25: Logistic Regression

Lisa Yan
June 3, 2020

Quick slide reference

Background

1. Weighted sum

If $\boldsymbol{X} = (X_1, X_2, \dots, X_m)$:

$$Z = \theta_1 X_1 + \theta_2 X_2 + \cdots + \theta_m X_m$$

$$= \sum_{j=1}^{m} \theta_j X_j$$

weighted sum

$$= \theta^T \boldsymbol{X}$$

dot product

$$\begin{bmatrix} \theta_1 & \theta_2 & & \theta_m \end{bmatrix} \begin{bmatrix} X_1 \\ \\ X_m \end{bmatrix}$$

1. Weighted sum

Dot product/ weighted sum $\theta^T \boldsymbol{X} = \sum_{j=1}^{m} \theta_j X_j$

Recall the linear regression model, where $\boldsymbol{X} = (X_1, X_2, \dots, X_m)$ and $Y \in \mathbb{R}$:

$$\hat{Y} = g(\boldsymbol{X}) = \theta_0 + \sum_{j=1}^{m} \theta_j X_j$$

How would you rewrite this expression as a single dot product?

1. Weighted sum

Dot product/ weighted sum $\theta^T \boldsymbol{X} = \sum_{j=1}^{m} \theta_j X_j$

Recall the linear regression model, where $\boldsymbol{X} = (X_1, X_2, \dots, X_m)$ and $Y \in \mathbb{R}$:

$$g(\boldsymbol{X}) = \theta_0 + \sum_{j=1}^{m} \theta_j X_j$$

How would you rewrite this expression as a single dot product?

$$g(\boldsymbol{X}) = \theta_0 X_0 + \theta_1 X_1 + \theta_2 X_2 + \cdots + \theta_m X_m$$

Define $X_0 = 1$

$$= \theta^T \boldsymbol{X}$$

New $\boldsymbol{X} = (1, X_1, X_2, \dots, X_m)$, $\theta = (\theta_0, \theta_1, \dots, \theta_m)$

Prepending $X_0 = 1$ to each feature vector $\boldsymbol{X}$ makes matrix operators more accessible.

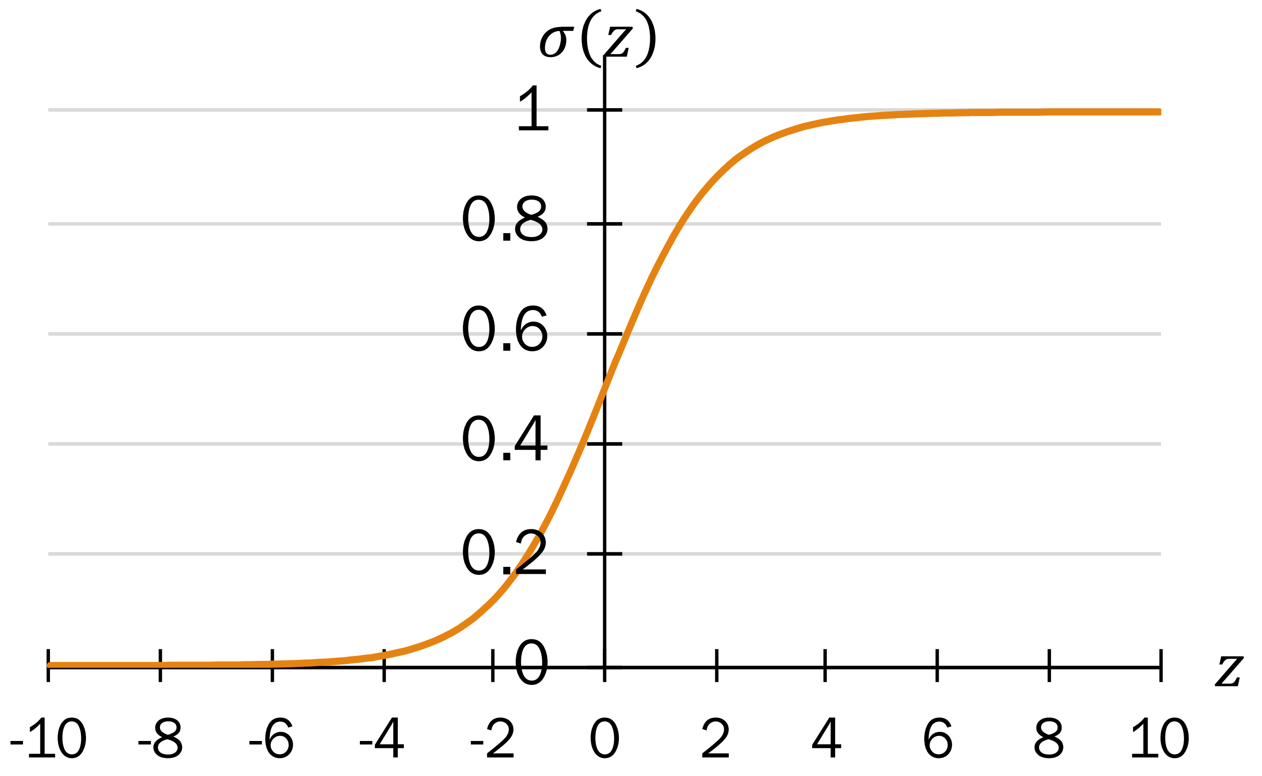
2. Sigmoid function $\sigma(z)$

- The sigmoid function:

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

- Sigmoid squashes z to a number between 0 and 1.

- Recall definition of probability: A number between 0 and 1



$\sigma(z)$ can represent a probability.

3. Conditional likelihood function

Training data (n datapoints):

- $\left(\boldsymbol{x}^{(i)}, y^{(i)}\right)$ drawn i.i.d. from a distribution $f\left(\boldsymbol{X} = \boldsymbol{x}^{(i)}, Y = y^{(i)} | \theta\right) = f\left(\boldsymbol{x}^{(i)}, y^{(i)} | \theta\right)$

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} | \, \boldsymbol{x}^{(i)}, \theta\right)$$

conditional likelihood of training data

$$= \arg\max_{\theta} \sum_{i=1}^{n} \log f\left(y^{(i)} | \, \boldsymbol{x}^{(i)}, \theta\right)$$

log conditional likelihood

$$= \arg\max_{\theta} LL(\theta)$$

- MLE in this lecture is estimator that maximizes conditional likelihood
- Confusingly, log conditional likelihood is also written as $LL(\theta)$

Logistic Regression

Prediction models so far

Review

Linear Regression (Regression)

$\boldsymbol{X} \Rightarrow \boxed{\theta_0 + \sum_{j=1}^{m} \theta_j X_j} \Rightarrow \hat{Y}$

$\hat{Y} = \theta_0 + \sum_{j=1}^{m} \theta_j X_j$

- ✅ $\boldsymbol{X}$ can be dependent
- 🙆 Regression model ($\hat{Y} \in \mathbb{R}$, not discrete)

Naïve Bayes (Classification)

$\boldsymbol{X}, Y \Rightarrow \boxed{\hat{P}(\boldsymbol{X}|Y)\hat{P}(Y)} \Rightarrow \hat{P}(\boldsymbol{X}, Y)$

$$\hat{Y} = \arg\max_{y=\{0,1\}} P(Y \mid \boldsymbol{X}) = \arg\max_{y=\{0,1\}} P(\boldsymbol{X}|Y)P(Y)$$

- ✅ Tractable with NB assumption, but...
- ⚠ Realistically, X_j features not necessarily conditionally independent
- 🙆 Actually models $P(\boldsymbol{X}, Y)$, not $P(Y|\boldsymbol{X})$?

Introducing Logistic Regression!

Linear Regression ideas

Classification models

+ compute power

Logistic Regression

$$\boldsymbol{X} \Rightarrow \boxed{\theta_0 + \sum_{j=1}^{m} \theta_j X_j} \xrightarrow{z} \boxed{\text{sigmoid function } \sigma(z) = \frac{1}{1+e^{-z}}} \Rightarrow P(Y = 1|\boldsymbol{X})$$

Logistic Regression Model:

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

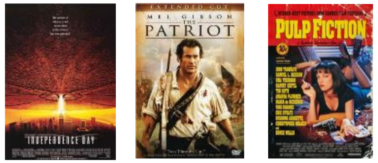
Predict $\hat{Y}$ as the most likely Y given our observation $\boldsymbol{X} = \boldsymbol{x}$:

$$\hat{Y} = \arg\max_{y=\{0,1\}} P(Y \mid \boldsymbol{X})$$

- Since $Y \in \{0,1\}$, $P(Y = 0|\boldsymbol{X} = \boldsymbol{x}) = 1 - \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$
- Sigmoid function also known as "logit" function

Logistic Regression

$\boldsymbol{x} = [0,1,1]$



θ parameter

0.81

$\boldsymbol{X}$

input features

$P(Y = 1 | \boldsymbol{X} = \boldsymbol{x})$

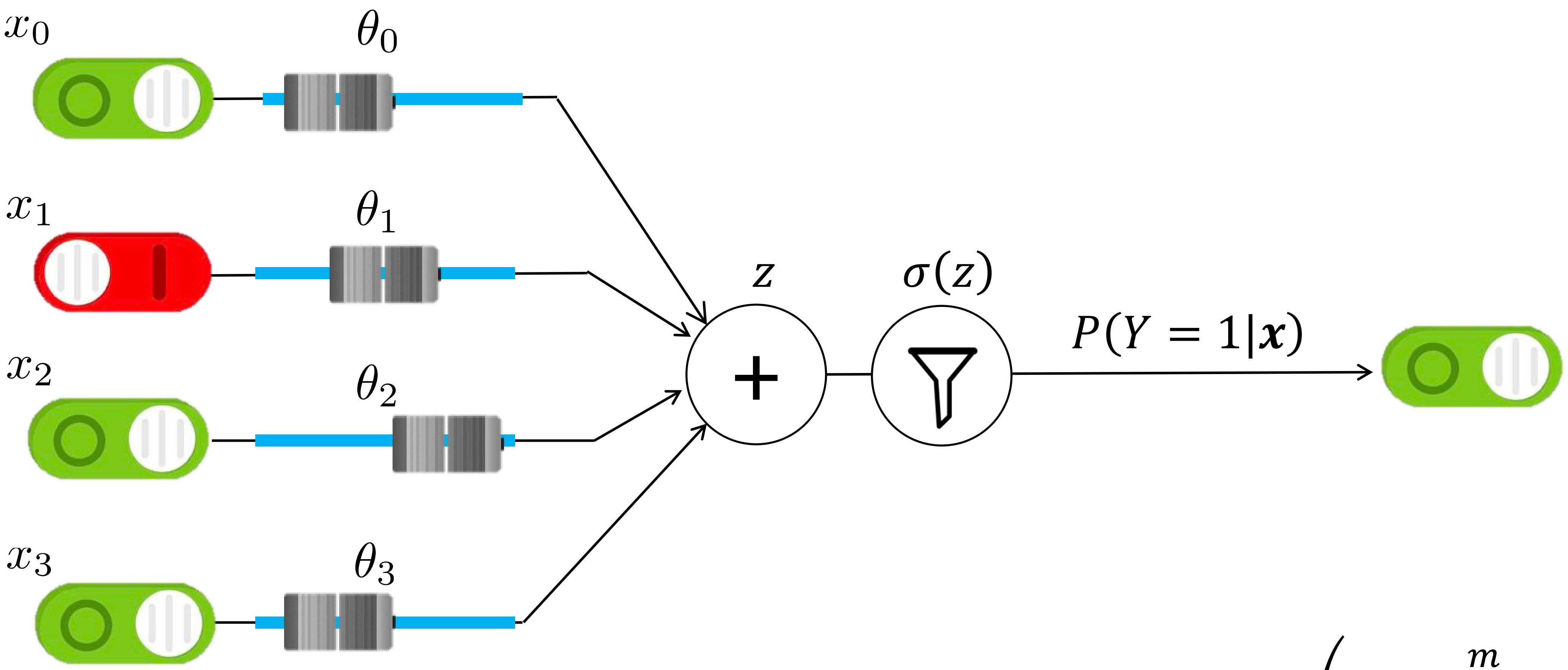
conditional likelihood

$$P(Y = 1 | \boldsymbol{X} = x) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

