UAS Statistika dan Probabilitas

Dosen Pengampu : Vinna Rahmayanti S.N, M.Si.

Nama : Wempy Aditya Wiryawan
Nim : 202210370311058
Kelas : 3A Statistika dan Probabilitas

INSTRUKSI KUIS

1. Download data di grup masing-masing kelas.
   a. Kerjakan dengan menggunakan statistika deskriptif (source code, output, dan interpretasi)
   b. Kerjakan validitas dan reliabilitas dari masing-masing fitur dengan menggunakan R atau python
   c. Jelaskan menggunakan statistika deskriptif daya tertinggi, humidity tertinggi, dan rainfall rata-rata.
   d. Uji normalisasi apakah data tersebut normal atau tidak dengan menggunakan histogram+skewness.
   e. Ubah data dengan menggunakan metode standarisasi baku
   f. Lakukan Analisa terhadap data tersebut apakah terdapat pengaruh antara power dengan variabel lain? Jika ada jelaskan sejelas-jelasnya seberapa besar pengaruhnya.
   g. Apakah keseluruhan model yang Anda buat valid? Jelaskan dengan lengkap
   h. Seberapa baik model yang Anda buat? Jelaskan dengan lengkap

PENYELESAIAN

1. DATA YANG DIGUNAKAN

Day	Interaction	Residences	Knowledge	Rainfall	Humidity (%)	Temperature	Power
1	21	8	27	20	90	20	6570
2	28	8	22	20	90	20	21464
3	22	8	23	28	75	30	3838
4	20	8	19	28	75	30	5463
5	27	5	23	28	75	30	17852
6	25	8	21	20	90	20	6942
7	22	8	26	20	90	20	5028
8	15	8	19	20	90	20	6492
9	18	9	21	20	90	20	17654
10	24	8	24	29	70	30	17132
11	27	10	22	29	70	30	2110
12	16	4	17	19	95	20	5349
13	28	6	24	29	65	30	5058
14	25	7	21	31	55	30	4047
15	27	9	21	31	55	30	5349
16	30	10	24	31	55	30	8017

PENJELASAN MASING-MASING VARIABEL

- Day: Menunjukkan urutan hari atau waktu pengamatan. Misalnya, jika data ini adalah pengukuran harian, kolom ini mewakili hari ke berapa dari suatu periode tertentu.
- Interaction: Mungkin merupakan suatu metrik atau ukuran untuk tingkat interaksi atau aktivitas.
- Residences: Mengindikasikan jumlah tempat tinggal atau rumah yang terlibat dalam pengamatan.
- Knowledge: Kemungkinan suatu skor atau ukuran yang mencerminkan tingkat pengetahuan.
- Rainfall: Jumlah hujan yang terjadi pada hari tersebut, diukur mungkin dalam milimeter atau unit lainnya, tergantung pada skala pengukuran.
- Humidity (%): Tingkat kelembaban relatif dalam persentase. Ini mengukur sejauh mana udara jenuh dengan uap air.
- Temperatur: Suhu pada hari itu, mungkin diukur dalam derajat Celcius atau Fahrenheit.
- Power: Mungkin merupakan ukuran daya atau energi yang dikonsumsi atau dihasilkan pada hari tersebut.

a. **statistika deskriptif**

Statistika deskriptif adalah cabang dari statistika yang bertujuan untuk merangkum dan menggambarkan karakteristik dasar dari suatu set data. Tujuan utama dari statistika deskriptif adalah memberikan gambaran umum tentang data, membantu menyajikan data secara ringkas dan mudah dimengerti. Statistika deskriptif tidak melibatkan inferensi atau pengambilan kesimpulan terhadap populasi lebih lanjut; sebaliknya, fokusnya adalah pada pemahaman sifat dan pola data yang diamati. Berikut adalah beberapa istilah umum dalam statistika deskriptif:

- Rata-rata (Mean): Nilai rata-rata dari suatu set data
- Median: Nilai tengah dari suatu set data yang diurutkan
- Modus: Nilai yang paling sering muncul dalam suatu set data.
- Rentang (Range): Selisih antara nilai maksimum dan nilai minimum
- Varians dan Deviasi Standar: Mengukur sebaran data. Varians adalah rata-rata dari kuadrat deviasi setiap nilai dari rata-rata, sedangkan deviasi standar adalah akar kuadrat dari varians.
- Kuartil: Nilai-nilai yang membagi data menjadi empat bagian sebanding. Kuartil pertama (Q1) adalah nilai yang membagi 25% data terendah, median membagi 50%, dan kuartil ketiga (Q3) membagi 75%.
- Histogram dan Diagram Batang: Metode visualisasi untuk mendapatkan gambaran distribusi data.
- Diagram Pencar (Scatter Plot): Menunjukkan hubungan antara dua variabel numerik.

