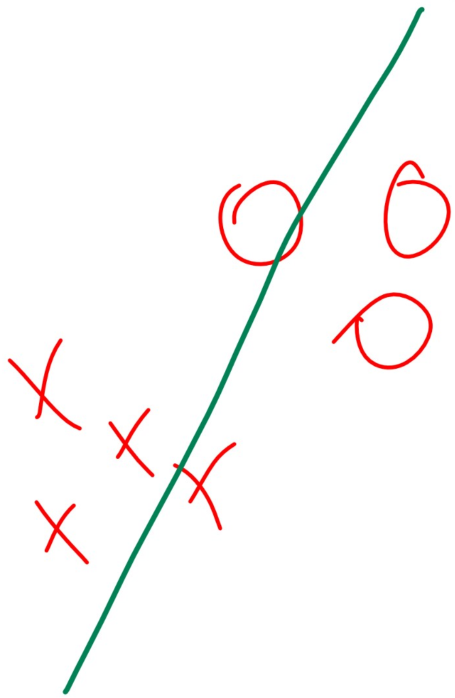
# CS540 Introduction to Artificial Intelligence
# Lecture 2

Young Wu
Based on lecture slides by Jerry Zhu, Yingyu Liang, and Charles Dyer

May 18, 2020

# Loss Function Diagram

## Motivation



random

↓

decreasing # mistake

min cost

# Zero-One Loss Function

## Motivation

$0.5 = y_i \quad [0, 1]$

$a_i = f(x_i) = 2.4, \quad -1.2$

\# of mistake

- An objective function is needed to select the "best" $\hat{f}$. An example is the zero-one loss.

classifier prediction

$$\hat{f} = \arg\min_{f} \sum_{i=1}^{n} \mathbb{1}_{\{f(x_i) \neq y_i\}}$$

line

1

- $\arg\min_{f}$ objective $(f)$ outputs the function that minimizes the objective.
- The objective function is called the cost function (or the loss function), and the objective is to minimize the cost.

# Squared Loss Function

## Motivation

- Zero-one loss counts the number of mistakes made by the classifier. The best classifier is the one that makes the fewest mistakes.
- Another example is the squared distance between the predicted and the actual $y$ value:

$a_i$ for perceptron

$$\hat{f} = \arg\min_f \frac{1}{2} \sum_{i=1}^{n} \left(f\left(x_i\right) - y_i\right)^2 \Longleftarrow \mathbb{1}\{f(x_i) \neq y_i\}$$

| $f(x_i)$ | $y_i$ | sq loss | |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |

# Loss Functions Equivalence

## Quiz

- Which ones (multiple) of the following functions are equivalent to the squared error for binary classification?

Q9

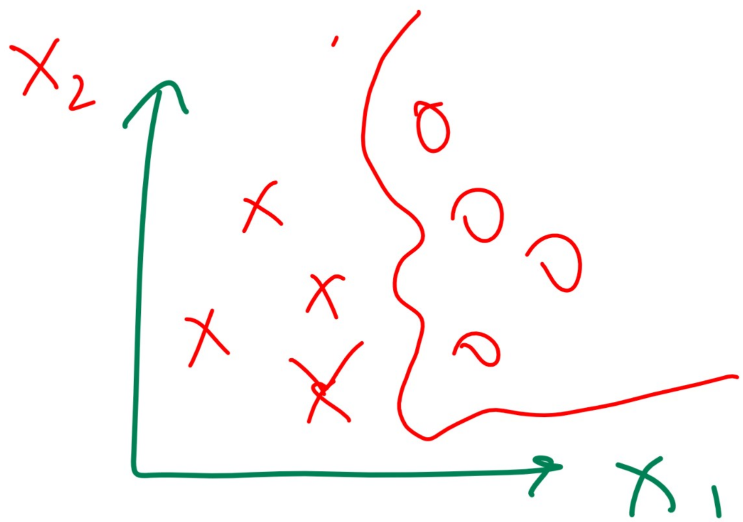
$$C = \sum_{i=1}^{n} \left(f(x_i) - y_i\right)^2, y_i \in \{0, 1\}$$

| | | sq | abs | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 2 |
| 1 | 0 | 1 | 1 | 2 |
| 1 | 1 | 0 | 0 | 0 |

