Lending Club Loan: Exploratory Data Analysis, Classifications, Predictions

Yiru Fang, Qiyang Wang, Andrew Cao

STAT 447 Final Project

Dec 6, 2024

Agenda

- Background
- Data Exploring
- Data Engineering
- Analyzing Method & Result
- Conclusion

Background

- **Growing Online Loan Market**: Rapid rise in online lending platforms.
- **Our Curiosity**: How do these platforms maintain profitability despite risks?
- **Objective**: implement EDA to identify key factors for profitability and risk management.

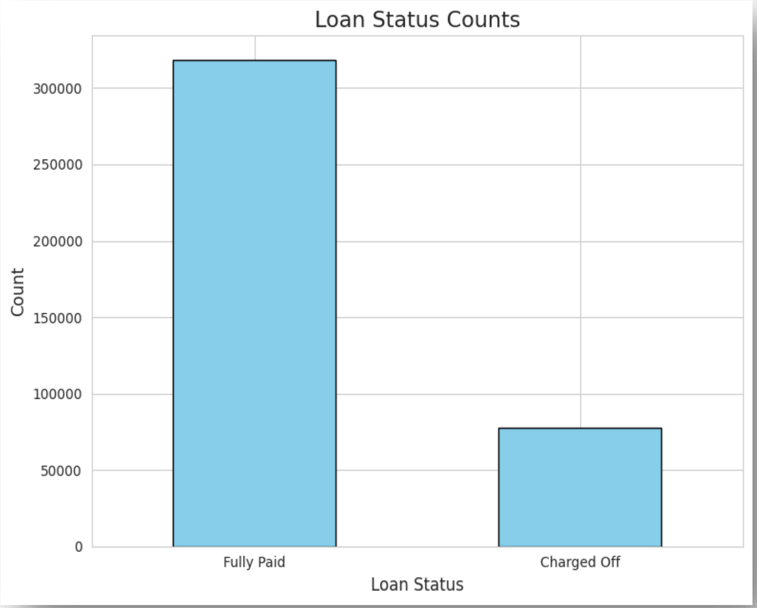
Data Exploring

- **Dataset**: Evaluation data from Lending Club Loan through Kaggle, a leading online lending company.
- **Exploratory Data Analysis (EDA) Objective**: Analyze relationships between variables and **loan status**.
- **Focus**: Loan status categories
  - "Fully Paid" (good credit).
  - "Charge Off" (bad credit).

```
Data columns (total 27 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   loan_amnt             396030 non-null  float64
 1   term                  396030 non-null  object
 2   int_rate              396030 non-null  float64
 3   installment           396030 non-null  float64
 4   grade                 396030 non-null  object
 5   sub_grade             396030 non-null  object
 6   emp_title             373103 non-null  object
 7   emp_length            377729 non-null  object
 8   home_ownership        396030 non-null  object
 9   annual_inc            396030 non-null  float64
 10  verification_status   396030 non-null  object
 11  issue_d               396030 non-null  object
 12  loan_status           396030 non-null  object
 13  purpose               396030 non-null  object
 14  title                 394274 non-null  object
 15  dti                   396030 non-null  float64
 16  earliest_cr_line      396030 non-null  object
 17  open_acc              396030 non-null  float64
 18  pub_rec               396030 non-null  float64
 19  revol_bal             396030 non-null  float64
...
 25  pub_rec_bankruptcies  395495 non-null  float64
 26  address               396030 non-null  object
dtypes: float64(12), object(15)
```

Data Exploring

- **Imbalance**: ~300,000 "Fully Paid" vs. ~50,000 "Charged Off."
- **Impact**: Models may favor "Fully Paid"; resampling is needed.
- **Focus**: Analyze "Charged Off" loans to identify high-risk factors.
- **Next Steps**: Explore key features



Data Exploring

Multicollinearity:

- `loan_amnt` ↔ `installment` (~1.0).

Action: Drop one or apply PCA.

