Prediction of Liver Toxicity using Machine Learning to aid Drug Discovery

Master's thesis in Computer Science and Engineering

DANIEL BRUNNSÅKER

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2020

Master's thesis 2020

Prediction of Liver Toxicity using Machine Learning to aid Drug Discovery

DANIEL BRUNNSÅKER

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2020

DANIEL BRUNNSÅKER

Supervisor: Alexander Schliep, Department of Computer Science and Engineering
Advisors: Johanna Sagemark & Bino John, AstraZeneca CPSS Data Science & AI
Examiner: Graham Kemp, Department of Computer Science and Engineering

Master's Thesis 2020
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2020

DANIEL BRUNNSÅKER
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

This thesis proposes a method for predicting drug incuded liver injury using transcriptomic data from the toxicogenomical databases *TG-GATEs* (Toxicogenomics Project-Genomics Assisted Toxicity Evaluation System) and *DrugMatrix* with the help of various machine learning algorithms. The possibility of using the toxicological database *CMap* in cooperation with the *NCI60 human tumor cell lines screen* to make prediction models for *in vitro* cytotoxicity using the same methodology was also investigated.

It was found that transcriptomic data can indeed be used to predict liver injury in rat with very high accuracy. The *in silico* models developed in this project also outperform similar existing solutions on completely external testing sets, generating models successfully predicting four different injuries in the context of liver: necrosis, fibrosis, hyperplasia and mitotic alterations.

In vitro cytotoxicity was also predicted by the models with relatively high accuracy, more specifically on the cancer cell line A-549. The model was also evaluated on primary human hepatocytes exposed to hepatotoxic agents, finding dose-response relationships. Additional learnings included the importance of selecting appropriate featuresets when predicting specific adverse effects and also the applicability of synthetic oversampling techniques in collaboration with transfer-learning when used on transcriptomic data.

Keywords: Computer, science, computer science, engineering, thesis, data science, dili, deep learning, machine learning, cytotoxicity, connectivity map, tg-gates, drugmatrix, nci60, biotechnology, transcriptomics, bioinformatics.

Acknowledgements

Firstly, I would like to thank Bino John and Johanna Sagemark for discussing potential project topics with me and allowing me to do my Master's thesis at AstraZeneca. I am incredibly thankful for all of the help you have given my during the course of the project, be it technical in nature or just regarding general life advice. Secondly, I would also like to thank my supervisor, Alexander Schliep, for being there to help with me with all of the technicalities of doing a Master's thesis and also being available for advice whenever I needed it.

All of my friends and family have been of tremendous help, and this list of acknowledgements would go on forever if I would thank each one of you separately, but I think you know who you are and how much you mean to me.

Lastly, I would like to thank all the members of the CPSS Data Science & AI team at AstraZeneca. You took me in and made me feel part of the group almost instantaneously, and helped me out throughout the entire project whenever I needed it. I hope I at least get to work in a place half as good as this in the near future.

Daniel Brunnsåker, Gothenburg, April 2020

Contents

List of Figures

List of Tables

1
Introduction

1.1 Background

Modern drug development is a costly and time consuming process, potentially spanning up to a decade for a single drug. During this time, several billion dollars are spent continuously trying to drive the process forward. However, most drugs fail during this undertaking. Drugs that initially seem promising in terms of overall efficacy could be stopped due to even minor safety issues and side effects during toxicity evaluations. If this happens late during the development phase, billions of dollars are potentially wasted [1].

Organ toxicity is a major problem when applying and developing drugs [1]. If this problem could be detected at an earlier stage, it would streamline the process. In extension, this would allow for a quicker, safer and much more effective drug development phase. For example, when using drugs to regulate gene expressions, it can be extremely hard to predict possible adverse affects, as a simple modulation of gene expression could introduce a cascade of changes unrelated to the original modification. Therefore, a helpful tool for use in this problem would be an *in silico* model that could analyze these complex interactions and predict possible adverse effects.

Machine learning is a subfield of artificial intelligence, a field in which a model can make predictions without explicitly being programmed to do so. It does so by statistical analysis and pattern recognition, allowing the model to train on existing data, and in extension, using that knowledge to then make said predictions on previously unseen inputs. Depending on the quality and size of the data-sets fed to the model, it can be used to make very accurate predictions.

A lot of effort has been put into creating and maintaining toxicogenomical databases which contain gene expression data for various kinds of organisms treated with different compounds, in the form of for example, the Toxicogenomics Project-Genomics Assisted Toxicity Evaluation System, DrugMatrix and the Connectivity Map (See sections 1.2.1-1.2.4). With this plethora of data available, the conditions to create a machine learning model with high efficacy and predictive power have never been better.

1.1.1 Objective

This thesis aims to test various machine learning approaches to build gene expression-based signatures of a key human organ such as the liver, with data retrieved from toxicogenomical databases. This is then to be used for toxicity analysis, including prediction of pathological endpoints in drug induced liver injury and cytotoxicity. In extension, this model could potentially be used to predict harmful signatures of specific drug-treated *in vitro* and *in vivo* models.

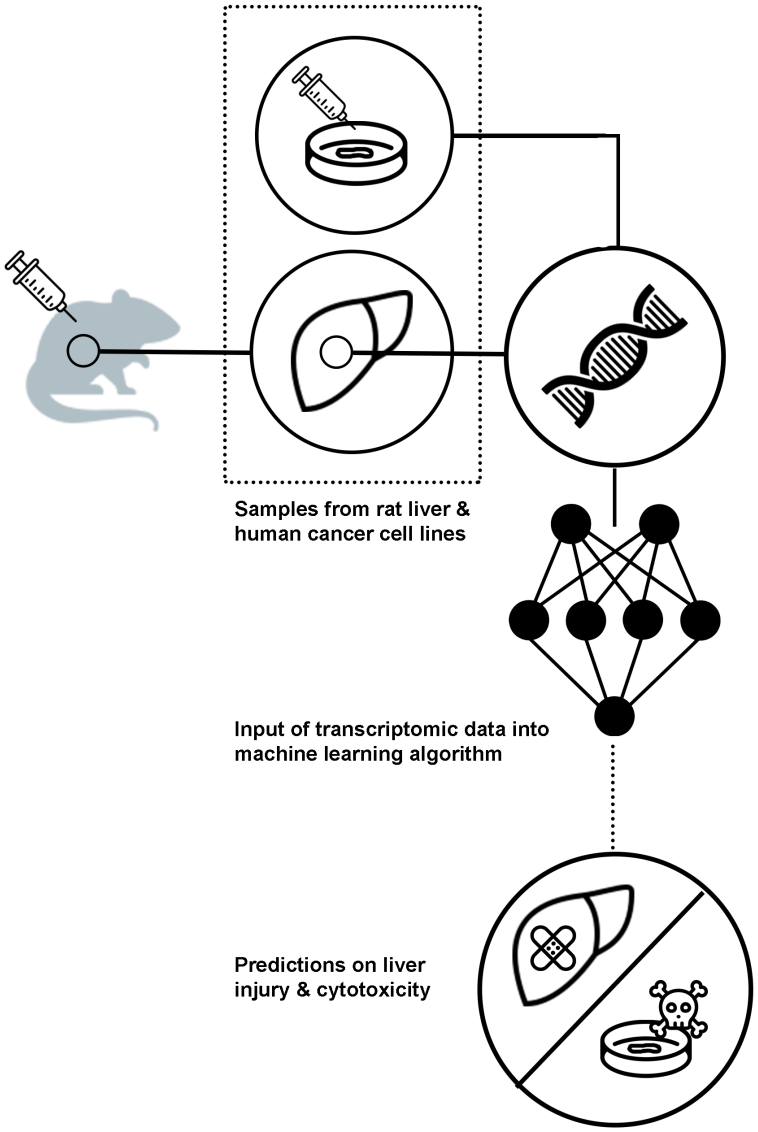


Figure 1.1: Overview of objective: Use of transcriptomic responses from immortalized cell lines[1] and rat liver exposed to chemical compounds for use in predicting cytotoxicity and liver injury using machine learning.

[1] A cell which, due to induced mutation, can proliferate endlessly.

1.2 Toxicogenomical Databases

This section introduces the different databases that enabled this project.

1.2.1 Open TG-GATEs

Open TG-GATEs (Toxicogenomics Project-Genomics Assisted Toxicity Evaluation System) is an extensive toxigenomic database, containing data on 158 different compounds, most of them reported as known hepatotoxicants (compounds harmful to liver) and nephrotoxicants (compounds harmful to kidney) [2]. This data is comprised of gene expression profiles, biochemistry, hematology and histopathological findings on three different platforms: rats, primary rat hepatocytes and primary human hepatocytes (specific liver-type cells). These in turn sampled with two different whole-genome type microarrays, to allow for complete measurements of gene response, for both rat and human studies respectively (see Section 2.2).

The database was developed during the course of ten years by the *Japanese Toxigenomics Project Consortium*, with the aim of providing a standardized and reliable source of data for use in drug safety assessments.

The *in vitro* part dataset is comprised of two different subcategories: studies performed in primary rat hepatocytes, measured in three different doses and times after initial exposure (2 hours, 8 hours and 24 hours). The same protocol was also applied to the primary human hepatocyte experiments [2].

The rat *in vivo* experiments are divided into two subcategories: single dose experiments, where the rats were sacrificed and sampled at four different timepoints (3 hours, 6 hours, 9 hours and 24 hours) after an initial dose. The other category is comprised of month-long studies, in which the rats were treated with compounds on a daily basis, sacrificed and in extension sampled at 4 different timepoints after initialization (3 days, 7 days, 14 days and 28 days). Both of these studies were also repeated for 3 different doses per compound [2]. A full schematic describing the database-structure can be seen in Figure 1.2.

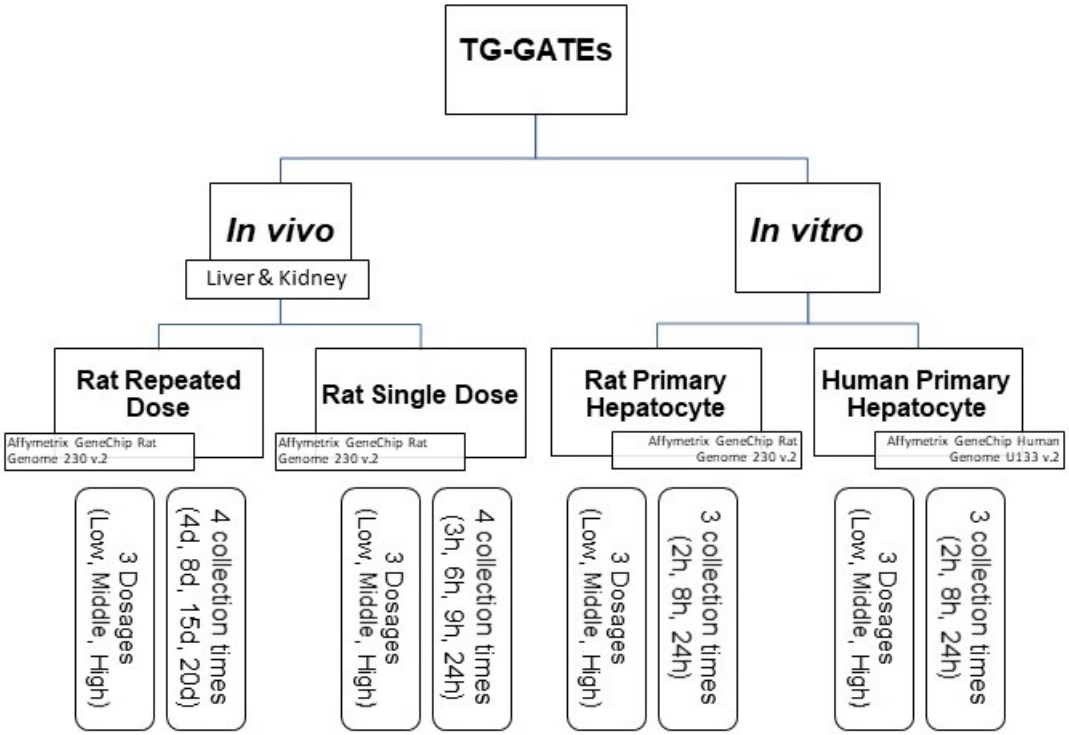


Figure 1.2: Overview of TG-GATEs database structure.

In conclusion, 6765 rats were studied, and the data available for each of subcategories are included in Table 1.1.

1.2.2 DrugMatrix

DrugMatrix (DM) is a large scale toxicogenomic database generated in 2006, with the intent of providing both *in vivo* and *in vitro* gene expression profiles for use with toxicological studies [3]. It is comprised of toxicological experiments on live rats and primary rat hepatocytes, in turn treated with over 600 different compounds. These compounds include among others therapeutic drugs and industrial chemicals [4].

The *in vivo* studies were comprised of samples taken at several different durations of exposure and doses. Extensive studies were performed in these experiments, which included pharmacology, clinical chemistry, hematology, histology, organ weights and clinical observations. The main part of the available data is the derived gene expression profiles from all of the aforementioned experiments, made possible by using Affymetrix whole-genome genechip arrays. All of these measurements are available on seven different types of tissues: liver, kidney, brain, heart, bone marrow, spleen and skeletal muscle [3].

The transcriptomic data in DrugMatrix roughly follows the outline explain in the previous section on TG-GATEs, but with a few modifications. The *in vivo* part of the study was performed with a repeated dosing model, in which the rats were subjected to a daily dosage, and in extension sacrificed and sampled at four different

times (6 hours, 24 hours, 3 days and 5 days). A minority of the compounds had experiments done for several different doses. The *in vitro* experiments had samples collected at 2 hours, 8 hours and 24 hours after the initial dose, with some compounds being repeated with different doses [3]. A full schematic describing the database-structure can be seen in Figure 1.3.

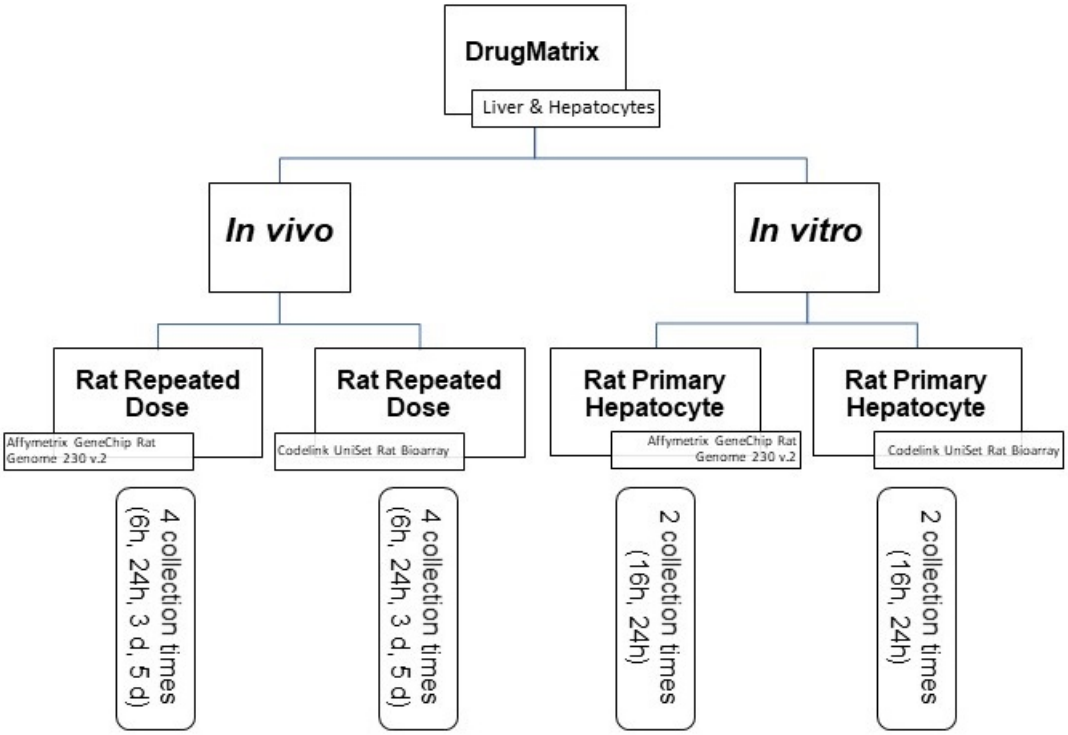


Figure 1.3: Overview of DrugMatrix database structure

2218 rats were studied, and a summary of the entirety of the gene expression samples available from the DrugMatrix database can be seen in Table 1.1.

Table 1.1: Number of microarray-derived gene expression samples and available compounds per study from the DrugMatrix and TG-GATEs toxicogenomic databases [2][3].

