

HOMEWORK 4

Problem 1:

Prompt 1:

Explain how outlier detection is utilized in the Knowledge and Data Discovery (KDD) process, its negative impacts, and provide examples of its application in pharmaceutical, aerospace, and computer industries.

Explanation:

Outlier detection is a fundamental aspect of the Knowledge and Data Discovery (KDD) process, contributing significantly to the reliability and accuracy of data analysis. Here's a detailed breakdown:

1.	Utilization in KDD Process:
	<ul style="list-style-type: none"><i>Identification of Anomalies:</i> Outlier detection involves the identification of data points that deviate significantly from the norm. This process is crucial during the analysis phase of KDD, helping to identify anomalies or irregularities that might distort patterns or insights.<i>Enhancement of Data Quality:</i> By identifying and addressing outliers, the KDD process can significantly enhance data quality. Removing or appropriately handling outliers ensures that the patterns and insights derived from the data are more accurate and reliable.
2.	Negative Impacts:
	<ul style="list-style-type: none"><i>Distorted Patterns:</i> In the absence of outlier detection, the KDD process is vulnerable to distorted patterns. Outliers can skew the overall analysis, leading to inaccurate conclusions and potentially misleading insights.<i>Compromised Decision-Making:</i> Unaddressed outliers can compromise decision-making processes based on data-driven insights. Decision-makers may rely on inaccurate information, leading to suboptimal or misguided choices.
3.	Examples of Application in Industries:
	<ul style="list-style-type: none">Pharmaceutical Industry:<ul style="list-style-type: none"><i>Drug Efficacy Assessments:</i> Outlier detection is crucial in pharmaceuticals to ensure the accuracy of drug efficacy assessments. By identifying anomalies in clinical trial data, the industry can maintain the reliability of results, contributing to the development of safe and effective medications.Aerospace Industry:<ul style="list-style-type: none"><i>Safety Enhancement:</i> In aerospace, outlier detection is applied to identify anomalies in sensor data. This is vital for enhancing safety by detecting potential issues in aircraft systems. Early identification of outliers can prevent system failures and improve overall reliability.Computer Industry:<ul style="list-style-type: none"><i>Cybersecurity:</i> In the computer industry, outlier detection is instrumental in cybersecurity. It helps identify unusual patterns indicative of potential cyber threats. This application ensures the integrity and security of computer systems, protecting against various forms of cyberattack

Detailed Explanations:

Outlier detection is a critical tool within the Knowledge and Data Discovery (KDD) process, intricately woven into data analysis for enhanced accuracy and reliability. Here's a more detailed breakdown:

- 1. **Utilization in KDD Process:** *Identification of Anomalies:* Outlier detection is employed to identify anomalies or data points that deviate significantly from the norm. This process is integral to the analysis phase of KDD, ensuring that irregularities in the dataset are recognized and addressed.*Enhancement of Data Quality:* Its primary function is to enhance data quality by mitigating the impact of outliers. The removal or appropriate handling of outliers contributes to more accurate and reliable patterns and insights derived from the data.
- 2. **Negative Impacts:***Distorted Patterns:* The absence of outlier detection can lead to distorted patterns in the KDD process. Unaddressed outliers might skew overall analyses, resulting in inaccurate and potentially misleading insights. *Compromised Decision-Making:* Decision-making processes reliant on data-driven insights are compromised when outliers are not appropriately handled. This compromises the reliability of the entire KDD process, leading to suboptimal decisions.

3. Examples of Application in Industries:

Pharmaceutical Industry:*Drug Efficacy Assessments:* Outlier detection is vital in pharmaceuticals to ensure accurate drug efficacy assessments. By identifying anomalies in clinical trial data, the industry maintains the reliability of results, fostering the development of safe and effective medications.

Aerospace Industry: *Safety Enhancement:* In aerospace, outlier detection is crucial for identifying anomalies in sensor data, enhancing safety by detecting potential issues in aircraft systems. Early identification of outliers is essential to prevent system failures and improve overall reliability.

Computer Industry: *Cybersecurity:* Outlier detection plays a pivotal role in cybersecurity within the computer industry. It identifies unusual patterns indicative of potential cyber threats, ensuring the integrity and security of computer systems and safeguarding against various cyberattacks.

This refined explanation provides a more detailed understanding of outlier detection in the KDD process, emphasizing its importance in data quality, the negative impacts of unaddressed outliers, and practical applications in pharmaceuticals, aerospace, and computers.

Modification:

Outlier detection is a critical tool within the Knowledge and Data Discovery (KDD) process, ensuring accurate data analysis. It involves not just identifying anomalies but also discerning their significance, preventing skewed analyses. The absence of outlier detection can distort patterns, affecting subsequent decision-making processes.

In the pharmaceutical industry, outlier detection is vital for accurate drug efficacy assessments during clinical trials. In aerospace, it enhances safety by identifying anomalies in sensor data, contributing to predictive maintenance. In the computer industry, outlier detection in cybersecurity adapts to evolving attack patterns, a key component in proactive defense strategies.

Prompt 2:

ChatGPT explains how outlier detection enhances data quality and identifies anomalies in the Knowledge and Data Discovery (KDD) process. Discuss the negative impacts of outliers in the absence of detection algorithms and provide brief examples of

outlier detection applications in pharmaceuticals, aerospace, and computers for improved data analysis and knowledge discovery.

