

## **PENGEMBANGAN BAHAN PENGAJARAN**



### **Program Komputer sebagai Alat Bantu dalam Perkuliahan Pengenalan Pola: Implementasi Algoritma K-Nearest Neighbor (KNN) untuk Memprediksi Saham Menggunakan Bahasa Pemrograman Python**

**Dibuat oleh:**

**Nur Uddin, PhD  
NIDN: 0325067804**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI DAN DESAIN  
UNIVERSITAS PEMBANGUNAN JAYA**

**TANGERANG SELATAN**

**Genap 2019/2020**

# Implementasi Algoritma K-Nearest Neighbor (KNN) untuk

## ▼ Memprediksi Saham Menggunakan Bahasa Pemrograman Python

Dibuat oleh: Nur Uddin

Kegunaan: Alat bantu dalam perkuliahan Kecerdasan Buatan guna memahami proses pembuatan sebuah machine learning.

Program ini menyajikan sebuah contoh bagaimana membangun sebuah machine learning untuk memprediksi harga saham menggunakan bahasa pemrograman Python. Algoritma yang digunakan dalam machine learning adalah K-nearest neighbor (KNN). Studi kasus dalam contoh ini adalah memprediksi harga saham PT. Bank Negara Indonesia 46 dengan kode saham BBNI. Program Python ini ditulis dengan menggunakan Google Colab yang merupakan sebuah aplikasi cloud computing. Tahapan pembuatan program adalah sebagai berikut:

1) Import library yang diperlukan:

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.neighbors import KNeighborsRegressor

from sklearn.metrics import confusion_matrix
from sklearn.metrics import f1_score
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error

import matplotlib.pyplot as plt
```

2) Menghubungkan Google Collab dengan Google Drive, dimana Google Drive merupakan tempat penyimpanan dataset yang digunakan dalam pembuatan machine learning ini.

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/c
```



3) Membaca dataset yang tersimpan dalam Google Drive dan menyimpannya sebagai variable **ds** :

```
ds = pd.read_excel('../content/drive/MyDrive/Data_AI/BBNI/data_BBNI_Jan2021_15Mar2022-tutorial.xls')
```

4) Melihat dan memeriksa isi dataset dengan menampilkan data pada lima baris pertama:

```
ds.head()
```

	Date	x(n-5)	x(n-4)	x(n-3)	x(n-2)	x(n-1)	x(n)	
0	2021-01-11	6326.403809	6450.450684	6351.212891	6376.022461	6425.641113	6475.260254	645
1	2021-01-12	6450.450684	6351.212891	6376.022461	6425.641113	6475.260254	6450.450684	647
2	2021-01-13	6351.212891	6376.022461	6425.641113	6475.260254	6450.450684	6475.260254	640
3	2021-01-14	6376.022461	6425.641113	6475.260254	6450.450684	6475.260254	6400.832031	627
4	2021-01-15	6425.641113	6475.260254	6450.450684	6475.260254	6400.832031	6276.784668	625

Data diatas menunjukkan data tanggal (Date) dan harga saham (x), dimana  $x(n)$  adalah harga saham pada tanggal tersebut,  $x(n+1)$  adalah harga saham pada satu hari sesudahnya, dan  $x(n-k)$  adalah harga saham pada k hari sebelumnya.

5) Melihat ukuran dataset untuk mengetahui jumlah baris dan kolom:

```
ds.shape
```

```
(289, 8)
```

6) Mendefinisikan input (fitur, X) dan output (target, y) dari machine learning. Dalam contoh ini, program dibuat untuk memprediksi harga saham satu hari sesudahnya sehingga  $x(n+1)$  merupakan target. Sementara, harga saham hari ini  $x(n)$  hingga lima hari sebelumnya  $x(n-5)$  digunakan sebagai fitur.

```
# split dataset
X = ds.iloc[:, 1:7]
y = ds.iloc[:, 7]
```

7) Membagi dataset menjadi data training dan data test. Dalam contoh ini, 80% data digunakan untuk training dan 20% digunakan untuk test, dan pembagian data tersebut dilakukan secara random.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=1, test_size=0.2)
```

8) Mendefinisikan sebuah model machine learning berbasis algoritma KNN

```
# Define the model: Init K-NN
model =KNeighborsRegressor(n_neighbors=5, p=2,metric='euclidean')
```

9) Melakukan proses training pada machine learning dengan menggunakan data training yang telah didefinisikan pada langkah 7.

