Prediksi Kelayakan Pinjaman Berdasarkan Profil Risiko Nasabah Menggunakan Logistic Regression dan Random Forest

Hafidh Ismu Azam[1], Anita Qoiriah[2]

[1,2] Program Studi Teknik Informatika, Fakultas Teknik, Universitas Negeri Surabaya
[1]hafidh.21027@mhs.unesa.ac.id
[2]anitaqoiriah@unesa.ac.id

***Abstrak*— Penilaian kelayakan pinjaman menjadi faktor krusial dalam menjaga kualitas pembiayaan, terutama bagi lembaga seperti PNM Mekaar yang menyasar kelompok perempuan prasejahtera sebagai target utama. Penelitian ini mengembangkan sistem prediksi kelayakan pinjaman berdasarkan profil risiko nasabah dengan memanfaatkan dua pendekatan algoritmik, yaitu *Logistic Regression* dan *Random Forest*. Model *Logistic Regression* digunakan untuk menilai kelayakan awal berdasarkan variabel usia, jenis kelamin, pendapatan mingguan, jumlah pinjaman, serta persetujuan kelompok dan penanggung jawab. Sementara itu, *Random Forest* diterapkan untuk memprediksi kelayakan top-up pinjaman dengan mempertimbangkan histori keterlambatan pembayaran dan durasi peminjaman, serta dilengkapi teknik SMOTE untuk menangani ketidakseimbangan kelas pada data pelatihan. Hasil menunjukkan bahwa *Logistic Regression* mampu memisahkan nasabah layak tanpa perlu penerapan oversampling, sedangkan *Random Forest* efektif dalam mengklasifikasikan risiko nasabah ke dalam tiga kategori, yaitu rendah, sedang, dan tinggi. Evaluasi dengan k-fold *cross-validation* menunjukkan bahwa kedua model memiliki performa yang andal dan stabil dalam melakukan klasifikasi. Temuan ini diharapkan dapat mendukung pengambilan keputusan yang lebih akurat dan berbasis data dalam proses pembiayaan, serta meningkatkan ketepatan dalam penilaian risiko nasabah secara menyeluruh.**

***Kata Kunci*— kelayakan pinjaman, profil risiko, *Logistic Regression*, *Random Forest*, SMOTE, PNM Mekaar.**

I. PENDAHULUAN

Industri keuangan memegang peran strategis dalam mendorong pertumbuhan ekonomi, salah satunya melalui penyediaan pinjaman kepada individu maupun pelaku usaha [1]. Namun, aktivitas pemberian pinjaman tidak terlepas dari risiko kredit yang berpotensi mengganggu stabilitas lembaga keuangan, terutama jika proses penetapan batas pinjaman dilakukan tanpa analisis risiko yang matang. Dalam praktiknya, lembaga keuangan harus mampu mengelola berbagai jenis risiko, termasuk risiko kredit, likuiditas, dan operasional, guna menjaga keberlanjutan bisnis [2].

Permodalan Nasional Madani (PNM) merupakan salah satu lembaga keuangan non-bank yang berfokus pada pemberdayaan pelaku usaha mikro, terutama melalui program PNM Mekaar. Program ini menyasar perempuan pra-sejahtera yang menjalankan usaha ultra mikro dalam bentuk kelompok, tanpa mensyaratkan agunan [3]. Meski mendukung inklusi keuangan, pendekatan ini juga menghadirkan tantangan tersendiri dalam menilai kelayakan pinjaman, mengingat keterbatasan jaminan serta ketidakpastian dalam kemampuan membayar yang dimiliki oleh segmen nasabah ini [4]. Oleh karena itu, dibutuhkan pendekatan yang lebih cermat dan sistematis untuk mengurangi risiko gagal bayar dan menjaga keberlanjutan program pembiayaan.

Masalah utama dalam konteks ini adalah bagaimana memprediksi kelayakan pinjaman dan potensi risiko kredit dari nasabah PNM Mekaar berdasarkan karakteristik dan riwayat pembayaran mereka. Faktor-faktor seperti usia, jenis kelamin, jenis usaha, persetujuan kelompok, dan persetujuan penanggung jawab menjadi elemen penting dalam proses pengambilan keputusan, terutama karena sistem pembiayaan yang dijalankan tidak bergantung pada jaminan fisik [5]. Risiko kredit dapat timbul karena berbagai penyebab, seperti keterlambatan pembayaran, kegagalan usaha, atau nasabah yang menghilang tanpa pelunasan, yang seluruhnya dapat berdampak pada stabilitas operasional [6].

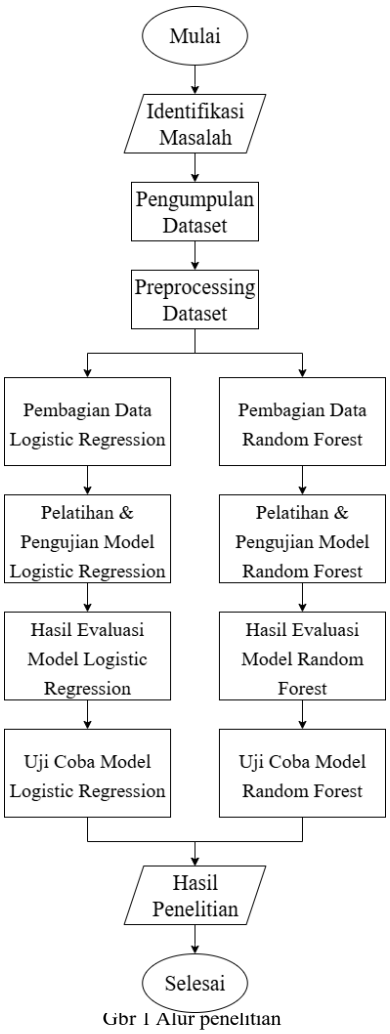
Seiring perkembangan teknologi informasi, proses prediksi kini tidak lagi sepenuhnya bergantung pada intuisi manusia. Sistem komputer telah berkembang menjadi alat bantu yang mampu mengolah data dan menyajikan keputusan prediktif secara lebih cepat dan akurat [7]. Sistem prediksi berbasis data dapat digunakan untuk menganalisis berbagai variabel risiko dan menghasilkan estimasi yang lebih objektif.

Dalam penelitian ini, digunakan dua pendekatan algoritma, yaitu Logistic Regression dan Random Forest. Logistic Regression diterapkan untuk mengklasifikasikan nasabah ke dalam kategori layak atau tidak layak menerima pinjaman awal, karena metode ini dapat menjelaskan hubungan antara satu atau lebih variabel independen terhadap variabel dependen yang

bersifat biner [8]. Sementara itu, Random Forest digunakan untuk mengidentifikasi tingkat risiko nasabah yang mengajukan pinjaman tambahan (top-up) setelah satu tahun pembiayaan [9]. Teknik ini mengombinasikan banyak pohon keputusan dan menggunakan seleksi fitur secara acak untuk meningkatkan akurasi klasifikasi [10].

Penelitian ini berkontribusi dalam penerapan metode pembelajaran mesin pada sistem pembiayaan berbasis kelompok, yang masih relatif jarang diteliti secara mendalam. Dengan pendekatan ini, diharapkan proses penilaian kelayakan pinjaman dan identifikasi risiko dapat dilakukan secara lebih efisien, obyektif, dan tepat sasaran, sehingga mampu memperkuat kualitas pembiayaan dan mendukung pemberdayaan pelaku usaha mikro secara berkelanjutan.

