



Nadine Haninta

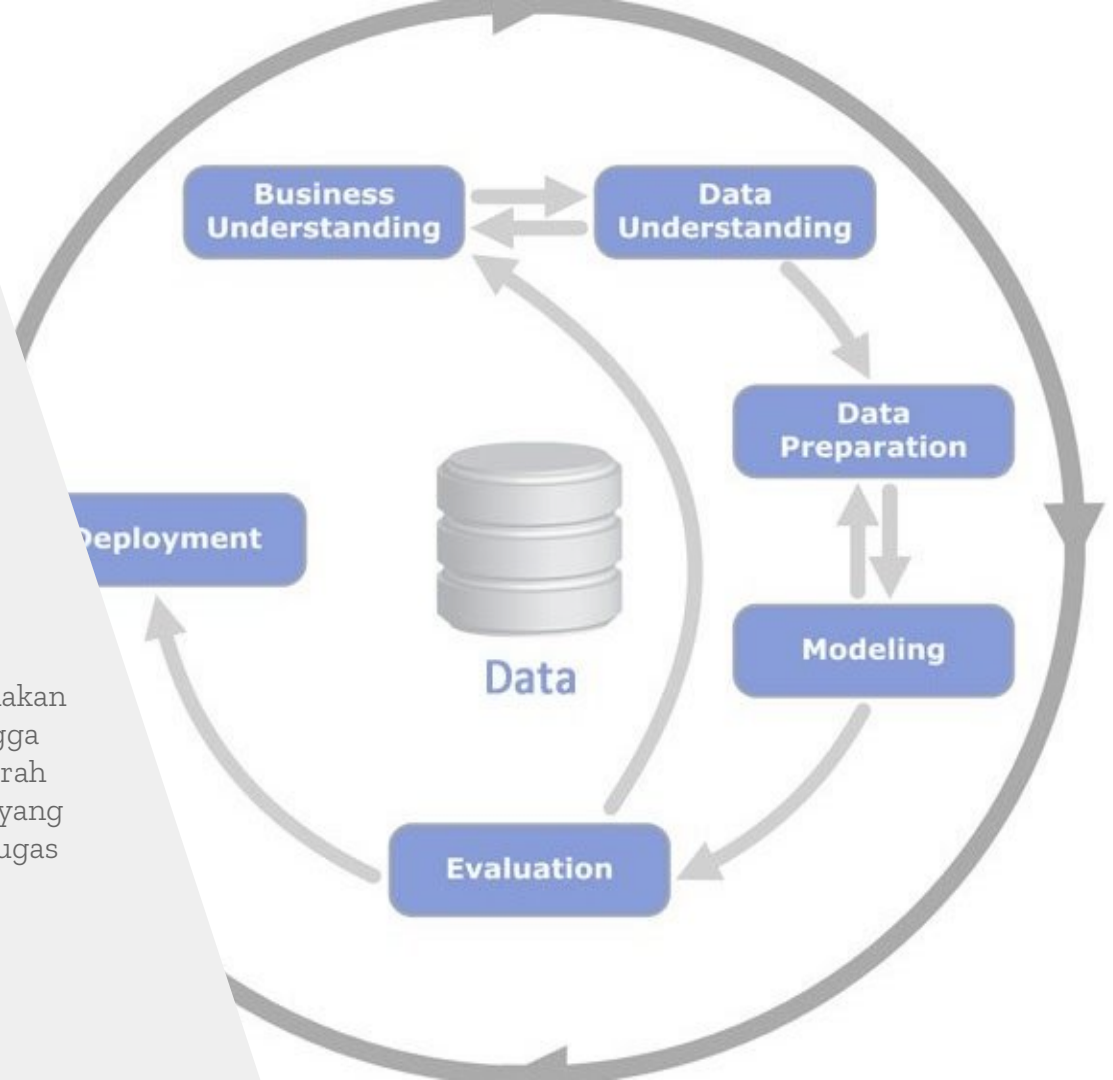
# Journey Pembelajaran VIX

BTPN SYARIAH: Data

Engineer

# KERANGKA YANG DIGUNAKAN

Dalam pengerjaan tugas ini, digunakan kerangka CRISP-DM (hanya hingga evaluasi) karena kerangka ini searah dengan deskripsi dan pencapaian yang harus dicapai dalam pengerjaan tugas ini.



