

Machine Learning, Spring 2017: Exercise Sheet 4

Please send your type-set solutions by email to our two TA's Xu He ("Owen") x.he@jacobs-university.de and Felix Schmoll f.schmoll@jacobs-university.de. Join into groups of two or three and submit a single solution sheet per group, indicating the group members' names on the sheet.

Deadline for submission is Friday March 24, 23:59 hrs (email sending timestamp). Submissions arriving later (even a second after midnight) will be corrected but not counted for the course grade.

Problem 1 (linear algebra training). Let $x_1, \dots, x_m \in \mathbb{R}^n$ be m linearly independent n -dimensional vectors, and let μ be their mean. Prove that the centered points $\bar{x}_1, \dots, \bar{x}_m = x_1 - \mu, \dots, x_m - \mu$ span an $m-1$ dimensional subspace of \mathbb{R}^n . (Recall that a set x_1, \dots, x_m of vectors is called linearly independent if $a_1 x_1 + \dots + a_m x_m = \mathbf{0}$ implies $a_1 = \dots = a_m = 0$.)

Problem 2 (programming). Working again with our by now familiar digits dataset, this exercise lets you go through an elementary setup for training a classifier by linear regression.

Step 1: split the available digits data into a training set (100 examples per digit) and a testing set (the other 100 examples per digit). Use only the training data in the next steps and use the testing data for a quality check only after you have trained a classifier. Your total training data set now has 1000 examples. You also have another set with 1000 test examples. The objective is to use the training data to train a classifier for the $k = 10$ digit classes.

Step 2: use PCA to reduce the original data dimension $n = 240$ to $m = 5$ extracted features, ending up with 1000 training feature vectors of size 10 each. Use linear regression (in a version that uses the Matlab or Python/numpy built-in pseudoinverse) to train a linear classifier, that is, compute an $k \times m$ sized weight matrix W . Use this to classify the 1000 training examples that you did not use in training. Record the number $M_{\text{train}}(m = 5)$ of *training misclassifications*. Then use W to classify the 1000 test examples, obtaining a number $M_{\text{test}}(m = 5)$ of *test misclassifications*.

Step 3: repeat step 2 for $m = 10, 15, 20, 25, \dots, 800$ (that is, in increments of 5), recording the corresponding M_{train} and M_{test} values.

Deliverable: a brief typeset documentation of your investigation with a single plot showing both the curves of M_{train} and M_{test} values vs. the m values $m = 5, 10, 15, 20, 25, \dots, 800$. And your code, too, please (with basic inline commenting).