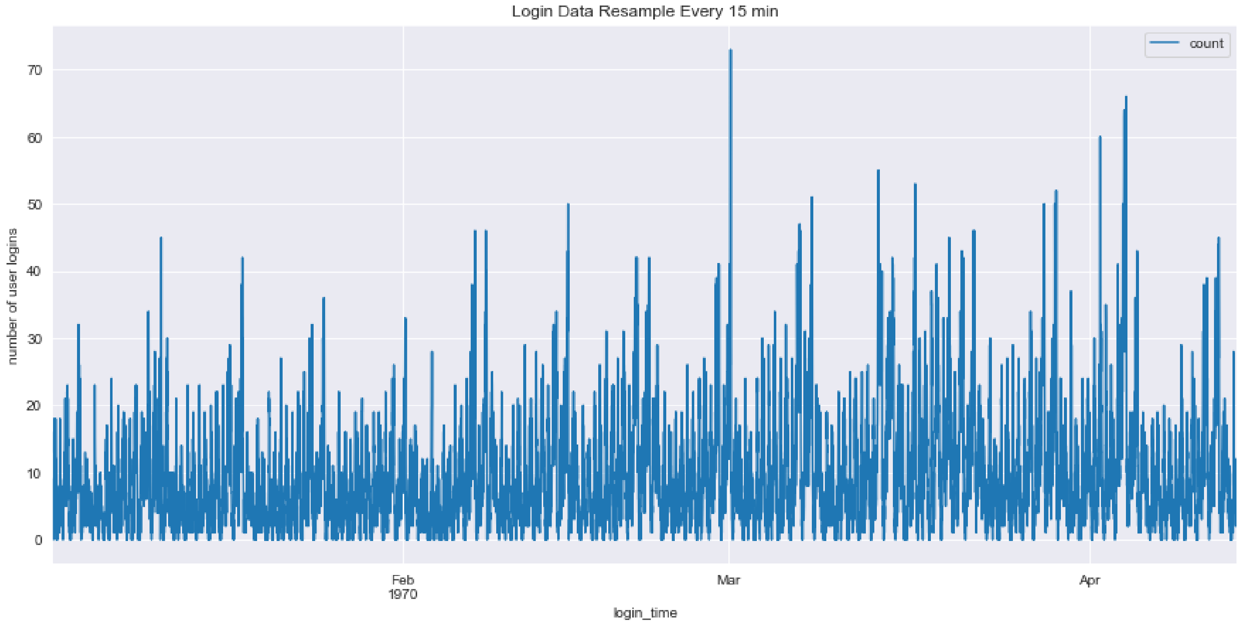
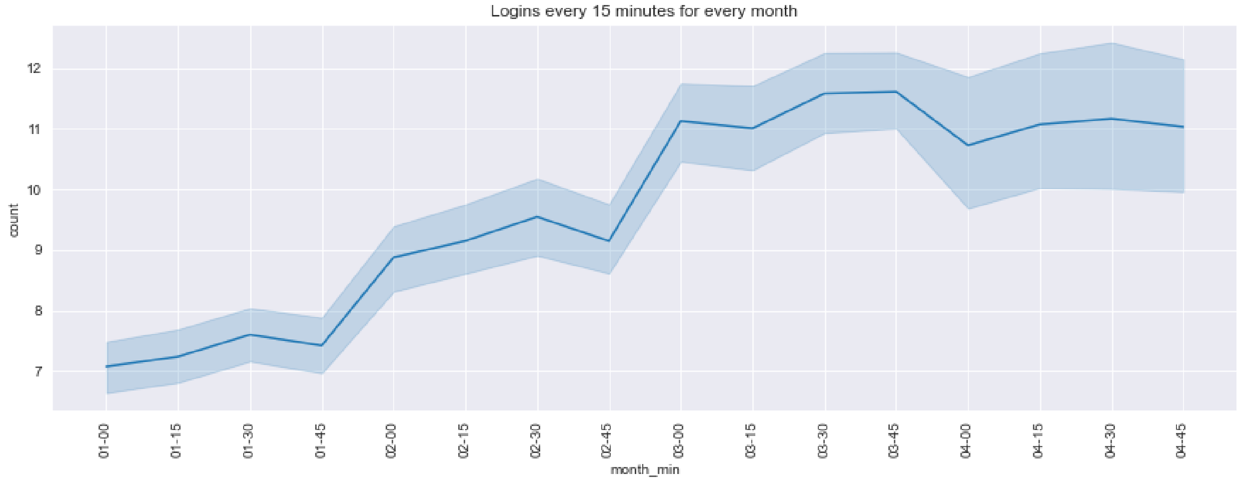
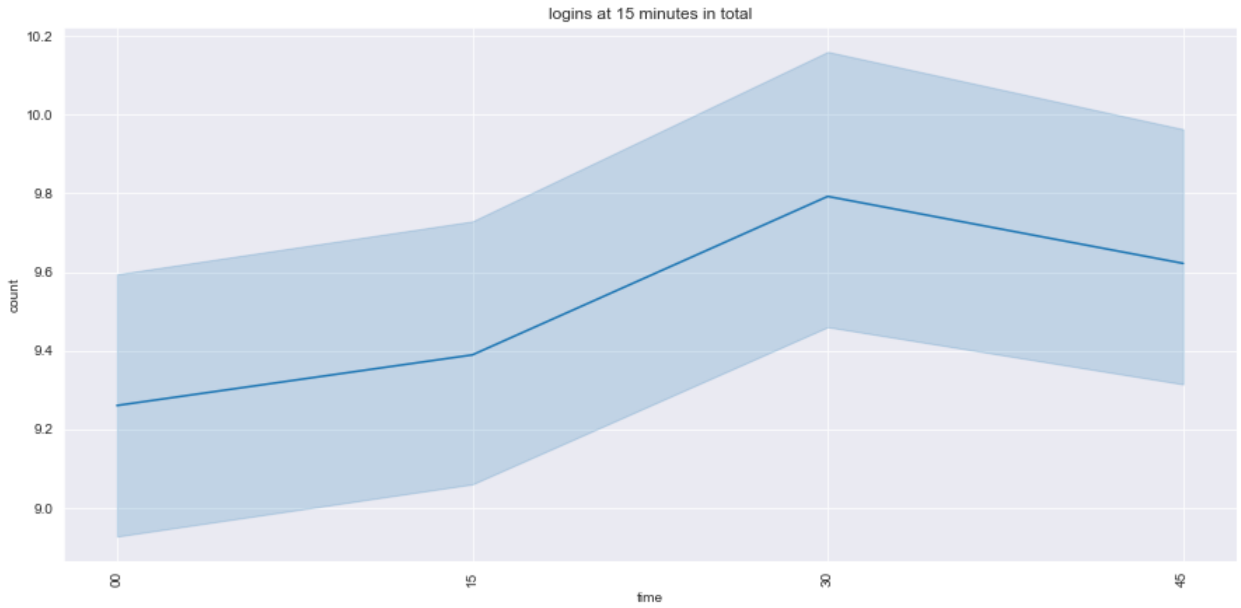
Part 1 - Exploratory data analysis:



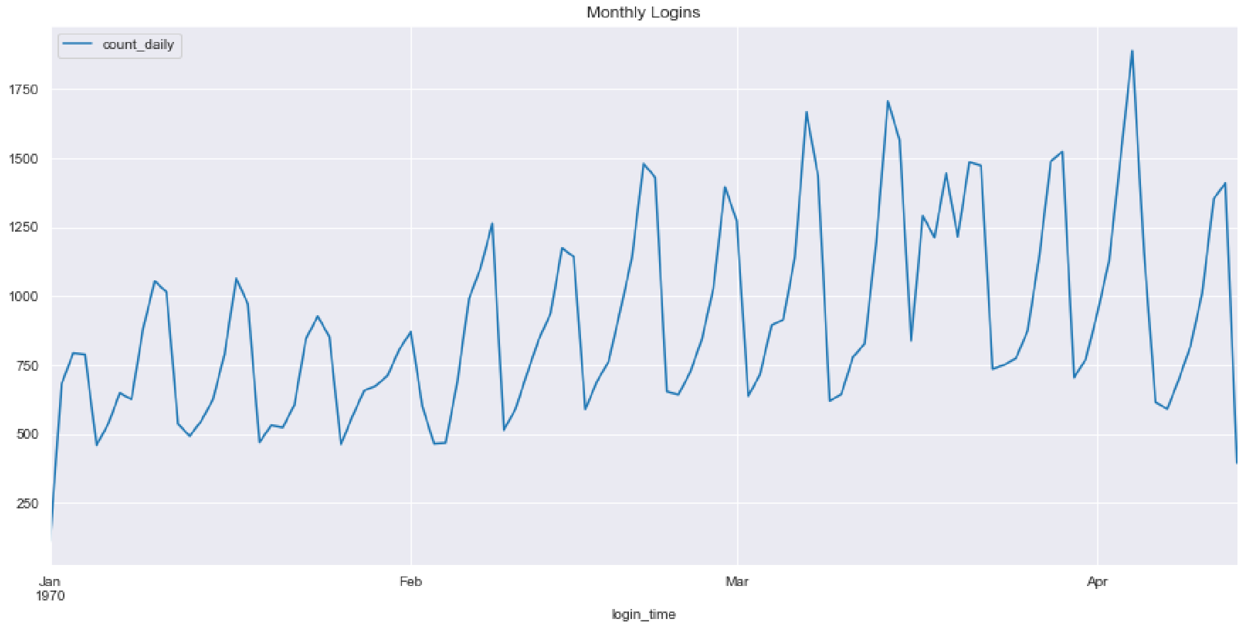
There are many spikes in the data where logins increase, but generally logins are increasing closer to April.



