced, methods such as those proposed in [101, 85] incorporate computer vision optimization methods in conjunction with microscopy information to automate the removal of noise that is specific to phase contrast microscopy. The Insights project's modularity allows for correction methods such as those

previously mentioned to be integrated into its pipeline with relative ease.

5.1.2 Image Segmentation

Image segmentation is responsible for identifying individual cells and their corresponding nuclei within a given image. Since the number of individual cells to be segmented can be on the order of tens to hundreds of millions depending on the size of the HTS campaign, computer vision is needed to accomplish this task in an automated and rapid manner. The location of each cell within each site must also be stored after identification for future use.

There are two primary measurements that are used in determining the quality of cell segmentation: (1) enumeration and (2) pixel area overlap. Enumeration compares the true number of cells and nuclei that are in an image to the number identified by a chosen segmentation method. Pixel area overlap is a measure that quantifies how well the appropriate pixel region, including the boundary of the cells and their nuclei, are identified. Methods such as those proposed in [2, 3] that identify both individual cell and nucleus regions by using fluorescent reagents result in both accurate enumeration and appropriate pixel area overlap and are obviously preferred. These methods rely on relatively fast thresholding and a variant of the watershed-based algorithm for segmentation, and require less than a minute to segment a site with 1392x1040 12-bit TIFF images when run on a single core using languages that compile to machine code, such as C/C++ or Fortran. With these algorithms implemented in an interpreted language, such as Python or R, the computational requirements undoubtedly increase. While each individual site is fast, the sheer number of sites where segmentation is done provides a computational challenge on traditional computer systems. For example, assuming a low-end value of 20 seconds to segment each site, 8 sites per well, and 384 wells per microtiter plate, it would still require approximately 17 hours of computation (core-hours) using a single-threaded program to segment all the sites for a single plate alone. Fortunately, the task of cell segmentation is ideal for parallel processing where each image can be evaluated independently of other images. With enough cores working in parallel, such as on a HPC system, an entire plate could be segmented in 20



(a) Nucleus Segmentation. (b) Cell Segmentation.

Figure 5.2: Using [3, 2], the cell and nucleus regions are labeled for each individual cells creating a label image. The gray shading of the images indicates the integer label assigned to the cell and corresponding pixels. The brighter the cell region, the higher the integer label assigned.

seconds of real time using the Insights project. This would require synchronized execution across 3,072 cores and would still use 17 core-hours of computational time.

The previously described cell segmentation methods used in the Insights project produce two mask images (Figure 5.2) per site in gray scale: one each for cells and nuclei. Each pixel in a mask image will either be 0 to indicate background or a positive integer to indicate that the pixel in the TIFF image is within the bounds of a cell or nucleus. Each cell-nucleus pair within a site is assigned a unique integer label. The mask images label the pixels of interest, meaning that each pixel is critical and that a compression algorithm that is used must be lossless. For the Insights project, compression is chosen at the image level using the PNG image file format. The mask images are made up of large areas where each pixel is the same value, bordered by areas that are likewise homogeneous. This type of data layout is well suited to compression. Assuming 8 sites in a well, there will be 16 mask images (8 cell masks and 8 nucleus mask). These PNG images tend to be relatively small where the size ranges between 5 and 50 kB dependent upon the number of cells or nuclei within a given site. Therefore, each well can produce between 80 and 800 kB worth of mask images. The number of files can be an issue again, but the Insights project ameliorates these issues by archiving the masks.

5.2 Phenotypic Analytics Implementation

Phenotypic analysis identifies pertinent image measurements that describe corresponding phenotype perturbations and cellular process disruption. It is done on each individual cell and nucleus region that has been segmented and labeled. In the bacteria-based assay example, there are 100 to 1000 individual cells per well out of a 384-well microtiter plate. This produces between 250,000 and 2.5 million cells for analysis even from a small HCA screening study with six to twelve microtiter plates. A high dimensional approach is implemented where the number of different phenotype measurements for each cell can be extremely large. Each value for each phenotype measurement is recorded for further analysis. Feature selection chooses the most useful phenotype set from across the exponentially large sets of possible feature spaces that exist, and its role in phenotypic analysis is discussed further below. The Insights project provides phenotypic analysis with HPC, intelligent data storage, and machine learning, efficiently obtaining and storing results.

5.2.1 Phenotype Measurements

There are no prescribed phenotype measurements or feature spaces for a HCS assay; it is dependent upon the biological inquiry the assay is seeking to answer. The most common phenotype measurements for antimicrobial compound assays are infection levels as defined by GFP reagents and cell density. HCS assays are image-based and are, therefore, capable of creating high dimensional features spaces of phenotype measurements. The Insights project expands upon single variate phenotype measurements to produce much more descriptive assays where thousands of measurements per cell can be taken [3, 92].

Single variate analysis is ideally suited for target-based assays where compound activity is based on its effects on a target protein i.e. inhibition or activation. However, the complexity of a cell and its respective response to a compound cannot be easily gleaned from a single measurement. Although high dimensional phenotype measurements increase data size and add complexity to an experiment, it is also better suited to providing phenotype

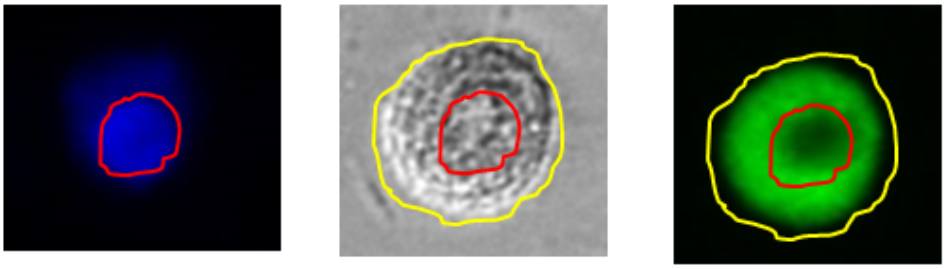


Figure 5.3: Phenotype measurements using [66, 3] in conjunction with the (Red) nucleus mask and (Yellow) cell mask can produce features that describe important cellular and sub-cellular characteristics.

information describing host-pathogen and host-compound interaction. Different biological image libraries such as WNDCharm [66] and those proposed by [3] produces image measurements based on well known edge, shape, texture, and intensity descriptors such as Gabor filters, Haralick, Laplacian, Gaussian features,*etc.* Utilizing the small set analysis in WNDCharm and those proposed by [3] alone, the Insights project is capable of producing over $11,000$ features per cell when used in conjunction with phase contrast, GFP, and nucleus channels in a host-pathogen designed assay. The number of features increases to close to $20,000$ when large set analysis is used in WNDCharm. Figure 5.3 demonstrates the region of the cell and nuclei phenotype measurements that are obtained from using cell and nucleus masks in conjunction with the different spectral channels.

Evaluation for the vast majority of features ($> 99.9\%$) at the cell level are independent allowing for massive parallelization to occur. Additionally, while a small group of features may be interrelated, separate groups of features are independent from each other. This domain decomposition in both the spatial and functional domains provides a wealth of parallelism to exploit. Depending on the hardware and compilers available, the normalized feature evaluation computational cost is between 30 core-seconds and one core-minute per cell. For a screen with with 100 million cells, that is roughly equal to 100-200 core-years of computational time. While segmentation benefits from HPC, the phenotype measurement phase is where HPC becomes critical to the Insights project.

