

Group 1 - Explorer



Dokumen
Laporan Final
Project



Project Team & Role



Project Manager

Imam Maghfir Ramadhan

Tugas: Leading project and set project timeline



Data Engineer

- Nivan Dumatubun
- Syaiful Adri

Tugas: Data pre-processing (Cleansing, encoding dll)



Data Analyst

- Putri Sausan
- Puspita Ayu Utami

Tugas: EDA, Insight (visualization data distribution), Recommendation



Data Scientist

- Marcellinus Putra Wijaya
- Muhamad Zen Fikri

Tugas: Modeling and Recommendation



Business Intelligence

- Wasis Prasetyo

Tugas: Visualization (showing to stakeholder), Recommendation

1. Latar Belakang Masalah

Masalah churn pada nasabah kartu kredit merupakan isu yang sangat umum untuk dihadapi lembaga keuangan. **Churn** terjadi ketika nasabah memutuskan untuk **berhenti** menggunakan produk dan layanan dari bank, termasuk kartu kredit. Hal ini **berdampak negatif** terhadap finansial bank karena dapat menyebabkan penurunan pendapatan hingga kerugian kerugian. Selain itu, **biaya untuk mendapatkan nasabah baru cenderung lebih tinggi** dibandingkan mempertahankan nasabah lama.

Case Study

Churn rate pada nasabah kartu kredit di sebuah bank dalam jangka waktu tertentu mencapai sebesar **20,37%**. Bila tidak ditindak, **bank akan mengalami kerugian yang semakin besar**.

Goal

Analytical approach yang akan dilakukan bertujuan **mengurangi atau menurunkan tingkat churn**.

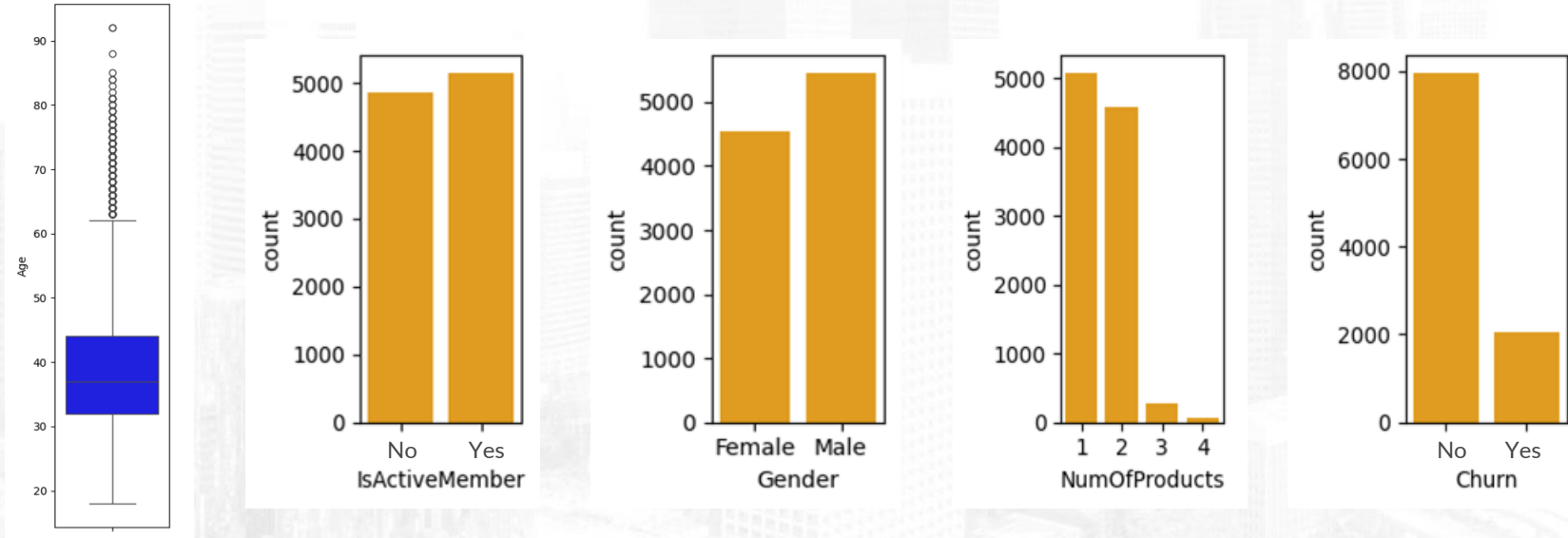
Objectives

- **Mengidentifikasi faktor-faktor** yang mempengaruhi nasabah untuk churn
- **Membangun model machine learning (ML)** yang akurat yang dapat memprediksi nasabah yang berpotensi churn berdasarkan faktor-faktor yang ada

Business Metric

Churn Rate (%)

2. EDA & Insight



Analysis

- Mayoritas nasabah customer yang ada berada pada umur **30 - 40 tahun**
- Mayoritas nasabah bank yang ada merupakan nasabah yang tergolong **aktif**, berjenis kelamin **laki-laki**, dan **memiliki 1 produk**
- Sekitar **2,037 (20.37%) nasabah churn** dari total 10,000 customer yang ada

2. EDA & Insight

Kelompok umur manakah yang paling banyak memiliki customer churn?

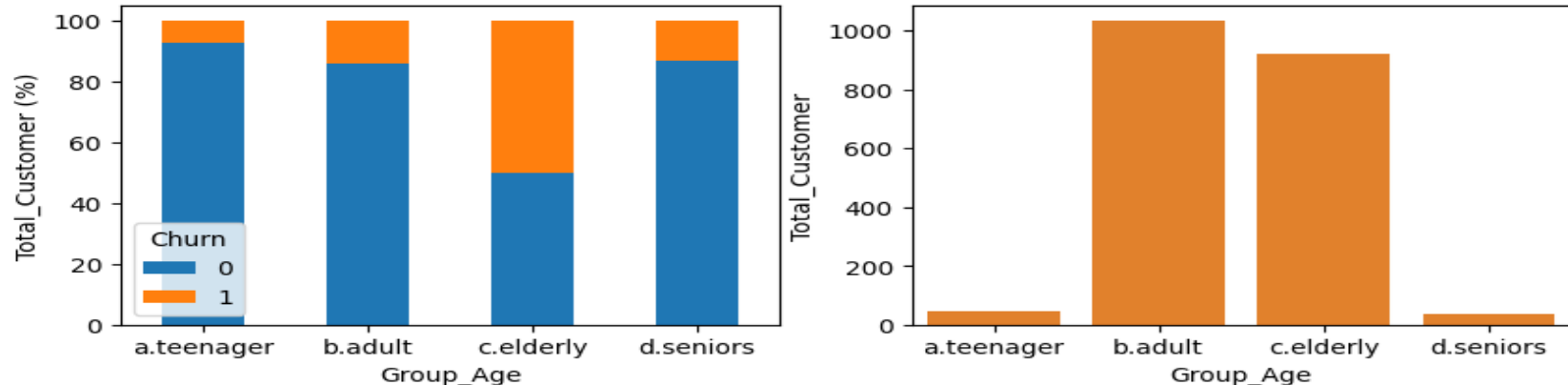
Berdasarkan Jurnal Urgensi Revisi Undang-Undang tentang Kesejahteraan Lanjut Usia (<http://jurnal.dpr.go.id/index.php/aspirasi/index>)

Penentuan range umur dapat dikelompokkan sebagai berikut:

- Umur 12 - 25 = Remaja (teenager)
- Umur 26 - 45 = Dewasa (adult)
- Umur 46 - 65 = Lansia (elderly)
- Umur > 65 = Manula (seniors)

