

UJIAN AKHIR SEMESTER

Machine Learning



DISUSUN OLEH:

Zhidan Pratama S

41155050190041

KELAS KARYAWAN C

FAKULTAS TEKNIK

JURUSAN TEKNIK INFORMATIKA

UNIVERSITAS LANGLANGBUANA

2023

Bagian 1

1. Apa itu *Linear dan Logistic Regresion* dan apa gunanya?

Jawab

Linear dan logistic regression merupakan metode pendekatan untuk pemodelan hubungan antara satu variabel dependen dan satu variabel independen. Kegunaan linear dan logictic regresion adalah untuk mengetahui variabel-variabel kunci yang memiliki pengaruh terhadap suatu variabel bergantung, pemodelan, serta pendugaan (estimation) atau peramalan (forecasting). Keduanya sangatlah mirip. Hanya sedikit perbedaan keduanya tapi krusial. Linear menyelesaikan masalah regresi (variabel keluarannya kontinu) sedangkan regresi logistik menyelesaikan klasifikasi (variabel keluarannya diskret). Dan juga Solusi regresi linear dapat ditemukan secara analitik sedangkan regresi logistik tidak.

2. Apa itu *Support Vector Machine* dan apa gunanya?

Jawab

Pada dasarnya, support-vector machine adalah sebuah algoritma klasifikasi untuk data linear dan non-linear. Support Vector Machine atau SVM juga bisa di katakan sebagai algoritme pembelajaran mesin yang diawasi dan dapat digunakan untuk klasifikasi dan regresi. SVM ini berguna untuk menentukan hyperplane atau pemisah dengan memilih bidang yang memiliki optimal margin, dengan generalisasi pada SVM yang dapat terjaga dengan sendirinya, tingkat generalisasi pada SVM juga tidak dipengaruhi oleh jumlah data latih. juga dengan SVM kita dapat menentukan parameter soft margin, dan noise dapat dikontrol pada kesalahan klasifikasi sehingga proses pelatihan menjadi jauh lebih ketat.

3. Apa itu *K-Nearest Neighbor* dan apa gunanya?

Jawab

K-Nearest Neighbor merupakan algoritma yang melakukan klasifikasi berdasarkan kedekatan jarak suatu data dengan data yang lain. Dekat atau jauh suatu jarak dihitung berdasarkan jarak Euclidean. KNN merupakan salah satu algoritma non parametrik yang digunakan dalam pengklasifikasian. K-Nearest Neighbor biasanya digunakan untuk menghitung jarak antar-instance, menghitung jarak x dengan semua instance lainnya berdasarkan fungsi tersebut dan menentukan kelas x sebagai kelas yang paling banyak muncul di k-instance.

4. Apa itu *Naive Bayes* dan apa gunanya?

Jawaban

Algoritma Naive Bayes adalah algoritma yang mempelajari probabilitas suatu objek dengan ciri-ciri tertentu yang termasuk dalam kelompok atau kelas tertentu. Fungsi dari

Naive Bayes yaitu untuk mengidentifikasi sesuatu dengan menggunakan ciri-ciri dari sebuah objek dengan data statistic dan filsuf.

5. Apa itu *Decision Tree* dan apa gunanya?

Jawaban

Decision Tree merupakan metode klasifikasi menjadi salah satu yang terpopuler karena mudah dipahami. Decision Tree adalah metode pengambilan keputusan yang menyusun setiap opsi menjadi bentuk yang bercabang. Dengan begitu membuat metode ini disebut pohon keputusan. Decision tree adalah metode yang biasa dilakukan untuk mengambil keputusan-keputusan informal atau sederhana.

6. Apa itu *Random Forest* dan apa gunanya?

Jawaban

Random forest adalah suatu algoritma yang digunakan pada klasifikasi data dalam jumlah yang besar. Klasifikasi random forest dilakukan melalui penggabungan pohon dengan melakukan training pada sampel data yang dimiliki. Kegunaan dari Random Forest ini yaitu untuk melakukan penerapan dari pendekatan diskriminasi stokastik pada klasifikasi, random forest juga digunakan untuk pengklasifikasian pada atribut yang tidak lengkap, dan mampu melakukan pengklasifikasian data serta dapat digunakan untuk menangani data sampel yang banyak.

