Regularizations

Ngoc Hoang Luong

University of Information Technology (UIT), VNU-HCM

April 13, 2023

Linear Regression

- In **linear regression**, the overall error function $E()$ is the mean squared error (MSE).

Linear Regression

- In **linear regression**, the overall error function $E()$ is the mean squared error (MSE).
- From the perspective of the parameters (i.e., the regression coefficients), we denote the error function as $E(\mathbf{b})$.

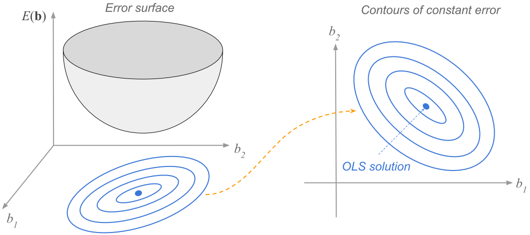
$$E(\mathbf{b}) = \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^\top(\mathbf{X}\mathbf{b} - \mathbf{y})$$

Linear Regression

- In **linear regression**, the overall error function $E()$ is the mean squared error (MSE).
- From the perspective of the parameters (i.e., the regression coefficients), we denote the error function as $E(\mathbf{b})$.

$$E(\mathbf{b}) = \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^{\top}(\mathbf{X}\mathbf{b} - \mathbf{y})$$

- Let's consider two inputs X_1 and X_2, and their corresponding parameters b_1 and b_2. The error function $E(\mathbf{b})$ generates a convex error surface with the shape of a bowl (or a paraboloid).



Linear Regression

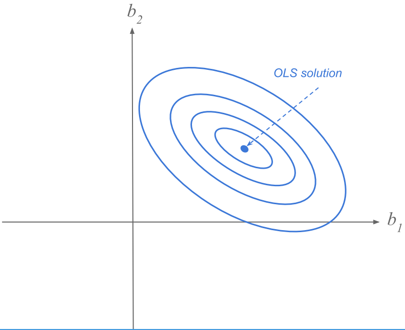
$$E(\mathbf{b}) = \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^\top(\mathbf{X}\mathbf{b} - \mathbf{y})$$

- In ordinary least squares (OLS), we minimize $E(\mathbf{b})$ **unconditionally**: without any restriction.

Linear Regression

$$E(\mathbf{b}) = \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^\top(\mathbf{X}\mathbf{b} - \mathbf{y})$$

- In ordinary least squares (OLS), we minimize $E(\mathbf{b})$ **unconditionally**: without any restriction.
- The solution is indicated with a blue dot at the center of the elliptical contours of constant error.



Linear Regression - Constraining Regression Coefficients

$$E(\mathbf{b}) = \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^\top(\mathbf{X}\mathbf{b} - \mathbf{y})$$

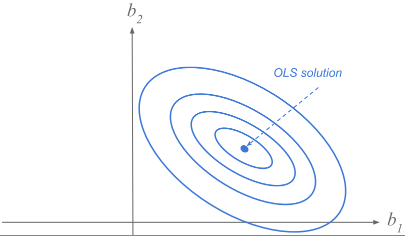
- We would like now to impose a restriction on the squared magnitude of the regression coefficients.

Linear Regression - Constraining Regression Coefficients

$$E(\mathbf{b}) = \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^\top(\mathbf{X}\mathbf{b} - \mathbf{y})$$

- We would like now to impose a restriction on the squared magnitude of the regression coefficients.
- We still minimize $E(\mathbf{b})$, but now we require the following condition on $b_1, b_2, \ldots, b_p$:

$$\sum_{j=1}^{p} b_j^2 \leq c \tag{1}$$

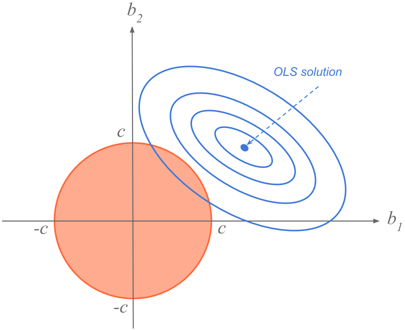


Linear Regression - Constraining Regression Coefficients

$$E(\mathbf{b}) = \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^\top(\mathbf{X}\mathbf{b} - \mathbf{y})$$

