SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

50.007 Machine Learning, Summer 2023
Homework 1

Due 9 June 2023, 11:59 pm

1. Classification [20 points]

Consider data points from a 2-d space where each point is of the form $x = (x_1, x_2)$. You are given a dataset with two positive examples: (1, 1) and (2, 2), and two negative examples (-1, 1) and (1, -1). For each of the following hypothesis spaces, find the parameters of a classifier (a member of the hypothesis space) that can correctly classify all the examples in the dataset, or explain why no such classifier exists.

(a) [10 points] Inside or outside of an origin-centered circle with radius r (r is the parameter).

(b) [10 points] Above or below a line through the origin with normal vector $\theta = (\theta_1, \theta_2)$ (or $[\theta_1, \theta_2]^T$).

2. Linear classification [30 points]

Automatic handwritten digit recognition is an important machine learning task. The US Postal Service Zip Code Database (http://www.unitedstateszipcodes.org/zip-code-database/) provides 16 $\times$ 16 pixel images preprocessed from scanned handwritten zip codes (US zip codes are the analogues of Singapore postal codes). The task is to recognize the digit in each image. We shall consider the simpler goal of recognizing only two digits: 1 and 5. To simplify our task even further, let's consider only two features: intensity and symmetry. Digit 5 generally occupies more black pixels and thus have higher average pixel intensity than digit 1. Digit 1 is usually symmetric but digit 5 is not. By defining asymmetry as the average difference between an image and its flipped versions, and symmetry as the negation of asymmetry, we can get higher symmetry values for digit 1.

Write an implementation of the perceptron algorithm. Train it on the training set (`train_1_5.csv`), and evaluate its accuracy on the test set (`test_1_5.csv`). The training and test sets are posted on eDimension. csv stands for comma-separated values. In the files, each row is an example. The first value is the symmetry, the second is the average intensity, and the third is the label.

Note: please do NOT shuffle the data. Visit the instances sequentially in the training set when running the perceptron algorithm.

(a) [10 points] Run the perceptron algorithm with offset on the training data for 1 epoch (i.e., traversing the training set 1 time), report the θ, offset and accuracy on the test set.

(b) [10 points] Run the perceptron algorithm with offset on the training data for 5 epochs, report the θ, offset and accuracy on the test set.

(c) [10 points] Submit your code together with crystal clear instructions to run the code (python version, package versions, etc.). The code must be ready to run code without requiring any changes. The TA will follow the instructions to run your code and grade accordingly.

3. Linear regression [30 points]

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) [25 points] 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.

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) [5 points] 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?

4. Ridge regression [20 points]

In this problem, we will explore the effects of ridge regression on generalization. We will use `hw1_ridge_x.dat` as the inputs and `hw1_ridge_y.dat` as the desired output. Please note that a column vector of 1s is already added to the inputs. Recall from Lecture Notes 4, the optimal weight for ridge regression is given by

$$\hat{\theta} = (n\lambda I + X^T X)^{-1} X^T Y \tag{1}$$

To find a suitable value for λ, we will set aside a small subset of the provided data set for estimating the test loss. This subset is called *validation set*, which we use to compute *validation loss*. The remainder of the data will be called the *training set*. Let the first 40 entries of the data set be the training set, and the last 10 entries be the validation set. Concatenate their features into matrices vX and tX, and their responses into vectors vY and tY.

(a) [10 points] Write a function *ridge_regression*(tX, tY, l) that takes the training features, training responses and regularizing parameter λ, and outputs the exact solution θ for ridge regression. Report the resulting value of θ for $\lambda = 0.15$.

(b) [10 points] Use the sample code snippet in `hw1q4_plot_samplecode.py` to plot graphs of the validation loss and training loss as λ varies on logarithmic scale from $\lambda = 10^{-5}$ to $\lambda = 10^{0}$. Write down the value of λ that minimizes the validation loss.

Annex A A row in each csv file for the warfarin problem contains the output and attributes for a patient in the following order.

1. Warfarin Dose (mg/week; output being predicted)
2. Normalized age in years
3. Normalized height in cm
4. Normalized weight in kg
5. VKORC1 genotype A/A (1: present; 0: absent)
6. VKORC1 genotype A/G (1: present; 0: absent)
7. VKORC1 genotype G/G (1: present; 0: absent)
8. VKORC1 genotype unknown (1: unknown; 0: known)
9. CYP2C9 genotype *1/*1 (1: present; 0: absent)
10. CYP2C9 genotype *1/*2 (1: present; 0: absent)
11. CYP2C9 genotype *1/*3 (1: present; 0: absent)
12. CYP2C9 genotype *2/*2 (1: present; 0: absent)
13. CYP2C9 genotype *2/*3 (1: present; 0: absent)
14. CYP2C9 genotype *3/*3 (1: present; 0: absent)
15. CYP2C9 genotype unknown (1: unknown; 0: known)
16. Race Asian (1: true; 0: false)

17. Race Black (1: true; 0: false)
18. Race White (1: true; 0: false)
19. Race Unknown (1: true; 0: false)
20. Taking Enzyme Inducer (1: Yes; 0: No)
21. Taking Amiodarone (1: Yes; 0: No)

Exactly one of the VKORC1 genotypes attributes is 1, all others are 0. Likewise for the CYP2C9 genotype attributes, and race attributes.