Master Program

KLASIFIKASI TINGKAT PEMBANGUNAN MANUSIA DARI PROVINSI – KABUPATEN - KOTA

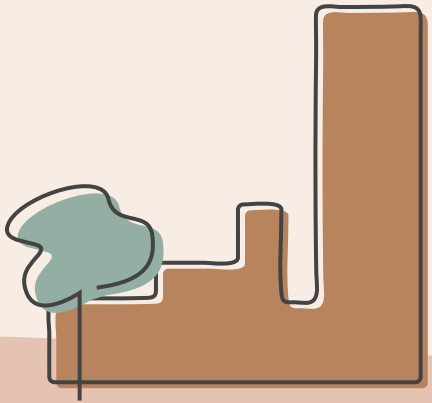
Willy Boen - 2702749733

Isi Konten

01
Pendahuluan

02
Pemahaman Data dan
Pra-Pemrosesan Data

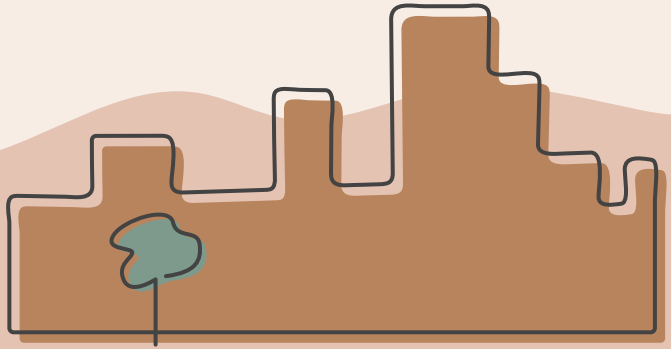
03
Pengembangan
Model dan Evaluasi



04
Kesimpulan

01

Pendahuluan



Latar Belakang

- IPM mengukur kualitas pembangunan melalui tiga dimensi: kesehatan, pendidikan, dan standar hidup.
- Menjadi indikator strategis pemerintah dalam merumuskan kebijakan dan memonitor kesenjangan pembangunan antar wilayah.
- Pemantauan IPM yang akurat diperlukan untuk evaluasi kinerja dan perencanaan pembangunan.
- Meningkatnya ketersediaan data mendorong penggunaan Machine Learning untuk analisis yang lebih efektif.
- ML mampu menangkap pola kompleks yang tidak terdeteksi metode statistik konvensional, sehingga relevan untuk memprediksi atau mengklasifikasikan tingkat IPM.

Tujuan dan Ruang Lingkup

Tujuan	• Mengklasifikasikan tingkat Indeks Pembangunan Manusia (IPM) dengan menggunakan model pembelajaran mesin • Mengevaluasi performa model pembelajaran mesin • Menganalisis pengaruh prapemrosesan data • Mengetahui model mana yang menunjukkan kinerja terbaik • Menyediakan gambaran komparatif
Ruang Lingkup	• Mengolah dataset IPM 2014–2024. • Melatih 7 model ML: Decision Tree, Random Forest, Gradient Boosting, AdaBoost, SVM, MLP, Multinomial Logistic Regression. • Evaluasi menggunakan: Akurasi, Weighted F1-score, Log-loss, Laporan Klasifikasi, dan Matriks Evaluasi Klasifikasi. • Fokus pada klasifikasi multikelas berbasis data tabular dengan metode ML konvensional.

02

Pemahaman Data dan Pra-Pemrosesan Data

Deskripsi dan Eksplorasi Data

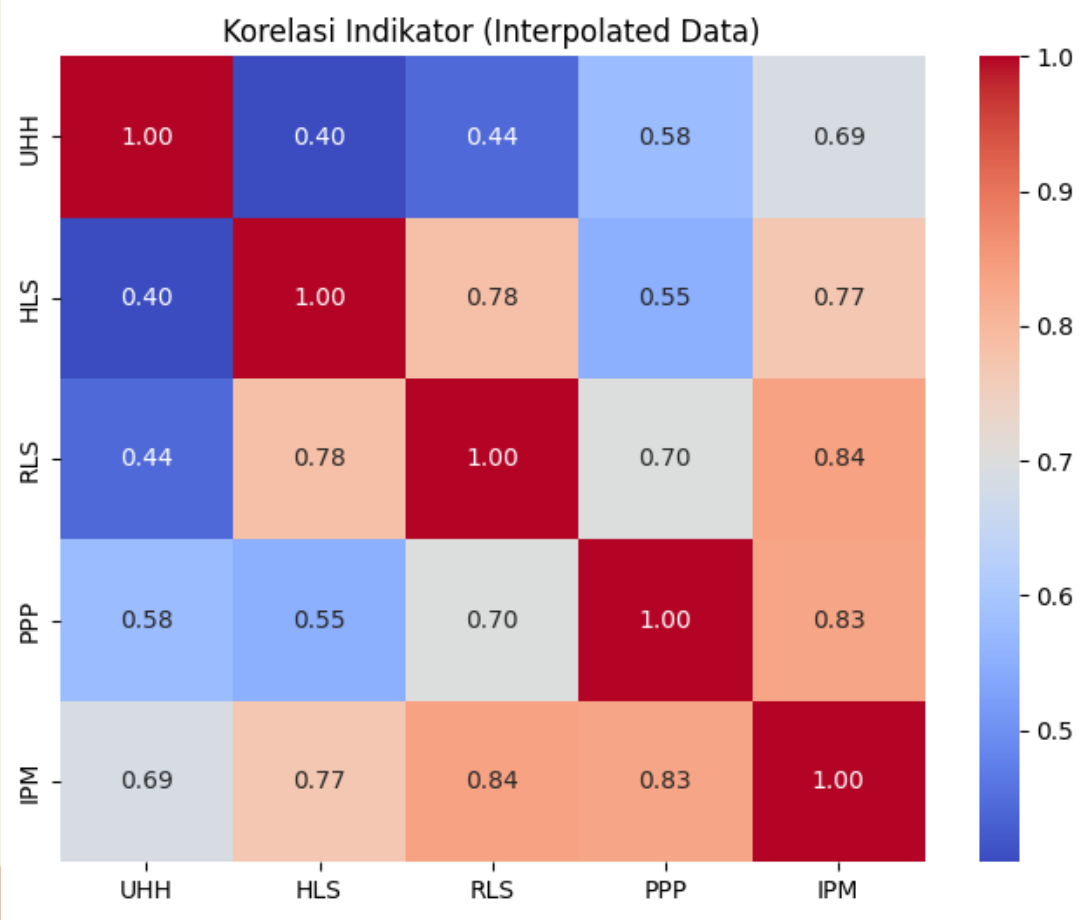
Sumber Data	• Data berasal dari Badan Pusat Statistik (BPS). • Menggunakan indikator: IPM, UHH, RLS, HLS, PPP. • Seluruh variabel mengikuti standar Booklet IPM BPS (kesehatan, pendidikan, standar hidup).
Jenis Data dan Karakteristiknya	• Data sekunder, berbentuk deret waktu tahunan. • Semua variabel bersifat numerik.Terdapat nilai hilang ditandai simbol “–”.Distribusi kelas IPM tidak seimbang setelah pelabelan. • Pemekaran wilayah menyebabkan beberapa daerah tidak memiliki data lengkap.
Ulasan Kualitas Data	• Kualitas data cukup baik • Terdapat nilai hilang karena pemekaran wilayah. • Ketidakseimbangan kelas IPM yang signifikan. • Masalah ditangani dengan: pembersihan data, interpolasi linier, standardisasi, dan oversampling (SMOTE). • Setelah preprocessing, data dianggap layak dan stabil untuk pelatihan model klasifikasi IPM.

Eksplorasi Analisis Data

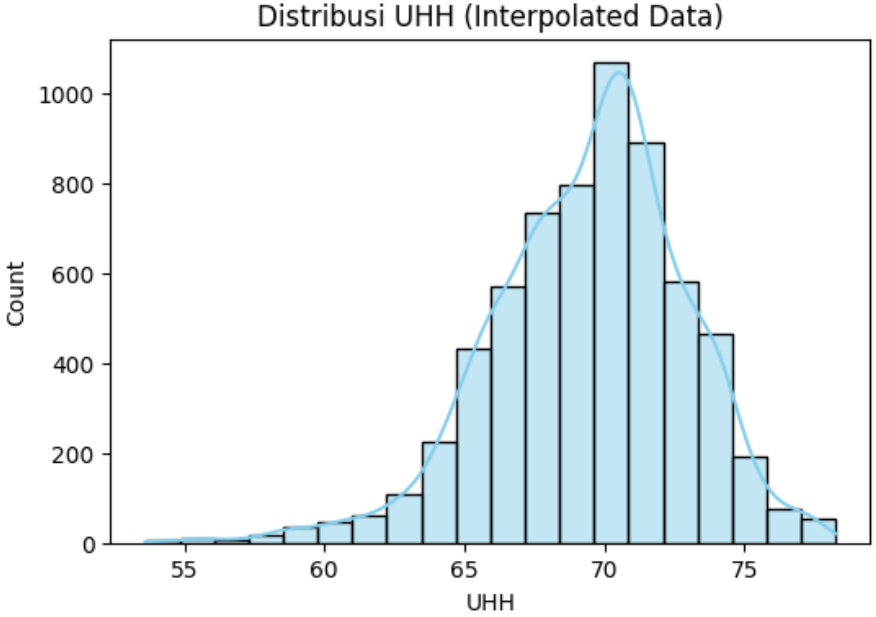
Statistik Deskriptif

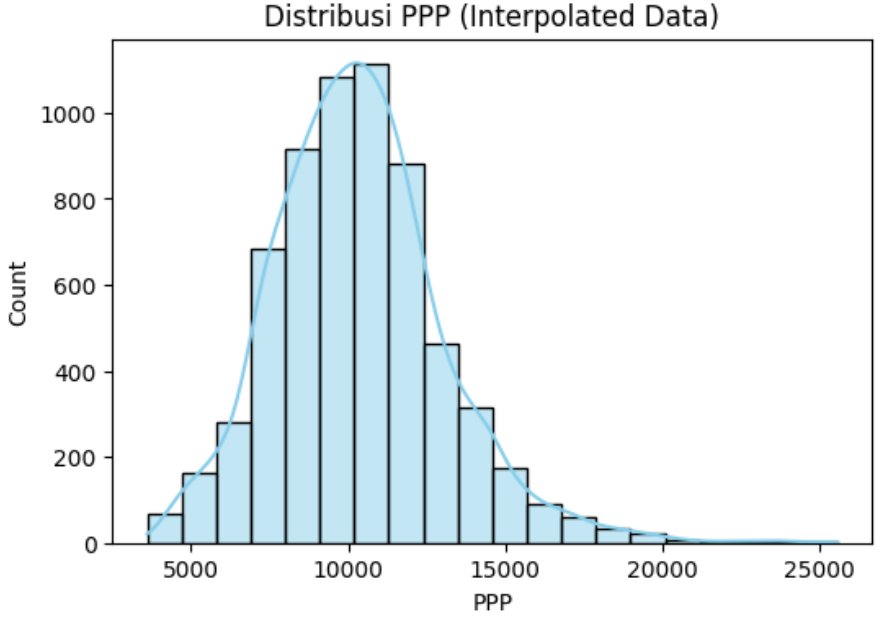
	UHH	HLS	RLS	PPP	IPM
count	6369	6369	6369	6369	6369
Mean	69.448	12.869	8.301	10334.818	69.340
std	3.453	1.350	1.652	2711.682	6.765
min	53.600	2.160	0.630	3607.000	25.380
25%	67.300	12.260	7.340	8515.000	65.900
50%	69.800	12.850	8.190	10191.000	69.360
75%	71.670	13.556	9.300	11818.000	73.040
max	78.260	17.940	13.100	25573.000	88.770