In ultra high dimensional space, raw feature data is too large to effectively store and retrieve in a traditional database. The Insights project solution is to aggregate the data

and store all of the features for a single plate in an HDF5 file [90] following the completion of phenotype measurement. The use of HDF5 provides a number of advantages over a traditional database:

- No server requirement.
- No file or dataset size limits (MySQL has a hard limit of 4096 columns per table. [MySQL 5.7 Reference Manual, Appendix C.10.4, p. 4146])
- A single file containing all features and meta-data can be sent to colleagues for further or alternative analysis.
- Built-in lossless compression.
- Hierarchical structure (file system within a file) allows for easy navigation to values.
- Well-supported in multiple languages, including popular post-processing languages such as Python and R.
- HDF5 files can be accessed in parallel.

The HDF group also provides tools for converting HDF5 files to text (comma separated variable), as well as a cross-platform spreadsheet viewer to explore an HDF5 file. Wells are stored separately within the HDF5 file as groups, which are analogous to directories in a file system. Within each well group, the results for each feature are stored continuously rather than storing the results for each cell contiguously. This reduces the time spent in the *Phenotype Identification* sub-component and the *Compound Analytics* component where only a small subset of features are used at any given time to prevent inefficient strided data accesses. In addition, the conversion from uncompressed text to compressed native-format floating point numbers yields a median compressed size $\approx 25\%$ that of the original.

5.2.2 Phenotype Identification

The Insights project generates high dimensional data that is often noisy and redundant. It, therefore, requires an additional preprocessing step to reduce redundancy and remove irrelevant and noisy phenotype measurements. A powerful tool in machine learning known as dimensionality reduction has become an integral component to reducing high dimensional feature space to the most important components. Methods such as factor analysis and principle component analysis project the multivariate data to a reduced subspace for further phenotypic analysis. Other supervised feature selection methods such as sparse learning [91], Fast Correlation-Based Filter (FCBF) [103], maximum relevance minimum redundancy (mRMR) [72] select the most descriptive phenotype measurements in the original data space making them more ideally suited for phenotypic analysis. The Insights project allows the end user to select from several of the previously mentioned dimensionality reduction techniques.

Supervised feature selection methods require additional data as they rely on ground truth labeled data to determine the importance of features. Control data has proven to be a viable substitute as ground truth training labeled data quite well, especially if the pathogen being used is extremely infectious. The computational complexity of the different feature selection algorithms may also require HPC if the control data being utilized is sufficiently large. For instance, mRMR and FCBF feature selection methods can be implemented in a parallel processing environment reducing real time spent on computation by an order of magnitude or more. The total amount of data generated in this stage is small, amounting to a list of feature names and associated values (scores) each time an algorithm is run and, therefore, can be redundantly stored in a traditional database and file system for convenient future access.

5.3 Compound Analytics Implementation

The *Compound Analysis* component is responsible for utilizing the previously generated data to identify active compounds that demonstrate therapeutic properties. Machine learning has proven its utility in providing a more robust multivariate analysis of chemical compounds compared to traditional single variate statistical methods. The Insights project is specifically designed for multivariate machine learning analysis of chemical compound activity. This component generates very little data by utilizing the data from the previous components to determine the reliability of the generated data and, subsequently, compound activity. Quality control is the fist step of *Compound Analysis*; it ensures that the data generated in the previous components is viable and capable of distinguishing truly active compounds from those that are inactive. Once the reliability of the bioassay is verified the initial primary screening is achieved using the compound activity measurement.

5.3.1 Quality Control

Two widely accepted quality control (QC) measurements are the Z'- and Z-factors [106]. These methods measure the overlap between two distributions obtained from control and compound data calculated using the dynamic range and variance. The dynamic range represents how distant the means of two different distributions are from each other. The variance provides information on how spread out each distribution is. The control data has traditionally been used to give an approximated upper and lower bound on a compound activity measurement. If the measurement contains too much overlap, then it results in a poor Z'-factor value. Similarly, the Z-factor determines the overlap between the distributions produced by test compounds and the control data.

There are a few major drawbacks with the Z'- and Z-factors QC measurements. First, they assume that the test and control data distributions are normal. Second, they require a significant number of data points to compute and obtain a confident QC measurement, increasing the required size of data for an HCS assay. Third, they are designed for single

dimension measurements requiring dimensionality reduction of multivariate data to a single feature.

Machine learning algorithms do not require massive amounts of data and are designed to work with multivariate data. In addition, there are many powerful learning algorithms that do not make any assumptions on data distributions. To assess the quality of an HCS assay, cross-validation can be performed on control wells, using part of the controls to train a classifier and testing that classifier on the rest of the controls. For instance, the Z'-factor, as previously described, is used to determine how well a phenotype measurement separates control wells; Trevino *et al.* previously demonstrated that even when a plate produces an unacceptably low Z'-factor for each feature, a random forest classifier can still separate infected from uninfected cells with greater than 95% accuracy [92]. Moreover, the amount of cells required to train a highly accurate classifier was approximately 10% of the total control data available. This shows the ability of machine learning algorithms to give a more robust and accurate quality assessment of bioassays in multidimensional space. Accordingly, the Insights project uses machine learning techniques and accomplishes quality control using cross-validation.

5.3.2 Hit Selection

In HCS platforms, compound activity is derived from the phenotype measurements of the cell population treated with a given compound. Traditionally, a single phenotype measurement was taken for each cell in a well. A well summary value would subsequently replace the individual cell phenotype measurements with a single value representative of the population. Different well summary methods include the mean, median, percentile scores, and other distribution characteristics of a cell population. Compound activity measurement was based on these well summary values making them a product of cell population distribution characteristics. With a single value representing compound activity, each individual compound was subsequently quantitatively compared to other compounds. Two of the most widely used comparison methods are the "top K" approach and the "outliers" approach.

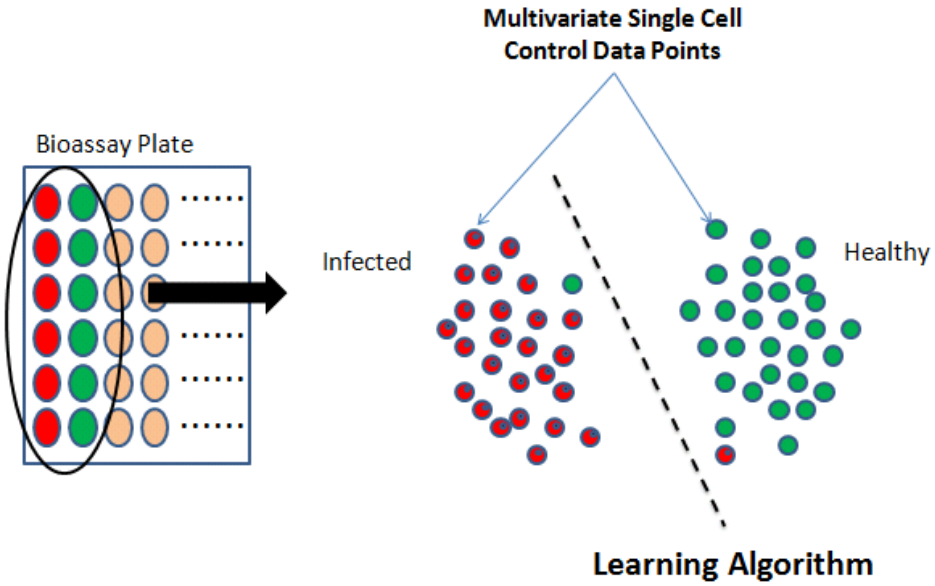


Figure 5.4: The quality of a plate can be defined by a learning algorithm as the accuracy of separating infected from uninfected control data points [92].

