Lecture 8: Regression

Shuai Li

John Hopcroft Center, Shanghai Jiao Tong University

https://shuaili8.github.io

https://shuaili8.github.io/Teaching/CS410/index.html

Linear Regression

Regression example

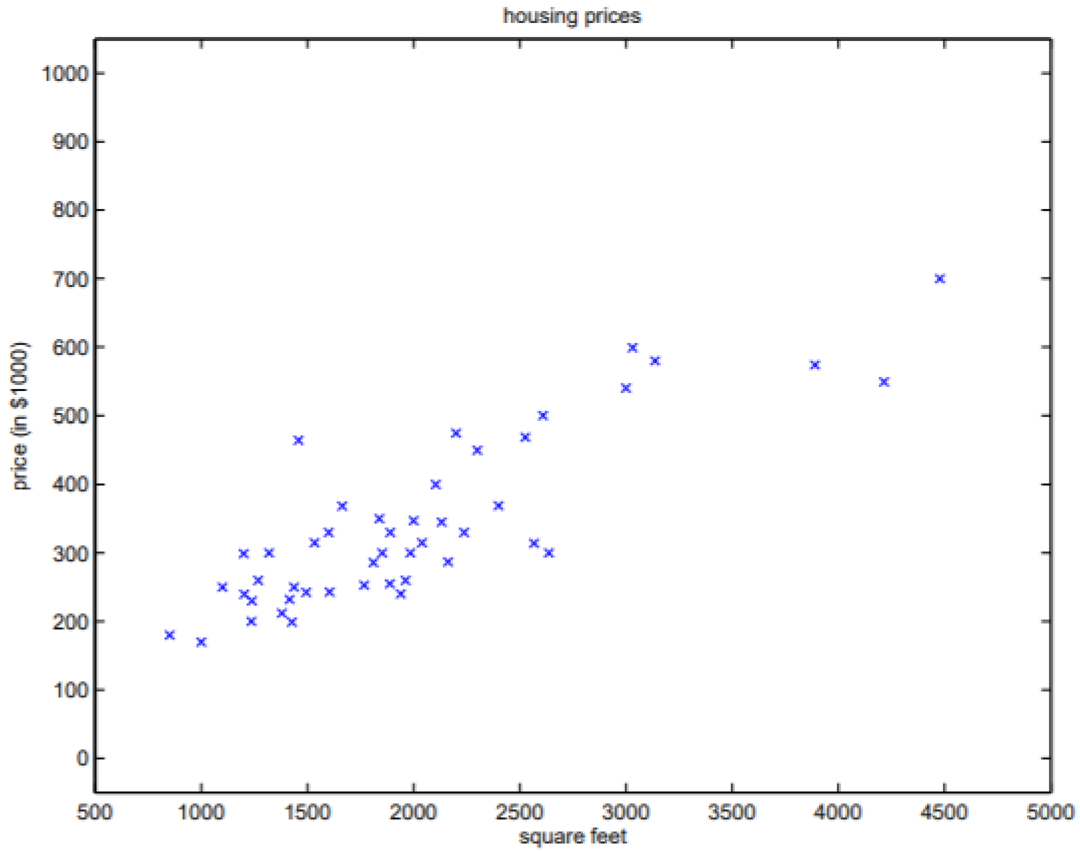
- Given the following values of X and Y :
  (1,1), (2,2), (4,4), (100,100), (20, 20)
  what is the value of Y when X = 5?
- The answer is 5, not difficult

- What if the given values are
  $$(1,1), (2,4), (4,16), (100,10000), (20, 400)$$
- Y is 25 when X =5, right?

- Rationale:
  - Look at some examples and then tries to identify the most suitable relationship between the sets X and Y
  - Using this identified relationship, try to predict the values for new examples

Regression example (cont.)

Living area (feet^2)	Price (1000\$s)
2104	400
1600	330
2400	369
1416	232
3000	540
⋮	⋮



Linear regression

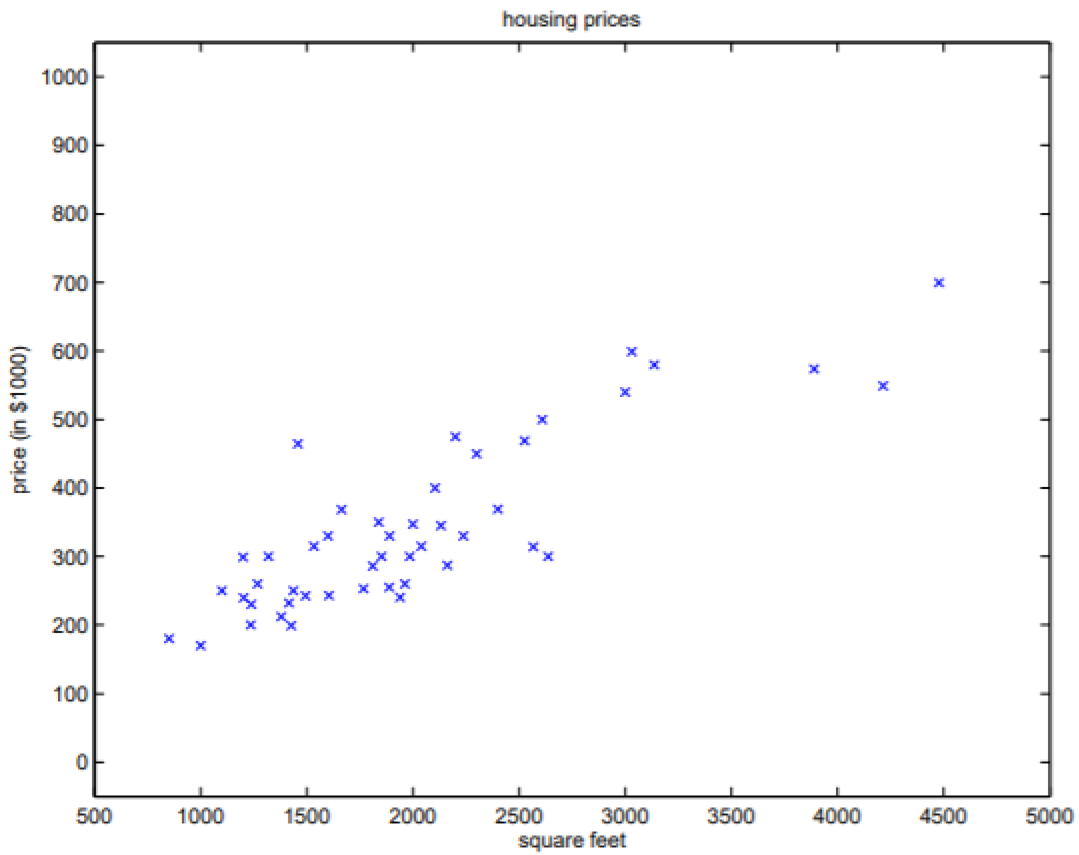
- Use linear relationship to approximate the function of Y on X

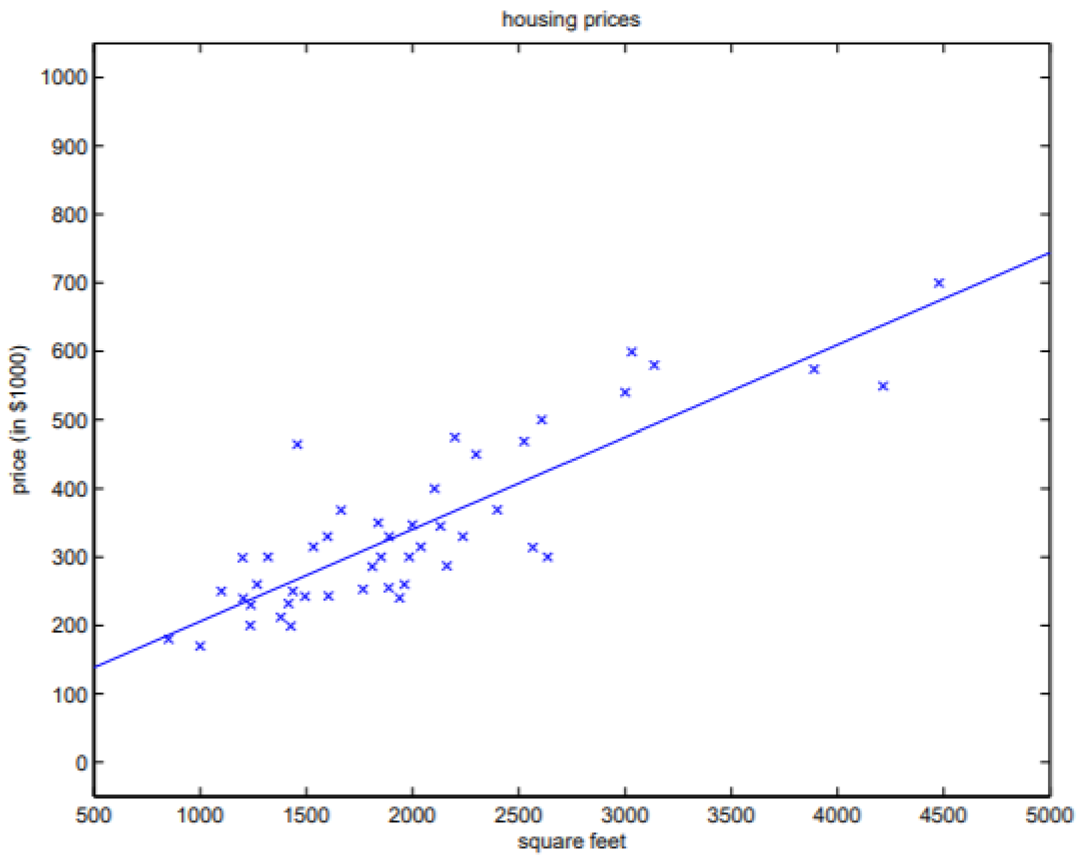
- How to select the most appropriate linear model?
- Error: Mean squared error (MSE)

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

  - Where Y and $\hat{Y}$ are the true values and predicted values respectively
- Find the linear model with the smallest MSE

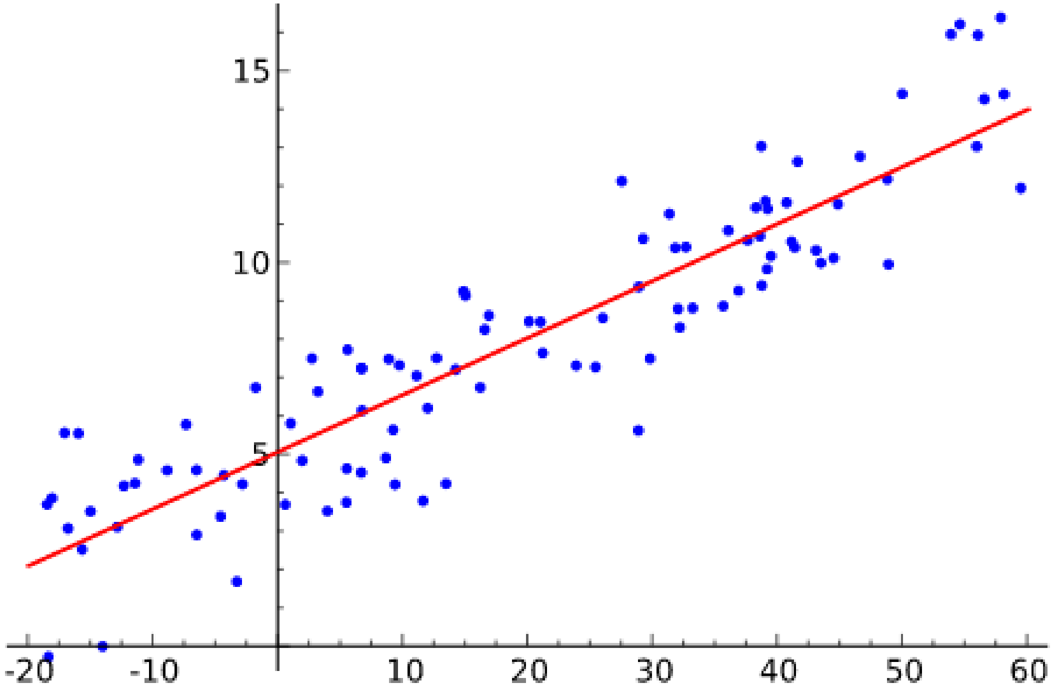
Use linear model to fit the given data





Linear regression 2D example

- In the 2D example, you are looking for a linear equation
  $y = x_1 * \theta_1 + \theta_0$ to fit the data with smallest MSE



https://upload.wikimedia.org/wikipedia/commons/thumb/3/3a/Linear_regression.svg/1200px-Linear_regression.svg.png

Question

- Given the dataset
$$(1,1), (2,4), (3,5)$$
and the linear model
$$Y = 2X + 1$$
- What is the mean squared error?

