

Lecture 5: Logistic Regression

Shuai Li

John Hopcroft Center, Shanghai Jiao Tong University

<https://shuaili8.github.io>

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

Outline

- Discriminative / Generative Models
- Logistic regression (binary classification)
 - Cross entropy
 - Formulation, sigmoid function
 - Training—gradient descent
- More measures for binary classification (AUC, AUPR)
- Class imbalance
- Multi-class logistic regression

Discriminative / Generative Models

Discriminative / Generative Models

- Discriminative models
 - Modeling the **dependence** of unobserved variables on observed ones
 - also called conditional models.
 - Deterministic: $y = f_{\theta}(x)$
 - Probabilistic: $p_{\theta}(y|x)$
- Generative models
 - Modeling the **joint** probabilistic distribution of data
 - Given some hidden parameters or variables

$$p_{\theta}(x, y)$$

- Then do the conditional inference

$$p_{\theta}(y|x) = \frac{p_{\theta}(x, y)}{p_{\theta}(x)} = \frac{p_{\theta}(x, y)}{\sum_{y'} p_{\theta}(x, y')}$$

Discriminative Models

- Discriminative models
 - Modeling the **dependence** of unobserved variables on observed ones
 - also called conditional models.
 - Deterministic: $y = f_{\theta}(x)$
 - Probabilistic: $p_{\theta}(y|x)$
- Directly model the dependence for label prediction
- Easy to define dependence on specific features and models
- Practically yielding higher prediction performance
- E.g. linear regression, logistic regression, k nearest neighbor, SVMs, (multi-layer) perceptrons, decision trees, random forest

Generative Models

- Generative models

- Modeling the **joint** probabilistic distribution of data
- Given some hidden parameters or variables

$$p_{\theta}(x, y)$$

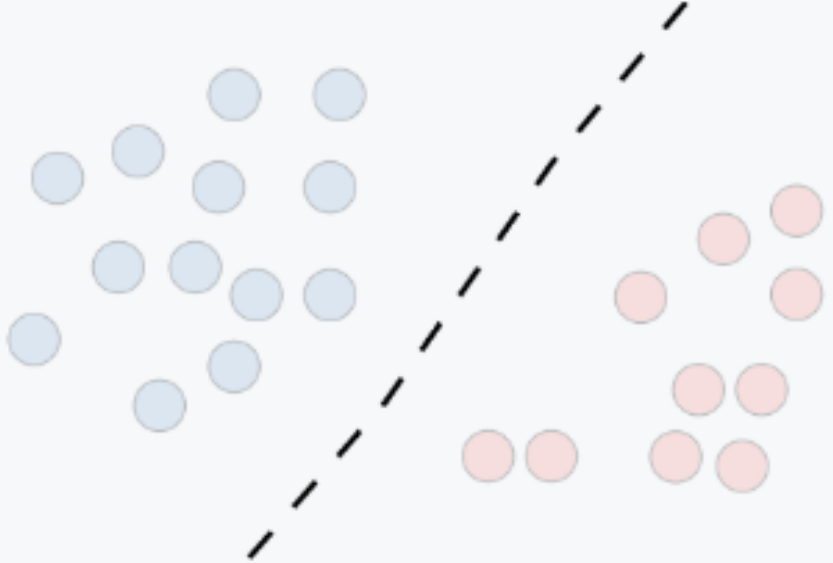
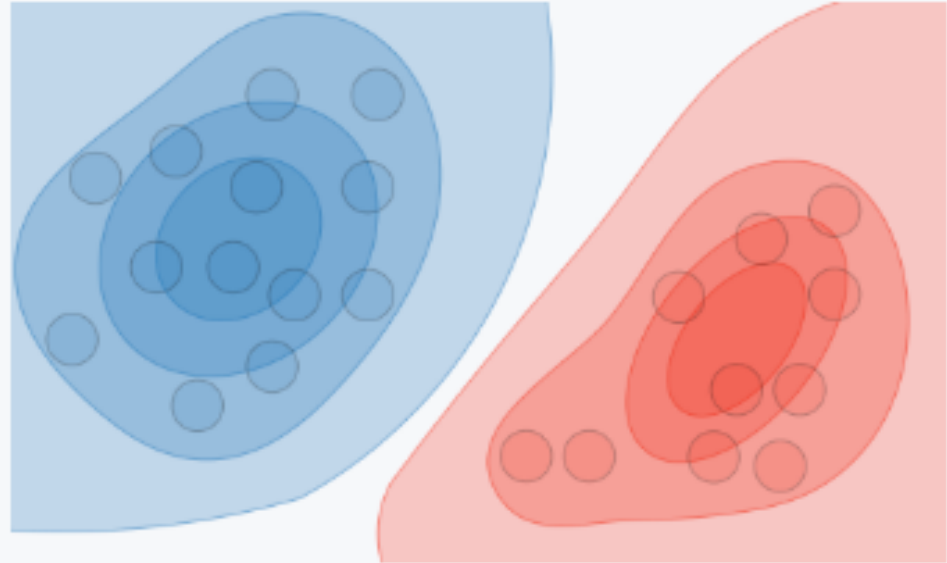
- Then do the conditional inference

$$p_{\theta}(y|x) = \frac{p_{\theta}(x, y)}{p_{\theta}(x)} = \frac{p_{\theta}(x, y)}{\sum_{y'} p_{\theta}(x, y')}$$

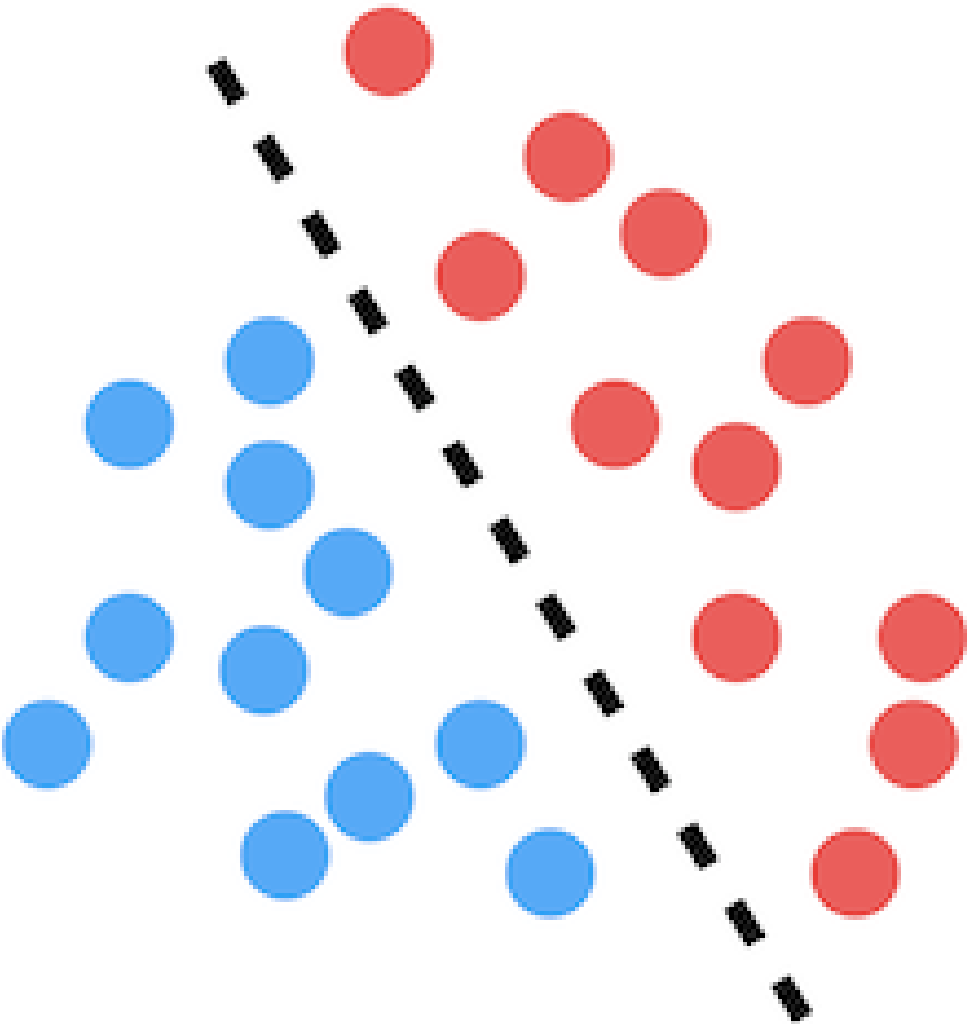
- Recover the data distribution [essence of data science]
- Benefit from hidden variables modeling
- E.g. Naive Bayes, Hidden Markov Model, Mixture Gaussian, Markov Random Fields, Latent Dirichlet Allocation

