



SECOM ANALYTICS - PART 2

TEAM 3

Gupta, Himansha

Pomay Polat, Ekin

Dsouza, Rashmi Carol

Pham, Quynh Dinh Hai

TABLE OF CONTENTS

01

DATA PREPARATION

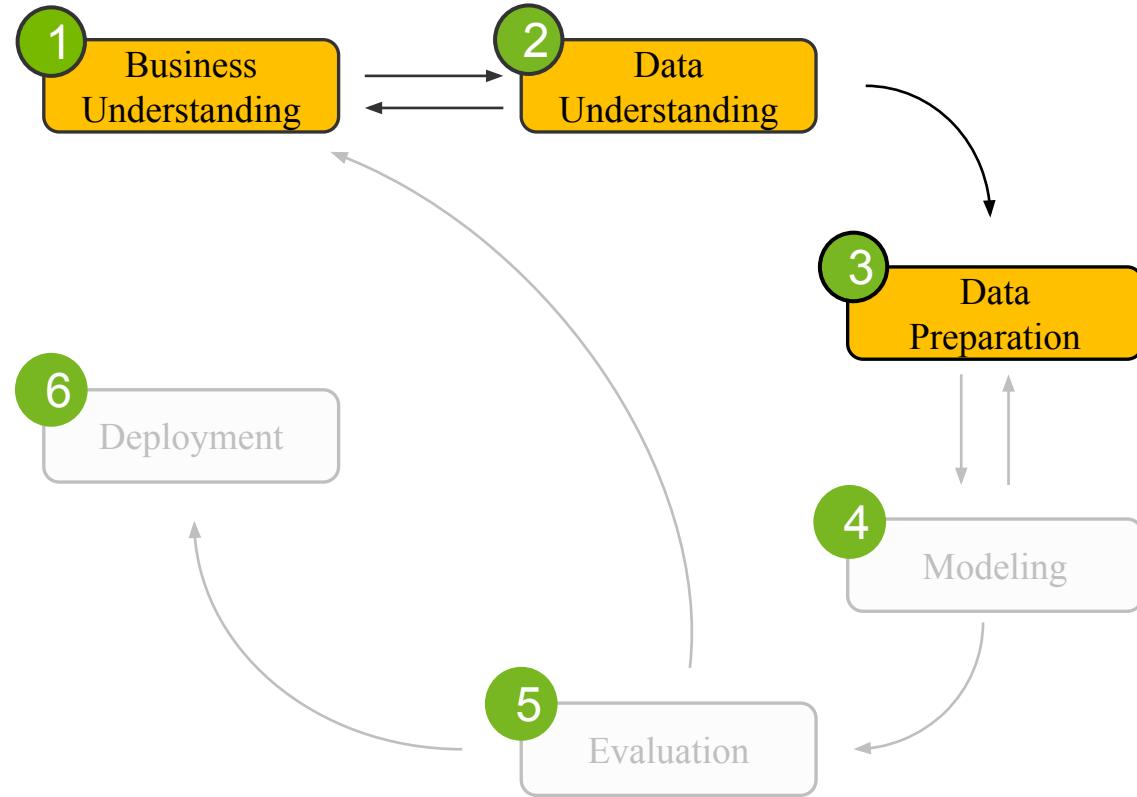
- Overview
- Our Approach
- Merge Data
- Separate Training & Test sets
- Data Preparation Steps
- Top 4 combination

02

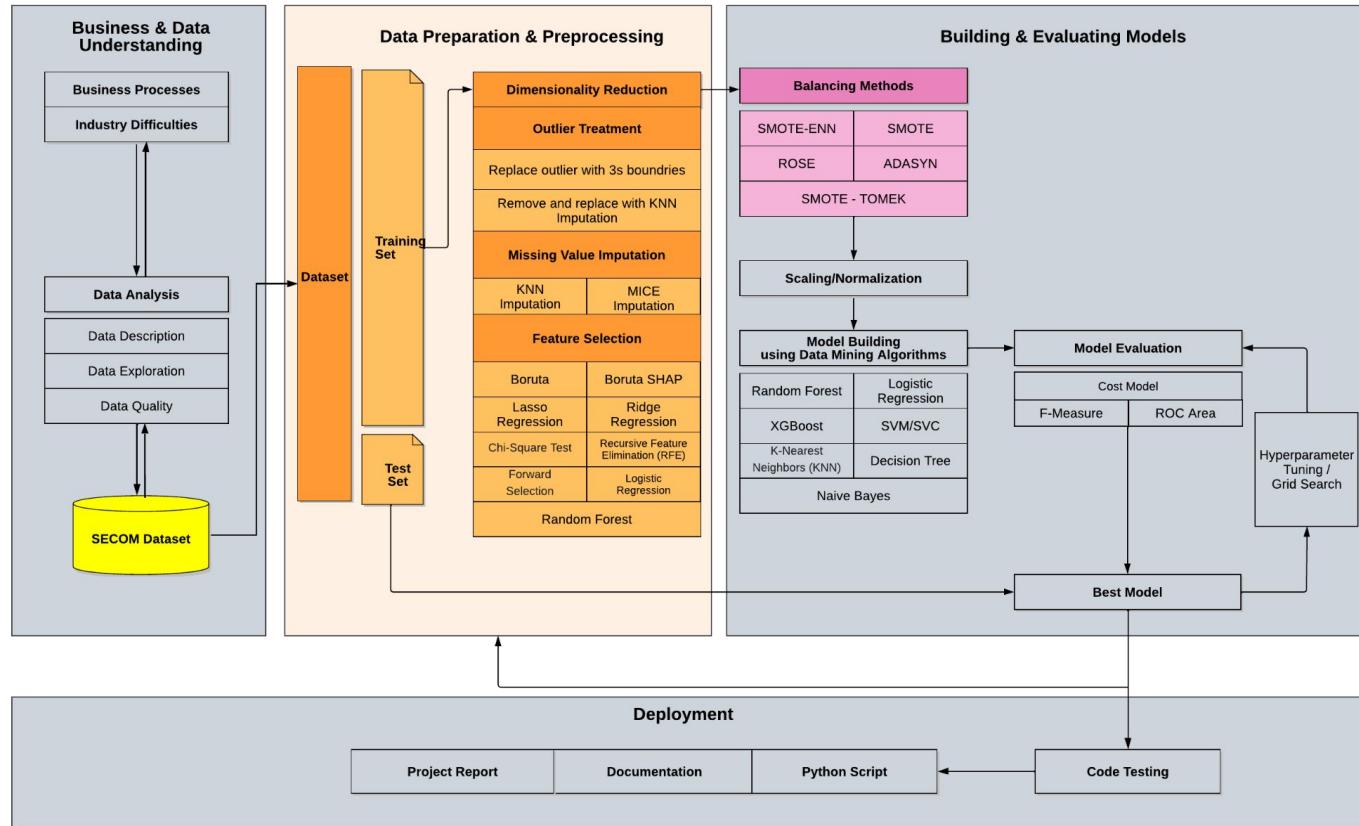
NEXT STEPS

01

DATA PREPARATION

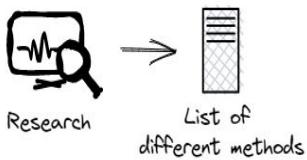


OVERVIEW



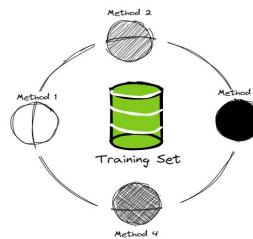
DATA PREPARATION APPROACH

01



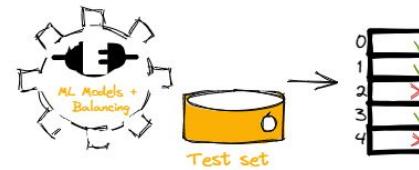
Research different methods for each step of data preparation

02



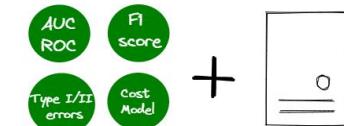
Use training set to train the machine with different method

03



Use different machine learning models, balancing methods & test set to test each method

04



Use AUC, ROC, type I/II errors, F1 score & cost model as measures for decision making & document our findings

RESULT MEASUREMENTS

RESULT MEASURES

PREDICTION

		Positive	Negative
ACTUAL	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

RESULT MEASURES

Precision

$$\frac{\text{TP}}{\text{TP} + \text{FP}}$$

Sensitivity

$$\frac{\text{TP}}{\text{TP} + \text{FN}}$$

Specificity

$$\frac{\text{TN}}{\text{TN} + \text{FP}}$$

Negative Predictive Value

$$\frac{\text{TN}}{\text{TN} + \text{FN}}$$

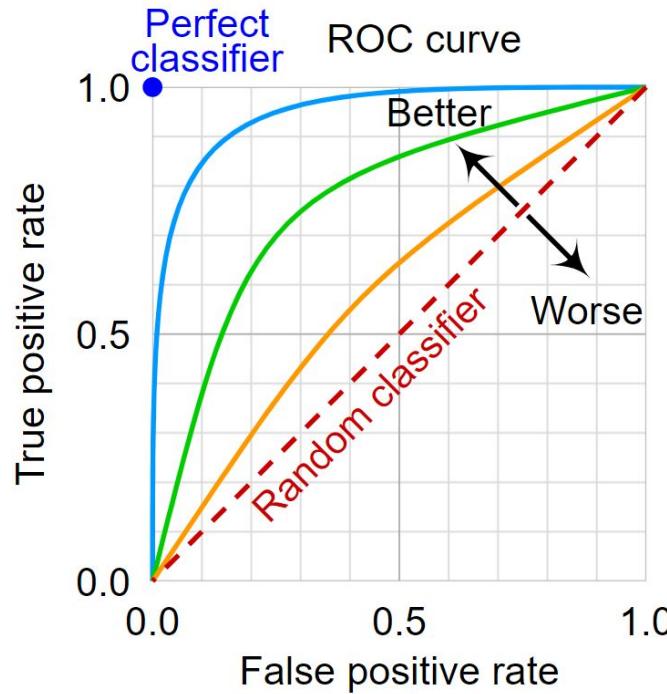
- **Sensitivity/Recall:** the ratio between how much were correctly identified as positive to how much were actually positive.
- **Specificity:** the ratio between how much were correctly classified as negative to how much was actually negative.
- **Precision:** How much were correctly classified as positive out of all positives.
- **NPV:** How much were correctly classified as negative out of all negatives

F1-Score

$$\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

F1 Score: combines precision and recall into one metric by calculating the harmonic mean between those two. It is primarily used to compare the performance of two classifiers. Classifier A has a higher recall, and classifier B has higher precision. In this case, the F1-scores for both the classifiers are used to determine which one produces better results.

