

Implementasi *Data Science* dalam Bidang Kesehatan

(Studi Kasus : *Heart Disease*)

1. Latar Belakang

Menurut WHO (World Health Organization), penyakit kardiovaskular adalah penyebab utama kematian secara global yang diperkirakan merenggut sekitar 17,9 juta tiap tahunnya. Penyakit kardiovaskular adalah sekelompok gangguan jantung dan pembuluh darah termasuk penyakit jantung koroner, penyakit serebrovaskular, penyakit jantung rematik, dan kondisi lainnya. Dari lima kematian akibat penyakit kardiovaskuler, empat diantaranya disebabkan oleh serangan jantung dan stroke. Selain itu, sepertiga dari kematian tersebut terjadi pada orang dengan usia di bawah 70 tahun.

Penyakit kardiovaskular dapat menimpa siapa saja tanpa memandang bulu. Tekanan darah tinggi, diabetes, kolesterol tinggi, kurangnya aktivitas fisik, serta pola hidup yang kurang baik merupakan beberapa alasan utama yang menyebabkan terjadinya peningkatan risiko penyakit kardiovaskular. Dengan mengurangi faktor risiko seperti kebiasaan merokok, pola makan yang tidak sehat, penggunaan alkohol, dan aktivitas fisik yang kurang, setidaknya dapat mengurangi risiko terkena penyakit kardiovaskular.

Prediksi penyakit jantung masih menjadi salah satu kasus penyakit yang rumit dan kompleks. Data kesehatan dikumpulkan dari beberapa sumber yang tersedia, salah satunya data yang disimpan dalam Electronic Health Record (EHR). Untuk mengekstrak informasi dari data tersebut, perlu adanya peran *data science*. *Data science* berperan untuk memanipulasi dan mengekstrak data yang semula mengandung informasi secara implisit menjadi informasi yang bermanfaat dalam menentukan kebijakan atau keputusan yang berpengaruh bagi perusahaan atau industri kedepannya.

Data science adalah ilmu yang menggabungkan kemampuan beberapa bidang ilmu dengan keahlian pemrograman, matematika, dan statistika. *Data science* sangat populer dalam industri kesehatan dengan menerapkan metode *machine learning*. Dalam *machine learning* terdapat dua teknik yaitu *supervised* dan *unsupervised*. Adanya *machine learning* dapat membantu memprediksi dan menentukan akurasi dari kumpulan data serta meningkatkan kecepatan mengolah dan menganalisis data. Berkaca dari paparan di atas, tujuan dari kepenulisan ini adalah untuk mengetahui peran *data science* dalam mendeteksi penyakit jantung dengan mengolah data dalam jumlah besar serta bekerja dengan variabel yang banyak.

2. Dasar Teori

2.1. *Data science*

Data science adalah ilmu interdisiplin yang menggabungkan berbagai ilmu pengetahuan. Suatu data mentah yang tidak diolah atau diproses akan menjadi sia-sia. Di sinilah peran *data science* diperlukan. Tujuan dari *data science* adalah mengolah suatu data menjadi pengetahuan atau informasi. Dengan kata lain, *data science* mengekstrak sebuah informasi dari suatu data. Data menjadi sesuatu yang sangat berharga bahkan muncul istilah “Data is the New Gold”. Adanya transformasi digital

di era revolusi industri 4.0 saat ini, menyebabkan keberadaan data sangat penting sehingga muncul istilah *Big Data*. *Big Data* mengacu pada data yang sangat besar, cepat, dan kompleks sehingga memerlukan proses pengolahan dan pengelolaan yang lebih kompleks pula. Dengan adanya *data science*, suatu data yang tidak terstruktur maupun terstruktur dalam ukuran yang sangat besar dapat dianalisis dengan baik. *Data science* dapat mengubah data menjadi pengetahuan atau informasi yang berperan penting dalam pengambilan keputusan atau kebijakan suatu bidang industri.

2.2. Algoritma K-Nearest Neighbor (KNN)

Algoritma K-Nearest Neighbour adalah algoritma yang tergolong dalam *supervised learning* yang digunakan untuk mengklasifikasi suatu objek berdasarkan K buah data latih yang jaraknya paling dekat dengan objek tersebut (Rivki, 2017). Nilai K harus berupa bilangan ganjil yang lebih dari satu dan nilai K tidak boleh lebih besar dari jumlah data latih. Algoritma KNN bergantung pada kedekatan antara data latih dengan data uji yang dapat dihitung dengan rumus jarak Euclidean seperti persamaan berikut (Yuniarto, 2020).

