SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

01.112 Machine Learning, Spring 2018
Homework 2

Sample Solutions

Question 1. Assume a function $f : R^n \to R$ is continuous. As discussed in class, if it is a convex function, then it satisfies the following property:

$$\textit{Property A:} \qquad f\left(\frac{x_1 + x_2}{2}\right) \leq \frac{f(x_1) + f(x_2)}{2}$$

for any $x_1, x_2 \in R^n$.

Show that the following statements are true (**with formal proofs**):

(a) If two functions $f(x)$ and $g(x)$ both satisfy Property A, then the following function also satisfies Property A:

$$h(x) = f(x) + g(x)$$

(5 points)

(b) If two functions $f(x)$ and $g(x)$ both satisfy Property A, then the following function also satisfies Property A:

$$h(x) = \max(f(x), g(x))$$

(5 points)

Answer

(a)

$$\begin{aligned} h\Big(\frac{x_1 + x_2}{2}\Big) &= f\Big(\frac{x_1 + x_2}{2}\Big) + g\Big(\frac{x_1 + x_2}{2}\Big) \\ &\leq \frac{f(x_1) + f(x_2)}{2} + \frac{g(x_1) + g(x_2)}{2} \\ &= \frac{f(x_1) + g(x_1)}{2} + \frac{f(x_2) + g(x_2)}{2} \\ &= \frac{h(x_1) + h(x_2)}{2} \end{aligned}$$

$h(x)$ satisfies *Property A* Proved.

(b)

$$h\Big(\frac{x_1+x_2}{2}\Big) = \max\Big(f\Big(\frac{x_1+x_2}{2}\Big), g\Big(\frac{x_1+x_2}{2}\Big)\Big)$$
$$\leq \max\Big(\frac{f(x_1)+f(x_2)}{2}, \frac{g(x_1)+g(x_2)}{2}\Big)$$

We have:

$$\begin{aligned}\frac{f(x_1)+f(x_2)}{2} &\leq \max(\frac{f(x_1)}{2}, \frac{g(x_1)}{2}) + \frac{f(x_2)}{2} \\ &\leq \max(\frac{f(x_1)}{2}, \frac{g(x_1)}{2}) + \max(\frac{f(x_2)}{2}, \frac{g(x_2)}{2}) \\ &= \frac{1}{2}\max(f(x_1), g(x_1)) + \frac{1}{2}\max(f(x_2), g(x_2)) \\ &= \frac{h(x_1)+h(x_2)}{2}\end{aligned}$$

and similarly:

$$\frac{g(x_1)+g(x_2)}{2} \leq \frac{h(x_1)+h(x_2)}{2}$$

Thus, we have:

$$\begin{aligned}h\Big(\frac{x_1+x_2}{2}\Big) &\leq \max\Big(\frac{f(x_1)+f(x_2)}{2}, \frac{g(x_1)+g(x_2)}{2}\Big) \\ &\leq \max\Big(\frac{h(x_1)+h(x_2)}{2}, \frac{h(x_1)+h(x_2)}{2}\Big) \\ &= \frac{h(x_1)+h(x_2)}{2}\end{aligned}$$

$h(x)$ satisfies *Property A* Proved.

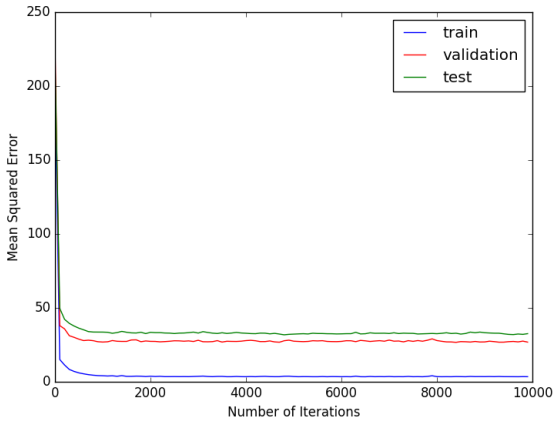
Question 2. Anticoagulants are drugs that reduce blood clotting and are used to prevent a wide variety of medical conditions such as deep vein thrombosis, pulmonary embolism, myocardial infarction and ischemic stroke. Warfarin is the most widely used oral anticoagulant worldwide (with more than 30 million prescriptions in the United States alone in 2004). The correct dose of warfarin is hard to determine because it can vary by as much as a factor of 10 among patients, and the consequences of taking a wrong dose can be lethal. In this problem, you shall **implement stochastic gradient descent to learn a linear regression model** to predict the correct dose of warfarin. You are provided with three files: `train_warfarin.csv` (training data), `validation_warfarin.csv` (training data that we have withheld for you to tune your algorithm parameters, if necessary), and `test_warfarin.csv` (test data). The format of the csv files are given in Annex A.

