Lecture 2: Linear Methods for Regression

Readings: ESL (Ch. 3), ISL (Ch. 3, 6), Bach (Ch. 3); code

Soon Hoe Lim

August 28, 2025

Outline

Linear Regression Models

We have an input $X = (X_1, \ldots, X_p) \in \mathbb{R}^p$ and want to predict an output $Y \in \mathbb{R}$.

Definition 1: Linear Regression Model

The linear regression model has the form:

$$f(X) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j = X^T \beta.$$

Here, $X = (1, X_1, \ldots, X_p) \in \mathbb{R}^{p+1}$ is the augmented input vector, and $\beta = (\beta_0, \beta_1, \ldots, \beta_p) \in \mathbb{R}^{p+1}$ is the vector of model parameters (unknown).

- The linear model either assumes the regression function $\mathbb{E}[Y|X]$ is linear or that the linear model is a reasonable assumption.
- Note that f is linear in β. In general, it needs not be linear in X: $f(X) = \varphi(X)^T \beta$ for some feature map $\varphi : \mathbb{R}^{p+1} \to \mathbb{R}^{p+1}$.
- The ERM problem can be solved easily using linear algebra.

Ordinary Least Squares (OLS)

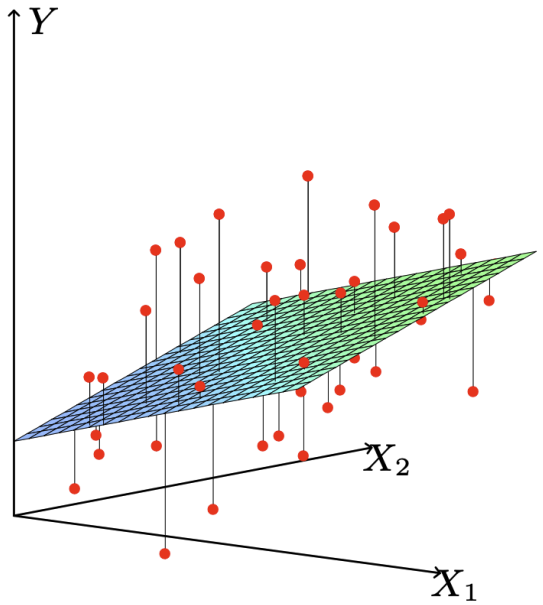
We fit the model by minimizing the **Residual Sum of Squares (RSS)**:

$$RSS(\beta) = \sum_{i=1}^{N}(y_i - f(x_i))^2 = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) = \|\mathbf{y} - \mathbf{X}\beta\|_2^2;$$

or, equivalently, minimizing the empirical risk for squared-error loss: $\hat{R}_N(\beta) = \frac{1}{N}RSS(\beta)$ (also denoted $MSE(\beta)$).

Denoting $d := p + 1$, here $\mathbf{X}$ is the $N \times d$ matrix (design matrix) with each row being an input row vector $(1, x_i^T)$, $\mathbf{y}$ is the column vector of d outputs, and β is the d-vector of parameters. The entire set of predictions is:

$$\hat{\mathbf{y}} = \mathbf{X}\beta \in \mathbb{R}^N.$$



Solution to the OLS

$$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{n} (y_i - x_i^T \beta)^2 = \arg\min_{\beta} (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta).$$

Proposition 1: OLS Solution

Assuming that $\mathbf{X}$ has full column rank so that $\mathbf{X}^T\mathbf{X}$ is invertible (why?), the unique solution of the ERM problem is given by:

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

Proof.

See blackboard. □

- We call $\hat{\beta}$ the OLS estimator.
- The predicted values at an input vector x_0 is $\hat{f}(x_0) = (1, x_0^T)^T \hat{\beta}$.
- The fitted vector at the training inputs is $\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} =: \mathbf{H}\mathbf{y}$, where $\hat{y}_i = \hat{f}(x_i)$.

❓ How should we compute $\hat{\beta}$ numerically when d is large?

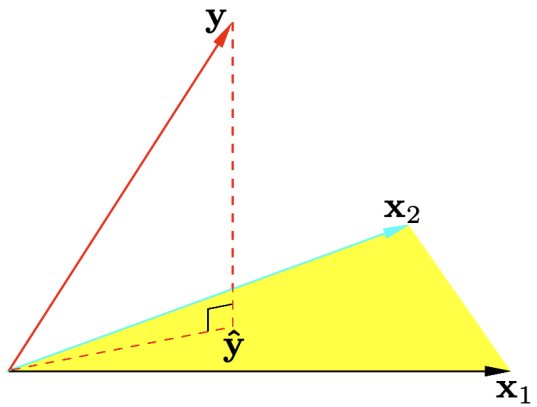
Geometric Interpretation of OLS Estimator

The OLS solution has an elegant geometric interpretation based on projections.

