CS 189 Introduction to Machine Learning
Spring 2021 Jonathan Shewchuk

Exam Prep 3 Solutions

This exam-prep discussion section covers regression.

1 Multiple Choice

(2) [3 pts] Duplicating a feature in linear regression

- ● Can reduce the L2-Penalized Residual Sum of Squares.
- ○ Can reduce the L1-Penalized Residual Sum of Squares (RSS).
- ● Does not reduce the Residual Sum of Squares (RSS).
- ○ None of the above

(24) [3 pts] Which of the following statements are true for a design matrix $X \in \mathbb{R}^{n\times d}$ with $d > n$? (The rows are n sample points and the columns represent d features.)

- ○ Least-squares linear regression computes the weights $w = (X^\top X)^{-1}X^\top y$.
- ○ The sample points are linearly separable.
- ○ X has exactly $d - n$ eigenvectors with eigenvalue zero.
- ● At least one principal component direction is orthogonal to a hyperplane that contains all the sample points.

(n) [?? pts] Let w^* be the solution you obtain in standard least-squares linear regression. What solution do you obtain if you scale all the input features (but not the labels y) by a factor of c before doing the regression?

- ● $\frac{1}{c}w^*$
- ○ cw^*
- ○ $\frac{1}{c^2}w^*$
- ○ c^2w^*

(o) [?? pts] In least-squares linear regression, adding a regularization term can

- ● increase training error.
- ● increase validation error.
- ○ decrease training error.
- ● decrease validation error.

(p) [?? pts] You have a design matrix $X \in \mathbb{R}^{n \times d}$ with $d = 100{,}000$ features and and vector $y \in \mathbb{R}^n$ of binary 0-1 labels. When you fit a logistic regression model to your design matrix, your test error is much worse than your training error. You suspect that many of the features are useless and are therefore causing overfitting. What are some ways to eliminate the useless features?

- ● Use ℓ_1 regularization.
- ○ Use ℓ_2 regularization.
- ● Iterate over features; check if removing feature i increases validation error; remove it if not.
- ○ If the ith eigenvalue λ_i of the sample covariance matrix is 0, remove the ith feature/column.

2 L2-Regularized Linear Regression with Newton's Method (Spring 2014)

Recall that the objective function for L2-regularized linear regression is

$$J(\mathbf{w}) = \|X\mathbf{w} - \mathbf{y}\|_2^2 + \lambda\|\mathbf{w}\|_2^2$$

where X is the design matrix (the rows of X are the data points).

The global minimizer of J is given by:

$$\mathbf{w}^* = (X^T X + \lambda I)^{-1} X^T \mathbf{y}$$

(a) [?? pts] Consider running Newton's method to minimize J.

Let $\mathbf{w}_0$ be an arbitrary initial guess for Newton's method. Show that $\mathbf{w}_1$, the value of the weights after one Newton step, is equal to $\mathbf{w}^*$.

Recall that Newton's Method for Optimization is

$$w_1 = w_0 - [H(J(w))]^{-1} \nabla_w J(w)$$

Solving for the gradient, we have:

$$\nabla_w J(w) = 2X^T X w - 2X^T Y + 2\lambda w = 2[(X^T X + \lambda I)w - X^T Y]$$

Solving for the Hessian, we have:

$$H(J(w)) = \nabla_w^2 J(w) = 2X^T X + 2\lambda I = 2(X^T X + \lambda I)$$

We initialize w_0 to some value. Note that this won't matter. Plugging this in, we have

$$\begin{aligned}
w_1 &= w_0 - (X^T X + \lambda I)^{-1} 2^{-1} 2[(X^T X + \lambda I)w_0 - X^T Y] \\
&= w_0 - (X^T X + \lambda I)^{-1}(X^T X + \lambda I)w_0 + (X^T X + \lambda I)^{-1} X^T Y \\
&= w_0 - w_0 + (X^T X + \lambda I)^{-1} X^T Y \\
&= (X^T X + \lambda I)^{-1} X^T Y
\end{aligned}$$

Thus, $w_1 = w^*$.

Q3. [?? pts] Error-Prone Sensors

We want to perform linear regression on the outputs of d building sensors measured at n different times, to predict the building's energy use. Unfortunately, some of the sensors are inaccurate and prone to large errors and, occasionally, complete failure. Fortunately, we have some knowledge of the relative accuracy and magnitudes of the sensors.

Let X be a $n \times (d+1)$ design matrix whose first d columns represent the sensor measurements and whose last column is all 1's. (Each sensor column has been normalized to have variance 1.) Let y be a vector of n target values, and let w be a vector of $d+1$ weights (the last being a bias term α). We decide to minimize the cost function

$$J(w) = \|Xw - y\|_1 + \lambda w^\top D w,$$

where D is a diagonal matrix with diagonal elements D_{ii} (with $D_{d+1,d+1} = 0$ so we don't penalize the bias term).

(a) [?? pts] Why might we choose to minimize the ℓ_1-norm $\|Xw - y\|_1$ as opposed to the ℓ_2-norm $|Xw - y|^2$ in this scenario?

Least-squares regression gives too much power to outliers, which is inappropriate for inaccurate or failing sensors. The ℓ_1-normalized cost function does not try as hard to fit the outliers.

(b) [?? pts] Why might we choose to minimize $w^\top D w$ as opposed to $|w'|^2$? What could the values D_{ii} in D represent?

We might want to more heavily penalize the weights associated with the less accurate sensors. Each D_{ii} can be thought of how much we don't trust sensor i.

(c) [?? pts] Derive the batch gradient descent rule to minimize our cost function. Hint: let p be a vector with components $p_i = \text{sign}(X_i^\top w - y_i)$, and observe that $\|Xw - y\|_1 = (Xw - y)^\top p$. For simplicity, assume that no $X_i^\top w - y_i$ is ever exactly zero.

$$\begin{aligned}
\nabla_w(\|Xw - y\|_1 + \lambda w^\top D w) &= \nabla_w((Xw - y)^\top p + \lambda w^\top D w) \\
&= \nabla_w(w^\top X^\top p - y^\top p + \lambda w^\top D w) \\
&= X^\top p + 2\lambda D w
\end{aligned}$$

Therefore, the update rule is $w^{(t+1)} \leftarrow w^{(t)} - \epsilon(X^\top p + 2\lambda D w)$.