- We have a constrained minimization of $E(\mathbf{b})$ for some "budget" c:

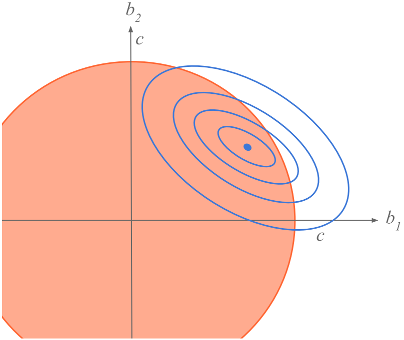
$$\min_{\mathbf{b}} \left\{ \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^\top(\mathbf{X}\mathbf{b} - \mathbf{y}) \right\} \text{ st } \|\mathbf{b}\|_2^2 = \mathbf{b}^\top\mathbf{b} \leq c$$



Linear Regression - Constraining Regression Coefficients

$$\min_{\mathbf{b}} \left\{ \frac{1}{n} (\mathbf{Xb} - \mathbf{y})^\top (\mathbf{Xb} - \mathbf{y}) \right\} \text{ st } \|\mathbf{b}\|_2^2 = \mathbf{b}^\top \mathbf{b} \leq c$$

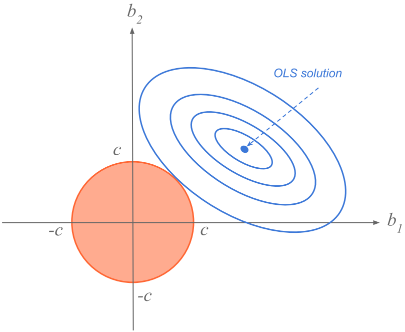
- If we choose too big values of c, we could have a big enough constraint that includes the OLS solution.



Linear Regression - Constraining Regression Coefficients

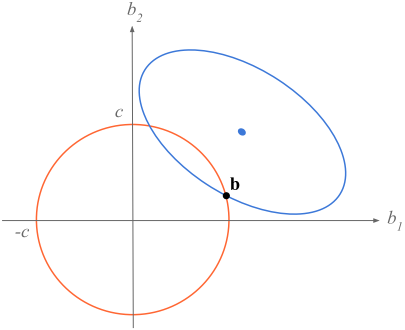
$$\min_{\mathbf{b}} \left\{ \frac{1}{n} (\mathbf{X}\mathbf{b} - \mathbf{y})^\top (\mathbf{X}\mathbf{b} - \mathbf{y}) \right\} \text{ st } \|\mathbf{b}\|_2^2 = \mathbf{b}^\top \mathbf{b} \leq c$$

- We could make the budget stricter by reducing the value of c



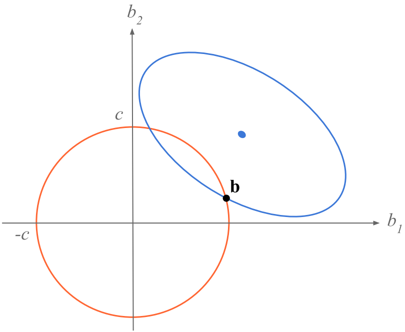
Linear Regression - A New Minimization Solution

- Let's consider one elliptical contour of constant error, a given budget c, and a point $\mathbf{b}$ satisfying the budget constraint



Linear Regression - A New Minimization Solution

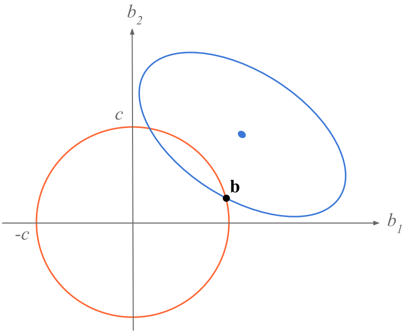
- Let's consider one elliptical contour of constant error, a given budget c, and a point $\mathbf{b}$ satisfying the budget constraint



- $\mathbf{b}^\top \mathbf{b} = c$. However, this point does not fully minimize $E(\mathbf{b})$.

