Scientific Machine Learning 08
Generalized Linear Regression, Subset Selection, and Shrinkage

Donghyun Ko

January 4, 2026

In this posting, we're going to build linear regression from first principles, prove the ordinary least squares (OLS) solution, understand when OLS is *best* (Gauss–Markov), generalize to correlated/heteroscedastic noise via generalized least squares (GLS), and then learn four subset selection strategies (Best subset, Forward/Backward Stepwise, Forward-Stagewise) and three regularizers (Ridge, Lasso, Elastic net). We will finish with the idea behind Least Angle Regression (LAR). Each section includes intuition, derivations, and training tips you can code immediately.

Contents

1 Motivation: Why start with linear models?

Linear models are simple, interpretable, and surprisingly strong when data are limited, signal-to-noise ratio is low, or features are sparse. Even many nonlinear methods extend linear models (e.g., adding basis expansions or kernels). Mastering linear regression clarifies core ideas such as design matrices, projections, loss minimization, regularization, and the bias–variance trade-off that transfer directly to generalized linear models (e.g., logistic/Poisson regression), basis-expansion models (e.g., polynomials, splines), and kernel methods (kernel ridge). These same ideas make it easier to approach Gaussian processes and deep networks later.

2 Linear Regression

Linear regression models the relationship between a scalar outcome Y and one or more explanatory variables $\{X_1, \ldots, X_d\}$. Here, Y is the *dependent variable* (or *response*), and the X_j are *independent variables* (also called *'regressors'* or *'predictors'*). The goal is to describe how the mean of Y changes with $\{X_1, \ldots, X_d\}$ via a function like $f(x) = \mathbb{E}[Y \mid X = x]$ that is typically taken to be linear (e.g., $f(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_d x_d$) so that the relationship between Y and $\{X_1, \ldots, X_d\}$ can be quantified and used to predict for new inputs.

2.1 Simple Linear Regression

The case of a single predictor is called simple linear regression. For multiple predictors, the process is called multiple linear regression. In this section, we deal with simple linear regression, with only 1 predictor X and 1 response Y. We begin with 'n' observations that come in pairs:

$$(x_1, y_1),\ (x_2, y_2),\ \ldots,\ (x_n, y_n),$$

where each x_i is the observed value of the predictor (independent variable) and y_i is the corresponding observed response (dependent variable).

Model equation. In simple linear regression, we model each response as

$$y_i = \beta_0 + \beta_1 x_i + e_i, \qquad i = 1, 2, \ldots, n,$$

where β_0 is the intercept, β_1 is the slope, and e_i is the ith error (or residual term). The error e_i captures the deviation of y_i from the regression line.

Mean and variance functions. Regression focuses on estimating the *mean function* of the response as a function of the predictor based on a given dataset s.t:

$$\mathbb{E}[Y \mid X] \ = \ \mathbb{E}[Y] \ = \ \beta_0 + \beta_1 X,$$

which says the expected value of Y changes linearly with x. In parallel, we specify a *variance function* for the spread of Y given x:

$$\mathrm{Var}(Y \mid X) \ = \ \mathrm{Var}(Y) \ = \ \sigma^2$$

Throughout, we will treat the observed X as fixed and known values rather than a random variable. Under this "fixed design" perspective, it is common to drop the conditioning bar and write $\mathbb{E}[Y]$ and $\mathrm{Var}(Y)$ in place of $\mathbb{E}[Y \mid X]$ and $\mathrm{Var}(Y \mid X)$ for notatioanl convenience; the intended meaning remains the conditional mean and variance as described above.

Estimation and fitted values. We estimate β_0 and β_1 from the data to build a model mapping between X and Y, and denote their estimates by $\widehat{\beta}_0$ and $\widehat{\beta}_1$. The fitted (predicted) value of y_i at x_i is then

$$\widehat{y}_i = \widehat{\beta}_0 + \widehat{\beta}_1 x_i$$

The residual is the difference between the observed and fitted values:

$$e_i = y_i - \widehat{y}_i$$

Gauss–Markov conditions. For ordinary least squares (OLS) to provide the *Best Linear Unbiased Estimator (BLUE)* in training, we require the errors $\{e_i\}$ to satisfy:

1. Zero mean: $\mathbb{E}[e_i] = 0$
2. Homoscedasticity: $\mathrm{Var}(e_i) = \sigma^2$ (constant across all i)
3. No correlation between errors: $\mathrm{Cov}(e_i, e_j) = 0$ for $i \neq j$

Least-squares objective and normal equations. The Ordinary Least Squares (OLS) method estimates β_0, β_1 by minimizing the *Residual Sum of Squares* (RSS):

$$\mathrm{RSS}(\beta_0, \beta_1) = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$

The most common way to find the minimizer is to take derivatives with respect to β_0 and β_1, set them to zero, and solve as follows:

1. **Differentiate the objective function w.r.t. β_0 and β_1.**

$$\frac{\partial \mathrm{RSS}(\beta_0, \beta_1)}{\partial \beta_0} = -2 \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i) = 0, \qquad \frac{\partial \mathrm{RSS}}{\partial \beta_1} = -2 \sum_{i=1}^{n} x_i (y_i - \beta_0 - \beta_1 x_i) = 0$$

2. **Rearrange to obtain the normal equations.**

$$n\beta_0 + \beta_1 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i, \qquad \beta_0 \sum_{i=1}^{n} x_i + \beta_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i$$

3. **Solve for $\widehat{\beta}_0$ and $\widehat{\beta}_1$.** From Step 2, we have the *normal equations*:

$$n\beta_0 + \beta_1 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i \tag{NE1}$$

$$\beta_0 \sum_{i=1}^{n} x_i + \beta_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i \tag{NE2}$$

(a) Express β_0 in terms of β_1. Divide (NE1) by 'n' with

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \qquad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

Then, (NE1) becomes

$$\beta_0 + \beta_1 \bar{x} = \bar{y} \quad \Longrightarrow \quad \boxed{\beta_0 = \bar{y} - \beta_1 \bar{x}}$$

(b) Plug into (NE2) **to solve for β_1.** Substitute $\beta_0 = \bar{y} - \beta_1 \bar{x}$ into (NE2):

$$(\bar{y} - \beta_1 \bar{x}) \sum_{i=1}^{n} x_i + \beta_1 \sum_{i=1}^{n} x_i^2 \;=\; \sum_{i=1}^{n} x_i y_i$$

Since $\sum_i x_i = n\bar{x}$, rearrange terms in β_1:

$$\beta_1 \left(\sum_{i=1}^{n} x_i^2 - n\bar{x}^{\,2} \right) \;=\; \sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}$$

(c) Introduce centered-sum shorthands s.t:

$$SXX := \sum_{i=1}^{n} (x_i - \bar{x})^2 = \sum_{i=1}^{n} x_i^2 - n\bar{x}^{\,2}, \qquad SXY := \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) = \sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}$$

With these, the previous line reads $\beta_1 SXX = SXY$, hence

$$\boxed{\widehat{\beta}_1 = \frac{SXY}{SXX}}$$

Takeaway.

- The fitted line $\hat{y}_i = \widehat{\beta}_0 + \widehat{\beta}_1 x_i$ is the *closest* line to the data in the least–squares sense: it minimizes the total squared gaps of $\sum_i (y_i - \hat{y}_i)^2$.
- Geometrically, think of the vector of responses $y = (y_1, \ldots, y_n)^\top \in \mathbb{R}^n$. The two vectors $\mathbf{1} = (1, \ldots, 1)^\top$ and $x = (x_1, \ldots, x_n)^\top$ span a 2-D plane in $\mathbb{R}^n$. The fitted values $\hat{y}$ are the *orthogonal projection* of y onto that plane (the "span of $\{1, x\}$"). Equivalently, the residuals $e = y - \hat{y}$ are orthogonal to both basis vectors ("This will be seen more intuitively in Multiple linear regression"):

  $$\mathbf{1}^\top e = 0 \quad \text{and} \quad x^\top e = 0,$$

  which is exactly the normal equations.
- What BLUE means. Under the Gauss–Markov conditions (errors have mean 0, constant variance σ^2, and are uncorrelated),
  - *Linear*: $\widehat{\beta}_0, \widehat{\beta}_1$ are linear functions of y.
  - *Unbiased*: $\mathbb{E}[\widehat{\beta}_j] = \beta_j$ for $j = 0, 1$.
  - *Best*: among *all* linear and unbiased estimators, OLS has the *smallest variance*. In practice, this means OLS is as statistically stable as possible within that class.
- If the Gauss–Markov conditions fail (e.g., nonconstant variance or correlated errors), OLS can lose the "Best" property; then weighted/ generalized least squares are appropriate.