Matriks Korelasi

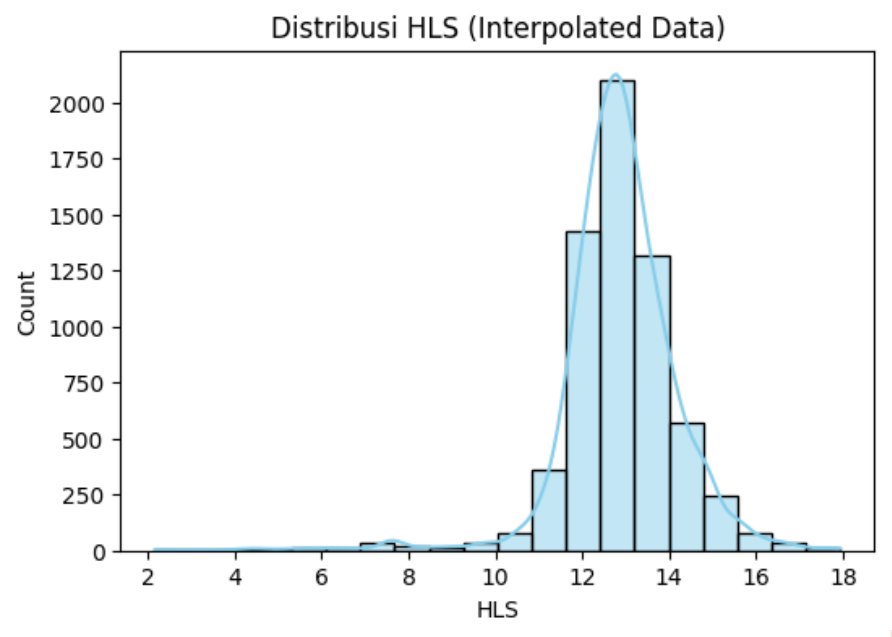


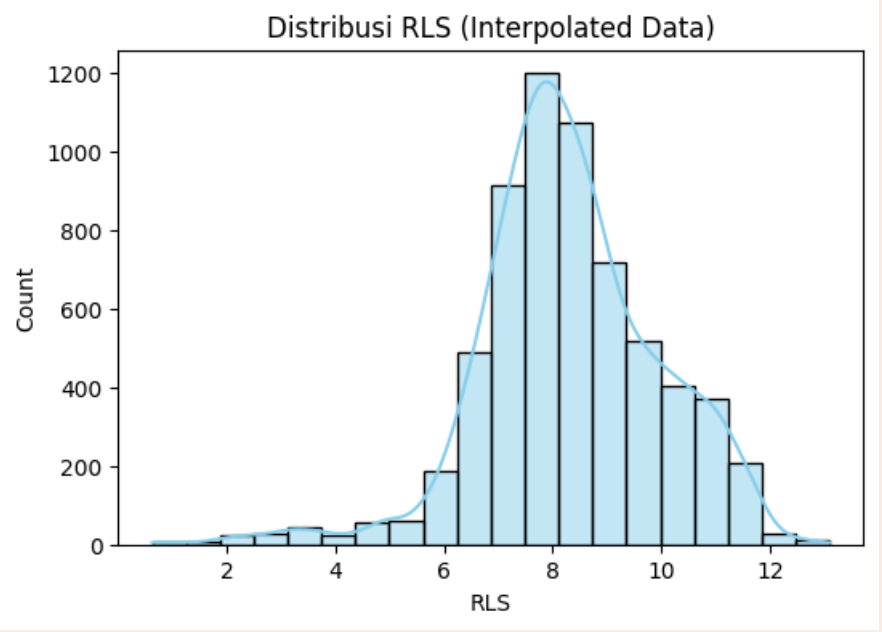
Distribusi per Attribut



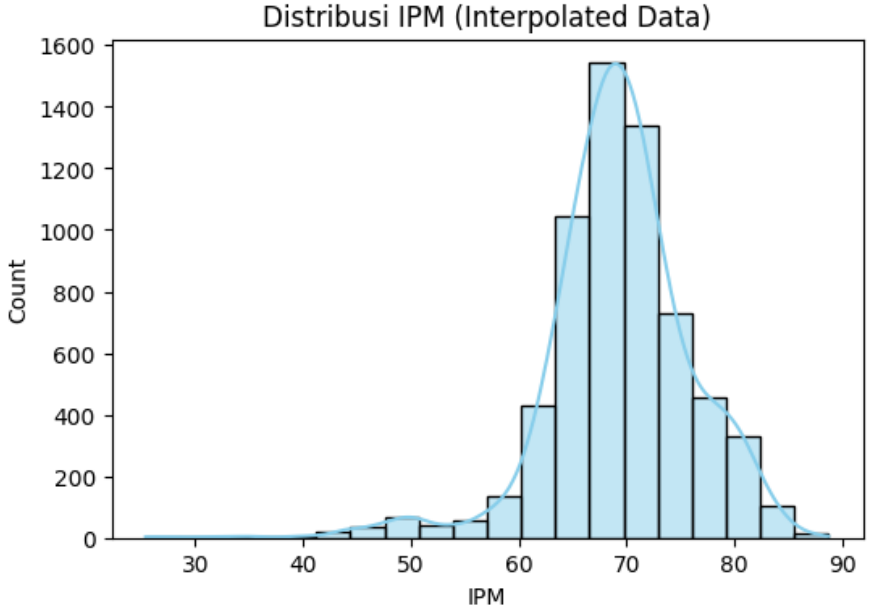


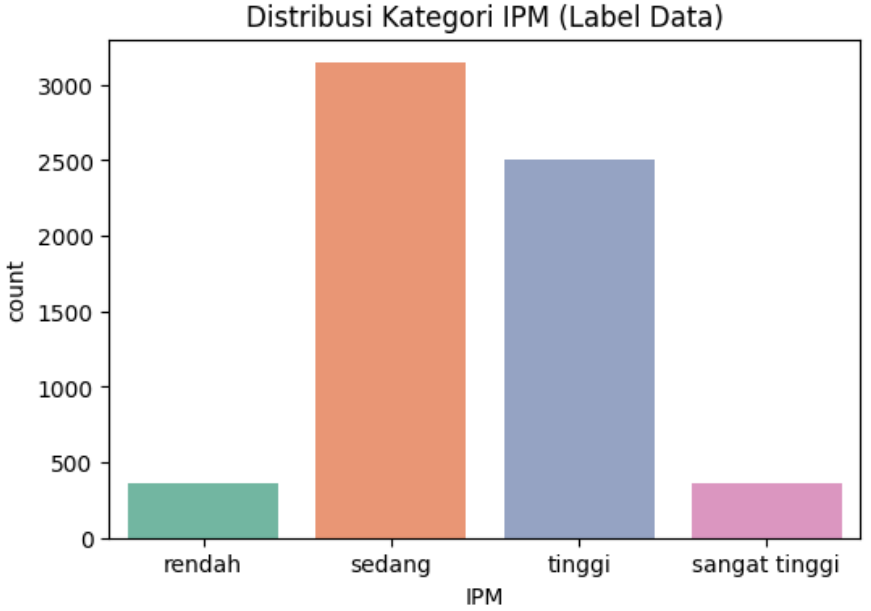
Distribusi per Attribut



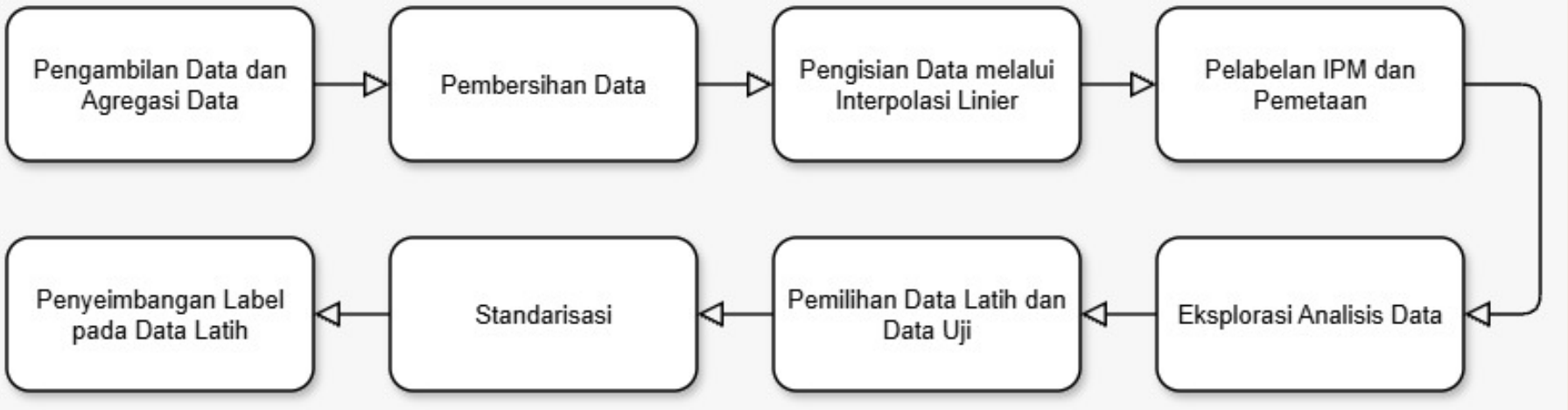


Distribusi per Attribut





Pra-Pemrosesan Data



Penentuan Model

- Model dipilih berdasarkan kemampuan menangani klasifikasi multikelas pada data tabular.
- Menggunakan 7 algoritma: Decision Tree, Random Forest, Gradient Boosting, AdaBoost, SVM, ANN, dan Logistic Regression
- Algoritma pohon (DT, RF, GB) → menangani multikelas secara langsung melalui pemisahan node bertingkat.
- AdaBoost → dapat dipakai pada multikelas dengan penyesuaian bobot kesalahan iteratif.
- SVM → mendukung multikelas lewat one-vs-rest / one-vs-one, efektif dengan kernel RBF untuk data non-linier.
- Logistic Regression → versi multinomial dengan softmax, standar untuk klasifikasi multikelas.
- ANN → menggunakan softmax di output layer, mampu mempelajari hubungan non-linier.

03

Pengembangan Model dan Evaluasi

Arsitektur Model

Decision Tree	• Struktur pohon: root → internal nodes → leaf nodes. • Setiap node memilih fitur terbaik (Gini / Entropy). • Pemisahan dilakukan secara hierarkis hingga mencapai kelas. • Interpretable dan menangani multikelas secara langsung.
Random Forest	• Ensemble banyak Decision Tree. • Menggunakan bootstrap sampling + pemilihan fitur acak. • Prediksi akhir = majority voting antarpohon. • Mengurangi overfitting, mendukung multikelas secara alami.
Gradient Boosting	• Kumpulan pohon secara sekuensial, tiap pohon memperbaiki kesalahan sebelumnya. • Mengoptimalkan loss function melalui gradien. • Untuk multikelas → mengoptimasi loss per kelas dan menggabungkannya. • Mampu menangkap pola kompleks.

Arsitektur Model

AdaBoost	• Terdiri dari weak learners (biasanya decision stump). • Dilatih berurutan dengan penyesuaian bobot kesalahan. • Multikelas menggunakan SAMME / SAMME.R. • Memperkuat prediksi pada area sulit dipelajari.
Support Vector Machine (SVM)	• Mencari hyperplane dengan margin maksimal. • Multikelas melalui One-Vs-Rest atau One-Vs-One. • Kernel (misal RBF) untuk data non-linier. • Efektif untuk pemisahan kelas berdimensi tinggi.

Arsitektur Model

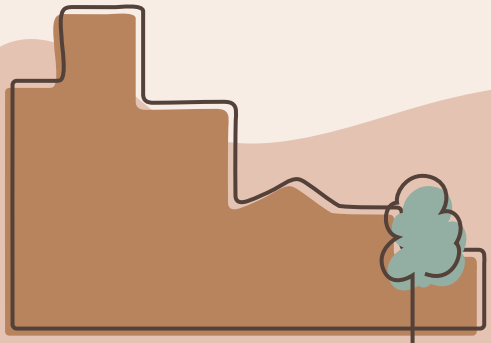
Artificial Neural Network (ANN)	• Terdiri dari: input layer → hidden layers → output layer. • Hidden layer memakai aktivasi ReLU (atau aktivasi non-linear lain). • Output layer memakai softmax untuk multikelas. • Belajar pola melalui feed-forward + backpropagation.
Logistic Regression	• Fungsi linear: $z = W_x + b$. • Output menggunakan softmax menghasilkan probabilitas tiap kelas. • Setiap kelas memiliki bobot sendiri. • Stabil, sederhana, dan interpretatif untuk multikelas.

Pelatihan Model

- 7 model dilatih: Decision Tree, Random Forest, Gradient Boosting, AdaBoost, SVM, ANN, Logistic Regression.
- Decision Tree: membentuk pohon hierarkis berdasarkan fitur terbaik.
- Random Forest / Gradient Boosting: metode ensemble meningkatkan akurasi.
- AdaBoost: menggunakan SAMME, bobot weak learner tergantung tingkat kesalahan.
- SVM: kernel RBF + pendekatan One-Vs-Rest untuk 4 kelas.
- ANN: 2 hidden layer (64 & 32 neuron), ReLU, softmax output, dilatih dengan backpropagation.
- Logistic Regression: versi multinomial, softmax + cross-entropy loss.

Hyperparameter Tuning* atau *Fine Tuning

Tidak ada yang digunakan



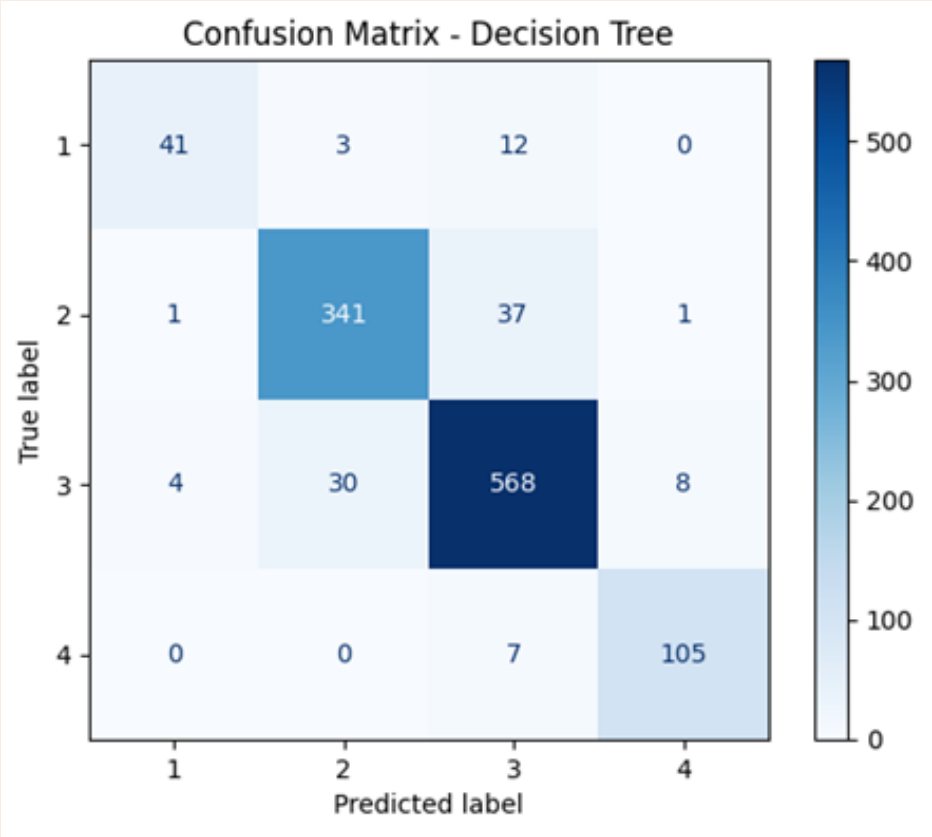
Hasil Evaluasi Model

Laporan Klasifikasi	Menampilkan metrik per kelas untuk melihat performa detail pada masing-masing kategori IPM, termasuk kemampuan model mendeteksi kelas minoritas.
Matriks Evaluasi Klasifikasi	Menunjukkan distribusi prediksi benar–salah untuk tiap kelas (rendah, sedang, tinggi, sangat tinggi). Memudahkan melihat pola kesalahan, kelas yang sering tertukar, dan kemampuan model membedakan tiap kategori.
Accuracy	Mengukur proporsi prediksi yang benar dari seluruh sampel data uji. Memberikan gambaran umum performa model, tetapi kurang efektif saat kelas tidak seimbang.
Weighted F1-score	Menilai keseimbangan precision dan recall untuk setiap kelas dengan mempertimbangkan proporsi kelas. Cocok digunakan ketika dataset imbalanced, sehingga kelas minoritas tetap diperhitungkan.
Log-loss	Mengukur kualitas prediksi probabilitas. Semakin kecil nilainya, semakin baik model dalam memberikan probabilitas yang akurat. Sensitif terhadap kesalahan prediksi pada model probabilistik.

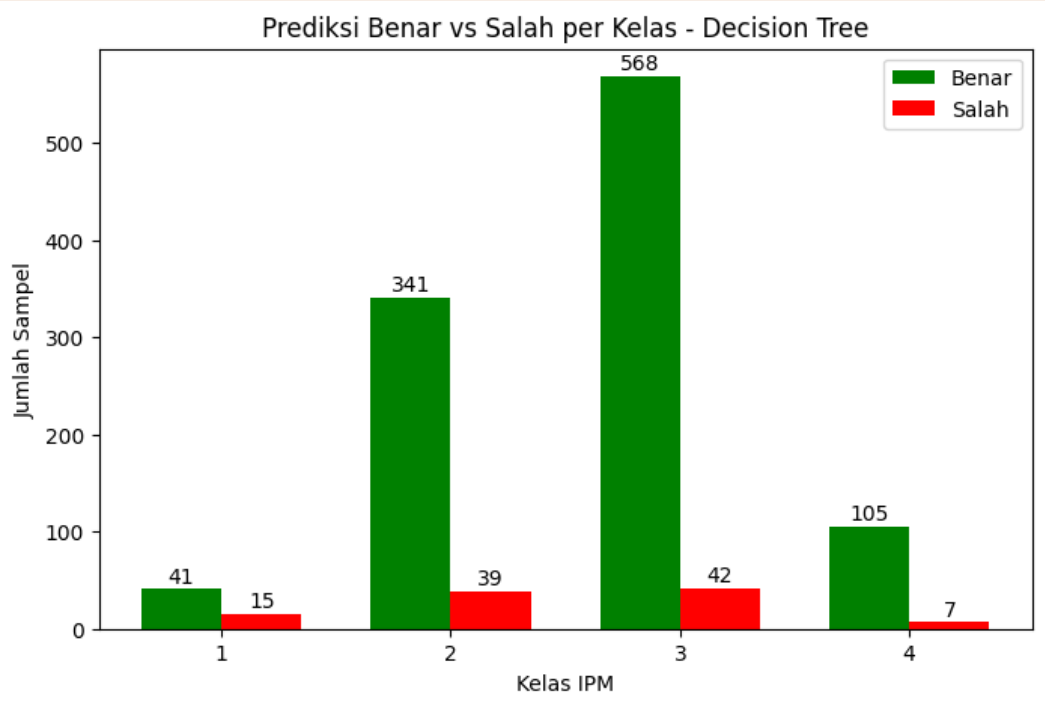
Analisis Evaluasi Model

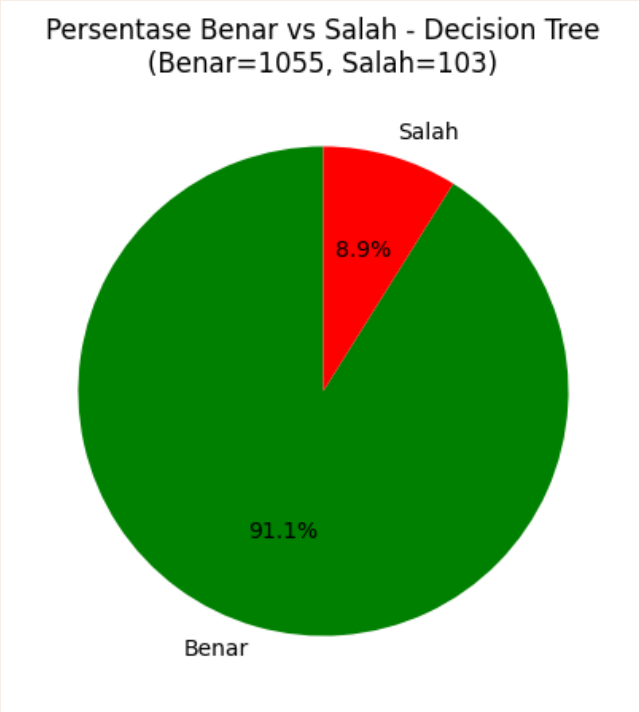
Decision Tree: SMOTE

Kelas	SMOTE		
	Presisi	Recall	F1-Score
1	0.891	0.732	0.804
2	0.912	0.897	0.905
3	0.910	0.931	0.921
4	0.921	0.938	0.929



