Nama : Mutiara Khairunnisa

NIM : 23/517062/PA/22149

TUGAS 1 PRA-UTS

A. ANALISIS DATA

Data yang digunakan dalam mengerjakan tugas 1 geostatistika Pra UTS merupakan data yang berisi 300 angka yang dibuat secara random melalui *website* https://octave-online.net/ dengan menginputkan 5digit terakhir NIM pada kode yang telah dibuat. Jumlah data yang dipunya merupakan bentuk sampel, karena jika kode di*run* ulang akan menghasilkan nilai yang berbeda. Sehingga, data tersebut merupakan 300 angka yang dicuplik secara random dari sebuah populasi angka yang sangat besar. Data yang telah diperoleh pada percobaan pertama memiliki nilai terkecil, yaitu -3.3379 dan yang terbesar yaitu 77.5966.

B. ANALISIS LANGKAH, HASIL, DAN GRAFIK

Dalam mengolah data yang dimiliki, digunakan python yang memanfaatkan *library* pandas untuk membaca data berformat csv. Kemudian, *library* NumPy, Matplotlib, dan sciPy, untuk mengolah dan *plotting* mean, median, mode, range, standar deviasi, beserta IQR. Data diurutkan terlebih dahulu dari angka terkecil ke angka terbesar untuk beberapa pengolahan, seperti median/nilai tengah, kuartil, IQR, dst. Kode yang digunakan untuk melakukan perhitungan sebagai berikut:

```
df = pd.read_csv("geostat.csv")
data = df['value']
data_sorted = sorted(data)

sample_id = np.arange(1, len(data) + 1)
mean = np.mean(data)
std_dev = np.std(data)
median = np.median(data_sorted)
mode = stats.mode(data, keepdims=True).mode[0]
variance = np.var(data)
range = max(data) - min(data)
q1 = np.percentile(data_sorted, 25)
q3 = np.percentile(data_sorted, 75)
iqr = q3 - q1
upper_bound = q3 + 1.5 * iqr
lower_bound = q1 - 1.5 * iqr
```

Gambar 1. Kode untuk Pengolahan Data

Proses pengolahan untuk mean, median, standar deviasi, variansi, Q1, beserta Q3, menggunakan *library* NumPy. Sedangkan untuk modus, menggunakan *library* sciPy

dengan modul stats. Dari pengolahan dengan python tersebut, diperoleh hasil sebagai berikut:

```
Mean                : 38.597772
Standard Deviation  : 12.744605811255312
Median              : 38.03365
Mode                : 57.8837
Variance            : 162.42497728428268
Range               : 80.9345
Q1                  : 30.155625
Q3                  : 46.198750000000004
IQR                 : 16.043125000000003
Upper Bound         : 70.26343750000001
Lower Bound         : 6.0909374999999955
Bin Width           : 4.793049101640663
Number of Bins      : 17
```

Gambar 2. Hasil Pengolahan Data

Dari hasil yang diperoleh, dapat dilanjutkan untuk menghitung lebar bin dengan ***Freedman-Diaconis Rule*** dengan formula $h = \frac{2(Q_3 - Q_1)}{n^{1/3}}$. Dari nilai lebar bin yang diperoleh, dapat digunakan untuk menghitung jumlah bin dengan formula $m = \frac{(data\ max - data\ min)}{h}$. Sehingga, diperoleh lebar bin sebesar 4.793 dan jumlah bin yaitu 17. Kodenya dapat dituliskan sebagai berikut:

```
bin_width = 2*(q3 - q1) / (len(data_sorted) ** (1/3))
num_bin = int(np.ceil(((max(data_sorted) - min(data_sorted)) / bin_width)))
```