Database	Subset	Samples	Compounds
DrugMatrix	Rat Repeated Dose Studies	1719	299
DrugMatrix	Primary Rat Hepatocytes	7483	130
TG-GATEs	Primary Human Hepatocytes	2004	168
TG-GATEs	Primary Rat Hepatocytes	3139	168
TG-GATEs	Rat Single Dose Studies	6427	162
TG-GATEs	Rat Repeated Dose Studies	6283	162

1.2.3 Connectivity Map

The Connectivity Map (CMap) is a large project consisting of gene-expression profiles from cultured human cells treated with bioactive small molecules. It is intended to provide information regarding the different small molecules and to find similarities in mechanism of action and physiological processes between them. It was originally an aggregation of Affymetrix-microarray derived gene expression profiles from four different cell-lines (MCF7, PC3, HL60 and SKMEL5) treated with 164 distinct perturbagens for a total of 564 unique gene expression profiles [5].

Recently, due to a collaboration with the NIH LINCS Consortium, the number of gene expression profiles in the CMap database has vastly increased. This is made possible by a new high-throughput screening method called L1000. This expansion of the database has resulted in data on 54 different cell lines, known as the *CMap cell panel*, and now has almost 1.3 million gene expression profiles [6].

1.2.4 NCI-60 Human Tumor Cell Lines Screen

The NCI-60 database is a staple of modern cancer research. It was initialized in 1990, and has data on 60 different human tumor cell lines and their response to various compounds and small molecules during *in vitro* experiments. It was developed with the intent of providing researchers with a vast amount of information regarding the mechanisms of growth inhibition and lethality of compounds and small molecules when applied to a plethora of different cancer cell-lines in *in vitro* experiments [7].

The measured responses of use in this thesis are the different degrees of cell line growth inhibition and lethal concentration when exposed to certain compounds. It is therefore an invaluable resource for information regarding cytotoxicity of compounds in different cells and tissues, as many of the cell-lines and compounds used in this study overlaps with the CMap cell panel and available perturbations.

2
Theory

This chapter encompasses the mechanisms behind drug induced liver-injury and the computational methods used in this thesis: preprocessing techniques, methods for extracting differential expression and various machine learning methods.

2.1 Drug Induced Liver Injury

Drug induced liver injury (DILI) is a patient specific adverse effect of drug-intake. It depends on a multitude of different factors, and has yet to been properly summarized by a single model [8].

DILI is responsible for more than half of the known cases of acute liver failure, and this through various chemical causes. However, it is mainly divided into two categories: idiosyncratic DILI and intrinsic DILI. The latter acts through direct causes of hepatotoxicity and the former through mechanisms causing adaptive immune responses eventually leading to injury [8]. Due to this project mostly using short time-series of treatments (up to 28 days of treatment), the latter will be the most prevalent.

Drug induced liver injury can become apparent in a multitude of cellular perturbations, but this thesis will cover the mechanisms of the three main ones: mitochondrial impairment, biliary efflux impairment and oxidative stress.

2.1.1 Mitochondrial Impairment & Oxidative stress

Mitochondrial dysfunction is one of the main components of DILI, and as much as 60% of currently known hepatotoxic drugs are known to cause some form of mitochondrial impairment in a clinical setting [8]. However, these hepatotoxicants rarely cause significant direct mitochondrial damage, but can do so in combination with other extrinsic effects. Drug induced stress is most often combated with up-regulation of anti-oxidation factors such as NRF2 (Erythroid 2-related factor 2). Although this defense can be hindered due to the aforementioned extrinsic strains such as immune system malfunctions, infections, genetics and other environmental factors. This could end in mitochondrial impairment, sometimes leading to hepatocyte injury, causing secondary-cascade triggers of death signaling pathways such as c-jun Kinase (JNK) [9].

In the case of directly mitotoxic compounds, for example *Stavudine, Tamoxifen* and *Valproic Acid*, they alter mitochondrial function such as mitochondrial respiration and $\beta-$oxidation, which can cause membrane disruption. This can in extension induce hepatic necrosis or apoptosis, eventually triggering death-signaling pathways such as the aforementioned JNK [9].

On top of the loss of produced ATP due to damage because of the involvement of these aforementioned pathways, oxidative stress is highly correlated with mitochondrial impairment. The liver is the most important detoxifying organ in the body, involved in the metabolism of various compounds. This metabolization can in turn generate *reactive oxidative species* (ROS), a type of highly reactive molecule containing oxygen, otherwise known as a free radical [10]. Some types of drugs can increase accumulation of these molecules by inadvertently targeting and impairing the regulatory mechanisms such as the antioxidative system. When this balance is disturbed, and ROS are unregulated, they interact with electrons from neighboring molecules, causing a chain reaction. Possibly leading to further injury such as lipid peroxidation (reaction between the free radicals and cell membranes, causing severe cell damage), DNA-damage and further mitochondrial impairment [11].

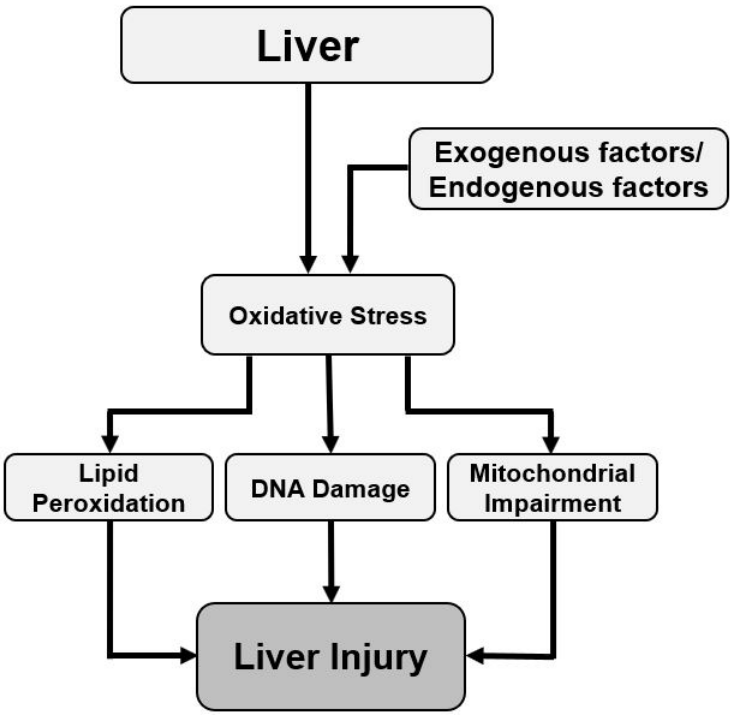


Figure 2.1: Overview of general mechanism of oxidative stress.

2.1.2 Biliary efflux impairment

Efflux proteins have an important role in drug metabolism, as they help with uptake clearance and excretion of drug-related compounds from hepatocytes and bloodstream into bile, forwarded through the bile ducts [9].

This is done primarily with the help from the so called ABC-superfamily of proteins, which are ATP-dependent transporters. Proteins belonging to this superfamily include, among others: multidrug-resistance proteins (MDR), multidrug-resistance associated proteins (MRP) and the bile salt export pump (BSEP). BSEP is entirely responsible for transportation of certain monoanionic drug derivates. As such, any inhibition, even mild, of this protein can cause repercussions in liver health and may lead to adverse pathological outcomes [12].

In fact, any functional perturbation of these transport proteins can cause adverse events, ranging from bile acid accumulation, cholestasis and even to severe liver injury [12].

2.1.3 Liver injury & histopathological findings

The aforementioned pathways of hepatocyte injury (and many more) can in turn cause severe injuries to the liver. These can present themselves in a variety of histopathologies, depending on the mechanism. Drug induced liver injury is commonly divided into one of 18 different patterns such as acute hepatitis, acute cholestasis and nodular regenerative hyperplasia. These patterns involve specific injuries such as, hepatocyte cell death (necrosis), scarring of the liver (fibrosis) and vastly increased cell proliferation in response to hepatocyte stress [13].

Quantification of injury is usually done by a pathologist, with the help of strict guidelines where patterns of injury and severity are standardized [14].

2.2 Microarrays

All of the transcriptomic data used in this study are collected by the means of microarrays. A microarray is a tool used to detect thousands of simultaneous gene expression levels. In essence, a microarray is a vast collection of different microscopic features, which can be probed with specific target molecules, leading to quantitative measurements of expression [15].

In this study, the microarrays in use are *in situ-synthesized oligonucleotide microarrays*. They make use of single stranded oligonucleotides as probes (with a length of 25 base pairs in this case), which are in turn directly synthesized onto the surface of the array. Essentially, this allows for high amount of features (genes) with an extremely high density of RNA-probes [15].

To perform the measurement, mRNA molecules are collected from the sample in question, and tagged with biotin. The biotin-tagged mRNA is then applied to the microarray, which allows the probes to attach to the target mRNA. The biotin then acts as a receiver for a fluorescent molecule. This molecule in turns allows for quantification of the amount of bonded RNA-molecules by way of fluroescence measurements [15].

2.2.1 Microarray Treatment

Microarrays have a lot of variation between measurements, so to allow for proper comparison between arrays, a lot of precautions and preprocessing steps need to be performed. This section will describe these methods, specifically, the *Robust Multichip Average Algorithm*, which is comprised of three different steps: *Background correction, Normalization and Summarization.*

Differential expression between microarrays will be encompassed in Chapter 3.

2.2.1.1 Background correction

The following step is performed due to the need to remove, for example, local artifacts and noise. The background correction process is based on the assumption that every real-valued signal/intensity is a combination of background intensity (B) and the actual signal intensity (S) described by the relation in Equations 2.1-2.3.

$$PM = S\ (Signal) + B\ (Background) \tag{2.1}$$
$$S \sim \exp \gamma \tag{2.2}$$
$$B \sim N(\mu, \sigma^2) \tag{2.3}$$

The background corrected intensity is then specified as the expected value of the actual intensity given the total intensity ($S + B$). The expected value of the PM-intensity in regards to the signal- and background noise can then be calculated using Equation 2.4 [16][17][18].

$$E\left(S|PM\right) = PM - \mu - \gamma\sigma^2 + \sigma\frac{\varphi\left(\frac{PM-\mu\gamma\sigma^2}{\sigma}\right) - \varphi\left(\frac{\mu+\gamma\sigma^2}{\sigma}\right)}{\phi\left(\frac{PM-\mu\gamma\sigma^2}{\sigma}\right) - \phi\left(\frac{\mu+\gamma\sigma^2}{\sigma}\right) - 1} \tag{2.4}$$

Where φ denotes the probability density function for $N(0,1)$ and ϕ the distribution function for $N(0,1)$.

2.2.1.2 Normalization

As previously mentioned, microarrays are prone to high variability between measurements, and as such, there will always exist discrepancies between the samples when comparing different microarrays. To solve this issue, this study makes use of quantile normalization. The objective of quantile normalization in this case is to force the intensities of the differing microarrays to adhere to the same distribution.

That is, if there exists n samples (s), the desired result is for the line given by the collection of unit vectors with length n, $\left(\frac{1}{s_1}, ... \ , \frac{1}{s_n}\right)$ to be in complete adherence with the, up to, nth-dimensional quantiles [16].

$$proj_d q_k = \left(\frac{1}{n}\sum_{j=1}^{n} q_{kj}, ..., \frac{1}{n}\sum_{j=1}^{n} q_{kj}\right) \tag{2.5}$$

The algorithm works with the following pattern [16]:

1. If n arrays of length p, form array X with the dimensions $p \times n$.
2. Sort the columns in X.
3. Take row-dependent means, assign mean to all elements in the row.
4. Rearrange X to original unsorted state.

After thorough iterations, the samples should now be more directly comparable.

2.2.1.3 Summarization

Summarization is the final step of the algorithm, it combines the background-adjusted and normalized intensity-values from all of the probes in the probe set to infer a single intensity for each gene.

The general methodology used in this project, namely *Tukey's median polishing*, is described as follows [19]:

1. Entire microarray normalized to median.
2. Each gene (all included probes) normalized to median.
3. Steps 1-2 are reiterated until the respective medians have converged to a similar value.

This methodology allows for a robust summarization, potentially lessening the impact of outliers.

2.3 Machine Learning Algorithms

This section includes the basic theory in the different machine learning algorithms used in this thesis, namely: *Support Vector Machines*, *Random Forest* and *Deep Neural Networks*.

2.3.1 Support Vector Machines

Support vector machines (SVM) is a type of supervised learning method, in that it uses labeled input data to produce input/output mappings. The type of output-mapping can either be a classification function, or a regression function. This section will only encompass the former.

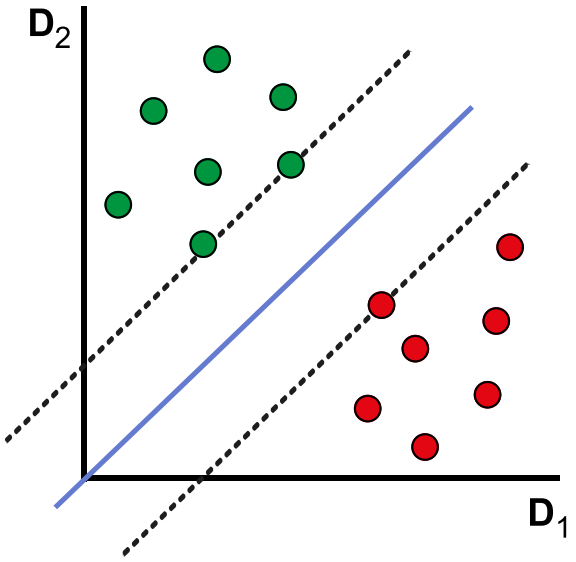


Figure 2.2: Overview of the type of hyperplane separation made by the support vector machine in 2 dimensions. The outmost lines denote the support vectors.

The SVM is capable of doing linear classifications and non-linear classifications on high-dimensional data. The latter performed through the use of a mathematical modification informally called the *kernel trick*, in which the input is remapped to a hyperspace in which the data can be linearly separated [20]. The type of kernel trick can differ between applications, but the kernel used in this thesis is the RBF kernel (Radial Basis Function). For a visualisation of the methodology in 2 dimensions, see Figure 2.2.

In general, the kernel has the following form [20]:

$$K(x, x') = \langle \psi(x), \psi(x') \rangle \tag{2.6}$$

ψ denotes a function that projects x and x' into another dimensional space. In the case of the RBF-function, the space being projected is in infinite dimensions, that is, $\psi_{RBF} : \mathbb{R}^n \to \mathbb{R}^\infty$.

The RBF-kernel has the following form:

$$K_{RBF}(x, x') = e^{-\gamma ||x - x'||^2} \tag{2.7}$$

This, in theory, allows for separation between classes without explicitly having to transform the data into an infinite dimension.

Support vector machines are very effective when the used data has more features than number of samples, which makes it a fitting methodology for this study.

2.3.2 Random Forest

Random forest makes use of decision trees. These are categorical structures based on nodes, branches and leaves. Each node represents a statement based on involved features, where in the branches are based on the answer to that specific question. These branches either lead to more nodes (i.e. more questions) or leaves (the categorical classification) [21].

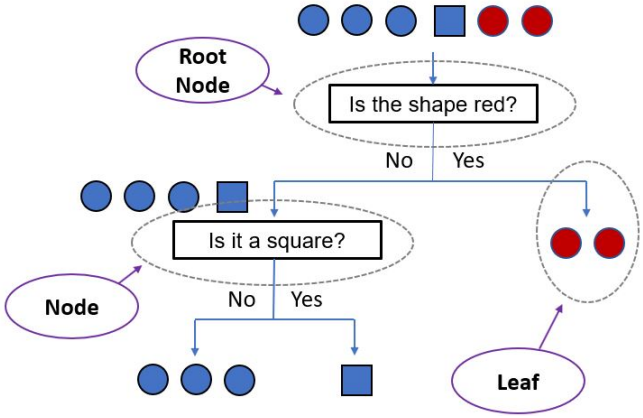


Figure 2.3: Generalized view of decision tree structure, showcasing the nomenclature and methodology.

When building the tree, one selects the feature that results in the lowest Gini impurity (the "question" which can separate the different samples most efficiently) as seen in Equation 2.8.

$$\sum_{i=1}^{C} = f_i(1 - f_i) \tag{2.8}$$

C being the different types of labels available for classification and f_i being the frequency of said classification.

It then calculates node importance using Equation 2.9 and sorts the nodes accordingly, with the root having the lowest impurity.

$$ni_j = w_j C_j - w_{left(j)} C_{left(j)} - w_{right(j)} C_{right(j)} \tag{2.9}$$

w denotes the weighted amount of samples.

This process is reiterated until branches no longer result in lower impurities, producing a decision tree classifier [21].

