

Penerapan Dekomposisi Matriks untuk Reduksi Kompleksitas Komputasi pada Algoritma *Machine Learning*

Safira Hasna Setiyani¹, Yusiana Rahma²

^{1,2}Informatika, Fakultas Hukum Manajemen dan Informatika, Universitas Karya Husada Semarang

¹safirahasna@unkaha.ac.id*, ²yusianarahma@unkaha.ac.id

Abstract

The increasing complexity of machine learning algorithms is often accompanied by higher computational costs particularly when dealing with high-dimensional data. This condition poses significant challenges in terms of computational efficiency and resource utilization. One mathematical approach that can address this issue is the application of linear algebra concepts, specifically matrix decomposition techniques. This study aims to apply matrix decomposition methods to reduce computational complexity in machine learning algorithms without significantly degrading model performance. The proposed approach employs matrix decomposition, such as Singular Value Decomposition (SVD), during the data preprocessing and model training stages. The performance of the algorithms is evaluated by comparing their behavior before and after the application of matrix decomposition in terms of computational time, accuracy, and memory efficiency. The experimental results demonstrate that matrix decomposition can significantly reduce computational complexity and improve learning efficiency, while maintaining stable or only slightly reduced accuracy. These findings indicate that matrix decomposition is an effective and practical approach for optimizing machine learning algorithms, particularly for large-scale and high-dimensional datasets.

Keywords: linear algebra, matrix decomposition, machine learning, singular value decomposition, dimensionality reduction

Abstrak

Perkembangan algoritma *machine learning* yang semakin kompleks sering kali diikuti oleh peningkatan kebutuhan komputasi terutama ketika dihadapkan pada data berdimensi tinggi. Kondisi ini menimbulkan tantangan dalam hal efisiensi waktu komputasi dan penggunaan sumber daya sistem. Salah satu pendekatan matematis yang berpotensi mengatasi permasalahan tersebut adalah pemanfaatan konsep aljabar linier, khususnya metode dekomposisi matriks. Penelitian ini bertujuan untuk menerapkan metode dekomposisi matriks dalam rangka mereduksi kompleksitas komputasi pada algoritma *machine learning* tanpa mengorbankan kinerja model secara signifikan. Metode yang digunakan meliputi penerapan dekomposisi matriks, seperti *Singular Value Decomposition* (SVD), pada tahap praproses data dan proses pembelajaran model. Evaluasi dilakukan dengan membandingkan kinerja algoritma sebelum dan sesudah penerapan dekomposisi matriks berdasarkan parameter waktu komputasi, akurasi, dan efisiensi memori. Hasil penelitian menunjukkan bahwa penerapan dekomposisi matriks mampu mengurangi kompleksitas komputasi secara signifikan serta meningkatkan efisiensi proses pembelajaran, dengan penurunan akurasi yang relatif kecil atau tetap stabil. Temuan ini mengindikasikan bahwa dekomposisi matriks merupakan pendekatan yang efektif dan relevan dalam optimasi algoritma *machine learning*, khususnya untuk pengolahan data berskala besar dan berdimensi tinggi.

Kata kunci: aljabar linier, dekomposisi matriks, *machine learning*, *singular value decomposition*, reduksi

©This work is licensed under a Creative Commons Attribution -ShareAlike 4.0 International License

1. Pendahuluan

Perkembangan teknologi informasi dan komputasi telah mendorong pemanfaatan *machine learning* secara luas dalam berbagai bidang, seperti pengolahan citra, pengenalan pola, sistem rekomendasi, hingga analisis data berskala besar [1] [2]. Seiring dengan meningkatnya kompleksitas permasalahan yang ditangani, algoritma *machine learning* juga mengalami peningkatan dalam hal struktur model dan jumlah parameter yang harus diproses [3]. Kondisi ini berdampak langsung pada meningkatnya kebutuhan komputasi, baik dari sisi waktu pemrosesan maupun penggunaan memori [4]. Tantangan tersebut menjadi semakin signifikan ketika algoritma *machine learning* diterapkan pada data berdimensi tinggi (*high-dimensional data*) [5], yang umum dijumpai dalam era

big data. Oleh karena itu, diperlukan pendekatan yang mampu meningkatkan efisiensi komputasi tanpa mengorbankan kinerja model secara signifikan [6].

Dalam konteks tersebut, aljabar linier memegang peranan fundamental dalam pengembangan dan optimasi algoritma *machine learning* [7]. Representasi data dalam bentuk vektor dan matriks memungkinkan proses komputasi dilakukan secara sistematis dan terstruktur [8]. Banyak algoritma *machine learning*, seperti regresi linier, *principal component analysis* (PCA), *clustering*, dan jaringan saraf tiruan, pada dasarnya dibangun di atas operasi-operasi aljabar linier [9]. Namun, semakin besar ukuran matriks yang terlibat, semakin tinggi pula kompleksitas komputasi yang harus ditangani [10] [11]. Hal ini mendorong perlunya teknik matematis yang dapat

menyederhanakan struktur matriks tanpa kehilangan informasi penting yang terkandung di dalamnya [12].