Gambar 3. Kode untuk Lebar Bin dan Jumlah Bin

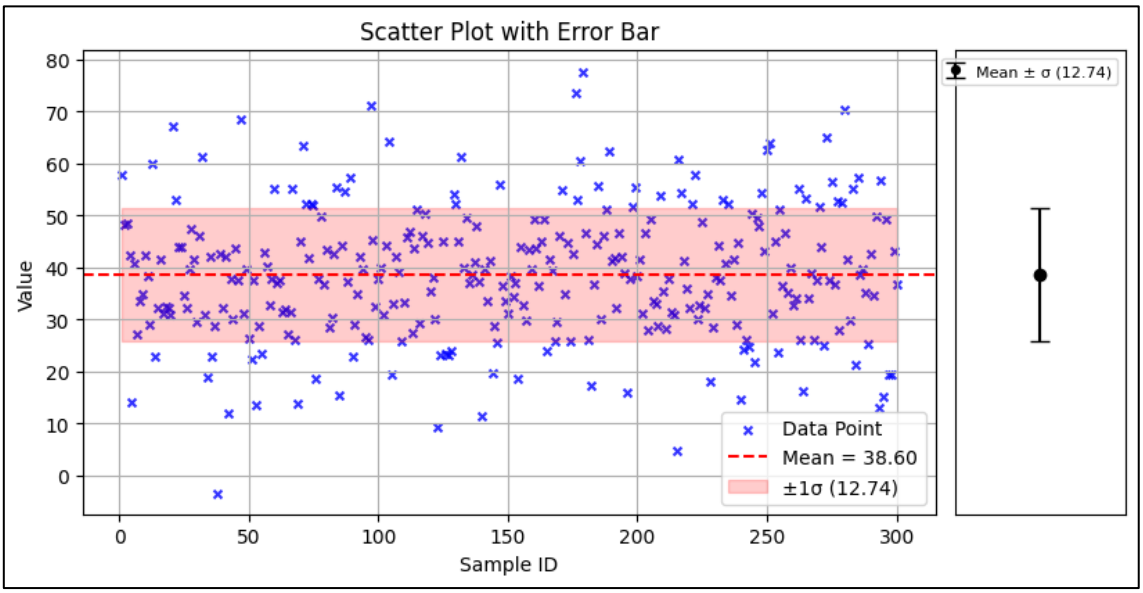
Kemudian, dilakukan *plotting* untuk scatter plot dengan *errorbar,* histogram dengan standar deviasi, scatter plot dengan *box and whisker plot*, serta histogram dengan IQR. Untuk menyusun *scatter plot* beserta *error bar*-nya, digunakan kode sebagai berikut:

```
# Plot scatter plot with error bar
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(8, 4), gridspec_kw={'width_ratios':[3.5, 0.7]})
ax1.scatter(sample_id, data, color='blue', label="Data Point", marker='x', alpha=0.8, s=18)
ax1.axhline(mean, color='red', linestyle='dashed', label=f"Mean = {mean:.2f}")
ax1.fill_between(sample_id, mean - std_dev, mean + std_dev, color='red', alpha=0.2, label=f"±1σ ({std_dev:.2f})")
ax1.set_xlabel("Sample ID")
ax1.set_ylabel("Value")
ax1.set_title("Scatter Plot with Error Bar")
ax1.legend()
ax1.grid(True)
plt.tight_layout(pad=0)

# Error bar
ax2.errorbar(len(data) + 5, mean, yerr=std_dev, fmt='o', color='black', capsize=5, label=f"Mean ± σ ({std_dev:.2f})")
ax2.set_xticks([])  # untuk menghilangkan angka sb x
ax2.set_yticks([])  # untuk menghilangkan angka sb y
ax2.set_ylim(ax1.get_ylim())
ax2.legend(fontsize=8, markerscale=0.7, handlelength=1)
plt.tight_layout(pad=0)
```

Gambar 4. Kode untuk *Scatter Plot* dan *Errorbar*

Dari kode tersebut, dibuat sebuah plot berukuran 8 x 4 dengan 2 bagian. Bagian pertama digunakan untuk plot scatter dari data. Dimana sumbu x mewakili ID sampel/indeks sampel, kemudian sumbu y mewakili nilai/value. Grafik dimodifikasi agar terdapat nilai tengah berupa garis merah putus-putus dan *highlight area* berwarna merah menggambarkan area sebesar 1 standar deviasi dari mean. Kemudian, elemen *errorbar* ditempatkan di kanan dari *scatter plot* dengan panjang *error bar* sebesar standar deviasi. Keempat hal tersebut (scatter, garis median, *highlight area,* dan *error bar)* dibuat dengan menggunakan fungsi dari *library* Matplotlib. Sehingga, kode tersebut menghasilkan gambar *scatter plot* beserta *error bar* sebagai berikut:



Gambar 5. *Scatter Plot* dengan *Error bar*

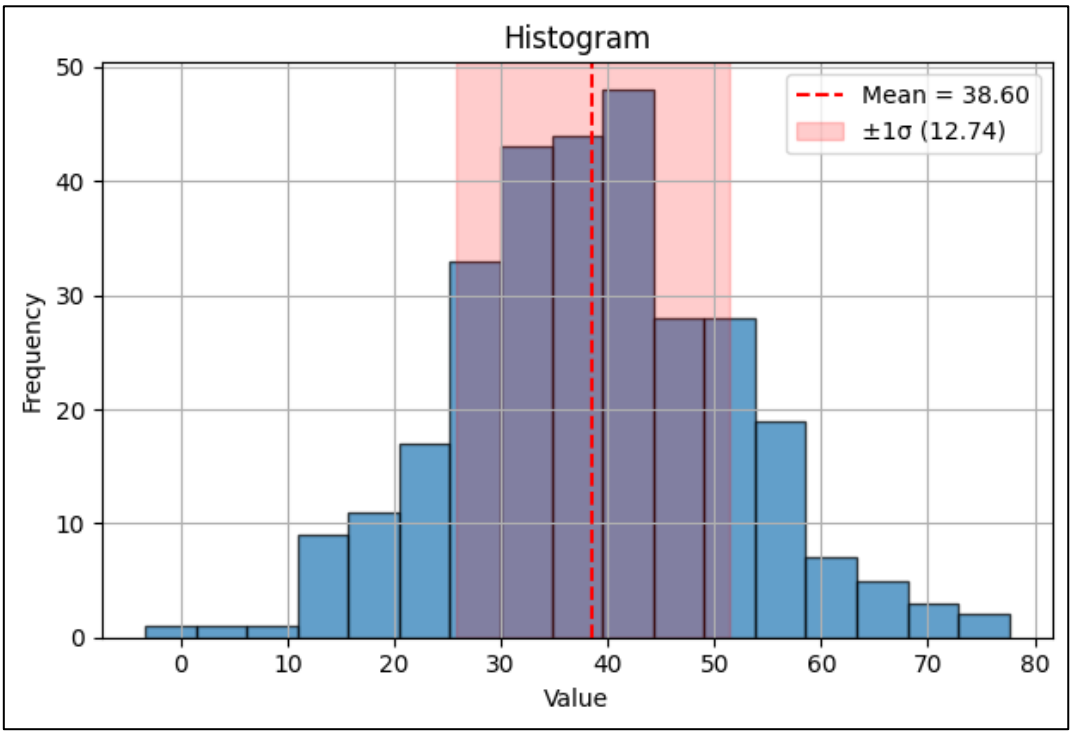
Selanjutnya, untuk penyusunan plot histogram, digunakan kode sebagai berikut:

```
# Plot Histogram with Mean
fig, ax3 = plt.subplots(1, 1, figsize=(6, 4))
ax3.hist(data, bins=num_bin, edgecolor='black', alpha=0.7)
ax3.axvline(mean, color='red', linestyle='dashed', label=f"Mean = {mean:.2f}")
ax3.axvspan(mean - std_dev, mean + std_dev, color='red', alpha=0.2, label=f"±1σ ({std_dev:.2f})")
ax3.set_xlabel("Value")
ax3.set_ylabel("Frequency")
ax3.set_title("Histogram")
ax3.grid(True)
ax3.legend()
plt.tight_layout(pad=0)
```

Gambar 6. Kode untuk Plot Histogram

Dari kode pada gambar 6 berikut, dibuat plot berukuran 6 x 4 yang hanya terdiri dari 1 bagian. Plot mempertimbangkan nilai sesuai dengan jumlah bin yang telah diperoleh dari perhitungan sebelumnya, yaitu 17 bin. Sedangkan frekuensi mewakili frekuensi nilai

sesuai rentang yang ditentukan dari lebar bin, yaitu 4.7 atau dibulatkan menjadi 5. Plot kemudian dimodifikasi dengan ditambahkannya garis *mean* yang berupa garis putus-putus warna merah, dan *highlight area* yang menggambarkan area sebesar 1 standar deviasi dari garis *mean.* Sehingga, dihasilkan sebuah histogram sebagai berikut:



Gambar 7. Histogram dengan Standar Deviasi

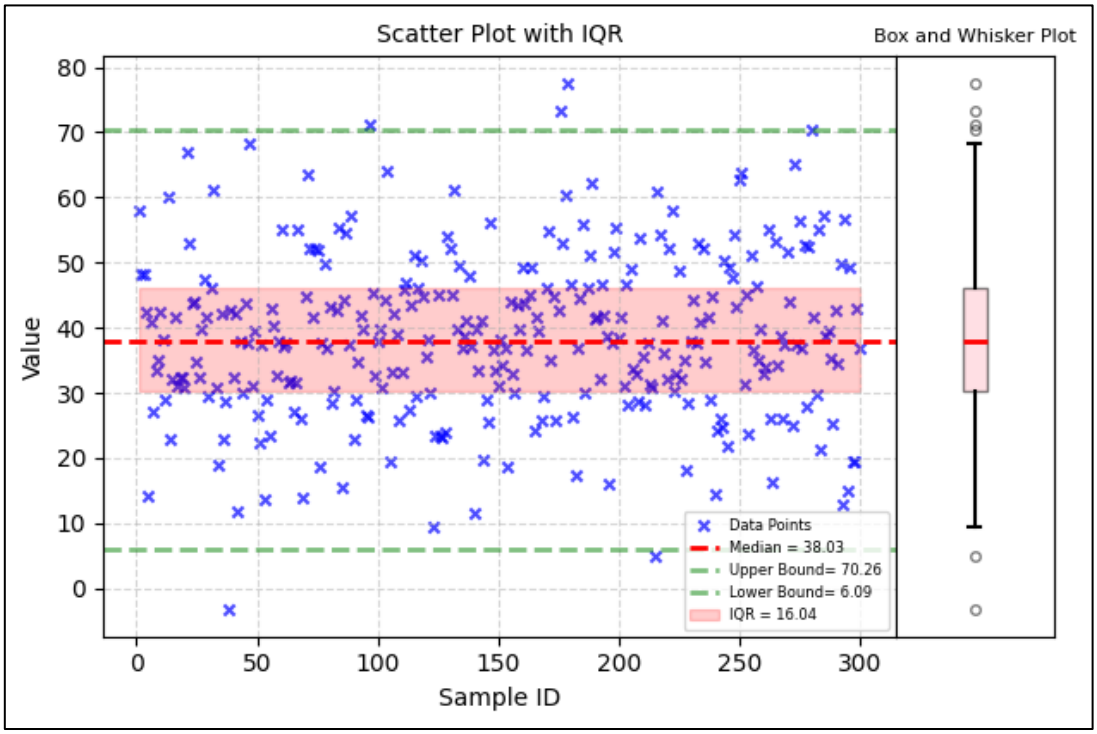
Kemudian, untuk pembuatan *scatter plot* beserta *box and whisker*-nya, digunakan kode berikut:

```
# Plot Scatter with Box and Whishker Plot
fig, (ax4, ax5) = plt.subplots(1, 2, figsize=(6, 4), gridspec_kw={'width_ratios': [3.5, 0.7]})
ax4.scatter(sample_id, data, color='blue', label="Data Points", marker='x', alpha=0.7, s=20)
ax4.axhline(median, color='red', linestyle='dashed', linewidth=2, label=f"Median = {median:.2f}")
ax4.axhline(upper_bound, color='green', linestyle='dashed', linewidth=2, label=f"Upper Bound= {upper_bound:.2f}", alpha = 0.5)
ax4.axhline(lower_bound, color='green', linestyle='dashed', linewidth=2, label=f"Lower Bound= {lower_bound:.2f}", alpha = 0.5)
ax4.fill_between(sample_id, q1, q3, color='red', alpha=0.2, label=f"IQR = {iqr:.2f}")
ax4.set_xlabel("Sample ID")
ax4.set_ylabel("Value")
ax4.set_title("Scatter Plot with IQR", fontsize=10)
ax4.legend(fontsize=6, loc='best')
ax4.grid(True, linestyle="--", alpha=0.5)

# BOx and Whisker Plot
ax5.boxplot(data, vert=True, patch_artist=True,
            boxprops=dict(facecolor='pink', alpha=0.5),
            medianprops=dict(color='red', linewidth=2),
            whiskerprops=dict(color='black', linewidth=1.5),
            capprops=dict(color='black', linewidth=1.5),
            flierprops=dict(marker='o', color='black', alpha=0.5, markersize=4))
ax5.set_xticks([])
ax5.set_yticks([])
ax5.set_title("Box and Whisker Plot", fontsize=8)
ax5.set_ylim(ax4.get_ylim())
plt.tight_layout(pad=0)
```

