
THE UNIVERSITY *of* EDINBURGH

# Lecture 1: Bayesian Regularization

Dr Sara Wade
sara.wade@ed.ac.uk

20-23 May, 2024

# Overview of Content

1. Bayesian Regularization
2. Variational Inference
3. Bayesian Neural Networks
4. ~~Gaussian Processes~~

# Outline

# Outline

# Supervised Learning

In **supervised learning**, our data consists of labelled input-output pairs, with inputs $\mathbf{x}_n \in \mathbb{R}^D$ and outputs $y_n \in \mathbb{R}$, for $n = 1, \ldots, N$:

$$\mathbf{X} := \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_N^T \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{1,2} & \ldots & x_{1,D} \\ x_{2,1} & x_{2,2} & \ldots & x_{2,D} \\ & & \vdots & \\ x_{N,1} & x_{N,2} & \ldots & x_{N,D} \end{bmatrix} \quad \mathbf{y} := \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$$

Aim: use the training data to learn the mapping from $\boldsymbol{x}$ to $y$ for generalization, i.e. to automatically label future inputs.

**Regression**: numerical output $y_n \in \mathbb{R}$ (or more generally, $\mathbf{y}_n \in \mathbb{R}^{D_y}$ in multivariate regression).

# Linear Regression

Much of machine learning is about fitting functions to data.

In regression tasks, **linear regression** is one of the most widely used models and assumes a simple linear form of the function:

$$y = \boldsymbol{w}^T \boldsymbol{x} + \epsilon = \sum_{d=1}^{D} w_d x_d + \epsilon, \tag{1}$$

where:

- $\boldsymbol{w}$ are the regression weights or weight vector[1],
- $\epsilon$ is the residual error, with zero mean and independent of $\boldsymbol{x}$.

[1] in statistics, $\boldsymbol{w}$ are called the coefficients, often denoted $\boldsymbol{\beta}$

# Linear Regression

The inputs are often augmented with a constant term of 1, such that $\boldsymbol{x} = (1, x_1, \ldots, x_D)^T$ and the linear mean function is

$$\mathrm{E}[y|\mathbf{x}, \boldsymbol{w}] = \boldsymbol{w}^T\mathbf{x} = w_0 + \sum_{d=1}^{D} w_d x_d,$$

where $w_0$ represents the intercept or bias term.

# Linear Regression

The inputs are often augmented with a constant term of 1, such that $\boldsymbol{x} = (1, x_1, \ldots, x_D)^T$ and the linear mean function is

$$\mathrm{E}[y|\mathbf{x}, \boldsymbol{w}] = \boldsymbol{w}^T\mathbf{x} = w_0 + \sum_{d=1}^{D} w_d x_d,$$

where $w_0$ represents the intercept or bias term.

Typically, we assume $\epsilon_n \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$, resulting in the model:

$$\mathbf{y}|\mathbf{X}, \boldsymbol{w}, \sigma^2 \sim \mathrm{N}_N(\mathbf{X}\boldsymbol{w}, \sigma^2\mathbf{I}_N).$$

# High-dimensional Regression

Problems arise in the linear regression model:

$$y = \boldsymbol{w}^T \boldsymbol{x} + \epsilon,$$

when the $D$ **is large relative to** $N$, including:

- high-dimensional inputs, e.g GWAS studies, large-scale data initiatives (UK Biobank, ADNI, etc), personal tracking devices, and more.
- basis function expansion for nonlinearity.

If $N$ is not much larger than $D$, the least squares fit can be highly variable $\rightarrow$ overfitting, high variance, and poor prediction.

If $D > N$, there is no unique least squares estimate of $\boldsymbol{w}$; each solution gives zero error on training data but poor prediction.

**Regularization** overcomes this by **shrinking** $\widehat{\boldsymbol{w}} \rightarrow$ improved prediction and model interpretability.

# Outline

# Regularization

**Regularization** is achieved by including a penalty term and selecting $\widehat{\boldsymbol{w}}$ to minimize the **penalized RSS**:

$$\sum_{n=1}^{N}(y_n - \mathbf{w}^T\mathbf{x}_n)^2 + \sum_{d=1}^{D} p_\lambda(w_d).$$

The **penalty function** $p_\lambda(w)$ penalizes for large weight values, with **tuning parameter** $\lambda$.

Note: shrinkage is not applied to the intercept $\widehat{w}_0$, which simply measures the mean value of the output when all inputs are zero.

# Penalty Functions

$$\widehat{\boldsymbol{w}} = \underset{\boldsymbol{w}\in\mathbb{R}^D}{\operatorname{argmin}} \sum_{n=1}^{N} (y_n - \boldsymbol{w}^T \boldsymbol{x}_n)^2 + \sum_{d=1}^{D} p_\lambda(w_d).$$

Depending on the choice of $p_\lambda(w)$, different regularization methods are obtained. The two most popular are **ridge** and **lasso**.

Lasso enables variable selection (threshold some coefficients to zero), and other widely-used choices that allow variable selection include:

- elastic net (hybrid between lasso and ridge)
- variants of lasso (e.g. adaptive and group lasso)
- smooth clipped absolute deviation (SCAD)
- minimax concave penalty (MCP)

The latter two (SCAD and MCP) also mitigate the well-known estimation bias of lasso.

# Ridge Regression

**Ridge regression** selects $\widehat{\boldsymbol{w}}$ to minimize:

$$\sum_{n=1}^{N}(y_n - \boldsymbol{w}^T\mathbf{x}_n)^2 + \lambda\sum_{d=1}^{D} w_d^2 = (\mathbf{y} - \mathbf{X}\boldsymbol{w})^T(\mathbf{y} - \mathbf{X}\boldsymbol{w}) + \lambda\boldsymbol{w}^T\boldsymbol{w}.$$

The tuning parameter $\lambda \geq 0$ controls the trade-off between the fit and shrinkage.

# Ridge Regression

**Ridge regression** selects $\widehat{\boldsymbol{w}}$ to minimize:

$$\sum_{n=1}^{N}(y_n - \boldsymbol{w}^T\mathbf{x}_n)^2 + \lambda\sum_{d=1}^{D} w_d^2 = (\mathbf{y} - \mathbf{X}\boldsymbol{w})^T(\mathbf{y} - \mathbf{X}\boldsymbol{w}) + \lambda\boldsymbol{w}^T\boldsymbol{w}.$$

The tuning parameter $\lambda \geq 0$ controls the trade-off between the fit and shrinkage.

- for $\lambda = 0$, $\widehat{\boldsymbol{w}}$ is the least squares estimate,
- as $\lambda \to \infty$, $\widehat{\boldsymbol{w}}$ approaches zero.

# Ridge Regression

The **ridge regression estimate**[2] is:

$$\widehat{\boldsymbol{w}}_{\text{ridge}} = (\lambda \mathbf{I_D} + \mathbf{X^T X})^{-1} \mathbf{X^T y}.$$

🤔 Show this by either 1) differentiating the loss directly or 2) use data augmentation to convert the problem to least squares.

[2] for ease of the notation, the intercept is omitted

# Ridge Regression

The **ridge regression estimate**[2] is:

$$\widehat{\boldsymbol{w}}_{\text{ridge}} = (\lambda \mathbf{I_D} + \mathbf{X^T X})^{-1} \mathbf{X^T y}.$$

🤔 Show this by either 1) differentiating the loss directly or 2) use data augmentation to convert the problem to least squares.

Note: while the least squares estimates are scale equivariant, the ridge estimates can change substantially when rescaling the inputs.

To mitigate this, the inputs are typically standardized before training regularized methods.

