MACHINE LEARNING IN ONCOLOGY FOR PREDICTING DRUG APPROVAL

A PROJECT REPORT

Submitted By

PRANAVI SHEKHAR 312217104109

PRANAV RAVEENDRAN 312217104110

PRANAV VIJAY 312217104111

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

SSN COLLEGE OF ENGINEERING

KALAVAKKAM 603110

ANNA UNIVERSITY :: CHENNAI - 600025

April 2021

ANNA UNIVERSITY : CHENNAI 600025

BONAFIDE CERTIFICATE

Certified that this project report titled **"MACHINE LEARNING IN ONCOLOGY FOR PREDICTING DRUG APPROVAL"** is the *bonafide* work of "**PRANAVI SHEKHAR (312217104109)**, **PRANAV RAVEENDRAN (312217104110)**, and **PRANAV VIJAY (312217104111)**" who carried out the project work under my supervision.

Dr. CHITRA BABU
HEAD OF THE DEPARTMENT
Professor,
Department of CSE,
SSN College of Engineering,
Kalavakkam - 603 110

Dr. T.T. MIRNALINEE
SUPERVISOR
Professor,
Department of CSE,
SSN College of Engineering,
Kalavakkam - 603 110

Place:
Date:

Submitted for the examination held on. . . . . .

Internal Examiner **External Examiner**

ACKNOWLEDGEMENTS

I thank GOD, the almighty for giving me strength and knowledge to do this project.

I would like to thank and deep sense of gratitude to my guide **Dr. T.T. MIRNALINEE**, Professor, Department of Computer Science and Engineering, for her valuable advice and suggestions as well as her continued guidance, patience and support that helped me to shape and refine my work.

My sincere thanks to **Dr. CHITRA BABU**, Professor and Head of the Department of Computer Science and Engineering, for her words of advice and encouragement and I would like to thank our project Coordinator **Dr.B. BHARATHI**, Associate Professor, Department of Computer Science and Engineering for her valuable suggestions throughout this project.

I express my deep respect to the founder **Dr. SHIV NADAR**, Chairman, SSN Institutions. I also express my appreciation to our **Dr. V. E. ANNAMALAI**, Principal, for all the help he has rendered during this course of study.

I would like to extend my sincere thanks to all the teaching and non-teaching staffs of our department who have contributed directly and indirectly during the course of my project work. Finally, I would like to thank my parents and friends for their patience, cooperation and moral support throughout my life.

PRANAVI SHEKHAR **PRANAV RAVEENDRAN** **PRANAV VIJAY**

ABSTRACT

Drug development in Oncology is presently confronting a combination of an expanding number of anti-cancer drugs for Phase I clinical trials (P1CTs) and a significant attrition rate for final approval. It was observed from previously reported literature that prediction of final approval rates from Phase 1 clincal trials was not conclusive. Since drug development is an extremely time and cost intensive process, the objective of this project is to build a machine learning model to predict whether or not the drug will be approved for release, based on the results of Phase 1 evaluation of the clinical trial. PubMed abstracts of Phase 1 clinical trials reporting on Anti-Neoplastic Agents were used together with pharmacologic data from the DrugBank5.0 database to model time to US Food and Drug Administration (FDA) approval. The result of this project will help determine the likelihood of FDA approval(the final phase) from Phase 1(the first phase) itself. This will help decide early on if the drug is worth developing enough to enter into subsequent phases, namely Phases 2 and 3, both of which are long-drawn and expensive.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background

Before a drug is deemed suitable for patients, it has to go through rigorous testing and cost-effectiveness analyses. Drug development comprises all such activities involved in transforming a compound from drug candidate (the end-product of the drug discovery phase) to a product approved for marketing by the appropriate regulatory authorities, such as the US Food and Drug Administration(USFDA or FDA). The entire process – from concept through pre-clinical testing in the laboratory to clinical trial development, including Phase 1–3 trials – to approved vaccine or drug typically takes more than a decade.

1.1.1 Clinical trials

Clinical trials are a type of research that studies new tests and treatments and evaluates their effects on human health outcomes. These trials are carried out in 3 specific phases, as outlined below.

Phase 1 : The safety of the drug is evaluated, safe dosage determined and any major side effects observed. Phase 1 evaluations can also give early signals of the efficacy of the compounds[4] which can help decide whether the drug is viable for further development.

Phase 2 : This phase involves testing the effectiveness of the drug on a large number of patients and digging deeper into how safe it is.

Phase 3 : In this phase the effectiveness of a drug is confirmed, its side effects are heavily monitored and the drug is compared to other available treatments. If a drug clears Phase 3, it is deemed fit for usage and approved for release and is granted FDA approval.

1.2 Motivation

Oncology is a branch of medicine that deals with the prevention, diagnosis, and treatment of cancer. Drug development in oncology is a rapidly changing field with various difficulties[1, 2].The drugs used to treat cancerous tumours/neoplasms are called **Anti-neoplastic agents(ANAs)**. As mentioned before, the process of developing such a drug involves a comprehensive clinical trial to assess its safety and efficacy before release.

Drug attrition rates for oncology are much higher than in other therapeutic areas. Only 5% of agents that have anticancer activity in Phase 1 development are licensed after demonstrating sufficient efficacy in Phases 2 and 3, which is much lower than, for example, 20% for cardiovascular disease. This results in significant losses to the overall development process - it hinders enrollment of patients in other investigations, results in considerable financial losses for the pharmaceutical industry and academic institutions and causes a massive wastage of time. effort ad resources.

Phase 1 trials in oncology are dedicated to analysing drug safety and can provide early signs of efficacy of the compounds. All the aforementioned costs can be reduced by making an early decision in Phase 1 regarding how feasible it is to develop the proposed drug. This will prevent drugs unlikely to get final approval from entering the long-drawn and expensive Phases 2 and 3.

1.3 Problem definition

- The objective of this work is to develop a machine learning model to predict the likelihood of FDA approval of a drug from Phase 1 clinical trial data.
- As suggested before, this will prevent drugs which have a very low likelihood of obtaining FDA approval, as determined in Phase 1, from entering into subsequent phases.
- We aim to demonstrate the utility and feasibility of a prediction system that could improve and supplement drug development in oncology by giving an early decision regarding approval as soon as Phase I trials are completed.

1.4 Scope

Likelihood of drug approval can be predicted by building a machine learning model that uses a combination of pharmacologic data obtained from the DrugBank database[5] and the data for Phase 1 trials extracted from PubMed. This data is called **survival data** as it provides information regarding the time taken for the event of interest (FDA approval) to occur.

The proposed approach uses a combination of Lasso Penalized Cox Regression models for training and feature selection. The likelihood of FDA approval is computed using the output of the trained model. The performance of the model was evaluated using weighted and non-weighted concordance indices and an optimum threshold was computed to classify the likelihood of drug approval into likely to be approved/non-approved.

1.4.1 Survival analysis

Survival analysis corresponds to a set of statistical approaches used to investigate the time it takes for an event of interest to occur. Survival data is essentially time-to-event data where information about a subject is collected over a length of time, from the time of origin to an endpoint of interest. Survival analysis methods are usually used to analyse data collected prospectively in time, such as data from a prospective cohort study or data collected for a clinical trial.

In the context of this project, the event of interest is FDA approval and the subjects of the clinical trial are the drugs for which survival data is available. The data we use describes the time taken for FDA approval of various drugs whose composition during Phase 1 clinical trials is considered.

One of the reasons why survival analysis requires 'special' techniques is the possibility of not observing the event of interest for some individuals. Another possibility is that there might be a time point at which the study finishes and thus if any subjects have not had their event yet, their event time will not have been observed. These incomplete observations cannot be ignored, but need to be handled differently. This is called **censoring**. Another feature of survival data is

that distributions are often skewed (asymmetric) and thus simple techniques based on the normal distribution cannot be directly used.

