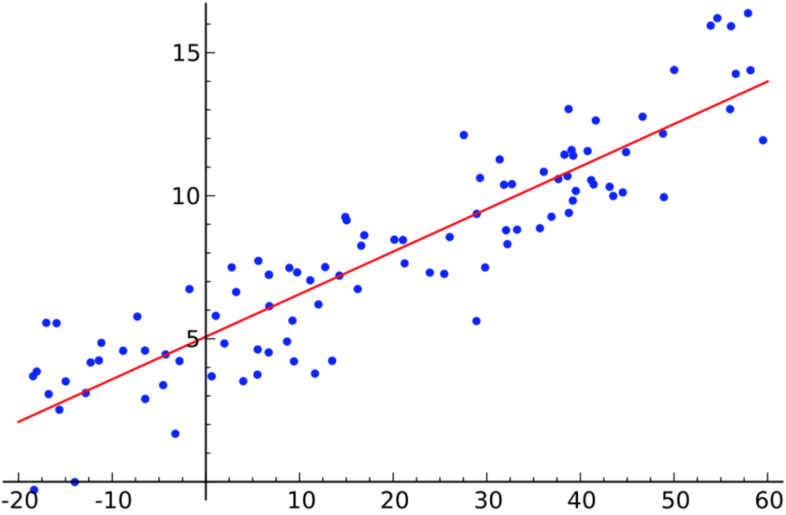
Intro to Regression

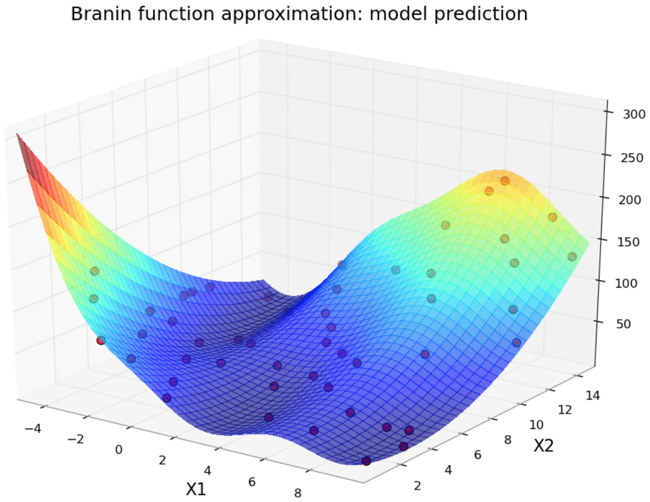
Evgeny Burnaev, Alexey Zaytsev

Skoltech, Moscow, Russia

Skolkovo Institute of Science and Technology

Outline


15
10
5
-20
-10
10
20
30
40
50
60


Branin function approximation: model prediction
300
250
200
150
100
50
−4
−2
0
2
4
6
8
X1
2
4
6
8
10
12
14
X2

- **Training data**: sample drawn i.i.d. from set X according to some distribution D

$$S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\} \in X \times Y,$$

with $Y \subseteq \mathbb{R}$ is a measurable set, $X \subseteq \mathbb{R}^d$, $\mathbf{x}_i \in \mathbb{R}^{1 \times d}$

- **Loss function**: $L : Y \times Y \to \mathbb{R}_+$ a measure of closeness, e.g. $L(y, y') = (y - y')^2$ or $L(y, y') = |y - y'|^p$ for some $p \geq 1$

- **Problem**: find hypothesis $\widehat{f} : X \to \mathbb{R}$ in $\mathbb{H}$ with small generalization error w.r.t. target f

$$R_D(\widehat{f}) = \mathbb{E}_{\mathbf{x} \sim D}[L(\widehat{f}(\mathbf{x}), f(\mathbf{x}))]$$

- **Training data**: sample drawn i.i.d. from set X according to some distribution D

$$S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\} \in X \times Y,$$

with $Y \subseteq \mathbb{R}$ is a measurable set, $X \subseteq \mathbb{R}^d$, $\mathbf{x}_i \in \mathbb{R}^{1 \times d}$

- **Loss function**: $L : Y \times Y \to \mathbb{R}_+$ a measure of closeness, e.g. $L(y, y') = (y - y')^2$ or $L(y, y') = |y - y'|^p$ for some $p \geq 1$

- **Problem**: find hypothesis $\widehat{f} : X \to \mathbb{R}$ in $\mathbb{H}$ with small generalization error w.r.t. target f

$$R_D(\widehat{f}) = \mathbb{E}_{\mathbf{x} \sim D}[L(\widehat{f}(\mathbf{x}), f(\mathbf{x}))]$$

- **Training data**: sample drawn i.i.d. from set X according to some distribution D

$$S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\} \in X \times Y,$$

with $Y \subseteq \mathbb{R}$ is a measurable set, $X \subseteq \mathbb{R}^d$, $\mathbf{x}_i \in \mathbb{R}^{1 \times d}$

- **Loss function**: $L : Y \times Y \to \mathbb{R}_+$ a measure of closeness, e.g. $L(y, y') = (y - y')^2$ or $L(y, y') = |y - y'|^p$ for some $p \geq 1$

- **Problem**: find hypothesis $\widehat{f} : X \to \mathbb{R}$ in $\mathbb{H}$ with small generalization error w.r.t. target f

$$R_D(\widehat{f}) = \mathbb{E}_{\mathbf{x} \sim D}[L(\widehat{f}(\mathbf{x}), f(\mathbf{x}))]$$

- Empirical error:

$$\widehat{R}_D(h) = \frac{1}{m}\sum_{i=1}^{m} L(\widehat{f}(\mathbf{x}_i), y_i)$$

- In much of what follows:
  - $Y = \mathbb{R}$ or $Y = [-M, M]$ for some $M > 0$
  - $L(y, y') = (y - y')^2$ is a mean squared error

- Empirical error:

$$\widehat{R}_D(h) = \frac{1}{m}\sum_{i=1}^{m} L(\widehat{f}(\mathbf{x}_i), y_i)$$

- In much of what follows:
  - $Y = \mathbb{R}$ or $Y = [-M, M]$ for some $M > 0$
  - $L(y, y') = (y - y')^2$ is a mean squared error

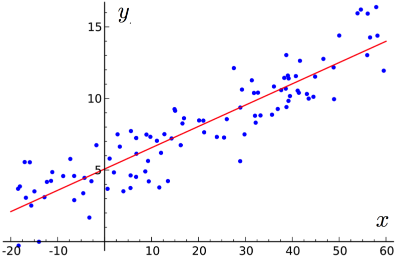
- **Object** $\mathbf{x}$: place to open a new restaurant
- **Label** y: revenue after one year of operation
- **Features**: demographic properties of a considered city district, prices for real estate in a local neighborhood, availability of offices nearby, etc.
- **Challenges**:
  - small sample size
  - a lot of features ($d \gg 1$)
  - outliers/incorrect measurements
  - non-homogeneous data (big cities vs. local towns)

- Hypothesis set: linear functions

$$\mathbb{H} = \{\mathbf{x} \to \mathbf{w} \cdot \mathbf{x}^{\mathrm{T}} + b : \mathbf{w} \in \mathbb{R}^{1 \times d}, b \in \mathbb{R}\}$$

- **Optimization problem**: empirical risk minimization

$$F(\mathbf{w}, b) = \frac{1}{m} \sum_{i=1}^{m} \left(\mathbf{w} \cdot \mathbf{x}_i^{\top} + b - y_i\right)^2 \to \min_{\mathbf{w}, b}$$

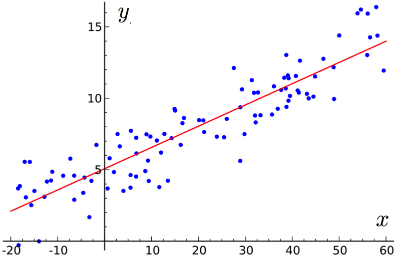


- Hypothesis set: linear functions

$$\mathbb{H} = \{\mathbf{x} \to \mathbf{w} \cdot \mathbf{x}^{\mathrm{T}} + b : \mathbf{w} \in \mathbb{R}^{1 \times d}, \, b \in \mathbb{R}\}$$

