Model To Predict The Stability of a Power Grid System

By Jessica .O. Ojadi

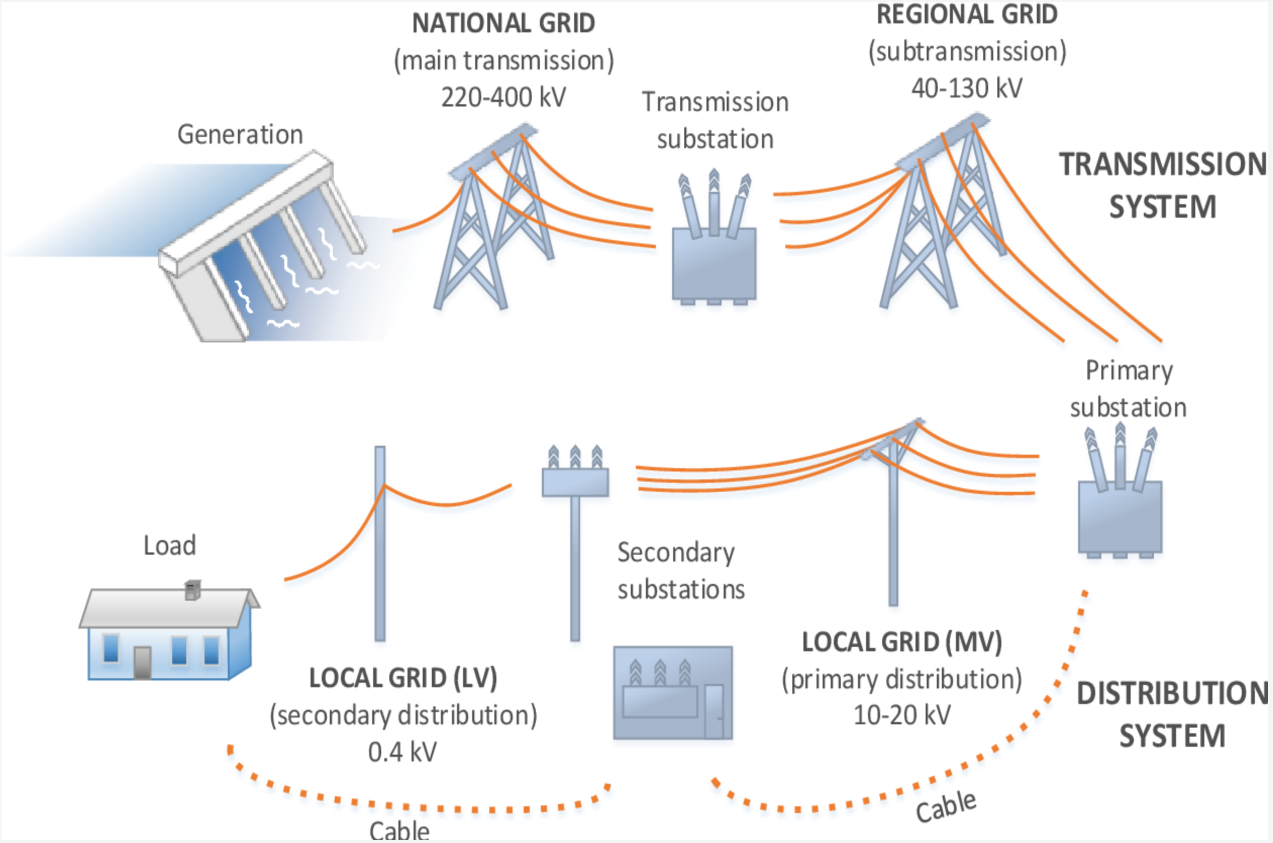
Introduction

Electrical grids require a balance between electricity supply and demand in order to be stable. Conventional systems achieve this balance through demand-driven electricity production.

For future grids with a high share of inflexible (i.e., renewable) energy source, the concept of demand response is a promising solution. This implies changes in electricity consumption in relation to electricity price changes.