- The predicted points are
$$(1,3), (2,5), (3,7)$$
- So the mean squared error (MSE) is
$$\frac{1}{3}(2^2 + 1^2 + 2^2) = 3$$

Question 2

- Given the <real value, predicted value> pairs as:
  $< 1.2, 1.7 >, < 0.9, 0.2 >, < -0.3, 0.1 >, < 1.3, 0.3 >, < 1.1, 1.2 >$
  compute the mean squared error

- The answer is

$$\frac{1}{5}(0.5^2 + 0.7^2 + 0.4^2 + 1^2 + 0.1^2) = 0.382$$

How to get linear model with minimal MSE

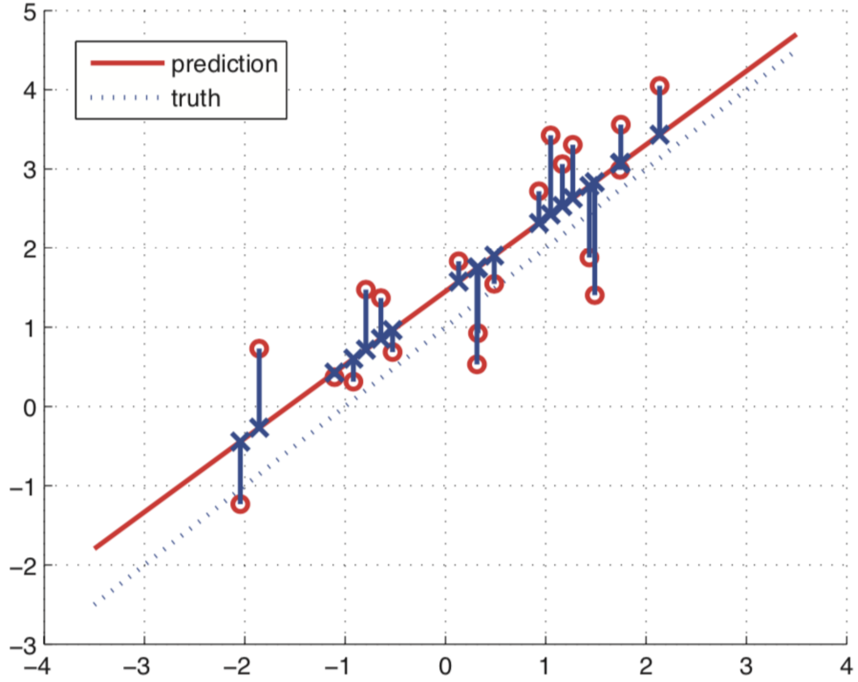
- MSE for model parameter θ:

$$J(\theta) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \theta^\top x_i)^2$$

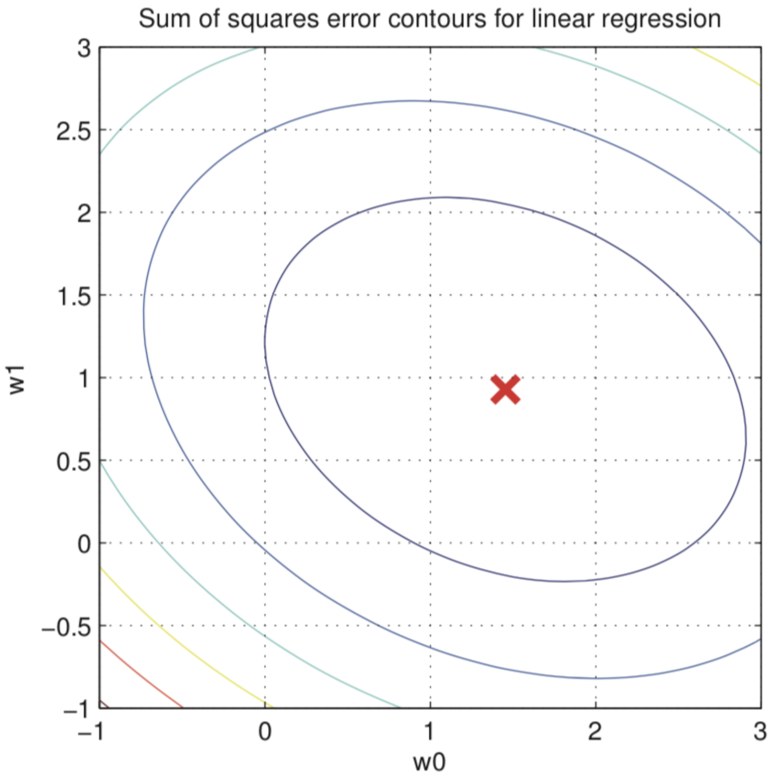
- Find an estimator $\hat{\theta}$ to minimize $J(\theta)$
- $y = \theta^\top x + b + \varepsilon$. Then we can write $x' = (1, x^1, \dots, x^d), \theta = (b, \theta_1, \dots, \theta_d)$, then $y = \theta^\top x' + \varepsilon$

- Note that $J(\theta)$ is a convex function in θ, so it has a unique minimal point

Interpretation



(a)



(b)

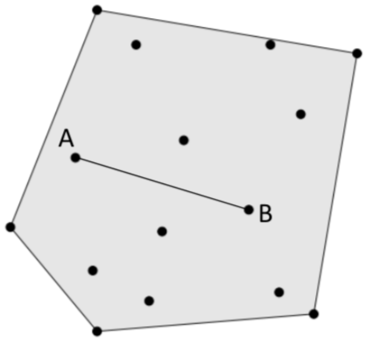
Figure credit: Kevin Murphy

Convex set

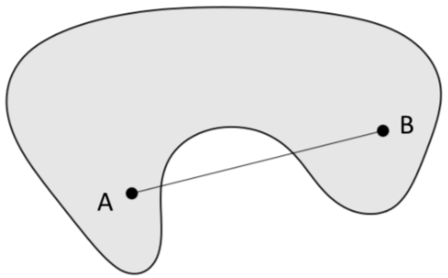
- A convex set S is a set of points such that, given any two points A, B in that set, the line AB joining them lies entirely within S.

$$tx_1 + (1-t)x_2 \in S$$

for all $x_1, x_2 \in S, 0 \leq t \leq 1$

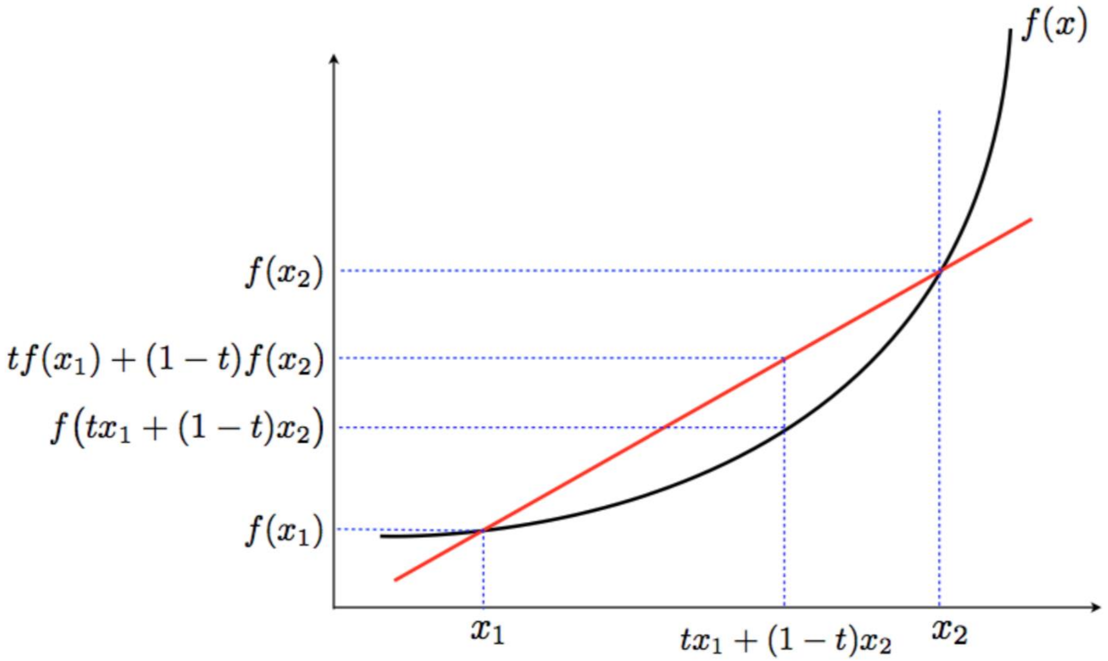


Convex set



Non-convex set

Convex function



$f : \mathbb{R}^n \to \mathbb{R}$ is convex if $\mathbf{dom}\ f$ is a convex set and

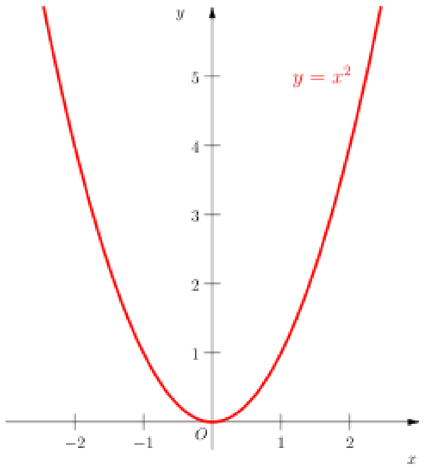
$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2)$$

for all $x_1, x_2 \in \mathbf{dom}\ f, 0 \leq t \leq 1$

$J(\theta)$ is convex

Check it by yourself !

- $f(x) = (y - x)^2 = (x - y)^2$ is convex in x



- $g(\theta) = f(\theta^\top x)$

$$
\begin{aligned}
&g\big((1-t)\theta_1 + t\theta_2\big) \\
&= f\left((1-t){\theta_1}^\top x + t{\theta_2}^\top x\right) \\
&\le (1-t)f\big({\theta_1}^\top x\big) + tf\big({\theta_2}^\top x\big) \\
&= (1-t)g(\theta_1) + tg(\theta_2)
\end{aligned}
$$

Convexity of f

- The sum of convex functions is convex
- Thus $J(\theta)$ is convex

Minimal point (Normal equation)

- $\frac{\partial J(\theta)}{\partial \theta} = \frac{2}{N}\sum_{i=1}^{N}(\theta^\top x_i - y_i)x_i = \frac{2}{N}\sum_{i=1}^{N}\left(x_i x_i^\top \theta - x_i y_i\right)$
- Letting the derivative be zero

$$\left(\sum_{i=1}^{N} x_i x_i^\top\right)\theta = \sum_{i=1}^{N} x_i y_i$$

- If we write $X = \begin{bmatrix} x_1^\top \\ \vdots \\ x_N^\top \end{bmatrix} = \begin{bmatrix} x_1^1 & \cdots & x_1^d \\ & \vdots & \\ x_N^1 & \cdots & x_N^d \end{bmatrix}, y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$, then

$$X^\top X\theta = X^\top y$$

Minimal point (Normal equation) (cont.)

- $X^\top X\theta = X^\top y$
- When $X^\top X$ is invertible

$$\hat{\theta} = (X^\top X)^{-1} X^\top y$$

- When $X^\top X$ is not invertible

pseudo-inverse

$$\hat{\theta} = (X^\top X)^{\dagger} X^\top y$$

- E.g. The pseudo-inverse of $\begin{bmatrix} 1 & & \\ & 2 & \\ & & 0 \end{bmatrix}$ is $\begin{bmatrix} 1 & & \\ & \frac{1}{2} & \\ & & 0 \end{bmatrix}$

Interpretation of Least square error

- A two-dim example: $x_1 * \theta_1 + x_2 * \theta_2 = y$, where x_i and y are vectors.

