

Homework 2

Due: Sep 30, 2003

In this assignment you will examine the problem of polynomial regression, that is, of fitting a polynomial curve to a set of continuous-valued measurements.

Let $\mathbf{X} = \{x_1, \dots, x_N\}$ be a set of 1-d inputs (i.e., $x_i \in \mathbb{R}$) and $\mathbf{Y} = \{y_1, \dots, y_N\}$ be the corresponding output measurements (also in \mathbb{R}). Our job is to find a function $f : \mathbb{R} \rightarrow \mathbb{R}$ that approximates the relationship between \mathbf{X} and \mathbf{Y} as well as possible without overfitting. For this assignment, we'll choose a polynomial of order k as our function. That is,

$$\begin{aligned} f_{\mathbf{w}}(x) &= w_0 + w_1x + w_2x^2 + \dots + w_kx^k \\ &= \sum_{i=0}^k w_i x^i \end{aligned} \tag{1}$$

where \mathbf{w} is the vector of the polynomial coefficients.

For “as well as possible”, we'll use a squared-error loss function:

$$L(\mathbf{X}, \mathbf{Y}) = \frac{1}{N} \sum_{i=1}^N (f_{\mathbf{w}}(x_i) - y_i)^2 \tag{2}$$

1. Using the vector $\mathbf{P}_i = [1 \ x_i \ x_i^2 \ \dots \ x_i^k]^\top$ (for “powers of x ”), write Equation 2 in linear algebraic (matrix) notation. Expand the square on the right hand side of the equation and comment on the shape (scalar, vector, matrix, and dimensions of each) of each term in the resulting equation.
2. We wish to find the \mathbf{w} that minimizes (2). Using your result from the previous question, minimize L with respect to \mathbf{w} by differentiating, setting equal to zero, and solving. (Hint: it is simplest if you continue to maintain this in linear algebraic notation and perform the minimization that way. You may leave your result expressed in linear algebra, including operations like inverse or trace.) Note that although \mathbf{P} is *nonlinear* in x , L itself is *linear* in \mathbf{w} – the terms of \mathbf{P} are simply constants with respect to this minimization.

With the theory out of the way now, we can look at some actual code and data (well, synthetic data anyway). The following questions can be done in the programming language of your choice, though I recommend a language such as Matlab or Mathematica that supports linear algebra with primitive operations. You should turn in a copy of your code with your homework. All plots should be legible and well formatted; all axes should be labeled and each plot should have some title, caption, or legend describing its content. Be sure to distinguish discrete points from continuous functions.

3. Generate a synthetic data set (\mathbf{X}, \mathbf{Y}) where $\mathbf{X} = \{x_1, \dots, x_{10}\}$ are 10 points uniformly spaced between 0 and 2π (inclusive) and $y_i = \sin(x_i) + N(0, 0.2)$. (I.e., y is the concept “ $\sin(x)$ ”, plus some small Gaussian noise.) Plot \mathbf{Y} vs. \mathbf{X} and the curve of the “true” concept.

4. Write a program that implements the least-squares solution of \mathbf{w} and use it to find the best-fit order- k polynomial for $k \in \{0, \dots, 9\}$. For each value of k , plot \mathbf{X} , \mathbf{Y} , the true concept curve, and the curve generated by your \mathbf{w} polynomial. Also, print the values of \mathbf{w} and the L_2 norm of \mathbf{w} , $\|\mathbf{w}\| = (\mathbf{w}^\top \mathbf{w})^{\frac{1}{2}}$. Finally, plot $\|\mathbf{w}\|$ vs. k . What do you observe about the quality of the fitted curve, the order of the polynomial (k) and the norm of \mathbf{w} ?

Let us now examine the *generalization error* of the functions that you locate in this way. Generalization error is the average error incurred on data that you have not seen before. In a theoretical sense, it is the integral of $L()$ between the true curve and the functional fit that you generated to it (i.e., between the two curves you plotted in Question 4). In practice, however, you never have direct access to the true curve; all you can get is a *sample* of the curve, as you did in Question 3.

5. Generate 20 new synthetic data sets, each of $N = 100$ points according to the recipe given in Question 3. For each data set, compute the loss between the data labels and the functional approximation *using the coefficients that you generated in Question 4*. I.e., do *not* generate new \mathbf{w} 's from the new data sets you generate in this question; use only the \mathbf{w} 's that you generated in Question 4 (the *training* data) to estimate the outputs the new data sets that you generate in this question (the *test* data).

Plot the *mean* loss against k . On the same plot, display the loss of the *training* data — i.e., the loss of the function estimator on the data used to set its weights. Comment on the relative shapes of these curves and the relation to the norm of \mathbf{w} , as you examined in Question 4.

Let us modify the loss function (2) with a **regularization term** that penalizes extreme values of $\|\mathbf{w}\|$:

$$L_r(\mathbf{X}, \mathbf{Y}) = \frac{1}{N} \sum_{i=1}^N (f_{\mathbf{w}}(x_i) - y_i)^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2 \quad (3)$$

where λ is a parameter that controls the relative strengths of the error and regularization terms.

6. Solve for the \mathbf{w} that minimizes the regularized loss of Eq. 3.
7. Repeat questions 4 and 5, this time holding k constant at 9 and varying

$$\lambda \in \{0, 10^{-50}, 10^{-40}, 10^{-30}, 10^{-20}, 10^{-10}, 10^{-1}, 1\}$$

. What do you observe about the effects of λ on the learned model, $\|\mathbf{w}\|$, and the generalization error rate? Comment on the effects of the regularization term. Can you suggest another reasonable form of regularization for this system?