Moderate Predictors:

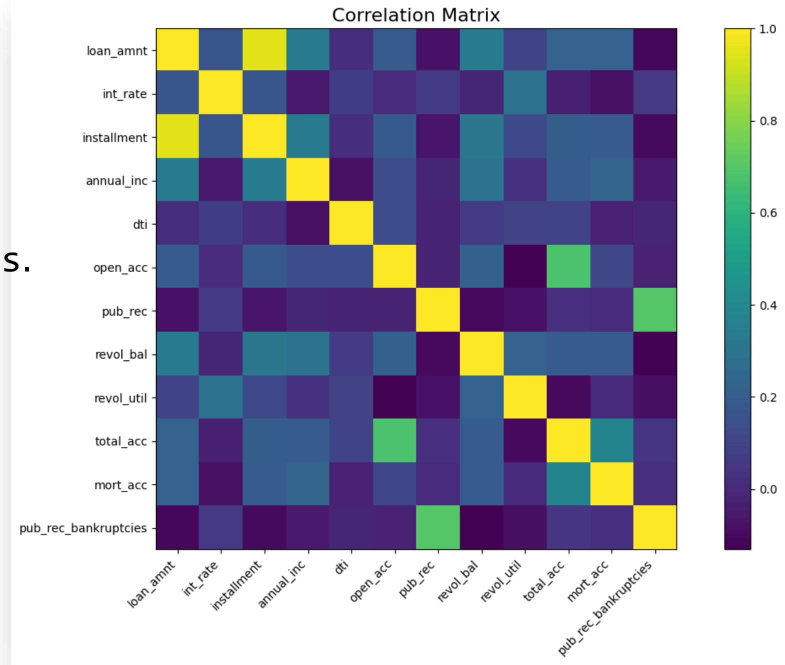
- `revol_bal` ↔ `revol_util` → Relevant for `loan_status`.

Action: Prioritize these features.

Weak Predictors:

- `pub_rec`, `total_acc`, `mort_accp`, `pub_rec_bankruptcies`.

Action: Evaluate for removal.



Note: **revol_bal** (total credit revolving balance), **revol_util** (amount of credit the borrower is using relative to all available revolving credit), **pub_rec** (number of derogatory public records), **total_acc** (total number of credit lines currently in the borrower's credit file), **mort_acc** (number of mortage accounts)

Data Engineering

- **Data Cleaning**
- Removed redundant columns and rows.
- Handled missing values.
- Converted categorical variables.
- **Data Preparation**:
- Split data (80% train, 20% test).
- Removed outliers (filtered extremes).
- Applied MinMaxScaler for normalization.

```
loan_amnt               float64
term                      int64
int_rate                float64
installment             float64
sub_grade                object
home_ownership           object
annual_inc              float64
verification_status      object
loan_status              object
purpose                  object
dti                     float64
open_acc                float64
pub_rec                 float64
revol_bal               float64
revol_util              float64
total_acc               float64
initial_list_status      object
application_type         object
mort_acc                float64
pub_rec_bankruptcies    float64
zip_code                 object
dtype: object
```

Analyzing Method

- **Feature Selection**
  - Chi-Square Test
  - PCA
  - L1 Regularization

- **Prediction Model**
  - KNN
  - K-Means
  - Logistic Model
  - Random Forest

Method – Chi-square Test

- **Purpose:**
- Identify which categorical variables are significantly associated with **Loan Status** (Fully Paid vs. Charged Off).
- **Key Steps:**
  - Create a contingency table (e.g., Loan Status vs. Loan Purpose).
  - Compute: $x^2 = \sum \left(\frac{(O-E)^2}{E} \right)$
  - Where :
    - X^2 = chi-square statistic.
    - O = observed frequency.
    - E = expected frequency.
  - Evaluate significance ($p < 0.05$).

Result – Chi-square Test

- **Significant Features (p<0.05)** :
  sub_grade_encoded
- **home_ownership_encoded**:
  ownership status
- **Not Significant (p ≥ 0.05)**:
  initial_list_status_encoded
  application_type_encoded

	Feature	Chi2_Score	P-Value
14	sub_grade_encoded	98126.36	0.00
19	home_ownership_encoded	2527.71	0.00
15	verification_status_encoded	1399.58	0.00
16	purpose_encoded	438.55	0.00
17	initial_list_status_encoded	0.31	0.58
18	application_type_encoded	0.10	0.75

