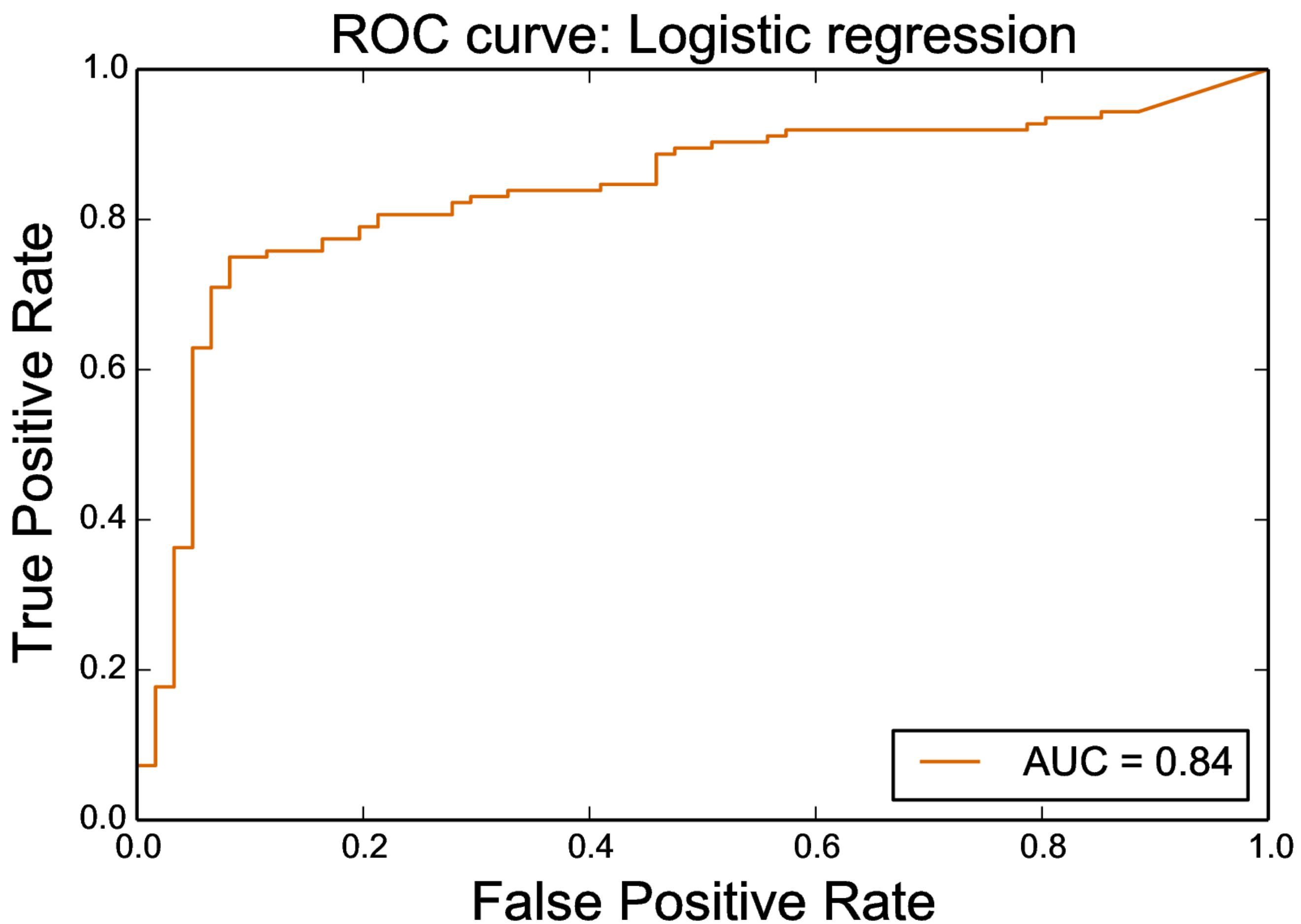
Foundations of Machine Learning
CentraleSupélec — Fall 2016

6. Regularized linear regression

Chloé-Agathe Azencott
Centre for Computational Biology, Mines ParisTech
chloe-agathe.azencott@mines-paristech.fr

Logistic regression on the Endometrium vs Uterus data



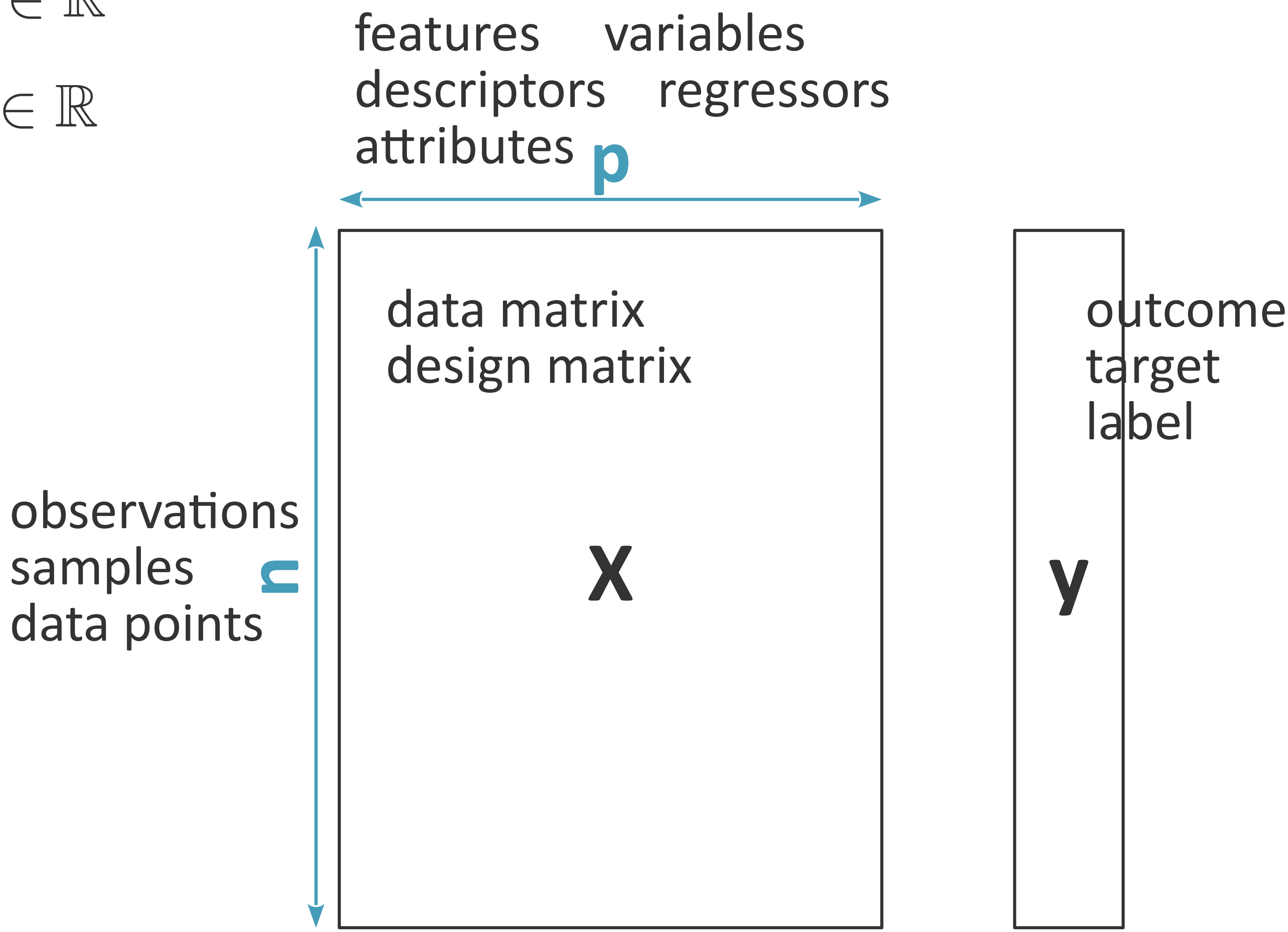
Learning objectives

- Understand **regularization** as a means to control model complexity.
- Define **Lasso**, **ridge regression**, **elastic net.**
- Understand the role of the **l1 and l2 norms** in regularization
- Interpret **solution paths** for Lasso and ridge regression.

