Introduction

An Initial Coin Offering allows entrepreneurs to raise funds within a specific time frame from investors (Xu et al., 2021). Thus, ICOs can be highly attractive to entrepreneurs and investors. On one hand, entrepreneurs can fund projects while avoiding the complexity of transactions. At the same time, investors can liberally manage rewards by transferring purchased tokens to another investor (Xu et al., 2021).

However, ICOs can be risky, particularly for an investor. When an ICO fails to reach its funding target, investors lose part of the money their investment, and entrepreneurs would fail to implement their idea (Xu et al., 2021). Connor Sephton (2019) of the crypto platform "Currency.com" states that regulatory concerns, scams, and the absence of a tangible product are other elements that make ICOs risky.

According to Xu et al. (2021), predicting the outcome of an ICO using a machine learning model would be valuable for an investor when evaluating a campaign's performance before investing. Given the numerous factors which influence the performance of an ICO, there is an array of potential predictor variables for training a predictive model.

As per the case study by Panin et al. (2019), factors such as fundraising team size, the platform used to launch the campaign, and the team's social media engagement have different effects on the chances of an ICO meeting its funding target.

Hence, this work intends test and compare a set of machine learning models using a dataset containing information on ICOs.

Data Understanding & Pre-Processing

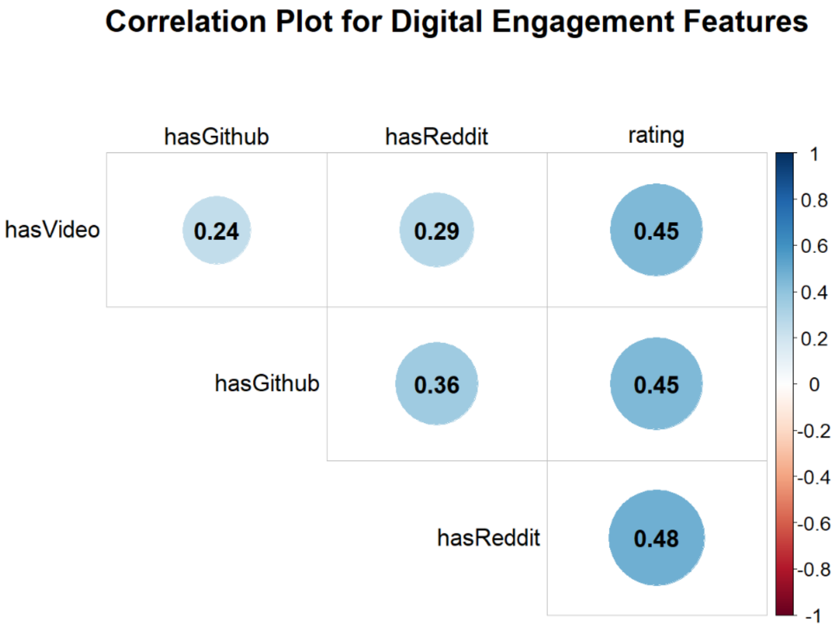
Importing the Dataset

The 16 features in the dataset provide information on 2,676 ICOs.

Feature	**Description**
ID	An ICO's ID.
success	An ICO's outcome.
brandSlogan	Fundraising team's slogan.
hasVideo	Whether a video was posted on the ICO's webpage.
rating	Expert investors' ratings of ICOs
priceUSD	ICO blockchain coin price.
countryRegion	Country where the fundraising team is based.
startDate	Campaign launch date.
endDate	Campaign end date.
teamSize	Number of individuals in the fundraising team.
hasGithub	Whether the team shared a link to the ICO's GitHub page.
hasReddit	Whether the team shared a link to the ICO's Reddit page.
platform	Platform used to launch the ICO.
coinNum	Number of blockchain coins to be issued
minInvestment	Whether there is a minimum investment requirement.
distributedPercentage	Percentage of issued blockchain coins distributed to investors.

Investigating the Relationships between Predictors

Most predictors do not exhibit significant linear relationships with low Pearson correlation coefficients. Below is a plot of low-moderate correlation values.

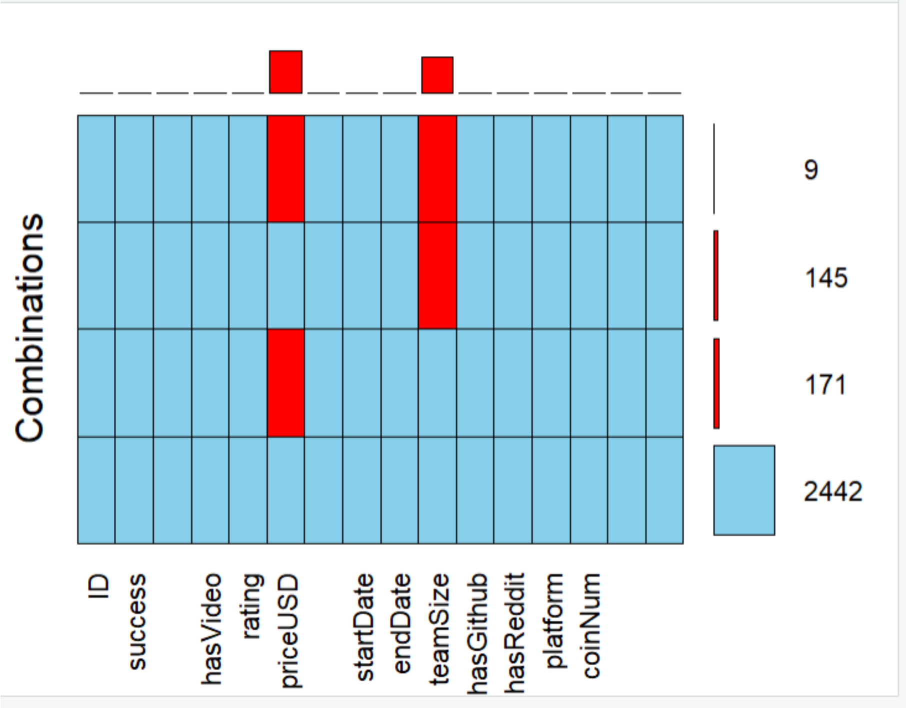


Fundraising teams that shared an ICO video and provided links to their official GitHub/Reddit webpages received a higher rating from expert investors.

Besides, the low-moderate correlation values confirmed the absence of multicollinearity issues when testing predictive models.

Examining the Missing Values

171 ICOs had an unknown coin price, 145 had an undefined team size, and 9 had both undefined.



Missing values were managed using models' built-in or other functionalities.

Dropping the ID Feature

The ID feature was dropped since it doesn't add value to the predictions.

Defining the Target Variable

The success column was defined as the target variable and converted to the factor type. 1,028 of the ICOs in the dataset reached their funding target.

success	
Failure	**Success**
1739	1028

Processing the Categorical Features

Online Engagement Features

The tables below show that 1,599 fundraising teams shared a link to their GitHub page while 1,751 shared one to their Reddit page. Also, 2,009 teams shared a video on the campaign's webpage.

hasGithub	
No	**Yes**
1168	1599

hasReddit	
No	**Yes**
1016	1751

hasVideo	
No	**Yes**
758	2009

minInvestment

The table shows that 1,254 teams set a lower investment limit for those interested in funding their idea.

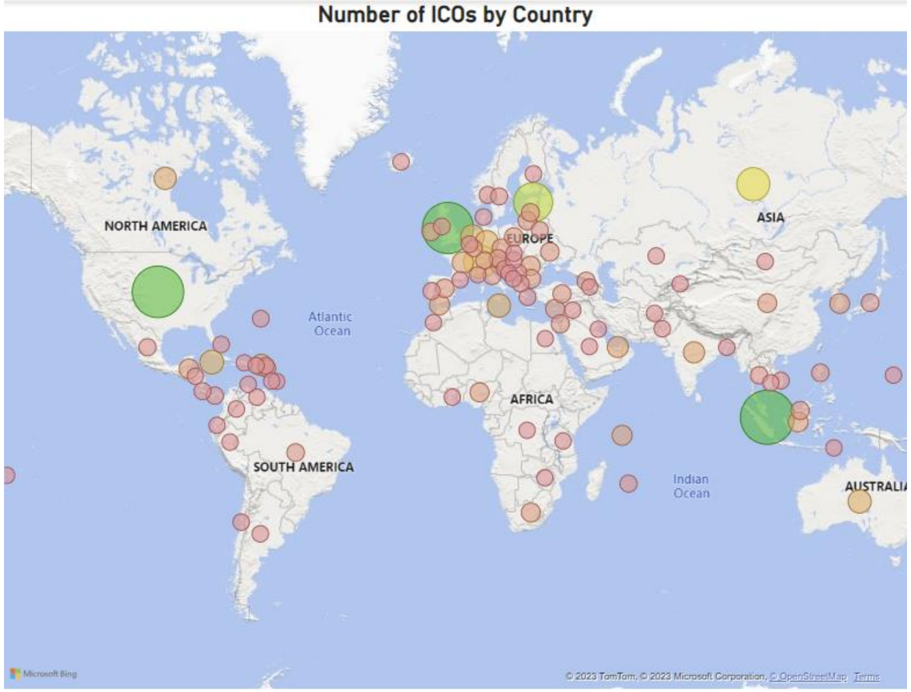
minInvestment	
No	**Yes**
1513	1254

countryRegion

The following pre-processing steps were taken:

- Conversion to uppercase.
- Replacement of null values with "Unknown".
- Correction of errors in spacing and grammar.

After pre-processing, the countryRegion feature had 116 unique countries. As evident from the map, some of the popular countries for launching ICOs are USA, UK, Singapore, Estonia, and Russia.


Number of ICOs by Country

Adding the ICO-Friendly Countries Feature

According to Offshore Protection (2021) and The Offshore Company (2019), the following countries are considered ICO-friendly:

- Switzerland
- Singapore
- Russia
- Estonia
- Gibraltar
- Cayman Islands
- Israel

Accordingly, the is_most_friendly feature was added to the dataset with a value of 1 assigned to ICOs originating from one of these countries. Otherwise, a 0 was assigned.