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (x_{ir} - x_{jr})^2}$$

Keterangan:

x_{ir} : data uji ke- i pada variabel ke- r
 x_{jr} : data latih ke- j pada variabel ke- r
 $d(x_i, x_j)$: jarak data uji i dengan data latih j
 n : dimensi data variabel bebas

2.3. Evaluasi Model

Evaluasi model dilakukan untuk mengetahui nilai *accuracy*, *sensitivity*, dan *specificity* algoritma KNN. Pada kasus ini, evaluasi model dilakukan dengan menggunakan *Confusion Matrix*. *Confusion Matrix* merupakan sebuah tabel yang menyatakan jumlah data uji yang benar diklasifikasikan dan jumlah data uji yang salah diklasifikasikan (Ahmad, 2021). Tabel *confusion matrix* dapat dilihat pada tabel berikut.

Correct Classification	Classified	
	Predicted “+”	Predicted “-”
Actual “+”	True Positives (TN)	False Negative (FN)
Actual “-”	False Positive (FP)	True Negative (TN)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP}$$

3. Metode Penelitian

3.1. Metode Pengumpulan Data

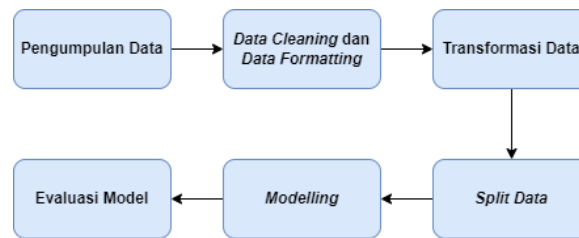
Dataset yang digunakan yaitu data penyakit jantung yang bersumber dari UCI Machine Learning Repository. Kumpulan data ini terdiri dari 297 instansi dengan 14 atribut. *Dataset* yang digunakan dalam mendiagnosis penyakit jantung adalah *dataset* yang terdiri dari 4 *dataset* yang berbeda. Namun, pada kepenulisan ini hanya digunakan *dataset* Cleveland. *Dataset* ini berisi 76 atribut, tetapi dalam penelitian yang telah dipublikasikan hanya 14 atribut yang diproses. Dalam *dataset* terdapat 14 atribut dengan 1 atribut sebagai variabel dependen. Deskripsi atribut ditunjukkan pada tabel berikut.

No.	Atribut	Tipe Data	Deskripsi Data	Nilai
1	Sex	Diskrit	Jenis kelamin pasien	1: Laki-laki 0: Perempuan
2	Age	Kontinu	Umur pasien	Beberapa nilai dengan rentang 28 dan 77 (tahun)
3	CP (Chest Pain) Type	Diskrit	Mewakili tipe nyeri dada yang diderita pasien	0: <i>typical angina</i> 1: <i>atypical angina</i> 2: <i>non-anginal pain</i> 3: <i>asymptomatic</i>
4	Rest Blood Pressure (Trestbps)	Kontinu	Mewakili tekanan darah pasien ketika dalam keadaan istirahat (dalam mmHg)	Beberapa nilai kontinu dalam mmHg
5	Serum Cholesterol (Chol)	Kontinu	Mewakili kadar kolesterol dalam darah pasien (dalam mg/dl)	Beberapa nilai kontinu dalam mg/dl
6	Fasting Blood Sugar (FBS)	Diskrit	Mewakili kadar gula darah setelah berpuasa selama 8 jam	0: <i>false</i> (FBS < 120 mg/dl) 1: <i>true</i> (FBS > 120 mg/dl)
7	Max Heart Rate (Thalach)	Kontinu	Mewakili rata-rata detak jantung pasien	Beberapa nilai dari 71 sampai 202 Rendah: di bawah 50 detak/menit Biasa: 51 – 119 detak/menit Tinggi: 120 – 180 detak/menit
8	Resting electrocardiographic results (Restecg)	Diskrit	Mewakili kondisi ECG pasien ketika dalam keadaan istirahat.	0: normal 1: keadaan ST-T wave <i>abnormality</i> yaitu keadaan di mana gelombang inversion T dan/atau ST meningkat atau menurun lebih dari 0,05 mV) 2: menunjukkan keadaan ventricular kiri mengalami hipertropi
9	Exercise Include Angina (Exang)	Diskrit	Mewakili keadaan di mana pasien mengalami nyeri dada apabila berolahraga	1: jika menyebabkan nyeri 0: jika tidak nyeri
10	Oldpeak	Diskrit	Menunjukkan penurunan ST akibat olahraga	Beberapa nilai desimal antara 0 dan 6.2