- We define the **projection matrix** (or "hat" matrix) as:

$$\Pi = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$$

- So, $\hat{\mathbf{y}} = \Pi\mathbf{y}$. This means $\hat{\mathbf{y}}$ is the **orthogonal projection** of $\mathbf{y}$ onto the column space of $\mathbf{X}$ (show this).
- The residual vector $\hat{\varepsilon} = \mathbf{y} - \hat{\mathbf{y}} = (\mathbf{I} - \Pi)\mathbf{y}$ is the projection of $\mathbf{y}$ onto the subspace orthogonal to the column space of $\mathbf{X}$.



- What if $\mathbf{X}$ does not have full column rank? $N \geq d$ vs. $N < d$ case?

Assumptions on True Data Distribution

To analyze the properties of our estimator $\hat{\beta}$, we assume a data-generating process:

$$y_i = \beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j^* + \varepsilon_i \quad \text{for } i = 1, \ldots, N,$$

for some $\beta^* \in \mathbb{R}^d$, where the errors ε_i are independent, and

- $\mathbb{E}[\varepsilon_i] = 0$
- $\text{Var}(\varepsilon_i) = \sigma^2$ for some constant $\sigma > 0$ (homoscedasticity).

We focus on the **fixed design** setting, where we treat x_i as fixed (non-random). Our goal is to minimize $R(\beta) = \frac{1}{N}\mathbb{E}_{\mathbf{y}}\|\mathbf{y} - \mathbf{X}\beta\|_2^2$, where $\mathbf{X}$ is deterministic.

A more realistic setting is the **random design** setting, where both the inputs and outputs are assumed to be random and sampled i.i.d.. But the analysis of $\hat{\beta}$ for this setting is more complicated (see Ch. 3.8 in Bach).

Risk Decomposition for OLS

Let $d := p + 1$ and denote by R^* the minimum value of $R(\beta) := \mathbb{E}_{\mathbf{y}} \frac{1}{N} \|\mathbf{y} - \mathbf{X}\beta\|_2^2$ over $\mathbb{R}^d$, we can show that it is attained at β^* and is equal to σ^2.

Proposition 2: Risk Decompositon for OLS

Assume $\mathbf{X}^T\mathbf{X}$ is invertible. Under the linear model and the fixed design assumptions made earlier, for any $\beta \in \mathbb{R}^d$, we have $R^* = \sigma^2$ and

$$R(\beta) - R^* = \|\beta - \beta^*\|_{\hat{\mathbf{\Sigma}}}^2,$$

where $\hat{\mathbf{\Sigma}} = \mathbf{X}^T\mathbf{X}/N$ and $\|\beta\|_Q^2 := \beta^T Q \beta$ (Mahalanobis distance).
Moreover, if $\hat{\beta}$ is a random variable, then

$$\mathbb{E}[R(\hat{\beta})] - R^* = \|\mathbb{E}[\hat{\beta}] - \beta^*\|_{\hat{\mathbf{\Sigma}}}^2 + \mathbb{E}[\|\hat{\beta} - \mathbb{E}[\hat{\beta}]\|_{\hat{\mathbf{\Sigma}}}^2].$$

Proof.

See blackboard. □

Statistical Properties of OLS Estimator

Proposition 3: Statistical Properties of OLS Estimator

Under the same set of assumptions used in Prop. 2,

- $\mathbb{E}[\hat{\beta}] = \beta^*$ (unbiased).
- $\text{Var}(\hat{\beta}) = (\mathbf{X}^T\mathbf{X})^{-1}\sigma^2$.

Proof.

See blackboard. □

We can estimate the variance σ^2 using $\hat{\sigma}^2 = \frac{1}{N-d} RSS(\hat{\beta})$ (check: $\mathbb{E}[\hat{\sigma}^2] = \sigma^2$).

If we further assume that the errors are Gaussian, i.e. $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$ for all i, then $\hat{\beta}$ is also Gaussian (with the above mean and variance) and is independent of $\hat{\sigma}^2$. Moreover, we can form tests of hypothesis and confidence intervals for the parameters β_j (see Ch. 3.2.1 in ESL).

Excess Risk of OLS Estimator

Theorem 1: Excess Risk of OLS Estimator

Under the same set of assumptions used in Prop. 2, the excess risk of the OLS estimator is $\mathbb{E}[R(\hat{\beta})] - R^* = \frac{\sigma^2 d}{N}$.