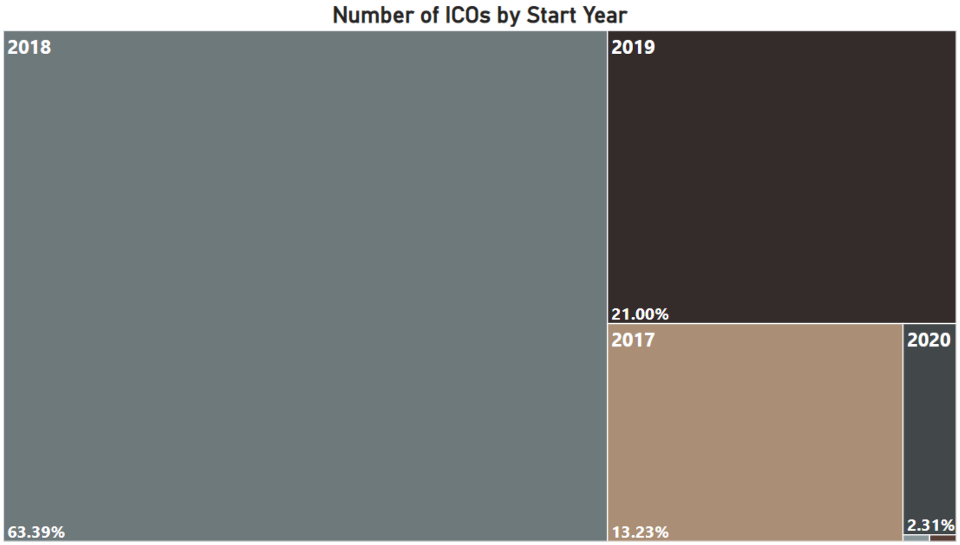
Adding the is_USA Feature

The USA has always been one of the most popular (but not the friendliest) countries for ICOs (Offshore Protection, 2021). Besides, 272 of the ICOs in this dataset were launched from the US. Thus, the logical is_USA feature was added to the dataset with ICOs launched from the U.S. being assigned a value of 1. Otherwise, a 0 was assigned.

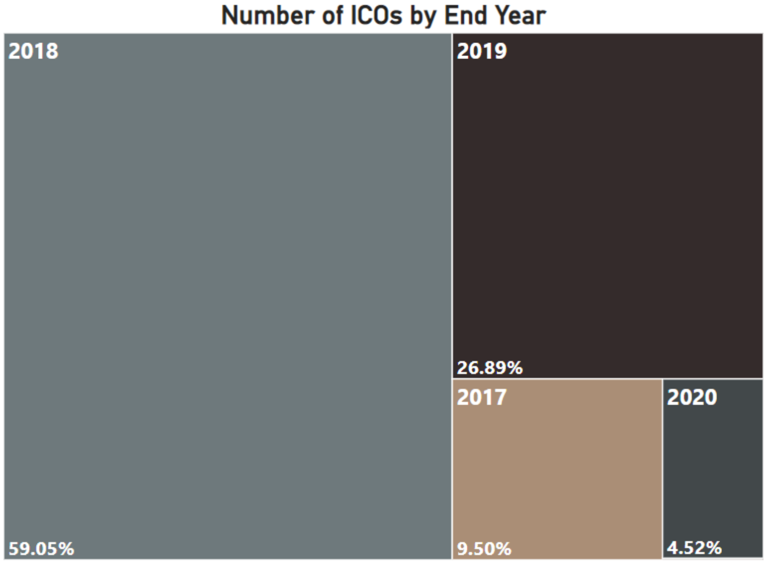
Start and End Dates

The startDate and endDate features were converted to the "Year/Month/Day" format.

According to the Treemaps below, around 85% of the ICOs in the dataset either started or ended in 2018 and 2019.



Number of ICOs by Start Year



Number of ICOs by End Year

ICOs with a startDate greater than their endDate had the date values swapped before extracting the year, month, and day as separate features.

Adding the ICO Duration Feature

The ICO duration feature (difference between the start and end dates) was created.

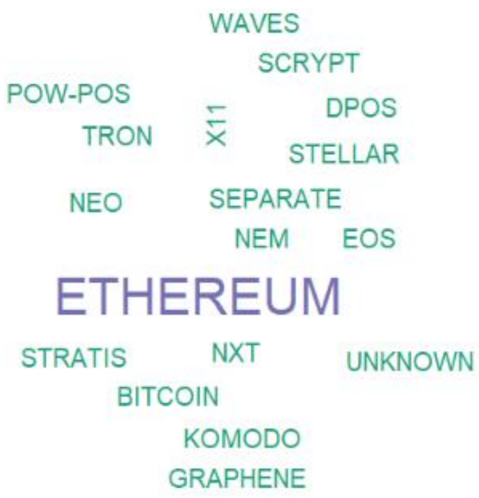
Platform

The following pre-processing steps were taken:

- Trimming the values and conversion to uppercase.
- Correction of spelling mistakes.
- Replacement of empty strings with "Unknown".
- Combination of unique values representing the same platform into one category (e.g., ETH and ETHEREUM as ETHEREUM).

Processing reduced the number of unique platforms from 130 to 96.

As evident from the word cloud below, the majority of ICOs were launched using the Ethereum platform.



Adding the is_ethereum Feature

The is_ethereum feature, which takes a value of 1 for ICOs launched using the Ethereum platform and 0 for others, was added to the dataset.

Processing the Numeric Features

Below are the statistical summaries of the numeric features in the dataset. Insights and steps taken to process each feature are also presented:

Before Processing							
Feature	**Min**	**Q1**	**Median**	**Mean**	**Q3**	**Max**	**NA's**
priceUSD	0	0.04	0.12	19.07	0.5	39384	180
Rating	1	2.7	3.1	3.13	3.7	4.7	0
coinNum	1.2	$5x10^7$	$2x10^8$	$8.5x10^{12}$	$6x10^8$	$2.26x10^{16}$	0
teamSize	1	7	12	13.19	17	75	146
distributedPercentage	0	0.4	0.55	1.102	0.7	869.75	0
Duration	-203	29	45	69.23	90	3722	0

priceUSD

Insights

- Half of the ICO blockchain coins had a price below $0.12.
- There are extreme outliers (seen in the scatterplot below and justified by the gap between the median and mean values) as well as 180 ICO coins with an undefined price.

Steps Taken

- Coins with prices of $4 or above were filtered out.

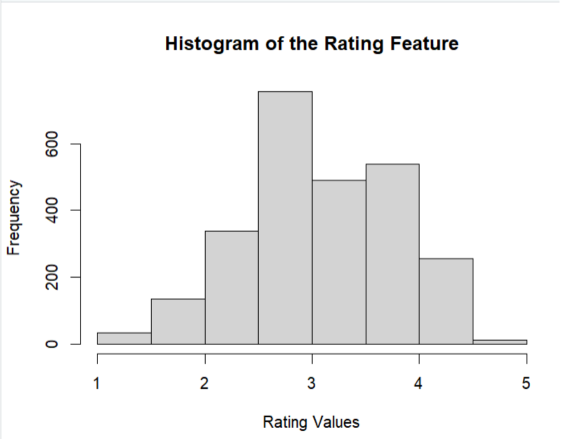
Rating

Insights

- On average, expert investors gave a 3.13 rating for the ICOs.
- Rating follows an approximate normal distribution (as per the histogram) with the mean and median being equal.
- The lowest rating was 1 while the highest was 4.7.

Steps Taken

- No further processing was performed.



coinNum

Insights

- Half of the fundraising teams released less than 2x10^8 coins upon launching their ICO.
- The substantial difference between the median and mean values provides evidence of extreme outliers (seen in the scatter plot below).

Steps Taken

- ICOs whose teams issued more than 10^{10} coins were filtered out. This reduces the effect of outliers without resulting in a substantial loss of data.

teamSize

Insights

- The average team consisted of 13 individuals with half of the teams being formed of less than 12 people.
- The smallest team was comprised of 1 individual while 75 individuals formed the largest one.
- Mild outliers are seen in the scatter plot.
- There are 146 teams with an undefined number of individuals.

Steps Taken

- Teams consisting of more than 50 individuals were filtered out.

distributedPercentage

Insights

- Half of the teams distributed less than 55% of the blockchain coins issued upon launching their ICO.
- Assuming all the percentage values lie between 0 and 1, the average value of 1.102 was considered erroneous.

Steps Taken

- All values greater than one were filtered out.

Duration

Insights

- Half of the fundraising teams aimed to achieve their funding target in less than 45 days with the average value being around 69 days.
- Anomalous negative values and extreme outliers are seen in the scatter plot and justified by the gap between the median and mean.

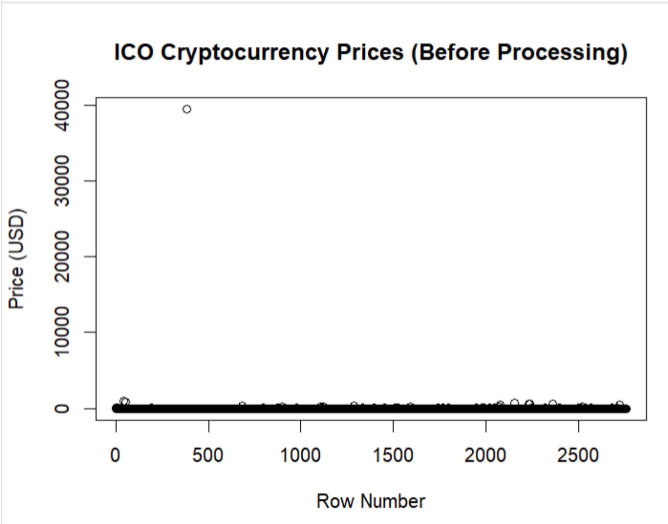
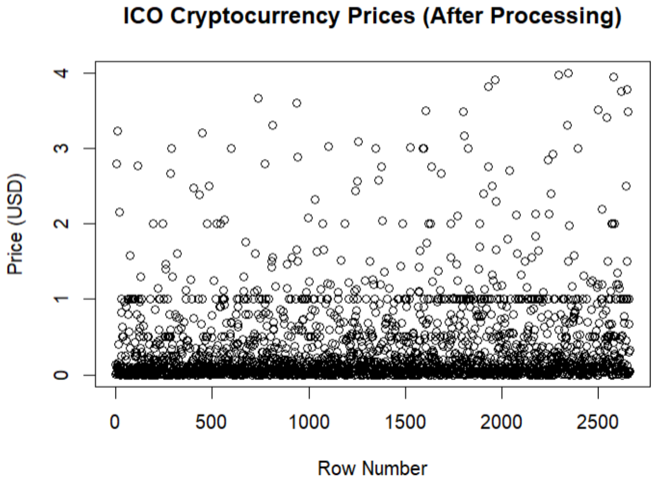
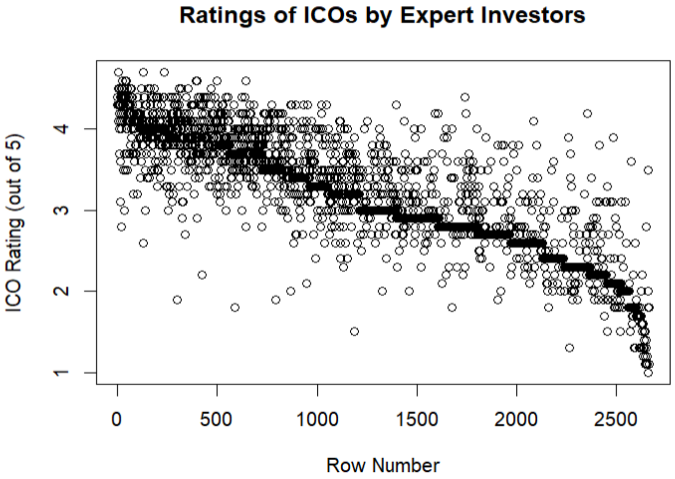
Steps Taken