Grafik Jumlah Customer Churn Sesuai Klasifikasi Umurnya

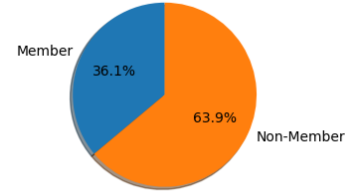
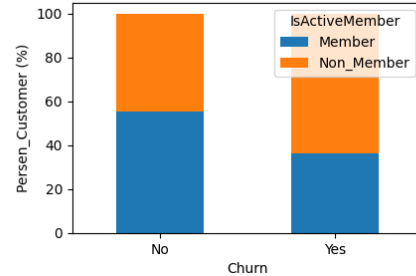
Jumlah nasabah churn terbanyak pada kelompok umur dewasa namun klasifikasi umur lansia juga mendominasi setelah kelompok umur dewasa



Dari grafik diatas, terlihat bahwa paling banyak customer churn adalah kelompok **Adult** dan **Eldery**

Customer Bank yang churn berdasarkan keaktifan member

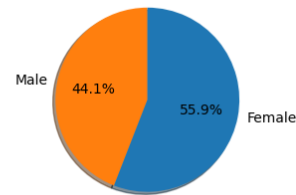
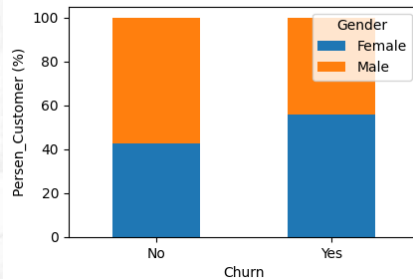
Grafik Jumlah Customer Churn Sesuai Keaktifan Member
Jumlah nasabah churn terbanyak pada orang yang bukan member (Non-Member)



Dari grafik diatas terlihat bahwa **customer yang bukan member aktif** (63,9 %) lebih banyak yang churn dibanding dengan yang **aktif** (36,1 %)

Customer Bank yang churn berdasarkan gender

Grafik Jumlah Customer Churn Sesuai Gender
Jumlah nasabah churn terbanyak pada wanita

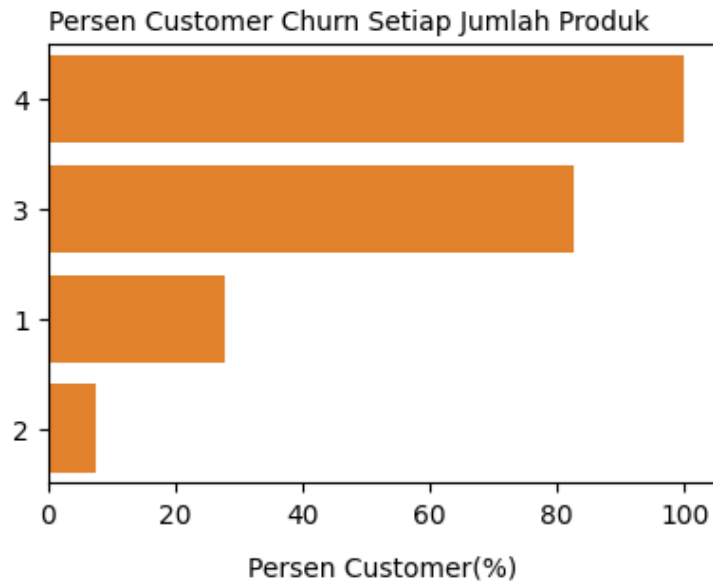
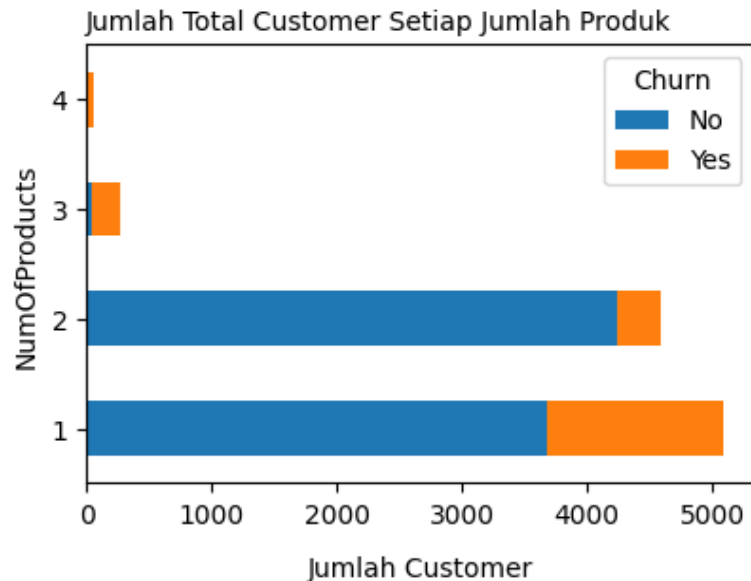


Dari grafik diatas terlihat bahwa **customer perempuan (female) sebesar 55,9%** lebih banyak yang churn dibanding laki-laki (male) sebesar 44,1 %

Customer Bank berdasarkan Kepemilikan Produk

Grafik Jumlah Produk yang Dipunyai Customer

Jumlah customer terbanyak berada di jumlah produk 1,
namun produk dengan jumlah 4 memiliki customer churn terbanyak secara persentase



Dari grafik diatas terlihat bahwa paling banyak customer bank memiliki **1 jenis produk** yaitu sebanyak **5.084 customer**. Tetapi untuk presentasi customer bank yang churn paling banyak adalah customer yang memiliki **4 produk**..

3. Pre-processing

Data Cleansing

- Handle Missing Value
- Handle Duplicated Data
- Handle Outlier

Feature Engineering

- Feature Selection
- Feature Extraction
- Feature Encoding
- Feature Transformation
- Handle Class Imbalance

3.1 Data Cleansing

Handle Missing Value, Duplicated Data & Outlier

Handle Missing Value

```
df.isna().sum() # menampilkan jumlah missing value setiap kolom
```

```
RowNumber      0  
CustomerId      0  
Surname         0  
CreditScore     0  
Geography       0  
Gender          0  
Age             0  
Tenure          0  
Balance         0  
NumOfProducts  0  
HasCrCard       0  
IsActiveMember  0  
EstimatedSalary 0  
Churn           0  
Group_Age       0  
dtype: int64
```

Semua tipe data sudah sesuai dan tidak ada data kosong

Handle Duplicated Data

```
df.duplicated().sum() # check data duplikat
```

0

Tidak ada data yang duplikat pada dataset

Handle Outlier

```
perbedaan jumlah data menggunakan z : 3.19%  
perbedaan jumlah data menggunakan IQR : 6.64%
```

Pada kasus ini, outlier kebanyakan terdapat pada kolom age, dimana distribusi dari umur nasabah masih tergolong wajar. Pada model ini outlier removal tidak dilakukan

3.2 Feature Engineering

Feature Selection & Feature Extraction

Feature Selection

Menghapus kolom 'RowNumber', 'Surname', 'CustomerId' karena tidak penting terhadap target

Feature Extraction

Balance_Category :

CreditScore_Range:

Tenure_Category:

Salary_Range:

Low = 0 - 97199

Poor = 349 - 584

Short Term = 0 - 3

Low = 0 - 51002

Medium = 97199 - 127644

Fair = 584 - 652

Medium Term = 3 - 5

Feature Tambahan: 51002 - 100194

†Menambahkan kolom baru berupa TenureByAge dan CreditScoreGivenAge^{5 - 10}

100194 - 149388

High =

3.2 Feature Engineering

Feature Encoding

- **One Hot Encoding**

Dilakukan pada kolom Geography menjadi tiga fitur yaitu France, Germany dan Spain

