FSAN/ELEG815: Statistical Learning

Gonzalo R. Arce

Department of Electrical and Computer Engineering
University of Delaware

X:Lasso Regression

Outline of the Course

1. Review of Probability
2. Stationary processes
3. Eigen Analysis, Singular Value Decomposition (SVD) and Principal Component Analysis (PCA)
4. The Learning Problem and the VC Dimension
5. Training vs Testing
6. The Wiener Filter
7. Adaptive Optimization: Steepest descent and the LMS algorithm
8. Nonlinear Transformation and Logistic Regression
9. Overfitting and Regularization (Ridge Regression)
10. Lasso Regression
11. Neural Networks
12. Matrix Completion

The ℓ_1 Norm and Sparsity

- The ℓ_0 norm is defined by: $\|x\|_0 = \#\{i : x(i) \neq 0\}$
  Sparsity of x is measured by its number of non-zero elements.

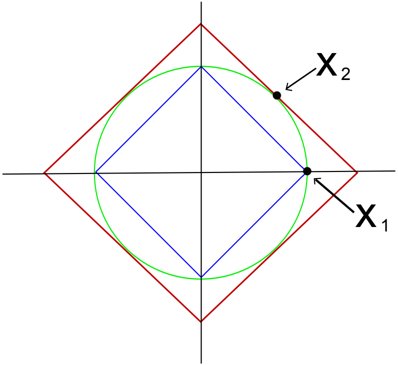
- The ℓ_1 norm is defined by: $\|x\|_1 = \sum_i |x(i)|$
  ℓ_1 norm has two key properties:
  - Robust data fitting
  - Sparsity inducing norm

- The ℓ_2 norm is defined by: $\|x\|_2 = (\sum_i |x(i)|^2)^{1/2}$
  ℓ_2 norm is not effective in measuring *sparsity* of x

Why ℓ_1 Norm Promotes Sparsity?

Given two N-dimensional signals:

- $x_1 = (1, 0, ..., 0) \rightarrow$ "Spike" signal
- $x_2 = (1/\sqrt{N}, 1/\sqrt{N}, ..., 1/\sqrt{N}) \rightarrow$ "Comb" signal



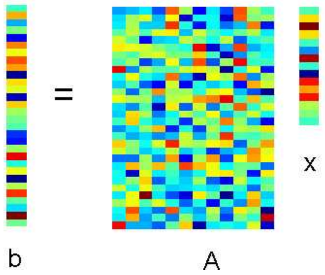
- x_1 and x_2 have the same ℓ_2 norm: $\|x_1\|_2 = 1$ and $\|x_2\|_2 = 1$.
- However, $\|x_1\|_1 = 1$ and $\|x_2\|_1 = \sqrt{N}$.

ℓ_1 Norm in Regression

- Linear regression is widely used in science and engineering.

$$\text{Given} \quad A \in R^{m\times n} \quad \text{and} \quad b \in R^m; \;\; m > n$$

$$\text{Find} \quad x \;\; \text{s.t.} \;\; b = Ax \quad \text{(overdetermined)}$$



ℓ_1 Norm Regression

Two approaches:

- Minimize the ℓ_2 norm of the residuals

$$\min_{x \in R^n} \|b - Ax\|_2$$

  The ℓ_2 norm penalizes large residuals

- Minimizes the ℓ_1 norm of the residuals

$$\min_{x \in R^n} \|b - Ax\|_1$$

  The ℓ_1 norm puts much more weight on small residuals

Matlab Code

- $\min_{x \in R^n} \|Ax - b\|_2$

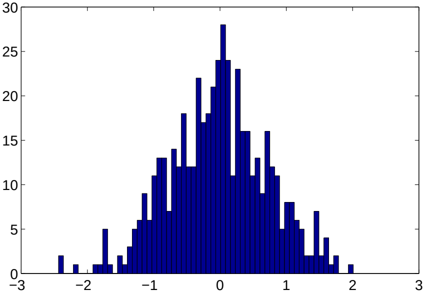
A=randn(500,150);
b=randn(500,1);
$x = (A' * A)^{(-1)} * A' * b$; Least Squares Solution

- $\min_{x \in R^n} \|Ax - b\|_1$

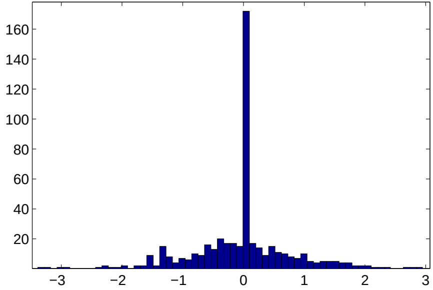
A=randn(500,150);
b=randn(500,1);
*X = medrec(b,A,max(A'*b),0,100,1e-5);*

ℓ_1 Norm Regression

$m = 500$, $n = 150$. $A = randn(m, n)$ and $b = randn(m, 1)$



ℓ_2 Residuals

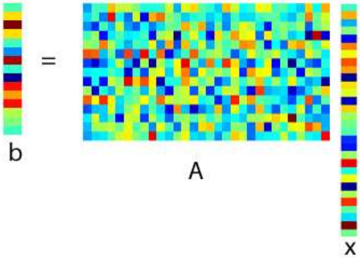


ℓ_1 Residuals

ℓ_1 Norm in Regression

Given $A \in R^{m \times n}$ and $b \in R^m$; $m < n$

Find x s.t. $b = Ax$ (underdetermined)



ℓ_1 Norm Regression

Two approaches:

- Minimize the ℓ_2 norm of x

$$\min_{x \in R^n} \|x\|_2 \quad \text{subject to} \quad Ax = b$$

- Minimize the ℓ_1 norm of x

$$\min_{x \in R^n} \|x\|_1 \quad \text{subject to} \quad Ax = b$$

Matlab Code

- $\min_{x \in R^n} \|x\|_2 \quad$ subject to $\quad Ax = b$

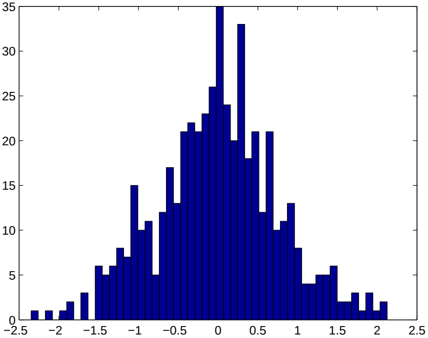
```
A=randn(150,500);
b=randn(150,1);
C=eye(150,500);
d=zeros(150,1);
X=lsqlin(C,d,[],[],A,b);
```

