CS-E4650 Methods of Data mining

Exercise 1 / Autumn 2022

1.1 Cows with numerical and categorical features

Learning goals: How to use distance/similarity measures when there are both numerical and categorical features in data; similarity graphs.

Look at the cow data in Table 1. The task is to evaluate distances between cows. Note that field 'name' is the cow identifier and not used in any distance calculations. You can calculate distances manually or make scripts, but **implement the distance measures yourself** (do not use ready-made library functions). You can use library functions for min, max, mean, and standard deviation, if you want. Remember to report intermediate steps and prepare to show your code and its outputs.

Table 1: Cow data: name, race, age (years), daily milk yield (litres/day), character and music taste.

name	race	age	milk	character	music
Clover	Holstein	2	10	calm	rock
Sunny	Ayrshire	2	15	lively	country
Rose	Holstein	5	20	calm	classical
Daisy	Ayrshire	4	25	kind	rock
Strawberry	Finncattle	7	35	calm	classical
Molly	Ayrshire	8	45	kind	country

a) In this part, use only numerical features. Scale the features with the min-max scaling described in the book (Aggarwal section 2.3.3) and calculate pairwise Euclidean distances (L_2 norm) between cows. Present the results as a nearest neighbour graph (as described in Lecture 1 and Aggarwal Sec. 2.2.2.9). Select the threshold ϵ as small as possible still keeping the graph connected.

b) In this part, use only categorical features. First, define Goodall distance measure d_G from the Goodall similarity measure G with $d_G = 1 - G$. The Goodall similarity measure is presented in Aggarwal sec. 3.2.2 and the slides of lecture 2 (use that version, since there are many alternative Goodall measures). Then calculate pairwise Goodall distances and

present the results as a nearest neighbour graph, once again selecting minimal ϵ such that the graph remains connected.

c) In this part, use both numerical and categorical features. Create a distance measure that combines the previous distance measures (L_2 and d_G) using Equation 3.9 in the book (Aggarwal sec. 3.2.3). (Note that Aggarwal gives similarity measure, but you can combine distance measures in the same manner.) Set λ as the proportion of numerical features. It is recommended to use the unbiased estimate of standard deviation. Create now a nearest neighbour graph using the combined measure and select minimal ϵ that keeps the graph connected.

d) Compare the results. Is the combined measure graph (c) more similar to the numerical (a) or categorical (b) measure graph? Can you explain why?

1.2 Similarity in social media profiles

Learning goal: To study distance functions and metrics for set form data.

Consider a social network where each user is associated with a set of labels that best describe a set of properties of the user. We define the profile of the user to be the set of associated labels, i.e., given a set $P = \{p_1, \ldots, p_n\}$ of user profiles and a universe of labels $L = \{l_1, \ldots, l_m\}$, each profile $p_i \in P$ is a set of labels $L_i \subseteq L$. The task is to design functions to measure the distance between two labels and similarity between two user profiles.

a) Propose a distance measure between labels, more precisely, given any two labels $l_1, l_2 \in L$, present a distance function d such that $d(l_1, l_2)$ returns a distance measure between labels l_1 and l_2. The distance function should be (i) intuitive and (ii) satisfy the metric properties (see next parts).

b) Discuss the intuition, strengths, and limitations of your measure.

c) Prove that your distance function is a metric. Depending on your measure, this can be tricky, but study at least the easy properties!

d) Now we want to compare the similarity of two user profiles. Propose an appropriate function $s(p_1, p_2)$ to compute the similarity of any two profiles $p_1, p_2 \in P$ and discuss its intuition.

e) Be prepared to show code that implements d and s and demonstrate its behavior with a small set of toy data.

1.3 Lower bounding a distance

Learning goal: To consider effective implementations of nearest neighbor search.

Consider the *nearest neighbor search problem*: Given a dataset of n objects $X = \{x_1, \ldots, x_n\}$ and a query object q, we want to find the object $x^* \in X$ that minimizes the distance $d(q, x)$, that is,

$$d(q, x^*) \leq d(q, x) \;\; \forall x \in X \; . \tag{1}$$

Assume that computing the distance function d is *very expensive*. Assume now that we are able to define another distance function d_{LB}, which is a *lower bound* of distance d. This means that for all pairs of objects x and y it should be

$$d_{\mathrm{LB}}(x, y) \leq d(x, y) \; . \tag{2}$$

Furthermore, assume that computing d_{LB} is *significantly more efficient* than computing d.

a) Write pseudocode of an algorithm for the nearest neighbor search using distance d.

b) Explain how to use the lower-bound distance d_{LB} to speed up the search algorithm of the previous part. Write pseudocode for the modified search algorithm.

c) What is a desirable property for the lower-bound distance d_{LB} to be as effective as possible for the modified algorithm? Explain why.

1.4 Homework: This will be added later

This task is homework that is done in groups of 2–3 students. Note that you cannot do the task alone or in a larger group, so it is recommended to search a group now. The TAs can help to find collaborators.