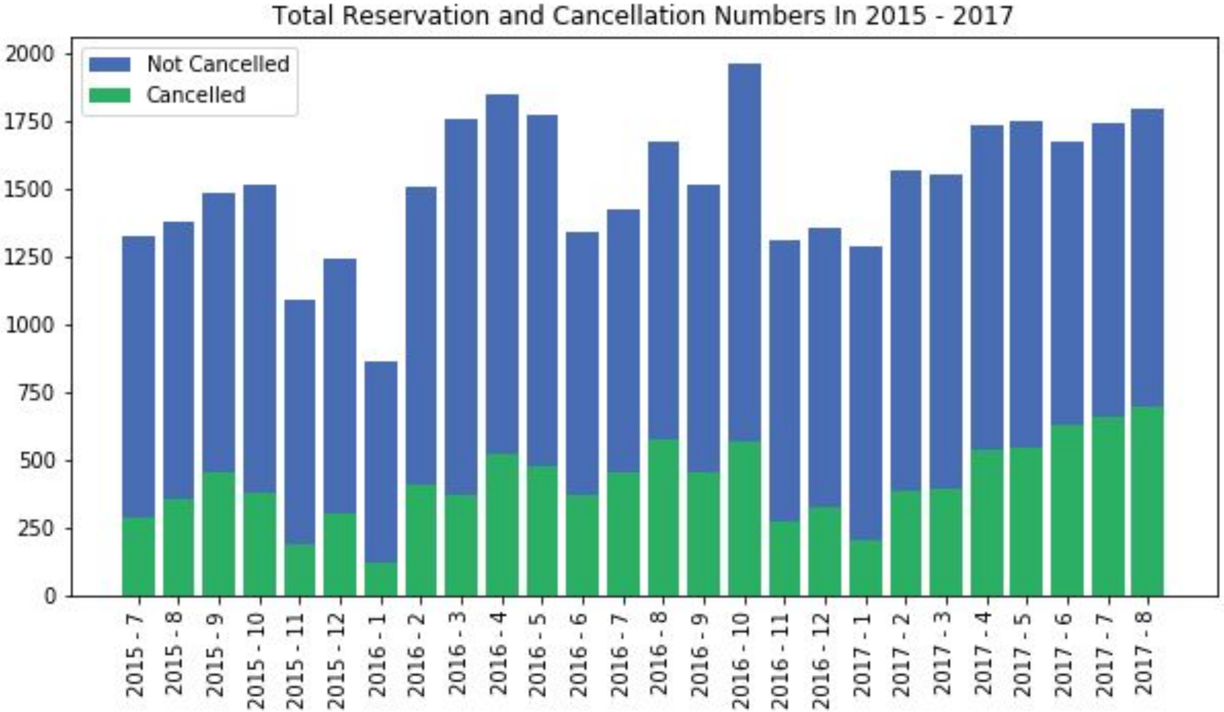
Hotel Reservation Cancellation Analysis

Is it possible to identify customers who will cancel their reservations?

Business Problem

- Reservations at hotel are frequently cancelled
- Average in the examined time period: **27.67%**
- Can go up to **35%** in certain months
- Reservations are made **93 days** in advance on average



Data

Source

- Hotel booking demand datasets article by Nuno Antonio, Ana de Almeida, Luis Nunes on ScienceDirect
- Collections of guest list from two hotels in Portugal from 2015 - 2017
- We decided to use H1, the resort hotel, about 40,000 observations

Challenges

- Data is anonymised
- Queried from hotel's SQL database containing dynamic updates of records
- Many categorical variables
- Cost function tuning - decided to go with relative cost of false negative 20% higher

