26: Logistic Regression

Lisa Yan and Jerry Cain

November 11, 2020

Quick slide reference

3	Background	26a_background
9	Logistic Regression	26b_logistic_regression
27	Training: The big picture	26c_lr_training
56	Training: The details, Testing	LIVE
59	Philosophy	LIVE
63	Gradient Derivation	26e_derivation

25a_background

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 $\boldsymbol{X} = (X_1, X_2, \ldots, X_m)$ 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 $X_0 = 1$

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

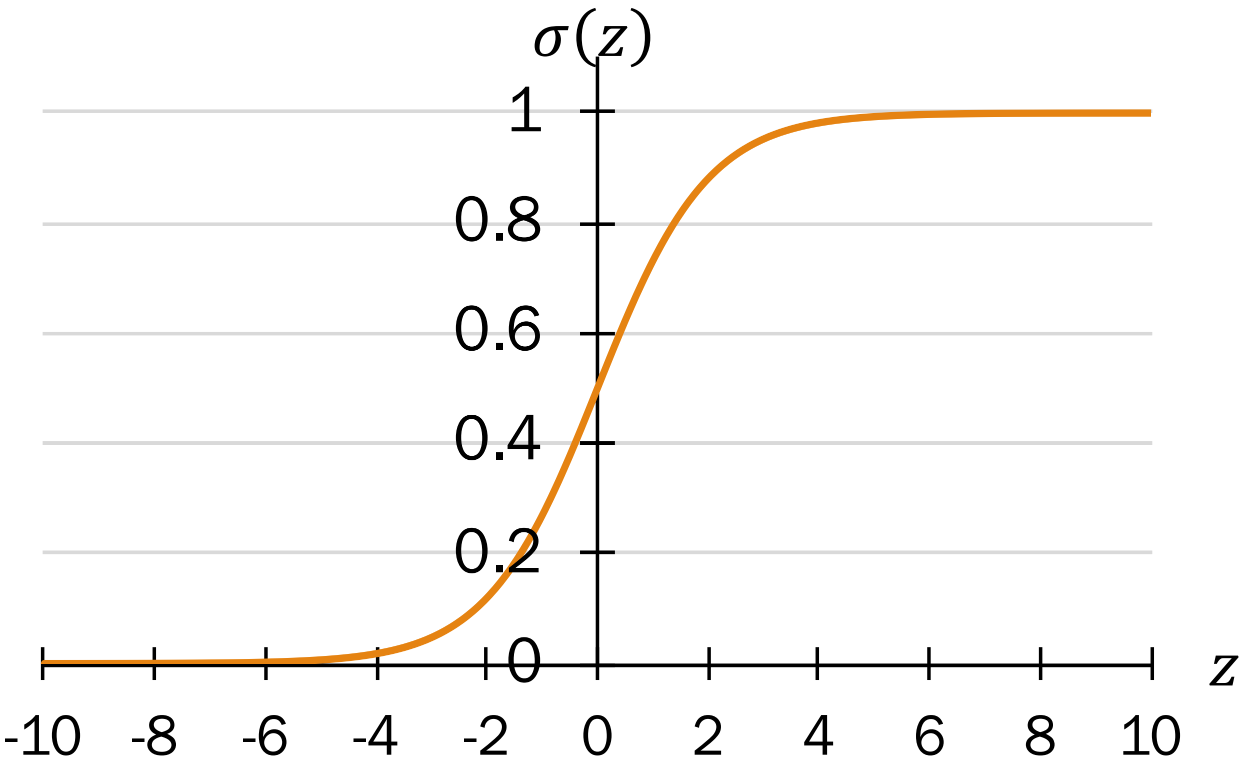
New $\boldsymbol{X} = (1, X_1, X_2, \ldots, X_m)$, $\theta = (\theta_0, \theta_1, \ldots, \theta_m)$

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

Review

Training data (n datapoints):

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

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

conditional likelihood of training data

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

log conditional likelihood

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

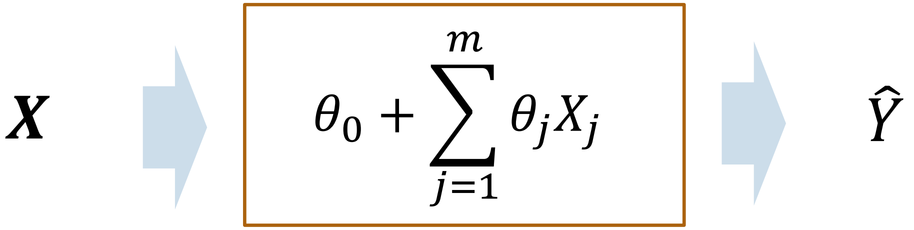
- MLE in this lecture is estimator that maximizes conditional likelihood
- Confusingly, log conditional likelihood is also written as $LL(\theta)$

Logistic Regression

Prediction models so far

Review

Linear Regression (Regression)



$\boldsymbol{X}$

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

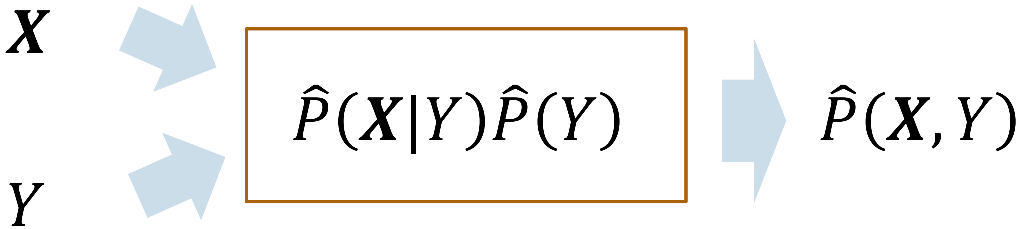
$\hat{Y}$

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

✅ $\boldsymbol{X}$ can be dependent

🤷 Regression model ($\hat{Y} \in \mathbb{R}$, not discrete)

Naïve Bayes (Classification)



$\boldsymbol{X}$

Y

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

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

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

✅ Tractable with NB assumption, but...

⚠️ Realistically, X_j features not necessarily conditionally independent

🤷 Actually models $P(\boldsymbol{X}, Y)$, not $P(Y|\boldsymbol{X})$?

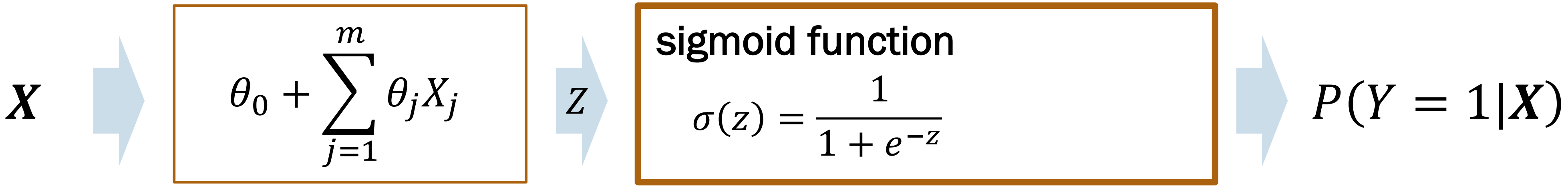
Introducing Logistic Regression!

Linear Regression ideas

Classification models

+ compute power

Logistic Regression



$\boldsymbol{X}$

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

z

sigmoid function

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

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

Logistic Regression Model:

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

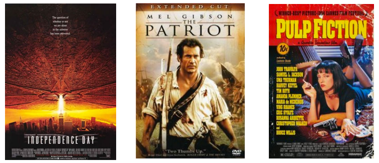
Predict $\hat{Y}$ as the most likely Y given our observation $\boldsymbol{X} = \boldsymbol{x}$:

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

- 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

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



$\boldsymbol{X}$

input features

θ parameter

0.81

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

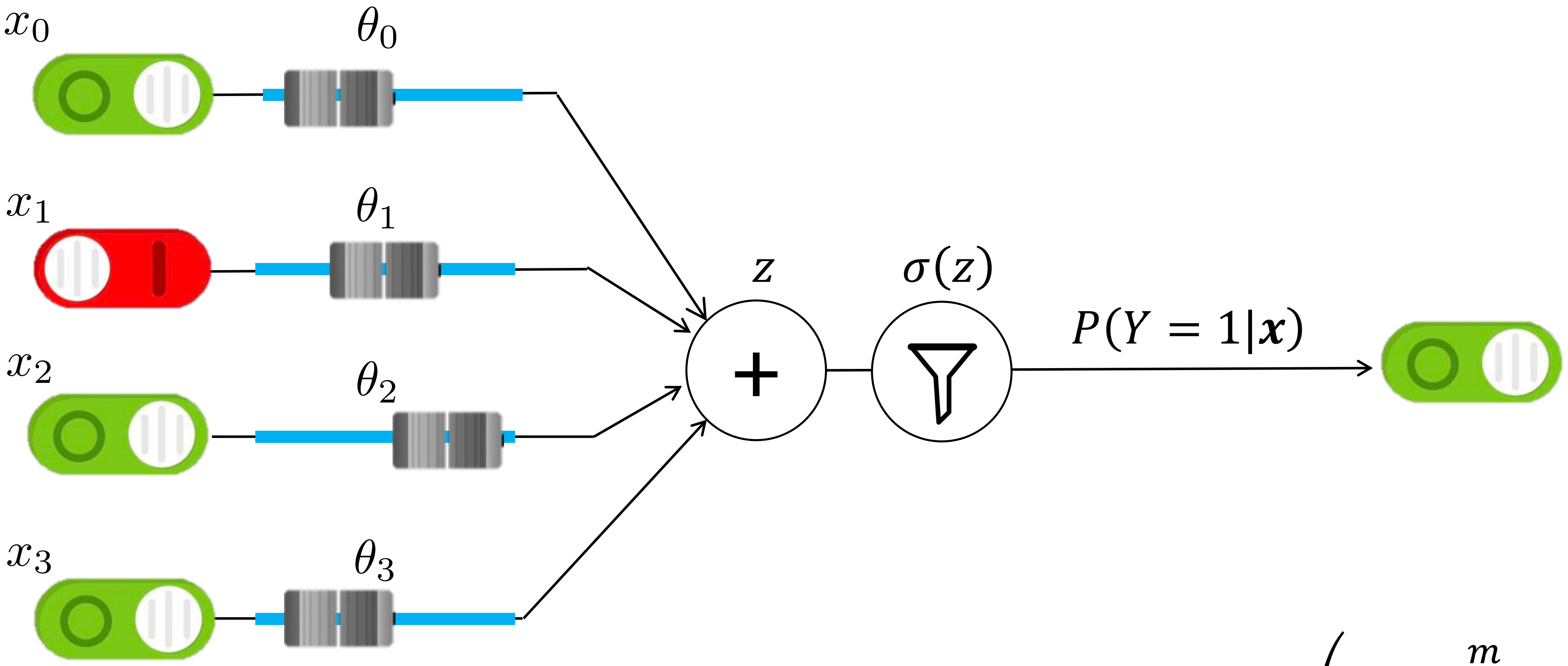
conditional likelihood

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

Logistic Regression cartoon

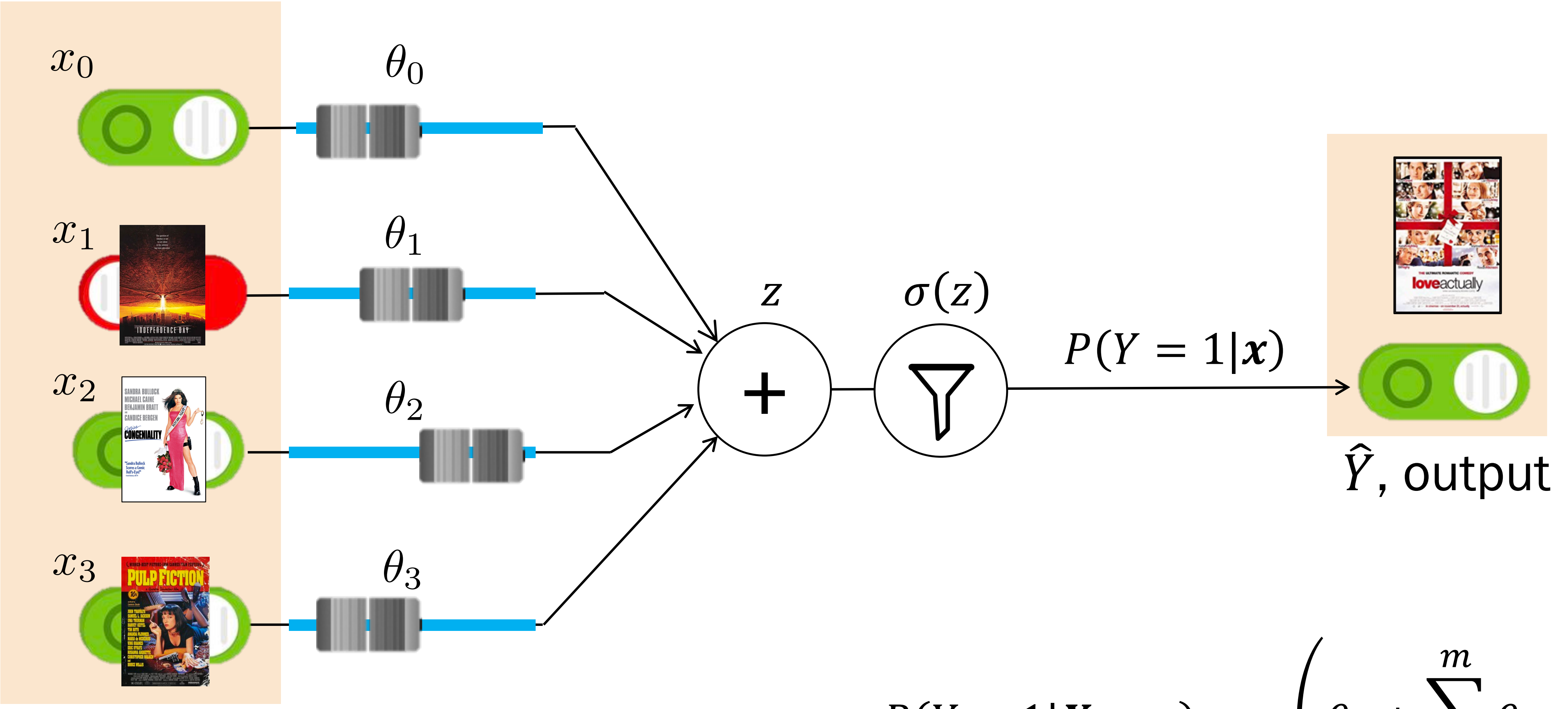
θ parameter

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

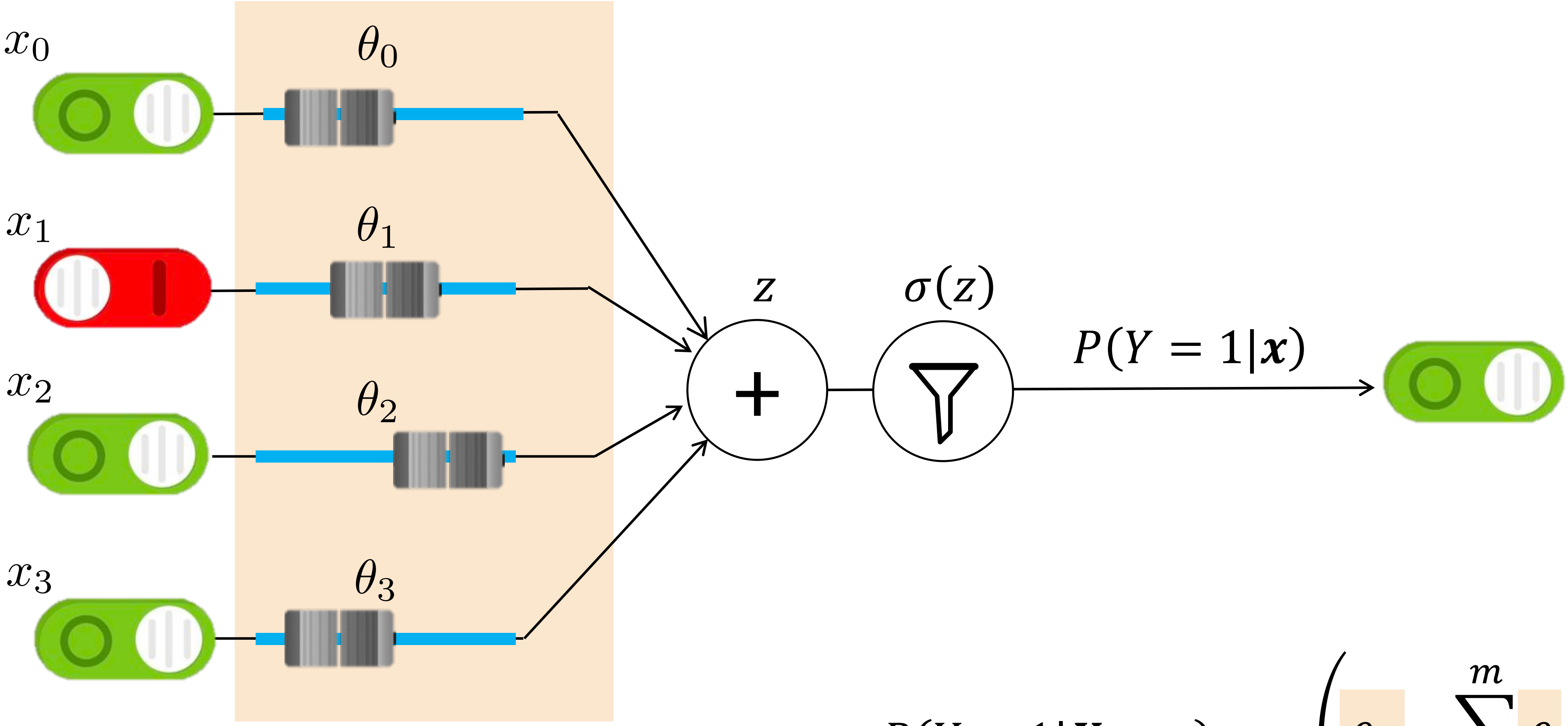


$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

$\boldsymbol{X}$, input features

[0,1,1]