Salah satu teknik aljabar linier yang banyak digunakan untuk tujuan tersebut adalah dekomposisi matriks [13]. Dekomposisi matriks merupakan proses pemfaktoran suatu matriks menjadi beberapa matriks lain dengan sifat-sifat tertentu, sehingga lebih mudah dianalisis dan dihitung [14]. Metode dekomposisi matriks, seperti *Singular Value Decomposition (SVD)*, *Eigen Decomposition*, *LU Decomposition*, dan *QR Decomposition*, telah terbukti efektif dalam berbagai aplikasi komputasi numerik [15]. Secara khusus, SVD memungkinkan pemisahan informasi utama dari data berdimensi tinggi dengan menghilangkan komponen yang kurang signifikan, sehingga dapat digunakan untuk reduksi dimensi dan penyederhanaan model [16].

Penerapan dekomposisi matriks dalam *machine learning* memberikan potensi besar dalam mereduksi kompleksitas komputasi [17]. Dengan menurunkan dimensi data atau menyederhanakan struktur matriks yang terlibat dalam proses pembelajaran, algoritma dapat berjalan lebih efisien tanpa harus memproses seluruh informasi mentah secara langsung [18]. Pendekatan ini tidak hanya berdampak pada percepatan waktu komputasi, tetapi juga pada efisiensi penggunaan memori dan stabilitas numerik algoritma [19]. Meskipun demikian, penerapan dekomposisi matriks juga memiliki tantangan tersendiri, terutama dalam menjaga keseimbangan antara efisiensi komputasi dan tingkat akurasi model yang dihasilkan.

Beberapa penelitian terdahulu telah mengkaji pemanfaatan dekomposisi matriks dalam konteks *machine learning*. Studi-studi sebelumnya menunjukkan bahwa penggunaan SVD dan teknik dekomposisi lainnya mampu meningkatkan efisiensi algoritma, khususnya pada tahap pra-proses data dan pelatihan model [20]. Penelitian lain juga menekankan peran dekomposisi matriks dalam reduksi dimensi untuk meningkatkan kinerja klasifikasi dan *clustering* [21]. Namun demikian, sebagian penelitian masih berfokus pada satu algoritma atau satu jenis metode dekomposisi saja, sehingga belum memberikan gambaran komprehensif mengenai pengaruh dekomposisi matriks terhadap kompleksitas komputasi secara umum pada algoritma *machine learning*.

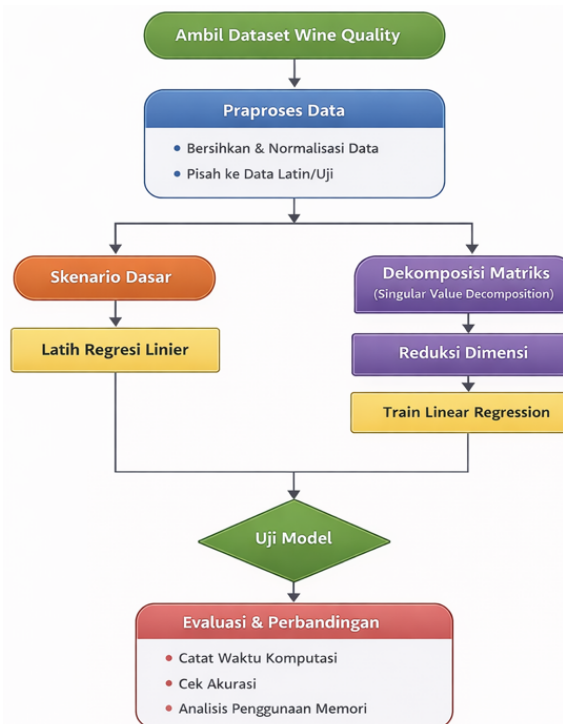
Berdasarkan uraian tersebut, penelitian ini bertujuan untuk menerapkan dan menganalisis metode dekomposisi matriks dalam upaya mereduksi kompleksitas komputasi pada algoritma *machine learning*. Penelitian ini difokuskan pada evaluasi kinerja algoritma sebelum dan sesudah penerapan dekomposisi matriks, dengan mempertimbangkan aspek waktu komputasi, efisiensi memori, dan akurasi model. Diharapkan hasil penelitian ini dapat memberikan kontribusi teoretis dalam penguatan peran aljabar linier dalam *machine learning*, serta kontribusi praktis berupa pendekatan optimasi yang efektif dan

aplikatif untuk pengolahan data berskala besar dan berdimensi tinggi.

Kebaruan penelitian ini terletak pada pendekatan komparatif dan komprehensif dalam menganalisis pengaruh berbagai metode dekomposisi matriks terhadap reduksi kompleksitas komputasi pada beberapa algoritma *machine learning* secara simultan, dengan mengintegrasikan evaluasi waktu komputasi, penggunaan memori, dan akurasi model. Berbeda dengan penelitian sebelumnya yang umumnya berfokus pada satu metode atau satu algoritma tertentu, penelitian ini memberikan kerangka evaluasi yang lebih menyeluruh untuk menentukan metode dekomposisi matriks yang paling efektif pada skenario data berdimensi tinggi dan berskala besar.

