Master
Project Management
and Data Science

SECOM ANALYTICS

TEAM 3

Gupta, Himansha
Pomay Polat, Ekin
Dsouza, Rashmi Carol
Pham, Quynh Dinh Hai

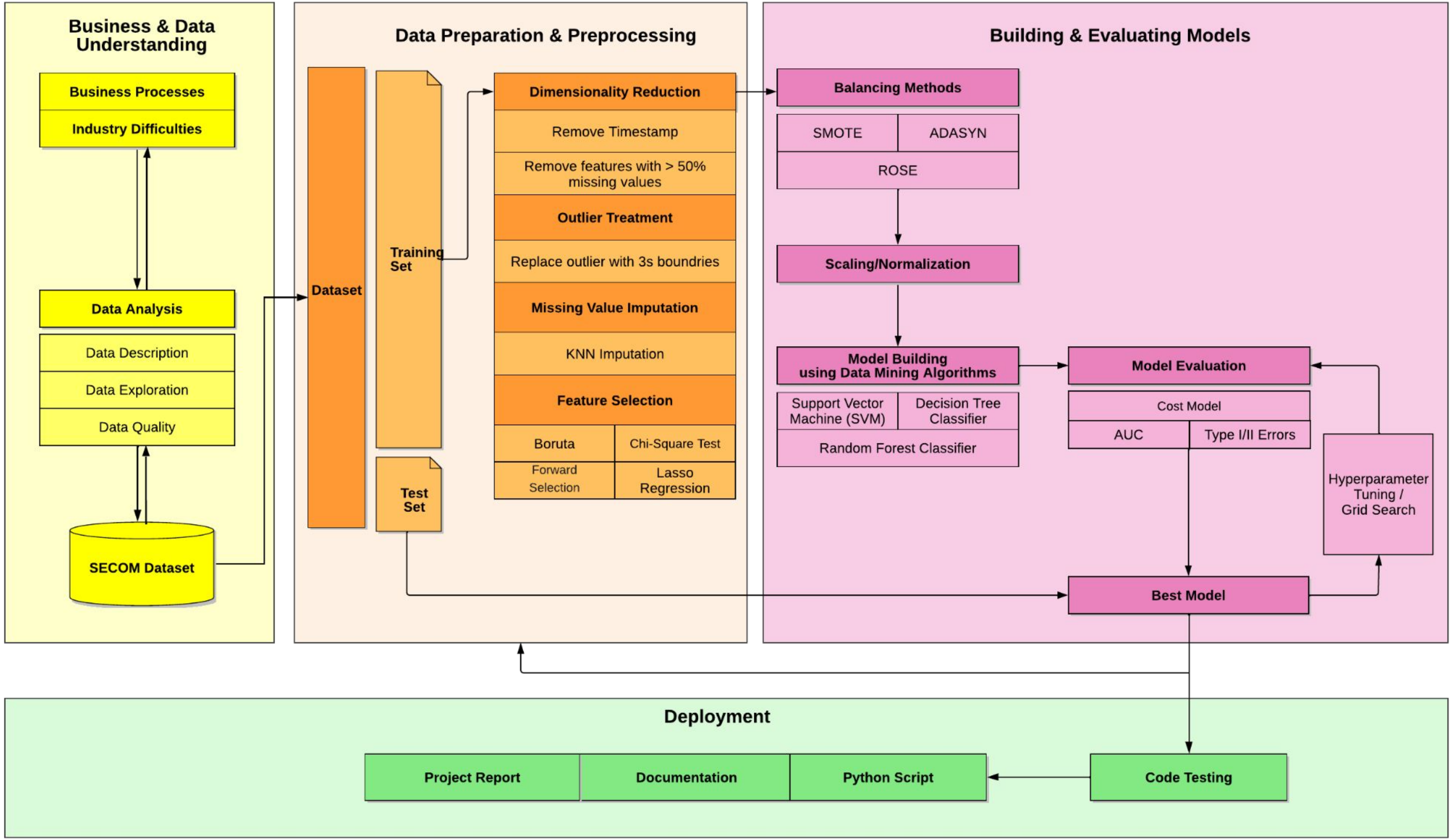
htw.

Master
Project Management
and Data Science

TABLE OF CONTENTS

Master
Project Management
and Data Science

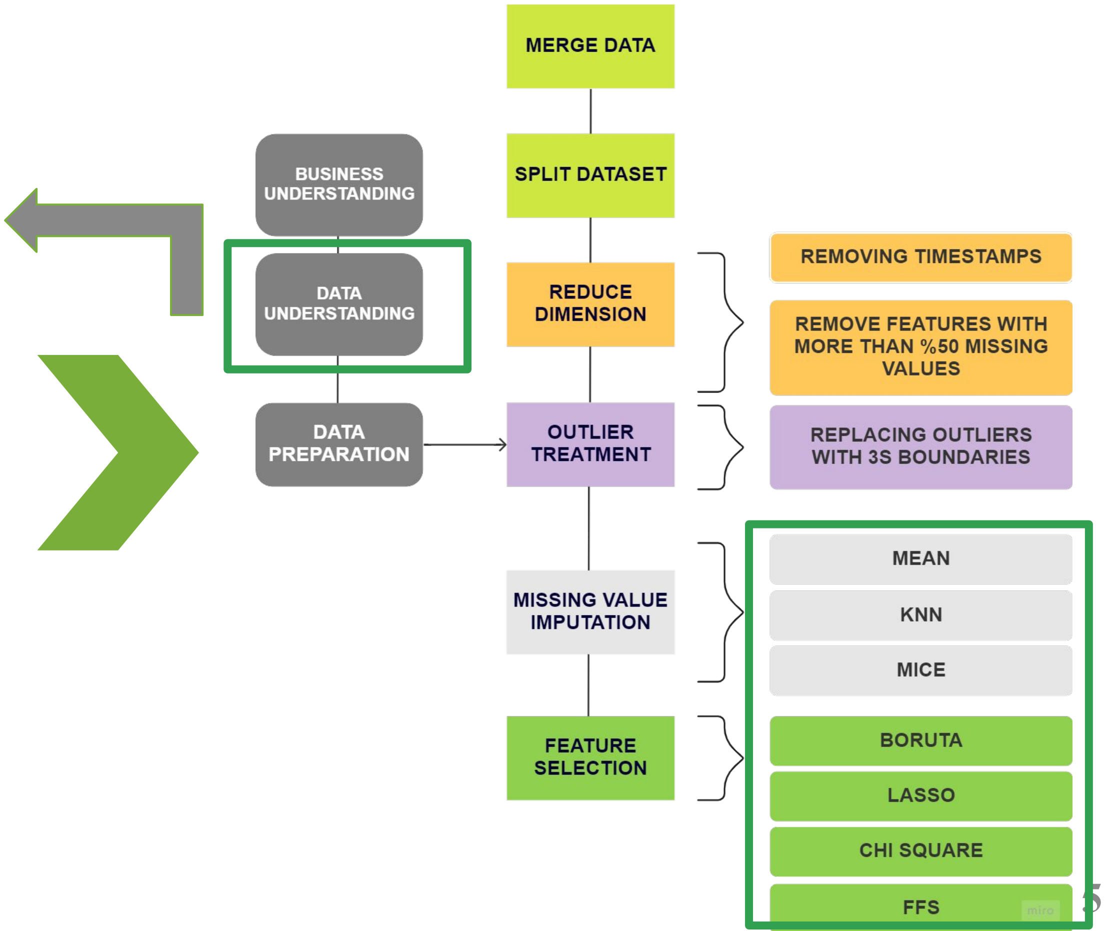
WORKFLOW



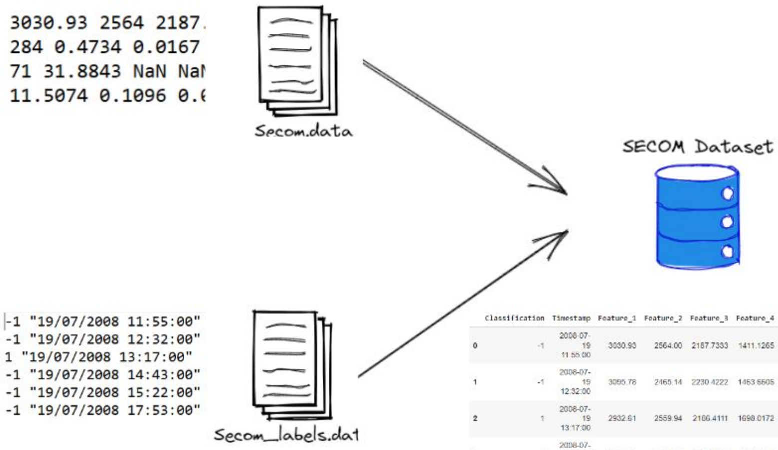
DATA UNDERSTANDING & PREPROCESSING

DATA PREPROCESSING

Problems with the dataset
Imbalance data between pass & fail test results
Missing values
Timestamp does not yield meaningful insight to build model
Many highly correlated features
Outliers