[2] for ease of the notation, the intercept is omitted

# Ridge Regression

The ridge solution can be written as:

$$\begin{aligned}\widehat{\boldsymbol{w}}_{\text{ridge}} &= (\lambda \mathbf{I_D} + \mathbf{X^T X})^{-1} \mathbf{X^T y} \\ &= (\lambda \mathbf{I_D} + \mathbf{X^T X})^{-1} \mathbf{X^T X} (\mathbf{X^T X})^{-1} \mathbf{X^T y} = (\lambda \mathbf{I_D} + \mathbf{X^T X})^{-1} \mathbf{X^T X} \widehat{\boldsymbol{w}}_{\text{LS}}.\end{aligned}$$

🤔 Suppose that $x_{n,d} \overset{\text{iid}}{\sim} \text{N}(0, s_d^2)$ and

$$\frac{1}{N} \mathbf{X}^T \mathbf{X} \approx \begin{bmatrix} s_1^2 & 0 & \dots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & & 0 \\ 0 & \dots & 0 & s_d^2 \end{bmatrix}.$$

Show that

$$\widehat{w}_{\text{ridge } d} = \frac{s_d^2}{s_d^2 + \lambda/N} \widehat{w}_{\text{LS } d}$$

How does the scale, $\lambda$, and $N$ effect the amount of shrinkage?

# Ridge Regression: Simulated Example

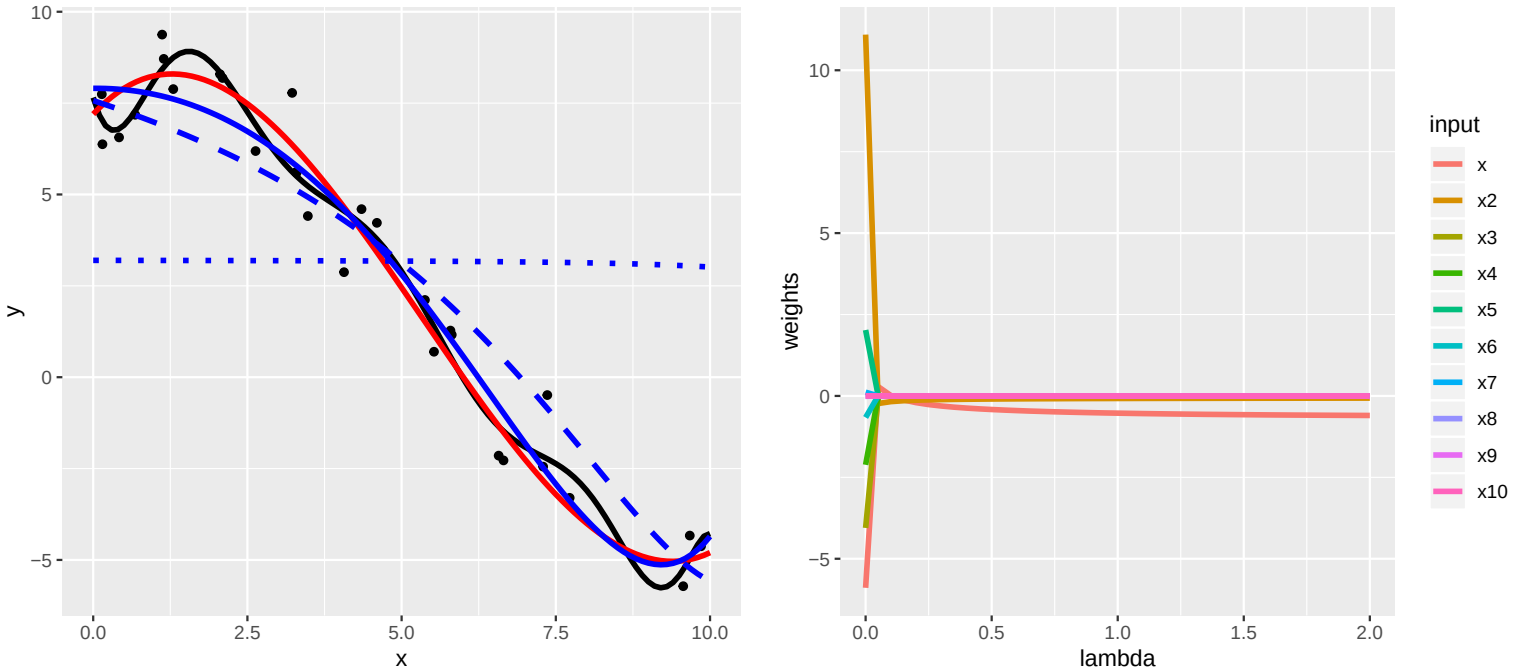



Figure: Ridge regression: Left: true regression function (red), least squares regression function (black), and ridge regression function (blue) for $\lambda = 0.1$ (solid), $\lambda = 10$ (dashed), and $\lambda = \exp(10)$ (dotted). Right: estimated ridge weights as a function of $\lambda$.

# How to select $\lambda$?

**Selecting** a good value of $\lambda$ is critical, and **cross-validation** provides a simple way to tackle this.

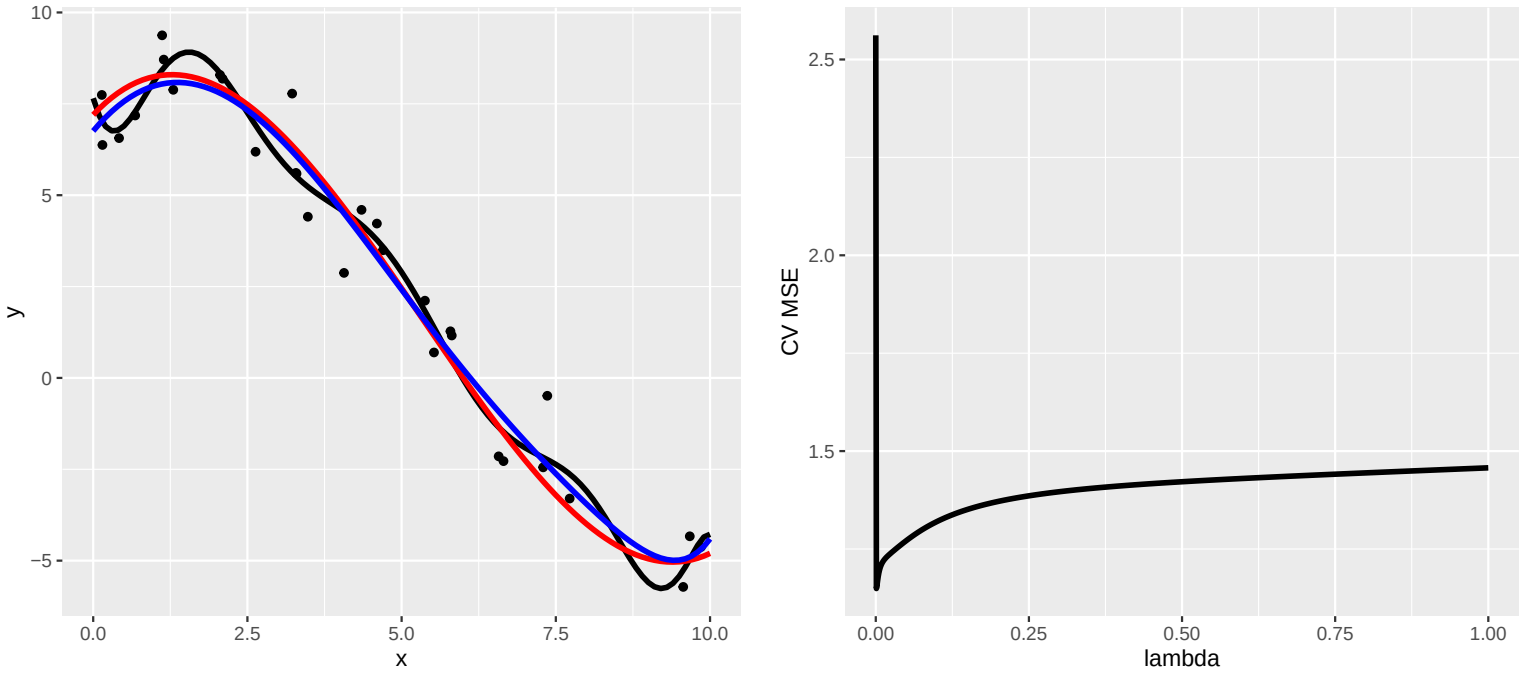



Figure: Ridge regression with CV: Left: true regression function (red), least squares regression function (black), and ridge regression function (blue) for the optimal $\lambda = 0.001$. Right: the LOO-CV MSE as a function of $\lambda$.