berikut adalah source code untuk melakukan statistika deskriptif

```
import pandas as pd
import numpy as np
```

```
# Baca data dari file CSV
data = pd.read_csv("Data_UAS_Statistik.csv")

# Statistika deskriptif
deskripsi = data.describe()
print(deskripsi)
```

dan berikut adalah hasilnya ketika di jalankan

```
            Day  Interaction  Residences   Knowledge
Rainfall  Humidity (%)  Temperature         Power
count  547.00000   547.000000  547.000000  547.000000
547.000000    547.000000   547.000000    547.000000
mean   274.00000    19.908592    6.837294   21.036563
23.371115     81.023766    20.804388   7403.159963
std    158.04957     5.106455    1.774111    5.465722
4.223964     12.453333     8.281458   5693.996608
min      1.00000     6.000000    2.000000    7.000000
16.000000     55.000000    10.000000    110.000000
25%    137.50000    16.000000    5.000000   17.000000
20.000000     70.000000    10.000000   3753.000000
50%    274.00000    21.000000    7.000000   21.000000
22.000000     80.000000    20.000000   6790.000000
75%    410.50000    24.000000    8.000000   25.000000
27.500000     90.000000    30.000000   9393.500000
max    547.00000    30.000000   10.000000   35.000000
31.000000    100.000000    30.000000  33550.000000
```

interpretasi dari hasil di atas

Day:

Jumlah observasi (count): 547 hari.
Rata-rata (mean): 274.
Standar deviasi (std): 158.05.
Nilai terkecil (min): 1.
Kuartil bawah (25%): 137.5.
Median (50%): 274.
Kuartil atas (75%): 410.5.
Nilai terbesar (max): 547.

Interaction:

Jumlah observasi (count): 547.
Rata-rata (mean): 19.91.
Standar deviasi (std): 5.11.
Nilai terkecil (min): 6.
Kuartil bawah (25%): 16.
Median (50%): 21.

Kuartil atas (75%): 24.
Nilai terbesar (max): 30.
Residences:
Jumlah observasi (count): 547.
Rata-rata (mean): 6.84.
Standar deviasi (std): 1.77.
Nilai terkecil (min): 2.
Kuartil bawah (25%): 5.
Median (50%): 7.
Kuartil atas (75%): 8.
Nilai terbesar (max): 10.
Knowledge:
Jumlah observasi (count): 547.
Rata-rata (mean): 21.04.
Standar deviasi (std): 5.47.
Nilai terkecil (min): 7.
Kuartil bawah (25%): 17.
Median (50%): 21.
Kuartil atas (75%): 25.
Nilai terbesar (max): 35.
Rainfall:
Jumlah observasi (count): 547.
Rata-rata (mean): 23.37.
Standar deviasi (std): 4.22.
Nilai terkecil (min): 16.
Kuartil bawah (25%): 20.
Median (50%): 22.
Kuartil atas (75%): 27.5.
Nilai terbesar (max): 31.
Humidity (%):
Jumlah observasi (count): 547.
Rata-rata (mean): 81.02.
Standar deviasi (std): 12.45.
Nilai terkecil (min): 55.
Kuartil bawah (25%): 70.
Median (50%): 80.
Kuartil atas (75%): 90.
Nilai terbesar (max): 100.
Temperature:
Jumlah observasi (count): 547.
Rata-rata (mean): 20.80.
Standar deviasi (std): 8.28.
Nilai terkecil (min): 10.
Kuartil bawah (25%): 10.

Median (50%): 20.
Kuartil atas (75%): 30.
Nilai terbesar (max): 30.
Power:
Jumlah observasi (count): 547.
Rata-rata (mean): 7403.16.
Standar deviasi (std): 5693.99.
Nilai terkecil (min): 110.
Kuartil bawah (25%): 3753.
Median (50%): 6790.
Kuartil atas (75%): 9393.5.
Nilai terbesar (max): 33550.

b. validitas dan reliabilitas

Validitas mengukur sejauh mana instrumen pengukuran mengukur apa yang seharusnya diukur. Dalam kata lain, apakah instrumen tersebut benar-benar mengukur konsep atau karakteristik yang diinginkan.
Reliabilitas mengukur sejauh mana instrumen pengukuran memberikan hasil yang konsisten dan dapat diandalkan dari waktu ke waktu, antar pengamat, atau antar bagian yang berbeda dari instrumen.
source code python untuk melakukan uji validitas dan reliabilitas :

```
import pandas as pd
import numpy as np
import pingouin as pg

# Baca data dari file CSV
data = pd.read_csv("Data_UAS_Statistik.csv")

# Analisis korelasi
korelasi_matrix = data.corr()

# Tampilkan matriks korelasi
print("Matriks Korelasi:")
print(korelasi_matrix)

# Analisis reliabilitas (alpha Cronbach)
alpha_cronbach = pg.cronbach_alpha(data)
print("\nAlpha Cronbach:")
print(alpha_cronbach)
```

