CS434 — Homework Assignment 1 — Due April 11th 11:59PM, 2015

General instruction.

1. The following languages are acceptable: Java, C/C++, Matlab, Python and R.
2. You can work in team of up to 3 people. Each team will only need to submit one copy of the source code and report.
3. You need to submit your source code (self contained, well documented and with clear instruction for how to run) and a report through the TEACH site `https://secure.engr.oregonstate.edu:8000/teach.php?type=want_auth`

   Please clearly indicate your team members' information.
4. Be sure to answer all the questions in your report. Your report should be typed, submitted in the pdf format. You will be graded based on both your code as well as the report. In particular, **the clarity and quality of the report will be worth 10% of the pts**. So please write your report in clear and concise manner. Clearly label your figures, legends, and tables.

Linear regression

In this assignment you will use the Boston Housing dataset from the CMU StatLib Library that concerns the housing prices in Boston suburbs. The data set contains 13 attributes describing each area (e.g., crime rate, accessibility to major highways) and the target variable is the median value of housing (in thousands) for that area. The description of the data is in the file housing desc.txt on the course web page. The goal is to predict the median value of housing of an area based on 13 attributes. For your convenience the data has been divided into two datasets: (1) a training dataset housing train.txt you should use in the learning phase, and (2) a testing dataset housing test.txt to be used for testing. Your task is to implement the linear regression learning algorithm presented in class and explore some variations with it. In particular:

1. Given the training data, load the data into the corresponding X and Y matrices, where X stores the features and Y stores the desired outputs. The rows of X and Y correspond to the examples and the columns of X correspond to the features.

2. Introduce the dummy variable to X and compute the optimal weight vector $\mathbf{w}$ using $\mathbf{w} = (X^T X)^{-1} X^T Y$. Most programming languages have numerical packages that you can directly use to perform the computation. You don't need to implement your own matrix inversion function. Report the learned weight vector.

3. Apply the learned weight vector to the testing data and compute the sum of squared error(SSE) on the testing data. Report the SSE value.

4. Consider the situation where we do not introduce the dummy variable to X, repeat 2 and 3. How does this influence the prediction accuracy?

5. Consider a variant of linear regression, where the optimal weight is computed as

   $$\mathbf{w} = (X^T X + \lambda I)^{-1} X^T Y,$$

   where I is the identity matrix of the same size as $X^T X$ and λ is a user specified parameter for learning. Compute the optimal $\mathbf{w}$ using this formula with different values of λ (e.g., 0.01, 0.05, 0.1, 0.5, 1, 5. Feel free to explore more choices.) Evaluate each of the learned $\mathbf{w}$ by computing the SSE on the testing data. Plot the SSE value as a function of λ. What behavior do you observe? What do you think is the best λ value for this problem?

6. Compare the different $\mathbf{w}$'s that you got in 5. As the λ value gets bigger, what impact do you observe it has on the weight values?

7. This variant that we introduce is the solution for minimizing the following modified objective:

   $$\sum_{i=1}^{n} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 + \lambda |\mathbf{w}|^2$$

   where the first term is the regular SSE and the second term is called a regularization term and computed as the norm of the weight vector $\mathbf{w}$. Can you use this objective to explain the behavior that you observe in 6?