FLIP ROBO

Micro Credit Defaulter Project

Submitted by:
Sharanya Susheel

ACKNOWLEDGMENT

Firstly, would like to thank Flip Robo for giving me this opportunity and this learning will stay with me for a life time , special thanks to my mentors and career coach at Data Trained who have been my constant source of support and last but not the least would like to thank my family without whose support I would ‘nt be able to finish this project.

Below are the references which helped me in competing my project,
https://www.researchgate.net/publication/265161200_Predicting_Credit_Default_among_Micro_Borrowers_in_Ghana

http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

https://www.researchgate.net/publication/265161200_Predicting_Credit_Default_among_Micro_Borrowers_in_Ghana

https://www.mdpi.com/2227-9091/9/3/50/pdf

INTRODUCTION

- Business Problem Framing

  Technology can save time and effort and make many process efficient. Inability of MFIs to reach the target group of customers to provide various loans, was not only time consuming but also it becomes hard in recovery and more than anything differentiation between good and bad loan repayment customers was a challenge for lack of data being available at the right time. Here is where the telecom industry has the advantage to use technology for real time transaction monitoring and rating, and classification of defaulters and non-defaulters based on MFIs rules or other credit scoring methods, and give seamless and transparent service to customers and save time.

- Conceptual Background of the Domain Problem

  a)We need to look at grievance handling

  b) We need to see the quantum of risk that is allowed per customer

  c) Customer reference mapping required or not

  d) Interest rates calculation based on customer repayment or fixed

  e) Add on loans during repayment tenure allowed or not

  f) Incase the customer looses life, who should be liable for pay back

.

- Review of Literature

  Telecom companies are good in what they do and MFIs in what they can do, so it is better they work hand in hand at any given point in time, and avoid any sort of competition as that will only lead to reduced returns for both and thereby stressing the supporting system. MFIs can roll out more and more products, and telecom companies can collect a lot more attributes of the customers and help MFIs to work better and give the desired inputs for decision making. So once the thought is established by the customers, that the loan is provided only to non-defaulters and at the ease which telecom companies are helping MFIs figure it out, it will draw a lot of discipline with the customers and promising customers get loans and fulfil their business needs.

- Motivation for the Problem Undertaken

  Clearly understand the each case scenario with analysing the data carefully and input the system with right advocacy so that the machines can be more smarter and efficient in providing faster MFSs to the end user and especially reduce the machine dependency on the backend for inputs for generic and routine process.

Analytical Problem Framing

- Data Sources and their formats

This problem is about loan defaulters prediction. The sample dataset is provided to us from our client Telecom Industry. They are a fixed wireless telecommunications network provider.

In order to improve the selection of customers for the credit, the client wants some predictions that will help them in further investment and improvement in the selection of their customers.

The data set contains 209593 columns and 37 rows.

Using necessary data, a machine learning model has been built to predict the probabity of each loan transaction,whether the customer will be paying back the loaned amount within 5 days of issuance of loan. In this case, Label '1' indicates that the loan has been payed i.e., non-defaulter, while Label '0' indicates that the loan has not been payed i.e., defaulter.

The project has been divided into two parts,

Part 1

Exploratory Data Analysis and Data Cleaning :

Part 2:

Training a machine learning model

Features Description

- label :Flag indicating whether the user paid back the credit amount within 5 days of is using 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.
- rental90 :Average main account balance over last 90 days.
- 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.
- 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.
- sumamnt_ma_rech90: Total amount of recharge in main account over last 90 days (in Indian Rupee).
- medianamnt_ma_rech90 :Median of amount of recharges done in main account over last 90 days at user level (in Indian Rupee).
- medianmarechprebal90 : Median of main account balance just before recharge in last 90 days at user level (in Indian Rupee).
- 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.
- 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.
- medianamnt_loans90: Median of amounts of loan taken by the user in last 90 days.
- payback30 :Average payback time in days over last 30 days.
- payback90: Average payback time in days over last 90 days.
- pcircle: telecom circle.
- pdate :date.

- Data Pre-processing Done

What were the steps followed for the cleaning of the data?

The steps followed for data cleaning are described as follows:
1)Firstly the information about the dataset was checked and it gave the data shape i.e the total number of rows and columns.
2)The datatypes of all the columns were checked to find out if they are object, integer or float.
3)Duplicate rows were dropped in the dataset
4)Null values were checked using df.isnull().sum() function.
5) After that we check the summary statistics of our dataset. This part tells about the statistics of our dataset i.e. mean, median, max value ,min values and also it tell whether outliers are present in our dataset or not.

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

  We know that this is classification problem so we use accuracy score, classification report and confusion matrix as our evaluation matrix. We also see the AUC score and plot the AUC_ROC curve for our final model.

- Identification of possible problem-solving approaches (methods)

As we know this dataset is imbalance, so we don’t focus much on accuracy score. We see the precision and recall value along with f1_score.
1) First we see the result without doing any sampling technique and for that we use Logistic Regression with K-Fold cross validation and hyper-parameter tuning.
2) We also use Random Forest Classifier as our evaluation model without using hyper-parameter tuning because our dataset is too large and it takes more than hours to give the result.

Testing of Identified Approaches (Algorithms)

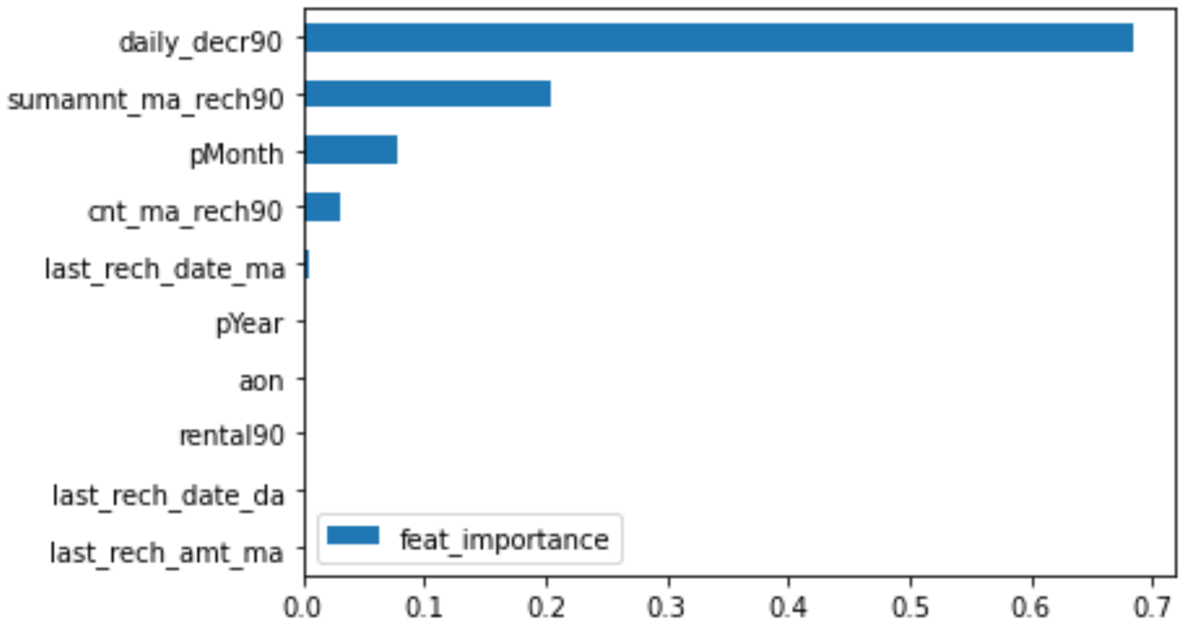
- KNN=KNeighborsClassifier(n_neighbors=10)
- LR=LogisticRegression()
- DT=DecisionTreeClassifier(random_state=20)
- GNB=GaussianNB()
- RF=RandomForestClassifier(
- Run and Evaluate selected models

DescisionTreeclasifier

1)Decision Trees usually mimic human thinking ability while making a decision, so it is easy to understand.

2)The logic behind the decision tree can be easily understood because it shows a tree-like structure.

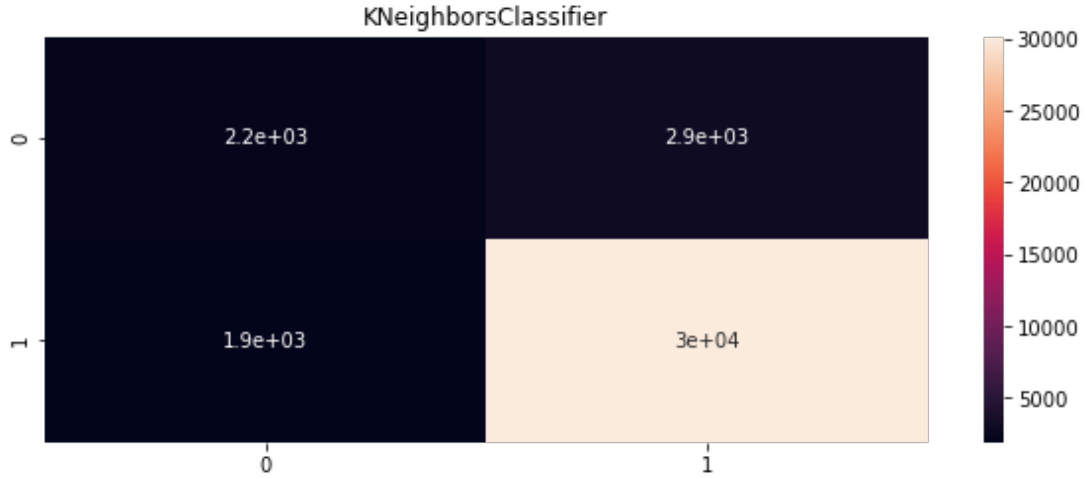
```
dt = DecisionTreeClassifier(max_depth=3)
dt.fit(x, y)
DecisionTreeClassifier(max_depth=3)

dt_features = pd.DataFrame(dt.feature_importances_, index=x.columns, columns=['feat_importance'])
dt_features.sort_values('feat_importance').tail(10).plot.barh()
plt.show()
```

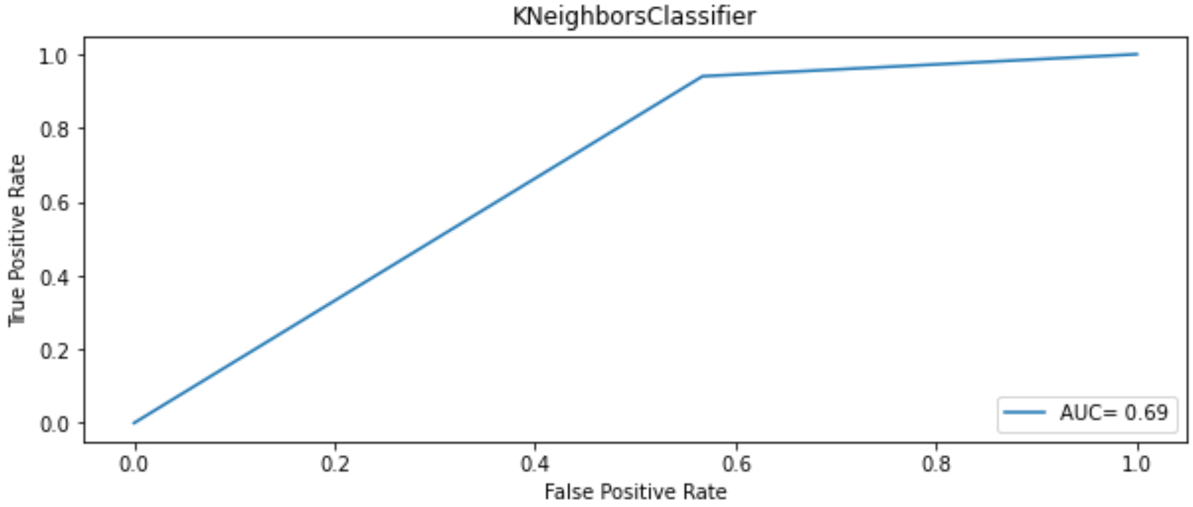


'daily_decr90'is the daily amount of money spent from main account, averaged over last 90 days, this feature helps us to discriminate the data. This feature basically brings insights for company.

KNeighborsClassifier

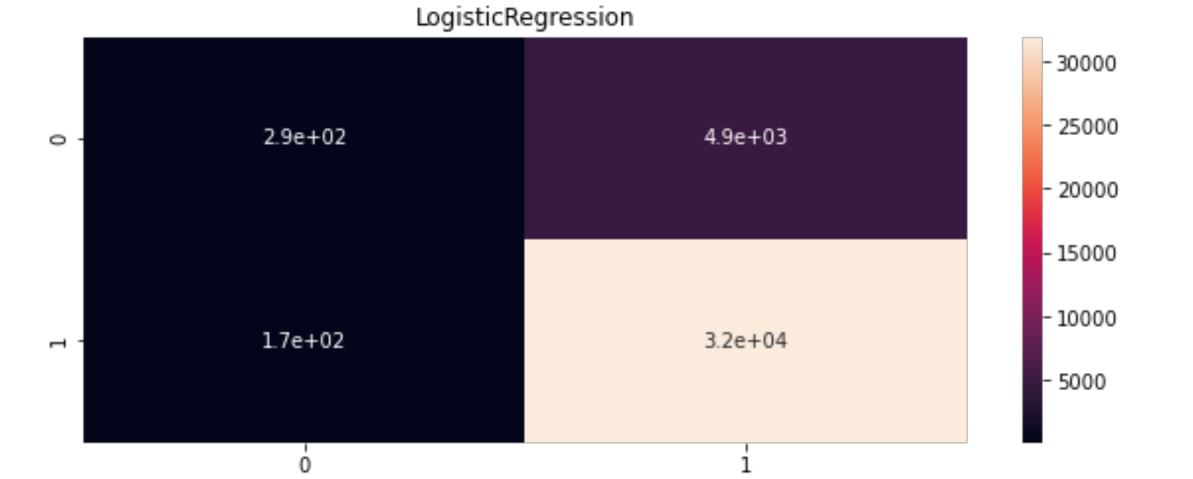
Classifier implementing the k-nearest neighbors vote. The K in the name of this classifier represents the k nearest neighbors, where k is an integer value specified by the user. Hence as the name suggests, this classifier implements learning based on the k nearest neighbors.

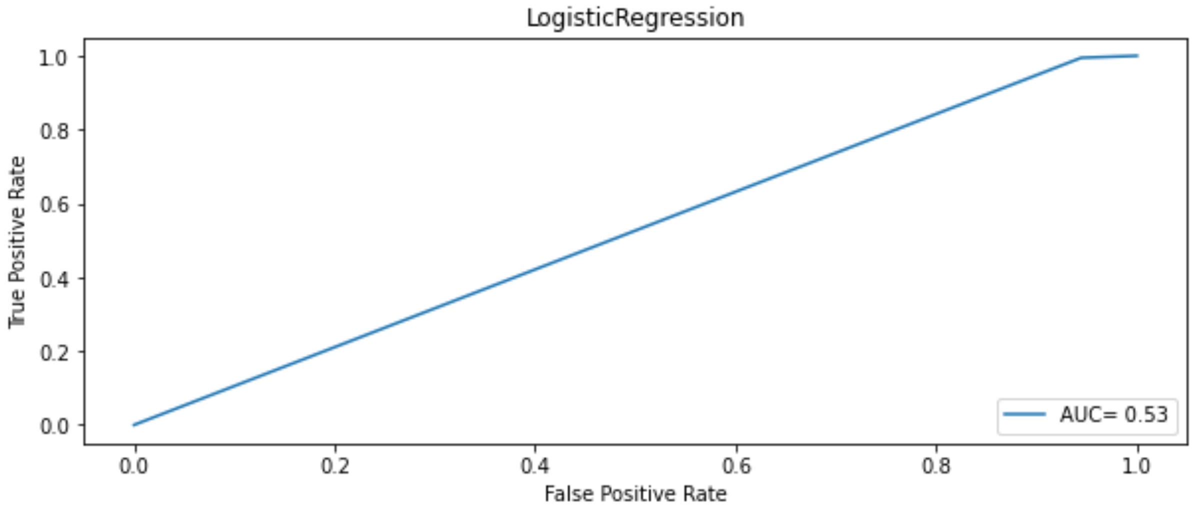




LogisticRegression

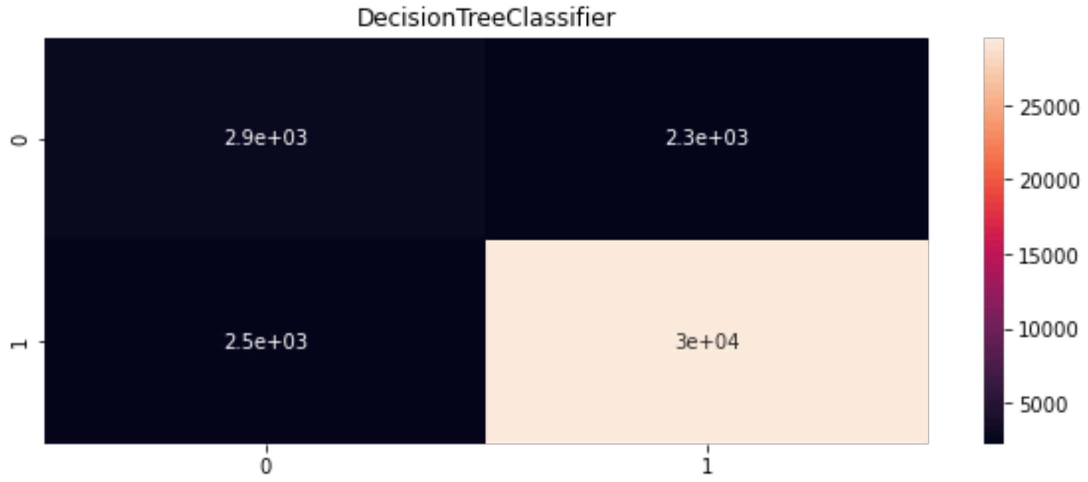
Logistic regression is a statistical analysis method used to predict a data value based on prior observations of a data set. A logistic regression model predicts a dependent data variable by analysing the relationship between one or more existing independent variables.

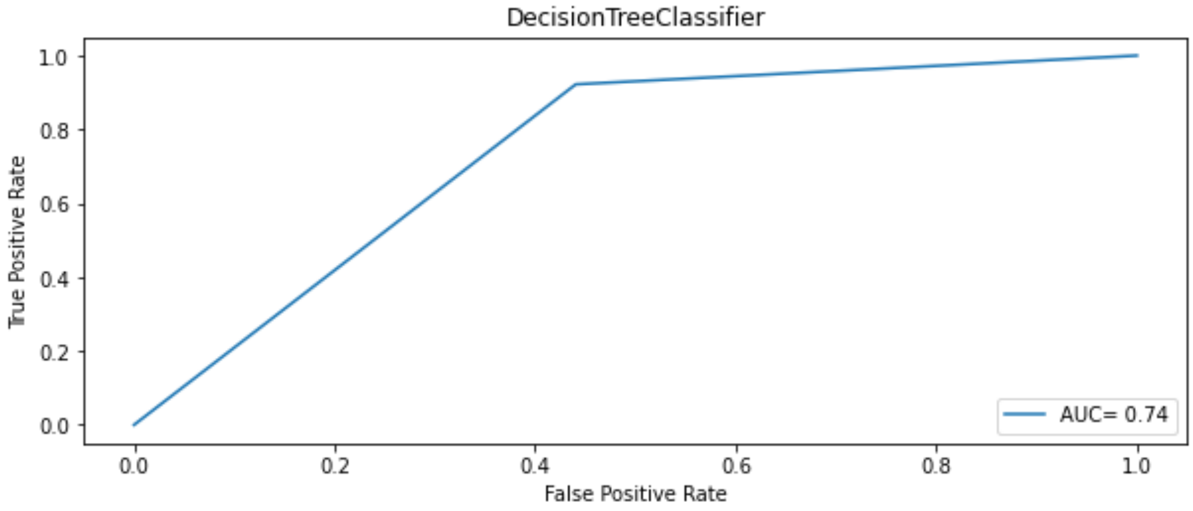




DecisionTreeClassifier

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A tree can be seen as a piecewise constant approximation.





