CS249: ADVANCED DATA MINING

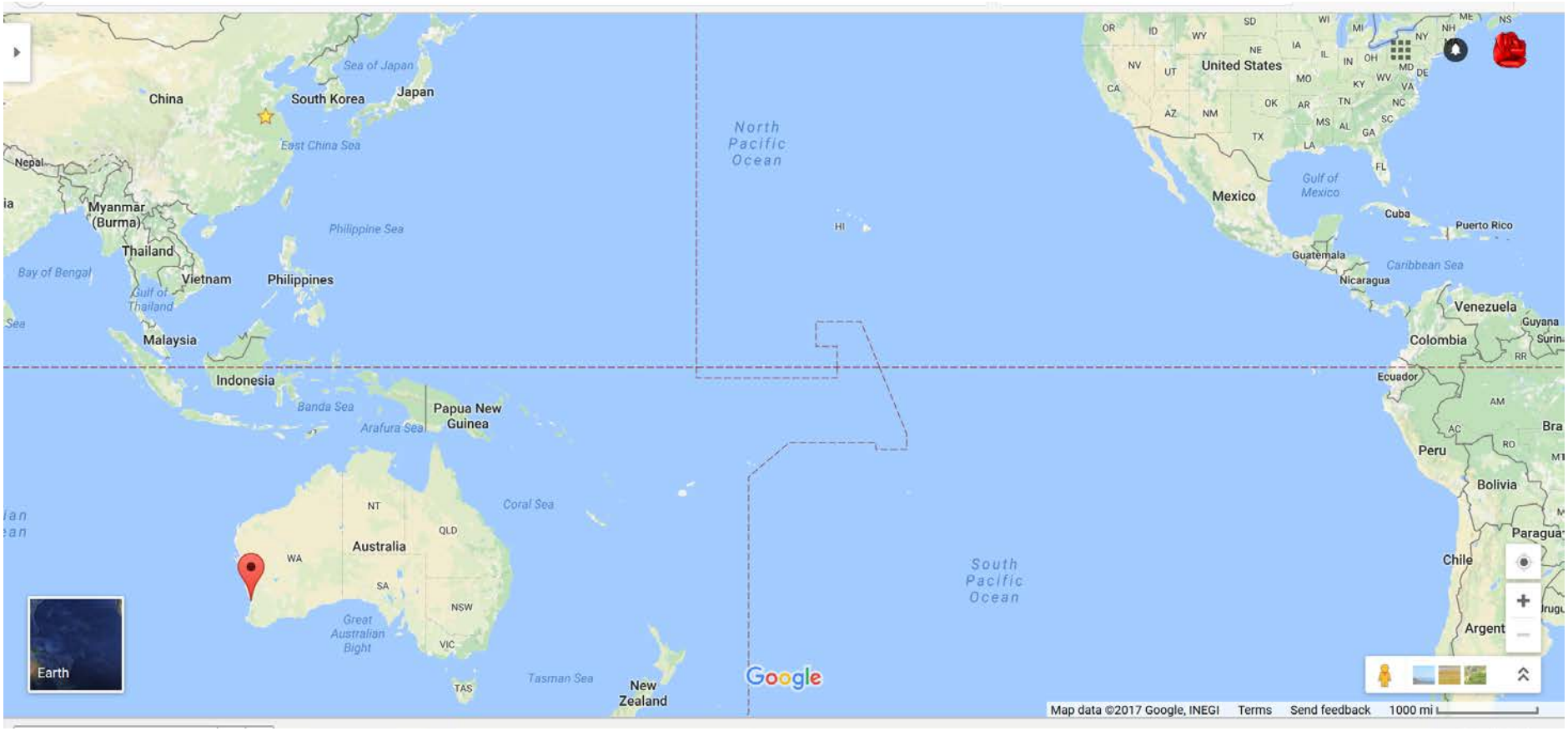
Linear Regression, Logistic Regression, and GLMs