# Bayesian Interpretation of Ridge Regression

Ridge regression can be viewed as a **Bayesian** approach to regression:[3]

Bayesian model:

- **Likelihood**: $p(\mathcal{D}|\boldsymbol{w}, \sigma^2) = \prod_{n=1}^{N} \mathrm{N}(y_n \mid \boldsymbol{w}^T \mathbf{x}_n, \sigma^2)$,
- **Prior**: on the model parameters $\pi(\boldsymbol{w}, \sigma^2)$,

Bayesian inference amounts to computing the **posterior**, which is proportional to the likelihood times the prior:

$$\pi(\boldsymbol{w} \mid \mathcal{D}, \sigma^2) \propto \prod_{n=1}^{N} \mathrm{N}(y_n \mid \boldsymbol{w}^T \mathbf{x}_n, \sigma^2)\, \pi(\boldsymbol{w} \mid \sigma^2)\pi(\sigma^2).$$

[3] For ease of notation, the intercept is omitted.

# Bayesian Interpretation of Ridge Regression

Ridge regression can be viewed as a **Bayesian** approach to regression:[4]

A zero-mean Gaussian prior can be used to encourage small weights:

$$\pi(\boldsymbol{w} \mid \sigma^2) = \mathrm{N}_D(\boldsymbol{w} \mid \mathbf{0}_D, s^2\sigma^2\mathbf{I_D}).$$

In this case, the posterior is:

$$\begin{aligned}\pi(\boldsymbol{w} \mid \sigma^2, \mathcal{D}) &\propto \prod_{n=1}^{N} \mathrm{N}(y_n \mid \boldsymbol{w}^T\mathbf{x}_n, \sigma^2)\,\mathrm{N}_D(\boldsymbol{w} \mid \mathbf{0}_D, s^2\sigma^2\mathbf{I_D}) \\ &= \mathrm{N}_D(\boldsymbol{w} \mid \mathbf{S}\mathbf{X}^T\mathbf{y}, \sigma^2\mathbf{S}),\end{aligned}$$

where $\mathbf{S} = (1/s^2\mathbf{I_D} + \mathbf{X^T}\mathbf{X})^{-1}$.

[4] For ease of notation, the intercept is omitted.

# Bayesian Interpretation of Ridge Regression

The **maximum aposteriori** (MAP) estimate of $\boldsymbol{w}$ is:

$$\widehat{\boldsymbol{w}}_{\text{MAP}} = \underset{\boldsymbol{w}\in\mathbb{R}^D}{\text{argmax}} \sum_{n=1}^{N} \log\left(\text{N}(y_n \mid \boldsymbol{w}^T\boldsymbol{x}_n, \sigma^2)\right) + \log\left(\text{N}_D(\boldsymbol{w} \mid \boldsymbol{0}_D, s^2\sigma^2\mathbf{I_D})\right)$$

$$= \underset{\boldsymbol{w}\in\mathbb{R}^D}{\text{argmin}} \sum_{n=1}^{N}(y_n - \boldsymbol{w}^T\boldsymbol{x}_n)^2 + \frac{1}{s^2}\sum_{d=1}^{D} w_d^2.$$

$\rightarrow$ Defining $\lambda = 1/s^2$, the ridge regression solution corresponds to MAP estimate under the zero-mean Gaussian prior.

# Bayesian Interpretation of Ridge Regression

→ Alternative approach to tune $s^2$ (or $\lambda$) through:

- Empirical Bayes: select $s^2$ to maximize the marginal likelihood.

  🤔 Find the marginal likelihood with $\sigma^2 \sim \text{IG}(\alpha, \beta)$.

- Hierarchical Bayes: prior on $s^2$; this renders a *non-separable* penalty.

  🤔 Write the Gibbs sampling steps with $\sigma^2 \sim \text{IG}(\alpha_\sigma, \beta_\sigma)$ and $s^2 \sim \text{IG}(\alpha_s, \beta_s)$.

# Lasso Regression

**Lasso regression:**[5] selects $\widehat{\boldsymbol{w}}$ to minimize:

$$\sum_{n=1}^{N}(y_n - \boldsymbol{w}^T\mathbf{x}_n)^2 + \lambda\sum_{d=1}^{D}|w_d|.$$

The tuning parameter $\lambda$ controls the amount of shrinkage.

- for $\lambda = 0$, $\widehat{\boldsymbol{w}}$ is the least squares estimate,
- as $\lambda \to \infty$, $\widehat{\boldsymbol{w}}$ approaches zero.

[5] Tibsharani (1996)

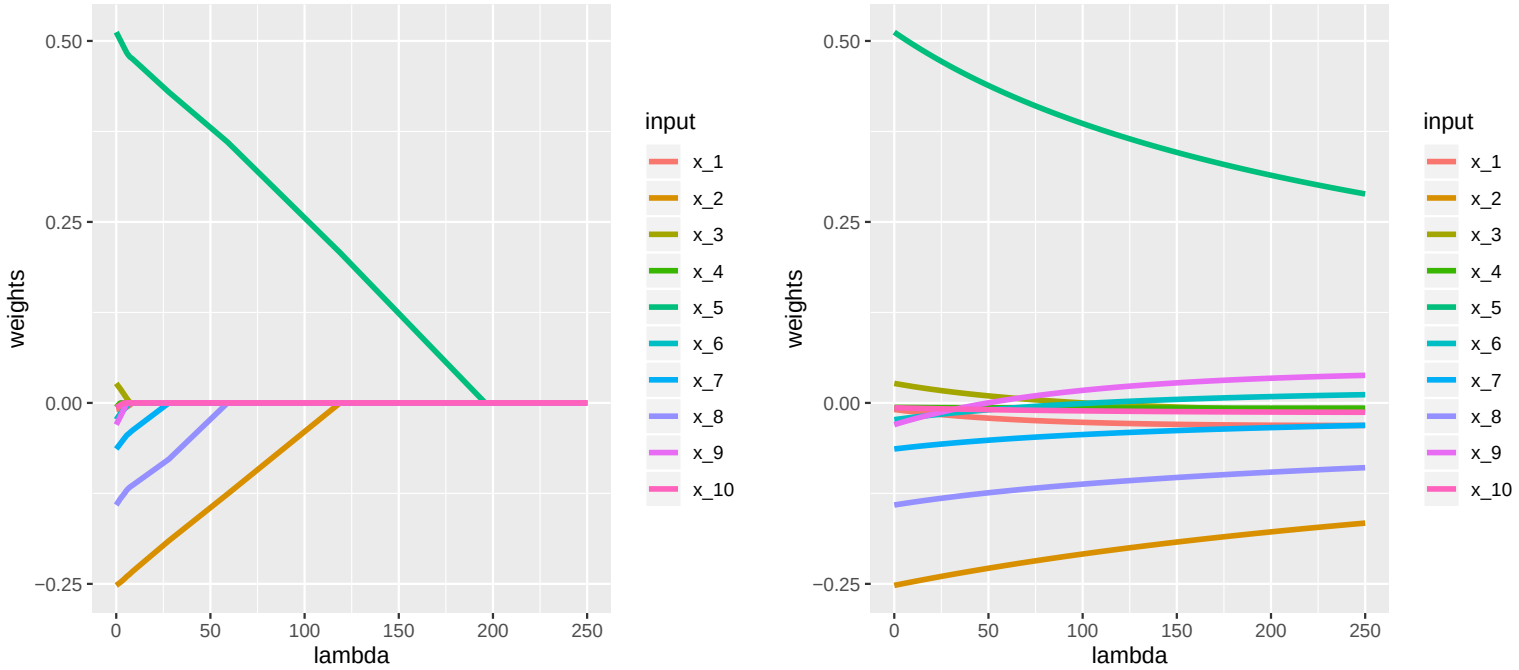
# Lasso vs. Ridge

**Ridge:** shrinks the weights towards zero but does not set any of them exactly to zero $\rightarrow$ all inputs are included in the final model.

**Lasso:** forces some of the weights to be exactly zero, for sufficiently large $\lambda$, $\rightarrow$ only a subset of inputs are included in the final model.

