ML&MEA (2024)
Lecture 4 - Ridge regression

Quanying Liu

BME, SUSTech

2024.3.5

Content

Recap Lecture 4

- **Probability theory**: random variable, PMF, PDF

  Gaussian distribution:

$$p(x|\mu,\sigma^2) = \mathcal{N}(\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$$

  Multivariate Gaussian distribution:

$$\begin{aligned} P(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) &= \mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma}) \\ &= \frac{1}{\sqrt{(2\pi)^p \det(\boldsymbol{\Sigma})}} \exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right] \end{aligned}$$

Recap Lecture 4

- **Probability theory**: random variable, PMF, PDF

  Gaussian distribution:

$$p(x|\mu, \sigma^2) = \mathcal{N}(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$$

  Multivariate Gaussian distribution:

$$\begin{aligned} P(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) &= \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \\ &= \frac{1}{\sqrt{(2\pi)^p \det(\boldsymbol{\Sigma})}} \exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right] \end{aligned}$$

- **Bayes' theorem**:

$$P(\theta|\mathbf{X}) = \frac{P(\mathbf{X}|\theta)P(\theta)}{P(\mathbf{X})} = \frac{P(\mathbf{X}|\theta)P(\theta)}{\int_\theta P(\mathbf{X}|\theta)P(\theta)d\theta} \propto P(\mathbf{X}|\theta)P(\theta)$$

Recap Lecture 4

- **Frequentist vs Bayesian**:

Recap Lecture 4

- **Frequentist vs Bayesian**:
  1. Frequentist: The parameters θ are constant.
     Maximum (log-)likelihood estimation (MLE) to estimate θ:

$$\theta_{MLE} = \arg\max_{\theta} \log P(\mathbf{X}|\theta)$$

Recap Lecture 4

- **Frequentist vs Bayesian**:
  1. Frequentist: The parameters θ are constant.
     Maximum (log-)likelihood estimation (MLE) to estimate θ:
     $$\theta_{MLE} = \arg\max_{\theta} \log P(\mathbf{X}|\theta)$$
  2. Bayesian: The parameters θ are not constant. We have some prior information of θ.
     Maximum a posterior (MAP) to estimate θ:
     $$\theta_{MAP} = \arg\max_{\theta} \log P(\theta|\mathbf{X}) = \arg\max_{\theta} \log P(\mathbf{X}|\theta) + \log P(\theta)$$

Recap Lecture 4

- **Frequentist vs Bayesian**:
  1. Frequentist: The parameters θ are constant.
     Maximum (log-)likelihood estimation (MLE) to estimate θ:
     $$\theta_{MLE} = \arg\max_{\theta} \log P(\mathbf{X}|\theta)$$
  2. Bayesian: The parameters θ are not constant. We have some prior information of θ.
     Maximum a posterior (MAP) to estimate θ:
     $$\theta_{MAP} = \arg\max_{\theta} \log P(\theta|\mathbf{X}) = \arg\max_{\theta} \log P(\mathbf{X}|\theta) + \log P(\theta)$$

- **Linear regression in frequentist view**:
  Assuming Gaussian prior of errors:
  MLE = minimizing *negative log-likelihood* = minimizing MSE

Recap Linear regression in frequentist view

- Probabilistic model for linear regression:

$$\text{Model: } y = \beta_0 + \beta_1 x + \epsilon = \beta^T \mathbf{x} + \epsilon$$
$$\text{Prior of error: } \epsilon \sim \mathcal{N}(0, \sigma^2)$$

Recap Linear regression in frequentist view

- Probabilistic model for linear regression:

$$\text{Model: } y = \beta_0 + \beta_1 x + \epsilon = \beta^T \mathbf{x} + \epsilon$$

$$\text{Prior of error: } \epsilon \sim \mathcal{N}(0, \sigma^2)$$

- For the entire training set $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$,

$$P(\mathbf{y}|\mathbf{X}, \beta) = \prod_{i=1} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(y^{(i)} - \beta^T \mathbf{x}^{(i)})^2}{2\sigma^2}\right]$$

Recap Linear regression in frequentist view

- Probabilistic model for linear regression:

$$\text{Model: } y = \beta_0 + \beta_1 x + \epsilon = \beta^T \mathbf{x} + \epsilon$$
$$\text{Prior of error: } \epsilon \sim \mathcal{N}(0, \sigma^2)$$

