Credit Risk of Landing Club Loans

With Classification Model

Presented By:
Muhammad Hudzaifah N

Overview

1. About Company
2. Business Understanding
3. Objective
4. Data Source
5. Tools
6. Data Understanding
7. Statistical Summary
8. Exploratory Data Analysis
9.

About Company

Company Profile

LendingClub is a financial services company headquartered in San Francisco, California. It was the first peer-to-peer lender to register its offerings as securities with the Securities and Exchange Commission (SEC) and to offer loan trading on a secondary market. At its height, LendingClub was the world's largest peer-to-peer lending platform.

Business Understanding

Step nº 01
the company receives a loan application, it has to make a decision for loan approval based on the applicant's profile

Step nº 02
The applicant is unlikely to repay the loan, is likely to default, then approving the loan may cause financial loss to the company

Step nº 03
The applicant is likely to repay the loan, then not approving the loan will result in the loss of the company's business.

Business Objectives

01

Understand business problems and seek insights from data provided by LandingClub

02

Develop a predictive model capable of predicting loan approval of applicants to minimize the risk of default.

03

Determine the important features that contribute to the approval of the applicant's loan.

Data Source

Credit Risk of Landing Club Loans Dataset

Source: Rakamin

Tools

Visual Studio Code

Data Understanding

	Unnamed: 0	id	member_id	loan_amnt	funded_amnt	funded_amnt_inv	term	int_rate	installment	grade	...	total_bal_il	il_util	open_rv_12m	open_rv_24m	max_bal_bc	all_util	total_rev_hi_lim
0	0	1077501	1296599	5000	5000	4975.0	36 months	10.65	162.87	B	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	1	1077430	1314167	2500	2500	2500.0	60 months	15.27	59.83	C	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	2	1077175	1313524	2400	2400	2400.0	36 months	15.96	84.33	C	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	3	1076863	1277178	10000	10000	10000.0	36 months	13.49	339.31	C	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	4	1075358	1311748	3000	3000	3000.0	60 months	12.69	67.79	B	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
464965	466280	8598660	1440975	18400	18400	18400.0	60 months	14.47	432.64	C	...	NaN	NaN	NaN	NaN	NaN	NaN	29900.0
464966	466281	9684700	11536848	22000	22000	22000.0	60 months	19.97	582.50	D	...	NaN	NaN	NaN	NaN	NaN	NaN	39400.0
464967	466282	9584776	11436914	20700	20700	20700.0	60 months	16.99	514.34	D	...	NaN	NaN	NaN	NaN	NaN	NaN	13100.0
464968	466283	9604874	11457002	2000	2000	2000.0	36 months	7.90	62.59	A	...	NaN	NaN	NaN	NaN	NaN	NaN	53100.0
464969	466284	9199665	11061576	10000	10000	9975.0	36 months	19.20	367.58	D	...	NaN	NaN	NaN	NaN	NaN	NaN	16000.0