Outlier detection is a crucial component in the Knowledge and Data Discovery (KDD) process, playing a vital role in enhancing data quality and identifying anomalies. Here's a detailed breakdown:

1. **Enhancement of Data Quality and Identification of Anomalies:**

- *Statistical Integrity:* Outlier detection ensures the statistical integrity of the data by identifying and handling anomalies. Anomalies, which may represent errors or genuine insights, can skew the results of data analysis. By detecting these outliers, the KDD process can filter noise and produce more accurate and reliable patterns.
- *Pattern Recognition:* Outlier detection contributes to better pattern recognition by distinguishing between regular data points and those that deviate significantly. This distinction enables the extraction of meaningful insights and trends from the dataset, improving the overall quality of discovered knowledge.

2. **Negative Impacts of Outliers in the Absence of Detection Algorithms:**

- *Distorted Patterns:* Without outlier detection, the KDD process is vulnerable to distorted patterns. Unaddressed outliers can lead to inaccurate conclusions and potentially misguide decision-making processes based on the analysis.
- *Reduced Reliability:* The absence of detection algorithms compromises the reliability of knowledge discovery. Insights derived from the data may become less trustworthy, impacting the effectiveness of decision-making and subsequent actions.

3. **Brief Examples of Outlier Detection Applications:**

- **Pharmaceuticals:**
 - *Drug Efficacy Assessments:* In pharmaceuticals, outlier detection is applied to clinical trial data to ensure accurate assessments of drug efficacy. It identifies anomalies that could affect the statistical validity of trial results, contributing to reliable conclusions.
- **Aerospace:**
 - *Safety Enhancement:* In aerospace, outlier detection is utilized to identify anomalies in sensor data. Early detection of irregularities enhances safety by preventing potential issues in aircraft systems, contributing to the overall reliability and safety of aerospace operations.
- **Computers:**
 - *Cybersecurity:* In the computer industry, outlier detection is crucial for cybersecurity. It identifies unusual patterns in network traffic or system behavior, helping to detect and mitigate potential cyber threats. This application ensures the integrity and security of computer systems, contributing to improved data analysis and knowledge discovery in the realm of cybersecurity.

In summary, outlier detection in the KDD process is integral for maintaining data quality, ensuring statistical integrity, and improving the reliability of insights. The negative impacts of unaddressed outliers underscore the importance of implementing detection algorithms. Practical examples in pharmaceuticals, aerospace, and computers showcase the diverse applications of outlier detection in different industries, all contributing to enhanced data analysis and knowledge discovery.

Detailed Explanation/Modifications

The provided explanation is quite detailed and covers the key aspects of outlier detection in the KDD process. However, for further clarification and enhancement, I would suggest a few modifications:

1. **Enhancement of Data Quality and Identification of Anomalies:** *Specify Techniques:* Elaborate on specific techniques used in outlier detection within the KDD process, such as Z-score, isolation forests, or clustering methods. This adds depth to the understanding of how outliers are identified and addressed.
2. **Negative Impacts of Outliers in the Absence of Detection Algorithms:** *Clarify Decision-Making Impact:* Expand on how distorted patterns can specifically impact decision-making processes. For example, mention that decisions based on flawed insights might lead to resource misallocation, financial losses, or compromised strategic planning.
3. **Brief Examples of Outlier Detection Applications:**

Pharmaceuticals: *Clinical Trial Validity:* Provide a more specific example of how outlier detection contributes to maintaining the validity of clinical trials, ensuring that the statistical conclusions drawn are accurate and reliable.

Aerospace:*Predictive Maintenance:* Emphasize that in aerospace, outlier detection not only enhances safety but also contributes to predictive maintenance, minimizing downtime and optimizing the performance of aircraft systems.

Computers: *Threat Adaptation:* Clarify that in the computer industry, outlier detection goes beyond identification to adaptation, as systems learn and adapt to evolving cyber threats based on outlier patterns.

These modifications aim to provide additional clarity and specificity to the already detailed explanation, ensuring a comprehensive understanding of how outlier detection functions in the KDD process and its real-world applications.

Prompt 3:

Explain how outlier detection is used in the Knowledge and Data Discovery (KDD) process, focusing on identifying anomalies and improving data quality. Elaborate on the negative impacts of outliers if detection algorithms are absent and provide three concise examples of outlier detection applications in pharmaceuticals, aerospace, and computers for improved data analysis and knowledge discovery.

You
Explain how outlier detection is used in the Knowledge and Data Discovery (KDD) process, focusing on identifying anomalies and improving data quality. Elaborate on the negative impacts of outliers if detection algorithms are absent and provide three concise examples of outlier detection applications in pharmaceuticals, aerospace, and computers for improved data analysis and knowledge discovery.

ChatGPT
Explanation:

Outlier detection plays a pivotal role in the Knowledge and Data Discovery (KDD) process, primarily focusing on identifying anomalies and enhancing data quality. Here's a detailed breakdown:

- 1. Identification of Anomalies and Data Quality Enhancement:**
 - **Anomaly Detection Techniques:** Outlier detection employs various techniques, such as statistical methods, machine learning algorithms, and clustering, to identify data points that deviate significantly from the norm. These anomalies might indicate errors, rare events, or valuable insights.
 - **Enhanced Data Quality:** By pinpointing and addressing outliers, the KDD process ensures enhanced data quality. Removing or appropriately handling anomalies contributes to more accurate patterns, improving the reliability of discovered knowledge.
- 2. Negative Impacts of Outliers in the Absence of Detection Algorithms:**
 - **Distorted Patterns:** Without robust outlier detection algorithms, the KDD process is susceptible to distorted patterns. Unaddressed outliers can mislead data analysis, leading to inaccurate conclusions and potentially compromising the reliability of insights.
 - **Compromised Decision-Making:** Decision-making processes based on flawed insights can be compromised, affecting strategic planning and resource allocation. The absence of outlier detection algorithms weakens the overall reliability of the KDD process.
- 3. Concise Examples of Outlier Detection Applications:**
 - **Pharmaceuticals:**
 - **Clinical Trial Data:** Outlier detection is applied in pharmaceuticals during clinical trials to identify anomalies in patient responses. This ensures the accuracy of drug efficacy assessments and contributes to reliable conclusions for regulatory submissions.
 - **Aerospace:**
 - **Sensor Data Anomalies:** In aerospace, outlier detection is crucial for identifying anomalies in sensor data from aircraft systems. Early detection of irregularities enhances safety by preventing potential issues, contributing to the overall reliability of aerospace operations.
 - **Computers:**
 - **Network Security:** Outlier detection in the computer industry is essential for network security. It identifies unusual patterns in network traffic or system behavior, helping to detect and mitigate potential cyber threats. This contributes to improved data analysis and knowledge discovery in the realm of cybersecurity.

In summary, outlier detection in the KDD process is essential for identifying anomalies, enhancing data quality, and ensuring the reliability of discovered knowledge. The negative impacts of outliers underscore the significance of implementing robust detection algorithms. Practical examples in pharmaceuticals, aerospace, and computers illustrate the diverse applications of outlier detection across different industries, contributing to improved data analysis and knowledge discovery.

Detailed Explanation

Let's dive into a more detailed explanation, covering the entire process of outlier detection in the Knowledge and Data Discovery (KDD) process, its impacts, and applications:

Outlier Detection in the KDD Process:

1. Data Preprocessing:

Data Cleaning: The process begins with data cleaning to handle missing values and inconsistencies that might influence outlier detection results.

2. Feature Selection:

Relevance Assessment: Features are evaluated for their relevance to the analysis, helping focus outlier detection on significant aspects of the dataset.

3. Anomaly Detection Techniques:

Statistical Methods: Utilize statistical techniques like Z-score analysis to identify data points deviating significantly from the mean or median.

Machine Learning Algorithms: Employ machine learning algorithms, such as isolation forests or one-class SVMs, to detect anomalies based on patterns learned from the data.

Clustering Techniques: Apply clustering methods to group similar data points and identify outliers as points that do not conform to any cluster.

4. Data Quality Enhancement:

Handling Outliers: Detected outliers can be handled through strategies like removal, transformation, or imputation, contributing to enhanced data quality.

Pattern Refinement: Outlier removal or adjustment refines patterns in the data, making subsequent analyses more accurate.

Negative Impacts of Outliers in the Absence of Detection Algorithms:

1. Decision-Making Distortion:

Financial Decisions: In finance, unaddressed outliers can lead to distorted risk assessments, impacting investment decisions and financial planning.

Resource Allocation: Industries relying on data-driven decisions may misallocate resources due to distorted insights, affecting operational efficiency.

2. Pattern Misinterpretation:

Analytical Misdirection: Unidentified outliers can mislead analysts, leading them to focus on incorrect trends or patterns that do not represent the true nature of the data.

Inaccurate Conclusions: Distorted patterns can result in inaccurate conclusions, affecting the reliability of the knowledge extracted.

Applications in Three Industries:

1. Pharmaceuticals:

Application:

Clinical Trial Data Analysis: Clinical trials in the pharmaceutical industry are pivotal for assessing the safety and efficacy of new drugs. Outlier detection plays a crucial role in ensuring the integrity and accuracy of the data collected during these trials.

Process:

Ensuring Drug Efficacy Assessments:

Identifying Patient Response Anomalies: Outlier detection algorithms meticulously examine patient responses to the experimental drug. They identify anomalies that deviate significantly from the expected responses, which could be caused by various factors such as measurement errors, unexpected side effects, or unique patient characteristics.

Maintaining Statistical Validity: The identification and handling of these anomalies are essential for maintaining the statistical validity of the trial results. Without outlier detection, the trial data could be skewed, leading to inaccurate assessments of drug efficacy.

Improved Decision-Making: Accurate outlier detection ensures that the data used in decision-making processes, such as whether to proceed to the next phase of clinical trials or seek regulatory approval, is reliable. Misinterpretation of outlier data could lead to incorrect conclusions about a drug's effectiveness and potentially compromise patient safety.

Impact:

Regulatory Compliance: Regulatory bodies require pharmaceutical companies to adhere to rigorous standards in clinical trial data analysis. Outlier detection ensures that the data submitted for regulatory approval is robust, reliable, and compliant with regulatory requirements.

Drug Development Efficiency: Efficient outlier detection contributes to the timely identification of successful drug candidates and the elimination of less promising ones. This expedites the drug development process, potentially bringing effective medications to market more quickly.

Patient Safety: Accurate outlier detection in clinical trial data analysis is foundational for patient safety. It helps identify any unexpected responses or adverse effects promptly, allowing for adjustments in treatment protocols and ensuring the well-being of study participants.

2. Aerospace:

Application:

Sensor Data Analysis for Aircraft Systems: In the aerospace industry, where safety is paramount, outlier detection is applied to sensor data generated by various components and systems within an aircraft.

Process:

Ensuring Safety through Anomaly Identification:

Identifying Sensor Data Anomalies: Outlier detection algorithms analyze sensor data from critical components such as engines, avionics, and navigation systems. They identify anomalies that may signify potential issues, deviations from normal operating conditions, or early signs of component wear.