- For the entire training set $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^N$,

$$P(\mathbf{y}|\mathbf{X}, \beta) = \prod_{i=1} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(y^{(i)} - \beta^T \mathbf{x}^{(i)})^2}{2\sigma^2}\right]$$

- MLE to estimate the parameter β

$$\begin{aligned}\beta_{MLE} &= \arg\max_\beta \log P(\mathbf{y}|\mathbf{X}, \beta) \\ &= \arg\min_\beta ||y - \mathbf{X}\beta||^2 \rightarrow \text{MSE loss}\end{aligned}$$

Recap Basic Python

- **Data types in python**
  Numeric types: int, long, float, complex
  Strings: s='abcde'
  List: thislist=["apple","banana","orange"]
- **Operators**
  Arithmetic operators: +, -, *, /, %, **, //
  Assignment operators: +=, -=, *=, /=, **=
  Comparison operators: ==, !=, >, <, >=, <=
  Logical operators: and, or, not
- **Conditions**: if ... else
- **Loops**: while; for
- **Functions**: define a function; call a function
- **Libraries in Python**: Numpy; SciPy; Matplotlib

Overfitting

- Recall the analytical solution of multiple linear regression:

$$\hat{\beta} = \left(\mathbf{X}^T \mathbf{X}\right)^{-1} \mathbf{X}^T \mathbf{y}$$

Overfitting

- Recall the analytical solution of multiple linear regression:

$$\hat{\beta} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$$

- When the number of samples is less than the number of features (which means $n < p$), $\mathbf{X}^T\mathbf{X} \in \mathbb{R}^{p\times p}$ is not a full rank matrix. $\longrightarrow$ We can NOT calculate $(\mathbf{X}^T\mathbf{X})^{-1}$.

Overfitting

- Recall the analytical solution of multiple linear regression:

$$\hat{\beta} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$$

- When the number of samples is less than the number of features (which means $n < p$), $\mathbf{X}^T\mathbf{X} \in \mathbb{R}^{p\times p}$ is not a full rank matrix. $\longrightarrow$ We can NOT calculate $(\mathbf{X}^T\mathbf{X})^{-1}$.
- This problem is also called **overfitting**.

Overfitting

- Recall the analytical solution of multiple linear regression:

$$\hat{\beta} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$$

- When the number of samples is less than the number of features (which means $n < p$), $\mathbf{X}^T\mathbf{X} \in \mathbb{R}^{p\times p}$ is not a full rank matrix. $\longrightarrow$ We can NOT calculate $(\mathbf{X}^T\mathbf{X})^{-1}$.
- This problem is also called **overfitting**.
- Some intuitions:

Overfitting

- Recall the analytical solution of multiple linear regression:

$$\hat{\beta} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$$

- When the number of samples is less than the number of features (which means $n < p$), $\mathbf{X}^T\mathbf{X} \in \mathbb{R}^{p\times p}$ is not a full rank matrix. $\longrightarrow$ We can NOT calculate $(\mathbf{X}^T\mathbf{X})^{-1}$.
- This problem is also called **overfitting**.
- Some intuitions:
  1. Overfitting means that the data is not enough. $\longrightarrow$ Collecting more data to train model.

Overfitting

- Recall the analytical solution of multiple linear regression:

$$\hat{\beta} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$$

- When the number of samples is less than the number of features (which means $n < p$), $\mathbf{X}^T\mathbf{X} \in \mathbb{R}^{p\times p}$ is not a full rank matrix. $\longrightarrow$ We can NOT calculate $(\mathbf{X}^T\mathbf{X})^{-1}$.
- This problem is also called **overfitting**.
- Some intuitions:
  1. Overfitting means that the data is not enough. $\longrightarrow$ Collecting more data to train model.
  2. Overfitting means that the features are too many. $\longrightarrow$ Reducing the number of features. Selecting the most effective features, which is called **Feature Engineering**.

