Insurance Claim Prediction

Table of Contents

Contents

Problem 2: Insurance Claim Prediction

Insurance Claim Prediction

Summary:

An Insurance firm providing tour insurance is facing higher claim frequency. The management decides to collect data from the past few years. You are assigned the task to make a model which predicts the claim status and provide recommendations to management. Use CART, RF & ANN and compare the models' performances in train and test sets.

Sample of the Dataset:

	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product Name	Destination
0	48	C2B	Airlines	No	0.70	Online	7	2.51	Customised Plan	ASIA
1	36	EPX	Travel Agency	No	0.00	Online	34	20.00	Customised Plan	ASIA
2	39	CWT	Travel Agency	No	5.94	Online	3	9.90	Customised Plan	Americas
3	36	EPX	Travel Agency	No	0.00	Online	4	26.00	Cancellation Plan	ASIA
4	33	JZI	Airlines	No	6.30	Online	53	18.00	Bronze Plan	ASIA

Exploratory Data Analysis:

Let us check the basic info of the data frame

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000 entries, 0 to 2999
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Age           3000 non-null   int64  
 1   Agency_Code   3000 non-null   object 
 2   Type          3000 non-null   object 
 3   Claimed       3000 non-null   object 
 4   Commision     3000 non-null   float64
 5   Channel       3000 non-null   object 
 6   Duration      3000 non-null   int64  
 7   Sales         3000 non-null   float64
 8   Product Name  3000 non-null   object 
 9   Destination   3000 non-null   object 
dtypes: float64(2), int64(2), object(6)
memory usage: 234.5+ KB
```

Data Clean-up and pre-processing

- First, we rename the column Product name to Product_Name
- there are 139 duplicate values, and as it's under 5% of the total observations (139/3000 = 4.6%), we'll be dropping these from our dataset

```
Number of duplicate rows = 139
```

	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product_Name	Destination
63	30	C2B	Airlines	Yes	15.0	Online	27	60.0	Bronze Plan	ASIA
329	36	EPX	Travel Agency	No	0.0	Online	5	20.0	Customised Plan	ASIA
407	36	EPX	Travel Agency	No	0.0	Online	11	19.0	Cancellation Plan	ASIA
411	35	EPX	Travel Agency	No	0.0	Online	2	20.0	Customised Plan	ASIA
422	36	EPX	Travel Agency	No	0.0	Online	5	20.0	Customised Plan	ASIA
...	...	...	...	...	...	...	...	...	...	...

- We'll also be converting all 'object' type features into Categorical as they'll not be useful for our models in the original form.
- Agency_Code, Type, Claimed, Channel, Product_Name and Destination are the features currently of 'object' data type; once converted they'll all be in ordinal values
- For eg- Destination has 3 main categories – Asia, Americas and Europe, these will be referenced by 0, 1 and 2 respectively

Questions:

2.1 Read the data, do the necessary initial steps, and exploratory data analysis (Univariate, Bi-variate, and multivariate analysis).

Univariate Analysis

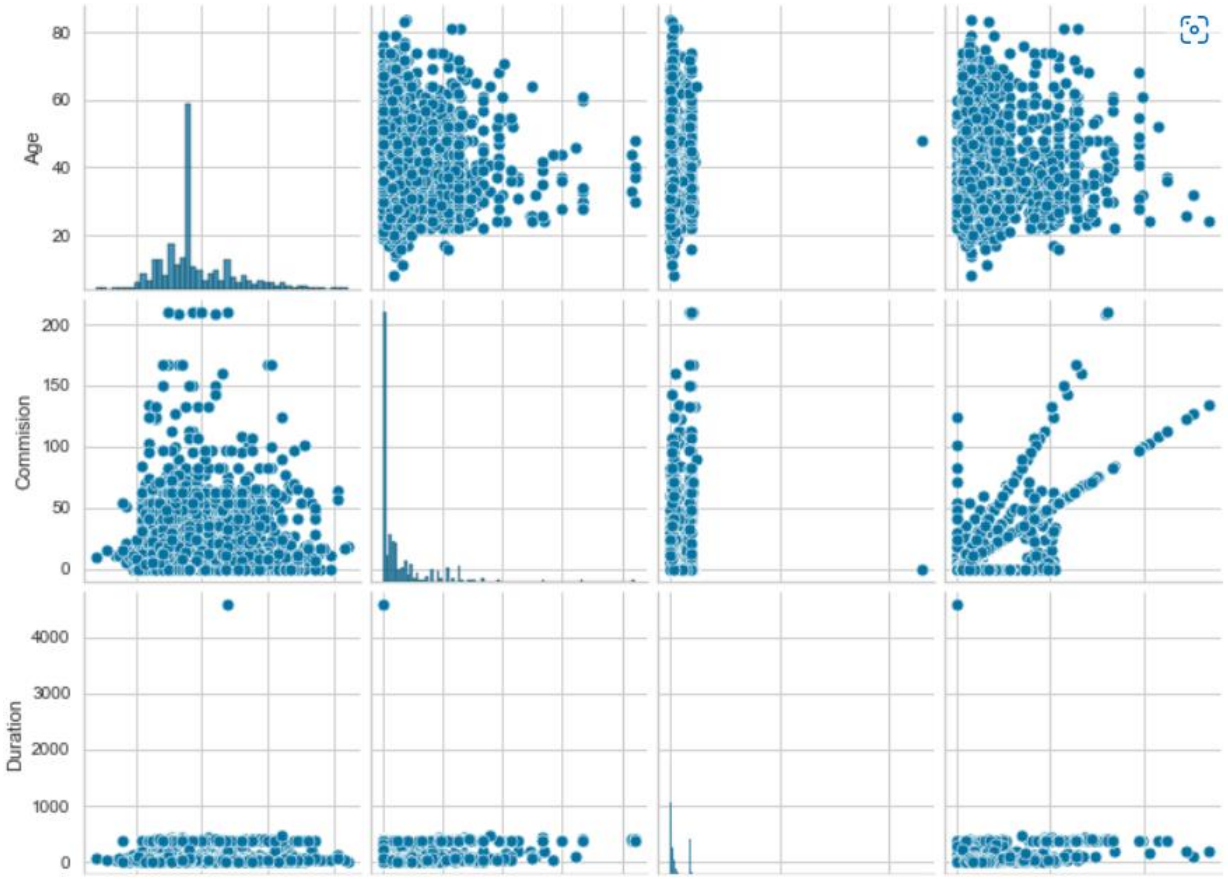
Basic info of the data frame post conversion

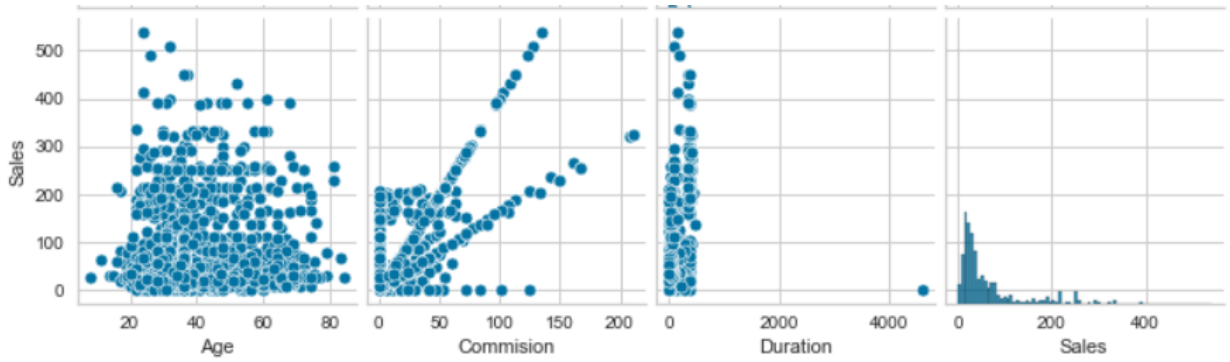
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2861 entries, 0 to 2999
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Age           2861 non-null   int64
 1   Agency_Code   2861 non-null   int8
 2   Type          2861 non-null   int8
 3   Claimed       2861 non-null   int8
 4   Commision     2861 non-null   float64
 5   Channel       2861 non-null   int8
 6   Duration      2861 non-null   int64
 7   Sales         2861 non-null   float64
 8   Product_Name  2861 non-null   int8
 9   Destination   2861 non-null   int8
dtypes: float64(2), int64(2), int8(6)
memory usage: 193.1 KB
```

