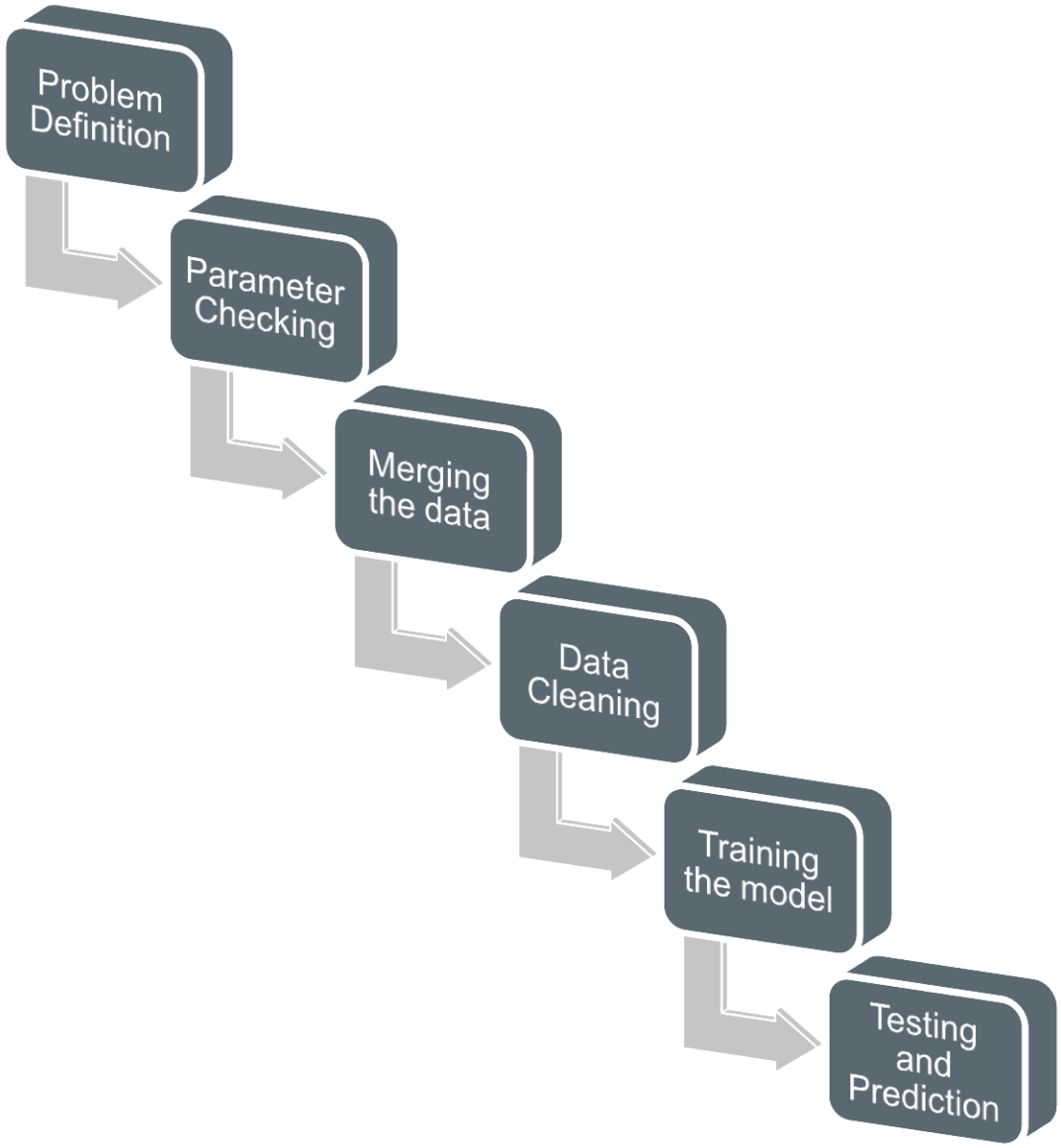
ELO MERCHANT CATEGORY RECOMMENDATION

By:
Ms. Anushri Laddha

WORK FLOW ANALYSIS



PROBLEM STATEMENT

- Elo, largest payment brand in Brazil, wants to provide recommendation to its customer based on their preferences and lifecycle. The models they have built till now are not specifically tailored for an individual.

GOAL TO BE ACHIEVED

- To develop algorithms to identify and serve the most relevant opportunities to individuals, by uncovering signal in customer loyalty

AVAILABLE DATASETS

- **Input Files:**
  - historical_transactions.csv
  - merchants.csv
  - new_merchant_transactions.csv
- **Train File**
  - train.csv
- **Test File**
  - test.csv
- **Submission File**
  - sample_submission.csv