11	Slope	Diskrit	Menunjukkan <i>slope</i> dari puncak ST setelah berolahraga	0: upsloping 1: flat 2: down sloping
12	Major Vessels (Ca)	Diskrit	Banyaknya pembuluh darah yang terdeteksi melalui proses pewarnaan flourosopy	Nilai 0-3
13	Thal	Diskrit	Mewakili detak jantung pasien.	0: normal 1: <i>fix defect</i> 2: <i>reversible defect</i>
14	Num	Diskrit	Hasil diagnose penyakit jantung	0: tidak terkena penyakit jantung 1,2,3,4: terkena penyakit jantung

3.2. Metode Penyelesaian

Algoritma yang digunakan pada studi kasus ini adalah algoritma K-Nearest Neighbor (KNN algorithm) dengan alur penyelesaian seperti pada *flowchart* berikut.



3.2.1. Pengumpulan Data

Pengumpulan data dapat dilakukan dengan *web scrapping* dan *data mining*. *Data mining* merupakan proses pengumpulan data dari berbagai sumber seperti *website* yang digunakan untuk keperluan analisis. Sedangkan *web scrapping* merupakan salah satu metode dari *data mining* yang dilakukan secara otomatis menggunakan bantuan coding untuk mengambil data dari suatu *website* dengan jumlah besar.

3.2.2. Data Preprocessing

Pada tahap *data preprocessing* mencakup *data cleaning* dan *data formatting*. *Data cleaning* adalah proses memperbaiki kesalahan dan masalah data yang teridentifikasi agar kumpulan data menjadi lengkap dan akurat. Terdapat beberapa hal yang dilakukan dalam *data cleaning* yaitu menghapus atau memperbaiki data yang salah dan mengisi nilai yang hilang. *Data formatting* dilakukan setelah sekumpulan data selesai dibersihkan. Tahap ini akan mengatasi masalah seperti beberapa format dalam data yang tidak konsisten.

3.2.3. Transformasi Data

Transformasi data digunakan untuk mengubah data ke dalam bentuk yang sesuai dalam proses data mining. Terdapat beberapa teknik dalam transformasi data salah satunya yaitu normalisasi. Normalisasi dilakukan untuk menskalakan nilai variabel dari data dalam rentang nilai tertentu sehingga variabel tersebut memiliki rentang nilai yang sama, tidak ada yang terlalu besar maupun terlalu kecil.

3.2.4. Split Data

Split data digunakan untuk membagi dataset menjadi data latih (*data training*) dan data uji (*data testing*). Data latih digunakan untuk proses pelatihan

sedangkan data uji digunakan untuk memvalidasi model. *Split data* dilakukan dengan proporsi tertentu.

3.2.5. Modelling

Pada tahap *modelling*, terdapat beberapa hal yang perlu dilakukan yaitu memilih teknik permodelan, melatih model dengan data latih, mengatur dan menyesuaikan parameter model, memvalidasi model, mengembangkan dan menguji model, memilih algoritma yang tepat, dan optimasi model.

3.2.6. Evaluasi Model

Evaluasi model digunakan untuk mengukur ketepatan dari model yang dibangun serta untuk mengetahui seberapa optimal model yang terbentuk dalam memecahkan masalah. Evaluasi model meliputi perhitungan *accuracy*, *specificity*, dan *sensitivity*.