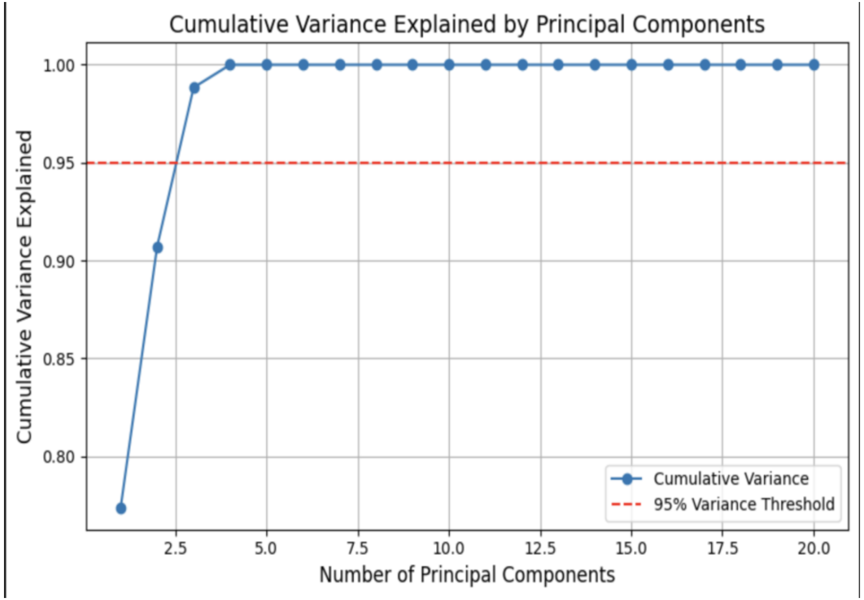
Method – PCA

Purpose:

Reduce the number of features while keeping the most important information about **Loan Status**.

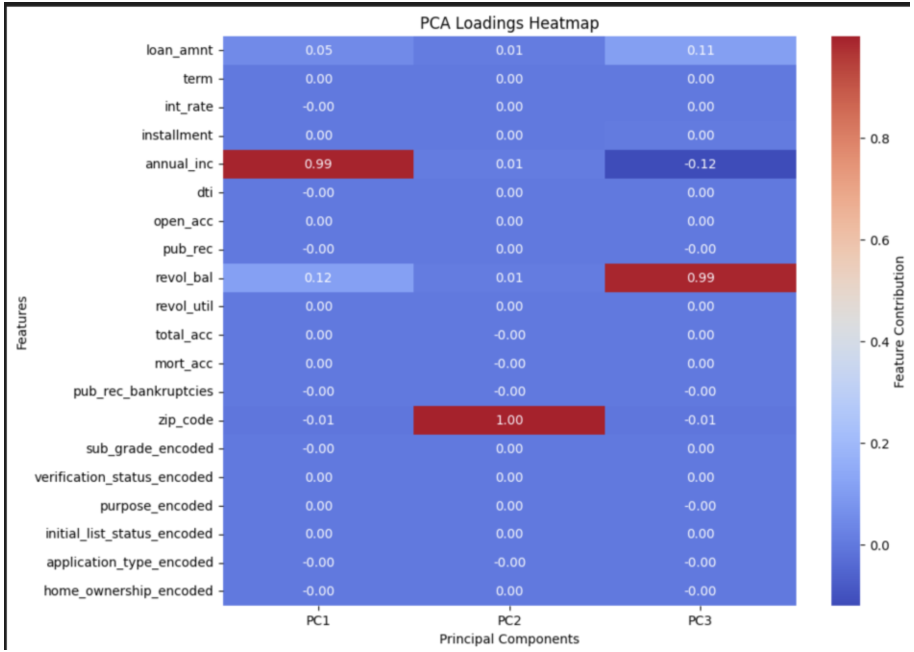
Key Steps:

Standardize Data: Scaled all features to have mean = 0 and variance = 1.
Identify Principal Components (PCs): Found directions of maximum variance.
Select Components: Retained PCs explaining 95% of total variance.



Result- PCA

- **PC1**: Dominated by annual_inc (99%).
- **PC2**: Strongly influenced by zip_code (100%).
- **PC3**: Driven by revol_bal (99%).



