FLIP ROBO

Micro-Credit Defaulter Model

Submitted by:

Priyanka Priyadarshni Pradhan

ACKNOWLEDGMENT

This includes mentioning of all the references, research papers, data sources, professionals and other resources that helped you and guided you in completion of the project.

INTRODUCTION

- Business Problem Framing

  Build a model which can be used to predict in terms of a probability for each loan transaction, whether the customer will be paying back the loaned amount within 5 days of insurance of loan.

- Conceptual Background of the Domain Problem

  Unable to identify cases where defaulters case and based on the transaction and specific criteria having a better pattern understanding to make informed decision

- Motivation for the Problem Undertaken

  Classic case of machine learning classification dependent on pattern analysis.

Analytical Problem Framing

.

- ## Data Sources and their formats

Variable	Definition	Comment
label	Flag indicating whether the user paid back the credit amount within 5 days of issuing the loan{1:success, 0:failure}	
msisdn	mobile number of user	
aon	age on cellular network in days	
daily_decr30	Daily amount spent from main account, averaged over last 30 days (in Indonesian Rupiah)	
daily_decr90	Daily amount spent from main account, averaged over last 90 days (in Indonesian Rupiah)	
rental30	Average main account balance over last 30 days	Unsure of given definition
rental90	Average main account balance over last 90 days	Unsure of given definition
last_rech_date_ma	Number of days till last recharge of main account	
last_rech_date_da	Number of days till last recharge of data account	
last_rech_amt_ma	Amount of last recharge of main account (in Indonesian Rupiah)	
cnt_ma_rech30	Number of times main account got recharged in last 30 days	
fr_ma_rech30	Frequency of main account recharged in last 30 days	Unsure of given definition
sumamnt_ma_rech30	Total amount of recharge in main account over last 30 days (in Indonesian Rupiah)	
medianamnt_ma_rech30	Median of amount of recharges done in main account over last 30 days at user level (in Indonesian Rupiah)	
medianmarechprebal30	Median of main account balance just before recharge in last 30 days at user level (in Indonesian Rupiah)	
cnt_ma_rech90	Number of times main account got recharged in last 90 days	
fr_ma_rech90	Frequency of main account recharged in last 90 days	Unsure of given definition
sumamnt_ma_rech90	Total amount of recharge in main account over last 90 days (in Indonasian Rupiah)	
medianamnt_ma_rech90	Median of amount of recharges done in main account over last 90 days at user level (in Indonasian Rupiah)	
medianmarechprebal90	Median of main account balance just before recharge in last 90 days at user level (in Indonasian Rupiah)	
cnt_da_rech30	Number of times data account got recharged in last 30 days	
fr_da_rech30	Frequency of data account recharged in last 30 days	
cnt_da_rech90	Number of times data account got recharged in last 90 days	
fr_da_rech90	Frequency of data account recharged in last 90 days	
cnt_loans30	Number of loans taken by user in last 30 days	
amnt_loans30	Total amount of loans taken by user in last 30 days	
maxamnt_loans30	maximum amount of loan taken by the user in last 30 days	There are only two options: !
medianamnt_loans30	Median of amounts of loan taken by the user in last 30 days	
cnt_loans90	Number of loans taken by user in last 90 days	
amnt_loans90	Total amount of loans taken by user in last 90 days	
maxamnt_loans90	maximum amount of loan taken by the user in last 90 days	

