

Optimizing Credit Scoring Performance Using Ensemble Feature Selection with Random Forest

Mengoptimalkan Performa Credit Scoring dengan Seleksi Fitur Ensembel Hybrid menggunakan Random Forest

Ana Fauziah

*Prodi Pendidikan Matematika, Fakultas Keguruan dan Ilmu Pendidikan,
Universitas Bakti Indonesia*

Email: ana@ubibanyuwangi.ac.id

Received: 11 November 2024, revised: 17 December 2024, accepted: 20 December 2024

Abstract

Credit scoring has a very important role in the financial industry to assess the eligibility of loan applicants and mitigate credit risk. However, the main challenge in credit scoring modeling is the large number of features that need to be considered. Feature selection becomes an inevitable step to improve model performance. This research proposes the use of hybrid ensemble boosting techniques through XGBoost, LightGBM, and CatBoost methods, as well as aggregation techniques for feature selection, the results of which are then used to build predictive models using Random Forest. Experimental results show that the aggregation technique using feature slices selected by the three methods provides the best model with the least number of features, which is only about 11% of the total features. The use of fewer features not only increases the computational speed and efficiency of the model but also improves the generalization ability, which allows the model to perform better on new data. In addition, this model shows the smallest difference between train accuracy and mean cross-validation score, indicating high model stability and reliability.

Keywords: classification, credit scoring, feature selection, ensemble method, random forest

Abstrak

Credit scoring memiliki peran yang sangat penting dalam industri keuangan untuk menilai kelayakan pemohon pinjaman dan memitigasi risiko kredit. Namun, tantangan utama dalam pemodelan credit scoring adalah banyaknya fitur yang harus dipertimbangkan. Seleksi fitur menjadi langkah yang tak terhindarkan untuk meningkatkan performa model. Penelitian ini mengusulkan penggunaan teknik hybrid ensemble boosting melalui metode XGBoost, LightGBM, dan CatBoost, serta teknik agregasi untuk seleksi fitur, yang hasilnya kemudian digunakan untuk membangun model prediktif menggunakan Random Forest. Hasil eksperimen menunjukkan bahwa



teknik agregasi menggunakan irisan fitur yang dipilih oleh ketiga metode memberikan model terbaik dengan jumlah fitur yang paling sedikit, yakni hanya sekitar 11% dari total fitur yang ada. Penggunaan fitur yang lebih sedikit tidak hanya meningkatkan kecepatan komputasi dan efisiensi model, tetapi juga memperbaiki kemampuan generalisasi, yang memungkinkan model untuk bekerja lebih baik pada data baru. Selain itu, model ini menunjukkan selisih terkecil antara *train accuracy* dan *mean cross-validation score*, yang mengindikasikan kestabilan dan keandalan model yang tinggi.

Kata kunci: klasifikasi, skor kredit, seleksi fitur, metode ensemble, random forest

1. PENDAHULUAN

Krisis finansial Asia 1997-1998 memiliki dampak yang signifikan dan luas terhadap ekonomi global terutama pada negara-negara di Asia yang sebelumnya mengalami lonjakan pertumbuhan ekonomi pesat. Beberapa negara seperti Indonesia, Thailand, Korea Selatan, Malaysia dan Filipina mengalami kontraksi ekonomi besar, inflasi tinggi, dan beban utang yang menghancurkan sektor bisnis dan perbankan. Krisis ini sebagian besar dipicu oleh praktik pinjaman yang longgar seperti pemberian pinjaman dalam jumlah besar tanpa penilaian risiko yang memadai. Kasus finansial besar tersebut menunjukkan bahwa mengabaikan penilaian risiko, dapat menyebabkan konsekuensi serius bagi bank dan ekonomi global.

Pada tahun 2004 Komite Basel, untuk Pengawasan Bank (BCBS) menyusun perjanjian yang dikenal dengan perjanjian Basel II untuk menjaga stabilitas sistem keuangan global dan meminimalkan risiko krisis keuangan. Perjanjian ini mengharuskan lembaga keuangan yang diawasi wajib menggunakan peringkat internal untuk mengukur risiko kredit. Kebutuhan untuk mengendalikan risiko kredit telah mendorong bank dan lembaga keuangan meningkatkan metode prediksi risiko kredit melalui prosedur *credit scoring* [1].

Model *credit scoring*, sebagai salah satu bentuk manajemen risiko kredit, bertujuan mengevaluasi kemungkinan terjadinya gagal bayar saat pinjaman diberikan. Model ini bekerja dengan mengklasifikasikan nasabah ke dalam dua kelompok, yakni nasabah yang layak dan tidak layak menerima pinjaman. Untuk mendukung proses tersebut, serangkaian fitur diperlukan guna mengenali karakteristik dan perilaku nasabah dengan lebih baik. Data keuangan yang digunakan biasanya berasal dari berbagai sumber yang mencakup aspek keuangan dan perilaku pembayaran. Namun, tantangan sering kali muncul karena data tersebut mengandung fitur yang tidak relevan atau fitur yang memiliki korelasi tinggi satu sama lain, yang dapat mengurangi efisiensi analisis.