- **Label Encoding**

Dilakukan pada Gender, Grup Age, Balance Category, CreditScoreRange, dan Salary Range

| Geography_France | Geography_Germany | Geography_Spain |
|------------------|-------------------|-----------------|
| 1 | 0 | 0 |
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 0 | 1 |

| Gender | Group_Age | Balance_Category | CreditScore_Range | Tenure_Category | Salary_Range |
|--------|-----------|------------------|-------------------|-----------------|--------------|
| 1 | 1 | 0 | 1 | 0 | 2 |
| 1 | 1 | 0 | 1 | 0 | 2 |
| 1 | 1 | 2 | 0 | 2 | 2 |
| 1 | 1 | 0 | 2 | 0 | 1 |
| 1 | 1 | 1 | 3 | 0 | 1 |
| ... | ... | ... | ... | ... | ... |
| 0 | 1 | 0 | 3 | 1 | 1 |

3.2 Feature Engineering

Feature Transformation & Handle Class Imbalance

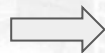
Feature Transformation

Menggunakan Standardisasi karena distribusi data relatif normal dan standardisasi lebih robust terhadap outlier

Handle Class Imbalance

Menggunakan Undersampling karena target prediksi category churn (value=1) lebih sedikit, jika menggunakan oversampling akan menyebabkan bias karena data sintesis yang tergenerate akan lebih banyak

```
Class distribution before oversampling:
0      6356
1      1644
Name: Churn, dtype: int64
```



```
Class distribution after undersampling:
0      1644
1      1644
Name: Churn, dtype: int64
```

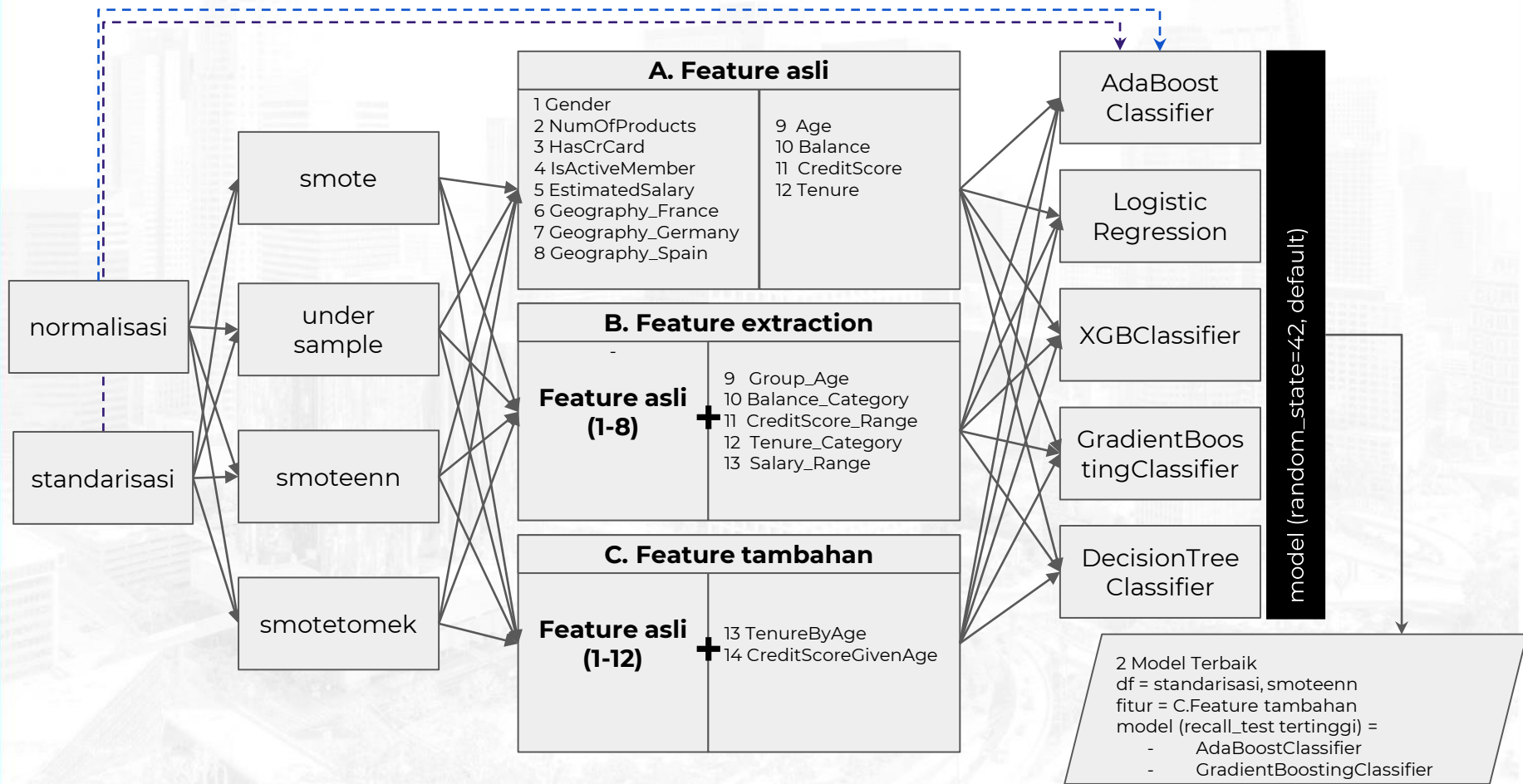

4.1 Modelling - Pemilihan Metrik untuk Evaluasi

| | | True Class | |
|-----------------|----------|------------|----------|
| | | Positive | Negative |
| Predicted Class | Positive | TP | FP |
| | Negative | FN | TN |

Recall

Apa alasannya ?

4.2 Modelling - Modelling Experiments



4.3 Modelling - Modelling Hypertuning Parameter

parameter default

| AdaBoostClassifier (random_state=42,default) | | GradientBoostingClassifier (random_state=42,default) | |
|---|-----------|---|-----------|
| Parameter | Nilai (%) | Parameter | Nilai (%) |
| recall train | 90.4 | recall train | 93.6 |
| recall test | 82.4 | recall test | 81.4 |

Hypertuning Parameter

- Manual Tuning
- GridSearchCV



| AdaBoost Classifier | Manual | GridSearchCV |
|---------------------|---------|--------------|
| algorithm | SAMME.R | SAMME.R |
| n_estimator | 10 | 200 |
| learning_rate | 0.8 | 0.01 |

| GradientBoosting Classifier | Manual | GridSearchCV |
|-----------------------------|---------|--------------|
| n_estimator | 12 | 10 |
| learning_rate | 0.2 | 0.01 |
| max_depth | 3 | 3 |
| min_samples_leaf | 65 | 1 |
| min_samples_split | default | 2 |
| subsample | default | 0.8 |

Hasil Tuning

| AdaBoost Classifier | Manual | GridSearchCV | Gradient Boosting | Manual | GridSearchCV |
|--|--------|--------------|--|--------|--------------|
| recall train (%) | 88.2 | 90.7 | recall train (%) | 90.1 | 100 |
| recall test (%) | 84.7 | 86.5 | recall test (%) | 83.7 | 100 |
| recall train (%) (cross validation) | 83.4 | 78.6 | recall train (%) (cross validation) | 84.4 | 87 |
| recall test (%) (cross validation) | 83.2 | 78.3 | recall test (%) (cross validation) | 85.1 | 86.2 |

4.3 Modelling - Modelling Hypertuning Parameter (Lanjutan)

AdaBoostClassifier Hypertuning (manual)

| | | | | |
|----------------|-----------|--------|----------|---------|
| report test : | | | | |
| | precision | recall | f1-score | support |
| 0 | 0.95 | 0.69 | 0.80 | 1607 |
| 1 | 0.40 | 0.85 | 0.54 | 393 |
| accuracy | | | 0.72 | 2000 |
| macro avg | 0.67 | 0.77 | 0.67 | 2000 |
| weighted avg | 0.84 | 0.72 | 0.75 | 2000 |
| report train : | | | | |
| | precision | recall | f1-score | support |
| 0 | 0.84 | 0.83 | 0.83 | 4111 |
| 1 | 0.87 | 0.88 | 0.88 | 5531 |
| accuracy | | | 0.86 | 9642 |
| macro avg | 0.86 | 0.86 | 0.86 | 9642 |
| weighted avg | 0.86 | 0.86 | 0.86 | 9642 |