B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
label	msisdn	aon	daily_decr	daily_decr	rental30	rental90	last_rech_	last_rech_	last_rech_	cnt_ma_re	fr_ma_rec	sumamnt_	medianam	medianma	cnt_ma_re	fr_ma_rec	sumamnt_	medianam	medianma	cnt_da_re
0	21408I707	272	3055.05	3065.15	220.13	260.13	2	0	1539	2	21	3078	1539	7.5	2	21	3078	1539	7.5	0
1	76462I703	712	12122	12124.75	3691.26	3691.26	20	0	5787	1	0	5787	5787	61.04	1	0	5787	5787	61.04	0
1	17943I703	535	1398	1398	900.13	900.13	3	0	1539	1	0	1539	1539	66.32	1	0	1539	1539	66.32	0
1	55773I707	241	21.228	21.228	159.42	159.42	41	0	947	0	0	0	0	0	1	0	947	947	2.5	0
1	03813I827	947	150.6193	150.6193	1098.9	1098.9	4	0	2309	7	2	20029	2309	29	8	2	23496	2888	35	0
1	35819I707	568	2257.363	2261.46	368.13	380.13	2	0	1539	4	10	6156	1539	15.4	8	0	11744	1539	55.9	0
1	96759I844	545	2876.642	2883.97	335.75	402.9	13	0	5787	1	0	5787	5787	277.8	1	0	5787	5787	277.8	0
1	09832I908	768	12905	17804.15	900.35	2549.11	4	55	3178	3	3	10404	3178	36	9	3	26095	3178	36	0
1	59772I844	1191	90.695	90.695	2287.5	2287.5	1	0	1539	4	1	6164	1539	39.9	4	1	6164	1539	39.9	0
1	56331I707	536	29.35733	29.35733	612.96	612.96	11	0	773	1	0	773	773	86.8	1	0	773	773	86.8	0
1	32893I827	1511	12.896	12.896	790.44	790.44	8	0	1539	2	5	2312	1156	16.83	2	5	2312	1156	16.83	0
0	82417I908	82	65.16667	65.16667	326.2	326.2	17	0	7526	2	0	9065	4532.5	489	2	0	9065	4532.5	489	0
1	11435I892	154	227.041	227.041	240.41	240.41	2	0	1547	4	2	19086	4773.5	63	7	30	28979	1720	92	0
1	66580I976	887	55.90933	55.90933	208.8	208.8	2	0	1539	5	5	7703	1539	20.9	7	5	8649	1539	22.9	0
1	63139I703	707	8919	10317.35	399.25	2453.78	3	0	770	3	6	3079	770	66	8	10	19185	1539	52.5	0
0	24075I892	1037	12	12	1216.8	1216.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	82053I853	1583	1000	1000	1000.8	1087.88	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	37204I844	929	10.688	10.688	40	40	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	44217I904	832	14.4	14.4	1660.96	1660.96	1	0	2309	3	26	4618	1539	88.8	3	26	4618	1539	88.8	0
1	19611I908	450	48.935	48.935	726.3	726.3	1	0	1539	2	8	9539	4769.5	12	2	8	9539	4769.5	12	0
1	67813I905	100	769.614	777.46	1050.57	1167.3	6	0	770	5	20	8867	770	168	8	31	14380	771.5	52	0
0	75522I707	378	514.6933	515.2	56.26	58.2	2	0	773	1	0	773	773	542	2	64	1546	773	283.5	0
1	61590I952	463	1540	1541	969.12	969.12	4	0	770	1	0	770	770	43	2	66	1543	771.5	26.5	0
1	95027I908	857	58.02333	58.02333	479.44	479.44	2	0	1539	4	12	6164	1539	115.5	4	12	6164	1539	115.5	0
0	59645I827	966	291.5633	291.5633	-2020.09	-2020.09	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	59102I703	656	63.25	63.25	2855.7	2855.7	1	0	770	15	1	12674	770	254.6	19	1	14566	770	134.6	0
1	44692I905	1179	3703.272	3712.84	340.96	376.42	2	0	770	6	5	5395	771.5	38.5	9	5	7114	773	53.5	0
0	49345I908	871	505.6	508	9276.68	10569.17	16	0	770	1	0	770	770	77.9	1	0	770	770	77.9	0

- ## Data Preprocessing Done

Checked for missing values none, checked for outliers but could not change it as it contain more that 8% of the data

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

Model assumptions

Model/s Development and Evaluation

- Testing of Identified Approaches (Algorithms)

Algorithms used are Logistic Regression, Support Vector Machine, Decision Tree, Neural Network, Random Forest.

- Run and Evaluate selected models

```
names = [
    "Logistic Regression",
    "Support Vector Machine",
    "Decision Tree",
    "Neural Network",
    "Random Forest",
]
models = [
    LogisticRegression(),
    SVC(),
    DecisionTreeClassifier(),
    MLPClassifier(),
    RandomForestClassifier(),
]

accuracy=[]
for model, name in zip(models,names):
    model.fit(X_train, y_train)

    y_pred = model.predict(X_test)
    print('Confusion matrix of ',name)
    print(confusion_matrix(y_test, y_pred))
    ac = accuracy_score(y_test, y_pred)
    print('Accuracy score is ',ac)
    accuracy.append(ac)
    print('='*50)
```

- Visualizations