Regression setting

$x_j^i \in \mathbb{R}$

$y^i \in \mathbb{R}$



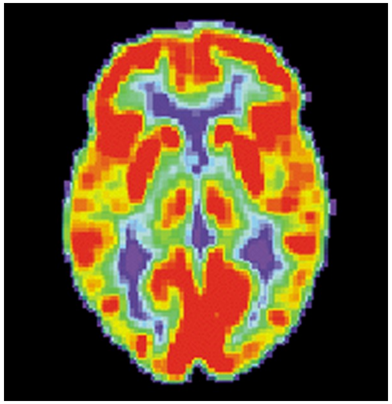
Large p, small n

E.g.

– **neuroimaging**

thousands of brain regions / pixels / voxels

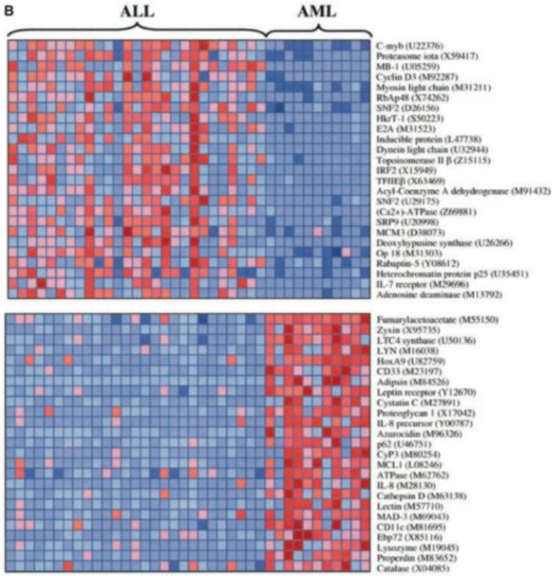
much fewer patients



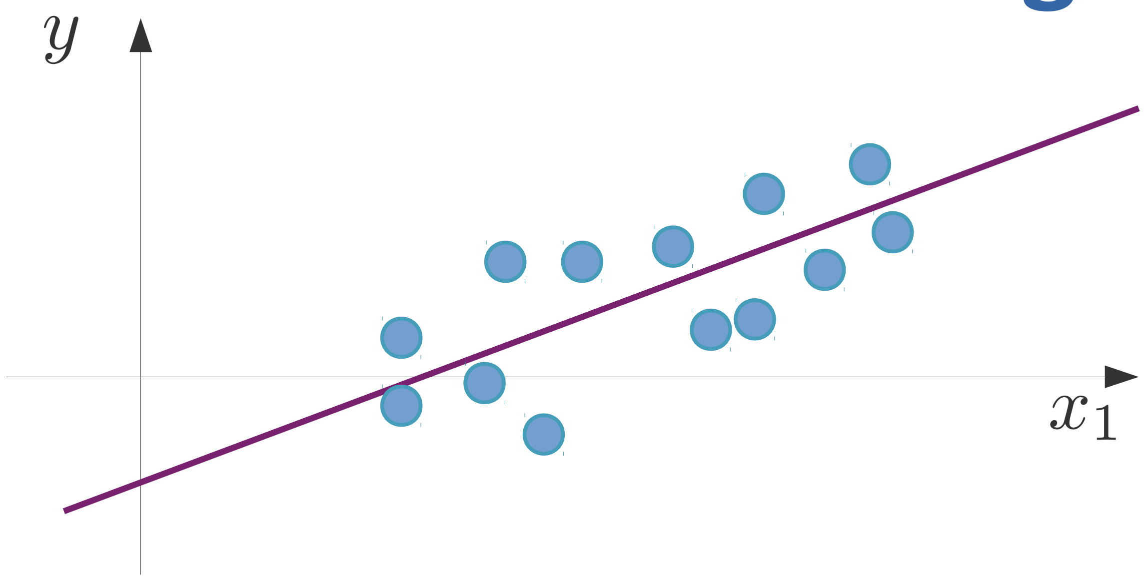
– **genetics and genomics**

thousands of genes, millions of SNPs...

usually, at best thousands of patients



Linear regression



$$\boldsymbol{x} \in \mathbb{R}^p$$

$$f(\boldsymbol{x}|\beta) = \sum_{j=1}^{p} \beta_j x_j + \beta_0$$

Least-squares fit (equivalent to MLE under the assumption of Gaussian noise):

$$\hat{\beta} = \arg\min_{\beta} (y - X\beta)^\top (y - X\beta) = (X^\top X)^{-1} X^\top y$$

The solution is uniquely defined when **$n > p$** and **$X^\top X$ invertible**.

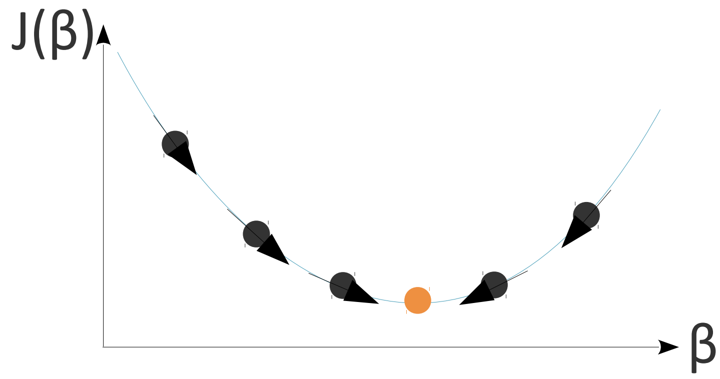
When $X^T X$ not inversible

$$(X^\top X)\hat{\beta} = X^\top y$$

- Pseudo-inverse
- Linear system of p equations:

  Numerical methods

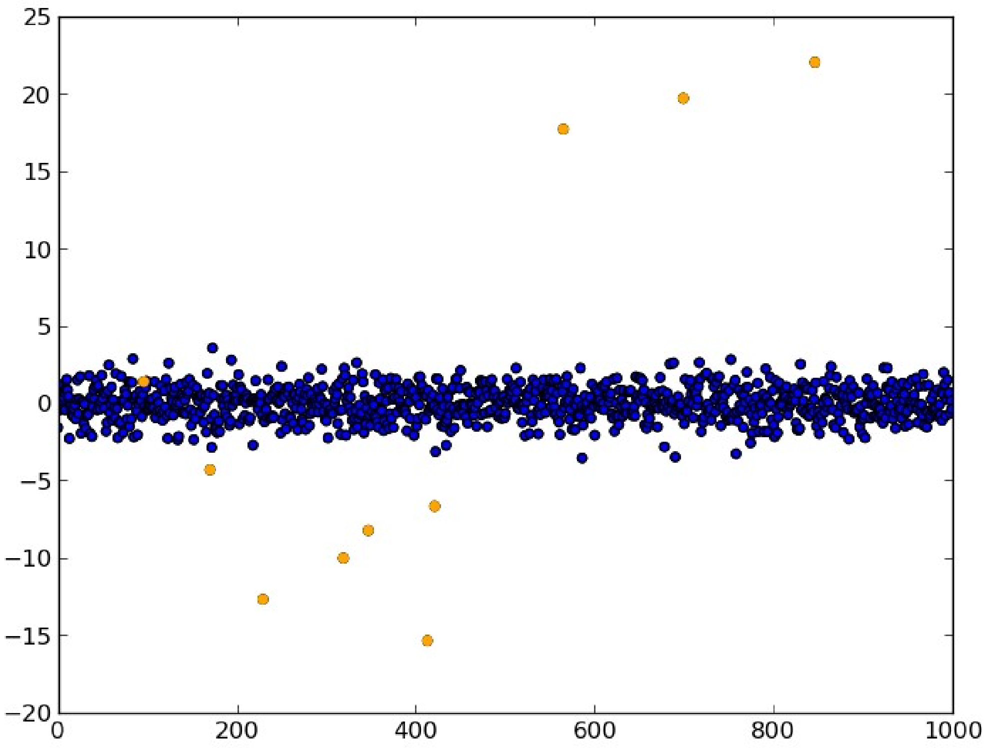
  - **Gaussian elimination**
  - **LU decomposition**
  - **Gradient descent**



