AML Project

Team 4: William Gu, Angad Nandwani, Jiaxi Zhou, Yue Wang, Yuxin Qian

Objective

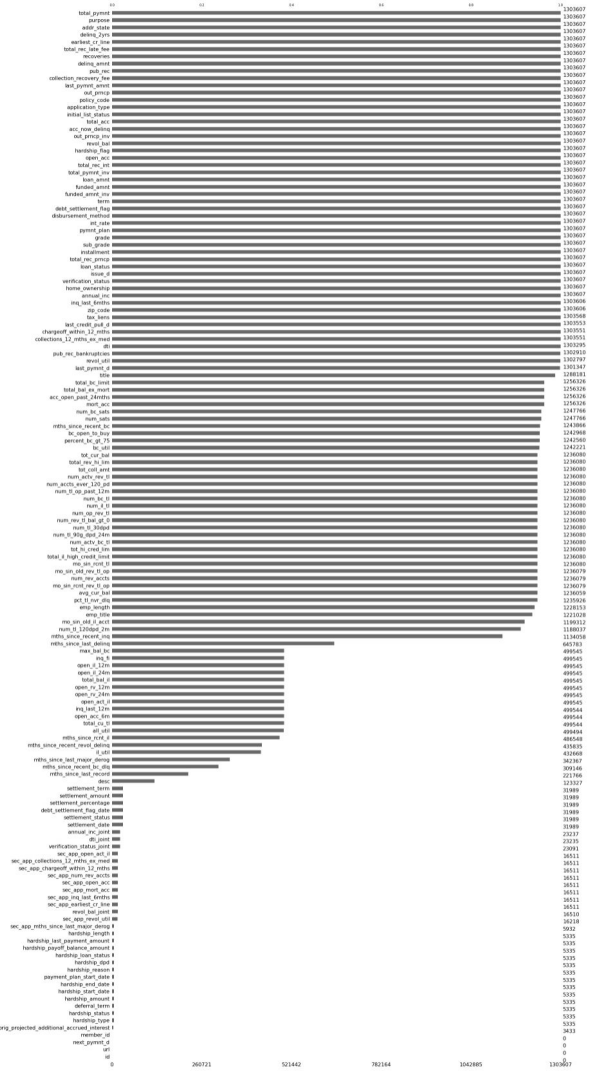
Predict loan status given applicant data: using available information, is this applicant likely to fully repay the loan?

Lending Club Dataset

- Raw dataset contains 2.2m rows of 144 features (109 numerical, 36 categorical)
- Label: “loan status”
  - Only terminal states are useful in making predictions
    - What use is there in predicting “current” if every loan starts off as current and in good standing?
  - Positive class: “charged off” - Loan for which there is no longer a reasonable expectation of further payments
  - Negative class: “fully paid” - Loan has been fully repaid

Overview of Missingness in Data

- The graph on the left shows the amount of missing values in each variable
- Majority of the variables are missing values in large proportion and their missingness behavior is correlated
- To deal with the missingness, we would set a threshold to only maintain variables that have less than 10% missing values, and start the preprocessing from there
- Any remaining missing numerical values can be handled using SimpleImputer & categorical missing values can be assigned a placeholder value



