**Study of Outlier Detection Techniques for Unsupervised Learning**

**Team Members**

**Amit Shetty (801100195)**

**Tabish Maniar (801077755)**

**Vamshi Goud (801100601)**

1. **Introduction**

This report details the study of three outlier detection techniques available as part of the Anomaly Detection plugin of the data mining software, Rapidminer Studio. How any attribute can be an outlier is determined based on the Outlier Score generated by an algorithm. The three approaches that determine this outlier score are discussed are as follows:

1. Histogram Based Outlier Score
2. Cluster Based Outlier Score
3. K Nearest Neighbour Based Outlier Score

All the algorithms described in this report are part of the many unsupervised learning techniques that are available. Furthermore, the outlier detection algorithms described here make use of non parametric techniques to detects data anomalies. Non parametric methods do not rely on any underlying uniform distribution such as a normal distribution compare against any outlier data.

Histogram based outlier score can be described in short as a highly efficient and versatile data anomaly detection algorithm. Efficient because it calculates the outlier score in linear time. Versatile because it has sub techniques that can account for narrow or wide gaps in the data provided.

Clustering based anomaly detection works on the output of a clustering algorithm. It uses the objects cluster knowledge to determine outlier factor.

Nearest neighbour algorithms detect anomalies by calculating either the distance between the data objects or by calculating their density with respective to their neighborhood. They work based on the assumption that if the data objects exhibit different features they can be distinguished as outliers by calculating the distances and densities with respective to their nearest neighbours

Each of the algorithms are discussed in detail below.

**2. Histogram Based Outlier Score (HBOS):**

The process of finding anomalies in data without prior training is called Unsupervised Anomaly detection. The technique currently in discussion is the Histogram Based Outlier Score (HBOS). It is a type of Statistical outlier detection algorithm. Statistical methods (also known as model-based methods) make assumptions of data normality. They assume that normal data objects are generated by a statistical (stochastic) model, and that data not following the model are outliers.

The technique will be discussed in detail further but in short, the reason for using this Algorithm is that it detects anomalies in real time. It assumes independence of the features making it much faster than multivariate approaches at the cost of less precision. This technique also falls under the category of non-parametric outlier calculation. This means that any idea that the algorithm has of “normal” data comes from the input file itself. This falls in line with unsupervised learning.

HBOS shines mostly in semi supervised and unsupervised anomaly detection techniques. Sine it detects anomalies in linear time it is considered highly efficient. HBOS is highly efficient on extremely large datasets because of the way it works. Regardless of how many variables are presented in the datasets, HBOS won’t work on multivariate datasets all at the same time. It calculates that outliers in one single column and uses statistics measurements to determine the outliers in that column. Once all the columns are traversed and the outliers are merged the final outlier score is calculated. The presented HBOS algorithm allows applying histogram-based anomaly detection in a general way and is also available as open source as part of the anomaly detection extension of RapidMiner.

For each single variable, a univariate histogram is constructed first. For category-based feature data, simple counting of the values of each category is performed and the relative frequency (height of the histogram) is computed. If the data is numerical, HBOS offers two different calculation techniques i.e. Static bin width and dynamic bin width histogram techniques.

Static bin width histograms use k equal width bins over the range of values present in each category. How frequent the categories fall into each bin determine the density of the bin.

Dynamic bin width histogram calculation works differently for numerical allocation. All values are first sorted. Then taking N i.e. the total number of values in the data sets and dividing it with the k bins of equal width as discussed in static bin width histograms and the result is stored in a single bin. Since the values are now sorted, the height of each bin can be determined by just taking the range and since the number of elements in each bin determine the density, it can be concluded that bins covering a very large value range will have less density and ones with a shorter range will have more density.

In the real world, both these techniques have their advantages and disadvantages. Companies that handle credit card and medical data needs to have large intervals in their data collection to mandate any sort of outlier alert. But outlier calculation in most cases is entirely subjective i.e. one needs to manually check what the threshold to determine an outlier is. In case of large intervals in the data, static bin widths will produce an inaccurate result since it might detect a valid data point as an outlier.