Overfitting

- Recall the analytical solution of multiple linear regression:

$$\hat{\beta} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$$

- When the number of samples is less than the number of features (which means $n < p$), $\mathbf{X}^T\mathbf{X} \in \mathbb{R}^{p\times p}$ is not a full rank matrix. $\longrightarrow$ We can NOT calculate $(\mathbf{X}^T\mathbf{X})^{-1}$.
- This problem is also called **overfitting**.
- Some intuitions:
  1. Overfitting means that the data is not enough. $\longrightarrow$ Collecting more data to train model.
  2. Overfitting means that the features are too many. $\longrightarrow$ Reducing the number of features. Selecting the most effective features, which is called **Feature Engineering**.
  3. Overfitting means the model is too complex. $\longrightarrow$ Any solution?

Regularization

- The model is too complex. $\longrightarrow$ Adding some prior knowledge, *e.g.*, Regularization (正则化), to constrain our model

Regularization

- The model is too complex. $\longrightarrow$ Adding some prior knowledge, *e.g.*, Regularization (正则化), to constrain our model
- Loss function with of regularization

$$\arg\min_{\beta} \left[\mathcal{L}(\beta) + \lambda P(\beta)\right],$$

where $P()$ is a penalty function, or regularizer; λ is a hyperparameter to tradeoff the $\mathcal{L}()$ loss and $P()$ regularizer.

Regularization

- The model is too complex. $\longrightarrow$ Adding some prior knowledge, *e.g.*, Regularization (正则化), to constrain our model
- Loss function with of regularization

$$\arg\min_{\beta}\left[\mathcal{L}\left(\beta\right)+\lambda P\left(\beta\right)\right],$$

where $P()$ is a penalty function, or regularizer; λ is a hyperparameter to tradeoff the $\mathcal{L}()$ loss and $P()$ regularizer.

- More regularizatioins:

$$\arg\min_{\beta}\left[\mathcal{L}\left(\beta\right)+\lambda_1 P_1\left(\beta\right)+\lambda_2 P_2\left(\beta\right)+\cdots\right]$$

Regularization

- The model is too complex. $\longrightarrow$ Adding some prior knowledge, *e.g.*, Regularization (正则化), to constrain our model
- Loss function with of regularization

$$\arg\min_{\beta} \left[\mathcal{L}\left(\beta\right) + \lambda P\left(\beta\right)\right],$$

where $P()$ is a penalty function, or regularizer; λ is a hyperparameter to <u>tradeoff</u> the $\mathcal{L}()$ loss and $P()$ regularizer.

- More regularizatioins:

$$\arg\min_{\beta} \left[\mathcal{L}\left(\beta\right) + \lambda_1 P_1\left(\beta\right) + \lambda_2 P_2\left(\beta\right) + \cdots\right]$$

- How to design the regularization term depends on our prior knowledge. (Example: the sparsity or smoothness of the data)

L1 and L2 regularization

- **Some basic (widely-used) regularization terms**

L1 and L2 regularization

- **Some basic (widely-used) regularization terms**
  1. L1 regularization: $P(\beta) = \|\beta\|_1 = \sum_i |\beta_i|$

L1 and L2 regularization

- **Some basic (widely-used) regularization terms**
  1. L1 regularization: $P(\beta) = \|\beta\|_1 = \sum_i |\beta_i|$
  2. L2 regularization: $P(\beta) = \|\beta\|_2^2 = \beta^T \beta$

L1 and L2 regularization

- **Some basic (widely-used) regularization terms**
  1. L1 regularization: $P(\beta) = \|\beta\|_1 = \sum_i |\beta_i|$
  2. L2 regularization: $P(\beta) = \|\beta\|_2^2 = \beta^T \beta$

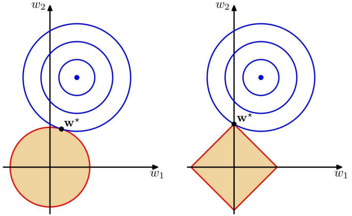

图 1: L2 and L1

L1 and L2 regularization

- **Some basic (widely-used) regularization terms**
  1. L1 regularization: $P(\beta) = \|\beta\|_1 = \sum_i |\beta_i|$
  2. L2 regularization: $P(\beta) = \|\beta\|_2^2 = \beta^T \beta$