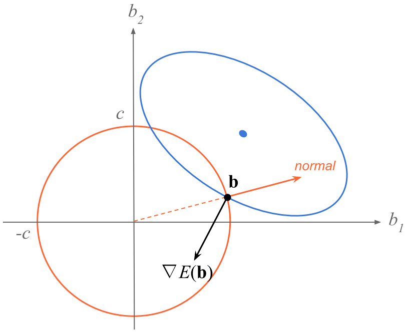
Linear Regression - A New Minimization Solution

- Let's consider one elliptical contour of constant error, a given budget c, and a point $\mathbf{b}$ satisfying the budget constraint



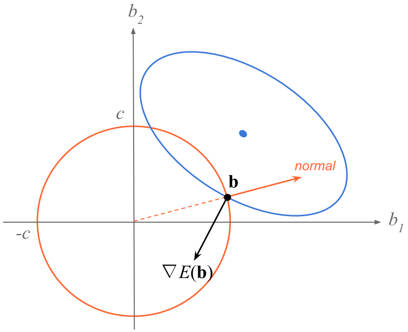
- $\mathbf{b}^\top \mathbf{b} = c$. However, this point does not fully minimize $E(\mathbf{b})$.
- We could still find other $\mathbf{b}$ along the circle that would give us smaller $E(\mathbf{b})$.

Linear Regression - A New Minimization Solution



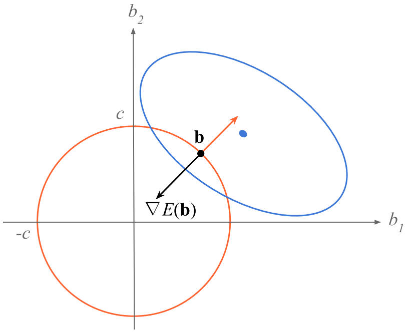
- The gradient $\nabla E(\mathbf{b})$ points in the direction orthogonal to the contour ellipse, i.e., the direction of largest change of $E(\mathbf{b})$.

Linear Regression - A New Minimization Solution



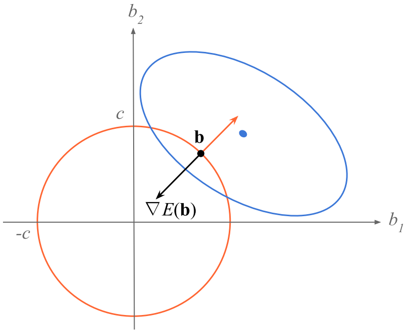
- The gradient $\nabla E(\mathbf{b})$ points in the direction orthogonal to the contour ellipse, i.e., the direction of largest change of $E(\mathbf{b})$.
- The direction of $\mathbf{b}$ is orthogonal to the circumference of the constraint (normal vector). The angle between the gradient and the normal vector is less than 180 degrees. We can find better $\mathbf{b}$ points that make the error smaller. Where is that optimal $\mathbf{b}^*$?

Linear Regression - A New Minimization Solution



- The optimal vector $\mathbf{b}^*$ corresponds to the one that is exactly the opposite of $\nabla E(\mathbf{b})$. The gradient and the normal vectors are anti-parallel: $\nabla E(\mathbf{b}^*) \propto -\mathbf{b}^*$.

Linear Regression - A New Minimization Solution



- The optimal vector $\mathbf{b}^*$ corresponds to the one that is exactly the opposite of $\nabla E(\mathbf{b})$. The gradient and the normal vectors are anti-parallel: $\nabla E(\mathbf{b}^*) \propto -\mathbf{b}^*$.
- We choose a proportionality constant of $-2(\lambda/n)$

$$\nabla E(\mathbf{b}^*) = -2\frac{\lambda}{n}\mathbf{b}^*$$

Linear Regression - A New Minimization Solution

$$\nabla E(\mathbf{b}^*) = -2\frac{\lambda}{n}\mathbf{b}^*$$

$$\nabla E(\mathbf{b}^*) + 2\frac{\lambda}{n}\mathbf{b}^* = \mathbf{0}$$

- The above expression is the gradient of the following function:

$$f(\mathbf{b}) = E(\mathbf{b}) + \frac{\lambda}{n}\mathbf{b}^\top\mathbf{b}$$

$$\nabla f(\mathbf{b}^*) = \nabla E(\mathbf{b}^*) + 2\frac{\lambda}{n}\mathbf{b}^*$$

