MATH-GA 2840 HW#3

Yifei(Fahy) Gao yg1753

1. (Augmented dataset) Ridge regression is equivalent to applying OLS on an expanded dataset that has additional examples. Describe these additional examples in detail. Intuitively, what effect do these additional examples have?

OLS cost function is:

$$\arg\min_{\beta} \left\|\tilde{y}_{\text{train}} - X^T\beta\right\|_2^2 = \arg\min_{\beta} (\beta - \beta_{\text{true}})^T XX^T (\beta - \beta_{\text{true}}) - 2\tilde{z}_{\text{train}}^T X^T\beta$$

Ridge regression cost function is:

$$\arg\min_{\beta} \left\|\tilde{y}_{\text{train}} - X^T\beta\right\|_2^2 + \lambda\|\beta\|_2^2 = \arg\min_{\beta} (\beta - \beta_{\text{true}})^T XX^T (\beta - \beta_{\text{true}}) + \lambda\beta^T\beta - 2\tilde{z}_{\text{train}}^T X^T\beta$$

So the ridge has the extra $\lambda\beta^T\beta$ term comparing to the OLS cost function.

And the quadratic form of OLS is (let $\beta = \begin{bmatrix} a \\ b \end{bmatrix}, \beta_{true} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, X = \begin{bmatrix} 1 & 0 \\ 0 & 0.01 \end{bmatrix}$), then:

$$c^2 = \arg\min_{\beta} (\beta - \beta_{\text{true}})^T XX^T (\beta - \beta_{\text{true}})$$

$$= \begin{bmatrix} a-1 \\ b-1 \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & 0.01 \end{bmatrix} \begin{bmatrix} a-1 \\ b-1 \end{bmatrix} = (a-1)^2 + \frac{(b-1)^2}{100}$$

Then the minima for OLS is:

$$\beta_{\text{OLS}} = \left(XX^T\right)^{-1} X\tilde{y}_{\text{train}}$$
$$= \beta_{\text{true}} + US^{-1}V^T\tilde{z}_{\text{train}}$$

So it is Gaussian with the mean equal: $\beta_{true} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ And for the quadratic form of Ridge regression:

$$c^2 = \arg\min_{\beta} (\beta - \beta_{\text{true}})^T XX^T (\beta - \beta_{\text{true}}) + \lambda\beta^T\beta$$

$$= \begin{bmatrix} a-1 \\ b-1 \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & 0.01 \end{bmatrix} \begin{bmatrix} a-1 \\ b-1 \end{bmatrix} + \lambda a^2 + \lambda b^2$$

(noted that $\lambda a^2 + \lambda b^2$ is a spherical equation)

continue the simplication:

$$f(c, \lambda) = (1+\lambda)(a - \frac{1}{1+\lambda})^2 + \frac{1+100\lambda}{100}(b - \frac{1}{1+100\lambda})^2$$

So when λ closes to 0, $f(c, \lambda) = \beta_{OLS}$(ellipse), when λ close to ∞, $f(c, \lambda) = 0$ (circle).

And the minima for Ridge Regression is:

$$\beta_{\text{RR}} = \left(XX^T + \lambda I\right)^{-1} X\left(X^T\beta_{\text{true}} + \tilde{z}_{\text{train}}\right) = \begin{bmatrix} \frac{1}{1+\lambda} \\ \frac{0.01}{0.01+\lambda} \end{bmatrix} + \begin{bmatrix} \frac{1}{1+\lambda} & 0 \\ 0 & \frac{1}{0.01+\lambda} \end{bmatrix} X\tilde{z}_{\text{train}}$$

So it is Gaussian with the mean equal $\begin{bmatrix} \frac{1}{1+\lambda} \\ \frac{0.01}{0.01+\lambda} \end{bmatrix}$, which is much less than the mean of OLS.

The effect of adding the extra term is to do regulation with much less variance and to avoid the overfitting situation that we do not have enough data but too many noises comparing to OLS.