- We are using the combination of x_i to approximate the projection of y at their plane

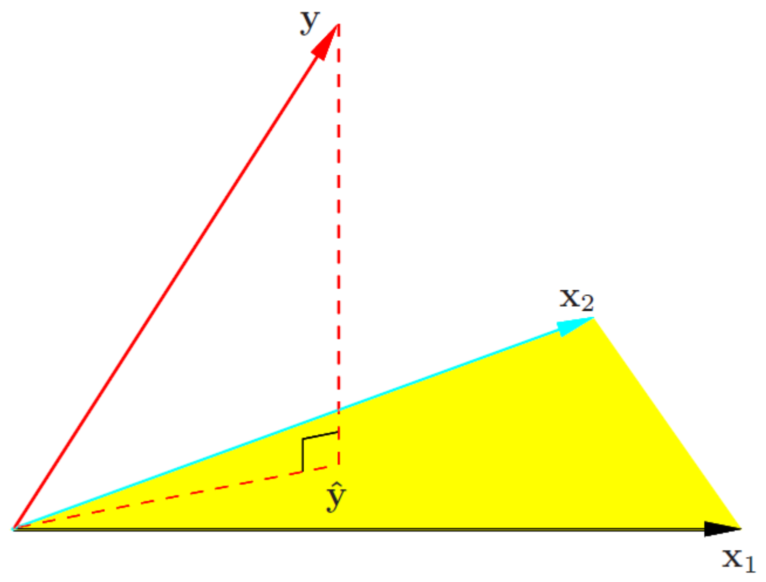


Figure credit: Trevor Hastie

Geometric interpretation

- N=3,d=2

$$\mathbf{X} = \begin{pmatrix} 1 & 2 \\ 1 & -2 \\ 1 & 2 \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} 8.8957 \\ 0.6130 \\ 1.7761 \end{pmatrix}$$

$$\underset{\hat{\mathbf{y}} \in \text{span}(\{\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_D\})}{\operatorname{argmin}} \|\mathbf{y} - \hat{\mathbf{y}}\|_2.$$

column vectors in X

- $\hat{\theta} = (X^\top X)^{-1} X^\top y$
- $\hat{y} = X\hat{\theta} = X(X^\top X)^{-1} X^\top y$

Projection
hat matrix (put a "hat" on y)

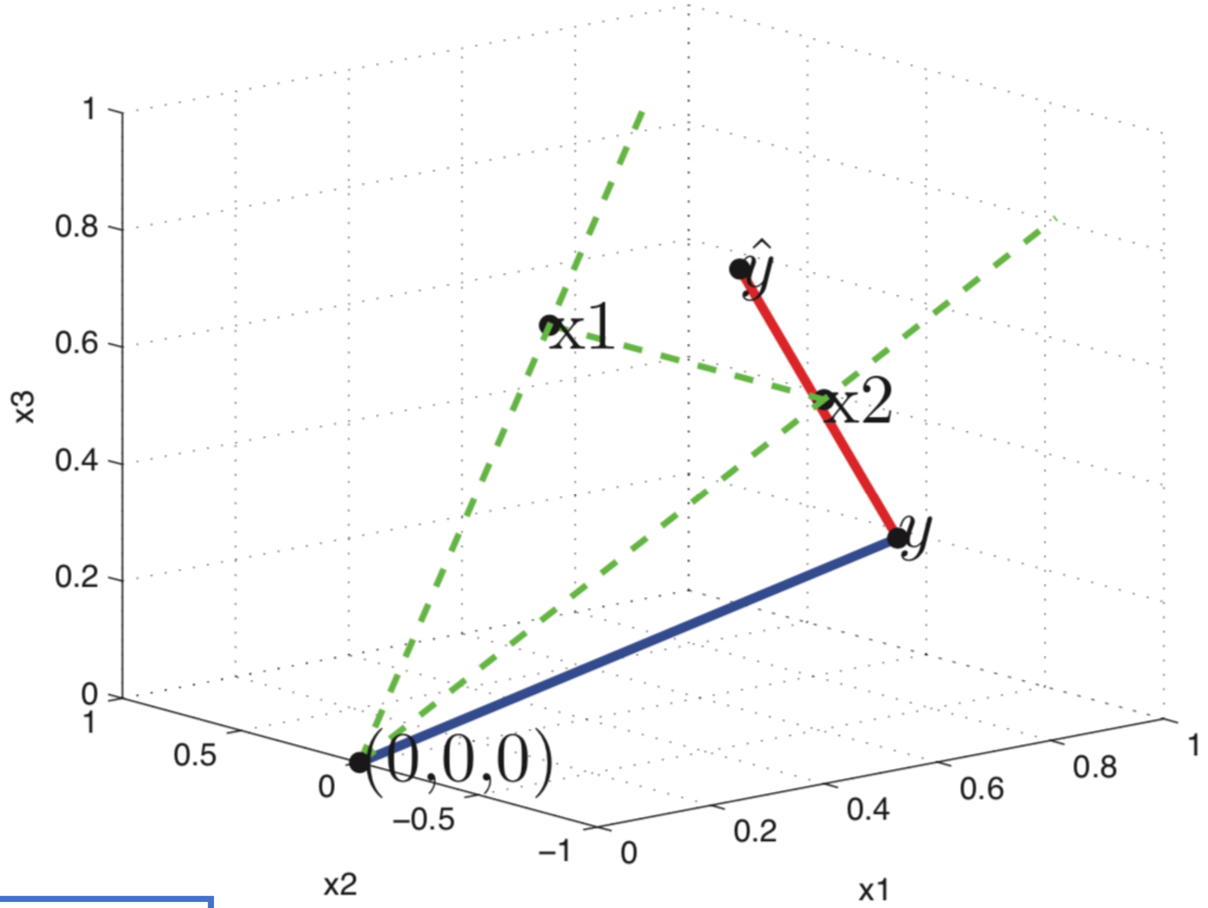


Figure credit: Kevin Murphy

Examples

Question 1

- Given the dataset

$$(1,1),(2,4),(3,5)$$

  compute the normal equation for θ, solve θ and compute the MSE

$$\boxed{X^\top X\theta = X^\top y}$$

- $X = \begin{bmatrix} x_1^\top \\ x_2^\top \\ x_3^\top \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix}, y = \begin{bmatrix} 1 \\ 4 \\ 5 \end{bmatrix}$

$$X^\top X = \begin{bmatrix} 3 & 6 \\ 6 & 14 \end{bmatrix}, X^\top y = \begin{bmatrix} 10 \\ 24 \end{bmatrix}$$

$$\begin{bmatrix} 3 & 6 \\ 6 & 14 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 10 \\ 24 \end{bmatrix}$$

- $\theta = \left[-\frac{2}{3}, 2\right], y = -\frac{2}{3} + 2x$. MSE=$\frac{2}{9}$

Question 2

- Some economist say that the impact of GDP in ‘current year’ will have effect on vehicle sales ‘next year’. So whichever year GDP was less, the coming year sales was lower and when GDP increased the next year vehicle sales also increased

- Let’s have the equation as $y = \theta_0 + \theta_1 x$, where
  y = number of vehicles sold in the year
  x = GDP of prior year
  We need to find θ_0 and θ_1

Question 2 (cont.)

- Here is the data between 2011 and 2016.

Year	GDP	Sales of vehicle
2011	6.2	
2012	6.5	26.3
2013	5.48	26.65
2014	6.54	25.03
2015	7.18	26.01
2016	7.93	27.9
2017		30.47
2018		

Homework

- Question 1: what is the normal equation?
- Question 2: suppose the GDP increasement in 2017 is 7%, how many vehicles will be sold in 2018?

Gradient methods

Motivation – large dataset

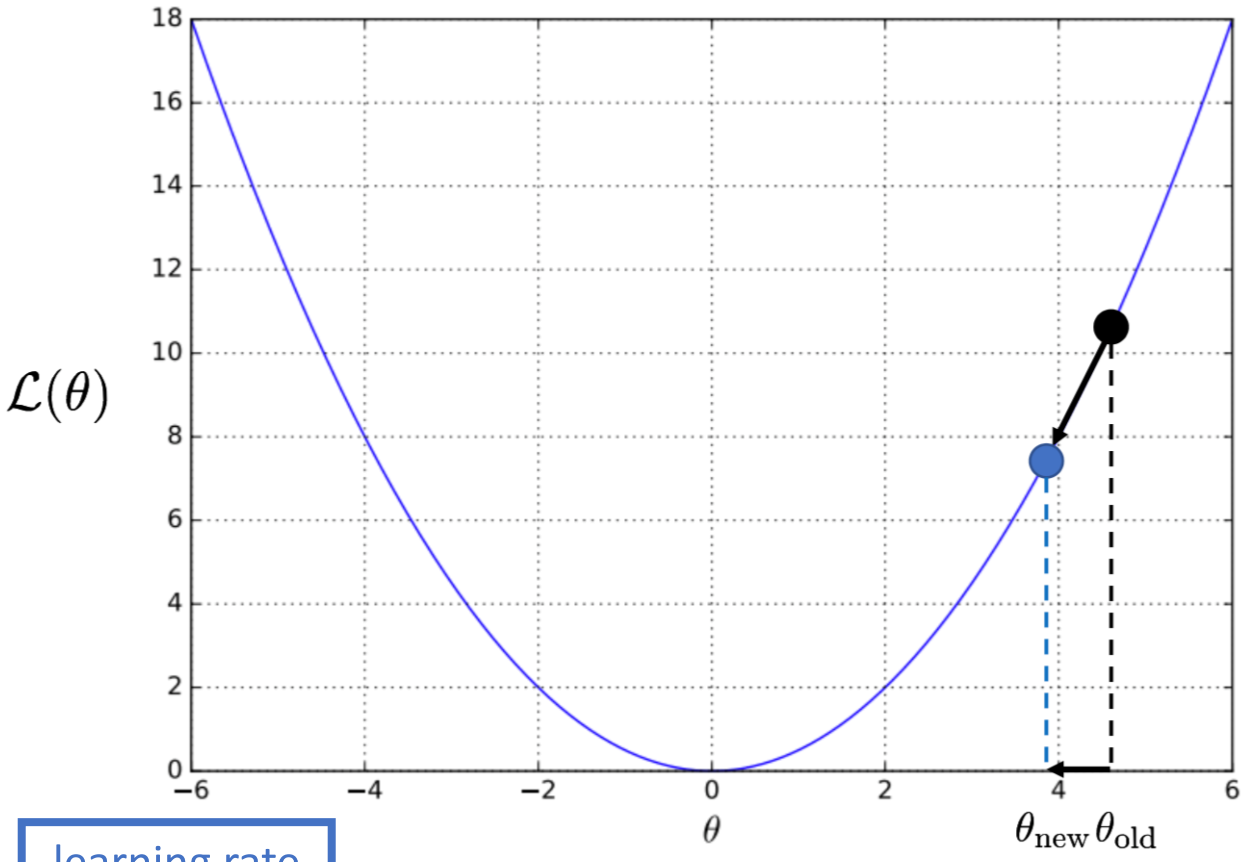
- Too big to compute directly

$$\hat{\theta} = (X^\top X)^{-1} X^\top y$$

- Recall the objective is to minimize the loss function

$$L(\theta) = J(\theta) = \frac{1}{N} \sum_{i=1}^{N} (y_i - \theta^\top x_i)^2$$

- Gradient descent method



learning rate

$$\theta_{\text{new}} \leftarrow \theta_{\text{old}} - \eta \frac{\partial \mathcal{L}(\theta)}{\partial \theta}$$

Batch gradient descent

- $f_\theta(x) = \theta^\top x$

$$J(\theta) = \frac{1}{N}\sum_{i=1}^{N}(y_i - f_\theta(x_i))^2 \qquad \min_\theta J(\theta)$$

- Update $\theta_{\text{new}} \leftarrow \theta_{\text{old}} - \eta \frac{\partial J(\theta)}{\partial \theta}$ for the whole batch

$$\frac{\partial J(\theta)}{\partial \theta} = -\frac{2}{N}\sum_{i=1}^{N}(y_i - f_\theta(x_i))\frac{\partial f_\theta(x_i)}{\partial \theta}$$

$$= -\frac{2}{N}\sum_{i=1}^{N}(y_i - f_\theta(x_i))x_i$$

$$\theta_{\text{new}} = \theta_{\text{old}} + \eta\frac{2}{N}\sum_{i=1}^{N}(y_i - f_\theta(x_i))x_i$$

Stochastic gradient descent

$$J^{(i)}(\theta) = \quad (y_i - f_\theta(x_i))^2 \qquad\qquad \min_\theta \frac{1}{N}\sum_i J^{(i)}(\theta)$$

- Update $\theta_{\text{new}} = \theta_{\text{old}} - \eta\frac{\partial J^{(i)}(\theta)}{\partial\theta}$ for every single instance

$$\begin{aligned}\frac{\partial J^{(i)}(\theta)}{\partial\theta} &= -(y_i - f_\theta(x_i))\frac{\partial f_\theta(x_i)}{\partial\theta} \\ &= -(y_i - f_\theta(x_i))x_i \\ \theta_{\text{new}} &= \theta_{\text{old}} + \eta(y_i - f_\theta(x_i))x_i\end{aligned}$$

- Compare with BGD
  - Faster learning
  - Uncertainty or fluctuation in learning