2. Metode Penelitian

Penelitian ini menggunakan pendekatan eksperimental kuantitatif dengan tujuan menganalisis pengaruh penerapan metode dekomposisi matriks terhadap kompleksitas komputasi algoritma *machine learning*. Eksperimen dilakukan dengan membandingkan kinerja algoritma *machine learning* sebelum dan sesudah diterapkan teknik dekomposisi matriks. Fokus utama penelitian adalah evaluasi efisiensi komputasi tanpa mengabaikan kinerja model yang diukur melalui akurasi. Alur penelitian pada Gambar 1 berikut.



Gambar 1. Diagram Alur Penelitian

Dataset yang digunakan dalam penelitian ini adalah *Wine Quality Dataset* yang diperoleh dari *UCI Machine Learning Repository*. Dataset ini terdiri dari sekitar 1.600 data sampel dengan 11 fitur numerik yang merepresentasikan karakteristik kimia anggur yang digunakan untuk memprediksi kualitas anggur

berdasarkan berbagai atribut numerik, serta satu variabel target berupa skor kualitas anggur.

Tahap awal penelitian dimulai dengan praproses data, yang meliputi pemisahan antara fitur dan variabel target. Seluruh fitur numerik kemudian dinormalisasi menggunakan metode *standardization* untuk menyamakan skala data, mengingat algoritma dekomposisi matriks dan machine learning sensitif terhadap perbedaan skala antar atribut [22]. Setelah proses normalisasi, dataset dibagi menjadi data latih dan data uji dengan perbandingan 80% untuk pelatihan dan 20% untuk pengujian. Pembagian ini bertujuan untuk memastikan bahwa evaluasi performa model dilakukan secara objektif dan tidak bias terhadap data pelatihan [23].

Selanjutnya, penelitian menerapkan teknik aljabar linier berupa *Singular Value Decomposition* (SVD) untuk melakukan reduksi dimensi pada matriks fitur data latih. SVD digunakan untuk memproyeksikan data ke ruang berdimensi lebih rendah dengan tetap mempertahankan sebagian besar informasi penting yang terkandung dalam data asli [24]. Proses dekomposisi hanya dilakukan pada data latih untuk menghindari terjadinya *data leakage* [25], sedangkan data uji ditransformasikan menggunakan model SVD yang telah dibentuk sebelumnya. Jumlah komponen SVD divariasikan untuk menganalisis pengaruhnya terhadap tingkat kesalahan prediksi.

Setelah tahap reduksi dimensi, model *machine learning* diterapkan untuk melakukan prediksi nilai kualitas anggur. Algoritma regresi digunakan karena variabel target bersifat kontinu. Model dilatih menggunakan dua tahap percobaan, yaitu menggunakan fitur asli tanpa reduksi dimensi sebagai model baseline dan menggunakan fitur hasil reduksi dimensi SVD sebagai model pembanding. Pendekatan ini bertujuan untuk mengevaluasi efektivitas SVD dalam meningkatkan efisiensi komputasi dan kinerja prediksi model [26].

Evaluasi performa model dilakukan menggunakan metrik *Root Mean Squared Error* (RMSE) dan koefisien determinasi (R^2). RMSE digunakan untuk mengukur rata-rata kesalahan prediksi antara nilai aktual dan nilai prediksi, sedangkan R^2 digunakan untuk menilai kemampuan model dalam menjelaskan variasi data target [27]. Selain itu, waktu pelatihan model juga dicatat sebagai indikator efisiensi komputasi [28]. Analisis dilakukan dengan membandingkan hasil evaluasi antara model baseline dan model dengan SVD, serta dengan mengamati hubungan antara jumlah komponen SVD dan nilai RMSE yang dihasilkan.

Hasil akhir penelitian disajikan dalam bentuk tabel perbandingan dan grafik hubungan antara RMSE dan jumlah komponen SVD. Grafik tersebut digunakan untuk mengidentifikasi jumlah komponen optimal yang memberikan keseimbangan terbaik antara kompleksitas model dan tingkat kesalahan prediksi. Dengan demikian, penelitian ini tidak hanya

mengevaluasi akurasi prediksi, tetapi juga memberikan analisis komprehensif mengenai peran aljabar linier, khususnya dekomposisi matriks, dalam meningkatkan efisiensi dan performa algoritma machine learning.

3. Hasil dan Pembahasan

3.1. Dataset dan Praproses Data

Berikut adalah *dataset* dari *Wine Quality Dataset* yang terdiri dari 1.600 data sampel dan 11 fitur numerik yang merepresentasikan karakteristik kimia anggur yang digunakan untuk memprediksi kualitas anggur berdasarkan berbagai atribut numerik, dapat dilihat pada Gambar 2 sebagai berikut.

```

Dataset berhasil dimuat
fixed acidity  volatile acidity  citric acid  residual sugar  chlorides \
0      7.4      0.70      0.00      1.9      0.076
1      7.8      0.88      0.00      2.6      0.098
2      7.8      0.76      0.04      2.3      0.092
3     11.2      0.28      0.56      1.9      0.075
4      7.4      0.70      0.00      1.9      0.076

free sulfur dioxide  total sulfur dioxide  density  pH  sulphates \
0      11.0      34.0  0.9978  3.51  0.56
1      25.0      67.0  0.9968  3.20  0.68
2      15.0      54.0  0.9970  3.26  0.65
3      17.0      60.0  0.9980  3.16  0.58
4      11.0      34.0  0.9978  3.51  0.56

alcohol  quality
0      9.4      5
1      9.8      5
2      9.8      5
3      9.8      6
4      9.4      5
    
```