GaussianNB

Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of conditional independence between every pair of features given the value of the class variable. Bayes’ theorem states the following relationship, given class variable y and dependent feature vector x1 through xn, :

P(y|x1,…,xn)=P(y)P(x1,…,xn|y)P(x1,…,xn)
Using the naive conditional independence assumption that

P(xi|y,x1,…,xi−1,xi+1,…,xn)=P(xi|y),
for all i, this relationship is simplified to

P(y|x1,…,xn)=P(y)∏i=1nP(xi|y)P(x1,…,xn)
Since P(x1,…,xn) is constant given the input, we can use the following classification rule:

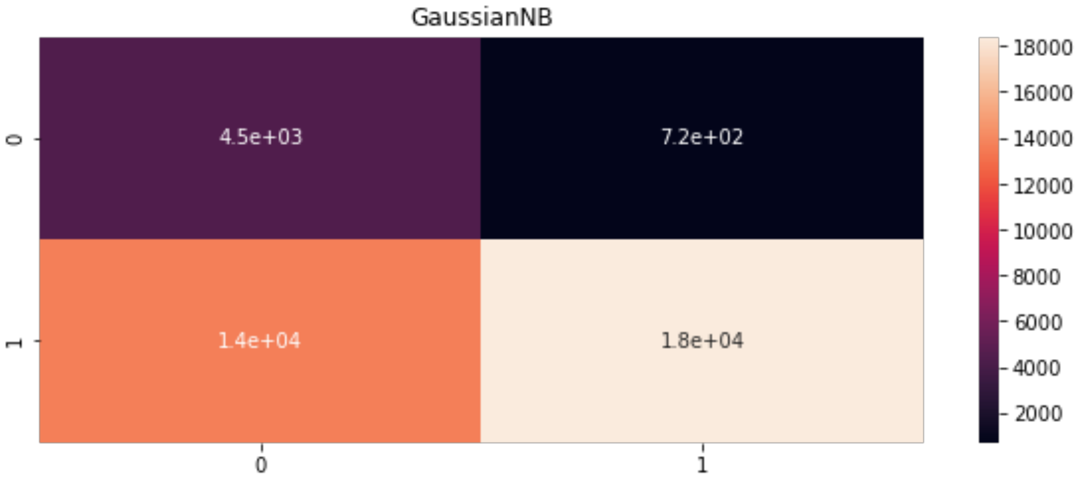
P(y|x1,…,xn)∝P(y)∏i=1nP(xi|y)⇓y^=argmaxyP(y)∏i=1nP(xi|y),
and we can use Maximum A Posteriori (MAP) estimation to
estimate P(y) and P(xi|y); the former is then the relative frequency of class y in the training set.

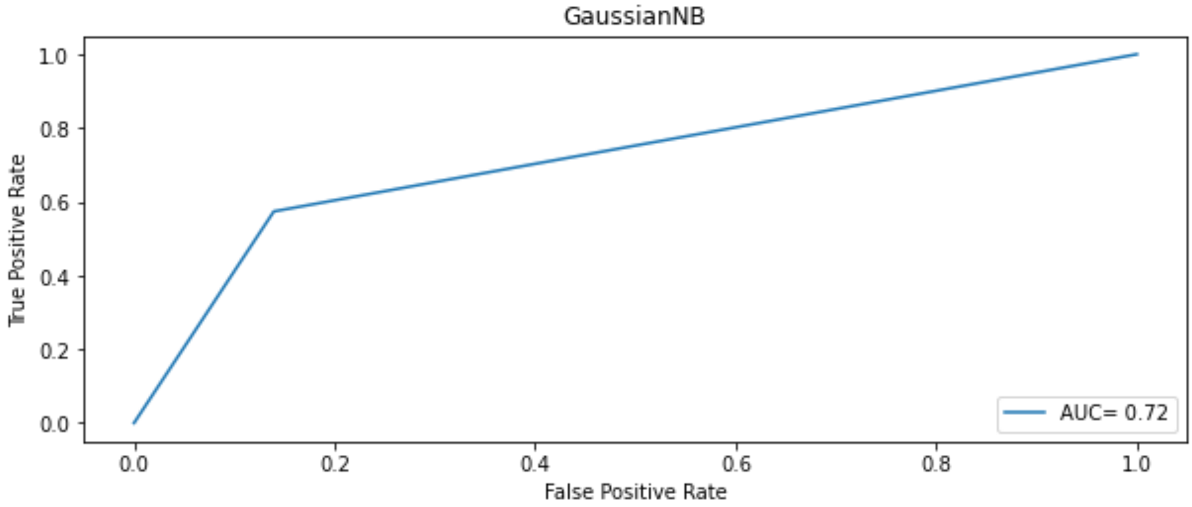
The different naive Bayes classifiers differ mainly by the assumptions they make regarding the distribution of P(xi|y).

In spite of their apparently over-simplified assumptions, naive Bayes classifiers have worked quite well in many real-world situations, famously document classification and spam filtering. They require a small amount of training data to estimate the necessary parameters. (For theoretical reasons why naive Bayes works well, and on which types of data it does, see the references below.)

Naive Bayes learners and classifiers can be extremely fast compared to more sophisticated methods. The decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one dimensional distribution. This in turn helps to alleviate problems stemming from the curse of dimensionality.

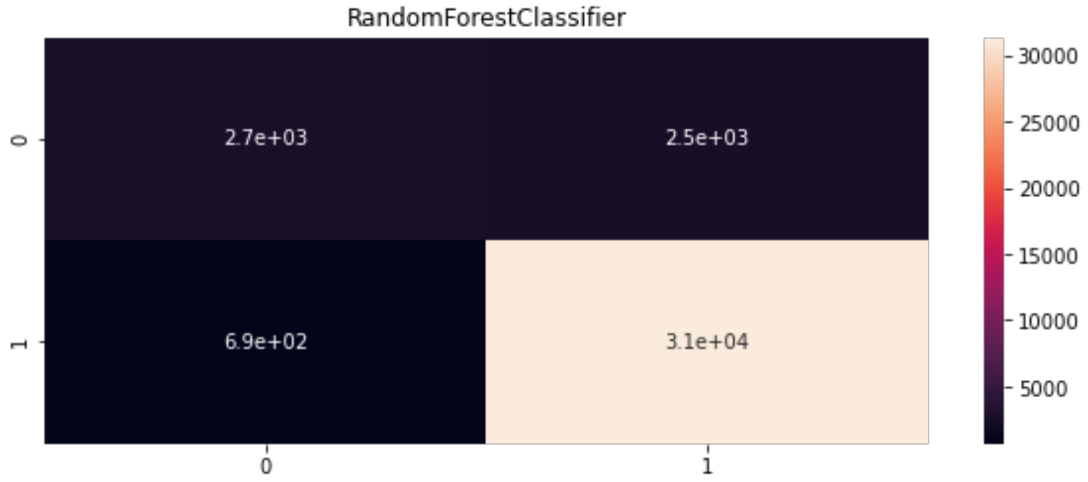
On the flip side, although naive Bayes is known as a decent classifier, it is known to be a bad estimator, so the probability outputs from predict problem are not to be taken too seriously.

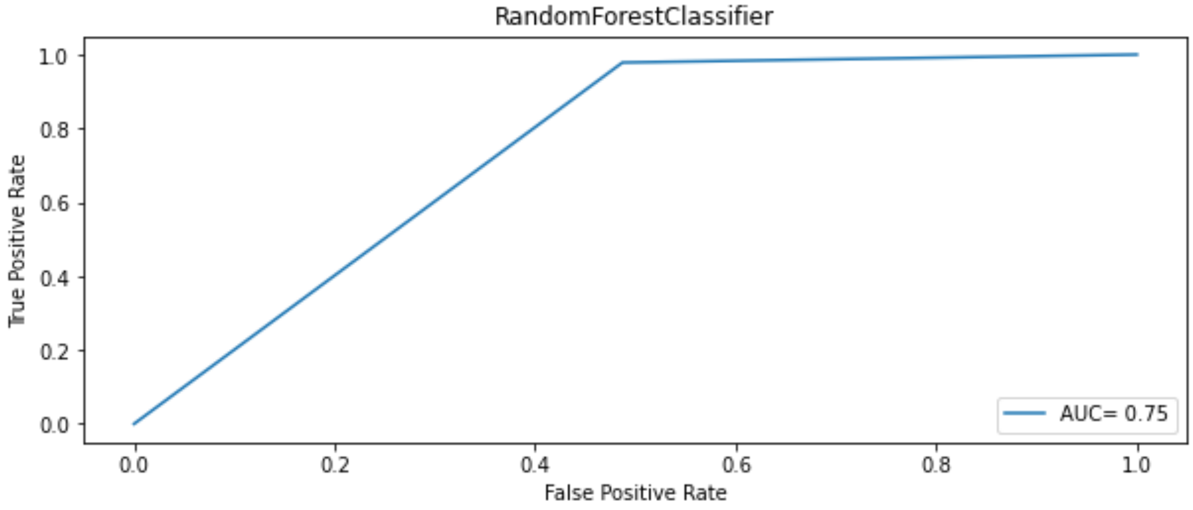




RandomForestClassifier

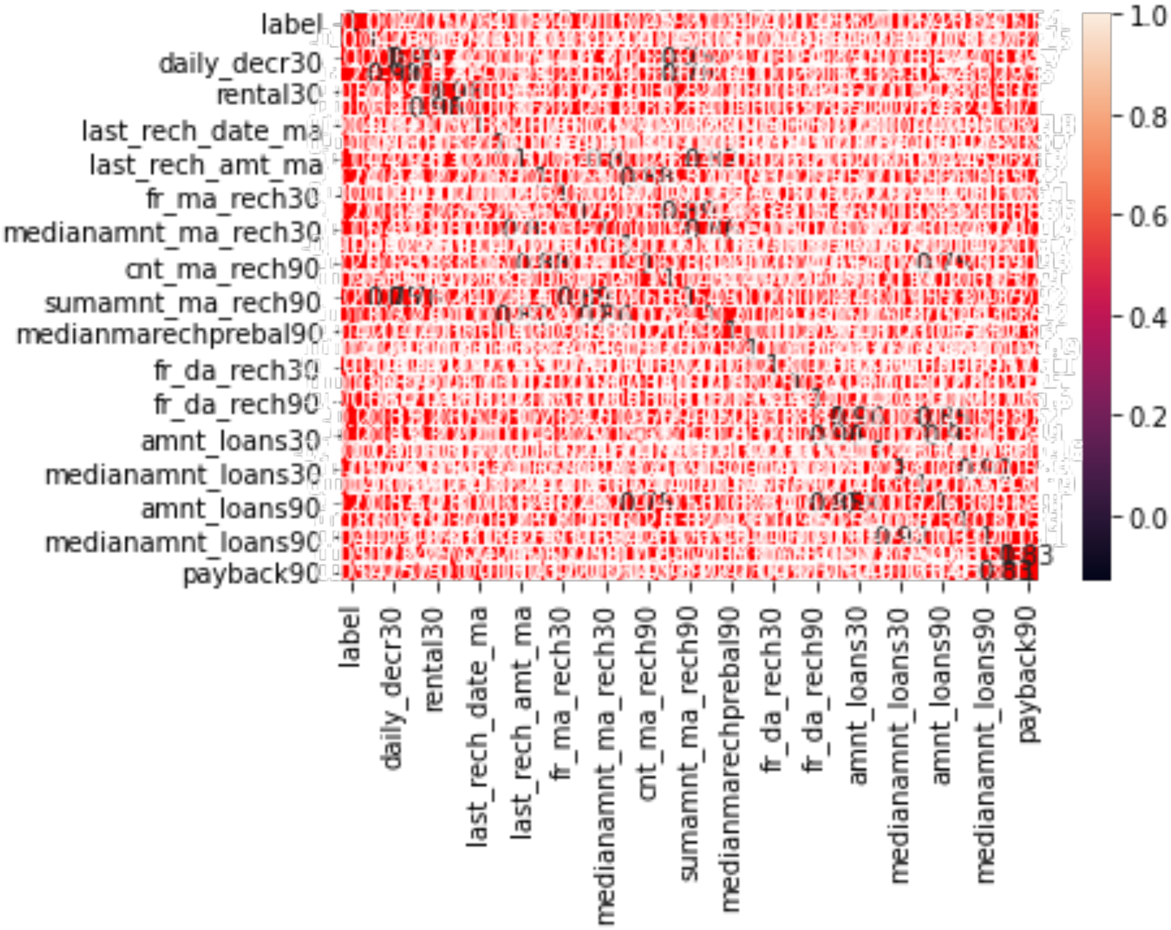
A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is controlled with the max_samplesparameter if bootstrap=True (default), otherwise the whole dataset is used to build each tree.



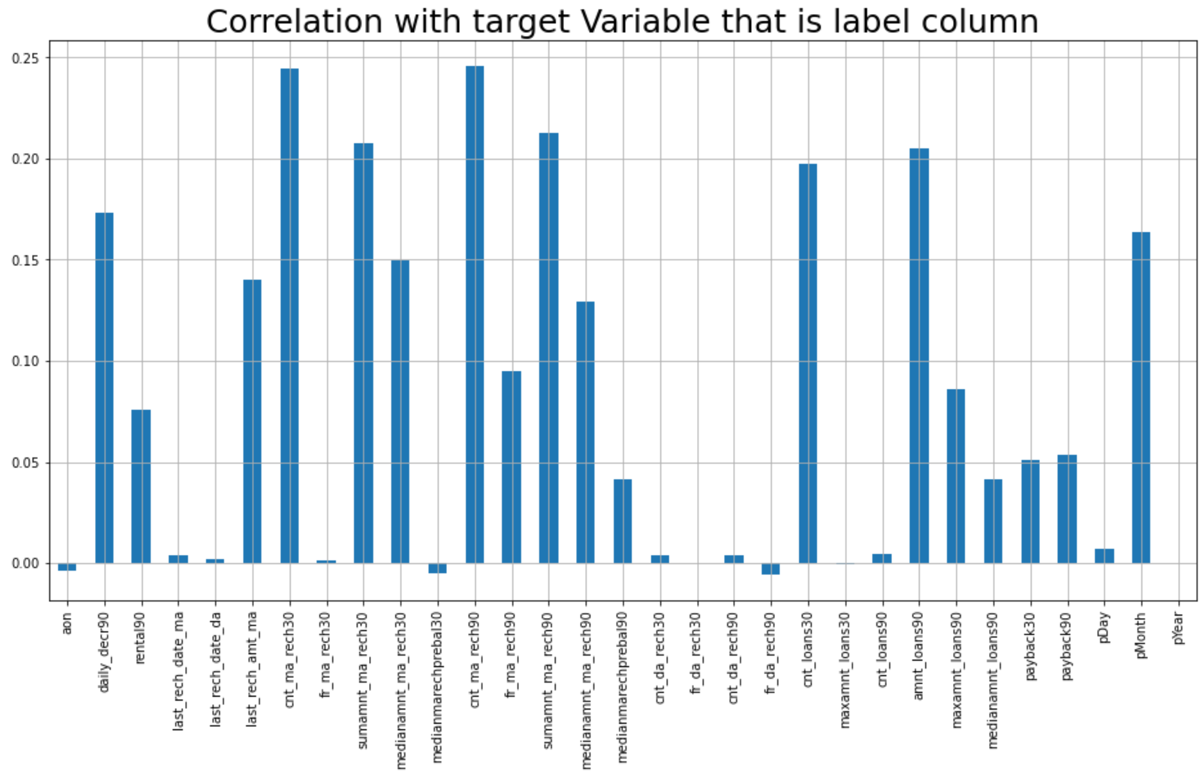


Visualizations

We plot correlation matrix via heatmap to see the correlation of the columns with other columns

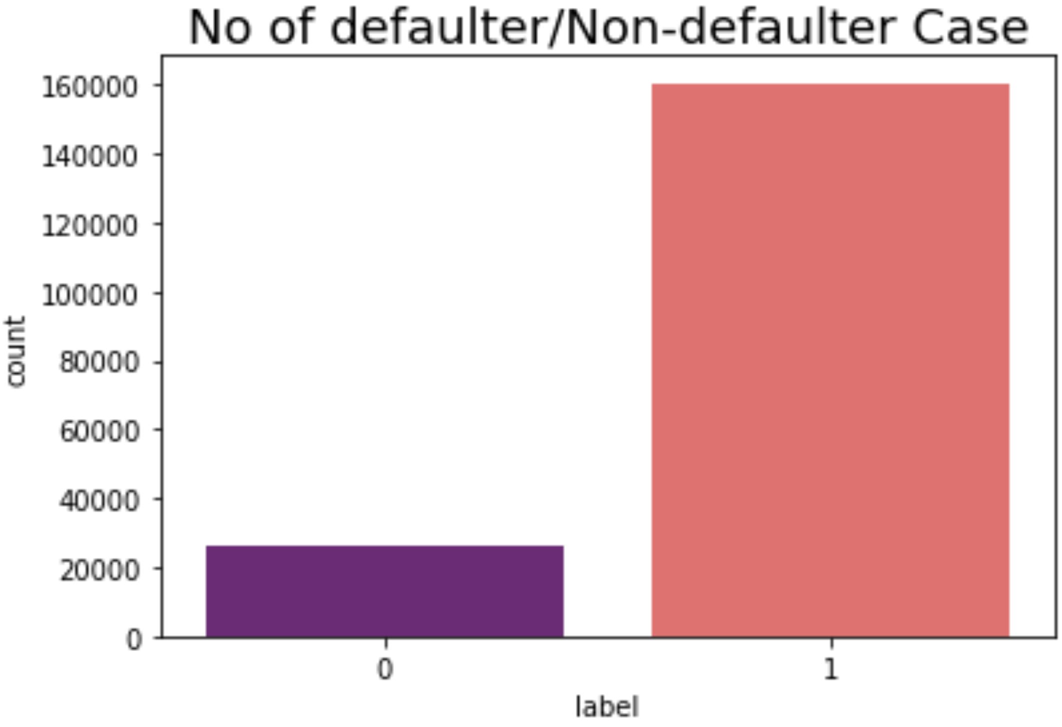


Since the heatmap was not very clear bar graph was used to determine the correlation.



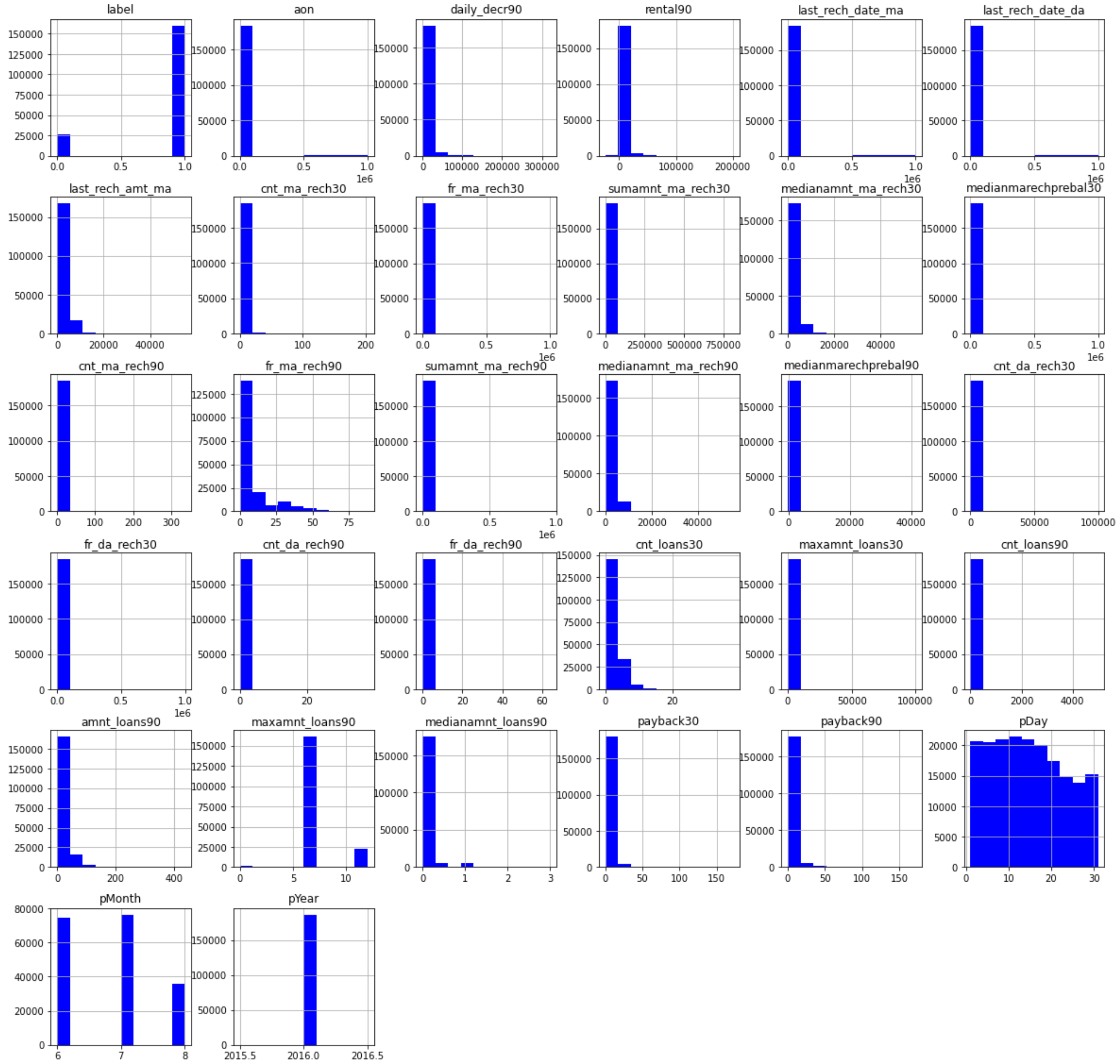
- Here we see the correlation of the columns with respect to the target column that is label.