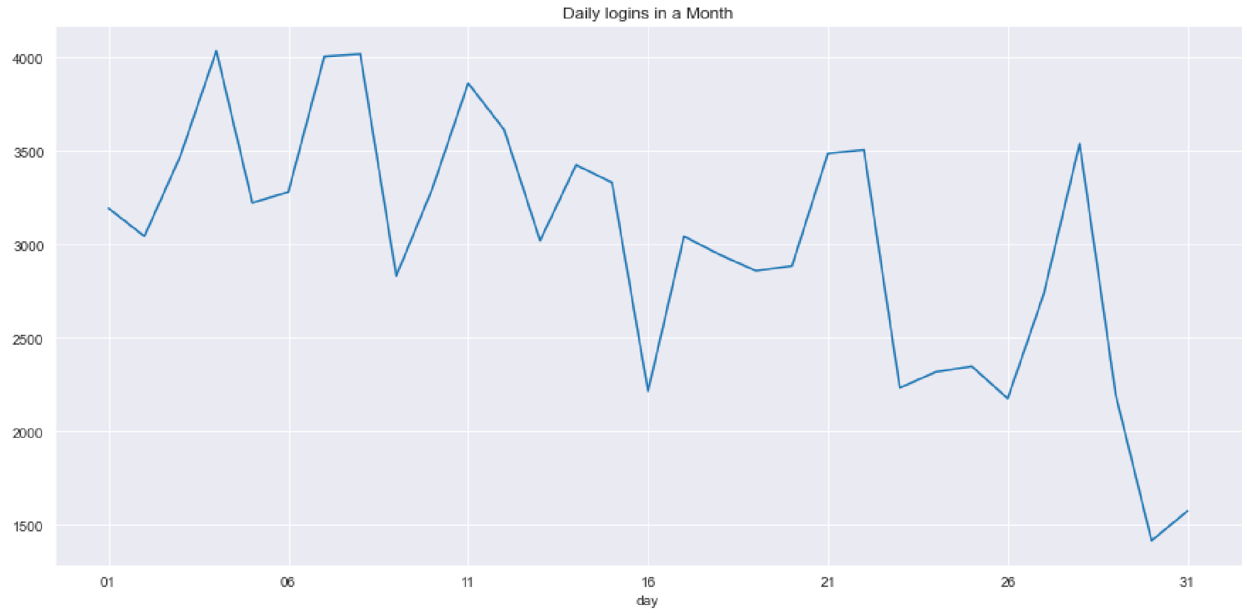
This plot confirms that there is an increase in loogins from month Jan to April



There is a spike every 30 minutes



There is an upward trend from Jan to Apr but for each month logins seem to decrease at the end of the month.



plot confirms that there is a decrease in logins towards the end of the month.

Part 2:

1. Since the experiment is to encourage driver partners to serve both cities, the key measure of success is the number of drivers that cross from one city to the other.
2. Take a portion of the drivers in one city and divide them into two parts, one gets told that they will be reimbursed for tolls and that is the test set, and the other will not and this one is the control. Then measure if there is an increase in drivers crossing the bridge in the test set compared to the control. The experiment is a success if there is an increase in drivers crossing from the test set compared with the control.

Part 3 - Predictive modeling

The data:

The response variable is weather the customer ordered a trip within the first 30 days. If it is not equal to 0 then retention is 1 else retention is 0. The

variable trips_in_first_30_days has to be removed before the models are run because there will be data leakage otherwise.
there are 69.22% of users that were retained. There were 7961 missing values for avg_rating_of_driver which were imputed using IterativeImputer and the data was standardized.

