CS 475/675 Machine Learning: Homework 1
Analytical Questions
15 Points Total Version 1.0

Mingtian Gao, Wenxuan Lu, Zhenhan Gan

mgao32, wlu15, zgan4

Instructions

We have provided this LaTeX document for completing the analytical portion of the assignment. We give you one or more boxes to answer each question. The question to answer for each box will be noted in the title of the box.

Other than your name, do not type anything outside the boxes. Leave the rest of the document unchanged.

Do not change any formatting in this document, or we may be unable to grade your work. This includes, but is not limited to, the height of textboxes, font sizes, and the spacing of text and tables. Additionally, do not add text outside of the answer boxes. Entering your answers are the only changes allowed.

We strongly recommend you review your answers in the generated PDF to ensure they appear correct. We will grade what appears in the answer boxes in the submitted PDF, NOT the original latex file.

Notation

$\mathbf{x_i}$ One input data vector. $\mathbf{x_i}$ is M dimensional. $\mathbf{x_i} \in \mathbb{R}^{M \times 1}$.

We assume $\mathbf{x_i}$ is augmented with a 1 to include a bias term.

$\mathbf{X}$ A matrix of concatenated $\mathbf{x_i}$'s. There are N input vectors, so $\mathbf{X} \in \mathbb{R}^{M \times N}$

y_i The true label for input vector $\mathbf{x_i}$. In regression problems, y_i is continuous.

In general ,y_i can be a vector, but for now we assume it's a scalar: $y_i \in \mathbb{R}^1$.

$\mathbf{y}$ A vector of concatenated y_i's. There are N input vectors, so $\mathbf{y} \in \mathbb{R}^{N \times 1}$

$\mathbf{w}$ A weight vector. We are trying to learn the elements of $\mathbf{w}$.

$\mathbf{w}$ is the same number of elements as $\mathbf{x_i}$ because we will end up computing the dot product $\mathbf{x_i} \cdot \mathbf{w}$.

$\mathbf{w} \in \mathbb{R}^{M \times 1}$. We assume the bias term is included in $\mathbf{w}$.

$h((x))$ The true regression function that describes the data.

i.i.d. Independently and identically distributed.

Bias-variance decomposition We can write $E_D[(f(x, D) - h(x))^2] =$

$(E_D[f(x, D) - h(x))^2 + E_D[(f(x, D) - E_D[f(x, D)])^2]$

where the first term is the bias squared, and the second term is the variance.

Notes: In general, a lowercase letter (not boldface), a, indicates a scalar.

A boldface lowercase letter, $\mathbf{a}$, indicates a vector.

A boldface uppercase letter, $\mathbf{A}$, indicates a matrix.

1) The Bias-Variance Decomposition (2 points)

Consider the linear model $y = f(\mathbf{x}) + \epsilon$, where $\mathbb{E}(\epsilon) = 0$, $Var(\epsilon) = \sigma_\epsilon^2$, $\mathbf{x}^T = (x_1, ..., x_p)$ is a vector of inputs and $f(\mathbf{x}) = \mathbf{x}^T\beta$. Given a training set $T = \left\{ \mathbf{y} = \begin{pmatrix} y^{(1)} \\ \vdots \\ y^{(N)} \end{pmatrix}, \mathbf{X} = \begin{bmatrix} x_1^{(1)} & \cdots & x_p^{(1)} \\ \vdots & & \vdots \\ x_1^{(N)} & \cdots & x_p^{(N)} \end{bmatrix} \right\}$, we derive the least squares estimate $\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$, so that for a new test point $\mathbf{x}^{(0)T} = (x_{01}, ..., x_{0p})$ we predict $y^{(0)}$ by $\hat{y}^{(0)} = \hat{f}(\mathbf{x}^{(0)}) = \mathbf{x}^{(0)}\hat{\beta} = \mathbf{x}^{(0)T}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$. Consider the expected prediction error:

$$\begin{aligned} Err &= \mathbb{E}_{\mathbf{x}^{(0)},y^{(0)},T}(y^{(0)} - \hat{f}(\mathbf{x}^{(0)}))^2 \\ &= \mathbb{E}_{\mathbf{x}^{(0)}}\mathbb{E}_{y^{(0)}|\mathbf{x}^{(0)},T}(y^{(0)} - \hat{f}(\mathbf{x}^{(0)}))^2 \quad . \end{aligned} \tag{1}$$