Countplot gives us the number of defaulter and non-defaulter customers.

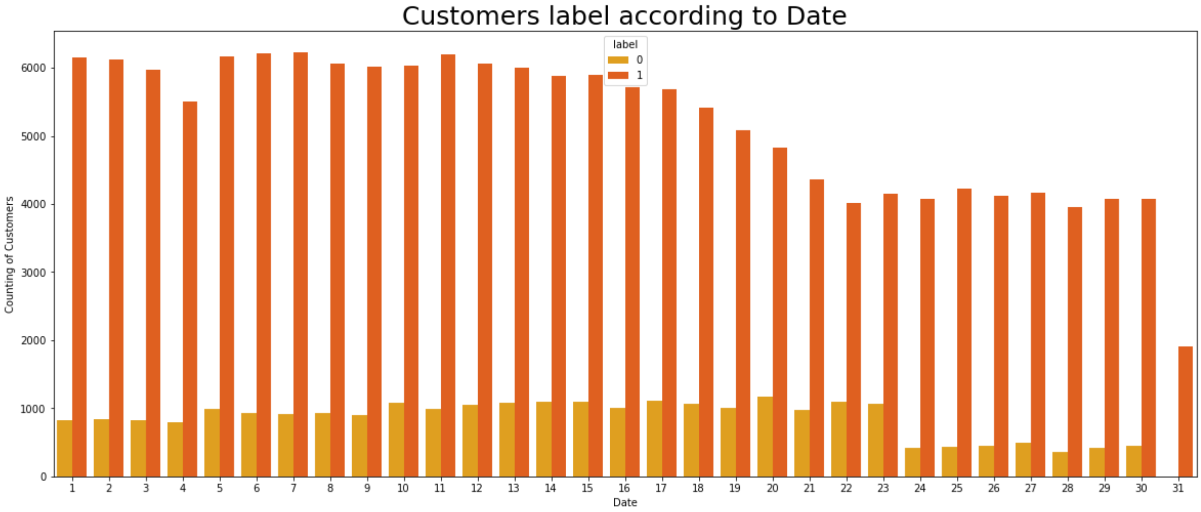


- Label 1 indicates loan has been payed i.e. Non-Defaulter and label 0 indicates that the loan has not been payed i.e. defaulter.

Histogram is used to determine the spread of continuous sample data.

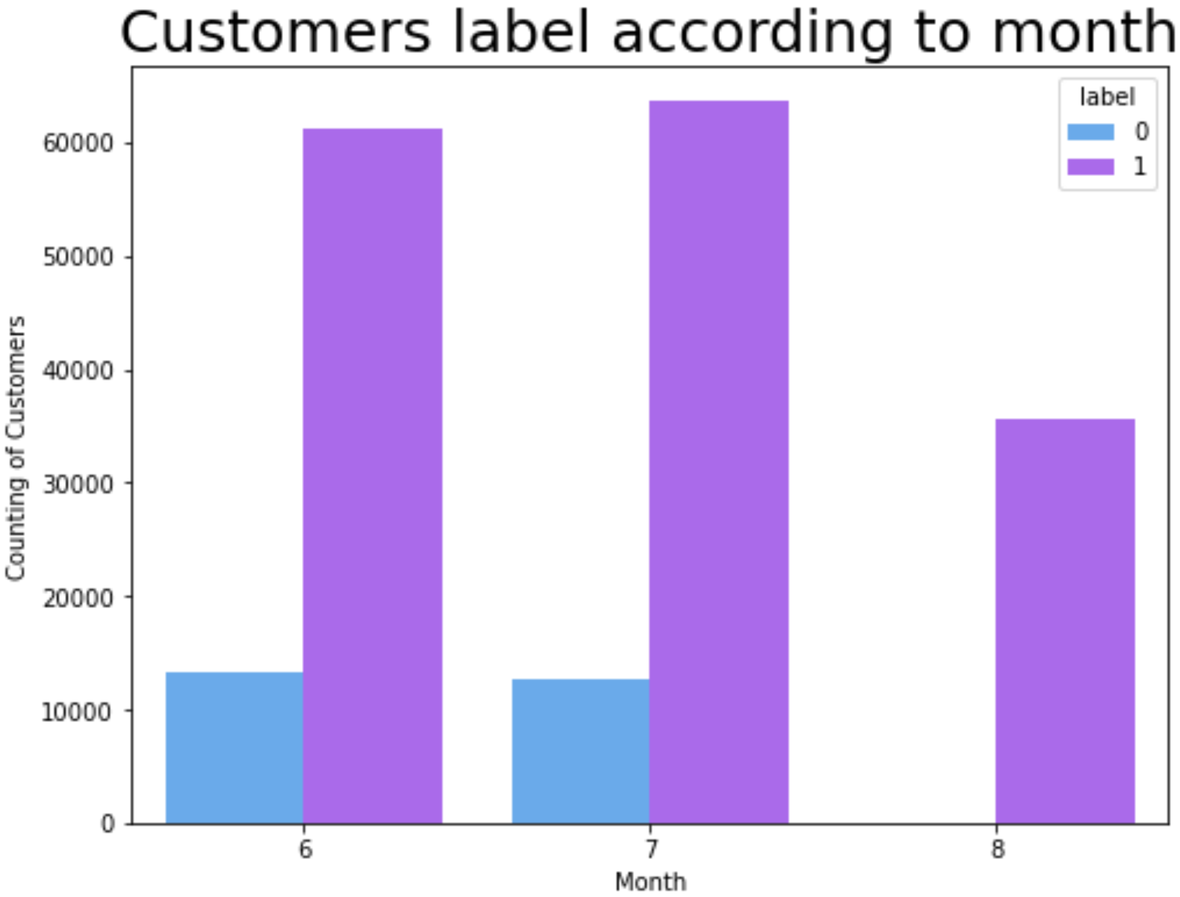


Countplot is used for customer label according to date



From the above graph we can draw information about the customers who did not pay their loans from date 10 to 23.

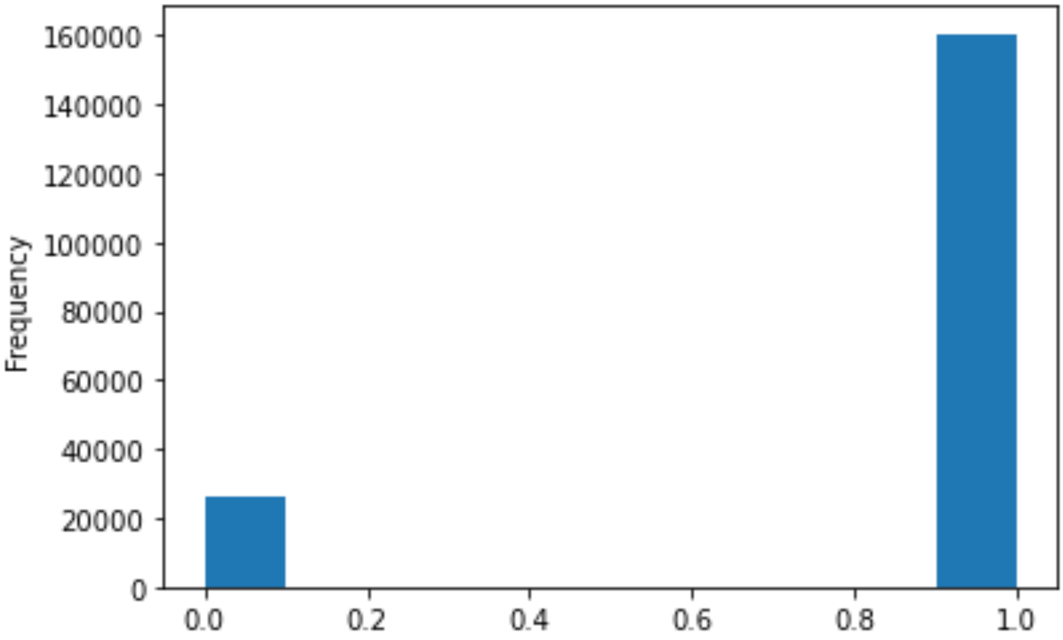
Countplot is used for customer label according to month.



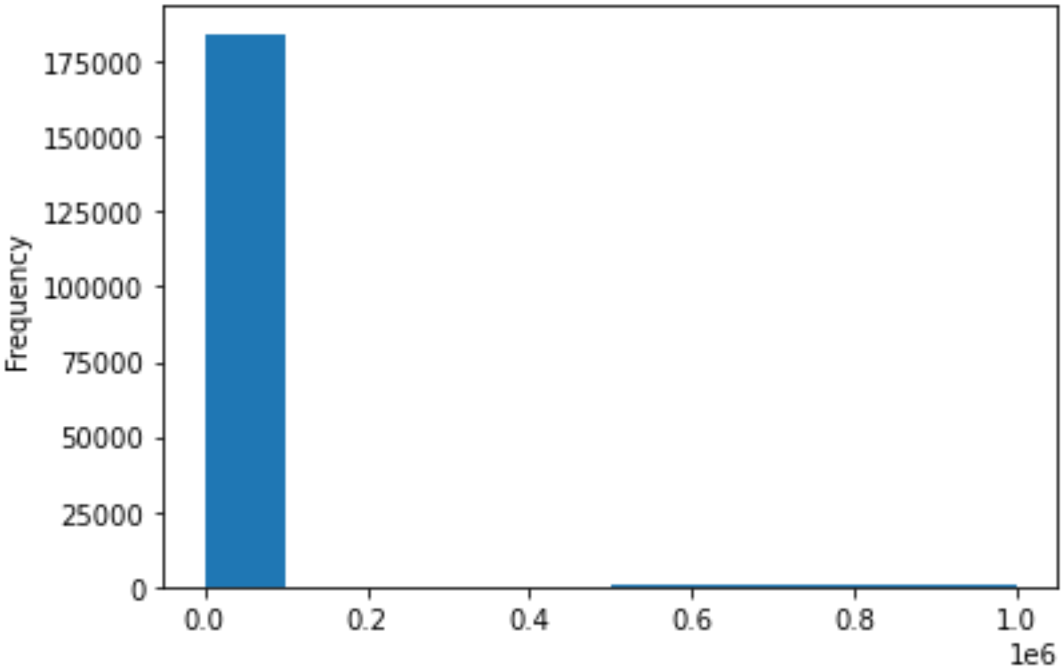
From the above graph we get to know the number of customers who did not pay their loans in the month of June and July.

Histogram was used to determine skewness of all the columns.

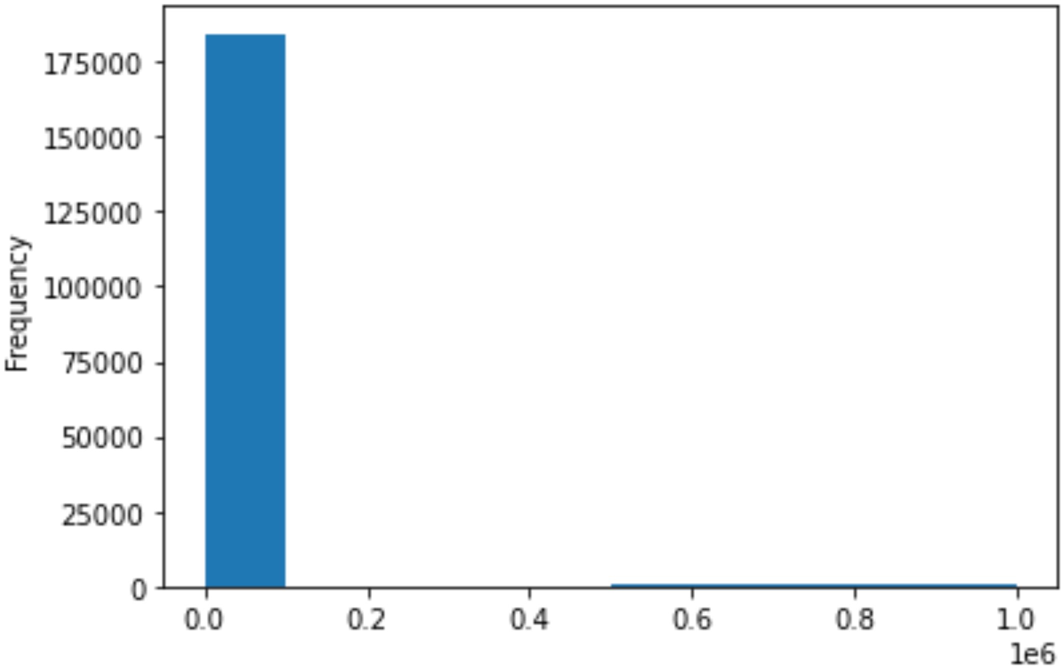
‘label’



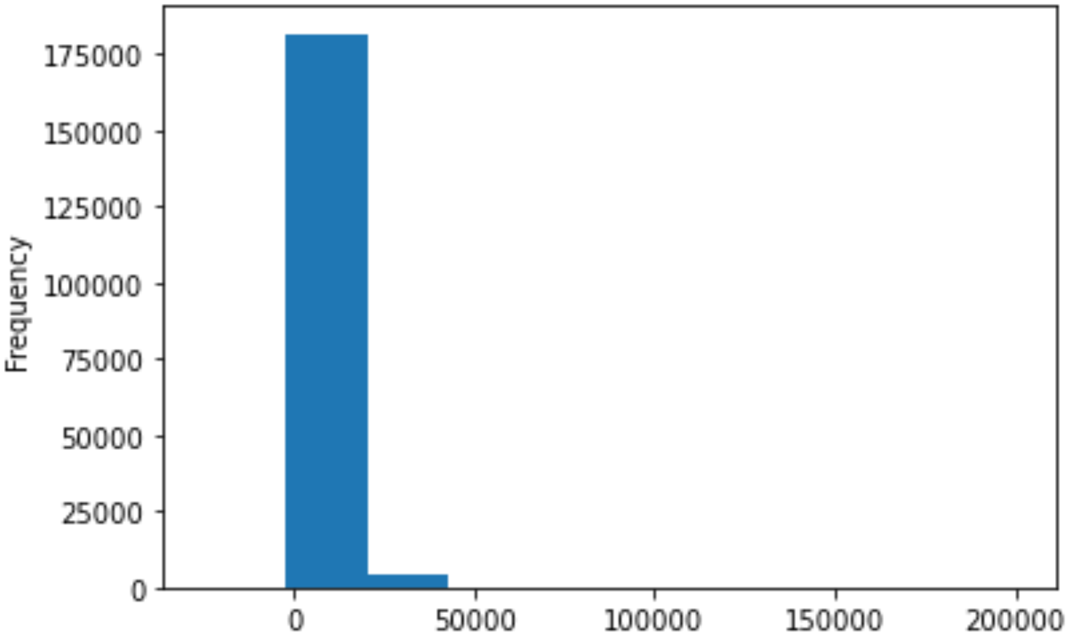
‘aon’



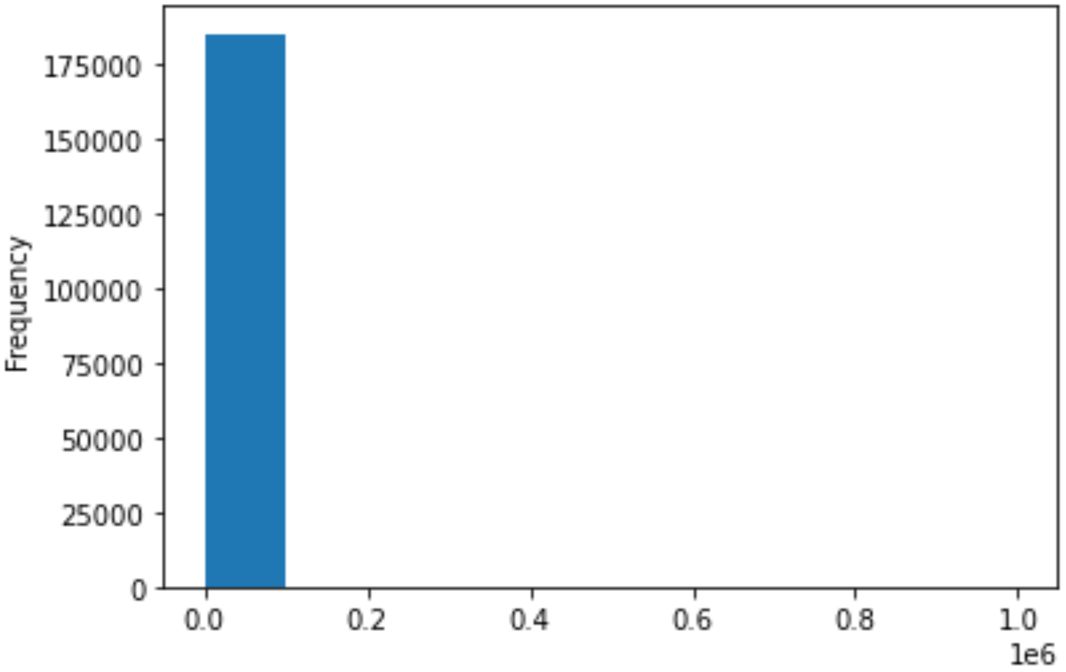
‘daily_decr90’



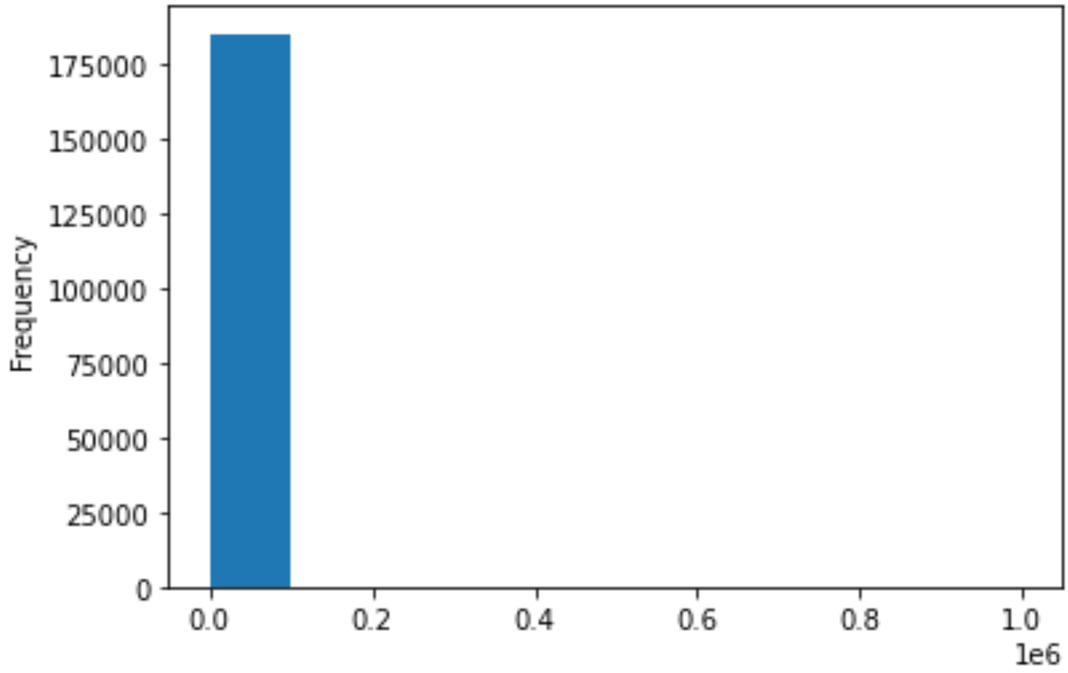
‘rental90’