A random forest is a type of classifier that is a combination of a set of decision tree-predictors. The set will act as a voting committee and thus, by majority vote, perform a classification of samples. Each of these trees are trained in isolation from one another [21].

Decision trees are very sensitive to the input data, even minor changes can bring

along massive changes in the tree structure. This is used to the algorithms advantage; by randomly sampling (with replacement) the data set in question, it gives a new representation for the data set with every iteration. Due to sampling with replacement, only about two thirds of the dataset is used each time. This process is called bagging (bootstrap aggregation) [21].

As previously mentioned, when constructing a decision tree, one would usually build the tree using the features giving the highest degree of separation between the end-classifications. In random forest, one randomly samples sets of features to use, and subsequently sorts them according to their Gini impurity. This is repeated for all of the trees, using the remaining one third for validation [21].

Random forest algorithms usually perform very well on data sets with a small amount of samples and can handle imbalances in the data very well, making it very fitting for the data used in this thesis. However, in optimal settings, random forest does not usually, by itself, reach the performance of other algorithms.

2.3.3 Artificial Neural Networks

Artificial neural networks, and in extension deep neural networks, are popular computational algorithms used in various applications of machine learning. Note that in this project, only *feed forward* type neural networks were used.

The general structure of neural networks was initially inspired by the biological systems present in the brain. The system is based on interconnected sub-units called *artificial neurons*. These neurons work by interacting with each other, by processing signals from connected neurons and transmitting those to other neurons down the line. This collection of interconnected neurons can in unison create an extremely complex system, learning from their inputs, and in extension performs specific tasks, despite not being explicitly programmed to do so [22].

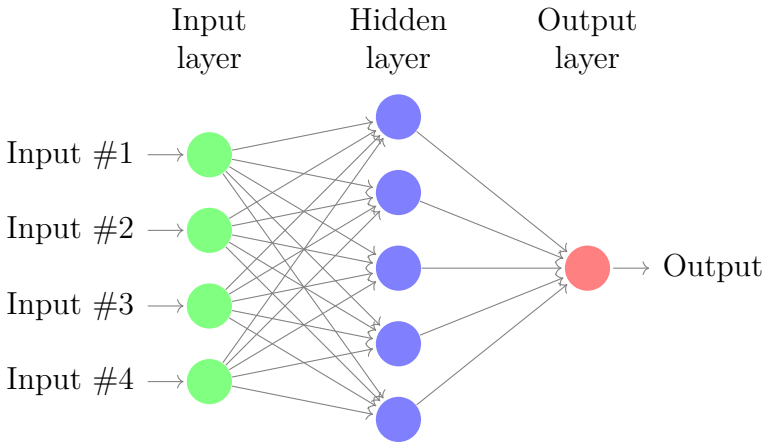


Figure 2.4: Image of typical neural network architecture. Each connection between neurons has an associated weight, defining behaviour.

The neural network makes use of the universal approximation theorem [22]. That is, there exists a set of parameters that can approximate any function. These sets of parameters (weights) are not initially known, however, the search for them can be formulated as an optimization problem.

The general procedure for training a neural network is outlined as following:

1. Randomly initialize weights.
2. Using the current settings, test samples from training set and compare output with actual result.
3. Modify weights.

Step 2 and 3 are iterated until a user defined threshold is reached.

Step 3 makes use of an algorithm called *backpropagation*, in which one uses the error of the output from the neural network to improve predictor performance by modifying the weights. This error is defined as the loss function and can take many different forms depending on the task. For binary classification problems, such as the one defined in this thesis, one can use *binary cross-entropy*, which is calculated from Equation 2.10 [23].

$$Loss = -ylog(p) - (1 - y)log(1 - p) \tag{2.10}$$

y denotes the actual binary classification of used sample, and p the probability of classification made by the model.

The backpropagation algorithm used in this thesis is the Adam optimizer (for more inforation, see *Adam: A Method for Stochastic Optimization* by Diederik Kingma and Jimmy Ba [24].)

Neural networks and in extension deep neural networks[1] work very well when there exists an abundance of data. They can, however, be very computationally expensive [22].

2.3.4 Synthetic Minority Oversampling Technique

When working with most supervised learning methods it is of utmost importance that there is an even distribution of categories in the training set; this might otherwise cause bias in the predictions [25].

The synthetic minority oversampling technique (SMOTE) is a method that utilizes random oversampling to even the distribution of categories, but with some synthetic modification. It produces these synthetic examples via Equation 2.11.

$$Synthetic\ Sample = x + u \cdot (x^R - x), \quad 0 \leq u \leq 1 \tag{2.11}$$

[1] A variation of a neural network with multiple layers inbetween input and output layers.

x specifies the targeted sample from the minority class, and x^R the classified nearest neighbour.

To summarize, the new synthetic samples are a linear combination of two similar samples from the minority-class. The relevant minority samples are classified and located by a K-nearest neighbour algorithm.

2.4 Evaluation metrics

When evaluating class-imbalanced datasets, one needs to consider the possibility that certain metrics might be unreliable. Below are some of the metrics used to evaluate algorithm performance in this thesis.

2.4.1 Confusion Matrix

The confusion matrix is a representative technique to evaluate performance in classification problems. It is used to classify the predictions made by the model into four different categories: True Positives (TP, an accurate positive prediction), True Negative (TN, an accurate negative prediction) and False Negatives/Positives (FN/FP, a misclassification) as seen in Figure 2.5.

This type of matrix can be very descriptive and allows for better troubleshooting as it allows for a more specific description of the classifiers performance. Depending on the application, the types of errors made might be more reasonable. It is also helpful in calculating other performance metrics, as seen in the next subsection.

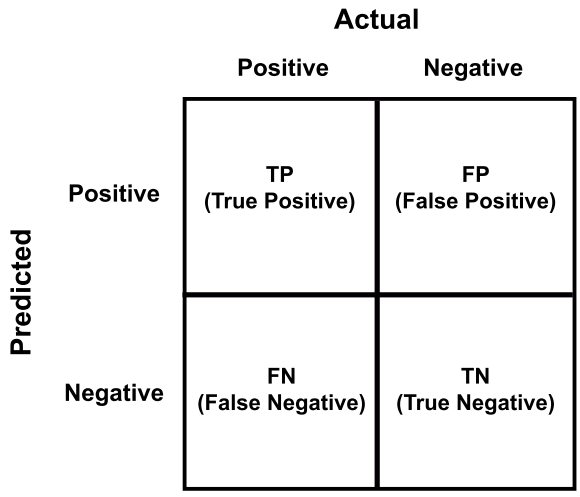


Figure 2.5: Vizualisation of a confusion matrix for binary classification.

2.4.2 Matthew Correlation Coefficient

In this study, all of the results will be measured by a metric called the *Matthew correlation coefficient* (MCC). This is widely used metric in binary classification problems. It is mostly used as an alternative metric when normal accuracy is incapable of showing an unbiased result, for example in class-imbalanced classification problems [26]. MCC can be defined as the discretization of the better known *Pearson correlation coefficient* seen in Equation 2.12.

$$r(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{2.12}$$

If we then define x and y as binary coefficients, further denoting them by their four potential outcomes, the following formula can be derived, as seen in Equation 2.13 [27].

$$r(x,y) = \frac{n \times TP - (TP + FN)(TP + FP)}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \tag{2.13}$$

Which in turn can be simplified to Equation 2.14.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{2.14}$$

The Matthew correlation coefficient can now, in the cases of binary outcomes, be calculated using a confusion matrix (see Section 2.4.1.)

To help with the actual interpretation of the metric, Table 2.1 can be used as reference.

Table 2.1: Interpretation of the Matthew correlation coefficient values [28].

Matthew Correlation Coefficient	Interpretation
0.70 to 1.0	Very strong positive relationship
0.40 to 0.69	Strong positive relationship
0.30 to 0.39	Moderate positive relationship
0.20 to 0.29	Weak positive relationship
0.01 to 0.19	No or negligible relationship
-0.01 to -0.19	No or negligible relationship
-0.20 to -0.29	Weak negative relationship
-0.30 to -0.39	Moderate negative relationship
-0.40 to -0.69	Strong negative relationship
-0.70 to -1.0	Very strong negative relationship

3

Differential Gene Expression

This chapter will describe how the data was processed and curated, but also how the differential gene expression analysis was performed, in turn producing the type of data used in this thesis.

3.1 Data Acquisition

The data in use are from the aforementioned databases (see Sections 1.2.1-1.2.3): TG-GATEs (Toxicogenomics Project-Genomics Assisted Toxicity Evaluation System), DrugMatrix and cMAP (Connectivity-Map). The entirety of DrugMatrix and TG-GATEs was accessed and downloaded via the query/download-package for ArrayExpress (v.1.46.0), integrated in the R/Bioconductor-project.

Pathology data from the TG-GATEs project was downloaded from their website (`https://toxico.nibiohn.go.jp/english/`). Since the DrugMatrix pathology repository is no longer publicly hosted, efforts were made to reach out to the current overseer of the database, Scott S. Auerbach.

Build 01 and 02 of the cMap data was downloaded from the online repository *Clue.io* (`https://portals.broadinstitute.org/cmap/`). It was downloaded completely preprocessed.

The NCI60 data was downloaded via the online tool $CellMiner^{TM}$ [29]. The molecular formulas of the compounds avaliable in the two CMap builds were used to query the NCI60 database.

Note that molecular formulas are not unique, further filtration was done later on.

3.2 Preprocessing & Outlier removal

The data was loaded using the R/Bioconductor package Oligo (v.1.50.0) [30]. An initial quality control of the data was performed, more specifically, inspecting metrics such as the maximum, minimum and median intensity values of each microarray, so as to ensure the absence of obvious outliers that would otherwise interfere with the normalization process. The data was then normalized and background-corrected using the RMA-algorithm included in the aforementioned Oligo-package (see Section 2.2.1).

Data was further curated using the R/Bioconductor package ArrayQualityMetrics (v.3.42.0). In extension, this means the usage of relative log expression of microrarray-intensities (see Figure 3.1), intensity heatmaps and Bland-Altman plots (MA-plots, a method in which the average measurement of two samples is plotted against the difference [31]). The latter was used to ensure that the background intensities of the different microarrays are similar, to ensure lack of strong batch-effects (see Figure 3.2).

PCA-analysis of the data was also performed; however, due to the high dimensionality of the data, it was difficult to make any definitive conclusions.

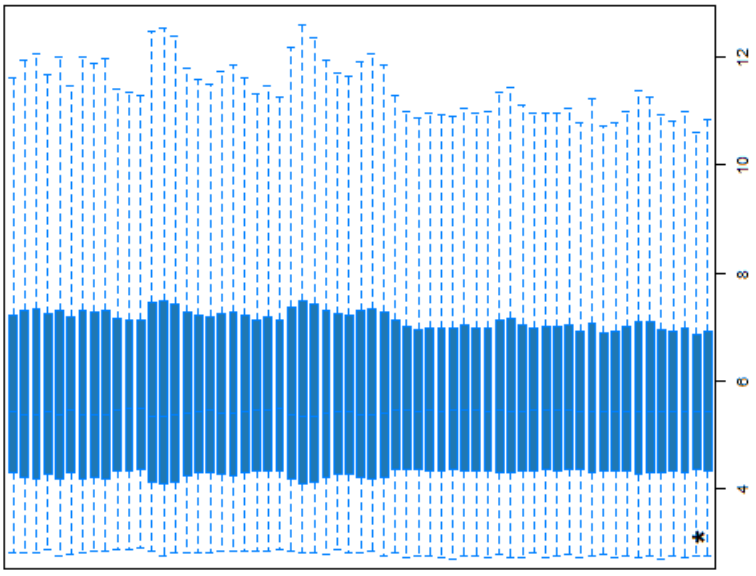


Figure 3.1: Slice of relative log intensities of the DrugMatrix *in vivo* dataset. Visualizes log2-intensity and variance in regards to the mean. The star indicates a probable outlier.

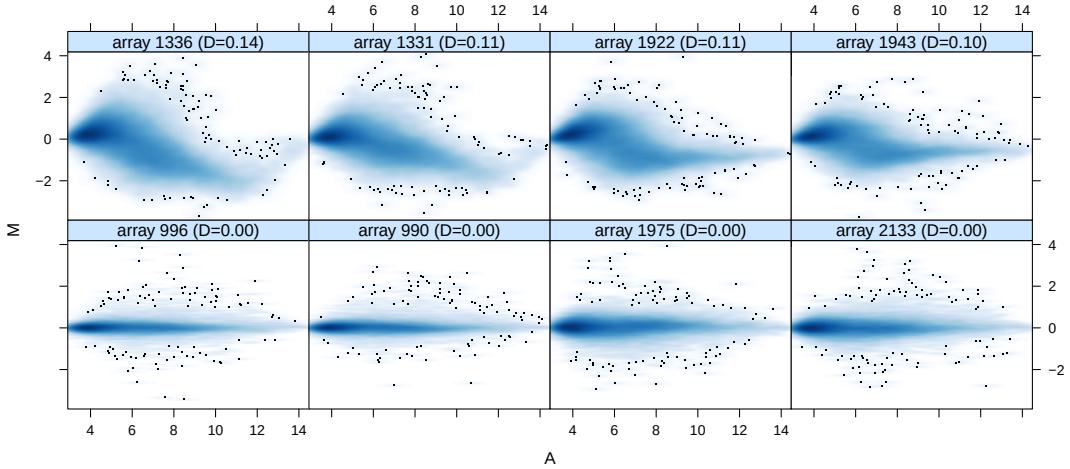


Figure 3.2: Examples of Bland-Altman plots of samples from the DrugMatrix *in vivo* dataset. The D-parameter indicates the discrepancy in background intensity between the specific array and the mean. The topmost figures would indicate probable outliers, as the difference between the measurements vary a lot from the average.

The next step of the process was to annotate the different transcript clusters present on the microarrays with their associated genes. This was done using the R/Bioconductor package AnnotationDbi (v.1.49.0), using the Arrays included ENSEMBL-identifiers[1]. If a cluster did not have an associated gene, it was removed from the dataset. If the transcript-clusters were mapped to several different genes (and therefore, cannot be unambigously assigned) they were removed. This process was done sequentially for all of the used microarray-based datasets: TG-GATEs *in vivo* rat liver, TG-GATEs human primary hepatocytes[2] and DrugMatrix *in vivo* rat liver.

The CMap data was downloaded as an already pre-processed version, as recommended by the publisher, *Broad Institute.*

Table 3.1: Overview of the amount of samples remaining after curation for TG-GATEs (TG-G) and DrugMatrix (DM).

Data items	DM (#)	DM (%)	TG-G (#)	TG-G (%)
Pre-curation				
Non-control samples	1931	100%	4289	100%
Post-curation				
Non-control samples	1883	97.5%	4263	99.3%
Post-annotation				
Non-control samples	857	44.4%	1354	31.7%

[1]A type of unique identifier used for biological data, containing information on species (rat or human in this case), object type (genes and proteins for example) and name.

[2]Primary cells are a type of cell which has been directly isolated from living tissue.

3.3 Extracting differential expression

In order to extract the differentially expressed genes (DEGs) for each compound (in relation to their control) a linear model has to be fitted to the data. This was done using the R/Bioconductor package *limma* (LInear Models for MicroArray data, v.3.43.0).

The compound-treated arrays were directly compared to their associated control measurements (i.e. control measurements from the same dose and collection time). These differential expression measurements were performed in two ways: Firstly, the experimental treatment groups were compared to their inherent controls (see Table 3.2 for design matrix). Lastly, each specific sample was compared to its experimental controls (see Table 3.3 for design matrix). After the model had been subsequently fitted using the design matrices, an empirical Bayes variance method was then applied to the results, resulting in moderated t-statistics, further improving the variance estimate. The resulting differential gene expressions (with their associated fold-changes and adjusted p-values) were then extracted and written to .csv-files, for use with implementations later on.

Note that for the data from TG-GATEs this means comparison with the controls in that specific experiment-series, and for DrugMatrix, an average of all of the controls using the same conditions and vehicle were used.

Table 3.2: Methodology 1, orthodox approach.

Treatment/ Sample	Treated	Control
$Sample_1$	1	0
$Sample_2$	1	0
$Sample_3$	1	0
$Control_1$	0	1
$Control_2$	0	1
$Control_3$	0	1

Table 3.3: Methodology 2, unorthodox approach.

Treatment/ Sample	Treated	Control
$Sample_1$	1	0
$Control_1$	0	1
$Control_2$	0	1
$Control_3$	0	1

The first methodology was used to supply foldchanges with accompanying p-values for other projects performed at AstraZeneca, whilst the second approach of doing a per-sample differential expression measurement was used for this project (see Section 4.2).

