Statistical Learning II

Lecture 6 - Ridge regression

Bruno Loureiro
@ CSD, DI-ENS & CNRS

brloureiro@gmail.com

DL3 IASO, Université Paris Dauphine-PSL
16.10.2024

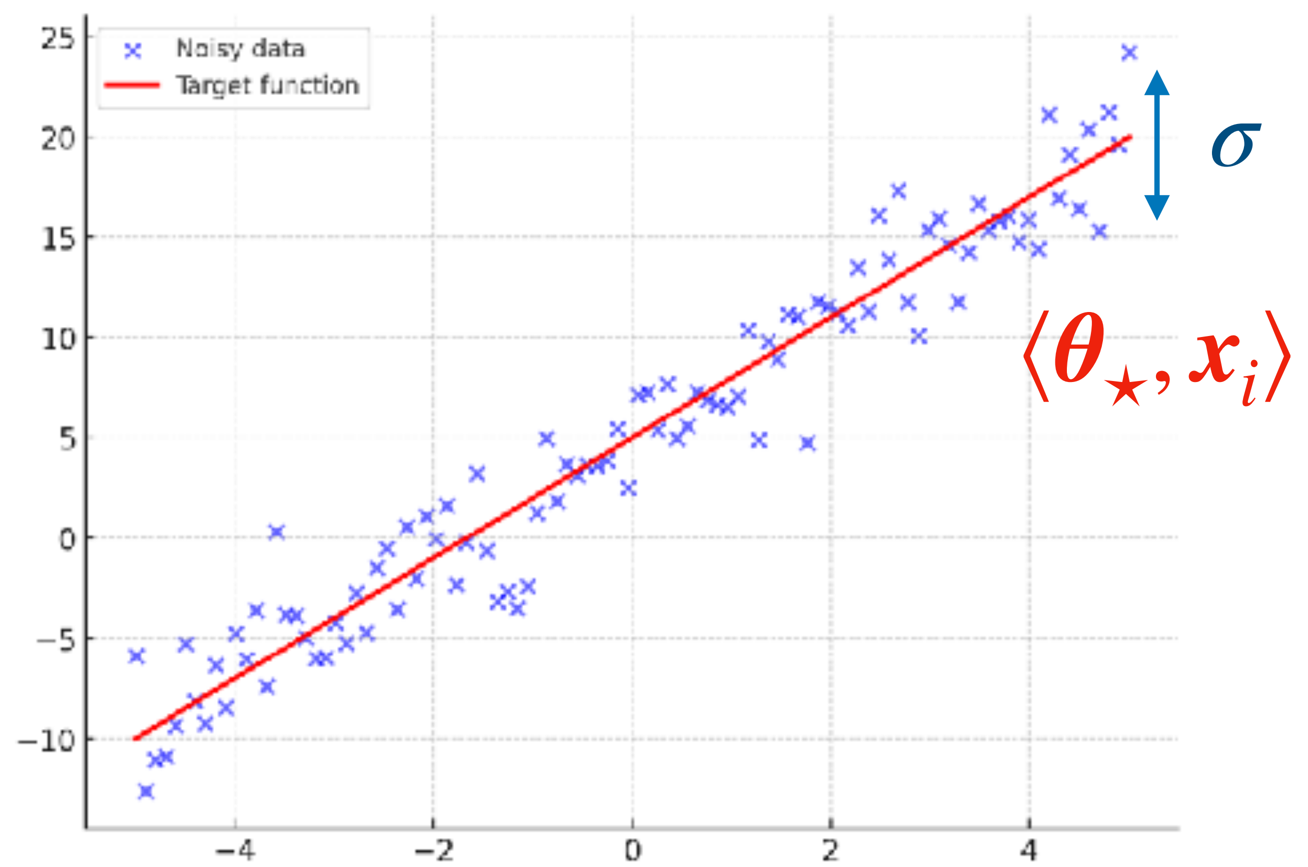
Assumptions

We now assume the following data generative model:

$$y_i = \langle \boldsymbol{\theta}_\star, \boldsymbol{x}_i \rangle + \varepsilon_i$$

With:

- Fixed $\boldsymbol{\theta}_\star \in \mathbb{R}^d$ and $\boldsymbol{x}_i \in \mathbb{R}^d$ "fixed design"
- $\mathbb{E}[\varepsilon_i] = 0$ and $\mathbb{E}[\varepsilon_i^2] = \sigma^2 < \infty$



