Summary of the report:

This project is based on random-forest classification model, also using smote and under-sampling to deal with bias training data. Even though the final result of testing data seems not so well, the data processing and the way of feature engineering still give a novel perspective. On the poster the group made, i can’t find out the score of Kaggle ‘s competition. Their code’s comment is well written, i can easily understand and read through all the code quickly. Their way to create new feature is quite useful. though their code, i can judge that they have a quite strong coding skill at all, but i think they might need to plot or print out their data structure after every step of the data processing, it really helps people to visualize the data and realize why you want to make such move.

In general, the prediction of the target might be hard to model by using random forest, they might need to find another model which good-fitting the data.we have to say the work on data cleaning goes well and i can learn a lot of stuff with it.

Strengths of the report

They use Sin-cos transformation for cyclic features, it is a alternative method to dual with those kind of non-numerical data. Maybe it will not have strong contribution to final prediction for this particular project, it is still helpful in many other way.usually we will just use one hot encoding function.

Another point is that they narrow some large type feature by checking the column with eyes.and they label and replace those category feature by self-coding, usually we done it by prepared function.

Also they check Multicollinearity , which might be a huge issue for later modeling. After we drop all the highly correlated feature, we can avoid overfitting problem and stabilize the model performance.

As the result in the poster,the accuracy_score is 92% . usually this means over-fitting problem(in this base-line model, definitely biased input data exist). Lately they find out the train data is unbalanced, and decide to use 2 tools to fix the problem. SMOTE is a method i never hear before, after i google it i know it is the regular method to fix the unbalance problem. But in the poster they haven’t

introduce anything about it, that makes a little bit regret.

Weaknesses of the report

They accidentally remove or replace some ‘abnormal ’ data, such as 365243 in day_employed. If we want to replace it , we need to add a new feature to state its abnormal status, if not we could miss its strength of prediction.

And i notice that the threshold of missing data dropping is not quite meaningful, if you drop nothing out, why we should do this step. Maybe should lower the threshold to 60%, then the data-set will be different.

They use RF for modeling, but i didn’t find where they test its model training parameter. Actually we need to try different parameter to find out the best one for the project

And i don’t get the point of selecting the top 10 importance of the feature when they fitting the RF model, in my opinion, this is a total mistake when we doing machining learning, we cant just limit the input train data like this which is consider a highly biased.

At last, I will recommend the team use CV to train the

model, not matter what kind of model you use. It is always welcome to apply CV to your project. It helps us to avoid the predictive result gap between train data and real world data.

Evaluation on clarity and quality of writing: 4.5

No typo found by me,only regret is the train data didn't plot out during the coding, the poster could be contain a little bit more about the model or technical tools they used

Evaluation on technical quality: 3

The technical stuff is not that bad, but i think it is not good for the project which aim to predict the risk probability. Using lots of package and statistical tools, and also define its own function to progress the data.

Overall rating: 4

Confidence on my assessment: 3