This is relevant to high-dimensional settings (e.g. DNA studies), when we are interested in **variable selection**: obtaining **sparse** estimates of $\boldsymbol{w}$ that are nonzero for only a subset of inputs.

# Lasso vs. Ridge: Simulated Example




(a) Lasso (b) Ridge

Figure: Lasso vs. Ridge regression: $N = 50$ data points with $D = 10$ are generated from a linear regression model where only the second, fifth, and eighth inputs have nonzero weights. The lasso (left) and ridge (right) regression weights are shown as a function of $\lambda$.

# Lasso and Ridge with CV: Simulated Example

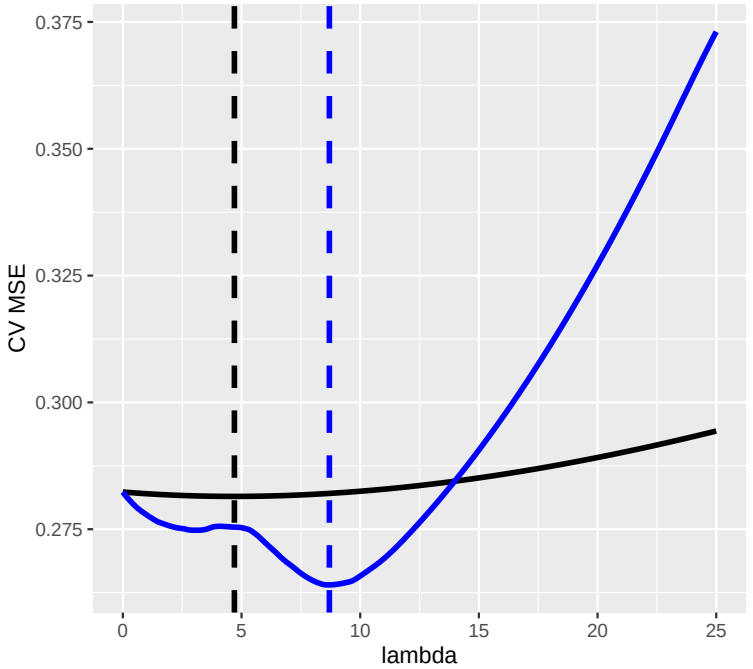



Figure: Cross-validation is also used to select the optimal $\lambda$ for lasso. LOO-CV MSE as a function of $\lambda$ for lasso in blue and ridge in black. The dashed vertical line indicates the value of lambda that minimizes the LOO-CV MSE. The optimal $\lambda \approx 8.7$ for lasso corresponds to the model with only $x_2$, $x_5$, $x_7$, and $x_8$. The minimum for ridge is not well pronounced, so there is a wide range of values that give similar error.

# Why does Lasso result in sparse solutions?

One can show that lasso equivalently solves the constrained problem:

$$\min_{\boldsymbol{w}\in\mathbb{R}^D} \text{RSS}(\boldsymbol{w}) \quad \text{s.t.} \quad \sum_{d=1}^{D} |w_d| \leq B,$$

and ridge equivalently solves:

$$\min_{\boldsymbol{w}\in\mathbb{R}^D} \text{RSS}(\boldsymbol{w}) \quad \text{s.t.} \quad \sum_{d=1}^{D} w_d^2 \leq B,$$

where the bound $B$ is defined by $\lambda$.

# Geometric Visualization of Lasso and Ridge

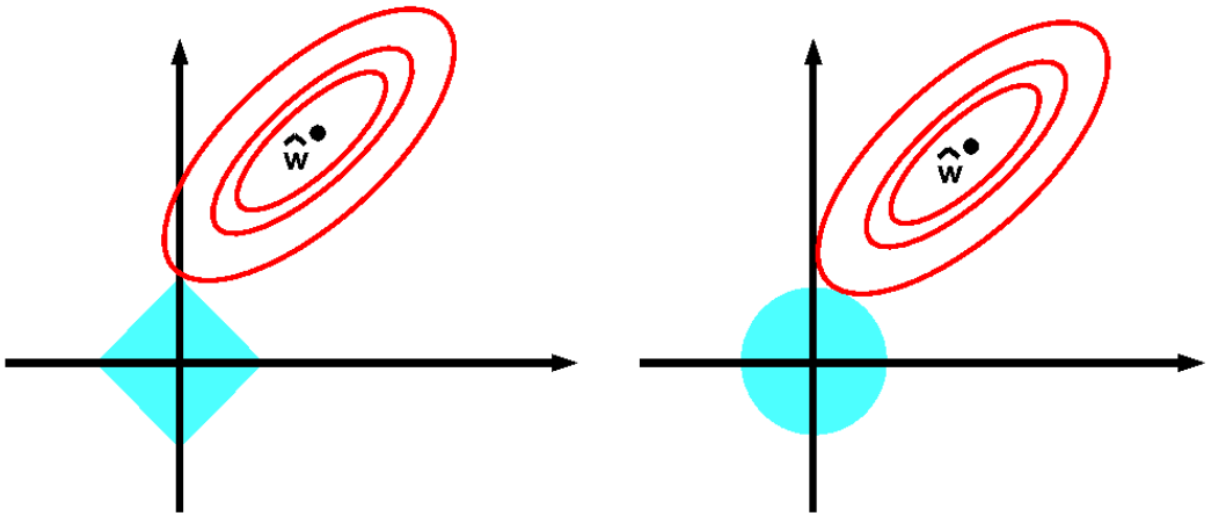



Figure: Geometric depiction of lasso and ridge for $D = 2$. The shaded areas are the constrained regions for lasso (left) and ridge (right), while the red ellipses around the least squares solution are the contours of the RSS.

From James et al. (2023), *An Introduction to Statistical Learning*.

# Why does Lasso result in sparse solutions?

The solution to the constrained optimization problem is found by the point at which the ellipse first contacts the constrained region:

- the lasso constrained region has corners and the ellipse will often intersect at a corner, especially in high-dimensions, $\rightarrow$ sparse $\widehat{\boldsymbol{w}}$.
- the ridge constrained region has no corners and the intersection can occur at any point $\rightarrow$ non-sparse $\widehat{\boldsymbol{w}}$.

# What are the lasso regression weights?

The lasso objective function:

$$\text{RSS}(\boldsymbol{w}) + \lambda \sum_{d=1}^{D} |w_d|,$$

is not differentiable at $w_d = 0$. Using *subgradients*[6], the optimal $\widehat{w}_d$ given $\widehat{\boldsymbol{w}}_{-d}$ is:

$$\widehat{w}_d = \begin{cases} (c_d + \lambda/2)/a_d & \text{if } c_d < -\lambda/2 \\ 0 & \text{if } c_d \in [-\lambda/2, \lambda/2] \\ (c_d - \lambda/2)/a_d & \text{if } c_d > \lambda/2 \end{cases},$$

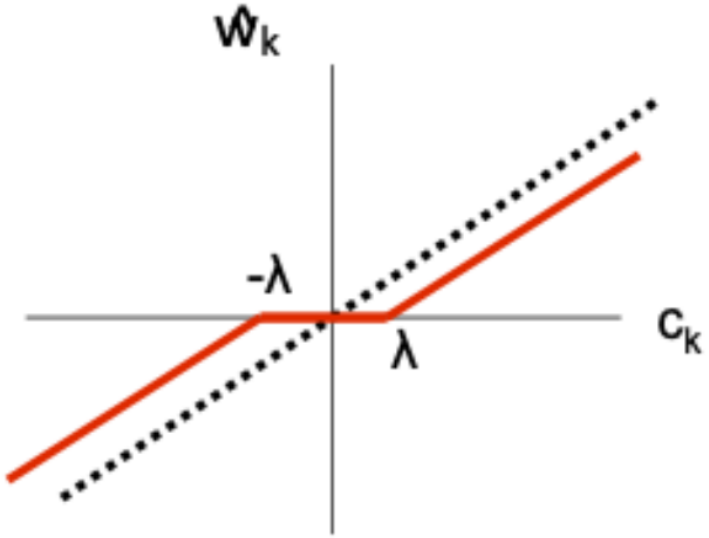
where

$$c_d = \sum_{n=1}^{N} x_{n,d}(y_n - \sum_{d' \neq d} w_{d'} x_{n,d'}), \quad a_d = \sum_{n=1}^{N} x_{n,d}^2.$$

$\Rightarrow$ provides a coordinate descent (shooting algorithm) for lasso.