Mini-Batch Gradient Descent

- A combination of batch GD and stochastic GD
- Split the whole dataset into *K* mini-batches

$$\{1, 2, 3, \ldots, K\}$$

- For each mini-batch *k*, perform one-step BGD towards minimizing

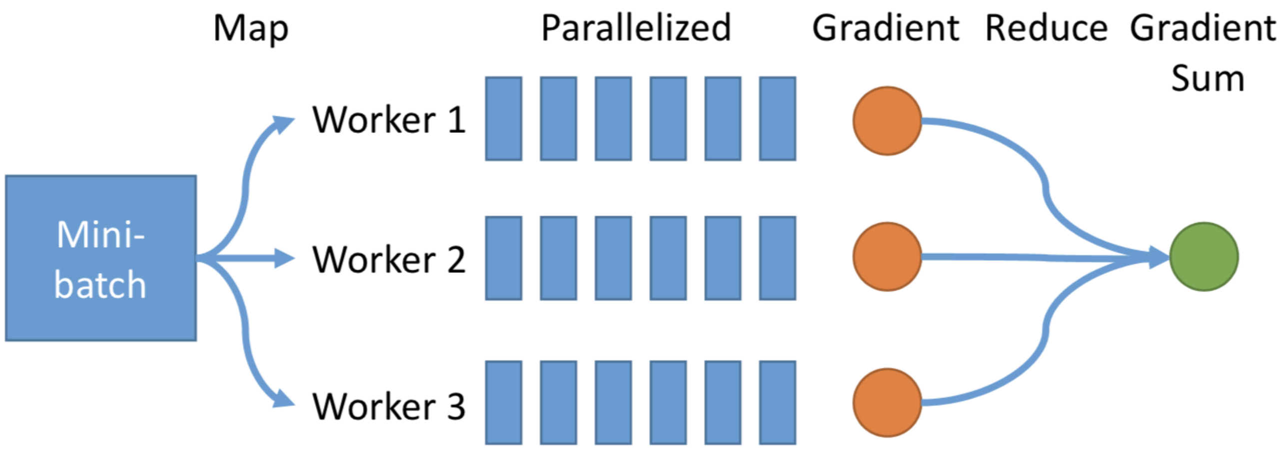
$$J^{(k)}(\theta) = \frac{1}{N_k} \sum_{i=1}^{N_k} (y_i - f_\theta(x_i))^2$$

- Update $\theta_{\text{new}} = \theta_{\text{old}} - \eta \frac{\partial J^{(k)}(\theta)}{\partial \theta}$ for each mini-batch

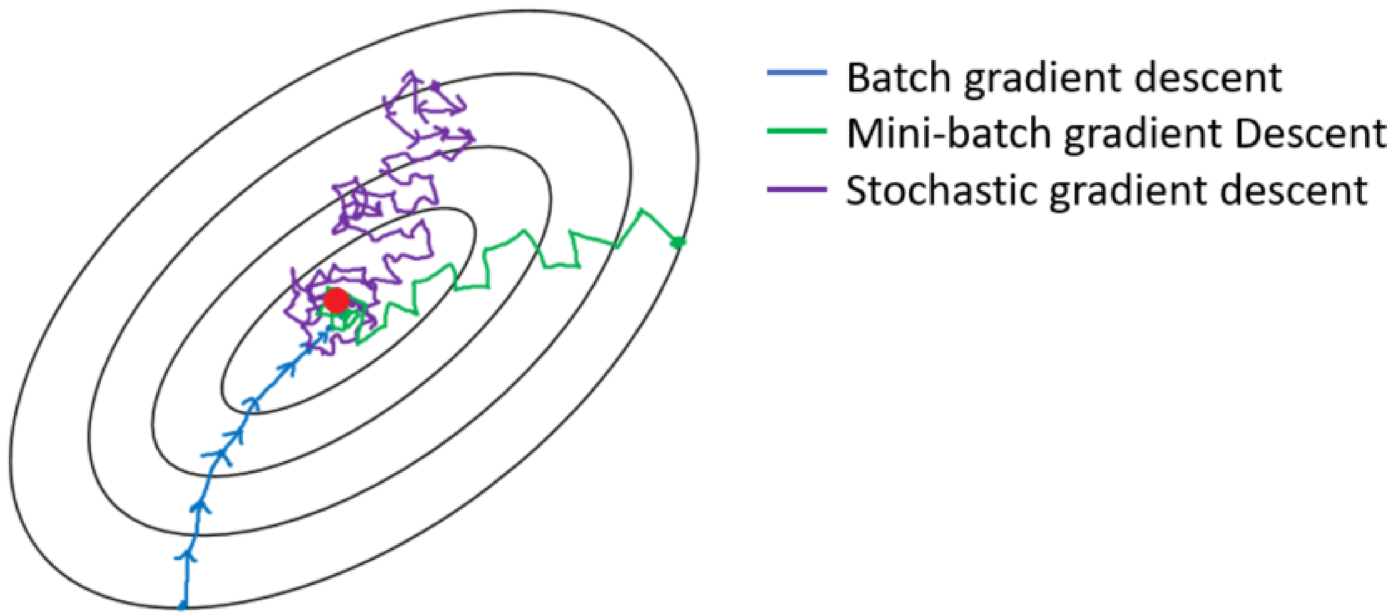
Mini-Batch Gradient Descent (cont.)

- Good learning stability (BGD)
- Good convergence rate (SGD)

- Easy to be parallelized
  - Parallelization within a mini-batch

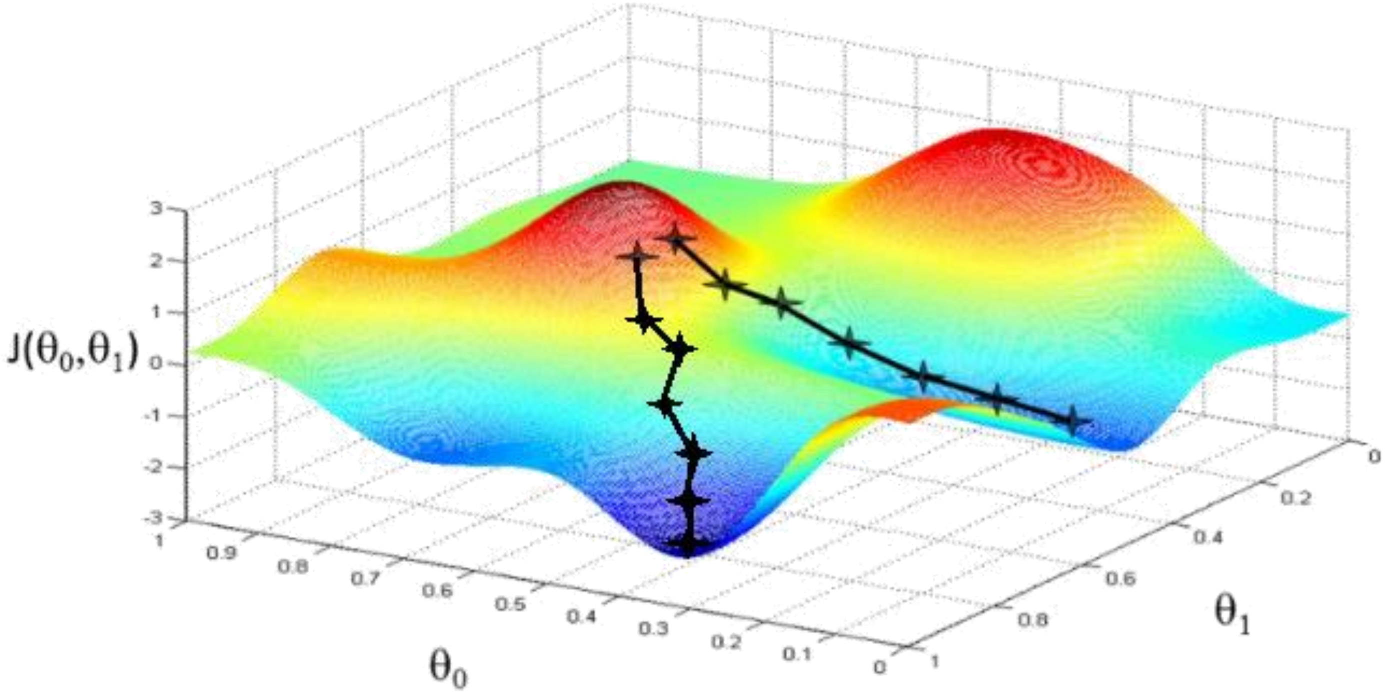


Comparisons

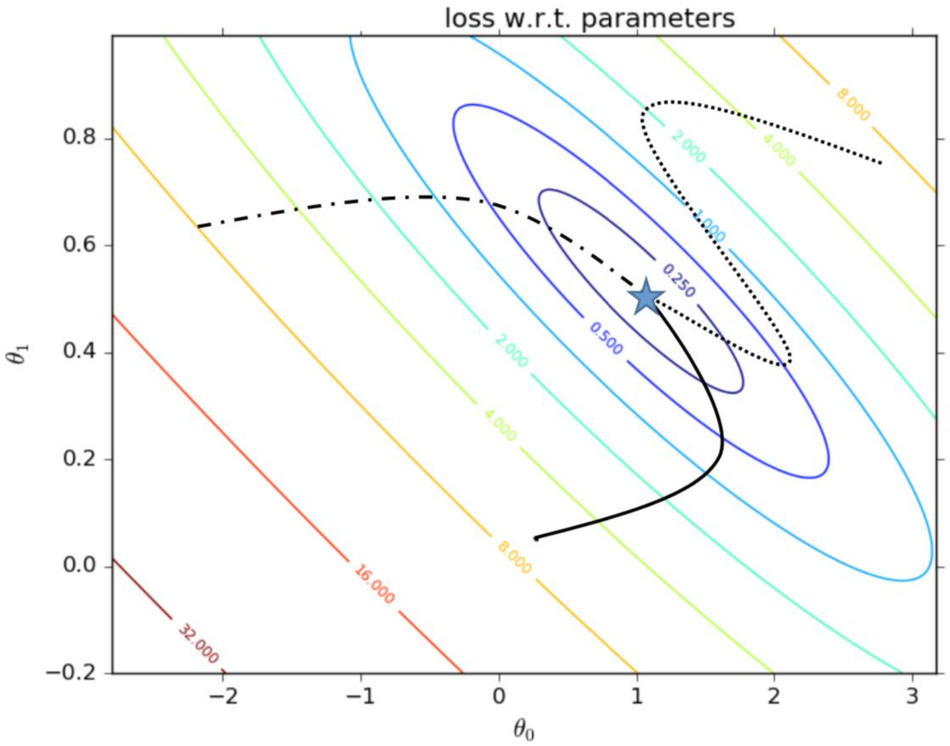


Searching

- Start with a new initial value θ
- Update θ iteratively (gradient descent)
- Ends at a minimum



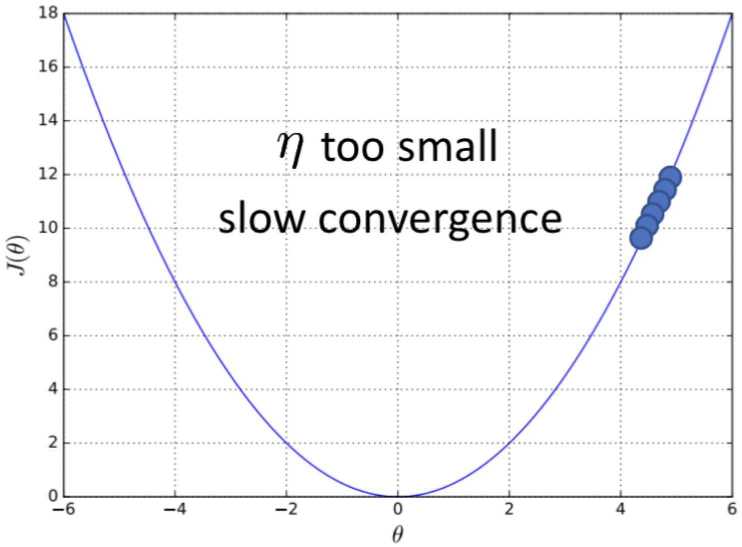
Uniqueness of minimum for convex objectives

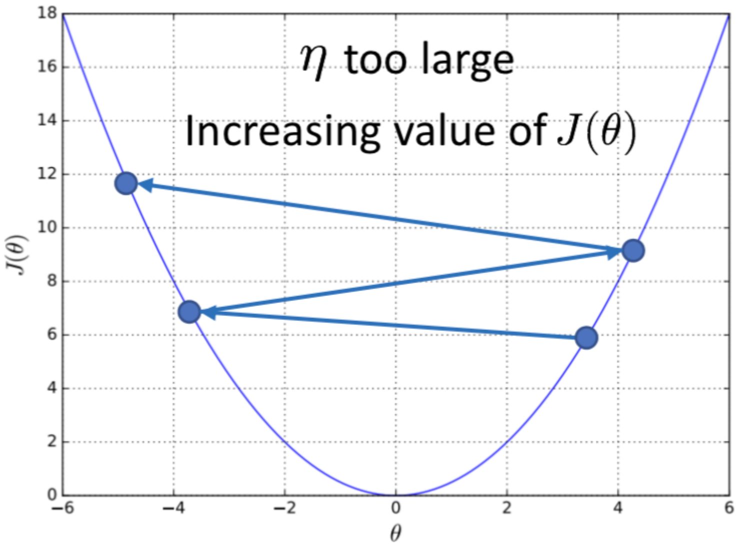


- Different initial parameters and different learning algorithm lead to the same optimum