Below is the statistical summary of the dataset after the conversion

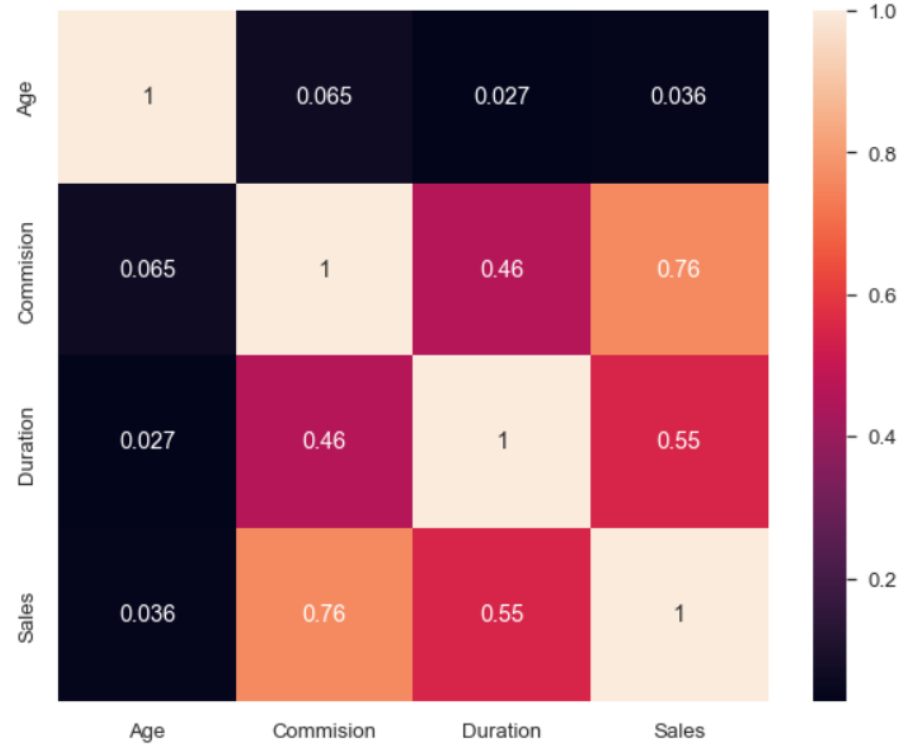
	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product_Name	Destination
count	2861.00	2861.00	2861.00	2861.00	2861.00	2861.00	2861.00	2861.00	2861.00	2861.00
mean	38.20	1.28	0.60	0.32	15.08	0.98	72.12	61.76	1.67	0.26
std	10.68	1.00	0.49	0.47	25.83	0.13	135.98	71.40	1.28	0.59
min	8.00	0.00	0.00	0.00	0.00	0.00	-1.00	0.00	0.00	0.00
25%	31.00	0.00	0.00	0.00	0.00	1.00	12.00	20.00	1.00	0.00
50%	36.00	2.00	1.00	0.00	5.63	1.00	28.00	33.50	2.00	0.00
75%	43.00	2.00	1.00	1.00	17.82	1.00	66.00	69.30	2.00	0.00
max	84.00	3.00	1.00	1.00	210.21	1.00	4580.00	539.00	4.00	2.00