Discriminative Models vs Generative Models

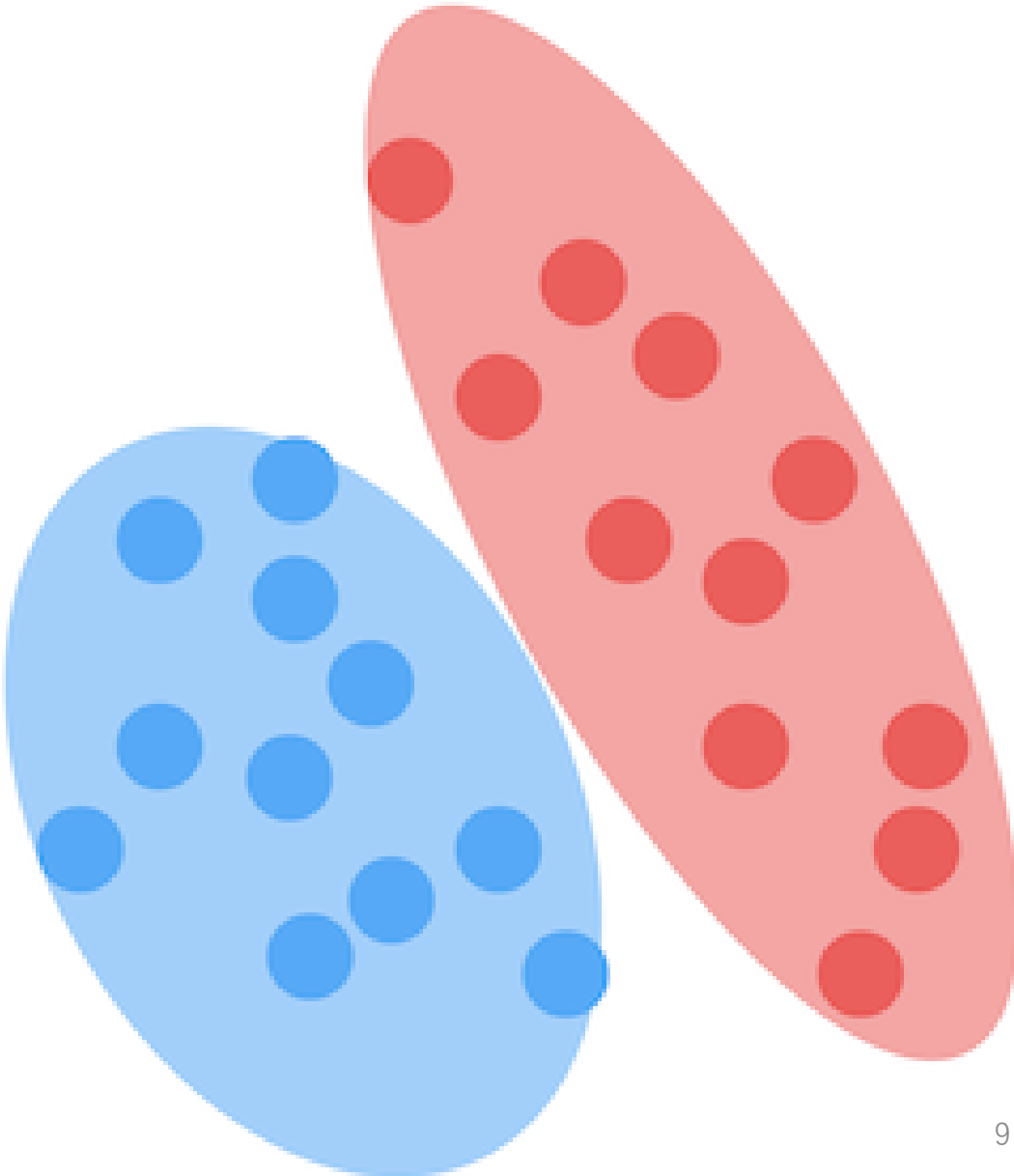
- In General
 - A Discriminative model models the **decision boundary between the classes**
 - A Generative Model explicitly models the **actual distribution of each class**
- Example: Our training set is a bag of fruits. Only **apples** and **oranges** Each labeled. Imagine a post-it note stuck to the fruit
 - A generative model will model various attributes of fruits such as color, weight, shape, etc
 - A discriminative model might model color alone, **should that suffice** to distinguish apples from oranges

	Discriminative model	Generative model
Goal	Directly estimate $P(y x)$	Estimate $P(x y)$ to then deduce $P(y x)$
What's learned	Decision boundary	Probability distributions of the data
Illustration		
Examples	Regressions, SVMs	GDA, Naive Bayes

Discriminative



Generative



Linear Discriminative Models

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

$$y = f_{\theta}(x) = \theta_0 + \sum_{j=1}^d \theta_j x_j = \theta^{\top} x$$

$$x = (1, x_1, x_2, \dots, x_d)$$

Logistic Regression

VS. linear regression

- Logistic regression
 - Similarity with 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

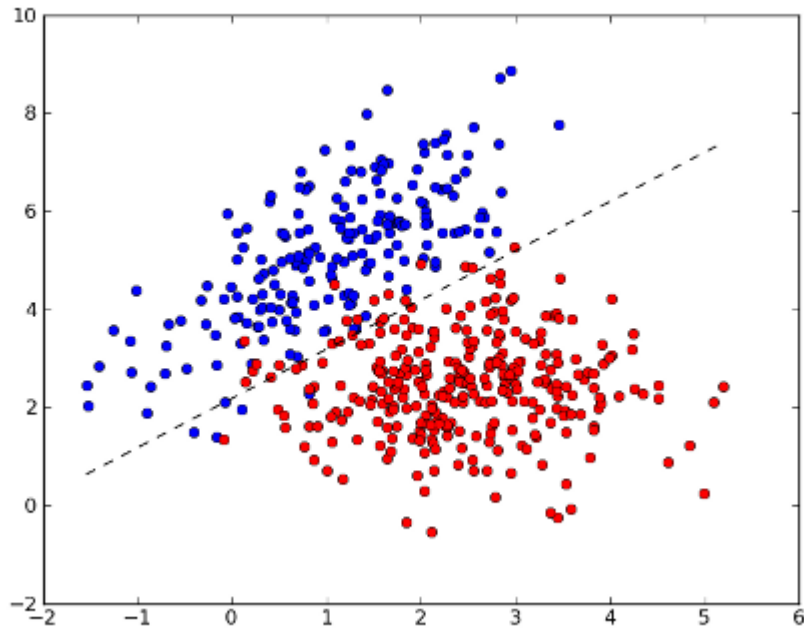
Logistics regression

- 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

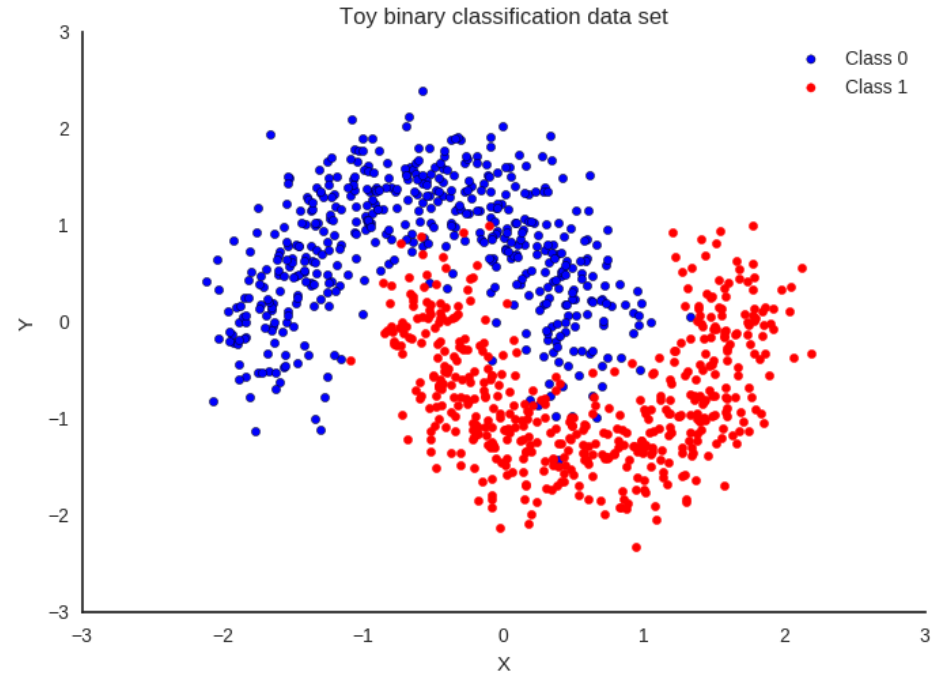
Classification problem

- Given:
 - A description of an instance $x \in X$
 - A fixed set of categories: $C = \{c_1, c_2, \dots, c_m\}$
- Determine:
 - The category of $x: f(x) \in C$ where $f(x)$ is a categorization function whose domain is 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
 - $p_{\theta}(y = 1 | x)$
 - $p_{\theta}(y = 0 | x) = 1 - p_{\theta}(y = 1 | x)$