## BUSINESS UNDERSTANDING



**Mengidentifikasi faktor apa saja yang menjadi penyebab customer terlambat membayar tunggakan kredit.**



**Mengidentifikasi faktor penyebab customer berhenti menggunakan layanan.**

# DATA UNDERSTANDING



## customer\_data\_history

Berisi data mengenai profil customer dan jumlah transaksi yang berkaitan dengan penggunaan data kredit.



## category\_db

Berisi data kategori jenis kartu yang digunakan oleh para pengguna layanan kartu kredit.



## status\_db

Berisi data terkait keaktifan pengguna, apakah pengguna masih aktif atau sudah berhenti menggunakan layanan?



## marital\_db

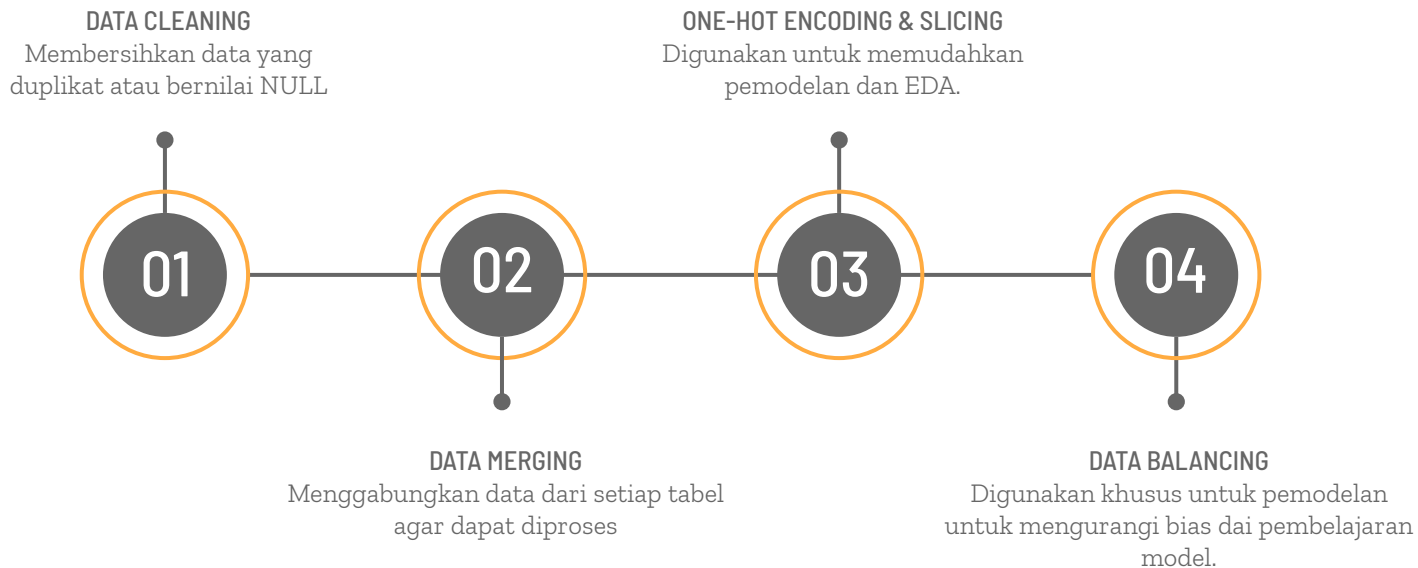
Berisi data kategori status pernikahan pengguna layanan kartu kredit.



## education\_db

Berisi data kategori pendidikan terakhir dari para pengguna

# DATA PREPARATION



Link Notebook (.ipynb): [NADINE HANINTA\\_TASK\\_5\\_DATA\\_ENGINEER\\_VIX\\_BTPNS](#)

## Mengidentifikasi faktor penyebab customer berhenti menggunakan layanan (df\_case2)

## DATA MODELLING

	accuracy	macro_avg_precision	macro_avg_recall	macro_avg_f1_score	roc_auc
model					
Logistic Regression	0.843529	0.844049	0.843657	0.843499	0.843657
Ridge Classifier	0.842588	0.844765	0.842850	0.842405	0.842850
KNN	0.826353	0.853196	0.827260	0.823262	0.827260
SVC	0.892706	0.894335	0.892921	0.892629	0.892921
Neural Network	0.931294	0.931902	0.931424	0.931281	0.931424
Decision Tree	0.928941	0.928981	0.928981	0.928941	0.928981
Gradient Boosting Classifier	0.960471	0.960542	0.960522	0.960471	0.960522
AdaBoost Classifier	0.947529	0.947821	0.947622	0.947526	0.947622
Hist Gradient Boosting	0.973412	0.973406	0.973421	0.973411	0.973421
XGBoost	0.960471	0.960542	0.960522	0.960471	0.960522
LightGBM	0.973647	0.973670	0.973630	0.973645	0.973630
Random Forest	0.967294	0.967385	0.967350	0.967294	0.967350
Catboost Classifier	0.975529	0.975532	0.975525	0.975528	0.975525

