The American Express Campus Super Bowl

Team Details

Team Name : Dopa-mean

Name	College	Course	Batch Year	Roll No	Mobile Number	College Email Id
Vignesh Kumar	IIT Madras	B.Tech: Chemical M.Tech: Data Science (Dual)	2023	CH18B118	9445662199	ch18b118@smail.iitm.ac.in
Yashwant	IIT Madras	B.Tech: Aerospace M.Tech: Aerospace (Dual)	2023	AE18B115	8708347213	ae18b115@smail.iitm.ac.in
Krishn Kumar	IIT Madras	B.S: Biotechnology M.S: Biotechnology (Dual)	2023	BS18B019	6387590469	bs18b019@smail.iitm.ac.in

AMERICAN EXPRESS

Introduction – Team Members

Vignesh Kumar S

- Final year
- B.Tech in Chemical Eng
- M.Tech in Data Sciences (Dual)

- Data Science intern at Honeywell
- Data Science intern at Fleek
- Finalist in Shaastra Data Science Hackathon conducted by AstraZeneca
- Violinist at IITM Music Team

Yashwant

- Final Year
- Dual degree in Aerospace Engineering

- Machine Learning research Intern at TCS
- Machine Learning Intern at Neos Healthtech Pvt. Ltd.
- Was part of team of Hockey and Athletics team at IIT Madras

Krishn Kumar

- Final Year
- Dual degree in BioTechnology

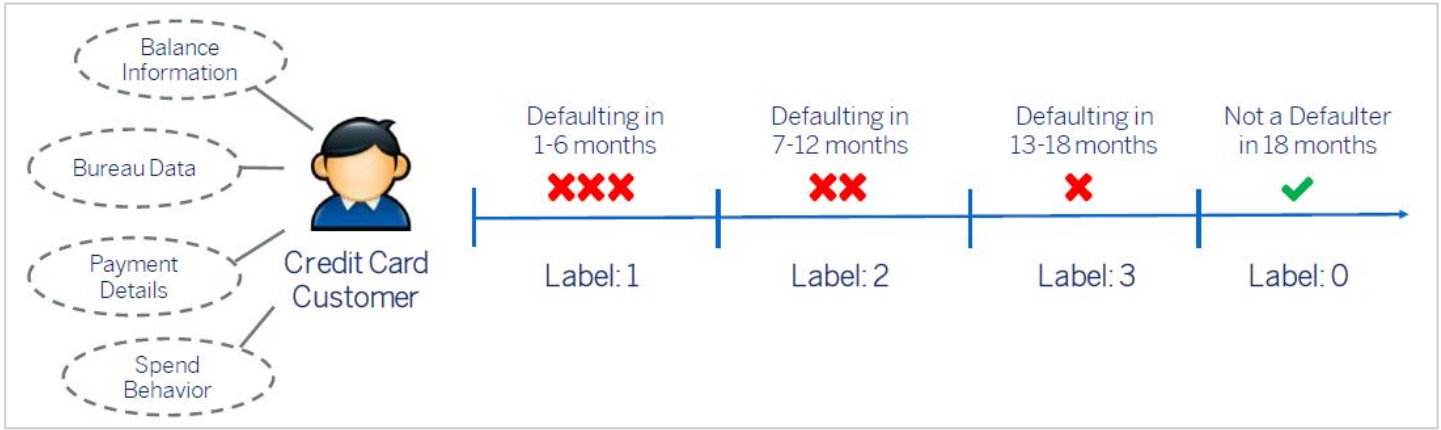
- Machine Learning Intern at Bitwise Academy
- Deep Learning Intern at Rudram Phronesis Infraz Pvt Ltd
- 4th runner-up in Data Science Hackathon organised by Dockship.io