Loss Functions

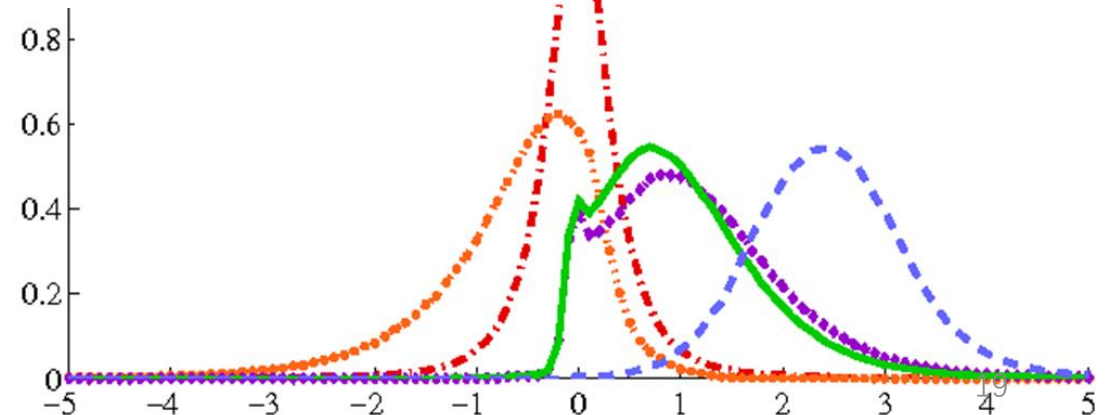
KL divergence

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

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

$$\mathbb{KL}(p||q) = \underbrace{\sum_k p_k \log p_k}_{\text{Entropy}} - \underbrace{\sum_k p_k \log q_k}_{\text{Cross entropy}} = -\mathbb{H}(p) + \mathbb{H}(p, q)$$

Question:
Which one is more similar to
norm distribution?



KL divergence (cont.)

- Information inequality

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

- 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

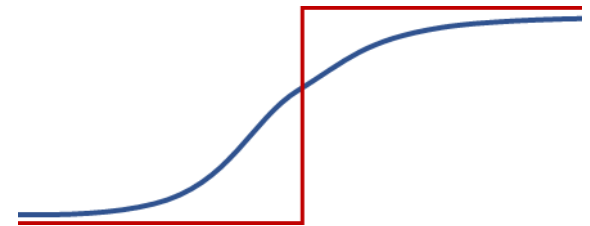
- Cross entropy

- $\mathbb{H}(p, q) \triangleq - \sum_k p_k \log q_k$

- 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

$$\begin{aligned} L &= -p(y = 0|x) \log q_\theta(y = 0|x) - p(y = 1|x) \log q_\theta(y = 1|x) \\ &= -\log(1 - 0.4) = \log \frac{5}{3} \end{aligned}$$

- 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

$$\begin{aligned} L(y, x, p_\theta) &= -1_{y=1} \log p_\theta(1|x) - 1_{y=0} \log p_\theta(0|x) \\ &= -y \log p_\theta(1|x) - (1 - y) \log (1 - p_\theta(1|x)) \end{aligned}$$

Cross entropy loss for multiple classification

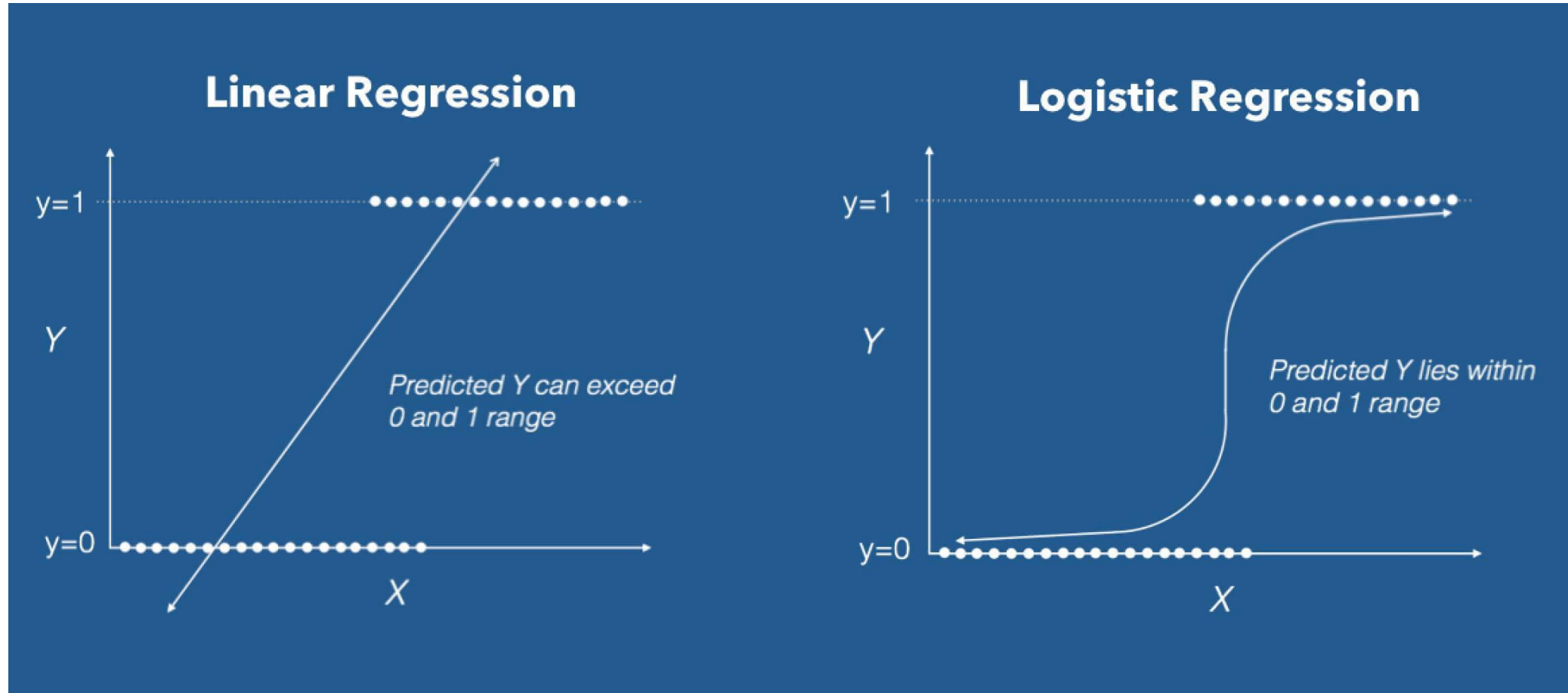
- Loss function for data point (x, y) with prediction model $p_\theta(\cdot | x)$

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 $h_{\theta}(x) = \theta^T x$

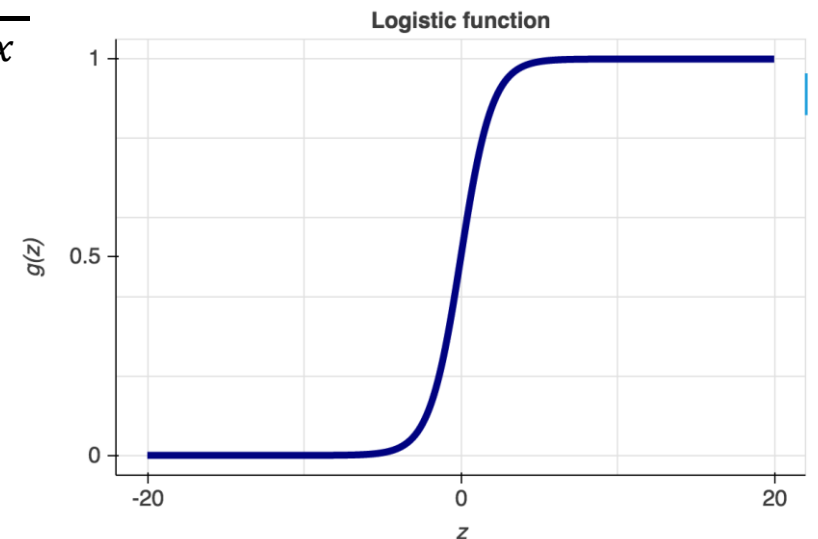
- Logistic regression

- Target is predicted by $h_{\theta}(x) = \sigma(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$

where

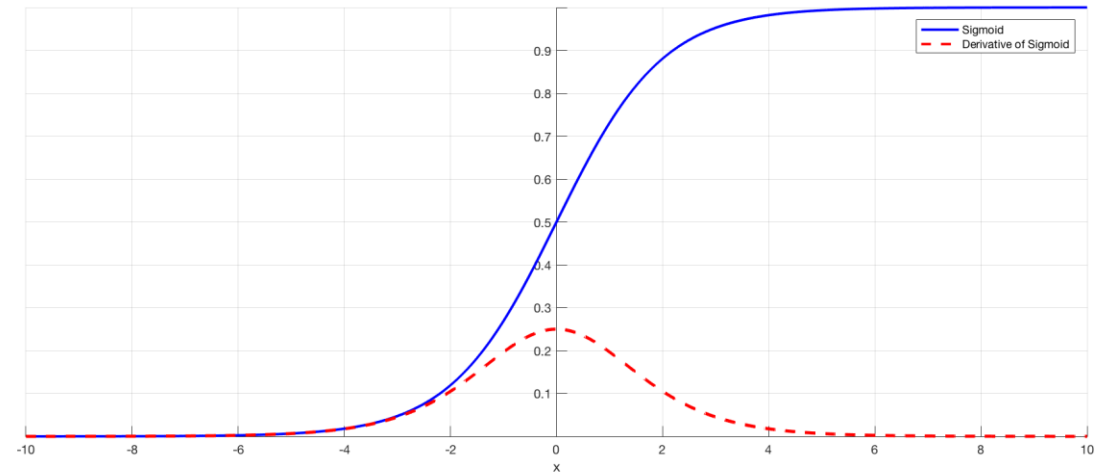
$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

