DATA MINING MODEL KLASIFIKASI MENGGUNAKAN ALGORITMA K-NEAREST NEIGHBOR DENGAN NORMALISASI UNTUK PREDIKSI PENYAKIT DIABETES

Muhammad Sholeh[1], Dina Andayati[2], Rr. Yuliana Rachmawati[3]
[1,3] Program Studi Informatika, Fakultas Teknologi Informasi dan Bisnis
[2]Program Studi Teknik Mesin, Fakultas Teknologi Industri
Institut Sains & Teknologi AKPRIND Yogyakarta
e-mail: [1]muhash@akprind.ac.id,[2]dina_asnawi@yahoo.com,[3]yuliana@akprind.ac.id

Abstrak

Model yang dibangun dengan menggunakan proses data mining dapat digunakan untuk melakukan prediksi dari suatu data. Model dapat dibangun dengan menggunakan datasheet yang berisi data yang diolah dari proses. Salah satu implementasi dari model dalam data mining adalah prediksi dari suatu penyakit seperti penyakit diabetes. Dalam penelitian ini, dilakukan pembuatan model data mining dengan menggunakan algoritma k-NN dan dilakukan normalisasi data. Metode normalisasi yang dilakukan adalah Z-Score dan Min-Max.. Metodologi penelitian dilakukan dengan terlebih dahulu melakukan menentukan datasheet, memilih model data mining serta membagi datasheet menjadi datasheet menjadi data training dan data testing serta melakukan evaluasi performance dari model yang dibuat . Proses pembuatan model menggunakan pemrograman python. Proses data mining menggunakan model klasifikasi dengan menggunakan algoritma k-NN. Datasheet yang digunakan merupakan datasheet public yaitu datasheet penyakit diabetes yang terdiri dari 768 record dan 8 atribut. Hasil dari pembuatan model ini menunjukkan proses normalisasi dapat memberikan nilai akurasi yang lebih baik. Model yang dikembangkan tanpa normalisasi menghasilkan nilai k=5 dengan akurasi 70%, normalisasi dengan metode Z-Score menghasilkan nilai k=21 dengan akurasi 72%, normalisasi dengan Min Max menghasilkan nilai k=3 dengan akurasi 74%. Model yang direkomendasi merupakan mode k-NN dengan nilai k=3.

Kata Kunci: Model, Data mining, k-NN, Normalisasi

DATA MINING MODEL CLASSIFICATION USING ALGORITHM K-NEAREST NEIGHBOR WITH NORMALIZATION FOR DIABETES PREDICTION

Abstract

The model built using the data mining process can be used to make predictions from the data. The model can be built using a datasheet that contains data that is processed from the process. One implementation of the model in data mining is the prediction of a disease such as diabetes. In this study, a data mining model was developed using the k-NN algorithm and data normalization was carried out. The normalization method used is Z-Score and Min-Max. The research methodology is carried out by first determining the datasheet, selecting the data mining model and dividing the datasheet into datasheets into training data and data testing and evaluating the performance of the model created. The process of modeling using python programming. The data mining process uses a classification model using the k-NN algorithm. The datasheet used is a public datasheet, namely the diabetes datasheet which consists of 768 records and 8 attributes. The results of this modeling show that the normalization process can provide better accuracy values. The model developed without normalization produces a value of k=5 with an accuracy of 70%, normalization with the Z-Score method produces a value of k=21 with an accuracy of

72%, normalization with Min Max produces a value of k=3 with an accuracy of 74%. The recommended model is k-NN mode with a value of k=3.

Keywords: *Keywords: Model, Data mining, k-NN, Normalization*

1. Pendahuluan

Diabetes menjadi salah satu jenis penyakit yang termasuk paling banyak penderita di Indonesia. Diabetes adalah gangguan metabolisme kronis atau kronis yang disebabkan oleh ketidakmampuan pankreas dalam memproduksi insulin yang diperlukan tubuh. Diabetes juga dikenal sebagai silent killer karena sering tidak disadari. Diabetes disebabkan oleh berbagai faktor, termasuk faktor genetik berat badan, kurang olahraga, usia, tekanan darah tinggi, dan kadar kolesterol dan trigliserida yang tinggi [1]. Jika diabetes tidak ditangani dengan benar, komplikasi yang lebih serius dapat terjadi dan dapat berdampak pada gangguan pada jantung serta pembuluh darah dan jenis kerusakan organ tubuh lainnya.