2. (Correlated features) Consider a regression problem where the response only depends on one feature, but we don't know it, so we incorporate an additional feature into the model that happens to be very correlated with the first feature. More specifically, let $y \in \mathbb{R}^n$ be defined by

$$y := \beta_{\text{true}}\, w_1 + z$$

where $\beta_{\text{true}} \in \mathbb{R}$ is the true coefficient, $w_1 \in \mathbb{R}^n$ is the first feature vector, and $z \in \mathbb{R}^n$ is additive noise. The second feature vector is given by $w_2 \in \mathbb{R}^n$ and can be decomposed into

$$w_2 = \alpha w_1 + \sqrt{1-\alpha^2}\, w_\perp$$

where $w_\perp$ is orthogonal to w_1. The vectors $w_1, w_2, w_\perp$ and z all have unit ℓ_2 norm. In addition, we assume

$$w_1^T z = 0.1$$
$$w_\perp^T z = 0.1$$

We fit a linear regression model to y using the feature matrix

$$X = \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix}$$

(a) What does the OLS estimator of the coefficients β_{OLS} equal to when $\alpha \to 1$? Explain what is happening. Hint: Use the fact that for any $a, b, c,$ and d such that $ad \neq bc$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Since we know

$$\begin{aligned} \beta_{\text{OLS}} &:= \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - x_i^T \beta\right)^2 \\ &= \left(XX^T\right)^{-1} Xy \end{aligned} \tag{Theorem 2.1}$$

Then:

$$\begin{aligned} \beta_{\text{OLS}} &= (\begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix} \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix}^T)^{-1} \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix} y \\ &= \begin{bmatrix} w_1^T w_1 & w_1^T w_2 \\ w_2^T w_1 & w_2^T w_2 \end{bmatrix}^{-1} \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix} y \end{aligned}$$

from the hint, we get:

$$\beta_{\text{OLS}} = \begin{bmatrix} 1 & w_1^T w_2 \\ w_2^T w_1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix} y = \frac{1}{1 - w_1^T w_2 w_2^T w_1} \begin{bmatrix} 1 & -w_1^T w_2 \\ -w_2^T w_1 & 1 \end{bmatrix} \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix} y$$

Since $w_2 = \alpha w_1 + \sqrt{1-\alpha^2} w_\perp$, $y := \beta_{true} w_1 + z$, $w_1^T z = 0.1 \quad w_\perp^T z = 0.1$ and $w_\perp w_1^T = w_\perp^T w_1 = 0$,

$$\begin{aligned} \beta_{\text{OLS}} &= \frac{1}{1 - w_1^T(\alpha w_1 + \sqrt{1-\alpha^2} w_\perp)(\alpha w_1 + \sqrt{1-\alpha^2} w_\perp)^T w_1} \begin{bmatrix} 1 & -w_1^T(\alpha w_1 + \sqrt{1-\alpha^2} w_\perp) \\ -(\alpha w_1 + \sqrt{1-\alpha^2} w_\perp)^T w_1 & 1 \end{bmatrix} \begin{bmatrix} w_1^T \\ (\alpha w_1 + \sqrt{1-\alpha^2} w_\perp)^T \end{bmatrix} (\beta_{true} w_1 + z) \\ &= \frac{1}{1-\alpha^2} \begin{bmatrix} 1 & -\alpha \\ -\alpha & 1 \end{bmatrix} \begin{bmatrix} w_1^T \\ (\alpha w_1 + \sqrt{1-\alpha^2} w_\perp)^T \end{bmatrix} (\beta_{true} w_1 + z) \\ &= \frac{1}{1-\alpha^2} \begin{bmatrix} w_1^T - \alpha(\alpha w_1^T + \sqrt{1-\alpha^2} w_\perp) \\ -\alpha w_1^T + \alpha w_1^T + \sqrt{1-\alpha^2} w_\perp \end{bmatrix} (\beta_{true} w_1 + z) \\ &= \frac{1}{1-\alpha^2} \begin{bmatrix} \beta_{true} + 0.1 - \alpha^2 \beta_{true} - 0.1\alpha^2 - \beta_{true}\sqrt{1-\alpha^2} w_\perp^T w_1 - 0.1\alpha\sqrt{1-\alpha^2} \\ \beta_{true}\sqrt{1-\alpha^2} w_\perp^T w_1 + \sqrt{1-\alpha^2} w_\perp^T z \end{bmatrix} \\ &= \frac{1}{1-\alpha^2} \begin{bmatrix} (1-\alpha^2)\beta_{true} + 0.1(1-\alpha^2) - 0.1\alpha\sqrt{1-\alpha^2} \\ 0.1\sqrt{1-\alpha^2} \end{bmatrix} \\ \beta_{OLS} &= \begin{bmatrix} \beta_{true} + 0.1 - \frac{0.1\alpha}{\sqrt{1-\alpha^2}} \\ \frac{0.1}{\sqrt{1-\alpha^2}} \end{bmatrix} \end{aligned}$$