Instructor: Yizhou Sun

yzsun@cs.ucla.edu

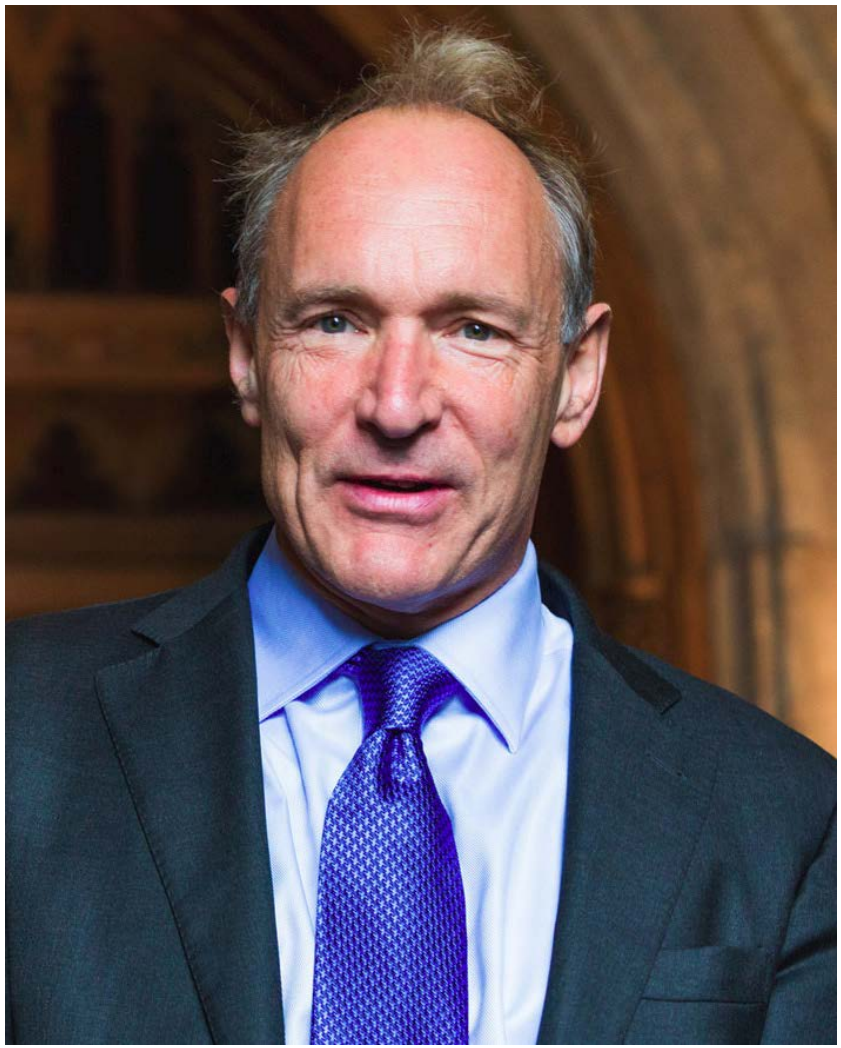
April 24, 2017

About WWW2017 Conference



Turing Award Winner

- Sir Tim Berners-Lee



Announcements

- Course Project
  - Team formation due this Wednesday
- PTE will be decided by this weekend
- Office hour for TA (Jae LEE):
  - 3-5 PM on Tuesdays @BH 2432

Methods to Learn

	Vector Data	Text Data	Recommender System	Graph & Network
Classification	Decision Tree; Naïve Bayes; **Logistic Regression** SVM; NN			Label Propagation
Clustering	K-means; hierarchical clustering; DBSCAN; Mixture Models; kernel k-means	PLSA; LDA	Matrix Factorization	SCAN; Spectral Clustering
Prediction	**Linear Regression GLM**		Collaborative Filtering	
Ranking				PageRank
Feature Representation		Word embedding		Network embedding

How to learn these algorithms?

- Three levels
  - When it is applicable?
    - Input, output, strengths, weaknesses, time complexity
  - How it works?
    - Pseudo-code, work flows, major steps
    - Can work out a toy problem by pen and paper
  - Why it works?
    - Intuition, philosophy, objective, derivation, proof

Vector Data: Regression

- Vector Data 
- Linear Regression Model
- Logistic Regression Model
- Generalized Linear Model
- Summary

Example

	Sex	Race	Height	Income	Marital Status	Years of Educ.	Liberal-ness
R1001	M	1	70	50	1	12	1.73
R1002	M	2	72	100	2	20	4.53
R1003	F	1	55	250	1	16	2.99
R1004	M	2	65	20	2	16	1.13
R1005	F	1	60	10	3	12	3.81
R1006	M	1	68	30	1	9	4.76
R1007	F	5	66	25	2	21	2.01
R1008	F	4	61	43	1	18	1.27
R1009	M	1	69	67	1	12	3.25

A matrix of n $\times$ p:

- n data objects / points
- p attributes / dimensions

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$

Attribute Type

- Numerical
  - E.g., height, income
- Categorical / discrete
  - E.g., Sex, Race

Categorical Attribute Types

- **Nominal:** categories, states, or "names of things"
  - *Hair_color* = {*auburn, black, blond, brown, grey, red, white*}
  - marital status, occupation, ID numbers, zip codes
- **Binary**
  - Nominal attribute with only 2 states (0 and 1)
  - Symmetric binary: both outcomes equally important
    - e.g., gender
  - Asymmetric binary: outcomes not equally important.
    - e.g., medical test (positive vs. negative)
    - Convention: assign 1 to most important outcome (e.g., HIV positive)
- **Ordinal**
  - Values have a meaningful order (ranking) but magnitude between successive values is not known.
  - *Size* = {*small, medium, large*}, grades, army rankings

Vector Data: Regression

- Vector Data
- Linear Regression Model 
- Logistic Regression Model
- Generalized Linear Model
- Summary

Linear Regression

- Ordinary Least Square Regression
  - Closed form solution
  - Gradient descent
- Linear Regression with Probabilistic Interpretation

The Linear Regression Problem

- Any Attributes to Continuous Value: **x** ⇒ y
  - {age; major ; gender; race} ⇒ GPA
  - {income; credit score; profession} ⇒ loan
  - {college; major ; GPA} ⇒ future income
  - …

