24: Naïve Bayes

Lisa Yan and Jerry Cain

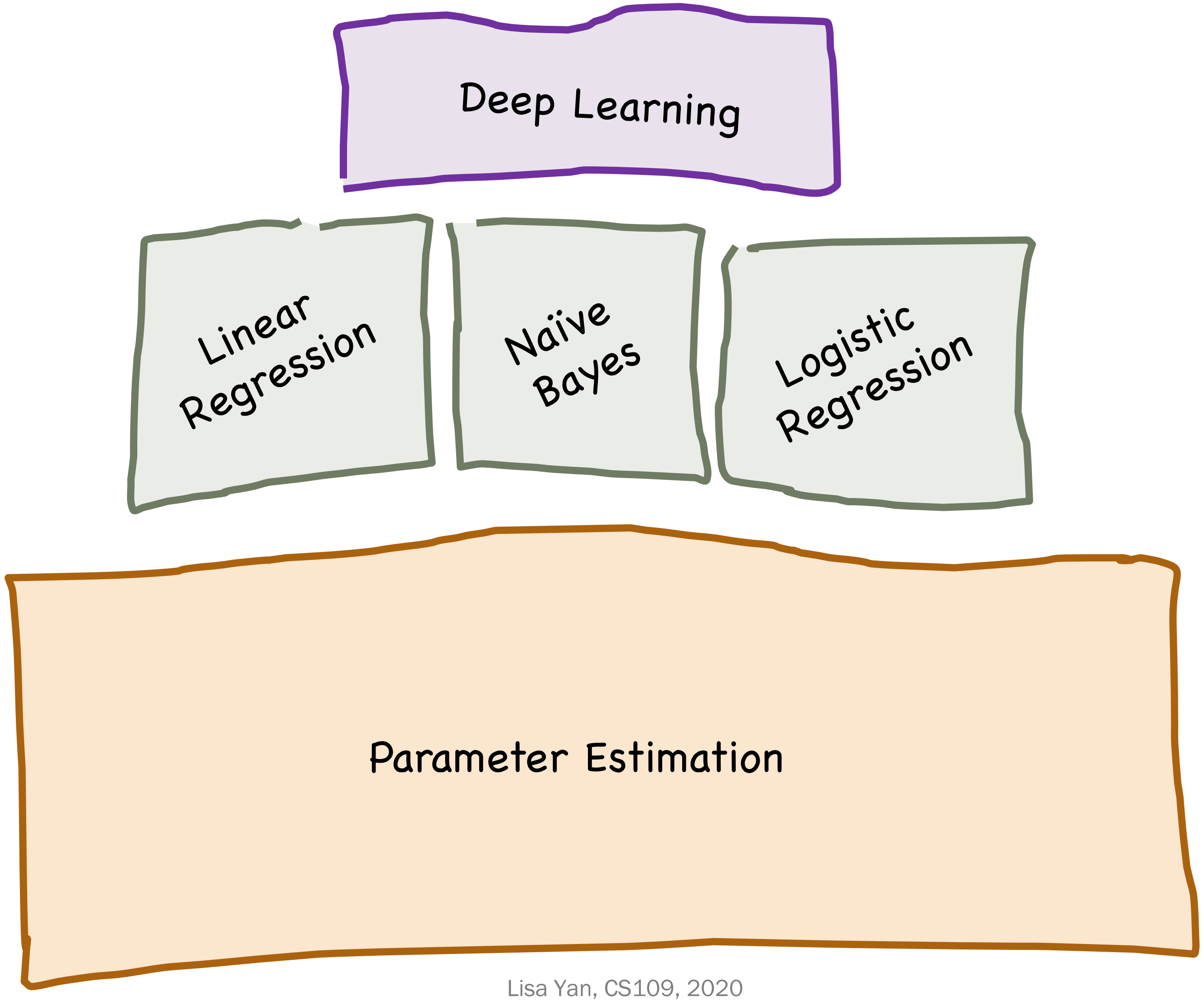
November 6, 2020

Quick slide reference

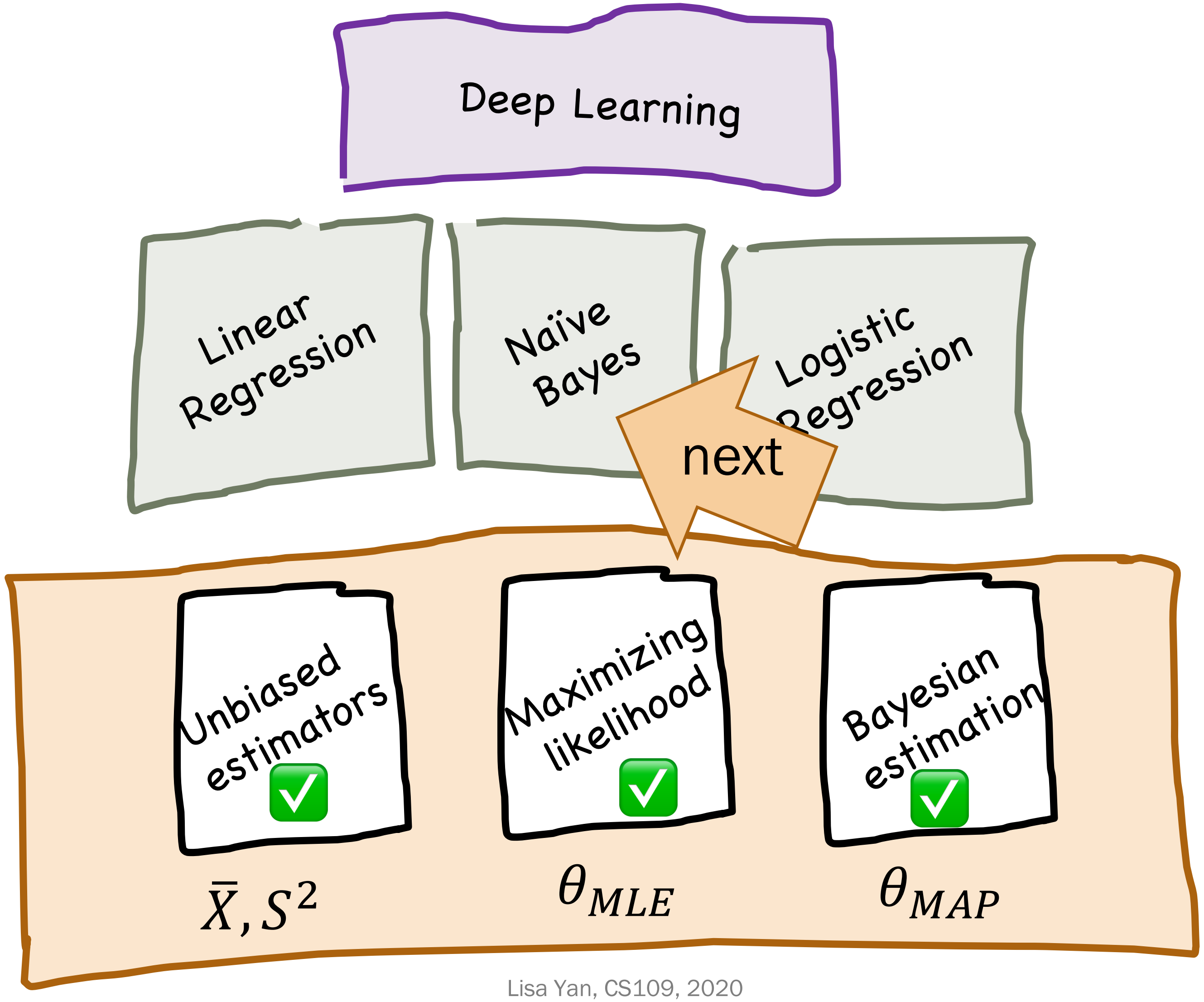
3	Intro: Machine Learning	24a_intro
21	“Brute Force Bayes”	24b_brute_force_bayes
32	Naïve Bayes Classifier	24c_naive_bayes
43	Naïve Bayes: MLE/MAP with TV shows	LIVE
71	Naïve Bayes: MAP with email classification	LIVE

Intro: Machine Learning

Our path from here



Our path from here



Machine Learning (formally)

Many different forms of "Machine Learning"