Derive an expression for the expected prediction error at a specific input point $\mathbf{x}^{(0)}$ in terms of the bias and variance of $\hat{f}$, i.e. respectively, $\mathbb{E}(\hat{f}) - f$, and $\mathbb{E}(\hat{f} - \mathbb{E}\hat{f})^2$.

$$\begin{aligned} Err &= \mathbb{E}_{\boldsymbol{x}^{(0)},y^{(0)},T}\left[(y^{(0)} - \hat{f}(\boldsymbol{x}^{(0)}))^2\right] \\ &= \mathbb{E}_{\boldsymbol{x}^{(0)},y^{(0)},T}\left[(f(\boldsymbol{x}^{(0)}) + \epsilon - \hat{f}(\boldsymbol{x}^{(0)}))^2\right] \\ &= \mathbb{E}_{\boldsymbol{x}^{(0)},y^{(0)},T}\left[\left(\hat{f}(\boldsymbol{x}^{(0)}) - f(\boldsymbol{x}^{(0)})\right)^2 + \epsilon^2 + 2\epsilon\left(f(\boldsymbol{x}^{(0)}) - \hat{f}(\boldsymbol{x}^{(0)})\right)\right] \end{aligned}$$

Note that $\mathbb{E}_{\boldsymbol{x}^{(0)},y^{(0)},T}[\epsilon^2] = \mathbb{E}_{T|\boldsymbol{x}^{(0)},y^{(0)}}[\mathbb{E}_{y^{(0)}|\boldsymbol{x}^{(0)}}[\mathbb{E}_{\boldsymbol{x}^{(0)}}[\epsilon^2]]] = \mathbb{E}_{T|\boldsymbol{x}^{(0)},y^{(0)}}[\mathbb{E}_{y^{(0)}|\boldsymbol{x}^{(0)}}[\epsilon^2]] = \mathbb{E}_{T|\boldsymbol{x}^{(0)},y^{(0)}}[\sigma_\epsilon^2 + 0^2] = \sigma_\epsilon^2$ and $\mathbb{E}_{\boldsymbol{x}^{(0)},y^{(0)},T}\left[2\epsilon\left(f(\boldsymbol{x}^{(0)}) - \hat{f}(\boldsymbol{x}^{(0)})\right)\right] = 2\,\mathbb{E}_{T|\boldsymbol{x}^{(0)},y^{(0)}}[\mathbb{E}_{y^{(0)}|\boldsymbol{x}^{(0)}}[\mathbb{E}_{\boldsymbol{x}^{(0)}}[\epsilon]]] \times \mathbb{E}_{\boldsymbol{x}^{(0)},T}\left[f(\boldsymbol{x}^{(0)}) - \hat{f}(\boldsymbol{x}^{(0)})\right] = 2\,\mathbb{E}_{T|\boldsymbol{x}^{(0)},y^{(0)}}[\mathbb{E}_{y^{(0)}|\boldsymbol{x}^{(0)}}[\epsilon]] \times \mathbb{E}_{\boldsymbol{x}^{(0)},T}\left[f(\boldsymbol{x}^{(0)}) - \hat{f}(\boldsymbol{x}^{(0)})\right] = 2 \times 0 \times \mathbb{E}_{\boldsymbol{x}^{(0)},T}\left[f(\boldsymbol{x}^{(0)}) - \hat{f}(\boldsymbol{x}^{(0)})\right] = 0$.
By linearity of expectation,

