Campaign Response Model

To train and test the campaign response model by using last 2 years data and comparing 2 model methods ; Logistic Regression and XGBoost

Data Preparation

- RFM model will be used to predict campaign response. Recency is calculated and create data set with RFM variables

```
campaign_date = dt.datetime(2015,3,17)
df_transactions['recent']= campaign_date - df_transactions['trans_date']
df_transactions['recent'].astype('timedelta64[D]')
df_transactions['recent']=df_transactions['recent'] / np.timedelta64(1, 'D')
df_transactions.head()
```

	customer_id	trans_date	tran_amount	recent
0	CS5295	2013-02-11	35	764.0
1	CS4768	2015-03-15	39	2.0
2	CS2122	2013-02-26	52	749.0
3	CS1217	2011-11-16	99	1217.0
4	CS1850	2013-11-20	78	482.0

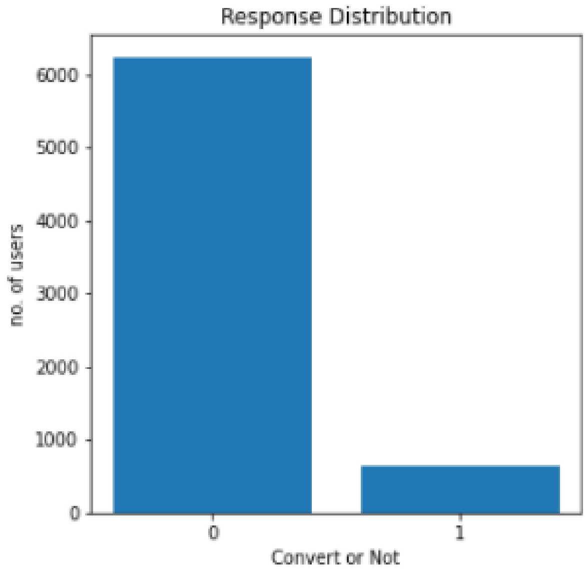
```
df_transactions = df_transactions[df_transactions['trans_date'] >= dt.datetime(2013,3,17)]
df_rfm = df_transactions.groupby('customer_id').agg({'recent': lambda x:x.min(),       # Recency
                                                     'customer_id': lambda x: len(x),   # Frequency
                                                     'tran_amount': lambda x: x.sum()}) # Monetary Value

df_rfm.rename(columns={'recent': 'recency',
                       'customer_id': 'frequency',
                       'tran_amount': 'monetary_value'}, inplace=True)
```

```
df_rfm = df_rfm.reset_index()
df_rfm.head()
```

	customer_id	recency	frequency	monetary_value
0	CS1112	62.0	6	358
1	CS1113	36.0	11	775
2	CS1114	33.0	11	804
3	CS1115	12.0	11	831
4	CS1116	204.0	5	333

Calculating response rate

- Found out that data is imbalanced

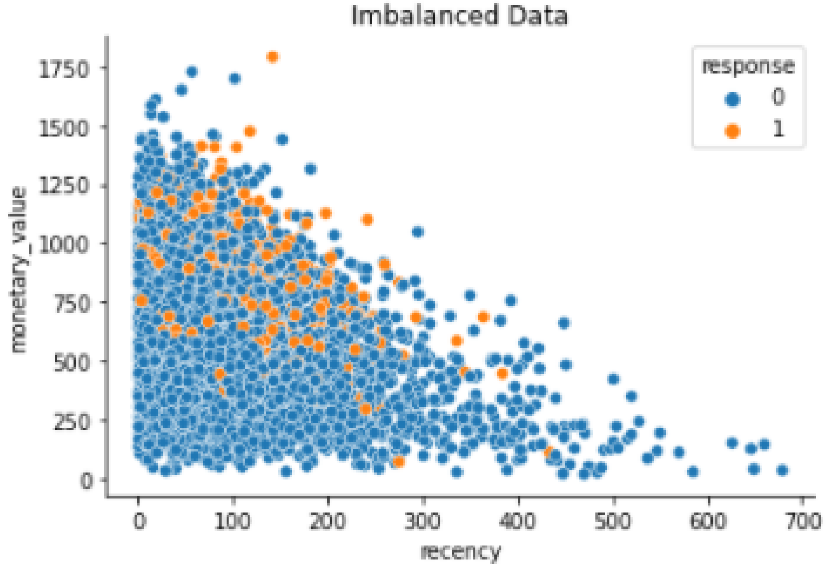


Creating train and test dataset

- Splitting data frame into X and y, using scatterplot to see the spread of data by recency, monetary value, and response