Gambar 2. Dataset

Pada Gambar 3, seluruh fitur bersifat numerik kontinu sehingga dapat direpresentasikan dalam bentuk matriks dan sesuai untuk penerapan metode aljabar linier.

```

Dataset Shape: (1599, 12)

Column Names:
['fixed acidity', 'volatile acidity', 'citric acid', 'residual sugar',
 'chlorides', 'free sulfur dioxide', 'total sulfur dioxide', 'density', 'pH',
 'sulphates', 'alcohol', 'quality']

fixed acidity  volatile acidity  citric acid  residual sugar  chlorides
1409           6.0           0.51         0.00           2.1           0.064
1475           5.3           0.47         0.11           2.2           0.048
1267           10.4          0.43         0.50           2.3           0.068
727            6.4           0.57         0.02           1.8           0.067

free sulfur dioxide  total sulfur dioxide  density  pH  sulphates  alcohol  quality
40.0                54.0  0.99500  3.54  0.93  10.700000  6
16.0                89.0  0.99182  3.54  0.88  13.566667  7
13.0                19.0  0.99600  3.10  0.87  11.400000  6
4.0                 11.0  0.99700  3.46  0.68  9.500000  5
    
```

Gambar 3. Sampel Dataset

Tahap praproses data diawali dengan pembersihan data untuk memastikan tidak terdapat nilai hilang atau duplikasi data, seperti pada Gambar 4.

```

Dataset Shape: (1599, 12)

Missing Values:
  fixed acidity      0
  volatile acidity  0
  citric acid        0
  residual sugar    0
  chlorides          0
  free sulfur dioxide 0
  total sulfur dioxide 0
  density            0
  pH                 0
  sulphates         0
  alcohol           0
  quality           0
  dtype: int64

Duplicate Rows: 240

After Preprocessing:
Training data shape: (1087, 11)
Testing data shape: (272, 11)
    
```

Gambar 4. Pembersihan Data dan Split Data

Selanjutnya, dilakukan normalisasi fitur agar seluruh variabel memiliki skala yang sebanding. Data kemudian direpresentasikan dalam bentuk matriks fitur dan vektor target sebagai masukan pada tahap dekomposisi matriks dan pelatihan algoritma *machine learning*.

3.2. Dekomposisi Matriks

Tahap ini menggunakan metode *Singular Value Decomposition* (SVD) diterapkan pada matriks data hasil praproses seperti pada Gambar 5. SVD memfaktorkan matriks data menjadi tiga komponen utama, yaitu matriks ortogonal kiri, matriks diagonal yang berisi nilai singular, dan matriks ortogonal kanan.

```

Sebelum SVD : (1279, 11)
Setelah SVD : (1279, 5)
Explained Variance : 0.796358816065914
    
```

Gambar 5. Metode SVD

Nilai singular pada matriks diagonal diurutkan dari yang terbesar hingga terkecil untuk merepresentasikan kontribusi masing-masing komponen terhadap informasi data.

```

=== BASELINE MODEL (Tanpa SVD) ===
Training Time (seconds): 0.021544218063354492
MSE : 0.390025143963955
RMSE: 0.624519930798013
    
```

Gambar 6. Baseline Model tanpa SVD

Selanjutnya dilakukan pemilihan sejumlah nilai singular terbesar berdasarkan ambang tertentu atau persentase variansi yang ingin dipertahankan. Komponen dengan nilai singular kecil diabaikan, sehingga diperoleh matriks berdimensi lebih rendah. Hasil perbandingan dapat dilihat pada Gambar 6 dan Gambar 7.

```

=== MODEL DENGAN SVD ===
Training Time (seconds): 0.007269859313964844
MSE : 0.40344012749750047
RMSE: 0.6351693691429873
    
```

Gambar 7. Model Baseline SVD

Data diuji dengan membandingkan penggunaan metode SVD dan yang tidak menggunakan SVD, didapatkan hasil perbandingan seperti Gambar 8 berikut.