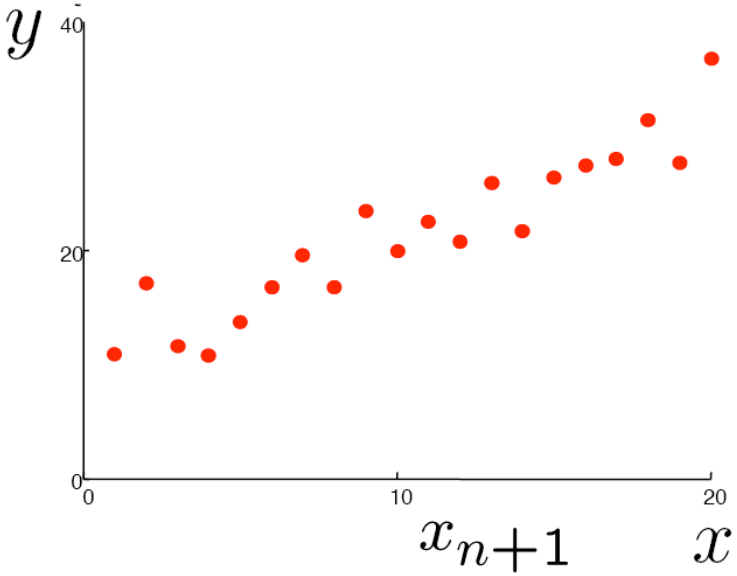
Example of House Price

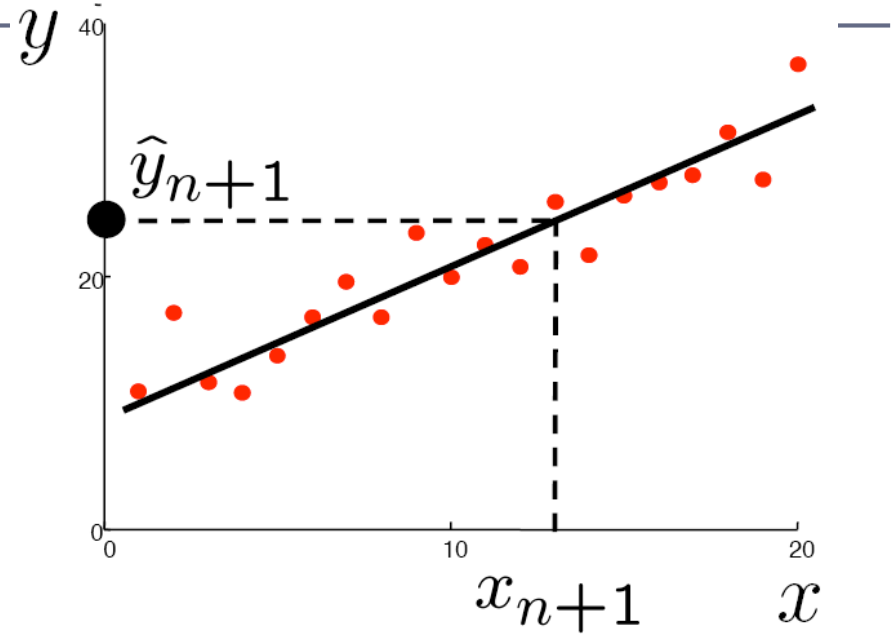
Living Area (sqft)	# of Beds	Price (1000$)
2104	3	400
1600	3	330
2400	3	369
1416	2	232
3000	4	540

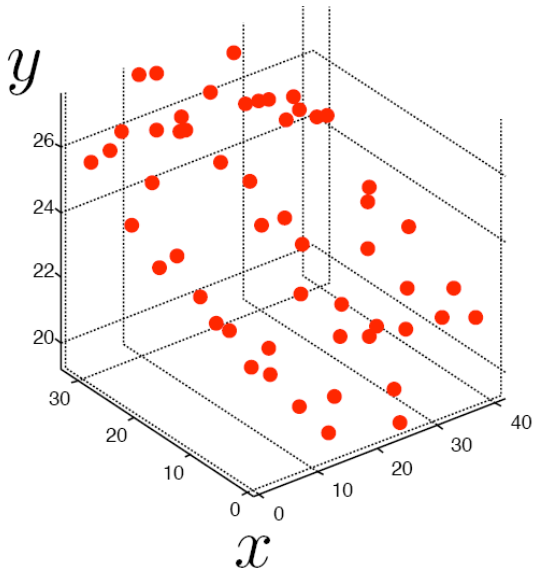
x

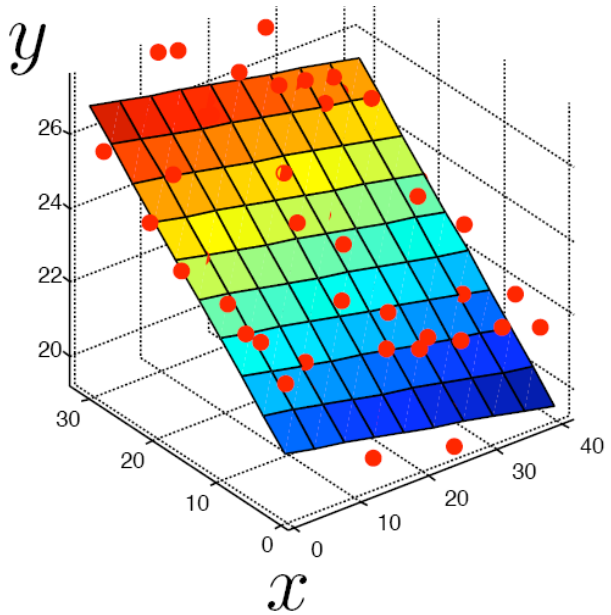
y

Illustration









Formalization

- Data: n independent data objects
  - $y_i, \mathrm{i} = 1, \dots, n$
  - $\boldsymbol{x}_i = \left(x_{i1}, x_{i2}, \dots, x_{ip}\right)^{\mathrm{T}}, \mathrm{i} = 1, \dots, n$
    - A constant factor is added to model the bias term, i. e. , $x_{i0} = 1$
    - New x: $\boldsymbol{x}_i = \left(x_{i0}, x_{i1}, x_{i2}, \dots, x_{ip}\right)^{\mathrm{T}}$
- Model:
  - y: *dependent variable*
  - $\boldsymbol{x}$: *explanatory variables*
  - $\boldsymbol{\beta} = \left(\beta_0, \beta_1, \dots, \beta_p\right)^T$: *weight vector*
  - $y = \boldsymbol{x}^T\boldsymbol{\beta} = \beta_0 + x_1\beta_1 + x_2\beta_2 + \cdots + x_p\beta_p$

A 3-step Process

- Model Construction
  - Use training data to find the best parameter $\boldsymbol{\beta}$, denoted as $\widehat{\boldsymbol{\beta}}$
- Model Selection
  - Use validation data to select the best model
    - E.g., Feature selection
- Model Usage
  - Apply the model to the unseen data (test data): $\hat{y} = \boldsymbol{x}^T \widehat{\boldsymbol{\beta}}$