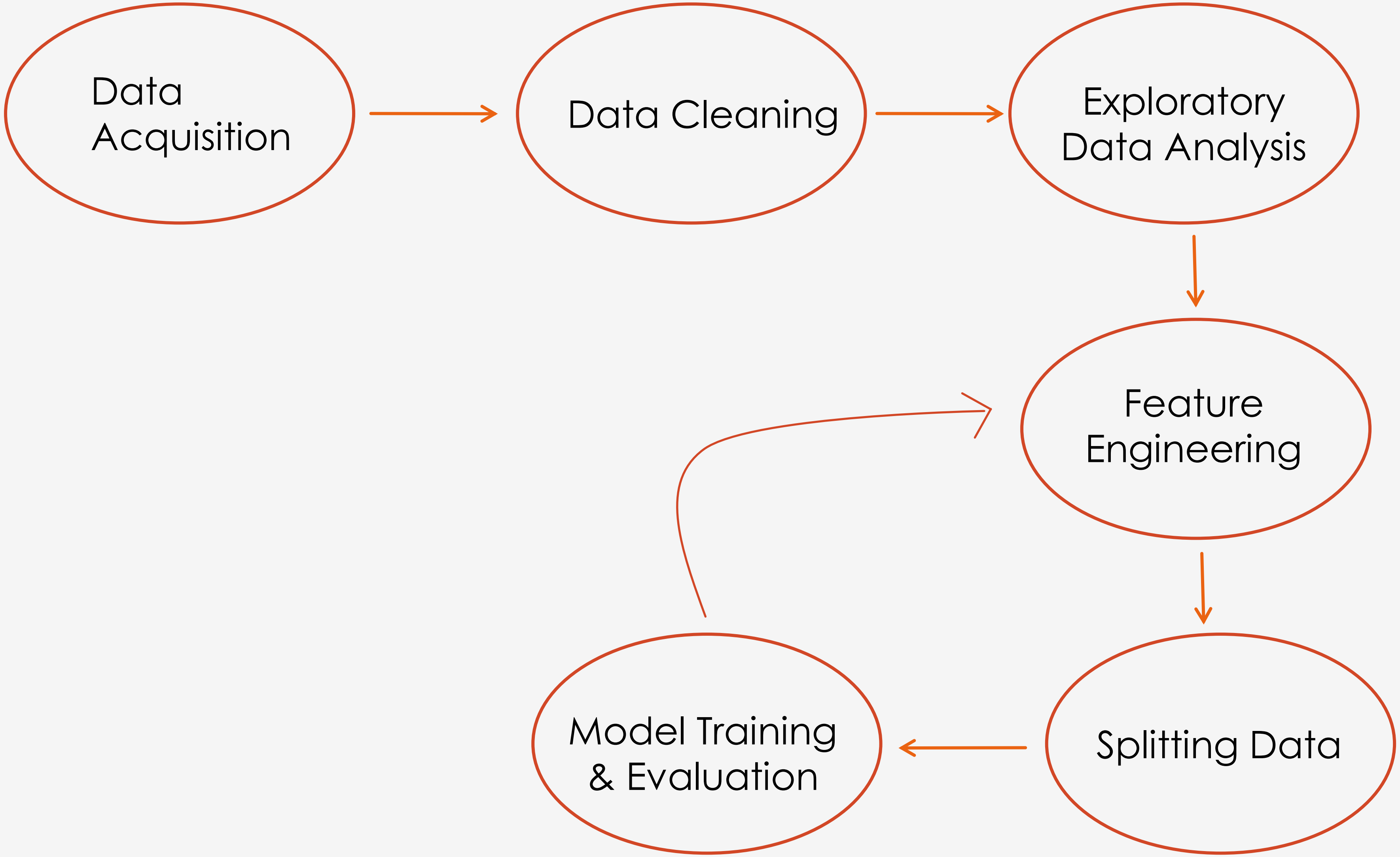
In this work, a binary classification model was built to predict if a grid is stable or unstable using the UCI Electrical Grid Stability Simulated dataset.

Different classifier models were used and the one that best suits the main objective(s) of this analysis was specified



Overview of the electric power system

Methodology



Data Acquisition

- The dataset was gootten from https://arprimary predictive features and two dependent variableschive.ics.uci.edu/ml/datasets/Electrical+Grid+Stability+Simulated+Data+. It has 12 primary predictive features and two dependent variables.

- **Predictive Features**

1. 'tau1' to 'tau4': the reaction time of each network participant, a real value within the range 0.5 to 10 ('tau1' corresponds to the supplier node, 'tau2' to 'tau4' to the consumer nodes);
2. 'p1' to 'p4': nominal power produced (positive) or consumed (negative) by each network participant, a real value within the range -2.0 to -0.5 for consumers ('p2' to 'p4'). As the total power consumed equals the total power generated, p1 (supplier node) = - (p2 + p3 + p4);
3. 'g1' to 'g4': price elasticity coefficient for each network participant, a real value within the range 0.05 to 1.00 ('g1' corresponds to the supplier node, 'g2' to 'g4' to the consumer nodes; 'g' stands for 'gamma');