Digunakan uji coba berbagai model untuk mencari nilai akurasi, presisi, dan parameter lainnya yang seimbang dan tertinggi.

# CODE SNIPPET (DATA MODELLING)

## Python query:

```
from sklearn.linear_model import LogisticRegression
from sklearn.linear_model import RidgeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.ensemble import HistGradientBoostingClassifier
from sklearn.neural_network import MLPClassifier
from xgboost import XGBClassifier
from xgboost import XGBRFClassifier
from lightgbm import LGBMClassifier
from sklearn.ensemble import VotingClassifier
from sklearn.ensemble import StackingClassifier
from catboost import CatBoostClassifier
```

```
model_list = {
    'Logistic Regression':LogisticRegression(max_iter=1000,
random_state=42),
    'Ridge Classifier':RidgeClassifier(random_state=42),
    'KNN':KNeighborsClassifier(),
    'SVC':SVC(random_state=42),
    'Neural Network':MLPClassifier(max_iter=1000, random_state=42),
    'Decision Tree':DecisionTreeClassifier(random_state=42),
    'Gradient Boosting
Classifier':GradientBoostingClassifier(random_state=42),
    'AdaBoost Classifier':AdaBoostClassifier(random_state=42),
    'Hist Gradient
Boosting':HistGradientBoostingClassifier(random_state=42),
    'XGBoost':XGBClassifier(random_state=42, use_label_encoder=False,
eval_metric='logloss'),
    'LightGBM':LGBMClassifier(random_state=42),
    'Random Forest':RandomForestClassifier(random_state=42),
    'Catboost Classifier':CatBoostClassifier(random_state=42, silent=True),
}
```

Dideklarasikan jenis model yang digunakan dan metode pemodelan pada masing-masing jenis model.

## CODE SNIPPET (DATA MODELLING)

### Python query:

```
from sklearn.metrics import accuracy_score
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
from sklearn.metrics import f1_score
from sklearn.metrics import roc_auc_score
from sklearn.metrics import log_loss
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
from sklearn.metrics import make_scorer

def get_score(y_pred_list, y_test, average=None, plot=True, axis=0,
cmap="Blues"):
    model_name = []
    accuracy = []
    precision = []
    recall = []
    f1 = []
    roc_auc = []

    for name, y_pred in y_pred_list.items():
        model_name.append(name)
        if average != None:
            accuracy.append(accuracy_score(y_test, y_pred))
            precision.append(precision_score(y_test, y_pred,
average=average))
            recall.append(recall_score(y_test, y_pred, average=average))
            f1.append(f1_score(y_test, y_pred, average=average))
            roc_auc.append(roc_auc_score(y_test, y_pred, average=average))
```

```
score_list = {
    'model':model_name,
    'accuracy':accuracy,
    f'{average}_avg_precision':precision,
    f'{average}_avg_recall':recall,
    f'{average}_avg_f1_score':f1,
    'roc_auc':roc_auc
}
else:
    accuracy.append(accuracy_score(y_test, y_pred))
    precision.append(precision_score(y_test, y_pred))
    recall.append(recall_score(y_test, y_pred))
    f1.append(f1_score(y_test, y_pred))
    roc_auc.append(roc_auc_score(y_test, y_pred))

score_list = {
    'model':model_name,
    'accuracy':accuracy,
    'precision':precision,
    'recall':recall,
    'f1_score':f1,
    'roc_auc':roc_auc
}

score_df = pd.DataFrame(score_list).set_index('model')

if plot:
    display(score_df.style.background_gradient(axis=axis,
cmap=cmap))

return score_df
```

Dilakukan iterasi untuk  
tiap pemodelan dan  
pendeklarasian fungsi  
untuk mengembalikan  
nilai score\_list dari tiap  
model.