- A: $\sum \mathbb{1}_{\{f(x_i) \neq y_i\}}$
- ~~B~~: $\sum \mathbb{1}_{\{f(x_i) = y_i\}}$
- C: $\sum |f(x_i) - y_i|$
- ~~D~~: $\sum \max\{0, 1 - f(x_i)\, y_i\}$
- E: $\frac{1}{2}\sum \max\{0, 1 - (2 \cdot f(x_i) - 1)(2 \cdot y_i - 1)\}$

# Function Space Diagram

## Motivation



# Hypothesis Space

## ~~Motivation~~

- There are too many functions to choose from.
- There should be a smaller set of functions to choose $\hat{f}$ from.

$$\hat{f} = \arg\min_{f \in \mathcal{H}} \frac{1}{2} \sum_{i=1}^{n} \left(f\left(x_i\right) - y_i\right)^2$$

- The set $\mathcal{H}$ is called the hypothesis space.

# Activation Function

## Motivation

- Suppose $\mathcal{H}$ is the set of functions that are compositions between another function $g$ and linear functions.

$$\left(\hat{w}, \hat{b}\right) = \arg\min_{w,b} \frac{1}{2} \sum_{i=1}^{n} (a_i - y_i)^2$$

where $a_i = g\left(w^T x + b\right)$

$\mathcal{H} \Rightarrow$ all functions that can be written like this

LTU perceptron

$a_i = \mathbb{1}(w^T x + b \geq 0)$

- $g$ is called the activation function.

# Linear Threshold Unit

## Motivation

- One simple choice is to use the step function as the activation function:

$$g\left(\boxed{\cdot}\right) = \mathbb{1}_{\{\boxed{\cdot} \geqslant 0\}} = \begin{cases} 1 & \text{if } \boxed{\cdot} \geqslant 0 \\ 0 & \text{if } \boxed{\cdot} < 0 \end{cases}$$

- This activation function is called linear threshold unit (LTU).

# Sigmoid Activation Function

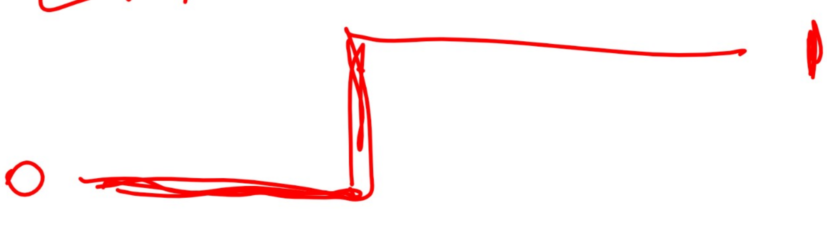
## Motivation

- When the activation function $g$ is the sigmoid function, the problem is called logistic regression.

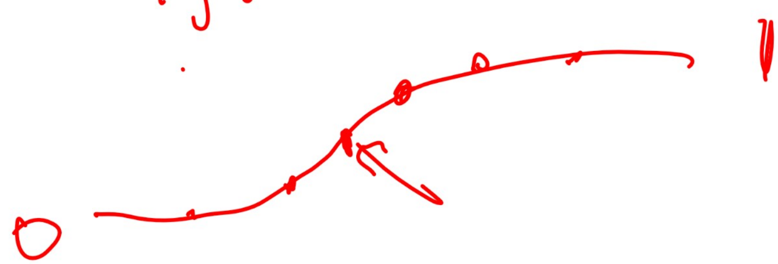
$$g\left(\boxed{\cdot}\right) = \frac{1}{1 + \exp\left(-\boxed{\cdot}\right)}$$

- This $g$ is also called the logistic function.