Gambar 8. Kode untuk *Scatter Plot* dan *Box and Whisker Plot*

Dari kode pada gambar di atas, dibuat plot berukuran 6 x 4 dengan 2 bagian. Bagian pertama untuk *scatter plot,* sedangkan bagian kedua untuk *box and whisker plot.* Plot ini memiliki kesamaan dengan sebelumnya, hanya saja memiliki beberapa perbedaan elemen. Pada plot kali ini, digunakan modifikasi berupa garis tengah putus-putus yang mewakili besar *median* atau nilai tengah dari data yang telah diurutkan. Kemudian, *highlight area* menunjukkan nilai IQR (*Interquartile Range)* yang merupakan sebaran dari 50% data. Untuk *highlight area* diperoleh dari besar Q1 hingga Q3 dari data. Selanjutnya, garis hijau putus-putus mewakili *lower bound* dan *upper bound* yang dihitung dari $Q3 + 1.5IQR$ dan $Q1 - 1.5IQR$. Garis ini digunakan untuk mendeteksi adanya *outlier* atau data yang berada di luar *whisker plot.* Terakhir, pembuatan *box and whisker plot* memanfaatkan fungsi dari Matplotlib, yaitu 'boxplot'. Dengan modifikasi yang dilakukan pada fungsi tersebut, dihasilkan *box plot* dan *whisker plot* yang dapat diamati sekaligus bersama *scatter plot* pada gambar berikut:



Gambar 9. Scatter Plot dengan Box and Whisker Plot

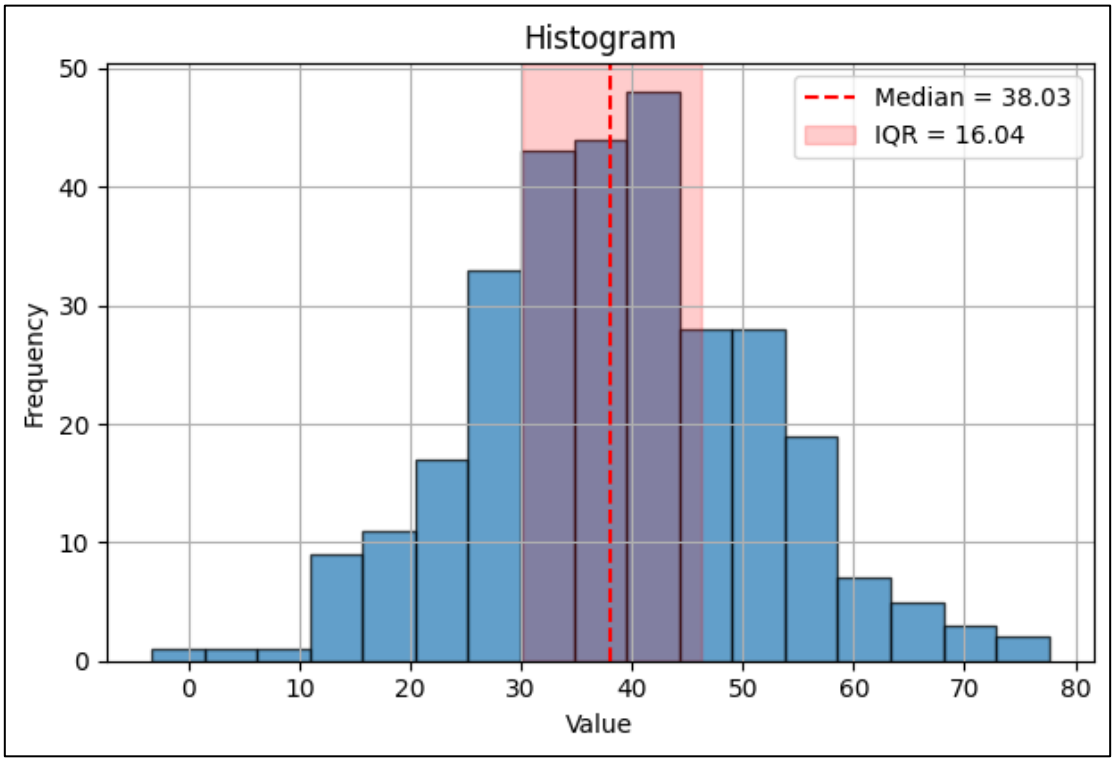
Terakhir, untuk penyusunan plot histogram dengan IQR, digunakan kode berikut:

```
# Plot Histrogram with IQR
fig, ax6 = plt.subplots(1, 1, figsize=(6, 4))
ax6.hist(data, bins=num_bin, edgecolor='black', alpha=0.7)
ax6.axvline(median, color='red', linestyle='dashed', label=f"Median = {median:.2f}")
ax6.axvspan(q1, q3, color='red', alpha=0.2, label=f"IQR = {iqr:.2f}")
ax6.set_xlabel("Value")
ax6.set_ylabel("Frequency")
ax6.set_title("Histogram")
ax6.grid(True)
ax6.legend()

plt.tight_layout(pad=0)
plt.show()
```