AMERICAN EXPRESS

Problem Statement

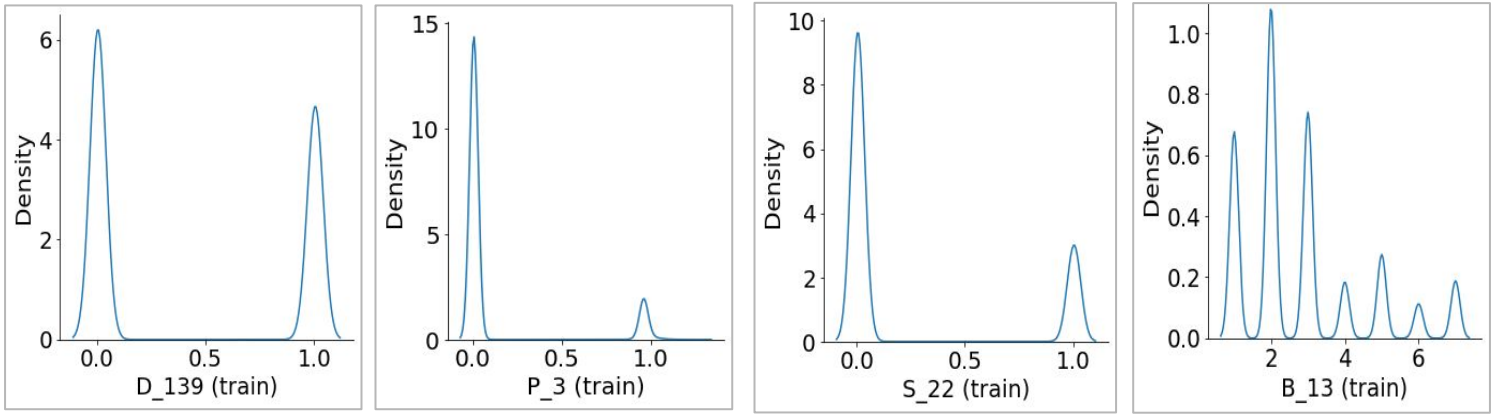
We predict the time to credit default for a Customer basis the variables provided across various categories

- In this, we need to figure out the right variables to use & create useful constructs from the same
- Basis these features and the choice of a modeling algorithm, we have to predict the time to default



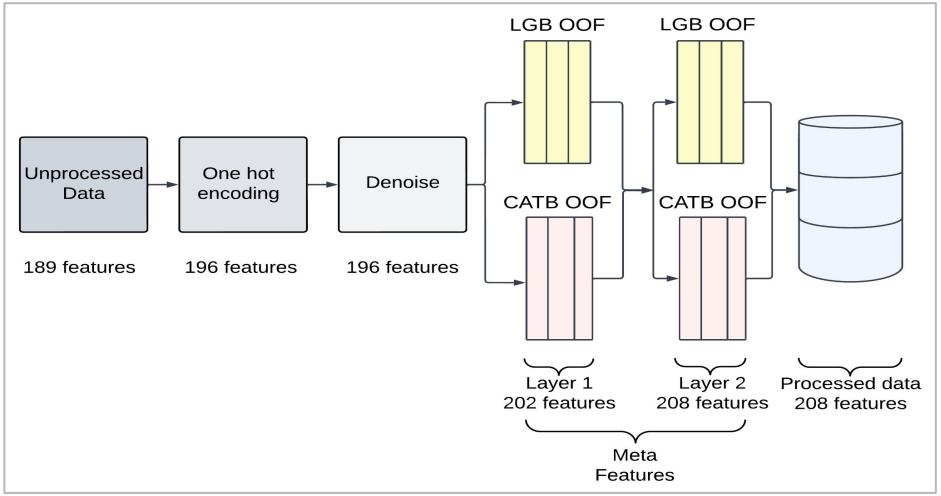
Assumptions / Insights

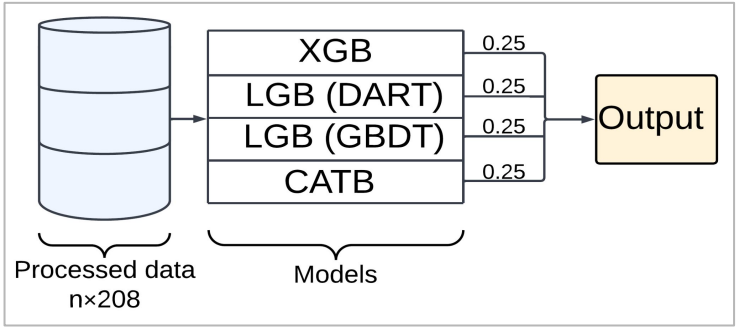
Denoising the data:



- We can observe **peaks at discrete points** in the features. We derive that there must be some noise present in the data in the range of [-0.01,0.01]
- The following function *np.floor(x*100)* is applied on all features which help to fix this in this context
- Since we work on tree-based methods, which on a higher level works by splitting the nodes, denoising helps in better learning