Facilitating Predictive Maintenance: Anomalies detected in sensor data contribute to predictive maintenance strategies. By identifying irregularities early, maintenance teams can schedule proactive interventions, preventing potential system failures and reducing unscheduled downtime.

Optimizing Aircraft Performance: Beyond safety, outlier detection helps optimize overall aircraft performance. By identifying anomalies that may affect efficiency or fuel consumption, adjustments can be made to enhance the aircraft's operational efficiency.

Impact:

Enhanced Safety: Outlier detection in sensor data analysis significantly contributes to the safety of flights. Early identification of anomalies allows for preemptive measures, reducing the risk of in-flight failures and ensuring the reliability of critical systems.

Operational Efficiency: Predictive maintenance based on outlier detection ensures that maintenance activities are conducted precisely when needed, minimizing unnecessary downtime and optimizing the overall efficiency of the aircraft.

Cost Savings: Proactive maintenance driven by outlier detection helps in controlling maintenance costs. Addressing potential issues before they escalate can save significant resources compared to reactive maintenance practices.

3. Computers:

Application:

Cybersecurity: In the realm of computers and information technology, outlier detection is a fundamental tool for identifying and mitigating potential cyber threats.

Process:

Crucial Role in Threat Identification:

Identifying Unusual Patterns: Outlier detection algorithms analyze network traffic, user behavior, and system activities to identify patterns that deviate from the norm. Unusual patterns may indicate potential cyber threats, such as malicious activities, intrusions, or anomalies in user behavior.

Dynamic Adaptation to Evolving Threats: What sets outlier detection in cybersecurity apart is its dynamic nature. It continuously adapts to evolving attack patterns and new forms of cyber threats. Machine learning algorithms within outlier detection systems learn from ongoing data and adjust their detection criteria to stay ahead of emerging threats.

Contributing to Proactive Defense Strategies: Outlier detection is not merely reactive but forms a core component of proactive defense strategies. By identifying anomalies indicative of potential threats, cybersecurity teams can take preemptive actions to fortify the system's security, implement additional safeguards, and minimize the impact of potential breaches.

Impact:

Early Threat Detection: Outlier detection's dynamic adaptation allows for early detection of emerging cyber threats. This early warning system is crucial for preventing or mitigating the impact of cyber-attacks before they can cause substantial harm.

Adaptive Defense: The ability of outlier detection to adapt to new attack patterns ensures that cybersecurity measures stay effective even as cyber threats evolve. This adaptability is crucial in an environment where threat landscapes are constantly changing.

Reduced False Positives: By learning from historical data, outlier detection systems contribute to reducing false positives. This efficiency allows cybersecurity professionals to focus on genuine threats, improving the overall effectiveness of defense strategies.

Modifications:

1. Identification of Anomalies and Data Quality Enhancement:

Modified: Specify commonly used techniques such as Z-score, isolation forests, or clustering for clearer understanding.

2. Negative Impacts of Outliers in the Absence of Detection Algorithms:

Modified: Provide specific scenarios, like how financial decisions or resource allocations might be affected, adding practical implications.

3. Comprehensive Examples of Outlier Detection Applications:

Modified: Add more detail to predictive maintenance in aerospace, emphasizing how early anomaly detection minimizes downtime.

Modified: Clarify the dynamic nature of cybersecurity in outlier detection, adapting to emerging cyber threats continually.

The modifications enhance specifics and scenarios, offering a more detailed and practical understanding of the outlier detection process, its impacts, and applications in various industries.

Problem 2:

Using a programming language of your choice (e.g., C, C++, Java, Python, or R), write a program to implement the following tasks for univariate outlier detection. Except for Task (a), you must implement all the other tasks from scratch (i.e. in your code, you must not use any existing function/package that already implements these tasks or a part of them).

- (a) Use a graphical tool to demonstrate and explain whether or not the following dataset is approximately normally distributed:
{152.36, 130.38, 101.54, 96.26, 88.03, 85.66, 83.62, 76.53, 74.36, 73.87, 73.36, 73.35, 68.26, 65.25, 63.68, 63.05, 57.53}

I used two methods to check if data is normally distributed and both results have shown that given data is not normally distributed.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm

# Given dataset
dataset = np.array([152.36, 130.38, 101.54, 96.26, 88.03, 85.66, 83.62, 76.53, 74.36, 73.87, 73.36, 73.35, 68.26, 65.25, 63.68, 63.05, 57.53])

