ML: Algo & Theory
SS 18

1	2	3	4	$\sum$

Florence Lopez (3878792),
florence.lopez@student.uni-tuebingen.de
Jennifer Them (3837649),
jennifer.them@student.uni-tuebingen.de

Assignment 8

(Abgabe am 26. Juni 2018)

Exercise 1

Exercise 2

Exercise 3

Exercise 4

a.) The following equation has to be shown: $\sum_{i \in C_k} \|X_i - m_k\|^2 \leq \sum_{i \in C_k} \|X_i - X_j\|^2, \forall j \in C_k$. In one of the latter exercises, we have already proven that a function with the form $w \to \|Y - Xw\|^2$ is convex. By setting $X = Id$, the function of the form $w \to \|Y - w\|$ is convex, too. Additionally, the sum of a two convex functions, is convex too. Since our function $\sum_{i \in C_k} \|X_i - m_k\|^2$ is a sum of convex functions, it is convex, too. Therefore we can derive it and set it to 0:
$\frac{\delta}{\delta x_l} \sum_{i \in C_k} \|X_i - x\|^2 = \frac{\delta}{\delta x_l} \sum_{i \in C_k} \sum_{j=0}^{d} (X_{ij} - x_j)^2 = \frac{\delta}{\delta x_l} \sum_{i \in C_k} \sum_{j=0}^{d} (X_{ij}^2 - 2X_{ij}x_j + x_j^2)^2 = \sum_{i \in C_k} 2x_l - 2X_{il} = 2|C_k|x_l - 2\sum_{i \in C_k} X_{il}$.
Set this to 0, with: $\sum_{i \in C_k} 2x_l - 2X_{il} = 2|C_k|x_l - 2\sum_{i \in C_k} X_{il} = 0 \Leftrightarrow 2|C_k|x_l = 2\sum_{i \in C_k} X_{il}X_{il} \Leftrightarrow x_l = \frac{\sum_{i \in C_k} X_{il}}{|C_k|} = \overline{X_{il}}$. This means that the cluster center in each dimension l for all data points is the best solution.

b.) The k-means algorithm computes the new cluster centers for every step. Therefore, with the results from (a) we get an optimal solution L_1 with respect to the given partition of the data. In the next step the algorithm calculates a new solution L_2, which represents a new partition. Given these two solutions, we have two different cases that can occur:

1. After getting two solutions L_1 and L_2 in two following time steps, the partitions remain the same. This means that our current cluster centers are optimal and the partition does not change anymore, it converges. Therefore the algorithm terminates here.
2. After getting two solutions L_1 and L_2 in two following time steps, the partitions are not the same. This means that the second solution is more optimal than the first solution, leading to $L_2 \leq L_1$. This brings again two cases, which need to be differed:

1. $L_2 < L_1$, meaning L_2 is truly smaller than L_1. In this case solutions from any later time step will always be smaller than L_2, meaning that the partition from L_1 will not occur again, since we already found a better partition with L_2.
2. $L_2 = L_1$, meaning the solutions of both time steps are equivalent, which leads to both partitions being equivalent to each other. In this case, it could be that the algorithm oscillates between two partitions, which would mean that it diverges. This problem could be solved by adjusting the algorithm in such a way that it always picks the old solution, if the new solution is equal to it. In this case the algorithm would converge and therefore terminate.