The top K approach orders the compound activity measurement from the value considered most effective to least effective. A threshold is manually selected to choose the top K compounds as being active. The outliers method assumes a normal distribution from all the compound activity measurements available. Those measurements that are three standard deviations or more away from the mean of the compound activity distribution that signify inhibition of bacteria are considered active.

There are drawbacks to these comparison methods. First, the comparison methods are based on activity relative to other compounds. If none of the compounds that are being analyzed are effective, the comparison methods will simply identify the ineffective compounds. On the other hand, if there are a substantial number of effective compounds, neither comparison method will identify all effective compounds. Second, the comparison methods are reliant upon a single value to represent compound activity. Not surprisingly, Singh *et al.* showed that most biological assessments still utilize 1 to 2 dimensional space when analyzing compound activity [82]. This requires that multivariate data at the cell level be reduced to a single value in order to properly utilize these methods. Using a single value

to represent the complexity of compound activity limits the analysis to a one dimensional description.

Multivariate cellomics can be used to define a more robust compound activity measurement. Loo *et al.* demonstrated the feasibility of machine learning in compound activity analysis by allowing a support vector machine (SVM) to define a compound activity profile based on cellomics [56]. A SVM hyperplane separates compound-treated cells from the control cells; this hyperplane represents the respective compound activity profile. In addition, Trevino *et al.* defined a "healthy cell" profile to identify compounds that inhibit bacterial infection by training a random forest classifier using control positive and control negative cells. This profile was subsequently used to predict compound-treated cells as either infected or uninfected [92]. Compound activity was then determined based on a compound effectiveness (CE) score that incorporated protection and uniqueness scores based on the number of cells classified as infected and uninfected. This method demonstrated the ability to properly identify known compounds that inhibited bacterial infection even when plate quality is low by traditional quality control standards as given by the Z'- factor.

The Insights project incorporates the use of both random forest and support vector machine, in addition to many other learning algorithms that can be used to analyze multivariate data. With the use of these algorithms, there is no need to compress phenotype pertubation measurements into a single value. The computational cost of training a machine learning system varies widely between algorithms and depends heavily on the size of the feature space the cell data is in, the classifying power of the cell features, and the amount of training data used. Although, the initial training of a learning algorithm may be computationally expensive, once a learning algorithm has been trained classification is typically computationally inexpensive.

5.4 Software Pipeline Description

The full implementation of the Insights project pipeline is a multifaceted approach heavily reliant on parallel processing in a HPC environment. Figure 5.6 provides an overview of

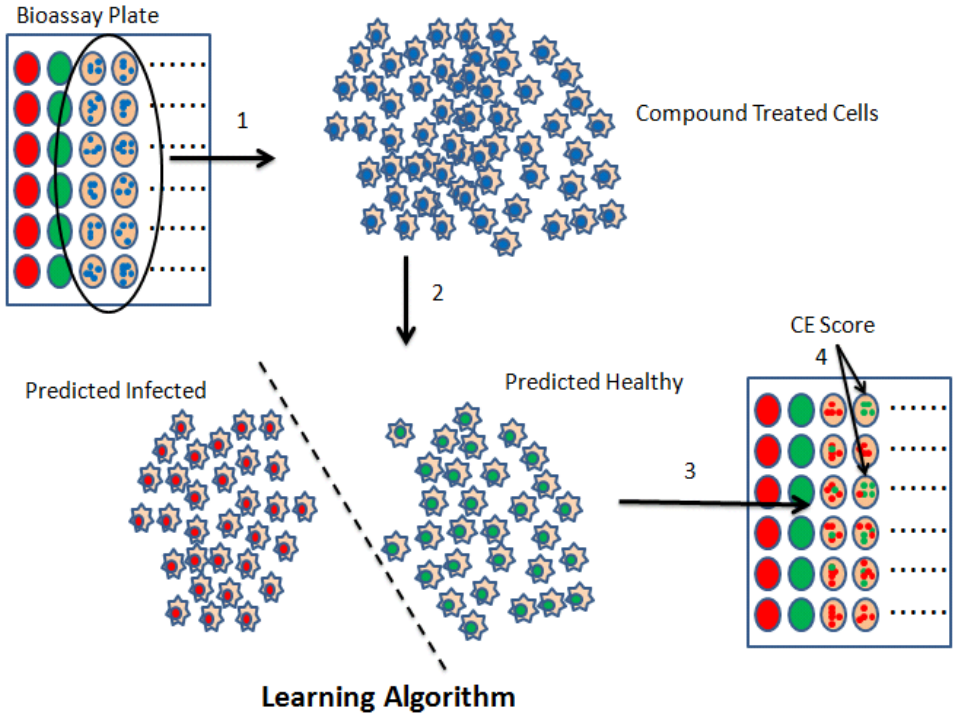


Figure 5.5: Hit selection using machine learning to determine compound activity based on cellomics as desribed in [92]. A compound activity score called compound effectiveness (CE) score determines the most active compounds.

the architecture used to fully implement the pipeline. At the heart of the distributed processing is the MindModeling@home project [38]. This project is built upon the Berkeley Open Infrastructure for Network Computing (BOINC) software [5] with a proprietary web-based user interface. The interface is composed of a web server and database package known as μBatch that is responsible for launching and tracking the status of different parallel jobs. A dedicated project web server is responsible for providing the MindModeling@Home project with the necessary information needed to run the pipeline. The project web server is also responsible for interfacing with the user or client and converting their requested pipeline implementation into a series of jobs or a batch. Once a batch request is made, the project web server takes the batch information and launches parallel jobs through automated messaging to the μBatch web server. The project web server also updates a project database with pertinent metadata information required by the different pipeline components. The μBatch web server, subsequently, updates the corresponding μBatch database with pertinent job

information. BOINC clients on supercomputer nodes are launched and retrieve jobs from the μBatch database following what is known as a "bag of tasks" paradigm. The different jobs retrieve the necessary data files for each pipeline component from a remote parallel file system and store that data in a more local parallel file system "workspace" that HPC compute nodes are able to access. This complex series of interactions allow the pipeline to be implemented in a highly parallelizable manner cutting down the amount of real time needed to obtain results. Once the results are obtained, the project web server provides the user with a hyper link to the location where the desired results are stored.