2.2 Multiple Linear Regression

Multiple linear regression generalizes the simple regression model to allow the model to include more than one explanatory variable. Suppose we observe a scalar outcome Y and a set of d predictors $\{X_1, \ldots, X_d\}$. The goal is to model how the mean of Y changes as a linear function of these predictors, and to estimate the unknown regression coefficients. We begin with n observations, each consisting of a response and d predictors such that:

$$\begin{gathered}(y_1,\, x_{11},\, x_{12},\, \ldots,\, x_{1d}) \\ (y_2,\, x_{21},\, x_{22},\, \ldots,\, x_{2d}) \\ \vdots \\ (y_n,\, x_{n1},\, x_{n2},\, \ldots,\, x_{nd})\end{gathered}$$

Model equation. The multiple regression model, given this dataset, posits:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_d x_{id} + e_i, \qquad i = 1, \ldots, n$$

where β_0 is the intercept, β_j $(j = 1, \ldots, d)$ are the slopes, and e_i are the error terms. Here, $\{e_1, e_e, \ldots, e_n\}$ still satisfy the Gauss-Markov conditions. In compact matrix form, let

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad X = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1d} \\ 1 & x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nd} \end{bmatrix}, \quad \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_d \end{bmatrix}, \quad e = \begin{bmatrix} e_1 \\ \vdots \\ e_n \end{bmatrix}$$

Then, the regression model can be written succinctly as

$$y = X\beta + e$$

Mean and variance functions. In multiple linear regression, we model the *mean function* of the response as a linear combination of the d predictors. In scalar notation,

$$\mathbb{E}[Y \mid X = x] \;=\; \mathbb{E}[Y] \;=\; \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_d X_d \;=\; \beta_0 + \sum_{j=1}^{d} X_j\, \beta_j$$

Equivalently, in matrix form (with the first column of X equal to ones for the intercept),

$$\mathbb{E}[y] \;=\; X\beta$$

The *variance function* is assumed constant across observations (homoscedasticity):

$$\mathrm{Var}(Y \mid X = x) \;=\; \mathrm{Var}(Y) \;=\; \sigma^2, \qquad \Longleftrightarrow \qquad \mathrm{Var}(y) = \sigma^2 I_n.$$

Least Squares Estimation in Multiple Regression For multiple regression with data (X, y), the Ordinary Least Squares (OLS) estimator chooses coefficients β that minimize the total squared residuals. The residual vector is

$$e = y - \hat{y} = y - X\beta,$$

and the objective function is the *Residual Sum of Squares* (RSS):

$$\begin{aligned} \text{RSS}(\beta) &= e^\top e \\ &= (y - X\beta)^\top (y - X\beta) \\ &= y^\top y - 2\beta^\top X^\top y + \beta^\top X^\top X\beta, \end{aligned}$$

where $\beta^\top X^\top y = y^\top X\beta$. Recall a useful equivalent form of the RSS by letting $x_i^\top$ denote the i-th row of X (including the leading 1 for the intercept), and writing $\beta = (\beta_0, \beta_1, \ldots, \beta_d)^\top$. Then,

$$\text{RSS}(\beta) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - x_i^\top \beta)^2 = \sum_{i=1}^{n} \Big(y_i - \beta_0 - \sum_{j=1}^{d} x_{ij}\beta_j \Big)^2$$

Normal equations. To minimize $\text{RSS}(\beta)$, we differentiate with respect to β:

$$\frac{\partial \text{RSS}}{\partial \beta} = -2X^\top y + 2X^\top X\beta = -2X^\top (y - X\beta)$$

Setting this derivative equal to zero gives the *normal equations* such that:

$$X^\top X\, \beta = X^\top y.$$

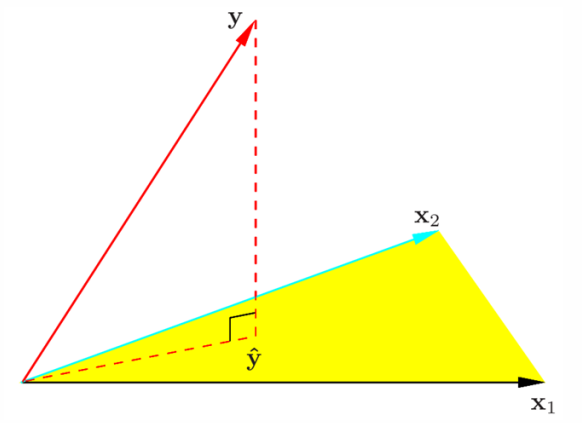
If $X^\top X$ is invertible (which requires the columns of X to be linearly independent), the unique solution is

$$\widehat{\beta} = (X^\top X)^{-1} X^\top y$$

Geometric interpretation. Now we know that the fitted values are $\widehat{y} = X\widehat{\beta}$. Geometrically, $\widehat{y}$ is the projection of the observed response vector y onto the column space of X. Equivalently, the residual vector $e = y - \widehat{y}$ is orthogonal to every column of X s.t:

$$X^\top e = 0$$

This orthogonality condition is another way of expressing the normal equations.



Properties of the OLS Estimator.

- **Unbiasedness.** Recall $\widehat{\beta} = (X^\top X)^{-1} X^\top y$ and $y = X\beta + e$ with $\mathbb{E}[e] = 0$. Then,

$$\begin{aligned}\mathbb{E}[\widehat{\beta}] &= (X^\top X)^{-1} X^\top \mathbb{E}[y] \\ &= (X^\top X)^{-1} X^\top (X\beta) \\ &= (X^\top X)^{-1} (X^\top X)\, \beta \\ &= \beta\end{aligned}$$

  so $\widehat{\beta}$ is an *unbiased* estimator of β.

- **Variance.** By using $\widehat{\beta} = (X^\top X)^{-1} X^\top y$ and $\mathrm{Var}(y) = \sigma^2 I$, and also noting that $Var(Ab) = AVar(b)A^T$:

$$\begin{aligned}\mathrm{Var}(\widehat{\beta}) &= (X^\top X)^{-1} X^\top \mathrm{Var}(y)\, X (X^\top X)^{-1} \\ &= (X^\top X)^{-1} X^\top (\sigma^2 I)\, X (X^\top X)^{-1} \\ &= \sigma^2 (X^\top X)^{-1}\end{aligned}$$

BLUE: Best Linear Unbiased Estimator. The Gauss–Markov theorem states that under the assumptions:

1. $\mathbb{E}[e] = 0$ (errors have zero mean),
2. $\mathrm{Var}(e) = \sigma^2 I$ (homoscedasticity),
3. $\mathrm{Cov}(e_i, e_j) = 0$ for $i \neq j$ (no correlation between errors),

the OLS estimator $\widehat{\beta}$ is the *Best Linear Unbiased Estimator (BLUE)* of β. Specifically:

- *Linear*: $\widehat{\beta}$ is a linear function of the observed responses y.
- *Unbiased*: $\mathbb{E}[\widehat{\beta}] = \beta$.
- *Best*: among all linear and unbiased estimators, $\widehat{\beta}$ has the smallest variance-covariance matrix.

Takeaway. Multiple linear regression generalizes the simple regression setup to a higher-dimensional predictor space. The estimation of OLS is equivalent to projecting the response vector y onto the subspace spanned by the predictors. Under the Gauss–Markov assumptions, this yields unbiased estimates of β with the lowest possible variance among linear unbiased methods.

2.3 Generalized Multiple Linear Regression

In ordinary linear regression, the Gauss–Markov assumptions play a central role. They state that the errors $e = (e_1, \ldots, e_n)^\top$ must have

$$\mathbb{E}[e] = 0, \qquad \mathrm{Var}(e) = \sigma^2 I,$$

so the errors are uncorrelated and have the same variance (homoscedasticity). Under these conditions, OLS is not only unbiased but also the *Best Linear Unbiased Estimator (BLUE)* of β: no other linear unbiased method can beat it in terms of variance. But, real-world data often break these neat assumptions. In practice:

- Measurements taken under different conditions may have very different noise levels (*heteroscedasticity*).
- Time series or spatial data often exhibit correlation across errors (*autocorrelation*).

When this happens, the OLS still produces unbiased estimates of β, but those estimates are no longer the best or efficient("They are no longer BLUE"). They may have unnecessarily large variance, and hence wider confidence intervals and weaker predictive power. This motivates the use of *Generalized Least Squares (GLS)*.

Model setup. Formally, suppose that we still have the linear model

$$y = X\beta + e, \qquad \mathbb{E}[e] = 0,$$

but now allow for a general covariance structure:

$$\mathrm{Var}(e) = \sigma^2 \Sigma,$$

where $\Sigma \in \mathbb{R}^{n \times n}$ is a *known*, symmetric positive definite (SPD) matrix that captures heteroscedasticity or correlations across observations.

Consequence for OLS. In this setting, the familiar OLS estimator

$$\widehat{\beta}_{\mathrm{OLS}} = (X^\top X)^{-1} X^\top y$$

remains unbiased. However, its variance is now

$$\mathrm{Var}(\widehat{\beta}_{\mathrm{OLS}}) = \sigma^2 (X^\top X)^{-1} X^\top \Sigma X (X^\top X)^{-1},$$

which depends on Σ in a way that OLS does not account for. This variance can be much larger than necessary.

Motivation for GLS. The key idea of GLS is to adjust the estimation procedure so that it directly accounts for Σ. By incorporating information about the noise structure, GLS produces estimates that are still unbiased but have the smallest possible variance among all linear unbiased estimators—even when the Gauss–Markov conditions are violated. In other words, GLS restores the BLUE property under more realistic error assumptions.

Generalized RSS and GLS problem. Define the *generalized residual sum of squares (GRSS)* as the quadratic form weighted by Σ^{-1} such that:

$$\begin{aligned}\mathrm{GRSS}(\beta) &= (y - X\beta)^\top \Sigma^{-1}(y - X\beta) \\ &= y^\top \Sigma^{-1} y \;-\; 2\,\beta^\top X^\top \Sigma^{-1} y \;+\; \beta^\top X^\top \Sigma^{-1} X\,\beta\end{aligned}$$

and define the *Generalized Least Squares (GLS)* estimator as

$$\widehat{\beta}_{\mathrm{GLS}} \;=\; \arg\min_{\beta}\; \mathrm{GRSS}(\beta) \;=\; \arg\min_{\beta}\; (y - X\beta)^\top \Sigma^{-1}(y - X\beta)$$

By standard matrix calculus[1], the gradient and Hessian are

$$\nabla_\beta \mathrm{GRSS}(\beta) \;=\; -2X^\top \Sigma^{-1} y \;+\; 2X^\top \Sigma^{-1} X\,\beta, \qquad \nabla^2_\beta \mathrm{GRSS}(\beta) \;=\; 2X^\top \Sigma^{-1} X$$

Setting the gradient to zero yields the *generalized normal equations*:

$$X^\top \Sigma^{-1} X\,\widehat{\beta}_{\mathrm{GLS}} \;=\; X^\top \Sigma^{-1} y$$

If X has the full column rank and $\Sigma \succ 0$ (symmetric and positive definite, hence invertible), then $X^\top \Sigma^{-1} X$ is invertible and the unique minimizer is

$$\boxed{\widehat{\beta}_{\mathrm{GLS}} = \left(X^\top \Sigma^{-1} X\right)^{-1} X^\top \Sigma^{-1} y}$$

Moreover, since $\nabla^2_\beta \mathrm{GRSS}(\beta) = 2X^\top \Sigma^{-1} X \succ 0$, this solution is the global minimum. This GLS estimator is a BLUE for this generalized multiple linear regression. The proof of this is given below.

