

Pokok Bahasan V K-Nearest Neighbor (KNN).

Kode Pokok Bahasan: TIK.RPL03.004.00.01

Deskripsi Pokok Bahasan:

Membahas bagaimana penerapan K-Nearest Neighbor untuk melakukan Klasifikasi.

No	Elemen Kompetensi	Indikator Kinerja	Jml Jam	Hal
1.	Menampilkan hasil Klasifikasi dari kasus yang diberikan.	Mampu melakukan analisis terhadap klasifikasi data dari diagram yang muncul.	1	
2.	Melakukan visualisasi hasil KNN menggunakan library ggplot.	Mampu melakukan visualisasi hasil KNN menggunakan library ggplot pada dataset yang ditentukan	1	

TUGAS PENDAHULUAN

Hal yang harus dilakukan dan acuan yang harus dibaca sebelum praktikum :

1. Menginstal R pada PC masing-masing praktikan.
2. Menginstal R Studio pada PC masing-masing praktikan.
3. Menginstal aplikasi pengolah data (Excel).

DAFTAR PERTANYAAN

1. Apa itu algoritma K-Nearest Neighbor?
2. Apa kegunaan K-Nearest Neighbor?
3. Sebutkan tahapan dari proses algoritma K-Nearest Neighbor!

1. sebuah algoritma untuk melakukan klasifikasi terhadap objek dengan data pembe- lajaran yang jaraknya paling dekat dengan objek tersebut.

2. digunakan untuk proses yang disebut missing data imputation yang memperkirakan nilai-nilai yang hilang.

3. Tentukan jumlah tetangga (K) yang akan digunakan untuk pertimbangan penentuan kelas. Hitung jarak dari data baru ke masing-masing data point di dataset. Ambil sejumlah K data dengan jarak terdekat, kemudian tentukan kelas dari data baru tersebut.

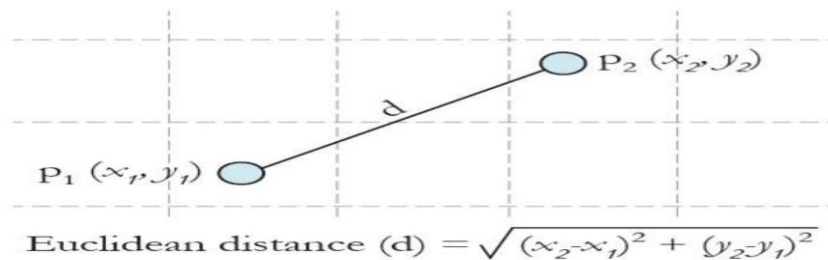


TEORI SINGKAT

Algoritma Nearest Neighbor Retrieval (K-Nearest Neighbor atau K-NN) adalah sebuah algoritma untuk melakukan klasifikasi terhadap objek dengan data pembe- lajaran yang jaraknya paling dekat dengan objek tersebut.

Alasan pemilihan Algoritma KNN adalah berdasarkan data yang digunakan yaitu menggunakan data sekunder dan tujuan dari algoritma ini adalah mengklasifikasikan obyek baru berdasarkan atribut dan training sample, tidak memerlukan model algoritma seperti yang dihasilkan oleh algoritma lain.

KNN, Dapat digunakan untuk tujuan klasifikasi, Tidak menyusun model atau mengekstrak aturan logika tertentu sebagai hasil dari analisis, Identifikasi k buah individu tetangga terdekat dilakukan dengan terlebih dahulu menghitung jarak dari individu yang akan diduga dengan setiap individu yang ada pada gugus data training. Jika ini sudah dilakukan maka tinggal mencari k buah amatan yang jaraknya paling kecil. Penghitungan jarak dari dua amatan A dan B dapat menggunakan formula Euclid distance.



LAB SETUP

Hal yang harus disiapkan dan dilakukan oleh praktikan untuk menjalankan praktikum modul ini.

1. Menginstall library yang dibutuhkan untuk mengerjakan modul.
2. Menjalankan R Studio.
3. Aplikasi pengolah data (Excel)

ELEMEN KOMPETENSI I

Deskripsi:

Menampilkan hasil klasifikasi dari kasus yang diberikan.

