CS 181 Spring 2019 Section 1 Notes
(Linear Regression)

Section Topic Requests/Feedback: https://goo.gl/forms/sLh3Wy3qTqa1E0373

1 Maximum Likelihood and Least Squares Regression

1.1 Linear Regression

The simplest model for regression involves a linear combination of the input variables:

$$h(\mathbf{x};\mathbf{w}) = w_1x_1 + w_2x_2 + \ldots + w_mx_m = \sum_{j=1}^{m} w_jx_j = \mathbf{w}^\top\mathbf{x} \tag{1}$$

where $x_j \in \mathbb{R}$ for $j \in \{1,\ldots,m\}$ are the features, $\mathbf{w} \in \mathbb{R}^m$ is the weight parameter, with $w_1 \in \mathbb{R}$ being the bias parameter. (Recall the trick of letting $x_1 = 1$ to merge bias.)

1.2 Least squares Loss Function

$$\mathcal{L}(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{n}\Big(y_i - \mathbf{w}^\top\mathbf{x}_i\Big)^2 \tag{2}$$

$$\mathbf{w}^* = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y} = \arg\min_{\mathbf{w}} \mathcal{L}(\mathbf{w}) \tag{3}$$

where $\mathbf{X} \in \mathbb{R}^{n\times m}$, where each row is one data point (i.e. one feature vector) and each column represents values of a given feature across all the data points.

Exercise: derive $\mathbf{w}^*$ for linear regression using non-matrix form and matrix form differentiation.

1.3 Regularized Least Squares

To penalize complexity, we add a regularization term to the error function. The total error function becomes:

$$\mathcal{L}(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{n}\Big(y_i - \mathbf{w}^\top \mathbf{x}_i\Big)^2 + \frac{\lambda}{2}\mathbf{w}^\top \mathbf{w} \tag{4}$$

This is known as *Ridge* regression.

$$\mathbf{w}^* = (\lambda \mathbf{I} + \mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y} \tag{5}$$

Exercise: derive $\mathbf{w}^*$ for Lasso and Ridge regression using non-matrix form and matrix form differentiation.

1.4 Linear Basis Function Regression

We allow $h(\mathbf{x};\mathbf{w})$ to be a non-linear function of the input vector $\mathbf{x}$, while remaining linear in $\mathbf{w} \in \mathbb{R}^d$:

$$h(\mathbf{x};\mathbf{w}) = \sum_{j=1}^{d} w_j \phi_j(\mathbf{x}) = \mathbf{w}^\top \phi(\mathbf{x}) \tag{6}$$

where $\phi(\mathbf{x}) : \mathbb{R}^m \to \mathbb{R}^d$ denotes the jth term of $\phi(\mathbf{x})$. To merge bias, we define $\phi_1(\mathbf{x}) = 1$.

2 Practice Questions

1. **MLE Estimate of the Bias Term (Bishop (3.19))**

Let $\mathbf{X} \in \mathbb{R}^{n \times m}$ be our design matrix, $\mathbf{y}$ our vector of n target values, $\mathbf{w}$ our vector of $m-1$ parameters, and w_0 our bias parameter. As Bishop notes in (3.18), the least squares error function of $\mathbf{w}$ and w_0 can be written as follows

$$\mathcal{L}(\mathbf{w}, w_0) = \frac{1}{2}\sum_{i=1}^{n}\left(y_i - w_0 - \sum_{j=1}^{m-1} w_j X_{ij}\right)^2 .$$

Show that the value of w_0 that minimizes $\mathcal{L}$ is

$$\begin{aligned} w_0^* &= \frac{1}{n}\sum_{i=1}^{n} y_i - \frac{1}{n}\sum_{j=1}^{m-1} w_j \left(\sum_{i=1}^{n} X_{ij}\right) \\ &= \frac{1}{n}\left(\mathbf{y}^\top \mathbf{1} - \sum_{i=1}^{n} \mathbf{w}^\top \mathbf{x}_i\right) \qquad \text{[compare Bishop (3.19)]} \end{aligned}$$

We have that $\frac{\partial L}{\partial w_0} = -\sum_{i=1}^{n}(y_i - w_0 - \sum_{j=1}^{m-1} w_j X_{ij})$.

Thus, we set $\sum_{i=1}^{n} y_i - nw_0 - \sum_{i=1}^{n}\sum_{j=1}^{m-1} w_j X_{ij} = 0$, and solving for w_0 gives the result. We justify this by saying that the MLE estimate for the bias is simply the average deviation of the outputs from the predictions obtained by multiplying the features and the weights. This makes sense if we imagine our predictions to be off in this systematic manner: the average deviation is a good (but uninformed) guess for a corrective constant. Indeed, we often find that MLE estimates for parameters have intuitive forms.