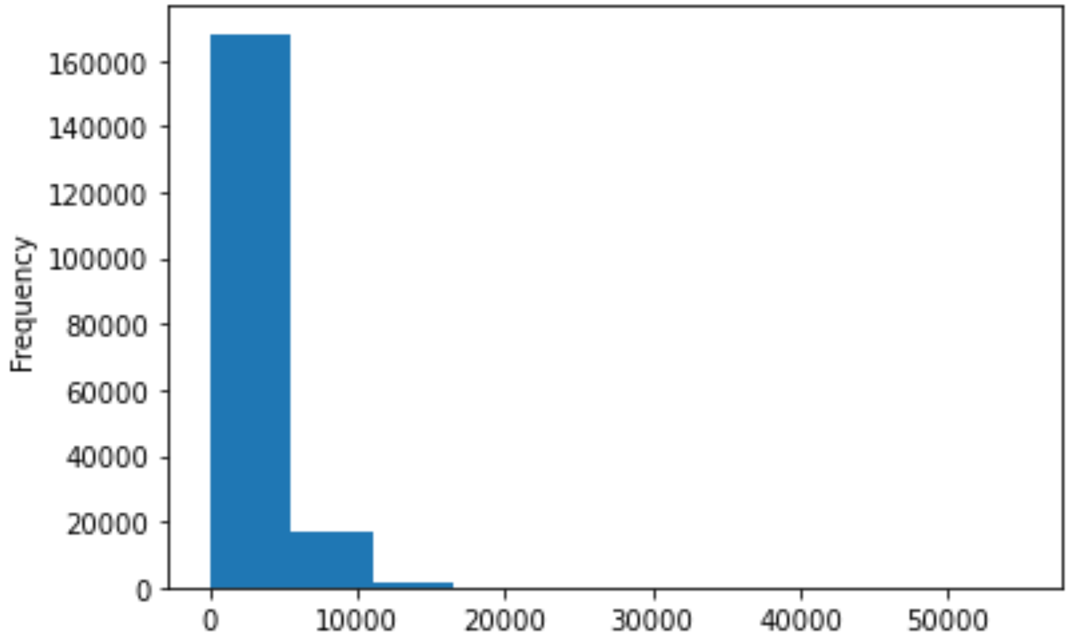
‘last_rech_date_ma’



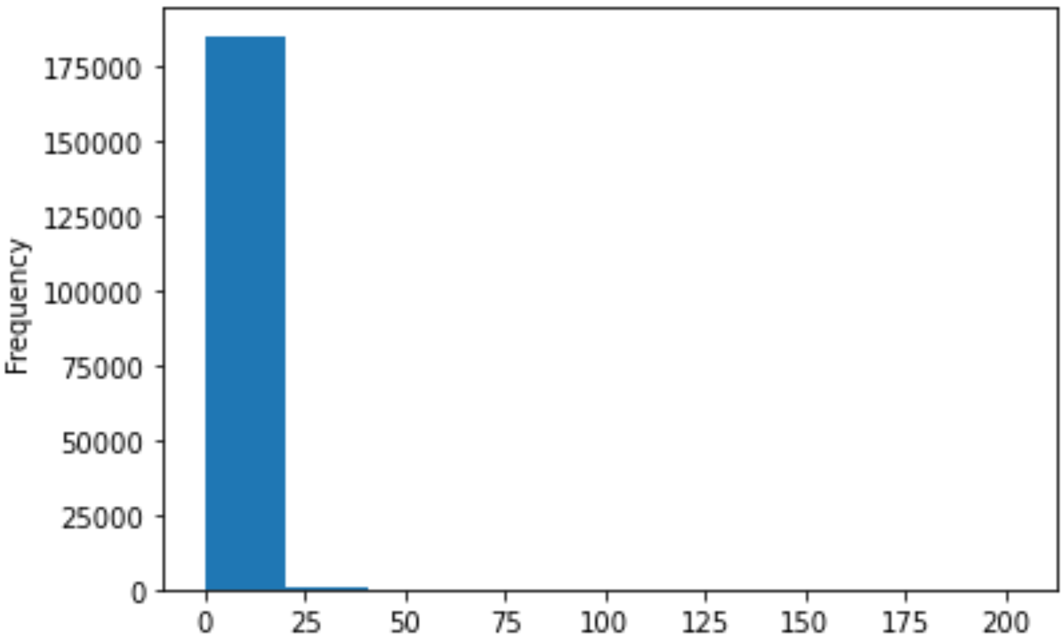
‘last_rech_date_da’



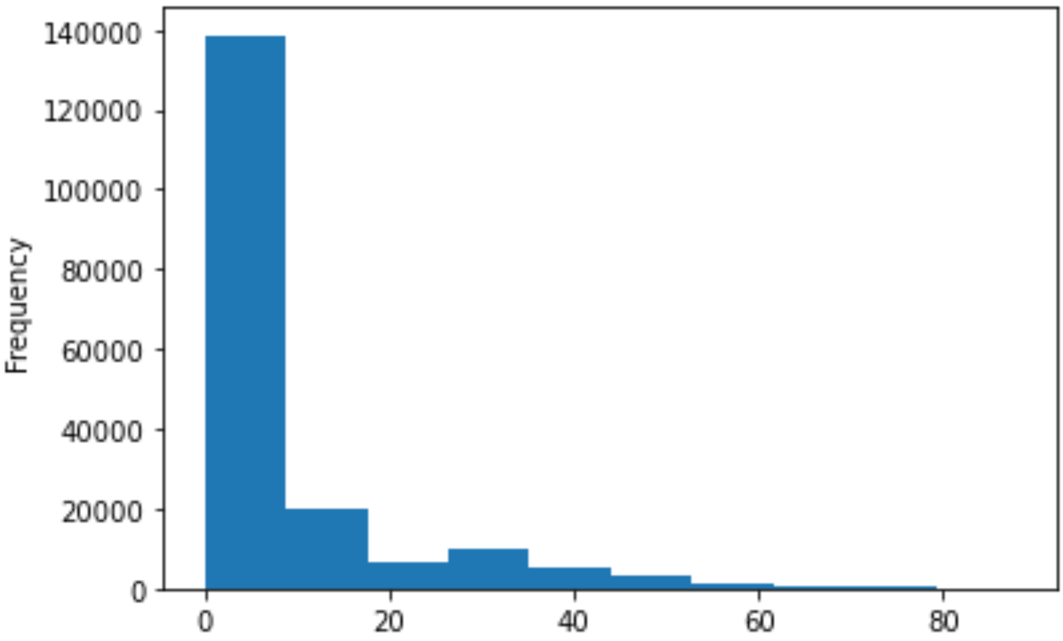
‘last_rech_amt_ma’



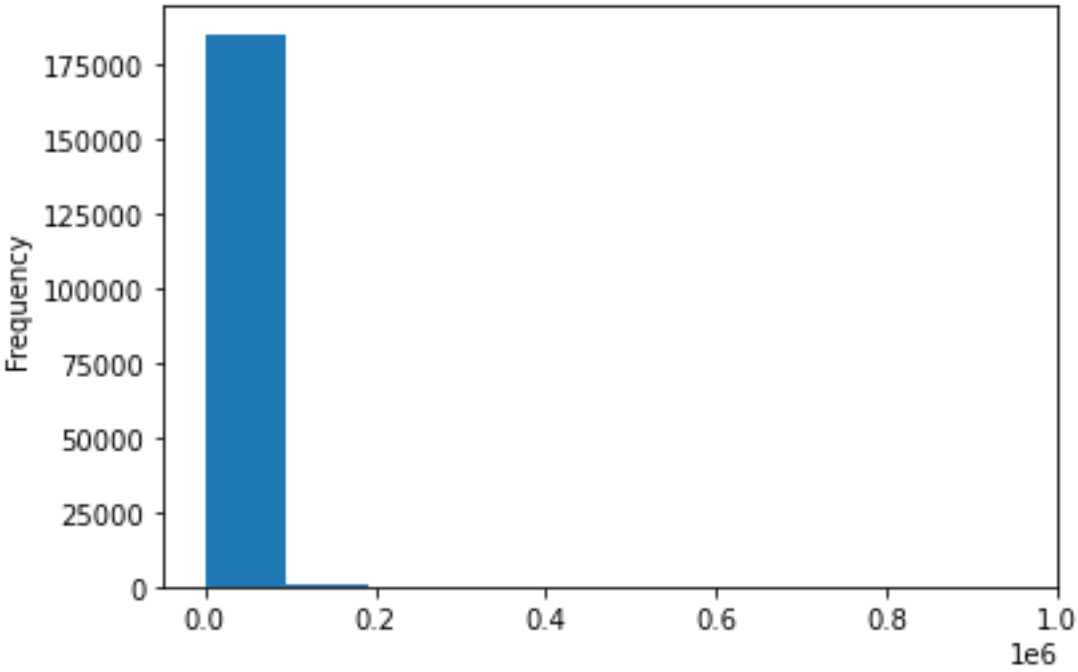
'cnt_ma_rech30'



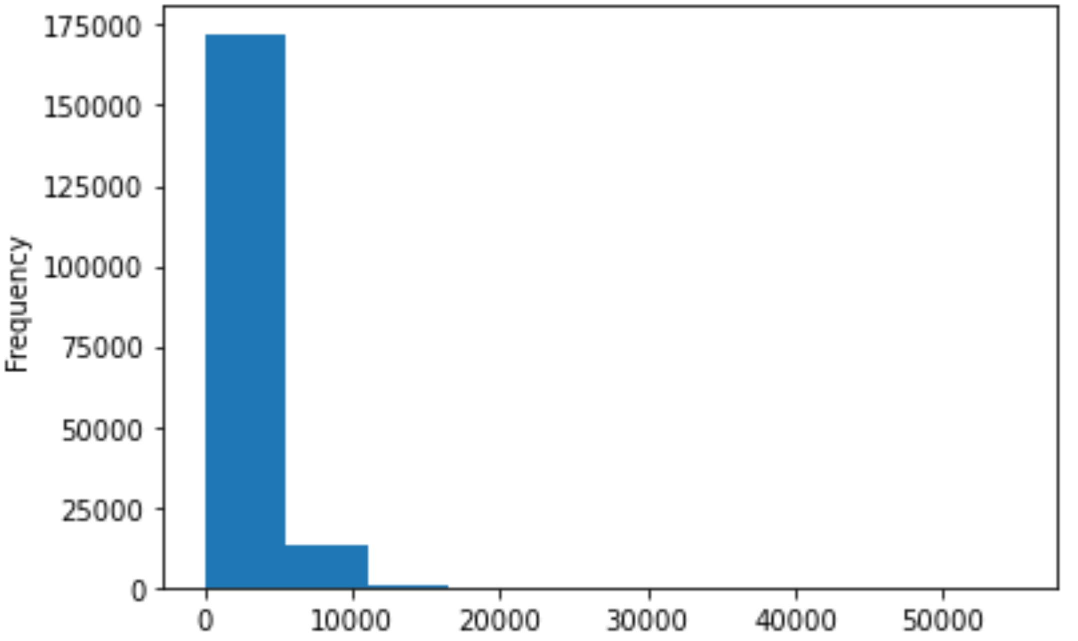
'fr_ma_rech90'



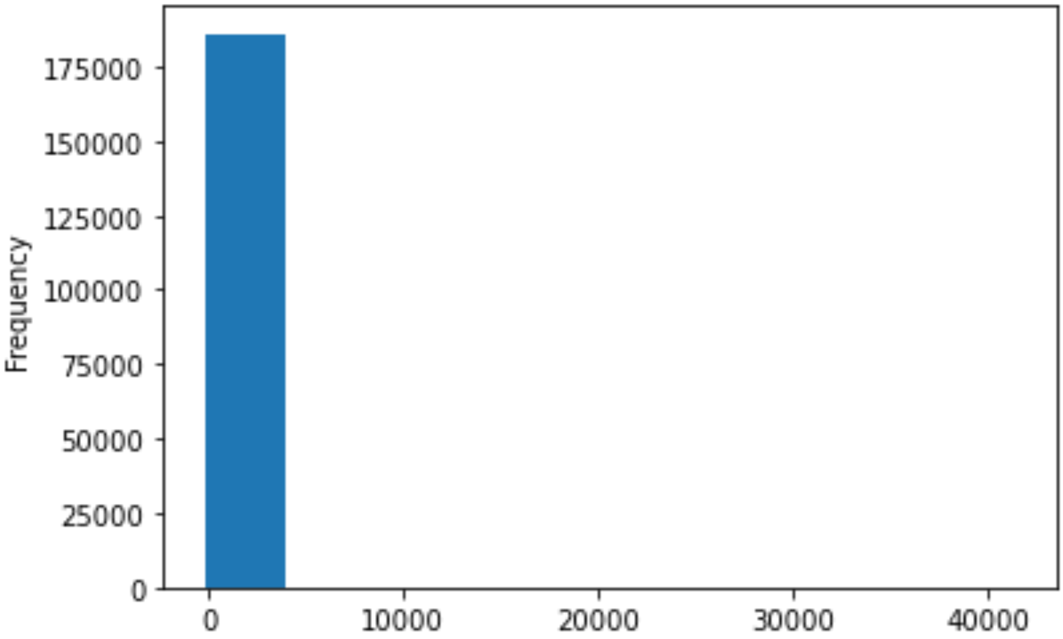
‘sumamnt_ma_rech90’



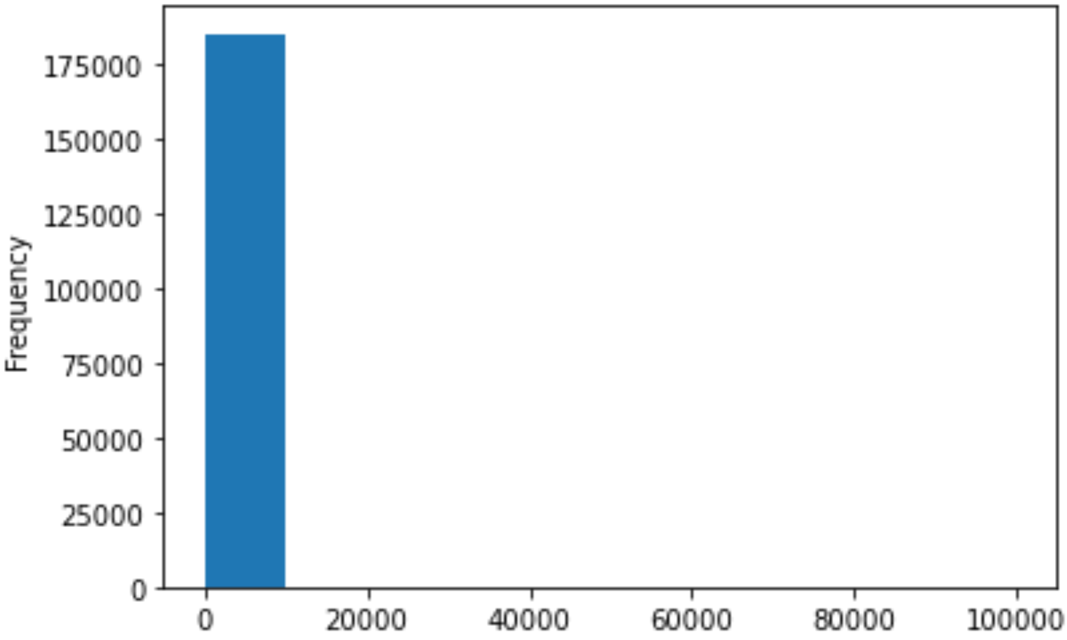
'medianamnt_ma_rech90'



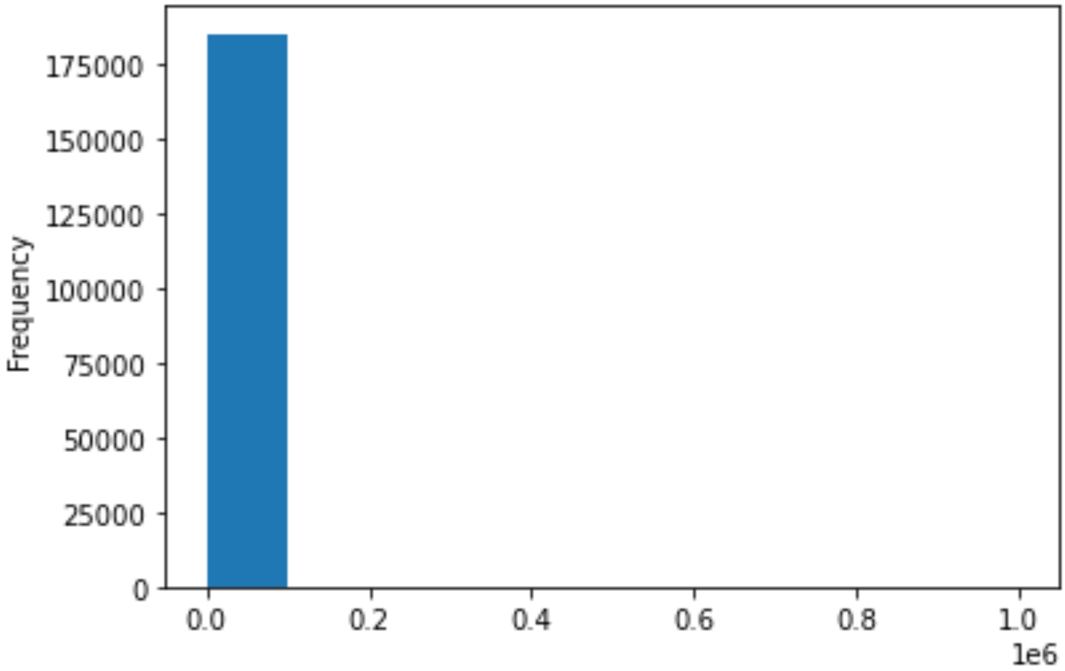
'medianmarechprebal90'



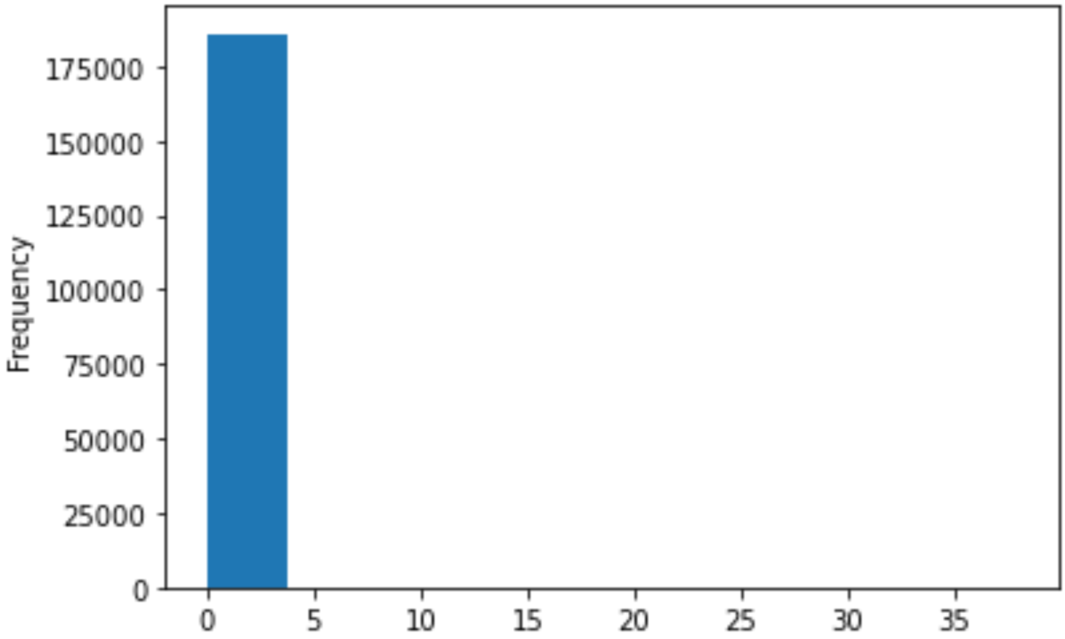
‘cnt_da_rech30’



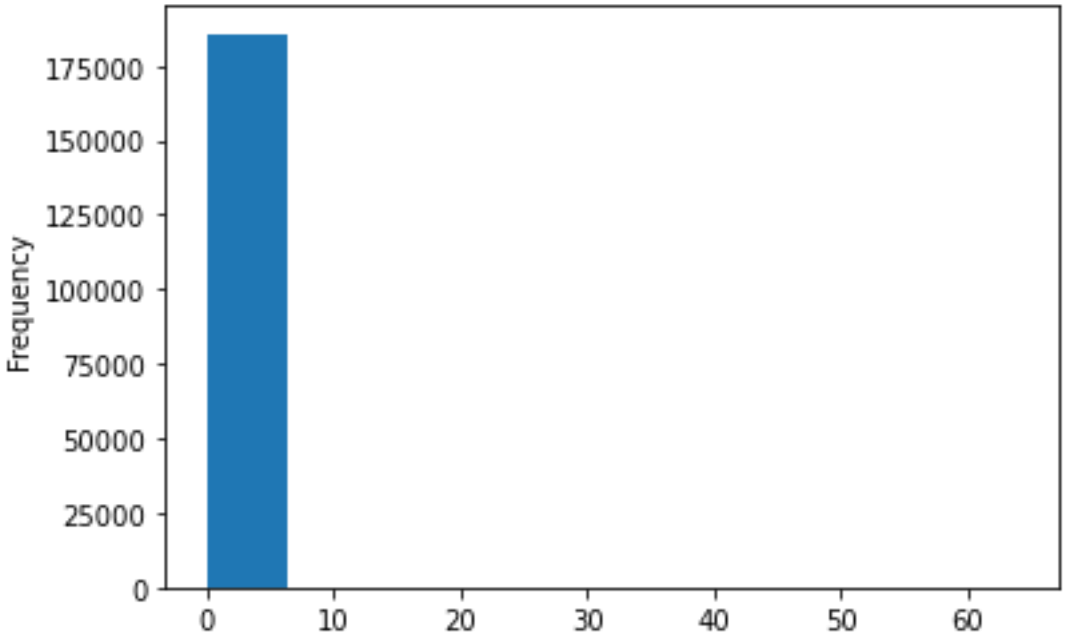
‘fr_da_rech30’