## CODE SNIPPET (DATA MODELLING)

**Python query:**

```
y_pred_list = dict()
```

```
for name, model in model_list.items():
```

```
    model.fit(X_train, y_train)
```

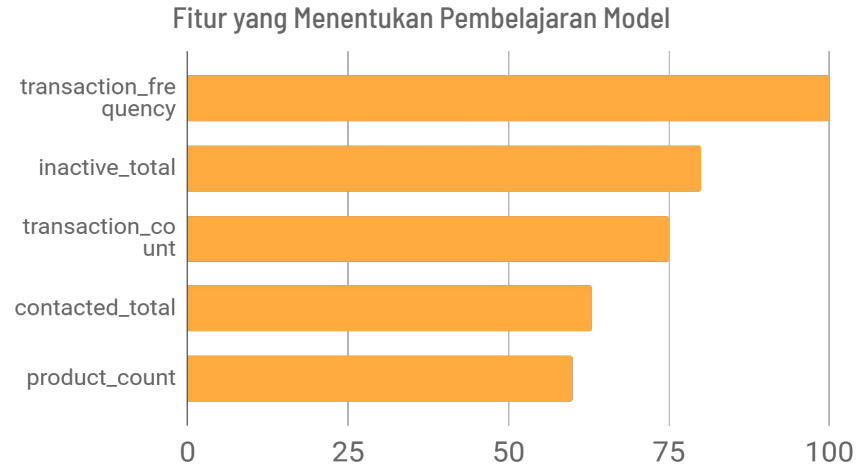
```
    y_pred_list[name] = model.predict(X_test)
```

```
score = get_score(y_pred_list, y_test, average='macro')
```

Digunakan untuk menampilkan hasil  
dari pembelajaran model

## Mengidentifikasi faktor penyebab customer berhenti menggunakan layanan (df\_case2)

## EXPLORATORY DATA ANALYSIS (EKSPLORASI DATA)



### Python Query:

```
feature_importance =  
catBoostModel.feature_importances_  
feature_importance = 100.0 * (feature_importance /  
feature_importance.max())  
sorted_idx = np.argsort(feature_importance)  
pos = np.arange(sorted_idx.shape[0]) + .5  
  
plt.figure(figsize=(8, 18))  
plt.barh(pos, feature_importance[sorted_idx],  
align='center')  
plt.yticks(pos, X_train.keys()[sorted_idx])  
plt.xlabel('Relative Importance')  
plt.title('Variable Importance')  
plt.show()
```

**5 Fitur teratas yang dijadikan pembelajaran model**

transaction\_frequency, inactive\_total, transaction\_count, contacted\_total, dan product\_count

## Mengidentifikasi faktor penyebab customer berhenti menggunakan layanan (df\_case2)

Rata-rata dari setiap fitur yang dikelompokkan berdasarkan keaktifan pengguna

is_attrited	transaction_frequency	inactive_total	transaction_count	contacted_total	product_count
0	68.672588	2.273765	4654.655882	2.356353	3.914588
1	44.933620	2.693301	3095.025814	2.972342	3.279656

is_attrited	transaction_frequency	inactive_total
Existing Customer	68.6726	2.2738
Attrited Customer	44.9336	2.6933

transaction_count	contacted_total	product_count
4654.6559	2.3564	3.9146
3095.0258	2.9723	3.2797

## EXPLORATORY DATA ANALYSIS (EKSPLORASI DATA)

### Python Query:

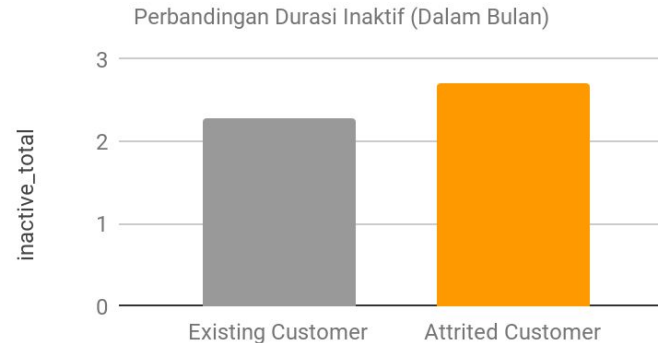
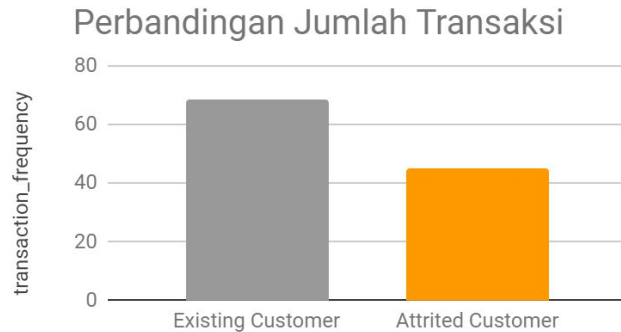
```
df_case2.groupby("is_attrited").mean()[['transaction_frequency','inactive_total','transaction_count','contacted_total','product_count']]
```