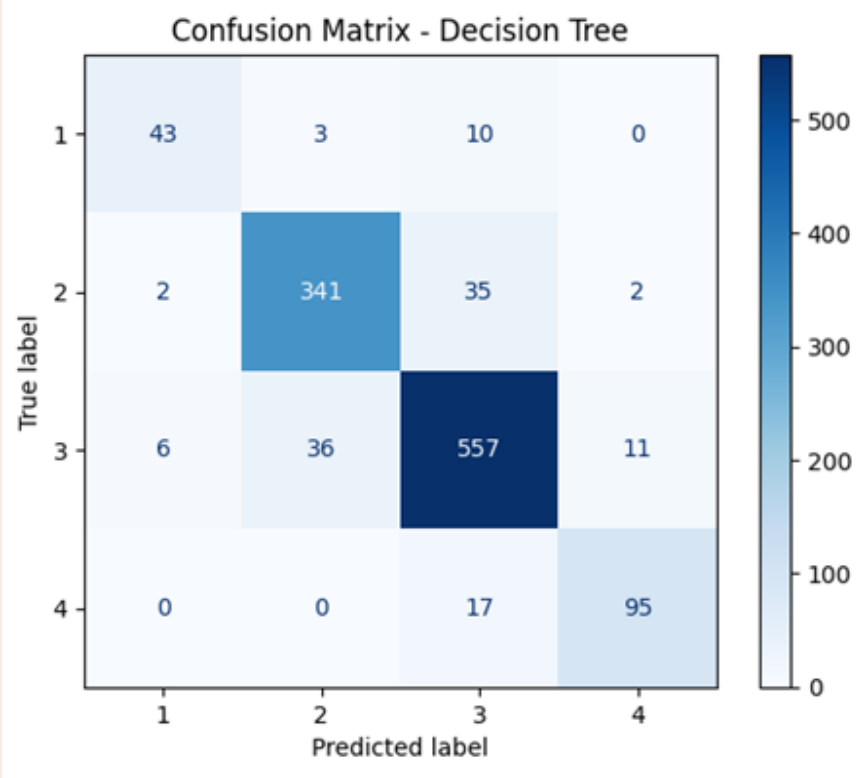
Decision Tree: SMOTE



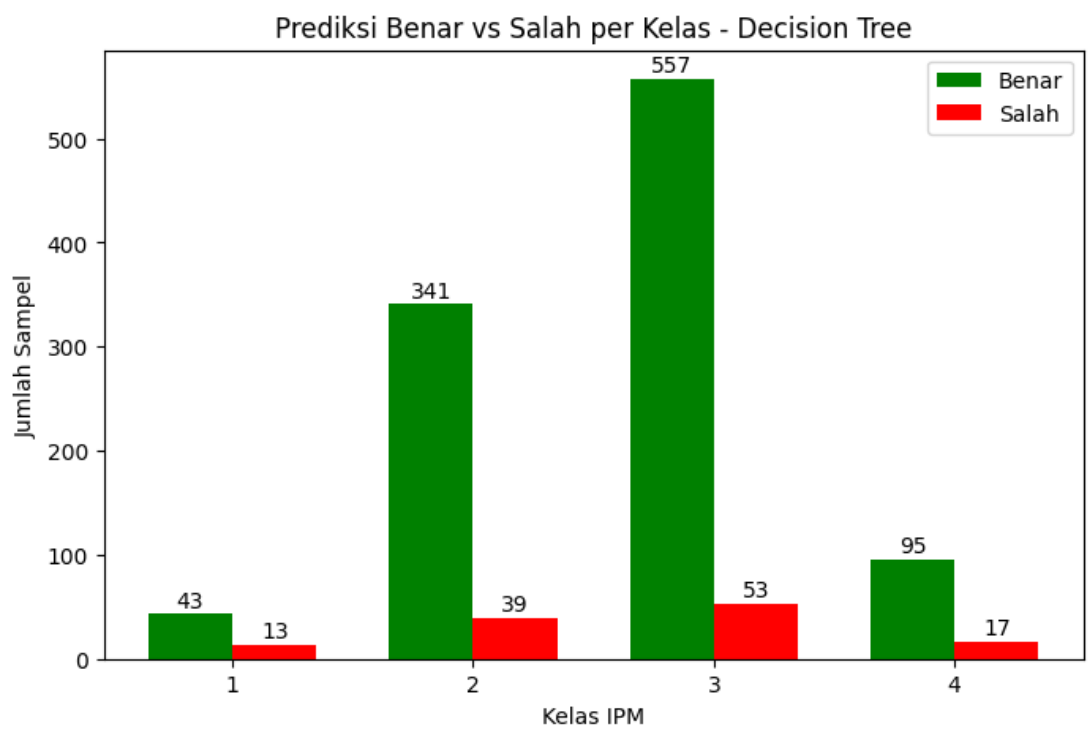


Decision Tree: Non-SMOTE

Kelas	Non-SMOTE		
	Presisi	Recall	F1-Score
1	0.843	0.768	0.804
2	0.897	0.897	0.897
3	0.900	0.913	0.906
4	0.880	0.848	0.864

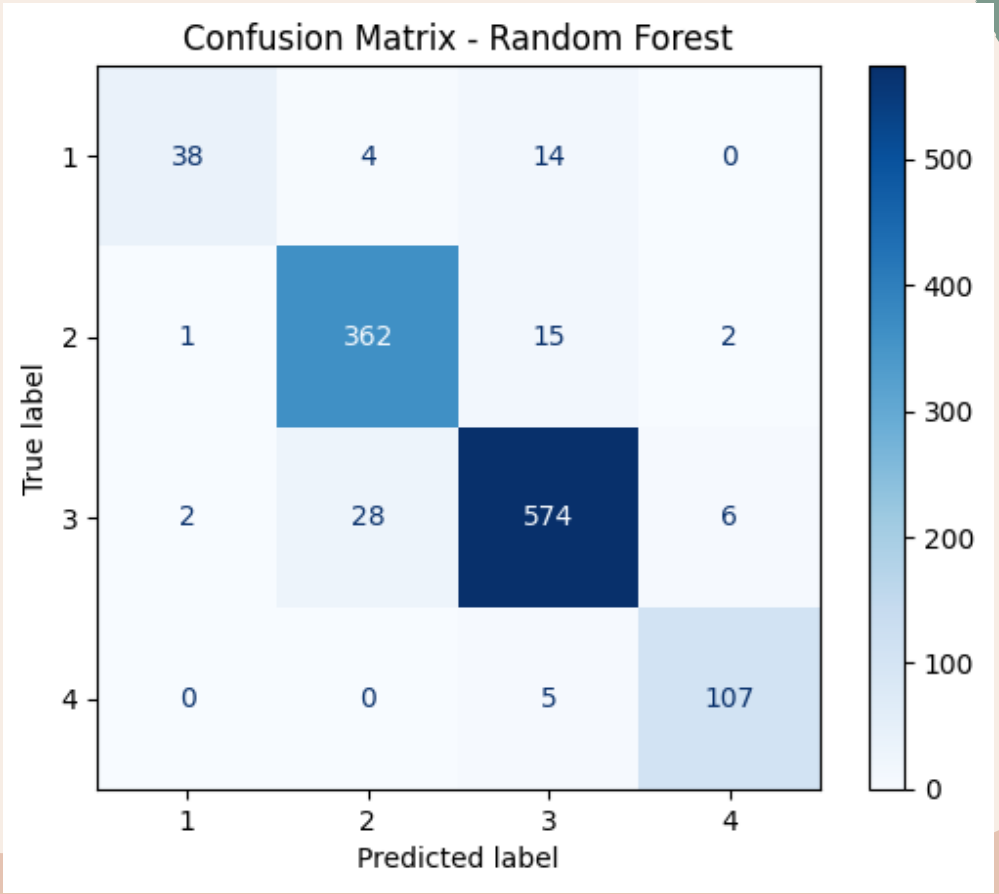


Decision Tree: Non-SMOTE

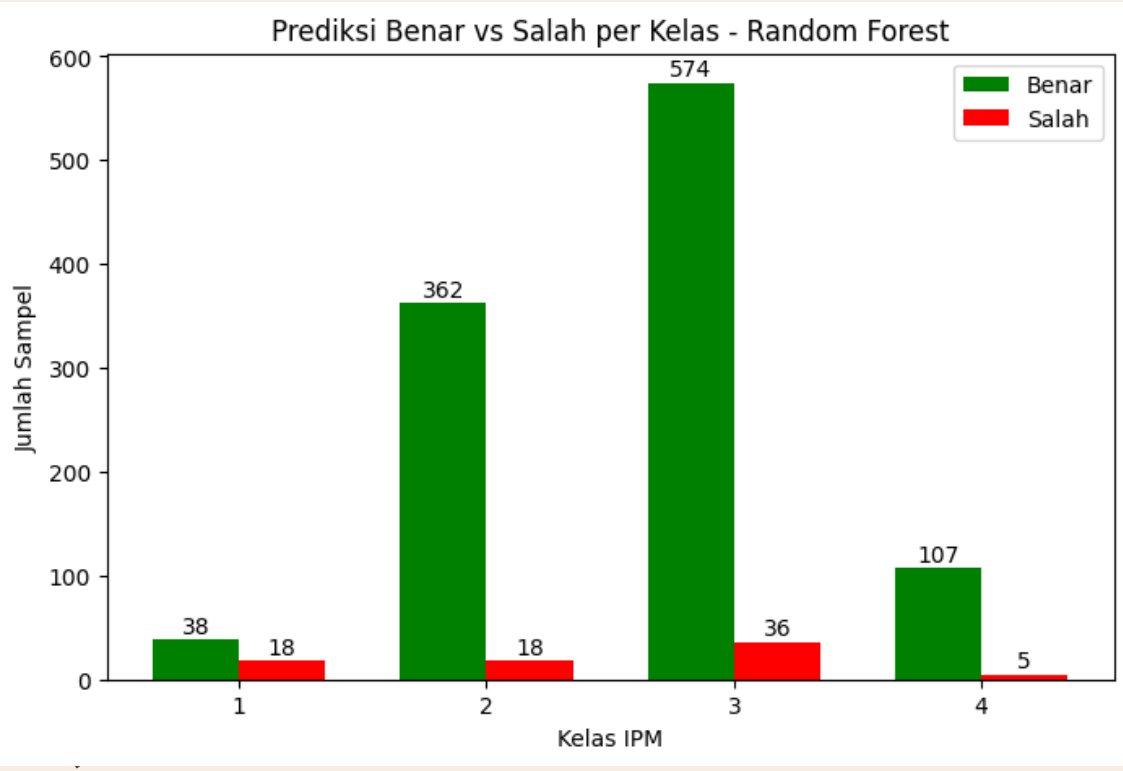


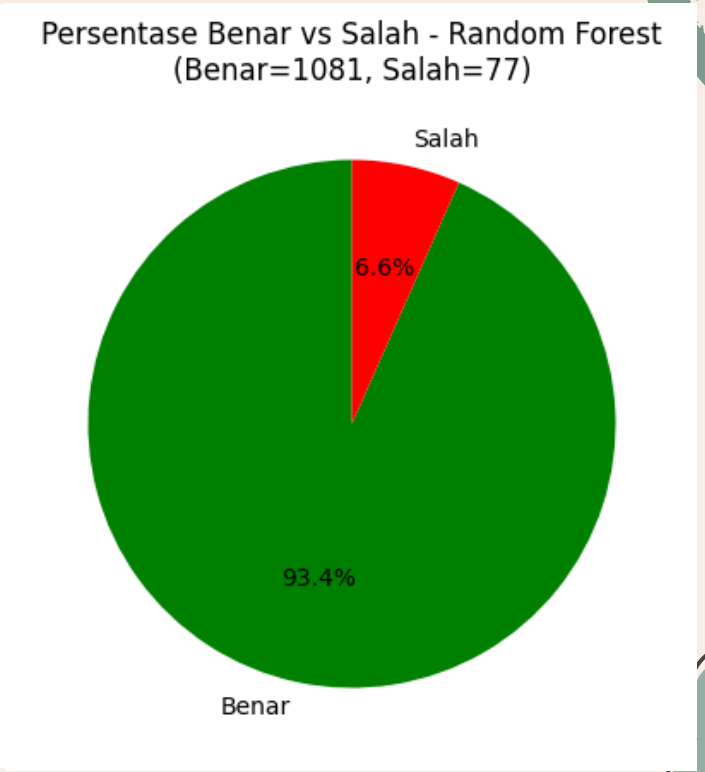
Random Forest: SMOTE

Kelas	SMOTE		
	Presisi	Recall	F1-Score
1	0.927	0.679	0.784
2	0.919	0.953	0.935
3	0.944	0.941	0.943
4	0.955	0.955	0.943