LTU




logistic



# Sigmoid Function Diagram

## Motivation

# Cross Entropy Loss Function

## Motivation

- The cost function used for logistic regression is usually the log cost function.

$$C(f) = -\sum_{i=1}^{n} \left(y_i \log\left(f\left(x_i\right)\right) + \left(1 - y_i\right) \log\left(1 - f\left(x_i\right)\right)\right)$$

- It is also called the cross-entropy loss function.

like counting
# mistake

# Logistic Regression Objective

## Motivation

- The logistic regression problem can be summarized as the following.

$$\left(\hat{w}, \hat{b}\right) = \arg\min_{w,b} - \sum_{i=1}^{n} \left(y_i \log\left(a_i\right) + \left(1 - y_i\right) \log\left(1 - a_i\right)\right)$$

where $a_i = \dfrac{1}{1 + \exp\left(-z_i\right)}$ and $z_i = w^T x_i + b$

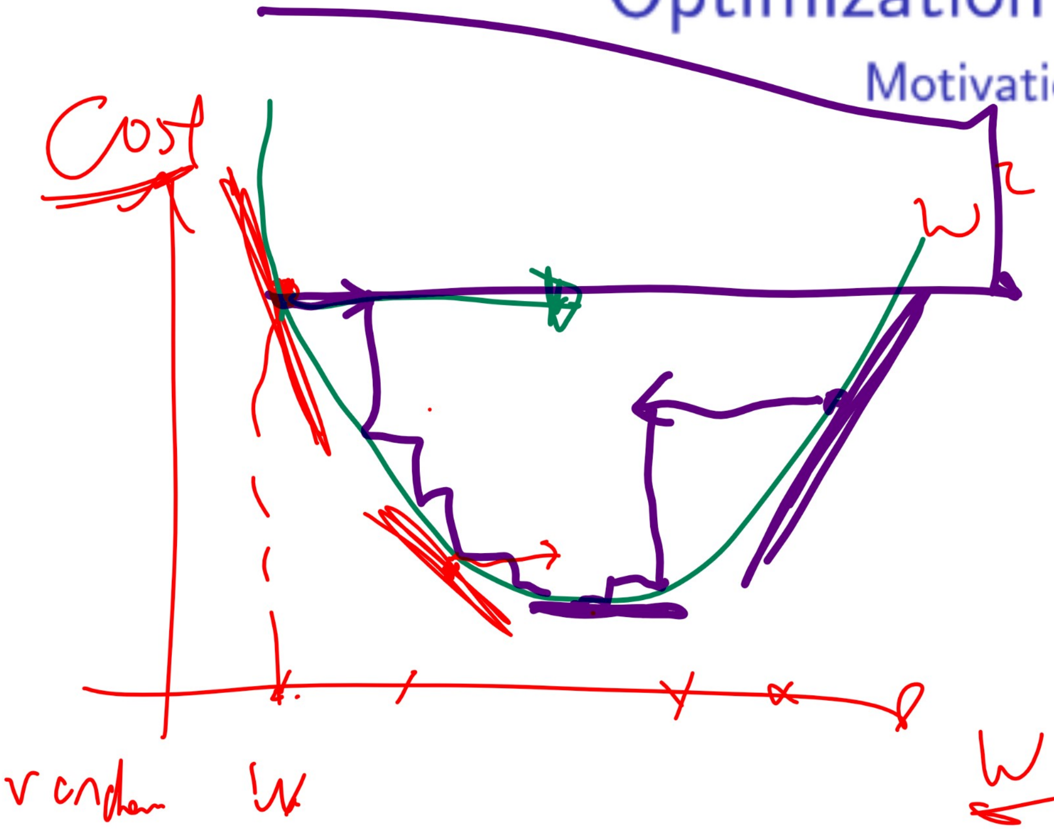
cost

logistic.

linear part

# Optimization Diagram

## Motivation




$$\frac{\partial C}{\partial w} \le 0 \Rightarrow w \uparrow$$

$$\frac{\partial C}{\partial w} > 0 \Rightarrow w \downarrow$$

$$\underline{GD} \rightarrow w - \alpha \underbrace{\frac{\partial C}{\partial w}}_{\text{gradient}}$$