$$\begin{aligned} Err &= \mathbb{E}_{\boldsymbol{x}^{(0)},y^{(0)}}\left[\mathbb{E}_T\left[\left(\hat{f}(\boldsymbol{x}^{(0)}) - f(\boldsymbol{x}^{(0)})\right)^2\right]\right] + \sigma_\epsilon^2 + 0 \\ &= \mathbb{E}_{y^{(0)}|\boldsymbol{x}^{(0)}}\mathbb{E}_{\boldsymbol{x}^{(0)}}\left[\mathbb{E}_T\left[\hat{f}^2(\boldsymbol{x}^{(0)}) + f^2(\boldsymbol{x}^{(0)}) - 2\hat{f}(\boldsymbol{x}^{(0)})f(\boldsymbol{x}^{(0)})\right]\right] + \sigma_\epsilon^2 \\ &= \mathbb{E}_{\boldsymbol{x}^{(0)}}\left[\mathbb{E}_T\left[\hat{f}^2(\boldsymbol{x}^{(0)})\right] + f^2(\boldsymbol{x}^{(0)}) - 2f(\boldsymbol{x}^{(0)})\mathbb{E}_T\left[\hat{f}(\boldsymbol{x}^{(0)})\right]\right] + \sigma_\epsilon^2 \\ &= \mathbb{E}_{\boldsymbol{x}^{(0)}}\left[Var_T\left[\hat{f}(\boldsymbol{x}^{(0)})\right] + \left(\mathbb{E}_T\left[\hat{f}(\boldsymbol{x}^{(0)})\right]\right)^2 + f^2(\boldsymbol{x}^{(0)}) - 2f(\boldsymbol{x}^{(0)})\mathbb{E}_T\left[\hat{f}(\boldsymbol{x}^{(0)})\right]\right] + \sigma_\epsilon^2 \\ &= \mathbb{E}_{\boldsymbol{x}^{(0)}}\left[Var_T\left[\hat{f}(\boldsymbol{x}^{(0)})\right] + \left(\mathbb{E}_T\left[\hat{f}(\boldsymbol{x}^{(0)})\right] - f(\boldsymbol{x}^{(0)})\right)^2\right] + \sigma_\epsilon^2 \\ &= \mathbb{E}_{\boldsymbol{x}^{(0)}}\left[\mathbb{E}_T\left[\left(\hat{f}(\boldsymbol{x}^{(0)}) - \mathbb{E}_T\left[\hat{f}(\boldsymbol{x}^{(0)})\right]\right)^2\right] + \left(\mathbb{E}_T\left[\hat{f}(\boldsymbol{x}^{(0)})\right] - f(\boldsymbol{x}^{(0)})\right)^2\right] + \sigma_\epsilon^2 \end{aligned}$$

Thus, for a specific input point $\boldsymbol{x}^{(0)}$, we have $Err = \mathbb{E}_T\left[\left(\hat{f}(\boldsymbol{x}^{(0)}) - \mathbb{E}_T\left[\hat{f}(\boldsymbol{x}^{(0)})\right]\right)^2\right] + \left(\mathbb{E}_T\left[\hat{f}(\boldsymbol{x}^{(0)})\right] - f(\boldsymbol{x}^{(0)})\right)^2 + \sigma_\epsilon^2$, which is variance of $\hat{f}$ + (bias of $\hat{f}$)^2 + variance of noise.

2) Weighted Linear Regression (8 points)

Consider a linear regression problem in which we want to "weigh" different training examples differently. Specifically, suppose we want to minimize

$$J(\theta) = \frac{1}{2}\sum_{i=1}^{N} w^{(i)}\left(\theta^T \mathbf{x}^{(i)} - y^{(i)}\right)^2 \quad . \tag{2}$$

(1) (2 points) Show that $J(\theta)$ can be re-written as:

$$J(\theta) = (\mathbf{X}\theta - \mathbf{y})^T \mathbf{W}(\mathbf{X}\theta - \mathbf{y}) \tag{3}$$

for a diagonal matrix $\mathbf{W}$ of appropriate dimensions, and $\mathbf{X}$ and $\mathbf{y}$ as defined in Question 1. Start by defining $\mathbf{W}$.

