Home Credit Default Risk

Mohanad Abdelhalim ibrahim

Business Problem

- We intend to classify new applicants for our financial Insititute based on a lot of attributes and information about him as defaulter or non-defaulter as we want to estimate the applicant ability to repay the loan in the intended time.

Insights and findings

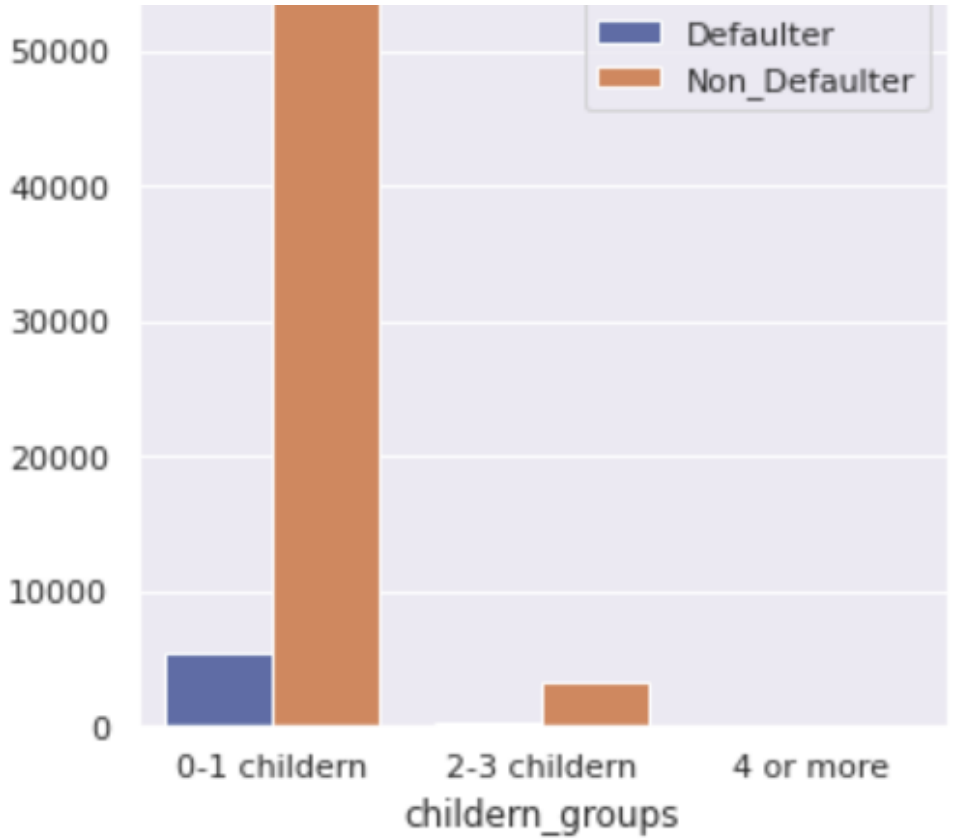
- Data provided by Kaggle competitions where it contains 8 csv files with info about applicants.
- Due to lack of time I used only features of main csv which were about 122 feature.
- As predicted the main challenge in our problem is the massive class imbalance as the positive class represented about 8% of the data .
- Also missing values was very challenging as a lot features has over 30% of its values are missing.
- Most of feature's distribution was highly skewed.
- I started to do some of EDA work then find hidden patterns and here is some [the full findings stated in the 00-EDA notebook] of them :

1-Most of loan applicants not in emergency state and married , working , females and applying for cash loans.

2-Laborers are the most occupation applying for loan and laborers with higher income are defaulters more than those with lower income.