Jumlah penduduk dunia yang mengidap penyakit diabetes diperkirakan sebanyak 427 juta. Penderita diabetes berusia 20-79 tahun. Dari jumlah penderita tersebut 212 juta tidak menyadari terkena penyakit diabetes. Empat dari lima pasien diabetes berasal dari negara dengan pendapatan rendah atau menengah (termasuk Indonesia) dan dua pertiga tinggal di perkotaan [1]. Walaupun penyakit diabetes masuk kategori penyakit yang berbahaya, penderita masih dapat menjalankan aktivitas dengan berbagai syarat menjaga pola hidup dan kesehatan [2].

Banyak penelitian yang mengupas penyakit diabetes baik dari sisi ilmu kesehatan [3][4][5] maupun dari bidang non kesehatan tetapi masih terkait dengan bidang kesehatan. Penelitian diabetes di luar ilmu kesehatan di antaranya adalah penelitian yang mengembangkan aplikasi sistem pakar dalam melakukan diagnosis suatu penyakit atau penelitian data mining yang dapat digunakan untuk melakukan prediksi dalam menentukan termasuk terkena risiko penyakit diabetes atau tidak terkena. [6][7][8].

Penelitian Lhaha [9], membuat data mining untuk membuat prediksi mengenai resiko seseorang terkena diabetes atau data. Data mining dibuat dengan membuat model prediksi yang dapat melakukan prediksi dengan memberikan data-data yang terkait dengan penyakit diabetes. Penelitian data mining dengan prediksi penyakit diabetes dilakukan Azrar, [10], model data mining diimplementasikan dengan algoritma decision Tree dan hasil akurasi terbaik sebesar 75,65%. Model dikembangkan Rapid Miner dan rasio data training dengan data test adalah 70:30.

Data mining merupakan proses melakukan pengumpulan dan melakukan pengolahan data yang bertujuan untuk mengekstraksi suatu informasi. Proses untuk mengumpulkan dan menggali informasi dilakukan dengan menggunakan aplikasi atau perangkat lunak dengan bantuan teknik komputasi statistik, matematika atau kecerdasan buatan [11]. Proses dalam data mining dibedakan menjadi beberapa model yaitu model regresi, forescating, klasifikasi, klastering dan asosiasi [12][13]. Model yang dapat digunakan dalam data mining dapat menggunakan model regresi, forescating, klasifikasi, klastering dan asosiasi. Salah model yang dapat digunakan dalam melakukan prediksi adalah model klasifikasi. Model klasifikasi merupakan suatu model yang dapat menemukan atau melakukan prediksi dalam kesamaan fitur dalam suatu kelompok atau kelas. Metode klasifikasi melakukan prediksi pada datasheet yang mempunyai atribut berupa label[14][15].

Penelitian data mining yang menggunakan model klasifikasi dalam melakukan prediksi diteliti oleh Novita. Data mining yang dikembangkan adalah digunakan untuk melakukan klasifikasi anggrek. Model klasifikasi dikembangkan dengan algoritma K-Nearest Neighbor. Sistem ini dirancang untuk membantu masyarakat dalam mengidentifikasi tanaman anggrek berdasarkan genus dan varietasnya. Pengujian sistematis dilakukan pada 15 data latih, dan akurasi yang dihasilkan adalah 53,33%, dengan total 8 nilai benar dan 7 nilai salah. [16]. Penelitian lain yang menggunakan algoritma k-NN untuk melakukan prediksi di antaramya [17][18][19].

Proses pembuatan model klasifikasi dapat menggunakan algoritma k-NN. Algoritma KNN melakukan proses prediksi dengan melakukan pendekatan terdekat pada data yang ada dalam datasheet. Pendekatan pada data ditentukan dengan mengambil sejumlah K. Nilai K digunakan sebagai acuan untuk menentukan kelas dari data baru. Algoritma kNN dapat mengklasifikasikan data berdasar pada kemiripan atau kedekatan data yang dicari terhadap data lainnya [14].