Multivariate Analysis



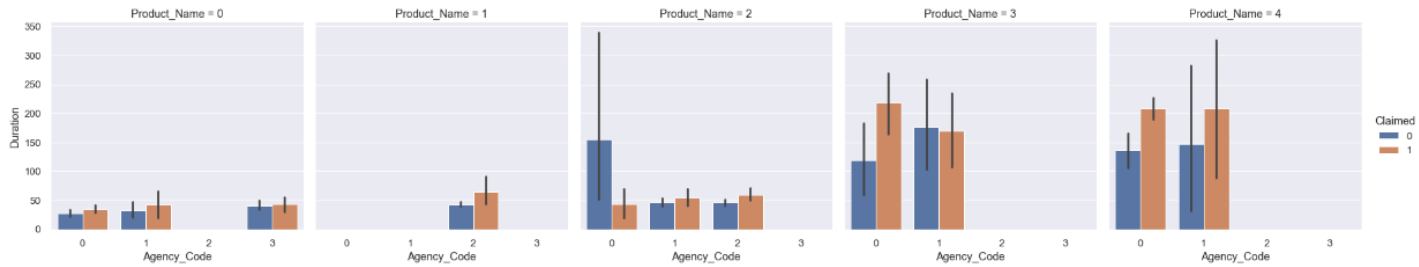


Correlation Heatmap

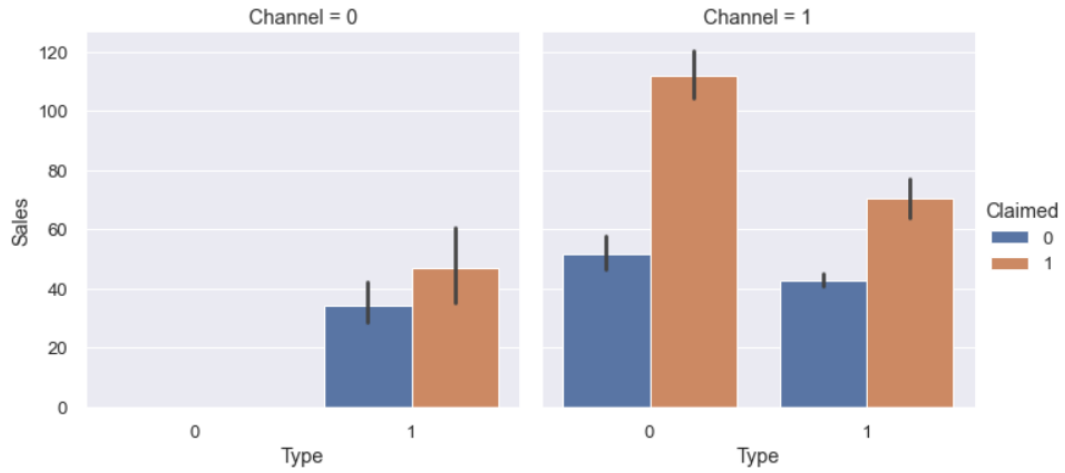


Catplot –

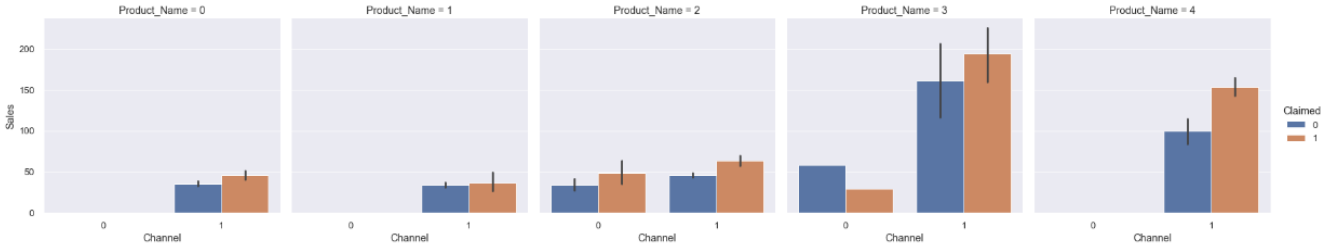
- Duration vs Agency Code for each Product along with Claim status



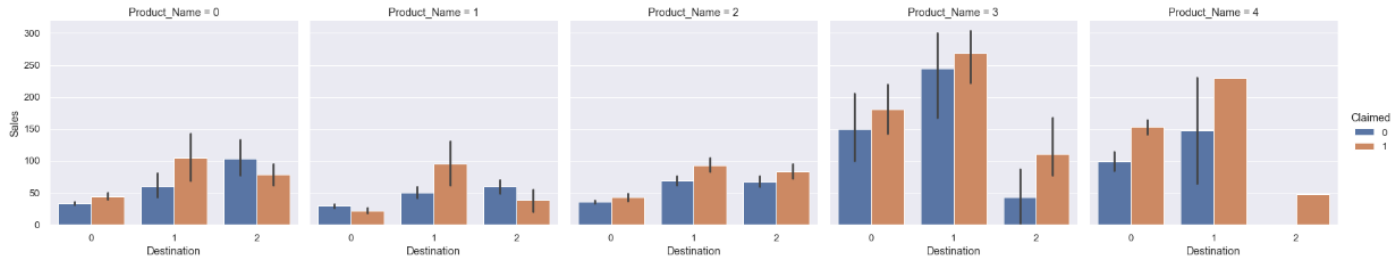
- Sales vs Type for each Channel along with Claim status



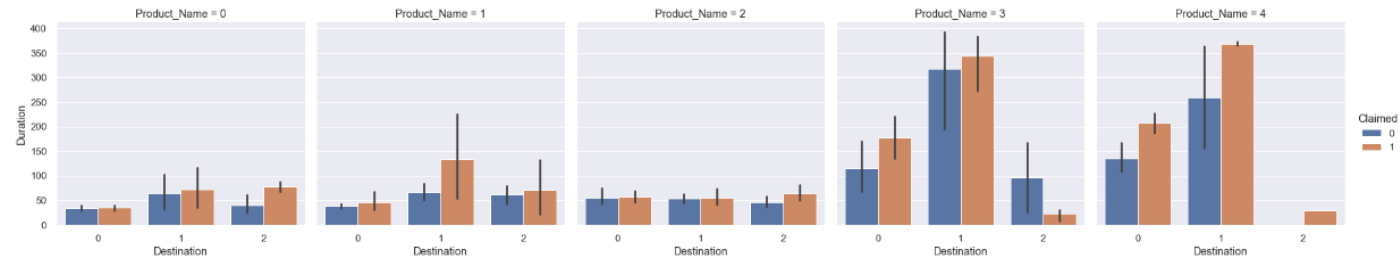
- Sales vs Channel for each Product along with Claim status