Kompetensi Dasar:

Menampilkan hasil Klasifikasi dari kasus yang diberikan.

Latihan 1.1.1

Penjelasan Singkat :

Pada latihan ini anda akan diminta melakukan analisis terhadap klasifikasi data dari diagram yang muncul.



Langkah-Langkah Praktikum:

1. Siapkan data : glass.csv (bisa diunduh pada link : <https://www.kaggle.com/datasets/uciml/glass>)
2. Buka Rstudio
3. Melakukan data preparation

```
> dataKNNGading = read.csv("C:/Users/Section/Downloads/glass.csv")
> data.training <- dataKNNGading[,1:9]
> kelas <- as.factor(dataKNNGading[,10])
> View(dataKNNGading)
> str(dataKNNGading)
```

Output:

	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe	Type
1	1.52101	13.64	4.49	1.10	71.78	0.06	8.75	0.00	0.00	1
2	1.51761	13.89	3.60	1.36	72.73	0.48	7.83	0.00	0.00	1
3	1.51618	13.53	3.55	1.54	72.99	0.39	7.78	0.00	0.00	1
4	1.51766	13.21	3.69	1.29	72.61	0.57	8.22	0.00	0.00	1
5	1.51742	13.27	3.62	1.24	73.08	0.55	8.07	0.00	0.00	1
6	1.51596	12.79	3.61	1.62	72.97	0.64	8.07	0.00	0.26	1
7	1.51743	13.30	3.60	1.14	73.09	0.58	8.17	0.00	0.00	1
8	1.51756	13.15	3.61	1.05	73.24	0.57	8.24	0.00	0.00	1
9	1.51918	14.04	3.58	1.37	72.08	0.56	8.30	0.00	0.00	1
10	1.51755	13.00	3.60	1.36	72.99	0.57	8.40	0.00	0.11	1
11	1.51571	12.72	3.46	1.56	73.20	0.67	8.09	0.00	0.24	1
12	1.51763	12.80	3.66	1.27	73.01	0.60	8.56	0.00	0.00	1
13	1.51589	12.88	3.43	1.40	73.28	0.69	8.05	0.00	0.24	1
14	1.51748	12.86	3.56	1.27	73.21	0.54	8.38	0.00	0.17	1

```
> str(dataKNNibnu)
'data.frame': 214 obs. of 10 variables:
 $ RI : num 1.52 1.52 1.52 1.52 1.52 ...
 $ Na : num 13.6 13.9 13.5 13.2 13.3 ...
 $ Mg : num 4.49 3.6 3.55 3.69 3.62 3.61 3.6 3.61 3.58 3.6 ...
 $ Al : num 1.1 1.36 1.54 1.29 1.24 1.62 1.14 1.05 1.37 1.36 ..
 $ Si : num 71.8 72.7 73 72.6 73.1 ...
 $ K : num 0.06 0.48 0.39 0.57 0.55 0.64 0.58 0.57 0.56 0.57 .
 $ Ca : num 8.75 7.83 7.78 8.22 8.07 8.07 8.17 8.24 8.3 8.4 ...
 $ Ba : num 0 0 0 0 0 0 0 0 0 0 ...
 $ Fe : num 0 0 0 0 0 0.26 0 0 0 0.11 ...
 $ Type: int 1 1 1 1 1 1 1 1 1 1 ...
```

4. Membuat data amatan dengan knn

```
> kelas <- as.factor(dataKNNGading[,10])
> amatan.baru <- c(1.51, 14.12, 1.78, 1.79, 73.1, 0.0, 8.70, 0.76,
0.0)
> library(class)
> knn(data.training, amatan.baru, kelas, k=1)
```

Output:

```
> kelas <- as.factor(dataKNNibnu[,10])
> amatan.baru <- c(1.51, 14.12, 1.78, 1.79, 73.1, 0.0, 8.70, 0.76, 0.0)
> library(class)
> knn(data.training, amatan.baru, kelas, k=1)
[1] 7
Levels: 1 2 3 5 6 7
```

