

Penerapan *Principal Component Analysis* (PCA) pada Segmentasi Pelanggan dengan Metode RFM

Stefan Matthew Susanto and 13523020^{1,2}

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

¹13523020@mahasiswa.itb.ac.id, ²stefanmattew246@gmail.com

Abstrak— Dalam era persaingan bisnis yang ketat, perusahaan perlu melakukan analisis data pelanggan. Hal ini menjadi salah satu hal yang penting untuk upaya memahami kebutuhan dan perilaku konsumen. Makalah ini membahas tentang penerapan *Principal Component Analysis* (PCA) pada segmentasi pelanggan dengan metode RFM (*Recency, Frequency, Monetary*). Metode RFM digunakan untuk mengukur perilaku pelanggan berdasarkan tiga dimensi utama: frekuensi pembelian, waktu terakhir pembelian, dan jumlah uang yang dibelanjakan. PCA diterapkan untuk mereduksi dimensi data tanpa kehilangan informasi yang penting, sehingga analisis dapat menjadi lebih efektif dan efisien. Hasil dari penelitian ini menunjukkan kombinasi RFM dan PCA dapat mengelompokkan pelanggan ke dalam segmen-segmen dan dapat mengembangkan sistem rekomendasi pemasaran yang tepat dan sesuai.

Keywords—*Principal Component Analysis, Segmentasi, Metode RFM (Recency, Frequency, Monetary), Eigen*

I. PENDAHULUAN

Di era *modern* saat ini dimana berbagai industry memiliki persaingan bisnis yang ketat, perusahaan kini perlu memahami secara mendalam mengenai perilaku dan karakteristik pelanggan untuk menjaga keunggulan kompetitif. Salah satu pendekatan yang cukup banyak digunakan adalah segmentasi berbasis data.

Segmentasi pelanggan membantu perusahaan untuk dapat mengidentifikasi kelompok pelanggan serta memahami karakteristiknya agar strategi *marketing* dapat disesuaikan secara efektif. Salah satu metode yang paling banyak digunakan adalah metode RFM. Metode RFM mengukur seberapa baru transaksi terakhir dilakukan (*Recency*), seberapa sering pelanggan bertransaksi (*Frequency*), dan berapa banyak nilai uang yang dibelanjakan (*Monetary*). Metode ini cukup efektif digunakan, namun terdapat tantangan yang dimiliki metode ini yaitu apabila data yang dimiliki cukup besar, sehingga sulit untuk menemukan kesimpulan. Untuk dapat mengatasi tantangan tersebut digunakan teknik *Principal Component Analysis* dalam mereduksi dimensi. Teknik tersebut banyak digunakan dalam *machine learning*. Teknik ini dapat menyederhanakan data yang memiliki banyak variabel diubah menjadi komponen data. Dengan gabungan metode tersebut, pengolahan data pelanggan

yang besar dapat disederhanakan tanpa kehilangan informasi penting. Pada makalah ini bertujuan untuk meninjau penerapan PCA dalam metode RFM dalam segmentasi pelanggan. Untuk dapat meninjau data diambil dari dataset pembelian online dan diproses dengan pemrograman bahasa Python.

II. DASAR TEORI

2.1. Segmentasi Pelanggan

Segmentasi pelanggan adalah proses untuk membagi pelanggan ke dalam beberapa grup berdasarkan kesamaan. Menurut Kotler (2018), segmentasi pelanggan dapat dibagi berdasarkan geografis, demografis, psikografis, dan perilaku. Segmentasi geografis membagi pasar ke dalam geografis yang berbeda seperti kota atau daerah. Segmentasi demografis membagi pasar ke dalam variable demografis seperti umur, jenis kelamin, pekerjaan, dan lain-lain. Segmentasi psikografis membagi pasar berdasarkan gaya hidup atau kelas sosial. Segmentasi perilaku membagi berdasarkan pengetahuan, sikap, penggunaan produk.

