CIS 520, Machine Learning, Fall 2019
Homework 2
Due: Monday, September 23rd, 11:59pm
Submit to Gradescope

Matthew Scharf

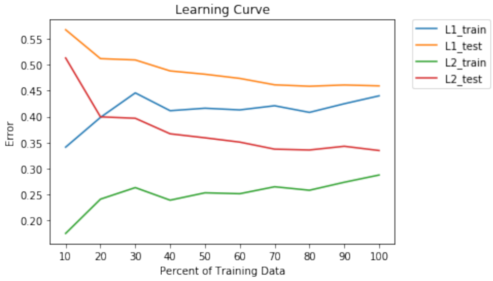
September 23, 2019

1 Programming: Least Squares Regression

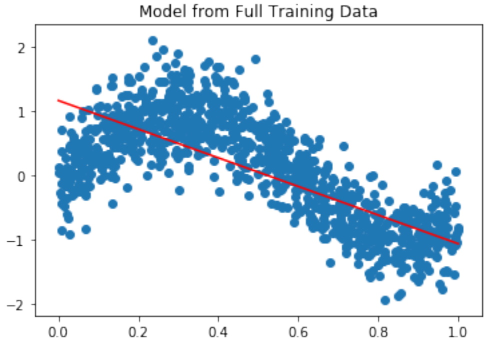
1.1 Data Set 1 (synthetic 1-dimensional data)

This data set contains 100 training examples and 1000 test examples, all generated i.i.d. from a fixed probability distribution. For this data set, you will run unregularized least squares regression.

1. **Learning Curve.**



2. **Analysis of model learned from full training data.**

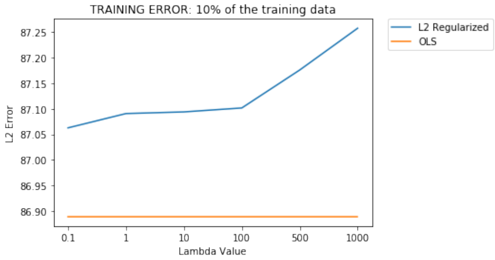


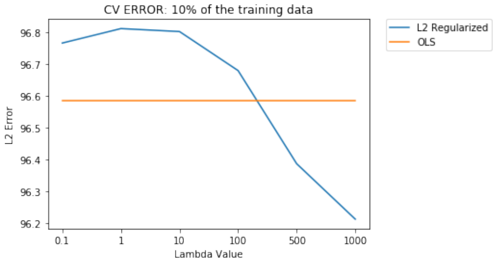
L_2 training error: 0.288
L_2 test error: 0.335
w: -2.234
b: 1.167

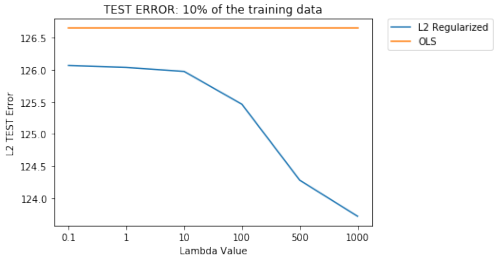
1.2 Data Set 2 (real 8-dimensional data)

This is a real data set that involves predicting concrete strength from 8 properties of concrete. The data set has 721 training examples and 309 test examples. For this data set, you will run both unregularized least squares regression and L_2-regularized least squares regression (ridge regression).

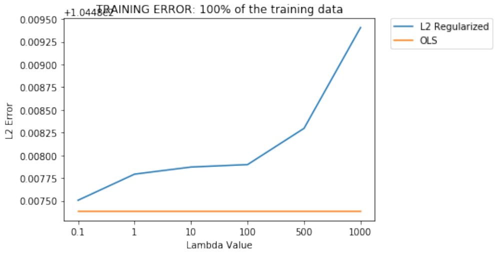
1. **Regression on 10% of the training data.**

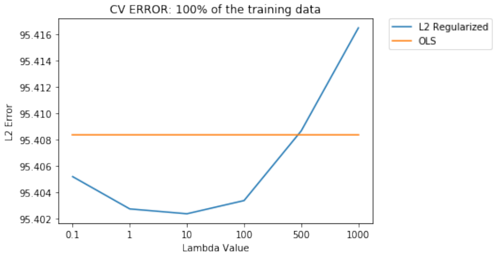


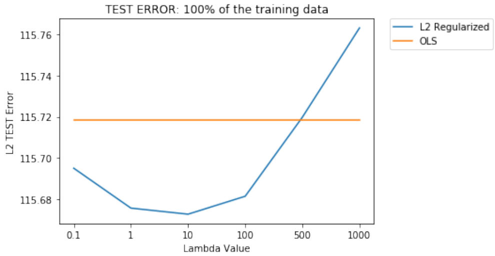




2. **Regression on 100% of the training data.**







3. **Comparison of models learned by two methods** For each of the two training sets considered above (10% and 100%), compare the training and test errors of the models learned using unregularized least squares regression and ridge regression. What can you conclude from this about the value of regularization for small and large training sets?

