Outlier detection in high-dimensional spaces using one-dimensional neighborhoods

Joris Falip, Frédéric Blanchard, Michel Herbin

Outlier detection in high-dimensional spaces using one-dimensional neighborhoods

Joris Falip*, Frédéric Blanchard* Michel Herbin*

*CReSTIC, Université de Reims Champagne Ardenne
joris.falip@univ-reims.fr,
http://crestic.univ-reims.fr

Abstract. Detecting outliers in a dataset is a problem with numerous applications in data analysis for fields such as medical care, finance, and banking or network surveillance. But in a majority of use-cases, data points are described by a lot of features, making outlier detection more complicated. As the number of dimensions increases, the notion of proximity becomes less meaningful: this is due to sparser data and elements becoming almost equally distant from each other. Medical datasets add a layer of complexity caused by their heterogeneous nature. Because of these caveats, standard algorithms become less relevant when hundred of dimensions are involved. This paper discusses the benefits of an outlier detection algorithm that uses a simple concept of one-dimensional neighborhood observations to circumvent the problems mentioned previously.

1 Introduction

The outlier detection problem has a lot of practical applications in data science for a variety of fields. Finding outliers can be used to filter them out when preprocessing a dataset, to detect unusual behavior for monitoring purposes or even to correct inaccuracies or typing errors during data collection. When it comes to medical data, outliers can be seen as patients with unique characteristics. These atypical cases are valuable to medical experts, as they represent rarely encountered diseases that could help doctors in treating new patients. Outliers detection algorithms usually rely on a notion of proximity which is calculated using a distance metric such as the Euclidean distance. While this method performs well on low-dimensional spaces, the curse of dimensionality (Donoho et al., 2000) quickly reduces the quality of results obtained using this approach. A high number of dimensions means that elements become almost equally distant from each other (Aggarwal et al., 2001). Moreover, medical data are heterogeneous, thus requiring flexible solutions to fit the information stored in electronic medical records.

This paper, part of an ongoing work on medical datasets (Falip et al., 2017), proposes an algorithmic description and implementation of the rareness concept formulated in a previous article by Herbin et al. (2017). The described outlier detection method performs a computation of the neighborhoods of each element, one dimension at a time, to find outliers. Aggregating such observations results in quantifying how much each data point can be seen as an outlier in the whole dataset, while being less prone to the curse of dimensionality.

2 Neighborhood-based outlier detection

Outliers in the dataset are found using their rareness. Let Ω a set of N elements, defined on D dimensions. Let $Knn^d(n)$ be the set of all K nearest neighbors of element n, on dimension d; and $Rank_e^d(n)$ be the rank of element n according to element e, on a given dimension d. To obtain $Rank_e^d(n)$, element e ranks all the other elements of the dataset, on dimension d, according to their proximity to e. For an element n, the closer to e it is, the lower its rank, with $Rank_e^d(n) = 2$ if n is the nearest neighbor of e.

The rareness of an element n defined by another element e, for a neighborhood size of K, on a dimension d, is the following :

$$Rareness_e^d(n) = \frac{1}{K} \cdot \min(Rank_e^d(n), K) \tag{1}$$

We can compute the rareness of n for a given neighborhood instead of a single element. The rareness for a neighborhood composed of the K nearest-neighbors of n on dimension d is the mean rareness $Rareness_e^d(n)$ for all elements $e \in Knn^d(n)$. It can be written as:

$$Rareness^d_{Knn^{d'}(n)}(n) = \frac{1}{K-1} \cdot \sum_{e \in Knn^{d'}(n)} Rareness_e^d(n) \tag{2}$$

Now that we can compute the rareness of an element for a given dimension and neighborhood, we define the rareness of an element n as the maximum rareness of n, for all dimension of D and for their K-neighborhood on each dimension of D:

$$Rareness(n) = \max_{\substack{d \in D \\ d' \in D}} \{Rareness^d_{Knn^{d'}(n)}(n)\} \tag{3}$$

Algorithm 1 illustrates the pseudo code used to compute $Rareness(n)$, the rareness of an element of the dataset. Algorithm 2 is the pseudo code illustrating $Rareness_B^d(n)$, the rareness of element n, for a dimension d and a neighborhood B.

Data: N elements defined on D dimensions, neighborhood size K
Result: list of outlier elements
$Knn^d(n) \longleftarrow$ *K-nearest-neighbors of n, on dimension d;*
foreach *element n in N* **do**
  foreach *dimension d in D* **do**
    $scores^d(n) \longleftarrow \max_{d' \in D}(\mathbf{Rareness}(n, d, Knn^{d'}(n), K))$;
  end
end
$Result \longleftarrow \{e \in N / \exists d \in D, scores^d(n) = 1\}$;

Algorithm 1: Outlier detection algorithm

Data: element n, dimension d, neighborhood $Knn^{d'}(n)$ *of size* K
Result: rareness of element n for a given dimension d and neighborhood $Knn^{d'}(n)$
$Rank_e^d(n) \longleftarrow$ ranking of n according to its distance from e on dimension d;
foreach *neighbor* e *in* $Knn^{d'}(n)$ **do**
  $scores_e^d(n) \longleftarrow \frac{1}{K} \times \min(Rank_e^d(n), K)$;
end
$Result \longleftarrow \frac{1}{K-1} \times \sum_{e \in Knn^{d'}(n)} score_e^d(n)$;

Algorithm 2: Rareness computing algorithm

To illustrate the algorithm described above, we simulate a simple dataset of 1000 elements described on 6 dimensions. The attributes of the first 998 items are randomly generated following a uniform and independent distribution for all dimensions. We introduce two outliers points, 999 and 1000: the former is outside the distribution on dimension 2 and outside if we consider both dimensions 3 and 4 simultaneously, and it fits the distribution on the three remaining dimensions. The latter point is outside the distribution if we consider dimensions 1 and 2 simultaneously, and dimensions 3 and 4 simultaneously. It fits the distribution of the two remaining dimensions.