Decomposition of OLS

$$\hat{\boldsymbol{\theta}}_{OLS}(\boldsymbol{X}, \boldsymbol{y}) = \boldsymbol{\theta}_\star + \frac{1}{n}\hat{\boldsymbol{\Sigma}}_n^{-1}\boldsymbol{X}^\top \boldsymbol{\varepsilon}$$

"signal" "noise"

In particular:

- Bias: $\mathbb{E}_{\boldsymbol{\varepsilon}}\left[\hat{\boldsymbol{\theta}}_{OLS}(\boldsymbol{X}, \boldsymbol{y})\right] = \boldsymbol{\theta}_\star$ "Unbiased"
- Variance: $\mathrm{Var}_{\boldsymbol{\varepsilon}}\left[\hat{\boldsymbol{\theta}}_{OLS}(\boldsymbol{X}, \boldsymbol{y})\right] = \frac{\sigma^2}{n}\hat{\boldsymbol{\Sigma}}_n^{-1}$

Informally, if $\hat{\boldsymbol{\Sigma}}_n \to \boldsymbol{\Sigma}$ a rank d matrix as $n \to \infty$, then:

$$\hat{\boldsymbol{\theta}}_{OLS} \to \boldsymbol{\theta}_\star \quad \text{as } n \to \infty$$

"Consistency"

Risk of OLS

Therefore, we have the following final result for the excess risk of OLS

$$\mathbb{E}_{\boldsymbol{\varepsilon}}\left[\mathscr{R}(\hat{\boldsymbol{\theta}}_{OLS})\right] - \sigma^2 = \sigma^2 \frac{d}{n}$$

Remarks:

- Excess risk is proportional to the noise level $\mathbb{E}[\varepsilon^2] = \sigma^2$.
- Excess risk is proportional to the data dimension.
- To achieve excess risk $\Delta\mathscr{R} < \delta$, need:

$$n > \frac{\sigma^2 d}{\delta}$$

samples.

Bias-variance decomposition

Generally, if we have a data generative model for the training data $\mathscr{D} = \{(\boldsymbol{x}_i, y_i) \in \mathbb{R}^{d+1} : i = 1, \ldots, n\}$:

$$y_i = f_\star(\boldsymbol{x}) + \varepsilon_i = \text{signal} + \text{noise}$$

With $\mathbb{E}[\varepsilon] = 0$ and $\mathbb{E}[\varepsilon^2] = \sigma^2$

Bias-variance decomposition

Generally, if we have a data generative model for the training data $\mathscr{D} = \{(\boldsymbol{x}_i, y_i) \in \mathbb{R}^{d+1} : i = 1, \ldots, n\}$:

$$y_i = f_\star(\boldsymbol{x}) + \varepsilon_i = \text{signal} + \text{noise}$$

With $\mathbb{E}[\varepsilon] = 0$ and $\mathbb{E}[\varepsilon^2] = \sigma^2$, we can decompose the excess risk:

$$\mathbb{E}_{\boldsymbol{\varepsilon}}[\mathscr{R}(\hat{\boldsymbol{\theta}})] - \sigma^2 = \mathbb{E}\left[(f_\star(\boldsymbol{x}) - f_{\hat{\theta}}(\boldsymbol{x})^2\right]$$

Bias-variance decomposition

Generally, if we have a data generative model for the training data $\mathscr{D} = \{(\boldsymbol{x}_i, y_i) \in \mathbb{R}^{d+1} : i = 1,\ldots,n\}$:

$$y_i = f_\star(\boldsymbol{x}) + \varepsilon_i = \text{signal} + \text{noise}$$

With $\mathbb{E}[\varepsilon] = 0$ and $\mathbb{E}[\varepsilon^2] = \sigma^2$, we can decompose the excess risk:

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\varepsilon}}[\mathscr{R}(\hat{\boldsymbol{\theta}})] - \sigma^2 &= \mathbb{E}\left[(f_\star(\boldsymbol{x}) - f_{\hat{\theta}}(\boldsymbol{x})^2\right] \\
&= \mathbb{E}\left[(f_\star(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(x)] + \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})] - f_{\hat{\theta}}(\boldsymbol{x}))^2\right]
\end{aligned}$$