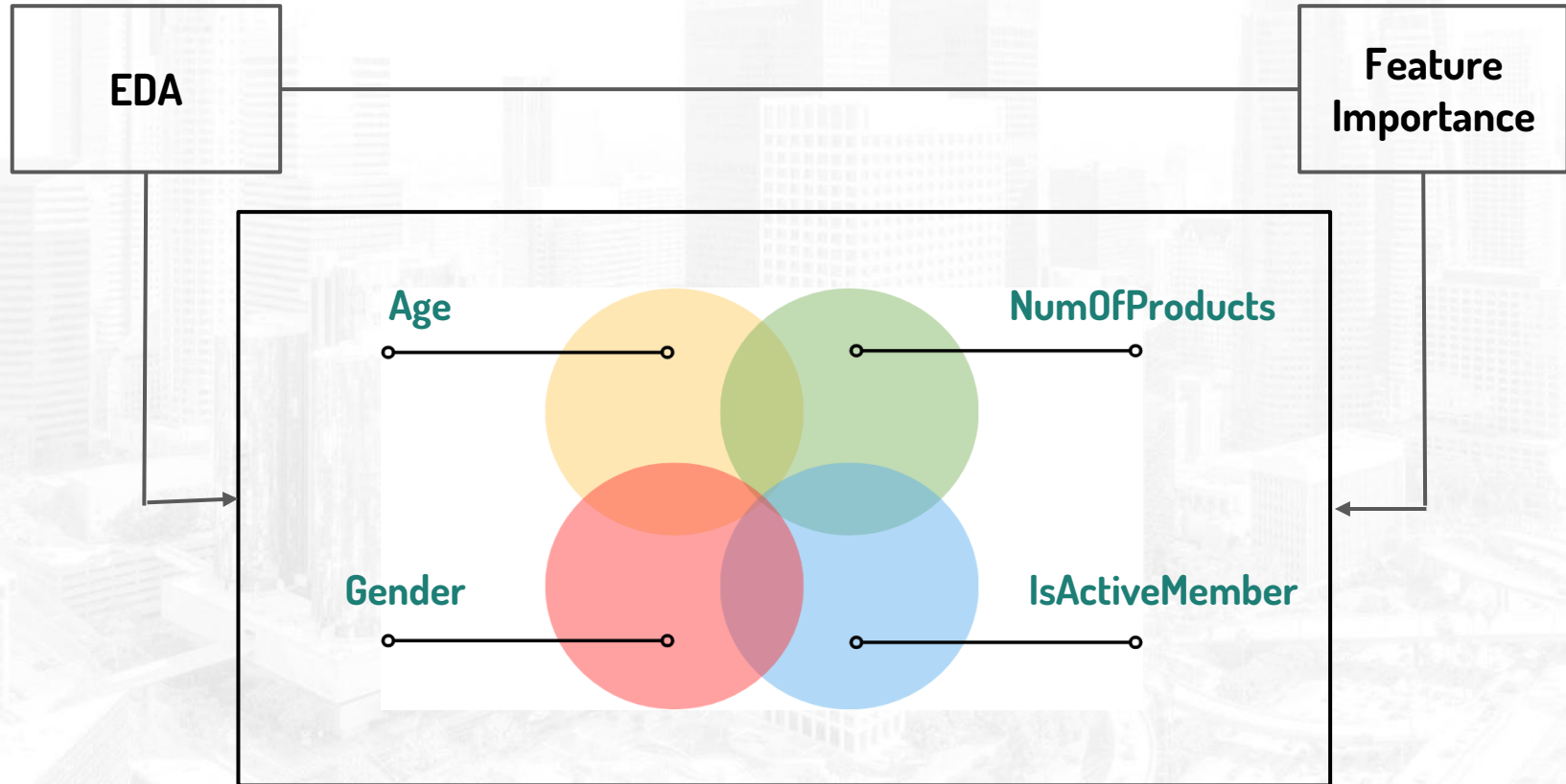
GradientBoostingClassifier Hypertuning (GridSearchCV)

| | | | | |
|----------------|-----------|--------|----------|---------|
| report test : | | | | |
| | precision | recall | f1-score | support |
| 0 | 0.00 | 0.00 | 0.00 | 1607 |
| 1 | 0.20 | 1.00 | 0.33 | 393 |
| accuracy | | | 0.20 | 2000 |
| macro avg | 0.10 | 0.50 | 0.16 | 2000 |
| weighted avg | 0.04 | 0.20 | 0.06 | 2000 |
| report train : | | | | |
| | precision | recall | f1-score | support |
| 0 | 0.00 | 0.00 | 0.00 | 4111 |
| 1 | 0.57 | 1.00 | 0.73 | 5531 |
| accuracy | | | 0.57 | 9642 |
| macro avg | 0.29 | 0.50 | 0.36 | 9642 |
| weighted avg | 0.33 | 0.57 | 0.42 | 9642 |

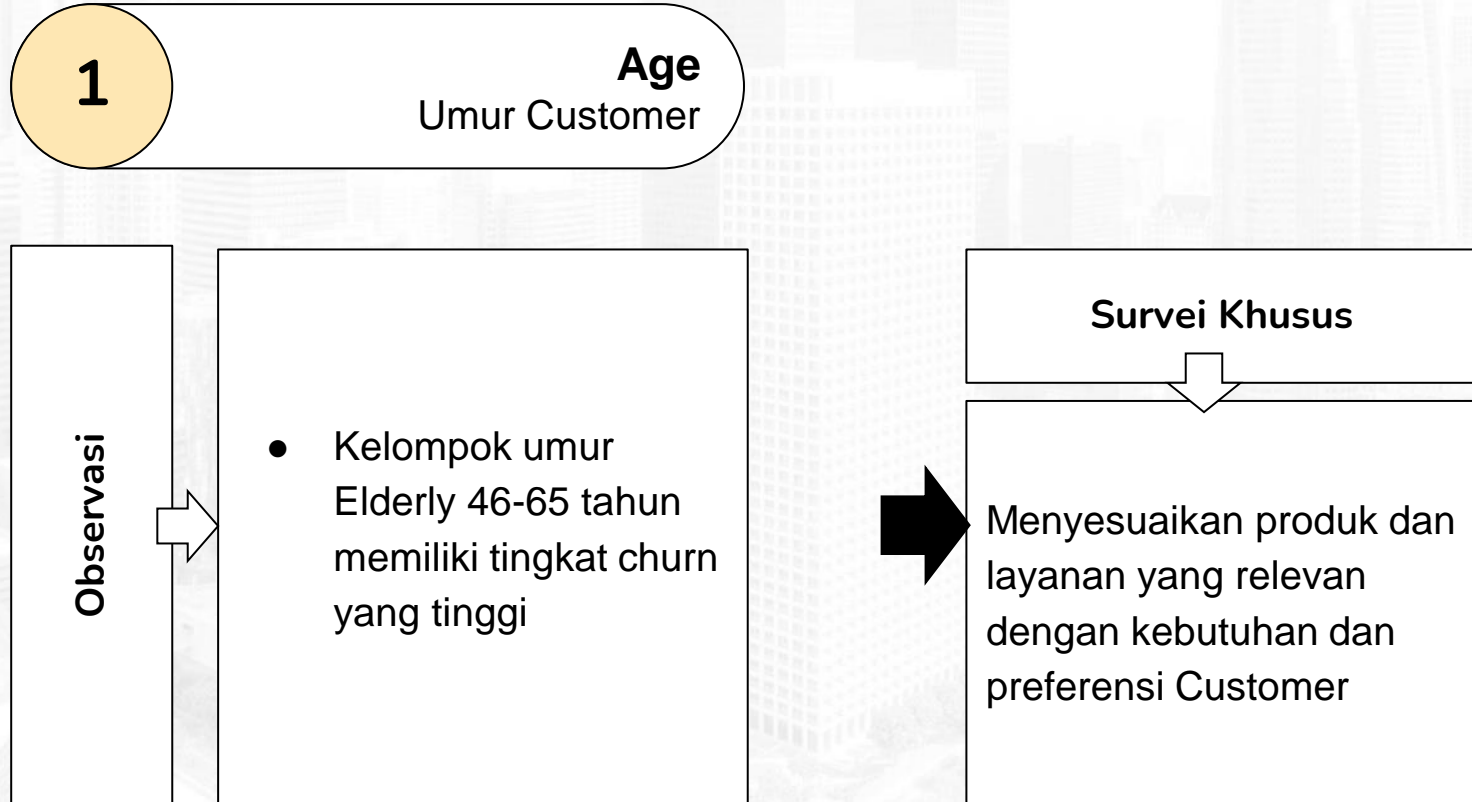
Model terbaik adalah dengan menggunakan AdaBoostClassifier