- In general:
  $\min_{x \in R^n} f(x) \quad$ subject to $\quad Ax = b$

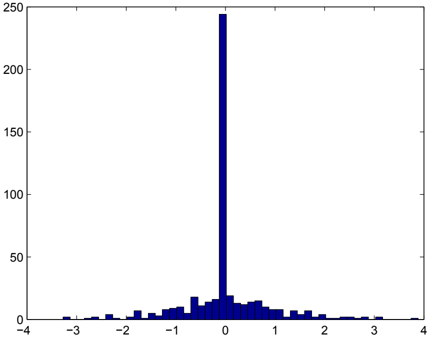
```
X= fmincon(@(x) f(x),zeros(500,1),[],[],A,b,[],[],options);
```

where $f(x)$ is a convex function.

ℓ_1 Norm Regression

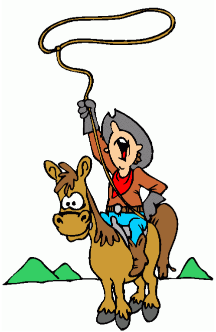


ℓ_2 Solution



ℓ_1 Solution

Least Absolute Shrinkage and Selection Operator (LASSO)



- LASSO combines shrinking of Ridge regression with variable selection. Tibshirani 1996.
- Difference between LASSO and Ridge regression is the penalty used

$$\hat{\mathbf{w}}^{ridge} = \arg\min_{\mathbf{w}\in\mathbb{R}^d}\left[\sum_{i=1}^{N}(y_i - w_0 - \sum_{j=1}^{d} x_{ij}w_j)^2 + \lambda\sum_{j=1}^{d} w_j^2\right]$$

$$\hat{\mathbf{w}}^{lasso} = \arg\min_{\mathbf{w}}\left[\sum_{i=1}^{N}(y_i - \sum_{j=1}^{d} x_{ij}w_j)^2 + \lambda\sum_{j=1}^{d} |w_j|\right]$$

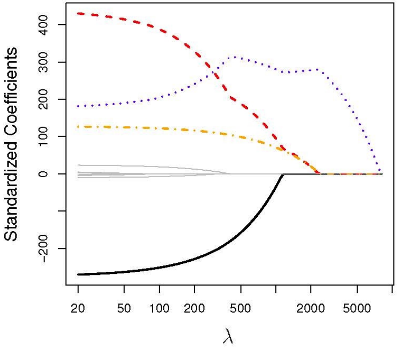
Least Absolute Shrinkage and Selection Operator (LASSO)

- LASSO coefficients are the solutions to the ℓ_1 optimization problem defined as

$$\begin{aligned}\hat{\mathbf{w}}^{lasso} &= \arg\min_{\mathbf{w}} \left[\sum_{i=1}^{N} (y_i - \sum_{j=1}^{d} x_{ij} w_j)^2 + \lambda \sum_{j=1}^{d} |w_j| \right] \\ &= \arg\min_{\mathbf{w}} \left[\sum_{i=1}^{N} (y_i - \mathbf{x}_i^T \mathbf{w})^2 + \lambda \sum_{j=1}^{d} |w_j| \right] \\ &= \arg\min_{\mathbf{w}} \left[(\mathbf{y} - \mathbf{X}\mathbf{w})^T (\mathbf{y} - \mathbf{X}\mathbf{w}) + \lambda ||w_j||_1 \right].\end{aligned}$$

- LASSO also shrinks the coefficients.
- ℓ_1 norm forces coefficients to zero when λ is large: **variable selection**.
- Lasso yields **sparse** models, keeping subset of variables.
- Unlike ridge regression, $\hat{\mathbf{w}}_{\lambda}^{lasso}$ has no closed form.

Lasso Regression Example Credit Data set



- Lasso performs better when a small number of predictors have strong coefficients, and the remaining predictors are small.
- Ridge regression performs better when the response is a function of many predictors.

The Variable Selection Property of the Lasso

One can show that the Ridge and Lasso regression coefficient estimates solve the following problems

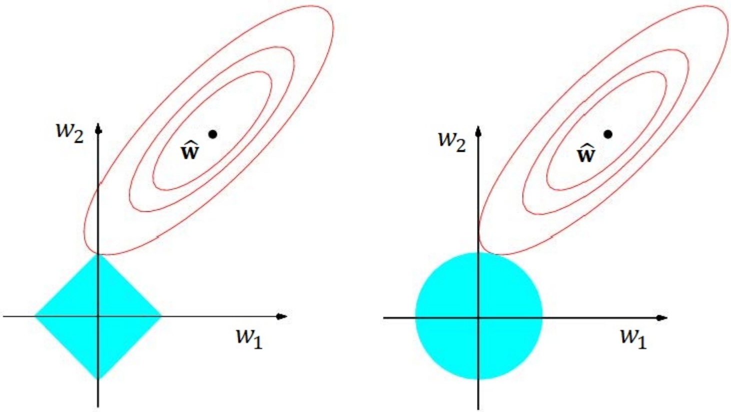
$$\hat{\mathbf{w}}^{ridge} = \underset{\mathbf{w}}{argmin}\{\sum_{i=1}^{N}(y_i - w_0 - \sum_{j=1}^{d} x_{ij}w_j)^2\} \quad (1)$$

subject to $\sum_{j=1}^{d} w_j^2 \leq t$

$$\hat{\mathbf{w}}^{lasso} = \underset{\mathbf{w}}{argmin}\{\sum_{i=1}^{N}(y_i - w_0 - \sum_{j=1}^{d} x_{ij}w_j)^2\} \quad (2)$$

subject to $\sum_{j=1}^{d} |w_j| \leq t$

The Variable Selection Property of the Lasso



- RSS has elliptical contours, centered at the LS estimate.
- Constraint regions, $w_1^2 + w_2^2 \leq t$, and $|w_1| + |w_2| \leq t$.