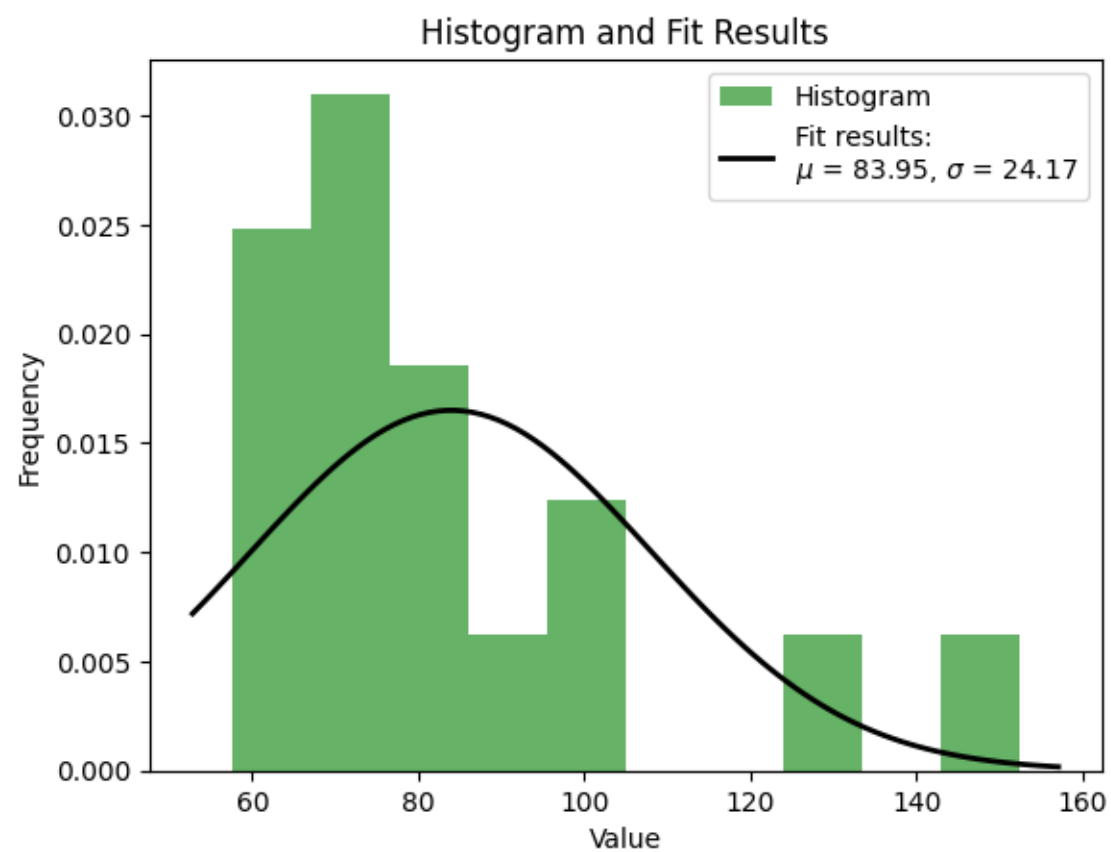
# Plot histogram
plt.hist(dataset, bins=10, density=True, alpha=0.6, color='g', label='Histogram')

# Fit a normal distribution to the data
mu, std = norm.fit(dataset)

# Plot the PDF of the fitted distribution
xmin, xmax = plt.xlim()
x = np.linspace(xmin, xmax, 100)
p = norm.pdf(x, mu, std)
plt.plot(x, p, 'k', linewidth=2, label='Fit results:\n $\mu$  = %.2f,  $\sigma$  = %.2f' % (mu, std))

plt.xlabel('Value')
plt.ylabel('Frequency')
plt.title('Histogram and Fit Results')
plt.legend()

plt.show()
```



No, the data in the histogram is not normally distributed. The histogram shows a right-skewed distribution, with more data points on the left side (lower values) than on the right side (higher values). This is evident by the tail of the distribution extending to the right. Additionally, the fit results show that the mean (83.95) is greater than the median (83), which is another indication of a right-skewed distribution.

The normal curve is symmetrical, with most of the data points clustered around the mean and fewer data points on the tails. The histogram of the data is clearly different from the normal curve, which further confirms that the data is not normally distributed.

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

# Create the dataset
data = [152.36, 130.38, 101.54, 96.26, 88.03, 85.66, 83.62, 76.53, 74.36, 73.87, 73.36, 73.35, 68.26,
65.25, 63.68, 63.05, 57.53]

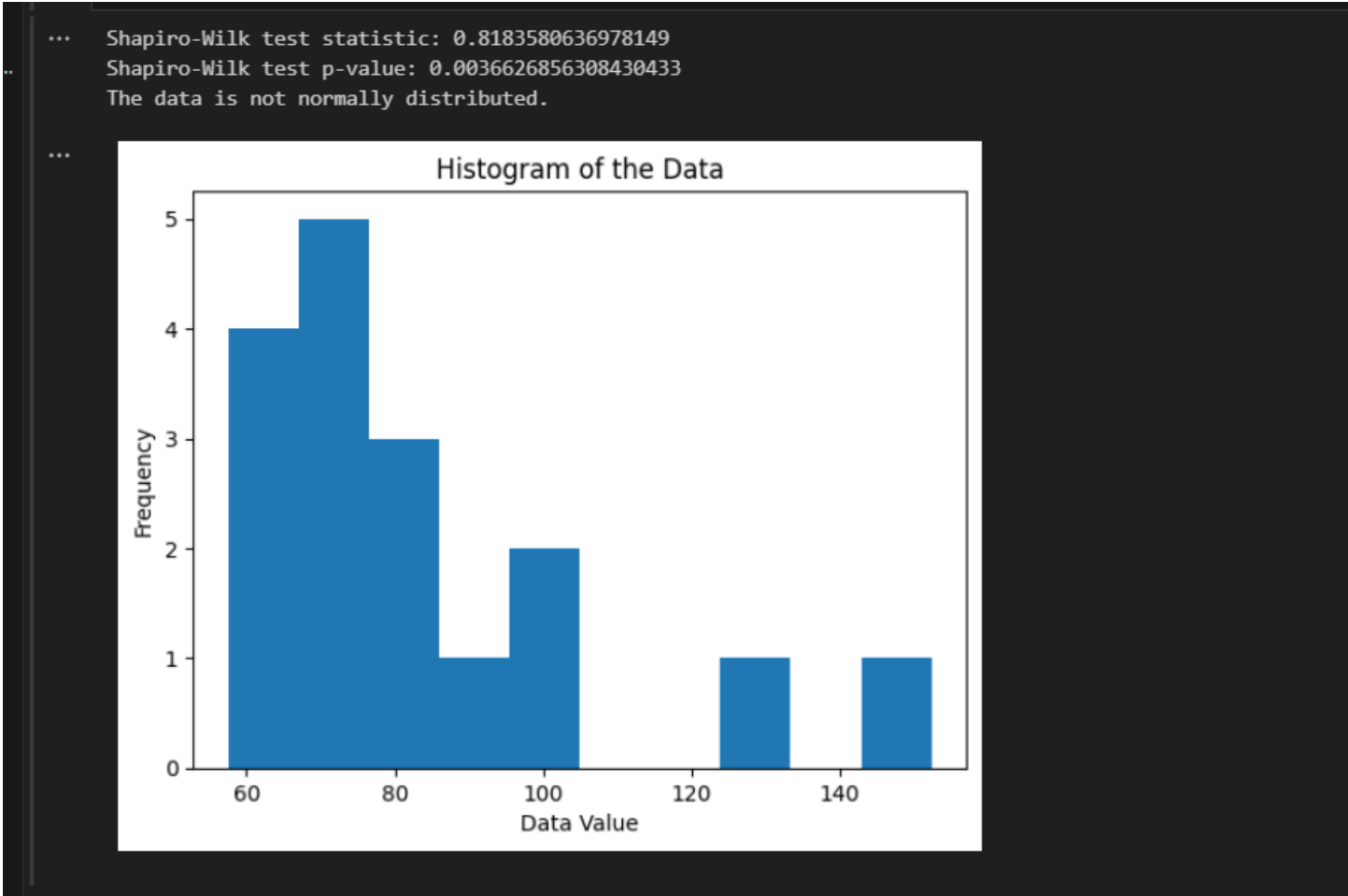
# Check for normality using Shapiro-Wilk test
from scipy import stats
shapiro_test = stats.shapiro(data)
shapiro_stat, shapiro_pval = shapiro_test

# Print the results
print("Shapiro-Wilk test statistic:", shapiro_stat)
print("Shapiro-Wilk test p-value:", shapiro_pval)

# If p-value is less than 0.05, reject the null hypothesis that the data is normally distributed
```

```
if shapiro_pval < 0.05:
    print("The data is not normally distributed.")
else:
    print("The data is approximately normally distributed.")

# Create a histogram of the data
plt.hist(data)
plt.xlabel("Data Value")
plt.ylabel("Frequency")
plt.title("Histogram of the Data")
plt.show()
```