1.4.1.1 Censoring

Censoring in a study is when there is incomplete information about a study participant, observation or value of a measurement or in our case, the drug of interest. Essentially, it's when the event (FDA approval) does not happen while the subject (drug) is being monitored.

The most commonly encountered type of censoring and easiest to handle in the analysis is *right censoring*. Right censoring occurs when a drug is followed up from a time origin t_0 up to some later time point t_c and it has not had the event of interest i.e. not been approved, such that all we know about the drug is that it has not been approved up to the censoring time t_c.

Right censoring can be *informative* or *non-informative*. *Non-informative* censoring occurs if the distribution of survival times provides no information about the distribution of censorship times and vice versa i.e. the reason for censorship should be unrelated to the study. *Informative* censoring occurs when subjects are lost to follow-up due to reasons related to the study.

1.4.1.2 Logrank Test

The logrank test[23] is used to test the null hypothesis that there is no difference between the populations in the probability of an event(here FDA Approval) at any time point. The analysis is based on the times taken for the event of interest to

occur. It is most likely to detect a difference between groups when the risk of an event is consistently greater for one group than another.

The log rank test is a non-parametric test, which makes no assumptions about the survival distributions. Essentially, the log rank test compares the observed number of events in each group to what would be expected if the null hypothesis were true. The log rank statistic is approximately distributed as a chi-square test statistic.

The logrank statistic L is given by equation 1.1:

$$L = \frac{\sum_{i=1}^{d} \left(X_i - \frac{n_{2i}}{n_{1i}+n_{2i}} \right)}{\left[\sum_{i=1}^{d} \frac{n_{1i} n_{2i}}{(n_{1i}+n_{2i})^2} \right]^{1/2}} \tag{1.1}$$

where X_i is an indicator for the control group, n_{2i} is the number at risk in the experimental group just before the ith event (in our case, FDA approval) and n_{1i} is the number at risk in the control group(non-approved ANAs) just before the ith event.

1.5 Organization

The document is organized into 5 chapters - Introduction, Literature Survey, Proposed system, Implementation and Results and Discussion.

Literature survey focuses on references to past research and a brief description of each work that has been explored. Following this, proposed system outlines the modular architecture of our solution and provides an in-depth view of all the methodologies and algorithms used.

Implementation talks about the actual computational steps involved in realising the architecture using R programming. The thesis concludes by looking at the performance of the model constructed, in addition to comparison with standard approaches, in the Results and Discussion chapter.

CHAPTER 2

LITERATURE SURVEY

AndrewW. Lo et.al.[6], Applied machine-learning techniques to predict drug approvals and phase transitions using drug-development and clinical-trial data from 2003 to 2015 involving several thousand drug-indication pairs with over 140 features across 15 disease groups. Imputation methods were used to deal with missing data, in order to fully exploit the entire data set.

Youran Qi et.al.[7], Proposed a framework to predict the overall treatment effect for patients enrolled into a Phase 3 trial, consisting of two models. First, an individual trough pharmacokinetic concentration (Ctrough) model was developed to predict the trough pharmacokinetic concentration for a potentially new treatment regime planned for Phase 3. Second, an individual treatment effect model was built to model the relationship between patient baseline characteristics, Ctrough and clinical outcomes.

Jelena Gligorijevic et.al.[8], Facilitated clinical trials optimization via a novel approach that builds on top of advances in deep learning which is designed to learn from both investigator and trial-related heterogeneous data sources and rank investigators based on their expected enrollment performance on new clinical trials.

Konstantina Kourou et.al.[9], This work presents a review of recent ML approaches employed in the modeling of cancer progression. The predictive

models discussed are based on various supervised ML techniques as well as on different input features and data samples. Given the growing trend on the application of ML methods in cancer research this paper presents the most recent publications that employ these techniques as an aim to model cancer risk or patient outcomes.

Robert Tibshirani et.al.[10], Proposed a new method for variable selection and shrinkage in Cox's proportional hazards model. This proposal minimizes the log partial likelihood subject to the sum of the absolute values of the parameters being bounded by a constant. Because of the nature of this constraint, it shrinks coefficients and produces some coefficients that are exactly zero. As a result it reduces the estimation variance while providing an interpretable final model. The method is a variation of the 'lasso' proposal of Tibshirani, designed for the linear regression context. Simulations indicate that the lasso can be more accurate than stepwise selection in this setting.

CHAPTER 3

PROPOSED SYSTEM

3.1 Overview of System architecture

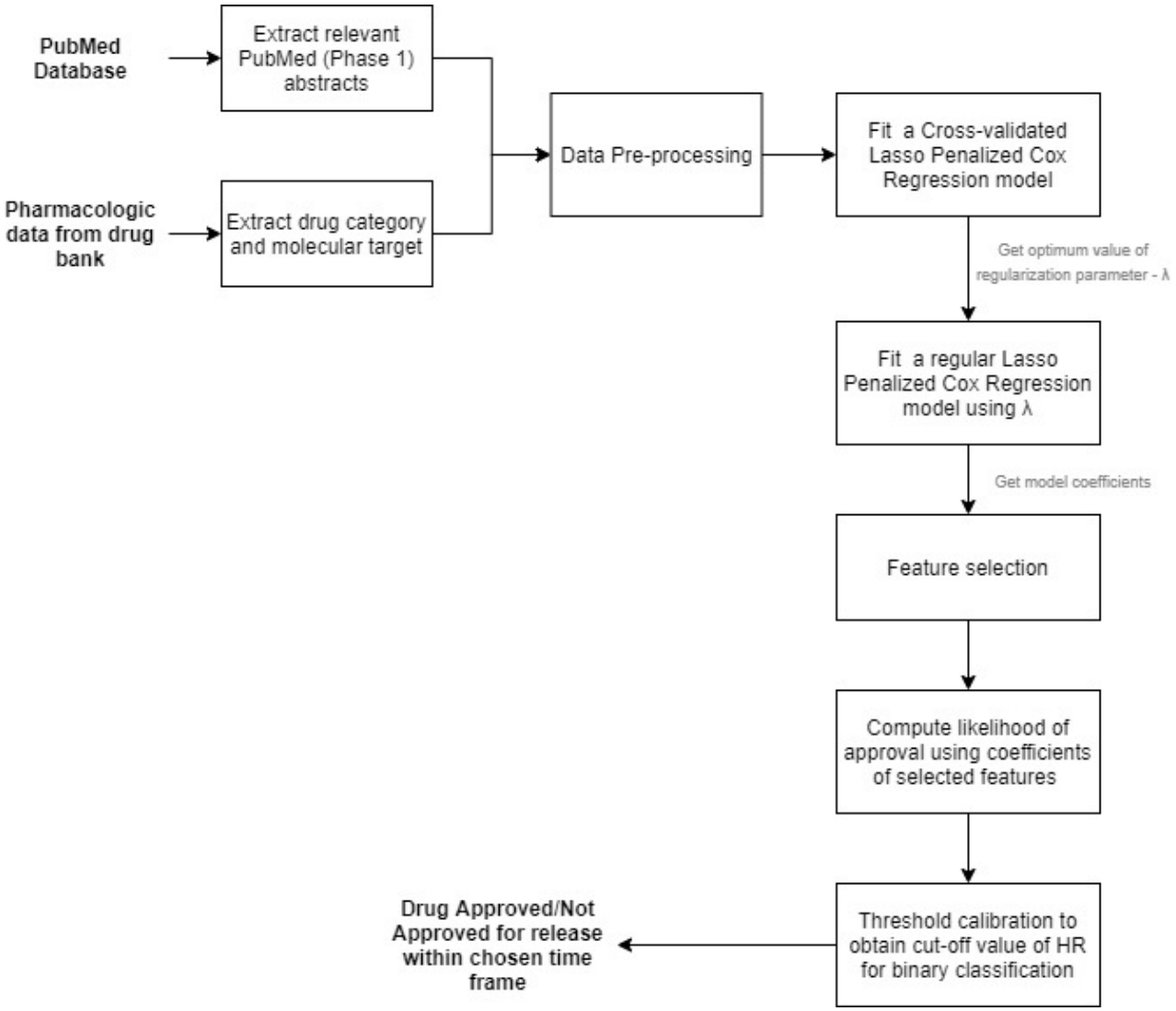

FIGURE 3.1: System architecture

3.1.1 Problem definition

The problem at hand is a binary classification problem which involves predicting whether a drug will be approved by the FDA or not using data pertaining to Phase-1 clinical trials of anti-neoplastic/anti-cancer drugs.