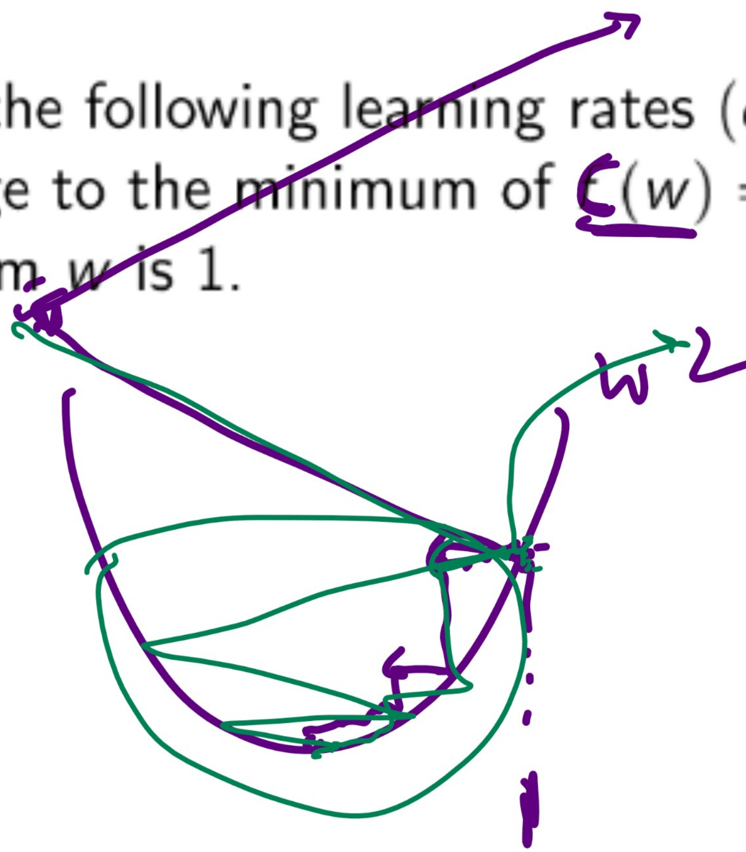
# Learning Rate Demo

## Motivation

# Simple Gradient Descent

## Quiz

- For which (multiple) of the following learning rates ($\alpha$) would gradient descent converge to the minimum of $C(w) = w^2$. Suppose the intial random $w$ is 1.
- A: 0.1
- B: 0.2
- C: 0.5
- D: 1
- E: 2




$1 - 4 = -3$

$\text{new } w = w - \alpha \cdot 2w$

$= |1 - \alpha \cdot 2 \cdot 1| < 1 = 0.8$

# Simple Gradient Descent, Answer

## Quiz

# Simple Gradient Descent, Another One

## Quiz

select largest

- For which (~~multiple~~) of the following learning rates ($\alpha$) would gradient descent converge to the minimum of $C(w) = \frac{1}{2}w^2$. Suppose the initial random $w$ is 1.
- A: 0.1
- B: 0.2
- C: 0.5
- D: 1
- E: 2

Q10

$$w = w - \alpha \frac{\partial c}{\partial w}$$

$$\left| \underbrace{1 - \alpha \cdot 1}_{2} \right| \leq 1$$

# Simple Gradient Descent, Another One, Answer

## Quiz

# Logistic Regression

## Description

use current weight to predict,

- Initialize random weights.
- Evaluate the activation function.
- Compute the gradient of the cost function with respect to each weight and bias.
- Update the weights and biases using gradient descent.
- Repeat until convergent.

# Logistic Gradient Derivation 1

Definition

$$C = -\sum_{i=1}^{n} y_i \log(a_i) + (1-y_i)\log(1-a_i)$$

$$a_i = \frac{1}{1+e^{-(w^T x_i + b)}}$$

Chain Rule

$(x_{i1}, x_{i2}, \ldots, x_{im})$

$$\frac{\partial C}{\partial w_j} = \sum_{i=1}^{n} \frac{\partial C}{\partial a_i}\frac{\partial a_i}{\partial w_j}$$

$$= \cancel{+}\sum_{i=1}^{n}\left(\boxed{\frac{y_i}{a_i} - \frac{1-y_i}{1-a_i}}\right)\left(\frac{\cancel{+}\, e^{-(w^T x_i + b)}}{\left(1+e^{-(w^T x_i+b)}\right)^2} \cdot x_{ij}\right)$$

$$= \sum_{i=1}^{n}\{y_i(1-a_i) - (1-y_i)a_i\}x_{ij} \qquad \underbrace{\frac{1}{1+e^{-(w^T x_i+b)}}}_{a_i} \qquad \underbrace{\frac{e^{-(w^T x_i+b)}}{1+e^{-(w^T x_i+b)}}}_{1-a_i} x_{ij}$$