Logistic Regression cartoon

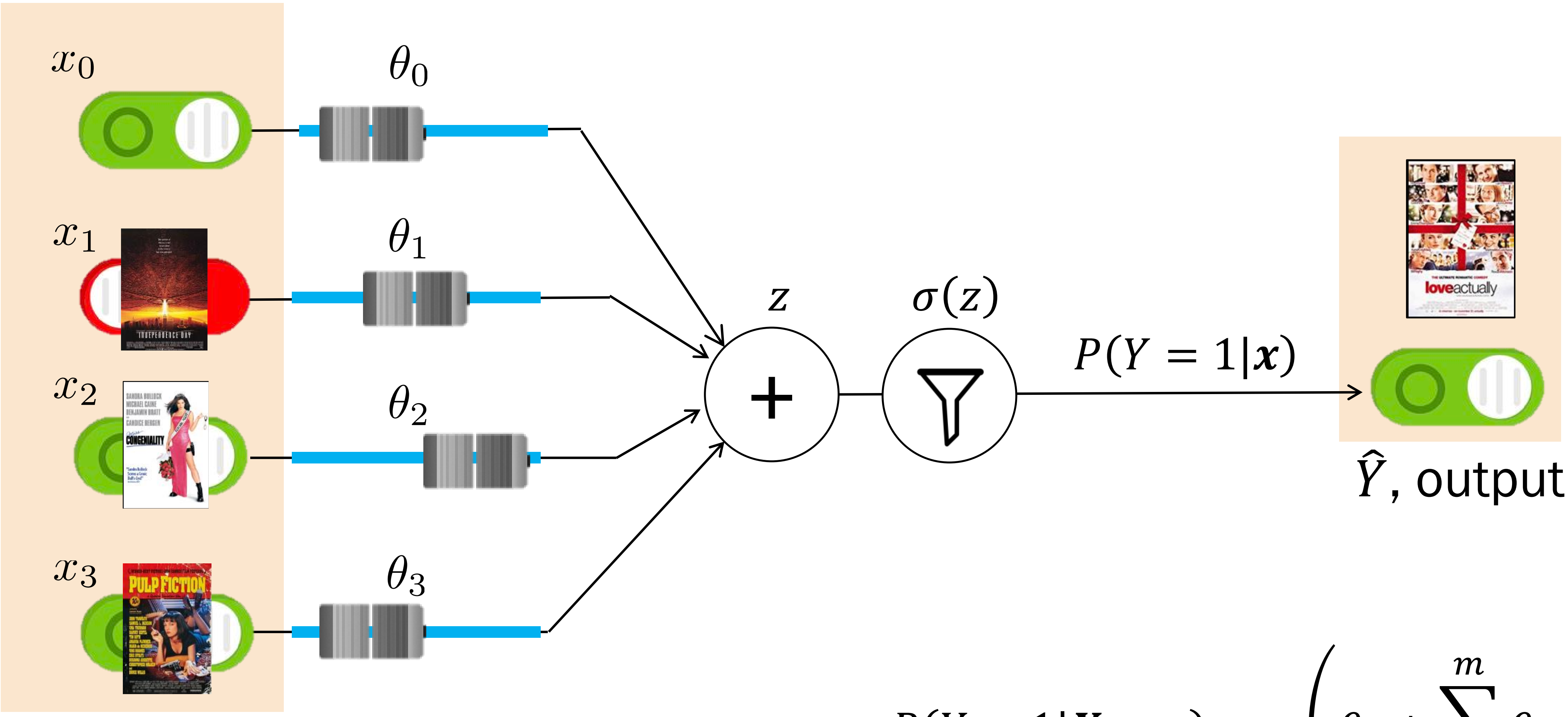
θ parameter

Logistic Regression cartoon



$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

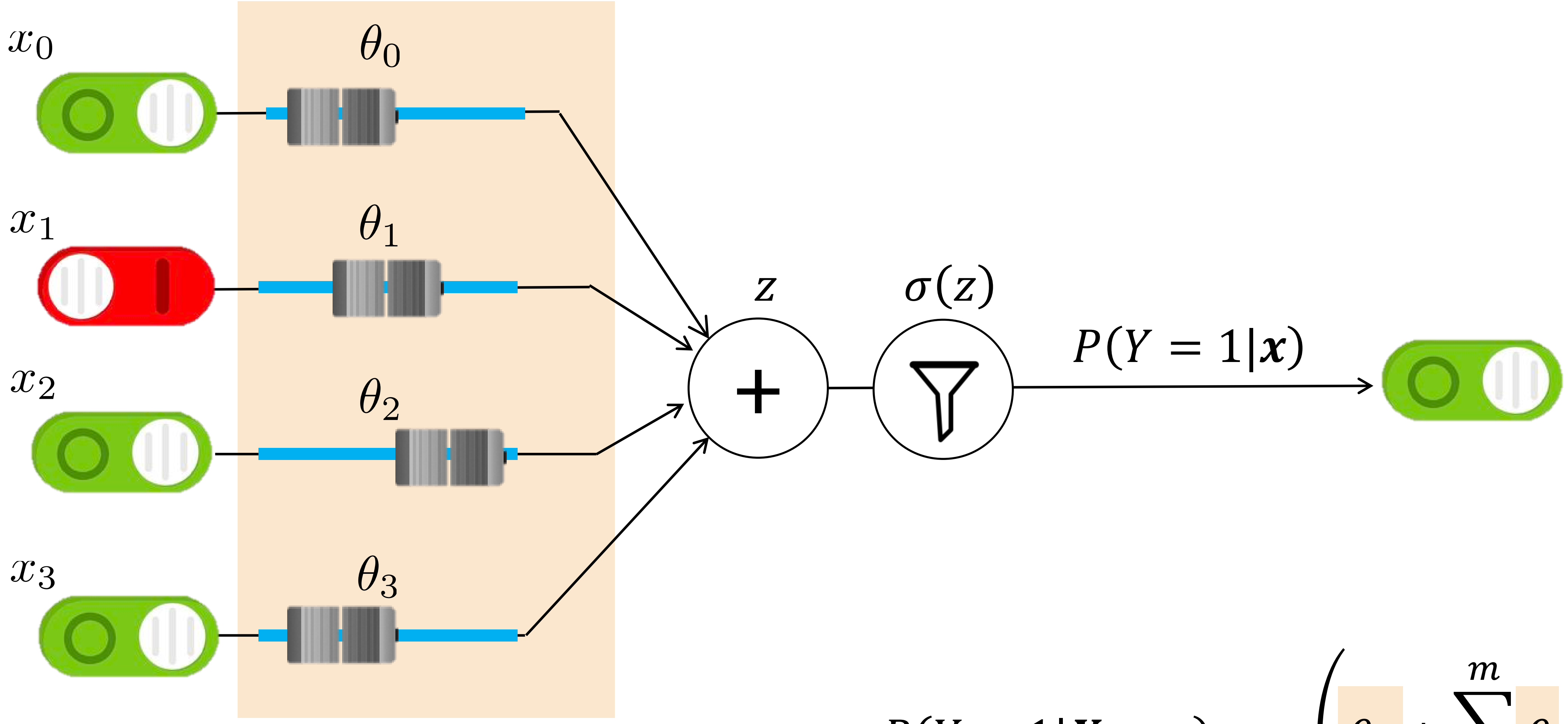
Logistic Regression cartoon



$\boldsymbol{X}$, input features
[0,1,1]

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

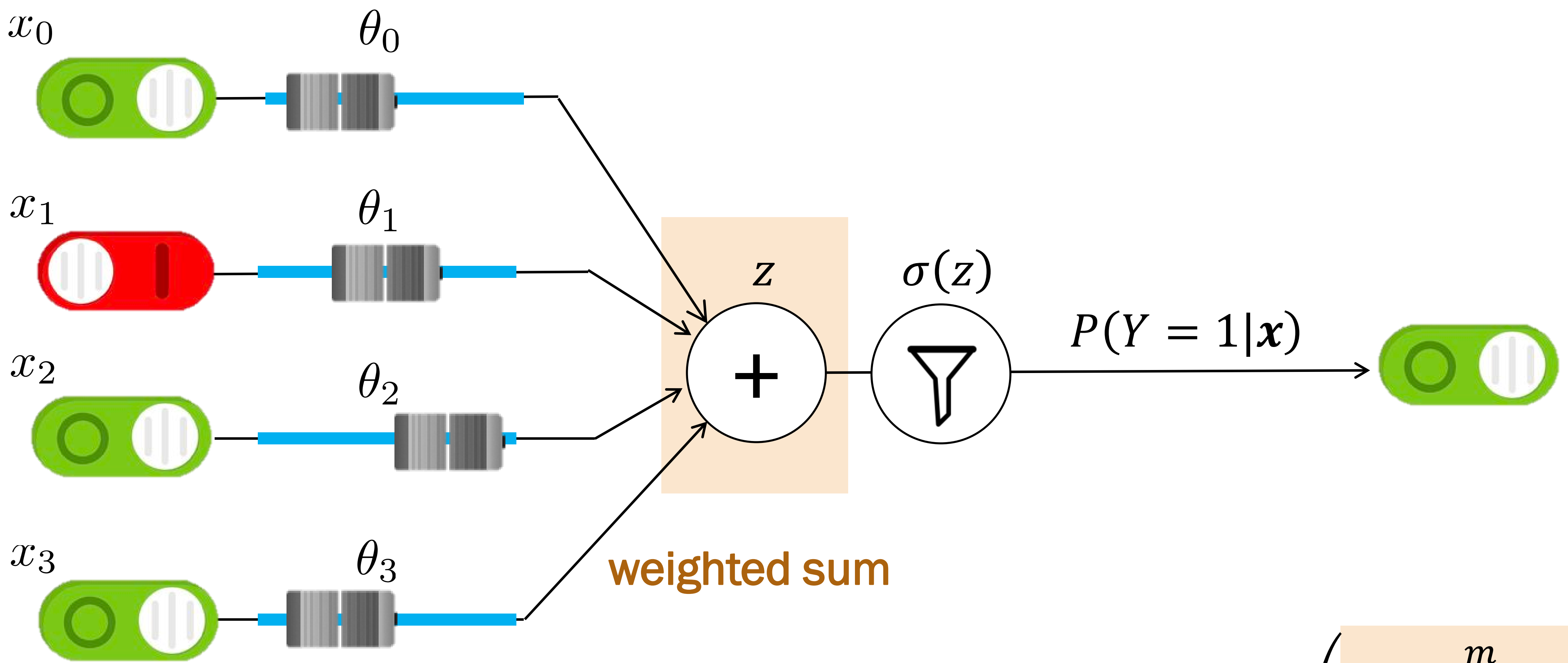
Components of Logistic Regression



θ **weights**
(aka parameters)

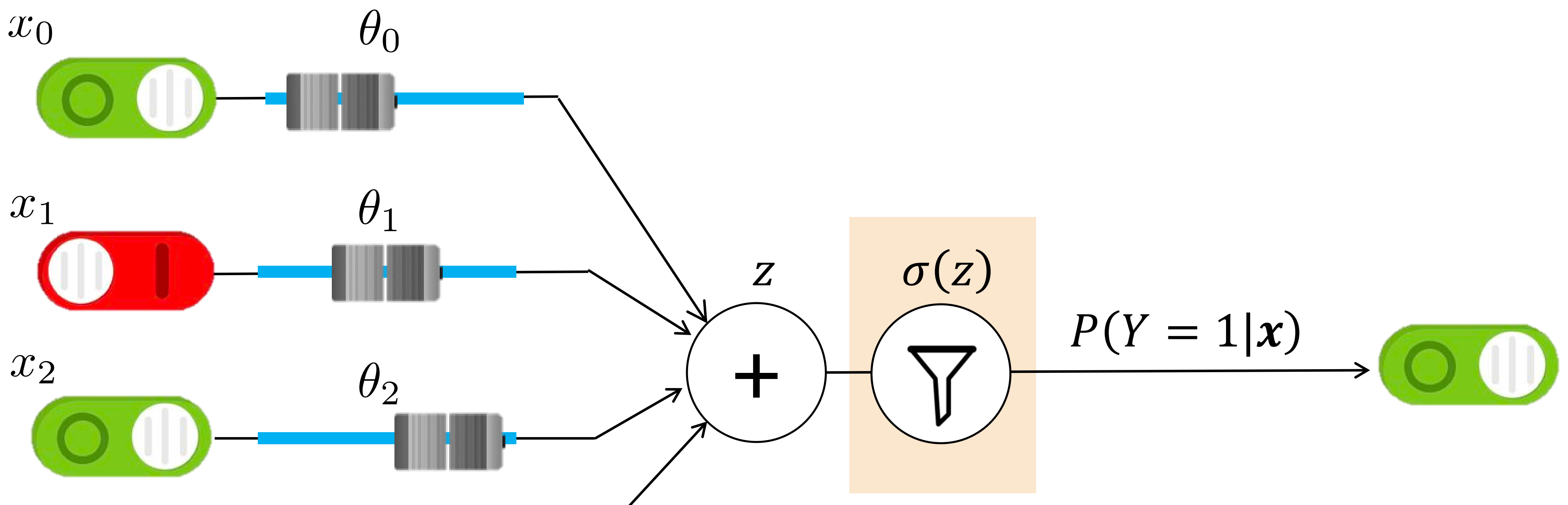
$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

