EE2211 Pre-Tutorial 6

Dr Feng LIN

feng_lin@nus.edu.sg

Agenda

- Recap
- Self-learning
- Tutorial 6

Today's Attendance

Recap

- Linear Classification
  - Binary classification
  - Multi-category classification
- Ridge regression
  - Penalty term
  - Primal and dual forms
- Polynomial Regression
  - Nonlinear decision boundary

Linear Regression

Collect Data → Model and Loss Function → Learning/Training: compute w → Prediction /Testing: Compute y_{new}

$$Xw = y$$

- X: Samples
- y: Target values

$$\frac{1}{m}\sum_{i=1}^{m}(f_{\mathbf{w},b}(\mathbf{x}_i) - y_i)^2$$

- Linear or Affine function
- Squared error loss function

$$\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

- Check the invertibility
- Least square approximation (left-inverse)

$$\hat{f}_{\mathbf{w}}(\mathbf{X}_{new}) = \mathbf{X}_{new}\hat{\mathbf{w}}$$

- Prediction for new inputs
- Testing: Mean Squared Error (MSE)

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Linear Classification

Linear Methods for Classification

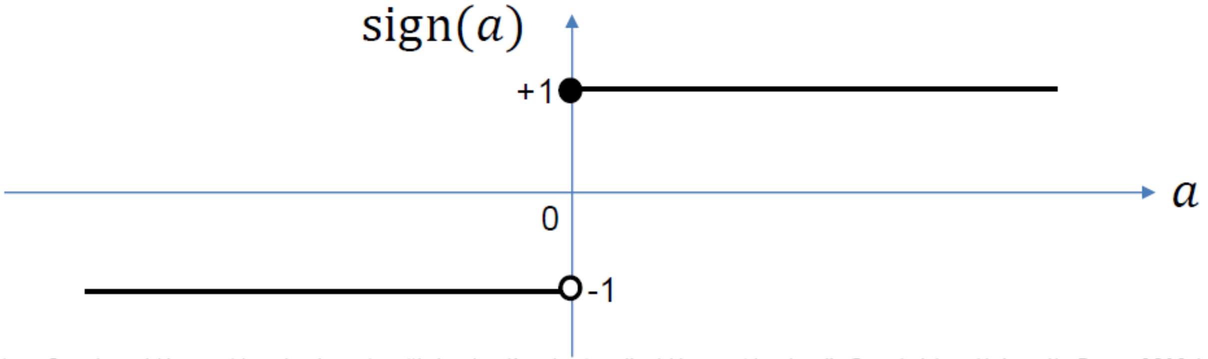
Binary Classification:

If $\mathbf{X}^T\mathbf{X}$ is invertible, then

Learning: $\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$, $\quad y_i \in \{-1, +1\}$, $i = 1, \ldots, m$

Prediction: $\hat{f}^c_{\mathbf{w}}(\mathbf{x}_{new}) = \mathrm{sign}(\mathbf{x}_{new}^T\hat{\mathbf{w}})$ for each row $\mathbf{x}_{new}^T$ of $\mathbf{X}_{new}$

$$\mathbf{sign}(a) = +1 \text{ for } a \geq 0 \text{ and } -1 \text{ for } a < 0$$



Ref: [Book4] Stephen Boyd and Lieven Vandenberghe, "Introduction to Applied Linear Algebra", Cambridge University Press, 2018 (chp.14)

Linear Classification

Linear Methods for Classification

Multi-Category Classification:

If $\mathbf{X}^T\mathbf{X}$ is invertible, then

Learning: $\widehat{\mathbf{W}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}, \quad \mathbf{Y} \in \mathbf{R}^{m\times C}$

Prediction: $\hat{f}^c_{\mathbf{w}}(\mathbf{x}_{new}) = \arg\max_{k=1,\dots,C}\left(\mathbf{x}^T_{new}\widehat{\mathbf{W}}(:,k)\right)$ for each $\mathbf{x}^T_{new}$ of $\mathbf{X}_{new}$

Each row (of $i = 1, \dots, m$) in $\mathbf{Y}$ has an **one-hot** encoding/assignment:

e.g., target for class-1 is labelled as $\mathbf{y}_i^T = [1, 0, 0, \dots, 0]$ for the ith sample,

target for class-2 is labelled as $\mathbf{y}_j^T = [0, 1, 0, \dots, 0]$ for the jth sample,

target for class-C is labelled as $\mathbf{y}_m^T = \underbrace{[0, 0, \dots, 0, 1]}_{C}$ for the mth sample.

