Theoretical Study by Hasin Us Sami:

An outlier may be explained as a portion of data or observation that lies exceptionally far from the average of the data set. Features/properties of these outlier vary to a great extent from that of normal data points.

Since our project focuses on detecting outlier in quasars, or in other way, data point that deviates from the normal behavior of quasars, a number of outlier detection algorithms has been studied. Among them, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) has been chosen for our project as it is one of the most efficient algorithms in detecting outliers. Density based clustering algorithms detect different clusters by finding the areas with a higher concentration of data points. And the data points that are scatted all over the space and not part of a concentrated area, are detected as outliers.

For DBSCAN algorithm, there are two parameters that need to be specified –

1. **Epsilon** : It defines a certain region around a data point where a minimum number of data points need to present in order to form a cluster. An optimal value for this parameter is chosen based on the distribution of data so that there would be less possibility for an outlier to detected as part of a cluster or a normal data point to be detected as an outlier.
2. **Minimum Sample Points**: It defines minimum number of data points that need to be present within epsilon in order to be qualified as a cluster. The more concentrated the data points, the larger value of minimum sample point is chosen.

In paper [1], drawbacks of traditional DBSCAN algorithm have been discussed and a more improved version of this algorithm has been introduced and applied into image segmentation. Though DBSCAN is one the most efficient algorithms in clustering and detecting outliers, the main drawback of this algorithm is- the parameters epsilon and minimum sample points need to be prespecified. By trial and method, an optimum value is chosen which works best for that specific dataset. If any changes in the dataset is made, those parameters need to be changed as well. That's why this paper has introduced a technique/formula to calculate these parameters by combining the concept of K-nearest neighbor. A curve is plotted to examine the behavior of the distance from a point to its k^{th} nearest neighbor. When points are placed on x axis based on their increasing distance from k^{th} nearest neighbor and the distance is placed on the y axis, the sharp modification/increase in the curve resembles to the optimum value for epsilon. Though this technique has been applied on image dataset for segmentation purpose, it can be extended to the quasar dataset in our project for outlier detection purpose. But the limitation of the paper is- the technique they developed to calculate optimum for min_sample points based on the size of the pixel image and grey scale value(256), can be used for image dataset only and thus cannot be used for our project.

In paper [2], another drawback of DBSCAN algorithm has been brought into light. Since dataset is very large nowadays, DBSCAN is susceptible to high computation cost and memory allocation problem. To overcome this issue, this paper has introduced an efficient DBSCAN algorithm using Map-Reduce where 3-stage approach is adopted- data partitioning, local clustering and global merging. Each partition is independently processed and thus computation time is significantly

reduced. Each partition is assumed small enough to fit on memory and so memory allocation issue is also overcome. But it has not shed light in any technique to determine optimum value for epsilon or minimum sample point which is very much important for our project.

DBSCAN Algorithm Implementation by Hasin Us Sami:

For each data point i:

-if not visited before, mark it as a visited point.

-find the neighboring points within the range of epsilon.

If number of neighboring points within the region is less than the minimum points:

-don't assign a cluster number.

else

-assign a unique cluster number to the point.

-Store all the neighboring points in an array.

-**For** all neighboring points:

-mark it as a visited point.

If a cluster number has not been already assigned to this point:

-assign the cluster number it belongs to.

-Find all other points within the range of epsilon.

-Concatenate these new points to the array of neighbors if they don't already belong to that array.

(Repeat the above steps for all the data points in the dataset. Those points that don't belong to any clusters are detected as anomaly/outliers)

Implementation in Python:

Before applying self-implemented algorithm, DBSCAN built-in function in python has been experimented on the pre-processed dataset with 144582 data points and 21 features just to verify how much accurate this detection technique would be in the case of our quasar dataset. Optimum values for DBSCAN parameters have been chosen using trial and error method. After performing DBSCAN, Principal Component Analysis(PCA) function has been used to map the data points into 2D plane for the purpose of visualization.