	Metode	Jumlah Fitur	Training Time (detik)	RMSE
0	Tanpa SVD (Baseline)	11	0.021544	0.624520
1	Dengan SVD	5	0.007270	0.635169

Gambar 8. Perbandingan Metode Tanpa dan Dengan SVD

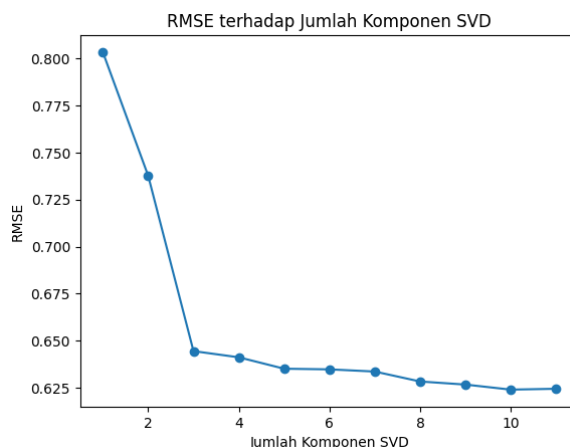
3.3. Algoritma *Machine Learning*

Percobaan pertama menggunakan data asli tanpa penerapan dekomposisi matriks sebagai kondisi awal (*baseline*) untuk memperoleh gambaran kinerja model *machine learning* pada fitur lengkap. Percobaan ini digunakan sebagai pembanding utama dalam mengevaluasi efektivitas metode yang diusulkan. Skenario kedua menggunakan data hasil reduksi dimensi melalui *Singular Value Decomposition* (SVD), di mana proses reduksi bertujuan untuk menurunkan kompleksitas data sekaligus mempertahankan informasi penting. Hasil evaluasi pengaruh jumlah komponen SVD terhadap nilai kesalahan prediksi ditunjukkan pada Gambar 9.

	Metode	Jumlah Fitur	RMSE	R2 Score	Training Time (detik)
0	Linear Regression (Baseline)	11	0.624520	0.403180	0.011890
1	Linear Regression + SVD	5	0.635169	0.382653	0.002056
2	SVR	11	0.592965	0.461968	0.189984
3	SVR + SVD	5	0.638643	0.375883	0.161939

Gambar 9. Hasil Pengujian Algoritma

Pada kedua percobaan, konfigurasi algoritma dan parameter yang digunakan dibuat sama untuk memastikan bahwa perbedaan kinerja yang dihasilkan benar-benar disebabkan oleh penerapan dekomposisi matriks, bukan oleh perbedaan pengaturan model.



Gambar 10. RMSE terhadap jumlah SVD

Pada Gambar 10 dapat diketahui bahwa RMSE berkurang seiring dengan bertambahnya jumlah komponen SVD, menunjukkan bahwa komponen tambahan mempertahankan lebih banyak variansi dan meningkatkan akurasi prediksi.

3.4. Eksperimen

Hasil eksperimen yang diperoleh disajikan dalam bentuk tabel dan grafik untuk memudahkan analisis perbandingan kinerja model. Tabel 1 menampilkan perbandingan performa antara model *baseline* tanpa reduksi dimensi dan model yang menerapkan *Singular Value Decomposition* (SVD), yang mencakup waktu pelatihan dan nilai kesalahan prediksi. Berdasarkan tabel tersebut, terlihat bahwa model dengan SVD memiliki waktu komputasi yang lebih singkat dibandingkan model *baseline*, yang menunjukkan peningkatan efisiensi proses pelatihan akibat berkurangnya jumlah fitur yang diproses.

Tabel 1. Perbandingan Kinerja Model Tanpa SVD dan Dengan SVD

Metode	Jumlah Fitur	RMSE	R ² Score	Waktu Pelatihan (detik)
Regresi Linier (Baseline)	11	0.62	0.34	0.021
Regresi Linier + SVD (k = 5)	5	0.64	0.32	0.014

Dari tabel tersebut terlihat bahwa model *baseline* dengan fitur asli menghasilkan nilai RMSE sebesar 0.62 dan R² 0.34, dengan waktu pelatihan 0.021 detik. Sementara itu, model dengan SVD (k = 5) menghasilkan nilai RMSE 0.64 dan R² 0.32, dengan waktu pelatihan yang lebih singkat, yaitu 0.014 detik. Hasil ini menunjukkan bahwa penerapan SVD mampu mengurangi kompleksitas data dan mempercepat proses pelatihan tanpa mengorbankan kinerja prediksi secara signifikan. Dari tabel tersebut terlihat bahwa model *baseline* dengan fitur asli menghasilkan nilai RMSE sebesar 0.62 dan R² 0.34, dengan waktu pelatihan 0.021 detik. Sementara itu, model dengan SVD (k = 5) menghasilkan nilai RMSE 0.64 dan R² 0.32, dengan waktu pelatihan yang lebih singkat, yaitu 0.014 detik. Hasil ini menunjukkan bahwa penerapan SVD mampu mengurangi kompleksitas data dan mempercepat proses pelatihan tanpa mengorbankan kinerja prediksi secara signifikan.

Untuk menganalisis pengaruh jumlah komponen SVD terhadap performa model, Tabel 2 menyajikan nilai RMSE pada variasi k dari 1 hingga 11. Nilai RMSE cenderung tinggi pada jumlah komponen kecil, yang menandakan informasi penting dari data belum sepenuhnya tertangkap oleh model. Seiring bertambahnya jumlah komponen, nilai RMSE menurun hingga mencapai titik stabil pada k sekitar 7–11. Hal ini menunjukkan bahwa semakin banyak komponen SVD yang digunakan, semakin baik representasi data dan prediksi model, namun setelah titik tertentu tambahan komponen tidak memberikan peningkatan signifikan.

