CSE/ECE 343/543: Machine Learning
Assignment-1

Max Marks: 130 **Due Date**: Sept. 15, 2018, 11:59PM

Instructions

- Try to attempt all questions. Question 6 is a bonus question for UG students, **but is required for PG students**.
- Keep collaborations at high level discussions. Copying/Plagiarism will be dealt with strictly.
- Start early, solve the problems yourself. Some of these questions may be asked in Quiz/Exams.
- Submission Instructions: Submissions will be through backpack. Create a single *firstname-A1.zip* file containing a report **A1.pdf** and your source folder **A1-src**. Report all your theory solutions and outputs of all programming questions e.g intrinsic and extrinsic parameters,figures, images etc in **A1.pdf**. List name of all the functions/scripts that you have implemented along with the two line summary in **A1.pdf**. Put all your programming functions/scripts in **A1-src**. You are allowed to use *numpy* and *scipy* only for Question 1, while for others you may use a library of your choice. In case of any doubt, initiate a discussion on backpack or drop an email to Aradhya {`aradhyam@iiitd.ac.in`} **and** Kushagra {`kushagra14055@iiitd.ac.in`} with the subject line [**ML18-A1-Doubt**]. Emails with other subject lines may suffer delays in response.
- Report(A1.pdf) is **required**. 50% of the total points of the programming question will be deducted if the results are not reported in A1.pdf
- Late submission penalty: As per course policy.

PROGRAMMING QUESTIONS

1. (50 points) **Linear Regression.**

   i) (*20 points*) Implement linear regression for the boston housing dataset. The dataset description can be found here. The output variable is variable no. 14, 'MEDV', and the input variables are the remaining 13 variables. You need to **implement gradient descent from scratch**, i.e., you cannot use any libraries for training the model. Choose an appropriate learning rate. You may need *feature normalization* (See [SS-Ch-25.2][1] for some notes and techniques of feature normalization) to deal with features having different scales. In your report, include the following plots (each should be reported over 5-fold validation sets)

[1] See course webpage to understand this reference code

a) The plot of Root Mean Squared Error (RMSE) vs. gradient descent iterations for both the training as well as the validation set. The curves should show the mean RMSE and the standard deviation computed over the five folds.

b) Compute the RMSE of the trained model on the training set and the validation set. Report the train and validation RMSE individually for each of the five folds, as well as the mean and standard deviation in $(\mu \pm \sigma)$ format.

ii) (*20 points*)**Regularization.** From the previous part, identify the validation set that has the lowest RMSE, and hold it out as a test set. Use the remaining 80% of the data as your new train+validation set. Note that your model should never see the test set (neither for training, nor for validation). You may use the routines from scikit-learn for this part. (**Note**: For more details on regularization, you (especially PhD and M.Tech. CS+AI students) may refer to [SS-Ch-13.1]. You may also refer to Andrew Ng's notes on regularization and model selection).

a) Use ℓ_2 regularization for training the linear regression model. Use 5-fold cross-validation with grid search on the train+val set (without using the test set) to find the appropriate regularization parameter (hyperparameter). Once the hyperparameter is fixed, use the entire train+val set to train an ℓ_2-regularized linear regression model and plot the RMSE error vs. iterations. Report the RMSE on the test set.

b) Repeat the last part for ℓ_1 regularization.

iii) (*10 points*) Comment on the test set RMSE results comparing the performance of the three linear regression models (none, ℓ_2 and ℓ_1 regularization). Are any of the models overfitting the data? Are any of them underfitting the data?

2. (40 points) **Logistic Regression.** You may use scikit-learn library for this question.

i) (*15 points*) Use the MNIST dataset and perform ℓ_2-regularized logistic regression for this multi-class classification problem. Use the default training and test set specified in the dataset. Compute and report the one vs rest accuracy for each of the 10 classes, for both the training and test sets. See this short tutorial for understanding one vs rest approach for using a binary classifier for a multi-class classification problem. Report the train and test accuracy.

ii) (*15 points*) Repeat the experiment for ℓ_1 regularized logistic regression and report the train and test accuracy

iii) (*10 points*) Comment on whether the models are a good fit, or do they overfit or underfit.

THEORY QUESTIONS

3. (10 points) The sigmoid function is defined as $\sigma(z) = \frac{1}{1+e^{-z}}$ and is used in logistic regression to model the probability. Despite sigmoid being a non-linear function, binary classification with logistic regression uses linear separation. Explain.

4. (10 points) Explain the logit transformation and derive the expression for logistic regression.

5. (10 points) Show that the entropy of the multivariate Gaussian variable $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu},\ \boldsymbol{\Sigma})$ is given by:

   $H[\mathbf{x}] = \frac{1}{2} \ln |\boldsymbol{\Sigma}| + \frac{D}{2}(1 + \ln(2\pi))$,

   where D is the dimensionality of $\mathbf{x}$.

6. (10 points) (**Bonus question**) You are given a function that maps a set of 2D points to another set of 2D points. The function is given by:

   $\mathbf{y} = (1/\lambda)R\mathbf{x} + B$, where $\mathbf{x} \subseteq \mathbb{R}^2$ and $\mathbf{y} \subseteq \mathbb{R}^2$ $B = [a\ b]^\top$

   $$R = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix}$$

   You are given a set of n noisy points in $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$ and $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n]$. Find θ, λ, a and b in terms of X and Y such that the squared ℓ_2 distance between the ground truth data points $\mathbf{Y}$ and the predicted data point $\widehat{\mathbf{Y}}$ is minimized. Find the closed form solution for these parameters, if possible? Alternately, can you also use gradient descent to solve the above problem? If yes, write the parameter update equation.