Chapter 2
Linear Least Squares Regression

Ernest K. Ryu
Seoul National University

Mathematical Machine Learning Theory
Spring 2024

Why learn about linear least squares?

Linear least squares (LS) is a classical topic within the realm of classical statistics. Why learn LS when we can learn about the more general machinery involving Rademacher complexity?

Informative of what is achievable in the general learning case.

LS analysis plays a crucial role in kernel methods.

Outline

Linear learning with finite nonlinear features

Least squares objective and its solution

Statistical properties

Linear learning with nonlinear features

Consider the setup with $\phi\colon \mathcal{X} \to \mathbb{R}^d$, where d may be smaller or larger than the "dimension" of $\mathcal{X}$. (We later consider infinite d.)

Consider

$$\underset{\theta}{\text{minimize}} \quad \mathop{\mathbb{E}}_{(X,Y)\sim P}[\ell(f_\theta(X), Y)],$$

where f_θ is a *linear*[1] *prediction function*

$$f_\theta(\cdot) = \langle \theta, \phi(\cdot) \rangle = \sum_{i=1}^{d} \theta_i \phi_i(\cdot)$$

and $\langle \cdot, \cdot \rangle$ denotes the standard inner product in $\mathbb{R}^d$.

Equivalently, consider the dataset

$$(\breve{X}_1, Y_1), \ldots, (\breve{X}_N, Y_N),$$

with $\breve{X}_i = \phi(X_i)$, and $f_\theta(X_i) = \langle \theta, \breve{X}_i \rangle$.

[1] Linear in the parameters θ, but nonlinear in the input X.

Absorbing bias into linear weights

What if we want a bias? So, what if we want to learn

$$f_{\theta,b}(\cdot) = \langle \theta, \phi(\cdot) \rangle + b.$$

Define

$$\tilde{\phi}(\cdot) = \begin{bmatrix} \phi(\cdot) \\ 1 \end{bmatrix} \in \mathbb{R}^{d+1}, \qquad \tilde{\theta} = \begin{bmatrix} \theta \\ b \end{bmatrix} \in \mathbb{R}^{d+1}$$

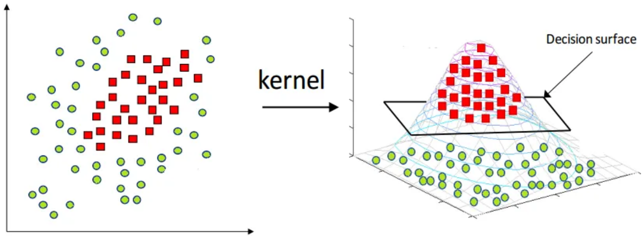
and note

$$\tilde{f}_{\tilde{\theta}}(\cdot) = \langle \tilde{\theta}, \tilde{\phi}(\cdot) \rangle = f_{\theta,b}(\cdot).$$

Trick: Absorb bias into linear weights.
WLOG, consider $f_\theta(\cdot) = \langle \theta, \phi(\cdot) \rangle$ without biases.

Decision boundaries linear in ϕ, nonlinear in X

Linear classifiers yield decision boundaries that are linear *in the features*.



Most ML tasks are nonlinear in X, and features nonlinear in X are needed to perform classification well.

Feature engineering

Feature engineering is the task of choosing (often hand-crafting) ϕ for a given ML task.

There was a time when ML was primarily about feature engineering.[2]
In modern deep learning, features are learned. (More on this soon.)

The output dimension d of ϕ can be lower or higher than the "dimension" of $\mathcal{X}$. Usually you want nonlinear but informative features of $\mathcal{X}$.

[2] One can argue that in modern machine learning *practice*, feature engineering is still the main engineering challenge.

Outline

Linear learning with finite nonlinear features

Least squares objective and its solution

Statistical properties

Linear least squares

Let $X_1, \ldots, X_N \in \mathcal{X}$ and $Y_1, \ldots, Y_N \in \mathcal{Y} = \mathbb{R}$ such that $(X_i, Y_i) \sim P$ IID for $i = 1, \ldots, N$. Consider the square loss

$$\mathcal{R}[f] = \mathbb{E}[(f(X) - Y)^2].$$

The Bayes optimal estimator is

$$f_\star(X) = \mathop{\mathbb{E}}_{Y \sim p_{Y\,|\,X}}[Y \,|\, X].$$

Of course, $f_\star$ depends on the joint distribution P.

In the context of linear least squares, we consider the linear function class

$$\mathcal{F} = \{f_\theta(x) = \theta^\mathsf{T}\phi(x) \,|\, \theta \in \Theta\},$$

where $\phi\colon \mathcal{X} \to \mathbb{R}^d$ is some feature map. In general, we expect $f_\star \notin \mathcal{F}$. In this sense, $\mathcal{F}$ is a *misspecified model*.