Ref: Hastie, Tibshirani, Friedman, "The Elements of Statistical Learning", (2nd ed., 12th printing) 2017 (chp.4)

Ridge Regression

Recall Linear regression

Objective: $\widehat{\mathbf{w}} = \text{argmin} \sum_{i=1}^{m} (f_{\mathbf{w}}(\mathbf{x}_i) - y_i)^2 = (\mathbf{Xw} - \mathbf{y})^T(\mathbf{Xw} - \mathbf{y})$

The learning computation: $\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$

We cannot guarantee that the matrix $\mathbf{X}^T\mathbf{X}$ is invertible

Ridge regression: shrinks the regression coefficients *w* by imposing a penalty on their size

Objective: $\widehat{\mathbf{w}} = \text{argmin} \sum_{i=1}^{m} (f_{\mathbf{w}}(\mathbf{x}_i) - y_i)^2 + \lambda \sum_{j=1}^{d} w_j^2$

$= \text{argmin}\ (\mathbf{Xw} - \mathbf{y})^T(\mathbf{Xw} - \mathbf{y}) + \lambda \mathbf{w}^T\mathbf{w}$

Here $\lambda \geq 0$ is a complexity parameter that controls the amount of shrinkage: the larger the value of λ, the greater the amount of shrinkage.

Note: *m* samples & *d* parameters

Ridge Regression

The learning computation: $\hat{\boldsymbol{w}} = (X^T X)^{-1} X^T y$

$$(X^T X)^{-1} = \frac{1}{|X^T X|}(X^T X)^* \rightarrow \frac{1}{0}(X^T X)^* \Rightarrow \hat{\boldsymbol{w}} \rightarrow \infty$$

If $X^T X$ is not invertible, that means its determination is 0. This causes the denominator of $(X^T X)^{-1}$ to approach 0, which in turn causes w to approach infinity, making it impossible to fit the data well.

Ridge regression: shrinks the regression coefficients w by impose penalty on their size

Objective: $\hat{\mathbf{w}} = \text{argmin} \sum_{i=1}^{m} (f_{\mathbf{w}}(\mathbf{x}_i) - \mathrm{y}_i)^2 + \lambda \sum_{j=1}^{d} w_j^2$

$= \text{argmin}\ (\mathbf{Xw} - \mathbf{y})^T(\mathbf{Xw} - \mathbf{y}) + \lambda \mathbf{w}^T \mathbf{w}$

$\lambda \|\boldsymbol{w}\|^2$

Regularization or penalty term or ridge term

Ridge Regression

Using a linear model:

$$\min_{\mathbf{w}} (\mathbf{X}\mathbf{w} - \mathbf{y})^T (\mathbf{X}\mathbf{w} - \mathbf{y}) + \lambda \mathbf{w}^T \mathbf{w}$$

Solution:

$$\frac{\partial}{\partial \mathbf{w}} \left((\mathbf{X}\mathbf{w} - \mathbf{y})^T (\mathbf{X}\mathbf{w} - \mathbf{y}) + \lambda \mathbf{w}^T \mathbf{w} \right) = \mathbf{0}$$

$$\Rightarrow 2\mathbf{X}^T\mathbf{X}\mathbf{w} - 2\mathbf{X}^T\mathbf{y} + 2\lambda\mathbf{w} = \mathbf{0}$$

$$\Rightarrow \mathbf{X}^T\mathbf{X}\mathbf{w} + \lambda\mathbf{w} = \mathbf{X}^T\mathbf{y}$$

$$\Rightarrow (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})\mathbf{w} = \mathbf{X}^T\mathbf{y}$$

where **I** is the *dxd* identity matrix

Here on, we shall focus on single column of output $\mathbf{y}$ in derivations in the sequel

Learning: $\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1} \mathbf{X}^T\mathbf{y}$

Ref: Hastie, Tibshirani, Friedman, "The Elements of Statistical Learning", (2nd ed., 12th printing) 2017 (chp.3)