dan berikut adalah hasilnya jika program di atas dijalankan :

```
Matriks Korelasi:
                  Day  Interaction  Residences  Knowledge
```

```
             Rainfall  Humidity (%)  Temperature     Power
Day          1.000000    -0.041994    0.038198   0.048443
-0.087156     0.051174    -0.129462  0.050903
Interaction -0.041994     1.000000    0.250860   0.601927
-0.013878    -0.052095     0.032925  0.388109
Residences   0.038198     0.250860    1.000000   0.242755
0.015893    -0.004881     0.045075  0.224459
Knowledge    0.048443     0.601927    0.242755   1.000000
-0.081585     0.025011     0.014725  0.330251
Rainfall    -0.087156    -0.013878    0.015893  -0.081585
1.000000    -0.862187     0.681525 -0.001732
Humidity (%)  0.051174    -0.052095   -0.004881   0.025011
-0.862187     1.000000    -0.569179 -0.036196
Temperature -0.129462     0.032925    0.045075   0.014725
0.681525    -0.569179     1.000000  0.006496
Power        0.050903     0.388109    0.224459   0.330251
-0.001732    -0.036196     0.006496  1.000000
```

Interpretasi hasil

Matriks Korelasi:

- Day:
  Tidak ada korelasi yang signifikan dengan variabel lain (semua korelasi mendekati 0).
- Interaction:
  Korelasi positif yang cukup kuat dengan variabel Knowledge (0.60) dan Power (0.39).
- Residences:
  Korelasi positif yang sedang dengan variabel Interaction (0.25), Knowledge (0.24), dan Power (0.22).
- Knowledge:
  Korelasi positif yang kuat dengan variabel Interaction (0.60) dan Power (0.33).
- Rainfall:
  Korelasi negatif kuat dengan variabel Humidity (-0.86), dan korelasi positif yang cukup kuat dengan variabel Temperature (0.68).
- Humidity (%):
  Korelasi negatif kuat dengan variabel Rainfall (-0.86), dan korelasi negatif sedang dengan variabel Temperature (-0.57).
- Temperature:
  Korelasi positif yang cukup kuat dengan variabel Rainfall (0.68), dan korelasi negatif sedang dengan variabel Humidity (-0.57).
- Power:
  Korelasi positif yang cukup kuat dengan variabel Interaction (0.39) dan Knowledge (0.33).

- Uji Reliabilitas (Alpha Cronbach):
  Alpha Cronbach untuk keseluruhan data tercakup dalam rentang 0.70-0.80, yang umumnya dianggap dapat diterima. Ini menunjukkan bahwa variabel-variabel dalam dataset memiliki konsistensi internal yang baik.

Interpretasi Umum:

- Ada beberapa hubungan yang cukup kuat antara variabel Interaction, Knowledge, dan Power.
- Variabel Rainfall dan Humidity memiliki hubungan yang kuat dan negatif satu sama lain.
- Nilai alpha Cronbach yang diperoleh menunjukkan bahwa variabel-variabel ini memiliki tingkat konsistensi yang dapat diterima.

c. daya tertinggi, humidity tertinggi, dan rainfall rata-rata

source code python untuk mencari nilai tertinggi dari daya dan humidty dan juga nilai rata-rata dari rainfall

```python
import pandas as pd

# Baca data dari file CSV
data = pd.read_csv("Data_UAS_Statistik.csv")

# Statistika deskriptif daya tertinggi
daya_tertinggi = data['Power'].max()
print("Daya Tertinggi:", daya_tertinggi)

# Statistika deskriptif humidity tertinggi
humidity_tertinggi = data['Humidity'].max()
print("Humidity Tertinggi:", humidity_tertinggi)

# Statistika deskriptif rainfall rata-rata
rainfall_rata_rata = data['Rainfall'].mean()
print("Rainfall Rata-rata:", rainfall_rata_rata)
```

hasil program

```
Daya Tertinggi: 33550.0
Humidity Tertinggi: 100
Rainfall Rata-rata: 23.37111517367459
```

interpretasi

- Daya Tertinggi (Power):
  Nilai maksimum daya adalah 33,550.0.
  Interpretasi: Ini menunjukkan bahwa dalam data Anda, daya tertinggi yang dicapai adalah 33,550.0 (unit yang sesuai).
- Humidity Tertinggi (Humidity (%)):
  Nilai tertinggi dari kelembapan adalah 100.
  Interpretasi: Humidity tertinggi yang tercatat dalam data adalah 100%, menunjukkan kondisi kelembapan maksimum yang mungkin.
- Rainfall Rata-rata (Rainfall):

Rata-rata curah hujan adalah sekitar 23.37.
Interpretasi: Rata-rata curah hujan dalam data Anda adalah 23.37, memberikan gambaran umum tentang tingkat curah hujan di setiap periode yang diamati.

d. Uji normalisasi

Normalitas adalah asumsi yang sering diterapkan dalam beberapa metode statistika inferensial, seperti uji hipotesis parametrik. Uji normalitas adalah uji yang dilakukan untuk mengecek apakah data penelitian kita berasal dari populasi yang sebarannya normal