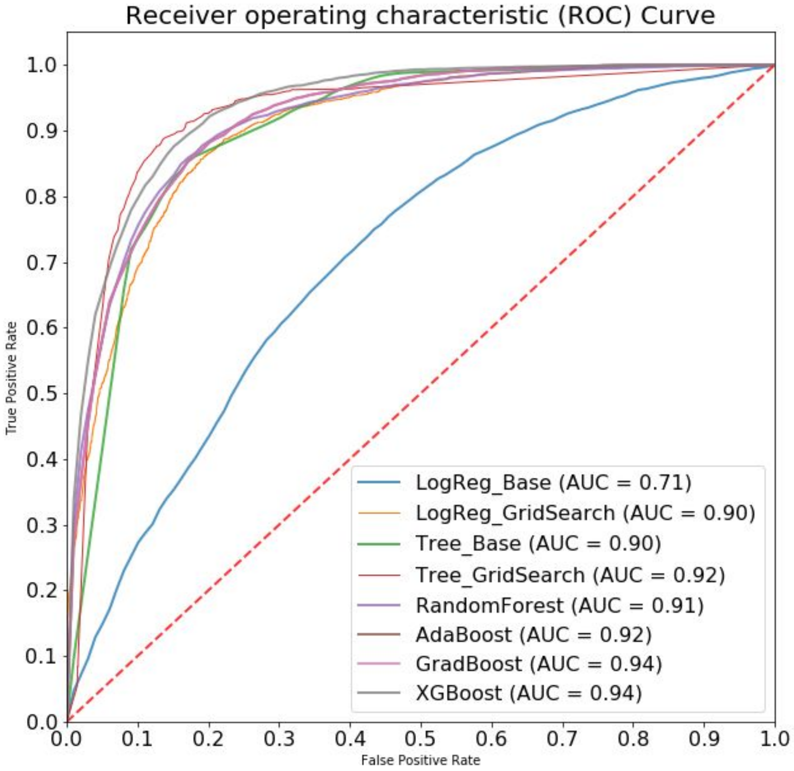
Modelling Approaches & Metrics

- Models we considered: Logistic Regression, Decision Tree, Random Forest, Gradient Boosting Techniques, K-Fold Method

- Methodology: 5-Fold Cross-Validation

- Metric: We considered the usual metrics like accuracy, AUC, etc, but the final decision was made based on he Zweig - Campbell score: True Positive Rate - m * False Positive Rate

- False Negative is assumed to be 20% worse than False Positive, m = 0.83,



Final Model - Performance

Random Forest Classifier with the following parameters:

- criterion: 'entropy'
- n_estimators: 100
- min_samples_leaf: 10
- max_features: 20

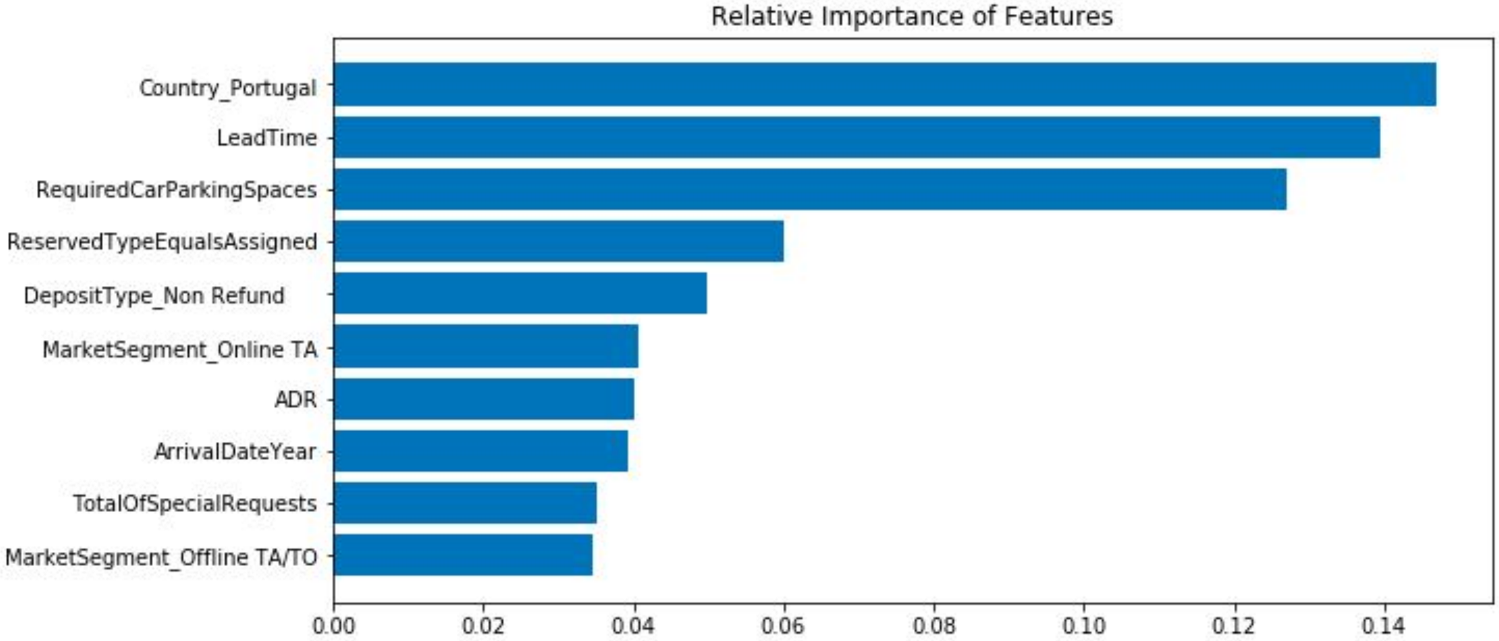
Performance:

- Training Data AUC: **97.8%**
- Validation Data AUC: **96.2%**
- Test Data AUC: **95.6%**

Confusion Matrix:

		predicted	
		not cancelled	cancelled
actual	not cancelled	**1,262**	**213**
	cancelled	**118**	**1,407**

Final Model - Feature Importances into Actionable Insights

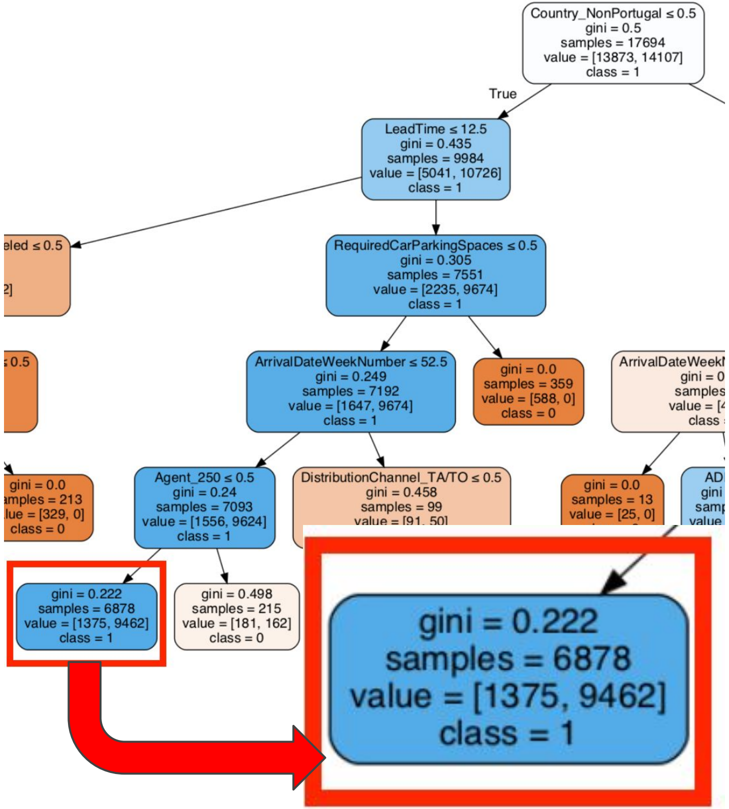


Final Model - Important Feature Deciders

Of those customers who:

1. Reside outside of Portugal
2. Made a booking over 12 days in advance
3. Did not request a parking space
4. Did not book via the specific travel agent (Agent_250)