Linear least squares

We consider the squared loss, leading to

$$\underset{\theta \in \mathbb{R}^d}{\text{minimize}} \quad \frac{1}{2} \sum_{i=1}^{N} (f_\theta(X_i) - Y_i)^2$$

which is equivalent to

$$\underset{\theta \in \mathbb{R}^d}{\text{minimize}} \quad \frac{1}{2} \sum_{i=1}^{N} (\theta^\intercal \phi(X_i) - Y_i)^2$$

which is, in turn, equivalent to

$$\underset{\theta \in \mathbb{R}^d}{\text{minimize}} \quad \tfrac{1}{2} \|\Phi\theta - Y\|^2$$

where

$$\Phi = \begin{bmatrix} \phi(X_1)^\intercal \\ \vdots \\ \phi(X_N)^\intercal \end{bmatrix} \in \mathbb{R}^{N \times d}, \qquad Y = \begin{bmatrix} Y_1 \\ \vdots \\ Y_N \end{bmatrix} \in \mathbb{R}^N.$$

Least-norm-least-squares solution

Theorem

Consider the linear least squares optimization problem

$$\underset{\theta \in \mathbb{R}^d}{\text{minimize}} \quad \frac{1}{2}\|\Phi\theta - Y\|^2,$$

where $\Phi \in \mathbb{R}^{N \times d}$ *and* $Y \in \mathbb{R}^N$. *Then,*

$$\theta^\star = \Phi^\dagger Y$$

is a solution (global minimizer) of the least squares problem. Let r *be the rank of* Φ. *If* $d = r \leq N$, *then* $\theta^\star$ *is the unique solution. Otherwise,* $\theta^\star$ *is not the unique solution, but it is the least-norm solution (achieving minimum value of* $\frac{1}{2}\|\Phi\theta - Y\|^2$ *while having smallest* $\|\theta\|^2$*).*

Proof. Since $\|\cdot\|$ is unitarily invariant,

$$\underset{\theta\in\mathbb{R}^d}{\text{minimize}} \quad \tfrac{1}{2}\|U\Sigma V^\intercal\theta - Y\|^2$$

is equal to

$$\underset{\theta\in\mathbb{R}^d}{\text{minimize}} \quad \tfrac{1}{2}\|\Sigma V^\intercal\theta - U^\intercal Y\|^2 + \tfrac{1}{2}\|\tilde{U}^\intercal Y\|^2,$$

where $\tilde{U} \in \mathbb{R}^{N\times(N-d)}$ contains orthonormal columns such that $[U\,\tilde{U}] \in \mathbb{R}^{N\times N}$ is an orthonormal matrix. In turn, this is equivalent to

$$\underset{\theta\in\mathbb{R}^d}{\text{minimize}} \quad \tfrac{1}{2}\|\Sigma V^\intercal\theta - U^\intercal Y\|^2.$$

In turn, this is equivalent to

$$\underset{\substack{\theta_1\in\mathcal{R}(V)\\ \theta_2\in\mathcal{R}(V)^\perp}}{\text{minimize}} \quad \tfrac{1}{2}\|\Sigma V^\intercal(\theta_1+\theta_2) - U^\intercal Y\|^2.$$

In turn, this is equivalent to

$$\underset{\substack{\theta_1\in\mathcal{R}(V)\\ \theta_2\in\mathcal{R}(V)^\perp}}{\text{minimize}} \quad \tfrac{1}{2}\|\Sigma V^\intercal\theta_1 - U^\intercal Y\|^2.$$

At $\Sigma V^\intercal \theta_1 = U^\intercal Y$, we achieve global optimality, so

$$V^\intercal \theta_1^\star = \Sigma^{-1} U^\intercal Y$$

Since $\theta_1^\star \in \mathcal{R}(V)$, we have $V V^\intercal \theta_1^\star = \theta_1^\star$, and we conclude

$$\theta_1^\star = \underbrace{V \Sigma^{-1} U^\intercal}_{=\Phi^\dagger} Y.$$

On the other hand, an arbitrary $\theta_2^\star \in \mathcal{R}(V)^\perp$ will not affect the objective value. The norm of the solution $\theta^\star$ given by

$$\|\theta^\star\|^2 = \|\theta_1^\star\|^2 + \|\theta_2^\star\|^2,$$

which is mimimized when $\theta_2 = 0$. When $d = r \leq N$, we have $\mathcal{R}(V)^\perp = \{0\}$, and $\theta_2^\star = 0$ and $\theta^\star$ is uniquely determined. □

LS solution with full column rank

Corollary

If $\Phi \in \mathbb{R}^{N \times d}$ *has full column rank (which requires that* $N \geq d$*), then*

$$\Phi^{\dagger} = (\Phi^{\intercal}\Phi)^{-1}\Phi^{\intercal},$$

and $\theta^{\star} = \Phi^{\dagger} Y$ *provides the unique solution.*

Proof. $\Phi^{\dagger} = (\Phi^{\intercal}\Phi)^{-1}\Phi^{\intercal}$ follows from the compact SVD. □

LS solution with full row rank

Corollary

If $\Phi \in \mathbb{R}^{N \times d}$ has full row rank (which requires that $N \leq d$), then

$$\Phi^{\dagger} = \Phi^{\intercal}(\Phi\Phi^{\intercal})^{-1}$$

and $\theta^{\star} = \Phi^{\dagger} Y$ provides the least-norm solution.

