CREDIT CARD DEFAULT PREDICTION

Subhradipta Paul &
Arnab Mitra

PROJECT DETAILS

❖**Problem Statement:**

Financial threats are displaying a trend in the credit risk of commercial banks as the incredible improvement in the financial industry has arisen. In this way, one of the biggest threats faces by commercial banks is the risk prediction of credit clients. The goal is to predict the probability of credit default based on the credit card owner's characteristics and payment history.

❖**Technologies:**

Machine Learning Technologies are used here to solve the problem.

❖**Domain :**

This problem is from Banking Domain.

OBJECTIVE & BENEFITS

❖ **Objective:**

Development of a predictive model for monitoring fraud insurance claims for Private Motor products. The model will determine whether a customer is placing a fraudulent insurance claim or not.

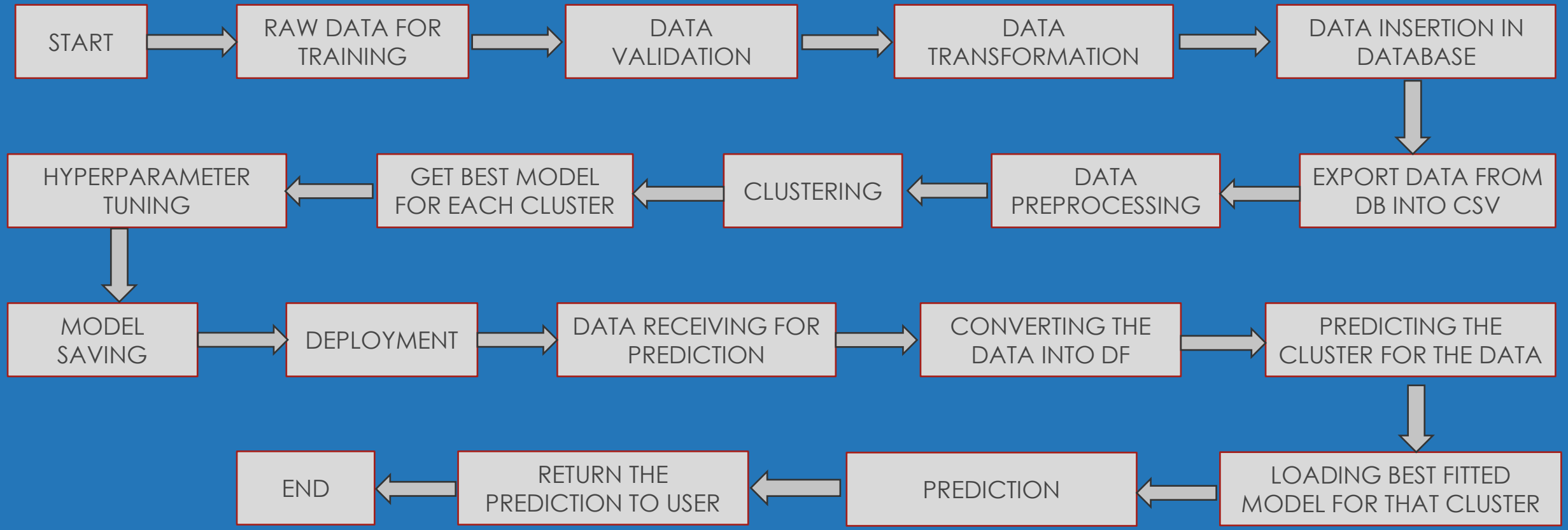
❖ **Benefits:**

- Detection of upcoming defaulter customers
- Gives better insight of the customer base.
- Helps in easy flow for managing resources.
- Manual inspection if the defaulter is identified.

Data Sharing Agreement

- Sample file name (ex fraudDetection_20062021_101010)
- Length of date stamp(8 digits)
- Length of timestamp (6 digits)
- Number of Columns (24 columns, 23 features & one label)
- Column names
- Column data type (all the data types are integer)

ARCHITECTURE



Data Validation & Transformation

- **Name Validation:** Validation of files name as per the DSA. We have created a regex pattern for validation. After it checks for date format and time format if these requirements are satisfied, we move such files to “Good Raw" or else “Bad Raw.“
- **Number of Columns:** Validation of the number of columns present in the files, and if it doesn't match then the file is moved to “Bad Raw.“
- **Name of Columns:** The name of the columns is validated and should be the same as given in the schema file. If not, then the file is moved to " Bad Raw".
- **Null columns validation:** If any of the columns in a file have all the values as NULL or missing, we discard such a file and move it to " Bad Raw ".

DATA INSERTION IN DATABASE

- **Table creation:** Table name “Good_Raw_Data" is created in the database for inserting the files. If the table is already present then new files are inserted in the same table.

- **Insertion of files in the table:** All the files in the " Good Raw " are inserted in the above-created table. If any file has an invalid data type in any of the columns, the file is not loaded in the table

MODEL TRAINING

❖ Data Export from Db :

- The accumulated data from DB is exported in CSV format for model training

❖ Data Preprocessing:

- Performing EDA to get an insight into data like identifying distribution, outliers, trends among data, etc.
- Check for null values in the columns. If present imputes the null values.
- Encode the categorical values with numeric values.
- Perform Standard Scalar to scale down the values.

❖ Clustering:

- KMeans algorithm is used to create clusters in the preprocessed data. The optimum number of clusters is selected by plotting the elbow plot, and for the dynamic selection of the number of clusters, we are using the KneeLocator function. The idea behind clustering is to implement different algorithms on the structured data.
- The Kmeans model is trained over preprocessed data, and the model is saved for further use in prediction

❖ Model Selection:

- After the clusters are created, we find the best model for each cluster. By using 2 algorithms “SVM” and “Random Forest". For each cluster both the hyper tunned algorithms are used. We calculate the AUC scores for both models and select the model with the best score. Similarly, the model is selected for each cluster. All the models for every cluster are saved for use in prediction.

PREDICTION

- The testing files are shared in the batches and we perform the same validation operations, data transformation and data insertion on them.
- The accumulated data from db is exported in csv format for prediction
- We perform data pre-processing techniques on it.
- KMeans model created during training is loaded and clusters for the preprocessed data is predicted
- Based on the cluster number respective model is loaded and is used to predict the data for that cluster.
- Once the prediction is done for all the clusters. The predictions are saved in csv format and shared.

THANK YOU