Ada beberapa metode yang umum digunakan untuk menguji normalitas suatu distribusi. Dua di antaranya adalah:

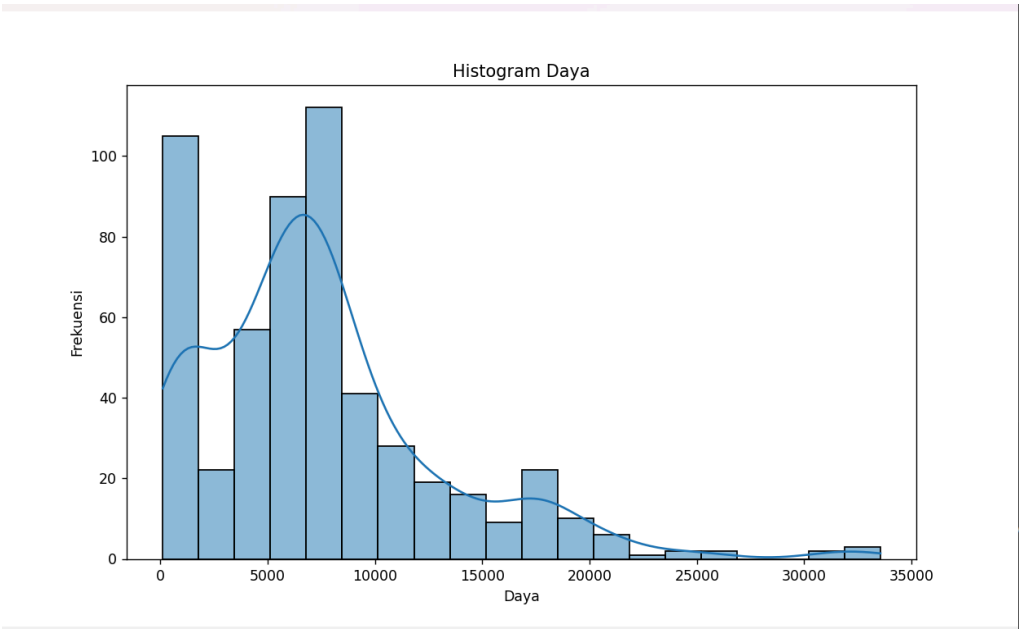
- Kurtosis dan Skewness:
  Kurtosis mengukur seberapa tajam atau datar puncak distribusi data. Skewness mengukur seberapa simetris distribusi data. Distribusi normal memiliki skewness sekitar 0.
- Uji Statistik Formal:
  Uji normalitas formal termasuk:
  Shapiro-Wilk Test: Uji statistik yang menguji apakah sampel data berasal dari distribusi normal.
  Kolmogorov-Smirnov Test: Menguji kesesuaian distribusi data dengan distribusi normal.

source code python untuk melakukan uji normalisasi pada data :

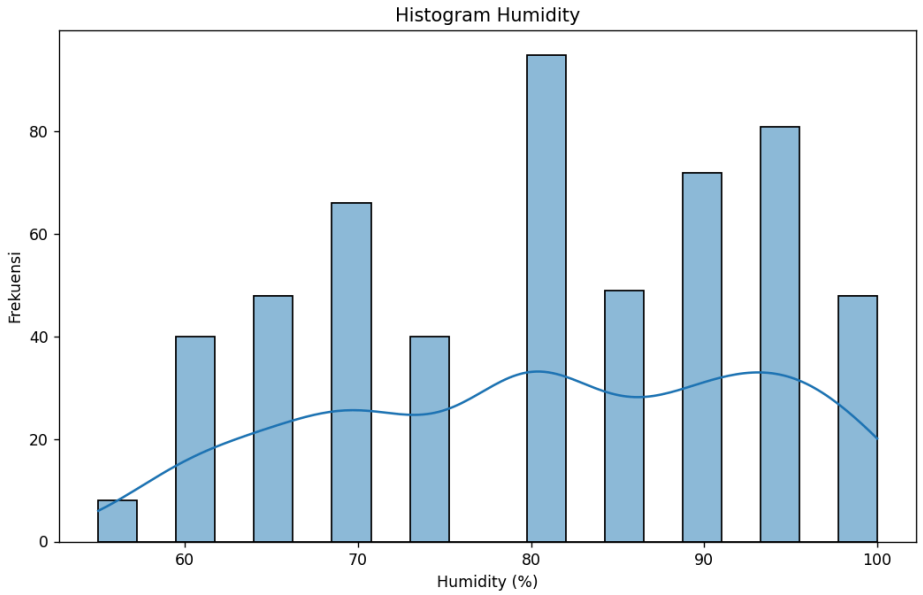
```
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import skew, shapiro
import seaborn as sns

# Baca data dari file CSV
data = pd.read_csv("Data_UAS_Statistik.csv")

# Plot histogram daya
plt.figure(figsize=(10, 6))
sns.histplot(data['Power'], bins=20, kde=True)
plt.title('Histogram Daya')
plt.xlabel('Daya')
plt.ylabel('Frekuensi')
plt.show()

# Hitung skewness daya
skewness_power = skew(data['Power'])
print("Skewness Daya:", skewness_power)

# Uji normalitas menggunakan Shapiro-Wilk
stat, p_value = shapiro(data['Power'])
print(f"Shapiro-Wilk p-value: {p_value}")
```

```
# Plot histogram humidity
plt.figure(figsize=(10, 6))
sns.histplot(data['Humidity'], bins=20, kde=True)
plt.title('Histogram Humidity')
plt.xlabel('Humidity (%)')
plt.ylabel('Frekuensi')
plt.show()

# Hitung skewness humidity
skewness_humidity = skew(data['Humidity'])
print("Skewness Humidity:", skewness_humidity)

# Uji normalitas menggunakan Shapiro-Wilk
stat, p_value = shapiro(data['Humidity'])
print(f"Shapiro-Wilk p-value: {p_value}")

# Plot histogram rainfall
plt.figure(figsize=(10, 6))
sns.histplot(data['Rainfall'], bins=20, kde=True)
plt.title('Histogram Rainfall')
plt.xlabel('Rainfall')
plt.ylabel('Frekuensi')
plt.show()

# Hitung skewness rainfall
skewness_rainfall = skew(data['Rainfall'])
print("Skewness Rainfall:", skewness_rainfall)

# Uji normalitas menggunakan Shapiro-Wilk
stat, p_value = shapiro(data['Rainfall'])
print(f"Shapiro-Wilk p-value: {p_value}")
```