Proof. Derivation by "whitening" (Cholesky trick)

1. Take a factor C such that $\Sigma^{-1} = C^\top C$ (e.g., Cholesky of Σ^{-1}). Here, C is a square matrix s.t $C^{-1}(C^T)^{-1} = \Sigma$. Then, multiply the model by C:

$$z \;=:\; Cy \;=\; CX\beta + Ce \;=:\; M\beta + \delta \quad (\text{Define } z = Cy, \quad M = CX, \quad \delta = Ce)$$

2. Now, this transformed noise satisfies Gauss–Markov such that:

$$\mathbb{E}[\delta] = C\mathbb{E}[e] = 0, \qquad \mathrm{Var}(\delta) = C\;\mathrm{Var}(e)\,C^\top = C(\sigma^2\Sigma)C^\top = \sigma^2 I$$

3. The BLUE is then obtained as the transformed model's OLS estimator:

$$\widehat{\beta} \;=\; (M^\top M)^{-1} M^\top z = \left(X^\top C^\top C X\right)^{-1} X^\top C^\top C y = \left(X^\top \Sigma^{-1} X\right)^{-1} X^\top \Sigma^{-1} y,$$

[1] $\frac{\partial}{\partial\beta}(\beta^\top A\beta) = (A + A^\top)\beta$, and $\frac{\partial}{\partial\beta}(b^\top\beta) = b$. Since Σ^{-1} is symmetric, $X^\top\Sigma^{-1}X$ is symmetric.

which is exactly the GLS solution. Moreover, with $\Sigma \succ 0$, let $\Sigma = LL^\top$(Cholesky Decomposition) and set $C := L^{-\top}$ so that $C^\top C = \Sigma^{-1}$. Define the whitened variables $z := Cy$ and $M := CX$. Then,

$$\begin{aligned}\|z - M\beta\|_2^2 &= (z - M\beta)^\top (z - M\beta) \\ &= [C(y - X\beta)]^\top [C(y - X\beta)] \\ &= (y - X\beta)^\top \underbrace{C^\top C}_{=\Sigma^{-1}} (y - X\beta) \\ &= (y - X\beta)^\top \Sigma^{-1} (y - X\beta) =: \mathrm{GRSS}(\beta)\end{aligned}$$

Hence, GLS is just OLS on the whitened problem of (z, M).

Geometric Interpretation of Whitening (Cholesky Trick)

The whitening view of GLS can be interpreted geometrically. Originally, the residual vector $r(\beta) = y - X\beta$ lives in an elliptical geometry induced by Σ; the ellipsoid is defined by

$$\{\, r : r^\top \Sigma^{-1} r = \text{const} \,\}.$$

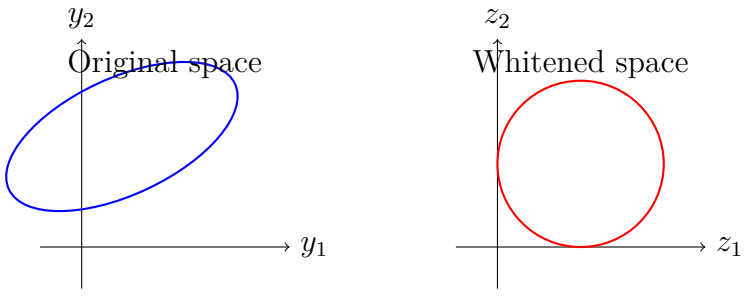
Thus, minimizing $\mathrm{GRSS}(\beta)$ corresponds to finding the point $X\beta$ such that the error $y - X\beta$ has minimal Mahalanobis length (measured in the metric Σ^{-1}), rather than minimal Euclidean length. Whitening via C transforms this elliptical geometry into a spherical one. Specifically, with $z = Cy$ and $M = CX$, the residual becomes

$$\tilde{r}(\beta) = z - M\beta = C(y - X\beta),$$

and its squared norm is simply

$$\|\tilde{r}(\beta)\|_2^2 = (y - X\beta)^\top \Sigma^{-1} (y - X\beta)$$

In this transformed space, the covariance of the noise is proportional to the identity, so the usual OLS geometry applies: the estimator is the orthogonal projection of z onto $\mathrm{col}(M)$ in the Euclidean metric. The figure below illustrates this idea. In the original space (left), the error ellipsoids are tilted and stretched according to Σ. After multiplying by $C = L^{-\top}$, the ellipsoids become circles (right), and GLS reduces to a standard OLS projection problem in the whitened coordinates.



Hence, GLS can be seen as *OLS in a whitened coordinate system*, where the whitening transformation C reshapes the ellipsoids defined by Σ into spheres, restoring the orthogonality geometry of Gauss–Markov.

Properties of the GLS estimator by Aitken's theorem

Unbiasedness. The expectation of the GLS estimator is

$$\begin{aligned}\mathbb{E}[\widehat{\beta}_{\text{GLS}}] &= \mathbb{E}\Big[\big(X^\top\Sigma^{-1}X\big)^{-1}X^\top\Sigma^{-1}y \;\Big|\; X\Big] \\ &= \big(X^\top\Sigma^{-1}X\big)^{-1}X^\top\Sigma^{-1}\,\mathbb{E}[y|X] \\ &= \big(X^\top\Sigma^{-1}X\big)^{-1}X^\top\Sigma^{-1}(X\beta) \\ &= \beta\end{aligned}$$

Thus, $\widehat{\beta}_{\text{GLS}}$ is an *unbiased estimator* of β.

Variance. The variance of the GLS estimator is

$$\begin{aligned}\text{Var}(\widehat{\beta}_{\text{GLS}}) &= \text{Var}\big((X^\top\Sigma^{-1}X)^{-1}X^\top\Sigma^{-1}y \;\big|\; X\big) \\ &= (X^\top\Sigma^{-1}X)^{-1}X^\top\Sigma^{-1}\,\text{Var}(y|X)\,\Sigma^{-1}X(X^\top\Sigma^{-1}X)^{-1} \\ &= (X^\top\Sigma^{-1}X)^{-1}X^\top\Sigma^{-1}\big[\sigma^2\Sigma\big]\Sigma^{-1}X(X^\top\Sigma^{-1}X)^{-1} \\ &= \sigma^2(X^\top\Sigma^{-1}X)^{-1}\end{aligned}$$

By Aitken's theorem, among all *linear, unbiased* estimators of β, the GLS estimator $\widehat{\beta}_{\text{GLS}}$ achieves the **minimum variance**. In other words, it is the **Best Linear Unbiased Estimator (BLUE)**. As a special case, when $\Sigma = I$, GLS reduces to the familiar OLS estimator. A good understanding of linear methods is essential for understanding nonlinear ones. In fact, many nonlinear techniques are direct generalizations of linear methods. In the next section, you will see that L_2 and L_1 regularization are essentially the same with some shrinkage methods for linear models. More specifically, the Ridge and the Lasso, respectively.

2.4 Bias–Variance Tradeoff and Shrinkage

Linear models were originally developed in the *precomputer* age of statistics, yet they continue to be highly valuable today. First, they are simple and often provide an *adequate and interpretable* description of how inputs affect outputs. Second, for prediction purposes, they can sometimes outperform more sophisticated nonlinear models, particularly in situations with *small n*, *low signal-to-noise ratio (SNR)*, or *sparse data*. Third, linear methods can be applied to *transformations of the inputs*, which considerably expands their scope as we already saw in the GLS framework. Therefore, a solid understanding of linear methods is essential for appreciating nonlinear ones. Many nonlinear techniques are, in fact, direct *generalizations* of linear models. For instance, when we later derive *Gaussian process* mean and variance estimators, the results will build directly on the theory of generalized multiple linear regression.

Bias–variance identities. Recall the bias-variance decomposition of squared error:

$$\mathbb{E}\Big[(y-\hat{f})^2\Big] = \Big(f-\mathbb{E}[\hat{f}]\Big)^2 + \mathbb{E}\Big[\Big(\hat{f}-\mathbb{E}[\hat{f}]\Big)^2\Big] + \mathrm{Var}[e] = \underbrace{\mathrm{Bias}(\hat{f})^2}_{\text{squared bias}} + \underbrace{\mathrm{Var}(\hat{f})}_{\text{estimation variance}} + \sigma^2$$

Similar equation holds for an estimator $\hat{\theta}$ in estimating θ:

$$\begin{aligned}
\mathrm{MSE}(\widehat{\theta}) &= \mathbb{E}\Big[(\widehat{\theta}-\theta)^2\Big] \\
&= \mathbb{E}\Big(\widehat{\theta}-\mathbb{E}[\widehat{\theta}]+\mathbb{E}[\widehat{\theta}]-\theta\Big)^2 \\
&= \mathbb{E}\Big[(\widehat{\theta}-\mathbb{E}[\widehat{\theta}])^2\Big] + \mathbb{E}\Big[(\mathbb{E}[\widehat{\theta}]-\theta)^2\Big] + 2\,\mathbb{E}\Big[\widehat{\theta}-\mathbb{E}[\widehat{\theta}]\Big]\cdot\mathbb{E}\Big[\mathbb{E}[\widehat{\theta}]-\theta\Big] \\
&= \underbrace{\mathrm{Var}(\widehat{\theta})}_{\text{estimation variance}} + \underbrace{\Big(\mathrm{Bias}(\widehat{\theta})\Big)^2}_{\text{squared bias}},
\end{aligned}$$

where the cross term is zero because $\mathbb{E}\Big[\widehat{\theta}-\mathbb{E}[\widehat{\theta}]\Big]=0$.

Takeaway: a **slightly biased** estimator can achieve **smaller MSE** if it yields a **larger reduction in variance**. This is the core idea behind shrinkage and subset selection.

