2018 IEEE Fourth International Conference on Big Data Computing Service and Applications

Knee/Elbow estimation based on first derivative threshold

Mário Antunes*, Diogo Gomes*, and Rui L. Aguiar*
*Instituto de Telecomunicações
Universidade de Aveiro
Aveiro, Portugal
Email: mario.antunes,dgomes,ruilaa@av.it.pt

***Abstract*—Estimating the knee/elbow point in error curves is a challenging task. However, most of the time these points represent ideal compromises or ideal parameters for several tasks, methods and algorithms. Our focus is determining the ideal number of clusters autonomously. In this paper, we formalize the notion of knee/elbow point, discuss known methods to determine it and propose our own method. Contrary to most methods, ours is resilient to long tails in the error curve. This behaviour is especially important when considering autonomous methods. The proposed method outperformed the competition on five datasets from UCI Machine Learning Repository.**

***Index Terms*—Knee, Clustering, Unsupervised Learning**

I. Introduction

One of the main difficulties for cluster analysis is to determine the correct number of clusters for different datasets. Most clustering algorithms are designed only to partition data objects according to a known number of clusters. As such, identifying the correct number of clusters is an important task for any clustering problem. On one hand, the common way to deal with the issue is to use domain knowledge over the underlying dataset. On the other hand, many statistical criteria and clustering validity indices have been investigated in order to automatically select an appropriate number of clusters. Especially important when considering complete autonomous systems and unsupervised learning methods.

Some methods rely on identifying the knee/elbow point in error curves and select it as the ideal number of clusters. In fact, the concept of knee/elbow point in error curves is used in many fields like fatigue damage theories [1]–[3], detecting the number of clusters [4], botnet detection [5], and system behaviour [6].

Our personal interest comes from clustering distributional profiles of words into categories in order to improve the accuracy of semantic similarity [7]. We developed an unsupervised method to learn a semantic model from public web services. The learning method uses clustering algorithms to identify word categories automatically (word category is closely related to concepts and the possible meaning of a word). Our prototype relied on k-means to cluster the distributional profile into categories. However, the algorithm has two main disadvantages: it is not deterministic and requires the number of clusters *a priori*. At the moment we rely on a statistical method [8] (similar to GAP statistics [9]) to determine the ideal number of clusters. Although the results were positive, we are interested in evaluating and researching other means to determine the ideal number of clusters. As such, our main contributions are: i) development of a new method to estimate the knee/elbow point in error curves, and ii) comparison of our method with other knee/elbow estimators.

The remainder of the paper is organized as follows. In Section II we formally define the concept of knee/elbow in error curves. An overview of knee/elbow detection algorithm is given in Section III. Our method is described in Section IV The results of our evaluation are given in Section V. Finally, discussion and conclusions are presented in Section VI.

II. Defining knee/elbow

The difficulty with defining a knee/elbow formally is that the "good enough" point in one situation may not be "good enough" in another. Thus, knee/elbow detection is an inherently heuristic process. However, to design knee/elbow detection algorithm, we require a consistent definition.

In previous works [6], [10], the authors used the mathematical definition of curvature for continuous functions as the basis for knee/elbow definition. For any continuous function $f(x)$ there exists a close-form $K_f(x)$ that defines the curvature of $f(x)$ at any given point:

$$K_f(x) = \frac{f''(x)}{(1 + f'(x)^2)^{\frac{3}{2}}} \tag{1}$$

The knee/elbow point is the point of maximum curvature, since curvature is a mathematical measure of how much a function differs from a straight line. In order to identify the maximum curvature, we can calculate the derivative of Equation 1 and find its roots. As depicted in Equation 2, it is important to mention that we discarded the denominator. The denominator is not relevant when finding roots because it cannot be equal to zero (otherwise the function tends to infinity). In order to find the roots of a fractional function we only use the numerator, the denominator is only relevant

0-7695-6396-1/18/$31.00 ©2018 IEEE
DOI 10.1109/BigDataService.2018.00042

to verify that the numerator roots are not also denominator roots (which makes the function tend to infinity).

$$K'_f(x) = f'''(x)(1 + f'(x)^2) - 3f''(x)^2 f'(x) = 0 \quad (2)$$

From the previous equations, we can derive two important pieces of information. First, inflection points (or even the first and second derivative by itself) are not sufficient criterion to reliable identify knee/elbow. In contrast, the curvature definition precisely matches the concept of a knee. Figure 1 depicts the function $f(x) = \frac{1}{x}$, and the respective curvature $K_f(x) = \frac{2}{(x^{-4}+1)^{3/2} \times x^3}$. Although $f(x)$ first and second derivatives do not have any roots, according to the $K_f(x)$ one is the point of maximum curvature.