is the **logistic function** or the **sigmoid function**



Properties for the sigmoid function

- $\sigma(z) = \frac{1}{1 + e^{-z}}$
 - Bounded in $(0,1)$
 - $\sigma(z) \rightarrow 1$ when $z \rightarrow \infty$
 - $\sigma(z) \rightarrow 0$ when $z \rightarrow -\infty$



$$\begin{aligned}\sigma'(z) &= \frac{d}{dz} \frac{1}{1 + e^{-z}} = -\frac{1}{(1 + e^{-z})^2} \cdot (-e^{-z}) \\ &= \frac{1}{1 + e^{-z}} \frac{e^{-z}}{1 + e^{-z}} \\ &= \frac{1}{1 + e^{-z}} \left(1 - \frac{1}{1 + e^{-z}} \right) \\ &= \sigma(z)(1 - \sigma(z))\end{aligned}$$

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 convex in θ

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

- Gradient

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

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

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

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

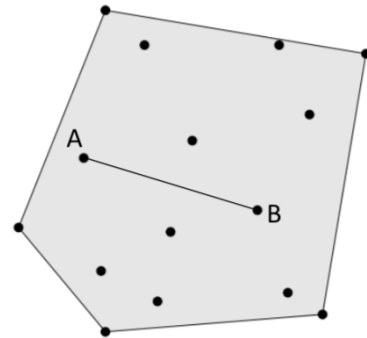
$$\theta_{\text{new}} \leftarrow \theta_{\text{old}} - \eta \frac{\partial \mathcal{L}(\theta)}{\partial \theta}$$

Convex set

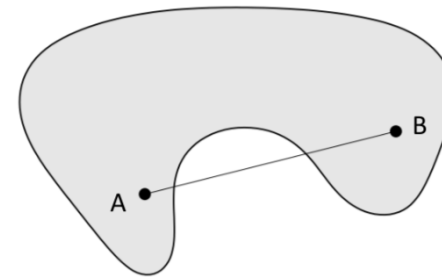
- A convex set S is a set of points such that, given any two points A, B in that set, the line AB joining them lies entirely within S .

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

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

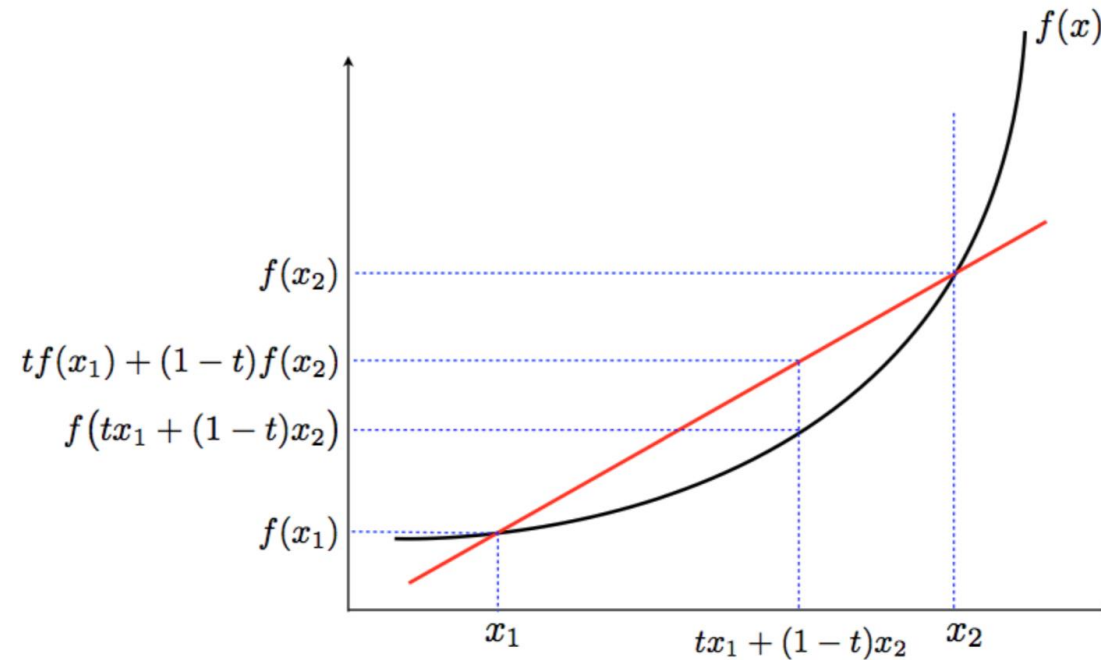


Convex set



Non-convex set

Convex function



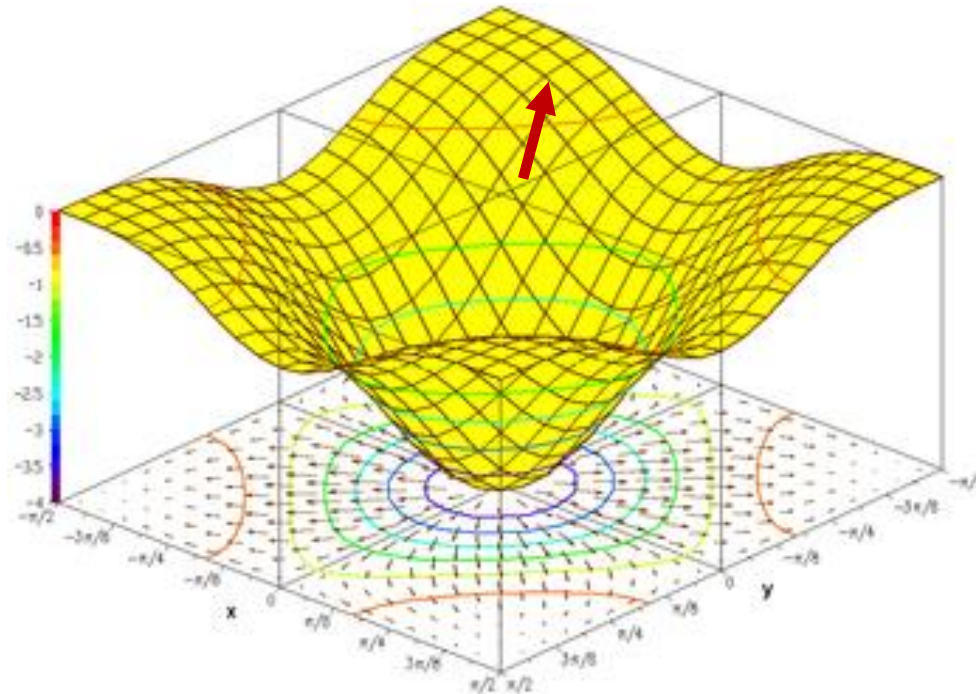
$f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex if **dom** f is a convex set and

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

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

Gradient interpretation

- Gradient is the vector (**the red one**) along which the value of the function increases most rapidly. Thus its opposite direction is where the value decreases most rapidly.



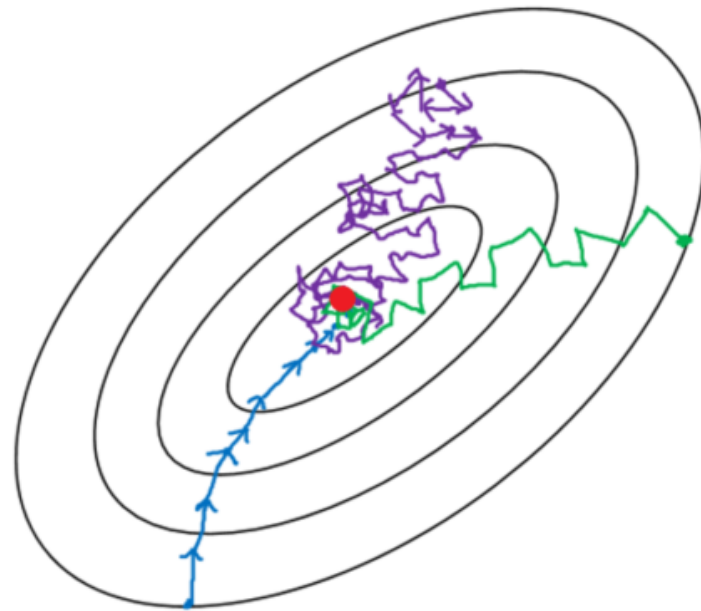
Gradient descent

- To find a (local) minimum of a function using gradient descent, one takes steps **proportional to the negative of the gradient** (or an approximation) of the function at the current point
- For a smooth function $f(x)$, $\frac{\partial f}{\partial x}$ is the direction that f increases most rapidly. So we apply