Method – Lasso regression

Purpose:

Adds a penalty term to the loss function to reduce model complexity:

$$Loss = \mathrm{MSE} + \lambda \sum |w_i|$$

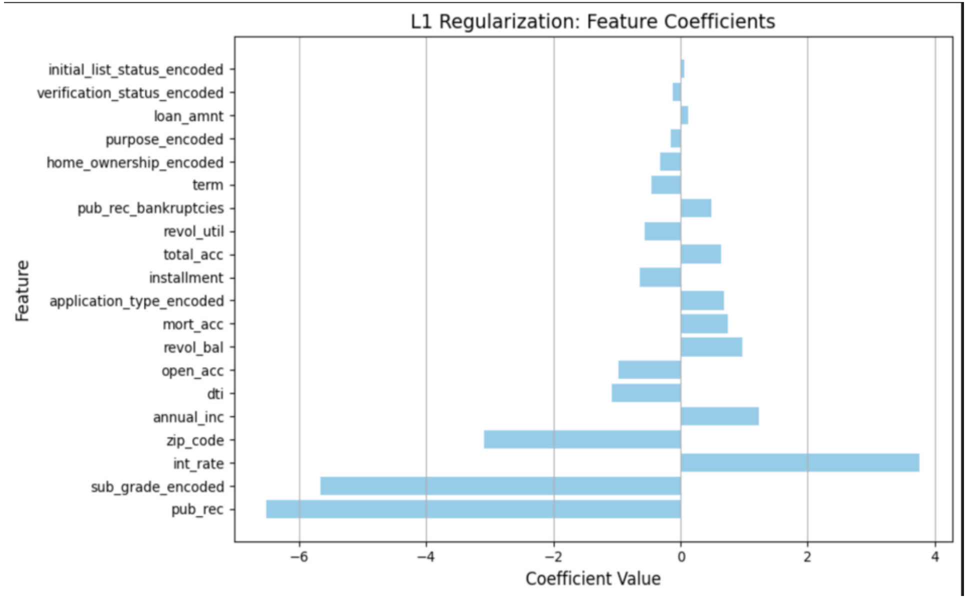
Key Steps:

1. Standardize the dataset to ensure equal scaling.
2. Apply Lasso regression with a tuning parameter (λ\lambdaλ) to control regularization strength.
3. Evaluate the model to identify selected features (non-zero coefficients).

Result – Lasso regression

sub_grade_encoded, **pub_rec**, and **zip_code** are critical predictors of the target variable

initial_list_status_encoded and **verification_status_encoded** have been reduced to zero.

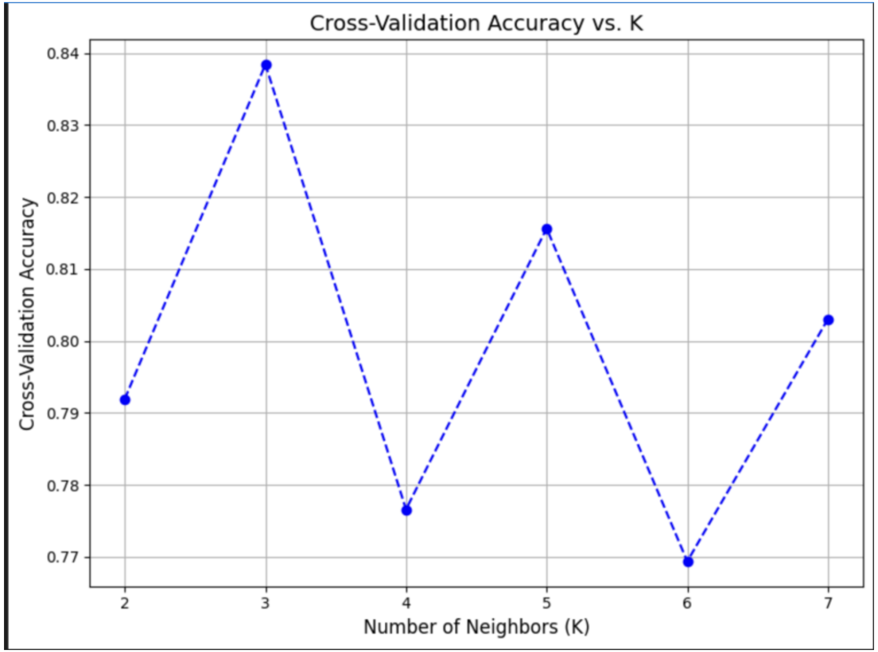


Method - KNN

- **Purpose**

Makes predictions based on similarity, useful for identifying patterns in loan repayment behavior.