- **Optimization problem**: empirical risk minimization

$$F(\mathbf{w}, b) = \frac{1}{m} \sum_{i=1}^{m} \left(\mathbf{w} \cdot \mathbf{x}_i^{\top} + b - y_i \right)^2 \to \min_{\mathbf{w}, b}$$



- Rewrite objective function as $F(\mathbf{W}) = \frac{1}{m}\|\mathbf{X}\mathbf{W} - \mathbf{Y}\|^2$, where

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & 1 \\ \vdots & \vdots \\ \mathbf{x}_m & 1 \end{bmatrix} \in \mathbb{R}^{m\times(d+1)},\ \mathbf{W} = \begin{bmatrix} w_1 \\ \vdots \\ w_d \\ b \end{bmatrix},\ \mathbf{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}$$

- Convex and differentiable function

- Rewrite objective function as $F(\mathbf{W}) = \frac{1}{m}\|\mathbf{X}\mathbf{W} - \mathbf{Y}\|^2$, where

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & 1 \\ \vdots & \vdots \\ \mathbf{x}_m & 1 \end{bmatrix} \in \mathbb{R}^{m\times(d+1)},\ \mathbf{W} = \begin{bmatrix} w_1 \\ \vdots \\ w_d \\ b \end{bmatrix},\ \mathbf{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}$$

- Convex and differentiable function

- Rewrite objective function as $F(\mathbf{W}) = \frac{1}{m}\left\|\mathbf{X}\mathbf{W} - \mathbf{Y}\right\|^2$, where

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & 1 \\ \vdots & \vdots \\ \mathbf{x}_m & 1 \end{bmatrix} \in \mathbb{R}^{m\times(d+1)},\ \mathbf{W} = \begin{bmatrix} w_1 \\ \vdots \\ w_d \\ b \end{bmatrix},\ \mathbf{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}$$

- Convex and differentiable function

$$\nabla F(\mathbf{W}) = \frac{2}{m}\mathbf{X}^\top\left(\mathbf{X}\mathbf{W} - \mathbf{Y}\right)$$

- Rewrite objective function as $F(\mathbf{W}) = \frac{1}{m} \|\mathbf{X}\mathbf{W} - \mathbf{Y}\|^2$, where

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & 1 \\ \vdots & \vdots \\ \mathbf{x}_m & 1 \end{bmatrix} \in \mathbb{R}^{m \times (d+1)}, \mathbf{W} = \begin{bmatrix} w_1 \\ \vdots \\ w_d \\ b \end{bmatrix}, \mathbf{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}$$

- Convex and differentiable function

$$\nabla F(\mathbf{W}) = \frac{2}{m} \mathbf{X}^\top (\mathbf{X}\mathbf{W} - \mathbf{Y})$$

$$\nabla F(\mathbf{W}) = 0 \Leftrightarrow \mathbf{X}^\top (\mathbf{X}\mathbf{W} - \mathbf{Y}) = 0 \Leftrightarrow \mathbf{X}^\top \mathbf{X}\mathbf{W} = \mathbf{X}^\top \mathbf{Y}$$

- **Solution**:

$$\mathbf{W} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{Y} \text{ if } \mathbf{X}^\top \mathbf{X} \text{ invertible}$$

- Computational complexity: $O(md + d^3)$ if matrix inversion is in $O(d^3)$
- Poor guarantees in general, no regularization
- For output labels in $\mathbb{R}^{d_y}$, $d_y > 1$, solve d_y distinct linear regression problems

- **Solution**:

$$\mathbf{W} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{Y} \text{ if } \mathbf{X}^\top \mathbf{X} \text{ invertible}$$

- Computational complexity: $O(md + d^3)$ if matrix inversion is in $O(d^3)$
- Poor guarantees in general, no regularization
- For output labels in $\mathbb{R}^{d_y}$, $d_y > 1$, solve d_y distinct linear regression problems

- **Solution**:

$$\mathbf{W} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{Y} \quad \text{if} \quad \mathbf{X}^\top \mathbf{X} \quad \text{invertible}$$

- Computational complexity: $O(md + d^3)$ if matrix inversion is in $O(d^3)$
- Poor guarantees in general, no regularization
- For output labels in $\mathbb{R}^{d_y}$, $d_y > 1$, solve d_y distinct linear regression problems

- **Solution**:

$$\mathbf{W} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{Y} \quad \text{if} \quad \mathbf{X}^\top \mathbf{X} \quad \text{invertible}$$

- Computational complexity: $O(md + d^3)$ if matrix inversion is in $O(d^3)$
- Poor guarantees in general, no regularization
- For output labels in $\mathbb{R}^{d_y}$, $d_y > 1$, solve d_y distinct linear regression problems

- **Optimization problem**:

$$F(\mathbf{w}, b) = \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 + \lambda \|\mathbf{w}\|^2 \to \min_{\mathbf{w}, b},$$

where $\lambda \geq 0$ is a regularization parameter

- **Benefits**:
  - directly based on generalization bound (strict result!)
  - generalization of linear regression
  - closed-form solution
  - can be used with kernels (next slides!)

- **Optimization problem**:

$$F(\mathbf{w}, b) = \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 + \lambda \|\mathbf{w}\|^2 \to \min_{\mathbf{w}, b},$$

where $\lambda \geq 0$ is a regularization parameter

- **Benefits**:
  - directly based on generalization bound (strict result!)
  - generalization of linear regression
  - closed-form solution
  - can be used with kernels (next slides!)

- Assume $b = 0$: often constant feature is used
- Rewrite objective function as

$$F(\mathbf{W}) = \|\mathbf{X}\mathbf{W} - \mathbf{Y}\|^2 + \lambda\|\mathbf{W}\|^2$$

- Convex and differentiable function
- **Solution**:

$$\mathbf{W} = \underbrace{(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}}_{\text{always invertible!}} \mathbf{X}^\top\mathbf{Y}$$

- Assume $b = 0$: often constant feature is used
- Rewrite objective function as

$$F(\mathbf{W}) = \|\mathbf{X}\mathbf{W} - \mathbf{Y}\|^2 + \lambda\|\mathbf{W}\|^2$$

- Convex and differentiable function
- **Solution**:

$$\mathbf{W} = \underbrace{(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}}_{\text{always invertible!}} \mathbf{X}^\top\mathbf{Y}$$

- Assume $b = 0$: often constant feature is used
- Rewrite objective function as

$$F(\mathbf{W}) = \|\mathbf{X}\mathbf{W} - \mathbf{Y}\|^2 + \lambda\|\mathbf{W}\|^2$$

- Convex and differentiable function
- **Solution**:

$$\mathbf{W} = \underbrace{(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}}_{\text{always invertible!}} \mathbf{X}^\top\mathbf{Y}$$

Ridge Regression: Solution

- Assume $b = 0$: often constant feature is used
- Rewrite objective function as

$$F(\mathbf{W}) = \|\mathbf{X}\mathbf{W} - \mathbf{Y}\|^2 + \lambda\|\mathbf{W}\|^2$$

- Convex and differentiable function

$$\nabla F(\mathbf{W}) = 2\lambda\mathbf{W} + 2\mathbf{X}^\top(\mathbf{X}\mathbf{W} - \mathbf{Y})$$

- **Solution**:

$$\mathbf{W} = \underbrace{(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}}_{\text{always invertible!}} \mathbf{X}^\top\mathbf{Y}$$

- Assume $b = 0$: often constant feature is used
- Rewrite objective function as

$$F(\mathbf{W}) = \|\mathbf{X}\mathbf{W} - \mathbf{Y}\|^2 + \lambda\|\mathbf{W}\|^2$$

- Convex and differentiable function

$$\nabla F(\mathbf{W}) = 2\lambda\mathbf{W} + 2\mathbf{X}^\top(\mathbf{X}\mathbf{W} - \mathbf{Y})$$
$$\nabla F(\mathbf{W}) = 0 \Leftrightarrow (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})\mathbf{W} = \mathbf{X}^\top\mathbf{Y}$$

- **Solution**:

$$\mathbf{W} = \underbrace{(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}}_{\text{always invertible!}} \mathbf{X}^\top\mathbf{Y}$$

