# KLASIFIKASI KECURANGAN TRANSAKSI KARTU KREDIT MENGGUNAKAN METODE *SUPPORT VECTOR MACHINE* BERBASIS *SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE* DAN *PRINCIPAL COMPONENT ANALYSIS*

**SKRIPSI**

**Oleh:**
**SHOFEY M. YUSUEF ALI**
**200605110016**

**PROGRAM STUDI TEKNIK INFORMATIKA**
**FAKULTAS SAINS DAN TEKNOLOGI**
**UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM**
**MALANG**
**2026**

# KLASIFIKASI KECURANGAN TRANSAKSI KARTU KREDIT MENGGUNAKAN METODE *SUPPORT VECTOR MACHINE* BERBASIS *SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE* DAN *PRINCIPAL COMPONENT ANALYSIS*

**SKRIPSI**

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

**Oleh:**
**SHOFEY M. YUSUEF ALI**
**NIM. 200605110016**

**PROGRAM STUDI TEKNIK INFORMATIKA**
**FAKULTAS SAINS DAN TEKNOLOGI**
**UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG**
**2026**

# HALAMAN PERSETUJUAN

**KLASIFIKASI KECURANGAN TRANSAKSI KARTU KREDIT MENGGUNAKAN METODE *SUPPORT VECTOR MACHINE* BERBASIS *SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE* DAN *PRINCIPAL COMPONENT ANALYSIS***

**SKRIPSI**

**Oleh :**
**SHOFEY M. YUSUEF ALI**
**NIM. 200605110016**

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 10 Maret 2026

| Pembimbing I, | Pembimbing II, |
|---|---|
| Prof. Dr. Suhartono S.Si M.Kom | Ajib Hanani, M.T |
| NIP. 196805192003121001 | NIP. 198407312023211013 |

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Supriyono, M. Kom
NIP. 19841010 201903 1 012

# HALAMAN PENGESAHAN

**KLASIFIKASI KECURANGAN TRANSAKSI KARTU KREDIT MENGGUNAKAN METODE *SUPPORT VECTOR MACHINE* BERBASIS *SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE* DAN *PRINCIPAL COMPONENT ANALYSIS***

**SKRIPSI**

**Oleh:**
**SHOFEY M. YUSUEF ALI**
**NIM. 200605110016**

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 10 April 2026

**Susunan Dewan Penguji**

| | | |
|---|---|---|
| Ketua Penguji | : Dr. Irwan Budi Santoso, M.Kom<br>NIP. 19770103 201101 1 004 | ( ) |
| Anggota Penguji I | : Okta Qomaruddin Aziz, M.Kom<br>NIP. 19911019 201903 1 013 | ( ) |
| Anggota Penguji II | : Prof. Dr. Suhartono, S.Si., M.Kom<br>NIP. 19680519 200312 1 001 | ( ) |
| Anggota Penguji III | : Ajib Hanani, M.T<br>NIP. 19840731 202321 1 013 | ( ) |

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Supriyono, M. Kom
NIP. 19841010 201903 1 012

# PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

| | |
|---|---|
| Nama | : Shofey M. Yusuef Ali |
| NIM | : 200605110016 |
| Fakultas / Program Studi | : Sains dan Teknologi / Teknik Informatika |
| Judul Skripsi | : Klasifikasi Kecurangan Transaksi Kartu Kredit Menggunakan Metode *Support Vector Machine* Berbasis *Synthetic Minority Over-Sampling Technique* Dan *Principal Component Analysis* |

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 10 April 2026
Yang membuat pernyataan,


Shofey M. Yusuef Ali
NIM. 200605110016

**MOTTO**

*“Tiada kekayaan yang lebih utama dari pada akal,*
*tiada keadaan yang lebih menyedihkan dari pada kebodohan,*
*dan tiada warisan yang lebih baik dari pada pendidikan.”*

*-Ali Bin Abi Thalib*

## HALAMAN PERSEMBAHAN

Saya persembahkan karya ini kepada: Abi saya, Ali Turmudi, M.Ag

Beliau adalah sebaik baik bentuk yang mungkin bagi orang tua, ayah, guru dan pelindung bagi dirinya dan keluarganya. Terimakasih atas segala doa, dukungan dan semangat bagi saya sampai pada titik ini.

Ibu saya, Sukismiati, S.Pd Beliau adalah seseorang yang bilang dengan bangga bahwa saya anaknya, yang menyebut nama saya pada tiap doanya, beliau jauh lebih tinggi dari aneka macam surga. Terimakasih atas segala doa, dukungan dan semangat bagi saya sampai pada titik ini.

Adik tercinta saya, Firda Ziadatur Rizqiyah Mutiara Ali Dengan segala tingkah menjengkelkanmu, tumbuh lebih baik cari panggilamu, jadi lebih baik dibanding diriku.

Sahabat-sahabat saya, Muhammad Izzun Nur, Muhammad Farjar Arrozi, Muhammad Dzikrullah, Dani Salim, Munawarul Haqq, Abdul Ghofur, Zaky Fathurrahman, Khalid Fahruddin, Ahmad Azhar Darmawan, Dani Ahmad Ikhramudien, Muhammad Imam Ghozali, Rokhim Nur Rifai, Alm. Helmi Noor Hafidz, dan kawan-kawan semuanya yang belum bisa saya sebut.

Terima kasih telah berbagi suka, duka, cerita, cinta dan pengalaman selama masa perkuliahan. Semoga kita semua selalu dikelilingi kabar baik, pertolongan, dan kemudahan oleh Allah SWT.

# KATA PENGANTAR

*Assalamualaikum Warahmatullahi Wabarakatuh.*

Segala puji hanya milik Allah Subhanahu Wa Ta'ala atas segala nikmat dan kasih sayang-Nya yang telah memudahkan penulis untuk menyelesaikan skripsi yang berjudul "Klasifikasi Kecurangan Transaksi Kartu Kredit Menggunakan Metode *Support vector machine* Berbasis *Synthetic Minority Over-sampling Technique* dan *Principal Component Analysis*". Semoga shalawat dan salam senantiasa terlimpah kepada Nabi Muhammad *Sallalahu 'Alaihi wa Sallam*. Dan semoga kita semua mendapat syafaatnya di hari kiamat nanti, Aamiin.

Penulis mengucapkan rasa syukur dan terima kasih yang tak terhingga kepada semua pihak-pihak yang selalu memberikan bantuan dan motivasi kepada penulis untuk dapat menyelesaikan skripsi ini. Ucapan ini penulis sampaikan kepada:

1. Prof. Dr. Hj. Ilfi Nur Diana, M.Si., CAHRM., CRMP. selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Agus Mulyono, M.Kes., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Supriyono, M.Kom, selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Prof Suhartono, M.Kom dan Ajib Hanani, M.T selaku dosen pembimbing yang dengan penuh kesabaran dan kebijaksanaan telah membimbing saya

dalam menyelesaikan skripsi ini. Bimbingan, nasihat, dan pengarahan yang diberikan sangat berarti bagi kemajuan dan kelancaran penelitian ini.

5. Dr. Irwan Budi Santoso, M.Kom dan Okta Qomaruddin Aziz, M.Kom selaku dosen penguji yang dengan sabar telah memberi arahan dan saran dalam proses penelitian ini.
6. Segenap civitas akademika Program Studi Teknik Informatika, terutama seluruh dosen. Terimakasih atas ilmu dan bimbingan yang telah diberikan selama masa perkuliahan ini.
7. Semua pihak yang telah membantu penulis dalam proses perkuliahan hingga menyelesaikan skripsi ini, baik secara langsung maupun tidak langsung yang tidak dapat disebutkan satu per satu.

Akhir kata, penulis menyadari bahwa penulisan pada skripsi ini masih jauh dari kata sempurna. Saya berdoa semoga skripsi ini dapat diterima sebagai amal ibadah yang ikhlas dan bermanfaat disisi Allah SWT. Semoga karya ini menjadi bentuk kontribusi dalam rangka memperkokoh dan mengembangkan ilmu pengetahuan dan menjalankan tugas sebagai hamba Allah SWT yang bertanggung jawab.

Malang, 07 April 2026

Penulis

## DAFTAR ISI

# DAFTAR TABEL

## DAFTAR GAMBAR

# ABSTRAK

Ali, Shofey M. Yusuef. 2026. **Klasifikasi Kecurangan Transaksi Kartu Kredit Menggunakan Metode *Support vector machine* Berbasis *Synthetic Minority Over-sampling Technique* Dan *Principal Component Analysis*** Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. Suhartono, S.Si., M.Kom. (II) Ajib Hannani, M.T.

**Kata kunci:** *Kartu Kredit, Klasifikasi Kecurangan, Principal Component Analysis, SMOTE, Support vector machine.*

Kecurangan transaksi kartu kredit merupakan tindakan ilegal yang menimbulkan kerugian finansial dengan memanfaatkan kelemahan pada sistem pembayaran digital. Seiring meningkatnya penggunaan transaksi elektronik, jumlah kasus kecurangan juga terus bertambah, sementara sistem deteksi konvensional masih menghadapi keterbatasan dalam membedakan transaksi normal dan transaksi *fraud* secara efektif. Oleh karena itu, penelitian ini mengimplementasikan algoritma *Support vector machine* (SVM) yang dikombinasikan dengan *Synthetic Minority Over-sampling Technique* (SMOTE) dan *Principal Component Analysis* (PCA) untuk meningkatkan performa klasifikasi kecurangan transaksi kartu kredit. Dataset yang digunakan berasal dari Kaggle dan terdiri atas 5.000 data transaksi dengan proporsi kelas *fraud* sebesar 1,7%, sehingga memiliki karakteristik data yang tidak seimbang. Tahapan pra-pemrosesan meliputi seleksi fitur, label encoding, standard scaling, penyeimbangan data menggunakan SMOTE, serta reduksi dimensi menggunakan PCA. Evaluasi model dilakukan menggunakan *confusion matrix* dengan metrik *accuracy*, *precision*, *recall*, dan *F1-score* pada empat skenario pembagian data latih dan data uji, yaitu 60:40, 70:30, 80:20, dan 90:10. Hasil pengujian menunjukkan bahwa integrasi SMOTE dan PCA pada algoritma SVM mampu meningkatkan kemampuan deteksi transaksi *fraud* sekaligus mempertahankan efisiensi komputasi. Performa terbaik diperoleh pada rasio pembagian data 70:30 dengan nilai *recall* sebesar 52%, *F1-score* sebesar 41%, dan akurasi yang stabil pada 97%. Temuan ini menunjukkan bahwa metode yang diusulkan efektif dalam mengatasi permasalahan klasifikasi data tidak seimbang pada kasus deteksi kecurangan transaksi kartu kredit.

# ABSTRACT

Ali, Shofey M. Yusuef. 2026. **Classification of Credit Card Transaction *Fraud* Using *Support vector machine* Method Based on *Synthetic Minority Over-sampling Technique* and *Principal Component Analysis***. Undergraduate Thesis. Informatics Engineering Study Program, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University Malang. Advisors: (I) Prof. Dr. Suhartono, S.Si., M.Kom., (II) Ajib Hannani, M.T.

**Keywords:** *Fraud Classification, Credit Card, Support vector machine, SMOTE, Principal Component Analysis.*

Credit card *fraud* has become a major challenge in digital payment systems due to its potential to cause significant financial losses. The increasing volume of electronic transactions has been accompanied by a rise in *fraud*ulent activities, while conventional detection methods often struggle to accurately distinguish between legitimate and *fraud*ulent transactions, particularly in highly *imbalance*d datasets. To address this issue, this study implements a *Support vector machine* (SVM) classifier enhanced with *Synthetic Minority Over-sampling Technique* (SMOTE) and *Principal Component Analysis* (PCA) to improve *fraud* detection performance. The experiments were conducted using a Kaggle credit card transaction dataset consisting of 5,000 records, in which *fraud*ulent transactions accounted for only 1.7% of the data. The *preprocessing* stage included feature selection, label encoding, standard scaling, class balancing through SMOTE, and dimensionality reduction using PCA. Model performance was evaluated using a *confusion matrix* based on *accuracy*, *precision*, *recall*, and *F1-score* metrics under four train-test split ratios: 60:40, 70:30, 80:20, and 90:10. The experimental results demonstrate that the integration of SMOTE and PCA with SVM enhances the detection of minority-class transactions while maintaining computational efficiency. The best performance was achieved with a 70:30 train-test split, yielding a *recall* of 52%, an *F1-score* of 41%, and a stable *accuracy* of 97%. These findings indicate that the proposed approach is effective in addressing class *imbalance* issues in credit card *fraud* classification.

## الملخص


علي، شوقي م. يوسف. 2026. **تصنيف احتيال معاملات بطاقات الائتمان باستخدام طريقة *آلة الدعم المخصصة* بناء على *تقنية أخذ العينات الزائدة للأقلية الاصطناعية* وأطروحة *تحليل المكونات الرئيسية***. برنامج دراسة هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية، مالانغ. المشرف: (الأول) البروفيسور الدكتور سوهارتونو، S.Si، م. كوم. (الثاني) أجب حناني، م.ت.

**الكلمات المفتاحية:** *تصنيف الاحتيال، بطاقة الائتمان، آلة الدعم، SMOTE، تحليل المكونات الرئيسية.*

*الاحتيال في معاملات بطاقات الائتمان هو نشاط غير قانوني يضر ماليا من خلال استغلال الثغرات في نظام الدفع الرقمي. الزيادة السريعة في المعاملات الإلكترونية تتناسب طرديا مع الزيادة في حالات الاحتيال، التي غالبا ما تتفاقم بسبب محدودية الأنظمة التقليدية في التمييز بدقة بين المعاملات العادية والمريبة. لتجاوز هذه المشكلات، تقترح هذه الدراسة تطبيق طريقة آلة متجه الدعم (SVM) المحسنة باستخدام تقنية أخذ العينات الزائدة للأقليات الاصطناعية (SMOTE) وتحليل المكونات الرئيسية (PCA) لتحسين أداء تصنيف الاحتيال. تستخدم هذه الدراسة مجموعة بيانات معاملات بطاقات الائتمان من كاجل تضم 5,000 بيانات مع نسبة غير متوازنة جدا من فئات الاحتيال (عدم التوازن)، والتي تبلغ 1.7٪. تشمل مراحل معالجة البيانات المسبق اختيار الميزات، وترميز التسميات، وتوسيع المعيار، والتعامل مع اختلال توازن البيانات باستخدام SMOTE، وتقليل الأبعاد باستخدام PCA. تم تقييم اختبار النماذج من خلال مصفوفة الالتباس باستخدام مقاييس الدقة والدقة والاسترجاع ودرجة F1 بناء على أربعة سيناريوهات لمشاركة بيانات التدريب والاختبار، وهي 60:40، 70:30، 80:20، و90:10. تظهر النتائج أن الجمع بين SMOTE وPCA في خوارزمية SVM قادر على تحسين قدرة اكتشاف فئات الأقليات بشكل كبير (الاحتيال) مع الحفاظ على الكفاءة الحاسوبية. تم تحقيق أفضل أداء في سيناريو مشاركة البيانات بنسبة 70:30، حيث تمكن النموذج من إنتاج قيمة استرجاع بنسبة 52٪ ودرجة F1 تبلغ 41٪، مع معدل دقة ظل مستقرا عند 97٪. يثبت هذا أن النهج المقترح فعال في معالجة مشكلة تصنيف بيانات غير متوازنة في **حالات كشف الاحتيال في بطاقات الائتمان.***

# BAB I

# PENDAHULUAN

## 1.1 Latar Belakang

Perkembangan teknologi informasi dan komunikasi yang semakin pesat telah mendorong transformasi di berbagai sektor, termasuk sektor keuangan. Salah satu bentuk perkembangan tersebut adalah meningkatnya penggunaan kartu kredit sebagai sarana pembayaran yang memberikan kemudahan, kecepatan, dan fleksibilitas dalam bertransaksi. Kartu kredit memungkinkan pemegang kartu untuk melakukan pembayaran maupun penarikan tunai berdasarkan batas kredit yang telah ditetapkan oleh penerbit kartu (Arifudin et al., 2021). Meskipun memberikan berbagai manfaat, kemajuan teknologi juga diikuti oleh munculnya berbagai bentuk kejahatan digital, salah satunya adalah kecurangan transaksi kartu kredit yang terus mengalami peningkatan seiring dengan bertambahnya penggunaan sistem pembayaran elektronik.

إِنَّ ٱللَّهَ يَأۡمُرُ بِٱلۡعَدۡلِ وَٱلۡإِحۡسَٰنِ وَإِيتَآيِٕ ذِي ٱلۡقُرۡبَىٰ وَيَنۡهَىٰ عَنِ ٱلۡفَحۡشَآءِ وَٱلۡمُنكَرِ وَٱلۡبَغۡيِۚ يَعِظُكُمۡ لَعَلَّكُمۡ تَذَكَّرُونَ

*"Sesungguhnya Allah menyuruh berlaku adil, berbuat kebajikan, dan memberikan bantuan kepada kerabat. Dia (juga) melarang perbuatan keji, kemungkaran, dan permusuhan. Dia memberi pelajaran kepadamu agar kamu selalu ingat."*(Q.S *An-Nahl* : 90).

Al-Ashfahani menjelaskan bahwa kata *'adl* memiliki makna kesetaraan dalam pembagian atau pemberian kepada setiap pihak secara proporsional. Di sisi lain, beberapa pakar mendefinisikan *'adl* sebagai tindakan menempatkan sesuatu pada posisi yang sesuai dengan hakikat dan fungsinya. Selain itu, terdapat

pandangan yang menyatakan bahwa *'adl* merupakan pemberian hak kepada pemiliknya secara tepat melalui jalan yang paling sesuai dengan ketentuan yang berlaku. (Ubaidillah, 2018). Ayat diatas memberikan pemahaman tentang pentingnya berlaku adil khususnya terhadap orang terdekat dan umumnya kepada seluruh manusia.

Dalam penelitan ini, ayat diatas dihubungkan dengan akibat yang cukup serius dari perbuatan curang yang dilakukan manusia khususnya dalam konteks *muamalah* dalam hal ini mengacu pada kecurangan transaksi menggunakan kartu kredit.

Dewasa ini praktik kecurangan kartu kredit sangat sering dijumpai. Laporan dari *European Payments Council* menunjukkan bahwa kecurangan kartu kredit secara global telah menyebabkan kerugian lebih dari 32 miliar USD pada tahun 2021, dan angka ini diperkirakan akan meningkat hingga 50 miliar USD pada tahun 2025 jika tidak ada sistem yang lebih canggih (Chunchu, 2024).

Perkembangan tingkat penggunaan kartu kredit di Indonesia mengalami kenaikan yang cukup besar dalam beberapa tahun terakhir, hal tersebut selaras dengan tingkat kejahatan yang terjadi pada perbankan khususnya pada penipuan transaksi. Dari negara-negara yang memiliki resiko kecurangan kartu kredit, Indonesia menjadi negara dengan nomor urut kedua yang memiliki tingkat kecurangan kartu kredit tertinggi sebesar 18,3% (Kumar Manjhvar & Goyal, 2020). Kecurangan kartu kredit merupakan tindakan ilegal yang dilakukan oleh pihak yang tidak berwenang untuk mendapatkan keuntungan finansial dengan cara yang tidak

sah. Kecurangan ini dapat terjadi dalam berbagai bentuk, seperti pencurian identitas, penggunaan kartu kredit palsu, dan transaksi tidak sah.

Peraturan Otoritas Jasa Keuangan Nomor 39/POJK.03/2019 mengenai Penerapan Strategi Anti *Fraud* bagi Bank Umum menjadi landasan dalam pelaksanaan pengendalian *fraud* pada sektor perbankan. Dalam peraturan tersebut dijelaskan bahwa penerapan strategi anti *fraud* sekurang-kurangnya mencakup empat pilar utama, yaitu pencegahan, deteksi, investigasi yang meliputi proses pelaporan dan penjatuhan sanksi, serta pemantauan dan evaluasi yang mencakup kegiatan tindak lanjut. Implementasi keempat pilar tersebut bertujuan untuk meningkatkan efektivitas pengawasan sekaligus meminimalkan potensi terjadinya *fraud* dalam operasional perbankan. Hasil penelitian menunjukkan bahwa *fraud* transaksi hampir terjadi di perbankan di Indonesia, tetapi beberapa masih belum menggunakan strategi anti *fraud* berbasis *machine learning* (Rivki et al., 2021).

Permasalahan utama dalam deteksi kecurangan transaksi kartu kredit terletak pada karakteristik data transaksi yang kompleks. Data transaksi kartu kredit umumnya termasuk ke dalam kategori data berdimensi tinggi dengan jumlah sampel yang besar. Selain itu, distribusi kelas pada data tersebut cenderung tidak seimbang karena transaksi *fraud* hanya mencakup sebagian kecil dari keseluruhan transaksi yang terjadi. Kondisi ini menyebabkan proses identifikasi kecurangan menjadi sulit, karena model klasifikasi cenderung bias terhadap kelas mayoritas dan berpotensi menghasilkan tingkat kesalahan deteksi yang tinggi, khususnya kesalahan dalam mengidentifikasi transaksi sah sebagai transaksi curang *(False Positive).*

Selain permasalahan ketidakseimbangan data, tingginya jumlah fitur transaksi dan adanya korelasi antar fitur juga meningkatkan kompleksitas komputasi dalam proses klasifikasi. Pendekatan konvensional berbasis aturan *(rule-based system)* dinilai kurang efektif karena bersifat statis, sulit beradaptasi terhadap pola kecurangan yang terus berkembang, serta membutuhkan pembaruan aturan secara manual. Oleh karena itu, dibutuhkan pendekatan yang lebih adaptif dan mampu mempelajari pola transaksi secara otomatis dari data historis.

*Machine learning* telah banyak diterapkan sebagai solusi dalam mendeteksi kecurangan transaksi kartu kredit karena kemampuannya dalam mengolah data berskala besar dan kompleks. Dalam bidang deteksi kecurangan, *Support vector machine* (SVM) menjadi salah satu algoritma klasifikasi yang banyak dimanfaatkan karena kemampuannya dalam membedakan data antar kelas secara efektif. SVM bekerja dengan menentukan *hyperplane* terbaik yang memaksimalkan jarak pemisah *margin* antara kelas yang berbeda, sehingga proses klasifikasi dapat dilakukan dengan lebih optimal. Namun demikian, performa SVM dapat menurun ketika diterapkan pada dataset berdimensi tinggi karena meningkatnya beban komputasi dan risiko *overfitting*.

Untuk mengurangi risiko *overfitting* akibat tingginya jumlah fitur, penelitian ini menerapkan *Principal Component Analysis* (PCA) sebagai teknik reduksi dimensi yang mentransformasikan fitur asli ke dalam sejumlah komponen utama yang mampu merepresentasikan sebagian besar informasi data. Selain itu, untuk mengatasi ketidakseimbangan kelas pada dataset, digunakan *Synthetic Minority Over-sampling Technique* (SMOTE) pada data latih yang telah

distandardisasi. SMOTE bekerja dengan menghasilkan sampel sintetis pada kelas minoritas berdasarkan kedekatan antar data menggunakan pendekatan *k-nearest neighbors* (KNN). Penerapan PCA dan SMOTE diharapkan dapat menyederhanakan kompleksitas data, memperbaiki distribusi kelas, meningkatkan efisiensi komputasi, serta mengoptimalkan performa klasifikasi SVM.

Efektivitas SVM dalam klasifikasi kecurangan transaksi kartu kredit telah dibuktikan oleh berbagai penelitian sebelumnya. Yazid dan Fiananta (2017), melalui pengujian menggunakan *German Credit Dataset* yang diperoleh dari UCI *Machine learning Repository*, melaporkan bahwa metode SVM mampu mencapai performa klasifikasi yang baik dengan tingkat akurasi yang tinggi. Temuan tersebut menunjukkan bahwa SVM memiliki kemampuan yang cukup andal dalam membedakan transaksi yang berpotensi *fraud* dan transaksi normal. (Yazid & Fiananta, 2017). Selain itu, penelitian oleh (Alshawi, 2024) juga menunjukkan metode SVM mampu menghasilkan performa klasifikasi yang memuaskan dengan nilai akurasi mencapai 95%, presisi 99%, dan *recall* 82%. Nilai evaluasi tersebut menunjukkan bahwa SVM memiliki kemampuan yang baik dalam mengidentifikasi dan membedakan karakteristik masing-masing kelas data. Oleh karena itu, SVM dapat dipertimbangkan sebagai salah satu metode yang efektif untuk diterapkan pada permasalahan klasifikasi, khususnya dalam deteksi kecurangan transaksi.

Performa yang diperoleh dari hasil pengujian menunjukkan bahwa metode SVM memiliki kemampuan klasifikasi yang sangat baik, ditunjukkan oleh nilai akurasi sebesar 95%, presisi sebesar 99%, dan *recall* sebesar 82%. Hasil tersebut mengindikasikan bahwa SVM efektif dalam mengidentifikasi karakteristik masing-

masing kelas serta meminimalkan kesalahan klasifikasi. Oleh sebab itu, metode ini dinilai memiliki potensi yang kuat untuk diterapkan pada permasalahan deteksi kecurangan transaksi maupun kasus klasifikasi lainnya.

Selain itu, pemanfaatan teknik reduksi dimensi dan penanganan sering kali hanya digunakan sebagai pelengkap *preprocessing*, tanpa dianalisis secara khusus pengaruhnya terhadap performa dan stabilitas model SVM pada dataset transaksi berskala besar dan tidak seimbang. Oleh karena itu, masih terdapat celah penelitian dalam mengkaji secara sistematis peran PCA dan SMOTE sebagai bagian integral dari proses klasifikasi *fraud* berbasis SVM, khususnya dalam meningkatkan efisiensi komputasi dan kemampuan generalisasi model.

## 1.2 Rumusan Masalah

Merujuk pada permasalahan yang telah diuraikan pada bagian latar belakang, maka rumusan masalah dalam penelitian ini dapat dinyatakan sebagai berikut:

1. bagaimana performa metode *Support vector machine* dalam klasifikasi kecurangan transaksi kartu kredit?
2. Bagaimana pengaruh penerapan *Principal Component Analysis* dan *Synthetic Minority Over-sampling Technique* dalam mengukur performa model klasifikasi metode *Support vector machine*?

## 1.3 Batasan Masalah

Batasan – Batasan pada penilitan ini adalah sebagai berikut:

1. Penelitian ini menggunakan dataset transaksi kartu kredit yang tersedia secara publik di *kaggle* dengan *link* sebagai berikut *Fraud* Detection

Dataset. Dataset ini mungkin tidak mencakup semua variabel yang relevan atau tidak sepenuhnya representatif terhadap pola penipuan terbaru.

2. Teknik *preprocessing* yang diterapkan meliputi seleksi variabel, *label encoding, standard scaling, Principal Component Analysis* (PCA) sebagai metode reduksi dimensi, dan *Synthetic Minority Over-sampling Technique* (SMOTE) untuk menangani ketidakseimbangan data.
3. Evaluasi performa model dilakukan menggunakan *confusion matrix*dengan metrik akurasi, *Precision, recall,* dan *F1-score*.
4. Model penelitian ini akan diuji dalam metode simulasi dan tidak diterapkan langsung dalam sistem perbankan atau *fintech* nyata.

## 1.4 Tujuan Penelitian

Merujuk pada pernyataan masalah yang telah di jelaskan, maka tujuan lain dari penelitian ini adalah:

1. Mengetahui performa metode *Support vector machine* dalam klasifikasi kecurangan transaksi kartu kredit?
2. Mengetahui pengaruh penerapan *Principal Component Analysis* dan *Synthetic Minority Over-sampling Technique* dalam mengukur performa model klasifikasi metode *Support vector machine*.