$$x_{t+1} = x_t - \eta \frac{\partial f}{\partial x}(x_t)$$

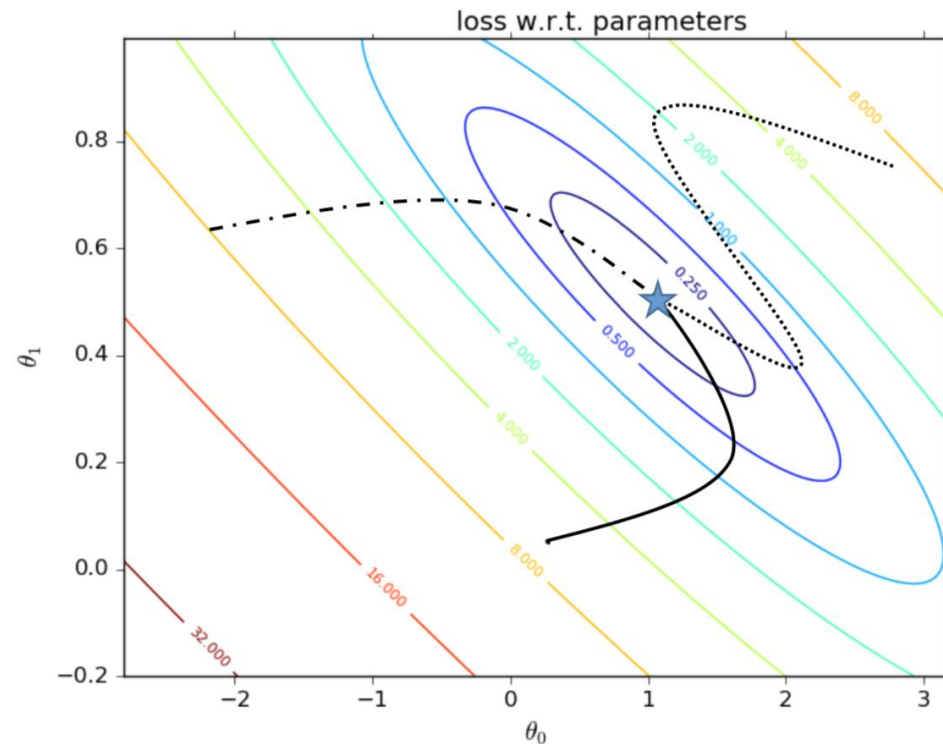
until x converges

Comparisons



- Batch gradient descent
- Mini-batch gradient Descent
- Stochastic gradient descent

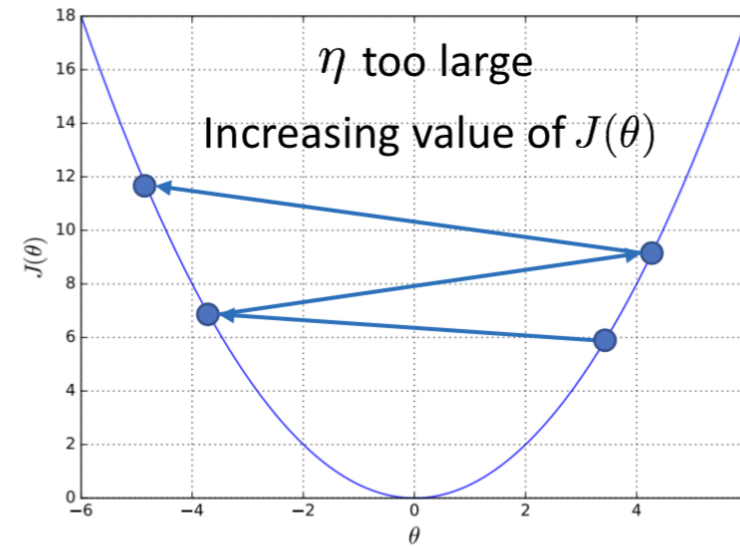
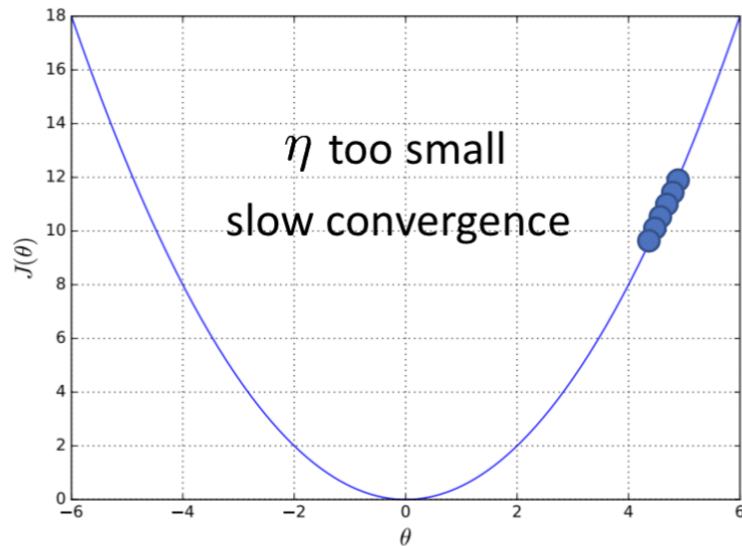
Uniqueness of minimum for convex objectives



- 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

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 is decided by setting a threshold h