Therefore when a is approaching to a, the β_{OLS} will be infinite, becuase the denominator of $\frac{0.1}{\sqrt{1-\alpha^2}}$ will be very close to zero.

(b) What does the corresponding estimate of the response $y_{\text{OLS}} := X^T \beta_{\text{OLS}}$ equal to when $\alpha \to 1$? Is it collinear with the true feature w_1 when $\alpha \to 1$? Explain what is happening.

$$\begin{aligned} y_{\text{OLS}} := X^T \beta_{\text{OLS}} &= \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix}^T \beta_{OLS} = \begin{bmatrix} w_1 & w_2 \end{bmatrix} \beta_{OLS} \\ &= \begin{bmatrix} w_1 & \alpha w_1 + \sqrt{1-\alpha^2} w_\perp \end{bmatrix} \begin{bmatrix} \beta_{true} + 0.1 - \frac{0.1\alpha}{\sqrt{1-\alpha^2}} \\ \frac{0.1}{\sqrt{1-\alpha^2}} \end{bmatrix} \\ &= \beta_{true} w_1 + 0.1 w_1 - \frac{0.1\alpha w_1}{\sqrt{1-\alpha^2}} + \frac{0.1\alpha w_1}{\sqrt{1-\alpha^2}} + 0.1 w_\perp \\ &= \beta_{true} w_1 + 0.1(w_1 + w_\perp) \end{aligned}$$

Therefore y_{OLS} is not collinear with the feature w_1 when $\alpha \to 1$, since there exits $w_\perp$ in the expression of y_{OLS}. The reason is that there are some noise in the same direction of $w_\perp$

(c) What does the ridge regression estimator of the coefficients β_{RR} equal to when $\alpha \to 1$ and the regularization parameter $\lambda > 0$ is fixed? Describe the difference with the OLS estimate.

Based on the *regulation lecture notes*:

$$\beta_{\text{RR}} = \left(XX^T + \lambda I\right)^{-1} Xy \tag{Theorem 1.2}$$

Since we have already calculate XX^T from the part a, then:

$$\beta_{RR} = \begin{bmatrix} 1+\lambda & w_1^T w_2 \\ w_2^T w_1 & 1+\lambda \end{bmatrix}^{-1} \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix} y$$

$$= \begin{bmatrix} 1+\lambda & \alpha \\ \alpha & 1+\lambda \end{bmatrix}^{-1} \begin{bmatrix} w_1^T \\ (\alpha w_1 + \sqrt{1-\alpha^2} w_\perp)^T \end{bmatrix} (\beta_{\text{true}}\, w_1 + z)$$

$$= \frac{1}{1+\lambda^2+2\lambda-\alpha^2} \begin{bmatrix} 1+\lambda & -\alpha \\ -\alpha & 1+\lambda \end{bmatrix} \begin{bmatrix} w_1^T \\ (\alpha w_1 + \sqrt{1-\alpha^2} w_\perp)^T \end{bmatrix} (\beta_{\text{true}}\, w_1 + z)$$

$$= \frac{1}{(1+\lambda)^2-\alpha^2} \begin{bmatrix} 1+\lambda & -\alpha \\ -\alpha & 1+\lambda \end{bmatrix} \begin{bmatrix} \beta_{\text{true}} + 0.1 \\ \alpha\beta_{\text{true}} + 0.1\left(\alpha + \sqrt{1-\alpha^2}\right) \end{bmatrix}$$

$$= \frac{1}{(1+\lambda)^2-\alpha^2} \begin{bmatrix} (1+\lambda)(\beta_{\text{true}} + 0.1) - \alpha(\alpha\beta_{\text{true}} + 0.1\left(\alpha + \sqrt{1-\alpha^2}\right)) \\ -\alpha(\beta_{\text{true}} + 0.1) + (1+\lambda)(\alpha\beta_{\text{true}} + 0.1\left(\alpha + \sqrt{1-\alpha^2}\right)) \end{bmatrix}$$