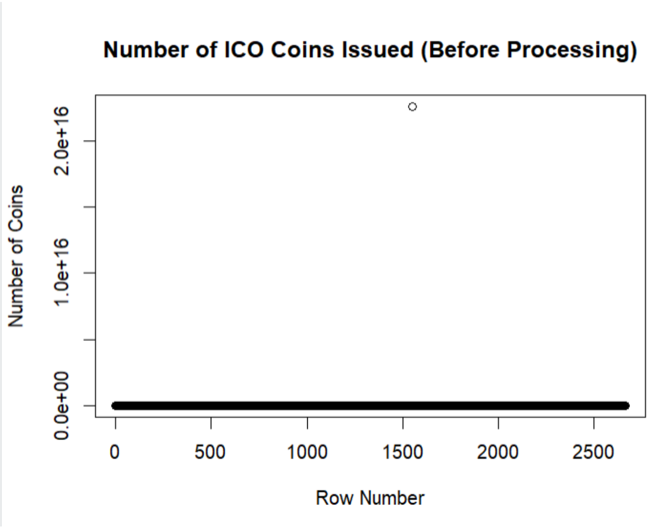
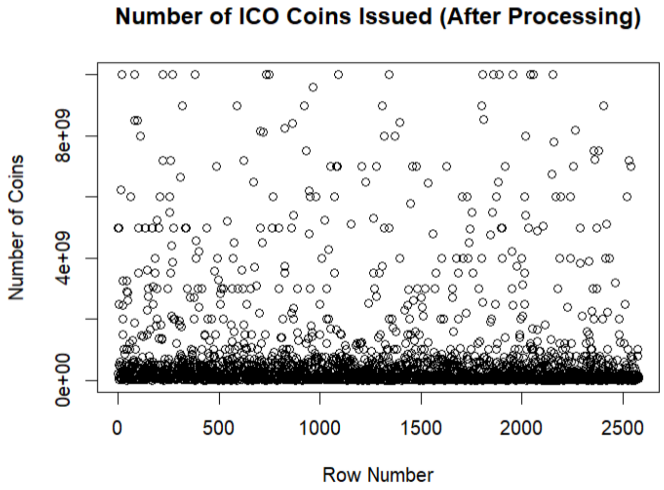
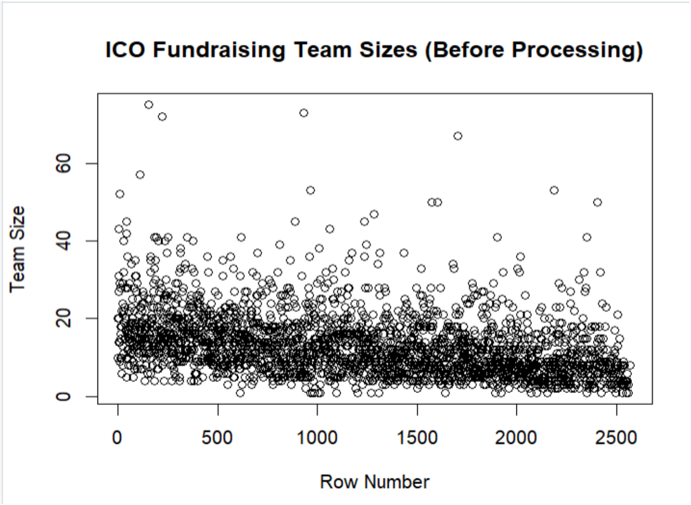
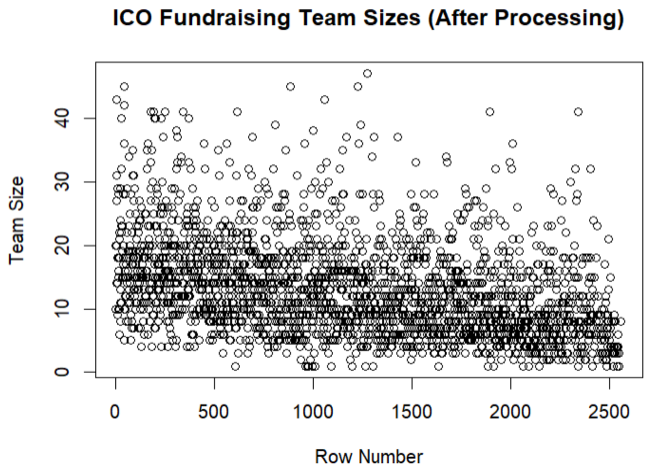
- ICOs with a duration above 500 days were filtered out.

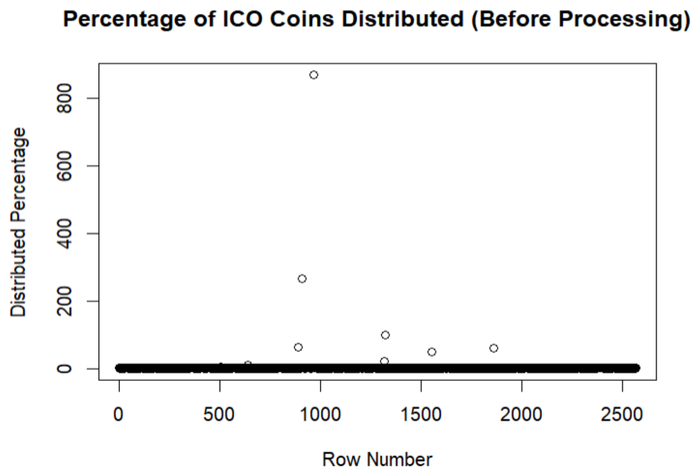
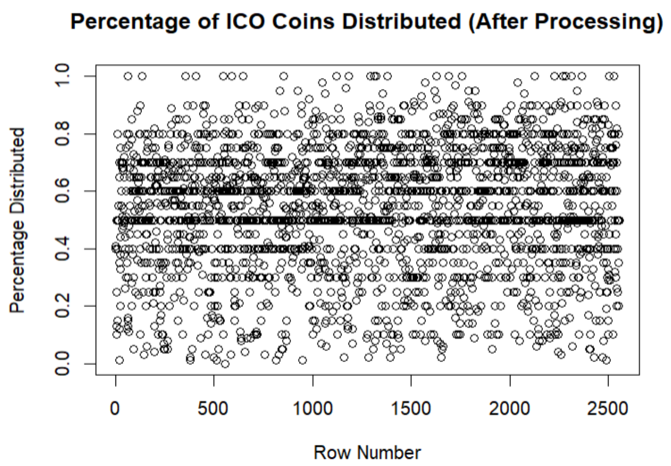
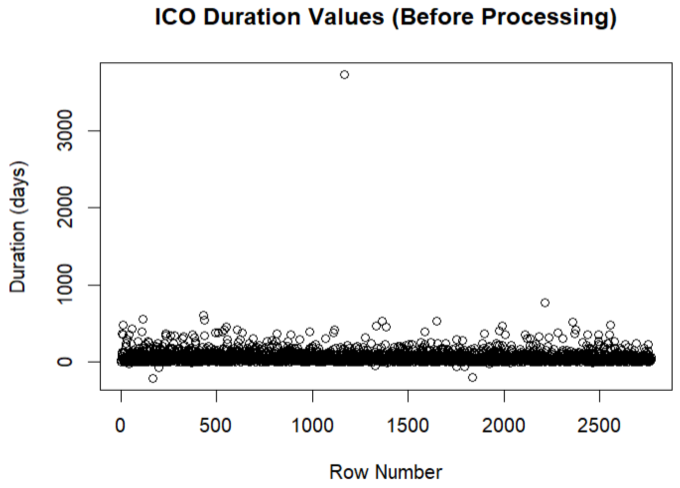
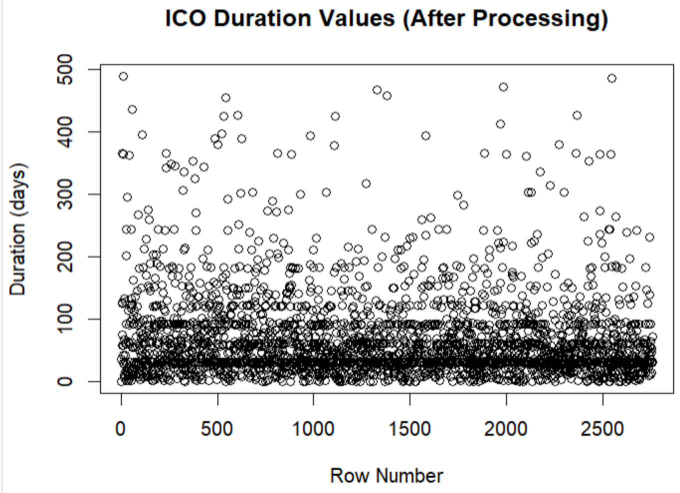
The statistical summaries of the numeric features after processing are as follows:

After Processing							
Feature	**Min**	**Q1**	**Median**	**Mean**	**Q3**	**Max**	**NA's**
priceUSD	0	0.04	0.1	0.35	0.46	3.99	180
Rating	1	2.7	3.1	3.131	3.7	4.7	0
coinNum	1.2	5x10^{7}	1.8x10^{8}	7.3x10^{8}	5.1x10^{8}	10^{10}	0
teamSize	1	7	12	12.98	17	47	146
distributedPercentage	0.00	0.40	0.55	0.54	0.70	1.00	0
Duration	0	29	45	67.1	90	488	0

Below are the scatter plots of numeric features before and after processing.

Feature Name	Before Processing	After Processing
priceUSD	 	
Rating	 	No processing was performed.

coinNum	 Number of ICO Coins Issued (Before Processing)	 Number of ICO Coins Issued (After Processing)
teamSize	 ICO Fundraising Team Sizes (Before Processing)	 ICO Fundraising Team Sizes (After Processing)

distributedPercentage		
Duration		

After processing, the number of rows in the dataset was reduced to 2,556 (7.62% of the original dataset).