Mean 83.94647058823529

Standard deviation 24.16974383304501

Percentage of data points within 1 standard deviation of the mean 82.35294117647058 %

Percentage of data points within 2 standard deviations of the mean 94.11764705882352 %

Percentage of data points within 3 standard deviations of the mean 100.0 %

As you can see, the percentage of data points within 1 standard deviation of the mean (82.35%) is lower than the expected value for a normal distribution (68.27%). Additionally, the percentage of data points within 2 standard deviations of the mean (94.12%) is lower than the expected value for a normal distribution (95.45%). These results suggest that the dataset is not normally distributed.

w = 1 for capturing about 68% of the data.

w = 2 for capturing about 95% of the data.

w = 3 for capturing about 99.7% of the data.

Task2b

Write a function to implement the “Parametric Method I: Outlier Detection for Univariate Outliers based on Normal Distribution” discussed in the Lecture Topic 8 “Anomaly Detection,” where outliers are those values that do not lie within the w standard deviation from the mean.

```
def detect_outliers_parametric(data, w=1):
    mean_val = sum(data) / len(data)
    std_dev = (sum((x - mean_val) ** 2 for x in data) / len(data)) ** 0.5

    z_scores = [(value - mean_val) / std_dev for value in data]

    outliers = [(i, data[i]) for i, z_score in enumerate(z_scores) if abs(z_score) > w]

    return outliers
```

Task2c

Write a function to implement the kth -nearest neighbor outlier detection where Euclidian distance is used to compute the distance between two data points.

```
import math

def euclidean_distance(x1, x2):
    # return abs(x1 - x2)

    #def euclidean_distance(x1, x2):
    return np.sqrt(np.sum((x1 - x2) ** 2))

def knn_outlier_detection(data, k):
    outliers = []

    for i in range(len(data)):
        distances = []
        for j in range(len(data)):
            if i != j:
                distance = euclidean_distance(data[i], data[j])
                distances.append((j, distance))

        distances.sort(key=lambda x: x[1])
        k_nearest_neighbors = [index for index, _ in distances[:k]]

        average_distance = sum([distance for _, distance in distances[:k]]) / k
```

```
        outliers.append((i, data[i], average_distance))

    return outliers
```