Each compound going through the pipeline produces a modest amount of data easily over 100 MB. As the number of compounds and control data increases, the amount of total data generated and computational resources used also increases. Logistical support for the pipeline is critical to ensure that data is generated and utilized in an efficient and accurate manner. Each component in a pipeline will generate data that will subsequently be utilized in other components as previously shown in Figure 5.1. The images (raw and mask), the phenotype measurements, the selected features, and the final hit selection all require tracking from one stage to the next in the pipeline. There is a combinatorial explosion of paths from image acquisition to hit selection due to the wide array of different algorithms that can be incorporated in each stage of a pipeline. Thus, the more modular and flexible each component in a pipeline is, and based on how many different algorithms have been incorporated, the more tracking of data is required from one stage to the next. One way to address this need is by utilizing a relational database to track the dependencies between the stages. For instance, each execution of a stage in a pipeline is assigned a job identification number (JID) and carries with it a set of metadata including the JID of the previous stage and stage dependent information. The data stored on the file system can then be stored in directories or HDF5 groups that include the JID for an organized, programmatic way of accessing the data. As mentioned before, the results from the phenotype identification and hit selection stages are very small, and can therefore be stored directly in the

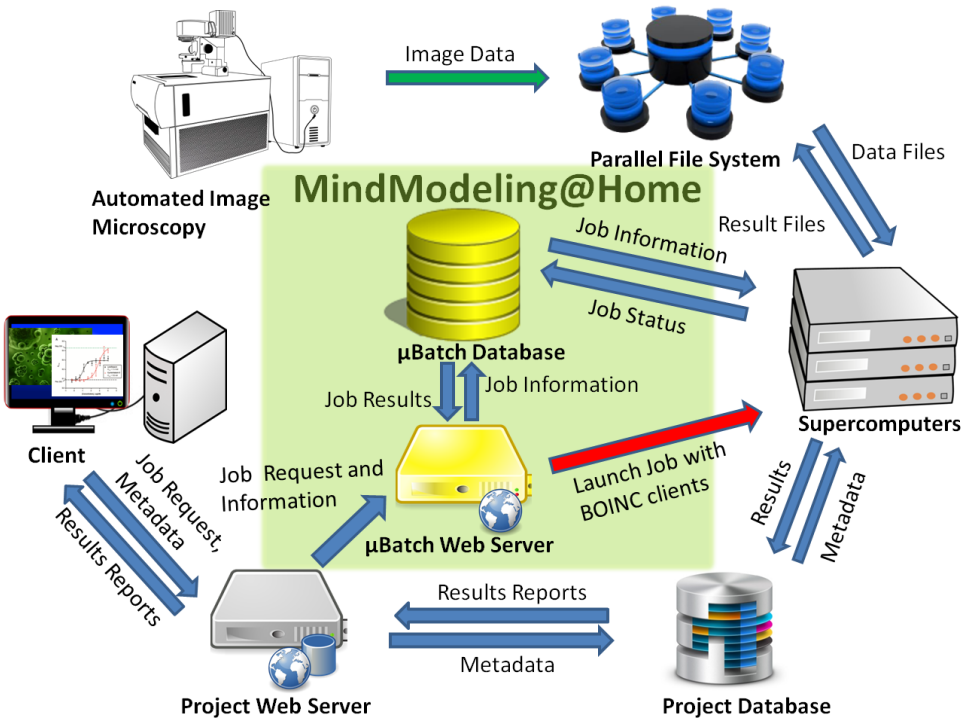


Figure 5.6: The Insights project utilizes the MindModeling@Home project to implement its virtual pipeline in a parallelized manner.

database. Utilizing a relational database allowed researchers to quickly identify all hits for a given screen, examine patterns in feature identification across different high-throughput biochemical assay campaigns and perform more efficient compound analysis to tease out novel relationships.

One significant challenge to overcome was determining the extent to which the different algorithms implemented could be parallelized given the data. This influences the number of computational resources that are expended in a specific component or sub-component of the pipeline. Larger data sizes in each component does not necessarily imply that more computational resources will be used. For instance, the Insights project was deployed and implemented the pipeline utilized in [92]. Figure 5.7 gives a summary of the data distribution that was generated with the following specifications:

- 384-well microtiter plate.
- 8 sites per well.

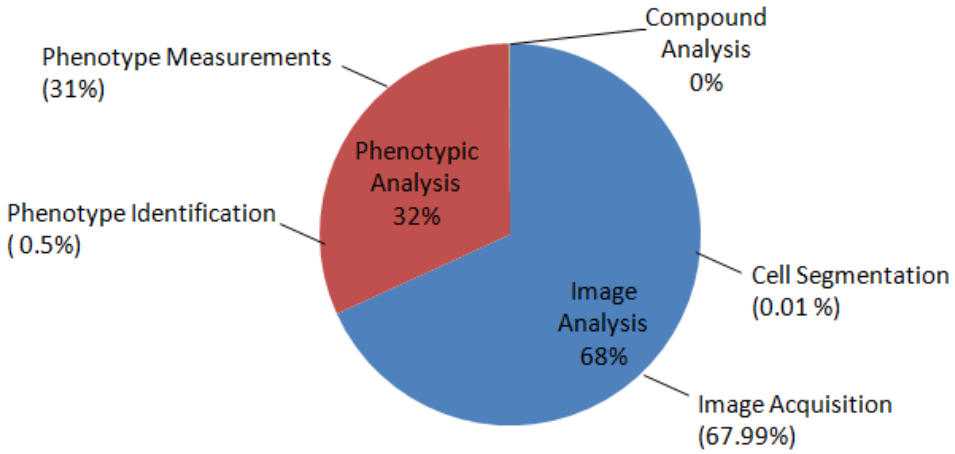


Figure 5.7: The distribution of data shows that the majority of the data comes from the *Image Analysis* component.

- 4 TIFF images per site (phase contrast, covalent florescent, non-covalent florescent, indicator florescent channels).
- 2 PNG label images (nucleus and cell regions) per site using image segmentation [3].
- 1 HDF5 file per plate containing 11,000 features per cell using [66, 3].
- Hit selection using [92].

This particular pipeline is capable of producing over 2 TB worth of information when conducting a medium scale screening of 25,000 compounds and over 123 TB worth of information in large scale screening comprising a million or more compounds. HPC computing resources were overwhelmingly relegated to calculating phenotype measurements in the *Phenotypic Analysis* component as shown in Table 5.1 even though the majority of data was generated in the *Image Analysis* component. Although identifying the most descriptive phenotypes and most active compounds requires the least amount of HPC resources, they can have memory requirements in the multi-GB range. In order to overcome this obstacle, algorithms, such as FCBF for feature selection, were re-implemented for parallelizing, reducing the real time to perform the calculation and allowing the memory burden to be distributed. In addition, different algorithms were ported over to C++ from a number of

Table 5.1: Compute hours in a sample image-based HCS pipeline for each step requiring HPC resources.

	Completed Jobs	Mean	Standard Deviation	Median
Image Segmentation	175	3.82	3.71	2.84
Phenotype Measurement	138	877.43	283.44	848.62
Phenotype Identification	157	0.25	0.20	0.31
Hit Selection	161	0.32	0.35	0.14

different languages including C# and Java to more efficiently execute in a HPC environment.

5.5 Summary

Image-based HCS assays have produced a massive amount of data in modern day antibiotic drug discovery endeavors. As a corollary, computer science and HPC have become necessary to sufficiently handle this increase in data. We introduce the Insights project, a software tool that exploits the latest technology in HPC, machine learning, and data management to more efficiently and effectively analyze cellular processes. The Insights project creates a virtual, automated, and contiguous pipeline through the use of distributed processing and is modular enough to allow for the expansion of learning algorithms across the different pipeline components.

The future of novel drug discovery will continue to witness an increase in the already massive amounts of data generated during HTS campaigns for antibiotic compounds. With next generation (next-gen) sequencing technology producing more specific genome information, more hybrid-like assays will be developed that combine HCS phenotypic analysis with next-gen sequencing. Next-gen sequencing has increased the number of DNA Base-pairs (Bp) per run from 96 kB to 1-3 GB per run [62]. This increase in DNA Bp allows a more nuanced target-based analysis that will incorporate corresponding phenotypic analysis. It

has also increased the amount of data needed to be analyzed and stored leading to computer science undoubtedly playing a more crucial role in future phenotype perturbation analysis.