- Assume $b = 0$: often constant feature is used
- Rewrite objective function as

$$F(\mathbf{W}) = \|\mathbf{X}\mathbf{W} - \mathbf{Y}\|^2 + \lambda\|\mathbf{W}\|^2$$

- Convex and differentiable function

$$\nabla F(\mathbf{W}) = 2\lambda\mathbf{W} + 2\mathbf{X}^\top(\mathbf{X}\mathbf{W} - \mathbf{Y})$$

$$\nabla F(\mathbf{W}) = 0 \Leftrightarrow (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})\mathbf{W} = \mathbf{X}^\top\mathbf{Y}$$

- **Solution**:

$$\mathbf{W} = \underbrace{(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}}_{\text{always invertible!}} \mathbf{X}^\top\mathbf{Y}$$

- We can easily prove that

$$(\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top = \mathbf{X}^\top (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1}$$

- **Dual solution**: thus we get that

$$\mathbf{W} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{Y} = \mathbf{X}^\top \underbrace{(\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1} \mathbf{Y}}_{\text{new variable } \alpha}$$

- With

$$\alpha = (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1} \mathbf{Y},$$

- We can use dual representation of the solution

- We can easily prove that

$$(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top = \mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}$$

- **Dual solution**: thus we get that

$$\mathbf{W} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{Y} = \mathbf{X}^\top \underbrace{(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{Y}}_{\text{new variable } \boldsymbol{\alpha}}$$

- With

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{Y},$$

- We can use dual representation of the solution

- We can easily prove that

$$(\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top = \mathbf{X}^\top (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1}$$

- **Dual solution**: thus we get that

$$\mathbf{W} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{Y} = \mathbf{X}^\top \underbrace{(\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1} \mathbf{Y}}_{\text{new variable } \boldsymbol{\alpha}}$$

- With

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1} \mathbf{Y},$$

- We can use dual representation of the solution

- We can easily prove that

$$(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top = \mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}$$

- **Dual solution**: thus we get that

$$\mathbf{W} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{Y} = \mathbf{X}^\top \underbrace{(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{Y}}_{\text{new variable } \boldsymbol{\alpha}}$$

- With

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{Y},$$

  we can represent $\mathbf{W}$ as

$$\mathbf{W} = \mathbf{X}^\top\boldsymbol{\alpha} = \sum_{i=1}^{m} \alpha_i \mathbf{x}_i^\top,$$

- We can use dual representation of the solution

- We can easily prove that

$$(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top = \mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}$$

- **Dual solution**: thus we get that

$$\mathbf{W} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{Y} = \mathbf{X}^\top \underbrace{(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{Y}}_{\text{new variable } \boldsymbol{\alpha}}$$

- With

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{Y},$$

we can represent $\mathbf{W}$ as

$$\mathbf{W} = \mathbf{X}^\top\boldsymbol{\alpha} = \sum_{i=1}^{m} \alpha_i \mathbf{x}_i^\top,$$

- We can use dual representation of the solution

- We can easily prove that

$$(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top = \mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}$$

- **Dual solution**: thus we get that

$$\mathbf{W} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{Y} = \mathbf{X}^\top \underbrace{(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{Y}}_{\text{new variable } \boldsymbol{\alpha}}$$

- With

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{Y},$$

  we can represent $\mathbf{W}$ as

$$\mathbf{W} = \mathbf{X}^\top\boldsymbol{\alpha} = \sum_{i=1}^{m} \alpha_i \mathbf{x}_i^\top,$$

- We can use dual representation of the solution

$$\widehat{f}(\mathbf{x}) = \mathbf{x} \cdot \mathbf{W}$$

- We can easily prove that

$$(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top = \mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}$$

- **Dual solution**: thus we get that

$$\mathbf{W} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{Y} = \mathbf{X}^\top \underbrace{(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{Y}}_{\text{new variable } \boldsymbol{\alpha}}$$

- With

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{Y},$$

  we can represent $\mathbf{W}$ as

$$\mathbf{W} = \mathbf{X}^\top\boldsymbol{\alpha} = \sum_{i=1}^{m} \alpha_i \mathbf{x}_i^\top,$$

- We can use dual representation of the solution

$$\widehat{f}(\mathbf{x}) = \mathbf{x}\cdot\mathbf{W} = \sum_{i=1}^{m} \alpha_i(\mathbf{x}\cdot\mathbf{x}_i^\top)$$

Type	Solution	Prediction
Primal	$O(md^2 + d^3)$	$O(d)$
Dual	$O(\varkappa m^2 + m^3)$	$O(\varkappa m)$

Here $\varkappa$ denotes the time complexity of computing a dot product; Euclidian dot product $\varkappa = O(d)$

- **Optimization problem**: "Least Absolute Shrinkage and Selection Operator"

$$F(\mathbf{w}, b) = \lambda \|\mathbf{w}\|_1 + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

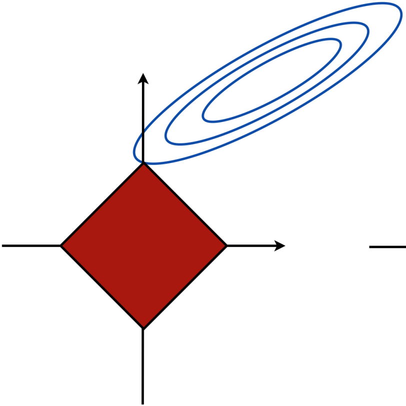
where $\lambda \geq 0$ is a regularization parameter

- **Solution**: equivalent convex quadratic programming (QP)
  - general: standard QP solvers
  - specific algorithms: LARS (least angle regression procedure), entire path of solution

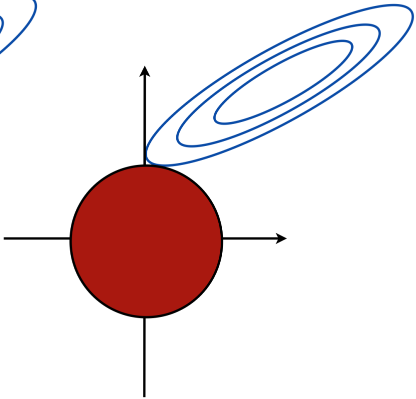
- **Optimization problem**: "Least Absolute Shrinkage and Selection Operator"

$$F(\mathbf{w}, b) = \lambda \|\mathbf{w}\|_1 + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

  where $\lambda \geq 0$ is a regularization parameter
- **Solution**: equivalent convex quadratic programming (QP)
  - general: standard QP solvers
  - specific algorithms: LARS (least angle regression procedure), entire path of solution



L1 regularization



L2 regularization

- Advantages
  - strong theoretical guarantees
  - sparse solution
  - feature selection
- Disadvantages
  - no natural use of kernels (next slides!)
  - no closed-form solution (not necessary, but can be convenient for theoretical analysis)
- **Empirical recipe** to provide better prediction performance:
  - First, perform variable selection using LASSO
  - Second, re-estimate model parameters with selected variables using Ridge Regression

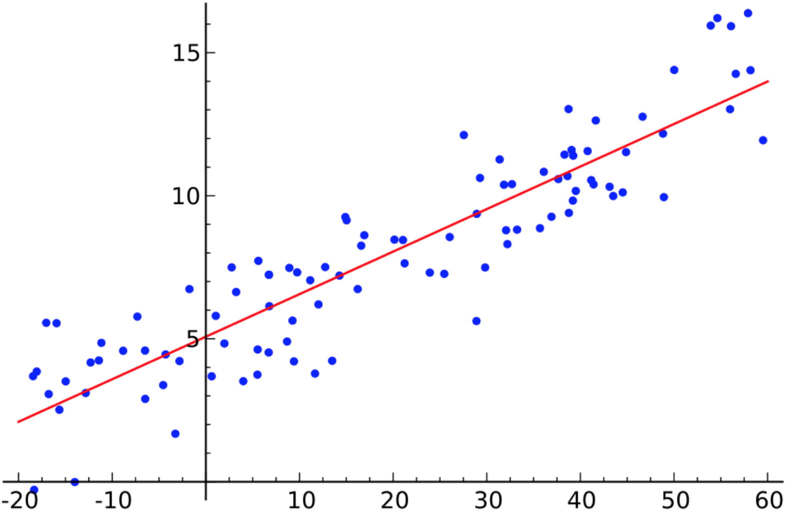
- Advantages
  - strong theoretical guarantees
  - sparse solution
  - feature selection
- Disadvantages
  - no natural use of kernels (next slides!)
  - no closed-form solution (not necessary, but can be convenient for theoretical analysis)
- **Empirical recipe** to provide better prediction performance:
  - First, perform variable selection using LASSO
  - Second, re-estimate model parameters with selected variables using Ridge Regression

