HOME CREDIT

CREDIT RISK MODEL STABILITY

TEAM

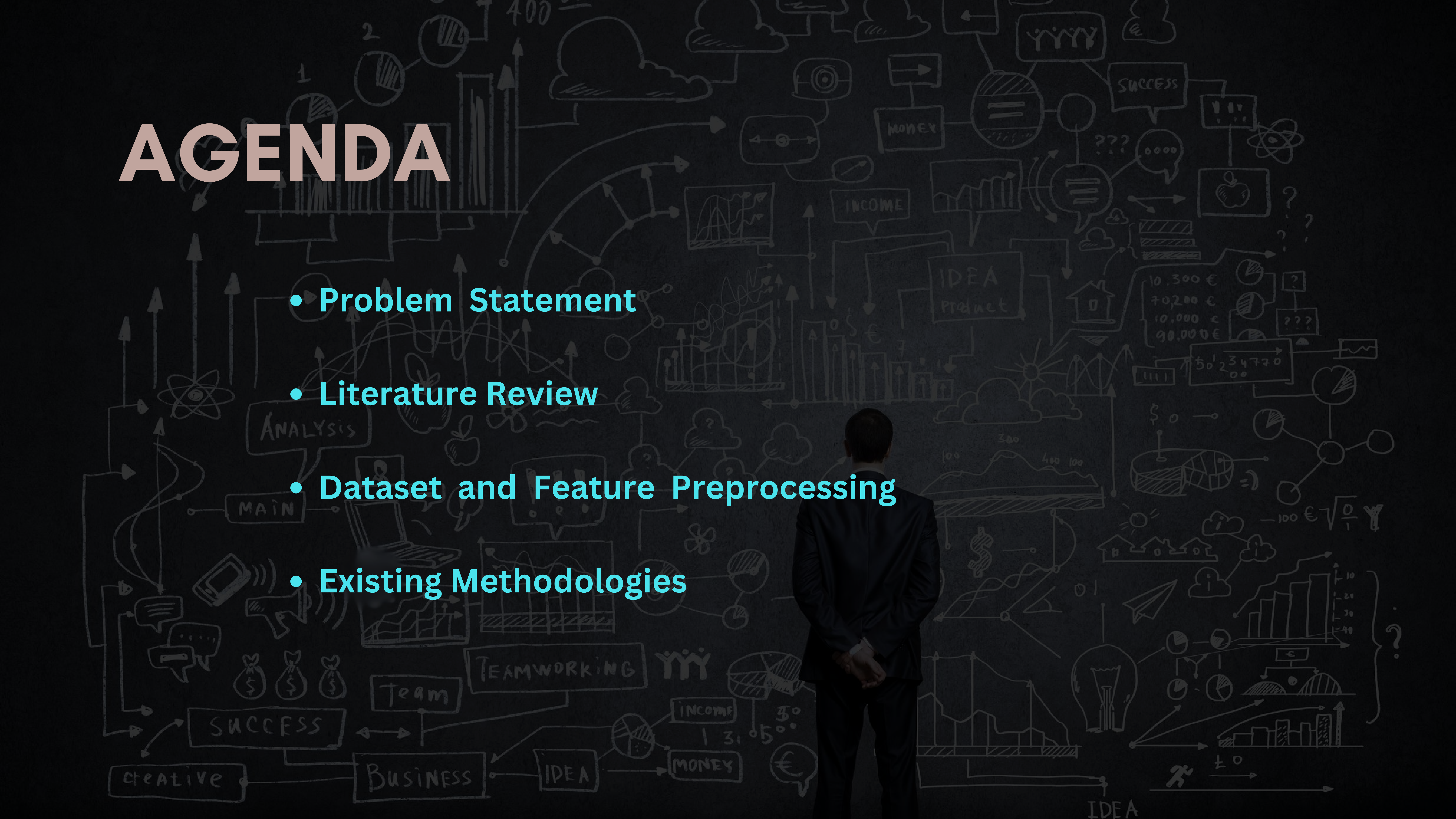
Krishna

Manan

Sudershan

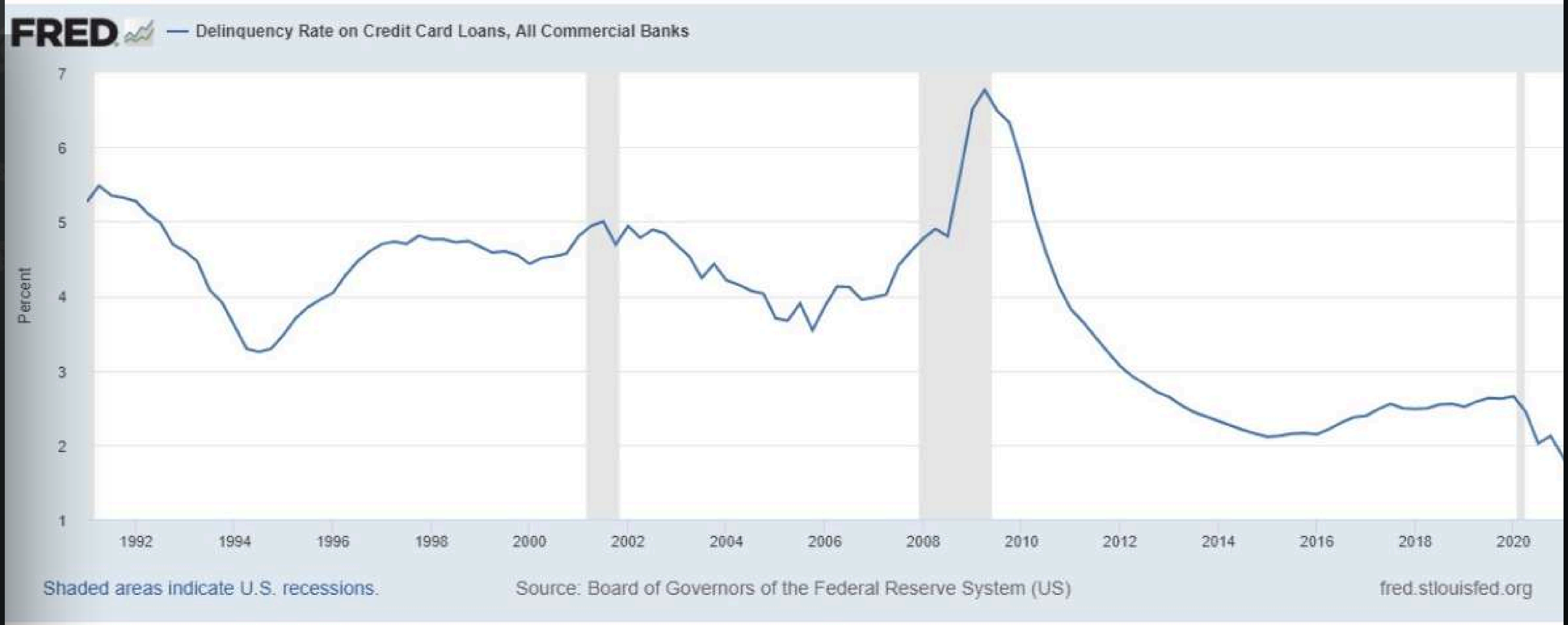
AGENDA

- Problem Statement
- Literature Review
- Dataset and Feature Preprocessing
- Existing Methodologies



PROBLEM STATEMENT

Developing a data-driven solution to accurately assess default risks for potential clients with little to no credit history, enabling consumer finance providers like Home Credit to expand loan accessibility while maintaining stability in loan performance.



Source: Board of Governors of the Federal Reserve System (US)

Figure 1: Delinquency Rate on Credit card loans [1]

LITERATURE REVIEW

RESEARCH PAPER 1

https://iopscience.iop.org/article/10.1088/1742-6596/1651/1/012111/pdf

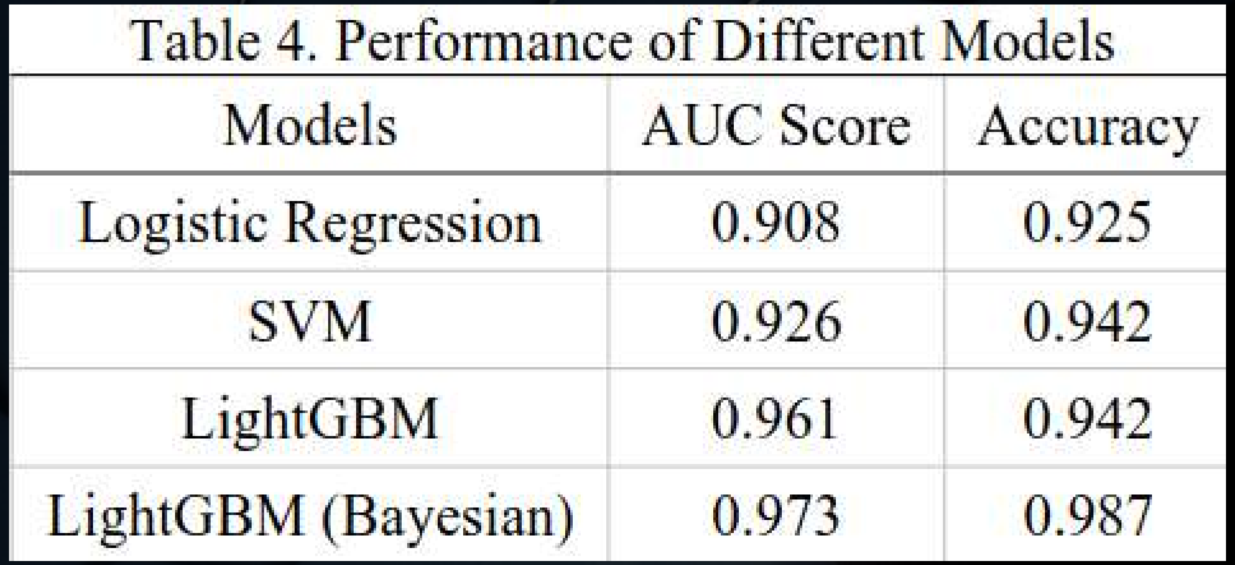


Table 4. Performance of Different Models

Models	AUC Score	Accuracy
Logistic Regression	0.908	0.925
SVM	0.926	0.942
LightGBM	0.961	0.942
LightGBM (Bayesian)	0.973	0.987

Aim of Study: Develop an effective predictive model for credit card fraud detection.

Dataset Used: IEEE-CIS Fraud Detection dataset from Vesta Corporation, containing over 1 million transaction records for about 28,000 credit cards.

Model Used and Features Extracted: Employed a LightGBM classification model with Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) optimizations. Key features engineered included aggregating transaction data by time, adding statistical features, one-hot encoding categorical variables, and recursive feature elimination.

Key Findings: The LightGBM model with Bayesian hyperparameter tuning achieved an AUC score of 0.973 and accuracy of 0.987, outperforming logistic regression (AUC 0.908) and SVM (AUC 0.926) models. Effective feature engineering and automated hyperparameter search contributed to the superior performance.