### SQL Query (Membuat VIEW):

```
CREATE VIEW view_df_case2_EDA1 AS
SELECT status is_attrited, AVG(total_trans_ct)
transaction_frequency,
AVG(months_inactive_12_mon) inactive_total,
AVG(total_trans_amt) transaction_count,
AVG(contacts_count_12_mon) contacted_total,
AVG(total_relationship_count) product_count
FROM customer_data_history, status_db
WHERE customer_data_history.idstatus =
status_db.id
GROUP BY status_db.status;
```

## Mengidentifikasi faktor penyebab customer berhenti menggunakan layanan (df\_case2)

## EXPLORATORY DATA ANALYSIS (INSIGHT)

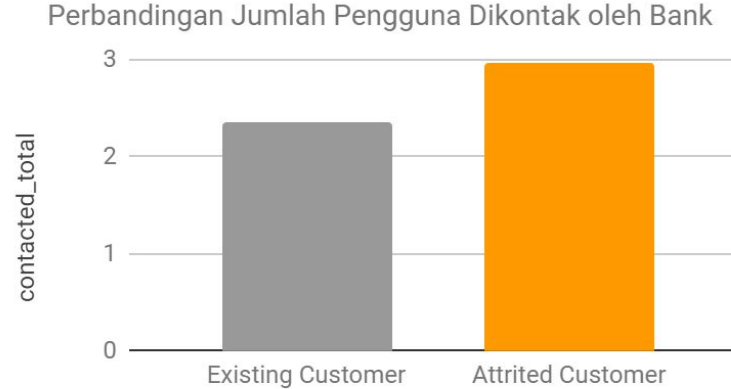


**Didapatkan adanya nominal atau angka yang cukup berbeda antar-kelompok**

Dari angka ini, dapat dilihat adanya kecenderungan bagi seseorang yang berhenti menggunakan layanan yaitu, lebih sedikit melakukan transaksi, lebih lama inaktif, lebih sering dikontak oleh pihak bank, dan lebih sedikit menggunakan produk layanan kredit dari bank.

## Mengidentifikasi faktor penyebab customer berhenti menggunakan layanan (df\_case2)

## EXPLORATORY DATA ANALYSIS (INSIGHT)

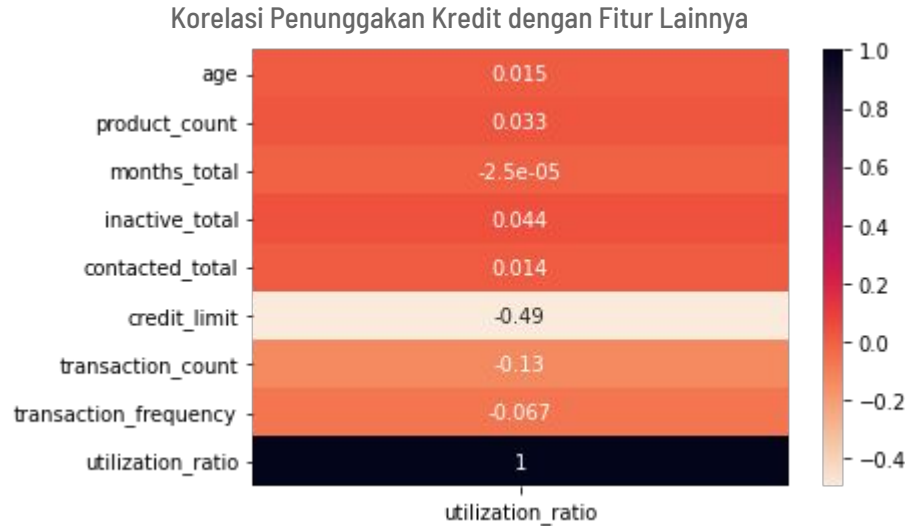


### Didapatkan adanya nominal atau angka yang cukup berbeda antar-kelompok

Dari angka ini, dapat dilihat adanya kecenderungan bagi seseorang yang berhenti menggunakan layanan yaitu, lebih sedikit melakukan transaksi, lebih lama inaktif, lebih sering dikontak oleh pihak bank, dan lebih sedikit menggunakan produk layanan kredit dari bank.