Linear Regression - A New Minimization Solution

$$\nabla E(\mathbf{b}^*) = -2\frac{\lambda}{n}\mathbf{b}^*$$

$$\nabla E(\mathbf{b}^*) + 2\frac{\lambda}{n}\mathbf{b}^* = \mathbf{0}$$

- The above expression is the gradient of the following function:

$$f(\mathbf{b}) = E(\mathbf{b}) + \frac{\lambda}{n}\mathbf{b}^\top\mathbf{b}$$

$$\nabla f(\mathbf{b}^*) = \nabla E(\mathbf{b}^*) + 2\frac{\lambda}{n}\mathbf{b}^*$$

- Now, our minimization problem becomes:

$$\min_{\mathbf{b}}\left\{E(\mathbf{b}) + \frac{\lambda}{n}\mathbf{b}^\top\mathbf{b}\right\}$$

Linear Regression - A New Minimization Solution

$$
\begin{aligned}
f(\mathbf{b}) = E(\mathbf{b}) + \frac{\lambda}{n}\mathbf{b}^\top\mathbf{b} &= \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^\top(\mathbf{X}\mathbf{b} - \mathbf{y}) + \frac{\lambda}{n}\mathbf{b}^\top\mathbf{b} \\
&= \frac{1}{n}\mathbf{b}^\top\mathbf{X}^\top\mathbf{X}\mathbf{b} - \frac{2}{n}\mathbf{b}^\top\mathbf{X}^\top\mathbf{y} + \frac{1}{n}\mathbf{y}^\top\mathbf{y} + \frac{\lambda}{n}\mathbf{b}^\top\mathbf{b}
\end{aligned}
$$

- Compute the gradient:

$$
\nabla f(\mathbf{b}) = \frac{2}{n}\mathbf{X}^\top\mathbf{X}\mathbf{b} - \frac{2}{n}\mathbf{X}^\top\mathbf{y} + \frac{2\lambda}{n}\mathbf{b}
$$

Linear Regression - A New Minimization Solution

$$
\begin{aligned}
f(\mathbf{b}) = E(\mathbf{b}) + \frac{\lambda}{n}\mathbf{b}^\top\mathbf{b} &= \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^\top(\mathbf{X}\mathbf{b} - \mathbf{y}) + \frac{\lambda}{n}\mathbf{b}^\top\mathbf{b} \\
&= \frac{1}{n}\mathbf{b}^\top\mathbf{X}^\top\mathbf{X}\mathbf{b} - \frac{2}{n}\mathbf{b}^\top\mathbf{X}^\top\mathbf{y} + \frac{1}{n}\mathbf{y}^\top\mathbf{y} + \frac{\lambda}{n}\mathbf{b}^\top\mathbf{b}
\end{aligned}
$$

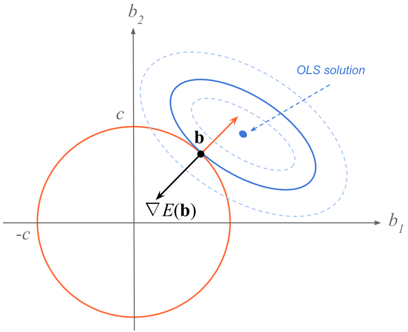
- Compute the gradient:

$$\nabla f(\mathbf{b}) = \frac{2}{n}\mathbf{X}^\top\mathbf{X}\mathbf{b} - \frac{2}{n}\mathbf{X}^\top\mathbf{y} + \frac{2\lambda}{n}\mathbf{b}$$

- Setting $\nabla f(\mathbf{b})$ to $\mathbf{0}$, and solve for the **ridge regression** of $\mathbf{b}$:

$$
\begin{aligned}
\mathbf{X}^\top\mathbf{X}\mathbf{b} - \mathbf{X}^\top\mathbf{y} + \lambda\mathbf{b} &= \mathbf{0} \\
\mathbf{b}(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}) &= \mathbf{X}^\top\mathbf{y} \\
\mathbf{b}_{RR} &= (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y}
\end{aligned}
$$

