Progress Report FinPro Stage 2

Dino Kuning

- Fany Okpiani
- Nadhilah Farhana
- Raditya Satria Gantara
- Rafindra Prihaztama

Dino Kuning's Group Member

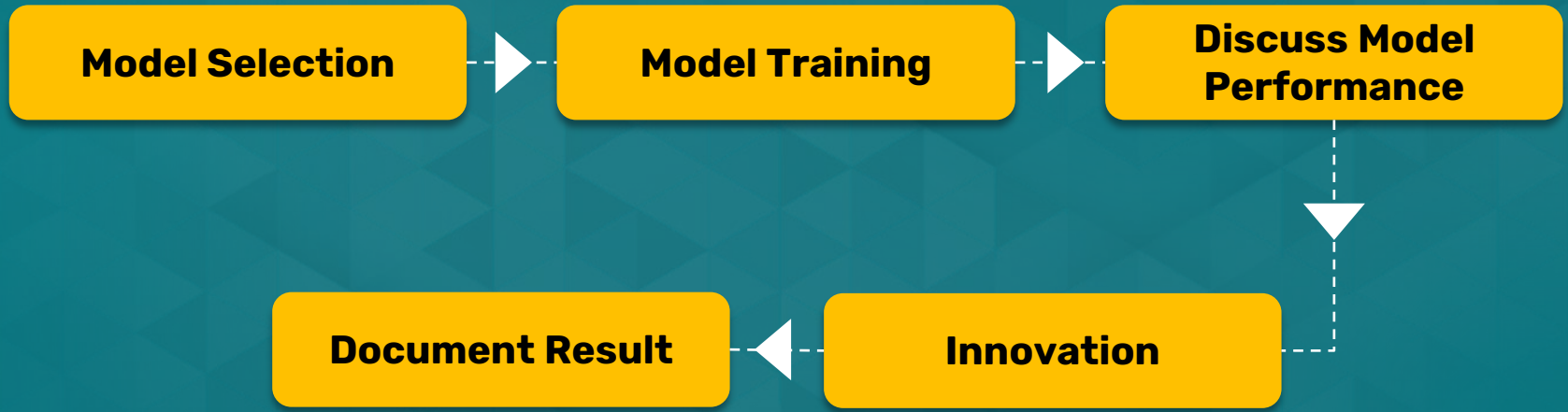
Fany Okpiani
Business / Data Analyst
Rakamin Academy

Nadhilah Farhana
Data Scientist
Rakamin Academy

Rafindra Prihaztama
Data Engineer
Rakamin Academy

Raditya Satria G.
Project Manager
Rakamin Academy

Outline - Stage 2



Model Selection

XGBoost (Extreme Gradient Boosting)	LightGBM (Light Gradient Boosting Machine)	Random Forest	Regresi Logistik
Algoritma boosting yang menggunakan pendekatan ensemble untuk meningkatkan performa model. Algoritma ini menggabungkan beberapa model decision tree yang dibangun secara bertahap, dengan setiap model baru berfokus pada kesalahan yang dibuat oleh model sebelumnya.	Varian dari gradient boosting yang didesain untuk efisiensi dan kecepatan. LightGBM menggunakan leaf-wise growth (berbeda dari level-wise pada XGBoost) yang cenderung lebih efisien dalam membangun pohon keputusan.	Algoritma ensemble yang menggunakan sejumlah besar decision tree untuk membuat prediksi. Setiap pohon dihasilkan dengan memilih subset acak dari fitur dan data, lalu hasilnya digabungkan untuk mendapatkan prediksi akhir.	model linear yang digunakan untuk prediksi probabilistik dalam kasus klasifikasi biner. Model ini memodelkan hubungan antara fitur dan probabilitas kelas target menggunakan fungsi logistik.
Kelebihan: Sangat efektif untuk dataset besar dan kompleks. Dikenal karena performa yang tinggi, kemampuan untuk menangani missing values, dan fleksibilitasnya dalam hyperparameter tuning.	**Kelebihan**: Lebih cepat dan efisien dalam hal memori dibandingkan XGBoost, dan sangat cocok untuk data besar dan tinggi dimensi. Menghasilkan model yang lebih ringan dan memiliki kecepatan pelatihan yang lebih baik.	**Kelebihan**: Sangat baik dalam menangani data yang bising dan kompleks, serta mudah diinterpretasikan. Tidak rentan terhadap overfitting meskipun memiliki banyak pohon.	**Kelebihan**: Sederhana, cepat, dan mudah dipahami. Cocok digunakan pada masalah klasifikasi dengan hubungan linier antara fitur dan target. Namun, tidak seefektif model non-linier seperti XGBoost atau Random Forest pada data yang lebih kompleks.