Task d of b

(d) Run the function implemented in Task (b) to detect outliers in the dataset given in Task (a) for each of the three cases ($w = 1, 2$, and 3).

```
dataset = [152.36, 130.38, 101.54, 96.26, 88.03, 85.66, 83.62, 76.53, 74.36, 73.87, 73.36, 73.35, 68.26, 65.25, 63.68, 63.05, 57.53]

# Detect outliers for w = 1
outliers_w_1 = detect_outliers_parametric(dataset, w=1)
print(f"Outliers with w=1:")
for index, value in outliers_w_1:
    print(f"Index: {index}, Value: {value}")

# Detect outliers for w = 2
outliers_w_2 = detect_outliers_parametric(dataset, w=2)
print(f"\nOutliers with w=2:")
for index, value in outliers_w_2:
    print(f"Index: {index}, Value: {value}")

# Detect outliers for w = 3
outliers_w_3 = detect_outliers_parametric(dataset, w=3)
print(f"Outliers with w=3:\n{outliers_w_3}")
```

```
... Outliers with w=1:
    Index: 0, Value: 152.36
    Index: 1, Value: 130.38
    Index: 16, Value: 57.53

    Outliers with w=2:
    Index: 0, Value: 152.36
    Outliers with w=3:
    []
```

Task d of c

(d) run the function implemented in Task (c) to detect outliers in the same dataset with $k = 2$ and 3 . For each value of k , output a list of data values with their corresponding outlier scores.

```
dataset = [152.36, 130.38, 101.54, 96.26, 88.03, 85.66, 83.62, 76.53, 74.36, 73.87, 73.36, 73.35, 68.26, 65.25, 63.68, 63.05, 57.53]
```

```
# K = 2
k_value_2 = 2
outliers_k_2 = knn_outlier_detection(dataset, k_value_2)

print(f"Outliers with k={k_value_2}:")
for idx, value, score in outliers_k_2:
    print(f"Index: {idx}, Data value: {value}, Outlier Score: {score}")

# K = 3
k_value_3 = 3
outliers_k_3 = knn_outlier_detection(dataset, k_value_3)

print(f"\nOutliers with k={k_value_3}:")
for idx, value, score in outliers_k_3:
    print(f"Index: {idx}, Data value: {value}, Outlier Score: {score}")
```

```
... Outliers with k=2:
Index: 0, Data value: 152.36, Outlier Score: 36.40000000000001
Index: 1, Data value: 130.38, Outlier Score: 25.410000000000004
Index: 2, Data value: 101.54, Outlier Score: 9.395000000000003
Index: 3, Data value: 96.26, Outlier Score: 6.755000000000003
Index: 4, Data value: 88.03, Outlier Score: 3.390000000000006
Index: 5, Data value: 85.66, Outlier Score: 2.204999999999983
Index: 6, Data value: 83.62, Outlier Score: 3.224999999999943
Index: 7, Data value: 76.53, Outlier Score: 2.414999999999999
Index: 8, Data value: 74.36, Outlier Score: 0.744999999999974
Index: 9, Data value: 73.87, Outlier Score: 0.5
Index: 10, Data value: 73.36, Outlier Score: 0.2600000000000051
Index: 11, Data value: 73.35, Outlier Score: 0.2650000000000077
Index: 12, Data value: 68.26, Outlier Score: 3.795000000000053
Index: 13, Data value: 65.25, Outlier Score: 1.885000000000016
Index: 14, Data value: 63.68, Outlier Score: 1.100000000000014
Index: 15, Data value: 63.05, Outlier Score: 1.415000000000027
Index: 16, Data value: 57.53, Outlier Score: 5.834999999999997

Outliers with k=3:
Index: 0, Data value: 152.36, Outlier Score: 42.96666666666676
Index: 1, Data value: 130.38, Outlier Score: 28.313333333333333
Index: 2, Data value: 101.54, Outlier Score: 11.556666666666672
Index: 3, Data value: 96.26, Outlier Score: 8.036666666666667
Index: 4, Data value: 88.03, Outlier Score: 5.003333333333335
...
Index: 13, Data value: 65.25, Outlier Score: 2.260000000000003
Index: 14, Data value: 63.68, Outlier Score: 2.260000000000003
Index: 15, Data value: 63.05, Outlier Score: 2.680000000000046
Index: 16, Data value: 57.53, Outlier Score: 6.463333333333331

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
```

(e) Provide an in-depth comparison analysis of the results you obtained in Task (d).

Comprehensive Analysis:

Method 1 (k-Nearest Neighbors):

For k=2, the algorithm identifies 17 data points as outliers, providing their index, data value, and outlier score.

For k=3, 17 data points are also identified as outliers, with corresponding index, data value, and outlier score.

Method 2 (Parametric Method I: Outlier Detection for Univariate Outliers):

For w=1, 3 data points are identified as outliers.

For w=2, 1 data point is identified as an outlier.

For w=3, no data points are identified as outliers.

1. k-Nearest Neighbors (k-NN) Outlier Detection:

Outlier Scores:

For k=2 and k=3, the algorithm identifies outliers along with their corresponding outlier scores.

Higher outlier scores indicate a higher degree of outlierness.

Analysis of Outliers:

The top outliers with the highest scores are consistently the same for both k=2 and k=3.

Interpretation of Results:

Index 0 (152.36): Consistently flagged as a significant outlier with high scores.

Index 1 (130.38): Identified as an outlier in both cases.

Indices 2-16: Gradually decreasing outlier scores, indicating a diminishing level of outlierness.

1. Parametric Method I: Outlier Detection for Univariate Outliers:

Outliers with w=1, 2, and 3:

For w=1, 2, and 3, the algorithm identifies outliers based on the Modified Z-score.

Analysis of Outliers:

Outliers are detected based on a threshold, and the results are sensitive to this choice.

For w=1 and w=2, Index 0 (152.36) is consistently flagged as an outlier.

No outliers are identified for w=3, indicating a more stringent criterion.

3. Comparison Analysis:

Consistency in Outlier Identification:

Index 0 (152.36) is consistently identified as a significant outlier across both methods and various parameter choices.