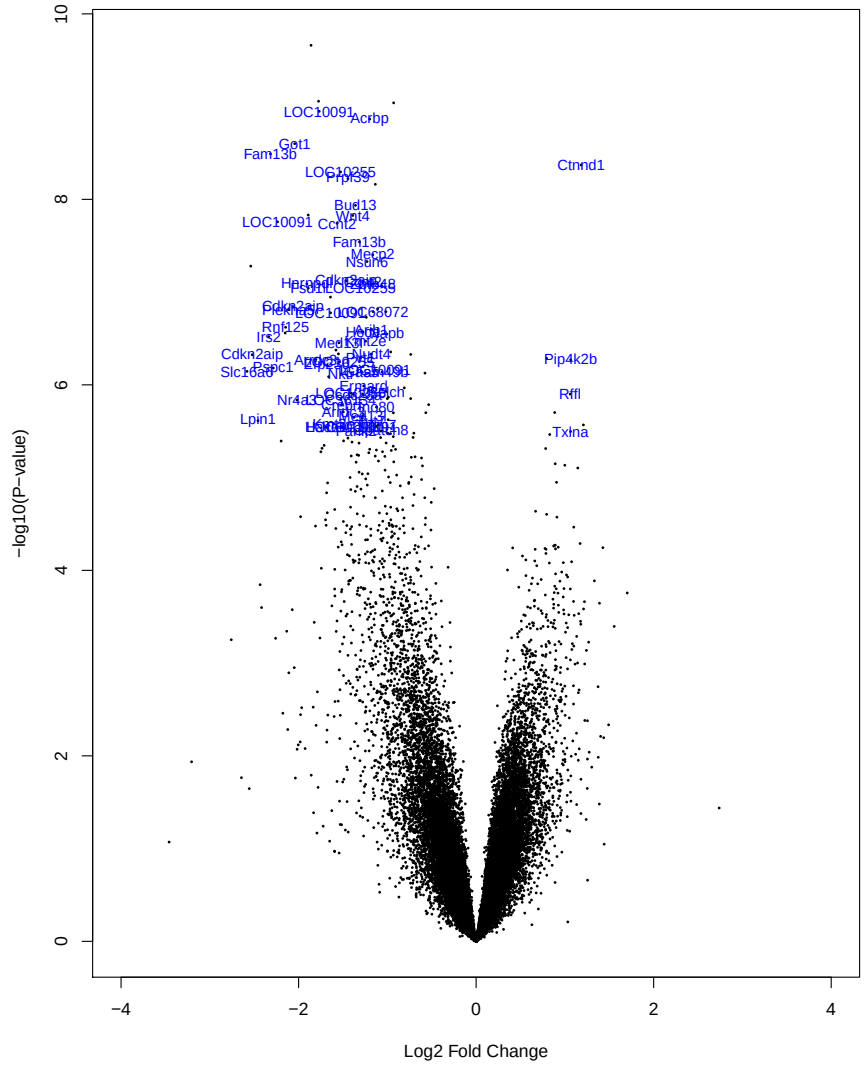


Figure 3.3: Volcano-plot showing example of differential expression in primary human hepatocytes exposed to a hepatotoxicant. The y-axis denotes the foldchange of said gene, whilst the x-axis denotes the increasing p-value.

As the CMap data had already been normalized and aggregated via the replicates, the differential expression was extracted by calculating the log2-ratio between the specific measurement and the average of the related controls. As there is a vastly differing amount of samples between compounds and doses in the CMap-dataset, only one measurement per dose, cell-line and compound was kept in the dataset. The aforementioned measurement was chosen by calculating the Euclidean distance (L2-norm) of the foldchange-vector, and then keeping the one with the median-most value. This in an effort to avoid compound-specific overfitting and potentially making the model more robust.

4
Annotation & Features

This chapter encompasses the methodology used in treating the data extracted from the previous chapter, features available for the samples and their annotations.

4.1 Feature selection

Due to the high dimensionality of the microarray data (around 15000 genes per array), selection of an appropriate subset becomes extremely important, as genes not causal to cytotoxicity and liver injury may provide the later models with non-trivial noise levels.

The different feature sets used in the study are the following:

1. 1331 PTGS-genes [32].
2. 299 core PTGS-genes, subset of PTGS deemed more relevant for DILI [32].
3. 551 core fitness genes [33].
4. 978 landmark genes from the L1000 platform [6].
5. 1000 randomized genes (not present in PTGS, L1000 or core fitness genes).
6. 500 randomized genes (not present in PTGS, L1000 or core fitness genes).

The PTGS-genes were chosen for their relevance for general cytotoxicity and drug induced liver injury [32]. Whilst the core fitness genes were chosen to see if genes critical for cell-survival, but not enriched in the liver [34], could be used for the same aforementioned prediction. The subset, L1000, was picked as it was determined to be a valid approximation of the whole genome according to the L1000-study [6]. Two other featuresets, comprised of 500 and 1000 randomized genes, are to be used as comparison, possibly denoting the importance of a relevant featureset.

Due to the different models of microarrays used in the studies, some of the genes contained in the aforementioned featuresets were absent from the lists of differentially expressed genes from the different types of compounds. When combining the data, the sets were reduced to the intersection between all of the different array-types, causing only a minor reduction in used genes as seen in Table 4.1.

Table 4.1: Number of genes available for analysis after intersection with microarrays.

Feature set	Number of genes	Available genes for analysis
PTGS	1331	1078
PTGS-DILI	299	255
Core	551	401
L1000	978	886
R1000	1000	1000
R500	500	500

To ensure that the feature sets were sufficiently dissimilar, efforts were made to analyze number of overlapping genes. See Figures 4.1 and 4.2.

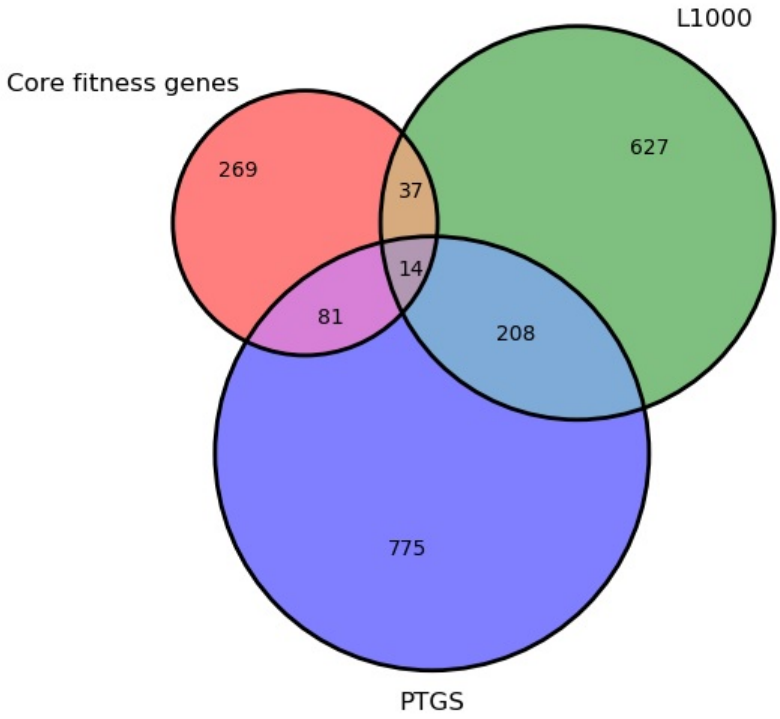


Figure 4.1: Venn-diagram showing intersections of genes between the different feature sets.

Note that the PTGS-DILI subgroup was not included in the Venn diagram as it is a part of the PTGS-space.

Albeit a relatively large overlap between L1000 and the PTGS-space, the subsets were deemed dissimilar enough for this study.

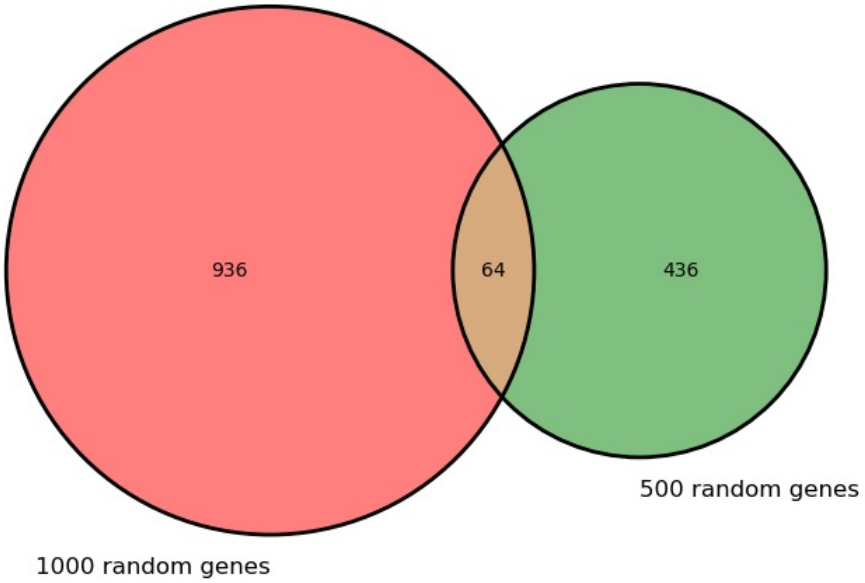


Figure 4.2: Venn diagram showing intersections of genes between the different feature sets.

The randomized feature-sets showed only a small overlap.

4.2 Annotation

4.2.1 *In vivo* liver injury

As the machine learning algorithms used in this project are fully supervised methods, there needs to be proper labeling for each of the samples. This was performed using the included histopathology data from TG-GATEs and DrugMatrix. The data was collected from `https://dbarchive.biosciencedbc.jp/en/open-tggates/desc.html` (TG-GATEs) and from personal correspondence with the current overseer of the DrugMatrix database (Scott S. Auerbach, Ph.D).

Histopathology findings were matched with their appropriate microarray-sample using experimental ID and sample ID and given a score associated with the presence of the finding (1 for presence of finding, 0 if not present). This being a slight modification of the method suggested by Grafström et al. [32].

Due to the nature of the reporting system for the different histopathology findings, downstream analysis was limited to necrosis, mitotic alterations, fibrosis and hyperplasia of the liver. A visualization of the selected pathological endpoints, properties and tissues are shown in Figure 4.3 for DrugMatrix and Figure 4.4 for TG-GATEs. As histopathology annotations adhered to different standards between the two datasets, nomenclature differences were reviewed manually.

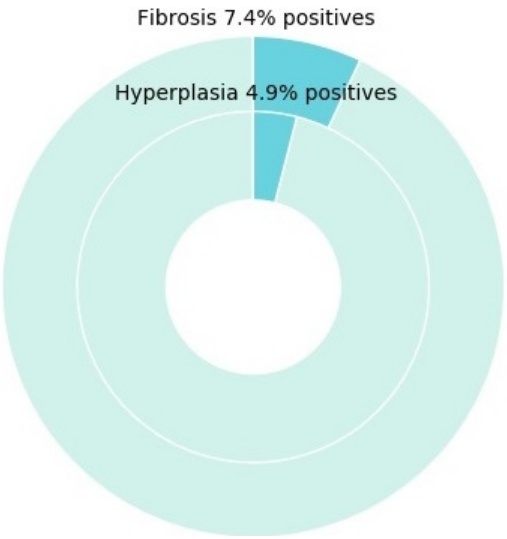
Samples without a related finding were removed from the dataset in an effort to improve the data distribution. For the more imbalanced categories (hyperplasia and fibrosis) the negative samples from TG-GATEs were removed. Final distributions of histopathology findings in the data can be seen in Table 4.2 and Figures 4.5a-4.5b.

Table 4.2: Table showing final amount of samples, label and their distribution for the entirety of the *in vivo* training set.

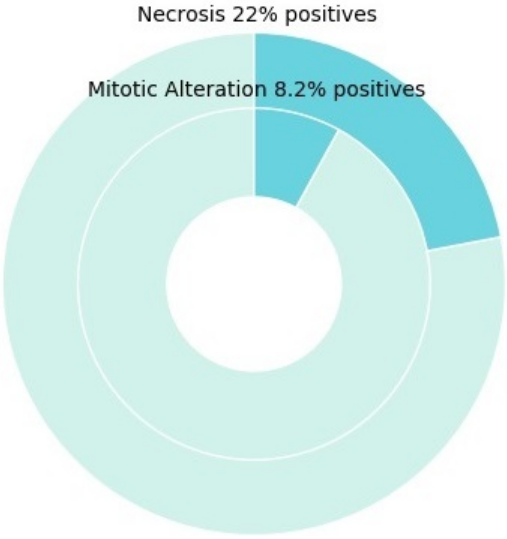
Histopathology	Positive samples	Negative samples	Fraction of positives
Necrosis	400	1812	0.22
Mitotic Alterations	168	2044	0.082
Hyperplasia	48	971	0.049
Fibrosis	70	949	0.074

Figure 4.3: Used pathological endpoints in DrugMatrix and their related properties.

Figure 4.4: Used pathological endpoints in TG-GATEs and their related properties.



(a) Class-distribution of fibrosis and hyperplasia.



(b) Class-distribution of necrosis and mitotic alterations.

Figure 4.5: Visualization of class-distribution for the different histopathology findings in the *in vivo* dataset.

4.2.2 *In vitro* cytotoxicity

The *in vitro* CMap-data was annotated with the help of the NCI60 human tumor cell line screen using the included GI_{50}-doses in combination with the doses included in the CMap experimental data.

4.2.3 Compound matching

As mentioned in Section 3.1, compounds from the NCI60-database were downloaded using the molecular formulas from CMap as the identifier. This due to the need for a broad search, minimizing the risk of lost datapoints. Do note however, that molecular formulas are not necessarily unique, and as such, the matches between the two datasets were further filtered. To perform this additional filtering, conversion tables from the $CellMiner^{TM}$ website were used. These tables contained the SMILES[1] for each of the compounds present in the NCI60-database. These were then converted to their inherent canonical SMILE[2]. In similar works, chemical names were used as the unique identifier [32][35]. This was deemed insufficient for this project, as there was too much variability in names and synonyms for the available compounds.

As each available chemical now had an *exact* match in the NCI60-database, each compound was assigned a GI_{50}-value (50 percent of maximal inhibition of cell proliferation) for each compound/cell-line combination. If several series of experiments were performed using the same parameters, the median of the GI_{50} values was used.

[1]Simplified Molecular-Input Line-Entry System, a line notation describing the chemical structure of said molecule.

[2]A canonical SMILE is an unique identifier, denoting exact structure and composition

4.2.4 Dose dependent toxicity

The labeling for each of the samples was annotated using dose dependent toxicity (DDT) as suggested by Grafström et al. [32]. The DDT was subsequently calculated by computing the sum of the logarithm of used dose minus the logarithm of the related GI_{50} dose, as seen in Equation 4.1 (note that the base-10 logarithm is used because of how GI_{50}-doses in the NCI60-database are presented):

$$DDT = log(Dose_{sample}) - log(Dose_{GI_{50}}) \quad \begin{cases} 1, & \text{if } DDT \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{4.1}$$

To specify, if the outcome was negative, that would mean that the dose is below the inhibitory concentration and as such, *non-toxic* using this form of annotation. This resulted in the number of available samples presented in Table 4.3.

Table 4.3: Amount of samples, compounds and cell lines available after intersection with NCI60.

Data items	CMap (Number)	CMap (%)
Pre-annotation		
Non-control samples	591697	100%
Cell lines	98	100%
Compounds	29667	100%
Post-annotation		
Non-control samples	4286	0.72%
Cell lines	6	6.1%
Compounds	355	1.2%

The intersection between NCI60 and CMap resulted in the cell lines and sample distributions seen in Table 4.4. A visualization of the distributions can be seen in Figure 4.6.

Table 4.4: Summary of included cell lines, amount of samples and short description.