Alasan : setelah dibandingkan dengan GradientBoosting, Gradient Boosting memiliki recall yang tinggi namun, recall 0 bernilai 0%. Artinya bahwa gradient boosting memprediksi banyak nasabah yang churn walaupun sebenarnya nasabah tersebut tidak churn

5. Executive Summary & Recommendation



5.1 Actionable Insights

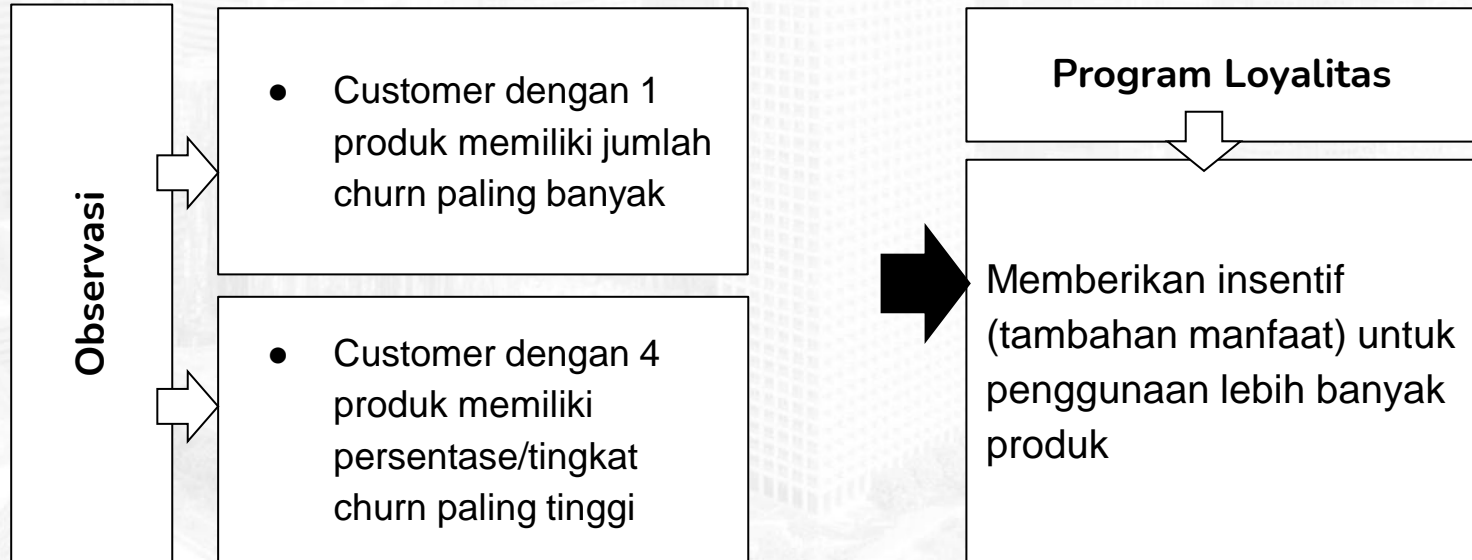


5.1 Actionable Insights

2

NumOfProducts

Jumlah Produk yang digunakan Customer

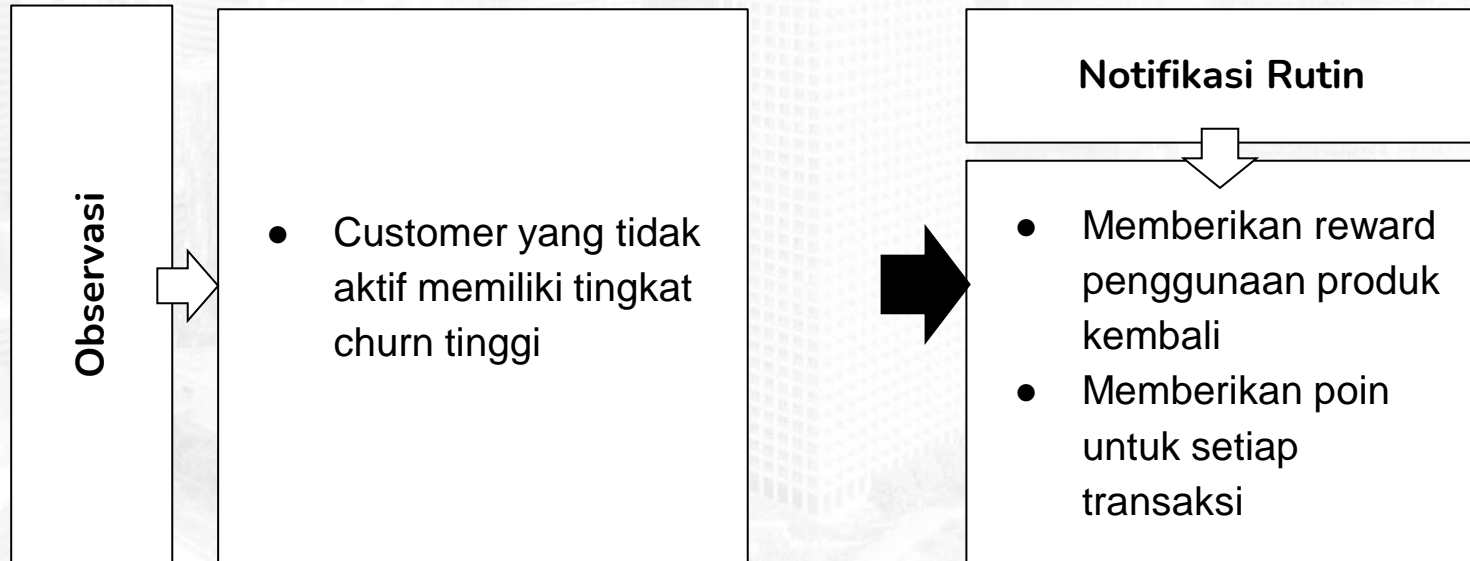


5.1 Actionable Insights

3

IsActiveMember

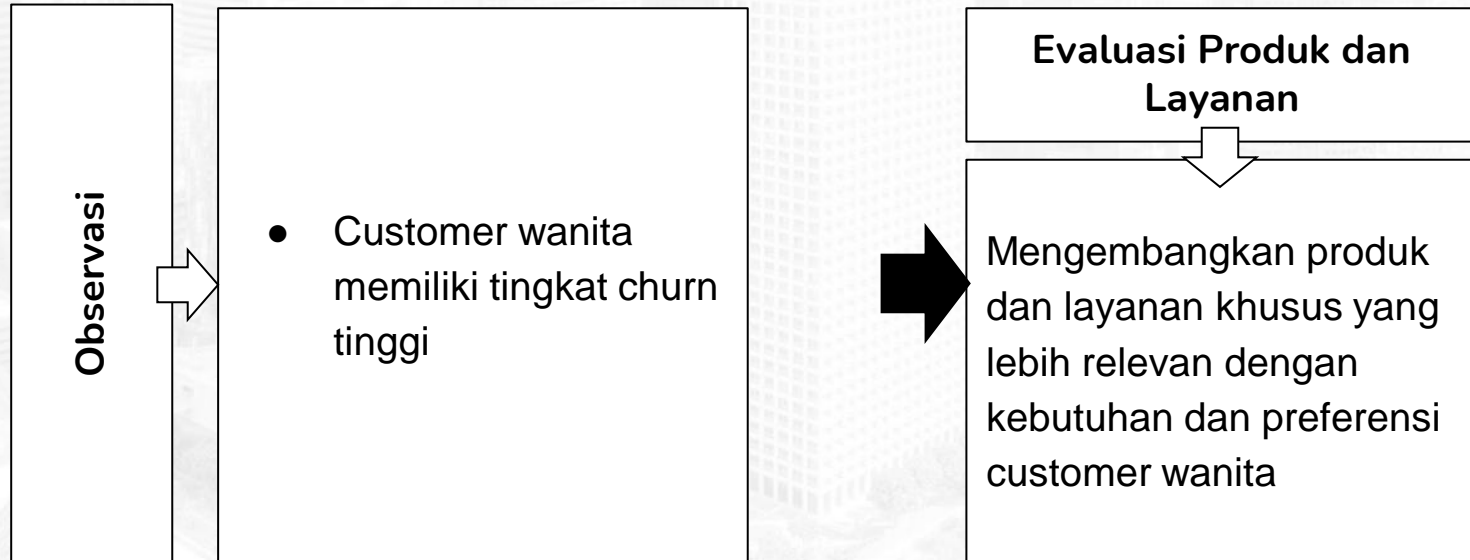
Nasabah yang aktif menggunakan produk dan layanan

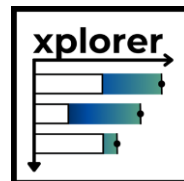


5.1 Actionable Insights

4

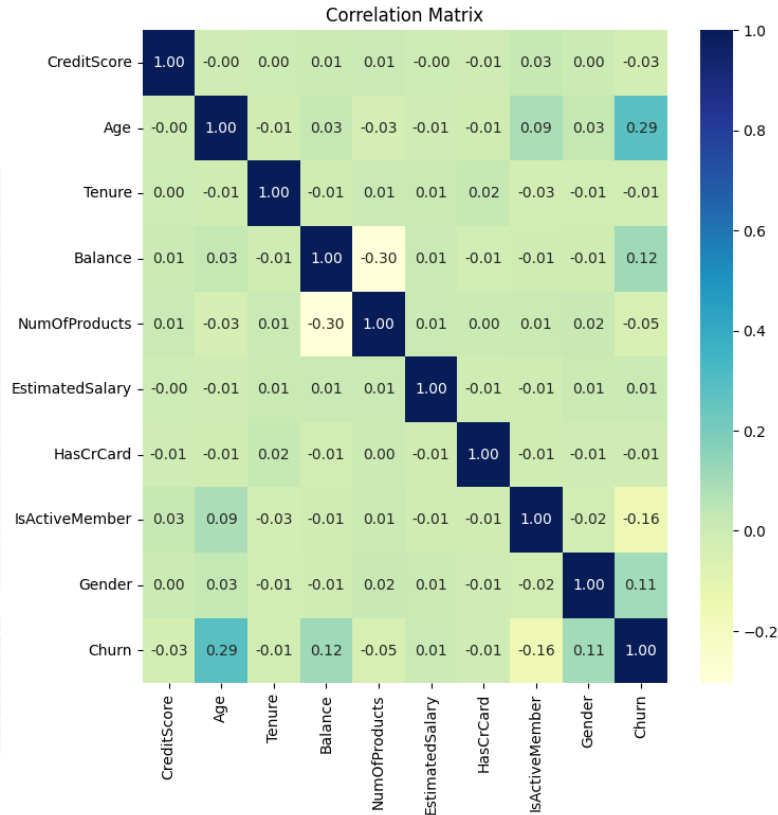
Gender
Jenis Kelamin Customer





TERIMA KASIH!

2.1 Exploratory Data Analysis (EDA)



Multivariate Analysis

- Feature to Feature

- 'Balance' dan 'NumOfProducts' berkorelasi negative (korelasi = -0.30)

- Feature to Label ("Churn")

- Korelasi positive:
 - 'Age' (korelasi = 0.29)
 - 'Balance' (korelasi = 0.12)
 - 'Gender' (korelasi = 0.11)
- Korelasi negative:
 - 'IsActiveMember' (korelasi = -0.16)

| | data | list_method | prec_train | prec_test | recall_train | recall_test | AUC_train | AUC_test |
|----|-------------|----------------------------|------------|-----------|--------------|-------------|-----------|----------|
| 0 | df9_x_train | AdaBoostClassifier | 90.254 | 43.801 | 90.418 | 82.697 | 96.084 | 84.729 |
| 1 | df7_x_train | LogisticRegression | 80.729 | 32.242 | 84.405 | 80.153 | 85.866 | 77.819 |