- Advantages
  - strong theoretical guarantees
  - sparse solution
  - feature selection

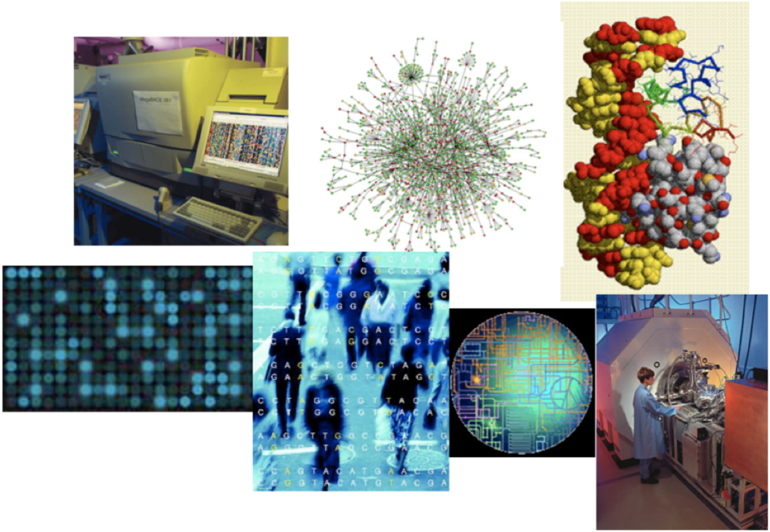
- Disadvantages
  - no natural use of kernels (next slides!)
  - no closed-form solution (not necessary, but can be convenient for theoretical analysis)

- **Empirical recipe** to provide better prediction performance:
  - First, perform variable selection using LASSO
  - Second, re-estimate model parameters with selected variables using Ridge Regression

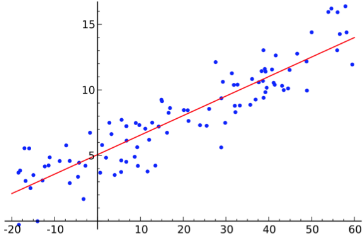
What we know how to solve

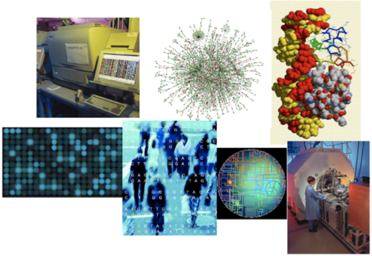


But real data are often more complicated ...



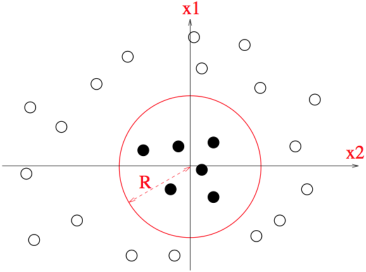
Main goal of using kernels



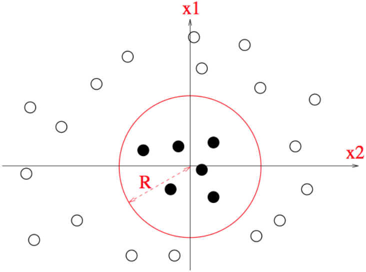


Show some classical examples how to extend well-understood, linear statistical learning techniques to real-world, complicated, structured, high-dimensional data (texts, time series, graphs, distributions, permutations, ...)

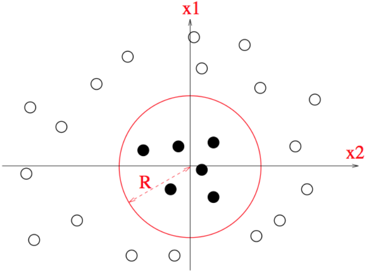
- Efficient computation of inner products in high dimension
- Non-linear decision boundary
- Learning with non-vectorial inputs
- More informative features
- Kernels allow to perform pairwise comparisons



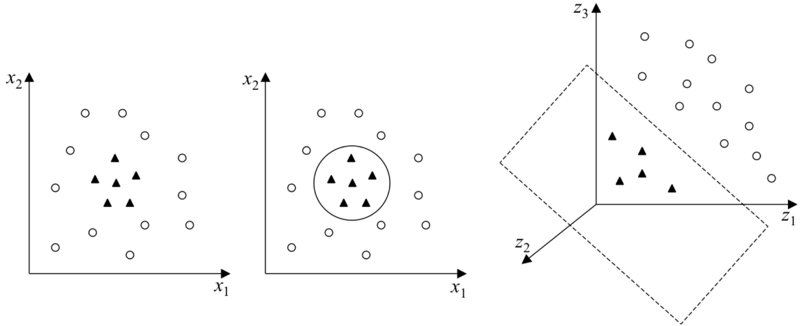
- Linear separation impossible in most problems
- Non-linear mapping $\Phi : X \to \mathbb{H}$ from input space to high-dimensional feature space
- Generalization ability: independent of $\dim(\mathbb{H})$, depends only on d and m



- Linear separation impossible in most problems
- Non-linear mapping $\Phi : X \to \mathbb{H}$ from input space to high-dimensional feature space
- Generalization ability: independent of $\dim(\mathbb{H})$, depends only on d and m

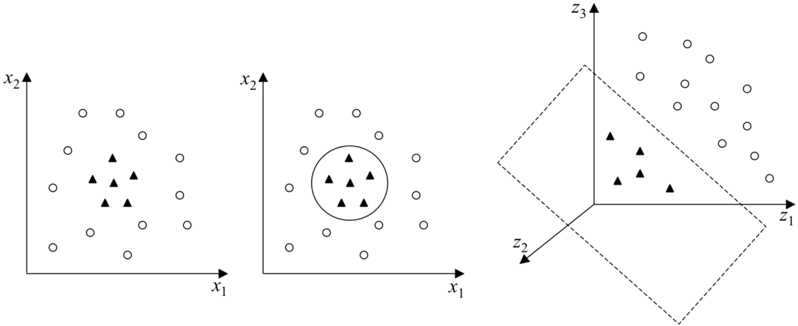


- Linear separation impossible in most problems
- Non-linear mapping $\Phi : X \to \mathbb{H}$ from input space to high-dimensional feature space
- Generalization ability: independent of $\dim(\mathbb{H})$, depends only on d and m



For $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$, let $\Phi(\mathbf{x}) = (x_1^2, \sqrt{2}x_1x_2, x_2^2) \in \mathbb{R}^3$. Then

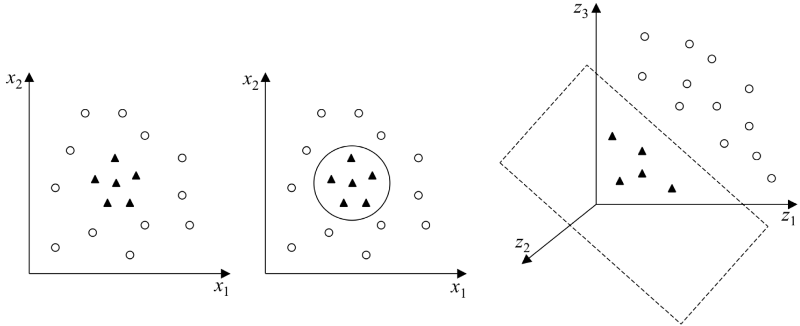
$$K(\mathbf{x}', \mathbf{x}) = \Phi(\mathbf{x}') \cdot \Phi(\mathbf{x})^\top \quad \text{[dot product of features]}$$