   On the small training set, the test error is much lower for the regularized regression than it is for OLS, and the error decreases significantly as λ increases.

Meanwhile, on the full training set, the test error for ridge and OLS are similar and the optimal λ of 10 is reasonably small.

This indicates that the smaller the data set, the more one must deal with overfitting using regularization, but as the data set grows in size, the need for regularization (as indicated by the optimal lambda) decreases.

4. **Theoretical Value of λ.** For each of the two training sets considered above (10% and 100%), what does theory predict the difference in the value of λ would be? Which λ should be larger? Do those values align with the conclusion you made in part 1.3?

   For the small data set, the optimal λ seems to be ≥ 1000, while for the full data set, the optimal λ appears to be about 10.

   This makes sense- as the size of the dataset grows, the size of the optimal λ should shrink, so this is in line with the theory.

2 Programming: Batch Gradient Descent

1. **OLS runtime.** Time the closed-form unregularized linear regression implementation you wrote in previous section on the full training data for Data Set 1. Write down the weight and bias terms, $\hat{w}$ and $\hat{b}$, learned from the full training data, as well as the L_2 error on the test data, and the time it took to run the full process.

$$\begin{aligned}\hat{w} &= -2.234\\ \hat{b} &= 1.167\\ L2 \text{ error} &= 0.3348\\ \text{time} &= 0.0019\end{aligned}$$

2. **Gradient descent runtime.** Time the gradient descent implementation you just wrote on the full training data for Data Set 1 with iterations from range {10, 100, 1000}, and a learning rate of 0.01. Write down the weight and bias terms, $\hat{w}$ and $\hat{b}$, learned from the full training data, as well as the L_2 error on the test data, and the time it took to run the full process.

$$\begin{aligned}iterations &= 10\\ \hat{w} &= 0.819\\ \hat{b} &= 0.739\\ L2 \text{ error} &= 2.0081\\ \text{time} &= 0.0046\end{aligned}$$

$$\begin{aligned}iterations &= 100\\ \hat{w} &= 0.09480728\\ \hat{b} &= 0.0300179\\ L2 \text{ error} &= 0.6844305189722777\\ \text{time} &= 0.006621837615966797\end{aligned}$$

$$\begin{aligned}
iterations &= 1000 \\
\hat{w} &= -1.47418892 \\
\hat{b} &= 0.75698276 \\
L2 \text{ error} &= 0.3573302313328567 \\
\text{time} &= 0.019657135009765625
\end{aligned}$$

3. **Comparison of algorithms.** Which algorithm runs faster? Why might that be the case? Why would we ever use gradient descent linear regression in practice while a closed form solution exists?

   In this example, the closed formed solution was a bit faster, but as the number of data points and the dimension get large, the gradient descent will become faster because the closed form solution requires inverting an n by p matrix which has high complexity. Meanwhile, gradient descent does not have that same high increase in complexity and converges fairly quickly, so is very useful as a numerical approximation.

3 Regression Models and Squared Errors

Regression problems involves instance spaces $\mathcal{X}$ and labels, and the predictions, which are real-valued as $\mathcal{Y} = \hat{\mathcal{Y}} = \mathbb{R}$. One is given a training sample $S = ((x_1, y_1), ..., (x_m, y_m)) \in (\mathcal{X} \times \mathbb{R})^m$, and the goal is to learn a regression model $f_S : \mathcal{X} \to \mathbb{R}$. The metric used to measure the performance of this regression model can vary, and one such metric is the squared loss function. The questions below ask you to work with regression problems and squared error losses.

1. The squared error is given by $\mathbb{E}_{(x,y)\sim p(X,Y)}[(f(x)-y)^2]$, where the examples are drawn from a joint probability distribution $p(X,Y)$ on $\mathcal{X}\times\mathbb{R}$. Find the lower bound of the expression $\mathbb{E}_{(x,y)\sim p(X,Y)}[(f(x)-y)^2]$. From this lower bound, what is the optimal expression of $f(x)$, in terms of x and Y?