## 1.5 Manfaat Penelitian

1. Manfaat Teoritis

   Penelitian ini diharapkan dapat memberikan kontribusi ilmiah dalam pengembangan metode klasifikasi kecurangan transaksi kartu kredit, khususnya terkait pemanfaatan PCA sebagai teknik reduksi dimensi dan

SMOTE sebagai teknik penanganan ketidakseimbangan data untuk meningkatkan performa dan efisiensi algoritma SVM.

2. Manfaat Praktis
    a. Memberikan solusi teknis bagi institusi perbankan dalam merancang sistem klasifikasi kecurangan transaksi kartu kredit yang lebih efisien dan akurat.
    b. Meningkatkan tingkat akurasi sistem deteksi kecurangan dalam meminimalkan kesalahan klasifikasi transaksi sah.

# BAB II

# STUDI PUSTAKA

## 2.1 Penelitian terdahulu

(Gedela & Karthikeyan, 2023), Melakukan penelitian untuk membandingkan performa beberapa algoritma *machine learning* dalam klasifikasi kecurangan transaksi. Algoritma *Support vector machine* (SVM) digunakan sebagai salah satu metode yang diuji dan dibandingkan *dengan Artificial Neural Network* (ANN), *Logistic Regression, Naïve Bayes,* serta *Decision Tree* guna menentukan algoritma dengan tingkat akurasi terbaik dalam mendeteksi *fraud.* Dataset yang digunakan dalam penelitian ini berasal dari basis data *Kaggle* yang terdiri dari 284.807 sampel transaksi kartu kredit; sampel ini dibagi menjadi dua bagian: 227.845 sampel, atau 80% dari sampel, digunakan untuk pelatihan model, dan 56.962 sampel, atau 20% dari sampel, digunakan untuk pengujian model. Hasil dari penelitian tersebut menunjukkan bahwa algoritma SVM memiliki akurasi 99,32%, sensitivitas 99,67%, spesifisitas 98,97%, presisi 98,98%, dan *f-score* 99,33%. Ini menunjukkan bahwa algoritma SVM jauh lebih baik daripada algoritma lainnya dalam menklasifikasi kecurangan transaksi kartu kredit.

(Sadineni, 2020) Melakukan penelitian yang berfokus pada klasifikasi transaksi curang pada kartu kredit menggunakan berbagai teknik *machine learning*, termasuk *Artificial Neural Network, Decision Trees, Support vector machine, Logistic Regression,* dan *Random Forest*. Tujuan dari penelitian ini adalah untuk membandingkan efektivitas masing-masing teknik tersebut dalam mengidentifikasi transaksi curang. Dataset yang digunakan dalam penelitian ini diperoleh dari

repositori data *Kaggle*. Dataset ini terdiri dari 150.000 transaksi kartu kredit, yang mencakup berbagai atribut seperti waktu transaksi, jumlah, dan klasifikasi transaksi. Kinerja setiap teknik diukur menggunakan metrik akurasi, presisi, dan *false alarm rate*. Algoritma SVM menunjukkan hasil yang cukup baik dalam penelitian ini dengan akurasi 95,16%, presisi 88,42%, dan *false alarm rate* 4,9%.

Selain itu, (Btoush et al., 2026) mengkaji berbagai metode *resampling* untuk menangani dataset yang sangat tidak seimbang, di mana teknik-teknik tersebut diuji bersama beragam algoritma *machine learning* termasuk SVM. Hasil penelitian ini menegaskan bahwa masalah *False Positives* dan *False Negatives* tetap menjadi isu penting meskipun menggunakan metode klasifikasi canggih, sehingga menuntut strategi *preprocessing* yang lebih optimal seperti penerapan teknik reduksi dimensi dan penanganan data *imbalance*.

Penelitian (Sudha & Akila, 2021) menganalisis performa algoritma SVM dan *Random Forrest Classification* dalam menklasifikasi transaksi curang. Dataset yang digunakan mencakup data transaksi kartu kredit dengan berbagai fitur yang merepresentasikan informasi transaksi dan operasional kartu. Atribut-atribut tersebut digunakan sebagai variabel masukan dalam proses klasifikasi untuk mendeteksi indikasi kecurangan transaksi. Dataset terdiri dari berbagai atribut yang mencakup informasi jumlah transaksi, jenis transaksi, dan klasifikasi apakah transaksi tersebut curang atau tidak. Dataset ini diolah untuk menyeimbangkan jumlah kelas dan memastikan bahwa model dapat menklasifikasi kecurangan dengan akurasi yang tinggi. Berdasarkan hasil pengujian, SVM menunjukkan kinerja yang optimal dalam proses klasifikasi transaksi *fraud*, terutama pada dataset

yang telah melalui proses penyeimbangan kelas. Performa model ditunjukkan oleh nilai akurasi sebesar 96%–98%, *recall* 81%–86%, *precision* 88%–94%, dan *F1-score* 84,5%–90%. Hasil tersebut menunjukkan bahwa SVM mampu menghasilkan klasifikasi yang akurat dan konsisten dalam mendeteksi transaksi yang terindikasi *fraud*.

Penelitian (Gawali et al., 2025) studi komparatif terhadap beberapa algoritma *machine learning* seperti *Logistic Regression, Support vector machine,* dan *K-Nearest Neighbors* untuk mengklasifikasi kecurangan pada dataset nyata. Penelitian ini juga memanfaatkan teknik SMOTE untuk menanggulangi ketidakseimbangan kelas yang menjadi tantangan utama dalam deteksi *fraud,* dan menunjukkan bahwa SVM mampu menghasilkan performa yang kompetitif di antara model lain dengan pengolahan data yang tepat

Berdasarkan penelitian (Fadilah et al., 2020) Penelitian tersebut mengkaji penerapan algoritma SVM dalam memprediksi harga saham PT Telekomunikasi Indonesia. Tujuan penelitian adalah menyediakan model prediksi yang dapat membantu investor dalam mengambil keputusan investasi secara lebih tepat berdasarkan pergerakan harga saham. Data yang digunakan berasal dari harga saham harian PT Telekomunikasi Indonesia yang diperoleh melalui situs finance.yahoo.com dengan periode pengamatan dari 29 April 2015 hingga 28 April 2020, yang terdiri atas 1.265 data. Hasil pengujian menunjukkan bahwa metode SVM menghasilkan performa prediksi yang lebih baik dibandingkan metode KNN, dengan tingkat akurasi sebesar 96,41% dan nilai *Root Mean Square Error* (RMSE) sebesar 0,0932. Sementara itu, KNN memperoleh akurasi sebesar 94,5% dengan

RMSE sebesar 0,1162. Temuan tersebut menunjukkan bahwa SVM mampu memberikan tingkat akurasi yang lebih tinggi serta kesalahan prediksi yang lebih rendah, sehingga berpotensi digunakan sebagai acuan dalam mendukung pengambilan keputusan investasi saham.

(Feng & Kim, 2024) menyajikan tinjauan tentang berbagai sistem deteksi kecurangan kartu kredit yang menggunakan *machine learning*. Penelitian ini menekankan bahwa model yang efektif harus dapat menangani data berdimensi tinggi dan pola *fraud* yang terus berubah, serta semakin menguatkan kebutuhan pendekatan yang bukan hanya berbasis akurasi, tetapi juga *feature engineering* dan reduksi dimensi. Beberapa Penelitian terdahulu yang memiliki kesamaan metode dengan penelitian ini ditunjukkan pada tabel 2.1:

Tabel 2. 1 Penelitian Terkait

| No | Peneliti | Judul | Metode | Hasil | Perbedaan |
|---|---|---|---|---|---|
| 1. | (Gedela & Karthikeyan, 2023) | *Credit Card Fraud Detection Using Support vector machine Algorithm in Comparison with Various Machine learning Algorithms* | *Support vector machine.* | Hasil dari penelitian tersebut menunjukkan bahwa algoritma SVM memiliki akurasi 99,32%, sensitivitas 99,67%, spesifisitas 98,97%, presisi 98,98%, dan *f-score* 99,33%. | Dalam jurnal ini peneliti menggunakan objek berbeda yang terdiri dari data kecurangan kartu kredit pada tahun 2017. |
| 2. | (Sadineni, 2020) | *Detection of Fraudulent Transactions in Credit Card using Machine learning Algorithms* | *Artificial Neural Network, Decision Trees, Support vector machine, Logistic* | SVM menghasilkan performa dengan *accuracy* 95,16%, *Precision* 88,42%, dan *false alarm rate* 4,9%. | Menggunakan beberapa metode yang berbeda dengan penelitian sebelumnya. |

| No | Peneliti | Judul | Metode | Hasil | Perbedaan |
|---|---|---|---|---|---|
| 3. | (Sudha & Akila, 2021) | *Credit Card Fraud Detection System based on Operational & Transaction features using SVM and Random Forest Classifiers* | *Support vector machine* dan *Random Forrest* | SVM menghasilkan klasifikasi transaksi curang yang cukup baik, terutama pada dataset yang seimbang setelah proses praproses data. | Peneliti membandingkan beberapa metode untuk melakukan klasifikasi kecurangan kartu kredit yakni *Support vector machine* dan *Random Forrest* |
| 4. | (Fadilah et al., 2020) | Analisis Prediksi Harga Saham PT. Telekomunikasi Indonesia Menggunakan Metode *Support vector machine* | *Support vector machine* | Hasil dari penelitian ini menunjukkan metode SVM adalah pilihan yang lebih baik untuk prediksi harga saham PT. Telekomunikasi Indonesia dengan akurasi 96,41% dan RMSE 0.0932 dibandingkan dengan KNN yang hanya memiliki akurasi 94,5% dan RMSE 0.1162 | penelitian ini melakukan analisis prediksi harga saham pada PT. Telekomuikasi Indonesia, dataset dikumpulkan dari situs finace.yahoo.com dengan total 1265 record data dengan 7 fitur yang disesuaikan. |
| | (Zhang et al., 2020) | *Credit Card Fraud Detection Using Weighted Support vector machine* | *Support vector machine (SVM)* | Tiga algoritma umum digunakan sebagai pembanding: *Logistic Regression (LR)***,** *Random Forest (RF)***,** *SVM***.** SVM mencapai presisi 89.5% dan *recall* 61.6%**,** serta financial recovery sebesar 47.5%**,** yang lebih tinggi dibandingkan model lain | Dataset yang digunakan dalam penelitian ini berasal dari data transaksi kartu kredit yang tersedia secara publik di Kaggle.com, terdiri dari 284,807 transaksi yang terjadi di sebuah bank Eropa selama dua hari di bulan September 2013 |

## 2.2 Kecurangan Kartu Kredit

*Fraud menurut Black's Law Dictionary didefinisikan sebagai tindakan yang dilakukan secara sengaja melalui perilaku tidak jujur atau penyalahgunaan kepercayaan untuk memperoleh keuntungan, khususnya dalam bentuk keuntungan finansial* (Gee, 2014). Dalam konteks perbankan, Peraturan Otoritas Jasa Keuangan Republik Indonesia Nomor 39/POJK.03/2019 menjelaskan bahwa *fraud* merupakan perbuatan penyimpangan atau kelalaian yang dilakukan secara sadar dengan tujuan mengelabui, menipu, atau memanipulasi pihak lain, baik bank maupun nasabah. Perbuatan tersebut mengakibatkan kerugian bagi pihak yang terdampak dan memberikan manfaat ekonomi kepada pelaku, baik secara langsung maupun tidak langsung. Penipuan kartu kredit diklasifikasikan dalam 2 kategori yakni penipuan *offline* dan *online* yakni penipuan kartu kredit yang dilakukan secara *offline* biasanya dilakukan dengan mencuri atau memanipulasi korban untuk menyerahkan data terkait kartu kredit miliknya. Sedangkan penipuan *online* yakni Penipuan yang dilakukan oleh pemegang kartu melalui Internet, telepon, dll. Contohnya seperti, *telecom fraud*, *computer intrusion*, *bankruptcy fraud*, dan *application fraud* (Kumar Manjhvar & Goyal, 2020).

## 2.3 *Machine learning*

*Machine learning* adalah bidang dalam ilmu komputer yang mempelajari pengembangan model dan algoritma yang memungkinkan komputer melakukan proses pembelajaran berdasarkan data yang tersedia. Melalui proses tersebut, sistem dapat mengenali pola, membuat prediksi, atau mengambil keputusan tanpa harus diprogram secara eksplisit untuk setiap kondisi. Oleh karena itu, *machine learning*

banyak diterapkan dalam penyelesaian masalah yang memiliki tingkat kompleksitas tinggi dan sulit diselesaikan dengan pendekatan pemrograman tradisional. Aplikasi sehari-hari seperti pencarian internet, pengenalan suara, dan banyak lagi telah melihat kemajuan dalam *machine learning*(Rebala et al., 2019). *Machine learning* sangat penting dalam klasifikasi penipuan, khususnya penipuan kartu kredit. Bank menggunakan berbagai teknik *machine learning* untuk memprediksi penipuan dengan menggunakan data historis dan fitur baru untuk meningkatkan ketepatan prediksi (Moumeni et al., 2022)

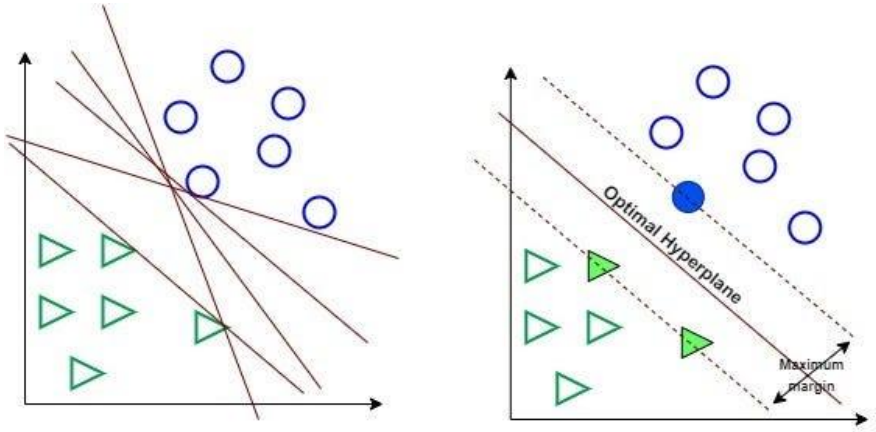
### *2.4 Support vector machine*

*Support vector machine* (SVM) adalah metode klasifikasi berbasis *supervised learning* yang dirancang untuk membangun model pemisah antar kelas secara optimal. Algoritma yang diperkenalkan oleh Cortes dan Vapnik (1995) ini bekerja dengan mencari *hyperplane* terbaik yang mampu memaksimalkan jarak pemisah *margin* antara kelas yang berbeda. Kemampuan tersebut menjadikan SVM sebagai salah satu metode yang efektif dan banyak digunakan dalam berbagai aplikasi klasifikasi, termasuk deteksi kecurangan transaksi kartu kredit.

Prinsip kerja SVM didasarkan pada pencarian *hyperplane* yang mampu memisahkan data dari kelas yang berbeda secara optimal dengan memaksimalkan nilai *margin*. Namun, pada kasus data yang tidak bersifat linear, pemisahan antar kelas menjadi lebih sulit dilakukan. Untuk mengatasi hal tersebut, SVM menerapkan *kernel trick*, yaitu mekanisme pemetaan data ke ruang berdimensi lebih tinggi sehingga pola yang sebelumnya tidak dapat dipisahkan secara linear menjadi lebih mudah dibedakan. Melalui pendekatan ini, SVM dapat menghasilkan model

klasifikasi yang lebih efektif dalam menangani data dengan karakteristik yang kompleks.(Zhang et al., 2020). *Posisi hyperplane* optimal ditentukan oleh *support vectors*, yaitu titik-titik data yang berada paling dekat dengan batas pemisah antar kelas. Data-data tersebut memiliki peran penting karena secara langsung memengaruhi letak dan orientasi *hyperplane* yang dibentuk oleh algoritma SVM.

Penelitian terbaru menunjukkan bahwa SVM memiliki performa yang kompetitif dalam klasifikasi transaksi curang, terutama dalam menangani dataset yang relatif besar. Studi yang dilakukan oleh (Du, n.d.) menemukan bahwa penerapan *probabilistic* SVM dalam klasifikasi transaksi mencurigakan mampu meningkatkan akurasi hingga 89.7% dengan tingkat *False Positive rate* yang lebih rendah dibandingkan metode klasifikasi lainnya.




*Gambar 2. 1 Konsep Optimasi Hyperplane*

Gambar 2.1 menunjukkan konsep optimasi *hyperplane* dalam SVM, di mana garis hitam tengah adalah *hyperplane* optimal yang memisahkan dua kelas data. Garis putus-putus di kedua sisi *hyperplane* menunjukkan margin maksimal, yang ditentukan oleh *support vector*. *Support vector* adalah titik data dalam dataset yang terletak paling dekat dengan *hyperplane*. Model SVM berusaha mencari

*hyperplane* dengan margin terluas, sehingga dapat mengklasifikasikan data dengan lebih akurat.

Dalam sebuah dataset himpunan data latih didefinisikan sebagai $xi = \{x1, \dots, xn\}, xi \in Rn$. Sementara itu label kelasnya dinyatakan sebagai $yi \in \{+1, -1\}$, $untuk\ i = 1,2, \dots, n$, di mana $n$ merepresentasikan jumlah total data. Persamaan umum yang digunakan untuk menggambarkan *hyperplane* yang memisahkan dua kelas dapat dirumuskan sebagai berikut.

$x \cdot w+b$ =0, untuk $y_i$ = 0 (*hyperplane*) (2.1)

$xi \cdot w + b \geq +1$, untuk $yi$ = +1 (positive class) (2.2)

$xxi \cdot w + b \leq -1$, untuk $yi$ = -1 (negative class) (2.3)

Keterangan:

$w$ = nilai bobot support vector yang terdekat dengan *hyperplane*

$xi$ = data ke-i

$b$ = nilai bias

$yi$ = kelas data ke-i

Vektor bobot ($w$) merupakan vektor yang tegak lurus terhadap *hyperplane* dan mengarah dari titik pusat koordinat. Sementara itu, bias ($b$) menunjukkan posisi relatif *hyperplane* terhadap titik koordinat. Nilai ($w$) dapat diperoleh melalui persamaan 2.4, sedangkan nilai ($b$) dihitung menggunakan persamaan 2.5.

$$w = \sum \alpha_i y_i x_i \quad (2.4)$$

$$b = -\frac{1}{2}( w.x^{+} + w.x^{-}) \quad (2.5)$$

Keterangan:

$b$ = nilai bias

$w \cdot x^{+}$ = nilai bobot untuk kelas data positif

$w \cdot x^{-}$ = nilai bobot untuk kelas data negatif

$w$ = bobot vektor

$\alpha i$= nilai bobot data ke-i

$y\ i$ = kelas data ke-i

$xi$ = data ke-i

Untuk memperoleh margin maksimal, dapat dilakukan dengan cara memperbesar jarak antara *hyperplane* dan titik data terdekat menggunakan rumus:

$$\frac{1-b-(1-b)}{w} = \frac{1}{||w||} \tag{2.6}$$

Pendekatan ini dapat diformulasikan sebagai masalah *quadratic programming* (QP), di mana proses optimasi dilakukan dengan meminimalkan persamaan 2.7 dengan tetap memenuhi batasan yang dinyatakan dalam persamaan 2.8.

Secara matematis, memaksimalkan $\frac{1}{||w||}$ setara dengan meminimalkan $||w||^2$sehingga optimasi dilakukan dengan:

$$^{min}_{w}\tau(w) = \frac{1}{2}\ ||w||^2 \tag{2.7}$$

Dengan $||w||$sebagai vektor normal *hyperplane*.

Batasan dalam optimasi ini diberikan oleh:

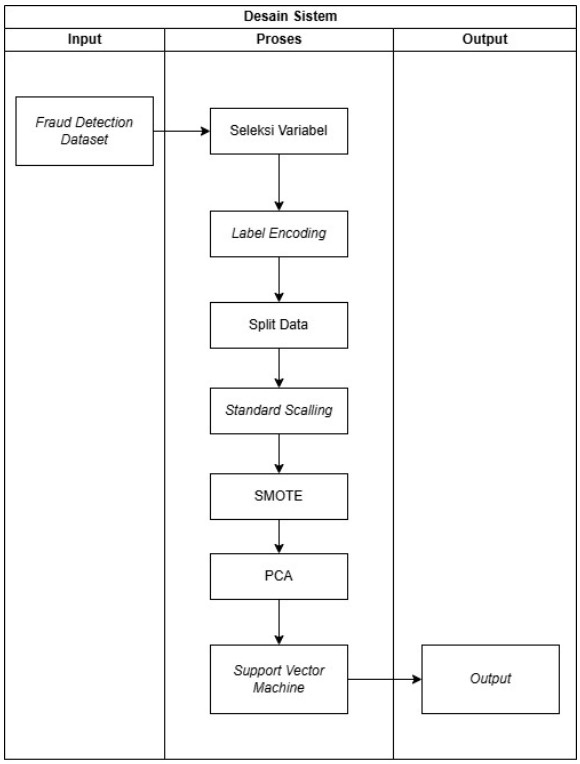
$$y_i(x_i.w + b) - 1 \geq 0, \text{untuk } i = 1, \ldots, n \tag{2.8}$$

Yang memastikan bahwa setiap data berada di sisi *hyperplane* yang sesuai berdasarkan label kelasnya.

# BAB III
# DESAIN DAN IMPLEMENTASI

## 3.1 Desain Sistem

Desain sistem berfungsi sebagai gambaran konseptual yang menjelaskan tahapan penelitian secara terstruktur, mulai dari proses pra-pemrosesan data hingga tahap klasifikasi dan evaluasi hasil menggunakan model *machine learning*. Dengan adanya desain sistem yang jelas, setiap tahapan penelitian dapat dipahami secara runtut serta memudahkan evaluasi terhadap metode yang digunakan. Gambar 3.1 menunjukkan desain sistem yang diterapkan dalam penelitian ini, yang dimulai dari proses pengolahan data hingga evaluasi hasil klasifikasi yang dihasilkan oleh model.




*Gambar 3. 1 Desain Sistem*

Tujuan penelitian ini yakni merancang sistem klasifikasi kecurangan transaksi kartu kredit menggunakan bahasa pemrograman *Python*. Sistem ini memanfaatkan dataset publik yang tersedia di situs *Kaggle*, yang berjudul *Fraud Detection Dataset* yang terdiri dari berbagai atribut yang merepresentasikan transaksi kartu kredit. Data tersebut melalui proses *preprocessing* yang mencakup Seleksi variabel untuk memilih fitur yang sesuai dalam proses pemodelan. Dilanjutkan dengan proses *Label Encoding* untuk mengubah data kategorikal menjadi representasi numerik sehingga dapat diproses oleh algoritma *machine learning*. Data kemudian dibagi ke dalam data *training* dan data *testing* (split data) untuk mengevaluasi kemampuan generalisasi model. Selanjutnya proses *Standard Scaling* yang bertujuan untuk menormalkan rentang nilai fitur agar distribusi data menjadi seragam. Hal ini penting terutama pada algoritma berbasis jarak seperti SVM agar tidak terjadi dominasi fitur tertentu.

Setelah proses standardisasi data, dilakukan penanganan ketidakseimbangan kelas menggunakan SMOTE. Teknik ini menghasilkan data sintetis pada kelas minoritas berdasarkan karakteristik data yang telah ada, sehingga distribusi kelas menjadi lebih proporsional. Dengan meningkatnya representasi kelas minoritas dalam ruang fitur, model diharapkan mampu mempelajari pola *fraud* secara lebih efektif dan meningkatkan performa deteksi terhadap transaksi yang terindikasi kecurangan. (Chawla et al., 2002). lalu mereduksi dimensi data dengan menggunakan *Principal Component Analysis* (PCA). Penggunaan PCA dalam pipeline ini bertujuan meningkatkan efisiensi komputasi sekaligus menjaga informasi utama dari data (Han, 2024). Pada tahap akhir, data yang telah melalui

seluruh proses pra-pemrosesan digunakan untuk membangun model klasifikasi menggunakan SVM. Metode ini bekerja dengan menentukan *hyperplane* optimal yang mampu memisahkan data antar kelas secara efektif. Apabila pola data tidak dapat dipisahkan secara linear, SVM memanfaatkan fungsi kernel untuk memetakan data ke ruang fitur yang lebih tinggi sehingga proses klasifikasi dapat dilakukan dengan lebih akurat. Secara keseluruhan desain sistem ini mengintegrasikan beberapa teknik *preprocessing* data, penanganan ketidakseimbangan data, reduksi dimensi, serta algoritma klasifikasi untuk meningkatkan performa deteksi *fraud*.

Penelitian ini menerapkan kernel *linear* pada algoritma *Support vector machine* sebagai fungsi pemisah antar kelas. Kinerja model kemudian dievaluasi untuk mengetahui kemampuan klasifikasi dalam mengidentifikasi transaksi yang tergolong *fraud* maupun *non-fraud*. Proses evaluasi dilakukan menggunakan *confusion matrix* yang menggambarkan hasil prediksi model terhadap data aktual. Selanjutnya, nilai *accuracy*, *precision*, *recall*, dan *F1-score* dihitung berdasarkan *confusion matrix* untuk memberikan gambaran yang lebih komprehensif mengenai performa model yang dihasilkan.

### 3.2 Pengumpulan Data

Setelah menentukan topik, langkah berikutnya adalah mencari data yang diperlukan untuk melakukan penelitian. Pada penelitian ini data diambil dari *Kaggle Repository : Fraud Detection Dataset*.

Data ini merupakan *open access* data *website Kaggle* (*Fraud* Detection Dataset). Dataset "*Fraud Detection Dataset*" yang disediakan oleh Ranjeet

Shrivastav di *kaggle* bertujuan untuk membangun model prediktif yang dapat menentukan apakah suatu transaksi merupakan kecurangan atau tidak. Dataset ini bersifat sintetis, yang berarti data transaksi dihasilkan secara artifisial menggunakan teknik simulasi atau generator data. Pendekatan ini umum digunakan untuk mengatasi keterbatasan akses ke data transaksi nyata karena alasan privasi dan keamanan. Dataset ini berisi 786.363 baris dan 29 kolom dengan jumlah transaksi *fraud* 1,58% dari jumlah data keseluruhan. Untuk keperluan meminimalkan waktu komputasi peneliti mengambil 5000 data dengan perbandingan *fraud* sebesar 1,7% dan *non-fraud* sebesar 98,3%

Tahap awal penelitian dilakukan dengan memeriksa kualitas data untuk mengidentifikasi keberadaan *missing values* pada dataset. *Missing values* merupakan nilai yang tidak tersedia pada suatu atribut dan berpotensi memengaruhi hasil analisis apabila tidak ditangani dengan baik. Proses identifikasi dilakukan menggunakan pustaka *Pandas* pada *Python* yang menyediakan fitur pengolahan data secara efektif dan mudah digunakan.