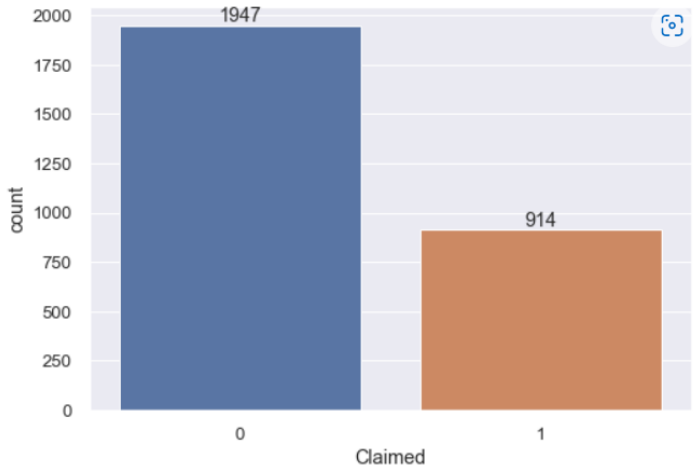
- Sales vs Destination for each Product along with Claim status



- Duration vs Destination for each Product along with Claim status



Proportion of 1s and 0s in the target variable



Insights: --

- The dataset has 10 columns and 3000 rows, of which 139 were duplicates and we removed them
- There are no Null values as indicated by non-null values
- The Age, Commission, Duration and Sales columns do have some outliers; however, we can ignore them as they'll not impact any of the 3 algorithms
- We have plotted scatter diagrams for all the numerical columns in the dataset. A scatter plot is a visual representation of the degree of correlation between any two columns
- We've plotted a heatmap to display the numerical values of the degree of correlation between any two columns
- Sales is highly correlated with Commission (0.76) and Duration (0.55) which is quite normal
- We've drawn 5 catplots, which shows frequencies of the categories of one, two or more categorical variables and is very useful to make inferences
- As we can see, the Claims are the highest for products 3 (Gold Plan) & 4 (Silver Plan) in agencies 0 (C2B) and 1 (CWT); also, longer the duration of the tour, higher the chances of Claims
- The Claims are highest for Channel 1, which is Online; meaning the customers prefer to make their bookings through this mode
- Linking the last 2 points, we can conclude that the Claims are highest for products 3 (Gold Plan) & 4 (Silver Plan) through the Online channel – this is the same inference from our 3rd catplot
- We've also drawn a countplot for the target variable in our dataset, Claimed and we see there are 1947 (68%) not claimed and 914 (32%) who have claimed
- **We conclude that we have an imbalanced dataset as only 32% of the observations have claimed for insurance which it makes it difficult to train the models and provide the optimum accuracy**

2.2 Data Split: Split the data into test and train, build classification model CART, Random Forest, Artificial Neural Network

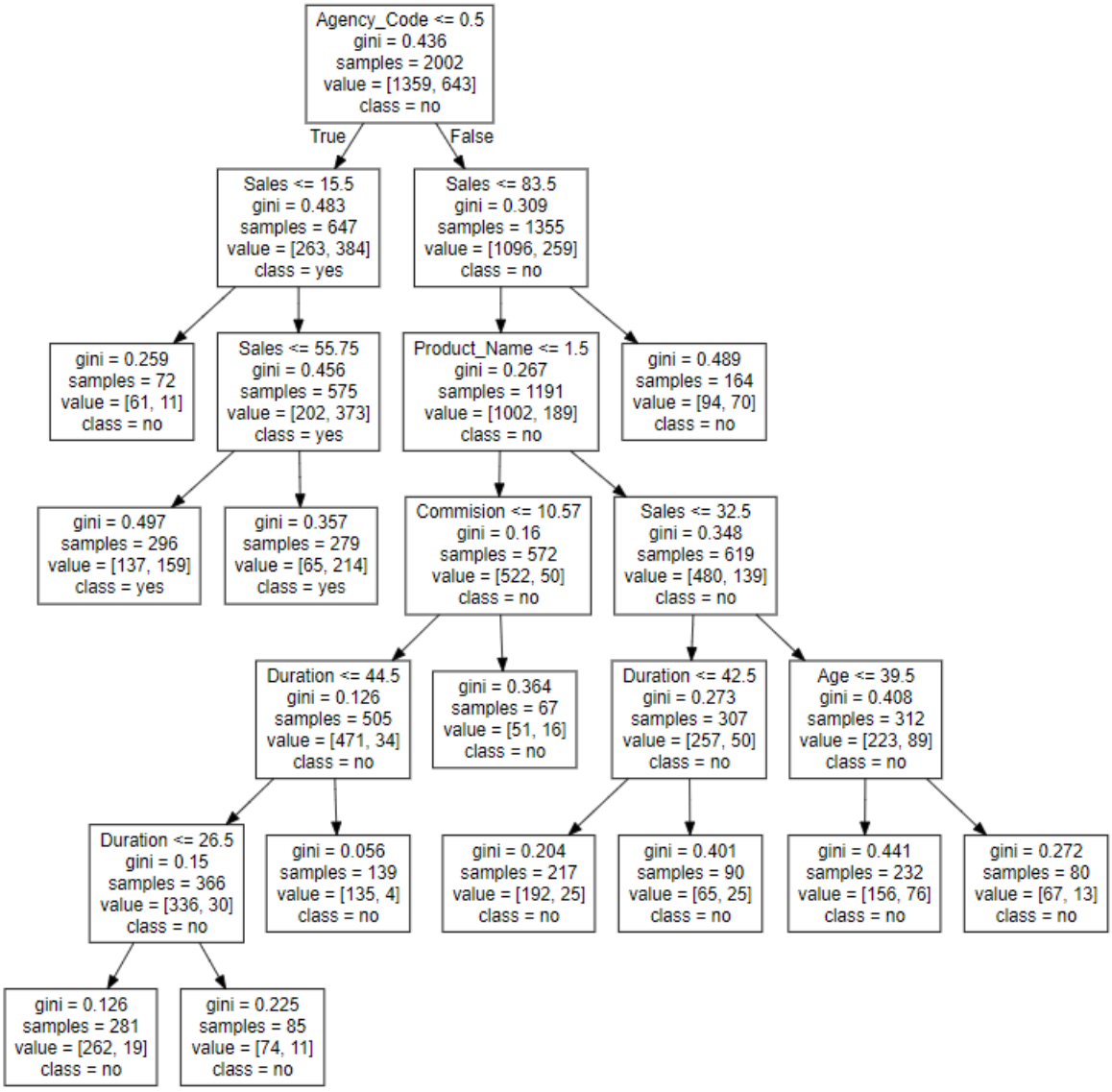
- Before we split, we create 2 datasets, wherein the Claimed feature is dropped from one and is the only feature in the other dataset
- We're going for the classic 70-30 split, wherein 70% of the observations, 2002 rows and 9 columns are in the 'training' dataset and 30%, 859 rows and 9 columns are in the 'testing' dataset
- We've used the grid search feature to obtain the best parameters to be used for each of the model

