# PENERAPAN ALGORITMA *K-NN* UNTUK KLASIFIKASI DATA MAHASISWA (Studi Kasus: STIMIK Sepuluh Nopember Jayapura)

**SITTI NUR ALAM**
azkadzar@gmail.com

**Staff Pengajar pada Program Studi Teknik Informatika**
**STIMIK Sepuluh Nopember Jayapura**

***Abstraksi -*** *Masalah kesehatan merupakan permasalahan yang sangat penting untuk diperhatikan setiap orang, salah satunya adalah dengan memperhatikan IMT (Indeks Masa Tubuh). Ketidaktahuan sesorang terhadap IMT-nya mengakibatkan seringnya seseorang memiliki IMT yang tidak normal. Penelitian ini dimaksudkan untuk melakukan klasifikasi IMT terhadap sebagian data mahasiswa di STIMIK Sepuluh Nopember Jayapura untuk mengetahui. Klasifikasi dilakukan ke dalam tiga kelompok, yaitu Kurus, Nomal, dan Obesitas. Klasifikasi dilakukan menggunakan salah satu algoritma klasifikasi di dalam data mining, yaitu Algoritma k-nearest neighbor (k-nn). Data yang dibutuhkan untuk melakukan klasifikasi terdiri dari Tinggi Badan (cm), berat badan (kg), % lemak tubuh dan lingkar perut (cm). Jumlah data training yang digunakan sebanyak 50 (lima puluh) data, sedangkan data sampel yang akan dikelompokkan adalah sebanyak 10 (sepuluh) data sampel. Penelitian ini dapat menghasilkan klasifikasi data terhadap semua data sampel, namun agar hasil klasifikasi dapat lebih akurat maka sebaiknya dalam proses pengumpulan data training maupun data sampel dilakukan oleh pihak-pihak yang berkompeten, khususnya bidang kesehatan.*

***Kata Kunci : klasifikasi, IMT, k-nn, data training, data sampel***

## 1. PENDAHULUAN

### 1.1 Latar Belakang

Di tengah kehidupan modern saat ini, masalah kesehatan merupakan isu yang sangat penting untuk diperhatikan setiap orang, seringkali masalah kesehatan muncul karena pola hidup yang kurang baik, misalnya terlalu seringnya seseorang mengkonsumsi makanan cepat saji. Salah satu mekanisme dalam mengontrol kesehatan tubuh adalah dengan memperhatikan IMT (Indeks Masa Tubuh). Ketidaktahuan sesorang terhadap IMT-nya mengakibatkan seringnya seseorang memiliki IMT yang tidak normal. Dengan mengetahui IMT-nya maka setiap orang dapat mengatur pola hidupnya sehingga kesehatannya dapat lebih terjaga. Salah satu yang dapat diketahui melalui informasi IMT seseorang adalah apakah seseorang masuk dalam kelompok berat badan normal, kurus atau obesitas. Dengan menggunakan salah satu metode dalam ilmu data mining, yaitu *k-nn*, klasifikasi terhadap suatu data dapat dilakukan.

*k-nn* digunakan untuk mengelompokkan suatu data baru berdasarkan data jarak data baru itu ke beberapa data/tetangga (*neighbor*) terdekat. Dalam hal ini jumlah data/tetangga terdekat yang dilibatkan dalam penentuan prediksi label kelas data uji ditentukan oleh user yang dinyatakan dengan *k*. Misalkan digunakan k = 5, maka setiap data testing dihitung jaraknya terhadap data training dan dipilih 5 data training yang datanya paling dekat ke data testing. Setelah diketahui 5 data ini, diperiksa output atau labelnya masing-masing. Lalu ditentukan output mana yang frekuensinya paling banyak, dimasukkan suatu data testing ke kelompok dengan output paling banyak.

### 1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah dalam penelitian ini adalah: Bagaimana mengklasifikasi data mahasiswa STIMIK Sepuluh Nopember Jayapura menggunakan Algoritma *k-nearest neighbor* (*k-nn)*?