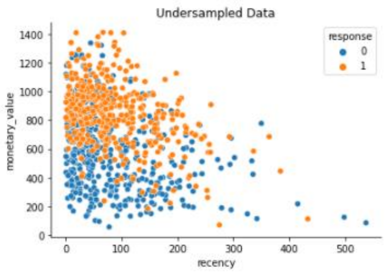
```
sns.scatterplot(data=df_modeling, x='recency', y='monetary_value', hue='response')
sns.despine()
plt.title("Imbalanced Data")
```

```
Text(0.5, 1.0, 'Imbalanced Data')
```



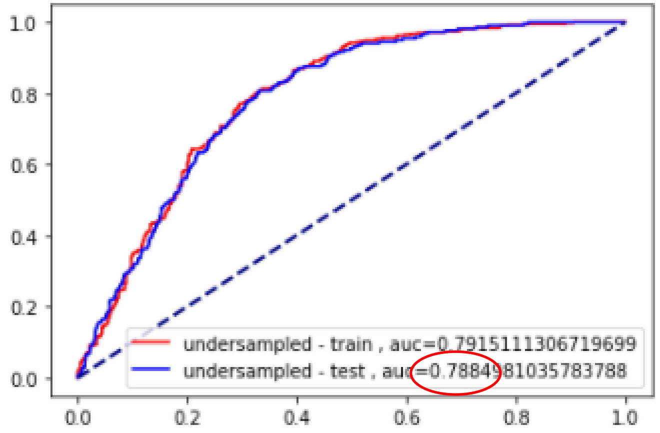
Fixing imbalanced with Undersampling

- Using the fixed data to training 2 models methods : Logistic Regression and XGBoost.



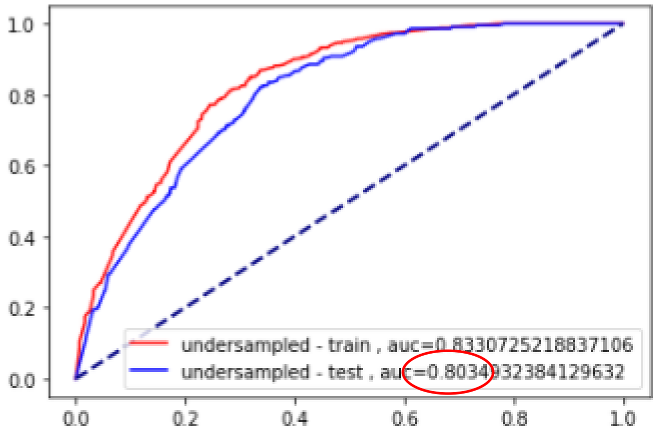
Logistic Regression

```
logistic regression model - undersampled
training set
              precision    recall  f1-score   support

           0       0.76      0.69      0.72       429
           1       0.71      0.78      0.75       429

    accuracy                           0.73       858
   macro avg       0.74      0.73      0.73       858
weighted avg       0.74      0.73      0.73       858

test set
              precision    recall  f1-score   support

           0       0.96      0.68      0.80      1848
           1       0.22      0.78      0.35       218

    accuracy                           0.69      2066
   macro avg       0.59      0.73      0.57      2066
weighted avg       0.89      0.69      0.75      2066
```