464970 rows × 75 columns

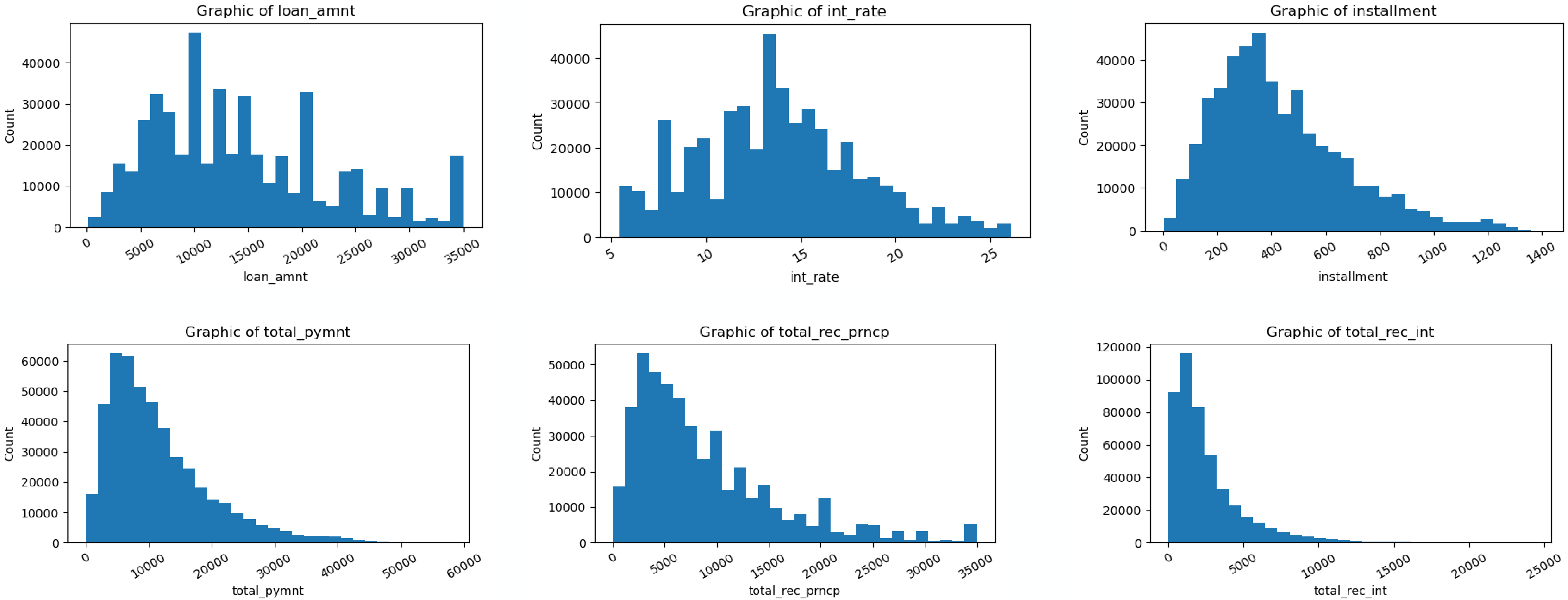
StaticTical Summary

	count	mean	std	min	25%	50%	75%	max
Unnamed: 0	464970.0	2.336444e+05	1.344628e+05	0.00	117557.25	2.337995e+05	3.500418e+05	4.662840e+05
id	464970.0	1.309108e+07	1.090698e+07	54734.00	3636524.50	1.011694e+07	2.074230e+07	3.809811e+07
member_id	464970.0	1.460918e+07	1.169656e+07	70473.00	4371763.25	1.194994e+07	2.301219e+07	4.086083e+07
loan_amnt	464970.0	1.431553e+04	8.286103e+03	130.00	8000.00	1.200000e+04	2.000000e+04	3.500000e+04
int_rate	464969.0	1.382795e+01	4.356569e+00	5.42	10.99	1.366000e+01	1.649000e+01	2.606000e+01
installment	464970.0	4.319761e+02	2.434333e+02	1.00	256.64	3.798100e+02	5.664400e+02	1.409990e+03
annual_inc	464966.0	7.327214e+04	5.497333e+04	0.00	45000.00	6.300000e+04	8.890000e+04	7.500000e+06
dti	464969.0	1.721926e+01	7.851894e+00	0.00	11.36	1.687000e+01	2.278000e+01	3.999000e+01
delinq_2yrs	464940.0	2.847894e-01	7.977655e-01	0.00	0.00	0.000000e+00	0.000000e+00	2.900000e+01
inq_last_6mths	464940.0	8.046608e-01	1.091690e+00	0.00	0.00	0.000000e+00	1.000000e+00	3.300000e+01
mths_since_last_delinq	215360.0	3.410543e+01	2.178012e+01	0.00	16.00	3.100000e+01	4.900000e+01	1.880000e+02
mths_since_last_record	62463.0	7.427871e+01	3.035736e+01	0.00	53.00	7.600000e+01	1.020000e+02	1.290000e+02
open_acc	464940.0	1.118646e+01	4.987993e+00	0.00	8.00	1.000000e+01	1.400000e+01	8.400000e+01
pub_rec	464940.0	1.605971e-01	5.111162e-01	0.00	0.00	0.000000e+00	0.000000e+00	6.300000e+01
revol_bal	464969.0	1.622941e+04	2.067144e+04	0.00	6411.00	1.176300e+04	2.033300e+04	2.568995e+06
revol_util	464629.0	5.617779e+01	2.373545e+01	0.00	39.20	5.760000e+01	7.470000e+01	8.923000e+02
total_acc	464940.0	2.506243e+01	1.160047e+01	1.00	17.00	2.300000e+01	3.200000e+01	1.560000e+02
out_prncp	464969.0	4.411796e+03	6.357597e+03	0.00	0.00	4.363700e+02	7.350920e+03	3.216038e+04
out_prncp_inv	464969.0	4.410185e+03	6.355716e+03	0.00	0.00	4.359700e+02	7.344860e+03	3.216038e+04
total_pymnt	464969.0	1.153614e+04	8.264623e+03	0.00	5549.40	9.415078e+03	1.530012e+04	5.777758e+04
total_pymnt_inv	464969.0	1.146515e+04	8.253087e+03	0.00	5497.46	9.349660e+03	1.522296e+04	5.777758e+04
total_rec_prncp	464969.0	8.862224e+03	7.030526e+03	0.00	3705.63	6.814530e+03	1.200000e+04	3.500003e+04
total_rec_int	464969.0	2.587889e+03	2.483440e+03	0.00	956.98	1.818090e+03	3.302840e+03	2.420562e+04
total_rec_late_fee	464969.0	6.510358e-01	5.270141e+00	0.00	0.00	0.000000e+00	0.000000e+00	3.586800e+02
recoveries	464969.0	8.537071e+01	5.522434e+02	0.00	0.00	0.000000e+00	0.000000e+00	3.352027e+04
collection_recovery_fee	464969.0	8.958576e+00	8.548359e+01	0.00	0.00	0.000000e+00	0.000000e+00	7.002190e+03
last_pymnt_amnt	464969.0	3.121482e+03	5.552849e+03	0.00	312.58	5.459600e+02	3.180790e+03	3.623444e+04
collections_12_mths_ex_med	464824.0	9.095916e-03	1.087132e-01	0.00	0.00	0.000000e+00	0.000000e+00	2.000000e+01
mths_since_last_major_derog	98700.0	4.285375e+01	2.166878e+01	0.00	26.00	4.200000e+01	5.900000e+01	1.880000e+02
policy_code	464969.0	1.000000e+00	0.000000e+00	1.00	1.00	1.000000e+00	1.000000e+00	1.000000e+00
annual_inc_joint	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
dti_joint	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
verification_status_joint	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
acc_now_delinq	464940.0	4.011270e-03	6.871802e-02	0.00	0.00	0.000000e+00	0.000000e+00	5.000000e+00
tot_coll_amt	394693.0	1.922999e+02	1.465453e+04	0.00	0.00	0.000000e+00	0.000000e+00	9.152545e+06
tot_cur_bal	394693.0	1.387972e+05	1.521027e+05	0.00	28614.00	8.150900e+04	2.089410e+05	8.000078e+06
open_acc_6m	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
open_il_6m	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
open_il_12m	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
open_il_24m	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
mths_since_rcnt_il	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
total_bal_il	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
il_util	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
open_rv_12m	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
open_rv_24m	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
max_bal_bc	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
all_util	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
total_rev_hi_lim	394693.0	3.037628e+04	3.726540e+04	0.00	13500.00	2.280000e+04	3.790000e+04	9.999999e+06
inq_fi	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
total_cu_tl	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
inq_last_12m	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Exploratory Data Analysis

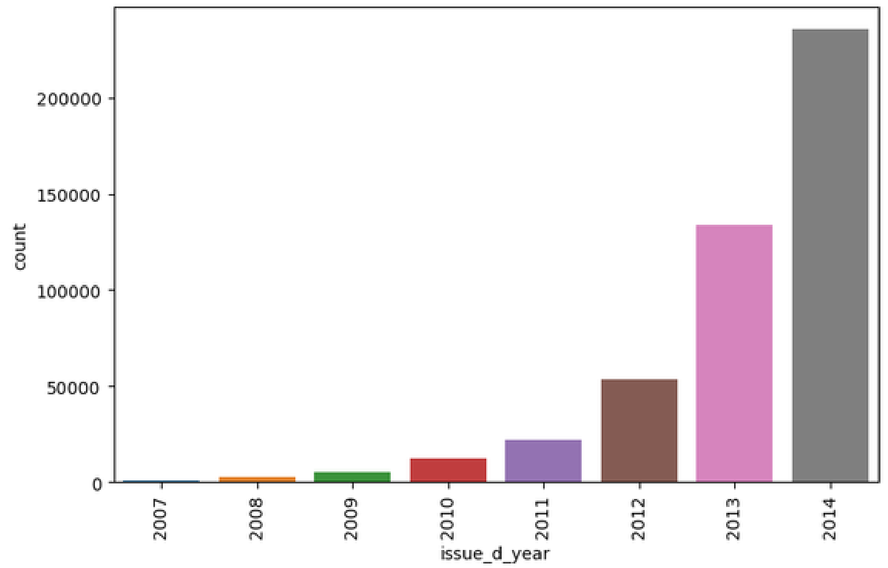
Univariate Analysis: Numerical Features

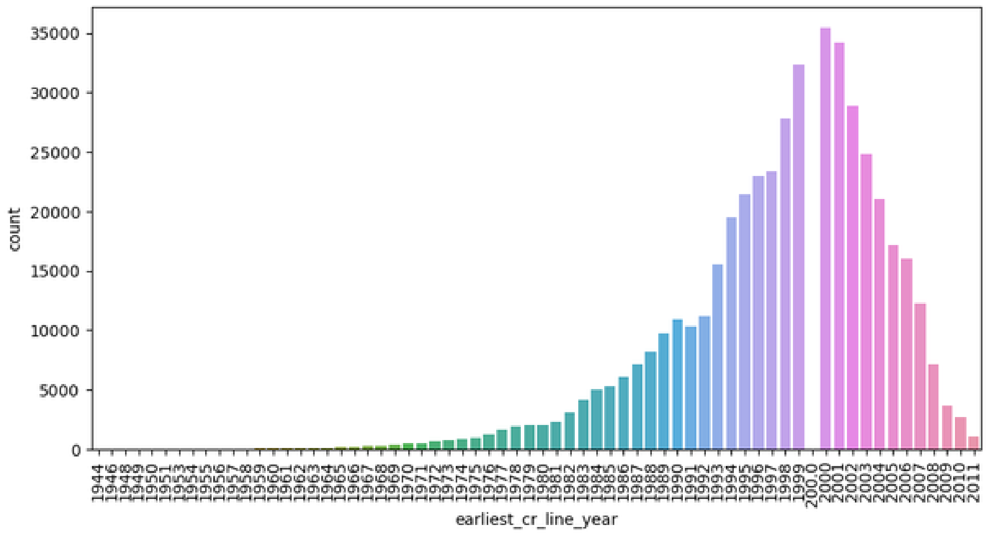
Contoh numerical distribution, dimana seitap data skewness