II. METODE PENELITIAN



Gbr 1 Alur penelitian

Penelitian ini merupakan penelitian kuantitatif yang bertujuan untuk mengembangkan model prediksi kelayakan pinjaman menggunakan *Logistic Regression* dan *Random Forest*. Fokus utama penelitian ini adalah untuk memprediksi kelayakan pinjaman nasabah pada program PNM Mekaar berdasarkan profil risiko nasabah. Alur dari penelitian ini dapat dilihat Gbr 1.

A. *Pengumpulan Dataset*

Dataset yang digunakan dalam penelitian ini berasal dari data historis nasabah PNM Mekaar, yang mencakup informasi profil, status kelayakan pinjaman, serta riwayat pembiayaan. Data dikumpulkan melalui wawancara langsung dengan nasabah di lokasi program untuk memastikan akurasi dan relevansi dengan kondisi nyata. Pengambilan sampel dilakukan secara acak *(random sampling)* dengan kriteria: (1) nasabah aktif atau pernah menerima pinjaman dari PNM Mekaar, dan (2) memiliki riwayat pembiayaan sebelumnya.

Data yang dihimpun meliputi sejumlah variabel, antara lain: usia, jenis kelamin, jenis usaha, pendapatan mingguan, jumlah pinjaman, jumlah keterlambatan, minggu meminjam, serta persetujuan dari kelompok dan penanggung jawab. Pengumpulan dilakukan dalam dua tahap. Tahap pertama digunakan untuk membangun model *Logistic Regression* dalam memprediksi kelayakan pinjaman awal (layak atau tidak layak), dengan menggunakan enam variabel utama. Tahap kedua ditujukan untuk model *Random Forest*, yang memprediksi tingkat risiko pemberian top-up pinjaman (risiko rendah, sedang, atau tinggi) dengan delapan variabel, termasuk riwayat keterlambatan pembayaran.

Dataset yang digunakan telah disiapkan dalam format CSV dan melalui proses penyaringan untuk memastikan kelengkapan data serta konsistensi dalam pelabelan target. Label ditetapkan berdasarkan penilaian pihak lapangan, dan apabila ditemukan perbedaan pada data yang serupa, maka penentuan label akhir dilakukan dengan mempertimbangkan nilai modus atau rata-rata yang paling merepresentasikan kondisi nasabah. Struktur dataset secara umum, yang mencakup nama kolom, tipe data, dan deskripsi singkat, disajikan pada Tabel I.

TABEL I
STRUKTUR DATASET NASABAH PNM MEKAAR

Nama kolom	Tipe data	Keterangan
usia	Numerik	Usia nasabah
jenis_kelamin	Kategorikal	Jenis kelamin nasabah
persetujuan_ kelompok	Kategorikal	Persetujuan oleh kelompok
persetujuan_ penanggungjawab	Kategorikal	Persetujuan penanggung jawab (suami)
pendapatan_ mingguan	Numerik	Pendapatan nasabah per minggu
jumlah_ pinjaman	Numerik	Pinjaman yang diberikan nasabah

jumlah_ keterlambatan	Numerik	Jumlah minggu keterlambatan
minggu_ meminjam	Numerik	Lama pinjaman sudah berjalan (minggu)
kelayakan_ pinjaman	Kategorikal	Label target: layak atau tidak layak
risiko_ nasabah	Kategorikal	Label risiko: risiko rendah, sedang, tinggi

B. *Preprocessing Dataset*

Dataset yang telah dikumpulkan dari data profil nasabah sebelumnya perlu melalui proses preprocessing untuk memastikan akurasi model yang optimal. Proses preprocessing yang dilakukan meliputi pembersihan data, pengisian data hilang, encoding variabel kategorikal, serta normalisasi data. Dataset hasil preprocessing kemudian digunakan untuk membangun dan menguji model *Logistic Regression* dan *Random Forest*.

1) *Pembersihan Data*

Tahap awal dalam preprocessing adalah pembersihan data untuk memastikan kualitas dan konsistensi informasi yang digunakan dalam pelatihan model. Pada tahap ini, data duplikat dan entri yang tidak relevan dihapus, serta diperiksa adanya kesalahan atau inkonsistensi pada variabel yang digunakan. Proses ini penting agar model tidak dipengaruhi oleh data yang tidak akurat atau bias.

2) *Pengisian Data Hilang*

Langkah selanjutnya adalah menangani data yang hilang, yang umumnya terjadi akibat kesalahan input atau format yang tidak sesuai. Untuk menjaga kualitas data, dilakukan imputasi dengan pendekatan yang sesuai. Nilai median atau rata-rata digunakan untuk variabel numerik, sementara modus digunakan untuk variabel kategorikal. Jika jumlah data hilang sangat sedikit, penghapusan baris tersebut menjadi pilihan agar tidak memengaruhi hasil analisis secara signifikan.

3) *Encoding Variabel Kategorikal*

Beberapa fitur dalam dataset, seperti jenis kelamin, persetujuan kelompok, dan persetujuan penanggung jawab, bersifat kategorikal dan tidak dapat langsung digunakan dalam model machine learning. Oleh karena itu, diperlukan proses encoding untuk mengubah nilai-nilai tersebut ke dalam format numerik. Teknik yang digunakan adalah One-Hot Encoding, di mana setiap kategori diubah menjadi kolom biner terpisah. Pendekatan ini memungkinkan model membaca setiap kategori secara independen tanpa mengasumsikan adanya hubungan berurutan antar kategori.

4) *Normalisasi Data*

Langkah terakhir dalam proses preprocessing adalah normalisasi data. Tujuannya adalah menyamakan skala antar fitur numerik agar tidak ada variabel yang mendominasi proses pembelajaran model. Normalisasi dilakukan menggunakan metode Min-Max Scaling, yang mengubah nilai numerik menjadi rentang antara 0 dan 1. Teknik ini penting untuk menjaga konsistensi input, terutama pada model seperti *Logistic Regression* dan *Random Forest* yang dapat dipengaruhi oleh perbedaan skala antar fitur.

C. *Pembagian Dataset*

Dalam penelitian ini, penulis menggunakan teknik *k-fold cross-validation* sebagai metode pembagian data untuk proses pelatihan dan pengujian model. Teknik ini dipilih karena mampu memberikan evaluasi yang lebih menyeluruh dengan memanfaatkan seluruh data secara bergantian sebagai data pelatihan dan pengujian. Dataset dibagi ke dalam *k* bagian (fold) dengan jumlah data yang hampir seimbang. Pada setiap iterasi, satu fold digunakan sebagai data pengujian (*validation set*), sedangkan fold lainnya digunakan sebagai data pelatihan (*training set*). Proses ini dilakukan sebanyak *k* kali, sehingga setiap bagian berperan sebagai data pengujian satu kali.

Untuk memperoleh hasil evaluasi yang stabil dan menghindari bias akibat pembagian data yang tidak merata, penulis menerapkan variasi nilai *k* yang berbeda, yaitu 5-fold, 7-fold, 10-fold, 12-fold, dan 15-fold. Perbandingan terhadap performa model pada tiap skenario digunakan untuk menentukan nilai *k* yang paling sesuai dengan karakteristik dataset dan tujuan analisis dalam penelitian ini.