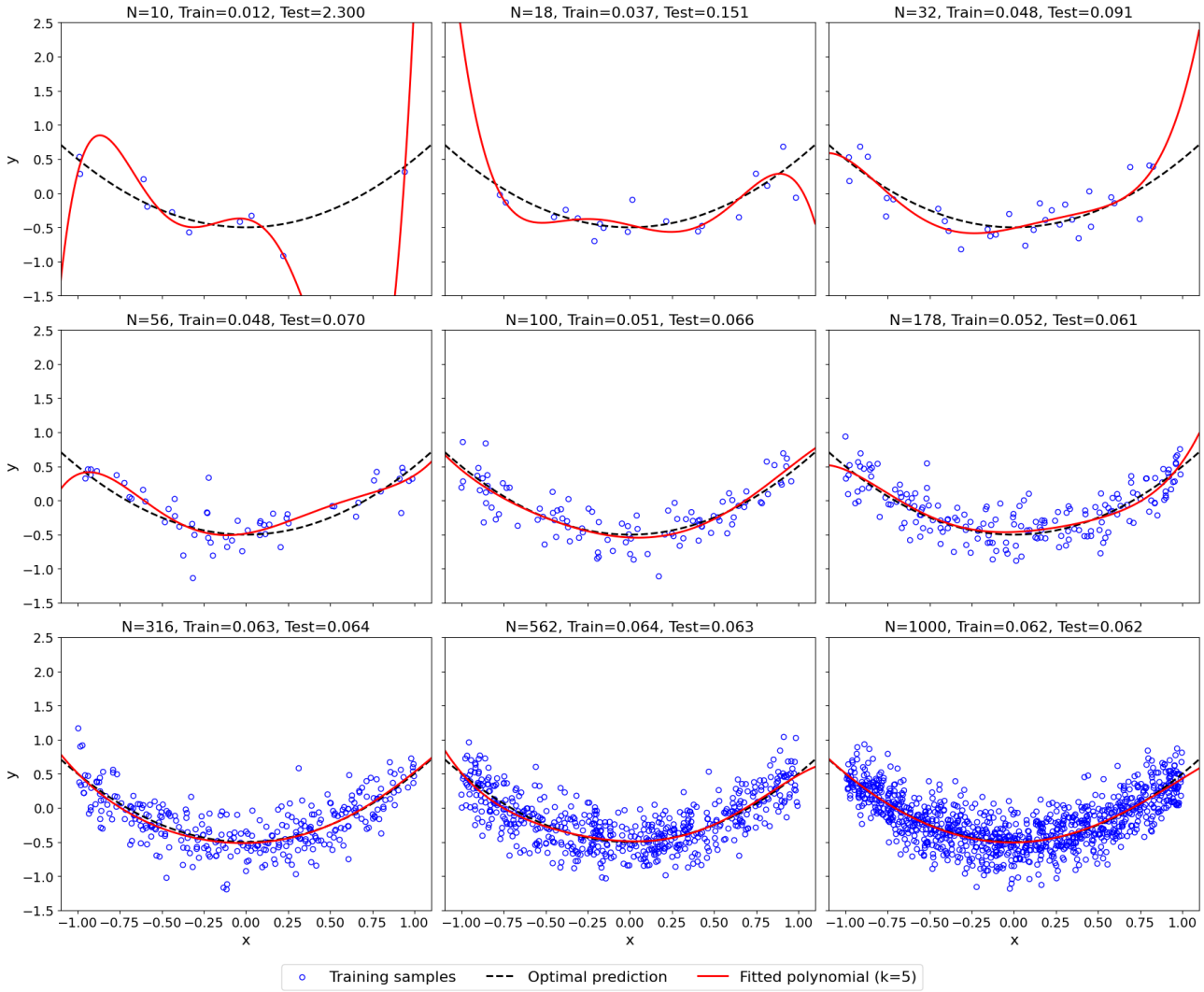
Proof.

See blackboard. □

- For OLS regression, the training error underestimates (on average) the testing error by a factor of $2\sigma^2 d/N$ (overfitting amount) – see Exercise 2.2.
- This result is for the fixed design setting. The random design setting is more involved mathematically (see Ch. 3.8 in Bach).
- The excess risk is large (compared to σ^2) in high dimensions ($d \geq N$).