Histogram based outlier score can be summed up with the following formula

HBOS(p)= =

where,

d = dimension data

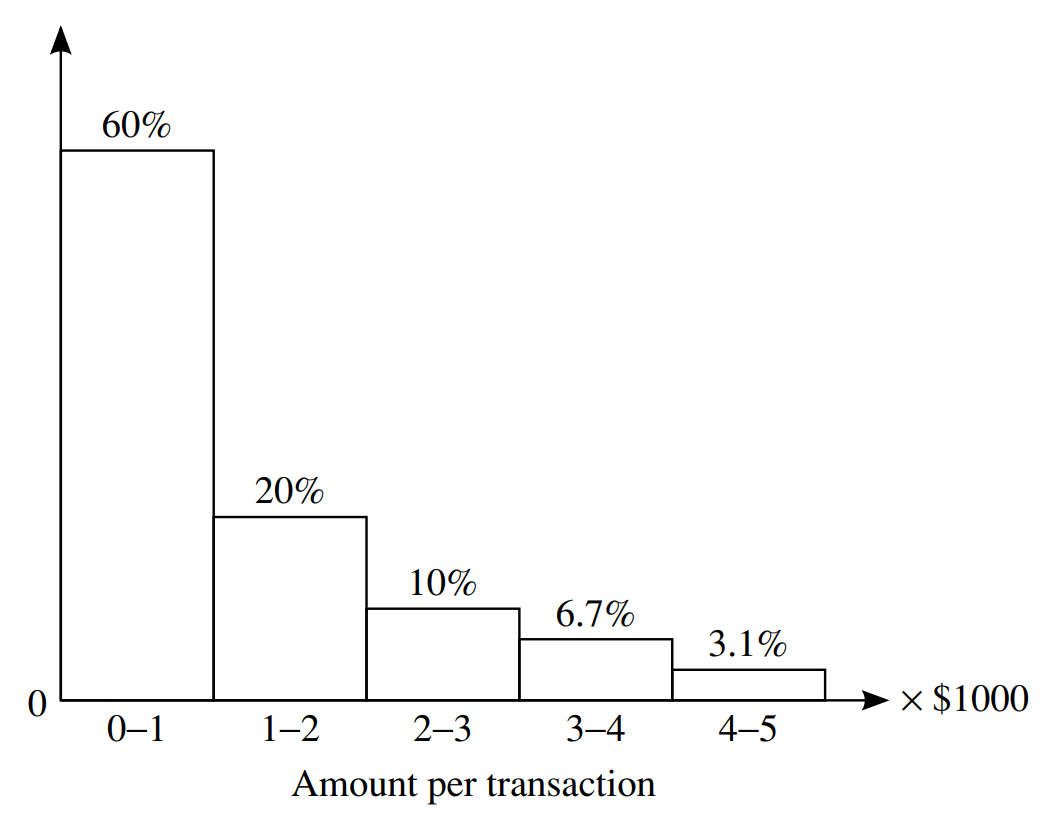
p= instance of classification of data

hist() = Histogram calculation of every attribute of the dataset which is normalized to fall in the range of [0,1]

Data for each instance is calculated and summed up to get a final outlier score for the entire dataset. The log values are considered since it reduces the error due to floating point precision.

The main disadvantage of HBOS is that it works very well on a large dataset where a global outlier score is needed but fails to get accurate results for local outlier score where local group data is considered.

**2.1 Application Example:**



Take for example, a credit card company is trying to detect anomalies in a random customer’s usage pattern. The first step would be to take into account the person financial history and plot a histogram as done in the image above. Ear bar here represents a bucket that stores the user’s credit card spending in thousand-dollar iterations. The maximum range the histogram is detecting is $5000. Anything above $5000 would return the following outlier score:

Formula is 1 – (densities of all the normal data in the input) for calculating the outlier.

Subtracting 1 since any amount above $5000 doesn’t fall in our histogram range.