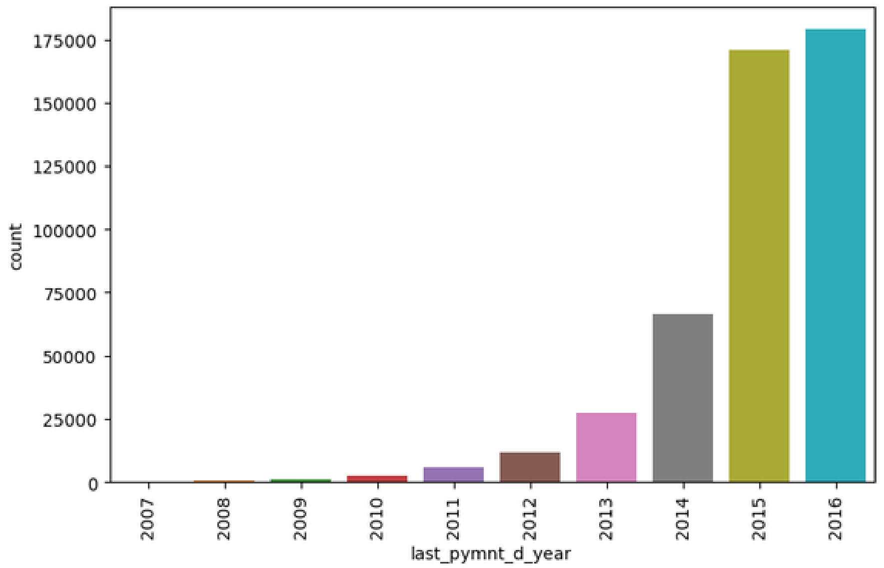


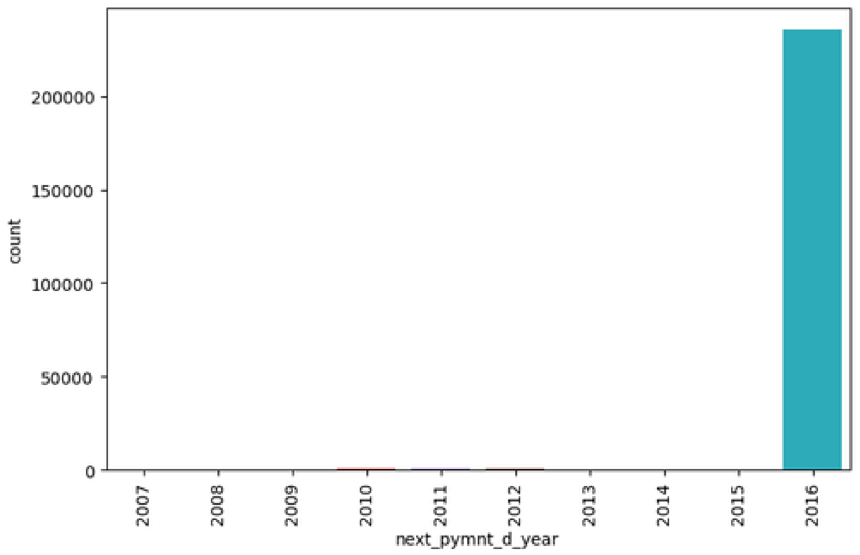
Exploratory Data Analysis

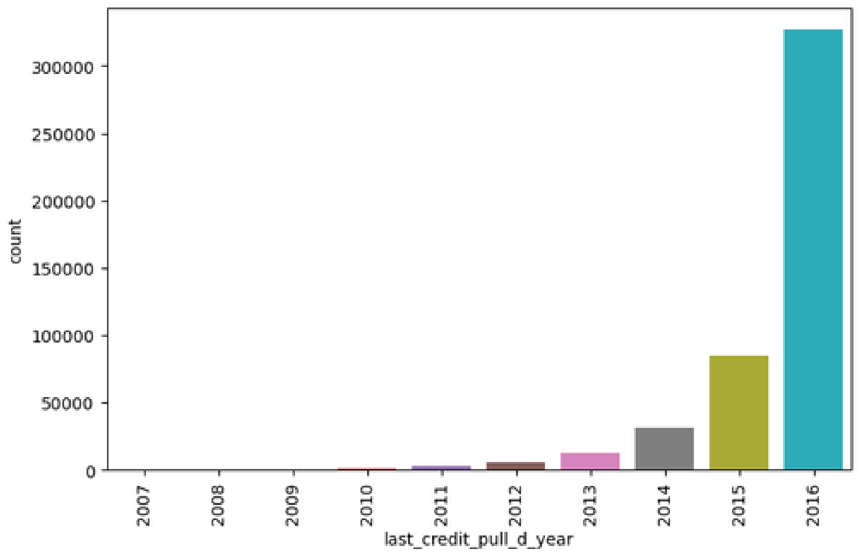
Univariate Analysis: Categorical Dates Features