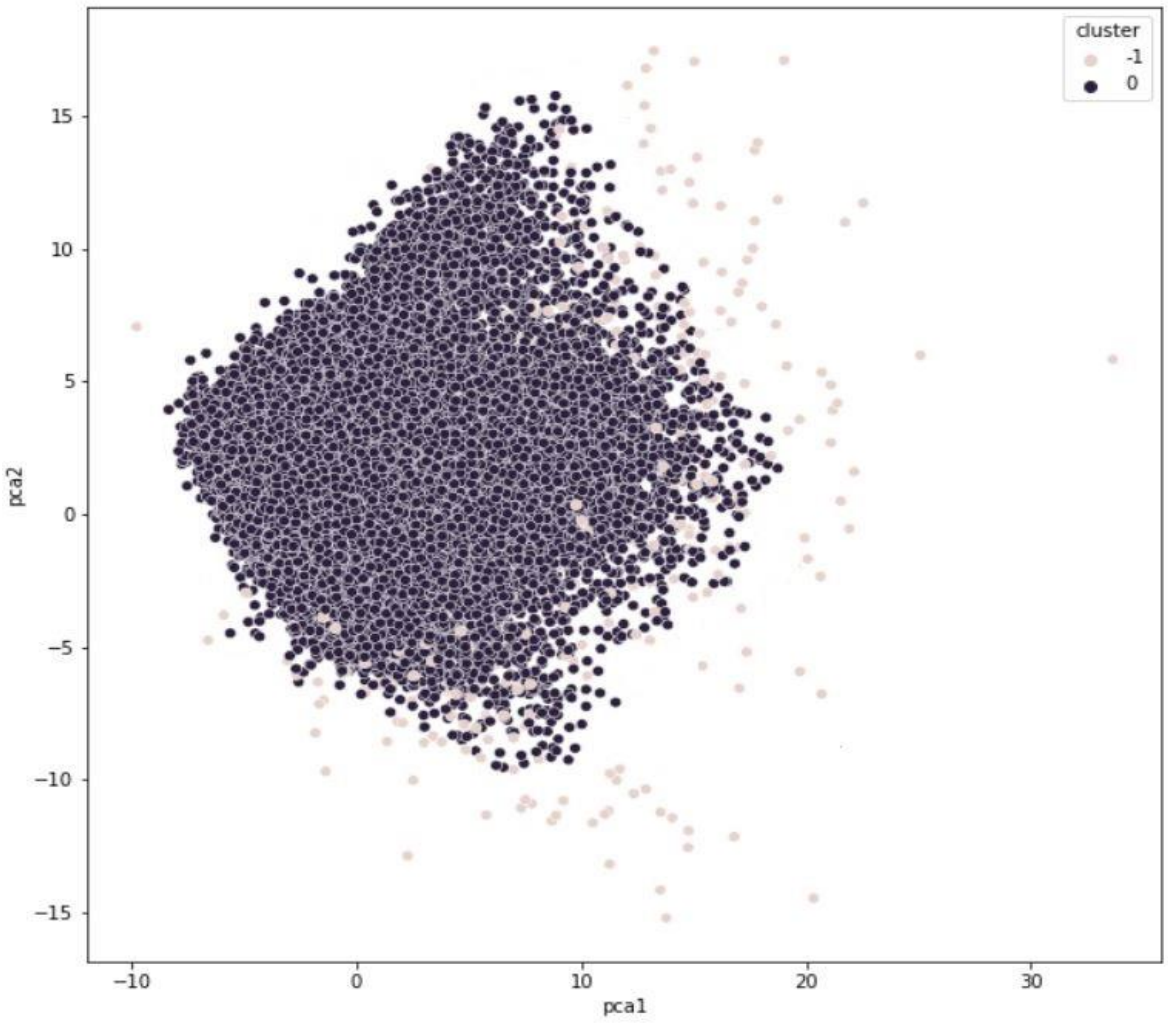

Figure: 2-D Visualization of the data points (-1 circles are outliers)

The work is in progress and focus is on to further increase the accuracy. The algorithm that has been described above, has been implemented in python but unfortunately there are several errors observed in the code that has to be dealt with. The next step to do is- fix the errors and run the algorithm on our quasar dataset to detect outliers.

Reference:

*1.**Suresh Kurumalla , P Srinivasa Rao, "K-Nearest Neighbor Based DBSCAN Clustering Algorithm for Image Segmentation" Journal of Theoretical and Applied Information Technology . Vol.92. No.2,2016***

*2. **He, Y., Tan, H., Luo, W., Feng, S., & Fan, J. (2013). "MR-DBSCAN: a scalable MapReduce-based DBSCAN algorithm for heavily skewed dat". Frontiers of Computer Science, 8(1), 83–99.***