Homework 1

September 15, 2018

Due: September 13, 2018, 11:59 PM EST

Instructions

Your homework submission must cite any references used (including articles, books, code, websites, and personal communications). All solutions must be written in your own words, and you must program the algorithms yourself. If you do work with others, you must list the people you worked with. If you solve any problems by hand just digitize that page and submit it (make sure the problem is labeled).

Your programs must be written in Python. All code must be able to compile and run for full credit. Comment all code following proper coding conventions. Remember, if we can't read it, we can't grade it! (For more information on python coding standards, refer to: *https://www.python.org/dev/peps/pep-0008/*)

You should submit your assignment via Github. Submit your solutions as a PDF named "hw(hw #).pdf". For example, homework 1 should be submitted as hw01.pdf. If the assignment requires coding, submit your working code as a .py file with the same name.

If you have any questions address them to:

- Connor McCurley (TA) – cmccurley@ufl.edu
- Xiaolei Guo (TA) – suninth@ufl.edu
- Daniel Wells (TA) – dwells@ufl.edu

Question 1 - 3.5 points

Consider the polynomial curve fitting example discussed in class. As discussed, when the model order is *too* small, the training data is generally *underfit* and when the model order is *too* high, the result can *overfit* the training data. Write a small script of code that mimics our polynomial curve fitting function. The code should generate simulated data from the true function with added zero-mean Gaussian noise (with the true function assumed to be sinc function). The code should also generate a separate validation test data set generated in the same way. Then, after fitting the polynomial to the training data across a range of model orders and evaluated on both the training and testing data, your code should generate a plot similar to the one shown in Figure 1. Also, provide a discussion based on your plot about which model order, M, should be used to avoid over-training.

$$sinc(x) = \begin{cases} 1 & for\ x = 0 \\ \frac{sin(x)}{x} & otherwise \end{cases}$$

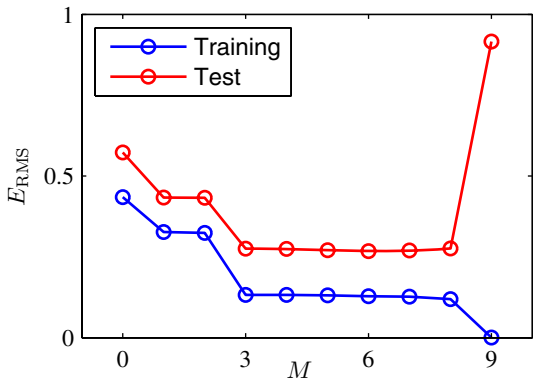

Figure 1: Figure 1.5 from the Bishop textbook. The y-axis corresponds to the root-mean-square error between the predicted and the true value (on either the training data or test data sets). The x-axis corresponds to the model order.

Discussion:
In the plot above, the optimal model order would be $M = 3$. Before 3, it can be observed that the model order was too small to fit the data. At $M = 9$, the training error goes to 0 and the validation shoots up. This is indicative of overfitting, where the training data (including noise) is 'memorized'. The overfit model is not generalized well enough to provide low error on the test set. The training and validation errors between $M = 3$ and $M = 8$ are similar, so it makes sense to select the smallest M as the optimal model order.

Overfitting often occurs from a combination of training on too few samples to accurately represent the true distribution and employing too complex of a model.
Some ways to *avoid overfitting* would be to use a less complex model, add regularization, or (the best way) train with more data.

Question 2 - 4.5 points

Recall:

Assuming a univariate Gaussian data likelihood given N i.i.d. data points:

$$p(\mathbf{X}|\mu) = \prod_{n=1}^{N} \mathcal{N}(x_n|\mu, \sigma^2) \tag{1}$$

and a Gaussian prior distribution on the mean:

$$p(\mu|\mu_0) = \mathcal{N}(\mu|\mu_0, \sigma_0^2) \tag{2}$$

with fixed variances (σ^2, σ_0^2, and $\sigma^2 \neq \sigma_0^2$), the posterior distribution is given by:

$$p(\mu|\mathbf{X}) = \mathcal{N}(\mu|\mu_N, \sigma_N^2) \tag{3}$$

where

$$\mu_N = \frac{\sigma^2}{N\sigma_0^2 + \sigma^2}\mu_0 + \frac{N\sigma_0^2}{N\sigma_0^2 + \sigma^2}\mu_{ML} \tag{4}$$

$$\frac{1}{\sigma_N^2} = \frac{1}{\sigma_0^2} + \frac{N}{\sigma^2} \tag{5}$$

where μ_{ML} is the maximum likelihood solution for μ given the N data points.