4. Pembahasan

4.1. Pengumpulan Data

Pada tahap ini, dataset yang digunakan merupakan data penyakit jantung dari UCI Cleveland yang diperoleh dari UCI Machine Learning Repository. Data dari website tersebut diimpor dan diperoleh ringkasan data sebagai berikut.

```
> head(heart)
  age sex cp trestbps chol fbs restecg thalach exang oldpeak slope ca thal num
1  63  1  1   145  233  1      2    150    0     2.3    3 0.0  6.0  0
2  67  1  4   160  286  0      2    108    1     1.5    2 3.0  3.0  2
3  67  1  4   120  229  0      2    129    1     2.6    2 2.0  7.0  1
4  37  1  3   130  250  0      0    187    0     3.5    3 0.0  3.0  0
5  41  0  2   130  204  0      2    172    0     1.4    1 0.0  3.0  0

> summary(heart)
      age          sex          cp          trestbps          chol          fbs
Min.   :29.00   Min.   :0.0000   Min.   :1.000   Min.   : 94.0   Min.   :126.0   Min.   :0.0000
1st Qu.:48.00   1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:120.0   1st Qu.:211.0   1st Qu.:0.0000
Median :56.00   Median :1.0000   Median :3.000   Median :130.0   Median :243.0   Median :0.0000
Mean   :54.54   Mean   :0.6768   Mean   :3.158   Mean   :131.7   Mean   :247.4   Mean   :0.1448
3rd Qu.:61.00   3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:140.0   3rd Qu.:276.0   3rd Qu.:0.0000
Max.   :77.00   Max.   :1.0000   Max.   :4.000   Max.   :200.0   Max.   :564.0   Max.   :1.0000

      restecg          thalach          exang          oldpeak          slope
Min.   :0.0000   Min.   : 71.0   Min.   :0.0000   Min.   :0.000   Min.   :1.000
1st Qu.:0.0000   1st Qu.:133.0   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:1.000
Median :1.0000   Median :153.0   Median :0.0000   Median :0.800   Median :2.000
Mean   :0.9966   Mean   :149.6   Mean   :0.3266   Mean   :1.056   Mean   :1.603
3rd Qu.:2.0000   3rd Qu.:166.0   3rd Qu.:1.0000   3rd Qu.:1.600   3rd Qu.:2.000
Max.   :2.0000   Max.   :202.0   Max.   :1.0000   Max.   :6.200   Max.   :3.000

      ca          thal          num
Length:297   Length:297   Min.   :0.0000
Class :character   Class :character   1st Qu.:0.0000
Mode  :character   Mode  :character   Median :0.0000
                                   Mean   :0.9461
                                   3rd Qu.:2.0000
                                   Max.   :4.0000
```

Terlihat bahwa dalam *dataset* tersebut terdapat 14 variabel dengan tipe datanya masing-masing.

4.2. Preprocessing Data

4.2.1. Data Cleaning

Pada tahap dilakukan proses identifikasi terhadap *missing value* (NA). Apabila terdapat *missing value*, maka akan dilakukan penghapusan pada kolom tersebut.

```
> heart[heart == "?"] <- NA
> heart <- na.omit(heart)
> sum(is.na(heart))
[1] 0
```

Berdasarkan output di atas, terlihat bahwa data sudah lengkap atau tidak terdapat missing value setelah dilakukan *data cleaning*. Karena data sudah lengkap, maka proses dilanjutkan ke tahap berikutnya.

4.2.2. *Data Formatting*

Pada ringkasan data, terlihat bahwa variabel *num* masih terdiri dari kategori 0, 1, 2, 3, dan 4. Data dalam variabel akan diubah menjadi 2 kategori saja yaitu 0 (tidak terkena penyakit jantung) dan 1 (terkena penyakit jantung) sehingga perlu dilakukan pengubahan data. Data dengan kategori 2, 3, dan 4 akan diubah menjadi 1 sedangkan data dengan kategori 0 dan 1 dibiarkan saja. Setelah dilakukan pengubahan data, variabel *num* akan diubah menjadi faktor dengan level 0 dan 1.

```
> #mengubah variabel integer "num" "2", "3", dan "4" menjadi "1" sehingga hanya terdapat variabel "0" dan "1"
> heart$num[heart$num=="4"] <- "1"
> heart$num[heart$num=="3"] <- "1"
> heart$num[heart$num=="2"] <- "1"
> head(heart$num)
[1] "0" "1" "1" "0" "0" "0"
> #mengubah variabel num menjadi faktor
> heart$num <- as.factor(heart$num)
> class(heart$num)
[1] "factor"
> head(heart$num)
[1] 0 1 1 0 0 0
Levels: 0 1
```