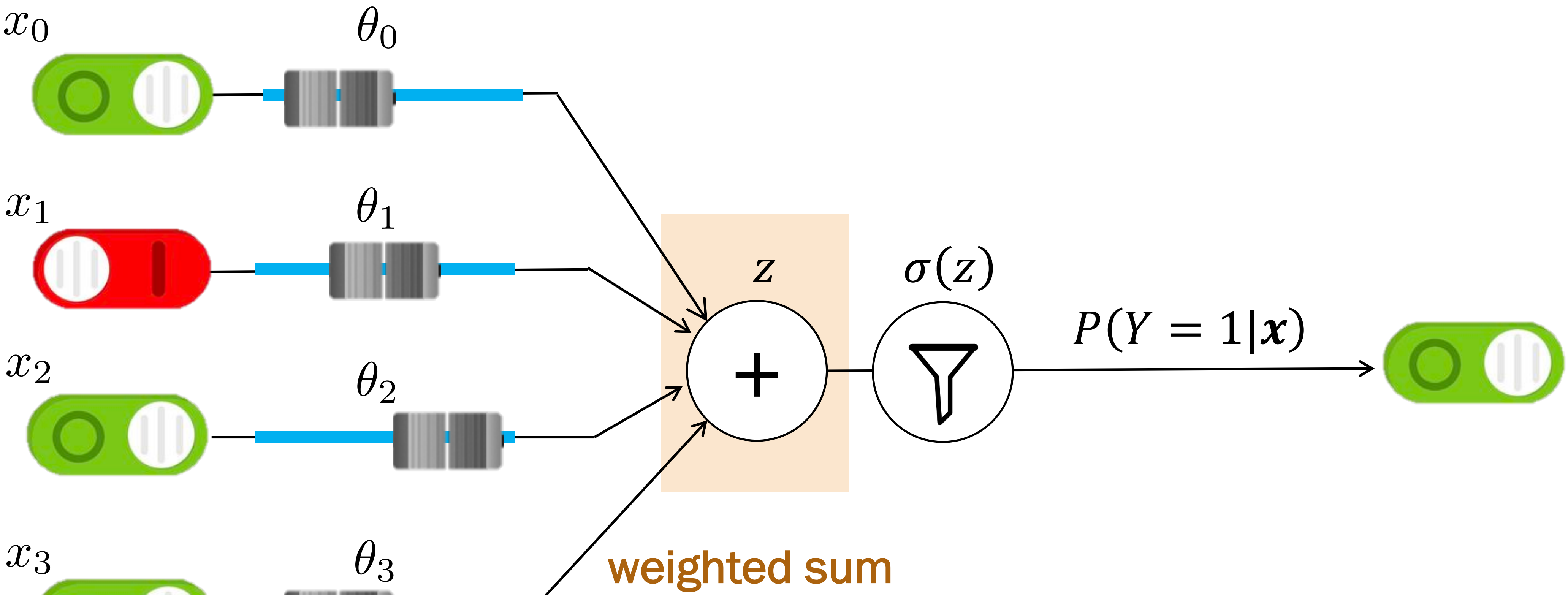
Components of Logistic Regression



θ weights
(aka parameters)

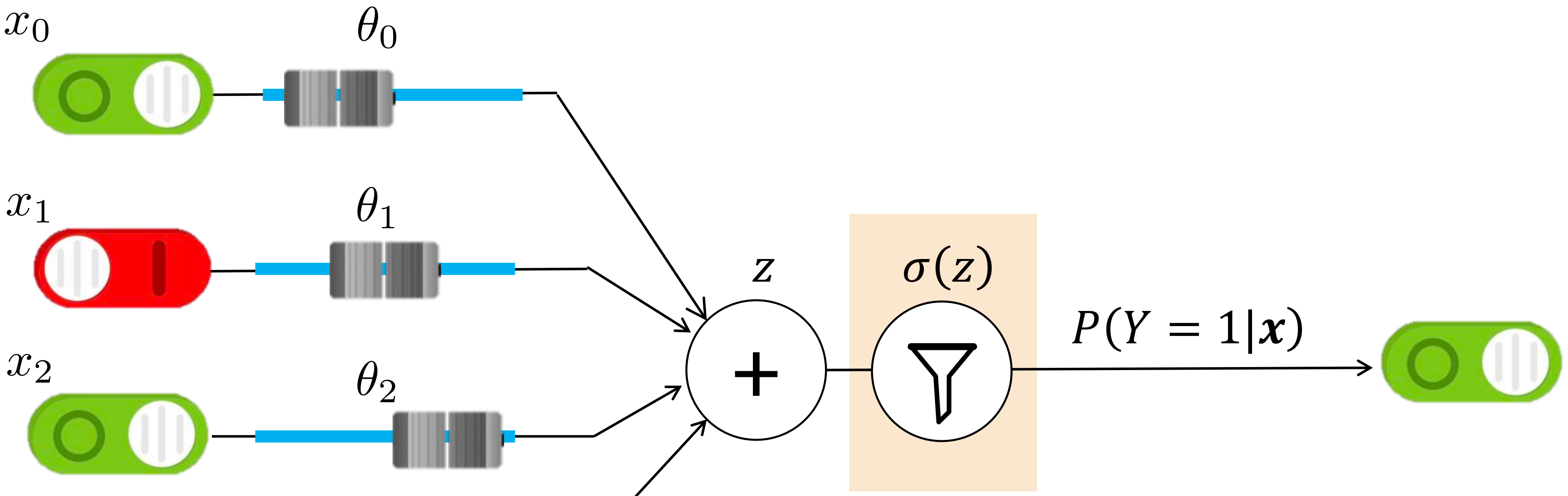
$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