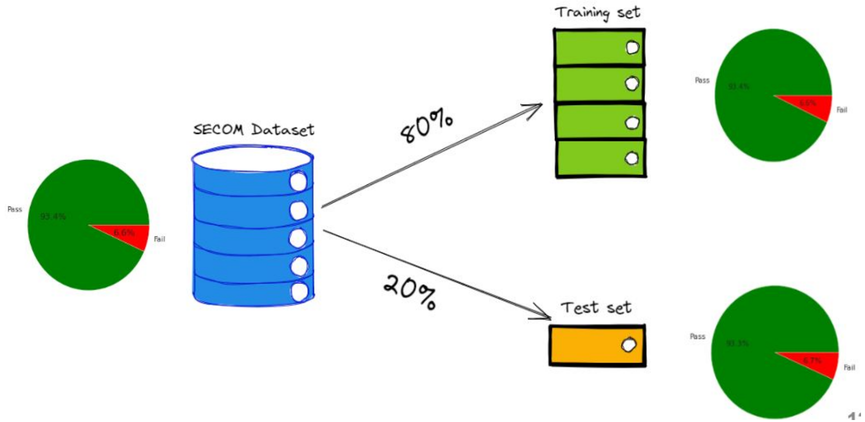


DATA PREPARATION



	Classification	Timestamp	Feature_1	Feature_2	Feature_3	Feature_4	Feature_5	Feature_6	Feature_7	Feature_8	...
0	-1	2008-07-19 11:55:00	3030.93	2564.00	2187.7333	1411.1265	1.3602	100.0	97.6133	0.1242	...
1	-1	2008-07-19 12:32:00	3095.78	2465.14	2230.4222	1463.6606	0.8294	100.0	102.3433	0.1247	...
2	1	2008-07-19 13:17:00	2932.61	2559.94	2186.4111	1698.0172	1.5102	100.0	95.4878	0.1241	...
3	-1	2008-07-19 14:43:00	2988.72	2479.90	2199.0333	909.7926	1.3204	100.0	104.2367	0.1217	...

MERGING DATA



SEPARATE TRAINING & TEST DATA

DATA PREPARATION

REDUCING DIMENSIONALITY

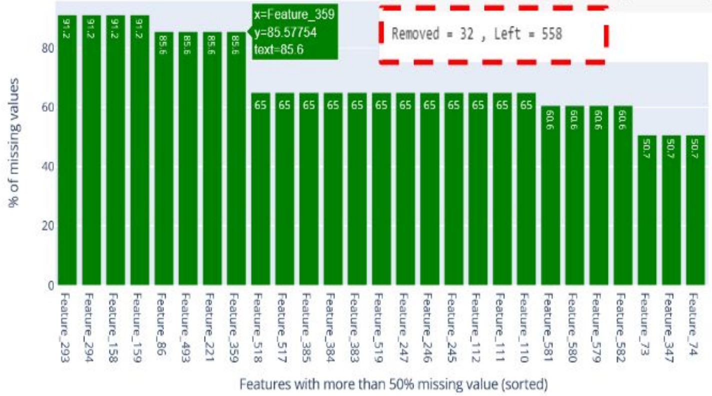
1. Remove the Timestamp feature
2. Remove features with missing values

Threshold:
Features with more than 50% missing value

```
def percent(dataframe, threshold):
    columns = dataframe.columns[(dataframe.isna().sum()/dataframe.shape[1])>threshold]
    return columns.tolist()

na_columns = percent(X_train, 0.5)
X_train_na = X_train.drop(na_columns, axis=1)
X_test_na = X_test.drop(na_columns, axis=1)
n_features1 = X_train_na.shape[1]
print(f'Removed = {len(na_columns)} , Left = {n_features1}')
```

```
Removed = 32 , Left = 558
```



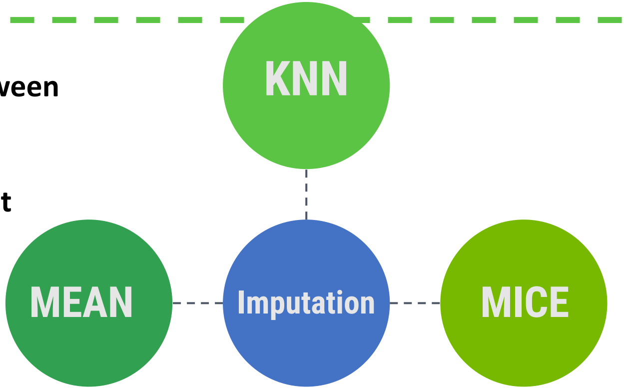
OUTLIER TREATMENT

Replace outliers with 3s boundaries

MISSING VALUE IMPUTATION

Which Imputation Method did we choose?

- Replacing missing values with actual occurring values or very similar values while calculating the average.
- Better imputer for highly dimensional dataset
- non-parametric approach, less prone to misspecifications.

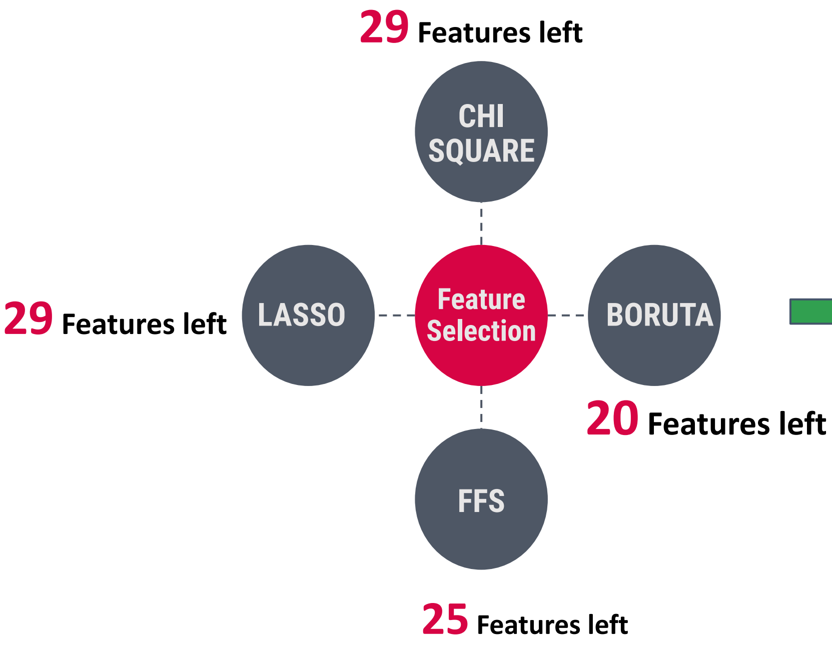


- Does not preserve relationships between variables such as correlations.
- Takes only single feature into account
- Reduces the variance of the imputed variables.

- Time Consuming
- Performance depends on the size of the dataset

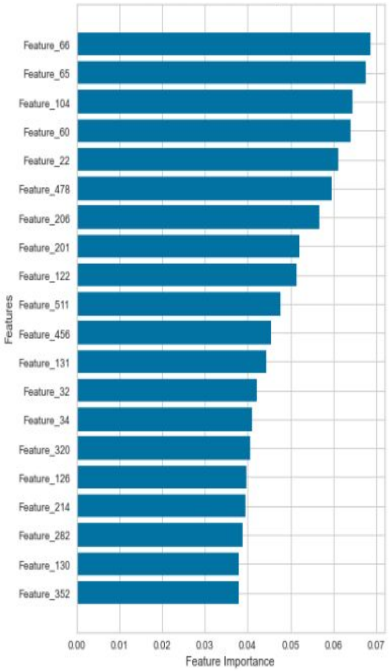
Master
Project Management
and Data Science