Methodology

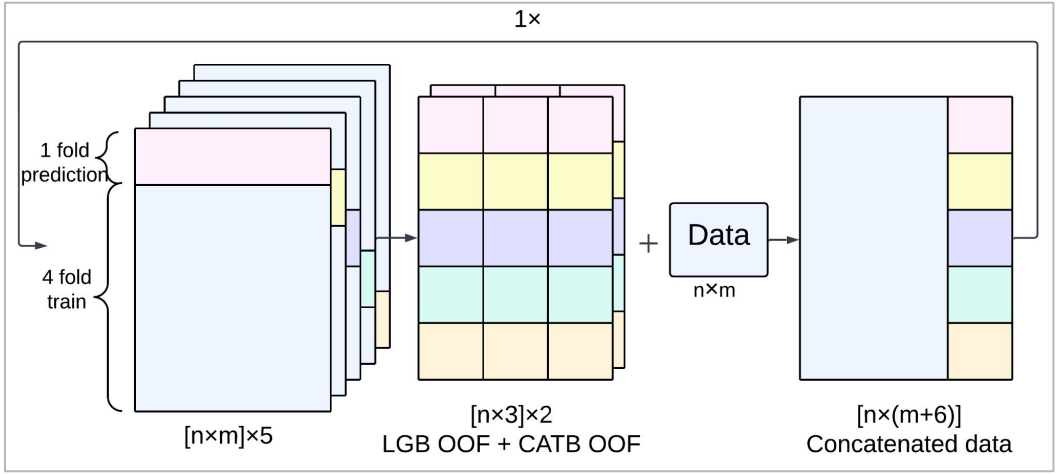




- **Tree-based methods** gave significantly better results compared to other methods
  - It also handles missing values
- Categorical Variables are one-hot encoded (col: D_36 and D_44)
- Data seems to have noise which is removed
- **Meta features** from out-of-fold (OOF) predictions are added to improve model performance
- The final output is a soft-voted prediction of 4 learners

Methodology (contd.)

- For generating meta-features, we take two classifiers, LightGBM and CatBoost, and get their predictions on **out-of-fold data** in 5 fold cross validation
- Each classifier results in 4 probability predictions. We take first three columns of probability predictions* and concatenate with data (meta-features)
- There are 2 classifiers and each add 3 features. The loop represents that this process is repeated one more time. Hence, 12 additional features are added overall.
- To get meta-features for Test/Validation data, the classifiers are retrained on whole train data and the predictions are made on Test/Validation data

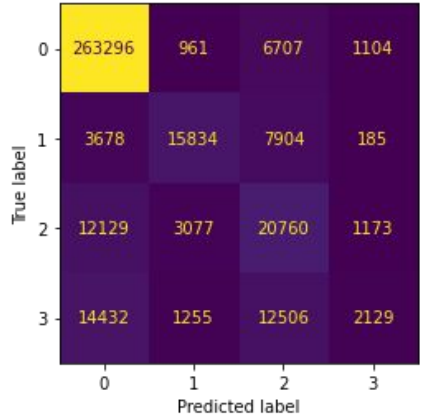


*fourth probability is 1-sum(three probabilities). Hence it is dropped

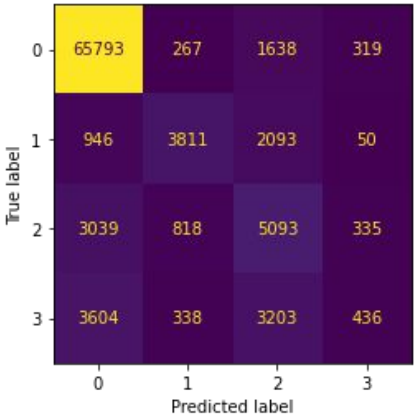
Model Performance

- Our proposed model solution is quite good at identifying records which don't default
- However, we can make few important observations
  - Our model mistakes quite a number of candidates which are actually label 1 as label2. By definition, label1 is defaulting within first 6 months and label2 is between 6-12 months.
  - As the duration gets longer, our model gets more mistaken and identifies the longer labels as 0 or the nearest label. Majority of label3 are either identified as label2 or label0
- From a general view point, we should not predict those that default as not default but it may be okay to make mistakes within label 1,2,3. By improving the class imbalance and accounting for a stricter metric while training, this aspect could be improved
- For future improvements, we can make additional classifiers if our predictions are either 2 or 3 to reassure if the records actually default

Train Data



Unseen Data



Results

- Analyzing the distributions of different features with respect to classes made us realize a strong imputation technique is required to handle the myriad of missing values. To counter this, tree-based gradient boosting methods come of great help as they handle missing values by default
- Many features showed spikes in distributions indicating categorical nature and presence of noise/errors. Any financial data is bound to have some categorical property considering the discrete property of money. This was balanced by adjusting for the floating point values to reduce noise
- Techniques such as feature engineering and model tuning over different variations of dataset were tried but all led to saturation of scores beyond a point
- Adding meta-features gave a significant boost in the overall model score, resulting in better learning with little tuning
- Ensembling further gave minor improvements to the overall results bringing in a score of **0.8241** on the test/submission data
- A list of experiments tried out is mentioned in brief in the pdf attached along with the code submission

AMERICAN EXPRESS