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

How to choose the threshold

- Precision-recall trade-off

- Precision = $\frac{TP}{TP+FP}$

- Recall = $\frac{TP}{TP+FN}$

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

- Higher threshold

- More FN and less FP
 - Higher precision
 - Lower recall

- Lower threshold

- More FP and less FN
 - Lower precision
 - Higher recall

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
```

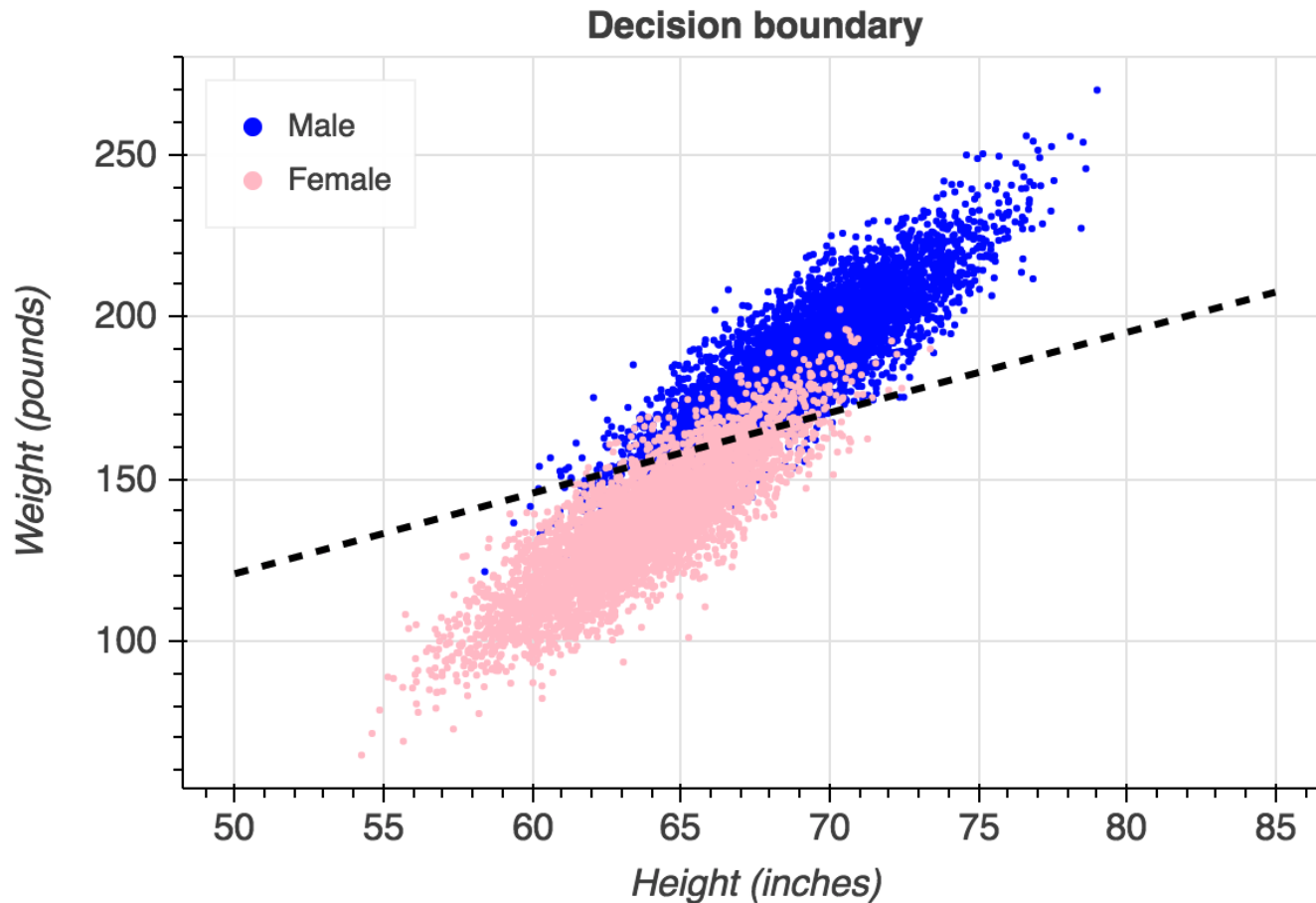
Example (cont.)

- As there are only two features, height and weight, the logistic regression equation is:
$$h_{\theta}(x) = \frac{1}{1+e^{-(\theta_0+\theta_1x_1+\theta_2x_2)}}$$

- Solve it by gradient descent

- 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)

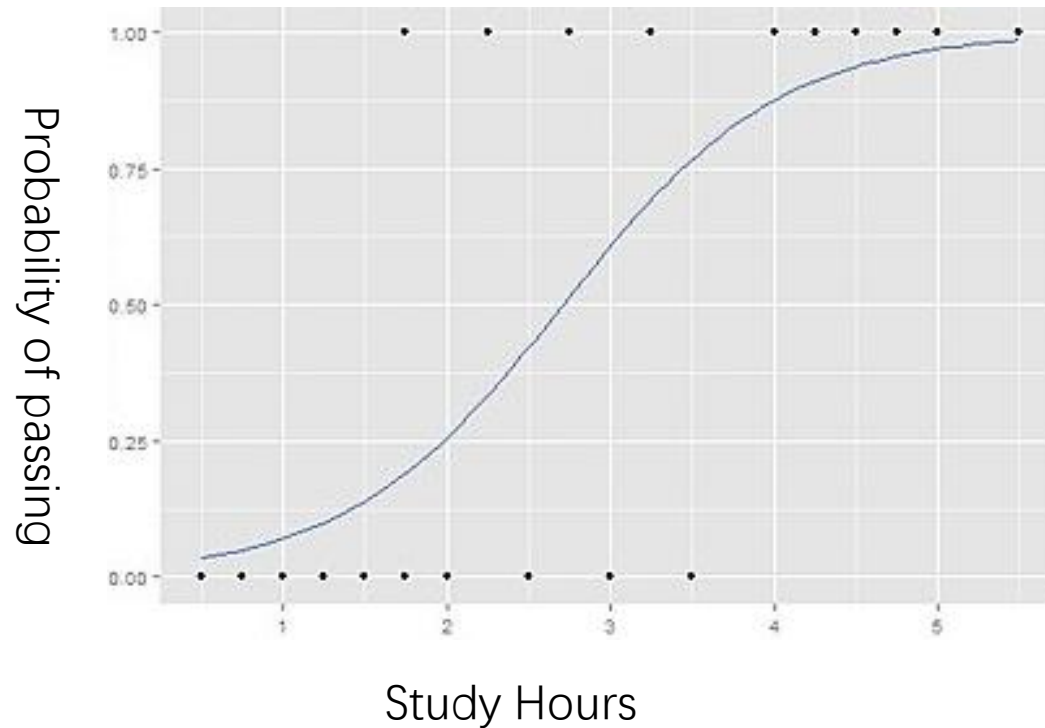
Example 2

- 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.)

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



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 $\ln(odds) = \theta^\top x$, 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

More Measures for Classification

Confusion matrix

- Remember what we have learned about the confusion matrix

		Prediction	
		1	0
Label	1	True Positive	False Negative
	0	False Positive	True Negative

- Precision:** the ratio of true class 1 cases in those with prediction 1

$$\text{Prec} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

		Prediction	
		1	0
Label	1	True Positive	False Negative
	0	False Positive	True Negative

- Recall:** the ratio of cases with prediction 1 in all true class 1 cases

$$\text{Rec} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F1} = \frac{2 \times \text{Prec} \times \text{Recall}}{\text{Prec} + \text{Rec}}$$

- These are the basic metrics to measure the classifier