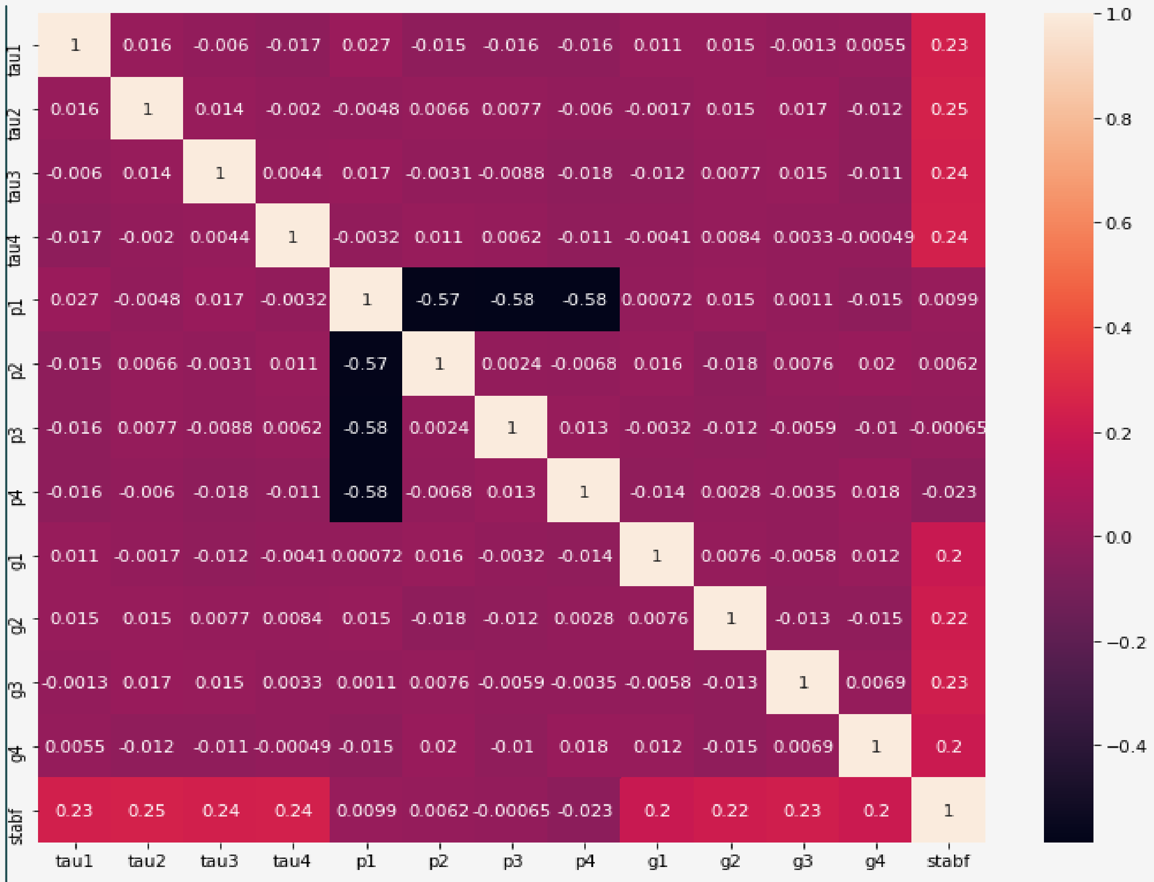
Tools and libraries used

Exploratory Analysis

1. Check for missing and duplicate data
2. Check for Outliners
3. Check Features Correlation with target Variable

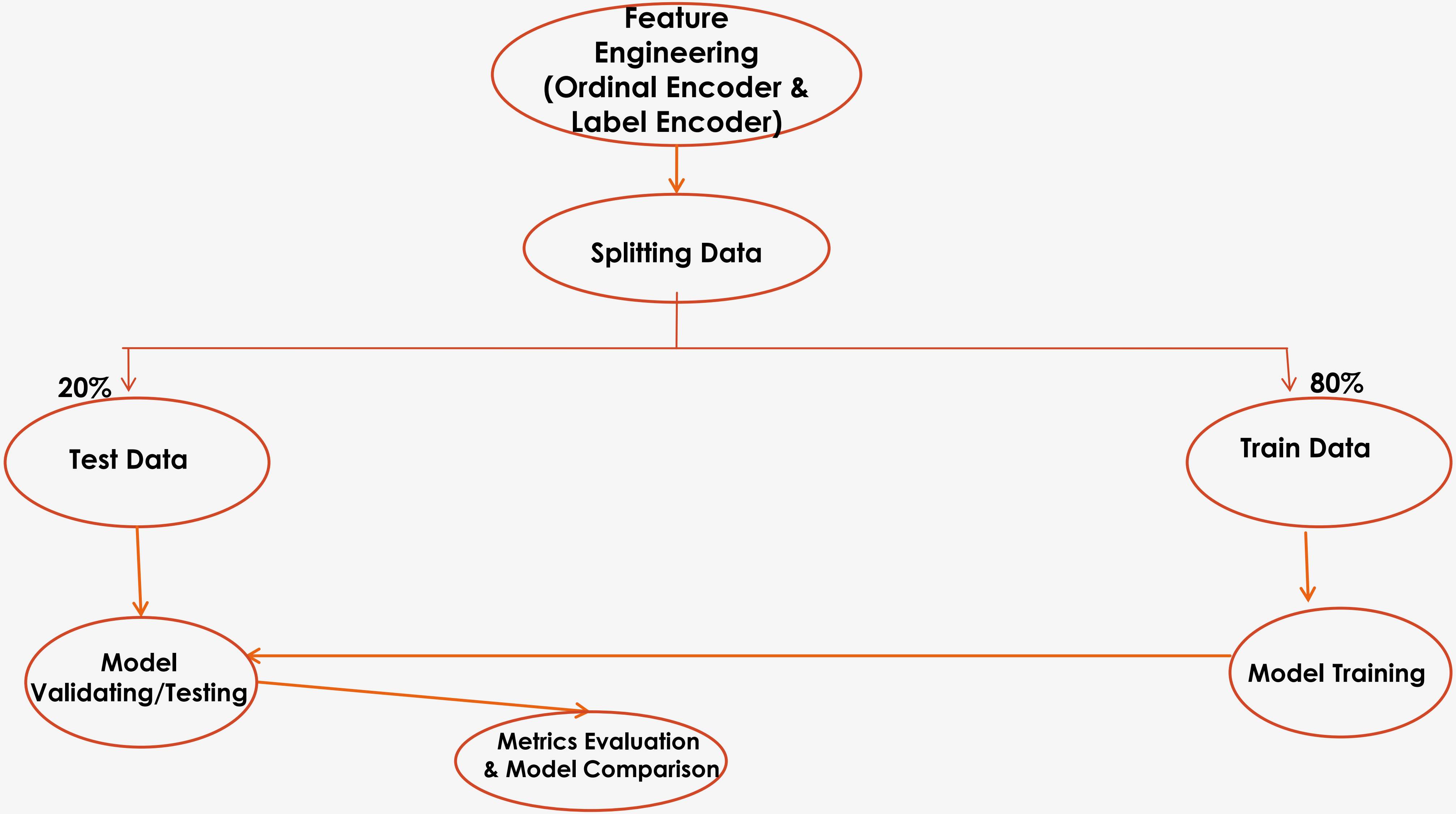
	stabf
tau1	0.234898
tau2	0.246280
tau3	0.237492
tau4	0.239375
p1	0.009938
p2	0.006173
p3	-0.000649
p4	-0.022785
g1	0.197664
g2	0.217341
g3	0.231774
g4	0.204931
stabf	1.000000

