Seminar Nasional Sains Data 2024 (SENADA 2024)
UPN "Veteran" Jawa Timur
E-ISSN 2808-5841
P-ISSN 2808-7283

Model Keputusan *Repeat Order* pada Vendor Menggunakan *Machine Learning Classification*

Ni'mah Tsabitah[1], Budi Aribowo[2]
[1, 2]*Teknik Industri, Universitas Al-Azhar Indonesia*
[1]tsabitahnimah28@gmail.com
Corresponding author email: budiaribowo@gmail.com

Abstract: *PT XYZ faces challenges in vendor selection that can impact the company's performance and operations. The main issues in the procurement process are product non-conformity and delivery delays. This study identifies vendor selection criteria using the Vendor Performance Index (VPI) and tests the normality of the data used. Additionally, Principal Component Analysis (PCA) is applied to reduce data dimensions and identify the main components of the VPI. A vendor classification model is developed using five machine learning methods: Logistic Regression, Support Vector Machine (SVM), Gaussian Naive Bayes, Random Forest Classifier, and K-nearest Neighbors. Data collected from questionnaires indicates a normal distribution through the Shapiro-Wilk and Kolmogorov-Smirnov tests, as the p-value > alpha, meaning the null hypothesis is accepted and the data is normally distributed. A parametric test then reveals a t-value of 3.991 and a p-value of 0.000855, indicating a significant difference between the "High Repeat Order" and "Low Repeat Order" groups. Subsequent PCA analysis reveals that two main components explain more than 90% of the data variance. Among the five tested machine learning models, the Gaussian Naive Bayes model demonstrates the best performance with an accuracy of 0.83 and an AUC of 0.78. This model aids the company in making more accurate decisions regarding repeat order vendors, enhancing efficiency and reducing business risks.*
Keywords: *machine learning, principal component analysis, vendor performance index*

Abstrak: PT XYZ menghadapi tantangan dalam pemilihan vendor yang dapat memengaruhi kinerja dan operasional perusahaan. Masalah ketidaksesuaian barang dan keterlambatan pengiriman merupakan isu utama dalam proses pengadaan barang. Penelitian ini mengidentifikasi kriteria pemilihan vendor menggunakan *Vendor Performance Index* (VPI) dan menguji normalitas data yang digunakan. Selain itu, *Principal Component Analysis* (PCA) diterapkan untuk mereduksi dimensi data dan mengidentifikasi komponen utama dari VPI. Model klasifikasi vendor dikembangkan menggunakan lima metode *Machine learning* yaitu *Logistic Regression*, *Support Vector Machine* (SVM), *Gaussian Naive Bayes*, *Random Forest* Classifier, dan *K-nearest Neighbors*. Berdasarkan pengumpulan data kuesioner menunjukkan distribusi normal melalui uji *Shapiro-Wilk* dan *Kolmogorov-Smirnov* karena p-*value* > *alpha*, yang berarti bahwa hipotesis nol diterima atau data terdistribusi normal. Kemudian dilakukan uji parametrik menghasilkan nilai t sebesar 3.991 dan p sebesar 0.000855, menandakan perbedaan signifikan antara kelompok "*High Repeat Order*" dan "*Low Repeat Order*". Kemudian pada analisis PCA mengungkapkan bahwa dua komponen utama menjelaskan lebih dari 90% varians data. Dari lima model *Machine Learning* yang diuji, *Gaussian Naive Bayes* menunjukkan performa terbaik dengan akurasi 0.83 dan AUC 0.78. Model ini membantu perusahaan membuat keputusan yang lebih tepat mengenai vendor *Repeat Order*, meningkatkan efisiensi dan mengurangi risiko bisnis.
Kata kunci: *machine learning, principal component analysis, vendor performance index*

I. PENDAHULUAN

PT XYZ merupakan perusahaan yang berfokus pada sumber daya strategis teknologi informasi dan telekomunikasi. Salah satu departemen pada perusahaan ini yaitu departemen *procurement* yang bertanggung jawab untuk melakukan pengadaan barang dan jasa yang diperlukan oleh perusahaan. Ini meliputi identifikasi kebutuhan, pencarian vendor, negosiasi kontrak, dan pemilihan vendor yang sesuai dengan kebutuhan perusahaan. Dalam konteks dunia bisnis saat ini, PT XYZ harus beroperasi dalam lingkungan yang dikenal dengan istilah VUCA. VUCA adalah singkatan dari *Volatility, Uncertainty, Complexity, Ambiguity*. *Volatility* adalah kecepatan dan besarnya perubahan yang tak terduga dalam industri atau pasar, seperti perubahan peraturan, harga komoditas, rantai pasokan, atau pandemi global. Pemimpin harus siap menghadapi tantangan ini. *Uncertainty* berarti kurangnya prediktabilitas, sehingga pemimpin perlu strategi tangkas untuk merespons lingkungan bisnis yang tidak menentu. *Complexity* mencakup dinamika organisasi dan lingkungan saling bergantung yang mempersulit pemahaman dampak keputusan, diperparah oleh teknologi, digitalisasi, dan globalisasi. *Ambiguity* berarti

ketidakjelasan arah karena situasi baru yang asing, seperti pandemi global, yang memerlukan adaptasi cepat di luar zona nyaman [1]

Berbagai penelitian telah mengeksplorasi penerapan metode *machine learning* untuk berbagai keperluan analisis dan prediksi dalam berbagai bidang. Model klasifikasi K-*Nearest Neighbor, Naive Bayes, Support Vector Machine, Boosting,* dan *Decision Tree Random Forest* digunakan untuk menentukan penerima Program Indonesia Pintar, namun menghadapi masalah akurasi dan evaluasi sistem [2]. Dalam penelitian Adrian membandingkan *Random Forest* dan SVM untuk analisis sentimen terhadap PSBB, namun terbatas oleh kurangnya data penelitian [3]. Penelitian Astoni dan Haris mengaplikasikan PCA dan *Random Forest* untuk deteksi kecurangan kartu kredit, tetapi tidak menguji performa model terhadap metode lain [4]. Penelitian-penelitian ini menunjukkan berbagai tantangan dan kontribusi dalam penerapan *machine learning*, serta memberikan arahan untuk penelitian masa depan guna meningkatkan akurasi dan efektivitas metode yang digunakan.

