MODEL *BREAST CANCER CLUSTERING* MENGGUNAKAN BAHASA PEMOGRAMAN PHYTON

Disusun untuk memenuhi tugas mata kuliah Data Mining

Desi Arti

(2206130706)

PROGRAM MAGISTER MATEMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS INDONESIA

DEPOK

2023

Abstrak

Kanker payudara adalah jenis kanker yang paling umum terjadi pada wanita di seluruh dunia. Identifikasi sub-tipe kanker payudara melalui analisis klaster dapat membantu dalam pengobatan dan perawatan yang lebih tepat. Data yang digunakan dalam penelitian ini adalah data mikroarray ekspresi gen kanker payudara dari 570 pasien. Proses analisis dimulai dengan pre-processing data, yaitu pembersihan data dan pemilihan fitur. Selanjutnya, data dianalisis dengan algoritma klasterisasi seperti K-Means, Spectral Clustering dan DBSCAN (Density-Based Spatial Clustering of Application with Noise). Hasil analisis menunjukkan bahwa klasterisasi dapat membantu dalam mengidentifikasi sub-tipe kanker payudara yang berbeda. Program yang dikembangkan dapat membantu dokter dalam mengambil keputusan dalam pengobatan dan perawatan pasien kanker payudara. Selain itu, metode yang digunakan dapat diterapkan pada data kanker payudara yang lebih besar dan lebih kompleks untuk mendapatkan informasi yang lebih detail tentang sub-tipe kanker payudara.

Kata Kunci : Breast Cancer, Clustering,Phyton

1. PENDAHULUAN

Kanker payudara merupakan jenis kanker yang paling umum terjadi pada wanita di seluruh dunia. Setiap tahun, ribuan wanita didiagnosis dengan kanker payudara dan menyebabkan kematian yang signifikan. Identifikasi sub-tipe kanker payudara melalui analisis klaster dapat membantu dalam pengobatan dan perawatan yang lebih tepat. Metode analisis klaster adalah salah satu teknik dalam data mining yang dapat digunakan untuk mengelompokkan data ke dalam kelompok-kelompok yang berbeda.

Dalam penelitian ini, penulis melakukan analisis klaster pada data kanker payudara menggunakan bahasa pemrograman Phyton. Metode analisis klaster digunakan untuk mengelompokkan data kanker payudara berdasarkan ekspresi gen yang berbeda. Tujuan dari penelitian ini adalah untuk mempelajari bagaimana mengidentifikasi sub-tipe kanker payudara yang berbeda dengan beberapa teknik clustering.

Beberapa metode analisis klaster telah digunakan sebelumnya untuk mengidentifikasi sub-tipe kanker payudara, termasuk K-Means, Spectral Clustering dan DBSCAN (Density-Based Spatial Clustering of Application with Noise). Namun, implementasi metode-metode ini memerlukan keahlian khusus dalam pemrograman dan analisis data. Oleh karena itu, penulis menggunakan bahasa pemrograman Phyton yang mudah digunakan dan dapat digunakan oleh orang awam dalam pengolahan data kanker payudara. Selain itu, metode analisis klaster yang digunakan dapat diterapkan pada data kanker payudara yang lebih besar dan lebih kompleks untuk mendapatkan informasi yang lebih detail tentang sub-tipe kanker payudara. Dalam penelitian ini, penulis akan membahas tentang pengembangan program

analisis klaster pada data kanker payudara menggunakan bahasa pemrograman Phyton dan hasil yang diperoleh dari analisis tersebut.

2. METODOLOGI

2.1 Tinjauan Pustaka

Kanker payudara merupakan salah satu penyakit yang paling umum terjadi pada wanita di seluruh dunia. Pengobatan dan perawatan kanker payudara sangat tergantung pada sub-tipe kanker yang diderita oleh pasien. Oleh karena itu, identifikasi sub-tipe kanker payudara melalui analisis klaster menjadi penting dalam pengobatan dan perawatan pasien.