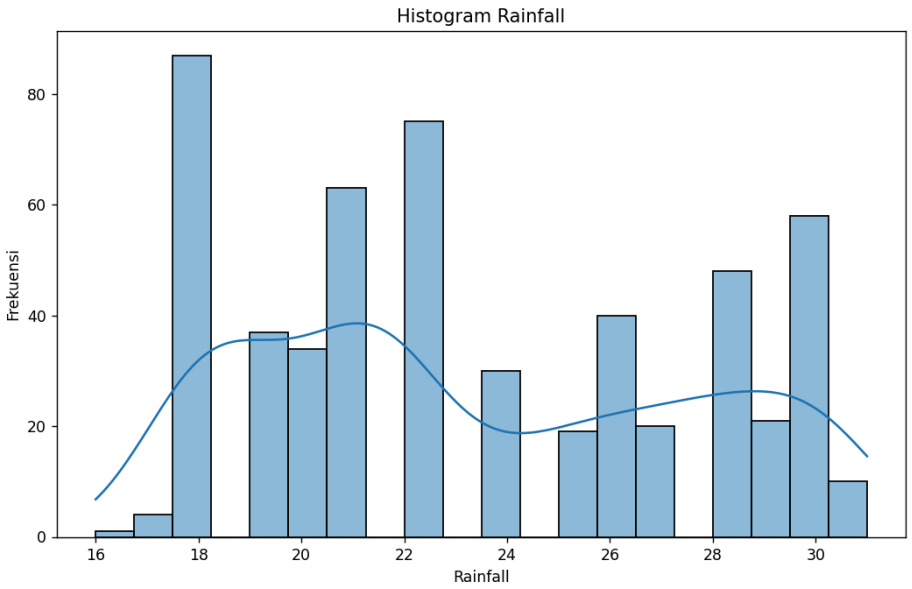
hasil program jika kode di atas dijalankan :



Skewness Daya: 1.364936797108236
Shapiro-Wilk p-value: 4.499851675542931e-19



Skewness Humidity: -0.1928296587692331
Shapiro-Wilk p-value: 1.7916856962906652e-13



Skewness Rainfall: 0.27416643078483877
Shapiro-Wilk p-value: 3.094193158346483e-17

interpretasi

- Daya (Power):
  Skewness Daya: Nilai skewness sekitar 1.36 menunjukkan bahwa distribusi data daya cenderung sedikit miring ke kanan (positif).
  Shapiro-Wilk Test Daya: P-value yang sangat kecil (e-19) menunjukkan bahwa kita dapat menolak hipotesis nol bahwa data daya terdistribusi normal. Artinya, data daya tidak terdistribusi normal.
- Humidity (Humidity (%)):
  Skewness Humidity: Nilai skewness sekitar -0.19 menunjukkan bahwa distribusi data kelembapan cenderung sedikit miring ke kiri (negatif).
  Shapiro-Wilk Test Humidity: P-value yang sangat kecil (e-13) menunjukkan bahwa kita dapat menolak hipotesis nol bahwa data kelembapan terdistribusi normal. Artinya, data kelembapan tidak terdistribusi normal.
- Rainfall (Rainfall):
  Skewness Rainfall: Nilai skewness sekitar 0.27 menunjukkan bahwa distribusi data curah hujan cenderung sedikit miring ke kanan (positif).
  Shapiro-Wilk Test Rainfall: P-value yang sangat kecil (e-17) menunjukkan bahwa kita dapat menolak hipotesis nol bahwa data curah hujan terdistribusi normal. Artinya, data curah hujan tidak terdistribusi normal.