3.1.2 Description of architecture

The system architecture is described in figure 3.1. As discussed previously, the data used for this work comes from the **DrugBank** and **PubMed** databases. These are the most comprehensive and exhaustive sources for all drug related data and also have dedicated resources to facilitate easy data extraction .

The first step therefore involves identifying and merging the common data from PubMed abstracts related to Phase 1 clinical trials and DrugBank5.0. This data will contain drug approval information for ANAs that underwent clinical trials between 1970-2017. This type of data is called **survival data** since it contains time-to-event information.

Once we have a usable data-set the next stage is model fitting. The model chosen for this work uses Lasso Penalized Cox Regression. Cox Regression has been chosen because survival data is used with the objective of understanding the effect of different predictors on drug approval. A lasso penalty is applied to Cox Regression to account for the bias-variance trade-off and enable feature selection.

Two types of Lasso Penalized Cox Regression models are fit to the data to obtain model coefficients. Subsequently, feature selection is performed using the

coefficients obtained from the model. Following this, hazard ratios are computed using the coefficients of the selected features - this tells us the likelihood of FDA approval.

Finally, threshold calibration is carried out to determine the optimum value of hazard ratio which can be used to separate the data into 2 classes - likely to get FDA approval or not likely to get FDA approval.

3.2 Data extraction and pre-processing

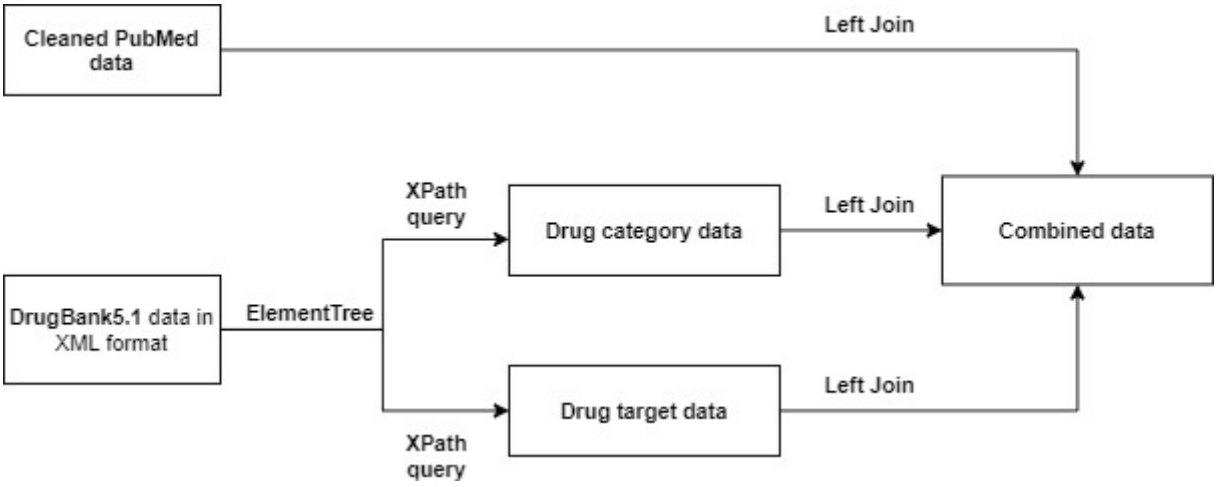

FIGURE 3.2: Data pre-processing

This section outlines the steps involved in accessing the **DrugBank** and **PubMed** databases, extracting the necessary data from them and pre-processing the data by combining it into a single data-set which will serve as the input for subsequent actions, as shown in figure 3.2.

Phase 1 clinical trial data is obtained from **PubMed** and pharmacologic information regarding the individual ANAs present in the PubMed data is obtained from **DrugBank**.

3.2.1 Sources

3.2.1.1 DrugBank

The **DrugBank** database is a comprehensive, freely accessible, online database containing information on drugs and drug targets. The knowledge base consists of proprietary authored content describing clinical level information about drugs such as side effects and drug interactions, as well as molecular level data such as chemical structures and what proteins a drug interacts with. DrugBank is available online and is widely used by the drug industry, medicinal chemists, pharmacists, physicians, students and the general public.

DrugBank enables us to extract vital information regarding the Anti-neoplastic agents of choice. Each drug is associated with a **category** and a **target** in the DrugBank database. Categories are assigned based on the kind of action the drug has in the body while targets are molecules in the body that are associated with a particular disease process that can be addressed by a drug to produce a desired therapeutic effect.

3.2.1.2 PubMed

PubMed is a free search engine accessing primarily the MEDLINE database of references and abstracts on life sciences and biomedical topics. The United States National Library of Medicine (NLM) at the National Institutes of Health maintain the database as part of the Entrez system of information retrieval. From 1971 to 1997, online access to the MEDLINE database had been primarily through

institutional facilities, such as university libraries. PubMed, first released in January 1996, ushered in the era of private, free, home- and office-based MEDLINE searching. The PubMed system was offered free to the public starting in June 1997.

As of 27 January 2020, PubMed has more than 30 million citations and abstracts dating back to 1966, selectively to the year 1865, and very selectively to 1809. As of the same date, 20 million of PubMed's records are listed with their abstracts, and 21.5 million records have links to full-text versions (of which 7.5 million articles are available, full-text for free). Over the last 10 years (ending 31 December 2019), an average of nearly 1 million new records were added each year. Approximately 12% of the records in PubMed correspond to cancer-related entries, which have grown from 6% in the 1950s to 16% in 2016.

PubMed can be used to obtain details about the drugs Phase 1 clinical trials of ANAs developed between 1972-2017.

3.2.2 Getting PubMed data

All English PubMed abstracts related to Phase I trials in oncology that evaluated the effect of Anti-neoplastic agents on adults are extracted. The key steps involved in this process are listed below :

- Summary information on the results of a query for any database of the PubMed database was obtained.

- Drug names were extracted from the titles of PubMed articles by using regular expressions.
- The paper referred to for the purpose of this work already contained data pre-processed in accordance with the above steps and was directly utilized.

3.2.3 Getting DrugBank data

The Drug Bank data is obtained from the official web source and is present in an XML format. From this data the *drug pharmacologic category* and *drug molecular target* are extracted.

As mentioned in previous sections, drug pharmacologic categories are assigned based on the kind of action the drug has in the body while drug molecular targets are molecules in the body that are associated with a particular disease process that can be addressed by a drug to produce a desired therapeutic effects.

The DrugBank data is mined/queried to extract category and target details of ANAs as binary variables, for those ANAs which are present in the cleaned PubMed data obtained in Step 1.

3.2.4 Combining data from PubMed and DrugBank

The data obtained from the two sources is combined in using a left-join to generate a data-set containing information about Phase 1 clinical trials *and* the pharmacologic information (drug category target) of each anti-neoplastic agent. The criterion used for joining was the common alias of each ANA.