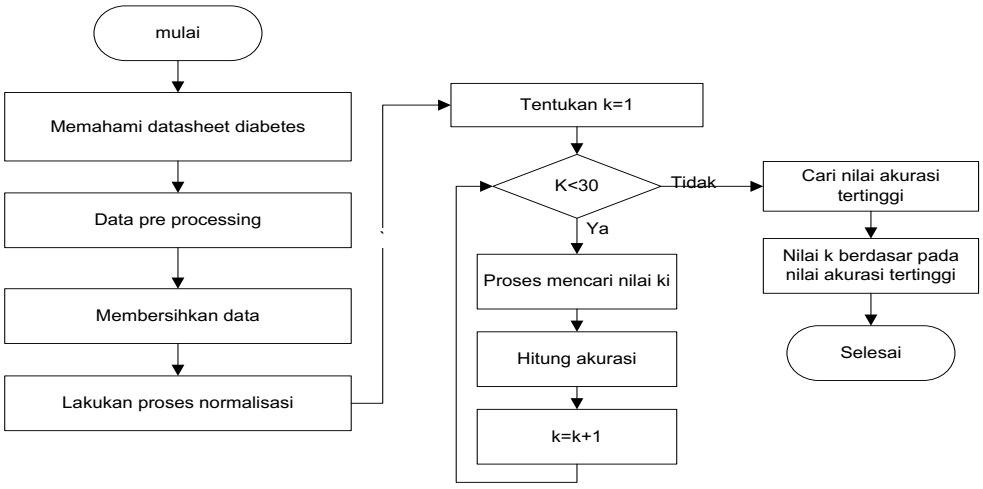
Data yang diolah dalam proses dalam pembuatan model dengan menggunakan algoritma k-NN juga harus memperhatikan sebaran data. Data yang mempunyai sebaran yang tinggi perlu dilakukan proses normalisasi. Normalisasi data merupakan proses untuk membuat beberapa data atau variabel memiliki rentang nilai yang sama sehingga masing-masing data mempunyai selisih nilai yang kecil. Proses normalisasi tidak menghilangkan isi data asli dan dalam proses data mining akan membuat proses analisis statistik menjadi lebih mudah [20][21].

Penelitian yang menggunakan normalisasi untuk memperkecil rentang data/ variabel dilakukan Nasution [22]. Dalam penelitian yang dilakukan Nasution, proses normalisasi digunakan untuk membuat rentang nilai yang seimbang pada setiap atribut. Penelitian lain yang menggunakan proses normalisasi dilakukan [23][24][25].

Berdasar pada latar belakang dan tinjauan pustaka, data mining terutama model klastering dapat digunakan untuk mengetahui suatu kelompok dari data yang diolah. Pembuatan model dapat menggunakan algoritma seperti k-NN. Algoritma k-NN digunakan untuk membuat model yang daat digunakan untuk melakukan prediksi penyakit diabetes. Model dikembangkan dengan membuat model dari datasheet penderita diabetes dan agar didapat rentang data yang tidak terlalu jauh, proses pembuatan model akan dilakukan normalisasi dengan menggunakan metode normalisasi simple feature scaling, Min-Max dan Z-score.

2. Metode Penelitian

Metodologi penelitian dilakukan dengan terlebih dahulu melakukan menentukan datasheet, memilih model data mining serta membagi datasheet menjadi datasheet menjadi data training dan data testing serta melakukan evaluasi performance dari model yang dibuat. Gambar 1, tahapan penelitian yang dilakukan.



Gambar 1 Proses tahapan dalam penelitian

Datasheet.

Datasheet yang digunakan merupakan datasheet public yang berisi ciri-ciri penyakit diabetes. Sumber datasheet adalah https://www.kaggle.com/datasets/mathchi/diabetes-data-set. Tabel 1, contoh datasheet diabetes

Tabel 1 Contoh datasheet ciri-ciri penyakit diabetes

Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes PedigreeFunction	Age	Outcome
0	125	96	0	0	22.5	0.262	21	0
1	81	72	18	40	26.6	0.283	24	0
2	85	65	0	0	39.6	0.93	27	0
1	126	56	29	152	28.7	0.801	21	0
1	96	122	0	0	22.4	0.207	27	0
4	144	58	28	140	29.5	0.287	37	0
3	83	58	31	18	34.3	0.336	25	0

Algoritma K-Nearest Neighbour (KNN)

Proses dari K-NN dilakukan dengan mencari jarak terdekat antara data yang akan dilakukan prediksi dengan data terdekat. Banyaknya data terdekat yang akan dibandingkan ditentukan sebanyak k. Nilai k yang digunakan berjumlah ganjil [26][27]. Dalam eksperimen ini, nilai dihitung mulai 1 sampai 30 dan nilai k dipilih yang mempunyai nilai akurasi yang paling tinggi. Gambar 2 rumus perhitungan dalam mencari data terdekat dalam algoritma kN

$$dis(x_1, x_2) = \sqrt{\sum_{i=0}^{n} (x_{1i} - x_{2i})^2} \quad (1)$$

Gambar 2 Rumus algoritma k-NN

Normalisasi data

Data yang mempunyai rentang selisih cukup banyak dapat menghasilkan model yang mempunyai nilai akurasi kecil. Upaya untuk mengubah data yang mempunyai rentang cukup banyak dapat dilakukan proses normalisasi data. Perubahan yang dilakukan dengan normalisasi tidak mengakibatkan berubahnya informasi [27]. Metode normalisasi yang digunakan dalam eksperimen adalah