Examining the Brand Slogan Column

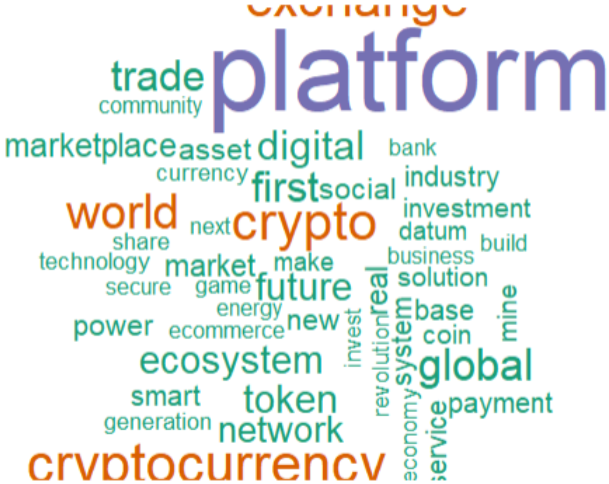
The following pre-processing steps were taken:

- Creation of the corpus.
- Replacement of the non-visible characters with a space.
- Conversion of the words to lower case.
- Removal of all characters that are not letters or a space, stopwords, and extra white spaces.
- Lemmatization.
- Conversion of the corpus to a plain text document and then to a Document Term Matrix.

After processing, the words in the documents were sorted by decreasing order of frequency. Below are the top 6 most repeated words:

Word	Frequency
platform	371
blockchain	363
decentralized	251
cryptocurrency	158
crypto	151
exchange	141

Then, the word cloud below was created:



Evidently, crypto-related terms such as “platform”, “crypto”, and “cryptocurrency” were some of the most used words in slogans.

Adding the Slogan_Length Feature

The length of each campaign’s slogan was computed and added as another feature.

Removing Unwanted Features

The startDate, endDate, brandSlogan, countryRegion and platform were removed from the dataset as they were replaced by the added features. The day, month, and year values replace the date features while the Slogan_Length replaces the brandSlogan feature and the is_ethereum feature replaces the platform one. Also, the is_most_friendly feature replaces the countryRegion one.

Choosing the Performance Metrics

The study by Krishna et al. (2016) evaluated various models in predicting the outcome of startups by focusing on precision, recall and AUC. According to the study, precision and recall provide insight into the model’s ability to collect relevant results while also minimizing the number of irrelevant results. Besides, Krishna et al. (2016) justify their choice of prioritizing the area under ROC curve over the accuracy measure by stating that the latter can be misleading when classes are imbalanced.

In the context of predicting the success of an ICO, it is a priority to minimize the number of failure ICOs that are predicted as successes. A satisfactory model would have a low false positive rate, which is equivalent to a high precision. The latter would alleviate the risks that investors face when deciding on investing in an ICO.

At the same time, it is important to capture as many of the successful ICOs as possible to attract new investors. Maximizing the number of relevant results would require maximizing recall/sensitivity; equivalent to minimizing the false negative rate.

Lastly, AUC values, which will be primarily relied on for comparing models, will be prioritized over accuracy. Other cross-validation performance metrics as well as the fold ROC curves, and confusion matrices are presented in the Appendices.

Decision Trees

This section outlines the additional processing steps taken before applying cross-validation on different decision tree models using the ideal combination of features and optimized model parameters.

Data Pre-Processing

Missing Values

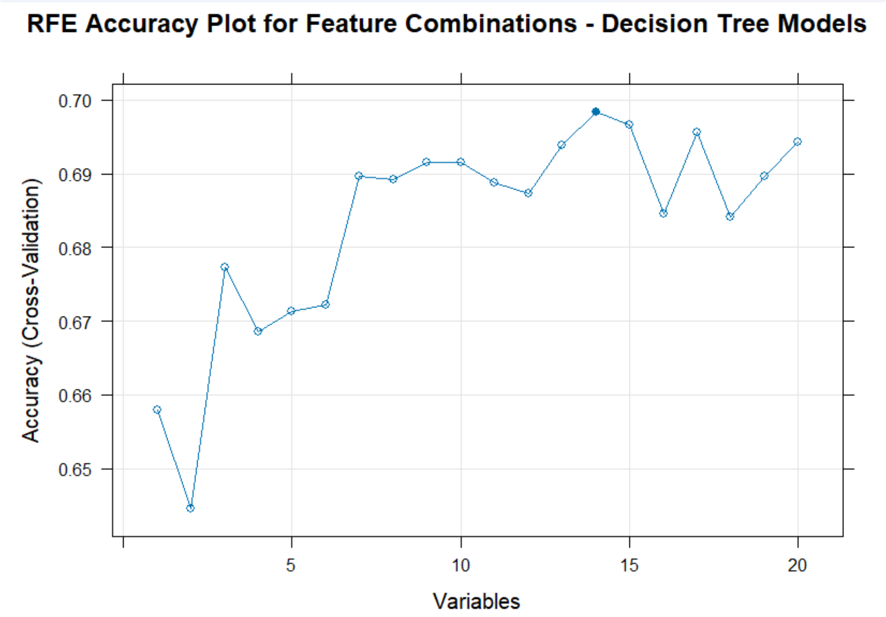
The missing values in the priceUSD and teamSize columns were imputed separately when training each of the DT algorithms.

Feature Selection

Recursive Feature Elimination (RFE) was applied on the data to test different combinations of features using CV and select the optimal one. Since the RFE algorithm does not accept missing values, those in the dataset were imputed using multiple imputation from the MICE package (to apply RFE).

The following 14 features were selected and retained for the application of CV on DT models:

- Rating
- teamSize
- endYear
- startYear
- hasVideo
- duration
- endMonth
- startMonth
- hasGithub
- priceUSD
- is_most_friendly
- is_USA
- startDay
- hasReddit



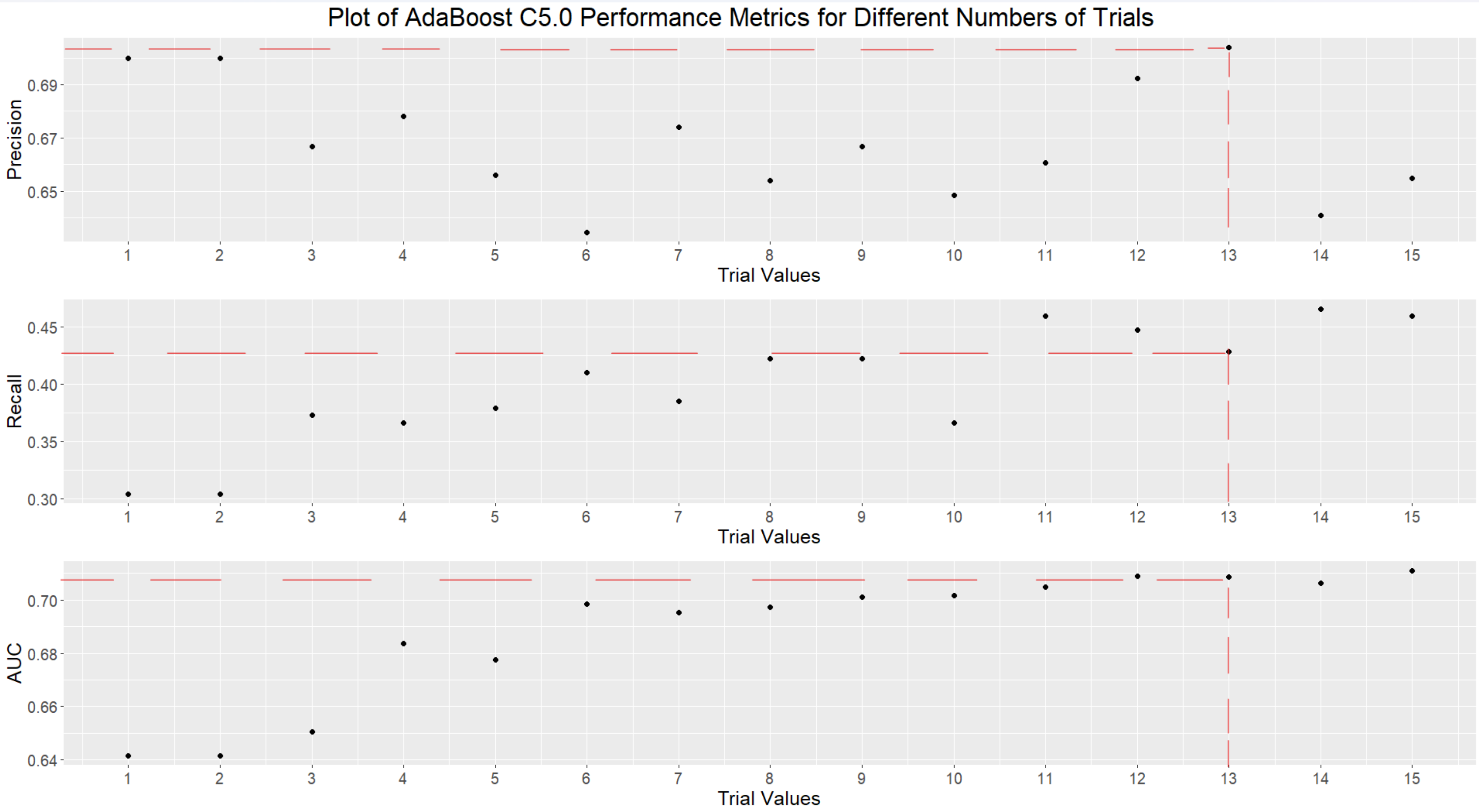
Modelling – AdaBoost C5.0

Missing Values

Missing values were imputed using the C5.0 built-in functionality.

Optimizing the Number of Trials

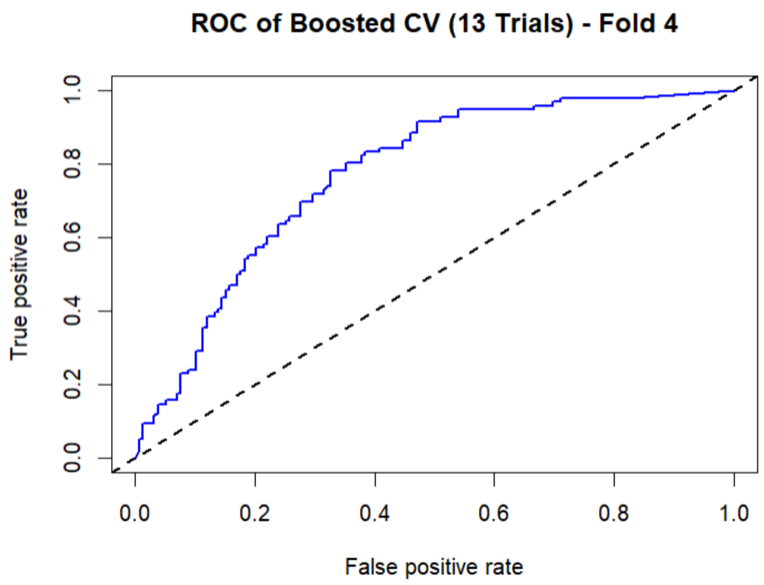
The performance metrics of C5.0 models with different numbers of trials are shown below. 10-fold CV was applied on a C5.0 model with 13 trials as it maximizes precision (70.4%) while attaining satisfactory recall (42.8%) and AUC (0.708) values.