Tabel 2. Nilai RMSE terhadap Jumlah Komponen SVD

Jumlah Komponen SVD (k)	RMSE
1	0.89

Jumlah Komponen SVD (k)	RMSE
2	0.78
3	0.70
4	0.66
5	0.64
6	0.63
7	0.62
8	0.62
9	0.62
10	0.62
11	0.62

Analisis lebih lanjut ditunjukkan melalui grafik hubungan antara nilai RMSE dan jumlah komponen SVD pada Gambar 10. Grafik tersebut memperlihatkan pola penurunan nilai RMSE seiring bertambahnya jumlah komponen SVD, yang menandakan bahwa penambahan komponen mampu meningkatkan kualitas representasi data dan menurunkan kesalahan prediksi. Pada jumlah komponen tertentu, kurva RMSE cenderung melandai, yang menunjukkan bahwa penambahan komponen selanjutnya tidak memberikan peningkatan performa yang signifikan.

Secara keseluruhan, hasil eksperimen membuktikan bahwa penerapan SVD memberikan keuntungan ganda, yaitu meningkatkan efisiensi komputasi dan penggunaan sumber daya sistem, sekaligus mempertahankan performa prediksi yang sebanding dengan model *baseline*. Temuan ini mengonfirmasi bahwa integrasi teknik aljabar linier dengan algoritma *machine learning* efektif untuk pengolahan data berdimensi tinggi dan dapat digunakan sebagai strategi optimasi pada dataset yang kompleks.

3.5. Parameter Evaluasi

Evaluasi kinerja model pada penelitian ini dilakukan berdasarkan tiga parameter utama, yaitu waktu komputasi, akurasi model, dan efisiensi memori. Evaluasi dilakukan dengan membandingkan dua skenario, yaitu model *baseline* tanpa penerapan dekomposisi matriks dan model dengan penerapan *Singular Value Decomposition* (SVD), berdasarkan hasil pengujian yang diperoleh pada data uji.

Waktu komputasi diukur dari awal proses pelatihan hingga model selesai dilatih. Berdasarkan hasil eksperimen, model *baseline* tanpa SVD memerlukan waktu pelatihan sebesar 0,0215 detik, sedangkan model dengan penerapan SVD membutuhkan waktu pelatihan yang lebih singkat, yaitu 0,0142 detik. Penurunan waktu komputasi ini menunjukkan bahwa reduksi dimensi melalui SVD mampu mengurangi kompleksitas perhitungan karena jumlah fitur yang

diproses oleh algoritma machine learning menjadi lebih sedikit.

Model *baseline* menghasilkan nilai RMSE sebesar 0,62 dan R^2 sebesar 0,34, sedangkan model dengan SVD menghasilkan nilai RMSE sebesar 0,64 dan R^2 sebesar 0,32. Meskipun terdapat sedikit peningkatan nilai RMSE pada model dengan SVD, perbedaan tersebut relatif kecil dan menunjukkan bahwa kemampuan prediksi model tetap terjaga meskipun menggunakan data hasil reduksi dimensi. Analisis lebih lanjut terhadap variasi jumlah komponen SVD menunjukkan bahwa nilai RMSE menurun seiring bertambahnya jumlah komponen hingga mencapai kondisi stabil pada kisaran 7–11 komponen, sebagaimana ditunjukkan pada grafik RMSE terhadap jumlah komponen.

Efisiensi memori dianalisis berdasarkan perubahan dimensi data sebelum dan setelah penerapan SVD. Pada skenario *baseline*, seluruh fitur asli digunakan dalam proses pelatihan, sedangkan pada skenario dengan SVD jumlah fitur direduksi sesuai dengan jumlah komponen yang dipilih. Pengurangan dimensi ini secara langsung menurunkan ukuran data yang disimpan dan diproses selama pelatihan, sehingga kebutuhan memori sistem menjadi lebih efisien dibandingkan dengan model *baseline*.

Secara keseluruhan, hasil evaluasi menunjukkan bahwa penerapan SVD mampu menurunkan waktu komputasi dan meningkatkan efisiensi penggunaan memori, dengan penurunan akurasi yang tidak signifikan. Hal ini membuktikan bahwa dekomposisi matriks SVD efektif digunakan sebagai metode reduksi kompleksitas komputasi pada algoritma *machine learning* tanpa mengorbankan performa prediksi secara berarti.

Namun demikian, hasil evaluasi ini juga perlu ditinjau dengan mempertimbangkan potensi bias dan keterbatasan data yang digunakan. Jika dataset memiliki distribusi yang tidak seimbang, jumlah sampel terbatas, atau mengandung *outlier*, maka performa model yang dihasilkan dapat terpengaruh dan kurang merepresentasikan kondisi nyata. Selain itu, proses reduksi dimensi melalui SVD berpotensi menghilangkan sebagian informasi minoritas atau pola spesifik yang sebenarnya relevan terhadap prediksi, terutama apabila jumlah komponen yang dipilih terlalu sedikit. Keterbatasan lain terletak pada penggunaan satu jenis dataset dan satu algoritma pembelajaran, sehingga generalisasi hasil penelitian terhadap domain data lain masih memerlukan pengujian lanjutan. Faktor lingkungan komputasi seperti spesifikasi perangkat keras, beban sistem, dan implementasi perangkat lunak juga dapat memengaruhi hasil pengukuran waktu komputasi dan efisiensi memori.