Components of Logistic Regression



$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

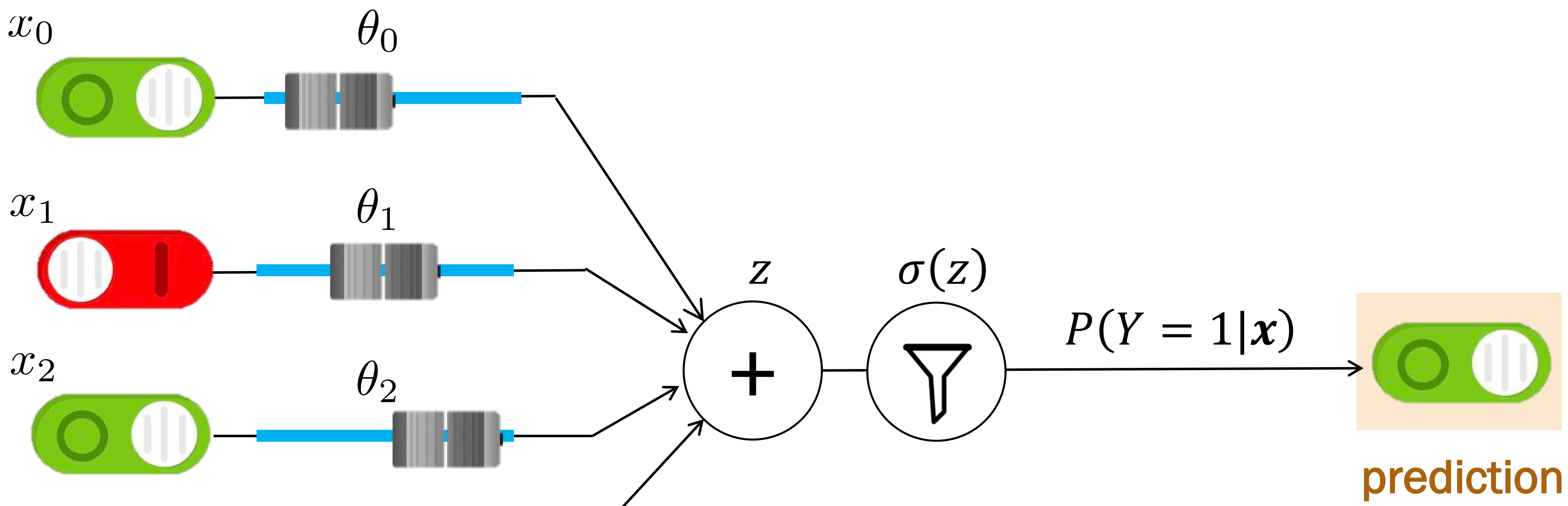
Components of Logistic Regression



squashing function b/t 0 and 1

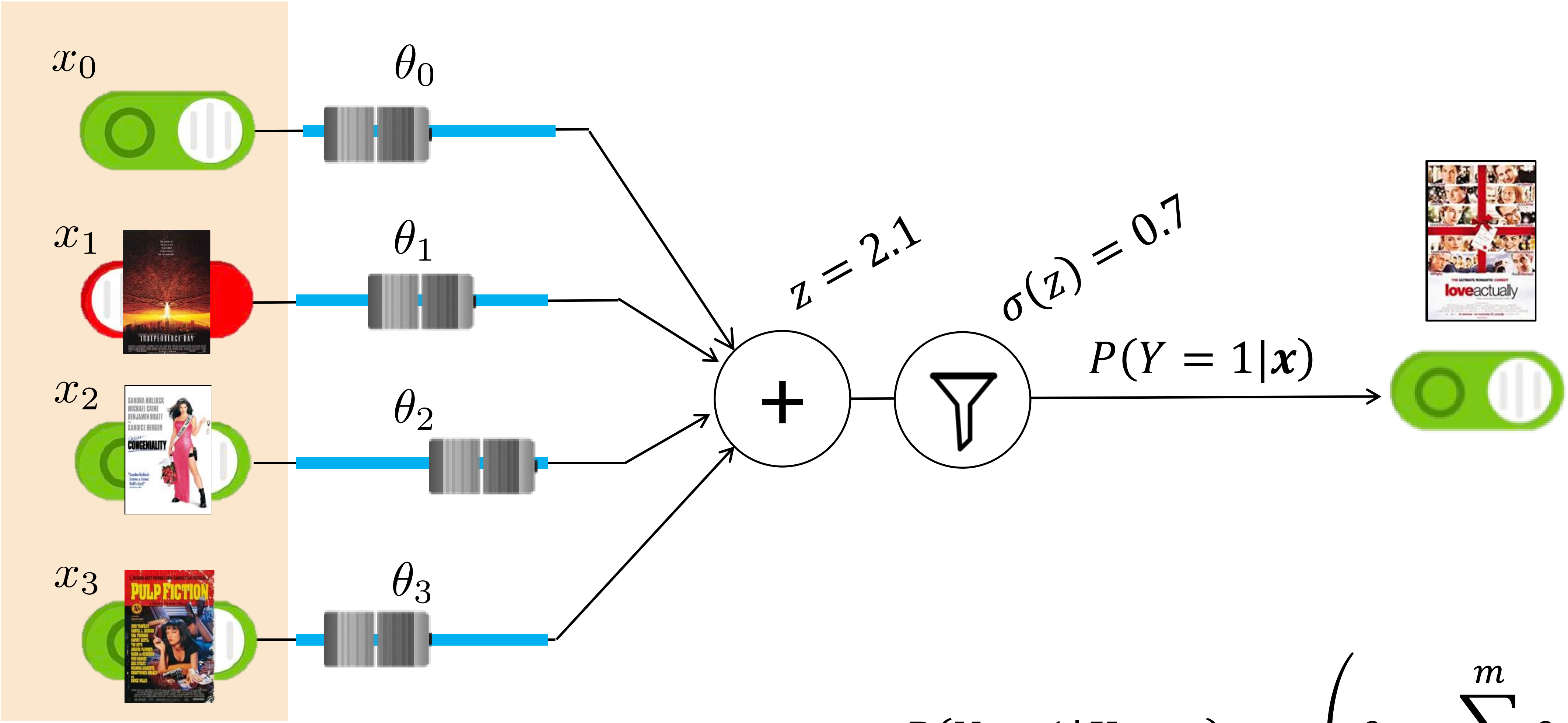
$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

Components of Logistic Regression



$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

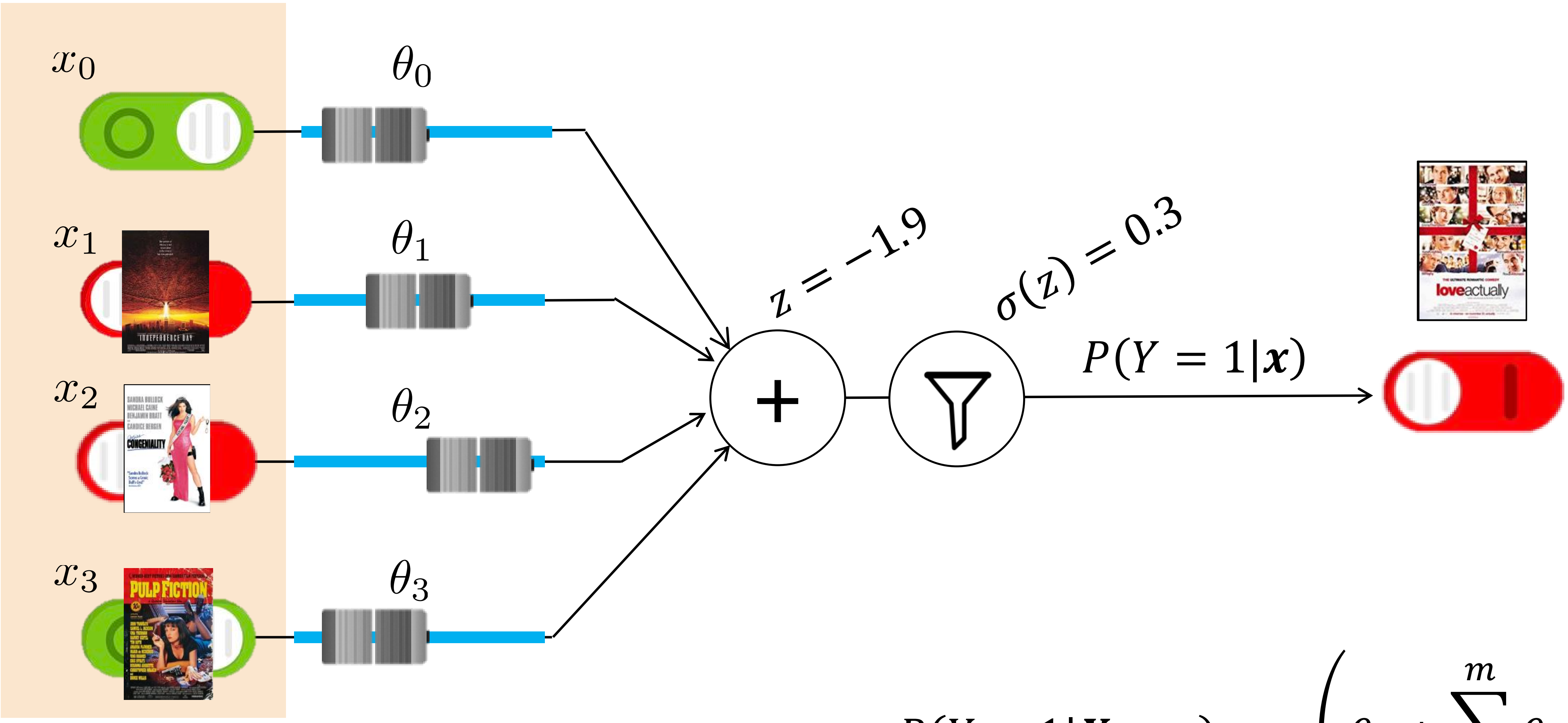
Different predictions for different inputs



$\boldsymbol{X}$, input features
[0,1,1]

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

Different predictions for different inputs

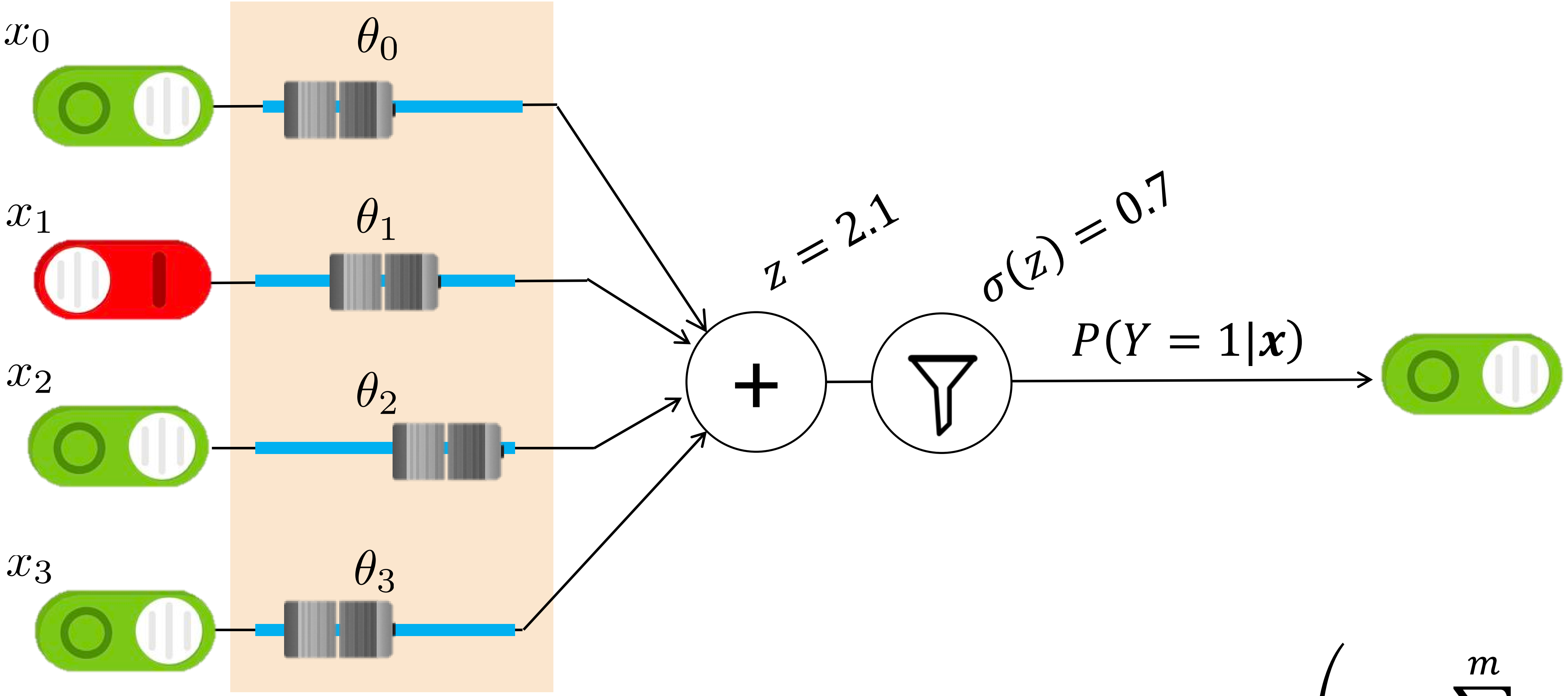


$\boldsymbol{X}$, input features

[0,0,1]