- In or Binomial/Beta example in class, we computed the ML and MAP solutions for the μ parameter of the Binomial distribution iteratively with an increasing number of trials/random draws. Recall, the parameter μ represented the probability of heads.
- In this homework question, you will do the same sort of experiment for random draws from a Gaussian distribution (i.e., a Gaussian data likelihood) with a Gaussian prior distribution on the mean parameter (assume a fixed, known variance for both the Gaussian likelihood and Gaussian prior).
- Write a script that iteratively draws one data point from the true Gaussian distribution (with known mean). Each iteration, compute and ML solution and the MAP solution for the Gaussian mean. After each draw, update the prior distribution to be replaced with the posterior distribution from the previous draw (just like the Binomal/Beta example in class).

- In your solution, provide:
  - Display multiple sample runs of your code and include a description of what the code shows you about ML vs MAP solutions. Your discussion should illustrate that you understand ML and MAP concepts and their differences. Your discussion should answer, at a minimum, the following questions:
    * What happens when the prior mean is initialized to the wrong value? To the correct value?
    * What happens as you vary the prior variance from small to large?
    * What happens when the likelihood variance is varied from small to large?
    * How do the initial values of the prior mean, prior variance, and likelihood variance interact to effect the final estimate of the mean?

Discussion:
MLE and MAP are two methods for estimating the parameters of an unknown distribution. Given enough, representative data, MLE will often provide reliable results. However, its weakness is revealed when working with small amounts of data. Recall the class example of flipping a coin. If you flipped a fair coin 5 times and each of the outcomes came out as tails, MLE would say that every flip will come out as tails! MAP addresses this problem by imposing a prior belief. Now if we get tails 5 times in a row, but we say that we believe with high certainty that the coin is fair, then our model's estimate (although not totally correct) will be closer to the true value. Given plenty of data, the MAP solution will converge to the MLE solution.
Some things to think about with MAP: when the prior mean is wrong (and has a small variance), the MAP solution takes a lot of data (many draws from the data likelihood) to converge to the true solution. When the prior variance is large, you are saying that "I believe the parameters to be some value, but with less certainty". With a large data likelihood variance it will take significantly more data for both MLE and MAP to accurately fit the distribution (although MAP has an advantage if your prior is relatively accurate).

Question 3 - 0.5 points

Consider $f(\mathbf{x}) = 3\mathbf{x}^T\mathbf{x} + 4\mathbf{y}^T\mathbf{x} - 1$ where $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$.

1. What is $\frac{\partial f}{\partial \mathbf{x}}$? Show your work.

Solution:

$$\begin{aligned}\frac{\delta f(\mathbf{x})}{\delta \mathbf{x}} &= 3\frac{\delta}{\delta \mathbf{x}}\mathbf{x}^T\mathbf{x} + 4\frac{\delta}{\delta \mathbf{x}}\mathbf{y}^T\mathbf{x} - \frac{\delta}{\delta \mathbf{x}}1 \\ &= 3((\mathbf{x})^T + \mathbf{x}^T) + 4\mathbf{y}^T + 0 \\ &= 3(2\mathbf{x}^T) + 4\mathbf{y}^T \\ &= 6\mathbf{x}^T + 4\mathbf{y}^T\end{aligned}$$

Question 4 - 0.5 points

Consider $f(\mathbf{x}) = -10\mathbf{x}^T\mathbf{Q}\mathbf{x} + 4\mathbf{y}^T\mathbf{x} + 2$ where $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, $\mathbf{Q} \in \mathbb{R}^{d\times d}$ and $\mathbf{Q}$ is symmetric.

