

CS489/698 Machine Learning

Lecture 3: Jan 10, 2018

Linear Regression

[RN] Sec. 18.6.1, [HTF] Sec. 2.3.1,
[D] Sec. 7.6, [B] Sec. 3.1, [M] Sec. 1.4.5

Linear model for regression

- Simple form of regression
- Picture:

Problem

- Data: $\{(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \dots, (\mathbf{x}_N, t_N)\}$
 - $\mathbf{x} = \langle x_1, x_2, \dots, x_M \rangle$: input vector
 - t : target (continuous value)
- Problem: find hypothesis h that maps \mathbf{x} to t
 - Assume that h is linear:

$$y(\mathbf{x}, \mathbf{w}) = w_0 + w_1 x_1 + \dots + w_M x_M = \mathbf{w}^T \begin{pmatrix} 1 \\ \mathbf{x} \end{pmatrix}$$

- Objective: minimize some loss function
 - Euclidean loss: $L_2(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N (y(\mathbf{x}_n, \mathbf{w}) - t_n)^2$

Optimization

- Find best w that minimizes Euclidean loss

$$\mathbf{w}^* = \operatorname{argmin}_{\mathbf{w}} \frac{1}{2} \sum_{n=1}^N \left(t_n - \mathbf{w}^T \begin{pmatrix} 1 \\ \mathbf{x}_n \end{pmatrix} \right)^2$$

- Convex optimization problem
 \Rightarrow unique optimum (global)

Solution

- Let $\bar{\mathbf{x}} = \begin{pmatrix} 1 \\ \mathbf{x} \end{pmatrix}$ then $\min_{\mathbf{w}} \frac{1}{2} \sum_{n=1}^N (t_n - \mathbf{w}^T \bar{\mathbf{x}}_n)^2$
- Find \mathbf{w}^* by setting the derivative to 0

$$\frac{\partial L_2}{\partial w_j} = \sum_{n=1}^N (t_n - \mathbf{w}^T \bar{\mathbf{x}}_n) \bar{x}_{nj} = 0 \quad \forall j$$

$$\Rightarrow \sum_{n=1}^N (t_n - \mathbf{w}^T \bar{\mathbf{x}}_n) \bar{\mathbf{x}}_n = 0$$

- This is a linear system in \mathbf{w} , therefore we rewrite it as $\mathbf{A}\mathbf{w} = \mathbf{b}$

$$\text{where } \mathbf{A} = \sum_{n=1}^N \bar{\mathbf{x}}_n \bar{\mathbf{x}}_n^T \text{ and } \mathbf{b} = \sum_{n=1}^N t_n \bar{\mathbf{x}}_n$$

Solution

- If training instances span \Re^{M+1} then \mathbf{A} is invertible:

$$\mathbf{w} = \mathbf{A}^{-1}\mathbf{b}$$

- In practice it is faster to solve the linear system $\mathbf{A}\mathbf{w} = \mathbf{b}$ directly instead of inverting \mathbf{A}
 - Gaussian elimination
 - Conjugate gradient
 - Iterative methods

Picture

Regularization

- Least square solution may not be stable
 - i.e., slight perturbation of the input may cause a dramatic change in the output
 - Form of **overfitting**

Example 1

- Training data: $\bar{\mathbf{x}}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ $\bar{\mathbf{x}}_2 = \begin{pmatrix} 1 \\ \epsilon \end{pmatrix}$
 $t_1 = 1$ $t_2 = 1$

- $\mathbf{A} =$

- $\mathbf{A}^{-1} =$ $\mathbf{b} =$

- $\mathbf{w} =$

Example 2

- Training data: $\bar{\mathbf{x}}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ $\bar{\mathbf{x}}_2 = \begin{pmatrix} 1 \\ \epsilon \end{pmatrix}$
 $t_1 = 1 + \epsilon$ $t_2 = 1$

- $\mathbf{A} =$

- $\mathbf{A}^{-1} =$ $\mathbf{b} =$

- $\mathbf{w} =$

Picture

Regularization

- Idea: favor smaller values
- Tikhonov regularization: add $||\mathbf{w}||_2^2$ as a penalty term
- Ridge regression:

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{2} \sum_{n=1}^N (t_n - \mathbf{w}^T \bar{\mathbf{x}}_n)^2 + \frac{\lambda}{2} ||\mathbf{w}||_2^2$$

where λ is a weight to adjust the importance of the penalty

Regularization

- Solution: $(\lambda I + A)\mathbf{w} = \mathbf{b}$
- Notes
 - Without regularization: eigenvalues of linear system may be arbitrarily close to 0 and the inverse may have arbitrarily large eigenvalues.
 - With Tikhonov regularization, eigenvalues of linear system are $\geq \lambda$ and therefore bounded away from 0. Similarly, eigenvalues of inverse are bounded above by $1/\lambda$.

Regularized Examples

Example 1

Example 2