Q1

The anomaly detection/feature selection is done by simply flagging each metric as a zero or a 1 if its value is inside or outside normal range (0 for within normal range; 1 for outside of normal range). I also calculate a “flag ratio” that expresses how far outside of normal the value is.

Q2 One way to evaluate anomaly detection models is to use precision and recall, which are commonly used for binary classification problems. Precision measures the fraction of detected anomalies that are actually true anomalies, while recall measures the fraction of true anomalies that are detected by the model. The most commonly used algorithms for this purpose are supervised Neural Networks, Support Vector Machine learning, K-Nearest Neighbors Classifier.Generally, in order to evaluate the quality of an anomaly detection technique, the confusion matrix and its derived metrics such as precision and recall are used. These metrics, however, do not take this temporal dimension into consideration. At inference time, the anomaly score can be calculated by calculating the difference between the predicted and actual value for that time point. Values that fall outside the prediction's confidence interval can be directly classified as anomalous.

Q3 DBSCAN is a density-based clustering algorithm that works on the assumption that clusters are dense regions in space separated by regions of lower density. It groups 'densely grouped' data points into a single cluster. The principle of DBSCAN is to find the neighborhoods of data points exceeds certain density threshold. The density threshold is defined by two parameters: the radius of the neighborhood (eps) and the minimum number of neighbors/data points (minPts) within the radius of the neighborhood.DBSCAN is a clustering algorithm that defines clusters as continuous regions of high density and works well if all the clusters are dense enough and well separated by low-density regions.

Q4 DBSCAN requires only two parameters: epsilon and minPoints. Epsilon is the radius of the circle to be created around each data point to check the density and minPoints is the minimum number of data points required inside that circle for that data point to be classified as a Core point.In other words, it is the distance that DBSCAN uses to determine if two points are similar and belong together. A larger epsilon will produce broader clusters (encompassing more data points) and a smaller epsilon will build smaller clusters. Another use case of DBSCAN for outlier detection is in anomaly detection for sensor data. For instance, if a manufacturing plant wants to monitor its equipment for any anomalies, it could use DBSCAN to cluster the sensor data and identify any data points that are not part of these clusters as potential outliers.

Q5 Core points are those that have a minimum number of points (MinPts) within a specified radius ε. Border points have fewer than MinPts within ε but are in the neighborhood of a core point. Noise points are all other points that are neither core nor border points.Another use case of DBSCAN for outlier detection is in anomaly detection for sensor data. For instance, if a manufacturing plant wants to monitor its equipment for

any anomalies, it could use DBSCAN to cluster the sensor data and identify any data points that are not part of these clusters as potential outliers.

Q6 DBSCAN requires only two parameters: epsilon and minPoints. Epsilon is the radius of the circle to be created around each data point to check the density and minPoints is the minimum number of data points required inside that circle for that data point to be classified as a Core point. DBSCAN can be used for anomaly detection in addition to clustering. One of the distinguishing features of DBSCAN is that, as I mentioned above, it separates data points into three types: core points, border points, and noise points.5. Steps in the DBSCAN algorithm:-

1.Classify the points. 2.Discard noise. 3.Assign cluster to a core point. 4.Color all the density connected points of a core point. 5.Color boundary points according to the nearest core point.

Q7 Make a large circle containing a smaller circle in 2d.A simple toy dataset to visualize clustering and classification algorithms.Some of the parameters:- n_samples : int, optional (default=100).The total number of points generated. shuffle: bool, optional (default=True) : Whether to shuffle the samples. noise : double or None (default=None) Standard deviation of Gaussian noise added to the data. factor : double < 1 (default=.8) Scale factor between inner and outer circle. N_samples:- X : array of shape [n_samples, 2]

The generated samples.

y : array of shape [n_samples]

The integer labels (0 or 1) for class membership of each sample

Q8 There are two general types of outlier detection: global and local. Global outliers fall outside the normal range for an entire dataset, whereas local outliers may fall within the normal range for the entire dataset, but outside the normal range for the surrounding data points. A point that falls outside the dataset inner fences is classified as a minor outlier, while one that falls outside the outer fences is classified as a major outlier. To find the inner fences for your data set, first multiply the interquartile range by 1.5; then add the result to Q3 and subtract it from Q1.

Q9 By comparing the density of the data point and density of all the data points in the neighborhood, whether the density of the data point is lower than the density of the neighborhood can be determined. This scenario indicates the presence of an outlier. The Local Outlier Factor (LOF) algorithm is an unsupervised anomaly detection method which computes the local density deviation of a given data point with respect to its neighbors. It considers as outliers the samples that have a substantially lower density than their neighbors.A. Most popular outlier detection methods are Z-Score, IQR (Interquartile Range), Mahalanobis Distance, DBSCAN (Density-Based Spatial Clustering of Applications with Noise, Local Outlier Factor (LOF), and One-Class SVM (Support Vector Machine).

Q10 Isolation Forest is a technique for identifying outliers in data that was first introduced by Fei Tony Liu and Zhi-Hua Zhou in 2008. The approach employs binary

trees to detect anomalies, resulting in a linear time complexity and low memory usage that is well-suited for processing large datasets.Isolation Forest does it by introducing (an ensemble of) binary trees that recursively generates partitions by randomly selecting a feature and then randomly selecting a split value for the feature. The partitioning process will continue until it separates all the data points from the rest of the samples.Isolation Forest is an unsupervised machine learning algorithm for anomaly detection. As the name implies, Isolation Forest is an ensemble method (similar to random forest). In other words, it use the average of the predictions by several decision trees when assigning the final anomaly score to a given data point.

Q11 A global outlier is a measured sample point that has a very high or a very low value relative to all the values in a dataset. For example, if 99 out of 100 points have values between 300 and 400, but the 100th point has a value of 750, the 100th point may be a global outlier.

Outlier detection is extensively used in a wide variety of applications such as military surveillance for enemy activities to prevent attacks, intrusion detection in cyber security, fraud detection for credit cards, insurance or health care and fault detection in safety critical systems and in various kind of images.