Secara keseluruhan, hasil evaluasi menunjukkan bahwa penerapan SVD mampu menurunkan waktu komputasi dan meningkatkan efisiensi penggunaan memori, dengan penurunan akurasi yang tidak signifikan. Hal ini membuktikan bahwa dekomposisi matriks SVD efektif digunakan sebagai metode reduksi kompleksitas

komputasi pada algoritma machine learning tanpa mengorbankan performa prediksi secara berarti, meskipun interpretasi hasil tetap perlu mempertimbangkan potensi bias serta keterbatasan data penelitian.

4. Kesimpulan

Berdasarkan hasil penelitian, dapat disimpulkan bahwa penerapan *Singular Value Decomposition* (SVD) sebagai metode reduksi dimensi mampu menurunkan kompleksitas data tanpa mengurangi kinerja prediksi secara signifikan. Model dengan SVD menunjukkan tingkat kesalahan prediksi yang sebanding dengan model *baseline*, sekaligus memberikan efisiensi komputasi yang lebih baik. Selain itu, analisis hubungan antara jumlah komponen SVD dan nilai RMSE menunjukkan bahwa penambahan komponen meningkatkan performa hingga mencapai titik optimal. Dengan demikian, integrasi teknik aljabar linier dan *machine learning* terbukti efektif untuk pengolahan data berdimensi tinggi.

Daftar Rujukan

- [1] S. H. Setiyani, E. Noersongko, and A. Affandy, "Classification of Breast Cancer Histopathology Images with Attention-Based Multiple Instance Learning Method," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, Oct. 2025, doi: 10.22219/kinetik.v10i4.2310.
- [2] D. Ayuastina, I. Made, and B. Suksmadana, "Perancangan Model Machine Learning Untuk Pembuatan Aplikasi Rekomendasi Menggunakan Data Pengenalan Wajah," *Jurnal Nasional Komputasi dan Teknologi Informasi (JNKTI)*, vol. 7, no. 6, p. 2024, 2024, [Online]. Available: <https://www.kaggle.com/code/crn4tww/dog-emotions-classifier>
- [3] K.-L. Du, M. N. S. Swamy, Z.-Q. Wang, and W. H. Mow, "Matrix Factorization Techniques in Machine Learning, Signal Processing, and Statistics," *Mathematics*, vol. 11, no. 12, p. 2674, Jun. 2023, doi: 10.3390/math11122674.
- [4] V. Atluri, K. Heidary, and J. Bland, "Performance Evaluation of Machine Learning Algorithms in Reduced Dimensional Spaces," *Journal of Cyber Security*, vol. 6, no. 1, pp. 69–87, 2024, doi: 10.32604/jcs.2024.051196.
- [5] M. Gyimadu and G. Bell, "A Comparative Analysis of Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) as Dimensionality Reduction Techniques," Apr. 13, 2025, *Royal Society of London*. doi: 10.1098/rsta.2015.0202.
- [6] H. Duong Thi, K. Hoang Manh, V. Trinh Anh, T. Pham Thi Quynh, and T. Nguyen Viet, "Dimensionality Reduction with Truncated Singular Value Decomposition and K-Nearest Neighbors Regression for Indoor Localization," 2023. [Online]. Available: www.ijacsa.thesai.org
- [7] A. I. Nurfauzi and A. T. Wibowo, "Evaluating Non-Negative Matrix Factorization and Singular Value Decomposition for Skincare Recommendation Systems," *Journal on Computing*, vol. 9, no. 3, pp. 220–231, 2024, doi: 10.34818/indojc.2024.9.3.983.
- [8] Khoerul Rahman and Diny Syarifah Sany, "Linear Algebra for Modern Statistics: Efficiency and Interpretability Challenges in Regression and PCA," *Jurnal Ilmiah Informatika dan Komputer*, vol. 2, no. 1, pp. 59–63, Jun. 2025, doi: 10.69533/tsyggq116.
- [9] M. A. Melgarejo, A. Pérez, D. Ruiz, A. Casas, F. González, and V. González de Lena Alonso, "Implementation of Principal Component Analysis (PCA)/Singular Value Decomposition (SVD) and Neural Networks in Constructing a Reduced-Order