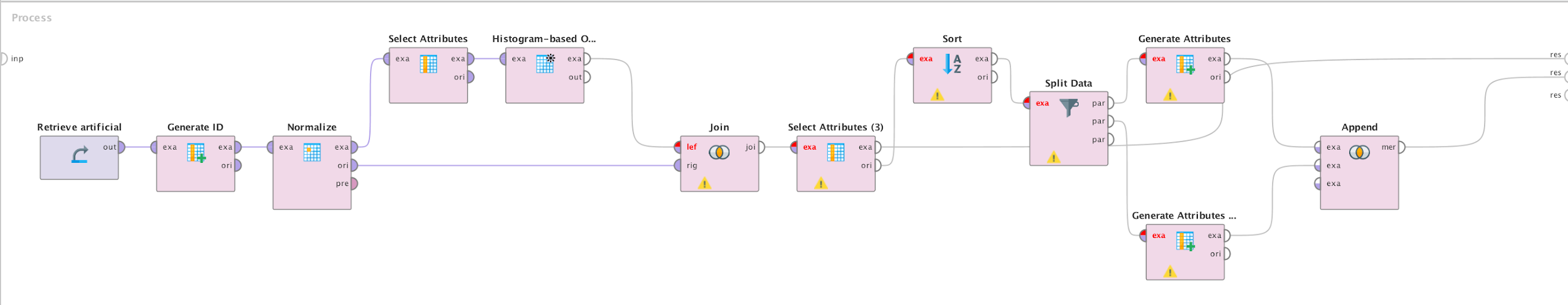


Since 2% doesn’t fall within the range of the histogram, it can be called an outlier.

**2.2 Performance Metrics:**

As discussed earlier, HBOS calculates the outlier per column in linear time. Based on the formula it then sums up all the results to get a final outlier score. Due to this HBOS runs on liner time. The time complexities for Static bin width histogram calculation is O(n) and dynamic width histogram calculation is O(nlog(n))

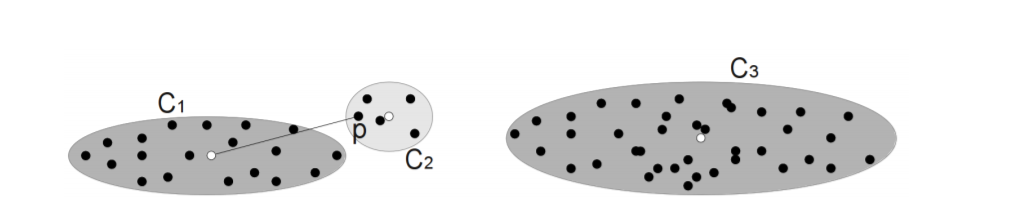
**2.3 RapidMiner Studio Process implementation**

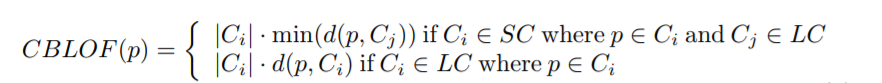


**3. Cluster Based Outlier Score and Anomaly Detection**

A clustering algorithm tries to merge similar objects into one cluster. Usually, cluster of different sizes is given ranging from 1 to n. The Cluster Based Outlier Score is calculated on the clustering algorithm which are highly efficient such as K-Means, K-medoids. The input is take from these clusters. Cluster Based Outlier Detection assumes that the objects which are anomaly are either in a small cluster away from the larger cluster or they are not assigned any cluster at all.

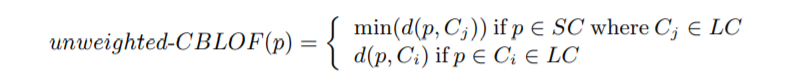
The CBLOF Score is calculated by distance to the nearest large cluster multiplied by the size of the cluster which the object belongs to.