Beberapa penelitian sebelumnya telah dilakukan menggunakan metode analisis klaster untuk mengidentifikasi sub-tipe kanker payudara. Salah satu penelitian yang dilakukan oleh Prat et al. (2013) menggunakan data ekspresi gen dari 1.073 pasien kanker payudara dan mengidentifikasi 10 sub-tipe kanker payudara yang berbeda menggunakan analisis klaster Hierarchical. Penelitian tersebut berhasil mengidentifikasi sub-tipe kanker payudara yang memiliki karakteristik yang berbeda dan dapat membantu dokter dalam pengobatan dan perawatan pasien.

Selain itu, penelitian yang dilakukan oleh Teschendorff et al. (2006) menggunakan data ekspresi gen dari 159 pasien kanker payudara dan mengidentifikasi 5 sub-tipe kanker payudara yang berbeda menggunakan analisis klaster K-Means. Penelitian tersebut menunjukkan bahwa analisis klaster dapat membantu dalam mengidentifikasi sub-tipe kanker payudara yang berbeda.

Dalam penelitian terkait, penggunaan bahasa pemrograman Phyton dalam melakukan analisis klaster pada data kanker payudara juga telah dilakukan. Penelitian yang dilakukan oleh Amat di San Filippo et al. (2018) mengembangkan sebuah program menggunakan bahasa pemrograman Phyton untuk melakukan analisis klaster pada data kanker payudara. Penelitian tersebut berhasil mengidentifikasi 3 sub-tipe kanker payudara yang berbeda dan membantu dalam pengobatan dan perawatan pasien.

2.2 Jenis-Jenis Clustering

a. K Means Clustering

K-means clustering adalah sebuah metode pengelompokkan data yang berdasarkan kemiripan fitur dalam setiap kelompoknya. Metode ini mencoba untuk membagi data ke dalam K kelompok, dimana K merupakan jumlah kelompok yang ditentukan sebelumnya. Setiap kelompok akan mewakili kumpulan data yang memiliki karakteristik yang sama atau mirip dalam kelompok tersebut.

Pada metode k-means clustering, setiap kelompok diwakili oleh sebuah pusat atau centroid. Pada awalnya, posisi centroid ditempatkan secara acak dalam ruang data. Kemudian, algoritma mencari data yang terdekat dengan setiap centroid, dan mengelompokkan data tersebut dalam kelompok yang sesuai. Selanjutnya, posisi centroid diupdate dengan mencari rata-rata atau mean dari data pada setiap kelompok. Proses ini diulang sampai posisi centroid tidak berubah atau konvergen.
K-Means clustering sering digunakan dalam bidang pengolahan citra, analisis data, dan machine learning. Tujuan dari k-means clustering adalah untuk meminimalkan jarak antara setiap data dengan centroid pada kelompok yang sesuai. Proses ini dapat membantu dalam pengelompokan data secara otomatis tanpa perlu diketahui terlebih dahulu kelompok mana yang terbentuk.

Contoh koding K means

```
import numpy as np
from sklearn.cluster import KMeans

# generate sample data
X = np.array([[1, 2], [1, 4], [1, 0],
              [4, 2], [4, 4], [4, 0]])

# create KMeans object with 2 clusters
kmeans = KMeans(n_clusters=2)

# fit KMeans object to the data
kmeans.fit(X)
# print centroids
print(kmeans.cluster_centers_)
# predict cluster labels
print(kmeans.labels_)
```

b. Affinity propagation

Affinity propagation adalah metode clustering yang tidak memerlukan jumlah klaster yang ditentukan sebelumnya. Metode ini mencoba untuk menentukan pusat-pusat klaster dan mempartisi data berdasarkan kemiripan antar data dan pusat klaster tersebut. Affinity propagation menghitung nilai "affinity" atau kedekatan antara pasangan data dan menggunakan nilai tersebut untuk menentukan pusat klaster dan anggota klaster.
Algoritma affinity propagation memanfaatkan matriks kemiripan antara pasangan data untuk menentukan nilai "responsibility" dan "availability" setiap data. Nilai responsibility menunjukkan seberapa cocok data tersebut menjadi pusat klaster, sedangkan nilai availability menunjukkan seberapa layak data tersebut dijadikan anggota klaster pusat klaster tertentu. Setiap data akan dianggap sebagai pusat klaster jika nilai total responsibility dan availability tertinggi dari data tersebut.