ROC CURVE



- ROC curve is used to visualize the trade off between Sensitivity and Specificity.
- AUC is the area under the curve
- **Sensitivity on the Y-axis**
- **Specificity on the X-axis**

COST MODEL

Because of trade-off between Sensitivity & Specificity, we think in terms of business and what decision to make that is good for business, which is to **minimize the cost** as much as possible → **build a cost model** for method comparison & decision making

01

Cost of TP

02

Cost of TN

03

Type I Error

Cost of FP = \$225

- Manufacturing & Shipping
- Recall cost
- Reshipping
- Damages

04

Type II Error

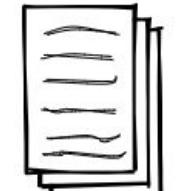
Cost of FN = \$100

- Manufacturing & Shipping

DATA PREPARATION

MERGING DATA

```
3030.93 2564 2187.  
284 0.4734 0.0167  
71 31.8843 NaN NaN  
11.5074 0.1096 0.6
```

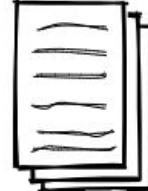


Secom.data

SECOM Dataset



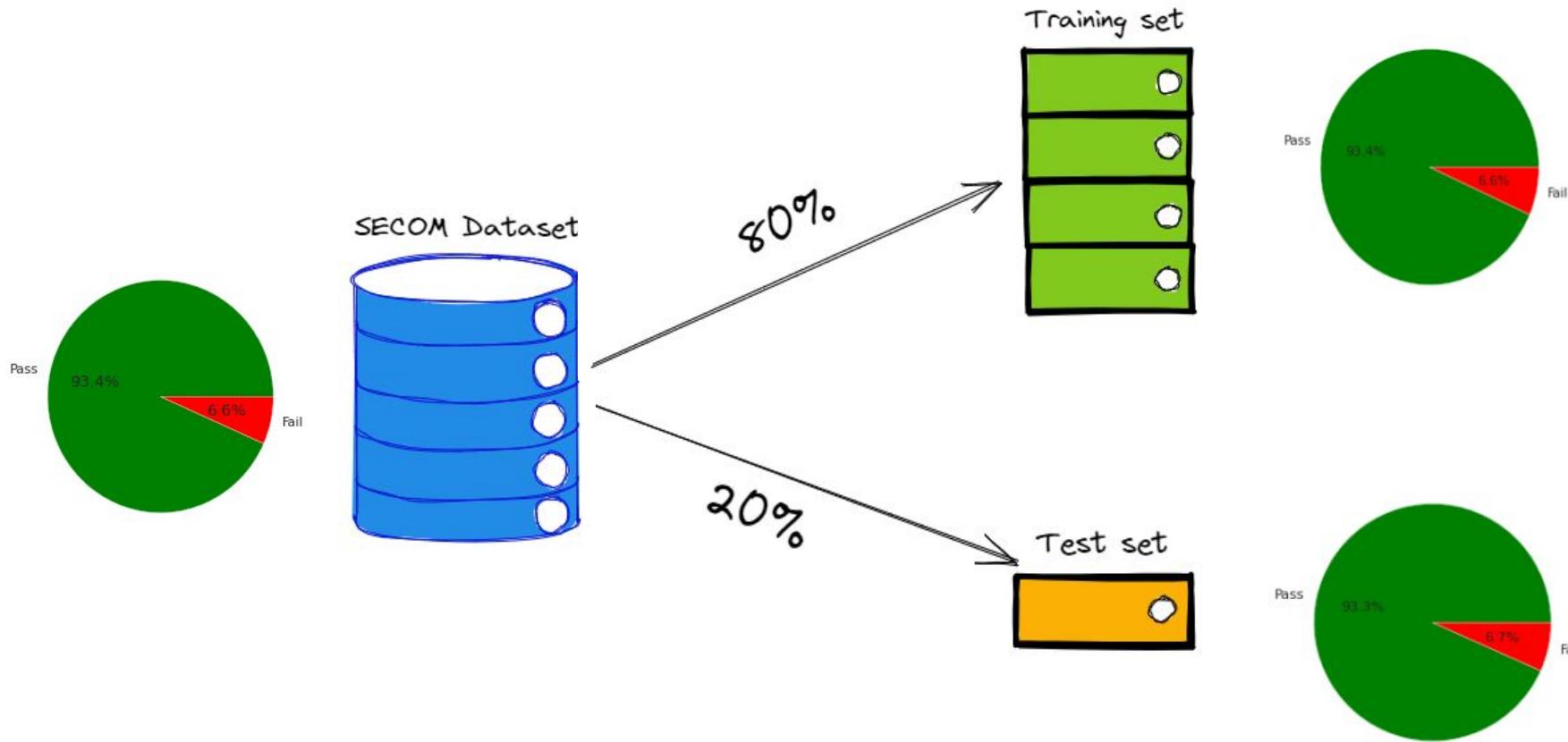
```
-1 "19/07/2008 11:55:00"  
-1 "19/07/2008 12:32:00"  
1 "19/07/2008 13:17:00"  
-1 "19/07/2008 14:43:00"  
-1 "19/07/2008 15:22:00"  
-1 "19/07/2008 17:53:00"
```



Secom_labels.dat

	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	...

SEPARATE TRAINING & TEST DATA

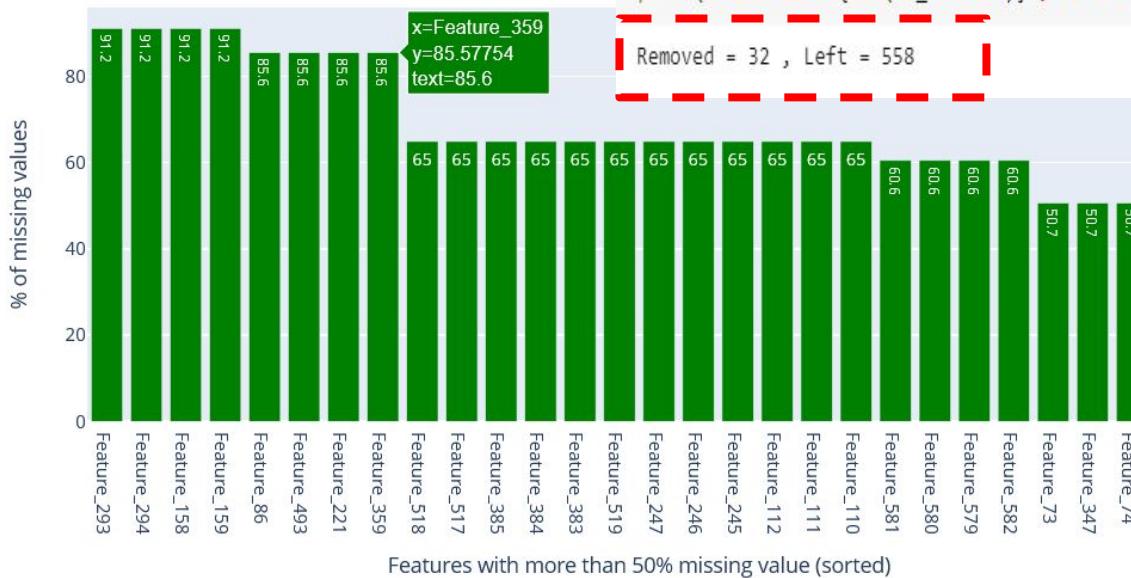


DIMENSIONALITY REDUCTION

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

Threshold:

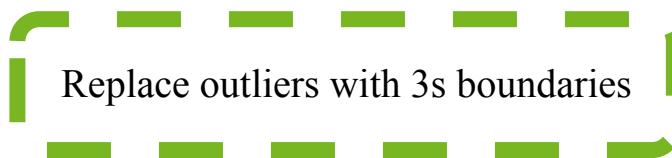
Features with more than 50% missing value



```
[ 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}')
```

OUTLIER TREATMENT



Replace outliers with 3s boundaries

Remove Outliers First and apply KNN
Imputation

We tried both methods and we get a better result with replacing outliers with 3s boundaries => **we chose to replace the outliers with 3s boundaries**

FEATURE EXPLORATION

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					\$ -	29
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**.

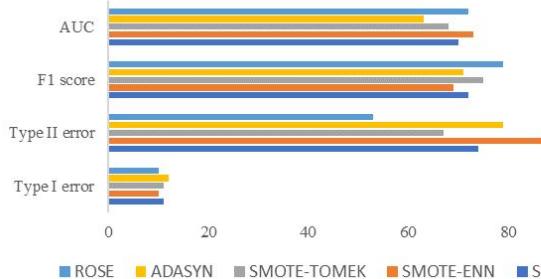
COMBINATIONS

Missing Value Imputation	Feature Selection	Balancing
KNN imputation	Boruta	SMOTE-ENN
MICE imputation	Boruta SHAP	SMOTE
	Lasso Regression	ROSE
	Ridge Regression	ADASYN
	Chi-Square Test	SMOTE-TOMEK
	Recursive Feature Elimination (RFE)	
	Forward Selection	
	Logistic Regression	
	Random Forest	

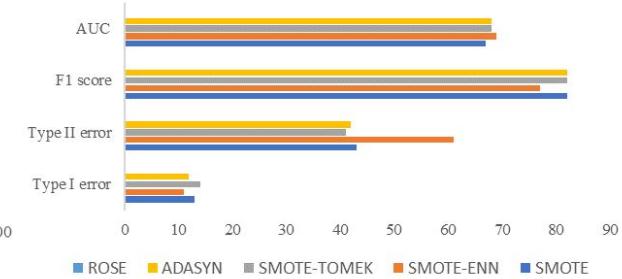
1. Missing Value Imputation
2. Feature Selection
3. Balancing

EVALUATION METRICS (KNN)