Components of Logistic Regression



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

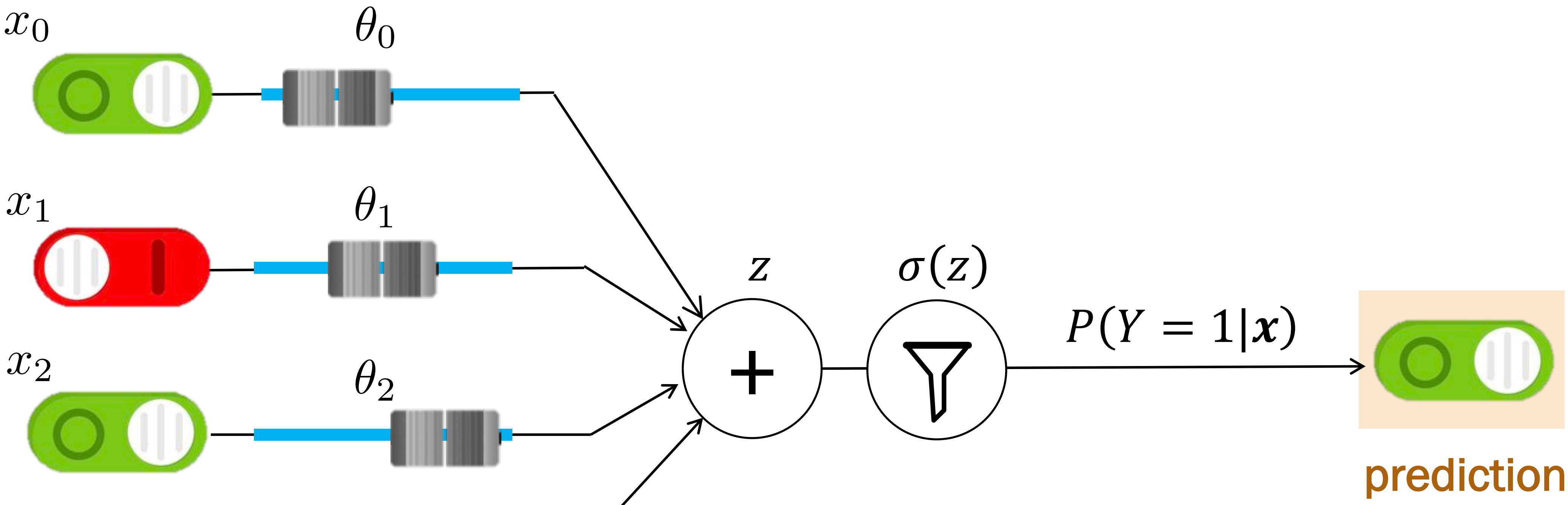
Components of Logistic Regression



squashing function
b/t 0 and 1

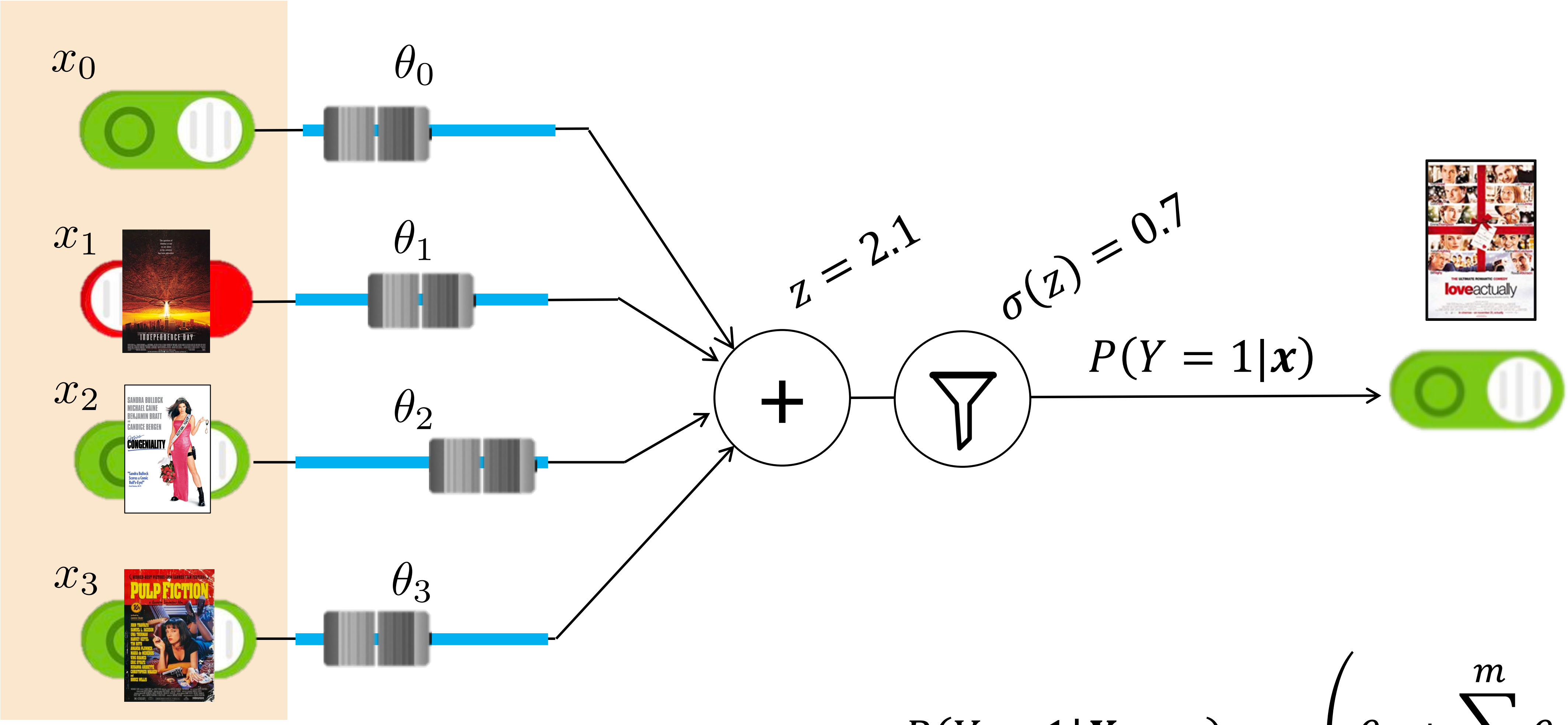
$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

Components of Logistic Regression



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

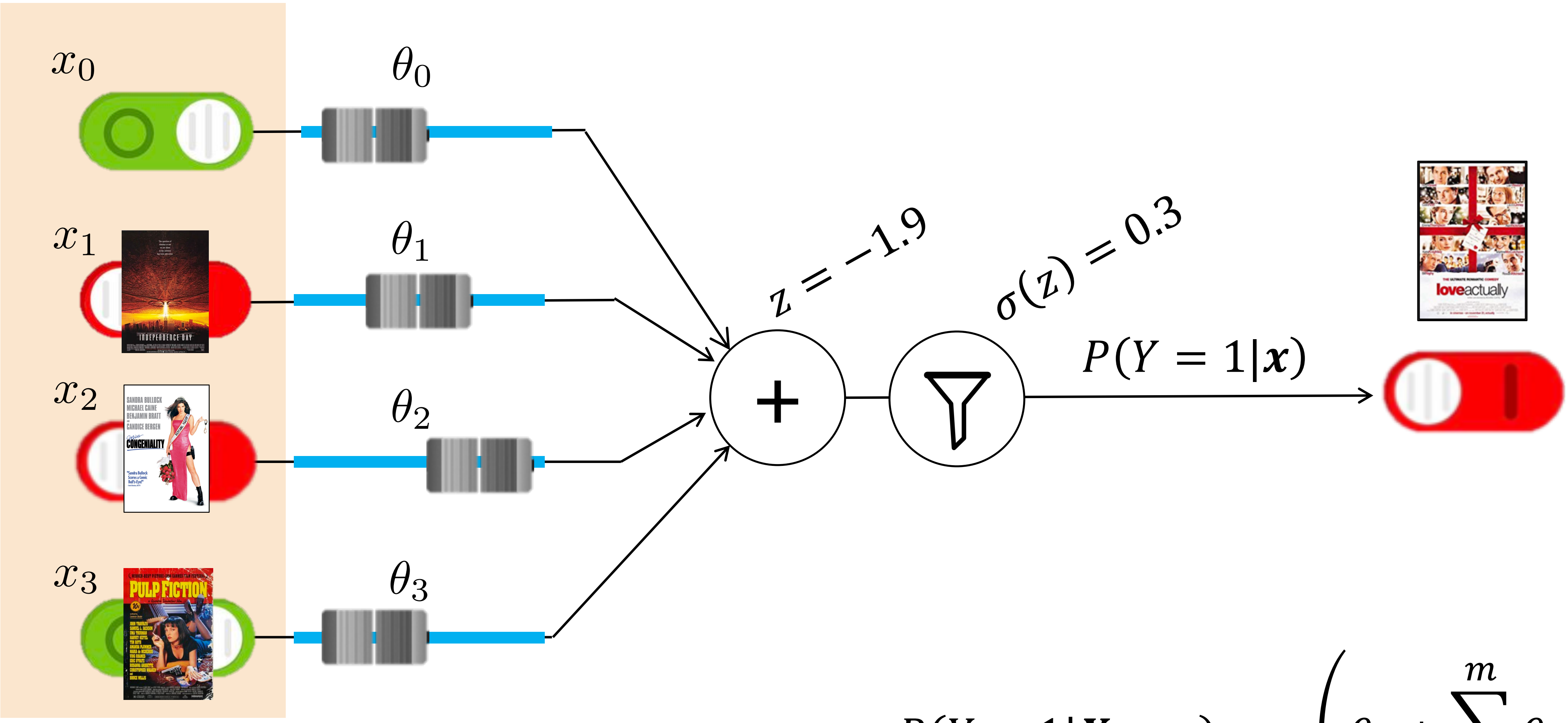
Different predictions for different inputs



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

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

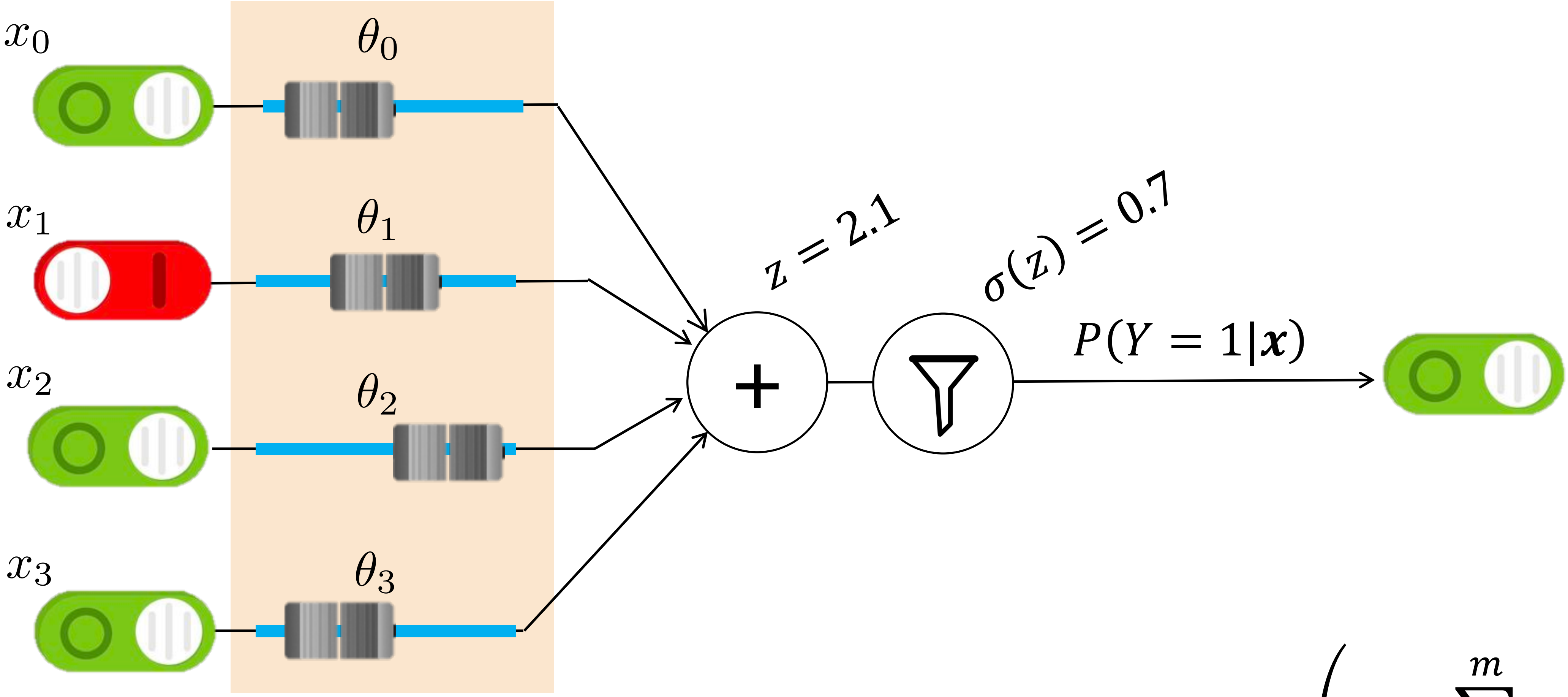
Different predictions for different inputs