Selain itu, terlihat bahwa terdapat dua variabel yang masih bertipe data karakter yaitu variabel *ca* (jumlah pembuluh darah) dan *thal* (detak jantung pasien). Pada tahap ini, kedua variabel tersebut yang semula bertipe karakter akan dikonversi menjadi numerik agar tipe data keseluruhan menjadi konsisten.

```
> #mengkonversi "thal" dan "ca" ke variabel angka
> heart$thal <- as.character(heart$thal)
> heart$thal <- as.numeric(heart$thal)
> str(heart)
'data.frame': 297 obs. of 14 variables:
 $ age      : num  63 67 67 37 41 56 62 57 63 53 ...
 $ sex      : num  1 1 1 1 0 1 0 0 1 1 ...
 $ cp       : num  1 4 4 3 2 2 4 4 4 4 ...
 $ trestbps : num  145 160 120 130 130 120 140 120 130 140 ...
 $ chol     : num  233 286 229 250 204 236 268 354 254 203 ...
 $ fbs      : num  1 0 0 0 0 0 0 0 0 1 ...
 $ restecg  : num  2 2 2 0 2 0 2 0 2 2 ...
 $ thalach  : num  150 108 129 187 172 178 160 163 147 155 ...
 $ exang    : num  0 1 1 0 0 0 0 1 0 1 ...
 $ oldpeak  : num  2.3 1.5 2.6 3.5 1.4 0.8 3.6 0.6 1.4 3.1 ...
 $ slope    : num  3 2 2 3 1 1 3 1 2 3 ...
 $ ca       : chr   "0.0" "3.0" "2.0" "0.0" ...
 $ thal     : num  6 3 7 3 3 3 3 3 7 7 ...
 $ num      : Factor w/ 2 levels "0","1": 1 2 2 1 1 1 2 1 2 2 ...
```

```

> heart$ca <- as.character(heart$ca)
> heart$ca <- as.numeric(heart$ca)
> str(heart)
'data.frame': 297 obs. of 14 variables:
 $ age      : num  63 67 67 37 41 56 62 57 63 53 ...
 $ sex      : num  1 1 1 1 0 1 0 0 1 1 ...
 $ cp       : num  1 4 4 3 2 2 4 4 4 4 ...
 $ trestbps : num  145 160 120 130 130 120 140 120 130 140 ...
 $ chol     : num  233 286 229 250 204 236 268 354 254 203 ...
 $ fbs      : num  1 0 0 0 0 0 0 0 0 1 ...
 $ restecg  : num  2 2 2 0 2 0 2 0 2 2 ...
 $ thalach  : num  150 108 129 187 172 178 160 163 147 155 ...
 $ exang     : num  0 1 1 0 0 0 0 1 0 1 ...
 $ oldpeak  : num  2.3 1.5 2.6 3.5 1.4 0.8 3.6 0.6 1.4 3.1 ...
 $ slope    : num  3 2 2 3 1 1 3 1 2 3 ...
 $ ca       : num  0 3 2 0 0 0 2 0 1 0 ...
 $ thal     : num  6 3 7 3 3 3 3 7 7 ...
 $ num      : Factor w/ 2 levels "0","1": 1 2 2 1 1 1 2 1 2 2 ...

```

4.3. Transformasi Data

Pada dataset ini dilakukan transformasi berupa normalisasi agar data berada dalam rentang yang sama yaitu antara 0 sampai 1.

```

> head(heart)
  age sex      cp trestbps      chol fbs restecg  thalach exang  oldpeak slope      ca thal num
1 0.7083333 1 0.0000000 0.4811321 0.2442922 1 1 0.6030534 0 0.3709677 1.0 0.0000000 0.75 0
2 0.7916667 1 1.0000000 0.6226415 0.3652968 0 1 0.2824427 1 0.2419355 0.5 1.0000000 0.00 1
3 0.7916667 1 1.0000000 0.2452830 0.2351598 0 1 0.4427481 1 0.4193548 0.5 0.6666667 1.00 1
4 0.1666667 1 0.6666667 0.3396226 0.2831050 0 0 0.8854962 0 0.5645161 1.0 0.0000000 0.00 0
5 0.2500000 0 0.3333333 0.3396226 0.1780822 0 1 0.7709924 0 0.2258065 0.0 0.0000000 0.00 0
6 0.5625000 1 0.3333333 0.2452830 0.2511416 0 0 0.8167939 0 0.1290323 0.0 0.0000000 0.00 0

```

Berdasarkan output terlihat bahwa semua variabel sudah berada dalam rentang data yang sama.