For $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$, let $\Phi(\mathbf{x}) = (x_1^2, \sqrt{2}x_1x_2, x_2^2) \in \mathbb{R}^3$. Then

$$K(\mathbf{x}', \mathbf{x}) = \Phi(\mathbf{x}') \cdot \Phi(\mathbf{x})^\top \quad \text{[dot product of features]}$$
$$= x_1^2(x_1')^2 + 2x_1x_2x_1'x_2' + x_2^2(x_2')^2$$

Example: polynomial kernel



For $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$, let $\Phi(\mathbf{x}) = (x_1^2, \sqrt{2}x_1 x_2, x_2^2) \in \mathbb{R}^3$. Then

$$K(\mathbf{x}', \mathbf{x}) = \Phi(\mathbf{x}') \cdot \Phi(\mathbf{x})^\top \quad \text{[dot product of features]}$$
$$= x_1^2 (x_1')^2 + 2 x_1 x_2 x_1' x_2' + x_2^2 (x_2')^2$$
$$= (x_1 x_1' + x_2 x_2')^2 = (\mathbf{x}' \cdot \mathbf{x}^\top)^2$$

- **Idea**:
  - Define $K : X \times X \to \mathbb{R}$ called kernel, such that

$$\Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}')^{\top} = K(\mathbf{x}, \mathbf{x}')$$

  - K is often interpreted as a similarity measure

- **Benefits**:
  - Efficiency: K is often more efficient to compute than Φ and the dot product
  - Flexibility: K can be chosen arbitrarily so long as the existence of Φ is guaranteed (PDS condition or Mercer's condition)

- **Idea**:
  - Define $K : X \times X \to \mathbb{R}$ called kernel, such that
  $$\Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}')^{\top} = K(\mathbf{x}, \mathbf{x}')$$
  - K is often interpreted as a similarity measure

- **Benefits**:
  - Efficiency: K is often more efficient to compute than Φ and the dot product
  - Flexibility: K can be chosen arbitrarily so long as the existence of Φ is guaranteed (PDS condition or Mercer's condition)

- **Idea**:
  - Define $K : X \times X \to \mathbb{R}$ called kernel, such that
  $$\Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}')^\top = K(\mathbf{x}, \mathbf{x}')$$
  - K is often interpreted as a similarity measure
- **Benefits**:
  - Efficiency: K is often more efficient to compute than Φ and the dot product
  - Flexibility: K can be chosen arbitrarily so long as the existence of Φ is guaranteed (PDS condition or Mercer's condition)

- **Idea**:
  - Define $K : X \times X \to \mathbb{R}$ called kernel, such that
  $$\Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}')^\top = K(\mathbf{x}, \mathbf{x}')$$
  - K is often interpreted as a similarity measure

- **Benefits**:
  - Efficiency: K is often more efficient to compute than Φ and the dot product
  - Flexibility: K can be chosen arbitrarily so long as the existence of Φ is guaranteed (PDS condition or Mercer's condition)

- **Definition**:

$$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^d,\ K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}' \cdot \mathbf{x}^\top + c)^p,\ c > 0$$

- **Example**: for $p = 2$ and $d = 2$,

- **Definition**:

$$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^d,\ K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}' \cdot \mathbf{x}^\top + c)^p,\ c > 0$$

- **Example**: for $p = 2$ and $d = 2$,

- **Definition**:

$$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^d,\ K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}' \cdot \mathbf{x}^\top + c)^p,\ c > 0$$

- **Example**: for $p = 2$ and $d = 2$,

$$K(\mathbf{x}, \mathbf{x}') = (x_1 x_1' + x_2 x_2' + c)^2$$

- **Definition**:

$$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^d,\ K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}' \cdot \mathbf{x}^\top + c)^p,\ c > 0$$

- **Example**: for $p = 2$ and $d = 2$,

$$\begin{aligned} K(\mathbf{x}, \mathbf{x}') &= (x_1 x_1' + x_2 x_2' + c)^2 \\ &= \left[x_1^2, x_2^2, \sqrt{2} x_1 x_2, \sqrt{2c} x_1, \sqrt{2c} x_2, c\right] \cdot \begin{bmatrix} (x_1')^2 \\ (x_2')^2 \\ \sqrt{2} x_1' x_2' \\ \sqrt{2c} x_1' \\ \sqrt{2c} x_2' \\ c \end{bmatrix} \end{aligned}$$

- **Gaussian kernels**:

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\sigma^2}\right), \, \sigma \neq 0$$

- **Sigmoid kernels**:

$$K(\mathbf{x}, \mathbf{x}') = \tanh(a(\mathbf{x} \cdot \mathbf{x}') + b), \;\; a, b > 0$$

- **Definition**: a kernel $K: X \times X \to \mathbb{R}$ is *positive definite symmetric* (PDS) is for any $\{\mathbf{x}_1, \dots, \mathbf{x}_m\} \subseteq X$ the matrix $\mathrm{K} = [K(\mathbf{x}_i, \mathbf{x}_j)]_{ij} \in \mathbb{R}^{m \times m}$ is symmetric positive semi-definite (SPSD)

- Matrix K SPSD if symmetric and one of the 2 equiv. cond.'s:
  - its eigenvalues are non-negative
  - for any $\mathbf{c} \in \mathbb{R}^{m \times 1}$, $\mathbf{c}^\top \mathrm{K} \mathbf{c} = \sum_{i,j=1}^m c_i c_j K(\mathbf{x}_i, \mathbf{x}_j) \geq 0$

- **Terminology**: PDS for kernels, SPDS for kernel matrices

- **Definition**: a kernel $K : X \times X \to \mathbb{R}$ is *positive definite symmetric* (PDS) is for any $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq X$ the matrix $\mathrm{K} = [K(\mathbf{x}_i, \mathbf{x}_j)]_{ij} \in \mathbb{R}^{m \times m}$ is symmetric positive semi-definite (SPSD)

- Matrix K SPSD if symmetric and one of the 2 equiv. cond.'s:
  - its eigenvalues are non-negative
  - for any $\mathbf{c} \in \mathbb{R}^{m \times 1}$, $\mathbf{c}^\top \mathrm{K} \mathbf{c} = \sum_{i,j=1}^{m} c_i c_j K(\mathbf{x}_i, \mathbf{x}_j) \geq 0$

- **Terminology**: PDS for kernels, SPDS for kernel matrices

- Usual linear ridge regression in dual representation

$$\widehat{f}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i (\mathbf{x} \cdot \mathbf{x}_i^\top)$$

with

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1} \mathbf{Y}$$

- Kernel ridge regression

- Usual linear ridge regression in dual representation

$$\widehat{f}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i (\mathbf{x} \cdot \mathbf{x}_i^\top)$$

with

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1}\mathbf{Y}$$

- Kernel ridge regression

- Usual linear ridge regression in dual representation

$$\widehat{f}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i (\mathbf{x} \cdot \mathbf{x}_i^\top)$$

with

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1}\mathbf{Y}$$

- Kernel ridge regression

$$\widehat{f}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i (\Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}_i)^\top) = \sum_{i=1}^{m} \alpha_i K(\mathbf{x}_i, \mathbf{x})$$

- Usual linear ridge regression in dual representation

$$\widehat{f}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i (\mathbf{x} \cdot \mathbf{x}_i^\top)$$

with

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1}\mathbf{Y}$$

- Kernel ridge regression

$$\widehat{f}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i (\Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}_i)^\top) = \sum_{i=1}^{m} \alpha_i K(\mathbf{x}_i, \mathbf{x})$$

with

$$\boldsymbol{\alpha} = (\Phi(\mathbf{X}) \cdot \Phi(\mathbf{X})^\top + \lambda \mathbf{I})^{-1}\mathbf{Y} = (\mathrm{K} + \lambda \mathbf{I})^{-1}\mathbf{Y},$$

- Usual linear ridge regression in dual representation

$$\widehat{f}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i (\mathbf{x} \cdot \mathbf{x}_i^\top)$$

with

$$\boldsymbol{\alpha} = (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1}\mathbf{Y}$$

- Kernel ridge regression

$$\widehat{f}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i (\Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}_i)^\top) = \sum_{i=1}^{m} \alpha_i K(\mathbf{x}_i, \mathbf{x})$$