Comparing the Lasso and Ridge Regression

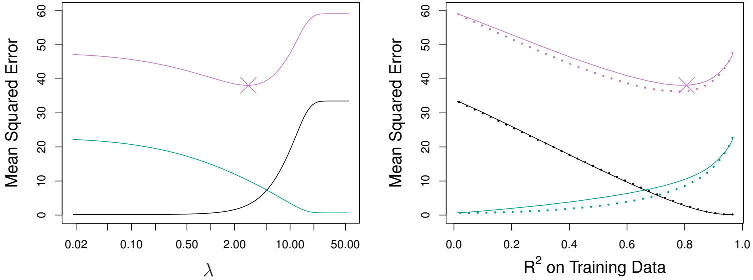
The criteria to be analyzed for each case:

- Bias: Error that is introduced by approximating a real-life problem, by a much simpler model.
- Variance: Amount by which y would change is we estimated it using a different training data set.
- Training MSE: Mean squared error computed using the training data.
- Test MSE: Mean squared error computed using the test data.

R-squared is a statistical measure of how close the data are to the fitted regression line. The better the linear regression fits the data in comparison to the simple average, the closer the value of R^2 is to 1.

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \overline{y}_i)^2} = 1 - \frac{\text{Residual sum of squares}}{\text{Total sum of squares}}. \tag{3}$$

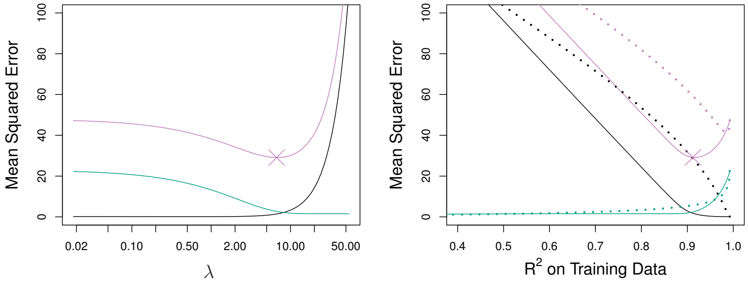
Comparing the Lasso and Ridge Regression



Simulated data set containing $d = 45$ predictors and $n = 50$ observations. For this figure all predictors were related to the response.

- Left: Plots of squared bias (black), variance (green), and test MSE (purple) for the lasso.
- Right: Comparison of squared bias, variance and test MSE between lasso (solid) and ridge (dashed).

Comparing the Lasso and Ridge Regression

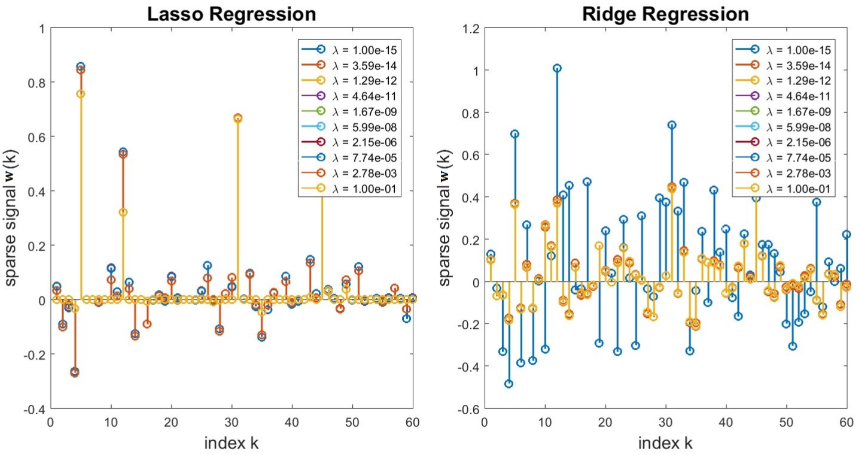


Here the the response is a function of only 2 out of 45 predictors.

- Left: Squared bias (black), variance (green), and test MSE (purple) for the lasso.
- Right: Comparison of squared bias, variance and test MSE between lasso (solid) and ridge (dashed).

Lasso vs Ridge regression

- $\mathbf{y} = \mathbf{X}\mathbf{w} + \boldsymbol{\epsilon}$, where $\mathbf{X} \in \mathbb{R}^{40\times 60}$ is random Gaussian and $\boldsymbol{\epsilon}$ is noise.
- Original sparse signal is
$w(k) = \delta(k-5) + 0.5\delta(k-12) + 0.9\delta(k-31) - 0.75\delta(k-45)$



Iterative Calculation

- LASSO does not have a close form solution. Solved iteratively.
- Define $F(\mathbf{w}) = ||\mathbf{y} - \mathbf{X}\mathbf{w}||_2^2 + \lambda||\mathbf{w}||_1$.
- The solution to the LASSO problem is denoted as $\mathbf{w}_S$.
- Define an iterative procedure adding the non-negative term, having zero value at $\mathbf{w}_S$, $G(\mathbf{w}) = (\mathbf{w} - \mathbf{w}_S)^T(\alpha\mathbf{I} - \mathbf{X}^T\mathbf{X})(\mathbf{w} - \mathbf{w}_S)$, to the function $F(\mathbf{w})$.

Iterative Calculation

The cost function is:

$$H(\mathbf{w}) = F(\mathbf{w}) + (\mathbf{w} - \mathbf{w}_S)^T(\alpha\mathbf{I} - \mathbf{X}^T\mathbf{X})(\mathbf{w} - \mathbf{w}_S), \tag{4}$$

where α is such that the added term is always nonnegative. It means $\alpha > \lambda_{max}$, where λ_{max} is the largest eigenvalue of $\mathbf{X}^T\mathbf{X}$.

$$\begin{aligned} H(\mathbf{w}) &= F(\mathbf{w}) + G(\mathbf{w}) \\ &= ||\mathbf{y} - \mathbf{X}\mathbf{w}||_2^2 + \lambda||\mathbf{w}||_1 + (\mathbf{w} - \mathbf{w}_S)^T(\alpha\mathbf{I} - \mathbf{X}^T\mathbf{X})(\mathbf{w} - \mathbf{w}_S) \end{aligned}$$

Since $||\mathbf{w}||_1 = \mathbf{w}^T \text{sign}\{\mathbf{w}\}$

$$\begin{aligned} H(\mathbf{w}) = \; & ||\mathbf{y}||_2^2 - \mathbf{w}^T\mathbf{X}^T\mathbf{y} - \mathbf{y}^T\mathbf{X}\mathbf{w} + \mathbf{w}^T\mathbf{X}^T\mathbf{X}\mathbf{w} + \lambda\mathbf{w}^T \text{sign}\{\mathbf{w}\} \\ & + (\mathbf{w} - \mathbf{w}_S)^T(\alpha\mathbf{I} - \mathbf{X}^T\mathbf{X})(\mathbf{w} - \mathbf{w}_S) \end{aligned}$$