Applying CV on AdaBoost C5.0

On average, the C5.0 model with 13 trials captured 44.1% of the successful ICOs with 60.5% of those predicted to succeed being successful. The standard deviation of the recall metric is considered slightly high at 7.56% thus reflecting an inconsistent false negative rate. However, the low standard deviation value of other metrics indicates a consistent model performance otherwise.

Besides, the 4th fold achieved the best performance by capturing 46.8% of the successful ICOs in the dataset with 64.2% of those predicted to succeed being successful.



AdaBoost C5.0 DT			
	Precision	**Recall**	**AUC**
Average	60.5%	44.1%	0.716
SD	4.54%	7.56%	3%

Fold 4		
Precision	**Recall**	**AUC**
64.2%	46.8%	0.771

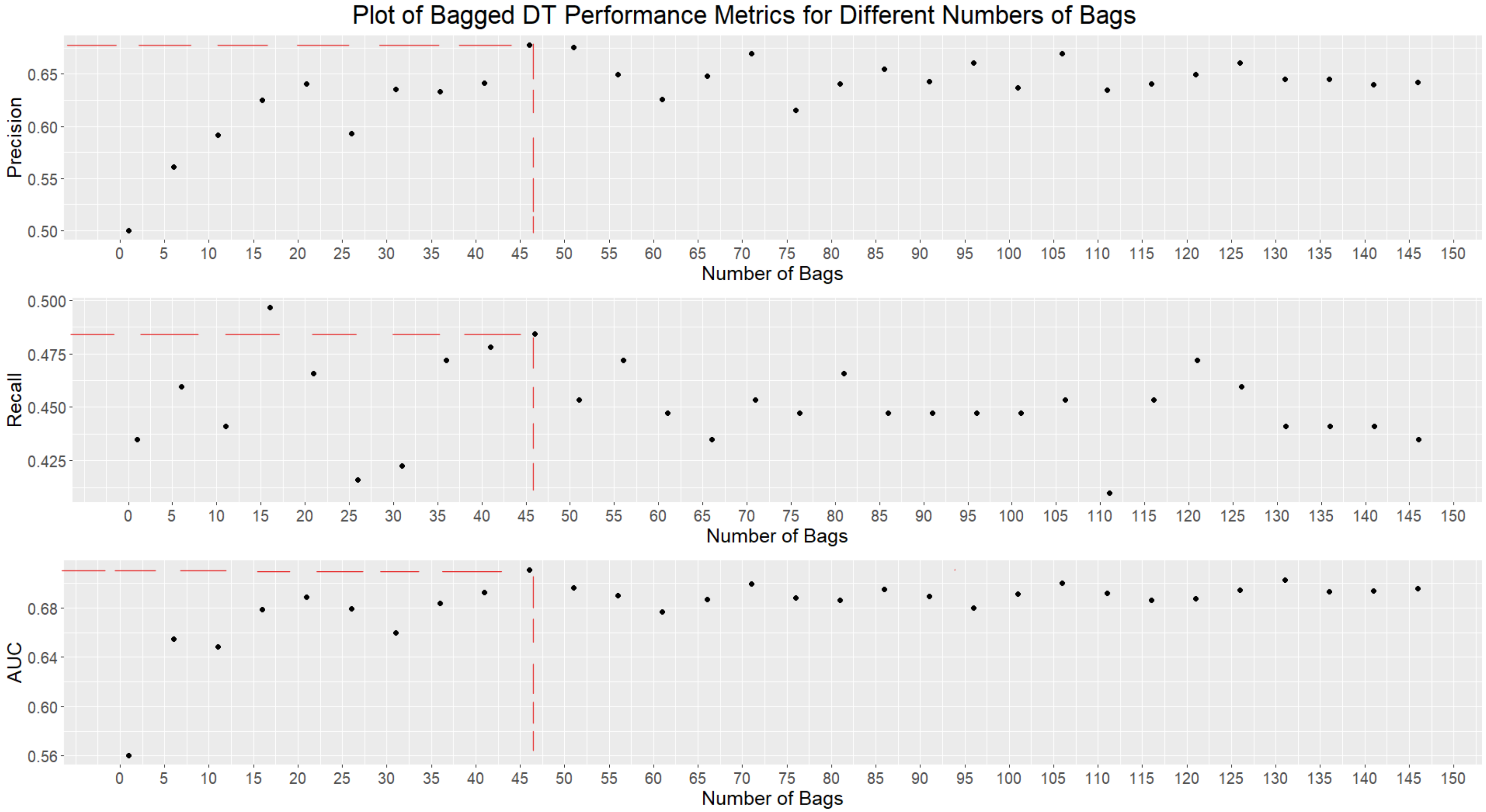
Modelling – Bagged DT

Missing Values

Since the Bootstrap Aggregation approach is not affected by missing data, the missing values in the priceUSD and teamSize features were retained.

Optimizing the Number of Bags

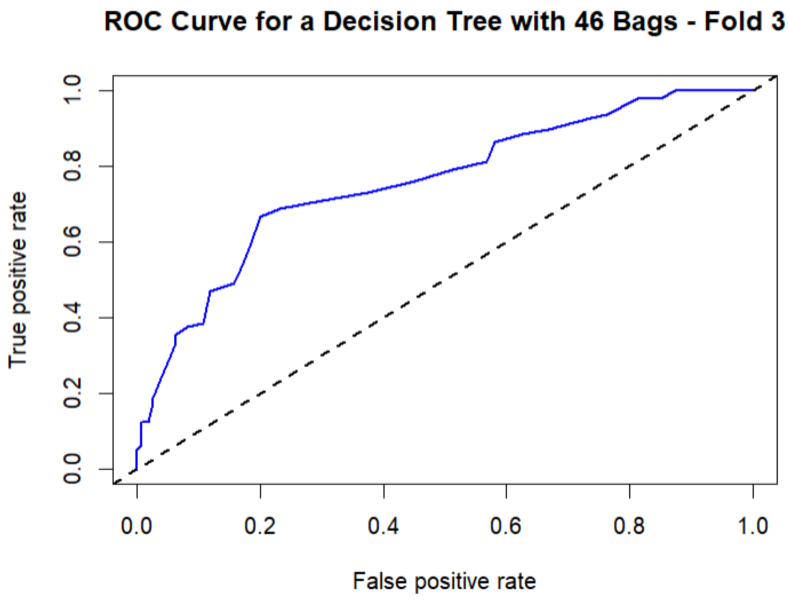
The performance metrics of DTs with different numbers of bags are shown below. 10-fold CV was applied on a DT with 46 bags as it maximizes precision (68%) while attaining satisfactory recall (48%) and AUC (0.71) values.



Applying CV on a DT with 46 Bags

On average, the Bagged DT captured 44.9% of the successful ICOs with 58.1% of those predicted to succeed being successful. Also, the low standard deviation values between 3% and 5% validate the model's consistent performance across the 10 folds.

Besides, the 3rd fold achieved the best performance by capturing 47.9% of the successful ICOs with 65.7% of those predicted to succeed being successful.



Bagged DT			
▨	**Precision**	**Recall**	**AUC**
Average	58.1%	44.9%	0.692
SD	4.81%	4.39%	3.26%

Fold 3		
Precision	**Recall**	**AUC**
65.7%	47.9%	0.758

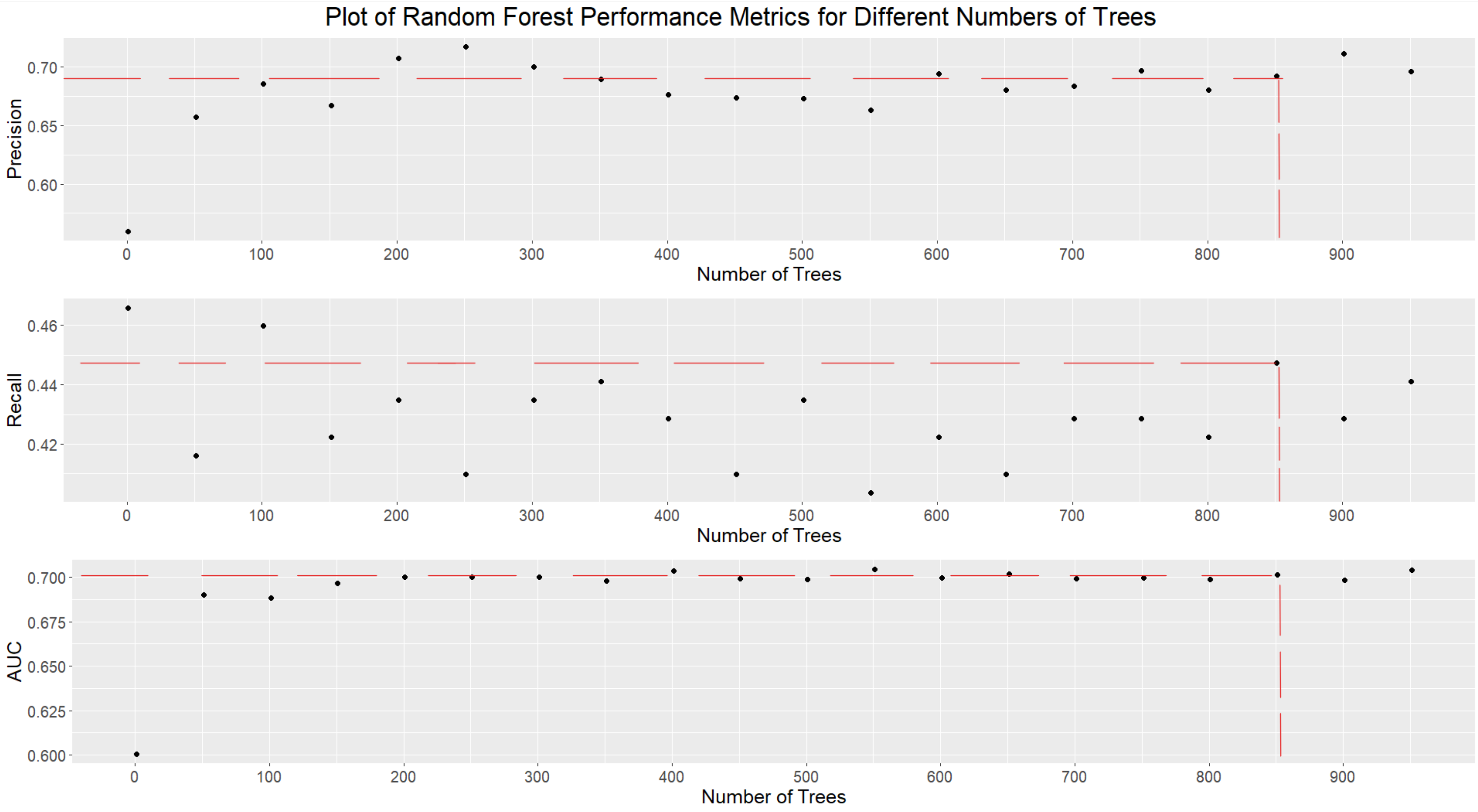
Modelling – Random Forest

Missing Values

Multiple imputation was performed on the priceUSD and teamSize features using the MICE package.