Then if a is close to 1, then the equation will be:

$$= \frac{1}{(1+\lambda)^2-1} \begin{bmatrix} (1+\lambda)(\beta_{\text{true}} + 0.1) - (\beta_{\text{true}} + 0.1\left(1 + \sqrt{1-1}\right)) \\ -(\beta_{\text{true}} + 0.1) + (1+\lambda)(\beta_{\text{true}} + 0.1\left(1 + \sqrt{1-1}\right)) \end{bmatrix}$$

$$= \frac{1}{\lambda^2+2\lambda} \begin{bmatrix} \lambda(\beta_{\text{true}} + 0.1) \\ \lambda(\beta_{\text{true}} + 0.1) \end{bmatrix} = \begin{bmatrix} \frac{\beta_{\text{true}}+0.1}{\lambda+2} \\ \frac{\beta_{\text{true}}+0.1}{\lambda+2} \end{bmatrix}$$

Comparing to the part a), we can see that β_{RR} is independent from α but λ, and the front part of β_{RR}:$\left(XX^T + \lambda I\right)^{-1}$ is always invertible.

(d) What does the corresponding estimate of the response $y_{\text{RR}} := X^T \beta_{\text{RR}}$ equal to when $\alpha \to 1$? Is it collinear with the true feature w_1?

$$y_{\text{RR}} := X^T \beta_{\text{RR}} = \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix}^T \beta_{RR} = \begin{bmatrix} w_1 & w_2 \end{bmatrix} \beta_{RR}$$

$$= \begin{bmatrix} w_1 & \alpha w_1 + \sqrt{1-\alpha^2} w_\perp \end{bmatrix} \begin{bmatrix} \frac{\beta_{\text{true}}+0.1}{\lambda+2} \\ \frac{\beta_{\text{true}}+0.1}{\lambda+2} \end{bmatrix}$$

As a close to 1,

$$\lim_{\alpha\to 1} y_{\text{RR}} = \lim_{\alpha\to 1}(\begin{bmatrix} w_1 & \alpha w_1 + \sqrt{1-\alpha^2} w_\perp \end{bmatrix} \begin{bmatrix} \frac{\beta_{\text{true}}+0.1}{\lambda+2} \\ \frac{\beta_{\text{true}}+0.1}{\lambda+2} \end{bmatrix})$$

$$= \begin{bmatrix} w_1 & w_1 \end{bmatrix} \begin{bmatrix} \frac{\beta_{\text{true}}+0.1}{\lambda+2} \\ \frac{\beta_{\text{true}}+0.1}{\lambda+2} \end{bmatrix} \qquad \text{(product rule of limit)}$$

$$= \frac{2}{\lambda+2}(\beta_{\text{true}} + 0.1) w_1$$

Therefore, we can say that y_{RR} is collinear with the true feature w_1, since it only contains w_1 unlike y_{OLS}

3. (Prior knowledge) Consider a linear regression problem where we have prior information indicating that the coefficients should be close to a certain value β_{prior}.

(a) How can you incorporate this prior knowledge if you are using ridge regression? Write the corresponding optimization problem.

