

**Knowledge Discovery in Databases II**  
SS 2019

**Exercise 1: High dimensional data introduction**

**Exercise 1-1 High dimensional data analysis**

- (a) The zip file *ArffGen.zip* on the website provides a list of datasets in *.arff* format with varies dimensionality. You can simply open it using a text editor. Each data point is aligned with  $d$  real attributes and an integer class label.

For each point calculate the ratio “farthest-neighbor-distance”/“nearest-neighbor-distance” by using the Euclidean distance and calculate the average ratio for all objects (of the same dataset). Plot the average ratio for the sequence of datasets with increasing dimensionality. What conclusions can be drawn from this result with respect to the empty space problem/curse of dimensionality? Do you get the same results when using the Manhattan-Distance or the Maximum-Metric instead of the Euclidean distance?

- (b) Use datasets in the previous task. Let us assume the data space is partitioned into a regular grid with 4 partitions per dimension. For each dataset, generate a histogram (bar chart) that counts the number of cells containing 1 object, 2 objects, 3 objects,  $\dots$ , 250 objects. How do the histograms change with increasing dimensionality of the data? What are your observations? Plot exemplarily the histograms for different dimensions  $d$  above.

- (c) Let  $U_d$  be a  $d$ -dimensional hypersphere with the radius 1 and the volume  $V_d$ . Calculate the radius  $r_d$  of the  $d$ -dimensional hypersphere  $X_d$  that comprises double the volume (i.e.  $V_{new} = 2V_d$ ). Provide a closed-form expression for  $r_d$ , give the limit of the function for  $d \rightarrow \infty$ , and plot the values of  $r_d$  in the range  $d \in [1 \dots 50]$ .

What conclusions can be drawn from these results with respect to the empty space problem/curse of dimensionality?

(Hint: you can use any programming language to play with the data.)

### Exercise 1-2 Subspace Clusters

Dataset *SubspaceData.csv* contains 3 dimensional data points. There are three clusters hidden in the subspace. Try algorithms you learned from KDD 1, such as the kmeans or EM algorithm, to find those clusters. If those algorithms do not work, what can we do? How should we start to look for clusters in subspaces?