Learning rate

$$\theta_{\text{new}} = \theta_{\text{old}} - \eta \frac{\partial J(\theta)}{\partial \theta}$$





- The initial point may be too far away from the optimal solution, which takes much time to converge

- May overshoot the minimum
- May fail to converge
- May even diverge

- To see if gradient descent is working, print out $J(\theta)$ for each or every several iterations. If $J(\theta)$ does not drop properly, adjust η

Probabilistic view

Probabilistic view

- Assume for each sampled $(x, y) \sim D$,
  $$y = \theta^\top x + \varepsilon$$
  where ε is Gaussian noise and $\varepsilon \sim \mathcal{N}(0, \sigma^2)$, or equivalently, $y \sim \mathcal{N}(\theta^\top x, \sigma^2)$

- The linear regression estimator $\hat{\theta}$ is the maximal likelihood estimator (MLE) of the data

Maximum likelihood estimation (MLE)

- Frequentists' view

$$
\begin{aligned}
\hat{\theta}_{\mathrm{MLE}} &= \arg\max P(X;\theta) \\
&= \arg\max P(x_1;\theta)P(x_2;\theta)\cdot\dots\cdot P(x_n;\theta) \\
&= \arg\max \log \prod_{i=1}^{n} P(x_i;\theta) \\
&= \arg\max \sum_{i=1}^{n} \log P(x_i;\theta) \\
&= \arg\min - \sum_{i=1}^{n} \log P(x_i;\theta)
\end{aligned}
$$

MLE view

- Given dataset S, find θ to maximize the likelihood of S, which is

$$\mathbb{P}[S|\theta] = \prod_{i=1}^{N} \mathbb{P}[y_i|x_i, \theta]$$

- $\hat{\theta} = \mathrm{argmax}_\theta \mathbb{P}[S|\theta]$

$$= \mathrm{argmax}_\theta \log \mathbb{P}[S|\theta]$$

$$= \mathrm{argmax}_\theta \sum_{i=1}^{N} \log \mathbb{P}[y_i|x_i, \theta]$$

- $\mathbb{P}[y_i|x_i, \theta] = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - \theta^\top x_i)^2}{2\sigma^2}}$

Gaussian distribution

MLE view (cont.)

$\mathbb{P}[y_i|x_i,\theta] = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i-\theta^\top x_i)^2}{2\sigma^2}}$

- $\hat{\theta} = \mathrm{argmax}_\theta \sum_{i=1}^N \log \mathbb{P}[y_i|x_i,\theta]$

$$= \mathrm{argmax}_\theta \sum_{i=1}^N \log \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{(y_i - \theta^\top x_i)^2}{2\sigma^2}$$

$$= \mathrm{argmax}_\theta - \sum_{i=1}^N (y_i - \theta^\top x_i)^2$$

$$= \mathrm{argmin}_\theta \sum_{i=1}^N (y_i - \theta^\top x_i)^2$$

residual sum of squares (RSS)

$$= \mathrm{argmin}_\theta \frac{1}{N} \sum_{i=1}^N (y_i - \theta^\top x_i)^2$$

Application Examples

Trend line

- A trend line represents a trend, the long-term movement in time series data after other components have been accounted for
- E.g. Stock price, heart rate, sales volume, temperature

- Given a set of points in time t and data values y_t, find the linear relationship of y_t with respect to t
- Find a and b to minimize

$$\sum_t [y_t - (\hat{a}t + \hat{b})]^2$$

Finance: Capital asset pricing model (CAPM)

- Describes the relationship between systematic risk and expected return for assets, particularly stocks
- Is widely used throughout finance for pricing risky securities and generating expected returns for assets given the risk of those assets and cost of capital

$$E(R_i) = R_f + \beta_i(E(R_m) - R_f)$$

where:

- $E(R_i)$ is the expected return on the capital asset
- R_f is the risk-free rate of interest such as interest arising from government bonds
- β_i (the *beta*) is the sensitivity of the expected excess asset returns to the expected excess market returns,
- $E(R_m)$ is the expected return of the market
- $E(R_m) - R_f$ is sometimes known as the *market premium*

Example of CAPM

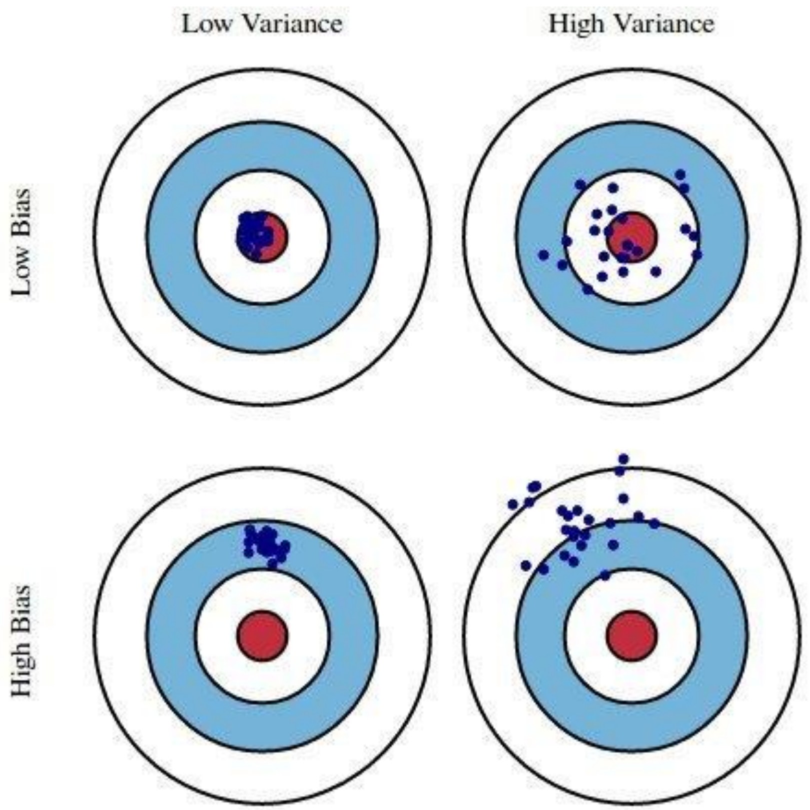
- The risk-free rate of return (which is usually the return rate of government bonds) is 3%, and average market rate is 5%. Suppose the beta in car industry is 1.4, what is the average return rate for the car industry?

- In another way, if we have the risk-free return rate, the market return rate and the return rate of 10 car companies, how to compute the beta for car industry?

Regularization

Problems of ordinary least squares (OLS)

- Best model is to minimize both the bias and the variance
- Ordinary least squares (OLS)
  - Previous linear regression
  - Unbiased
  - Can have huge variance
    - Multi-collinearity among data
      - When predictor variables are correlated to each other and to the response variable
      - E.g. To predict patient weight by the height, sex, and diet. But height and sex are correlated
    - Many predictor variables
      - Feature dimension close to number of data points
- Solution
  - Reduce variance at the cost of introducing some bias
  - Add a penalty term to the OLS equation

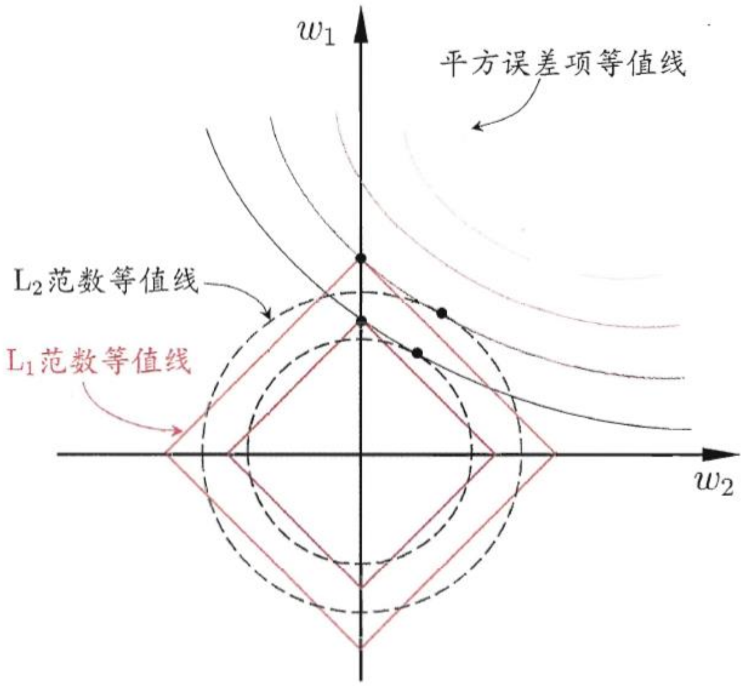


Ridge regression

- Regularization with L2 norm

$$L_{Ridge} = (y - X\theta)^2 + \lambda\|\theta\|_2^2$$

- $\lambda \to 0, \hat{\theta}_{Ridge} \to \hat{\theta}_{OLS}$
- $\lambda \to \infty, \hat{\theta} \to 0$
- As λ becomes larger, the variance decreases but the bias increases
- λ: Trade-off between bias and variance
    - Choose by cross-validation
- Ridge regression decreases the complexity of a model but does not reduce the number of variables (compared to other regularization like Lasso)



Solution of the ridge regression

- $\frac{\partial L_{Ridge}}{\partial \theta} = 2\sum_{i=1}^{N}(\theta^\top x_i - y_i)x_i + 2\lambda\theta$
- Letting the derivative be zero

$$\left(\lambda I + \sum_{i=1}^{N} x_i x_i^\top\right)\theta = \sum_{i=1}^{N} x_i y_i$$

- If we write $X = \begin{bmatrix} x_1^\top \\ \vdots \\ x_N^\top \end{bmatrix} = \begin{bmatrix} x_1^1 & \cdots & x_1^d \\ & \vdots & \\ x_N^1 & \cdots & x_N^d \end{bmatrix}, y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$, then

$$(\lambda I + X^\top X)\theta = X^\top y$$

$$\hat{\theta}_{\text{ridge}} = (\lambda I + X^\top X)^{-1} X^\top y$$

Recall the normal equation for OLS is $X^\top X\theta = X^\top y$

Always invertible

Maximum A Posteriori (MAP)

- Bayesians' view

$$P(\theta|X) = \frac{P(X|\theta) \times P(\theta)}{P(X)}$$

$$\begin{aligned}
\hat{\theta}_{\text{MAP}} &= \arg\max P(\theta|X) \\
&= \arg\min -\log P(\theta|X) \\
&= \arg\min -\log P(X|\theta) - \log P(\theta) + \log P(X) \\
&= \arg\min -\log P(X|\theta) - \log P(\theta)
\end{aligned}$$

Probabilistic view (MAP)

- Ridge regression estimator is an MAP estimator with Gaussian prior
- Suppose θ has the prior $P(\theta) = \mathcal{N}(0, \tau^2 I)$
- $\hat{\theta}_{MAP} = \operatorname{argmax}_\theta \sum_{i=1}^{N} \log \mathbb{P}[y_i | x_i, \theta] + \log P(\theta)$

$$f_{\mathbf{X}}(x_1, \ldots, x_k) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})\right)}{\sqrt{(2\pi)^k |\boldsymbol{\Sigma}|}}$$

$$\mathbb{P}[y_i | x_i, \theta] = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - \theta^\top x_i)^2}{2\sigma^2}}$$

$$= \operatorname{argmax}_\theta \sum_{i=1}^{N} -\frac{(y_i - \theta^\top x_i)^2}{2\sigma^2} - \frac{1}{2\tau^2} \|\theta\|_2^2$$

$$= \operatorname{argmax}_\theta -\frac{1}{N} \sum_{i=1}^{N} (y_i - \theta^\top x_i)^2 - \lambda \|\theta\|_2^2$$

$\lambda = \sigma^2 / \mathrm{N}\tau^2$

$$= \operatorname{argmin}_\theta \frac{1}{N} \sum_{i=1}^{N} (y_i - \theta^\top x_i)^2 + \lambda \|\theta\|_2^2$$

Linear regression with non-linear relationships

- E.g. $\phi(x) = (1, x, x^2, \ldots, x^d)$ and $y \sim \mathcal{N}(\theta^\top \phi(x), \sigma^2)$
  - Features: Last hidden layer of Neural Networks