Correlation between features and target column



Correlation between data columns

FEATURE ENGINEERING, SPLITTING DATA, MODEL TRAINING & EVALUATION

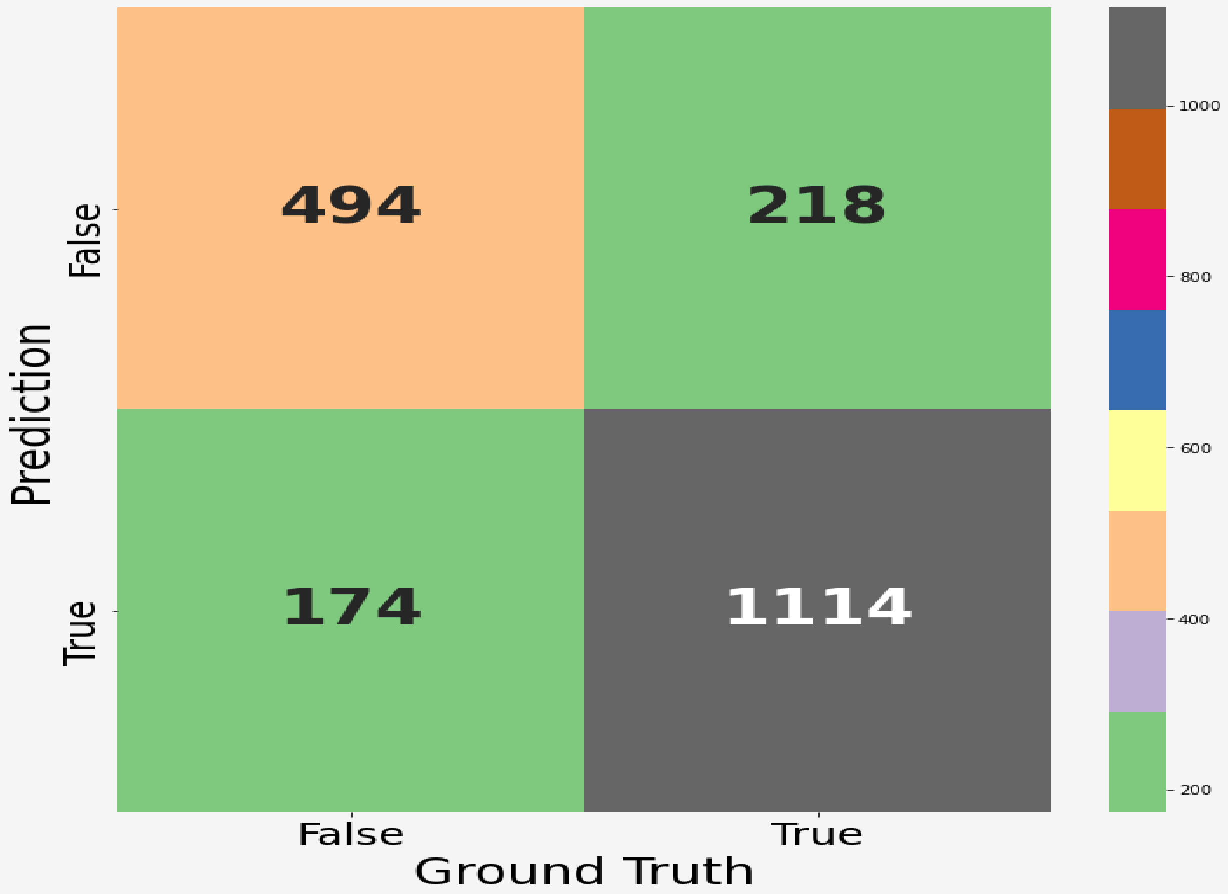


Fitting Model with Logistics Classifier

❖Accuracy, Precision, Recall and F1 score were error metrics used

	train	test
accuracy	0.820250	0.804000
precision	0.843550	0.836336
recall	0.880990	0.864907
f1	0.861864	0.850382