(a) Train your linear regression model using stochastic gradient descent on `train_warfarin.csv`. Run 10000 iterations of stochastic gradient descent. Save the weights of your model after every 100 iterations of stochastic gradient descent. Plot the mean squared error (i.e., $\frac{1}{n}\sum_{i=1}^{n}(y^{(i)} - \theta \cdot \mathbf{x}^{(i)} - \theta_0)^2$ where $(\mathbf{x}^{(i)}, y^{(i)})$ is an example and n is the number of examples) for each set of weights that are saved (error values on the vertical axis; iterations on the horizontal axis). **On the same graph**, draw one plot each for the mean squared errors on the training set, validation set, and test set (clearly label the three plots). We suggest you try a fixed learning rate of 0.1. If you can get better performance with another learning rate, please do so. Provide crystal clear instructions along with the source code on how to execute it. *(8 points)*

Hints: 1) the stochastic gradient descent algorithm for linear regression presented in the notes/class does not involve θ_0. You will need to figure out what should be the update equation for θ_0, 2) for this question we do not ask you to consider the regularization term. You are, however, free to investigate the effectiveness of the regularization on your own - no submission of such results is required.

(b) Explain in English how could you use the validation set to select the model (with the parameters θ, θ_0) to use on the test set? *(2 points)*

(Note that this is a real-world application published in *The New England Journal of Medicine*[1]. We are using the same attributes as the paper, but the data is artificially generated. The licensing of the paper's data makes it tricky to distribute in class, but if you want to experiment with it, you may download it for your own purposes from www.pharmgkb.org. Make sure to read the license!)



(a) The learning rate is set to 0.01. See the sample plots.

(b) The results on the validation set is a reflection of how the trained model will perform on unseen data. Thus, select the set of weights that perform the best on the validation set. You can

[1]Estimation of the Warfarin Dose with Clinical and Pharmacogenetic Data (http://www.nejm.org/doi/full/10.1056/NEJMoa0809329#t=articleBackground)

figure out the lowest error on the validation set from the plots and choose the corresponding weights.

Question 3. In clustering, Euclidean distance is not the only way to measure the distance between two points/vectors. l_p norms is a family of distance measures that are parameterized by $p \geq 1$. The l_p norm of a vector is:

$$\|x\|_p = \left(\sum_j |x_j|_p \right)^{\frac{1}{p}}.$$

Euclidean distance is the l_2 norm of the vector difference between two points, i.e.,

$$\|x - y\|_2 = \left(\sum_j |x_j - y_j|_2 \right)^{\frac{1}{2}}.$$

The Manhattan distance is the l_1 norm of the vector difference between two points, i.e.,

$$\|x - y\|_1 = \sum_j |x_j - y_j|.$$

The l_∞ distance is the maximum absolute element in the vector difference between two points, i.e.,

$$\|x - y\|_\infty = \max_j |x_j - y_j|.$$

(Think about why this is so.)

(a) Consider a set of points $X = (0.6, 0.8), (0.8, 0.6), (-0.8, 0.6)$. Compute the value of z that minimizes $\sum_{x \in X} d(x, z)$ when $d(x, z)$ is defined as follows respectively: 1) the Euclidean distance between x and z, and 2) the Manhattan distance between x and z. *(5 points)*

(b) The following figures (points in the same cluster have the same color) are produced by the k-medoids algorithm for $k = 3$ clusters using l_1 , l_2 , and l_∞ distance measures. Indicate which distance measure is used for each figure. *(5 points)*

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

01.112 Machine Learning, Spring 2018
Homework 2

Question 3. In clustering, Euclidean distance is not the only way to measure the distance between two points/vectors. l_p norms is a family of distance measures that are parameterized by $p \geq 1$. The l_p norm of a vector is:

$$\|x\|_p = \left(\sum_j |x_j|^p \right)^{\frac{1}{p}}.$$

Euclidean distance is the l_2 norm of the vector difference between two points, i.e.,

$$\|x - y\|_2 = \left(\sum_j |x_j - y_j|^2 \right)^{\frac{1}{2}}.$$

The Manhattan distance is the l_1 norm of the vector difference between two points, i.e.,