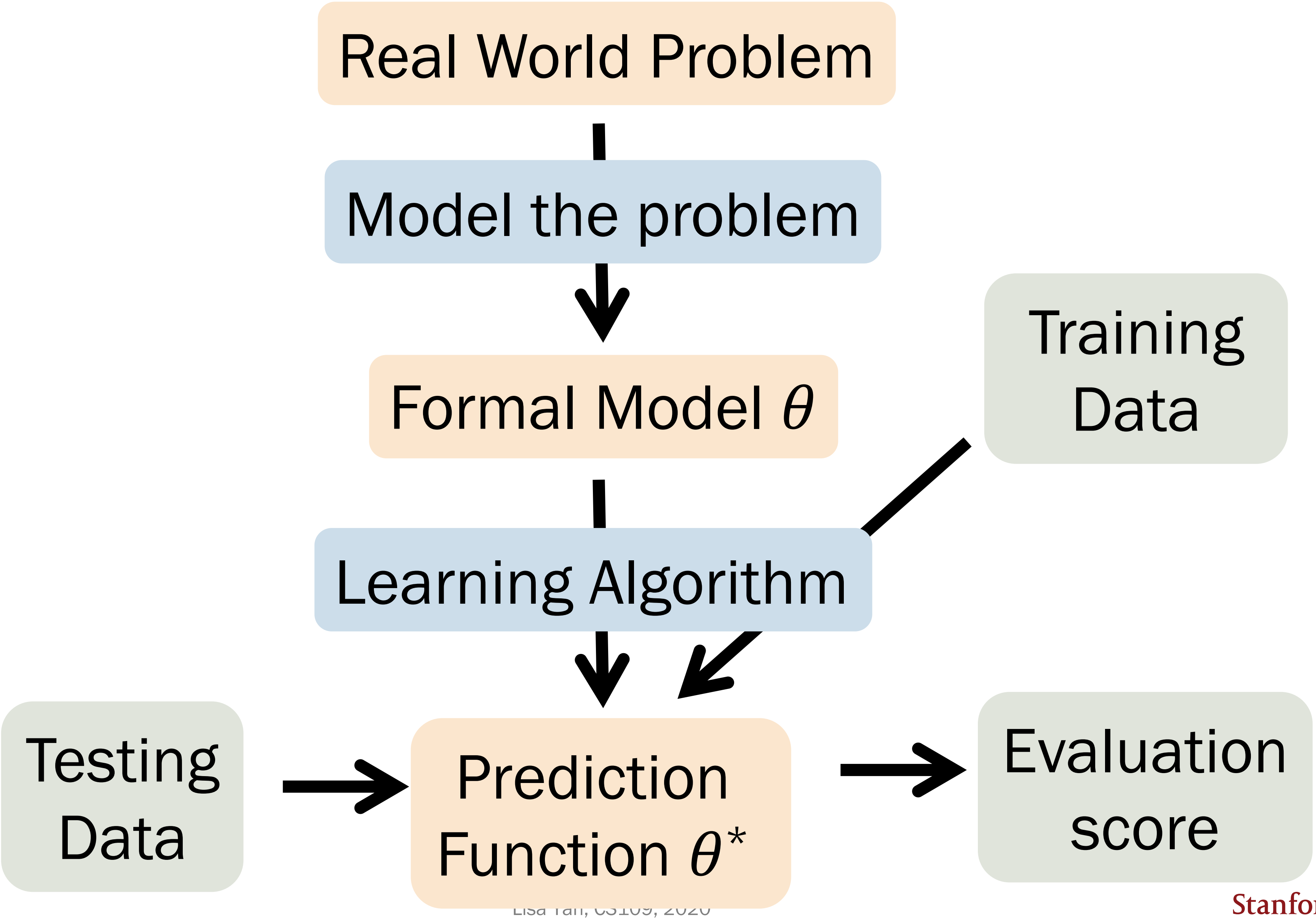
- We focus on the problem of prediction based on observations.

Machine Learning uses a lot of data.

Supervised learning: A category of machine learning where you have labeled data on the problem you are solving.