K-Nearest Neighbor merupakan salah satu algoritma klasifikasi terbaik, yang cara kerjanya menentukan jarat paling dekat [5], KNN mempunyai kelebihan yaitu mudah untuk dipelajari, algoritma yang sederhana, dan kinerja yang baik untuk data besar [6]. *Naive Bayes* terkenal sebagai algoritma yang mempunyai kinerja baik [7] dan merupakan algoritma yang mampu memperkirakan variable bersifat bebas [8]. Algoritma *Random Forest* didesain oleh J. Ross Quinlan, dinamakan *Random Forest* karena merupakan keturunan dari pendekatan ID3 untuk membangun pohon keputusan. *Random Forest* merupakan algoritma yang cocok digunakan untuk masalah klasifikasi pada *machine learning* dan data mining [9]. Metode klasifikasi lain yang dipakai adalah *Support Vector Machine* karena diketahui algoritma ini sangat baik untuk dilakukan pada klasifikasi teks dan tidak memerlukan kemampuan komputasi yang berat, metode ini sangat mudah di implementasi pada perangkat yang tidak terlalu mumpuni untuk melakukan pembelajaran mesin dan sangat sering menjadi benchmark untuk metode – metode pembelaran mesin lainnya [10]. *Logistic Regression* merupakan klasifikasi linier yang telah terbukti menghasilkan klasifikasi yang powerful dengan statistik probabilitas dan menangani masalah klasifikasi multi kelas [11].

Dengan menggunakan model klasifikasi berbasis *Machine Learning* yang dikembangkan, perusahaan dapat membuat keputusan yang lebih terinformasi dan tepat mengenai kelayakan vendor untuk menerima *Repeat Order*. Model ini menganalisis berbagai faktor, termasuk kualitas produk, keandalan pengiriman, dan umpan balik dari klien sebelumnya, untuk memprediksi apakah suatu vendor layak untuk dipilih kembali. Dengan pendekatan ini, PT XYZ dapat meningkatkan efisiensi dan efektivitas dalam proses pengambilan keputusan, serta mengurangi risiko terhadap praktik bisnis yang merugikan. Hal ini memastikan bahwa perusahaan dapat menjaga kinerja operasional yang optimal dan memenuhi kebutuhan gudang dengan tepat waktu dan kualitas yang baik.

II. METODE PENELTIAN

Metodologi penelitian terdiri dari setiap kegiatan yang dilakukan dalam suatu penelitian yang terdiri dari pengumpulan data dan pengolahan data serta metode apa saja yang digunakan pada penelitian ini.

2.1. Pengumpulan Data

Data yang digunakan pada penelitian ini merupakan data primer. Pengumpulan data pada peneltian ini dilakukan secara langsung pada PT XYZ. Data diperoleh berdasarkan hasil dari kuesioner. Pada penelitian ini menggunakan metode *Vendor Performance Index* (VPI) untuk menentukan kriteria dan subkriteria yang digunakan dalam penilaian kinerja pemasok yang sesuai dengan perusahaan. Metode *Vendor Performance Index* (VPI) yang digunakan pada penelitian ini berkerangka QDCFR (*Quality, Delivery, Cost, Flexibility, dan Responsiveness*). Salah satu metode penilaian kinerja pemasok diperkenalkan oleh YP fun dan Js Hung (1997) dalam jurnal yang berjudul "*A new measure for pemasok performance evaluation*".

2.2. Pengolahan Data

Apabila semua data yang dikumpulkan sudah cukup, maka tahap selanjutnya ialah melakukan pengolahan data.

2.2.1. Uji Normalitas

Salah satu syarat utama dalam analisis statistika parametrik adalah terpenuhinya kenormalan data. Untuk mengetahui kepastian sebaran data yang diperoleh, haruslah dilakukan uji normalitas terhadap data yang bersangkutan. Pengujian untuk membuktikan normal atau tidaknya suatu data dapat dilakukan dengan menggunakan analisis *Kolmogorov – Smirnov* dan *Shapiro – Wilk* [12]. Selanjutnya, tingkat signifikansi (*alpha*) ditetapkan, misalnya 0.01 atau 0.05. Hasil ini dibandingkan dengan *alpha*: jika p-*value* kurang dari *alpha*, hipotesis nol (H0) ditolak, yang berarti data tidak terdistribusi normal; jika p-*value* lebih besar atau sama dengan *alpha*, H0 diterima, yang berarti data terdistribusi normal [13].

2.2.2. Uji Statistika

Uji parametrik berfungsi untuk menentukan apakah terdapat perbedaan yang signifikan antara rata-rata dua kelompok proses. Hasil dari uji t-independen ini meliputi nilai statistik t, p-*value*, dan derajat kebebasan (df). Berdasarkan p-*value* yang diperoleh, hasil uji dianalisis untuk menentukan apakah ada perbedaan yang signifikan secara statistik antara dua set data tersebut. Jika p-*value* kurang dari 0.05, maka perbedaan dianggap signifikan secara statistik. Proses ini berakhir dengan interpretasi hasil, menentukan apakah perbedaan antara dua set data tersebut signifikan atau tidak, menandai akhir dari uji t-independen. salah satu syarat utama dalam analisis statistika parametrik adalah terpenuhinya kenormalan data.

2.2.3. *Principal Component Analysis* (PCA)

Principal Component Analysis (PCA) adalah salah satu metode yang dapat dilakukan untuk mengurangi dimensi fitur [14]. Metode PCA sangat bergun adigunakan jika data yang ada memiliki jumlah variable yang besar dan memiliki korelasi antar variabelnya. Perhitungan dari *Principal Component Analysis* didasarkan pada perhitungan nilai *eigen* dan *vector eigen* yang menyatakan penyebaran data dari suatu dataset [15]. Dengan menggunakan PCA, variabel yang tadinya sebanyak n variabel akan diseleksi menjadi k variabel baru yang disebut *principal component*, dengan jumlah k lebih sedikit dari n. Dengan hanya menggunakan k principal component akan menghasilkan nilai yang sama dengan menggunakan n variabel. Variabel hasil dari seleksi disebut *principal component* [16].