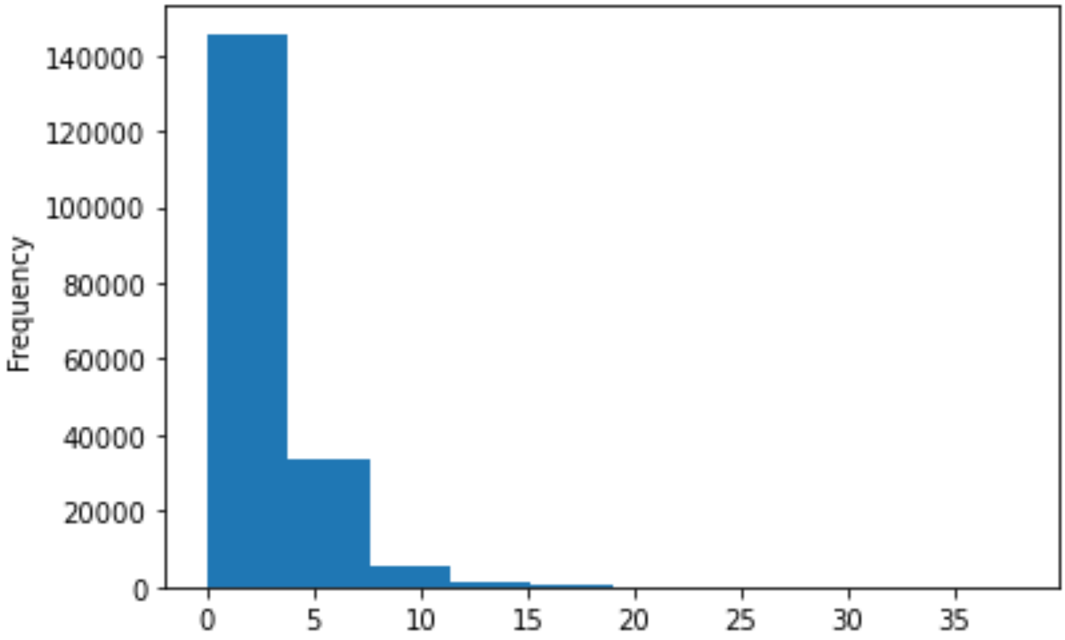
‘cnt_da_rech90’



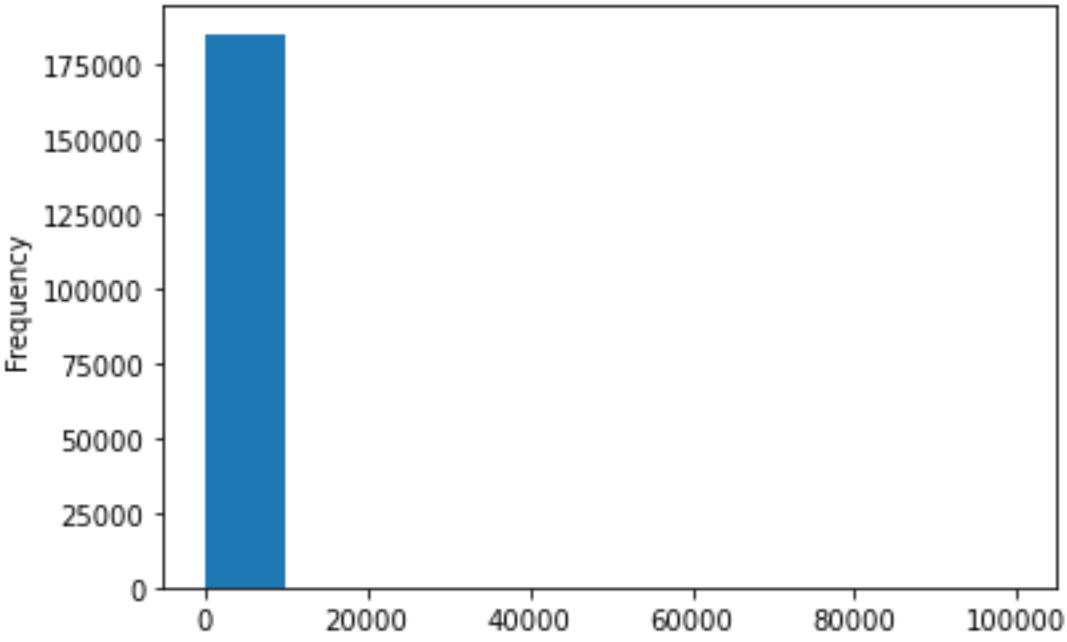
‘fr_da_rech90’



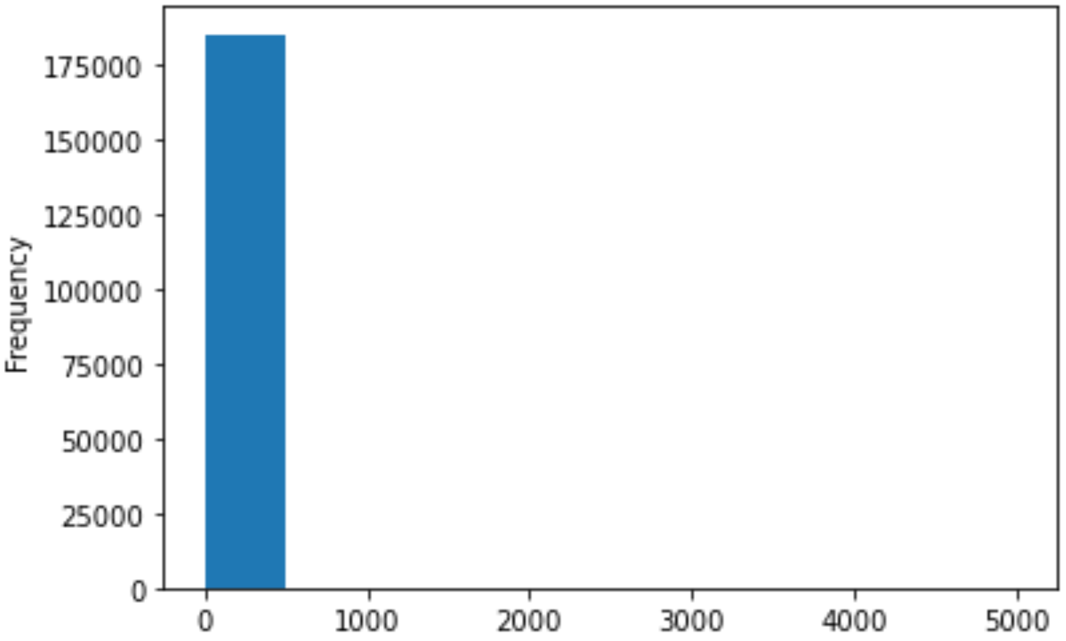
'cnt_loans30'



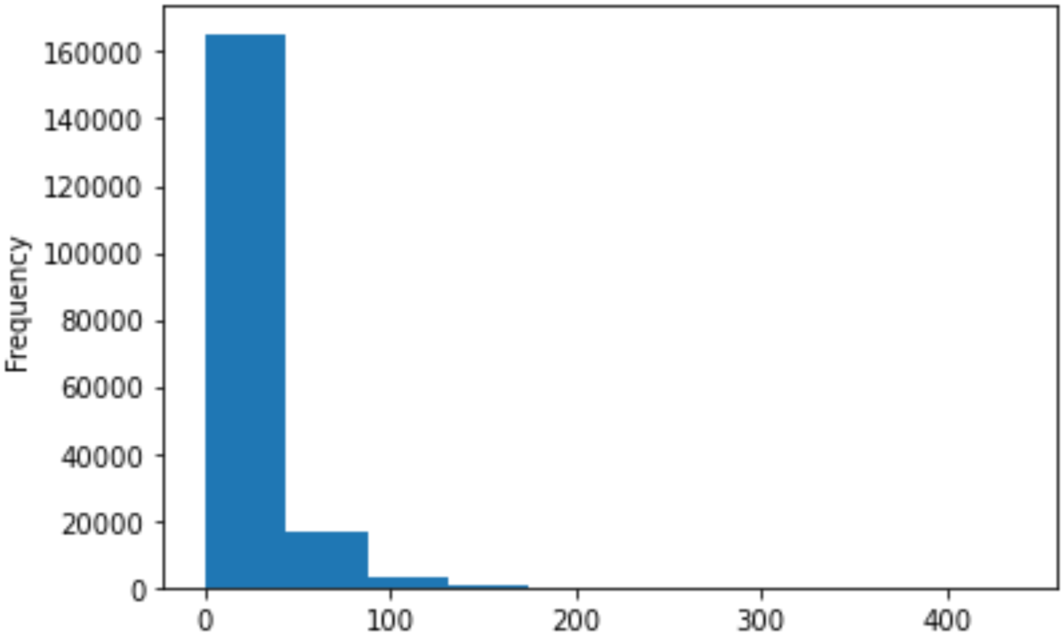
'maxamnt_loans30'



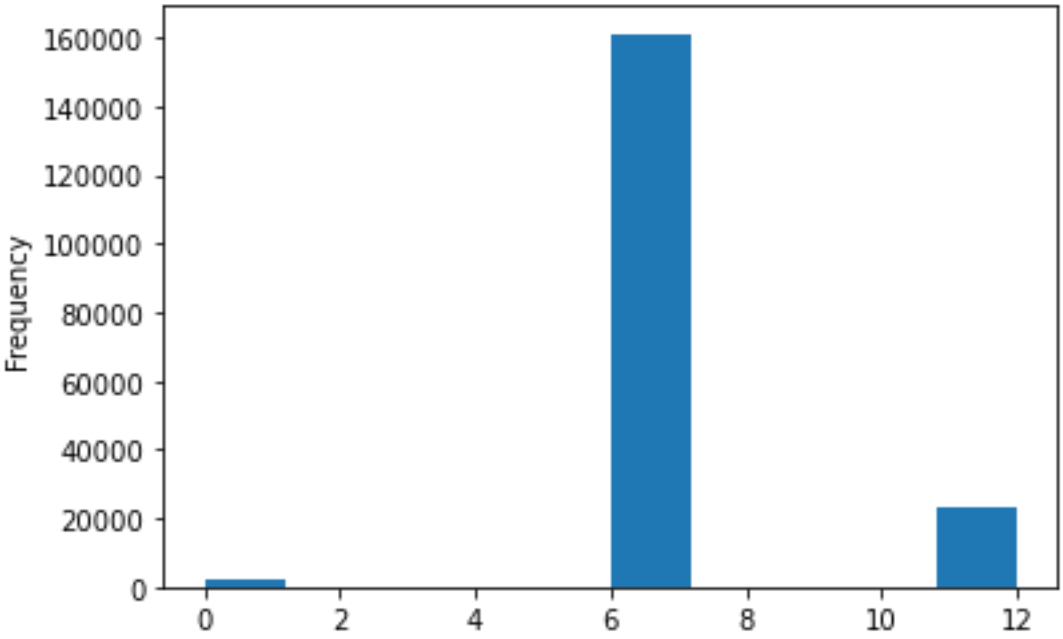
‘cnt_loans90’



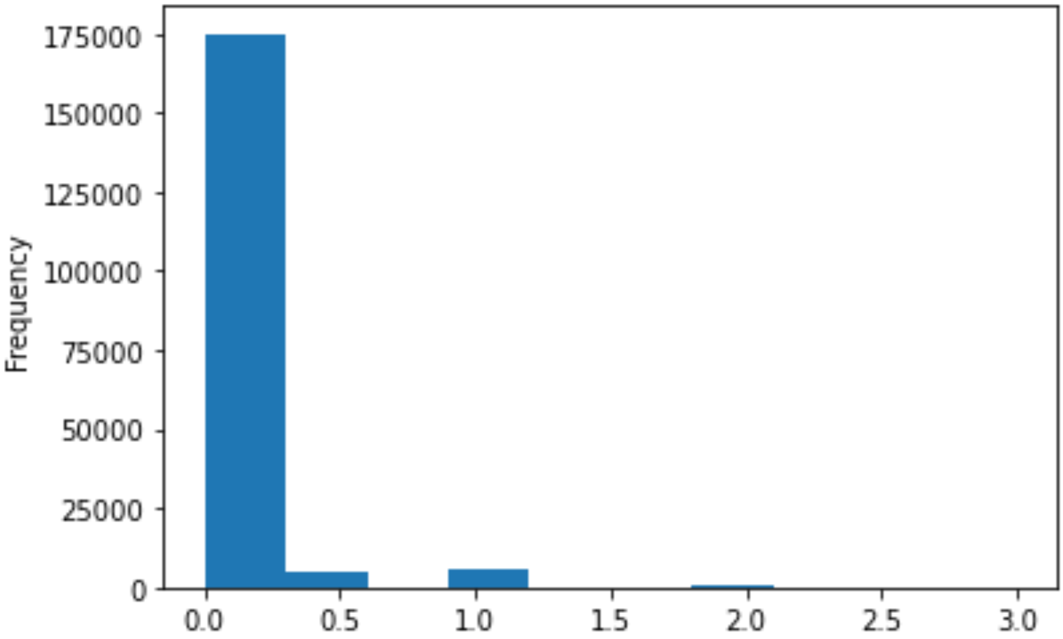
‘amnt_loans90’



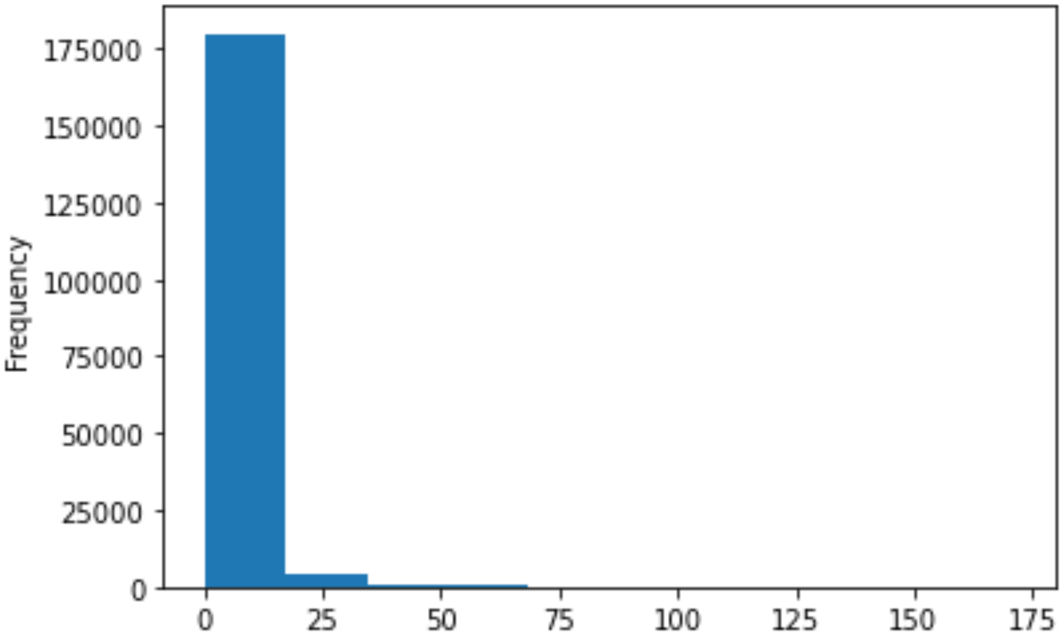
‘maxamnt_loans90’



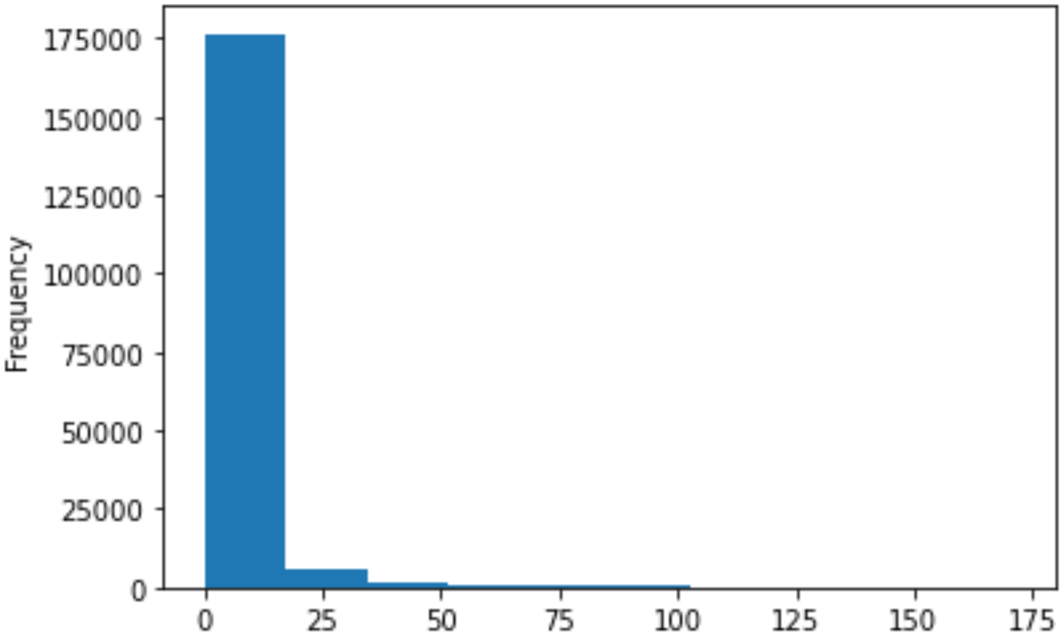
‘medianamnt_loans90’



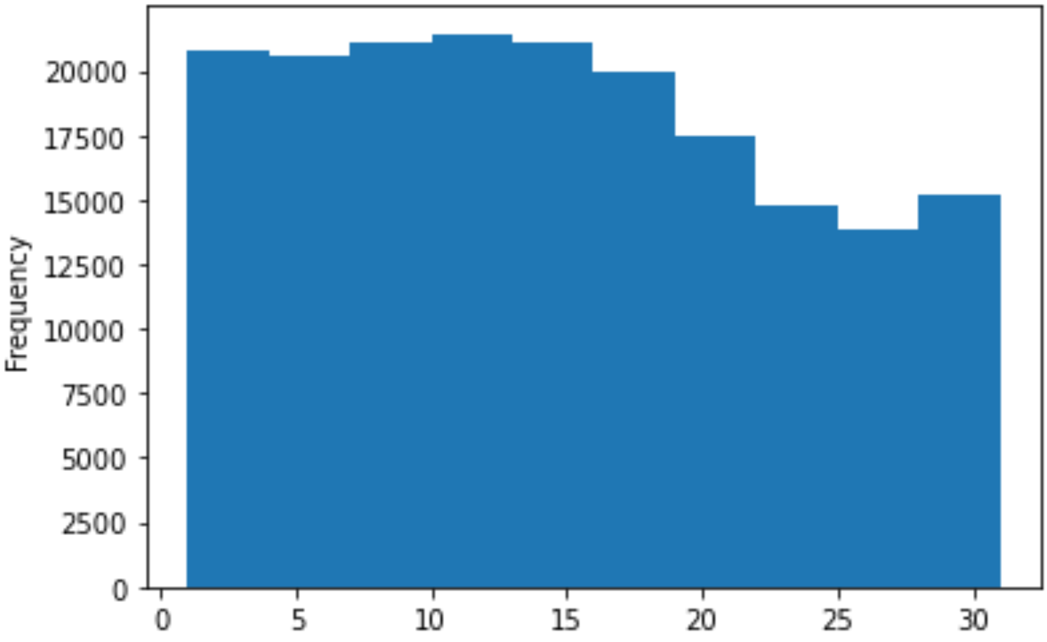
‘payback30’