RESEARCH PAPER 2

https://arxiv.org/pdf/2110.02206.pdf

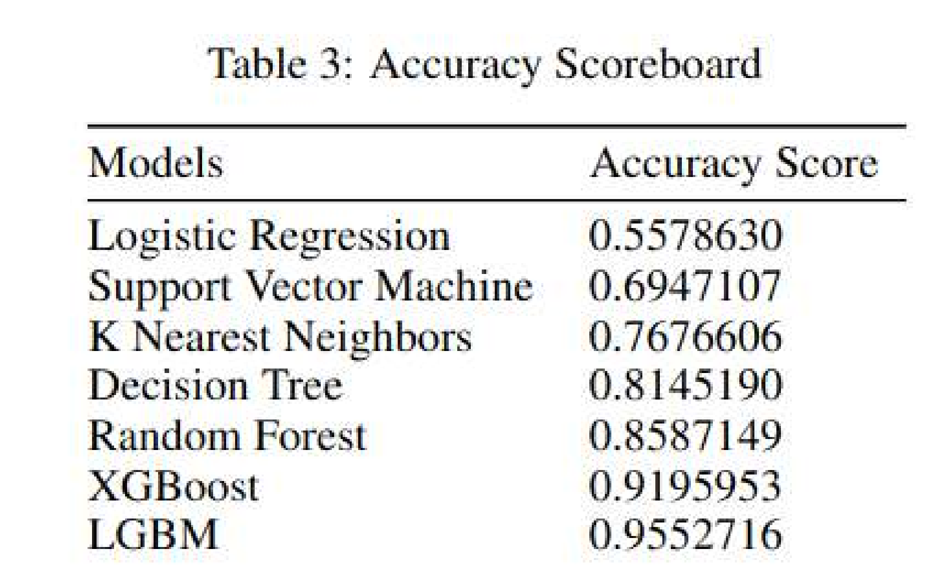


Table 3: Accuracy Scoreboard

Models	Accuracy Score
Logistic Regression	0.5578630
Support Vector Machine	0.6947107
K Nearest Neighbors	0.7676606
Decision Tree	0.8145190
Random Forest	0.8587149
XGBoost	0.9195953
LGBM	0.9552716

Aim of Study: Build a contemporary credit scoring model to forecast credit defaults for unsecured lending (credit cards) by employing machine learning techniques.

Dataset Used: Two datasets - one with customer personal details (18 features) and another with credit history (1,048,575 rows, 3 features capturing monthly payment status).

Model Used and Features Extracted: Evaluated 7 machine learning classification models - Logistic Regression, Support Vector Machines, K-Nearest Neighbors, Decision Tree, Random Forest, XGBoost, and LightGBM. Handled imbalanced data using SMOTE oversampling technique. Performed feature engineering like label encoding categorical variables.

Key Findings: The LightGBM classifier model outperformed other techniques with an accuracy of 95.53% and AUC of 0.99. Its advantages include capability to handle large datasets efficiently, lower memory usage, higher training speed, and better management of high dimensionality through approaches like gradient-based one-side sampling and exclusive feature bundling.

DATASET

DATA COLLECTION

- Data was downloaded from Kaggle.

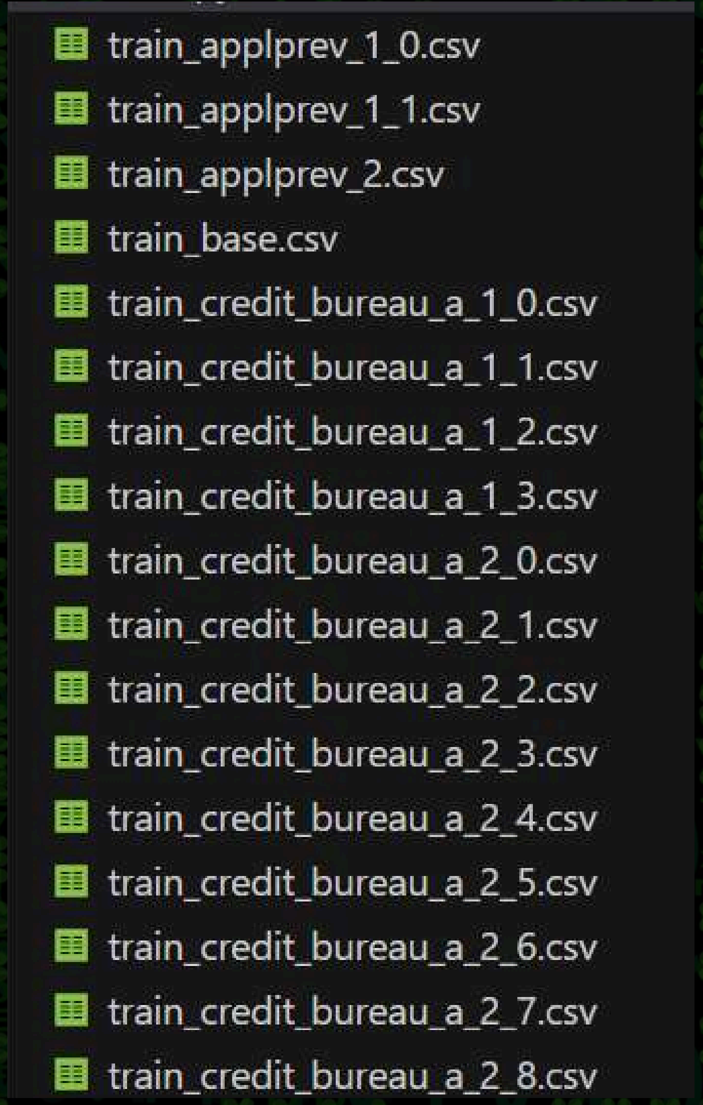
- Data Sources:
  ⟶ **Internal:** Generated within the organization.