FEATURE SELECTION



We compared AUC and ROC

Random Forest Feature Importance



TRIAL AND ERROR PROCESS

SELECTION AND ELIMINATION

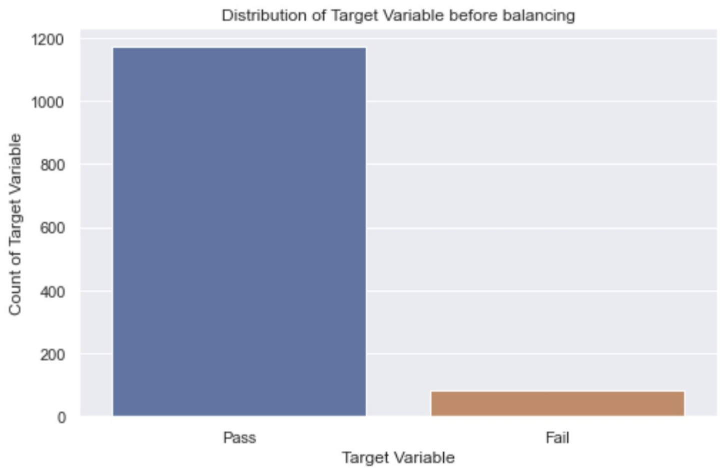
	Type I error	Type II error	F1 score	AUC	Cost	Remaining features
KNN - Boruta - SMOTE	11	74	72	70	$ 98.75	12
KNN - Boruta - SMOTE-ENN	10	88	69	73	$ 110.50	12
KNN - Boruta - SMOTE-TOMEK	11	67	75	68	$ 91.75	12
KNN - Boruta - ADASYN	12	79	71	63	$ 106.00	12
KNN - Boruta - ROSE	10	53	79	72	$ 75.50	12
KNN - Chi-Square Test - SMOTE	13	43	82	67	$ 72.25	29
KNN - Chi-Square Test - SMOTE-ENN	11	61	77	69	$ 85.75	29
KNN - Chi-Square Test - SMOTE-TOMEK	14	41	82	68	$ 72.50	29
KNN - Chi-Square Test - ADASYN	12	42	82	68	$ 69.00	29
KNN - Chi-Square Test - ROSE					$ -	29
KNN - Lasso Regression - SMOTE	14	25	87	73	$ 56.50	29
KNN - Lasso Regression – SMOTE-ENN	11	62	76	72	$ 86.75	29
KNN - Lasso Regression – SMOTE-TOMEK	16	20	87	72	$ 56.00	29
KNN - Lasso Regression - ADASYN	14	34	84	71	$ 65.50	29
KNN - Lasso Regression - ROSE					$ -	29
KNN - Boruta SHAP - SMOTE					$ -	
KNN - Forward Selection - SMOTE	13	30	86	70	$ 59.25	15
KNN - Forward Selection - SMOTE-ENN	9	49	81	73	$ 69.25	15
KNN - Forward Selection - SMOTE-TOMEK	13	30	86	72	$ 59.25	15
KNN - Forward Selection - ADASYN	14	30	85	73	$ 61.50	15
KNN - Forward Selection - ROSE	11	28	87	75	$ 52.75	15
KNN - RFE - SMOTE	14	45	81	73	$ 76.50	15
KNN - RFE - SMOTE-ENN	11	77	71	70	$ 101.75	15

	Type I error	Type II error	F1 score	AUC	Cost	Remaining Features
MICE - Boruta - SMOTE	11	76	68	72	$100.75	12
MICE - Boruta - SMOTE-ENN	9	84	74	70	$104.25	12
MICE - Boruta - SMOTE-TOMEK	11	78	71	69	$102.75	12
MICE - Boruta - ADASYN	11	82	70	63	$106.75	12
MICE - Boruta - ROSE	9	53	80	71	$ 73.25	12
MICE - Chi-Square Test - SMOTE	14	43	81	65	$ 74.50	29
MICE - Chi-Square Test - SMOTE-ENN	13	61	76	67	$ 90.25	29
MICE - Chi-Square Test - SMOTE-TOMEK	15	46	80	66	$ 79.75	29
MICE - Chi-Square Test - ADASYN	11	42	83	67	$ 66.75	29
MICE - Chi-Square Test - ROSE					$ -	29
MICE - Lasso Regression - SMOTE	14	26	87	74	$ 57.50	29
MICE - Lasso Regression - SMOTE-ENN	10	64	76	71	$ 86.50	29
MICE - Lasso Regression - SMOTE-TOMEK	14	26	87	74	$ 57.50	29
MICE - Lasso Regression - ADASYN	15	36	83	74	$ 69.75	29
MICE - Lasso Regression - ROSE					$ -	29
MICE - Boruta SHAP - SMOTE					$ -	6
MICE - Forward Selection - SMOTE	12	29	86	73	$ 56.00	
MICE - Forward Selection - SMOTE-ENN	10	46	82	70	$ 68.50	
MICE - Forward Selection - SMOTE-TOMEK	14	27	86	71	$ 58.50	
MICE - Forward Selection - ADASYN	12	32	85	72	$ 59.00	
MICE - Forward Selection - ROSE	10	35	85	76	$ 57.50	

There is *no best feature selection method*. Instead, we have discovered what works best for the specific problems related to this Dataset using careful systematic experimentation/trial and error method. We tried a range of different models fit on different subsets of features chosen via different statistical measures and *discovered the ones works best*.

BALANCING METHODS

BALANCING METHODS



Problem with Imbalance Dataset

Standard learners are often biased towards the majority class. The reason is because these classifiers attempt to reduce overall quantities such as error rate, not taking the data distribution into consideration, therefore, tend to bias performance towards the majority class.

Solutions:

Re-sampling provides a simple way of biasing the generalization process. It can do so by:

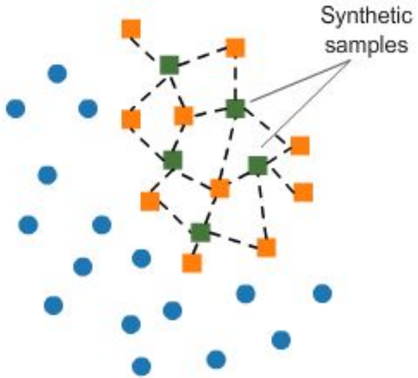
- Generating synthetic samples accordingly biased
- Controlling the amount and placement of the new samples

Methods:

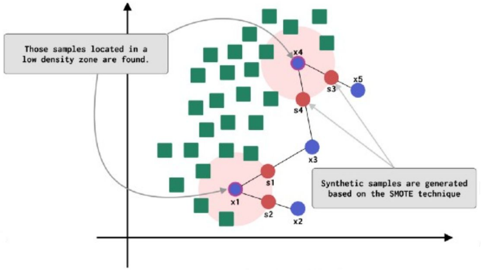
- Synthetic Minority Oversampling Technique (SMOTE)
- Adaptive Synthetic (ADASYN)
- Random Over-Sampling Examples (ROSE)

Master
Project Management
and Data Science

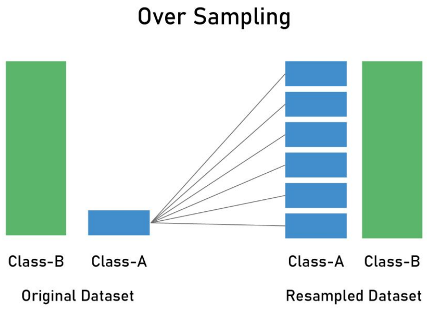
BALANCING METHODS



SMOTE



ADASYN



ROSE

Synthetic Minority Oversampling Technique (SMOTE)

It combines Informed Oversampling of the minority class with Random Undersampling of the majority class.

Adaptive Synthetic (ADASYN)

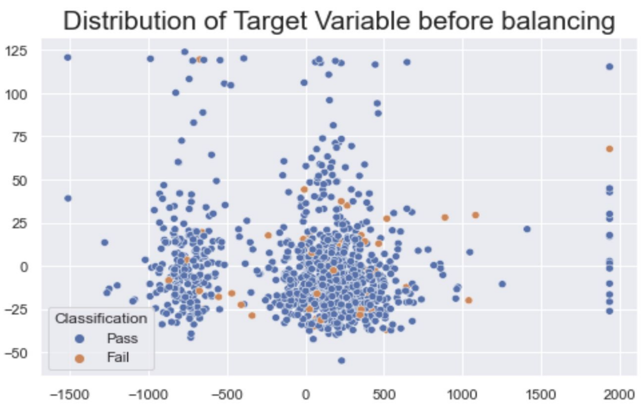
It uses a weighted distribution for different minority class examples according to their level of difficulty in learning, where more synthetic data generated for minority class examples that are harder to learn compared to those minority examples that are easier to learn

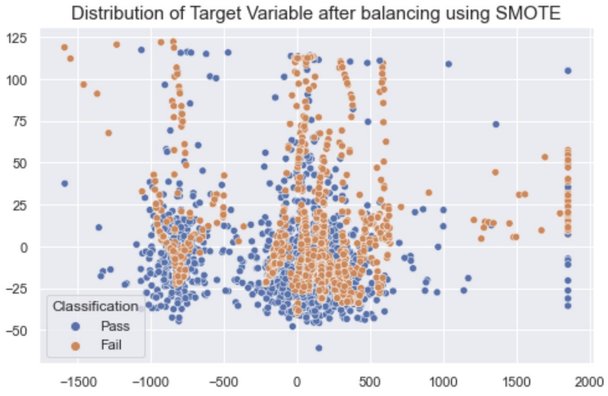
Random Over-Sampling Examples (ROSE)

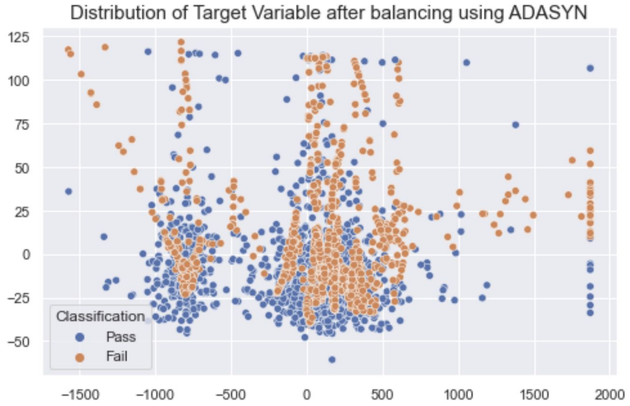
It is a bootstrap-based technique which aids the task of binary classification in the presence of rare classes.