Based on the *regularization lecture notes*, the Ridge Regression estimator is

$$\beta_{\text{RR}} := \arg\min_\beta \left\| y - X^T\beta \right\|_2^2 + \lambda \|\beta\|_2^2 \qquad \text{(Definition 1.1)}$$

We can try to incorporate the β_{prior} into the λ norm term, so that:

$$\beta_{RR} = \arg\min_\beta \left\| y - X^T\beta \right\|_2^2 + \lambda \|\beta - \beta_{prior}\|_2^2$$

By decomposition, we get:

$$\beta_{RR} = y^T y - 2X^T\beta y + X^T\beta\beta^T X + \lambda\beta^T\beta - 2\lambda\beta\beta_{prior} + \lambda\beta_{prior}^T\beta_{prior}$$

Then differentiate the whole equation by β and let $\frac{\partial\beta_{RR}}{\partial\beta} = 0$, then

$$\nabla f(\beta) = 2X^T X\beta - 2X^T y + 2\lambda\beta - 2\lambda\beta_{\text{prior}}$$
$$0 = -2X^T y + 2X^T X\beta + 2\lambda\beta - 2\lambda\beta_{prior}$$
$$0 = (-X^T X - \lambda)\beta + X^T y + \lambda\beta_{prior}$$
$$\beta_{RR} = (XX^T + \lambda)^{-1}(Xy + \lambda\beta_{prior})$$

(b) Assume that the data are generated according to a linear model $\tilde{y} := X^T \beta_{\text{true}} + \tilde{z}$, where $\beta_{\text{true}} \in \mathbb{R}^p$ and $X \in \mathbb{R}^{p \times n}$ are fixed and $\tilde{z}$ is an iid Gaussian random vector with zero mean and variance σ^2. Does the modification change the mean or the covariance matrix of the coefficient estimate with respect to the ridge-regression estimate? If so, report the new value.

The Mean will change but the covariance matrix will not.

By Singular-value-decomposition, we can let $X = USV^T$ to be the SVD of the feature matrix, then:

$$\begin{aligned}
\beta_{RR} &= (X^T X + \lambda)^{-1}(Xy + \lambda\beta_{prior}) \\
&= (XX^T + \lambda)^{-1}(X(X^T\beta_{\text{true}} + \tilde{z}) + \lambda\beta_{prior}) \\
&= (US^2U^T + \lambda)^{-1}(US^2U^T\beta_{\text{true}} + USV^T\tilde{z} + \lambda\beta_{prior}) \\
&= (U(S^2 + \lambda)^{-1}U^T)(US^2U^T\beta_{\text{true}} + USV^T\tilde{z} + \lambda\beta_{prior}) \\
&= U\left(S^2 + \lambda \mathrm{Id}_{\mathrm{p}}\right)^{-1} S^2U^T\beta_{\text{true}} + U\left(S^2 + \lambda \mathrm{Id}_{\mathrm{p}}\right)^{-1} SV^T\tilde{z} + U\left(S^2 + \lambda \mathrm{Id}_{\mathrm{p}}\right)^{-1} U^T\lambda\beta_{\text{prior}}
\end{aligned}$$

Therefore,

$$E[\beta_{RR}] = \sum_{j=1}^{p} \frac{s_j^2 \left\langle u_j, \beta_{\text{true}} \right\rangle}{s_j^2 + \lambda} u_j + \lambda \sum_{j=1}^{p} \frac{\left\langle u_j, \beta_{\text{prior}} \right\rangle}{s_j^2 + \lambda} u_j$$

And the original one is:

$$E[\beta_{OLS}] = \sum_{j=1}^{p} \frac{s_j^2 \left\langle u_j, \beta_{\text{true}} \right\rangle}{s_j^2 + \lambda} u_j$$

And the covariance will not change, which is still the same as:

$$\mathrm{Cov}(\beta_{\mathrm{RR}}) = \left(XX^T + \lambda\right)_1^{-1} X\sigma^2 \left(\left(XX^T + \lambda\right)^{-1} X\right)^T = \sigma^2 U \,\mathrm{diag}_{j=1}^{p}\left(\frac{s_j^2}{\left(s_j^2 + \lambda\right)^2}\right) U^T$$

(c) How can you incorporate this prior knowledge if you are using gradient descent with early stopping? Write the corresponding update equation as a function of β_{prior} .

let new $\nabla f(\beta)$ equal to : $X^T X\beta - X^T y + \lambda\beta - \lambda\beta_{\text{prior}}$