TRAIN & TEST DATA

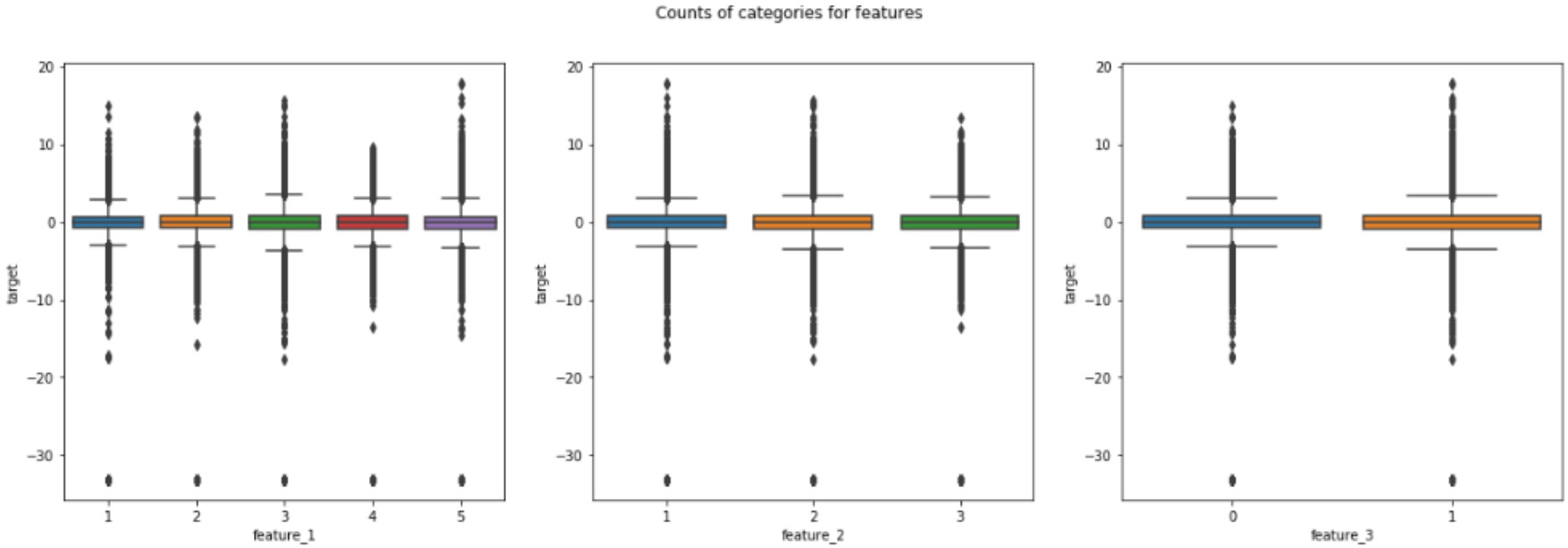
Description of Dataset:

- *card_id:* Unique card identifier
- *first_active_month:* 'YYYY-MM, month of first purchase
- *Feature_1:* Anonymized card categorical feature
- *Feature_2:* Anonymized card categorical feature
- *Feature_3:* Anonymized card categorical feature
- *Target:* Loyalty numerical score calculated 2 months after historical and evaluation period

Note: Test Data set doesn't contains target column initially. It has to be predicted after we train the model and then this column is to be appended in the file.

PARAMETER CHECKING

- We have 3 features in train.csv file. Checking if these 3 parameters can be used for training the data. Hence, we plot a boxplot for the purpose.



Conclusion: These 3 features happens to be similar in distribution across the target value(loyalty score). So its not enough to train model for finding target based on only these parameters. We need some extra features. We can get this by merging the historical transactions data and new merchants data because these data contains extra information about each card id.

MERGING THE DATA

- From the conclusion of parameter checking, it was concluded that, historical transactions data, new transactions data are to be merged with train and test data individually.
- But when checked for historical transactions and new transactions data, for one card_id it was found there were more than one entries in database. So direct merge cannot be done.
- Hence, aggregated data for each card_id, (i.e., grouping up by card_id and then aggregating) was used in merging
- For aggregation, different data types were aggregated differently.

Data Types	Aggregation Function
Int and Float data	Sum, Mean, Max, Min, Std
Categorical data	Sum, Mean
Datetime data	Max, Min
Object data	Nunique

DATA CLEANING

- After data merging process, the data was cleaned.
  - Unnecessary columns were dropped.
  - Null values were filled
  - All the values were checked to be either int or float

  The clean data was then stored in the file for further training of model.

TRAINING THE MODEL

- The final cleaned dataset was large (in gBs). Also, the target value was continuous.
- So, *Regression model* was needed to be used for training.
- Since the dataset was large, *LightGBM Regression* model was appropriate choice.
  - Light GBM is a gradient boosting framework that uses tree based learning algorithm.
  - Light GBM is prefixed as ‘Light’ because of its high speed.
  - Light GBM can handle the large size of data and takes lower memory to run.
  - Light GBM is popular is because it focuses on accuracy of results.

Note: Kfold model was used for cross validation of data. Root Mean Square Error was the metric to measure the error.

TESTING THE MODEL

- The trained model was then tested on the cleaned test file created after merging the data.
- For each fold, feature importance for each feature parameter used in training was calculated.
- Finally the data was loyalty score (target score) was predicted for the given card_id
- The predicted score was then stored in *sample_submission* file.

REFERENCES

- https://www.kaggle.com/artgor/elo-eda-and-models
- https://medium.com/@pushkarmandot/https-medium-com-pushkarmandot-what-is-lightgbm-how-to-implement-it-how-to-fine-tune-the-parameters-60347819b7fc
- https://lightgbm.readthedocs.io/en/latest/Parameters.html
- https://seaborn.pydata.org/generated/seaborn.boxplot.html

THANK YOU