Ensemble Model: Stacking

Ensemble Stacking adalah teknik ensemble yang menggabungkan beberapa model (disebut *base models*) dan menggabungkannya dengan model lain (disebut *meta-model* atau *final estimator*) untuk membuat prediksi yang lebih akurat. Pada teknik stacking, model-model dasar belajar secara independen, dan hasil dari model-model tersebut digunakan sebagai input untuk model meta (final estimator) yang akan membuat prediksi akhir.

1. **Base Models (Model Dasar)**:
   Pada penelitiaan ini, dua model dasar dipilih:
   - **RandomForestClassifier** (rf): Model pohon keputusan berbasis *ensemble*.
   - **XGBClassifier** (xgb): Model gradient boosting yang kuat dan efisien.

2. **Meta-Model (Model Meta)**:
   LogisticRegression: Model regresi logistik dipilih sebagai model meta yang bertugas untuk membuat prediksi akhir berdasarkan prediksi yang diberikan oleh model dasar.

3. **StackingClassifier**:
   StackingClassifier menggabungkan model-model dasar tersebut. Pada penelitian ini dilatih menggunakan teknik *cross-validation* (cv=5) untuk meningkatkan kestabilan model.

Rakamin Academy

Model Training: Alur Analisis

1 Data Cleaning & Preprocessing

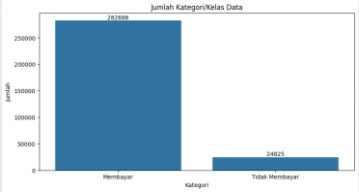
- Menghapus kolom dengan missing values >10%
- Feature Engineering:
  - Ubah satuan data negatif menjadi positif (DAYS_BIRTH)
  - Menambahkan kolom DAYS_EMPLOYED_ANOM
- Encoding data kategorik

2 Train-Test Split

80% data train,
20% data test

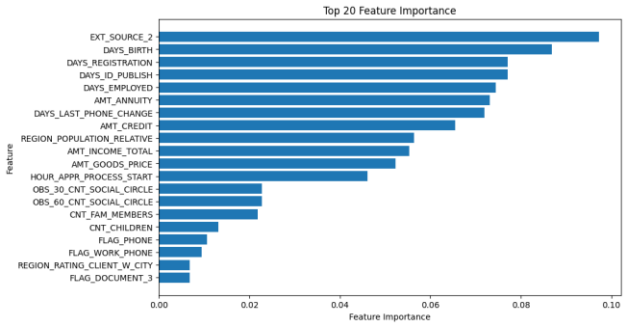
3 Oversampling

Hanya diterapkan untuk data train



4 Feature Selection

Memilih Top 20 berdasarkan hasil feature importance



5 Scaling

Data train: fit_transform,
Data test: transform

6 Training Model

XGBoost, Random Forest, Voting Ensemble

7 Testing Model & Evaluasi

Accuracy, ROC-AUC, Precision, Recall

Model Training

```
Accuracy (XGBoost): 0.9104759117441426
ROC-AUC (XGBoost): 0.5242995440003579
Recall (XGBoost): 0.0640533441099212
Precision (XGBoost): 0.2661628883291352
              precision    recall  f1-score   support

           0       0.92      0.98      0.95     56554
           1       0.27      0.06      0.10      4949

    accuracy                           0.91     61503
   macro avg       0.59      0.52      0.53     61503
weighted avg       0.87      0.91      0.88     61503

Confusion Matrix (XGBoost):
 [[55680   874]
 [ 4632   317]]
```

```
Accuracy (LightGBM): 0.9143456416760158
ROC-AUC (LightGBM): 0.514050347530024
Recall (LightGBM): 0.036977167104465546
Precision (LightGBM): 0.2671532846715328
              precision    recall  f1-score   support

           0       0.92      0.99      0.96     56554
           1       0.27      0.04      0.06      4949

    accuracy                           0.91     61503
   macro avg       0.59      0.51      0.51     61503
weighted avg       0.87      0.91      0.88     61503

Confusion Matrix (LightGBM):
 [[56052   502]
 [ 4766   183]]
```

```
Accuracy (Random Forest): 0.8907045184787734
ROC-AUC (Random Forest): 0.530603796807859
Recall (Random Forest): 0.10143463325924429
Precision (Random Forest): 0.18077061577241627
              precision    recall  f1-score   support

           0       0.92      0.96      0.94     56554
           1       0.18      0.10      0.13      4949

    accuracy                           0.89     61503
   macro avg       0.55      0.53      0.54     61503
weighted avg       0.86      0.89      0.88     61503

Confusion Matrix (Random Forest):
 [[54279  2275]
 [ 4447   502]]
```

```
Accuracy (Logistic Regression): 0.6811700242264604
ROC-AUC (Logistic Regression): 0.6575593545946422
Recall (Logistic Regression): 0.6294200848656294
Precision (Logistic Regression): 0.14911440880804214
              precision    recall  f1-score   support

           0       0.95      0.69      0.80     56554
           1       0.15      0.63      0.24      4949

    accuracy                           0.68     61503
   macro avg       0.55      0.66      0.52     61503
weighted avg       0.89      0.68      0.75     61503

Confusion Matrix (Logistic Regression):
 [[38779 17775]
 [ 1834  3115]]
```