Chapter 6

FUTURE WORK AND CONCLUSION

This dissertation investigates the ever increasing role that machine learning plays in drug discovery endeavors. Specifically, parallelized cellomics is investigated to determine whether improvements in initial chemical compound screening, called "hit selection',' can be accomplished over traditional single readout methods. Novel methods were proposed and investigated for the three major components that define an automated HCS assay. In the *Image Analytics* component, a promising method was introduced that segments individual cells by identifying the corresponding nucleus regions using a convolutional neural network that uses spatial and texture information in conjunction with a modified set cover algorithm. In the *Phenotypic Analytics* component, two novel methods were introduced. The first transfers information from the bug domain to the cell domain to improve feature selection in the cell domain. The second novel method is a non-parametric feature selection algorithm that does not assume a Gaussian distribution in datasets. The algorithm outperformed well known feature selection algorithms that assume Gaussian distribution. In the *Compound Analytics* component, a parallelized cellomics-based method was proposed that demonstrated its ability to develop a highly accurate classifier using a small number of cells as data points. The proposed method further demonstrated the ability to identify active chemical compounds using a "compound effectiveness" score. A software analytics tool called the Insights project was also described that provides a virtual pipeline for analysis in the different components. The Insights project provides a robust and efficient platform for handling the massive amounts of data generated in automated HCS assays.

There are many extensions, theoretical and applicable, that are worth further exploring. For instance, research into whether matrix completion is a viable tool for transferring information from the rich and descriptive bug and nuc domains to the cell domain is worth

pursuing in a continued attempt to minimize fluorescent reagent use. If this is a viable alternative, than the cost and time to analyze HCS data will undoubtedly decrease. Another exciting direction will be to allow deep learning algorithms to select chemical compounds from start to finish. This will be a computationally intensive task that can currently only be accomplished in a parallelized HPC environment. Allowing deep learning networks to communicate between themselves on the information needed to make proper decisions at each of the different components previously described will undoubtedly create a sophisticated and powerful hit selection process with the potential to increase the success of identifying candidate chemical compounds that eventually become novel therapeutic drugs.

The Insights project will also continue to evolve into a more powerful and effective automated HCS software pipeline. Its modularity allows for expansion with ease across the *Image Analytics*, *Phenotypic Analytics*, and *Compound Analytics* components. As the data continues to increase, big data platforms such as Apache Hadoop and Spark will be investigated to determine the feasibility of integrating them into the software pipeline.

The current state of the drug discovery process is inefficient and costly with present trends making the future look bleak. However, computer science has the unique opportunity to revolutionize and propel novel drug discovery into a new golden age of discovery. This will undoubtedly have a social and economic impact on society and humanity, as a whole. As trends continue to move towards personalized medicine, consequently increasing the amount of data generated, big data analytics using machine learning algorithms will continue to expand its role in important biological endeavors such as novel drug discovery.

REFERENCES

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2281, 2012.

[2] U. Adiga, B. L. Bell, L. Ponomareva, D. Taylor, R. Saldanha, S. Nelson, and T. J. Lamkin. Mapping infected cell phenotype. *IEEE Transaction On Biomedical Engineering*, 59(8), 2012.

[3] U. Adiga, D. Taylor, R. Kramer, S. Roland, S. Nelson, and T. J. Lamkin. Automated analysis and classification of infected macrophages using bright-field amplitude contrast data. *Journal of biomolecular screening*, 17(3):401–408, 2012.

[4] U. P. S. Adiga, R. Malladi, W. Baxter, and R. M. Glaeser. A binary segmentation approach for boxing ribosome particles in cryo EM micrographs. *Journal of Structural Biology*, 145:142–151, 2004.

[5] D. P. Anderson. BOINC : A System for Public-Resource Computing and Storage. 2015.

[6] A. Argyriou, T. Evgeniou, and M. Ponti. Multi-Task Feature Learning. *Advances in Neural Information Processing Systems*, 19, 2007.

[7] Q. Au, P. Kanchanastit, J. R. Barber, S. H. I. C. Ng, and B. I. N. Zhang. Chaperone Amplifiers in Heat Shock. *Society for Biomolecular Sciences*, pages 953–959, 2008.

[8] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio. Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

[9] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio. Theano: a {CPU} and {GPU} Math Expression Compiler. In *Proceedings of the Python for Scientific Computing Conference ({SciPy})*, jun 2010.

[10] S. Beucher. *Watershed, Hierarchical Segmentation and Waterfall Algorithm*, pages 69–76. Springer Netherlands, Dordrecht, 1994.

[11] M. Blachnik, W. Duch, A. Kachel, and J. Biesiada. Feature Selection for Supervised Classification : A Kolmogorov- Smirnov Class Correlation-Based Filter. *Methods of Artificial Intelligence*, pages 33–40, 2009.

[12] S. R. Bowling, M. T. Khasawneh, S. Kaewkuekool, and B. R. Cho. A logistic approximation to the cumulative normal distribution. *Journal of Industrial Engineering and Management*, 2(1):114–127, 2009.

[13] L. Breiman. Random forests. *Machine learning*, pages 5–32, 2001.

[14] P. Brodin and T. Christophe. High-content screening in infectious diseases. *Current Opinion in Chemical Biology*, 15(4):534–539, 2011.

[15] D. Calcoen, L. Elias, and X. Yu. What does it take to produce a breakthrough drug? *Nature Reviews Drug Discovery*, 14(3):161–162, 2015.

[16] A. E. Carpenter, T. R. Jones, M. R. Lamprecht, C. Clarke, I. H. Kang, O. Friman, D. a. Guertin, J. H. Chang, R. a. Lindquist, J. Moffat, P. Golland, and D. M. Sabatini. CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome biology*, 7(10):R100, 2006.

[17] T. Chen. *A Practical Guide to Assay Development and High-Throughput Screening in Drug Discovery (Critical Reviews in Combinatorial Chemistry)*. CRC Press, Boca Raton, 2010.

[18] T. Christophe, M. Jackson, K. J. Hee, D. Fenistein, M. Contreras-Dominguez, J. Kim, A. Genovesio, J. P. Carralot, F. Ewann, E. H. Kim, S. Y. Lee, S. Kang, M. J. Seo, J. P. Eun, H. Škovierová, H. Pham, G. Riccardi, N. J. Youn, L. Marsollier, M. Kempf, M. L. Joly-Guillou, T. Oh, K. S. Won, Z. No, U. Nehrbass, R. Brosch, S. T. Cole, and P. Brodin. High content screening identifies decaprenyl-phosphoribose 2 epimerase as a target for intracellular antimycobacterial inhibitors. *PLoS Pathogens*, 5(10), 2009.

[19] D. C. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images. In *NIPS*, pages 2852–2860, 2012.

[20] D. C. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Mitosis Detection in Breast Cancer Histology Images with Deep Neural Networks. In *MICCAI*, volume 2, pages 411–418, 2013.

[21] D. C. Ciresan, U. Meier, J. Masci, and J. Schmidhuber. A Committee of Neural Networks for Traffic Sign Classification. In *International Joint Conference on Neural Networks*, pages 1918–1921, 2011.

[22] D. C. Ciresan, U. Meier, J. Masci, and J. Schmidhuber. Multi-Column Deep Neural Network for Traffic Sign Classification. *Neural Networks*, 32:333–338, 2012.

[23] W. Dai, Y. Chen, G.-r. Xue, Q. Yang, and Y. Yu. Translated Learning: Transfer Learning across Different Feature Spaces. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 353–360. 2008.

