Term Paper Data Science 1

Docent: Prof. Dr. Lena Wiese
Semester: Summer Term 2021

Institute of Computer Science
Goethe-Universität Frankfurt a. M.

Authors: Franziska Hicking
your Student ID
your.email@ddre.ss
branch of study (Bachelor/Master, semester count)
Jonas Elpelt
your Student ID
your.email@ddre.ss
branch of study (Bachelor/Master, semester count)
Julian Rummel
6673334
s9594673@stud.uni-frankfurt.de
Master Bioinformatics, 2
Niklas Conen
6599913
conen@stud.uni-frankfurt.de
branch of study (Bachelor Computer Science, 8)

Date: May 17, 2021

Chosen Project Topic:
T4 - DISTANCE MEASURES AND CLUSTERING

Abstract

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet,

Contents

Some Latex-specific hints:

- a

1 Definition of Distance Measure

ANMERKUNGEN:

- What general problem is addressed?
- What is the general methodology that is used?

A distance measure is a function $d(x, y)$ that calculates a real value between two points in a space, containing two sets of points. This function must satisfy the four following axioms:

1. No negative distances:
   $d(x, y) \geq 0$
2. Identity of indiscernibles:
   $d(x, y) = 0$, iff $x = y$
3. Symmetry:
   $d(x, y) = d(y, x)$
4. Triangle inequality:
   $d(x, y) \leq d(x, z) + d(z, y)$

The triangle-inequality impose the condition that a distance reflects the shortest path between two points. Thus, it is not possible to achieve a distance improvement by traveling via an intermediate point z. [1]

2 Different Distance Measurements

ANMERKUNGEN:

- What specific problem is addressed?
- What is the specific methodology that is used?
- What improvement is shown?

2.1 Euclidean Distances

The Euclidean distance calculates the distance of two points, represented as vectors of n real numbers, in an n-dimensional euclidean space. In general, a Euclidean distance measure d is called L_r-*norm*, for arbitrary constants r.

$$d([x_1, x_2, ..., x_n], [y_1, y_2, ..., y_n]) = \sum_{i=1}^{n} (|x_i - y_i|^r)^{\frac{1}{r}}$$

The typical euclidean norm refers to the L_2-*norm* and is calculated as the positive square root of the sum of all squared distances in each dimension.

$$d([x_1, x_2, ..., x_n], [y_1, y_2, ..., y_n]) = \sqrt{\sum_{i=1}^{n} (|x_i - y_i|)^2}$$

It is simple to verify three of the aforementioned axioms (1).

1. No negative distances:
   The nonnegative values are given by the positive square root. let $x \neq y$, then the square of any real number is always positive.

2. Identity of indiscernibles:
   For $x = y$ the value is obviously 0. Let $x = y$, then $(|x - y|)^2 = 0$ and $\sqrt{0} = 0$.

3. Symmetry:
   Symmetrie is cleary given by the square of each distance.
   $(x - y)^2 = (y - x)^2$.

4. Triangle inequality:
   As a matter of fact, this axiom requires a more difficult proof. However, to keep it simple, the Euclidean space possesses the property that the sum of the lengths of Cathetus and Ancathetus is always longer than the length of the Hypothenuse. [1]

3 Data Set Description

ANMERKUNGEN:

- What benchmark data sets are used?

3.1 Solar Flare

The solar flare dataset is taken from the UCI Machine Learning Repository [2]. Each point contains data recorded for on active region of the sun. The first three attributes are the McIntosh classification of sunspot groups:

1. Z-value: modified Zurich sunspot class, $\{A, B, C, D, E, F, H\}$
2. p-value: description of the penumbra of the largest spot. A penumbra is a part of a sunspot that is darker than the suns surface. $\{x, r, s, a, h, k\}$
3. c-value: description of the distribution of sunspots in a group $\{x, o, i, c\}$

A detailed descriptions of the letter codes can be found in the original paper by McIntosh [3]. The following entries are as follows:

4. Activity (1: reduced, 2: unchanged)
5. Evolution (1: decay, 2: no growth, 3: growth)
6. Code for the previous 24h flare activity
7. Is the region historically complex? (1: yes, 2: no)
8. Todoooooo
9. Area (1: small, 2: large)
10. Area of the largest spot (1: ≥ 5 deg^2, 2: > 5 deg^2)