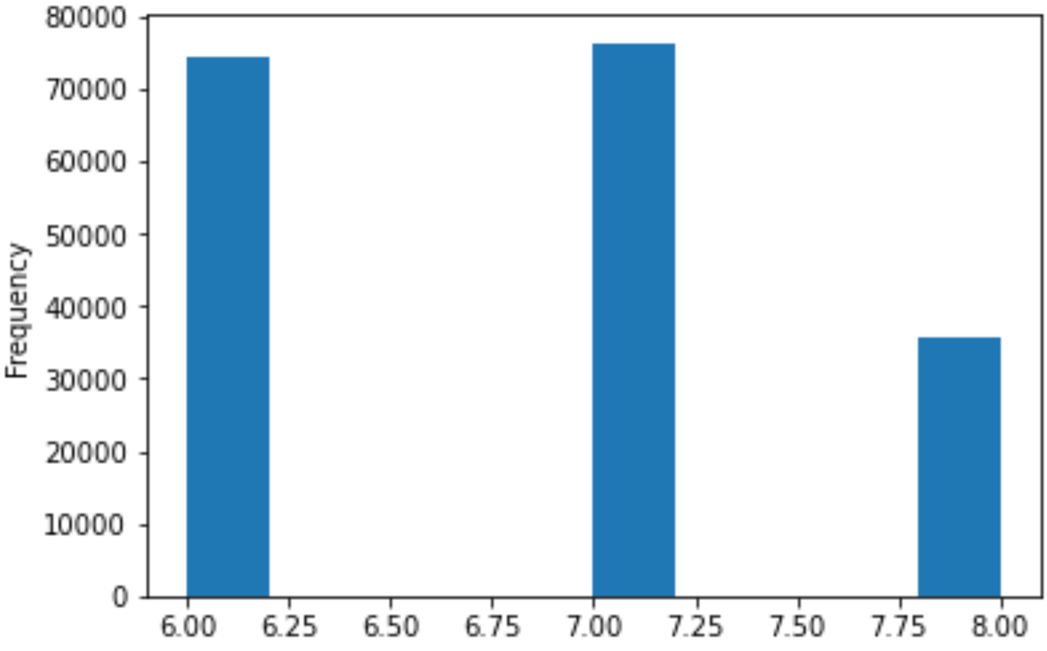
‘payback90’



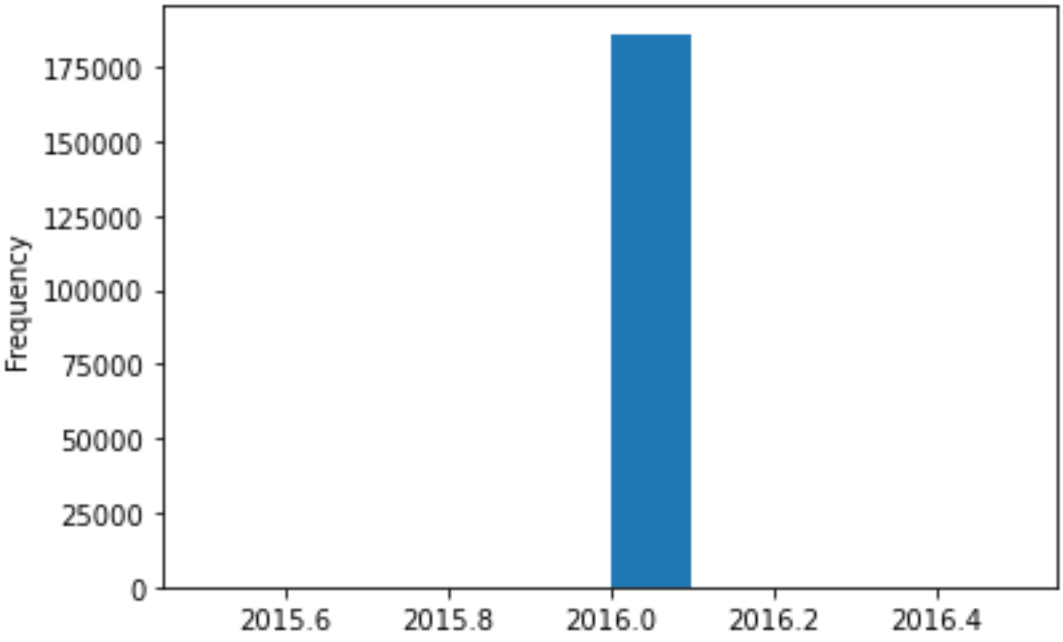
‘pDay’



'pMonth'

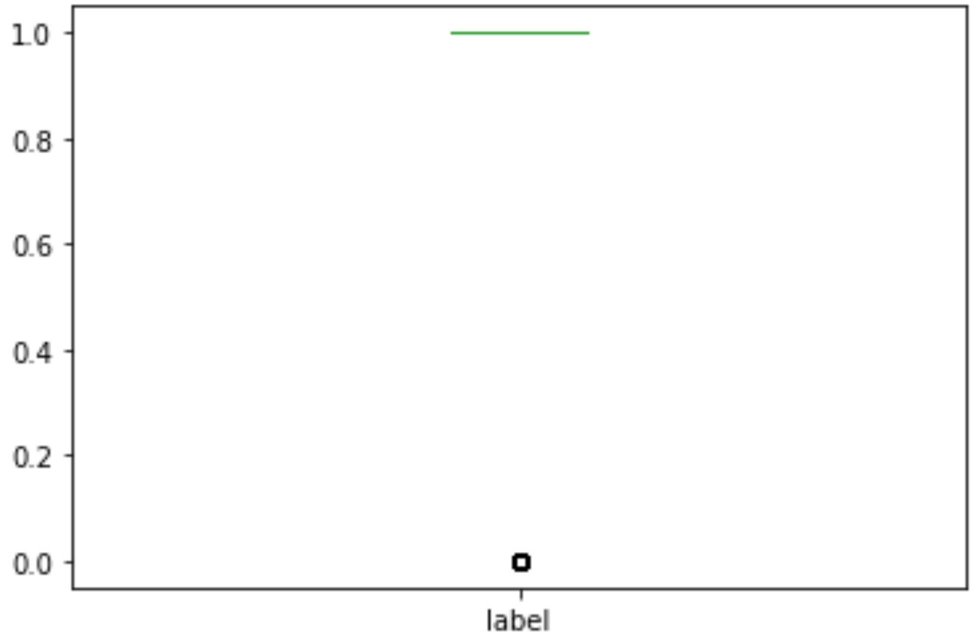


‘pYear’

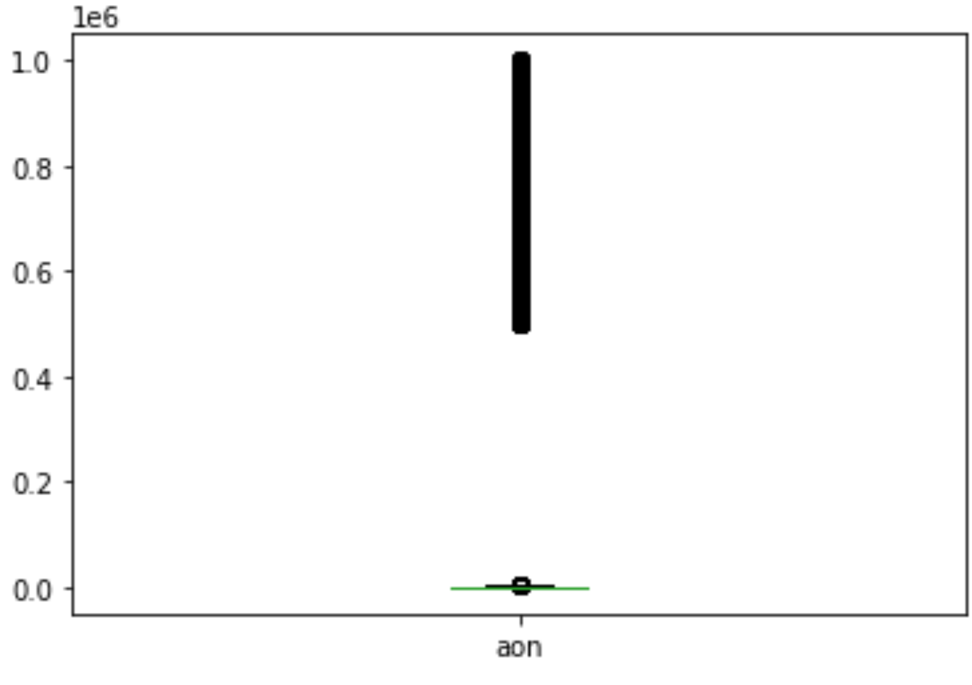


The presence of outliers in all the columns were checked using boxplot.

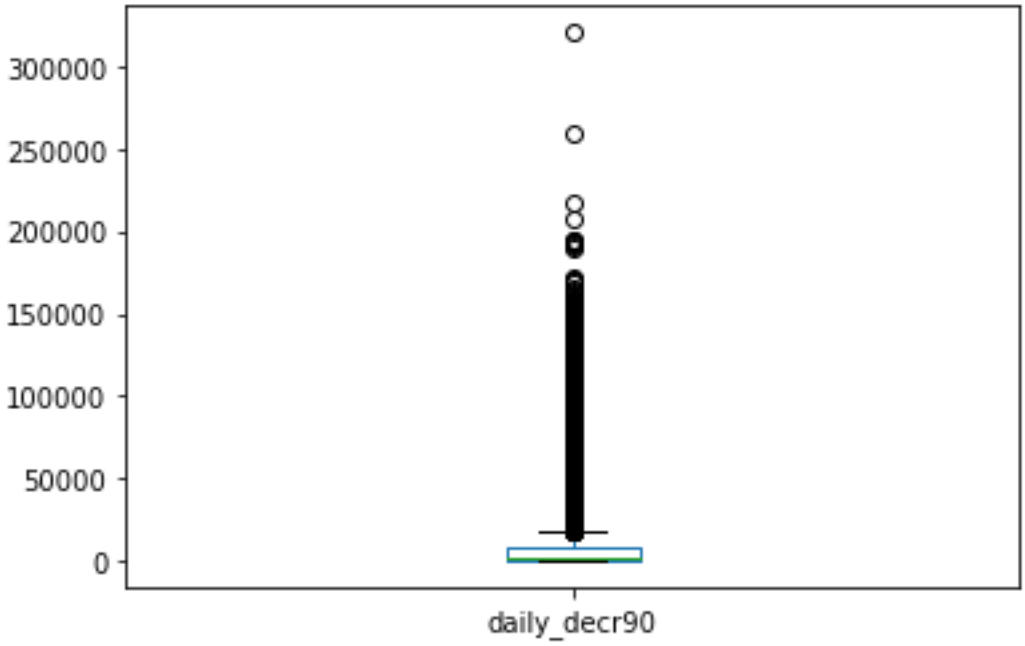
'label'



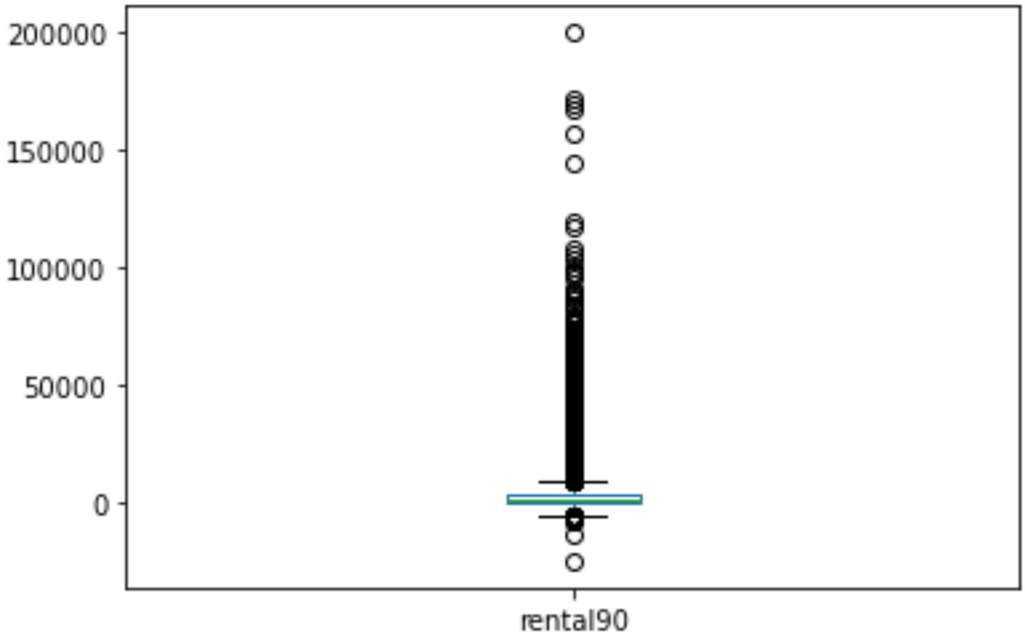
'aon'



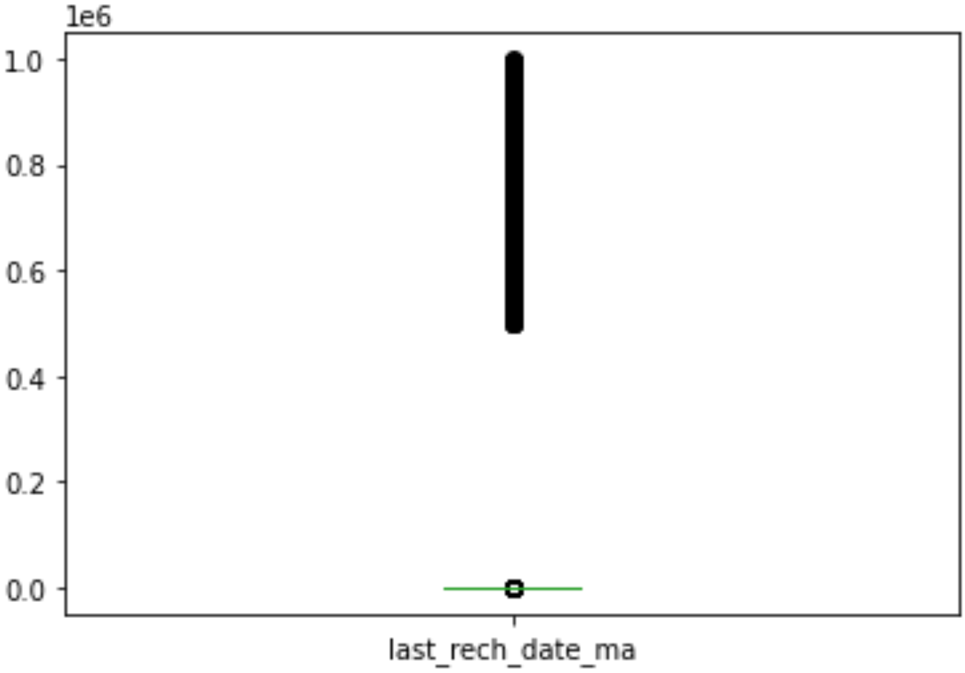
'daily_decr90'



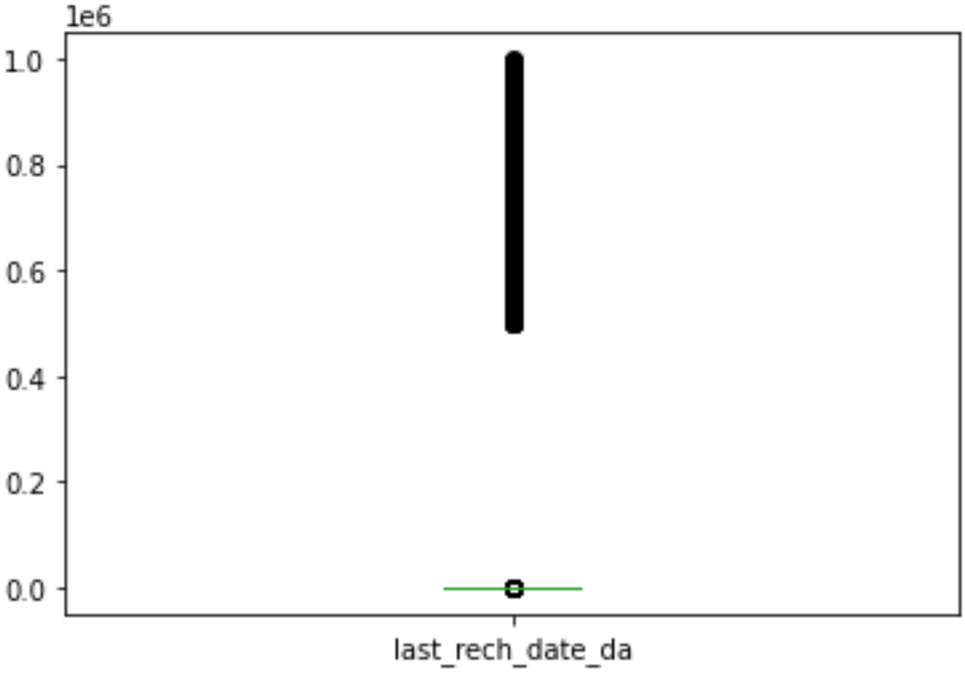
'rental90'



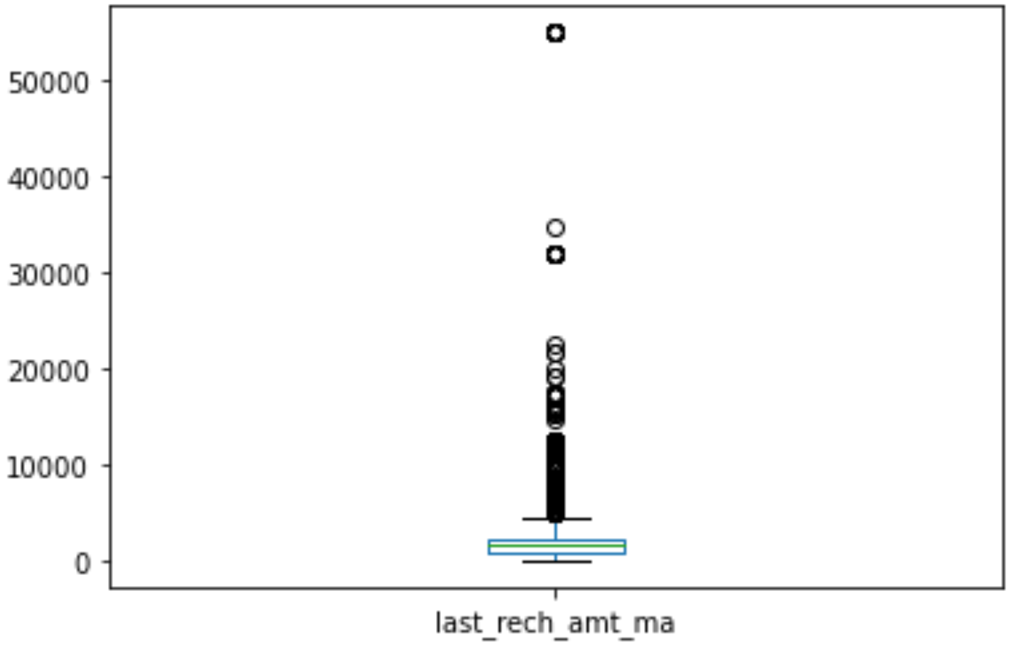
'last_rech_date_ma'



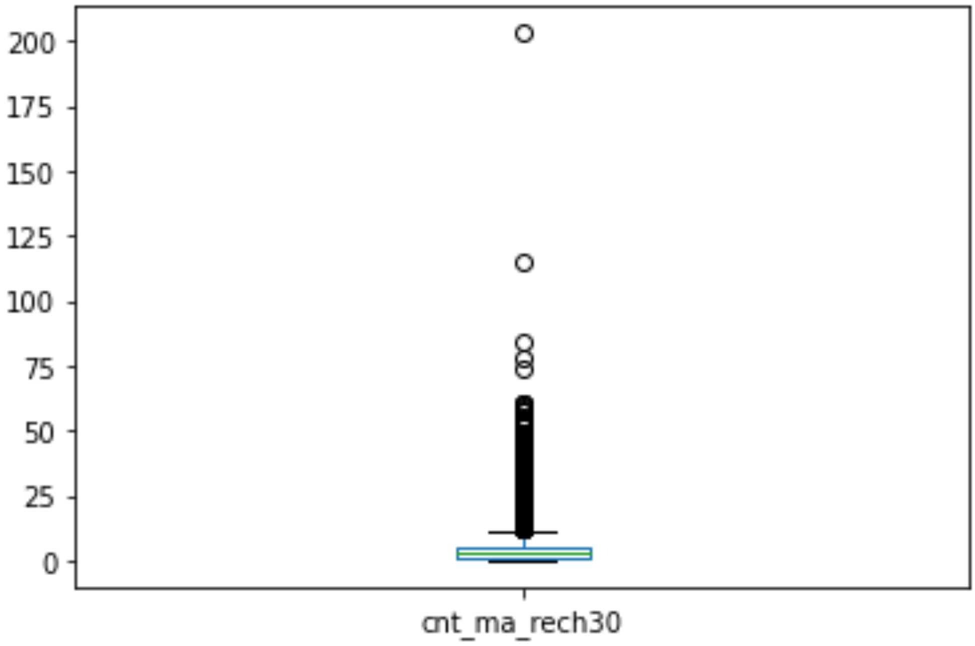
‘last_rech_date_da’