Error Results for Logistics Classification Model



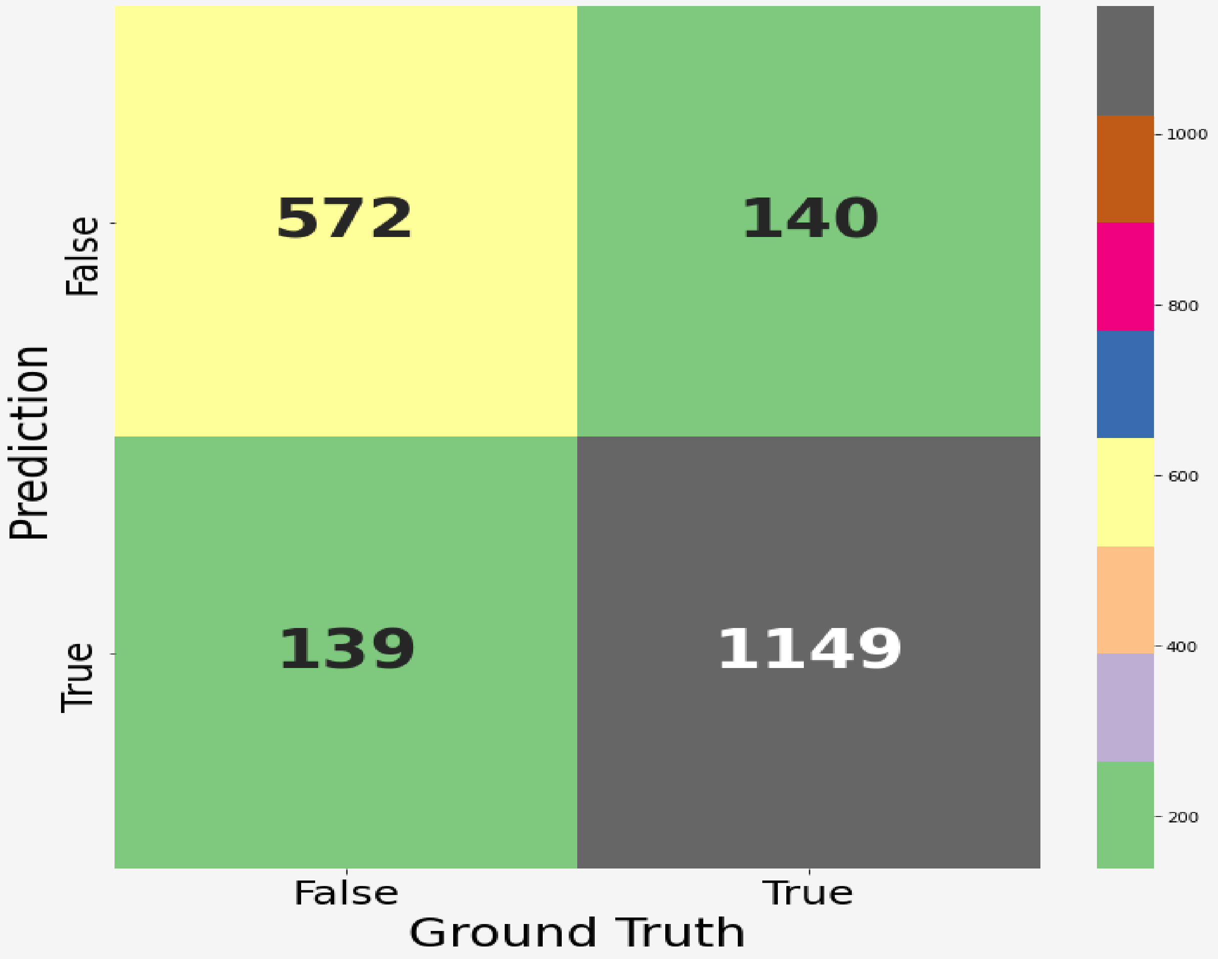
Confusion Matrix for Logistics Classification Model

Fitting Model Using Decision Tree Classifier

- Using grid search with cross validation, the tree that performs best on the test data set has tree node of 623 and tree maximum depth of 9.

	train	test
accuracy	0.953250	0.860500
precision	0.959844	0.891389
recall	0.967007	0.892081
f1	0.963412	0.891735