- **Advantages**
    - Easy to implement and interpret
    - Non-parametric



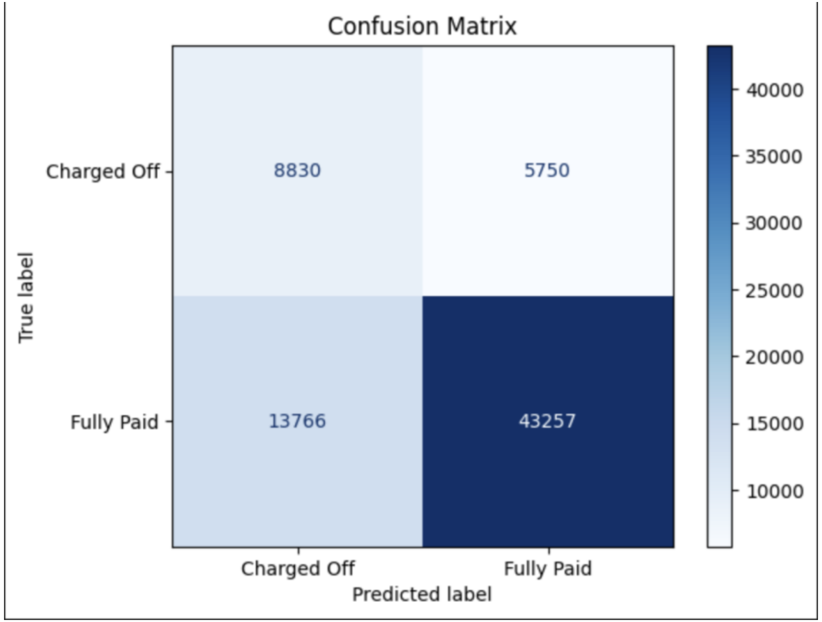
Result- KNN

Overall Performance:

- Achieved **82.6% accuracy** in predicting loan repayment status.
- Strong performance for "Fully Paid" loans (**F1-score: 0.90**) but weaker for "Charged Off" loans (**F1-score: 0.48**).

Strengths:

- High **recall (0.94)** for "Fully Paid" loans ensures most positive cases are correctly identified.
- Model effectively captures repayment trends.

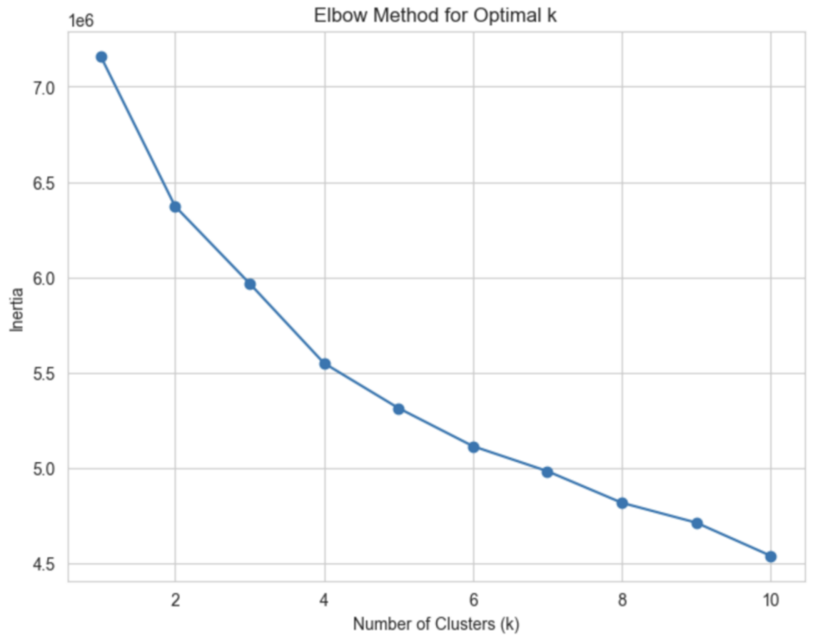


Method – K means

- **Purpose**

Identify natural groupings of loans based on borrower characteristics.