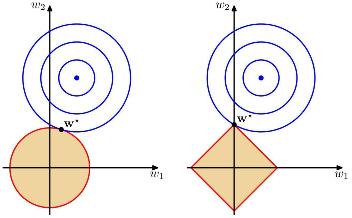

图 1: L2 and L1

- **Two variants of linear regression**
  Lasso regression: $\mathcal{L}(\beta) = (1/n)\|\boldsymbol{y} - \boldsymbol{X}\beta\|^2 + \lambda\|\beta\|_1$
  Ridge regression: $\mathcal{L}(\beta) = (1/n)\|\boldsymbol{y} - \boldsymbol{X}\beta\|^2 + \lambda\|\beta\|_2^2$

Ridge regression: MSE loss with L2 regularization

- Loss function of Ridge regression:

$$\mathcal{L}(\beta) = \underbrace{\frac{1}{n}\|\mathbf{y} - \mathbf{X}\beta\|^2}_{\text{MSE Loss}} + \lambda \underbrace{\|\beta\|_2^2}_{\text{L2 reg.}} \tag{1}$$

Ridge regression: MSE loss with L2 regularization

- Loss function of Ridge regression:

$$\mathcal{L}(\beta) = \underbrace{\frac{1}{n}\|\mathbf{y} - \mathbf{X}\beta\|^2}_{\text{MSE Loss}} + \lambda \underbrace{\|\beta\|_2^2}_{\text{L2 reg.}} \quad (1)$$

- Let us derive the loss for ridge regression:

$$\begin{aligned}
\mathcal{L}(\beta) &= \frac{1}{n}\|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda\beta^T\beta \\
&= \frac{1}{n}(\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta^T\beta \\
&= \frac{1}{n}\left(\beta^T\mathbf{X}^T\mathbf{X}\beta - 2\beta^T\mathbf{X}^T\mathbf{y} + \mathbf{y}^T\mathbf{y}\right) + \frac{1}{n}\left(n\lambda\beta^T\beta\right) \\
&= \frac{1}{n}\left(\beta^T\mathbf{X}^T\mathbf{X}\beta - 2\beta^T\mathbf{X}^T\mathbf{y} + \mathbf{y}^T\mathbf{y} + n\lambda\beta^T\beta\right) \\
&= \frac{1}{n}\left[\beta^T\left(\mathbf{X}^T\mathbf{X} + n\lambda\mathbf{I}\right)\beta - 2\beta^T\mathbf{X}^T\mathbf{y} + \mathbf{y}^T\mathbf{y}\right]
\end{aligned}$$

Analytical solution of ridge regression

- The loss function of ridge regression:

$$\mathcal{L}(\beta) = \frac{1}{n}\left[\beta^T\left(\mathbf{X}^T\mathbf{X} + n\lambda\mathbf{I}\right)\beta - 2\beta^T\mathbf{X}^T\mathbf{y} + \mathbf{y}^T\mathbf{y}\right] \tag{2}$$

Analytical solution of ridge regression

- The loss function of ridge regression:

$$\mathcal{L}(\beta) = \frac{1}{n}\left[\beta^T\left(\mathbf{X}^T\mathbf{X} + n\lambda\mathbf{I}\right)\beta - 2\beta^T\mathbf{X}^T\mathbf{y} + \mathbf{y}^T\mathbf{y}\right] \tag{2}$$

- The gradient of $\mathcal{L}(\beta)$ is:

$$\nabla\mathcal{L} = \frac{2}{n}\left[\left(\mathbf{X}^T\mathbf{X} + n\lambda\mathbf{I}\right)\beta - \mathbf{X}^T\mathbf{y}\right] \tag{3}$$

Analytical solution of ridge regression

- The loss function of ridge regression:

$$\mathcal{L}(\beta) = \frac{1}{n}\left[\beta^T\left(\mathbf{X}^T\mathbf{X} + n\lambda\mathbf{I}\right)\beta - 2\beta^T\mathbf{X}^T\mathbf{y} + \mathbf{y}^T\mathbf{y}\right] \tag{2}$$

- The gradient of $\mathcal{L}(\beta)$ is:

$$\nabla\mathcal{L} = \frac{2}{n}\left[\left(\mathbf{X}^T\mathbf{X} + n\lambda\mathbf{I}\right)\beta - \mathbf{X}^T\mathbf{y}\right] \tag{3}$$