$$\|x - y\|_1 = \sum_j |x_j - y_j|.$$

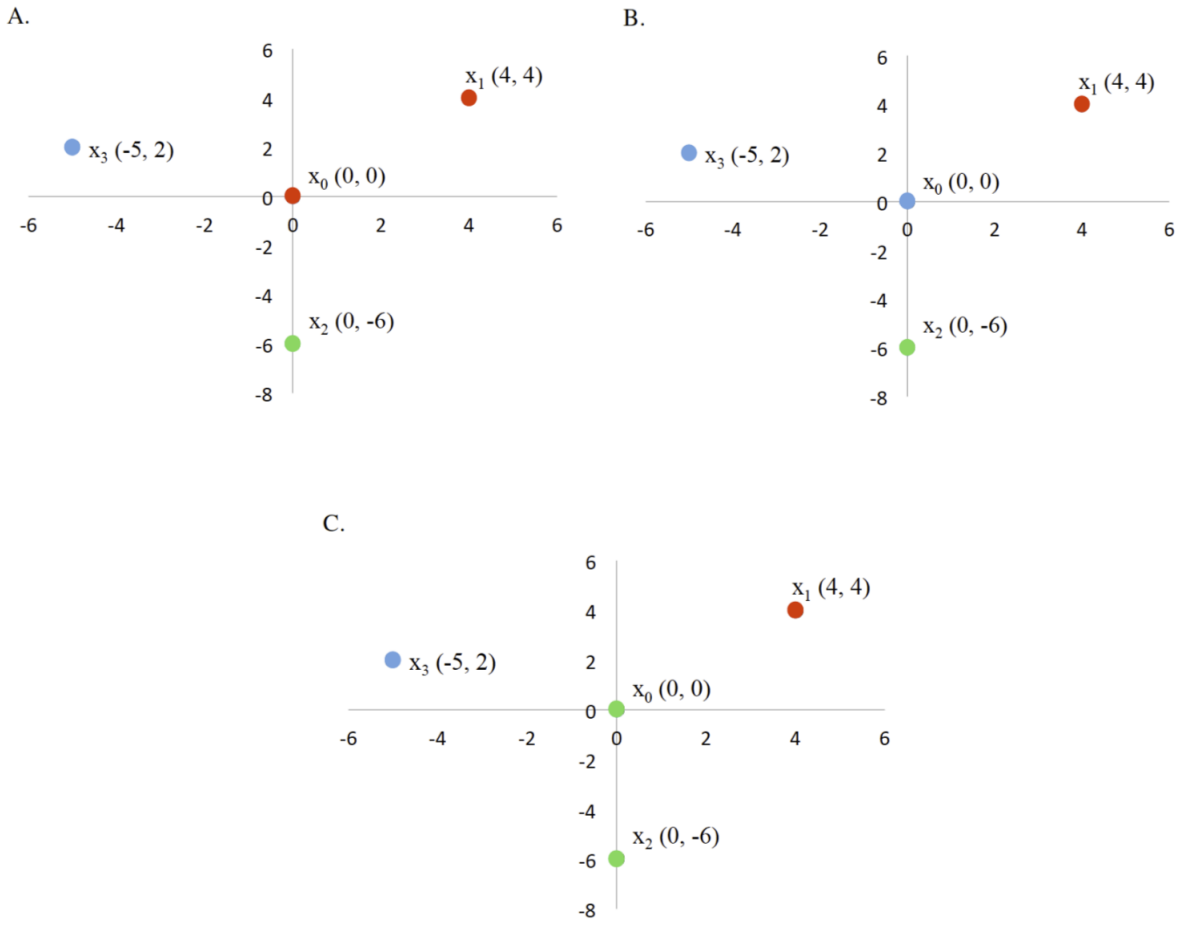
The l_∞ distance is the maximum absolute element in the vector difference between two points, i.e.,

$$\|x - y\|_\infty = \max_j |x_j - y_j|.$$

(Think about why this is so.)

(a) Consider a set of points $X = (0.6, 0.8), (0.8, 0.6), (-0.8, 0.6)$. Compute the value of z that minimizes $\sum_{x \in X} d(x, z)$ when $d(x, z)$ is defined as follows respectively: 1) the Euclidean distance between x and z, and 2) the Manhattan distance between x and z. *(5 points)*

(b) The following figures (points in the same cluster have the same color) are produced by the k-medoids algorithm for $k = 3$ clusters using l_1 , l_2 , and l_∞ distance measures. Indicate which distance measure is used for each figure. *(5 points)*



Solution for Q3 (b):

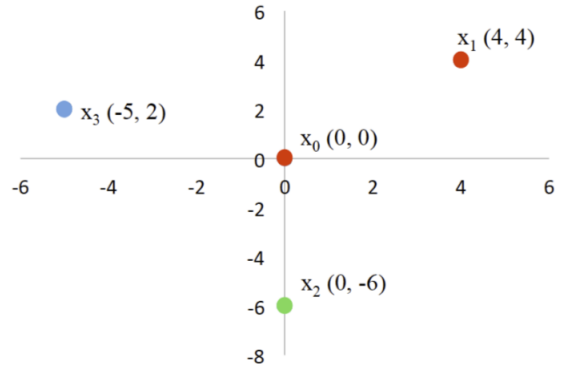
As $k = 3$, it is straightforward that we can select x_1, x_2 and x_3 as the medoids (i.e., centers). The remaining problem is to assign x_0 to one of these clusters.