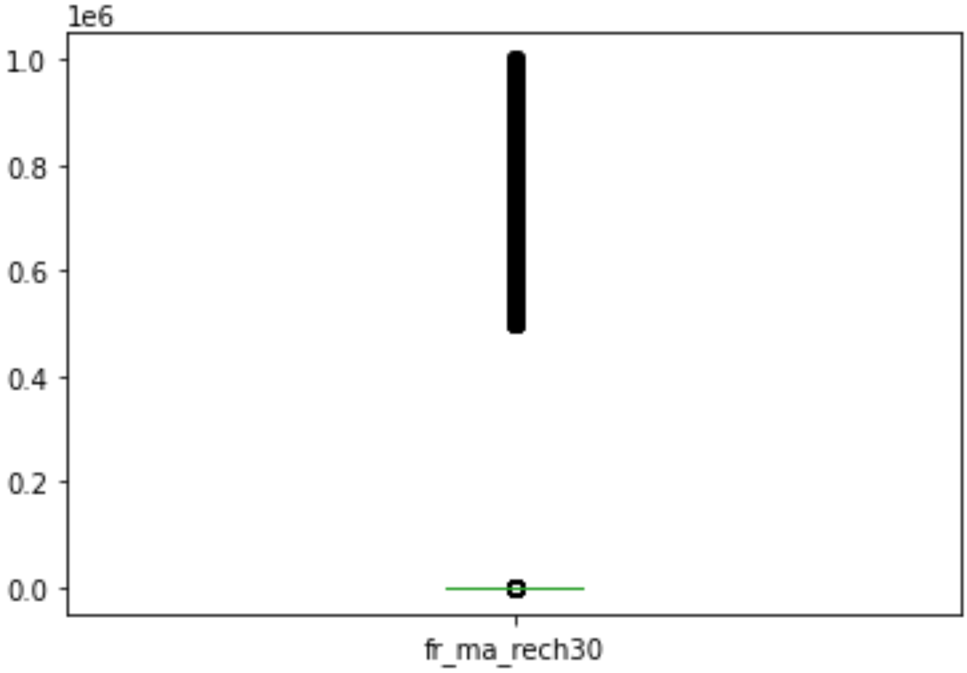
'last_rech_amt_ma'



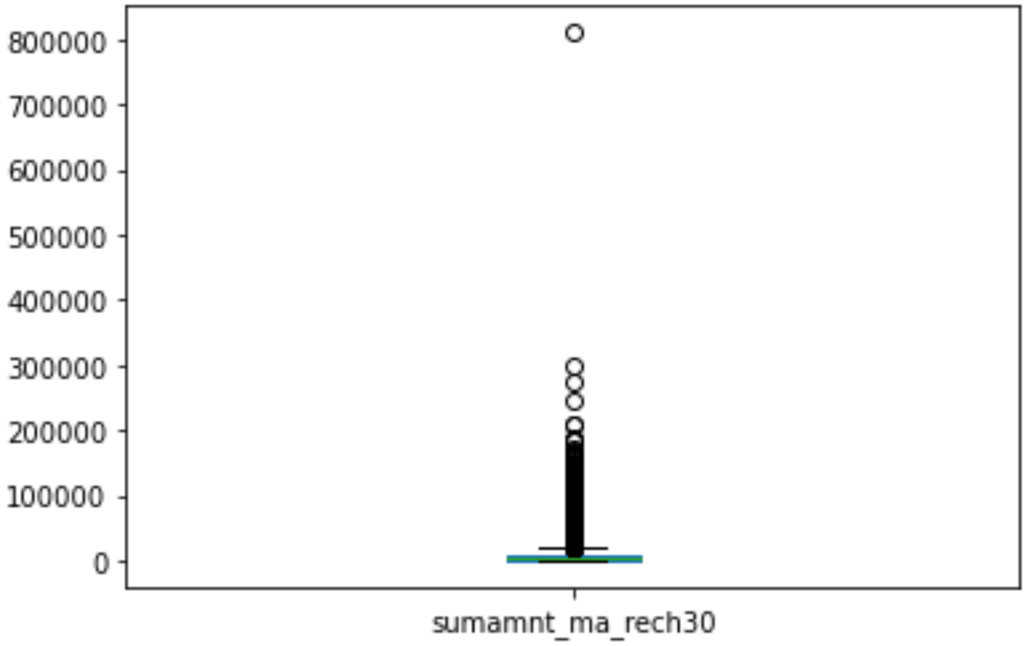
'cnt_ma_rech30'



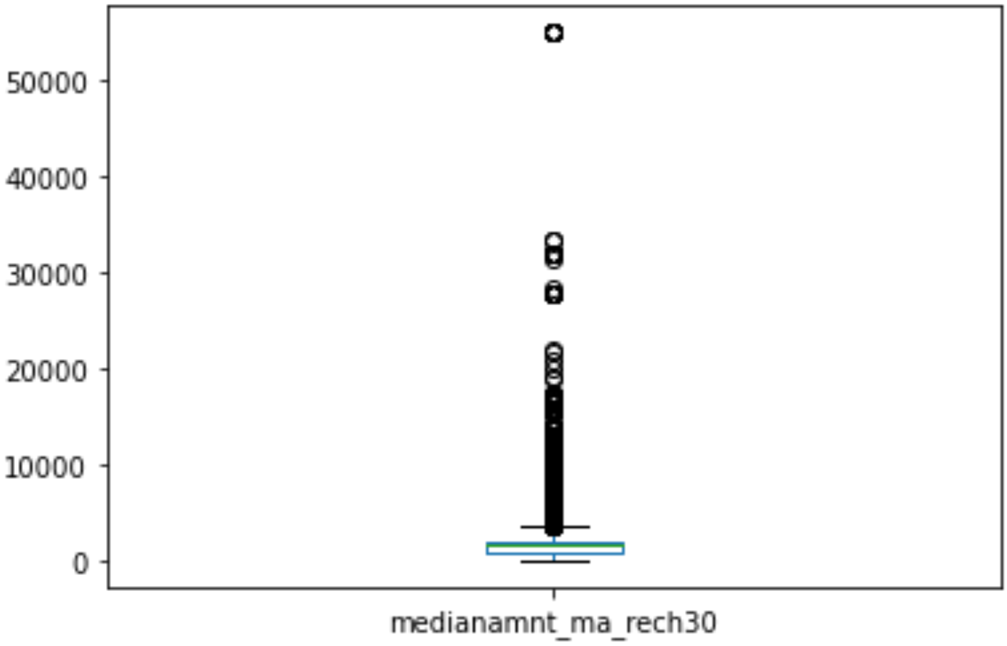
'fr_ma_rech30'



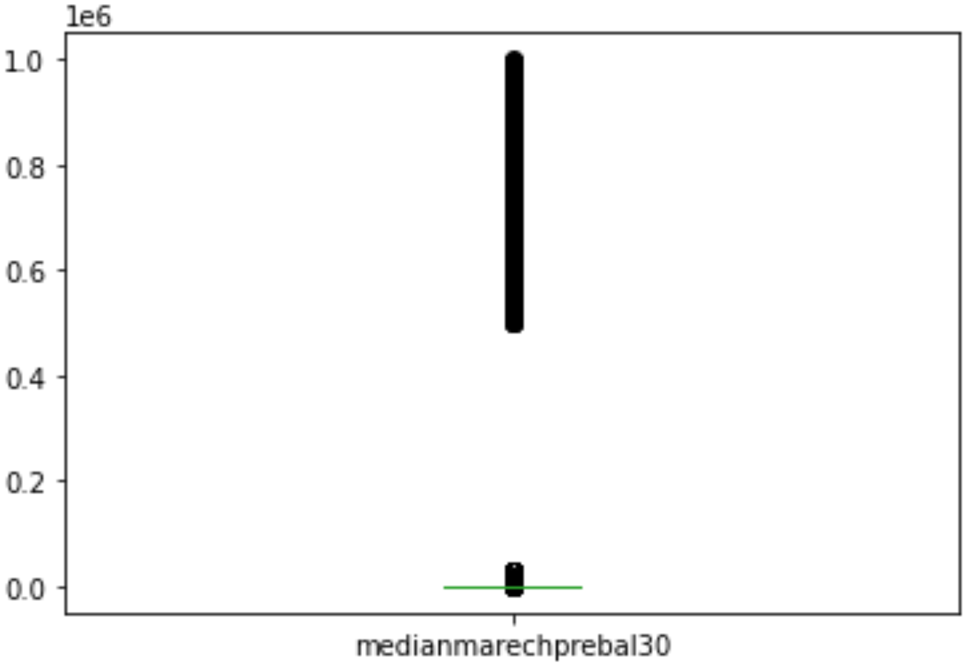
'sumamnt_ma_rech30'



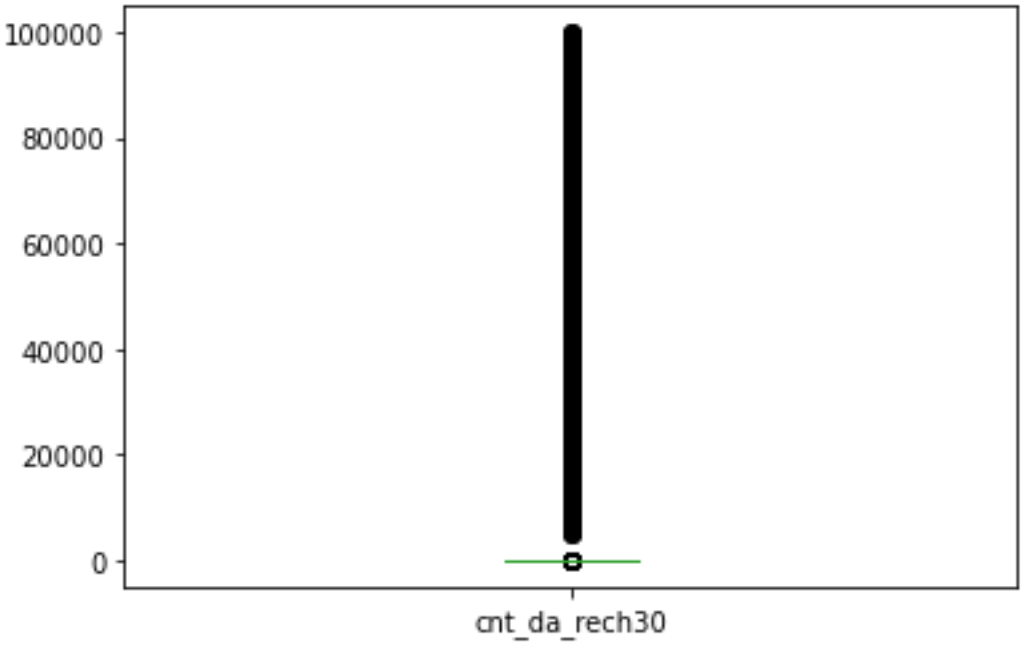
'medianamnt_ma_rech30'



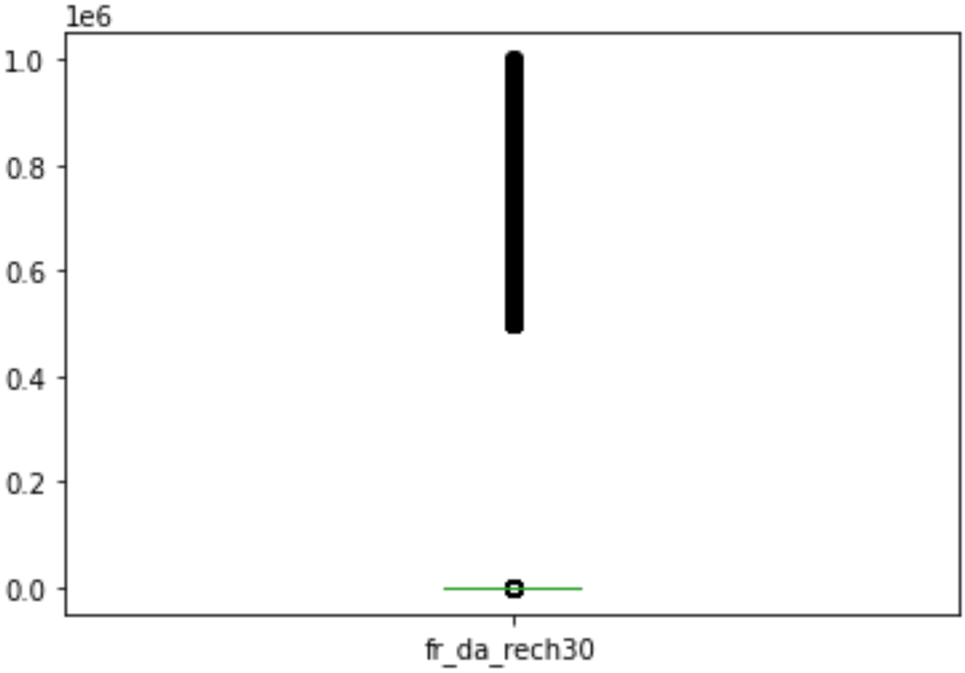
'medianmarechprebal30'



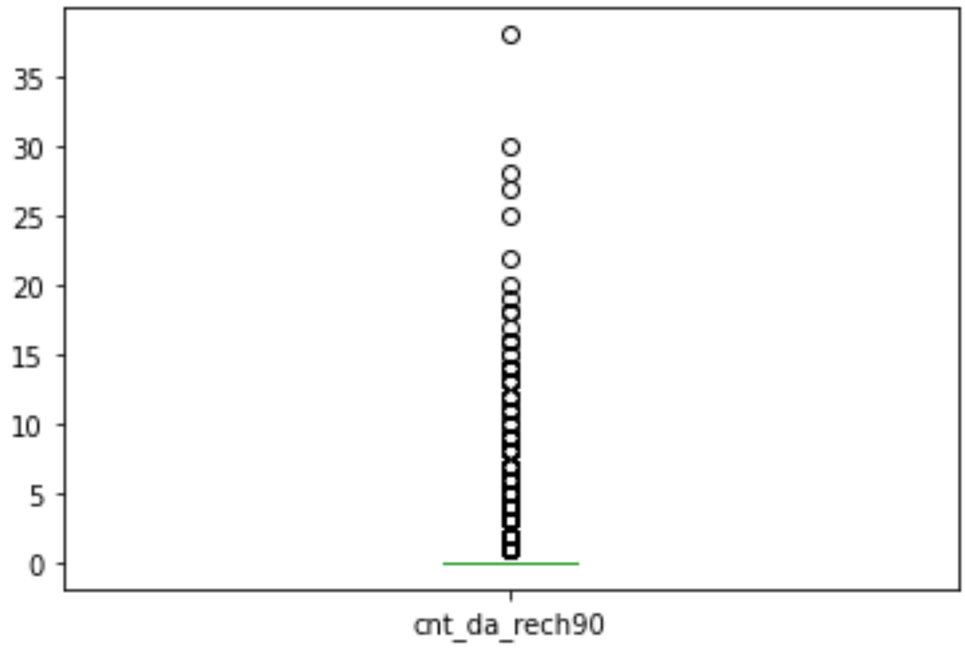
'cnt_da_rech30'



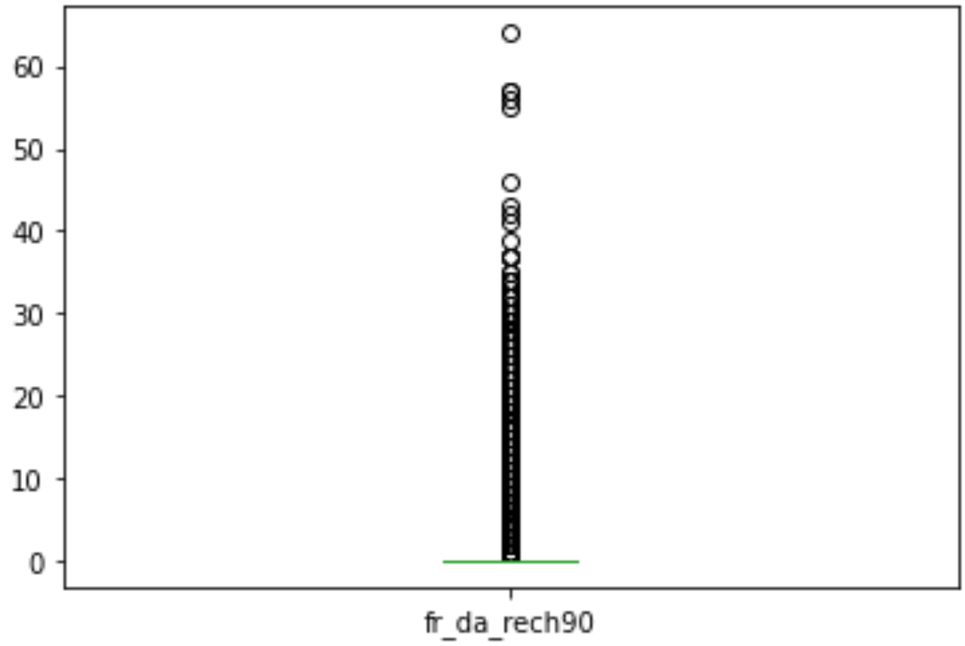
'fr_da_rech30'



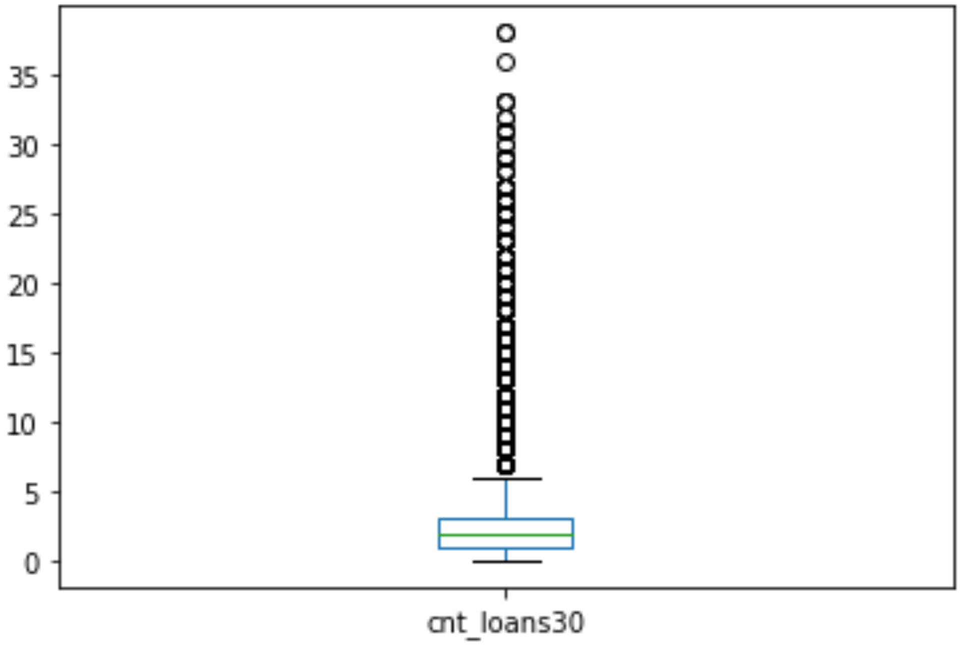
'cnt_da_rech90'



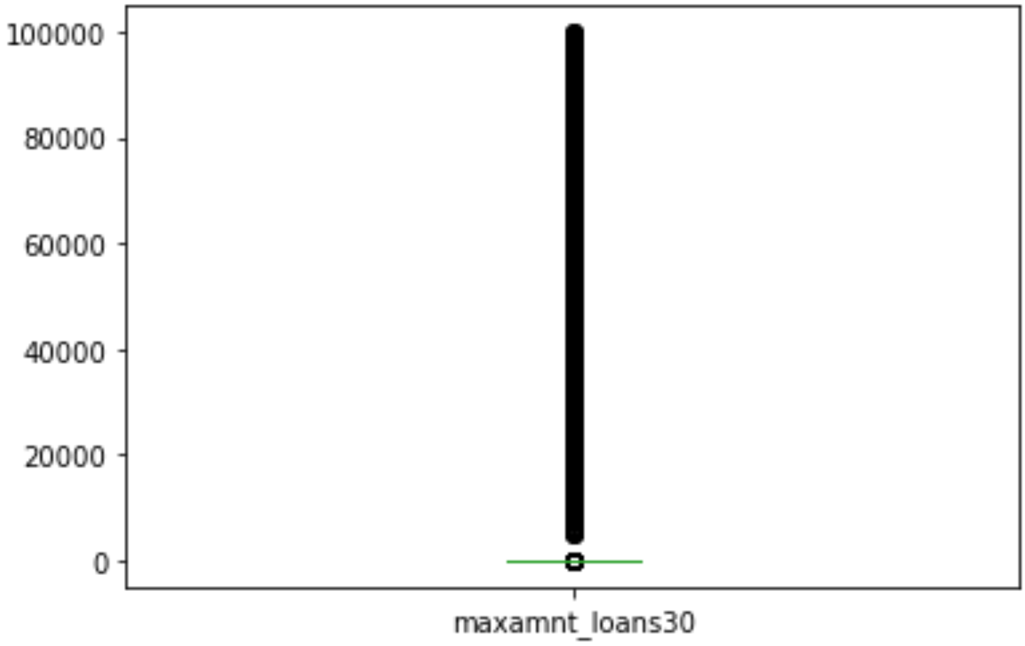
'fr_da_rech90'