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

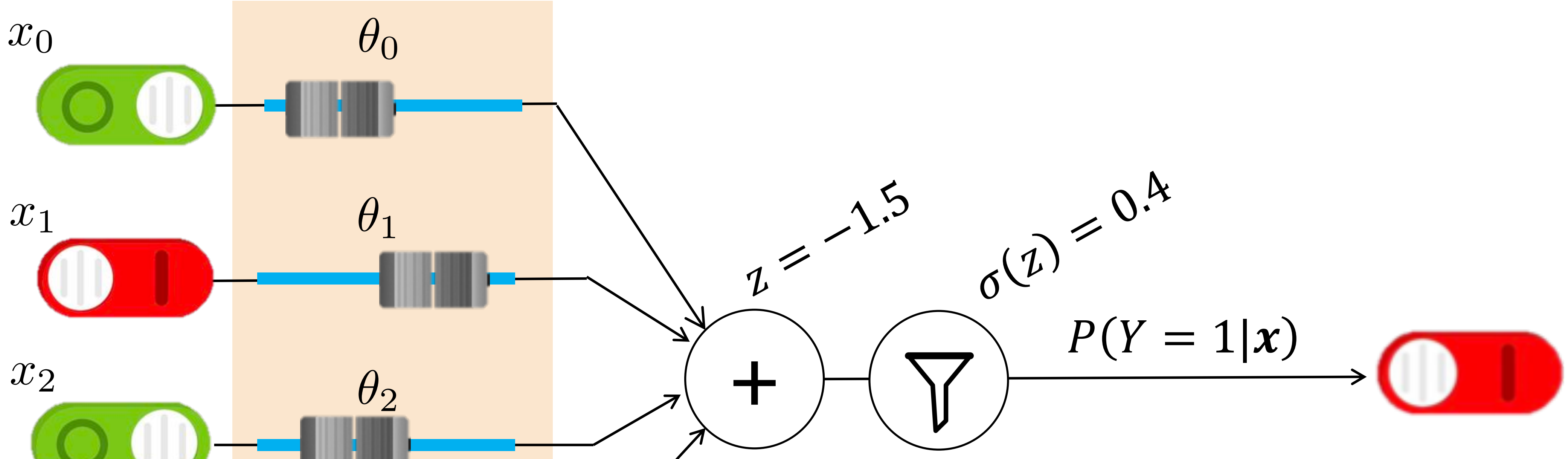
$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\theta_0 + \sum_{j=1}^{m} \theta_j x_j\right)$$

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

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

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

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

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

Training: The big picture

Logistic regression classifier

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

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

Training

Estimate parameters from training data

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

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

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

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

First: Interpret conditional likelihood with Logistic Regression

Second: Write a differentiable expression for log conditional likelihood

1. Determine objective function (interpret)

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

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

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)

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

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

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)

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

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

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

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

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

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

Recall Bernoulli MLE!

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

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

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:

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

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

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)

26: Logistic Regression

Lisa Yan and Jerry Cain

November 11, 2020

Logistic Regression Model

Review

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

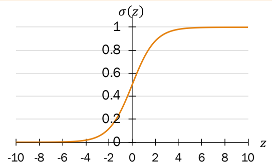
$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \sigma\left(\sum_{j=0}^{m} \theta_j x_j\right) = \sigma(\theta^T \boldsymbol{x})$$

where $x_0 = 1$

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

sigmoid function

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



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. $\hat{Y} \in \{0,1\}$

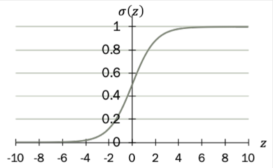
where $x_0 = 1$

$\boldsymbol{X}$

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

sigmoid function

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



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

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

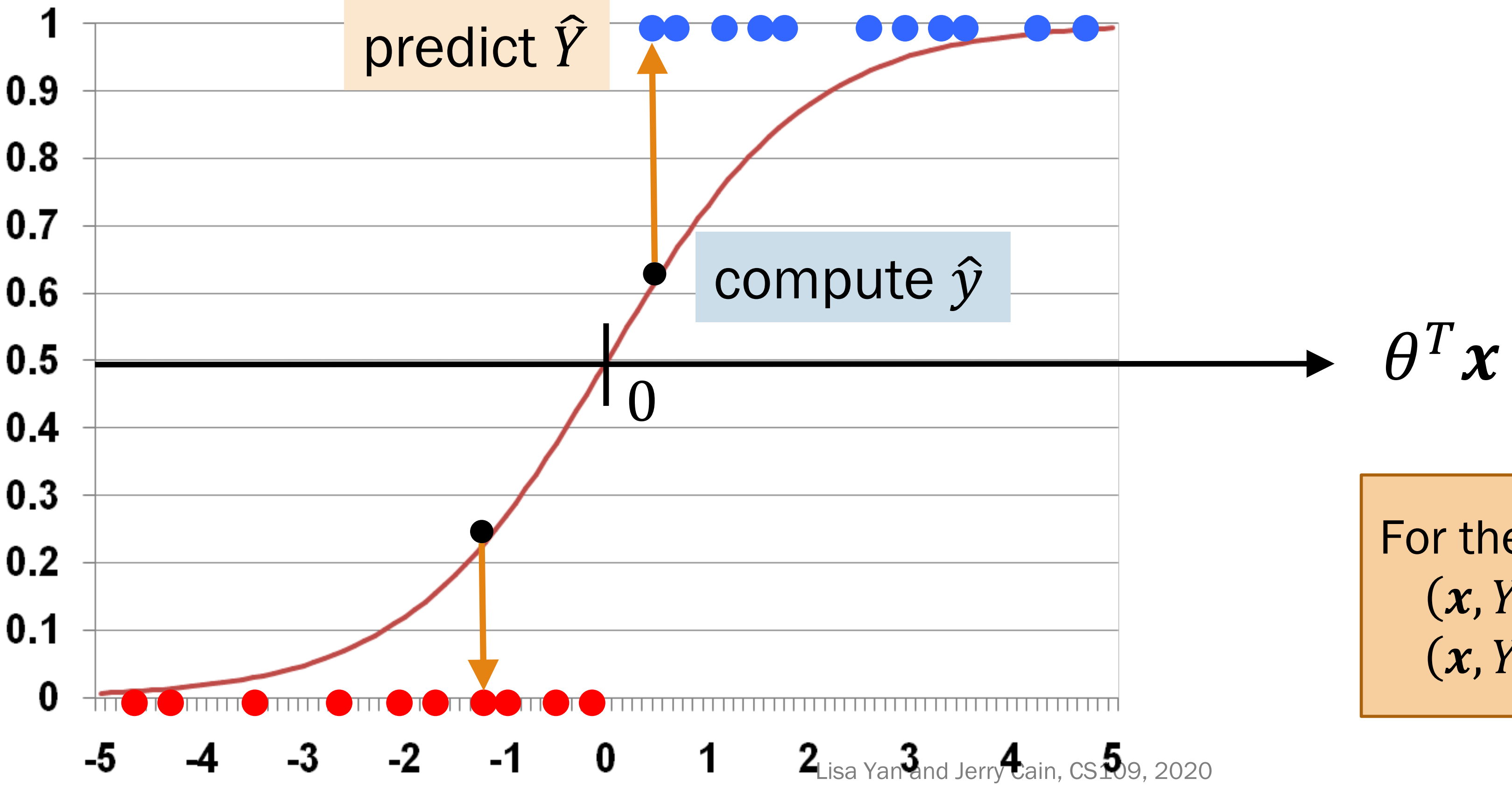
$$P(Y = y|\boldsymbol{X} = \boldsymbol{x}) = \begin{cases} \hat{y} & \text{if } y = 1 \\ 1 - \hat{y} & \text{if } y = 0 \end{cases}$$

Small $\hat{y}$ is conditional probability of $Y = 1$ given $\boldsymbol{X} = \boldsymbol{x}$. $\hat{y} \in [0,1]$

Another view of Logistic Regression

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

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



For the "correct" parameters θ:
$(\boldsymbol{x}, Y = 1)$ should have $\theta^T \boldsymbol{x} > 0$
$(\boldsymbol{x}, Y = 0)$ should have $\theta^T \boldsymbol{x} \leq 0$

Today’s goals: Logistic Regresison

✅ At a high level

- Understand the model
- Training: Use gradient ascent

Details

- Gradient ascent pseudocode
- Testing

For the problem set

Philosophy

- Logistic Regression vs Naïve Bayes
- Linearly separable functions

Derivation of gradient (Calculus)

Machine learning insights

Training: The details

Training: Learning parameters

Review

Training

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

that maximize log conditional likelihood of training data

Some reminders:

- Log conditional likelihood: $$LL(\theta) = \sum_{i=1}^{n} y^{(i)} \log \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right)$$
- Gradient with respect to θ: $$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right)\right] x_j^{(i)} \qquad \text{for } j = 0, 1, \dots, m$$ (derived at end of lecture)
- No analytical solution; optimize with gradient ascent

Training: Gradient ascent step

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

```
repeat many times:

    for all thetas:
            $\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \frac{\partial LL(\theta^{\text{old}})}{\partial \theta_j^{\text{old}}}$
                $= \theta_j^{\text{old}} + \eta \cdot \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left({\theta^{\text{old}}}^T \boldsymbol{x}^{(i)}\right)\right] x_j^{(i)}$
```

What does this look like in code?

Think

Slide 50 has code to think over by yourself.