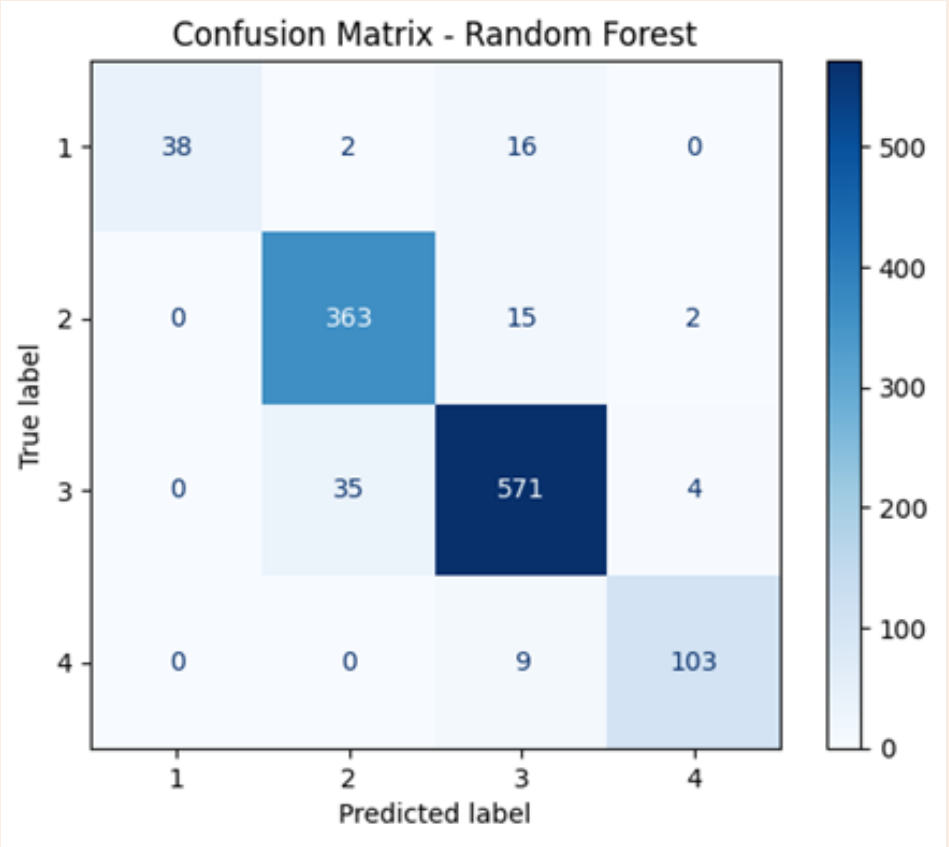
Random Forest: SMOTE



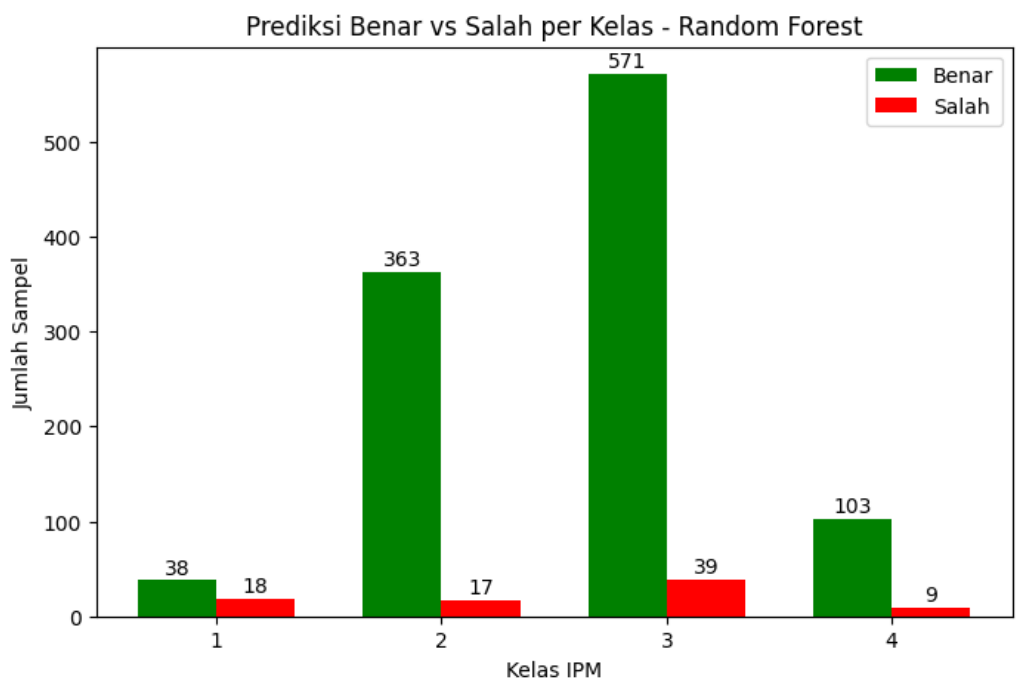


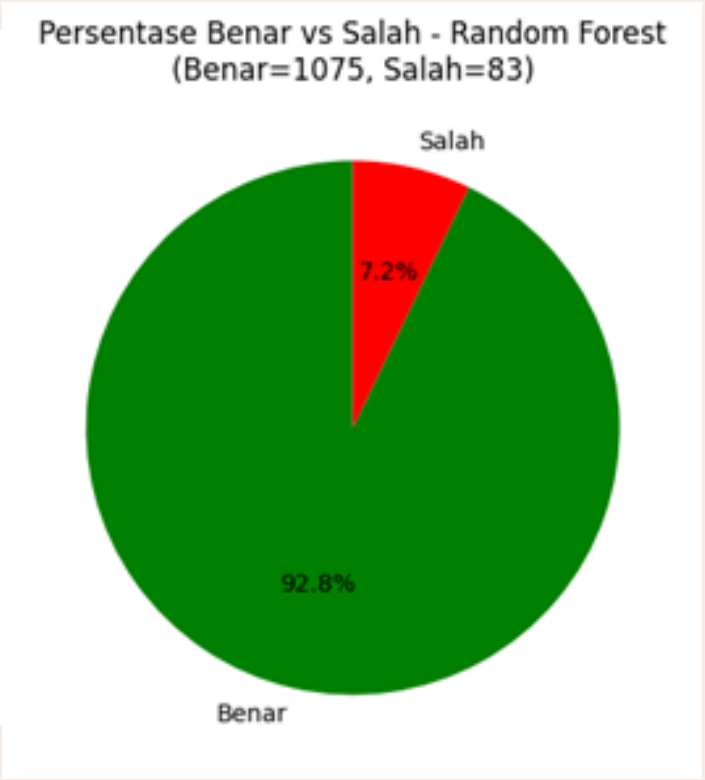
Random Forest: Non-SMOTE

Kelas	Non-SMOTE		
	Presisi	Recall	F1-Score
1	1.000	0.679	0.809
2	0.907	0.955	0.931
3	0.935	0.936	0.935
4	0.945	0.920	0.932



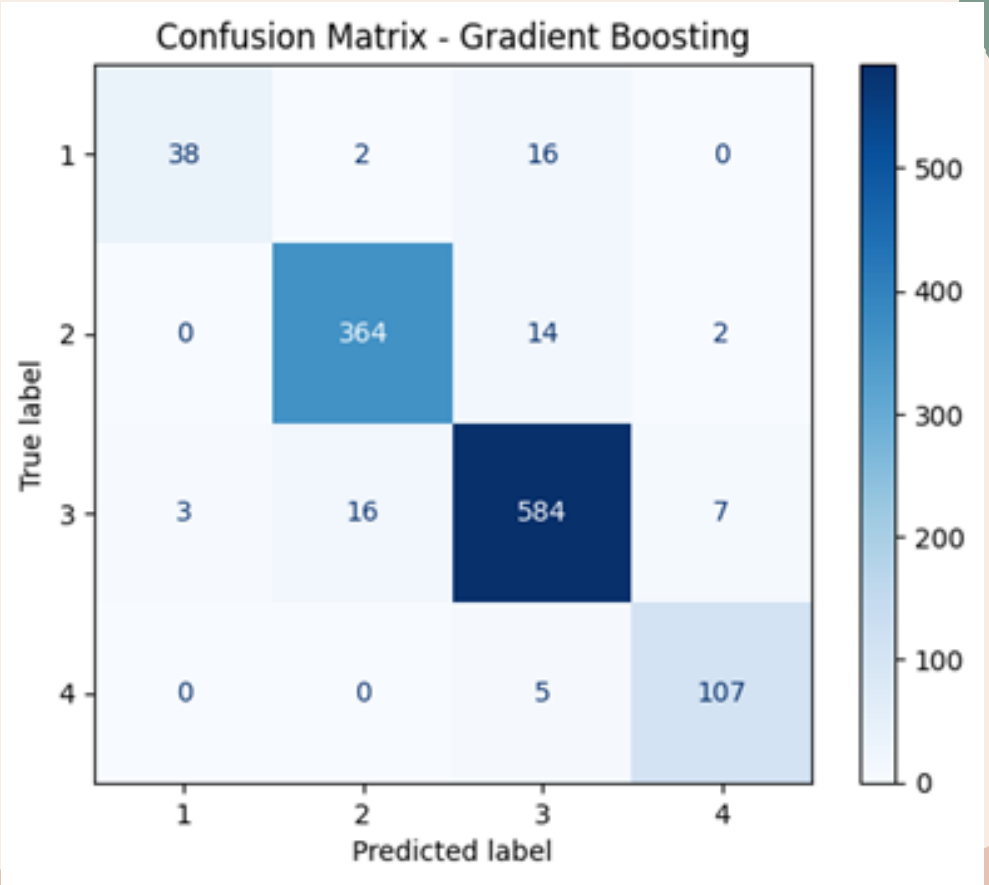
Random Forest: Non-SMOTE



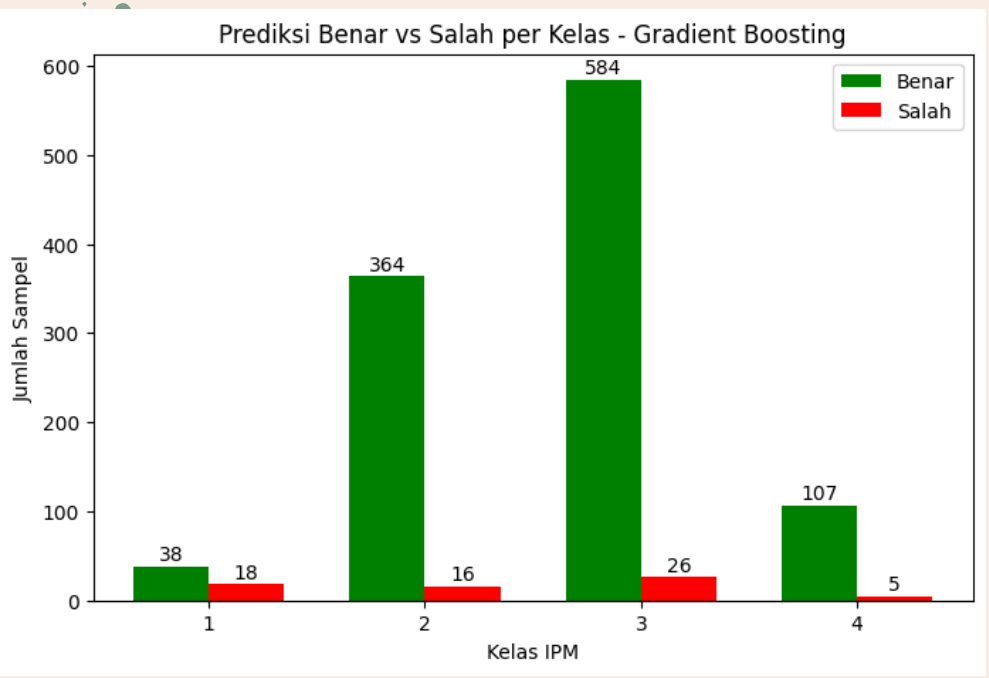


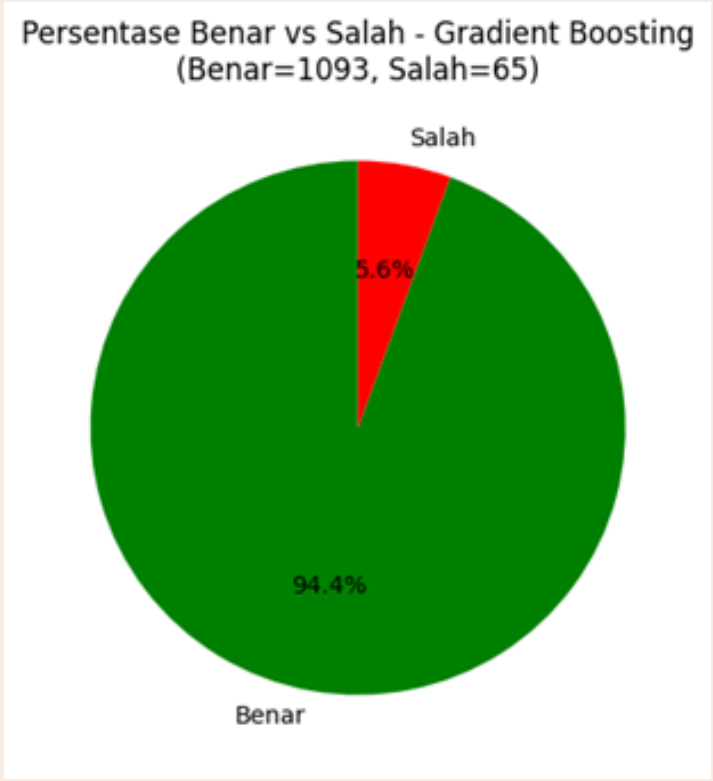
Gradient Boosting: SMOTE

Kelas	SMOTE		
	Presisi	Recall	F1-Score
1	0.927	0.679	0.784
2	0.953	0.958	0.955
3	0.943	0.957	0.950
4	0.922	0.955	0.939



