Machine Learning (Homework 2)

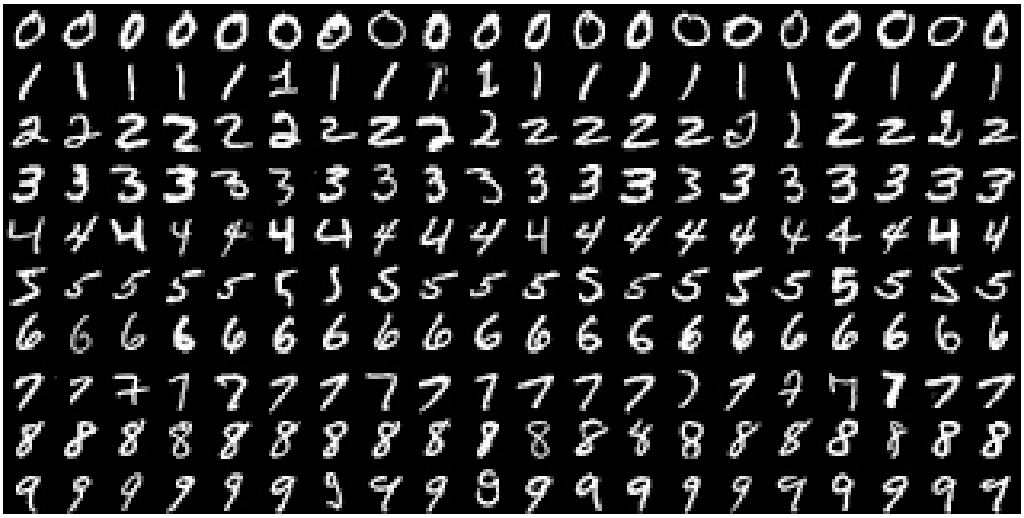
Due date : 2022/11/25 23:59:59

1 Classification Problem (45%)

You are given a dataset of handwritten digits (MNIST.zip) from the MNIST dataset. The dataset contains 10 classes with 128 different images in each class. In this exercise, you need to implement

(1) least squares for classification

(2) logistic regression model for classification



Note: You need to normalize the data samples before training and randomly select 32 images as test data for each class and the remaining images as training data.

1.1 Implement the least squares for classification. You should use a 1-of-K binary coding scheme for the target vector $\mathbf{t}$. **Show** the classification accuracy and the loss value of training and test data. (10%)

1.2 Implement the logistic regression model using batch GD (batch gradient descent), SGD (stochastic gradient descent) and mini-batch SGD with softmax activation. Set the initial weight vector $\mathbf{w}_k = [w_{k1}, \ldots, w_{kF}]$ to be a zero vector where F is the number of features and k is the number of classes. (30%)

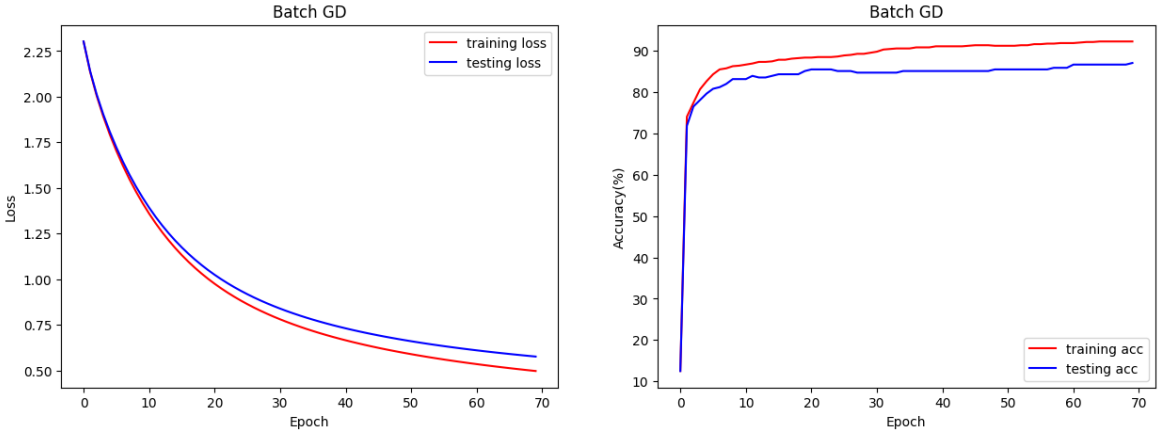
Algorithms	Batch size	Iterations in one epoch
batch GD	N	1
SGD	1	N
mini-batch SGD	B	N/B

N: number of training data, B: batch size (can be selected by yourself)

The error function is defined as

$$E(\mathbf{w}) = -\sum_{m=1}^{N}\sum_{k=1}^{K} t_{nk} \log y_{nk}.$$

(a) **Plot** the learning curves of the loss function and the classification accuracy versus the number of epochs until convergence for training data as well as test data, e.g.



(b) **Show** the final classification accuracy and loss value of training and test data.

(c) Based on your observation about the different algorithms (batch GD, SGD and mini-batch SGD), please **make some discussion**.