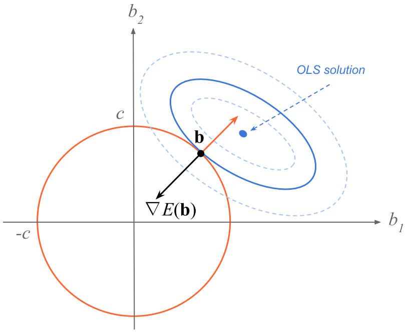
Ridge Regression



$$\texttt{ridge coefficients } \mathbf{b}_{RR} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{y}$$

- The optimal vector $\mathbf{b}$ that minimizes $E(\mathbf{b})$ and satisfies the constraint will be on a contour of constant error.

Ridge Regression



$$\texttt{ridge coefficients } \mathbf{b}_{RR} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{y}$$

- The optimal vector $\mathbf{b}$ that minimizes $E(\mathbf{b})$ and satisfies the constraint will be on a contour of constant error.
- The above illustration shows that the direction of the optimal vector $\mathbf{b}$ does not point in the direction of the OLS solution.

Ridge Regression

- If we set the hyperparameter $\lambda = 0$, the ridge solution is the same as the OLS solution.

$$\mathbf{b}_{RR} = (\mathbf{X}^\top \mathbf{X} + 0\mathbf{I})^{-1}\mathbf{X}^\top \mathbf{y} = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y} = \mathbf{b}_{OLS}$$

Ridge Regression

- If we set the hyperparameter $\lambda = 0$, the ridge solution is the same as the OLS solution.

$$\mathbf{b}_{RR} = (\mathbf{X}^\top \mathbf{X} + 0\mathbf{I})^{-1}\mathbf{X}^\top \mathbf{y} = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y} = \mathbf{b}_{OLS}$$

- If λ becomes larger and larger, the terms on the diagonal of $(\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})$ will be dominated by such λ values.

Ridge Regression

- If we set the hyperparameter $\lambda = 0$, the ridge solution is the same as the OLS solution.

$$\mathbf{b}_{RR} = (\mathbf{X}^\top\mathbf{X} + 0\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y} = \mathbf{b}_{OLS}$$

- If λ becomes larger and larger, the terms on the diagonal of $(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})$ will be dominated by such λ values.
- The matrix inverse, $(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}$, will be dominated by the inverse of the terms on its diagonal:

$$\begin{aligned}
\lambda \gg 0 &\Rightarrow (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1} \rightarrow \frac{1}{\lambda}\mathbf{I} \\
\lambda \rightarrow \infty &\Rightarrow (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1} \rightarrow \mathbf{0}_{(p,p)} \\
\lambda \rightarrow \infty &\Rightarrow \mathbf{b}_{RR} = \mathbf{0}
\end{aligned}$$

Ridge Regression - Relation between λ and c

- Our two minimization problems:

$$\min_{\mathbf{b}} \left\{ \frac{1}{n} (\mathbf{X}\mathbf{b} - \mathbf{y})^\top (\mathbf{X}\mathbf{b} - \mathbf{y}) \right\} \text{ st } \|\mathbf{b}\|_2^2 = \mathbf{b}^\top \mathbf{b} \leq c$$

$$\min_{\mathbf{b}} \left\{ E(\mathbf{b}) + \frac{\lambda}{n} \mathbf{b}^\top \mathbf{b} \right\}$$

Ridge Regression - Relation between λ and c

- Our two minimization problems:

$$\min_{\mathbf{b}} \left\{ \frac{1}{n} (\mathbf{X}\mathbf{b} - \mathbf{y})^\top (\mathbf{X}\mathbf{b} - \mathbf{y}) \right\} \text{ st } \|\mathbf{b}\|_2^2 = \mathbf{b}^\top \mathbf{b} \leq c$$

$$\min_{\mathbf{b}} \left\{ E(\mathbf{b}) + \frac{\lambda}{n} \mathbf{b}^\top \mathbf{b} \right\}$$

- Imposing a large budget constraint c causes λ to become smaller. The smaller the λ, the closer $\mathbf{b}_{RR}$ to the OLS solution.