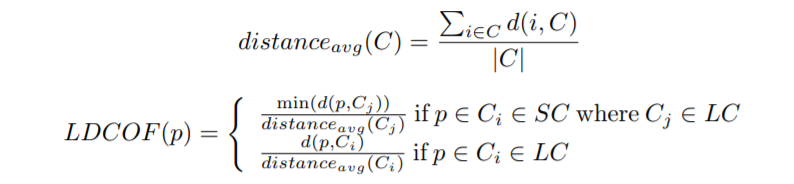


Here, for the calculation of Outlier score, we consider the point p with the nearest large cluster i.e C1 and multiplied by the size of cluster C2 belongs.

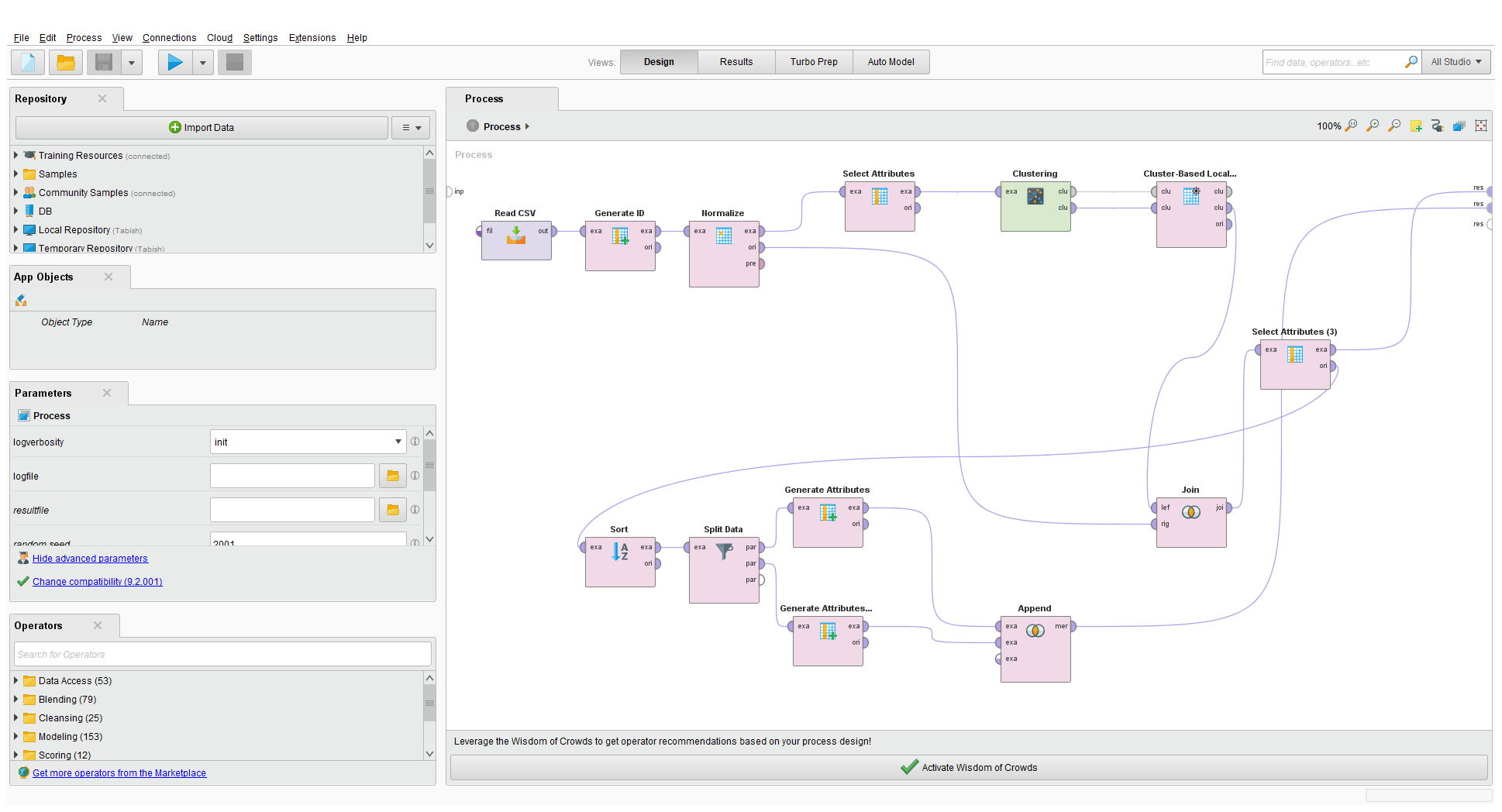
In Unweighted CBLOF, the size of the cluster where the object belongs is not considered only the distance to nearest large cluster is considered.