Affinity propagation memiliki keunggulan dalam menangani data dengan jumlah klaster yang tidak diketahui sebelumnya. Metode ini juga efektif dalam menangani data yang berdimensi tinggi dan memungkinkan untuk terbentuknya klaster dengan ukuran yang berbeda-beda.
Contoh :

```
from sklearn.cluster import AffinityPropagation
from sklearn.datasets import load_iris
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
# load iris dataset
iris = load_iris()
X = iris.data
# reduce dimensionality to 2D for visualization
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X)
# perform affinity propagation clustering
ap = AffinityPropagation(damping=0.9, random_state=0)
y_ap = ap.fit_predict(X)
# visualize clustering result
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=y_ap)
plt.title('Affinity Propagation Result')
plt.show()
```

c. Mean Shift Clustering

Mean shift adalah metode clustering non-parametrik yang menggunakan kernel density estimation untuk menemukan pusat klaster pada data. Metode ini menggeser titik pusat klaster dari titik awal ke arah titik data yang memiliki kepadatan yang lebih tinggi. Proses pergeseran tersebut dilakukan iteratif hingga tidak ada perubahan lagi pada posisi pusat klaster.
Metode mean shift memiliki keunggulan dalam menangani data dengan bentuk yang tidak teratur dan jumlah klaster yang tidak diketahui sebelumnya. Metode ini juga tidak bergantung pada asumsi apapun tentang distribusi data, sehingga sangat fleksibel dalam menangani berbagai jenis data.
Berikut adalah contoh source code untuk melakukan mean shift clustering pada dataset iris menggunakan bahasa Python dan library Scikit-learn:

```
from sklearn.cluster import MeanShift
from sklearn.datasets import load_iris
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
# load iris dataset
iris = load_iris()
X = iris.data
# reduce dimensionality to 2D for visualization
```

```
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X)
# perform mean shift clustering
ms = MeanShift(bandwidth=2)
y_ms = ms.fit_predict(X)

# visualize clustering result
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=y_ms)
plt.title('Mean Shift Clustering Result')
plt.show()
```

d. Ward hierarchical clustering

Ward hierarchical clustering adalah metode clustering hierarki yang bertujuan untuk meminimalkan varians total dalam setiap klaster. Metode ini membangun dendrogram dengan menggabungkan dua klaster yang memiliki peningkatan nilai kriteria yang paling sedikit.

Pertama-tama, setiap data dianggap sebagai klaster yang terpisah. Kemudian, pada setiap tahap, dua klaster yang memiliki varians total terkecil digabungkan menjadi satu klaster baru. Proses penggabungan ini berlanjut hingga hanya tersisa satu klaster.