Gauss–Markov and Aitken's theorem. The classical Gauss–Markov theorem shows that, under homoskedastic and uncorrelated errors, the OLS estimator

$$\widehat{\beta}_{\mathrm{OLS}} = (X^\top X)^{-1}X^\top y$$

has the *smallest variance among all linear unbiased estimators*—that is, OLS is BLUE (Best Linear Unbiased Estimator). Aitken (1935) extended this result: when $\mathrm{Var}(e)=\sigma^2\Sigma$ with $\Sigma \succ 0$, the GLS estimator

$$\widehat{\beta}_{\mathrm{GLS}} = (X^\top\Sigma^{-1}X)^{-1}X^\top\Sigma^{-1}y$$

is BLUE. However, note that *BLUE does not guarantee the minimum MSE overall*: unbiased estimators eliminate bias, but a *biased* estimator may still achieve *lower MSE* by trading a little bias for a larger variance reduction. This explains why biased estimates are commonly used in many methods that shrink or set some least-squares coefficients to zero deliberately introduce bias.

Expanding to Shrinkage and subset selection. Prominent examples of this tradeoff are shrinkage methods. In particular, L_2 and L_1 regularization correspond to the canonical shrinkage approaches: **Ridge** and **Lasso**, respectively. Both methods add bias but substantially reduce variance, which can be especially advantageous with small sample sizes, low SNR, or sparse data, thus improving MSE and predictive performance. The selection of variable subsets represents another, more discrete, way to navigate the same bias–variance tradeoff.

3 Subset selection: making models smaller and better interpretable

When building regression models, we often want to retain only a *subset of predictors* that both (i) **predict well** and (ii) are **interpretable**. This is because ordinary least squares (OLS/GLS) estimates, while unbiased, often suffer from *high variance*, which can harm prediction accuracy. By performing subset selection, we can sacrifice a little bias to reduce variance, thereby improving generalization and making the model easier to interpret.

Motivation

The motivation to perform subset selection can be summarized in two key points:

1. **Prediction accuracy.** Least squares estimates often have *low bias (even zero bias)* but *high variance*. Shrinking or eliminating some coefficients can reduce this variance, which often leads to improved prediction accuracy. In other words, we are willing to accept a little bias to reduce the variance.

2. **Interpretation.** With many predictors, we usually prefer a smaller subset that exhibits the strongest effects. By focusing on the "big picture," we can improve interpretability while sacrificing some of the small details.

Subset selection strategies

Subset selection means that we retain only a subset of the variables and eliminate the rest from the model. This often improves interpretability and can even improve prediction accuracy. The least squares regression is still used to fit the coefficients of the retained variables, but the challenge is to choose *which subset* to keep. There are four major strategies:

1. Best-Subset Selection.

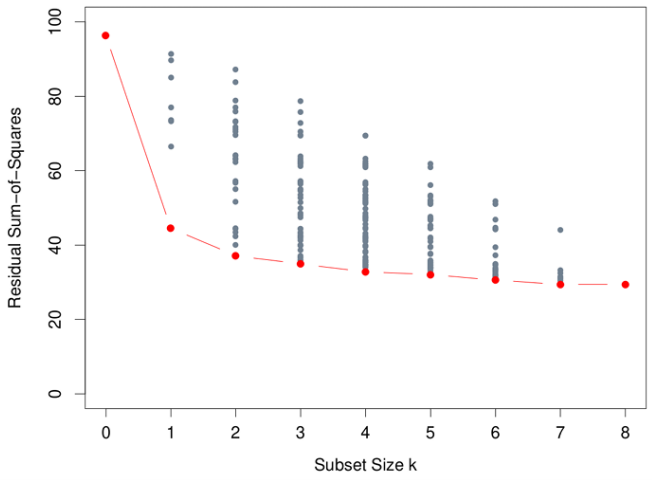
- For each possible subset size $k \in \{0, 1, \ldots, d\}$, we consider *all* possible choices of k predictors out of the total d. In other words, we must fit $\binom{d}{k}$ different regression models.

- For a fixed k, we then compare these $\binom{d}{k}$ fitted models and retain the one that achieves the smallest residual sum of squares (RSS), i.e., the model that fits the training data best among all subsets of that size.

- After this process is repeated for every subset size k, we are left with a sequence of "best" models (one for each k). We then need to decide the final subset size k^*, typically by applying a model selection criterion such as cross-validation, AIC, BIC, or by directly estimating the prediction error.

- In principle, this best-subset search is **statistically optimal**, because it explores the entire model space and guarantees the best fit for each subset size. However, it becomes **computationally infeasible** when d is large, since the number of candidate models

grows combinatorially: for example, if $d = 40$, the total number of possible subsets is $2^{40} \approx 10^{12}$. This exponential explosion makes exhaustive search impractical for even moderately large d, which is why heuristic or greedy alternatives (forward/backward stepwise, stagewise regression) are commonly used in practice.



2. Forward Stepwise Selection.

- Start with the intercept-only model ($\beta_0 = \bar{y}$). Standardize predictors for fair comparison; encode categorical variables as dummies (treat one factor's dummies as a group).
- At step t, with current selected set S_{t-1}, consider each candidate $j \notin S_{t-1}$. For each j, fit the model on $S_{t-1} \cup \{j\}$ and compute an improvement metric (e.g., decrease in RSS, increase in R^2, improvement in AIC/BIC, or lower cross-validation error).
- Choose the variable $j^\star$ that yields the largest improvement and update $S_t = S_{t-1} \cup \{j^\star\}$. After adding $j^\star$, **refit all coefficients by OLS** using the variables in S_t.
- Repeating this produces a **greedy, nested** sequence of models $S_0 \subset S_1 \subset \cdots \subset S_k$ (one best model per size), which is easy to scan/plot and compare across sizes—analogous to the size-indexed frontier from the best-subset.
- Choose the final size k^* using a model-selection criterion such as cross-validation, AIC/BIC, or hold-out validation error. In practice, you may also stop early when a max size $k_{\max}$ is reached, the improvement falls below a threshold, CV error stops decreasing, or no candidate passes a significance test.
- **Computational cost.** Forward stepwise is far cheaper than best–subset: it evaluates about $\sum_{t=1}^{k}(d - t + 1) \approx O(dk)$ models instead of 2^d. This makes it practical even with hundreds of predictors.
- **Advantages.** Forward stepwise often achieves prediction close to best–subset while producing a clear, size–indexed, interpretable path of models.
- **Disadvantages.** Forward stepwise has several important limitations:

- *Greedy search.* Because variables are added one at a time, the algorithm may fail to select groups of predictors that are only useful when included together. It focuses on short–term gains and can miss long–term combinations.
- *Collinearity.* When predictors are highly correlated, the order of selection can be unstable. Small changes in the data may lead to very different choices, making the results harder to trust.
- *Inference.* Standard p–values reported in the final model are overly optimistic, since the model was chosen after searching through many alternatives. Correct inference requires adjustments or resampling methods.
- *Sample size requirement.* The method generally works best when the number of observations n exceeds the number of predictors d. If d is large relative to n, shrinkage methods such as Ridge or Lasso are usually more reliable.

3. Backward Stepwise Selection.

- **Start from the full model.** Fit the regression with *all* d predictors (after standardizing and properly encoding categoricals).
- **One-step removal (greedy).** At each step, consider dropping each variable in turn; for every candidate removal, refit the model and compute a deterioration metric (e.g., increase in RSS, worse AIC/BIC, higher CV error). Remove the variable that harms the fit *least*.
- **Common test statistic.** A convenient proxy is the smallest standardized coefficient

$$z_j = \frac{\widehat{\beta}_j}{\widehat{\sigma}\sqrt{v_j}},$$

  where $\widehat{\beta}_j$ is the OLS coefficient, $\widehat{\sigma}$ is the residual s.e., and v_j is the jth diagonal of $(X^\top X)^{-1}$. Variables with the smallest $|z_j|$ are least supported by the data and are natural candidates for removal.
- **Refitting each step.** After removing one variable, **refit all remaining coefficients by OLS**. Repeat until a stopping rule is met.
- **Path of models & final size k^*.** This procedure yields a nested decreasing path (full $\rightarrow$ smaller). Choose the final size using cross-validation, AIC/BIC, or hold-out validation error.
- **Stopping rules (practical).** Stop when you reach a target size $k_{\min}$, when further removals degrade CV/validation error, or when all remaining variables pass a significance threshold (e.g., partial F-tests).
- **Computational notes.** Typically cheaper than best-subset and comparable to forward stepwise: about $O(dk)$ refits over the path, versus 2^d for exhaustive search.

- **Advantages.** Starts from the richest model (good when you suspect many relevant variables); produces an interpretable size-indexed path; often competitive in prediction while being far cheaper than best-subset.
- **Disadvantages.**
  - *Requires $n > d$.* The full OLS fit must be feasible and stable; otherwise backward stepwise cannot start (consider regularization instead).
  - *Greedy myopia.* May retain variables that look useful only because correlated partners remain; early removals can be hard to undo.
  - *Collinearity sensitivity.* With highly correlated predictors, the order of deletion can be unstable.
  - *Inference caveat.* As with other search procedures, naive p-values from the selected model are optimistic unless adjusted.
- **Forward vs. Backward (at a glance).** Forward adds from empty; backward prunes from full. Forward works when n may be comparable to d; backward needs $n > d$. Results can differ when predictors are correlated or when important effects appear only in groups.