   $\mathrm{E}_{(x,y)\sim p(X,Y)}[(f(x) - y)^2] =$
   $= \mathbb{E}_{(x,y)\sim p(X,Y)}[((f(x) - \hat{y}) + (\hat{y} - y))^2]$
   $= \mathbb{E}_{(x,y)\sim p(X,Y)}[((f(x) - E[y \mid x]) + (E[y \mid x] - y))^2]$

Now, note that $f(x)$ does not effect $(E[y \mid x] - y)$ but minimizes $(f(x) - E[y \mid x])$
(and therefore $\mathbb{E}_{(x,y)\sim p(X,Y)}[(f(x) - y)^2]$) when $f(x) = E[y \mid x]$. Then, $f(x) - E[y \mid x] = 0$ and so continuing from above:

$\geq \mathbb{E}_{(x,y)\sim p(X,Y)}[(E[y \mid x] - y)^2]$
$= E_x[E_y[(E[y \mid x] - y \mid x)^2]]$
$= E_x[Var(y \mid x)]$

where:
$f(x) = E[y \mid x]$.

2. With this result, complete the following two problems. Consider the regression task in which instances contain two features, each taking values in $[0, 1]$, so that the instance space is $\mathcal{X} = [0, 1]^2$, and with label and prediction spaces belonging to the real space. Suppose examples $(\mathbf{x}, y)$ are drawn from the joint probability distribution D, whose marginal density on $\mathcal{X}$ is given by

$$\mu(\mathbf{x}) = 2x_1, \quad \forall \mathbf{x} = (x_1, x_2) \in \mathcal{X}$$

and the conditional distribution of Y given $\mathbf{x}$ is given by

$$Y|X = \mathbf{x} \sim \mathcal{N}(x_1 - x_2 + 1, 1)$$

What is the optimal regression model $f^*(X)$ and the minimum achievable squared error for D?

As described in 3.1:

$$f(x) = E[y \mid x] = x_1 - x_2 + 1$$

$$L_D[f^*] = E_x[Var(y \mid x)] = Var(y \mid x) = 1$$

3. Suppose you give your friend a training sample $S = ((\mathbf{x}_1, y_1), ..., (\mathbf{x}_m, y_m))$ containing m examples drawn i.i.d from D, and your friend learns a regression model given by

$$f_S(\mathbf{x}) = x_1 - x_2, \quad \forall \mathbf{x} = (x_1, x_2) \in \mathcal{X}$$

Find the squared error of f_S with respect to D.

$$L_D[f_S] = 1 + E_x[Var(y \mid x)] = 1 + 1 = 2$$

4. Consider a linear model of the form

$$f(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{i=1}^{P} w_i x_i$$

together with a sum of squares error function of the form

$$L_P(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} (f(\mathbf{x}_n, \mathbf{w}) - \mathbf{t}_n)^2$$

where P is the dimensionality of the vector $\mathbf{x}$, N is the number of training examples, and $\mathbf{t}$ is the ground truth target . Now suppose that the Gaussian noise ϵ_i with zero mean and variance σ^2 is added independently to each of the input variables x_i. By making use of $\mathbb{E}[\epsilon_i] = 0$ and $\mathbb{E}[\epsilon_i \epsilon_j] = \delta_{ij}\sigma^2$, show that minimizing L_P averaged over the noise distribution is equivalent to minimizing the sum of squares error for noise-free input variables L_P with the addition of a weight-decay regularization term, in which the bias parameter w_0 is omitted from the regularizer.

$$\begin{aligned}
\mathbb{E}[\tilde{L}] =& \; \mathbb{E}_\epsilon[\frac{1}{2} \sum_{n=1}^{N} ((w_0 + \sum_{i=1}^{P} w_i(x_i + \epsilon_i)) - \mathbf{t}_n)^2] \\
=& \; \frac{1}{2} \sum_{n=1}^{N} \mathbb{E}_\epsilon[((w_0 + \sum_{i=1}^{P} w_i x_i - \mathbf{t}_n) + (\sum_{i=1}^{P} w_i \epsilon_i))^2] \\
=& \; \frac{1}{2} \sum_{n=1}^{N} \mathbb{E}_\epsilon[(f(\mathbf{x}, \mathbf{w}) - \mathbf{t}_n) + (\sum_{i=1}^{P} w_i \epsilon_i))^2] \\
=& \; \frac{1}{2} \sum_{n=1}^{N} (\mathbb{E}_\epsilon[(f(\mathbf{x}, \mathbf{w}) - \mathbf{t}_n)^2] + \mathbb{E}_\epsilon[(\sum_{i=1}^{P} w_i \epsilon_i)^2] + 2([(f(\mathbf{x}, \mathbf{w}) - \mathbf{t}_n) * (\sum_{i=1}^{P} w_i \mathbb{E}_{\epsilon_i}[\epsilon_i]))) \\
=& \; \frac{1}{2} \sum_{n=1}^{N} [(f(\mathbf{x}, \mathbf{w}) - \mathbf{t}_n)^2 + \sum_{i=1,j=1}^{P} (w_i w_j \delta_{ij} \sigma^2)]
\end{aligned}$$

This is the sum of squares error for noise free input variables L_p with a weight-decay regularization term.