Predictive models:

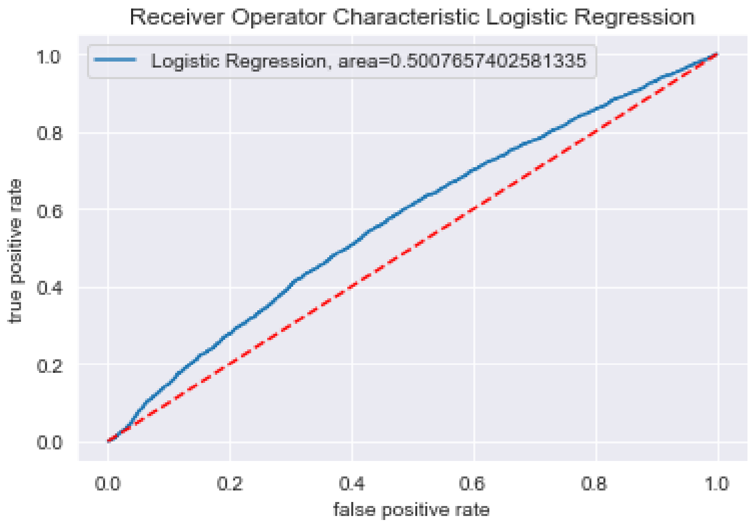
The company needs to find customers who are not returning so the model needs to reduce false positives, where positives are returning customers when they are actually negatives, so as not to lose customers. So I need to improve precision.

I tried three predictive algorithms, Logistic regression, Random forest and Kneighbor classifier.

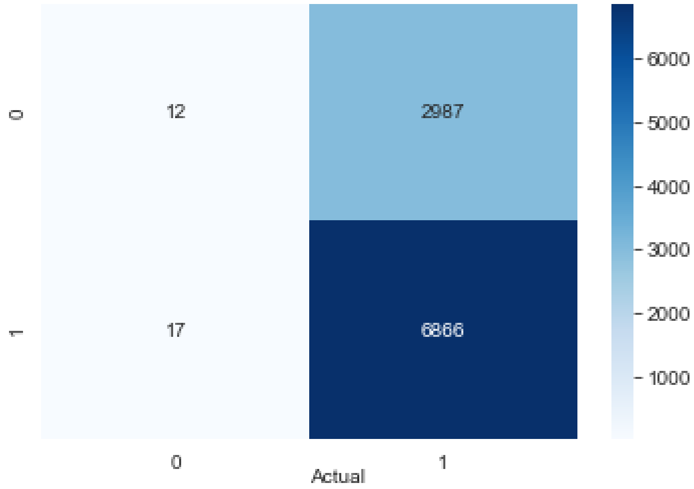
Logistic Regression: This is the easiest model and good place to start
A gridsearch was performed with the parameters penalty of l1 and l2 and C with 7 values ranging from logspace -3 to 3. This is the classification report:

```
              precision    recall  f1-score   support

           0       0.41      0.00      0.01      2999
           1       0.70      1.00      0.82      6883

    accuracy                           0.70      9882
   macro avg       0.56      0.50      0.41      9882
weighted avg       0.61      0.70      0.57      9882
```

Precision tells percentage of results found were actually true and recall tells how many of the data results were found by the model. Here the model found none of the negative results but found all the positives, however only 70% of the positives were correctly classified.
Since it is important for the company to identify customers who will be returning and those who are not, precision and precision important metrics. The results are not great for this model with only 70% precision.



ROC curve has smal area of 50% with the results not much better than guessing(the red line).



The data is not balanced and it shows here because many of the positive cases were predicted correctly, but it model does not do well with negative cases. Out of 2999 negatives, 2987 were mispredicted to be positives, and that is a lot of mislabeled customers, so this model is not good.