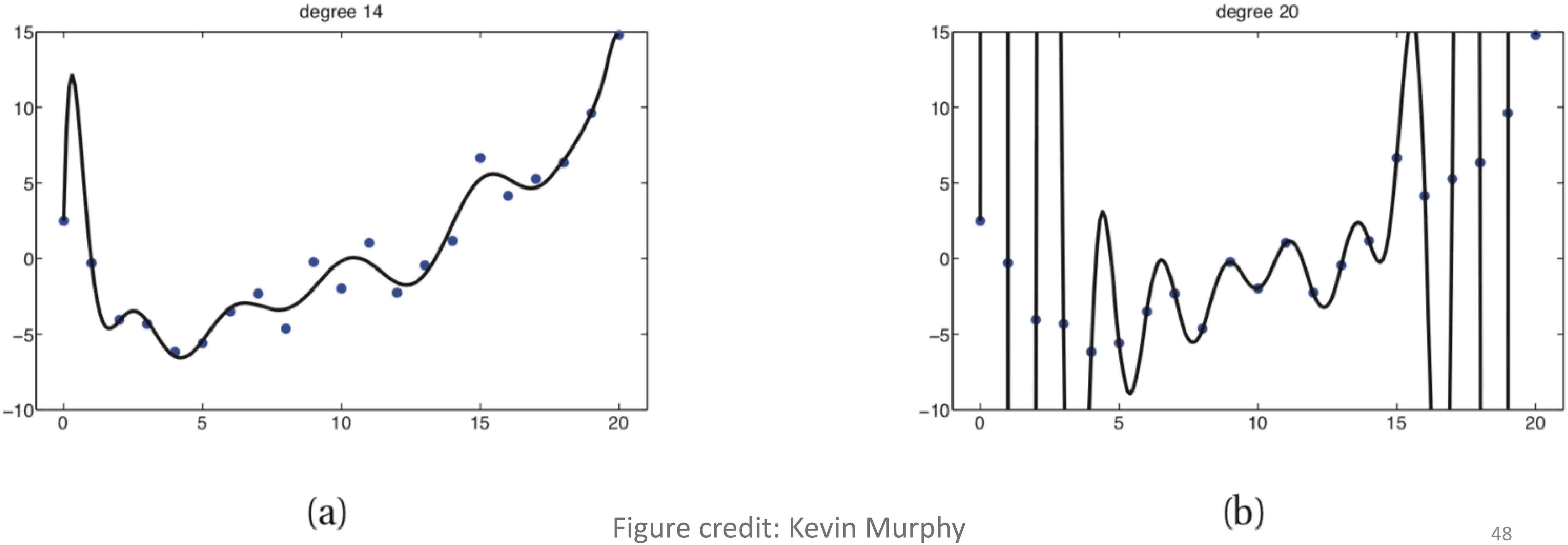


Figure credit: Kevin Murphy

Logistic Regression

From linear regression to logistic regression

- Logistic regression
  - Similar to linear regression
    - Given the numerical features of a sample, predict the numerical label value
    - E.g. given the size, weight, and thickness of the cell wall, predict the age of the cell
  - The values y we now want to predict take on only a small number of discrete values
    - E.g. to predict the cell is benign or malignant

Example

- Given the data of cancer cells below, how to predict they are benign or malignant?

Id	Cl.thickness	Cell.size	Cell.shape	Marg.adhesion	Epith.c.size	Bare.nuclei	Bl.cromatin	Normal.nucleoli	Mitoses	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	benign
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10	9	7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	1	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign
1035283	1	1	1	1	1	1	3	1	1	benign
1036172	2	1	1	1	2	1	2	1	1	benign
1041801	5	3	3	3	2	3	4	4	1	malignant

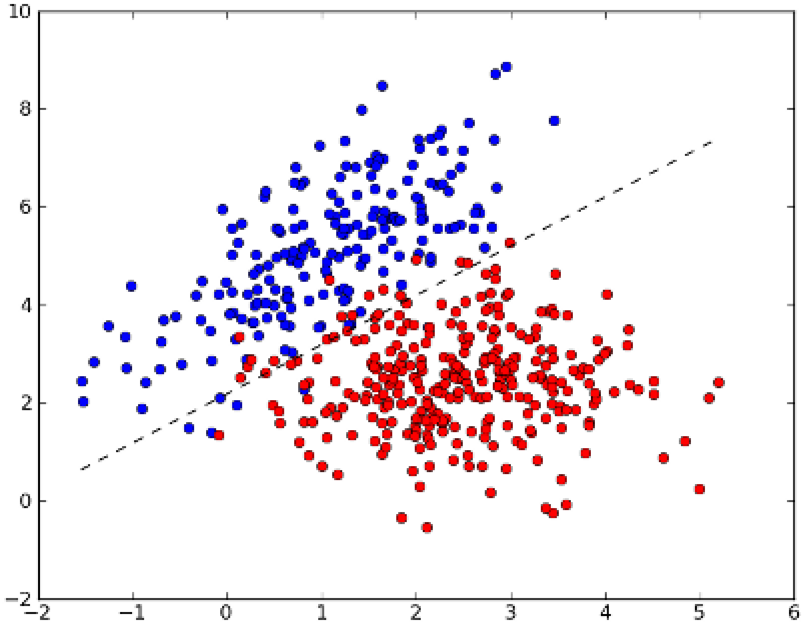
Logistics regression

- It is a Classification problem
  - Compared to regression problem, which predicts the labels from many numerical features
- Many applications
  - Spam Detection: Predicting if an email is Spam or not based on word frequencies
  - Credit Card Fraud: Predicting if a given credit card transaction is fraud or not based on their previous usage
  - Health: Predicting if a given mass of tissue is benign or malignant
  - Marketing: Predicting if a given user will buy an insurance product or not

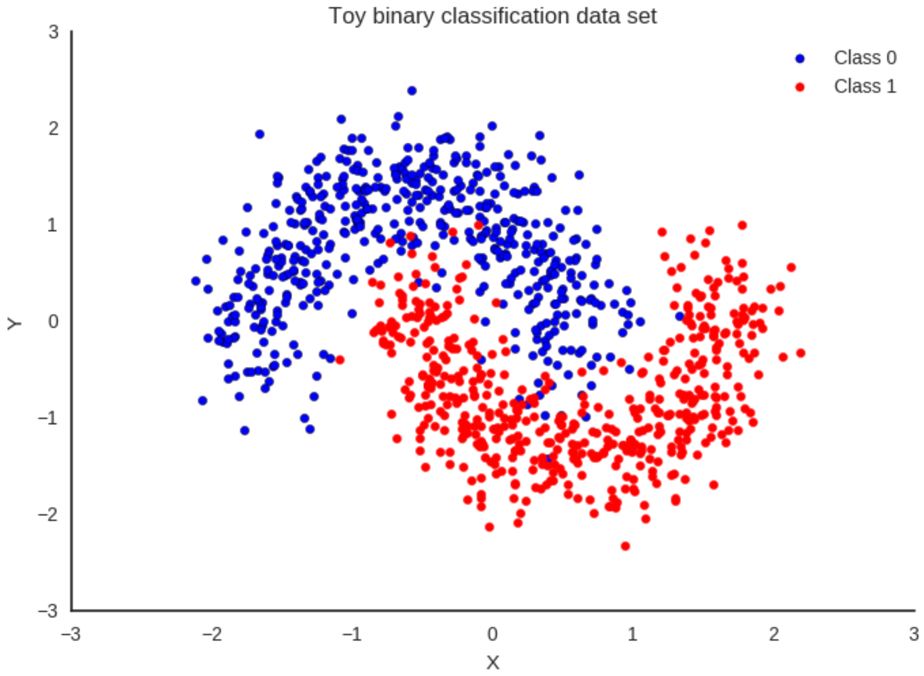
Classification problem

- Given:
  - A description of an instance $x \in X$
  - A fixed set of categories: $C = \{c_1, c_2, \ldots, c_m\}$
- Determine:
  - The category of $x : f(x) \in C$ where $f(x)$ is a categorization function whose domain is $\mathbb{X}$ and whose range is C
  - If the category set binary, i.e. $C = \{0, 1\}$ ({false, true}, {negative, positive}) then it is called binary classification

Binary classification



Linearly separable



Nonlinearly separable

Linear discriminative model

- Discriminative model
  - modeling the dependence of unobserved variables on observed ones
  - also called conditional models.
  - Deterministic: $y = f_\theta(x)$
  - **Probabilistic**: $p_\theta(y|x)$

- For binary classification
  - $p_\theta(y = 1 \mid x)$
  - $p_\theta(y = 0 \mid x) = 1 - p_\theta(y = 1 \mid x)$

Loss Functions

KL divergence

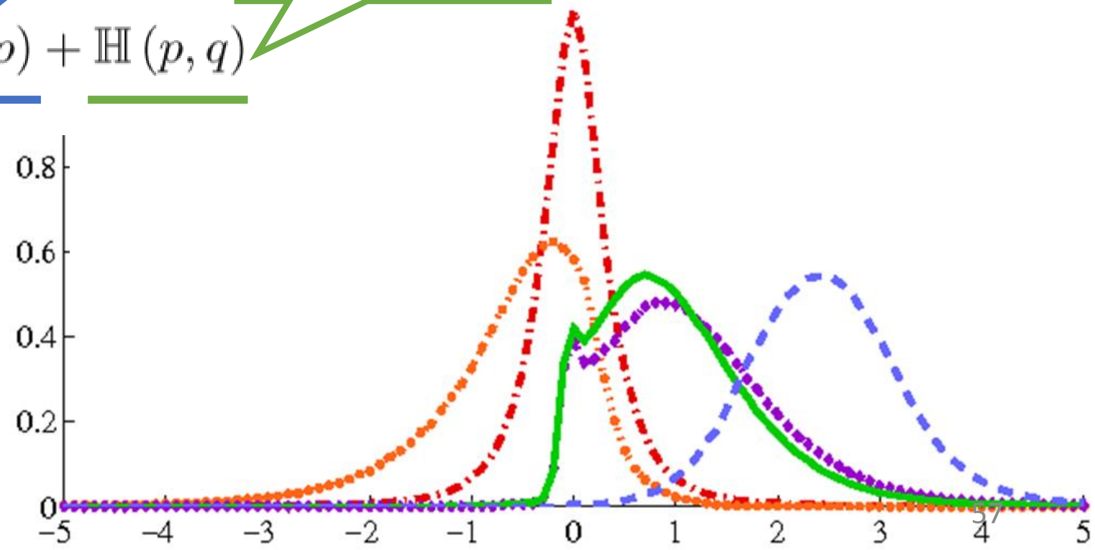
- Regression: mean squared error (MSE)
- Kullback-Leibler divergence (KL divergence)
  - Measure the dissimilarity of two probability distributions

$$\mathbb{KL}\left(p||q\right) \triangleq \sum_{k=1}^{K} p_k \log \frac{p_k}{q_k}$$

Entropy

Cross entropy

$$\mathbb{KL}\left(p||q\right) = \sum_{k} p_k \log p_k - \sum_{k} p_k \log q_k = -\mathbb{H}\left(p\right) + \mathbb{H}\left(p, q\right)$$



Question:
Which one is more similar to normal distribution?

KL divergence (cont.)

- Information inequality

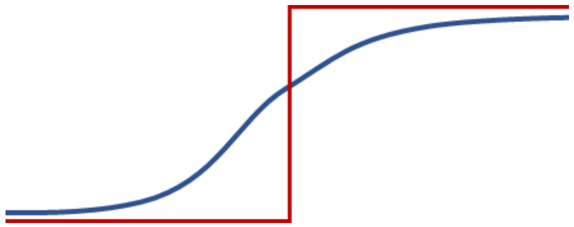
$$\mathbb{KL}(p||q) \geq 0 \textit{ with equality iff } p = q.$$

- Entropy
  - $\mathbb{H}(X) \triangleq -\sum_{k=1}^{K} p(X=k) \log_2 p(X=k)$
  - Is a measure of the uncertainty
  - Discrete distribution with the maximum entropy is the uniform distribution

- Cross entropy
  - $\mathbb{H}(p, q) \triangleq -\sum_{k} p_k \log q_k$
  - Is the average number of bits needed to encode data coming from a source with distribution p when we use model q to define our codebook

Cross entropy loss

- Cross entropy
  - Discrete case: $H(p, q) = -\sum_x p(x) \log q(x)$
  - Continuous case: $H(p, q) = -\int_x \ p(x) \log q(x)$
- Cross entropy loss in classification:
  - Red line p: the ground truth label distribution.
  - Blue line q: the predicted label distribution.