Least Square Estimation

- Cost function (Total Square Error):
  - $J(\boldsymbol{\beta}) = \frac{1}{2}\sum_i \left(\boldsymbol{x}_i^T \boldsymbol{\beta} - y_i\right)^2$
- Matrix form:
  - $J(\boldsymbol{\beta}) = (\mathrm{X}\boldsymbol{\beta} - \boldsymbol{y})^T (X\boldsymbol{\beta} - \boldsymbol{y})/2$

$$or \left|\left|\mathrm{X}\boldsymbol{\beta} - \boldsymbol{y}\right|\right|^2/2$$

$$\begin{bmatrix} 1, x_{11} & \dots & x_{1f} & \dots & x_{1p} \\ \dots & \dots & \dots & \dots & \dots \\ 1, x_{i1} & \dots & x_{if} & \dots & x_{ip} \\ \dots & \dots & \dots & \dots & \dots \\ 1, x_{n1} & \dots & x_{nf} & \dots & x_{np} \end{bmatrix} \qquad \begin{pmatrix} y_1 \\ \vdots \\ y_i \\ \vdots \\ y_n \end{pmatrix}$$

$\boldsymbol{X}$: $\boldsymbol{n} \times (\boldsymbol{p}+\mathbf{1})$ matrix

y: $\boldsymbol{n} \times \mathbf{1}$ vector

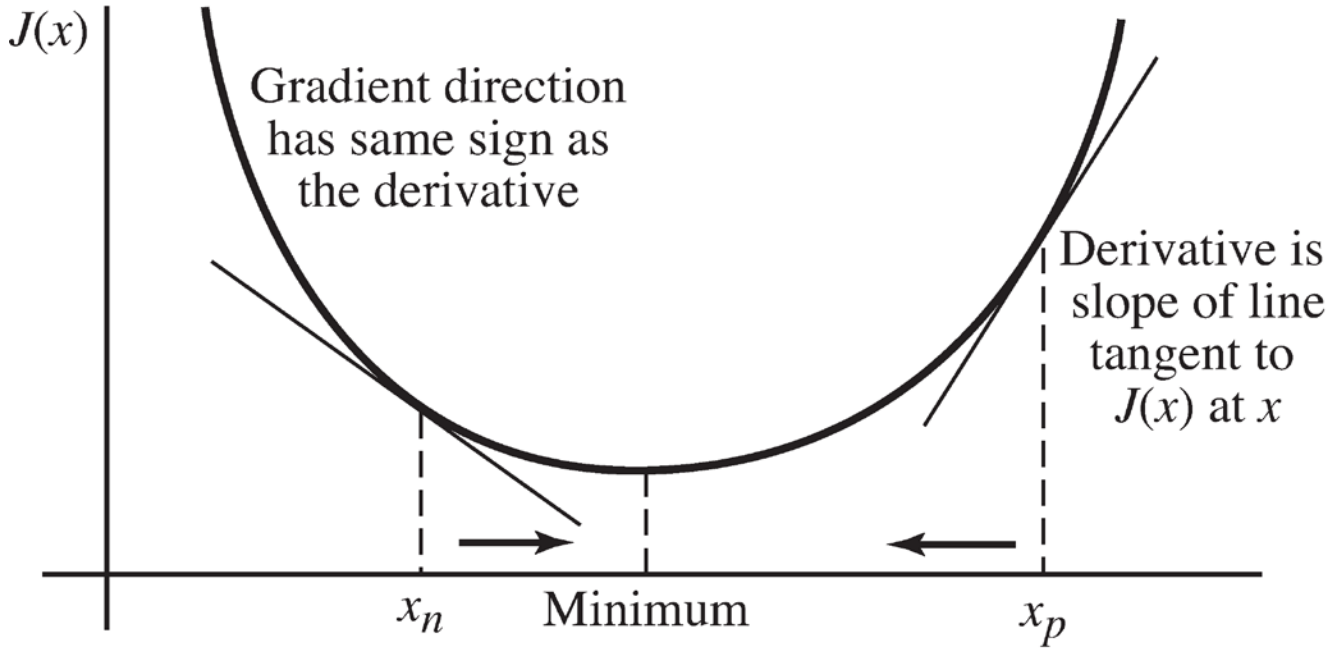
Ordinary Least Squares (OLS)

- Goal: find $\widehat{\boldsymbol{\beta}}$ that minimizes $J(\boldsymbol{\beta})$
  - $J(\boldsymbol{\beta}) = \frac{1}{2}(\mathrm{X}\boldsymbol{\beta} - y)^T(X\boldsymbol{\beta} - y)$

    $= \frac{1}{2}(\boldsymbol{\beta}^T X^T X\boldsymbol{\beta} - y^T X\boldsymbol{\beta} - \boldsymbol{\beta}^T X^T y + y^T y)$
- Ordinary least squares
  - Set first derivative of $J(\boldsymbol{\beta})$ as 0
    - $\frac{\partial J}{\partial \boldsymbol{\beta}} = \boldsymbol{\beta}^T \mathrm{X}^T \mathrm{X} - y^T X = 0$
    - $\Rightarrow \widehat{\boldsymbol{\beta}} = \left(X^T X\right)^{-1} X^T y$

Gradient Descent

- Minimize the cost function by moving down in the steepest direction



Arrows point in minus gradient direction towards the minimum

Batch Gradient Descent

- Move in the direction of steepest descend

Repeat until converge {

$$\boldsymbol{\beta}^{(t+1)} := \boldsymbol{\beta}^{(t)} - \eta \frac{\partial J}{\partial \boldsymbol{\beta}}\Big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(t)}}, \qquad \text{e.g., } \eta = 0.01$$

}

Where $J(\boldsymbol{\beta}) = \frac{1}{2}\sum_i \left(\boldsymbol{x}_i^T \boldsymbol{\beta} - y_i\right)^2 = \sum_i J_i(\boldsymbol{\beta})$ *and*

$$\frac{\partial J}{\partial \boldsymbol{\beta}} = \sum_i \frac{\partial J_i}{\partial \boldsymbol{\beta}} = \sum_i \boldsymbol{x}_i \left(\boldsymbol{x}_i^T \boldsymbol{\beta} - y_i\right)$$