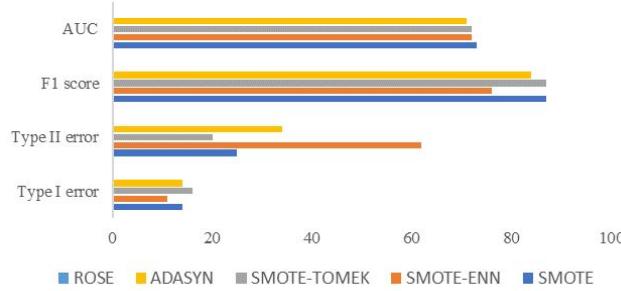
KNN - Boruta



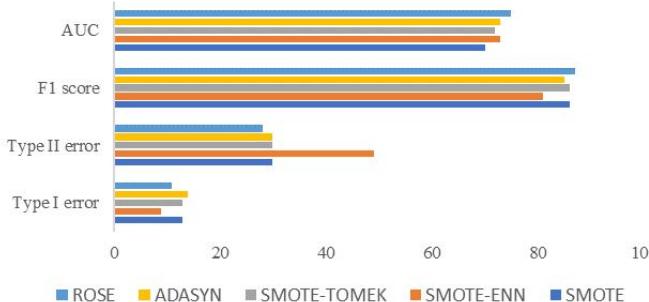
KNN - Chi-Square Test



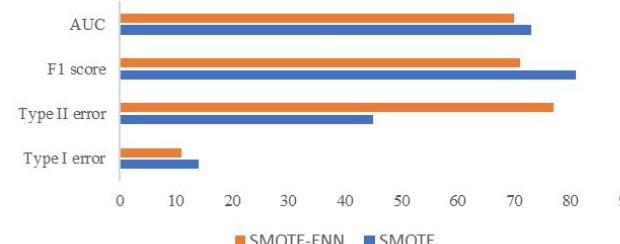
KNN - Lasso Regression



KNN - Forward Selection

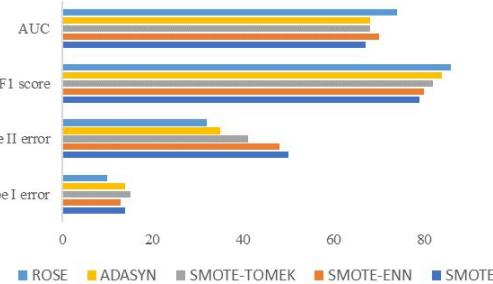


KNN - Recursive Feature Elimination



EVALUATION METRICS (KNN)

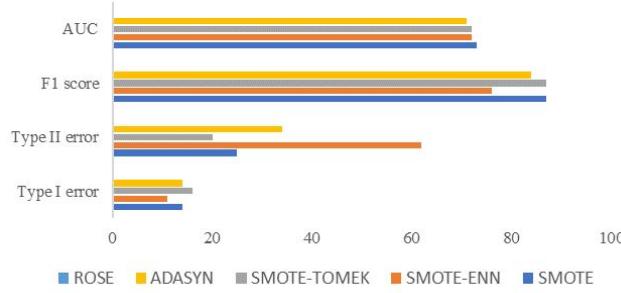
KNN - Boruta



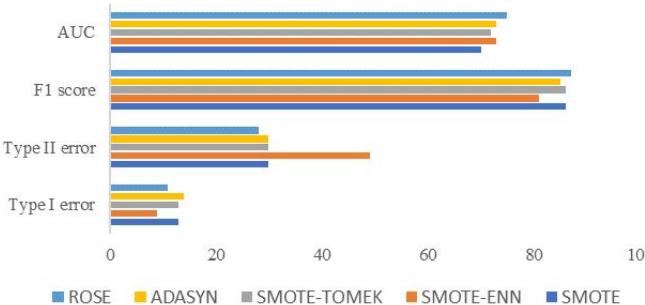
KNN - Chi-Square Test



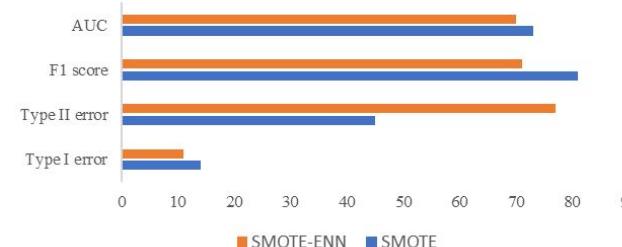
KNN - Lasso Regression



KNN - Forward Selection

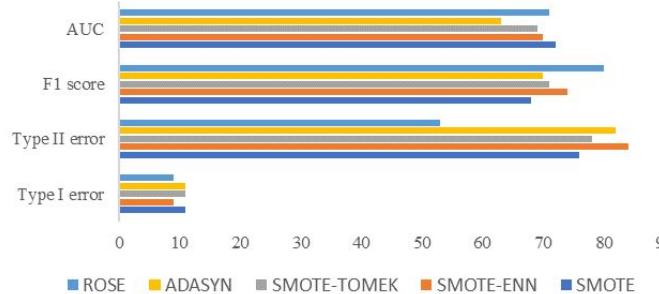


KNN - Recursive Feature Elimination

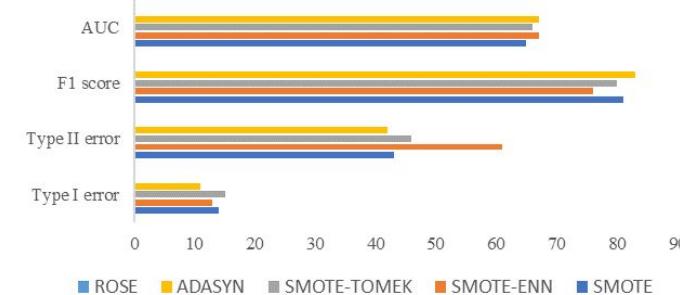


EVALUATION METRICS (MICE)

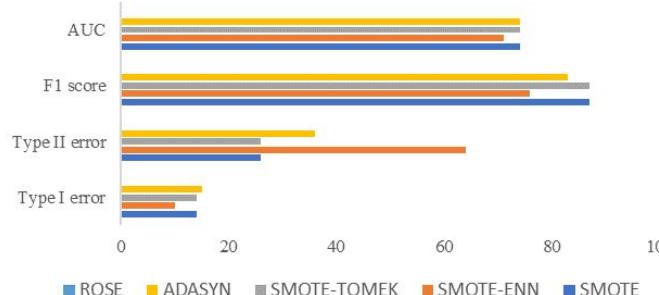
MICE - Boruta



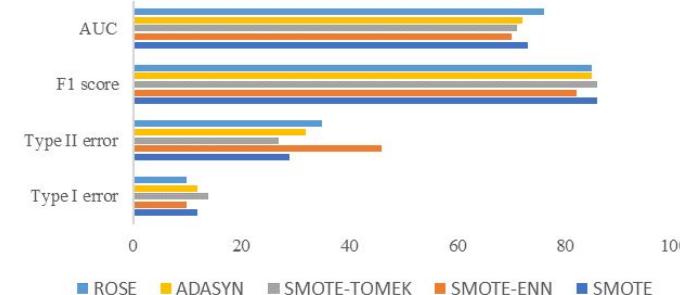
MICE - Chi-Square Test



MICE - Lasso Regression



MICE - Forward Selection



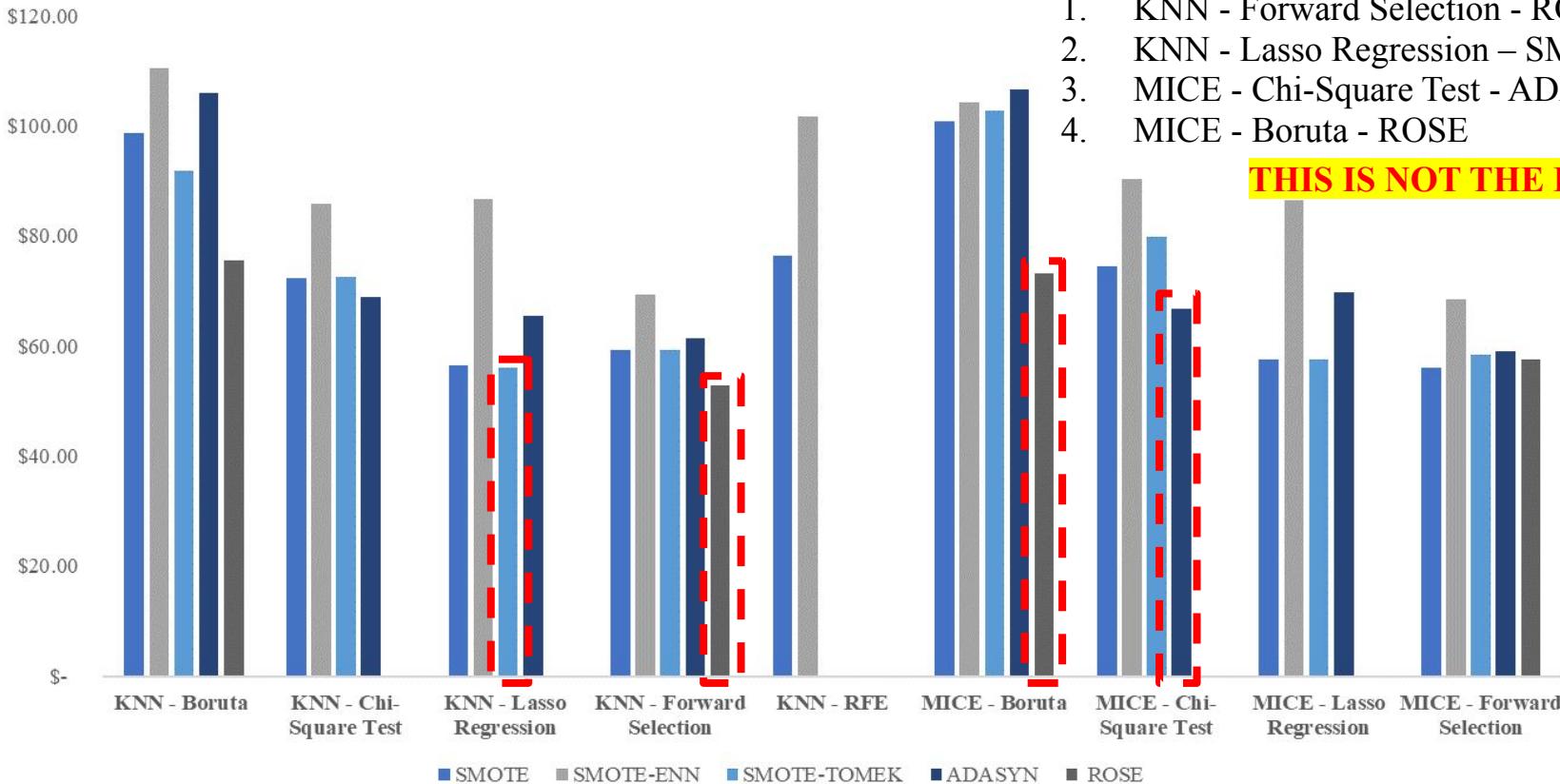
TRIAL RESULTS

Cost Model

	SMOTE	SMOTE-ENN	SMOTE-TOMEK	ADASYN	ROSE
KNN - Boruta	\$98.75	\$110.50	\$91.75	\$106.00	\$75.50
KNN - Chi-Square Test	\$72.25	\$85.75	\$72.50	\$69.00	\$0.00
KNN - Lasso Regression	\$56.50	\$86.75	\$56.00	\$65.50	\$0.00
KNN - Forward Selection	\$59.25	\$69.25	\$59.25	\$61.50	\$52.75
KNN - RFE	\$76.50	\$101.75			
MICE - Boruta	\$100.75	\$104.25	\$102.75	\$106.75	\$73.25
MICE - Chi-Square Test	\$74.50	\$90.25	\$79.75	\$66.75	\$0.00
MICE - Lasso Regression	\$57.50	\$86.50	\$57.50	\$69.75	\$0.00
MICE - Forward Selection	\$56.00	\$68.50	\$58.50	\$59.00	\$57.50

