Predicting Success of Bank Telemarketing Machine Learning Approach

Atul Joshi, Anurag Sharma
2018msbda005@curaj.ac.in, 2018msbda004@curaj.ac.in

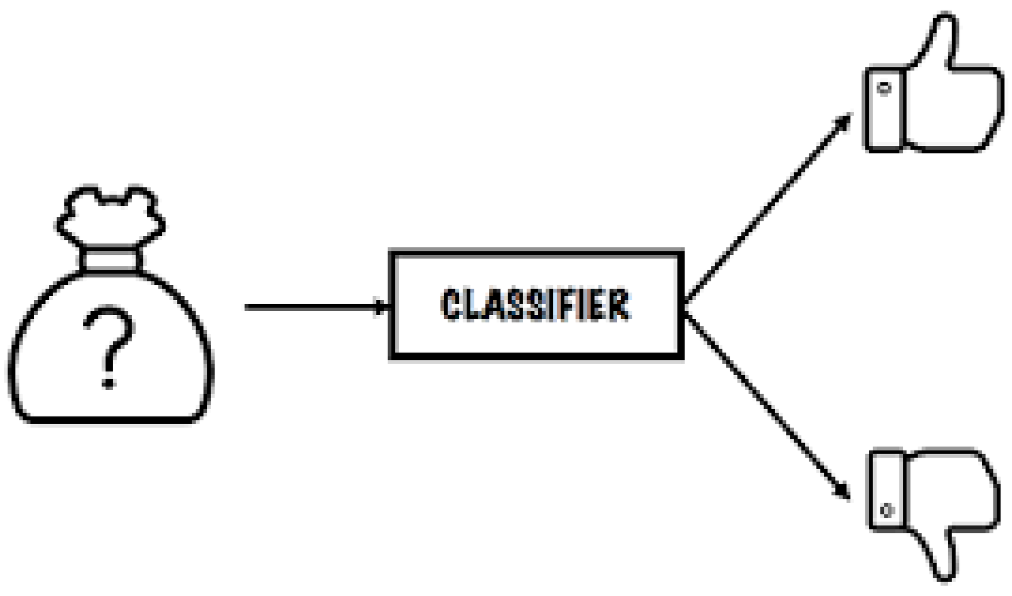
Problem

To design a Decision Support System to predict the success of bank telemarketing calls for selling Long Term Deposits .

Objective: Selecting the best set of clients or targeting the right segments of customers, i.e., those who are more likely to subscribe a product.

The given problem can be posed as a Classification problem(Supervised Learning)

Classification Goal: Assigning predicted/ dependent variable **Y**, values Yes/No, based on whether a client has subscribed a term deposit or not .



Dataset Description

Data is related to direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls.
Variables: 20
Observations: 41188
Numeric: 8(2 were removed)
Categorical: 10
Independent Variables are based on Client, Current campaign, Previous Campaign and Socio-economic context .
High Correlation:
nr.employed, euribor3m
euribor3m, emp.var.rate
hence, euribor3m and nr.employed removed

Models

Logistic Regression (Self developed and sklearn)
Decision Trees(IDB Tree)
Naive Bayes (GaussianNB)
Neural Networks (MLP Classifier)

References

[1] [Moro et al., 2014] S. Moro, P. Cortez and P. Rita. :A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014

[2] David Arnott, Graham Pervan, Eight key issues for the decision support systems discipline, Decision Support Systems 44 (3) (2008) .

[3] David L. Olson, Dursun Delen, Yanyan Meng, Comparative analysis of data mining methods for bankruptcy prediction, Decision Support Systems 52 (2) (2012).

Acknowledgements

This project would not have been possible without the kind support and guidance of our faculty. I would like to extend my sincere thanks to all of them.

Thanks to Mr. S. K. Chaudhary for his guidance and constant supervision as well as for providing necessary information regarding the project also for his support in completing the project.

Comparing Results from the Trained Models

Classification Metrics for Naive Bayes

```
print(classification_report(ytest_num,y_predNB,target_names= target_names))
```

	precision	recall	f1-score	support
No	0.93	0.90	0.92	7345
Yes	0.37	0.48	0.42	893
micro avg	0.85	0.85	0.85	8238
macro avg	0.65	0.69	0.67	8238
weighted avg	0.87	0.85	0.86	8238

Classification Metrics for sklearn LR

```
target_names = ['No', 'Yes']
print(classification_report(ytest_num, ypredLR, target_names=target_names))
```

	precision	recall	f1-score	support
No	0.93	0.98	0.95	7345
Yes	0.68	0.38	0.49	893
micro avg	0.91	0.91	0.91	8238
macro avg	0.81	0.68	0.72	8238
weighted avg	0.90	0.91	0.90	8238