D. *Pelatihan dan Pengujian Model*

Tahap ini melibatkan pelatihan dan pengujian dua model prediktif yang digunakan secara terpisah, *Logistic Regression* untuk prediksi kelayakan awal pinjaman, dan *Random Forest* untuk klasifikasi tingkat risiko pinjaman setelah 50 minggu. Setiap model menggunakan subset fitur yang berbeda sesuai tujuan masing-masing, serta dievaluasi dengan teknik *k-fold cross-validation* untuk menguji stabilitas dan kemampuan generalisasi model. Data dibagi menjadi data latih dan data uji, dan pemilihan *hyperparameter tuning* dilakukan untuk mengoptimalkan performa model.

Model *Logistic Regression* dilatih menggunakan fitur-fitur seperti usia, jenis kelamin, persetujuan kelompok, persetujuan penanggung jawab, pendapatan mingguan, dan jumlah pinjaman. Proses pelatihan mencakup *hyperparameter tuning* terhadap parameter seperti C, solver, class_weight, dan max_iter. Model ini menghasilkan klasifikasi biner untuk menentukan apakah nasabah "layak" atau "tidak layak" menerima pinjaman awal.

Model *Random Forest* digunakan untuk memprediksi tingkat risiko nasabah berdasarkan fitur yang mencakup usia, jenis kelamin, persetujuan kelompok, persetujuan penanggung

jawab, pendapatan mingguan, jumlah pinjaman, jumlah keterlambatan, dan minggu meminjam. Untuk menangani ketidakseimbangan kelas dalam data pelatihan, digunakan metode SMOTE pada setiap fold. *Hyperparameter tuning* yang disesuaikan meliputi n_estimators, max_depth, min_samples_split, dan class_weight. Model ini menghasilkan klasifikasi multikelas ke dalam tiga kategori risiko: rendah, sedang, dan tinggi.

E. *Hasil Evaluasi Model*

Setelah proses pelatihan, langkah selanjutnya adalah mengevaluasi kinerja model terhadap data uji. Evaluasi ini bertujuan untuk mengukur sejauh mana model mampu memprediksi kelayakan pinjaman dan tingkat risiko nasabah secara akurat berdasarkan fitur-fitur yang tersedia.

Beberapa metrik evaluasi digunakan untuk menilai performa model, yaitu akurasi, presisi, recall, dan F1-score. Metrik-metrik ini dipilih karena mampu memberikan gambaran menyeluruh terhadap kemampuan model, baik dalam mengidentifikasi prediksi yang benar, mengurangi kesalahan klasifikasi, maupun menjaga keseimbangan antara presisi dan recall.

Secara matematis, metrik evaluasi yang digunakan dirumuskan sebagai berikut:

$$Akurasi = \frac{Jumlah\ Prediksi\ Benar}{Total\ Jumlah\ Prediksi} \quad (1)$$

$$Presisi = \frac{Jumlah\ Prediksi\ Positif\ Benar}{Total\ Jumlah\ Prediksi\ Positif} \quad (2)$$

$$Recall = \frac{Jumlah\ Prediksi\ Positif\ Benar}{Total\ Jumlah\ Data\ Positif} \quad (3)$$

$$F1 - Score = 2 \times \frac{Presisi \times Recall}{Presisi + Recall} \quad (4)$$

Evaluasi dilakukan terhadap masing-masing model: *Logistic Regression*, yang digunakan untuk prediksi awal kelayakan pinjaman, serta *Random Forest*, yang digunakan untuk mengklasifikasikan risiko nasabah. Hasil dari evaluasi ini akan menjadi dasar untuk menganalisis efektivitas model dan menilai keandalannya dalam mendukung pengambilan keputusan pinjaman berdasarkan analisis data yang objektif.

III. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil evaluasi dari dua model yang digunakan dalam penelitian, yaitu *Logistic Regression* untuk prediksi kelayakan awal pinjaman, dan *Random Forest* untuk klasifikasi tingkat risiko nasabah dalam pemberian top-up pinjaman. Evaluasi dilakukan menggunakan data uji dengan menerapkan skema k-fold *cross-validation* serta pengukuran performa berdasarkan metrik akurasi, presisi, recall, dan F1-score.

Hasil yang diperoleh kemudian dianalisis untuk menilai efektivitas model dalam memprediksi dengan tepat berdasarkan profil dan perilaku pembayaran nasabah. Selain itu, pembahasan juga mencakup interpretasi hasil untuk mengetahui relevansi antara prediksi model dengan aturan kelayakan yang diterapkan oleh PNM Mekaar.

A. *Pengumpulan Dataset*

Dataset yang digunakan dalam penelitian ini diperoleh secara langsung dari ketua kelompok nasabah aktif Program PNM Mekaar. Data diserahkan dalam format spreadsheet (.csv) dan mencakup atribut-atribut penting yang merepresentasikan profil risiko nasabah, seperti usia, jenis kelamin, pendapatan mingguan, jumlah pinjaman, persetujuan dari kelompok dan penanggung jawab, jumlah keterlambatan pembayaran, serta durasi meminjam (dalam minggu).

Pemilihan sumber data melalui ketua kelompok didasarkan pada perannya sebagai pihak yang secara langsung mencatat aktivitas pembayaran dan informasi keanggotaan, sehingga dianggap kredibel dan relevan untuk tujuan penelitian. Proses pengumpulan data dilakukan secara langsung di lapangan dengan total 213 entri, dan didukung dengan surat permohonan resmi yang ditandatangani oleh petugas lapangan PNM Mekaar.

Dataset ini digunakan untuk dua tahap klasifikasi, yaitu:

(1) klasifikasi kelayakan pinjaman menggunakan model *Logistic Regression*, dan

(2) klasifikasi risiko nasabah menggunakan model *Random Forest*.

Distribusi data untuk masing-masing model ditunjukkan pada Tabel II dan Tabel III.

TABEL II
TOTAL DATASET MODEL LOGISTIC REGRESSION

Tahap model	Kategori	Jumlah data
Logistic Regression	Layak	174
	Tidak layak	39
	Total	**213**

TABEL III
TOTAL DATASET MODEL RANDOM FOREST

Tahap model	Kategori	Jumlah data
Random Forest	Risiko rendah	102
	Risiko sedang	52
	Risiko tinggi	20
	Total	**174**

Distribusi pada model *Logistic Regression* tergolong seimbang, sehingga tidak diperlukan penanganan tambahan

terhadap data. Sementara itu, pada model *Random Forest*, dilakukan penyeimbangan kelas menggunakan SMOTE *(Synthetic Minority Oversampling Technique)* guna memastikan distribusi kelas yang proporsional sebelum pelatihan model dilakukan.

B. *Preprocessing Dataset*

Tahap preprocessing merupakan langkah awal yang krusial sebelum data digunakan dalam pelatihan model. Tujuan utamanya adalah memastikan data memiliki format dan kualitas yang sesuai agar proses pelatihan berjalan optimal.

Dalam penelitian ini, preprocessing dilakukan untuk menyiapkan data nasabah sebelum digunakan oleh algoritma *Logistic Regression* dan *Random Forest*. Proses ini meliputi beberapa tahapan, seperti Pembersihan data, Pengisian data hilang, Encoding variabel kategorikal, serta Normalisasi data, agar sesuai dengan kebutuhan masing-masing model. Dengan preprocessing yang tepat, data dapat diolah secara konsisten dan hasil prediksi menjadi lebih akurat dan representatif.

1) *Pembersihan Data*

Tahap awal dalam preprocessing adalah pembersihan data, yang bertujuan untuk menghilangkan data yang tidak relevan atau berpotensi mengganggu proses pelatihan model. Langkah ini penting untuk memastikan data yang digunakan bersih, akurat, dan sesuai dengan tujuan analisis.

