

UAS Machiene Learning



Muhammad Ridzky Nur Rohim

41155050190082

Informatika A2

Semester 7

Bagian 1

1.

- Regresi logistik adalah teknik analisis data yang menggunakan matematika untuk menemukan hubungan antara dua faktor data. Kemudian menggunakan hubungan ini untuk memprediksi nilai dari salah satu faktor tersebut berdasarkan faktor yang lain. Prediksi biasanya memiliki jumlah hasil yang terbatas, seperti ya atau tidak

Kegunaan : Memodelkan variabel yang tidak diketahui atau tergantung dan variabel yang dikenal atau independen sebagai persamaan linier.

- Regresi logistik adalah teknik analisis data yang menggunakan matematika untuk menemukan hubungan antara dua faktor data. Kemudian menggunakan hubungan ini untuk memprediksi nilai dari salah satu faktor tersebut berdasarkan faktor yang lain. Prediksi biasanya memiliki jumlah hasil yang terbatas, seperti ya atau tidak

Kegunaan : Ilmuwan data menggunakan regresi logistik untuk mengukur probabilitas suatu peristiwa terjadi. Prediksi adalah nilai antara 0 dan 1. Angka 0 menunjukkan suatu peristiwa yang tidak mungkin terjadi dan angka 1 menunjukkan kemungkinan maksimum bahwa peristiwa itu akan terjadi. Persamaan logistik menggunakan fungsi logaritma untuk menghitung garis regresi.

2. Support Vector Machine atau SVM adalah algoritme pembelajaran mesin yang diawasi yang dapat digunakan untuk klasifikasi dan regresi. Cara kerja SVM didasarkan pada SRM atau Structural Risk Minimization yang dirancang untuk mengolah data menjadi Hyperplane yang mengklasifikasikan ruang input menjadi dua kelas.

3. Algoritma K-Nearest Neighbor (KNN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek yang berdasarkan dari data pembelajaran yang jaraknya paling dekat dengan objek tersebut.

Kegunaan : Melakukan klasifikasi terhadap objek dengan data pembelajaran yang jaraknya paling dekat dengan objek tersebut.

4. Naive Bayes adalah algoritma machine learning untuk masalah klasifikasi. Ini didasarkan pada teorema probabilitas Bayes.

Kegunaan : Untuk klasifikasi teks yang melibatkan set data pelatihan dimensi tinggi. Beberapa contohnya adalah penyaringan spam, analisis sentimental, dan klasifikasi artikel berita.

5. Decision tree adalah algoritma machine learning yang menggunakan seperangkat aturan untuk membuat keputusan dengan struktur seperti pohon yang memodelkan kemungkinan hasil, biaya sumber daya, utilitas dan kemungkinan konsekuensi atau resiko.

Kegunaan : Membuat keputusan dengan struktur seperti pohon yang memodelkan kemungkinan hasil, biaya sumber daya, utilitas dan kemungkinan konsekuensi atau resiko.

6. Random Forest adalah algoritma pembelajaran yang supervised. "Forest" yang dibangunnya adalah kumpulan pohon keputusan, biasanya dilatih dengan metode "bagging". Ide umum dari metode bagging adalah kombinasi model pembelajaran meningkatkan hasil keseluruhan.

Kegunaan : Membangun beberapa decision tree dan menggabungkannya untuk mendapatkan prediksi yang lebih akurat dan stabil. Satu keuntungan besar dari Random Forest adalah dapat digunakan untuk masalah klasifikasi dan regresi, yang merupakan mayoritas sistem pembelajaran mesin saat ini.

7. K-Means merupakan algoritma yang umum digunakan untuk clustering dokumen.

Kegunaan K-Means adalah menyusun k prototype atau pusat massa (centroid) dari sekumpulan data berdimensi. Sebelum diterapkan proses algoritma K-means, dokumen akan di preprocessing terlebih dahulu.

8. Agglomerative (metode penggabungan) adalah strategi pengelompokan hirarki yang dimulai dengan setiap objek dalam satu cluster yang terpisah kemudian membentuk cluster yang semakin membesar.

Algoritma clustering ini adalah algoritma berbasis centroid, sehingga tujuan dari algoritma ini adalah untuk menemukan titik pusat dari setiap cluster. Berbeda dengan algoritma sebelumnya, algoritma ini tidak perlu memilih jumlah cluster karena algoritma ini secara otomatis dapat menemukan jumlah cluster optimal.

