Anomaly Detection using proximity and clustering based technique

DATA MINING

By: AMNA ZAFAR
ID: 21090449

Anomaly Detection using proximity and clustering based technique

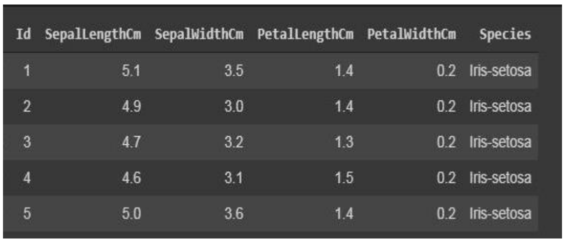
Introduction: Anomaly detection is a process of identifying or detecting data points that fall outside the expected behaviour of normal data points in a dataset. Those unusual data points are called outliers or anomalies. This kind of outliers generate deviations in system so there is need to detect them by using different anomaly detection mechanisms. This kind of mechanisms uses different data mining techniques to detect abnormal patterns in data.

Anomaly detection methods: There are two major anomaly detection techniques: supervised anomaly detection and unsupervised anomaly detection. Supervised methods require a labelled training set that need human intervention for performing classification that comprises both normal and anomalous samples. On the other hand, unsupervised methods need a computer that may learn to detect complex anomalies and outliers without the assistance of a person. Unsupervised anomaly detection includes neural network, clustering or proximity-based algorithms. The algorithms that are used for performing anomaly detection is clustering-based approach (K-mean clustering) and proximity-based approach (local factor outlier).

Clustering based (k-means) approach: Clustering is a process of dividing all data points present in population or dataset into a number of groups such that data points with similar attributes come into same group. K mean clustering is one of the most commonly used clustering algorithms in which there is no need to label the data. In this model, we have to limit cluster numbers and fit the model on dataset to specify center of each cluster known as centroid. After that, we need to divide our data into different clusters based on the similarity in their characteristics. For this division, we calculate the distance of each data point from closest centroid. Mathematically, distance measurement function like Euclidean distance is used for this purpose. K-means clustering which helps to cluster the data points uses Euclidean distance internally. In this type of clustering, inter-clustering and intra-clustering distances both are used to specify the data points that are far from centroid of clusters. Such kind of long-distance data points are considered as anomalies by showing their distance from the edge of clusters.

Proximity based (Local outlier factor) approach: Proximity-based methods perform anomaly detection by assuming an object an outlier if its nearest neighbors are far away from it. Local outlier factor, a type of proximity-based anomaly detection method detects anomalies by computing the local density deviation of each data point and giving a score in the range of [0,1] as output. After defining the model, we have to compare the local density of each data point to the local densities of its k nearest neighbors. This comparison helps in identifying the regions of similar density that are helpful to find data points that have considerably lower density values than their neighbors. Such data points have different behavior than normal data points. If a data point has a noticeably lower density than its neighbors then it is considered as outlier because of having low score value.

Dataset: In this project, Iris flower dataset is used for anomaly detection. We will perform unsupervised anomaly detection on its three related species. It contains information on four attributes of each specie: Sepal length, Sepal width, Petal length, Petal width. Learning models (K-means clustering and local outlier factor) are performed on selected dataset to detect the mis-labelled entries (anomalies).



Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3.0	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
4	4.6	3.1	1.5	0.2	Iris-setosa
5	5.0	3.6	1.4	0.2	Iris-setosa

Implementation of anomaly detection: Anomaly detection can be performed using boxplot as shown in Figure 1. Notice some tiny circles in second box that are the data points that deviates tremendously from the rest of the data.

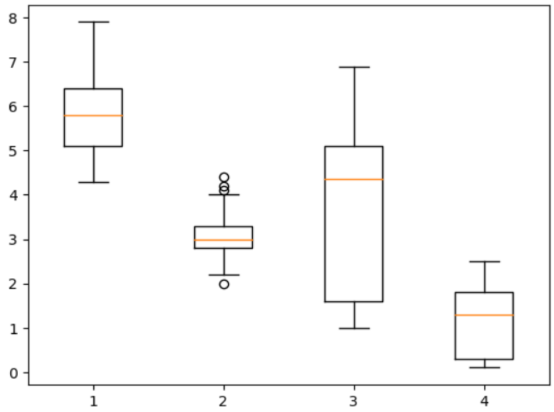


Figure 1: Boxplot of Iris Dataset

K-means clustering approach: In this approach, first of all Scikit-learn API is used to import K-means class. Furthermore, after defining and fitting the model, elbow method is used to determine the optimal number of clusters for performing anomaly detection on dataset as shown in Figure 2.

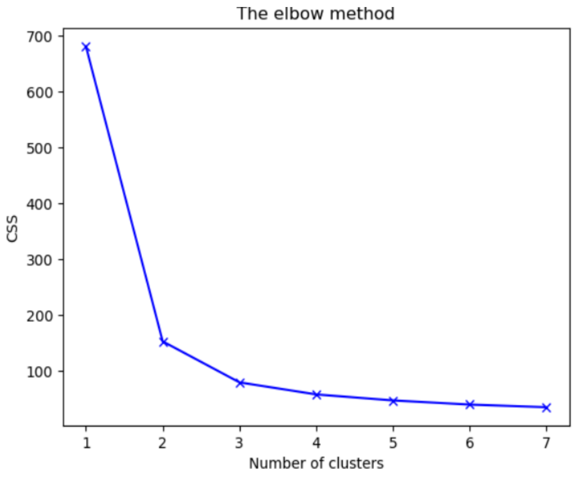


Figure 2: Elbow Method

The above figure shows change at point 3. That's why K-means clustering is applied on dataset with clusters K = 3. Results of K-means clustering are shown in Figure 3:

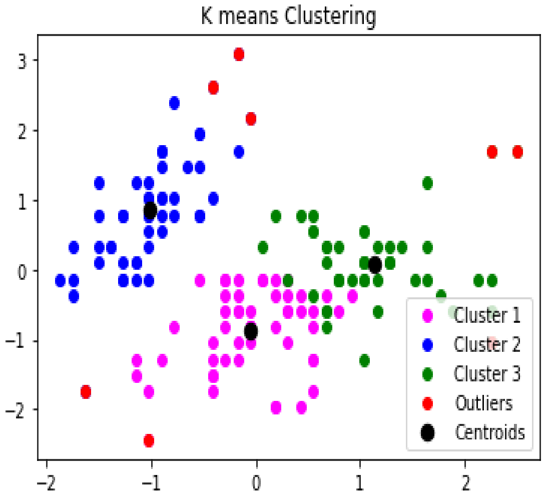


Figure 3: K-means based Anomaly Detection

The above scatter plot demonstrates that there are three centroids present that are black in colors and there are also three clusters representing three species that are plotted around them by calculating distance of each data point from centroid. Whereas, red colored data points are anomalies that are behaving differently in dataset.

But after applying this algorithm, it is concluded that K-means is not the ideal method for detecting anomalies or outliers. This method is quite simple and easy to implement that's why, it is used widely, but its simplicity also arises several disadvantages. One of them is its sensitivity to anomalies or outliers, as it uses Euclidean distance for detecting noisy data present in dataset. As we know, real-world data sets come with different types of outliers, by using K-means, we might be unable to detect and remove them completely. Moreover, simple K-means algorithm cannot perform clustering effectively several times due to the existence of outliers in dataset. An insight we get clearly from above scatterplot is K-means model's accuracy in determining cluster 1 and cluster 3 is comparatively less to cluster 2.

Local outlier factor approach: In this method, local factor outlier class is imported by using Scikit-learn API. Then, we define and fit the model by using the default value of 20 nearest neighbors to measure the local density of each data points. After that, anomalies are visualized by using matplotlib, where the LOF score is not 1.

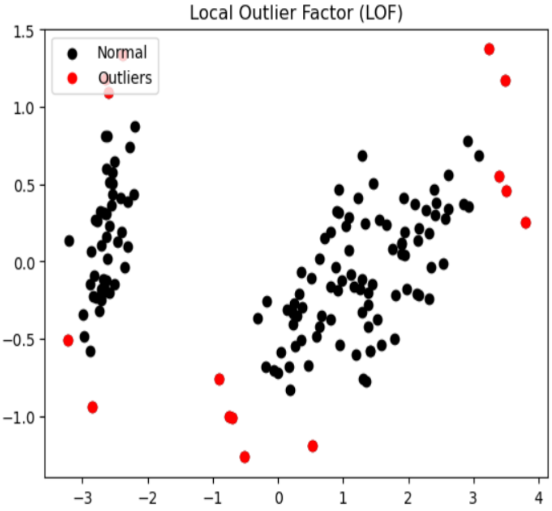


Figure 4: Local Outlier Factor based Anomaly Detection

Figure 4 is clearly differentiating outliers from normal dataset. Red data points are considered anomalies that behave differently than normal data points where as black points are normal data points of iris dataset.

This algorithm shows that LOF is considered as an effective anomaly or outlier detection algorithm that easily identifies abnormal data points present in dataset depending on their neighborhood. Moreover, it gives better results as compared to other simple distance-based models because it can easily handle data present in different regions with different densities.

References:

1) Agrawal, S. and Agrawal, J. (2015) "Survey on anomaly detection using data mining techniques," *Procedia computer science*, 60, pp. 708–713. doi: 10.1016/j.procs.2015.08.220.
2) Gan, G. and Ng, M. K.-P. (2017) "K -means clustering with outlier removal," *Pattern recognition letters*, 90, pp. 8–14. doi: 10.1016/j.patrec.2017.03.008.
3) Samariya, D. and Thakkar, A. (2021) "A comprehensive survey of anomaly detection algorithms," *Annals of data science*. doi: 10.1007/s40745-021-00362-9.
4) Dino, L. (2022b) *Outlier detection using K-means clustering in python*, *Towards Dev*. Available at: https://towardsdev.com/outlier-detection-using-k-means-clustering-in-python-214188fc90e8 (Accessed: January 16, 2023).
5) *Anomaly detection (K-means)* (no date) *Edgeimpulse.com*. Available at: https://docs.edgeimpulse.com/docs/edge-impulse-studio/learning-blocks/anomaly-detection (Accessed: January 16, 2023).
6) Jayaswal, V. (2020) *Local Outlier Factor (LOF) — Algorithm for outlier identification*, *Towards Data Science*. Available at: https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843 (Accessed: January 16, 2023).
7) Kumar, P. (2020) *Understanding LOF (Local Outlier Factor) —perspective for implementation*, *Medium*. Available at: https://medium.com/@pramodch/understanding-lof-local-outlier-factor-for-implementation-1f6d4ff13ab9 (Accessed: January 16, 2023).
8) Bajaj, A. (2022) *Anomaly detection in time series*, *neptune.ai*. Available at: https://neptune.ai/blog/anomaly-detection-in-time-series (Accessed: January 16, 2023).