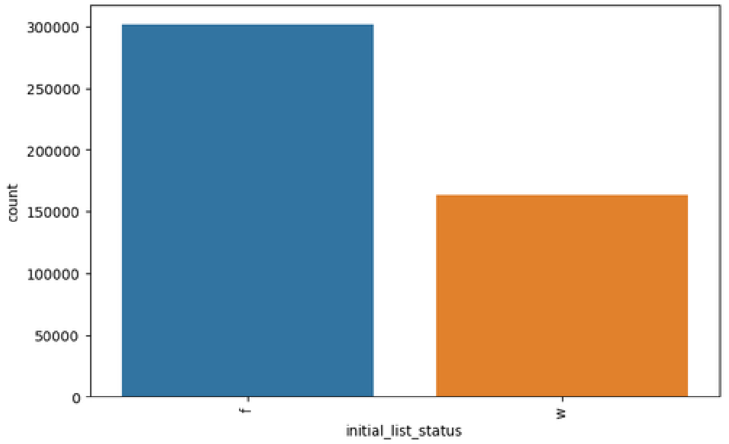


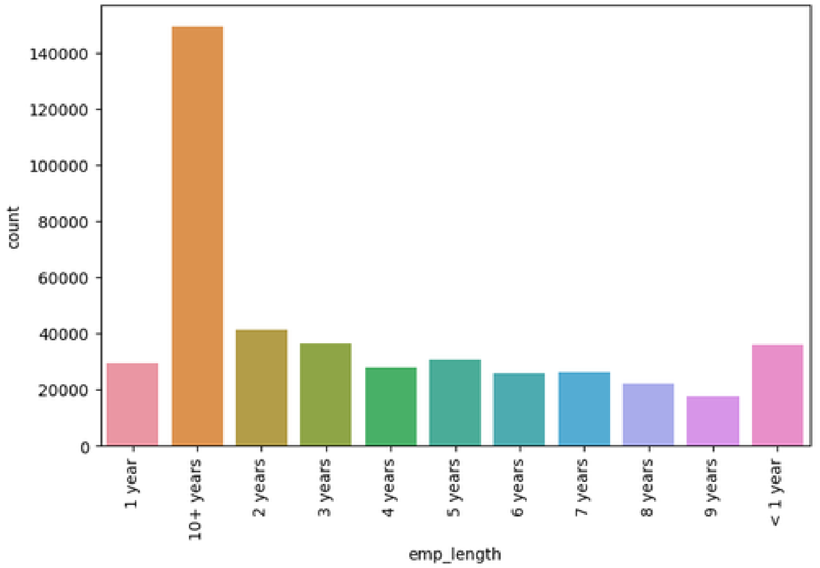


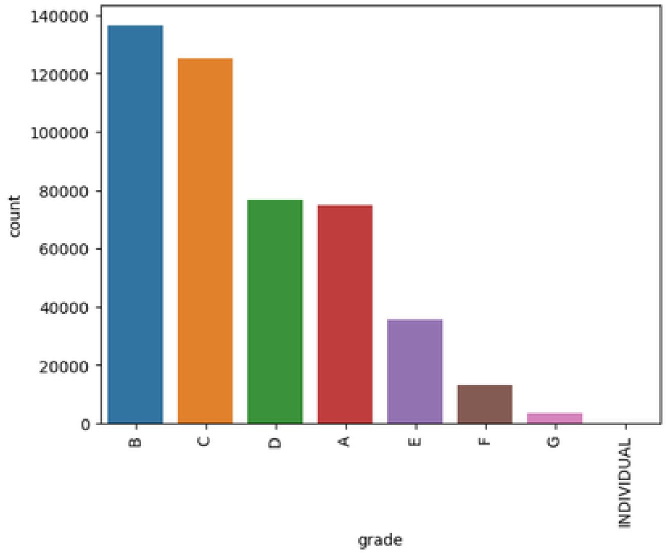


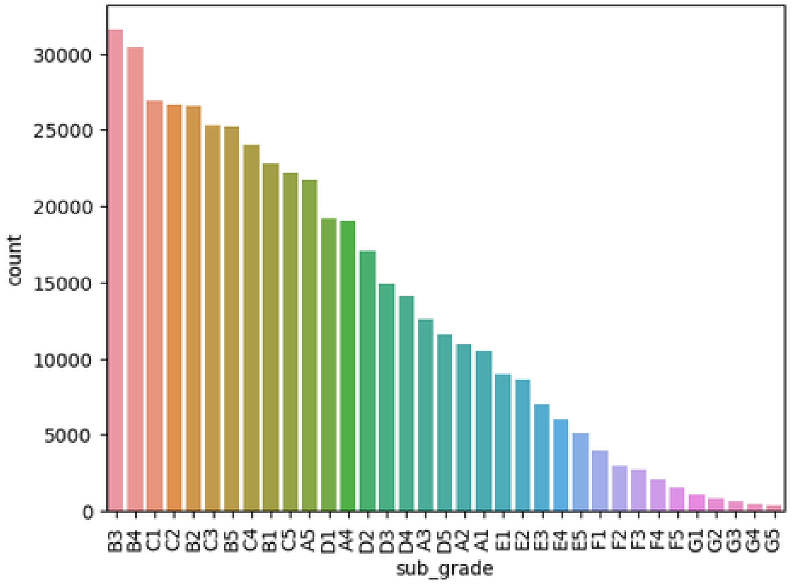
Exploratory Data Analysis

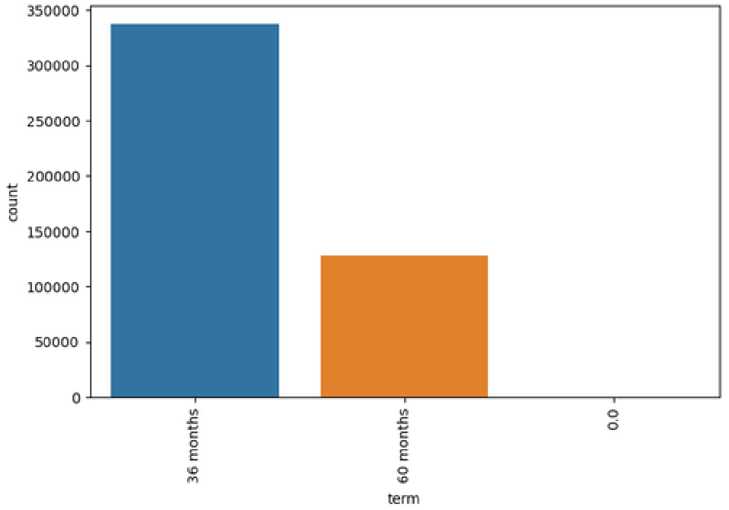
Univariate Analysis: Categorical Features