Tabel 3. 1. Atribut Dataset

| No. | Nama Atribut | Penjelasan |
|---|---|---|
| 1 | *accountNumber* | Nomor akun pelanggan |
| 2 | customerId | ID unik pelanggan |
| 3 | creditLimit | Batas kredit yang diberikan kepada pelanggan. |
| 4 | *availableMoney* | Jumlah uang yang tersedia untuk digunakan oleh pelanggan |
| 5 | *transactionDateTime* | Tanggal dan waktu transaksi dilakukan |
| 6 | *transactionAmount* | Jumlah uang yang ditransaksikan |
| 7 | *merchantName* | Nama tempat transaksi dilakukan |
| 8 | *acqCountry* | Negara tempat akuisisi transaksi |
| 9 | *merchantCountryCode* | Kode negara pedagang. |
| 10 | *posEntryMode* | Mode entri POS (*Point of Sale*). |
| 11 | *posConditionCode* | Kode kondisi POS. |
| 12 | *merchantCategoryCode* | Kode kategori pedagang. |
| 13 | *currentExpDate* | Tanggal kedaluwarsa kartu saat ini. |
| 14 | *accountOpenDate* | Tanggal pembukaan akun. |

| No. | Nama Atribut | Penjelasan |
|---|---|---|
| 15 | *dateOfLastAddressChange* | Tanggal terakhir perubahan alamat |
| 16 | *cardCVV* | CVV asli yang tertera pada kartu kredit. |
| 17 | *enteredCVV* | CVV yang dimasukkan oleh pengguna saat melakukan transaksi. Jika cocok dengan cardCVV, maka transaksi lebih mungkin valid. |
| 18 | *cardLast4Digits* | Empat digit terakhir nomor kartu |
| 19 | *transactionType* | Jenis transaksi (misalnya, PURCHASE) |
| 20 | *echoBuffer* | Buffer tambahan (tidak selalu terisi). |
| 21 | *currentBalance* | Saldo saat ini pada akun. |
| 22 | *merchantCity* | Kota pedagang |
| 23 | *merchantState* | Negara bagian pedagang. |
| 24 | *merchantZip* | Kode pos pedagang. |
| 25 | *cardPresent* | Indikator apakah kartu fisik hadir saat transaksi. |
| 26 | *posOnPremises* | Indikator lokasi POS. |
| 27 | *recurringAuthInd* | Indikator otorisasi berulang |
| 28 | *expirationDateKeyInMatch* | Kecocokan tanggal kedaluwarsa yang dimasukkan. |
| 29 | *isFraud* | Indikator apakah transaksi tersebut merupakan kecurangan (true/false). |

Tabel 3.1 menyajikan penjelasan atribut dalam dataset beserta penjelasan fungsinya dalam konteks deteksi kecurangan transaksi kartu kredit. Setiap atribut merepresentasikan karakteristik tertentu yang berkaitan dengan identitas akun, jumlah saldo, informasi transaksi, serta validasi kartu dan lokasi *merchant*. Atribut-atribut tersebut dipilih karena memiliki potensi untuk menggambarkan pola transaksi yang dapat membedakan antara transaksi normal dan transaksi *fraud*.

Atribut numerik seperti *creditLimit, availableMoney, transactionAmount,* dan *currentBalance* digunakan untuk merepresentasikan kondisi finansial akun sebelum dan sesudah transaksi, yang sering menjadi indikator utama dalam mendeteksi transaksi dengan nilai yang tidak wajar. Sementara itu, atribut kategorikal seperti *merchantName, transactionType,* dan *merchantCategoryCode* digunakan untuk menangkap karakteristik perilaku transaksi berdasarkan jenis merchant dan tipe transaksi. Selain itu, atribut yang berkaitan dengan validasi kartu

dan transaksi, seperti *cardCVV, enteredCVV, expirationDateKeyInMatch,* serta *cardPresent*, memiliki peran penting dalam mengidentifikasi potensi penyalahgunaan kartu kredit.

Tabel 3. 2. Contoh Dataset

| posEntryMode | posConditionCode | merchantCategoryCode | currentExpDate | accountOpenDate | dateOfLastAddressChange |
|---|---|---|---|---|---|
| 9 | 8 | health | Oct-21 | 27/07/2015 | 25/06/2016 |
| 9 | 1 | online_retail | Sep-28 | 19/12/2015 | 19/12/2015 |
| 9 | 1 | health | May-28 | 19/12/2015 | 19/12/2015 |
| 9 | 99 | online_retail | Mar-29 | 19/12/2015 | 10/06/2016 |
| 9 | 1 | food_delivery | Aug-24 | 06/08/2015 | 06/08/2015 |
| 9 | 1 | food_delivery | Aug-24 | 06/08/2015 | 06/08/2015 |
| 9 | 1 | food_delivery | Aug-24 | 06/08/2015 | 06/08/2015 |
| 2 | 1 | auto | Dec-31 | 13/10/2015 | 13/10/2015 |
| 9 | 1 | auto | Sep-27 | 13/10/2015 | 13/10/2015 |

Tabel 3.2 menampilkan contoh data transaksi dalam dataset pada penelitian ini. Penyajian contoh data bertujuan untuk memberikan gambaran nyata mengenai format, tipe nilai, serta variasi data pada masing-masing atribut. Dari tabel tersebut dapat diamati transaksi kartu kredit yang mencakup informasi akun, nilai transaksi, lokasi merchant, serta status validasi kartu.

Melalui contoh data yang ditampilkan terlihat bahwa atribut numerik memiliki rentang nilai yang cukup luas dan heterogen. Variasi ini mencerminkan perbedaan karakteristik finansial antar nasabah, seperti limit kredit, frekuensi penggunaan kartu, serta nominal transaksi yang dilakukan. Kondisi ini mengindikasikan bahwa data memiliki skala yang tidak seragam, sehingga pada tahap *preprocessing* diperlukan teknik normalisasi atau standarisasi agar model klasifikasi dapat bekerja secara optimal dan tidak bias terhadap atribut dengan skala besar. Atribut kategorikal menunjukkan tingkat keragaman yang tinggi, seperti pada nama merchant, jenis transaksi, serta lokasi transaksi. Variasi ini

mencerminkan perilaku penggunaan kartu kredit yang berbeda-beda pada setiap individu. Algoritma SVM mengharuskan seluruh data masukan berada dalam format numerik. Oleh sebab itu, atribut yang masih berbentuk kategorikal perlu ditransformasikan terlebih dahulu melalui teknik *label encoding*. Proses ini dilakukan dengan memberikan nilai numerik pada setiap kategori sehingga data dapat diolah dan digunakan dalam tahap pelatihan maupun pengujian model.

Analisis distribusi data menunjukkan adanya fenomena *class imbalance*, yaitu kondisi ketika jumlah sampel pada kelas *non-fraud* jauh melebihi jumlah sampel pada kelas *fraud*. Ketidakseimbangan distribusi kelas ini dapat memengaruhi proses pembelajaran model karena algoritma cenderung mengoptimalkan prediksi pada kelas mayoritas. Akibatnya, kemampuan model dalam mengenali dan mengklasifikasikan transaksi *fraud* dapat menurun meskipun nilai akurasi keseluruhan terlihat tinggi. Sehingga, penelitian ini mengintegrasikan teknik penanganan *imbalance data* untuk meningkatkan representasi kelas minoritas dalam proses pelatihan model. Selain permasalahan ketidakseimbangan data, kompleksitas dataset yang tinggi juga disebabkan oleh banyaknya jumlah atribut yang digunakan. Dimensi data yang besar dapat meningkatkan risiko terjadinya *overfitting* serta menambah beban komputasi pada proses pelatihan model. Untuk mengatasi hal tersebut, penelitian ini menggunakan teknik reduksi dimensi menggunakan *Principal Component Analysis* (PCA). PCA digunakan untuk mengekstraksi fitur-fitur utama yang mampu merepresentasikan sebagian besar variasi data, sehingga informasi penting tetap dipertahankan dengan jumlah dimensi yang lebih rendah.

Berdasarkan penjelasan atribut dan contoh data transaksi yang disajikan pada tabel-tabel tersebut, dapat disimpulkan bahwa dataset memiliki karakteristik multidimensi yang mencakup aspek finansial, perilaku, serta konteks transaksi. Kompleksitas dan keberagaman data ini menjadi dasar dalam pemilihan metode *preprocessing* dan algoritma klasifikasi pada penelitian ini. Kombinasi antara teknik normalisasi, encoding, penanganan ketidakseimbangan data, serta reduksi dimensi diharapkan mampu menghasilkan representasi data yang lebih optimal.

*Support vector machine* (SVM) dipilih sebagai metode klasifikasi pada penelitian ini karena memiliki kemampuan yang baik dalam mengolah data dengan jumlah fitur yang besar. Selain itu, SVM mampu membentuk batas pemisah yang optimal antar kelas sehingga proses klasifikasi dapat dilakukan secara lebih efektif. Kemampuan tersebut menjadikan SVM sesuai untuk digunakan dalam membedakan transaksi *fraud* dan *non-fraud* pada data transaksi kartu kredit. Dengan dukungan teknik *preprocessing* yang tepat, model SVM diharapkan mampu mengidentifikasi pola-pola kompleks pada data transaksi, khususnya dalam membedakan antara transaksi *fraud* dan *non-fraud* secara lebih akurat. Dengan demikian, keseluruhan tahapan yang dilakukan dalam penelitian ini dirancang untuk mengatasi karakteristik data yang kompleks sekaligus meningkatkan kinerja model dalam mendeteksi kecurangan transaksi kartu kredit.

### 3.3 *Preprocessing*

*Preprocessing* data merupakan tahap krusial dalam pengolahan data yang bertujuan untuk mempersiapkan dan menyempurnakan data sebelum digunakan dalam proses pemodelan. Langkah ini sangat penting karena kualitas data secara

langsung memengaruhi kinerja model yang akan digunakan dalam analisis dan prediksi. Data yang belum melalui tahap *preprocessing* sering kali mengandung *noise*, nilai yang hilang, tidak konsisten, atau format yang tidak sesuai, yang dapat menyebabkan kesalahan dalam interpretasi dan hasil analisis yang kurang akurat. Dalam penelitian ini, peneliti menggunakan bebererapa teknik seleksi variabel, *label encoding, standard scalling, Principal Component Analysis,* dan *Synthetic Minority Over-sampling Technique.*

### 3.3.1 Seleksi Variabel

Seleksi variabel merupakan salah satu tahapan *preprocessing* yang dilakukan setelah analisis awal dataset dan sebelum penerapan metode klasifikasi. Tahap ini bertujuan untuk menyeleksi atribut-atribut yang relevan dan menghapus atribut yang tidak memberikan kontribusi yang cukup besar terhadap proses klasifikasi. Pemilihan fitur yang tepat diharapkan dapat meningkatkan akurasi model serta mengurangi kompleksitas data yang diproses. tahapan ini dilakukan dengan menghapus atribut-atribut yang memiliki tingkat *missing value* tinggi serta atribut yang tidak memiliki kontribusi berarti terhadap proses klasifikasi *fraud*. tahap ini bertujuan untuk mempertahankan sebanyak mungkin data transaksi yang valid, sekaligus menjaga kualitas informasi yang relevan bagi model klasifikasi.

Tabel 3.3 menunjukkan data sebelum dilakukan proses seleksi variabel. Masih terdapat beberapa atribut yang tidak diperlukan dan memliki *missing value* sehingga dapat berpengaruh pada hasil klasifikasi. Oleh karena itu tahapan ini dilakukan untuk menghapus atribut yang tidak sesuai agar model klasifikasi dapat bekerja secara optimal.

Tabel 3. 3. Dataset Sebelum Proses Seleksi variabel

| No | Atribut | Jumlah |
|---|---|---|
| 0 | *Unnamed: 0* | 786363 |
| 1 | *accountNumber* | 786363 |
| 2 | *customerId* | 786363 |
| 3 | *creditLimit* | 786363 |
| 4 | *availableMoney* | 786363 |
| 5 | *transactionDateTime* | 786363 |
| 6 | *transactionAmount* | 786363 |
| 7 | *merchantName* | 786363 |
| 8 | *acqCountry* | 786363 |
| 9 | *merchantCountryCode* | 786363 |
| 10 | *posEntryMode* | 786363 |
| 11 | *posConditionCode* | 786363 |
| 12 | *merchantCategoryCode* | 786363 |
| 13 | *currentExpDate* | 786363 |
| 14 | *accountOpenDate* | 786363 |
| 15 | *dateOfLastAddressChange* | 786363 |
| 16 | *cardCVV* | 786363 |
| 17 | *enteredCVV* | 786363 |
| 18 | *cardLast4Digits* | 786363 |
| 19 | *transactionType* | 786363 |
| 20 | *echoBuffer* | 0 |
| 21 | *currentBalance* | 786363 |
| 22 | *merchantCity* | 0 |
| 23 | *merchantState* | 0 |
| 24 | *merchantZip* | 0 |
| 25 | *cardPresent* | 786363 |
| 26 | *posOnPremises* | 0 |
| 27 | *recurringAuthInd* | 0 |
| 28 | *expirationDateKeyInMatch* | 786363 |
| 29 | *isFraud* | 786363 |

Atribut yang dihapus pada tahap seleksi atribut meliputi *Unnamed: 0*, *merchantCity*, *merchantState*, *merchantZip*, *echoBuffer*, *posOnPremises*, dan *recurringAuthInd*. Atribut *Unnamed: 0* dihapus karena merupakan indeks data yang tidak memiliki makna analitis. Atribut lokasi merchant seperti *merchantCity*, *merchantState*, *merchantName* dan *merchantZip* memiliki tingkat *missing value* yang tinggi serta bersifat redundan dengan atribut lokasi lainnya, sehingga berpotensi menimbulkan *noise* pada proses klasifiksi model. Sementara itu, atribut *echoBuffer*, *posOnPremises*, dan *recurringAuthInd* merupakan atribut teknis dan kontekstual yang tidak selalu muncul pada setiap transaksi, serta tidak secara

langsung merepresentasikan pola utama transaksi *fraud*. penelitian ini juga melakukan penghapusan atribut bertipe *datetime* yang tidak digunakan secara langsung dalam proses klasifikasi. Atribut-atribut tersebut meliputi tanggal dan waktu transaksi serta informasi waktu administratif lainnya. Penghapusan atribut ini dilakukan untuk memastikan bahwa dataset yang digunakan dalam proses *standard scaling* hanya terdiri dari fitur-fitur numerik yang kompatibel dengan algoritma *Support vector machine*. Tabel 3.4 menyajikan atribut setelah tahapan fiture selection dilakukan. Melalui seleksi atribut, penelitian ini berhasil menyeimbangkan antara menjaga kuantitas data dan mempertahankan kualitas informasi yang relevan untuk deteksi *fraud*.

Tabel 3. 4. Dataset Setelah Proses Seleksi variabel

| No | Atribut | Jumlah |
|---|---|---|
| 1 | *accountNumber* | 786363 |
| 2 | *customerId* | 786363 |
| 3 | *creditLimit* | 786363 |
| 4 | *availableMoney* | 786363 |
| 5 | *transactionAmount* | 786363 |
| 6 | *posEntryMode* | 786363 |
| 7 | *posConditionCode* | 786363 |
| 8 | *merchantCategoryCode* | 786363 |
| 9 | *cardCVV* | 786363 |
| 10 | *enteredCVV* | 786363 |
| 11 | *cardLast4Digits* | 786363 |
| 12 | *transactionType* | 786363 |
| 13 | *currentBalance* | 786363 |
| 14 | *cardPresent* | 786363 |
| 15 | *expirationDateKeyInMatch* | 786363 |
| 16 | *isFraud* | 786363 |

### 3.3.1.2 Label Encoding

Salah satu tahapan penting dalam proses *preprocessing* adalah transformasi data kategorikal melalui teknik *label encoding*. Teknik ini digunakan untuk mengubah data yang berbentuk kategori atau label menjadi representasi numerik

sehingga dapat diproses oleh algoritma *machine learning*. Fitur kategorikal umumnya berupa atribut yang memiliki nilai dalam bentuk kelompok tertentu, seperti jenis transaksi, lokasi transaksi, atau metode pembayaran yang digunakan. Karena sebagian besar algoritma *machine learning*, termasuk SVM, memerlukan data dalam format numerik, maka proses *label encoding* perlu dilakukan agar informasi yang terkandung dalam atribut kategorikal dapat dimanfaatkan secara optimal selama proses pelatihan model.

Metode *label encoding* pada penelitian ini digunakan melalui fungsi *LabelEncoder* dari pustaka *scikit-learn*. Teknik ini digunakan untuk mengonversi fitur kategorikal yang terdapat dalam dataset, salah satunya pada atribut "*Result*", yang berfungsi untuk menunjukkan hasil klasifikasi apakah suatu transaksi dikategorikan sebagai *fraud* atau *non-fraud*. Proses *encoding* dilakukan dengan menggantikan kategori dengan nilai numerik, misalnya:

1. *Fraudulent Transaction* (Transaksi Curang) → 1
2. *Legitimate Transaction* (Transaksi Normal) → 0

Dengan adanya konversi ini, model dapat menginterpretasikan kategori dalam bentuk angka tanpa mengubah makna dari data itu sendiri. Sebelum dilakukan proses *label encoding*, dataset masih mengandung beberapa atribut bertipe kategorikal yang direpresentasikan dalam bentuk teks atau simbol. Atribut-atribut tersebut mencerminkan karakteristik transaksi, seperti tipe transaksi, kategori *merchant*, dan kondisi perangkat transaksi. Meskipun informasi yang terkandung pada atribut tersebut bersifat penting, representasi non-numerik menyebabkan dataset belum dapat di eksekusi langsung oleh algoritma *Support*

*vector machine*. Tabel 3.5 menyajikan data sebelum tahapan *label encoding* dilakukan.

*Table 3.5 Data sebelum proses Label Encoding*

| No. | *merchantCategoryCode* | *transactionType* |
|---|---|---|
| 1. | *rideshare* | *PURCHASE* |
| 2. | *entertainment* | *PURCHASE* |
| 3. | *mobileapps* | *PURCHASE* |
| 4. | *mobileapps* | *PURCHASE* |
| 5. | *fastfood* | *PURCHASE* |

Table 3.6 Data sesudah proses Label Encoding

| No. | *merchantCategoryCode* | *transactionType* |
|---|---|---|
| 1. | *16* | *1* |
| 2. | *2* | 1 |
| 3. | *11* | 1 |
| 4. | *11* | 1 |
| 5. | *3* | 1 |

Gambar 3.3 menyajikan data setelah tahapan *label encoding*. Dapat diketahui bahwa seluruh atribut kategorikal pada dataset berhasil dikonversi ke dalam bentuk numerik. Setiap nilai kategori yang sebelumnya berupa teks direpresentasikan oleh bilangan bulat tertentu sesuai dengan hasil proses *encoding*. Perubahan ini tidak menghilangkan informasi yang terkandung dalam atribut, melainkan hanya mengubah bentuk representasinya agar sesuai dengan kebutuhan algoritma klasifikasi.

#### 3.3.1.3 Standard Scaling

Dalam proses normalisasi data, salah satu metode yang sering digunakan adalah *StandardScaler*. Metode ini melakukan standarisasi nilai pada setiap fitur dengan mengubahnya ke dalam bentuk *z-score*, yaitu nilai yang menunjukkan posisi suatu data terhadap rata-ratanya berdasarkan satuan standar deviasi. Melalui proses ini, perbedaan skala antar fitur dapat diminimalkan sehingga setiap variabel

memiliki kontribusi yang lebih seimbang dalam pembentukan model. Hasil transformasi menggunakan *StandardScaler* menghasilkan data dengan rata-rata *mean* bernilai nol dan standar deviasi sebesar satu. Secara matematis, proses standarisasi tersebut dapat dinyatakan pada Persamaan 3.1.:

$$z = \frac{x - \mu}{\sigma} \tag{3.1}$$

Keterangan :
z : nilai *z-score*
x : nilai asli dari fitur
μ : nilai rata-rata dari fitur
σ : standar defiasi dari fitur

Dalam dataset transaksi kartu kredit, beberapa fitur seperti *transaction amount* dapat memiliki rentang nilai yang sangat besar dibandingkan dengan variabel lain seperti *customer age* atau *merchant category*. Standard Scaling memastikan bahwa model tidak memberikan bobot lebih besar pada fitur tertentu hanya karena perbedaan skala angka. selain itu, Data yang telah dinormalisasi memungkinkan model untuk lebih akurat dalam menklasifikasi transaksi anomali, karena model dapat lebih fokus pada pola hubungan antar fitur daripada nilai absolutnya. Tabel 3.7 menunjukkan data sebelum dilakukan proses *standard scaling*, Setiap atribut numerik dalam dataset umumnya memiliki karakteristik skala dan rentang nilai yang berbeda. Sebagai contoh, atribut *credit limit* dan *available money* cenderung memiliki nilai yang jauh lebih besar dibandingkan atribut numerik lainnya. Perbedaan tersebut dapat menyebabkan ketidakseimbangan kontribusi antar fitur selama proses pembelajaran model. Pada algoritma SVM, kondisi ini perlu diperhatikan karena SVM bekerja berdasarkan

perhitungan jarak antar data untuk menentukan *hyperplane* yang optimal. Oleh karena itu, perbedaan skala yang terlalu besar berpotensi memengaruhi performa model dan menghasilkan proses klasifikasi yang kurang optimal.

Table 3.7 Dataset sebelum proses standard scalling

| No. | *AccountNumber* | *customerId* | *creditLimit* | *AvailableMoney* | *tansactionAmount* |
|---|---|---|---|---|---|
| 1. | 737265056 | 737265056 | 5000 | 5000.0 | 98.55 |
| 2. | 737265056 | 737265056 | 5000 | 5000.0 | 74.51 |
| 3. | 737265056 | 737265056 | 5000 | 5000.0 | .7.47 |
| 4. | 737265056 | 737265056 | 5000 | 5000.0 | 7.47 |
| 5. | 830329091 | 830329091 | 5000 | 5000.0 | 71.18 |

Peneliti menerapkan metode *standard scaling* menggunakan pendekatan normalisasi berbasis distribusi normal. Proses ini mengubah atribut numerik sehingga memiliki nilai *mean* mendekati nol *standard deviation* mendekati satu. Dengan demikian, semua atribut berada pada rentang yang sama dan tidak saling mendominasi dalam proses pembelajaran model.

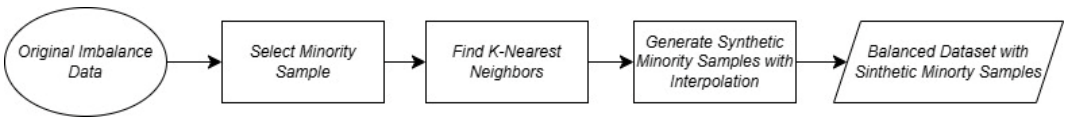
Table 3.8 Dataset setelah proses standard scalling

| No. | *AccountNumber* | *customerId* | *creditLimit* | *AvailableMoney* | *tansactionAmount* |
|---|---|---|---|---|---|
| 1. | -0.788586 | -0.788586 | -0.763877 | -0.507078 | -0.575134 |
| 2. | -0.659409 | -0.659409 | -0.616498 | -0.260709 | 0.647780 |
| 3. | 0.756324 | 0.756324 | -0.616498 | -0.671949 | -0.754850 |
| 4. | 0.040749 | -0.040749 | -0.763877 | -0.501029 | -0.888689 |
| 5. | -0.659409 | -0.659409 | -0.616498 | -0.498909 | -0.178429 |

Tabel 3.8 menyajikan hasil penerapan *standard scaling* yang menunjukkan bahwa *mean* setiap atribut mendekati nol dan *standard deviation* mendekati satu. Perubahan ini mengindikasikan bahwa proses *standard scaling* berhasil menormalkan skala data. Dataset yang telah distandarisasi ini selanjutnya digunakan pada tahap *preprocessing* berikutnya, yaitu penanganan ketidakseimbangan kelas menggunakan SMOTE dan reduksi dimensi menggunakan PCA, sebelum dilakukan proses *training* dan *testing* model SVM.

### 3.3.1.4 *Synthetic Minority Over-sampling Technique* (SMOTE)

Dataset yang digunakan oleh peneliti memiliki total 5000 transaksi, namun jumlah transaksi yang tergolong curang hanya sebesar 85 data (1,7%), sementara transaksi normal mendominasi sebanyak 4915 data (98,3%). Ketimpangan ekstrem ini dapat menyebabkan algoritma SVM memiliki bias tinggi terhadap kelas *non-fraud*, sehingga performa model dalam mendeteksi kelas *fraud* akan menurun. Alur sistem SMOTE ditunjukkan pada gambar 3.6.




Gambar 3. 2 Alur Sistem Smote

Dalam penelitian ini, penanganan ketidakseimbangan kelas dilakukan menggunakan metode SMOTE. Metode ini bekerja dengan membentuk data sintetis pada kelas *fraud* berdasarkan hubungan antar sampel yang berdekatan dalam ruang fitur. Berbeda dengan proses duplikasi data, SMOTE menghasilkan sampel baru melalui interpolasi linear antara suatu data minoritas dan salah satu *nearest neighbor*. Pendekatan tersebut memungkinkan distribusi kelas minoritas menjadi lebih representatif sehingga dapat membantu meningkatkan kemampuan model dalam mengenali pola transaksi *fraud*. Mekanisme pembentukan data sintetis oleh SMOTE dirumuskan pada Persamaan 3.2.:

$$X_{new} = X_i + rand(0,1)\ x\ (\hat{X}i - X_i) \quad (3.2)$$

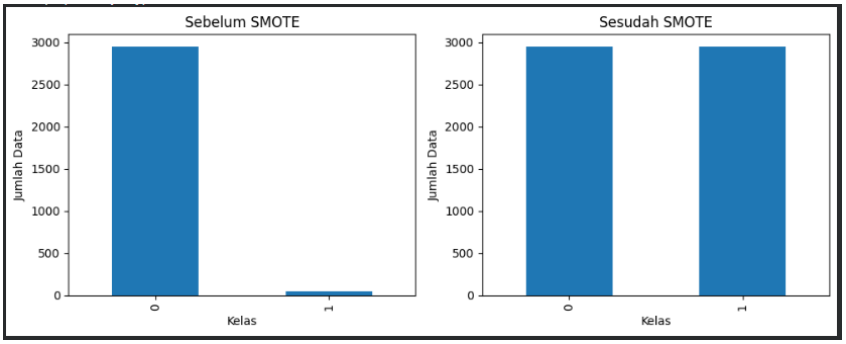
Keterangan:

$X_i$ : Titik sampel minoritas yang dipilih.

$\hat{X}i$ : Salah satu dari $X_{knn}$ dari kelas minoritas yang sama.

$rand(0,1)$ : Bilangan acak antara 0 dan 1 yang menentukan posisi titik sintetis di sepanjang garis antara dua sampel asli.

Tahap SMOTE pada penelitian ini dimulai hanya menggunakan data *training* saja data *testing* tetap menggunakan data asli. SMOTE bekerja dengan mengidentifikasi kelas untuk menghitung rasio antara transaksi sah *(isFraud: False)* dan transaksi kecurangan *(isFraud: True)* untuk menentukan tingkat ketimpangan data. Selanjutnya adalah tahap penentuan tetangga terdekat $X_{knn}$ di mana setiap sampel *fraud* dipetakan berdasarkan kedekatan fitur menggunakan perhitungan jarak *Euclidean*. Proses inti dilakukan melalui tahap interpolasi untuk menghasilkan sampel sintetis baru. Secara teknis, sistem menghitung selisih nilai antara sampel minoritas asli dengan tetangganya, lalu mengalikannya dengan bilangan acak antara 0 dan 1 untuk menciptakan titik data baru yang merepresentasikan karakteristik kecurangan. Selanjutnya dilakukan pembentukan dataset seimbang, di mana seluruh sampel sintetis digabungkan ke dalam dataset asli hingga mencapai rasio seimbang.




*Gambar 3. 3 Hasil Penerapan Smote*

Gambar 3.7 memperlihatkan distribusi kelas pada data pelatihan sebelum dan sesudah penerapan SMOTE. Sebelum proses *oversampling* dilakukan, jumlah transaksi *non-fraud* mendominasi dataset, sedangkan transaksi *fraud* hanya memiliki proporsi yang sangat kecil. Kondisi tersebut menunjukkan adanya ketidakseimbangan kelas *class imbalance* yang berpotensi memengaruhi proses pembelajaran model. Setelah SMOTE diterapkan, jumlah sampel pada kelas *fraud* meningkat secara signifikan hingga mendekati jumlah sampel pada kelas *non-fraud*. Perubahan distribusi tersebut menunjukkan bahwa SMOTE berhasil menciptakan representasi yang lebih seimbang pada data pelatihan sehingga permasalahan ketidakseimbangan kelas dapat diminimalkan.