| 2 | df9_x_train | GradientBoostingClassifier | 93.615 | 46.244 | 93.582 | 79.898 | 98.049 | 86.166 |
| 3 | df9_x_train | LogisticRegression | 80.465 | 33.018 | 82.661 | 79.898 | 85.592 | 77.877 |
| 4 | df6_x_train | GradientBoostingClassifier | 83.119 | 50.000 | 78.771 | 79.389 | 89.995 | 86.823 |
| 5 | df4_x_train | GradientBoostingClassifier | 83.119 | 50.000 | 78.771 | 79.389 | 89.995 | 86.823 |
| 6 | df7_x_train | GradientBoostingClassifier | 92.900 | 46.837 | 93.834 | 79.135 | 97.818 | 86.334 |
| 7 | df7_x_train | AdaBoostClassifier | 90.315 | 44.556 | 91.292 | 79.135 | 95.958 | 84.638 |
| 8 | df6_x_train | AdaBoostClassifier | 77.968 | 46.341 | 74.696 | 77.354 | 85.885 | 84.409 |
| 9 | df4_x_train | AdaBoostClassifier | 77.968 | 46.341 | 74.696 | 77.354 | 85.885 | 84.409 |
| 10 | df6_x_train | XGBClassifier | 98.960 | 45.706 | 98.358 | 75.827 | 99.898 | 84.625 |
| 11 | df4_x_train | XGBClassifier | 98.960 | 45.706 | 98.358 | 75.827 | 99.898 | 84.625 |
| 12 | df9_x_train | XGBClassifier | 99.764 | 52.788 | 99.530 | 72.265 | 99.992 | 85.552 |
| 13 | df3_x_train | AdaBoostClassifier | 83.674 | 50.536 | 84.346 | 72.010 | 92.116 | 84.278 |
| 14 | df6_x_train | LogisticRegression | 70.900 | 38.897 | 69.951 | 71.756 | 77.251 | 78.110 |
| 15 | df7_x_train | XGBClassifier | 99.677 | 53.935 | 99.469 | 71.501 | 99.989 | 85.433 |
| 16 | df3_x_train | LogisticRegression | 71.312 | 38.943 | 70.359 | 71.247 | 77.590 | 78.121 |
| 17 | df8_x_train | LogisticRegression | 71.637 | 38.997 | 70.676 | 71.247 | 77.945 | 78.128 |