Proof. $\Phi^{\dagger} = \Phi^{\intercal}(\Phi\Phi^{\intercal})^{-1}$ follows from the compact SVD. □

Geometric interpretation of LS solution

Lemma

When Φ has full column rank, the vector of predictions

$$\Phi\hat{\theta} = \Phi(\Phi^\intercal\Phi)^{-1}\Phi^\intercal Y$$

is the orthogonal projection of Y onto $\mathcal{R}(\Phi)$.

Proof. Follows from simple arguments using the SVD. □

Thus, we can interpret the LS solution as doing the following:

1. Compute $\bar{Y} = \mathrm{Proj}_{\mathcal{R}(\Phi)}(Y)$.
2. Solve the linear system $\bar{Y} = \Phi\theta$, which has a unique solution.

Outline

Linear learning with finite nonlinear features

Least squares objective and its solution

Statistical properties

Fixed vs. random design setups

Consider

$$Y_i = \theta_\star^\intercal \phi(X_i) + \varepsilon_i$$

for $i = 1, \ldots, N$. Assume that $\varepsilon_1, \ldots, \varepsilon_N$ is an IID sequence such that

$$\mathbb{E}[\varepsilon_i] = 0, \qquad \mathbb{E}[\varepsilon_i^2] = \sigma^2$$

for $i = 1, \ldots, N$.

There are two settings we consider

- Fixed design: $X_1, \ldots, X_N$ is fixed (non-random).
- Random design: $X_1, \ldots, X_N$ is IID random (and independent from $\varepsilon_1, \ldots, \varepsilon_N$).

The random design setting is more realistic in machine learning[3] but the fixed design setting is easier to analyze. (Whether $X_1, \ldots, X_N$ is fixed or random has no affect on training. Only generalization is affected.)

---

[3]The fixed design setting is more relevant in statistics, where $X_1, \ldots, X_N$ are chosen/designed for efficient learning.

Fixed vs. random design setups

The model

$$Y_i = \theta_\star^\intercal \phi(X_i) + \varepsilon_i,$$

is a well-specified assumption. In general, additional approximation error is incurred because of a misspecified model.

Define the uncentered *empirical covariance matrix* as

$$\widehat{\Sigma} = \frac{1}{N}\Phi^\intercal \Phi = \frac{1}{N}\sum_{i=1}^{N} \phi(X_i)\phi(X_i)^\intercal, \qquad \Phi = \begin{bmatrix} \phi(X_1)^\intercal \\ \vdots \\ \phi(X_N)^\intercal \end{bmatrix} \in \mathbb{R}^{N\times d}.$$

For the fixed design setup, $\widehat{\Sigma} \in \mathbb{R}^{d\times d}$ is a fixed, deterministic matrix. In the random design setup, $\widehat{\Sigma} \to \Sigma =$ as $N \to \infty$, where

$$\Sigma = \mathop{\mathbb{E}}_{X}[\phi(X)\phi(X)^\intercal]$$

is the uncentered *covariance matrix*.

Fixed vs. random design setups

For reference, we formally state the definitions of the two setups. Let

$$\widehat{\Sigma} = \frac{1}{N}\Phi^\intercal \Phi, \qquad \Phi = \begin{bmatrix} \phi(X_1)^\intercal \\ \vdots \\ \phi(X_N)^\intercal \end{bmatrix} \in \mathbb{R}^{N\times d}.$$

Fixed design setup:

- $X_1, \dots, X_N$ is fixed and given.
- Φ has full column rank and $\widehat{\Sigma}$ is invertible.
- $\varepsilon_1, \dots, \varepsilon_N$ is an IID sequence such that $\mathbb{E}[\varepsilon_i] = 0$ and $\mathbb{E}[\varepsilon_i^2] = \sigma^2$ for $i = 1, \dots, N$.
- $Y_i = \theta_\star^\intercal \phi(X_i) + \varepsilon_i$ for $i = 1, \dots, N$.

