

Micro Credit

Submitted by:

Sabuj Ghosh

**ACKNOWLEDGMENT**

* Harsh Ayush from flipnwork.com.
* Various git repositories.
* Stack overflow.
* Geeksforgeeks.

**INTRODUCTION**

* Business Problem Framing

The client is closely related to a mobile service provider who offers loan amount to users for their convenience. But some of the customers see this as an opportunity to take loan amount and do not repay. Based on customer history the job is to predict whether he/she will repay or not.

* Conceptual Background of the Domain Problem

Mobile users often unknowingly behave in a particular pattern. This behaviour include the data consumption habit, call making time as well as how often they recharge how often they get loan amount and so on. The history of these domain is exploited in this problem to find out who is a potential defaulter and who is not.

* Motivation for the Problem Undertaken

The main motivation of this project is to minimize client’s costing from loan defaulters.

**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem

The data has high entropy so branching or decision tree like logic should work the best.

* Data Sources and their formats

Data was provided by www.flipnwork.com in csv format.

* Data Preprocessing Done

Outliers removed.

Scaled and normalized with the standard deviation of the data columns.

Some of the unnecessary columns removed.

Some new features were engneered.

* Data Inputs- Logic- Output Relationships

This part is a bit bleak as none of the features have shown more than 20% correlation with the target so we have to be satisfied with mathematical abstraction.

* State the set of assumptions (if any) related to the problem under consideration

Unsupervised clustering should reveal favouritism between target classes.

* Hardware and Software Requirements and Tools Used

sklearn

pandas

imbalance learn

numpy

matplotlib

seaborn

RAM constraint refrained me from clustering and Gird search cv as well as randomized search cv.

**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

Imbalance in target variable was solved by under sampling as over sampling resulted in overfitting.

Decision tree performed better than naive Bayes and logistic regression.

Used ensemble approach by using Random forest.

* Testing of Identified Approaches (Algorithms)

Precision is preferred with more weight compared to recall as the company needs to identify defaulters before they defaulted to stop loss

* Run and Evaluate selected models

ovs=RandomUnderSampler()

X\_res,y\_res=ovs.fit\_resample(x,y)

X\_train,X\_test,y\_train,y\_test=train\_test\_split(X\_res,y\_res)

models=[lr,tree,nb,rf]

for model in models:

model.fit(X\_train,y\_train)

print('---'\*30,'\n',model,'\n','---'\*30)

print(model.score(X\_test,y\_test))

print(confusion\_matrix(model.predict(X\_test),y\_test))

print(classification\_report(model.predict(X\_test),y\_test))

* Key Metrics for success in solving problem under consideration

As this is a classification problem we observed precision, recall and f1 score. Now our main focus was on the precision of ‘0’ labels. As just declaring everyone a non-defaulter gains a 87% accuracy but this is fruitless. Also the same this can affect the f1 score also as it gives very high precision and recall for ‘1’ labels.

* Visualizations

We mainly used scaled and normalized independent variable in suitable bins and then did a count plot. It revealed skewness in certain variables. Also some values of independent variables had no ‘0’ labels while other values did have ‘0’ labels. This information was leveraged to engineer new features.

for c in graph\_col:

data[c]=scale(data[c])

bi=[-10,-3,-2,-1.5,-1,-0.8,-0.6,-0.4,- 0.2,0,0.2,0.4,0.6,0.8,1.0,1.5,2,3,10]

data['binned']=pd.cut(x=data[c], bins=bi)

plt.figure(figsize=(18,5))

ax=sns.countplot(x='binned',hue='label',data=data)

ax.set(xlabel=c, ylabel='count')

plt.show()

* Interpretation of the Results

The pattern revealed is not plainly laid, the graphical representation of the tree reveals a few critical values of features that can be avoided by the client or at least be regarded as red flag. But when we attain higher accuracy through making ensemble of the trees into a forest we lose that leverage and end up with a sheer mathematical abstraction which can spit out whether a customer can default or not but with higher accuracy.

**CONCLUSION**

* Key Findings and Conclusions of the Study

The month of August has no defaulters.

Through unsupervised clustering it is possible to segregate the non-defaulters.

In this problem precision is more important than any other metric.

* Learning Outcomes of the Study in respect of Data Science

Unsupervised learning can give an edge in supervised learning.

Target imbalance can make the entire concept of accuracy obsolete.

Scaling is important.

With huge number of data set it is difficult to have hyper-parameter tuning via grid search cv or randomized search cv. Under these condition one needs to rely on reasoning.

* Limitations of this work and Scope for Future Work

Needed a higher computing power than my pc.