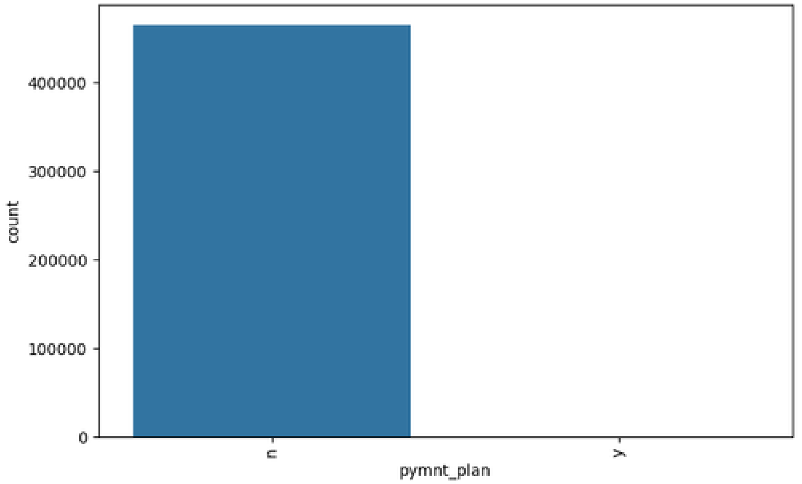


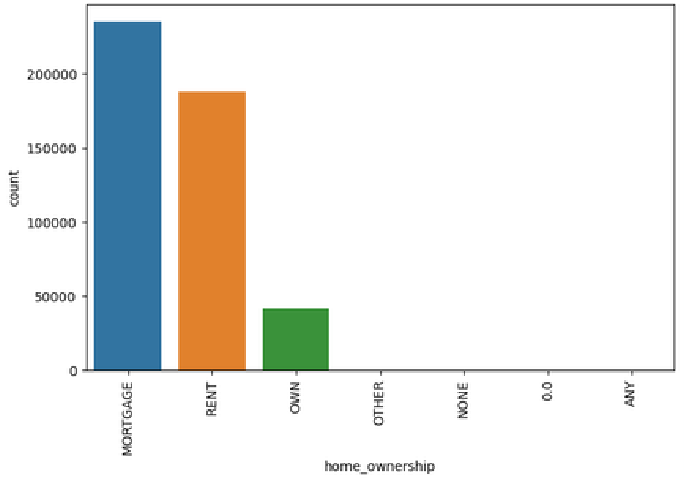


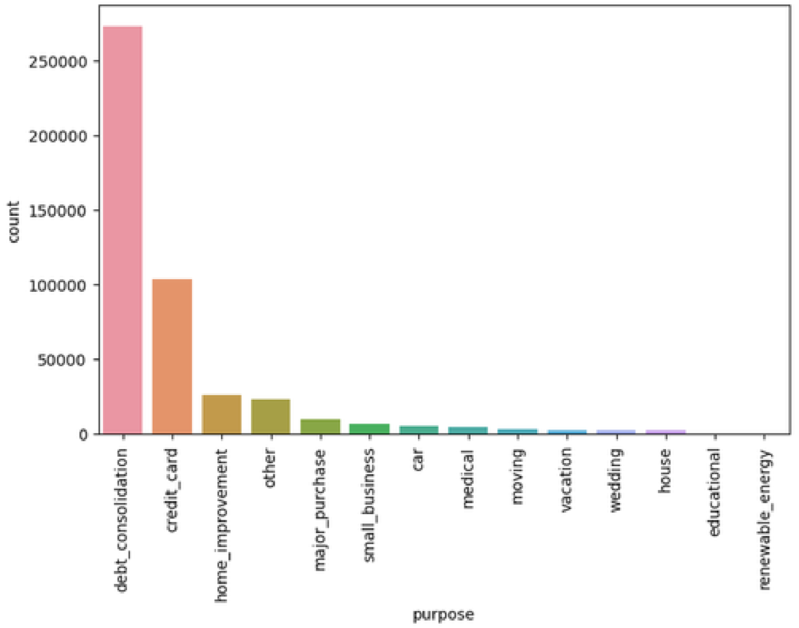






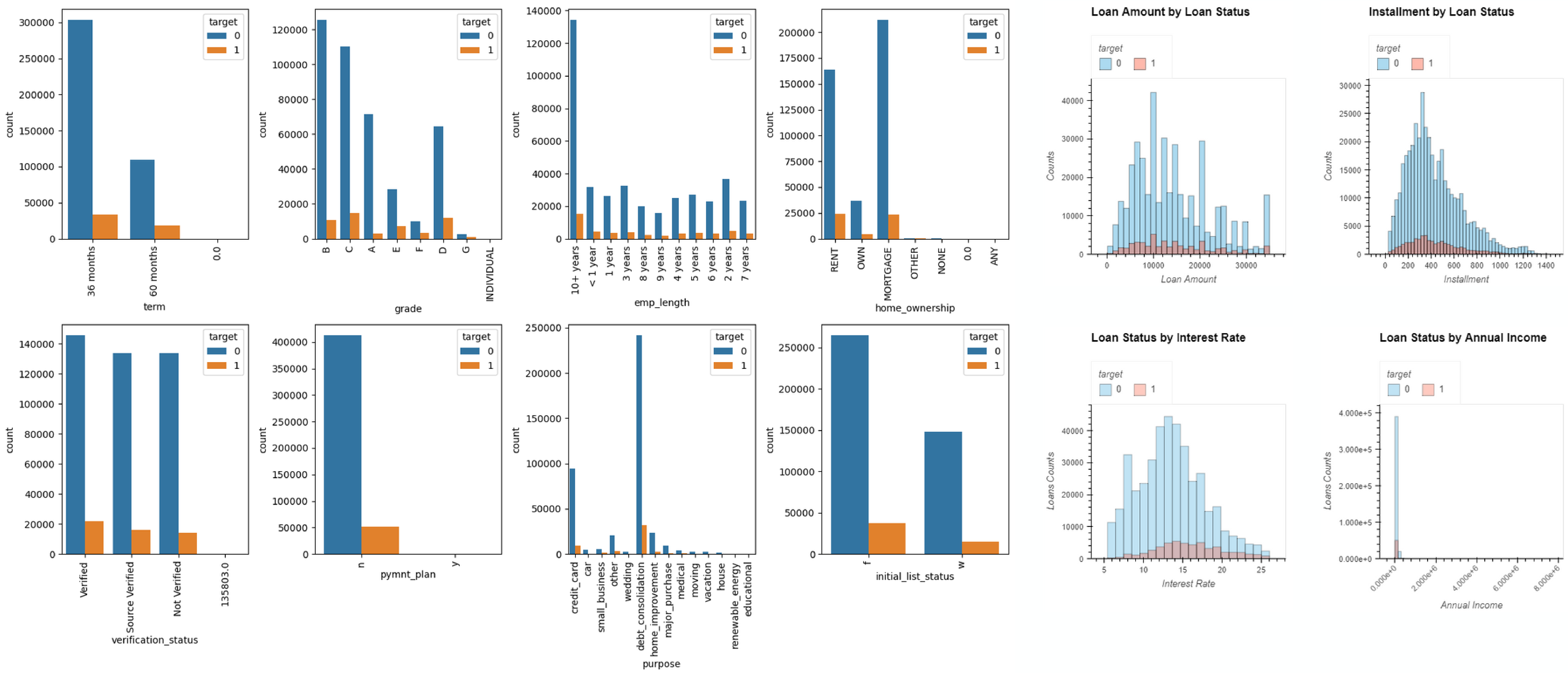






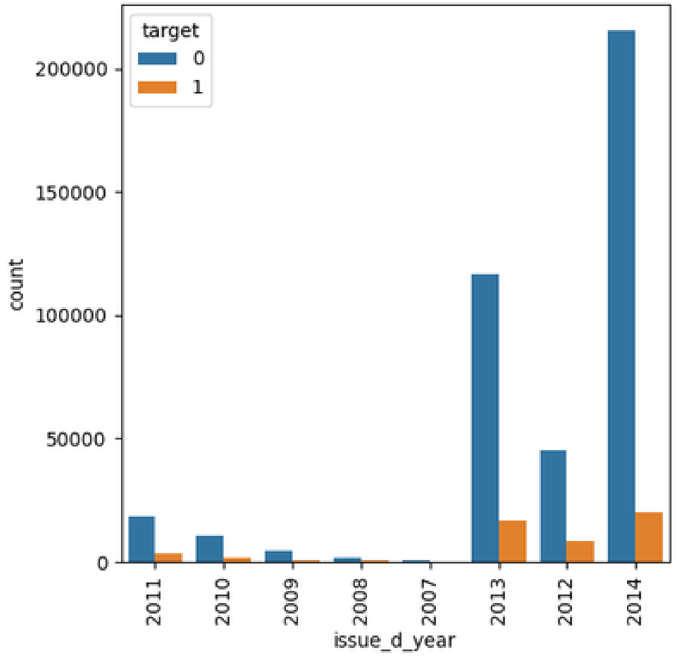
Exploratory Data Analysis

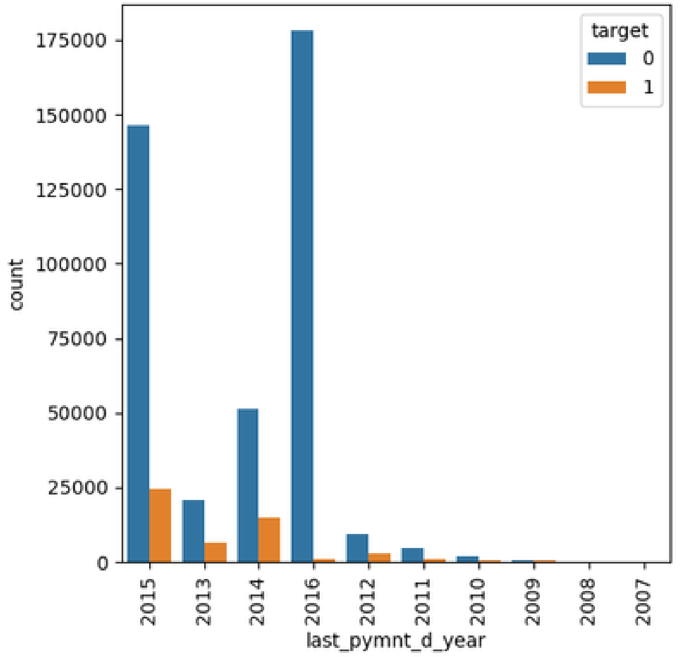
Bivariate Categorical Analysis

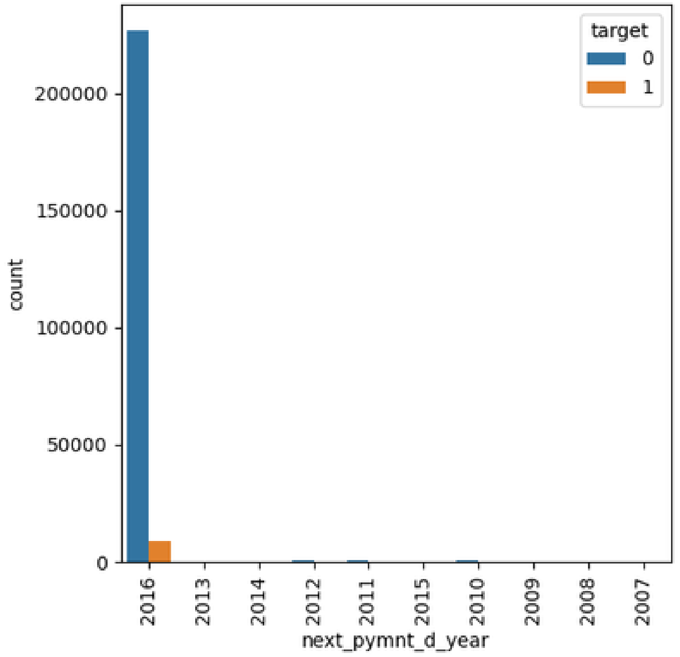


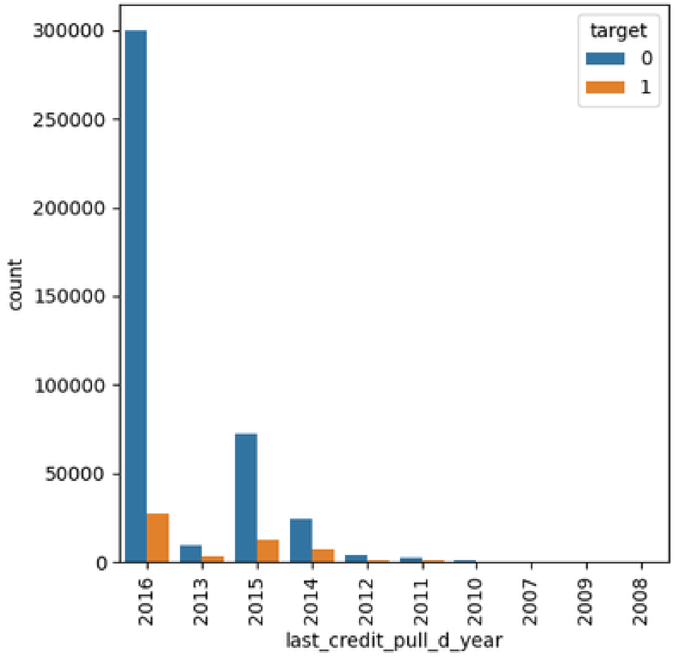
Exploratory Data Analysis

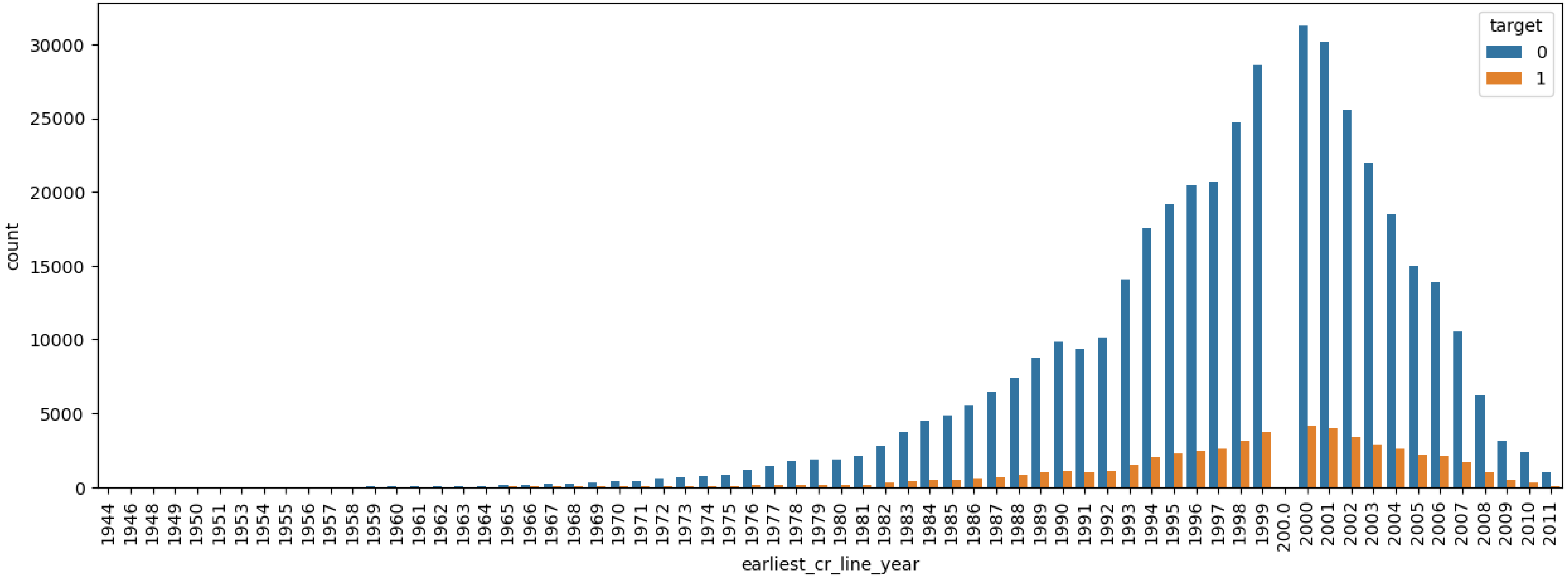
Bivariate Numerical Analysis









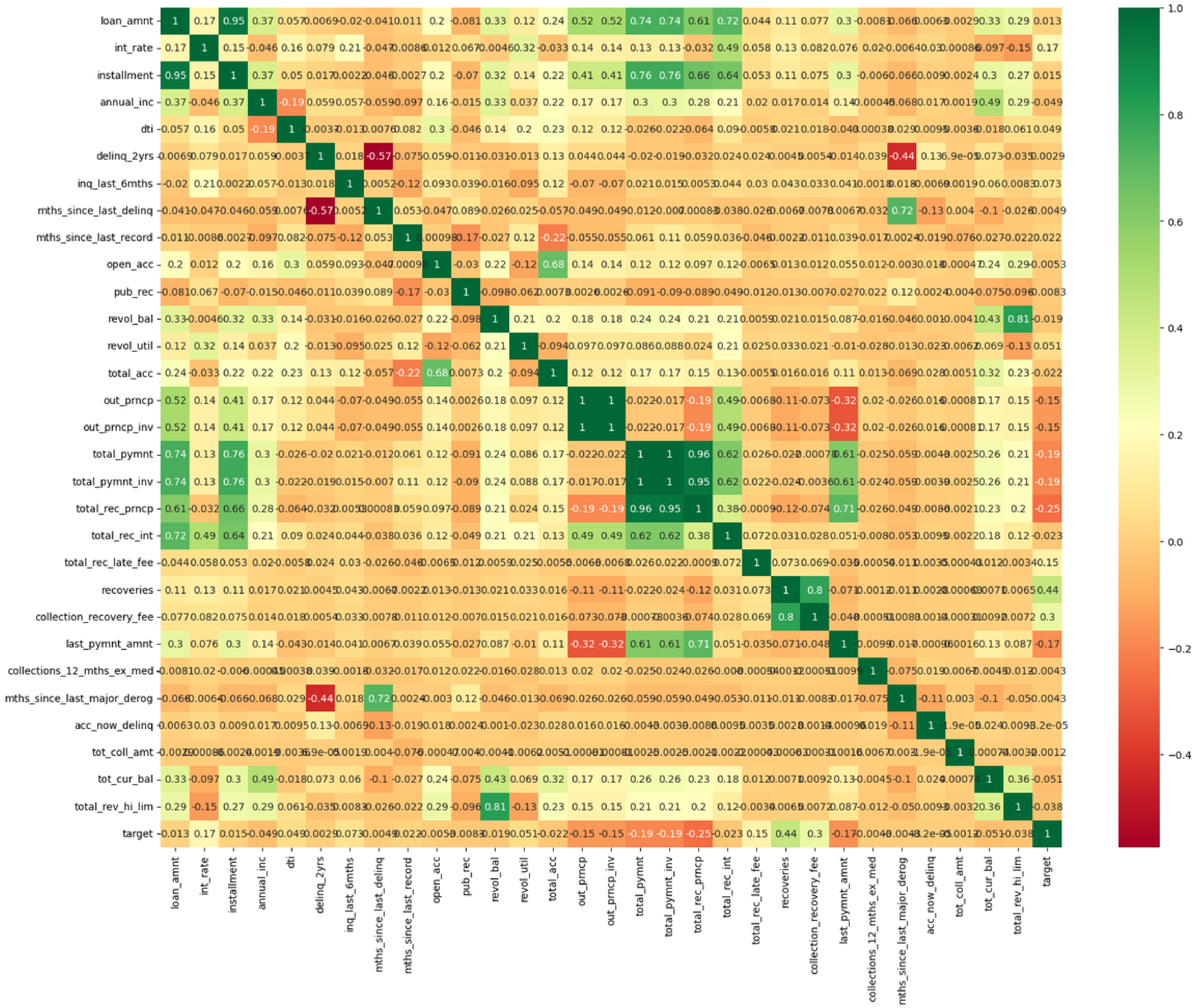


Exploratory Data Analysis

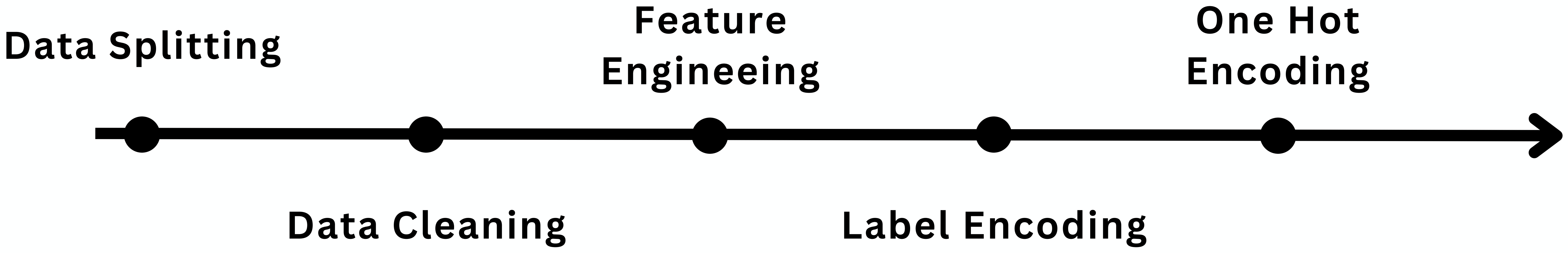
Correlation Matrix Heatmap

Correlation Matrix Heatmap menunjukan korelasi antara fitur atau fitur dengan target. semakin terang warnanya maka semakin dekat hubungannya

Dengan Correlation Matrix Heatmap mempermudah menganalisa fitur yang saling berhubungan dan saling mempengaruhi satu dengan yang lainnya.



Data Pre-Processing



Data Splitting

Data splitting dilakukan sebelum melakukan data cleaning, agar tidak menyebabkan data leakage saat proses feature engineering berlangsung

Data Cleaning

Proses data cleaning dibagi menjadi dua, yaitu categorical dan numerical data. metode ini mempermudah proses cleaning dengan jumlah rows yang banyak

Feature Engineering

Proses ini menyeleksi, mengisi, mengubah data dan menghapus data yang tidak diperlukan. langkahnya feature selection, handling missing value dan membagi proses menjadi kategorical dan numerical

Label Encoding

Pada tahap ini, melakukan labeling ulang pada data categorical, seperti gender, grade, dll

One Hot Encodeing

One Hot Encoding dilakukan karena ada beberapa data categorical yang dapat dipisahkan menjadi fitur baru