Ridge Regression - Relation between λ and c

- Our two minimization problems:

$$\min_{\mathbf{b}} \left\{ \frac{1}{n} (\mathbf{Xb} - \mathbf{y})^\top (\mathbf{Xb} - \mathbf{y}) \right\} \text{ st } \|\mathbf{b}\|_2^2 = \mathbf{b}^\top \mathbf{b} \le c$$

$$\min_{\mathbf{b}} \left\{ E(\mathbf{b}) + \frac{\lambda}{n} \mathbf{b}^\top \mathbf{b} \right\}$$

- Imposing a large budget constraint c causes λ to become smaller. The smaller the λ, the closer $\mathbf{b}_{RR}$ to the OLS solution.
- Imposing a small budget constraint c causes λ to become larger. The larger the λ, the smaller the ridge coefficients.

$$\uparrow c \Rightarrow \downarrow \lambda$$

$$\downarrow c \Rightarrow \uparrow \lambda$$

Ridge Regression - How to find λ?

- In ridge regression, λ is a hyperparameter that we need to tune.

Ridge Regression - How to find λ?

- In ridge regression, λ is a hyperparameter that we need to tune.
- We randomly split the training data:

$$\mathcal{D}_{train} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$$

into K folds:

$$\mathcal{D}_{train} = \mathcal{D}_{fold-1} \cup \mathcal{D}_{fold-2} \cup \ldots \cup \mathcal{D}_{fold-K}$$

Ridge Regression - How to find λ?

- In ridge regression, λ is a hyperparameter that we need to tune.
- We randomly split the training data:

$$\mathcal{D}_{train} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$$

into K folds:

$$\mathcal{D}_{train} = \mathcal{D}_{fold-1} \cup \mathcal{D}_{fold-2} \cup \ldots \cup \mathcal{D}_{fold-K}$$

- Each fold set $\mathcal{D}_{fold-k}$ plays the role of an evaluation $\mathcal{D}_{eval-k}$. The corresponding K training sets:

Ridge Regression - How to find λ?

- In ridge regression, λ is a hyperparameter that we need to tune.
- We randomly split the training data:

$$\mathcal{D}_{train} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$$

into K folds:

$$\mathcal{D}_{train} = \mathcal{D}_{fold-1} \cup \mathcal{D}_{fold-2} \cup \ldots \cup \mathcal{D}_{fold-K}$$

- Each fold set $\mathcal{D}_{fold-k}$ plays the role of an evaluation $\mathcal{D}_{eval-k}$. The corresponding K training sets:
  - $\mathcal{D}_{train-1} = \mathcal{D}_{train} \setminus \mathcal{D}_{fold-1}$

Ridge Regression - How to find λ?

- In ridge regression, λ is a hyperparameter that we need to tune.
- We randomly split the training data:

$$\mathcal{D}_{train} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$$

  into K folds:

$$\mathcal{D}_{train} = \mathcal{D}_{fold-1} \cup \mathcal{D}_{fold-2} \cup \dots \cup \mathcal{D}_{fold-K}$$

- Each fold set $\mathcal{D}_{fold-k}$ plays the role of an evaluation $\mathcal{D}_{eval-k}$. The corresponding K training sets:
  - $\mathcal{D}_{train-1} = \mathcal{D}_{train} \Big\backslash \mathcal{D}_{fold-1}$
  - $\mathcal{D}_{train-2} = \mathcal{D}_{train} \Big\backslash \mathcal{D}_{fold-2}$

Ridge Regression - How to find λ?

- In ridge regression, λ is a hyperparameter that we need to tune.
- We randomly split the training data:

$$\mathcal{D}_{train} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$$

into K folds:

$$\mathcal{D}_{train} = \mathcal{D}_{fold-1} \cup \mathcal{D}_{fold-2} \cup \ldots \cup \mathcal{D}_{fold-K}$$

- Each fold set $\mathcal{D}_{fold-k}$ plays the role of an evaluation $\mathcal{D}_{eval-k}$. The corresponding K training sets:
  - $\mathcal{D}_{train-1} = \mathcal{D}_{train} \Big\backslash \mathcal{D}_{fold-1}$
  - $\mathcal{D}_{train-2} = \mathcal{D}_{train} \Big\backslash \mathcal{D}_{fold-2}$
  - $\ldots$

Ridge Regression - How to find λ?