7. Apa itu *K-Means* dan apa gunanya?

Jawaban

K-means merupakan salah satu algoritma yang bersifat unsupervised learning. K-Means memiliki kegunaan untuk mengelompokkan data kedalam data cluster. Pengklasteran k-means adalah algoritme untuk membagi n pengamatan menjadi k kelompok sedemikian hingga tiap pengamatan termasuk ke dalam kelompok dengan rata-rata terdekat.

8. Apa itu *Agglomerate Clustering* dan apa gunanya?

Jawab

Agglomerative (metode penggabungan) adalah strategi pengelompokan hirarki yang dimulai dengan setiap objek dalam satu cluster yang terpisah kemudian membentuk cluster yang semakin membesar. Adapun tujuan dari agglomerate Clustering adalah bertujuan untuk mengelompokkan objek-objek berdasarkan karakteristik yang dimilikinya, yang dimulai dengan objek-objek individual sampai objek-objek tersebut bergabung menjadi satu cluster tunggal.

9. Apa itu *Apriori Algorithm* dan apa gunanya?

Jawab

Algoritma apriori adalah proses ekstraksi informasi dari suatu database, dilanjutkan dengan melakukan frequent item/itemset dan candidate generation dalam pembentukan asosiasi rule mining guna mendapatkan hasil nilai minimum support dan hasil nilai minimum confidence. Adapun kegunaan dari apriori algorithm yaitu untuk menghitung aturan asosiasi antar objek. Aturan asosiasi menjelaskan bagaimana dua atau lebih objek terkait satu sama lain.

10. Apa itu *Self Organizing Map* dan apa gunanya?

Jawaban

Self Organizing Map (SOM) merupakan metode untuk membagi pola masukan ke dalam beberapa kelompok cluster. Kegunaan dari self organizing map ini adalah untuk melakukan representasi terpisah atas ruang *input* sampel pelatihan dengan dimensi rendah (biasanya dua dimensi).

Bagian 2

1. Pemodelan data machine learning pada data Liga120192021

A	B	C	D	E	F	G	H	I	J	K
Tanding	Pass1	Pass2	Pass3	Pass4	Pass5	Pass6	Pass7	Pass8	Pass9	Pass10
0	11	24	2	20	10	11	13	11	16	71
1	10	11	13	11	20	12	13	20	77	71
2	16	8	16	17	21	22	3	20	10	13
3	22	16	8	16	2	17	23	8	82	4
4	20	12	16	8	16	17	21	23	22	13
5	20	11	22	17	21	8	77	13	91	10
6	16	17	21	22	21	93	91	11	3	10
7	20	11	22	78	2	77	8	17	23	71
8	24	11	3	20	12	93	91	10	8	13
9	11	22	78	2	77	20	12	93	4	13
10	22	16	2	17	23	8	78	2	12	10
11	22	21	93	91	11	12	93	91	13	9
12	21	8	77	13	91	10	20	93	77	16
13	8	16	17	21	22	24	11	3	10	77
14	16	11	12	93	91	93	77	16	13	9
15	1	9	20	24	9	10	93	77	22	10
16	8	22	16	2	17	23	11	22	10	77
17	2	77	20	12	3	20	12	10	12	93
18	17	23	8	93	20	13	11	23	77	2
19	12	3	20	12	22	78	2	10	9	77
20	1	2	1	9	20	24	9	10	20	10
21	24	11	3	20	12	13	20	13	11	10
22	24	11	20	11	22	78	2	77	8	13
23	22	16	8	16	17	21	22	21	93	23
24	22	8	22	16	2	17	23	8	93	91
25	21	22	24	11	3	8	10	23	91	21
26	93	91	93	77	16	2	77	20	12	13
27	24	9	10	93	77	17	23	8	93	13
28	2	17	23	11	22	12	3	20	12	16
29	12	3	20	12	10	16	8	16	17	20
30	11	22	12	3	20	20	12	10	16	13
31	77	16	2	77	20	12	12	10	16	21
32	93	77	17	23	8	93	12	16	3	10
33	2	77	20	12	12	10	16	10	3	20
34	20	11	22	78	11	22	12	8	13	10
35	9	10	93	77	17	20	13	11	91	10
36	17	23	8	93	10	23	91	10	93	13
37	20	12	13	20	13	11	22	21	93	23
38	11	22	78	2	77	8	23	11	91	21
39	16	17	21	22	21	93	20	12	10	77
40	16	2	17	23	8	93	12	3	91	10