5. Membuat data amatan baru

```
> amatan.baru2 <- c(1.51, 13.12, 1.78, 1.0, 73.1, 0.2, 7.70, 0.76, 0.0)
> knn(data.training, amatan.baru2, kelas, k=5)
```

Output:

```
> amatan.baru2 <- c(1.51, 13.12, 1.78, 1.0, 73.1, 0.2, 7.70, 0.76, 0.0)
> knn(data.training, amatan.baru2, kelas, k=5)
[1] 2
Levels: 1 2 3 5 6 7
```

ELEMEN KOMPETENSI II

Deskripsi:

Menampilkan hasil klasifikasi dari kasus yang diberikan berdasarkan dataset iris.

Kompetensi Dasar:

Menampilkan hasil Klasifikasi dari kasus yang diberikan.

Latihan 1.2.1

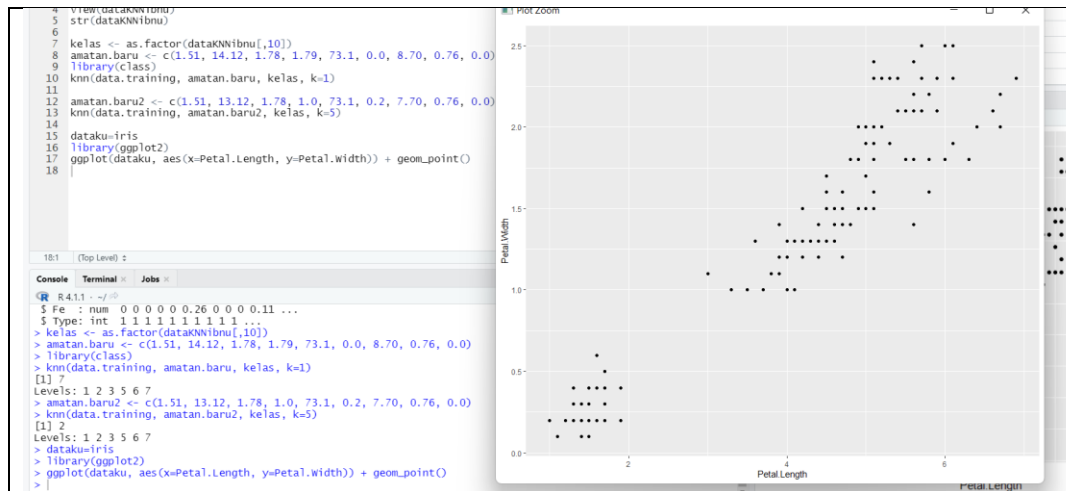
Penjelasan Singkat :

Pada latihan ini anda akan diminta melakukan analisis terhadap klasifikasi data dari diagram yang muncul.

1. Membuat plot sebelum melakukan Proses KNN

```
> dataku=iris
> library(ggplot2)
> ggplot(dataku, aes(x=Petal.Length, y=Petal.Width)) +
  geom_point()
```

Output:



2. Melakukan pengamatan pada data

```

> kelas <- as.factor(dataku[,4])
> pengamatan.baru <- c(3.8, 1.5)
> library(class)
> data.training <- dataku[,3:4]
> kelas <- as.factor(dataku[,5])
> knn(data.training, pengamatan.baru, kelas, k=5)

```

Output:

```

> kelas <- as.factor(dataku[,4])
> pengamatan.baru <- c(3.8, 1.5)
> library(class)
> data.training <- dataku[,3:4]
> kelas <- as.factor(dataku[,5])
> knn(data.training, pengamatan.baru, kelas, k=5)
[1] versicolor
Levels: setosa versicolor virginica

```

3. Buat fungsi Rscript datagambar

```

x <- seq(1.00, 6.00, by=0.2)
y <- seq(0.00, 2.50, by=0.05)
grid <- NULL
for (i in x) {
  for (j in y) {
    grid <- rbind(grid, c(i, j))
  }
}
datagambar <- data.frame(grid)