CART MODEL

- For CART, we've got the below parameters as the optimum values to generate our Decision Tree

```
DecisionTreeClassifier
DecisionTreeClassifier(max_depth=10, min_samples_leaf=50, min_samples_split=300,
                       random_state=1)
```

- As we can see, the first node has 2002 samples, which is equal to our testing dataset
- There are only 7 levels making further splits as our criteria for max_depth was 10; this determines the maximum depth of the tree
- The minimum samples in each leaf were 50 and least value is 67 in our tree
- The minimum number of samples required to split an internal node is 300 in our case; this can be evidenced by the fact that in leaves where there are less than 300 samples, no further split occurs. For e.g.- 72, 164, 296, 279, 67, 139, 217 are some of the leaves where the split doesn't continue
- The random_state is a very important component as setting the random state assures that the CART implementation works through the same randomized list of features when looking for the minimum
- rs=1 maintains the initial and last value and randomizes the rest while rs = 0 just does normal randomization of all data



- We can get the predicted classes and the probability for our testing dataset
- As we can see from the below there's a 97% probability the 2nd observation belongs to 0 i.e. not claimed

	0	1
0	0.573171	0.426829
1	0.971223	0.028777
2	0.232975	0.767025
3	0.837500	0.162500
4	0.837500	0.162500

Random Forest (RF)

- For RF, we've got the below parameters as the optimum values to generate our set of Decision Trees, called the Forest

```
RandomForestClassifier
RandomForestClassifier(max_depth=10, max_features=9, min_samples_leaf=10,
                       min_samples_split=60, n_estimators=300, random_state=1)
```

- n_estimators -- This is the number of trees you want to build before taking the maximum voting or averages of predictions. Higher number of trees give you better performance but makes your code slower

Artificial Neural Network (ANN)

- For ANN, we've got the below parameters as the optimum values to generate our Decision

```
MLPClassifier
MLPClassifier(hidden_layer_sizes=100, max_iter=2500, random_state=1, tol=0.01)
```

- hidden_layer_sizes : This parameter allows us to set the number of layers and the number of nodes we wish to have in the Neural Network Classifier
- max_iter: It denotes the number of epochs (one round of forward and backward propagation)
- tol: It is the tolerance for the stopping criteria. This tells the model to stop searching for a minimum (or maximum) once some tolerance is achieved, i.e., once its close enough

Variable Importance – this denotes the most important features which contribute to the prediction of our target variable

  - In CART, Agency Code and Sales are the top features
  - In Random Forest, Agency Code, Sales and Duration are the top features
  - In Artificial Neural Network, this function is not available

2.3 Performance Metrics: Comment and Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC_AUC score, classification reports for each model.

A few definitions before we get into model evaluation

- Confusion Matrix – A 2X2 tabular structure reflecting the performance of the model in four blocks

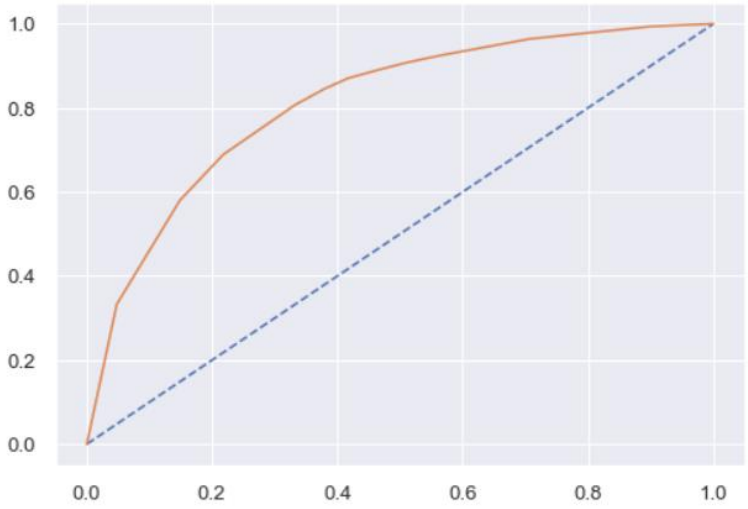
Confusion Matrix	Predicted Positive	Predicted Negative
Actual Positive	True Positive	False Negative
Actual Negative	False Positive	True Negative

