CSE474/574 Introduction to Machine Learning
Programming Assignment 2
Regression

Due Date: April 16^{th} 2014
Maximum Score: 100

Note A zipped file containing skeleton Matlab function files and data is provided. Note that for each problem, you need to write code in the specified function file. **Do not use any MATLAB toolboxes, built-in functions, or external tools/libraries that directly perform regression, function fitting, or ridge regression**. Using any external code will fetch you 0 points.

Evaluation We will be evaluating your code by executing `script.m` file. Also submit an assignment report (pdf file) summarizing your findings. In the problem statements below, the portions under RECORD heading need to be discussed in the assignment report.

Data Set A data set is provided in the file “diabetes.mat” along with the assignment. The input variables correspond to measurements (physical, physiological, and blood related) for a given patient and the target variable corresponds to the level of diabetic condition in the patient. It contains:

- $\mathbf{x_{train}}$ (242×64) and $\mathbf{y_{train}}$ (242×1) for training.
- $\mathbf{x_{test}}$ (200×64) and $\mathbf{y_{test}}$ (200×1) for testing.

Problem 1 (5 code + 5 report = 10 Points): Experiment with Linear Regression

Implement *ordinary least squares* method to estimate regression parameters by minimizing the squared loss. Note that this is same as maximizing the log-likelihood in the Bayesian setting. You need to implement in the file `learnOLERegression.m`. Also provide code in `script.m` to apply the learnt weights for prediction on both training and testing data and to calculate the root squared error (RSE):

$$J(\mathbf{w}) = \sqrt{\sum_{i=1}^{N}(y_i - \mathbf{w}^\top \mathbf{x}_i)^2} \tag{1}$$

REPORT 1.

Calculate and report the RSE for training and test data for two cases: first, without using an intercept (or bias) term, and second with using an intercept. Which one is better?

Problem 2 (10 code + 10 report = 20 Points): Experiment with Ridge Regression

Implement parameter estimation for ridge regression by minimizing the regularized squared loss as follows:

$$J(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \mathbf{w}^\top \mathbf{x}_i)^2 + \lambda \mathbf{w}^\top \mathbf{w} \tag{2}$$

The analytical derivation has been discussed in Lecture 23. You need to implement it in `learnRidgeRegression.m`.

REPORT 2.
Calculate and report the RSE for training and test data using ridge regression parameters. Use data with intercept. Plot the errors on train and test data for different values of λ. Vary λ from 0 (no regularization) to 0.5 in steps of 0.001. Compare the relative magnitudes of weights learnt using OLE (Problem 1) and weights learnt using ridge regression. Compare the two approaches in terms of errors on train and test data. What is the optimal value for λ and why?

Problem 3 (20 code + 10 report = 30 Points): Using Gradient Descent for Ridge Regression Learning

As discussed in class, regression parameters can be calculated directly using analytical expressions (as in Problem 1 and 2). To avoid computation of $(\mathbf{X}^\top \mathbf{X})^{-1}$, another option is to use gradient descent to minimize the loss function (or to maximize the log-likelihood) function. In this problem, you have to implement the gradient descent procedure for estimating the weights $\mathbf{w}$.

You need to use the `fmincg.m` function which is same as the minimizer that you used for first assignment. You need to implement a function `regressionObjVal.m` to compute the regularized squared error (See (**??**)) and its gradient with respect to $\mathbf{w}$. In `script.m`, this objective function will be used within the minimizer (line 45 in `script.m`).

REPORT 3.
Plot the errors on train and test data obtained by using the gradient descent based learning by varying the regularization parameter λ. Compare with the results obtained in Problem 2.

Problem 4 (10 code + 10 report = 20 Points): Non-linear Regression

In this problem we will investigate the impact of using higher order polynomials for the input features. For this problem use the third variable as the only input variable:

```
x_train = x_train(:,3);
x_test = x_test(:,3);
```

Implement a function in the file `mapNonLinear.m` which converts a single attribute $\mathbf{x}$ into a vector of d attributes, $1, x, x^2, \ldots, x^d$.

REPORT 4.
Using the $\lambda = 0$ and the optimal value of λ found in Problem 2, train ridge regression weights using the non-linear mapping of the data. Vary d from 0 to 6. Note that $d = 0$ means using a horizontal line as the regression line, $d = 1$ is the same as linear ridge regression. Compute the errors on train and test data. Compare the results for both values of λ. What is the optimal value of d in terms of test error in each setting? Plot the curve for the optimal value of d for both values of λ and compare.

Problem 5 (0 code + 20 report = 20 points) Interpreting Results

Using the results obtained for previous 4 problems, make final recommendations for anyone using regression for predicting diabetes level using the input features.

REPORT 5.
Compare the various approaches in terms of training and testing error. What metric should be used to choose the best setting?

Submission

You are required to submit an only file *proj2.zip* to CSE server by using the following script:
submit cse474 proj2.zip (for undergraduate students)
submit cse574 proj2.zip (for graduate student)

File *proj2.zip* must contain 2 folders: *report* and *code*.

- Folder *report* contains your report file (in pdf format).
- Folder *code* must contains the following files:
  1. `script.m`
  2. `learnOLERegression.m`
  3. `learnRidgeRegression.m`
  4. `regressionObjVal.m`
  5. `mapNonLinear.m`

Project report: The hard-copy of report will be collected in class at due date. Your report should include the required analysis for all four problems.