2.2. Metode RFM

Metode RFM adalah salah satu metode yang paling sering digunakan dalam teknik segmentasi, yang terdiri dari *recency, frequency, dan monetary*. (Wei, 2010). Metode RFM terdiri dari kombinasi 3-digit nilai RFM yang mewakili 5 kuantil (20% grup). (Miglausch, 2000) Metode rata-rata dengan membagi rata nilai R, F, dan M secara rata. Kemudian variable tersebut dibagi menjadi dua yaitu nilai di bawah rata-rata dan di atas rata-rata. Selanjutnya, metode kuintil yang membagi keseluruhan populasi menjadi lima bagian yang sama. Masing-masing variable diberi skor 5 (tertinggi) hingga terendah. (Mc.Carty, 2007). Berikut ini merupakan penjelasan dari masing-masing variable *recency, frequency, dan monetary*.

2.3.1 *Recency*

Recency merupakan variable berapa lama waktu periode sejak transaksi terakhir pelanggan. Semakin baru pelanggan melakukan transaksi, maka skor *recency*nya semakin tinggi. Semakin baru pelanggan melakukan transaksi, pelanggan tersebut cenderung memiliki kemungkinan membuat transaksi kembali.

2.3.2 Frequency

Frequency menjelaskan seberapa sering pelanggan dalam melakukan transaksi. Semakin banyak pelanggan melakukan transaksi, maka skor frekuensinya semakin tinggi. Hal ini dikarenakan pelanggan yang sering bertransaksi memiliki demand yang tinggi kepada produk.

2.3.3 Monetary

Monetary menjelaskan total uang yang dihabiskan dalam transaksi di spesifik periode transaksi. Semakin banyak total uang transaksi yang dilakukan, maka skor moneternya akan semakin tinggi. Pelanggan yang melakukan transaksi dengan jumlah uang banyak merupakan asset yang baik untuk dipertahankan perusahaan.

2.3 Principal Component Analysis

Principal Component Analysis (PCA) adalah metode statistik yang digunakan untuk mengurangi dimensi data namun tetap mempertahankan informasi yang penting.

PCA mengubah data ke dalam ruang baru yang disebut komponen utama, di mana dimensi pertama merepresentasikan varians terbesar, dimensi kedua merepresentasikan varians terbesar berikutnya, dan seterusnya.

Tujuan PCA :

- Mengurangi Dimensi Data
Mengurangi data data yang tidak terpakai dengan baik tanpa menghapus data penting.
- Menghilangkan Redundansi
Menghapus korelasi antar fitur.
- Meningkatkan Efektifitas
Menyederhanakan data supaya lebih efisien

2.4 Nilai Eigen dan Vektor Eigen

Jika A merupakan matriks $n \times n$ maka skalar λ adalah nilai eigen dari A, dan x merupakan vector eigen yang berkoresponden dengan λ . Vector eigen menyatakan hasil perkalian matriks A dengan x yang searah dengan x dan panjangnya berubah sebesar factor λ .

$$Ax = \lambda x \quad (1)$$

Dari persamaan (1) dapat dibentuk $(\lambda I - A)x = 0$. $x=0$ adalah solusi trivial dari $(\lambda I - A)x = 0$, supaya memiliki Solusi non trivial, maka nilai determinannya sama dengan nol. Didapat persamaan karakteristik : $\det(\lambda I - A) = 0$

Sifat – sifat nilai eigen dan vector eigen:

- Sebuah matriks memiliki invers jika dan hanya jika $\lambda \neq 0$ bukan nilai eigen.
- Matriks diagonal memiliki elemen diagonal berupa nilai eigen. Matriks A dapat didiagonalisasi jika ada matriks P sehingga $P^{-1}AP$ adalah matriks diagonal dengan kolom P adalah basis ruang eigen matriks A

2.5 Standardisasi Data

Standardisasi Z-Score merupakan salah satu teknik yang sering digunakan dalam menstandarisasi data. Teknik berikut mengubah nilai data menjadi rata-rata 0 dan standar deviasi 1. Berikut ini merupakan rumus untuk mengetahui standardisasi z-score.

$$z = \frac{x - \mu}{\sigma} \quad (2)$$

dengan:

- z: nilai terstandarisasi
- x: nilai data
- μ : rata-rata
- σ : standar deviasi

2.6 Matriks Kovarian

Matriks kovarians merepresentasikan matriks persegi yang digunakan untuk merepresentasikan nilai kovarians pasangan elemen

$$c = \frac{1}{n - 1} Z^T Z \quad (3)$$

Dengan:

- Z: matriks data terstandarisasi
- N: jumlah sampel

III. IMPLEMENTASI

A. Dataset

Dataset yang diambil dari makalah ini diambil dari Kaggle. Kaggle adalah suatu platform yang digunakan untuk *data science* dan *machine learning* di kalangan profesional. Untuk makalah ini, data original diambil dari data pembelian di perusahaan retail online di UK selama 8 bulan (Terry, 2024).