Min-Max

Metode normalisasi dilakukan dengan melakukan pengurangan setiap data pada nilai yang ada pada fitur dengan nilai minimum fitur dan hasilnya dilakukan pembagian nilai maksimum dikurangi dengan nilai minimum dari fitur tersebut [27]. Gambar 3 rumus metode min-max

$$x_{new} = \frac{x_{old} - x_{min}}{x_{max} - x_{min}} \quad (2)$$

Gambar 3 Rumus metode normalisasi min-max

Z-score

Metode ini dihitung dengan melakukan pencarian nilai ukuran penyimpangan data dari hasil nilai dari rata-rata yang diukur dalam satuan standar deviasi [27]. Gambar 4 rumus metode Z-score

$$x_{new} = \frac{x_{old} - \mu}{\sigma} \quad (3)$$

Gambar 4 Rumus metode normalisasi Z-score

Evaluasi

Pemilihan model yang akan digunakan didasarkan pada hasil performance dari model yang dibuat. Evaluasi yang digunakan dalam mengukur model klasifikasi dapat menggunakan Confusion matrix. Confusion matrik terdiri tabel yang berisi True Positive (TP), False Positive (FP), False Negative (FN), dan True Negative (TN) [27]. Gambar 5 tabel confusion matrik dan gambar 6 rumus perhitungan accuracy.

Nilai Aktual

Nilai Prediksi	Positive	Negative
Positive	TP	FP
Negative	FN	TN

Gambar 5 Tabel confusion matrik

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (4)$$

Gambar 6 Rumus perhitungan accuracy.

3. Hasil

Model data mining yang dibuat menggunakan algoritma k-NN dan sebelum pembuatan model dilakukan proses normalisasi dengan 3 metode yaitu Simple Feature Scaling, min-max da z score. Hasil dari proses pembuatan model akan dipilih nilai k yang paling baik. Pemilihan nilai k dilakukan dengan membandingkan nilai performance pada proses evaluasi.

Tahap normalisasi dengan Z-Score

Implementasi normalisasi Z-Score dengan python menggunakan library sklearn terutama perintah StandardScaler

from sklearn.preprocessing import StandardScaler

X = StandardScaler().fit(df_features).transform(df_features.astype(float)).

Program yang digunakan proses normalisasi dengan Z-Score dan hasilnya ditampilkan pada gambar 7.

```
# Normalisasi data  dengan algoritma Z
from sklearn.preprocessing import StandardScaler
X = StandardScaler().fit(df_features).transform(df_features.astype(float))
X_dataframe = pd.DataFrame(X)
X_dataframe
```

	0	1	2	3	4	5	6	7
0	0.639947	0.848324	0.149641	0.907270	-0.692891	0.204013	0.468492	1.425995
1	-0.844885	-1.123396	-0.160546	0.530902	-0.692891	-0.684422	-0.365061	-0.190672
2	1.233880	1.943724	-0.263941	-1.288212	-0.692891	-1.103255	0.604397	-0.105584
3	-0.844885	-0.998208	-0.160546	0.154533	0.123302	-0.494043	-0.920763	-1.041549
4	-1.141852	0.504055	-1.504687	0.907270	0.765836	1.409746	5.484909	-0.020496
...	...	...	...	...	...	...	...	...
763	1.827813	-0.622642	0.356432	1.722735	0.870031	0.115169	-0.908682	2.532136
764	-0.547919	0.034598	0.046245	0.405445	-0.692891	0.610154	-0.398282	-0.531023
765	0.342981	0.003301	0.149641	0.154533	0.279594	-0.735190	-0.685193	-0.275760
766	-0.844885	0.159787	-0.470732	-1.288212	-0.692891	-0.240205	-0.371101	1.170732
767	-0.844885	-0.873019	0.046245	0.656358	-0.692891	-0.202129	-0.473785	-0.871374