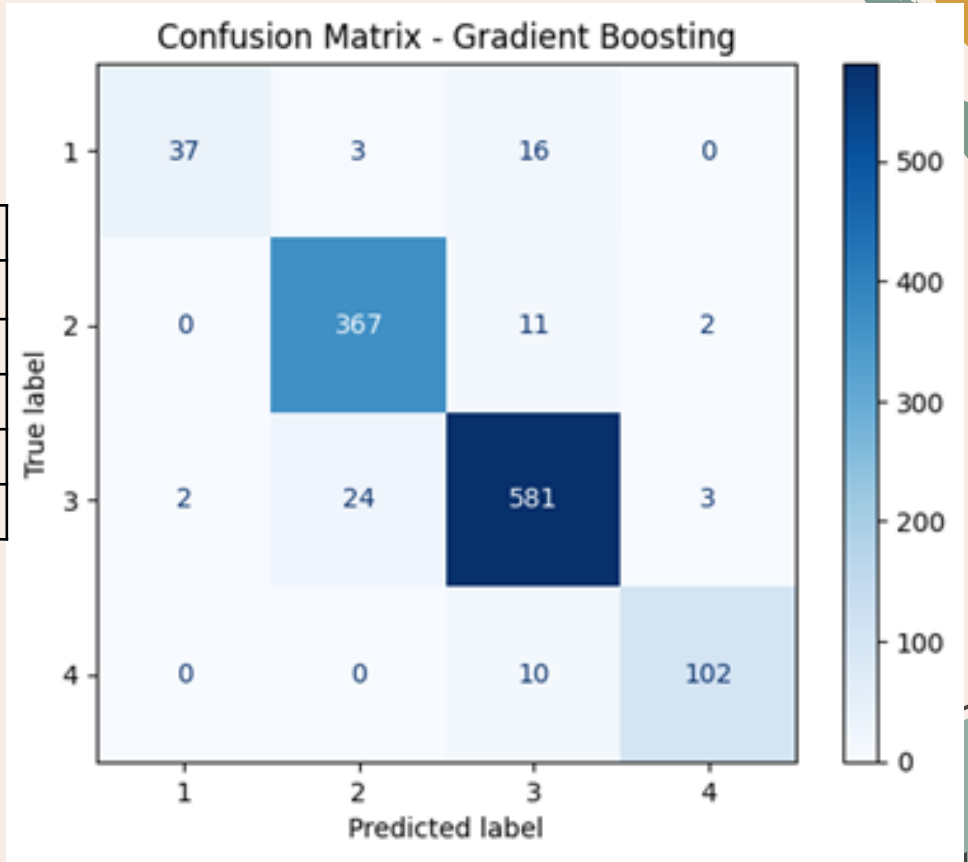
Gradient Boosting: SMOTE



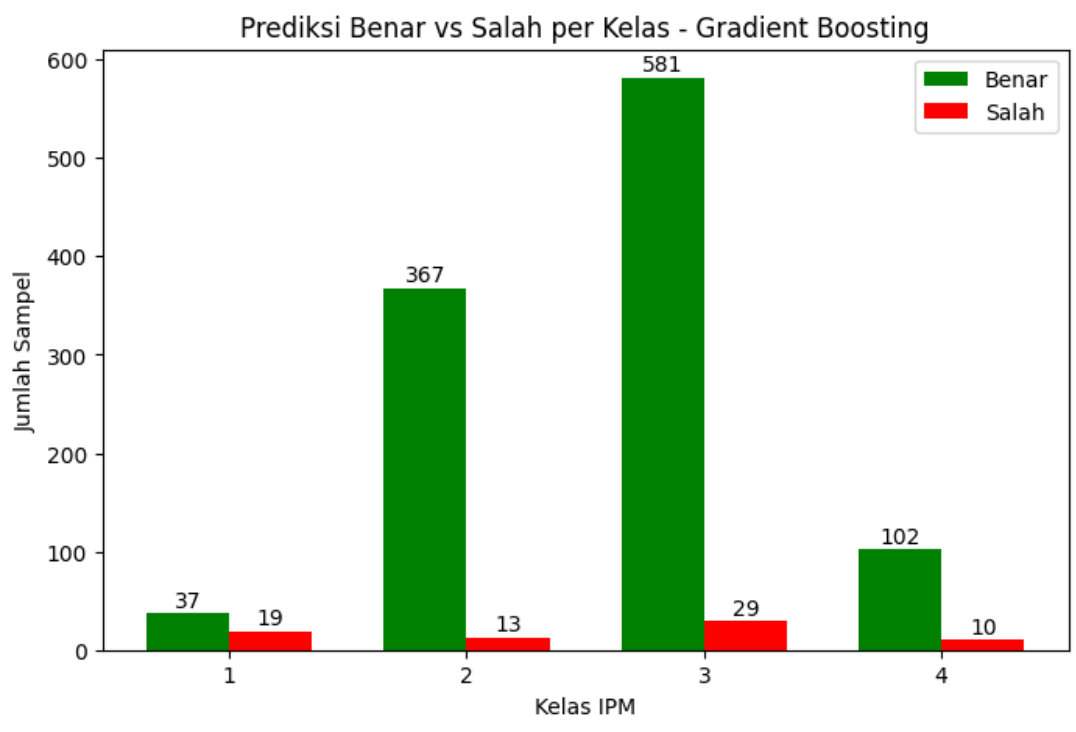


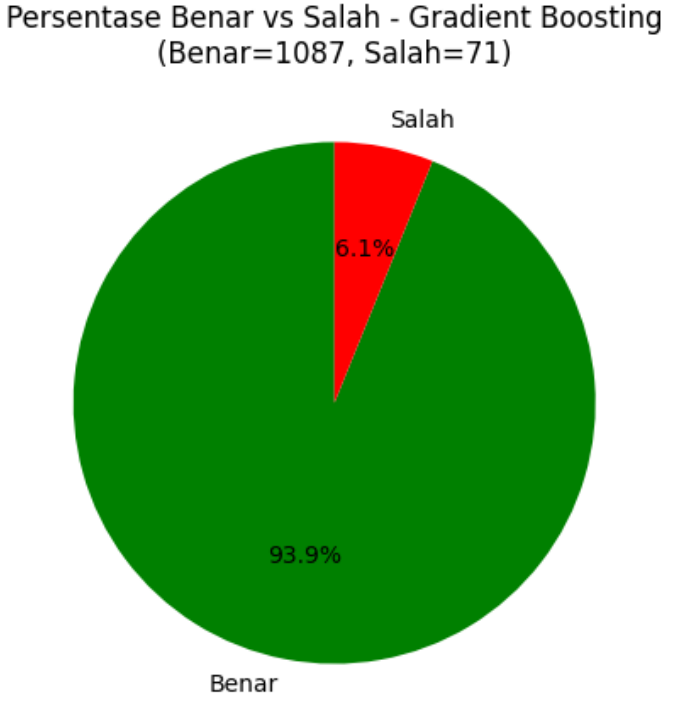
Gradient Boosting: Non-SMOTE

Kelas	Non-SMOTE		
	Presisi	Recall	F1-Score
1	0.949	0.661	0.779
2	0.931	0.966	0.948
3	0.940	0.952	0.946
4	0.953	0.911	0.932



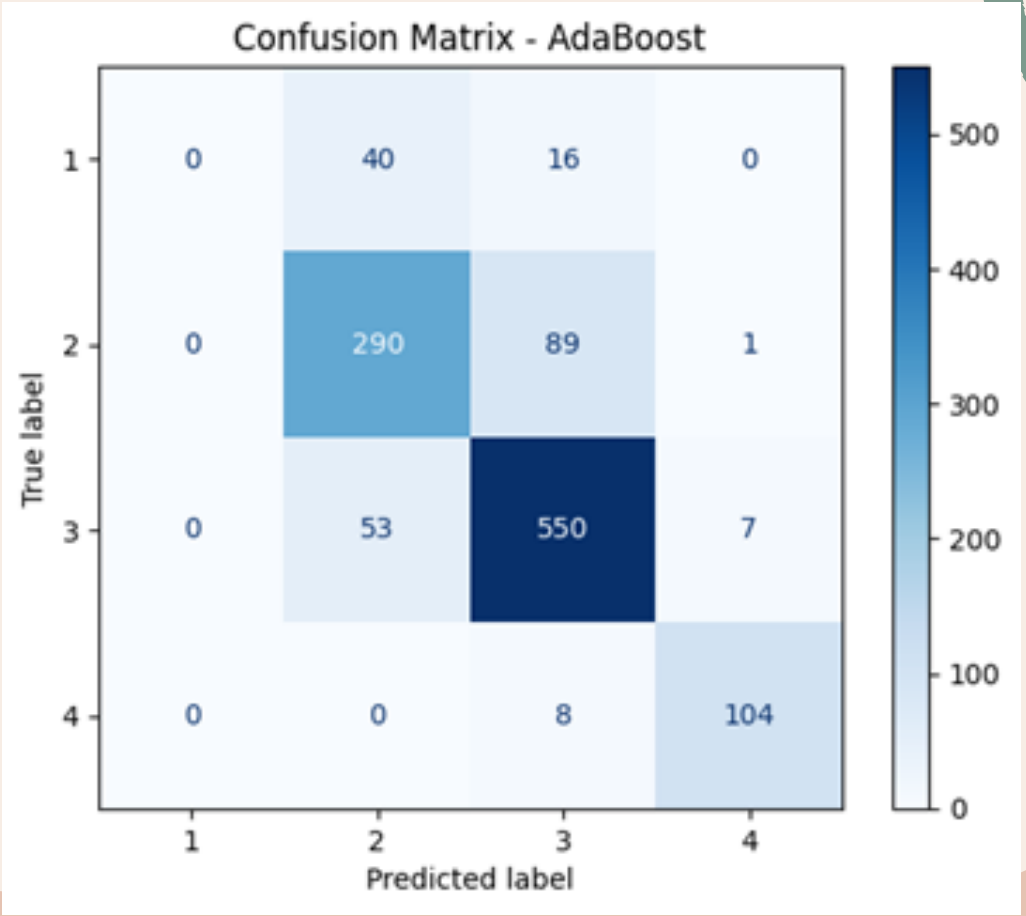
Gradient Boosting: Non-SMOTE



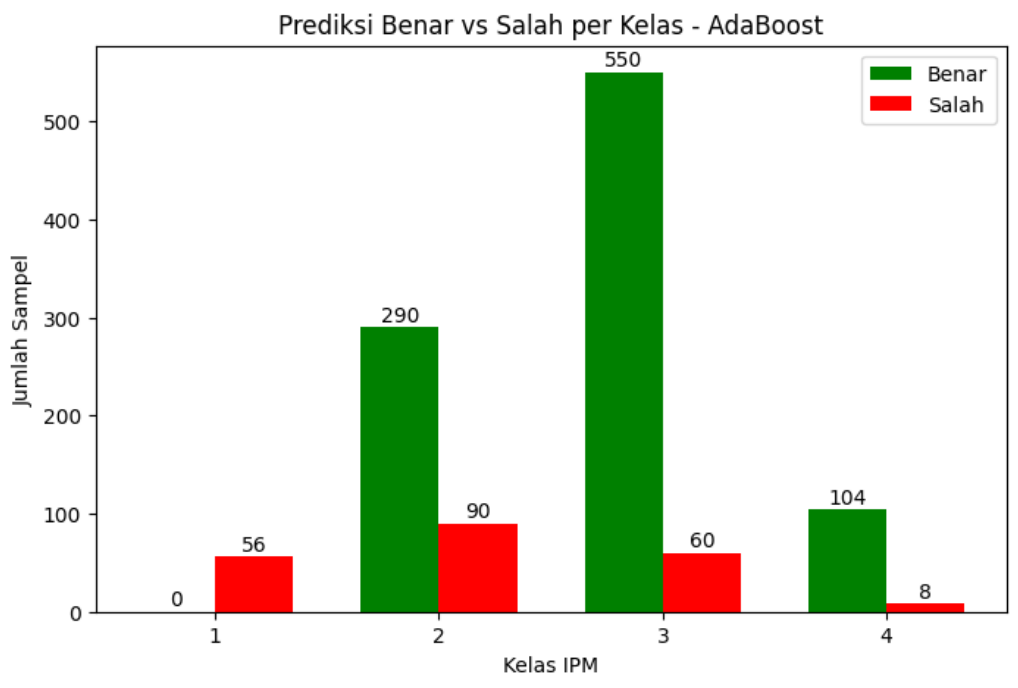


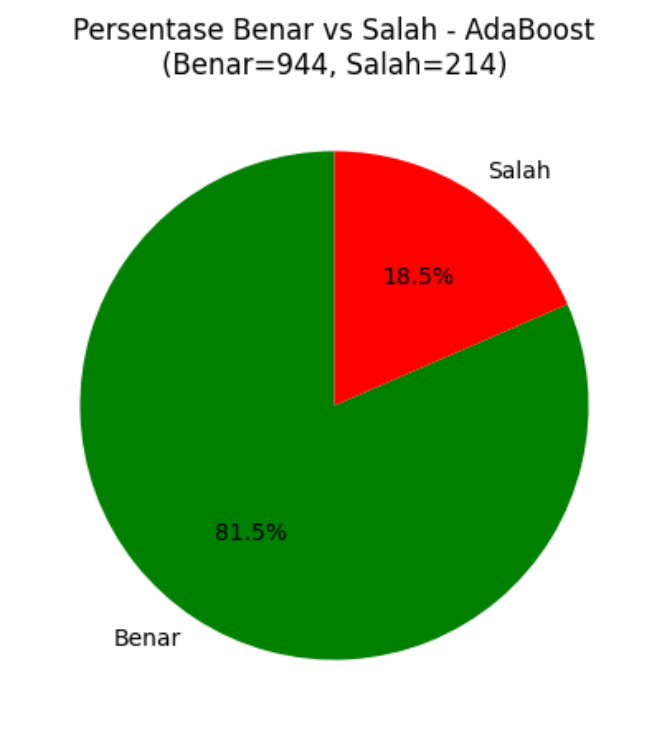
AdaBoost: SMOTE

Kelas	SMOTE		
	Presisi	Recall	F1-Score
1	0.000	0.000	0.000
2	0.757	0.763	0.760
3	0.830	0.902	0.864
4	0.929	0.929	0.929



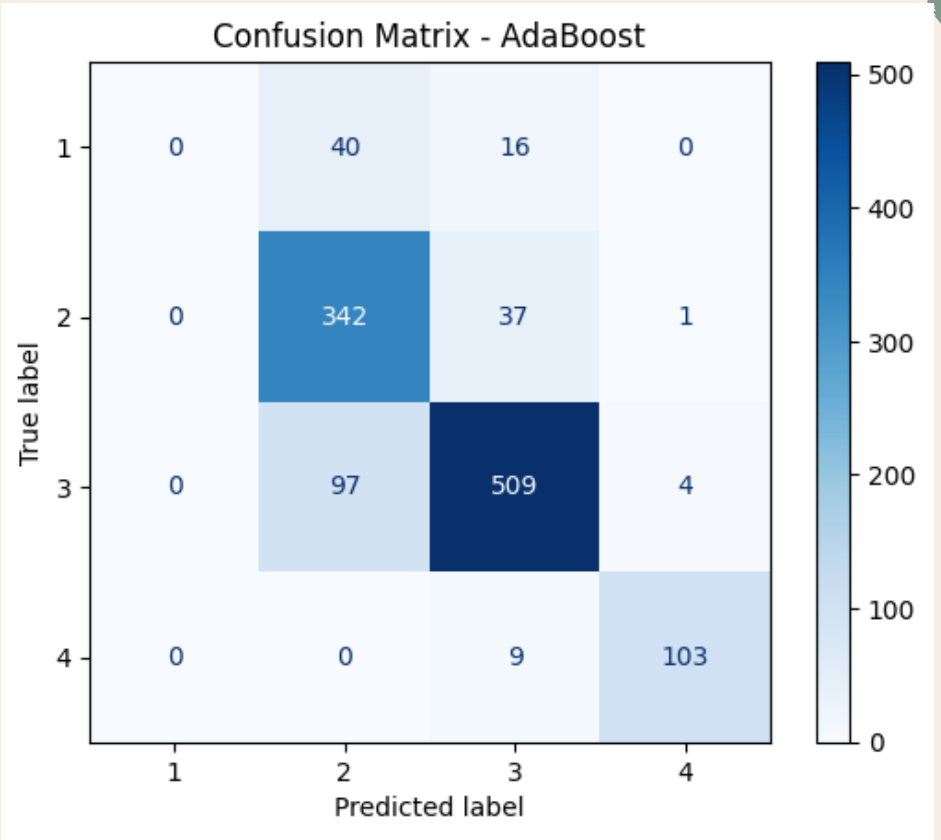
AdaBoost: SMOTE



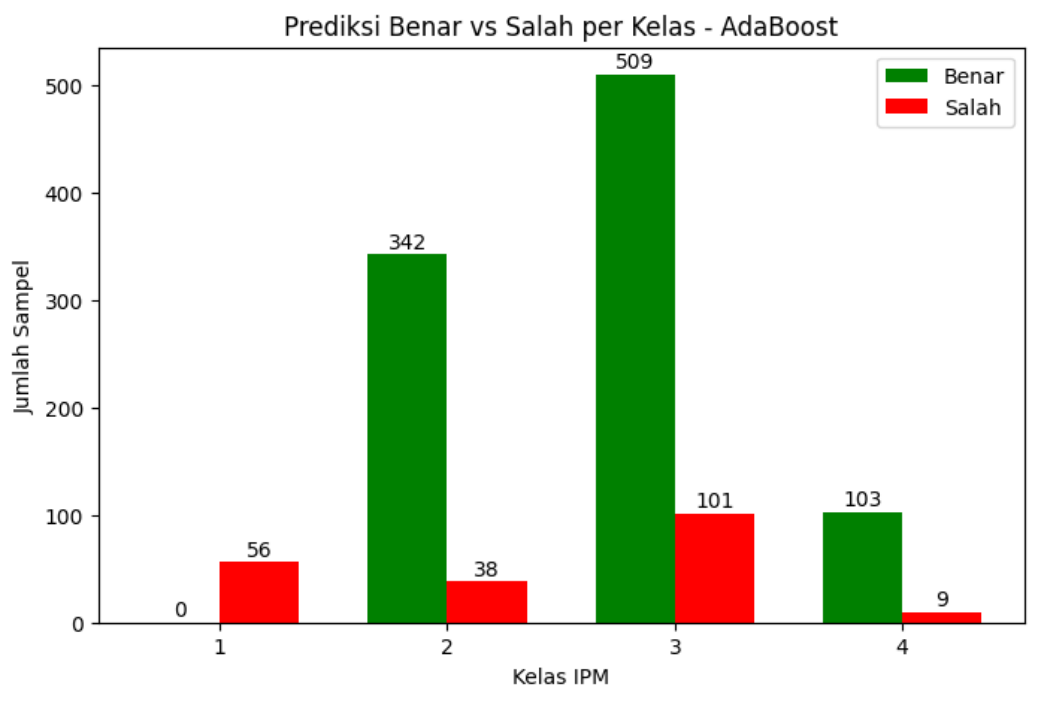


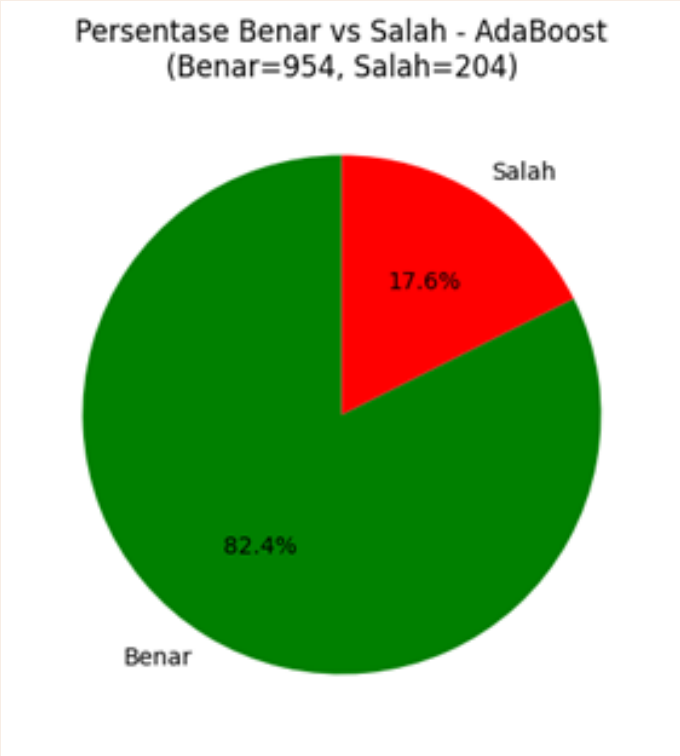
AdaBoost: Non-SMOTE

Kelas	Non-SMOTE		
	Presisi	Recall	F1-Score
1	0.000	0.000	0.000
2	0.714	0.900	0.796
3	0.891	0.834	0.862
4	0.954	0.920	0.936