Dataset terdiri dari 8 kolom yaitu nomor invoice, kode postal, deskripsi produk, kuantitas, tanggal invoice, harga per unit, ID customer, dan negara.

Online Retail.csv

Tabel 1 Variabel dalam Dataset

Variabel	Deskripsi
InvoiceNo	Nomor invoice pembelian
StockCode	Kode postal yang dimiliki pelanggan
Description	nama produk yang dibeli oleh pelanggan
#Quantity	Kuantitas yang dibeli per masing-masing produk
Invoice Date	Tanggal invoice pembelian yang terjadi
#Unit Price	Harga satuan masing-masing produk
CustomerID	Identifier unik setiap pelanggan
Country	Negara asal pelanggan

B. Data Pre-Processing

Data *pre-processing* merupakan langkah pertama untuk memproses suatu data dari .csv yang diperoleh dari dataset link tersebut. Untuk memproses data kita mengambil data .csv yang telah diunduh sebelumnya, dengan menggunakan *library* pandas. Isi dari kolom Invoice Date akan diubah menjadi *datetime*. Karena data.csv yang digunakan cukup besar memungkinkan terdapat data yang hilang / tidak lengkap.

```
import pandas as pd
from datetime import datetime
import numpy as np
import matplotlib.pyplot as plt

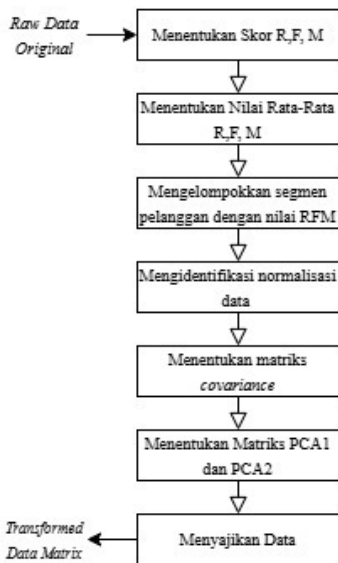
# Data Pre-processing
# DATASET = https://www.kaggle.com/code/alfatherry/customer-segmentation-rfm
file_path = "OnlineRetail.csv"
data = pd.read_csv(file_path)

data['InvoiceDate'] = pd.to_datetime(data['InvoiceDate']) # Mengubah menjadi datetime contoh 02/12/2010
data = data.dropna(subset=['CustomerID']) # Menghapus baris atau isi dari data yang hilang atau Nan
```

Gambar 1 Tahap Data Pre-Processing

C. Data Processing

Terdapat beberapa langkah dalam memproses data dan secara garis besar dibagi menjadi dua yaitu langkah penentuan RFM dan PCA.



Lanjutan dari matriks *Recency*, *Frequency*, *Monetary* yang unik setiap *customerid*. Skor R,F,dan M dibagi dengan menggunakan metode kuintil yang membagi menjadi lima bagian yang sama.

```
rfm.columns = ['CustomerID', 'Recency', 'Frequency', 'Monetary']
```