- In ridge regression, λ is a hyperparameter that we need to tune.
- We randomly split the training data:

$$\mathcal{D}_{train} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$$

  into K folds:

$$\mathcal{D}_{train} = \mathcal{D}_{fold-1} \cup \mathcal{D}_{fold-2} \cup \ldots \cup \mathcal{D}_{fold-K}$$

- Each fold set $\mathcal{D}_{fold-k}$ plays the role of an evaluation $\mathcal{D}_{eval-k}$. The corresponding K training sets:
  - $\mathcal{D}_{train-1} = \mathcal{D}_{train} \Big\backslash \mathcal{D}_{fold-1}$
  - $\mathcal{D}_{train-2} = \mathcal{D}_{train} \Big\backslash \mathcal{D}_{fold-2}$
  - $\ldots$
  - $\mathcal{D}_{train-K} = \mathcal{D}_{train} \Big\backslash \mathcal{D}_{fold-K}$

Ridge Regression - How to find λ?

1. For $\lambda_b = 0.001, 0.002, \ldots, \lambda_B$:

Ridge Regression - How to find λ?

1. For $\lambda_b = 0.001, 0.002, \ldots, \lambda_B$:
   - For $k = 1, \ldots, K$:

Ridge Regression - How to find λ?

1. For $\lambda_b = 0.001, 0.002, \ldots, \lambda_B$:
   - For $k = 1, \ldots, K$:
     - Fit RR model $h_{b,k}$ with λ_b on $\mathcal{D}_{train-k}$

Ridge Regression - How to find λ?

1. For $\lambda_b = 0.001, 0.002, \ldots, \lambda_B$:
   - For $k = 1, \ldots, K$:
     - Fit RR model $h_{b,k}$ with λ_b on $\mathcal{D}_{train-k}$
     - Compute and store $E_{eval-k}(h_{b,k})$ using $\mathcal{D}_{eval-k}$

Ridge Regression - How to find λ?

1. For $\lambda_b = 0.001, 0.002, \ldots, \lambda_B$:
   - For $k = 1, \ldots, K$:
     - Fit RR model $h_{b,k}$ with λ_b on $\mathcal{D}_{train-k}$
     - Compute and store $E_{eval-k}(h_{b,k})$ using $\mathcal{D}_{eval-k}$
   - Compute and store $E_{cv_b} = \frac{1}{K} \sum_k E_{eval-k}(h_{b,k})$

Ridge Regression - How to find λ?

1. For $\lambda_b = 0.001, 0.002, \ldots, \lambda_B$:
   - For $k = 1, \ldots, K$:
     - Fit RR model $h_{b,k}$ with λ_b on $\mathcal{D}_{train-k}$
     - Compute and store $E_{eval-k}(h_{b,k})$ using $\mathcal{D}_{eval-k}$
   - Compute and store $E_{cv_b} = \frac{1}{K}\sum_k E_{eval-k}(h_{b,k})$
2. Compute all cross validation errors $E_{cv_1}, E_{cv_2}, \ldots, E_{cv_B}$ and choose the smallest $E_{cv_{b^*}}$.

Ridge Regression - How to find λ?

1. For $\lambda_b = 0.001, 0.002, \ldots, \lambda_B$:
   - For $k = 1, \ldots, K$:
     - Fit RR model $h_{b,k}$ with λ_b on $\mathcal{D}_{train-k}$
     - Compute and store $E_{eval-k}(h_{b,k})$ using $\mathcal{D}_{eval-k}$
   - Compute and store $E_{cv_b} = \frac{1}{K}\sum_k E_{eval-k}(h_{b,k})$
2. Compute all cross validation errors $E_{cv_1}, E_{cv_2}, \ldots, E_{cv_B}$ and choose the smallest $E_{cv_{b^*}}$.
3. Use λ^* to fit the final Ridge Regression model:

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

Least Absolute Shrinkage and Selection Operator - LASSO

- Instead of using L_2 norm as in Ridge Regression:

$$\min_{\mathbf{b}\in\mathbb{R}^p}\left\{\frac{1}{n}\|\mathbf{X}\mathbf{b}-\mathbf{y}\|_2^2\right\} \quad \texttt{subject to} \quad \|\mathbf{b}\|_2^2 \le c$$

Least Absolute Shrinkage and Selection Operator - LASSO

- Instead of using L_2 norm as in Ridge Regression:

$$\min_{\mathbf{b}\in\mathbb{R}^p} \left\{ \frac{1}{n} \|\mathbf{X}\mathbf{b} - \mathbf{y}\|_2^2 \right\} \quad \texttt{subject to} \quad \|\mathbf{b}\|_2^2 \le c$$

- We use L_1 norm:

$$\min_{\mathbf{b}\in\mathbb{R}^p} \left\{ \frac{1}{n} \|\mathbf{X}\mathbf{b} - \mathbf{y}\|_2^2 \right\} \quad \texttt{subject to} \quad \|\mathbf{b}\|_1 \le c$$

Least Absolute Shrinkage and Selection Operator - LASSO

- Instead of using L_2 norm as in Ridge Regression:

$$\min_{\mathbf{b}\in\mathbb{R}^p}\left\{\frac{1}{n}\|\mathbf{X}\mathbf{b}-\mathbf{y}\|_2^2\right\} \quad \texttt{subject to} \quad \|\mathbf{b}\|_2^2 \le c$$

- We use L_1 norm:

$$\min_{\mathbf{b}\in\mathbb{R}^p}\left\{\frac{1}{n}\|\mathbf{X}\mathbf{b}-\mathbf{y}\|_2^2\right\} \quad \texttt{subject to} \quad \|\mathbf{b}\|_1 \le c$$

- The L_1 norm constraint is:

$$\|\mathbf{b}\|_1 \le c \iff \sum_{j=1}^{p} |b_j| \le c$$

Least Absolute Shrinkage and Selection Operator - LASSO

- Instead of using L_2 norm as in Ridge Regression:

$$\min_{\mathbf{b}\in\mathbb{R}^p}\left\{\frac{1}{n}\|\mathbf{X}\mathbf{b}-\mathbf{y}\|_2^2\right\} \quad \texttt{subject to} \quad \|\mathbf{b}\|_2^2 \le c$$

- We use L_1 norm:

$$\min_{\mathbf{b}\in\mathbb{R}^p}\left\{\frac{1}{n}\|\mathbf{X}\mathbf{b}-\mathbf{y}\|_2^2\right\} \quad \texttt{subject to} \quad \|\mathbf{b}\|_1 \le c$$

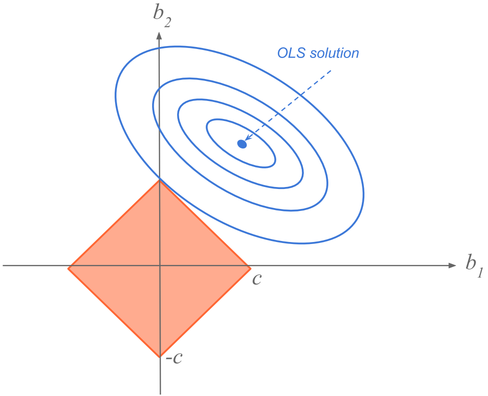
- The L_1 norm constraint is:

$$\|\mathbf{b}\|_1 \le c \iff \sum_{j=1}^{p}|b_j| \le c$$

- Our minimization problem then becomes:

$$\min_{\mathbf{b}\in\mathbb{R}^p}\left\{\frac{1}{n}(\mathbf{X}\mathbf{b}-\mathbf{y})^\top(\mathbf{X}\mathbf{b}-\mathbf{y}) + \lambda\sum_{j=1}^{p}|b_j|\right\}$$

LASSO - Variable Selection



- The ideal point has b_1 coordinate equal to 0.
- LASSO completely zero-ed out b_1 in the model, reducing the number of coefficients from 2 down to 1.