  ⟶ **External:** From sources like tax registries and credit bureaus.

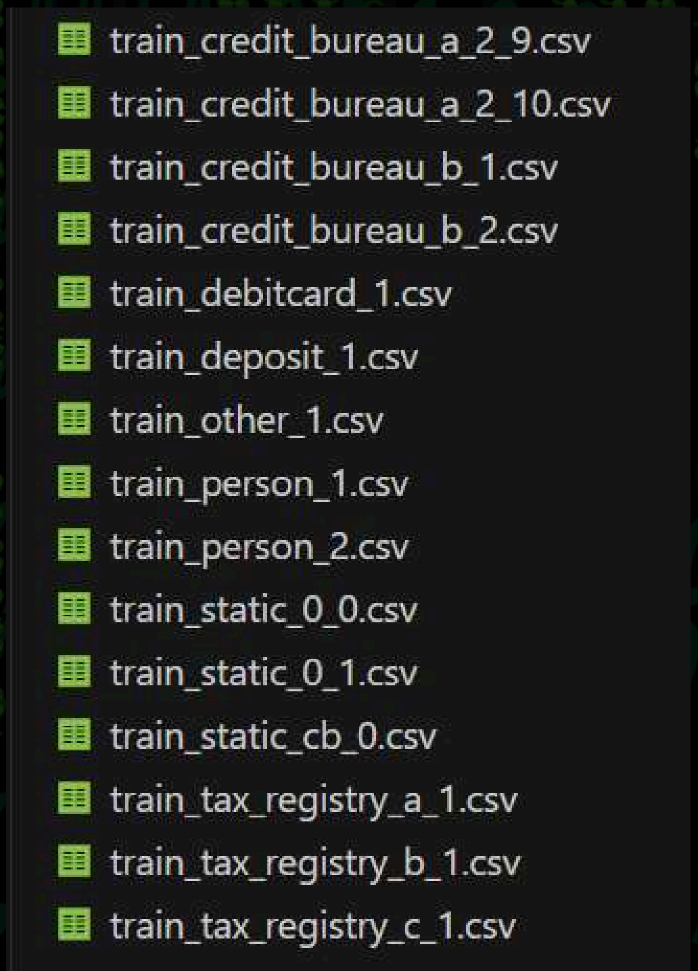
- Training dataset has entries of "1526559" users

- Training and Test Dataset has 32 files encompassing data of multiple features.

- Test Dataset has "10" entries



```
train_applprev_1_0.csv
train_applprev_1_1.csv
train_applprev_2.csv
train_base.csv
train_credit_bureau_a_1_0.csv
train_credit_bureau_a_1_1.csv
train_credit_bureau_a_1_2.csv
train_credit_bureau_a_1_3.csv
train_credit_bureau_a_2_0.csv
train_credit_bureau_a_2_1.csv
train_credit_bureau_a_2_2.csv
train_credit_bureau_a_2_3.csv
train_credit_bureau_a_2_4.csv
train_credit_bureau_a_2_5.csv
train_credit_bureau_a_2_6.csv
train_credit_bureau_a_2_7.csv
train_credit_bureau_a_2_8.csv
```



```
train_credit_bureau_a_2_9.csv
train_credit_bureau_a_2_10.csv
train_credit_bureau_b_1.csv
train_credit_bureau_b_2.csv
train_debitcard_1.csv
train_deposit_1.csv
train_other_1.csv
train_person_1.csv
train_person_2.csv
train_static_0_0.csv
train_static_0_1.csv
train_static_cb_0.csv
train_tax_registry_a_1.csv
train_tax_registry_b_1.csv
train_tax_registry_c_1.csv
```

DATA COMPOSITION

Depth Levels:

- **Depth 0:** Static features tied to case_id.
- **Depth 1:** Historical records associated with case_id.
- **Depth 2:** Historical records indexed by two groups (num_group_1 & 2).

Predictors:

Features used for modeling, with transformations denoted by letters.

Aggregation will be required for historical records.

- **P -** Transform DPD (Days past due)
- **M -** Masking categories
- **A -** Transform amount
- **D -** Transform date
- **T -** Unspecified Transform
- **L -** Unspecified Transform

ETHICAL CONCERNS

Ethical concerns in this credit risk prediction project include maintaining data privacy and confidentiality by anonymizing or securing sensitive personal information. Additionally, assessing and addressing the societal impacts, such as financial inclusion and access to credit, should be considered.

FEATURE PREPROCESSING

- For numerical features we used Median and for categorical features we used Mode to fill null values.
- We used standardization.
- For encoding, we used target encoding.

FEATURE SELECTION

- Removed features with over 70% missing values.
- Excluded categorical features with one unique value or over 200 unique values.
- Grouped highly correlated features (correlation ≥ 0.8).
- Selected one representative feature based on highest unique values from each correlated group.

AGGREGATION

1. Numerical Features Aggregation:

   - Add summary statistics including maximum, last, and mean values for numerical features.

2. Date Features Aggregation:

   - Incorporate summary statistics such as maximum, last, and mean dates for date-type features.

3. String Features Aggregation:

   - Integrate summary statistics like maximum and last values for string-type features.