This comprehensive data-set will serve as the basis for the rest of the work. This data-set was subjected to basic pre-processing in which rows containing N/A values are dropped and certain columns which were deemed irrelevant (based on references) were also removed.

3.3 Cox Regression

3.3.1 The need for multivariate statistical modeling

In clinical investigations, there are many situations, where several known quantities (known as **covariates**), potentially affect prognosis of events such as drug approval and patient health. Statistical modeling is a frequently used tool that allows to analyze survival or approval with respect to several factors simultaneously. Additionally, statistical model provides the effect size for each factor.

The cox proportional-hazards model is one of the most important methods used for modeling historical data. The next section introduces the basics of the Cox regression model.

3.3.2 Basics of Cox Regression

Cox regression (or proportional hazards regression)[11] is method for investigating the effect of several variables upon the time a specified event takes to happen.

In the context of an outcome such as death this is known as Cox regression for survival analysis.

The method does not assume any particular survival model but it is not truly non-parametric because it does assume that the effects of the predictor variables upon survival are constant over time and are additive in one scale. Essentially, this model states that risk of the event in any group is a constant multiple of the risk in any other. In other words, if a drug has a risk of non-approval at some initial time point that is twice as high as that of another drug, then at all later times the risk of non-approval remains twice as high.

The purpose of the model is to evaluate simultaneously the effect of several factors on survival. In other words, it allows us to examine how specified factors influence the rate of a particular event happening (e.g., infection, death, drug approval) at a particular point in time. This rate is commonly referred as the **hazard rate**[12]. Predictor variables (or factors) are usually termed *covariates* in the survival-analysis literature.

The Cox model is expressed by the hazard function denoted by *h(t)*. Briefly, the hazard function can be interpreted as the risk of dying at time *t*. It can be estimated as shown in the equation 3.1:

$$h(t) = h_0(t) \times exp(b_1x_1 + b_2x_2 + ... + b_px_p) \tag{3.1}$$

where,

- *t* represents the survival time
- *h(t)* is the hazard function determined by a set of p covariates $(x_1,x_2,...,x_p)$

- The coefficients $(b_1,b_2,...,b_p)$ measure the impact (i.e., the effect size) of covariates
- The term h_0 is called the baseline hazard. It corresponds to the value of the hazard if all the x_i are equal to zero (the quantity *exp(0)* equals 1). The 't' in *h(t)* reminds us that the hazard may vary over time.

The Cox model can be written as a multiple linear regression of the logarithm of the hazard on the variables x_i, with the baseline hazard being an 'intercept' term that varies with time.

The quantities $exp(b_i)$ are called **Hazard ratios (HR)**. A value of b_i greater than zero, or equivalently a hazard ratio greater than one, indicates that as the value of the *i*th covariate increases, the event hazard increases and thus the length of survival decreases.

Put another way, a hazard ratio above 1 indicates a covariate that is positively associated with the event probability, and thus negatively associated with the length of survival.

In summary,

- HR = 1: No effect
- HR $<$ 1: Reduction in the hazard
- HR $>$ 1: Increase in Hazard

Cox Regression has been chosen for this project because the data being used was collected over many decades (from 1970-2017) and the objective is to understand the effect of the predictors on drug approval.

In our case, the hazard is defined by drug approval. Therefore, a HR$<$1 indicates reduced risk/chance of approval whereas a HR$>$ 1 indicates an increased risk/chance of approval.

3.4 Lasso Penalization

3.4.1 Bias-Variance Trade-Off in Multiple Regression

In the simple linear regression model, the aim is to predict n observations of the response variable, Y, with a linear combination of m predictor variables, X, and a normally distributed error term with variance σ^2 as shown in equation 3.2:

$$\begin{aligned} Y &= X\beta + \varepsilon \\ \varepsilon &\sim N\left(0, \sigma^2\right) \end{aligned} \tag{3.2}$$

As we don't know the true parameters, β, we have to estimate them from the sample. In the Ordinary Least Squares (OLS)[15] approach, we estimate them as β in such a way, that the sum of squares of residuals is as small as possible. In other words, we minimize the following loss function as shown in equation 3.3:

$$L_{OLS}(\hat{\beta}) = \sum_{i=1}^{n} \left(y_i - x_i'\hat{\beta}\right)^2 = \|y - X\hat{\beta}\| \tag{3.3}$$

In order to obtain the famous OLS parameter estimates, β_{OLS}=$(XX)^{-1}(XY)$, there are two critical characteristics of estimators to be considered: the bias and the variance.

The bias is the difference between the true population parameter and the expected estimator, as shown in equation 3.4:

$$\text{Bias}\left(\hat{\beta}_{\text{OLS}}\right) = E\left(\hat{\beta}_{\text{OLS}}\right) - \beta \tag{3.4}$$

Bias measures the accuracy of the estimates. Variance, on the other hand, measures the spread, or uncertainty, in these estimates, as shown in equation 3.5:

$$\text{Var}\left(\hat{\beta}_{\text{OLS}}\right) = \sigma^2 \left(X'X\right)^{-1} \tag{3.5}$$

The unknown error variance σ^2 can be estimated from the residuals, as shown in equation 3.6:

$$\begin{aligned} \hat{\sigma}^2 &= \frac{e'e}{n-m} \\ e &= y - X\hat{\beta} \end{aligned} \tag{3.6}$$

Both the bias and the variance are desired to be low, as large values result in poor predictions from the model. In fact, the model's error can be decomposed into three parts: error resulting from a large variance, error resulting from significant bias, and the remainder - the unexplained part.

The OLS estimator has the desired property of being unbiased. However, it can have a huge variance. Specifically, this happens when:

- The predictor variables are highly correlated with each other
- There are many predictors. This is reflected in the formula for variance given above: if m approaches n, the variance approaches infinity.

The general solution to this is to reduce variance at the cost of introducing some bias. This approach is called **regularization** and is almost always beneficial for the predictive performance of the model.

As the model complexity, which in the case of regression can be thought of as the number of predictors, increases, the estimates variance also increases and the bias decreases. Unbiased OLS is far from optimum at handling such cases. That's why we regularize: to lower the variance at the cost of some bias.

3.4.2 Lasso Penalty

Lasso[16] stands for *Least Absolute Shrinkage and Selection Operator*. As the name suggests, this model uses shrinkage i.e. it shrinks regression coefficients toward zero by penalizing the regression model with a penalty term called L1-norm, which is the sum of the absolute coefficients. The penalty has the effect of forcing some of the coefficient estimates, with a minor contribution to the model, to be exactly equal to zero.

This particular type of regression is well-suited for models showing high levels of multi-collinearity or when you want to automate certain parts of model selection, like variable selection/parameter elimination. In our case, this will enable us to remove such features from the model and thereby reduce its complexity.

3.4.2.1 Penalty specification

The Lasso penalty can be defined by the equation, shown in equation 3.7:

$$L_{\text{lasso}}(\hat{\beta}) = \sum_{i=1}^{n} \left(y_i - x_i'\hat{\beta}\right)^2 + \lambda \sum_{j=1}^{m} \left|\hat{\beta}_j\right| \tag{3.7}$$

Solving this for β gives the lasso regression estimates β_{lasso}=(XX+I)$^{-1}$(XY), where I denotes the identity matrix.

The λ parameter is called the regularization penalty. As λ becomes larger, the variance decreases, and the bias increases. In order to decide the level of acceptable bias in a way the minimises vaiance to the optimal value, the right value of λ must be chosen. Essentially, λ controls the strength of the lasso penalty. The shrinkage can be varied as λ varies:

- When λ = 0, no parameters are eliminated. The estimate is equal to the one found with linear regression.
- As λ increases, more and more coefficients are set to zero and eliminated (theoretically, when λ = ∞, all coefficients are eliminated).
- As λ increases, *bias* increases.
- As λ decreases, *variance* increases.