```
# Fit Model
model.fit(X_train, y_train)
```

```
KNeighborsRegressor(metric='euclidean')
```

10) Menggunakan machine learning yang telah dibangun untuk memprediksi harga saham dengan menggunakan data test. Data test ini berisikan fitur X\_test dan target y\_test. Machine learning diberikan input berupa X\_test untuk menghitung harga prediksi yang disebut y\_pr. Hasil prediksi yang diperoleh (y\_pr) selanjutnya dibandingkan dengan target (y\_test) yang merupakan harga saham aktual.

```
# Predict the test set results
```

```
y_pr = model.predict(X_test)
```

```
y_pr
```

```
array([[5000.      , 7305.      , 6865.      , 5780.      ,
        4836.      , 6053.5     , 5305.      , 6655.30859375,
        5365.      , 7085.      , 5310.      , 6142.81386719,
        5136.      , 5415.      , 5904.64335937, 6275.90449219,
        7905.      , 5720.      , 6795.      , 7045.      ,
        4746.      , 4818.      , 5505.      , 5031.      ,
        5270.      , 5335.      , 7830.      , 7045.      ,
        5889.7578125, 6795.      , 6910.      , 5355.      ,
        5600.      , 7110.      , 4744.      , 7145.      ,
        6735.      , 7795.      , 6850.      , 7790.      ,
        7755.      , 7890.      , 5345.      , 7000.      ,
        5765.      , 4744.      , 5325.      , 5160.      ,
        7110.      , 5420.      , 6995.      , 5355.      ,
        5964.18603516, 5855.39511719, 5400.      , 6965.      ,
        4818.      , 6356.17490234])
```

```
y_test
```

```
265  7375.000000
230  6750.000000
62   5800.000000
110  4760.000000
27   5954.262207
91   5200.000000
187  6750.000000
85   5250.000000
218  6800.000000
88   5300.000000
11   5929.452637
139  5125.000000
158  5450.000000
12   5730.977539
18   6251.975586
288  8100.000000
59   5800.000000
239  6725.000000
204  7075.000000
119  4760.000000
127  4900.000000
102  5525.000000
175  5225.000000
168  5150.000000
90   5150.000000
282  7900.000000
201  6950.000000
51   6003.881348
238  6700.000000

225  6825.000000
105  4940.000000
```

```
179 5700.000000
205 7025.000000
123 4580.000000
257 6800.000000
231 6750.000000
286 7775.000000
229 6875.000000
278 7825.000000
279 8000.000000
272 7950.000000
150 5250.000000
212 7175.000000
181 5975.000000
120 4640.000000
89 5325.000000
176 5375.000000
206 6900.000000
99 5675.000000
211 7025.000000
164 5400.000000
39 5929.452637
58 5730.977539
177 5350.000000
198 6800.000000
117 4710.000000
4 6251.975586
Name: x(n+1), dtype: float64
```

11) Melakukan evaluasi performance dari machine learning yang dibangun dengan menghitung tingkat akurasi dan root mean squared error (RMSE):

```
# Evaluate Model
model.score(X_test, y_test)

0.9796089020081477

rmse = (mean_squared_error(y_test, y_pr))**0.5
print('RMSE = ',rmse )

RMSE = 140.08260173214939
```

Berdasarkan hasil evaluasi tersebut, machine learning yang dibangun mampu melakukan prediksi harga saham BBNI dengan tingkat akurasi 98% dengan RMSE 140.1