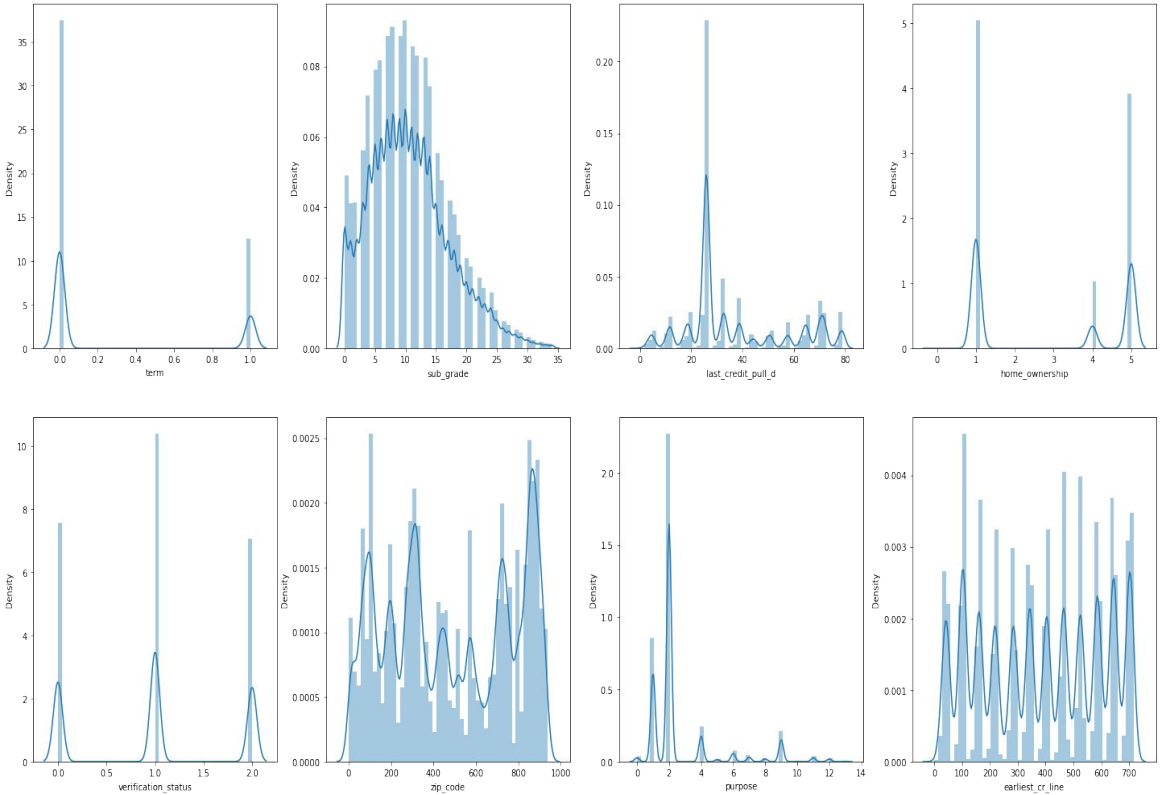
Feature Selection - Numerical Variables

- Based on the heatmap, we observed that some variables are highly correlated.
- Given the large number of variables, this would allow us to remove some variables that offer the same information.
- Loan_Status_Coded is the new feature created, where we coded fully paid as 1 and charged off as 0. We also observe that some variables are not correlated with the target, which suggests we can further trim down the number of variables.
- We will focus on top 30 variables most correlated with the target

Correlation Heatmap Between Subset of Variables

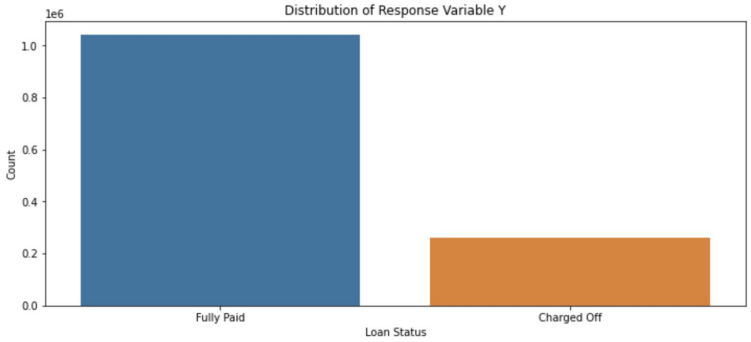
Feature Selection - Categorical Variables

- After dropping features which had large proportion of missing values, we had 22 categorical variables left.
- We performed the chi_square test to select the top 8 features
- The graph on the right shows the density curve for those 8 best features



Distribution of Target Variable

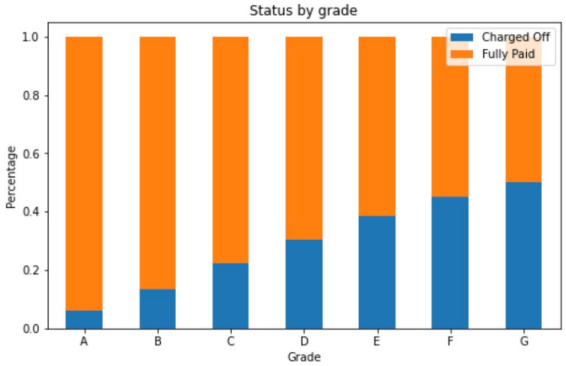
- The target variable, Loan Status, includes 2 relevant categories
  - All but “Charged Off” (positive class) and “Fully Paid” (negative class) represent intermediate states
- The cost of having an applicant default on their loan is high, so we want to avoid false negatives and emphasize recall