Penerapan SMOTE dalam penelitian ini hanya dilakukan pada data latih, sedangkan data uji tetap dipertahankan dalam kondisi aslinya. Pendekatan ini bertujuan untuk mencegah terjadinya *data leakage* yang dapat menyebabkan hasil evaluasi menjadi bias dan tidak mencerminkan performa model yang sebenarnya. Dengan demikian, hasil pengujian yang diperoleh dapat menggambarkan kemampuan model dalam melakukan generalisasi terhadap data baru yang belum pernah digunakan selama proses pelatihan. Selain itu, pengaturan parameter seperti *sampling strategy* perlu diperhatikan karena berpengaruh terhadap jumlah sampel sintetis yang dihasilkan. Pemilihan rasio *oversampling* yang terlalu tinggi berpotensi mengubah karakteristik distribusi data dan dapat memengaruhi performa klasifikasi. Oleh karena itu, diperlukan strategi penyeimbangan yang proporsional agar data tetap representatif dan model mampu mempelajari pola pada kelas *fraud*

secara lebih optimal. (Chawla et al., 2002). Tabel 3.9 berikut menyajikan dataset sintetis setelah dilakukan proses SMOTE.

Table 3.9 Data sintetis hasil porses smote

| No. | *AccountNumber* | *customerId* | *creditLimit* | *AvailableMoney* | *tansactionAmount* |
|---|---|---|---|---|---|
| 3000. | -1.533796 | -1.533796 | -0.028289 | 0.162832 | 0.906577 |
| 3001. | 0.936516 | 0.936516 | -0.765878 | -0.607045 | -0.838707 |
| 3002. | 0.936516 | 0.936516 | -0.765878 | -0.698659 | -0.090204 |
| 3003. | 0.936516 | 0.936516 | -0.765878 | -0.577080 | -0.765585 |
| 3004. | 0.936516 | 0.936516 | -0.765878 | -0.675312 | -0.624553 |

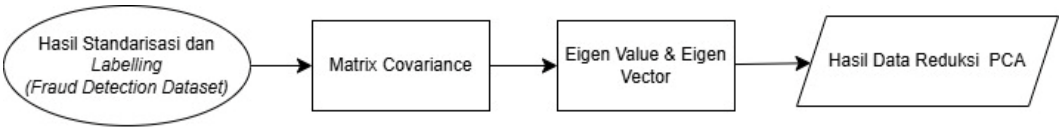
Hasil eksekusi program menunjukkan bahwa jumlah data latih (*training data*) asli sebelum dikenakan SMOTE adalah sebanyak 3000 baris. Untuk mengatasi ketimpangan kelas, algoritma berhasil membangkitkan 2.898 baris data sintetis baru secara spesifik untuk kelas minoritas (transaksi *fraud*). Penambahan kuantitas ini secara otomatis mengeliminasi kondisi *imbalanced data*, sehingga kelas minoritas kini memiliki proporsi yang setara dengan kelas mayoritas. Pada tabel sampel data sintetis, nilai-nilai numerik penyusun fiturnya (seperti pada *availableMoney accountNumber*, *creditLimit*, hingga *transactionAmount*) menampilkan angka *decimal* yang unik. Angka-angka ini bukanlah hasil replikasi dari observasi *fraud* yang sudah ada, melainkan representasi titik-titik data baru yang dihasilkan dari perhitungan matematis jarak ketetanggaan (*k-nearest neighbors*).

Distribusi kelas yang seimbang setelah penerapan SMOTE bertujuan meningkatkan kinerja model SVM, khususnya dalam mengklasifikasi transaksi *fraud* yang sebelumnya kurang terwakili. Kondisi ini menjadi landasan penting bagi pengujian pada metode-metode selanjutnya, baik dengan maupun tanpa reduksi dimensi menggunakan PCA. Dengan data latih yang telah seimbang, model SVM

memiliki peluang yang lebih besar untuk mempelajari batas pemisah yang optimal antara kelas *fraud* dan *non-fraud*, sehingga diharapkan mampu meningkatkan nilai *recall* dan *F1-score* pada kelas *fraud*.

#### 3.3.1.6 *Principal Component Analysis* (PCA)

*Principal Component Analysis* (PCA) bertujuan untuk mereduksi dimensi data dengan tetap mempertahankan informasi utama. Alur sistem PCA disajikan pada gambar 3.4.




Gambar 3. 4 Alur Sistem PCA

PCA mentransformasikan data asli ke fitur baru yang dibentuk oleh kombinasi linier dari fitur-fitur awal. Langkah matematis PCA adalah sebagai berikut:

1. Menghitung *matrix kovarians* disajikan pada persamaan 3.3.

$$C = \frac{1}{n-1} X^T X \tag{3.3}$$

2. Menentukan *Eigenvalue*$(\lambda)$ dan *Eigenvector* $(v)$ dari *matrix kovarians* dengan persamaan 3.4.

$$Cv = \lambda v \tag{3.4}$$

3. Mengurutkan *eigenvector* berdasarkan *eigenvalue* terbesar
4. Memilih komponen utama dengan akumulasi variansi ≥ 95%
5. Melakukan transformasi data melalui persamaan 3.5.

$$XPC\ A = X\ .W \tag{3.5}$$

PCA terbukti efektif dalam mengurangi kompleksitas data berdimensi tinggi serta meningkatkan stabilitas model klasifikasi (Alshawi, 2024). Penggunaan PCA dengan mempertahankan proporsi variansi yang tinggi seperti 95% hingga 99% dapat membantu menjaga informasi penting dalam dataset sehingga performa model klasifikasi tidak mengalami penurunan. Sehingga penggunaan *threshold* yang lebih tinggi pada PCA dianggap lebih efektif dalam mempertahankan karakteristik data yang relevan untuk proses klasifikasi(Rakovski, n.d.).

Berdasarkan proses transformasi fitur yang telah dilakukan, ringkasan nilai *Explained Variance Ratio* dan *Cumulative Variance* untuk setiap Komponen Utama (Principal Component) dapat dilihat pada Tabel 3.10.

Tabel 3.10. Contoh data hasil PCA

| *Principal Component* | *Explained Variance Ratio* | *Cumulative Variance* |
|---|---|---|
| PC1 | 0.216570 | 0.216570 |
| PC2 | 0.182851 | 0.399421 |
| PC3 | 0.113569 | 0.512990 |
| PC4 | 0.080789 | 0.593779 |
| PC5 | 0.078477 | 0.672256 |
| PC6 | 0.067565 | 0.739821 |
| PC7 | 0.060340 | 0.800162 |
| PC8 | 0.055082 | 0.855243 |
| PC9 | 0.048396 | 0.903639 |
| PC10 | 0.041547 | 0.945187 |
| PC11 | 0.036142 | 0.981329 |
| PC12 | 0.018495 | 0.999823 |

Berdasarkan hasil PCA yang disajikan pada tabel 3.5, komponen pertama (PC1) mampu menjelaskan variansi sebesar 21,65% dari keseluruhan data. Ketika digabungkan dengan komponen kedua (PC2), total variansi yang dapat dijelaskan meningkat menjadi 39,94%. Penambahan komponen berikutnya secara bertahap meningkatkan nilai variansi kumulatif, sehingga pada PC9 variansi kumulatif telah

mencapai 90,36%. Selanjutnya pada PC10 nilai tersebut meningkat menjadi 94,52%, dan pada PC12 mencapai 99,98% dari total variansi data.

Dalam penelitian ini, ambang batas *(threshold)* retensi informasi yang ditetapkan adalah sebesar 99%. Berdasarkan hasil pada tabel yang telah disajikan, penggunaan hingga 10 komponen utama (PC1–PC12) menghasilkan variansi kumulatif sebesar 99,98%. Nilai tersebut menunjukkan bahwa informasi yang berhasil dipertahankan sudah sangat tinggi dan mendekati ambang batas yang ditetapkan. Transformasi ini selanjutnya diterapkan dalam proses klasifikasi model SVM untuk mengevaluasi pengaruh PCA terhadap performa klasifikasi pada metode pengujian yang telah ditentukan.

#### 3.3.1.7 Split Data

Salah satu tahapan penting dalam pengembangan model *machine learning* adalah proses pembagian dataset *data splitting*. Tahap ini dilakukan untuk memastikan bahwa model yang dibangun tidak hanya mampu mengenali pola pada data pelatihan, tetapi juga dapat memberikan prediksi yang baik terhadap data baru yang belum pernah dipelajari sebelumnya. Kemampuan tersebut dikenal sebagai kemampuan generalisasi model dan menjadi salah satu indikator penting dalam evaluasi kinerja model.

Sebelum model digunakan untuk melakukan klasifikasi, dataset perlu dipisahkan menjadi dua kelompok data yang memiliki fungsi berbeda. Sebagian data digunakan selama proses pembelajaran model untuk mengenali pola dan karakteristik yang terdapat dalam dataset, sedangkan sebagian lainnya disimpan sebagai data pembanding untuk menguji hasil yang diperoleh model. Pendekatan

ini memungkinkan proses evaluasi dilakukan pada data yang belum pernah dilihat sebelumnya sehingga kemampuan model dalam menghadapi data baru dapat diukur secara lebih realistis. Selain itu, pembagian data juga membantu mengidentifikasi apakah model terlalu bergantung pada data pelatihan (*overfitting*) atau justru belum mampu mempelajari pola yang tersedia dengan baik (*underfitting*). Dengan demikian, performa yang diperoleh dari tahap pengujian dapat digunakan sebagai indikator kualitas model dalam menghasilkan prediksi.

### 3.4 *Support vector machine* (SVM)

Implementasi SVM dalam penelitian ini dilakukan menggunakan bahasa pemrograman *python* dengan proses pelatihan dan pengujian dijalankan melalui *google colaboratory*. Untuk memberikan pemahaman yang lebih mendalam terhadap mekanisme kerja algoritma SVM, pada penelitian ini disajikan contoh perhitungan manual menggunakan data hasil reduksi dimensi PCA. Contoh ini bertujuan untuk menunjukkan dasar matematis SVM sehingga penelitian tidak hanya bergantung pada penggunaan pustaka pemrograman, tetapi juga memiliki landasan berpikir yang kuat secara teoritis. Dalam contoh ini digunakan *treshhold* sebesar 95% yang diambil dari hasil PCA dengan dua kelas klasifikasi, yakni transaksi *fraud* dan *non-fraud*. Kelas *fraud* direpresentasikan dengan label +1, sedangkan kelas *non-fraud* direpresentasikan dengan label −1. SVM kernel linear bertujuan untuk mencari *hyperplane* optimal yang dapat memisahkan kedua kelas dengan margin maksimum.

Berdasarkan data *training* yang digunakan, ditentukan beberapa titik yang berperan sebagai *support vector*. Selanjutnya dilakukan perhitungan vektor bobot

dan bias yang membentuk fungsi keputusan SVM. Fungsi keputusan tersebut kemudian digunakan untuk menentukan kelas suatu data baru berdasarkan tanda dari hasil perhitungan fungsi tersebut. Apabila nilai fungsi keputusan bernilai positif, maka data diklasifikasikan sebagai *fraud*, sedangkan apabila bernilai negatif maka diklasifikasikan sebagai *non-fraud*.

Tabel 3.11. Data hasil transformasi PCA

| Data | PC1 | PC2 | Kelas |
|---|---|---|---|
| D1 | -0.897840 | 0.166304 | 1 |
| D2 | -1.993149 | 0.213033 | 1 |
| D3 | 0.574764 | 12.705519 | 1 |
| D4 | 0.260571 | 0.035765 | 1 |
| D5 | 5.374092 | 0.199477 | 1 |
| D6 | -0.495650 | 0.107723 | 0 |
| D7 | -0.161321 | 0.087191 | 0 |
| D8 | -0.823855 | 0.037431 | 0 |
| D9 | -0.453904 | 0.145167 | 0 |
| D10 | -2.133362 | 0.202731 | 0 |

Contoh perhitungan manual SVM pada penelitian ini menggunakan data asli hasil transformasi PCA yang ditunjukkan pada tabel 3.11. Data yang digunakan terdiri dari sepuluh data transaksi yang mewakili kelas *fraud* dan *non-fraud*. Secara matematis *hyperplane* dinyatakan pada persamaan 3.6.

$$w.x + b = 0 \tag{3.6}$$

Keterangan:

- $W = (w1, w2)$ adalah vector bobot yang menentukan arah *hyperplane*
- $X = (x1, x2)$ adalah vector fitur PC1 dan PC2
- $b$ adalah bias yang menggeser posisi *hyperplane*

Fungsi keputusan SVM linear untuk menentukan kelas data dinyatakan dengan persamaan 3.7.

$$f(x) = w.x + b \tag{3.7}$$

Dengan aturan klasifikasi:

- Jika $f(x) \geq 0$ = Data diklasifikasikan sebagai *fraud.*
- Jika $f(x) \leq 0$ = Data diklasifikasikan sebagai *non-fraud.*

Margin dan *hyperplane* optimal dinyatakan dalam persamaan 3.8.

$$MARGIN = \frac{2}{||W||} \tag{3.8}$$

Margin adalah jarak terdekat antara *hyperplane* dengan titik data dari masing-masing kelas. SVM berusaha memaksimalkan margin ini agar pemisahan kelas menjadi lebih *robust* terhadap *noise*. Secara matematis, besar margin berbanding terbalik dengan norma vektor bobot oleh karena itu, memaksimalkan margin setara dengan meminimalkan norma vektor bobot. Selanjutnya fungsi optimasi SVM di formulasikan pada persamaan 3.9 yang bertujuan untuk menjaga *hyperplane* tetap sejauh mungkin dari data terdekat dan Menjaga *hyperplane* tetap sejauh mungkin dari data terdekat. Faktor $\frac{1}{2}$ pada persamaan ini ditambahkan untuk mempermudah proses deferensiasi pada optimasi matematis.

$$\frac{min}{w,b} \frac{1}{2} ||W||^2 \tag{3.9}$$

Tahap selanjutnya dilakukan agar setiap data diklasifikasikan dengan benar dan berada di luar margin, maka dilakukan kendala yang diformulasikan pada persamaan 3.10.

$$yi(w \,.\, xi) + b \;\geq 1 \tag{3.10}$$

Keterangan:

- $yi \in \{+1, -1\}$ sebagai label kelas
- $xi$ adalah data ke-i
- Untuk kelas *fraud* $yi = +1$ dinyatakan dengan $w.xi = b \geq 1$
- Untuk kelas *non- fraud* $yi = +1$ dinyatakan dengan $w.xi = b \leq -1$

Support vector diformulasikan dalam persamaan 3.11.

$$yi\ (w\ .xi + b) = 1 \tag{3.11}$$

Langkah awal dalam perhitungan manual penelitian ini adalah mengidentifikasi *support vector* terlebih dahulu. *Suppor vector* adalah titik data yang berada paling dekat dengan *hyperplane* dan biasanya memiliki nilai PC1 dan PC2 yang saling berdekatan antar kelas. Mengacu pada hasil PCA pada tabel 3.11, titik yang berada paling dekat antar kelas adalah *fraud* (D1) dengan nilai (−0.897840,0.166304) dan *Non-fraud* (D9) dengan nilai (−0.453904,0.145167). koefiensi lagrange pada pehitungan kali ini diasumsikan dengan $a1 = a2 = 0{,}5$. Selanjutnya akan dilakukan perhitungan vektor bobot (w) dengan menggunakan persamaan 3.12.

$$w = \sum aiyixi \tag{3.12}$$

Dengan subtitusi nila i:

W= 0.5 (+1) (−0.897840,0.166304) + 0.5 (−1) (−0.453904,0.145167)

= (−0.448920,0.083152) + (0.226952,−0.072584)

W = (−0.448920,0.083152)+(0.226952,−0.072584)

Selanjutnya yakni perhitungan nilai bias dengan rumus $y(w.x+b)=1$ sehingga didapatkan nilai:

(+1)[(−0.221968)(−0.897840)+(0.010568)(0.166304)+b]=1

0.1994+0.0018+b=1

b=0.7988

Maka fungsi keputusan SVM adalah:

$$f(x) = 0.221968x_1 + 0.010568x_2 + 0.7988$$

Contoh klasifikasi data asli (D2):

X = (−1.993149,0.213033)

f(x) = (−0.221968)(−1.993149)+(0.010568)(0.213033)+0.7988

=0.442+0.002+0.7988=1.2428

f(x)>0 : *Fraud*

Tahapan perhitungan meliputi transformasi label kelas, identifikasi *support vector*, perhitungan vektor bobot dan bias, serta pembentukan fungsi keputusan. Hasil perhitungan menunjukkan bahwa fungsi keputusan SVM mampu memisahkan sebagian besar data sesuai dengan label aslinya. Dengan demikian, perhitungan manual ini membuktikan bahwa implementasi SVM pada penelitian ini memiliki dasar matematis yang jelas dan menggunakan data yang sama dengan proses pelatihan model.

### 3.5. Evaluasi Peforma

Setelah Evaluasi performa dilakukan untuk menilai kemampuan model dalam mengklasifikasikan data secara tepat. Pada penelitian ini, evaluasi menggunakan *confusion matrix* yang membandingkan hasil prediksi dengan label aktual pada data uji. Melalui confusion matrix, dapat diperoleh nilai *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) yang menjadi dasar perhitungan metrik evaluasi.

Berdasarkan nilai tersebut, dihitung *accuracy*, *precision*, *recall*, dan *F1-score* untuk mengukur performa model. Penggunaan beberapa metrik evaluasi diperlukan agar kemampuan model dalam mendeteksi transaksi *fraud* dan *non-fraud* dapat dianalisis secara lebih menyeluruh. Rumus perhitungan masing-masing metrik disajikan pada Persamaan 3.13.

$$Accuracy = \frac{\mathrm{TP+TN}}{TP+FP+TN+FN} \times 100\% \quad (3.13)$$

*accuracy* merupakan hasil dari total data yang terklasifikasi dengan benar dirumuskan dalam persamaan 3.14.

$$Precision = \frac{\mathrm{TP}}{TP+FP} \times 100\% \quad (3.14)$$

Berdasarkan Persamaan 3.15, *precision* diimplementasikan untuk mengevaluasi ketepatan klasifikasi positif yang dihasilkan model. Nilai ini menunjukkan proporsi data yang diprediksi sebagai kelas positif dan benar-benar merupakan anggota kelas tersebut. Semakin tinggi nilai *precision*, maka model akan semakin baik dalam melakukan klasifikasi.

$$Recall = \frac{\text{TP}}{TP+FN} \times 100\% \quad (3.15)$$

*recall* adalah nilai perbandingan ketepatan prediksi benar dengan jumlah seluruh prediksi yang seharusnya benar. disajikan pada persamaan 3.16.

$$f1\ score = 2\frac{(recall*precision)}{(recall+precision)} \times 100\% \quad (3.16)$$

*f1 score* adalah perhitungan yang digunakan untuk menggabungkan nilai *Precision*dan *recall*. Contoh penggunaan *confusion matrix* dalam peneltian ini dijelaskan sebagai pada tabel 3.7.

Tabel 3.12. Confusion matrix

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TP : 6 | FN : 2 |
| *Non-fraud* | FP : 1 | TN : 11 |

Perhitungan Manual Metrik evaluasi :

$$Accuracy = \frac{6+11}{6+11+1+2} = \frac{17}{20} = 0.85$$

$$Precision = \frac{6}{6+1} = 0.857$$

$$Recall = \frac{6}{6+2} = 0.75$$

$$F1 = 2\frac{0.857\ .\ 0.75}{0.857+0.75} = 0.799$$

Berdasarkan hasil pengujian, diperoleh nilai TP, FP, TN, dan FN yang membentuk *confusion matrix*. Komponen-komponen tersebut kemudian digunakan untuk menghitung nilai *accuracy, , recall,* dan *F1-score* sebagai indikator performa model. *Accuracy* menunjukkan proporsi prediksi yang berhasil diklasifikasikan dengan benar terhadap seluruh data uji. Pada kasus deteksi *fraud*, metrik *precision*

dan *recall* menjadi perhatian utama karena mampu menggambarkan efektivitas model dalam mengenali transaksi yang terindikasi *fraud* serta meminimalkan kesalahan klasifikasi pada kelas minoritas.

### 3.6 Metode Pengujian

Metode pengujian bertujuan untuk mengevaluasi sistem yang dikembangkan. Dalam proses ini, dataset akan dibagi ke dalam empat metode berbeda untuk memisahkan data pelatihan dan data pengujian. Tabel 3.9 merupakan metode pengujian yang dilakukan oleh peneliti.

Tabel 3. 13. Metode Pegujian

| Metode | Rasio Perbandingan | Fokus Evaluasi |
|---|---|---|
| Pengujian dengan Metode SVM PCA SMOTE | 60:40 | Performa klasifikasi *fraud* dengan fitur optimal & data seimbang. |
| | 70:30 | |
| | 80:20 | |
| | 90:10 | |
| Pengujian dengan Metode SVM SMOTE tanpa PCA | 60:40 | Pengaruh data seimbang tanpa pengurangan dimensi. |
| | 70:30 | |
| | 80:20 | |
| | 90:10 | |
| Pengujian dengan Metode SVM PCA tanpa SMOTE | 60:40 | Pengaruh pengurangan dimensi pada data yang masih *imbalanced*. |
| | 70:30 | |
| | 80:20 | |
| | 90:10 | |
| Pengujian dengan Metode SVM tanpa PCA SMOTE | 60:40 | Performa dasar SVM pada data asli. |
| | 70:30 | |
| | 80:20 | |
| | 90:10 | |

Pengujian sistem dalam penelitian ini dirancang untuk mengevaluasi kinerja algoritma SVM melalui empat metode yang berbeda guna memahami pengaruh integrasi teknik reduksi dimensi dan penyeimbangan data. Metode pertama menggabungkan penggunaan SVM, PCA, dan SMOTE secara simultan. Dalam metode ini, model diuji untuk melihat efektivitas kerja sama antara PCA dalam

menyederhanakan fitur dan SMOTE dalam menangani ketidakseimbangan kelas, yang diharapkan mampu menghasilkan performa deteksi transaksi *fraud* yang paling optimal dan efisien secara komputasi.

Pada metode kedua, pengujian difokuskan pada kombinasi metode SVM dan SMOTE tanpa melibatkan PCA. Langkah ini bertujuan untuk mengisolasi dan mengamati sejauh mana teknik penyeimbangan data sintetis dapat meningkatkan kemampuan model dalam mengenali kelas minoritas ketika seluruh fitur asli tetap dipertahankan tanpa reduksi. Selanjutnya, metode ketiga dilakukan dengan menerapkan metode SVM dan PCA tanpa bantuan SMOTE. Fokus utama dari metode ini adalah untuk menganalisis apakah transformasi fitur ke dalam ruang dimensi yang lebih rendah dapat membantu SVM dalam memisahkan pola transaksi pada kondisi data yang masih timpang atau *imbalanced*. Sebagai parameter pembanding terakhir, metode keempat menerapkan metode SVM secara mandiri tanpa intervensi dari PCA maupun SMOTE. Metode ini berfungsi sebagai model dasar atau *baseline* untuk mengukur performa murni algoritma SVM pada dataset asli.

Seluruh metode tersebut diuji secara konsisten menggunakan empat variasi rasio pembagian data, yakni 90:10 dengan menggunakan total 5000 data, yang terdiri dari 4500 data latih dengan distribusi data 4423 transaksi *non fraud* dan 77 transaksi *fraud*. Data uji menggunakan 500 data, yang terdiri dari 492 transaksi *non fraud* dan 8 *fraud*.

Perbandingan 80:20 menggunakan total 5000 data, yang terdiri dari 4000 data latih dengan distribusi data 3932 transaksi *non fraud* dan 68 transaksi *fraud*.

Data uji menggunakan 1000 data, yang terdiri dari 983 transaksi *non-fraud* dan 17 *fraud*.

Perbandingan 70:30 menggunakan total 5000 data, yang terdiri dari 300 data latih dengan distribusi data 3440 transaksi *non fraud*, dan 60 transaksi *fraud*. Data uji menggunakan 1500 data, yang terdiri dari 1475 transaksi *non fraud* dan 25 *fraud*. Perbandingan 60:40 menggunakan total 5000 data, yang terdiri dari 3000 data latih dengan distribusi data 2949 transaksi *non-fraud*, dan 51 transaksi *fraud*. Data uji menggunakan 2000 data, yang terdiri dari 1966 transaksi *non-fraud* dan 34 *fraud*.

Dengan metode tersebut penelitian dapat memetakan stabilitas serta tingkat akurasi model dalam berbagai kondisi distribusi data, sehingga memberikan gambaran yang komprehensif mengenai kontribusi masing-masing metode terhadap hasil akhir klasifikasi penipuan.

# BAB IV

# HASIL DAN PEMBAHASAN

Tahap pengujian dilakukan untuk mengevaluasi performa model SVM dalam mendeteksi transaksi *fraud* serta menganalisis pengaruh penerapan SMOTE dan PCA terhadap hasil klasifikasi. Seluruh pengujian dijalankan menggunakan bahasa pemrograman *Python* pada platform *Google Colab*.

Pengujian terdiri atas 16 skenario yang merupakan kombinasi dari empat metode dan empat rasio pembagian data latih dan data uji, yaitu 60:40, 70:30, 80:20, dan 90:10. Kinerja model dievaluasi menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score*, serta dianalisis melalui *confusion matrix* untuk melihat kemampuan model dalam membedakan transaksi *fraud* dan *non-fraud*.

## 4.1 Metode Pengujian

Untuk mendapatkan analisis yang komprehensif pengujian disusun ke dalam beberapa metode eksperimen. Rancangan metode ini bertujuan untuk membandingkan dampak dari kombinasi teknik *Preprocessing* (SMOTE dan PCA) terhadap performa SVM. Terdapat empat metode utama yang diujikan dalam penelitian ini, yaitu:

1. Pengujian Metode SVM *baseline*: Model SVM dilatih menggunakan data mentah *(raw data)* yang hanya melalui proses standarisasi *(Standard Scaler)*, tanpa PCA maupun SMOTE. Metode ini berfungsi sebagai *baseline* atau pembanding dasar.

2. Pengujian Metode SVM dan SMOTE: Model SVM dilatih menggunakan data yang telah disetarakan dengan SMOTE, namun tanpa melalui proses reduksi dimensi PCA. Metode ini bertujuan melihat dampak SMOTE secara isolasi.
3. Pengujian Metode SVM dan PCA: Model SVM dilatih menggunakan data hasil reduksi PCA, namun dibiarkan dalam kondisi tidak seimbang (tanpa SMOTE). Metode ini bertujuan melihat dampak PCA pada *data imbalanced.*
4. Pengujian Metode SVM dengan SMOTE dan PCA: Model SVM dilatih menggunakan data yang telah direduksi dimensinya dengan PCA dan disetarakan jumlah kelasnya menggunakan SMOTE.

Setiap metode di atas diujikan kembali dengan menggunakan empat variasi rasio pembagian data latih dan data uji untuk memastikan konsistensi model.

### 4.2 Hasil Pengujian

Subbab ini menyajikan hasil implementasi dan pengujian model klasifikasi kecurangan transaksi kartu kredit yang telah dirancang pada tahap sebelumnya. Pengujian dilakukan untuk mengetahui kemampuan model dalam membedakan transaksi *fraud* dan *non-fraud* berdasarkan dataset yang digunakan. Hasil yang diperoleh selanjutnya digunakan untuk menilai efektivitas metode yang diterapkan serta menganalisis pengaruh setiap tahapan pra-pemrosesan terhadap performa klasifikasi.

