HW Assignment 1 (Due by 1:30pm on Sep 12)

1 Theory (40 points)

1. [**Polynomial Curve Fitting**, 20 points]
Consider the problem of fitting a dataset of N points with a polynomial of degree M, by minimizing the sum-of-squares error:

$$J(\mathbf{w}) = \frac{1}{2N}\sum_{n=1}^{N}\left(h_{\mathbf{w}}(\mathbf{x}_n) - t_n\right)^2 \tag{1}$$

where $h_{\mathbf{w}}(\mathbf{x}) = \sum_{j=0}^{M} w_j x^j$. We have shown in class that the solution to minimizing $J(\mathbf{w})$ satisfies the following set of linear equations:

$$\sum_{j=0}^{M} A_{ij} w_j = T_i \tag{2}$$

$$\text{where } A_{ij} = \sum_{n=1}^{N} x_n^{i+j} \text{ and } T_i \;=\; \sum_{n=1}^{N} x_n^i t_n \tag{3}$$

Derive the solution for the regularized version of polynomial curve fitting, which minimizes the objective function below:

$$J(\mathbf{w}) = \frac{1}{2N}\sum_{n=1}^{N}\left(h_{\mathbf{w}}(\mathbf{x}_n) - t_n\right)^2 + \frac{\lambda}{2}||\mathbf{w}||^2 \tag{4}$$

Show the vectorized version, i.e. using matrices and vectors.

2. [**Probability Theory**, 20 points] *Exercise 1.3, page 58 in PRML.*
Suppose that we have three coloured boxes r (red), b (blue), and g (green). Box r contains 3 apples, 4 oranges, and 3 limes, box b contains 1 apple, 1 orange, and 0 limes, and box g contains 3 apples, 3 oranges, and 4 limes. If a box is chosen at random with probabilities $p(r) = 0.2$, $p(b) = 0.2$, $p(g) = 0.6$, and a piece of fruit is removed from the box (with equal probability of selecting any of the items in the box), then what is the probability of selecting an apple? If we observe that the selected fruit is in fact an orange, what is the probability that it came from the green box?

3. [**Ridge Regression (*)**, 20 points]
Consider the regularized linear regression objective shown below:

$$J(\mathbf{w}) = \frac{1}{2N}\sum_{n=1}^{N}\left(h(\mathbf{x}_n, \mathbf{w}) - t_n\right)^2 + \frac{\lambda}{2}||\mathbf{w}||^2$$

   (a) Minimizing the L2 norm of $\mathbf{w}$ drives all parameters, including w_0, towards 0. Are there situations in which we do not want to constrain w_0 to be small? If yes, give an example, if not show why it is useful to constrain all the weights to be small, including w_0 .

   (b) We have seen in class how to compute the weights $\mathbf{w}$ that minimize $J(\mathbf{w})$. Assume now that we replace $||\mathbf{w}||$ in $J(\mathbf{w})$ with $||\mathbf{w}_{[\mathbf{1}:]}||$, where $\mathbf{w}_{[\mathbf{1}:]} = [w_1, w_2, ..., w_M]$. Derive the solution for $\mathbf{w}$ that minimizes this new objective function.

2 Implementation (80 points)

In this exercise, you are asked to run an experimental evaluation of linear regression on the Athens houses dataset, and on an artificial dataset with and without L2 regularization. The input data is available at `http://ace.cs.ohio.edu/~razvan/courses/ml4900/hw01.zip`. Make sure that you organize your code in folders as shown in the table below. Write code only in the Python files indicated in bold.

```
ml4900/
    hw01/
      code/
        simple.py
        multiple.py
        polyfit.py
        train_test_line.png
      data/
        simple/
          train.txt, test.txt
        multiple/
          train.txt, test.txt
        polyfit/
          train.txt, test.txt, devel.txt
```

1. [**Simple Regression**, 20 points]
   Train a simple linear regression model to predict house prices as a function of their floor size, based on the solution to the system with 2 linear equations discussed in class. Use the dataset from the folder `hw01/data/simple`. Python3 skeleton code is provided in `simple.py`. After training print the parameters and report the RMSE and the objective function values on the training and test data. Plot the training using the default blue circles and test examples using lime green triangles. On the same graph also plot the linear approximation.

2. [**Multiple Regression**, 20 points]
   Train a multiple linear regression model to predict house prices as a function of their floor size, number of bedrooms, and year. Use the normal equations discussed in class, and evaluate on the dataset from the folder `hw01/data/multiple`. Python3 skeleton code is provided in `multiple.py`. After training print the parameters and report the RMSE and the objective function values on the training and test data. Compare the test RMSE with the one from the simple case above.

3. [**Polynomial Curve Fitting**, 40 points]
   In this exercise, you are asked to run an experimental evaluation of a linear regression model, with and without regularization. Use the normal equations discussed in class, and evaluate on the dataset from the folder `hw01/data/polyfit`.

   (a) Select 30 values for $x \in [0, 1]$ uniformly spaced, and generate corresponding t values according to $t(x) = sin(2\pi x) + x(x + 1)/4 + \epsilon$, where $\epsilon = N(0, 0.005)$ is a

zero mean Gaussian with variance 0.005. Save and plot all the values. Done in `dataset.txt`.

(b) Split the 30 samples (x_n, t_n) in three sets: 10 samples for training, 10 samples for validation, and 10 samples for testing. Save and plot the 3 datasets separately. Done in `train.txt, test.txt, devel.txt`.

(c) Consider a linear regression model with polynomial basis functions, trained with the objective shown below:

$$J(\mathbf{w}) = \frac{1}{2N} \sum_{n=1}^{N} \left(h(x_n, \mathbf{w}) - t_n\right)^2 + \frac{\lambda}{2} ||\mathbf{w}||^2$$

Show the closed form solution (vectorized) for the weights $\mathbf{w}$ that minimize $J(\mathbf{w})$.

(d) Train and evaluate the linear regression model in the following scenarios:

1. Without regularization: Use the training data to infer the parameters $\mathbf{w}$ for all values of $M \in [0, 9]$. For each order M, compute the RMSE separately for the training and test data, and plot all the values on the same graph, as shown in class.
2. With regularization: Fixing $M = 9$, use the training data to infer the parameters $\mathbf{w}$, one parameter vector for each value of $\ln \lambda \in [-50, 0]$ in steps of 5. For each parameter vector (lambda value), compute the RMSE separately for the training and validation data, and plot all the values on the same graph, as shown in class. Select the regularization parameter that leads to the parameter vector that obtains the lowest RMSE on the validation data, and use it to evaluate the model on the test data. Report and compare the test RMSE with the one obtained without regularization.

(e) [20 Bonus points] Train and evaluate the linear regression model above using `sklearn`. For ridge regression, add the validation data to the training data and use the `RidgeCV` function to tune the hyper-parameter λ (use 10 folds).

3 Submission

Turn in a hard copy of your homework report at the beginning of class on the due date. Electronically submit on Blackboard a hw01.zip file that contains the hw01 folder in which your code is in the 3 required files.

On a Linux system, creating the archive can be done using the command:
`> zip -r hw01.zip hw01`.

Please observe the following when handing in homework:

1. Structure, indent, and format your code well.
2. Use adequate comments, both block and in-line to document your code.
3. On the theory assignment, **clear and complete explanations and proofs of your results are as important as getting the right answer**.
4. Make sure your code runs correctly when used in the directory structure shown above.