Fitur Extraction

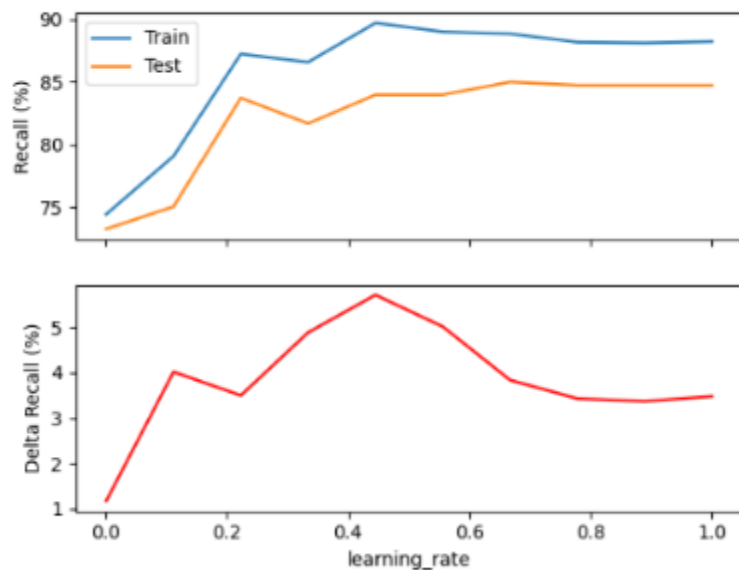
| | data | list_method | prec_train | prec_test | recall_train | recall_test | AUC_train | AUC_test |
|----|--------------|----------------------------|------------|-----------|--------------|-------------|-----------|----------|
| 0 | df9_x_train | AdaBoostClassifier | 88.704 | 40.293 | 90.436 | 77.099 | 95.300 | 82.292 |
| 1 | df7_x_train | LogisticRegression | 78.450 | 30.101 | 80.250 | 75.827 | 81.761 | 73.467 |
| 2 | df9_x_train | LogisticRegression | 77.857 | 30.512 | 78.828 | 74.300 | 80.991 | 73.457 |
| 3 | df7_x_train | AdaBoostClassifier | 91.194 | 45.928 | 90.969 | 71.756 | 95.894 | 82.361 |
| 4 | df6_x_train | AdaBoostClassifier | 76.451 | 43.012 | 71.290 | 70.483 | 82.916 | 81.235 |
| 5 | df4_x_train | AdaBoostClassifier | 76.451 | 43.012 | 71.290 | 70.483 | 82.916 | 81.235 |
| 6 | df9_x_train | GradientBoostingClassifier | 94.299 | 49.104 | 92.406 | 69.720 | 97.565 | 82.484 |
| 7 | df6_x_train | XGBClassifier | 92.467 | 39.000 | 89.599 | 69.466 | 97.604 | 78.788 |
| 8 | df4_x_train | XGBClassifier | 92.467 | 39.000 | 89.599 | 69.466 | 97.604 | 78.788 |
| 9 | df6_x_train | GradientBoostingClassifier | 80.000 | 45.777 | 73.236 | 68.957 | 86.164 | 82.025 |
| 10 | df4_x_train | GradientBoostingClassifier | 80.000 | 45.700 | 73.236 | 68.957 | 86.164 | 82.025 |
| 11 | df7_x_train | GradientBoostingClassifier | 94.155 | 50.482 | 92.297 | 66.667 | 97.670 | 82.642 |
| 12 | df4_x_train | DecisionTreeClassifier | 100.000 | 32.952 | 99.392 | 65.903 | 99.998 | 66.502 |
| 13 | df6_x_train | DecisionTreeClassifier | 100.000 | 32.700 | 99.392 | 65.649 | 99.998 | 66.239 |
| 14 | df4_x_train | LogisticRegression | 70.968 | 35.989 | 64.234 | 64.377 | 73.786 | 73.756 |
| 15 | df9_x_train | DecisionTreeClassifier | 100.000 | 41.653 | 99.873 | 64.122 | 100.000 | 71.067 |
| 16 | df5_x_train | LogisticRegression | 70.475 | 35.806 | 65.308 | 63.868 | 73.457 | 73.768 |
| 17 | df10_x_train | LogisticRegression | 70.649 | 35.857 | 65.524 | 63.868 | 73.638 | 73.761 |

Fitur Tambahan

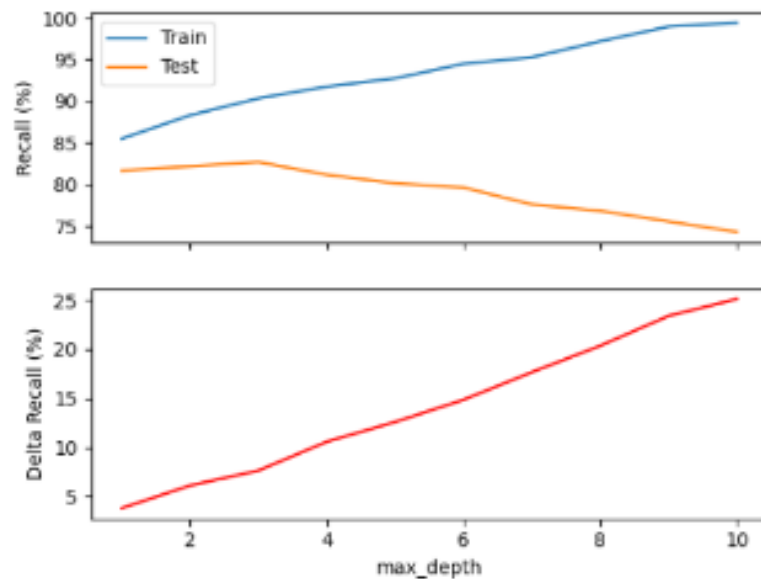
| | data | list_method | acc_train | acc_test | prec_train | prec_test | recall_train | recall_test | AUC_train | AUC_test |
|----|--------------|----------------------------|-----------|----------|------------|-----------|--------------|-------------|-----------|----------|
| 0 | df9_x_train | AdaBoostClassifier | 88.965 | 75.35 | 90.396 | 43.316 | 90.363 | 82.443 | 96.187 | 85.097 |
| 1 | df9_x_train | GradientBoostingClassifier | 92.709 | 77.50 | 93.716 | 45.911 | 93.564 | 81.425 | 98.051 | 86.315 |
| 2 | df7_x_train | LogisticRegression | 78.634 | 61.75 | 79.986 | 31.511 | 84.462 | 80.662 | 86.044 | 77.851 |
| 3 | df9_x_train | LogisticRegression | 78.117 | 62.55 | 79.681 | 31.874 | 83.023 | 79.644 | 85.847 | 77.941 |
| 4 | df6_x_train | GradientBoostingClassifier | 81.569 | 80.20 | 83.614 | 49.761 | 78.528 | 79.389 | 90.291 | 86.478 |
| 5 | df7_x_train | GradientBoostingClassifier | 92.141 | 78.10 | 92.922 | 46.637 | 93.644 | 79.389 | 97.758 | 86.211 |
| 6 | df4_x_train | GradientBoostingClassifier | 81.569 | 80.25 | 83.614 | 49.840 | 78.528 | 79.389 | 90.291 | 86.481 |
| 7 | df7_x_train | AdaBoostClassifier | 89.223 | 76.55 | 90.315 | 44.556 | 91.292 | 79.135 | 95.958 | 84.638 |
| 8 | df6_x_train | AdaBoostClassifier | 77.342 | 77.50 | 78.324 | 45.714 | 75.608 | 77.354 | 85.982 | 84.381 |
| 9 | df4_x_train | AdaBoostClassifier | 77.342 | 77.50 | 78.324 | 45.714 | 75.608 | 77.354 | 85.982 | 84.381 |
| 10 | df6_x_train | XGBClassifier | 99.179 | 75.50 | 99.209 | 42.836 | 99.148 | 73.791 | 99.963 | 84.476 |
| 11 | df4_x_train | XGBClassifier | 99.179 | 75.50 | 99.209 | 42.836 | 99.148 | 73.791 | 99.963 | 84.476 |
| 12 | df5_x_train | LogisticRegression | 70.839 | 71.35 | 70.380 | 38.064 | 71.963 | 73.028 | 77.885 | 78.216 |
| 13 | df4_x_train | LogisticRegression | 70.620 | 71.50 | 70.348 | 38.184 | 71.290 | 72.774 | 77.254 | 78.132 |
| 14 | df6_x_train | LogisticRegression | 70.803 | 71.05 | 70.652 | 37.731 | 71.168 | 72.774 | 77.308 | 78.168 |
| 15 | df10_x_train | LogisticRegression | 71.000 | 71.50 | 70.550 | 38.184 | 72.095 | 72.774 | 78.109 | 78.231 |
| 16 | df9_x_train | XGBClassifier | 99.751 | 81.80 | 99.873 | 52.690 | 99.693 | 72.265 | 99.997 | 85.567 |