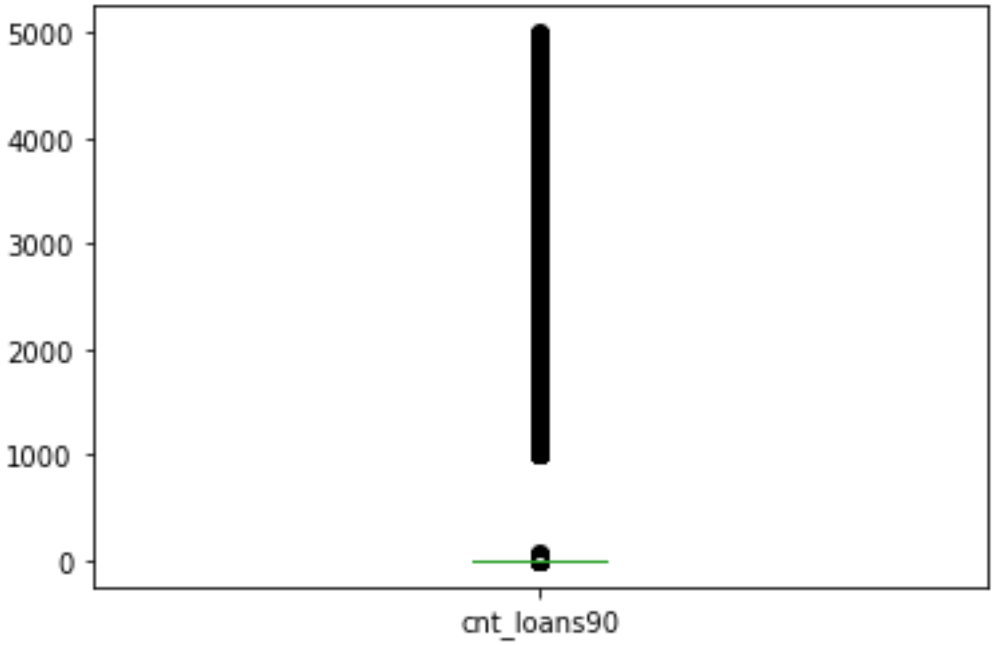
'cnt_loans30'



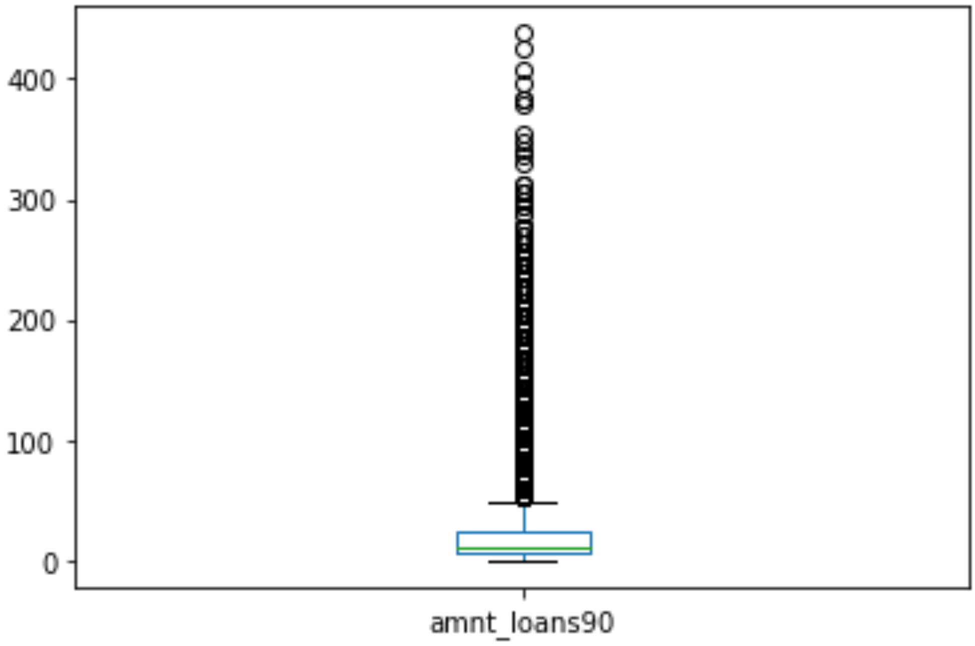
'maxamnt_loans30'



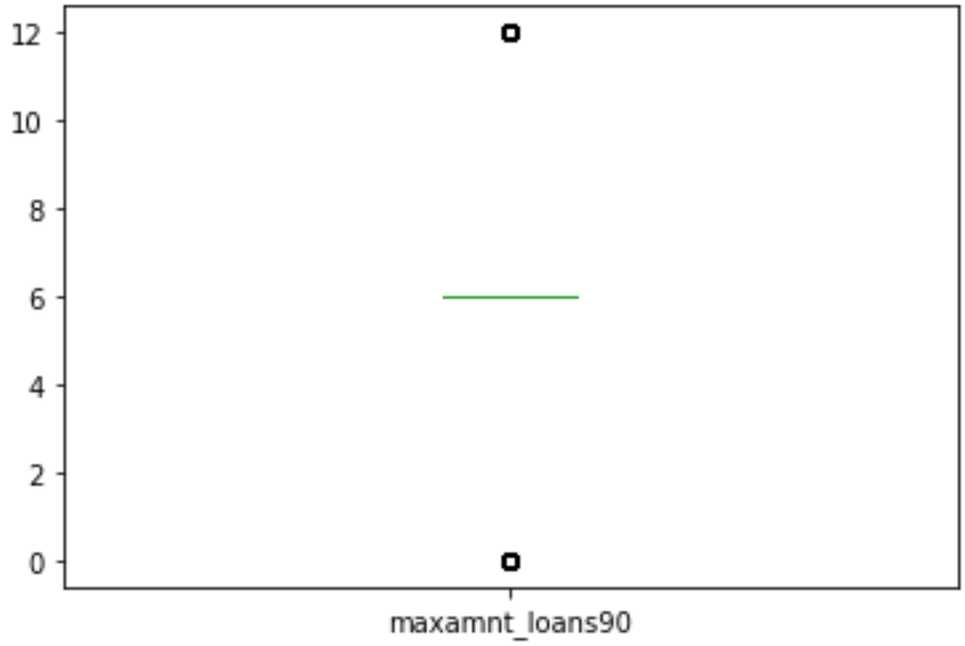
'cnt_loans90'



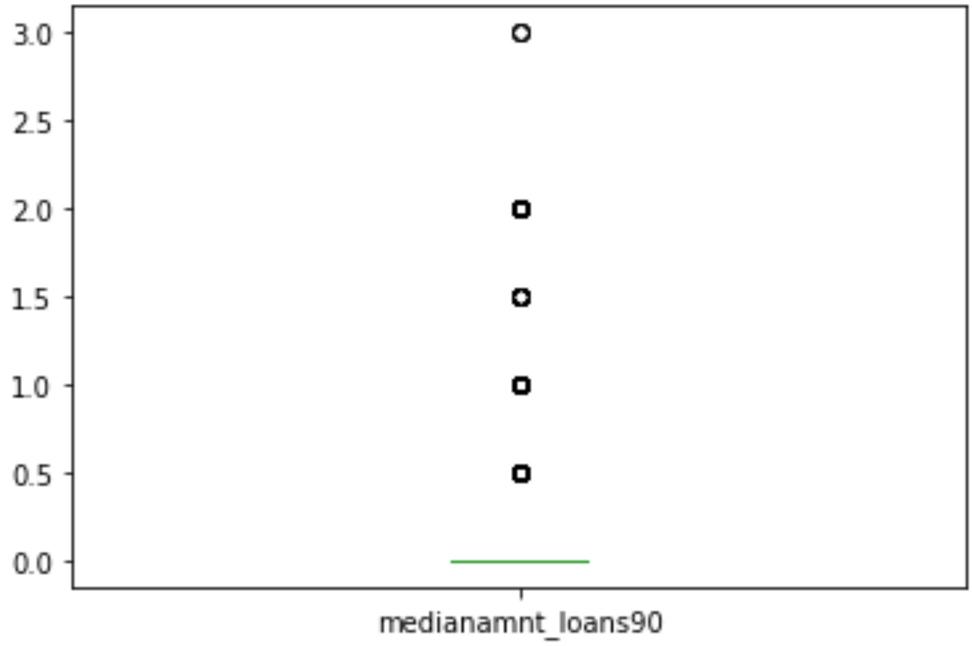
'amnt_loans90'



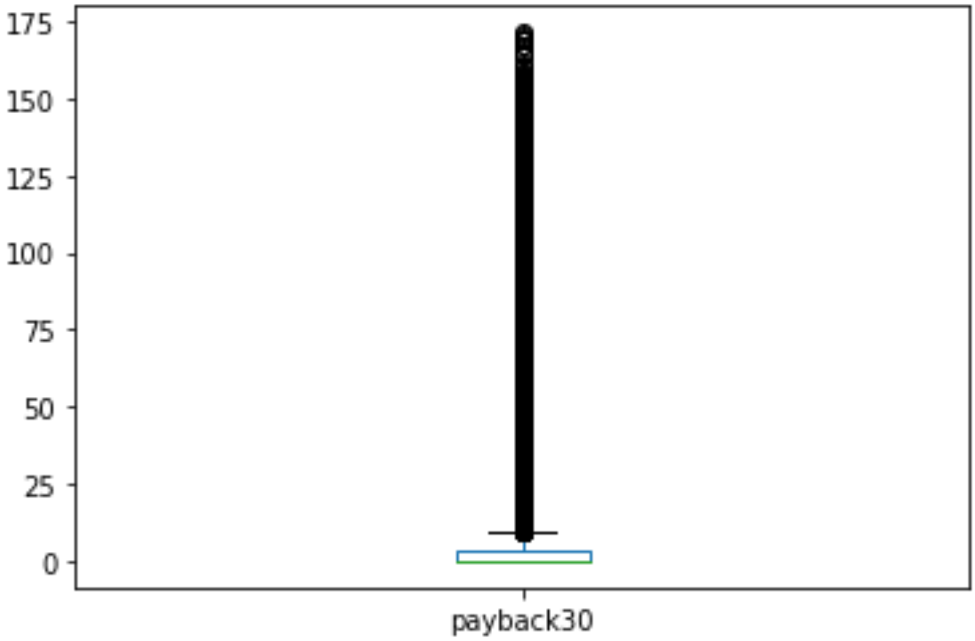
'maxamnt_loans90'



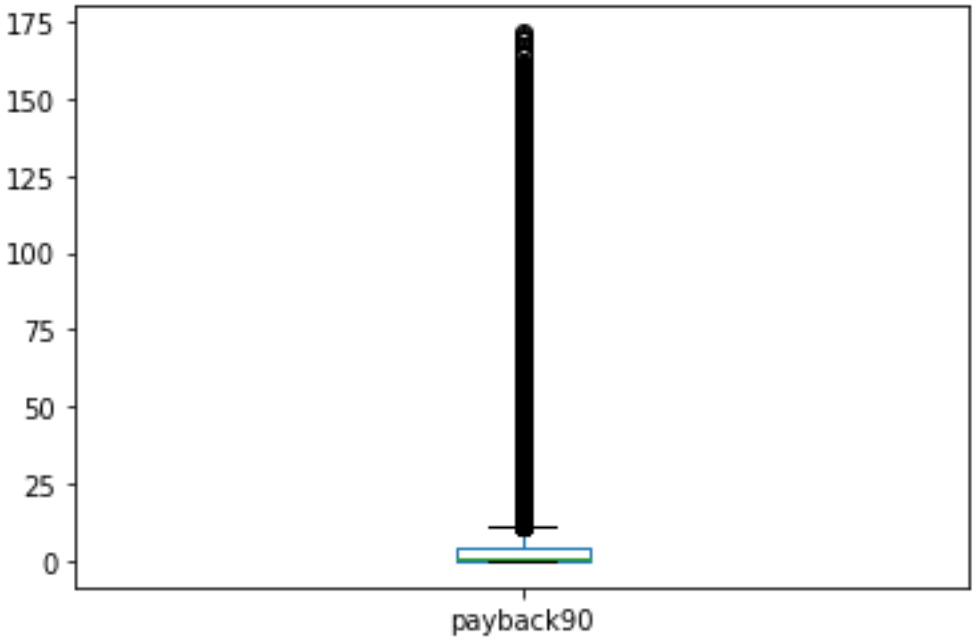
'medianamnt_loans90'



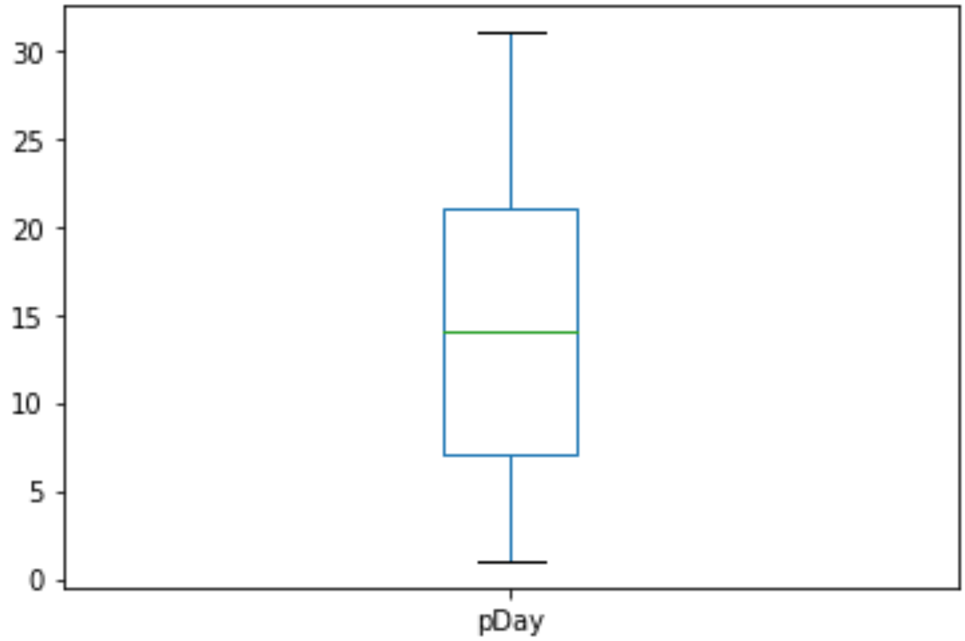
'payback30'



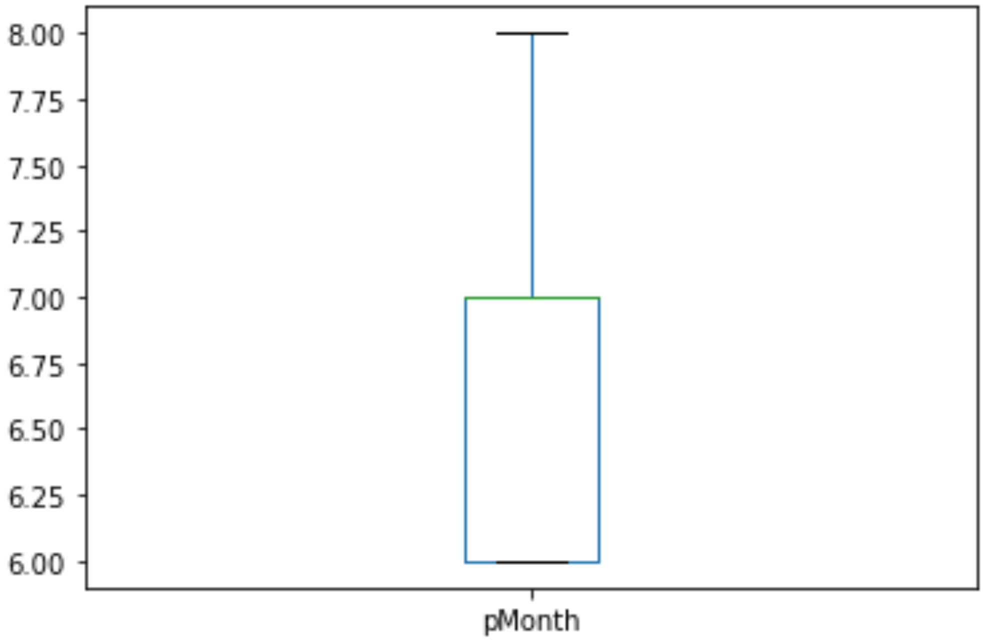
'payback90'



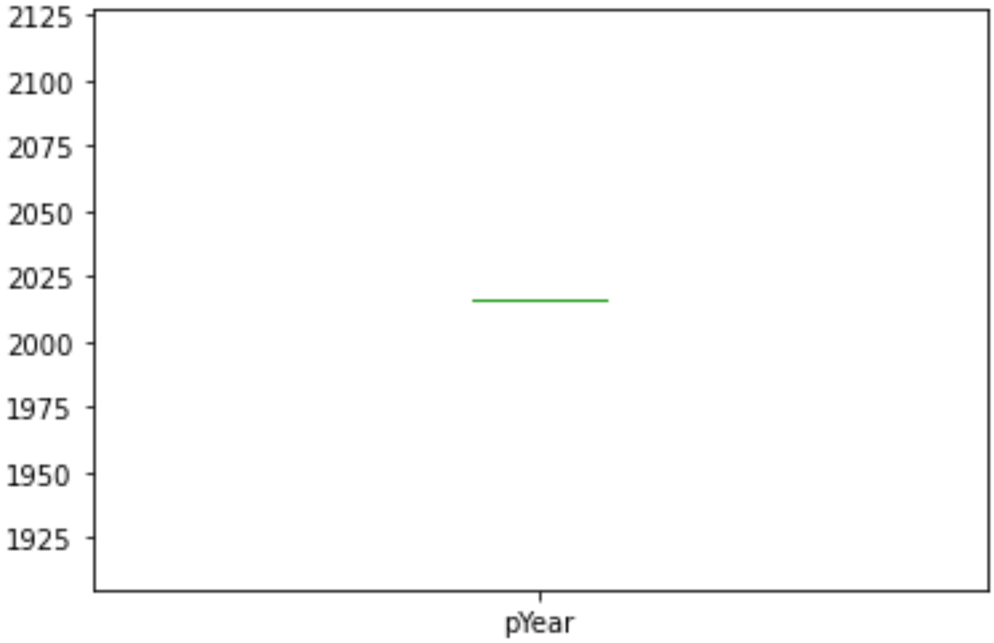
'pDay'



'pMonth'



'pYear'



- Interpretation of the Results

From the corelation table and graph we observe the following:

'daily_decr30' and 'daily_decr90' are highly correlated with each other.
'rental30' and 'rental90' also are highly correlated with each other.
'cnt_loans30' and 'amount_loans30' columns are highly correlated with each other.
'amount_loans30' is also highly correlated with 'amount_loans90' column.
'medianamnt_loans30' and 'medianamnt_loans90' is highly correlated with each other.

From the countplot we observe the following:

Label 1 indicates loan has been payed i.e. Non-Defaulter and label 0 indicates that the loan has not been payed i.e. defaulter
Defaulters=25860
Non-defaulter=160383.

From the histogram we observe the distribution in data in all the columns.

From the countplot for customer label according to date:

The figure which is date vs label shows that the customers who did not pay their loans are from date 10 to 23.

From the countplot for customer label according to month:

There are several customers in June and July month who did not pay their loan.

From the histograms of skewness we can observe ‘medianmarechprebal90’ with highest skewness and ‘pyear’ has 0 skewness.

From the boxplot of outliers we can observe the highest number of outliers in ‘cnt_da_rech30’, ‘maxamnt_loans30’ and ‘payback30’columnsand no ouliers i n ‘pDay’,’pMonth’ and ‘pYear’ columns.

From the modelling point of view we can draw the following observations:

1. We know that this is classification problem so we use accuracy score, classification report and confusion matrix as our evaluation matrix. We also see the AUC score and also plot the AUC_ROC curve for our final model.
2. As we know this dataset is imbalance so we don’t too much focus on accuracy score . We see the precision and recall value along with f1_score.
3. First we see the result without doing any sampling technique and for that I use Logistic Regression with K-Fold cross validation and hyper-parameter tuning.
4. We also use Random Forest Classifier as our evaluation model without using hyper-parameter tuning because our dataset is too large and it takes more than hour to give the result.

CONCLUSION

- Key Findings and Conclusions of the Study

  So here ‘RandomForestClassifier’ is the best model out of all model tested above and by looking this we can conclude that our model is predicting around 92% of correct results for Label ‘0’ indicates that the loan has not been payed i.e. defaulter.