Keberadaan fitur yang tidak signifikan terhadap risiko kredit atau fitur yang berkorelasi tinggi dapat membawa dampak negatif terhadap model *credit scoring*. Masalah ini dapat menyebabkan interpretasi model yang bias, di mana hubungan antar-fitur yang tidak relevan dapat disalahartikan sebagai faktor penting. Akibatnya, kinerja model menjadi kurang optimal, baik dalam hal akurasi klasifikasi maupun kemampuannya dalam memberikan prediksi yang andal [13]. Oleh karena itu, pengelolaan data dan seleksi fitur menjadi langkah esensial dalam memastikan model bekerja dengan maksimal.

Strategi pemilihan fitur sangat penting untuk menyaring fitur yang tidak relevan. Proses ini tidak hanya membantu mengurangi kompleksitas, tetapi juga meningkatkan interpretabilitas dan mengurangi risiko *overfitting* dalam pemodelan *credit scoring* menggunakan Machine Learning. Secara umum, teknik seleksi fitur dapat dikategorikan ke dalam tiga kelompok utama: *Filter Methods*, *Wrapper Methods*, dan *Embedded Methods* [7]. Metode *Filter* menghitung relevansi fitur secara independen dari algoritma prediksi, sering kali dengan cara menghitung korelasi antara fitur dan label kelas. Meskipun metode ini cepat, mereka mungkin kurang efektif dalam mempertimbangkan hubungan antar fitur. Sebaliknya, *Wrapper Methods* menggunakan model prediksi untuk menilai kombinasi fitur dan memilih subset yang optimal. Pendekatan ini menguji

beberapa subset fitur menggunakan model pembelajaran untuk mengidentifikasi set fitur terbaik. Meskipun efektif, *Wrapper Methods* umumnya memerlukan waktu komputasi yang lebih besar [8]. Sementara itu, *Embedded Methods* mengintegrasikan pemilihan fitur ke dalam proses pelatihan model, memungkinkan pemilihan fitur dilakukan secara otomatis selama pelatihan [3].

Berbagai penelitian mengenai penerapan seleksi fitur dalam pemodelan *credit scoring* telah dilakukan, di antaranya adalah studi oleh [14] dan [8] yang mengimplementasikan *Filter* dan *Wrapper Methods*. Di sisi lain, [2] dan [17] menerapkan *Embedded Methods* dengan teknik regularisasi dalam penelitian mereka. Berdasarkan temuan dari beberapa studi tersebut, tulisan ini akan mengkaji lebih dalam mengenai pemilihan fitur menggunakan *Embedded Methods* berbasis pohon.

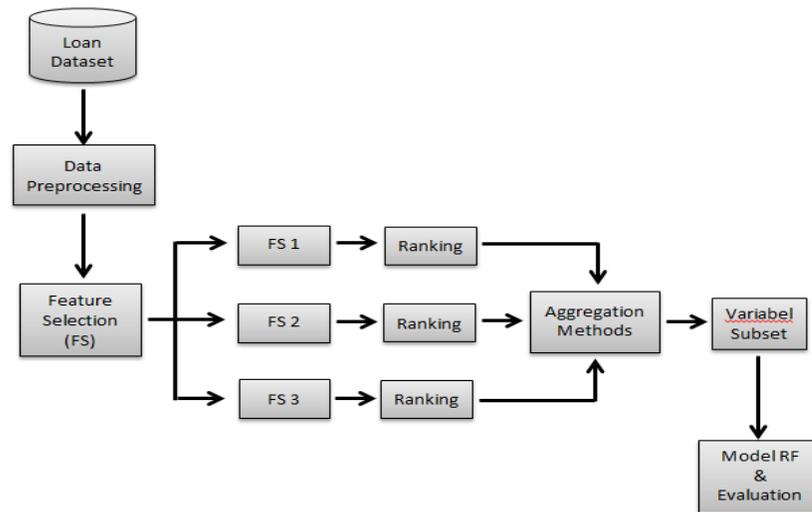
Seiring dengan perkembangan teknik seleksi fitur, *Embedded Methods* berbasis pohon telah menjadi salah satu pendekatan yang banyak digunakan dalam seleksi fitur karena kemampuannya untuk secara bersamaan mengidentifikasi fitur yang relevan selama proses pelatihan model. Algoritma seperti XGBoost, LightGBM, dan CatBoost memanfaatkan struktur pohon untuk membangun model prediktif yang efektif, sekaligus memberikan wawasan penting mengenai relevansi fitur dalam pemodelan [6][12]. Salah satu keunggulan metode ini adalah kemampuannya dalam menangani interaksi antar fitur dan mengurangi risiko *overfitting*, sehingga menghasilkan model yang lebih tangguh.

Melalui penerapan strategi *hybrid*, hasil seleksi fitur dari ketiga algoritma *boosting* dapat digabungkan untuk menciptakan pendekatan yang lebih menyeluruh. Teknik ini berfungsi untuk meningkatkan keandalan fitur yang dipilih, karena fitur-fitur tersebut telah diuji melalui beberapa pendekatan *ensemble* [16]. Dengan demikian, penggunaan gabungan atau irisan fitur dari ketiga metode ini memungkinkan untuk mengidentifikasi fitur-fitur yang secara konsisten penting di berbagai model (dalam irisan) serta fitur dengan kontribusi potensial dari masing-masing algoritma (dalam gabungan).