Classification Metrics for MLP Classifier

```
print(classification_report(ytest_num,ypredMLP,target_names=target_names))
```

	precision	recall	f1-score	support
No	0.91	0.99	0.95	7345
Yes	0.72	0.19	0.30	893
micro avg	0.90	0.90	0.90	8238
macro avg	0.82	0.59	0.62	8238
weighted avg	0.89	0.90	0.88	8238

Classification Metrics for self developed LR

```
print(classification_report(ytest_num, finalout, target_names=target_names))
```

	precision	recall	f1-score	support
No	0.92	0.98	0.95	7345
Yes	0.64	0.34	0.45	893
micro avg	0.91	0.91	0.91	8238
macro avg	0.78	0.66	0.70	8238
weighted avg	0.89	0.91	0.90	8238

Classification Metrics for Decision Tree Classifier

```
print(classification_report(ytest_num,ypredDT,target_names=target_names))
```

	precision	recall	f1-score	support
No	0.94	0.94	0.94	7345
Yes	0.50	0.54	0.52	893
micro avg	0.89	0.89	0.89	8238
macro avg	0.72	0.74	0.73	8238
weighted avg	0.90	0.89	0.89	8238

ROC - AUC SCORE

```
roc_auc_score(ytest_num,probSLR)
```
0.8706899876470297

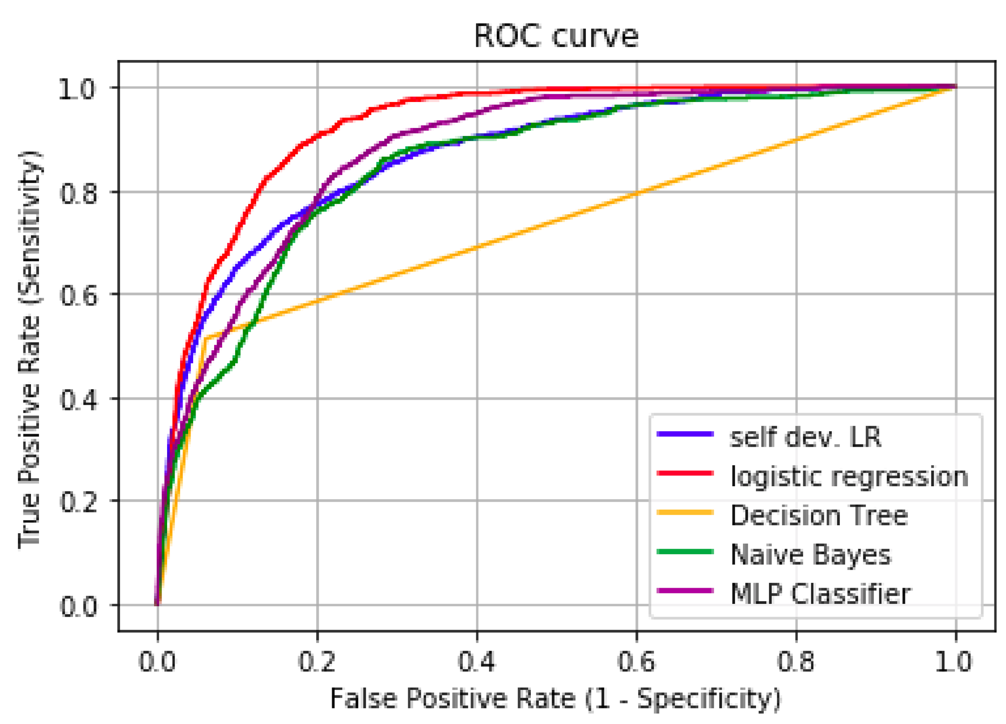
```
roc_auc_score(ytest_num,probNB)
```
0.847934273678733

```
roc_auc_score(ytest_num,probLR)
```
0.9234242479712003

```
roc_auc_score(ytest_num,probDT)
```
0.7259488359081159

```
roc_auc_score(ytest_num,probMLP)
```
0.8783862738122785

l0.3



Accuracy scores

Self developed Logistic Regression

```
accuracy_score(ytest_num,finalout)
```
0.9015537751881525

sklearn Logistic Regression

```
accuracy_score(ytest_num,ypredLR)
```
0.9073804321437242

Naive Bayes' Classifier

```
accuracy_score(ytest_num,y_predNB)
```
0.8555474629764506

Decision Tree Classifier

```
accuracy_score(ytest_num,ypredDT)
```
0.8914785142024764

MLP Classifier

```
accuracy_score(ytest_num,ypredMLP)
```
0.8810390871570769