Proses pengujian dilakukan melalui beberapa skenario eksperimen dengan kombinasi metode yang berbeda. Setiap skenario dirancang untuk mengevaluasi dampak penerapan teknik tertentu, seperti penanganan ketidakseimbangan kelas

menggunakan SMOTE, reduksi dimensi menggunakan PCA, serta penerapan SVM sebagai metode klasifikasi. Kinerja model dievaluasi menggunakan metrik *accuracy, precision, recall,* dan *F1-score* sehingga kemampuan model dapat dianalisis secara lebih menyeluruh. Seluruh hasil pengujian kemudian dibandingkan dan dibahas untuk mengidentifikasi konfigurasi yang memberikan performa terbaik serta memahami pengaruh semua metode terhadap kemampuan model dalam mendeteksi transaksi kecurangan.

### 1.1.1. Pengujian Menggunakan Metode *Support vector machine Baseline Imbalance Data*

Pengujian ini bertujuan untuk mengevaluasi performa awal algoritma SVM dalam mendeteksi transaksi *fraud* tanpa penerapan teknik penyeimbangan data. Model dilatih menggunakan distribusi data asli yang didominasi oleh transaksi *non-fraud*, sementara seluruh fitur tetap melalui proses standarisasi untuk menyesuaikan karakteristik data dengan kebutuhan algoritma SVM. Hasil pengujian pada skenario ini digunakan sebagai acuan dalam membandingkan pengaruh penerapan SMOTE dan PCA terhadap performa klasifikasi.

#### 1.1.1.1.Pengujian Model A

Pengujian pada skenario ini menggunakan rasio pembagian data latih dan data uji sebesar 60:40. Hasil evaluasi yang ditampilkan pada Tabel 4.1 digunakan sebagai *baseline* untuk membandingkan performa model pada skenario berikutnya yang menerapkan teknik penanganan ketidakseimbangan data.

Tabel 4. 1. Hasil uji coba model SVM *baseline* rasio 60:40

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| Non *Fraud* | 99% | 90% | 95% | 90% |
| *Fraud* | 0.9% | 59% | 16% | |

Berdasarkan hasil pengujian, SVM mampu mengklasifikasikan transaksi *non-fraud* dengan sangat baik. Sebagian besar transaksi normal berhasil dikenali secara tepat, yang tercermin dari nilai *precision*, *recall*, dan *F1-score* yang tinggi. Namun, performa pada kelas *fraud* masih belum memuaskan. Meskipun model mampu mendeteksi lebih dari separuh transaksi *fraud* yang terdapat pada data uji, sebagian besar prediksi *fraud* yang dihasilkan tidak sesuai dengan kondisi sebenarnya. Hal ini menunjukkan bahwa model masih mengalami kesulitan dalam membedakan transaksi *fraud* dan *non-fraud* ketika distribusi data tidak seimbang.

Tabel 4. 2. Confusion matriks SVM *baseline* 60:40

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 1775 | FP :191 |
| *Non-fraud* | FN : 14 | TP : 20 |

Hasil *confusion matrix* pada Tabel 4.2 menunjukkan bahwa model berhasil mengidentifikasi 1.775 transaksi *non-fraud* dan 20 transaksi *fraud* secara benar. Di sisi lain, masih terdapat 191 transaksi *non-fraud* yang salah terdeteksi sebagai *fraud* serta 14 transaksi *fraud* yang tidak berhasil dikenali. Temuan ini mengindikasikan bahwa model lebih efektif dalam mempelajari pola pada kelas mayoritas dibandingkan kelas minoritas. Akibatnya, kemampuan deteksi *fraud* belum optimal dan masih dipengaruhi oleh kondisi class *imbalance* pada dataset.

#### 1.1.1.2.Pengujian Model B

Pengujian Pengujian pada skenario ini menggunakan rasio pembagian data latih dan data uji sebesar 70:40. Hasil evaluasi yang ditampilkan pada Tabel 4.3.

Tabel 4. 3. Hasil pengujian SVM *baseline* rasio 70:30

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 90% | 94% | 90% |
| *Fraud* | 10% | 64% | 17% | |

Berdasarkan hasil evaluasi pada Tabel 4.3, model menghasilkan *accuracy* sebesar 90%. Performa klasifikasi pada kelas *non-fraud* tergolong sangat baik, yang terlihat dari tingginya nilai *precision*, *recall*, dan *F1-score*. Hasil tersebut menunjukkan bahwa sebagian besar transaksi normal dapat dikenali dengan benar oleh model. Kondisi ini tidak terlepas dari dominasi jumlah data *non-fraud* pada proses pelatihan, sehingga karakteristik kelas tersebut lebih mudah dipelajari oleh model.

Berbeda dengan kelas *non-fraud*, performa pada kelas *fraud* masih belum optimal. Meskipun sebagian besar transaksi *fraud* berhasil teridentifikasi, tingkat ketepatan prediksi pada kelas ini masih rendah. Dengan kata lain, banyak transaksi yang diprediksi sebagai *fraud* ternyata merupakan transaksi normal. Akibatnya, keseimbangan antara *precision* dan *recall* belum tercapai sehingga nilai *F1-score* pada kelas *fraud* masih relatif rendah.

Tabel 4. 4. Confusion matrixSVM *baseline* rasio 70:30

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 1325 | FP :150 |
| *Non-fraud* | FN : 9 | TP : 16 |

Hasil *confusion matrix* pada Tabel 4.4 memperlihatkan bahwa model berhasil mengklasifikasikan 1.325 transaksi *non-fraud* dan 16 transaksi *fraud*

secara benar. Namun, masih terdapat 150 transaksi normal yang salah terdeteksi sebagai *fraud* serta 9 transaksi *fraud* yang tidak berhasil dikenali. Temuan ini mengindikasikan bahwa model sudah cukup baik dalam mengenali pola pada kelas mayoritas, tetapi masih mengalami kesulitan dalam membedakan transaksi *fraud* dan *non-fraud* secara konsisten. Keberadaan *False Positive* dan *False Negative* yang masih cukup tinggi menunjukkan bahwa pengaruh ketidakseimbangan kelas belum sepenuhnya dapat diatasi pada skenario pengujian ini.

#### 1.1.1.3.Pengujian Model C

Pengujian ini dilakukan dengan rasio pembagian data latih dan data uji 80:20 dengan nilai peforma dari sistem yang meliputi *accuracy, pressision, recall, dan F1-score* seperti yang ditunjukkan pada tabel 4.5.

Tabel 4. 5. Hasil Pengujian SVM *baseline* rasio 80:20

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 89% | 94% | 88% |
| *Fraud* | 0.8% | 64% | 17% | |

Pada rasio pembagian data 80:20, model menghasilkan *accuracy* sebesar 88%. Performa klasifikasi pada kelas *non-fraud* tetap menunjukkan hasil yang sangat baik dengan nilai *precision*, *recall*, dan *F1-score* yang tinggi. Hasil ini mengindikasikan bahwa model mampu mengenali pola transaksi normal secara konsisten, sehingga sebagian besar data pada kelas mayoritas berhasil diklasifikasikan dengan benar. Sebaliknya, performa pada kelas *fraud* masih belum mengalami peningkatan yang signifikan. Meskipun model mampu mendeteksi lebih dari separuh transaksi *fraud* yang terdapat pada data uji, ketepatan prediksi pada kelas ini masih rendah. Banyak transaksi normal yang teridentifikasi sebagai

*fraud* sehingga jumlah *False Positive* tetap cukup besar. Kondisi tersebut menyebabkan nilai *F1-score* pada kelas *fraud* masih rendah dan menunjukkan bahwa kemampuan model dalam membedakan transaksi *fraud* dan *non-fraud* belum optimal. Hasil ini mengindikasikan bahwa ketidakseimbangan distribusi kelas masih memberikan pengaruh terhadap proses klasifikasi, meskipun jumlah data latih telah ditingkatkan dibandingkan skenario sebelumnya.

Table 4. 6. Confusion matrixSVM *baseline* 80:20

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 871 | FP :150 |
| *Non-fraud* | FN : 7 | TP : 10 |

Hasil *confusion matrix* yang ditunjukan pada tabel 4.6 diperoleh model berhasil mengklasifikasikan 871 transaksi *non-fraud* secara benar dan 10 transaksi *fraud* dengan akurat. Namun demikian, 150 transaksi *non-fraud* masih salah diklasifikasi sebagai *fraud* serta 7 transaksi *fraud* yang tidak berhasil diklasifikasi karena diprediksi sebagai *non-fraud*. Hasil tersebut menunjukkan bahwa model cukup optimal dalam mengenali kelas mayoritas. Akan tetapi, jumlah *False Positive* yang relatif besar menunjukkan bahwa model masih menghasilkan cukup banyak ketidaktepatan dalam bentuk prediksi *fraud* terhadap transaksi normal. Di sisi lain, meskipun jumlah *False Negative* tidak terlalu besar, keberadaannya tetap menunjukkan bahwa sistem belum sepenuhnya optimal dalam mendeteksi seluruh transaksi *fraud*.

#### 1.1.1.4.Pengujian Model D

Pengujian ini Pengujian pada skenario ini menggunakan rasio pembagian data latih dan data uji sebesar 90:10. Hasil evaluasi model berdasarkan nilai *accuracy*, *precision*, *recall*, dan *F1-score* disajikan pada Tabel 4.7.

Tabel 4.7. Hasil Pengujian SVM rasio 90:10

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 88% | 93% | 88% |
| *Fraud* | 0.8% | 62% | 18% | |

Berdasarkan hasil pengujian, model memperoleh *accuracy* sebesar 88%. Performa pada kelas *non-fraud* tetap menunjukkan hasil yang baik dengan tingkat klasifikasi yang konsisten. Sebagian besar transaksi normal berhasil dikenali dengan benar, yang menunjukkan bahwa model mampu mempelajari pola pada kelas mayoritas secara efektif. Namun, kondisi tersebut tidak diikuti oleh peningkatan performa yang signifikan pada kelas *fraud*. Meskipun sebagian besar transaksi *fraud* masih dapat terdeteksi, ketepatan prediksi pada kelas ini tetap rendah sehingga nilai *F1-score* belum mengalami perbaikan yang berarti.

Tabel 4. 8. convusion matrix SVM rasio 90:10

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 434 | FP :58 |
| *Non-fraud* | FN : 3 | TP : 5 |

Hasil *confusion matrix* menunjukkan bahwa model berhasil mengklasifikasikan 434 transaksi *non-fraud* dan 5 transaksi *fraud* secara benar. Di sisi lain, masih terdapat 58 transaksi normal yang salah diprediksi sebagai *fraud* serta 3 transaksi *fraud* yang tidak berhasil dikenali. Temuan ini menunjukkan bahwa penambahan proporsi data latih hingga 90% belum mampu mengatasi keterbatasan model dalam menangani kelas minoritas. Dengan kata lain, peningkatan jumlah data latih tidak secara otomatis meningkatkan kemampuan deteksi *fraud* ketika distribusi data masih tidak seimbang. Kondisi tersebut mengindikasikan bahwa faktor ketidakseimbangan kelas memiliki pengaruh yang lebih besar terhadap performa model dibandingkan penambahan jumlah data

pelatihan. Oleh karena itu, diperlukan pendekatan tambahan seperti SMOTE untuk meningkatkan representasi kelas *fraud* dan membantu model mempelajari karakteristik transaksi kecurangan secara lebih baik.

### 1.1.2. Pengujian Metode *Support vector machine* dengan Penerapan *Synthetic Minority Over-sampling Technique* (SMOTE)

Pengujian Metode ini menerapkan SMOTE pada data latih untuk mengatasi ketidakseimbangan jumlah data antara kelas *fraud* dan *non-fraud*. Proses oversampling dilakukan setelah pembagian data dengan rasio 60:40, 70:30, 80:20, dan 90:10. Tujuan utama penerapan SMOTE adalah meningkatkan representasi kelas *fraud* sehingga model SVM dapat mempelajari karakteristik kedua kelas secara lebih seimbang.

Pengujian dilakukan untuk mengetahui pengaruh SMOTE terhadap kemampuan model dalam mendeteksi transaksi *fraud*. Hasil yang diperoleh kemudian dibandingkan dengan skenario *baseline* guna mengevaluasi efektivitas penanganan class *imbalance* yang diterapkan.

#### 1.1.2.1. Pengujian Model A

Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 60:40. Hasil evaluasi model disajikan pada Tabel 4.9.

Tabel 4. 17. Hasil Pengujian SVM + SMOTE rasio 60:40

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 98% | 99% | 97% |
| *Fraud* | 31% | 50% | 38% | |

Penerapan SMOTE menghasilkan *accuracy* sebesar 97% dengan performa yang tetap sangat baik pada kelas *non-fraud*. Pada kelas *fraud*, model memperoleh

*precision* sebesar 31%, *recall* sebesar 50%, dan *F1-score* sebesar 38%. Dibandingkan dengan skenario *baseline*, terjadi peningkatan yang cukup signifikan pada nilai *precision* dan *F1-score*. Hasil ini menunjukkan bahwa penambahan data sintetis pada kelas *fraud* membantu model mengenali pola transaksi kecurangan dengan lebih baik serta mengurangi kecenderungan model untuk hanya berfokus pada kelas mayoritas. Meskipun masih terdapat transaksi *fraud* yang belum terdeteksi, penerapan SMOTE terbukti memberikan peningkatan performa yang lebih seimbang dibandingkan model tanpa penanganan ketidakseimbangan data.

Tabel 4. 10. convusion matrix SVM + SMOTE rasio 60:40

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 1928 | FP :38 |
| *Non-fraud* | FN : 17 | TP : 17 |

Analisis *confusion matrix* pada Tabel 4.10 menunjukkan bahwa model memiliki akurasi yang sangat tinggi dalam mengenali kelas mayoritas, dengan 1.928 transaksi *non-fraud* berhasil diklasifikasikan secara tepat. Rendahnya jumlah *False Positive* (38 transaksi) mengindikasikan peningkatan ketepatan model dalam membedakan transaksi normal dibandingkan metode sebelumnya. Namun pada kelas minoritas, model hanya mampu mendeteksi 17 transaksi *fraud* secara benar, sementara 17 transaksi lainnya gagal terdeteksi (*False Negative*). Kesamaan jumlah antara *True Positive* dan *False Negative* ini menunjukkan bahwa meskipun terdapat peningkatan keseimbangan performa dibandingkan model tanpa penanganan data, kemampuan deteksi *fraud* sistem masih belum optimal sepenuhnya.

### 1.1.2.2. Pengujian Model B

Pengujian Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 70:30. Hasil evaluasi model disajikan pada Tabel 4.11.

Tabel 4. 11. Hasil Pengujian SVM + SMOTE rasio 70:30

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 98% | 99% | 97% |
| *Fraud* | 33% | 52% | 41% | |

Pada rasio pembagian data 70:30, model SVM yang dikombinasikan dengan SMOTE menghasilkan *accuracy* sebesar 97%. Performa pada kelas *non-fraud* tetap berada pada tingkat yang sangat baik, menunjukkan bahwa proses penyeimbangan data tidak mengurangi kemampuan model dalam mengenali transaksi normal. Di sisi lain, peningkatan performa pada kelas *fraud* mulai terlihat melalui kenaikan nilai *precision, recall*, dan *F1-score* dibandingkan skenario *baseline*. Hasil tersebut menunjukkan bahwa SMOTE membantu model mempelajari karakteristik transaksi *fraud* secara lebih efektif. Meskipun masih terdapat sejumlah transaksi normal yang salah teridentifikasi sebagai *fraud*, kemampuan model dalam mendeteksi transaksi kecurangan menjadi lebih baik dan lebih seimbang. Peningkatan nilai *F1-score* pada kelas *fraud* mengindikasikan bahwa model tidak hanya lebih sensitif terhadap transaksi *fraud*, tetapi juga lebih akurat dalam memberikan prediksi dibandingkan sebelum penerapan SMOTE. Temuan ini memperlihatkan bahwa penanganan class *imbalance* melalui SMOTE memberikan kontribusi positif terhadap performa klasifikasi, terutama pada kelas minoritas yang menjadi fokus utama penelitian.

Tabel 4.12. convusion matrix SVM+SMOTE rasio 70:30

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 1449 | FP :26 |
| *Non-fraud* | FN : 12 | TP : 13 |

Analisis *confusion matrix* pada Tabel 4.12 untuk rasio 70:30 menunjukkan bahwa model SVM dengan SMOTE berhasil mengklasifikasikan 1.449 transaksi *non-fraud* dan 13 transaksi *fraud* secara akurat. Rendahnya jumlah *False Positive* (26 transaksi) mengindikasikan kemampuan sistem yang sangat baik dalam mengenali kelas mayoritas tanpa memicu banyak alarm palsu pada transaksi normal. Meskipun jumlah *True Positive* (13) hampir setara dengan *False Negative* (12). penggunaan SMOTE pada kelas *fraud* terbukti menambah ketepatan model mengenali pola *fraud* dibandingkan model tanpa penanganan ketidakseimbangan data. Secara keseluruhan, kombinasi SVM dan SMOTE pada rasio ini menjadikan model lebih stabil dan proporsional dalam menangani data transaksi kartu kredit yang tidak seimbang.

#### 1.1.2.3. Pengujian Model C

Pengujian Pengujian Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 70:30. Hasil evaluasi model disajikan pada Tabel 4.13.

Tabel 4. 13. Hasil Pengujian SVM + SMOTE rasio 80:20

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 98% | 98% | 97% |
| *Fraud* | 26% | 41% | 32% | |

Pengujian model SVM dengan SMOTE pembagian data 80:20 mendapatkan akurasi global 97%, dengan performa kelas mayoritas tetap stabil pada nilai *Precision* 99% dan *recall* 98%. Hal ini menunjukkan efektivitas tinggi dalam mengidentifikasi transaksi normal dengan kesalahan klasifikasi yang sangat rendah. Pada kelas minoritas model mencapai *recall* 41%, *Precision* 26%, dan *F1-*

*score* 32%. Meskipun penerapan SMOTE membantu meningkatkan deteksi dibandingkan model tanpa penyeimbangan, terdapat penurunan performa dibandingkan rasio 70:30. Fenomena ini menegaskan bahwa penambahan proporsi data latih tidak secara otomatis meningkatkan kemampuan pengenalan pola *fraud*, sehingga keseimbangan distribusi data tetap menjadi faktor krusial dalam optimalisasi model deteksi kecurangan.

Tabel 4.14. convusion matrix SVM+SMOTE rasio 80:20

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 963 | FP :20 |
| *Non-fraud* | FN : 17 | TP : 10 |

Analisis *confusion matrix* pada Tabel 4.14 untuk rasio 80:20 menunjukkan bahwa model SVM dengan SMOTE berhasil mengklasifikasikan 963 transaksi *non-fraud* dan 10 transaksi *fraud* secara akurat. Tingginya jumlah *True Negative* serta rendahnya *False Positive* (20 transaksi) membuktikan efektivitas model dalam mengenali transaksi normal dengan tingkat alarm palsu yang rendah. Namun pada kelas minoritas, jumlah *False Negative* (17) yang melebihi *True Positive* (10) mengindikasikan bahwa model masih memiliki keterbatasan dalam mendeteksi seluruh transaksi *fraud*. Meskipun penerapan SMOTE telah meningkatkan sensitivitas terhadap kelas minoritas dibandingkan model tanpa penanganan data, hasil ini menunjukkan bahwa optimalisasi lebih lanjut masih diperlukan untuk memperkuat kemampuan deteksi pada kategori *fraud*.

### 1.1.2.4.Pengujian Model D

Pengujian Pengujian Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 70:30. Hasil evaluasi model disajikan pada Tabel 4.15.

Tabel 4. 15. Hasil Pengujian SVM + SMOTE rasio 90:10

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 99% | 99% | 98% |
| *Fraud* | 33% | 25% | 29% | |

Pada rasio pembagian data 90:10, model SVM yang dikombinasikan dengan SMOTE memperoleh *accuracy* sebesar 98%. Performa pada kelas *non-fraud* tetap sangat tinggi, menunjukkan bahwa model mampu mempertahankan kemampuan klasifikasi pada kelas mayoritas dengan tingkat kesalahan yang sangat rendah. Namun, peningkatan tersebut tidak diikuti oleh performa yang sebanding pada kelas *fraud*.

Hasil pengujian menunjukkan bahwa kemampuan model dalam mendeteksi transaksi *fraud* justru mengalami penurunan dibandingkan beberapa rasio sebelumnya. Sebagian besar transaksi *fraud* masih belum berhasil dikenali, sehingga nilai *F1-score* pada kelas minoritas tetap rendah. Kondisi ini mengindikasikan bahwa proporsi data latih yang lebih besar tidak selalu menghasilkan performa deteksi *fraud* yang lebih baik. Temuan tersebut menunjukkan bahwa keberhasilan klasifikasi tidak hanya dipengaruhi oleh jumlah data pelatihan, tetapi juga oleh kemampuan model dalam menangkap karakteristik transaksi *fraud* yang cenderung kompleks dan sulit dipisahkan. Dengan demikian, rasio 90:10 belum mampu memberikan keseimbangan performa yang optimal

antara kelas *fraud* dan *non-fraud* meskipun menghasilkan tingkat akurasi yang tinggi.

Tabel 4.16. convusion matrix SVM+SMOTE rasio 90:10

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 488 | FP :4 |
| *Non-fraud* | FN : 2 | TP : 6 |

Analisis *confusion matrix*pada Tabel 4.16 untuk rasio 90:10 menunjukkan bahwa model SVM dengan SMOTE berhasil mengklasifikasikan 488 transaksi *non-fraud* dan 6 transaksi *fraud* secara akurat. Dominasi *True Negative* serta rendahnya *False Positive* (4 transaksi) mencerminkan tingkat ketepatan yang sangat tinggi dalam mengenali transaksi normal pada rasio ini. Meskipun jumlah data uji *fraud* sangat terbatas, model tetap mampu mendeteksi sebagian transaksi, walaupun performa pada kelas minoritas menjadi sangat sensitif terhadap perubahan kecil dalam prediksi benar maupun salah.

Secara keseluruhan pengujian pada metode SVM dengan SMOTE menyimpulkan bahwa teknik oversampling memberikan peningkatan performa yang konsisten pada kelas *fraud* dibandingkan model *baseline*. Di seluruh rasio, model menunjukkan stabilitas tinggi pada kelas mayoritas dengan *Precision* dan *recall* mendekati 99%, menghasilkan akurasi global antara 97% hingga 98%. Penerapan SMOTE mengangkat nilai *Precision*, *recall*, dan *F1-score* kelas minoritas, terutama pada rasio 60:40 dan 70:30 yang menunjukkan keseimbangan deteksi terbaik dengan *recall* di kisaran 50% dan *F1-score* antara 38%–41%.

Namun pada rasio 80:20 dan 90:10, nilai *recall* cenderung menurun meskipun akurasi meningkat, membuktikan bahwa penambahan proporsi data latih tidak selalu berbanding lurus dengan kemampuan deteksi *fraud*. Sebagai simpulan

akhir, kombinasi SVM dan SMOTE merupakan pendekatan paling stabil dan optimal dalam penelitian ini karena berhasil mengurangi bias kelas mayoritas tanpa mengorbankan akurasi. Rasio 70:30 diidentifikasi sebagai konfigurasi paling seimbang karena mampu mempertahankan stabilitas akurasi sekaligus performa deteksi pada kelas minoritas secara proporsional.

### 1.1.3. Pengujian Model *Support vector machine* dengan Reduksi Dimensi Menggunakan PCA

Pada metode ini, *Principal Component Analysis* (PCA) diterapkan sebelum proses pelatihan model SVM untuk mengurangi jumlah fitur yang digunakan. Reduksi dimensi dilakukan dengan tujuan menyederhanakan representasi data, mengurangi redundansi antar fitur, serta meningkatkan efisiensi proses komputasi. Pengujian dilakukan pada empat rasio pembagian data, yaitu 60:40, 70:30, 80:20, dan 90:10, untuk mengetahui pengaruh PCA terhadap performa klasifikasi dibandingkan dengan model tanpa reduksi dimensi.

#### 1.1.3.1.Pengujian Model A

Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 60:40. Hasil evaluasi model ditunjukkan pada Tabel 4.17.

Tabel 4.17. Hasil Pengujian SVM + PCA rasio 60:40

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 87% | 93% | 87% |
| *Fraud* | 0.7% | 59% | 13% | |

Berdasarkan hasil pengujian, model masih mampu mengklasifikasikan transaksi *non-fraud* dengan baik. Namun, performa pada kelas *fraud* belum menunjukkan perbaikan yang berarti. Meskipun sebagian transaksi *fraud* berhasil

dikenali, ketepatan prediksi pada kelas tersebut masih sangat rendah sehingga nilai *F1-score* yang dihasilkan tetap kecil. Kondisi ini mengindikasikan bahwa proses reduksi dimensi menyebabkan sebagian informasi yang berperan dalam membedakan transaksi *fraud* dan *non-fraud* menjadi kurang optimal untuk dipelajari oleh model. Akibatnya, kemampuan deteksi terhadap kelas minoritas tidak mengalami peningkatan meskipun kompleksitas data telah berkurang.

*Tabel 4.18. convusion matrix SVM+PCA rasio 60:40*

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Non-fraud* | TN : 1711 | FP :255 |
| *fraud* | FN : 14 | TP : 20 |

Hasil *confusion matrix* pada Tabel 4.18 menunjukkan bahwa model berhasil mengklasifikasikan 1.711 transaksi *non-fraud* secara benar, sementara 255 transaksi normal masih salah terdeteksi sebagai *fraud*. Temuan ini menunjukkan bahwa PCA mampu mempertahankan kemampuan model dalam mengenali kelas mayoritas, tetapi belum memberikan kontribusi yang signifikan terhadap peningkatan deteksi transaksi *fraud*. Dengan kata lain, penyederhanaan dimensi fitur memang dapat meningkatkan efisiensi data, namun belum tentu menghasilkan performa klasifikasi yang lebih baik pada kasus *fraud* detection yang memiliki distribusi kelas tidak seimbang.

### 1.1.3.2.Pengujian Model B

Pengujian Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 70:30. Hasil evaluasi model ditunjukkan pada Tabel 4.19.

Tabel 4.19. Hasil Pengujian SVM + PCA rasio 70:30

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 85% | 91% | 84% |
| *Fraud* | 0.06% | 60% | 11% | |

Pada rasio pembagian data 70:30, model SVM dengan reduksi dimensi PCA menghasilkan *accuracy* sebesar 84%. Performa klasifikasi pada kelas *non-fraud* masih tergolong baik, menunjukkan bahwa sebagian besar transaksi normal dapat dikenali dengan benar. Namun, hasil yang berbeda terlihat pada kelas *fraud*, di mana kemampuan model dalam memberikan prediksi yang tepat masih sangat terbatas. Meskipun sebagian transaksi *fraud* berhasil terdeteksi, banyak transaksi normal yang turut diklasifikasikan sebagai *fraud* sehingga performa pada kelas minoritas belum menunjukkan perbaikan yang signifikan.

Tabel 4.20. convusion matrix SVM+PCA rasio 70:30

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Non-fraud* | TN : 1247 | FP :228 |
| *fraud* | FN : 10 | TP : 15 |

Hasil *confusion matrix* pada Tabel 4.20 memperlihatkan bahwa model lebih efektif dalam mengenali transaksi *non-fraud* dibandingkan transaksi *fraud*. Jumlah prediksi benar pada kelas mayoritas jauh lebih tinggi, sementara kesalahan klasifikasi masih cukup sering terjadi pada kelas *fraud*. Kondisi ini menunjukkan bahwa reduksi dimensi menggunakan PCA belum mampu meningkatkan kemampuan model dalam membedakan karakteristik transaksi *fraud* secara optimal. Berkurangnya jumlah fitur kemungkinan menyebabkan sebagian informasi yang berperan dalam membedakan kedua kelas tidak lagi direpresentasikan secara kuat pada data hasil transformasi. Akibatnya, meskipun kompleksitas data menjadi lebih sederhana, kemampuan deteksi *fraud* belum mengalami peningkatan yang berarti.