[6] for details, see Section 13.3.2 of K. Murphy (2012)

# What are the lasso regression weights?




The lasso regression weights use a type of shrinkage known as **soft-thresholding**. Even large weights are shrunk towards zero $\rightarrow$ lasso is biased. A simple solution is to re-estimate using least squares for the non-zero features.

# When should we use lasso vs. ridge?

Neither ridge nor lasso will universally dominate in terms of predictive accuracy:

- lasso performs better when a small number of inputs have substantial weights and the remaining are close to zero.
- ridge regression performs better when the output is a function of many inputs, all with roughly equal weight.

If such information is not known, cross-validation can be used to determine which approach is better for a particular dataset.

# Lasso and Ridge with CV: Simulated Example

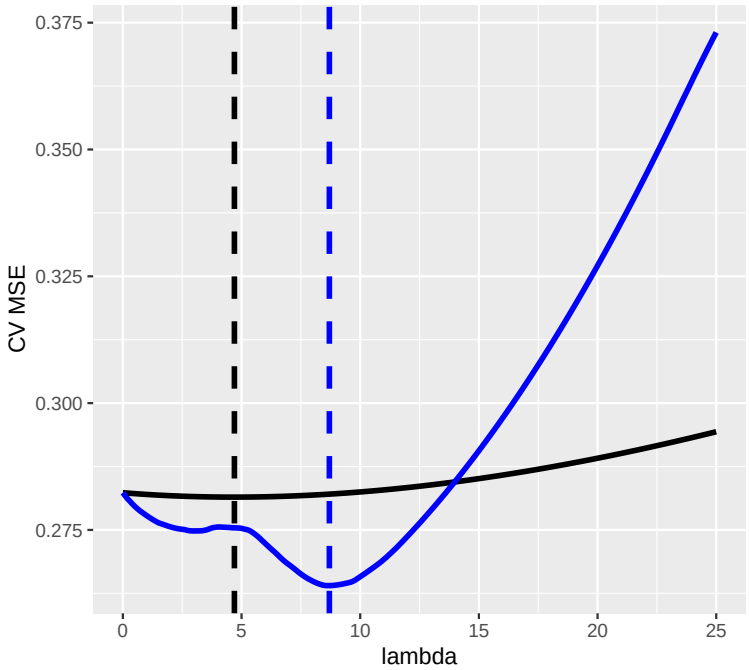



Figure: Is lasso or ridge better? Does this make sense?

# Bayesian Interpretation of Lasso Regression

Lasso regression can also be viewed as a **Bayesian** approach to regression[a]. In this case, a **double-exponential** (or **Laplace**) prior with zero-mean is used to encourage the weights to be small:

$$\pi(\boldsymbol{w} \mid \sigma^2) = \prod_{d=1}^{D} \frac{1}{2s\sigma} \exp\left(-\frac{1}{s\sigma}|w_d|\right).$$

[a] Park and Casella (2008)

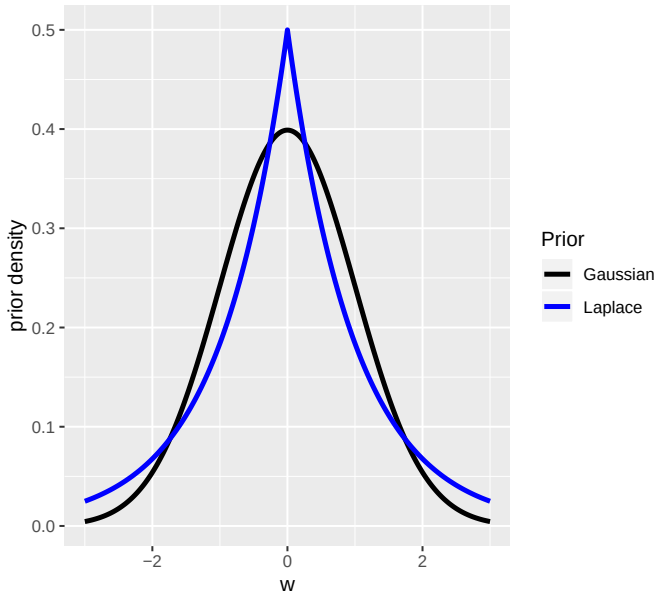



Figure: Gaussian and Laplace priors: Laplace prior is steeply peaked at zero, while the Gaussian prior is flatter and fatter at zero.

# Bayesian Interpretation of Lasso Regression

Under the Laplace prior, the MAP estimate of $\boldsymbol{w}$ is[7]:

$$\begin{aligned}\widehat{\boldsymbol{w}}_{\text{MAP}} &= \underset{\boldsymbol{w}\in\mathbb{R}^D}{\text{argmax}} \log(\pi(\boldsymbol{w}|\sigma^2, \mathcal{D})) \\ &= \underset{\boldsymbol{w}\in\mathbb{R}^D}{\text{argmax}} \sum_{n=1}^{N} \log\left(\text{N}(y_n \mid \boldsymbol{w}^T\boldsymbol{x}_n, \sigma^2)\right) - \sum_{d=1}^{D} \frac{1}{s\sigma}|w_d| \\ &= \underset{\boldsymbol{w}\in\mathbb{R}^D}{\text{argmin}} \sum_{n=1}^{N}(y_n - \boldsymbol{w}^T\boldsymbol{x}_n)^2 + \frac{2\sigma}{s}\sum_{d=1}^{D}|w_d|.\end{aligned}$$

Defining $\lambda = 2\sigma/s$, the lasso solution corresponds to the MAP estimate under the zero-mean Laplace prior.

[7] For ease of notation, the intercept is omitted and we condition on $\sigma^2$.

# Bayesian Lasso Posterior

The posterior is not available in closed-form:

$$\pi(\boldsymbol{w} \mid \sigma^2, \mathcal{D}) \propto \prod_{n=1}^{N} \mathrm{N}(y_n \mid \boldsymbol{w}^T \mathbf{x}_n, \sigma^2) \prod_{d=1}^{D} \frac{1}{2s\sigma} \exp\left(-\frac{1}{s\sigma}|w_d|\right).$$

However, we can use a data augmentation trick, by noting that:

$$\frac{1}{2s} \exp\left(-\frac{1}{s}|w|\right) = \int \frac{1}{\sqrt{2\pi\tau}} \exp\left(-\frac{1}{2\tau}w^2\right) \frac{1}{2s^2} \exp\left(-\frac{1}{2s^2}\tau\right) d\tau,$$

i.e. the Laplace distribution $w \sim \text{Laplace}(0, s)$ can be represented as a scale mixture of normals (with an gamma mixing density):

$$w \mid \tau \sim \mathrm{N}(0, \tau), \quad \tau \sim \text{Gamma}(1, 1/(2s^2)).$$

# The Bayesian Lasso Hierarchical Model

Thus, we can obtain a hierarchical representation of the Bayesian lasso:

$$
\begin{aligned}
y_n \mid \mathbf{x}_n, \boldsymbol{w}, \sigma^2 &\sim \mathrm{N}(y_n \mid \boldsymbol{w}^T \mathbf{x}_n, \sigma^2), \\
\boldsymbol{w} \mid \sigma^2, \boldsymbol{\tau} &\sim \mathrm{N}(0, \sigma^2 \mathbf{D}), \\
\tau_d &\sim \mathrm{Gamma}(1, 1/(2s^2)), \\
\sigma^2 &\sim \mathrm{IG}(\alpha_\sigma, \beta_\sigma),
\end{aligned}
$$

where $\mathbf{D} = \mathrm{diag}(\tau_1, \ldots, \tau_D)$.

$\rightarrow$ Used to derive a Gibbs sampling algorithm.