TRIAL RESULTS

Cost Model



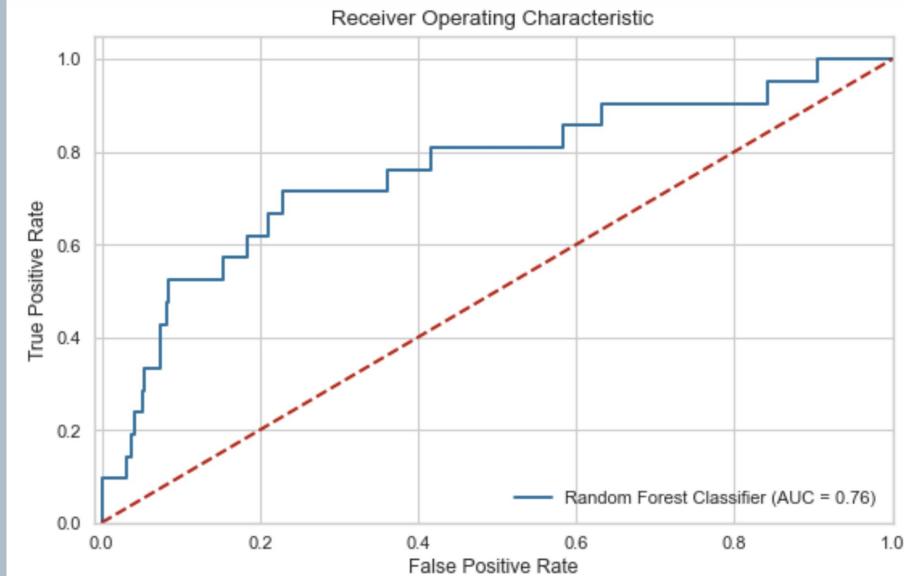
KNN + Forward Selection + ROSE

KNN - Forward Selection - ROSE

Confusion Matrix



AUC/ROC

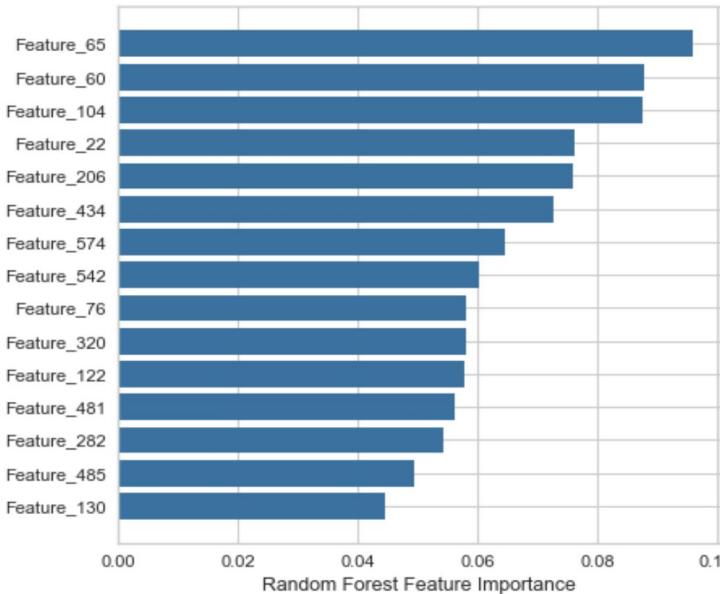


Total cost: \$ 49,500

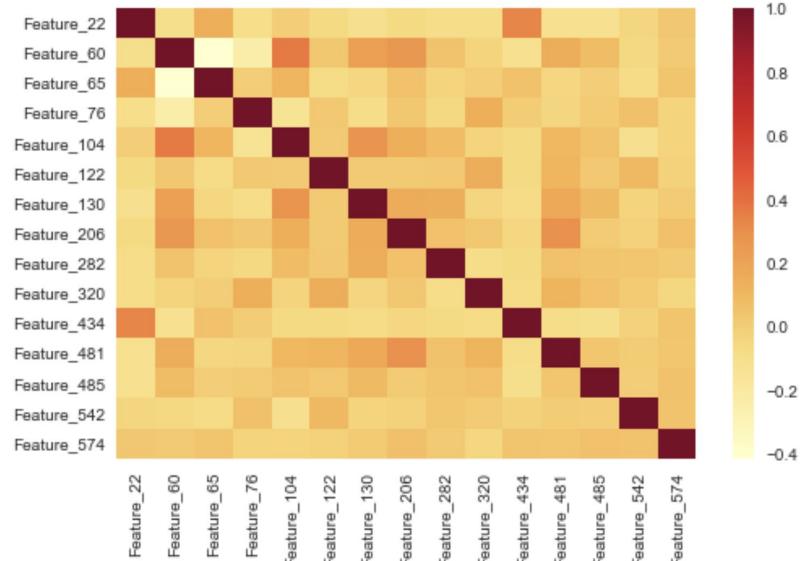
Number of remaining features: 15

KNN - Forward Selection - ROSE

Feature Importance



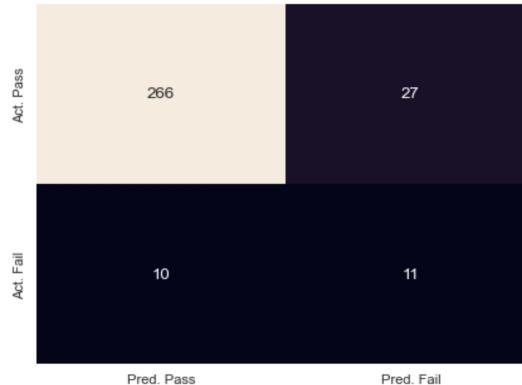
Correlation between remaining features



COMPARATIVE ANALYSIS

COMPARISON

KNN - Forward Selection - ROSE
\$49,000; 15 Features



KNN - Lasso Regression – SMOTE-TOMEK
\$56,000; 29 Features



MICE - Boruta - ROSE
\$72,500; 12 Features

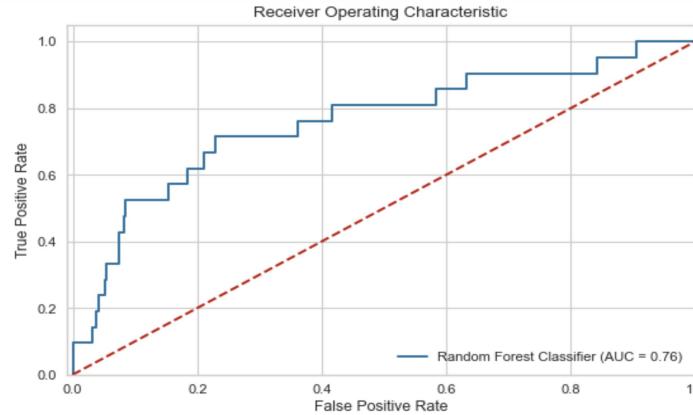


MICE - Chi-Square Test - ADASYN
\$67,000; 29 Features

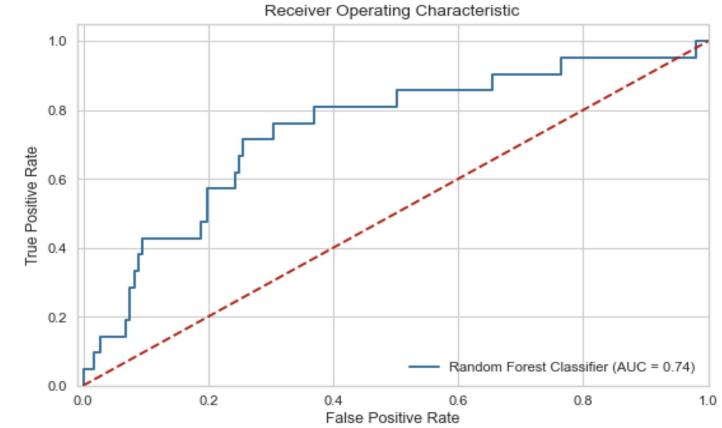


AUC/ROC

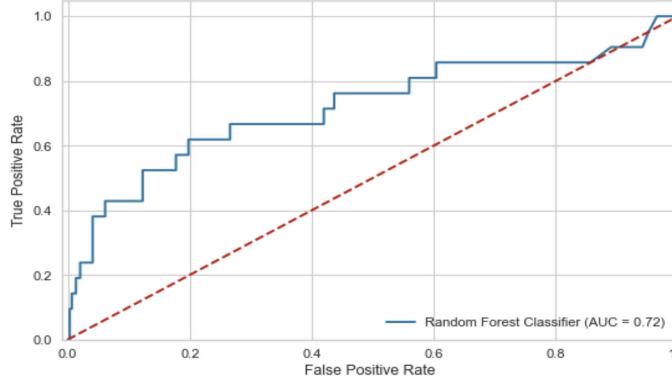
KNN - Forward Selection - ROSE



KNN - Lasso Regression – SMOTE-TOMEK

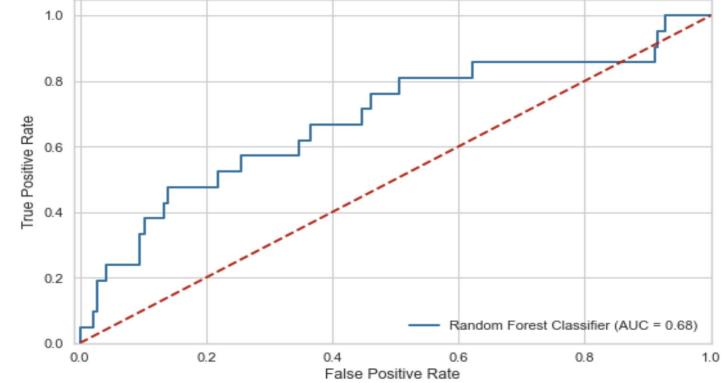