Random design setup:

- $X_1, \dots, X_N$ is a random IID sequence
- Φ has full column rank and $\widehat{\Sigma}$ is invertible with probability 1.
- $\varepsilon_1, \dots, \varepsilon_N$ is an IID sequence such that $\mathbb{E}[\varepsilon_i] = 0$ and $\mathbb{E}[\varepsilon_i^2] = \sigma^2$ for $i = 1, \dots, N$. Also, $X_1, \dots, X_N$ and $\varepsilon_1, \dots, \varepsilon_N$ are independent.
- $Y_i = \theta_\star^\intercal \phi(X_i) + \varepsilon_i$ for $i = 1, \dots, N$.

Risk for fixed design

In the fixed design setting, we use the risk

$$\mathcal{R}(\theta) = \mathop{\mathbb{E}}_{\varepsilon_1,\dots,\varepsilon_N}\left[\frac{1}{N}\|\Phi\theta - Y\|^2\right]$$

Denote

$$\mathcal{R}^\star = \inf_\theta \mathcal{R}(\theta).$$

In the fixed design setup, the goal is to learn θ that performs well on $X_1, \dots, X_N$ and only $X_1, \dots, X_N$. The uncertainty comes from the noisiness of the labels $Y_1, \dots, Y_N$, originating from $\varepsilon_1, \dots, \varepsilon_N$.

Risk for fixed design

Lemma

In the fixed design setting,

$$\mathcal{R}(\theta) - \mathcal{R}_\star = \|\theta - \theta_\star\|^2_{\widehat{\Sigma}}.$$

($\|v\|^2_{\widehat{\Sigma}} = v^\mathsf{T} \widehat{\Sigma} v$ *is called the (squared) Mahalanobis distance.)*

Proof.

$$\begin{aligned}
\mathcal{R}(\theta) &= \mathop{\mathbb{E}}_{\varepsilon_1,\dots,\varepsilon_N}\left[\tfrac{1}{N}\|\Phi\theta - Y\|^2\right] = \mathop{\mathbb{E}}_{\varepsilon_1,\dots,\varepsilon_N}\left[\tfrac{1}{N}\|\Phi\theta - \Phi\theta_\star - \varepsilon\|^2\right] \\
&= \mathop{\mathbb{E}}_{\varepsilon_1,\dots,\varepsilon_N}\left[\tfrac{1}{N}\|\Phi(\theta - \theta_\star) - \varepsilon\|^2\right] \\
&\overset{(*)}{=} \tfrac{1}{N}(\theta - \theta_\star)^\mathsf{T}\Phi^\mathsf{T}\Phi(\theta - \theta_\star) + \mathop{\mathbb{E}}_{\varepsilon_1,\dots,\varepsilon_N}\left[\tfrac{1}{N}\|\varepsilon\|^2\right] \\
&= \|\theta - \theta_\star\|^2_{\widehat{\Sigma}} + \sigma^2
\end{aligned}$$

In step (*), we use the fact that ε has zero-mean and the other term is deterministic. Finally, note

$$\mathcal{R}_\star = \inf_\theta \mathcal{R}(\theta) = \sigma^2.$$

□

Bias-variance decomposition for fixed design

Lemma

If $\hat{\theta} \in \mathbb{R}^d$ *be random. In the fixed design setting, we have*

$$\mathbb{E}[\mathcal{R}(\hat{\theta})] - \mathcal{R}_\star = \underbrace{\|\mathbb{E}[\hat{\theta}] - \theta_\star\|_{\widehat{\Sigma}}^2}_{\textit{bias}} + \underbrace{\mathbb{E}[\|\hat{\theta} - \mathbb{E}[\hat{\theta}]\|_{\widehat{\Sigma}}^2]}_{\textit{variance}},$$

where, to clarify, $\mathbb{E}[\cdot] = \mathbb{E}_{\hat{\theta}}[\cdot]$.

Proof.

$$\begin{aligned}
\mathop{\mathbb{E}}_{\hat{\theta}}[\mathcal{R}(\hat{\theta}) - \mathcal{R}_\star] &= \mathop{\mathbb{E}}_{\hat{\theta}}[\|\hat{\theta} - \mathbb{E}[\hat{\theta}] + \mathbb{E}[\hat{\theta}] - \theta_\star\|_{\widehat{\Sigma}}^2] \\
&= \mathop{\mathbb{E}}_{\hat{\theta}}[\|\hat{\theta} - \mathbb{E}[\hat{\theta}]\|_{\widehat{\Sigma}}^2] + \mathop{\mathbb{E}}_{\hat{\theta}}[\|\mathbb{E}[\hat{\theta}] - \theta_\star\|_{\widehat{\Sigma}}^2] + 2\mathop{\mathbb{E}}_{\hat{\theta}}\left[(\hat{\theta} - \mathbb{E}[\hat{\theta}])^\mathsf{T}\widehat{\Sigma}(\mathbb{E}[\hat{\theta}] - \theta_\star)\right] \\
&= \text{variance} + \text{bias} + 2(\mathop{\mathbb{E}}_{\hat{\theta}}[\hat{\theta}] - \mathbb{E}[\hat{\theta}])^\mathsf{T}\widehat{\Sigma}(\mathbb{E}[\hat{\theta}] - \theta_\star) \\
&= \text{bias} + \text{variance}.
\end{aligned}$$