Iterative Calculation

$$\begin{aligned} H(\mathbf{w}) &= ||\mathbf{y}||_2^2 - \mathbf{w}^T\mathbf{X}^T\mathbf{y} - \mathbf{y}^T\mathbf{X}\mathbf{w} + \mathbf{w}^T\mathbf{X}^T\mathbf{X}\mathbf{w} + \lambda\mathbf{w}^T\text{sign}\{\mathbf{w}\} \\ &+ (\mathbf{w}-\mathbf{w}_S)^T(\alpha\mathbf{I} - \mathbf{X}^T\mathbf{X})(\mathbf{w}-\mathbf{w}_S) \end{aligned}$$

Equating the gradient of $H(\mathbf{w})$ to zero:

$$\begin{aligned} \frac{\partial H(\mathbf{w})}{\partial \mathbf{w}^T} &= -2\mathbf{X}^T\mathbf{y} + 2\mathbf{X}^T\mathbf{X}\mathbf{w} + \lambda\text{sign}\{\mathbf{w}\} + 2(\alpha\mathbf{I} - \mathbf{X}^T\mathbf{X})(\mathbf{w}-\mathbf{w}_S) \\ 0 &= -\mathbf{X}^T\mathbf{y} + \mathbf{X}^T\mathbf{X}\mathbf{w} + \frac{\lambda}{2}\text{sign}\{\mathbf{w}\} + \alpha\mathbf{w} - \mathbf{X}^T\mathbf{X}\mathbf{w} - (\alpha\mathbf{I} - \mathbf{X}^T\mathbf{X})\mathbf{w}_S \\ 0 &= -\mathbf{X}^T\mathbf{y} + \frac{\lambda}{2}\text{sign}\{\mathbf{w}\} + \alpha\mathbf{w} - (\alpha\mathbf{I} - \mathbf{X}^T\mathbf{X})\mathbf{w}_S \end{aligned}$$

Rearranging the terms,

$$\mathbf{w} + \frac{\lambda}{2\alpha}\text{sign}\{\mathbf{w}\} = \frac{1}{\alpha}\mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{w}_S) + \mathbf{w}_S$$

Iterative Calculation

Corresponding iterative update

$$\mathbf{w}_{s+1} + \frac{\lambda}{2\alpha}\text{sign}\left\{\mathbf{w}_{s+1}\right\} = \frac{1}{\alpha}\mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{w}_s) + \mathbf{w}_s \tag{5}$$

How to solve it?

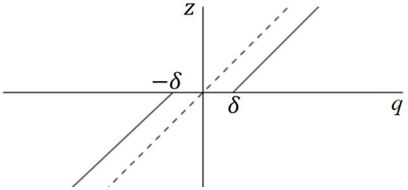
Note

The solution of the scalar equation $z+\delta\text{sign}(z)=q$, is obtained using soft-thresholding rule defined by a function $\text{soft}(q,\delta)$ as:

$$z=\text{soft}(q,\delta)=\begin{cases} q+\delta & \text{for} \quad q<-\delta \\ 0 & \text{for} \quad |q|\leq\delta \\ q-\delta & \text{for} \quad q>\delta \end{cases}$$

or

$$\text{soft}(q,\delta)=\text{sign}(q)\max\{0,|q|-\delta\}$$



Iterative Calculation

- The solution of $z+\delta\text{sign}(z)=q$ is $z=\text{soft}(q,\delta)$
- $\underbrace{\mathbf{w}_{s+1}}_{z}+\underbrace{\frac{\lambda}{2\alpha}}_{\delta}\text{sign}\left\{\underbrace{\mathbf{w}_{s+1}}_{z}\right\}=\underbrace{\frac{1}{\alpha}\mathbf{X}^T(\mathbf{y}-\mathbf{X}\mathbf{w}_s)+\mathbf{w}_s}_{q}$

Thus,

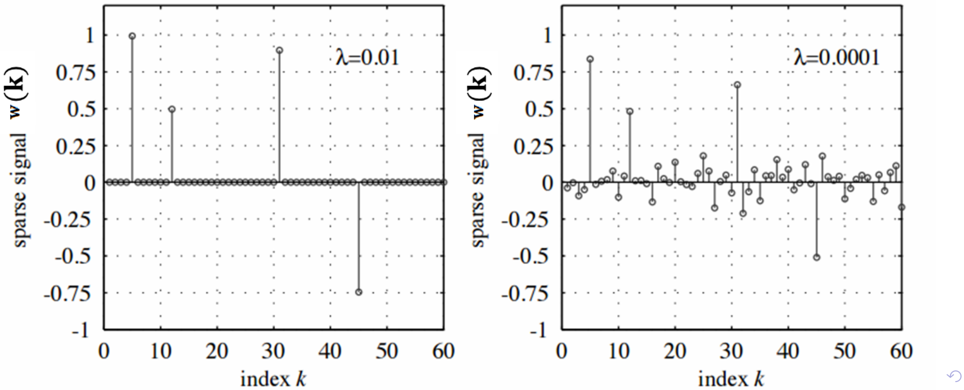
$$\mathbf{w}_{s+1}=\text{soft}\left(\frac{1}{\alpha}\mathbf{X}^T(\mathbf{y}-\mathbf{X}\mathbf{w}_s)+\mathbf{w}_s,\frac{\lambda}{2\alpha}\right) \quad (6)$$

This is the iterative soft-thresholding algorithm (ISTA) for LASSO minimization.