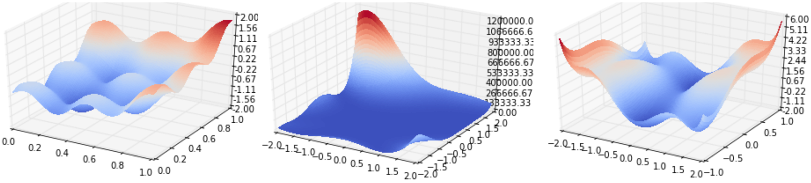
with

$$\boldsymbol{\alpha} = (\Phi(\mathbf{X}) \cdot \Phi(\mathbf{X})^\top + \lambda \mathbf{I})^{-1}\mathbf{Y} = (\mathrm{K} + \lambda \mathbf{I})^{-1}\mathbf{Y},$$

where

$$\mathrm{K} = \{\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)^\top\}_{i,j=1}^{m} = \{K(\mathbf{x}_i, \mathbf{x}_j)\}_{i,j=1}^{m}$$

Example: Kernel ridge regression (I)



Example: Kernel ridge regression (II)

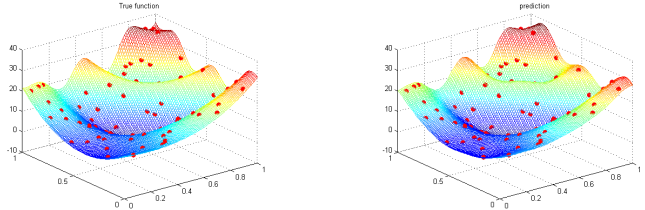


Figure – Mystery function (left) and an approximation (right) for the training sample of size 80

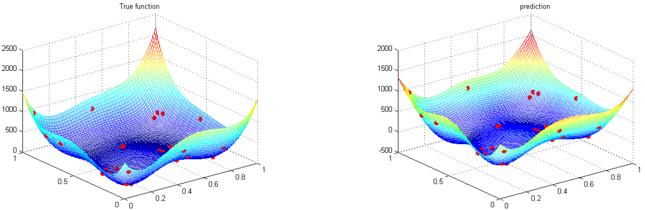


Figure – Himmelblau function (left) and an approximation (right) for the training sample of size 40

- Advantages
  - strong theoretical guarantees
  - generalization to outputs in $\mathbb{R}^p$: single matrix inversion
  - use of kernels
- Disadvantages
  - solution is not sparse
  - training time for large matrices: low-rank approximations of kernel matrix, e.g., Nyström approximation, partial Cholesky decomposition
- In practice in many cases regression based on decision trees and gradient boosting is used (next lectures!)

Comments

- Advantages
  - strong theoretical guarantees
  - generalization to outputs in $\mathbb{R}^p$: single matrix inversion
  - use of kernels
- Disadvantages
  - solution is not sparse
  - training time for large matrices: low-rank approximations of kernel matrix, e.g., Nyström approximation, partial Cholesky decomposition
- In practice in many cases regression based on decision trees and gradient boosting is used (next lectures!)

- Advantages
  - strong theoretical guarantees
  - generalization to outputs in $\mathbb{R}^p$: single matrix inversion
  - use of kernels
- Disadvantages
  - solution is not sparse
  - training time for large matrices: low-rank approximations of kernel matrix, e.g., Nyström approximation, partial Cholesky decomposition
- In practice in many cases regression based on decision trees and gradient boosting is used (next lectures!)

- **Ordinary Least Squares** (OLS):

$$F(\mathbf{w}, b) = \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

- **Ridge Regression**:

$$F(\mathbf{w}, b) = \lambda \|\mathbf{w}\|_2^2 + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 t \to \min_{\mathbf{w}, b},$$

where $\lambda \geq 0$ is a regularization parameter

- **LASSO**:

$$F(\mathbf{w}, b) = \lambda \|\mathbf{w}\|_1 + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

where $\lambda \geq 0$ is a regularization parameter

- **Elastic Net**:

$$F(\mathbf{w}, b) = \lambda \left((1 - \alpha) \|\mathbf{w}\|_1 + \alpha \|\mathbf{w}\|_2^2 \right) + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

where $\lambda \geq 0$ is a regularization parameter, $\alpha \in [0, 1]$ is a parameter of a convex combination

- **Ordinary Least Squares** (OLS):

$$F(\mathbf{w}, b) = \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w},b},$$

- **Ridge Regression**:

$$F(\mathbf{w}, b) = \lambda \|\mathbf{w}\|_2^2 + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 t \to \min_{\mathbf{w},b},$$

where $\lambda \geq 0$ is a regularization parameter

- **LASSO**:

$$F(\mathbf{w}, b) = \lambda \|\mathbf{w}\|_1 + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w},b},$$

where $\lambda \geq 0$ is a regularization parameter

- **Elastic Net**:

$$F(\mathbf{w}, b) = \lambda \left((1-\alpha)\|\mathbf{w}\|_1 + \alpha \|\mathbf{w}\|_2^2\right) + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w},b},$$

where $\lambda \geq 0$ is a regularization parameter, $\alpha \in [0, 1]$ is a parameter of a convex combination

- **Ordinary Least Squares** (OLS):

$$F(\mathbf{w}, b) = \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

- **Ridge Regression**:

$$F(\mathbf{w}, b) = \lambda \|\mathbf{w}\|_2^2 + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 t \to \min_{\mathbf{w}, b},$$

  where $\lambda \geq 0$ is a regularization parameter

- **LASSO**:

$$F(\mathbf{w}, b) = \lambda \|\mathbf{w}\|_1 + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

  where $\lambda \geq 0$ is a regularization parameter

- **Elastic Net**:

$$F(\mathbf{w}, b) = \lambda \left((1 - \alpha)\|\mathbf{w}\|_1 + \alpha \|\mathbf{w}\|_2^2\right) + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

  where $\lambda \geq 0$ is a regularization parameter, $\alpha \in [0, 1]$ is a parameter of a convex combination

- **Ordinary Least Squares** (OLS):
$$F(\mathbf{w}, b) = \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$
- **Ridge Regression**:
$$F(\mathbf{w}, b) = \lambda \|\mathbf{w}\|_2^2 + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 t \to \min_{\mathbf{w}, b},$$
where $\lambda \geq 0$ is a regularization parameter
- **LASSO**:
$$F(\mathbf{w}, b) = \lambda \|\mathbf{w}\|_1 + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$
where $\lambda \geq 0$ is a regularization parameter
- **Elastic Net**:
$$F(\mathbf{w}, b) = \lambda \left((1 - \alpha)\|\mathbf{w}\|_1 + \alpha \|\mathbf{w}\|_2^2\right) + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$
where $\lambda \geq 0$ is a regularization parameter, $\alpha \in [0, 1]$ is a parameter of a convex combination

- **Prediction accuracy** and **model interpretation** are two important aspects of regression models
- LASSO is a penalized regression method to improve OLS and Ridge regression
- LASSO does shrinkage and variable selection simultaneously for better prediction and model interpretation
- Disadvantages of LASSO:
  - In the $d > m$ case, the lasso selects at most m variables before it saturates, because of the nature of the convex optimization problem. However, e.g. in gene selection problems typically $d > m$
  - Lasso can not do **group selection**, which is important e.g. in gene selection problems

- **Prediction accuracy** and **model interpretation** are two important aspects of regression models
- LASSO is a penalized regression method to improve OLS and Ridge regression
- LASSO does shrinkage and variable selection simultaneously for better prediction and model interpretation
- Disadvantages of LASSO:
  - In the $d > m$ case, the lasso selects at most m variables before it saturates, because of the nature of the convex optimization problem. However, e.g. in gene selection problems typically $d > m$
  - Lasso can not do **group selection**, which is important e.g. in gene selection problems

- **Prediction accuracy** and **model interpretation** are two important aspects of regression models
- LASSO is a penalized regression method to improve OLS and Ridge regression
- LASSO does shrinkage and variable selection simultaneously for better prediction and model interpretation
- Disadvantages of LASSO:
  - In the $d > m$ case, the lasso selects at most m variables before it saturates, because of the nature of the convex optimization problem. However, e.g. in gene selection problems typically $d > m$
  - Lasso can not do **group selection**, which is important e.g. in gene selection problems