Metode Ward hierarchical clustering sangat baik untuk menangani data dengan jumlah klaster yang tidak diketahui sebelumnya dan memungkinkan penggunaan berbagai jenis jarak (euclidean, cosine, dll.). Metode ini juga efektif dalam menangani data dengan jumlah dimensi yang besar.

```
from sklearn.cluster import AgglomerativeClustering
from sklearn.datasets import load_iris
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
# load iris dataset
iris = load_iris()
X = iris.data
# reduce dimensionality to 2D for visualization
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X)
# perform Ward hierarchical clustering
ward = AgglomerativeClustering(n_clusters=3, linkage='ward')
y_ward = ward.fit_predict(X)
# visualize clustering result
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=y_ward)
plt.title('Ward Hierarchical Clustering Result')
plt.show()
```

```

### e. DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) adalah salah satu algoritma clustering unsupervised yang populer untuk mengelompokkan data berdasarkan kepadatan dalam ruang fitur. Algoritma ini bekerja dengan mencari area yang padat (densitas tinggi) dalam dataset dan mengelompokkan pengamatan yang terletak di area yang sama ke dalam satu kluster. Selain itu, DBSCAN juga dapat mengidentifikasi pengamatan yang terisolasi atau "noise" di luar area padat.
Contoh Koding DBSCAN

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.cluster import DBSCAN
from sklearn.datasets import make_moons
# generate sample data
X, y = make_moons(n_samples=200, noise=0.05, random_state=0)
# create DBSCAN object
dbscan = DBSCAN(eps=0.3, min_samples=5)
# fit DBSCAN object to the data
dbscan.fit(X)
# retrieve cluster labels
labels = dbscan.labels_

# plot clusters
plt.scatter(X[:, 0], X[:, 1], c=labels)
plt.show()
```

### f. Spectral Clustering

Spectral clustering adalah metode pengelompokan data yang menggunakan spektrum dari matriks kedekatan antara data. Metode ini bertujuan untuk mempartisi data ke dalam beberapa kelompok berdasarkan struktur internal data. Spektral clustering memanfaatkan pemrosesan matriks dan aljabar linier untuk menghasilkan klaster yang optimal dari data.
Pertama, metode spectral clustering mengkonstruksi matriks kedekatan antara data. Matriks ini dapat dihasilkan dari berbagai metode seperti k-nearest neighbor atau matriks afinitas Gauss. Selanjutnya, matriks tersebut diproses dengan mempertimbangkan nilai eigen (atau spektrum) dari matriks. Nilai eigen dari matriks ini digunakan untuk menghitung vektor embedding atau representasi masing-masing data. Vektor embedding ini kemudian digunakan sebagai input untuk algoritma clustering seperti k-means clustering atau normalized cut.

Kelebihan dari metode spectral clustering adalah mampu mengelompokkan data yang tidak memiliki struktur linear atau bentuk geometris yang jelas. Metode ini juga memiliki
```

kemampuan untuk menangani data yang memiliki dimensi yang tinggi. Selain itu, spectral clustering dapat menghasilkan klaster yang saling terhubung dan saling eksklusif, yang dapat memudahkan analisis data lebih lanjut.

Namun, kelemahan dari metode spectral clustering adalah waktu komputasi yang relatif lebih lama dibandingkan dengan metode clustering lainnya, terutama untuk data yang sangat besar. Selain itu, pemilihan parameter seperti jumlah klaster atau nilai eigen dapat mempengaruhi hasil dari clustering dan harus ditentukan dengan hati-hati.

Contoh koding spectral Clustering

```
from sklearn.cluster import SpectralClustering
from sklearn.datasets import load_iris
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

# load iris dataset
iris = load_iris()
X = iris.data

# reduce dimensionality to 2D for visualization
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X)

# perform spectral clustering with 3 clusters
sc = SpectralClustering(n_clusters=3, affinity='rbf', random_state=0)
y_sc = sc.fit_predict(X)

