Rakamin Academy X id/x partners

# Credit Risk Prediction Modeling

## ID/X Partners - Data Scientist

Presented by

Zalfy Putra Rezky

Jakarta

zalfyputra@gmail.com

linkedin.com/in/zalfyputra

Rakamin Academy

## Biodata

| | |
|---|---|
| Nama | : Zalfy Putra Rezky |
| NPM | : 2106731453 |
| Universitas | : Universitas Indonesia |
| Jurusan | : Teknik Komputer |

# Project Portfolio

Perusahaan multifinance perlu meningkatkan keakuratan penilaian risiko kredit untuk mengoptimalkan keputusan bisnis dan mengurangi kerugian. Kami mengembangkan model machine learning menggunakan data pinjaman dari Lending Club (2007-2014) untuk memprediksi risiko kredit, dengan fokus pada metrik bisnis seperti kerugian dan margin keuntungan bersih. Analisis data ini bertujuan untuk mengidentifikasi pola yang mengindikasikan pinjaman berpotensi buruk atau berisiko, tanpa asumsi yang kuat, untuk mendukung pengambilan keputusan investasi.

Project explanation video here

# About Company

id/x partners didirikan pada tahun 2002 oleh mantan bankir dan konsultan manajemen yang memiliki pengalaman luas dalam manajemen siklus dan proses kredit, pengembangan skoring, dan manajemen kinerja. Pengalaman gabungan kami telah melayani korporasi di seluruh wilayah Asia dan Australia serta di berbagai industri, khususnya layanan keuangan, telekomunikasi, manufaktur, dan ritel.

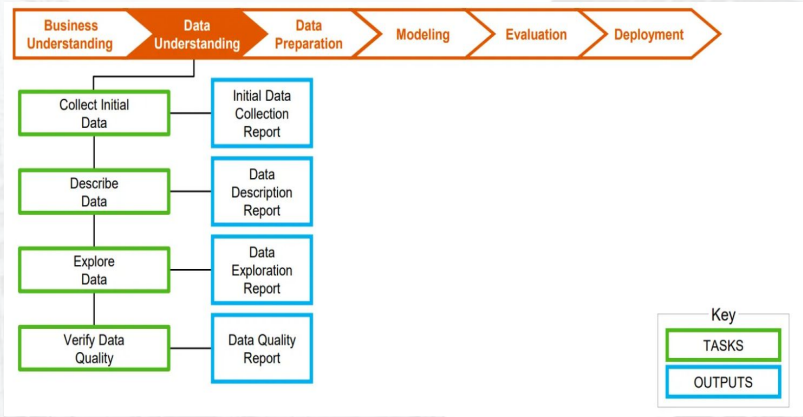
id/x partners menyediakan layanan konsultasi yang mengkhususkan diri dalam memanfaatkan solusi analitik data dan pengambilan keputusan (DAD) yang dikombinasikan dengan disiplin manajemen risiko dan pemasaran terintegrasi untuk membantu klien mengoptimalkan profitabilitas portofolio dan proses bisnis.

Layanan konsultasi yang komprehensif dan solusi teknologi yang ditawarkan oleh id/x partners menjadikannya sebagai penyedia layanan terpadu.

# 1. Data Understanding

Data Understanding adalah tahap kedua dalam proses CRISP-DM (Cross-Industry Standard Process for Data Mining) yang fokus pada pengumpulan dan penilaian kualitas data. Tahap ini melibatkan empat tugas utama:

1. **Mengumpulkan Data Awal**: Mengidentifikasi data yang tersedia, metode pengambilan, dan masalah yang mungkin dihadapi.
2. **Mendeskripsikan Data**: Memeriksa properti data yang diperoleh, termasuk format, kuantitas, dan isi dari setiap tabel atau dataset.
3. **Menjelajahi Data**: Menggunakan pertanyaan ilmu data untuk mendapatkan wawasan awal melalui kueri, visualisasi, dan laporan ringkasan.
4. **Memverifikasi Kualitas Data**: Memastikan data cukup bersih dan relevan untuk analisis yang akan dilakukan