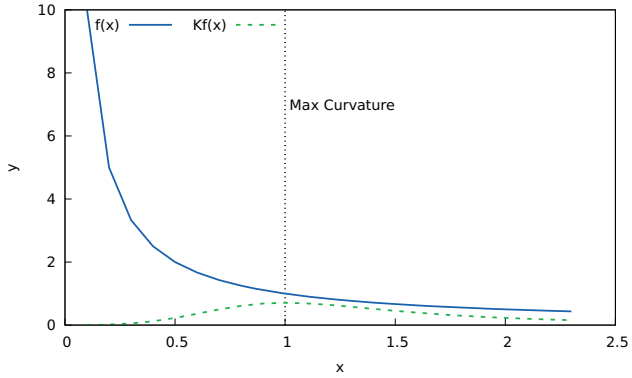

Fig. 1: Graphical representation of $f(x) = \frac{1}{x}$, the respective curvature $K_f(x) = \frac{2}{(x^{-4}+1)^{3/2} \times x^3}$ and the point with maximum curvature 1

Second, rescaling the function $f(x)$ alters the value of the knee/elbow point. A rescaling operation can be expressed as a multiplication and addition of constants: $af(x) + b$, where a and b are constants. Applying the previous results to the curvature equation we end up with Equation 3.

$$K_{af+b}(x) = \frac{af''(x)}{(1 + a^2 f'(x)^2)^{\frac{3}{2}}} \quad (3)$$

Again, in order to identify the maximum curvature, we can calculate the derivative of Equation 3 and find its roots. The Equation 4 is only similar to the original equation (Equation 4) when $a = 1$. In short, rescaling the original function $f(x)$ may alter the curvature of the function, and impact the detection of the knee/elbow point.

$$K'_f(x) = f'''(x)(1 + a^2 f'(x)^2) - 3a^2 f''(x)^2 f'(x) = 0 \quad (4)$$

While curvature is well-defined for continuous functions, it is not well-defined for discrete datasets. One possibility would be fitting a continuous function to the discrete dataset and using the Equation 2 to identify the point with the highest curvature. However, fitting a continuous function to a set of arbitrary data points is difficult (computational expensive, noisy data, missing values). As such, several researchers have developed methods to estimate the knee/elbow point without the close-form expressed in Equation 1.

III. Background and Related Work

Although not heavily researched, there are several approaches to detecting knee/elbow points in discrete data. In this section, we describe the most common approaches.

While curvature is not well-defined for discrete datasets, Menger curvature [6], [11] defines the curvature for three discrete points as the curvature of the circle circumscribed by those points. This is the only local criterion considered in this section. In other words, this method relies on three points to estimate the knee/elbow point without considering all the points in the function. As such, noisy data can lead to poor accuracy when estimating the knee/elbow point. This is the major drawback of this method.

Kneedle [6] uses the point further away from a line defined by the head and tails points of the error curve. Both coordinates of the original curve are rescaled to $[0, 1]$, in order to easily find the point with maximum curvature. It is important to notice that the method supports detection knee/elbow points on-line. The authors use this feature to detect relevant points in congestion control and network latency. Nonetheless, this method suffers from two drawbacks. First, if the tail of the error curve is long enough it may alter the knee/elbow point detection. Second, the method applies rescaling to the original error curve. As seen in Section II, rescaling operations may alter the curvature of the function. These drawbacks are most evident in the off-line scenarios.

L-method [10] fits two straight lines from the head to candidate point and from the candidate point to the tail of the curve. The candidate point that minimizes the root-mean-square error (RMSE) between the straight lines and the points of the curve is returned as the knee/elbow point. The resulting point represents the point with the sharper angle in the curve. A long tail in the error curve can also alter the value of the knee/elbow point with this method. To minimize this issue the authors proposed an iterative method that incrementally reduces the tail, while the knee/elbow point is refined. The major drawback of this method is the high computational cost necessary to fit the straight lines.

IV. Dynamic first derivative thresholding

As stated, our interest in knee/elbow point estimation is due to the need to autonomously identify the ideal number of clusters. In several clustering algorithms, the method to identify the ideal parameters is finding a compromise in an error curve (typically with a sharp curvature). K-means [12] uses a method properly named elbow method. DBSCAN [13] uses the elbow of a k-distance graph (plotting the average distance to the k nearest neighbours ordered from the largest to the smallest value [14]) as the ideal value for ϵ. Hierarchical clustering relies on knee/elbow detection to select the ideal cluster

from the dendrogram. Furthermore, as listed in Section I, several other areas use knee/elbow point estimation to identify optimal compromise or other concepts.