Post any clarifications here or in chat!

https://us.edstem.org/courses/2678/discussion/171556

Think by yourself: 2 min

🤔 (by yourself)

Training: Gradient Ascent

Gradient Ascent Step

for $j = 0, 1, \dots, m$:

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

```
initialize $\theta_j$ = 0 for 0 ≤ j ≤ m
repeat many times:
    gradient[j] = 0 for 0 ≤ j ≤ m
    // TODO: your code here

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

    $\theta_j$ += η  * gradient[j] for all 0 ≤ j ≤ m
```

(by yourself)

Training: Gradient Ascent

inner loop

for $j = 0, 1, \ldots, m$:

Gradient Ascent Step

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

outer loop

```
initialize $\theta_j$ = 0 for 0 ≤ j ≤ m
repeat many times:
    gradient[j] = 0 for 0 ≤ j ≤ m
    for each training example $(\boldsymbol{x}, y)$:
        for each 0 ≤ j ≤ m:
            // update gradient[j] for
            // current $(\boldsymbol{x}, y)$ example
    $\theta_j$ += η  ∗ gradient[j] for all 0 ≤ j ≤ m
```

Training: Gradient Ascent

inner loop

for $j = 0, 1, \dots, m$:

Gradient Ascent Step

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

outer loop

```
initialize $\theta_j$ = 0 for 0 ≤ j ≤ m
repeat many times:
    gradient[j] = 0 for 0 ≤ j ≤ m
    for each training example ($\boldsymbol{x}$,$y$):
        for each 0 ≤ j ≤ m:
            gradient[j] += $\left[y - \frac{1}{1 + e^{-\theta^T \boldsymbol{x}}}\right] x_j$
    $\theta_j$ += η * gradient[j] for all 0 ≤ j ≤ m
```

Some 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 $\theta_j$ = 0 for 0 ≤ j ≤ m
repeat many times:
    gradient[j] = 0 for 0 ≤ j ≤ m
    for each training example $(\boldsymbol{x}, y)$:
        for each 0 ≤ j ≤ m:
            gradient[j] += $\left[ y - \frac{1}{1 + e^{-\theta^T \boldsymbol{x}}} \right] x_j$
    $\theta_j$ += η * gradient[j] for all 0 ≤ j ≤ m
```

- Finish computing gradient with θ^{old} prior to any θ update

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 $\theta_j$ = 0 for 0 ≤ j ≤ m
repeat many times:
    gradient[j] = 0 for 0 ≤ j ≤ m
    for each training example ($\boldsymbol{x}$,$y$):
        for each 0 ≤ j ≤ m:
            gradient[j] += $\left[y - \frac{1}{1 + e^{-\theta^T \boldsymbol{x}}}\right] x_j$
    $\theta_j$ += $\boxed{\eta}$ * gradient[j] for all 0 ≤ j ≤ m
```

- Finish computing gradient with θ^{old} prior to any θ update
- 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 $\theta_j$ = 0 for 0 ≤ j ≤ m
repeat many times:
    gradient[j] = 0 for 0 ≤ j ≤ m
    for each training example $(\boldsymbol{x}, y)$:
        for each 0 ≤ j ≤ m:
            gradient[j] += $\left[ y - \frac{1}{1 + e^{-\theta^T \boldsymbol{x}}} \right] \boxed{x_j}$
    $\theta_j$ += η * gradient[j] for all 0 ≤ j ≤ m
```

- Finish computing gradient with θ^{old} prior to any θ update
- Learning rate η is a constant you set before training
- 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 $\theta_j$ = 0 for 0 ≤ j ≤ m
repeat many times:
    gradient[j] = 0 for 0 ≤ j ≤ m
    for each training example ($\boldsymbol{x}, y$):
        for each 0 ≤ j ≤ m:
            gradient[j] += $\left[ y - \frac{1}{1 + \boxed{e^{-\theta^T \boldsymbol{x}}}} \right] x_j$
    $\theta_j$ += η * gradient[j] for all 0 ≤ j ≤ m
```