Cell line	Samples	Pos. Samples	Neg. Samples	Description
PC-3	1251	326	925	Prostate cancer cell line
HT-29	1255	329	926	Colon cancer cell line
MCF-7	1295	396	899	Breast cancer cell line
A-549	459	161	298	Lung cancer cell line
SKMEL-28	26	13	13	Melanoma cell line

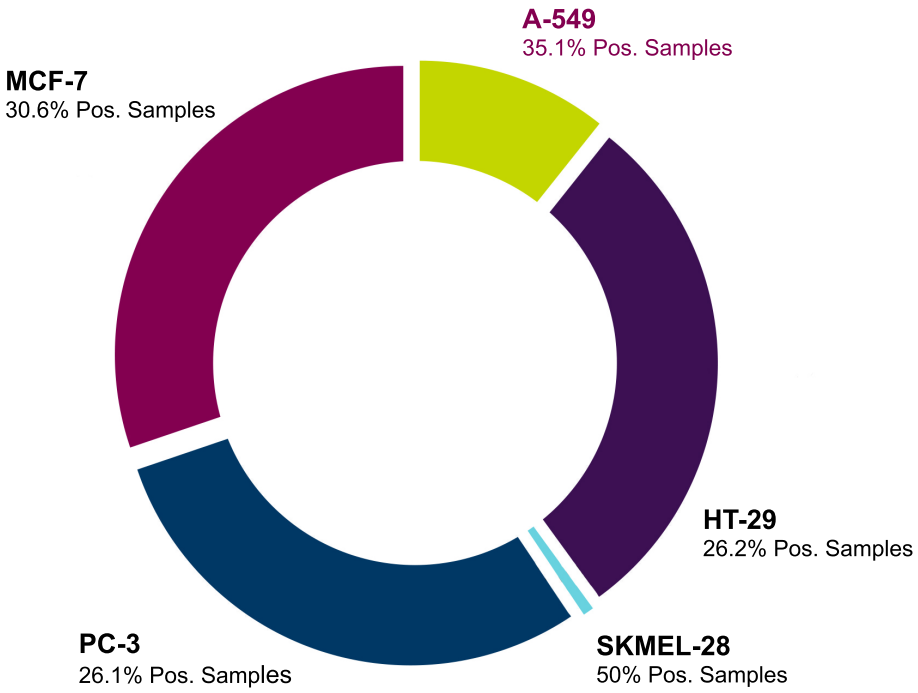


Figure 4.6: Visual representation of data-distribution from included CMap cell-lines. A-549 is highlighted due to being used as test later on.

5
Supervised Learning

This chapter will encompass the used methodology regarding selection of machine learning algorithms, optimization and validation.

5.1 Preprocessing

Each sample was matched with its annotation, and subsequently oversampled with the *SMOTE* algorithm (see Section 2.3.4). As per standard procedure, all of the data was scaled using the *Scikit-learn* StandardScaler. This methodology was used to remove potential variance and cardinality differences between the different integrated data sets.

5.2 Selected algorithms

The scaled data was then used as the input to the supervised learning models. The libraries and algortihms used in the project were:

- Python/Scikit-learn (v.0.22) for support vector machines (SVM) and Random Forest (RF).
- Tensorflow (v.1.13.1) for the design and implementation of deep neural networks (DNN).
- Imbalanced-learn (v.0.5.0) for the SMOTE-implementations.

The three different machine learning algorithms (RF, SVM, DNN) were chosen due to simplicity of implementation, interpretability or because of existence of comparative results using similar data in literature [32][35].

An additional modification of deep learning was also attempted, namely, transfer learning. More specifically, using the different histopathological categories as source and target domains.

5.3 Cross-validation

General performance was evaluated using K-fold cross-validation, in which we iteratively evaluate the model on a randomized majority split of the entirety of data, while simultaneously testing on the remaining minority. A visualization of the procedure can be seen in Table 5.1.

Table 5.1: Visualization of a 5K-fold cross-validation split. Fraction marked in grey is used to evaluate performance in each split.

Split	Dataset				
$Split_1$	Test	Train	Train	Train	Train
$Split_2$	Train	Test	Train	Train	Train
$Split_3$	Train	Train	Test	Train	Train
$Split_4$	Train	Train	Train	Test	Train
$Split_5$	Train	Train	Train	Train	Test

The cross-validation algorithm was designed as in the pseudo-code example in Algorithm 1. The scaler was trained on the training-part of the split to avoid any sort of data leakage. *Note that the fraction in this case denotes the size of different folds.*

Algorithm 1 Splitting and Evaluating

```
1:  procedure CrossValidation(Data, SplitFraction)
2:      TrainingData ← SplitFraction of Data
3:      TestData ← (1-SplitFraction) of Data
4:
5:      Scaler ← fit.Scaler(TrainingData)              ▷ Fit scaler & transform
6:      TrainingDataTransformed ← Scaler(TrainingData)
7:      TestDataTransformed ← Scaler(TestData)
8:
9:      model.fit(TrainingDataTransformed)      ▷ Train model with Training Data
10:     Predictions ← model.predict(TestDataTransformed)
11:
12:     Results ← EvaluationMetric(Predictions)        ▷ Save results from iteration
13:     goto top.
```

The results from each of the splits are aggregated and used to estimate performance of the model.

5.4 Gridsearching

For some of the models used in this thesis, especially support vector machines, hyperparameter tuning is very important [20]. This due to directly controlling the behaviour of the model, making it a possible make-or-break tweak. The optimal value for these parameters usually differ due to the nature of the problem in question, and as such, is also a necessary part of improving predictor performance and making them directly comparable.

Gridsearching in this study was kept simple due to time constraints and to avoid unnecessary complexity, and because of that, only two parameters were optimized at a time. This was performed by using an iterative approach, evaluating cross-validation scores for a selection of different hyperparameters. The best parameters from each model were then chosen to represent the model. This was done for every feature set and input, as to allow for a more fair comparison and optimize end-results. See Algorithm 2 for a general overview of the procedure.

Algorithm 2 Gridsearching for two hyperparameters

Require: $A_1 \dots A_N$, $B_1 \dots B_N$

```
for P1 ← A to B do
    for P2 ← C to D do
        Model ← Algorithm(Hyperparameter1 = P1, Hyperparameter2 = P2)
        Score ← CrossValidation(model, data)

        SavedScores ← CVScore, P1, P2
Evaluate SavedScores, Select best combination
```

In the case of deep neural networks, the same architecture and hyperparameters used in the study *Deep Neural Network Models for Predicting Chemically Induced Liver Toxicity Endpoints From Transcriptomic Responses* by Wang et al. were selected [35]. However, the optimization involved evaluating the amount of frozen layers and/or addition of trainable layers.

5.5 Model & feature set selection

The best performing models for the different feature sets, with optimized hyperparameters were compared to one another. Due to the stochastic nature of synthetic oversampling, neural networks and random forest, the training and evaluation process was re-iterated 10 times. The mean and standard deviation was then calculated, to be used as the metric for comparison between the models.

In the case of *in vivo* liver injury, the best performing models for each *in vivo* histopathology were then aggregated in a wrapper, producing four different predictions based on the same input, as illustrated in Figure 5.1. This was then used for final evaluation on an external test-set.

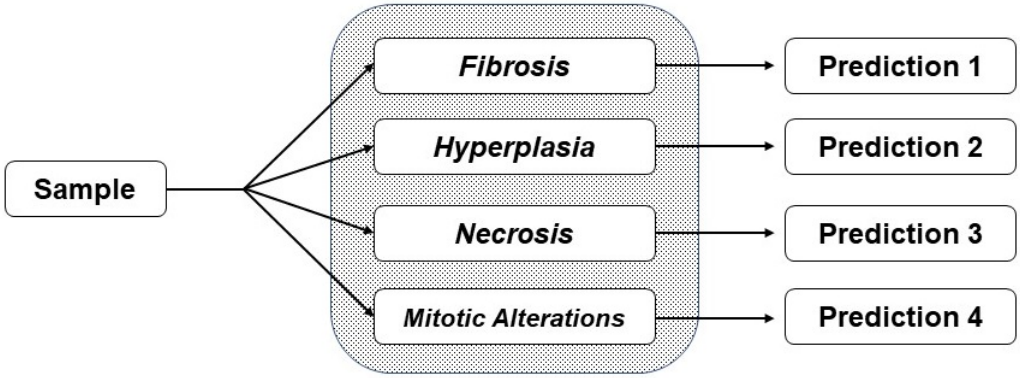


Figure 5.1: Visualization of the *in vivo* aggregated-model. A sample input will yield four different predictions, one for each of the histopathologies.

5.6 Testing procedure

When assessing performance of a model, one traditionally uses a train-test split. However, due to availability of external studies measuring similar responses, the models were assessed using data not related to the one used in training.

5.6.1 *In vitro*

The cell line A-549 was removed from the training data and used separately as a test set (see Figure 4.6). This was done as it was deemed more descriptive of performance in a foreign cell line than a standard train-test split.

The best-performing model based on cross-validation was used and the test-data was scaled using the pretrained scaler.

As another means of validation, the trained model was also applied on the TG-GATEs Primary Human Hepatocytes (see Section 1.2.1), using the different dosages and timestamps as unique samples. This was done to see if the model could capture dose-response relationships, similar to predictions made in similar studies [32]. Toxicity is dependent on the dose, and ideally the model should capture toxic responses in relation to an incremental increase of said dose [32]. Results can be seen in Section 6.3.

5.6.2 *In vivo*

Two completely external test-sets were used to evaluate performance for the histopathology predictive models. The studies chosen for this task were selected due to their relevance regarding histopathology and treatment procedures, but also due to availability of sample-based histopathology findings [36][37].

The first dataset is a study wherein rats of the species *Rattus Norvegicus* had been treated with four different hepatotoxic compounds and sampled with whole-genome microarrays [36]. The compounds present in this set were removed from the training data. Sample distribution can be seen in Table 5.2.

Table 5.2: Overview of available transcriptomic samples with related histopathology for study by Ippolito et al.

Histopathology	Number of afflicted rats	Number of unafflicted rats
Necrosis	31	46
Fibrosis	19	58
Hyperplasia	20	57
Mitotic Alterations	42	35

The second study is a study on the effects of bile-duct ligation[1] and the transcriptomic changes due to said treatment [37]. The transcriptomic data was sampled via microrrays. This specific study was also chosen due to test model-performance on non-chemical based treatments. Amount of samples available can be seen in Table 5.3.

Table 5.3: Overview of available transcriptomic samples with related histopathology for study by Sutherland et al.

Histopathology	Number of afflicted rats	Number of unafflicted rats
Necrosis	24	0
Fibrosis	14	10
Hyperplasia	16	8
Mitotic Alterations	2	22

The differential gene expression from these databases was extracted using the same methodology described in Chapter 3.

5.6.2.1 Y-scrambling

The model and feature set combinations were also evaluated using Y-label randomization (or Y-scrambling for short). This is a methodology in which one randomly shuffles the sample labels and then consequently trains the model. This is done to ensure the model is not currently overfitting to the noise in the model [38]. Care was taken to ensure that the shuffle was sufficiently different from the original ordered state by calculating the Jaccard index[2]. The model was then evaluated and retrained 10 times.

[1] A procedure in which the rats are the subjects of an operation designed to induce liver fibrosis.
[2] A metric for evaluating similarity between two sets of samples.

6
Cytotoxicity Predictions

This chapter contains the results produced by one of the models developed for this thesis, more specifically regarding cytotoxicity. It describes predictions made on general *in vitro* dose dependent cytotoxicity in immortalized human cancer cells and human primary hepatocytes.

6.1 Cross-validation & Model Selection

The models were optimized using basic gridsearching, as described in Section 5.4.

The different featuresets were evaluated using only support vector machines and random forest algorithms, with the subsequent help of 10 K-fold cross-validation. Only the best performing models for each hyperparameter-combination are represented. Shown in Figure 6.1 are the results, bar height denoting the average score from the cross-validation process, while the error bar denotes the standard deviation.

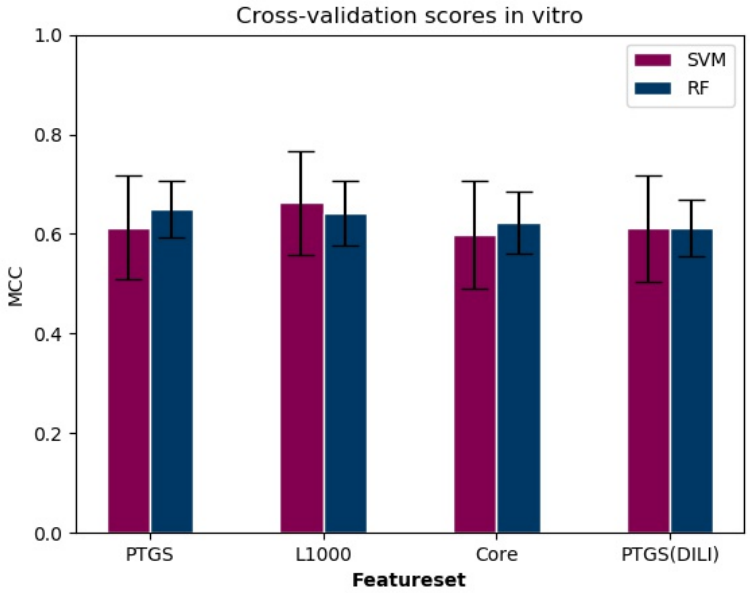


Figure 6.1: Cross-validation scores using the different subsets with optimized SVM and Random Forest algorithms.

Note that the high variance is due to the imbalanced nature of the dataset, as the temporary fraction used as the test is ***not*** *oversampled via SMOTE, causing large discrepancies in classifier performance depending on the fold.*

The performance difference between algorithms seem negligible. With differences between the worst and best performing combinations showing discrepancies less than 0.03 MCC. In general, random forest seems to perform the best. However, L1000, using a support vector machine algorithm had the highest mean performance, and as such was selected for further testing.

6.2 A-549 Cytotoxicity

In the interest of attaining a more valid description of the importance of a curated featureset, all of the above models were applied to the leave-one-out set consisting of transcriptomic data on the cell line A-549. Results can be seen in Figure 6.2.

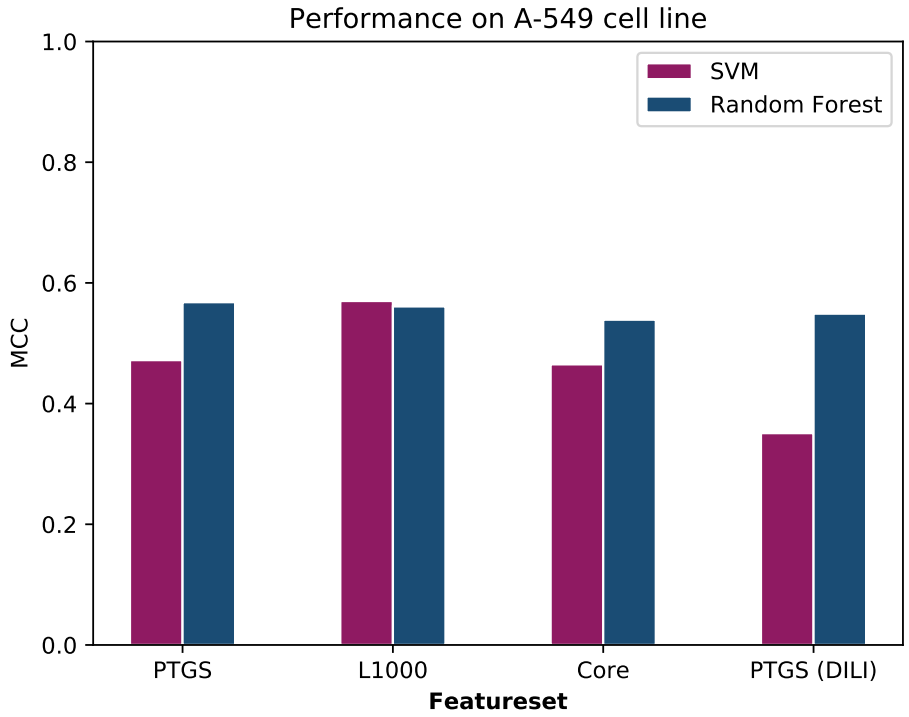


Figure 6.2: Scores regarding cytotoxicity predictions on cell line A-549 for the different featuresets and algorithms.

When applied to the cell line A-549, the same general trend of random forest being the better performer is still valid, but with a much larger discrepancy in performance, especially for the PTGS, core-fitness genes and the PTGS-DILI feaure sets. However, the support vector machine-model using the L1000 genes still manage to outperform the other models, further showcasing that the right selection was made during cross-validation.

6.3 Primary Human Hepatocytes

As mentioned in Section 5.6.1, the model was evaluated on the transcriptomic data from the TG-GATEs primary human hepatocytes. These were performed using the best performing model shown in Figure 6.1. Out of all the data available, 10 compounds were randomly selected. Predictions were made on all the different dosages and timepoints of the included compounds, shown in Figures 6.4 and 6.5. A summarization of the predictions for each of the different compounds is visualized in Figure 6.3.