768 rows × 8 columns

Gambar 7 Proses Normalisasi dengan Z-Score

Tahap pembagi data training dan data test

Proses pembuatan model dan pengujian untuk mendapatkan nilai k terbaik dilakukan dengan membuat model dengan data training dan menguji model dengan data test. Pembagian data training dan data testing masing-masing adalah 80% dan 20%. Proses pembagian data dan hasilnya ditampilkan pada gambar 8.

```
# Train test split untuk membagi data training dan testing

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=10)

print ('Train set:', X_train.shape,  y_train.shape)
print ('Test set:', X_test.shape,  y_test.shape)
```

```
Train set: (614, 8) (614,)
Test set: (154, 8) (154,)
```

Gambar 8 Pembagian data training dan data testing

Tahap pengujian model

Nilai K yang dipilih diuji dengan melakukan proses pengulangan nilai k dari 1 sampai 30. Masing-masing nilai k diuji pada model dan mendapatkan nilai akurasi. Nilai akurasi yang tinggi menjadi pilihan untuk menentukan nilai k. Hasil pengujian sebanyak k=30 didapat nilai akurasi tertinggi pada nilai k=21 dengan nilai akurasi 72%. Hasil pengujian model dengan proses pengulangan dari k=1 sampai k=30 ditampilkan pada gambar 9, gambar 10 menampilkan akurasi dari masing-masing pengujian k dan gambar 11 hasil masing-masing nilai akurasi dalam bentuk grafik.

```
# Mencari nilai K dengan akurasi terbaik
from sklearn.metrics import accuracy_score
K = 30
mean_acc = np.zeros((K-1))

for n in range(1, K):

    #Train Model and Predict
    model_knn = KNeighborsClassifier(n_neighbors = n).fit(X_train, y_train)
    y_pred = model_knn.predict(X_test)

    mean_acc[n-1] = accuracy_score(y_test, y_pred)


print( 'Akurasi terbaik adalah ', mean_acc.max(), 'dengan nilai k =', mean_acc.argmax()+1)

```

```
Akurasi terbaik adalah  0.7207792207792207 dengan nilai k = 21
```

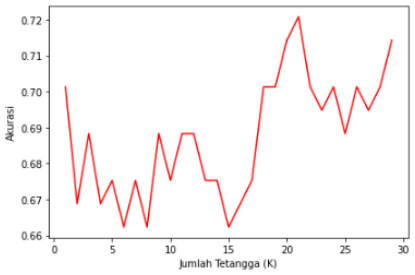
Gambar 9 proses pengujian k dari 1 – 30 dan hasil akurasi tertinggi

```
for n in range(1, K,2):
    print(n,"  ",mean_acc[n-1])
```

```
1   0.7012987012987013
3   0.6883116883116883
5   0.6753246753246753
7   0.6753246753246753
9   0.6883116883116883
11   0.6883116883116883
13   0.6753246753246753
15   0.6623376623376623
17   0.6753246753246753
19   0.7012987012987013
21   0.7207792207792207
23   0.6948051948051948
25   0.6883116883116883
27   0.6948051948051948
29   0.7142857142857143
```

Gambar 10 Hasil akurasi proses pengujian dari k=1-30

```
plt.plot(range(1,K), mean_acc, 'r')
plt.ylabel('Akurasi')
plt.xlabel('Jumlah Tetangga (K)')
plt.tight_layout()
plt.show()
```



Gambar 11 Hasil masing-masing akurasi dalam bentuk grafik

Hasil proses nilai akurasi dapat dilihat dalam bentuk confusion matrik. Hasil perhitungan True Positive (TP) =87, False Positive (FP)=8, False Negative (FN)=36, dan True Negative (TN)=23. Proses dan Hasil dalam bentuk confusion matrik ditampilkan pada gambar 12

```
from sklearn.metrics import confusion_matrix
model_knn = KNeighborsClassifier(n_neighbors = mean_acc.argmax()).fit(X_train, y_train)
y_pred = model_knn.predict(X_test)
confusion_matrix(y_test,y_pred)
pd.crosstab(y_test, y_pred, rownames=['True'], colnames=['Predicted'], margins=True)
```

Predicted	0	1	All
True			
0	87	8	95
1	36	23	59
All	123	31	154

Gambar 12 Matrik Confusion untuk nilai k =21

Tahap normalisasi dengan Min-Max

Implementasi normalisasi Min-Max dengan python menggunakan library sklearn terutama perintah MinMaxScaler

from sklearn.preprocessing import MinMaxScaler
X = MinMaxScaler ().fit(df_features).transform(df_features.astype(float)).