- Accuracy – How accurately / cleanly does the model classify the data points. Lesser the false predictions, more the accuracy
- Sensitivity / Recall – How many of the actual True data points are identified as True data points by the model.
- Specificity – How many of the actual Negative data points are identified as negative by the model
- Precision – Among the points identified as Positive by the model, how many are really Positive
- Receiver Operating Characteristic, ROC curve -- It's a graph showing the performance of a classification model at all classification thresholds.
    - It shows the trade-off between sensitivity (or TPR) and specificity (1 – FPR)
    - Classifiers that give curves closer to the top-left corner indicate a better performance
- Area under the ROC Curve, AUC -- It provides an aggregate measure of performance across all possible classification thresholds

CART Model

Training dataset

ROC Curve and AUC score is 81%



Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	1157	202
Actual Negative	270	373

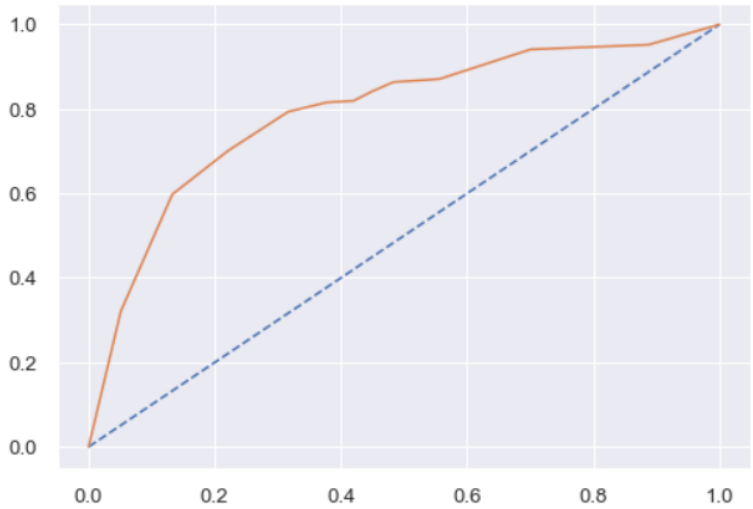
Classification Report

	precision	recall	f1-score	support
0	0.81	0.85	0.83	1359
1	0.65	0.58	0.61	643
accuracy			0.76	2002
macro avg	0.73	0.72	0.72	2002
weighted avg	0.76	0.76	0.76	2002

- The Accuracy of the CART model on the training dataset is 76%

Testing dataset

ROC Curve and AUC score is 79%



Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	510	78
Actual Negative	109	162

Classification Report

	precision	recall	f1-score	support
0	0.82	0.87	0.85	588
1	0.68	0.60	0.63	271
accuracy			0.78	859
macro avg	0.75	0.73	0.74	859
weighted avg	0.78	0.78	0.78	859

- The Accuracy of the CART model on the testing dataset is 78%

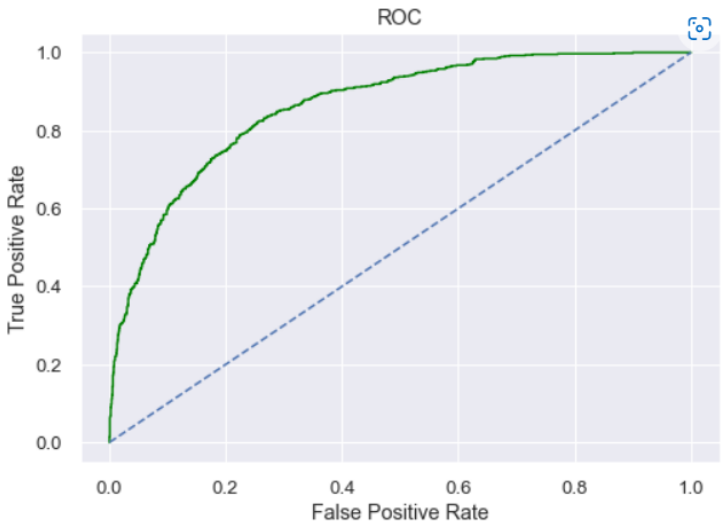
CART Conclusion

- Although the Accuracy on the testing dataset at 78% is slightly more than the accuracy on the training dataset at 76%, it's alright as it's within the threshold of +/- 10%
- Accuracy, AUC, Precision and Recall for test data is almost in line with training data.
- This proves no overfitting or underfitting has happened, and overall, the model is a good model for classification

Random Forest Model

Training dataset

ROC Curve and AUC score is 86%



Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	1225	134
Actual Negative	266	377

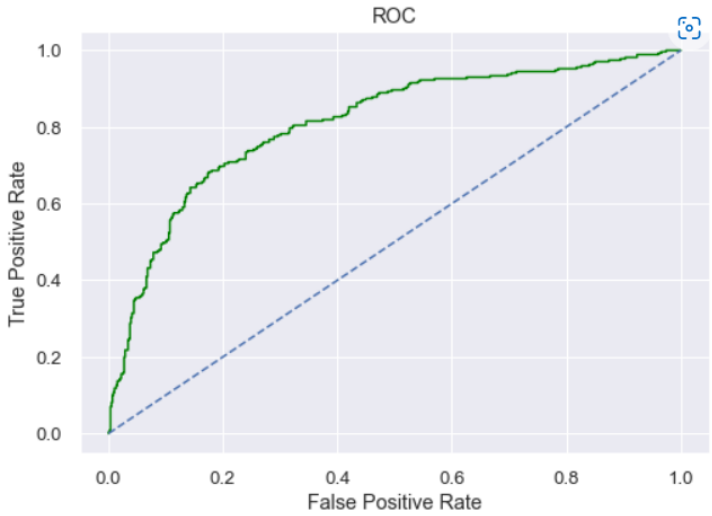
Classification Report

	precision	recall	f1-score	support
0	0.82	0.90	0.86	1359
1	0.74	0.59	0.65	643
accuracy			0.80	2002
macro avg	0.78	0.74	0.76	2002
weighted avg	0.79	0.80	0.79	2002