Task: Identify what a chair is
Data: All the chairs ever

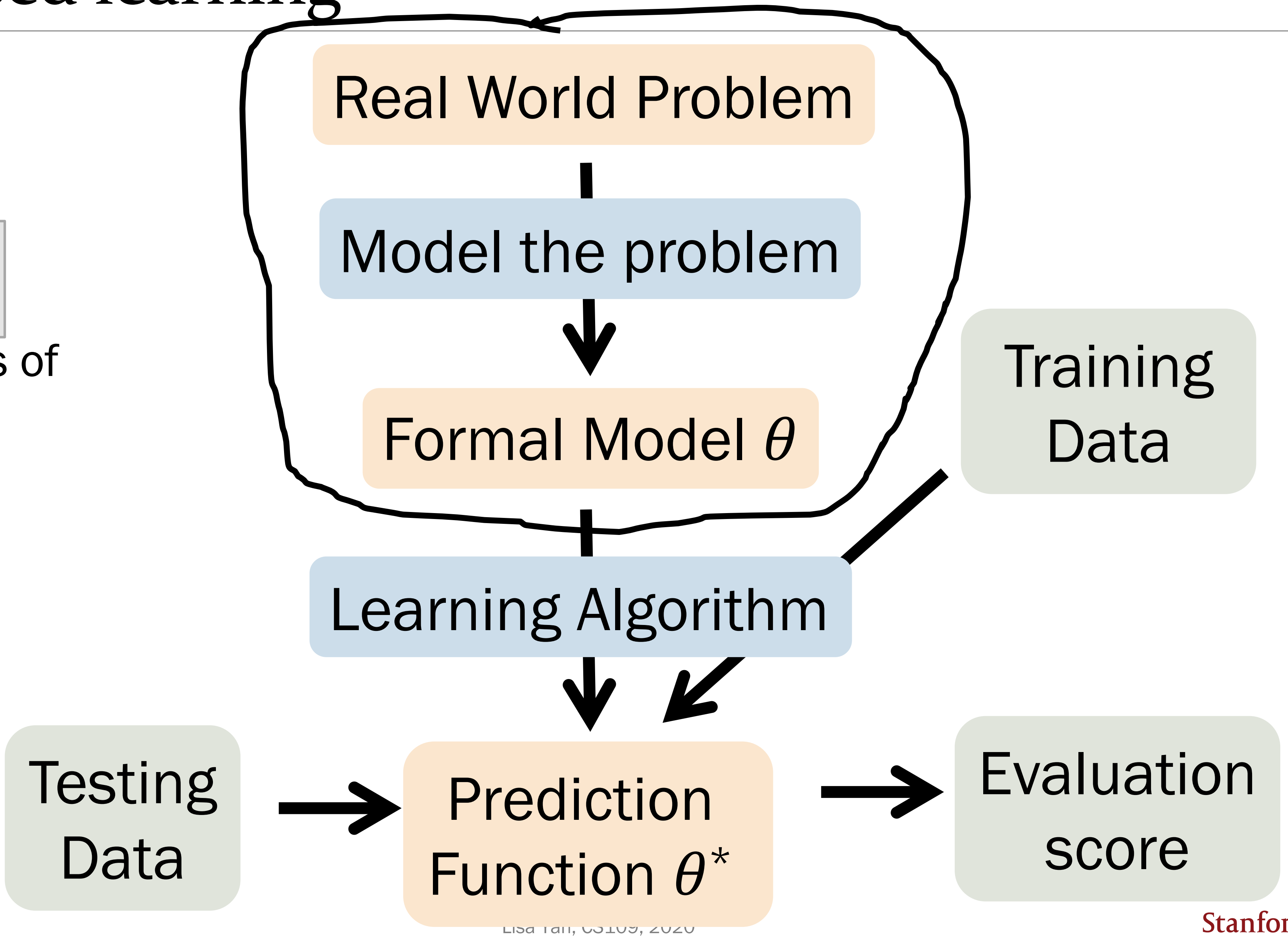
Supervised learning



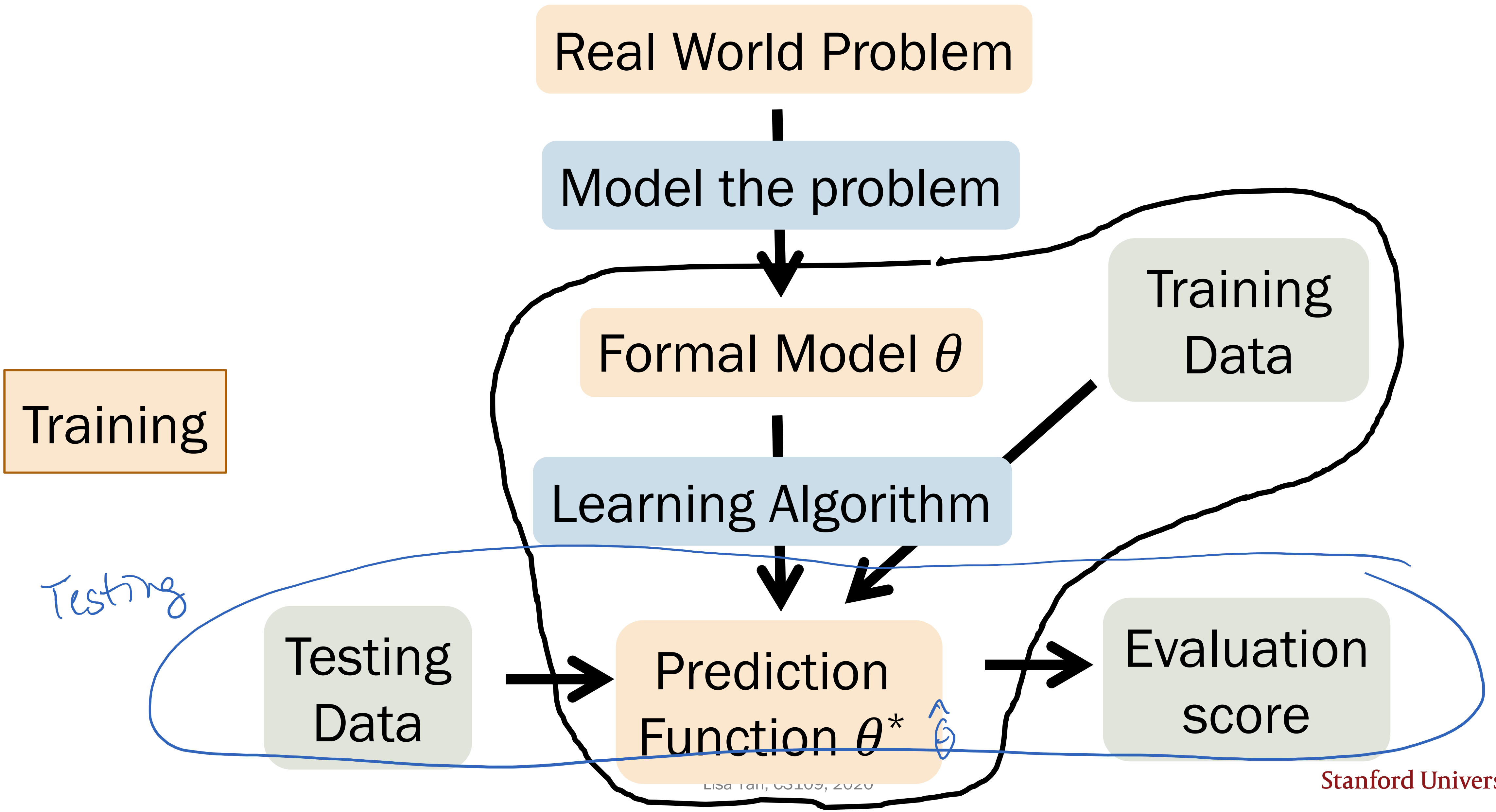
Supervised learning

Modeling

(not the focus of this class)



Supervised learning



Model and dataset

Many different forms of "Machine Learning"

- We focus on the problem of **prediction** based on observations.

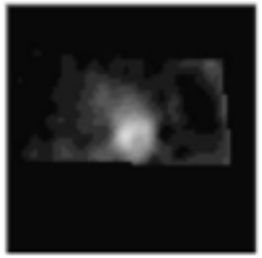
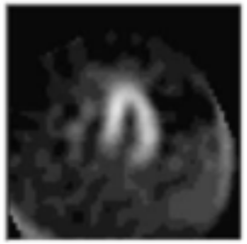
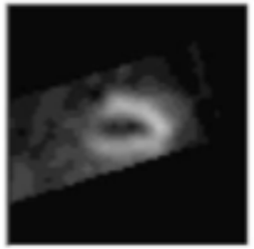
Goal Based on observed $\boldsymbol{X}$, predict unseen Y

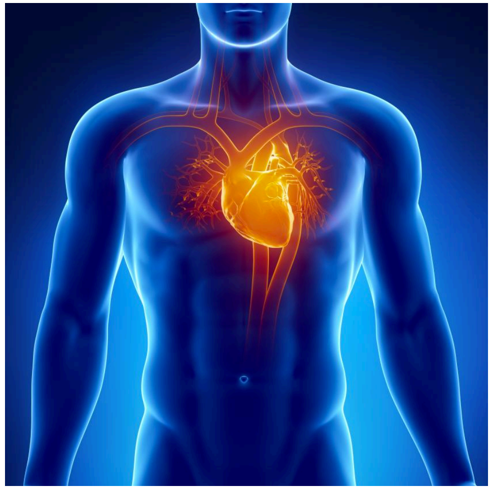
- **Features** Vector $\boldsymbol{X}$ of m observed variables $\boldsymbol{X} = (X_1, X_2, \ldots, X_m)$
- **Output** Variable Y (also called **class label** if discrete)

Model $\hat{Y} = g(\boldsymbol{X})$, a function of observations $\boldsymbol{X}$

Training data

$$\boldsymbol{X} = (X_1, X_2, X_3, \ldots, X_{300})$$

  … 



	Feature 1	Feature 2		Feature 300	Output
	binary vector				
Patient 1	1	0	…	1	1
Patient 2	1	1	…	0	0
…			⋮		⋮
Patient n	0	0	…	1	1

Training data notation

$$\left(\boldsymbol{x}^{(1)}, y^{(1)}\right), \left(\boldsymbol{x}^{(2)}, y^{(2)}\right), \dots, \left(\boldsymbol{x}^{(n)}, y^{(n)}\right)$$

n datapoints, generated i.i.d.

i–th datapoint $\left(\overbrace{\boldsymbol{x}^{(i)}}^{m\text{-dimensional observation}}, \overbrace{y^{(i)}}^{1\text{-dimensional output}}\right)$:

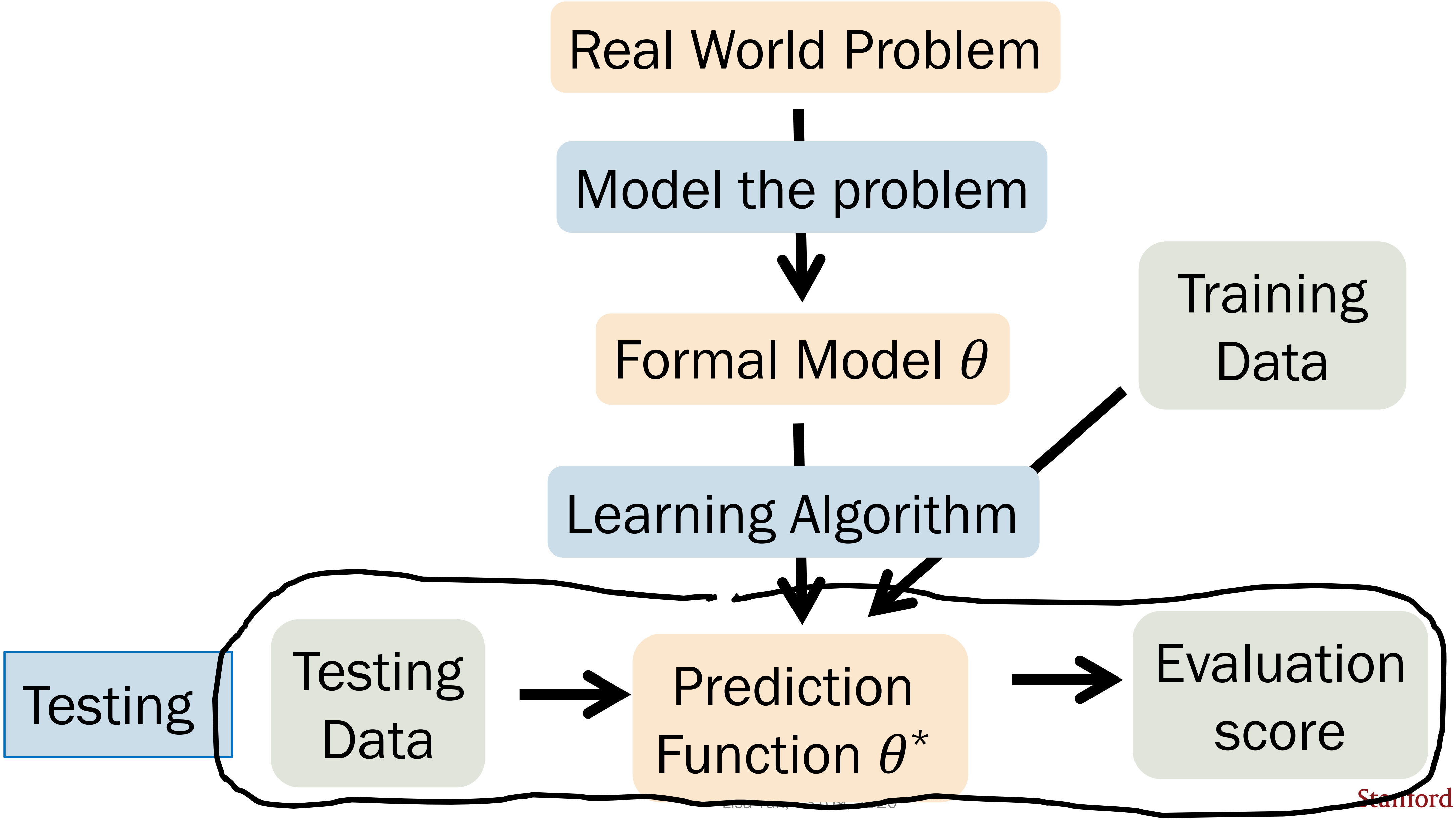
- m features: $\boldsymbol{x}^{(i)} = \left(x_1^{(i)}, x_2^{(i)}, \dots, x_m^{(i)}\right)$

  ith datapoint's feature #2
- A single output $y^{(i)}$
- Independent of all other datapoints

Training Goal:

> Use these n datapoints to learn a model $\hat{Y} = g(\boldsymbol{X})$ that predicts Y

Supervised learning



Testing data notation

$$\left(\boldsymbol{x}^{(1)}, y^{(1)}\right), \left(\boldsymbol{x}^{(2)}, y^{(2)}\right), \ldots, \left(\boldsymbol{x}^{(n)}, y^{(n)}\right)$$

n_{test} <u>other</u> datapoints, generated i.i.d.

i–th datapoint $\left(\boldsymbol{x}^{(i)}, y^{(i)}\right)$:

- Has the same structure as your training data

Testing Goal:

Using the model $\hat{Y} = g(\boldsymbol{X})$ that you trained, see how well you can predict Y on known data

Two prediction tasks

Many different forms of "Machine Learning"

- We focus on the problem of **prediction** based on observations.