Fitur-fitur yang telah terbukti signifikan dari berbagai metode *boosting* akan digunakan untuk membangun model *credit scoring* menggunakan Random Forest. Pendekatan *hybrid* ini, diharapkan dapat mengoptimalkan hasil prediksi, meningkatkan akurasi model, serta meningkatkan efisiensi dalam proses seleksi fitur. Kontribusi utama penelitian ini adalah mengembangkan teknik seleksi fitur yang efektif pada *credit scoring* untuk meningkatkan akurasi model prediktif, serta menyediakan wawasan tentang fitur-fitur yang paling relevan dalam pengambilan keputusan kredit.

2. METODE PENELITIAN

Tahapan yang digunakan dalam pengembangan model *credit scoring* pada penelitian ini disajikan dalam Gambar 2.1. Proses penelitian meliputi tahap persiapan data, proses seleksi fitur, pengembangan model prediksi, serta evaluasi kinerja model.



Gambar 2.1. Skema Penelitian

2.1. Data observasi

Dataset yang digunakan dalam penelitian ini berasal dari Lending Club, sebuah platform pinjaman *peer-to-peer* yang dapat diakses melalui Kaggle. Data ini mencakup informasi mengenai pinjaman yang diberikan melalui platform tersebut selama periode 2007 hingga 2014. Terdapat total 227.214 entri pinjaman dan 74 variabel yang mencakup berbagai aspek, termasuk informasi tentang pinjaman, profil peminjam, dan informasi tambahan lainnya. Dataset ini dapat diakses melalui <https://www.kaggle.com/wordsforthewise/lending-club>. Dalam penelitian ini, *loan status* akan digunakan sebagai variabel respon, sementara 73 variabel lainnya akan berfungsi sebagai variabel prediktor.

2.2. Data Preprocessing

Data mentah yang diperoleh tidak dapat langsung digunakan untuk pemodelan. Untuk itu, tahap pra-pemrosesan data, yang sering disebut sebagai data *preprocessing*, sangat penting dilakukan. Proses ini bertujuan untuk mengubah data menjadi format yang lebih mudah dipahami dan sesuai dengan kebutuhan analisis yang diinginkan [15]. Beberapa langkah *preprocessing* yang dilakukan antara lain:

1. Pembersihan data
 - a. Menghapus beberapa informasi yang tidak diperlukan atau menghilangkan kolom yang mengandung lebih dari 45% *missing value*.
 - b. Mengidentifikasi dan mengatasi nilai yang hilang, baik dengan menghapus entri tersebut atau mengisi nilai yang hilang dengan metode tertentu (misalnya, rata-rata, median, atau modus).
2. Transformasi data
 - a. Normalisasi dan Standarisasi: Mengubah skala data agar berada dalam rentang tertentu, yang membantu dalam algoritma yang sensitif terhadap skala.
 - b. Encoding Kategori: Mengubah variabel kategori menjadi format numerik (misalnya, menggunakan *one-hot encoding* atau *label encoding*).
3. Penanganan data tidak seimbang
umumnya menggunakan teknik *oversampling* seperti SMOTE
4. Pengurangan dimensi
Menghilangkan fitur yang tidak relevan atau berlebihan melalui teknik seleksi fitur.

2.3. Seleksi fitur

Penggunaan variabel *dummy* melalui *one-hot encoding* akan meningkatkan jumlah kolom fitur. Penambahan kolom ini dapat menyebabkan masalah dimensionalitas tinggi, terutama jika variabel kategori memiliki banyak kategori, yang membuat model menjadi lebih kompleks, sulit untuk dilatih, serta meningkatkan risiko *overfitting*. Untuk mengatasi masalah ini, salah satu solusinya adalah melakukan prosedur seleksi fitur agar hanya fitur-fitur yang relevan yang digunakan dalam pemodelan.

2.3.1 Boosting Feature Selection Methods

Metode boosting, seperti XGBoost, LightGBM dan CatBoots, termasuk dalam kategori *embedded methods*. Ini karena algoritma *boosting* secara otomatis menghitung pentingnya fitur selama proses pelatihan. Dalam *boosting*, setiap model baru yang ditambahkan berusaha untuk memperbaiki kesalahan dari model sebelumnya, dan selama proses ini, fitur yang lebih relevan untuk memprediksi target akan mendapatkan bobot yang lebih tinggi. Oleh karena itu, *boosting* tidak hanya melakukan prediksi tetapi juga melakukan pemilihan fitur secara implisit.

Ketiga metode *boosting* utama, XGBoost, LightGBM, dan CatBoots, bekerja dengan dasar *split gain* untuk memilih fitur terbaik dalam melakukan *split*. *Split gain* adalah ukuran yang digunakan dalam algoritma pembelajaran mesin, khususnya dalam pohon keputusan untuk menilai seberapa efektif suatu fitur dalam membantu model membuat prediksi yang lebih baik. Cara kerja *split gain* adalah dengan membagi data menjadi subset yang lebih informatif berdasarkan target yang ingin diprediksi dengan mengacu pada peningkatan akurasi atau penurunan error dari setiap *split* pada pohon.