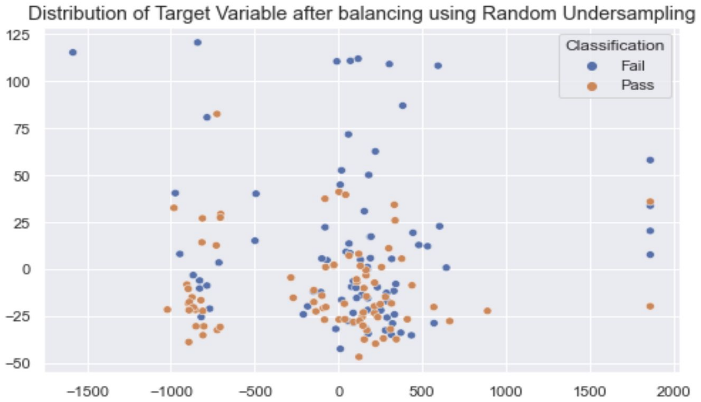
Master
Project Management
and Data Science

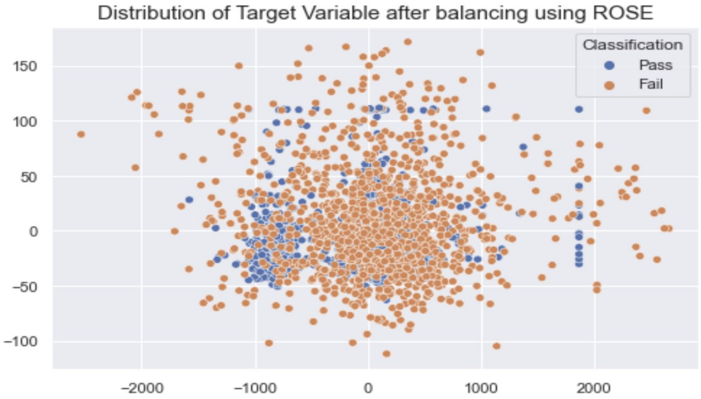
BALANCING METHODS







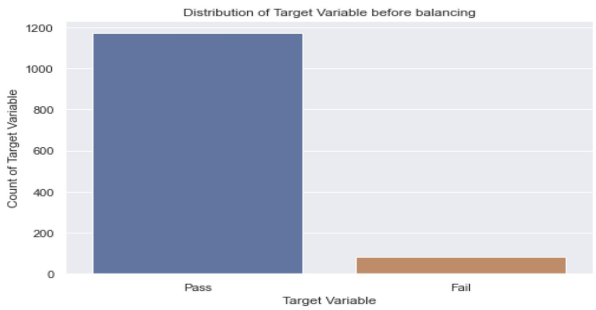


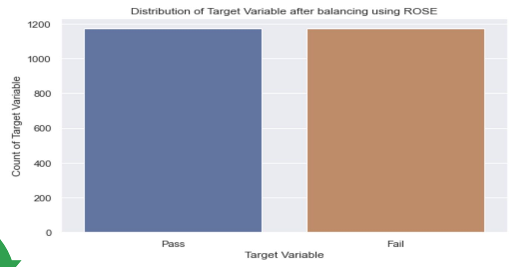


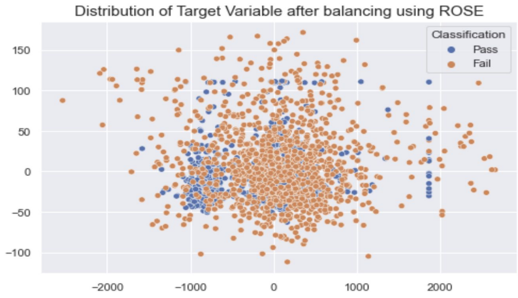
BALANCING METHOD

Why we chose a certain balancing method?

ROSE







SCALING

SCALING

We are almost ready to fit the model, but…

Features have different Dimensions…

We applied scaling in KNN. Scaling is recommended

KNN

Since the approach relying on measuring distances

SCALING needs RESCALING

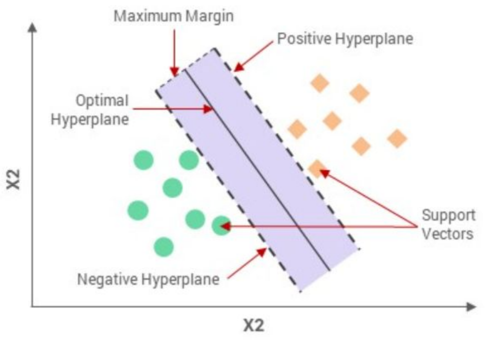
To evaluate the results on the same scale, the final model is applied to generate the predictions.

TEMPORARY SCALING

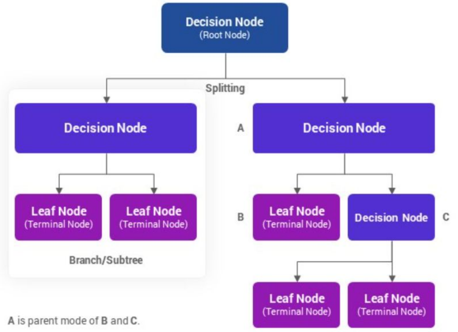
MACHINE LEARNING MODELS

Master
Project Management
and Data Science

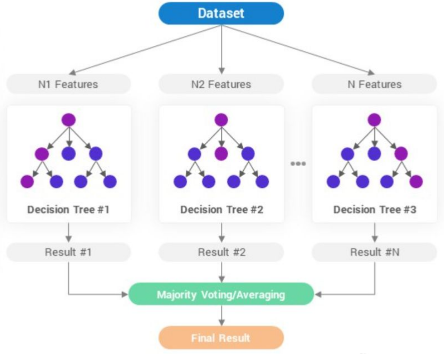
MACHINE LEARNING MODELS



Support Vector Machine (SVM)



Decision Tree Classifier



Random Forest Classifier

Support Vector Machine (SVM)

It creates the best line or decision boundary (hyperplane) that can segregate n-dimensional space into classes so that new data point can be easily put in correct category in the future.