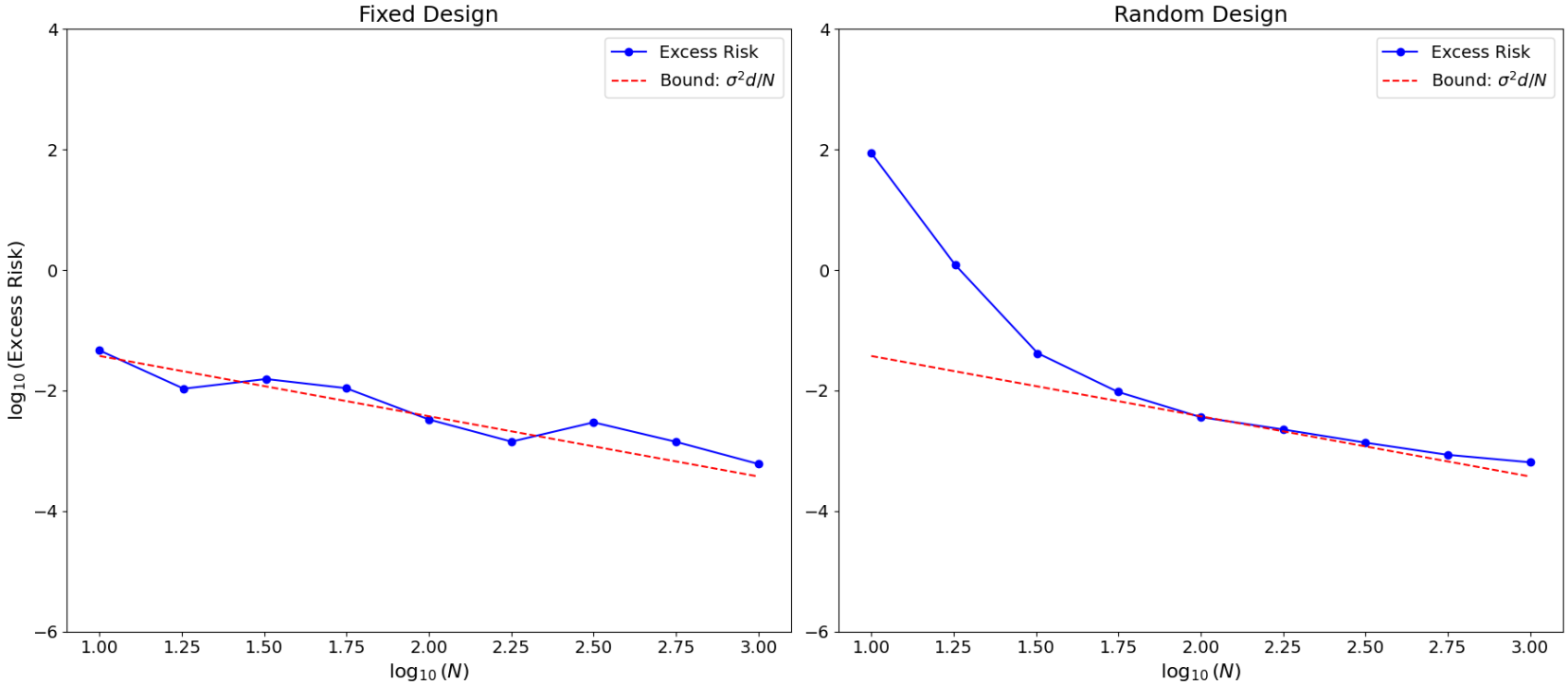
Example: Polynomial Regression in 1D (Bach Ch. 3.5.2)

Same data generation setup as the one from Lecture 1. Look at regression with polynomials of order $k = 5$ (so $d = 5$), using different training sample size N.



Expected Excess Risk vs. N

The expected excess risk is estimated using 32 replications of the experiment.



In-sample error vs. generalization error!

⚠ The bound seems to be valid only for large N in the random design setting.

The Gauss-Markov Theorem

Linear Unbiased Estimators

Consider any estimator $\tilde{\beta}$ that is a linear function of $\mathbf{y}$, written as $\tilde{\beta} = \mathbf{C}^T\mathbf{y}$.

- For $\tilde{\beta}$ to be an unbiased estimator of β, $\mathbf{C}$ must satisfy $\mathbf{C}^T\mathbf{X} = \mathbf{I}$ (why?).
- The OLS estimator $\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ is one such estimator.

Theorem 2: Gauss-Markov Theorem

The OLS estimator $\hat{\beta}$ has the smallest variance among all linear unbiased estimators.

This means that for any other linear unbiased estimator $\tilde{\beta}$, $\text{Var}(\tilde{\beta}) - \text{Var}(\hat{\beta})$ is a positive semi-definite matrix.

Proof.

See Exercise 2.1. □

When Does OLS Fail?

The variance of the coefficients is $\text{Var}(\hat{\beta}) = (\mathbf{X}^T\mathbf{X})^{-1}\sigma^2$. This can be large if:

- **Collinearity:** Predictors are highly correlated, making $\mathbf{X}^T\mathbf{X}$ nearly singular (ill-conditioned). Small changes in the data can lead to huge swings in the coefficients.
- **High Dimensions ($d > N$):** When the number of predictors exceeds the number of samples, $\mathbf{X}^T\mathbf{X}$ is singular and the OLS solution is no longer unique (it admits a linear subspace of solutions).

Also, when $d = N$, the OLS leads to a perfect fit (typically not good for generalization to unseen data).

The next families of methods—subset selection and shrinkage—are designed to address this problem by reducing model complexity and stabilizing the estimates.

Subset Selection

The OLS model can be improved in two main ways by selecting a subset of predictors (retain a subset, discard the rest):

Prediction Accuracy

- OLS can have high variance if d is large or predictors are collinear.
- Removing irrelevant predictors can reduce variance, potentially leading to a lower overall prediction error, even if it introduces a small amount of bias.

Interpretation

- A smaller model with only the most important predictors is easier to understand and explain.
- It provides a parsimonious description of the process.

Best-Subset and Stepwise Selection

Best-Subset Selection

Finds, for each $k \in \{0, 1, \ldots, p\}$, the subset of size k that gives the smallest RSS.

⚠ **Challenge:** Computationally infeasible for large p, as it requires checking 2^p models.