- Finish computing gradient with θ^{old} prior to any θ update
- Learning rate η is a constant you set before training
- x_j is j-th feature of input $\boldsymbol{x} = (x_1, \ldots, x_m)$
- 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 $\theta_j$ = 0 for 0 ≤ j ≤ m
repeat many times:
    gradient[j] = 0 for 0 ≤ j ≤ m
    for each training example ($\boldsymbol{x}$,$y$):
        for each 0 ≤ j ≤ m:
            gradient[j] += $\left[y - \frac{1}{1 + e^{-\theta^T \boldsymbol{x}}}\right] x_j$
    $\theta_j$ += η * gradient[j] for all 0 ≤ j ≤ m
```

- Finish computing gradient with θ^{old} prior to any θ update
- Learning rate η is a constant you set before training
- x_j is j-th feature of input $\boldsymbol{x} = (x_1, \dots, x_m)$
- Insert $x_0 = 1$ before training

Testing

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

https://www.bagelbakerygainesville.com/top-8-bagel-jokes-of-all-time/

Announcements

Quiz #3

Time frame:	Wednesday 11/18 2:00pm – Friday 11/20 12:59pm PT
Covers:	Up to and including logistic regression
Info and practice:	Quizzes page

Next week: Last section

Review session for Quiz #3

Probability Reference (Overleaf)

Updated to include all of Quiz 3-relevant material (sampling defs, MLE/MAP, classifiers)

Interesting probability news

The Time Everyone "Corrected" the World's Smartest Woman

https://priceonomics.com/the-time-everyone-corrected-the-worlds-smartest/

Today's goals: Logistic Regression

✅ At a high level

- Understand the model
- Training: Use gradient ascent

Details

✅
- Gradient ascent pseudocode
- Testing

For the problem set

Philosophy

- Logistic Regression vs Naïve Bayes
- Linearly separable functions

Derivation of gradient (Calculus)

Machine learning insights

Philosophy

Think

Slide 64 asks you to think over by yourself.

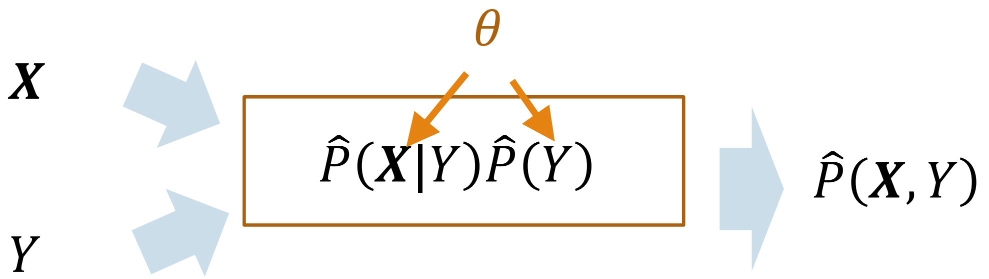
Post any clarifications here or in chat!

https://us.edstem.org/courses/2678/discussion/171556

Think by yourself: 2 min

(by yourself)

Naïve Bayes vs Logistic Regression



$\boldsymbol{X}$

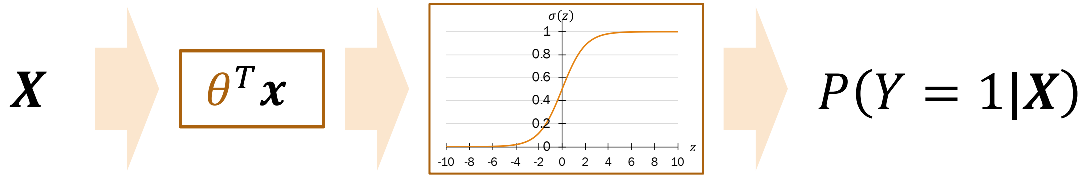
Y

θ

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

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

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



$\boldsymbol{X}$

$\theta^T \boldsymbol{x}$

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

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

Compare/contrast:

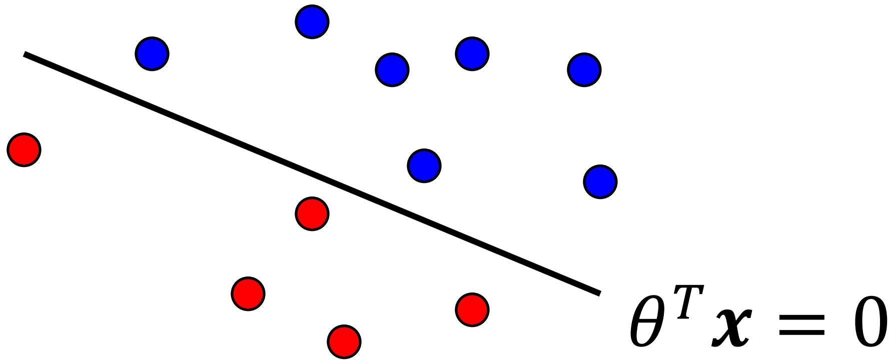
1. What **distributions** are we modeling?
2. After learning our parameters, could we randomly **generate** a new datapoint $(\boldsymbol{x}, y)$?
3. Could we model a **continuous** X_j feature (e.g., $X_j \sim$Normal, or $X_j \sim$Unknown)?
4. Could we model a non-binary **discrete** X_j (e.g., $X_j \in \{1,2,\ldots,6\}$)?

(by yourself)

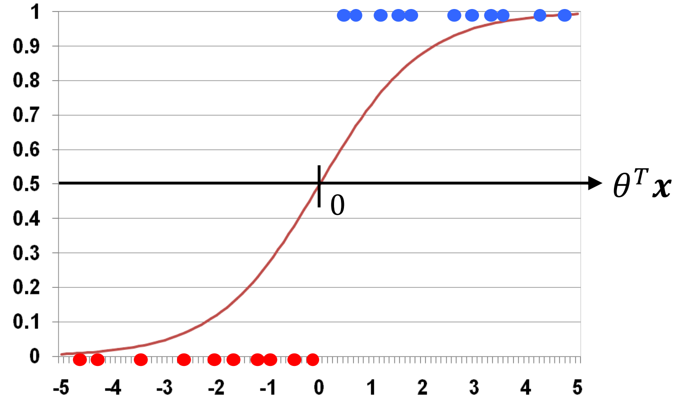
Tradeoffs:	Naïve Bayes	Logistic Regression
1. Modeling goal	$P(\boldsymbol{X}, Y)$	$P(Y \mid \boldsymbol{X})$
2. Generative or discriminative?	**Generative**: could use joint distribution to generate new points (⚠ but you might not need this extra effort)	**Discriminative**: just tries to discriminate $y = 0$ vs $y = 1$ (❌ cannot generate new points b/c no $P(\boldsymbol{X}, Y)$)
3. Continuous input features	⚠ Needs parametric form (e.g., Gaussian) or discretized buckets (for multinomial features)	✅ Yes, easily
4. 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 $\{1, 2, 3\}$

Linearly separable data

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



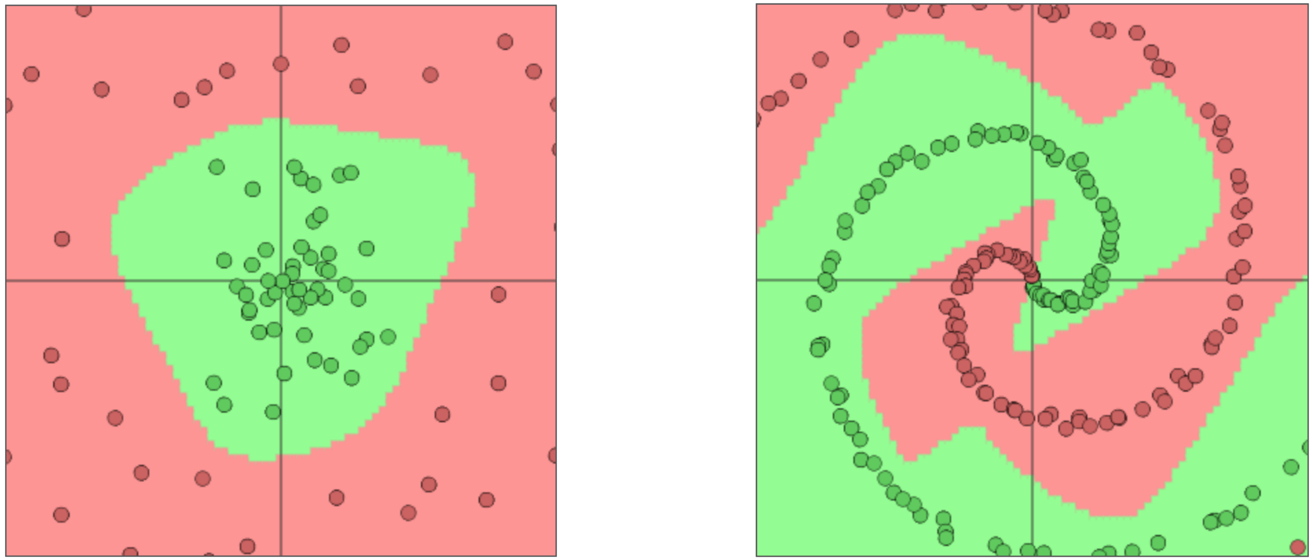
- We call such data (or functions generating the data) **linearly separable.**



- Naïve Bayes is linear too, because there is one parameter for each feature (and no parameters that involve multiple features).

$$\hat{P}(\boldsymbol{X}|Y) = \prod_{j=1}^{m} \hat{P}(X_j|Y)$$

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

LIVE

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 🌟 🌟 🌟

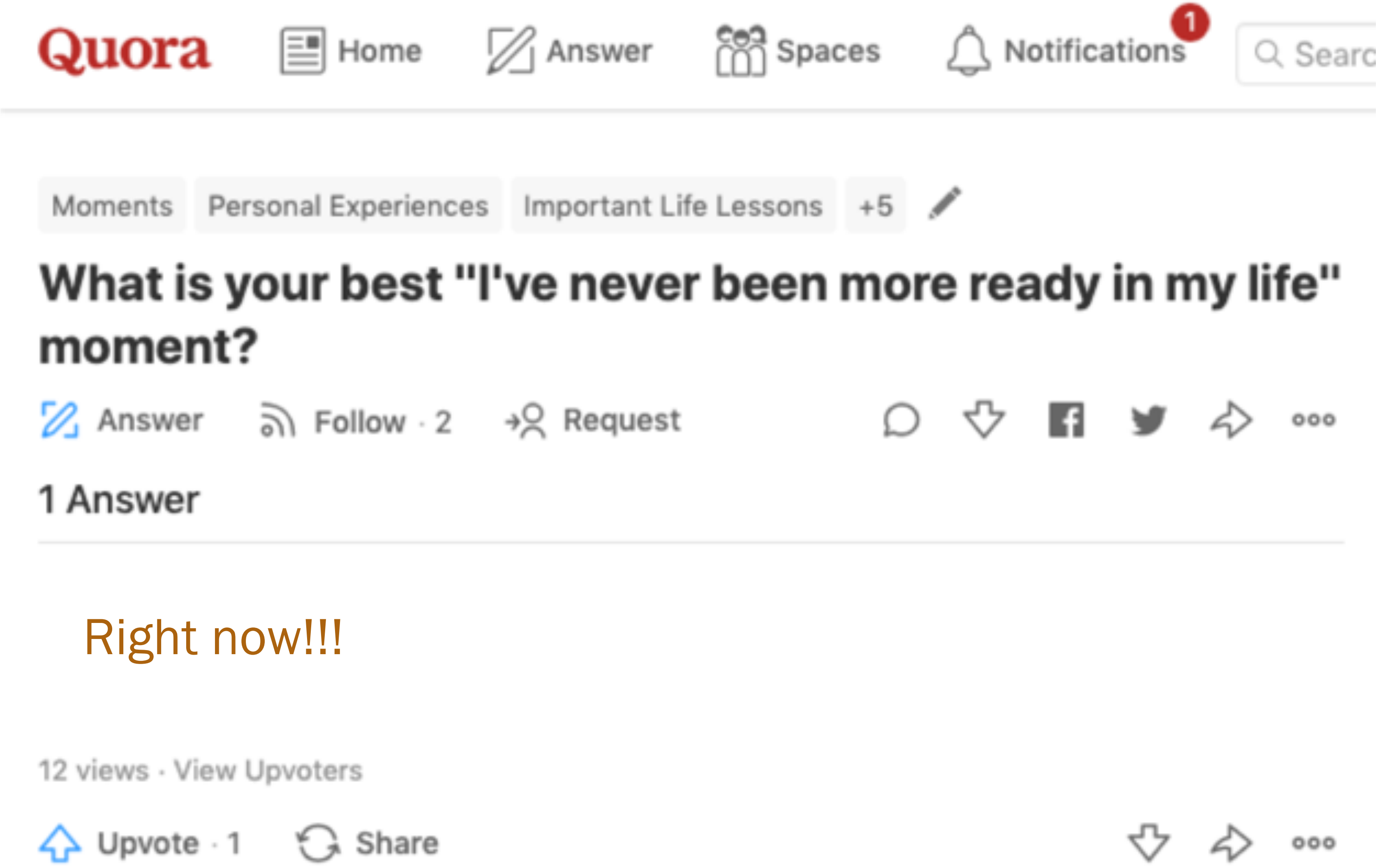
$$\frac{\partial f(x)}{\partial x} = \frac{\partial f(z)}{\partial z}\frac{\partial z}{\partial x}$$