4. Other Features Aggregation:

   - Include summary statistics such as maximum and last values for features of other types.

5. Count Features Aggregation:

   - Integrate a summary statistic, specifically the maximum count, for numerical group features.

OUR APPROACH

SOME IMPORTANT FUCNTIONS WE USED

1. Horizontal_Stacking
2. Memory_optimization
3. Optimizing memory by downcasting the columns
4. reduce_group
5. typecasting
6. handling_dates

ML METHODOLOGY: LIGHTGBM

Open-source distributed framework by Microsoft
Designed for high performance, scalability & accuracy
Based on optimized decision tree algorithms

KEY FEATURES:

Gradient-based One-Side Sampling (GOSS):

- Retains instances with large gradients
- Optimizes memory & training time

Histogram-based Tree Construction

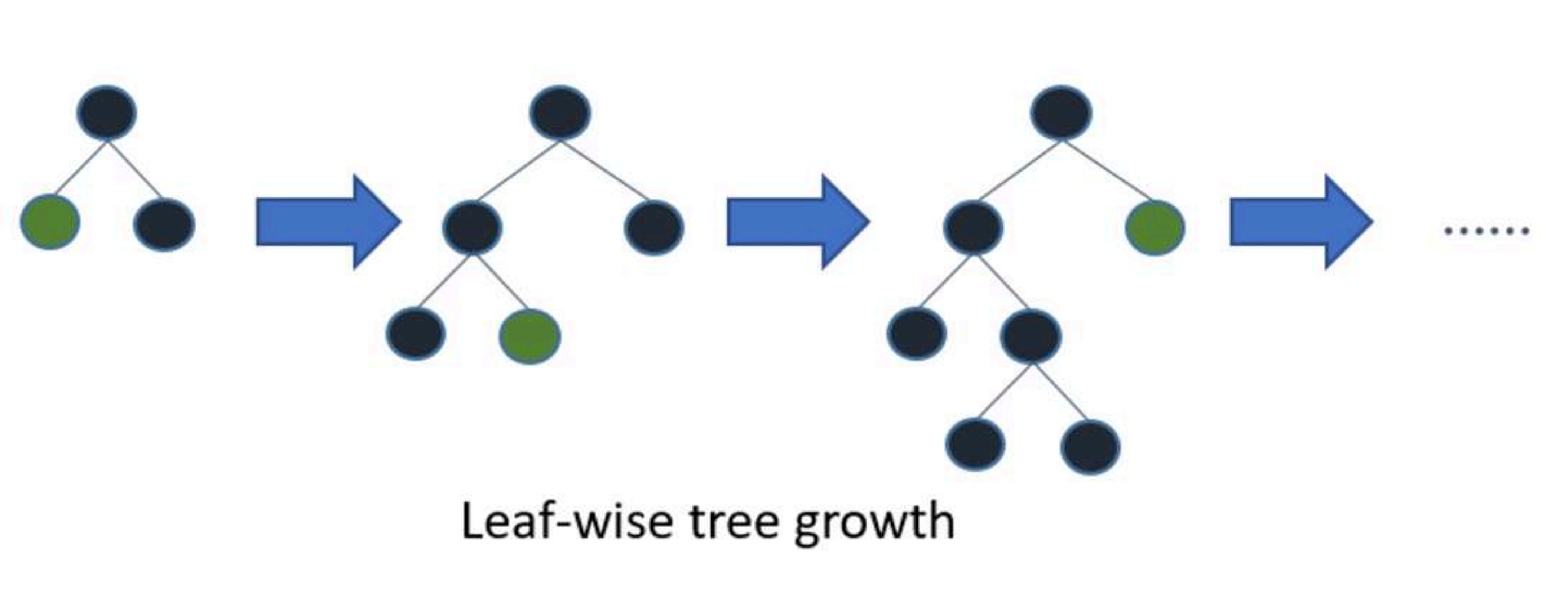
- Efficient decision tree building
- Faster than level-wise growth

Leaf-wise Tree Growth

- Better accuracy than depth-wise growth

Efficient Data Storage Formats

- Reduces memory footprint & accelerates training



Leaf-wise tree growth

WHY LIGHTGBM?

We have used Lightgbm because:

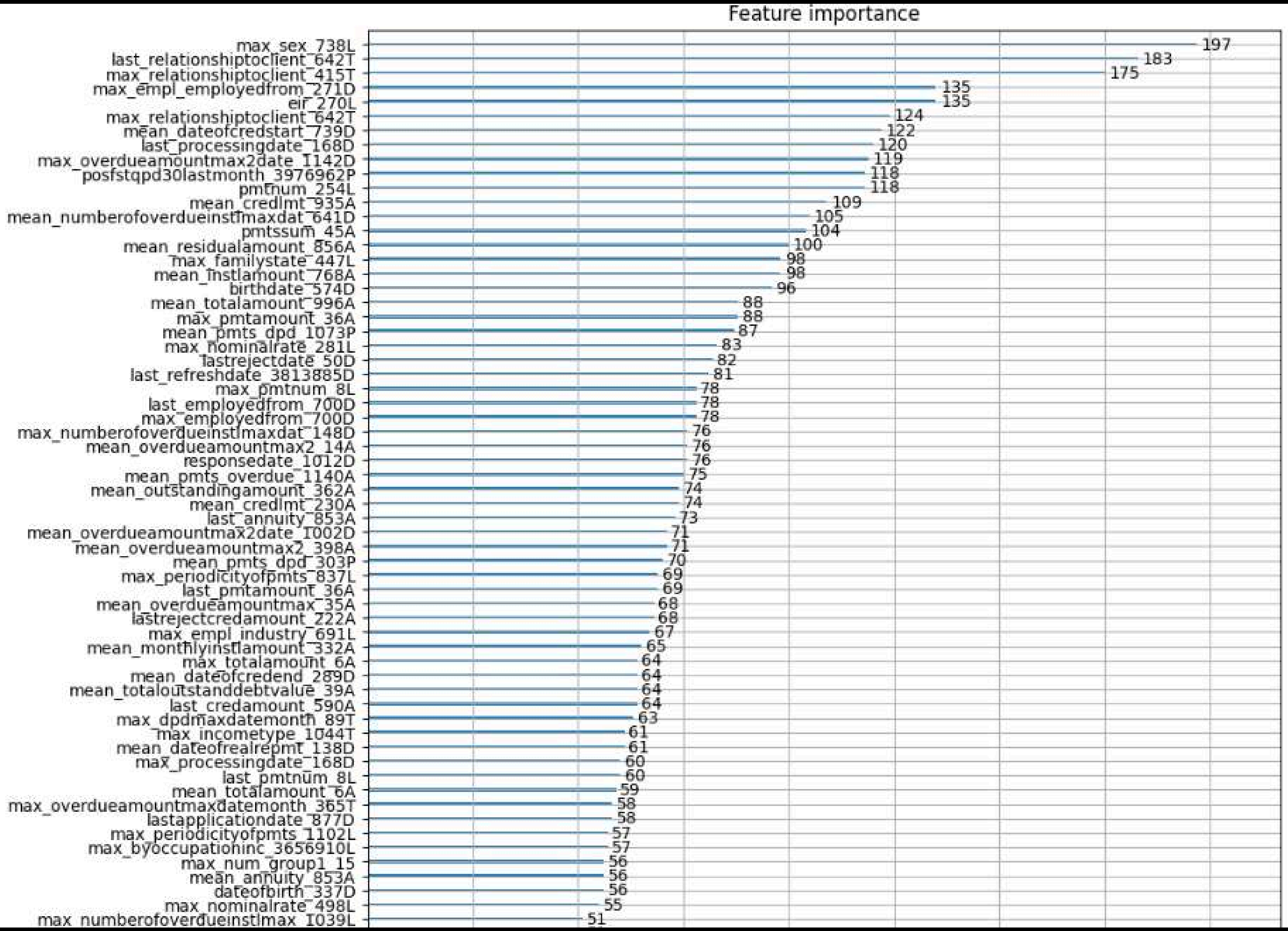
- **Faster Speed and Higher Accuracy:** LightGBM algorithm offers faster training times and higher accuracy compared to other gradient boosting algorithms, making it suitable for large-scale datasets and time-sensitive applications.

- **Lower Memory Usage:** LightGBM is designed to optimize memory usage efficiently, allowing it to handle large datasets with minimal memory requirements, which can lead to cost savings and improved performance.

- **Better Accuracy:** LightGBM's innovative algorithms, such as leaf-wise tree growth and histogram-based learning, contribute to better accuracy in model predictions, resulting in more reliable and precise outcomes.

- **Support for Parallel and Distributed GPU Learning:** LightGBM supports parallel training on multi-core CPUs and distributed GPU learning, enabling efficient utilization of computational resources and faster training times for large-scale datasets.

- **Capability to Handle Large-Scale Data:** LightGBM is capable of handling large-scale datasets efficiently, thanks to its optimization techniques and support for parallel processing, making it suitable for big data applications in various industries.


Feature importance
max_sex_738L 197
last_relationshiptoclient_642T 183
max_relationshiptoclient_415T 175
max_empl_employedfrom_271D 135
eir_270L 135
max_relationshiptoclient_642T 124
mean_dateofcredstart_739D 122
last_processingdate_168D 120
max_overdueamountmax2date_1142D 119
posfstqpd30lastmonth_3976962P 118
pmtnum_254L 118
mean_credlmt_935A 109
mean_numberofoverdueinstlmaxdat_641D 105
pmtssum_45A 104
mean_residualamount_856A 100
max_familystate_447L 98
mean_instlamount_768A 98
birthdate_574D 96
mean_totalamount_996A 88
max_pmtamount_36A 88
mean_pmts_dpd_1073P 87
max_nominalrate_281L 83
lastrejectdate_50D 82
last_refreshdate_3813885D 81
max_pmtnum_8L 78
last_employedfrom_700D 78
max_employedfrom_700D 78
max_numberofoverdueinstlmaxdat_148D 76
mean_overdueamountmax2_14A 76
responsedate_1012D 76
mean_pmts_overdue_1140A 75
mean_outstandingamount_362A 74
mean_credlmt_230A 74
last_annuity_853A 73
mean_overdueamountmax2date_1002D 71
mean_overdueamountmax2_398A 71
mean_pmts_dpd_303P 70
max_periodicityofpmts_837L 69
last_pmtamount_36A 69
mean_overdueamountmax_35A 68
lastrejectcredamount_222A 68
max_empl_industry_691L 67
mean_monthlyinstlamount_332A 65
max_totalamount_6A 64
mean_dateofcredend_289D 64
mean_totaloutstanddebtvalue_39A 64
last_credamount_590A 64
max_dpdmaxdatemonth_89T 63
max_incometype_1044T 61
mean_dateofrealrepmt_138D 61
max_processingdate_168D 60
last_pmtnum_8L 60
mean_totalamount_6A 59
max_overdueamountmaxdatemonth_365T 58
lastapplicationdate_877D 58
max_periodicityofpmts_1102L 57
max_byoccupationinc_3656910L 57
max_num_group1_15 56
mean_annuity_853A 56
dateofbirth_337D 56
max_nominalrate_498L 55
max_numberofoverdueinstlmax_1039L 51

ML METHODOLOGY: CATBOOST

CatBoost is an open-source gradient boosting library that efficiently handles large datasets with categorical features using ordered target encoding and missing values using symmetric weighted quantile sketches, reducing overfitting and improving performance.