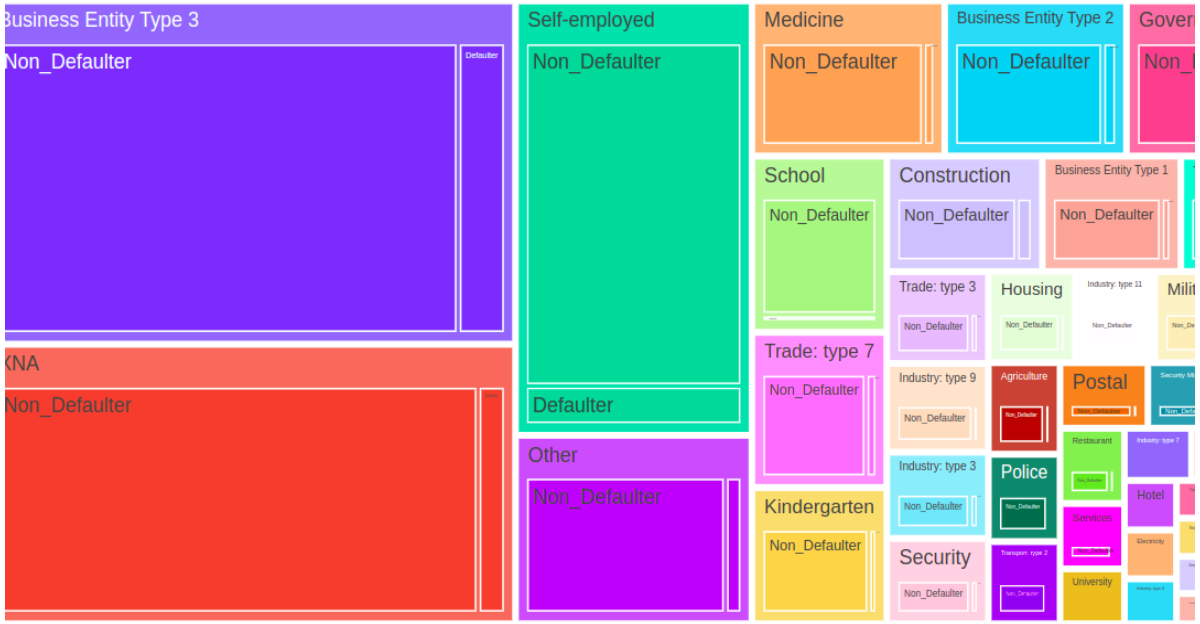
Cont..

- Most of Loan applicant has from 0-1 children which means child's burden is not the first motivation for applying for loans.



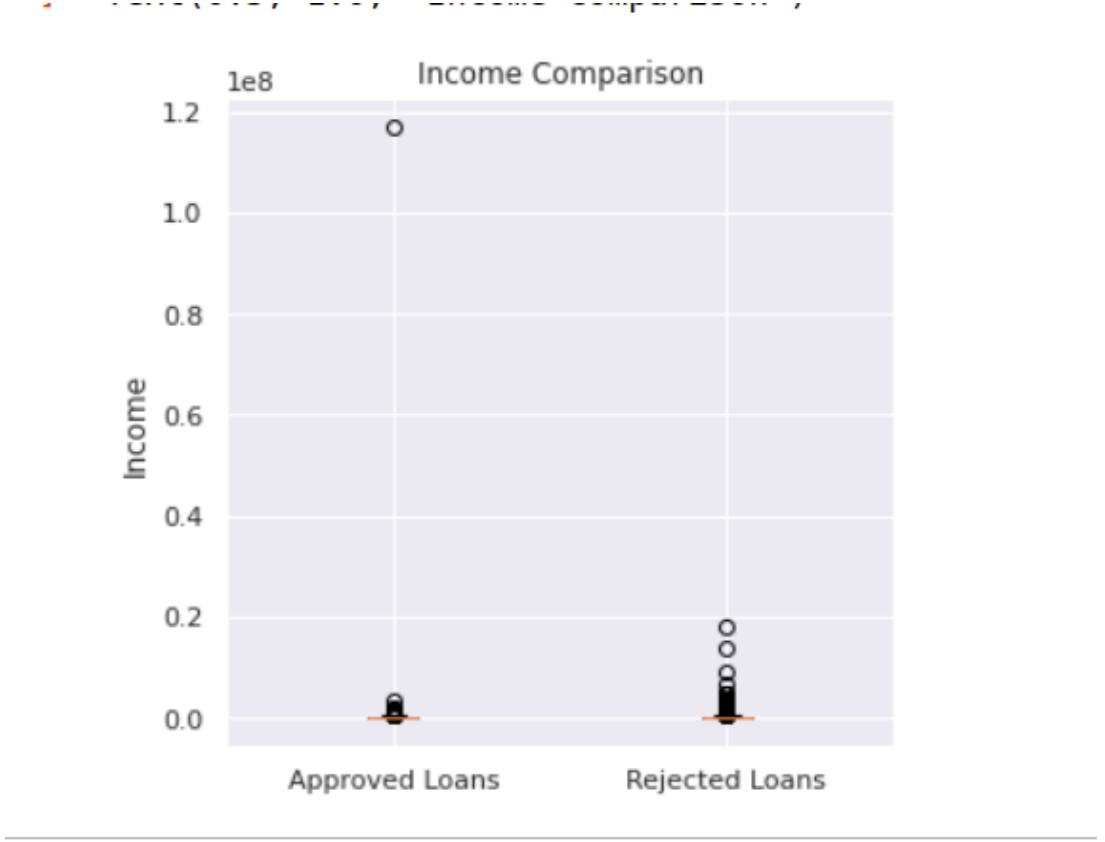
Cont...