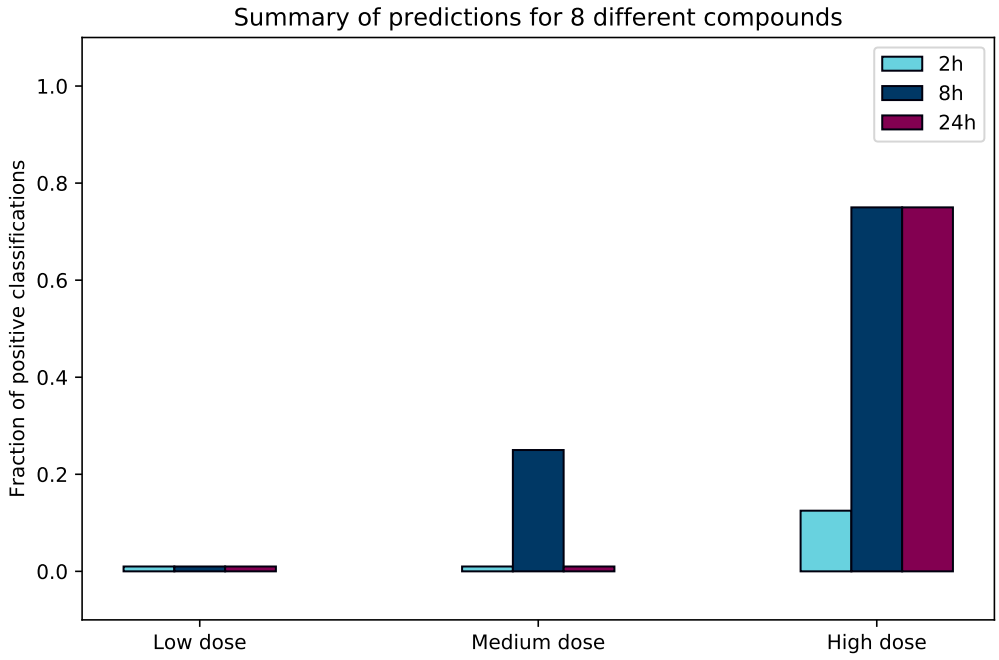
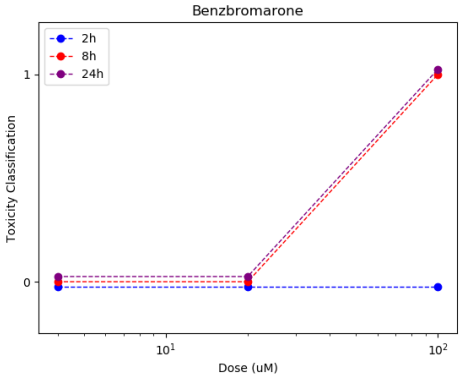


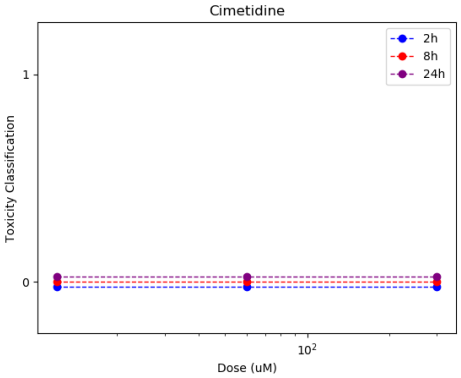
Figure 6.3: Bar plot showing fraction of positive cytotoxicity classifications in increasing dosages and collection times.

The model seemingly captures the relationship between increased dosages and increased likelihood of cytotoxicity, with an increasing fraction of positive classifications, providing further validation that the model captures relevant features in dose-based transcriptomic profiles. Due to the randomness of the selected compounds, they lack the same pharmocokinetic curves (time dependent uptake and metabolization in the recipient). However, an assumption is usually made that 6-8 hours after treatment is a good reference for transcriptomic changes [2][3][6].

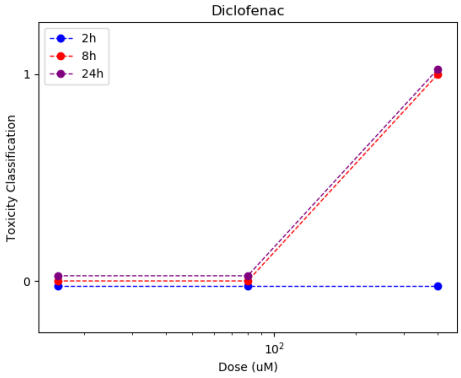
Below, in Figures 6.4-6.5 are all of the predictions based on the drug-treated human primary hepatocytes:



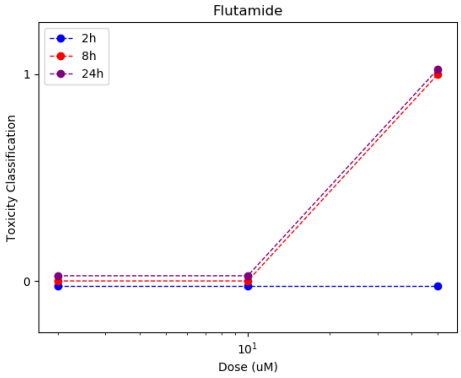
(a) Cytotoxicity predictions on primary hepatocytes exposed to benzbromarone.



(b) Cytotoxicity predictions on primary hepatocytes exposed to cimetidine.



(c) Cytotoxicity predictions on primary hepatocytes exposed to diclofenac.

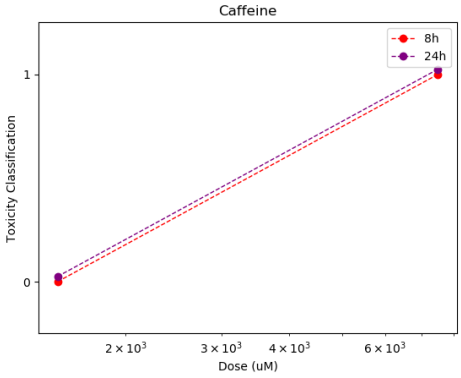


(d) Cytotoxicity predictions on primary hepatocytes exposed to flutamide.

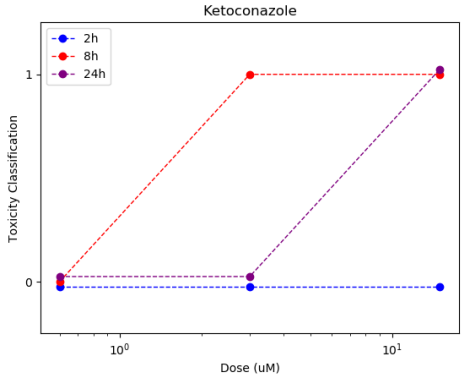
Figure 6.4: Dose- and timebased cytotoxicity predictions on four primary hepatocytes exposed to four different componds: *benzbromarone, cimetidine, diclofenac and flutamide.*

The predictions made on the four compounds seen above seem to correlate somewhat with literature. Where *benzbromarone, diclofenac* and *flutamide* are regarded as drugs with high concern of liver injury and *cimetidine* is regarded as a lesser concern. This according to the largest reference database of DILI-causing compounds, *DILIRank* [39].

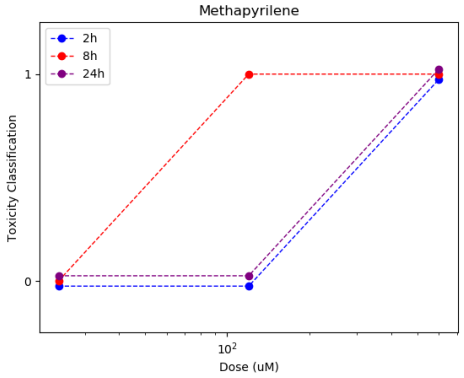
Note that DILIRank does not provide pharmacokinetic information regarding the toxicity.



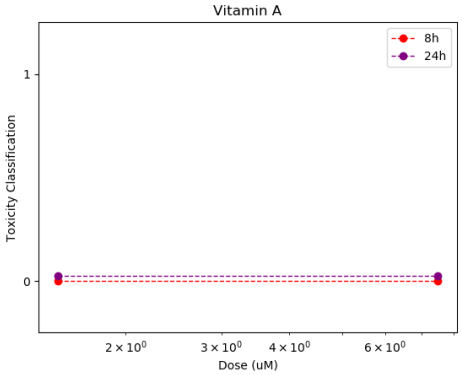
(a) Cytotoxicity predictions on primary hepatocytes exposed to caffeine.



(b) Cytotoxicity predictions on primary hepatocytes exposed to ketoconazole.



(c) Cytotoxicity predictions on primary hepatocytes exposed to methapyrilene.



(d) Cytotoxicity predictions on primary hepatocytes exposed to vitamin A.

Figure 6.5: Dose- and timebased cytotoxicity predictions on human hepatocytes exposed to four different componds: *caffeine, ketoconazole, methapyrilene and vitamin A.*

According to the model, all of the compounds seemingly shows cytotoxic profiles at higher doses with the exception of vitamin A, which is of lesser liver injury concern [39]. Methapyrilene and ketoconazole toxicity also correlates with results in literature [39][40]. However, caffeine is not classified in literature as a hepatotoxicant [39].

6.4 Secondary Validation

In order to assess the selection of feature sets and avoidance of overfitted models, models trained using randomized feature sets and scrambled labels were evaluated.

6.4.1 Randomized featuresets

Randomized featuresets were used as control, to estimate the performance of a well-curated featureset. The results from cross-validation and two external tests are shown in Table 6.1.

Table 6.1: Results from cross-validation and test performance on the cell line A-549.

Featureset	Prediction	Cross-validation	A-549
R1000	Cytotoxicity	0.43 ± 0.076	0.39
R500	Cytotoxicity	0.32 ± 0.083	0.32

The randomized featuresets seemingly still allow for a semblance of prediction performance. However, at a great loss when compared to the curated featuresets presented in Sections 6.1 and 6.2.

6.4.2 Y-scrambling

Y-scrambled models showed no predictive power in cross-validation and on external performance when evaluated on the cancer cell line A-549.

Table 6.2: Results of Y-shuffled cytotoxicity model presented in MCC. Note that they are an average of 10 different models.

Prediction	CV (μ)	CV (σ)	A-549 (μ)	A-549 (σ)
Cytotoxicity	0.012	0.00011	-0.056	0.0078

7
Liver-Injury Predictions

The following chapter will summarize the results of the histopathology-predictive models generated in this thesis. These results will include scores based on both cross-validation and on external test sets. It will also showcase an example of a future application.

7.1 Cross-validation & Model selection

All of the following models were optimized using basic gridsearching, as described in Section 5.4.

Attempts were made at assessing the potential applicability of transfer-learning, using different pathological findings as source/target-domains. In Table 7.1 are cross-validation results using the PTGS-subset (see Section 4.1).

Table 7.1: Results of cross-validation using different source/target-domain combinations with a deep learning architecture. Cells marked in grey are classical approaches, i.e. networks trained only on that input.

Target/ Source	Fibrosis	Necrosis	Hyperplasia
Fibrosis	0.609 ± 0.226	0.104 ± 0.162	0.631 ± 0.186
Necrosis	0.629 ± 0.240	0.322 ± 0.190	0.618 ± 0.232
Hyperplasia	0.572 ± 0.096	0.133 ± 0.117	0.560 ± 0.251

Two of the combinations showed better results than a classical approach, namely, fibrosis to hyperplasia, necrosis to fibrosis and necrosis to hyperplasia. Out of those, the first two were chosen as the models going further. Random Forest was chosen for the other histopathologies, as deep learning showed poor performance in other studies [35]. Support vector machines were omitted in this part of the study due to the need for classification probabilities, a feature which that algorithm usually lacks (see Section 2.3.1).

The models of interest are presented in Table 7.2.

Table 7.2: Summary of the selected model for each of the histopathological findings.

Histopathology	Algorithm
Necrosis	Random Forest
Mitotic alterations	Random Forest
Fibrosis	Deep Neural Network (Transfer-based)
Hyperplasia	Deep Neural Network (Transfer-based)

The best combinations of hyperparameters for each of the feature sets and selected models were compared to each other via their cross-validation results. Scores are presented in Figure 7.1.

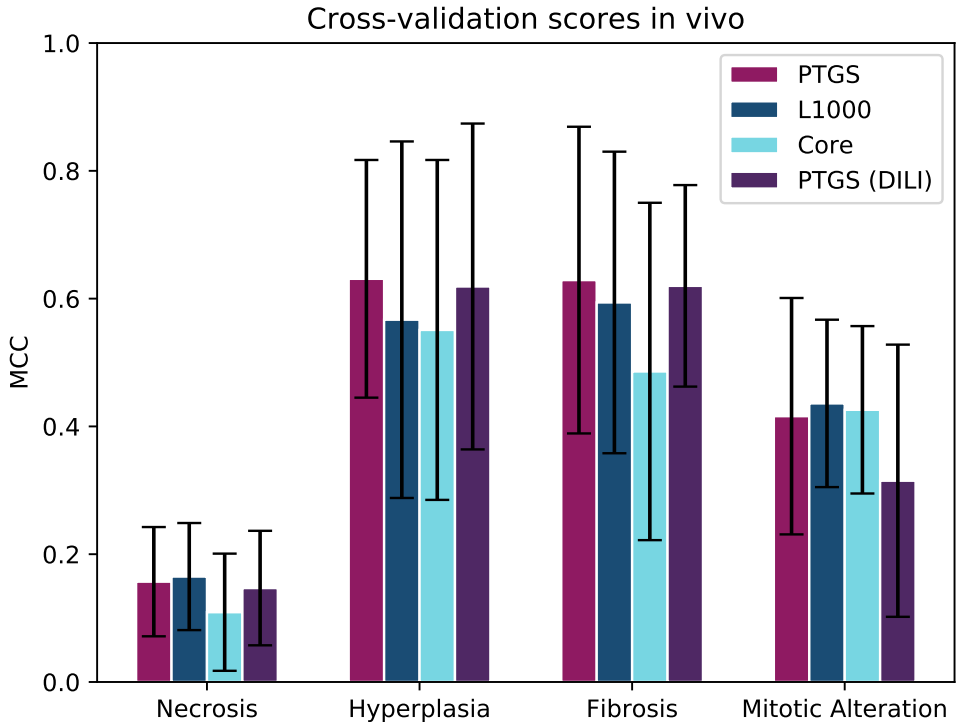


Figure 7.1: Mean cross-validation scores on the four pathologies using different featuresets. The error-bars denotes the standard deviation.

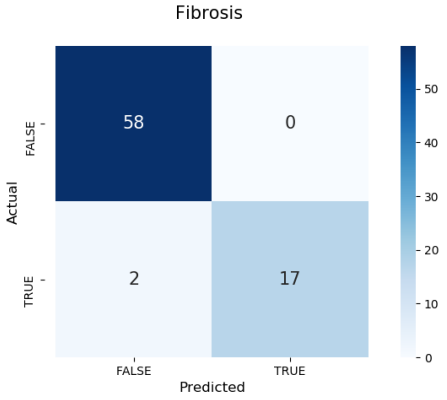
As in results presented for the *in vitro* models, some of the differences between the results show no significant improvement over the others, for example, PTGS and PTGS-DILI in fibrosis and hyperplasia. However, due to having the higher mean, the following models were selected: L1000 for necrosis, L1000 for mitotic alterations, PTGS for fibrosis and PTGS for hyperplasia.

7.2 External validation

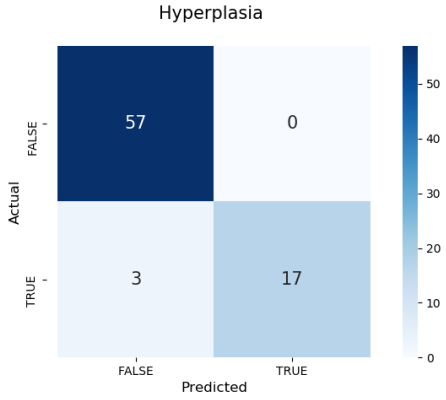
In order to properly validate the developed models, two external test sets were chosen, as explained in Section 5.6.2. This to measure the performance on the type of data the model would be used on. That is, non-related data from a different experimental setting.

7.2.1 Rats exposed to hepatoxicants

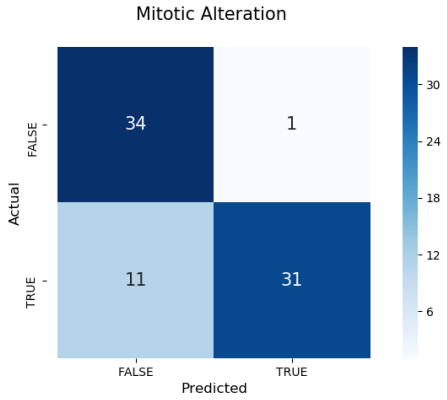
Shown in Figure 7.2 are the confusion-matrices (see Section 2.4.1) for the predictions made on transriptomic data from rats exposed to four different hepatotoxic compounds, collected from the study by Ippolito et al. [36]. *Note that the compounds present in this study were removed from the training data beforehand.*



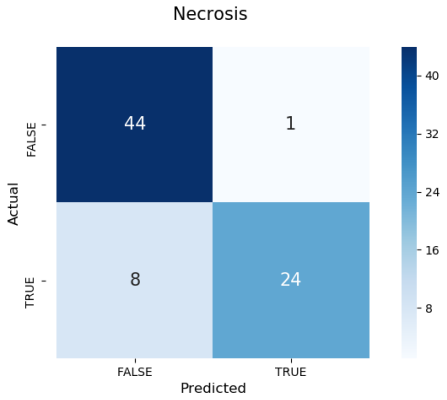
(a) CM for fibrosis.