Goal — Based on observed $\boldsymbol{X}$, predict unseen Y

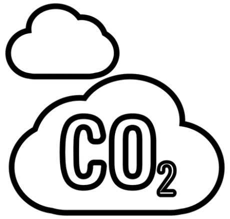
- **Features** — Vector $\boldsymbol{X}$ of m observed variables $\boldsymbol{X} = (X_1, X_2, \ldots, X_m)$
- **Output** — Variable Y (also called class label if discrete)

Model — $\hat{Y} = g(\boldsymbol{X})$, a function of observations $\boldsymbol{X}$

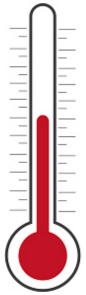
- **Regression** — prediction when Y is continuous
- **Classification** — prediction when Y is discrete

Regression: Predicting real numbers

Training data: $(\boldsymbol{x}^{(1)}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \ldots, (\boldsymbol{x}^{(n)}, y^{(n)})$





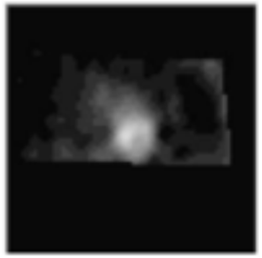
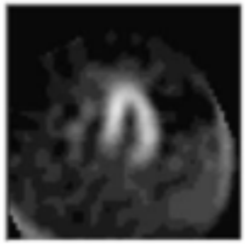
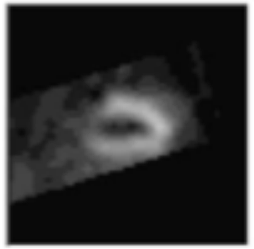


Global Land-Ocean temperature

	CO2 levels	Sea level	…	Feature m	Output
Year 1	338.8	0	…	1	0.26
Year 2	340.0	1	…	0	0.32
…			⋮		⋮
Year n	340.76	0	…	1	0.14

Classification: Predicting class labels

$$\boldsymbol{X} = (X_1, X_2, X_3, \ldots, X_{300})$$



…


	Feature 1	Feature 2		Feature 300	Output
Patient 1	1	0	…	1	1
Patient 2	1	1	…	0	0
…			⋮		⋮
Patient n	0	0	…	1	1

Classification: Harry Potter Sorting Hat

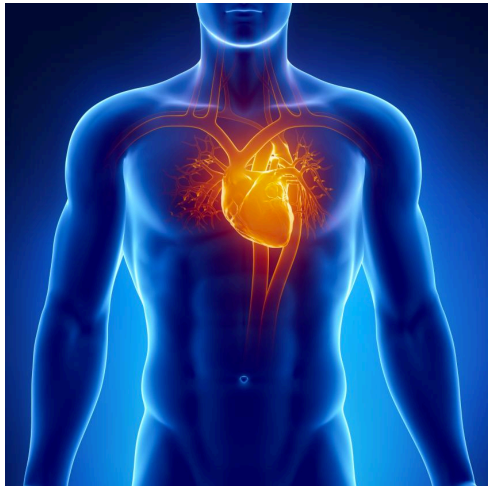
$\hat{Y} = 1$

$\boldsymbol{X} = (1, 1, 1, 0, 0, \ldots, 1)$

Our focus today!

Classification: Example datasets

Heart



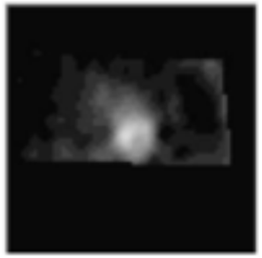
Ancestry

Netflix

“Brute Force Bayes”

Classification: Having a healthy heart

$\boldsymbol{X} = (X_1)$ "feature vector" = observation



	Feature 1	Output
Patient 1	1	0
Patient 2	1	1
	⋮	⋮
Patient n	0	1

Single feature: Region of Interest (ROI) is healthy (1) or unhealthy (0)

How can we predict the class label heart is healthy (1) or unhealthy (0)?

The following strategy is **not used in practice** but helps us understand how we approach classification.

Classification: "Brute Force Bayes"

Y: Fact

X: Evidence/

Observation

$\hat{Y} = g(\boldsymbol{X})$

Our prediction for Y is a function of $\boldsymbol{X}$

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

Proposed model: Choose the Y that is most likely given $\boldsymbol{X}$

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

(Bayes' Theorem)

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

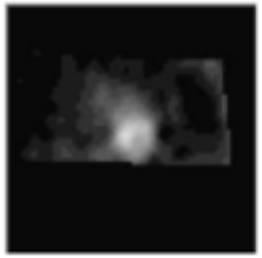
($1/P(\boldsymbol{X})$ is constant w.r.t. y)

If we estimate $P(\boldsymbol{X}|Y)$ and $P(Y)$, we can classify datapoints!

Training: Estimate parameters

$\boldsymbol{X} = (X_1)$

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



	Feature 1	Output
Patient 1	1	0
Patient 2	1	1
	⋮	⋮
Patient n	0	1

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

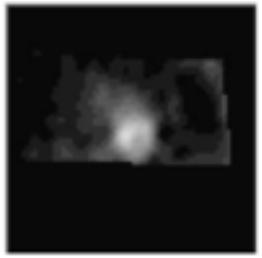
	$\hat{P}(\boldsymbol{X}\|Y=0)$	$\hat{P}(\boldsymbol{X}\|Y=1)$
$X_1 = 0$	θ_1	θ_3
$X_1 = 1$	θ_2	θ_4

Marginal probability table $\hat{P}(Y)$

	$\hat{P}(Y)$
$Y = 0$	θ_5
$Y = 1$	θ_6

Training Goal: Use n datapoints to learn $2 \cdot 2 + 2 = 6$ parameters.

Training: Estimate parameters $\hat{P}(\boldsymbol{X}|Y)$



Feature 1

Output

	Feature 1	Output
Patient 1	1	0
Patient 2	1	1
	⋮	⋮
Patient n	0	1

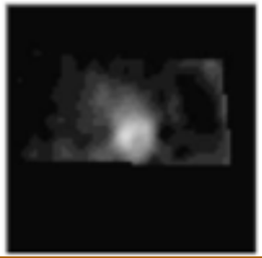
	$\hat{P}(\boldsymbol{X}\|Y=0)$	$\hat{P}(\boldsymbol{X}\|Y=1)$
$X_1 = 0$	θ_1	θ_3
$X_1 = 1$	θ_2 $=1-\theta_1$	θ_4 $=1-\theta_3$

$\boldsymbol{X}|Y = 0$ and $\boldsymbol{X}|Y = 1$
are each multinomials with 2 outcomes!

Use MLE or Laplace (MAP) estimate for parameters $\hat{P}(\boldsymbol{X}|Y)$ and $\hat{P}(Y)$

Training: MLE estimates, $\hat{P}(\boldsymbol{X}|Y)$

"impossible



Count:	# datapoints
X_1 = 0, Y = 0:	4
X_1 = 1, Y = 0:	6
X_1 = 0, Y = 1:	0
X_1 = 1, Y = 1:	100
Total:	110

Pa

Pa

Patient n 0 | 1

MLE

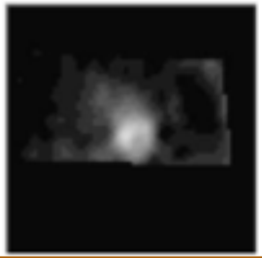
	$\hat{P}(\boldsymbol{X}\|Y=0)$	$\hat{P}(\boldsymbol{X}\|Y=1)$
$X_1=0$	$0.4 = \frac{4}{10}$	$0.0 = 0/100$
$X_1=1$	$0.6 = 6/10$	$1.0 = 100/100$

MLE of $\hat{P}(X_1 = x|Y=y) = \dfrac{\#(X_1 = x, Y = y)}{\#(Y=y)}$

Just count!

Training: Laplace (MAP) estimates, $\hat{P}(\boldsymbol{X}|Y)$

	$\hat{P}(\boldsymbol{X}\|Y=0)$	$\hat{P}(\boldsymbol{X}\|Y=1)$
$X_1 = 0$	0.4	0.0
$X_1 = 1$	0.6	1.0



Count: <u># datapoints</u>

X_1 = 0, Y = 0: 4 +1

X_1 = 1, Y = 0: 6 +1

X_1 = 0, Y = 1: 0 +1

X_1 = 1, Y = 1: 100 +1

Total: 110

Pa

Pa

Patient n 0 1

MLE

MLE of $\hat{P}(X_1 = x|Y = y) = \dfrac{\#(X_1 = x, Y = y)}{\#(Y = y)}$

Just count!