Dalam proses ini, tiga kolom dihapus karena tidak memberikan kontribusi signifikan terhadap pemodelan, yaitu nama, jenis_usaha, dan rasio_keterlambatan. Kolom nama hanya berfungsi sebagai identitas, jenis_usaha dianggap redundan karena informasi usaha telah direpresentasikan melalui fitur pendapatan_mingguan, sedangkan rasio_keterlambatan merupakan hasil turunan dari fitur lain yang telah digunakan secara langsung. Proses penghapusan kolom tersebut ditunjukkan pada Gbr 2.

```
df_model = df.drop(columns=['nama', 'jenis_usaha', 'rasio_keterlambatan'])
```

Gbr 2 Pembersihan kolom tidak relevan

Selanjutnya, dilakukan pemeriksaan terhadap kemungkinan adanya data duplikat menggunakan fungsi duplicated() dari pustaka *pandas*. Jumlah data duplikat yang terdeteksi ditampilkan terlebih dahulu, kemudian dilakukan penghapusan menggunakan fungsi drop_duplicates() seperti diperlihatkan pada Gbr 3.

```
jumlah_duplikat = df_model.duplicated().sum()
print(f"Jumlah data duplikat yang ditemukan: {jumlah_duplikat}")

df_model = df_model.drop_duplicates()
```

Gbr 3 Pembersihan data duplikat

Hasil akhir dari proses pembersihan data ditampilkan pada Gbr 4.

```
Ukuran data sebelum pembersihan: (213, 13)
Jumlah data duplikat yang ditemukan: 0
Jumlah kolom yang dihapus: 3
Ukuran data setelah pembersihan: (213, 10)
```

Gbr 4 Hasil pembersihan dataset

Dataset awal terdiri dari 213 baris dan 13 kolom. Setelah dilakukan pemeriksaan dan penghapusan tiga kolom tidak relevan, jumlah kolom tersisa menjadi 10, tanpa adanya data duplikat. Dataset ini kemudian digunakan untuk tahapan preprocessing selanjutnya.

2) *Pengisian Data Hilang*

Langkah berikutnya dalam preprocessing adalah pemeriksaan dan penanganan data hilang *(missing values)*. Kehadiran nilai kosong dapat berdampak negatif terhadap performa model, terutama jika terjadi pada fitur-fitur yang memiliki kontribusi penting dalam proses prediksi.

Pemeriksaan dilakukan dengan menggunakan fungsi isnull().sum() dari pustaka *pandas*, untuk mendeteksi apakah ada fitur dalam dataset yang mengandung nilai kosong. Hasil pemeriksaan menunjukkan bahwa tidak terdapat nilai kosong pada seluruh fitur, Seperti ditunjukkan pada Gbr 5.

```
print("Jumlah missing values per kolom:\n", df.isnull().sum())
```

Gbr 5 Proses pengisian data hilang

Seluruh fitur dalam dataset diketahui tidak memiliki nilai kosong (semua bernilai nol), sehingga tidak diperlukan proses imputasi maupun penghapusan data, sebagaimana ditunjukkan pada Gbr 6.

```
Jumlah missing values per kolom:
 nama                          0
usia                           0
jenis_kelamin                  0
persetujuan_kelompok           0
persetujuan_penanggungjawab    0
jenis_usaha                    0
pendapatan_mingguan            0
jumlah_pinjaman                0
jumlah_keterlambatan           0
minggu_meminjam                0
rasio_keterlambatan            0
kelayakan_pinjaman             0
risiko_nasabah                 0
```

Gbr 6 Hasil pengisian data hilang

Dengan hasil tersebut, maka tidak diperlukan proses imputasi maupun penghapusan data. Hal ini memastikan bahwa data yang digunakan sudah lengkap dan siap untuk dilanjutkan ke tahap preprocessing berikutnya.

3) *Encoding Variabel Kategorikal*

Tahapan berikutnya dalam preprocessing adalah mengonversi fitur kategorikal menjadi bentuk numerik. Langkah ini krusial karena sebagian besar algoritma pembelajaran mesin, seperti *Logistic Regression* dan *Random Forest*, tidak dapat bekerja langsung dengan data berupa string.

Dalam dataset ini, terdapat tiga fitur kategorikal, yaitu jenis_kelamin, persetujuan_kelompok, dan persetujuan_penanggungjawab. Masing-masing hanya memiliki dua nilai unik, misalnya jenis_kelamin bernilai *perempuan* atau *laki-laki*, sedangkan dua fitur lainnya bernilai *ya* atau *tidak*.

Untuk menghindari adanya asumsi urutan atau bobot antar kategori, digunakan metode One-Hot Encoding. Teknik ini mengubah setiap kategori menjadi kolom biner yang merepresentasikan keberadaan nilai tersebut dalam data. Implementasi dilakukan menggunakan fungsi get_dummies() dari pustaka *pandas*, sebagaimana ditunjukkan pada Gbr 7.

```
kolom_kategorikal = ['jenis_kelamin', 'persetujuan_kelompok',
                     'persetujuan_penanggungjawab']
df_encoded = pd.get_dummies(df_model, columns=kolom_kategorikal)

for col in df_encoded.columns:
    if df_encoded[col].dtype == 'bool':
        df_encoded[col] = df_encoded[col].astype(int)
```

Gbr 7 Proses one-hot encoding

Hasil transformasi nilai kategorikal menjadi representasi numerik diperlihatkan pada Tabel IV.

TABEL IV
TRANSFORMASI NILAI BENTUK NUMERIK

Fitur	Nilai Asli	Kolom Hasil Encoding	Nilai
jenis_kelamin	perempuan	jenis_kelamin_perempuan	1
	laki-laki	jenis_kelamin_perempuan	0
persetujuan_kelompok	ya	persetujuan_kelompok_ya	1
	tidak	persetujuan_kelompok_ya	0
persetujuan_penanggungjawab	ya	persetujuan_penanggungjawab_ya	1
	tidak	persetujuan_penanggungjawab_ya	0

Karena setiap fitur hanya memiliki dua nilai unik, proses one-hot encoding tidak menimbulkan dummy variable trap. Dengan demikian, seluruh kolom hasil encoding dapat digunakan tanpa perlu dihapus salah satunya.

Struktur akhir dataset setelah encoding bersifat sepenuhnya numerik dan siap digunakan dalam tahapan normalisasi maupun pelatihan model. Contoh hasil akhir dari proses one-hot encoding ditunjukkan pada Gbr 8.

	jenis_kelamin_laki-laki	jenis_kelamin_perempuan	persetujuan_kelompok_tidak	persetujuan_kelompok_ya	\
	0	1	0	1	
	0	1	0	1	
	0	1	0	1	
	0	1	0	1	
	0	1	0	1	

	persetujuan_penanggungjawab_tidak	persetujuan_penanggungjawab_ya
	0	1
	0	1
	0	1
	0	1
	0	1

Gbr 8 Hasil one-hot encoding

4) *Normalisasi Data*

Langkah akhir dalam preprocessing adalah normalisasi data, yaitu menyamakan skala seluruh fitur numerik agar setara dalam kontribusinya terhadap model. Normalisasi diperlukan untuk menghindari dominasi fitur dengan nilai besar terhadap algoritma pembelajaran, terutama pada metode seperti *Logistic Regression* yang sensitif terhadap skala fitur.