The last three entries are a predicted flare classes

4 Clustering Algorithms

4.1 K-Means

4.2 K-Medoids

4.3 K-Median

4.4 DBSCAN

DBSCAN was developed by Martin Ester, Hans-Peter Kriegel, Jiirg Sander and Xiaowei Xu. All following definitions and descriptions are taken from

their original publication [4] or their revisit of DBSCAN [5].
Contrary to the aforementioned centroid-based partitioning algorithms (k-means, k-medoids and k-median) the DBSCAN (*Density Based Spatial Clustering of Applications with Noise*) algorithm uses point densities to determine clusters.
To introduce the definition of the density of a cluster, first the Eps-neighbourhood of a point is defined:

Definition 1: *Eps-neighbourhood*
A point q is part of the Eps-neighbourhood N_{Eps} of point p if the distance between them is smaller than a threshold distance called Eps.
The Eps-neighbourhood therefore is defined as $N_{Eps} = \{q \in D \mid ||p, q|| \leq Eps\}$ with D denoting the entirety of points that are supposed to be clustered and $||p, q||$ being the distance between p and q for an arbitrary distance measure.

The Eps-neighbourhood fails at being a reliable measure for the point density if a point is located at the border of a cluster. These points are called *border point*. Points that are located on the inside of a cluster are called *core points*. Hence the following definition is made:

Definition 2: *directly density-reachable and density-reachable*
A point p is directly density-reachable from a point q when

1. $p \in N_{Eps}(q)$
2. $|N_{Eps}(q)| \geq$ MinPts

with MinPts being the minimal number of points that $N_{eps}(q)$ should contain so that q is considered a core point of a cluster.
A point is *density-reachable* if there is a chain of points between p and q so that all neighbouring points in the chain are directly density reachable.

To complete the definition of what is considered part of a cluster density-connectivity is defined:

Definition 3: *density-connected*
Two points p and q are considered density-connected if there is a common point o which is density-reachable from p and q.

Now a cluster can be described as:

Definition 4: *cluster*
A cluster is a non empty subset $C \in D$ so that:

1. $\forall p, q : p \in C \wedge q$ is density reachable from $p \Rightarrow q \in C$
2. $\forall p, q \in C :$ p is density-connected to q

Noise is easily defined as every point that is not part of a Cluster C_i.

Using these definitions DBSCAN can begin the clustering process with given values for Eps and MinPts. In the beginning all points are not labeled. Beginning with an arbitrary point p all points are iterated in a linear fashion. For each point a `RangeQuery` function is executed finding all density-reachable neighbours of p. If `RangeQuery` finds more than MinPts neighbours then p is a core point and is labeled as such. Otherwise p is marked as Noise.
In the next step every point in the Neighbourhood excluding p is expanded. Unlabled Points get checked for the core point condition (which equals a `RangeQuery` call). Points that got labeled as Noise before are labeled as core points. When the expansion comes to an end a cluster is yielded and the next unlabeled point is chosen as p.

Two clusters may be merged if their distance is below Eps. The distance between two clusters C_1 and C_2 is defined as $||C_1, C_2|| = \min\{||q, p|| \mid p \in C_1, q \in C_2\}$.

The runtime complexity of DBSCAN heavily depends on the runtime of the `RangeQuery` function and the distance measure. Thus the runtime can exceed $\mathcal{O}(n^2)$ depending on the chosen implementations. A detailed discussion of DBSCANs runtime can be found in [5].

5 Description of Python libraries used

ANMERKUNGEN:
Libraries:

- pyclustering
- sklearn

6 Description of Evaluation Module

ANMERKUNGEN:

- What are the results and how are they measured?

7 Web Frontend and User Manual

ANMERKUNGEN:

- Describe the implementation and write a brief user manual with screenshots.

8 Conclusion

ANMERKUNGEN:

- Summarize the main points and achievements
- Add your own assessment/criticism on the topic

References

[1] Jure Leskovec, Anand Rajaraman, and Jeffrey David Ullman. Definition of a distance measure. In *Mining of Massive Datasets - THIRD EDITION*, page 97. Infolab Stanford EDU, 2020.

[2] Dheeru Dua and Casey Graff. UCI Machine Learning Repository, 2017.

[3] Patrick S. McIntosh. The classification of sunspot groups. *Solar Physics*, 125(2):251–267, Sep 1990.

[4] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 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*, KDD'96, page 226–231. AAAI Press, 1996.

[5] Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Trans. Database Syst.*, 42(3), July 2017.

[6] Andrei Novikov. PyClustering: Data Mining Library. *Journal of Open Source Software*, 4(36):1230, apr 2019.

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.