# visualize clustering result
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=y_sc)
plt.title('Spectral Clustering Result')
plt.show()
```

g. Optics (Ordering Points To Identify the Clustering Structure)

adalah metode clustering yang menggunakan konsep core points dan reachable points untuk mengidentifikasi struktur klaster pada data. Metode ini berbeda dengan metode clustering lainnya yang membutuhkan jumlah klaster yang diinginkan sebelumnya.

Pertama-tama, setiap titik data pada dataset diberi nilai core distance yang menunjukkan jarak minimum ke titik data lain yang memiliki jumlah tetangga (neighborhood) yang cukup. Kemudian, setiap titik data diberi nilai reachability distance yang menunjukkan jarak minimum antara titik tersebut dan core point terdekat.

Proses ini dilakukan secara iteratif hingga semua titik data memiliki nilai core distance dan reachability distance yang sesuai. Setelah itu, nilai reachability distance diurutkan secara menurun dan dibuat plot reachability distance terhadap urutan data. Struktur klaster pada data dapat diidentifikasi dari plot ini sebagai daerah-daerah yang memiliki reachability distance yang rendah.

Optics memiliki keunggulan dalam menangani data dengan jumlah klaster yang tidak diketahui sebelumnya dan data dengan kepadatan yang berbeda-beda. Metode ini juga dapat menghasilkan klaster berbentuk yang tidak teratur.

Berikut adalah contoh source code untuk melakukan Optics clustering pada dataset iris menggunakan bahasa Python dan library Scikit-learn:

```python
from sklearn.cluster import OPTICS
from sklearn.datasets import load_iris
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

# load iris dataset
iris = load_iris()
X = iris.data

# reduce dimensionality to 2D for visualization
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X)

# perform OPTICS clustering
optics = OPTICS(min_samples=5, xi=.05)
optics.fit(X)
reachability = optics.reachability_
labels = optics.labels_