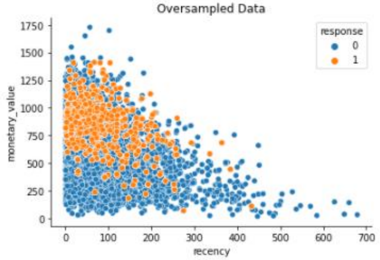
XGBoost

```
Stopping. Best iteration:
[16]    validation_0-auc:0.803493

training set
              precision    recall  f1-score   support

           0       0.82      0.68      0.74       429
           1       0.73      0.85      0.78       429

    accuracy                           0.76       858
   macro avg       0.77      0.76      0.76       858
weighted avg       0.77      0.76      0.76       858

test set
              precision    recall  f1-score   support

           0       0.97      0.65      0.78      1848
           1       0.22      0.83      0.35       218

    accuracy                           0.67      2066
   macro avg       0.59      0.74      0.56      2066
weighted avg       0.89      0.67      0.73      2066
```



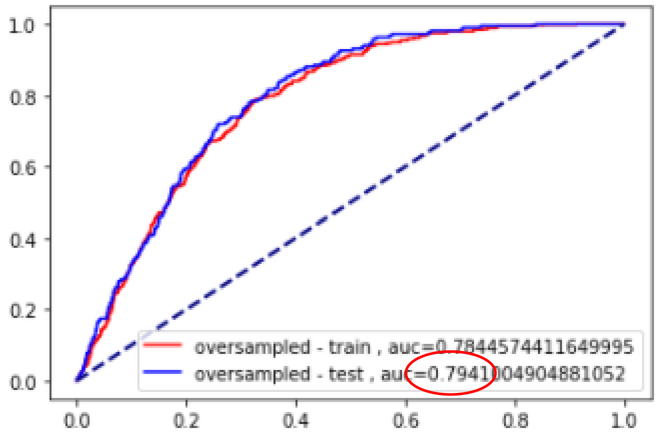
Fixing imbalanced with Oversampling

- Using the fixed data to training 2 models methods : Logistic Regression and XGBoost.



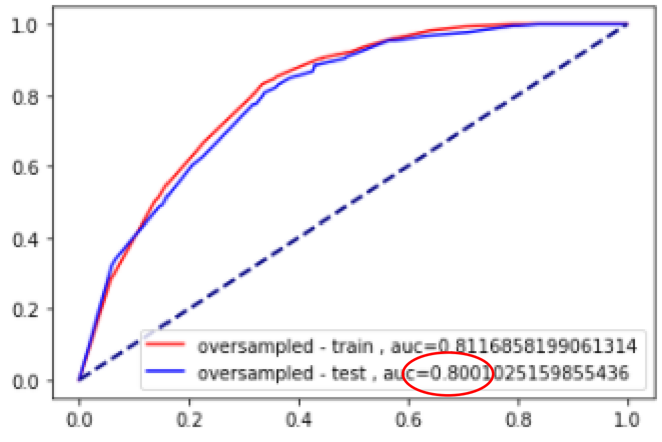
Logistic Regression

```
logistic regression model - oversampled
training set
              precision    recall  f1-score   support

           0       0.73      0.70      0.71      4389
           1       0.71      0.74      0.72      4389

    accuracy                           0.72      8778
   macro avg       0.72      0.72      0.72      8778
weighted avg       0.72      0.72      0.72      8778

test set
              precision    recall  f1-score   support

           0       0.96      0.69      0.80      1848
           1       0.23      0.77      0.35       218

    accuracy                           0.70      2066
   macro avg       0.59      0.73      0.58      2066
weighted avg       0.88      0.70      0.76      2066
```

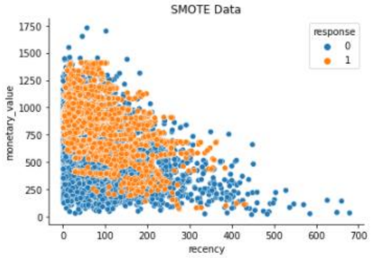


XGBoost

```
Stopping. Best iteration:
[3]     validation_0-auc:0.800103

training set
              precision    recall  f1-score   support

           0       0.80      0.66      0.72      4389
           1       0.71      0.83      0.77      4389

    accuracy                           0.75      8778
   macro avg       0.76      0.75      0.75      8778
weighted avg       0.76      0.75      0.75      8778

test set
              precision    recall  f1-score   support

           0       0.97      0.66      0.79      1848
           1       0.22      0.81      0.35       218

    accuracy                           0.68      2066
   macro avg       0.59      0.73      0.57      2066
weighted avg       0.89      0.68      0.74      2066
```