Gambar 3 Membentuk Kolom Matriks R,F,M

Langkah selanjutnya adalah menentukan masing-masing skor R (*Recency*), F(*Frequency*), dan M(*Monetary*) pada setiap *customerid*. Untuk menampilkan grafik hasil olahan datamenggunakan library matplotlib.pyplot.

```
# Skor R
rfm['R_Score'] = pd.qcut(rfm['Recency'], 5, labels=[5, 4, 3, 2, 1]).astype(int) # menggunakan pd.qcut supaya lebih efisien

plt.subplot(3, 3, 1)
plt.bar(rfm['R_Score'].value_counts().index, rfm['R_Score'].value_counts().values)
plt.xlabel('R Score (Recency)')
plt.ylabel('Number of Customers')
plt.title('R Score Distribution')
```

```
# Skor F
rfm['F_Score'] = pd.qcut(rfm['Frequency'], q=6, labels=False, duplicates='drop') + 1

plt.subplot(1, 3, 2)
plt.bar(rfm['F_Score'].value_counts().index, rfm['F_Score'].value_counts().values)
plt.xlabel('F Score (Recency)')
plt.ylabel('Number of Customers')
plt.title('F Score Distribution')

# Skor M
rfm['M_Score'] = rfm['Monetary'].rank(method='dense', ascending=True).apply(
    lambda x: min(int((x / len(rfm)) * 5) + 1, 5)
).astype(int)

plt.subplot(1, 3, 3)
plt.bar(rfm['M_Score'].value_counts().index, rfm['M_Score'].value_counts().values)
plt.xlabel('M Score (Recency)')
plt.ylabel('Number of Customers')
plt.title('M Score Distribution')
```

Gambar 4 Menghitung skor RFM

Untuk menampilkan grafik kita perlu menggunakan rumus `plt.tight_layout()` dan `plt.show()`. Langkah berikutnya adalah membentuk masing-masing matrix R, F, dan M. Tujuannya adalah mengurutkan secara berurutan nilai RFM.

```
R_matrix = rfm['R_Score'].value_counts().sort_index()
F_matrix = rfm['F_Score'].value_counts().sort_index()
M_matrix = rfm['M_Score'].value_counts().sort_index()
```

Gambar 5 Mengurutkan nilai RFM

Setelah mengurutkan data, kita perlu menghitung rata-rata R, F, dan M setiap *customerid*. Rata-rata nilai RFM dibutuhkan untuk mengelompokkan ribuan pelanggan ke dalam segmen-segmen.

```
# Menghitung rata-rata R,F,dan M
rfm['RFM_Score'] = (rfm['R_Score'] + rfm['F_Score'] + rfm['M_Score']) / 3
```

Gambar 6 Menghitung Skor Rata-Rata RFM

Supaya kita dapat mereduksi dimensi, kita menggunakan PCA. Dengan PCA kita dapat meringkas informasi *Recency*, *Frequency* dan *Monetary* kedalam 3 dimensi utama tanpa kehilangan banyak informasi penting.

PCA sebaiknya dilakukan setelah dilakukannya RFM karena RFM perlu menganalisis perilaku pelanggan. Jika dilakukan PCA terlebih dahulu, maka komponen-komponen penting kurang dapat dibaca dan dianalisis dengan mudah.

Data perlu dinormalisasi dengan menggunakan standar deviasi. Rumus standarisasi yang digunakan adalah rumus yang digunakan pada formula (1).

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

```
# PCA
# Melakukan standarisasi terlebih dahulu
rfm_scaled = (rfm[['Recency', 'Frequency', 'Monetary']] - \
    rfm[['Recency', 'Frequency', 'Monetary']].mean()) / rfm[['Recency', 'Frequency', 'Monetary']].std()
```

Gambar 7 Melakukan standarisasi data

Selanjutnya, kita perlu mencari hubungan linier antar variable. Untuk mencarinya dapat dilakukan dengan cara menghitung matriks kovarian dengan numpy pada

$$c = \frac{1}{n - 1} Z^T Z$$

```
# Matriks kovarina dengan numpy
cov_matrix = np.cov(rfm_scaled.T)
```

Gambar 8 Menghitung Matriks Kovarina dengan Numpy

Setelah mengetahui hubungan linear antara variable. Tahap selanjutnya yang dilakukan adalah kita mencari PCA1 dan PCA2 untuk mengurangi data.

```
pca_result = np.dot(rfm_scaled, eigenvector[:, :2])
rfm['PCA1'] = pca_result[:, 0]
rfm['PCA2'] = pca_result[:, 1]
```

Gambar 9 Mencari nilai PCA1 dan PCA2

Untuk menampilkan PCA 10 teratas berdasarkan customer id.

```
#Menampilkan hasil PCA 10 teratas berdasarkan customer id
print("Hasil PCA :")
print(rfm[['CustomerID', 'PCA1', 'PCA2']].head(10))
```