# 1. Data Understanding

```
df.describe()
```
✓ 0.9s Python

| | Unnamed: 0 | id | member_id | loan_amnt | funded_amnt | funded_amnt_inv | int_rate | 4 |
|---|---|---|---|---|---|---|---|---|
| count | 466285.000000 | 4.662850e+05 | 4.662850e+05 | 466285.000000 | 466285.000000 | 466285.000000 | 466285.000000 | |
| mean | 233142.000000 | 1.307973e+07 | 1.459766e+07 | 14317.277577 | 14291.801044 | 14222.329888 | 13.829236 | |
| std | 134605.029472 | 1.089371e+07 | 1.168237e+07 | 8286.509164 | 8274.371300 | 8297.637788 | 4.357587 | |
| min | 0.000000 | 5.473400e+04 | 7.047300e+04 | 500.000000 | 500.000000 | 0.000000 | 5.420000 | |
| 25% | 116571.000000 | 3.639987e+06 | 4.379705e+06 | 8000.000000 | 8000.000000 | 8000.000000 | 10.990000 | |
| 50% | 233142.000000 | 1.010790e+07 | 1.194108e+07 | 12000.000000 | 12000.000000 | 12000.000000 | 13.660000 | |
| 75% | 349713.000000 | 2.073121e+07 | 2.300154e+07 | 20000.000000 | 20000.000000 | 19950.000000 | 16.490000 | |
| max | 466284.000000 | 3.809811e+07 | 4.086083e+07 | 35000.000000 | 35000.000000 | 35000.000000 | 26.060000 | |

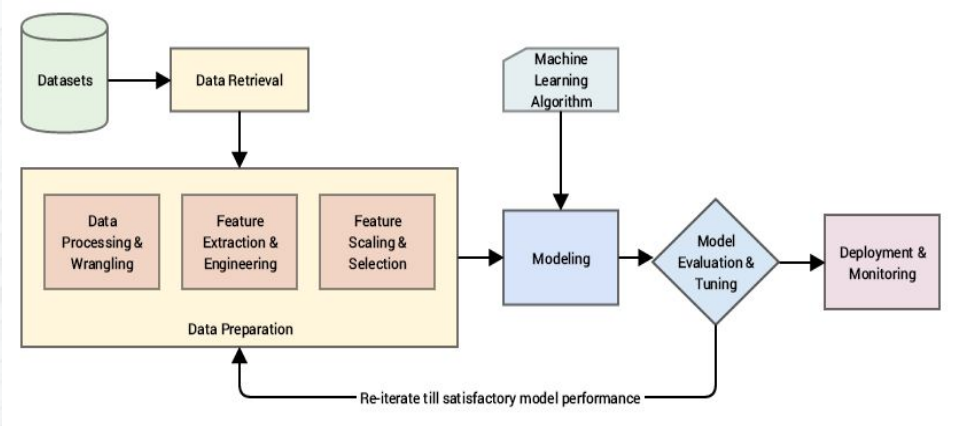
```
df = pd.read_csv('loan_data_2007_2014.csv')
df.info()
```
✓ 3.2s

```
C:\Users\ACER\AppData\Local\Temp\ipykernel_27292\1171339239
  df = pd.read_csv('loan_data_2007_2014.csv')
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 466285 entries, 0 to 466284
Data columns (total 75 columns):
 #   Column                      Non-Null Count   Dtype
---  ------                      --------------   -----
 0   Unnamed: 0                  466285 non-null  int64
 1   id                          466285 non-null  int64
 2   member_id                   466285 non-null  int64
 3   loan_amnt                   466285 non-null  int64
 4   funded_amnt                 466285 non-null  int64
 5   funded_amnt_inv             466285 non-null  float64
 6   term                        466285 non-null  object
 7   int_rate                    466285 non-null  float64
 8   installment                 466285 non-null  float64
 9   grade                       466285 non-null  object
 10  sub_grade                   466285 non-null  object
 11  emp_title                   438697 non-null  object
 12  emp_length                  445277 non-null  object
 13  home_ownership              466285 non-null  object
 14  annual_inc                  466281 non-null  float64
 15  verification_status         466285 non-null  object
 16  issue_d                     466285 non-null  object
 17  loan_status                 466285 non-null  object
 18  pymnt_plan                  466285 non-null  object
 19  url                         466285 non-null  object
...
 73  total_cu_tl                 0 non-null       float64
 74  inq_last_12m                0 non-null       float64
```

