Computer Science Department, Ben Gurion University of the Negev

Introduction to Data Science - Spring 2021

Home assignment 2

Due on June 18th.

May 19, 2021

1. **Minimizations with different norms lead to different answer**
   We are trying to approximate a vector $[x_1, x_2, x_3]$ by a constant c using ℓ_p norms. Assume $x_1 < x_2 < x_3$. Find the best approximation of the vector using a constant c in the following norms:

   (a) ℓ_2 norm (Least squares): $\min_c\{(c-x_1)^2 + (c-x_2)^2 + (c-x_3)^2\}$.

   (b) ℓ_∞ norm: $\min_c\{\max(|c-x_1|, |c-x_2|, |c-x_3|)\}$.

   (c) ℓ_1 norm: $\min_c\{|c-x_1| + |c-x_2| + |c-x_3|\}$.

   Clarification: the x_i's are given, and you need to find c.
   Hint for (b) and (c): the solution is obtained by logic, not by calculations as in (a).

2. **Maximum Likelihood estimation**
   Some background: The geometric distribution gives the probability that the first occurrence of success in k independent $Bernouli(\theta)$ trials with success probability θ (like coin tosses) is achieved at the k-th trial. If the probability of success on each trial is θ, then the probability that the k-th trial (out of k trials) is the first success is given by

   $$\mathbb{P}(X = k) = (1-\theta)^{k-1}\theta.$$

   (a) Let $x_1, x_2, x_3, ..., x_n$ be random samples, which are assumed to be from a $Geometric(\theta)$ distribution. That is, each x_i is the number of Bernouli trials it took to get success, like in the example in the next section. The parameter θ is unknown. Find $\hat{\theta}_{ML}$ the maximum likelihood estimation of θ based on such random sample.

   (b) Suppose that we got the samples: $(x_1, x_2, x_3, x_4) = 2, 3, 3, 5$. What is the maximum likelihood estimation for θ?

3. **Least Squares**

   (a) Consider the following observed values of points $\{(x_i, y_i)\}$:

   $$(-5,-2),(-3,1),(0,4),(2,6),(1,3)$$

   Find the estimated regression line:

   $$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

   where

   $$\hat{\beta}_0, \hat{\beta}_1 = \arg\min_{\beta_0,\beta_1} \sum_i (\beta_0 + \beta_1 x_i - y_i)^2$$

   (b) Compute the error for each of the samples: $e_i = y_i - \hat{y}_i$, where $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$.

   (c) Explain why $\sum_i e_i = 0$. Will that always be the case of such linear modeling?

   (d) Assume that the model noise e_i is a normally distributed random variable. Estimate the expectation and variance of e_i that you got.

   (e) Suppose you were told that one of the samples is not good and better be discarded. Which one would you discard? Why?

4. **Unsupervised Learning: clustering/classifying images of handwritten numbers**

   In this question we will classify handwritten digits from the MNIST data set. The data can be downloaded from: `http://yann.lecun.com/exdb/mnist/`
   Reading the data from Python can be obtained using this `https://www.kaggle.com/hojjatk/read-mnist-dataset` or through the file on the course website.

   We will use the unsupervised learning algorithm Kmeans to classify the images, therefore, for learning the classifier we will use only the training data set (the one in `train-images-idx3-ubyte.gz`).

   (a) Write a program for applying the Kmeans clustering algorithm using k clusters. Use random initialization (random values in [0,1]).

   (b) Using $k = 10$, classify the images from the training data set. As the images in the data come in the range 0-255, you may divide them by 255 to make them between 0 and 1.

   (c) See which of the clusters you found corresponds to which digit. Assign a digit to a cluster that you found using the most common label in that cluster - use the train labels to determine that.

(d) Test your success: for each of the images in the test data, estimate its label using closest centroid (and its cluster's label) and check the percentage of true estimations using the test labels. Report your model's results.

(e) See if the process above is consistent or not with respect to the random initialization. Try this 3 times and report your results.

(f) Try initializing each of the Kmeans centroids using a single image that you found from each label. Are the results better now?