- Most of loan applicants works for business entity and the 2nd highest segment are the self-employed after discarding the missing values group.



Cont...

- The median income for both defaulters and non-defaulters is nearly same with higher outliers in the approved loans which means at certain point non-defaulters have higher income.



Note: there is more visuals in the EDA notebook to consider but I will move forward to next step so that to avoid exceeding no of pages.

Feature preparation and engineering

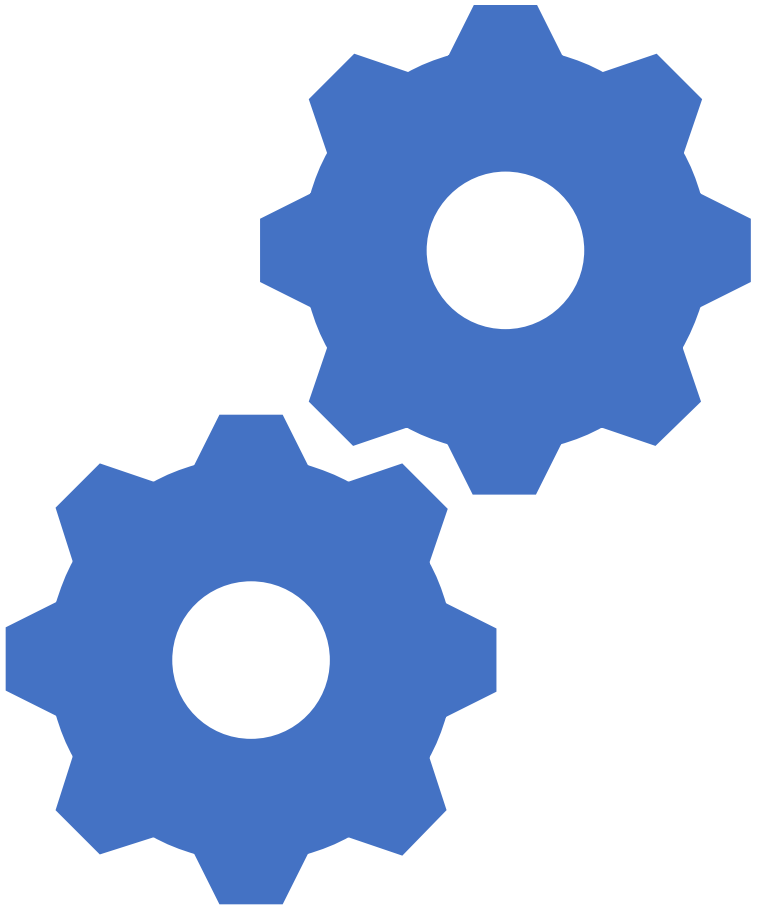
1-Started by discarding some features according to some criteria:

1.1-remove features with missing value percentage > 30%.