$y_i - a_i$

# Logistic Gradient Derivation 2

## Definition

$$= \sum_{i=1}^{n} (a_i - y_i) x_{ij}$$

cross entropy loss

$$\boxed{w = w - \alpha \sum_{i=1}^{n} (a_i - y_i) x_i}$$

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \end{bmatrix} - \qquad \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}$$

# Gradient Descent Step

## Definition

- For logistic regression, use chain rule twice.

$$w = w - \alpha \sum_{i=1}^{n} (a_i - y_i) x_i$$

$$b = b - \alpha \sum_{i=1}^{n} (a_i - y_i)$$

$$a_i = g\left(w^T x_i + b\right), g\left(\boxed{\cdot}\right) = \frac{1}{1 + \exp\left(-\boxed{\cdot}\right)}$$

- $\alpha$ is the learning rate. It is the step size for each step of gradient descent.

# Perceptron Algorithm

## Definition

- Update weights using the following rule.

$$
\begin{aligned}
w &= w - \alpha \left(a_i - y_i\right) x_i \\
b &= b - \alpha \left(a_i - y_i\right) \\
a_i &= \mathbb{1}_{\{w^T x_i + b \geqslant 0\}}
\end{aligned}
$$

# Gradient Descent

## Quiz

- What is the gradient descent step for $w$ if the objective (cost) function is the squared error?

Q 11

$a_i = \frac{1}{1+e^{-w^Tx+b}}$

$$C = \boxed{\frac{1}{2}\sum_{i=1}^{n}(a_i - y_i)^2}, a_i = g\left(w^T x_i + b\right), g'(z) = z \cdot (1 - z)$$

$\boxed{a_i(1-a_i)x_i}$

$\boxed{\frac{1}{2}\cdot 2(a_i - y_i)}$

- A: $w = w - \alpha \sum (a_i - y_i)$
- B: $w = w - \alpha \sum (a_i - y_i)\, x_i$ ← cross entropy
- C: $w = w - \alpha \sum (a_i - y_i)\, a_i x_i$
- D: $w = w - \alpha \sum (a_i - y_i)\, (1 - a_i)\, x_i$
- E: $w = w - \alpha \sum \boxed{(a_i - y_i)}\, \boxed{a_i (1 - a_i)\, x_i}$

$$\boxed{\frac{\partial C}{\partial a_i}}\ \boxed{\frac{\partial a_i}{\partial w}}$$

# Gradient Descent, Answer

## Quiz

# Gradient Descent, Another One

## Quiz

- What is the gradient descent step for $w$ if the activation funtion is the identity function?

Q12

linear regression

$w_1 x_1 + w_2 x_2$

$$C = \frac{1}{2}\sum_{i=1}^{n}(a_i - y_i)^2, \; a_i = w^T x_i + b$$

$$\frac{\partial C}{\partial w} = \sum \frac{\partial C}{\partial a_i} \cdot \frac{\partial a_i}{\partial w}$$

$$(a_i - y_i)\, x_i$$

- A: $w = w - \alpha \sum (a_i - y_i)$
- $\boxed{\text{B: } w = w - \alpha \sum (a_i - y_i)\, x_i}$
- C: $w = w - \alpha \sum (a_i - y_i)\, a_i x_i$
- D: $w = w - \alpha \sum (a_i - y_i)\,(1 - a_i)\, x_i$
- E: $w = w - \alpha \sum (a_i - y_i)\, a_i\,(1 - a_i)\, x_i$

# Gradient Descent, Another One, Answer

## Quiz

# Other Non-linear Activation Function

## Discussion

- Activation function: $g\left(\boxed{\cdot}\right) = \tanh\left(\boxed{\cdot}\right) = \dfrac{e^{\boxed{\cdot}} - e^{-\boxed{\cdot}}}{e^{\boxed{\cdot}} + e^{-\boxed{\cdot}}}$
- Activation function: $g\left(\boxed{\cdot}\right) = \arctan\left(\boxed{\cdot}\right)$
- Activation function (rectified linear unit): $g\left(\boxed{\cdot}\right) = \boxed{\cdot}\mathbb{1}_{\left\{\boxed{\cdot}\geqslant 0\right\}}$
- All these functions lead to objective functions that are convex and differentiable (almost everywhere). Gradient descent can be used.

# Convexity Diagram

## Discussion