### 1.3 Tujuan Penelitian

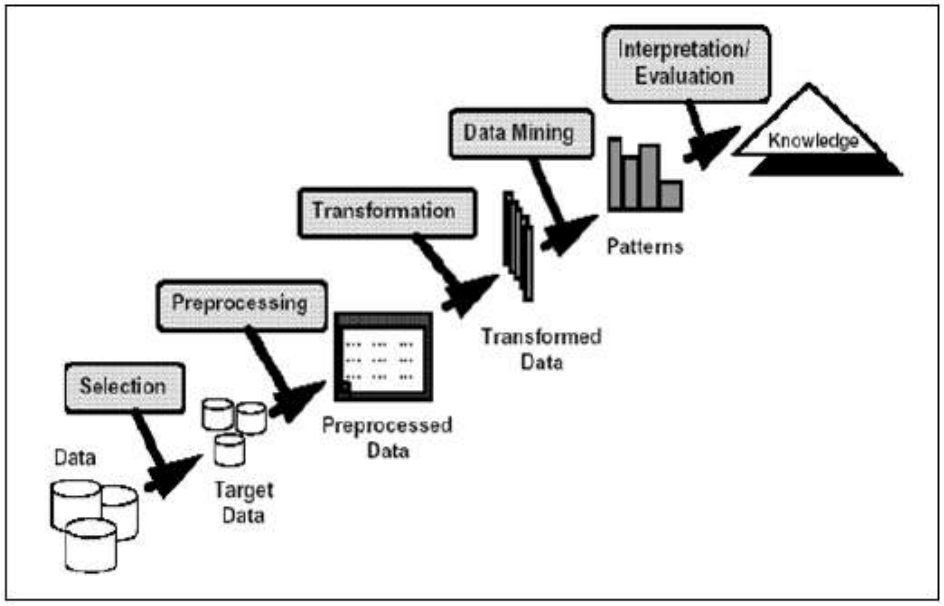
Tujuan penelitian ini adalah mengetahui klasifikasi data sebagian mahasiswa STIMIK Sepuluh Nopember Jayapura berdasarkan data tinggi badan, berat badan, % lemak dalam tubuh dan lingkar perut menggunakan Algoritma *k-nn.*

## 2. LANDASAN TEORI

### 2.1 Data Mining

*Data Mining* adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam database. *Data mining* adalah proses yang menggunakan teknik statistic, matematika, kecerdasan buatan dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terakit dari berbagai database besar.

*Knowledge discovery data (*KDD) adalah keseluruhan proses *non-trivial* untuk mencari dan mengidentifikasikan pola *(pattern)* dalam data, dimana pola yang ditemukan bersifat sah, baru dapat bermanfaat dan dapat dimengerti.




Gambar 1. Proses Knowledge Discovery in Database (KDD)

### 2.2 Algoritma *K-Nearest Neighbor (k-nn)*

*K-Nearest Neighbor* (KNN) termasuk kelompok *instance-based learning*. *Algoritma* ini juga merupakan salah satu teknik *lazy learning*. KNN dilakukan dengan mencari kelompok k objek dalam data training yang paling dekat (mirip) dengan objek pada data baru atau data testing. *Algoritma K-Nearest Neighbor* adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. *Nearest Neighbor* adalah pendekatan untuk mencari kasus dengan menghitung kedekatan antara kasus baru dan kasus lama yaitu berdasarkan pada pencocokan bobot dari sejumlah fitur yang ada.

Untuk mendefinisikan jarak antara dua titik yaitu titik pada data training (x) dan titik pada data testing (y) maka digunakan rumus *Euclidean*, seperti yang ditunjukkan pada persamaan (1)

$$D(x,y) = \sqrt{\sum_{k-1}^{n}(x_k - y_k)^2} \qquad (1)$$

Dengan D adalah jarak antara titik pada data training x dan titik data testing y yang akan diklasifikasi, dimana x=x1,x2,…,xi dan y=y1,y2,…,yi dan I merepresentasikan nilai atribut serta n merupakan dimensi atribut.

Pada *fase training*, *algoritma* ini hanya melakukan penyimpanan *vektor-vektorfitur* dan klasifikasi data *training sample.* Pada *fase* klasifikasi, *fitur-fitur* yang sama dihitung untuk *testing data* (yang klasifikasinya tidak diketahui). Jarak dari *vektor* baru yang ini terhadap seluruh vektor *training sample* dihitung dan sejumlah *k* buah yang paling dekat diambil.

Langkah-langkah untuk menghitung metode *Algoritma K-Nearest Neighbor:*

a. Menentukan Parameter K (Jumlah tetangga paling dekat).
b. Menghitung kuadrat jarak *Euclid (queri instance)* masing-masing objek terhadap data sampel yang diberikan.
c. Kemudian mengurutkan objek-objek tersebut ke dalam kelompok yang mempunyai jarak *Euclid* terkecil.
d. Mengumpulkan kategori Y (*Klasifikasi Nearest Neighbor*)
e. Dengan menggunakan kategori *Nearest Neighbor* yang paling mayoritas maka dapat diprediksi nilai *queri instance* yang telah dihitung.