2.2.4. *Machine learning Classification*

Menurut Penelitian yang dilakukan oleh (Gholamy et al.,2018). Penggunaan 20-30% untuk data uji dan 70-80% untuk data latih menunjukkan hasil yang terbaik secara empirik. Pada Penelitian ini, penulis membagi tiap data set menjadi 70% atau 0,7 data *training* dan 30% atau 0,3 data *testing*. Data *training* digunakan untuk melatih model klasifikasi sedangkan data *testing* digunakan untuk menguji performa model dengan data yang diasumsikan belum pernah 'dilihat' atau dicerna oleh model. Pada tahap berikutnya, beberapa algoritma *machine learning* diterapkan secara paralel untuk membangun model klasifikasi vendor. Algoritma-algoritma tersebut meliputi *K-nearest Neighbors* (KNN), *Gaussian Naive Bayes* (Gaussian NB), *Logistic Regression*, *Random Forest*, dan *Support Vector Machine* (SVM).

2.2.5. *Confusion Matrix*

Setelah model-model tersebut dilatih, kinerja masing-masing model dievaluasi menggunakan *Confusion Matrix* yang membandingkan hasil prediksi dengan data sebenarnya. Akurasi dari setiap model dihitung untuk menentukan seberapa baik model tersebut dalam mengklasifikasikan data dengan benar. Dalam Python, *Confusion Matrix* dapat diperoleh dengan

menggunakan fungsi "*confusion_matrix*()" yang merupakan bagian dari pustaka "*sklearn*" [17]. Fungsi ini dapat diimpor ke dalam Python dengan menggunakan "*from sklearn.metrics import confusion_matrix*." Untuk mendapatkan *Confusion Matrix*, pengguna harus memberikan nilai aktual dan nilai prediksi ke fungsi tersebut.

2.2.6. *Receiver Operating Characteristic* (ROC)

Selain itu, *Area Under the Curve* (AUC) dari *Receiver Operating Characteristic* (ROC) *curve* juga dihitung untuk mengevaluasi kemampuan model dalam membedakan antara kelas-kelas. Nilai AUC secara teoritis berada di antara 0 dan 1. Nilai AUC memberikan gambaran tentang keseluruhan pengukuran atas kesesuaian dari model yang digunakan. Semakin besar area under *curve* maka semakin baik variabel yang diteliti dalam memprediksi kejadian [18]. Kurva ROC kemudian dibuat untuk setiap model untuk memvisualisasikan *trade-off* antara *true positive rate* dan *false positive rate*.

2.3. Analisis Hasil Komparasi dan Simulasi Model

Analisis data dilakukan untuk memperoleh informasi yang dapat dijadikan dasar dalam memperoleh kesimpulan pada penelitian ini. Dari penyajian hasil evaluasi dengan *Confusion Matrix* dan ROC *curve*, langkah berikutnya adalah menganalisis hasil komparasi dari berbagai model *machine learning* yang telah digunakan. Setelah model terbaik dipilih, langkah berikutnya adalah melakukan simulasi untuk memastikan bahwa model tersebut dapat berfungsi dengan baik dalam situasi nyata.

III. HASIL DAN PEMBAHASAN

3.1. *Purchase Order* 2022-2023

Tabel 4.1 menunjukkan jumlah *Purchase Order* (PO) yang diterima oleh 20 vendor selama periode 2022-2023 serta klasifikasi mereka berdasarkan frekuensi *Repeat Order*. Kolom jumlah *Purchase Order* menampilkan total PO yang diterima oleh setiap vendor selama periode 2022-2023. Ini memberikan gambaran tentang seberapa sering setiap vendor digunakan oleh perusahaan. Kolom Klasifikasi membagi vendor menjadi dua kelompok berdasarkan frekuensi *Repeat Order* yang diterima. "*High Repeat Order*" untuk vendor yang sering mendapatkan *Repeat Order* dengan jumlah ≥ 20, menunjukkan tingkat kepercayaan dan kepuasan yang tinggi dari perusahaan; dan "*Low Repeat Order*" untuk vendor yang jarang mendapatkan *Repeat Order* dengan jumlah < 20, yang mungkin menunjukkan adanya masalah terkait kualitas, pengiriman, atau faktor lain yang mempengaruhi kepercayaan perusahaan.

Tabel 1. Jumlah *Purchase Order* pada Vendor Tahun 2022-2023

No	Nama Vendor	Jumlah Purchase Order	Klasifikasi
1	Vendor A	90	*High Repeat Order*
2	Vendor B	80	*High Repeat Order*
3	Vendor C	52	*High Repeat Order*
4	Vendor D	37	*High Repeat Order*
5	Vendor E	37	*High Repeat Order*
6	Vendor F	36	*High Repeat Order*
7	Vendor G	33	*High Repeat Order*
8	Vendor H	32	*High Repeat Order*
9	Vendor I	24	*High Repeat Order*
10	Vendor J	20	*High Repeat Order*
11	Vendor K	19	*Low Repeat Order*

12	Vendor L	17	*Low Repeat Order*
13	Vendor M	16	*Low Repeat Order*
14	Vendor N	16	*Low Repeat Order*
15	Vendor O	15	*Low Repeat Order*
16	Vendor P	15	*Low Repeat Order*
17	Vendor Q	14	*Low Repeat Order*
18	Vendor R	12	*Low Repeat Order*
19	Vendor S	11	*Low Repeat Order*
20	Vendor T	10	*Low Repeat Order*

3.2. Perhitungan *Vendor Performance Index* Menggunakan *Vendor Scorecard*

Pada tabel 2 menunjukkan bobot (*weight*) yang diberikan kepada setiap kriteria dalam evaluasi kinerja vendor. Setiap kriteria memiliki nilai bobot tertentu yang mencerminkan tingkat kepentingannya dalam keseluruhan penilaian. Tabel bobot kriteria ini sangat penting dalam proses evaluasi kinerja vendor karena menentukan bagaimana setiap kriteria mempengaruhi skor akhir. *Quality* dan *Cost* diberi bobot tertinggi (masing-masing 0.25), yang menunjukkan bahwa kedua aspek ini paling mempengaruhi keputusan pemilihan vendor. *Delivery*, dengan bobot 0.20, juga merupakan faktor penting, sementara *Flexibility* dan *Responsiveness*, masing-masing dengan bobot 0.15, tetap signifikan meskipun sedikit kurang berpengaruh. Penetapan bobot ini membantu perusahaan dalam mengidentifikasi dan memilih vendor yang paling sesuai dengan prioritas dan kebutuhan mereka.