```

Output:

```
> datagambar
      X1  X2
1  1.0 0.00
2  1.0 0.05
3  1.0 0.10
4  1.0 0.15
5  1.0 0.20
6  1.0 0.25
7  1.0 0.30
8  1.0 0.35
9  1.0 0.40
10 1.0 0.45
11 1.0 0.50
12 1.0 0.55
13 1.0 0.60
14 1.0 0.65
15 1.0 0.70
16 1.0 0.75
17 1.0 0.80
18 1.0 0.85
19 1.0 0.90
20 1.0 0.95
21 1.0 1.00
22 1.0 1.05
23 1.0 1.10
24 1.0 1.15
25 1.0 1.20
26 1.0 1.25
27 1.0 1.30
28 1.0 1.35
29 1.0 1.40
30 1.0 1.45
31 1.0 1.50
32 1.0 1.55
33 1.0 1.60
34 1.0 1.65
35 1.0 1.70
36 1.0 1.75
37 1.0 1.80
38 1.0 1.85
```

```

463 2.8 0.15
464 2.8 0.20
465 2.8 0.25
466 2.8 0.30
467 2.8 0.35
468 2.8 0.40
469 2.8 0.45
470 2.8 0.50
471 2.8 0.55
472 2.8 0.60
473 2.8 0.65
474 2.8 0.70
475 2.8 0.75
476 2.8 0.80
477 2.8 0.85
478 2.8 0.90
479 2.8 0.95
480 2.8 1.00
481 2.8 1.05
482 2.8 1.10
483 2.8 1.15
484 2.8 1.20
485 2.8 1.25
486 2.8 1.30
487 2.8 1.35
488 2.8 1.40
489 2.8 1.45
490 2.8 1.50
491 2.8 1.55
492 2.8 1.60
493 2.8 1.65
494 2.8 1.70
495 2.8 1.75
496 2.8 1.80
497 2.8 1.85
498 2.8 1.90
499 2.8 1.95
500 2.8 2.00
[ reached 'max' / getOption("max.print") -- omitted 826 rows ]
> |

```

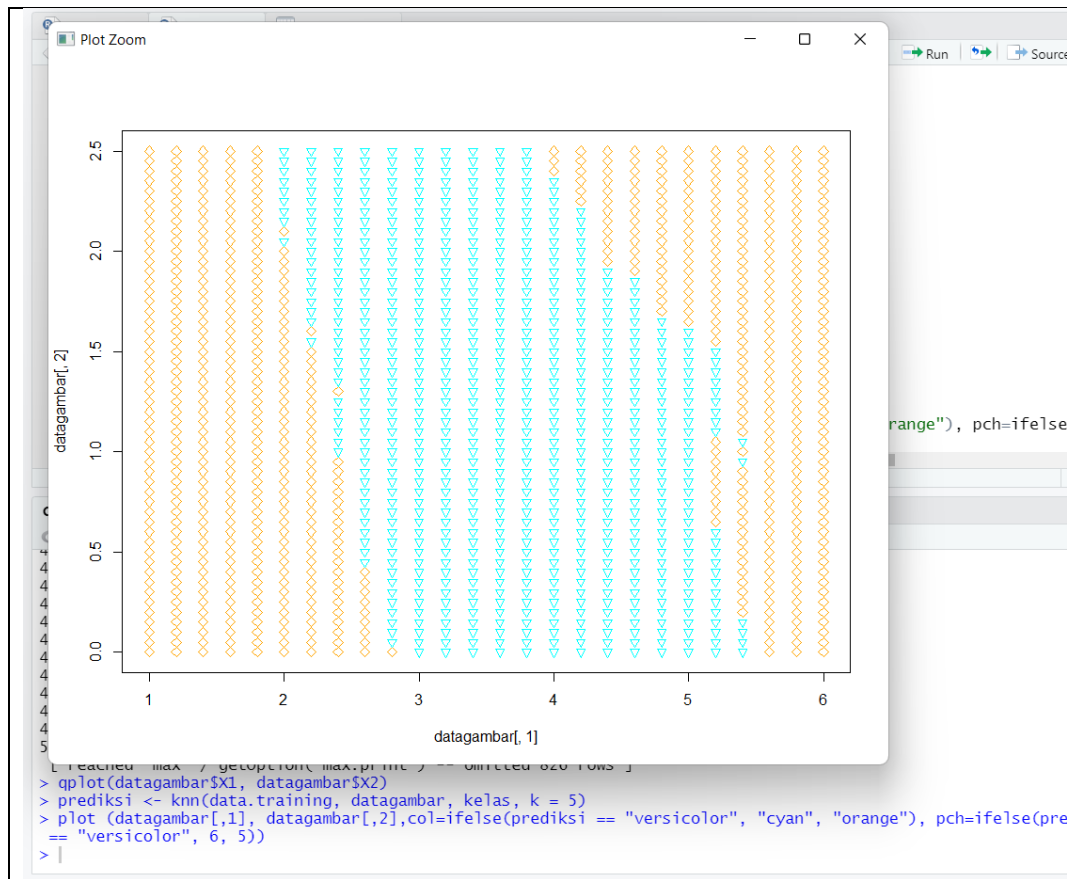
4. Membuat plot dari hasil knn

```

> qplot(datagambar$X1, datagambar$X2)
> prediksi <- knn(data.training, datagambar, kelas, k =
5)
> plot (datagambar[,1],
datagambar[,2], col=ifelse(prediksi == "versicolor",
"cyan", "orange"), pch=ifelse(prediksi == "versicolor",
6, 5))