Optimizing the Number of Trees

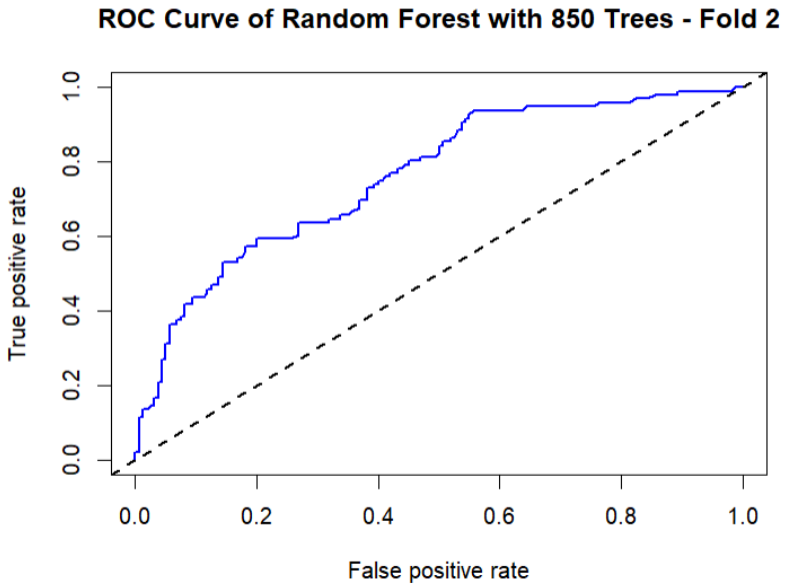
The performance metrics for RF models with different numbers of trees are shown below. 10-fold CV was applied on a RF with 850 trees as it balances precision (69.2%) and recall (44.7%) while achieving a satisfactory AUC (0.701) value.



Apply CV to a Random Forest with 850 Trees

On average, the RF with 850 Trees captured 43.4% of the successful ICOs with 62.3% of those predicted to succeed being successful. Besides, the low standard deviation values between 3% and 5% reflect the model's consistent performance across the 10 folds.

The 3rd fold achieved the best performance by capturing 43.7% of the successful ICOs with 70% of those predicted to succeed being successful.



Random Forest			
	Precision	**Recall**	**AUC**
Average	62.3%	43.4%	0.71
SD	5%	4.27%	3.61%

Fold 2		
Precision	**Recall**	**AUC**
70%	43.7%	0.761

Nearest Neighbours

This section outlines the additional processing steps taken before applying 10-fold CV on a kNN model using the ideal combination of features and an optimized number of neighbours.

Data Pre-Processing

Normalizing the Numeric Features

The numeric features in the dataset were normalized using the z-score method.

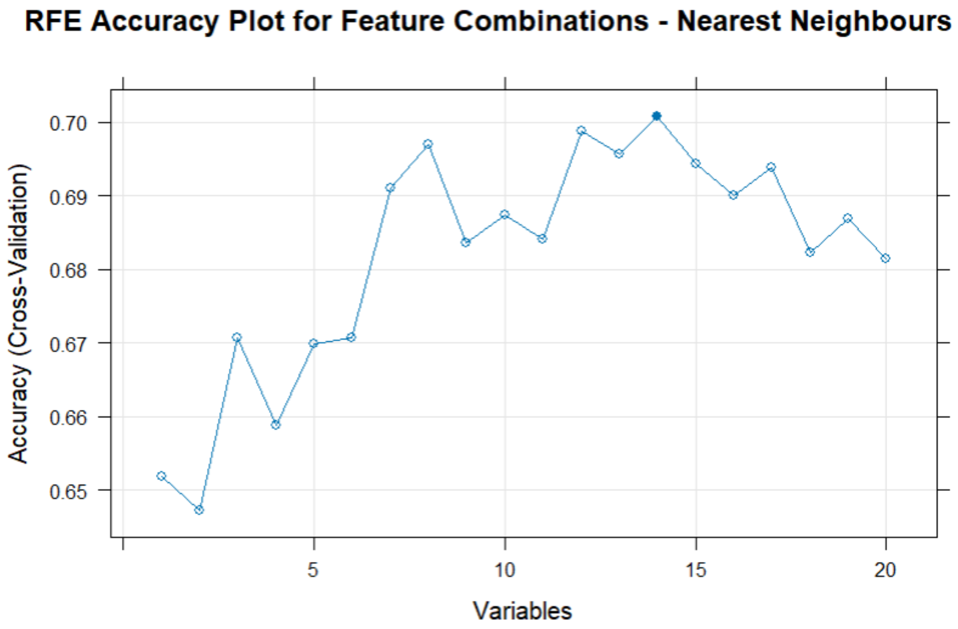
Missing Values

The missing values in the priceUSD and teamSize columns were imputed using the kNN imputation function from the VIM package.

Feature Selection

The following 14 features were selected and retained for the application of CV:

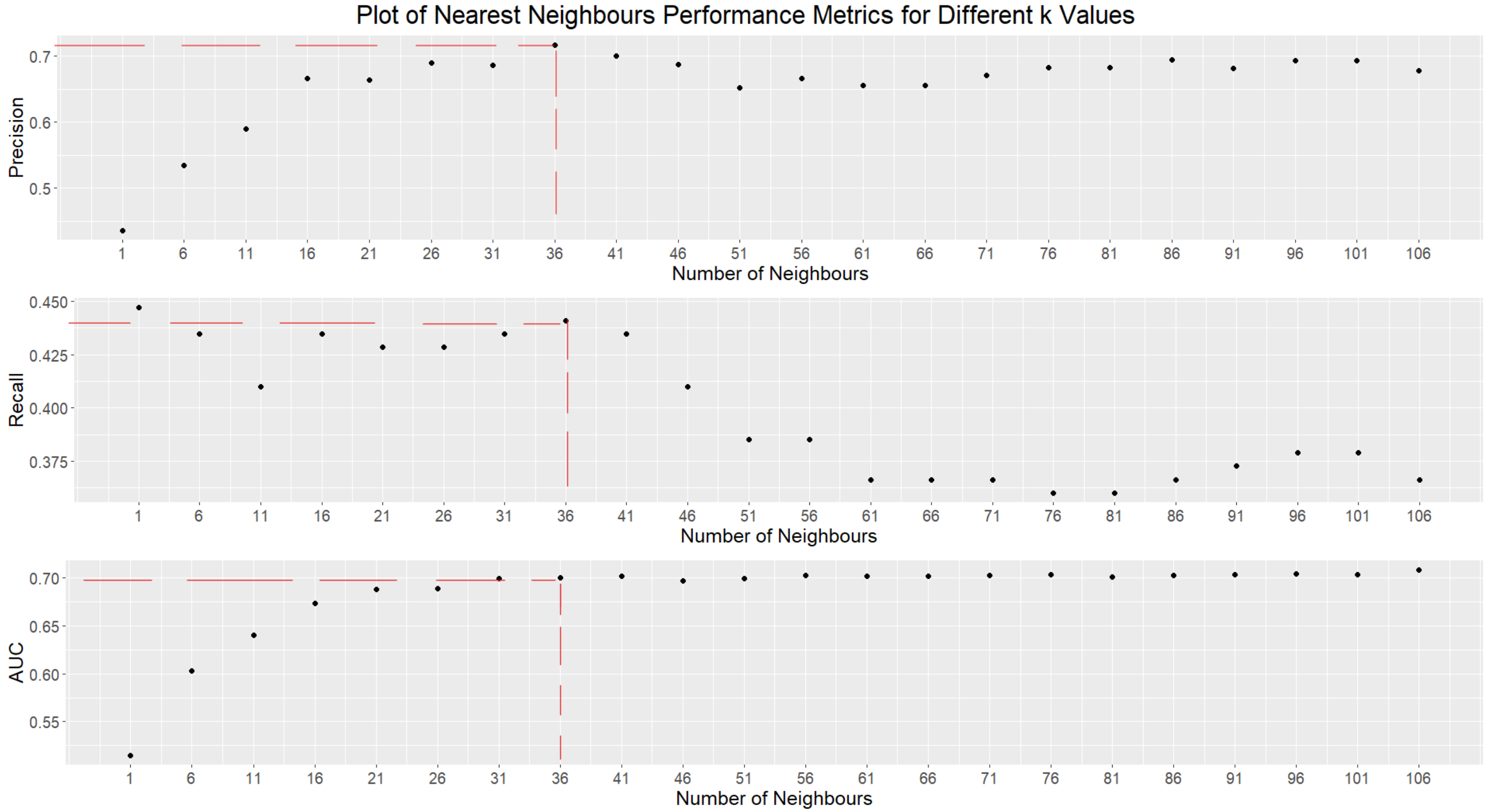
- Rating
- teamSize
- startYear
- endYear
- hasVideo
- duration
- endMonth
- hasReddit
- startMonth
- hasGithub
- priceUSD
- is_most_friendly
- startDay
- hasReddit



Modelling – Nearest Neighbours

Optimize the Number of Neighbours

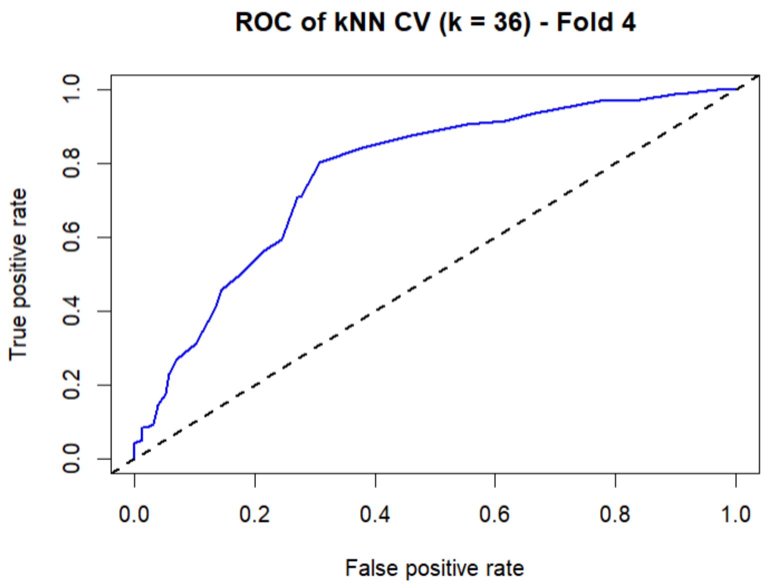
The performance metrics for kNN models with different k values are shown below. 10-fold CV was applied on a kNN model with 36 neighbours as it maximizes precision (71%) while attaining satisfactory recall (45%) and AUC (0.7) values.