Stepwise Selection (Greedy Algorithms)

- **Forward-Stepwise:** Start with the null model. Iteratively add the one predictor that results in the largest decrease in RSS. A more constrained version is forward-stagewise regression.
- **Backward-Stepwise:** Start with the full model. Iteratively remove the one predictor that results in the smallest increase in RSS.

These are computationally efficient alternatives to best-subset selection but may not find the true best model.

Shrinkage Methods (Regularization)

Instead of hard selection, shrinkage methods regularize the coefficients by shrinking them towards zero, which is a way of controlling model variance.

Examples are ridge regression, the Lasso, least angle regression (not covered).

Definition 2: Ridge Regression

Ridge regression coefficients are the solution to the penalized ERM problem:

$$\hat{\beta}_{\lambda}^{\text{ridge}} = \arg\min_{\beta} \left\{ \frac{1}{N} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \|\beta\|_2^2 \right\}.$$

The term $\lambda\|\beta\|_2^2$ is an L2 penalty and $\lambda \geq 0$ is a complexity parameter.

See Exercise 2.3.

Ridge Regression (L2 Regularization)

Proposition 4: Ridge Least-Square Regression (RLS) Estimator

The solution has a closed form (stable even if $\mathbf{X}^T\mathbf{X}$ is singular):

$$\hat{\beta}_\lambda^{\text{ridge}} = \frac{1}{N}\left(\frac{\mathbf{X}^T\mathbf{X}}{N} + \lambda\mathbf{I}\right)^{-1}\mathbf{X}^T\mathbf{y} =: \mathbf{H}_\lambda\mathbf{y},$$

where $\mathbf{I}$ is the $d \times d$ identity matrix.

Proof.

See blackboard. □

- As $\lambda \to \infty$, the coefficients are shrunk to zero.
- Ridge is effective when many predictors have small-to-moderate effects. It does not set any coefficients to exactly zero (no variable selection).
- Simple relation between $\hat{\beta}$ and $\hat{\beta}_\lambda^{\text{ridge}}$ when the inputs are orthonormal (see Exercise 2.4).

Excess Risk of Ridge Least-Square Estimator

Proposition 5: Excess Risk of RLS Estimator

Under the same assumptions as the ones in Proposition 2-3 for the OLS estimator, the ridge least-square estimator has the following excess risk:

$$\mathbb{E}[R(\hat{\beta}_\lambda^{ridge})] - R^* = \underbrace{\lambda^2 (\beta^*)^T (\hat{\mathbf{\Sigma}} + \lambda I)^{-2} \hat{\mathbf{\Sigma}} \beta^*}_{\text{bias}} + \underbrace{\frac{\sigma^2}{N} \text{tr}[\hat{\mathbf{\Sigma}}^2 (\hat{\mathbf{\Sigma}} + \lambda I)^{-2}]}_{\text{variance}}.$$

Proof.

See Exercise 2.5. □

Based on the expression for the risk, we can tune λ to obtain a potentially better bound than with the OLS (e.g., Proposition 3.8 in Bach).

The Lasso (L1 Regularization)

The Lasso can perform both shrinkage and automatic variable selection.

Definition 3: The Lasso

The Lasso coefficients are the solution to:

$$\hat{\beta}_{\lambda}^{\text{lasso}} = \arg\min_{\beta} \left\{ \frac{1}{N} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \|\beta\|_1 \right\}.$$

The term $\lambda\|\beta\|_1$ is an L1 penalty and $\lambda \geq 0$ is a complexity parameter.

- The L_1 penalty performs continuous shrinkage like Ridge.
- Crucially, due to the nature of the $|\cdot|$ function, it is able to shrink some coefficients to be **exactly zero**. Thus, it performs automatic variable selection, yielding sparse models.
- There is no closed-form solution; it's solved via iterated methods such as coordinate descent.

Geometric Interpretation: Ridge vs. Lasso

The difference between Ridge and Lasso can be viewed as optimizing RSS subject to different constraints. Let's look at $p = 2$ case:

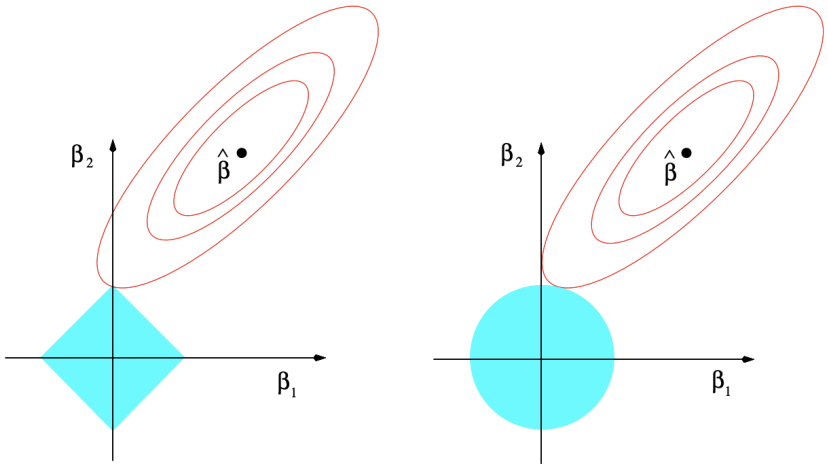


FIGURE 3.11. *Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions $|\beta_1| + |\beta_2| \leq t$ and $\beta_1^2 + \beta_2^2 \leq t^2$, respectively, while the red ellipses are the contours of the least squares error function.*