Initially, we tried to use Kneedle to estimate the ideal number of clusters. During our evaluation, we observed that the knee/elbow point is incorrectly estimated when the tail of the curve is long enough. Also, the rescaling used by Kneedle may alter the estimation of the knee/elbow point. L-method is resilient to these drawbacks but requires multiple iterations to estimate the correct point.

As such, we devised a method to estimate the knee/elbow point that is resilient to a long tail and does not rescale the initial error curve. Our method, named dynamic first derivative thresholding (DFDT) is a hybrid between Menger curvature and L-method. It estimates the first derivative of the curve, a local criterion similar to the Menger curvature. The first derivative of the curve represents the slope of the tangent line. Similar to the L-method, our method tries to identify the point where the function was a sharp angle. Instead of fitting two straight lines, we use a threshold algorithm to find the ideal slip between high and low values of the first derivative. Dividing the slopes into two classes, the head of the curve with high values and tail with low values, these type of algorithm separate data points in two different groups (high and low groups). These methods are used in image processing to convert a grey scale image into black and white. The algorithm used to estimate the ideal threshold was IsoData [15], due to its simplicity this algorithm can be adapted to several use cases. The point closer to the threshold is used as the knee/elbow estimation.

Algorithm 1 Dynamic First Derivative Threshold

```
1: function DFDT(x, y)
2:     m ← firstDerivative(x, y)
3:     t ← IsoData(m)
4:     rv ← 0
5:     minDist ← ||m[0] − t||
6:     for i ← 1; i < m.length; i++ do
7:         if ||m[0] − t|| < minDist then
8:             minDist ← ||m[0] − t||
9:             rv ← i
10:        end if
11:    end for
12:    return rv + 1
13: end function
```

The major advantage of this method is that the threshold algorithm slips data based on their value and not on quantities. As such, the effect of long tails is minimized. As stated, long tails may alter the value of the knee/elbow point estimation Another advantage of the proposed method is that it can be used to select a portion of the curve (n points before and after the threshold) to be further refined by other knee/elbow estimation strategies.

The code was developed in Java and is publicly available[1]. Furthermore, the other methods mentioned in Section III were also implemented and are also publicly available in the same library.

V. Performance evaluation

In order to evaluate our method, we selected five datasets from UCI Machine Learning Repository[2]. Previous authors [4] have evaluated similar methods with the same datasets. The five selected datasets were: Iris, Yeast, Control, Wine and Vehicle. **Iris** contains three classes of 50 instances each, where each class refers to a type of iris plant. **Yeast** is originally used for protein localization sites prediction. The class distribution from a rule-based expert system indicates the optimum number of clusters as 10. However, as the size of six clusters among them is too small, we considered 5 as the optimal number of clusters. **Control** contains 600 examples of control charts synthetically generated. There are six different classes of control charts. **Wine** contains 178 data points that are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. All data points are composed of 13 attributes, divided into 3 classes. Finally, **Vehicle** contains 946 data points that represent vehicle silhouettes. All data points are composed of 18 attributes, divided into 4 classes.

We clustered the five datasets with K-means algorithm and used four knee/elbow estimation methods to identify the ideal number of clusters. One is the method proposed in this paper, the other three are Menger curvature, Kneedle and L-Method. The knee/elbow estimation methods were also compared with the average silhouette method. **Average silhouette method**, this approach measures the quality of the clustering method, it determines how well each object lies within its cluster. It uses a metric named silhouette [16], a high average silhouette width indicates a good clustering. Average silhouette method computes the average silhouette for different values of k. The optimal number of clusters k is the one that maximizes the average silhouette.

All the datasets were clustered twice, with different ranges of minimum and maximum allowed number of clusters ($[0, 20]$ and $[0, 40]$ respectively). This provides insight on how long tails alter the estimation of the knee/elbow point. In order to evaluate the performance of each method, we computed the root-mean-square error between the correct number of cluster and the estimated.

Table I and Table II contains the result of our evaluation. When the clustering range was $[0, 20]$ all the method performed well, except for Menger curvature. This is expected since the method only relies on a local criterion. However, when the clustering range was increased to $[2, 40]$ all the methods, except for DFDT, were penalized. As

[1]https://github.com/mariolpantunes/ml
[2]https://archive.ics.uci.edu/ml/index.php

discussed in this paper, long tails may alter the value of the knee/elbow point, leading to poor estimation. The proposed method was resilient to this factor.

TABLE I: Results of the evaluation when clustering with range $[0, 20]$.

Dataset/ Method	Iris 3	Yeast 5	Control 6	Wine 4	Vehicle 4	RMSE
Menger	12	18	16	15	14	10.9
Kneedle	6	6	7	7	6	2.5
L-method	6	6	7	5	5	1.8
DFDT	4	5	5	5	4	**1.1**
Silhouette	2	2	7	3	2	1.7

TABLE II: Results of the evaluation when clustering with range $[0, 40]$.