# Gibbs Sampling

Step 1 [8] sample $\boldsymbol{w}$:

$$\boldsymbol{w} \mid \ldots \sim \mathrm{N}_D(\boldsymbol{w} \mid \mathbf{S}\mathbf{X}^T\mathbf{y}, \sigma^2\mathbf{S}),$$

where $\mathbf{S} = (\mathbf{D}^{-1} + \mathbf{X}^T\mathbf{X})^{-1}$.

Step 2 sample $\sigma^2$:

$$\sigma^2 \mid \ldots \sim \mathrm{IG}(\widehat{\alpha}_\sigma, \widehat{\beta}_\sigma),$$

where $\widehat{\alpha}_\sigma = \alpha_\sigma + N/2 + D/2$ and
$\widehat{\beta}_\sigma = \beta_\sigma + 1/2((\mathbf{y} - \mathbf{X}\boldsymbol{w})^T(\mathbf{y} - \mathbf{X}\boldsymbol{w}) + \boldsymbol{w}\mathbf{D}^{-1}\boldsymbol{w})$.

Step 3 sample $\boldsymbol{\tau}$:

$$1/\boldsymbol{\tau} \sim \mathrm{InvGaus}\left(\sqrt{\frac{\sigma^2}{s^2 w_d^2}}, 1/s^2\right).$$

[8] 🤔 It is also possible to sample $\boldsymbol{w}$ and $\sigma^2$ jointly from a normal-inverse gamma. Write the steps and compare the computational complexity.

# Tuning $\lambda$ in Bayesian Lasso

Bayesian lasso offers an alternative approach to tune $s$ (or $\lambda$) through:

- Hierarchical Bayes: prior[9] on $s^2$; this again renders a *non-separable* penalty.

  🤔 Show that with the conjugate prior $s^2 \sim \text{IG}(\alpha_s, \beta_s)$, the Gibbs sampling step is:

$$s^2 \mid \ldots \sim \text{IG}\left(\alpha_s + D, \beta_s + 1/2 \sum_{d=1}^{D} \tau_d\right)$$

[9] "the prior should be relatively flat and place high probability near the maximum likelihood estimate" (Park and Casella (2008))

# Tuning $\lambda$ in Bayesian Lasso

- Empirical Bayes: select $s$ to an approximate maximum marginal likelihood estimate through Monte Carlo EM:

  Note that the *complete* log-likelihood for $s$ is:

$$\ell(s) = \text{const} - 2D\log(s) - \frac{1}{2s^2}\sum_{d=1}^{D}\tau_d.$$

  Monte Carlo EM algorithm iterates between:

E-step : take expectation given $s^{(k-1)}$

$$\mathrm{E}[\ell(s) \mid \mathcal{D}] = \text{const} - 2D\log(s) - \frac{1}{2s^2}\sum_{d=1}^{D}\mathrm{E}[\tau_d \mid \mathcal{D}].$$

M-step : maximize to obtain:

$$s^{(k)} = \sqrt{\frac{1}{2D}\sum_{d=1}^{D}\mathrm{E}[\tau_d \mid \mathcal{D}]}$$

# How do we select variables with Bayesian Lasso?

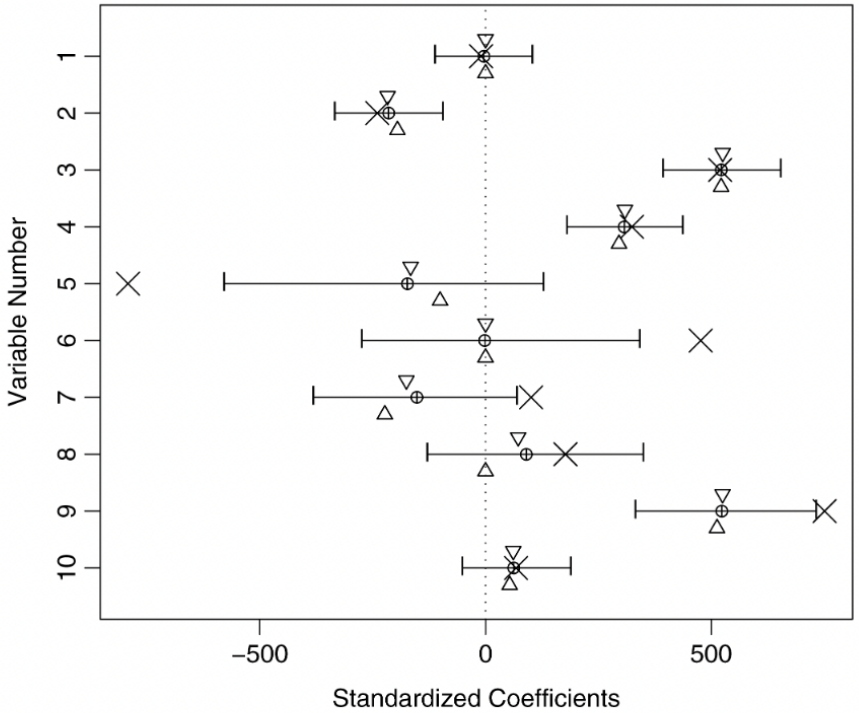



Figure 2. Posterior median Bayesian Lasso estimates ($\oplus$) and corresponding 95% credible intervals (equal-tailed) with $\lambda$ selected according to marginal maximum likelihood (Sec. 3.1). Overlaid are the least squares estimates ($\times$), Lasso estimates based on $n$-fold cross-validation ($\triangle$), and Lasso estimates chosen to match the $L_1$ norm of the Bayes estimates ($\triangledown$). The variables were described by Efron et al. (2004): (1) age, (2) sex, (3) bmi, (4) map, (5) tc, (6) ldl, (7) hdl, (8) tch, (9) ltg, and (10) glu.

# How do we select variables with Bayesian Lasso?

Note that the posterior mean of $\boldsymbol{w}$ does not provide variable selection, instead one may use:

- credible interval criterion (Li and Lin (2010)), with standard threshold (e.g. 95%) or selected threshold to minimize the estimated false discovery rate.

$$\widehat{\text{FDR}}(\kappa) = \frac{\sum_{d=1}^{D}(1-p_d)(p_d > \kappa)}{\sum_{d=1}^{D}(p_d > \kappa)},$$

  where $p_d = \max(P(w_d > 0 \mid \mathcal{D}), P(w_d < 0 \mid \mathcal{D}))$.
- Various other proposals motivated from decision theory include: Hahn and Carvalho (2014); Li and Pati (2017); Ray and Bhattacharya (2018); Piironen et al (2020); Griffin (2024), and more.

# Drawbacks of ridge and lasso

- lasso suffers when groups of inputs are highly correlated, with less stable solutions.
- lasso can only select at most $N$ inputs when $D > N$.
- ridge does not produce sparse solutions.

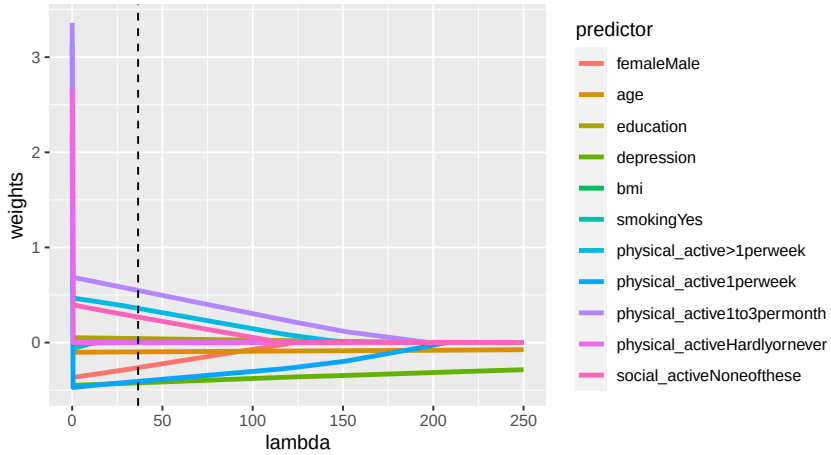



Figure: Predicting dementia severity (SHARE data): The lasso regression weights are shown as a function of $\lambda$. The optimal $\lambda$ is depicted with a dashed vertical line and corresponds to the model with all predictors except, BMI and smoking.