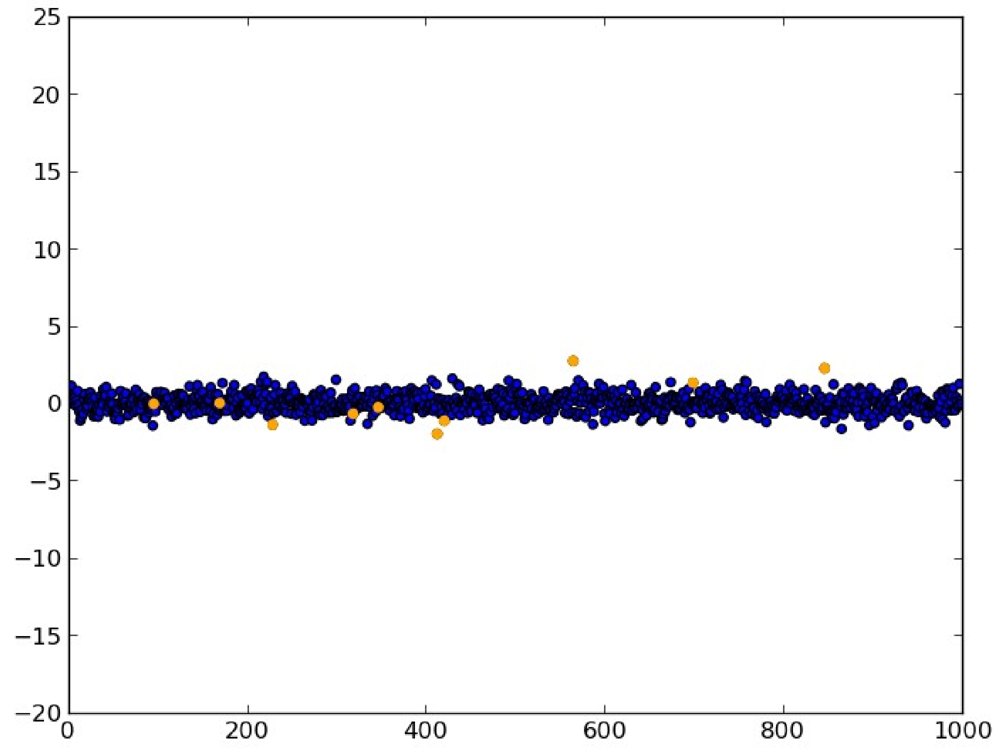
Linear regression when p >> n

Simulated data: p=1000, n=100, 10 causal features

True coefficients



Predicted coefficients



Advantages of least-squares fit

- **Unbiased** $E[\hat{\beta}] = \beta$
- **Explicit form**
- **Computational time?**

Advantages of least-squares fit

- **Unbiased** $E[\hat{\beta}] = \beta$
- **Explicit form**
- **Computational time:** $O(np^2 + p^3)$

compute $X^T X$ invert $X^T X$

computation of $X^T y$: $O(np)$
computation of $(X^T X)^{-1} X^T y$: $O(np)$

Cons of least-squares fit

- **Multicollinearity** leads to high variance of the estimator
- Requires $n > p$
- Prediction error increases linearly as a function of p
- Hard to **interpret** when p is large

Would prefer a small subset with strong effects.

Regularization

Regularization

- Minimize

SSE + **λ** penalty on model complexity

- **Biased estimator** when $\lambda \neq 0$.
- Trade bias for a smaller variance.
- λ can be set by cross-validation.

- Simpler model ≈ fewer parameters

  → **shrinkage:** drive the coefficients of the parameters towards 0.

Ridge regression

Ridge regression

- **Sum-of-squares penalty**

$$\hat{\beta}_{\text{ridge}} = \arg\min_{\beta} ||y - X\beta||_2^2 + \lambda||\beta||_2^2$$

- **Compute the ridge regression estimator.**

Ridge regression

- **Sum-of-squares penalty**

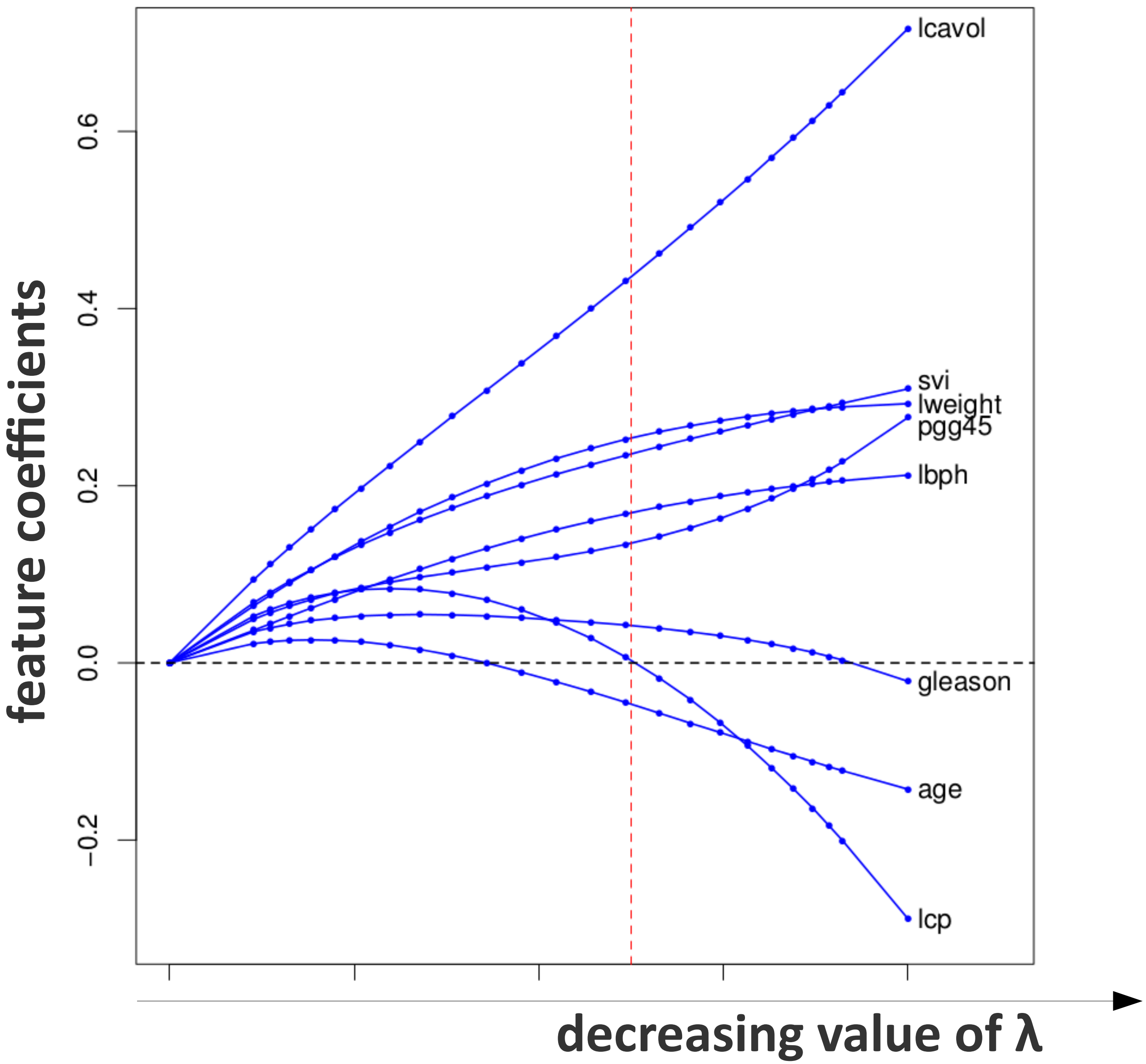
$$\hat{\beta}_{\text{ridge}} = \arg\min_{\beta} ||y - X\beta||_2^2 + \lambda||\beta||_2^2$$