Program yang digunakan untuk melakukan proses normalisasi dengan Min-Max dan hasilnya ditampilkan pada gambar 13

```
# Normalisasi data  dengan algoritma Mix-Max
from sklearn.preprocessing import MinMaxScaler
X = MinMaxScaler().fit(df_features).transform(df_features.astype(float))
X_dataframe = pd.DataFrame(X)
X_dataframe
```

	0	1	2	3	4	5	6	7
0	0.352941	0.743719	0.590164	0.353535	0.000000	0.500745	0.234415	0.483333
1	0.058824	0.427136	0.540984	0.292929	0.000000	0.396423	0.116567	0.166667
2	0.470588	0.919598	0.524590	0.000000	0.000000	0.347243	0.253629	0.183333
3	0.058824	0.447236	0.540984	0.232323	0.111111	0.418778	0.038002	0.000000
4	0.000000	0.688442	0.327869	0.353535	0.198582	0.642325	0.943638	0.200000
...	...	...	...	...	...	...	...	...
763	0.588235	0.507538	0.622951	0.484848	0.212766	0.490313	0.039710	0.700000
764	0.117647	0.613065	0.573770	0.272727	0.000000	0.548435	0.111870	0.100000
765	0.294118	0.608040	0.590164	0.232323	0.132388	0.390462	0.071307	0.150000
766	0.058824	0.633166	0.491803	0.000000	0.000000	0.448584	0.115713	0.433333
767	0.058824	0.467337	0.573770	0.313131	0.000000	0.453055	0.101196	0.033333

768 rows × 8 columns

Gambar 13 Proses Normalisasi dengan Min-Max

Tahap penguji model

Nilai K yang dipilih diuji dengan melakukan proses pengulangan nilai k dari 1 sampai 30. Masing-masing nilai k diuji pada model dan mendapatkan nilai akurasi. Nilai akurasi yang tinggi menjadi pilihan untuk menentukan nilai k. Hasil pengujian sebanyak k=30 didapat nilai akurasi tertinggi pada nilai k=21 dengan nilai akurasi 72%. Hasil pengujian model dengan proses pengulangan dari k=1 sampai k=30 ditampilkan pada gambar 14, gambar 15 menampilkan akurasi dari masing-masing pengujian k dan gambar 16 hasil masing-masing nilai akurasi dalam bentuk grafik.

```
# Mencari nilai K dengan akurasi terbaik
from sklearn.metrics import accuracy_score
K = 30
mean_acc = np.zeros((K-1))

for n in range(1, K):

    #Train Model and Predict
    model_knn = KNeighborsClassifier(n_neighbors = n).fit(X_train, y_train)
    y_pred = model_knn.predict(X_test)

    mean_acc[n-1] = accuracy_score(y_test, y_pred)


print( 'Akurasi terbaik adalah ', mean_acc.max(), 'dengan nilai k =', mean_acc.argmax()+1)
```

```
Akurasi terbaik adalah  0.7467532467532467 dengan nilai k = 3
```

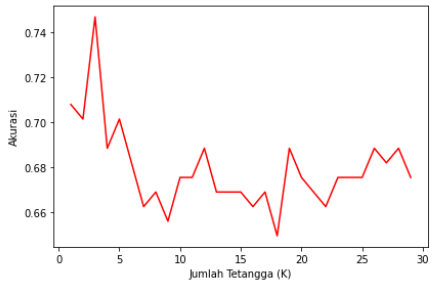
Gambar 14 Proses pengujian k dari 1 – 30 dan hasil akurasi tertinggi

```
for n in range(1, K,2):
     print(n,"  ",mean_acc[n-1])
```

```
1    0.7077922077922078
3    0.7467532467532467
5    0.7012987012987013
7    0.6623376623376623
9    0.6558441558441559
11    0.6753246753246753
13    0.6688311688311688
15    0.6688311688311688
17    0.6688311688311688
19    0.6883116883116883
21    0.6688311688311688
23    0.6753246753246753
25    0.6753246753246753
27    0.6818181818181818
29    0.6753246753246753
```

Gambar 15 Hasil akurasi proses pengujian dari k=1-30

```
plt.plot(range(1,K), mean_acc, 'r')
plt.ylabel('Akurasi')
plt.xlabel('Jumlah Tetangga (K)')
plt.tight_layout()
plt.show()
```