- Let us set $\nabla\mathcal{L} = \mathbf{0}$ to derive the analytical solution of ridge regression $\hat{\beta}$:

$$\hat{\beta} = \left(\mathbf{X}^T\mathbf{X} + n\lambda\mathbf{I}\right)^{-1}\mathbf{X}^T\mathbf{y} \tag{4}$$

Analytical solution of ridge regression

- The loss function of ridge regression:

$$\mathcal{L}(\beta) = \frac{1}{n}\left[\beta^T\left(\mathbf{X}^T\mathbf{X} + n\lambda\mathbf{I}\right)\beta - 2\beta^T\mathbf{X}^T\mathbf{y} + \mathbf{y}^T\mathbf{y}\right] \tag{2}$$

- The gradient of $\mathcal{L}(\beta)$ is:

$$\nabla\mathcal{L} = \frac{2}{n}\left[\left(\mathbf{X}^T\mathbf{X} + n\lambda\mathbf{I}\right)\beta - \mathbf{X}^T\mathbf{y}\right] \tag{3}$$

- Let us set $\nabla\mathcal{L} = \mathbf{0}$ to derive the analytical solution of ridge regression $\hat{\beta}$:

$$\hat{\beta} = \left(\mathbf{X}^T\mathbf{X} + n\lambda\mathbf{I}\right)^{-1}\mathbf{X}^T\mathbf{y} \tag{4}$$

- Q: Why does the L2 penalty solve the overfitting problem?

1. Recap
2. Ridge regression
3. Ridge regression in Bayesian view
4. Linear regression in Bayesian view
5. Summary

Ridge regression in Bayesian view

Model and Priors:

Probabilistic model for linear regression with Gaussian priors of error and β

$$y = \beta^T \mathbf{x} + \epsilon$$

$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$

$$\beta \sim \mathcal{N}(0, \sigma_0^2)$$

Ridge regression in Bayesian view

Model and Priors:

Probabilistic model for linear regression with Gaussian priors of error and β

$$y = \beta^T \mathbf{x} + \epsilon$$
$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$
$$\beta \sim \mathcal{N}(0, \sigma_0^2)$$

- For a given β and given $\mathbf{x}^{(i)}$, the likelihood $P(y|\mathbf{x}^{(i)}; \beta)$ is:

$$y^{(i)} \sim \mathcal{N}(\beta^T \mathbf{x}^{(i)}, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(y - \beta^T \mathbf{x}^{(i)})^2}{2\sigma^2}\right\} \quad (5)$$

Ridge regression in Bayesian view

- For the entire training set $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$, the likelihood is:

$$P(\mathbf{y}|\mathbf{X}; \beta) = \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(y^{(i)} - \beta^T \mathbf{x}^{(i)})^2}{2\sigma^2}\right] \quad (6)$$

Ridge regression in Bayesian view

- For the entire training set $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$, the likelihood is:

$$P(\mathbf{y}|\mathbf{X};\beta) = \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(y^{(i)} - \beta^T \mathbf{x}^{(i)})^2}{2\sigma^2}\right] \tag{6}$$

- The prior of β is $\beta \sim \mathcal{N}(0, \sigma_0^2)$, which is

$$P(\beta) = \frac{1}{\sqrt{2\pi\sigma_0^2}} \exp\left[-\frac{\|\beta\|^2}{2\sigma_0^2}\right] \tag{7}$$

Ridge regression in Bayesian view

- For the entire training set $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$, the likelihood is:

$$P(\mathbf{y}|\mathbf{X}; \beta) = \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(y^{(i)} - \beta^T \mathbf{x}^{(i)})^2}{2\sigma^2}\right] \tag{6}$$

- The prior of β is $\beta \sim \mathcal{N}(0, \sigma_0^2)$, which is

$$P(\beta) = \frac{1}{\sqrt{2\pi\sigma_0^2}} \exp\left[-\frac{\|\beta\|^2}{2\sigma_0^2}\right] \tag{7}$$

- The posterior is:

$$P(\beta|\mathbf{y}; \mathbf{X}) \propto P(\mathbf{y}|\mathbf{X}; \beta) P(\beta) \tag{8}$$