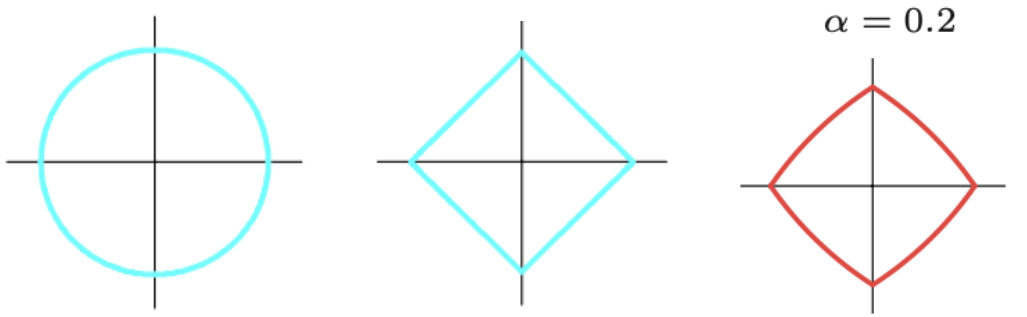
# Elastic Net

**Elastic net**[10] is a hybrid between lasso and ridge:

$$p_{\lambda,\alpha}(w) = \lambda(1-\alpha)|w| + \lambda\alpha w^2.$$

+ exhibits a **grouping effect**: coefficients of highly correlated variables tend to be equal.
+ can select more than $N$ non-zero inputs.

🤔 Lasso and ridge are special cases for what choices of $\alpha$?




(a) Ridge (b) Lasso (c) Elastic Net

Figure: From ESL (2009)

[10] Zou and Hastie (2005)

# Elastic Net

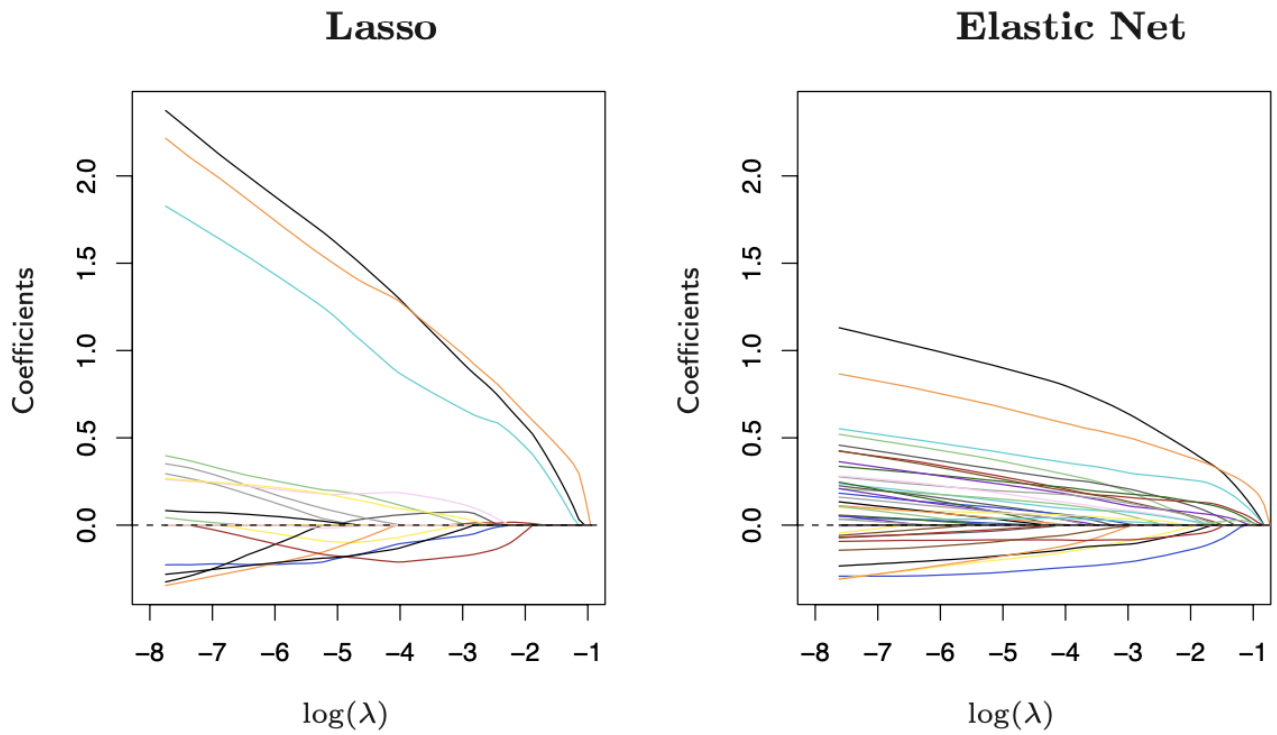



Figure: From ESL (2009). Solution paths for the leukemia data, for lasso (left) and elastic net with $\alpha = 0.8$ (right). At the end of the path (extreme left), there are 19 nonzero coefficients for lasso and 39 for elastic net; elastic net results in more non-zero coefficients but with smaller magnitudes.

# Elastic Net

Consider the augmented data:

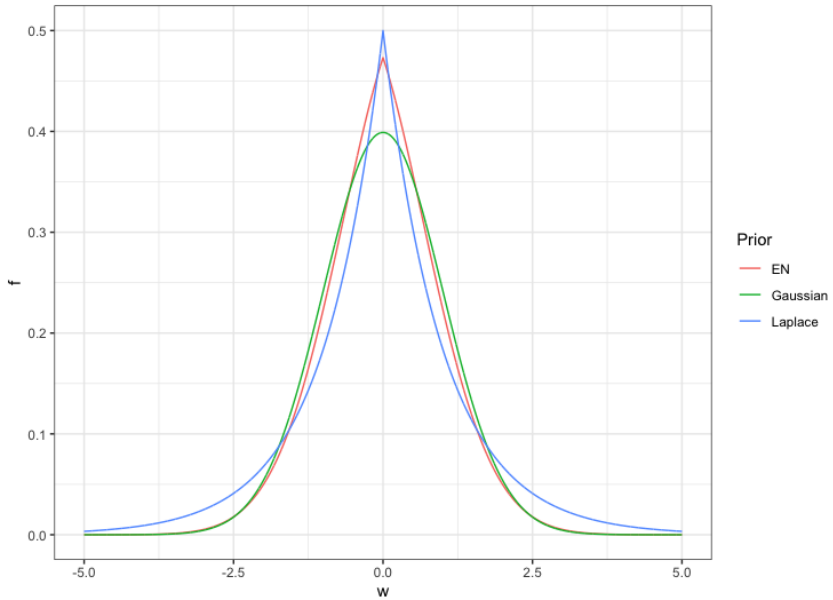
$$\mathbf{y}' = \begin{bmatrix} \mathbf{y} \\ \mathbf{0}_D \end{bmatrix} \quad \text{and} \quad \mathbf{X}' = \begin{bmatrix} \mathbf{X} \\ \sqrt{\lambda\alpha}\mathbf{I_D} \end{bmatrix}.$$

🤔 Show that the elastic net problem reduces to a lasso problem on the augmented data.

# Bayesian Elastic Net

Elastic net can also be viewed as MAP estimation in a Bayesian approach[11], with the following prior:

$$\pi(\boldsymbol{w} \mid \sigma^2) \propto \prod_{d=1}^{D} \exp\left(-\frac{\lambda}{2\sigma^2}(\alpha w_d^2 + (1-\alpha)|w_d|)\right).$$




[11] Li and Lin (2010)

# Bayesian Elastic Net

Under the Elastic Net prior, the MAP estimate of $\boldsymbol{w}$ is[12]:

$$\begin{aligned}
\widehat{\boldsymbol{w}}_{\text{MAP}} &= \underset{\boldsymbol{w}\in\mathbb{R}^D}{\text{argmax}} \log(\pi(\boldsymbol{w}|\sigma^2, \mathcal{D})) \\
&= \underset{\boldsymbol{w}\in\mathbb{R}^D}{\text{argmax}} \sum_{n=1}^{N} \log\left(\text{N}(y_n \mid \boldsymbol{w}^T\boldsymbol{x}_n, \sigma^2)\right) - \sum_{d=1}^{D} \frac{\lambda}{2\sigma^2}(\alpha w_d^2 + (1-\alpha)|w_d|) \\
&= \underset{\boldsymbol{w}\in\mathbb{R}^D}{\text{argmin}} \sum_{n=1}^{N}(y_n - \boldsymbol{w}^T\boldsymbol{x}_n)^2 + \lambda \sum_{d=1}^{D}(\alpha w_d^2 + (1-\alpha)|w_d|).
\end{aligned}$$

[12] For ease of notation, the intercept is omitted and we condition on $\sigma^2$.