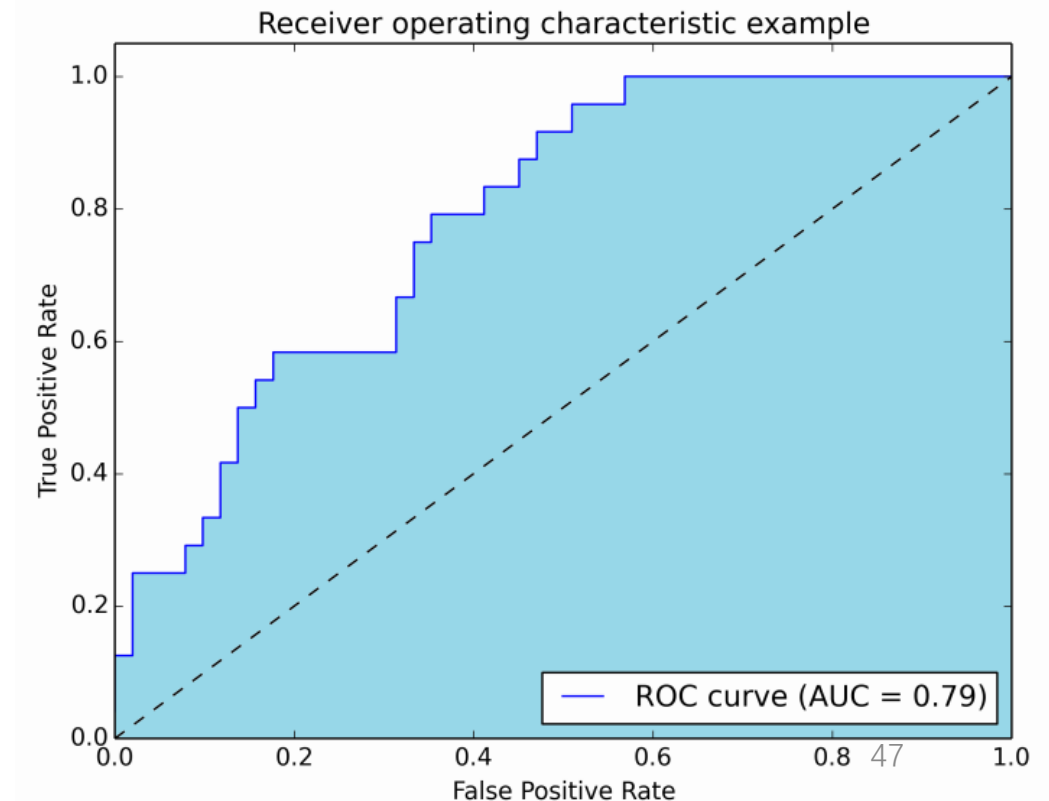
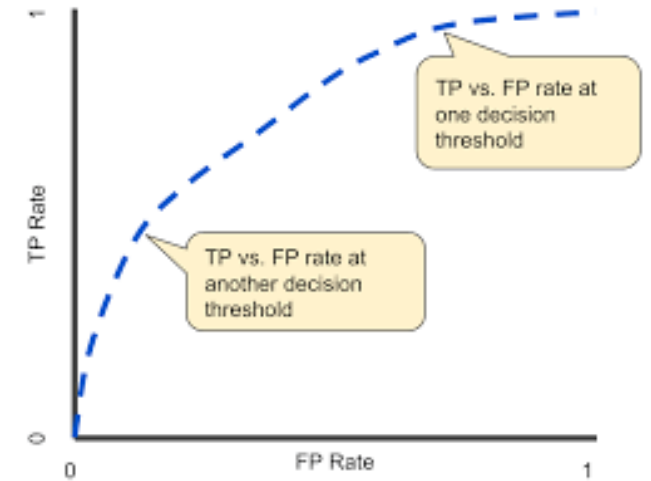
Area Under ROC Curve (AUC)

- A performance measurement for classification problem at various thresholds settings
- Tells how much the model is capable of distinguishing between classes
- **The higher, the better**
- Receiver Operating Characteristic (ROC) Curve

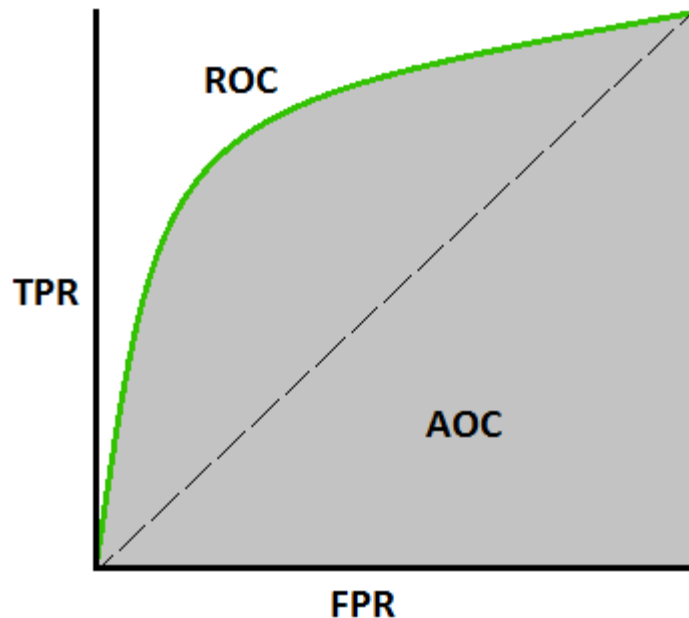
- TPR against FPR

$$\text{TPR/Recall/Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{FPR} = 1 - \text{Specificity} = \frac{\text{FP}}{\text{TN} + \text{FP}}$$



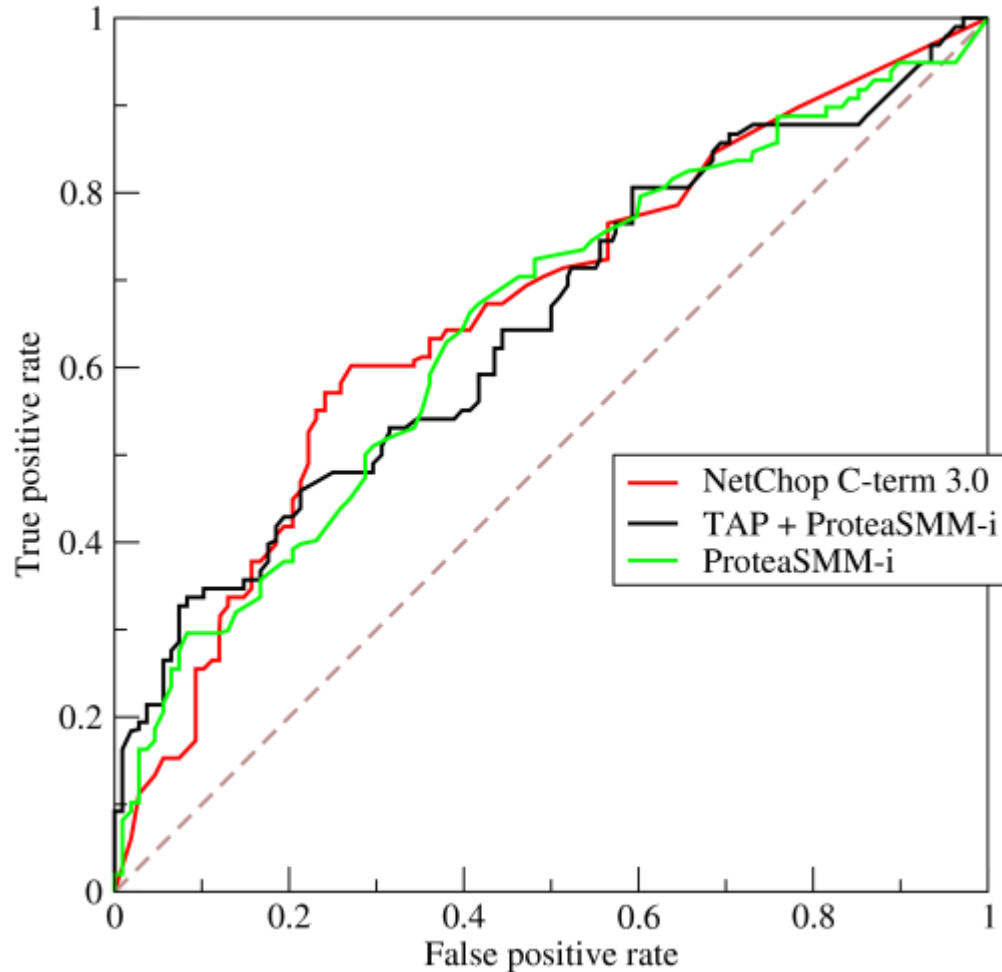
AUC (cont.)



TPR: true positive rate
FPR: false positive rate

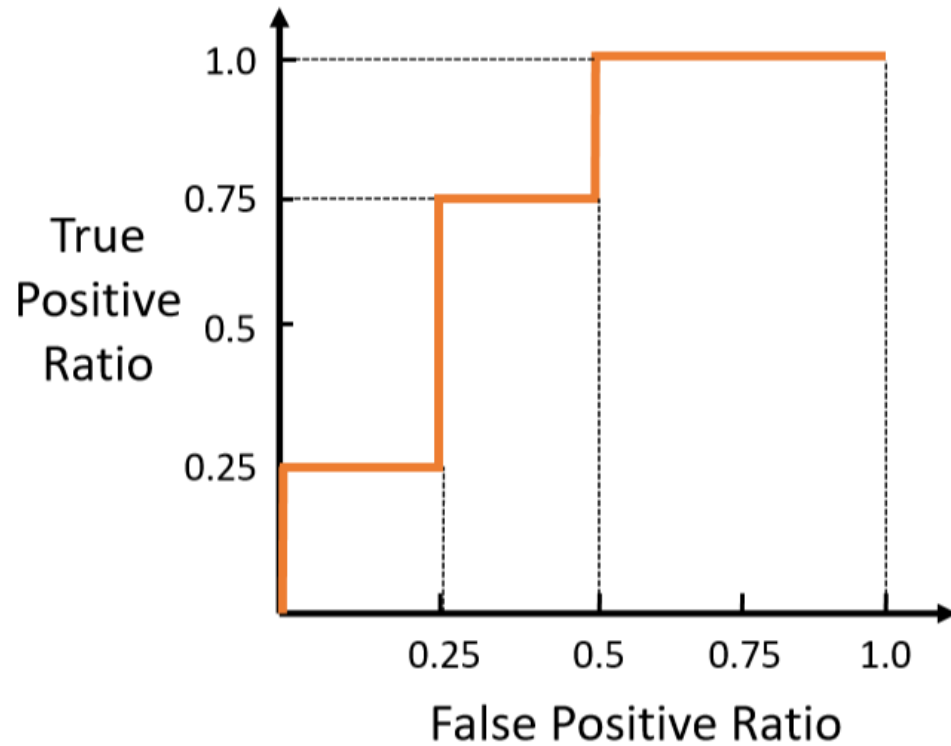
- It's the relationship between TPR and FPR when the threshold is changed from 0 to 1
- In the top right corner, threshold is 0, and every thing is predicted to be positive, so both TPR and FPR is 1
- In the bottom left corner, threshold is 1, and every thing is predicted to be negative, so both TPR and FPR is 0
- The size of the area under this curve (AUC) is an important metric to binary classifier
- Perfect classifier get $AUC=1$ and random classifier get $AUC = 0.5$

AUC (cont.)



- It considers all possible thresholds.
- Various thresholds result in different true/false positive rates.
- As you decrease the threshold, you get more true positives, but also more false positives.
- From a random classifier you can expect as many true positives as false positives. That's the dashed line on the plot. AUC score for the case is 0.5. A score for a perfect classifier would be 1. Most often you get something in between.

AUC example

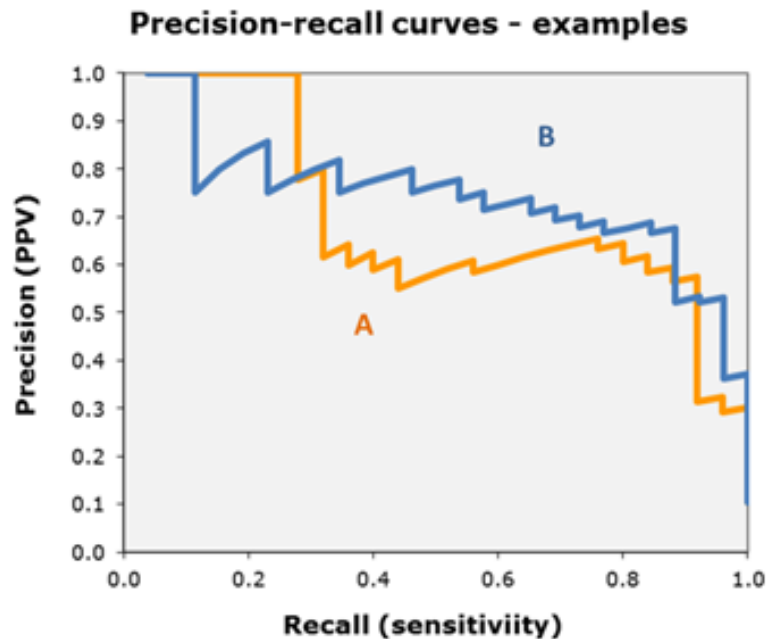


AUC = 0.75

Prediction	Label
0.91	1
0.85	0
0.77	1
0.72	1
0.61	0
0.48	1
0.42	0
0.33	0

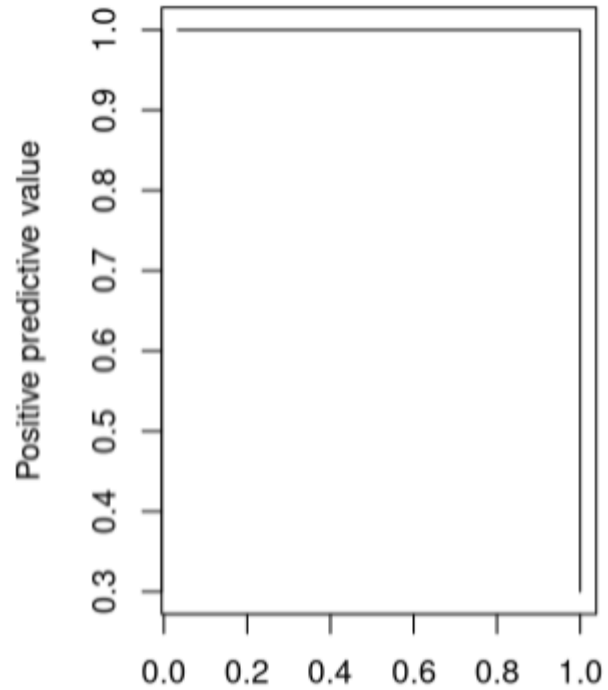