Gambar 16 Hasil masing-masing akurasi dalam bentuk grafik

Hasil proses nilai akurasi dapat dilihat dalam bentuk confusion matrik. Hasil perhitungan True Positive (TP) =90, False Positive (FP)=5, False Negative (FN)=41, dan True Negative (TN)=18. Proses dan Hasil dalam bentuk confusion matrik ditampilkan pada gambar 17.

```
from sklearn.metrics import confusion_matrix
model_knn = KNeighborsClassifier(n_neighbors = mean_acc.argmax()).fit(X_train, y_train)
y_pred = model_knn.predict(X_test)
confusion_matrix(y_test,y_pred)
pd.crosstab(y_test, y_pred, rownames=['True'], colnames=['Predicted'], margins=True)
```

Predicted	0	1	All
True			
0	90	5	95
1	41	18	59
All	131	23	154

Gambar 17 Matrik Confusion untuk nilai k =3

Evaluasi Model

Pemilihan model terutama pemilihan nilai k, dilakukan dengan membandingkan nilai akurasi dari masing-masing metode normalisasi yang digunakan. Hasil perbandingan akurasi disajikan pada tabel 2.

Tabel 2 Hasil perbandingan akurasi

Pengujian	Nilai K	akurasi
Tanpa normalisasi	5	70%
Normalisasi dengan Z-Score	21	72%
Normalisasi dengan Min-Max	3	74%

Berdasar pada tabel 2, nilai k yang dipilih adalah 2. Hal ini karena hasil nilai akurasi tertinggi sebesar 74%. Nilai akurasi ini merupakan perhitungan dengan metode normalisasi Min-Max.

4. Kesimpulan

Salah satu model data mining yang dapat digunakan untuk melakukan prediksi adalah dengan model klasifikasi. Salah satu algoritma dalam pembuatan model klasifikasi adalah kNN. Dalam penelitian dengan menggunakan datasheet penyakit diabetes, model klasifikasi yang dikerjakan dengan k-NN dilakukan proses pengujian nilai dari k=1 sampai k=30. Proses pengujian dilakukan dengan melakukan normalisasi data dengan menggunakan Z-Score, Min-Max dan tanpa melakukan normalisasi. Hasil pengujian menghasilkan nilai akurasi yang terbaik adalah dengan melakukan normalisasi Min-Max dan diperoleh nilai k=3 dengan nilai akurasi 74%.

5. Daftar Pustaka

[1] H. Tandra, *Penderita Diabetes Boleh Makan Apa Saja*. Jakarta: Gramedia Pustaka Utama, 2021.

[2] V. Tjahjadi, *Mengenal, Mencegah, Mengatasi Silent Killer, "Diabetes."* Jakarta: Hikam Pustaka, 2017.

[3] D. W. Hestiana, "FAKTOR-FAKTOR YANG BERHUBUNGAN DENGAN KEPATUHAN DALAM PENGELOLAAN DIET PADA PASIEN RAWAT JALAN DIABETES MELLITUS TIPE 2 DI KOTA SEMARANG," *Jurnal of Health Education*, vol. 2, no. 2, pp. 138–145, 2017.

[4] Z. M. Syahid, "Literature Review Faktor yang Berhubungan dengan Kepatuhan Pengobatan Diabetes Mellitus," *JIKSH : Jurnal Ilmiah Kesehatan Sandi Husada*, vol. 10, no. 1, pp. 147–155, 2021.

[5] I. Istianah, Septiani, and G. K. Dewi, "Mengidentifikasi Faktor Gizi pada Pasien Diabetes Mellitus Tipe 2 di Kota Depok Tahun 2019," *Jurnal Kesehatan Indonesia (The Indonesian Journal of Health)*, vol. X, no. 2, pp. 72–78, 2020.

[6] M. Shouman, T. Turner, and R. Stocker, "Applying k-Nearest Neighbour in Diagnosing Heart Disease

Patients," *Applying k-Nearest Neighbour in Diagnosing Heart Disease Patients*, vol. 2, no. 3, pp. 220–223, 2012.

[7] S. Wiyono and T. Abidin, "Implementation of K-Nearest Neighbour (Knn) Algorithm To Predict Student'S Performance," *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, vol. 9, no. 2, pp. 873–878, 2018, doi: 10.24176/simet.v9i2.2424.

[8] S. A. D. Alalwan, "Diabetic analytics: Proposed conceptual data mining approaches in type 2 diabetes dataset," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 1, pp. 85–95, 2019, doi: 10.11591/ijeecs.v14.i1.pp88-95.