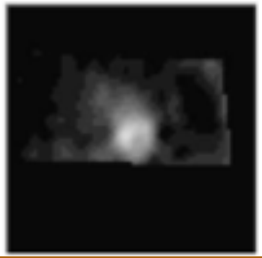
MAP

$$\hat{P}(X_1 = 1 | Y = 0) = \frac{\#(X_1 = 1, Y = 0) + 1}{\#(Y = 0) + 2}$$

Laplace of $\hat{P}(X_1 = x|Y = y) = \ ?$

Just count + add imaginary trials!

Training: Laplace (MAP) estimates, $\hat{P}(\boldsymbol{X}|Y)$



Count:	# datapoints
X_1 = 0, Y = 0:	4
X_1 = 1, Y = 0:	6
X_1 = 0, Y = 1:	0
X_1 = 1, Y = 1:	100
Total:	110

Pa

Pa

Patient n 0 1

MLE

	$\hat{P}(\boldsymbol{X}\|Y=0)$	$\hat{P}(\boldsymbol{X}\|Y=1)$
$X_1 = 0$	0.4	0.0 = 0/100
$X_1 = 1$	0.6	1.0

MLE of $\hat{P}(X_1 = x|Y = y) = \dfrac{\#(X_1 = x, Y = y)}{\#(Y = y)}$

Just count!

MAP

	$\hat{P}(\boldsymbol{X}\|Y=0)$	$\hat{P}(\boldsymbol{X}\|Y=1)$
$X_1 = 0$	0.42 = 5/12	0.01 = 1/102
$X_1 = 1$	0.58 = 7/12	0.99 = 101/102

Laplace of $\hat{P}(X_1 = x|Y = y) = \dfrac{\#(X_1 = x, Y = y) + 1}{\#(Y = y) + 2}$

Just count + add imaginary trials!

Testing

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

Laplace estimates

(MAP)	$\hat{P}(\mathbf{X}\|Y=0)$	$\hat{P}(\mathbf{X}\|Y=1)$
$X_1 = 0$	0.42	0.01
$X_1 = 1$	0.58	0.99

(MLE)	$\hat{P}(Y)$
$Y = 0$	0.09 $= \frac{10}{110}$
$Y = 1$	0.91 $= \frac{100}{110}$

New patient has a healthy ROI ($X_1 = 1$). What is your prediction, $\hat{Y}$?

$$\hat{P}(X_1 = 1|Y = 0)\hat{P}(Y = 0) = 0.58 \cdot 0.09 \approx 0.052$$
$$\hat{P}(X_1 = 1|Y = 1)\hat{P}(Y = 1) = 0.99 \cdot 0.91 \approx 0.901$$

A. $0.052 < 0.5 \quad \Rightarrow \quad \hat{Y} = 1$
B. $0.901 > 0.5 \quad \Rightarrow \quad \hat{Y} = 1$
C. $0.052 < 0.901 \Rightarrow \quad \hat{Y} = 1$

Sanity check: Why don't these sum to 1?

Testing

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

(MAP)	$\hat{P}(\boldsymbol{X}\|Y=0)$	$\hat{P}(\boldsymbol{X}\|Y=1)$
$X_1 = 0$	0.42	0.01
$X_1 = 1$	0.58	0.99

(MLE)	$\hat{P}(Y)$
$Y = 0$	0.09
$Y = 1$	0.91

New patient has a healthy ROI ($X_1 = 1$). What is your prediction, $\hat{Y}$?

$$\hat{P}(X_1 = 1|Y = 0)\hat{P}(Y = 0) = 0.58 \cdot 0.09 \approx 0.052 \leftarrow \hat{P}(X_1=1, Y=0)$$
$$\hat{P}(X_1 = 1|Y = 1)\hat{P}(Y = 1) = 0.99 \cdot 0.91 \approx 0.901 \leftarrow \hat{P}(X_1=1, Y=1)$$

A. $0.052 < 0.5 \Rightarrow \hat{Y} = 1$
B. $0.901 > 0.5 \Rightarrow \hat{Y} = 1$
C. $0.052 < 0.901 \Rightarrow \hat{Y} = 1$

Sanity check: Why don't these sum to 1?

"Brute Force Bayes" classifier

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

($\hat{P}(Y)$ is an estimate of $P(Y)$,
$\hat{P}(\boldsymbol{X}|Y)$ is an estimate of $P(\boldsymbol{X}|Y)$)

Training

Estimate these probabilities, i.e., "learn" these parameters using MLE or Laplace (MAP)

$\hat{P}(X_1, X_2, \ldots, X_m|Y = 1)$
$\hat{P}(X_1, X_2, \ldots, X_m|Y = 0)$
$\hat{P}(Y = 1)$ $\hat{P}(Y = 0)$

Testing

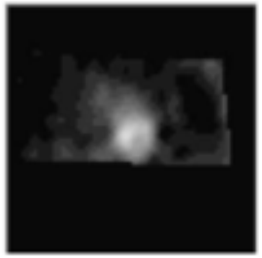
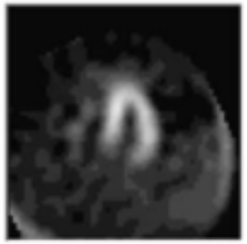
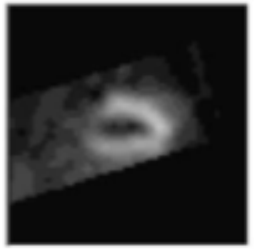
Given an observation $\boldsymbol{X} = (X_1, X_2, \ldots, X_m)$, predict

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\hat{P}(X_1, X_2, \ldots, X_m|Y)\hat{P}(Y)\right)$$

Naïve Bayes Classifier

Brute Force Bayes: $m = 300$ (# features)

$$\boldsymbol{X} = (X_1, X_2, X_3, \dots, X_{300})$$

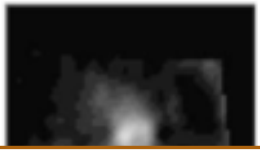
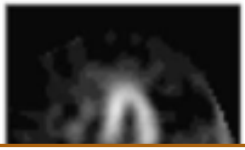
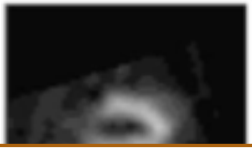


…



	Feature 1	Feature 2		Feature 300	Output
Patient 1	1	0	…	1	1
Patient 2	1	1	…	0	0
…			⋮		⋮
Patient n	0	0	…	1	1

This won't be too bad, right?

Brute Force Bayes: $m = 300$ (# features)

$$\boldsymbol{X} = (X_1, X_2, X_3, \dots, X_{300})$$

Count: # datapoints

$X_1 = 0, X_2 = 0, \dots, X_{299} = 0, X_{300} = 0$, Y = 0:	0
$X_1 = 0, X_2 = 0, \dots, X_{299} = 0, X_{300} = 1$, Y = 0:	0
$X_1 = 0, X_2 = 0, \dots, X_{299} = 1, X_{300} = 0$, Y = 0:	1
…	
$X_1 = 0, X_2 = 0, \dots, X_{299} = 0, X_{300} = 0$, Y = 1:	2
$X_1 = 0, X_2 = 0, \dots, X_{299} = 0, X_{300} = 1$, Y = 1:	1
$X_1 = 0, X_2 = 0, \dots, X_{299} = 1, X_{300} = 0$, Y = 1:	1

Pat

Pat

Patient n 0 0 … 1 1

This won't be too bad, right?

Brute Force Bayes

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

Choose the Y that is most likely given $\boldsymbol{X}$

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

(Bayes' Theorem)

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

($1/P(\boldsymbol{X})$ is constant w.r.t. y)

Learn parameters through MLE or MAP

Brute Force Bayes: $m = 300$ (# features)

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

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

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

Learn parameters through MLE or MAP

- $\hat{P}(Y = 1 \mid \boldsymbol{x})$: estimated probability a heart is healthy given $\boldsymbol{x}$
- $\boldsymbol{X} = (X_1, X_2, \dots, X_{300})$: whether 300 regions of interest (ROI) are healthy (1) or unhealthy (0)

How many parameters do we have to learn?

	$\hat{P}(\boldsymbol{X}\|Y)$	$\hat{P}(Y)$	
A.	$2 \cdot 2$	$+ 2$	$= 6$
B.	$2 \cdot 300$	$+ 2$	$= 602$
C.	$2 \cdot 2^{300}$	$+ 2$	$=$ a lot

Brute Force Bayes: $m = 300$ (# features)

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

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

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

Learn parameters through MLE or MAP

This approach requires you to learn $O(2^m)$ parameters.

- $\hat{P}(Y = 1 \mid \boldsymbol{x})$: estimated probability a heart is healthy given $\boldsymbol{x}$
- $\boldsymbol{X} = (X_1, X_2, \ldots, X_{300})$: whether 300 regions of interest (ROI) are healthy (1) or unhealthy (0)

How many parameters do we have to learn?