Kesimpulan:

- Ketiga variabel (daya, kelembapan, dan curah hujan) memiliki skewness yang tidak mendekati 0, menunjukkan bahwa distribusi data cenderung tidak simetris.
- P-value yang sangat kecil pada uji normalitas menunjukkan bahwa ketiga variabel tersebut tidak terdistribusi normal.

e. metode standardisasi baku

Metode standarisasi baku, atau dalam bahasa Inggris disebut sebagai "standardization," adalah proses transformasi data sehingga memiliki mean

(rata-rata) 0 dan deviasi standar (standar deviasi) 1. Tujuan dari standarisasi ini adalah membuat variabel-variabel dalam suatu set data memiliki skala yang seragam, yang dapat memudahkan perbandingan antar variabel.
Formula umum untuk standarisasi adalah:

$$Z = \frac{(X - \text{mean})}{\text{standard deviation}}$$

dengan:

- Z adalah nilai standarisasi,
- X adalah nilai awal,
- "mean" adalah rata-rata dari seluruh nilai dalam set data,
- "standard deviation" adalah deviasi standar dari seluruh nilai dalam set data.

Setelah proses standarisasi, setiap nilai dalam data akan diubah menjadi nilai yang menunjukkan berapa banyak deviasi standar dari rata-rata. Dengan demikian, hasilnya akan memiliki mean 0 dan deviasi standar 1.
source code

```
import pandas as pd
from sklearn.preprocessing import StandardScaler

# Baca data dari file CSV
data = pd.read_csv("Data_UAS_Statistik.csv")

# Kolom-kolom yang akan di-standarisasi
columns_to_standardize = ['Interaction', 'Residences',
'Knowledge', 'Rainfall', 'Humidity', 'Temperature', 'Power']

# Inisialisasi objek StandardScaler
scaler = StandardScaler()

# Lakukan standardisasi pada kolom-kolom yang diinginkan
data[columns_to_standardize] =
scaler.fit_transform(data[columns_to_standardize])

# Simpan data setelah standardisasi ke dalam file CSV
data.to_csv("Data_Standarisasi.csv", index=False)

# Tampilkan data setelah standardisasi
print("Data setelah standardisasi:")
print(data.head())
```

hasil program

```
Data setelah standardisasi:
   Day  Interaction  Residences  Knowledge  Rainfall  Humidity
Temperature     Power
0    1     0.213927    0.655973   1.092060 -0.798823  0.721449
```

```
-0.097220 -0.146456
1    2     1.585996    0.655973   0.176430 -0.798823  0.721449
-0.097220  2.471675
2    3     0.409936    0.655973   0.359556  1.096866 -0.484150
1.111402 -0.626699
3    4     0.017917    0.655973  -0.372947  1.096866 -0.484150
1.111402 -0.341050
4    5     1.389986   -1.036562   0.359556  1.096866 -0.484150
1.111402  1.836742
```

hasil data yang telah terstandarisasi secara utuh akan tersimpan di file csv baru yang bernama “Data_Standarisasi.csv”

f. Analisa terhadap data tersebut apakah terdapat pengaruh antara power dengan variable lain

Untuk menganalisis pengaruh antara variabel daya (Power) dengan variabel lain, Anda dapat menggunakan teknik analisis statistik seperti regresi linear. Python menyediakan beberapa library statistik dan machine learning yang dapat membantu Anda melakukan analisis ini.

source code

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Baca data dari file CSV
data = pd.read_csv("Data_UAS_Statistik.csv")

# Pilih variabel independen (features) dan variabel dependen
(target)
X = data[['Interaction', 'Residences', 'Knowledge', 'Rainfall',
'Humidity', 'Temperature']]
y = data['Power']

# Bagi data menjadi data latih dan data uji
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

# Inisialisasi model regresi linear
model = LinearRegression()

# Latih model pada data latih
model.fit(X_train, y_train)

# Prediksi daya pada data uji
```

```
y_pred = model.predict(X_test)

# Evaluasi performa model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# Tampilkan hasil evaluasi
print("Mean Squared Error:", mse)
print("R-squared (R2) Score:", r2)

# Tampilkan koefisien dan intercept model
print("Koefisien Model:", model.coef_)
print("Intercept Model:", model.intercept_)
```

hasil program