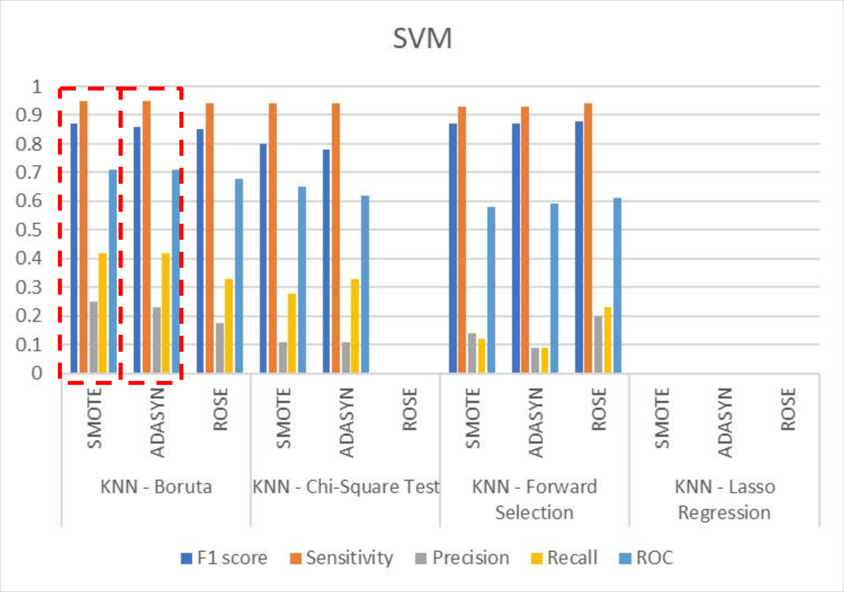
Decision Tree Classifier

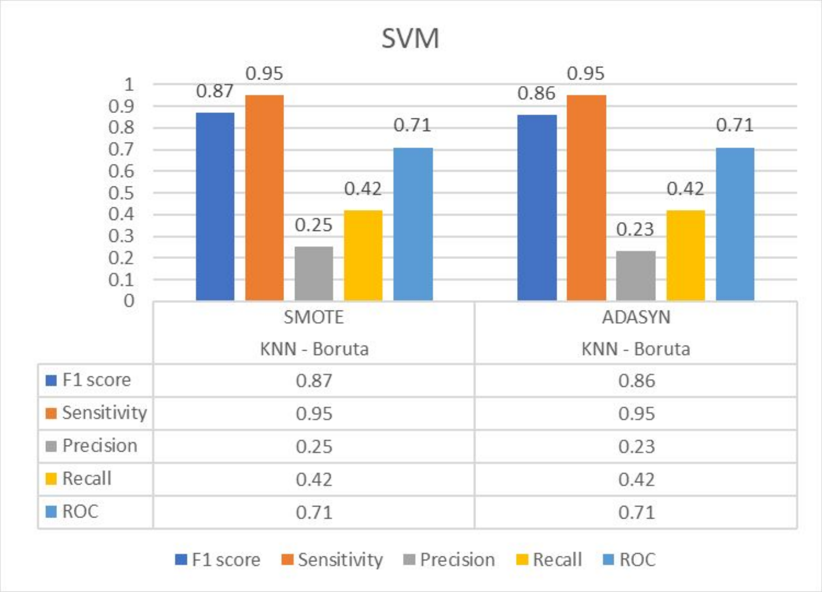
It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.

Random Forest Classifier

It combines hundreds of decision trees and trains each decision tree on a different sample of the observations. The final predictions are made by averaging the predictions of each individual tree.

RESULTS WITH SVM





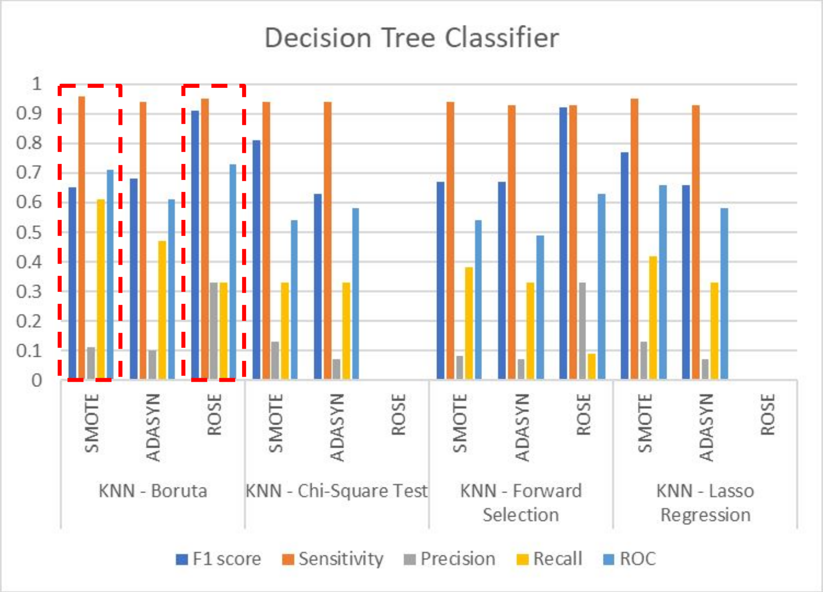
	SMOTE KNN - Boruta	ADASYN KNN - Boruta
F1 score	0.87	0.86
Sensitivity	0.95	0.95
Precision	0.25	0.23
Recall	0.42	0.42
ROC	0.71	0.71

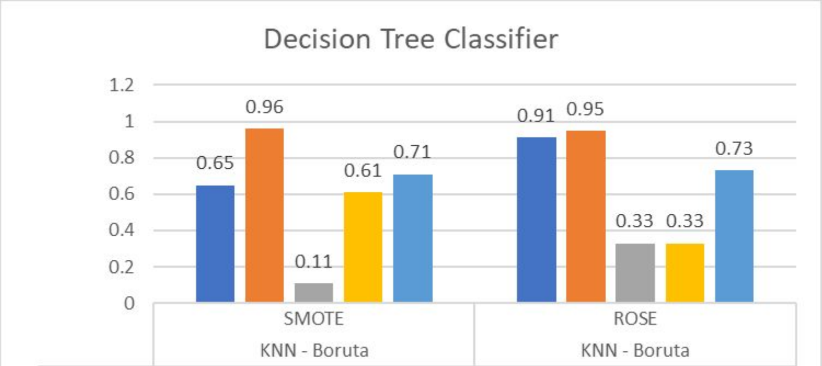
Best combinations in terms of accuracy:

- KNN-Boruta-SMOTE
- KNN-Boruta-ADASYN

Master
Project Management
and Data Science

RESULTS WITH DECISION TREE CLASSIFIER





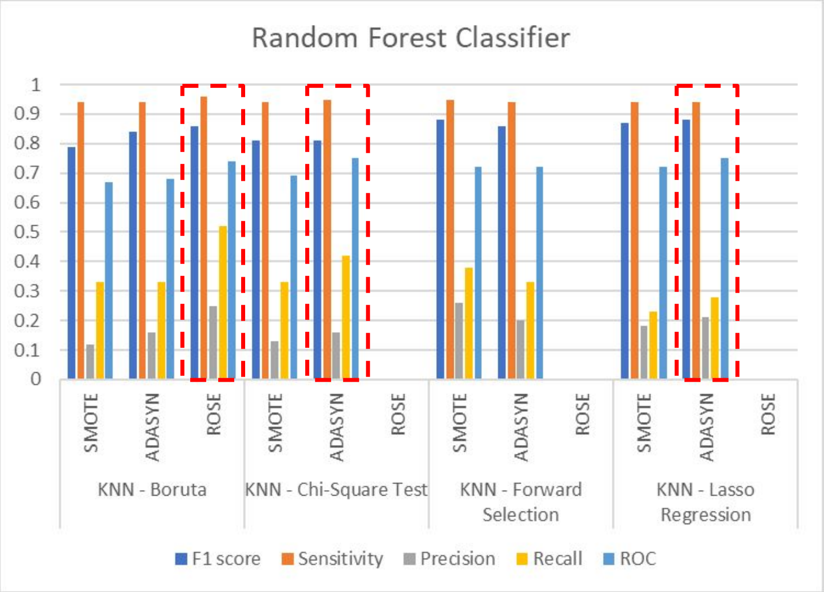
	SMOTE KNN - Boruta	ROSE KNN - Boruta
F1 score	0.65	0.91
Sensitivity	0.96	0.95
Precision	0.11	0.33
Recall	0.61	0.33
ROC	0.71	0.73

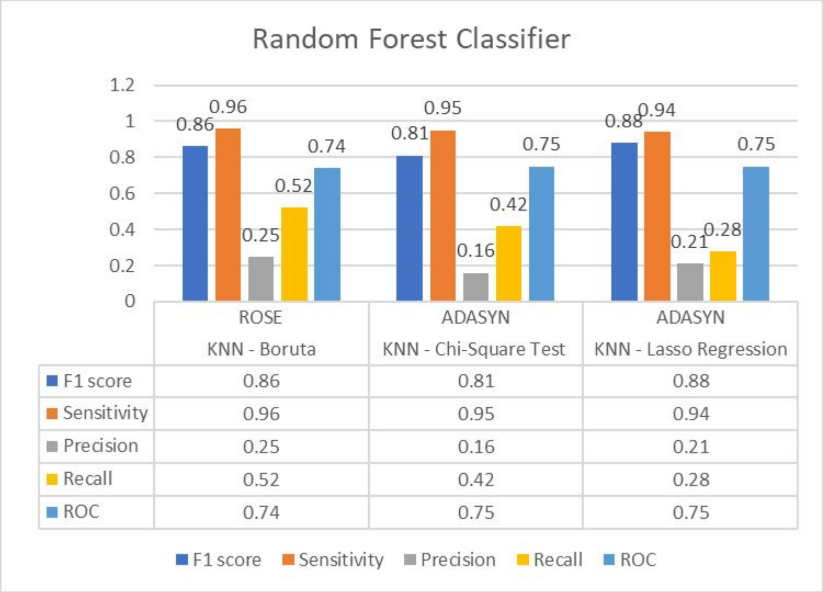
Best combinations in terms of accuracy:

- KNN-Boruta-SMOTE
- KNN-Boruta-ROSE

Master
Project Management
and Data Science

RESULTS WITH RANDOM FOREST CLASSIFIER





Best combinations in terms of accuracy:

- KNN-Boruta-SMOTE
- KNN-Chi-Square Test-ADASYN
- KNN-Lasso Regression-ADASYN

EVALUATION PROCESS

CONFUSION MATRIX

PREDICTION

ACTUAL	Positive	Negative
Positive	TRUE POSITIVE (TP)	TYPE II ERROR FALSE NEGATIVE (FN)
Negative	TYPE I ERROR FALSE POSITIVE (FP)	TRUE NEGATIVE (TN)

CONFUSION MATRIX

Positive = **Pass** classification
Negative = **Fail** classification

True Positive: Pass classification correctly identified as Pass

True Negative: Fail classification correctly identified as Fail

False Positive: Pass classification identified as Fail

False Negative: Fail classification identified as Pass

COST MODEL

Cost of TP

....