9. Algoritma Apriori adalah algoritma yang digunakan untuk menghitung aturan asosiasi antar objek. Aturan asosiasi menjelaskan bagaimana dua atau lebih objek terkait satu sama lain. Dengan kata lain, algoritma apriori adalah algoritma berbasis aturan asosiasi yang menganalisis apakah orang yang membeli produk A juga membeli produk B.

Kegunaan : Untuk menentukan itemset yang paling sering muncul, algoritma Apriori menggunakan pendekatan "bottom-up". Pendekatan ini menggunakan Breadth-first search dan struktur data Hash tree untuk menghitung kandidat itemset secara efisien.

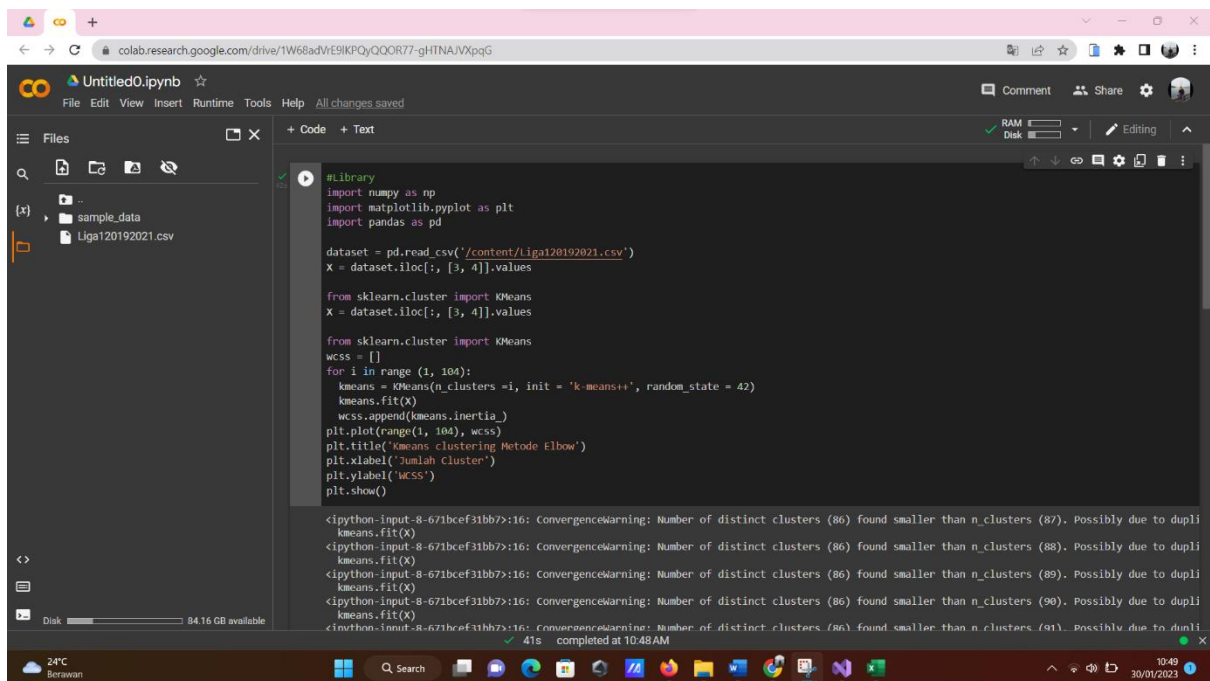
Pendekatan bottom up dimulai dari setiap item dalam daftar itemset. Kemudian, kandidat dibentuk dengan self-join (penggabungan). Setiap iterasi, kita memperluas panjang itemset satu item yang membentuk subhimpunan. Setelah itu dilakukan proses pengujian terhadap subhimpunan. Apabila ditemukan itemset yang berisi subhimpunan yang jarang, maka dilakukan pruning (pemangkasan). Proses dilakukan secara berulang-ulang sampai tidak ada lagi itemset yang berhasil diturunkan dari data.

10. Self-organizing maps (SOM) merupakan suatu jenis artificial neural network yang dilatih dengan metode unsupervised learning. Jaringan ini mampu menghasilkan sebuah representasi terpisah atas ruang input sampel pelatihan dengan dimensi rendah (biasanya dua dimensi).