# 2. Feature Engineering

Feature Engineering dalam ilmu data adalah proses kreatif yang melibatkan pemilihan, transformasi, dan pembuatan fitur baru dari data mentah untuk meningkatkan kinerja model pembelajaran mesin. Ini termasuk:

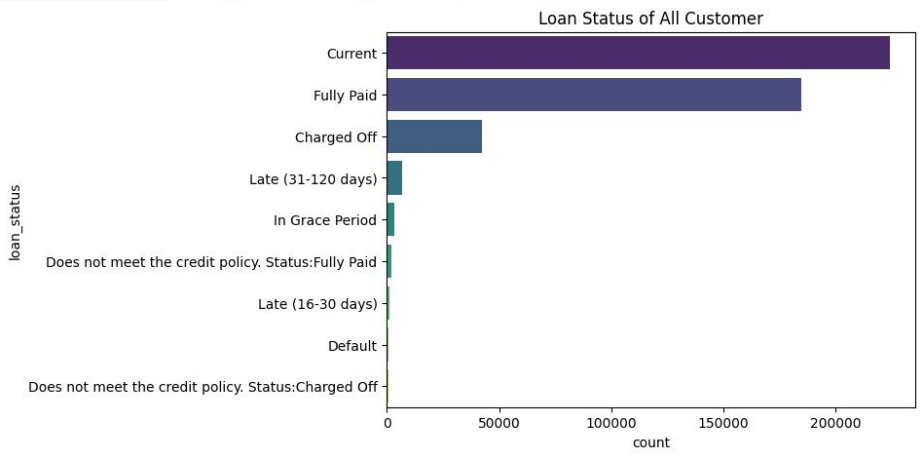
1. **Seleksi Fitur**: Memilih fitur yang paling relevan dengan masalah yang sedang dihadapi.
2. **Transformasi Fitur**: Mengubah skala atau distribusi fitur untuk meningkatkan interpretasi oleh model.
3. **Penciptaan Fitur**: Menggabungkan atau memodifikasi fitur untuk menghasilkan informasi yang lebih berguna.
4. **Ekstraksi Fitur**: Mengidentifikasi dan mengekstrak informasi penting dari data mentah.



# 2. Feature Engineering

Transformasi features yang dilakukan antara lain:

- Memberikan label Good Loan dan Bad Loan untuk status loan yang sesuai.
- Mengubah kolom kategori ke kolom numerik menggunakan One Hot Encoding.
- Menerapkan Feature Scaling agar jangkauan nilai-nilai tiap kolom mendekati satu sama lain.




## One Hot Encoding

Convert the remaining categorical columns into numerical columns.

```
# Get object columns and create dummy variables
onehot = pd.get_dummies(df.select_dtypes(include='object'))
onehot = onehot.astype(int)

# Combine the dummy variables with the original dataframe
df = pd.concat([df, onehot], axis=1)

# Drop the object columns
df = df.drop(columns=df.select_dtypes(include='object').columns)
df.info()
```

