

Tugas 3 kNN

ditujukan untuk memenuhi salah satu tugas mata kuliah Pengantar Kecerdasan Buatan

Dosen Pengampu Isman Kurniawan, Ph.D.

Disusun oleh :

Luqman Haries - 1301180072



FAKULTAS TEKNIK INFORMATIKA

UNIVERSITAS TELKOM

BANDUNG

2020

Perhitungan Jarak

Formula untuk menghitung jarak titik a dan b menggunakan Manhattan Distance. Perhitungan tersebut Di mana x1 adalah baris data pertama, x2 adalah data baris kedua dan i adalah indeks untuk kolom tertentu saat kita menjumlahkan di semua kolom. Dengan formula ini untuk mencari jarak hanya dengan menjumlahkan semua selisih dari jarak xi dan yi.

$$d(x, y) = \sum_{i=1}^m |x_i - y_i|$$

Figure 1 Manhatta Distance

Prapemrosesan

Setelah jarak didapat, kemudia jarak tersebut disimpan kedalam list distance dan disorting dengan urutan terkecil ke terbesar. Kemudian memilih k teratas untuk dikembalikan sebagai tetangga yang dekat, dilakukan dengan melacak jarak untuk setiap record dari list distance dengan dataset.

```
#praproses menghitung distance dan menyimpan dalam list
def praProcess(k, train, x):
    distance = list()
    for i in train:
        if i+1 == train[-1]:
            break
        distance.append(manDistance(listData[x],listData[i])
    )
    distance.sort()
```

Figure 2 praposes

Klasifikasi kNN

Melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Titik yang terdekat akan dikumpulkan dari set data kedalam list, dan titik terjauh dimasukkan kedalam list, kemudia list tersebut dibandingkan, dan akan dibandingkan dengan outcome dari titik, dan hasilnya akan mengembalikan nilai True atau False.

K terbaik

Percobaan nilai K dimulai dari 1 – 5, dari percobaan tersebut akurasi knn dengan nilai k=2 merupakan nilai k yang paling optimal dengan tingkat akurasi sebesar 0.79, berikut hasil dalam bentuk tabel:

Nilai K	Data set 1	Data set 2	Data set 3	Data set 4	Data set 5
1	0.6233766233766 234	0.7	0.720779220779 2207	0.6339869281045 751	0.63870967741935 48
2	0.6493506493506 493	0.79444444444444 444	0.707792207792 2078	0.6274509803921 569	0.64516129032258 06
3	0.6753246753246 753	0.74444444444444 445	0.727272727272 7273	0.6274509803921 569	0.67096774193548 39
4	0.7012987012987 013	0.78888888888888 889	0.727272727272 7273	0.6535947712418 301	0.70967741935483 87
5	0.6688311688311 688	0.78333333333333 333	0.733766233766 2337	0.6405228758169 934	0.73548387096774 19

Rata-rata Akurasi menggunakan **fold cross-validation** dengan nilai rata rata tertinggi pada k= 4 dengan dataset-5, dapat dilihat dalam tabel sebagai berikut:

Nilai K	Data set 1	Data set 2	Data set 3	Data set 4	Data set 5
1	0.004047900151796255	0.007352092352092352	0.01327373924776522	0.01750420112588509	0.021399048062450153
2	0.004216562658121099	0.008021083854417186	0.013971346114203257	0.018163648901826524	0.02209161014388412
3	0.004385225164445943	0.007887606220939555	0.01394183017559641	0.01813394004858826	0.02222877786689992
4	0.004553887670770787	0.00827881994548661	0.014399092970521542	0.018765065939229723	0.023101500051980556
5	0.004343059537864733	0.008067580567580567	0.014194355428121662	0.018473553017959013	0.022980241823970782

Output

```

.py"
##### K = 1 #####
data set ke = 1
rata-rata akurasi 0.004047900151796255
akurasi 0.6233766233766234

data set ke = 2
rata-rata akurasi 0.007352092352092352
akurasi 0.7

data set ke = 3
rata-rata akurasi 0.01327373924776522
akurasi 0.7207792207792207

data set ke = 4
rata-rata akurasi 0.01750420112588509
akurasi 0.6339869281045751

data set ke = 5
rata-rata akurasi 0.021399048062450153
akurasi 0.6387096774193548

result = [0.6233766233766234, 0.7, 0.7207792207792207, 0.6339869281045751, 0.6387096774193548]

##### K = 2 #####
data set ke = 1
rata-rata akurasi 0.004216562658121099
akurasi 0.6493506493506493

data set ke = 2
rata-rata akurasi 0.008021083854417186
akurasi 0.7944444444444444

data set ke = 3
rata-rata akurasi 0.013971346114203257

data set ke = 4
rata-rata akurasi 0.018163648901826524
akurasi 0.6535947712418301

data set ke = 5
rata-rata akurasi 0.023101500051980556
akurasi 0.7096774193548387

result = [0.7012987012987013, 0.7888888888888889, 0.7272727272727273, 0.6535947712418301, 0.7096774193548387]

##### K = 5 #####
data set ke = 1
rata-rata akurasi 0.004343059537864733
akurasi 0.6688311688311688

data set ke = 2
rata-rata akurasi 0.008067580567580567
akurasi 0.7833333333333333

data set ke = 3
rata-rata akurasi 0.014194355428121662
akurasi 0.7337662337662337

data set ke = 4
rata-rata akurasi 0.018473553017959013
akurasi 0.6405228758169934

data set ke = 5
rata-rata akurasi 0.022980241823970782
akurasi 0.7354838709677419

result = [0.6688311688311688, 0.7833333333333333, 0.7337662337662337, 0.6405228758169934, 0.7354838709677419]

```