- **Compute the ridge regression estimator.**

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

$$\text{if } (X^\top X + \lambda I) \text{ invertible.}$$

Ridge regression solution path



Standardization

- **What happens if we multiply x_j by a constant?**
  - For **standard linear regression**
  - For **ridge regression**

Standardization

- What happens if we multiply x_j by a constant?
  - For **standard linear regression:**

$$\hat{\beta}_j \rightarrow \frac{1}{c}\hat{\beta}_j$$

  - For **ridge regression:**

    Not so clear, because of the penalization term $\lambda\beta_j^2$
- Need to **standardize** the features

$$\tilde{x}_j^i = \frac{x_j^i}{\sqrt{\frac{1}{n}\sum_{i=1}^n (x_j^i - \bar{x}_j)^2}}$$

average value of x_j

Ridge regression

- **Grouped selection:**
  - correlated variables get similar weights
  - identical variables get identical weights
- Ridge regression shrinks coefficients towards 0 but does not result in a **sparse model**.
- **Sparsity:**
  - many coefficients get a weight of 0
  - they can be eliminated from the model.

Lasso

Lasso

$$||\beta||_1 = \sum_{j=1}^{p} |\beta_p|$$

- **L1 penalty**

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} ||y - X\beta||_2^2 + \lambda||\beta||_{\boxed{1}}$$

- aka **basis pursuit** (signal processing)
- no closed-form solution
- **quadratic programming** problem: equivalent to

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} ||y - X\beta||_2^2 \quad \text{s.t. } ||\beta||_1 \leq t$$

for a unique one-to-one match between t and λ.

QP: maximize a quadratic form under linear constraints.

Equivalence between the formulations

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} ||y - X\beta||_2^2 \quad \text{s.t. } ||\beta||_1 \leq t$$

- **minimize f(β) under the constraint g(β) ≤ 0**

$$f(\beta) = ||y - X\beta||_2^2 \qquad\qquad g(\beta) = ||\beta||_1 - t$$

Equivalence between the formulations

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} ||y - X\beta||_2^2 \quad \text{s.t. } ||\beta||_1 \leq t$$

- **minimize f(β) under the constraint g(β) ≤ 0**

$f(\beta) = ||y - X\beta||_2^2$ $\qquad$ $g(\beta) = ||\beta||_1 - t$

Case 1: the unconstrained minimum lies in the **feasible region.** $\{\beta : g(\beta) \leq 0\}$

Equivalence between the formulations

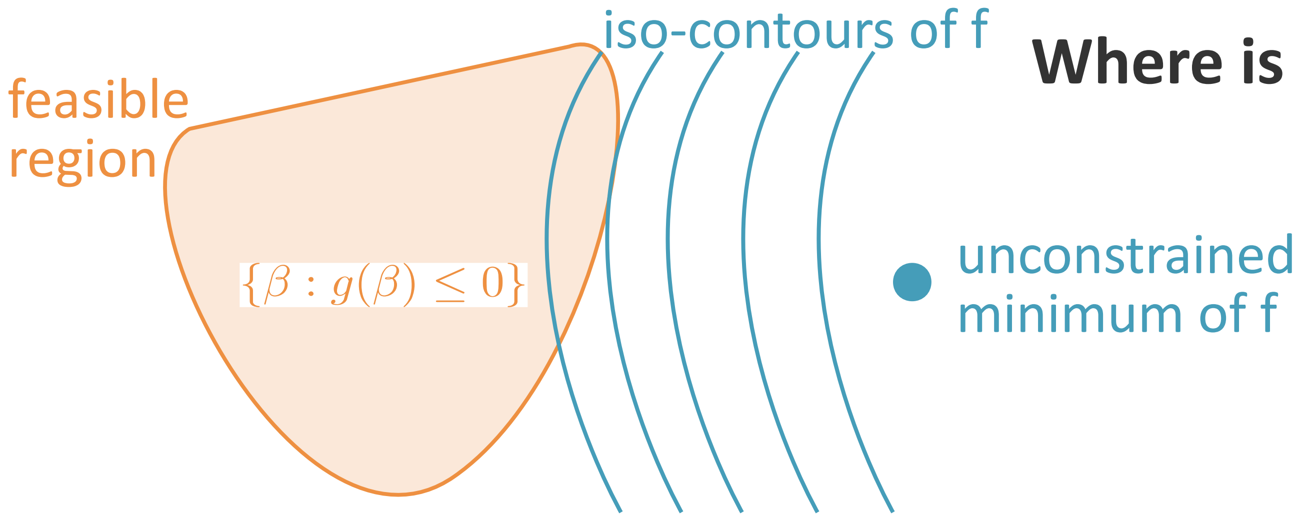
$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} ||y - X\beta||_2^2 \quad \text{s.t.} \quad ||\beta||_1 \leq t$$

- **minimize f(β) under the constraint g(β) ≤ 0**

$$f(\beta) = ||y - X\beta||_2^2 \qquad\qquad g(\beta) = ||\beta||_1 - t$$

Case 1: the unconstrained minimum lies in the feasible region.

Case 2: it does not.



Where is the solution?

Equivalence between the formulations

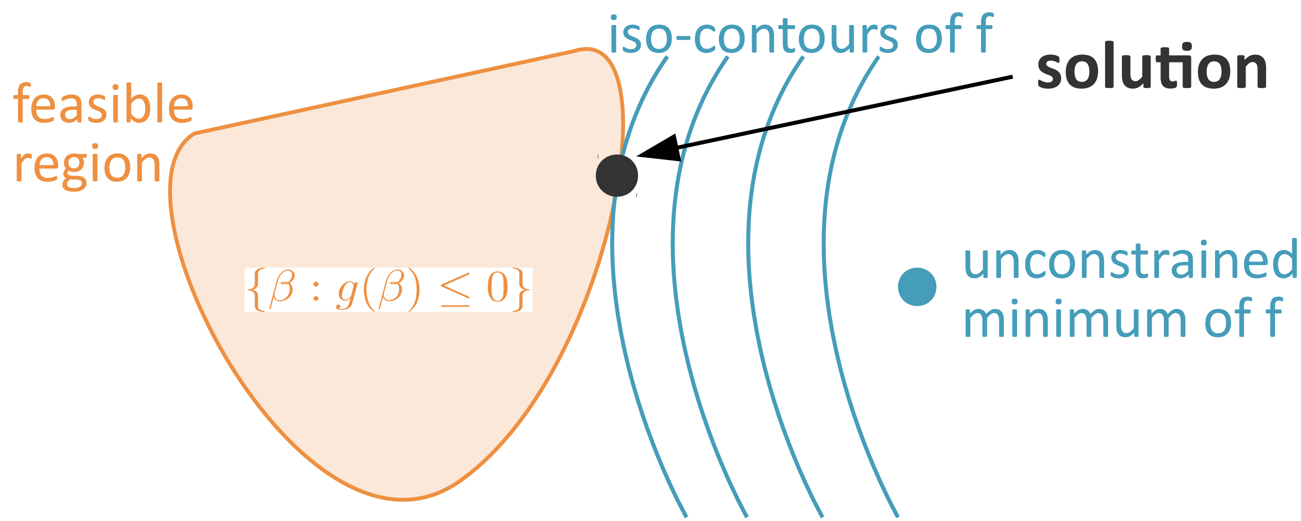
$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} ||y - X\beta||_2^2 \quad \text{s.t. } ||\beta||_1 \leq t$$

- **minimize f(β) under the constraint g(β) ≤ 0**

$$f(\beta) = ||y - X\beta||_2^2 \qquad\qquad g(\beta) = ||\beta||_1 - t$$

Case 1: the unconstrained minimum lies in the feasible region.