□

Statistical properties of LS estimator for fixed design

Theorem

In the fixed design setting, the least-square estimator

$$\hat{\theta} = (\Phi^\mathsf{T}\Phi)^{-1}\Phi^\mathsf{T}Y = \widehat{\Sigma}^{-1}\frac{1}{N}\Phi^\mathsf{T}Y$$

satisfies

$$\mathbb{E}[\hat{\theta}] = \theta_\star$$

and

$$\mathrm{Cov}[\hat{\theta}] = \mathbb{E}[(\hat{\theta} - \theta_\star)(\hat{\theta} - \theta_\star)^\mathsf{T}] = \frac{\sigma^2}{N}\widehat{\Sigma}^{-1}.$$

($\widehat{\Sigma}^{-1}$ is often called the *precision matrix*.)

Proof. First, note that

$$\hat{\theta} = (\Phi^\intercal \Phi)^{-1}\Phi^\intercal Y = (\Phi^\intercal \Phi)^{-1}\Phi^\intercal(\Phi\theta_\star + \varepsilon) = \theta_\star + (\Phi^\intercal \Phi)^{-1}\Phi^\intercal \varepsilon.$$

Then we have

$$\mathbb{E}[\hat{\theta}] = \theta_\star + (\Phi^\intercal \Phi)^{-1}\Phi^\intercal \mathbb{E}[\varepsilon] = \theta_\star$$

and

$$\begin{aligned} \mathrm{Cov}[\hat{\theta}] &= \mathbb{E}[(\hat{\theta} - \theta_\star)(\hat{\theta} - \theta_\star)^\intercal] = \mathbb{E}[(\Phi^\intercal \Phi)^{-1}\Phi^\intercal \varepsilon\varepsilon^\intercal \Phi(\Phi^\intercal \Phi)^{-1}] \\ &= (\Phi^\intercal \Phi)^{-1}\Phi^\intercal \mathbb{E}[\varepsilon\varepsilon^\intercal]\Phi(\Phi^\intercal \Phi)^{-1} \\ &= \sigma^2(\Phi^\intercal \Phi)^{-1}\Phi^\intercal \Phi(\Phi^\intercal \Phi)^{-1} \\ &= \sigma^2(\Phi^\intercal \Phi)^{-1} = \frac{\sigma^2}{N}\widehat{\Sigma}^{-1}. \end{aligned}$$

□

Excess risk of LS estimator for fixed design

Corollary

In the fixed design setting, the expected excess risk of the least-square estimator is

$$\mathbb{E}[\mathcal{R}(\hat{\theta})] - \mathcal{R}^\star = \frac{\sigma^2 d}{N}.$$

Proof. From the previous theorem, we have

$$\mathbb{E}[\hat{\theta}] = \theta_\star, \qquad \mathrm{Cov}[\hat{\theta}] = \frac{\sigma^2}{N}\widehat{\Sigma}^{-1}.$$

Plug this into the bias-variance decomposition of a previous lemma to get

$$\begin{aligned}
\mathbb{E}[\mathcal{R}(\hat{\theta}) - \mathcal{R}_\star] &= \|\mathbb{E}[\hat{\theta}] - \theta_\star\|_{\widehat{\Sigma}}^2 + \mathbb{E}[\|\hat{\theta} - \mathbb{E}[\hat{\theta}]\|_{\widehat{\Sigma}}^2] \\
&= 0 + \mathbb{E}[\|\hat{\theta} - \theta_\star\|_{\widehat{\Sigma}}^2] = \mathbb{E}[\mathrm{Tr}((\hat{\theta} - \theta_\star)^\mathsf{T}\widehat{\Sigma}(\hat{\theta} - \theta_\star))] \\
&= \mathrm{Tr}(\widehat{\Sigma}\mathbb{E}[(\hat{\theta} - \theta_\star)(\hat{\theta} - \theta_\star)^\mathsf{T}]) \\
&= \frac{\sigma^2}{N}\mathrm{Tr}(\widehat{\Sigma}\widehat{\Sigma}^{-1}) = \frac{\sigma^2}{N}\mathrm{Tr}(I) = \frac{\sigma^2 d}{N}.
\end{aligned}$$

□

Risk for random design

In the random design setting, we use the risk

$$\mathcal{R}(\theta) = \mathop{\mathbb{E}}_{X_1,\varepsilon_1}[(\phi(X_1)^\mathsf{T}\theta - Y_1)^2] = \mathop{\mathbb{E}}_{\substack{X_1,\dots,X_N\\ \varepsilon_1,\dots,\varepsilon_N}}[\tfrac{1}{N}\|\Phi\theta - Y\|^2].$$

In the random design setup, the goal is to learn θ that performs well on a new data-label pair. The uncertainty comes from the noisiness of the labels $Y_1, \dots, Y_N$, originating from $\varepsilon_1, \dots, \varepsilon_N$, and from the randomness the data $X_1, \dots, X_N$.