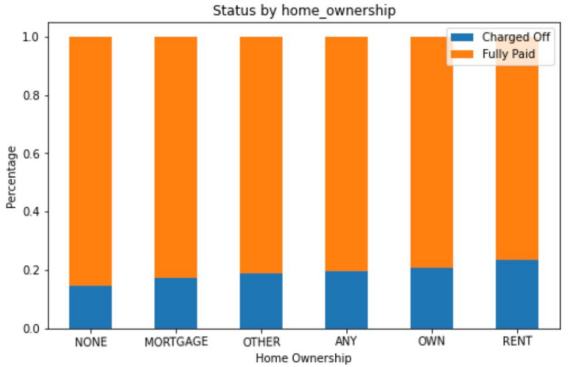


- The data is imbalanced, with roughly 80% of loans being fully paid

Points of interest:

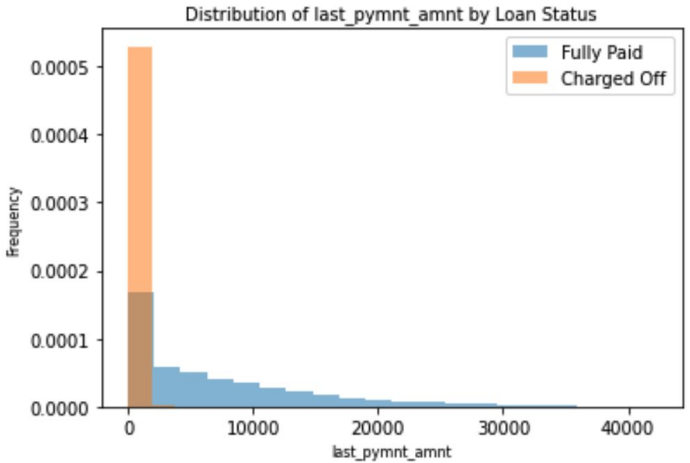
- The proportion of “Charged Off” loans increases linearly with lower Grades
- The distribution of loan status does not vary significantly with home ownership.