Example

$\mathbf{y} = \mathbf{X}\mathbf{w}$, where

- $\mathbf{X}$ is a random Gaussian matrix $\in \mathbb{R}^{40\times 60}$.
- Original sparse signal is $w(k) = \delta(k-5) + 0.5\delta(k-12) + 0.9\delta(k-31) - 0.75\delta(k-45)$.
- The results for $\lambda = 0.01$ and $\lambda = 0.0001$ are presented



Coordinate Descent Optimization

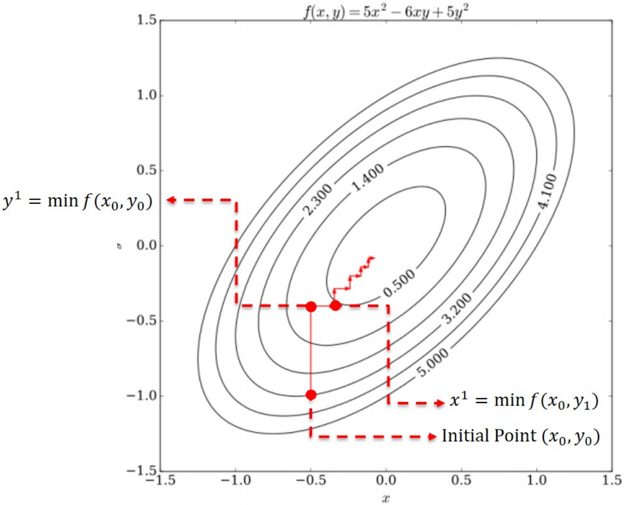
Objective: Minimize a function $f : \mathbb{R}^n \to \mathbb{R}$. **Strategy**: Minimize each coordinate separately while cycling through the coordinates.

$$
\begin{aligned}
x_1^{(k+1)} &= \min_x f(x, x_2^{(k)}, x_3^{(k)}, \cdots, x_p^{(k)}) \\
x_2^{(k+1)} &= \min_x f(x_1^{(k+1)}, x, x_3^{(k)}, \cdots, x_p^{(k)}) \\
x_3^{(k+1)} &= \min_x f(x_1^{(k+1)}, x_2^{(k+1)}, x, x_4^{(k)}, \cdots, x_p^{(k)}) \\
&\vdots \\
x_p^{(k+1)} &= \min_x f(x_1^{(k+1)}, x_2^{(k+1)}, \cdots, x_{p-1}^{(k+1)}, x).
\end{aligned}
$$

Neglected technique in the past that gained popularity recently. Can be very efficient when the coordinate-wise problems are easy to solve (e.g. if they admit a closed-form solution).

Coordinate Descent Optimization

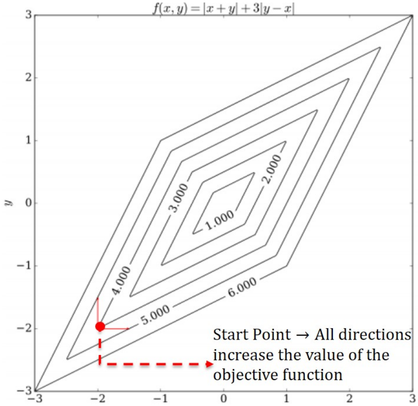
In each iteration a line search is done to find the next step



Source: Wikipedia (Nicoguaro).

Convergence

- This procedure **Does not** always converge to an extreme point of the objective function.
- Example: Coordinate descend iteration gets stuck at a non-stationary point since the level curves are not smooth.



Source: Wikipedia (Nicoguaro).

Coordinate Descent for the LASSO

- Recall the LASSO objective function:

$$f(\mathbf{w}) = \underbrace{\sum_{i=1}^{N}(y_i - \sum_{j=1}^{d} X_{ij}w_j)^2}_{RSS(\mathbf{w})} + \underbrace{\lambda\sum_{j=1}^{d}|w_j|}_{\lambda||\mathbf{w}||_1} \quad (7)$$

- Fix all coordinates w_{-j} and take partial derivative with respect to w_j.

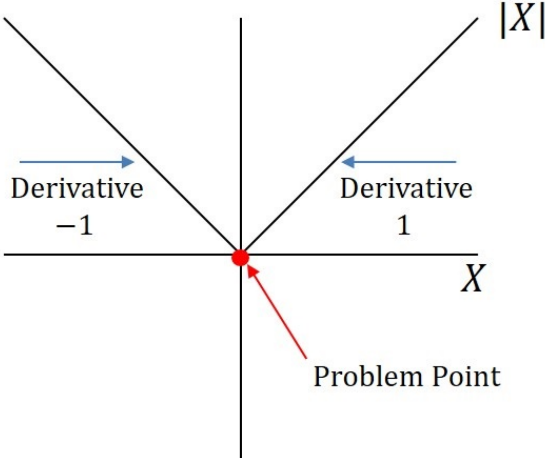
$$\frac{\partial f(\mathbf{w})}{\partial w_j} = \frac{\partial RSS(\mathbf{w})}{\partial w_j} + \frac{\partial \lambda||\mathbf{w}||_1}{\partial w_j}$$

$$\begin{aligned}\frac{\partial RSS(\mathbf{w})}{\partial w_j} &= -2\sum_{i=1}^{N} X_{ij}(y_i - \sum_{j=0}^{d} X_{ij}w_j) \\ &= -2\sum_{i=1}^{N} X_{ij}(y_i - \sum_{k\neq j} X_{ik}w_k - X_{ij}w_j) \\ &= \underbrace{-2\sum_{i=1}^{N} X_{ij}(y_i - \sum_{k\neq j} X_{ik}w_k)}_{\rho_j} + 2w_j \underbrace{\sum_{i=1}^{N} X_{ij}^2}_{z_j}\end{aligned}$$

Coordinate Descent for the LASSO

Compute the partial derivative of the second term with respect to w_j of $\lambda||\mathbf{w}||_1$.

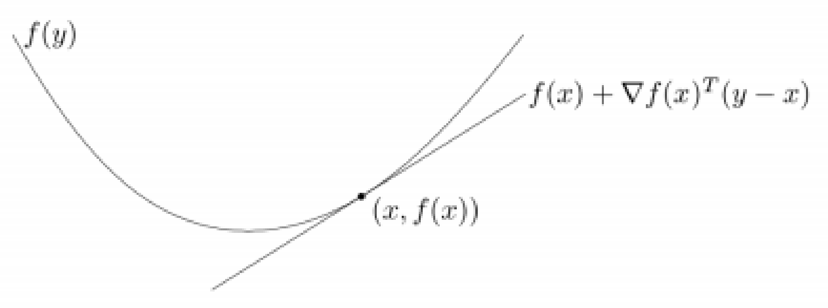
$$\lambda \frac{\partial}{\partial w_j} |w_j| = ???$$



Subgradients of Convex Functions

Suppose f is convex and differentiable. Then

$$f(y) \geq f(x) + \nabla f(x)^T (y - x)$$

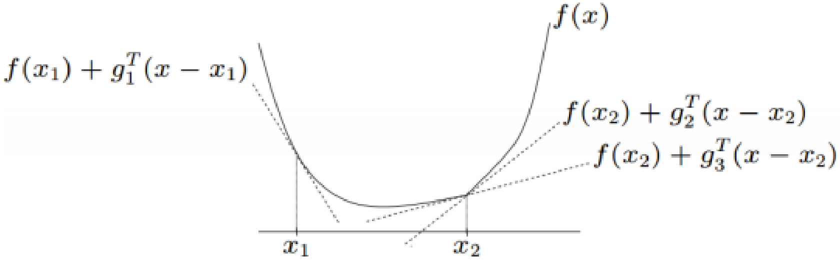


Boyd & Vandenberghe, Figure 3.2.