Dataset/ Method	Iris 3	Yeast 5	Control 6	Wine 4	Vehicle 4	RMSE
Menger	11	17	25	11	28	15.5
Kneedle	7	9	9	9	7	4.1
L-method	7	8	8	8	6	3.4
DFDT	4	5	5	5	4	**1.1**
Silhouette	2	3	3	2	2	1.9

VI. Discussion and Conclusions

Estimating the knee/elbow point in error curves is a challenging task. However, most of the time these points represent ideal compromises or ideal parameters for several tasks, methods and algorithm. Our focus is determining the ideal number of clusters autonomously. A requirement of our semantic model [7].

In this paper, we formalized the concept of knee/elbow point based on the notion of curvature and showed that rescaling the original function may impact the curvature (as a consequence, it may alter the value of the knee/elbow point). Several methods to estimate the knee/elbow point were analysed, and we proposed our own method.

The evaluation showed that the proposed method outperformed the competition for the specific datasets. As such, the proposed method shows promise and should be researched further. Possible enhancements of the method are to use other threshold algorithms and consider the second derivative.

In order to identify the limitation of the proposed method, a more thorough evaluation is required. Considering other clustering algorithms, synthetic datasets and applications. Nonetheless, the proposed method works well on real data and will be explored in future publications.

Acknowledgement

The present study was developed in the scope of the **Smart Green Homes** Project [POCI-01-0247-FEDER-007678], a co-promotion between **Bosch Termotecnologia S.A.** and the **University of Aveiro**. It is financed by Portugal 2020 under the Competitiveness and Internationalization Operational Program, and by the European Regional Development Fund. This work was also partially supported by research grant SFRH/BD/94270/2013.

References

[1] K. Endo and H. Goto, "Initiation and propagation of fretting fatigue cracks," *Wear*, vol. 38, no. 2, pp. 311–324, jul 1976.

[2] A. Fatemi and L. Yang, "Cumulative fatigue damage and life prediction theories: a survey of the state of the art for homogeneous materials," *International Journal of Fatigue*, vol. 20, no. 1, pp. 9–34, jan 1998.

[3] L. Franke, "A non-linear fatigue damage rule with an exponent based on a crack growth boundary condition," *International Journal of Fatigue*, vol. 21, no. 8, pp. 761–767, sep 1999.

[4] Q. Zhao, V. Hautamaki, and P. Fränti, "Knee point detection in BIC for detecting the number of clusters," in *Advanced Concepts for Intelligent Vision Systems*. Springer Berlin Heidelberg, 2008, pp. 664–673.

[5] A. Karasaridis, B. Rexroad, and D. Hoeflin, "Wide-scale botnet detection and characterization," in *Proceedings of the First Conference on First Workshop on Hot Topics in Understanding Botnets*, ser. HotBots'07. Berkeley, CA, USA: USENIX Association, 2007, pp. 7–7. [Online]. Available: http://dl.acm.org/citation.cfm?id=1323128.1323135

[6] V. Satopaa, J. Albrecht, D. Irwin, and B. Raghavan, "Finding a "kneedle" in a haystack: Detecting knee points in system behavior," in *2011 31st International Conference on Distributed Computing Systems Workshops*. IEEE, jun 2011.

[7] M. Antunes, D. Gomes, and R. L. Aguiar, "Towards IoT data classification through semantic features," *Future Generation Computer Systems*, dec 2017.

[8] D. T. Pham, S. S. Dimov, and C. D. Nguyen, "Selection of k in k-means clustering," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 219, no. 1, pp. 103–119, 2005.

[9] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 2, pp. 411–423, 2001.

[10] S. Salvador and P. Chan, "Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms," in *16th IEEE International Conference on Tools with Artificial Intelligence*. IEEE Comput. Soc, 2004.

[11] X. Tolsa, "Principal values for the cauchy integral and rectifiability," *Proceedings of the American Mathematical Society*, vol. 128, no. 07, pp. 2111–2120, jul 2000.

[12] S. Lloyd, "Least squares quantization in pcm," *Information Theory, IEEE Transactions on*, vol. 28, no. 2, pp. 129–137, 1982.

[13] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD'96. AAAI Press, 1996, pp. 226–231. [Online]. Available: http://dl.acm.org/citation.cfm?id=3001460.3001507

[14] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "DBSCAN revisited, revisited," *ACM Transactions on Database Systems*, vol. 42, no. 3, pp. 1–21, jul 2017.

[15] T. Ridler and S. Calvard, "Picture thresholding using an iterative selection method," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 8, no. 8, pp. 630–632, 1978.

[16] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, nov 1987.