```

Output:



Tugas

Gunakan algoritma k-NN pada R untuk kasus berikut ini dengan k=3 :

sepal.length	sepal.width	species
5.4	3.7	setosa
7.2	3.2	virginica
5.4	3.7	setosa
5.5	3.3	setosa
5.4	3.9	setosa
7.1	2.8	virginica
6.1	2.8	versicolor
7.2	2.9	virginica
6.1	2.7	versicolor
5.8	2.8	virginica
6.9	2.3	versicolor
5.1	2.7	versicolor
6.3	2.4	versicolor
5.4	2.3	versicolor

Data baru:

sepal.lengt h	sepal.width	species
4.2	3.7	?



Script R :

```

>dataKNNibnu = read.csv("D:/File Kuliah Semester 5/Penambangan Data/Prak-
5/tugasprak5.csv", sep=";")
>dataKNNibnu
>View(dataKNNibnu)
>str(dataKNNibnu)
>library(class)
>kelas <- as.factor(dataKNNibnu[,3])
>data.training <- dataKNNibnu[,1:2]
>amatan.baru <- c(4.2, 3.7)
>knn(data.training, amatan.baru, kelas, k=3)

```

Output :

```

> knn(data.training, amatan.baru, kelas, k=3)
[1] setosa
Levels: setosa versicolor virginica

```

CEK LIST

Elemen Kompetensi	No Latihan	Penyelesaian	
		Selesai	Tidak selesai
1	1.1.1	✓	
2	1.2.1	✓	

FORM UMPAN BALIK

Elemen Kompetensi	Tingkat Kesulitan	Tingkat Ketertarikan	Waktu Penyelesaian dalam menit
Menampilkan hasil Klasifikasi dari kasus yang diberikan. yang diberikan.	<input type="checkbox"/> Sangat Mudah <input type="checkbox"/> Mudah <input checked="" type="checkbox"/> ✓ Biasa <input type="checkbox"/> Sulit <input type="checkbox"/> Sangat Sulit	<input type="checkbox"/> Tidak Tertarik <input type="checkbox"/> Cukup Tertarik <input type="checkbox"/> Tertarik <input checked="" type="checkbox"/> ✓ Sangat Tertarik	



<p>Melakukan perhitungan manual menggunakan excel.</p>	<p><input type="checkbox"/> Sangat Mudah</p> <p><input type="checkbox"/> Mudah</p> <p><input type="checkbox"/> ✓ Biasa</p> <p><input type="checkbox"/> Sulit</p> <p><input type="checkbox"/> Sangat Sulit</p>	<p><input type="checkbox"/> Tidak Tertarik</p> <p><input type="checkbox"/> Cukup Tertarik</p> <p><input type="checkbox"/> Tertarik</p> <p><input type="checkbox"/> ✓ Sangat Tertarik</p>	
--	---	--	--