Subgradients of Convex Functions

We say that g is a **subgradient** of f at x if

$$f(y) \geq f(x) + g^T(y - x) \qquad \forall y$$



Boyd, lecture notes.

Subgradients of Convex Functions

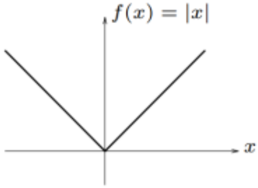
We define:

$$\partial f(x) := \text{all subgradients of } f \text{ at } x$$

- $\partial f(x)$ is a closed convex set (can be empty).
- $\partial f(x) = \nabla f(x)$ if f is differentiable at x.
- If $\partial f(x) = g$, then f is differentiable at x and $\nabla f(x) = g$

Basic Properties:

- $\partial(\alpha f) = \alpha \partial f$ if $\alpha > 0$.
- $\partial(f_1 + f_2) = \partial f_1 + \partial f_2$



$$\partial f(x) = \begin{cases} \{-1\} & \text{if } x < 0 \\ [-1, 1] & \text{if } x = 0 \\ \{1\} & \text{if } x > 0 \end{cases}.$$

Subgradient of L_1 Term

Using the subgradient theory to compute the partial derivative of the second term with respect to w_j of $\lambda||\mathbf{w}||_1$:

$$\lambda\frac{\partial}{\partial w_j}|w_j| = \begin{cases} -\lambda & if \quad w_j < 0 \\ [-\lambda, \lambda] & if \quad w_j = 0 \\ \lambda & if \quad w_j > 0 \end{cases}$$

LASSO-Coordinate Descent

Putting it all together

$$\begin{aligned}
\lambda\frac{\partial f(x)}{\partial w_j} &= \frac{\partial RSS(\mathbf{w})}{\partial w_j}+\frac{\partial \lambda||\mathbf{w}||_1}{\partial w_j}\\
&= 2z_jw_j-2\rho_j+\frac{\partial \lambda||\mathbf{w}||_1}{\partial w_j}\\
&= \begin{cases} 2z_jw_j-2\rho_j-\lambda & if \quad w_j<0\\ [2z_jw_j-2\rho_j-\lambda, 2z_jw_j-2\rho_j+\lambda] & if \quad w_j=0\\ 2z_jw_j-2\rho_j+\lambda & if \quad w_j>0 \end{cases}
\end{aligned}$$

Optimal Solution

Set subgradient to zero

$$\lambda \frac{\partial f(x)}{\partial w_j} = \begin{cases} 2z_j w_j - 2\rho_j - \lambda & if \quad w_j < 0 \\ [2z_j w_j - 2\rho_j - \lambda, 2z_j w_j - 2\rho_j + \lambda] & if \quad w_j = 0 \\ 2z_j w_j - 2\rho_j + \lambda & if \quad w_j > 0 \end{cases}$$

Case 1: $w_j < 0 \Rightarrow 2z_j w_j - 2\rho_j - \lambda = 0$, then

$$\hat{w}_j = \frac{2\rho_j + \lambda}{2z_j} = \frac{\rho_j + \lambda/2}{z_j}$$

Thus for $\hat{w}_j < 0$ we need $\rho_j < -\frac{\lambda}{2}$

Optimal Solution

Set subgradient to zero

$$\lambda\frac{\partial f(x)}{\partial w_j} = \begin{cases} 2z_jw_j - 2\rho_j - \lambda & if \quad w_j < 0 \\ [2z_jw_j - 2\rho_j - \lambda, 2z_jw_j - 2\rho_j + \lambda] & if \quad w_j = 0 \\ 2z_jw_j - 2\rho_j + \lambda & if \quad w_j > 0 \end{cases}$$

Case 2: $w_j = 0 \Rightarrow -2\rho_j - \lambda \leq 0 \leq -2\rho_j + \lambda$ so that $\hat{w}_j = 0$, then

$$\begin{aligned} -2\rho_j + \lambda \geq 0 \quad &\Rightarrow \quad \rho_j \leq \lambda/2 \\ -2\rho_j - \lambda \leq 0 \quad &\Rightarrow \quad \rho_j \geq -\lambda/2 \end{aligned}$$

Thus $-\frac{\lambda}{2} \leq \rho_j \leq \frac{\lambda}{2}$

Optimal Solution

Set subgradient to zero

$$\lambda\frac{\partial f(x)}{\partial w_j} = \begin{cases} 2z_jw_j - 2\rho_j - \lambda & if \quad w_j < 0 \\ [2z_jw_j - 2\rho_j - \lambda, 2z_jw_j - 2\rho_j + \lambda] & if \quad w_j = 0 \\ 2z_jw_j - 2\rho_j + \lambda & if \quad w_j > 0 \end{cases}$$

Case 3: $w_j > 0 \Rightarrow 2z_jw_j - 2\rho_j + \lambda = 0$, then

$$\hat{w}_j = \frac{\rho_j - \lambda/2}{z_j}$$

Thus for $\hat{w}_j > 0$ we need $\rho_j > \frac{\lambda}{2}$

Optimal Solution

From the three cases

$$\lambda \frac{\partial f(x)}{\partial w_j} = \begin{cases} 2z_j w_j - 2\rho_j - \lambda & if \quad w_j < 0 \\ [2z_j w_j - 2\rho_j - \lambda, 2z_j w_j - 2\rho_j + \lambda] & if \quad w_j = 0 \\ 2z_j w_j - 2\rho_j + \lambda & if \quad w_j > 0 \end{cases}$$

$$w_j = \begin{cases} \frac{\rho_j + \lambda/2}{z_j} & if \quad \rho_j < -\lambda/2 \\ 0 & if \quad -\lambda/2 < \rho_j < \lambda/2 \\ \frac{\rho_j - \lambda/2}{z_j} & if \quad \rho_j > \lambda/2 \end{cases}$$