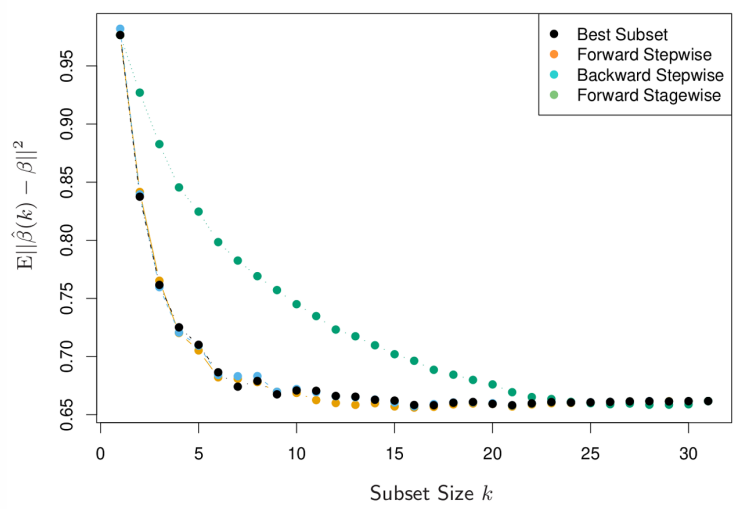
4. Forward Stagewise Regression.

- **Setup (more constrained than forward stepwise).** Center y and predictors; set the intercept to $\bar{y}$ and *initialize all coefficients at zero.* Standardizing predictors is recommended so the tiny updates are comparable across variables.
- **Greedy direction through residual correlation.** At each iteration t, compute the current residual $r^{(t)} = y - \hat{y}^{(t)}$. Find the predictor x_j with the largest absolute correlation with $r^{(t)}$:
$$j^\star \;=\; \arg\max_j \; |\langle x_j,\, r^{(t)} \rangle|$$
- **Tiny incremental update (no full refit).** Move *a very small step* $\epsilon > 0$ in the sign of that correlation:
$$\beta_{j^\star}^{(t+1)} \;=\; \beta_{j^\star}^{(t)} \;+\; \epsilon \cdot \operatorname{sign}\Big(\langle x_{j^\star}, r^{(t)} \rangle\Big),$$
and keep all other coefficients unchanged. (Contrast: forward *stepwise* fully refits all selected coefficients each time.)
- **Iterate until no signal remains.** Update the fitted values $\hat{y}^{(t+1)} = X\beta^{(t+1)}$ and repeat. Stop when no predictor has meaningful correlation with the residual (or when validation error stops improving, or a step budget is reached).
- **Path and interpretation.** The algorithm traces a *very fine, monotone, nested path* that adds signal gradually. Because it only takes tiny steps in the most correlated direction, it strongly *regularizes* the fit and can mimic ℓ_1–type shrinkage behavior.

- **Practical choices.** The step size ϵ controls smoothness and speed (small ϵ = slower but more stable). Use cross–validation to select the stopping time.
- **When it shines.** Works surprisingly well in high dimensions and with correlated predictors, producing *sparse and stable* solutions without solving a full penalized problem at each step.
- **Limitations.** Converges slowly (may require many iterations); coefficients are not refit jointly, so progress per step is small. As with other search/greedy procedures, naive inference is optimistic unless adjusted.

Summary

Subset selection reduces model complexity, improves interpretability, and can increase prediction accuracy by managing the bias–variance tradeoff. Each has its own tradeoffs between computational feasibility, bias, and variance. In practice, forward and backward stepwise methods are often used as compromises between the optimality of best-subset and the efficiency of stagewise approaches.



4 Shrinkage (regularization): ridge, lasso, elastic net

We have seen that, by retaining a subset of the predictors and discarding the rest, subset selection produces a model that is interpretable and can have a lower prediction error than the full model. However, because it is a discrete process (i.e., variables are either retained or discarded), it often exhibits high variance, and so it does not always reduce the prediction error of the full model. Shrinkage methods add a penalty that pulls coefficients toward zero, so the fitted model varies less from sample to sample. This introduces a small bias, but the accompanying drop in variance often *reduces overall MSE* and improves prediction—especially with small n, low SNR, or correlated predictors. Unlike *subset selection*, which makes a discrete keep/drop decision and can be high variance, shrinkage traces a *continuous path*

of models indexed by a tuning parameter λ (e.g., ℓ_2 for ridge and ℓ_1 for lasso) to smoothly control model complexity. i.e., Shrinkage methods provides a single tuning knob, λ. As we increase or decrease λ, the fitted model changes gradually, from simpler to more flexible. In this way, we can smoothly control the model complexity.

4.1 Ridge regression (ℓ_2)

Ridge regression shrinks the coefficients by imposing a penalty on their magnitude. To do this, the ridge coefficients minimize a penalized RSS (residual sum of squares) with a *complexity parameter* $\lambda \geq 0$ (also called weight decay or L_2 regularization in neural network) that control the amount of shrinkage:

$$\begin{aligned}
\widehat{\beta}_{\text{ridge}} &= \arg\min_{\beta} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{d} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{d} \beta_j^2 \right\} \\
&= \arg\min_{\beta} \underbrace{(y - X\beta)^\top (y - X\beta)}_{\text{RSS}} + \lambda \underbrace{\sum_{j=1}^{d} \beta_j^2}_{\text{penalty}} \\
&= \arg\min_{\beta} \underbrace{\|y - X\beta\|_2^2 + \lambda \|\beta\|_2^2}_{\text{Penalized RSS}}
\end{aligned}$$

The larger the value of λ, the greater the amount of shrinkage. An equivalent *constrained* form to write the ridge problem is:

$$\begin{aligned}
\widehat{\beta}_{\text{ridge}} &= \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{d} x_{ij}\beta_j \right)^2 \quad \text{s.t.} \quad \sum_{j=1}^{d} \beta_j^2 \leq t \\
&= \arg\min_{\beta} \|y - X\beta\|_2^2 \quad \text{s.t.} \quad \|\beta\|_2^2 \leq t,
\end{aligned}$$

where there is a one–to–one correspondence between λ and t.

Derivation (normal equations). In order to minimize the penalized RSS w.r.t coefficients, we need to expand the objective function, differentiate, and set to zero:

$$\begin{aligned}
\text{RSS}(\beta, \lambda) &= (y - X\beta)^\top (y - X\beta) + \lambda\, \beta^\top \beta \\
&= \|y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \\
&= y^\top y \;-\; 2\beta^\top X^\top y \;+\; \beta^\top X^\top X\, \beta \;+\; \lambda\, \beta^\top \beta, \\
\nabla_\beta \,\text{RSS}(\beta, \lambda) &= -2X^\top y \;+\; 2X^\top X\, \beta \;+\; 2\lambda\, \beta \;=\; 0
\end{aligned}$$

The ridge regression solutions are easily seen to be:

$$\left(X^\top X + \lambda I\right) \widehat{\beta}_{\text{ridge}} = X^\top y \quad \Longrightarrow \quad \boxed{\widehat{\beta}_{\text{ridge}} = (X^\top X + \lambda I)^{-1} X^\top y}$$

Notice that:

(i) **Linearity in y.** Because the ridge penalty is quadratic ($\beta^\top\beta$), the entire objective is a quadratic function of β. Differentiating yields linear normal equations:

$$(X^\top X + \lambda I)\,\widehat{\beta}_{\text{ridge}} = X^\top y,$$

whose solution is

$$\widehat{\beta}_{\text{ridge}} = (X^\top X + \lambda I)^{-1} X^\top y$$

The matrix $(X^\top X + \lambda I)^{-1} X^\top$ depends only on X and λ, not on y. Therefore, $\widehat{\beta}_{\text{ridge}}$ is obtained by multiplying y by a fixed matrix. Multiplication by a fixed matrix is a linear operation, so the estimator is a *linear function of* y.

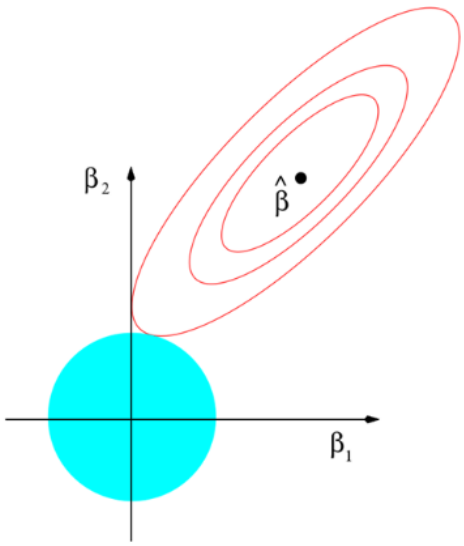
(ii) **Why adding λI stabilizes inversion.** When predictors are highly correlated, $X^\top X$ can be singular or nearly singular, making the inversion unstable. Ridge adds λI, which shifts every eigenvalue by $+\lambda$ and guarantees positive definiteness:

$$v^\top (X^\top X + \lambda I) v = \|Xv\|_2^2 + \lambda \|v\|_2^2 \;>\; 0 \quad \text{for all } v \neq 0$$

Hence, $X^\top X + \lambda I$ is always invertible for $\lambda > 0$. This adjustment prevents degeneracy and improves the condition number, leading to numerically stable estimates even under multicollinearity.

(iii) However, ridge regression cannot produce a parsimonious model as it always keeps all the predictors in the model. It does not shrink any of the regression coefficients to zero.

Geometrical analysis. In the constrained view, the feasible set is a *disk* $\{\beta : \beta_1^2 + \beta_2^2 \le t\}$. The RSS has *elliptical* contours centered at the full least–squares solution $\widehat{\beta}^{\text{LS}}$. The ridge estimator is the *first point* where an RSS ellipse touches the disk: as t shrinks (or λ grows), the point slides toward the origin. As the disk has no corners, the touch point never occurs on an axis, so the coefficients are *shrunk* toward 0 but are *not exactly* 0 (no sparsity).



