

PENGEMBANGAN BAHAN PENGAJARAN



Program Komputer sebagai Alat Bantu dalam Perkuliahan Kecerdasan Buatan: Machine Learning untuk Mengestimasi Harga Rumah

Dibuat oleh:

**Nur Uddin, PhD
NIDN: 0325067804**

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI DAN DESAIN
UNIVERSITAS PEMBANGUNAN JAYA
TANGERANG SELATAN
Genap 2019/2020**

Machine Learning untuk Estimasi Harga Rumah

Program komputer ini bertujuan untuk membangun sebuah machine learning yang digunakan untuk melakukan estimasi harga rumah. Machine learning tersebut dibuat dengan menggunakan algoritma K-Nearest Neighbors (KNN). Studi kasus dalam program ini adalah harga rumah di daerah Jakarta Selatan. Dataset yang digunakan dalam pembuatan machine learning ini bersumber dari dari:

<https://www.kaggle.com/wisnuanggara/daftar-harga-rumah>

Program ini dibuat menggunakan bahasa Python dan ditulis dalam Google Colab. Tahapan dalam membangun machine learning tersebut adalah sebagai berikut:

1. Import libraries yang diperlukan

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import sklearn.metrics as metrics
```

2. Load data harga rumah di Jakarta Selatan yang telah didownload dan disimpan dalam google drive

```
data_location='/content/drive/My Drive/Data_AI/rumahJakarta/Harga_Rumah_Jaksel_02.xlsx'
raw_data = pd.read_excel(data_location)
raw_data.shape

(1001, 7)
```

3. Melakukan explorasi isi dataset, sebagai contoh melihat pada pada lima baris pertama dan lima baris terakhir:

```
# Menampilkan data pada lima baris pertama
data.head()
```

	HARGA	LuasTanah	LuasBangunan	KamarTidur	KamarMandi	Garasi	KOTA
0	28000000000	1100	700	5	6	ADA	JAKSEL
1	19000000000	824	800	4	4	ADA	JAKSEL
2	4700000000	500	400	4	3	ADA	JAKSEL
3	4900000000	251	300	5	4	ADA	JAKSEL
4	28000000000	1340	575	4	5	ADA	JAKSEL

```
# Menampilkan data pada lima baris terakhir
data.tail()
```

	HARGA	LuasTanah	LuasBangunan	KamarTidur	KamarMandi	Garasi	KOT.
996	16000000000	488	550	6	5	ADA	JAKSE
997	4500000000	209	270	4	4	ADA	JAKSE
998	29000000000	692	400	4	3	TIDAK ADA	JAKSE
999	17000000000	102	140	4	3	TIDAK ADA	JAKSE

4. Membersihkan data yang tidak lengkap (jika ada)

```
data = raw_data.dropna(axis=0)
data.describe()
```

	HARGA	LuasTanah	LuasBangunan	KamarTidur	KamarMandi
count	1.001000e+03	1001.000000	1001.000000	1001.000000	1001.000000
mean	1.747472e+10	530.504496	487.275724	4.457542	3.940060
std	2.079546e+10	531.069773	452.872262	2.004606	1.903261

5. Mendefinisikan fitur dan target

```

features = ['LuasTanah', 'LuasBangunan', 'KamarTidur', 'KamarMandi']
x = data[features]
y = data['HARGA']
x.shape, y.shape

((1001, 4), (1001,))

```

6. Membagi data menjadi data training dan data validasi (test)

```

from sklearn.model_selection import train_test_split
train_x, test_x, train_y, test_y = train_test_split(x,y, random_state = 70)

```

7. Melihat data train_x dan train_y untuk lima baris pertama

```

train_x.head()

```

	LuasTanah	LuasBangunan	KamarTidur	KamarMandi
800	340	419	4	4
604	443	450	5	5
667	225	245	3	3
370	651	350	5	3
646	1017	500	5	3

```

train_y.head()

800    11750000000
604    18500000000
667     5000000000
370    14000000000
646    30000000000
Name: HARGA, dtype: int64

```

8. Membangun machine learning model dengan algoritma K-Nearest Neighbor (KNN)

```

from sklearn.neighbors import KNeighborsRegressor as KNN_Reg
from sklearn.metrics import mean_squared_error as mse

model = KNN_Reg(n_neighbors = 17)

# training the model:
model.fit(train_x,train_y)
acc1 = model.score(test_x, test_y)

# test for prediction
test_predict = model.predict(test_x)
#a = test_predict- test_y
#b = np.dot(a,a)/251
#c = b**0.5
score = mse(test_predict, test_y)
print(' MSE: ', score, '\n','Accuracy: ', acc1)

MSE:  1.4599273281251464e+20
Accuracy:  0.5945004429593623

```

9. Menggunakan elbow method untuk menentukan nilai K terbaik

```

def Elbow(K):
    #initiating an empty list
    test_mse =[]