1.3 **Make some discussion** about the difference between the results of 1.1 and 1.2. (5%)

2 Gaussian Process for Regression (55%)

In this exercise, you will implement Gaussin process (GP) for regression. The files x.csv and t.csv have input data $\mathbf{x} : \{x_1, x_2, \ldots, x_{300}\}, 0 < x_i < 10$ and the corresponding target data $\mathbf{t} : \{t_1, t_2, \ldots, t_{300}\}$ respectively. Please take the first 150 points as the training set and the rest as the test set. A regression function $y(\cdot)$ is used to express the target value by

$$t_n = y(x_n) + \epsilon_n$$

where the noisy signal ϵ_n is Gaussian distributed, $\epsilon_n \sim \mathcal{N}\left(0, \beta^{-1}\right)$ with $\beta^{-1} = 1$.

1. Please construct a kernel function using the basis functions in the following polynomial model and implement the Gaussian process for regression. (10%)

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{i=1}^{D} w_i x_i + \sum_{i=1}^{D}\sum_{j=1}^{D} w_{ij} x_i x_j \quad (M = 2).$$

   Please **plot** the prediction result like Figure 6.8 of textbook for training set but one standard deviation instead of two and without the green curve. The red line shows the mean $m(\cdot)$ of the Gaussian process predictive distribution. The pink region corresponds to the band with positive and negative of one standard deviation. Training data points are shown in blue. Besides, please **show** the corresponding root-mean-square errors (shown below) for both training and test sets in **.ipynb** file.

$$E_{\mathrm{RMS}} = \sqrt{\frac{1}{N}\left(m(x_n) - t_n\right)^2}.$$

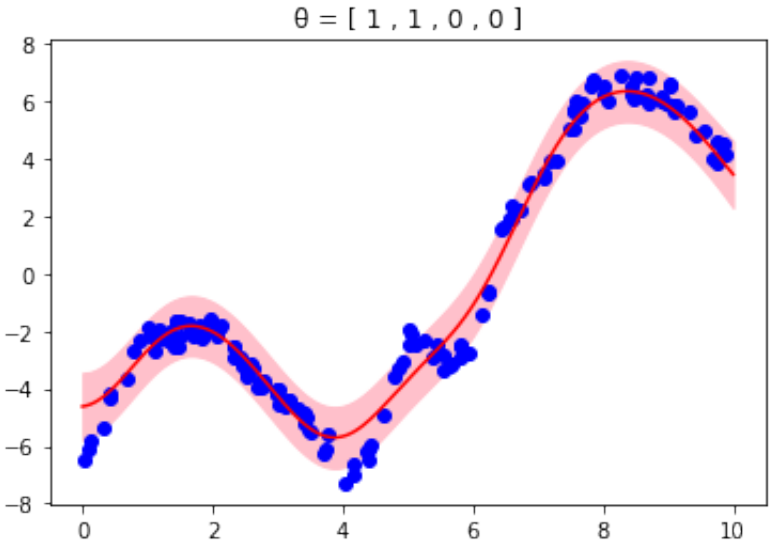
2. Repeat 1 by using the widely used exponential-quadratic kernel function given by

$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left\{-\frac{\theta_1}{2}\|\mathbf{x}_n - \mathbf{x}_m\|^2\right\} + \theta_2 + \theta_3 \mathbf{x}_n^{\top}\mathbf{x}_m$$

   where the hyperparameters $\boldsymbol{\theta} = \{\theta_0, \theta_1, \theta_2, \theta_3\}$ are fixed. Please use the training set with four different combinations (25%)

   - linear kernel $\boldsymbol{\theta} = \{0, 0, 0, 1\}$
   - squared exponential kernel $\boldsymbol{\theta} = \{1, 1, 0, 0\}$
   - exponential-quadratic kernel $\boldsymbol{\theta} = \{1, 1, 0, 16\}$
   - exponential-quadratic kernel $\boldsymbol{\theta} = \{1, 2, 16, 0\}$

   Each combination needs to **plot** the prediction result where the **title of the figure** should be the value of hyperparameter used in the model and **show** the corresponding root-mean-square error. An example of figure is provided below.



3. Try to **tune the hyperparameter** in 2 to find the best combination for the dataset. Use automatic relevance determination (ARD) in Chapter 6.4.4 of textbook. (15%)
4. Explain your findings and **do some discussion**. (5%)

3 Rules

- Please name the assignment as hw2_StudentID.zip (e.g. hw2_0123456.zip).
- In your submission, it needs to contain two files.
  - **hw2_StudentID.ipynb** file which contains all the results and codes for this homework. Also, it should contain the description or explanation for this homework. (Please write all codes in one file.)
  - **hw2_StudentID.py** file which is downloaded from the .ipynb file
- Implementation will be graded by
  - completeness
  - algorithm Correctness
  - model description
  - discussion
- Only Python implementation is acceptable.
- Only the packages we provided is acceptable.
- DO NOT PLAGIARIZE. (We will check program similarity score.)