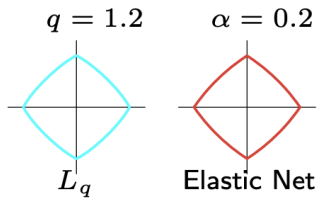


FIGURE 3.13. *Contours of constant value of $\sum_j |\beta_j|^q$ for $q = 1.2$ (left plot), and the elastic-net penalty $\sum_j (\alpha\beta_j^2 + (1-\alpha)|\beta_j|)$ for $\alpha = 0.2$ (right plot). Although visually very similar, the elastic-net has sharp (non-differentiable) corners, while the $q = 1.2$ penalty does not.*

⚠ Obtaining excess risk bounds for the Lasso estimator is much more involved than the OLS and RLS estimator (see Ch. 8 in Bach).

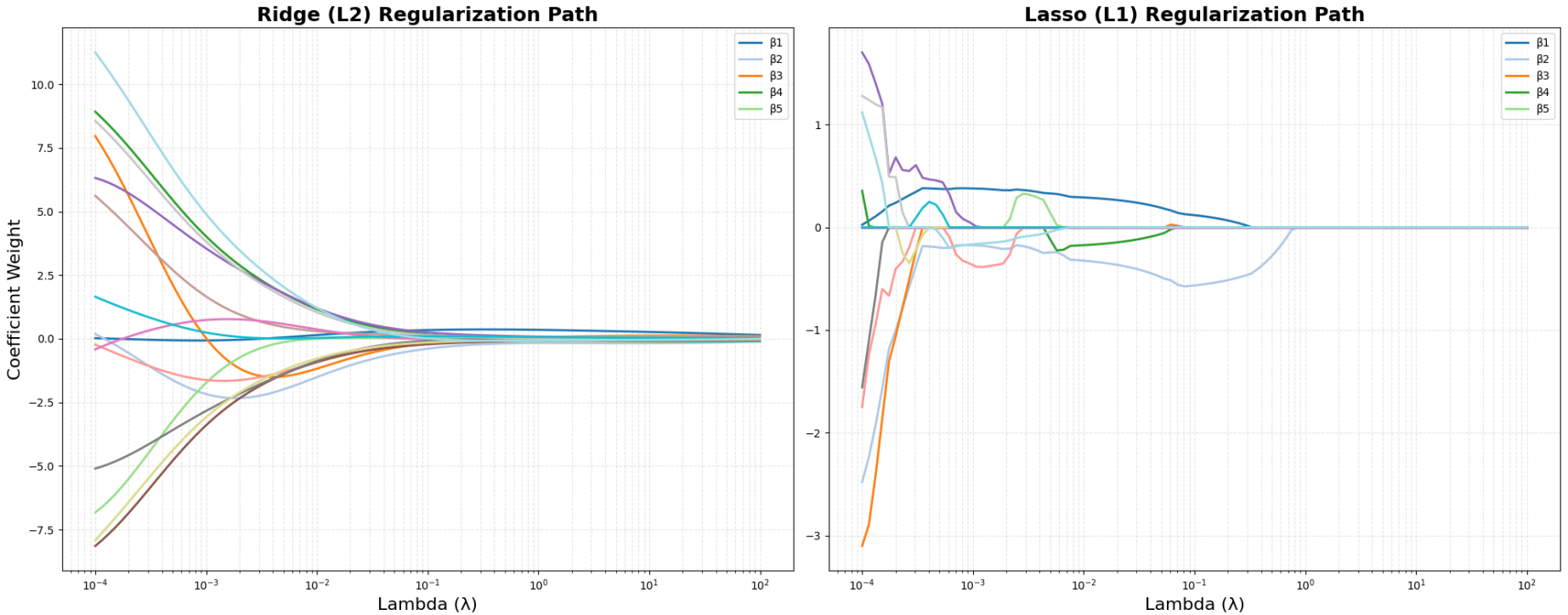
Empirical Demonstration: OLS vs. RLS vs. Lasso

- Create synthetic data from $f(x) = 1.5x^3 - 0.8x^2 + 0.3x + 0.1$ with added Gaussian noise.
- Generate 150 samples uniformly from $[-1, 1]$.
- Split into training (60%), validation (20%), and test (20%).