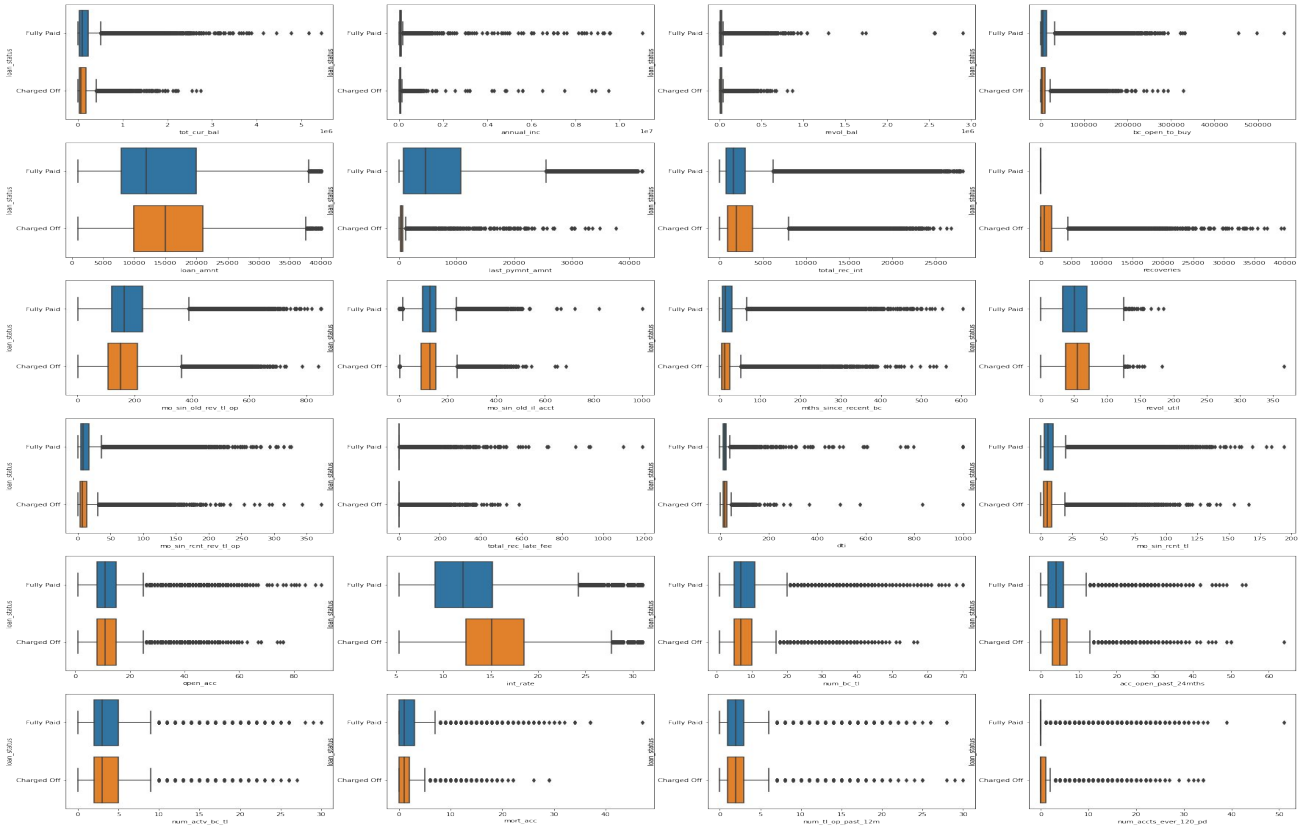
Label Leakage

- Some features may leak information about the label
- In practice, we will not have access to information recorded after a loan has reached a terminal state
    - E.g last payment amount before each loan is deemed “fully paid” or “charged off”
- These features must be manually identified and removed

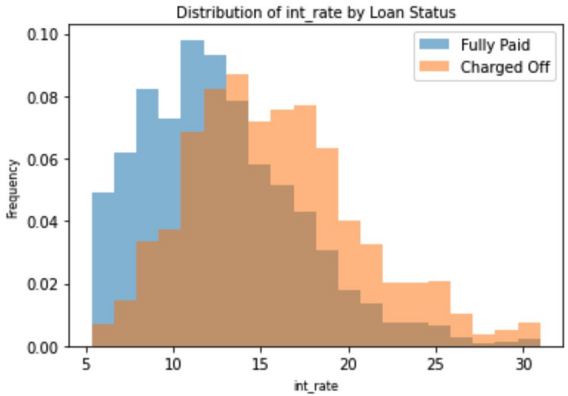


Relationship between Target and Numerical Features

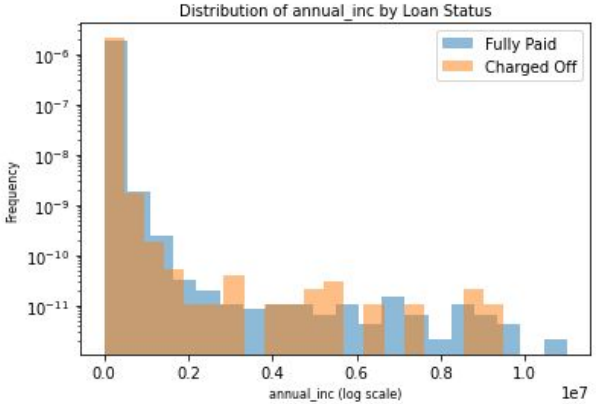
- There are many outliers for each of the numerical features.
- For non-tree model types, we will want to scale the data and remove outliers.



Relationship between Target and Numerical Features



- Loans which are eventually “Charged Off” are likely to have a higher interest rate
- The distribution of interest rate is a bit right-skewed



- Even with annual income on a log scale, the distribution skews heavily to the right
- Regardless of final loan status, applicants’ annual incomes appear to be drawn from similar distributions

Cleaning and Sampling

- Based on the previous EDA results, we did some cleaning steps and got the finalized dataset:
  - Dealing with missing value: select features with missing values < 10%
  - Feature selection:
    - Categorical features: select top 8 with chi-square test
    - Numerical features: selected the top 30 that are most correlated with the target value; removed the features with variance <= 1
  - Rescaling & Outlier removal: varies depending on model type (tree-based models are robust to scale & outliers)
  - Target, Ordinal-encoding, Normalizing: depends on the models (no need to encode for tree-based models)
- The graph on the left shows the schema of our preprocessed dataset, with 'loan_status' as the target variable, 8 categorical variables, and 24 numeric variables. There are totally 1,056,242 data records with no missing values.