A	B	C	D	E	F	G	H	I	J	K
41	11	3	8	10	23	91	2	77	13	2
42	77	16	2	77	20	12	17	23	11	10
43	93	77	17	23	8	93	16	8	10	9
44	11	22	12	3	20	12	20	12	22	10
45	17	20	13	11	8	93	12	77	13	10
46	22	12	8	13	20	12	93	12	91	24
47	78	11	22	12	8	13	10	16	24	10
48	17	23	8	93	16	20	22	12	91	10
49	3	20	2	14	32	91	10	91	9	20
50	91	23	2	20	2	91	23	91	20	9
51	2	32	9	2	14	2	14	22	21	9
52	14	32	3	23	14	23	22	23	20	9
53	14	32	9	10	11	91	11	10	11	9
54	3	10	3	91	23	91	23	91	9	20
55	32	78	2	9	10	13	22	11	22	9
56	30	23	7	32	3	10	74	10	74	23
57	32	3	10	2	9	30	2	23	10	74
58	3	23	32	9	22	1	9	3	7	74
59	11	30	12	2	16	2	30	11	2	13
60	30	2	27	14	13	16	10	5	9	30
61	21	27	74	14	16	30	2	74	21	9
62	23	21	23	20	9	27	9	30	27	74
63	9	21	30	74	9	16	9	12	20	74
64	10	2	30	23	12	9	12	9	23	9
65	30	74	13	30	12	13	12	23	2	13
66	10	11	13	2	32	2	30	74	10	9
67	13	11	74	27	11	74	27	30	27	9
68	21	27	13	21	8	21	8	11	12	20
69	12	32	3	12	93	12	14	9	74	20
70	23	2	32	2	74	93	23	32	9	93
71	9	3	32	23	12	74	93	74	12	74
72	23	74	22	2	14	16	74	2	9	23
73	16	2	22	2	16	74	21	20	9	20
74	13	74	2	13	2	14	16	13	21	9
75	4	74	23	22	23	13	22	74	20	9
76	3	11	7	11	7	12	23	2	9	21
77	2	9	12	9	13	12	9	13	11	21
78	25	13	74	23	14	23	25	13	37	37
79	23	74	37	23	32	32	23	2	7	74
80	10	21	8	93	21	93	21	74	21	5
81	93	37	3	37	30	21	32	27	37	93
82	32	21	27	21	14	37	21	23	30	25
A	B	C	D	E	F	G	H	I	J	K
83	3	21	7	11	16	7	21	3	11	23
84	3	7	12	13	11	13	11	13	25	7
85	13	2	7	13	12	2	11	7	21	7
86	21	3	27	3	27	3	30	13	37	27
87	74	7	74	25	21	7	74	25	23	25
88	2	14	25	37	93	25	37	21	23	25
89	74	25	12	2	30	23	12	21	93	21
90	25	30	23	3	74	23	2	16	23	93
91	78	93	2	74	2	93	30	25	23	25
92	30	32	3	37	23	13	23	2	21	37
93	13	30	23	3	27	3	30	13	74	37
94	2	7	93	12	93	2	74	7	30	37
95	37	74	2	12	93	2	37	3	30	37
96	11	12	11	12	2	30	74	30	21	27
97	19	10	93	10	4	19	10	66	21	19
98	53	77	10	66	10	55	66	55	11	10
99	30	22	23	22	74	23	12	23	13	7
100	25	27	74	93	27	11	93	74	27	25
101	27	7	27	7	25	12	27	13	21	7
102	13	11	23	2	23	12	11	13	21	25

- Metode yang akan digunakan yaitu metode Logistic Regression, logictic regresion digunakan untuk mengetahui variabel-variabel kunci yang memiliki pengaruh terhadap suatu variabel bergantung, pemodelan, serta pendugaan (estimation) atau peramalan (forecasting).

Tahap pertama kita akan melakukan import libraries

ZHIDAN PRATAMA 41155050190041

Import Library

```
In [1]: import pandas as pd
import numpy as np
from scipy import stats
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.cluster import KMeans
from sklearn.metrics import confusion_matrix
```

Tahapan ke dua kita akan melakukan Load Data yang sudah kita upload sebelumnya.

