Credit Card Default Prediction

Problem Statement

Financial threats are displaying a trend about 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 credit card owner's characteristics and payment history.

Data Description

The client will send data in multiple sets of files in batches at a given location. The data has been extracted from the census bureau.

The data contains 32561 instances with the following attributes:

Features:

1. **LIMIT_BAL:** continuous Credit Limit of the person.
2. **SEX:** Categorical: 1 = male; 2 = female
3. **EDUCATION:** Categorical: 1 = graduate school; 2 = university; 3 = high school; 4 = others
4. **MARRIAGE:** 1 = married; 2 = single; 3 = others
5. **AGE**-num: continuous.
6. **PAY_0 to PAY_6:** History of past payment. We tracked the past monthly payment records (from April to September, 2005)
7. **BILL_AMT1 to BILL_AMT6:** Amount of bill statements.
8. **PAY_AMT1 to PAY_AMT6:** Amount of previous payments.

Target Label:

Whether a person shall default in the credit card payment or not.

1. default payment next month: Yes = 1, No = 0.

Credit Card Default Prediction

Apart from training files, we also require a "schema" file from the client, which contains all the relevant information about the training files such as:

Name of the files, Length of Date value in Filename, Length of Time value in Filename, Number of Columns, Name of the Columns, and their datatype.

.

Data Validation

In this step, we perform different sets of validation on the given set of training files.

1. Name Validation- We validate the name of the files based on the given name in the schema file. We have created a regex pattern as per the name given in the schema file to use for validation. After validating the pattern in the name, we check for the length of date in the file name as well as the length of time in the file name. If all the values are as per requirement, we move such files to "Good_Data_Folder" else we move such files to "Bad_Data_Folder."
2. Number of Columns - We validate the number of columns present in the files, and if it doesn't match with the value given in the schema file, then the file is moved to "Bad_Data_Folder."
3. 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_Data_Folder".
4. The data type of columns - The data type of columns is given in the schema file. This is validated when we insert the files into Database. If the data type is wrong, then the file is moved to "Bad_Data_Folder".
5. Null values in columns - 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_Data_Folder".

Data Insertion in Database

1. Database Creation and connection - Create a database with the given name passed. If the database is already created, open the connection to the database.

Credit Card Default Prediction

2. Table creation in the database - Table with name - "Good_Data", is created in the database for inserting the files in the "Good_Data_Folder" based on given column names and data type in the schema file. If the table is already present, then the new table is not created and new files are inserted in the already present table as we want training to be done on new as well as old training files.

3. Insertion of files in the table - All the files in the "Good_Data_Folder" are inserted in the above-created table. If any file has invalid data type in any of the columns, the file is not loaded in the table and is moved to "Bad_Data_Folder".

Model Training

1) Data Export from Db - The data in a stored database is exported as a CSV file to be used for model training.

2) Data Preprocessing

a) Check for null values in the columns. If present, impute the null values using the KNN imputer.

b) Check if any column has zero standard deviation, remove such columns as they don't give any information during model training.

3) 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 "KneeLocator" function. The idea behind clustering is to implement different algorithms, to train data in different clusters. The Kmeans model is trained over preprocessed data and the model is saved for further use in prediction.

4) Model Selection - After clusters are created, we find the best model for each cluster. We are using two algorithms, "Random Forest" and "XGBoost". For each cluster, both the algorithms are passed with the best parameters derived from GridSearch. 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 Data Description

Client will send the data in multiple set of files in batches at a given location. Data will contain Wafer names and 590 columns of different sensor values for each wafer.

Credit Card Default Prediction

Apart from prediction files, we also require a "schema" file from client which contains all the relevant information about the training files such as:

Name of the files, Length of Date value in FileName, Length of Time value in FileName, Number of Columns, Name of the Columns and their data type.

Data Validation

In this step, we perform different sets of validation on the given set of training files.

1) Name Validation- We validate the name of the files on the basis of given Name in the schema file. We have created a regex pattern as per the name given in schema file, to use for validation. After validating the pattern in the name, we check for length of date in the file name as well as length of time in the file name. If all the values are as per requirement, we move such files to "Good_Data_Folder" else we move such files to "Bad_Data_Folder".

2) Number of Columns - We validate the number of columns present in the files, if it doesn't match with the value given in the schema file then the file is moved to "Bad_Data_Folder".

3) Name of Columns - The name of the columns is validated and should be same as given in the schema file. If not, then the file is moved to "Bad_Data_Folder".

4) Data type of columns - The data type of columns is given in the schema file. This is validated when we insert the files into Database. If data type is wrong then the file is moved to "Bad_Data_Folder".

5) Null values in columns - If any of the columns in a file has all the values as NULL or missing, we discard such file and move it to "Bad_Data_Folder".