Finanalized dataset

```
Data columns (total 33 columns):
 #   Column                 Non-Null Count    Dtype
---  ------                 --------------    -----
 0   tot_cur_bal            1056242 non-null  float64
 1   annual_inc             1056242 non-null  float64
 2   revol_bal              1056242 non-null  int64
 3   bc_open_to_buy         1056242 non-null  float64
 4   loan_amnt              1056242 non-null  int64
 5   last_pymnt_amnt        1056242 non-null  float64
 6   total_rec_int          1056242 non-null  float64
 7   recoveries             1056242 non-null  float64
 8   mo_sin_old_rev_tl_op   1056242 non-null  float64
 9   mo_sin_old_il_acct     1056242 non-null  float64
 10  mths_since_recent_bc   1056242 non-null  float64
 11  revol_util             1056242 non-null  float64
 12  mo_sin_rcnt_rev_tl_op  1056242 non-null  float64
 13  total_rec_late_fee     1056242 non-null  float64
 14  dti                    1056242 non-null  float64
 15  mo_sin_rcnt_tl         1056242 non-null  float64
 16  open_acc               1056242 non-null  float64
 17  int_rate               1056242 non-null  float64
 18  num_bc_tl              1056242 non-null  float64
 19  acc_open_past_24mths   1056242 non-null  float64
 20  num_actv_bc_tl         1056242 non-null  float64
 21  mort_acc               1056242 non-null  float64
 22  num_tl_op_past_12m     1056242 non-null  float64
 23  num_accts_ever_120_pd  1056242 non-null  float64
 24  term                   1056242 non-null  object
 25  sub_grade              1056242 non-null  object
 26  last_credit_pull_d     1056242 non-null  object
 27  home_ownership         1056242 non-null  object
 28  verification_status    1056242 non-null  object
 29  zip_code               1056242 non-null  object
 30  purpose                1056242 non-null  object
 31  earliest_cr_line       1056242 non-null  object
 32  loan_status            1056242 non-null  object
```

Cleaning and Sampling

- Data Sampling:
  - After cleaning, we are left with roughly 1m rows and 32 features
  - Split data to 80:20 for development and testing
  - Split development data to 80:20 training and validation
  - Use stratified splitting

```
from sklearn.model_selection import train_test_split

X = final_set.drop(['loan_status'], axis = 1)
y = final_set['loan_status'].map({'Fully Paid': 1, 'Charged Off': 0})

X_dev, X_test, y_dev, y_test = train_test_split(X, y, stratify = y, test_size = 0.2, random_state = 42)
X_train, X_val, y_train, y_val = train_test_split(X_dev, y_dev, stratify = y_dev, test_size = 0.2, random_state = 84)
```

```
print(X_dev.shape)
print(X_train.shape)
print(X_val.shape)
print(X_test.shape)
```

```
(844993, 32)
(675994, 32)
(168999, 32)
(211249, 32)
```

Machine Learning Techniques

1. **Preprocessing**:
    a. Dropping Columns with large proportion of missing data, imputing values where necessary
    b. Feature Selection with Correlation & Variance - Numerical data
    c. Feature Selection with Chi-Square Test - Categorical data
    d. Outlier Detection & scaling applied to numerical data for non-tree models
    e. Encoding: Ordinal where applicable; Target & Label depending on model type
2. **Modeling**:
    a. Logistic regression, SVM, and boosted trees
    b. Ensemble methods that utilize some combination of these models.
3. **Cross-Validation & Hyperparameter Tuning**:
    a. K-fold cross validation & Grid search method
    b. Incorporation of some regularization techniques such as Lasso and Ridge.
4. **Evaluation**:
    a. As a classification problem, we plan to use metrics below to evaluate the performance of our model. Specifically, we will focus more on recall since we care more about false negatives.
        i. Precision, recall, accuracy, F1 score
        ii. PR curve, ROC- AUC