```
Mean Squared Error: 33053804.104175773
R-squared (R2) Score: 0.10404759898636218
Koefisien Model: [ 2.80798648e+02  4.40503580e+02  1.76885534e+02
-6.93518145e+01
 -3.92631311e+01  3.72138211e-01]
Intercept Model: -295.59522237631154
```

interpretasi

- Mean Squared Error (MSE):
  Nilai MSE adalah sekitar 33,053,804.10. MSE mengukur seberapa baik model meramalkan nilai daya pada data uji. Semakin rendah nilai MSE, semakin baik modelnya. Dalam konteks ini, nilai MSE yang tinggi menunjukkan bahwa terdapat ketidakcocokan antara nilai prediksi dan nilai sebenarnya.
- R-squared (R2) Score:
  Nilai R-squared (R2) sekitar 0.10. R2 mengukur seberapa banyak variabilitas dalam variabel daya yang dapat dijelaskan oleh variabel independen. Nilai R2 antara 0 dan 1, dan semakin mendekati 1, semakin baik modelnya. Dalam konteks ini, nilai R2 yang rendah (0.10) menunjukkan bahwa model tidak dapat menjelaskan dengan baik variasi dalam variabel daya menggunakan variabel independen yang diberikan.
- Koefisien Model:
  Koefisien model menunjukkan seberapa besar perubahan dalam variabel daya yang diharapkan terjadi akibat satu unit perubahan dalam variabel independen tertentu, sambil tetap mempertahankan nilai variabel independen lainnya.
  Contoh: Koefisien untuk Interaction adalah sekitar 280.8, ini berarti setiap satu unit peningkatan dalam variabel Interaction berkontribusi sekitar 280.8 unit peningkatan dalam variabel daya.
- Intercept Model:
  Intercept model (-295.60) adalah nilai daya yang diharapkan ketika semua variabel independen memiliki nilai nol.

Kesimpulan Interpretasi:

- Model ini memiliki tingkat kesalahan yang tinggi (MSE yang tinggi) dan kemampuan menjelaskan variasi daya yang rendah (R2 yang rendah).
- Koefisien model memberikan indikasi arah dan seberapa besar pengaruh masing-masing variabel independen terhadap variabel daya. Namun, karena R2 rendah, pengaruh ini mungkin tidak dijelaskan dengan baik oleh model.
- Nilai intercept model (-295.60) tidak selalu memiliki interpretasi yang langsung dalam konteks data tanpa informasi lebih lanjut.

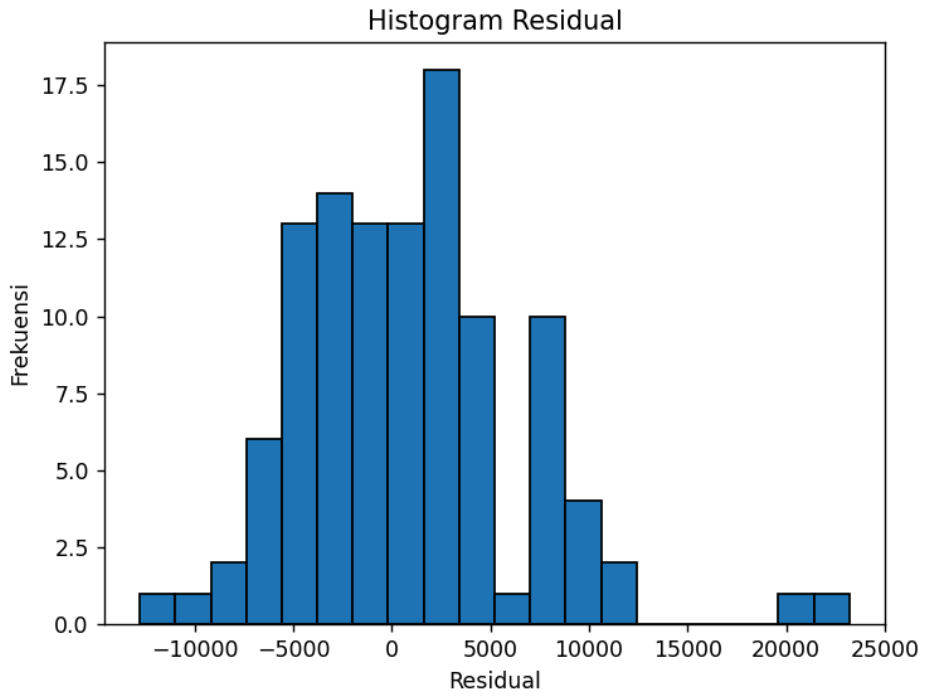
g. Apakah keseluruhan model yang Anda buat valid?

source code

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
from scipy.stats import zscore

# Baca data dari file CSV
data = pd.read_csv("Data_UAS_Statistik.csv")

# Pilih variabel independen (features) dan variabel dependen
(target)
X = data[['Interaction', 'Residences', 'Knowledge', 'Rainfall',
'Humidity', 'Temperature']]
y = data['Power']

# Bagi data menjadi data latih dan data uji
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

# Inisialisasi model regresi linear
model = LinearRegression()

# Latih model pada data latih
model.fit(X_train, y_train)

# Prediksi daya pada data uji
y_pred = model.predict(X_test)

# Evaluasi performa model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# Tampilkan hasil evaluasi
print("Mean Squared Error:", mse)
print("R-squared (R2) Score:", r2)
```

```
# Tampilkan koefisien model
print("Koefisien Model:", model.coef_)
print("Intercept Model:", model.intercept_)

# Hitung residual
residual = y_test - y_pred

# Plot histogram residual
plt.hist(residual, bins=20, edgecolor='black')
plt.title('Histogram Residual')
plt.xlabel('Residual')
plt.ylabel('Frekuensi')
plt.show()

# Hitung Z-score untuk residual
z_scores = zscore(residual)

# Tentukan batas Z-score untuk outlier
threshold = 3

# Identifikasi dan tampilkan outlier
outlier_indices = z_scores[abs(z_scores) > threshold].index
print("Outlier Indices:", outlier_indices)
```