2. **Maximum Likelihood for the Gaussian (Sequential Estimation of Parameters)**

(a) We are given a data set $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ where each observation is drawn independently from a multivariate Gaussian distribution:

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{|(2\pi)\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right) \tag{7}$$

where $\boldsymbol{\mu}$ is a m-dimensional mean vector, $\boldsymbol{\Sigma}$ is a m by m covariance matrix, and $|\boldsymbol{\Sigma}|$ denotes the determinant of $\boldsymbol{\Sigma}$.

Find the maximum likelihood value of the mean, $\boldsymbol{\mu}_{MLE}$.

(b) Let $\boldsymbol{\mu}_{MLE}^{(n)}$ denote the maximum likelihood estimator of the mean based on n observations. Show that

$$\boldsymbol{\mu}_{MLE}^{(n)} = \boldsymbol{\mu}_{MLE}^{(n-1)} + \frac{1}{n}(\mathbf{x}_n - \boldsymbol{\mu}_{MLE}^{(n-1)}) \tag{8}$$

(a) The likelihood of all the data is

$$\prod_{i=1}^{n} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

Taking the log, we get that the log likelihood equals:

$$\begin{aligned}\log \prod_{i=1}^{n} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}, \boldsymbol{\Sigma}) &= \sum_{i=1}^{n} \log(\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}, \boldsymbol{\Sigma})) \\ &= -\frac{nm}{2}\log(2\pi) - \frac{n}{2}\log(|\boldsymbol{\Sigma}|) - \frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i-\boldsymbol{\mu})\end{aligned}$$

Taking the derivative with respect to $\boldsymbol{\mu}$ and setting it equal to 0, we get

$$0 = \frac{\partial}{\partial \boldsymbol{\mu}} \log p(\mathbf{X}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^{n} \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu})$$

and solving gives us that

$$\boldsymbol{\mu}_{MLE} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i$$

(b)

$$\begin{aligned}
\boldsymbol{\mu}_{MLE}^{(n)} &= \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i \\
&= \frac{1}{n}\mathbf{x}_i + \frac{1}{n}\sum_{i=1}^{n-1}\mathbf{x}_i \\
&= \frac{1}{n}\mathbf{x}_i + \frac{n-1}{n}\boldsymbol{\mu}_{MLE}^{(n-1)} \\
&= \boldsymbol{\mu}_{MLE}^{(n-1)} + \frac{1}{n}(\mathbf{x}_i - \boldsymbol{\mu}_{MLE}^{(n-1)})
\end{aligned}$$

Intuition: When we observe a new data point, we revise our estimate by moving our previous estimate over in the direction of the error $(\mathbf{x}_n - \mu_{MLE}^{(n-1)})$, but scaled by $\frac{1}{n}$ (since this is only one data point out of n total ones).

3. **OLS on Augmented Data (HTF 3.12 & MIT 6.867 Fall '12 Recitation Problems)**

Let $\mathbf{X} \in \mathbb{R}^{n \times m}$ be our design matrix and $\mathbf{y}$ be our vector of n target values. Assume $\mathbf{X}$ and y are both centered, that is assume the mean of each row is 0. Let $\tilde{\mathbf{X}}$ be the $(n+m)$ by m matrix formed by vertically stacking $\mathbf{X}$ on top of $\sqrt{\lambda}\mathbf{I}$, and let $\tilde{\mathbf{y}}$ be the $(n+m)$-length vector formed by vertically stacking $\mathbf{y}$ on top of a vector of m zeros.

That is, let $\tilde{\mathbf{X}} = \begin{bmatrix} X_{11} & \cdots & X_{1m} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{nm} \\ \sqrt{\lambda} & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & \sqrt{\lambda} \end{bmatrix}$ and $\tilde{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \\ 0 \\ \vdots \\ 0 \end{bmatrix}$.

(a) Show that the least squares error function induced by viewing $\tilde{\mathbf{X}}$ as our design matrix and $\tilde{\mathbf{y}}$ as our target values can be written as

$$\frac{1}{2}\sum_{i=1}^{n}\left(y_i - \mathbf{w}^\top \mathbf{x}_i\right)^2 + \frac{\lambda}{2}\mathbf{w}^\top \mathbf{w}$$

(b) Why is this cool?

(a) We have

$$\begin{aligned} \mathcal{L} &= \frac{1}{2}\sum_{i=1}^{n+m}(\tilde{y}_i - \mathbf{w}^\top \tilde{\mathbf{x}}_i)^2 \\ &= \frac{1}{2}\sum_{i=1}^{n}\left(y_i - \mathbf{w}^\top \mathbf{x}_i\right)^2 + \sum_{i=1}^{m}(0 - w_k\sqrt{\lambda})^2 \\ &= \frac{1}{2}\sum_{i=1}^{n}\left(y_i - \mathbf{w}^\top \mathbf{x}_i\right)^2 + \frac{\lambda}{2}\mathbf{w}^\top \mathbf{w} \end{aligned}$$

We know from the previous question (and Bishop (3.19)) that this is the Ridge Regression error function (written with the bias parameter made explicit) exactly.

(b) We see that adding artificial zero-response data is equivalent to regularizing via Ridge Regression!