Tabel 2. *Score Vendor Scorecard*

Criteria Scores	***Weight***
Quality	0.25
Cost	0.25
Delivery	0.20
Flexibility	0.15
Responsiveness	0.15
Total Score	1.00

Pada tabel 3 merupakan hasil perhitungan masing-masing nilai dikalikan dengan bobot kriteria. Hasil perkalian ini dijumlahkan untuk mendapatkan skor total setiap vendor. Vendor dengan skor total tinggi, seperti Vendor F dengan skor 4.33 dan Vendor D dengan skor 4.10, menunjukkan performa yang sangat baik di hampir semua kriteria. Vendor dengan skor total rendah, seperti Vendor N dengan skor 2.38 dan Vendor I dengan skor 2.53, menunjukkan bahwa mereka mungkin perlu meningkatkan performa mereka di beberapa atau semua kriteria.

Tabel 3. Perhitungan Nilai Kriteria dengan Bobot

Vendor	***CRITERIA***					**Total *Score***
	Quality	***Cost***	***Delivery***	***Flexibility***	***Responsiveness***	
Vendor A	1.00	1.00	0.80	0.68	0.60	4.08
Vendor B	1.13	0.75	0.60	0.30	0.45	3.23
Vendor C	0.63	0.75	0.60	0.45	0.45	2.88
Vendor D	0.75	1.08	0.87	0.75	0.65	4.10
Vendor E	1.00	1.00	0.80	0.53	0.60	3.93

Vendor	*CRITERIA*					Total *Score*
	Quality	*Cost*	*Delivery*	*Flexibility*	*Responsiveness*	
Vendor F	1.13	1.08	0.87	0.60	0.65	4.33
Vendor G	0.88	1.00	0.80	0.53	0.60	3.80
Vendor H	0.75	1.08	0.87	0.60	0.65	3.95
Vendor I	0.75	0.58	0.47	0.38	0.35	2.53
Vendor J	1.00	0.92	0.73	0.60	0.55	3.80
Vendor K	0.63	0.67	0.53	0.30	0.40	2.53
Vendor L	0.88	0.58	0.47	0.30	0.35	2.58
Vendor M	1.00	0.75	0.60	0.53	0.45	3.33
Vendor N	0.88	0.50	0.40	0.30	0.30	2.38
Vendor O	0.50	0.83	0.67	0.53	0.50	3.03
Vendor P	0.50	0.75	0.60	0.53	0.45	2.83
Vendor Q	1.00	0.83	0.67	0.45	0.50	3.45
Vendor R	0.63	1.08	0.87	0.68	0.65	3.90
Vendor S	0.63	0.92	0.73	0.53	0.55	3.35
Vendor T	0.88	0.75	0.73	0.68	0.55	3.58

3.3. Pengolahan Data dengan Uji Normalitas

Pada normalitas menggunakan uji *Shapiro-Wilk* dan uji *Kolmogorov-Smirnov* dengan menggunakan pustaka scipy dalam bahasa pemrograman Python untuk menentukan apakah data yang diberikan terdistribusi normal. Data yang digunakan adalah jumlah *Purchase Order* pada vendor dalam waktu 2 tahun yaitu 2022-2023. Fungsi ini menghitung nilai statistik uji dan p-*value* untuk menguji hipotesis nol (H0) yang menyatakan bahwa data terdistribusi normal. Tingkat signifikansi (*alpha*) ditetapkan sebesar 0,01. Dari hasil uji *Shapiro-Wilk*, nilai statistik uji adalah 0,6727677877578735 dan p-*value* adalah 1,8483885197808664e-05. Dengan membandingkan p-*value* dengan *alpha*, jika p-*value* lebih kecil dari *alpha*, maka hipotesis nol ditolak. Jika p-*value* lebih besar atau sama dengan *alpha*, maka hipotesis nol tidak dapat ditolak. Dalam kasus ini, p-*value* jauh lebih besar dari *alpha*, yang berarti bahwa hipotesis nol diterima. Artinya, data terdistribusi normal (terima H0).

Tabel 4. Uji Normalitas (*Shapiro-Wilk*)

Nilai p-*value*	*Alpha*	**Statistik Uji**
1,848	0,01	0,672

Kemudian berdasarkan hasil uji *Kolmogorov-Smirnov* diperoleh statistik uji sebesar 0.214 dan nilai p-*value* sebesar 0.276. Nilai p-*value* yang lebih besar dari 0.01 ini menunjukkan bahwa kita gagal menolak hipotesis nol, yang menyatakan bahwa data mengikuti distribusi normal. Dengan kata lain, tidak ada bukti yang cukup untuk menyimpulkan bahwa data tersebut tidak berasal dari distribusi normal. Hal ini mengindikasikan bahwa data yang dianalisis cenderung mengikuti distribusi normal.