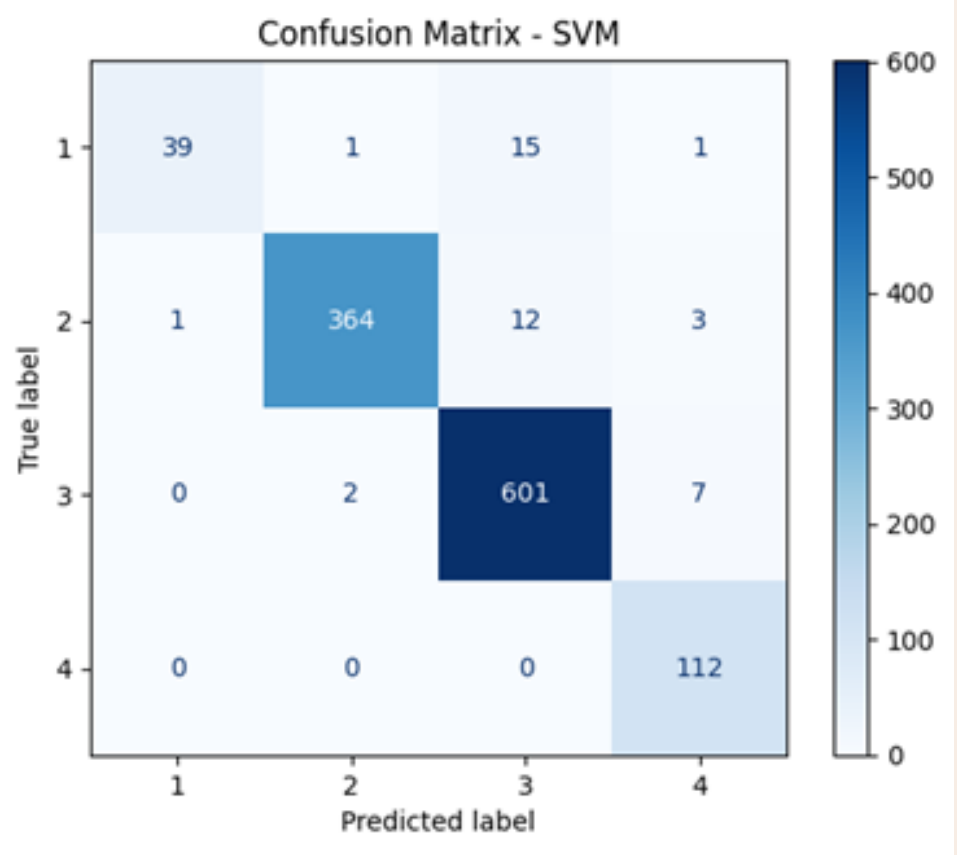
AdaBoost: Non-SMOTE



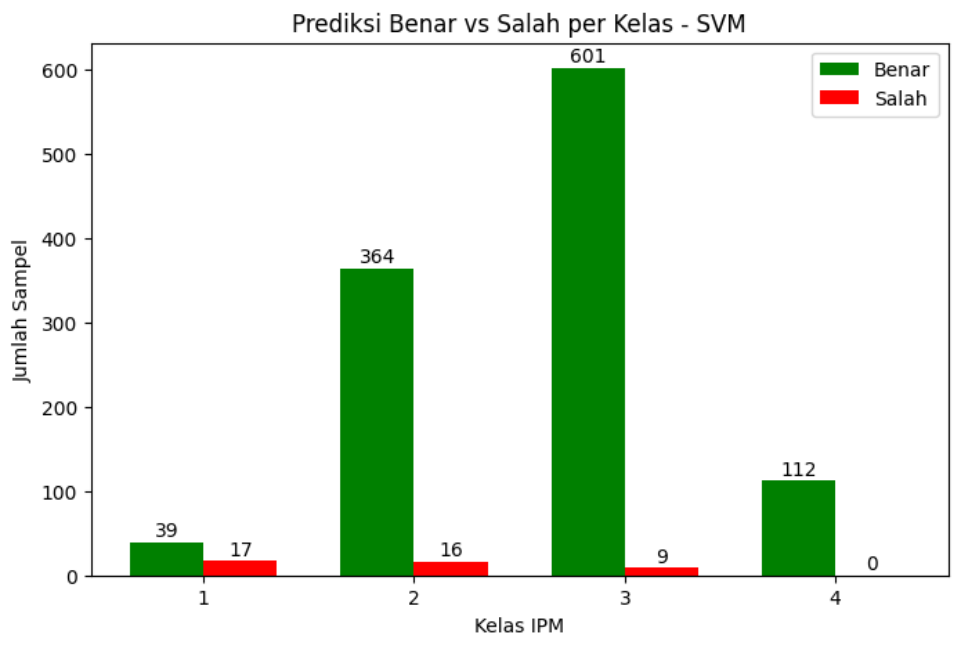


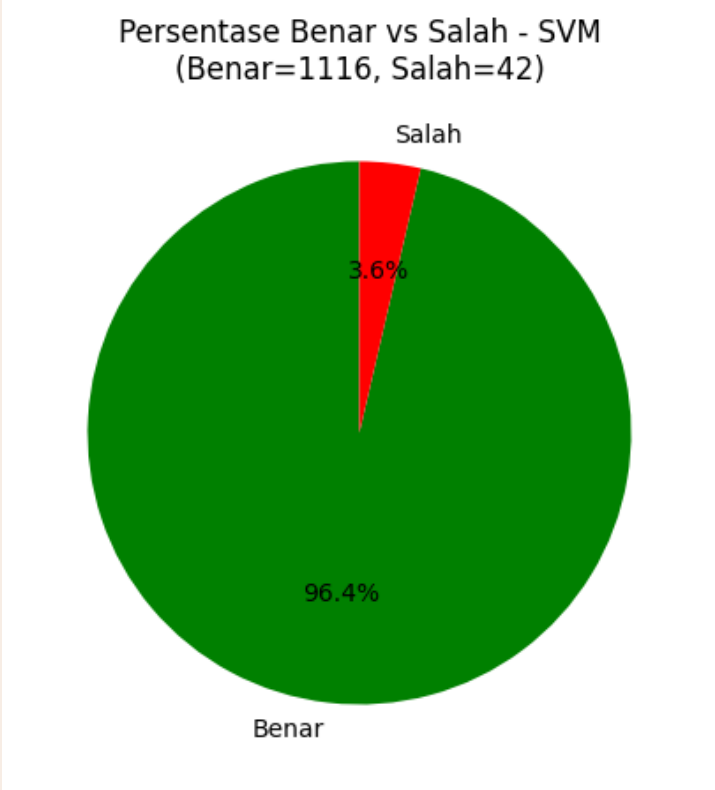
SVM: SMOTE

Kelas	SMOTE		
	Presisi	Recall	F1-Score
1	0.975	0.696	0.812
2	0.992	0.958	0.975
3	0.957	0.985	0.971
4	0.911	1.000	0.953



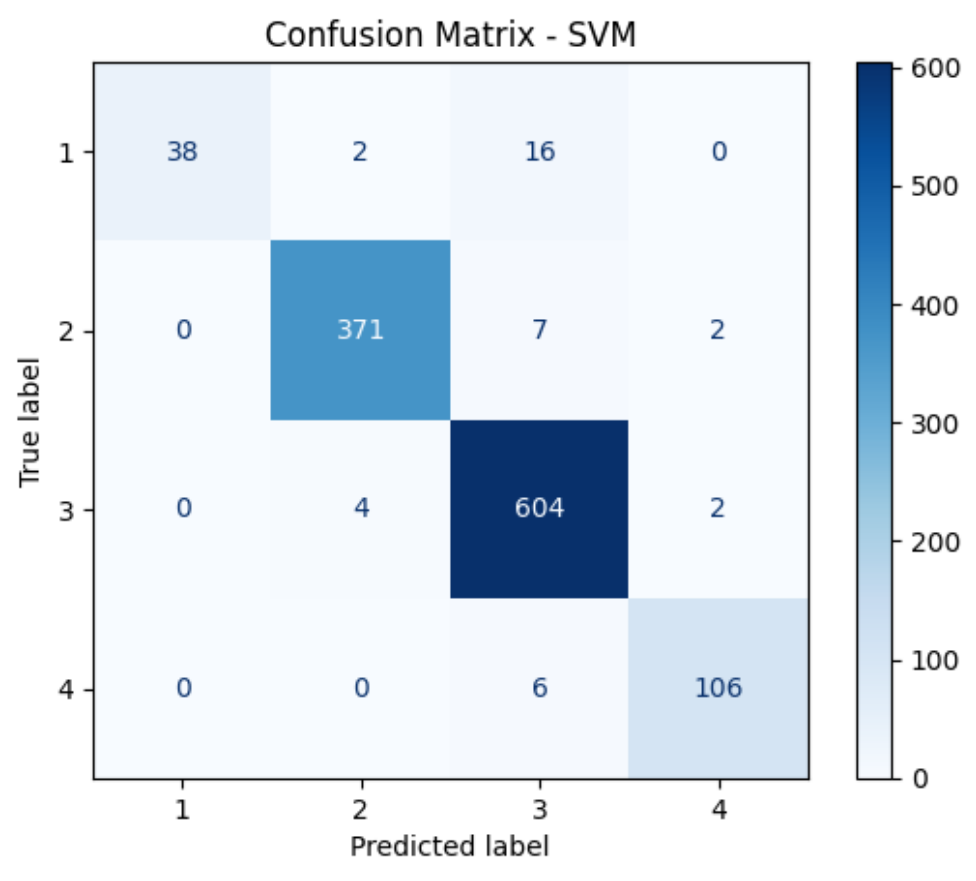
SVM: SMOTE


Prediksi Benar vs Salah per Kelas - SVM
Benar
Salah
Jumlah Sampel
Kelas IPM
0
100
200
300
400
500
600
1
2
3
4
39
17
364
16
601
9
112
0

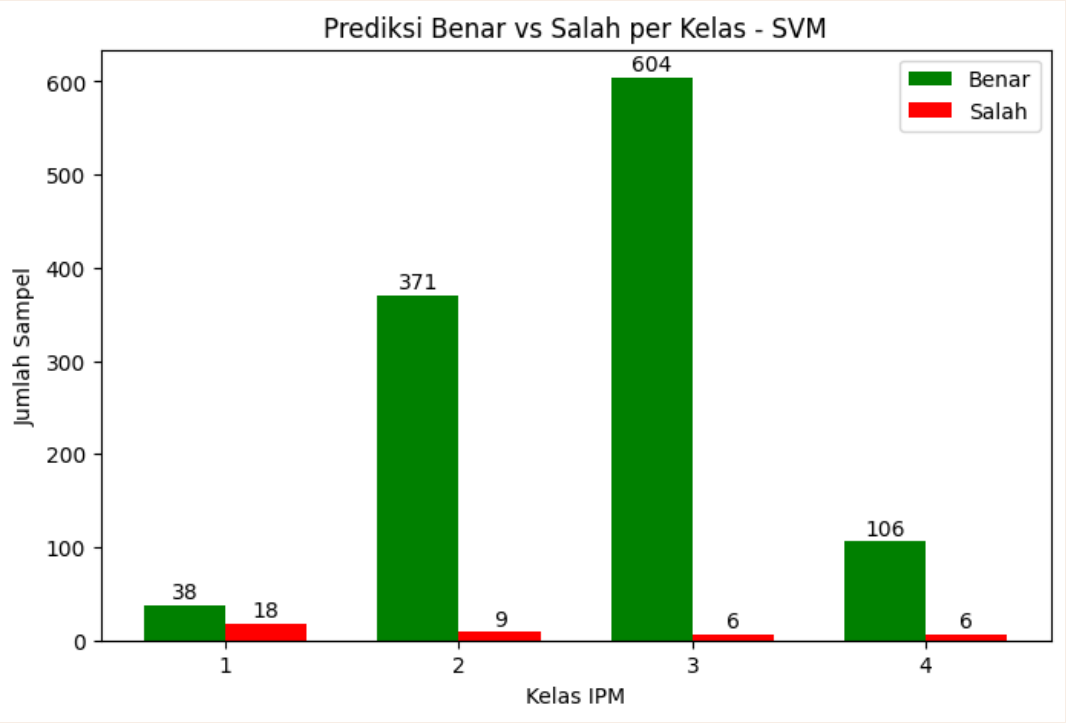

Persentase Benar vs Salah - SVM
(Benar=1116, Salah=42)
Salah
3.6%
96.4%
Benar

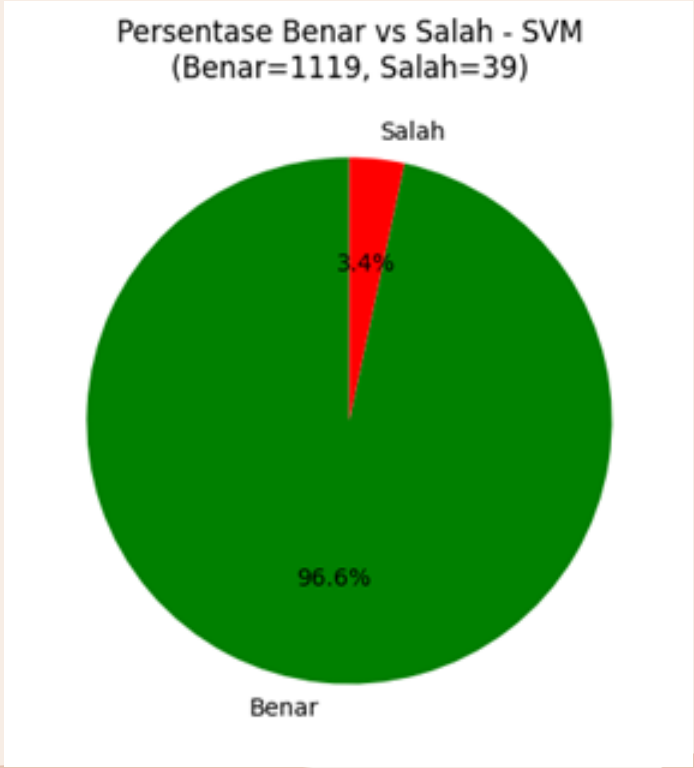
SVM: Non-SMOTE

Kelas	Non-SMOTE		
	Presisi	Recall	F1-Score
1	1.000	0.679	0.809
2	0.984	0.976	0.980
3	0.954	0.990	0.972
4	0.964	0.946	0.955



