CS4442B & CS9542B: Artificial Intelligence II – Assignment #1

Due: 23:59pm, February 9, 2020

1 Refreshing Mathematics [20 points]

Let $w \in \mathbb{R}^n$ is an n-dimensional column vector, and $f(w) \in \mathbb{R}$ is a function of w. In Lecture 2, we have defined the *gradient* $\nabla f(w) \in \mathbb{R}^n$ and *Hessian matrix* $H \in \mathbb{R}^{n\times n}$ of f with respect to w.

(a) [5 points] Let $f(w) = w^\top x$, where $x \in \mathbb{R}^n$ is a n-dimensional vector. Compute $\nabla f(w)$ using the definition of gradient.

(b) [5 points] Let $f(w) = \text{tr}(ww^\top A)$, where $A \in \mathbb{R}^{n\times n}$ is a squared matrix of size $n \times n$, and $\text{tr}(A)$ is the *trace* of the squared matrix A. Using the definition of gradient, compute $\nabla f(w)$. (Hint: you can use the property of trace: $\text{tr}(AB) = \text{tr}(BA)$)

(c) [5 points] Let $f(w) = \text{tr}(ww^\top A)$. Compute the *Hessian matrix* H of f with respect to w using the definition.

(d) [5 points] In Lecture 5, we have define the sigmoid function: $\sigma(a) = \frac{1}{1+e^{-a}}$. Let $f(w) = \log(\sigma(w^\top x))$, where log is the natural logarithmic function. Compute $\nabla f(w)$ using the definition of gradient. (Hint: let $a = w^\top x$, then you can use the chain rule to first compute the derivative $\frac{d\log(\sigma(a))}{da}$ with respect to a and then compute the gradient of a with respect to w)

2 Linear and Polynomial Regression [50 points]

For this exercise, you will implement linear and polynomial regression in any programming language of your choice (e.g., Python/Matlab/R). The training data set consists of the features `hw1xtr.dat` and their desired outputs `hw1ytr.dat`. The test data set consists of the features `hw1xte.dat` and their desired outputs `hw1yte.dat`.

(a) [5 points] Load the training data `hw1xtr.dat` and `hw1ytr.dat` into the memory and plot it on one graph. Load the test data `hw1xte.dat` and `hw1yte.dat` into the memory and plot it on another graph.

(b) [10 points] Add a column vector of 1's to the features, then use the linear regression formula discussed in Lecture 3 to obtain a 2-dimensional weight vector. Plot both the linear regression line and the training data on the same graph. Also report the average error on the training set using Eq. (1).

$$err = \frac{1}{m}\sum_{i=1}^{m}(w^\top x_i - y_i) \tag{1}$$

(c) [5 points] Plot both the regression line and the test data on the same graph. Also report the average error on the test set using Eq. (1).

(d) [10 points] Implement the 2nd-order polynomial regression by adding new features x^2 to the inputs. Repeat (b) and (c). Compare the training error and test error. Is it a better fit than linear regression?

(e) [10 points] Implement the 3rd-order polynomial regression by adding new features x^2, x^3 to the inputs. Repeat (b) and (c). Compare the training error and test error. Is it a better fit than linear regression and 2nd-order polynomial regression?

(d) [10 points] Implement the 4th-order polynomial regression by adding new features x^2, x^3, x^4 to the inputs. Repeat (b) and (c). Compare the training error and test error. Compared with the previous results, which order is the best for fitting the data?

3 Regularization and Cross-Validation [30 points]

(a) [10 points] Using the training data to implement ℓ_2-regularized for the 4th-order polynomial regression (page 12 of Lecture 4, note that we do not penalize the bias term w_0), vary the regularization parameter $\lambda \in \{0.01, 0.1, 1, 10, 100, 1000, 10000\}$. Plot the training and test error (averaged over all instances) using Eq. (1) as a function of λ (you should use a $\log_{10}$ scale for λ). Which λ is the best for fitting the data?

(b) [10 points] Plot the value of each weight parameter (including the bias term w_0) as a function of λ.

(c) [10 points] Write a procedure that performs five-fold cross-validation on your training data (page 7 of Lecture 4). Use it to determine the best value for λ. Show the average error on the validation set as a function of λ. Is the the same as the best λ in (a)? For the best fit, plot the test data and the ℓ_2-regularized 4th-order polynomial regression line obtained.