- The Accuracy of the RF model on the training dataset is 80%

Testing dataset

ROC Curve and AUC score is 81%



Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	522	66
Actual Negative	117	154

Classification Report

	precision	recall	f1-score	support
0	0.82	0.89	0.85	588
1	0.70	0.57	0.63	271
accuracy			0.79	859
macro avg	0.76	0.73	0.74	859
weighted avg	0.78	0.79	0.78	859

- The Accuracy of the RF model on the testing dataset is 79%

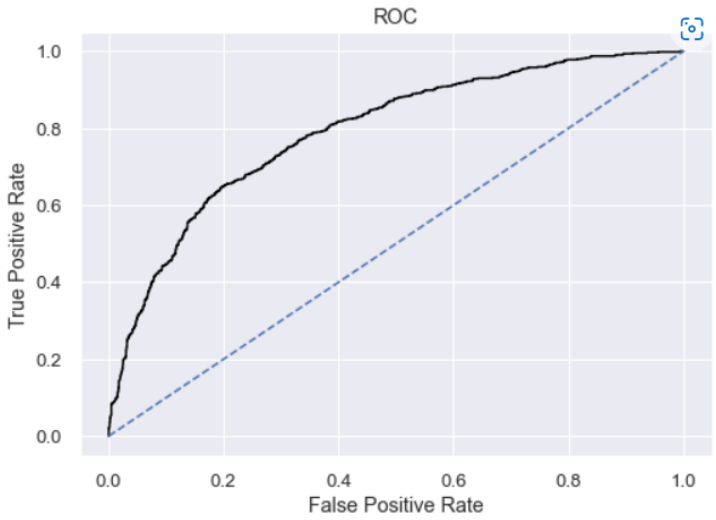
RF Conclusion

- Accuracy, AUC, Precision and Recall for test data is almost in line with training data.
- This proves no overfitting or underfitting has happened, and overall, the model is a good model for classification

Artificial Neural Network

Training dataset

ROC Curve and AUC score is 79%



Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	1163	196
Actual Negative	280	363

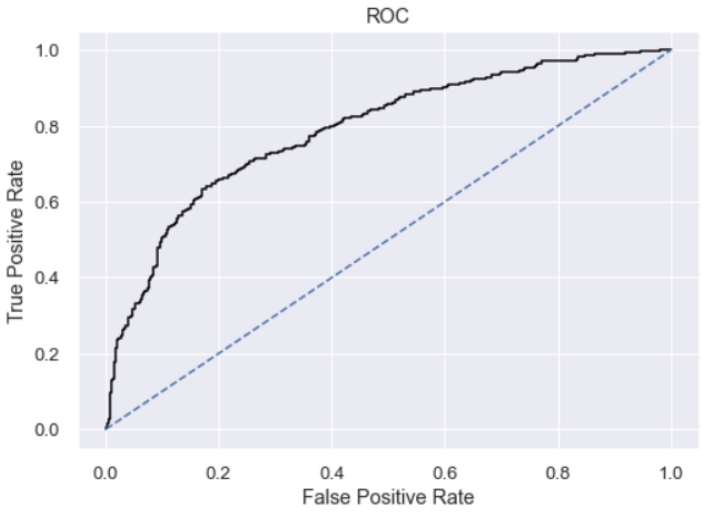
Classification Report

	precision	recall	f1-score	support
0	0.81	0.86	0.83	1359
1	0.65	0.56	0.60	643
accuracy			0.76	2002
macro avg	0.73	0.71	0.72	2002
weighted avg	0.76	0.76	0.76	2002

- The Accuracy of the ANN model on the training dataset is 76%

Testing dataset

ROC Curve and AUC score is 79%



Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	510	78
Actual Negative	118	153

Classification Report

	precision	recall	f1-score	support
0	0.81	0.87	0.84	588
1	0.66	0.56	0.61	271
accuracy			0.77	859
macro avg	0.74	0.72	0.72	859
weighted avg	0.76	0.77	0.77	859

- The Accuracy of the ANN model on the testing dataset is 77%

ANN Conclusion

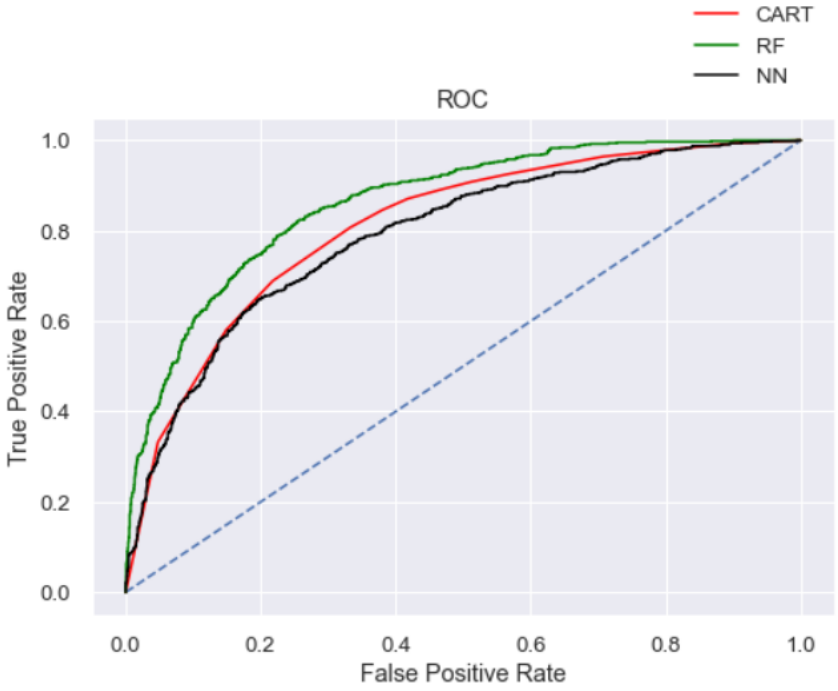
- Although the Accuracy on the testing dataset at 77% is slightly more than the accuracy on the training dataset at 76%, it's alright as it's within the threshold of +/- 10%
- Accuracy, AUC, Precision and Recall for test data is almost in line with training data.
- This proves no overfitting or underfitting has happened, and overall, the model is a good model for classification