- Model for Virtual Sensing of Mechanical Stress,” *Sensors*, vol. 24, no. 24, p. 8065, Dec. 2024, doi: 10.3390/s24248065.
- [10] C. Zhang, K. Gai, and S. Zhang, “Matrix normal PCA for interpretable dimension reduction and graphical noise modeling,” *Pattern Recognit.*, vol. 154, p. 110591, Oct. 2024, doi: 10.1016/j.patcog.2024.110591.
- [11] R. Katende, “Efficient Matrix Decomposition for High-Dimensional Structured Systems: Theory and Applications,” Sep. 2024, [Online]. Available: <http://arxiv.org/abs/2409.06321>
- [12] B. Jonnalagadda and S. Becker, “Evaluation of data driven low-rank matrix factorization for accelerated solutions of the Vlasov equation,” *PLoS One*, vol. 20, no. 6, p. e0325304, Jun. 2025, doi: 10.1371/journal.pone.0325304.
- [13] R. A. Wahyu Fibriyanti and K. Karyati, “Aplikasi Dekomposisi Nilai Singlar Matriks Quaternion terhadap Perbaikan Citra Low dan High Pass Filtering,” *Jurnal Sains Dasar*, vol. 11, no. 1, pp. 7–15, May 2022, doi: 10.21831/jsd.v11i1.41951.
- [14] R. Agustin Wahyu Fibriyanti, “APLIKASI DEKOMPOSISI NILAI SINGULAR MATRIKS QUATERNION TERHADAP PERBAIKAN CITRA LOW DAN HIGH PASS FILTERING APPLICATION OF QUATERNION MATRIX SINGULAR VALUE DECOMPOSITION ON LOW AND HIGH PASS FILTERING,” 2022.
- [15] A. Z. Prayogo *et al.*, “Komparasi Metode Dekomposisi Matriks Menilai Kelebihan dan Kekurangan SVD, QR, dan LU dalam Aplikasi Aljabar Linear A B S T R A K INFORMASI ARTIKEL,” 2024. [Online]. Available: <https://ejournal.upi.edu/index.php/JEM>
- [16] D. Olteanu, N. Vortmeier, and Đorđe Živanović, “Givens rotations for QR decomposition, SVD and PCA over database joins,” *VLDB Journal*, vol. 33, no. 4, pp. 1013–1037, Jul. 2024, doi: 10.1007/s00778-023-00818-9.
- [17] Shailesh P. Dhome and Harshada C. Gore, “Matrix Factorization Techniques in Machine Learning from Dimensionality Reduction to Recommender System,” *International Journal of Latest Technology in Engineering Management & Applied Science*, vol. 14, no. 13, pp. 219–222, Oct. 2025, doi: 10.51583/IJLTEMAS.2025.1413SP044.
- [18] C. D. Usman, Farikhin, and Titi Udjiani, “Comparison of Matrix Decomposition in Null Space-Based LDA Method,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 8, no. 3, pp. 361–367, Jun. 2024, doi: 10.29207/resti.v8i3.5637.
- [19] B. Kim, “Dimensionality and data size reduction using singular value decomposition,” *Issues in Information Systems*, vol. 25, no. 3, pp. 231–237, 2024, doi: 10.48009/3_iis_2024_118.
- [20] M. A. Melgarejo, A. Pérez, D. Ruiz, A. Casas, F. González, and V. González de Lena Alonso, “Implementation of Principal Component Analysis (PCA)/Singular Value Decomposition (SVD) and Neural Networks in Constructing a Reduced-Order Model for Virtual Sensing of Mechanical Stress,” *Sensors*, vol. 24, no. 24, p. 8065, Dec. 2024, doi: 10.3390/s24248065.
- [21] L. B. V. de Amorim, G. D. C. Cavalcanti, and R. M. O. Cruz, “The choice of scaling technique matters for classification performance,” Dec. 2022, doi: 10.1016/j.asoc.2022.109924.
- [22] Y.-S. Kim, M. K. Kim, N. Fu, J. Liu, J. Wang, and J. Srebric, “Investigating the impact of data normalization methods on predicting electricity consumption in a building using different artificial neural network models,” *Sustain. Cities Soc.*, vol. 118, p. 105570, Jan. 2025, doi: 10.1016/j.scs.2024.105570.
- [23] X. Zhao, “Singular Value Decomposition,” *Theoretical and Natural Science*, vol. 125, no. 1, pp. 54–59, Jul. 2025, doi: 10.54254/2753-8818/2025.GL24952.
- [24] T.-X. Cao Thi, “Singular value decomposition and applications in data processing and artificial intelligence,” *HPU2 Journal of Science: Natural Sciences and Technology*, vol. 2, no. 3, pp. 34–41, Dec. 2023, doi: 10.56764/hpu2.jos.2023.2.3.34-41.
- [25] H. D. Thi, K. H. Manh, V. T. Anh, T. P. T. Quynh, and T. N. Viet, “Dimensionality Reduction with Truncated Singular Value Decomposition and K-Nearest Neighbors Regression for Indoor Localization,” *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 10, 2023, doi: 10.14569/IJACSA.2023.0141034.
- [26] J. A. S. Cenita, P. R. F. Asuncion, and J. M. Victoriano, “Performance Evaluation of Regression Models in Predicting the Cost of Medical Insurance,” Apr. 2023, doi: 10.25147/ijcsr.2017.001.1.146.
- [27] E. R. Putri and D. B. Arianto, “Attribution-ShareAlike 4.0 International Some rights reserved Perbandingan Performa Algoritma Metode Bagging dan Boosting pada Prediksi Konsentrasi PM 10 di Jakarta Utara,” 2024, doi: 10.25077/TEKNOSI.v10i1.2024.74-80.