Lemma

In the random design setting,

$$\mathcal{R}(\theta) - \mathcal{R}_\star = \|\theta - \theta_\star\|_{\Sigma}^2.$$

Proof.

$$\begin{aligned}\mathcal{R}(\theta) &= \mathop{\mathbb{E}}_{X,\varepsilon}[(\phi(X)^\mathsf{T}(\theta - \theta_\star) - \varepsilon)^2] \\ &= (\theta - \theta_\star)^\mathsf{T} \mathop{\mathbb{E}}_{X}[\phi(X)\phi(X)^\mathsf{T}](\theta - \theta_\star) + \mathop{\mathbb{E}}_{\varepsilon}[\varepsilon^2] = \|\theta - \theta_\star\|_\Sigma^2 + \sigma^2. \quad \square\end{aligned}$$

Bias-variance decomposition for random design

Lemma
If $\hat{\theta} \in \mathbb{R}^d$ be random. In the random design setting, we have

$$\mathbb{E}[\mathcal{R}(\hat{\theta})] - \mathcal{R}_\star = \underbrace{\|\mathbb{E}[\hat{\theta}] - \theta_\star\|_{\Sigma}{}^2}_{\text{bias}} + \underbrace{\mathbb{E}[\|\hat{\theta} - \mathbb{E}[\hat{\theta}]\|_{\Sigma}{}^2]}_{\text{variance}},$$

where, to clarify, $\mathbb{E}[\cdot] = \mathbb{E}_{\hat{\theta}}[\cdot]$.

Proof. Same argument as in the fixed design case. □

Statistical properties of LS estimator for random design

Theorem

In the random design setting, the least-square estimator

$$\hat{\theta} = (\Phi^\intercal \Phi)^{-1} \Phi^\intercal Y$$

satisfies

$$\mathbb{E}[\hat{\theta}] = \theta_\star$$

and

$$\operatorname{Cov}[\hat{\theta}] = \mathbb{E}[(\hat{\theta} - \theta_\star)(\hat{\theta} - \theta_\star)^\intercal] = \frac{\sigma^2}{N} \mathbb{E}[\widehat{\Sigma}^{-1}].$$

Proof. Same argument as in the fixed design case. □

Excess risk of LS estimator for random design

Corollary

In the random design setting, the expected excess risk of the least-square estimator is

$$\mathbb{E}[\mathcal{R}(\hat{\theta})] - \mathcal{R}^\star = \frac{\sigma^2}{N}\mathbb{E}[\mathrm{Tr}(\Sigma\widehat{\Sigma}^{-1})].$$

Proof. From the previous theorem, we have

$$\mathbb{E}[\hat{\theta}] = \theta_\star, \qquad \mathrm{Cov}[\hat{\theta}] = \frac{\sigma^2}{N}\mathbb{E}[\widehat{\Sigma}^{-1}].$$

Plug this into the bias-variance decomposition of a previous lemma to get

$$\begin{aligned}
\mathbb{E}[\mathcal{R}(\hat{\theta}) - \mathcal{R}_\star] &= \|\mathbb{E}[\hat{\theta}] - \theta_\star\|_\Sigma^2 + \mathbb{E}[\|\hat{\theta} - \mathbb{E}[\hat{\theta}]\|_\Sigma^2] \\
&= 0 + \mathbb{E}[\|\hat{\theta} - \theta_\star\|_\Sigma^2] = \mathbb{E}[\mathrm{Tr}((\hat{\theta} - \theta_\star)^\intercal \Sigma(\hat{\theta} - \theta_\star))] \\
&= \mathrm{Tr}(\Sigma\mathbb{E}[(\hat{\theta} - \theta_\star)(\hat{\theta} - \theta_\star)^\intercal]) \\
&= \frac{\sigma^2}{N}\mathbb{E}[\mathrm{Tr}(\Sigma\widehat{\Sigma}^{-1})].
\end{aligned}$$

□

Excess risk of LS estimator for random design: Gaussian features

Corollary

In the random design setting, assume $\phi(X_1)$ is Gaussian with zero mean and a symmetric (strictly) positive definite covariance matrix Σ. Then

$$\mathbb{E}[\mathcal{R}(\hat{\theta})] - \mathcal{R}^\star = \frac{\sigma^2 d}{N - d - 1}.$$