Gambar 10. Kode untuk Histogram

Dari kode tersebut, secara keseluruhan memiliki kesamaan pada histogram gambar ketujuh. Namun, hal membedakan adalah garis putus-putus pada histogram kali ini mewakili *median* atau nilai tengah, dan *highlight area* yang mewakili IQR (besar Q1 hingga Q3). Hasil histogram dari kode tersebut dapat diamati sebagai berikut:



Gambar 11. Histogram dengan IQR

C. INTREPRETASI HASIL

Dari proses perngolahan yang telah dilakukan pada gambar pertama, diperoleh hasil yang dapat diamati pada gambar kedua. Dari hasil tersebut, dapat dianalisis terkait dengan pusat datanya, penyebaran datanya, beserta persebaran kuartilnya.

Dari nilai *median, mean,* dan modus, diperoleh bahwasanya nilai *mean* (38.60) ≈ nilai *median* (38.03). Nilai *mean* sendiri menunjukkan titik pusat data. Kemiripan nilai *mean* dan *median,* menunjukkan distribusi data relatif simetris atau distribusi relatif normal. Akan tetapi, nilai modus (57.88), yang cukup jauh dari nilai *mean* dan *median* menandakan adanya *skewness* atau kemiringan. Namun, kemiringan tersebut tidaklah ekstrem, pusat data tetap ditunjukkan berada relatif di tengah. Hal ini dapat didukung dengan histogram pada gambar 7. Histogram yang dihasilkan tidak memiliki *skewness* yang signifikan, serta histogram tersebut berbentuk seperti lonceng, yang menjadi ciri distribusi normal (gaussian)

Kemudian, nilai standar deviasi yang diperoleh, yaitu 12.74, menandakan sebaran data yang cukup besar. Nilai standar deviasi sendiri mengukur jauhnya sebaran data dari *mean.* Apabila nilainya besar, berarti dataset yang dimiliki lebih bervariasi. Dari hasil variansi (162.42) dan *range* (80.93) yang diperoleh, mendukung bahwasanya dataset yang memiliki sebaran cukup besar. Apabila mengamati pada gambar 5, yaitu *scatter plot* dengan *errorbar*, sebagian besar data memang terkonsetrasi di sekitar *mean* serta berada di ± 1 standar deviasi dari *mean,* yang menandakan dataset masih memiliki karakteristik distribusi normal. Adanya nilai yang berada di luar dari *highlight area,* merupakan nilai *outlier.*

Menilik hasil pengolahan data untuk kuartil, diperoleh nilai Q1 (25% bagian bawah) sebesar 30.16, dan nilai Q2 (75% bagian atas) sebesar 46.20. Sehingga diperoleh nilai IQR (*Interquartile Range)* atau rentang antarkuartil sebesar 16.04. Dengan nilai IQR ini, dapat diperoleh *upper bound* (70.26) dan *lower bound* (6.09) yang digunakan sebagai titik batas untuk mendeteksi *outlier.* Apabila mengamati pada gambar 9, yaitu *scatter plot* dengan *box and whisker plot,* grafik tersebut menampilan distribusi data berdasarkan median dan IQR. Dapat diamati bahwasanya sebagian besar data berada di rentang IQR yang mana juga berada di sekitar garis *median.* Tetapi, terdapat titik *outlier* yang digambarkan di luar dari garis *upper bound* dan *lower bound,* atau di luar garis *whisker plot.* Selanjutnya, argumentasi mengenai distribusi relatif simetris turut didukung oleh garis *median* pada *boxplot* yang berada hampir di tengah box. Hal tersebut menunjukkan bahwasanya distribusi relatif simetris.