Case 2: it does not.



Equivalence between the formulations

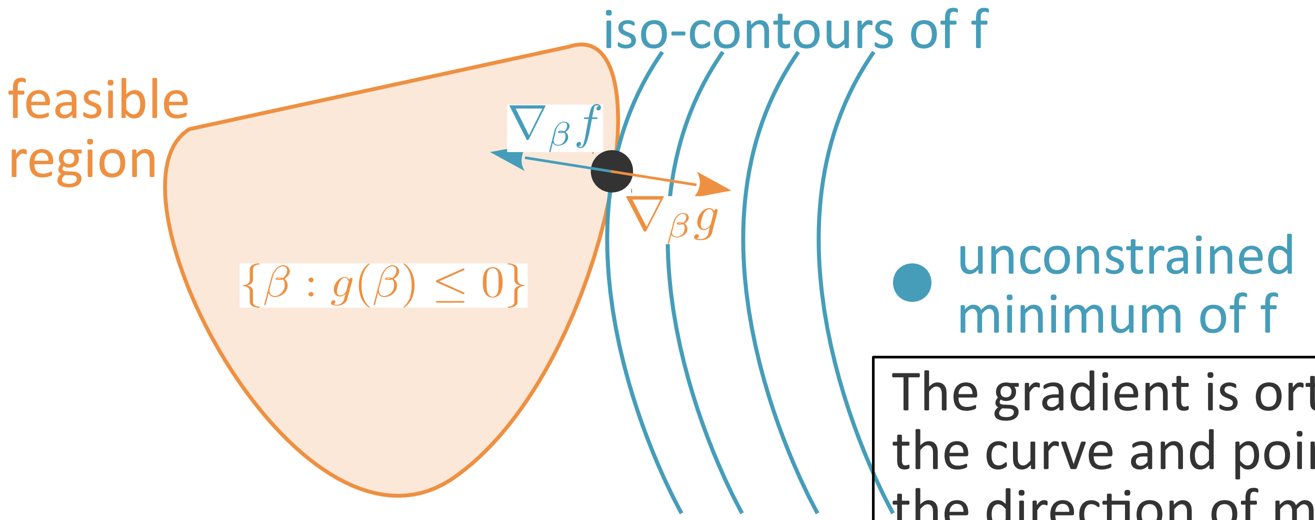
$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} ||y - X\beta||_2^2 \quad \text{s.t.} \quad ||\beta||_1 \leq t$$

- **minimize f(β) under the constraint g(β) ≤ 0**

$$f(\beta) = ||y - X\beta||_2^2 \qquad\qquad g(\beta) = ||\beta||_1 - t$$

Case 1: the unconstrained minimum lies in the feasible region.

Case 2: it does not.



The gradient is orthonormal to the curve and points towards the direction of max increase.

Equivalence between the formulations

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} ||y - X\beta||_2^2 \quad \text{s.t. } ||\beta||_1 \leq t$$

- **minimize f(β) under the constraint g(β) ≤ 0**

$$f(\beta) = ||y - X\beta||_2^2 \qquad\qquad g(\beta) = ||\beta||_1 - t$$

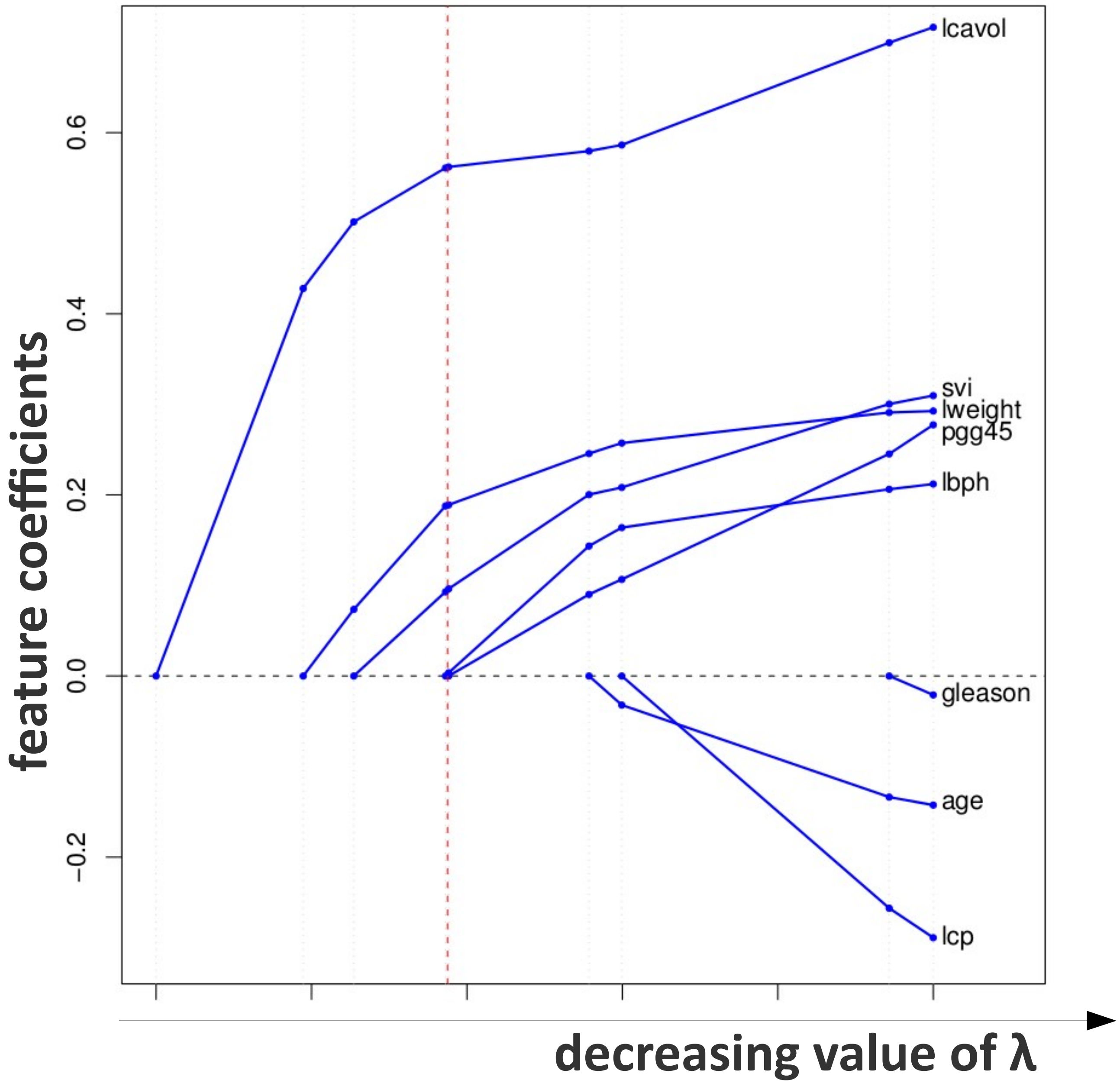
Case 1: the unconstrained minimum lies in the feasible region.

Case 2: it does not. Then it lies at a point where the feasible region and the iso-contours of f are tangent and the gradients are in opposite directions.

The **Lagrangian f(β) + λ g(β) must be minimized (λ ≥ 0)**

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} ||y - X\beta||_2^2 + \lambda||\beta||_1$$

Lasso solution path



Forward stepwise regression

- Build model **sequentially**, adding one variable at a time
  - Start with the intercept
  - At each step, add the variable that **most improves the fit**
  - **Stop when** $||\beta||_1 \leq t$
- Greedy solution

Least Angle Regression

At each step, add "only as much of a variable as needed"

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_1, \beta_2, \ldots, \beta_p = 0$.

Least Angle Regression

At each step, add "only as much of a variable as needed"

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_1, \beta_2, \ldots, \beta_p = 0$.