(b) CM for hyperplasia.



(c) CM for mitotic alterations.



(d) CM for necrosis.

Figure 7.2: Confusion matrices for four different injury predictions on data set by Ippolito et al.

The Matthew correlation coefficients were calculated from the aforementioned confusion matrices using Equation 2.14. The scores for the different histopathology predicting models can be seen in Table 7.3.

Table 7.3: MCC-scores on predictions made on rats exposed to four hepatotoxic compounds.

Featureset	Algorithm	Histopathology	MCC
L1000	Random Forest	Necrosis	0.77
PTGS	DNN (Transfer)	Hyperplasia	0.90
PTGS	DNN (Transfer)	Fibrosis	0.93
L1000	Random Forest	Mitotic Alteration	0.75

The models seemingly perform very well, predicting the different histopathologies with very high correlation. Although these results may seem unreasonably high, similar results have been shown in related studies, such as in the study by Wang et al. [35]. A comparison between the achieved performances can be seen in Figure 7.3.

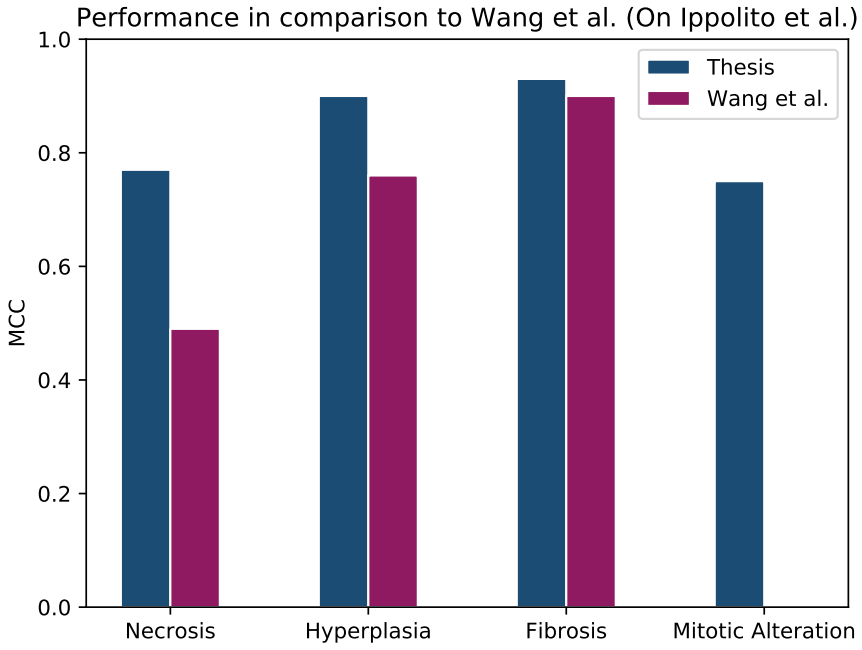


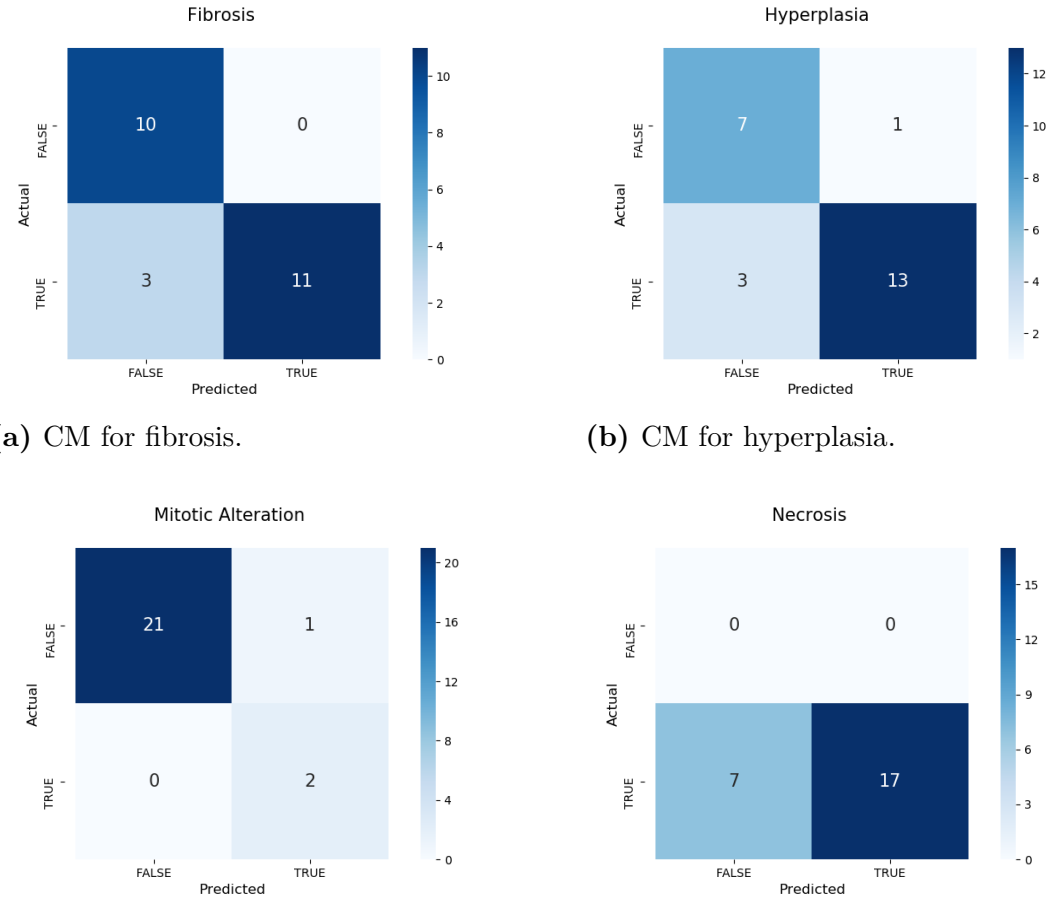
Figure 7.3: Comparison with results from similar study by Wang et al. [35].

As seen in the figure, the models generated in this thesis outperform results previously achieved in literature in all comparable categories (necrosis, hyperplasia and fibrosis).

Do note that this project uses the same input-data, however, with different methodology and featuresets.

7.2.2 Rats exposed to bile duct ligation

The next test was done on rats exposed to bile duct ligation, as seen in Section 5.6.2. Shown in Figure 7.4 are the confusion matrices for the predictions made on transriptomic data collected from the study by Sutherland et al. [37].



(a) CM for fibrosis.

(b) CM for hyperplasia.

(c) CM for mitotic alterations.

(d) CM for necrosis.

Figure 7.4: Confusion matrices for four different histopathological predictions on data set by Sutherland et al.

The Matthew correlation coefficients were calculated from the aforementioned confusion matrices using Equation 2.14. The scores for the different histopathology predicting models can be seen in Table 7.4 or visualized in Figure 7.5.

Table 7.4: MCC-scores on Sutherland et al.

Featureset	Algorithm	Histopathology	Sutherland et al.
L1000	Random Forest	Necrosis	71%
PTGS	DNN (Transfer)	Hyperplasia	0.66
PTGS	DNN (Transfer)	Fibrosis	0.84
L1000	Random Forest	Mitotic Alteration	0.80

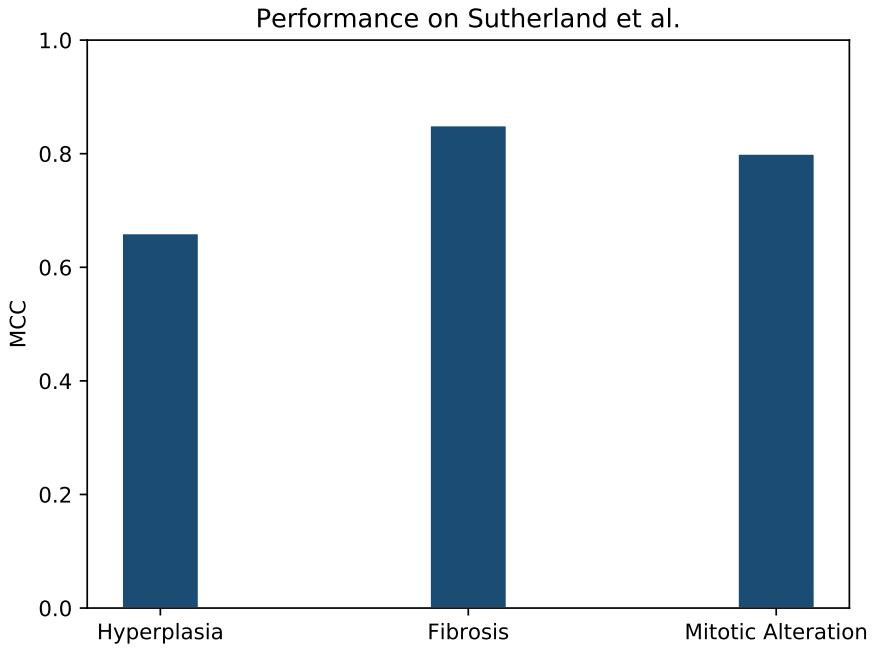


Figure 7.5: Performance on rats exposed bile duct ligation by Sutherland et al.

The models seemingly perform very well in predictions of liver injuries, even when exposed to a non-chemical based treatments, such as bile duct ligation. The Matthews correlation coefficient was omitted when estimating performance on *necrosis*, this due to all of the presented samples having shown that particular injury.

7.2.3 Pathology prediction profiles

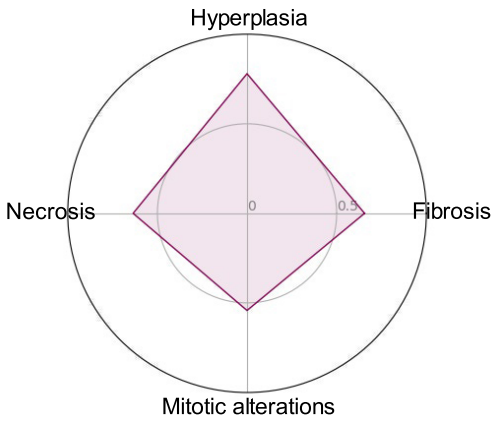
In this subsection, a potential application of the models will be presented.

Table 7.5 shows actual pathological endpoints from three animals from the species *Rattus Norvegicus*, while Figure 7.6 shows the developed models prediction probabilities using the transcriptomic responses from those same individuals.

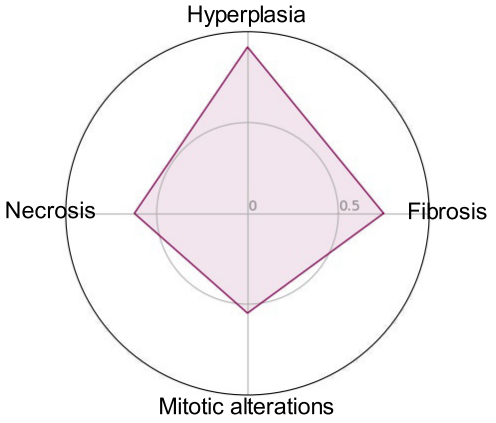
Table 7.5: Measured pathological outcome and severity for individuals treated with bile duct ligation, 3 days after treatment.

Sample	Necrosis	Fibrosis	Hyperplasia	Mitotic Alterations
1	Minimal	Minimal	Moderate	None
2	Moderate	Minimal	Moderate	Minimal
3	Minimal	Minimal	Moderate	Minimal

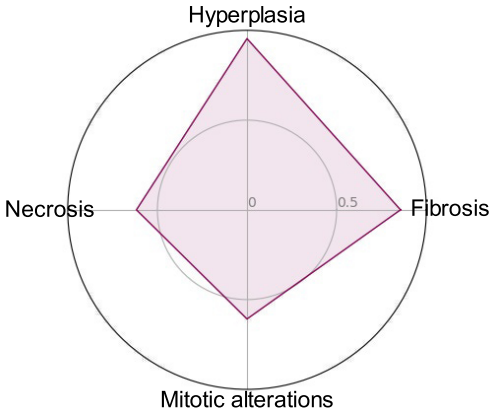
Note that a prediction probability of over 0.5 returns a positive prediction.



(a) Treatment group 1, sample 1.



(b) Treatment group 1, sample 2.



(c) Treatment group 1, sample 3.

Figure 7.6: Prediction probabilities for four different pathological endpoints.

As seen in Figure 7.4c, these are the only samples having been predicted with this specific histopathology. Looking at Figure 7.6a, one can clearly see that the model returns a false positive for mitotic alterations. This could however indicate that the histopathology has not yet been developed, due to all the other samples present in that treatment group having been diagnosed with it.

7.3 Secondary Validation

In order to assess the selection of feature sets and avoidance of overfitted models, models trained using randomized feature sets and scrambled labels were evaluated.

7.3.1 Randomized feature sets

Randomized feature sets were used as control, to estimate the performance of a well-curated feature set. The results from cross-validation and two external tests are shown in Table 7.6. *Note that necrosis was not measured as all cases had reports of said pathology.*

Table 7.6: Cross-validation scores and performance on external test-sets using using randomized featuresets.

Featureset	Histopathology	CV-Score	Ippolito et al.	Sutherland et al.
R1000	Necrosis	0.25 ± 0.10	0.63	N/A
	Hyperplasia	0.65 ± 0.28	0.70	0.45
	Fibrosis	0.59 ± 0.22	0.76	0.53
	Mitotic Alterations	0.38 ± 0.17	0.33	0.0
R500	Necrosis	0.14 ± 0.093	0.13	N/A
	Hyperplasia	0.62 ± 0.27	0.49	0.21
	Fibrosis	0.61 ± 0.17	0.71	0.0
	Mitotic Alterations	0.37 ± 0.14	0.47	0.69

Although comparable in cross-validation scores, the randomized feature sets suffers some losses in generalization when applied to foreign test sets.

7.3.2 Y-scrambling

Y-scrambled models showed no predictive power in cross-validation and on external data as seen in Table 7.7.

Table 7.7: Results of Y-scrambled injury models, presented in MCC. Note that they are an average of 10 different models.

Histopathology	CV (μ)	CV (σ)	Ippolito et al. (μ)	Ippolito et al. (σ)
Necrosis	0.0056	0.013	0.047	0.11
Mitotic Alterations	0.010	0.070	-0.078	0.088
Hyperplasia	0.041	0.030	0.011	0.027
Fibrosis	0.033	0.0013	0.042	0.0

8

Discussion & Conclusion

Predictive models can only perform as well as the quality of used data allows for. Depending on the variance inherent to the platform, development of these models can become problematic. This statement holds especially true for microarray-based data as mentioned in Section 2.2. Another complication when using biological data is the somewhat inexact type of annotations used in this study; the *in vivo* models make use of sample-based histopathology findings, which are subjectively located and ranked by trained professionals. This introduces a human factor, potentially adding more noise to the data. This type of problem is also occuring for the cytotoxicity model, in which the label of choice is dose dependent growth inhibition. Experimental validation by assay is not an end-all be-all solution, with a track-record being far from perfect. However, despite this, the *in silico* models developed during this project indicate that prediction of *in vitro* cytotoxicity and *in vivo* liver injury is indeed possible.

On the case of performance for the different injury-categories, they all seem to produce extremely high prediction metrics, even in regards to the conditions previously mentioned. Fibrosis and hyperplasia are the top performers in the test set created in the study by Ippolito et al., wherein rats are exposed to four hepatotoxicants [36]. Fibrosis goes on to produce very impressive predictions on non-related test-set in which rats are exposed to bile duct ligation [37]; A very surprising discovery due to the lack of positive samples in these categories, and also due to the non-chemical treatment present in that set. The other two categories, mitotic alterations and necrosis, reach respectable performance, especially the former one. However, regarding necrosis, the model seemingly was not able to capture all of the essential features, suffering from a large loss in performance in some settings, as seen in Figure 7.1.

Keeping the previous paragraph in mind, these results could potentially indicate that transcriptomic signatures are more well defined in some categories of injury. Necrosis is a relatively well-studied phenomena, and involves the use of a plethora of different pathways, while also interfering with the functionality of nearby tissue [41]. The features in use do not seem able to capture this process in its entirety, causing loss of performance. The other three categories show differing results, by having a profile that is seemingly well captured in the different feature sets, whilst still having a smaller amount of samples to train on. For example, mitotic alterations showed higher performance using the L1000 feature set, a set which is designed to infer the complete genome. This also correlates to an interesting result regarding the appli-

cability of transfer learning between the different pathologies. Some injuries, such as fibrosis, are often preceded by necrosis. By looking at Table 7.1, one sees that using necrosis as a source domain results in improved classifier performance for all relevant categories, while not being, in itself, an indicator of strong performance.