Another algorithm is Locality Density Cluster-Based Outlier Factor. The LDCOF score is defined as the distance to the nearest large cluster as divided by the average distance to the cluster center of the elements in that large cluster. The intuition behind it as the objects in the smaller cluster is assigned to the larger cluster which becomes its local neighborhood.

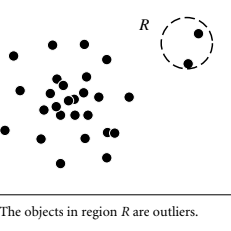


**3.1 RapidMiner Studio Process Implementation**



**4. Nearest Neighbour Based Algorithms**

Nearest neighbour based algorithms (proximity based methods) are used to detect data objects that are outliers, they assume that an object is an outlier if the nearest neighbors of the object are far away in feature space, that is, the proximity of the object to its neighbors significantly deviates from the proximity of most of the other objects to their neighbors in the same data set.



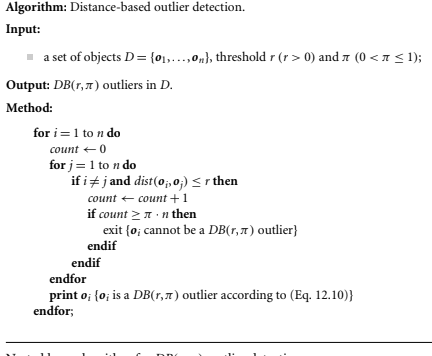
When we use the concept of nearest neighbour to detect the outliers in the above figure, suppose we use 3-NN algorithm the second and third nearest neighbors of the objects in R are farther away from it therefore based on the proximity we can name the data objects in R as outliers.

There are basically two ways of detecting outliers using proximity based methods

1. Distance- based outlier detection: A distance-based outlier detection method consults the neighborhood of an object, which is defined by a given radius. An object is then considered an outlier if its neighborhood does not have enough other points.
2. Density-based outlier detection : A density-based outlier detection method investigates the density of an object and that of its neighbors. Here, an object is identified as an outlier if its density is relatively much lower than that of its neighbors.

To compute distance based outliers two approaches are used first approach is a straight forward nested loop approach is used to check the r-neighbourhood of every object For any object, oi (1 ≤ i ≤ n), we distance between oi and the other object, and count the number of other objects in the r-neighborhood of oi. Once we find π ·n other data objects.

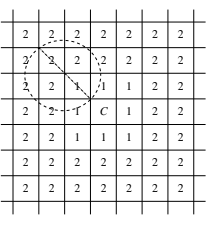
Below picture depicts the algorithm used



For the objects within a distance of r from oi, the inner loop can be terminated because oi already violates (Eq. 12.10), and thus is not a DB(r,π)-outlier. On the other hand, if the inner loop completes for oi, this means that oi has less than π ·n neighbors in a radius of r, and thus is a DB(r,π)-outlier.

The straightforward nested loop approach takes O(n2) time.

Another approach used is CELL which is a grid-based method



Here there are two rules based on which we classify a group of data objects as either outliers or non-outliers.

Level-1 cell pruning rule: Based on the level-1 cell property, if a+ b1 > ⌈πn⌉, then every object o in C is not a DB(r,π)-outlier because all those objects in C and the level-1 cells are in the r-neighborhood of o, and there are at least ⌈πn⌉ such neighbors.