Stochastic Gradient Descent

- When a new observation, *i*, comes in, update weight immediately (extremely useful for large-scale datasets):

Repeat {

for i=1:n {

$$\boldsymbol{\beta}^{(t+1)} := \boldsymbol{\beta}^{(t)} + \eta(y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}^{(t)})\boldsymbol{x}_i$$

}

}

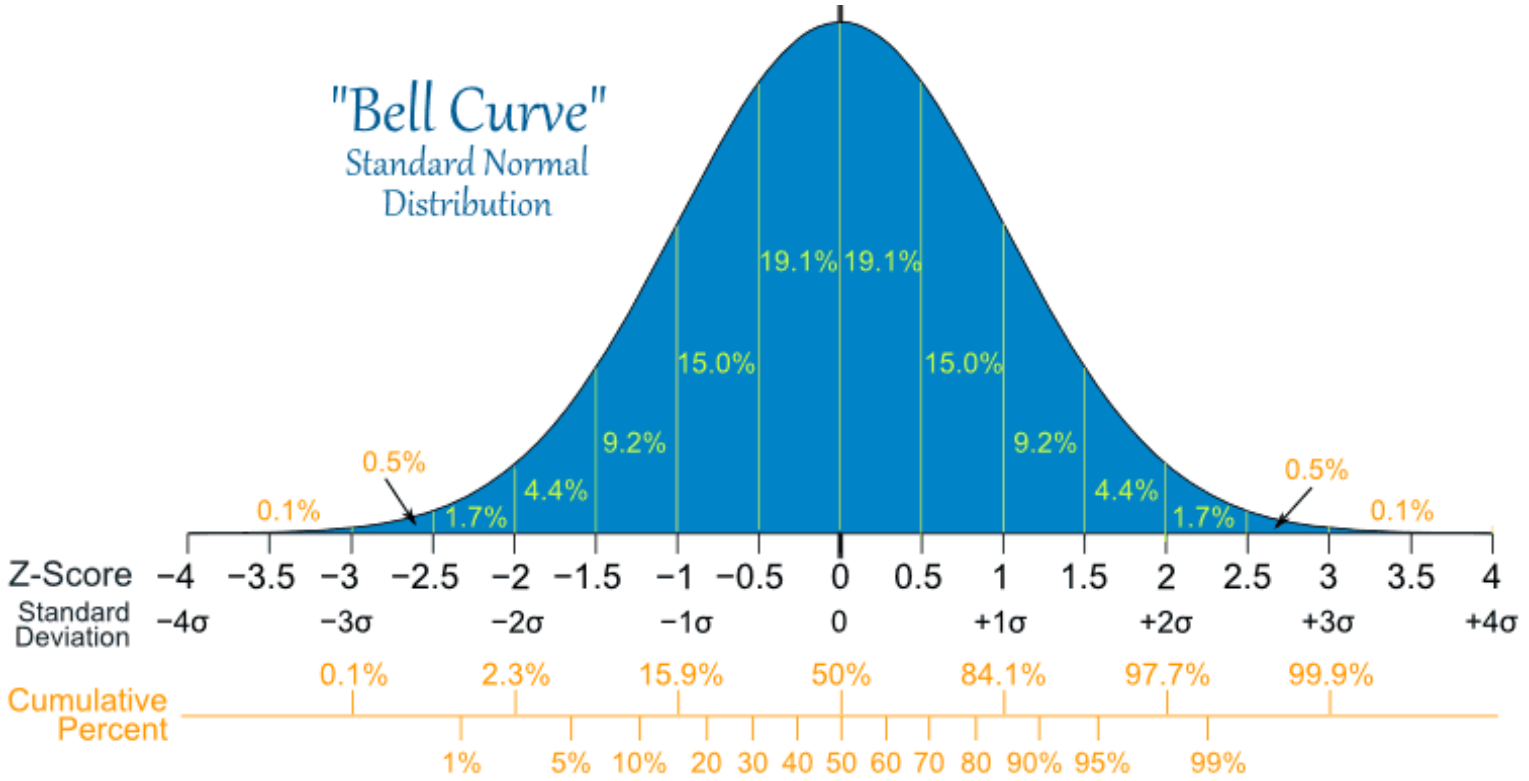
If the prediction for object *i* is smaller than the real value, $\boldsymbol{\beta}$ should move forward to the direction of $\boldsymbol{x}_i$

Other Practical Issues

- What if $X^T X$ is not invertible?
  - Add a small portion of identity matrix, λI, to it (ridge regression*) $\sum_i (y_i - \mathbf{x}_i^T \beta)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$
- What if some attributes are categorical?
  - Set dummy variables
    - E.g., $x = 1, if\ sex = F; x = 0, if\ sex = M$
    - Nominal variable with multiple values?
      - Create more dummy variables for one variable
- What if non-linear correlation exists?
  - Transform features, say, x to x^2

Probabilistic Interpretation

- Review of normal distribution
  - $X \sim N(\mu, \sigma^2) \Rightarrow f(X = x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$



Probabilistic Interpretation

- Model: $y_i = x_i^T \beta + \varepsilon_i$
  - $\varepsilon_i \sim N(0, \sigma^2)$
  - $y_i | x_i, \beta \sim N(x_i^T \beta, \sigma^2)$
    - $E(y_i | x_i, \beta) = x_i^T \beta$
- Likelihood:
  - $L(\boldsymbol{\beta}) = \prod_i p(y_i | x_i, \beta)$

$$= \prod_i \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{(y_i - \boldsymbol{x_i^T \beta})^2}{2\sigma^2}\}$$

- Maximum Likelihood Estimation
  - find $\widehat{\boldsymbol{\beta}}$ that maximizes $\mathrm{L}(\boldsymbol{\beta})$
  - $\arg\max L = \arg\min J$, Equivalent to OLS!