Bias-variance decomposition

Generally, if we have a data generative model for the training data $\mathscr{D} = \{(\boldsymbol{x}_i, y_i) \in \mathbb{R}^{d+1} : i = 1, \ldots, n\}$:

$$y_i = f_\star(\boldsymbol{x}) + \varepsilon_i \; = \text{signal} + \text{noise}$$

With $\mathbb{E}[\varepsilon] = 0$ and $\mathbb{E}[\varepsilon^2] = \sigma^2$, we can decompose the excess risk:

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\varepsilon}}[\mathscr{R}(\hat{\boldsymbol{\theta}})] - \sigma^2 &= \mathbb{E}\left[(f_\star(\boldsymbol{x}) - f_{\hat{\theta}}(\boldsymbol{x})^2\right] \\
&= \mathbb{E}\left[(f_\star(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(x)] + \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})] - f_{\hat{\theta}}(\boldsymbol{x}))^2\right] \\
&= \mathbb{E}\left[(f_\star(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(x)])^2\right] + \mathbb{E}\left[(\mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})] - f_{\hat{\theta}}(\boldsymbol{x}))^2\right]
\end{aligned}$$

Bias-variance decomposition

Generally, if we have a data generative model for the training data $\mathscr{D} = \{(\boldsymbol{x}_i, y_i) \in \mathbb{R}^{d+1} : i = 1,\ldots,n\}$:

$$y_i = f_\star(\boldsymbol{x}) + \varepsilon_i = \text{signal} + \text{noise}$$

With $\mathbb{E}[\varepsilon] = 0$ and $\mathbb{E}[\varepsilon^2] = \sigma^2$, we can decompose the excess risk:

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\varepsilon}}[\mathscr{R}(\hat{\boldsymbol{\theta}})] - \sigma^2 &= \mathbb{E}\left[(f_\star(\boldsymbol{x}) - f_{\hat{\theta}}(\boldsymbol{x}))^2\right] \\
&= \mathbb{E}\left[(f_\star(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(x)] + \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})] - f_{\hat{\theta}}(\boldsymbol{x}))^2\right] \\
&= \mathbb{E}\left[(f_\star(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(x)])^2\right] + \mathbb{E}\left[(\mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})] - f_{\hat{\theta}}(\boldsymbol{x}))^2\right]
\end{aligned}$$

$$= \text{bias}^2 + \text{variance}$$

Bias-variance decomposition

$$\mathbb{E}_{\boldsymbol{\varepsilon}}[\mathscr{R}(\hat{\boldsymbol{\theta}})] - \sigma^2 = \mathscr{B} + \mathscr{V}$$

$$\mathscr{B} = \mathbb{E}\left[(f_\star(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})])^2\right]$$

$$\mathscr{V} = \mathbb{E}\left[(\mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})] - f_{\hat{\theta}}(\boldsymbol{x}))^2\right]$$

Bias-variance decomposition

$$\mathbb{E}_{\boldsymbol{\varepsilon}}[\mathscr{R}(\hat{\boldsymbol{\theta}})] - \sigma^2 = \mathscr{B} + \mathscr{V}$$

$$\mathscr{B} = \mathbb{E}\left[(f_\star(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})])^2\right]$$

$$\mathscr{V} = \mathbb{E}\left[(\mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})] - f_{\hat{\theta}}(\boldsymbol{x}))^2\right]$$

Recall the the approximation + estimation decomposition from lecture 3:

$$\mathscr{R}(\theta) - \mathscr{R}_\star = \left(\mathscr{R}(\theta) - \inf_{\theta' \in \Theta} \mathscr{R}(\theta')\right) + \left(\inf_{\theta' \in \Theta} \mathscr{R}(\theta') - \mathscr{R}_\star\right)$$

Bias-variance decomposition

$$\mathbb{E}_{\boldsymbol{\varepsilon}}[\mathscr{R}(\hat{\boldsymbol{\theta}})] - \sigma^2 = \mathscr{B} + \mathscr{V}$$

$$\mathscr{B} = \mathbb{E}\left[\left(f_\star(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})]\right)^2\right]$$

$$\mathscr{V} = \mathbb{E}\left[\left(\mathbb{E}_{\boldsymbol{\varepsilon}}[f_{\hat{\theta}}(\boldsymbol{x})] - f_{\hat{\theta}}(\boldsymbol{x})\right)^2\right]$$

