STAT C161/261 Machine Learning

Ying Nian Wu ywu@stat.ucla.edu

OH: TR 2:30-3:20 PM, subject to change if not convenient for class

Coding in Python, PyTorch, ok if no background in Python

Zoom for 413 Lab: Wednesday 5-6 PM

Canvas Modules

- LaTeX notes will be uploaded
- Python coding manual

Topics:

linear regression, logistic regression (basics of machine learning)
classification

(basics: linear algebra, multivariate calculus)

regularization to avoid overfitting, L2 → ridge regression
L1 → lasso

Trees, random forest, boosting (adaboost, XG boost)
↳ extreme gradient boosting

kernels, regression, support vector machine (important for classification, used commonly prior to deep Learning)

deep learning, neural network, CNN, RNN (LSTM), Transformer
(more popular now that there is a lot of training data & advances in computing power)
→ long short term memory

Embedding, representative learning, generative models

Reinforcement Learning, Alpha Go (model based reinforcement learning method).
↓
"reward"

Homeworks: weekly/bi-weekly, consisting of written & coding parts (not writing code from scratch)

for second half of class:
Google Colab necessary for GPU usage.

Final Exam: final HW or project

linear regression

simplest:

	X (father height)	Y (son height)
1		
2		
⋮	x_i input	y_i output
n		

supervised

— training example

column vectors:

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{pmatrix} \quad Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_i \\ \vdots \\ y_n \end{pmatrix}$$

training data $\xrightarrow{\text{generalize}}$ testing data

$$y_i = x_i\beta + e_i, \quad i = 1, \dots, n$$

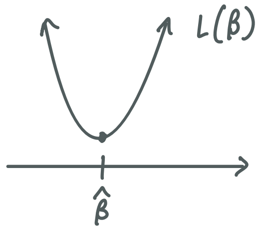
e_i ↓ error, residual

- no intercept
- 1 parameter β

Loss function (always start w/ well defined loss function, function of β)

$$L(\beta) = \sum_{i=1}^{n} (y_i - x_i\beta)^2$$

least squares loss



To find $\hat{\beta}$, for $\hat{\beta}$ to be a minimum, $L'(\beta) = 0$, $L''(\beta) = +$

$$L'(\beta) = -\sum_{i=1}^{n} (y_i - x_i\beta)x_i = 0 = \langle e, X \rangle$$

$$\sum_{i=1}^{n} y_i x_i = \sum_{i=1}^{n} x_i^2 \beta$$

$$\hat{\beta} = \frac{\sum_{i=1}^{n} x_i y_i \longrightarrow \langle Y, X \rangle}{\sum_{i=1}^{n} x_i^2 \longrightarrow |X|^2} = \left(\sum_{i=1}^{n} x_i x_i^T\right)^{-1} \left(\sum_{i=1}^{n} x_i y_i\right)$$

$$L''(\beta) = \sum_{i=1}^{n} x_i^2 \geq 0$$

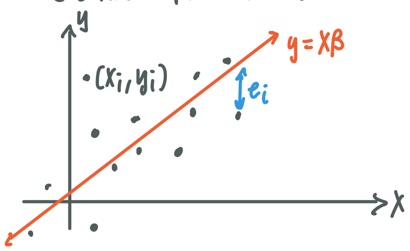
for any β, this holds.
here, $L(\beta)$ is a convex function w/ a global minimum.

$$L(\beta) = \frac{1}{2}\sum_{i=1}^{n} e_i^2 = \frac{1}{2}|e|^2 = \frac{1}{2}|Y - X\beta|^2$$

$$e_i = y_i - x_i$$

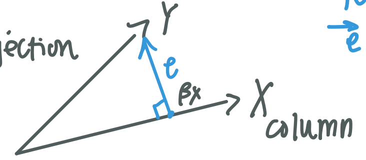
$$X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{pmatrix} \quad Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_i \\ \vdots \\ y_n \end{pmatrix} \quad e = \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_i \\ \vdots \\ e_n \end{pmatrix} = \begin{pmatrix} y_i - x_i\beta \end{pmatrix} = Y - X\beta$$

scatter plot (2-D)



vector plot (n-D)

projection



to minimize error, $\vec{e}$ should be perpendicular to $\vec{X}$

$$\langle e, X \rangle = 0 \Leftrightarrow e \perp X$$

$$\langle Y - X\beta, X \rangle = 0$$

$$\langle Y, X \rangle = \langle X, X \rangle \beta = 0$$

$$\hat{\beta} = \frac{\langle Y, X \rangle}{|X|^2} = (X^T X)^{-1} X^T Y$$

	1 2 ⋯ j ⋯ p	
1		
2		
⋮		
i	$x_{i1}, x_{i2}, \dots x_{ij}, \dots x_{ip}$	y_i
⋮		
n		
	X_j	Y

$$\underset{n \times p}{X}$$

$$x_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{ij} \\ \vdots \\ x_{ip} \end{pmatrix} \text{ row vector} \qquad \beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_j \\ \vdots \\ \beta_p \end{pmatrix}$$

$$X_j = \begin{pmatrix} x_{1j} \\ x_{2j} \\ \vdots \\ x_{ij} \\ \vdots \\ x_{nj} \end{pmatrix} \text{ column vector}$$

$$y_i = \sum_{j=1}^{n} x_{ij} \beta_j + e_i$$

$$Y = X_1 \beta_1 + X_2 \beta_2 + \dots + X_j \beta_j + \dots X_p \beta_p + e$$