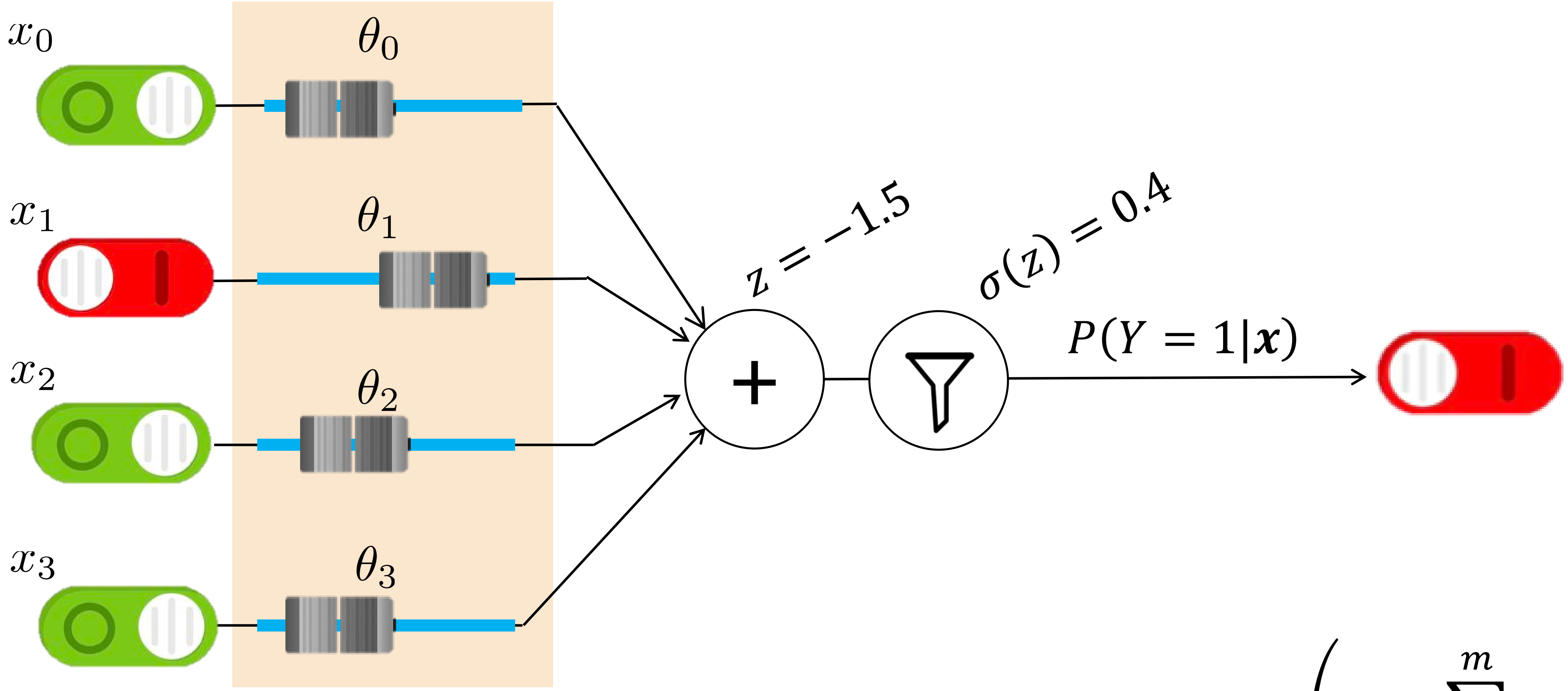
$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

Parameters affect prediction



$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

Parameters affect prediction



$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

For simplicity

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\sum_{j=0}^{m} \theta_j x_j\right) = \sigma(\theta^T \boldsymbol{x}) \quad \text{where } x_0 = 1$$

Logistic regression classifier

$$\hat{Y} = \arg\max_{y=\{0,1\}} P(Y|\boldsymbol{X})$$

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\sum_{j=0}^{m} \theta_j x_j\right) = \sigma(\theta^T \boldsymbol{x})$$

Training

Estimate parameters from training data

$$\theta = (\theta_0, \theta_1, \theta_2, \dots, \theta_m)$$

Testing

Given an observation $\boldsymbol{X} = (X_1, X_2, \dots, X_m)$, predict

$$\hat{Y} = \arg\max_{y=\{0,1\}} P(Y|\boldsymbol{X})$$

Training: The big picture

Logistic regression classifier

$$\hat{Y} = \arg\max_{y=\{0,1\}} P(Y|\boldsymbol{X})$$

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\sum_{j=0}^{m} \theta_j x_j\right) = \sigma(\theta^T \boldsymbol{x})$$

Training

Estimate parameters from training data

$$\theta = (\theta_0, \theta_1, \theta_2, \ldots, \theta_m)$$

Choose θ that optimizes some objective:

1. Determine objective function
2. Find gradient with respect to θ
3. Solve analytically by setting to 0, or computationally with gradient ascent

We are modeling $P(Y|X)$ directly, so we maximize the **conditional likelihood** of training data.

Estimating θ

1. Determine objective function

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right)$$

2. Gradient w.r.t. θ_j, for $j = 0, 1, \ldots, m$

3. Solve
   - No analytical derivation of θ_{MLE}…
   - …but can still compute θ_{MLE} with gradient ascent!

```
initialize x
repeat many times:
  compute gradient
     x += η * gradient
```

1. Determine objective function

$$\theta_{MLE} = \boxed{\arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right)} = \boxed{\arg\max_{\theta} \; LL(\theta)}$$

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\sum_{j=0}^{m} \theta_j x_j\right)$$
$$= \sigma(\theta^T \boldsymbol{x})$$

First: Interpret conditional likelihood with Logistic Regression

Second: Write a differentiable expression for log conditional likelihood

1. Determine objective function (interpret)

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right) = \arg\max_{\theta} LL(\theta)$$

$$P(Y = 1 \mid \boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\sum_{j=0}^{m} \theta_j x_j\right) = \sigma(\theta^T \boldsymbol{x})$$

Suppose you have $n = 2$ training datapoints: $\left(\boldsymbol{x}^{(1)}, 1\right), \left(\boldsymbol{x}^{(2)}, 0\right)$

Consider the following expressions for a given θ:

A. $\sigma\left(\theta^T \boldsymbol{x}^{(1)}\right) \sigma\left(\theta^T \boldsymbol{x}^{(2)}\right)$

B. $\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(1)}\right)\right) \sigma\left(\theta^T \boldsymbol{x}^{(2)}\right)$

C. $\sigma\left(\theta^T \boldsymbol{x}^{(1)}\right) \left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(2)}\right)\right)$

D. $\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(1)}\right)\right) \left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(2)}\right)\right)$

1. Interpret the above expressions as probabilities.
2. If we let $\theta = \theta_{MLE}$, which probability should be highest?

1. Determine objective function (interpret)

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right) = \arg\max_{\theta} LL(\theta)$$

$$P(Y = 1 | \boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\sum_{j=0}^{m} \theta_j x_j\right) = \sigma(\theta^T \boldsymbol{x})$$

Suppose you have $n = 2$ training datapoints: $\left(\boldsymbol{x}^{(1)}, 1\right), \left(\boldsymbol{x}^{(2)}, 0\right)$

Consider the following expressions for a given θ:

A. $\sigma\left(\theta^T \boldsymbol{x}^{(1)}\right) \sigma\left(\theta^T \boldsymbol{x}^{(2)}\right)$

B. $\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(1)}\right)\right) \sigma\left(\theta^T \boldsymbol{x}^{(2)}\right)$

C. $\sigma\left(\theta^T \boldsymbol{x}^{(1)}\right) \left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(2)}\right)\right)$

D. $\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(1)}\right)\right) \left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(2)}\right)\right)$

1. Interpret the above expressions as probabilities.
2. If we let $\theta = \theta_{MLE}$, which probability should be highest?

1. Determine objective function (write)

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right) = \arg\max_{\theta} \; LL(\theta)$$

$$P(Y = 1 | \boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\textstyle\sum_{j=0}^{m} \theta_j x_j\right)$$
$$= \sigma(\theta^T \boldsymbol{x})$$

1. What is a differentiable expression for $P(Y = y \mid \boldsymbol{X} = \boldsymbol{x})$?

$$P(Y = y | \boldsymbol{X} = \boldsymbol{x}) = \begin{cases} \sigma(\theta^T \boldsymbol{x}) & \text{if } y = 1 \\ 1 - \sigma(\theta^T \boldsymbol{x}) & \text{if } y = 0 \end{cases}$$

2. What is a differentiable expression for $LL(\theta)$, log conditional likelihood?

$$LL(\theta) = \log \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right)$$

1. Determine objective function (write)

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \middle| \boldsymbol{x}^{(i)}, \theta\right) = \arg\max_{\theta} LL(\theta)$$

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\textstyle\sum_{j=0}^{m} \theta_j x_j\right) = \sigma(\theta^T \boldsymbol{x})$$

1. What is a differentiable expression for $P(Y = y|\,\boldsymbol{X} = \boldsymbol{x})$?

$$P(Y = y|\boldsymbol{X} = \boldsymbol{x}) = \begin{cases} \sigma(\theta^T \boldsymbol{x}) & \text{if } y = 1 \\ 1 - \sigma(\theta^T \boldsymbol{x}) & \text{if } y = 0 \end{cases}$$

Recall Bernoulli MLE!

2. What is a differentiable expression for $LL(\theta)$, log conditional likelihood?

$$LL(\theta) = \log \prod_{i=1}^{n} f\left(y^{(i)} \middle| \boldsymbol{x}^{(i)}, \theta\right)$$

1. Determine objective function (write)

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right) = \arg\max_{\theta} \, LL(\theta)$$

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\textstyle\sum_{j=0}^{m} \theta_j x_j\right)$$
$$= \sigma(\theta^T \boldsymbol{x})$$

1. What is a differentiable expression for $P(Y = y| \boldsymbol{X} = \boldsymbol{x})$?

$$P(Y = y|\boldsymbol{X} = \boldsymbol{x}) = \left(\sigma(\theta^T \boldsymbol{x})\right)^y \left(1 - \sigma(\theta^T \boldsymbol{x})\right)^{1-y}$$

2. What is a differentiable expression for $LL(\theta)$, log conditional likelihood?

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right)$$

2. Find gradient with respect to θ

Optimization problem:

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right) = \arg\max_{\theta} LL(\theta)$$

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right)$$

Gradient w.r.t. θ_j, for $j = 0, 1, \ldots, m$:

$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right] x_j^{(i)}$$

(derived later)

How do we interpret the gradient contribution of the i-th training datapoint?

2. Find gradient with respect to θ

Optimization problem:

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right) = \arg\max_{\theta} \; LL(\theta)$$

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right)$$

Gradient w.r.t. θ_j, for $j = 0, 1, \ldots, m$:

$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right] x_j^{(i)}$$

(derived later)

scale by j-th feature

2. Find gradient with respect to θ

Optimization problem:

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right) = \arg\max_{\theta} LL(\theta)$$

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right)$$

Gradient w.r.t. θ_j, for $j = 0, 1, \ldots, m$:

$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \left[\underbrace{y^{(i)}}_{\text{1 or 0}} - \underbrace{\sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)}_{P\left(Y = 1 \mid \boldsymbol{X} = \boldsymbol{x}^{(i)}\right)}\right] x_j^{(i)}$$

(derived later)

2. Find gradient with respect to θ

Optimization problem:

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right) = \arg\max_{\theta} \; LL(\theta)$$

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right)$$

Gradient w.r.t. θ_j, for $j = 0, 1, \ldots, m$:

$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \underbrace{\left[y^{(i)} - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right]} x_j^{(i)}$$

(derived later)

Suppose $y^{(i)} = 1$ (the true class label for i-th datapoint):

- If $\sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) \geq 0.5$, correct
- If $\sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) < 0.5$, incorrect → change θ_j more

3. Solve

1. Optimization problem:

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f\left(y^{(i)} \mid \boldsymbol{x}^{(i)}, \theta\right) = \arg\max_{\theta} \; LL(\theta)$$

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right)$$

2. Gradient w.r.t. θ_j, for $j = 0, 1, \dots, m$:

$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right] x_j^{(i)}$$

3. Solve

Stay tuned!

(live)
25: Logistic Regression

Slides by Lisa Yan

August 12, 2020

Logistic Regression Model

Review

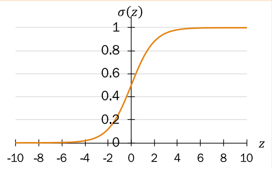
$$\hat{Y} = \underset{y=\{0,1\}}{\arg\max}\, P(Y|\boldsymbol{X})$$

$\hat{Y}$ is prediction of Y

$$P(Y=1|\boldsymbol{X}=\boldsymbol{x}) = \sigma\left(\sum_{j=0}^{m} \theta_j x_j\right) = \sigma(\theta^T \boldsymbol{x})$$