Vector Data: Regression

- Vector Data
- Linear Regression Model
- Logistic Regression Model 
- Generalized Linear Model
- Summary

Linear Regression VS. Logistic Regression

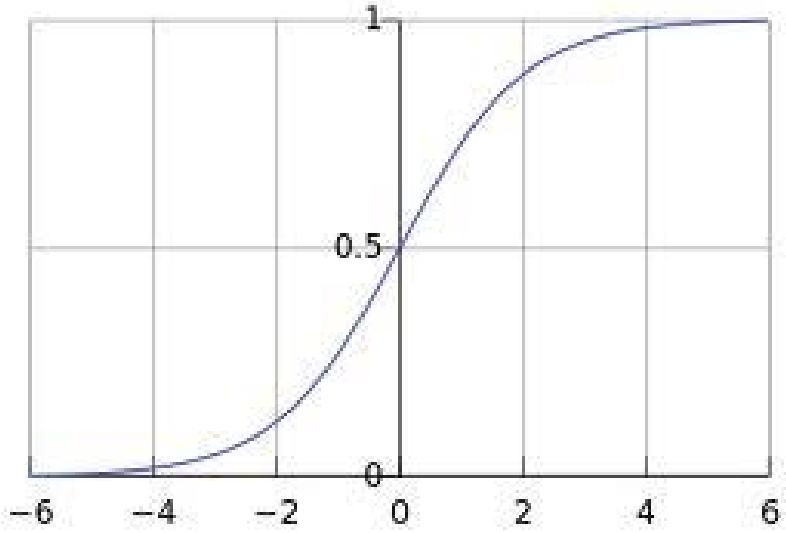
- Linear Regression (prediction)
  - Y: $continuous\ value\ (-\infty, +\infty)$
    - $Y = \boldsymbol{x}^T\boldsymbol{\beta} = \beta_0 + x_1\beta_1 + x_2\beta_2 + \cdots + x_p\beta_p$
    - $Y|\boldsymbol{x}, \beta \sim N(\boldsymbol{x}^T\beta, \sigma^2)$

- Logistic Regression (classification)
  - Y: $discrete\ value\ from\ m\ classes$
  - $p(Y = C_j) \in [0,1]\ and\ \sum_j p(Y = C_j) = 1$

Logistic Function

- Logistic Function / sigmoid function:

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



$$Note: \sigma'(x) = \sigma(x)(1 - \sigma(x))$$

Modeling Probabilities of Two Classes

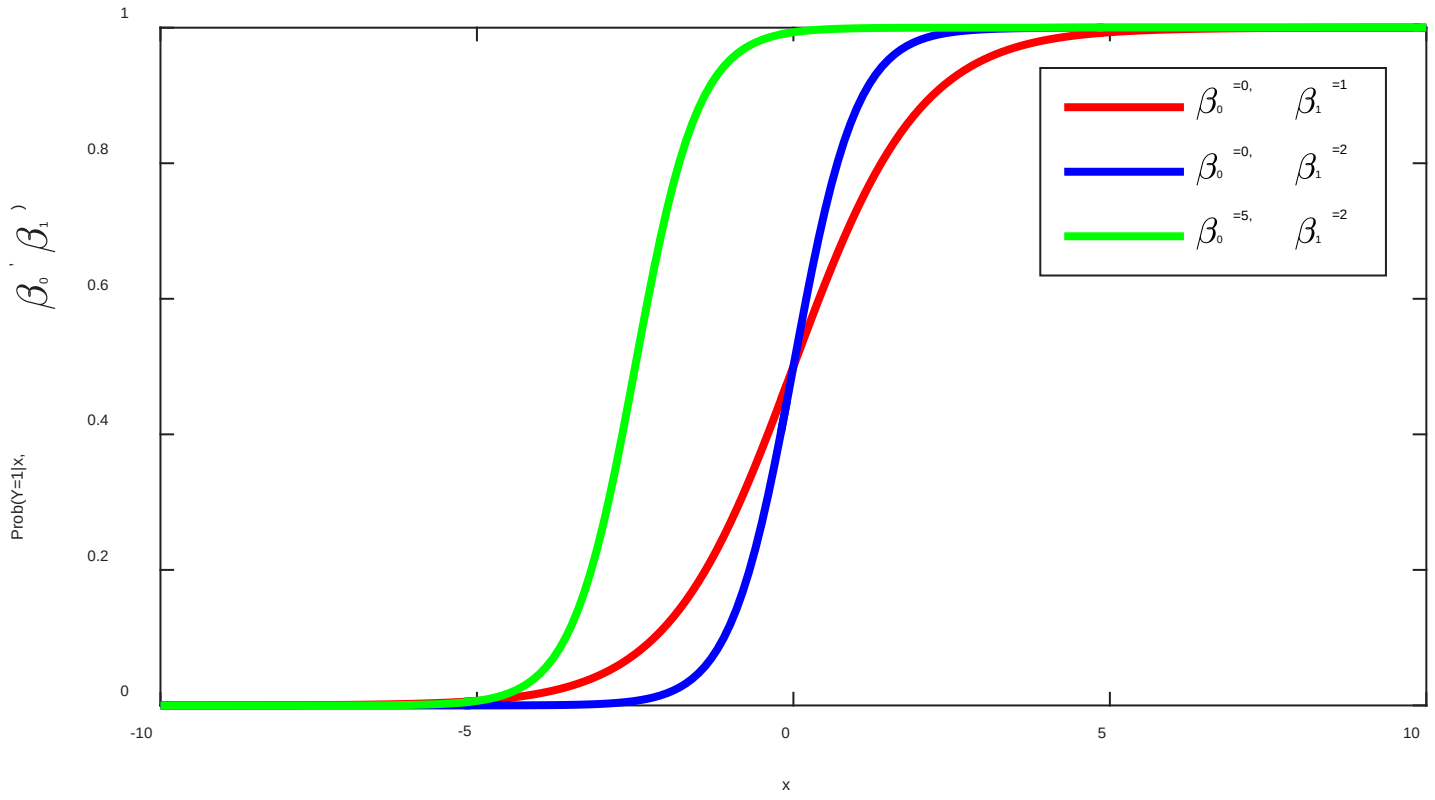
- $P(Y = 1|X, \beta) = \sigma(X^T\beta) = \frac{1}{1+\exp\{-X^T\beta\}} = \frac{\exp\{X^T\beta\}}{1+\exp\{X^T\beta\}}$
- $P(Y = 0|X, \beta) = 1 - \sigma(X^T\beta) = \frac{\exp\{-X^T\beta\}}{1+\exp\{-X^T\beta\}} = \frac{1}{1+\exp\{X^T\beta\}}$