[24] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 886–893, Washington, DC, USA, 2005. IEEE Computer Society.

[25] H. Daumé III. Frustratingly easy domain adaptation. *ACL*, 2007.

[26] J. C. Davis and R. J. Sampson. *Statistics and data analysis in geology*. New York, 1986.

[27] M. A. A. Dewan, M. O. Ahmad, and M. N. S. Swamy. A Method for Automatic Segmentation of Nuclei in Phase-Contrast Images Based on Intensity, Convexity and Texture. *Biomedical Circuits and Systems, IEEE Transactions on*, 8(5):716–728, oct 2014.

[28] J. A. Dimasi, H. G. Grabowski, and R. W. Hansen. Innovation in the pharmaceutical industry : New estimates of R & D costs. *Journal of Health Economics*, 47:20–33, 2016.

[29] H. Djaballah. Chemical space , high throughput screening and the world of. *Drug Discovery World Spring 2013*, pages 73–78, 2013.

[30] L. Duan, D. Xu, and I. W. Tsang. Learning with Augmented Features for Heterogeneous Domain Adaptation. In *Proceedings of the 29th International Conference on Machine Learning*, 2012.

[31] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, New York, 2 edition, 2001.

[32] L. Elden. *Matrix Methods in Data Mining and Pattern Recognition*. Society for Industrial and Applied Mathematics, Philadelphia, 2007.

[33] U. M. Fayyad and K. B. Irani. Multi-Interval Discretization of Continuos-Valued Attributes for Classification Learning, 1993.

[34] J. Gorenstein, B. Zack, J. R. Marszalek, A. Bagchi, S. Subramaniam, P. Carroll, and C. Elbi. Benchmarks Reducing the multidimensionality of powerful descriptors. *Benchmarks*, 49(3):663–665, 2010.

[35] Y. Guo and M. Xiao. Cross Language Text Classification via Subspace Co-Regularized Multi-View Learning. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 1615–1622, 2012.

[36] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural Features for Image Classification. *Systems, Man and Cybernetics, IEEE Transactions on*, SMC-3(6):610–621, nov 1973.

[37] M. Harel and S. Mannor. Learning from Multiple Outlooks. *Proceedings of the 28th International Conference on Machine Learning*, 2011.

[38] J. Harris, K. A. Gluck, L. Martin, and S. K. St. MindModeling @ Home . . . and Anywhere Else You Have Idle Processors. *Proceedings of the ninth international conference on cognitive modeling*, pages 448–449, 2009.

[39] R. He, T. Tan, L. Wang, and W.-s. Zheng. l 2 ,1 Regularized Correntropy for Robust Feature Selection. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2504–2511, 2012.

[40] A. A. Hill, P. Lapan, Y. Li, and S. Haney. Impact of image segmentation on high-content screening data quality for SK-BR-3 cells. *BMC Bioinformatics*, 13:1–13, 2007.

[41] L. Hou, D. Samaras, T. M. Kurc, Y. Gao, J. E. Davis, and J. H. Saltz. Resolution Image Classification. 2015.

[42] N. Jaccard, L. D. Griffin, A. Keser, R. J. Macown, A. Super, F. S. Veraitch, N. Szita, F. S. Varaitch, N. Szita, F. S. Veraitch, and N. Szita. Automated method for the rapid and precise estimation of adherent cell culture characteristics from phase contrast microscopy images. *Biotechnology and Bioengineering*, 111(3):504–517, 2014.

[43] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 2146–2153, sep 2009.

[44] T. Kazmar, M. Smid, M. Fuchs, B. Luber, and J. Mattes. Learning cellular texture features in microscopic cancer cell images for automated cell-detection. In *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pages 49–52, aug 2010.

[45] I. Kononenko. Estimating attributes: Analysis and extensions of RELIEF. *Machine Learning: ECML-94*, 784:171–182, 1994.

[46] P. Kovesi. Image Segmentation using SLIC SuperPixels and DBSCAN Clustering, 2013.

[47] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[48] B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *In Proc. of CVPR*, 2011.

[49] T. Kurita, N. Otsu, and N. Abdelmalik. Maximum likelihood thresholding based on population mixture models. *Pattern Recognition*, 25(1231-1240), 1992.

[50] F. Lampariello. On the Use of the Kolmogorov-Smirnov Statistical Test for Immunofluorescence Histogram Comparison. *Cytometry*, 188:179–188, 2000.

[51] J. Li, C. Kewei, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu. Feature Selection: A Data Perspective. *CoRR*, pages 1–73, 2016.

[52] E. Limpert, W. A. Stahel, and M. Abbt. Log-normal Distributions across the Sciences : Keys and Clues. *BioScience*, 51(5):341–352, 2001.

[53] H. Liu, F. Hussain, C. L. I. M. Tan, and M. Dash. Discretization : An Enabling Technique. *Journal of Data Mining and Knowledge Discovery*, 6(4):393–423, 2002.

[54] H. Liu and R. Sentiono. Chi2: Feature Selection and Discretization of Numeric Attributes. *IEEE*, 1995.

[55] T. Liu, D. Lu, H. Zhang, M. Zheng, H. Yang, Y. Xu, C. Luo, W. Zhu, K. Yu, and H. Jiang. and molecular simulation. *National Science Review*, pages 49–63, 2016.

[56] L.-H. Loo, L. F. Wu, and S. J. Altschuler. Image-based multivariate profiling of drug responses from single cells. *Nature methods*, 4(5):445–453, 2007.

[57] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision*, 60(2):91–110, nov 2004.

[58] S. K. Lyman, S. C. Crawley, R. Gong, J. I. Adamkewicz, G. Mcgrath, J. Y. Chew, J. Choi, C. R. Holst, L. H. Goon, S. A. Detmer, M. E. Gerritsen, and R. A. Blake. High-Content , High-Throughput Analysis of Cell Cycle Perturbations Induced by the HSP90 Inhibitor XL888. *PLoS ONE*, 6(3), 2011.

[59] N. Malo, J. a. Hanley, S. Cerquozzi, J. Pelletier, and R. Nadon. Statistical practice in high-throughput screening data analysis. *Nature biotechnology*, 24(2):167–175, 2006.

[60] M. L. Mchugh. Lessons in biostatistics The Chi-square test of independence. *Biochemia Medica*, 23(2):143–149, 2013.

[61] F. Meyer. Topographic distance and watershed lines. *Signal Processing*, 38(1):113–125, 1994.

[62] O. Morozova and M. A. Marra. Genomics Applications of next-generation sequencing technologies in functional genomics. *Genomics*, 92:255–264, 2008.

[63] V. Nair and G. E. Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines. *International Conference on Machine Learning (ICML)*, (3), 2010.

[64] L. Najman and M. Schmitt. Geodesic Saliency of Watershed Contours and Hierarchical Segmentation. *IEEE Transaction On Pattern Analysis and Machine Intelligence*, 18(12):1163–1173, 1996.

[65] F. Nie, H. Huang, X. Cai, and C. Ding. Efficient and Robust Feature Selection via Joint. *Advances in Neural Information Processing Systems*, 23:1813–1821, 2010.

[66] N. Orlov, L. Shamir, T. Macura, J. Johnston, D. M. Eckley, and I. G. Goldberg. WND-CHARM: Multi-purpose image classification using compound image transforms. *Pattern Recognition Letters*, 29(11):1684–1693, 2008.

[67] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Systems, Man, and Cybernetics Society*, 9(1):62–66, jan 1979.