Receiver Operating Characteristic



MICE - Boruta - ROSE

Receiver Operating Characteristic

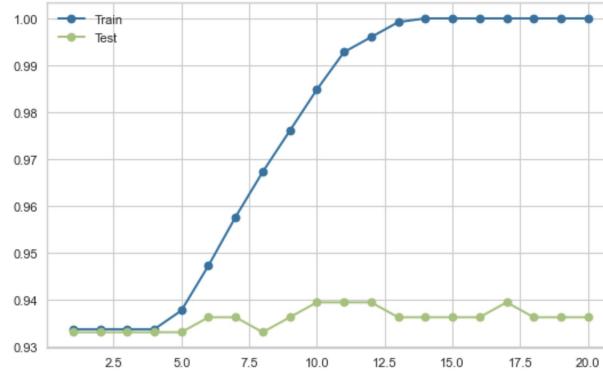


MICE - Chi-Square Test - ADASYN

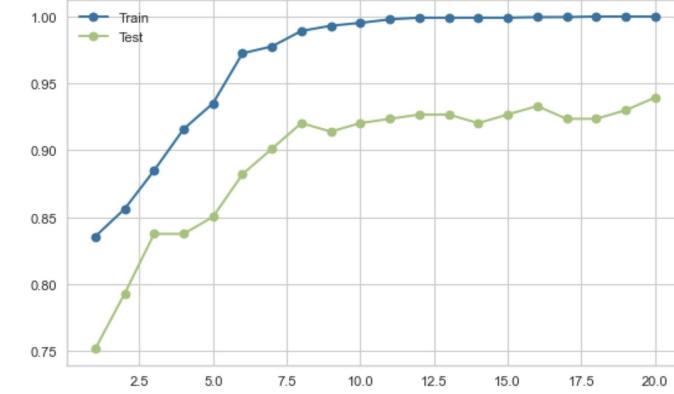
UNDERSTANDING MODELLING

OVERFITTING ANALYSIS

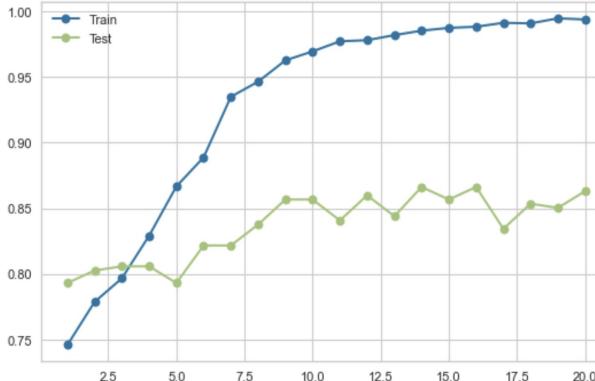
KNN - Forward Selection - ROSE



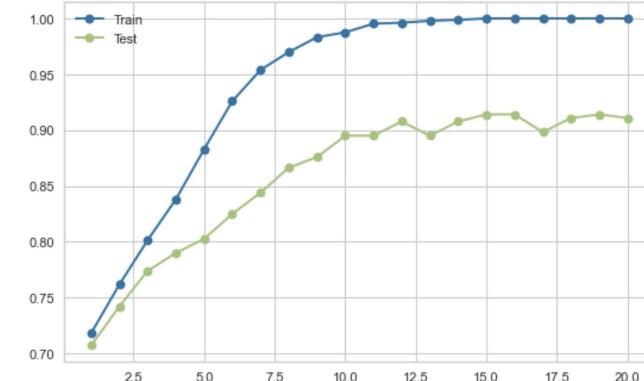
KNN - Lasso Regression – SMOTE-TOMEK



MICE - Boruta - ROSE



MICE - Chi-Square Test - ADASYN

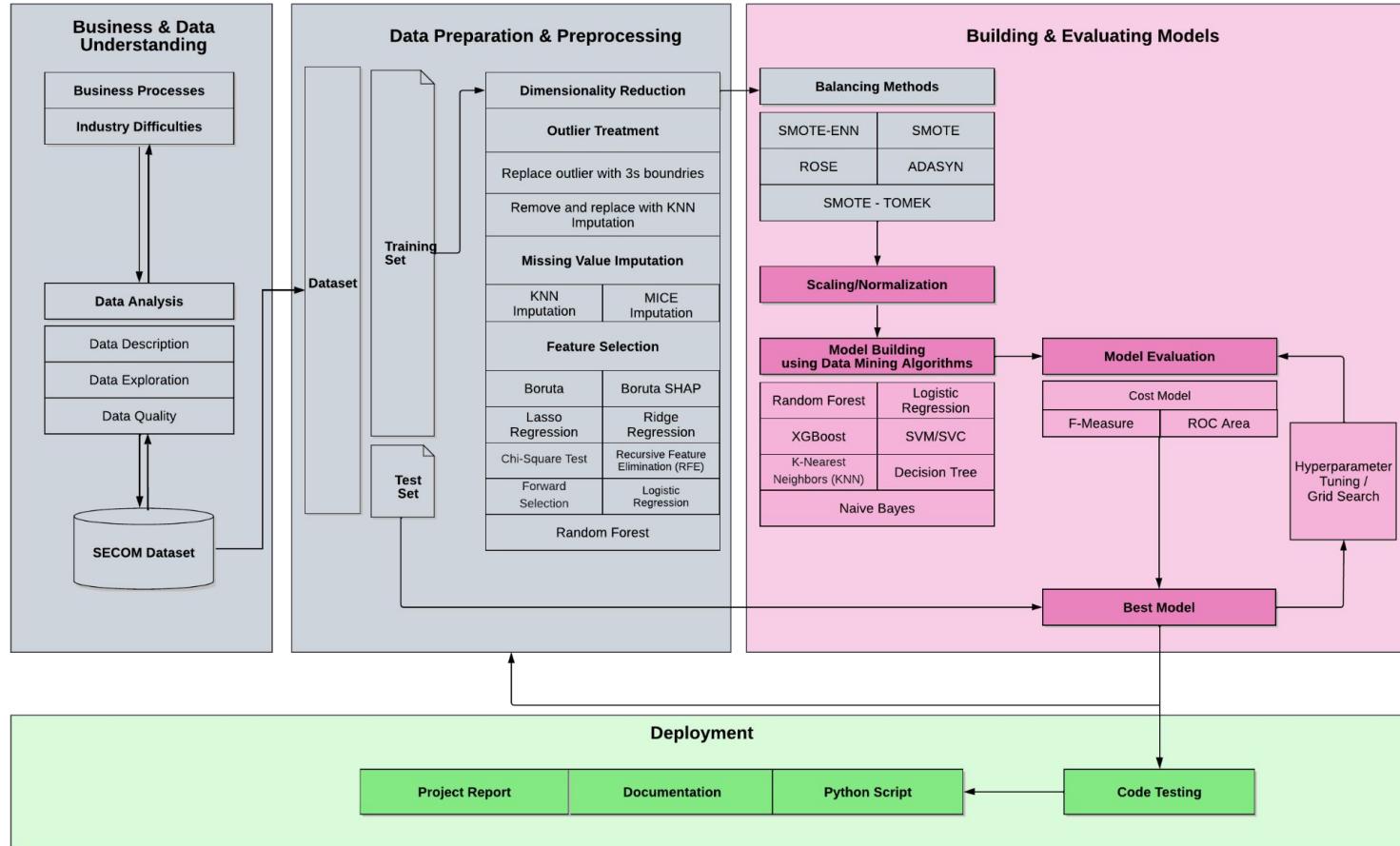


02

Next Steps

- Modeling
- Evaluation
- Deployment

Upcoming...



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

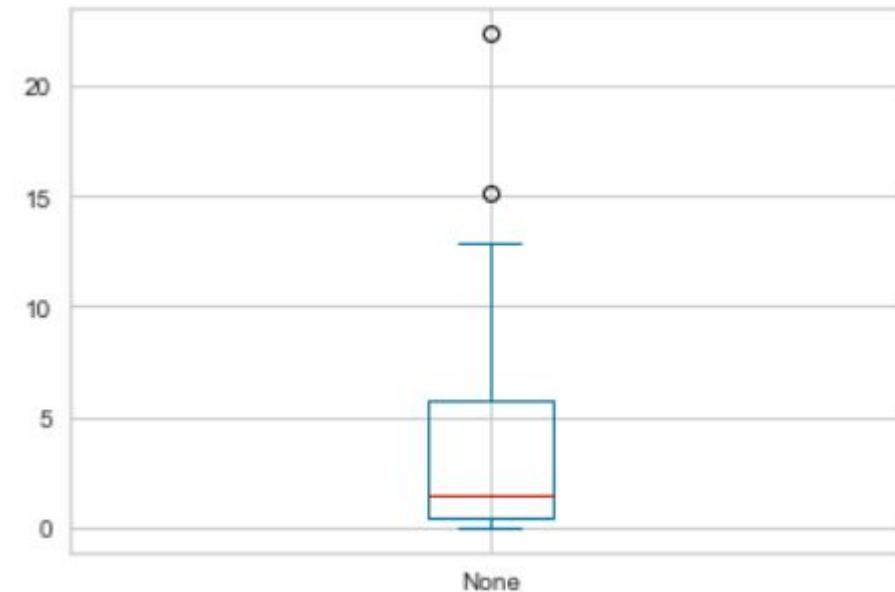
Outlier Treatment

No.	Outlier Treatment Methods	AUC	ROC	Number of features removed	Number of remaining features	Decision
1	Replace outlier & replace with 3s boundaries					
2	Remove outlier and replace with KNN imputation					
3	Remove outlier and replace with MICE imputation					