Load Data Liga120192021.csv

```
In [6]: df = pd.read_csv("input/Liga120192021.csv")
```

```
In [7]: df
```

```
Out[7]:
```

	Tanding	Pass1	Pass2	Pass3	Pass4	Pass5	Pass6	Pass7	Pass8	Pass9	Pass10
0	0	11	24	2	20	10	11	13	11	16	71
1	1	10	11	13	11	20	12	13	20	77	71
2	2	16	8	16	17	21	22	3	20	10	13
3	3	22	16	8	16	2	17	23	8	82	4
4	4	20	12	16	8	16	17	21	23	22	13
...
98	98	53	77	10	66	10	55	66	55	11	10
99	99	30	22	23	22	74	23	12	23	13	7
100	100	25	27	74	93	27	11	93	74	27	25
101	101	27	7	27	7	25	12	27	13	21	7
102	102	13	11	23	2	23	12	11	13	21	25

103 rows × 11 columns

Kemudian kita akan mencari kolom apasaja yang ada pada data tersebut, dan kita akan melakukan pencarian modus pada setiap kolom.

```
In [9]: df.columns
```

```
Out[9]: Index(['Tanding', 'Pass1', 'Pass2', 'Pass3', 'Pass4', 'Pass5', 'Pass6',
'Pass7', 'Pass8', 'Pass9', 'Pass10'],
dtype='object')
```

Mendapatkan modus dari tiap kolom

```
In [12]: df = pd.read_csv("input/Liga120192021.csv")
x = stats.mode(df, keepdims=True)
print(x)
```

```
ModeResult(mode=array([[ 0, 11, 11,  2,  2, 20, 12, 12, 10, 21, 10]], dtype=int64), count=array([[ 1,  8, 13, 10, 12,  8, 12, 1
2, 10, 10, 16]]))
```

3. Model perkolom

Pada pengklasifikasian logistic regression ini kita akan menggunakan data yang ada pada kolom pass 1 terhadap kolom pass2, dan Kolom pass 9 terhadap Kolom Pass 10

Penklasifikasian Data Pada Kolom Pass1 Terhadap Kolom Pass2

```
In [16]: models = {}
for col in df.columns[1:]:
    x = df['Pass1'].values.reshape(-1, 1)
    y = df[col].values
    x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.33, random_state=42)
    model = LogisticRegression(max_iter=5000).fit(x_train, y_train)
    models[col] = model
    print("Accuracy for", col, ":", model.score(x_test, y_test))
```

Accuracy for Pass1 : 0.08823529411764706
 Accuracy for Pass2 : 0.08823529411764706
 Accuracy for Pass3 : 0.08823529411764706
 Accuracy for Pass4 : 0.08823529411764706
 Accuracy for Pass5 : 0.08823529411764706
 Accuracy for Pass6 : 0.08823529411764706
 Accuracy for Pass7 : 0.08823529411764706
 Accuracy for Pass8 : 0.08823529411764706
 Accuracy for Pass9 : 0.08823529411764706
 Accuracy for Pass10 : 0.08823529411764706

Pengklasifikasian Data Pada Kolom Pass9 Terhadap Kolom Pass 10

```
In [17]: models = {}
for col in df.columns[1:]:
    x = df['Pass9'].values.reshape(-1, 1)
    y = df['Pass10'].values
    x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.33, random_state=42)
    model = LogisticRegression(max_iter=5000).fit(x_train, y_train)
    models[col] = model
    print("Accuracy for", col, ":", model.score(x_test, y_test))

Accuracy for Pass1 : 0.11764705882352941
Accuracy for Pass2 : 0.11764705882352941
Accuracy for Pass3 : 0.11764705882352941
Accuracy for Pass4 : 0.11764705882352941
Accuracy for Pass5 : 0.11764705882352941
Accuracy for Pass6 : 0.11764705882352941
Accuracy for Pass7 : 0.11764705882352941
Accuracy for Pass8 : 0.11764705882352941
Accuracy for Pass9 : 0.11764705882352941
Accuracy for Pass10 : 0.11764705882352941
```

Confusion Matrix pada Pass 1 dan Pass 9

[illegible]