$$\beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}$$

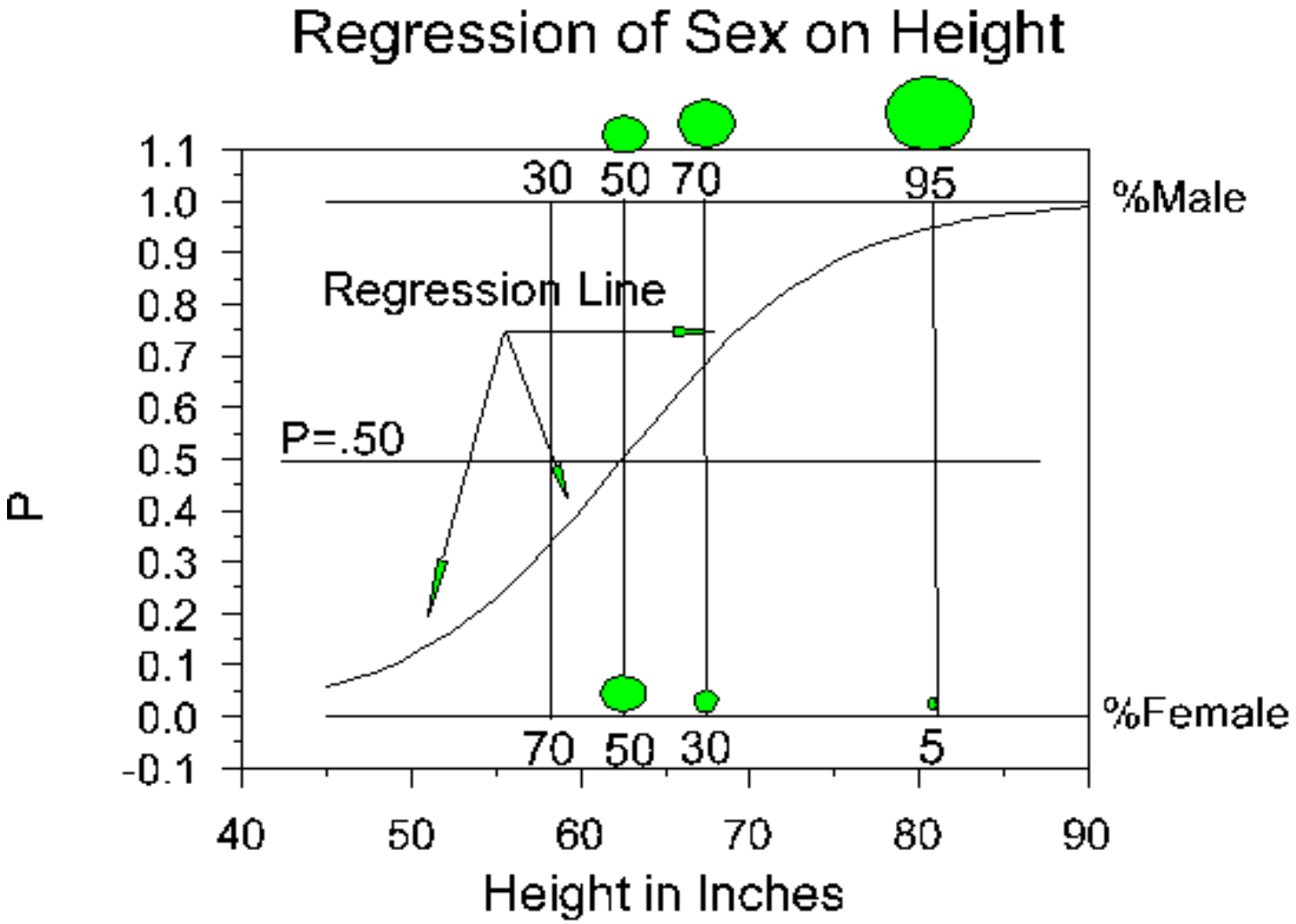
- In other words
  - $Y|\mathrm{X}, \beta \sim Bernoulli(\sigma(X^T\beta))$

The 1-d Situation

- $P(Y = 1|x, \beta_0, \beta_1) = \sigma(\beta_1 x + \beta_0)$



Example



Q: What is β_0 here?