### 1.1.3.3.Pengujian Model C

Pengujian Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 80:20. Hasil evaluasi model ditunjukkan pada Tabel 4.21.

Tabel 4.21. Hasil Pengujian SVM + PCA rasio 80:20

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 84% | 91% | 83% |
| *Fraud* | 0.5% | 53% | 10% | |

Pada rasio pembagian data 80:20, model SVM yang menerapkan PCA menghasilkan *accuracy* sebesar 83%. Performa pada kelas *non-fraud* masih tergolong baik, yang menunjukkan bahwa sebagian besar transaksi normal dapat diklasifikasikan dengan benar. Namun, kemampuan model dalam mengenali transaksi *fraud* masih belum memadai. Meskipun sebagian transaksi *fraud* berhasil terdeteksi, tingkat ketepatan prediksi pada kelas tersebut tetap rendah sehingga performa keseluruhan pada kelas minoritas belum mengalami peningkatan yang berarti.

Tabel 4.22. convusion matrix SVM+PCA rasio 80:20

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Non-fraud* | TN : 823 | FP :160 |
| *fraud* | FN : 8 | TP : 9 |

Hasil *confusion matrix* pada Tabel 4.22 memperlihatkan bahwa model lebih banyak menghasilkan prediksi yang benar pada kelas *non-fraud* dibandingkan kelas *fraud*. Kesalahan klasifikasi masih cukup sering terjadi, baik dalam bentuk transaksi normal yang diprediksi sebagai *fraud* maupun transaksi *fraud* yang gagal terdeteksi. Kondisi ini menunjukkan bahwa reduksi dimensi menggunakan PCA belum mampu mengatasi permasalahan ketidakseimbangan kelas pada dataset. Meskipun jumlah fitur telah dikurangi untuk menyederhanakan data, informasi yang diperlukan untuk membedakan karakteristik transaksi *fraud* dan *non-fraud*

belum dapat dipertahankan secara optimal. Akibatnya, performa deteksi *fraud* masih relatif rendah dan tidak menunjukkan peningkatan dibandingkan skenario sebelumnya.

#### 1.1.3.4. Pengujian Model D

Pengujian Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 90:10. Hasil evaluasi model ditunjukkan pada Tabel 4.23.

Tabel 4.23. Hasil Pengujian SVM + PCA rasio 90:10

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 83% | 90% | 83% |
| *Fraud* | 0.06% | 62% | 10% | |

Pada rasio pembagian data 90:10, model SVM dengan PCA menghasilkan *accuracy* sebesar 83%. Meskipun performa pada kelas *non-fraud* masih relatif baik, kemampuan model dalam mengidentifikasi transaksi *fraud* belum menunjukkan peningkatan yang berarti. Hasil ini mengindikasikan bahwa reduksi dimensi yang dilakukan melalui PCA belum mampu membantu model membedakan karakteristik transaksi *fraud* dan *non-fraud* secara lebih efektif. Tingginya nilai akurasi yang diperoleh juga lebih banyak dipengaruhi oleh dominasi kelas *non-fraud* dibandingkan kemampuan model dalam mendeteksi transaksi *fraud*.

Tabel 4. 24. convusion matrix SVM+PCA rasio 90:10

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Non-fraud* | TN : 408 | FP :84 |
| *fraud* | FN : 3 | TP : 5 |

Hasil *confusion matrix* pada Tabel 4.24 memperlihatkan bahwa sebagian besar transaksi normal berhasil diklasifikasikan dengan benar, tetapi kesalahan prediksi masih sering terjadi pada kelas *fraud*. Selain itu, jumlah transaksi normal yang salah terdeteksi sebagai *fraud* juga masih cukup tinggi. Kondisi tersebut

menunjukkan bahwa model belum mampu membangun batas pemisah yang jelas antara kedua kelas setelah proses reduksi dimensi dilakukan.

Secara teoritis, hasil ini dapat dijelaskan oleh mekanisme PCA yang mempertahankan komponen dengan variansi terbesar dalam data. Pada dataset yang tidak seimbang, variansi tersebut cenderung didominasi oleh karakteristik kelas *non-fraud* karena jumlah datanya jauh lebih besar. Akibatnya, informasi penting yang berkaitan dengan pola transaksi *fraud* berpotensi berkurang selama proses transformasi fitur. Temuan ini menunjukkan bahwa penggunaan PCA tanpa disertai teknik penyeimbangan data belum efektif untuk meningkatkan performa deteksi *fraud*. Dengan demikian, reduksi dimensi saja tidak cukup untuk mengatasi permasalahan class *imbalance* pada kasus deteksi kecurangan transaksi kartu kredit dalam penelitian ini.

### 1.1.4. Pengujian Model SVM dengan Penerapan SMOTE dan PCA

Pengujian ini mengombinasikan metode penanganan ketidakseimbangan data dan reduksi dimensi, yaitu *Synthetic Minority Over-sampling Technique* (SMOTE) dan *Principal Component Analysis* (PCA), dalam proses pelatihan model *Support vector machine* (SVM). Metode ini dirancang untuk mengatasi dua permasalahan utama dalam deteksi kecurangan transaksi kartu kredit, yaitu ketidakseimbangan data dan tingginya dimensi fitur.

Pengujian dilakukan dengan variasi rasio pembagian data latih dan data uji, yaitu 60:40, 70:30, 80:20, dan 90:10. Model SVM dilatih menggunakan data hasil kombinasi SMOTE dan PCA dengan parameter yang sama seperti pada metode sebelumnya. Hasil dari metode ini diharapkan dapat menunjukkan performa

optimal model dalam mendeteksi transaksi kecurangan dengan mempertimbangkan keseimbangan data dan efisiensi dimensi fitur.

#### 1.1.4.1. Pengujian Model A

Pengujian ini Pengujian Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 60:40. Hasil evaluasi model ditunjukkan pada Tabel 4.25.

Tabel 4.25. Hasil Pengujian SVM + PCA + SMOTE rasio 60:40

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 98% | 99% | 97% |
| *Fraud* | 31% | 50% | 38% | |

Pada rasio pembagian data 60:40, kombinasi SVM, SMOTE, dan PCA menghasilkan *accuracy* sebesar 97%. Performa model pada kelas *non-fraud* maupun *fraud* menunjukkan hasil yang sama dengan skenario SVM yang hanya menerapkan SMOTE. Kondisi ini mengindikasikan bahwa proses reduksi dimensi tidak memberikan perubahan yang berarti terhadap kemampuan klasifikasi model.

Hasil tersebut menunjukkan bahwa informasi penting yang digunakan untuk membedakan transaksi *fraud* dan *non-fraud* masih dapat dipertahankan setelah proses PCA. Dengan kata lain, reduksi dimensi yang dilakukan tidak menyebabkan hilangnya informasi yang berpengaruh terhadap proses pembelajaran model. Akibatnya, pola klasifikasi yang terbentuk tetap serupa dengan model SVM dengan SMOTE tanpa PCA.

Tabel 4.26. convusion matrix SVM+PCA+SMOTE rasio 60:40

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Non-fraud* | TN : 1928 | FP :38 |
| *Fraud* | FN : 17 | TP : 17 |

Berdasarkan *confusion matrix* pada Tabel 4.26, model mampu mengklasifikasikan sebagian besar transaksi *non-fraud* dengan benar, sementara kemampuan deteksi pada kelas *fraud* masih berada pada tingkat yang sama seperti skenario sebelumnya. Meskipun sejumlah transaksi *fraud* berhasil dikenali, masih terdapat transaksi *fraud* yang tidak terdeteksi serta beberapa transaksi normal yang salah diprediksi sebagai *fraud*. Temuan ini menunjukkan bahwa peningkatan performa yang diperoleh pada skenario ini lebih banyak dipengaruhi oleh penerapan SMOTE dibandingkan proses reduksi dimensi menggunakan PCA.

#### 1.1.4.2. Pengujian Model B

Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 70:30. Hasil evaluasi model ditunjukkan pada Tabel 4.27.

Tabel 4.27. Hasil Pengujian SVM + PCA + SMOTE rasio 70:30

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 98% | 99% | 97% |
| *Fraud* | 33% | 52% | 41% | |

Pada rasio pembagian data 70:30, kombinasi SVM, SMOTE, dan PCA menghasilkan *accuracy* sebesar 97% dengan performa yang stabil pada kedua kelas. Model mampu mempertahankan kemampuan klasifikasi yang sangat baik pada kelas *non-fraud*, sekaligus menunjukkan peningkatan kemampuan dalam mengenali transaksi *fraud* dibandingkan skenario tanpa penanganan ketidakseimbangan data. Hasil ini menunjukkan bahwa proses penyeimbangan data melalui SMOTE membantu model mempelajari karakteristik kelas *fraud* secara lebih efektif tanpa mengurangi akurasi pada kelas mayoritas.

Tabel 4.28. Hasil Pengujian SVM+PCA+SMOTE rasio 70:30

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Non-fraud* | TN : 1449 | FP :26 |
| *Fraud* | FN : 12 | TP : 13 |

Berdasarkan *confusion matrix* pada Tabel 4.28, sebagian besar transaksi *non-fraud* berhasil diklasifikasikan dengan benar dan jumlah kesalahan prediksi pada kelas tersebut relatif rendah. Pada kelas *fraud*, model mampu mendeteksi lebih dari separuh transaksi *fraud* yang terdapat pada data uji, meskipun masih terdapat beberapa transaksi yang tidak berhasil dikenali. Dibandingkan dengan skenario *baseline* maupun SVM + PCA, hasil ini menunjukkan peningkatan yang lebih seimbang antara kemampuan deteksi *fraud* dan ketepatan klasifikasi. Temuan tersebut mengindikasikan bahwa kombinasi SMOTE dan PCA mampu membantu model mempertahankan informasi penting pada data sekaligus mengurangi dampak ketidakseimbangan kelas, sehingga menghasilkan performa yang lebih optimal pada rasio pembagian data 70:30.

### 1.1.4.3. Pengujian Model C

Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 80:20. Hasil evaluasi model ditunjukkan pada Tabel 4.29.

Tabel 4.29. Hasil Pengujian SVM + PCA + SMOTE rasio 80:20

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 98% | 98% | 97% |
| *Fraud* | 26% | 41% | 32% | |

Pada rasio pembagian data 80:20, kombinasi SVM, SMOTE, dan PCA menghasilkan *accuracy* sebesar 97%. Model tetap menunjukkan performa yang sangat baik dalam mengklasifikasikan transaksi *non-fraud*, sehingga sebagian besar transaksi normal dapat dikenali dengan benar. Namun, kemampuan deteksi pada

kelas *fraud* mengalami penurunan dibandingkan skenario 70:30. Meskipun sejumlah transaksi *fraud* berhasil teridentifikasi, masih terdapat cukup banyak transaksi *fraud* yang tidak terdeteksi sehingga performa pada kelas minoritas belum mencapai hasil yang optimal.

Tabel 4.30. Hasil Pengujian SVM+PCA+SMOTE rasio 80:20

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 963 | FP :20 |
| *Non-fraud* | FN : 17 | TP : 10 |

Hasil *confusion matrix* pada Tabel 4.30 memperlihatkan bahwa jumlah kesalahan klasifikasi pada kelas *non-fraud* relatif rendah, yang menunjukkan bahwa model mampu mempertahankan tingkat akurasi yang tinggi pada kelas mayoritas. Di sisi lain, masih terdapat sejumlah transaksi *fraud* yang salah diprediksi sebagai *non-fraud*, sehingga kemampuan model dalam mendeteksi seluruh transaksi kecurangan belum maksimal. Temuan ini menunjukkan bahwa kombinasi SMOTE dan PCA tetap memberikan performa yang lebih baik dibandingkan skenario tanpa penanganan class *imbalance*, namun peningkatan proporsi data latih dari 70% menjadi 80% tidak menghasilkan perbaikan performa pada kelas *fraud*. Kondisi tersebut mengindikasikan bahwa jumlah data latih yang lebih besar tidak selalu berbanding lurus dengan peningkatan kemampuan deteksi *fraud*, terutama ketika karakteristik kelas minoritas masih sulit dipelajari secara optimal oleh model.

#### 1.1.4.4. Pengujian Model D

Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 70:30. Hasil evaluasi model ditunjukkan pada Tabel 4.31.

Tabel 4.31. Hasil Pengujian SVM + PCA + SMOTE rasio 90:10

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 99% | 99% | 99% | 98% |
| *Fraud* | 33% | 25% | 29% | |

Pada rasio pembagian data 90:10, kombinasi SVM, SMOTE, dan PCA menghasilkan *accuracy* sebesar 98%. Performa model pada kelas *non-fraud* tetap berada pada tingkat yang sangat baik, menunjukkan bahwa sebagian besar transaksi normal dapat dikenali dengan benar. Namun, peningkatan proporsi data latih tidak memberikan dampak yang signifikan terhadap kemampuan model dalam mendeteksi transaksi *fraud*. Hal ini terlihat dari masih rendahnya kemampuan model dalam mengidentifikasi seluruh transaksi *fraud* yang terdapat pada data uji.

Tabel 4.32. Hasil Pengujian SVM+PCA+SMOTE rasio 90:10

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 488 | FP :4 |
| *Non-fraud* | FN : 2 | TP : 6 |

Hasil *confusion matrix* pada Tabel 4.32 menunjukkan bahwa kesalahan klasifikasi pada kelas *non-fraud* relatif kecil, sehingga model mampu mempertahankan tingkat akurasi yang tinggi. Akan tetapi, masih terdapat transaksi *fraud* yang gagal dikenali, yang mengindikasikan bahwa model belum sepenuhnya mampu menangkap karakteristik kelas minoritas. Kondisi ini menunjukkan bahwa performa tinggi pada kelas mayoritas belum tentu diikuti oleh peningkatan kemampuan deteksi *fraud*.

Secara keseluruhan, kombinasi SMOTE dan PCA menghasilkan performa yang lebih baik dibandingkan skenario *baseline* maupun penggunaan PCA tanpa SMOTE. Penerapan SMOTE terbukti membantu meningkatkan representasi kelas *fraud* sehingga model dapat mempelajari pola transaksi kecurangan dengan lebih baik. Sementara itu, PCA berkontribusi dalam menyederhanakan dimensi data

tanpa menurunkan performa klasifikasi secara signifikan. Meskipun demikian, peningkatan jumlah data latih hingga rasio 90:10 tidak menghasilkan performa terbaik pada kelas *fraud*. Hasil pengujian menunjukkan bahwa rasio 70:30 memberikan keseimbangan yang lebih baik antara kemampuan deteksi *fraud* dan performa keseluruhan model, dengan nilai *F1-score* tertinggi pada kelas *fraud*. Temuan ini mengindikasikan bahwa keberhasilan model dalam mendeteksi transaksi kecurangan tidak hanya ditentukan oleh banyaknya data latih, tetapi juga oleh kualitas representasi kelas minoritas dan kemampuan model dalam mempelajari pola *fraud* yang kompleks.

Kombinasi SVM dengan PCA dan SMOTE efektif dalam mempertahankan akurasi tinggi dan stabilitas klasifikasi pada kelas mayoritas, tetapi belum mampu secara jelas meningkatkan kemampuan deteksi pada kelas minoritas namun tidak mengalami perubahan dibandingkan dengan model SVM dengan SMOTE tanpa PCA. Kondisi ini mengindikasikan bahwa penerapan PCA tidak memberikan pengaruh jelas terhadap performa klasifikasi. Hal tersebut disebabkan karena jumlah fitur pada dataset relatif tidak terlalu besar dan sebagian besar informasi penting telah dipertahankan dalam komponen utama PCA, sehingga proses reduksi dimensi tidak mengubah pola data yang digunakan oleh model SVM dalam proses pembelajaran.

### 1.1.5. Pengujian Metode *Support vector machine Baseline Balanced Data*

Untuk melengkapi analisis performa model, penelitian ini menambahkan metode pengujian menggunakan *Support vector machine baseline* pada dataset yang telah diseimbangkan. Pada metode ini, jumlah data antara kelas *fraud* dan *non-*

*fraud* disusun secara proporsional dengan komposisi masing-masing sebanyak 2500 data. Pengujian pada data seimbang dilakukan untuk mengetahui performa dasar algoritma SVM ketika jumlah data *fraud* dan *non-fraud* berada pada proporsi yang sama. Skenario ini digunakan sebagai pembanding terhadap pengujian pada data tidak seimbang sehingga pengaruh distribusi kelas terhadap hasil klasifikasi dapat dianalisis dengan lebih jelas..

#### 1.1.5.1. Pengujian Model A

Pengujian Pengujian ini Pengujian Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 60:40. Hasil evaluasi model ditunjukkan pada Tabel 4.33.

Tabel 4. 33. Hasil Pengujian SVM Balanced Data rasio 60:40

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 63% | 64% | 64% | 63% |
| *Fraud* | 63% | 62% | 63% | |

Pada rasio pembagian data 60:40, model menunjukkan performa yang relatif seimbang pada kedua kelas. Nilai *precision*, *recall*, dan *F1-score* yang tidak berbeda jauh antara kelas *fraud* dan *non-fraud* mengindikasikan bahwa model tidak cenderung memihak salah satu kelas tertentu. Berbeda dengan skenario *imbalance*d data, kemampuan klasifikasi pada kelas *fraud* meningkat karena model memperoleh representasi data yang setara selama proses pelatihan.

Nilai *accuracy* yang diperoleh sebesar 63% mengindikasikan bahwa secara keseluruhan model memiliki kemampuan klasifikasi pada tingkat moderat Keseimbangan nilai *precision* dan *recall* pada kedua kelas menunjukkan bahwa model tidak mengalami bias terhadap kelas tertentu, yang umumnya terjadi pada

dataset tidak seimbang. Dengan demikian model ini memberikan gambaran bahwa keseimbangan data mampu meningkatkan stabilitas prediksi model, meskipun belum secara signifikan meningkatkan akurasi keseluruhan.

Tabel 4. 10. convusion matrix SVM Balanced Data rasio 60:40

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 515 | FP :285 |
| *Non-fraud* | FN : 305 | TP : 495 |

Hasil *confusion matrix* pada Tabel 4.10 juga memperlihatkan distribusi kesalahan yang lebih merata antara kedua kelas. Meskipun masih terdapat sejumlah data yang salah diklasifikasikan, model mampu mengenali transaksi *fraud* dan *non-fraud* dengan tingkat keberhasilan yang relatif seimbang. Temuan ini menunjukkan bahwa distribusi kelas memiliki pengaruh yang signifikan terhadap performa SVM, di mana penyeimbangan data mampu mengurangi bias model terhadap kelas mayoritas dan menghasilkan kemampuan klasifikasi yang lebih proporsional.

### 1.1.5.2.Pengujian Model B

Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 60:40. Hasil evaluasi model ditunjukkan pada Tabel 4.33.

*Tabel 4. 11. Hasil Pengujian SVM Balanced Data rasio 70:40*

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 59% | 73% | 65% | 61% |
| *Fraud* | 65% | 49% | 56% | |

Pada rasio pembagian data 70:30, model SVM yang dilatih menggunakan data seimbang masih menunjukkan performa yang relatif konsisten pada kedua kelas. Tidak terlihat adanya dominasi prediksi pada kelas tertentu sebagaimana yang terjadi pada skenario data tidak seimbang. Hal ini menunjukkan bahwa

distribusi data yang seimbang membantu model mempelajari karakteristik transaksi *fraud* dan *non-fraud* secara lebih proporsional.

Tabel 4. 12. convusion matrix SVM Balanced Data rasio 70:30

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 547 | FP :203 |
| *Non-fraud* | FN : 380 | TP : 370 |

Hasil *confusion matrix* pada Tabel 4.12 memperlihatkan bahwa jumlah prediksi benar dan salah pada kedua kelas cenderung lebih seimbang. Temuan ini menunjukkan bahwa penyeimbangan data berhasil mengurangi bias model terhadap kelas mayoritas. Namun, keseimbangan distribusi kelas saja belum cukup untuk menghasilkan performa klasifikasi yang optimal, karena model masih mengalami kesulitan dalam membentuk batas pemisah yang mampu membedakan kedua kelas secara akurat.

### 1.1.5.3. Pengujian Model C

Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 80:20. Hasil evaluasi model ditunjukkan pada Tabel 4.33.

*Tabel 4. 13. Hasil Pengujian SVM Balanced Data rasio 80:20*

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 61% | 72% | 66% | 63% |
| *Fraud* | 66% | 54% | 59% | |

Pada rasio pembagian data 80:20, model SVM yang dilatih menggunakan data seimbang masih menunjukkan performa yang relatif konsisten pada kedua kelas. Tidak terdapat perbedaan performa yang terlalu jauh antara kelas *fraud* dan *non-fraud*, yang menandakan bahwa model tidak lagi terlalu dipengaruhi oleh dominasi salah satu kelas. Kondisi ini menunjukkan bahwa proses penyeimbangan

data berhasil membantu model mempelajari pola pada kedua kelas secara lebih proporsional. Meskipun demikian, kemampuan klasifikasi yang dihasilkan masih belum optimal. Hasil pengujian menunjukkan bahwa sejumlah transaksi *fraud* masih gagal terdeteksi, sementara sebagian transaksi *non-fraud* juga masih salah diklasifikasikan sebagai *fraud*. Akibatnya, performa keseluruhan model belum mengalami peningkatan yang signifikan meskipun distribusi data telah diseimbangkan. Hal ini mengindikasikan bahwa permasalahan klasifikasi tidak hanya dipengaruhi oleh ketidakseimbangan data, tetapi juga oleh tingkat kemiripan karakteristik antara kedua kelas.

Tabel 4. 14. convusion matrix SVM Balanced Data rasio 80:20

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 360 | FP :140 |
| *Non-fraud* | FN : 229 | TP : 271 |

Hasil *confusion matrix* pada Tabel 4.14 memperlihatkan bahwa kesalahan klasifikasi masih terjadi pada kedua kelas dengan proporsi yang relatif berimbang. Temuan ini menunjukkan bahwa penggunaan balanced data mampu mengurangi bias model terhadap kelas mayoritas, namun belum sepenuhnya meningkatkan kemampuan model dalam memisahkan transaksi *fraud* dan *non-fraud* secara akurat. Dengan demikian, meskipun stabilitas performa antar kelas meningkat, model masih memerlukan pendekatan tambahan untuk meningkatkan kemampuan deteksi *fraud*.

#### 1.1.5.4. Pengujian Model D

Pengujian pertama menggunakan rasio pembagian data latih dan data uji sebesar 90:10. Hasil evaluasi model ditunjukkan pada Tabel 4.33.

.

Tabel 4. 15. Hasil Pengujian SVM Balanced Data rasio 90:10

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 63% | 68% | 66% | 64% |
| *Fraud* | 65% | 60% | 63% | |

Pada rasio pembagian data 90:10, model SVM yang dilatih menggunakan data seimbang menunjukkan performa yang paling stabil dibandingkan skenario balanced data lainnya. Kemampuan model dalam mengenali transaksi *fraud* dan *non-fraud* berada pada tingkat yang relatif seimbang, sehingga tidak terlihat kecenderungan model untuk lebih memprioritaskan salah satu kelas. Kondisi ini menunjukkan bahwa penambahan proporsi data latih memberikan dampak positif terhadap proses pembelajaran model.

Hasil evaluasi juga menunjukkan adanya peningkatan kemampuan deteksi pada kelas *fraud* dibandingkan beberapa rasio sebelumnya. Meskipun masih terdapat kesalahan klasifikasi, model mulai mampu membedakan karakteristik transaksi *fraud* dan *non-fraud* dengan lebih baik. Hal ini mengindikasikan bahwa jumlah data pelatihan yang lebih besar membantu model memperoleh representasi pola yang lebih baik dari kedua kelas.

Tabel 4. 16. convusion matrix SVM Balanced Data rasio 90:10

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 170 | FP :80 |
| *Non-fraud* | FN : 99 | TP : 151 |

Berdasarkan *confusion matrix* pada Tabel 4.16, distribusi prediksi benar dan salah pada kedua kelas terlihat lebih seimbang dibandingkan skenario sebelumnya. Jumlah transaksi *fraud* yang berhasil dikenali meningkat, sementara kesalahan klasifikasi pada kedua kelas relatif terkendali. Temuan ini menunjukkan bahwa

kombinasi data seimbang dan proporsi data latih yang lebih besar mampu menghasilkan performa klasifikasi yang lebih konsisten. Meskipun akurasi keseluruhan masih berada di bawah skenario *imbalance*d data, hasil ini memberikan gambaran yang lebih realistis mengenai kemampuan model dalam mengklasifikasikan kedua kelas secara adil tanpa dipengaruhi dominasi kelas mayoritas.

### 1.1.6. Pengujian Metode *Support vector machine Baseline Balanced Data* Kernel *Radial Basis Function*.

Percobaan selanjutnya pada penelitian ini menambahkan metode pengujian menggunakan *Support vector machine baseline* dengan kernel *Radial Basis Function* (RBF) pada dataset yang telah diseimbangkan. Pada metode ini, jumlah data antara kelas *fraud* dan *non-fraud* disusun secara proporsional dengan komposisi masing-masing sebanyak 2500 data. Pengujian ini bertujuan untuk mengetahui pengaruh penggunaan kernel *non-linear* terhadap performa klasifikasi transaksi *fraud* dan *non-fraud*. Dengan demikian, hasil dari pengujian ini diharapkan dapat memperkuat analisis terkait pengaruh ketidakseimbangan data dan perbedaan kernel terhadap kinerja model klasifikasi.

#### 1.1.6.1. Pengujian Model A

Pengujian ini dilakukan dengan rasio pembagian data latih dan data uji 60:40 dengan nilai peforma dari sistem yang meliputi *accuracy, pressision, recall, dan F1-score* seperti yang ditunjukkan pada tabel 4.17.

Tabel 4. 17. Hasil Pengujian SVM Balanced  Data Kernel RBF rasio 60:40

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 66% | 66% | 66% | 66% |
| *Fraud* | 66% | 66% | 66% | |

Pengujian SVM kernel RBF pada *balanced data* rasio 60:40 menunjukkan bahwa model menghasilkan performa yang seimbang pada kedua kelas. Pada kelas *non-fraud* dan *fraud* diperoleh nilai *precision*, *recall*, dan *F1-score* yang sama, yaitu sebesar 66%. Selain itu, model menghasilkan *accuracy* sebesar 66%. Dibandingkan dengan skenario SVM *baseline balanced data* sebelumnya, penggunaan kernel RBF memberikan peningkatan performa yang kecil. Namun demikian, penggunaan kernel RBF menghasilkan performa yang lebih stabil dan seimbang antara kelas *fraud* dan *non-fraud* dibandingkan beberapa skenario sebelumnya.

Tabel 4. 18. convusion matrix SVM Balance Data Kernel RBF rasio 60:40

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN : 665 | FP :345 |
| *Non-fraud* | FN : 339 | TP : 661 |

Berdasarkan Tabel 4.18, *confusion matrix* menunjukkan bahwa model SVM kernel RBF pada *balanced data* rasio 60:40 mampu mengklasifikasikan 665 data *fraud* dengan benar, sedangkan 345 data *fraud* lainnya salah diprediksi sebagai *non-fraud*. Pada kelas *non-fraud*, model berhasil mengklasifikasikan 661 data dengan benar, namun masih terdapat 339 data yang salah diprediksi sebagai *fraud.* Hasil tersebut menunjukkan bahwa model menghasilkan distribusi prediksi yang relatif seimbang antara kelas *fraud* dan *non-fraud*. Dibandingkan dengan skenario SVM *baseline balanced data* sebelumnya, penggunaan kernel RBF memberikan peningkatan kecil pada kemampuan klasifikasi.

### 1.1.6.2. Pengujian Model B

Pengujian ini dilakukan dengan rasio pembagian data latih dan data uji 60:40 dengan nilai peforma dari sistem yang meliputi *accuracy*, *pressision, recall, dan F1-score* seperti yang ditunjukkan pada tabel 4.19.