MAP to estimate $\hat{\beta}_{MAP}$ for ridge regression

$$
\begin{aligned}
\hat{\beta}_{MAP} &= \arg\max_{\beta} \log P(\mathbf{y}|\mathbf{X};\beta) + \log p(\beta) \\
&= \arg\max_{\beta} \sum_{i=1}^{N} \log\left(\frac{1}{\sigma\sqrt{2\pi}}\right) + \log\left(\frac{1}{\sigma_0\sqrt{2\pi}}\right) \\
&\quad - \left[\sum_{i=1}^{N} \frac{(y^{(i)} - \beta^T \mathbf{x}^{(i)})^2}{2\sigma^2} + \frac{\|\beta\|^2}{2\sigma_0^2}\right] \\
&= \arg\min_{\beta} \sum_{i=1}^{N} \frac{\left(y^{(i)} - \beta^T \mathbf{x}^{(i)}\right)^2}{2\sigma^2} + \frac{\|\beta\|^2}{2\sigma_0^2} \\
&= \arg\min_{\beta} \sum_{i=1}^{N} \left(y^{(i)} - \beta^T \mathbf{x}^{(i)}\right)^2 + \frac{\sigma^2}{\sigma_0^2}\|\beta\|^2
\end{aligned}
$$

MAP (with Gaussian prior) = ridge regression (LSE with L2 reg.)

- We have derived the MAP as Eq. (9):

$$
\begin{aligned}
\hat{\beta} &= \arg\min_{\beta} \sum_{i=1}^{N} \left(y^{(i)} - \beta^T \mathbf{x}^{(i)}\right)^2 + \frac{\sigma^2}{\sigma_0^2} \|\beta\|^2 \\
&= \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \frac{\sigma^2}{\sigma_0^2} \|\beta\|^2
\end{aligned} \tag{9}
$$

MAP (with Gaussian prior) = ridge regression (LSE with L2 reg.)

- We have derived the MAP as Eq. (9):

$$\begin{aligned}\hat{\beta} &= \arg\min_{\beta} \sum_{i=1}^{N} \left(y^{(i)} - \beta^T \mathbf{x}^{(i)}\right)^2 + \frac{\sigma^2}{\sigma_0^2} \|\beta\|^2 \\ &= \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \frac{\sigma^2}{\sigma_0^2} \|\beta\|^2\end{aligned} \tag{9}$$

- Recall the loss function (with L2 regularization) of ridge regression in Eq. (1):

$$\mathcal{L}(\beta) = \frac{1}{n} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda \|\beta\|^2$$

MAP (with Gaussian prior) = ridge regression (LSE with L2 reg.)

- We have derived the MAP as Eq. (9):

$$\begin{aligned}\hat{\beta} &= \arg\min_{\beta} \sum_{i=1}^{N} \left(y^{(i)} - \beta^T \mathbf{x}^{(i)}\right)^2 + \frac{\sigma^2}{\sigma_0^2} \|\beta\|^2 \\ &= \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \frac{\sigma^2}{\sigma_0^2} \|\beta\|^2\end{aligned} \tag{9}$$

- Recall the loss function (with L2 regularization) of ridge regression in Eq. (1):

$$\mathcal{L}(\beta) = \frac{1}{n} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda \|\beta\|^2$$

- When both the noise of the model and prior of parameters are followed Gaussian distribution, MAP is the same as minimizing MSE loss with L2 regularization.

MAP (with Laplace prior) = LASSO regression (LSE with L1 reg.)

- Loss function (with L1 regularization) of LASSO regression:

$$\mathcal{L}(\beta) = (1/n) \left\| \boldsymbol{y} - \boldsymbol{X}\beta \right\|^2 + \lambda \left\| \beta \right\|_1$$

MAP (with Laplace prior) = LASSO regression (LSE with L1 reg.)

- Loss function (with L1 regularization) of LASSO regression:

$$\mathcal{L}(\beta) = (1/n) \|\boldsymbol{y} - \boldsymbol{X}\beta\|^2 + \lambda \|\beta\|_1$$

- In Bayesian view, what prior distribution for β would derive L1 reg. in LASSO?

MAP (with Laplace prior) = LASSO regression (LSE with L1 reg.)