Cost of TN

.....

$225

Cost of FP

- Manufacturing & Shipping
- Recall cost
- Reshipping
- Damages

Type I Error

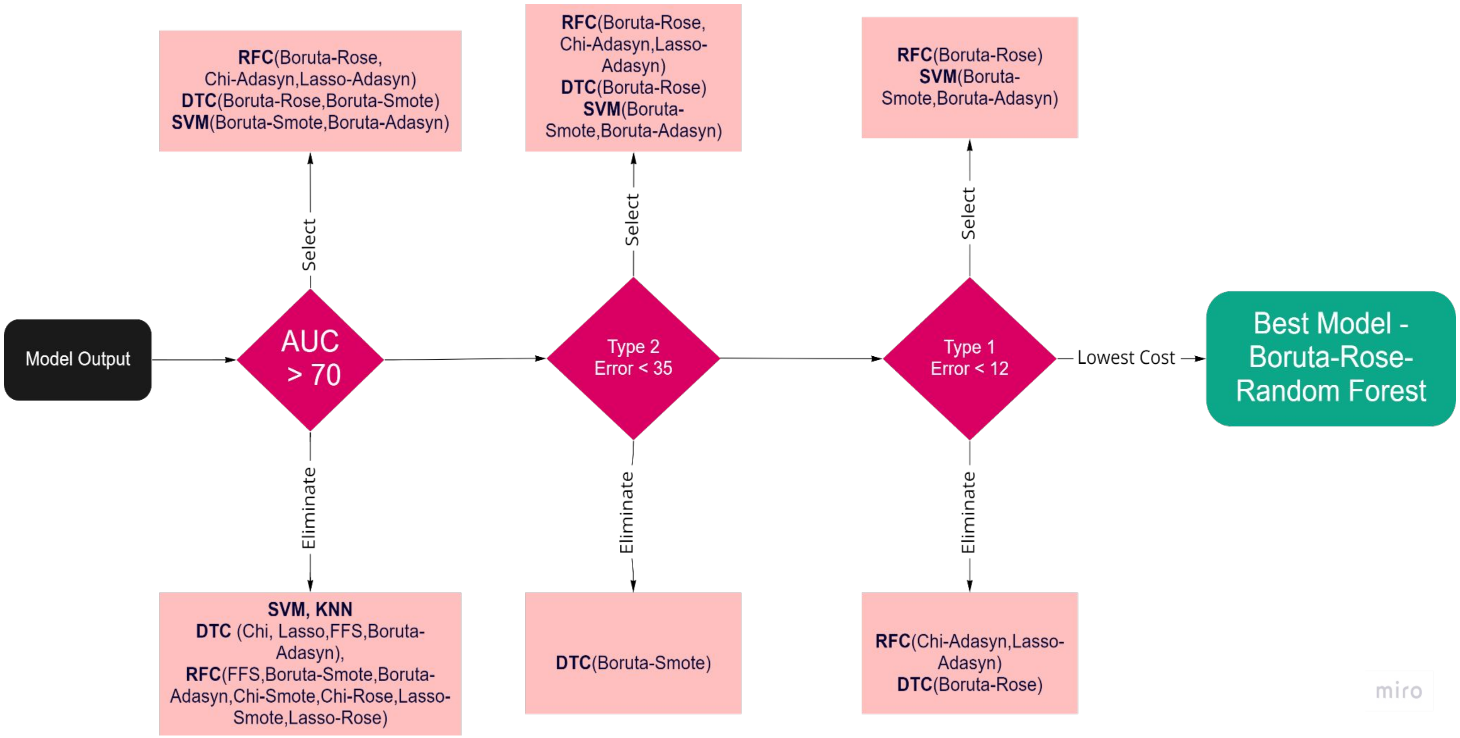
$100

Cost of FN

Manufacturing & Shipping

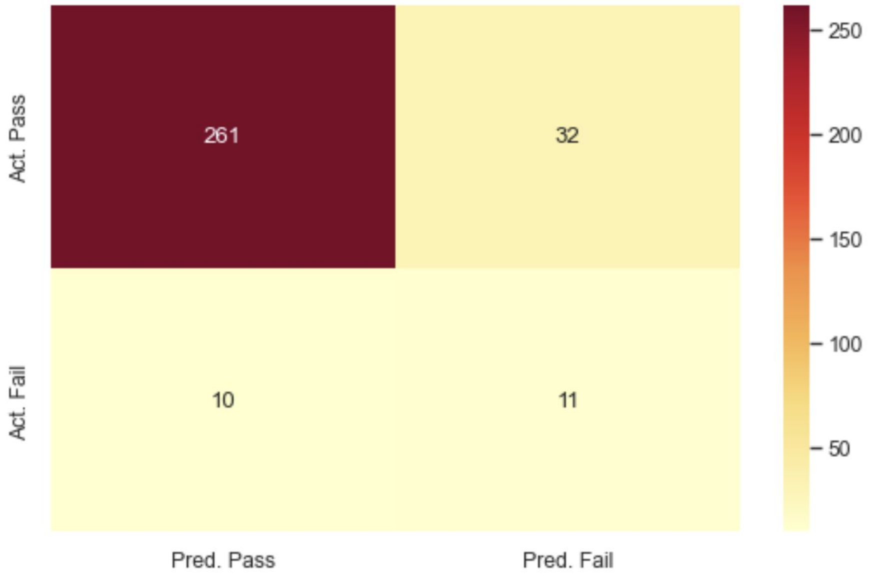
Type II Error

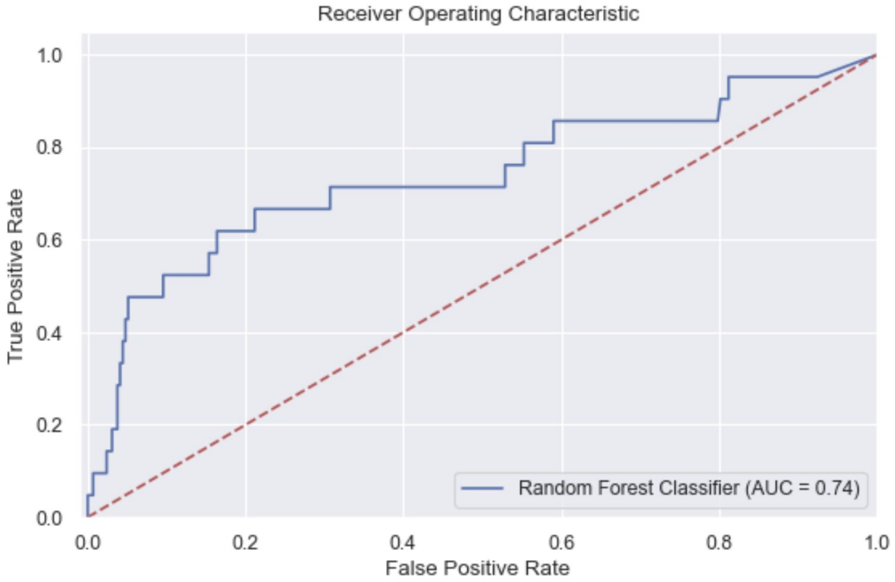
ELIMINATION OF MODELS



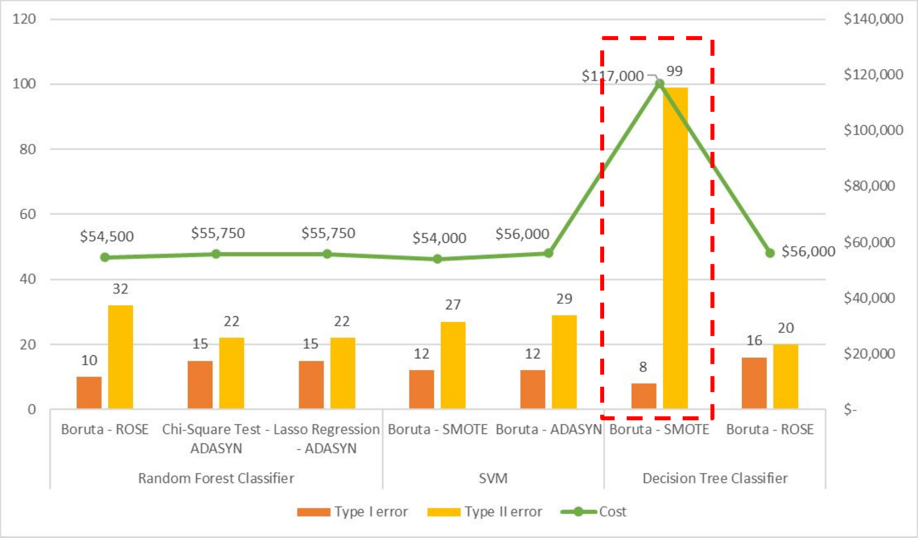
RANDOM FOREST CLASSIFIER

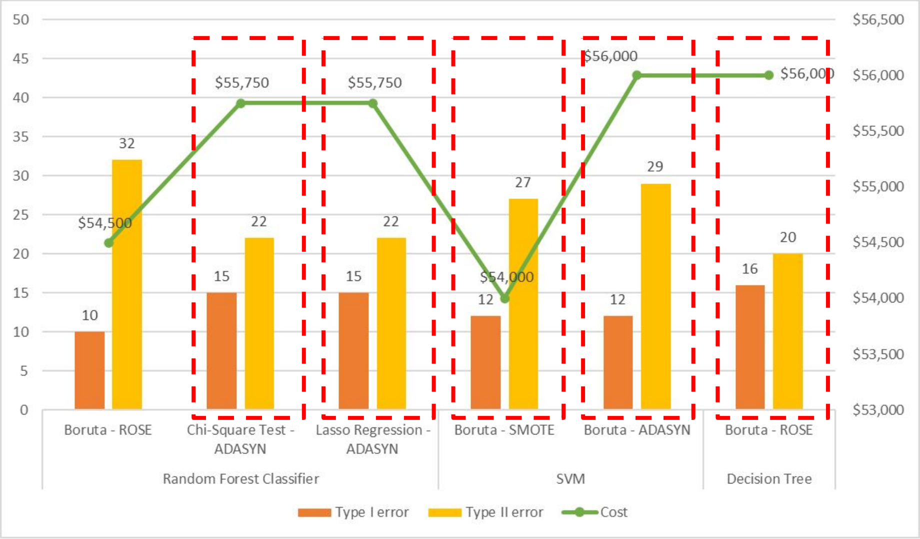
KNN-Boruta-ROSE





RESULTS WITH COST MODEL





Type 2 Error < 35

Lowest Type 1 Error

Best Model: KNN-Boruta-ROSE using Random Forest Classifier

- Type I error: 10
- Type II error: 32
- Cost: $54,500

MODEL OPTIMISATION

MODEL OPTIMISATION

...

MODEL EVALUATION

02

Error Function

An error function serves to evaluate the prediction of the model. If there are known examples, an error function can make a comparison to assess the accuracy of the model

……

……

01

Decision Process