In the study by Wang et al. they use more classical approaches when predicting the different types of injury, more specifically using a deep neural network to predict the endpoints [35]. These results are evidently outperformed by the changes made for this thesis, namely, applying transfer learning between the different injuries, and using a simpler random forest model for necrosis. However, a large caveat is the size of the studies used for external validation. To further ensure actual performance, a larger test would be needed, minimizing variation.

A transcriptomic change usually occurs before a physiological one, denoting the potential of this model to produce predictions regarding future injuries as shown in Section 7.2.3. A property that, in theory, could be used to great effect in short-term studies, where the allotted times are too short to induce physiological injuries or responses.

The second set of models produced in this study are on the topic of predicting cytoxicity using transcriptomic responses from immortalized cancer cell lines, with data collected from CMap and annotation given by growth inhibition data from the NCI60 cell line panel. The model was tested on a foreign cell line, A-549, indicating good performance as seen in Figure 6.2. However, as previously mentioned, a constantly occurring hindrance during this thesis is the problem of validation. In order to find liver-specific cytotoxicity markers, there needs to be studies made on that specific type of endpoint, which is hard to experimentally validate. Another important observation is the non-specificity of not just the different tissues available, but the fact that they are immortalized cancer cells. This might not in reality be representative of normal primary human cells, as they are not identical in behaviour.

Under the assumption that cytotoxicity has a strong signature shared by the different kinds of cell lines and tissues, predictions were made on primary human hepatocytes from the TG-GATEs database. Although the model predicts cytotoxicity for higher doses, one could argue that it is not liver specific, or even accurate, due to lack of validation. But it is a good indicator that the model at least captures relevant features, as compounds which are not toxic at all can still trigger cell-death in high enough doses, an observation than can be made from analyzing Figure 6.3.

When investigating the predictions on the set of randomized compounds this seems especially clear. The two prominent examples being caffeine and vitamin A. Caffeine is not a typical hepatotoxicant, but the doses are high, causing a strong response, which the models classifies as a toxic signature. The opposite can be said about vitamin A, which is a an example of a typical hepatotoxicant. However, in this case, the doses administrated are relatively low, not causing a strong response, subsequently causing the model to classify the sample as non-toxic. These two observations are

examples of the limitations of the model, as it is unable to handle more specific intricacies regarding liver toxicity.

When measuring for cross-validation performance, a randomized feature set exhibited predictive power similar to that of a curated one, such as the L1000-genes or the PTGS-genes, an observation which can also be seen in the work by Wang et al. [35]. However, the performance on completely external testing seems lacking, although it still capable of producing non-random predictions which can be seen in Figure 8.1. Correlation instead of causation is a typical problem in machine learning, that is, when a model deems non-causative features important due to correlation. When using transcriptomic responses, one needs to consider the possibility of interaction between the different genes. The results shown in Sections 6.4 and 7.3 indicate that correlated genes can produce strong performance. However, due to the selection process being completely random, causative genes might have been captured. Further study needs to be done before a more definitive conclusion can be made.

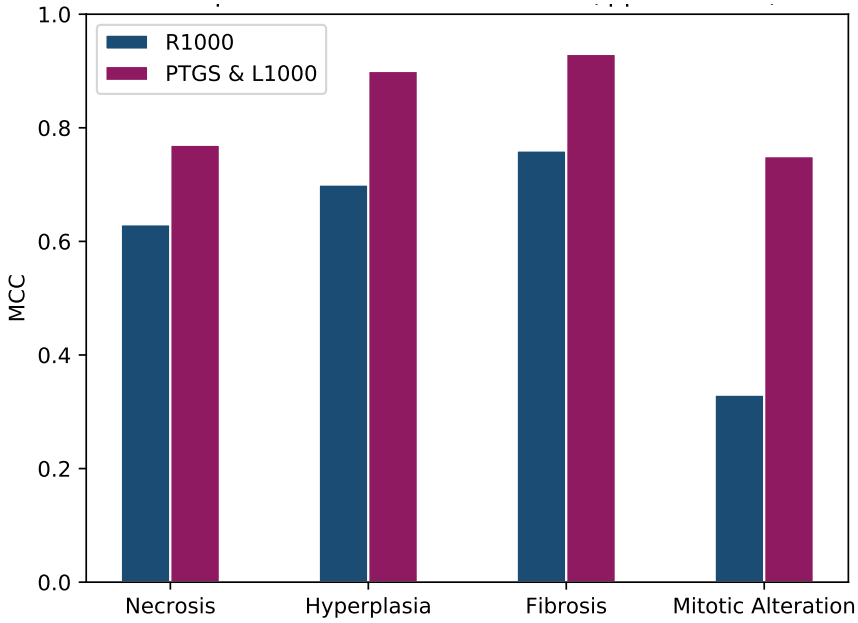


Figure 8.1: Bar plot showing the discrepancy in predictor performance when using a curated feature set versus a randomized one (data from Section 7.3).

Another significant finding is the applicability of the synthetic minority oversampling technique for use in transcriptomic data, a statement mirrored in the study by Wang et al. Even though it is traditionally seen as a subpar method for high-dimensional data, due to the dimensionality reduction caused by using the curated feature sets mentioned in Section 4.1 it seemingly provides a reasonable way to improve balance without losing performance.

8.1 Future work

A cell line which was not included in this study but does have data available in CMap, namely HL-60 (Human Leukemia), shows promise as a cell line which can be highly representative of human hepatocytes, shown in the work by by Liu et al. [42]. However, one needs to keep in mind that liver-specific pathways might not be captured fully, denoting the importance of a well defined prediction.

Due to there being a higher prevalence of data available for these less specific types of cell lines, in part due to the difficulties inherent to working with primary cells, the previous paragraph could be used as an indication for future work regarding the subject of drug induced liver injury [43]. As more data becomes available, the efficacy of these types of studies will increase. The same can be said about high-throughput sequencing technologies; microarrays, which were used in this study is a relatively outdated technology, having been replaced with the likes of the more qualitative RNASeq. The availability of more data from these different techniques should result in better predictive performance.

The data produced in this thesis could also be used for more types of predictive models. There exists a plethora of biochemical and hematological measurements in both the DrugMatrix and TG-GATEs databases, potentially allowing for regression models of important liver health metrics such as alanine-aminotransferase.

Bibliography

[1] D. Cook, D. Brown, R. Alexander et al. Lessons learned from the fate of AstraZeneca's drug pipeline: a five-dimensional framework. *Nature Reviews Drug Discovery 13*, 419–431, 2014.

[2] Y. Igarashi, N. Nakatsu, T. Yamashita, A. Ono, Y. Ohno, T. Urushidani & H. Yamada. Open TG-GATEs: a large-scale toxicogenomics database. *Nucleic acids research*, 43 (Database issue), 921–927, 2015.

[3] B. Ganter et al. Development of a large-scale chemogenomics database to improve drug candidate selection and to understand mechanisms of chemical toxicity and action, *Journal of Biotechnology*, Volume 119, Issue 3, 219-244, 2005.

[4] D.L. Svoboda, T. Saddler, S.S. Auerbach. An Overview of National Toxicology Program's Toxicogenomic Applications: DrugMatrix and ToxFX. *Advances in Computational Toxicology. Challenges and Advances in Computational Chemistry and Physics*, vol 30. Springer, Cham, 2019.

[5] J. Lamb, E.D. Crawford, D. Peck, J.W. Modell, I.C. Blat, M.J. Wrobel, J. Lerner, JP. Brunet, A. Subramanian, K.N. Ross, M. Reich, H. Hieronymus, G. Wei, S.A. Armstrong, S.J. Haggarty, P.A. Clemons, R. Wei, S.A. Carr, E.S. Lander, T.R. Golub. The Connectivity Map: Using Gene-Expression Signatures to Connect Small Molecules, Genes, and Disease. *Science* 29, Vol. 313, Issue 5795, 1929-1935, Sep 2006.

[6] A. Subramanian, R. Narayan, S.M. Corsello, D. Peck, T.E. Natoli, X. Lu, T.R. Golub. A Next Generation Connectivity Map: L1000 Platform and the First 1,000,000 Profiles. *Cell*, 171(6), 1437–1452, 2017.

[7] R. Shoemaker.The NCI60 human tumour cell line anticancer drug screen. *Nature Reviews Cancer* 6, 813–823, 2006.

[8] R.J. Weaver, E.A. Blomme, A.E. Chadwick et al. Managing the challenge of drug-induced liver injury: a roadmap for the development and deployment of preclinical predictive models. *Nature Reviews Drug Discovery* 19, 131–148, 2019

[9] M. Chen, A. Suzuku, J. Borlak, R.J. Andrade, M.I. Lucena. Drug-induced liver injury: Interactions between drug properties and host factors. *Journal of Hepatology* 63, 503.514, 2015

[10] C.E. Eapen. The liver: Oxidative stress and dietary antioxidants. *The Indian Journal of Medical Research* vol. 149, 2019.

[11] M. Mosedale & P.B. Watkins. Drug-induced liver injury: Advances in mechanistic understanding that will inform risk management. *Clinical pharmacology and therapeutics*, 101(4), 469–480, 2017.

[12] K. Köck & K.L. Brouwer. A perspective on efflux transport proteins in the liver. *Clinical pharmacology and therapeutics*, 92(5), 599–612, 2012.

[13] D.E. Kleiner, N.P. Chalasani, W.M. Lee, R.J. Fontana, H.L. Bonkovsky, P.B. Watkins, P.H. Hayashi, T.J. Davern, V. Navarro, R. Reddy, J.A. Talwalkar, A. Stolz, J. Gu, H. Barnhart, J.H. Hoofnagle & Drug-Induced Liver Injury Network. Hepatic histological findings in suspected drug-induced liver injury: systematic evaluation and clinical associations. *Hepatology (Baltimore, Md.)* 59(2), 661–670, 2014.

[14] D.E. Kleiner. Drug-induced Liver Injury: The Hepatic Pathologist's Approach. *Gastroenterology clinics of North America*, 46(2), 273–296, 2017.

[15] X. Gao, E. Gulari & X. Zhou. In situ synthesis of oligonucleotide microarrays. *Biopolymers*, 73: 579-596, 2004.

[16] B.M. Bolstad, R.A. Irizarry, M. Astrand & T.P. Speed., A Comparison of Normalization Methods for High Density Oligonucleotide Array Data Based on Bias and Variance. *Bioinformatics* 19(2):185-193, 2003.

[17] R.A. Irizarry, B.M. Bolstad, F. Collin, L.M. Cope, B. Hobbs & T.P. Speed. Summaries of Affymetrix GeneChip probe level data. *Nucleic Acids Research*, 31(4):e15, 2003.

[18] R.A. Irizarry, B. Hobbs, F. Collin, Y.D. Beazer-Barclay, K.J. Antonellis, U. Scherf, T.P. Speed. Exploration, Normalization and Summaries of High Density Oligonucleotide Array Probe Level Data. *Biostatistics*, vol. 4, Number 2: 249-264, 2003.

[19] F. Mosteller, *J.W. Tukey. Data Analysis and Regression: A Second Course in Statistics*. Addison-Wesley Pub. Co., 1977.

[20] C.C. Chang & C.J Lin. LIBSVM: A Library for Support Vector Machines. *Department of Computer Science, National Taiwan University*, Taipei, Taiwan, 2001.

[21] L. Breiman. Random Forests. *Machine Learning* 45, 5–32, 2001.

[22] S. Haykin. *Neural Networks and Learning Machines*. Pearson, London, third edition, 2008.

[23] I. Goodfellow, Y. Bengio & A. Courville. *Deep Learning*. MIT press, 2016.

[24] D. Kingma & J. Ba. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*, 2014.

[25] R. Blagus & L. Lusa. SMOTE for high-dimensional class-imbalanced data. *BMC bioinformatics*, 14, 106, 2013.

[26] B.W. Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme, *Biochimica et Biophysica Acta (BBA) - Protein Structure*, Vol. 405, Issue 2, 442-451, 1975.

[27] D. Chicco. Ten quick tips for machine learning in computational biology. *BioData mining*, Vol. 10, 35, 2017.

[28] J.L. Devore. *Probability & Statistics for Engineering and the Sciences*. California Polytechnic State University, San Luis Obispo, 8th edition, 2012.

[29] W.C. Reinhold, M. Sunshine, H. Liu, S. Varma, K.W. Kohn, J. Morris, J. Doroshow & Y. Pommier. CellMiner: A Web-Based Suite of Genomic and Pharmacologic Tools to Explore Transcript and Drug Patterns in the NCI-60 Cell Line Set. *Cancer Research*, Vol. 72(14), 3499-3511, 2012.

[30] B.S. Carvalho & R.A. Irizarry. A Framework for Oligonucleotide Microarray Preprocessing. *Bioinformatics*, Vol. 16(19), 2363-2367, 2010.

[31] J.M. Bland & D.G. Altman. Statistical methods for assessing agreement between two methods of clinical measurement. *Measurement*, Vol. 327, Issue 8476, 307-310, 1986.

[32] P. Kohonen, R.C. Grafström, Willighagen, E. et al. A transcriptomics data-driven gene space accurately predicts liver cytopathology and drug-induced liver injury. *Nature Communications* 8, 15932, 2017.

[33] T. Hart, M. Chandrashekhar, M. Aregger, Z. Steinhart, K.R. Brown, G. MacLeod et al. High-Resolution CRISPR Screens Reveal Fitness Genes and Genotype-Specific Cancer Liabilities. *Cell*, Vol. 163(6), 1515–1526, 2015.

[34] C. Kampf, A. Mardinoglu, L. Fagerberg, B.M. Hallström, K. Edlund, E. Lundberg, F. Pontén, J. Nielsen, & M. Uhlen. The human liver-specific proteome defined by transcriptomics and antibody-based profiling. *The FASEB Journal* 28:7, 2901-2914. 2014.

[35] H. Wang, R. Liu, P. Schyman & A. Wallqvist. Deep Neural Network Models for Predicting Chemically Induced Liver Toxicity Endpoints From Transcriptomic Responses. *Front. Pharmacology*, 10-42, 2019.

[36] D.L. Ippolito, M.D. AbdulHameed, G.J. Tawa, C.E. Baer, M.G. Permenter, B.C. McDyre, William E. Dennis, M.H. Boyle, C.A. Hobbs, M.A. Streicker, B.S. Snowden, J.A. Lewis, A. Wallqvist, J.D. Stallings. Gene Expression Patterns Associated With Histopathology in Toxic Liver Fibrosis. *Toxicological Sciences*, Vol. 149, Issue 1, 67-88, 2016.

[37] J. Sutherland, Y. Webster, J. Willy et al. Toxicogenomic module associations with pathogenesis: a network-based approach to understanding drug toxicity. *Pharmacogenomics Journal* 18, 377–390, 2018.

[38] C. Rücker, G. Rücker & M. Meringer. y-Randomization and Its Variants in QSPR/QSAR. *Journal of Chemical Information and Modeling*, Vol. 47(6), 2345-2357, 2007.

[39] M. Chen, A. Suzuki, S. Thakkar, K. Yu, C. Hu, W. Tong. DILIrank: the largest reference drug list ranked by the risk for developing drug-induced liver injury in humans. *Drug Discovery Today*, Vol. 21, Issue 4, 648-653, 2016.

[40] H.K. Hamadeh, B.L. Knight, A.C. Haugen, S. Sieber, R.P. Amin, P.R. Bushel, R.S. Paules. Methapyrilene Toxicity: Anchorage of Pathologic Observations to Gene Expression Alterations. *Toxicologic Pathology*, Vol. 30(4), 470–482, 2002.

[41] N. Festjens, T.V. Berghe, P. Vandenabeele. Necrosis, a well-orchestrated form of cell demise: Signalling cascades, important mediators and concomitant immune response. *Biochimica et Biophysica Acta (BBA) - Bioenergetics*, Vol. 1757, Issues 9–10, 1371-1387, 2006.

[42] Z. Liu, L. Zhu, S. Thakkar, R. Roberts & W. Tong. Can Transcriptomic Profiles from Cancer Cell Lines Be Used for Toxicity Assessment? *Chemical Research in Toxicology*, Vol. 33(1), 271-280, 2020.

[43] Sigma-Aldrich, Primary Cell Culture Basics. Retrieved from: `https://www.sigmaaldrich.com/technical-documents/articles/biology/primary-cell-culture.html`, 2020.