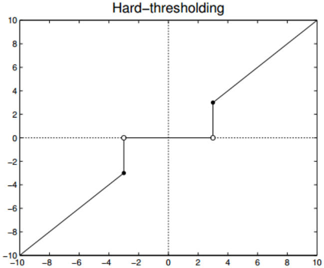
Recall: Soft-thresholding

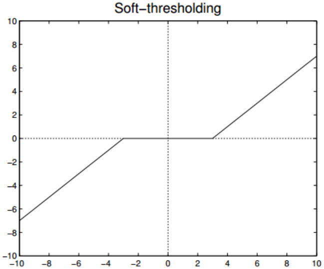
Hard-thresholding:

$$\eta_\epsilon^H(x) = x\mathbf{1}_{|x|>\epsilon}.$$

Soft-thresholding:

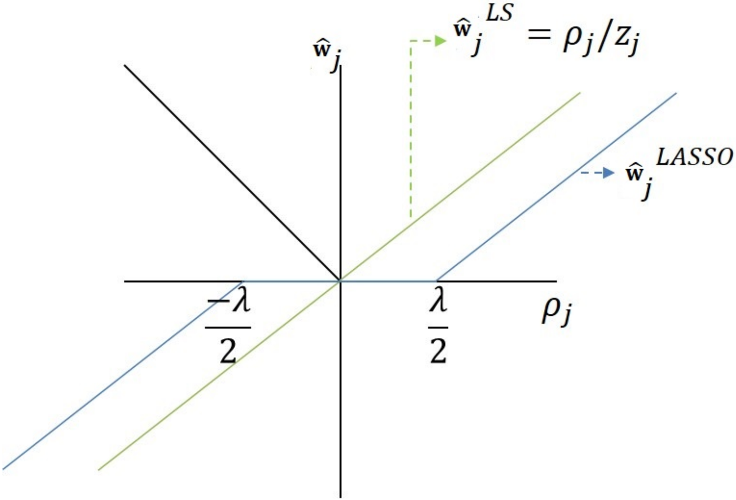
$$\eta_\epsilon^S(x) = \mathrm{sgn}(x)(|x|-\epsilon)_+$$





$$\eta_\in^S(x) = \begin{cases} x-\in & if \quad x>\in \\ x+\in & if \quad x<-\in \\ 0 & if \quad -\in<x<\in \end{cases}$$

Soft Thresholding- LASSO Coordinate Descent



$$w_j = \frac{1}{z_j}\eta^S_{\lambda/2}(\rho_j) \tag{8}$$

Coordinate Descent LASSO

- Precompute:

$$z_j = \sum_{i=1}^{N} (X_{ij})^2$$

- Initialize $\hat{w}_j = 0$
- While not converged
- for $j = 0, 1, \cdots, p$
- Compute:

$$\rho_j = \sum_{i=1}^{N} X_{ij}(y_i - \hat{y}_i(\hat{w}_{-j}))$$

- set:

$$w_j = \frac{1}{z_j} \eta^S_{\lambda/2}(\rho_j)$$

$$z_{j=1} = X_{11}^2 + X_{21}^2 + \cdots + X_{N1}^2$$

$$\mathbf{X} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1p} \\ X_{21} & X_{22} & \cdots & X_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ X_{N1} & X_{N2} & \cdots & X_{Np} \end{bmatrix}$$

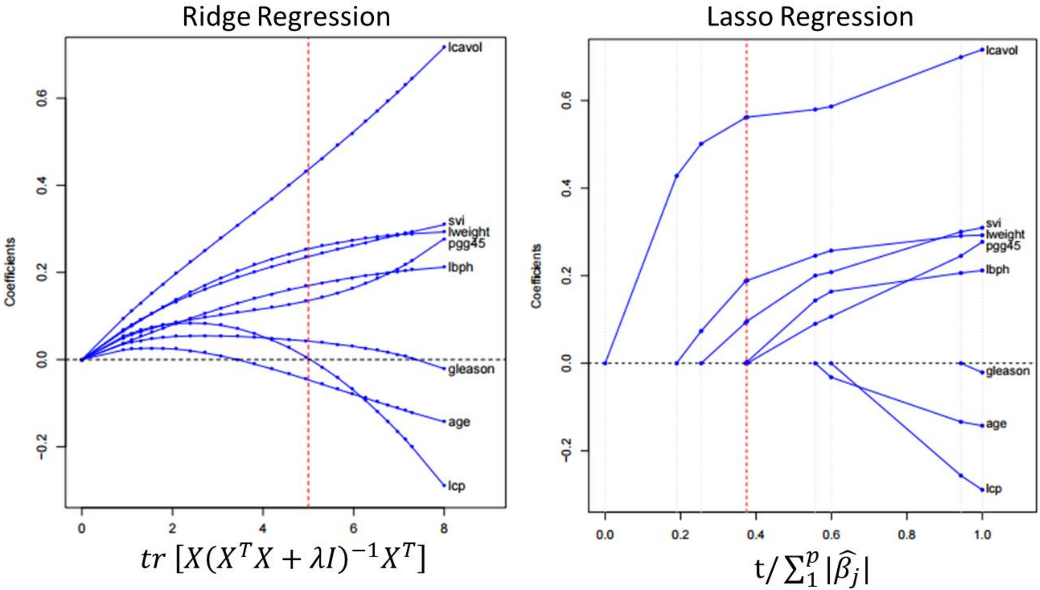
$$X_{i=N,j=2}$$

$$\rho_j = \sum_{i=1}^{N} X_{ij} \left(y_i - \sum_{k \neq j} X_{ik} \mathbf{w}_k \right)$$

Example: Prostate Cancer

- Study by Stamey et al. (1989)
- Examines the correlation between the level of prostate-specific antigen and a number of clinical measures in men who were about to receive radical prostatectomy.
- Variables: log cancer volume (lcavol), log prostate weight (lweight), age, log of the amount of benign prostatic hyperplasia (lbph), seminal vesicle invasion (svi), log of capsular penetration (lcp), Gleason score (gleason), and percent of Gleason scores 4 or 5 (pgg45).