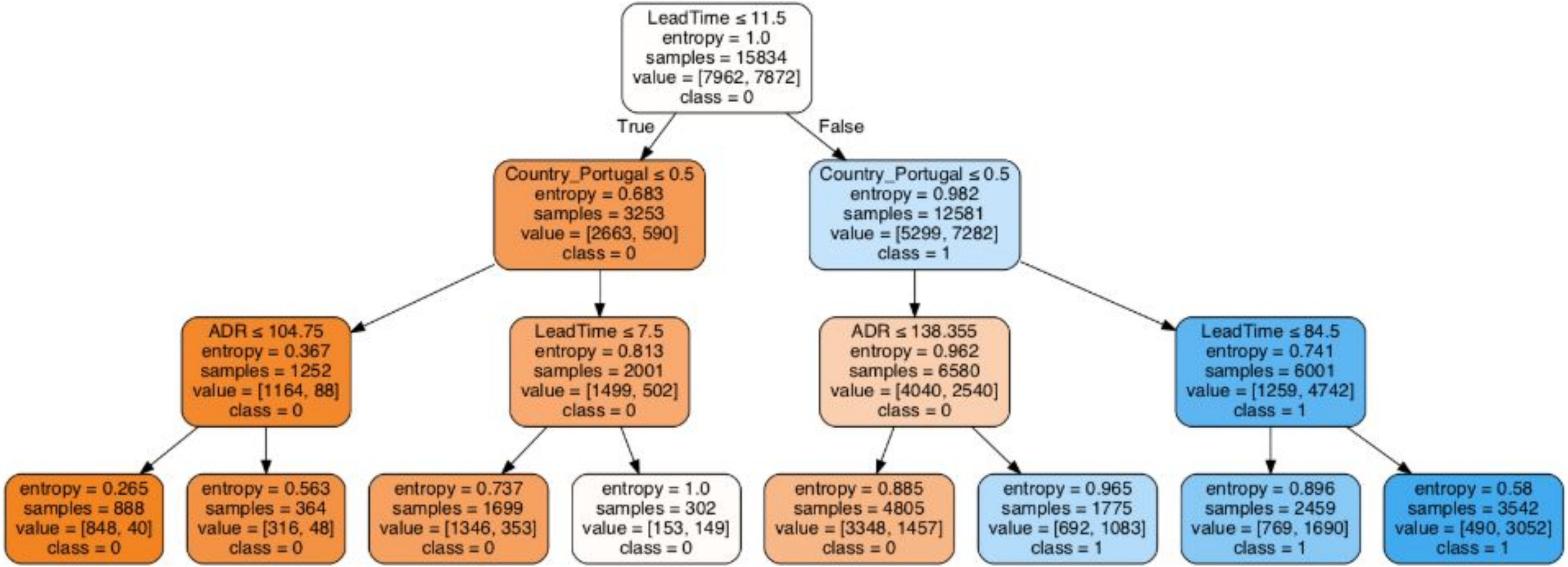
80% of these customers ended up cancelling and this subset of 'canceling customers' accounted for approx 67% of all 'canceling customers'



Thank you!

Questions?

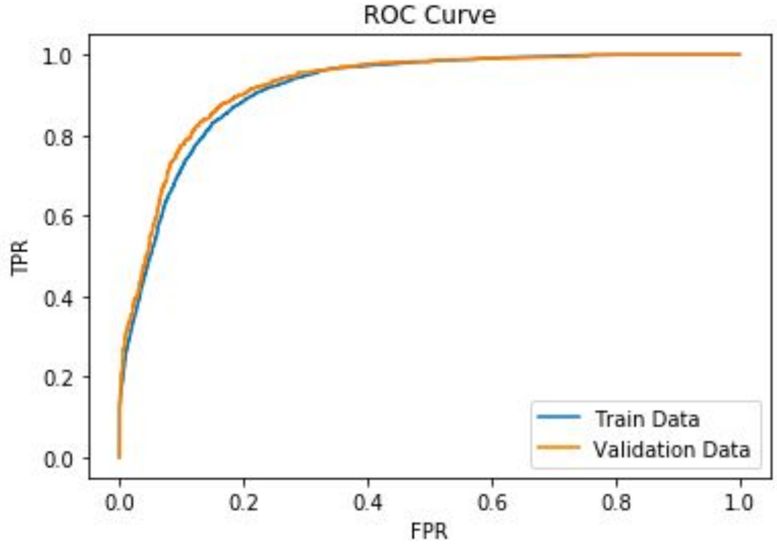
Appendix - Example of a Decision Tree



Appendix - Logistic Regression Performance

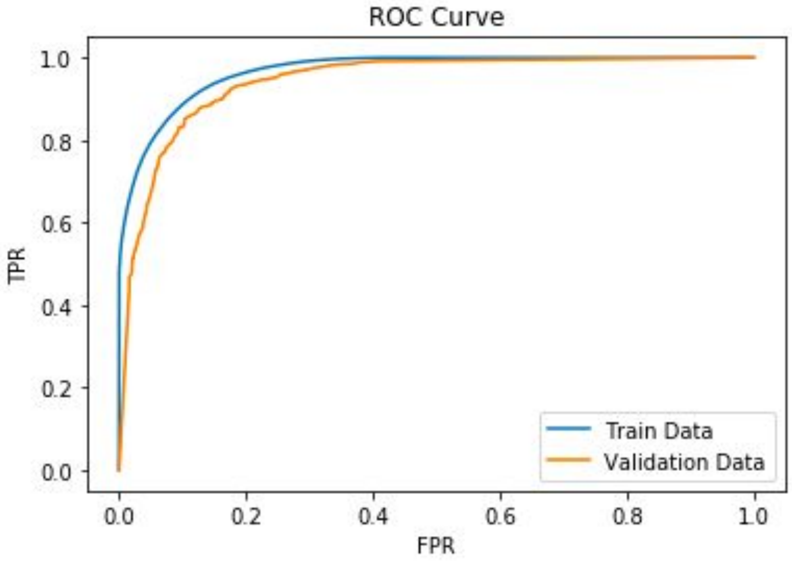
```
LogisticRegression(C=500, max_iter=50,
penalty='l2', solver='liblinear')
```

```
Train accuracy : 0.8421119110774283
Validation accuracy : 0.8516666666666667
Train F1 : 0.8507819028291752
Validation F1 : 0.8615001556178026
Train AUC : 0.8424693839181233
Validation AUC : 0.8508712660028449
Train Zweig-Campbell : 0.7195757113821138
Validation Zweig-Campbell : 0.7345506545337057
```