Tabel 5. Uji Normalitas (*Shapiro-Wilk*)

Nilai p-*value*	*Alpha*	**Statistik Uji**
0,276	0,01	0,214

3.4. Pengolahan Data dengan Uji Parametrik

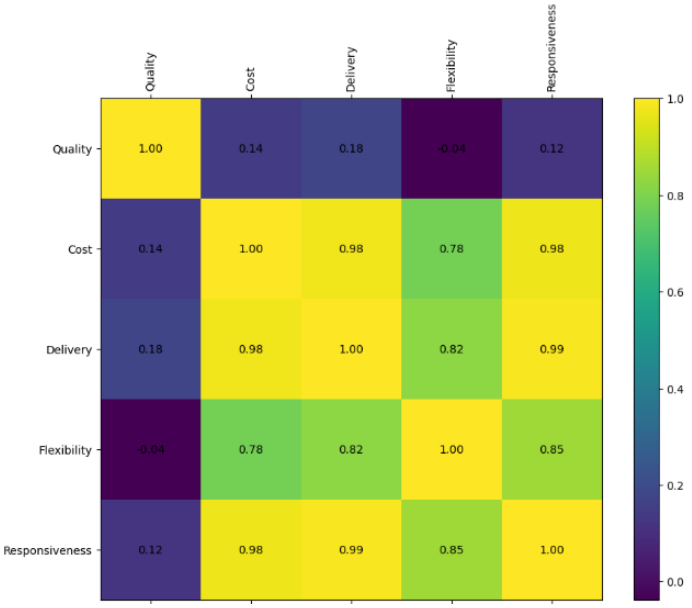
Selanjutnya melakukan uji t dua sampel independen pada dua variabel, yaitu "*High Repeat Order*" dan "*Low Repeat Order*". Uji t dua sampel independen adalah metode parametrik yang digunakan untuk membandingkan rata-rata dari dua grup yang tidak saling berhubungan guna menentukan apakah perbedaan rata-rata tersebut signifikan secara statistic. Dari analisis ini dapat disimpulkan bahwa terdapat cukup bukti untuk menolak hipotesis nol yang menyatakan bahwa rata-rata "*High Repeat Order*" dan "*Low Repeat Order*" sama, dan menyimpulkan bahwa rata-rata kedua grup tersebut berbeda secara signifikan.

Tabel 6. Uji Parametrik (Uji T)

Nilai p-*value*	*Alpha*	**Statistik Uji**
0,000855	0,05	3,991

3.5. Pengolahan Data *Correlation Matrix*

Matriks korelasi menunjukkan hubungan antara kriteria-kriteria dalam penilaian kinerja vendor, baik secara visual maupun numerik. Elemen diagonal dalam matriks ini semuanya bernilai 1, karena setiap kriteria berkorelasi sempurna dengan dirinya sendiri. Elemen off-diagonal menunjukkan korelasi antara dua kriteria yang berbeda, dengan nilai berkisar dari -1 hingga 1. Nilai positif mengindikasikan korelasi positif, di mana peningkatan satu kriteria terkait dengan peningkatan kriteria lainnya, sementara nilai negatif menunjukkan korelasi negatif, di mana peningkatan satu kriteria terkait dengan penurunan kriteria lainnya. Nilai yang mendekati 0 menunjukkan sedikit atau tidak ada hubungan linier antara kriteria-kriteria tersebut. Observasi kunci dari matriks korelasi ini mencakup beberapa poin penting. Pertama, kriteria "*Quality*" dan "*Flexibility*" memiliki korelasi negatif yang sangat kecil (-0.04), menunjukkan hubungan terbalik yang sangat lemah. Kriteria "*Cost*", "*Delivery*", dan "*Responsiveness*" menunjukkan korelasi positif yang sangat tinggi satu sama lain (mendekati 0.98 dan 0.99), menunjukkan bahwa mereka cenderung bergerak bersama. Jika satu kriteria meningkat, kriteria lainnya juga cenderung meningkat. Selain itu, terdapat korelasi positif yang kuat antara "*Cost*" dan "*Flexibility*" (0.78), menunjukkan bahwa biaya yang lebih tinggi terkait dengan fleksibilitas yang lebih besar. Kriteria "*Flexibility*" dan "*Responsiveness*" juga memiliki korelasi positif tinggi (0.85), menunjukkan bahwa fleksibilitas yang lebih besar terkait dengan responsivitas yang lebih besar. Sebelum melakukan PCA, mengamati matriks korelasi ini membantu kita memahami hubungan antar kriteria.



Gambar 1. *Correlation Matrix*

3.6. Pengolahan Data dengan *Metode Principal Component Analysis (PCA)*

PCA adalah teknik statistik yang digunakan untuk mereduksi dimensi data, sambil mempertahankan sebanyak mungkin variasi yang ada dalam data tersebut. Proses PCA dimulai dengan menghitung matriks kovarian dari dataset yang diberikan. Matriks kovarian ini menggambarkan bagaimana variabel-variabel dalam dataset berinteraksi satu sama lain. Setelah menghitung matriks kovarian, langkah berikutnya adalah menghitung nilai *eigen* (*eigenvalue*s) dan vektor *eigen* (*eigenvector*s) dari matriks kovarian. Nilai *eigen* menunjukkan seberapa besar variasi yang dijelaskan oleh masing-masing komponen utama, sementara vektor *eigen* menunjukkan arah dari komponen utama tersebut. Kemudian, proporsi variasi yang dijelaskan oleh setiap komponen utama dihitung dengan membagi setiap nilai *eigen* dengan jumlah total nilai *eigen*. Hasil perhitungannya menunjukkan bahwa komponen pertama menjelaskan sekitar 65.49% dari total variasi, komponen kedua menjelaskan sekitar 29.95%, dan seterusnya. Dengan menggunakan PCA, dimensi data dapat direduksi dengan memilih beberapa komponen utama pertama yang menjelaskan sebagian besar variasi dalam data. Hasil akhir dari kumulatif ini menunjukkan berapa persen varians data yang dapat dijelaskan jika kita menggunakan sejumlah komponen utama tertentu. Dari *output* yang diberikan, dapat dilihat bahwa sekitar 65% varians dapat dijelaskan oleh komponen utama pertama, sekitar 95% oleh dua komponen utama, dan seterusnya hingga 100% oleh keseluruhan komponen. Setelah melakukan *Principal Component Analysis* (PCA) dan mendapatkan hasil reduksi dimensi data, langkah selanjutnya adalah membuat matriks *loading*.

```
# standarisasi
#Construct the covarian matrix, calculate eigenvalues dan eigen vectors
import numpy as np
cov_mat = np.cov(dataset)
# From this covariance matrix, caluclate the Eigenvalues and the Eigenvectors
eigen_vals, eigen_vecs = np.linalg.eig(cov_mat)
# print the Eigenvalues
print("Raw Eigenvalues: \n", eigen_vals)
# the sum of the Eigenvalues
print("Percentage of Variance Explained by Each Component: \n", eigen_vals/sum(eigen_vals))
```

```
Raw Eigenvalues:
 [0.08336152 0.0380994  0.00514584 0.00054349 0.00012264]
Percentage of Variance Explained by Each Component:
 [0.65498253 0.29935206 0.04043151 0.0042703  0.0009636 ]
```