1. What is $\frac{\partial f}{\partial \mathbf{x}}$? Show your work.

Solution:

$$\begin{aligned}\frac{\delta f(\mathbf{x})}{\delta \mathbf{x}} &= -10\frac{\delta}{\delta \mathbf{x}}\mathbf{x}^T\mathbf{Q}\mathbf{x} + 4\frac{\delta}{\delta \mathbf{x}}\mathbf{y}^T\mathbf{x} - \frac{\delta}{\delta \mathbf{x}}2 \\ &= -10((\mathbf{Q}\mathbf{x})^T + \mathbf{x}^T\mathbf{Q}) + 4\mathbf{y}^T + 0 \\ &= -10(\mathbf{x}^T\mathbf{Q}^T + \mathbf{x}^T\mathbf{Q}) + 4\mathbf{y}^T \quad (\mathbf{Q} \quad \text{is symmetric means} \quad \mathbf{Q} = \mathbf{Q}^T) \\ &= -10(\mathbf{x}^T\mathbf{Q} + \mathbf{x}^T\mathbf{Q}) + 4\mathbf{y}^T \\ &= -10(2\mathbf{x}^T\mathbf{Q}) + 4\mathbf{y}^T \\ &= -20\mathbf{x}^T\mathbf{Q} + 4\mathbf{y}^T\end{aligned}$$

Question 5 - 0.5 points

Consider $f(\mathbf{x}) = 8\mathbf{x}^T\mathbf{Q}\mathbf{x} - 2\mathbf{y}^T\mathbf{Q}^T\mathbf{x} + 6$ where $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, $\mathbf{Q} \in \mathbb{R}^{d\times d}$ and $\mathbf{Q}$ is symmetric.

1. What is $\frac{\partial f}{\partial \mathbf{x}}$? Show your work.

Solution:

$$\begin{aligned}\frac{\delta f(\mathbf{x})}{\delta \mathbf{x}} &= 8\frac{\delta}{\delta \mathbf{x}}\mathbf{x}^T\mathbf{Q}\mathbf{x} - 2\frac{\delta}{\delta \mathbf{x}}\mathbf{y}^T\mathbf{Q}^T\mathbf{x} - \frac{\delta}{\delta \mathbf{x}}6 \\ &= 8((\mathbf{Q}\mathbf{x})^T + \mathbf{x}^T\mathbf{Q}) - 2\mathbf{y}^T\mathbf{Q}^T + 0 \\ &= 8(\mathbf{x}^T\mathbf{Q}^T + \mathbf{x}^T\mathbf{Q}) - 2\mathbf{y}^T\mathbf{Q}^T \quad (\mathbf{Q} \quad \text{is symmetric means} \quad \mathbf{Q} = \mathbf{Q}^T) \\ &= 8(\mathbf{x}^T\mathbf{Q} + \mathbf{x}^T\mathbf{Q}) - 2\mathbf{y}^T\mathbf{Q} \\ &= 8(2\mathbf{x}^T\mathbf{Q}) - 2\mathbf{y}^T\mathbf{Q} \\ &= 16\mathbf{x}^T\mathbf{Q} - 2\mathbf{y}^T\mathbf{Q} \\ &= (16\mathbf{x}^T - 2\mathbf{y}^T)\mathbf{Q}\end{aligned}$$

Question 6 - 0.5 points

Consider $f(\mathbf{x}) = \|\mathbf{4x}\|_2^2$ where $\mathbf{x} \in \mathbb{R}^d$.

1. What is $\frac{\partial f}{\partial \mathbf{x}}$? Show your work.

Solution: Recall: $\|\mathbf{w}\|_2^2 = \mathbf{w}^T\mathbf{w}$

$$
\begin{aligned}
So \quad f(\mathbf{x}) &= \|4\mathbf{x}\|_2^2 = (4\mathbf{x})^T(4\mathbf{x}) \\
\frac{\delta f(\mathbf{x})}{\delta \mathbf{x}} &= 16\frac{\delta}{\delta \mathbf{x}}(\mathbf{x}^T\mathbf{x}) \\
&= 16((\mathbf{x})^T + \mathbf{x}) \\
&= 16(2\mathbf{x}) \\
&= 32\mathbf{x}
\end{aligned}
$$