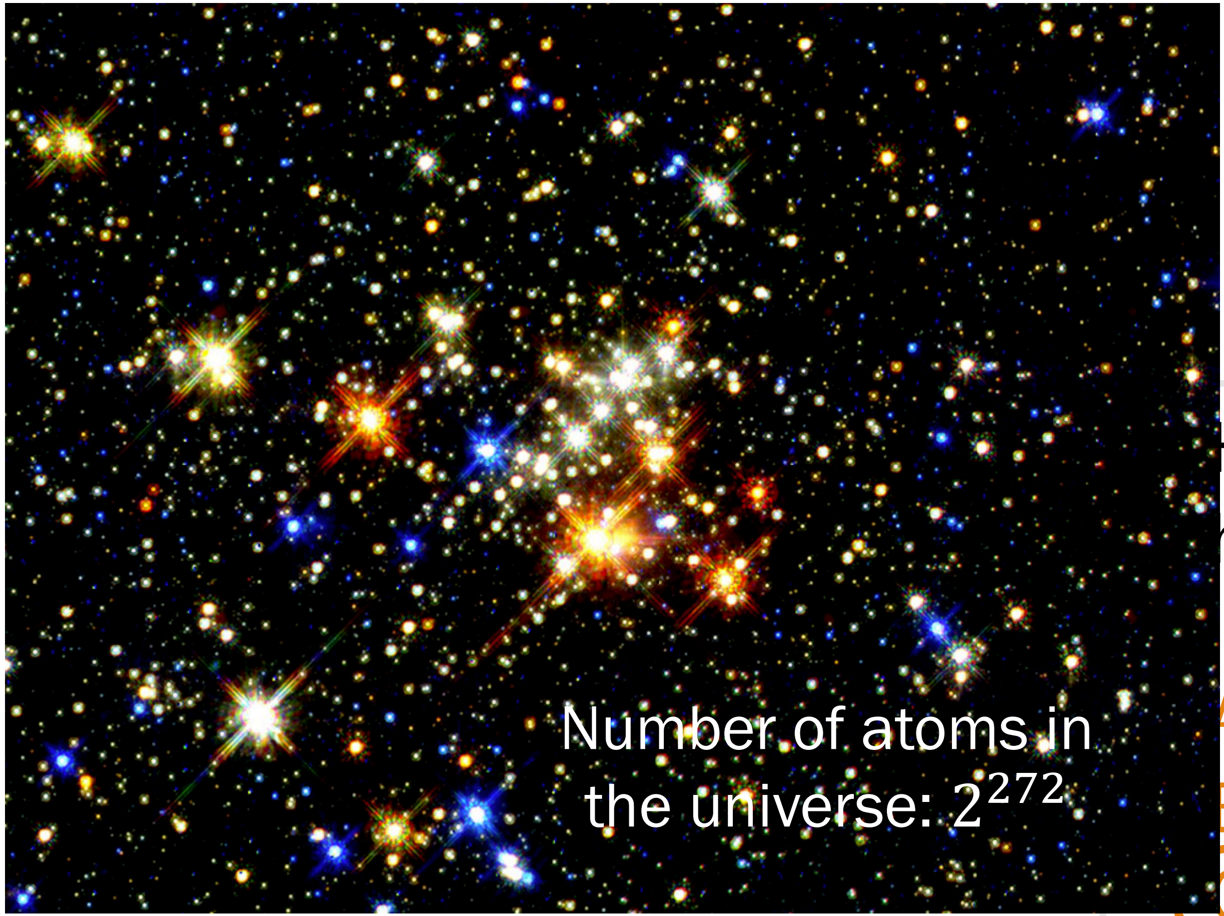
	$\hat{P}(\boldsymbol{X}\|Y)$	$\hat{P}(Y)$	
A.	$2 \cdot 2$	$+\, 2$	$= 6$
B.	$2 \cdot 300$	$+\, 2$	$= 602$
C.	$2 \cdot 2^{300}$	$+\, 2$	$=$ a lot

2^{300}

$\hat{P}(X_1 = x_1, X_2 = x_2, \ldots, X_{300} = x_{300} \mid Y = 0)$ $\hat{P}(Y=1)$

$\hat{P}(X_1 = x_1, X_2 = x_2, \ldots, X_{300} = x_{300} \mid Y = 1)$ $\hat{P}(Y=0)$

Brute Force Bayes: $m = 300$ (# features)



$\hat{P}(Y = 1 \mid \boldsymbol{x})$: estimated probability a heart is healthy given $\boldsymbol{x}$

$\boldsymbol{X} = (X_1, X_2, \ldots, X_{300})$: whether 300 regions of interest (ROI) are healthy (1) or unhealthy (0)

How many parameters do we have to learn?

	$\hat{P}(\boldsymbol{X}\|Y)$	$\hat{P}(Y)$	
A.	$2 \cdot 2$	$+\,2$	$= 6$
B.	$2 \cdot 300$	$+\,2$	$= 602$
C.	$2 \cdot 2^{300}$	$+\,2$	$=$ a lot

This approach requires you to learn $O(2^m)$ parameters.

The problem with our current classifier

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

Choose the Y that is most likely given $\boldsymbol{X}$

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

(Bayes' Theorem)

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

($1/P(\boldsymbol{X})$ is constant w.r.t. y)

$$\hat{P}(X_1, X_2, \dots, X_m|Y)$$

Estimating this joint conditional distribution is often intractable.

What if we could make a simplifying (but naïve) assumption to make estimation easier?

The Naïve Bayes assumption

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

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

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

$$= \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y)\right)\hat{P}(Y)$$

Assumption:

$X_1, \ldots, X_m$ are **conditionally independent** given Y.

$$\hat{P}(X|Y) = \hat{P}(X_1, X_2, \ldots, X_{300} | Y)$$

$$= \prod_{i=1}^{m} \hat{P}(X_i | Y)$$

Naïve Bayes Assumption

- X_i are often only mildly conditionally dep. given Y
- # of params becomes tractable to compute

Naïve Bayes Classifier

$$\hat{Y} = \underset{y=\{0,1\}}{\arg\max} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Training

What is the Big-O of # of parameters we need to learn?

A. $O(m)$
B. $O(2^m)$
C. other

🤔

Naïve Bayes Classifier

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Training

for $j = 1, \ldots, m$: $\hat{P}(X_j = 1|Y = 0)$, $= 1 - \hat{P}(X_i = 0|Y = 0)$

$\hat{P}(X_j = 1|Y = 1)$ $1 - \hat{P}(X_i = 0|Y = 1)$

$\hat{P}(Y = 1)$ $= 1 - \hat{P}(Y = 0)$

$4 \cdot m + 2 = O(m)$

Use MLE or Laplace (MAP)

Testing

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\log \hat{P}(Y) + \sum_{j=1}^{m} \log \hat{P}(X_j|Y) \right)$$

(for numeric stability)

24: Naïve Bayes (live)

Lisa Yan and Jerry Cain

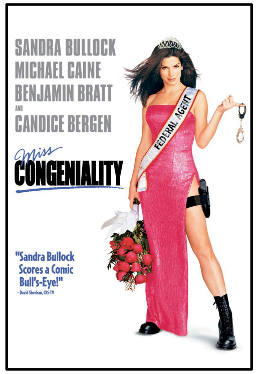
November 6, 2020

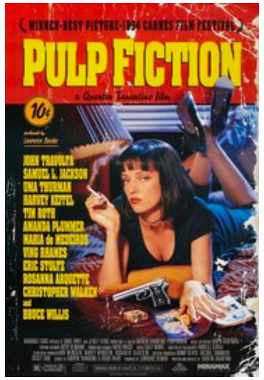
Classification terminology check

Training data: $(\boldsymbol{x}^{(1)}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \ldots, (\boldsymbol{x}^{(n)}, y^{(n)})$

A. $\boldsymbol{x}^{(i)}$
B. $y^{(i)}$
C. $(\boldsymbol{x}^{(i)}, y^{(i)})$
D. $x_j^{(i)}$

1: like movie
0: dislike movie





	Movie 1	Movie 2	…	Movie m	Output
User 1	1. 1	0	…	1	2. 1
User 2	3. 1	1	…	0	0
…			⋮		⋮
User n	0	4. 0	…	1	1

(by yourself)

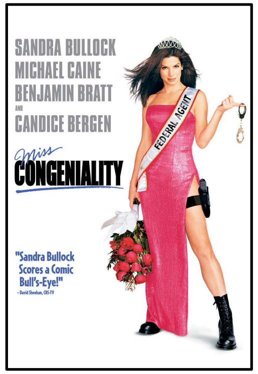
Classification terminology check

Training data: $(\boldsymbol{x}^{(1)}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \ldots, (\boldsymbol{x}^{(n)}, y^{(n)})$

A. $\boldsymbol{x}^{(i)}$
B. $y^{(i)}$
C. $(\boldsymbol{x}^{(i)}, y^{(i)})$
D. $x_j^{(i)}$

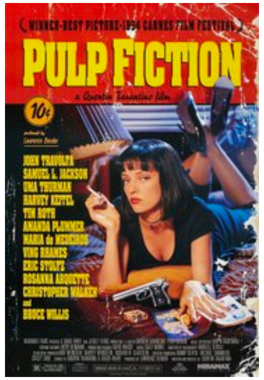
1: like movie
0: dislike movie

Movie 1



Movie 2

…



Movie m

Output

		Movie 1	Movie 2	…	Movie m		Output
User 1	1.	1	0	…	1	2.	1
User 2	3.	1	1	…	0		0
…				⋮			⋮
User n		0	4. 0	…	1		1

1. feature vector $\vec{x}$ $\boldsymbol{x}^{(1)}$, $i=1$ A
2. output/label $y^{(1)}$ B
3. $(\boldsymbol{x}^{(2)}, y^{(2)})$ $i=2$
4. $x_2^{(n)}$ $i=n$ $j=2$

i: i-th user
j: movie j

NETFLIX

and Learn

Predicting user TV preferences

Will a user like the Pokémon TV series?