Tabel 7. Perhitungan *eigenvalues* dan *eigenvectors*

Criteria	***Quality***	***Cost***	***Delivery***	***Flexibility***	***Responsiveness***
Eigenvalues	0.08336152	0.0380994	0.00514584	0.00054349	0.00012264
Percentage of Variance	0.65498253	0.29935206	0.04043151	0.0042703	0.0009636
Cumulative Variance	0.65498253	0.95433459	0.9947661	0.999036	1

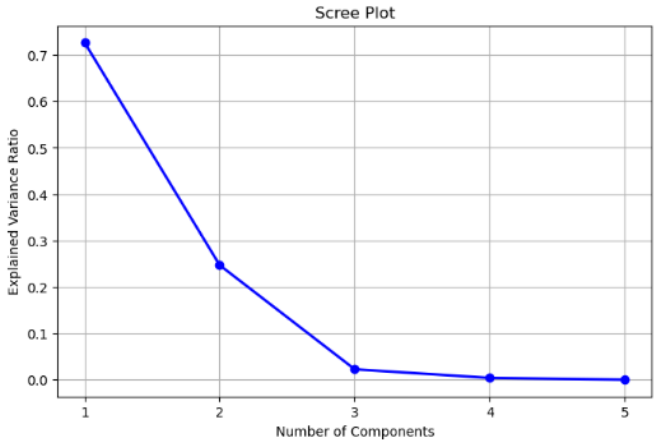
Pada tabel 8 merupakan *loading matrix* yang menunjukkan seberapa besar kontribusi setiap fitur asli terhadap komponen utama yang dihasilkan oleh PCA. Setiap baris dalam matriks ini mewakili fitur asli, dan setiap kolom (PC1 dan PC2) menunjukkan nilai *loading* untuk komponen utama yang sesuai. Sebagai contoh, fitur pertama memiliki *loading* sebesar -0.076447 pada PC1 dan 0.976563 pada PC2. *Loading* ini menunjukkan sejauh mana fitur tersebut berkontribusi pada masing-masing komponen utama. Nilai *loading* yang tinggi (baik positif maupun negatif) menunjukkan bahwa fitur tersebut memiliki kontribusi yang signifikan terhadap komponen utama tersebut.

Tabel 8. *Loading Matrix*

	PC1	**PC2**
Quality	-0.076447	0.976563
Cost	-0.504915	0.018970
Delivery	-0.512573	0.042582
Flexibility	-0.459966	-0.209259
Responsiveness	-0.514700	-0.019055

3.7. Pengolahan Data dengan *Scree Plot*

Scree Plot adalah grafik yang menunjukkan rasio variansi yang dijelaskan (*explained variance ratio*) oleh setiap komponen utama. Pada grafik dibawah, sumbu horizontal (x) merepresentasikan komponen utama (PC1 dan PC2), sementara sumbu vertikal (y) menunjukkan rasio variansi yang dijelaskan. Berdasarkan *scree plot* yang ditampilkan, kita dapat menganalisis pentingnya masing-masing komponen utama dalam menjelaskan varians dataset. Plot menunjukkan bahwa komponen pertama menjelaskan lebih dari 70% varians dalam data, yang menandakan bahwa komponen ini sangat signifikan dan menangkap sebagian besar informasi dari dataset. Komponen kedua menjelaskan sekitar 20% varians, masih cukup signifikan namun jauh lebih rendah dibandingkan komponen pertama. Komponen ketiga menjelaskan sekitar 10% varians, dan setelah ini, jumlah varians yang dijelaskan oleh masing-masing komponen utama mulai menurun drastis. Komponen keempat dan kelima masing-masing hanya menjelaskan kurang dari 5% varians, menunjukkan bahwa kontribusi mereka terhadap total varians dalam dataset sangat kecil. Terdapat titik "*elbow*" yang jelas pada komponen ketiga, yaitu titik di mana penambahan komponen utama berikutnya tidak lagi memberikan peningkatan yang signifikan dalam varians yang dijelaskan. Berdasarkan hal ini, jumlah komponen utama yang optimal kemungkinan berada di sekitar dua atau tiga komponen.



