Exercise Sheet 4

Data Mining I

Submit your solutions until Monday, 04.04.2016, noon by uploading them to ILIAS. Later submissions won't be considered. Every solution should contain the ***name(s)***, ***email adress(es)*** *and* ***registration number(s)*** *of its (co-)editor(s).*

1 Supervised vs Unsupervised Learning (5 Points)

The following questions deal with properties of input data for supervised and unsupervised learning:

- If we want to use supervised learning, what property is mandatory for the input data?
- What do you think is generally cheaper: Creating a data set for supervised learning or for unsupervised learning?
- Do you expect unsupervised or supervised learning to produce better results? Why?

2 Agglomerative Hierarchical Clustering (14 Points)

Figure 1 shows data points in a coordinate system that shall be divided into clusters with the help of agglomerative clustering. The distance between data points $\vec{x} = (x_1, \ldots, x_n)$ and $\vec{y} = (y_1, \ldots, y_n)$ can be computed with the Euclidean Distance:

$$distance(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

At the beginning every data point is considered to be a singleton cluster.

2.1 Single Link (4 Points)

Use single link similarity to cluster the data points:

$sim(c_i, c_j) = \min_{\vec{x} \in c_i, \vec{y} \in c_j} distance(\vec{x}, \vec{y})$

The minimum distance is identical to the maximum similarity. **A graphical solution is sufficient**. Depict the solution with help of a *cluster dendogram*.

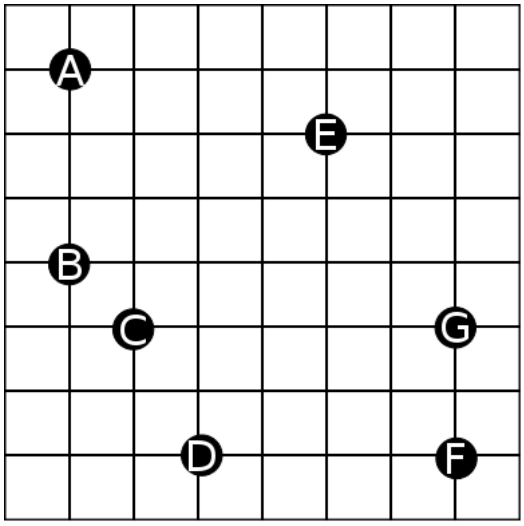

Figure 1: Data to be clustered with single link clustering

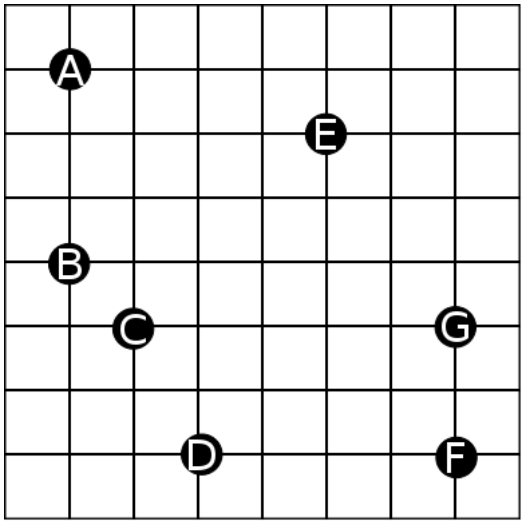

Figure 2: Data to be clustered with complete link clustering

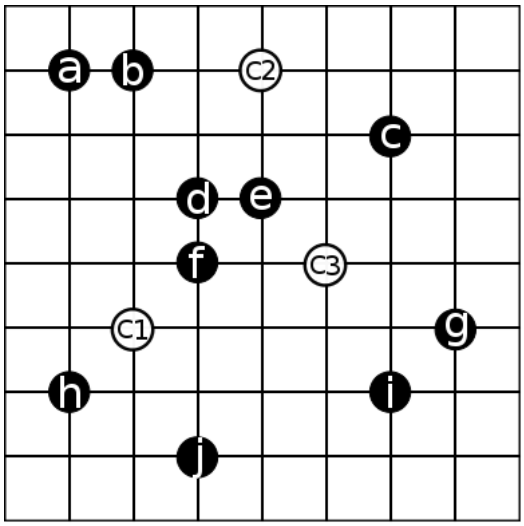

Figure 3: Data for K-Means Algorithm

2.2 Complete Link (10 Points)

Use complete link similarity this time for the clustering:

$sim(c_i, c_j) = \max_{\vec{x} \in c_i, \vec{y} \in c_j} distance(\vec{x}, \vec{y})$

Show your computations of cluster similarities in detail. Depict the solution with help of a *cluster dendogram*.