Meski demikian ketiganya memilih dan memperlakukan *split* secara berbeda. XGBoost menghitung efektivitas atau pentingnya fitur berdasarkan seberapa sering fitur digunakan dan memberikan kontribusi besar pada *split gain*. Fitur dengan nilai *gain* yang tinggi dianggap memberikan kontribusi besar pada *split gain* atau memiliki *importance* yang lebih tinggi. Sebaliknya Fitur dengan *gain* terendah dapat diabaikan dalam model, karena kontribusinya terhadap peningkatan akurasi sangat kecil yang menunjukkan bahwa fitur-fitur tersebut kurang relevan untuk model.

$$Gain(f) = \frac{1}{2} \left(\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right) - \gamma \quad (2.1)$$

Persamaan 2.1 merupakan rumus *gain* dalam XGBoost untuk sebuah fitur f . Dimana, G_L dan G_R adalah jumlah gradien di cabang kiri dan kanan hasil *split*, H_L dan H_R adalah jumlah hessian (nilai kuadrat gradien) di cabang kiri dan kanan hasil *split*, dan λ parameter regularisasi untuk mencegah *overfitting*. Sementara γ nilai penalti yang diberikan untuk setiap *split*, membantu menghindari *split* yang tidak terlalu signifikan [6].

LightGBM mirip dengan XGBoost dalam menentukan frekuensi penggunaan fitur untuk pemisahan dengan menghitung *split*. Namun, metode ini juga memiliki pendekatan unik dengan menggunakan teknik seperti Gradient-Based One-Side Sampling (GOSS) untuk meningkatkan efisiensi dalam pemilihan fitur dan pemisahan. GOSS memilih sampel dengan gradien terbesar (yaitu sampel dengan error terbesar) untuk selalu diproses dalam setiap iterasi.

$$g_i = \frac{\partial L(y_i, \hat{y}_i)}{\partial \hat{y}_i} \quad (2.2)$$

g_i adalah gradien dari *loss* untuk sampel ke i , $\partial L(y_i, \hat{y}_i)$ adalah fungsi *loss*, y_i adalah nilai target aktual dan \hat{y}_i adalah prediksi model untuk sampel ke i . Pada kasus klasifikasi biner dengan *Binary Cross-Entropy Loss*, fungsi *loss* dinyatakan seperti pada persamaan 2.3

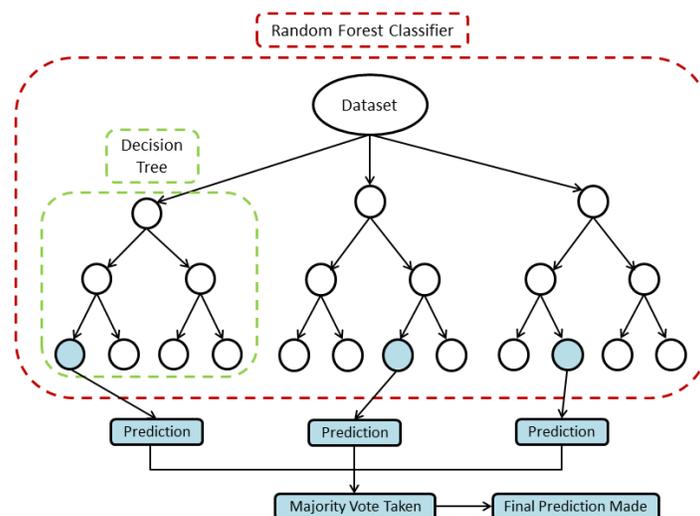
$$L(y, \hat{y}) = -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \quad (2.3)$$

Pada pemilihan fitur, CatBoost menggunakan *Loss Function Change* untuk mengukur seberapa signifikan kontribusi fitur dalam meningkatkan akurasi atau performa model. *Loss Function Change* menghitung akumulasi dari semua pengurangan *loss* dari fitur tersebut selama proses pelatihan yang ideal. CatBoost juga mengimplementasikan teknik yang disebut Minimal Variance Sampling (MVS) untuk meningkatkan akurasi pemisahan dengan cara yang lebih efisien [11]. Teknik ini bekerja dengan memilih sampel yang memiliki varian error rendah, sehingga model bisa fokus pada informasi yang lebih konsisten. Dengan demikian, MVS membantu model *boosting* untuk mencapai performa optimal dengan lebih efisien tanpa perlu memproses seluruh data pada setiap iterasi.

2.4 Random forest

Random Forest adalah salah satu algoritma *ensemble learning* yang populer dalam Machine Learning, terutama untuk tugas klasifikasi dan regresi. Algoritma ini bekerja dengan menggabungkan prediksi dari banyak *decision tree* yang dibangun secara acak untuk meningkatkan akurasi dan ketahanan model terhadap *overfitting*. Konsep utama dalam Random Forest adalah *bagging* (*Bootstrap Aggregating*), di mana sejumlah subset acak dari data pelatihan diambil dengan pengembalian [10]. Ini berarti beberapa data mungkin muncul lebih dari sekali dalam subset, sementara yang lain mungkin tidak muncul sama sekali.

Algoritma Random Forest membuat beberapa *decision tree* independen dengan data latih yang berbeda-beda. Setiap *node* dalam pohon tidak menggunakan semua fitur yang ada, melainkan hanya subset fitur yang dipilih secara acak. Pemilihan acak ini menambah variasi antar pohon dan mencegah pohon tertentu mendominasi keseluruhan model, yang pada akhirnya membantu mengurangi *overfitting*. Setelah semua pohon dibangun, Random Forest menghasilkan prediksi dengan cara mengambil rata-rata atau pemungutan suara terbanyak dari prediksi yang diberikan oleh setiap pohon. Kombinasi prediksi dari banyak pohon ini membuat hasil akhir lebih stabil dan akurat. Secara keseluruhan, cara kerja algoritma Random Forest dapat dilihat pada ilustrasi yang disajikan di Gambar 2.2.



