6.867: Homework 1

1. Gradient descent

As in the past few years, ICML will rely exclusively on electronic formats for submission and review.

1.1. Templates for Papers

2. Linear basis function regression

We consider linear basis function regression as a method to benchmark the robustness of the gradient descent solution presented above. By using the closed-form maximum likelihood equation, we can calculate the maximum likelihood weight vector for our list of basis functions to approximate the data in the form of our basis. In this scenario, we are using data generated by $y(x) = \cos(\pi x) + 1.5\cos(2\pi x) + \epsilon(x)$, where $\epsilon(x)$ is some added noise to the dataset. Running linear regression on a simple polynomial basis of order M, where $\phi_0(x) = x^0$, $\phi_1(x) = x^1$, $\phi_2(x) = x^2$, ..., $\phi_M(x) = x^M$, we calculate the maximum likelihood weight vector by the following:

$$w_{ML} = (\Phi^T\Phi)^{-1}\Phi^T y$$

where w_{ML} is the maximum likelihood weight vector and Φ is given by:

$$\Phi = \begin{bmatrix} \phi_0(x_0) & \phi_1(x_0) & \phi_2(x_0) & \dots & \phi_M(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \phi_2(x_1) & \dots & \phi_M(x_1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \phi_2(x_n) & \dots & \phi_M(x_n) \end{bmatrix}$$

Our choice of M, the degree of our polynomial basis, largely determines the fit of the regression to the data (Figure 2). More specifically, as small values of M, the polynomial basis cannot adequately capture all the data points. At higher values of M however, overfitting occurs in which the weight vector performs well on the training data, but is not well generalized to new data. The value of M therefore must be carefully considered in order to prevent too high variability in our generated regression polynomial.

We can also instead choose our set of basis functions to be the set of cosine functions, where $\phi_1(x) = \cos(\pi x)$, $\phi_2(x) = \cos(2\pi x)$, ..., $\phi_M(x) = \cos(M\pi x)$. This is again calculated for multiple values of M and compared in Figure 3. Interestingly, even when we use the same family of basis functions as used to generate the initial data ($M = 2$), due to the noise $\epsilon(x)$ added to the initial dataset, the maximum likelihood weight vector does not identically match the actual function used:

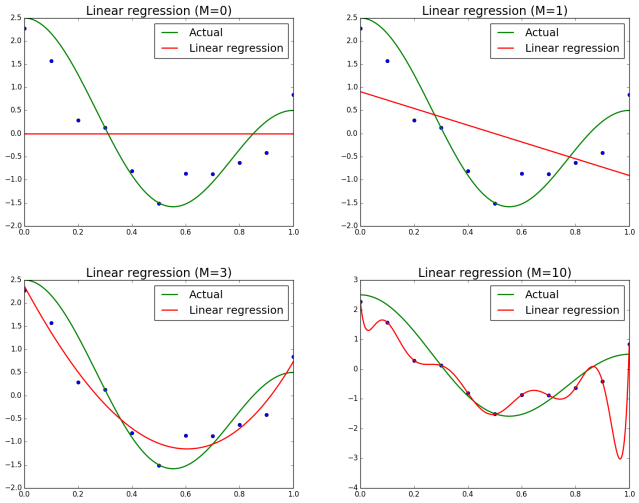

Figure 1. Linear regression for varying values of M.

$$w = \begin{bmatrix} 1 \\ 1.5 \end{bmatrix}$$

$$w_{MLE} = \begin{bmatrix} 0.779 \\ 1.174 \end{bmatrix}$$

where w is the actual weight vector and w_{MLE} is the maximum likelihood estimated weight vector.

3. Ridge regression

We can also instead choose our set of basis functions to be the set of cosine functions, where $\phi_1(x) = \cos(\pi x)$, $\phi_2(x) = \cos(2\pi x)$, ..., $\phi_M(x) = \cos(M\pi x)$. Interestingly, even when we use the same family of basis functions as used to generate the initial data, due to the noise $\epsilon(x)$ added to our dataset, the maximum likelihood weight vector does not identically match the actual function used. This is visualized for varying

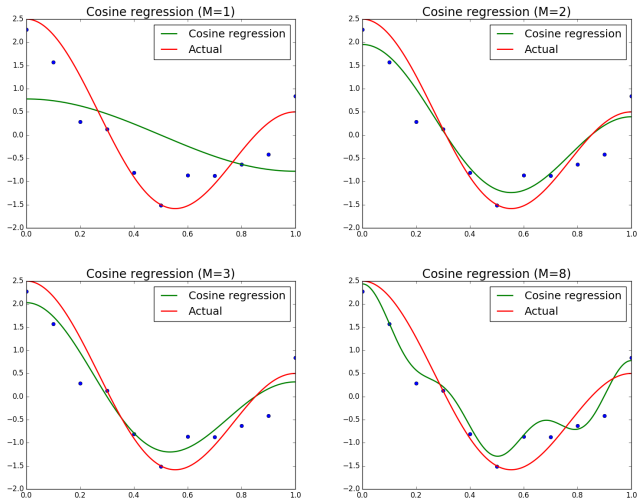

Figure 2. Linear regression for varying values of M.

values of M in Figure 3.

We can also instead choose our set of basis functions to be the set of cosine functions, where $\phi_1(x) = \cos(\pi x)$, $\phi_2(x) = \cos(2\pi x)$, ..., $\phi_M(x) = \cos(M\pi x)$. Interestingly, even when we use the same family of basis functions as used to generate the initial data, due to the noise $\epsilon(x)$ added to our dataset, the maximum likelihood weight vector does not identically match the actual function used. This is visualized for varying values of M in Figure 3.

We can also instead choose our set of basis functions to be the set of cosine functions, where $\phi_1(x) = \cos(\pi x)$, $\phi_2(x) = \cos(2\pi x)$, ..., $\phi_M(x) = \cos(M\pi x)$. Interestingly, even when we use the same family of basis functions as used to generate the initial data, due to the noise $\epsilon(x)$ added to our dataset, the maximum likelihood weight vector does not identically match the actual function used. This is visualized for varying values of M in Figure 3.