Measure	dist(x_0, x_1)	dist(x_0, x_2)	dist(x_0, x_3)
l_1	8	6	7
l_2	$4\sqrt{2} \approx 5.66$	6	$\sqrt{29} \approx 5.39$
l_∞	4	6	5

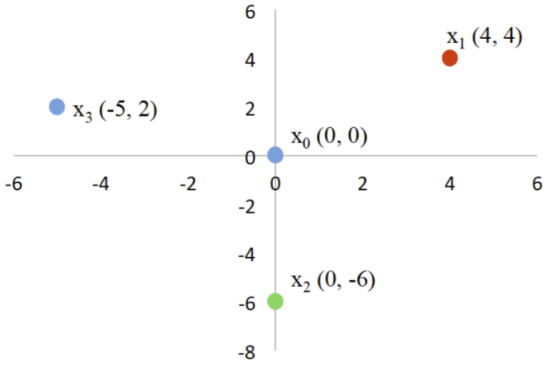
Table 1: Distance measure.

- Figure A: x_0 should be closest to x_1 compared x_2 and x_3, where l_∞ satisfies the condition based on Table 1.
- Figure B: x_0 should be closest to x_3 compared x_1 and x_2, where l_2 satisfies the condition based on Table 1.
- Figure C: x_0 should be closest to x_2 compared x_1 and x_3, where l_1 satisfies the condition based on Table 1.

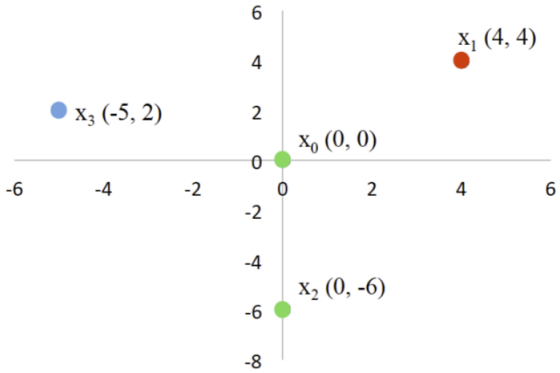
A.



B.



C.



(a) We have two solutions for this question depending on minimizing Euclidean distance or minimizing squared Euclidean distance.

Solution 1 (Euclidean distance):
We wish to minimize the following sum with respect to $z = (z_1, z_2)$:

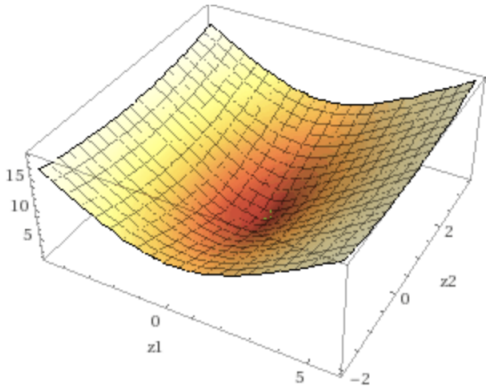
$$\min \sum_{x \in X} d(x, z) = d((0.6, 0.8), z) + d((0.8, 0.6), z) + d((-0.8, 0.6), z).$$

The sum of the Euclidean distance of x_i to a point z is minimized when z is the geometric median, for triangle, also called Fermat point. Let $A = (0.6, 0.8)$, $B = (-0.8, 0.6)$ and $C = (0.8, 0.6)$. The answer is $(0.6, 0.8)$.