Ridge Regression

Ridge Regression in Primal Form (when m > d)

$(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})$ is invertible for $\lambda > 0$,

Learning: $\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\,\mathbf{X}^T\mathbf{y}$

Prediction: $\hat{f}_{\mathbf{w}}(\mathbf{X}_{new}) = \mathbf{X}_{new}\hat{\mathbf{w}}$

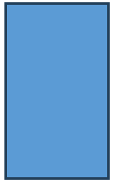
Ridge Regression in Dual Form (when m < d)

$(\mathbf{X}\mathbf{X}^T + \lambda\mathbf{I})$ is invertible for $\lambda > 0$,

Learning: $\hat{\mathbf{w}} = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \lambda\mathbf{I})^{-1}\,\mathbf{y}$

Prediction: $\hat{f}_{\mathbf{w}}(\mathbf{X}_{new}) = \mathbf{X}_{new}\hat{\mathbf{w}}$

Linear and Ridge Regression



	Linear Regression	Ridge Regression
Over-determined system $(m > d)$	Left inverse $\hat{\boldsymbol{w}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$	Primal Form $\hat{\boldsymbol{w}} = \left(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$
Under-determined system $(m < d)$	Right inverse $\hat{\boldsymbol{w}} = \boldsymbol{X}^T\left(\boldsymbol{X}\boldsymbol{X}^T\right)^{-1}\boldsymbol{y}$	Dual Form $\hat{\boldsymbol{w}} = \boldsymbol{X}^T\left(\boldsymbol{X}\boldsymbol{X}^T + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{y}$

Noted: 1) The primal form can be used to solve under-determined system, but it is better suited for over-determined system. 2) The dual form of ridge regression is often more computationally efficient in under-determined system than the primal form.

Polynomial Regression

Motivation: nonlinear decision surface

- Based on the sum of products of the variables
- E.g. when the input dimension is $d=2$,

a polynomial function of degree = 2 is:

$$f_{\mathbf{w}}(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + w_{12} x_1 x_2 + w_{11} x_1^2 + w_{22} x_2^2 .$$

XOR problem

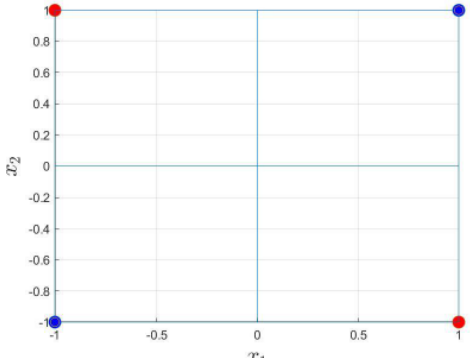
$\mathbf{x}_1 = [+1 \quad +1]^\top \qquad y_1 = +1$

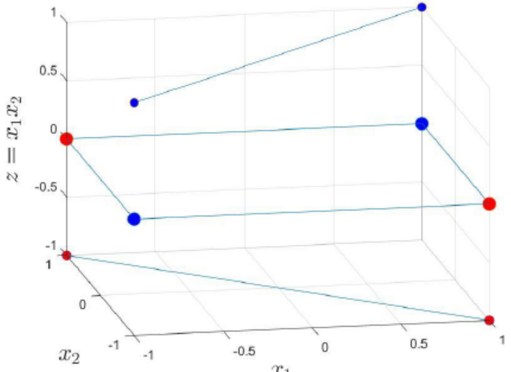
$\mathbf{x}_2 = [-1 \quad +1]^\top \qquad y_2 = -1$