KEY FEATURES:

Automated Data Handling

- Efficient handling of categorical features without encoding
- Built-in missing value handling without imputation
- Automatic internal feature scaling

High Performance Out-of-the-Box

- Excellent results with default parameters
- Minimal need for extensive parameter tuning
- Built-in cross-validation for hyperparameter selection

Overfitting Prevention Techniques

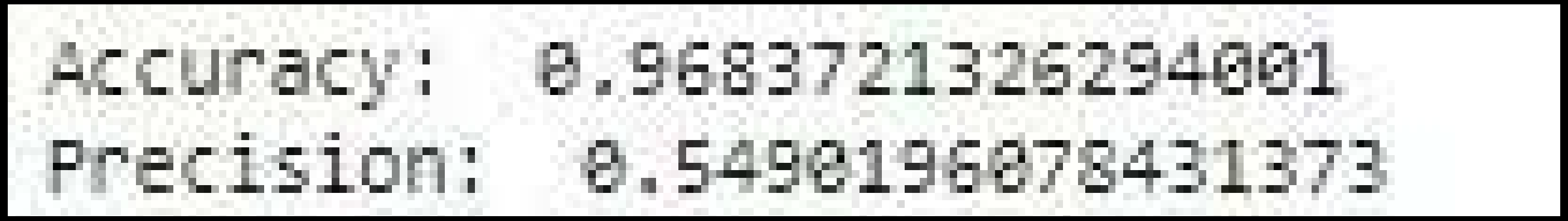
- Robust boosting, ordered boosting, random permutations
- Techniques to improve generalization on unseen data

WHY CATBOOST?

- **Automated Categorical Feature Handling -** Handles categorical data natively without encoding required.
- **Robust Missing Value Handling -** Uses symmetric weighted quantile sketches (SWQS) to automatically handle missing values.
- **Overfitting Prevention -** Employs ordered boosting, random permutations to reduce overfitting and improve generalization.
- **High Default Performance -** Provides excellent results with little parameter tuning needed.
- **Scalable GPU Training -** GPU-accelerated version enables faster training on large datasets and multi-GPU scalability.

LIGHTGBM

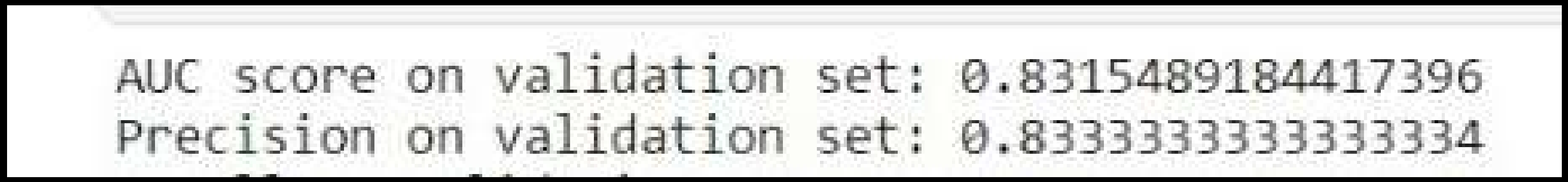
LightGBM Accuracy is 0.96 and Precision is 0.54. But the recall was less than 10%.



```
Accuracy:  0.9683721326294001
Precision:  0.5490196078431373
```

CATBOOST

CatBoost AUC score is 0.83 and Precision is also 0.83. But the recall was less than 10%.



```
AUC score on validation set: 0.8315489184417396
Precision on validation set: 0.8333333333333334
```