Then:

$$\begin{aligned}
\beta^{k+1} &= \beta^k - \alpha(X^T X\beta^k - Xy + \lambda\beta^k - \lambda\beta_{prior}) \\
\beta^{k+1} &= (I - \alpha X^T X - \alpha\lambda)\beta^k + \alpha(Xy + \lambda\beta_{prior})
\end{aligned}$$

since $\beta^0 = \beta_{prior}$

$$\beta^{(k+1)} = \left(I - \alpha XX^T\right)^{k+1}\beta_{prior} + \sum_{i=0}^{k}\left(I - \alpha XX^T\right)^i \alpha(Xy + \lambda\beta_{prior}) \qquad \text{(Theorem 2.1)}$$

$$\beta^{k+1} = \sum_{i=0}^{k}\left(I - \alpha XX^T\right)^i \alpha\left(Xy + \lambda\beta_{\text{prior}}\right) \qquad \text{(from the early stopping video)}$$

Since from the part b, $X = USV^T$, $I = UU^T$,

$$\begin{aligned}
\beta^{k+1} &= \textstyle\sum_{i=0}^{k}\left(UU^T - \alpha US^2U^T\right)^i \alpha\left(USV^T y + \lambda\beta_{\text{prior}}\right) \\
\beta^{k+1} &= \alpha\textstyle\sum_{i=0}^{k}\left(U\left(I - \alpha S^2\right)U^T\right)^i\left(USV^T y + \lambda\beta_{\text{prior}}\right) \\
\beta^{k+1} &= \alpha\textstyle\sum_{i=0}^{k} U\left(I - \alpha S^2\right)^i\left(SV^T y + U^T\lambda\beta_{\text{prior}}\right) \\
\beta^{k+1} &= \alpha U\,\mathrm{diag}_{j=1}^{p}\left(\tfrac{1-\left(1-\alpha s_j^2\right)^{k+1}}{\alpha s_j^2}\right)\left(SV^T y + U^T\lambda\beta_{\text{prior}}\right)
\end{aligned}$$

(d) Assume that the data are generated according to the linear model described above. Does the modification change the mean or the covariance matrix of the coefficient estimate with respect to the gradient descent estimate initialized at the origin? If so, report the new value.

Same the as the question (b), the mean will change bu the covariance matrix will not.

$$E[\beta^{k+1}] = E[\alpha U\,\mathrm{diag}_{j=1}^{p}\left(\frac{1-\left(1-\alpha s_j^2\right)^{k+1}}{\alpha s_j^2}\right)\left(SV^T + U^T\lambda\beta_{\text{prior}}\right)]$$

Since $y = X^T\beta_{true} + \tilde{z}$, then

$$\begin{aligned}
E[\beta^{k+1}] &= E[\alpha U\,\mathrm{diag}_{j=1}^{p}\left(\frac{1-\left(1-\alpha s_j^2\right)^{k+1}}{\alpha s_j^2}\right)\left(SV^T(X^T\beta_{true} + \tilde{z}) + U^T\lambda\beta_{\text{prior}}\right)] \\
&= E[\alpha U\,\mathrm{diag}_{j=1}^{p}\left(\frac{1-\left(1-\alpha s_j^2\right)^{k+1}}{\alpha s_j^2}\right)\left(SV^T X^T\beta_{true} + SV^T\tilde{z} + U^T\lambda\beta_{\text{prior}}\right)]
\end{aligned}$$

By linearity of the expectation value,

$$E[\beta^{k+1}] = E[\alpha U \operatorname{diag}_{j=1}^{p}\left(\frac{1-\left(1-\alpha s_j^2\right)^{k+1}}{\alpha s_j^2}\right)(SV^T X^T \beta_{true})]$$
$$+E[\alpha U \operatorname{diag}_{j=1}^{p}\left(\frac{1-\left(1-\alpha s_j^2\right)^{k+1}}{\alpha s_j^2}\right)(SV^T \tilde{z})]$$
$$+E[\alpha U \operatorname{diag}_{j=1}^{p}\left(\frac{1-\left(1-\alpha s_j^2\right)^{k+1}}{\alpha s_j^2}\right)(U^T \lambda \beta_{\text{prior}})]$$