Data Insertion in Database

1) Database Creation and connection - Create database with the given name passed. If the database is already created, open the connection to the database.

2) Table creation in the database - Table with name - "Good Data", is created in the database for inserting the files in the "Good_Data_Folder" on the basis of given column names and data type in the schema file. If table is already present then new table is not created, and new files are inserted the already present table as we want training to be done on new as well old training files.

Credit Card Default Prediction

3) Insertion of files in the table - All the files in the "Good_Data_Folder" are inserted in the above-created table. If any file has invalid data type in any of the columns, the file is not loaded in the table and is moved to "Bad_Data_Folder".

Prediction

1) Data Export from Db - The data in the stored database is exported as a CSV file to be used for prediction.

2) Data Preprocessing

a) Check for null values in the columns. If present, impute the null values using the KNN imputer.

b) Check if any column has zero standard deviation, remove such columns as we did in training.

3) Clustering - KMeans model created during training is loaded, and clusters for the preprocessed prediction data is predicted.

4) Prediction - Based on the cluster number, the respective model is loaded and is used to predict the data for that cluster.

5) Once the prediction is made for all the clusters, the predictions along with the Wafer names are saved in a CSV file at a given location and the location is returned to the client.

Credit Card Default Prediction

Q & A:

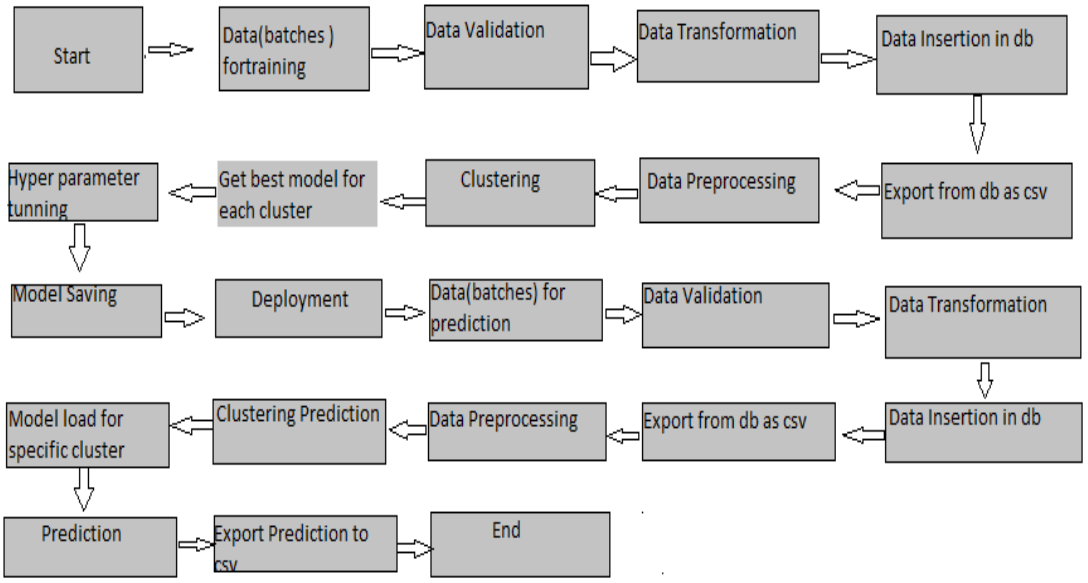
Q1) What's the source of data?

The data for training is provided by the client in multiple batches and each batch contain multiple files

Q 2) What was the type of data?

The data was the combination of numerical and Categorical values.

Q 3) What's the complete flow you followed in this Project?



Q 4) After the File validation what you do with incompatible file or files which didn't pass the validation? Files like these are moved to the Achieve Folder and a list of these files has been shared with the client and we removed the bad data folder.

Credit Card Default Prediction

Q 5) How logs are managed?

We are using different logs as per the steps that we follow in validation and modeling like File validation log , Data Insertion ,Model Training log , prediction log etc.

Q 6) What techniques were you using for data pre-processing?

▶ Removing unwanted attributes

▶ Visualizing relation of independent variables with each other and output variables

▶ Checking and changing Distribution of continuous values

▶ Removing outliers

▶ Cleaning data and imputing if null values are present.

▶ Converting categorical data into numeric values.

▶ Scaling the data

Q 7) How training was done or what models were used?

▶ Before diving the data in training and validation set we performed clustering over fit to divide the data into clusters.

▶ As per cluster the training and validation data were divided.

▶ The scaling was performed over training and validation data

Credit Card Default Prediction

▶ Algorithms like SVM , XGBoost were used based on the recall final model was used for each cluster and we saved that model .

Q 8) How Prediction was done?

The testing files are shared by the client .We Perform the same life cycle till the data is clustered .Then on the basis of cluster number model is loaded and perform prediction. In the end we get the accumulated data of predictions.