Ridge vs Lasso Regression

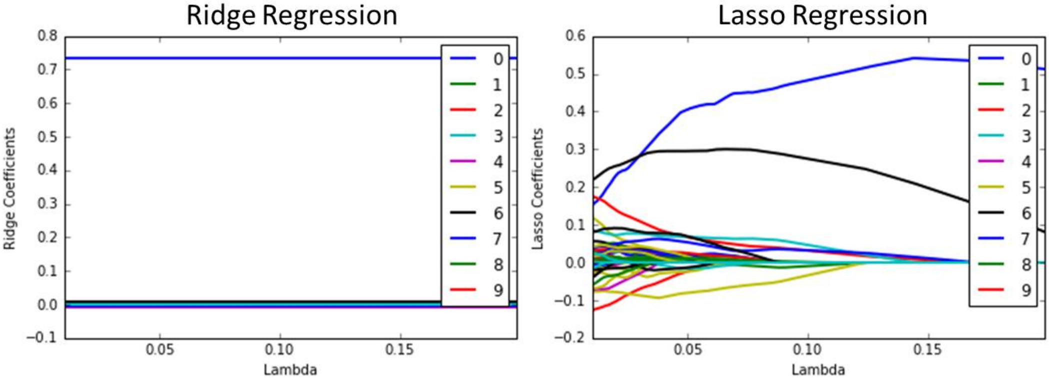


Example: Breast Cancer

We consider a classification problem involving a binary response variable $Y \in 0, 1$, describing the lymph node status of a cancer patient, and we have a covariate with $p = 7129$ gene expression measurements. There are $n = 49$ breast cancer tumor samples. The data is taken from West et al. (2001). It is known that this is a difficult, high noise classification problem.

Ridge vs Lasso Regression

Results for the 7129 predictors (Only first 10 labeled)



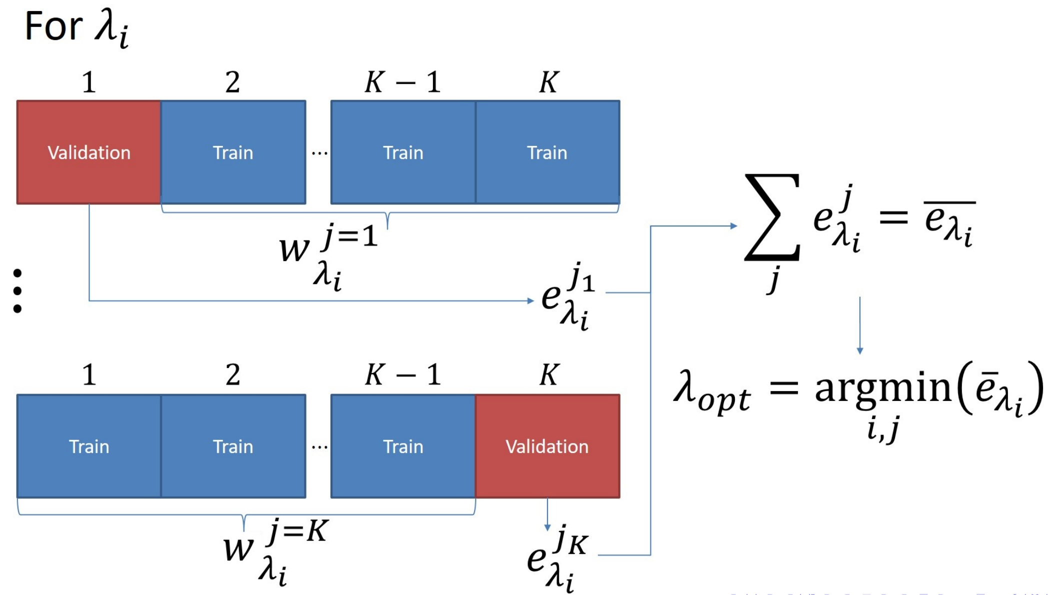
Choosing parameters: cross-validation

- Ridge and Lasso have regularization parameters.
- An *optimal* parameter needs to be chosen in a principled way

K- fold cross-validation: Split data into K equal (or almost equal) parts/folds at random.

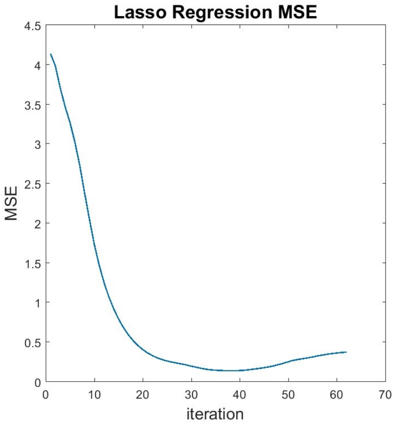
for each value λ_i **do**
    for $j = 1, \cdots, K$ **do**
        Fit model on data with fold j removed
        Test model on remaining fold j^{th} test error
    end for
    Compute average test errors for parameter λ_i
end for
Pick parameter with smallest average error

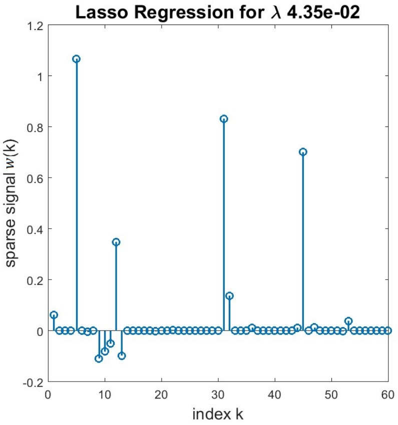
Choosing parameters: cross validation



Cross validation- Example K=5

- $\mathbf{y} = \mathbf{X}\mathbf{w} + \boldsymbol{\epsilon}$, where $\mathbf{X} \in \mathbb{R}^{40\times 60}$ is random Gaussian and $\boldsymbol{\epsilon}$ is noise.
- Original sparse signal is
  $w(k) = \delta(k-5) + 0.5\delta(k-12) + 0.9\delta(k-31) - 0.75\delta(k-45)$





Model selection vs Model assesment

- **Model selection:** estimate performance of different models in order to choose the “best" one
- **Model assessment:** having a chosen model, estimate its prediction error on new data
- When enough data is available, it is better to separate the data into three parts: train/validate, and test
- Typically: 50% train, 25 % validate, 25 % test.
- Test data is “kept in a vault", i.e. it is not used to fitting or choosing the model