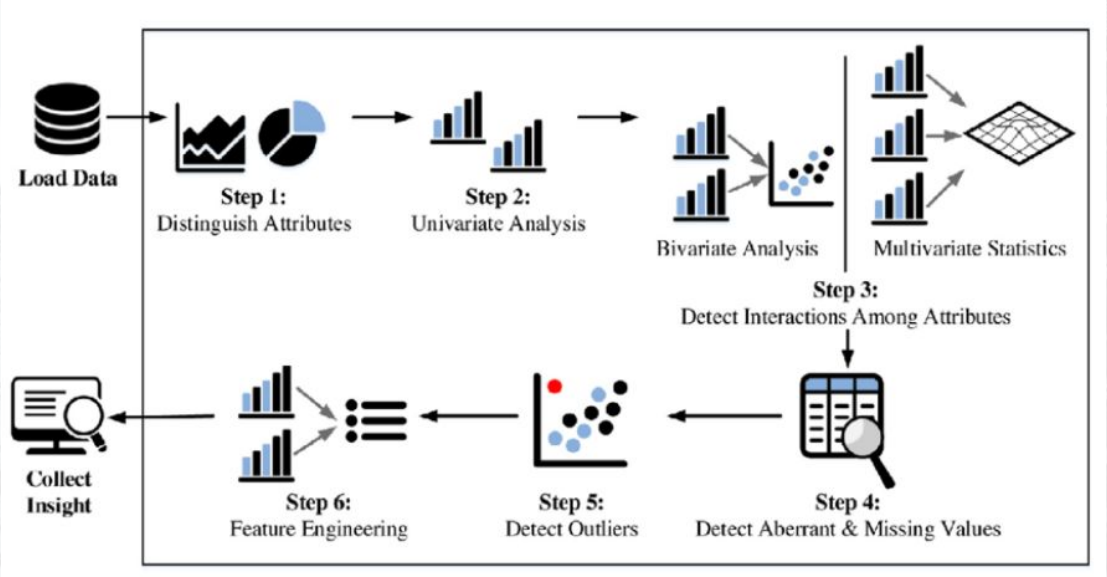
## Feature Scaling

```
sc = StandardScaler()
scaled = pd.DataFrame(sc.fit_transform(numerical_df), columns=numerical_df.columns)
scaled.head()
```

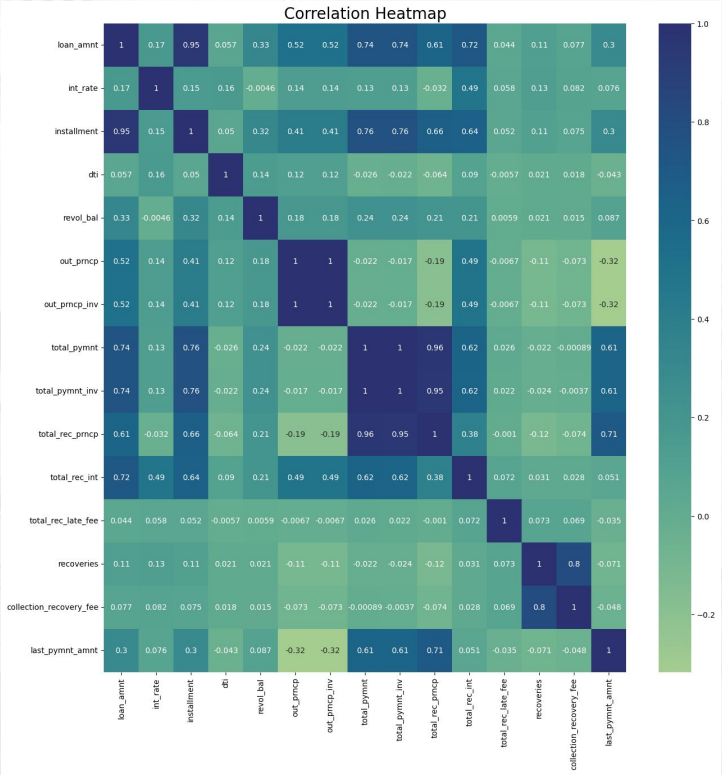
# 3. Exploratory Data Analysis

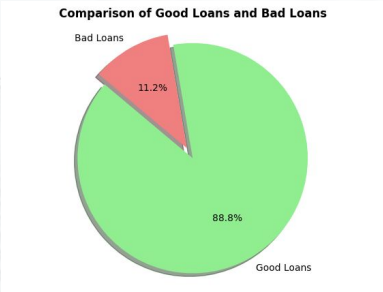
Exploratory Data Analysis (EDA) dalam ilmu data adalah proses analisis awal data untuk memahami karakteristik utama, menemukan pola, anomali, dan hubungan antar variabel. Proses EDA biasanya meliputi:

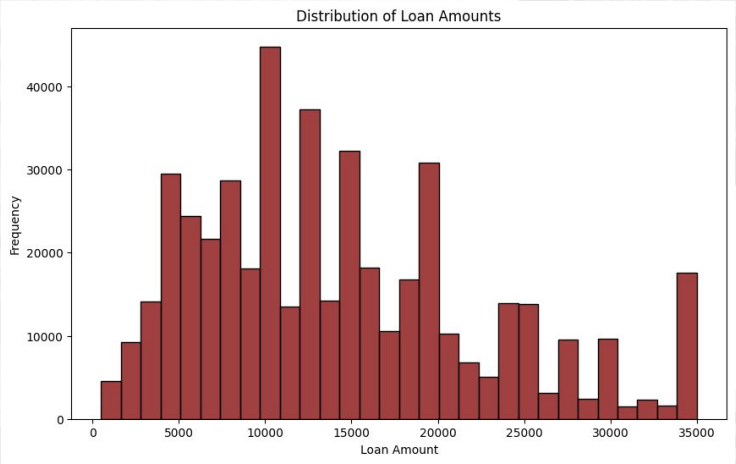
1. **Gambaran Umum Dataset**: Memahami jumlah observasi, jenis fitur, dan data yang hilang.
2. **Statistik Deskriptif**: Meringkas data numerik melalui ukuran tendensi sentral dan dispersi.
3. **Visualisasi Data**: Menggunakan grafik dan diagram untuk menggambarkan distribusi dan hubungan data.
4. **Evaluasi Kualitas Data**: Memeriksa kebersihan dan konsistensi data.



# 3. Exploratory Data Analysis



# 4. Data Preparation

Beberapa langkah yang dilakukan:

- Data cleaning dengan menghapus kolom-kolom yang berisi nilai NULL.
- Feature engineering dengan one hot encoding dan feature scaling.
- Data splitting untuk membagi data menjadi data training dan data testing.

```
# Drop unnecessary columns
columns_to_drop = [
    'issue_d',
    'pymnt_plan',
    'url',
    'zip_code',
    'addr_state',
    'application_type',
]

# Drop the columns
df = df.drop(columns=columns_to_drop)
df.info()
```

Data Splitting

```
# Define the features and target
X = df.drop(columns='loan_status', axis=1)
y = df['loan_status']
y.value_counts()
```

```
# Create a list of good loans
good_loans = [
    'Current',
    'Fully Paid',
    'In Grace Period',
    'Does not meet the credit policy. Status:Fully Paid'
]

# Update the loan status column
df['loan_status'] = np.where(df['loan_status'].isin(good_loans), 1, 0)
df['loan_status'].value_counts()
```

```
loan_status
1    414099
0     52186
Name: count, dtype: int64
```

```
# Split the data with a 70:30 ratio
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
print ('Train set:', X_train.shape,  y_train.shape)
print ('Test set:', X_test.shape,  y_test.shape)
```

# 5. Data Modeling

Logistic Regression dengan hyperparameter tuning menggunakan GridSearchCV

```
Best Parameters: {'C': 10, 'l1_ratio': 0.2, 'penalty': 'l2', 'solver': 'saga'}
Best ROC AUC Score (CV): 0.96542610922119
Training Accuracy: 0.9749026882700494
Testing Accuracy: 0.9741788820142188
Training ROC AUC: 0.9659976354664078
Testing ROC AUC: 0.9650360825400572
```

Naive Bayes

```
Training Accuracy: 0.9121888973481883
Testing Accuracy: 0.9102694703882819
Training ROC AUC: 0.8801140229033413
Testing ROC AUC: 0.8786825316442916
```

# 6. Evaluation

- **Logistic Regression**

  Model menunjukkan performa yang baik dengan Training ROC AUC sebesar 96.59% dan Testing ROC AUC sebesar 96.5%, dengan selisih skor hanya 0.09%, model yang dibuat termasuk *good fitting*.

- **Naive Bayes**

  Model menunjukkan performa yang baik dengan Training ROC AUC sebesar 88.01% dan Testing ROC AUC sebesar 87.86%, dengan selisih skor hanya 0.15%, model yang dibuat termasuk *good fitting*.

# 7. Conclusion

1. Training dengan Logistic Regression menghasilkan skor prediksi yang paling akurat, terutama ketika melakukan hyperparameter tuning.
2. Training dengan Naive Bayes dapat menghasilkan hasil akhir yang lebih cepat, tetapi kurang akurat karena data independen diasumsikan.
3. Berdasarkan hasil skor terbaik, model dapat memprediksi risiko peminjaman kredit dengan akurasi 96.5%.

# Thank You