Parameter Estimation

- MLE estimation
  - Given a dataset $D, with\ n\ data\ points$
  - For a single data object with attributes $\boldsymbol{x}_i$, class label y_i
    - Let $p_i = p(Y = 1|\boldsymbol{x}_i, \beta), the\ prob.\ of\ i\ in\ class\ 1$
    - The probability of observing y_i would be
      - If $y_i = 1, then\ \ p_i$
      - If $y_i = 0, then\ 1 - p_i$
      - Combing the two cases: $p_i^{y_i}(1 - p_i)^{1-y_i}$

$$L = \prod_i p_i^{y_i}(1 - p_i)^{1-y_i} =$$

$$\prod_i \left(\frac{\exp\{X^T\beta\}}{1+\exp\{X^T\beta\}}\right)^{y_i} \left(\frac{1}{1+\exp\{X^T\beta\}}\right)^{1-y_i}$$

Optimization

- Equivalent to maximize log likelihood
- $L = \sum_i y_i \boldsymbol{x}_i^T \beta - \log\left(1 + \exp\{\boldsymbol{x}_i^T \beta\}\right)$
- Gradient ascent update:
  - $$\beta^{new} = \beta^{old} + \eta \frac{\partial L(\beta)}{\partial \beta}$$

Step size

- Newton-Raphson update
  - $$\beta^{new} = \beta^{old} - \left(\frac{\partial^2 L(\beta)}{\partial \beta \partial \beta^T}\right)^{-1} \frac{\partial L(\beta)}{\partial \beta}$$
  - where derivatives at evaluated at β^{old}

First Derivative

$$\frac{\partial L(\beta)}{\beta_{1j}} = \sum_{i=1}^{N} y_i x_{ij} - \sum_{i=1}^{N} \frac{x_{ij} e^{\beta^T x_i}}{1 + e^{\beta^T x_i}}$$

$$= \sum_{i=1}^{N} y_i x_{ij} - \sum_{i=1}^{N} p(x; \beta) x_{ij}$$

$$= \sum_{i=1}^{N} x_{ij} (y_i - p(x_i; \beta))$$

$p(x_i; \beta)$

j = 0, 1, …, p

Second Derivative

- It is a (p+1) by (p+1) matrix, Hessian Matrix, with jth row and nth column as

$$
\begin{aligned}
& \frac{\partial L(\beta)}{\partial \beta_{1j} \partial \beta_{1n}} \\
= & -\sum_{i=1}^{N} \frac{(1+e^{\beta^T x_i}) e^{\beta^T x_i} x_{ij} x_{in} - (e^{\beta^T x_i})^2 x_{ij} x_{in}}{(1+e^{\beta^T x_i})^2} \\
= & -\sum_{i=1}^{N} x_{ij} x_{in} p(x_i;\beta) - x_{ij} x_{in} p(x_i;\beta)^2 \\
= & -\sum_{i=1}^{N} x_{ij} x_{in} p(x_i;\beta)(1-p(x_i;\beta)) \; .
\end{aligned}
$$

What about Multiclass Classification?

- It is easy to handle under logistic regression, say M classes
  - $P(Y = j|X) = \frac{\exp\{X^T\beta_j\}}{1+\sum_{m=1}^{M-1}\exp\{X^T\beta_m\}}$, for j = $1, \dots, M-1$
  - $P(Y = M|X) = \frac{1}{1+\sum_{m=1}^{M-1}\exp\{X^T\beta_m\}}$

Vector Data: Regression

- Vector Data
- Linear Regression Model
- Logistic Regression Model
- Generalized Linear Model 
- Summary

Recall Linear Regression and Logistic Regression

- Linear Regression
  - $y|\boldsymbol{x}, \beta \sim N(\boldsymbol{x}^T \beta, \sigma^2)$
- Logistic Regression
  - $y|\boldsymbol{x}, \beta \sim Bernoulli(\sigma(\boldsymbol{x}^T \beta))$

- How about other distributions?
  - Yes, as long as they belong to exponential family

Exponential Family

- Canonical Form
  - $p(y;\eta) = b(y)\exp\left(\eta^T T(y) - a(\eta)\right)$
  - η: natural parameter
  - $T(y)$: sufficient statistic
  - $a(\eta)$: log partition function for normalization
  - $b(y)$: function that only dependent on y

Examples of Exponential Family

- Many:

$$p(y;\eta) = b(y)\exp\left(\eta^T T(y) - a(\eta)\right)$$

  - Gaussian, Bernoulli, Poisson, beta, Dirichlet, categorical, …

- For Gaussian (not interested in σ)

$$\begin{aligned} p(y;\mu) &= \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}(y-\mu)^2\right) \\ &= \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}y^2\right)\cdot\exp\left(\mu y - \frac{1}{2}\mu^2\right) \end{aligned}$$

$$\begin{aligned} \eta &= \mu \\ T(y) &= y \\ a(\eta) &= \mu^2/2 \\ &= \eta^2/2 \\ b(y) &= (1/\sqrt{2\pi})\exp(-y^2/2) \end{aligned}$$

- For Bernoulli

$$\begin{aligned} p(y;\phi) &= \phi^y(1-\phi)^{1-y} \\ &= \exp(y\log\phi + (1-y)\log(1-\phi)) \\ &= \exp\left(\left(\boxed{\log\left(\frac{\phi}{1-\phi}\right)}\right)y + \log(1-\phi)\right) \end{aligned}$$

η

$$\begin{aligned} T(y) &= y \\ a(\eta) &= -\log(1-\phi) \\ &= \log(1+e^\eta) \\ b(y) &= 1 \end{aligned}$$

Recipe of GLMs

- Determines a distribution for y
  - E.g., Gaussian, Bernoulli, Poisson
- Form the linear predictor for η
  - $\eta = \boldsymbol{x}^T \beta$
- Determines a link function: $\mu = g^{-1}(\eta)$
  - Connects the linear predictor to the mean of the distribution
  - E.g., $\mu = \eta$ for Gaussian, $\mu = \sigma(\eta)$ for Bernoulli, $\mu = exp(\eta)$ for Poisson

Vector Data: Regression

- Vector Data
- Linear Regression Model
- Logistic Regression Model
- Generalized Linear Model
- Summary 

Summary

- What is vector data?
  - Attribute types
- Linear regression
  - OLS, Probabilistic interpretation
- Logistic regression
  - Sigmoid function, multiclass classification
- Generalized linear model
  - Exponential family, link function