Example for binary classification

- Cross entropy: $H(p, q) = -\sum_x p(x) \log q(x)$

- Given a data point $(x, 0)$ with prediction probability
$$q_\theta(y = 1|x) = 0.4$$
the cross entropy loss on this point is
$$L = -p(y = 0|x) \log q_\theta(y = 0|x) - p(y = 1|x) \log q_\theta(y = 1|x)$$
$$= -\log(1 - 0.4) = \log \frac{5}{3}$$

- What is the cross entropy loss for data point $(x, 1)$ with prediction probability
$$q_\theta(y = 1|x) = 0.3$$

Cross entropy loss for binary classification

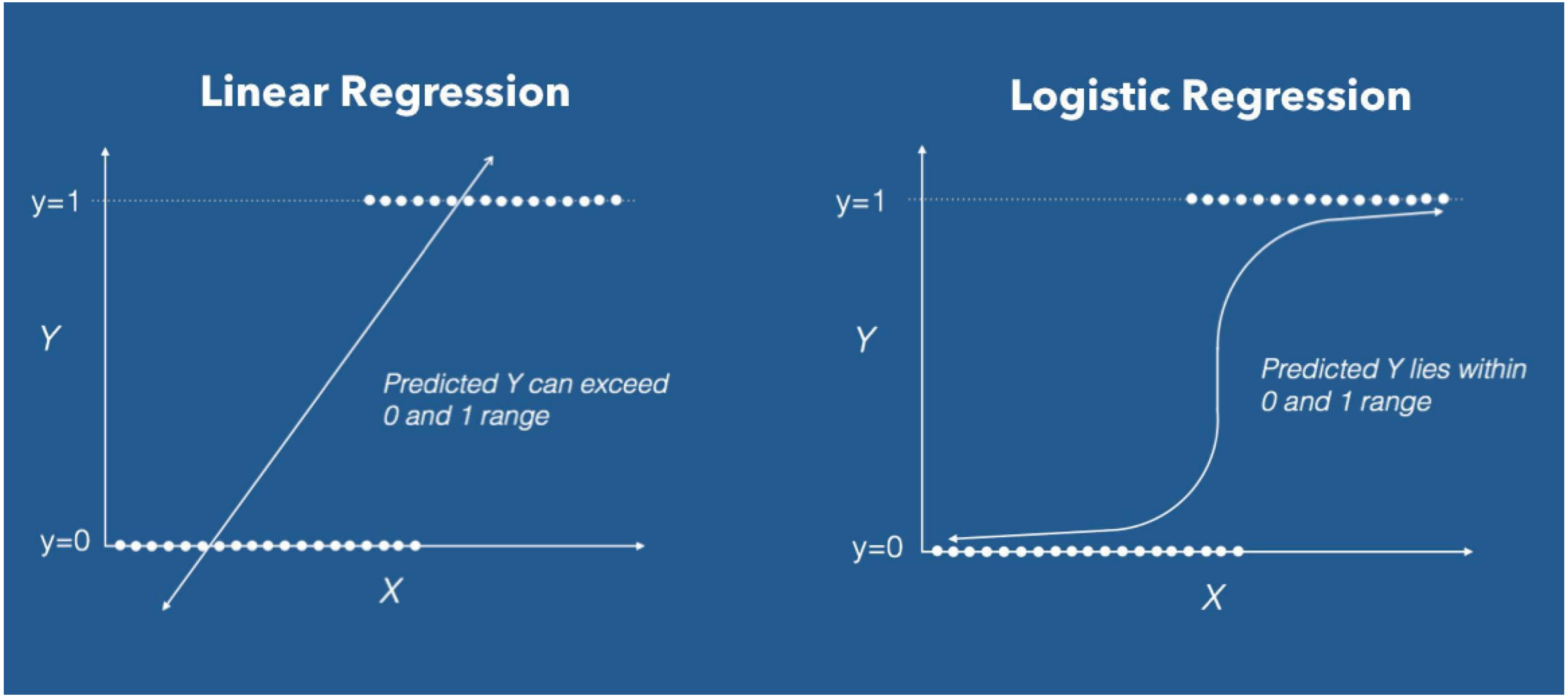
- Loss function for data point (x, y) with prediction model
$$p_\theta(\cdot\,|x)$$
is
$$L(y, x, p_\theta)$$
$$= -1_{y=1} \log p_\theta(1|x) - 1_{y=0} \log p_\theta(0|x)$$
$$= -y \log p_\theta(1|x) - (1 - \mathrm{y}) \log\left(1 - p_\theta(1|x)\right)$$

Cross entropy loss for multiple classification

- Loss function for data point (x, y) with prediction model
$$p_\theta(\cdot\,|x)$$
is
$$L(y, x, p_\theta) = -\sum_{i=1}^{m} 1_{y=C_k} \log p_\theta(C_k|x)$$

Binary Classification

Binary classification: linear and logistic

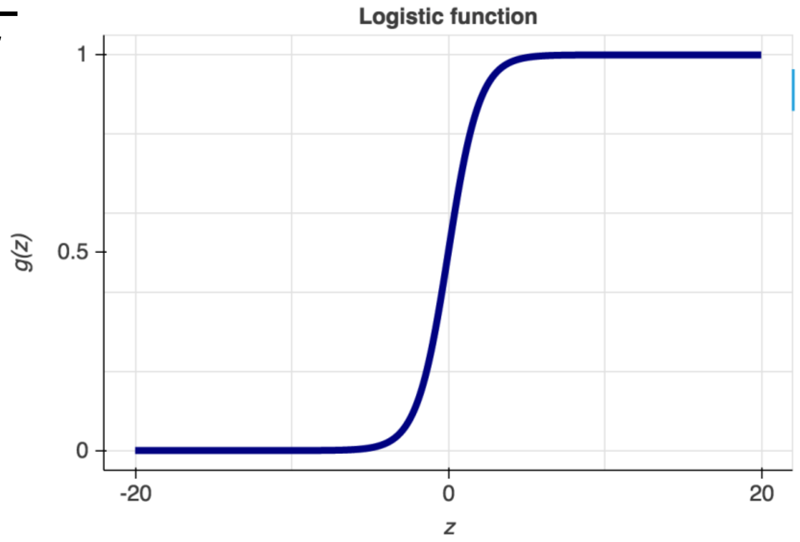


Binary classification: linear and logistic

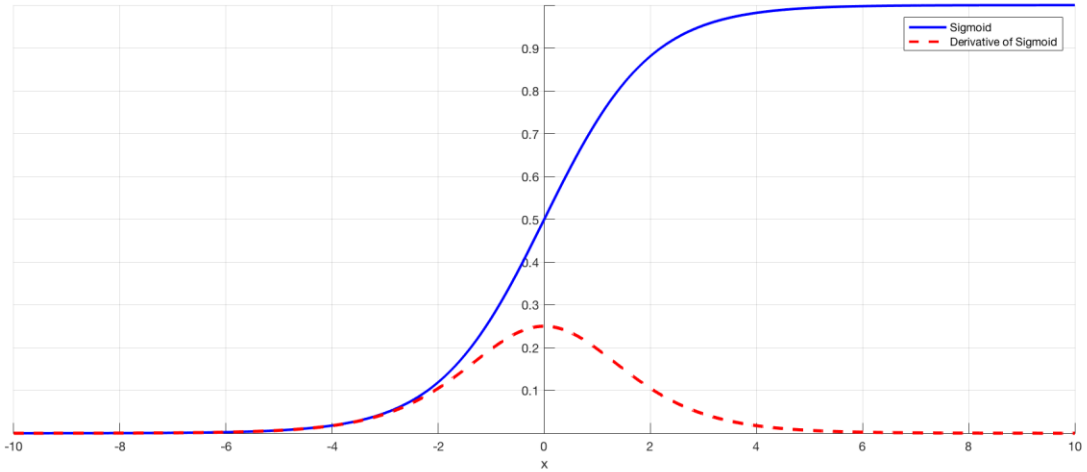
- Linear regression:
  - Target is predicted by $h_\theta(x) = \theta^\top x$

- Logistic regression
  - Target is predicted by $h_\theta(x) = \sigma(\theta^\top x) = \dfrac{1}{1+ e^{-\theta^\top x}}$

    where
    $$\sigma(z) = \frac{1}{1 + e^{-z}}$$
    is the logistic function or the sigmoid function



Properties for the sigmoid function



- $\sigma(z) = \dfrac{1}{1+e^{-z}}$
  - Bounded in $(0,1)$
  - $\sigma(z) \to 1$ when $z \to \infty$
  - $\sigma(z) \to 0$ when $z \to -\infty$
- $\underline{\sigma'(z)} = \dfrac{d}{dz}\dfrac{1}{1+e^{-z}} = -(1+e^{-z})^{-2} \cdot (-e^{-z})$

$$= \frac{1}{1+e^{-z}} \frac{e^{-z}}{1+e^{-z}}$$

$$= \frac{1}{1+e^{-z}}\left(1 - \frac{1}{1+e^{-z}}\right)$$

$$\underline{= \sigma(z)\big(1-\sigma(z)\big)}$$

Logistic regression

- Binary classification

$$p_\theta(y=1|x) = \sigma(\theta^\top x) = \frac{1}{1+e^{-\theta^\top x}}$$

$$p_\theta(y=0|x) = \frac{e^{-\theta^\top x}}{1+e^{-\theta^\top x}}$$

- Cross entropy loss function

is also convex in θ

$$\mathcal{L}(y, x, p_\theta) = -y \log \sigma(\theta^\top x) - (1-y)\log(1-\sigma(\theta^\top x))$$

- Gradient

$$\frac{\partial \mathcal{L}(y, x, p_\theta)}{\partial \theta} = -y\frac{1}{\sigma(\theta^\top x)}\sigma(z)(1-\sigma(z))x - (1-y)\frac{-1}{1-\sigma(\theta^\top x)}\sigma(z)(1-\sigma(z))x$$

$$= (\sigma(\theta^\top x) - y)x$$

$$\theta \leftarrow \theta + \eta(y - \sigma(\theta^\top x))x$$

$$\boxed{\frac{\partial \sigma(z)}{\partial z} = \sigma(z)(1-\sigma(z))}$$

$$\theta_{\text{new}} \leftarrow \theta_{\text{old}} - \eta\frac{\partial \mathcal{L}(\theta)}{\partial \theta}$$

Label decision

- Logistic regression provides the probability

$$p_\theta(y=1|x) = \sigma(\theta^\top x) = \frac{1}{1+e^{-\theta^\top x}}$$

$$p_\theta(y=0|x) = \frac{e^{-\theta^\top x}}{1+e^{-\theta^\top x}}$$

- The final label of an instance could be decided, for example, by setting a threshold h

$$\hat{y} = \begin{cases} 1, & p_\theta(y=1|x) > h \\ 0, & \text{otherwise} \end{cases}$$

Threshold is trade-off

- Precision-recall trade-off
  - $\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}$
  - $\text{Recall} = \frac{\text{TP}}{\text{TP+FN}}$
  - Higher threshold
    - More FN and less FP
    - Higher precision
    - Lower recall
  - Lower threshold
    - More FP and less FN
    - Lower precision
    - Higher recall

$$\hat{y} = \begin{cases} 1, & p_\theta(y = 1|x) > h \\ 0, & \text{otherwise} \end{cases}$$

Example

- We have the heights and weights of a group of students
    - Height: in inches,
    - Weight: in pounds
    - Male: 1, female, 0

- Please build a logistic regression model to predict their genders

```
"Height","Weight","Male"
73.847017017515,241.893563180437,1
68.7819040458903,162.3104725213,1
74.1101053917849,212.7408555565,1
71.7309784033377,220.042470303077,1
69.8817958611153,206.349800623871,1
67.2530156878065,152.212155757083,1
68.7850812516616,183.927888604031,1
68.3485155115879,167.971110489509,1
67.018949662883,175.92944039571,1
63.4564939783664,156.399676387112,1
...
63.1794982498071,141.266099582434,0
62.6366749337994,102.85356321483,0
62.0778316936514,138.691680275738,0
60.0304337715611,97.6874322554917,0
59.0982500313486,110.529685683049,0
66.1726521477708,136.777454183235,0
67.067154649054,170.867905890713,0
63.8679922137577,128.475318784122,0
69.0342431307346,163.852461346571,0
61.9442458795172,113.649102675312,0
```

Example (cont.)