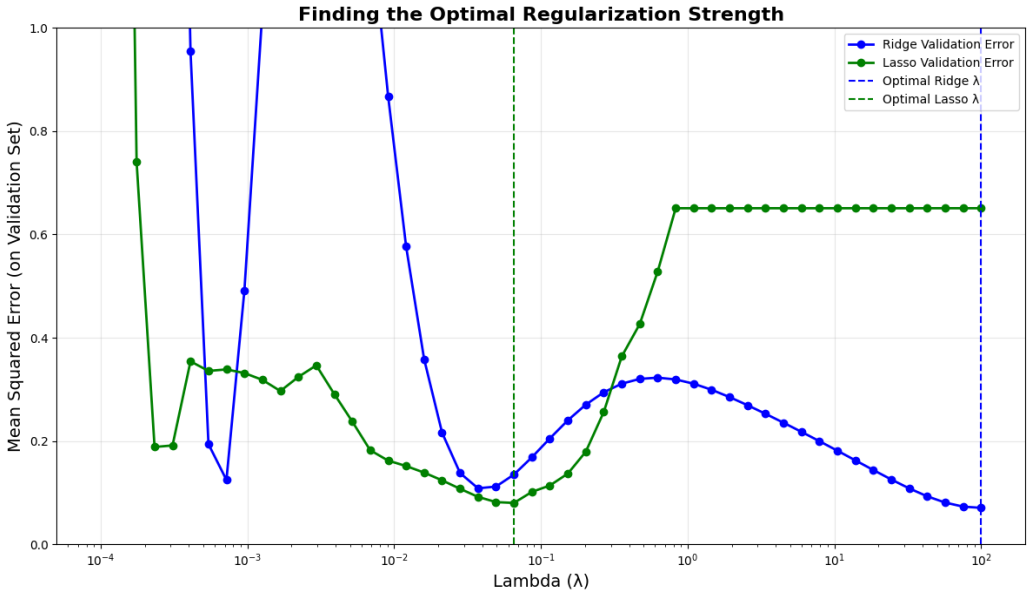
Fit Ridge (L2) and Lasso (L1) regression models with degree-15 polynomial features (so $d = 16$) on the training data. Vary λ over a grid and identify optimal λ for each method (using the validation set).

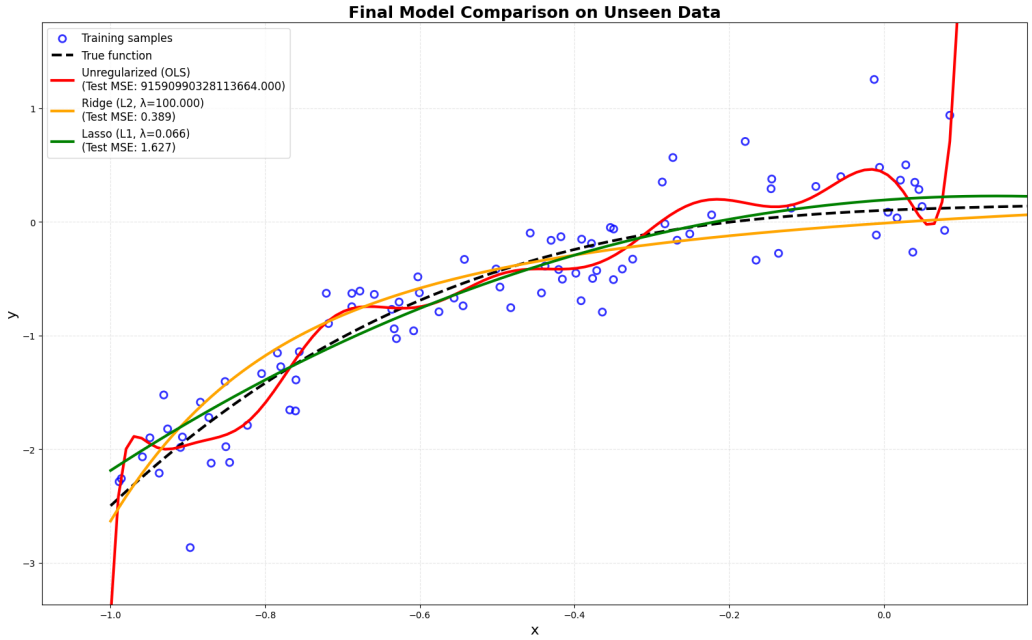
Compare test MSE of the RLS model with optimal λ vs. the Lasso model with optimal λ vs. the OLS model on the test data.