Gambar 10 Menampilkan PCA 10 teratas

Selanjutnya, semua dari hasil pengolahan PCA dan RFM kedalam file .csv yang baru dengan nama rfm_pca.csv

```
#Menuliskan hasil pca dan rfm ke dalam file CSV
rfm.to_csv('rfm_pca.csv', index=False)
print("File RFM dengan PCA telah disimpan sebagai 'rfm_pca.csv'")
```

Gambar 11 Menuliskan Hasil PCA dan RFM

Selanjutnya adalah menampilkan hasil skor dari customer.

```
#Menampilkan hasil skor dari customer
print("Jumlah CustomerID untuk setiap skor R:")
print(R_matrix)
print()
print("Jumlah CustomerID untuk setiap skor F:")
print(F_matrix)
print()
print("Jumlah CustomerID untuk setiap skor M:")
print(M_matrix)
print()

#Menampilkan skor RFM per customer
print("RFM Scores per Customer:")
print(rfm[['CustomerID', 'R_Score', 'F_Score', 'M_Score', 'RFM_Score']])
```

Gambar 12 Menampilkan Skor RFM

IV. HASIL DAN PEMBAHASAN

a. Hasil RFM

Setelah melakukan implementasi, kita mendapatkan jumlah hasil dari R_score tiap customerID. R_score menggambarkan seberapa baru pelanggan melakukan transaksi terakhir. Kita membagi nilai R menjadi 5 kuantil. R_score 1 melambangkan pelanggan yang telah lama tidak melakukan transaksi kembali. R_score 5 melambangkan pelanggan terbaru yang melakukan transaksi. Pada *recency*, persebaran jumlah skor R

cukup merata antar kelompok dengan *range* 850-905 *customerid*.

Selanjutnya kita mendapatkan jumlah nilai F_score dari hasil dari F_score tiap customerID. F_score menggambarkan seberapa sering pelanggan melakukan transaksi dalam suatu periode. F_score bernilai 5 melambangkan customer yang sering melakukan transaksi. F_score bernilai 1 melambangkan customer yang jarang melakukan transaksi. Pada komponen *frequency*, mayoritas pelanggan yang cenderung membeli dalam jumlah sedikit.

Selanjutnya kita mendapatkan jumlah nilai M_score dari tiap customerID. M_score menggambarkan seberapa banyak total uang yang dihabiskan oleh pelanggan. M_score bernilai 5 menggambarkan pelanggan yang menghabiskan total belanja yang besar. Pada *monetary*, nilai persebarannya cukup merata.

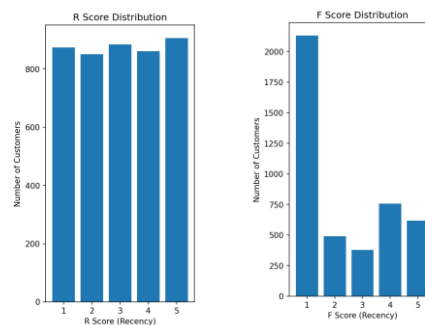
Jumlah CustomerID untuk setiap skor R:	
R_Score	Count
1	873
2	850
3	884
4	860
5	905

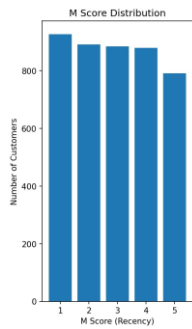
Jumlah CustomerID untuk setiap skor F:	
F_Score	Count
1	2130
2	490
3	377
4	758
5	617

Jumlah CustomerID untuk setiap skor M:	
M_Score	Count
1	926
2	891
3	885
4	879
5	791

Gambar 13 Hasil RFM masing-masing Skor

Penjelasan distribusinya banyak dimana





Gambar 14 Distribusi Customer pada segmen RFM

Pada gambar 14 menampilkan distribusi dalam segmen RFM untuk mempermudah dalam mengetahui karakteristik data. Gambar 15 berisi cuplikan data skor RFM pada masing-masing customerid secara unik. RFM skor final didapat dengan mencari nilai rata-rata R,F,dan M.

```
RFM Scores per Customer:
CustomerID  R_Score  F_Score  M_Score  RFM_Score
0           12346.0    1         1         1  1.000000
1           12347.0    5         4         5  4.666667
2           12348.0    2         3         4  3.000000
3           12349.0    4         1         4  3.000000
4           12350.0    1         1         2  1.333333
...
4367        18280.0    1         1         1  1.000000
4368        18281.0    1         1         1  1.000000
4369        18282.0    5         2         1  2.666667
4370        18283.0    5         5         4  4.666667
4371        18287.0    3         2         4  3.000000
[4372 rows x 5 columns]
```