# visualize clustering result
plt.figure(figsize=(10, 6))
plt.plot(reachability, 'k.')
plt.title('OPTICS Reachability Plot')
plt.xlabel('Data Points')
plt.ylabel('Reachability Distance')
plt.show()
```

2.3 Metodologi:

Penelitian ini menggunakan metode analisis klaster pada data ekspresi gen dari pasien kanker payudara menggunakan bahasa pemrograman Phyton. Metode analisis klaster digunakan untuk mengelompokkan pasien kanker payudara berdasarkan ekspresi gen yang berbeda. Berikut adalah langkah-langkah metodologi yang dilakukan dalam penelitian ini:

1. Pengumpulan data: Data ekspresi gen dari pasien kanker payudara diambil dari database publik https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data

2. Pra-pemrosesan data: Data ekspresi gen yang diperoleh diproses untuk menghilangkan data yang tidak penting dan untuk melakukan normalisasi data.

3. Pemrosesan data: Metode analisis klaster yang digunakan adalah Spectral Clustering , DBSCAN (Density-Based Spatial Clustering of Application with Noise)dan K-Means Clustering. Pemrosesan data dilakukan dengan menggunakan program Phyton dan modul Scikit-learn.

4. Pengujian klaster: Pengujian klaster dilakukan dengan menggunakan metode internal dan eksternal. Metode internal digunakan untuk mengevaluasi kualitas klaster yang dihasilkan oleh algoritma.

5. Interpretasi hasil: Hasil analisis klaster dianalisis dan diinterpretasikan untuk mengidentifikasi sub-tipe kanker payudara yang berbeda.

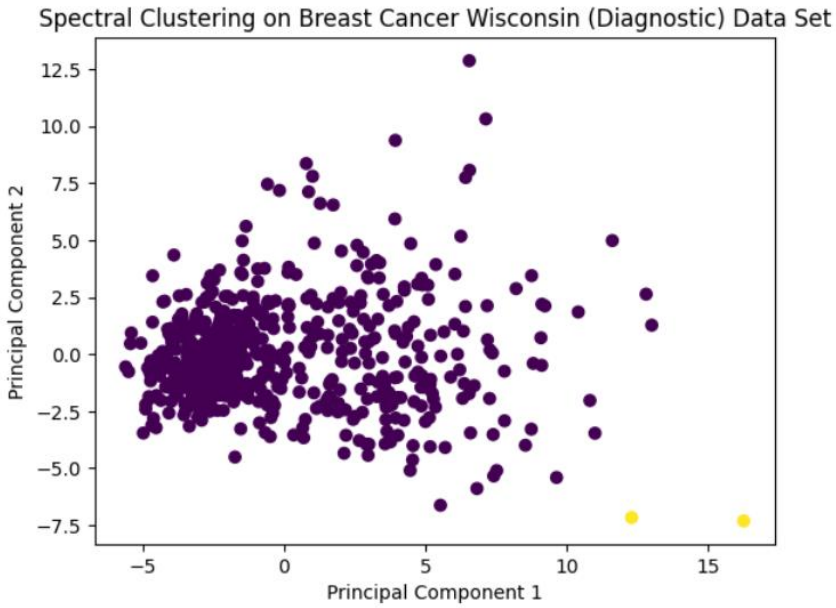
6. Validasi: Validasi dilakukan dengan menggunakan data validasi yang berbeda dari data yang digunakan dalam pengujian.

3. HASIL DAN PEMBAHASAN

Dari beberapa Jenis Model Clustering, ada 3 model yang penulis gunakan untuk klaster

data

yaitu Spectral Clustering , DBSCAN (Density-Based Spatial Clustering of Application with Noise)dan K-Means Clustering. Pemrosesan data dilakukan dengan menggunakan program Phyton dan modul Scikit-learn.



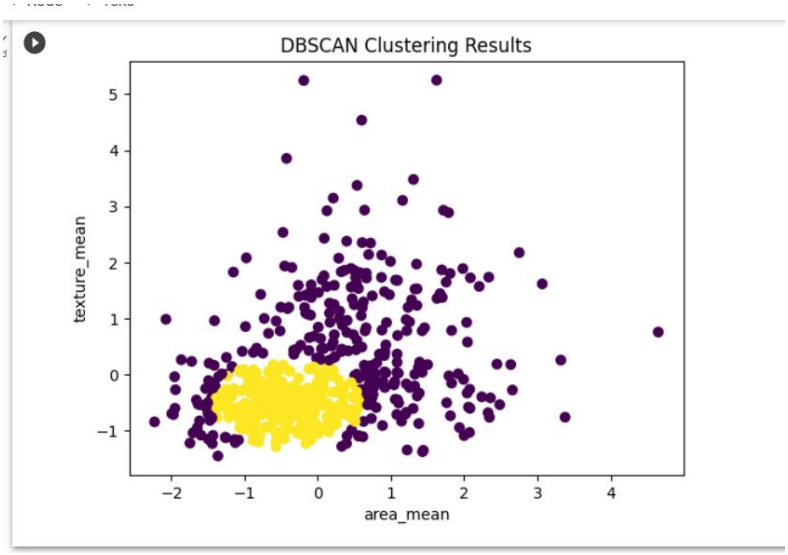
Gambar 1. Visualisasi Data Spectral Clustering

Hasil dari kodingan tersebut adalah sebuah scatter plot yang menunjukkan hasil dari penerapan Spectral Clustering pada dataset Breast Cancer Wisconsin (Diagnostic). Data terlebih dahulu diproses dengan menghapus kolom yang tidak diperlukan, melakukan encoding pada kolom diagnosis, dan melakukan standardisasi pada data. Selanjutnya, dilakukan PCA untuk mengurangi jumlah fitur menjadi 2, sehingga dapat divisualisasikan dengan mudah. Kemudian, Spectral Clustering diterapkan dengan menggunakan jumlah cluster 2, serta parameter affinity dan gamma yang telah ditentukan.