Gambar 2. *Scree Plot*

3.8. Analisis Hasil Komparasi Model

Tahap selanjutnya adalah membandingkan diantara lima algoritma tersebut berdasarkan performa klasifikasinya. Membandingkan kinerja beberapa algoritma *machine learning* berdasarkan tiga metrik: Akurasi, rasio *Train-Test*, dan nilai AUC (*Area Under the Curve*), serta memberikan peringkat berdasarkan kinerja keseluruhan. Pada tabel 9 menunjukkan hasil evaluasi beberapa algoritma menggunakan semua variable. Dalam tabel tersebut, Support Vector Machine (SVM) dan Random Forest menunjukkan akurasi tertinggi sebesar 0.83, dengan nilai AUC masing-masing 0.33 dan 0.75. Logistic Regression memiliki akurasi 0.67 dan nilai AUC sempurna 1.00, sedangkan G*aussian Naive Bayes* memiliki akurasi rendah sebesar 0.33 dan nilai AUC 0.50. KNeighbors Classifier menunjukkan akurasi 0.67 dan nilai AUC 0.62. Setelah menggunakan PCA untuk mereduksi variabel, terlihat adanya perubahan signifikan dalam performa beberapa algoritma. SVM mengalami penurunan akurasi dari 0.83 menjadi 0.50 dan nilai AUC dari 0.33 menjadi 0.25. Random Forest juga mengalami penurunan akurasi dari 0.83 menjadi 0.50 dan nilai AUC dari 0.75 menjadi 0.62. Logistic Regression menunjukkan penurunan akurasi dari 0.67 menjadi 0.33, meskipun nilai AUC tetap tinggi, turun dari 1.00 menjadi 0.75. Menariknya, Gaussian Naive Bayes mengalami peningkatan signifikan dalam akurasi dari 0.33 menjadi 0.83 dan peningkatan nilai AUC dari 0.50 menjadi 0.78. KNeighbors Classifier mengalami penurunan akurasi dari 0.67 menjadi 0.50 dan penurunan nilai AUC dari 0.62 menjadi 0.38.

Penerapan PCA untuk mereduksi variabel dapat mempengaruhi performa model secara signifikan, tergantung pada algoritma yang digunakan. Sementara Gaussian Naive Bayes menunjukkan peningkatan performa yang signifikan setelah menggunakan PCA, algoritma lain seperti SVM dan *Random Forest* mengalami penurunan performa. Hal ini menunjukkan bahwa PCA dapat meningkatkan atau menurunkan performa model tergantung pada sifat data dan algoritma yang digunakan.

Tabel 9. Hasil Komparasi Model *Machine learning*

Algoritma	Akurasi	*Train-Test*	Nilai AUC	Rank
Support Vector Machine	0,83	0,3	0,33	4
Random Forest	0,83	0,3	0,75	1
Logistic Regression	0,67	0,3	1,00	2
Gaussian Naive Bayes	0,33	0,3	0,50	5
KNNeighbors Classifier	0,67	0,3	0.62	3

Tabel 10. Hasil Komparasi Model *Machine learning* Menggunakan PCA

Algoritma	Akurasi	*Train-Test*	Nilai AUC	Rank
Support Vector Machine	0,50	0,3	0,25	4
Random Forest	0,50	0,3	0,62	2
Logistic Regression	0,33	0,3	0,75	5
Gaussian Naive Bayes	0,83	0,3	0,78	1
KNNeighbors Classifier	0,50	0,3	0.38	3

3.9. Simulasi Model

Berdasarkan hasil perbandingan masing-masing model algoritma yang telah dijabarkan sebelumnya, diketahui bahwa model algorirtma *Gaussian Naive Bayes* memiliki performa paling terbaik diantara model algoritma lainnya yang dilihat dari nilai AUC nya. Oleh karena itu, simulasi model akan diterapkan pada model algoritma *Gaussian Naive Bayes*. Pada contoh yang diberikan, hasil prediksi menunjukkan bahwa vendor tersebut masuk dalam kategori "*High Repeat Order*," yang membantu dalam pengambilan keputusan manajemen vendor. Berikut merupakan kode program untuk mensimulasikan model algoritma *Gaussian Naive Bayes*.

```
# Nilai Bobot Kriteria
# Quality = 0.25
# Cost = 0.25
# Delivery = 0.20
# Flexibility = 0.15
# Responsiveness = 0.15

# Perhitungan Nilai Kriteria
a = float(input('Q1 :'))
b = float(input('Q2 :'))
c = float(input('C1 :'))
d = float(input('C2 :'))
e = float(input('C3 :'))
f = float(input('D1 :'))
g= float(input('D2 :'))
h = float(input('D3 :'))
i = float(input('F1 :'))
j = float(input('F2 :'))
k = float(input('R1 :'))
l = float(input('R2 :'))
m = float(input('R3 :'))

X1 = (a + b)/2
X2 = (c + d + e)/3
X3 = (f + g + h)/3
X4 = (i + j)/2
X5 = (k + l + m)/3

Quality_Excellent = 0.25*X1
Operational_Efficiency = (0.25*X2)+(0.20*X3) +(0.15*X4) + (0.15*X5)
```

```
Q1 : 5
Q2 : 3
C1 : 4
C2 : 5
C3 : 3
D1 : 4
D2 : 5
D3 : 3
F1 : 4
F2 : 5
R1 : 4
R2 : 5
R3 : 3
```

```
print(gnb.predict([[Quality_Excellent,Operational_Efficiency]]))
['High Repeat Order']
```

Gambar 3. Simulasi Model Klasifikasi "*High Repeat Order*"

IV. KESIMPULAN

Penelitian ini bertujuan membangun model klasifikasi vendor yang akurat dan efisien menggunakan *Machine learning* untuk menentukan *Repeat Order* pada vendor di PT XYZ. Responden adalah karyawan departemen procuremen dengan pengalaman kerja lebih dari lima tahun. Beberapa kriteria penting dalam pemilihan vendor berdasarkan *Vendor Performance Index* (VPI). Berdasarkan pengumpulan data dari hasil kuesioner lalu dilakukan uji normalitas menggunakan *Shapiro-Wilk* dan *Kolmogorov-Smirnov* menunjukkan data terdistribusi normal. Dari hasil uji parametrik didapatkan nilai statistik t sebesar 3.991 dengan nilai p sebesar 0.000855 pada tingkat signifikansi 5%. Hasil ini menunjukkan adanya perbedaan yang signifikan secara statistik antara rata-rata "*High Repeat Order*" dan "*Low Repeat Order*". Kemudian dilakukan analisis PCA dan scree plot untuk mereduksi dimensi data dengan mengekstraksi informasi penting dari dataset. Komponen utama pertama secara signifikan menjelaskan sebagian besar varians dalam data, dengan persentase varians yang dijelaskan mencapai lebih dari 70%. Ini menandakan bahwa komponen pertama memiliki kontribusi yang sangat besar dalam merepresentasikan pola atau struktur dalam data asli. Komponen utama kedua juga memiliki kontribusi yang signifikan dengan menjelaskan sekitar 20% varians. Dari lima model *Machine learning* yang diuji, *Gaussian Naive Bayes* memiliki performa terbaik dengan akurasi 0.83 dan nilai AUC 0.78. Model ini membantu PT XYZ dalam membuat keputusan yang lebih terinformasi mengenai kelayakan vendor untuk *Repeat Order*, meningkatkan efisiensi dan efektivitas pengambilan keputusan serta mengurangi risiko praktik bisnis yang merugikan.