Empirical Demonstration: OLS vs. RLS vs. Lasso



Empirical Demonstration: OLS vs. RLS vs. Lasso

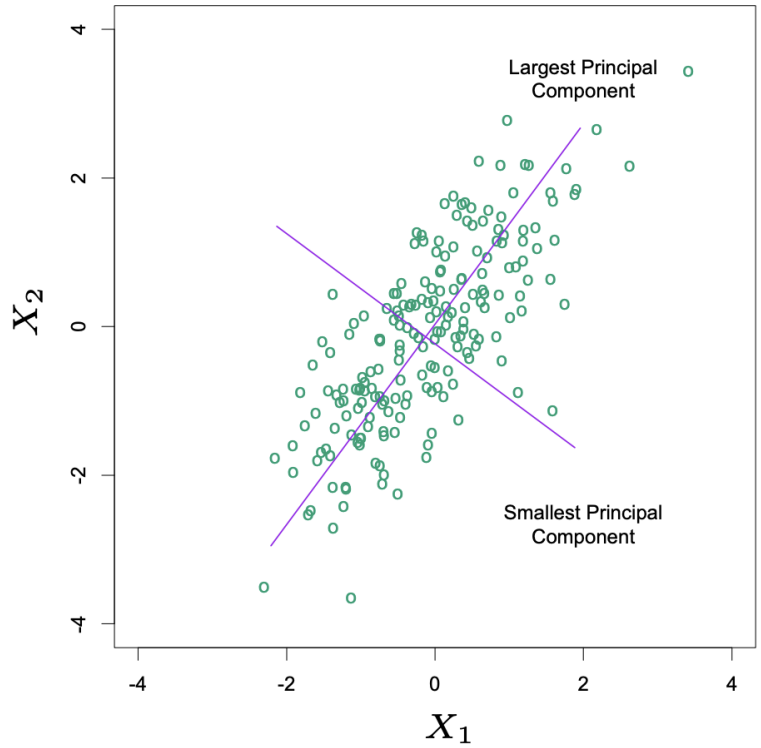




Methods Using Derived Inputs

What if we have a large number of inputs, often very correlated?

One approach is to first derive a small number of linear combinations Z_m, $m = 1, \ldots, M$, of the original inputs X_j, and then use the Z_m in place of the X_j as inputs in the regression.



Principal Components Regression (PCR)

The mth principal component of $\mathbf{X}$ is the eigenvector associated with the mth largest eigenvalue of $\mathbf{X}^T\mathbf{X}$, assuming the columns of $\mathbf{X}$ are centered.

- Compute the first $M < p$ principal components $Z_1, \ldots, Z_M$ of the input matrix $\mathbf{X}$. These are linear combinations of the original predictors that capture the most variance.
- Regress the response $\mathbf{y}$ on these M principal components using OLS.

⚠ **Drawback:** The principal components are derived in an **unsupervised** manner. The directions of high variance in X might not be the directions that best predict Y. More on this later in the course.

Partial Least Squares (PLS)

PLS is similar to PCR but addresses its main drawback by deriving the new input directions in a supervised way.

Core Idea of PLS

PLS builds a sequence of derived inputs, $Z_1, \ldots, Z_M$, by iteratively finding directions in the predictor space that are highly correlated with the response.

- At each step m, the direction is formed as a weighted average of the predictors, where the weights are the correlations of the predictors with the **current response vector**.
- After deriving a component Z_m, the predictors are made orthogonal to it. This ensures that the next component focuses on explaining the remaining variance in the data.

This supervised, iterative fitting process makes PLS a powerful tool for prediction and tends to perform well for datasets that have many correlated predictors and relatively few samples ($p \gg N$).

A High-Level Summary of Linear Methods

Method	Interpretability	Variable Selection	Common Use
OLS	High	No	Baseline, Inference ($N > d$)
Subset	Very High	Yes (Hard)	Small d, Finding "true" model
Stepwise	High	Yes (Hard)	Larger d, Efficient search
Ridge	Moderate	No	Prediction, Collinearity, $d > N$
Lasso	High	Yes (Soft)	Prediction with sparsity, $d > N$
PCR/PLS	Low	No	High-dim data, $d \gg N$

Regularization provides a powerful framework to balance the bias-variance tradeoff.

The Lasso has become quite popular because it offers a good compromise between the predictive power and stability of Ridge regression vs. the sparsity and interpretability of subset selection.

Exercises

Exercise 2

1. Prove the Gauss-Markov theorem.
2. Under the same set of assumptions used in Proposition 3, show that the expected empirical risk (or expected training error) is equal to $\mathbb{E}[\hat{R}(\hat{\beta})] = \frac{N-d}{N}\sigma^2$. In particular, when $N > d$, deduce that an unbiased estimator of the noise variance σ^2 is given by $\frac{1}{N-d}\|\mathbf{y} - \mathbf{X}\hat{\beta}\|_2^2$.
3. Show that the Ridge Regression solution $\hat{\beta}^{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$ is equivalent to solving the OLS problem subject to a spherical constraint, i.e., minimizing $\|\mathbf{y} - \mathbf{X}\beta\|_2^2$ subject to $\|\beta\|_2^2 \leq t$ for some t.
4. Show that in the case of orthonormal inputs, the ridge estimates are a scaled version of the least squares estimates: $\hat{\beta}_{ridge} = \hat{\beta}/(1+\lambda)$. The effective degrees of freedom of the ridge regression fit is defined as $\text{df}(\lambda) = \text{tr}(\mathbf{X}\mathbf{H}_\lambda)$. Compute $\text{df}(\lambda)$ and relate it to the scaling.
5. Prove Proposition 6. Validate the bias-variance decomposition empirically by providing a Python implementation to produce a figure like Figure 3.3 in Bach.