- **Prediction accuracy** and **model interpretation** are two important aspects of regression models
- LASSO is a penalized regression method to improve OLS and Ridge regression
- LASSO does shrinkage and variable selection simultaneously for better prediction and model interpretation
- Disadvantages of LASSO:
  - In the $d > m$ case, the lasso selects at most m variables before it saturates, because of the nature of the convex optimization problem. However, e.g. in gene selection problems typically $d > m$
  - Lasso can not do **group selection**, which is important e.g. in gene selection problems

- **Prediction accuracy** and **model interpretation** are two important aspects of regression models
- LASSO is a penalized regression method to improve OLS and Ridge regression
- LASSO does shrinkage and variable selection simultaneously for better prediction and model interpretation
- Disadvantages of LASSO:
  - In the $d > m$ case, the lasso selects at most m variables before it saturates, because of the nature of the convex optimization problem. However, e.g. in gene selection problems typically $d > m$
  - Lasso can not do **group selection**, which is important e.g. in gene selection problems

- If there is a group of variables among which the pairwise correlations are very high, then the LASSO tends to select only one variable from the group and does not care which one is selected
- It has been empirically observed that the prediction performance of the LASSO is dominated by ridge regression
- **Elastic Net**: Regression, variable selection, with the capacity of selecting groups of correlated variables.
- *Elastic Net is like a stretchable fishing net that retains 'all the big fish'*

- If there is a group of variables among which the pairwise correlations are very high, then the LASSO tends to select only one variable from the group and does not care which one is selected
- It has been empirically observed that the prediction performance of the LASSO is dominated by ridge regression
- **Elastic Net**: Regression, variable selection, with the capacity of selecting groups of correlated variables.
- *Elastic Net is like a stretchable fishing net that retains 'all the big fish'*

- If there is a group of variables among which the pairwise correlations are very high, then the LASSO tends to select only one variable from the group and does not care which one is selected
- It has been empirically observed that the prediction performance of the LASSO is dominated by ridge regression
- **Elastic Net**: Regression, variable selection, with the capacity of selecting groups of correlated variables.
- *Elastic Net is like a stretchable fishing net that retains 'all the big fish'*

- If there is a group of variables among which the pairwise correlations are very high, then the LASSO tends to select only one variable from the group and does not care which one is selected
- It has been empirically observed that the prediction performance of the LASSO is dominated by ridge regression
- **Elastic Net**: Regression, variable selection, with the capacity of selecting groups of correlated variables.
- *Elastic Net is like a stretchable fishing net that retains 'all the big fish'*

The optimization problem for **Elastic Net** is

$$F(\mathbf{w}, b) = \lambda \left((1-\alpha)\|\mathbf{w}\|_1 + \alpha\|\mathbf{w}\|_2^2\right) + \sum_{i=1}^{m} (\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

- **Elastic Net** combines l_1 and l_2 penalties
- Ridge regression: $\alpha \to 1$; LASSO: $\alpha \to 0$
- It can be proved that in case of orthogonal design matrix ($\mathbf{X}^\top \mathbf{X} = \mathrm{I}$) analytical solution exists

The optimization problem for **Elastic Net** is

$$F(\mathbf{w}, b) = \lambda\left((1-\alpha)\|\mathbf{w}\|_1 + \alpha\|\mathbf{w}\|_2^2\right) + \sum_{i=1}^{m}(\mathbf{w}\cdot\mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w},b},$$

- **Elastic Net** combines l_1 and l_2 penalties
- Ridge regression: $\alpha \to 1$; LASSO: $\alpha \to 0$
- It can be proved that in case of orthogonal design matrix ($\mathbf{X}^\top\mathbf{X} = \mathrm{I}$) analytical solution exists

The optimization problem for **Elastic Net** is

$$F(\mathbf{w}, b) = \lambda \left((1-\alpha)\|\mathbf{w}\|_1 + \alpha\|\mathbf{w}\|_2^2\right) + \sum_{i=1}^{m}(\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

- **Elastic Net** combines l_1 and l_2 penalties
- Ridge regression: $\alpha \to 1$; LASSO: $\alpha \to 0$
- It can be proved that in case of orthogonal design matrix ($\mathbf{X}^\top \mathbf{X} = \mathrm{I}$) analytical solution exists

The optimization problem for **Elastic Net** is

$$F(\mathbf{w}, b) = \lambda \left((1-\alpha)\|\mathbf{w}\|_1 + \alpha\|\mathbf{w}\|_2^2\right) + \sum_{i=1}^{m}(\mathbf{w}\cdot\mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w},b},$$

- **Elastic Net** combines l_1 and l_2 penalties
- Ridge regression: $\alpha \to 1$; LASSO: $\alpha \to 0$
- It can be proved that in case of orthogonal design matrix ($\mathbf{X}^\top\mathbf{X} = \mathrm{I}$) analytical solution exists

$$\widehat{w}_i(Ridge) = \widehat{w}_i(OLS)/(1+\lambda_2),$$

The optimization problem for **Elastic Net** is

$$F(\mathbf{w}, b) = \lambda \left((1-\alpha)\|\mathbf{w}\|_1 + \alpha\|\mathbf{w}\|_2^2\right) + \sum_{i=1}^{m}(\mathbf{w} \cdot \mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

- **Elastic Net** combines l_1 and l_2 penalties
- Ridge regression: $\alpha \to 1$; LASSO: $\alpha \to 0$
- It can be proved that in case of orthogonal design matrix ($\mathbf{X}^\top\mathbf{X} = \mathrm{I}$) analytical solution exists

$$\widehat{w}_i(Ridge) = \widehat{w}_i(OLS)/(1+\lambda_2),$$
$$\widehat{w}_i(LASSO) = \left(|\widehat{w}_i(OLS)| - \lambda_1/2\right)_+ \cdot \mathrm{sign}(\widehat{w}_i(OLS)),$$

The optimization problem for **Elastic Net** is

$$F(\mathbf{w}, b) = \lambda \left((1-\alpha)\|\mathbf{w}\|_1 + \alpha\|\mathbf{w}\|_2^2\right) + \sum_{i=1}^{m}(\mathbf{w}\cdot\mathbf{x}_i^\top + b - y_i)^2 \to \min_{\mathbf{w}, b},$$

- **Elastic Net** combines l_1 and l_2 penalties
- Ridge regression: $\alpha \to 1$; LASSO: $\alpha \to 0$
- It can be proved that in case of orthogonal design matrix ($\mathbf{X}^\top\mathbf{X} = \mathrm{I}$) analytical solution exists

$$\begin{aligned} \widehat{w}_i(Ridge) &= \widehat{w}_i(OLS)/(1+\lambda_2), \\ \widehat{w}_i(LASSO) &= (|\widehat{w}_i(OLS)| - \lambda_1/2)_+ \cdot \mathrm{sign}(\widehat{w}_i(OLS)), \\ \widehat{w}_i(EN) &= \frac{(|\widehat{w}_i(OLS)| - \lambda_1/2)_+}{1+\lambda_2} \cdot \mathrm{sign}(\widehat{w}_i(OLS)) \end{aligned}$$

where $(x)_+ = \max(x, 0)$

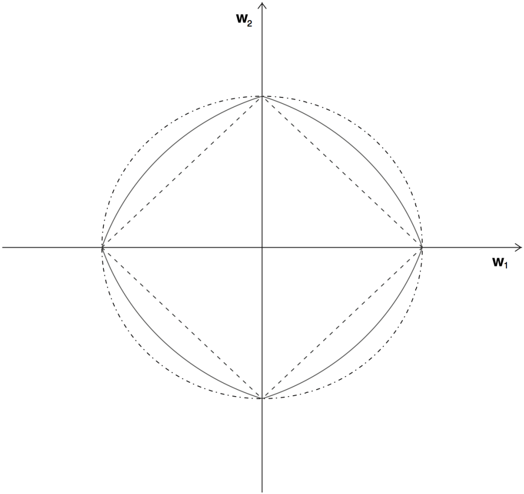


Figure – Two-dimensional contour plots of the penalty ($\cdot - \cdot - \cdot -$, shape of the ridge penalty; $- - -$, contour of the lasso penalty; ——, contour of the elastic net penalty with $\alpha = 0.5$): we see that singularities at the vertices and the edges are strictly convex; the strength of convexity varies with α [Hui Zou, Trevor Hastie]