Handling Imbalance Dataset

```
y_train.value_counts(normalize=True) * 100
```

✓ 0.0s

```
0    88.803041
1    11.196959
Name: target, dtype: float64
```

X_train: (371975, 57)   X_test: (92994, 57)

y_train memiliki masalah ketidakseimbangan data, di mana nilai 1 adalah minoritas dan nilai 0 adalah mayoritas. Efek ketidakseimbangan ini dapat menyebabkan nilai f1 menurun sehingga harus dilakukan balancing data.

SMOTE Oversampling Methode

```
from imblearn.combine import SMOTETomek

# Implementing Oversampling for Handling Imbalanced
smk = SMOTETomek(random_state=42)
X_res, y_res = smk.fit_resample(X_train, y_train)
```

[138] ✓ 8m 50.7s

```
print("X_resampled.shape:", X_res.shape)
print("y_resampled.shape:", y_res.shape)
```

[139] ✓ 0.0s

```
X_resampled.shape: (654802, 61)
y_resampled.shape: (654802,)
```

Metode yang dipilih adalah SMOTE karena lebih efektif menurut beberapa penelitian. perhatikan tipe data saat melakukan SMOTE.

Modelling

Default Parameter 80:20 balance dataset

Decision Tree

```
from sklearn.tree import DecisionTreeClassifier
dt = DecisionTreeClassifier(random_state=42)
dt.fit(X_res,y_res)

y_train_pred_dt = dt.predict(X_res)
y_pred_dt = dt.predict(X_test)
```

[141] ✓ 49.0s

```
Accuracy:  0.956653117405424
AUC Score:  0.914825847016881
f1 Score:  0.8164306207022177
Precission Score:  0.7763055339049104
Recall Score:  0.8609296965040338
------------------------------
```

Random Forest

```
from sklearn.ensemble import RandomForestClassifier

rf = RandomForestClassifier(random_state=42)
rf.fit(X_res, y_res)

y_train_pred_rf = rf.predict(X_res)
y_pred_rf = rf.predict(X_test)
```

43] ✓ 6m 49.1s

```
Accuracy:  0.9931500957050993
AUC Score:  0.9703335681728903