3.4.2.2 Choice of regularization parameter λ

Essentially, the amount of the penalty to be applied to a regression model can be fine-tuned using λ. Selecting an optimum value for λ is therefore critical.

There are two ways to select optimum λ - **minimize information criteria** or **cross-validation**. An information criterion is a measure of the quality of a

statistical model. It takes into account how well the model fits the data and the complexity of the model. Information criteria are used to compare alternative models fitted to the same data set. All else being equal, a model with a lower information criterion is superior to a model with a higher value. Cross-validation, sometimes called rotation estimation or out-of-sample testing, is any of various similar model validation techniques for assessing how the results of a statistical analysis will generalize to an independent data set. It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice.

The information criterion approach emphasizes the model's fit to the data i.e. how well a model can predict data points you've already used to estimate its parameters, while the cross-validation approach is more focused on its predictive performance i.e. how well a model can predict new data points, for which it hasn't yet seen the true value of the dependent variable.

For the purpose of this work, **cross-validation** is chosen to estimate optimum λ because the objective is to predict whether a drug is approved or not - this means that better predictive performance is more important to us than goodness of fit. Cross-validation enables us to select the value of λ that minimizes the cross-validated sum of squared residuals.

3.4.2.3 Algorithm to compute λ using cross-validation

Choose a set of P values of λ to test, split the data-set into K folds and follow the algorithm below. Here, β_{Lasso} denotes the coefficients of predictor variables obtained after performing Lasso penalization.

Algorithm 1 Optimum lambda value using cross-validation

for p in 1:P: **do**

    for k in 1:K: **do**
        keep fold k as bold-out data
        Use remaining folds and $\lambda = \lambda_p$ to estimate β_{Lasso}
        Predict hold out data: $y_{test,k} = X_{test,k} * \beta_{Lasso}$
        Compute a Sum of Squared Residuals: $SSR_k = ||y - y_{test,k}||^2$
end for k
Average SSR over the folds from k=1 to k=K: $SSR_p = (\sum SSR\text{k})/k$
end for *p*

Choose optimal value : λ_{opt}**= Minimum(SSR**$_p$)

3.5 Feature selection using Lasso Penalized Cox Regression

3.5.1 Need for feature selection

The data-set being used consists of 1415 features. For such high dimensionality feature selection is essential because :

- It reduces the complexity of the model and makes it easier to interpret.
- Choosing the right subset of features improves accuracy.
- It reduces over-fitting.

3.5.2 Algorithm for feature selection

1. Fit a *k-fold* Cross-validated Lasso Penalized Cox Regression model to the input data with *k=100* (obtained via reference).
2. Obtain a set of residual sum-of-squares corresponding to the λ values computed for each fold.
3. Choose the optimum λ as the one that minimises the sum-of-square residuals.
4. Fit a regular Lasso Penalized Cox Regression model (no cross-validation) to the input data with the λ obtained in the previous step as one of the inputs.
5. A set of coefficients - one for each feature - is obtained as the output from the previous step (similar to most regression techniques).
6. Select only those features having a coefficients$>$0.
7. The coefficients of the selected features are then used for computing likelihood of approval in further stages.

3.6 Computing likelihood of FDA approval

The output coefficients of a Cox Regression model (obtained in the previous section) are called Hazard Ratios(HR). These are computed as the ratio of risks for approval and non-approval at any particular point in time and tell us how a given feature affects FDA approval of the ANA.

A feature having a HR$>$1 indicates that it's positively correlated with event probability and negatively with survival time of the subject. Since in our case the

event is FDA approval and subject is the ANA being considered, a HR$>$1 means that the feature contributes to a shorter time to drug approval, which is desirable. The reverse is true for HR$<$1.

A linear combination of HRs for each ANA gives us the likelihood of FDA approval for said ANA. We call this the approval score.

The HR obtained from cox regression is typically in a logarithmic form. This does not facilitate easy interpretation of results or calculation of overall likelihood of approval. In order to overcome this limitation, the exponent of each HR is taken to get the actual non-logarithmic value before combining them.

The likelihood of FDA approval is estimated for each ANA present in the input data. This score is the final output of our regression algorithm and is will subsequently be used to classify the ANAs into likely to be approved/non-approved for release.

3.7 Threshold calibration

Classification in predictive modeling typically involves predicting a class label. Many machine learning algorithms, especially ones involving regression, predict a probability or scoring of class membership, and this must be interpreted before it can be mapped to a crisp class label. This is achieved by using a **threshold**, such as 0.5, where all values equal or greater than the threshold are mapped to one class and all other values are mapped to another class.

The standard 0.5 threshold holds good only when there is an equal distribution of classes in the input data and the output is a probability value. In our case the

output (approval score) is not a probability value between 0-1 and and the input has a class imbalance problem - only 26 of the observations are FDA approved so the output is likely to be biased in favor of non-approval. In order to address these issues, we need to calibrate the threshold i.e. compute a threshold suited to our data.

3.7.1 Computing classification threshold

The first step is to identify a list of possible threshold values. This can be done by looking at the range of approval scores obtained and generating a linear, evenly-spaced sequence with the same.

Subsequently, we define our null and alternate hypothesis. The null hypothesis states that for the selected threshold value, there is no difference between the two classes - approved/non-approved - while the alternate hypothesis states that a significant difference exists between the two. The idea is to choose a threshold value that has the strongest support for the alternate hypothesis.

The **Logrank test** is used to determine the strength of the alternate hypothesis for each threshold value in the list. For each of the said thresholds, the logrank test is applied and a **p-value** is obtained as the output. As is common in most hypothesis tests, the smaller the p-value the more strongly the alternate hypothesis holds true and the null hypothesis can be rejected.

Therefore, the threshold value that produces the smallest p-value in the logrank test is chosen as the cut-off value for our classification.

CHAPTER 4

IMPLEMENTATION

This chapter discusses the steps involved in the implementation of the system architecture, described in the previous chapter. The process is described in a modular fashion, as showcased in the architecture.

4.1 Data extraction and pre-processing

This section outlines the steps involved in accessing the **DrugBank** and **PubMed** databases, extracting the necessary data from them and pre-processing the data by combining it into a single data-set which will serve as the input for subsequent actions.

- The **EUtilsSummary()**[20] method of the *RISmed* package was used to get summary information on the results of a query for any database of the PubMed database.
- Drug names were extracted from the titles of PubMed articles by regular expressions by using the *stringr*[20] package of R.
- The paper referred to for the purpose of this work already contained data pre-processed in accordance with the above steps and was directly utilized.
- The **ElementTree**[21] library of Python is used to mine/query the DrugBank data and extract category and target details of ANAs as binary variables, for those ANAs which are present in the cleaned PubMed data obtained in Step 1.

- The **inputs** given to the mining code are the selected list of ANAs, the query in **XPath** syntax, the DrugBank data and the name of the output file.
- The **output** is a data-set containing the ANAs and drug pharmacological categories as binary variables indicating membership.
- The drug molecular target details are extracted in a similar manner by changing the XPath query.
- The data obtained from the two sources is then combined in using a left-join on the common alias of each ANA to generate a data-set containing information about Phase 1 clinical trials *and* the pharmacologic information (drug category target) of each anti-neoplastic agent. These operations are performed using the **left_join** method of the *dplyr* package in R.

This data-set contains **462** observations/rows and **1415** features/columns. Each row represents an ANA. Some of the key features of this data-set are:

- COMMON_DRUGBANK_ALIAS : This column gives the commonly used name for each ANA, as specified in DrugBank
- FDA_APPROVAL : Whether the ANA has been approved or not, as obtained from PubMed data. 1 denotes approval and 0 denotes non-approval.
- DELAY_FROM_OLDEST_PMID_TO_FDA_APPROVAL : The number of days an ANA took to to get FDA approval, after Phase 1 clinical trials. This is also obtained from PubMed.
- Columns starting with ***category.*** represent drug pharmacologic category and those staring with ***target.*** represent drug target. These are binary features - 1

denotes that the ANA belongs to the category/target and 0 denotes that it is not a member of said feature.