Error Results for Decision Tree



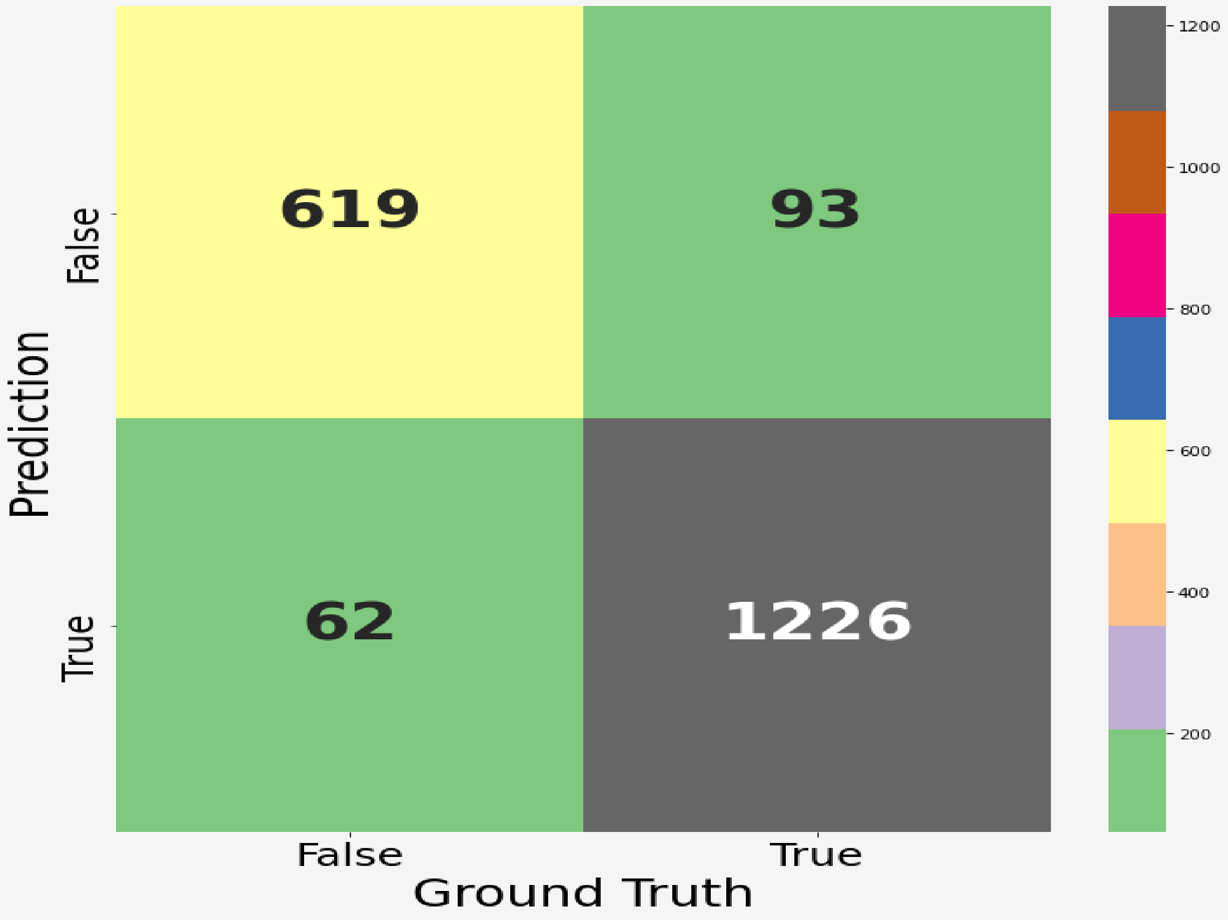
Confusion Matrix for Decision Tree Model

Fitting Model Using Random Forest Classifier

- Using grid search with cross validation, RandomForestClassifier with n_estimators =500, random_state=1 was the best estimator for the model

	train	test
accuracy	1.0	0.922500
precision	1.0	0.929492
recall	1.0	0.951863
f1	1.0	0.940545

Error Results for Random Forest Model



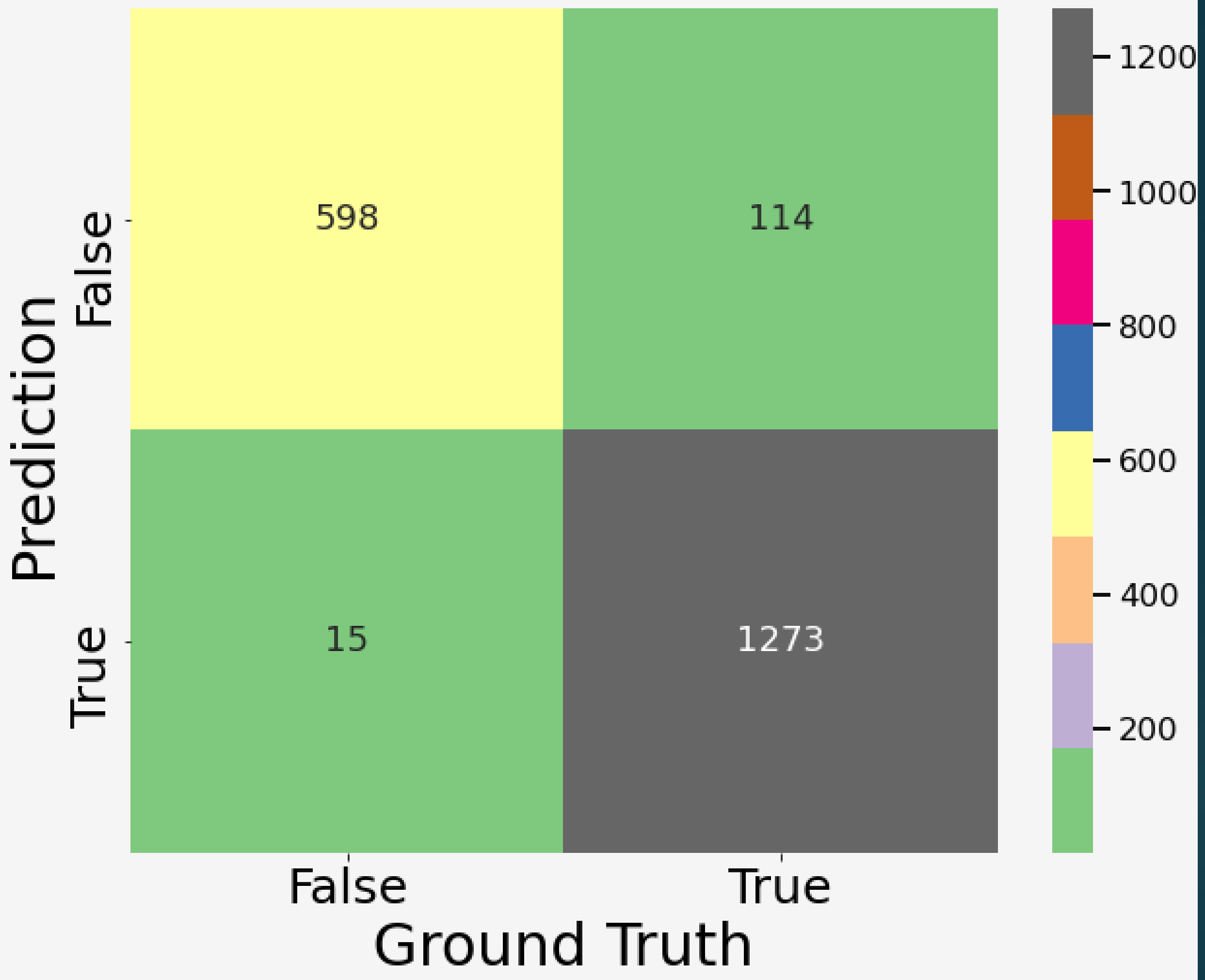
Confusion Matrix for Random Forest Model