f1 Score:  0.9685136671444812
Precission Score:  0.9977594459720949
Recall Score:  0.9409335382251248
------------------------------
```

Modelling

Hyperparameter 80:20 balance dataset

Random Forest

```
RandomizedSearchCV

RandomizedSearchCV(cv=5, estimator=RandomForestClassifier(random_state=42),
                   n_iter=3, n_jobs=1,
                   param_distributions={'max_depth': [5, 10, 15, 20, 25, 30],
                                        'max_features': ['sqrt', 'log2'],
                                        'min_samples_leaf': [1, 2, 5, 10],
                                        'min_samples_split': [2, 5, 10, 15,
                                                              100],
                                        'n_estimators': [100, 200, 300, 400,
                                                         500, 600, 700, 800,
                                                         900, 1000, 1100,
                                                         1200]},
                   verbose=2)
            estimator: RandomForestClassifier
                RandomForestClassifier
```

```
Hyperparameter Evaluation Random Forest
Accuracy:  0.99236509882358
AUC Score:  0.96653422919901
f1 Score:  0.9647537728355837
Precission Score:  0.9984586929716399
Recall Score:  0.9332500960430272
------------------------------

Default Evaluation Random Forest
Accuracy:  0.9931500957050993
AUC Score:  0.9703335681728903
f1 Score:  0.9685136671444812
Precission Score:  0.9977594459720949
Recall Score:  0.9409335382251248
------------------------------
```

Random Forest dengan hyperparameter menurunkan beberapa hasil penting, lebih baik default model karena target nya f1 score