1.2-remove features with high multicollinearity according to corr matrix.

2-Engineer some relevant features:

I tried to get some ideas from previous kaggler's notebooks as I don't have enough domain knowledge help me in this part.

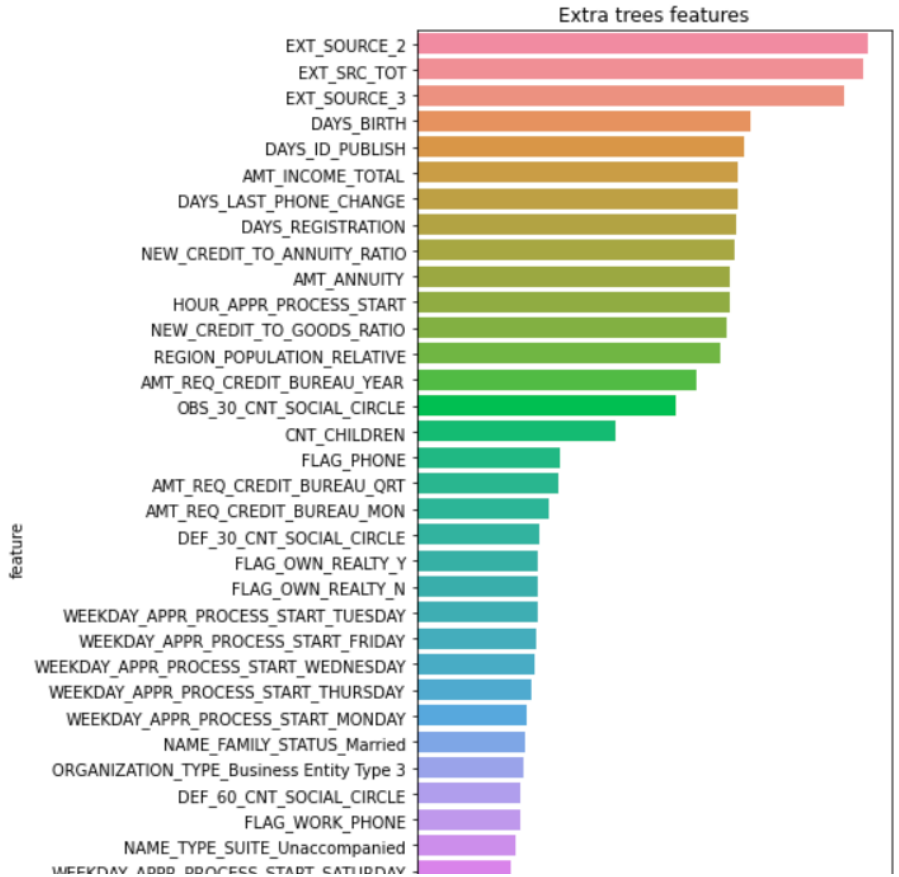


Cont...

3-then fix data problems by imputing outliers and encoding categorical features and.....etc.

4-Selecting relevant features according to two approaches:

4.1-Feature importance from base Extra trees model.



Cont....

4.2-most correlated features to the target in correlation matrix.

Model building

OUR MODEL SHOULD FOLLOW SOME SPECS:

1-ENSEMBLE MODEL THAT COULD CAPTURE COMPLEX PATTERN IN DATA.

2-GIVE OUTPUT AS PROBABILITY FOR INTERPRETABILITY.

3-SHOULD BE ROBUST TO OUTLIERS, SCALING, AND LOW VARIANCE.

4-NO CONSTRAINT ON SPEED OF INFERENCE.

```
In [38]: rf=RandomForestClassifier(max_depth=25)
         rf.fit(X_train,y_train)

Out[38]: RandomForestClassifier(max_depth=25)
```

```
In [39]: 
         prob_preds=rf.predict_proba(X_valid)
         print('AUC: ', roc_auc_score(y_valid, prob_preds[:,1]))

         AUC:  0.7201892821008867
```

Cont...