- It can be proved that in Elastic Net, highly correlated predictors will have similar regression coefficients
- Given data $\{\mathbf{X}, \mathbf{y}\}$ and parameters (λ, α), the response $\mathbf{y}$ is centred and the predictors $\mathbf{X}$ are standardized
- Let $\widehat{\mathbf{w}}(\lambda, \alpha)$ be the elastic net estimate. Suppose that $\widehat{w}_i(\lambda, \alpha)\widehat{w}_j(\lambda, \alpha) > 0$
- Define
$$D_{(\lambda,\alpha)}(i,j) = \frac{1}{\|\mathbf{y}\|_1}|\widehat{w}_i(\lambda, \alpha) - \widehat{w}_j(\lambda, \alpha)|,$$
then
$$D_{(\lambda,\alpha)}(i,j) \leq \frac{1}{\lambda\alpha}\sqrt{\{2(1-\rho)\}},$$
where ρ is the sample correlation between the i-th and j-th columns of the matrix $\mathbf{X}$
- This theorem provides a quantitative description for the grouping effect of Elastic Net

- It can be proved that in Elastic Net, highly correlated predictors will have similar regression coefficients
- Given data $\{\mathbf{X}, \mathbf{y}\}$ and parameters (λ, α), the response $\mathbf{y}$ is centred and the predictors $\mathbf{X}$ are standardized
- Let $\widehat{\mathbf{w}}(\lambda, \alpha)$ be the elastic net estimate. Suppose that $\widehat{w}_i(\lambda, \alpha)\widehat{w}_j(\lambda, \alpha) > 0$
- Define
$$D_{(\lambda,\alpha)}(i,j) = \frac{1}{\|\mathbf{y}\|_1} |\widehat{w}_i(\lambda, \alpha) - \widehat{w}_j(\lambda, \alpha)|,$$
then
$$D_{(\lambda,\alpha)}(i,j) \leq \frac{1}{\lambda\alpha}\sqrt{\{2(1-\rho)\}},$$
where ρ is the sample correlation between the i-th and j-th columns of the matrix $\mathbf{X}$
- This theorem provides a quantitative description for the grouping effect of Elastic Net

- It can be proved that in Elastic Net, highly correlated predictors will have similar regression coefficients
- Given data $\{\mathbf{X}, \mathbf{y}\}$ and parameters (λ, α), the response $\mathbf{y}$ is centred and the predictors $\mathbf{X}$ are standardized
- Let $\widehat{\mathbf{w}}(\lambda, \alpha)$ be the elastic net estimate. Suppose that $\widehat{w}_i(\lambda, \alpha)\widehat{w}_j(\lambda, \alpha) > 0$
- Define
$$D_{(\lambda,\alpha)}(i,j) = \frac{1}{\|\mathbf{y}\|_1} |\widehat{w}_i(\lambda, \alpha) - \widehat{w}_j(\lambda, \alpha)|,$$
then
$$D_{(\lambda,\alpha)}(i,j) \leq \frac{1}{\lambda\alpha} \sqrt{\{2(1-\rho)\}},$$
where ρ is the sample correlation between the i-th and j-th columns of the matrix $\mathbf{X}$
- This theorem provides a quantitative description for the grouping effect of Elastic Net

- It can be proved that in Elastic Net, highly correlated predictors will have similar regression coefficients
- Given data $\{\mathbf{X}, \mathbf{y}\}$ and parameters (λ, α), the response $\mathbf{y}$ is centred and the predictors $\mathbf{X}$ are standardized
- Let $\widehat{\mathbf{w}}(\lambda, \alpha)$ be the elastic net estimate. Suppose that $\widehat{w}_i(\lambda, \alpha)\widehat{w}_j(\lambda, \alpha) > 0$
- Define
$$D_{(\lambda,\alpha)}(i, j) = \frac{1}{\|\mathbf{y}\|_1} |\widehat{w}_i(\lambda, \alpha) - \widehat{w}_j(\lambda, \alpha)|,$$
then
$$D_{(\lambda,\alpha)}(i, j) \leq \frac{1}{\lambda\alpha}\sqrt{\{2(1-\rho)\}},$$
where ρ is the sample correlation between the i-th and j-th columns of the matrix $\mathbf{X}$
- This theorem provides a quantitative description for the grouping effect of Elastic Net

- It can be proved that in Elastic Net, highly correlated predictors will have similar regression coefficients
- Given data $\{\mathbf{X}, \mathbf{y}\}$ and parameters (λ, α), the response $\mathbf{y}$ is centred and the predictors $\mathbf{X}$ are standardized
- Let $\widehat{\mathbf{w}}(\lambda, \alpha)$ be the elastic net estimate. Suppose that $\widehat{w}_i(\lambda, \alpha)\widehat{w}_j(\lambda, \alpha) > 0$
- Define
$$D_{(\lambda,\alpha)}(i,j) = \frac{1}{\|\mathbf{y}\|_1}|\widehat{w}_i(\lambda, \alpha) - \widehat{w}_j(\lambda, \alpha)|,$$
then
$$D_{(\lambda,\alpha)}(i,j) \leq \frac{1}{\lambda\alpha}\sqrt{\{2(1-\rho)\}},$$
where ρ is the sample correlation between the i-th and j-th columns of the matrix $\mathbf{X}$
- This theorem provides a quantitative description for the grouping effect of Elastic Net

- Elastic Net produces a sparse model with good prediction accuracy, while encouraging a grouping effect
- Efficient computation algorithm for Elastic Net is derived based on LARS
- Empirical results and simulations demonstrate its superiority over LASSO (LASSO can be viewed as a special case of Elastic Net)
- For Elastic Net, two parameters should be tuned/selected on training and validation data set. For LASSO, there is only one tuning parameter
- Although Elastic Net is proposed with the regression model, it can also be extend to classification problems (such as gene selection)

- Elastic Net produces a sparse model with good prediction accuracy, while encouraging a grouping effect
- Efficient computation algorithm for Elastic Net is derived based on LARS
- Empirical results and simulations demonstrate its superiority over LASSO (LASSO can be viewed as a special case of Elastic Net)
- For Elastic Net, two parameters should be tuned/selected on training and validation data set. For LASSO, there is only one tuning parameter
- Although Elastic Net is proposed with the regression model, it can also be extend to classification problems (such as gene selection)

- Elastic Net produces a sparse model with good prediction accuracy, while encouraging a grouping effect
- Efficient computation algorithm for Elastic Net is derived based on LARS
- Empirical results and simulations demonstrate its superiority over LASSO (LASSO can be viewed as a special case of Elastic Net)
- For Elastic Net, two parameters should be tuned/selected on training and validation data set. For LASSO, there is only one tuning parameter
- Although Elastic Net is proposed with the regression model, it can also be extend to classification problems (such as gene selection)

- Elastic Net produces a sparse model with good prediction accuracy, while encouraging a grouping effect
- Efficient computation algorithm for Elastic Net is derived based on LARS
- Empirical results and simulations demonstrate its superiority over LASSO (LASSO can be viewed as a special case of Elastic Net)
- For Elastic Net, two parameters should be tuned/selected on training and validation data set. For LASSO, there is only one tuning parameter
- Although Elastic Net is proposed with the regression model, it can also be extend to classification problems (such as gene selection)

- Elastic Net produces a sparse model with good prediction accuracy, while encouraging a grouping effect
- Efficient computation algorithm for Elastic Net is derived based on LARS
- Empirical results and simulations demonstrate its superiority over LASSO (LASSO can be viewed as a special case of Elastic Net)
- For Elastic Net, two parameters should be tuned/selected on training and validation data set. For LASSO, there is only one tuning parameter
- Although Elastic Net is proposed with the regression model, it can also be extend to classification problems (such as gene selection)