Dalam penelitian ini, digunakan teknik *Min-Max Scaling* untuk mentransformasikan setiap nilai numerik ke rentang [0, 1]. Proses ini dilakukan menggunakan fungsi *MinMaxScaler* dari pustaka sklearn.preprocessing, sebagaimana ditunjukkan pada Gbr 9.

```
from sklearn.preprocessing import MinMaxScaler

fitur_numerik = ['usia', 'pendapatan_mingguan', 'jumlah_pinjaman',
                 'jumlah_keterlambatan', 'minggu_meminjam']

scaler = MinMaxScaler()
df_encoded[fitur_numerik] = scaler.fit_transform(df_encoded[fitur_numerik])
```

Gbr 9 Proses MinMaxScaler

Setelah normalisasi diterapkan, seluruh nilai numerik memiliki skala yang seragam. Dengan skala yang konsisten, setiap fitur memiliki pengaruh yang setara dalam proses pelatihan model, serta membantu mencegah bias terhadap fitur tertentu. Contoh hasil transformasi dapat dilihat pada Gbr 10.

	usia	pendapatan_mingguan	jumlah_pinjaman	jumlah_keterlambatan	minggu_meminjam
0.465116	1.000000	0.8	0.000000	0.520833	
0.232558	0.230769	0.0	0.000000	0.645833	
0.837209	0.269231	0.0	0.363636	1.000000	
0.558140	0.192308	0.0	0.136364	0.375000	
0.697674	0.500000	0.4	0.000000	0.062500	

Gbr 10 Hasil Min-Max Scaling

Normalisasi ini sangat relevan bagi algoritma *Logistic Regression*, yang mengoptimalkan bobot berdasarkan perbedaan nilai antar fitur. Dengan demikian, data numerik telah siap sepenuhnya untuk digunakan pada tahap pelatihan model.

C. *Pembagian Dataset*

Tahap ini mencakup proses pembagian data menggunakan teknik *Stratified K-Fold Cross-Validation* untuk kedua model yang digunakan, yaitu *Logistic Regression* dan *Random Forest*. Teknik ini dipilih karena mampu mempertahankan proporsi distribusi kelas target pada setiap fold, sehingga hasil evaluasi model menjadi lebih representatif.

Pembagian data dilakukan dengan memanfaatkan class StratifiedKFold dari pustaka *scikit-learn*. Untuk mendukung proses ini, dibuat fungsi run_kfold() yang bertugas membagi data menjadi *training set* dan *test set* sebanyak *k* kali secara stratifikasi, sekaligus menampilkan jumlah data latih dan uji di setiap fold untuk memastikan pembagian dilakukan secara adil dan seimbang. Kode implementasi fungsi ini ditunjukkan pada Gbr 11.

```
from sklearn.model_selection import StratifiedKFold

# --- Fungsi Umum untuk K-Fold Cross Validation ---
def run_kfold(X, y, k=5):
    skf = StratifiedKFold(n_splits=k, shuffle=True, random_state=42)

    print(f"\n=== {k}-Fold Cross Validation ===")
    for fold, (train_index, test_index) in enumerate(skf.split(X, y), 1):
        X_train, X_test = X.iloc[train_index], X.iloc[test_index]
        y_train, y_test = y.iloc[train_index], y.iloc[test_index]

        print(f"Fold {fold}:")
        print(f"  Jumlah data latih : {len(X_train)}")
        print(f"  Jumlah data uji   : {len(X_test)}")
```

Gbr 11 Fungsi pembagian dataset

1) *Pembagian Dataset Model Logistic Regression*

Pada tahap ini, data untuk model *Logistic Regression* dibagi menggunakan teknik Stratified K-Fold Cross-Validation. Metode ini dipilih karena mampu menjaga proporsi label target (layak dan tidak layak) di setiap fold, sehingga distribusi data tetap seimbang seperti pada data asli.

Model dilatih menggunakan 213 data, terdiri dari 174 data kategori "layak" dan 39 data "tidak layak". Pembagian data dilakukan menggunakan fungsi run_kfold() dengan variasi jumlah fold, k = 5, 7, 10, 12, dan 15. Tujuannya adalah untuk mengevaluasi konsistensi performa model dalam skenario pembagian data yang berbeda.

Semakin besar nilai *k*, maka jumlah data latih akan meningkat dan data uji akan berkurang. Ini memungkinkan model diuji dengan porsi data latih yang lebih besar, tetapi juga memerlukan waktu komputasi yang lebih tinggi karena proses pelatihan dan pengujian dilakukan berulang kali.

Jumlah rata-rata data latih dan data uji pada setiap nilai *k* ditunjukkan pada Tabel V.

TABEL V
JUMLAH PEMBAGIAN DATA MODEL LOGISTIC REGRESSION

Fold (*k*)	Jumlah Data Latih	Jumlah Data Uji
5	± 170	± 43
7	± 183	± 30
10	± 192	± 21
12	± 195	± 18
15	± 199	± 14

Dengan pendekatan ini, evaluasi model menjadi lebih menyeluruh, karena performanya dihitung berdasarkan rata-rata dari beberapa skenario pembagian data, bukan hanya satu kali pemisahan.

2) *Pembagian Data Model Random Forest*

Model Random Forest dalam penelitian ini dikembangkan berdasarkan data nasabah yang telah dinyatakan "layak" menerima pinjaman, dengan tambahan label risiko (rendah, sedang, dan tinggi) yang ditentukan dari fitur numerik seperti jumlah keterlambatan dan lama meminjam. Komposisi awal data menunjukkan ketidakseimbangan kelas yang cukup mencolok, dengan 102 nasabah berisiko rendah, 52 sedang, dan hanya 20 berisiko tinggi.

Untuk menjaga validitas evaluasi, digunakan teknik Stratified K-Fold Cross-Validation agar distribusi label tetap proporsional pada setiap fold. Pendekatan ini dipilih karena mampu menjaga representasi kelas minoritas sekaligus mengurangi potensi bias selama proses pelatihan dan pengujian.

Ketimpangan kelas menjadi tantangan utama, terutama dalam mengidentifikasi nasabah berisiko tinggi. Untuk itu, diterapkan metode SMOTE *(Synthetic Minority Oversampling Technique)* secara khusus pada data latih di setiap fold. Strategi ini dilakukan agar data uji tetap murni dan terhindar dari kebocoran data, sehingga hasil evaluasi lebih objektif.

Eksperimen dilakukan dengan variasi jumlah fold *k* = 5, 7, 10, 12, dan 15, melalui fungsi run_kfold() yang dijalankan pada fitur X_rf dan target y_rf. Dengan mencoba beberapa skenario jumlah fold, dapat diamati stabilitas performa model. Nilai *k* yang lebih besar berarti proporsi data pelatihan semakin

besar, namun juga meningkatkan beban komputasi karena proses pelatihan dilakukan lebih sering.

Jumlah rata-rata data latih dan data uji pada setiap nilai *k* sebelum dilakukan SMOTE ditunjukkan pada Tabel VI.

TABEL VI
JUMLAH PEMBAGIAN DATA MODEL RANDOM FOREST

Fold (*k*)	Jumlah Data Latih	Jumlah Data Uji
5	± 139	± 35
7	± 149	± 25
10	± 157	± 17
12	± 159	± 15
15	± 162	± 12

Dengan menerapkan SMOTE secara selektif di data latih, proses validasi model menjadi lebih adil dan mewakili performa sebenarnya terhadap seluruh kelas risiko. Teknik ini juga memungkinkan evaluasi yang lebih komprehensif dan reliabel, tanpa mengorbankan integritas data uji.