2. Find the predictor $\mathbf{x}_j$ most correlated with $\mathbf{r}$.

Least Angle Regression

At each step, add "only as much of a variable as needed"

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_1, \beta_2, \ldots, \beta_p = 0$.
2. Find the predictor $\mathbf{x}_j$ most correlated with $\mathbf{r}$.
3. Move β_j from 0 towards its least-squares coefficient $\langle \mathbf{x}_j, \mathbf{r} \rangle$, until some other competitor $\mathbf{x}_k$ has as much correlation with the current residual as does $\mathbf{x}_j$.

$$\boxed{\begin{aligned} \beta_j &\leftarrow \beta_j + \alpha \frac{1}{\sum_{i=1}^{n} (x_j^i)^2} \sum_{i=1}^{n} x_j^i r^i \\ &= \beta_j + \alpha (x_j^\top x_j)^{-1} x_j^\top r \\ &= \beta_j + \alpha \langle x_j^\top, x_j \rangle^{-1} \langle x_j, r \rangle \end{aligned}}$$

$$\boxed{r = (y - \bar{y}) - \beta_j x_j}$$

step size

Least Angle Regression

At each step, add “only as much of a variable as needed”

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_1, \beta_2, \ldots, \beta_p = 0$.
2. Find the predictor $\mathbf{x}_j$ most correlated with $\mathbf{r}$.
3. Move β_j from 0 towards its least-squares coefficient $\langle \mathbf{x}_j, \mathbf{r} \rangle$, until some other competitor $\mathbf{x}_k$ has as much correlation with the current residual as does $\mathbf{x}_j$.
4. Move β_j and β_k in the direction defined by their joint least squares coefficient of the current residual on $(\mathbf{x}_j, \mathbf{x}_k)$, until some other competitor $\mathbf{x}_l$ has as much correlation with the current residual.

$$\boxed{r = (y - \bar{y}) - \beta_j x_j - \beta_k x_k}$$

Least Angle Regression

At each step, add "only as much of a variable as needed"

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_1, \beta_2, \ldots, \beta_p = 0$.
2. Find the predictor $\mathbf{x}_j$ most correlated with $\mathbf{r}$.
3. Move β_j from 0 towards its least-squares coefficient $\langle \mathbf{x}_j, \mathbf{r} \rangle$, until some other competitor $\mathbf{x}_k$ has as much correlation with the current residual as does $\mathbf{x}_j$.
4. Move β_j and β_k in the direction defined by their joint least squares coefficient of the current residual on $(\mathbf{x}_j, \mathbf{x}_k)$, until some other competitor $\mathbf{x}_l$ has as much correlation with the current residual.

4a. If a non-zero coefficient hits zero, drop its variable from the active set of variables and recompute the current joint least squares direction.

Least Angle Regression

At each step, add "only as much of a variable as needed"

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_1, \beta_2, \ldots, \beta_p = 0$.
2. Find the predictor $\mathbf{x}_j$ most correlated with $\mathbf{r}$.
3. Move β_j from 0 towards its least-squares coefficient $\langle \mathbf{x}_j, \mathbf{r} \rangle$, until some other competitor $\mathbf{x}_k$ has as much correlation with the current residual as does $\mathbf{x}_j$.
4. Move β_j and β_k in the direction defined by their joint least squares coefficient of the current residual on $(\mathbf{x}_j, \mathbf{x}_k)$, until some other competitor $\mathbf{x}_l$ has as much correlation with the current residual.

4a. If a non-zero coefficient hits zero, drop its variable from the active set of variables and recompute the current joint least squares direction.

5. Continue in this way until all p predictors have been entered.

Least Angle Regression

At each step, add “only as much of a variable as needed”

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_1, \beta_2, \ldots, \beta_p = 0$.
2. Find the predictor $\mathbf{x}_j$ most correlated with $\mathbf{r}$.
3. Move β_j from 0 towards its least-squares coefficient $\langle \mathbf{x}_j, \mathbf{r} \rangle$, until some other competitor $\mathbf{x}_k$ has as much correlation with the current residual as does $\mathbf{x}_j$.
4. Move β_j and β_k in the direction defined by their joint least squares coefficient of the current residual on $(\mathbf{x}_j, \mathbf{x}_k)$, until some other competitor $\mathbf{x}_l$ has as much correlation with the current residual.

4a. If a non-zero coefficient hits zero, drop its variable from the active set of variables and recompute the current joint least squares direction.

5. Continue in this way until all p predictors have been entered.

Maximum number of steps: max(n-1, p)

Elastic Net

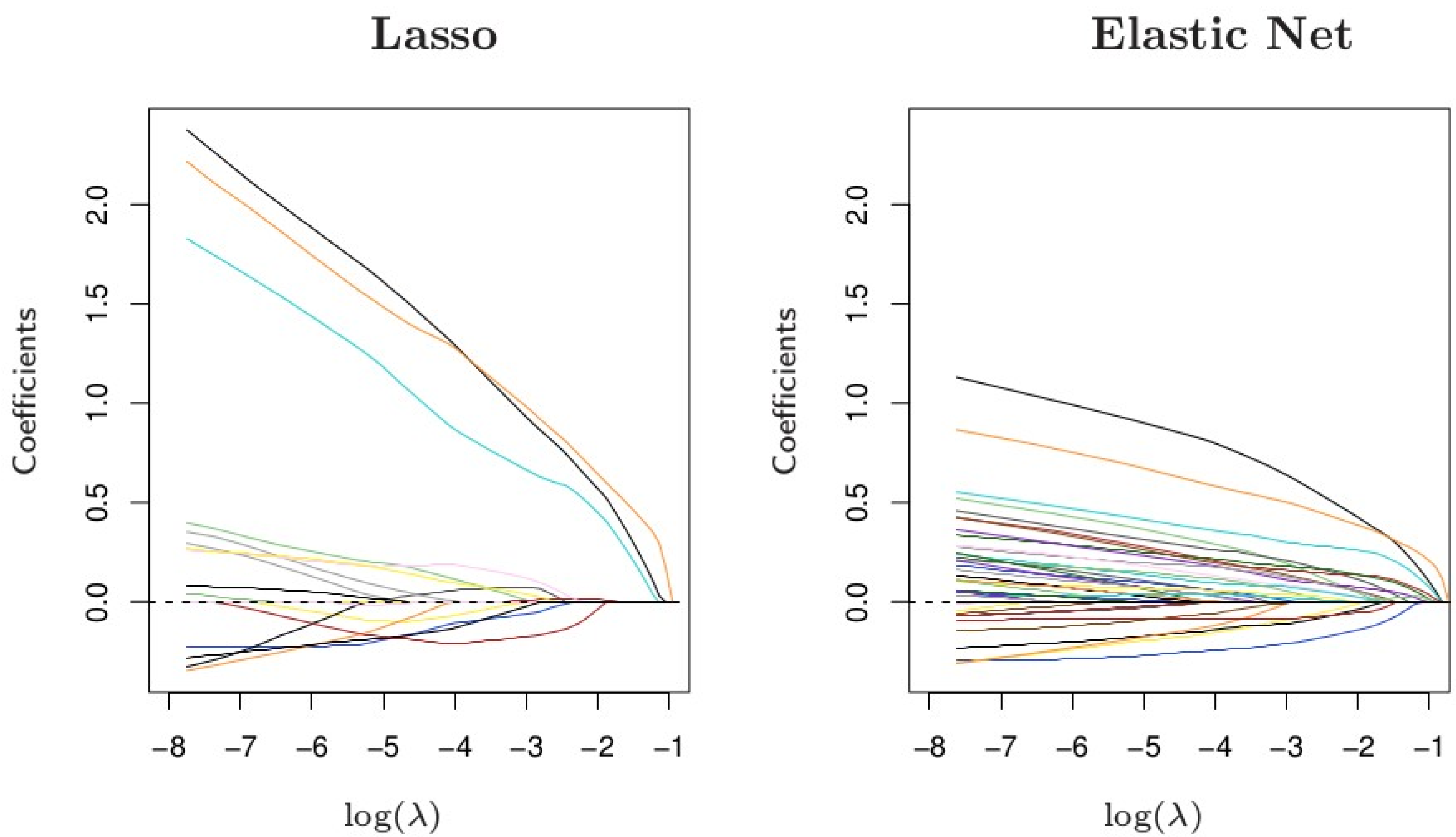
Elastic Net

- **Combine lasso** and **ridge regression**

$$\hat{\beta}_{\text{enet}} = \arg\min_{\beta} ||y - X\beta||_2^2 + \lambda \left(\alpha ||\beta||_2^2 + (1-\alpha)||\beta||_1 \right)$$

  - **Select variables** like the lasso.
  - **Shrinks together coefficients of correlated variables** like the ridge regression.