[9] O. Llaha and A. Rista, "Prediction and detection of diabetes using machine learning," in *CEUR Workshop Proceedings*, 2021, vol. 2872, pp. 94–102.

[10] A. Azrar, M. Awais, Y. Ali, and K. Zaheer, "Data mining models comparison for diabetes prediction," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 8, pp. 320–323, 2018, doi: 10.14569/ijacsa.2018.090841.

[11] D. Cielen, A. D. B. Meysman, and M. Ali, *Introducing Data Science*. 2016.

[12] M. Arhami and M. Nasir, *Data Mining - Algoritma dan Implementasi*. Yogyakarta: Penerbit Andi, 2020.

[13] D. Jollyta, W. Ramdhan, and M. Zarlis, *Konsep Data Mining Dan Penerapan*. Yogyakarta: Deepublish Publisher, 2020.

[14] A. Wanto *et al.*, *Data Mining : Algoritma dan Implementasi*. Medan: Yayasan Kita Menulis, 2020.

[15] Suyanto, *Data Mining untuk Klasifikasi dan Klasterisasi Data*. Bandung: Informatika, 2017.

[16] S. Novita, P. Harsani, and A. Qurania, "Penerapan K-Nearest Neighbor (KNN) untuk Klasifikasi Anggrek Berdasarkan Karakter Morfologi Daun dan Bunga," *KOMPUTASI*, vol. 15, no. 1, pp. 118–125, 2018.

[17] Y. Yahya and W. Puspita Hidayanti, "Penerapan Algoritma K-Nearest Neighbor Untuk Klasifikasi Efektivitas Penjualan Vape (Rokok Elektrik) pada 'Lombok Vape On,'" *Infotek : Jurnal Informatika dan Teknologi*, vol. 3, no. 2, pp. 104–114, 2020, doi: 10.29408/jit.v3i2.2279.

[18] N. Hidayati and A. Hermawan, "K-Nearest Neighbor (K-NN) algorithm with Euclidean and Manhattan in classification of student graduation," *Journal of Engineering and Applied Technology*, vol. 2, no. 2, pp. 86–91, 2021, doi: 10.21831/jeatech.v2i2.42777.

[19] P. Cunningham and S. J. Delany, "K-Nearest Neighbour Classifiers-A Tutorial," *ACM Computing Surveys*, vol. 54, no. 6, 2021, doi: 10.1145/3459665.

[20] B. Santosa and A. Umam, *Buku Data Mining dan Big Data Analytics*. Bantul: Penebar Media Pustaka, 2018.

[21] M. Fhadli and F. Tempola, *Data Mining dengan Python untuk Pemula*. Bogor: Guepedia, 2020.

[22] D. A. Nasution, H. H. Khotimah, and N. Chamidah, "Perbandingan Normalisasi Data untuk Klasifikasi Wine Menggunakan Algoritma K-NN," *Computer Engineering, Science and System Journal*, vol. 4, no. 1, p. 78, 2019, doi: 10.24114/cess.v4i1.11458.

[23] Ahmad Harmain, P. Paiman, H. Kurniawan, K. Kusrini, and Dina Maulina, "Normalisasi Data Untuk Efisiensi K-Means Pada Pengelompokan Wilayah Berpotensi Kebakaran Hutan Dan Lahan Berdasarkan Sebaran Titik Panas," *TEKNIMEDIA: Teknologi Informasi dan Multimedia*, vol. 2, no. 2, pp. 83–89, 2022, doi: 10.46764/teknimedia.v2i2.49.

[24] H. A. Prihanditya and A. Alamsyah, "The Implementation of Z-Score Normalization and Boosting Techniques to Increase Accuracy of C4.5 Algorithm in Diagnosing Chronic Kidney Disease," *Journal of Soft Computing Exploration*, vol. 1, no. 1, pp. 63–69, 2020.

[25] E. Alshdaifat, "The Impact of Data Normalization on Predicting Student Performance: A Case Study from Hashemite University," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 4, pp. 4580–4588, 2020, doi: 10.30534/ijatcse/2020/57942020.

[26] Provost & Fawcett, "Data science-what you need to know about analytic-thinking and decision-making," *Journal of Chemical Information and Modeling*, vol. 53, no. 9, pp. 1689–1699, 2013.

[27] Jiawei Han and M. Kamber, *Data Mining: Concepts and Techniques*. 2019.

.