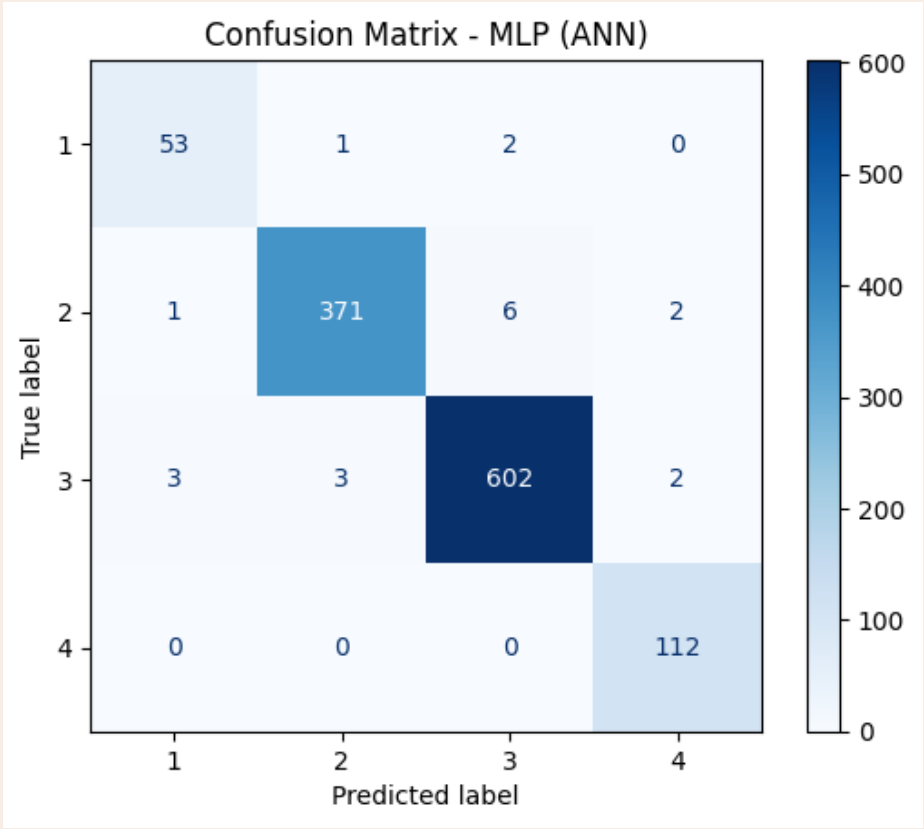
SVM: Non-SMOTE



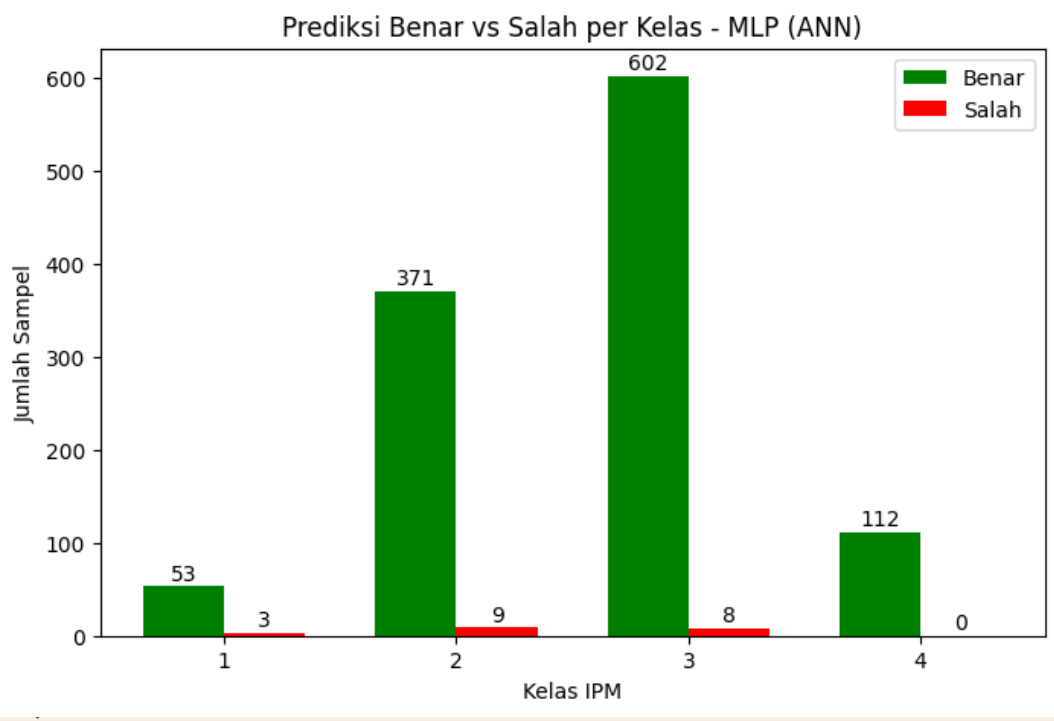


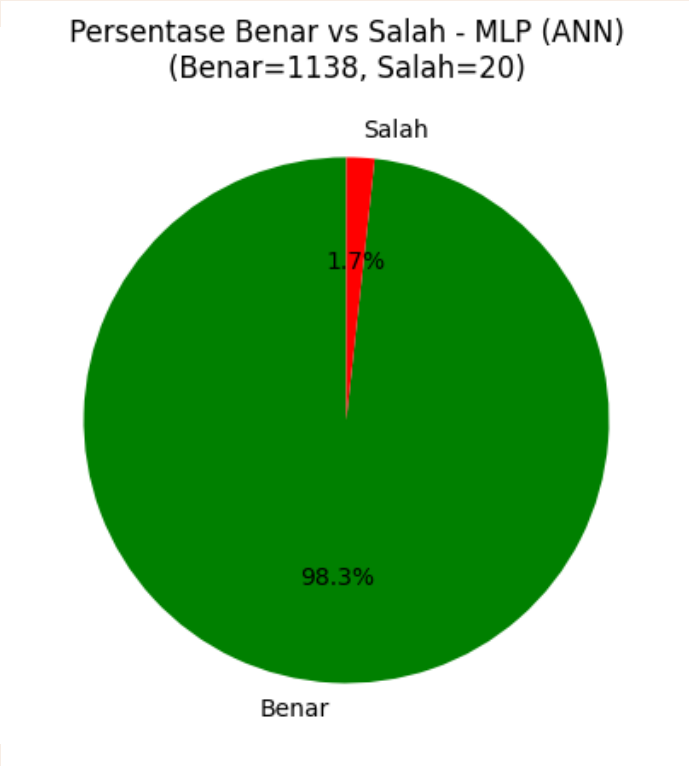
ANN: SMOTE

Kelas	SMOTE		
	Presisi	Recall	F1-Score
1	0.930	0.946	0.938
2	0.989	0.976	0.983
3	0.987	0.987	0.987
4	0.966	1.000	0.982



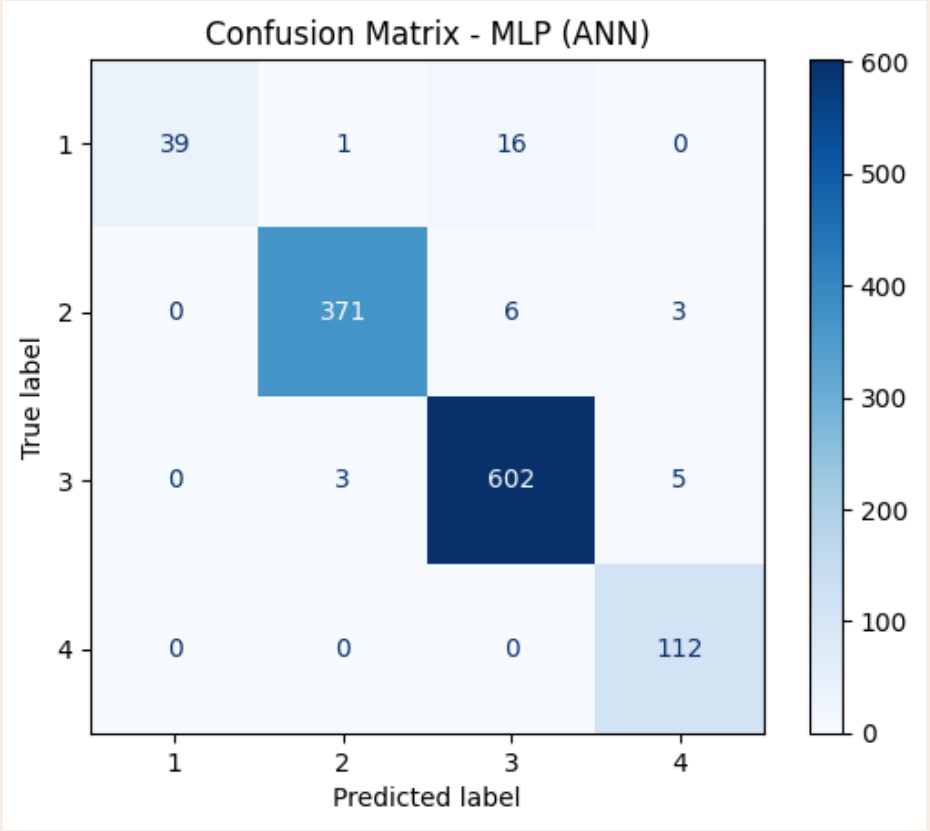
ANN: SMOTE


Prediksi Benar vs Salah per Kelas - MLP (ANN)
Benar
Salah
Jumlah Sampel
Kelas IPM
0
100
200
300
400
500
600
1
2
3
4
53
3
371
9
602
8
112
0

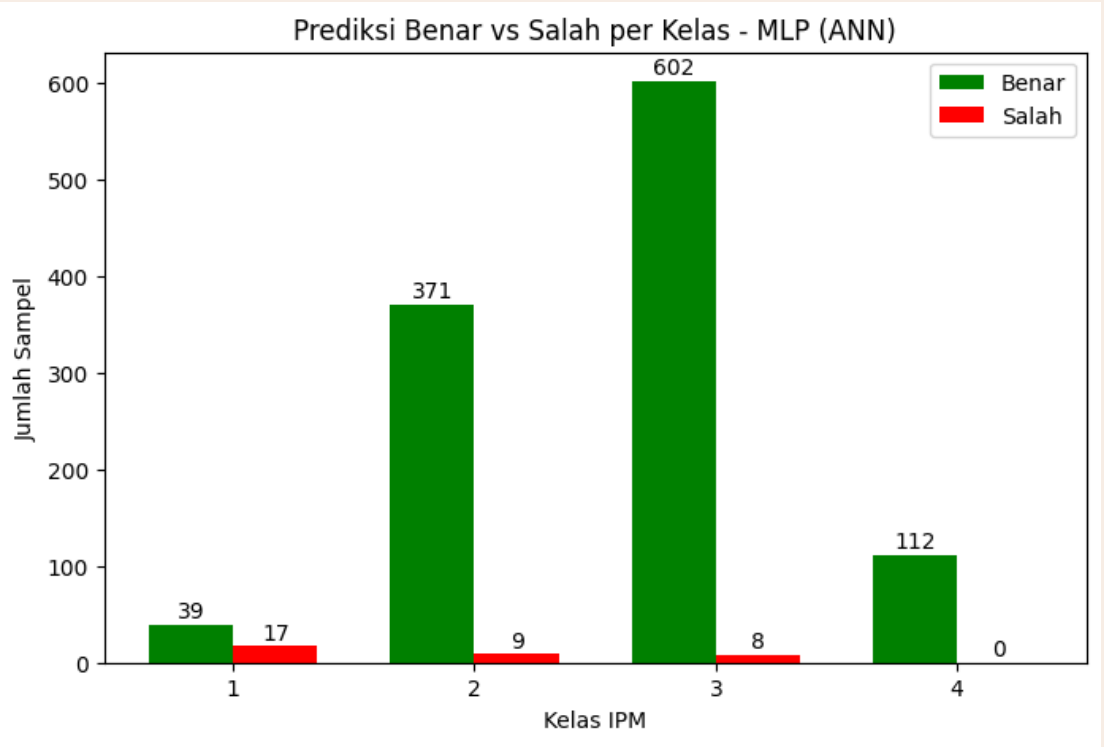

Persentase Benar vs Salah - MLP (ANN)
(Benar=1138, Salah=20)
Salah
1.7%
98.3%
Benar

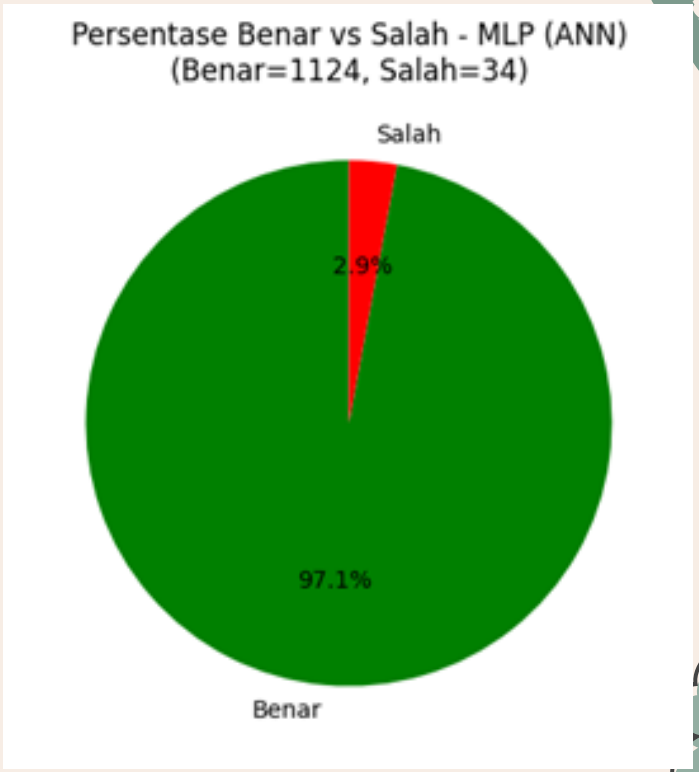
ANN: Non-SMOTE

Kelas	Non-SMOTE		
	Presisi	Recall	F1-Score
1	1.000	0.696	0.821
2	0.989	0.976	0.983
3	0.965	0.987	0.976
4	0.933	1.000	0.966