Level-2 cell pruning rule: Based on the level-2 cell property, if a+ b1 + b2 < ⌈πn⌉+1, then all objects in C are DB(r,π)-outliers because each of their r- neighbourhoods has less than ⌈πn⌉ other objects. Based on these rules we can determine whether a group of data objects are outliers or not and there is no need to check the objects one by one thereby reducing the time complexity.

Here let *a* be the number of objects in cell *C*, *b*1 be the total number of objects in the level-1 cells, and *b*2 be the total number of objects in the level-2 cells

**Density Based outlier Detection**

The basic assumption of density-based outlier detection methods is that the density around a nonoutlier object is similar to the density around its neighbors, while the density around an outlier object is significantly different from the density around its neighbors.

The local reachability density of an object o is defined by

C:\Users\Vamshi\Desktop\formula.PNG

So generally there are six types of nearest neighbour based algorithms used to detect the outliers.

1. k-NN Global Anomaly Score: It is most commonly used nearest neighbour algorithm.The anomaly score is calculated as the average distance of the k-nearest-neighbors to avoid statistical fluctuations.

2. Local Outlier Factor (LOF): The local density of a data instance is inversely proportional to the average distance to the k-nearest- neighbors. The LOF score is the ratio of the local density of the data instance to the average local density of its neighbors.

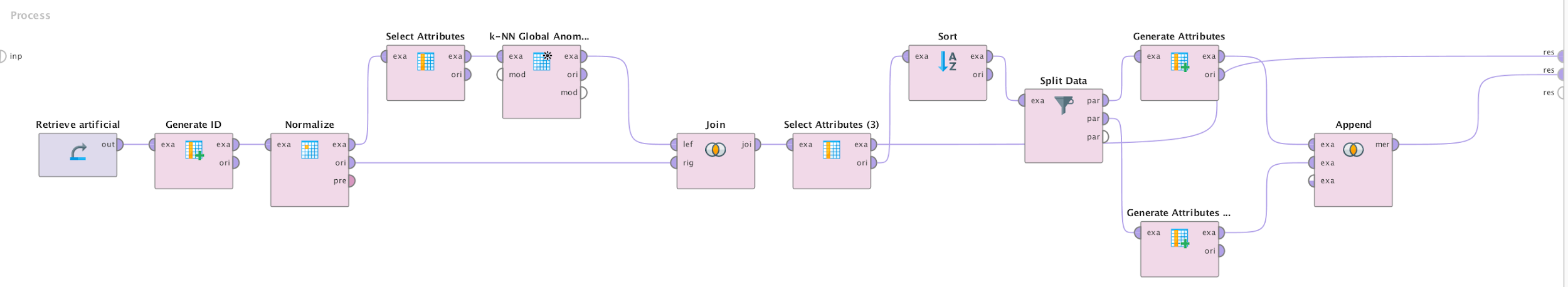
3. Connectivity based Outlier Factor (COF) : this was proposed in order to handle outliers deviating from spherical density patterns, for example straight lines.

4. Local Outlier Probability (LoOP) : incorporates some statistical concepts in order to output the final score as a probability that a particular data instance is a local density outlier. These probabilities facilitate the comparison of a data instance with data in the same dataset as well as in other datasets.

5. Influenced Outlierness (INFLO): it was introduced in order to give more accurate results in case of clusters with varying densities that lie near to each other.

6. Local Correlation Integral (LOCI):It is variant of LOF. In contrast to LOF, the density of a data instance in LOCI is proportional to the number of objects within a particular radius, the r-neighbourhood. Unfortunately it also causes an increase in both, time O(n3) and space complexity O(n2), restricting its use to very small datasets.

**4.1 RapidMiner Studio Process Implementation**



**5.0 Conclusion**

As shown above, the three outlier detection algorithms showcase how the outlier score can be different for the same dataset. This is entirely due to the fact that they make use of entirely different techniques and principles to get the outlier score. Outlier detection in many cases are subjective. Any score can be determined on a case to case to be an outlier. The detection algorithms studies here can give a user the options they can exercise to detect any anomalies regardless of the domain they work in.