$\mathbf{x}_3 = [+1 \quad -1]^\top \qquad y_3 = -1$

$\mathbf{x}_4 = [-1 \quad -1]^\top \qquad y_4 = +1$

$$f_{\mathbf{w}}(\mathbf{x}) = x_1 x_2$$





Polynomial Regression

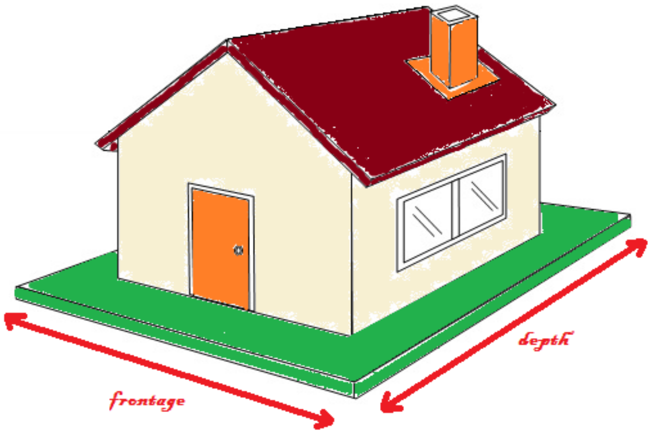
Motivation: Nonlinear Prediction

E.g. predicting the price of the house. Suppose you have two features:

- x_1: the frontage of house (the width of the property)
- x_2: the depth of the house.

We might build a linear regression model like this

$$f_{\mathbf{w}}(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2$$



If we want to predict house prices, we might focus on the house or land area as key factors and create a new feature accordingly.

$$f_{\mathbf{w}}(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + w_{12} x_1 x_2$$

Aera

Polynomial Regression

Polynomial Expansion

- The linear model $f_{\mathbf{w}}(\mathbf{x}) = \mathbf{x}^T\mathbf{w}$ can be written as

$$f_{\mathbf{w}}(\mathbf{x}) = \mathbf{x}^T\mathbf{w}$$
$$= \sum_{i=0}^{d} x_i w_i \,, \qquad x_0 = 1$$
$$= w_0 + \sum_{i=1}^{d} x_i w_i \,.$$

- By including additional terms involving the products of pairs of components of $\mathbf{x}$, we obtain a quadratic model:

$$f_{\mathbf{w}}(\mathbf{x}) = w_0 + \sum_{i=1}^{d} w_i x_i + \sum_{i=1}^{d}\sum_{j=1}^{d} w_{ij} x_i x_j \,.$$

2nd order: $f_{\mathbf{w}}(\mathbf{x}) = w_0 + w_1x_1 + w_2x_2 + w_{12}x_1x_2 + w_{11}x_1^2 + w_{22}x_2^2$

3rd order: $f_{\mathbf{w}}(\mathbf{x}) = w_0 + w_1x_1 + w_2x_2 + w_{12}x_1x_2 + w_{11}x_1^2 + w_{22}x_2^2 + \sum_{i=1}^{d}\sum_{j=1}^{d}\sum_{k=1}^{d} w_{ijk}x_ix_jx_k, \; d = 2$

n: number of variables

r: number of order

$$C(n,r) = \frac{(n+r-1)!}{r!\,(n-1)!}$$

$$C(3,0) = \frac{(3+0-1)!}{0!\,(3-1)!} = 1$$

$$C(3,1) = \frac{(3+1-1)!}{1!\,(3-1)!} = 3$$

$$C(3,2) = \frac{(3+2-1)!}{2!\,(3-1)!} = 6$$

$$C(3,3) = \frac{(3+3-1)!}{3!\,(3-1)!} = 10$$

Ref: Duda, Hart, and Stork, "Pattern Classification", 2001 (Chp.5)

Polynomial Regression

Ridge Regression in Primal Form (m > d)

For $\lambda > 0$,

Learning: $$\hat{\mathbf{w}} = (\mathbf{P}^T\mathbf{P} + \lambda\mathbf{I})^{-1}\,\mathbf{P}^T\mathbf{y}$$

Prediction: $$\hat{f}_{\mathbf{w}}(\mathbf{P}(\mathbf{X}_{new})) = \mathbf{P}_{new}\hat{\mathbf{w}}$$

Ridge Regression in Dual Form (m < d)

For $\lambda > 0$,

Learning: $$\hat{\mathbf{w}} = \mathbf{P}^T(\mathbf{P}\mathbf{P}^T + \lambda\mathbf{I})^{-1}\,\mathbf{y}$$

Prediction: $$\hat{f}_{\mathbf{w}}(\mathbf{P}(\mathbf{X}_{new})) = \mathbf{P}_{new}\hat{\mathbf{w}}$$

Note: Change **X** to **P** with reference to slides 15/16; m & d refers to the size of **P** (not **X**)

THANK YOU