- Loss function (with L1 regularization) of LASSO regression:

$$\mathcal{L}(\beta) = (1/n) \left\| \boldsymbol{y} - \boldsymbol{X}\beta \right\|^2 + \lambda \left\| \beta \right\|_1$$

- In Bayesian view, what prior distribution for β would derive L1 reg. in LASSO?
- Laplace prior for β: $P(\beta|\mu, b) = \frac{1}{2b} \exp(-\frac{|\beta-\mu|}{b})$

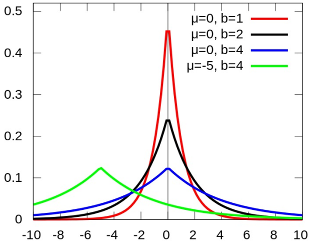

图 2: Laplace prior

Linear regression in Bayesian view

The probabilistic model for linear regression:

$$\text{Model: } y = \beta_0 + \beta_1 x + \cdots + \beta_p x_p + \epsilon = \beta^T \mathbf{x} + \epsilon$$

Ordinary linear regression: Gaussian prior of ϵ and uniform prior of β.

$$\text{Gaussian Prior of } \epsilon : \epsilon \sim \mathcal{N}(0, \sigma^2)$$
$$\text{Uniform Prior of } \beta : \beta \sim \mathit{Uniform}(-\infty, +\infty)$$

Ridge regression: Gaussian prior of ϵ and Gaussian prior of β.

$$\text{Gaussian Prior of } \epsilon : \epsilon \sim \mathcal{N}(0, \sigma^2)$$
$$\text{Gaussian Prior of } \beta : \beta \sim \mathcal{N}(0, \sigma_0^2)$$

The bias-variance trade-off

What is the bias-variance trade-off?

- **Bias**: Bias is the error due to overly simplistic assumptions in the learning algorithm. High bias can cause an algorithm to miss the relevant relations between features and target outputs (*underfitting*). Essentially, a high-bias model is one that pays little attention to the training data and oversimplifies the model, which leads to a high error on both training and test data.
- **Variance**: Variance is the error due to too much complexity in the learning algorithm. High variance can cause an algorithm to model the random noise in the training data, rather than the intended outputs (*overfitting*). A high-variance model pays too much attention to the training data and does not generalize well to new data.

Homework set 1: theory part

- For a given β and given $\mathbf{x}^{(i)}$, the likelihood $P(y|\mathbf{x}^{(i)};\beta)$ is:

$$y^{(i)} \sim \mathcal{N}(\beta^T\mathbf{x}^{(i)}, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(y-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right\} \quad (10)$$

Homework set 1: theory part

- For a given β and given $\mathbf{x}^{(i)}$, the likelihood $P(y|\mathbf{x}^{(i)};\beta)$ is:

$$y^{(i)} \sim \mathcal{N}(\beta^T\mathbf{x}^{(i)}, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(y-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right\} \quad (10)$$

- For the entire training set $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^N$,

$$P(\mathbf{y}|\mathbf{X};\beta) = \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(y^{(i)}-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right] \quad (11)$$

Homework set 1: theory part

- For a given β and given $\mathbf{x}^{(i)}$, the likelihood $P(y|\mathbf{x}^{(i)};\beta)$ is:

$$y^{(i)} \sim \mathcal{N}(\beta^T\mathbf{x}^{(i)}, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(y-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right\} \quad (10)$$

- For the entire training set $(\mathbf{X},\mathbf{y}) = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$,

$$P(\mathbf{y}|\mathbf{X};\beta) = \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(y^{(i)}-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right] \quad (11)$$

- **Homework set 1**

Homework set 1: theory part

- For a given β and given $\mathbf{x}^{(i)}$, the likelihood $P(y|\mathbf{x}^{(i)};\beta)$ is:

$$y^{(i)} \sim \mathcal{N}(\beta^T \mathbf{x}^{(i)}, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(y-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right\} \quad (10)$$

- For the entire training set $(\mathbf{X},\mathbf{y}) = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$,

$$P(\mathbf{y}|\mathbf{X};\beta) = \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(y^{(i)}-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right] \quad (11)$$

- **Homework set 1**
  1. Prior: $\beta \sim Laplace(\mu_0, b_0)$. Derive the posterior $P(\beta|\mathbf{y},\mathbf{X})$
     Hint: $P(\beta|\mathbf{y};\mathbf{X}) \propto P(\mathbf{y}|\mathbf{X},\beta)P(\beta)$