3 K-Means Clustering (20 Points)

Consider the following 10 data points in a 2-dimensional feature space:

a(1,7) ; b(2,7) ; c(6,6); d(3,5); e(4,5); f(3,4); g(7,3); h(1,2); i(6,2); j(3,1)

Your task is the execution of the K-Means algorithm with $k = 3$ based on these data points. Again, assume the *Euclidean Distance* as the distance measure. The initial centroids of the three clusters are:

centroid1(2,3); centroid2(4,7); centroid3(5,4)

A visualization of the data points and centroids can be seen in Figure 3.

Round the centroids to integer positions. For instance, if the new position $centroid1_{new}$ would be found at coordinates 2.5 and 5.3, assign $centroid1_{rounded}(3, 5)$ to be the new position of the centroid. If a point has the same minimum distance to more than one centroid you can assign the point randomly to one of the centroid candidates.

You can stop the calculation if none of the centroids moved in two succeeding iterations. Your result includes 3 stable cluster centroids and the corresponding assignments of the data points

to the clusters.

Write down every step of your calculation for the first iteration only. For other iterations just provide the 3 rounded centroids and the current cluster members.

4 Evaluation (6 Points)

Table 1 shows the gold standard for the K-Means clustering task:

Cluster I	Cluster II	Cluster III	Cluster IV
a	d	h	c
b	e	j	g
	f		i

Table 1: Gold Standard

Compute the purity score of your K-Means clustering results of exercise 3

As a reminder, the purity is calculated with help of this formula:

$$Purity(k_i) = \frac{1}{n_i} \times \max\{n_{ij} \mid c_j \in C\}$$

$C = \{c_1, \ldots, c_q\}$ classes in gold standard
$K = \{k_1, \ldots, k_q\}$ clusters generated by clustering algorithm
$n_i =$ size of cluster k_i
$n_{ij} =$ number of instances in k_i belonging to c_j

5 Clustering with RapidMiner (15 Bonus Points)

RapidMiner is a framework that provides a user interface to create processing pipelines using all sorts of machine learning algorithms and data processinig libraries. Download and install it from http://www.rapid-i.com.

Besides the videos on the RapidMiner website there is a useful playlist on youtube with a tutorial on text analytics with RapidMiner. The first part starts here: https://www.youtube.com/watch?v=hpvda_Rfg3s&list=PL7669FFBBA1825900. Watch these videos to learn how to work with RapidMiner, the following tasks assume that you have seen them.

We will work with a text corpus containing travel information that you can download from ILIAS (lp.zip). In this .zip file there is *corpus.zip* file. Unpack it to have a directory containing the relevant travel-related texts.

5.1 Creating a repository from Files (10 Bonus Points)

The first task is to load a part of the corpus into RapidMiner and start some preprocessing. **Submit the XML description of your process as a solution!** It is available in a tab per default and can be found under "view=>show view=>XML" otherwise.

a) Process Documents from Specific Files

First, use the *Process Documents from Files*-operator to load the corpus. We are only interested in the files describing activities, thus you can use the file pattern "*_activities*".

Double clicking the *Process Documents from Files*-operator lets you specify a preprocessing pipeline for each document. Apply the following operators in the given order:

- Tokenize (mode = non letters)
- Transform tokens to lowercase
- Filter stopwords
- Filter tokens by length (min = 2; max = 100)
- Apply stemming (Snowball stemmer)

b) Store results in a Repository

Return to the overview of the process (go up one level) and store the results in a repository by adding a *store* operator.

5.2 Clustering (5 Bonus Points)

Load the repository you just created in the previous task and apply the *k-means*-operator with k=3 to the data. You can find it in the operators under "Modeling=>Clustering and Segmentation".

Analyse the results by sorting each cluster in the "centroid table" view.

- Report the top 10 words of each cluster
- Think of a headline for each cluster based on the associations you have with the top words[1]

[1] For example if the top words are "spa, massage, sleep" then a headline could be "regeneration" or "relaxing"