Observe indicator variables $\boldsymbol{X} = (X_1, X_2)$:

$X_1 = 1$:
"likes Star Wars"

$X_2 = 1$:
"likes Harry Potter"

Output Y indicator:

$Y = 1$:
"likes Pokémon"

$\hat{p}(Y = \_ \mid \vec{X} = (1,1))$

$\hat{Y} = 0$ or 1

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

The model, and the Naïve Bayes assumption

$\leftarrow X = (1, 1)$

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

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

Bayes

$\leftarrow$ constant

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

$$= \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y)\right)\hat{P}(Y)$$

Naïve Bayes Assumption:

$X_1, \ldots, X_m$ are **conditionally independent** given Y.

$P(X_1, X_2, \ldots, X_m \mid Y = \_\_)$

$=$

Breakout Rooms

Check out the questions on the next slide (Slide 50). Post any clarifications here!

https://us.edstem.org/courses/2678/discussion/169796

Breakout rooms: 3 min

Predicting user TV preferences

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

Naïve Bayes Assumption $\quad P(\boldsymbol{X}|Y) = \prod_{j=1}^{m} P(X_j|Y)$

1. Which probabilities do you need to estimate? How many are there?

$\hat{P}(X_1 = \_\_, X_2 = \_\_ \mid Y = \_\_) \qquad \hat{P}(Y = \_\_)$

- Brute Force Bayes (strawman, without NB assumption)
- Naïve Bayes

2. During training, how to estimate the prob $\hat{P}(X_1 = 1, X_2 = 1 | Y = 0)$ with MLE? with Laplace?

- Brute Force Bayes
- Naïve Bayes

Predicting user TV preferences

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

Naïve Bayes Assumption $\quad P(\boldsymbol{X}|Y) = \prod_{j=1}^{m} P(X_j|Y)$

Which probabilities do you need to estimate?
How many are there?

- Brute Force Bayes (strawman, without NB assumption)

$\hat{P}(Y=0) \quad \hat{P}(Y=1) \qquad 2$

$\hat{P}(X_1 = \_, X_2 = \_ \,|\, Y=0) \qquad 2^m = 2^2$

$\hat{P}(X_1 = \_, X_2 = \_ \,|\, Y=1) \qquad 2^m = 2^2 \qquad 2 + 2^{m+1}$

10

- Naïve Bayes

$\hat{P}(Y=0), \hat{P}(Y=1) \qquad 2$

$\forall j = 1, \ldots, m \quad \hat{P}(X_j = \_ \,|\, Y=0) \qquad 2m \qquad 2 + 4m$

$\hat{P}(X_j = \_ \,|\, Y=1) \qquad 2m$

10

During training, how to estimate the prob $\hat{P}(X_1 = 1, X_2 = 1 | Y = 0)$ with MLE? with Laplace?

- Brute Force Bayes
- Naïve Bayes

(strawman brute force) Multinomial MLE and MAP

Model: Multinomial, m outcomes:
p_j probability of outcome j

Observe: n_j = # of trials with outcome j
Total of $\sum_{j=1}^{m} n_j$ trials

MLE
$$\hat{p}_j = \frac{n_j}{\sum_{j=1}^{m} n_j}$$

Laplace estimate (MAP w/Laplace smoothing)
$$\hat{p}_j = \frac{n_j + 1}{\sum_{j=1}^{m} n_j + m}$$

X_1	X_2	Y
0	1	1
1	1	0
1	0	1
1	0	0
…	…	…
1	1	1

training data

$$\hat{P}(X_1 = 1, X_2 = 1 | Y = 0)$$

MLE $\dfrac{\#(X_1=1 \cap X_2=1 \cap Y=0)}{\#(Y=0)}$

MAP $\dfrac{\#(X_1=1 \cap X_2=1 \cap Y=0) + 1}{\#(Y=0) + \boxed{4}}$

(Naïve Bayes) Multinomial MLE and MAP

Model: Multinomial, m outcomes: p_j probability of outcome j

Observe: n_j = # of trials with outcome j
Total of $\sum_{j=1}^{m} n_j$ trials

MLE $\hat{p}_j = \frac{n_j}{\sum_{j=1}^{m} n_j}$

Laplace estimate (MAP w/Laplace smoothing) $\hat{p}_j = \frac{n_j + 1}{\sum_{j=1}^{m} n_j + m}$

X_1	X_2	Y
0	1	1
1	1	0
1	0	1
… 0	… 1	… 0
1	1	1

training data

$\hat{P}(X_1 = 1, X_2 = 1 | Y = 0) \xrightarrow{NB} \hat{P}(X_1 = 1 | Y = 0)\ \hat{P}(X_2 = 1 | Y = 0)$

$\hat{P}(X_1 = 1 | Y = 0) \xrightarrow{MLE} \frac{1}{2}$

$\xrightarrow{MAP} \frac{1+1}{2+2}$

$\hat{P}(X_2 = 1 | Y = 0)$

NETFLIX

and Learn

naively

Ex 1. Naïve Bayes Classifier (MLE)

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Training

$\forall i$: $\hat{P}(X_j = 1|Y = 0), \hat{P}(X_j = 0|Y = 0),$
$\hat{P}(X_j = 1|Y = 1), \hat{P}(X_j = 0|Y = 0),$
$\hat{P}(Y = 1), \hat{P}(Y = 0)$

Use MLE or Laplace (MAP)

Testing

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Think

Slide 59 has two questions to go over by yourself.

Post any clarifications here!

https://us.edstem.org/courses/2678/discussion/153773

Think by yourself: 1 min

(by yourself)

Training: Naïve Bayes for TV shows (**MLE**)

Observe indicator vars. $\boldsymbol{X} = (X_1, X_2)$:

- X_1: "likes Star Wars"
- X_2: "likes Harry Potter"

Predict Y: "likes Pokémon"

$Y \backslash X_1$	0	1
0	3	10
1	4	13

$Y \backslash X_2$	0	1
0	5	8
1	7	10

Training data counts

1. How many datapoints (n) are in our train data?
2. Compute MLE estimates for $\hat{P}(X_1|Y)$:

$Y \backslash X_1$	0	1
0	$\hat{P}(X_1 = 0 \| Y = 0)$	$\hat{P}(X_1 = 1 \| Y = 0)$
1	$\hat{P}(X_1 = 0 \| Y = 1)$	$\hat{P}(X_1 = 1 \| Y = 1)$

(by yourself)

Training: Naïve Bayes for TV shows (MLE)

X_1	X_2	Y
0	1	0

Observe indicator vars. $\boldsymbol{X} = (X_1, X_2)$:

- X_1: “likes Star Wars”
- X_2: “likes Harry Potter”

Predict Y: “likes Pokémon”

Y \ X_1	0	1
0	3	10
1	4	13

Y \ X_2	0	1
0	5	8
1	7	10

Training data counts

Y	
0	13
1	17

1. How many datapoints (n) are in our train data?

$n = 30$

2. Compute MLE estimates for $\hat{P}(X_1|Y)$:

Y \ X_1	0	1
0	$P(X_1=0 \mid Y=0) = 3/13$	$10/13$
1	$4/17$	$13/17$

Training: Naïve Bayes for TV shows (MLE)

Observe indicator vars. $\boldsymbol{X} = (X_1, X_2)$:

- X_1: "likes Star Wars"
- X_2: "likes Harry Potter"

Predict Y: "likes Pokémon"

$Y \backslash X_1$	0	1
0	3	10
1	4	13

$Y \backslash X_2$	0	1
0	5	8
1	7	10

Y	
0	13
1	17

Training data counts

$Y \backslash X_1$	0	1
0	0.23	0.77
1	0.24	0.76

(from last slide)

$\hat{P}(X_1 = \_ \mid Y = \_)$

$Y \backslash X_2$	0	1
0	$5/13 \approx 0.38$	$8/13 \approx 0.62$
1	$7/17 \approx 0.41$	$10/17 \approx 0.59$

$\hat{P}(X_2 = \_ \mid Y = \_)$

Y	
0	$13/30 \approx 0.43$
1	$17/30 \approx 0.57$

$\hat{P}(Y = \_)$

Training : Naïve Bayes for TV shows (MLE)

Observe indicator vars. $\boldsymbol{X} = (X_1, X_2)$:

- X_1: "likes Star Wars"
- X_2: "likes Harry Potter"

Predict Y: "likes Pokémon"

$Y \backslash X_1$	0	1
0	0.23	0.77
1	0.24	0.76

$Y \backslash X_2$	0	1
0	0.38	0.62
1	0.41	0.59

Y	
0	0.43
1	0.57

Now that we've trained and found parameters,
It's time to classify new users!