Mengidentifikasi faktor apa saja yang menjadi penyebab customer terlambat membayar tunggakan kredit. (df\_case1)

## EXPLORATORY DATA ANALYSIS (EKSPLORASI DATA)



### Python Query:

```
df_corr = df_case1.corr()[['utilization_ratio']]  
dataplot = sns.heatmap(df_corr, cmap="rocket_r",  
annot=True)
```

Perlu diketahui bahwa df\_case1 hanya menggunakan data dari pengguna yang memiliki utilization\_ratio di atas 0,3 atau 30%. Hal ini didasari oleh parameter umum yang digunakan sebagai ambang batas nilai rasio penggunaan.

3 Fitur teratas yang dapat ditelusuri lebih lanjut  
credit\_limit, transaction\_count, dan transaction\_frequency

## Mengidentifikasi faktor apa saja yang menjadi penyebab customer terlambat membayar tunggakan kredit (df\_case1)

## EXPLORATORY DATA ANALYSIS (EKSPLORASI DATA)

credit_limit	income	transaction_count	transaction_frequency	utilization_ratio
3544.0	Less than \$40K	15149	111	0.5
3421.0	\$40K - \$60K	992	21	0.7
2834.0	Less than \$40K	1598	39	0.5
5723.0	Less than \$40K	2732	63	0.3
2679.0	Less than \$40K	4943	85	0.9
1438.3	\$40K - \$60K	2928	48	0.6
2072.0	Less than \$40K	4210	64	0.7
2616.0	Less than \$40K	4655	68	0.3
4906.0	\$40K - \$60K	3999	71	0.4
1438.3	Less than \$40K	2336	46	0.6
3031.0	\$80K - \$120K	2188	50	0.8

### SQL Query (Membuat VIEW Gabungan):

```
CREATE VIEW view_df_case1_EDA1 AS  
SELECT credit_limit, income_category income,  
total_trans_amt transaction_count, total_trans_ct  
transaction_frequency, round(avg_utilization_ratio, 1)  
utilization_ratio  
FROM customer_data_history  
WHERE avg_utilization_ratio > 0.30
```

## Mengidentifikasi faktor apa saja yang menjadi penyebab customer terlambat membayar tunggakan kredit (df\_case1)

Rata-rata Batas Kredit (credit\_limit) dari  
Pembulatan Rasio Penggunaan

utilization_ratio	avg_credit_limit
0.3	4708.14358
0.4	3848.66256
0.5	2872.69986
0.6	2573.61283
0.7	2571.82067
0.8	2530.43285
0.9	2387.30652
1.0	2156.71053

## EXPLORATORY DATA ANALYSIS (EKSPLORASI DATA)

### SQL Query (Membuat VIEW):

```
CREATE VIEW view_df_case1_avg_credit_limit AS  
SELECT utilization_ratio, AVG(credit_limit)  
avg_credit_limit  
FROM view_df_case1_edat  
GROUP BY utilization_ratio  
ORDER BY utilization_ratio ASC;
```



## Mengidentifikasi faktor apa saja yang menjadi penyebab customer terlambat membayar tunggakan kredit (df\_case1)

Rata-rata Total Transaksi (transaction\_count) dari  
Pembulatan Rasio Penggunaan

utilization_ratio	avg_transaction_count
0.3	4343.8631
0.4	4461.0378
0.5	4335.0085
0.6	4060.3258
0.7	3967.7447
0.8	3493.6198
0.9	3407.8348
1.0	3047.2632

## EXPLORATORY DATA ANALYSIS (EKSPLORASI DATA)

### SQL Query (Membuat VIEW):

```
CREATE VIEW view_df_case1_avg_tran_count AS  
SELECT utilization_ratio, AVG(transaction_count)  
avg_transaction_count  
FROM view_df_case1_edat  
GROUP BY utilization_ratio  
ORDER BY `view_df_case1_edat`.`utilization_ratio`  
ASC
```