SVD / shrinkage factors (Componentwise Analysis). Let $X \in \mathbb{R}^{n\times d}$ have thin SVD $X = UDV^\top$, where $U \in \mathbb{R}^{n\times r}$, $V \in \mathbb{R}^{d\times r}$ have orthonormal columns, $D = \mathrm{diag}(d_1, \ldots, d_r)$ with $d_1 \geq \cdots \geq d_r > 0$, and $r = \mathrm{rank}(X)$. The ridge estimator

$$\widehat{\beta}_{\text{ridge}} = (X^\top X + \lambda I)^{-1} X^\top y$$

can be written using the SVD as

$$\begin{aligned} X^\top X &= V D^2 V^\top, \qquad X^\top y = V D U^\top y, \\ \widehat{\beta}_{\text{ridge}} &= (V D^2 V^\top + \lambda I)^{-1} V D U^\top y \\ &= V\,(D^2 + \lambda I)^{-1} V^\top V D U^\top y \\ &= V\ \mathrm{diag}\Big(\tfrac{d_1}{d_1^2+\lambda}, \ldots, \tfrac{d_r}{d_r^2+\lambda}\Big)\, U^\top y \end{aligned}$$

The fitted values are

$$\begin{aligned} \widehat{y}_\lambda = X\widehat{\beta}_{\text{ridge}} &= U D V^\top \widehat{\beta}_{\text{ridge}} \\ &= U\ \mathrm{diag}\Big(\tfrac{d_1^2}{d_1^2+\lambda}, \ldots, \tfrac{d_r^2}{d_r^2+\lambda}\Big)\, U^\top y \end{aligned}$$

Hence, each principal–component direction u_i is *shrunk* by the factor $d_i^2/(d_i^2 + \lambda) \in (0, 1)$, with stronger shrinkage when d_i is small (ill–conditioned directions).

Orthonormal special case. If $X^\top X = I$ (orthonormal columns), then

$$\widehat{\beta}_{\text{ridge}} = (I + \lambda I)^{-1} X^\top y = \frac{1}{1+\lambda}\,\widehat{\beta}^{\text{LS}},$$

so ridge performs *proportional shrinkage*. In other words, every coefficient is scaled by the same factor of $1/(1 + \lambda)$.

4.2 Lasso regression (ℓ_1)

The *Lasso* (least absolute shrinkage and selection operator) is a shrinkage method like ridge, with subtle but important differences. It replaces the L_2 ridge penalty by the L_1 penalty:

$$\sum_{j=1}^{d} \beta_j^2 \ \longrightarrow\ \sum_{j=1}^{d} |\beta_j|$$

Formally, the lasso estimator minimizes the penalized RSS such that:

$$\begin{aligned} \widehat{\beta}_{\text{lasso}} &= \arg\min_{\beta\in\mathbb{R}^d} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{d} x_{ij}\beta_j \right)^2 \ +\ \lambda \sum_{j=1}^{d} |\beta_j| \right\} \\ &= \arg\min_{\beta\in\mathbb{R}^d}\ \|y - X\beta\|_2^2 + \lambda\|\beta\|_1, \qquad \lambda \geq 0, \end{aligned}$$

where $\lambda \geq 0$ is a complexity parameter that controls the amount of shrinkage: The larger the value of λ, the greater the amount of shrinkage. An equivalent *constrained* form to write the lasso problem is:

$$\widehat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^d} \left\{ \sum_{i=1}^{n} \Big(y_i - \beta_0 - \sum_{j=1}^{d} x_{ij}\beta_j \Big)^2 \right\} \quad \text{s.t.} \quad \sum_{j=1}^{d} |\beta_j| \leq t \qquad \text{(coordinate form)}$$

$$= \arg\min_{\beta \in \mathbb{R}^d} \left\{ \|y - X\beta\|_2^2 \right\} \quad \text{s.t.} \quad \|\beta\|_1 \leq t \qquad \text{(matrix form)}$$

with a one–to–one correspondence between λ and t (via KKT multipliers).

Derivation. Unlike ridge, the lasso objective is not differentiable at $\beta_j = 0$, so we use subgradients and KKT conditions. For this, we work with the convex objective such that:

$$f(\beta) = \underbrace{\|y - X\beta\|_2^2}_{\phi(\beta)} + \lambda \underbrace{\|\beta\|_1}_{g(\beta)}, \qquad \lambda \geq 0$$

Here, ϕ is smooth, while g is nonsmooth at coordinates equal to 0. As usual, assume y and the columns of X are centered so the intercept is unpenalized and omitted below.

1) Subgradient of ℓ_1 term. For the subgradient of $g(\beta) = \sum_j |\beta_j|$, each coordinate behaves like the absolute-value function $|\cdot|$. Away from zero, $|\cdot|$ is differentiable with slope $\text{sign}(\beta_j)$. At zero, the left-slope is -1 and the right-slope is $+1$, so every slope between them is a valid supporting slope (subgradient). Thus,

$$\partial g(\beta)_j = \begin{cases} \{\text{sign}(\beta_j)\}, & \beta_j \neq 0, \\ [-1,\, 1], & \beta_j = 0 \end{cases}$$

2) Gradient of the squared-error part. Expanding $\phi(\beta) = (y - X\beta)^\top (y - X\beta) = y^\top y - 2\beta^\top X^\top y + \beta^\top X^\top X\, \beta$ and taking derivatives gives:

$$\nabla \phi(\beta) = -2X^\top y + 2X^\top X\, \beta \;=\; -2X^\top \Big(y - X\beta \Big)$$

3) First-order optimality with a subgradient (Fermat's rule). For a convex objective $f = \phi + \lambda g$ with smooth ϕ and possibly nonsmooth g, a point $\widehat{\beta}$ is optimal iff the zero vector belongs to the subdifferential of f at $\widehat{\beta}$:

$$\mathbf{0} \in \partial f(\widehat{\beta}) \iff \mathbf{0} \in \nabla \phi(\widehat{\beta}) + \lambda\, \partial g(\widehat{\beta})$$

Here, "$\in$" means the membership of the set. Since $\partial g(\widehat{\beta})$ is a *set* of valid slopes (subgradients), $\mathbf{0} \in \nabla \phi(\widehat{\beta}) + \lambda\, \partial g(\widehat{\beta})$ is equivalent to *"there exists $z \in \partial g(\widehat{\beta})$"* such that

$$\nabla \phi(\widehat{\beta}) + \lambda z = \mathbf{0}$$

This is the subgradient form of the Fermat rule (necessary and sufficient by convexity). We can apply this to LASSO. With $\phi(\beta) = \|y - X\beta\|_2^2$, we have $\nabla\phi(\beta) = -2X^\top(y - X\beta)$. Then,

$$\mathbf{0} \in -2X^\top\Big(y - X\widehat{\beta}\Big) + \lambda\,\partial g(\widehat{\beta}) \iff \exists\, z \in \partial g(\widehat{\beta}) \text{ s.t. } -2X^\top\Big(y - X\widehat{\beta}\Big) + \lambda z = \mathbf{0}$$

$$\iff 2X^\top\Big(X\widehat{\beta} - y\Big) + \lambda z = \mathbf{0}, \quad \text{for} \quad z \in \partial g(\widehat{\beta}) = \partial\|\widehat{\beta}\|_1$$

Rearranging this gives the convenient form:

$$X^\top\Big(y - X\widehat{\beta}\Big) = \frac{\lambda}{2}\,z, \quad \text{for} \quad z \in \partial\|\widehat{\beta}\|_1$$

If we write the residual $r := y - X\widehat{\beta}$, then $x_j^\top r = (\lambda/2)\,z_j$ with $z_j \in \partial|\widehat{\beta}_j| = \{\text{sign}(\widehat{\beta}_j)\}$ if $\widehat{\beta}_j \neq 0$, and $[-1, 1]$ if $\widehat{\beta}_j = 0$, which leads directly to the familiar lasso KKT conditions.

4) Coordinate-wise KKT conditions (via the residual). Let $r := y - X\widehat{\beta}$ so that $X^\top r = (\lambda/2)\,z$. For each coordinate j,

$$x_j^\top r = \frac{\lambda}{2}\,z_j, \qquad z_j \in \partial|\widehat{\beta}_j| = \begin{cases} \{\text{sign}(\widehat{\beta}_j)\}, & \widehat{\beta}_j \neq 0, \\ [-1,\,1], & \widehat{\beta}_j = 0 \end{cases}$$

Therefore, the (necessary *and sufficient*, by convexity) KKT system is:

$$\boxed{\begin{aligned} &\widehat{\beta}_j \neq 0 \;\Rightarrow\; x_j^\top r = \tfrac{\lambda}{2}\,\text{sign}(\widehat{\beta}_j), \\ &\widehat{\beta}_j = 0 \;\Rightarrow\; |x_j^\top r| \leq \tfrac{\lambda}{2}. \end{aligned}}$$

The quantity $x_j^\top r$ is the correlation of feature j with the current residual. If it is inside the threshold band $\left[-\frac{\lambda}{2},\,\frac{\lambda}{2}\right]$, lasso sets $\widehat{\beta}_j = 0$ (not predictive enough under the ℓ_1 penalty). If it lies outside, $\widehat{\beta}_j$ remains nonzero with the matching sign and shrunken magnitude.