Gambar 2.2. Proses Random Forest

2.5 Model Evaluasi

Evaluasi model merupakan langkah penting dalam analisis data yang membantu kita mengetahui seberapa baik model dalam memberikan hasil yang tepat. Evaluasi model klasifikasi sering dibagi ke dalam tiga kategori metrik utama, yang membantu memberikan perspektif menyeluruh terhadap performa model yakni *threshold*, *probability* dan *ranking metrics* [4]. Dalam praktinya *threshold* dan *ranking* adalah metrik yang paling umum digunakan oleh para peneliti untuk mengukur kinerja klasifikasi.

2.5.1 Threshold Metrics

Threshold Metrics menggunakan ambang batas (*threshold*) untuk menentukan apakah prediksi model termasuk dalam kelas tertentu atau tidak. Accuracy, Precision, Recall, dan F1-Score adalah contoh dari *threshold metrics* yang nilainya bergantung pada *confusion matrix*. Dalam *confusion matrix*, ada 4 bagian yang penting untuk kita ketahui. Seluruh bagian tersebut digabung menjadi tabel sederhana seperti pada Table 2.1.

Table 2.1. *Confusion Matrix* untuk *Binary Classification*

Prediksi	Aktual	
	Kelas Positif	Kelas Negatif
Kelas Positif	<i>True Positives</i> (TP)	<i>False Positives</i> (FP)
Kelas Negatif	<i>False Negatives</i> (FN)	<i>True Negatives</i> (TN)

Berdasarkan *confusion matrix*, selanjutnya kita dapat menentukan nilai dari *threshold metrics* seperti yang tersaji pada persamaan (2.4) - (2.7).

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.4)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2.5)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2.6)$$

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (2.7)$$

2.5.2 Ranking Metrics

Metode ini berfokus pada seberapa baik model dapat memberi peringkat atau urutan pada prediksi untuk kelas positif dibandingkan dengan kelas negatif. Contoh metrik ini adalah Receiver Operating Characteristic-Area Under the Curve (ROC-AUC). Metrik ini mengukur kemampuan model untuk membedakan antara kelas. ROC *curve* memplot True Positive Rate (Recall) versus False Positive Rate, dan AUC adalah luas area di bawah kurva tersebut. Nilai AUC berkisar dari 0.5 (acak) hingga 1 (sempurna), dan semakin tinggi AUC, semakin baik performa model. Untuk klasifikasi dua kelas, nilai AUC dihitung menggunakan persamaan (2.8) [9]

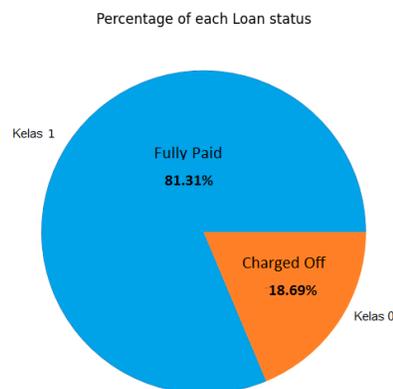
$$AUC = \frac{S_p - n_p(n_n + 1)/2}{n_p n_n} \quad (2.8)$$

S_p adalah jumlah seluruh contoh kelas positif yang diperingkat, n_p adalah jumlah kelas positif dan n_n adalah jumlah kelas negatif.

3. HASIL DAN PEMBAHASAN

3.1 Exploratory Data

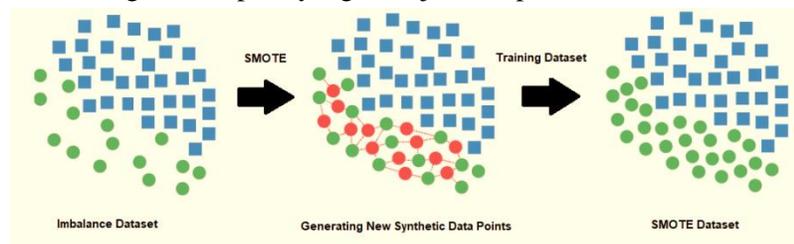
Data yang tersedia berisi informasi tentang pemohon pinjaman di masa lalu, serta status pembayaran pinjaman mereka, apakah gagal bayar (*charged off*) atau lunas (*fully paid*). Setelah data dibersihkan pada langkah *preprocessing*, jumlah data yang tersisa menjadi sebanyak 227.214. dari jumlah tersebut, sebanyak 184.739 pinjaman (sekitar 81,31% telah dilunasi dengan baik, sementara sisanya, yaitu 42.475 pinjaman (sekitar 18,69%), mengalami gagal bayar. Pinjaman yang dilunasi dikategorikan ke dalam label kelas 1, sementara pinjaman yang gagal bayar dimasukkan ke dalam label kelas 0.



Gambar 3.1. Perbandingan kelas *loan_status*

Gambar 3.1 menunjukkan adanya ketidakseimbangan distribusi kelas yang signifikan. Kondisi ini dapat mempengaruhi kinerja model secara substansial. Model cenderung lebih akurat dalam memprediksi kelas mayoritas yang lebih sering muncul selama pelatihan. Sebaliknya, model akan kesulitan untuk mengenali pola dari kelas minoritas yang frekuensinya rendah. Hal ini tentu saja dapat mengurangi efektivitas model dalam memprediksi kejadian-kejadian langka yang sebenarnya sangat penting.