Precision recall curve

- The precision recall curve, or pr curve, is another plot to measure the performance of binary classifier.



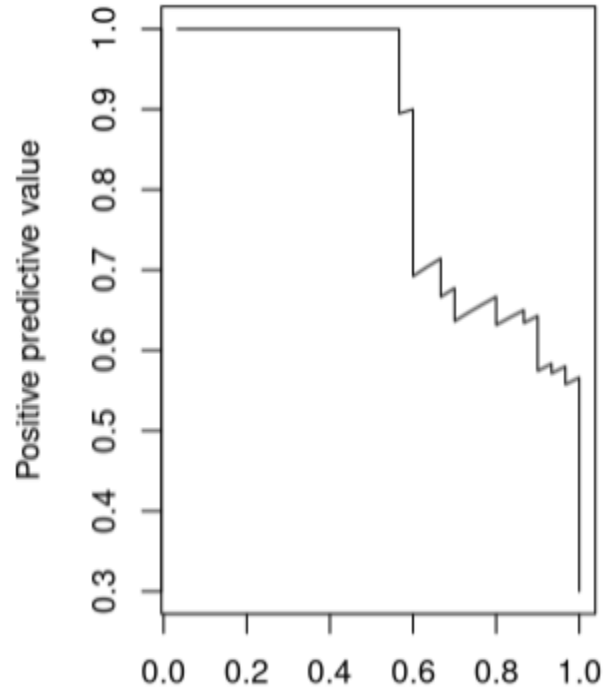
- It's the relationship between Precision and Recall when the threshold is changed from 0 to 1
- It's more complex than the ROC curve
- The size of the area under this curve is an important metric to binary classifier
- It can handle **imbalanced** dataset
- Usually, the classifiers gets lower **AUPR** value than AUC value

AUPR examples



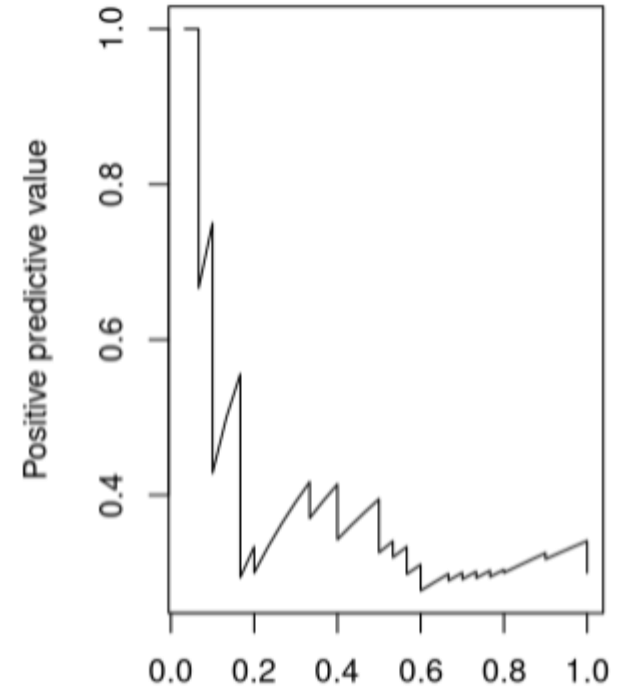
True positive rate

Perfect: 1



True positive rate

Good: 0.92



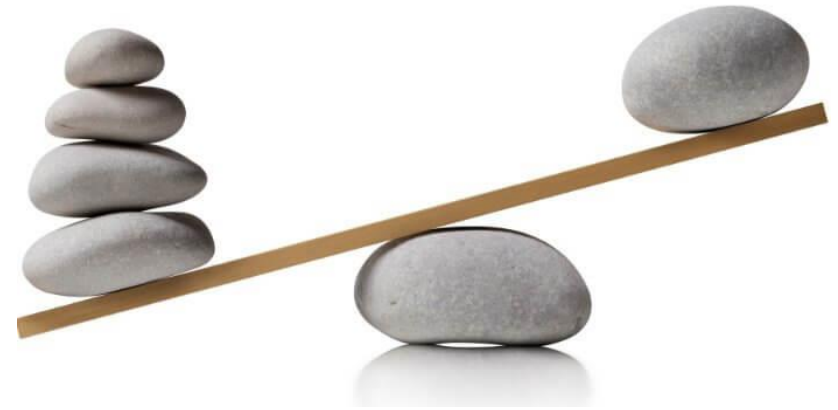
True positive rate

Random: 0.56

Class Imbalance

Class imbalance

- Down sampling
 - Sample less on frequent class
- Up sampling
 - Sample more on infrequent class
- Hybrid Sampling
 - Combine them two



Weighted loss functions

$$L(y, x, p_{\theta}) = -y \log p_{\theta}(1|x) - (1 - y) \log (1 - p_{\theta}(1|x))$$

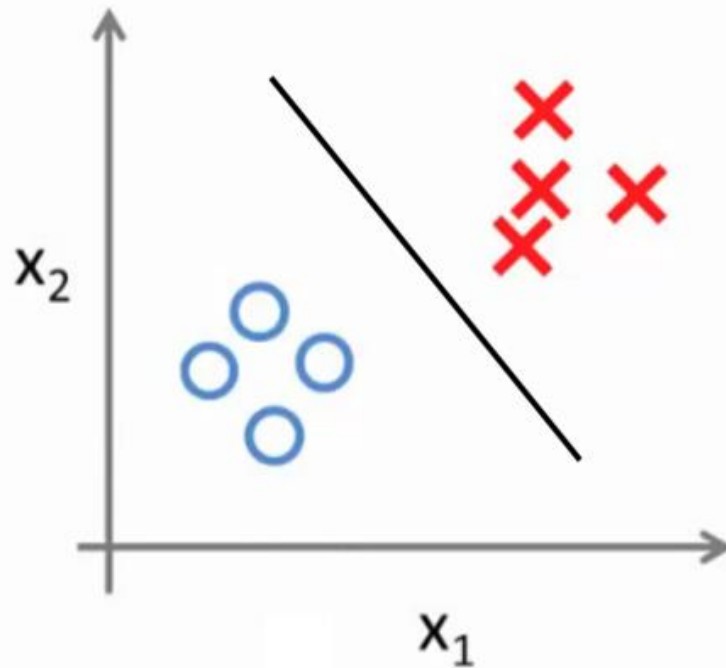
$$L(y, x, p_{\theta}) = -w_1 y \log p_{\theta}(1|x) - w_0 (1 - y) \log (1 - p_{\theta}(1|x))$$

Multi-Class Logistic Regression

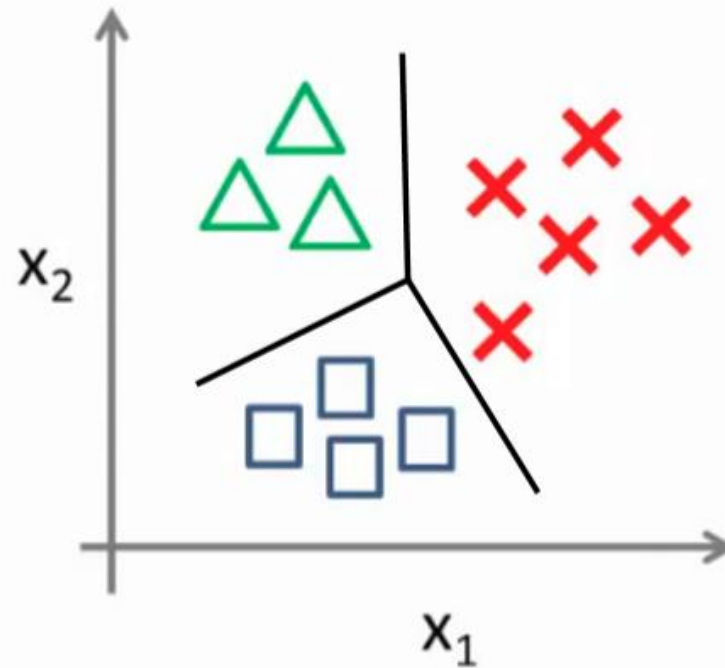
Multi-class classification

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

Binary classification:



Multi-class classification:



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

- 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

- Discriminative / Generative Models
- Logistic regression (binary classification)
 - Cross entropy
 - Formulation, sigmoid function
 - Training—gradient descent
- More measures for binary classification (AUC, AUPR)
- Class imbalance
- Multi-class logistic regression

Shuai Li

<https://shuaili8.github.io>

Questions?