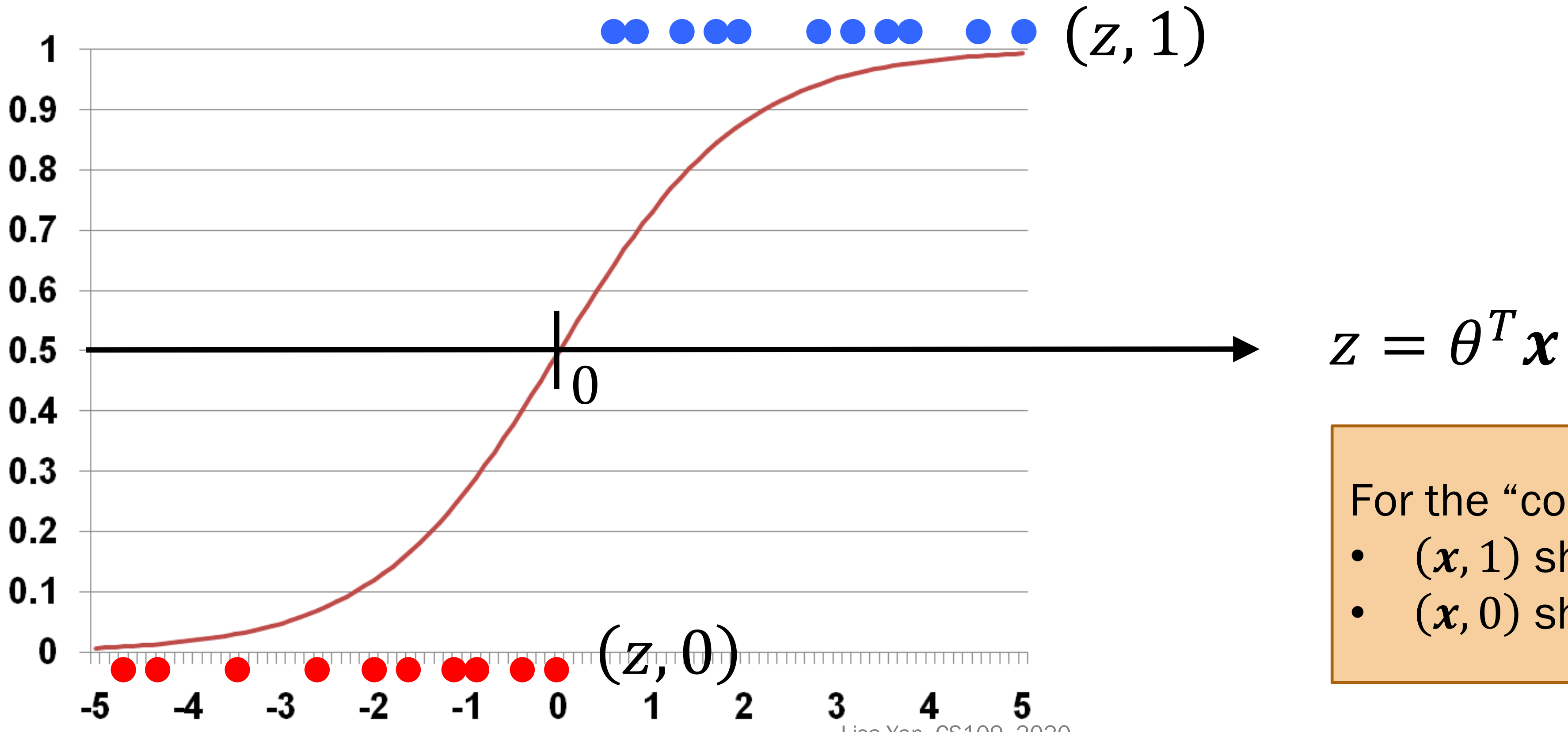
where $x_0 = 1$

$\boldsymbol{X}$ → $\theta_0 + \sum_{j=1}^{m} \theta_j X_j$ → sigmoid function $\sigma(z) = \frac{1}{1+e^{-z}}$ → $\hat{P}(Y=1|\boldsymbol{X})$



Another view of Logistic Regression

Logistic Regression Model $P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma(\theta^T \boldsymbol{x})$ where $\theta^T \boldsymbol{x} = \sum_{j=0}^{m} \theta_j x_j$



For the "correct" parameters θ:

- $(\boldsymbol{x}, 1)$ should have $\theta^T x > 0$
- $(\boldsymbol{x}, 0)$ should have $\theta^T x \leq 0$

Learning parameters

Training

Learn parameters $\theta = (\theta_0, \theta_1, \dots, \theta_m)$

$$\theta_{MLE} = \arg\max_{\theta} LL(\theta)$$

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right)$$

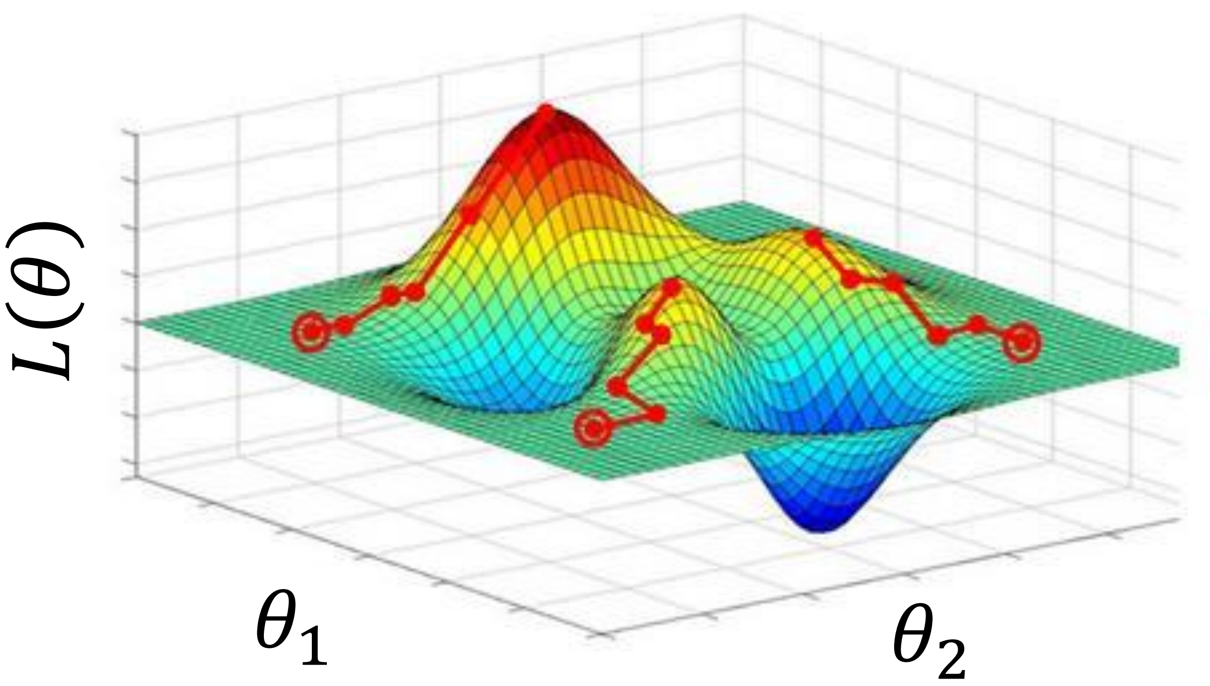
$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right] x_j^{(i)} \qquad \text{for } j = 0, 1, \dots, m$$

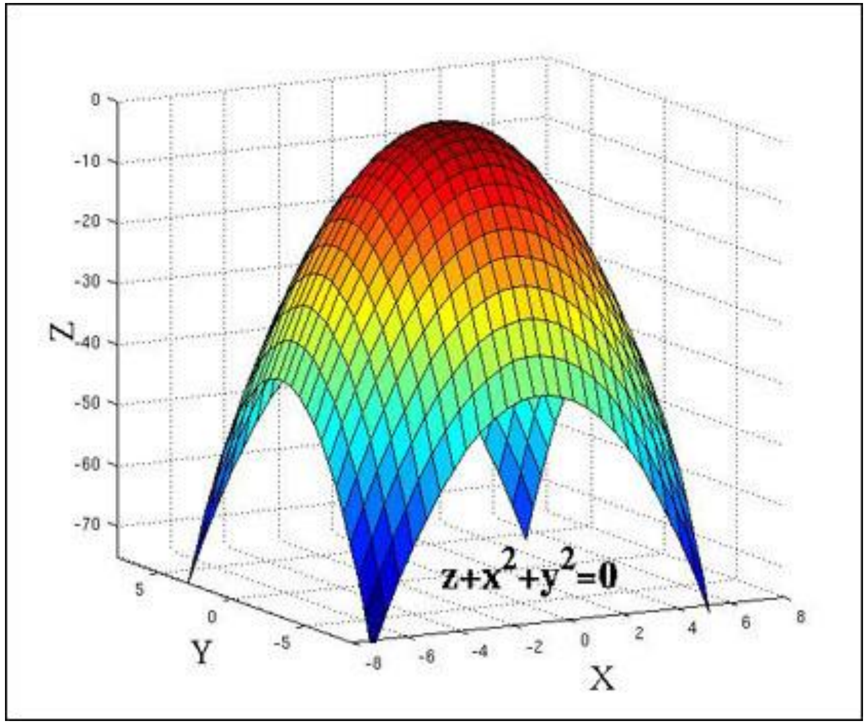
- No analytical derivation of θ_{MLE}...
- ...but can still compute θ_{MLE} with gradient ascent!

Gradient Ascent

Review

Walk uphill and you will find a local maxima
(if your step is small enough).





Logistic regression $LL(\theta)$
is concave

Training: The details

Training: Gradient ascent step

3. Optimize.

$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^T \boldsymbol{x}^{(\boldsymbol{i})}\right) \right] x_j^{(i)}$$

repeat many times:

for all thetas:

$$\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \frac{\partial LL\left(\theta^{\text{old}}\right)}{\partial \theta_j^{\text{old}}}$$

$$= \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left({\theta^{\text{old}}}^T \boldsymbol{x}^{(\boldsymbol{i})} \right) \right] x_j^{(i)}$$

What does this look like in code?

Training: Gradient Ascent

Gradient Ascent Step $\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^{\text{old}^T} \boldsymbol{x}^{(i)} \right) \right] x_j^{(i)}$

initialize θ_j = 0 for 0 ≤ j ≤ m
repeat many times:

gradient[j] = 0 for 0 ≤ j ≤ m

// compute all gradient[j]'s
// based on n training examples

θ_j += η * gradient[j] for all 0 ≤ j ≤ m

Training: Gradient Ascent

Gradient Ascent Step $\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^{\text{old}^T} \boldsymbol{x}^{(i)} \right) \right] x_j^{(i)}$

initialize θ_j = 0 for 0 ≤ j ≤ m
repeat many times:

    gradient[j] = 0 for 0 ≤ j ≤ m

    for each training example (x, y):

        for each 0 ≤ j ≤ m:

            // update gradient[j] for
            // current (x,y) example

    θ_j += η * gradient[j] for all 0 ≤ j ≤ m

Training: Gradient Ascent

Gradient Ascent Step $\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left({\theta^{\text{old}}}^T \boldsymbol{x}^{(i)}\right)\right] x_j^{(i)}$

initialize θ_j = 0 for 0 ≤ j ≤ m

repeat many times:

    gradient[j] = 0 for 0 ≤ j ≤ m

    for each training example (x, y):

        for each 0 ≤ j ≤ m:

            gradient[j] += $\left[y - \frac{1}{1 + e^{-\theta^T \boldsymbol{x}}} \right] x_j$

    θ_j += η * gradient[j] for all 0 ≤ j ≤ m

What are the important details?

Training: Gradient Ascent

Gradient Ascent Step $\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left({\theta^{\text{old}}}^T \boldsymbol{x}^{(i)} \right) \right] x_j^{(i)}$

initialize θ_j = 0 for 0 ≤ j ≤ m
repeat many times:

gradient[j] = 0 for 0 ≤ j ≤ m

for each training example (x, y):

for each 0 ≤ j ≤ m:

gradient[j] += $\left[y - \frac{1}{1 + e^{-\theta^T \boldsymbol{x}}} \right] \boxed{x_j}$

θ_j += η * gradient[j] for all 0 ≤ j ≤ m

- x_j is j-th feature of input $\boldsymbol{x} = (x_1, \ldots, x_m)$

Training: Gradient Ascent

Gradient Ascent Step $\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left({\theta^{\text{old}}}^T \boldsymbol{x}^{(i)} \right) \right] x_j^{(i)}$

initialize θ_j = 0 for 0 ≤ j ≤ m
repeat many times:

gradient[j] = 0 for 0 ≤ j ≤ m

for each training example (x, y):

for each 0 ≤ j ≤ m:

gradient[j] += $\left[y - \frac{1}{1 + \boxed{e^{-\theta^T \boldsymbol{x}}}} \right] x_j$

θ_j += η * gradient[j] for all 0 ≤ j ≤ m

- x_j is j-th feature of input $\boldsymbol{x} = (x_1, \dots, x_m)$
- Insert $x_0 = 1$ before training

Training: Gradient Ascent

Gradient Ascent Step $\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^{\text{old}^T} \boldsymbol{x}^{(i)} \right) \right] x_j^{(i)}$

initialize θ_j = 0 for 0 ≤ j ≤ m
repeat many times:

    gradient[j] = 0 for 0 ≤ j ≤ m

    for each training example (x, y):

        for each 0 ≤ j ≤ m:

            gradient[j] += $\left[y - \dfrac{1}{1 + e^{-\theta^T \boldsymbol{x}}} \right] x_j$

    θ_j += η * gradient[j] for all 0 ≤ j ≤ m

- x_j is j-th feature of input $\boldsymbol{x} = (x_1, \ldots, x_m)$
- Insert $x_0 = 1$ before training
- Finish computing gradient before updating any part of θ

Training: Gradient Ascent

Gradient Ascent Step $\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left({\theta^{\text{old}}}^T \boldsymbol{x}^{(i)} \right) \right] x_j^{(i)}$

initialize θ_j = 0 for 0 ≤ j ≤ m

repeat many times:

  gradient[j] = 0 for 0 ≤ j ≤ m

  for each training example (x, y):

    for each 0 ≤ j ≤ m:

      gradient[j] += $\left[y - \frac{1}{1 + e^{-\theta^T \boldsymbol{x}}} \right] x_j$

  θ_j += η * $\boxed{\text{gra}}$dient[j] for all 0 ≤ j ≤ m

- x_j is j-th feature of input $\boldsymbol{x} = (x_1, \ldots, x_m)$
- Insert $x_0 = 1$ before training
- Finish computing gradient before updating any part of θ
- Learning rate η is a constant you set before training