## Mengidentifikasi faktor apa saja yang menjadi penyebab customer terlambat membayar tunggakan kredit (df\_case1)

Rata-rata Frekuensi Transaksi (transaction\_frequency) dari  
Pembulatan Rasio Penggunaan

utilization_ratio	avg_transaction_frequency
0.3	64.0642
0.4	65.9409
0.5	67.1567
0.6	66.3597
0.7	66.1140
0.8	60.9855
0.9	61.0217
1.0	55.2632

## EXPLORATORY DATA ANALYSIS (EKSPLORASI DATA)

### SQL Query (Membuat VIEW)::

```
CREATE VIEW view_df_case1_avg_tran_freq AS  
SELECT utilization_ratio,  
AVG(transaction_frequency)  
  avg_transaction_frequency  
FROM view_df_case1_edat  
GROUP BY utilization_ratio  
ORDER BY `view_df_case1_edat`.`utilization_ratio`  
ASC
```

## Mengidentifikasi faktor apa saja yang menjadi penyebab customer terlambat membayar tunggakan kredit (df\_case1)

## EXPLORATORY DATA ANALYSIS (EKSPLORASI DATA)

Modus Jenis Pemasukan pada Masing-masing Rasio  
Penggunaan Kredit

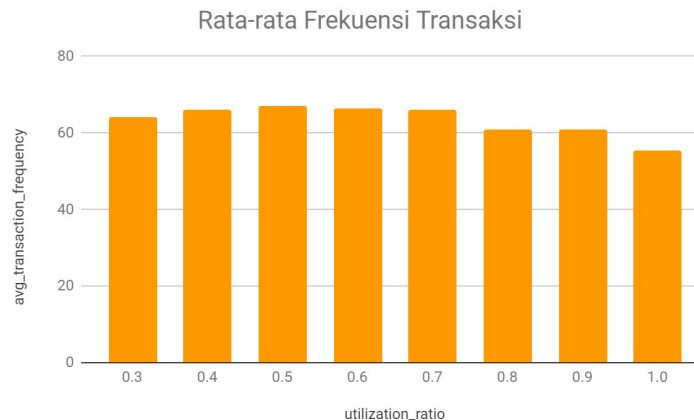
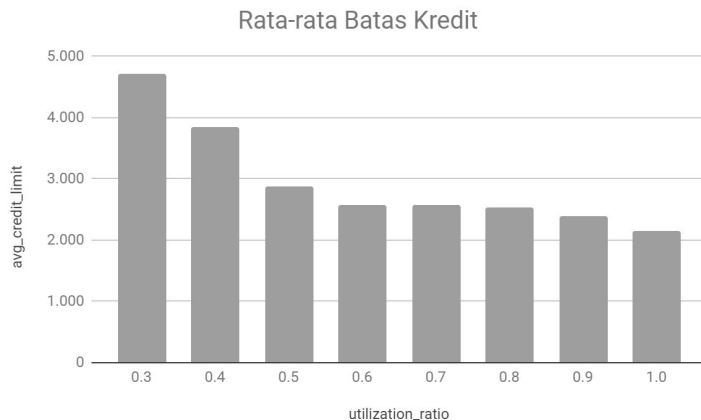
utilization_ratio	income
0.3	Less than \$40K
0.4	Less than \$40K
0.5	Less than \$40K
0.6	Less than \$40K
0.7	Less than \$40K
0.8	Less than \$40K
0.9	Less than \$40K
1.0	Less than \$40K

### SQL Query (Membuatt VIEW):

```
CREATE VIEW view_df_case1_income_mode AS(
    WITH look_for_mode AS(
        SELECT
            utilization_ratio,
            income,
            COUNT(income) AS frequency,
            RANK() OVER (PARTITION BY
utilization_ratio ORDER BY COUNT(income) DESC) As
rank_num
        FROM view_df_case1_eda1
        GROUP BY utilization_ratio, income
    )
    SELECT
        utilization_ratio,
        income
    FROM look_for_mode
    WHERE rank_num = 1
)
```

## Mengidentifikasi faktor apa saja yang menjadi penyebab customer terlambat membayar tunggakan kredit (df\_case1)

## EXPLORATORY DATA ANALYSIS (INSIGHT)



### Didapatkan Pola

Dari angka ini, dapat dilihat adanya kecenderungan bagi seseorang yang memiliki rasio penggunaan yang tinggi, semakin rendah pula batas kredit dan rata-rata total transaksi yang pernah mereka lakukan.