Proof. For $i = 1, \ldots, N$, since $\phi(X_i)$ is Gaussian with covariance Σ,

$$Z_i = \Sigma^{-1/2}\phi(X_i)$$

is an IID Gaussian since $\mathbb{E}[Z_i] = \Sigma^{-1/2}\mathbb{E}[\phi(X_i)] = 0$ and

$$\mathbb{E}[Z_i Z_i^\intercal] = \Sigma^{-1/2}\mathbb{E}[\phi(X_i)\phi(X_i)^\intercal]\Sigma^{-1/2} = \Sigma^{-1/2}\Sigma\Sigma^{-1/2} = I.$$

Let

$$Z = \begin{bmatrix} Z_1^\intercal \\ \vdots \\ Z_N^\intercal \end{bmatrix} \in \mathbb{R}^{N\times d}, \qquad \Phi = \begin{bmatrix} \phi(X_1)^\intercal \\ \vdots \\ \phi(X_N)^\intercal \end{bmatrix} \in \mathbb{R}^{N\times d}.$$

Then $Z = \Phi\Sigma^{-1/2}$ and

$$\widehat{\Sigma} = \frac{1}{N}\Phi^\intercal\Phi = \frac{1}{N}\Sigma^{1/2}(Z^\intercal Z)\Sigma^{1/2}, \qquad \widehat{\Sigma}^{-1} = N\Sigma^{-1/2}(Z^\intercal Z)^{-1}\Sigma^{-1/2}$$

By the previous corollary, we have

$$\begin{aligned}\mathbb{E}[\mathcal{R}(\hat{\theta})] - \mathcal{R}^\star &= \frac{\sigma^2}{N}\mathbb{E}[\mathrm{Tr}(\Sigma\widehat{\Sigma}^{-1})] = \sigma^2\mathrm{Tr}(\Sigma\Sigma^{-1/2}\mathbb{E}[(Z^\intercal Z)^{-1}]\Sigma^{-1/2}) \\ &= \sigma^2\mathrm{Tr}(\mathbb{E}[(Z^\intercal Z)^{-1}]),\end{aligned}$$

where the Nd elements of $Z \in \mathbb{R}^{N\times d}$ are IID unit Gaussians. Then $(Z^\intercal Z)^{-1}$ is known to follow the *inverse Wishart distribution*, and it is known that

$$\mathbb{E}[(Z^\intercal Z)^{-1}] = \frac{1}{n-d-1}I.$$

Therefore,

$$\mathbb{E}[\mathcal{R}(\hat{\theta})] - \mathcal{R}^\star = \frac{\sigma^2 d}{N-d-1}.$$

□

Excess risk of LS estimator for random design

Lemma

In the random design setting, the expected excess risk of the least-square estimator conditioned on Φ is

$$\mathop{\mathbb{E}}_{\varepsilon}[\mathcal{R}(\hat{\theta}) - \mathcal{R}^\star \mid \Phi] = \frac{\sigma^2}{N}\mathrm{Tr}(\Sigma\widehat{\Sigma}^{-1}).$$

Proof. Recall that we had established $\mathcal{R}(\theta) - \mathcal{R}_\star = \|\theta - \theta_\star\|_\Sigma^2$. Plugging in $\hat{\theta} = \theta_\star + (\Phi^\intercal\Phi)^{-1}\Phi^\intercal\varepsilon$, we get

$$\mathcal{R}(\hat{\theta}) - \mathcal{R}_\star = \|(\Phi^\intercal\Phi)^{-1}\Phi^\intercal\varepsilon\|_\Sigma^2 = \varepsilon^\intercal\Phi(\Phi^\intercal\Phi)^{-1}\Sigma(\Phi^\intercal\Phi)^{-1}\Phi^\intercal\varepsilon$$

Then we have

$$\begin{aligned}
\mathop{\mathbb{E}}_{\varepsilon}[\mathcal{R}(\hat{\theta}) - \mathcal{R}^\star \mid \Phi] &= \mathop{\mathbb{E}}_{\varepsilon}[\mathrm{Tr}(\varepsilon^\intercal\Phi(\Phi^\intercal\Phi)^{-1}\Sigma(\Phi^\intercal\Phi)^{-1}\Phi^\intercal\varepsilon) \mid \Phi] \\
&= \mathrm{Tr}(\mathop{\mathbb{E}}_{\varepsilon}[\Phi(\Phi^\intercal\Phi)^{-1}\Sigma(\Phi^\intercal\Phi)^{-1}\Phi^\intercal\varepsilon\varepsilon^\intercal \mid \Phi]) \\
&= \sigma^2\mathrm{Tr}((\Phi^\intercal\Phi)^{-1}\Sigma(\Phi^\intercal\Phi)^{-1}\Phi^\intercal\Phi) \\
&= \frac{\sigma^2}{N}\mathrm{Tr}(\widehat{\Sigma}^{-1}\Sigma)
\end{aligned}$$

□

PAC bound of LS estimator for random design

Theorem

In the random design setting, assume there is a $\rho \geq 1$ *such that*

$$\mathbb{E}\big[\phi(X)^\intercal \Sigma^{-1} \phi(X) \phi(X) \phi(X)^\intercal\big] \preceq \rho \Sigma.$$

If $N \geq 5\rho \log(d/\delta)$*, then*

$$\Sigma^{1/2} \widehat{\Sigma}^{-1} \Sigma^{1/2} \preceq 4I$$

with probability $\geq 1 - \delta$.