4.2 Feature Selection

The data-set being used consists of 1415 features. For such high dimensionality feature selection is essential. This section outlines the steps to perform feature selection in R, using a combination of Lasso Penalized Cox Regression models.

4.2.1 Obtaining optimum λ

The optimum value of regularization parameter is obtained by fitting a cross-validated Lasso Penalized Cox regression model to the data, using the **cv.glmnet()** method of the *glmnet* package in R.

FUNCTION DEFINITION : ***cv.glmnet(x, y, family, nfolds, alpha)***

- *x* : Input matrix where each row is an observation vector.
  Here, *x* is the rows of the data-set in a matrix form.
- *family* : Either a character string representing one of the built-in families representing the type of distribution used by the regression such as ”gaussian”,”poisson” etc. Here the value is **”cox”**.
- *y* : Response variable. For family= ”cox” a Surv object from the survival package should be specified. The Surv object is obtained using the function

Surv(time,event) which tells us about the state of the ”event” variable for each interval in ”time”. In our case :

  - **time** = Delay in months to FDA approval
  - **event** = FDA approval

- *nfolds* : Number of folds to use in cross-validation. Here **nfolds = 100**.
- *alpha* : Type of penalty to use. Here the **alpha = 1**, which denotes lasso penalization.
- **Output :** Optimum λ value

4.2.2 Estimating model coefficients

Model coefficients are obtained by fitting a regular (non cross-validated) Lasso Penalized Cox Regression model to the data, using the λ value previously obtained. The **glmnet()** method of the *glmnet* package in R is used.

FUNCTION DEFINITION: ***glmnet(x, y, family, lambda, alpha)***

- *x, y, family* and *alpha* remain the same as the previous step.
- *lambda* - Optimum value of λ, as obtained from **cv.glmnet()** in the previous step.
- **Output :** The coefficients calculated for each feature.

4.2.3 Selected features

Only those features having a coefficients>0 are selected. A total of **31** features were selected from 1415, as shown in figure 4.1.

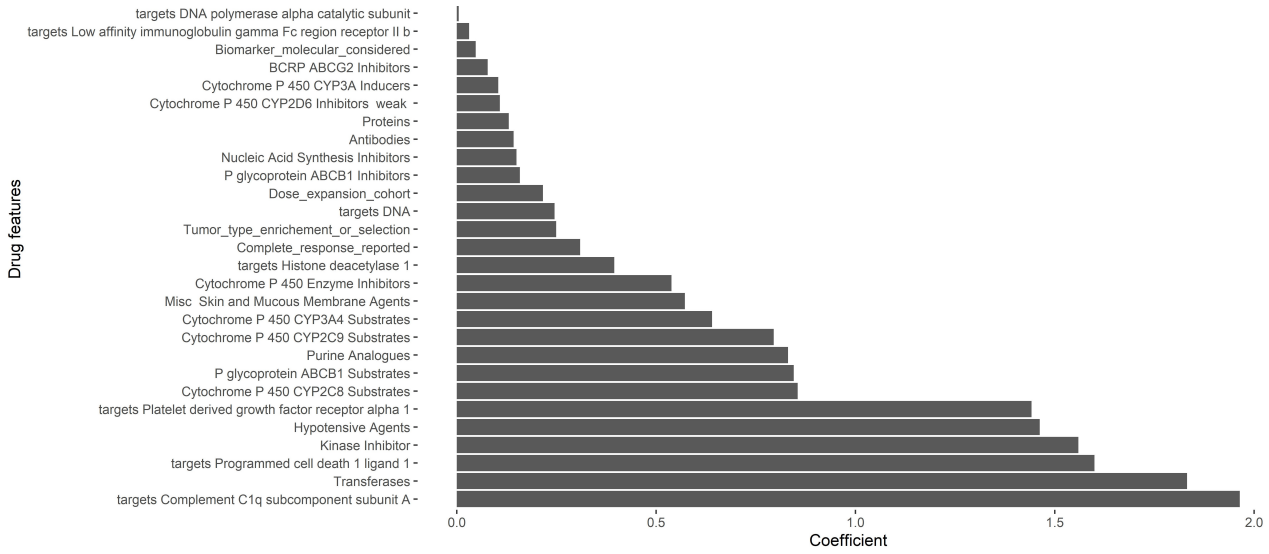

FIGURE 4.1: Coefficients of selected features

4.3 Computing likelihood of FDA approval

This section outlines the steps to compute the approval score which determines likelihood of FDA approval, using the model coefficients (called **hazard ratios**), in R, as shown below:

- To compute non-logarithmic HRs, the data was modified to contain only the features obtained during feature selection using **select()** function of the *dplyr* package.

- A matrix was prepared from the test data containing the the selected features. This was done using **as.matrix()** method. The dimensions of the matrix are **135 rows x 31 columns**. Let this be *test_mat*.

- Another matrix containing the non-zero coefficient values was created. The coefficients were represented in a single column. The dimensions of the matrix are **31 rows x 1 column**. Let this be *coefficients_nonzero*.

- The **approval score** for each observation/row in the data-set was calculated by taking the ***exponent*** of the matrix multiplication of *test_mat* and *coefficients_nonzero*.

4.4 Threshold calibration

This section describes the steps involved in computing a classification threshold on the basis of our approval score, using the **logrank test**, as given below:

- Generate a linear sequence of possible threshold values using the **seq()** method in R, which generates an evenly spaced sequence of a specified number of items, in the given range.

- The range of thresholds can be computed by looking at the maximum and minimum values of approval scores for each observation in the data-set. Outliers can be ignored, if few in number.

- Place the generated thresholds in a data frame using the as.data.frame() method. Create and additional column to store the p-value when it's computed.

- For each threshold value in the list, we need to compute a p-value using the logrank test. This can be done using the **survdiff()** method of the *survival* package in R.
- **survdiff()** takes in the null and alternate hypothesis as a formulaic input representing the survival model of choice.
- The output of **survdiff()** is a chi-square value. In order to obtain the p-value (to gauge significance) for the test, we use the **pchisq()** method which takes in a chi-square score as input and returns the correct p-value for the same.
- A list of p-values can be computed in this manner, one for each threshold in the list. Finally, the thresholds are sorted in ascending order of p-values using the **order()** method of *dplyr* and the threshold with the smallest p-value is chosen as the optimum one.

Based on the results of the logrank test,the threshold value was chosen as **4.55**.

All ANAs having an FDA score greater than 4.55 are classified as likely to be approved for release and those having approval scores below are classified as likely to be disapproved for release.

CHAPTER 5

RESULTS AND DISCUSSION

This chapter discusses the performance metrics used to evaluate the model built in the preceding sections and the results obtained when it's compared to standard approaches currently in use.

The **Concordance index** or **C-index** is commonly used to evaluate the predictive power of models which utilize survival data. In our case, it measures the model's ability to correctly provide a reliable hazard ratio. The following sections describe the working of the concordance index and it's computation in R.

5.1 Concordance index

Developing a prognostic model for clinical applications typically requires mapping an individual's set of covariates to a measure of the risk that he or she may experience the event to be predicted. Many scenarios, however, especially those involving clinical outcomes, are better described by explicitly accounting for the timing of these events, as well as their probability. As a result, in these cases, traditional classification or ranking metrics may be inadequate to inform model evaluation or selection.

