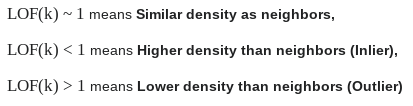
# **Anomaly Analysis**

## Local Outlier Factor

Finds the average distance to the nearest neighbour, then compares it with the k nearest neighbours of that neighbour. This process is iterated across the first k neighbours, and intuitively, it provides a means of comparing local density of a point with densities of its neighbouring points. More specifically, local density is estimated in LOF by the mean distance at which a data point is separated from its neighbours. These density values are normalised to such that



## Isolation Forest

Method of detecting anomalies through the employment of decision trees. At each node is a randomly selected split point with respect to a random variable. An ensemble of these random trees create the isolation forest, whereby an anomaly score is created using the path-length measure for a data point. The more binary splits required to isolate the data point, the less of an outlier it is considered to be. On average, the outlier data can be described in less information. In order to isolate an outlier, the algorithm recursively generates partitions on the sample by randomly selecting an attribute and then randomly selecting a split value in the feature appropriate range.

## Advantages - LOF

Local comparisons is an advantage - allows for clustered data to exist in a distribution of densities, which when viewed globally has high variance. This beats the global approach which may misidentify a sparse data cluster as outliers.

Can be easily generalised to fit many things (local outlier factor could be used for geographic based data)

## Disadvantages - LOF

No clear rule to distinguish when an object is considered ‘outlier’. USE BAGGING - SEE ENSEMBLE LEARNING APPROACH TO OUTLIER DETECTION

## Advantages - Isolation Forest

Works well for high-volume data due to low complexity.

Works well with low sampling size.

## Disadvantages - Isolation Forest

Due to random nature, can be subject to random error in classifications.

# Dominic’s Custom Anomaly Analysis:

For use on data that has been regressed:

Steps of the model:

1. Splits data into folds - i.e if there are k folds, you separate the data in k sections and then for each fold you use it as testing data, and the rest as training/regressing data.
2. Then calculate the median absolute error on the testing fold against the regression line on the training fold, *for each fold*.
3. Remove outliers *with each value of the hyperparameter known as contamination*

using LOF, and calculate the loss with respect to the testing fold.

1. You select the value of the hyperparameter that reduces the loss on all the folds the most, with consistency in mind.
2. Once found, you use this contamination value to remove outliers on the original data using LOF.

## Sensitivity analysis

References:

<https://medium.com/@sumitpatil351/decoding-outliers-a-comprehensive-guide-to-techniques-and-implementation-for-robust-data-analysis-6810296c2af7>

<https://en.wikipedia.org/wiki/Isolation_forest#:~:text=Isolation%20Forest%20is%20an%20algorithm,well%20with%20high%2Dvolume%20data>.

# **Dominic Notes**

* LocalOutlierFactor
* IsolationForest
* Use any regression model as reg
  + Splits data into folds
  + For each fold:
    - Calculate median loss
    - Remove outliers with each value of hyperparameter
    - Calculate new losses
    - If losses below original losses for all folds, it is treated as a significant result
  + Use imputer to fill in missing data (if necessary)
  + Go forward as normal to rest of model z