Recall the the approximation + estimation decomposition from lecture 3:

$$\mathscr{R}(\theta) - \mathscr{R}_\star = \left(\mathscr{R}(\theta) - \inf_{\theta' \in \Theta} \mathscr{R}(\theta')\right) + \left(\inf_{\theta' \in \Theta} \mathscr{R}(\theta') - \mathscr{R}_\star\right)$$

For the OLS setting from before $(\mathrm{rank}(X) = d < n)$:

$$\mathbb{E}[f_{\hat{\theta}}(\boldsymbol{x})] = \langle \boldsymbol{\theta}_\star, \boldsymbol{x} \rangle = f_\star(\boldsymbol{x}) \quad \Rightarrow \quad \mathscr{B} = 0 \qquad \mathscr{V} = \sigma^2 \frac{d}{n}$$

Marvels and pitfalls of OLS

To summarise, the OLS estimator $\hat{\boldsymbol{\theta}}_{\mathrm{OLS}}(\boldsymbol{X}, y) = \boldsymbol{X}^{+}\boldsymbol{y}$:

- Can only fit linear functions.
- For $n > d$, has low bias $\mathscr{B} = 0$
- When, $n \gg d$, has low variance $\mathscr{V} = \sigma^2 \dfrac{d}{n}$

Marvels and pitfalls of OLS

To summarise, the OLS estimator $\hat{\boldsymbol{\theta}}_{\text{OLS}}(\boldsymbol{X}, y) = \boldsymbol{X}^{+}\boldsymbol{y}$:

- Can only fit linear functions.
- For $n > d$, has low bias $\mathscr{B} = 0$
- When, $n \gg d$, has low variance $\mathscr{V} = \sigma^2 \dfrac{d}{n}$

But what about $n \approx d$? Consider for instance $n = d$.

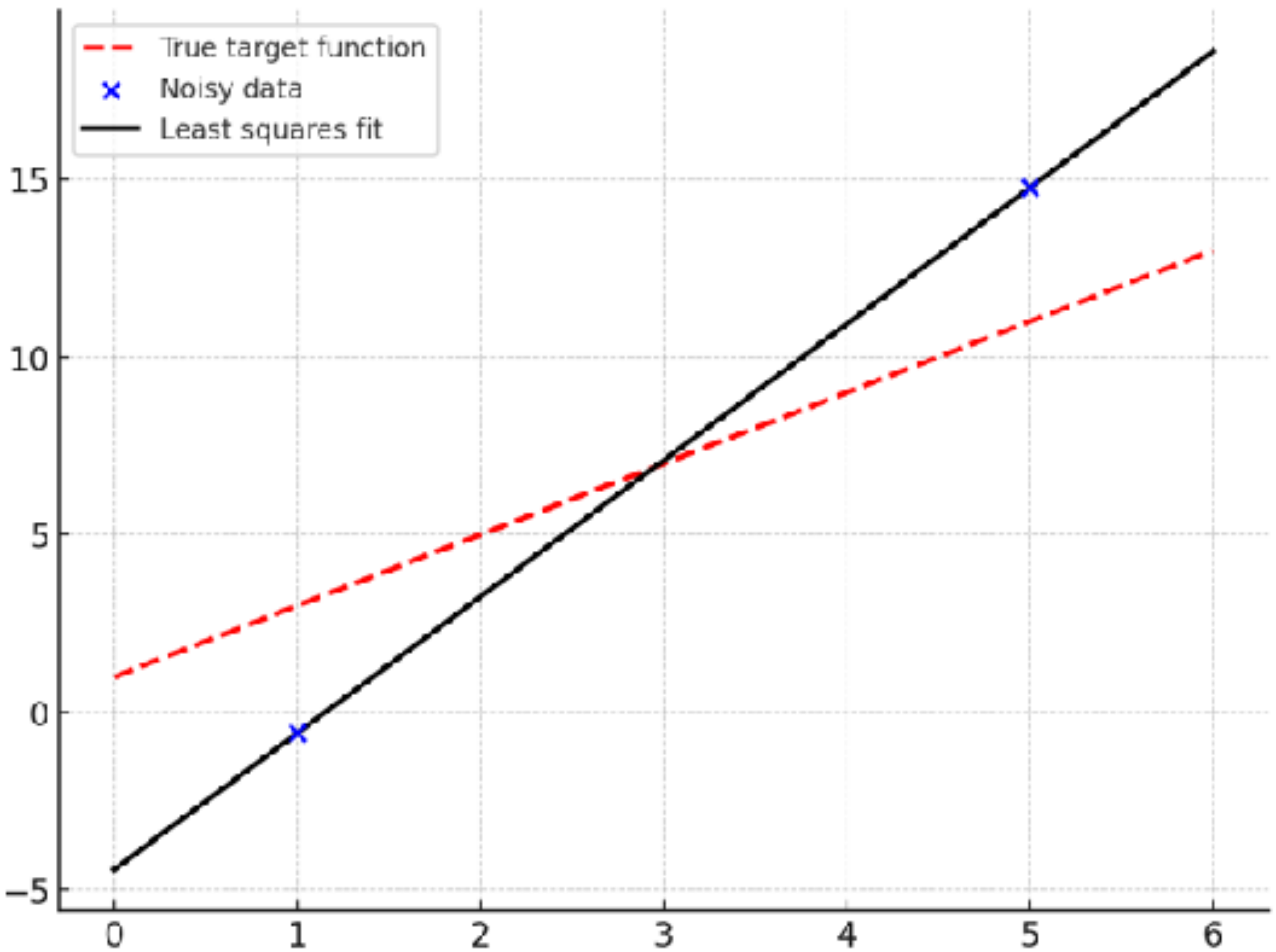
$\boldsymbol{X} \in \mathbb{R}^{d \times d}$ is invertible $\Rightarrow$ $\boldsymbol{y} = \boldsymbol{X}\hat{\boldsymbol{\theta}}_{\text{OLS}}$ interpolates the training data.