Gambar 15 Cuplikan Data Skor RFM

Pada gambar 16, ditampilkan rata-rata skor R, F, M, dan RFM dari tiap customer. Hasil ini didapat dengan merata-ratakan nilai tiap skor. Dapat diperhatikan dari data tersebut, kita bisa menganalisis pola pola dan karakter keseluruhan dari customer.

```
Rata-rata skor RFM:
Rata-rata R: 3.02
Rata-rata F: 2.37
Rata-rata M: 2.94
Rata-rata RFM: 2.77
File RFM dengan PCA telah disimpan sebagai 'rfm_pca.csv'
```

Gambar 16 Rata-Rata Skor RFM Keseluruhan

```
# 31
count_s1 = rfm[(rfm['M_Score'] == 5) & (rfm['F_Score'] == 5) & (rfm['R_Score'] == 5)][['M_Score']].count()
print(f"#31 : {count_s1}")

# 32
count_s2 = rfm[(rfm['M_Score'] == 4) & (rfm['F_Score'] == 4) & (rfm['R_Score'] == 4)][['M_Score']].count()
print(f"#32 : {count_s2}")

# 33
count_s3 = rfm[(rfm['M_Score'] == 3) & (rfm['F_Score'] == 4) & (rfm['R_Score'] == 5)][['M_Score']].count()
print(f"#33 : {count_s3}")

# 34
count_s4 = rfm[(rfm['M_Score'] == 3) & (rfm['F_Score'] == 2) & (rfm['R_Score'] == 5)][['M_Score']].count()
print(f"#34 : {count_s4}")

# 35
count_s5 = rfm[(rfm['M_Score'] == 3) & (rfm['F_Score'] == 3) & (rfm['R_Score'] == 3)][['M_Score']].count()
print(f"#35 : {count_s5}")

# 36
count_s6 = rfm[(rfm['M_Score'] == 4) & (rfm['F_Score'] == 3) & (rfm['R_Score'] == 2)][['M_Score']].count()
print(f"#36 : {count_s6}")

# 37
count_s7 = rfm[(rfm['M_Score'] == 2) & (rfm['F_Score'] == 2) & (rfm['R_Score'] == 2)][['M_Score']].count()
print(f"#37 : {count_s7}")
```

```
# 99
count_s9 = rfm[(rfm['M_Score'] == 4) & (rfm['F_Score'] == 4) & (rfm['R_Score'] == 2)][['M_Score']].count()
print(f"#99 : {count_s9}")

# 110
count_s10 = rfm[(rfm['M_Score'] == 2) & (rfm['F_Score'] == 2) & (rfm['R_Score'] == 2)][['M_Score']].count()
print(f"#110 : {count_s10}")

# 111
count_s11 = rfm[(rfm['M_Score'] == 1) & (rfm['F_Score'] == 1) & (rfm['R_Score'] == 1)][['M_Score']].count()
print(f"#111 : {count_s11}")
```

Gambar 17 Formula untuk pengelompokan segmen RFM

Dari 4372 pelanggan yang memiliki nilai RFM yang beragam, kita perlu mengelompokkan berdasarkan karakteristik yang dimiliki. Pada gambar 17, untuk menampilkan nilai dari setiap s maka perlu mencari nilai R_score, F_score dan M_score yang sama lalu dijumlahkan dengan fungsi count.

Terdapat beberapa segmentasi yang dapat dilakukan. Peneliti menggunakan model RFM dari Moengage. (Moengage, 2022) Moengage merupakan platform yang digunakan untuk aktivitas CRM, berdasarkan *insight-led customer engagement platform*. Setelah skor diidentifikasi dan menghasilkan 10 segmen berdasarkan atribut RFM yang ditampilkan pada tabel 2.