Training: Gradient Ascent

Gradient Ascent Step $\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left({\theta^{\text{old}}}^T \boldsymbol{x}^{(i)} \right) \right] x_j^{(i)}$

initialize θ_j = 0 for 0 ≤ j ≤ m

repeat many times:

  gradient[j] = 0 for 0 ≤ j ≤ m

  for each training example (x, y):

    for each 0 ≤ j ≤ m:

      gradient[j] += $\left[y - \frac{1}{1 + e^{-\theta^T \boldsymbol{x}}} \right] x_j$

  θ_j += η * gradient[j] for all 0 ≤ j ≤ m

- x_j is *j*-th feature of input $\boldsymbol{x} = (x_1, \ldots, x_m)$
- Insert $x_0 = 1$ before training
- Finish computing gradient before updating any part of θ
- Learning rate η is a constant you set before training

Testing

Introducing notation $\hat{y}$

$$\hat{Y} = \arg\max_{y=\{0,1\}} P(Y|\boldsymbol{X})$$

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\textstyle\sum_{j=0}^{m} \theta_j x_j\right) = \sigma(\theta^T \boldsymbol{x})$$

$\hat{Y}$ is prediction of Y

$$\hat{y} = P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma(\theta^T \boldsymbol{x})$$

Small $\hat{y}$ is conditional probability

$$P(Y = y|\boldsymbol{X} = \boldsymbol{x}) = \begin{cases} \hat{y} & \text{if } y = 1 \\ 1 - \hat{y} & \text{if } y = 0 \end{cases}$$

Testing: Classification with Logistic Regression

Training

Learn parameters $\theta = (\theta_0, \theta_1, \dots, \theta_m)$

via gradient ascent:

$$\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left({\theta^{\text{old}}}^T \boldsymbol{x}^{(i)}\right)\right] x_j^{(i)}$$

Testing

- Compute $\hat{y} = P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma(\theta^T \boldsymbol{x}) = \dfrac{1}{1 + e^{-\theta^T \boldsymbol{x}}}$
- Classify instance as:

$$\begin{cases} 1 & \hat{y} > 0.5, \text{ equivalently } \theta^T \boldsymbol{x} > 0 \\ 0 & \text{otherwise} \end{cases}$$

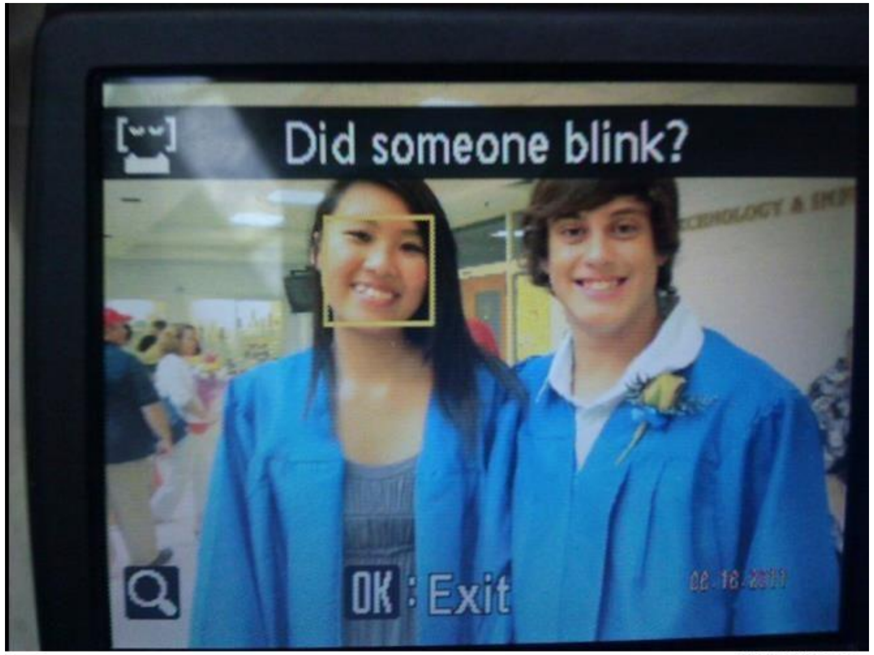
⚠ Parameters θ_j are <u>not</u> updated during testing phase

Interlude for jokes/announcements

Announcements

1. Pset 6 due tomorrow at 1pm. No late days or on-time bonus for this pset.
2. Look out for extra office hours + review session for the Final Quiz
3. Final Quiz begins Friday 5pm and ends Sunday 5pm.
4. You're so close, you got this!

Ethics and datasets



Sometimes machine learning feels universally unbiased.

We can even prove our estimators are “unbiased” (mathematically).

Google/Nikon/HP had biased datasets.

Should your data be unbiased?

Dataset: Google News

$$\overrightarrow{\text{man}} - \overrightarrow{\text{woman}} \approx \overrightarrow{\text{king}} - \overrightarrow{\text{queen}}$$

$$\overrightarrow{\text{man}} - \overrightarrow{\text{woman}} \approx \overrightarrow{\text{computer programmer}} - \overrightarrow{\text{homemaker}}.$$

Should our unbiased data collection reflect society's systemic bias?

Bolukbasi et al., Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings. NIPS 2016

How can we explain decisions?

If your task is image classification, reasoning about high-level features is relatively easy.

Everything can be visualized.

What if you are trying to classify social outcomes?

- Criminal recidivism
- Job performance
- Policing
- Terrorist risk
- At-risk kids

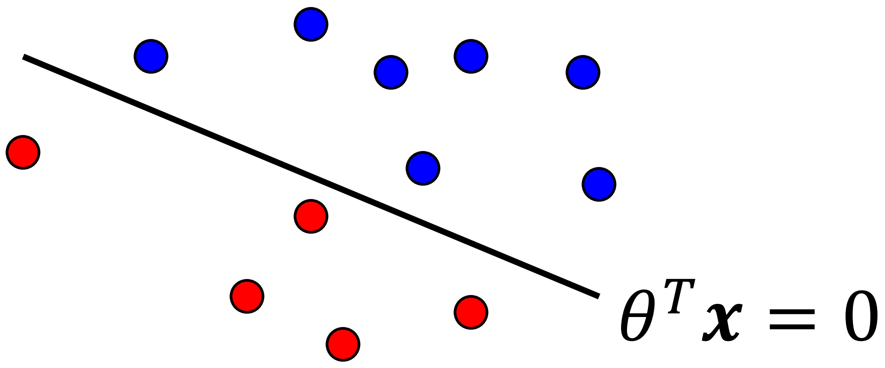
Ethics in Machine Learning is a whole new field. ☺

Philosophy

Intuition about Logistic Regression

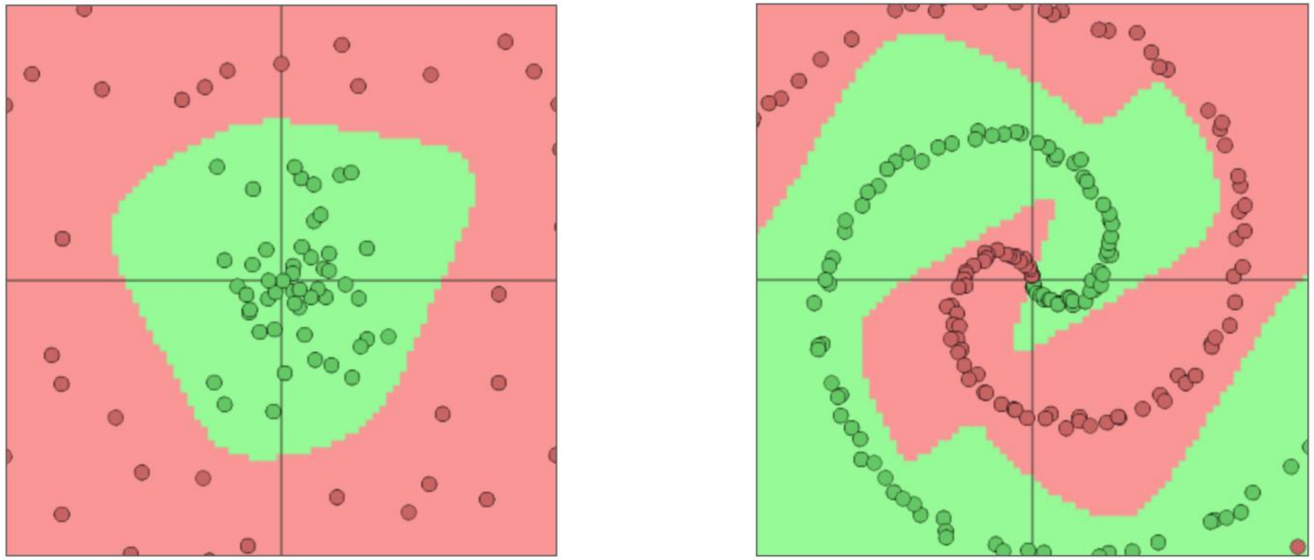
Logistic Regression Model $P(Y = 1 | \boldsymbol{X} = \boldsymbol{x}) = \sigma(\theta^T \boldsymbol{x})$ where $\theta^T \boldsymbol{x} = \sum_{j=0}^{m} \theta_j x_j$

Logistic Regression is trying to fit a **line** that separates data instances where $y = 1$ from those where $y = 0$:



- We call such data (or functions generating the data **linearly separable**.
- Naïve Bayes is linear too, because there is no interaction between different features.

Data is often not linearly separable



- Not possible to draw a line that successfully separates all the $y = 1$ points (green) from the $y = 0$ points (red)
- Despite this fact, Logistic Regression and Naive Bayes still often work well in practice

Many tradeoffs in choosing an algorithm

	Naïve Bayes	Logistic Regression
Modeling goal	$P(\boldsymbol{X}, Y)$	$P(Y \mid \boldsymbol{X})$
Generative or **discriminative**?	**Generative**: could use joint distribution to generate new points (⚠ but you might not need this extra effort)	**Discriminative**: just tries to discriminate $y = 0$ vs $y = 1$ (✕ cannot generate new points b/c no $P(\boldsymbol{X}, Y)$)
Continuous input features	⚠ Needs parametric form (e.g., Gaussian) or discretized buckets (for multinomial features)	☑ Yes, easily
Discrete input features	☑ Yes, multi-value discrete data = multinomial $P(X_i \mid Y)$	⚠ Multi-valued discrete data hard (e.g., if $X_i \in \{A, B, C\}$, not necessarily good to encode as $\{1, 2, 3\}$

Gradient Derivation

Background: Calculus

Calculus refresher #1:
Derivative(sum) =
sum(derivative)

$$\frac{\partial}{\partial x}\sum_{i=1}^{n} f_i(x) = \sum_{i=1}^{n} \frac{\partial f_i(x)}{\partial x}$$

Calculus refresher #2:
Chain rule ✨ ✨ ✨

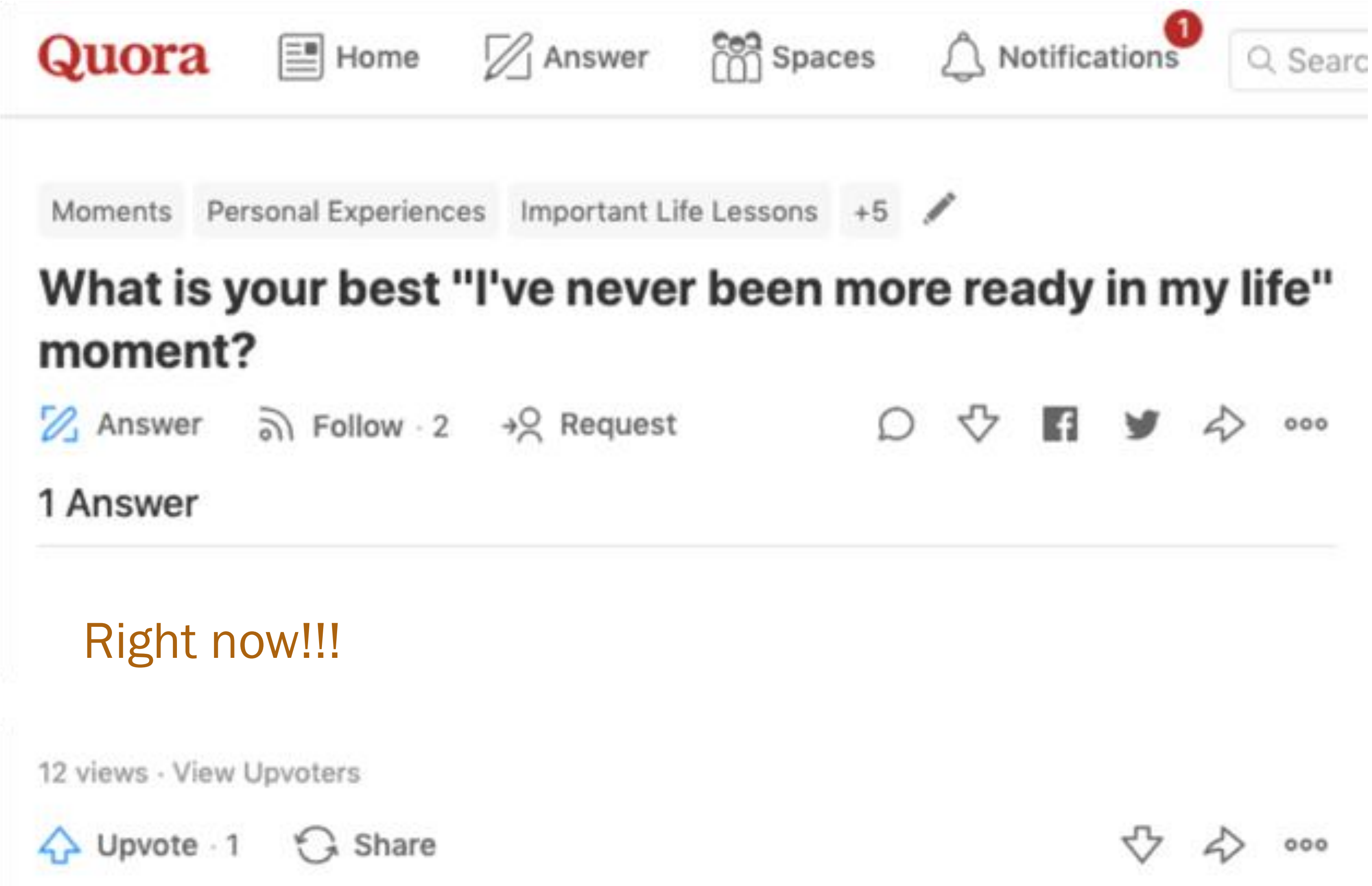
$$\frac{\partial f(x)}{\partial x} = \frac{\partial f(z)}{\partial z}\frac{\partial z}{\partial x}$$

Calculus Chain Rule

$$f(x) = f\big(z(x)\big)$$

aka decomposition
of composed functions

Are you ready?