[68] S. J. Pan, X. Ni, J.-T. Sun, Q. Yang, and Z. Chen. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th international conference on World wide web*, pages 751–760. ACM, 2010.

[69] S. N. Pandeya and D. Thakkar. Combinatorial chemistry : A novel method in drug discovery and its application. *Indian Jounal of Chemistry*, 44(February):335–348, 2005.

[70] S. M. Paul, D. S. Mytelka, C. T. Dunwiddie, C. C. Persinger, B. H. Munos, S. R. Lindborg, and A. L. Schacht. How to improve R&D productivity: the pharmaceutical industry's grand challenge. *Nature reviews. Drug discovery*, 9(3):203–214, 2010.

[71] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, Others, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in {P}ython. *Journal of Machine Learning Research*, 12(Oct):2825—-2830, 2011.

[72] H. C. Peng, F. H. Long, and C. Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *Ieee Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.

[73] Z. E. Perlman, M. D. Slack, Y. Feng, T. J. Mitchison, L. F. Wu, and S. J. Altschuler. Multidimensional drug profiling by automated microscopy. *Science (New York, N.Y.)*, 306(5699):1194–1198, 2004.

[74] P. Rämö, R. Sacher, B. Snijder, B. Begemann, and L. Pelkmans. CellClassifier: Supervised learning of cellular phenotypes. *Bioinformatics*, 25(22):3028–3030, 2009.

[75] M. Ranzato, F.-J. Huang, Y.-L. Boureau, and Y. LeCun. Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition. In *Proc. Computer Vision and Pattern Recognition Conference (CVPR'07)*. IEEE Press, 2007.

[76] D. Scherer, A. Müller, and S. Behnke. Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition. In *Proceedings of the 20th International Conference on Artificial Neural Networks: Part III*, ICANN'10, pages 92–101, Berlin, Heidelberg, 2010. Springer-Verlag.

[77] N. J. Schork, A. B. Weder, and M. A. Schork. On the Asymmetry of Biological Frequency Distributions. *Genetic Epidemiology*, 446(7):427–446, 1990.

[78] P. Schwan. Lustre: Building a File System for 1,000-node Clusters. *Proceedings of the Linux Symposium*, page 9, 2003.

[79] X. Shi, Q. Liu, W. Fan, P. S. Yu, and R. Zhu. Transfer Learning on Heterogenous Feature Spaces via Spectral Transformation. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 1049–1054, dec 2010.

[80] C. Sima, S. Attoor, U. Brag-Neto, J. Lowey, E. Suh, and E. R. Dougherty. Impact of error estimation on feature selection. *Pattern Recognition*, 38(12):2472–2482, 2005.

[81] C. Sima, U. Braga-Neto, and E. R. Dougherty. Superior feature-set ranking for small samples using bolstered error estimation. *Bioinformatics*, 21(7):1046–1054, 2005.

[82] S. Singh, A. E. Carpenter, and A. Genovesio. Increasing the Content of High-Content Screening: An Overview. *Journal of biomolecular screening*, 19(5):640–650, 2014.

[83] Y. Song, L. Zhang, S. Chen, D. Ni, B. Li, Y. Zhou, B. Lei, and T. Wang. A deep learning based framework for accurate segmentation of cervical cytoplasm and nuclei. In *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*, pages 2903–2906, aug 2014.

[84] N. Srivastav, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout : A Simple Way to Prevent Neural Networks from Overfitting. *Jounal of Machine Learning Research*, 15:1929–1958, 2014.

[85] H. Su, Z. Yin, T. Kanade, and S. Huh. Phase contrast image restoration via dictionary representation of diffraction patterns. *Medical image computing and computer-assisted intervention : MICCAI ... International Conference on Medical Image Computing and Computer-Assisted Intervention*, 15:615–622, 2012.

[86] K. Subrahmanyam, N. S. Sankar, and S. P. Baggam. A Modified KS-test for Feature Selection. *IOSR Journal of Computer Engineering*, 13(3):73–79, 2013.

[87] D. L. Taylor and K. A. Giuliano. Multiplexed high content screening assays create a systems cell biology approach to drug discovery. *Drug Discovery Today: Technologies*, 2(2):149–154, 2005.

[88] D. L. Taylor, J. R. Haskins, and K. A. Giuliano. *High Content Screening: A Powerful Approach to Systems Cell Biology and Drug Discovery*. Human Press, 2007.

[89] U. The. Drug Discovery and Development. *Pharmaceutical Biology*, 33(s1):1, 1995.

[90] The HDF Group. Hierarchical Data Format, version 5.

[91] R. Tibshirani. Regression Selection and Shrinkage via the Lasso, 1994.

[92] R. P. Trevino, S. A. Kawamoto, T. J. Lamkin, and H. Liu. Cell Analytics in Compound Hit Selection of Bacterial Inhibitors. *In Proceedings of IEEE International Conference on Big Data.*, 2015.

[93] L. Vincent and P. Soille. Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. *IEEE Transaction On Pattern Analysis and Machine Intelligence*, 33(6), 1991.

[94] C. Wang and S. Mahadevan. Heterogeneous Domain Adaptation Using Manifold Alignment. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, IJCAI'11, pages 1541–1546. AAAI Press, 2011.

[95] J. Warner, J. Ramsbotham, E. Tunia, and J. J. Valdes. Analysis of the Threat of Genetically Modified Organisms for Biological Warfare. *Political Science*, (May), 2011.

[96] Z. Wen and W. Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1-2):397–434, 2010.

[97] Who. Antimicrobial Resistance Global Report on Surveillance 2014. *World Health Organization*, 2014.

[98] S. Wright. The Interpretation of Population Structure by F-Statistics with Special Regard to Systems of Mating. *Society for the Study of Evolution*, 19(3):395–420, 1965.

[99] Y. Yang and J. O. Pedersen. A comparative Study on Feature Selection in Text Categorization. *International Conference on Machine Learning (ICML)*, pages 412–420, 1997.

[100] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou. Regularized Discriminative Feature Selection for Unsupervised Learning. *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, 2:1589–1594, 2011.

[101] Z. Yin, T. Kanade, and M. Chen. Understanding the phase contrast optics to restore artifact-free microscopy images for segmentation. *Medical Image Analysis*, 16(5):1047–1062, 2012.

[102] D. W. Young, A. Bender, J. Hoyt, E. McWhinnie, C. Y. Chirn Gung-Wei adn Tao, J. A. Tallarico, M. Labow, J. L. Jenkins, T. J. Mitchison, and Y. Feng. Interegating high-content screening and ligand-target prediction to identify mechanism of action. *nature chemical biology*, 4(1), 2008.

[103] L. Yu and H. Liu. Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution. *International Conference on Machine Learning (ICML)*, pages 1–8, 2003.

[104] F. Zanella, J. B. Lorens, and W. Link. High content screening : seeing is believing. *Trends in Biotechnology*, 28(5):237–245, 2010.

[105] J. Zhang, J. Yu, J. Wan, and Z. Zeng. l2,1 Norm regularized fisher criterion for optimal feature selection{$}. *Neurocomputing*, 166:455–463, 2015.

[106] J.-h. Zhang, T. D. Y. Chung, and K. R. Oldenburg. A Simple Statistical Parameter for Use in Evaluation and Validation of High Throughput Screening Assays. *Jounal of Biomolecular Screening*, 4(2), 1999.