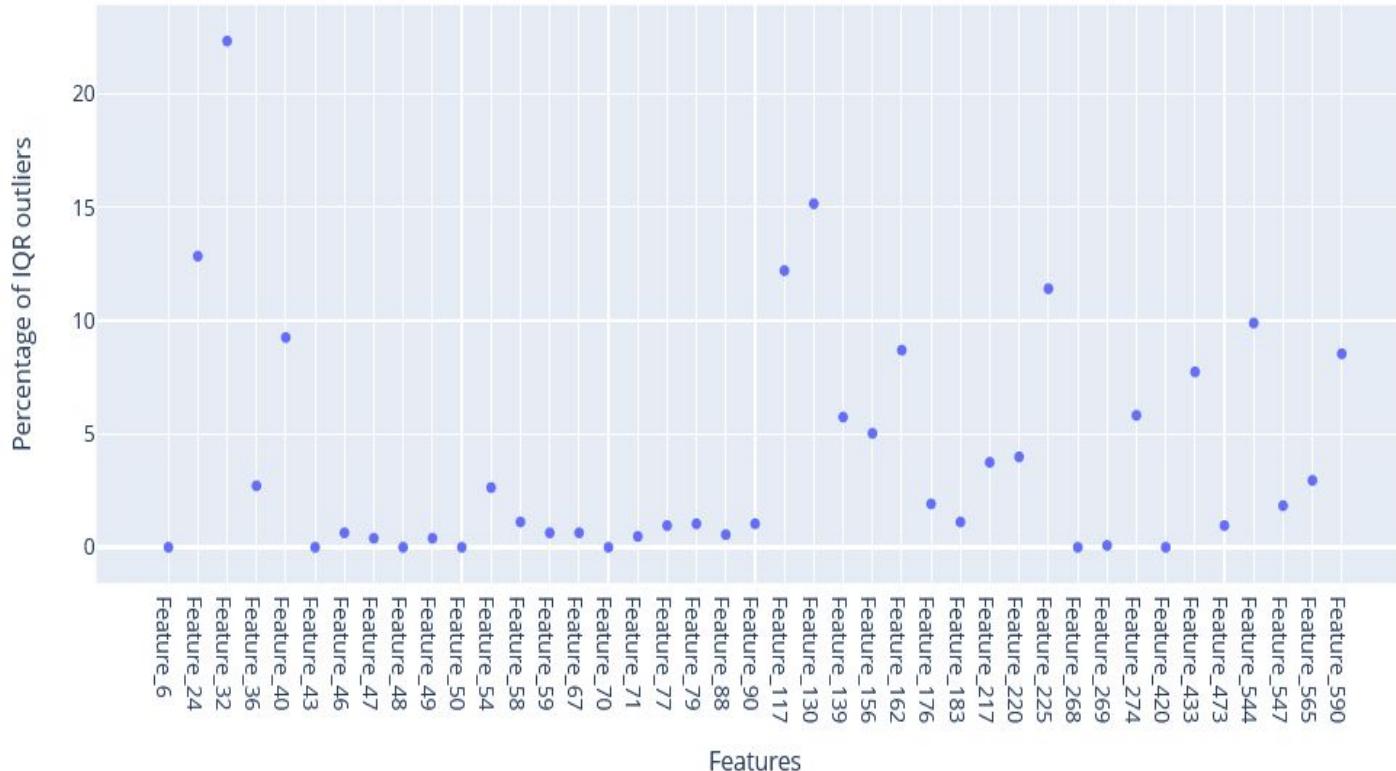
Outlier Analysis

```
def IQR_outliers(data,limit=1.5):
    numColumns = data.select_dtypes(include=np.number).columns.tolist();
    Q1 = data.quantile(0.25)
    Q3 = data.quantile(0.75)
    IQR = Q3-Q1;
    outliers=((data[numColumns] < (Q1 - limit*IQR)) | (data[numColumns] > (Q3 + limit*IQR))).sum()*100/data.shape[0]
    return outliers
```

```
fig3 = plt.figure("Figure 3")
outliers_train = IQR_outliers(X_train_cow)
outliers_train.plot(kind = 'box')
plt.show()
```



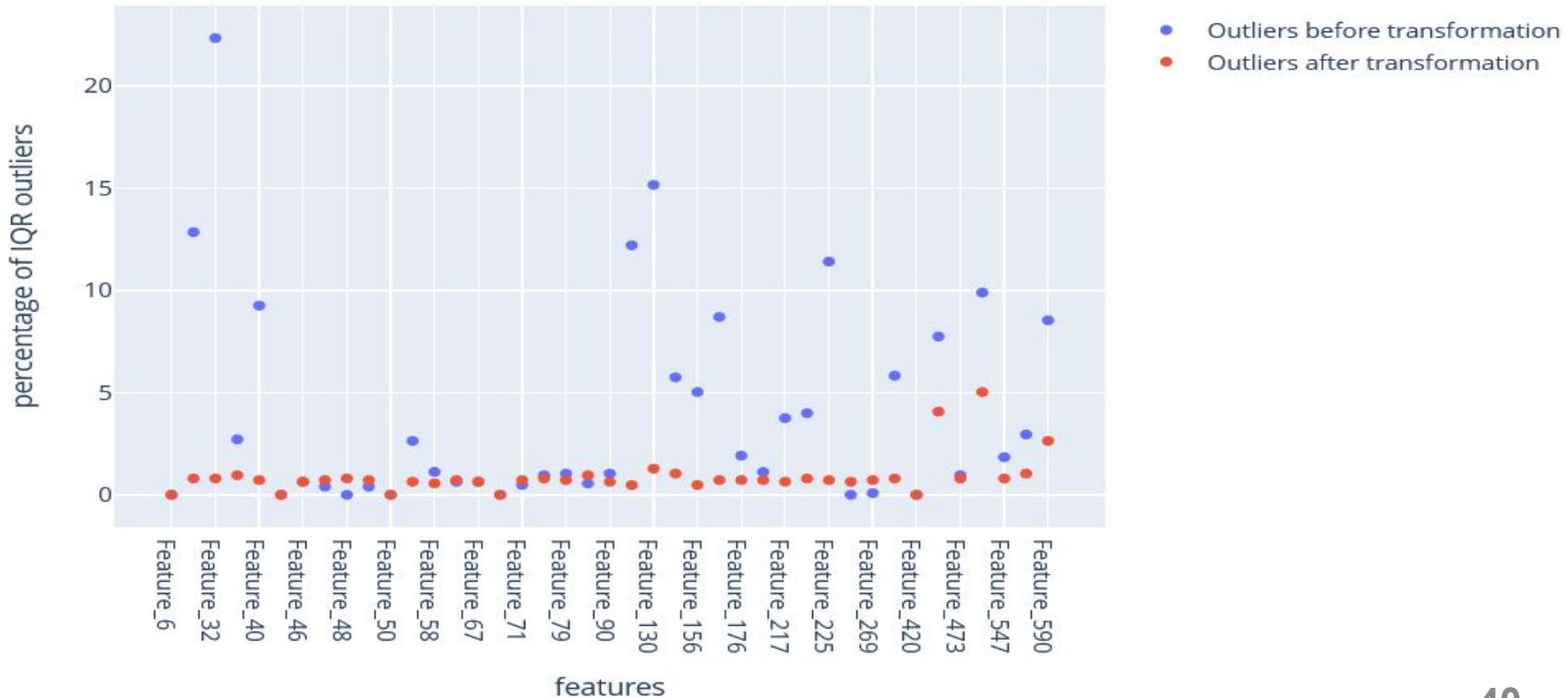
Outlier Analysis



Outlier Transformation

```
from sklearn.preprocessing import QuantileTransformer
quantile_transformer = QuantileTransformer(output_distribution='normal', random_state=42)
df1 = pd.DataFrame(quantile_transformer.fit_transform(X_train_corw),columns=X_train_corw.columns)
outliers = IQR_outliers(X_train_corw)
outliers1=IQR_outliers(df1)
```