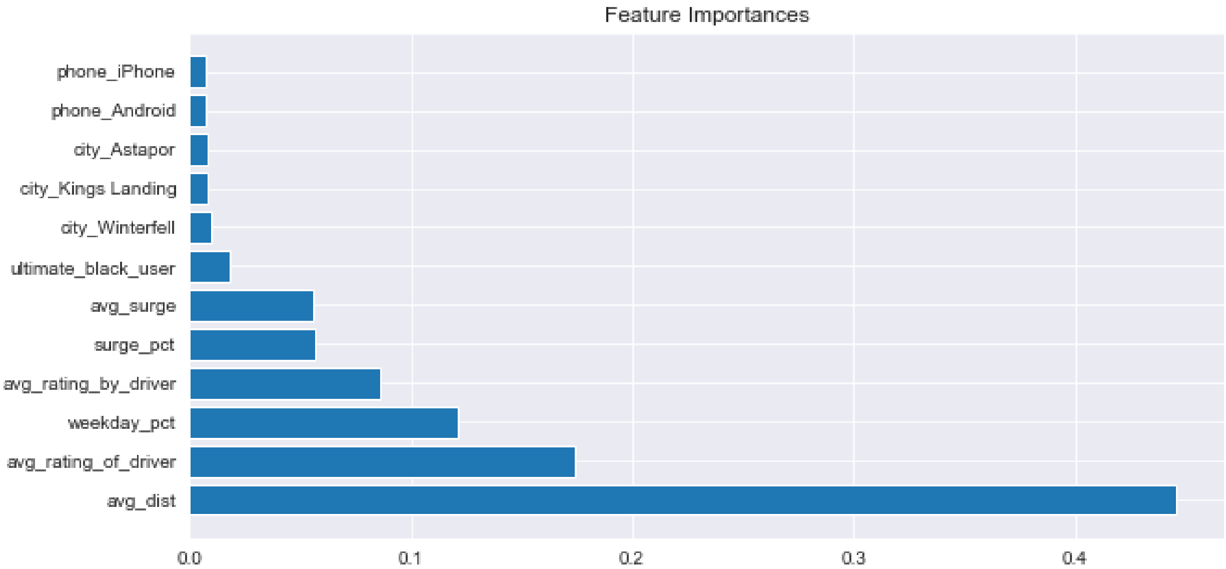
Random Forest:

For random forest a random search was performed with the following metrics:

n_estimators': [200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, 2000],

```
 'max_features': ['auto', 'sqrt'],
 'max_depth': [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, None],
 'min_samples_split': [2, 5, 10],
 'min_samples_leaf': [1, 2, 4],
 'bootstrap': [True, False]
```

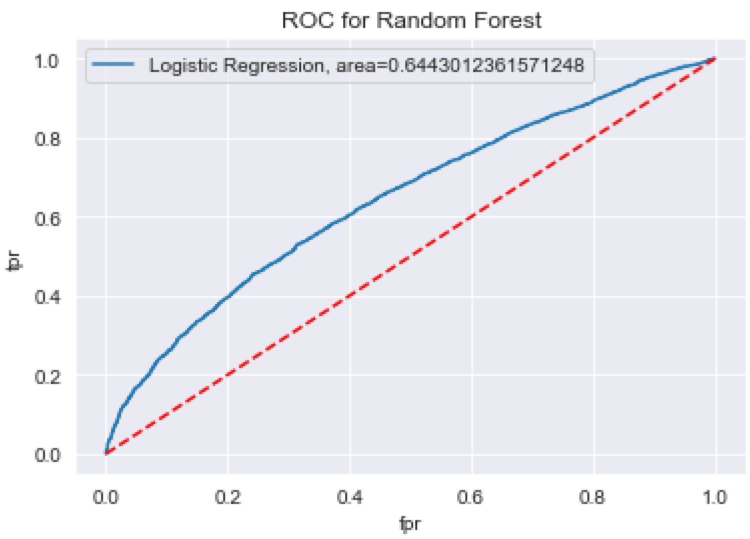
Then best model was fit to data and these are the feature importances:



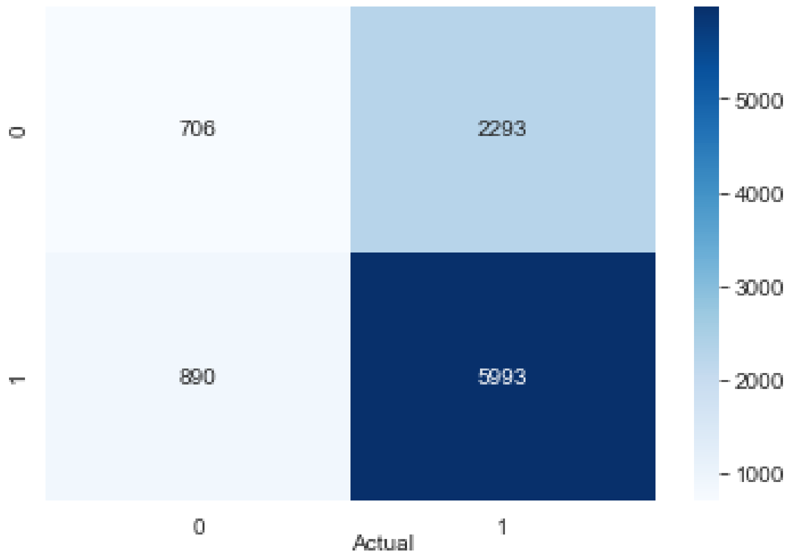
The least important features were removed to try to improve the model and then this model model was fit to this new data and these is the classification report

```
              precision    recall  f1-score   support

           0       0.44      0.24      0.31      2999
           1       0.72      0.87      0.79      6883

    accuracy                           0.68      9882
   macro avg       0.58      0.55      0.55      9882
weighted avg       0.64      0.68      0.64      9882
```