Untuk mengatasi masalah ini, teknik resampling Synthetic Minority Over-sampling Technique (SMOTE) digunakan pada tahap *preprocessing* untuk menghasilkan kelas yang seimbang. Teknik SMOTE bekerja dengan cara menambah jumlah sampel dari kelas minoritas secara sintesis, bukan hanya sekadar menduplikasi sampel yang sudah ada [5]. Untuk setiap sampel dalam kelas minoritas, SMOTE memilih sejumlah tetangga terdekat (umumnya dengan menggunakan metrik jarak Euclidean) dalam ruang fitur, seperti yang ditunjukkan pada Gambar 3.2.



Gambar 3.2. Teknik resampling Synthetic Minority Over-sampling Technique (SMOTE)

Selanjutnya SMOTE akan membuat sampel baru yang dihasilkan dengan interpolasi linear antara sampel asli dan tetangganya dengan rumus berikut:

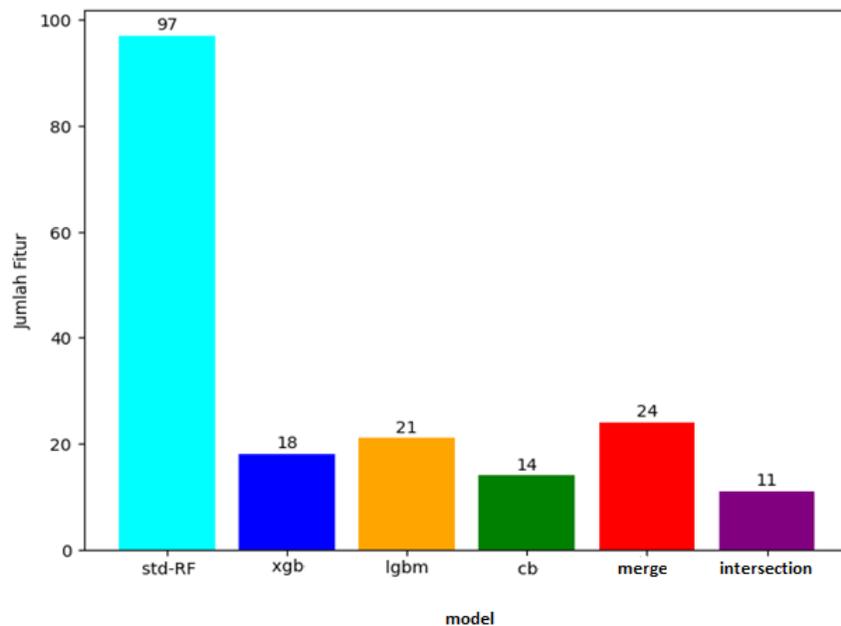
$$y' = y^i + (y^j - y^i)\gamma \quad (3.1)$$

y' merupakan sampel sintetis yang dibuat. y^i merepresentasikan kelas minoritas pada sampel asli, y^j merupakan sampel yang merupakan tetangga terdekat dari y^i dan γ adalah angka acak antara 0 dan 1 [18].

3.2 Data Experimental

Data yang telah melalui *preprocessing* mengalami kenaikan jumlah fitur akibat proses *encoding*. Sebanyak 24 fitur *dummy* ditambahkan pada fitur awal sehingga menghasilkan total 97 fitur. Langkah selanjutnya akan dilakukan seleksi fitur menggunakan beberapa metode berbasis *boosting*, yaitu XGBoost, LightGBM, dan CatBoost untuk memilih fitur-fitur yang paling relevan. Setiap metode menghasilkan rangkaian fitur yang diprioritaskan berdasarkan tingkat kepentingannya terhadap prediksi status pinjaman.

Pada Gambar 3.2. terlihat bahwa pada penelitian ini secara umum metode seleksi *boosting* mampu mereduksi sekitar 80% dari jumlah fitur yang ada. Catboost menghasilkan jumlah fitur yang paling sedikit jika dibandingkan dua metode *boosting* lainnya. Gabungan hasil seleksi dari ketiga metode *boosting* menghasilkan 24 fitur sedangkan irisan dari hasil seleksinya menghasilkan jumlah 11 fitur saja.



Gambar 3.2. Perbandingan jumlah fitur awal dan hasil seleksi

Fitur hasil seleksi selanjutnya digunakan dalam pemodelan *credit scoring* menggunakan Random Forest. *Hyperparameter* yang digunakan tersaji pada Table 3.1.

Tabel 3.1. Random Forest *Hyperparameter*

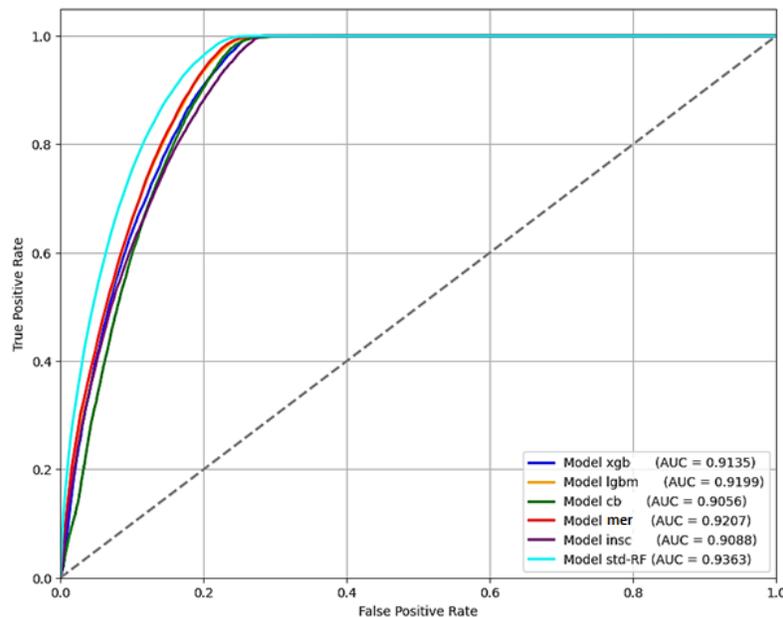
<i>Hyperparameter</i>	Nilai
n_estimators	100
max_features	Square root of the total features