To address this limitation we use the concordance index (C-index), which summarises how well a predicted risk score describes an observed sequence of events. In our case it measures the model's ability to correctly provide a reliable hazard ratio.

5.1.1 Computing C-index

The C-index[17] represents the global assessment of the model discrimination power - the model's ability to correctly provide a reliable ranking of the survival times based on the individual risk scores.

The intuition behind C-index is as follows : for drug i, our model assigns a risk score η_i. If our model is any good, drugs which had shorter time-to-approval should have higher risk scores.

Boiling this intuition down to two drugs: the drug with the higher risk score should have a shorter time-to-approval. Note that in the case of this work, the risk is the event of drug approval.

5.1.1.1 Algorithm

For every pair of observations i and j (with i $\neq$ j), let time-to-event be T.

- If both T_i and T_j are not censored, then we can observe when both drugs got approval. We say that the pair (i, j) is a concordant pair if $\eta_i > \eta_j$ and $T_i < T_j$, and it is a discordant pair if $\eta_i > \eta_j$ and $T_i > T_j$.
- If both T_i and T_j are censored, then we don't know which drug got the approval first (if at all), so we don't consider this pair in the computation.
- If one of T_i and T_j is censored, we only observe one disease. Let's say we observe drug i getting approval at time T_i, and that T_j is censored (true for vice-versa).

- If $T_j < T_i$, then we don't know for sure which got approval first, so we don't consider this pair in the computation.
- If $T_j > T_i$, then we know for sure that drug i got approval first. Hence, (i, j) is a concordant pair if $\eta_i > \eta_j$, and is a discordant pair if $\eta_i < \eta_j$.

The C-index can therefore be represented, as shown in equation 5.1:

$$c = \frac{\#\text{ concordant pairs}}{\#\text{ concordant pairs} + \#\text{ discordant pairs}} \tag{5.1}$$

The above equation can be summarized, as shown in equation 5.2:

$$c = \frac{\sum_{i \neq j} 1\{\eta_i < \eta_j\} 1\{T_i > T_j\} d_j}{\sum_{i \neq j} 1\{T_i > T_j\} d_j} \tag{5.2}$$

The value of C-index varies from 0.5 to 1:

- A value below 0.5 indicates a very poor model
- A value of 0.5 means that the model is no better than predicting an outcome than random chance
- Values over 0.7 indicate a good model and over 0.8 indicate a strong model
- A value of 1 means that the model perfectly predicts those group members who will experience a certain outcome and those who will not

5.1.2 IPCW Estimator

The normal assumption while calculating C-index is that the censoring is non-informative. However, this may not always be the case. The Inverse Probability of Censoring Weighted (IPCW)[19] estimator is used to account for the presence of informative censoring. The basic idea of this estimator is to correct for censored subjects by giving extra weight to subjects who are not censored. In this way, the model is fit as if censoring was absent. Its interpretation is very intuitive, and therefore easy to understand.

IPCW is used when the cost of failing to predict a positive outcome (like a test for cancer) is higher than benefit of correctly predicting a negative outcome. This is an extremely important consideration in this project since the cost of incorrectly predicting positive drug approval can cause potentially life saving medicines to go undeveloped.

5.2 Performance analysis

The **concordance.index()** method of the *survcomp* package of R is used for estimating non-weighted C-index and the **cindex()** method of *pec* package is used to estimate weighted C-index (IPCW). The results of performance analysis of our model gave a C-index of **0.898** and IPCW value of **0.890**.

It can be observed from the C-index and IPCW values obtained that our model is strong and provides reliable hazard ratios.

5.3 Comparison with standard approaches

In order to assess the overall efficacy of the model, it was compared to two standard algorithms used in analysis of clinical trial data - the EffTox approach and Random Forests for Survival, Regression, and Classification (RF-SRC). Two separate models were trained using these approaches, with same input data. Concordance indices were computed for both EffTox and RF to determine relative performance.

5.3.1 EffTox

EffTox[24] is a Bayesian adaptive dose-finding trial design that jointly scrutinises binary efficacy and toxicity outcomes. According to standard practices, the features commonly considered in the EffTox model are clinical activity, complete drug response and toxicity. *It is to be noted that the EffTox approach only tells us which features to use - not the choice of the model.*

A Lasso Penalized Cox Regression model was fit on input data containing only the above mentioned features. The procedures used were the same as previously elucidated. C-index and IPCW were computed for this approach, as shown below. It can be observed that the values are lower than what was obtained for our model.

5.3.2 RF-SRC

Random Forests for Survival[25], Regression, and Classification (RF-SRC) is an ensemble tree method for the analysis of data sets using a variety of models. As is

well known, constructing ensembles from base learners such as trees can significantly improve learning performance.

RF-SRC extends the common Random Forests method and provides a unified treatment of the methodology for models including right censored survival (single and multiple event competing risk), multivariate regression or classification, and mixed outcome (more than one continuous, discrete, and/or categorical outcome).

The *randomForestSRC* package in R was used to perform classification. A weighted concordance index/IPCW was computed for this model to understand relative performance when compared to cox regression. The C-index obtained was **0.81** and IPCW value was **0.818**.

The C-index and IPCW values for all the approaches are listed below. It can be observed from the results that our model performs better than the standard approaches taken to predict drug approval.

Methodology	**C-index**	**IPCW**
Cox Regression	0.898	0.890
Efftox	0.792	0.684
Rf-src	0.81	0.818

TABLE 5.1: Comparison with standard approaches

CHAPTER 6

CONCLUSION AND FUTURE WORK

This report has discussed the detailed design, implementation and related algorithms for a project to predict the likelihood drug approval from Phase 1 oncological clinical trials using machine learning. The drugs where classified into 2 classes namely likely to be FDA approved and not likely to be FDA approved by using a calibrated score obtained from a Lasso Penalized Cox Regression model.

Since one of the biggest shortcoming of this project is the relatively small size of the training data, as a part of our future work, we intend to focus on improving upon the predictions by implementing a deep neural network by collating even more data to the existing data-set.

Appendix A

EffTox

Traditionally, Phase I oncology trials evaluate the safety profile of a novel agent and identify a maximum tolerable dose based on toxicity alone. With the development of biologically targeted agents, investigators believe the efficacy of a novel agent may plateau or diminish before reaching the maximum tolerable dose while toxicity continues to increase. This motivates dose-finding based on the simultaneous evaluation of toxicity and efficacy - hence the name EffTox model. This approach can also be extended to identifying drug approval by considering data points related to the clinical and molecular activity and safety of a drug.

The most desirable drug is of high efficacy, low toxicity (side effects), low chance of drug resistance, low cost, and low deleterious effect on the environment, e.g., no re-activation by bacterial species after human use. Drugs can be classified broadly into restorative and disruptive drugs. Restorative drugs aim to restore cellular functions. In contrast to restorative drugs, disruptive drugs are intended to disrupt cell growth and proliferation and to induce apoptosis. These drugs are used in the fight against pathogens or malignant cells such as those causing cancer.

Drug efficacy of disruptive drugs can be directly measured by the proportion of cancer cells or pathogens killed, from which one can obtain an estimate of the propensity of cancer cell or pathogen mortality. From a transcriptomic perspective, drug efficacy can be defined as an index of disruption, measured by the drug-induced difference in transcriptomic profile of malignant cells before and after drug use, especially the induction of apoptosis genes and activation of apoptosis pathways. The drug toxicity could be conceptually defined as drug-induced transcriptomic differences of normal cells before and after drug administration. This is is the basis for feature selection in the EffTox approach.