D. *Pelatihan dan Pengujian Model*

Tahap ini bertujuan untuk mengevaluasi kemampuan model dalam mempelajari pola dari data pelatihan serta mengukur performanya terhadap data yang belum pernah dilihat sebelumnya. Proses pelatihan dilakukan menggunakan data hasil pembagian sebelumnya, sedangkan evaluasi dilakukan pada setiap fold menggunakan metrik akurasi, presisi, recall, dan f1-score. Dengan pendekatan *k*-fold *cross-validation*, hasil evaluasi mencerminkan performa model secara menyeluruh dan tidak bias terhadap satu skenario pembagian data.

Untuk memperoleh hasil yang optimal, masing-masing algoritma juga menjalani proses *hyperparameter tuning* guna mencari kombinasi parameter terbaik. Seluruh proses pelatihan dan pengujian dilakukan secara terpisah untuk model *Logistic Regression* dan *Random Forest*.

1) *Pelatihan dan Pengujian Model Logistic Regression*

Model *Logistic Regression* dilatih untuk memprediksi kelayakan pinjaman berdasarkan data profil risiko nasabah yang telah melalui proses prapemrosesan. Proses pelatihan bertujuan agar model dapat mengenali pola dari data historis, sementara pengujian dilakukan untuk menilai performa model dalam memprediksi data baru yang belum pernah dilihat sebelumnya.

Dalam proses pelatihan, digunakan teknik pencarian kombinasi hyperparameter menggunakan pendekatan *grid search* untuk meningkatkan akurasi prediksi. Pemilihan nilai parameter dilakukan berdasarkan karakteristik data dan hasil uji coba awal. Parameter yang diuji dan nilai-nilainya disajikan pada Tabel VII.

TABEL VII
JENIS PARAMETER MODEL LOGISTIC REGRESSION

Nama Parameter	Nilai yang Diuji
C	1, 10, 100
Solver	'lbfgs', 'liblinear'
class_weight	None, 'balanced'
max_iter	100, 400, 700, 1000

Seluruh kombinasi dari parameter di atas menghasilkan total 48 konfigurasi. Setiap konfigurasi digunakan untuk melatih model dan dievaluasi menggunakan metrik klasifikasi, yaitu akurasi, presisi, recall, dan F1-score. Evaluasi ini memberikan gambaran menyeluruh mengenai kemampuan model dalam membedakan nasabah yang layak dan tidak layak menerima pinjaman.

Hasil evaluasi menunjukkan bahwa pemilihan kombinasi hyperparameter yang tepat dapat meningkatkan performa model secara signifikan. Model *Logistic Regression* terbukti mampu melakukan klasifikasi dengan cukup baik dan dapat dijadikan dasar dalam pengambilan keputusan kelayakan pinjaman secara data-driven di lingkungan PNM Mekaar.

2) *Pelatihan dan Pengujian Model Random Forest*

Model *Random Forest* digunakan untuk memprediksi tingkat risiko pinjaman nasabah PNM Mekaar, yang dikategorikan ke dalam tiga kelas (rendah, sedang, dan tinggi). Pelatihan dilakukan pada data yang telah melalui tahap prapemrosesan dan penyeimbangan kelas menggunakan SMOTE, agar model tidak bias terhadap kelas mayoritas dan mampu mengenali pola dari seluruh kategori risiko secara adil.

Untuk mengoptimalkan performa model, dilakukan eksplorasi kombinasi *hyperparameter* melalui *grid search*. Parameter yang diuji meliputi jumlah pohon (*n_estimators*), kedalaman maksimum pohon (*max_depth*), jumlah minimal sampel untuk pemisahan internal (*min_samples_split*), serta bobot kelas (*class_weight*). Kombinasi parameter tersebut disajikan pada Tabel VIII.

TABEL VIII
JENIS PARAMETER MODEL RANDOM FOREST

Nama Parameter	Nilai yang Diuji
n_estimators	100, 200, 300
max_depth	None, 5, 10
min_samples_split	2, 5, 10
class_weight	None, 'balanced'

Seluruh kombinasi dari nilai-nilai tersebut menghasilkan 54 konfigurasi yang diuji menggunakan *Stratified K-Fold Cross-Validation*. Pada setiap fold, data pelatihan terlebih dahulu diseimbangkan dengan SMOTE agar distribusi kelas risiko menjadi seimbang.

Evaluasi model dilakukan menggunakan metrik akurasi, presisi, *recall*, dan F1-score, yang dihitung rata-rata dari seluruh fold. Strategi ini memungkinkan pemilihan model terbaik berdasarkan kinerja rata-rata yang merepresentasikan kemampuan generalisasi terhadap data baru.

Dengan pendekatan ini, model Random Forest yang dihasilkan tidak hanya akurat, tetapi juga sensitif terhadap perbedaan antar kelas, sehingga dapat digunakan untuk mendukung keputusan pemberian pinjaman dan top-up yang lebih tepat sasaran.

E. *Hasil Evaluasi Model*

Bagian ini menyajikan hasil evaluasi terhadap model yang telah dibangun, yaitu *Logistic Regression* dan *Random Forest*. Evaluasi dilakukan dengan menggunakan metrik akurasi, presisi, recall, dan F1-score melalui skema K-Fold Cross-Validation. Hasil ini digunakan untuk menilai seberapa baik model dalam memprediksi kelayakan dan tingkat risiko pinjaman nasabah PNM Mekaar secara obyektif, serta untuk membandingkan stabilitas dan kemampuan generalisasi masing-masing algoritma.

1) *Hasil Evaluasi Model Logistic Regression*

Evaluasi terhadap model *Logistic Regression* dilakukan menggunakan validasi silang dengan variasi nilai *k* (5, 7, 10, 12, dan 15) guna mengukur stabilitas dan konsistensi performa model. Masing-masing nilai *k* menghasilkan pembagian data yang berbeda sebagai data latih dan data uji. Evaluasi dilakukan dengan menghitung rata-rata dari empat metrik utama: akurasi, presisi, recall, dan F1-score.

Hasil evaluasi menunjukkan bahwa model memiliki performa yang relatif stabil pada seluruh skema *k*-fold, dengan sedikit variasi pada masing-masing metrik. Secara umum, model mampu mengklasifikasikan nasabah dengan cukup baik, terutama dalam mendeteksi nasabah yang layak menerima pinjaman. Rangkuman hasil evaluasi untuk setiap variasi *k* disajikan pada Tabel IX.

TABEL IX
RATA-RATA HASIL TIAP FOLD MODEL LOGISTIC REGRESSION

Nilai *k*	Akurasi	Presisi	Recall	F1-score
5	82.18	82.72	98.86	90.07
7	81.71	82.63	98.29	89.76
10	81.23	81.60	99.44	89.63
12	81.75	82.98	97.70	89.71
15	81.75	82.68	98.28	89.75

Dari tabel tersebut terlihat bahwa konfigurasi $k = 5$ memberikan hasil evaluasi paling optimal secara keseluruhan. F1-score tertinggi serta keseimbangan antara presisi dan recall menjadi indikator utama bahwa skema ini mampu menangkap pola klasifikasi secara lebih andal. Oleh karena itu, model *Logistic Regression* dengan validasi 5-fold digunakan pada tahap akhir pengujian model untuk memperoleh hasil prediksi yang paling representatif.

2) *Hasil Evaluasi Model Random Forest*

Evaluasi model *Random Forest* dilakukan untuk mengukur kemampuannya dalam mengklasifikasikan tingkat risiko nasabah yang sebelumnya telah dinyatakan *layak* oleh model Logistic Regression. Proses evaluasi menggunakan skema K-Fold Cross-Validation dengan variasi nilai *k* (5, 7, 10, 12, dan 15). Untuk mengatasi ketimpangan kelas, metode SMOTE diterapkan pada data latih sebelum pelatihan dimulai.