    #train model for every value of K
    for i in K:
        model = KNN_Reg(n_neighbors=i)
        model.fit(train_x, train_y)
        tmp = model.predict(test_x)
        tmp = mse(tmp, test_y)
        test_mse.append(tmp)

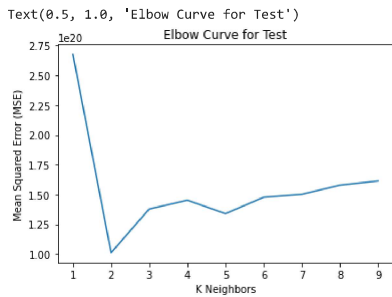
    return test_mse

```

10. Mencari nilai K terbaik dengan menggunakan elbow method

```
K = range(1,10)
test = Elbow(K)

#plotting
plt.plot(K,test)
plt.xlabel('K Neighbors')
plt.ylabel('Mean Squared Error (MSE)')
plt.title('Elbow Curve for Test')
```



Nilai K terbaik adalah nilai K yang menghasilkan MSE minimum, berdasarkan grafik diatas, nilai K=2 merupakan K terbaik.

11. Membangun machine learning yang baru dengan menggunakan nilai K terbaik.

- Machine learning yang baru ini (new model) merupakan revisi dari machine learning sebelumnya (old model).

```
new_model = KNN_Reg(n_neighbors=2)

# Train model
new_model.fit(train_x, train_y)
acc2 = new_model.score(test_x, test_y)

# Prediction test

print(' Accuracy of new model (%): ', acc2*100,'\n', 'Accuracy of old model (%): ', acc1*100, '\n Improvement (%): ', (acc2-acc1)*100 )

Accuracy of new model (%): 71.90114228587117
Accuracy of old model (%): 59.45004429593623
Improvement (%): 12.451097989934933
```

12. Dari data diatas telah dihasilkan dua buah machine learning dengan nama "old model" dan "new model" yang ditujukan untuk mengestimasi harga rumah. Keduanya memiliki akurasi yang berbeda, dimana new model memiliki akurasi 71.90% dan old model memiliki akurasi 59.45%.

Berikut diberikan contoh penggunaan kedua machine learning tersebut dalam melakukan estimasi harga rumah.

a) Estimasi harga berdasarkan data test (test_x dan test_y)

```
# data test_x pada lima baris pertama yang berisikan fitur lima buah rumah yang akan diestimasi harganya.
test_x.head()
```

	LuasTanah	LuasBangunan	KamarTidur	KamarMandi	
324	464	350	4	3	
777	100	186	4	3	
715	474	450	4	3	
817	60	70	3	2	
755	135	250	3	4	

```
# data test_y pada lima baris pertama yang menunjukkan harga sebenarnya untuk kelima rumah dengan fitur diatas.
test_y.head()
```

```
324 1400000000
777 3500000000
715 1440000000
817 1200000000
755 4950000000
Name: HARGA, dtype: int64
```

```
# estimasi harga yang dihasilkan oleh machine learning "old model" untuk kelima rumah tersebut:
fitur_rumah = test_x.head()
estimasi_harga_old_model = model.predict(fitur_rumah)
print(estimasi_harga_old_model)
```

```
[1.47411765e+10 4.00882353e+09 1.61500000e+10 1.08817647e+09
5.21647059e+09]
```

```
# estimasi harga yang dihasilkan oleh machine learning "new model" untuk kelima rumah tersebut:
estimasi_harga_new_model = new_model.predict(fitur_rumah)
print(estimasi_harga_new_model)

[1.175e+10  4.200e+09  1.665e+10  1.215e+09  4.875e+09]
```

▾ b) Estimasi harga rumah yang lain:

Data rumah:

Luas tanah=100 m2, Luas bangunan=80 m2, Kamar tidur=3, dan Kamar mandi=2.

Format input data:

```
data_rumah = ['LuasTanah', 'LuasBangunan', 'KamarTidur', 'KamarMandi']
```

```
data_rumah = np.array([[68,110,3,3]])
prediction_old = model.predict(data_rumah)
prediction_new = new_model.predict(data_rumah)
print(' Hasil Prediksi harga rumah dengan old model: Rp', prediction_old *1e-9, 'Milyar', '\n',
      ' Hasil Prediksi harga rumah dengan new model: Rp', prediction_new *1e-9, 'Milyar', '\n')
```

```
Hasil Prediksi harga rumah dengan old model: Rp [1.67352941] Milyar
Hasil Prediksi harga rumah dengan new model: Rp [1.275] Milyar
```

```
/usr/local/lib/python3.8/dist-packages/sklearn/base.py:450: UserWarning: X does not have valid feature names, but KNeighborsRegressor was fitted with feature names
warnings.warn(
/usr/local/lib/python3.8/dist-packages/sklearn/base.py:450: UserWarning: X does not have valid feature names, but KNeighborsRegressor was fitted with feature names
warnings.warn(
```