Sampel Learning Curve

Learning Curve - list_learning - AdaBoostClassifier

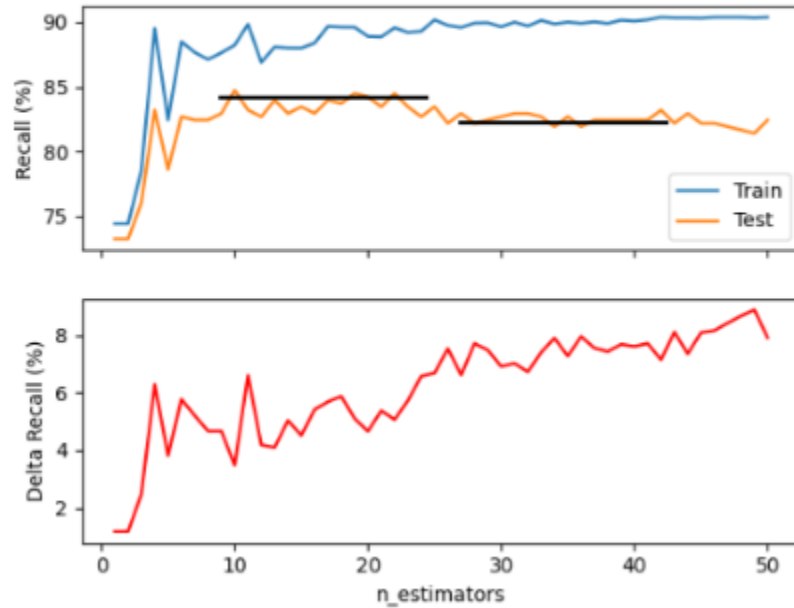


Learning Curve - max_depth - GradientBoostingClassifier

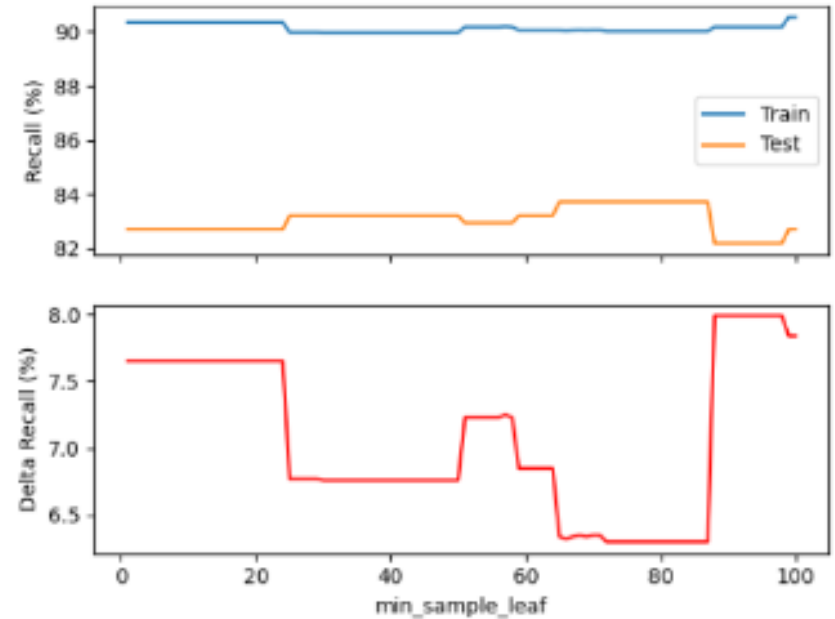


SampeL Learning Curve

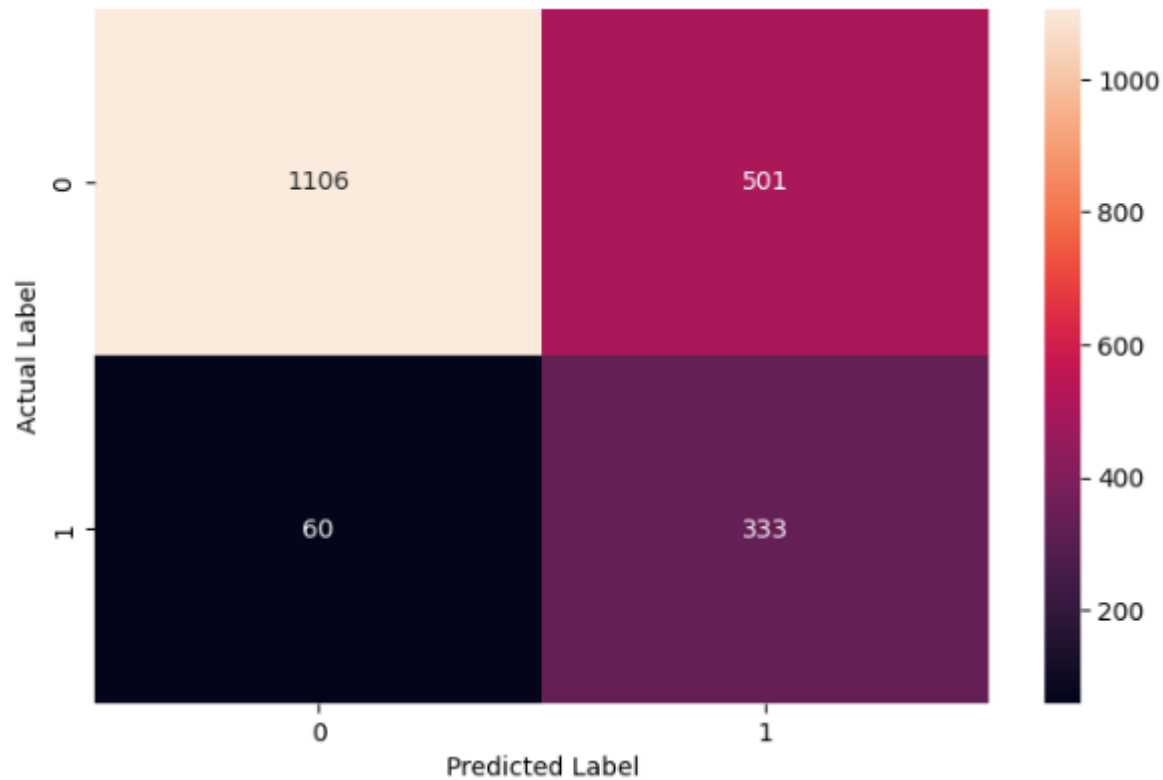
Learning Curve - n_estimators - AdaBoostClassifier



Learning Curve - min_sample_leaf - GradientBoostingClassifier



Confusion Matrix



5. Executive Summary & Recommendation