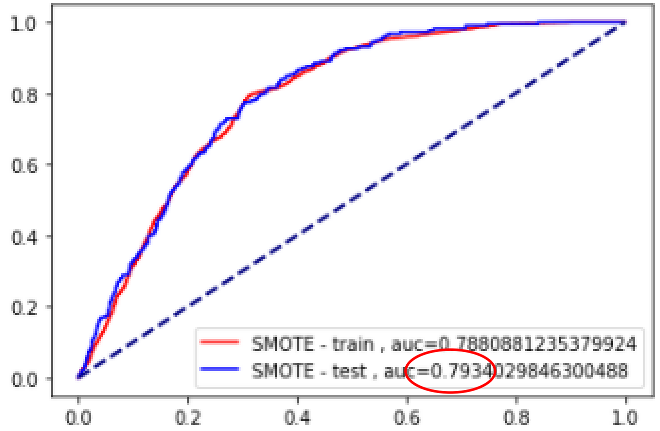
Fixing imbalanced with SMOTE



- Using the fixed data to training 2 models methods : Logistic Regression and XGBoost.

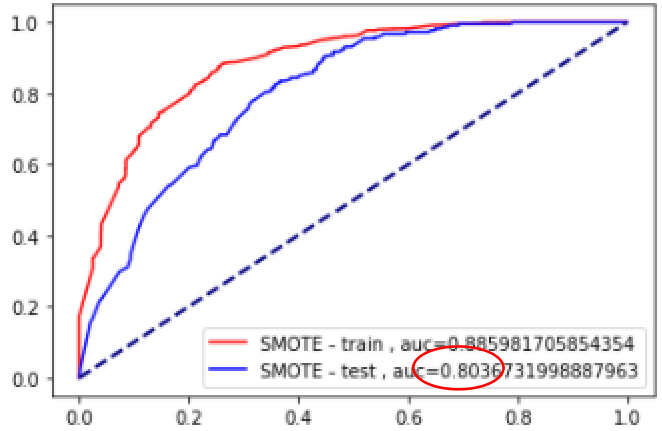
Logistic Regression

```
logistic regression model - SMOTE
training set
              precision    recall  f1-score   support

           0       0.75      0.70      0.72      4389
           1       0.72      0.77      0.74      4389

    accuracy                           0.73      8778
   macro avg       0.73      0.73      0.73      8778
weighted avg       0.73      0.73      0.73      8778

test set
              precision    recall  f1-score   support

           0       0.96      0.70      0.81      1848
           1       0.23      0.78      0.36       218

    accuracy                           0.70      2066
   macro avg       0.60      0.74      0.58      2066
weighted avg       0.89      0.70      0.76      2066
```



XGBoost

```
Stopping. Best iteration:
[28]    validation_0-auc:0.803673

training set
              precision    recall  f1-score   support

           0       0.86      0.74      0.80      4389
           1       0.77      0.87      0.82      4389

    accuracy                           0.81      8778
   macro avg       0.82      0.81      0.81      8778
weighted avg       0.82      0.81      0.81      8778

test set
              precision    recall  f1-score   support

           0       0.95      0.74      0.83      1848
           1       0.24      0.68      0.35       218

    accuracy                           0.73      2066
   macro avg       0.59      0.71      0.59      2066
weighted avg       0.88      0.73      0.78      2066
```



Conclusion

The highest accuracy test set and highest AUC are XGBoost model with SMOTE data

Logistic Regression

Modeling from last 2 years data	Undersampling	Oversampling	SMOTE
Test Set Accuracy	0.69	0.70	0.70
Test Set AUC	0.788	0.794	0.793

XGBoost

Modeling from last 2 years data	Undersampling	Oversampling	SMOTE
Test Set Accuracy	0.67	0.68	0.73
Test Set AUC	0.803	0.800	0.803