Tabel 4. 19. Hasil Pengujian SVM Balanced  Data Kernel RBF rasio 70:30

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 67% | 67% | 67% | 67% |
| *Fraud* | 67% | 67% | 67% | |

Pengujian SVM kernel RBF pada *balanced data* rasio 70:30 menunjukkan bahwa model menghasilkan performa yang seimbang pada kedua kelas. Pada kelas *non-fraud* dan *fraud* diperoleh nilai *precision*, *recall*, dan *F1-score* yang sama, yaitu sebesar 67%. Selain itu, model menghasilkan *accuracy* sebesar 67%. Dibandingkan dengan skenario SVM *baseline balanced data* sebelumnya, penggunaan kernel RBF menunjukkan peningkatan performa yang kecil. Hasil tersebut mengindikasikan bahwa kernel *non-linear* mampu memberikan kemampuan klasifikasi yang sedikit lebih baik dibandingkan kernel linear. Penggunaan kernel RBF juga menghasilkan performa yang lebih stabil dan seimbang antara kelas *fraud* dan *non-fraud* dibandingkan beberapa skenario sebelumnya.

Tabel 4. 20. convusion matrix SVM Balance Data Kernel RBF rasio 70:30

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN :499 | FP :345 |
| *Non-fraud* | FN : 339 | TP : 503 |

Berdasarkan Tabel 4.20, *confusion matrix* menunjukkan bahwa model SVM kernel RBF pada *balanced data* rasio 70:30 mampu mengklasifikasikan 499 data *fraud* dengan benar, sedangkan 345 data *fraud* lainnya salah diprediksi sebagai *non-*

*fraud*. Pada kelas *non-fraud*, model berhasil mengklasifikasikan 503 data dengan benar, namun masih terdapat 339 data yang salah diprediksi sebagai *fraud*. Hasil tersebut menunjukkan bahwa model menghasilkan distribusi prediksi yang relatif seimbang antara kelas *fraud* dan *non-fraud*. Dibandingkan dengan skenario SVM *baseline balanced data* sebelumnya, penggunaan kernel RBF memberikan peningkatan kecil pada kemampuan klasifikasi. Namun demikian, jumlah *False Positive* dan *False Negative* yang masih cukup tinggi menunjukkan bahwa model belum mampu memisahkan kedua kelas cukup baik.

#### 1.1.6.3. Pengujian Model C

Pengujian ini dilakukan dengan rasio pembagian data latih dan data uji 60:40 dengan nilai peforma dari sistem yang meliputi *accuracy, pressision, recall, dan F1-score* seperti yang ditunjukkan pada tabel 4.21.

Tabel 4. 21. Hasil Pengujian SVM Balanced  Data Kernel RBF rasio 80:20

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 67% | 67% | 67% | 67% |
| *Fraud* | 67% | 67% | 67% | |

Pengujian SVM kernel RBF pada *balanced data* rasio 80:20 menunjukkan bahwa model menghasilkan performa yang seimbang pada kedua kelas. Pada kelas *non-fraud* dan *fraud* diperoleh nilai *precision*, *recall*, dan *F1-score* yang sama, yaitu sebesar 67%. Selain itu, model menghasilkan *accuracy* sebesar 67%. Dibandingkan dengan skenario SVM *baseline balanced data* sebelumnya, penggunaan kernel RBF menunjukkan peningkatan performa yang kecil. Hasil tersebut mengindikasikan bahwa kernel *non-linear* mampu memberikan kemampuan klasifikasi yang sedikit lebih baik dibandingkan kernel linear. Penggunaan kernel RBF juga menghasilkan performa yang lebih stabil dan

seimbang antara kelas *fraud* dan *non-fraud* dibandingkan beberapa skenario sebelumnya.

Tabel 4. 22. convusion matrix SVM Balance Data Kernel RBF rasio 80:20

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN :333 | FP :167 |
| *Non-fraud* | FN : 165 | TP : 335 |

Berdasarkan Tabel 4.22, *confusion matrix* menunjukkan bahwa model SVM kernel RBF pada *balanced data* rasio 80:20 mampu mengklasifikasikan 333 data *fraud* dengan benar, sedangkan 167 data *fraud* lainnya salah diprediksi sebagai *non-fraud*. Pada kelas *non-fraud*, model berhasil mengklasifikasikan 335 data dengan benar, namun masih terdapat 165 data yang salah diprediksi sebagai *fraud.* Hasil tersebut menunjukkan bahwa model menghasilkan distribusi prediksi yang relatif seimbang antara kelas *fraud* dan *non-fraud*. Dibandingkan dengan skenario SVM *baseline balanced data* sebelumnya, penggunaan kernel RBF memberikan peningkatan kecil pada kemampuan klasifikasi.

### 1.1.6.4. Pengujian Model D

Pengujian ini dilakukan dengan rasio pembagian data latih dan data uji 60:40 dengan nilai peforma dari sistem yang meliputi *accuracy, pressision, recall, dan F1-score* seperti yang ditunjukkan pada tabel 4.23.

Tabel 4. 23. Hasil Pengujian SVM Balanced  Data Kernel RBF rasio 90:10

| Kelas | *Precission* | *Recall* | *F1-score* | *Accuracy* |
|---|---|---|---|---|
| *Non-fraud* | 66% | 66% | 66% | 66% |
| *Fraud* | 66% | 66% | 66% | |

Pengujian SVM kernel RBF pada *balanced data* rasio 90:10 menunjukkan bahwa model menghasilkan performa yang seimbang pada kedua kelas. Pada kelas *non-fraud* dan *fraud* diperoleh nilai *precision*, *recall*, dan *F1-score* yang sama,

yaitu sebesar 66%. Selain itu, model menghasilkan *accuracy* sebesar 66%. Dibandingkan dengan skenario SVM *baseline balanced data* sebelumnya, penggunaan kernel RBF memberikan peningkatan performa yang kecil. Hasil tersebut menunjukkan bahwa kernel *non-linear* mampu menghasilkan kemampuan klasifikasi yang lebih stabil dibandingkan kernel *linear*. Penggunaan kernel RBF juga menghasilkan performa yang konsisten pada setiap rasio pembagian data dibandingkan beberapa skenario sebelumnya.

Tabel 4. 24. convusion matrix SVM Balance Data Kernel RBF rasio 80:20

| Data Aktual | Prediksi *fraud* | Prediksi *Non-fraud* |
|---|---|---|
| *Fraud* | TN :333 | FP :167 |
| *Non-fraud* | FN : 165 | TP : 335 |

Berdasarkan Tabel 4.24, *confusion matrix* menunjukkan bahwa model SVM kernel RBF pada *balanced data* rasio 90:10 mampu mengklasifikasikan 333 data *fraud* dengan benar, sedangkan 167 data *fraud* lainnya salah diprediksi sebagai *non-fraud*. Pada kelas *non-fraud*, model berhasil mengklasifikasikan 335 data dengan benar, namun masih terdapat 165 data yang salah diprediksi sebagai *fraud*. Hasil tersebut menunjukkan bahwa model menghasilkan distribusi prediksi yang seimbang antara kelas *fraud* dan *non-fraud*. Dibandingkan dengan skenario SVM *baseline balanced data* sebelumnya, penggunaan kernel RBF memberikan peningkatan kecil pada kemampuan klasifikasi.

## 4.2 Pembahasan

Implementasi metode *Support vector machine* yang diintegrasikan dengan *Synthetic Minority Over-sampling Technique* dan *Principal Component Analysis* dalam studi ini dilakukan melalui serangkaian tahapan sistematis untuk

menanggulangi kendala klasifikasi pada data tidak seimbang. Penelitian ini memanfaatkan dataset "*Fraud Detection Dataset*" dari *Kaggle Repository* yang memiliki karakteristik ketimpangan kelas ekstrem, dengan proporsi transaksi *fraud* hanya sebesar 1,7% (85 data) berbanding 98,3% transaksi *non-fraud* dari total sampel 5.000 baris data. Dominasi kelas mayoritas ini menjadi determinan utama yang dapat memicu bias model apabila tidak diintervensi melalui teknik pra-pemrosesan yang tepat.

Tahap awal penelitian difokuskan pada *preprocessing* data yang meliputi seleksi variabel untuk mengeliminasi atribut redundan dan tidak relevan, seperti fitur identitas *merchant* dan *datetime* yang tidak berkontribusi pada pola klasifikasi. Transformasi data dilakukan melalui *label encoding* untuk mengonversi variabel kategorikal menjadi representasi numerik biner, serta penerapan *standard scaling* untuk menormalisasi skala fitur. Standardisasi ini bersifat krusial bagi algoritma SVM guna memastikan pembentukan *hyperplane* yang stabil tanpa dominasi fitur dengan rentang nilai besar.

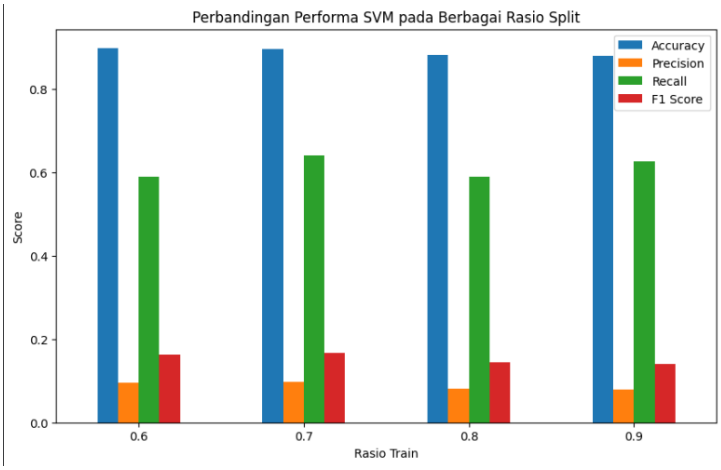
Untuk mengatasi ketidakseimbangan data, teknik SMOTE diterapkan pada data latih saja untuk mencegah terjadinya data *leakage*. Prosedur ini diikuti dengan reduksi dimensi menggunakan PCA yang parameterisasinya didasarkan pada *cumulative variance* di atas 95%. Integrasi PCA terbukti mampu mereduksi kompleksitas komputasi dan meminimalkan risiko overfitting tanpa mengorbankan integritas informasi esensial dari dataset asli.

Model klasifikasi SVM dengan kernel linear dipilih berdasarkan efisiensi dan kompatibilitasnya terhadap data hasil reduksi dimensi. Melalui 16 metode

pengujian yang dievaluasi menggunakan *confusion matrix*, hasil penelitian menunjukkan bahwa sinergi antara SMOTE dan PCA memberikan dampak yang cukup besar terhadap peningkatan metrik *recall* dan *F1-score* pada kelas *fraud*. Secara teoretis dan empiris, penerapan SMOTE berhasil meningkatkan sensitivitas model dalam mendeteksi transaksi ilegal, sementara PCA menjaga stabilitas akurasi dan efisiensi operasional sistem. Kombinasi ini menghasilkan model klasifikasi yang tidak hanya akurat secara global, namun juga memiliki keseimbangan performa yang mumpuni dalam mendeteksi kedua kelas transaksi.

### 4.2.1. Analisis Model Terbaik

Sebagai titik awal analisis, penelitian ini terlebih dahulu menguji algoritma SVM menggunakan distribusi data asli yang masih bersifat *imbalance*d. Skenario ini digunakan sebagai model *baseline* untuk menggambarkan kemampuan dasar SVM dalam membedakan transaksi *fraud* dan *non-fraud* tanpa penerapan teknik penyeimbangan data maupun reduksi dimensi. Hasil pengujian pada model *baseline* menjadi acuan untuk mengevaluasi pengaruh penerapan SMOTE, PCA, dan kombinasi keduanya terhadap performa klasifikasi. Visualisasi hasil pengujian disajikan pada Gambar 4.1.

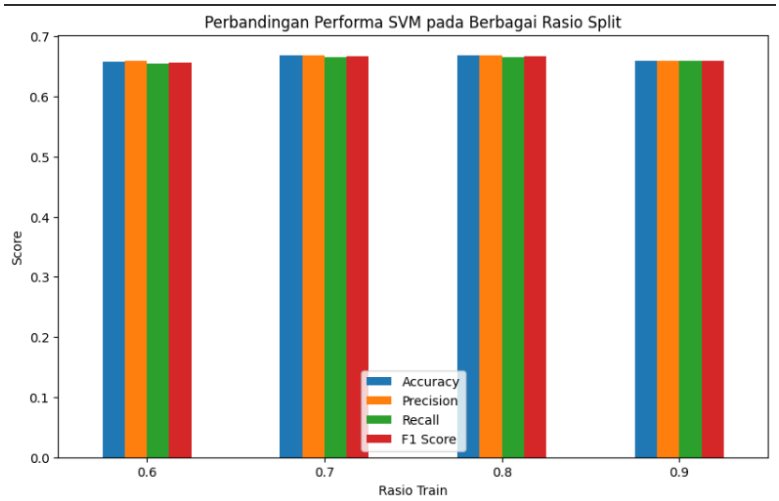



Gambar 4. 1 Perbandingan perfirma SVM *baseline* linear *imbalance* data pada berbagai rasio split

Hasil pengujian menunjukkan tingkat akurasi yang relatif stabil pada rentang 0,88–0,91. Namun, tingginya angka akurasi tersebut cenderung merepresentasikan keberhasilan klasifikasi pada kelas mayoritas (*non-fraud*) dan bukan merupakan indikator performa deteksi kecurangan yang andal. Hal ini terkonfirmasi melalui rendahnya nilai metrik evaluasi lainnya, di mana *Precision*hanya berada pada kisaran 0,07–0,10, yang menandakan tingginya frekuensi kesalahan prediksi positif (*False Positive*). Meskipun nilai *recall* menunjukkan tren peningkatan pada rasio 70:30 dan 80:20 (0,59–0,64) seiring bertambahnya volume data latih, nilai *F1-score* yang tertahan di angka 0,16–0,18. Dapat disimpulkan bahwa pada metode *baseline*, model SVM masih sangat terpengaruh oleh fenomena ketidakseimbangan kelas. Kesenjangan performa antar metrik ini mengindikasikan perlunya metode penyeimbangan data atau optimasi parameter guna meningkatkan kapabilitas deteksi pada kelas minoritas.

Berdasarkan komparasi keseluruhan skenario pengujian, model klasifikasi dengan performa terbaik dan paling tangguh adalah metode *Support vector machine*

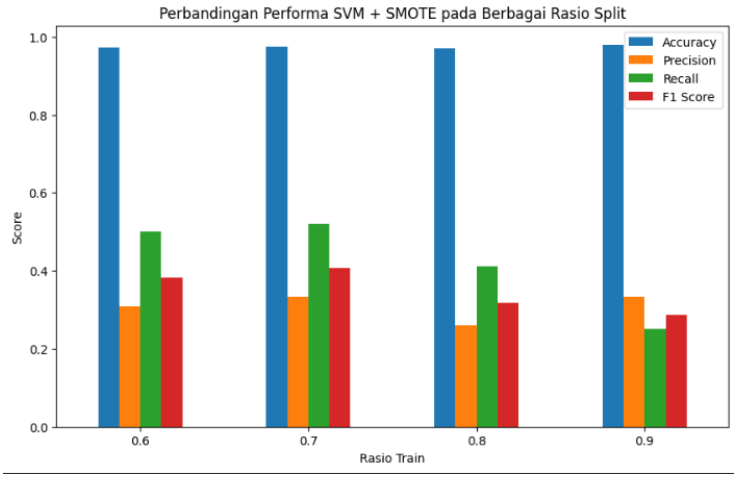
*baseline* pada data seimbang menggunakan kernel *Radial Basis Function* (RBF). Skenario ini menghasilkan distribusi performa yang sangat stabil, di mana tingkat akurasi, presisi, *recall*, dan *F1-score* secara konsisten mencapai angka 67% pada rasio pembagian data optimal 70:30 dan 80:20 Seperti ditunjukkan pada gambar 4.1. Keunggulan utama dari model ini adalah kemampuannya dalam mengeliminasi *accuracy paradox*. Kombinasi data yang proporsional dan fleksibilitas *hyperplane non-linear* dari kernel RBF memungkinkan model memisahkan margin kelas *fraud* dan *non-fraud* secara objektif tanpa bias dari kelas mayoritas.




Gambar 4. 2 Perbandingan performa SVM *baseline* Kernel RBF pada berbagai rasio split

Sebagai pembanding, skenario SVM yang diintegrasikan dengan *Synthetic Minority Over-sampling Technique* pada kondisi data awal yang tidak seimbang juga mencatatkan hasil yang cukup baik dalam mengangkat performa kelas minoritas. Performa terbaik pada skenario ini dicapai pada rasio 70:30, dengan capaian *recall* sebesar 52%, *F1-score* 41%, dan akurasi global yang tetap tinggi di angka 97% seperti yang ditunjukkan pada gambar 4.2. Namun, keterbatasan utama

model ini terletak pada rendahnya nilai presisi 33%, yang mengindikasikan bahwa model masih memberikan tingkat *False Positive* yang cukup tinggi terhadap transaksi normal.




Gambar 4. 3 Perbandingan performa model SVM+SMOTE pada berbagai rasio split

Hasil penelitian menunjukkan bahwa model terbaik bergantung pada karakteristik data yang digunakan. Pada kondisi *balanced data*, metode SVM dengan kernel RBF menghasilkan performa klasifikasi yang paling seimbang antara kedua kelas. Sementara itu, pada kondisi dataset yang bersifat *imbalanced*, penerapan SMOTE memberikan peningkatan kemampuan deteksi transaksi *fraud* dibandingkan model *baseline*. Temuan ini menunjukkan bahwa penanganan ketidakseimbangan data memiliki pengaruh yang lebih besar terhadap peningkatan performa model dibandingkan penambahan kompleksitas algoritma, sedangkan penerapan PCA lebih berperan dalam menjaga efisiensi komputasi tanpa mengurangi kualitas klasifikasi yang dihasilkan.

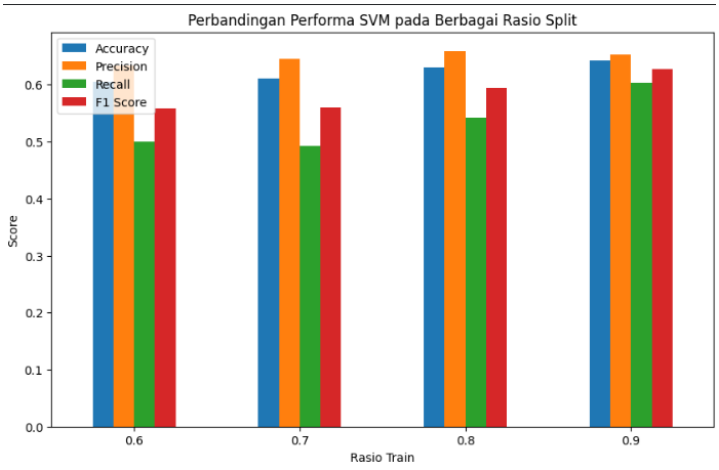
### 4.2.2. Pengaruh Penggunaan *Balanced Data* dan *Imbalance Data* Pada Performa Model

Distribusi kelas pada dataset berdampak banyak terhadap performa model klasifikasi. Pada kondisi *imbalanced data*, model cenderung bias dalam mempelajari pola kelas *non-fraud*. Kondisi ini menghasilkan tingkat akurasi global yang tinggi, namun sensitivitas model (presisi, *recall*, dan *F1-score*) sangat lemah dalam mendeteksi kelas minoritas. Hal ini membuktikan bahwa pada data yang tidak seimbang, tingginya akurasi tidak dapat dijadikan tolok ukur efektivitas deteksi kecurangan. Ketidakseimbangan data (*imbalance*) menjadi penyebab utama terjadinya *accuracy* paradox, di mana model menunjukkan akurasi global tinggi (hingga 90%) namun gagal mengidentifikasi kelas minoritas secara presisi. Ketika model dievaluasi menggunakan data seimbang (2.500 transaksi *fraud* dan 2.500 transaksi *non-fraud*), dominasi prediksi pada kelas mayoritas berhasil dieliminasi. Distribusi metrik presisi dan *recall* menjadi jauh lebih seimbang pada kedua kelas, meskipun hal ini diikuti dengan penyesuaian akurasi global menjadi sekitar 60%–67%. Hal ini membuktikan bahwa proporsi kelas yang proporsional merupakan prasyarat fundamental agar *hyperplane* SVM dapat memisahkan margin kelas secara objektif.

### 4.2.3. Pengaruh Kernel Linear dan RBF terhadap Performa Model

Selain distribusi data, jenis kernel yang digunakan pada algoritma SVM turut memengaruhi hasil klasifikasi yang diperoleh. Pada penelitian ini, pengujian dilakukan menggunakan kernel linear dan kernel *Radial Basis Function* (RBF) untuk melihat kemampuan masing-masing kernel dalam memisahkan transaksi

*fraud* dan *non-fraud*. Perbedaan mekanisme pembentukan batas keputusan pada kedua kernel menyebabkan perbedaan performa dalam mengenali pola transaksi yang terdapat pada dataset. Berdasarkan hasil pengujian, kernel linear mampu menghasilkan performa yang cukup baik dan stabil, terutama pada kelas *non-fraud*. Namun, kemampuan deteksi terhadap transaksi *fraud* masih terbatas karena pola transaksi kecurangan cenderung lebih kompleks dan tidak selalu dapat dipisahkan secara linier. Akibatnya, meskipun model mampu mempertahankan tingkat akurasi yang tinggi, performa pada kelas *fraud* belum menunjukkan hasil yang optimal. Kondisi ini mengindikasikan bahwa batas pemisah linear belum sepenuhnya mampu merepresentasikan karakteristik transaksi *fraud* yang terdapat dalam dataset. Visualisasi perbandingan performa kernel linear dan RBF ditunjukkan pada Gambar 4.4.



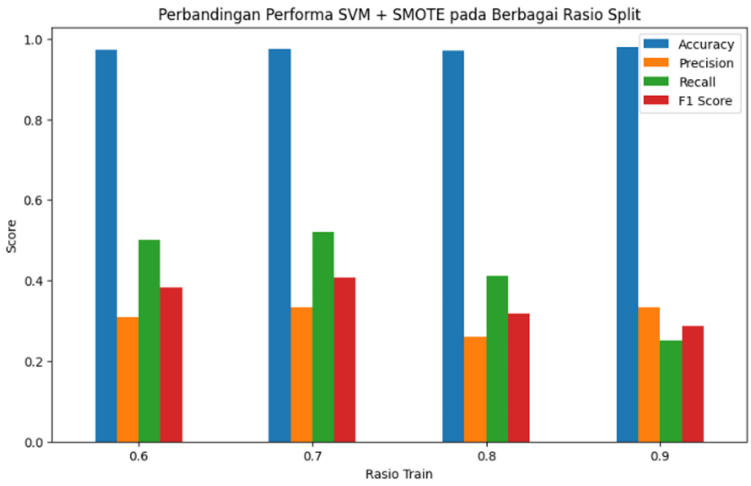

Gambar 4. 4 Perbandingan performa SVM BAsline kernel linear pada berbagai rasio split

Berbeda dengan kernel *linear*, kernel RBF mampu membentuk batas keputusan *non-linear* sehingga hubungan antar fitur yang kompleks dapat dipetakan

ke ruang dimensi yang lebih tinggi. Hasil pengujian menunjukkan bahwa penggunaan kernel RBF menghasilkan performa klasifikasi yang lebih baik dibandingkan kernel *linear*. Model mampu menghasilkan nilai *precision*, *recall*, dan *F1-score* yang lebih seimbang antara kelas *fraud* dan *non-fraud* sehingga kemampuan generalisasi model dalam mengklasifikasikan kedua kelas menjadi lebih baik.

### 4.2.4. Pengaruh Penerapan *Synthetic Minority Over-sampling Technique*

Analisis terhadap visualisasi grafik performa SVM dan SMOTE pada gambar 4.4 menunjukkan dinamika metrik evaluasi yang kontras di seluruh metode pembagian data. Meskipun nilai *Accuracy* menunjukkan stabilitas yang konsisten pada kisaran 97%. Kesenjangan antara akurasi dengan metrik *Precision Recall*, dan *F1-score* menegaskan bahwa model masih menghadapi kendala dalam mengklasifikasikan kelas minoritas secara presisi meskipun teknik oversampling telah diterapkan.

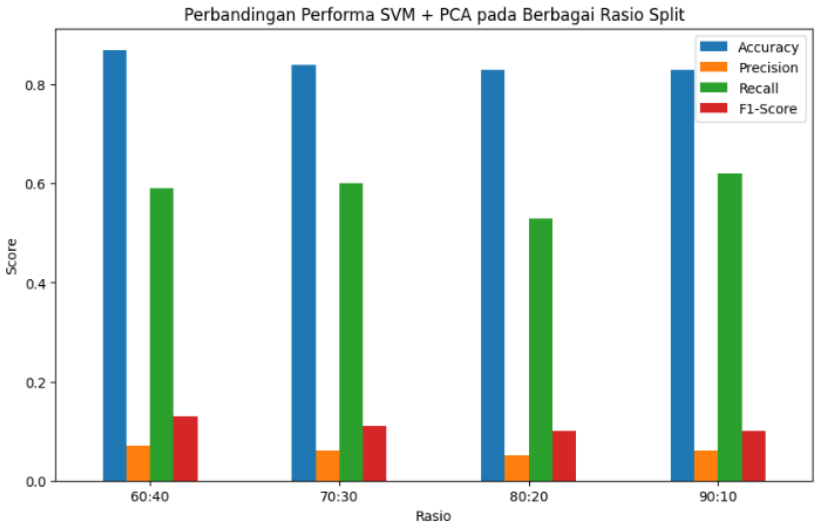



Gambar 4. 5 Perbandingan Performa SVM + SMOTE Pada berbagai rasio split

Metrik *Recall* mencapai performa optimal pada rasio 70:30 dengan nilai puncak mendekati 0,52, namun mengalami degradasi drastis hingga 0,25 pada rasio 90:10. Fenomena ini mengisyaratkan bahwa reduksi data uji hingga batas minimal mengakibatkan model kehilangan kemampuan generalisasi dalam mengenali variasi sampel minoritas. Sementara itu nilai *Precision*yang rendah (0,26–0,33) merepresentasikan tingginya angka *False Positive,* dan *F1-score* menunjukkan tren penurunan linear seiring bertambahnya persentase data latih. Secara kolektif rasio 70:30 diidentifikasi sebagai metode paling moderat karena mampu mempertahankan sensitivitas (*Recall*) pada level tertinggi, meskipun keseimbangan performa ideal di seluruh parameter evaluasi masih menjadi tantangan utama.

### 4.2.5. Pengaruh Penerapan *Principal Component Analysis* (PCA)

Selain menangani ketidakseimbangan kelas, penelitian ini juga mengevaluasi pengaruh reduksi dimensi menggunakan PCA terhadap performa model SVM. Melalui PCA, sejumlah fitur yang saling berkorelasi ditransformasikan menjadi beberapa komponen utama yang mampu merepresentasikan sebagian besar informasi dari data asli. Pada penelitian ini, pemilihan komponen dilakukan dengan mempertahankan cumulative explained variance lebih dari 95%, sehingga proses reduksi dimensi dapat dilakukan tanpa kehilangan informasi yang signifikan. Dengan cara tersebut, model diharapkan mampu bekerja pada data yang lebih sederhana sekaligus mempertahankan kemampuan klasifikasi yang optimal.