For this question we only need an answer, so you are free to use any method you like. In fact you may use tools such as Wolfram Alpha to help you. See below.

$$\min\left\{\sqrt{(0.6 - z1)^2 + (0.8 - z2)^2} + \sqrt{(0.8 - z1)^2 + (0.6 - z2)^2} + \sqrt{(-0.8 - z1)^2 + (0.6 - z2)^2}\right\} \approx 1.69706 \text{ at } (z1, z2) = (0.6, 0.8)$$

3D plot:



Solution 2 (squared Euclidean distance):
We wish to minimize the following sum with respect to $z = (z_1, z_2)$:

$$\min \sum_{x \in X} d^2(x, z) = d^2((0.6, 0.8), z) + d^2((0.8, 0.6), z) + d^2((-0.8, 0.6), z).$$

The sum of the squared Euclidean distances of x_i to a point z is minimized when z is the mean. Thus,

$$z = \frac{(0.6, 0.8) + (0.8, 0.6) + (-0.8, 0.6)}{3} = (\frac{1}{5}, \frac{2}{3}).$$

When d is Manhattan distance, the problem decomposes into coordinate-wise minimization:

$$\arg\min_{z_1}(|0.6 - z_1| + |0.8 - z_1| + |-0.8 - z_1|) = 0.6$$

$$\arg\min_{z_2}(|0.8 - z_2| + |0.6 - z_2| + |0.6 - z_2|) = 0.6$$

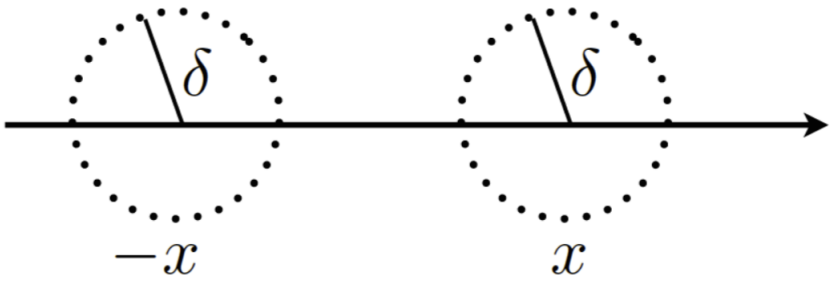
(b) $A : l_\infty, B : l_2, C : l_1$

Question 4. Each iteration of the K-means algorithm consists of two steps: assigning points to centroids, and updating the centroids based on the points assigned to them. Assume that the number of clusters k = 2.

(a) If the centroids are initialized to be the means of two *well-separated* clusters, will the centroids change after the first iteration? (A yes/no answer suffices.) *(1 point)*

(b) If the centroids are initialized by setting each to a random point from each of the two well-separated clusters, how many iterations does it take for k-means to converge? Explain your answer. *(4 point)*

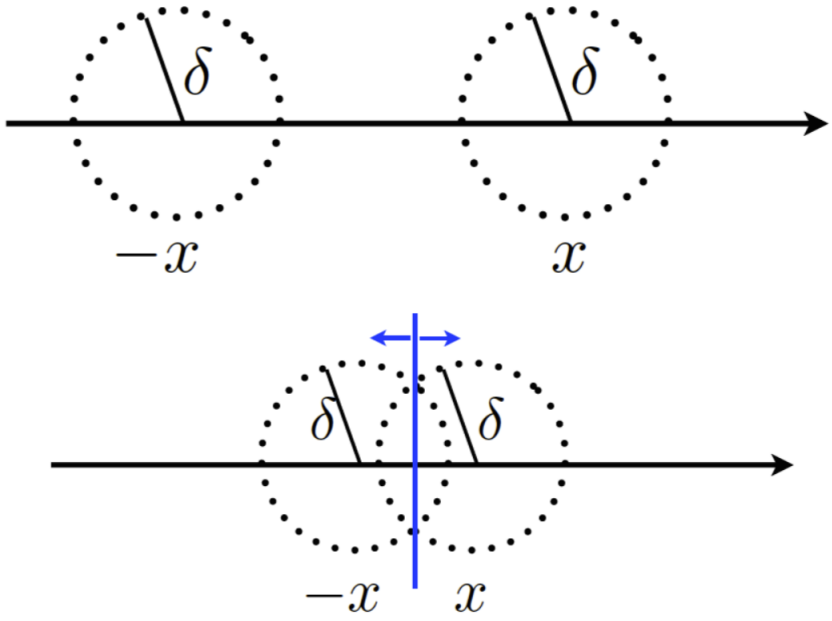
(c) In the figure below, we have two spherical (circle-like) clusters of radius δ that are centered at locations $-x$ and x. For what values of x would the k-means algorithm fail to find the centers of the two clusters regardless of the initialization? Explain your answer. *(5 points)*



(a) No.

(b) It takes one iteration for K-means to converge. Because the clusters are well-separated, the initialization will not change the assignment of points to centroids. The centroids become the cluster means after the first iteration, and algorithm converges.

(c) $|x| < \delta$. The best initialization would be to start with the centers of the two spherical clusters. When $|x| < \delta$, the clusters overlap. Because points are assigned to their closest centroid, divided according to the blue line, the resulting cluster means cannot lie at the center of the original clusters.


δ
δ
−x
x
δ
δ
−x
x