Appendix B

RF-SRC

Random Forests for Survival, Regression, and Classification (RF-SRC) is an ensemble tree method for the analysis of data sets using a variety of models. As is well known, constructing ensembles from base learners such as trees can significantly improve learning performance. Ensemble learning can be further improved by injecting randomization into the base learning process — a method called Random Forests. RF-SRC extends the classic Random Forests method and provides a unified treatment of the methodology for models including right censored survival (single and multiple event competing risk), multivariate regression or classification, and mixed outcome (more than one continuous, discrete, and/or categorical outcome). When one continuous or categorical outcome is present, the model reduces to uni-variate regression or classification respectively. When no outcome is present, the model implements unsupervised learning.

RF-SRC introduces new split rules for each model and gives the user the ability to define and code custom split rules. Deterministic or random splitting is available for all models. Variable predictiveness can be assessed using variable importance measures for single as well as grouped variables. Variable selection is implemented using minimal dept. Missing data (for x-variables and y-outcomes) can be imputed on both training and test data.

B.1 Methodology

Building a Random Forests model involves growing a binary tree using user supplied training data and parameters. The data types must be real valued,

discrete or categorical. The response can be right-censored time and censoring information, or any combination of real, discrete or categorical information. The response can also be absent entirely. The resulting forest contains many useful values which can be directly extracted by the user and parsed using additional functions.

The process iterates over *ntree*, the number of trees that are to be grown. In practice, the iteration is actually parallelized and trees are grown concurrently, not iteratively. The recursive nature of the algorithm is reflected in the repeated calls to split a node until conditions determine that a node is terminal. Another key aspect of the algorithm is the injection of randomization during model creation. Randomization reduces variation. Bootstrapping at the root node reduces variation. Feature selection is also randomized with the use of the parameter *mtry*. In the recursive algorithm N is defined as the number of records in the data set, and P as the number of x-variables in the data set. The parameter *mtry* is such that $1 <= mtry <= P$.

At each node, the algorithm selects *mtry* random x-variables according to a probability vector *xvar.wt*. The resulting subset of x-variables are examined for best splits according to a spli trule. The parameter *nsplit* also allows one to specify the number of random split points at which an x-variable is tested. The depth of trees can be controlled using the parameters *nodesize* and *nodedepth*. The parameter *nodesize* ensures that the average *nodesize* across the forest will be at least *nodesize*. The parameter *nodedepth* forces the termination of splitting when the depth of a node reaches the value specified. Node depth is zero-based from the root node onward. Reasonable models can be formed with the judicious selection of *mtry*, *nsplit*, *nodesize*, and *nodedepth* without exhaustive and deterministic splitting.

Appendix C

Tools and packages used

This enitre work was carried out using the R language. The key packages and methods used are illustrated below.

- **glmnet** - Package containing extremely efficient procedures for fitting the entire lasso or elastic-net regularization path for linear regression, logistic and multinomial regression models, Poisson regression, Cox model,multiple-response,Gaussian and the grouped multinomial regression.
- **survival** - Contains the core survival analysis routines, including definition of Surv objects, Kaplan-Meier and Aalen-Johansen (multi-state) curves, Cox models, and parametric accelerated failure time models.
- **pec** - Used for validation of risk predictions obtained from survival models and competing risk models based on censored data using inverse weighting and cross-validation.
- **survcomp** - Package containing functions to perform the performance assessment and comparison of risk prediction (survival) models.
- **dplyr** - Grammar of data manipulation, providing a consistent set of verbs that help you solve the most common data manipulation challenges.

REFERENCES

1. Biotechnology Innovation Organization: Clinical Development Success Rates(2016) 2006-2015,

2. Jonsson B, Bergh J(2012): Hurdles in anticancer drug development from a regulatory perspective. Nat Rev Clin Oncol 9:236-243, .

3. Paul SM, Mytelka DS, Dunwiddie CT, et (2010): How to improve R and D productivity: The pharmaceutical industry's grand challenge. Nat Rev Drug Discov 9:203-214,.

4. Chakiba C, Grellety T, Bellera C, et al(2018): Encouraging trends in modern phase 1 oncology trials. N Engl J Med 378:2242-2243,.

5. Wishart DS, Feunang YD, Guo AC, et al:(2018) DrugBank 5.0: A major update to the DrugBank database . Nucleic Acids Res 46:D1074-D1082, 2018.

6. Lo, Andrew Siah, Kien Wong, Chi. (2019). Machine Learning with Statistical Imputation for Predicting Drug Approval. Harvard Data Science Review.

7. i, Youran and Qi Tang(2019). "Predicting Phase 3 Clinical Trial Results by Modeling Phase 2 Clinical Trial Subject Level Data Using Deep Learning." MLHC .

8. Jelena Gligorijevic, Djordje Gligorijevic, Martin Pavlovski, Elizabeth Milkovits, Lucas Glass, Kevin Grier, Praveen Vankireddy, Zoran Obradovic,(2019) Optimizing clinical trials recruitment via deep learning, Journal of the American Medical Informatics Association, Volume 26, Issue 11, Pages 1195–1202.

9. Kourou, Konstantina Exarchos, Themis Exarchos, Konstantinos Karamouzis, Michalis Fotiadis, Dimitrios. (2014). Machine learning applications in cancer prognosis and prediction. Computational and Structural Biotechnology Journal

10. Tibshirani R.(1996) Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society, Series B. ;58:267–288.

11. Cox, D.R.(1972): Regression models and life tables. Journal of the Royal Statistical Society, Series B 34(2), 187–220

12. Symons MJ, Moore DT.(2008) Hazard rate ratio and prospective epidemiological studies. J Clin Epidemiol.;55(9):893-89912393077

13. Wishart, D. S.(2018) et al. DrugBank 5.0: a major update to the DrugBank database for 2018. Nucleic Acids Res. 46, D1074–D1082 .

14. Canese K, Jentsch J, Myers C(2003): PubMed: The Bibliographic Database. In The NCBI Handbook. , Bethesda: National Center for Biotechnology Information, USA

15. Hutcheson, G. D. (1999). Ordinary least-squares regression. In The multivariate social scientist (pp. 56-113). SAGE Publications, Ltd., https://www.doi.org/10.4135/9780857028075

16. Chris Hans(2009), Bayesian lasso regression, Biometrika, Volume 96, Issue 4, , Pages 835–845, https://doi.org/10.1093/biomet/asp047

17. Bozdogan, H.(1987) Model selection and Akaike's Information Criterion (AIC): The general theory and its analytical extensions. Psychometrika 52, 345–370 . https://doi.org/10.1007/BF02294361

18. Yan X, Zhai L, Fan W (2013) C-index: a weighted network node centrality measure for collaboration competence. J Informetr 7:223–239

19. Robins JM, Finkelstein DM(2000). Correcting for noncompliance and dependent censoring in an AIDS Clinical Trial with inverse probability of censoring weighted (IPCW) log-rank tests. Biometrics ;56:779-788

20. Viechtbauer W (2010) Conducting meta-analyses in R with the meta for package. Journal of Statistical Software 36: 1–48.

21. G. van Rossum. Python library reference. http://www.python.org/doc/2.3.4/lib/lib.html.

22. Burke, A., N. Snook, D. J. Gagne II, S. McCorkle, and A. McGovern, (2020): Calibration of machine learning-based probabilistic hail predictions for operational forecasting. Wea. Forecasting, 35, 149–168, https://doi.org/10.1175/WAF-D-19-0105.1.

23. Lin RS, Leon LF (2017) Estimation of treatment effects in weighted log-rank tests. Contemp Clin Trials Commun 8:147–155

24. Brock K, Billingham L, Copland M, Siddique S, Sirovica M, Yap C.(2017) Implementing the EffTox dose-finding design in the Matchpoint trial. BMC Med Res Methodol. 112. doi: 10.1186/s12874-017-0381-x. PMID: 28728594; PMCID: PMC5520236.

25. Ehrlinger, J.,(2016) “ggRandomForests: Exploring Random Forest Survival”,arXiv e-prints,