$$E[\beta^{k+1}] = E[\alpha U \operatorname{diag}_{j=1}^{p}\left(\frac{1-\left(1-\alpha s_j^2\right)^{k+1}}{\alpha s_j^2}\right)(SV^T X^T \beta_{true})] + E[\alpha U \operatorname{diag}_{j=1}^{p}\left(\frac{1-\left(1-\alpha s_j^2\right)^{k+1}}{\alpha s_j^2}\right)(U^T \lambda \beta_{\text{prior}})]$$

$$E[\beta^{k+1}] = \alpha U \operatorname{diag}_{j=1}^{p}\left(\frac{1-\left(1-\alpha s_j^2\right)^{k+1}}{\alpha s_j^2}\right)\left(S^2 U^T \beta_{\text{true}} + U^T \lambda \beta_{\text{prior}}\right)$$

$$E[\beta^{k+1}] = U \operatorname{diag}_{j=1}^{p}\left(1-\left(1-\alpha s_j^2\right)^{k+1}\right)U^T \beta_{\text{true}} + U \operatorname{diag}_{j=1}^{p}\left(\left(1-\alpha s_j^2\right)^{k+1}\right)U^T \beta_{\text{prior}}$$

And the covariance will not change since the extra term we added do not affect the noise.

4. (Stochastic Gradient Descent) In class we saw how to use gradient descent to solve an optimization problem and applied it to ordinary least squares and ridge regression. However, gradient descent is often not used in practice. When the training data set is very large, computing the gradient of the objective function can take a long time as we need to go through the whole dataset for a single gradient step. When the objective function takes the form of an average of many values, such as in the case of linear regression

$$J(\beta) = \frac{1}{m}\sum_{i=1}^{m}\left(\beta^T x_i - y_i\right)^2$$

stochastic gradient descent (SGD) can be very effective.

(a) Here In SGD, rather than taking $-\nabla J(\beta)$ as our step direction (as in gradient descent), we take the gradient of objective function assuming there there is only data point i chosen uniformly at random from $\{1, \ldots, m\}$. Show that the SGD gradient is an unbiased estimator of the real gradient $-\nabla J(\beta)$.

Since we know the linear regression from the question is:

$$J(\beta) = \frac{1}{m}\sum_{i=1}^{m}\left(\beta^T x_i - y_i\right)^2$$

$$\begin{aligned}\nabla J(\beta) &= XX^T\beta - Xy \\ &= \frac{1}{m}\sum_{i=1}^{m} 2x_i x^T\beta - 2x_i^T y_i\end{aligned}$$

Then

$$\nabla \tilde{J}_n(\beta) = \frac{1}{n}\sum_{i=1}^{n} 2x_i x^T\beta - 2x_i^T y_i, \text{ where } n \le m$$

And the expectation will be:

$$\begin{aligned}E[\nabla \tilde{J}_n(\beta)] &= E[\frac{1}{n}\sum_{i=1}^{n} 2x_i x^T\beta - 2x_i^T y_i] \\ &= \frac{1}{n}\sum_{i=1}^{n} E[2x_i x^T\beta - 2x_i^T y_i] \\ &= E[2x_i x^T\beta - 2x_i^T y_i] \\ &= \frac{1}{m}\sum_{i=1}^{m} 2x_i x^T\beta - 2x_i^T y_i = \nabla J(\beta)\end{aligned}$$

Therefore, since $E[\nabla \tilde{J}_n(\beta)] = \nabla J(\beta)$, then the SGD is an unbaised estimator.

(b) Derive the SGD update rule for the linear regression

Based on the formula on *regularization lecture notes*,

$$\beta^{(k+1)} := \beta^{(k)} + \alpha_k X\left(y - X^T\beta^{(k)}\right) \tag{19}$$

Then the SGD update rule will be:

$$\beta^{(k+1)} := \beta^{(k)} + \alpha_k\left(y_n - x_n^T\beta^{(k)}\right)x_n \text{ for } n = 1, 2, \ldots, m$$

Please see (c) (d) (e) in another file.