Survey of Outlier Detection Methods

Brandon Sim
AC 299r

1 Summary

The literature review classifies outlier detection methods into three groups: types I, II, and III. Type I consists of **unsupervised clustering** algorithms, in which data is processed as a static distribution, and the most remote points are flagged as outliers. Type II consists of **supervised classification** algorithms, in which pre-labelled data is required and tagged as either normal or abnormal. Such algorithms require data that represents a good spread of both normal and abnormal data, allowing for learning to be done appropriately by the classifier. Finally, type III consists of **semi-superverised recognition or detection** algorithms, in which normality is modelled (taught) but the algorithm must learn to recognize abnormality. The approach requires pre-classified data but needs only to learn data marked as normal; it can learn the abnormality as new data is added.

1.1 Proximity-based techniques

1.1.1 Type I

Methods of this type include:

1. k-nearest neighbor algorithm: using a suitable distance metric (Euclidean, Mahalanobis)
2. Optimized k-NN (Ramaswamy et al, 2000) to produce a ranked list of potential outliers; a point p is an outlier if no more than $n-1$ other points in the data set have a higher D_m (distance to m-th neighbor) where m is user-specified.
3. Knorr and Ng (1998) use an efficient type 1 k-NN approach. If m of the k nearest neighbors (for $m < k$) lie within a threshold d then the point lies in a sufficiently dense neighborhood (i.e., is not an outlier). However, this requires learning the parameters d, m, k.
4. Weighted k-NN with connectivity-based approach (Tang 2002): calculates a weighted distance score rather than a weighted classification; calculates the average chaining distance (path length) between a point p and its k neighbors. If it is higher than a certain cutoff t then it is deemed abnormal.
5. k-means; k-medoids; if a new point lies outside existing clusters (where radius of cluster is defined as center to farthest point), then it is an outlier.

1.1.2 Type II

:

1. Majority voting approach (Wettschereck, 1994): using a labelled data set with normal and abnormal vectors classified, classifies a point according to the majority classification of the nearest neighbors. Or, where the voting power of nearest neighbors decreases according to its distance from the point.

1.2 Parametric methods

Proximity-based techniques often do not scale well to large datasets due to speed concerns. On the other hand, parametric methods can be evaluated rapidly for new data; model grows with model complexity and not data size. However, a pre-selected model must then be enforced to the data, losing some flexibility.

1. Minimum volume ellipsoid estimation (Rousseuw and Leroy, 1996): fit the smallest permissible ellipsoid volume around the majority of the data distribution model.

2. Convex peeling (Rousseuw and Leroy, 1996): construct a convex hull around points, peel away points on the boundaries as outliers.

3. Maximal influence regression line (Torr and Murray, 1993): run OLSR; remove point which has maximum influence (causes greatest deviation in placement of regression line).

1.3 Semi-parametric methods

1. Gaussian mixture models (Roberts and Tarassenko, 1995; Bishop 1994)

2. Extreme value theory in Gaussian mixture models (Roberts, 1998): examine distribution tails and estimate probability that a given instance is an extreme value in an exponential distribution model.

3. Support vector machines (Tax et al, 1999; Decoste and Levine, 2000).

1.4 Supervised neural methods

1. Multi-layer perceptron (Nairac et al, 1999; Bishop, 1994)

1.5 Unsupervised neural methods

1. Self organizing maps (Kohonen, 1997): competitive, unsupervised neural networks. Perform vector quantization and non-linear mapping to project data distribution onto lower dimension grid with user-specified topology.

2. Adaptive resonance theory (ART) (Caudell and Newman, 1993): network which is plastic while learning, stable while classifying, can return to plasticity to learn again; ideal for time-series monitoring.

1.6 Machine Learning

1. Decision trees