DAFTAR PUSTAKA

[1] Y. Dao, "Kepemimpinan Strategis Di Era Vuca (Volatility, Uncertainty, Complexity And Ambiguity)," no. March, 2023, [Online]. Available: https://www.mohagadate.com/

[2] A. Nata and Suparmadi, "ANALISIS SISTEM PENDUKUNG KEPUTUSAN DENGAN MODEL KLASIFIKASI BERBASIS MACHINE LEARNING DALAM PENENTUAN PENERIMA PROGRAM," *J. Sci. Soc. Res.*, vol. 4307, no. 3, pp. 697–702, 2022.

[3] M. R. Adrian, M. P. Putra, M. H. Rafialdy, N. A. Rakhmawati, and D. S. Informasi, "Perbandingan Metode Klasifikasi Random Forest dan SVM Pada Analisis Sentimen PSBB," vol. 7, no. 1, pp. 36–40, 2021.

[4] K. Astoni and M. Haris, "ANALISIS PENERAPAN PRINCIPAL COMPONENT ANALYSIS (PCA) PADA DETEKSI KECURANGAN KARTU KREDIT MENGGUNAKAN RANDOM FOREST AN ANALYSIS OF PRINCIPAL COMPONENT ANALYSIS IMPLEMENTATION ON CREDIT CARD FRAUD DETECTION," vol. 9, no. 1, pp. 1152–1161, 2022.

[5] R. Kesuma, H. Akbar, and A. Hasdyna, "Algoritma K-Nearest Neighbor dengan Euclidean

Distance dan Manhattan Distance untuk Klasifikasi Transportasi Bus," *Ilk. J. Ilm.*, vol. 12, no. 2, pp. 104–111, 2020.

[6] A. P. Ayudhitama *et al.*, "ANALISA 4 ALGORITMA DALAM KLASIFIKASI PENYAKIT LIVER," pp. 1–9.

[7] M. Siddik, Hendri, R. Noratama Putri, Y. Desnelita, and Gustientiedina, "KLASIFIKASI KEPUASAN MAHASISWA TERHADAP PELAYANAN PERGURUAN TINGGI MENGGUNAKAN ALGORITMA NAÏVE BAYES," *J. Inf. Technol. Comput. Sci.*, vol. 3, pp. 162–166, 2020.

[8] A. Ciputra, D. R. I. Moses Setiadi, E. H. Rachmawanto, and A. Susanto, "KLASIFIKASI TINGKAT KEMATANGAN BUAH APEL MANALAGI DENGAN ALGORITMA NAIVE BAYES DAN EKSTRAKSI FITUR CITRA DIGITAL," *J. SIMETRIS*, vol. 9, no. 1, pp. 465–472, 2018.

[9] N. L. Hanun and A. U. Zailani, "PENERAPAN ALGORITMA KLASIFIKASI RANDOM FOREST UNTUK PENENTUAN KELAYAKAN PEMBERIAN KREDIT DI KOPERASI MITRA SEJAHTERA," *J. Technol. Inf.*, vol. 6, no. 1, pp. 7–14, 2020.

[10] M. Ahmad, S. Aftab, M. S. Bashir, N. Hameed, I. Ali, and Z. Nawaz, "SVM Optimization for Sentiment Analysis," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. April, pp. 393–398, 2018, doi: 10.14569/IJACSA.2018.090455.

[11] H. Rianto and R. S. Wahono, "Resampling Logistic Regression untuk Penanganan Ketidakseimbangan Class pada Prediksi Cacat Software," *J. Softw. Eng.*, vol. 1, no. 1, pp. 46–53, 2015.

[12] A. Quraisy, "Normalitas Data Menggunakan Uji Kolmogorov-Smirnov dan Saphiro-Wilk," *J-HEST J. Heal. Educ. Econ. Sci. Technol.*, vol. 3, no. 1, pp. 7–11, 2022, doi: 10.36339/jhest.v3i1.42.

[13] G. D. Ahadi, N. Nur, and L. Ersela, "The Simulation Study of Normality Test Using Kolmogorov-Smirnov ," vol. 6, no. 1, 2023.

[14] Muhtadi, "Penerapan Principal Component Analysis (PCA) dalam Algoritma K-Means untuk Menentukan Centroid pada Clustering," vol. Vol. 1, No, pp. 122–142, 2017.

[15] M. Z. Nasution, J. Jendral, G. Subroto, S. S. Medan, and S. Utara-indonesia, "PENERAPAN PRINCIPAL COMPONENT ANALYSIS (PCA) DALAM PENENTUAN FAKTOR DOMINAN YANG MEMPENGARUHI PRESTASI BELAJAR SISWA (Studi Kasus : SMK Raksana 2 Medan)," vol. 3, no. 1, 2019.

[16] V. Kotu and D. Bala, *Predictive Analytics and Data Mining*. San Francisco: Morgan Kaufmann Publisher, 2015.

[17] A. Kulkarni, D. Chong, and F. A. Batarseh, "Foundations of data imbalance and solutions for a data democracy," *Data Democr. Nexus Artif. Intell. Softw. Dev. Knowl. Eng.*, pp. 83–106, Jan. 2020, doi: 10.1016/B978-0-12-818366-3.00005-8.

[18] T. T. Maskoen and D. Purnama, "Area Under the Curve dan Akurasi Cystatin C untuk Diagnosis Acute Kidney Injury pada Pasien Politrauma," *Maj. Kedokt. Bandung*, vol. 50, no. 4, pp. 259–264, 2018, doi: 10.15395/mkb.v50n4.1342.