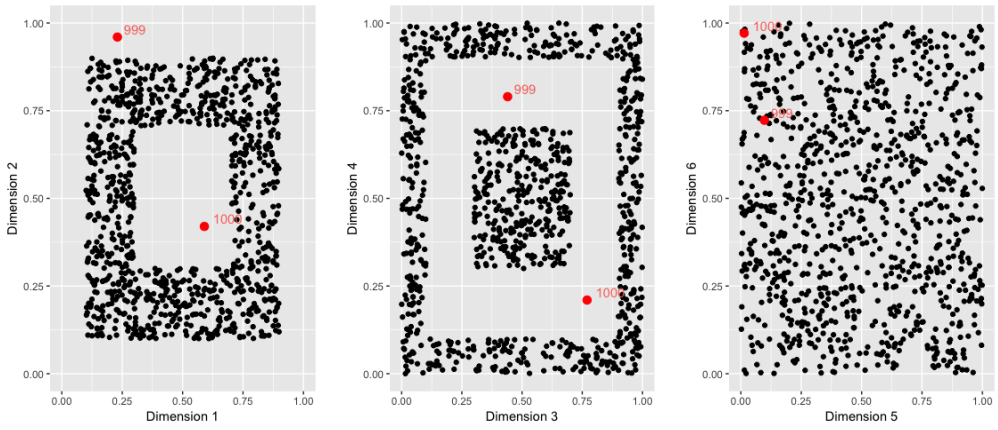

FIG. 1 – *1000 elements defined on 6 dimensions, including 2 outliers.*

Given this dataset, in order to find the outliers, we compute the rareness of each element. To choose the neighborhood size K, we iteratively decrease K from 1000 to 1 and keep the highest value that gives us two outliers. In this example, $K = 160$ is the first value returning two outliers: elements 999 and 1000. A good approximation for K is a quarter of the total number of elements. Using our example, with $K = 250$, the algorithm returns no outliers, but looking at the two items with the highest rareness we find elements 999 and 1000.

3 Conclusion and future works

We expressed the computation of rareness as an algorithm and detailed an example using synthetic data. This neighborhood-based approach as a few advantages such as retaining its meaningfulness even with hundreds of dimensions. To better fit the kind of data stored on each dimension, it allows different ranking methods for the elements, be it quantitative or qualitative. Using this approach, we obtained reliable results for synthetic datasets of up to ten thousand individuals and a thousand dimensions. It was also tested successfully on a real dataset consisting of electronic medical records of a thousand diabetic patients.

There is still a lot of possible leads to further this work. First of all, we are currently comparing this approach with other popular algorithms to adequately measure the relevance of this method. The significance of the results needs to be assessed with synthetic and real datasets and a variety of outlier data (outliers on one or multiple dimensions or with qualitative values). We also plan to thoroughly benchmark our solution and its time and space complexity. The algorithm is easy to parallelize, and the first experiments indicate linear performance gains: datasets up to thousands of individuals and dimensions are processed easily.

References

Aggarwal, C. C., A. Hinneburg, and D. A. Keim (2001). On the surprising behavior of distance metrics in high dimensional space. *International Conference on Database Theory 1973*(Chapter 27), 420–434.

Donoho, D. L. et al. (2000). High-dimensional data analysis: The curses and blessings of dimensionality. *AMS Math Challenges Lecture 1*, 32.

Falip, J., A. A. Younes, F. Blanchard, and M. Herbin (2017). Représentativité, généricité et singularité: augmentation de données pour l'exploration de dossiers médicaux. In *Atelier VIF: Visualisation d'informations, Interaction, et Fouille de données-EGC 2017*.

Herbin, M., D. Gillard, and L. Hussenet (2017). Concept of Observer to Detect Special Cases in a Multidimensional Dataset. *I4CS 717*(2), 34–44.

Résumé

La détection des données atypiques au sein d'un jeu de données est un problème aux applications nombreuses. On le retrouve notamment utilisé pour la détection de cas inhabituels en médecine, de fraudes dans le milieu bancaire, ou d'intrusion dans la sécurité réseau. La plupart de ces usages nécessite toutefois des données décrites par de nombreux attributs. Plus le nombre de dimensions augmente, plus la notion de proximité entre éléments perd de son sens : les données deviennent plus éparses et les éléments quasiment équidistants. De plus, l'hétérogénéité des données médicales les rend complexes à analyser. A cause de ces inconvénients, les algorithmes classiques pour ce problème perdent en efficacité. Ce papier présente un algorithme de détection des données atypiques adapté à la haute dimensionnalité. L'approche choisie repose sur l'analyse de voisinages sur des projections unidimensionnelles.