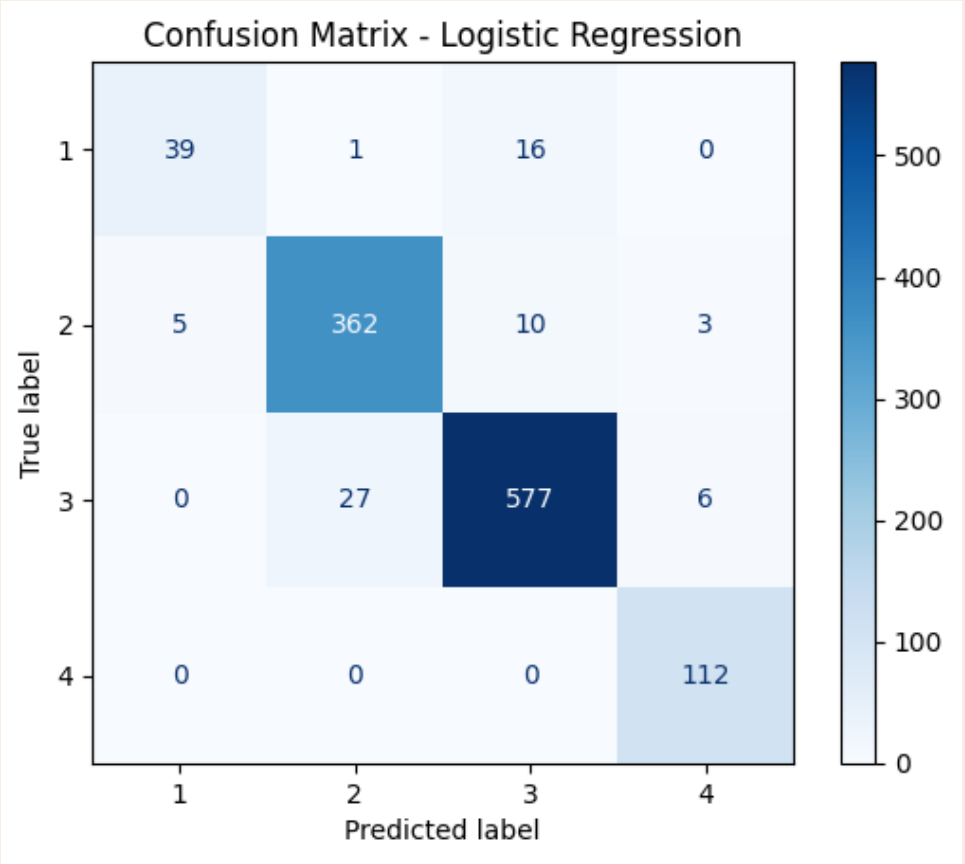
ANN: Non-SMOTE



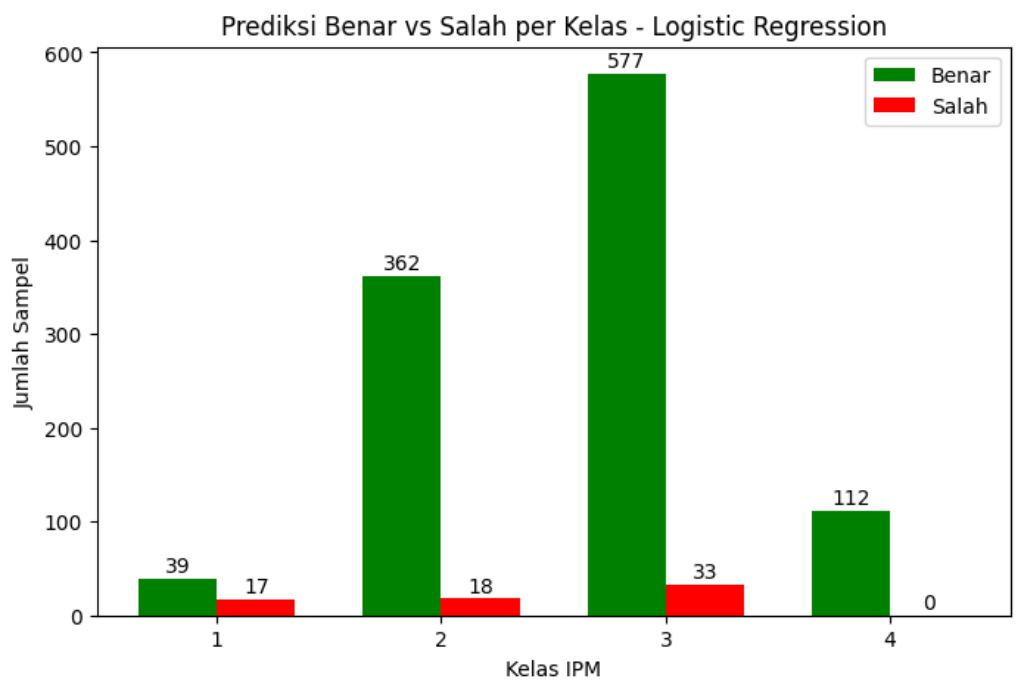


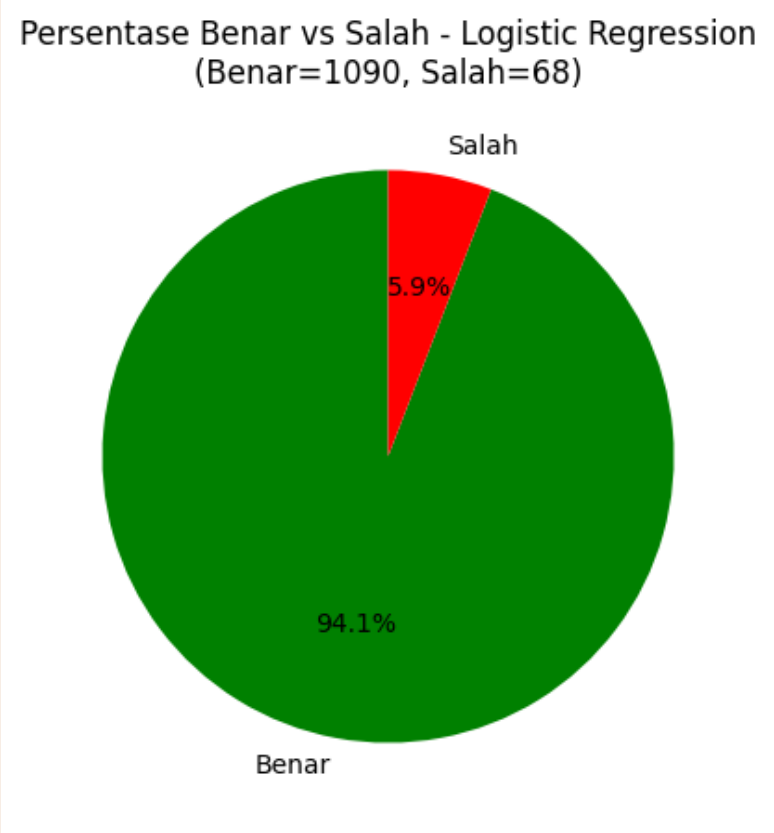
Regresi Logistik: SMOTE

Kelas	SMOTE		
	Presisi	Recall	F1-Score
1	0.886	0.696	0.780
2	0.928	0.953	0.940
3	0.957	0.946	0.951
4	0.926	1.000	0.961



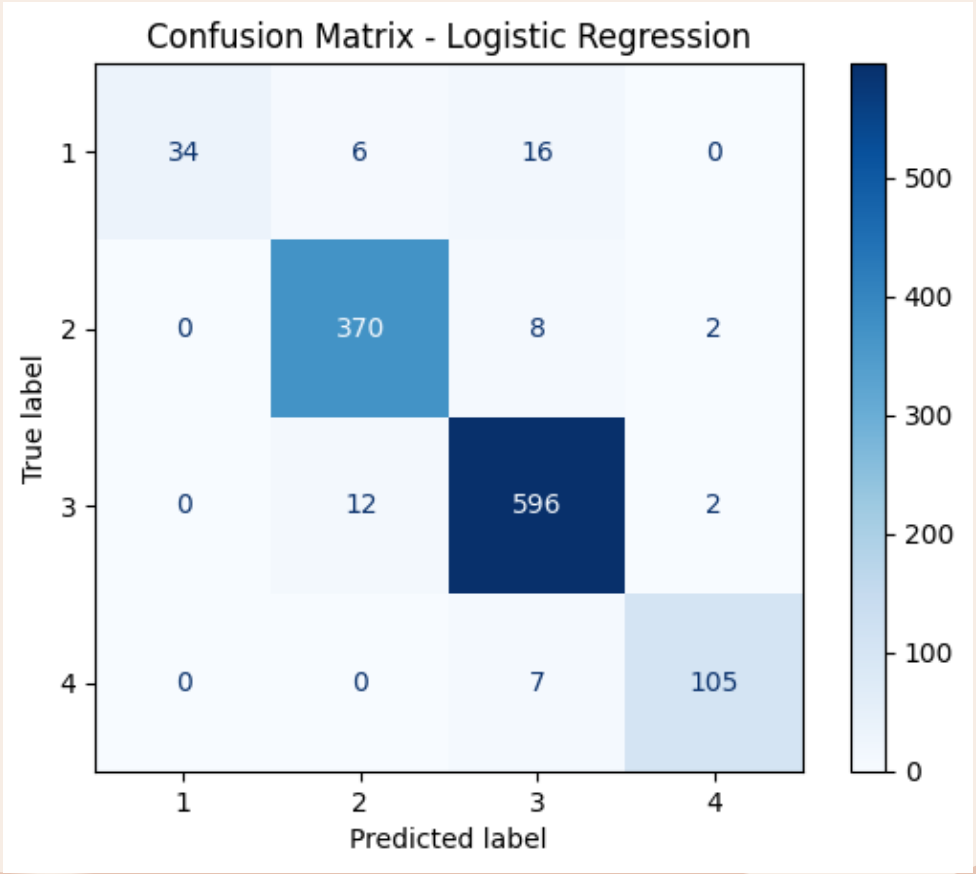
Regresi Logistik: SMOTE



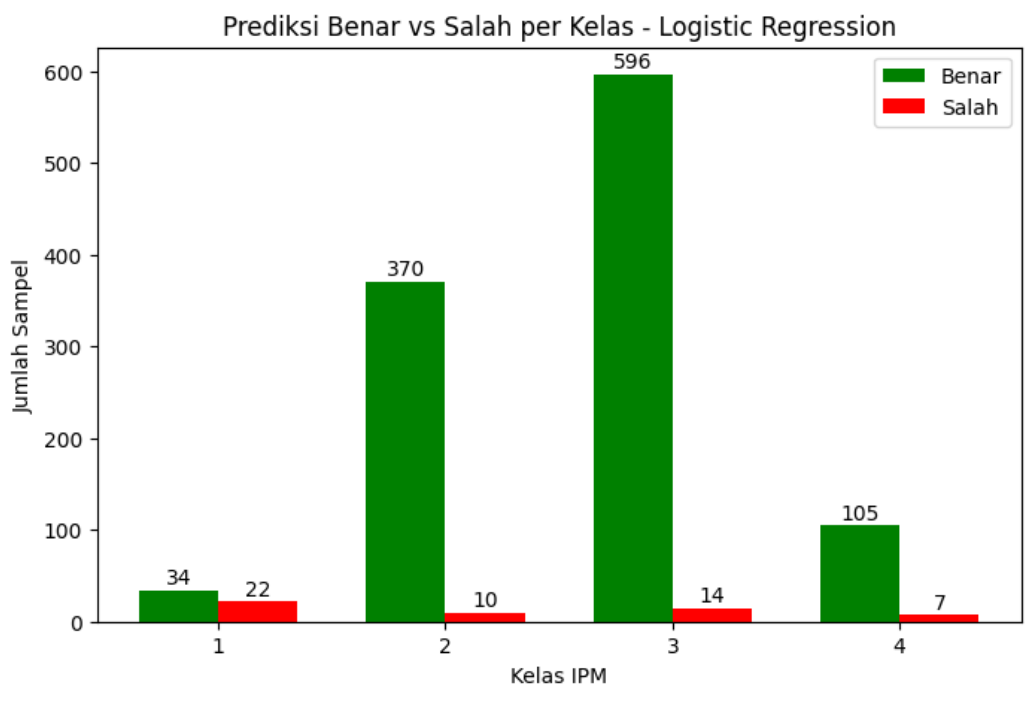


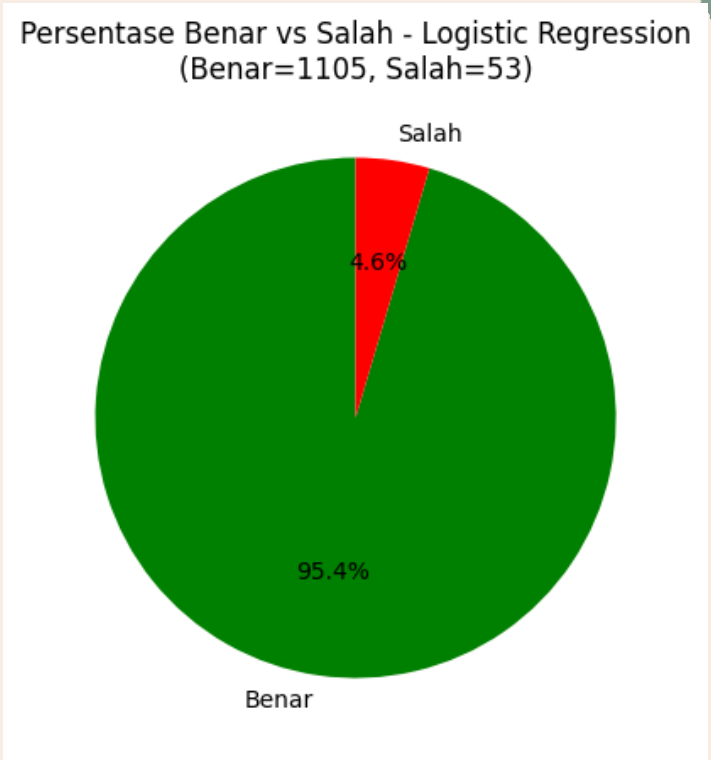
Regresi Logistik: Non-SMOTE

Kelas	Non-SMOTE		
	Presisi	Recall	F1-Score
1	1.000	0.607	0.756
2	0.954	0.974	0.964
3	0.951	0.977	0.964
4	0.963	0.938	0.950



Regresi Logistik: Non-SMOTE





Rangkuman Semua Model

Nama Model	SMOTE			Non-SMOTE		
	Accuracy	Weighted F1	Log-loss	Accuracy	Weighted F1	Log-loss
Decision Tree	0.911054	0.910501	3.205955	0.894646	0.894350	3.797345
Random Forest	0.933506	0.932519	0.259283	0.928325	0.927374	0.244688
Gradient Boosting	0.943869	0.942804	0.208718	0.938687	0.937415	0.226510
AdaBoost	0.815199	0.794440	1.340290	0.823834	0.805929	1.338794
SVM	0.963731	0.962741	0.201838	0.966321	0.965048	0.198663
ANN	0.982729	0.982749	0.081766	0.970639	0.969554	0.106151
Logistic Regression	0.941278	0.940399	0.443743	0.954231	0.952238	0.530161

Peringkat Semua Model

Rank	Model
1	ANN - SMOTE
2	ANN - Non-SMOTE
3	SVM - Non-SMOTE
4	SVM - SMOTE
5	Gradient Boosting - SMOTE
6	Gradient Boosting - Non-SMOTE
7	Logistic Regression - Non-SMOTE
8	Logistic Regression - SMOTE
9	Random Forest - SMOTE
10	Random Forest - Non-SMOTE
11	Decision Tree - SMOTE
12	Decision Tree - Non-SMOTE
13	AdaBoost - Non-SMOTE
14	AdaBoost - SMOTE

04 Kesimpulan

Kesimpulan

- Variabel kesehatan, pendidikan, dan kesejahteraan terbukti memiliki korelasi yang kuat dengan IPM.
- Permasalahan utama dataset adalah imbalance kelas IPM, dan penggunaan SMOTE terbukti meningkatkan performa sebagian besar model untuk mempelajari kelas minoritas.
- Perbandingan tujuh model Machine Learning menunjukkan bahwa Artificial Neural Network (ANN) dengan SMOTE merupakan model yang paling optimal dan bagus.
- SMOTE terbukti efektif, tetapi tidak cocok untuk semua model.
- Hasil pemeringkatan akhir (1-14) konsisten dengan evaluasi numerik dan analisis visual (confusion matrix dan distribusi salah–benar), sehingga model terbaik sudah ditentukan secara komprehensif.
- Model paling buruk adalah AdaBoost, dimana menunjukkan underfit.

Saran

- Menerapkan Teknik Penanganan Imbalance Lain Selain SMOTE
- Menguji dan Mengubah Arsitektur dan Model yang Lebih Kompleks
- Melakukan Hyperparameter Tuning Secara Menyeluruh
- Mengintegrasikan Model ke Dalam Sistem Prediksi Berbasis Web atau Dashboard
- Mempertimbangkan Penggunaan Ensemble Berbasis Voting atau Stacking


Thank
you