Compute gradient of log conditional likelihood

Find: $\dfrac{\partial LL(\theta)}{\partial \theta_j}$ where

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right)$$

log conditional likelihood

Aside: Sigmoid has a beautiful derivative

Sigmoid function:

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

Derivative:

$$\frac{d}{dz}\sigma(z) = \sigma(z)[1 - \sigma(z)]$$

What is $\frac{\partial}{\partial \theta_j}\sigma(\theta^T \boldsymbol{x})$?

A. $\sigma(x_j)[1 - \sigma(x_j)]x_j$

B. $\sigma(\theta^T \boldsymbol{x})[1 - \sigma(\theta^T \boldsymbol{x})]\boldsymbol{x}$

C. $\sigma(\theta^T \boldsymbol{x})[1 - \sigma(\theta^T \boldsymbol{x})]x_j$

D. $\sigma(\theta^T \boldsymbol{x})x_j[1 - \sigma(\theta^T \boldsymbol{x})x_j]$

E. None/other

Aside: Sigmoid has a beautiful derivative

Sigmoid function:

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

Derivative:

$$\frac{d}{dz}\sigma(z) = \sigma(z)[1-\sigma(z)]$$

What is $\frac{\partial}{\partial \theta_j}\sigma(\theta^T \boldsymbol{x})$?

A. $\sigma(x_j)[1-\sigma(x_j)]x_j$

B. $\sigma(\theta^T \boldsymbol{x})[1-\sigma(\theta^T \boldsymbol{x})]\boldsymbol{x}$

C. $\sigma(\theta^T \boldsymbol{x})[1-\sigma(\theta^T \boldsymbol{x})]x_j$

D. $\sigma(\theta^T \boldsymbol{x})x_j[1-\sigma(\theta^T \boldsymbol{x})x_j]$

E. None/other

Let $z = \theta^T \boldsymbol{x} = \sum_{k=0}^{m} \theta_k x_k$.

$$\frac{\partial}{\partial \theta_j}\sigma(\theta^T \boldsymbol{x}) = \frac{\partial}{\partial z}\sigma(z) \cdot \frac{\partial z}{\partial \theta_j}$$

(Chain Rule)

$$= \sigma(\theta^T \boldsymbol{x})[1-\sigma(\theta^T \boldsymbol{x})]x_j$$

Re-itroducing notation $\hat{y}$

$$\hat{Y} = \arg\max_{y=\{0,1\}} P(Y|\boldsymbol{X})$$

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\sum_{j=0}^{m} \theta_j x_j\right) = \sigma(\theta^T \boldsymbol{x})$$

$$\hat{y} = P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma(\theta^T \boldsymbol{x})$$

$$P(Y = y|\boldsymbol{X} = \boldsymbol{x}) = \begin{cases} \hat{y} & \text{if } y = 1 \\ 1 - \hat{y} & \text{if } y = 0 \end{cases}$$

$$P(Y = y|\boldsymbol{X} = \boldsymbol{x}) = (\hat{y})^y (1 - \hat{y})^{1-y}$$

Compute gradient of log conditional likelihood

Find: $\dfrac{\partial LL(\theta)}{\partial \theta_j}$ where

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right)$$

log conditional likelihood

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right)$$

Compute gradient of log conditional likelihood

$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \frac{\partial}{\partial \theta_j}\left[y^{(i)} \log\left(\hat{y}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right)\right]$$

Let $\hat{y}^{(i)} = \sigma\left(\theta^T \boldsymbol{x}^{(\boldsymbol{i})}\right)$

$$= \sum_{i=1}^{n} \frac{\partial}{\partial \hat{y}^{(i)}}\left[y^{(i)} \log\left(\hat{y}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right)\right] \cdot \frac{\partial \hat{y}^{(i)}}{\partial \theta_j}$$

(Chain Rule)

$$= \sum_{i=1}^{n} \left[y^{(i)} \frac{1}{\hat{y}^{(i)}} - \left(1 - y^{(i)}\right) \frac{1}{1 - \hat{y}^{(i)}}\right] \cdot \hat{y}^{(i)}\left(1 - \hat{y}^{(i)}\right) x_j^{(i)}$$

(calculus)

$$= \sum_{i=1}^{n} \left[y^{(i)} - \hat{y}^{(i)}\right] x_j^{(i)} \qquad = \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right] x_j^{(i)}$$

(simplify)

Compute gradient of log conditional likelihood

$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \frac{\partial}{\partial \theta_j} \left[y^{(i)} \log\left(\hat{y}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right) \right]$$

Let $\hat{y}^{(i)} = \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)$

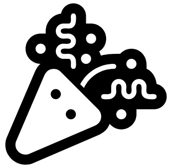
$$= \sum_{i=1}^{n} \frac{\partial}{\partial \hat{y}^{(i)}} \left[y^{(i)} \log\left(\hat{y}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right) \right] \cdot \frac{\partial \hat{y}^{(i)}}{\partial \theta_j}$$

(Chain Rule)

$$= \sum_{i=1}^{n} \left[y^{(i)} \frac{1}{\hat{y}^{(i)}} - \left(1 - y^{(i)}\right) \frac{1}{1 - \hat{y}^{(i)}} \right] \cdot \hat{y}^{(i)}\left(1 - \hat{y}^{(i)}\right) x_j^{(i)}$$

(calculus)

$$= \sum_{i=1}^{n} \left[y^{(i)} - \hat{y}^{(i)} \right] x_j^{(i)} \qquad = \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) \right] x_j^{(i)}$$



(simplify)

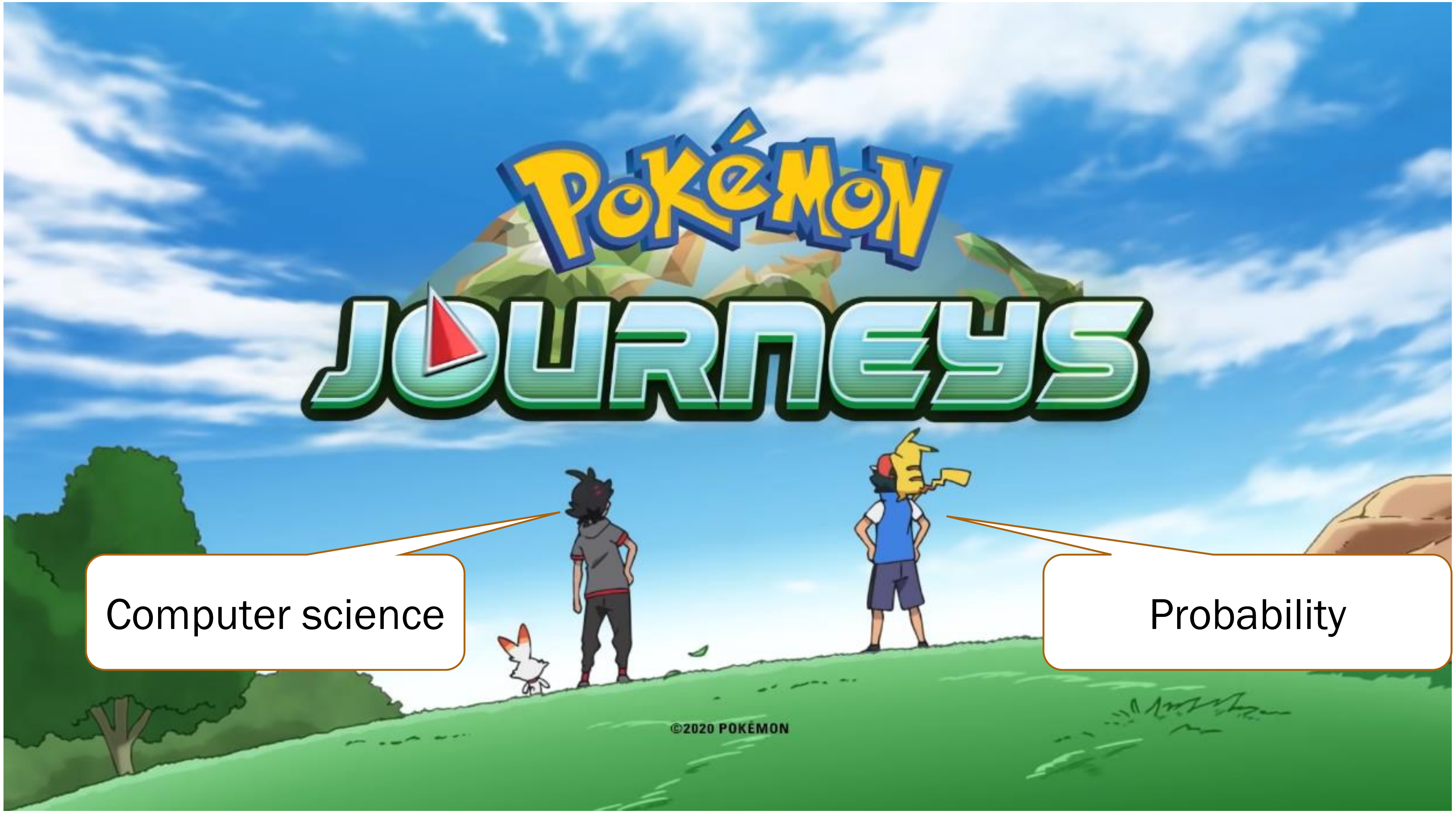
Wow. We did it!

LIVE

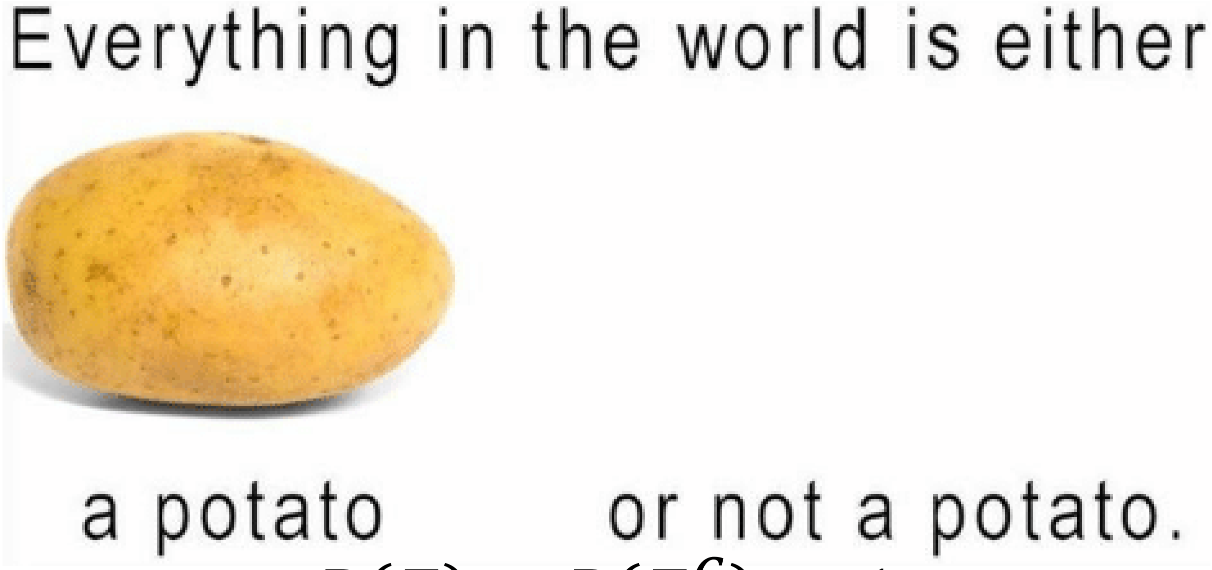
CS109 Wrap-

What have we learned in CS109?