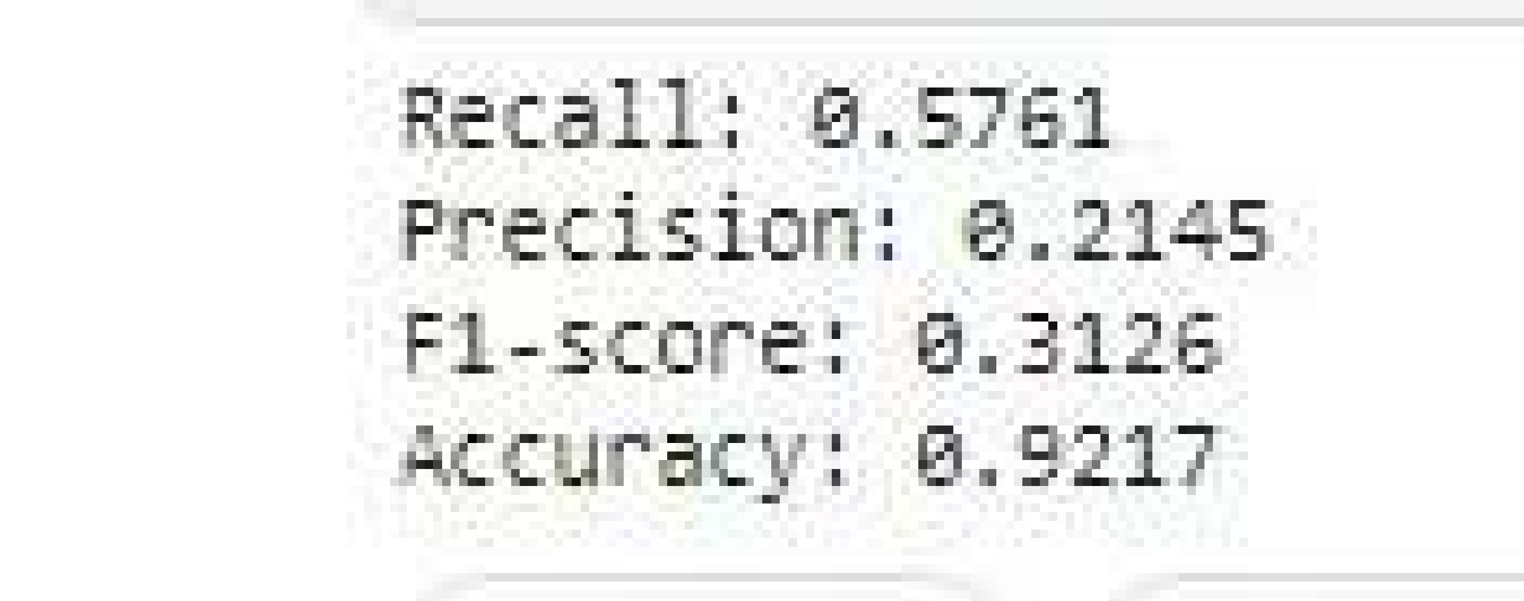
VOTING CLASSIFIER

- Trains on an ensemble of models
- Predicts output class based on majority voting
- Aggregates findings of each model

Why Use a Voting Classifier?

- Avoids creating separate dedicated models
- Predicts based on combined majority voting
- Can improve overall accuracy and robustness

Final Outcomes:



```
Recall: 0.5761
Precision: 0.2145
F1-score: 0.3126
Accuracy: 0.9217
```

How does our voting classifier work?

CHALLENGES

- Dealing with the depth 1 and depth 2 datasets was problematic as the data needed to be aggregated.
- Personal laptops faced computational issues due to limitations.
- The large dataset size led to extensive runtimes.
- Initially, there was a lot of uncertainty about how to approach and proceed with the problem.

PERFORMANCE METRIC

Submissions are evaluated using a **gini stability metric**. A gini score is calculated for predictions corresponding to each **WEEK_NUM.**

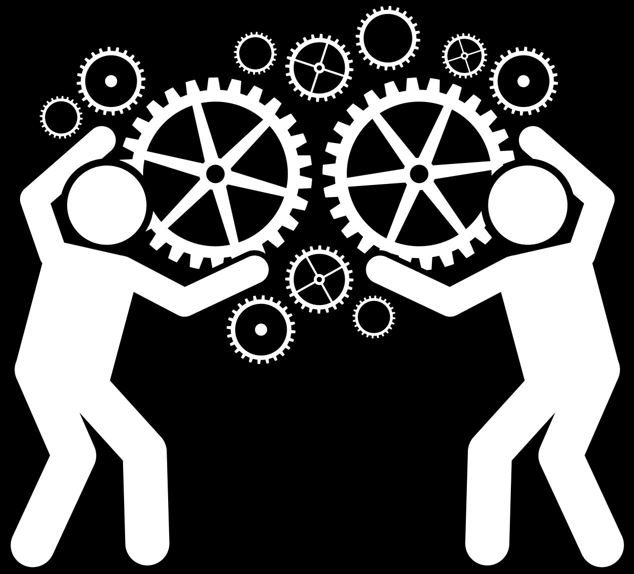
$$\text{gini} = 2 * \text{AUC} - 1$$

A linear regression, $a \cdot x+b$, is fit through the weekly gini scores, and a falling_rate is calculated as min(0,a).

This is used to penalize models that drop off in predictive ability.
Finally, the variability of the predictions are calculated by taking the standard deviation of the residuals from the above linear regression, applying a penalty to model variablity.
The final metric is calculated as

$$\text{stability metric} = mean(gini) + 88.0 \cdot min(0, a) - 0.5 \cdot std(\text{residuals})$$

MODEL DEPLOYABILITY



1. Admission Office: Plaksha’s admission office can utilize the credit risk model to assess whether incoming students require financial aid or not. This information can aid the admission office in making more informed decisions regarding the allocation of financial aid resources to students who genuinely require assistance.

2. Student-run Funds: If the university has any student-run funds or loan programs, the credit risk model can be employed to evaluate the creditworthiness of students applying for such funds. These student-run initiatives often have limited resources and need to carefully assess the ability of borrowers to repay the loans or funds provided. By integrating the credit risk model into their decision-making process, these student-run funds can assess the potential risk of default and make more informed lending decisions. This can help them mitigate financial losses and ensure the sustainability of their programs.

REFERENCES:

1. https://iopscience.iop.org/article/10.1088/1742-6596/1651/1/012111/pdf
2. https://arxiv.org/pdf/2110.02206.pdf
3. https://www.geeksforgeeks.org/lightgbm-light-gradient-boosting-machine/
4. https://www.analyticsvidhya.com/blog/2021/08/complete-guide-on-how-to-use-lightgbm-in-python/
5. https://www.analyticsvidhya.com/blog/2023/07/catboost-building-model-with-categorical-data/
6. https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/iet-its.2020.0396
7. https://www.mdpi.com/2076-3417/10/9/3227

Thank
You