Apply CV with the Chosen Number of Neighbours (k = 36)

On average, a kNN model with 36 neighbours captured 42.8% of the successful ICOs with 64.1% of those predicted to succeed being successful. The low standard deviation values between 2% and 5% validate the model's consistent performance across the 10 folds.

Besides, the 4th fold achieved the best performance by capturing 47.9% of the successful ICOs with 63.8% of those predicted to succeed being successful.



Nearest Neighbours			
	Precision	**Recall**	**AUC**
Average	64.1%	42.8%	0.717
SD	4.92%	2.73%	4.19%

Fold 4		
Precision	**Recall**	**AUC**
63.8%	47.9%	0.773

Support Vector Machine

This section outlines the additional processing steps taken before applying 10-fold CV on an SVM model using the ideal combination of features and an optimized cost parameter.

Data Pre-Processing

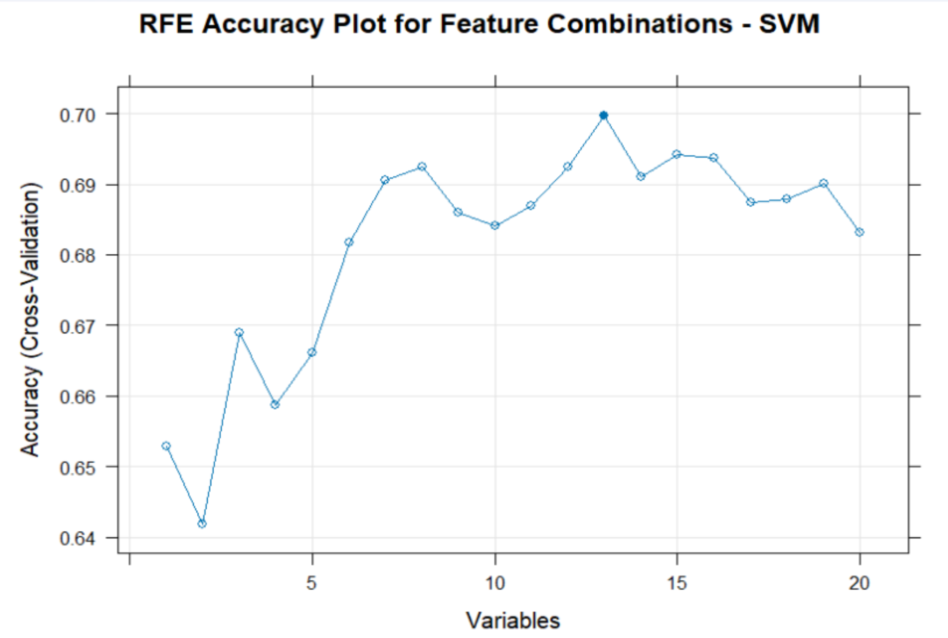
Missing Values

Missing values from the priceUSD and teamSize features were imputed using the missForest package.

Feature Selection

The following 13 features were selected and retained for the application of CV:

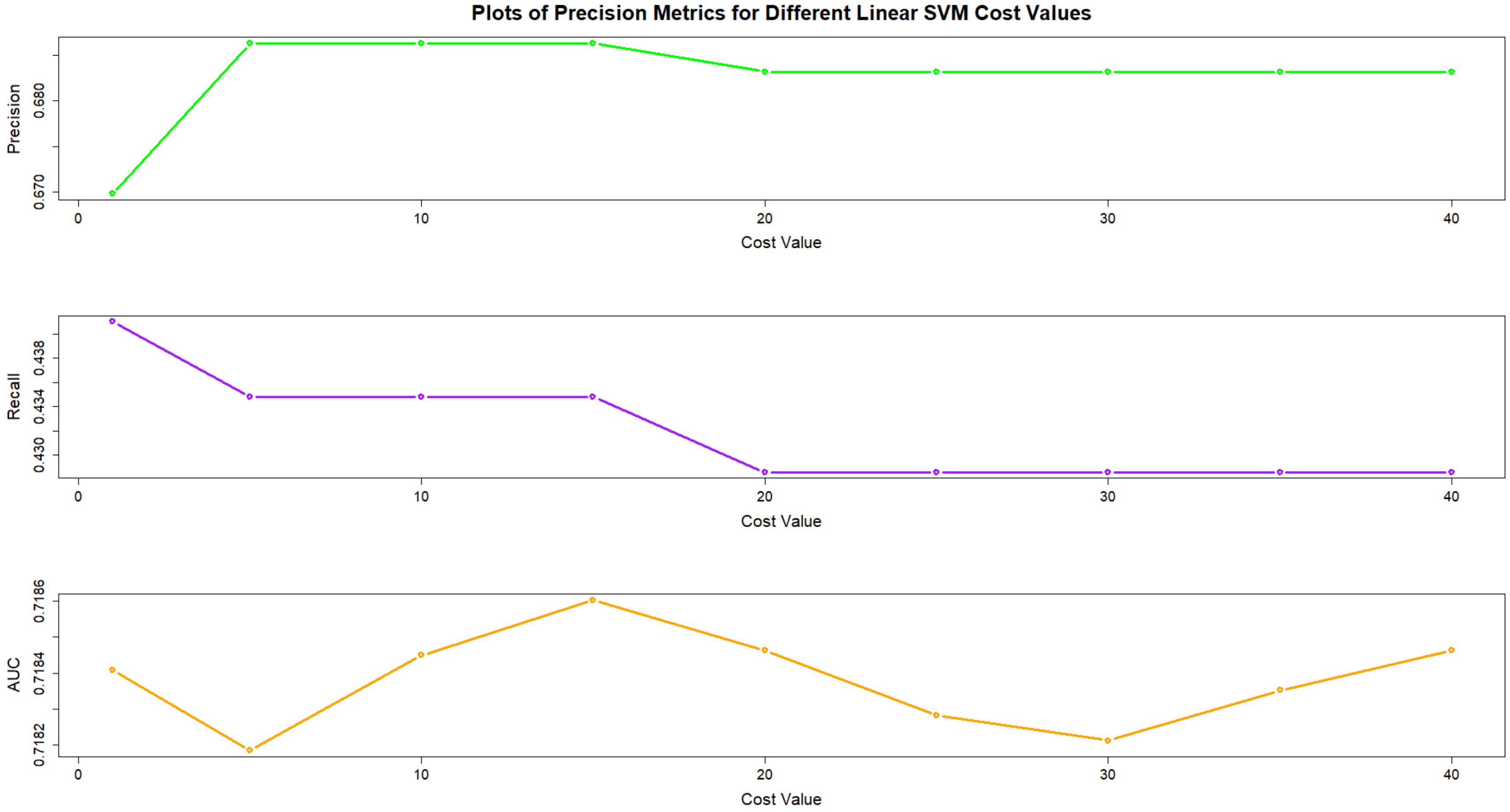
- Rating
- teamSize
- startYear
- endYear
- duration
- hasVideo
- hasReddit
- endMonth
- startMonth
- hasGithub
- priceUSD
- is_most_friendly
- startDay



Modelling – Support Vector Machine

Optimizing the Cost Parameter for Linear SVM

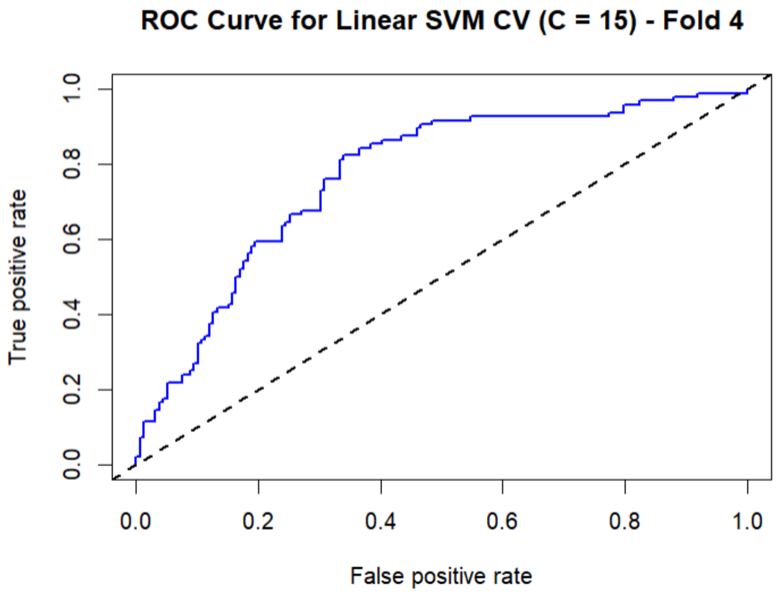
As seen in the plots below, the difference in performance between different cost values is negligible. However, a cost value of 15 would maximize precision and AUC while attaining a satisfactory recall.



Apply CV on Linear SVM (C = 15)

On average, the Linear SVM with C=15 captured 44.6% of the successful ICOs with 63.9% of those predicted to succeed being successful. Also, the low standard deviation values ranging between 3% and 6% reflect the model's consistent performance across the 10 folds.

The 10th fold, which achieved the best performance, captured 52% of the successful ICOs in the dataset with 64.1% of those predicted to succeed being successful.



Linear SVM			
	Precision	**Recall**	**AUC**
Average	63.9%	44.6%	0.728
SD	6.15%	3.55%	3.54%

Fold 4		
Precision	**Recall**	**AUC**
64.1%	52%	0.768

Artificial Neural Networks

This section outlines the additional processing steps taken before applying 10-fold CV on an ANN model using the ideal combination of features and optimized size & decay parameters.

Data Pre-Processing

Normalizing the Numeric Features

The numeric features in the dataset were normalized using the z-score method.