Tabel 2 Pengelompokan 10 Segment RFM

Segmen	Nama Segment	Atribut	Jumlah Pelanggan
S1	Pelanggan juara (<i>champions</i>)	{R5, F5, M5}	296
S2	Pelanggan setia (<i>loyal customers</i>)	{R4, F4, M4}	111
S3	Pelanggan yang berpotensi setia (<i>potential loyalist</i>)	{R5, F4, M3}	29
S4	Pelanggan baru (<i>recent users</i>)	{R5, F2, M3}	31
S5	Pelanggan yang tidak dibiarkan hilang (<i>can't lose them</i>)	{R3, F3, M3}	34
S6	Pelanggan yang perlu diperhatikan (<i>needs attention</i>)	{R2, F3, M4}	31
S7	Pelanggan yang sensitive terhadap harga (<i>price sensitive</i>)	{R2, F2, M2}	28
S8	Pelanggan yang akan menghilang (<i>about to sleep</i>)	{R5, F4, M2}	4
S9	Pelanggan yang sudah lama tidak bertransaksi (<i>hibernating</i>)	{R2, F4, M4}	46
S10	Pelanggan yang sudah hilang (<i>lost customers</i>)	{R1, F1, M1}	378

Dari segment yang didapat diketahui segmen pelanggan juara (*champions*) sebesar 296 pelanggan. Segment tersebut merupakan segment yang melakukan transaksi paling baru, sering, dan paling banyak. Perusahaan perlu menjaga hubungan baik dengan segmen ini karena segmen ini memiliki peran penting dan memiliki jumlah yang banyak.

Selanjutnya adalah segmen yang sudah hilang (*lost customers*) sebesar 378 orang. Pelanggan tersebut sudah mulai berhenti menggunakan brand. Perusahaan perlu mengajak pelanggan ini kembali. Perusahaan juga dapat mengidentifikasi segmen-segmen yang lainnya dan dapat memberikan tindakan pemasaran khusus sesuai karakteristik pelanggan.

b. Hasil PCA

Hasil PCA :			
	CustomerID	PCA1	PCA2
0	12346.0	1.252951	-1.965249
1	12347.0	-0.663796	0.661120
2	12348.0	0.022738	0.170137
3	12349.0	0.025941	0.731629
4	12350.0	1.242390	-1.830818
5	12352.0	-0.610125	0.426175
6	12353.0	0.859528	-0.864019
7	12354.0	0.889331	-1.163850
8	12355.0	0.868886	-0.971903
9	12356.0	-0.184244	0.614284

Gambar 18 Hasil PCA

PCA1 adalah komponen utama pertama yang dihasilkan dari analisis PCA. Nilai PCA1 dapat digunakan untuk mengukur kontribusi atau karakteristik utama yang memengaruhi perbedaan antara pelanggan. Sedangkan, PCA2 adalah komponen utama kedua yang dihasilkan. PCA2 menambahkan informasi tambahan yang tidak tertangkap oleh PCA1.

VI. LAMPIRAN

Dokumentasi data dapat dilihat pada <https://github.com/StefanMatthew/PCA-in-RFM.git>

VII. UCAPAN TERIMA KASIH

Ucapan terimakasih kepada keluarga tercinta dan sahabat yang telah memberikan dukungan kepada penulis dalam menyelesaikan tugas berikut.

REFERENSI

- Kotler, P. &. (2018). *Principles of Marketing 17th edition*. . New York: Pearson Higher Education.
- Mc.Carty, J. &. (2007). Segmentation approaches in data-mining: A comparison of RFM, CHAID, and logistic regression. *Journal of Business Research*, 6.
- Miglautsch, J. R. (2000). Thoughts on RFM scoring. *Journal of Database Marketing*, 68.
- Moengage. (2022, February). Retrieved from RFM Segments: <https://help.moengage.com/hc/en-us/articles/360037365132-RFM-Segments#01H845E559R4NW5MJKZK1R5GTD>
- Munir, R. (2024, Februari). Retrieved from IF2123 Aljabar Linier dan Geometri: <https://informatika.stei.itb.ac.id/~rinaldi.munir/>
- Terry, A. f. (2024, April). Retrieved from Kaggle: <https://www.kaggle.com/code/alfatherry/customer-segmentation-rfm/notebook>
- Wei, J.-T. L.-Y.-H. (2010). A review of the application of RFM Model. *Journal of Business Management*. 19.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 27 Desember 2024



Stefan Matthew Susanto dan 13523020