Coursework 2021 - ML4ENG

February 2, 2021

1 Instructions

- For this coursework, you will return a *single MATLAB .m file*. The template to be used can be found on Keats as file “k00000000.m”.
- Before submission, rename your .m file as “k12345678.m”, where “k12345678” represents your knumber.
- The .m file should be executable by pressing the “run” button in the MATLAB’s editor *only once*. If the code does not run, we will be unable to assess your work, and you will obtain a zero mark. To ensure that the code will run, avoid using functions that are only included in toolboxes that are not available in MATLAB’s standard installation and use only the standard MATLAB functions used in the lectures. Some useful functions are included on k00000000.m, and you are encouraged to use them.
- The axes in each figure should be properly labelled. Each figure without labels will cause a loss of 5 points.
- No partial marks will be given: each point is either awarded the full mark, as specified below, or zero points.

2 Description

- To start, import data from the USPS data set “USPS_dataset9296.mat” that you can find on Keats. It contains 9296 16×16 images of handwritten digits that are stacked into vectors of size 256×1 and placed, transposed, as rows in matrix X, along with the corresponding label vector $\mathtt{t}$.
- Extract all the images corresponding to digits 0 and 1, and divide them into a training set $\mathcal{D}$ containing the first 70% of examples from each class and a validation set contaning the last 30% of examples from each class. (Since the number of examples needs to be an integer, you can round the number of examples that go in either set.) Let us denote as N the number of examples in the training set.

- Note that the code in the provided .m file "k00000000.m" guides you through the selection of data points and through the visualization of the data.
- Stack the training examples into a $N \times D$ matrix $X_{\mathcal{D}}$ with $D = 256$ and the corresponding labels in vector $t_{\mathcal{D}}$. Include the examples of digits 0 first and then the examples for digit 1.

2.1 Section 1: Supervised Learning Using Least Squares (30 points)

- While this is a classification problem, in this first section, we are going to treat it as a regression problem. This will allow you to go over this part based on the material in Chapter 4.
- To this end, we will consider the 257×1 vector of features $u(x_n) = [1, x_n^T]^T$ and the linear model $\hat{t}(x|\theta) = \theta^T u(x)$, along with the quadratic loss as in Chapter 4.
- Formulate the ERM problem with quadratic loss using the training data, and obtain the ERM solution θ^{ERM}. Then, address the following points.

1. (10 points) In Figure 1, plot the prediction $\hat{t}(x_n|\theta^{ERM}) = (\theta^{ERM})^T u(x_n)$ as a function of $n = 1, 2, ..., N$ as a green line. In the same figure (using "hold on"), plot the true labels t_n as a black dashed line.
2. (5 points) Calculate and display on the command window (by removing the semi-colon) the training loss and the validation loss. Call the variables including the training loss as "traininglossLS_257" and the validation loss as "validationlossLS_257".
3. (10 points) Now consider a vector of features $u(x_n) = [1, \tilde{x}_n^T]^T$ where $\tilde{x}_n^T$ contains the first 9 entries of the vector x_n. In Figure 1 (using "hold on"), add the resulting prediction $\hat{t}(x|\theta^{ERM})$ as a red line.
4. (5 points) Comment on the comparison between red and green line by using the display command, i.e., display('The predictions with the longer and shorter feature vectors are different because...').

2.2 Section 2: Supervised Learning Using Logistic Regression (30 points)

- In this section, we consider the implementation of logistic regression for the problem under study using the vector of features $u(x_n) = [1, x_n^T]^T$. You will be able to complete this section after going over the material in Chapter 6.

1. (20 points) In Figure 2, plot the training log-loss as a function of the number of iterations for 50 iterations. Make sure to choose a "good" learning rate and mini-batch size. Comment on the reason for your choice using the display comment, i.e., display('I have chosen S=... and gamma=... because...'). Please note that you will have to implement logistic regression using only basic mathematical functions – You are not allowed to use "pre-packaged" functions implementing logistic regression.

2. (10 points) In Figure 2 (using "hold on"), plot the validation loss as a function of the number of iterations for the same choice of learning rate and mini-batch size.

2.3 Section 3: Unsupervised Learning Using PCA (30 points)

- In this section, we consider the implementation of PCA for the same training data set. You will be able to complete this section after going over the material in Chapter 7.

1. (10 points) In Figure 3, plot the first three principal components as gray scale images (use "subplot" to include all three images in the same figure). You can scale the images with a positive scalar in order to use the full dynamic range.

2. (10 point) In Figure 4, plot the first image of the training set, as well the reconstruction of that image using the first three principal components (use "subplot" to include both images in the same figure).

3. (10 points) In Figure 5, plot the contributions of the first three components in a 3D plot by using different markers for images of digit 0 and for images of digit 1 across the entire training data set.

2.4 Section 4: Unsupervised Learning and Supervised Learning (10 points)

- In this section, we consider the three-dimensional vector of features $u(x_n) = [1, z_{n,1}, z_{n,2}]^T$ obtained for each training point x_n by computing the two contributions $z_{n,1}$ and $z_{n,2}$ of the the first two principal components using PCA. You will be able to complete this section after going over the material in Chapter 7.

1. (10 points) In Figure 6, plot the training log-loss as a function of the number of iterations using the same learning rate and mini-batch size selected in Section 2. Comment on the comparison with your results in Section 2, i.e., display('Comparing with the solution in Section 2, I conclude that...').