min_samples_split	2
min_samples_leaf	1
bootstrap	True

Pada Tabel 3.2, *threshold metrics* seperti precision, recall, F1-score dan accuracy menunjukkan nilai yang relatif berdekatan pada tiap metode seleksi maupun hasil dari metode agregasi. Konsistensi nilai ini cenderung menyebabkan metrik tambahan, seperti ROC *curve* dan AUC, juga menunjukkan variasi yang minimal atau perbedaan yang tidak signifikan seperti ditampilkan pada Gambar 3.3.

Tabel 3.2. *Threshold Metrics* dari berbagai teknik seleksi fitur

Metode seleksi fitur	Evaluasi						Accuracy
	Precision		Recall		F1-Score		
	Kelas 0	Kelas 1	Kelas 0	Kelas 1	Kelas 0	Kelas 1	
XGBoost	0,97	0,80	0,76	0,98	0,85	0,88	0,87
LightGBM	0,94	0,82	0,79	0,95	0,86	0,88	0,87
CatBoost	0,94	0,81	0,78	0,95	0,85	0,87	0,86
Merged method	0,94	0,82	0,79	0,95	0,86	0,88	0,87
Intersection method	0,99	0,79	0,73	0,99	0,84	0,88	0,86

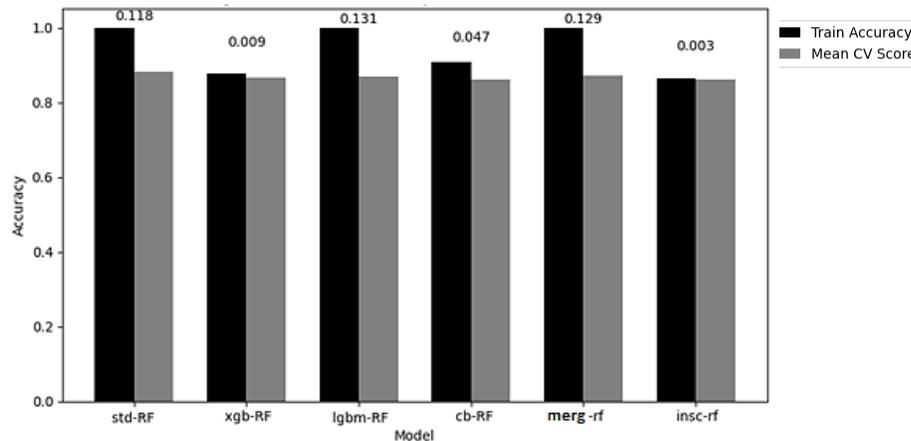


Gambar 3.3. Perbandingan ROC - AUC Model dari berbagai fitur seleksi

Secara keseluruhan, model yang dihasilkan menunjukkan kemampuan generalisasi yang baik. Hal ini terlihat dari selisih antara *train accuracy* dan *mean cross-validation score*, yang digunakan untuk mendeteksi potensi masalah *overfitting* atau *underfitting*, menunjukkan nilai yang relatif kecil. Semakin kecil selisih antara *train accuracy* dan *mean cross-validation score*, semakin ideal hasil pemodelan yang diperoleh.

JURNAL MATEMATIKA, STATISTIKA DAN KOMPUTASI

Ana Fauziah



Gambar 3.4. Perbandingan selisih *train accuracy* dan *mean cross-validation score*

Meskipun perbedaan nilai evaluasi yang dihasilkan oleh beberapa metode seleksi tidak terlalu signifikan, model dengan fitur dari hasil irisan metode *boosting* menunjukkan performa terbaik. Hal ini didasarkan pada jumlah fitur yang lebih sedikit namun konsisten penting di berbagai model, yang memungkinkan model mempertahankan performa setara dengan model yang menggunakan lebih banyak fitur. Selain itu, penggunaan fitur hasil irisan juga berpotensi meningkatkan kemampuan generalisasi model, karena hanya fitur yang relevansi dan kekonsistennya tinggi yang terpilih. Fitur-fitur terbaik dari hasil irisan seleksi metode *boosting* disajikan pada Tabel 3.3.