Machine Learning algorithms are used to make a prediction or classification. Based on the input data, the algorithm will produce an estimate about a pattern in the data

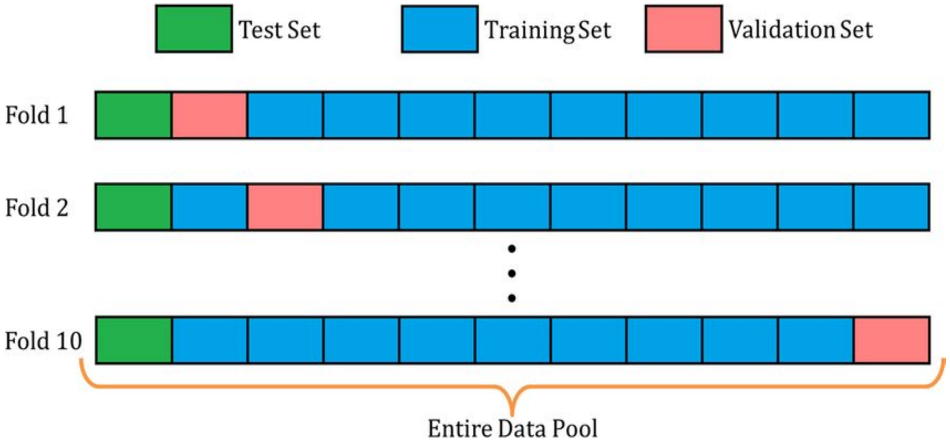
03

Model Optimisation Process

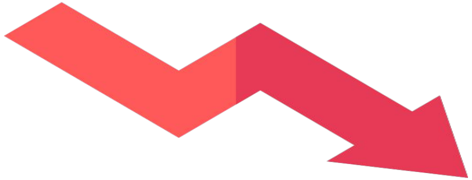
If the model can fit better to the data points in the training set, weights are adjusted to reduce the discrepancy between the known example and the model estimate. The algorithm will repeat this evaluate and optimize process, updating weights autonomously until a threshold of accuracy has been met

CROSS VALIDATION

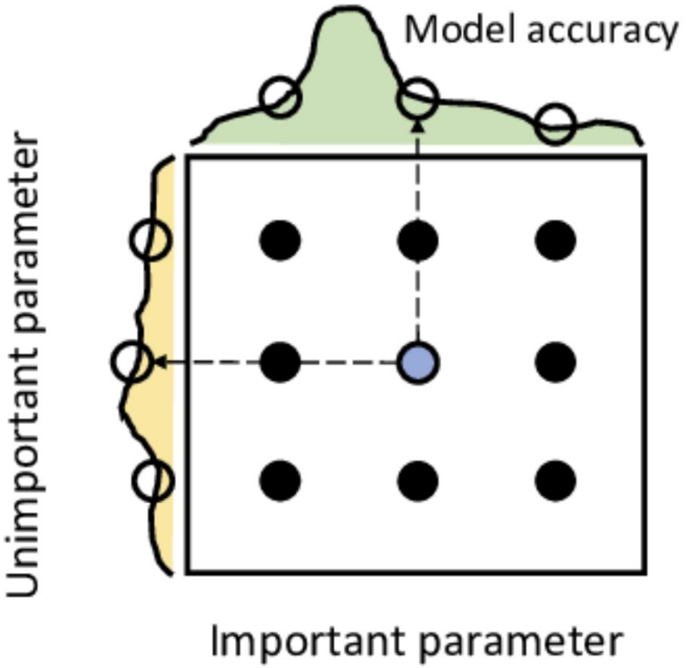
For Cross Validation we have used a **10-fold** approach in which the data is split randomly in 10 subsets that have the same number of samples. The steps described in the next subsections are repeated 10 times and each time the testing data will be one distinct fold from the set of the 10 folds and the training data will consist of the remaining 9 folds.