5) Penalized $\iff$ Constrained equivalence (Lagrangian view). Consider the constrained form

$$\min_{\beta}\; \|y - X\beta\|_2^2 \quad \text{s.t.} \quad \|\beta\|_1 \leq t$$

Its (inequality) Lagrangian with multiplier $\eta \geq 0$ is

$$\mathcal{L}(\beta, \eta) = \|y - X\beta\|_2^2 + \eta\Big(\|\beta\|_1 - t\Big), \qquad \eta \geq 0$$

The KKT conditions are:

$$\begin{aligned} \text{(Stationarity)} \quad & \mathbf{0} \in \nabla_\beta\|y - X\beta\|_2^2 + \eta\,\partial\|\beta\|_1, \\ \text{(Primal feasibility)} \quad & \|\beta\|_1 \leq t, \\ \text{(Dual feasibility)} \quad & \eta \geq 0, \\ \text{(Complementary slackness)} \quad & \eta\Big(\|\beta\|_1 - t\Big) = 0 \end{aligned}$$

Compare stationarity with the penalized problem of $\min_\beta \ \|y - X\beta\|_2^2 + \lambda\|\beta\|_1$:

$$\mathbf{0} \in \nabla_\beta \|y - X\beta\|_2^2 + \lambda\, \partial\|\beta\|_1$$

Thus, *whenever the constraint is active* ($\|\widehat{\beta}\|_1 = t$), complementary slackness implies $\eta > 0$, and stationarity matches the penalized form with the identification

$$\boxed{\lambda = \eta}$$

Active constraint ($\|\widehat{\beta}\|_1 = t$) means that the ℓ_1 budget is tight. Then, $\eta > 0$ and the constrained solution coincides with the penalized solution for $\lambda = \eta$. *Inactive constraint*($\|\widehat{\beta}\|_1 < t$) means that there is unused ℓ_1 budget. Complementary slackness forces $\eta = 0$, so the stationarity reduces to $\nabla_\beta \|y - X\beta\|_2^2 = \mathbf{0}$, i.e., ordinary least squares (OLS). In this penalized view, this corresponds to $\lambda = 0$.
Along the *active* part of the solution path, the mapping between the penalty and the budget

$$\lambda \iff t = \|\widehat{\beta}_\lambda\|_1,$$

is one-to-one and monotone: decreasing t (tighter budget) increases the associated λ (stronger penalty), and vice versa. In particular, given any $\lambda > 0$, the penalized solution $\widehat{\beta}_\lambda$ satisfies the constrained problem with $t = \|\widehat{\beta}_\lambda\|_1$; conversely, for any t at which the constraint is active, there exists a unique $\lambda = \eta$ such that the penalized and constrained optimizers coincide.

6) Useful special cases.

(a) Orthonormal columns. Suppose the design matrix has orthonormal columns, i.e. $X^\top X = I$. In this case, the objective fully decouples across coordinates:

$$\|y - X\beta\|_2^2 + \lambda\|\beta\|_1 = \sum_{j=1}^{d} \Big((y^\top x_j - \beta_j)^2 + \lambda|\beta_j| \Big),$$

so each coefficient β_j can be solved independently. The solution is the well-known *soft-thresholding rule*:

$$\widehat{\beta}_j^{\text{lasso}} = \text{sign}\Big(\widehat{\beta}_j^{\text{LS}}\Big) \ \Big(|\widehat{\beta}_j^{\text{LS}}| - \lambda \Big)_+,$$

where $(a)_+ = \max(a, 0)$. Thus if the least-squares coefficient is small in magnitude ($|\widehat{\beta}_j^{LS}| \le \lambda$), it is shrunk exactly to zero; otherwise it is reduced in magnitude by λ. This makes explicit why lasso performs both *shrinkage* and *variable selection*.

(b) Coordinate descent (general X). When $X^\top X \neq I$, coordinates interact, so we cannot solve all at once. However, holding all but one coordinate fixed, the one-dimensional subproblem is still a quadratic plus an ℓ_1 penalty. Define the *partial residual*

$$r^{(j)} = y - \sum_{k \neq j} x_k \beta_k,$$

i.e. the current residual after removing the contribution of all predictors except x_j. Then the coordinate-wise minimization is

$$\min_{\beta_j} \ \|r^{(j)} - x_j\beta_j\|_2^2 + \lambda|\beta_j|$$

This has the closed-form update

$$\beta_j \leftarrow \frac{1}{x_j^\top x_j} S_{\lambda/2}\Big(x_j^\top r^{(j)}\Big), \qquad S_\tau(a) = \operatorname{sign}(a)\,(|a| - \tau)_+,$$

the least-squares update $x_j^\top r^{(j)}/(x_j^\top x_j)$ is *soft-thresholded* by $\tau = \lambda/2$ under this scaling.

Each coordinate update checks how strongly x_j correlates with the current residual. If that correlation is weaker than the threshold, the update becomes exactly zero; if it is stronger, the coefficient is reduced in magnitude by the threshold amount. Iterating these updates cyclically over all j converges to the lasso solution.

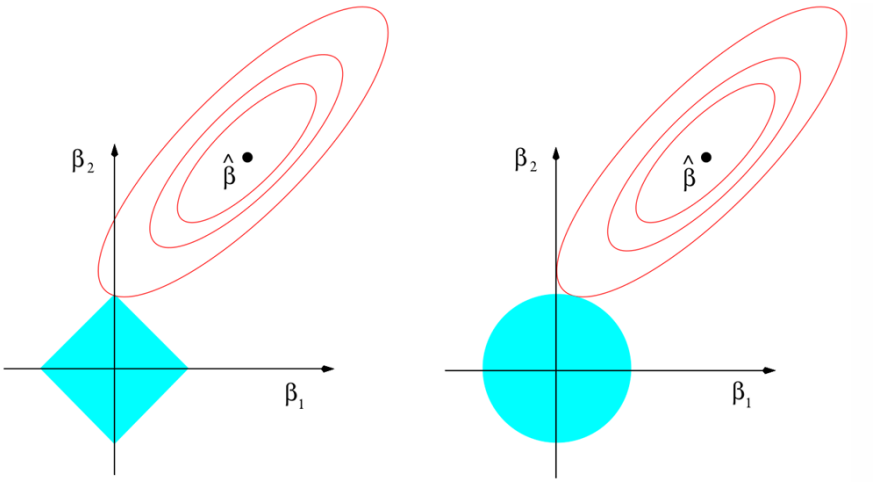
Geometrical analysis. The geometry of lasso is best understood by contrasting it with ridge. Recall that in the constrained view, ridge uses an ℓ_2 ball

$$\{\beta : \|\beta\|_2^2 \le t\},$$

which is a circle in 2D or sphere in higher-dimensions with smooth, round boundaries. By contrast, lasso uses an ℓ_1 ball

$$\{\beta : \|\beta\|_1 \le t\},$$

which is a diamond in 2D, a polytope in higher dimensions, with *sharp corners aligned with coordinate axes*. The residual sum of squares $\|y - X\beta\|_2^2$ has elliptical level sets centered at the least-squares solution $\widehat{\beta}^{\text{LS}}$. The constrained estimator is the point where an ellipse first touches the feasible set (the ℓ_1 ball).



Lasso (left) vs. Ridge (right)

Because the ℓ_1 ball has corners exactly on the coordinate axes, the ellipse often touches the feasible set at a corner. At such a corner, some coordinates are exactly zero. This geometric "corner-hitting" is the analytic reason why lasso can set coefficients to zero, producing a sparse solution. In contrast, ridge's ℓ_2 ball has no corners, so its solution always lies on a smooth boundary, shrinking coefficients toward the origin but never exactly to zero.

Higher-dimensional intuition. In d dimensions, the ℓ_1 ball is a cross-polytope with $2d$ sharp corners. The lasso solution path as λ varies traces along faces and edges of this polytope.

Whenever the ellipse (RSS contour) meets a face not containing a certain coordinate, that coordinate is forced to zero. As λ decreases, the feasible set expands, more ellipses can fit inside, and additional variables can enter the model.

Reference: Subset Selection, Ridge, and Lasso

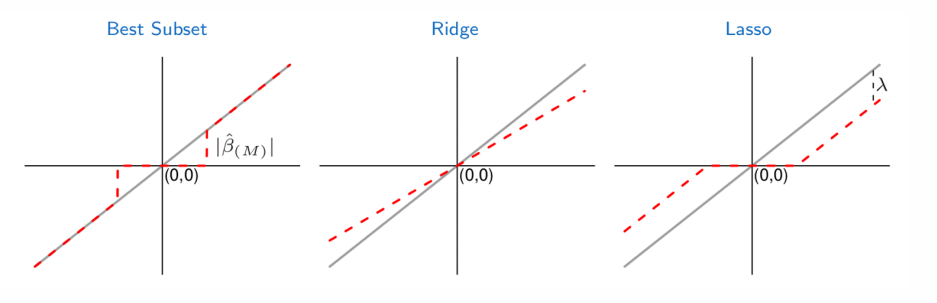
- **Best-subset selection: hard-thresholding**. Keep M largest $|\widehat{\beta}_j^{\mathrm{LS}}|$ and set the rest to zero (exact sparsity, potentially unstable). *Figure (left):* red dashed mapping is a step function: if $|\widehat{\beta}_j^{\mathrm{LS}}| < |\widehat{\beta}_{(M)}^{\mathrm{LS}}|$ then 0, else unchanged (on gray line)
- **Ridge regression: proportional shrinkage**. Coefficients move toward zero but never become exactly zero, so the model is dense. *Figure (middle):* gray line is $y = x$ (OLS); red dashed line has slope $1/(1+\lambda)$, i.e.

$$\widehat{\beta}_j^{\mathrm{ridge}} = \frac{1}{1+\lambda}\,\widehat{\beta}_j^{\mathrm{LS}}$$

- **Lasso: soft-thresholding**. Each coefficient is reduced by λ and truncated at zero: small $\to 0$, large $\downarrow$ in magnitude (sparsity + shrinkage). *Figure (right):* red dashed mapping is piecewise linear with a kink at 0 and a flat "dead zone"; it is

$$\widehat{\beta}_j^{\mathrm{lasso}} = \mathrm{sign}\Big(\widehat{\beta}_j^{\mathrm{LS}}\Big)\,\Big(|\widehat{\beta}_j^{\mathrm{LS}}| - \lambda\Big)_+,$$

so $|\widehat{\beta}_j^{\mathrm{LS}}| \le \lambda \Rightarrow 0$, otherwise shrink by λ.



When X has *orthonormal* columns ($X^\top X = I$), each method becomes a simple transform of the least-squares coefficient $\widehat{\beta}_j^{\mathrm{LS}}$:

Estimator	**Closed form (per coordinate j)**
Best subset (size M)	$\widehat{\beta}_j^{\mathrm{LS}} \cdot \mathbf{1}\Big(|\widehat{\beta}_j^{\mathrm{LS}}| \ge |\widehat{\beta}_{(M)}^{\mathrm{LS}}|\Big)$
Ridge	$\dfrac{1}{1+\lambda}\,\widehat{\beta}_j^{\mathrm{LS}}$
Lasso	$\mathrm{sign}\Big(\widehat{\beta}_j^{\mathrm{LS}}\Big)\,\Big(|\widehat{\beta}_j^{\mathrm{LS}}| - \lambda\Big)_+$

Notes. $\mathbf{1}(\cdot)$ is the indicator function; $(a)_+ = \max(a, 0)$ is the positive part; $|\widehat{\beta}_{(M)}^{\mathrm{LS}}|$ is the

M-th largest absolute LS coefficient. Under orthonormality, ridge performs *proportional shrinkage*, lasso performs *soft-thresholding* (exact zeros when $|\widehat{\beta}_j^{\mathrm{LS}}| \le \lambda$), and best subset performs *hard-thresholding* by keeping only the M largest in magnitude.