Marvels and pitfalls of OLS

But what about $n \approx d$? Consider for instance $n = d$.

$$\boldsymbol{X} \in \mathbb{R}^{d\times d} \text{ is invertible} \quad \Rightarrow \quad \boldsymbol{y} = \boldsymbol{X}\hat{\boldsymbol{\theta}}_{\mathrm{OLS}}$$

interpolates the training data.



$$\mathbb{E}_{\boldsymbol{\varepsilon}}[\mathcal{R}(\hat{\boldsymbol{\theta}}_{\mathrm{OLS}})] = 2\sigma^2$$

$$\hat{\mathcal{R}}_n(\hat{\boldsymbol{\theta}}_{\mathrm{OLS}}) = 0$$

The test error above is valid for the fixed design.

Marvels and pitfalls of OLS

Recall that:

$$\hat{\boldsymbol{\theta}}_{OLS}(\boldsymbol{X}, \boldsymbol{y}) = \boldsymbol{\theta}_{\star} + \frac{1}{n}\hat{\boldsymbol{\Sigma}}_n^{-1}\boldsymbol{X}^\top\boldsymbol{\varepsilon}$$

Marvels and pitfalls of OLS

Recall that:

$$\begin{aligned}\hat{\boldsymbol{\theta}}_{OLS}(\boldsymbol{X}, \boldsymbol{y}) &= \boldsymbol{\theta}_{\star} + \frac{1}{n}\hat{\boldsymbol{\Sigma}}_n^{-1}\boldsymbol{X}^{\top}\boldsymbol{\varepsilon} \\ &= \boldsymbol{\theta}_{\star} + \sum_{j=1}^{d} \frac{1}{\sigma_j}\langle \boldsymbol{u}_j, \boldsymbol{\varepsilon}\rangle \boldsymbol{v}_j\end{aligned}$$

Marvels and pitfalls of OLS

Recall that:

$$\hat{\boldsymbol{\theta}}_{OLS}(\boldsymbol{X}, \boldsymbol{y}) = \boldsymbol{\theta}_\star + \frac{1}{n}\hat{\boldsymbol{\Sigma}}_n^{-1}\boldsymbol{X}^\top \boldsymbol{\varepsilon}$$
$$= \boldsymbol{\theta}_\star + \sum_{j=1}^{d} \frac{1}{\sigma_j}\langle \boldsymbol{u}_j, \boldsymbol{\varepsilon} \rangle \boldsymbol{v}_j$$

Hence:
- signal is stronger in directions with larger s.v.
- noise dominates directions with smaller s.v.

OLS has larger variance for data with small "effective dimension".

What to do?

Classical strategies to mitigate variance:

- Dimensionality reduction: PCA, random projections (sketching), etc.

- Variable subset selection: Stepwise selection, best Subset Selection, etc.

- Regularisation: ridge, LASSO, etc.