2.4 Final Model: Compare all the models and write an inference which model is best/optimized.

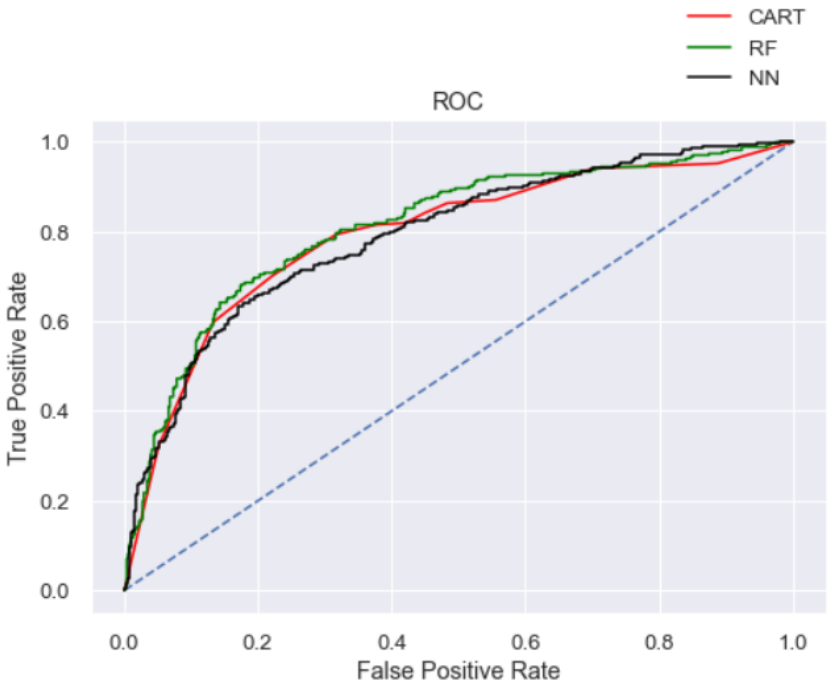
- We've created the below dataframe to compare the models
- Since we have an imbalanced dataset (proportion of Claimed is less than half of Unclaimed), Accuracy is not the appropriate metric to compare the 3 models
- Since Accuracy cannot be considered, we look at Precision and Recall
- Precision = TP / (TP + FP) and Recall = TP / (TP + FN)
- As both have TP in the numerator, reducing FP or FN will increase the predicting capability as our motive is to get the model with the highest True Positives
- In our business scenario, False Positives are fairly acceptable as they are predicted to apply for a claim but do not end up doing so
- Hence reducing the False Negatives (data points which should have been identified as True) will help the business to plan for the appropriate claims and be prepared
- With Recall being our preferred metric, CART comes out on top with a score of 60% being the highest, RF is 2nd at 57% and ANN is the least at 56%
- However, we should also consider the ROC and AUC score to get a holistic view
- As we can see from the 2 ROC curves on Training and Test data, RF model covers the highest area and is closer to the top-left indicating better performance
- The Test AUC is also the highest for the RF model at 81% when compared to the other 2 models (both at 79%)
- Overall, all the 3 models are reasonably stable enough to be used for making any future predictions
- **Considering the overall attributes, we conclude that Random Forest has slightly better performance than the Cart and Neural network model**
- From Cart and Random Forest Model, the variable change is found to be a very useful feature amongst all other features for predicting if a person travelling will apply for a Claim
    - In CART, Agency Code and Sales contribute to the prediction
    - In RF, Agency Code, Sales and Duration contribute to the prediction

	CART Train	CART Test	Random Forest Train	Random Forest Test	Neural Network Train	Neural Network Test
Accuracy	0.76	0.78	0.80	0.79	0.76	0.77
AUC	0.81	0.79	0.86	0.81	0.79	0.79
Recall	0.58	0.60	0.59	0.57	0.56	0.56
Precision	0.65	0.68	0.74	0.70	0.65	0.66
F1 Score	0.61	0.63	0.65	0.63	0.60	0.61

Training data ROC



Testing data ROC



2.5 Inference: Based on the whole Analysis, what are the business insights and recommendations

Insights

- Claims are the highest for products 3 (Gold Plan) & 4 (Silver Plan) in agencies 0 (C2B) and 1 (CWT); also, longer the duration of the tour, higher the chances of Claims
- Claims are highest for Channel 1, which is Online; meaning the customers prefer to make their bookings through this mode
- Linking the last 2 points, we can deduce that the Claims are highest for products 3 (Gold Plan) & 4 (Silver Plan) through the Online channel
- For bookings made through the Online channel, Claims are quite high for both types of tour insurance firms – Airlines and Travel Agency
- For bookings made through the Offline channel, Claims are high when made through Travel Agency
- Across all 4 products, Claims are highest for people travelling to the Americas

Recommendations

- For travel duration of more than 90 days, the Insurance company can stop covering for pre-existing diseases as these Claims tend to be more expensive when treated abroad
    - For travel duration of more than 150 days, the company could place certain claims under the restricted list
- The company should consider increasing the premiums it charges for longer duration travels; a progressive increase in relation to duration would be good
- The company should consider not covering for claims related to Adventure Sports; the customer can buy insurance at his/her destination
- Given only 11% (319/2861) of the people have travelled to the Americas, a further analysis on their age and the agencies through which the bookings were made can tell us more on the profile of such customers
- The company should consider revising the Commissions paid to the travel agents for selling the Gold Plan; perhaps they're being sold solely for the purpose of receiving high commissions
- The company should consider increasing their investment allocation in risky assets to earn better returns and offset the increase in costs
- The company can also look into closing the branches that are consistently under-performing over the last 15-24 months