Evaluasi difokuskan pada nasabah yang telah menerima pinjaman awal, dengan mempertimbangkan riwayat pembayaran serta durasi peminjaman. Setiap skema dievaluasi berdasarkan akurasi, presisi, recall, dan F1-score, yang kemudian dirata-rata untuk setiap nilai k. Hasil lengkapnya disajikan pada Tabel X.

TABEL X
RATA-RATA HASIL TIAP FOLD MODEL RANDOM FOREST

Nilai *k*	Akurasi	Presisi	Recall	F1-score
5	86.25	83.09	82.02	81.64
7	86.81	84.35	81.88	82.32
10	83.79	80.60	77.22	77.53
12	82.54	75.09	74.68	73.16
15	84.34	73.19	77.04	73.18

Berdasarkan hasil tersebut, konfigurasi 7-Fold menghasilkan performa terbaik secara keseluruhan. Skema ini dipilih karena menunjukkan keseimbangan yang baik di keempat metrik utama, serta memberikan hasil yang relatif stabil dan konsisten antar fold. Oleh karena itu, model *Random Forest* dengan validasi 7-Fold digunakan sebagai dasar dalam uji coba akhir klasifikasi risiko nasabah.

F. *Uji Coba Data Baru*

Setelah proses pelatihan dan evaluasi dilakukan, tahap selanjutnya adalah melakukan uji coba untuk melihat kemampuan model dalam memprediksi data baru. Uji coba ini bertujuan untuk menilai sejauh mana model dapat menggeneralisasi terhadap data yang tidak pernah dilibatkan dalam pelatihan maupun validasi.

Pada penelitian ini, uji coba dilakukan terhadap dua model, *Logistic Regression* yang digunakan untuk menentukan kelayakan pinjaman awal, dan *Random Forest* yang digunakan untuk mengklasifikasikan tingkat risiko nasabah terhadap kemungkinan keterlambatan pembayaran. Model terbaik dari masing-masing algoritma berdasarkan hasil evaluasi k-fold *cross-validation* digunakan pada tahap ini, yaitu 5-fold untuk *Logistic Regression* dan 7-fold untuk *Random Forest*.

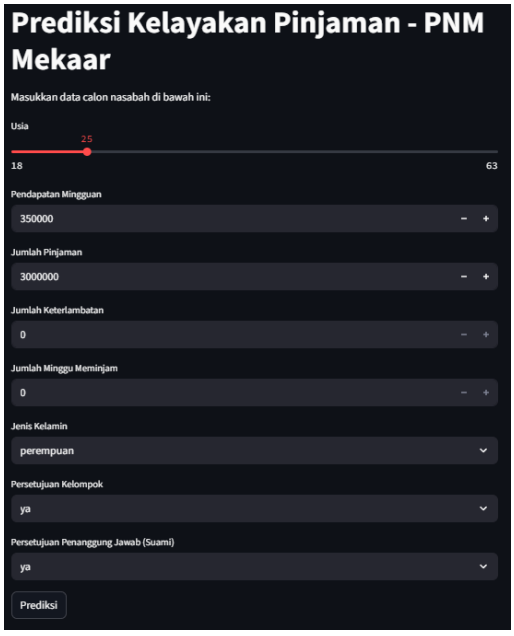
Pengujian dilakukan melalui antarmuka interaktif berbasis Streamlit, yang memungkinkan pengguna memasukkan data secara langsung dan memperoleh hasil prediksi secara real-time. Pendekatan ini dirancang untuk mencerminkan situasi nyata dalam proses penentuan kelayakan dan risiko pinjaman di lapangan.

1) *Uji Coba Model Logistic Regression*

Uji coba terhadap model *Logistic Regression* dilakukan menggunakan data baru yang tidak terlibat dalam proses pelatihan maupun validasi. Tujuan dari tahap ini adalah untuk menilai kemampuan model dalam memprediksi kelayakan pinjaman berdasarkan profil risiko nasabah, serta sejauh mana hasil prediksi selaras dengan aturan kelayakan yang berlaku di PNM Mekaar.

Sebelum dilakukan prediksi oleh model, sistem menerapkan sejumlah aturan validasi awal (*hard rules*) yang bersifat deterministik. Beberapa di antaranya mencakup syarat bahwa nasabah harus berjenis kelamin perempuan, berada pada rentang usia 18 hingga 63 tahun, serta mendapat persetujuan dari kelompok dan penanggung jawab (suami). Apabila salah satu syarat tidak terpenuhi, maka status pinjaman langsung dinyatakan "tidak layak" tanpa perlu diproses lebih lanjut oleh model.

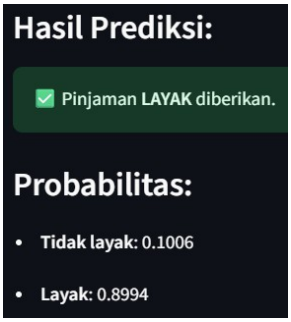
Proses uji coba dilakukan melalui antarmuka interaktif berbasis Streamlit, yang memungkinkan input data secara manual sesuai struktur data pelatihan. Model yang digunakan merupakan hasil pelatihan terbaik dari skema 5-fold cross-validation.



Gbr 12 Input uji coba model logistic regression

Gbr 12 merepresentasikan seorang calon nasabah perempuan berusia 25 tahun, dengan persetujuan dari kelompok dan penanggung jawab (suami), serta memiliki pendapatan mingguan sebesar Rp350.000 dan ingin meminjam sebesar Rp3.000.000.

Setelah pengguna memasukkan data, hasil prediksi akan menghasilkan seperti pada Gbr 13.



Gbr 13 Hasil uji coba model logistic regression

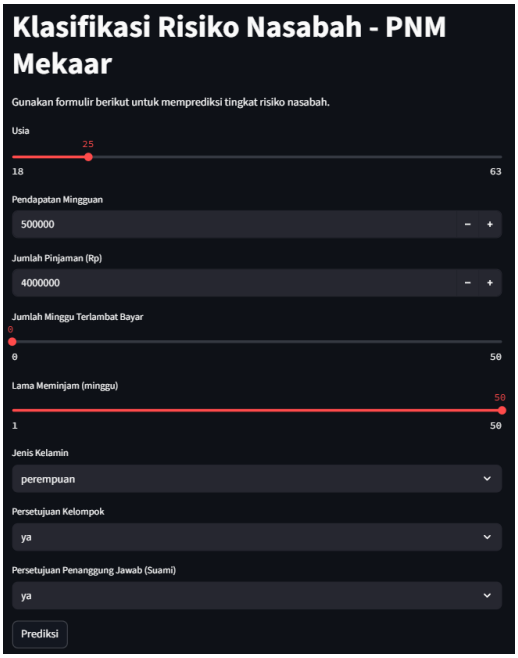
Berdasarkan hasil uji coba, model memprediksi bahwa nasabah tergolong "layak" dengan tingkat probabilitas sebesar 89,94%. Nilai ini menunjukkan keyakinan model yang tinggi terhadap klasifikasi tersebut, dengan hanya 10,06% probabilitas masuk ke kelas "tidak layak". Secara umum, hasil prediksi konsisten dengan ketentuan kelayakan yang telah ditetapkan sebelumnya, sehingga model *Logistic Regression* dapat berfungsi sebagai alat bantu dalam proses evaluasi awal pengajuan pinjaman di PNM Mekaar.