- 1-Baseline model:

Random Forrest without fine tuning or class weighting.

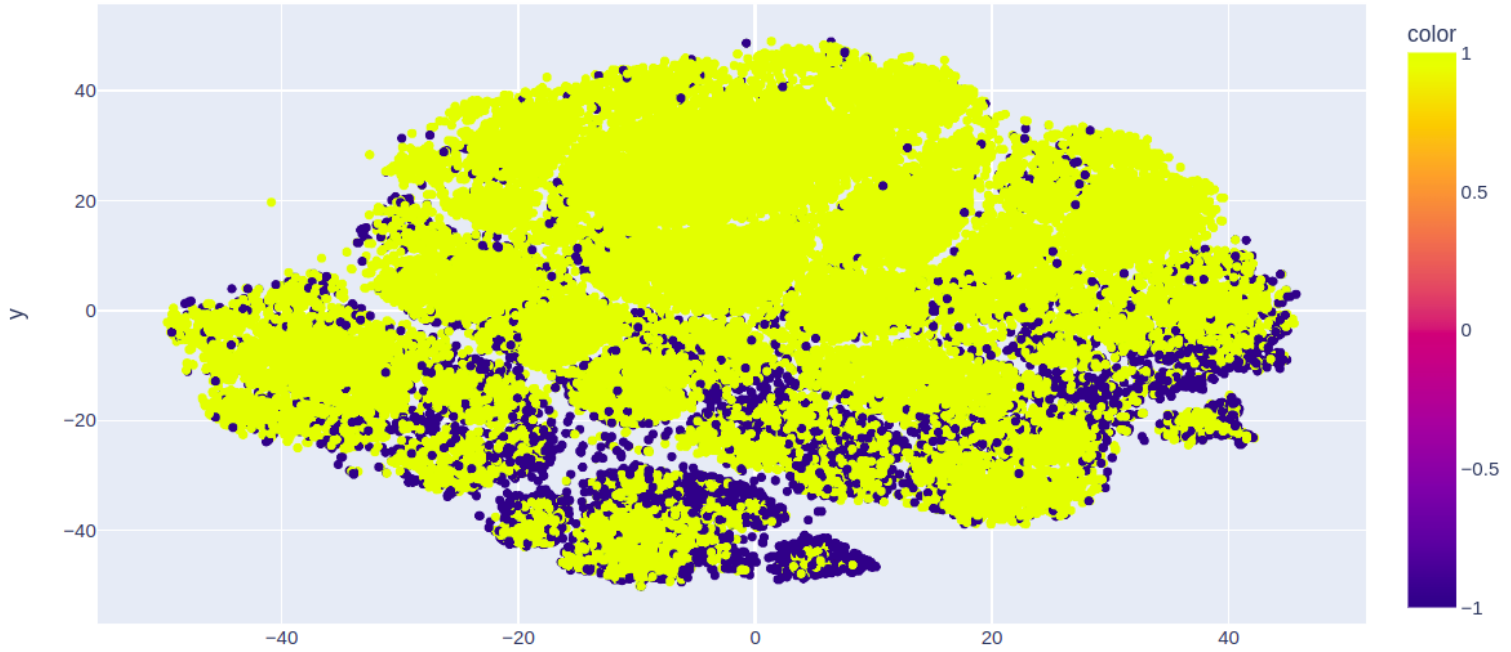
```
In [21]: predictions = pipeline.predict_proba(X_valid)
         print('AUC: ', roc_auc_score(y_valid, predictions[:,1]))

         AUC:  0.7615475864391146
```

Cont....

2-LGBM model with randomized search for hyperparameter tuning and class balancing by weight control.

Anomaly Detection

- Another way to formulate business problem is to treat it as Anomaly detection problem where non-defaulter points are usual points and treat defaulters as outliers, this may give better performance as despite optimizing the lgbm model it still has high rate of FN.



Cont...

- I used Isolated random Forrest model with 10% contamination and here is the result on test data.

Yellow points are for non-defaulters.

Blue points for defaulters.

Thank you