Model Training

```
Accuracy (Stacking): 0.8989317594263695
ROC-AUC (Stacking): 0.6848814658821533
Recall (Stacking): 0.09315013133966459
Precision (Stacking): 0.2105984467793513
              precision    recall  f1-score   support

           0       0.92      0.97      0.95     56554
           1       0.21      0.09      0.13      4949

    accuracy                           0.90     61503
   macro avg       0.57      0.53      0.54     61503
weighted avg       0.87      0.90      0.88     61503

Confusion Matrix (Stacking):
 [[54826  1728]
 [ 4488   461]]
```

Rangkuman hasil evaluasi dari seluruh model

Model	Akurasi	ROC-AUC	Recall (0)	Precision (0)
XGBoost	**91%**	52%	98%	92%
LightGBM	**91%**	51%	**99%**	92%
Random Forest	89%	53%	96%	92%
Regresi Logistik	68%	66%	69%	**95%**
Ensemble Stacking	90%	**68%**	97%	92%

Model Training

- **Akurasinya Cukup Baik untuk Semua Model**:

  Semua model yang diuji (XGBoost, LightGBM, Random Forest, Logistic Regression, dan Stacking) memiliki akurasi yang relatif tinggi, berkisar antara 0.89 hingga 0.91. Ini menunjukkan bahwa model-model tersebut cukup baik dalam mengklasifikasikan nasabah potensial secara umum, meskipun ada perbedaan pada prediksi masing-masing kelas.

- **Model XGBoost dan LightGBM Memiliki Performa Terbaik**:
  - **XGBoost** dan **LightGBM** menunjukkan hasil yang sangat baik pada kelas 0 (nasabah potensial), dengan recall yang tinggi (0.98 dan 0.99, masing-masing) dan nilai precision yang lebih baik dibandingkan dengan model lain.
  - **Precision** untuk kelas 0 pada kedua model ini sangat tinggi (XGBoost: 0.92, LightGBM: 0.92), yang berarti model ini lebih cenderung memprediksi nasabah potensial dengan benar.

- **Stacking Memberikan Performa yang Baik di Kelas 0**:

  Stacking menghasilkan hasil yang sangat mirip dengan model XGBoost dan LightGBM, dengan akurasi 0.90 dan nilai **recall** untuk kelas 0 yang relatif tinggi (0.97). Precision dan F1-score untuk kelas 0 juga cukup baik (Precision: 0.92, F1-score: 0.95), menunjukkan model ini cukup efektif dalam mengidentifikasi nasabah potensial (kelas 0).

Dashboard Customer Risk

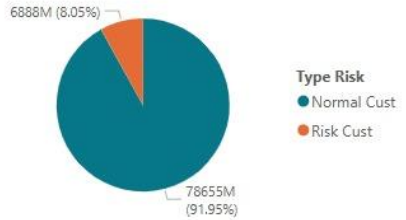
Number of Customer	Number of Credit	Number of Income
307.507K	184.21bn	51.91bn

Type Risk
- Normal Cust
- Risk Cust

Gender
- F
- M

Own Car
- N
- Y

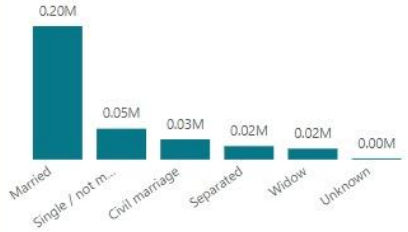
Customer by Risk Type



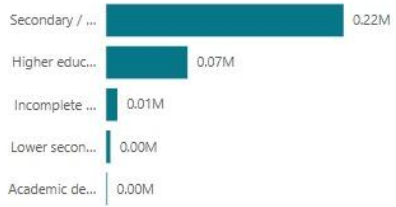
Customer by Payment Type

Customer by Housing Type

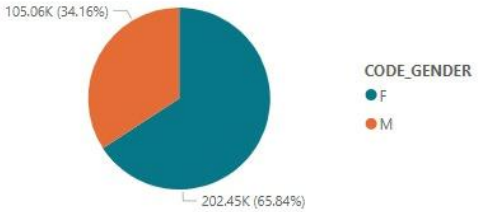
Customer by Status



Customer by Education



Customer by Gender



Dokumentasi

Dokumentasi github:

https://github.com/farhanadhilah/homecredit-analysis/tree/main


Thank you!
Raih Mimpi #TanpaBatas