Let $\boldsymbol{W}$ be a $N \times N$ diagonal matrix, and $\boldsymbol{W}[i,i] = \frac{w^{(i)}}{2}$ for $i \in \{1, \dots, N\}$.

$$\begin{aligned}
J(\theta) &= (\boldsymbol{X}\theta - \boldsymbol{y})^T \boldsymbol{W}(\boldsymbol{X}\theta - \boldsymbol{y}) \\
&= \left[\theta^T \boldsymbol{x}^{(1)} - y^{(1)}, \dots, \theta^T \boldsymbol{x}^{(N)} - y^{(N)}\right]^T \boldsymbol{W}(\boldsymbol{X}\theta - \boldsymbol{y}) \\
&= \left[\frac{w^{(1)}}{2}(\theta^T \boldsymbol{x}^{(1)} - y^{(1)}), \dots, \frac{w^{(N)}}{2}(\theta^T \boldsymbol{x}^{(N)} - y^{(N)})\right]^T (\boldsymbol{X}\theta - \boldsymbol{y}) \\
&= \sum_{i=1}^{N} \frac{w^{(i)}}{2}(\theta^T \boldsymbol{x}^{(i)} - y^{(i)})^2 = \frac{1}{2}\sum_{i=1}^{N} w^{(i)}(\theta^T \boldsymbol{x}^{(i)} - y^{(i)})^2 \quad \text{as desired.}
\end{aligned}$$

(2) (3 points) If all the $w^{(i)}$'s equal 1, then the normal equation is $\mathbf{X}^T\mathbf{X}\theta = \mathbf{X}^T\mathbf{y}$, and the value of θ that minimizes $J(\theta)$ is given by $(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$. By finding the derivative $\nabla_\theta J(\theta)$ and setting that to zero, generalize the normal equation to this weighted setting, and give the new value of θ that minimizes $J(\theta)$ as a function of $\mathbf{X}$, $\mathbf{W}$ and $\mathbf{y}$.

By calculating $\nabla_\theta J(\theta)$, we get

$$\nabla_\theta J(\theta) = \frac{\partial}{\partial\theta}((\boldsymbol{X}\theta - \boldsymbol{y})^T \boldsymbol{W}(\boldsymbol{X}\theta - \boldsymbol{y})) = 2\boldsymbol{X}^T\boldsymbol{W}(\boldsymbol{X}\theta - \boldsymbol{y})$$

Set $\nabla_\theta J(\theta) = 0$. Then

$$\begin{aligned}
2\boldsymbol{X}^T\boldsymbol{W}(\boldsymbol{X}\theta - \boldsymbol{y}) &= \mathbf{0} \\
\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}\theta &= \boldsymbol{X}^T\boldsymbol{W}\boldsymbol{y} \\
\theta &= \left(\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{y}
\end{aligned}$$

Since $\boldsymbol{W}$ is positive definite, then $\theta = \left(\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{y}$ minimizes $J(\theta)$.
Note that when the weights are 1, $\boldsymbol{W} = \boldsymbol{I}$, and $\theta = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$.

(3) (3 points) Suppose we have a training set $(\mathbf{x}^{(i)}, y^{(i)}); i = 1..., N$ of N independent examples, in which the $y^{(i)}$'s have different variances. Specifically, suppose

$$p(y^{(i)}|\mathbf{x}^{(i)};\theta) = \frac{1}{\sqrt{2\pi}\sigma^{(i)}}\exp\Big(-\frac{(y^{(i)}-\theta^T x^{(i)})^2}{2(\sigma^{(i)})^2}\Big) \tag{4}$$

for fixed, known $\sigma^{(i)}$'s. Show that finding the maximum likelihood estimate of θ reduces to solving a weighted linear problem. State what the $w^{(i)}$'s are in terms of the $\sigma^{(i)}$'s.

Let Σ be a $n \times n$ diagonal matrix with $\Sigma[i,i] = (\sigma^{(i)})^2$ for $i \in \{1, \dots, N\}$.

$$\begin{aligned} p(\boldsymbol{y}|\boldsymbol{X};\theta) &= \Pi_{i=1}^N p(y^{(i)}|\boldsymbol{x}^{(i)};\theta) \\ &= \Pi_{i=1}^N \frac{1}{\sqrt{2\pi}\sigma^{(i)}}\exp\Big(-\frac{(y^{(i)}-\theta^T\boldsymbol{x}^{(i)})^2}{2(\sigma^{(i)})^2}\Big) \\ &= \frac{1}{\sqrt{det(2\pi\Sigma)}}\exp\Big(-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{X}\theta)^T\Sigma^{-1}(\boldsymbol{y}-\boldsymbol{X}\theta)\Big) \end{aligned}$$