4 Maximum Likelihood Estimation

1. We have a dataset with N records in which the i^{th} record has one real-valued input attribute x_i and one real-valued output attribute y_i. The model has one unknown parameter w to be learned from data, and the distribution of y_i is given by

$$y_i \sim \mathcal{N}(\log(wx_i), 1)$$

Suppose you decide to do a maximum likelihood estimation of w. What equation does w need to satisfy to be a maximum likelihood estimate?

We want $E_i[N(\log(wx_i), 1)] = E_i[y_i]$

So, optimally, $E_i[\log(wx_i)] = E_i[y_i]$

So, MLE: $w = E_i[\frac{e^{y_i}}{x_i}]$

.

2. Consider a linear basis function regression model for a multivariate target variable $\mathbf{t}$ having a Gaussian distribution of the form

$$p(\mathbf{t}|\mathbf{W}, \mathbf{\Sigma}) = \mathcal{N}(\mathbf{t}|f(\mathbf{x}, \mathbf{W}), \mathbf{\Sigma})$$

where

$$f(\mathbf{x}, \mathbf{W}) = \mathbf{W}^T \phi(\mathbf{x})$$

together with a training data set comprising input basis vectors $\phi(\mathbf{x}_n)$ and corresponding target vectors $\mathbf{t}_n$ with $n = 1, 2, \ldots, N$. Show that the maximum likelihood solution $\mathbf{W}^*$ for the parameter matrix $\mathbf{W}$ has the property that each column is given by the solution to a univariate target variable. Note that this is independent of the covariance matrix $\mathbf{\Sigma}$.

Also, give the maximum likelihood solution for $\mathbf{\Sigma}$ – feel free to use standard results for the MLE solution of Σ in your answer.

$LL = \sum_{n=1}^{N} log(p(t_n; W, X, \Sigma))$

Then,
$0 = \frac{dLL}{dW} = \sum_{n=1}^{N} \Sigma^{-1}(t_n - W^T\phi(x_n)$

So,
$\sum_{n=1}^{N} t_n = W^T \sum_{n=1}^{N} \phi(x_n)$

Equality of vectors means that all the components are equal so:
$\sum_{n=1}^{N} t_n^i = W_i^T \sum_{n=1}^{N} \phi(x_n)$ $\forall i = 1...k$ where k is the dimension of t_n.

So, each row of W^T (aka the columns of W) is the solution to the univariate target variable: t_n^i.

By the standard result for the MLE solution of Σ, Σ is simply the sample covariance matrix:

$\Sigma = \frac{1}{n} \sum_{n=1}^{N} [(t_n - E[t_n])(t_n - E[t_n])^T]$

3. Let us take n samples, namely $x_1, \ldots, x_n$ drawn independently from an exponential distribution given by $f(x; \theta) = \theta \exp(-\theta x)$. Find the log likelihood function $LL(\theta|x_1, \ldots, x_n)$ as well as the MLE estimate for θ.

The log likelihood is:

$$\begin{aligned} LL(\theta|x_1, \dots, x_n) &= \sum_{i=1}^{n} log(\theta e^{-\theta x_i}) \\ &= nlog(\theta) - \theta \sum_{i=1}^{n} x_i \\ &= n(log(\theta) - \theta E_i[x_i]) \end{aligned}$$

Then,
$\frac{d}{d\theta} LL = n(\frac{1}{\theta} - E_i[x_i]) = 0$

So, the MLE: $\theta = \frac{1}{E_i[x_i]}$

4. Building on the previous question, suppose we observe $\mathbf{x} = \{3.2, 4.9, 1.4, 2.9, 8.9\}$ drawn independently from an exponential distribution with unknown θ. What is the MLE estimate of θ? How does this value relate to the sample mean?

   The MLE estimate of θ is:

$$\frac{1}{E_i[x_i]} = \frac{1}{4.26} = 0.2347$$

   The sample mean is:

$$4.26$$

   The MLE estimate for θ is simply the inverse of the sample mean.