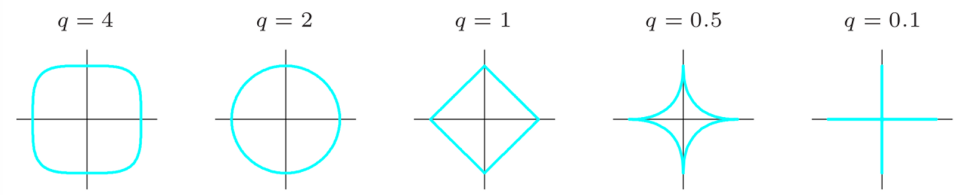
4.3 Bridge regression

$$\widehat{\beta}(q) = \arg\min_{\beta \in \mathbb{R}^d} \underbrace{\|y - X\beta\|_2^2}_{\text{RSS}} + \lambda \underbrace{\sum_{j=1}^{d} |\beta_j|^q}_{\ell_q \text{ penalty}}, \qquad q \ge 0,\ \lambda \ge 0$$

Relation to ridge and lasso (the "bridge").

- $q = 2 \Rightarrow$ **ridge** (quadratic, smooth, no sparsity; proportional shrinkage).
- $q = 1 \Rightarrow$ **lasso** (nonsmooth at 0; soft-thresholding and exact zeros).
- $0 < q < 1 \Rightarrow$ **nonconvex** penalty (stronger pull to 0; encourages even sparser models but introduces local minima).
- $q > 1$ (including $1 < q < 2$) $\Rightarrow$ **convex and smooth** at 0 (no subgradient band at 0; tends to keep coefficients nonzero while shrinking them).

Geometry (constrained view). Bridge can be written in constrained form such that $\min \|y - X\beta\|_2^2$ s.t. $\sum_j |\beta_j|^q \le t$. The feasible set is the ℓ_q ball in $\mathbb{R}^d$: in 2D, it morphs from a *diamond* ($q = 1$) with sharp axis-aligned corners (promoting sparsity) to a *circle* ($q = 2$) with smooth boundary (promoting smooth shrinkage). As $q \downarrow 1$, level sets develop sharper corners and are more likely to meet an RSS ellipse at a corner $\Rightarrow$ some coefficients become exactly 0; as $q \uparrow 2$, the boundary becomes rounder $\Rightarrow$ coefficients shrink but rarely hit 0.



Constraint shapes of the ℓ_q ball in 2D. Sharper corners as $q \downarrow 1$ promote sparsity (lasso-like); rounder shapes as $q \uparrow 2$ promote smooth, non-sparse shrinkage (ridge-like).

Algorithms (how to compute).

- **Convex regime ($q \ge 1$):** Coordinate descent is effective. For $q = 2$ and $q = 1$, updates are closed-form. For $1 < q < 2$, each 1D update solves

  $$\min_{\beta_j} \ \|r^{(j)} - x_j\beta_j\|_2^2 + \lambda|\beta_j|^q$$

  via a small Newton step or bisection (the proximal of $|\cdot|^q$ has no elementary closed form for general q).

- **Nonconvex regime ($0 < q < 1$):** Use MM/IRLS or *iteratively reweighted* schemes. A common surrogate at iterate $\beta^{(t)}$ is a weighted ridge or lasso:

$$\sum_j |\beta_j|^q \;\approx\; \sum_j w_j^{(t)}\,\beta_j^2 \quad \text{with} \quad w_j^{(t)} \propto \left(|\beta_j^{(t)}| + \varepsilon\right)^{q-2},$$

  or a local linear approximation (reweighted ℓ_1). These converge to a stationary point (not guaranteed global optimum).

Modeling guidance (choosing q, λ).

- q controls the *shape* of shrinkage: closer to 1 $\Rightarrow$ more sparsity; closer to 2 $\Rightarrow$ smoother shrinkage and stability.
- Tune q and λ by cross-validation or information criteria. In high correlation settings, $q \approx 1$ often improves interpretability; when prediction stability is critical, $q \approx 2$ is safer.
- Standardize predictors and keep the intercept unpenalized (as with ridge/lasso) to make q, λ comparable across features.

4.4 Elastic Net (ridge + lasso)

The lasso is powerful but has two weaknesses: (i) with $d \gg n$, it can select at most n variables out of d candidates; (ii) with strongly correlated predictors (e.g., grouped variables situation), it tends to select only a part of them and ignore the rest, thus it lacks the ability to reveal the grouping information. Ridge regression, in contrast, keeps correlated variables together but never sets coefficients exactly to zero. The elastic net (EN) combines both: it performs *sparsity* through the ℓ_1 part and a *grouping effect* through the ℓ_2 part. Thus, EN yields stable models that can both select variables and retain correlated groups—often outperforming lasso in practice.

$$\widehat{\beta}_{\text{EN}} = \arg\min_{\beta}\; \|y - X\beta\|_2^2 + \lambda\Big(\alpha\|\beta\|_1 + (1-\alpha)\|\beta\|_2^2\Big), \quad \alpha \in [0,1]$$

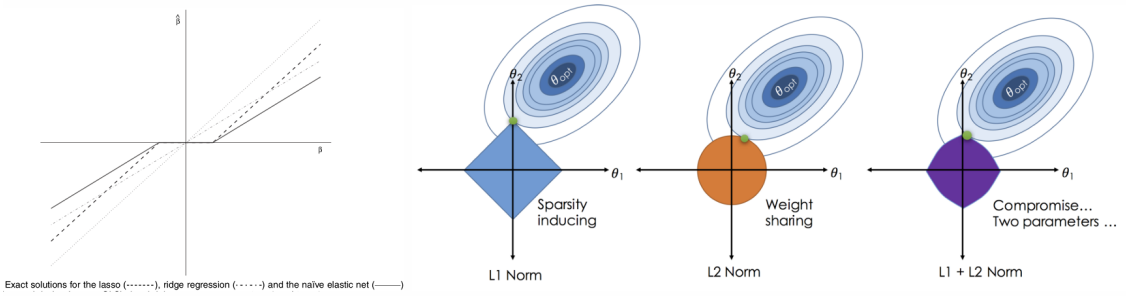
Penalty structure.

- ℓ_1 term ($\|\beta\|_1$): generates sparsity, enabling automatic variable selection.
- ℓ_2 term ($\|\beta\|_2^2$): encourages continuous shrinkage and groups correlated predictors in or out together.
- α interpolates between the extremes:
  - $\alpha = 1$: lasso (pure ℓ_1).
  - $\alpha = 0$: ridge (pure ℓ_2).
  - $0 < \alpha < 1$: elastic net compromise.

Geometry. In the constrained view of $\min \|y - X\beta\|_2^2$ s.t. $\alpha\|\beta\|_1 + (1-\alpha)\|\beta\|_2^2 \leq t$, the feasible region blends the sharp corners of the ℓ_1 ball (sparsity-inducing) with the roundness of the ℓ_2 ball (grouping effect). Singularities at the corners make exact zeros possible, while convex curvature keeps correlated predictors together. This is why EN is sometimes described as "a stretchable fishing net that keeps all the big fish together."

Tuning and practice.

- **Penalty strength λ:** chosen by cross-validation.
- **Mixing α:** balances lasso vs ridge; typically tuned on a grid along with λ.
- **Scaling:** standardize predictors before fitting to ensure penalties are comparable.
- **Interpretation:** lasso/EN perform variable selection, ridge does not. EN can include whole correlated groups while doing variable selection like Group LASSO.
- **Connections:** ridge corresponds to L_2 regularization (weight decay) in machine learning; EN is widely used for high-dimensional genomics, text, and finance data where correlation structures are strong.



Constraint sets in 2D: ℓ_1 (diamond), ℓ_2 (circle), $\ell_1 + \ell_2$ (elastic net, compromise).

5 Least Angle Regression (LAR)

Least Angle Regression (LAR) can be viewed as a more "democratic" version of forward stepwise regression. We have learned that Forward Stepwise Regression builds a model sequentially, adding one variable at a time. At each step, it identifies the best variable to include in the active set, and then updates the least squares fit to include all the active variables. Like forward stepwise, it builds the model sequentially, but instead of fully fitting one variable at a time, it moves active coefficients *gradually* and *equitably*.

Algorithmic idea.

- Start by identifying the predictor most correlated with the current residual.
- Rather than fitting this predictor to its full least-squares coefficient immediately, LAR moves its coefficient *continuously toward its OLS value*, causing its correlation with the evolving residual to decrease.

- As soon as another variable's correlation with the residual *catches up*, the process is paused.
- That second variable then enters the active set, and from then on, the two coefficients are moved together along an equiangular direction that keeps their correlations equal and decreasing.
- The procedure repeats: whenever a new variable's correlation catches up, it joins the active set, and all active coefficients are moved in lockstep.
- The process ends when all predictors have entered, yielding the full OLS solution.

Key features.

- LAR only enters "as much" of each predictor as justified by its correlation with the residual.
- It is *extremely efficient*: the computation requires about the same order of work as fitting a single OLS model with all d predictors.
- LAR provides the foundation for the *lasso path algorithm*, since the piecewise-linear path of coefficients under lasso can be obtained by a modification of the LAR steps.

This is a posting that I summarized with study-purpose and is adapted from lecture notes of NE-795(Scientific Machine Learning), given by professor Xu Wu, NC State University.