Fitting Model using ExtraTree Classifier

- Using grid search with cross validation, ExtraTreesClassifier with minimum samples split=5, n_estimators =1000 and random_state=1, was found to be the best estimator for the model

	train	test
accuracy	1.0	0.935500
precision	1.0	0.917808
recall	1.0	0.988354
f1	1.0	0.951776

Error Results for ExtraTree Classifier Model



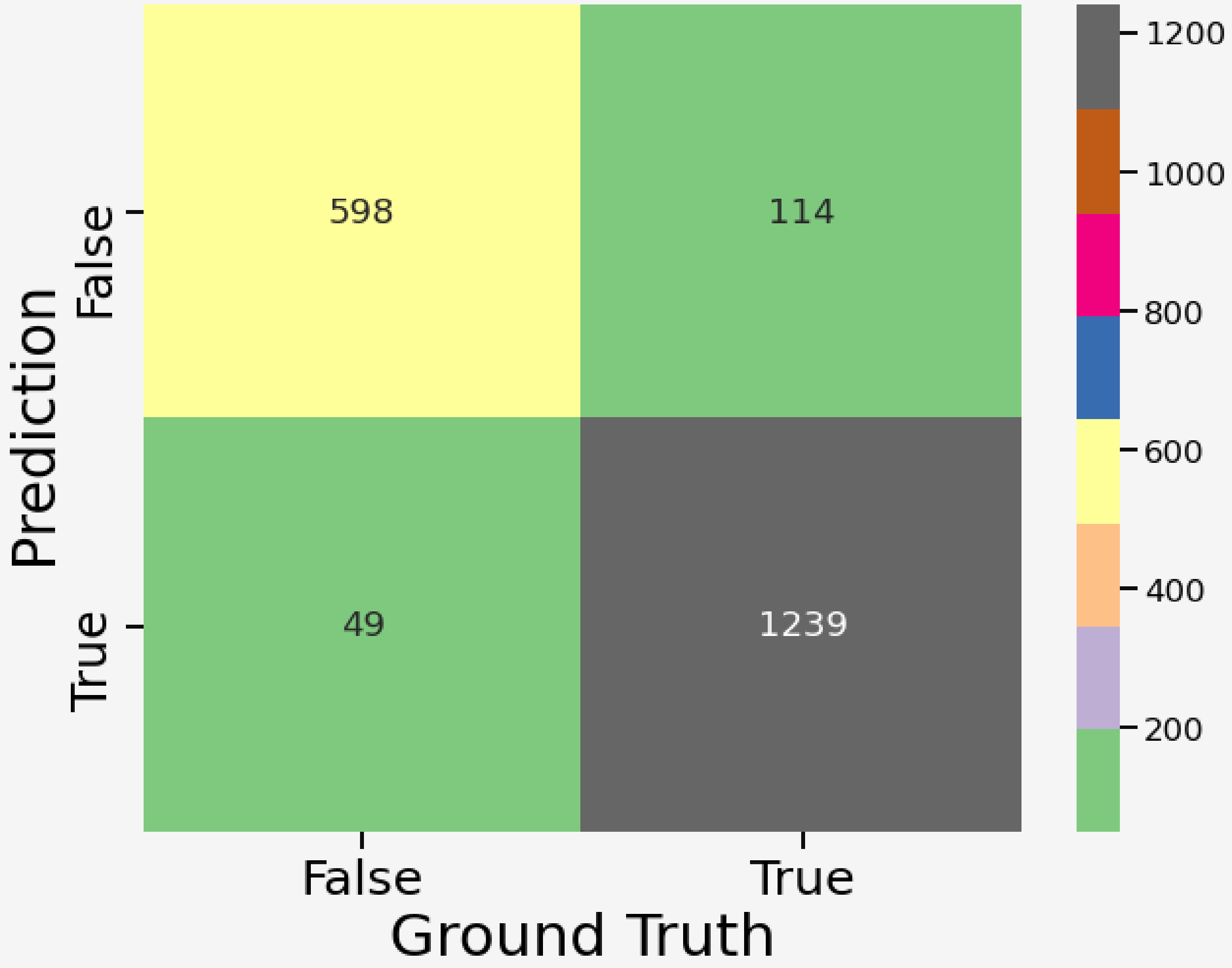
Confusion Matrix for ExtraTree Classifier Model Model