2) *Uji Coba Model Random Forest*

Model *Random Forest* juga diuji menggunakan data baru untuk mengevaluasi kemampuan dalam memprediksi tingkat risiko nasabah berdasarkan profil dan riwayat pinjaman. Uji coba ini ditujukan untuk melihat sejauh mana model mampu mengklasifikasikan risiko sesuai dengan ketentuan kelayakan top-up pinjaman yang diterapkan oleh PNM Mekaar.

Sebelum proses prediksi dilakukan, sistem terlebih dahulu menerapkan aturan validasi awal, seperti memastikan nasabah adalah perempuan, berusia antara 18 hingga 63 tahun, dan telah mendapat persetujuan dari kelompok serta penanggung jawab. Jika salah satu dari syarat ini tidak terpenuhi, maka nasabah langsung diklasifikasikan ke dalam kategori "risiko tinggi", tanpa melibatkan model prediktif.

Proses uji coba dilakukan melalui antarmuka berbasis Streamlit dengan menggunakan sampel data yang mengikuti struktur data saat pelatihan. Model yang digunakan merupakan hasil pelatihan dengan skema 7-fold cross-validation pada fold dengan performa terbaik. Selain itu, data latih telah diseimbangkan menggunakan metode SMOTE untuk memastikan distribusi kelas risiko yang proporsional.



Gbr 14 Input uji coba model random forest

Gbr 14 merepresentasikan seorang nasabah perempuan berusia 25 tahun, dengan persetujuan dari kelompok dan penanggung jawab (suami), memiliki pendapatan mingguan sebesar Rp500.000 dengan pinjaman sebesar Rp4.000.000, Berdasarkan riwayatnya, nasabah telah meminjam selama 50 minggu dengan jumlah keterlambatan pembayaran sebanyak 0 kali.

Setelah pengguna memasukkan data, hasil prediksi akan menghasilkan seperti pada Gbr 15.

Gbr 15 Hasil uji coba model random forest

Hasil uji coba menunjukkan bahwa model memprediksi nasabah berada dalam kategori "risiko rendah", dengan probabilitas sebesar 80,23%, diikuti oleh 18,47% untuk kategori "sedang", dan 1,30% untuk kategori "tinggi". Prediksi ini mengindikasikan bahwa nasabah dinilai memiliki potensi gagal bayar yang rendah, sesuai dengan profil yang stabil dan telah memenuhi seluruh kriteria kelayakan utama. Secara keseluruhan, model mampu memberikan klasifikasi yang konsisten dan relevan dalam mengidentifikasi tingkat risiko nasabah.

IV. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, berikut adalah kesimpulan dari penelitian ini:

1. Model *Logistic Regression* terbukti mampu memprediksi kelayakan pinjaman nasabah PNM Mekaar dengan cukup baik, berdasarkan sejumlah variabel seperti usia, jenis kelamin, persetujuan kelompok dan penanggung jawab, pendapatan mingguan, serta jumlah pinjaman. Evaluasi menggunakan k-fold *cross-validation* menunjukkan performa model yang stabil dalam membedakan nasabah yang layak dan tidak layak.
2. Model *Random Forest* digunakan untuk mengklasifikasikan tingkat risiko dari nasabah yang sudah dinyatakan layak, sebagai dasar pertimbangan pemberian top-up pinjaman setelah 50 minggu. Dengan menambahkan variabel keterlambatan pembayaran dan durasi pinjaman, serta penerapan teknik SMOTE untuk menyeimbangkan data, model ini berhasil mengelompokkan nasabah ke dalam kategori risiko rendah, sedang, dan tinggi secara efektif.
3. Hasil penelitian menunjukkan bahwa kombinasi kedua model ini dapat menjadi pendekatan yang bermanfaat dalam mendukung pengambilan keputusan strategis di PNM Mekaar. Logistic Regression dapat digunakan sebagai penyaring awal, sedangkan Random Forest memberikan gambaran risiko lanjutan. Pendekatan ini dinilai mampu membantu meminimalkan risiko kredit bermasalah dan meningkatkan akurasi dalam penyaluran top-up pinjaman.

DAFTAR PUSTAKA

[1] A. R. Fadillah *et al.*, "ANALISIS PERBANDINGAN LINEAR REGRESSION DAN RANDOM FOREST REGRESSION UNTUK PREDIKSI BATAS KREDIT :," hal. 543–550, 2024.

[2] D. Sunaryo, D. Kurnia, Y. Adiyanto, dan I. Quraysin, "Pengaruh Risiko Kredit, Risiko Likuiditas Dan Risiko Operasional Terhadap Profitabilitas Perbankan Pada Bank Umum Di Asia Tenggara Periode 2012-2018," *J. Ilmu Keuang. dan Perbank.*, vol. 11, no. 1, hal. 62–79, 2021, doi: 10.34010/jika.v11i1.3731.

[3] PNM, "PNM Mekaar & PNM Mekaar Syariah," *Pnm*, hal. 1, 2023, [Daring]. Tersedia pada: https://www.pnm.co.id/bisnis/pnm-mekaar

[4] M. Naili dan Y. Lahrichi, "Banks' credit risk, systematic determinants and specific factors: recent evidence from emerging markets," *Heliyon*, vol. 8, no. 2, hal. 1–16, 2022, doi: 10.1016/j.heliyon.2022.e08960.

[5] W. Rohimah, E. W. H. Budianto, dan N. D. T. Dewi, "Pemetaan Penelitian seputar Bank CIMB Niaga Syariah dan Konvensional: Studi Bibliometrik VOSviewer dan Literature Review," *JEMPER (Jurnal Ekon. Manaj. Perbank.)*, vol. 5, no. 1, hal. 30, 2023, doi: 10.32897/jemper.v5i1.2607.

[6] N. Ajeng, B. W. Sari, dan D. Prabowo, "Prediksi Pemberian Kelayakan Pinjaman Dengan Metode Fuzzy Tsukamoto," *Inf. Syst. J.*, vol. 3, no. 1, hal. 19, 2020.

[7] R. H. Situngkir dan P. Sembiring, "Analisis Regresi Logistik Untuk Menentukan Faktor-Faktor YangMempengaruhi Kesejahteraan Masyarakat Kabupaten/Kota Di Pulau Nias," *J. Mat. dan Pendidik. Mat.*, vol. 6, no. 1, hal. 25–31, 2023.

[8] J. Junifer Pangaribuan, H. Tanjaya, dan K. Kenichi, "Mendeteksi Penyakit Jantung Menggunakan Machine Learning Dengan Algoritma Logistic Regression," *J. Inf. Syst. Dev.*, vol. 06, no. 02, hal. 1–10, 2021.

[9] M. R. Adrian, M. P. Putra, M. H. Rafialdy, dan N. A. Rakhmawati, "Perbandingan Metode Klasifikasi Random Forest dan SVM Pada Analisis Sentimen PSBB," *J. Inform. Upgris*, vol. 7, no. 1, hal. 36–40, 2021, doi: 10.26877/jiu.v7i1.7099.

[10] B. Prasojo dan E. Haryatmi, "Analisa Prediksi Kelayakan Pemberian Kredit Pinjaman dengan Metode Random Forest," *J. Nas. Teknol. dan Sist. Inf.*, vol. 7, no. 2, hal. 79–89, 2021, doi: 10.25077/teknosi.v7i2.2021.79-89.