# Bayesian Elastic Net Posterior

The posterior is not available in closed-form:

$$\pi(\mathbf{w} \mid \sigma^2, \mathcal{D}) \propto \prod_{n=1}^{N} \mathrm{N}(y_n \mid \mathbf{w}^T \mathbf{x}_n, \sigma^2) \prod_{d=1}^{D} \exp\left(-\frac{\lambda}{2\sigma^2}(\alpha w_d^2 + (1-\alpha)|w_d|)\right).$$

Again, we can use a data augmentation trick, by noting that:

$$\exp\left(-\frac{\lambda}{2\sigma^2}(\alpha w^2 + (1-\alpha)|w|)\right)$$
$$= \int_1^\infty \sqrt{\frac{\tau}{\tau-1}} \exp\left(-\frac{w^2}{2}\frac{\lambda\alpha}{\sigma^2}\frac{\tau}{\tau-1}\right) \tau^{-1/2} \exp\left(-\frac{1}{2\sigma^2}\frac{\lambda(1-\alpha)^2}{4\alpha}\tau\right) d\tau.$$

# The Bayesian Elastic Net Hierarchical Model

Thus, we can obtain a hierarchical representation of the Bayesian elastic net:

$$
\begin{aligned}
y_n \mid \boldsymbol{x}_n, \boldsymbol{w}, \sigma^2 &\sim \mathrm{N}(y_n \mid \boldsymbol{w}^T \boldsymbol{x}_n, \sigma^2), \\
\boldsymbol{w} \mid \sigma^2, \boldsymbol{\tau} &\sim \mathrm{N}(0, \sigma^2 \mathbf{D}), \\
\tau_d \mid \sigma^2 &\sim \mathrm{TrGam}_{(1,\infty)} \left( \frac{1}{2}, \frac{\lambda(1-\alpha)^2}{8\sigma^2 \alpha} \right), \\
\sigma^2 &\sim \mathrm{IG}(\alpha_\sigma, \beta_\sigma),
\end{aligned}
$$

where $\mathbf{D} = 1/(\lambda\alpha)\mathrm{diag}((1-\tau_1)/\tau_1, \ldots, (1-\tau_D)/\tau_D)$.

🤔 derive a Gibbs sampling algorithm.

# Bayesian Elastic Net: Prostate Example

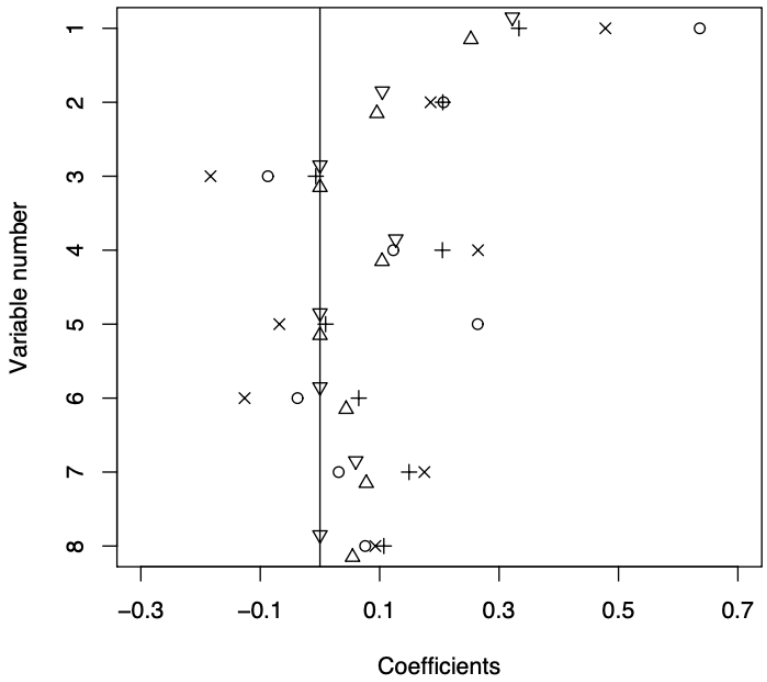



Figure 5: The estimates of the predictor effects for the prostate cancer data using different methods: posterior median BEN estimates (+), posterior median BL estimates (○), EN estimates (△), lasso estimates (∇) and the OLS estimates (×).

# Summary

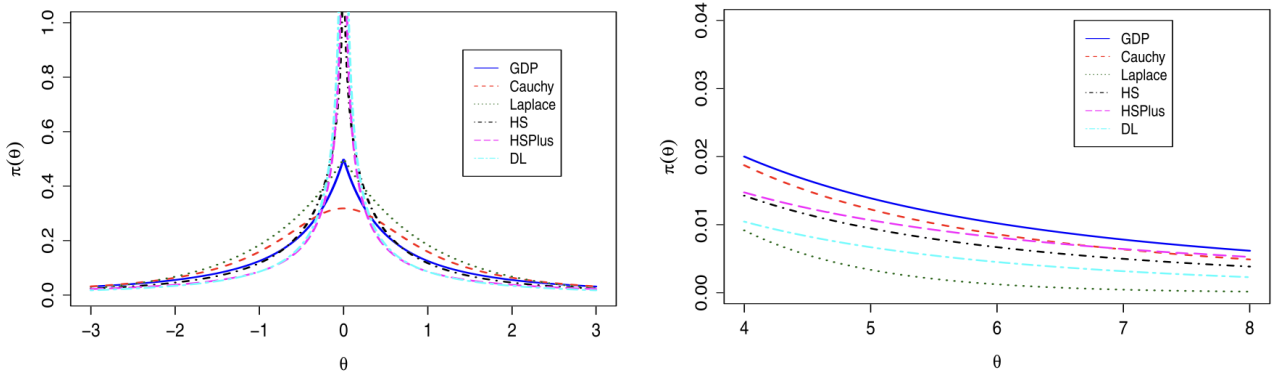
Regularization methods are critical to cope with both high-dimensional data and heavily over-parameterized models.

- Ridge, lasso, and elastic net are widely used; but only provide estimates of the coefficients and require tuning of the hyperparameters via cross-validation.
- They can naturally be viewed as MAP estimates in a Bayesian analysis, providing:
  - alternative interpretation of the penalty and extensions (non-separable);
  - posterior inference and uncertainty for the coefficients;
  - hyperparameter tuning via hierarchical or empirical Bayes.

# Shrinkage Priors Beyond Ridge, Lasso, Elastic Net

However, the Bayesian Lasso cannot simultaneously adapt to sparsity and avoid the estimation bias issue of the original LASSO (Rockova and George (2016)).

To avoid this, alternative priors place greater mass around zero and have heavier tails, notably global-local shrinkage priors (Bhadra et al (2019)).




On the other hand, spike-and-slab priors are a natural choice for variable selection, but lead to substantial computational challenges.

# Further Reading

See various papers referenced within, as well as the books and chapters:

- Chp. 6 in *An Introduction to Statistical Learning*, James et al. (2023).
- Chp. 11 in *Probabilistic Machine Learning: An Introduction*, K. Murphy (2022).
- Chp. 3 & 18 in *Elements of Statistical Learning*, Hastie et al. (2009).
- *Handbook of Bayesian Variable Selection*, Tadesse & Vanucci (2022).

For a tutorial in `Rstan`: https://betanalpha.github.io/assets/case_studies/modeling_sparsity.html