Coursework **Fall 2016**
Statistical Pattern Recognition (COMP136) **Roni Khardon**
Department of Computer Science, Tufts University **Rishit Sheth**

Programming Project 2

This assignment is due by Monday, October 31 1:30pm which is class start time.
Please submit both a hardcopy (in class) and electronically (via provide).

Overview: Experiments with Bayesian Linear Regression

On the course web page you will find 5 datasets for experimentation with regularized linear regression. Each dataset comes in 4 files with the training set in `train-name.csv` the corresponding labels (regression values) in `trainR-name.csv` and similarly for test set. We have both artificial data and real data. The files for the artificial data are named as `NumExamples-NumFeatures` for easy identification of dataset characteristics. Note that the different artificial datasets have different underlying predictive functions (hidden vector w) so they should not be mixed together. The artificial data was generated using the regression model and is thus useful to test the algorithms when their assumptions hold.

Your goals in this assignment are (i) to investigate the effect of the number of examples, the number of features, and the regularization parameter on performance of the corresponding algorithms, and (ii) to investigate the quality of techniques for model selection, i.e., for choosing the regularization parameter.

In all your experiments you should report the performance in terms of the mean square error

$$\text{MSE} = \frac{1}{N}\sum_i (\phi(x_i)^T w - t_i)^2$$

where the number of examples in the corresponding dataset is N.

For the artificial data, you can compare the results to the MSE of the hidden true functions generating the data that give 3.78 (for 100-10), 3.78 (for 100-100), and 4.015 (on 1000-100) on these datasets.

To prepare for the experiments, start by creating 3 additional training sets from the training dataset 1000-100, using the initial 50, 100, and 150 examples respectively. Call these 50(1000)-100, 100(1000)-100, and 150(1000)-100. The test set does not need to be modified. This will allow us to investigate the performance more closely in some cases.

Task 1: Regularization

In this part we use regularized linear regression, i.e., given a dataset, the solution vector w is given by equation (3.28) of Bishop's text.

For each of the 8 datasets (5 original and 3 you created) plot the training set MSE and the test set MSE as a function of the regularization parameter λ (use integer values in the range 0 to 150). In addition, compare these to the MSE of the true functions given above.

In your report provide the results/plots and discuss them: Why can't the training set be used to select λ? How does λ affect error on the test set? How does the choice of the optimal λ vary with the number of features and number of examples? Consider both the cases where the number of features is fixed and where the number of examples is fixed. How do you explain these variations? You might want to plan your plots so that the answers to these questions are easily visible.

Task 2: Learning Curves

Now pick three "representative" values of λ from the first part. For each of these values plot a learning curve for the learned regularized linear regression using the dataset 1000-100.

A learning curve plots the performance of the algorithm as a function of the size of the training set. To produce these curves you will need to draw random subsets of the training set (of increasing sizes) and record the performance (on the fixed test set) when training on these subsets. To get smooth curves approximating the mean performance you will need to repeat the above several times (at least 10 times) and average the results. Use enough training set sizes between 10 and 800 samples to generate smooth curves.

In your report provide the results/plots and discuss them: What can you observe from the plots regarding the dependence on λ and the number of samples? Consider both the case of small training set sizes and large training set sizes. How do you explain these variations?

Task 3: Cross Validation

The previous experiments tell us which value of λ is best in every case *in hindsight*. That is, we need to see the test data and its labels in order to choose λ. This is clearly not a realistic setting and it does not give reliable error estimates. In this part and the next we investigate methods for choosing λ automatically without using the test set.

In this part we use cross validation to pick the regularization parameter λ. This works as follows:

- Use 10 fold cross validation *on the training set* to pick the value of λ in the same range as above. Cross validation is explained below for the benefit of those who have not seen it before.
- Once the value is chosen, we train on the entire training set using this value of λ.
- We can then evaluate and calculate the MSE on the test set.

Implement this scheme and apply it to the 8 datasets. In your report provide the results and discuss them: How do the results compare to the best test-set results from part 1 both in terms of the choice of λ and test set MSE? What is the run time cost of this scheme? How does the quality depend on the number of examples and features?

Task 4: Bayesian Model Selection

In this part we consider the formulation of Bayesian linear regression with the simple prior $w \sim \mathcal{N}(0, \frac{1}{\alpha}I)$. Recall that the evidence function (and evidence approximation) gives a method to pick the parameters α and β. Referring to Bishop's book, the solution is given in equations (3.91), (3.92), (3.95), where m_N and S_N are given in (3.53) and (3.54). These yield an iterative algorithm for selecting α and β using the training set. We can then calculate the MSE on the test set using the MAP (m_N) for prediction.

Implement this scheme, apply it to the 8 datasets, report results, and discuss as in the previous part.

Task 5: Comparison

How do the two model selection methods compare in terms of the test set MSE and in terms of run time? What are the important factors affecting performance for each method? Given these factors, what general conclusions can you make about deciding which model selection method to use?

Submission

- You should **submit** the following items **both electronically and in hardcopy**:
  (1) All your source code for the assignment. Please write clear code with documentation as needed. The source code should (i) run on *homework.eecs.tufts.edu*, (ii) run from the command line *without editing* with a single command (if there is more than one execution command required, include those commands in a Bash script which we can run), and (iii) output the requested results.
  You can assume the data files will be available in the same directory as where the code is executed. Please use filenames as provided for the data. Please include a short README file with the code execution command.
  (2) A PDF report on the experiments, their results, and your conclusions as requested above.
- For electronic submission, put all the files into a zip or tar archive, for example `myfile.zip` (you do not need to submit the data we give you). Please do not use another compression format such as RAR. Then submit using `provide comp136 pp2 myfile.zip`.
- Your assignment will be graded based on the clarity and correctness of the code, and presentation and discussion of the results.

Addendum: 10 Fold Cross Validation for Parameter Selection

Cross Validation is the standard method for evaluation in empirical machine learning. It can also be used for parameter selection if we make sure to use the train set only.

To select parameter a of algorithm A(a) over an enumerated range $a \in V_1, \ldots, V_K$ using dataset D we do the following:

1. Split the data D into 10 disjoint portions.
2. For each value of a in $V_1, \ldots, V_K$:
   (a) For each i in $1 \ldots 10$
      i. Train $A(a)$ on all portions but i and test on i recording the error on portion i
   (b) Record the average performance of a on the 10 folds.
3. Pick the value of a with the best average performance.

Now, in the above, D only includes the training set and the parameter is chosen without knowledge of the test data. We then retrain on the entire train set D using the chosen value and evaluate the result on the test set.