Thus, $\boldsymbol{y} \sim MN(\boldsymbol{X}\theta, \Sigma)$ follows a multivariate normal distribution.
The log likelihood is $\mathcal{L}(\theta) = log\left(\frac{1}{\sqrt{det(2\pi\Sigma)}}\right) - \frac{1}{2}(\boldsymbol{y}-\boldsymbol{X}\theta)^T\Sigma^{-1}(\boldsymbol{y}-\boldsymbol{X}\theta)$.
For $L'(\theta) = -\boldsymbol{X}^T\Sigma^{-1}(\boldsymbol{y}-\boldsymbol{X}\theta)$, we set $L'(\theta) = 0$ and get

$$\begin{aligned} \boldsymbol{X}^T\Sigma^{-1}(\boldsymbol{y}-\boldsymbol{X}\theta) &= 0 \\ \boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X}\theta &= \boldsymbol{X}^T\Sigma^{-1}\boldsymbol{y} \\ \theta &= \left(\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{y} \end{aligned}$$

Thus, the maximum likelihood estimator is $\theta_{MLE} = \left(\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{y}$. By comparing with the result from part (2), we conclude that $\Sigma^{-1} = \boldsymbol{W}$, which indicates that $w^{(i)} = \frac{1}{\sigma^2_{(i)}}$.

3) Regularization (5 points)

(1) (2 points) Consider the following two regression models: (i) Ridge regression, (ii) Lasso regression. For each model, describe a data setting for which you would want to use that regression model.

(i) Ridge regression
Data setting: We want to study how social, economic, and health factors can influence the COVID mortality rate in N counties. We have $\boldsymbol{y} \in \mathbb{R}^{N\times 1}$ containing the mortality rates and $\boldsymbol{X} \in \mathbb{R}^{M\times N}$, where each column represents the covariates for one county. Our goal is to estimate the effect of each covariate on the mortality rate while avoid overfitting.
Reasoning: We punish on the L2-norm of the weight $\boldsymbol{w} \in \mathbb{R}^{M\times 1}$ to avoid overfitting and minimize $||\boldsymbol{X}^T\boldsymbol{w} - \boldsymbol{y}||_2^2 + \lambda||\boldsymbol{w}||_2^2$. Since the L2-norm is quadratic, its geometric representation is a high dimensional "ball" with smooth gradient near the axis. The optimal solution is where the lowest level set of MSE intersects with the L2 constraint. On the smooth surface of L2-norm, the intersection is likely to occur at any place. Since the constraint favors small coefficients, the model will also favor small coefficients.
(ii) Lasso regression
Data setting: The data is the same as above. This time, we want to find the covariates that have strong influence on the mortality rate and ignore the others. We also want to know the effect these covariates have on the response variable while avoid overfitting.
Reasoning: When minimizing $||\boldsymbol{X}^T\boldsymbol{w} - \boldsymbol{y}||_2^2 + \lambda||\boldsymbol{w}||_1$, we note that the L1-norm is not smooth because it contains points on the axis which is not differentiable. Moreover, rather than decreasing, its gradient stays as a constant as it approaches the axis. Since L1-norm contour looks like a high dimensional "diamond" with corners on the axis, the intersection between level sets of MSE and L1-norm constraints is likely to occur on the axis, turning some coefficients to 0. Thus, the sparse result enables us to perform feature selection.

(2) (2 points) Describe how we could use cross-validation to select the optimal regression penalty parameter λ.

In a k-fold cross validation, we randomly split the training data into k parts $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_k$. For a particular λ, $\forall i \in \{1, \ldots, k\}$, we obtain the $error_i$ by training on $\boldsymbol{X} \setminus \boldsymbol{X}_i$ and testing on $\boldsymbol{X}_i$. Set $\frac{1}{k}\sum_{i=1}^{k} error_i$ to be the error of the λ. After repeating the procedure for all λs, we obtain the optimal regression penalty parameter by choosing the one with the lowest error.

(3) (1 point) When might you prefer to use 10-fold cross-validation instead of a train/dev/test validation setting.

When sample size is small, cross validation can use as many data as possible for training to increase training accuracy. It also enlarges the validation set to obtain a more reliable validation result. Train and dev are not shared in the train/dev/test validation setting.