Outlier Transformation



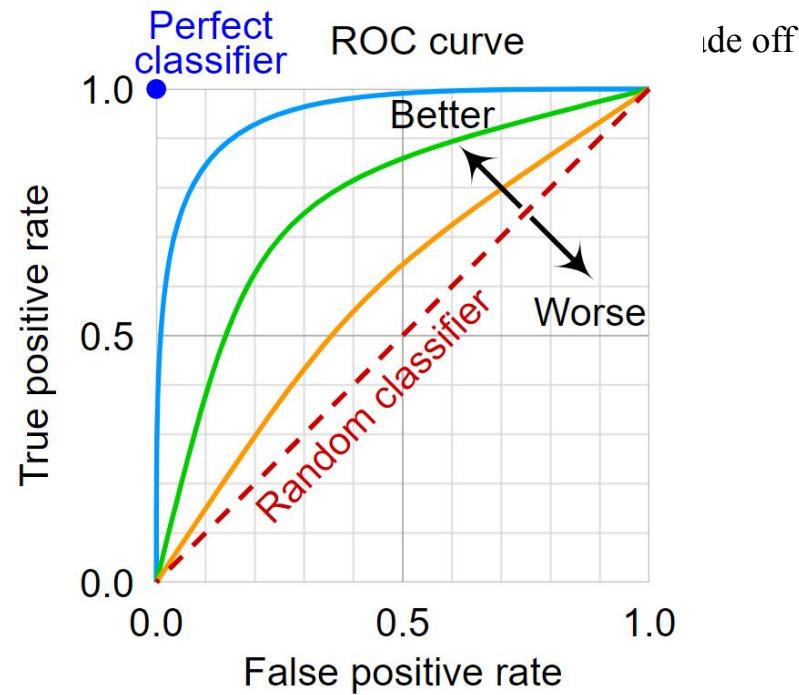
Result Measures

AUC/ ROC

$$\frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$$

F1-Score

$$\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

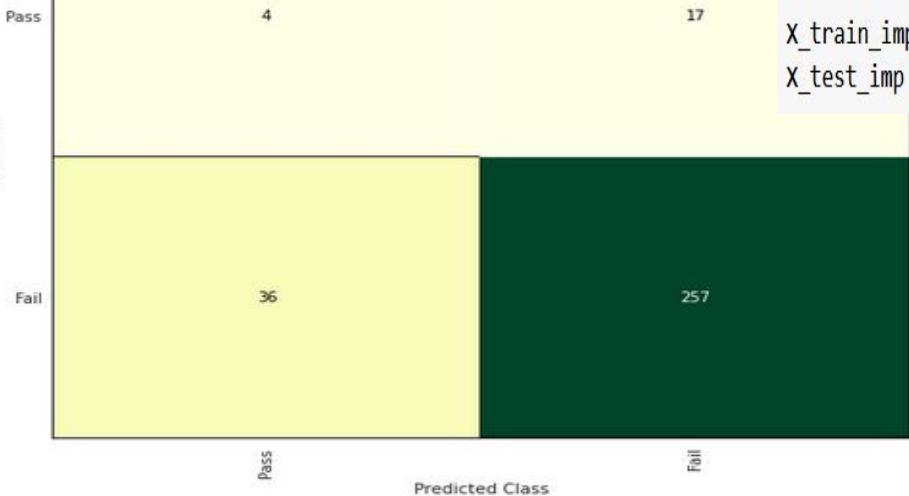


Feature Selection

2. KNN Imputer

```
imputer = KNNImputer()  
imputer.fit(X_train_na)
```

```
KNNImputer()
```

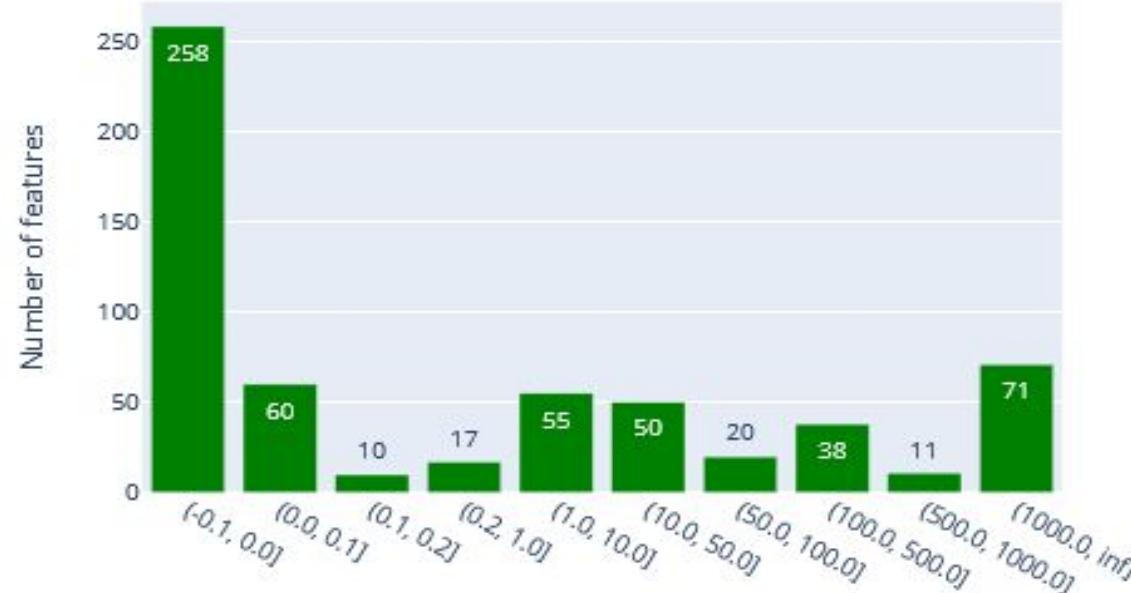


Feature Selection

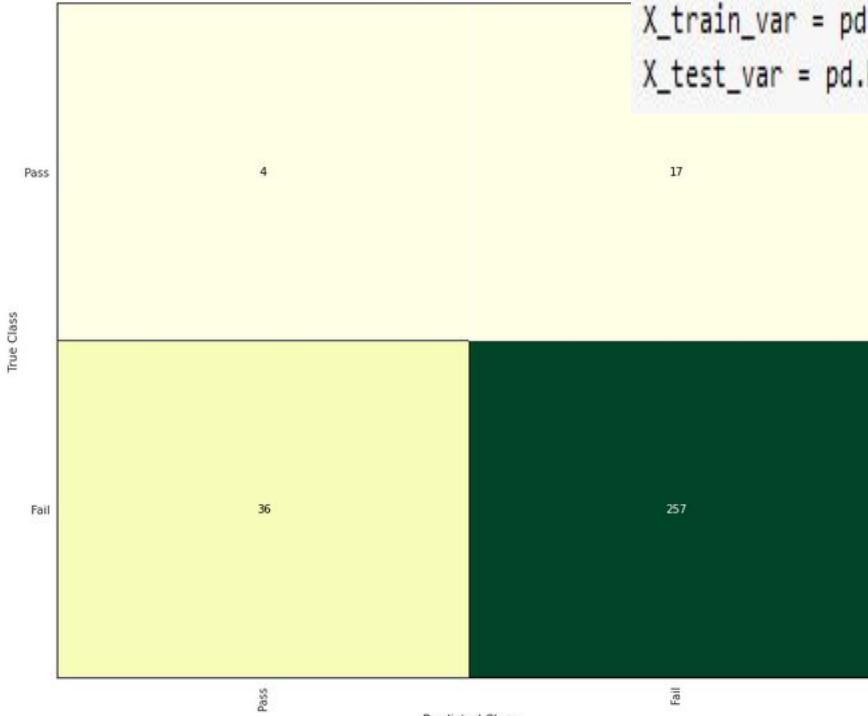
3. Normalization and Variance Threshold

```
normalizer = Normalizer()  
normalizer.fit(X_train_imp)  
  
Normalizer()
```

```
vr = VarianceThreshold()  
vr.fit(X_train_nrm)  
  
VarianceThreshold()
```



3. Variance Threshold

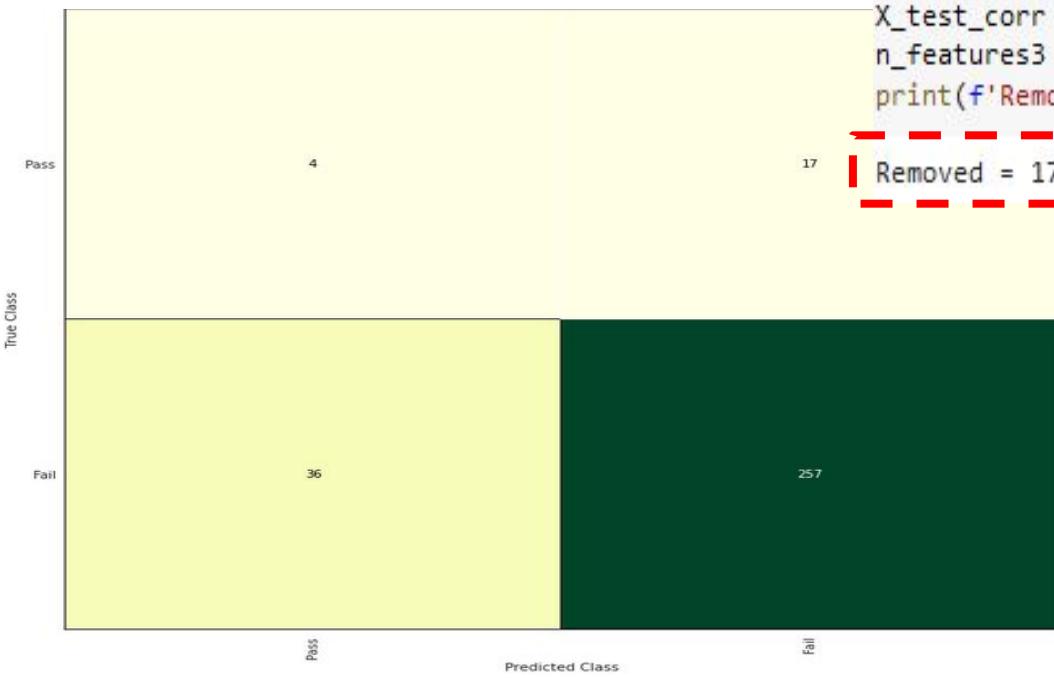


```
X_train_var = pd.DataFrame(vr.transform(X_train_imp), columns = selected_cols)  
X_test_var = pd.DataFrame(vr.transform(X_test_imp), columns = selected_cols)
```

remaining features: 446

Feature Selection

4. Correlation pairs



F1 Score: 0.8312101910828026
MCC Score: 0.05065759831945425

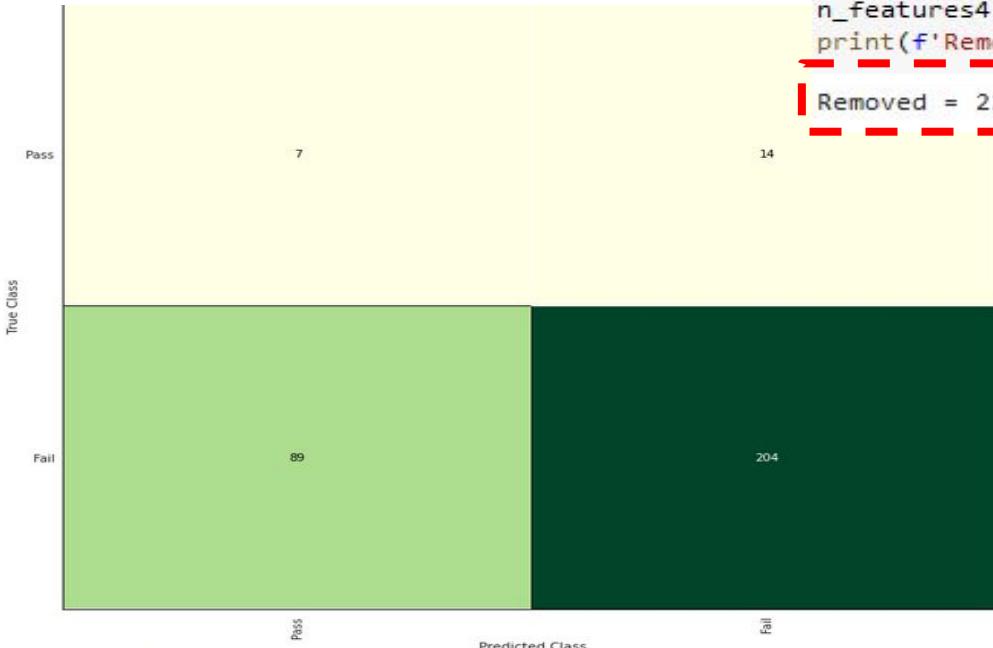
```
corr_features = correlation(X_train_var, 0.95)
X_train_corr = X_train_var.drop(corr_features, axis=1)
X_test_corr = X_test_var.drop(corr_features, axis=1)
n_features3 = X_train_corr.shape[1]
print(f'Removed = {len(corr_features)}, Left = {n_features3}')
```

17

Removed = 177, Left = 269

Feature Selection

5. Correlation with Target



```
corrwith_cols = corrwith_target(dummy_train, 'target', 0.05)
X_train_cow = X_train_corr.drop(corrwith_cols, axis=1)
X_test_cow = X_test_corr.drop(corrwith_cols, axis=1)
n_features4 = X_train_cow.shape[1]
print(f'Removed = {len(corrwith_cols)}, Left = {n_features4}')

Removed = 228, Left = 41
```

Result measurements

CONFUSION MATRIX

A confusion matrix allows visualisation of the performance of an algorithm. Each column of the Matrix represents the instances in predicted class while each row represents actual class or vice versa.

The most informative metric to evaluate a confusion metric is the Matthews correlation coefficient (MCC).