E.g. Leukemia data



Elastic Net results in more non-zero coefficients than Lasso, but with smaller amplitudes.

Lq-norm regularization

Lq-norm regularization

$$\hat{\beta} = \arg\min_{\beta} ||Y - X\beta||_2^2 + \lambda ||\beta||_q^q \qquad ||\beta||_q = \left(\sum_{j=1}^{p} |\beta_j|^q \right)^{1/q}$$

Equivalently:

$$\hat{\beta} = \arg\min_{\beta} ||Y - X\beta||_2^2 \text{ s. t. } ||\beta||_q^q \leq s$$

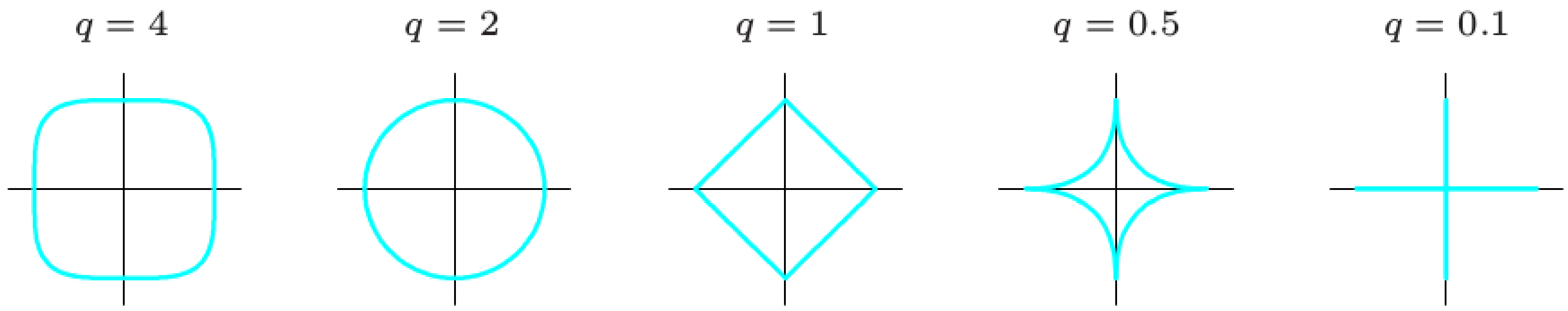
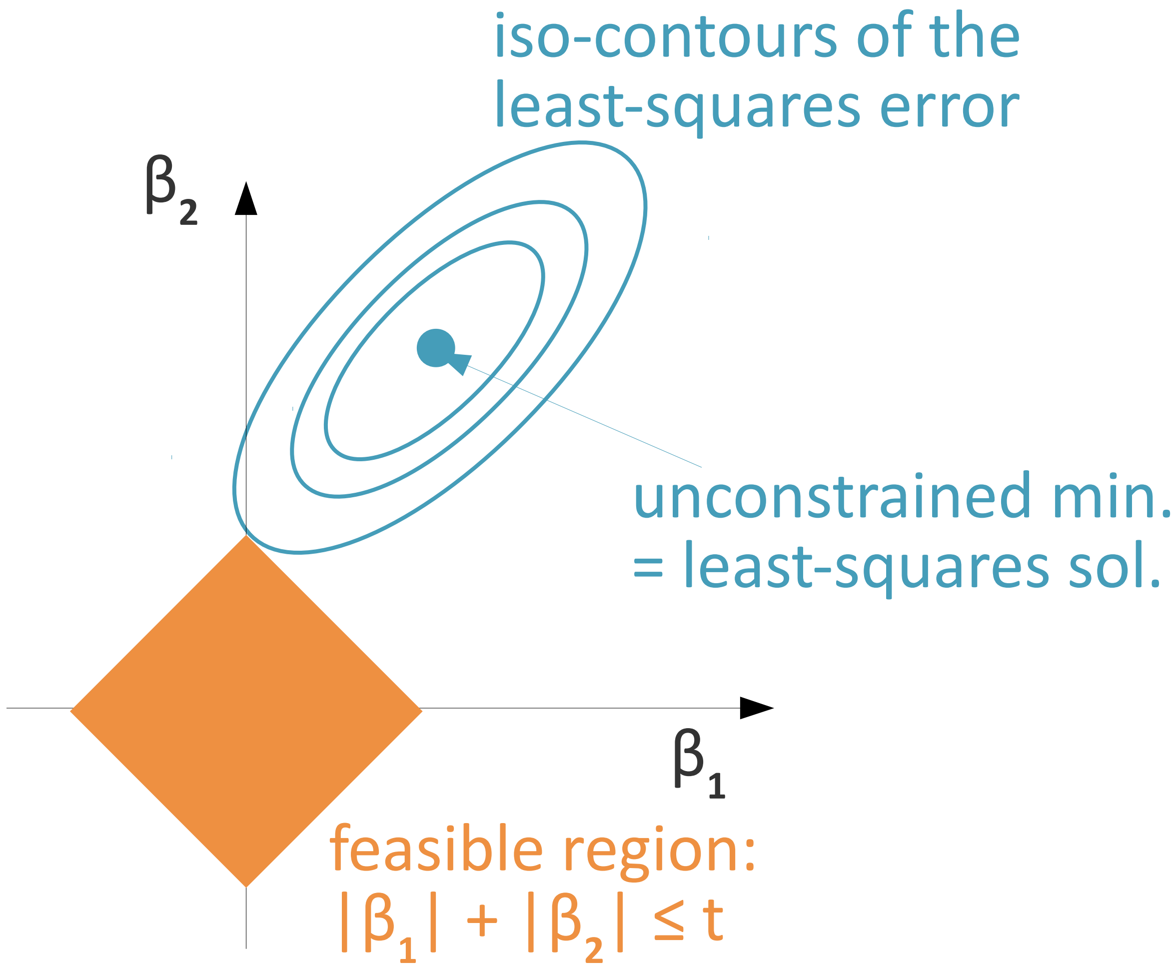


FIGURE 3.12. *Contours of constant value of* $\sum_j |\beta_j|^q$ *for given values of* q.