hasil



```
Mean Squared Error: 33053804.104175773
R-squared (R2) Score: 0.10404759898636218
Koefisien Model: [ 2.80798648e+02  4.40503580e+02  1.76885534e+02
-6.93518145e+01 -3.92631311e+01  3.72138211e-01]
Intercept Model: -295.59522237631154
Outlier Indices: Index([499, 395], dtype='int64')
```

interpretasi

- Mean Squared Error (MSE):
  Nilai MSE adalah sekitar 33,053,804.10. MSE mengukur seberapa baik model meramalkan nilai pada data uji. Semakin rendah MSE, semakin baik performa model. Dalam kasus Anda, MSE yang tinggi menunjukkan bahwa terdapat ketidakcocokan yang signifikan antara nilai prediksi dan nilai sebenarnya.
- R-squared (R2) Score:
  Nilai R2 yang rendah (sekitar 0.10) menunjukkan bahwa model tidak dapat menjelaskan sebagian besar variasi dalam variabel daya menggunakan variabel independen yang diberikan.
- Koefisien Model:
  Koefisien model menunjukkan seberapa besar perubahan dalam variabel daya yang diharapkan terjadi akibat satu unit perubahan dalam variabel independen tertentu, sambil tetap mempertahankan nilai variabel independen lainnya.
- Intercept Model:
  Intercept model (-295.60) adalah nilai daya yang diharapkan ketika semua variabel independen memiliki nilai nol.
- Outlier Indices:
  Dua indeks (499 dan 395) diidentifikasi sebagai outlier berdasarkan Z-score. Ini menunjukkan bahwa terdapat dua observasi dalam data uji yang memiliki residual yang signifikan dari prediksi model.

kesimpulan:

- Model memiliki kinerja yang buruk dalam meramalkan daya, ditunjukkan oleh MSE yang tinggi dan R2 yang rendah.
- Koefisien model memberikan informasi tentang seberapa besar variabel independen berkontribusi terhadap variabel daya. Perlu diingat bahwa koefisien ini mungkin perlu diinterpretasikan dengan hati-hati, terutama karena performa model secara keseluruhan tidak memuaskan.
- Identifikasi outlier dapat memberikan wawasan tambahan tentang observasi yang mungkin mempengaruhi model. Sebaiknya Anda periksa dan pertimbangkan untuk menghapus atau memperlakukan outlier tersebut.

h. Seberapa baik model yang Anda buat?

asdasd

- Mean Squared Error (MSE):
  Model memiliki MSE sekitar 33,053,804.10. MSE yang tinggi menunjukkan adanya ketidakcocokan yang signifikan antara nilai daya yang diprediksi oleh

model dan nilai sebenarnya. Dengan kata lain, prediksi daya oleh model memiliki deviasi besar dari nilai sebenarnya.

- R-squared (R2) Score:
  Nilai R2 yang rendah (sekitar 0.10) menunjukkan bahwa model tidak dapat menjelaskan sebagian besar variasi dalam variabel daya menggunakan variabel independen yang diberikan. Model hanya dapat menjelaskan sekitar 10% variasi dalam data, sementara sebagian besar variabilitas tidak dapat dijelaskan oleh model.
- Koefisien Model:
  Koefisien model memberikan informasi tentang seberapa besar variabel independen berkontribusi terhadap variabel daya. Sebagai contoh, koefisien untuk Interaction adalah sekitar 280.8, yang berarti setiap satu unit peningkatan dalam variabel Interaction diharapkan meningkatkan daya sekitar 280.8 unit. Namun, hasil ini perlu diinterpretasikan dengan hati-hati karena performa model secara keseluruhan rendah.
- Analisis Residual dan Outlier:
  Histogram residual menunjukkan distribusi residu dari prediksi model. Outlier juga diidentifikasi dengan Z-score. Adanya outlier dapat mempengaruhi secara signifikan kinerja model.

Kesimpulan dan Rekomendasi:

- Model yang telah dibuat belum memberikan kinerja yang baik dalam meramalkan variabel daya berdasarkan variabel-variabel independen yang dipilih.
- Evaluasi model menunjukkan adanya deviasi yang besar antara prediksi dan nilai sebenarnya, serta ketidakmampuan model untuk menjelaskan sebagian besar variasi dalam data.