Pada scatter plot hasil visualisasi, sumbu x dan y merepresentasikan Principal Component 1 dan Principal Component 2 hasil PCA. Warna titik-titik pada scatter plot merepresentasikan klaster yang terbentuk dari hasil Spectral Clustering.

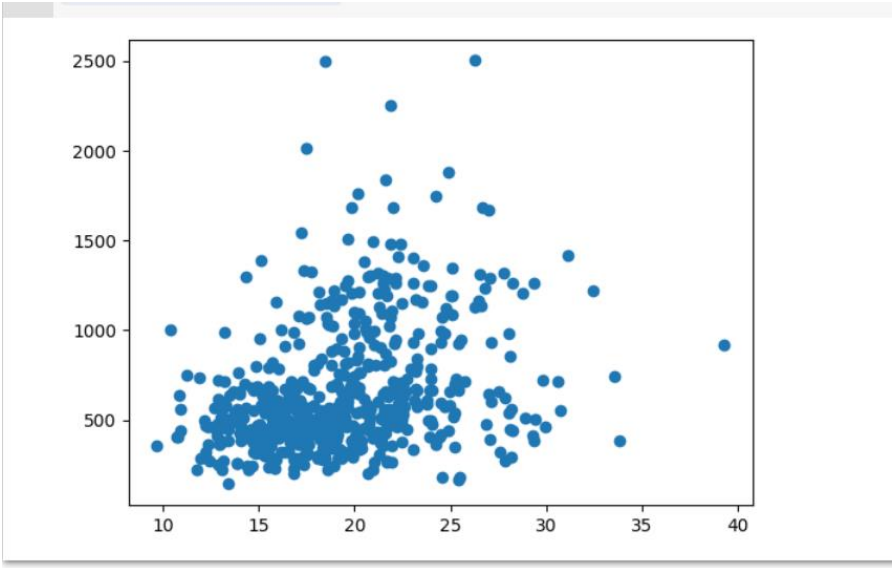
Dari hasil visualisasi tersebut, dapat dilihat bahwa data terbagi menjadi 2 klaster, yang diwakili oleh warna biru dan orange pada scatter plot. Namun, terdapat beberapa titik yang muncul di antara klaster, yang menunjukkan bahwa terdapat beberapa data yang ambigu dan sulit untuk dipisahkan secara jelas antara kedua klaster.
Penerapan Spectral Clustering pada dataset ini dapat membantu dalam memahami pola-pola yang ada dalam data dan mengidentifikasi kelompok-kelompok yang terbentuk

dari data. Selain itu, visualisasi yang dihasilkan juga dapat membantu dalam menginterpretasi hasil clustering dan membuat keputusan berdasarkan analisis data.



Gambar 2. Visualisasi DBSCAN Clustering

Dari hasil clustering menggunakan algoritma DBSCAN pada data cancer breast, dapat dilihat bahwa data point terbagi menjadi beberapa cluster dengan berbagai ukuran. Terdapat cluster besar di bagian tengah plot, dan beberapa cluster kecil di sekitarnya. Hal ini menunjukkan bahwa ada beberapa kelompok data point yang memiliki kesamaan karakteristik pada feature "texture_mean" dan "area_mean". Clustering dapat membantu mengelompokkan data point menjadi beberapa kelompok yang memiliki kemiripan dalam karakteristik tertentu, sehingga dapat digunakan untuk membuat keputusan yang lebih baik dalam berbagai aplikasi data mining.



Gambar 3. Visualisasi Clustering K-Means

Dari visualisasi gambar diatas, dapat dilihat bahwa pengelompokan terbentuk dengan jelas. Hal ini menunjukkan bahwa algoritma K-means mampu memisahkan kelompok sample dengan cukup baik berdasarkan fitur-fitur yang ada, namu Analisa clustering tidak menentukan kausalitas dan hasil diagnosis, melainkan hanya membantu dalam mengelompokkan sample ke dalam kelompok yang homogen berdasarkan fitur-fitur yang ada.Oleh karena itu, hasil ini hanya menjadi awal bagi analisi yang lebih mendalam dalam memahami factor-faktor yang mempengaruhi diagnosis kanker payudara.

4. KESIMPULAN

Berdasarkan hasil diatas dapat disimpulkan bahwa metode analisis klaster dapat membantu dalam mengidentifikasi sub-tipe kanker payudara yang berbeda. Selain itu, penggunaan bahasa pemrograman Phyton dalam melakukan analisis klaster pada data kanker payudara dapat menjadi alternatif yang lebih mudah digunakan dalam pengolahan data kanker payudara.