EE5600: Introduction to AI & ML, Fall 2018 (12)
Indian Institute of Technology Hyderabad
HW 0, Assigned: Sunday 12.08.2018.
Due: Saturday 18.08.2018 at 11:59 pm.

Note: The programming exercises must be solved in Python.

1 Theory

1. Formulate and solve the simplest case of linear regression using the sum of squared error (SSE) as the cost function. Follow the notation used in class for the inputs, labels, number of training examples etc. (5)

2. Repeat the above but now using basis functions for estimating the labels. Assume $M+1$ parameters (with the zeroth component corresponding to bias). (5)

3. The sigmoid function is given by $\sigma(x) = \frac{1}{1+\exp(-x)}$. Show that $\tanh(x) = 2\sigma(2x) - 1$. Now show that a general linear combination of logistic sigmoid functions of the form $\hat{y}(x, \mathbf{w}) = w_0 + \sum_{j=1}^{M} w_j \sigma(\frac{x-\mu_j}{s})$ is equivalent to a linear combination of 'tanh' functions of the form $\hat{y}(x, \mathbf{u}) = u_0 + \sum_{j=1}^{M} u_j \tanh(\frac{x-\mu_j}{s})$. Also find the expressions to relate the new parameters $\{u_0, u_1, \ldots, u_M\}$ to the original parameters $\{w_0, w_1, \ldots, w_M\}$. (10)

4. Repeat the first two problems above but for the case where the labels are multi-dimensional with dimension K i.e., $\mathbf{y} \in \mathbb{R}^K$. (5)

5. Solve the simplest case of linear regression when the cost function is a weighted sum of squared error i.e., $E(\mathbf{w}) = \sum_{i=1}^{N} r_i (y^{(i)} - \hat{y}^{(i)})^2 = \sum_{i=1}^{N} r_i (y^{(i)} - \sum_{j=0}^{d} x_j^{(i)} w_j)^2$, and $r_i > 0$. (5)

6. Solve the simplest case of linear regression (with SSE as the cost function) with regularization of the cost function using the $l2$ norm of the weight vector $\mathbf{w}$. Clearly explain when regularization is helpful. (5)

7. Consider the simplest case of the linear regression discussed in class. Now suppose that Gaussian noise η_i with zero mean and variance σ^2 is added independently to each of the input variables x_i. By making use of $E[\eta_i] = 0$ and $E[\eta_i \eta_j] = \delta_{ij}\sigma^2$, show that minimizing the sum of squared error averaged over the noise distribution is equivalent to minimizing the sum-of-squares error for noise-free input variables with the addition of a weight-decay regularization term, in which the bias parameter w_0 is omitted from the regularizer. (5)

8. Show that the ridge regression estimate is the mean (and mode) of the posterior distribution, under a Gaussian prior $\mathbf{w} \sim \mathcal{N}(0, \alpha^2 \mathbf{I})$, and Gaussian sampling model $\mathbf{y} \sim \mathcal{N}(\mathbf{Xw}, \sigma^2 \mathbf{I})$. Find the relationship between the regularization parameter λ in the ridge formula, and the variances α^2 and σ^2. (10)

2 Programming

1. Following the example discussed in class where the underlying relation between the input and label is sinusoidal, implement the following:

   (a) Vanilla linear regression. (10)

   (b) Linear regression using a polynomial basis function. (10)

   (c) $l2$ regularized linear regression using a polynomial basis function. (10)

   (d) Maximum likelihood weight estimation assuming that the labels follow a Gaussian distribution when conditioned on the inputs and weights. (10)

   (e) Maximum posterior weight estimation assuming that the labels follow a Gaussian distribution when conditioned on the inputs and weights, and the weights themselves follow a Gaussian distribution with parameter α. (10)

   In each of your implementations, experiment with various values for the parameters of the model, number of training samples, the Lagrangian multiplier λ, hyperparameter α and report your findings. Clearly show when regularization helps.

   Use the following code snippet to generate training samples.

```
import numpy as np
# Number of training samples
N = 10
# Generate equispaced floats in the interval [0, 2π]
x = np.linspace(0, 2*np.pi, N)
# Generate noise
mean = 0
std = 0.05
# Generate some numbers from the sine function
y = np.sin(x)
# Add noise
y += np.random.normal(mean, std, N)
```