Kegunaan : Mempertahankan informasi struktural dari data pelatihan. Selain itu, SOM juga menghasilkan data yang tidak linier secara inheren.

Bagian 2

Clustering data *Liga120192021* Menggunakan Metode *K-Means*



The screenshot shows a Google Colab notebook titled 'Untitled0.ipynb'. The left sidebar displays the file explorer with a folder named 'sample_data' containing a file 'Liga120192021.csv'. The main code cell contains the following Python code:

```
#Library
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

dataset = pd.read_csv('/content/Liga120192021.csv')
x = dataset.iloc[:, [3, 4]].values

from sklearn.cluster import KMeans
x = dataset.iloc[:, [3, 4]].values

from sklearn.cluster import KMeans
wcss = []
for i in range(1, 104):
    kmeans = KMeans(n_clusters=i, init='k-means++', random_state=42)
    kmeans.fit(x)
    wcss.append(kmeans.inertia_)
plt.plot(range(1, 104), wcss)
plt.title('Kmeans clustering Metode Elbow')
plt.xlabel('Jumlah Cluster')
plt.ylabel('WCSS')
plt.show()
```

The output of the code cell shows several 'ConvergenceWarning' messages, indicating that the number of distinct clusters found (86) is smaller than the specified number of clusters (n_clusters). The warnings are repeated for different values of n_clusters (87, 88, 89, 90, 91). The final output shows the execution completed at 10:48 AM.

colab.research.google.com/drive/1W68adVrE9IKPQyQQOR77-gHTNAJVXpqG

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

RAM 100% Disk 84.16 GB available

Files

- sample_data
- Liga120192021.csv

```
kmeans = KMeans(n_clusters=2, init='k-means++', random_state=42)
y_kmeans = kmeans.fit_predict(X)

plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 10, c = 'red', label = 'kluster 1')
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 10, c = 'blue', label = 'kluster 2')
plt.scatter(kmeans.cluster_centers_[0, 0], kmeans.cluster_centers_[0, 1], s = 300, c = 'yellow', label = 'centroid')

plt.title('Cluster Liga 1')
plt.xlabel('Skor Liga 1')
plt.ylabel('Skor Liga 1 2019 2021')
plt.legend()
plt.show()
```

<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (87). Possibly due to duplicate poi
kmeans.fit(X)
<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (88). Possibly due to duplicate poi
kmeans.fit(X)
<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (89). Possibly due to duplicate poi
kmeans.fit(X)
<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (90). Possibly due to duplicate poi
kmeans.fit(X)
<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (91). Possibly due to duplicate poi
kmeans.fit(X)
<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (92). Possibly due to duplicate poi
kmeans.fit(X)
<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (93). Possibly due to duplicate poi
kmeans.fit(X)
<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (94). Possibly due to duplicate poi
kmeans.fit(X)
<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (95). Possibly due to duplicate poi
kmeans.fit(X)
<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (96). Possibly due to duplicate poi
kmeans.fit(X)
<ipython-input-16-acdef4faeaba>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (97). Possibly due to duplicate poi
kmeans.fit(X)

39s completed at 11:13 AM

24°C Berawan

Files

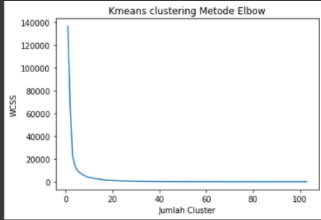
- sample_data
- Liga120192021.csv

```
kmeans.fit(X)
<ipython-input-8-671bcef31bb7>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (98). Possibly due to dupli
kmeans.fit(X)
<ipython-input-8-671bcef31bb7>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (99). Possibly due to dupli
kmeans.fit(X)
<ipython-input-8-671bcef31bb7>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (100). Possibly due to dupl
kmeans.fit(X)
<ipython-input-8-671bcef31bb7>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (101). Possibly due to dupl
kmeans.fit(X)
<ipython-input-8-671bcef31bb7>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (102). Possibly due to dupl
kmeans.fit(X)
<ipython-input-8-671bcef31bb7>:16: ConvergenceWarning: Number of distinct clusters (86) found smaller than n_clusters (103). Possibly due to dupl
kmeans.fit(X)
```

41s completed at 10:48 AM

24°C Berawan

Kmeans clustering Metode Elbow



Jumlah Cluster	WCSS (approx.)
2	135000
5	100000
10	60000
20	20000
40	10000
60	5000
80	2000
100	1000