A wild journey



From combinatorics to probability…

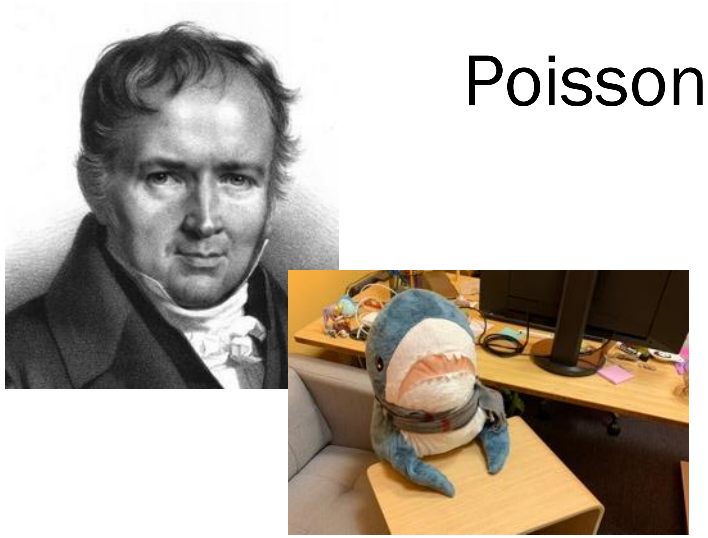


$$P(E) + P(E^C) = 1$$

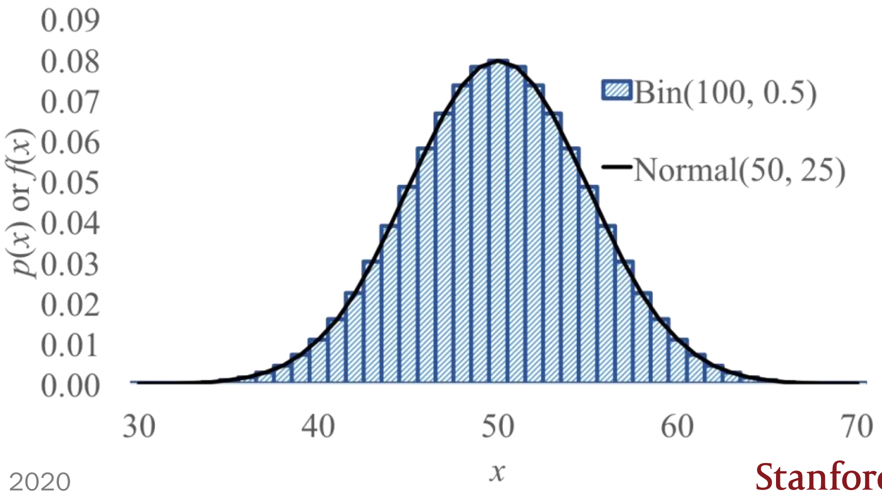
…to random variables and the Central Limit Theorem…

Bernoulli

Gaussian

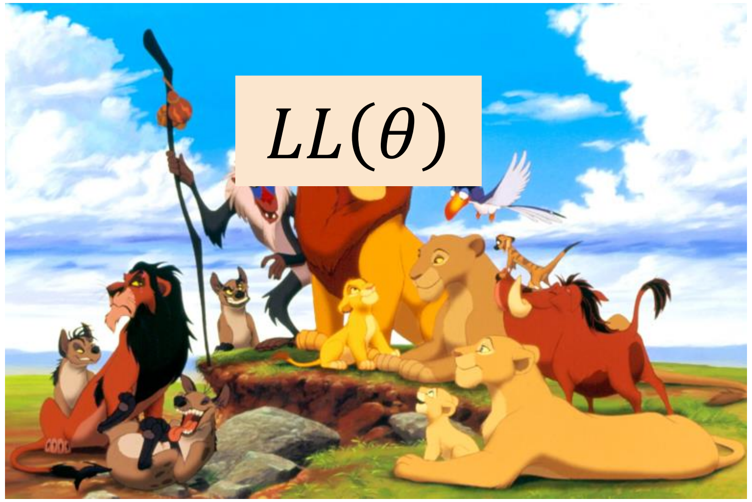


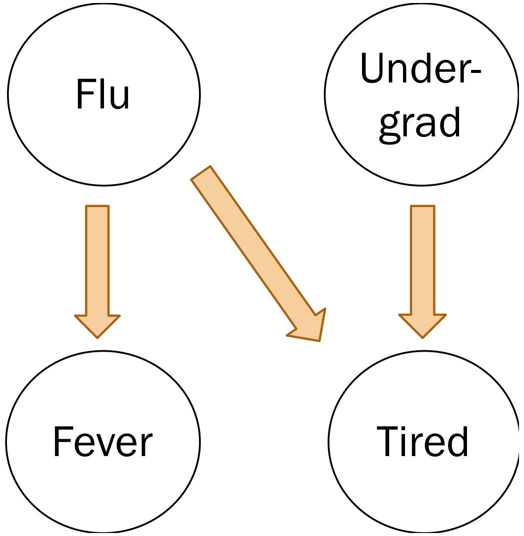
Poisson



…to statistics, parameter estimation, and machine learning

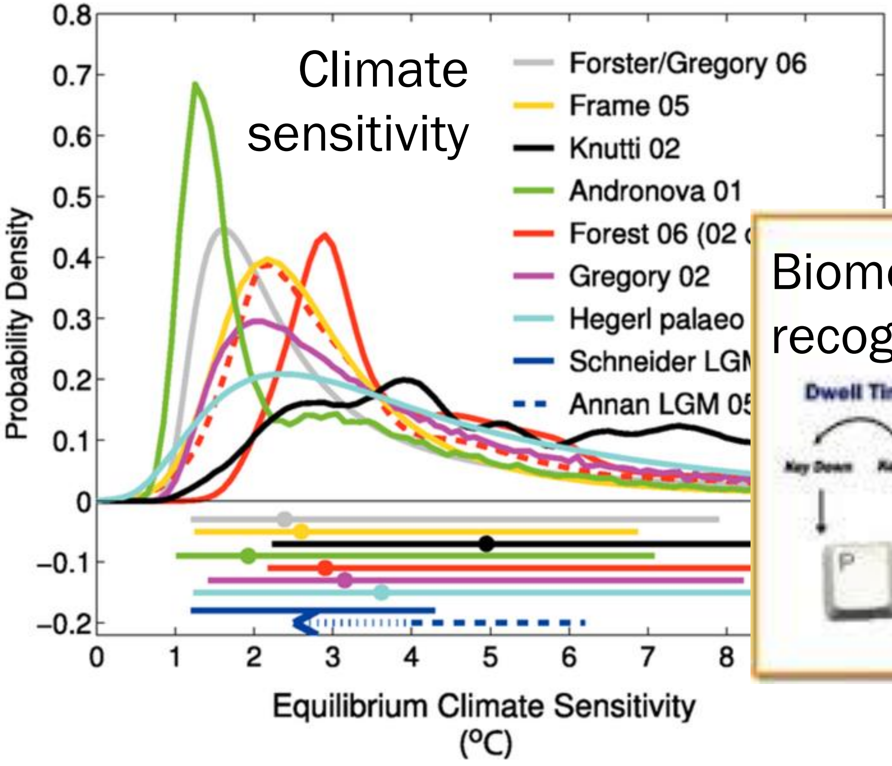
A happy
Bhutanese person



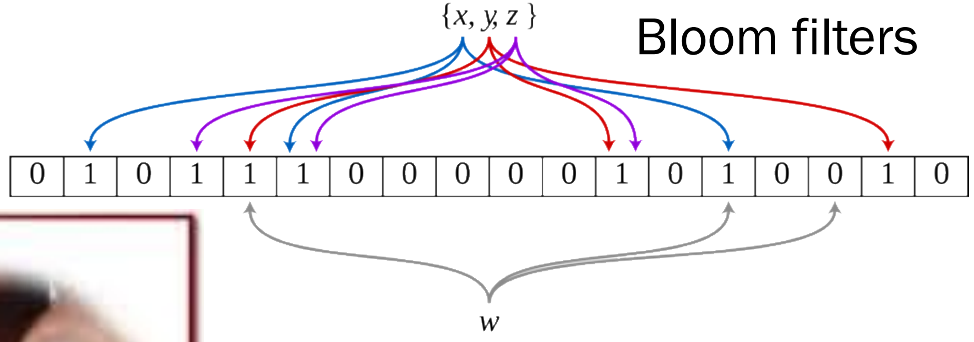


NETFLIX
and Learn

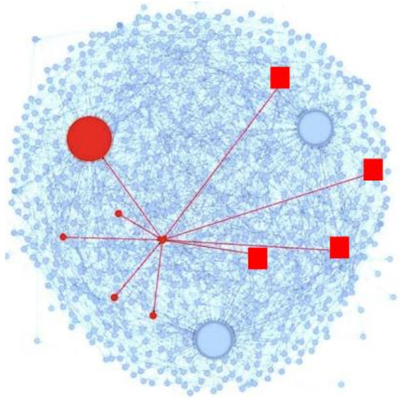
Lots and lots of analysis

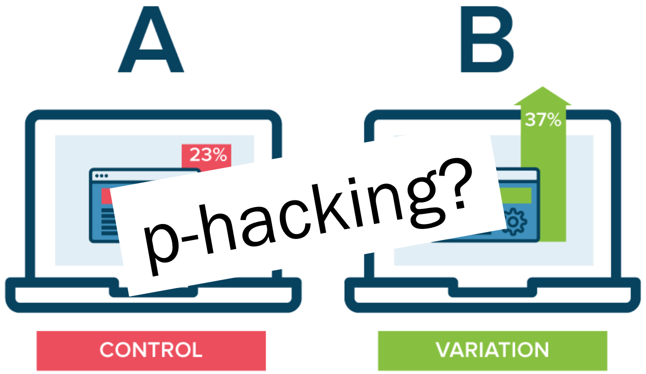


Bloom filters



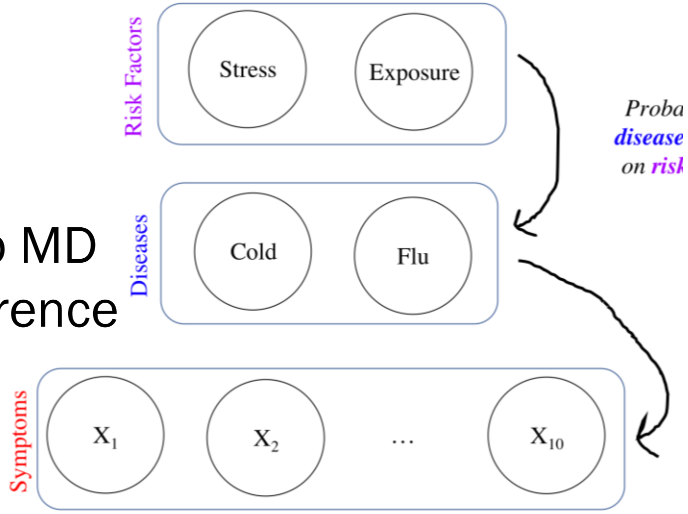
Peer
grading





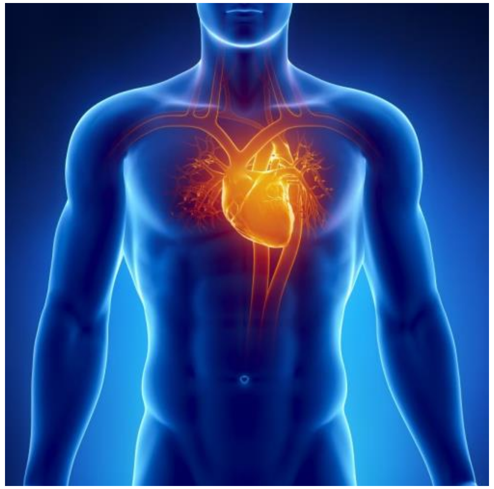
Coursera A/B testing

Web MD
inference



Lots and lots of analysis

Heart



Ancestry

Netflix

After CS109

Theory

CS161 – Algorithmic analysis
CS168 - ~Modern~ Algorithmic Analysis
Stats 217 – Stochastic Processes
CS238 – Decision Making Under Uncertainty
CS228 – Probabilistic Graphical Models

Statistics

Stats 200 – Statistical Inference
Stats 208 – Intro to the Bootstrap
Stats 209 – Group Methods/Causal Inference

After CS109

AI

CS 221 – Intro to AI
CS 229 – Machine Learning
CS 230 – Deep Learning
CS 224N – Natural Language Processing
CS 231N – Conv Neural Nets for Visual Recognition
CS 234 – Reinforcement Learning

Applications

CS 279 – Bio Computation
Literally any class with numbers in it

What do you want to remember in 5 years?

Why study probability + CS?

Why study probability + CS?

Fastest growing occupations: 20 occupations with the highest percent change of employment between 2018-28.

Click on an occupation name to see the full occupational profile.

OCCUPATION	GROWTH RATE, 2018-28	2018 MEDIAN PAY
Physician assistants	31%	$108,610 per year
Nurse practitioners	28%	$107,030 per year
Software developers, applications	26%	$103,620 per year
Mathematicians	26%	$101,900 per year
Information security analysts	32%	$98,350 per year
Health specialties teachers, postsecondary	23%	$97,370 per year
Statisticians	31%	$87,780 per year
Operations research analysts	26%	$83,390 per year
Genetic counselors	27%	$80,370 per year

Source: US Bureau of Labor Statistics

Why study probability + CS?

Interdisciplinary

Closest thing to magic

Why study probability + CS?

Everyone is welcome!

Technology magnifies.

What do we want magnified?

You are all one step closer to improving the world.

(all of you!)

The end