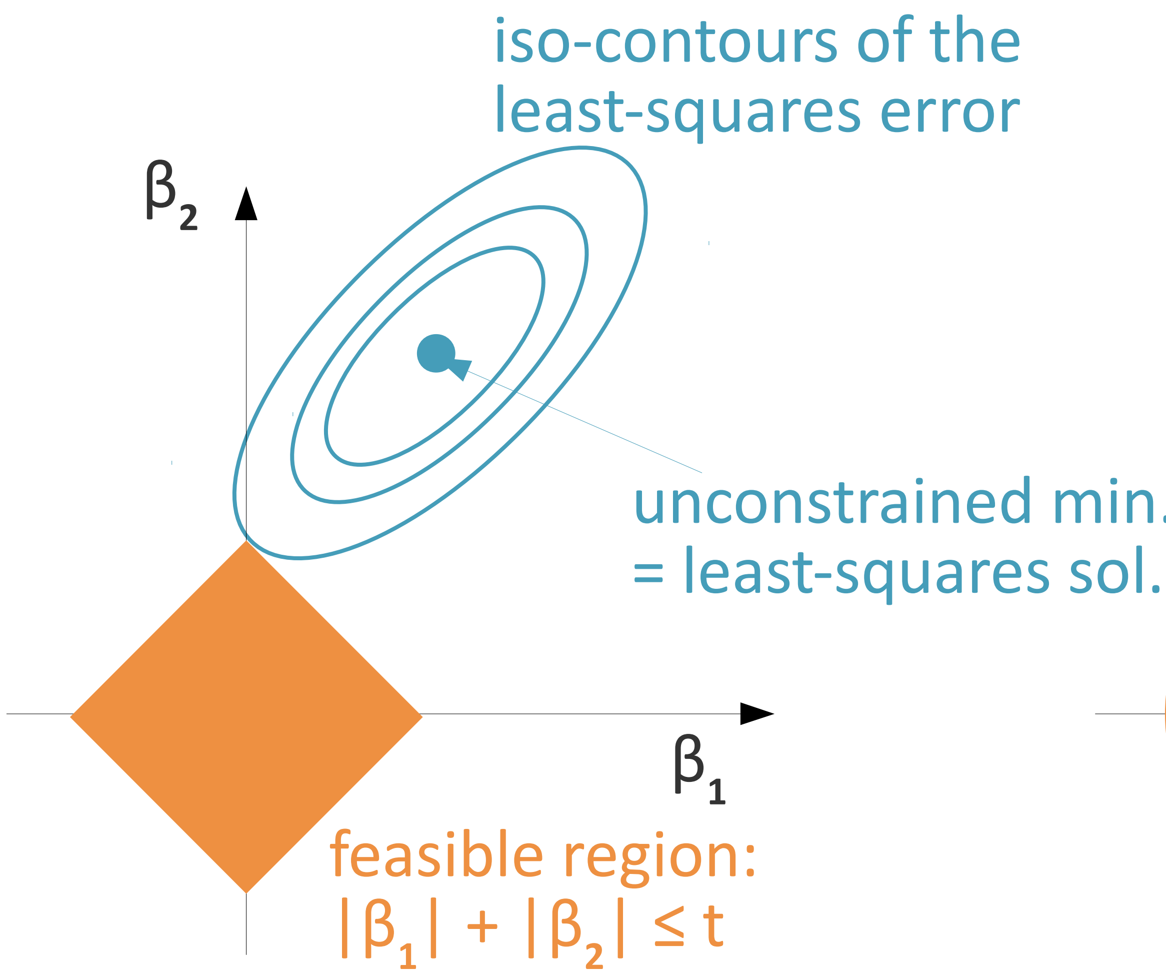
Lasso vs. ridge

L1 norm

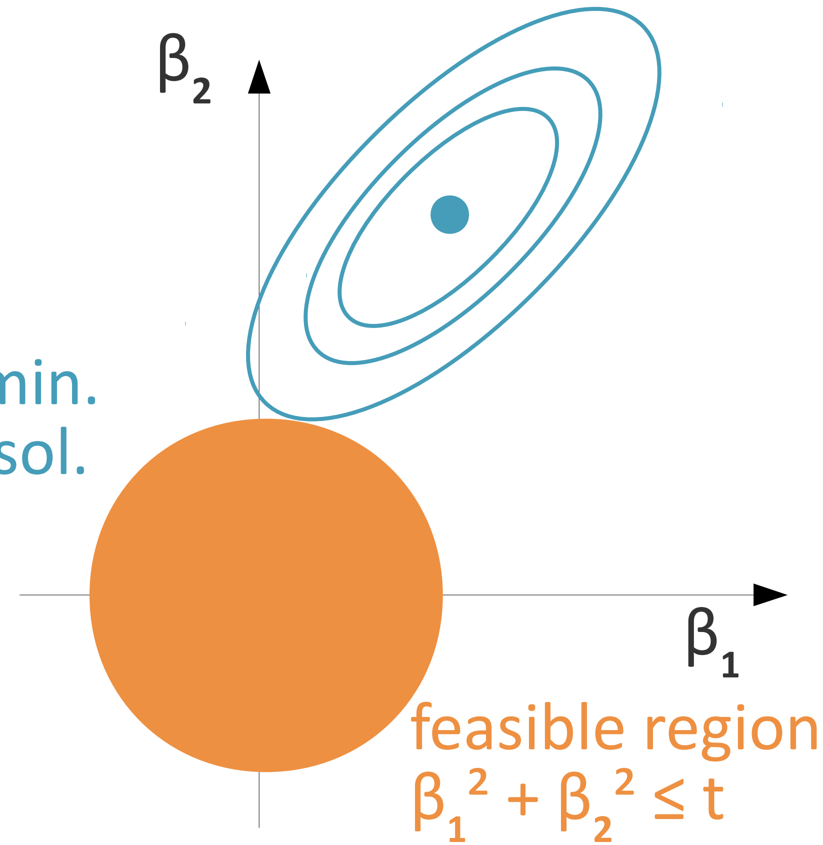


Lasso vs. ridge

L1 norm



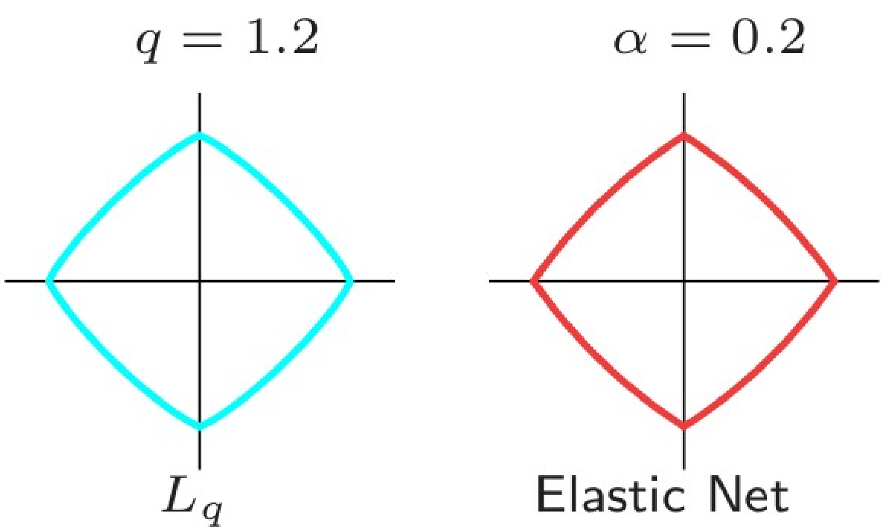
L2 norm



Elastic net

- **Elastic penalty**

$$\hat{\beta} = \arg\min_{\beta} ||y - X\beta||_2^2 + \lambda \left(\alpha ||\beta||_2^2 + (1-\alpha)||\beta||_1 \right)$$



Structured regularization

Group lasso

Use K predefined groups of variables that are known to "work"together and expected to be either all active or all inactive together.

E.g.

- genes belonging to the same biological pathway.

$$\hat{\beta} = \arg\min_{\beta} ||y - \sum_{k=1}^{K} X_k \beta_k||_2^2 + \lambda \sum_{k=1}^{K} \sqrt{p_k}\, ||\beta_k||_2$$

Size of group k

Features belonging to group k

Other examples of structured penalties

- **Overlapping groups**

  Jacob et al. (2009). Group lasso with overlap and graph lasso. *ICML.*

- **Graphs**

  Li & Li (2010). Variable selection and regression analysis for graph-structured covariates with an application to genomics. *Ann. App. Stats.*

- **Trees**

  Zhao et al. (2006). Grouped and hierarchical model selection through composite absolute penalties. *Ann. Stat.*

- **Multiple related tasks**

  Obozinski et al. (2006). Multitask feature selection. *Technical Report, UC Berkeley.*

Minimize SSE + λ x regularizer

- **Ridge**
  - gives similar weights to similar variables
  - not very sparse
  - analytical solution
- **Lasso**
  - randomly picks one of several correlated variables
  - sparse
  - LAR algorithm
- **Elastic net**
  - selects variables like the lasso
  - shrinks together the coefficients of correlated variables.
- **Many other regularizers** are possible

  Lp norms, groups, graphs, trees...