Calculus Chain Rule

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

aka decomposition
of composed functions

Are you ready?



What is your best "I've never been more ready in my life" moment?

1 Answer

Right now!!!

12 views · View Upvoters

Our goal

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

Two "pre-processing" steps to prepare for chain rule

1. Rewrite $LL(\theta)$ with $\hat{y}$
2. Compute gradient of $\hat{y}$

1. Rewriting $LL(\theta)$ with $\hat{y}$

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

$$\downarrow$$

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

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

2. Compute gradient of $\hat{y} = \sigma(\theta^T \boldsymbol{x})$

Aside: Sigmoid has a beautiful derivative!

Sigmoid function:

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

Derivative:

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

Think

Slide 72 has code to think over by yourself.

Post any in chat!

Think by yourself: 2 min

(by yourself)

2. Compute gradient of $\hat{y} = \sigma(\theta^T \boldsymbol{x})$

Sigmoid function:

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

Derivative:

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

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

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

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

E. None/other

🤔 (by yourself)

2. Compute gradient of $\hat{y} = \sigma(\theta^T \boldsymbol{x})$

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)

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

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)

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)

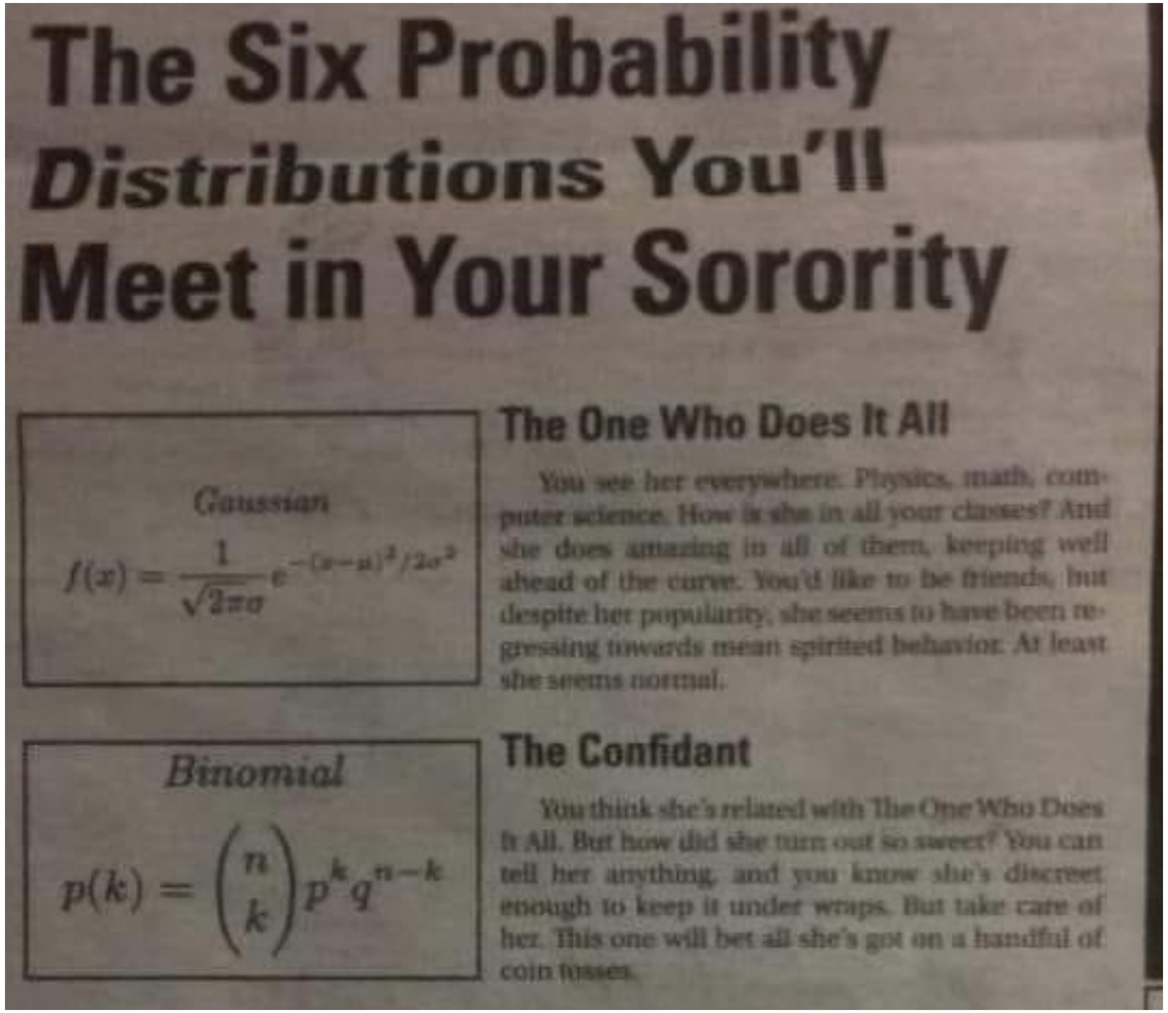
$$= \sum_{i=1}^{n} \left[y^{(i)} - \hat{y}^{(i)} \right] x_j^{(i)} \qquad = \sum_{i=1}^{n} \left[y^{(i)} - \sigma\left(\theta^T \boldsymbol{x}^{(i)}\right) \right] x_j^{(i)}$$

🎉

(simplify)

Interlude for jokes

Probability as college students



The Six Probability Distributions You'll Meet in Your Sorority

Gaussian

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(x-\mu)^2/2\sigma^2}$$

The One Who Does It All

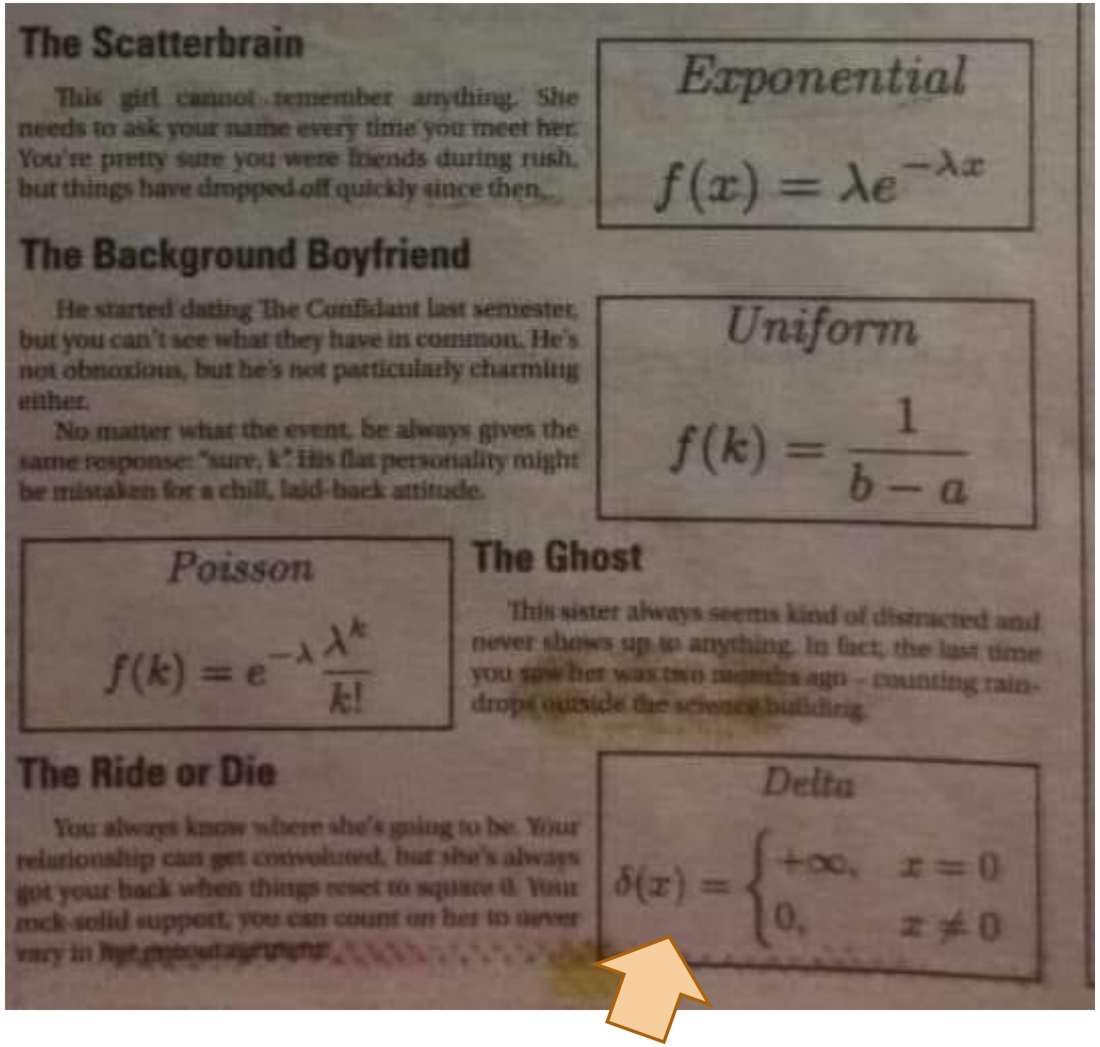
You see her everywhere: Physics, math, computer science. How is she in all your classes? And she does amazing in all of them, keeping well ahead of the curve. You'd like to be friends, but despite her popularity, she seems to have been regressing towards mean spirited behavior. At least she seems normal.

Binomial

$$p(k) = \binom{n}{k} p^k q^{n-k}$$

The Confidant

You think she's related with The One Who Does It All. But how did she turn out so sweet? You can tell her anything, and you know she's discreet enough to keep it under wraps. But take care of her. This one will bet all she's got on a handful of coin tosses.



The Scatterbrain

This girl cannot remember anything. She needs to ask your name every time you meet her. You're pretty sure you were friends during rush, but things have dropped off quickly since then.

Exponential

$$f(x) = \lambda e^{-\lambda x}$$

The Background Boyfriend

He started dating The Confidant last semester, but you can't see what they have in common. He's not obnoxious, but he's not particularly charming either.

No matter what the event, he always gives the same response: "sure, k." His flat personality might be mistaken for a chill, laid-back attitude.

Uniform

$$f(k) = \frac{1}{b-a}$$

Poisson

$$f(k) = e^{-\lambda} \frac{\lambda^k}{k!}$$

The Ghost

This sister always seems kind of distracted and never shows up to anything. In fact, the last time you saw her was two months ago – counting raindrops outside the science building.

The Ride or Die

You always know where she's going to be. Your relationship can get convoluted, but she's always got your back when things reset to square 0. Your rock-solid support, you can count on her to never vary in her [illegible]

Delta

$$\delta(x) = \begin{cases} +\infty, & x = 0 \\ 0, & x \neq 0 \end{cases}$$

(A useful construct that connects discrete PMF to continuous PDF)