Appendix - Decision Tree Performance

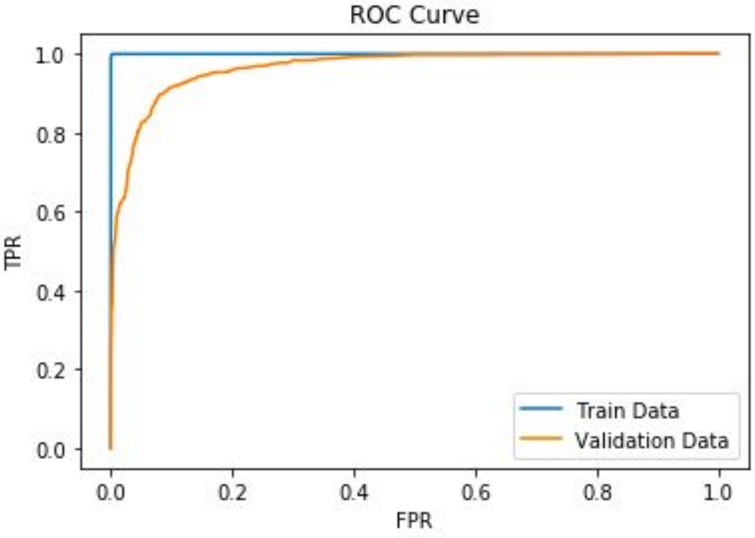
```
DecisionTreeClassifier (criterion='entropy',
max_depth=15, max_features=65,
min_samples_leaf=20)

Train accuracy : 0.8949728432487053
Validation accuracy : 0.8713333333333333
Train F1 : 0.8974659350144892
Validation F1 : 0.8754034861200775
Train AUC : 0.8951399674367881
Validation AUC : 0.8710526315789473
Train Zweig-Campbell : 0.811377879403794
Validation Zweig-Campbell : 0.765676147304236
```



Appendix - Random Forest (no Hyperparameter Tuning) Performance

```
RandomForestClassifier(criterion=gini,
max_depth=None, max_features='auto',
min_samples_leaf=1, oob_score=True,
n_estimators=100)
```

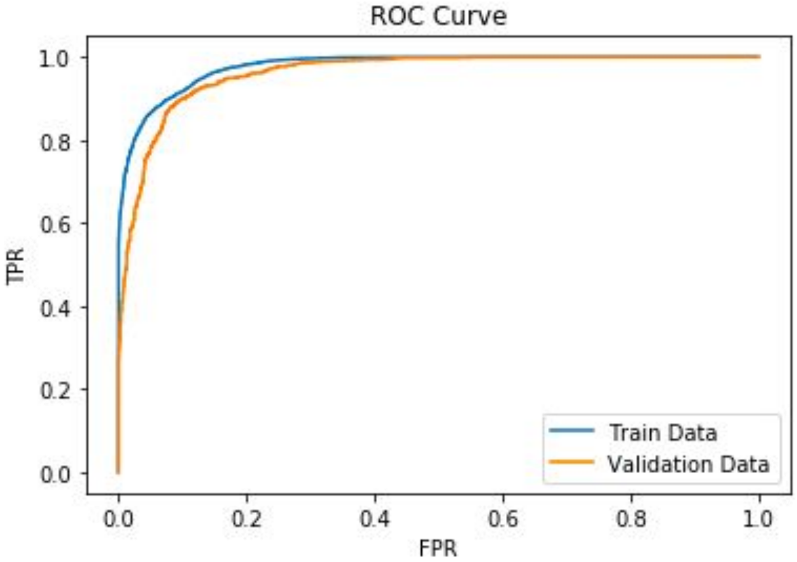
```
Train accuracy : 0.9980421876973601
Validation accuracy : 0.9016666666666666
Train F1 : 0.9980333692824971
Validation F1 : 0.9059611093401339
Train AUC : 0.9980489451418629
Validation AUC : 0.9012179943100996
Train Zweig-Campbell : 0.9965912940379404
Validation Zweig-Campbell : 0.8232463187674195
```