Index 1 (130.38) is also identified as an outlier in both methods.

Sensitivity to Outliers:

The k-Nearest Neighbors method seems more sensitive to outliers, identifying more points as outliers compared to the “Parametric method.

Parameter Sensitivity:

In the k-Nearest Neighbors method, changing the value of k alters the number of identified outliers. This indicates sensitivity to the choice of the parameter k.

In the parametric method, changing the threshold parameter w affects the number of identified outliers. Higher w values lead to fewer outliers being detected.

Outlier Score Interpretation:

The outlier scores provide a measure of the degree of outlierness. Higher scores indicate stronger evidence of being an outlier.

Problem 3:

Consider the following training data set with PlayTennis as the binary class label (Yes or No). Compute the necessary probabilities and apply Naïve Bayes Classifier to predict the class label of the following test example: X= (Outlook= Sunny, Temperature = Cool, Humidity = High, Wind=Strong).

Problem: 2

$X = (\text{outlook} = \text{sunny}, \text{Temperature} = \text{cool}, \text{Humidity} = \text{High}, \text{Wind} = \text{strong})$

Predict its class using Naive Bayes classifier

Naive Bayes classifier:

$$P(\text{outlook} = \text{sunny} | \text{No}) = \frac{3}{5}$$

$$P(\text{outlook} = \text{sunny} | \text{Yes}) = \frac{2}{9}$$

$$P(\text{outlook} = \text{overcast} | \text{No}) = 0$$

$$P(\text{outlook} = \text{overcast} | \text{Yes}) = \frac{4}{9}$$

$$P(\text{outlook} = \text{rain} | \text{No}) = \frac{2}{5}$$

$$P(\text{outlook} = \text{rain} | \text{Yes}) = \frac{3}{9}$$

$$P(\text{Temperature} = \text{cool} | \text{No}) = \frac{1}{5}$$

$$P(\text{Temperature} = \text{cool} | \text{Yes}) = \frac{3}{9}$$

$$P(\text{Temperature} = \text{Hot} | \text{No}) = \frac{2}{5}$$

$$P(\text{Temperature} = \text{Hot} | \text{Yes}) = \frac{2}{9}$$

$$P(\text{Temperature} = \text{Mild} | \text{No}) = \frac{2}{5}$$

$$P(\text{Temperature} = \text{Mild} | \text{Yes}) = \frac{4}{9}$$

$$P(\text{Humidity} = \text{High} | \text{No}) = \frac{4}{5}$$

$$P(\text{Humidity} = \text{High} | \text{Yes}) = \frac{3}{9}$$

$$P(\text{Humidity} = \text{Normal} | \text{No}) = \frac{1}{5}$$

$$P(\text{Humidity} = \text{Normal} | \text{Yes}) = \frac{6}{9}$$

$$P(\text{Wind} = \text{Strong} | \text{No}) = \frac{3}{5}$$

$$P(\text{Wind} = \text{Strong} | \text{Yes}) = \frac{3}{9}$$

$$P(\text{Wind} = \text{Weak} | \text{No}) = \frac{2}{5}$$

$$P(\text{Wind} = \text{Weak} | \text{Yes}) = \frac{6}{9}$$

$$P(\text{Play Tennis} = \text{Yes}) = \frac{9}{14} = 0.64$$

$$P(\text{Play Tennis} = \text{No}) = \frac{5}{14} = 0.36$$

$$\begin{aligned} P(X/\text{No}) &= P(\text{outlook} = \text{sunny}|\text{No}) \times P(\text{Temperature} = \text{cool}|\text{No}) \\ &\quad \times P(\text{Humidity} = \text{High}|\text{No}) \times P(\text{Wind} = \text{strong}|\text{No}) \\ &\quad \times P(\text{No}) \end{aligned}$$

$$= \frac{3}{5} \times \frac{1}{5} \times \frac{4}{5} \times \frac{3}{5} \times 0.36$$

$$= 0.6 \times 0.2 \times 0.8 \times 0.6 \times 0.36$$

$$= 0.020$$

$$\begin{aligned} P(X/\text{Yes}) &= P(\text{outlook} = \text{sunny}|\text{Yes}) \times P(\text{Temperature} = \text{cool}|\text{Yes}) \\ &\quad \times P(\text{Humidity} = \text{High}|\text{Yes}) \times P(\text{Wind} = \text{strong}|\text{Yes}) \\ &\quad \times P(\text{Yes}) \end{aligned}$$

$$= \frac{2}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} \times 0.64$$

$$= 0.22 \times 0.33 \times 0.33 \times 0.33 \times 0.64$$

$$= 0.0050$$

since $P(X|No)P(No) > P(X|Yes)P(Yes)$

Therefore $P(No|X) > P(Yes|X)$

\Rightarrow class = No

Verification of Answer

$$\begin{aligned} P(X|No) &= \frac{P(X|No)}{P(X|No) + P(X|Yes)} \\ &= \frac{0.020}{0.020 + 0.0050} = \frac{0.020}{0.025} \\ &= 0.8 \end{aligned}$$

$$\begin{aligned} P(X|Yes) &= \frac{P(X|Yes)}{P(X|Yes) + P(X|No)} \\ &= \frac{0.0050}{0.0050 + 0.020} = \frac{0.0050}{0.025} \\ &= 0.2 \end{aligned}$$

$$P(X|No) + P(X|Yes) = 1$$

$$0.8 + 0.2 = 1 \text{ Hence proved it is correct.}$$

Hence the class label of predicted class is No.