PAC bound of LS estimator for random design: Discussion of assumption

Let $Z_i = \Sigma^{-1/2}\phi(X_i)$, so that $\mathbb{E}[Z_i Z_i^\intercal] = I$ for $i = 1, \dots, N$. Let

$$Z = \begin{bmatrix} Z_1^\intercal \\ \vdots \\ Z_N^\intercal \end{bmatrix} \in \mathbb{R}^{N \times d}.$$

Then, the assumption is equivalent to

$$\lambda_{\max}\big(\mathbb{E}[\|Z_i\|^2 Z_i Z_i^\intercal]\big) \leq \rho$$

In particular, this condition is implied if $\|Z_i\|^2 \leq \rho$ almost surely. When $Z_i \sim \mathcal{N}(0, I_{d \times d})$, then

$$\lambda_{\max}\big(\mathbb{E}[\|Z_i\|^2 Z_i Z_i^\intercal]\big) = 2 + d.$$

(Proof in homework.)

PAC bound of LS estimator for random design

Proof. Let $M_i = I - Z_i Z_i^\intercal$. Then, $\mathbb{E}[M_i] = 0$ and $\lambda_{\max}(M_i) \leq 1$. Also, $\mathbb{E}[M_i^2] = \mathbb{E}[\|Z_i\|^2 Z_i Z_i^\intercal] - I$, so

$$\lambda_{\max}(\mathbb{E}[M_i^2]) \leq \rho - 1 \leq \rho.$$

Since $\lambda_{\max}$ is convex (as you will show in your homework), Jensen's inequality implies

$$\lambda_{\max}\Big(\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[M_i^2]\Big) \leq \rho.$$

With the Matrix Bernstein's inequality, we have

$$\mathbb{P}\Big(\lambda_{\max}(I - \tfrac{1}{N}Z^\intercal Z) \geq \varepsilon\Big) \leq d\exp\Big(-\frac{N\varepsilon^2/2}{\rho + \varepsilon/3}\Big),$$

By plugging in $\varepsilon = 3/4$, setting the probability to δ, and solving for N, we get the stated condition $N \geq (32\rho/9 + 8/9)\log(d/\delta)$, which is implied by $N \geq 5\rho\log(d/\delta)$, since $\rho \geq 1$.

PAC bound of LS estimator for random design

So, with probability $\geq 1 - \delta$,

$$\frac{1}{N} Z^{\intercal} Z \succeq \frac{1}{4} I,$$

which is equivalent to

$$\begin{aligned} \Sigma^{-1/2} \widehat{\Sigma} \Sigma^{-1/2} &\succeq \frac{1}{4} I \\ \Sigma^{1/2} \widehat{\Sigma}^{-1} \Sigma^{1/2} &\preceq 4I. \end{aligned}$$

□

PAC bound of LS estimator for random design

Corollary

In the random design setting, assume there is a $\rho \geq 1$ *such that*

$$\mathbb{E}\left[\phi(X)^\intercal \Sigma^{-1} \phi(X) \phi(X) \phi(X)^\intercal\right] \preceq \rho \Sigma.$$

If $N \geq 5\rho \log(d/\delta)$, *then*

$$\mathcal{R}(\hat{\theta}) - \mathcal{R}^\star < \frac{4\sigma^2 d}{\delta N}$$

with probability $\geq (1-\delta)^2$.

Proof. By the previous theorem, with probability $\geq 1-\delta$, we get a "good" Φ such that $\mathbb{E}_\varepsilon[\mathcal{R}(\hat{\theta}) - \mathcal{R}^\star \mid \Phi] \leq \frac{4\sigma^2 d}{N}$. On this good event, we can apply Markov's inequality, to get

$$\mathbb{P}_{\varepsilon}(\mathcal{R}(\hat{\theta}) - \mathcal{R}^\star \geq \eta \mid \Phi) \leq \frac{4\sigma^2 d}{\eta N}.$$

We set the RHS equal to δ and solve to get $\eta = \frac{4\sigma^2 d}{\delta N}$. Then, the stated bound holds with probability $\geq (1-\delta)^2$. □

(δ-dependence can be improved with further assump., but we stop here.)