### 2.3 Algoritma Pengurutan *Bubble Sort*

Pengurutan merupakan proses dasar dalam algoritma dan struktur data, salah satu algoritma pengurutan yang dapat digunakan dalam mengurutkan sekumpulan data adalah Algoritma *Bubble Sort*. Algoritma ini merupakan proses pengurutan data yang secara berangsur-angsur berpindah ke posisi yang tepat, karena itulah algoritma ini dinamakan *bubble* atau gelembung. Secara sederhana, algoritma *bubble sort* melakukan pengurutan data dengan data disebelahnya secara terus menerus sampai dalam satu iterasi tertentu tidak ada lagi perubahan urutan kelompok data yang hendak diurutkan. Berikut adalah salah satu bentuk algoritma *bubble sort*:

```
procedure bubblesort (A : list of sorttable items) defined as:
 do
  swapped := false
  for each I in 0 to length (A) – 2 inclusive do:
    if A[i] > A[i+1] then
      swap (A[i], A[i+1])
      swapped := true
    end if
  end for
 while swapped
end procedure
```

## 3. ANALISIS DATA

### a. Metode Pengolahan Data

Berikut adalah tahapan pengumpulan dan pengolahan data menggunakan Algoritma *k-nn*:

1. Mengumpulkan data sampel yang akan digunakan, jumlah data sampel yang digunakan sebanyak 50 (lima puluh) data sampel. Data dikumpulkan secara acak dari mahasiswa di lingkungan STIMIK Sepuluh Nopember Jayapura. Data yang dikumpulkan terdiri dari nama mahasiswa, tinggi badan (cm), berat badan (kg), % lemak (%), dan lingkar perut (cm)
2. Menentukan jumlah tetangga terdekat (*k*)
3. Menghitung jarak antar objek menggunakan *Euclidean Distance*
4. Mengurutkan data hasil hitung menggunakan Algoritma *Bubble Sort*
5. Proses penyiapan data menggunakan Aplikasi Microsoft Excel, sedangkan proses perhitungan Algoritma *k-nn* menggunakan *Software* Matlab.

**b. Sumber Data**

Data yang digunakan berasal dari 50 (lima puluh) mahasiswa. Berikut ditampilkan 10 (sepuluh) data yang digunakan:

**Tabel 1. Data sampel yang digunakan**

| No | Inisial Mahasiswa | Tinggi Badan | Berat Badan | % Lemak | Lingkar Perut | Status |
|---|---|---|---|---|---|---|
| 1 | A | 163 | 59 | 25,4 | 74 | Normal |
| 2 | B | 170 | 85 | 42,9 | 112 | Obesitas |
| 3 | C | 167 | 59 | 16,5 | 71 | Kurus |
| 4 | D | 172 | 75 | 31 | 79 | Normal |
| 5 | E | 168 | 50 | 10,4 | 62 | Kurus |
| 6 | F | 165 | 73 | 29,1 | 83 | Obesitas |
| 7 | G | 160 | 54 | 15,7 | 73 | Normal |
| 8 | H | 169,5 | 55 | 15,2 | 75 | Kurus |
| 9 | I | 169 | 79 | 22 | 84 | Obesitas |
| 10 | J | 172 | 68 | 22,8 | 79 | Normal |

## 4. HASIL DAN PEMBAHASAN

### 1.1 Pengubahan tipe data

Agar nantinya data sampel yang diperoleh dapat diolah di *Software* MATLAB maka setiap data yang akan digunaan harus bertipe *Numeric*, namun pada data sampel yang dimiliki pada Tabel 1 terdapat data yang bertipe *String*, yaitu **Status**. Untuk itu data string tersebut perlu diubah ke dalam data *Numeric*. Perubahan tipe data tersebut adalah **1 = Normal, 2 = Obesitas, 3 = Kurus**. Tabel 2 berikut adalah hasil perubahan tipe data tersebut.