- **Advantage**
  - Easy computing
  - helps evaluate the separability of features



Result – K means

- High accuracy in identifying "Charged Off" cases
- Good precision (84%) ensures reliable predictions for "Charged Off."
- Recall (80%) indicates 20% of "Charged Off" cases were missed.

```
Confusion Matrix:
 [[ 5624  8956]
 [11220 45803]]
```

Method – Logistic Regression

- **Purpose**

Estimates the probability of a categorical dependent variable (e.g., Fully Paid vs. Charged Off).

- **Advantages**
  - Binary Classification
  - Provides Probabilities
  - Handles Linearly Separable Data

Result– Logistic Regression

- **83% accuracy**, performing well for "Fully Paid" loans (F1: 0.90) but struggling with "Charged Off" loans (F1: 0.44).

- Strong recall for "Fully Paid" (97%) highlights reliability, while low recall for "Charged Off" (31%) indicates room for improvement.

Classification Report:

	precision	recall	f1-score	support
Charged Off	0.71	0.31	0.44	14580
Fully Paid	0.85	0.97	0.90	57023
accuracy			0.83	71603
macro avg	0.78	0.64	0.67	71603
weighted avg	0.82	0.83	0.81	71603

Method – Random Forest

- **Purpose**

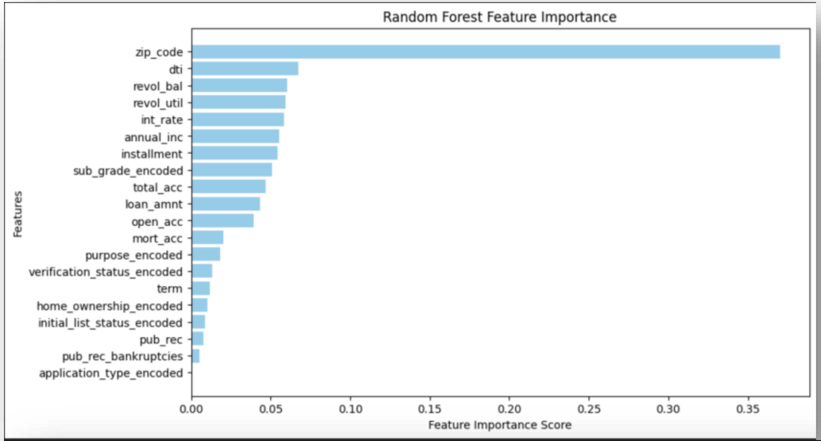
Combines predictions of individual trees to improve accuracy and reduce overfitting.

- **Step**
  - High Accuracy
  - Handles Complexity
  - Feature Importance

Result– Random Forest

- High accuracy (**88.6%)** and excellent performance for the majority class with an **F1-score(0.93)**

- Key predictors include **zip_code**, **dti**, and **revol_bal**, as shown in the feature importance chart.



Accuracy: 0.8859405332178819
Classification Report:

	precision	recall	f1-score	support
Charged Off	0.93	0.47	0.63	14580
Fully Paid	0.88	0.99	0.93	57023
accuracy			0.89	71603
macro avg	0.91	0.73	0.78	71603
weighted avg	0.89	0.89	0.87	71603

Conclusion – Feature Selection

	Chisqure Test	PCA	L1
Feature1	Sub_grade	annual_inc	Pub_rec
Feature2	Home_ownership	Zip_code	Sub_grade
Feature	Verification_status	revol_bal	-

Conclusion – Model Predicton

	KNN	K Means	Logistic	RF
F1 score	0.81	0.82	0.81	0.87
Accuracy	0.83	0.72	0.83	0.86

- **Most Optimal Model**: Random Forest performs the best and should be the first choice if computational resources are not a concern.
- **Future Step**:
  - Imbalanced data: weight
  - Advanced models like **XGBoost**, **LightGBM**

Q & A

Thank you!