Hal ini sejalan dengan rendahnya batas saldo yang diberikan terhadap para pengguna.

## Mengidentifikasi faktor apa saja yang menjadi penyebab customer terlambat membayar tunggakan kredit (df\_case1)

## EXPLORATORY DATA ANALYSIS (INSIGHT)

utilization_ratio	income_mode
0.3	Less than \$40K
0.4	Less than \$40K
0.5	Less than \$40K
0.6	Less than \$40K

utilization_ratio	income_mode
0.7	Less than \$40K
0.8	Less than \$40K
0.9	Less than \$40K
1.0	Less than \$40K

### Sebagian Besar Pengguna Memiliki Pendapatan di Bawah \$40K

Didapatkan pada masing-masing rasio penggunaan, pendapatan mayoritas pengguna yang memiliki rasio penggunaan di atas 0,3 adalah di bawah \$40K, yang mana hal ini tidak menutup kemungkinan bahwa pendapatan jauh ada di bawah batas kredit (credit\_limit). Sehingga, menyebabkan ketidakmampuan pengguna untuk membayar tunggakan secara berketerusan.

## KESIMPULAN

### PENYEBAB KETERLAMBATAN PEMBAYARAN TUNGGAKAN

Pengguna dengan **credit limit** rendah cenderung terlambat membayar tunggakan, tapi hal ini tidak dapat dijadikan sebagai faktor utama. Sehingga, didapatkan penyebab utama keterlambatan adalah **dana/keuangan customer**.

### FAKTOR PENYEBAB CUSTOMER BERHENTI MENGGUNAKAN LAYANAN

Hasil EDA menunjukkan bahwa customer yang berhenti menggunakan layanan kredit adalah yang **sudah tidak membutuhkan layanan kartu kredit**. Hal ini terlihat dari jumlah transaksi dan frekuensi transaksi yang lebih sedikit dibanding customer yang masih menggunakan layanan hingga saat ini.

## Strategi Menghadapi Customer yang Terlambat Membayar Tunggakan Kredit

### Antisipasi

- ◀ **Memberi tahu risiko** dan kesepakatan pembayaran pada awal pembukaan layanan. Ditujukan agar customer mengerti hal apa saja yang akan dihadapi serta didapatkan melalui layanan kredit ini.
- ◀ **Melihat riwayat kredit pengguna** dan potensial dari pengguna. Agar tidak terjadi adanya hambatan dalam pembayaran tagihan.

### Menghadapi Kasus yang Sudah Terjadi

- ◀ **Mengusahakan *reconditioning* pada pengontrakan** dengan niat untuk membantu customer. *reconditioning* ini dapat berupa meningkatkan batas kredit atau penurunan bunga pembayaran apabila dalam usaha pengguna masih memiliki potensial.
- ◀ **Memberikan tambahan kredit atas aset yang dimiliki oleh customer**, selain membantu dalam pelunasann tunggakan, hal ini dapat membantu atas terbatasnya modal yang dimiliki oleh customer.

# SOLUSI

## Strategi Mencegah Seorang Customer Menggunakan Layanan Kredit

### Antisipasi

- ◀ **Rewarding** bagi pengguna kredit yang sudah lama menggunakan layanan,
- ◀ **Melakukan pengamatan lebih jauh** terkait penggunaan kredit bagi customer sehingga layanan menjadi lebih tepat dan sesuai bagi kebutuhan pengguna.

### Menghadapi Kasus yang Sudah Terjadi

- ◀ **Melakukan campaign** agar pengguna kembali menggunakan layanan dapat dilakukan. Adapun bentuknya dapat berupa penambahan produk dalam layanan dan dapat digunakan pula untuk strategi *upselling* layanan kredit tersebut.
- ◀ **Memberikan informasi secara berkala** pada beberapa akun yang masih memiliki potensi menggunakan layanan kembali.
- ◀ **Membuka layanan kritik & saran**, karena ada kemungkinan bahwa seorang customer terkendala dalam menggunakan layanan.

# SOLUSI



# TERIMA KASIH

Adapun *snippet* program lebih  
detil dapat dilihat di  
[bit.ly/NadineH\\_Task5](https://bit.ly/NadineH_Task5)



- ◀ Presentation template by [Slidesgo](#)
- ◀ Icons by [Flaticon](#)

# CREDITS