Confusion Matrix pada Pass 9

```
In [19]: x = df.iloc[:, 1:].values
         y = df['Pass9'].values
         x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.33, random_state=42)
         model = LogisticRegression(max_iter=10000).fit(x_train, y_train)
         y_pred = model.predict(x_test)
         print("\nConfusion Matrix:")
         print(confusion_matrix(y_test, y_pred))
```

[illegible]

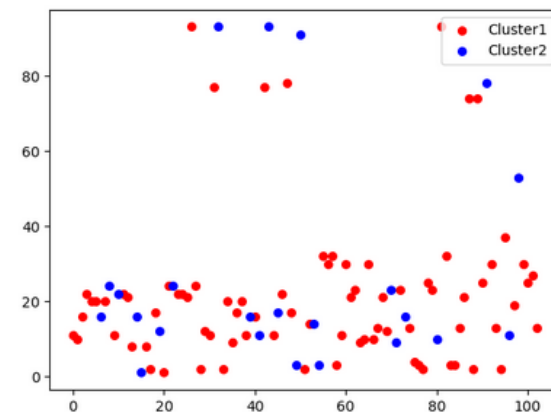
Clustering 2 Kolom

Cluster 2 model kolom

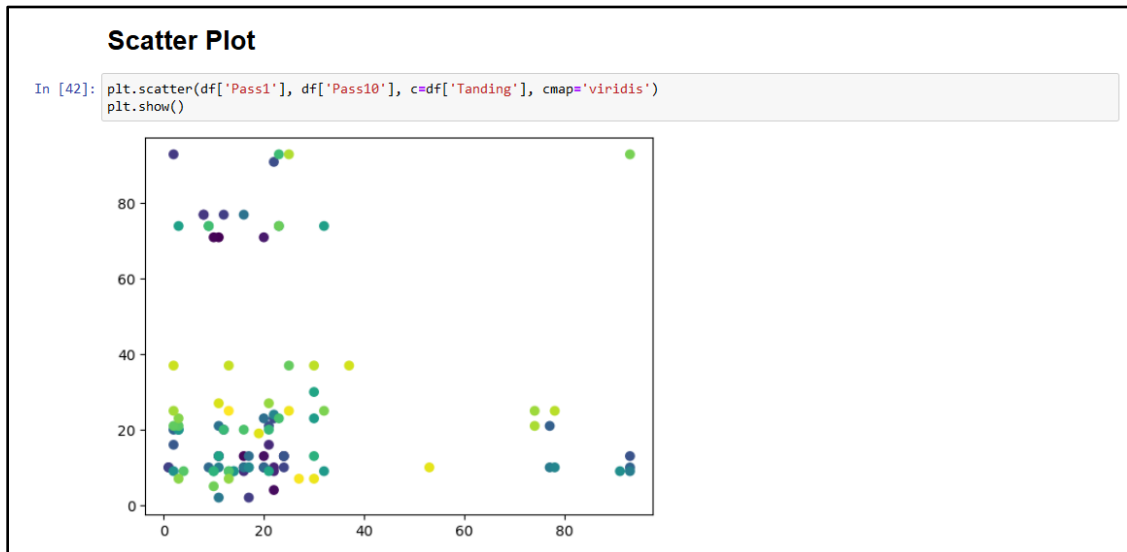
```
In [21]: x = df.iloc[:, 1:].values
kmeans = KMeans(n_clusters=2, random_state=0).fit(x)
print("\nCluster Centers:")
print(kmeans.cluster_centers_)
```

```
Cluster Centers:
[[21.62820513 23.97435897 23.23076923 22.71794872 24.15384615 14.94871795
 25.33333333 19.62820513 34.12820513 27.35897436]
 [28.72      24.08      22.16      38.16      19.92      80.44
 35.16      43.68      16.2      25.56      ]]
```

```
In [30]: from sklearn.cluster import AgglomerativeClustering
import matplotlib.pyplot as plt
x = df.iloc[:, :-1].values
k = 2
agg_clustering = AgglomerativeClustering(n_clusters=k)
y_pred = agg_clustering.fit_predict(x)
colors = ['red', 'blue']
for i in range(k):
    plt.scatter(x[y_pred == i, 0], x[y_pred == i, 1], s=30, c=colors[i], label='Cluster' + str(i+1))
plt.legend()
plt.show()
```



4. Scatter Plot



5. Berikut dilampirkan Link Github

<https://github.com/zhidanpratama/Tugas-Machine-Learning>