REDUCES OVERFITTING PROBLEM



HYPERPARAMETER TUNING WITH GRID SEARCH CV

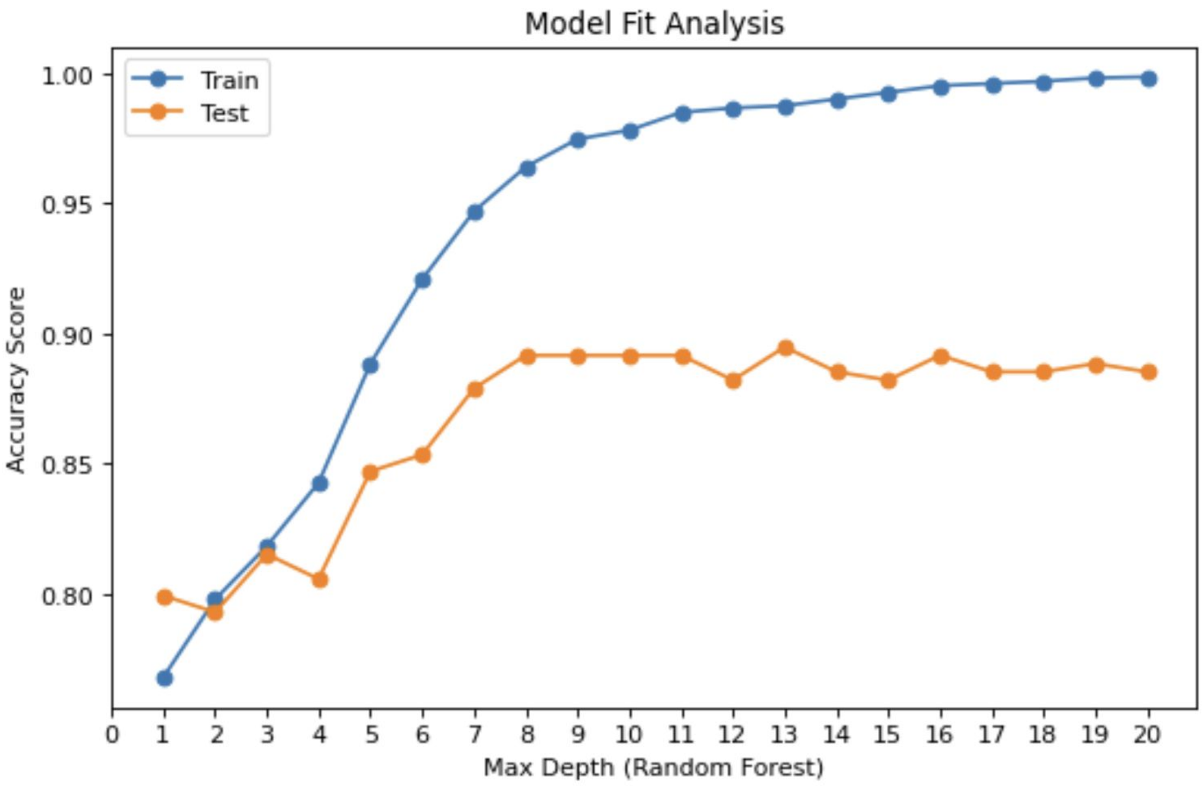


GridSearchCV

```
GridSearchCV(cv=10,
             estimator=RandomForestClassifier(n_jobs=-1, random_state=42),
             param_grid={'criterion': ['gini', 'entropy', 'log_loss'],
                         'max_depth': [4, 5, 6, 7, 8, 9, 10],
                         'max_features': ['sqrt', 'log2', 'auto', None]},
             scoring='roc_auc', verbose=1)
```

Master
Project Management
and Data Science

HYPERTUNING FOR MAXIMUM DEPTH



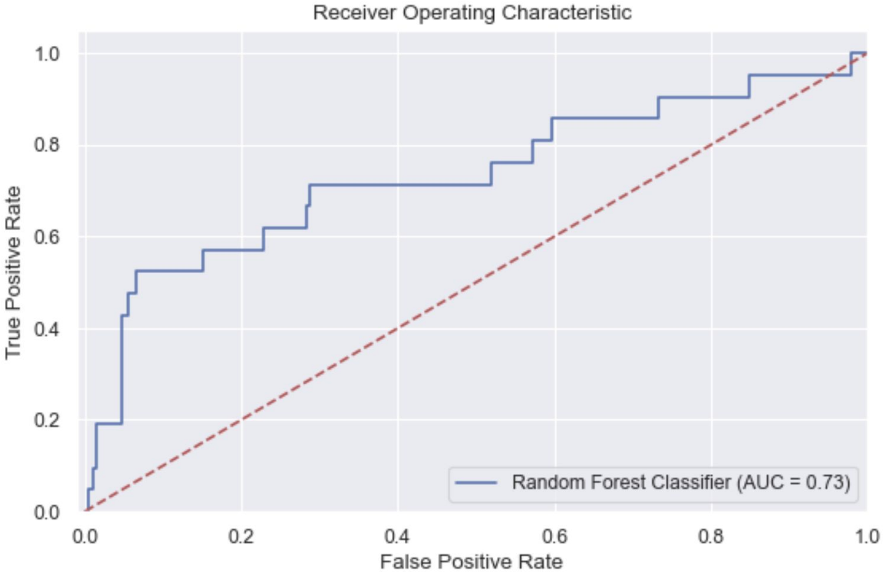
RESULTS

Master
Project Management
and Data Science

FINAL RESULT (KNN - Boruta - ROSE with Random Forest Classifier)

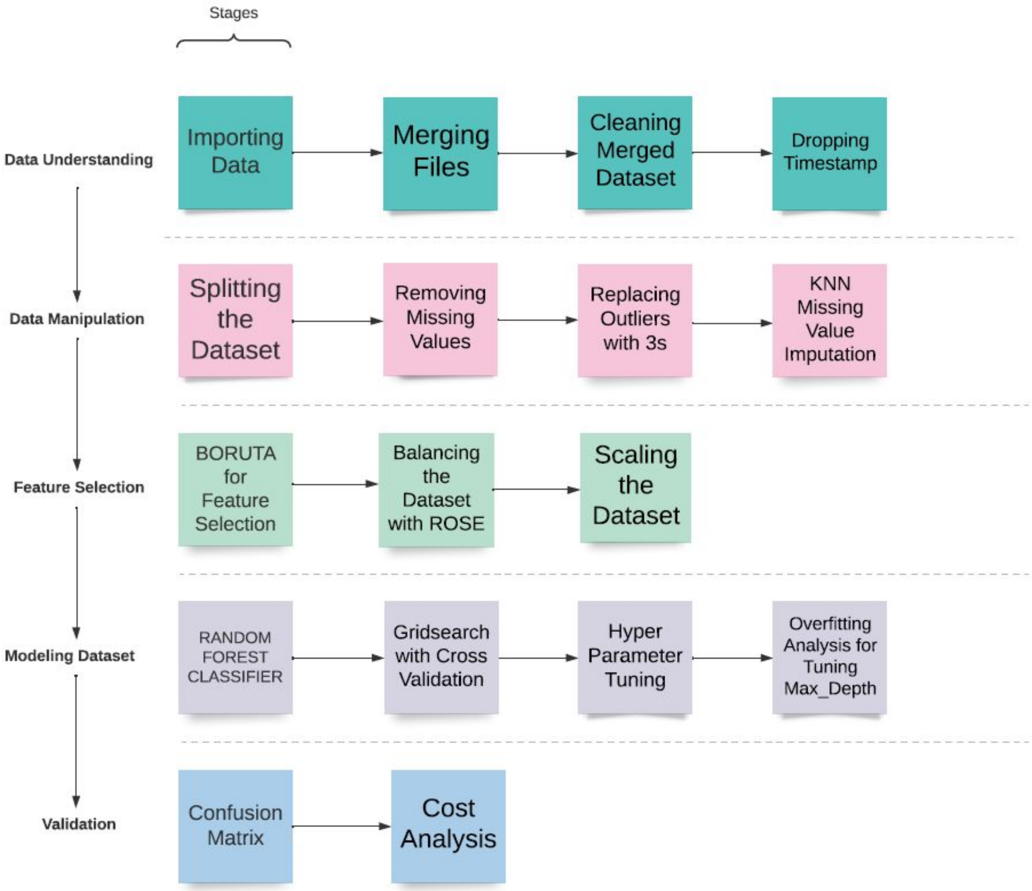
CONFUSION MATRIX

AUC



From $ 54500 **$ 46500**

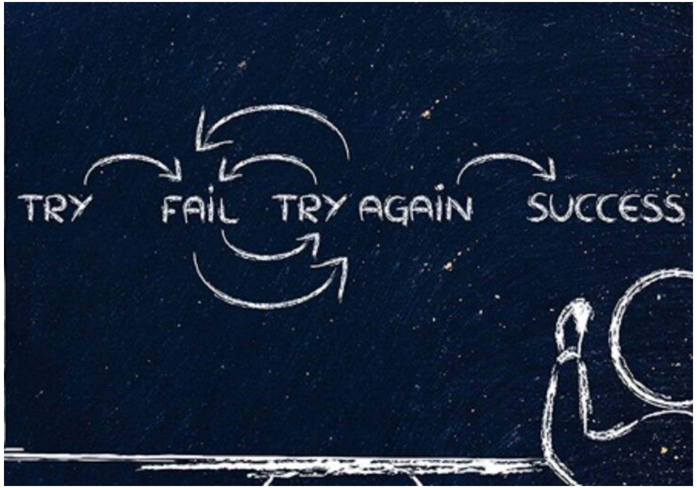
SUMMARY


Stages
Data Understanding
Importing Data
Merging Files
Cleaning Merged Dataset
Dropping Timestamp
Data Manipulation
Splitting the Dataset
Removing Missing Values
Replacing Outliers with 3s
KNN Missing Value Imputation
Feature Selection
BORUTA for Feature Selection
Balancing the Dataset with ROSE
Scaling the Dataset
Modeling Dataset
RANDOM FOREST CLASSIFIER
Gridsearch with Cross Validation
Hyper Parameter Tuning
Overfitting Analysis for Tuning Max_Depth
Validation
Confusion Matrix
Cost Analysis

Master
Project Management
and Data Science

Lesson Learned and Best Practices

1. Using CRISP DM
2. Complete Data
3. Outlier treatment and imputation of data
4. Balancing the data
5. Scaling the data
6. Business Understanding and defining model
7. Model Quality



Master
Project Management
and Data Science

Vielen Dank!

Gupta, Himansha
Himansha.Gupta@student.htw-berlin.de

Pomay Polat, Ekin
Ekin.PomayPolat@student.htw-berlin.de

Dsouza, Rashmi Carol
Rashmi.Dsouza@Student.HTW-Berlin.de

Pham, Quynh Dinh Hai
Quynh.Pham@Student.HTW-Berlin.de

www.mpmd.htw-berlin.de