EXPERIMENT NO: 6

AIM: Outlier detection using distance based/density based method.

THEORY:

Outlier detection, also known as anomaly detection, is the process of identifying observations that deviate significantly from the majority of the data points. Outliers can represent data instances that are rare, erroneous, or carry valuable information, making their detection crucial in various applications such as fraud detection, network security, medical diagnosis, and more.

There are several methods for outlier detection, and distance-based methods, including Knearest neighbors (KNN), are widely used. Here's an overview of how distance-based outlier detection works using KNN:

- K-nearest neighbors (KNN): KNN is a non-parametric method used for classification and regression tasks. However, it can also be utilized for outlier detection. In KNN, each data point is classified based on the majority class of its nearest neighbors. For outlier detection, instead of classifying data points, we focus on their distances to their nearest neighbors.

- Distance Calculation: For each data point, we calculate its distance to its k-nearest neighbors. Typically, Euclidean distance is used, but other distance metrics like Manhattan distance can also be employed based on the nature of the data and problem.

- Mean Distance Calculation: Once distances to the k-nearest neighbors are computed for each data point, we calculate the mean distance for each data point by averaging its distances to its neighbors.

- Identifying Outliers: Outliers can be identified based on the mean distances calculated. Data points with unusually large mean distances compared to the majority of the data are considered outliers. A cutoff threshold can be determined to distinguish outliers from normal data points. This threshold can be set empirically or based on statistical measures like percentiles of the mean distances.

- Visualization: Outliers can be visualized on a scatter plot where normal data points are plotted along with outliers. This visualization helps in understanding the distribution of data and the presence of outliers.

It's important to note that while distance-based methods like KNN are effective for many datasets, they may not perform well in high-dimensional spaces due to the curse of dimensionality. In such cases, other outlier detection methods such as density-based methods like DBSCAN (Density-Based Spatial Clustering of Applications with Noise) may be more suitable. These methods focus on identifying regions of higher density in the data space, which can be indicative of normal data points, and flagging data points in sparser regions as outliers.

CODE:

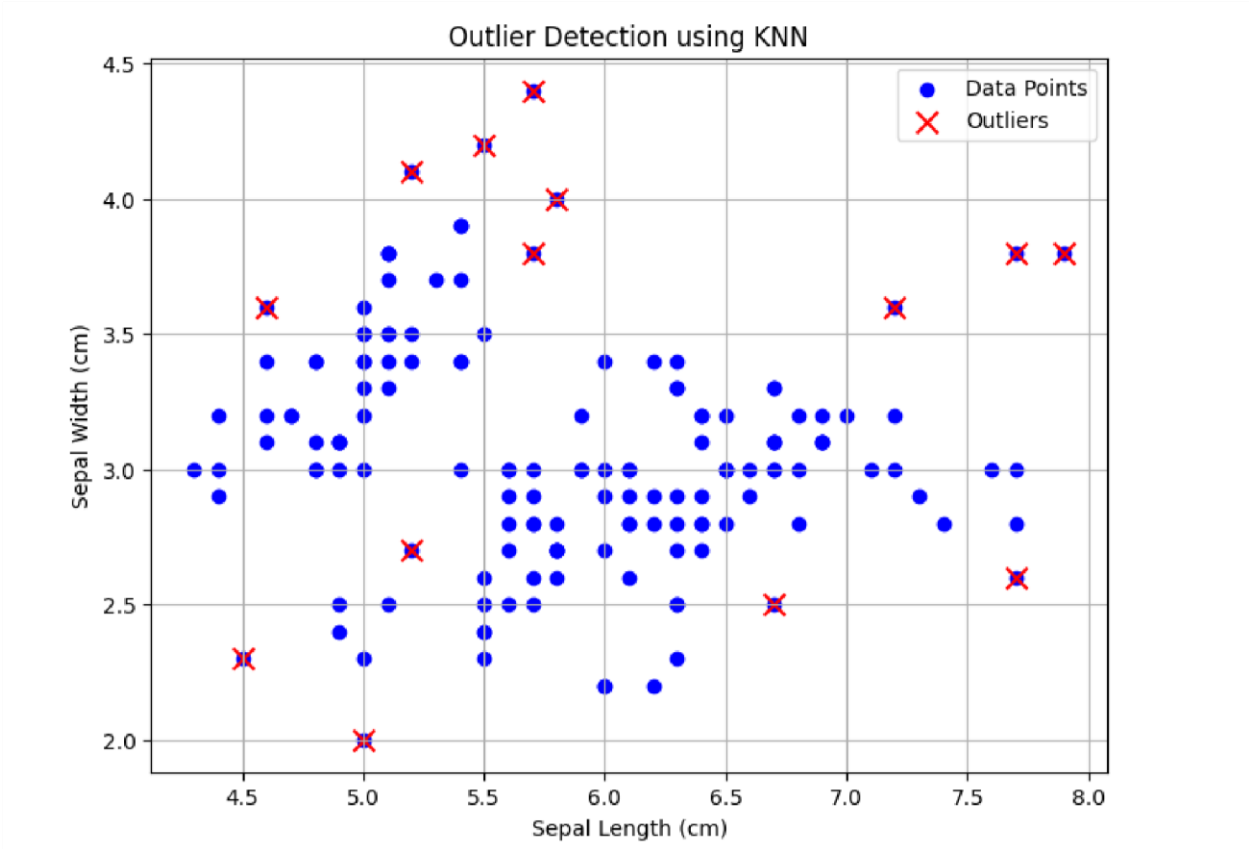
```
# Importing libraries import pandas as pd import

numpy as np import matplotlib.pyplot as plt from

sklearn.neighbors import NearestNeighbors



# Loading data data = pd.read_csv("Iris.csv") # Assuming

iris.csv is in the same directory as your script



# Selecting input features

X = data[["SepalLengthCm", "SepalWidthCm"]] # Adjust column names if

necessary

# Instantiating KNN model k = 3 #

Number of neighbors nbrs =

NearestNeighbors(n_neighbors=k)

nbrs.fit(X)
```

```
# Calculating distances and indices of k-nearest neighbors distances,

indices = nbrs.kneighbors(X)


# Calculating mean distance for each point mean_distances

= distances.mean(axis=1)


# Determining cutoff threshold for identifying outliers cutoff

= 0.15 # Adjust this threshold as needed


# Identifying outliers based on mean distances outlier_index

= np.where(mean_distances > cutoff)[0] outliers =

X.iloc[outlier_index]

# Plotting data points and outliers plt.figure(figsize=(8, 6))

plt.scatter(X["SepalLengthCm"], X["SepalWidthCm"],

label="Data Points", color='blue')

plt.scatter(outliers["SepalLengthCm"],

outliers["SepalWidthCm"], label="Outliers", color='red',

marker='x', s=100) plt.xlabel("Sepal Length (cm)")
```

plt.ylabel("Sepal Width (cm)") plt.title("Outlier Detection using

KNN") plt.legend() plt.grid(True) plt.show()

OUTPUT:



CONCLUSION:

In this experiment, we utilized K-nearest neighbors (KNN) for outlier detection by measuring the mean distances of data points to their nearest neighbors. By setting a threshold, outliers were identified and visualized on a scatter plot, demonstrating the importance and application of distance-based methods in identifying unusual patterns within datasets.