Accuracy has not improved much but there is an improvement for recall and precision for negatives.



There is also an improvement in the roc curve which means this model is predicting better logistic regression.



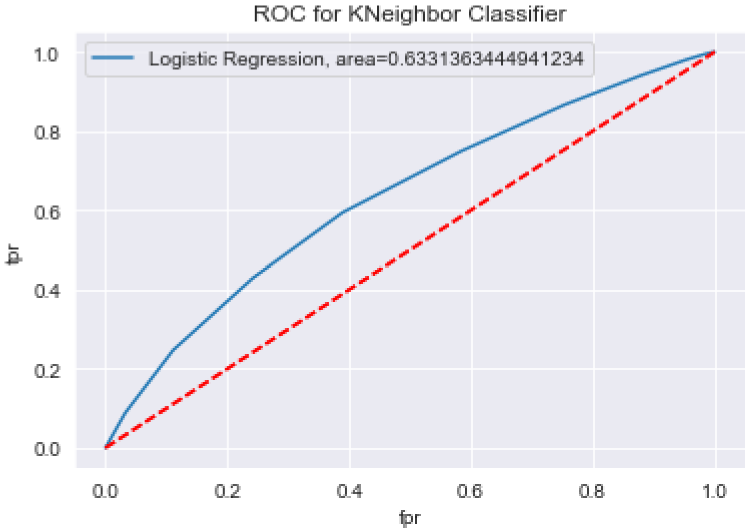
Out of 2999 negative, or not returning customers, 706 were correctly predicted and 2293 were mislabeled, an improvement but still not great.

KNeighbor Classifier:

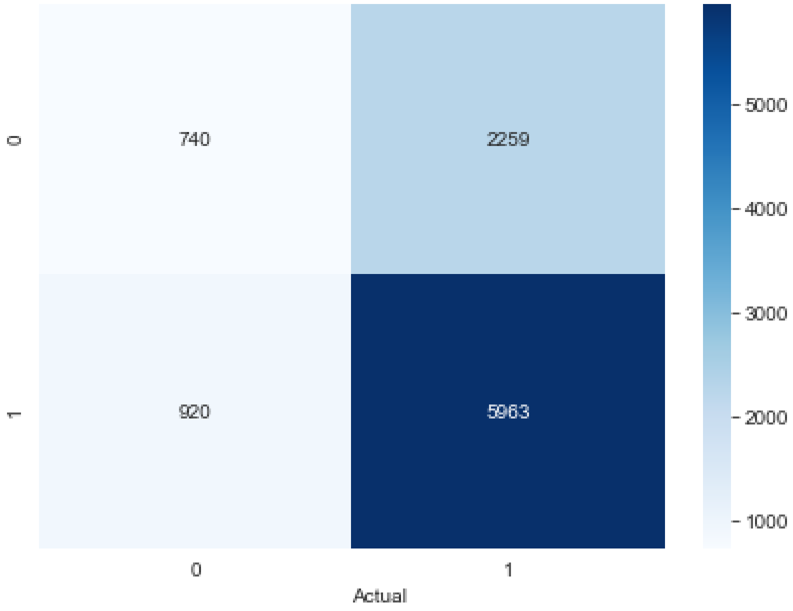
```
      precision    recall  f1-score   support

   0       0.45      0.25      0.32      2999
   1       0.73      0.87      0.79      6883

accuracy                       0.68      9882
```

```
   macro avg       0.59      0.56      0.55      9882
weighted avg       0.64      0.68      0.65      9882
```

There is no great improvement in precision and recall.



Not much difference in the roc curve either



I think I need to try more models like SVM to improve the results but for now the one I would go with is random forest because it performed better than the logistic model and slightly better than nearest neighbor.