[107] X. Zhang, S. E. Wong, and F. C. Lightstone. Toward Fully Automated High Performance Computing Drug Discovery: A Massively Parallel Virtual Screening Pipeline for Docking and Molecular Mechanics/Generalized Born Surface Area Rescoring to Improve Enrichment. *Journal of Chemical Information and Modeling*, 54:324–327, 2014.

[108] X. D. Zhang. A pair of new statistical parameters for quality control in RNA interference high-throughput screening assays. *Genomics*, 89:552–561, 2007.

[109] Z. Zhao, F. Morstatter, S. Sharma, S. Alelyani, A. Anand, and H. Liu. Advancing Feature Selection Research. *ASU Feature Selection Repository Arizona State University*, pages 1–28, 2010.

APPENDIX A

RAW DATA IN MDME ANALYSIS

Table A.1: Syn_V1 dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Gaussian NB	**99.02**	89.78	89.79	89.51	89.79	96.83
Linear SVM	**77.02**	65.93	65.93	65.67	65.93	74.61
Random Forest	**97.94**	82.22	82.08	82.44	82.62	95.03
Decision Tree	**94.6**	86.66	86.58	86.22	86.8	93.57

Table A.2: Syn_V2 dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Gaussian NB	**71.94**	63.91	63.92	64.06	63.92	67.49
Linear SVM	**60.47**	57.05	57.05	56.65	57.05	59.13
Random Forest	**69.8**	61.7	61.84	61.59	61.77	65.72
Decision Tree	**69.46**	63.99	64.0	64.47	63.88	66.44

Table A.3: Arcene dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	**77.5**	73.5	72.5	70.0	72.5	68.0
Gaussian NB	**69.0**	66.5	66.5	65.5	66.5	65.5
Random Forest	**76.5**	67.5	68.0	67.0	66.5	62.0
Decision Tree	**77.5**	64.5	64.0	64.5	66.0	67.5

Table A.4: Gisette dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	**94.01**	93.76	93.76	93.76	93.76	93.86
Gaussian NB	88.33	88.01	88.01	88.06	88.01	**88.4**
Random Forest	**95.87**	95.67	95.33	95.29	95.39	95.31
Decision Tree	92.21	**92.69**	**92.69**	92.39	92.44	92.24

Table A.5: ALL AML dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	**98.61**	**98.61**	**98.61**	**98.61**	**98.61**	95.83
Gaussian NB	**95.83**	**95.83**	**95.83**	**95.83**	**95.83**	**95.83**
Random Forest	**98.61**	97.22	95.83	**98.61**	97.22	95.83
Decision Tree	93.06	90.28	91.67	**95.83**	90.28	90.28

Table A.6: Madelon dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	61.69	**61.85**	**61.85**	**61.85**	**61.85**	**61.85**
Gaussian NB	**61.77**	61.62	61.62	61.62	61.62	**61.77**
Random Forest	84.92	85.19	85.46	**85.73**	85.04	85.27
Decision Tree	78.5	78.58	78.5	**79.19**	78.35	78.58

Table A.7: SMK Can 187 dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	73.71	**76.46**	**76.46**	74.88	**76.46**	74.27
Gaussian NB	72.63	73.65	73.65	**73.68**	73.65	67.28
Random Forest	**75.41**	74.18	74.18	72.66	73.68	73.65
Decision Tree	71.43	69.3	70.44	**71.64**	69.44	67.28

Table A.8: Prostate GE dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	91.18	**92.27**	**92.27**	**92.27**	**92.27**	90.18
Gaussian NB	**93.09**	**93.09**	**93.09**	**93.09**	**93.09**	88.18
Random Forest	92.09	91.09	92.09	**93.18**	92.18	92.09
Decision Tree	85.27	86.36	85.36	**87.18**	86.27	86.27

Table A.9: High content screening plate 201100812 1 well dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	**76.77**	76.28	76.28	76.28	76.28	74.99
Gaussian NB	**73.39**	71.41	71.41	71.41	71.41	68.31
Random Forest	**75.84**	74.69	74.75	74.79	74.56	74.08
Decision Tree	**69.68**	68.63	68.62	68.49	68.59	67.7

Table A.10: High content screening plate 201100812 2 well dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	**78.66**	78.51	78.5	78.5	78.5	77.8
Gaussian NB	**75.86**	73.44	73.44	73.44	73.44	70.64
Random Forest	**77.9**	77.0	76.96	76.97	77.12	76.38
Decision Tree	**70.74**	69.96	69.92	69.97	70.02	69.33

Table A.11: High content screening plate 201104270 1 well dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	**77.65**	76.9	76.9	76.9	76.9	75.68
Gaussian NB	**77.18**	76.26	76.26	76.26	76.26	73.05
Random Forest	**77.35**	77.01	77.09	76.97	76.94	76.89
Decision Tree	**71.26**	70.68	70.48	70.5	70.43	69.97

Table A.12: High content screening plate 201104270 2 well dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	79.39	**79.56**	**79.56**	**79.56**	**79.56**	78.62
Gaussian NB	**77.5**	77.05	77.05	77.05	77.05	74.24
Random Forest	**78.51**	78.11	77.89	77.94	77.93	77.77
Decision Tree	71.6	71.66	71.72	**71.96**	71.62	70.67

Table A.13: High content screening plate 201104288 1 well dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	72.18	**72.39**	72.3	72.33	72.33	70.07
Gaussian NB	**69.79**	68.95	68.95	68.95	68.95	62.29
Random Forest	**71.65**	71.01	71.13	70.97	71.05	70.92
Decision Tree	**66.37**	66.1	66.27	66.29	66.07	65.61

Table A.14: High content screening plate 201104288 2 well dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	**74.53**	74.31	74.3	74.3	74.3	72.07
Gaussian NB	**70.97**	69.58	69.57	69.57	69.57	61.76
Random Forest	**74.13**	72.79	72.8	73.04	73.08	72.09
Decision Tree	**66.96**	65.94	66.07	66.29	66.09	64.85

Table A.15: High content screening plate 201101095 1 well dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	**79.66**	78.76	78.89	78.75	78.75	77.16
Gaussian NB	**77.98**	77.03	77.03	77.03	77.03	73.06
Random Forest	**79.04**	78.76	78.7	78.71	78.62	77.93
Decision Tree	**71.9**	71.79	71.66	71.74	71.66	71.04

Table A.16: High content screening plate 201101095 2 well dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	**81.58**	81.32	81.31	81.31	81.31	80.51
Gaussian NB	**78.68**	77.65	77.65	77.65	77.65	75.19
Random Forest	**80.3**	80.02	79.75	80.03	79.92	79.81
Decision Tree	**73.22**	72.57	72.53	72.61	72.52	72.41

Table A.17: High content screening plate 201101097 1 well dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	**76.18**	75.85	75.85	75.85	75.85	75.24
Gaussian NB	**75.93**	74.93	74.91	74.91	74.91	71.43
Random Forest	**76.39**	76.06	76.08	75.86	75.79	75.28
Decision Tree	**69.69**	69.35	69.28	69.27	69.29	68.23

Table A.18: High content screening plate 201101097 2 well dataset.

	MDME	mRMR	Fisher Score	T score	F score	Chi Square
Linear SVM	77.73	78.02	**78.06**	**78.06**	**78.06**	76.24
Gaussian NB	**76.41**	75.55	75.55	75.55	75.55	74.01
Random Forest	**77.55**	76.88	76.93	76.95	76.79	76.09
Decision Tree	**69.9**	69.53	69.69	69.59	69.52	68.55