**Tabel 2. Data sampel setelah perubahan tipe data**

| No | Inisial Mahasiswa | Tinggi Badan | Berat Badan | % Lemak | Lingkar Perut | Status |
|---|---|---|---|---|---|---|
| 1 | A | 163 | 59 | 25,4 | 74 | 1 |
| 2 | B | 170 | 85 | 42,9 | 112 | 2 |
| 3 | C | 167 | 59 | 16,5 | 71 | 3 |
| 4 | D | 172 | 75 | 31 | 79 | 1 |
| 5 | E | 168 | 50 | 10,4 | 62 | 3 |
| 6 | F | 165 | 73 | 29,1 | 83 | 2 |
| 7 | G | 160 | 54 | 15,7 | 73 | 1 |
| 8 | H | 169,5 | 55 | 15,2 | 75 | 3 |
| 9 | I | 169 | 79 | 22 | 84 | 2 |
| 10 | J | 172 | 68 | 22,8 | 79 | 1 |

### 1.2 Pengolahan data

1. Data sampel yang telah dikumpulkan diolah menggunakan Aplikasi Microsoft Excel untuk selanjutnya dibaca melalui *Software Matlab*

2. Menentukan jumlah *k* yang akan digunakan. Dalam penelitian ini jumlah k yang digunakan adalah 5

3. Menghitung jarak setiap data sampel terhadap data baru menggunakan *Eucliedian Distance,* seperti ditunjukkan pada Persamaan 1. Berikut ditampilkan potongan program untuk menghitung jarak antar objek

```
for i=1:n,
d(i)=sqrt(((X(1)-B(i,1))^2) + ((X(2)-B(i,2))^2) + ((X(3)-B(i,3))^2) + ((X(4)-B(i,4))^2))
end;
```

4. Data hasil perhitungan jarak selanjutnya diurutkan menggunakan Algoritma *Bubble Sort,* seperti ditunjukkan melalui potongan program berikut

```
for i=2:m
   for j=m:-1:i
      if C(j,n) < C(j-1,n)
         temp=C(j,:);
         C(j,:)=C(j-1,:);
         C(j-1,:)=temp;
      end;
   end;
end;
```

5. Selanjutnya dilakukan proses untuk menghitung frekuensi masing-masing data yang telah diurutkan, selanjutnya menampilkan hasil perhitungan. Proses tersebut dilakukan melalui sebagian potongan programberikut:

```
kurus=0;
normal=0;
obesitas=0;
for i=1:length(X)
   if X(i)==1,
      kurus=kurus+1;
   elseif X(i)==2,
      normal=normal+1;
   else
      obesitas=obesitas+1;
   end;
end;
kurus
normal
obesitas
```

6. Berikut ditampilkan sebagian hasil perhitungan untuk satu data sampel dengan inisial X, yaitu **[170 60 20 80]**

```
>> knn
A =
  163.0000   59.0000   25.4000   74.0000    1.0000
  170.0000  125.0000   42.9000  112.0000    2.0000
  167.0000   59.0000   16.5000   71.0000    3.0000
  172.0000   75.0000   31.0000   79.0000    1.0000
  168.0000   50.0000   10.4000   62.0000    3.0000
  165.0000   73.0000   29.1000   83.0000    2.0000
  160.0000   54.0000   15.7000   73.0000    1.0000
  169.5000   55.0000   15.2000   75.0000    3.0000
  169.0000   79.0000   22.0000   84.0000    2.0000
  172.0000   68.0000   22.8000   79.0000    1.0000
```

```
K =
     5

B =
  163.0000   59.0000   25.4000   74.0000
  170.0000  125.0000   42.9000  112.0000
  167.0000   59.0000   16.5000   71.0000
  172.0000   75.0000   31.0000   79.0000
  168.0000   50.0000   10.4000   62.0000
  165.0000   73.0000   29.1000   83.0000
  160.0000   54.0000   15.7000   73.0000
  169.5000   55.0000   15.2000   75.0000
  169.0000   79.0000   22.0000   84.0000
  172.0000   68.0000   22.8000   79.0000

X =
   170    60    20    80

n =
    10

d =
  Columns 1 through 9

   10.7313   75.9830   10.1612   18.7350   22.8070   16.9059   14.2650    8.5610   19.5448

  Column 10

    8.7658

e =
   10.7313
   75.9830
   10.1612
   18.7350
   22.8070
   16.9059
   14.2650
    8.5610
   19.5448
    8.7658

C =
  163.0000   59.0000   25.4000   74.0000    1.0000   10.7313
  170.0000  125.0000   42.9000  112.0000    2.0000   75.9830
  167.0000   59.0000   16.5000   71.0000    3.0000   10.1612
  172.0000   75.0000   31.0000   79.0000    1.0000   18.7350
  168.0000   50.0000   10.4000   62.0000    3.0000   22.8070
  165.0000   73.0000   29.1000   83.0000    2.0000   16.9059
  160.0000   54.0000   15.7000   73.0000    1.0000   14.2650
  169.5000   55.0000   15.2000   75.0000    3.0000    8.5610
  169.0000   79.0000   22.0000   84.0000    2.0000   19.5448
  172.0000   68.0000   22.8000   79.0000    1.0000    8.7658

m =
    10
```

```
n =
     6

C =
  169.5000   55.0000   15.2000   75.0000    3.0000    8.5610
  172.0000   68.0000   22.8000   79.0000    1.0000    8.7658
  167.0000   59.0000   16.5000   71.0000    3.0000   10.1612
  163.0000   59.0000   25.4000   74.0000    1.0000   10.7313
  160.0000   54.0000   15.7000   73.0000    1.0000   14.2650
  165.0000   73.0000   29.1000   83.0000    2.0000   16.9059
  172.0000   75.0000   31.0000   79.0000    1.0000   18.7350
  169.0000   79.0000   22.0000   84.0000    2.0000   19.5448
  168.0000   50.0000   10.4000   62.0000    3.0000   22.8070
  170.0000  125.0000   42.9000  112.0000    2.0000   75.9830

D =
  169.5000   55.0000   15.2000   75.0000    3.0000    8.5610
  172.0000   68.0000   22.8000   79.0000    1.0000    8.7658
  167.0000   59.0000   16.5000   71.0000    3.0000   10.1612
  163.0000   59.0000   25.4000   74.0000    1.0000   10.7313
  160.0000   54.0000   15.7000   73.0000    1.0000   14.2650

X =
     3
     1
     3
     1
     1

kurus =
     3

normal =
     0

obesitas =
     2

hasil_akhir =

Kurus
```