Homework set 1: theory part

- For a given β and given $\mathbf{x}^{(i)}$, the likelihood $P(y|\mathbf{x}^{(i)};\beta)$ is:

$$y^{(i)} \sim \mathcal{N}(\beta^T\mathbf{x}^{(i)}, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(y-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right\} \quad (10)$$

- For the entire training set $(\mathbf{X},\mathbf{y}) = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^N$,

$$P(\mathbf{y}|\mathbf{X};\beta) = \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(y^{(i)}-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right] \quad (11)$$

- **Homework set 1**
  1. Prior: $\beta \sim Laplace(\mu_0, b_0)$. Derive the posterior $P(\beta|\mathbf{y},\mathbf{X})$
     Hint: $P(\beta|\mathbf{y};\mathbf{X}) \propto P(\mathbf{y}|\mathbf{X},\beta)P(\beta)$
  2. Derive $\hat{\beta}_{MAP}$ using maximum a posterior

Homework set 1: theory part

- For a given β and given $\mathbf{x}^{(i)}$, the likelihood $P(y|\mathbf{x}^{(i)};\beta)$ is:

$$y^{(i)} \sim \mathcal{N}(\beta^T\mathbf{x}^{(i)},\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left\{-\frac{(y-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right\} \quad (10)$$

- For the entire training set $(\mathbf{X},\mathbf{y}) = \{(\mathbf{x}^{(i)},y^{(i)})\}_{i=1}^N$,

$$P(\mathbf{y}|\mathbf{X};\beta) = \prod_{i=1}^{N}\frac{1}{\sigma\sqrt{2\pi}}\exp\left[-\frac{(y^{(i)}-\beta^T\mathbf{x}^{(i)})^2}{2\sigma^2}\right] \quad (11)$$

- **Homework set 1**
  1. Prior: $\beta \sim Laplace(\mu_0, b_0)$. Derive the posterior $P(\beta|\mathbf{y},\mathbf{X})$
     Hint: $P(\beta|\mathbf{y};\mathbf{X}) \propto P(\mathbf{y}|\mathbf{X},\beta)P(\beta)$
  2. Derive $\hat{\beta}_{MAP}$ using maximum a posterior
  3. Update β with Gradient Descent

Summary of Lecture 5

- Bayesian view: The parameters β are not constant, which have their prior distributions $P(\beta)$.
  Now we assume $\beta \sim \mathcal{N}(0, \sigma_0^2)$, it becomes ridge regression.
- Probabilistic model for ridge regression:

$$y = \beta^T \mathbf{x} + \epsilon,$$
$$\epsilon \sim \mathcal{N}\left(0, \sigma^2\right),$$
$$\beta \sim \mathcal{N}(0, \sigma_0^2).$$

- Likelihood:

$$P(y \mid \beta) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{\left(y - \beta^T \mathbf{x}\right)^2}{2\sigma^2}\right\}$$

- Bayes' theorem:

$$P(\beta|\mathbf{y}) = \frac{P(\mathbf{y}|\beta)P(\beta)}{P(\mathbf{y})}$$

Summary of Lecture 5

- Let's derive the MAP:

$$
\begin{aligned}
\hat{\beta}_{MAP} &= \arg\max_{\beta} \log P(\mathbf{y}|\mathbf{X};\beta) + \log p(\beta) \\
&= \arg\max_{\beta} \sum_{i=1}^{N} \log\left(\frac{1}{\sigma\sqrt{2\pi}}\right) + \log\left(\frac{1}{\sigma_0\sqrt{2\pi}}\right) \\
&\quad - \left[\sum_{i=1}^{N} \frac{(y^{(i)} - \beta^T \mathbf{x}^{(i)})^2}{2\sigma^2} + \frac{\|\beta\|^2}{2\sigma_0^2}\right] \\
&= \arg\min_{\beta} \sum_{i=1}^{N} \left(y^{(i)} - \beta^T \mathbf{x}^{(i)}\right)^2 + \frac{\sigma^2}{\sigma_0^2}\|\beta\|^2 \\
&= \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \frac{\sigma^2}{\sigma_0^2}\|\beta\|^2
\end{aligned}
$$