4.4. Split Data

Setelah tahap persiapan data selesai, dataset dibagi menjadi 2 yaitu data latih (data training) dan data uji (data testing). Data latih akan melatih model yang akan divalidasi dengan data uji untuk melihat keakuratan hasil klasifikasi. Dalam kasus ini, diambil proporsi data latih dan data uji sebesar 80 : 20. Artinya data latih diambil sebesar 80% dari dataset dan data uji diambil sebesar 20% dari dataset. Alasan menggunakan proporsi 80:20 agar dapat model dapat lebih banyak melakukan pelatihan tetapi pengujian yang dilakukan juga tidak sedikit.

4.5. Modelling dan Evaluasi Model

Langkah terakhir dari tahapan ini adalah *modelling* dan evaluasi model. Pada tahap ini, data akan diuji dengan algoritma K-Nearest Neighbour. Nilai K yang digunakan yaitu angka ganjil dimulai dari 3 hingga 11. Hasil pengujian beberapa nilai K diperoleh sebagai berikut.

k	Accuracy	Specificity	Sensitivity
3	87.69%	89.19%	85.71%
5	86.15%	89.19%	82.14%
7	86.15%	89.19%	82.14%
9	84.62%	86.49%	82.14%
11	83.08%	83.78%	82.14%

Berdasarkan hasil pengujian beberapa nilai K yang ditunjukkan pada tabel di atas, terlihat bahwa diperoleh nilai *specificity* yang sama yaitu 89.19% ketika nilai K yang diuji sebesar 3, 5, dan 7. Selain itu, terlihat pula bahwa nilai *accuracy* dan *sensitivity* tertinggi dihasilkan pada K=3. Dari informasi tersebut, dapat disimpulkan bahwa

performa terbaik dihasilkan pada $K=3$ dengan nilai *accuracy* 87.69%, *specificity* 89.19%, dan *sensitivity* 85.71%.

5. Kesimpulan

Berdasarkan hasil implementasi algoritma K-Nearest Neighbour untuk diagnosis penyakit jantung diperoleh kesimpulan sebagai berikut.

1. Perubahan nilai K pada K-Nearest Neighbour mempengaruhi performa model yang dapat dilihat dari nilai *accuracy*, *specificity*, dan *sensitivity*.
2. Pengujian nilai K dilakukan mulai dari nilai 3 sampai dengan 11. Berdasarkan hasil pengujian diperoleh nilai *specificity* yang sama ketika nilai K yang diuji yaitu 3, 5, dan 7. Namun, nilai $K=3$ menghasilkan nilai *accuracy* dan *sensitivity* tertinggi dibandingkan nilai K yang lain sehingga performa terbaik dari model diperoleh ketika nilai $K=3$.
3. Hasil performa terbaik dari pengujian beberapa nilai K dengan metode K-Nearest Neighbour menghasilkan nilai *accuracy* 87.69%, *specificity* 89.19%, dan *sensitivity* 85.71%.
4. Dengan menggunakan algoritma K-Nearest Neighbour diharapkan dapat membantu dalam mendeteksi adanya penyakit jantung dan memberikan diagnosis yang akurat sehingga tenaga kesehatan dan tim medis dapat memberikan perawatan dan tindak lanjut yang tepat.

Daftar Pustaka

Ahmad, A., & Latief, A. (2021). Perbandingan Metode KNN Dan LBPH Pada Klasifikasi Daun Herbal. Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi), 5(3), 557-564.

Rivki, M., & Bachtiar, A. M. (2017). Implementasi algoritma K-Nearest Neighbor dalam pengklasifikasian follower twitter yang menggunakan Bahasa Indonesia. Jurnal Sistem Informasi, 13(1), 31-37.

Yuniarto, M. S., & Sarwoko, E. A. (2020). Implementasi metode K-Nearest Neighbor untuk diagnosis kanker kolorektal dengan Biomarker Micro-RNA. Jurnal Masyarakat Informatika, 11(1), 35-48.

Syntax dapat diakses [di sini](#)

LinkedIn dan GitHub

LinkedIn : <https://www.linkedin.com/in/nurindahsetyaningsih/>

GitHub : <https://github.com/NurIndahSetyaningsih/NurIndahSetyaningsih.github.io>
<https://github.com/NurIndahSetyaningsih>