Modelling

Hyperparameter 80:20 imbalance dataset

Random Forest

```
RandomizedSearchCV
RandomizedSearchCV(cv=5, estimator=RandomForestClassifier(random_state=42),
                   n_iter=5, n_jobs=1,
                   param_distributions={'max_depth': [5, 10, 15, 20, 25, 30],
                                        'max_features': ['sqrt', 'log2'],
                                        'min_samples_leaf': [1, 2, 5, 10],
                                        'min_samples_split': [2, 5, 10, 15,
                                                              100],
                                        'n_estimators': [100, 200, 300, 400,
                                                         500, 600, 700, 800,
                                                         900, 1000, 1100,
                                                         1200]},
                   verbose=2)
    estimator: RandomForestClassifier
        RandomForestClassifier
```

```
Hyperparameter Evaluation Random Forest
Accuracy:  0.9923328386777641
AUC Score:  0.9658026277014988
f1 Score:  0.9645502908566599
Precission Score:  0.9998969178435213
Recall Score:  0.9316173645793315
------------------------------

Default Evaluation Random Forest
Accuracy:  0.9931500957050993
AUC Score:  0.9703335681728903
f1 Score:  0.9685136671444812
Precission Score:  0.9977594459720949
Recall Score:  0.9409335382251248
------------------------------
```

Modelling

Hyperparameter 80:20 balance dataset

Decision Tree

```
RandomizedSearchCV

RandomizedSearchCV(cv=5, error_score='raise',
                   estimator=DecisionTreeClassifier(random_state=42), n_jobs=1,
                   param_distributions={'max_depth': [5, 10, 25, 50],
                                        'max_features': ['log2', 'sqrt'],
                                        'max_leaf_nodes': [10, 20, 30, 40, 50,
                                                           60, 70],
                                        'min_samples_leaf': [25, 50, 100, 125],
                                        'min_samples_split': [5, 10, 25, 50,
                                                              100]},
                   random_state=1, verbose=2)
  estimator: DecisionTreeClassifier
    DecisionTreeClassifier
```

```
Hyperparameter Evaluation for Decision Tree
------------------------------

Accuracy:  0.9351033400004302
AUC Score:  0.8726021667675392
f1 Score:  0.7321230414132895
Precision Score:  0.680614013369646
Recall Score:  0.7920668459469843
------------------------------

Default Evaluation for Decision Tree
------------------------------

Accuracy:  0.956653117405424
AUC Score:  0.914825847016881
f1 Score:  0.8164306207022177
Precision Score:  0.7763055339049104
Recall Score:  0.8609296965040338
------------------------------
```

Modelling

Hyperparameter 80:20 imbalance dataset

Decision Tree

```
RandomizedSearchCV

RandomizedSearchCV(cv=5, error_score='raise',
                   estimator=DecisionTreeClassifier(random_state=42), n_jobs=1,
                   param_distributions={'max_depth': [5, 10, 25, 50],
                                        'max_features': ['log2', 'sqrt'],
                                        'max_leaf_nodes': [10, 20, 30, 40, 50,
                                                           60, 70],
                                        'min_samples_leaf': [25, 50, 100, 125],
                                        'min_samples_split': [5, 10, 25, 50,
                                                              100]},
                   random_state=1, verbose=2)

estimator: DecisionTreeClassifier
DecisionTreeClassifier
```

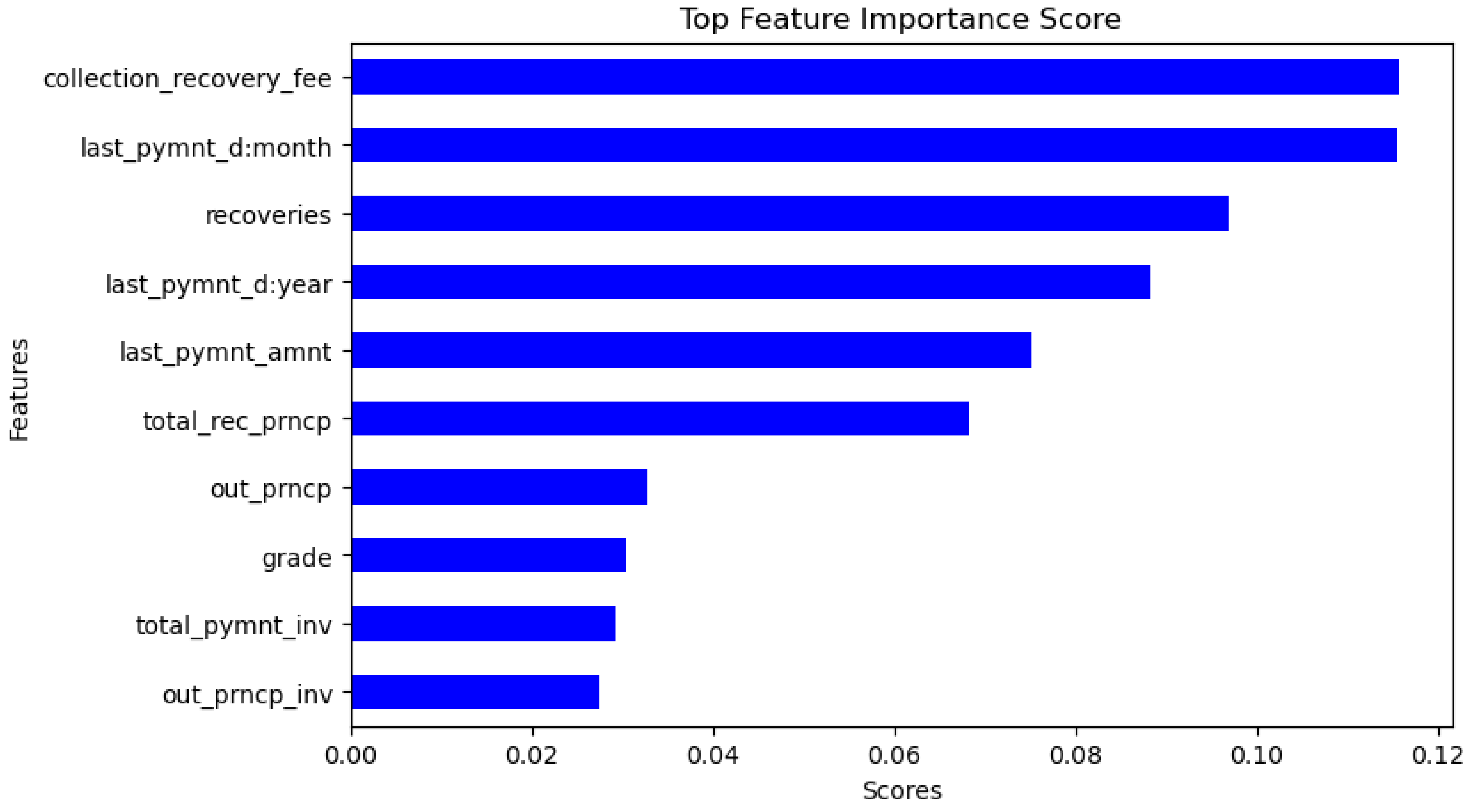
```
Hyperparameter Evaluation for Decision Tree
------------------------------

Accuracy:  0.9690517667806525
AUC Score:  0.8819801749011655
f1 Score:  0.8477892955362809
Precision Score:  0.9433851224105462
Recall Score:  0.7697848636189013
------------------------------

Default Evaluation for Decision Tree
------------------------------

Accuracy:  0.956653117405424
AUC Score:  0.914825847016881
f1 Score:  0.8164306207022177
Precision Score:  0.7763055339049104
Recall Score:  0.8609296965040338
------------------------------
```

Feature Importance Based on Best Model



GOT QUESTIONS?

Reach out.

https://www.linkedin.com/in/muhammad-hudzaifah-nasrullah-709033205/

ujai757@gmail.com