Dari hasil perhitungan di atas, dapat diketahui bahwa mahasiswa dengan inisial X yang mempunyai data **[170 60 20 80]** masuk klasifikasi **kurus**.

## 5. KESIMPULAN

Berikut ditampilkan 10 (sepuluh) hasil klasifikasi data mahasiswa yang dijadikan sebagai data sampel:

| No | Inisial Mahasiswa | Tinggi Badan | Berat Badan | % Lemak | Lingkar Perut | Status | Keterangan |
|---|---|---|---|---|---|---|---|
| 1 | A1 | 165 | 55 | 27,4 | 70 | 3 | Kurus |
| 2 | B1 | 169 | 60 | 39,9 | 80 | 3 | Kurus |
| 3 | C1 | 162 | 54 | 18,5 | 71 | 2 | Obesitas |
| 4 | D1 | 172 | 67 | 31 | 79 | 3 | Kurus |
| 5 | E1 | 164 | 52 | 12,4 | 64 | 2 | Obesitas |
| 6 | F1 | 165 | 61 | 28,1 | 83 | 3 | Kurus |
| 7 | G1 | 161 | 52 | 17,7 | 74 | 2 | Obesitas |
| 8 | H1 | 175 | 60 | 19,2 | 70 | 3 | Kurus |
| 9 | I1 | 169 | 75 | 24 | 84 | 3 | Kurus |
| 10 | J1 | 172 | 77 | 25 | 79 | 3 | Kurus |

Agar hasil klasifikasi terhadap data lebih akurat, sebaiknya dalam melakukan pengumpulan data sampel terutama dalam menghitung Lingkar Perut dan % Lemak Tubuh melibatkan pihak yang memiliki kompetensi, seperti dalam bidang kesehatan.

## 6. DAFTAR PUSTAKA

Budi Santoso, 2007, *Data Mining - Teknik Pemanfaatan Data Untuk Keperluan Bisnis*, Graha Ilmu, Yogyakarta

Eko Prasetyo, 2014, *Data Mining - Mengolah Data Menjadi Informasi Menggunakan Matlab*, Andi Offset, Yogyakarta

Emerensye S.Y. Pandie, 2012, *Implementasi Algoritma Data Mining K-Nearest Neighbour (K-NN) Dalam Pengambilan Keputusan Pengajuan Kredit*, Seminar Nasional Sains dan Teknik (SAINSTEK 2012), Kupang 13 Nopember 2012

Kusrini, Ehma T. Luthfi, 2009, *Algoritma Data Mining*, Andi Offset, Yogyakarta

Ricky Imanuel Ndaumanu, Kusrini, M. Rudyanto Arief, 2014, *Analisis Prediksi Tingkat Pengunduran Diri Mahasiswa Dengan Metode K-Nearest Neigbhor*, Jatasi Vo. 1 No. 1 September 2014

Turban Efraim, Aronson Jay E, dan LiangTing Peng, 2005. *Decision Support Sistems and Intelligent Sistems*. Edisi 7, Jilid 1, Versi Bahasa Inonesia, Andi Offset, Jogyakarta