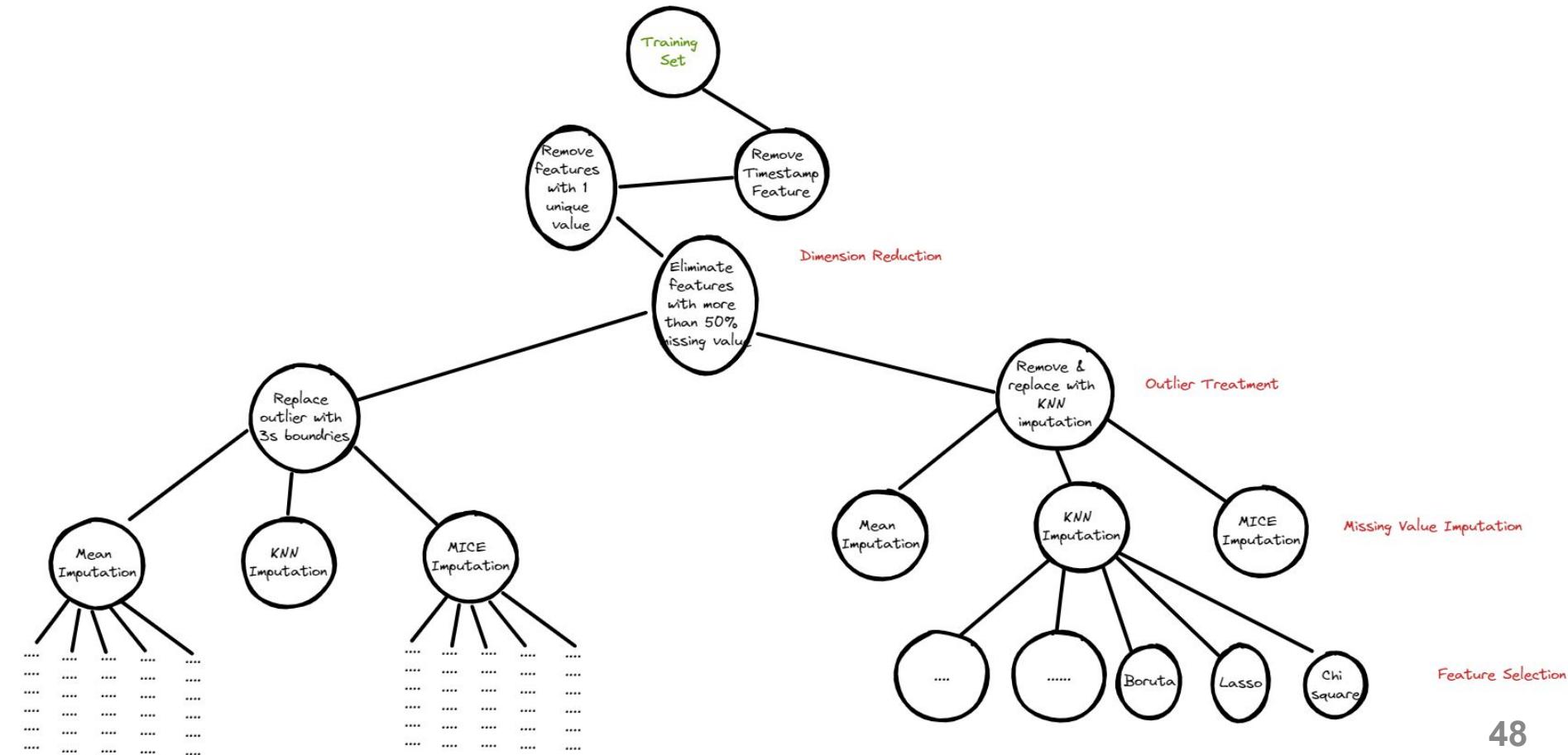
MCC is a correlation coefficient between observed and predicted binary classifications, returns a value between 1 and -1.

A coefficient of +1 represents a perfect prediction, 0 no better than random prediction and -1 indicates total disagreement between prediction and observation.

F score test's accuracy of binary classification. It does not take negative values in account

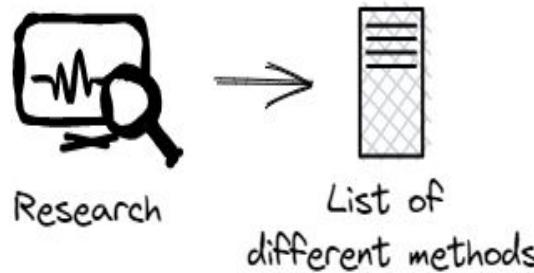


Our Approach for Data Preparation

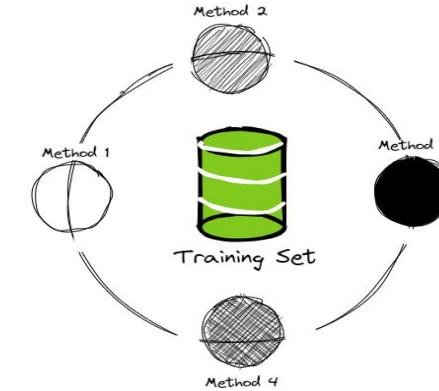


DATA PREPARATION APPROACH

01



02

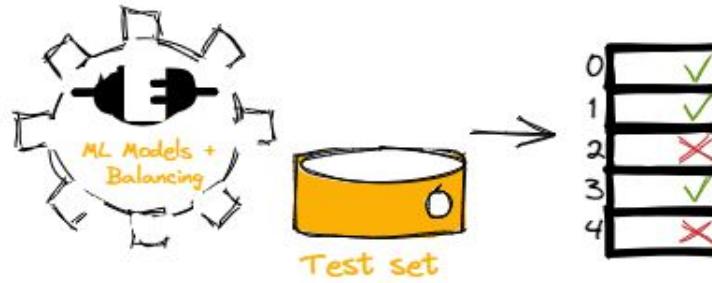


Research different methods for each step of data preparation

Use training set to train the machine with different method

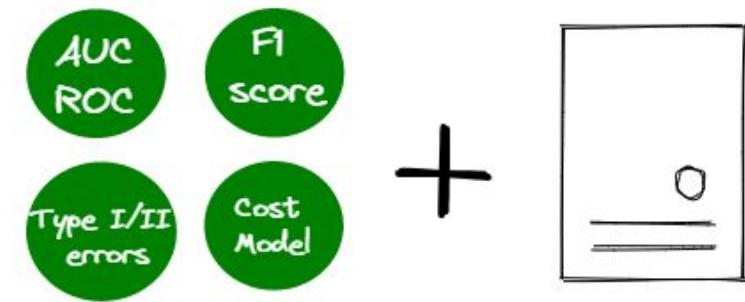
DATA PREPARATION APPROACH

03



Use different machine learning models, balancing methods & test set to test each method

04



Use AUC, ROC, type I/II errors, F1 score & cost model as measures for decision making & document our findings

01

**Lorem ipsum dolor
sit amet,
Consetetur ester
Sadipscing magna**

Lorem ipsum dolor sit amet, Consetetur ester magnat

**Lorem ipsum dolor sit amet, consectetuer
adipiscing elit et altera dominus estis.**

Fusce posuere, magna sed pulvinar ultricies, purus
lectus malesuada libero, At vero eos et accusam et
justo duo dolores et easit amet commodo magna
eros quis quod plusquam erat demonstrandum.

- Lorem ipsum dolor sit amet, consectetuer
adipiscing elit. Maecenas porttitor congue
massa.
- Lorem ipsum dolor sit amet, consectetuer
adipiscing elit. Maecenas porttitor congue
massa.

01

At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet

02

At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet

03

At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet

01

At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Et accusam et justo duo dolores et ea rebum sine fins erat demonstradum

02

At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Et accusam et justo duo dolores et ea rebum sine fins erat demonstradum

01

**Lorem ipsum dolor
sit amet, Consetetur
ester Sadipscing
magna**

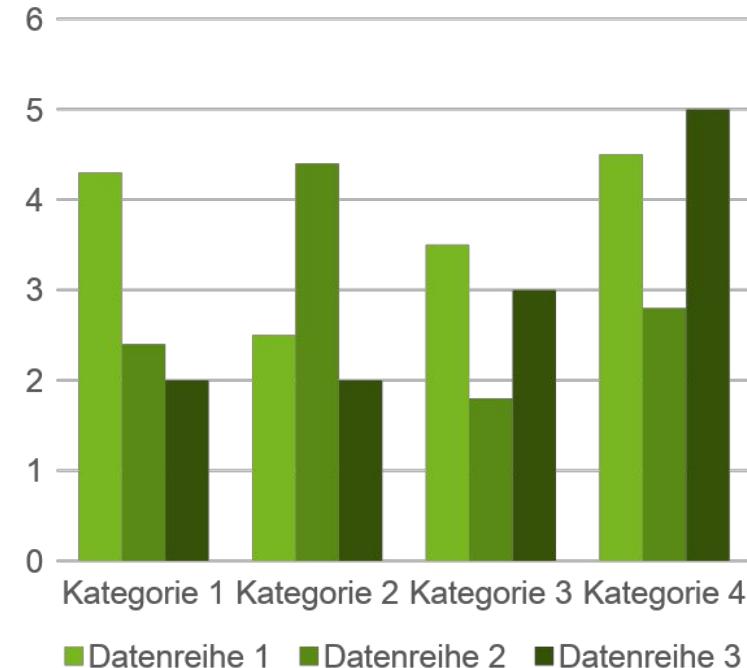


Lorem ipsum dolor sit

**Lorem ipsum dolor sit amet, consectetur
adipiscing elit et eternam etipisis.**

Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, At vero eos et accusam et justo duo dolores et easit amet commodo magna eros quis urna.

- • Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Maecenas porttitor congue massa



Lorem ipsum dolor sit

Loremm ipsum dolor sit amet, consectetuer
adipiscing elit et eternam etipisis.****

Fusce posuere, magna sed pulvinar ultricies, purus
lectus malesuada libero, At vero eos et accusam et
justo duo dolores et easit amet commodo magna
eros quis magna sed pulvinar ultricies, purus lectus
malesuada urna est.







Lorem ipsum dolor sit amet, Consetetur ester



Fusce posuere, At vero eos et accusam et
justo duo magna sed pulvinar ultricies,
purus lectus malesuada libero, At vero
eos et accusam et justo duo dolores et
easit amet commodo magna eros quis



Fusce posuere, At vero eos et accusam et
justo duo magna sed pulvinar ultricies,
purus lectus malesuada libero, At vero
eos et accusam et justo duo dolores et
easit amet commodo magna eros quis





01

**Lorem ipsum dolor sit amet, Consetetur
ester Sadipscing magna**



At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Et accusam et justo duo dolores et ea rebum sine fins erat demonstradum

Vielen Dank!



Name Referent*in
Kontakt und weitere Informationen

www.mpmd.htw-berlin.de



htw



Vielen Dank!

Name Referent*in
Kontakt und weitere Informationen

www.mpmd.htw-berlin.de



Vielen Dank!

Name Referent*in

Kontakt und weitere Informationen

Vielen Dank!



Name Referent*in

Kontakt und weitere Informationen



Name Referent*in

Kontakt und weitere Informationen



Name Referent*in

Kontakt und weitere Informationen



Name Referent*in

Kontakt und weitere Informationen



Name Referent*in

Kontakt und weitere Informationen



Name Referent*in

Kontakt und weitere Informationen

Vielen Dank!

Name Referent*in

Kontakt und weitere Informationen

Name Referent*in

Kontakt und weitere Informationen