Gambar 4. 6 Perbandingan Performa SVM+PCA pada berbagai rasio split

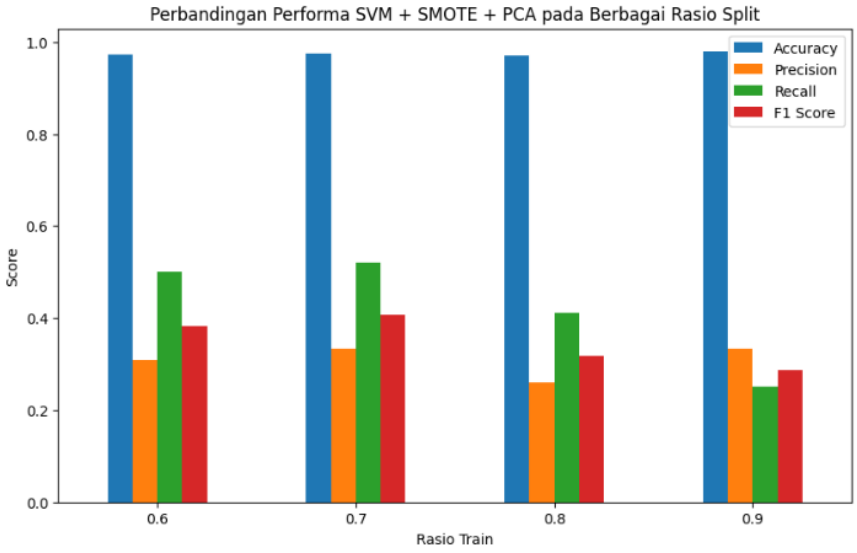
Gambar 4.5 menunjukkan hasil pengujian model SVM yang dikombinasikan dengan *Principal Component Analysis* PCA memperlihatkan nilai *accuracy* menunjukkan kecenderungan yang cukup tinggi pada seluruh rasio pembagian data, dengan nilai berkisar antara 0,83 hingga 0,88. Nilai *accuracy* tertinggi diperoleh pada rasio 60:40, sedangkan rasio lainnya menunjukkan sedikit penurunan, namun tetap berada pada tingkat yang relatif tinggi. Hal ini menunjukkan bahwa model SVM dengan reduksi dimensi menggunakan PCA mampu mengklasifikasikan sebagian besar data dengan benar, khususnya pada kelas mayoritas.

Namun tingginya sensitivitas tersebut tidak dibarengi dengan tingkat presisi yang memadai. Nilai *Precision*dan *F1-score* masih relatif rendah pada semua rasio pembagian data. Kondisi ini menunjukkan bahwa kemampuan model dalam memprediksi kelas *fraud* secara tepat masih terbatas. Sementara itu, nilai *recall* berada pada kisaran 0,53 hingga 0,62, yang menunjukkan bahwa model masih

mampu mendeteksi sebagian data *fraud* meskipun belum optimal. Hasil tersebut menandakan bahwa meskipun model agresif dalam deteksi, keseimbangan antara ketepatan dan jangkauan deteksi belum tercapai. Permasalahan utama pada penelitian ini tetap berkaitan dengan ketidakseimbangan distribusi kelas, sehingga diperlukan pendekatan tambahan seperti teknik penyeimbangan data untuk meningkatkan performa deteksi pada kelas *fraud*.

### 4.2.6. Pengaruh Kombinasi SMOTE dan PCA

Kombinasi SMOTE dan PCA dilakukan dengan tujuan untuk mengatasi dua permasalahan utama pada dataset transaksi kartu kredit, yaitu ketidakseimbangan distribusi kelas dan tingginya dimensi data. SMOTE diterapkan untuk meningkatkan representasi kelas minoritas, sedangkan PCA digunakan untuk menyederhanakan struktur fitur tanpa menghilangkan sebagian besar informasi penting. Visualisasi hasil pengujian ditunjukkan pada gambar 4.6.




Gambar 4. 7 Perbandingan performa SVM+SMOTE+PCA pada berbagai rasio split

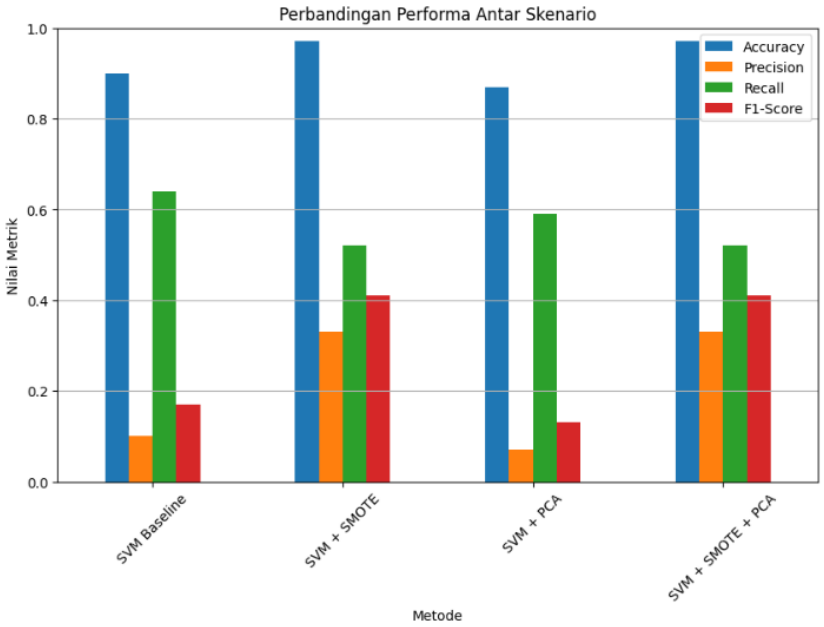
Hasil analisis terhadap visualisasi grafik performa model SVM yang diintegrasikan dengan teknik SMOTE dan PCA menunjukkan karakteristik yang dinamis pada berbagai rasio pembagian data. Secara umum, nilai *accuracy* berada pada tingkat yang sangat tinggi dan konsisten stabil mendekati 0,98 di seluruh variasi rasio. Namun evaluasi menggunakan metrik yang lebih sensitif terhadap ketidakseimbangan kelas mengungkapkan pola degradasi performa seiring bertambahnya proporsi data pelatihan. Nilai *precision*berada pada kisaran 0,26–0,34, dengan performa terbaik diperoleh pada rasio 70:30. Hasil ini menunjukkan bahwa kemampuan model dalam membedakan transaksi *fraud* dan *non-fraud* masih terbatas, sehingga beberapa transaksi normal masih teridentifikasi sebagai *fraud*. Meskipun demikian, penerapan SMOTE dan PCA mampu mempertahankan tingkat ketepatan prediksi yang relatif stabil pada seluruh skenario pengujian.

Variasi yang lebih jelas terlihat pada nilai *recall*. Rasio 70:30 menghasilkan kemampuan deteksi *fraud* tertinggi, sedangkan rasio 90:10 menunjukkan performa terendah. Temuan ini mengindikasikan bahwa peningkatan proporsi data latih tidak selalu berbanding lurus dengan kemampuan model dalam mengenali transaksi *fraud*. Ketika jumlah data uji semakin kecil, variasi pola *fraud* yang tersedia untuk proses evaluasi juga berkurang sehingga kemampuan generalisasi model menjadi kurang optimal.

Pola yang sama terlihat pada nilai *F1-score*, yang mencapai hasil terbaik pada rasio 70:30. Hasil tersebut menunjukkan bahwa rasio 70:30 memberikan keseimbangan yang paling baik antara ketepatan prediksi dan kemampuan deteksi *fraud*. Dengan demikian, dibandingkan rasio lainnya, konfigurasi ini dapat

dianggap sebagai skenario yang paling optimal untuk kombinasi SVM, SMOTE, dan PCA pada penelitian ini.

Hasil eksperimen menunjukkan bahwa performa pada metode SVM dengan SMOTE dan PCA identik dengan hasil pada metode SVM dengan SMOTE tanpa PCA. Hal ini mengindikasikan bahwa penerapan PCA pada dataset tidak memberikan pengaruh berarti terhadap performa klasifikasi. Kondisi tersebut disebabkan oleh jumlah fitur pada dataset yang relatif tidak terlalu besar serta penggunaan threshold variansi PCA yang tinggi, sehingga sebagian besar informasi penting pada data tetap dipertahankan dalam komponen utama. Akibatnya, struktur data yang digunakan oleh model SVM dalam proses pembelajaran tidak mengalami perubahan yang berarti, sehingga menghasilkan performa klasifikasi yang sama dengan metode tanpa PCA. Perbandingan seluruh metode disajikan pada gambar 4.8.




Gambar 4. 8 Perbandingan Performa antar metode

Perbandingan hasil pengujian menunjukkan bahwa model SVM *baseline* masih mengalami kesulitan dalam mendeteksi transaksi *fraud* meskipun menghasilkan nilai akurasi yang relatif tinggi. Kondisi ini mengindikasikan adanya kecenderungan model untuk lebih mengutamakan kelas *non-fraud* yang jumlahnya mendominasi dataset. Akibatnya, performa pada kelas *fraud* belum dapat merepresentasikan kemampuan deteksi yang sebenarnya karena model masih dipengaruhi oleh ketidakseimbangan distribusi data.

Penerapan PCA tanpa disertai teknik penyeimbangan data tidak memberikan perbaikan performa yang signifikan. Bahkan, terjadi penurunan pada beberapa metrik evaluasi dibandingkan model *baseline*. Hasil tersebut menunjukkan bahwa reduksi dimensi belum mampu meningkatkan kemampuan model dalam membedakan karakteristik transaksi *fraud* dan *non-fraud*. Kemungkinan terdapat informasi yang berperan dalam proses klasifikasi yang berkurang selama proses transformasi fitur, sehingga kemampuan deteksi *fraud* tidak mengalami peningkatan.

Berbeda dengan PCA, penerapan SMOTE memberikan dampak yang jauh lebih terlihat terhadap performa model. Penambahan sampel sintetis pada kelas *fraud* membantu mengurangi ketimpangan distribusi data sehingga model memperoleh representasi yang lebih baik terhadap kelas minoritas. Dampaknya terlihat pada peningkatan kemampuan deteksi *fraud* yang lebih konsisten dibandingkan skenario sebelumnya. Ketika PCA dikombinasikan dengan SMOTE, performa yang dihasilkan tetap berada pada tingkat yang sama dengan penggunaan SMOTE saja. Temuan ini menunjukkan bahwa peningkatan performa lebih banyak

dipengaruhi oleh keberhasilan SMOTE dalam menangani class *imbalance*, sementara kontribusi PCA terhadap hasil klasifikasi relatif kecil.

Secara keseluruhan, hasil penelitian menunjukkan bahwa faktor yang paling berpengaruh terhadap peningkatan performa model adalah penanganan ketidakseimbangan data, bukan reduksi dimensi. Oleh karena itu, kombinasi SVM dengan SMOTE, baik dengan maupun tanpa PCA, menjadi pendekatan yang paling efektif dalam penelitian ini karena mampu meningkatkan kemampuan model dalam mengenali transaksi *fraud* tanpa mengorbankan performa pada kelas *non-fraud*.

### 4.2.6. Tinjauan Keislaman Dalam Klasifikasi Kecurangan Kartu Kredit

Penelitian ini tidak hanya memiliki kontribusi dalam bidang teknologi informasi, tetapi juga relevan dengan nilai-nilai yang diajarkan dalam Islam. Upaya mendeteksi dan mencegah kecurangan transaksi merupakan bentuk penerapan prinsip kejujuran, keadilan, dan amanah dalam aktivitas ekonomi. Dalam Islam, segala bentuk tindakan yang merugikan pihak lain melalui penipuan atau manipulasi dilarang karena bertentangan dengan prinsip keadilan yang menjadi dasar dalam bermuamalah. Hal tersebut sebagaimana dijelaskan dalam Surah *An-Nahl* ayat 90, yang menegaskan pentingnya menegakkan keadilan dan berbuat kebaikan dalam kehidupan bermasyarakat dan aktivitas ekonomi.:

إِنَّ ٱللَّهَ يَأۡمُرُ بِٱلۡعَدۡلِ وَٱلۡإِحۡسَٰنِ وَإِيتَآيِٕ ذِي ٱلۡقُرۡبَىٰ وَيَنۡهَىٰ عَنِ ٱلۡفَحۡشَآءِ وَٱلۡمُنكَرِ وَٱلۡبَغۡيِۚ يَعِظُكُمۡ لَعَلَّكُمۡ تَذَكَّرُونَ

*"Sesungguhnya Allah menyuruh (kamu) berlaku adil dan berbuat kebajikan, memberi kepada kaum kerabat, dan Allah melarang dari perbuatan keji, kemungkaran dan permusuhan. Dia memberi pengajaran kepadamu agar kamu dapat mengambil pelajaran."* (Q.S. *An-Nahl* [16]: 90)

Menurut Tafsir Ibnu Katsir, Allah memerintahkan hamba-hambanya agar berlaku adil, yakni proposional dan seimbang, serta anjuran berbuat kebaikan (Mutmainah, n.d.). ayat tersebut menegaskan bahwa prinsip keadilan *(‘adl)* merupakan perintah langsung dari Allah SWT yang dalam konteks transaksi keuangan modern keadilan dapat dimaknai sebagai perlindungan terhadap hak-hak pihak yang terlibat, baik nasabah maupun lembaga keuangan. Praktik curang merupakan bentuk kezaliman karena merugikan pihak lain secara finansial dan melanggar prinsip keadilan yang diperintahkan dalam ayat tersebut. Sehingga pengembangan sistem klasifikasi kecurangan berbasis *machine learning* dapat dipandang sebagai ikhtiar ilmiah untuk menegakkan nilai keadilan dalam sistem keuangan.

Rasulullah SAW juga menegaskan larangan terhadap praktik penipuan dalam sebuah hadits yang diriwayatkan oleh Imam Muslim:

مَنْ غَشَّ فَلَيْسَ مِنِّي

Yang artinya: *“Barang siapa yang menipu maka ia bukan termasuk golongan kami.*”(HR. *Muslim,* No. 102)

Institusi perbankan dan lembaga keuangan memegang amanah dalam mengelola dana nasabah. Penggunaan teknologi *machine learning* untuk mendeteksi transaksi curang merupakan bentuk tanggung jawab profesional dalam menjaga amanah tersebut. Sistem klasifikasi berbasis SVM yang dirancang untuk meminimalkan kesalahan klasifikasi mencerminkan upaya untuk menegakkan keadilan serta meminimalisir tindak penipuan.

Dengan demikian integrasi antara sains dan Islam dalam penelitian ini terletak pada keselarasan tujuan antara pengembangan teknologi dan nilai-nilai syariah. Secara ilmiah, penelitian ini bertujuan meningkatkan akurasi dan stabilitas sistem klasifikasi kecurangan melalui pendekatan komputasional. Penelitian ini mendukung prinsip keadilan *(‘adl)*, kejujuran *(shidq)*, amanah, serta larangan terhadap penipuan *(ghisy)* dan pengambilan harta secara batil.

# BAB V

# KESIMPULAN DAN SARAN

## 5.1. Kesimpulan

Berdasarkan hasil penelitian, algoritma *Support vector machine* (SVM) dapat digunakan untuk mengklasifikasikan transaksi kartu kredit ke dalam kategori *fraud* dan *non-fraud*. Namun, performa model sangat dipengaruhi oleh distribusi data yang digunakan. Pada kondisi data tidak seimbang, model cenderung menghasilkan akurasi yang tinggi tetapi kurang optimal dalam mendeteksi transaksi *fraud*. Hasil pengujian menunjukkan bahwa kombinasi SVM dan SMOTE, baik dengan maupun tanpa PCA, memberikan performa terbaik dengan nilai *F1-score fraud* tertinggi sebesar 0,41. Selain itu, rasio pembagian data 70:30 menghasilkan keseimbangan terbaik antara kemampuan deteksi *fraud* dan performa model secara keseluruhan.

Penerapan SMOTE terbukti memberikan pengaruh yang lebih besar dibandingkan PCA dalam meningkatkan performa klasifikasi, terutama pada kemampuan model mendeteksi transaksi *fraud*. Sementara itu, PCA berperan dalam menyederhanakan dimensi data dan menjaga efisiensi komputasi tanpa memberikan peningkatan performa yang signifikan. Dengan demikian, penanganan ketidakseimbangan data melalui SMOTE menjadi faktor utama yang menentukan keberhasilan klasifikasi kecurangan transaksi kartu kredit pada penelitian ini.

## 5.2 Saran

1. Penelitian berikutnya dapat melakukan eksplorasi terhadap variasi parameter SVM, seperti nilai C atau penggunaan kernel *non-linear*, untuk mengetahui kemungkinan peningkatan performa klasifikasi yang lebih optimal.
2. Penelitian selanjutnya dapat membandingkan performa SVM dengan algoritma klasifikasi lain seperti *Random Forest, Gradient Boosting,* atau *Neural Network* untuk memperoleh gambaran yang lebih komprehensif mengenai metode terbaik dalam deteksi *fraud.*
3. Implementasi model dalam sistem berbasis real-time atau integrasi dengan sistem monitoring transaksi aktual dapat menjadi langkah lanjutan guna menguji performa model dalam lingkungan operasional.

Dengan adanya pengembangan tersebut, diharapkan sistem deteksi kecurangan transaksi kartu kredit dapat semakin akurat, adaptif, dan mampu memberikan kontribusi nyata dalam meningkatkan keamanan sistem keuangan digital.

## DAFTAR PUSTAKA


Alshawi, B. (2024). Comparison of SVM kernels in Credit Card *Fraud* Detection using GANs. *International Journal of Advanced Computer Science and Applications*, *15*(1), 330–337. https://doi.org/10.14569/IJACSA.2024.0150131

Arifudin, O., Juhadi, Sofyan, Y., Tanjung, R., & Rusmana, F. D. (2021). Pengaruh Kelas Sosial, Pengalaman dan Gaya Hidup terhadap perilaku Penggunaan Kartu Kredit. *Ilmiah MEA*, *5*(1), 286–298. http://www.journal.stiemb.ac.id/index.php/mea/article/view/868/381

Btoush, E., Kobbaey, T., & Tamimi, H. (2026). *Machine learning-Based Cyber Fraud Detection : A Comparative Study of Resampling Methods for Imbalanced Credit Card Data*. 1–34.

Chawla, N. V, Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). *SMOTE : Synthetic Minority Over-sampling Technique*. *16*, 321–357.

Chunchu, A. (2024). Artificial Intelligence in Retail *Fraud* Detection : Enhancing Payment Security Artificial Intelligence in Retail *Fraud* Detection : Enhancing Payment Security. *International Research Journal of Engineering and Technology (IRJET)*, *11*(July 2024), 767–781. www.irjet.net

Du, J. (n.d.). *A Structured Reasoning Framework for Unbalanced Data Classification Using Probabilistic Models*.

Fadilah, W. R. U., Agfiannisa, D., & Azhar, Y. (2020). Analisis Prediksi Harga Saham PT. Telekomunikasi Indonesia Menggunakan Metode *Support vector machine*. *Fountain of Informatics Journal*, *5*(2), 45. https://doi.org/10.21111/fij.v5i2.4449

Feng, X., & Kim, S. (2024). *Novel Machine learning Based Credit Card Fraud Detection Systems*.

Gawali, T. K., Albert, A. H., Deshmuk, S. S., Engineering, E., Engineering, E., Engineering, E., Engineering, C., Chinchwad, P., & Engineering, E. (2025). *Machine learning-Based Credit Card Fraud Detection : A Comparative Study Using Logistic Regression , SVM , and*. *11*(8), 374–380.

Gedela, B., & Karthikeyan, P. R. (2023). Credit Card *Fraud* Detection Using *Support vector machine* Algorithm in Comparison with Various *Machine learning* Algorithms to Measure *Accuracy*, Sensitivity, Specificity, *Precision* and F-Score. *AIP Conference Proceedings*, *2587*(1). https://doi.org/10.1063/5.0150792

Gee, S. (2014). *Fraud* and *Fraud* Detection. In *Fraud and Fraud Detection*. https://doi.org/10.1002/9781118936764

Han, Y. (2024). *applied sciences Enhancing Machine learning Models Through PCA , SMOTE-ENN , and Stochastic Weighted Averaging*.

Kumar Manjhvar, A., & Goyal, R. (2020). Review on Credit Card *Fraud* Detection using Data Mining Classification Techniques & *Machine learning* Algorithms. *IJRAR (International Journal of Research and Analytical Reviews)*, *7*(1), 972–975. www.ijrar.org

Moumeni, L., Saber, M., Slimani, I., Elfarissi, I., & Bougroun, Z. (2022). *Machine learning* for Credit Card *Fraud* Detection. *Lecture Notes in Electrical Engineering*, *745*(24), 211–221. https://doi.org/10.1007/978-981-33-6893-4_20

Mutmainah, I. (n.d.). *Etika Ekonomi Islam Dalam Surat an-Nahl Ayat 90*. *x*.

Rakovski, C. (n.d.). *On the Application of Principal Component Analysis to Classification Problems*. 1–6. https://doi.org/10.5334/dsj-2021-026

Rebala, G., Ravi, A., & Churiwala, S. (2019). *Machine learning* Definition and Basics BT - An Introduction to *Machine learning*. In *An Introduction to Machine learning*. https://doi.org/10.1007/978-3-030-15729-6_1

Rivki, M., Bachtiar, A. M., Informatika, T., Teknik, F., & Indonesia, U. K. (2021). *Penerapan Data Mining Clasification Dalam Pengecekan Kecurangan Transaksi Yang Akurat*. *112*.

Sadineni, P. K. (2020). Detection of *fraud*ulent transactions in credit card using *machine learning* Algorithms. *Proceedings of the 4th International Conference on IoT in Social, Mobile, Analytics and Cloud, ISMAC 2020*, 659–660. https://doi.org/10.1109/I-SMAC49090.2020.9243545

Sudha, C., & Akila, D. (2021). Credit card *fraud* detection system based on operational transaction features using SVM and random forest classifiers. *Proceedings of 2nd International Conference on Computation, Automation and Knowledge Management, ICCAKM 2021*, 133–138. https://doi.org/10.1109/ICCAKM50778.2021.9357709

Ubaidillah. (2018). *Islam dan Pendidikan Karakter (Analisis Nilai Karakter dalam QS: an Nahl:90)*. *2*, 106–122.

Yazid, Y., & Fiananta, A. (2017). Mendeteksi Kecurangan Pada Transaksi Kartu Kredit Untuk Verifikasi Transaksi Menggunakan Metode Svm. *Indonesian Journal of Applied Informatics*, *1*(2), 61–66.

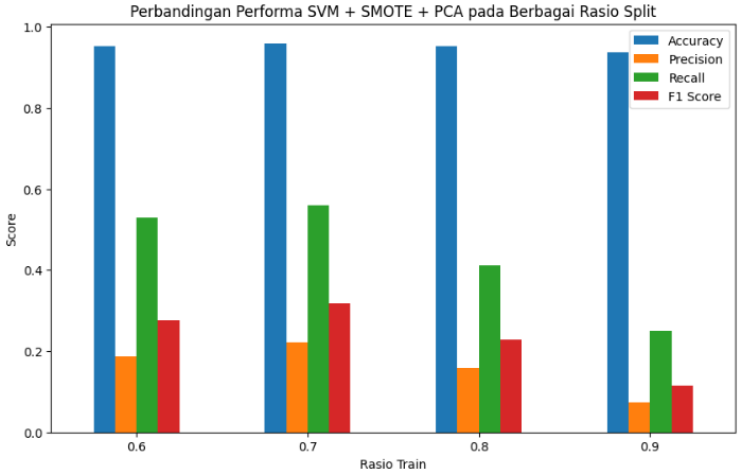
Zhang, D., Bhandari, B., & Black, D. (2020). Credit Card *Fraud* Detection Using Weighted *Support vector machine*. *Applied Mathematics*, *11*(12), 1275–1291. https://doi.org/10.4236/am.2020.1112087

# LAMPIRAN

Percobaan Menggunakan SMOTE 50%

| Rasio Train:Test | *Accuracy* | *Precision* | *Recall* | *F1-score* |
|---|---|---|---|---|
| 60 : 40 | 0.953 | 0.187500 | 0.529412 | 0.276923 |
| 70 : 30 | 0.960 | 0.222222 | 0.560000 | 0.318182 |
| 80 : 20 | 0.953 | 0.159091 | 0.411765 | 0.229508 |
| 90 : 10 | 0.938 | 0.074074 | 0.250000 | 0.114286 |

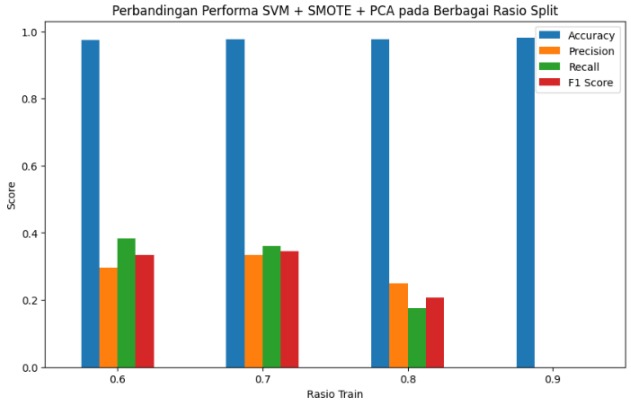
Diagram percobaan SMOTE 50%




Lampiran penggunaan PCA threshold 90 dengan 10 komponen :

| Rasio Train:Test | *Accuracy* | *Precision* | *Recall* | *F1-score* |
|---|---|---|---|---|
| 60 : 40 | 97% | 30% | 38% | 33% |
| 70 : 30 | 98% | 33% | 36% | 35% |
| 80 : 20 | 98% | 25% | 18% | 21% |
| 90 : 10 | 98% | 0% | 0% | 0% |

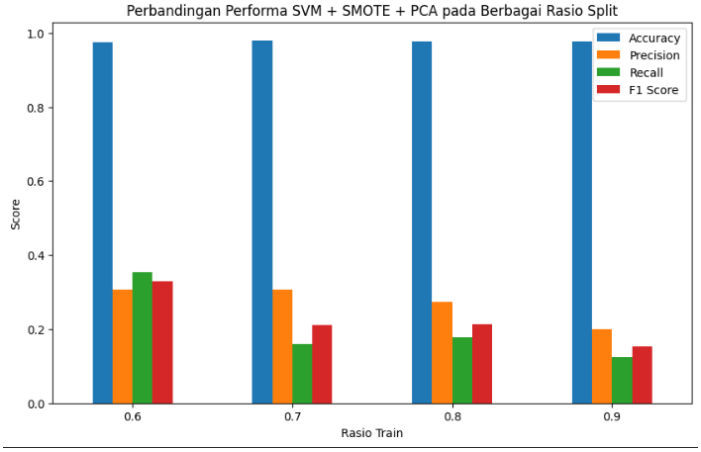
Diagram Perbandingan PCA threshold 90 dengan 10 komponen :



Penggunaan PCA threshold 95 dengan 11 komponen :

| Rasio Train:Test | *Accuracy* | *Precision* | *Recall* | *F1-score* |
|---|---|---|---|---|
| 60:40 | 0.9755 | 0.307692 | 0.352941 | 0.328767 |
| 70:30 | 0.9800 | 0.307692 | 0.160000 | 0.210526 |
| 80:20 | 0.9780 | 0.272727 | 0.176471 | 0.214286 |
| 90:10 | 0.9780 | 0.200000 | 0.125000 | 0.153846 |

Diagram perbandingan PCA threshold 95 dengan 11 komponen :




Penggunaan PCA threshold 99 dengan 12 komponen :

| Rasio Train:Test | *Accuracy* | *Precision* | *Recall* | *F1-score* |
|---|---|---|---|---|
| 60:40 | 0.972500 | 0.309091 | 0.500000 | 0.382022 |
| 70:30 | 0.974667 | 0.333333 | 0.520000 | 0.406250 |
| 80:20 | 0.970000 | 0.259259 | 0.411765 | 0.318182 |
| 90:10 | 0.980000 | 0.333333 | 0.250000 | 0.285714 |

Diagram perbandingan PCA threshold 99 dengan 12 komponen :