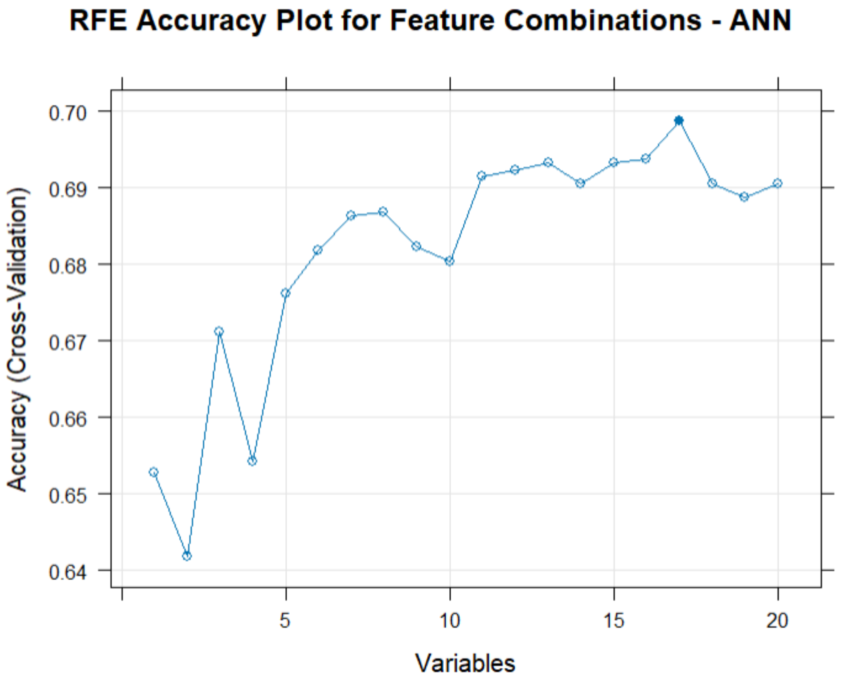
Missing Values

The missing values in the priceUSD and teamSize features were imputed using the missForest package.

Feature Selection

The following 17 features were selected and retained for the application of CV:

- rating
- teamSize
- endYear
- startYear
- duration
- hasReddit
- endMonth
- hasVideo
- startMonth
- hasGithub
- priceUSD
- is_most_friendly
- startDay
- is_USA
- distributedPercentage
- endDay
- coinNum



Modelling – Artificial Neural Networks

Tuning ANN Parameters

By tuning ANN parameters through repeated CV using the caret package, a neural network with 1 node in the hidden layer and a decay of 0.0001 was found to maximize CV accuracy. Accordingly, these parameter settings were used for training an ANN model.

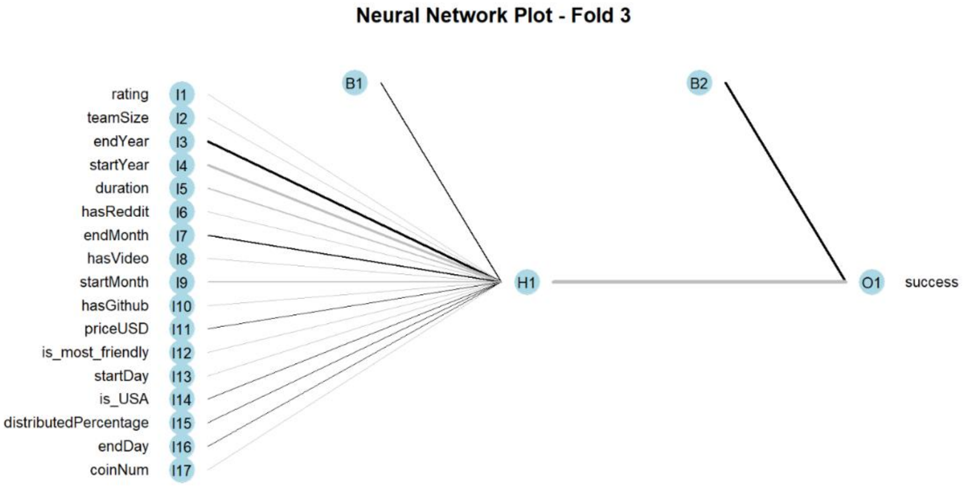
Applying CV on ANN

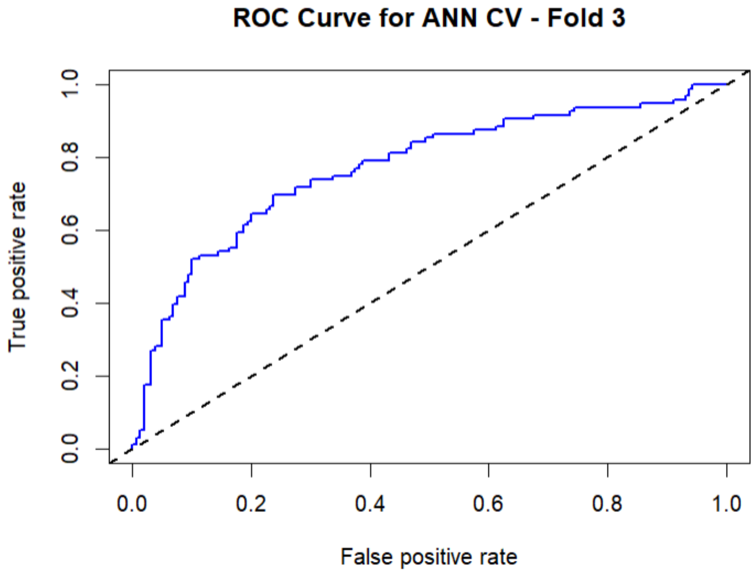
On average, the ANN with the chosen parameters captured 45.1% of the successful ICOs with 65.1% of those predicted to succeed being successful. Also, the low standard deviation values ranging between 2% and 6% validate the model's consistent performance across the 10 folds.

Besides, the ANN tested on the 3rd fold achieved the best performance by capturing 48.9% of the successful ICOs with 74.6% of those predicted to succeed being successful.

ANN			
	Precision	**Recall**	**AUC**
Average	65.1%	45.1%	0.729
SD	5.75%	2.8%	3.52%

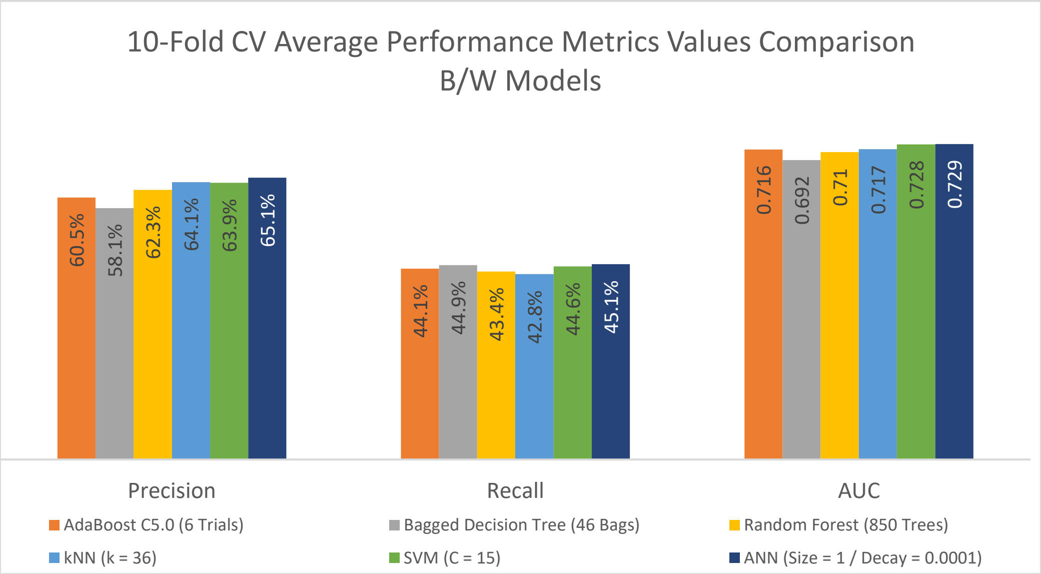
Fold 3		
Precision	**Recall**	**AUC**
74.6%	48.9%	0.771

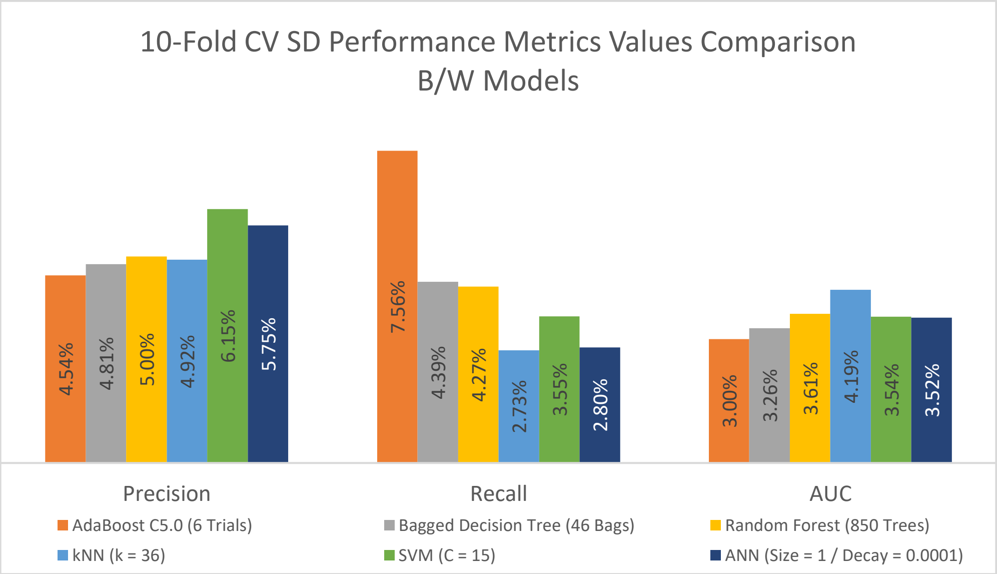




Model Performance Metrics Comparison & Recommendation

The column charts below compare the 10-fold CV average performance metrics values and standard deviations achieved by each of the tested models.





ANN achieved the highest precision, recall, and AUC values. Therefore, it is recommended to apply the ANN model if optimal and all-round performance is key. The ANN model minimizes the number of failing ICOs that are predicted to succeed (by maximizing precision) and minimizes the number of successful ICOs that are predicted to fail (by maximizing recall).

However, Black Box models (ANN and SVM) have their disadvantages when used for high-stakes decisions. According to Cynthia (2022) and Kenton (2020), the difficulty of interpreting and troubleshooting the predictions of such models poses questions regarding ethics, accountability, transparency, and governance. Black Box models, which are also computationally costly, are not risky by nature but can be destructive when the risks associated with an investment decision only become apparent after the losses are incurred. Therefore, when ethics and governance is a priority or computational resources are limited, it is recommended to apply kNN or RF instead. While kNN under performs compared to RF in terms of recall, it still more consistent (with a lower recall SD) and attains a higher precision. Besides, the AUC values of the two models are almost equal. Given that the difference in precision between the two models is greater than that in recall, the kNN model is considered superior to RF.

As for AdaBoost C5.0 and Bagged DT, both models did not manage the precision-recall trade-off well by achieving low precision and high recall values thus are inferior to other models.