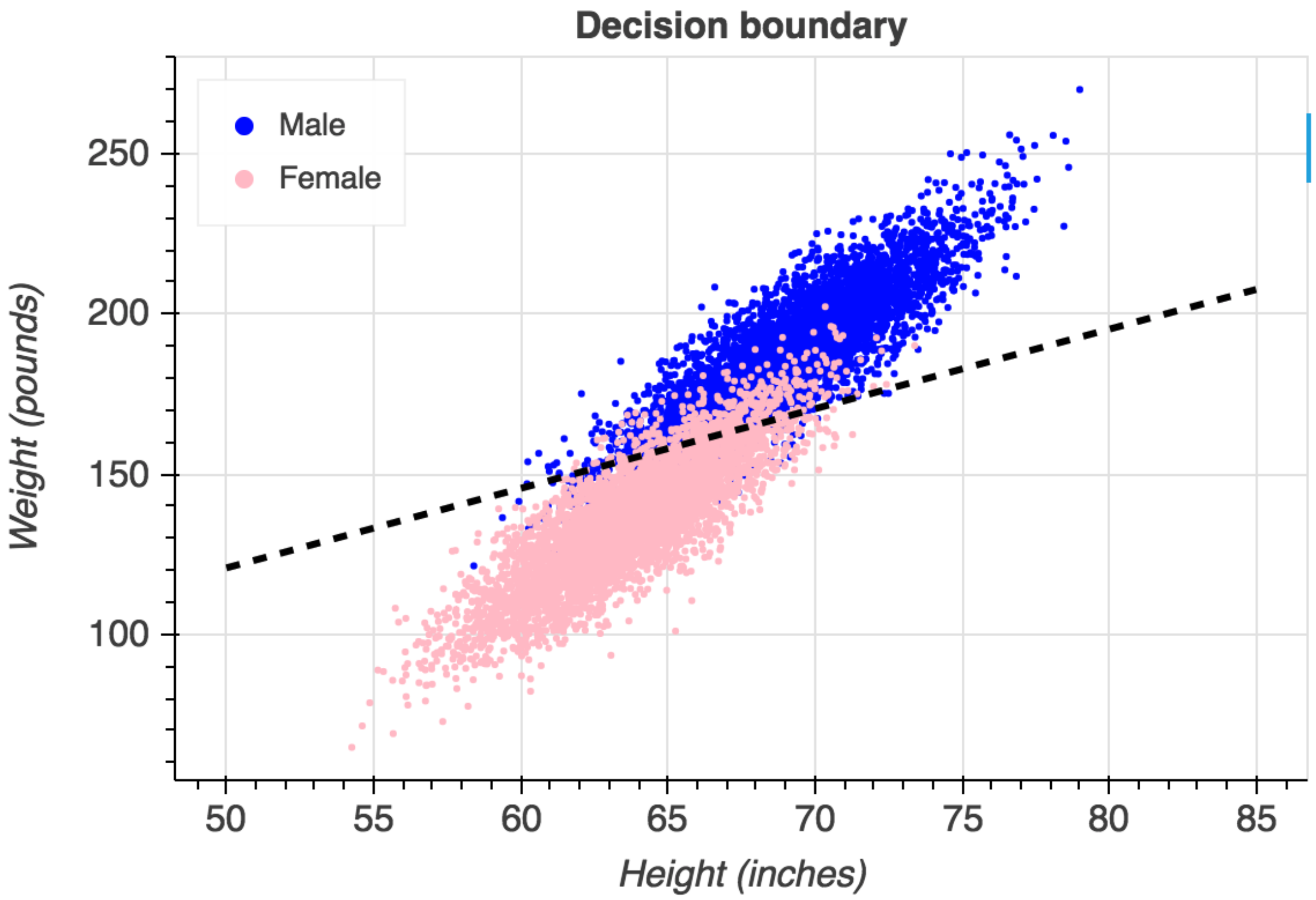
- As there are only two features, height and weight, the logistic regression equation is: $h_\theta(x) = \frac{1}{1+e^{-(\theta_0+\theta_1 x_1+\theta_2 x_2)}}$

- Solve it by gradient descent

There will be a lab/hw on logistic regression

- The solution is $\theta = \begin{bmatrix} 0.69254 \\ -0.49269 \\ 0.19834 \end{bmatrix}$

Example (cont.)



- Threshold $h = 0.5$
- Decision boundary is
$$\theta_0 + \theta_1 x_1 + \theta_2 x_2 = 0$$

- Above the decision boundary lie most of the blue points that correspond to the Male class, and below it all the pink points that correspond to the Female class.

- The predictions won't be perfect and can be improved by including more features (beyond weight and height), and by potentially using a different decision boundary (e.g. nonlinear)

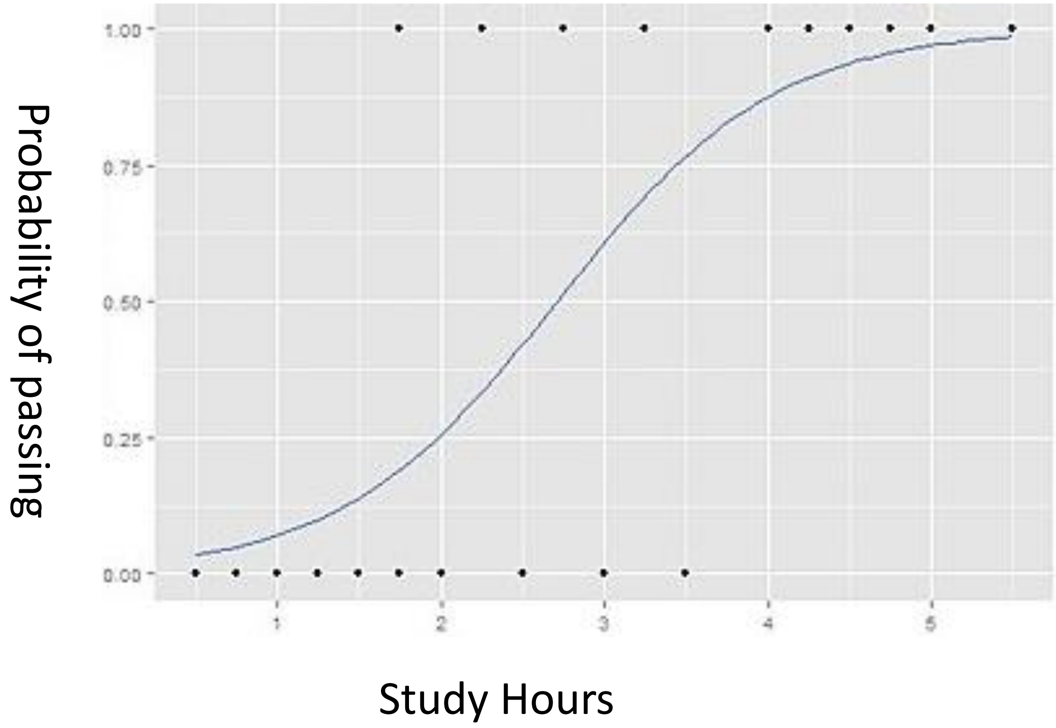
Example 2

- A group of 20 students spends between 0 and 6 hours studying for an exam. How does the number of hours spent studying affect the probability of the student passing the exam?

Hours	Pass		Hours	Pass
0.50	0		2.75	1
0.75	0		3.00	0
1.00	0		3.25	1
1.25	0		3.50	0
1.50	0		4.00	1
1.75	0		4.25	1
1.75	1		4.50	1
2.00	0		4.75	1
2.25	1		5.00	1
2.50	0		5.50	1

Example 2 (cont.)

- $h_\theta(x) = \frac{1}{1+e^{-(1.5046*hours-4.0777)}}$



Interpretation of logistic regression

- Given a probability p, the odds of p is defined as $odds = \frac{p}{1-p}$
- The logit is defined as the log of the odds: $\ln(odds) = \ln\left(\frac{p}{1-p}\right)$
- Let $\ln(odds) = \theta^\top x$, then we will have $\ln\left(\frac{p}{1-p}\right) = \theta^\top x$, and

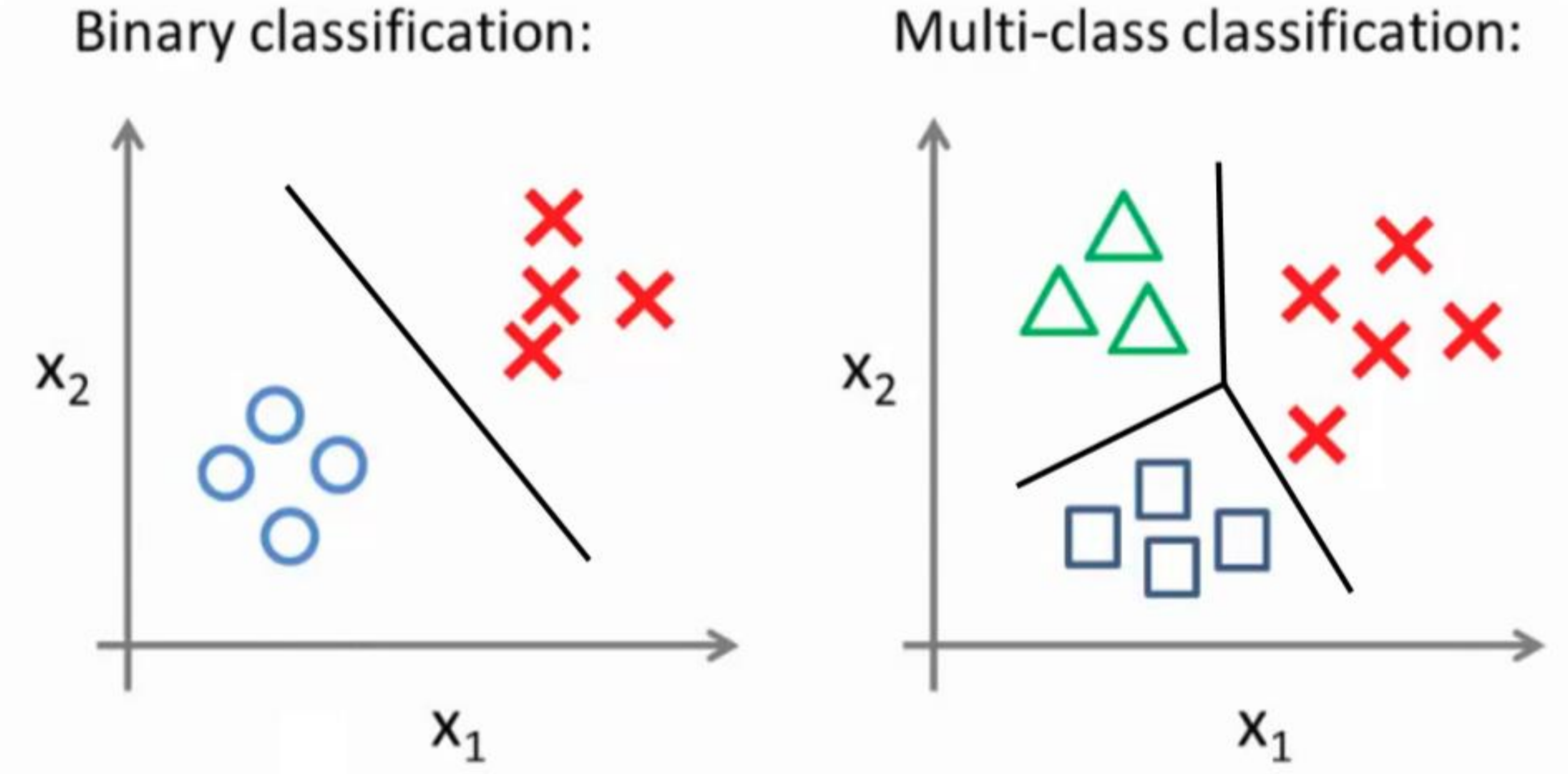
$$p = \frac{1}{1 + e^{-\theta^\top x}}$$

- So in logistic regression, the logit of an event (predicted positive)'s probability is defined as a result of linear regression

Multi-Class Logistic Regression

Multi-class classification

- $L(y, x, p_\theta) = -\sum_{i=1}^{m} 1_{y=C_k} \log p_\theta(C_k|x)$



Multi-Class Logistic Regression

- Class set $C = \{c_1, c_2, \dots, c_m\}$

- Predicting the probability of $p_\theta(y = c_j|x)$

$$p_\theta(y = c_j|x) = \frac{e^{\theta_j^\top x}}{\sum_{k=1}^m e^{\theta_k^\top x}} \quad \text{for } j = 1, \dots, m$$

- Softmax
  - Parameters $\theta = \{\theta_1, \theta_2, \dots, \theta_m\}$
  - Can be normalized with m-1 groups of parameters

Multi-Class Logistic Regression 2

- Learning on one instance $(x, y = c_j)$
  - Maximize log-likelihood

$$\max_{\theta} \log p_\theta(y = c_j | x)$$

  - Gradient

$$\begin{aligned}\frac{\partial \log p_\theta(y = c_j|x)}{\partial \theta_j} &= \frac{\partial}{\partial \theta_j} \log \frac{e^{\theta_j^\top x}}{\sum_{k=1}^m e^{\theta_k^\top x}} \\ &= x - \frac{\partial}{\partial \theta_j} \log \sum_{k=1}^m e^{\theta_k^\top x} \\ &= x - \frac{e^{\theta_j^\top x} x}{\sum_{k=1}^m e^{\theta_k^\top x}}\end{aligned}$$

Summary

Shuai Li

https://shuaili8.github.io

- Linear regression
  - Normal equation
  - Gradient methods
  - Examples
  - Probabilistic view
  - Applications
  - Regularization
- Logistic regression (binary classification)
  - Cross entropy
  - Formulation, sigmoid function
  - Training—gradient descent
- Multi-class logistic regression

Questions?