Tabel 3.3. Fitur hasil irisan metode *boosting*

Fitur	Keterangan
Grade	Peringkat risiko kredit
Home_ownership_MORTGAGE	Status kepemilikan rumah peminjam yang berupa hipotek
Home_ownership_RENT	Status kepemilikan rumah peminjam yang berupa sewa
Initial_list_status_f	Pinjaman berasal dari pendanaan berbagi
Initial_list_status_w	Pinjaman sepenuhnya didanai oleh satu pemberi pinjaman.
Inq_last_6mths	Informasi transaksi setoran 6 bulan terakhir
Term_36 months	Jangka waktu pinjaman selama 36 bulan (3 tahun)
Term_60 months	Jangka waktu pinjaman selama 60 bulan (5 tahun)
Verification_status_Not Verified	Informasi peminjam tidak diverifikasi
Verification_status_Source verified	Informasi peminjam diverifikasi langsung dari sumbernya
Verification_status_Verified	Informasi peminjam telah diverifikasi

4. KESIMPULAN

Metode *boosting* secara keseluruhan memiliki kemampuan yang sangat baik dalam melakukan seleksi fitur untuk membangun model *credit scoring*. Model yang dihasilkan menunjukkan performa yang baik dan efektif dalam memilih fitur-fitur penting yang mendukung stabilitas klasifikasi. Proses agregasi melalui penggabungan (*merged method*) dan irisan hasil seleksi fitur (*intersection method*) dari beberapa metode dapat memperoleh fitur yang konsisten penting di berbagai model (dalam irisan) maupun fitur dengan kontribusi signifikan dari setiap model (dalam gabungan). Pendekatan ini memperkuat keandalan fitur yang dipilih karena telah diuji melalui sejumlah teknik ensemble. Selain itu, pendekatan ini merupakan salah satu ciri utama *hybrid methods* yang fokus

pada fitur yang terbukti signifikan di berbagai metode boosting. Ke depan, pengembangan seleksi fitur *hybrid ensemble* diharapkan dapat lebih ditingkatkan dengan menggabungkan metode seleksi lain seperti metode *filter* dan *wrapper* untuk mencapai hasil yang lebih optimal.

DAFTAR PUSTAKA

- [1] Abellán, J., & Castellano, J. G., 2017. A comparative study on base classifiers in ensemble methods for credit scoring. *Expert Systems with Applications*, Vol. 73, 1–10.
- [2] Arora, N., & Kaur, P. D., 2020. A Bolasso based consistent feature selection enabled random forest classification algorithm: An application to credit risk assessment. *Applied Soft Computing Journal*, Vol. 86, No. 105936.
- [3] Bashir, S., Khattak, I. U., Khan, A., Khan, F. H., Gani, A., & Shiraz, M., 2022. A Novel Feature Selection Method for Classification of Medical Data Using Filters, Wrappers, and Embedded Approaches. *Complexity*, Vol. 2022. <https://doi.org/10.1155/2022/8190814>. [08 November 2024]
- [4] Caruana, R., & Niculescu-Mizil, A., 2004. Data mining in metric space: An empirical analysis of supervised learning performance criteria. *KDD-2004 - Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 69–78.
- [5] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P., 2002. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, Vol. 16, 321–357.
- [6] Chen, T., & Guestrin, C., 2016. XGBoost : A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 785–794. Association for Computing Machinery, New York.
- [7] Chuang, L. Y., Yang, C. H., Wu, K. C., & Yang, C. H., 2011. A hybrid feature selection method for DNA microarray data. *Computers in Biology and Medicine*, Vol. 41, No. 4, 228–237.
- [8] Ha, V., & Nguyen, H., 2016. Credit scoring with a feature selection approach based deep learning. *7th International Conference on Mechanical, Industrial, and Manufacturing Technologies*, Vol. 54 thn 2016, 1–5. EDP Sciences, Les Ulis.
- [9] Hand, D. J., & Till, R. J., 2001. A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems. *Machine Learning*, Vol. 45, No. 2, 171–186.
- [10] Hastie, T., Robert, T., & Jerome, F., 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Predictions*, Second Edition. Springer, California.
- [11] Ibragimov, B., & Gusev, G., 2019. Minimal variance sampling in stochastic gradient boosting., *Advances in Neural Information Processing Systems 32: Proceedings of Annual Conference on Neural Information Processing Systems 2019*. NeurIPS, Vancouver.
- [12] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T. Y., 2017. LightGBM: A highly efficient gradient decision tree. *Advances in Neural Information Processing Systems 30: Proceedings of Annual Conference on Neural Information Processing Systems 2017*, 3147–3155. NIPS, California.
- [13] Laborda, J., & Ryoo, S., 2021. Feature selection in a credit scoring model. *Mathematics*, Vol. 9, No. 7, 746-769.

- [14] Liang, D., Tsai, C. F., & Wu, H. T., 2015. The effect of feature selection on financial distress prediction. *Knowledge-Based Systems*, Vol. 73, No. 1, 289–297.
- [15] Rerung, R. R., 2018. Penerapan Data Mining dengan Memanfaatkan Metode Association Rule untuk Promosi Produk. *Jurnal Teknologi Rekayasa*, Vol. 3, No. 1, 89-98.
- [16] Zhou, Z.H., 2012. *Ensemble Methods Foundations and Algorithms*. CRC Press, Boca Raton.
- [17] Zhou, Y., Uddin, M. S., Habib, T., & Chi, G., 2021. Feature selection in credit risk modeling : an international evidence an international evidence. *Economic Research-Ekonomiska Istraživanja*, Vol. 34, No. 2, 1–31.
- [18] Zhu, T., Lin, Y., & Liu, Y., 2017. Synthetic minority oversampling technique for multiclass imbalance problems. *Pattern Recognition*, Vol. 72, 327-340.