Fitting Model using Adaboost Classifier

- Using grid search with cross validation, Adaboost Classifier with base_estimator = DecisionTreeClassifier (max_depth=9), learning_rate = 0.01, n_estimators =1000, was found to be the best estimator for the model

	train	test
accuracy	1.0	0.918500
precision	1.0	0.915743
recall	1.0	0.961957
f1	1.0	0.938281

Error Results for Adaboost Classifier Model



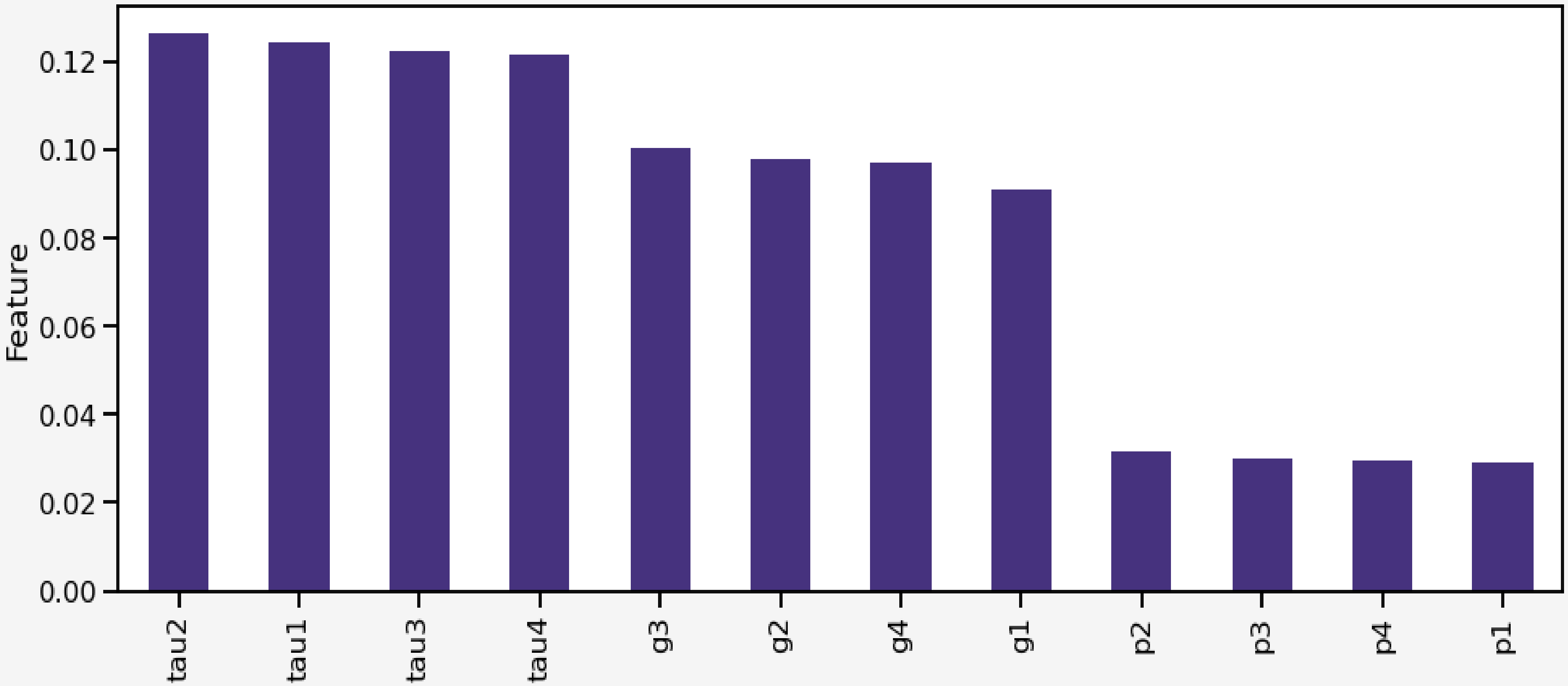
Confusion Matrix for Adaboost Classifier Model Model

Result Summary

	Error_matrix	Logistics Classifier	DecisionTree Classifier	RandomForest Classifier	ExtraTree Classifier	Adaboost Classifier
0	accuracy	0.8040	0.8605	0.9225	0.9355	0.9185
1	precision	0.8363	0.8914	0.9295	0.9178	0.9157
2	recall	0.8649	0.8921	0.9519	0.9884	0.9620
3	f1 score	0.8504	0.8917	0.9405	0.9518	0.9383

From error table and confusion matrix, ExtraTrees Classifier with minimum samples split=5, n_estimators =1000 and random state=1 is the best fitting model because it has the best error scores and predicted the most positive outcome in the confusion Matrix.

Feature Importance



tau2 is the most important feature in the model.

Conclusion

The scope of this project is to apply machine learning concepts taught in class on our data set. An open dataset was used. I was able to apply several classification machine learning methods on the electricity data set to achieve the goal of the project.
From error table and confusion matrix, ExtraTrees Classifier with minimum samples split=5, n_estimators =1000 and random state=1 is the best fitting model because it has the best error scores and predicted the most positive outcome in the confusion Matrix.

From the feature Importance chart, it can be seen that reaction time of network participants contributes the most to determining if a grid is stable or not.

Further work will be to apply more classification methods on the dataset and also to expand the scope of the methods used here.