Appendix - Random Forest (GridSearchCV) Performance

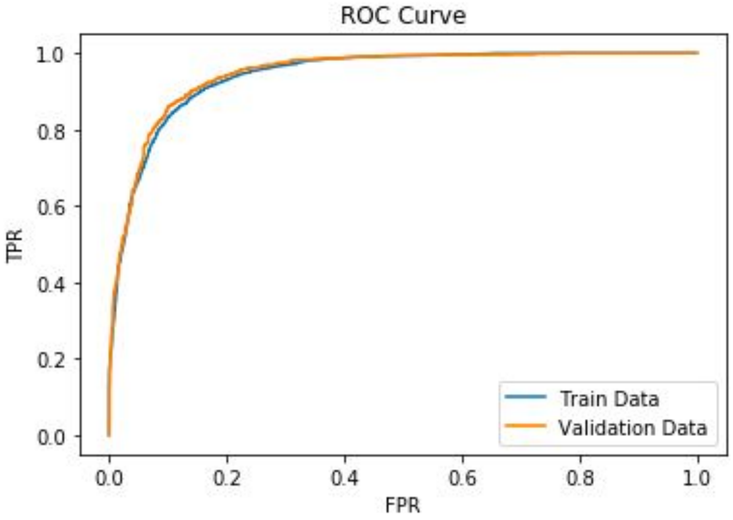
```
RandomForestClassifier(criterion='entropy',
max_depth=50, max_features=20,
min_samples_leaf=10,  n_estimators=100)
```

```
Train accuracy : 0.9095617026651509
Validation accuracy : 0.8963333333333333
Train F1 : 0.9120285047303108
Validation F1 : 0.9008606949314633
Train AUC : 0.9097504782626009
Validation AUC : 0.8958837126600284
Train Zweig-Campbell : 0.8389015921409214
Validation Zweig-Campbell : 0.813427156697776
```



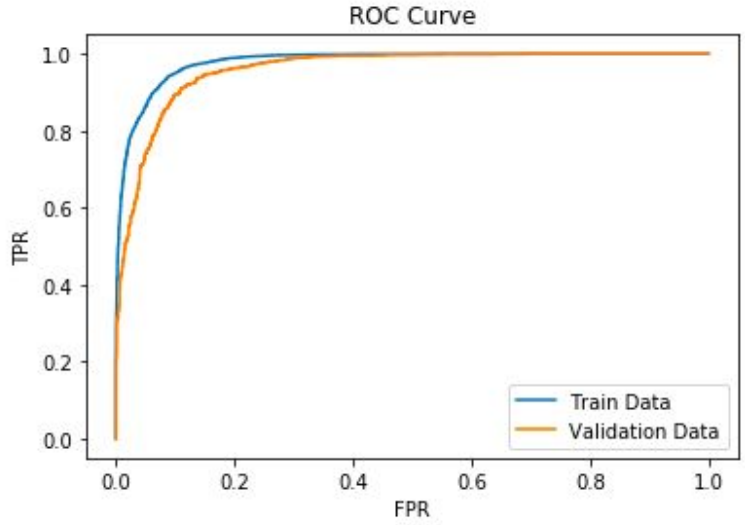
Appendix - XGBoost (no Hyperparameter Tuning) Performance

```
XGBClassifier(learning_rate=0.1, max_depth=3,
n_estimators=100)

Train accuracy : 0.872363268914993
Validation accuracy : 0.8773333333333333
Train F1 : 0.8762779308233853
Validation F1 : 0.8829516539440204
Train AUC : 0.8725713047800223
Validation AUC : 0.8768492176386913
Train Zweig-Campbell : 0.7709180216802168
Validation Zweig-Campbell : 0.778755771577097
```



Appendix - XGBoost (GridSearchCV) Performance

```
XGBClassifier(learning_rate=0.5, max_depth=10,
n_estimators=50)

Train accuracy : 0.9261715296198055
Validation accuracy : 0.899
Train F1 : 0.9269876959590282
Validation F1 : 0.9022895840051596
Train AUC : 0.926264993092188
Validation AUC : 0.8987108819345662
Train Zweig-Campbell : 0.8667005420054201
Validation Zweig-Campbell : 0.8167456418629798
```


ROC Curve
TPR
FPR
Train Data
Validation Data