Ex 1. Naïve Bayes Classifier (MLE)

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Training

$\forall i$: $\hat{P}(X_j = 1|Y = 0), \hat{P}(X_j = 0|Y = 0),$
$\hat{P}(X_j = 1|Y = 1), \hat{P}(X_j = 0|Y = 0),$
$\hat{P}(Y = 1), \hat{P}(Y = 0)$

Use MLE or Laplace (MAP)

Testing

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Testing: Naïve Bayes for TV shows (MLE)

Observe indicator vars. $\boldsymbol{X} = (X_1, X_2)$:

- X_1: "likes Star Wars"
- X_2: "likes Harry Potter"

Predict Y: "likes Pokémon"

$Y \backslash X_1$	0	1
0	0.23	0.77
1	0.24	0.76

$Y \backslash X_2$	0	1
0	0.38	0.62
1	0.41	0.59

Y	
0	0.43
1	0.57

Suppose a **new person** "likes Star Wars" ($X_1 = 1$) but "dislikes Harry Potter" ($X_2 = 0$).

Will they like Pokemon? Need to predict Y:

$$\hat{Y} = \arg\max_{y=\{0,1\}} \hat{P}(\boldsymbol{X}|Y)\hat{P}(Y) \quad = \arg\max_{y=\{0,1\}} \hat{P}(X_1|Y)\hat{P}(X_2|Y)\hat{P}(Y)$$

If $Y = 0$: $\quad \hat{P}(X_1 = 1|Y = 0)\hat{P}(X_2 = 0|Y = 0)\hat{P}(Y = 0) = 0.77 \cdot 0.38 \cdot 0.43 = 0.126$

If $Y = 1$: $\quad \hat{P}(X_1 = 1|Y = 1)\hat{P}(X_2 = 0|Y = 1)\hat{P}(Y = 1) = 0.76 \cdot 0.41 \cdot 0.57 = 0.178$

Since term is greatest when Y = 1, predict $\hat{Y} = 1$

history of life

homo apriorus

θ, X

homo sapien

$\theta \mid X$

homo ~~posterior~~ bayesianus

Interlude for jokes/announcements

Announcements

Problem Set 5

Super on-time due (+8%):	earlier today
On-time due (+5%):	Monday 11/9 1:00pm
Grace period ends (+0%):	~~Wednesday 11/11~~ 1:00pm Tuesday 11/10

Problem Set 6

Out:	later today
Due:	Monday 11/16
Grace period:	Wednesday 11/18
Covers:	through Lecture 26 Wed

Ex 2. Naïve Bayes Classifier (**MAP**) Laplace

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Training

$\forall i$: $\hat{P}(X_j = 1|Y = 0), \hat{P}(X_j = 0|Y = 0),$
$\hat{P}(X_j = 1|Y = 1), \hat{P}(X_j = 0|Y = 0),$
$\hat{P}(Y = 1), \hat{P}(Y = 0)$

Use MLE or **Laplace (MAP)**

Testing

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

(note the same as before)

Training: Naïve Bayes for TV shows (**MAP**)

Observe indicator vars. $\boldsymbol{X} = (X_1, X_2)$:

- X_1: "likes Star Wars"
- X_2: "likes Harry Potter"

Predict Y: "likes Pokémon"

$Y \backslash X_1$	0	1
0	3	10
1	4	13

$Y \backslash X_2$	0	1
0	5	8
1	7	10

Training data counts

What are our MAP estimates using Laplace smoothing for $\hat{P}(X_j|Y)$?

$\hat{P}(X_j = x|Y = y)$:

A. $\dfrac{\#(X_j = x, Y = y)}{\#(Y = y)}$

B. $\dfrac{\#(X_j = x, Y = y) + 1}{\#(Y = y) + 2}$

C. $\dfrac{\#(X_j = x, Y = y) + 1}{\#(Y = y) + 4}$

D. other

(by yourself)

Training: Naïve Bayes for TV shows (**MAP**)

Observe indicator vars. $\boldsymbol{X} = (X_1, X_2)$:

- X_1: "likes Star Wars"
- X_2: "likes Harry Potter"

Predict Y: "likes Pokémon"

Y \ X_1	0	1
0	3	10
1	4	13

Y \ X_2	0	1
0	5	8
1	7	10

Training data counts

What are our MAP estimates using Laplace smoothing for $\hat{P}(X_j|Y)$ and $\hat{P}(Y)$?

$\hat{P}(X_j = x|Y = y)$:

A. $\dfrac{\#(X_j=x, Y=y)}{\#(Y=y)}$

B. $\dfrac{\#(X_j=\boxed{x}, Y=y)+1}{\#(Y=y)+2}$

C. $\dfrac{\#(X_j=x, Y=y)+1}{\#(Y=y)+4}$

D. other

Training: Naïve Bayes for TV shows (MAP)

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{i=1}^{m} \hat{P}(X_i|Y) \right) \hat{P}(Y)$$

Observe indicator vars. $\boldsymbol{X} = (X_1, X_2)$:

- X_1: "likes Star Wars"
- X_2: "likes Harry Potter"

Predict Y: "likes Pokémon"

Y \ X_1	0	1
0	3	10
1	4	13

Y \ X_2	0	1
0	5	8
1	7	10

Y	
0	13
1	17

Training data counts

Y \ X_1	0	1
0	0.27	0.73
1	0.26	0.74

Y \ X_2	0	1
0	0.40	0.60
1	0.42	0.58

$\frac{3+1}{13+2} = 0.27$ $\frac{10+1}{13+2} = 0.73$

$\frac{4+1}{17+2} = 0.26$

In practice:

- We use Laplace for $\hat{P}(X_j|Y)$ in case some events $X_j = x_j$ don't appear
- We don't use Laplace for $\hat{P}(Y)$, because all class labels should appear reasonably often

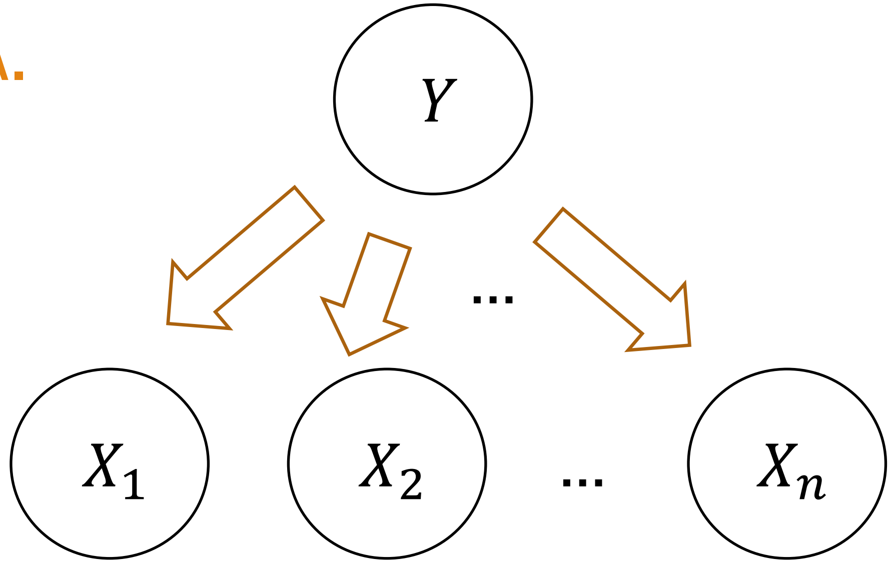
Naïve Bayes Model is a Bayesian Network

Naïve Bayes Assumption

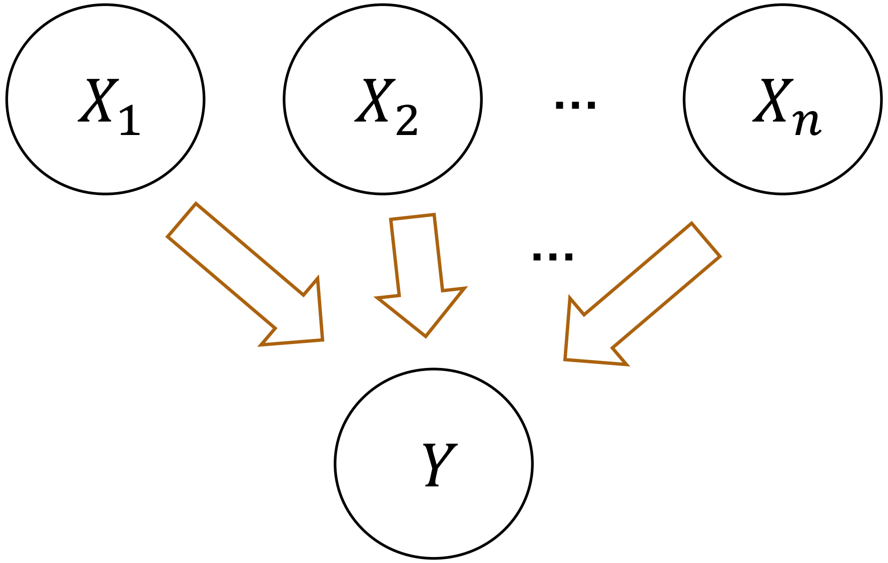
$$P(\boldsymbol{X}|Y) = \prod_{j=1}^{m} P(X_j|Y)$$

Which Bayesian Network encodes this conditional independence?

A.



B.



X_i are conditionally independent given Y

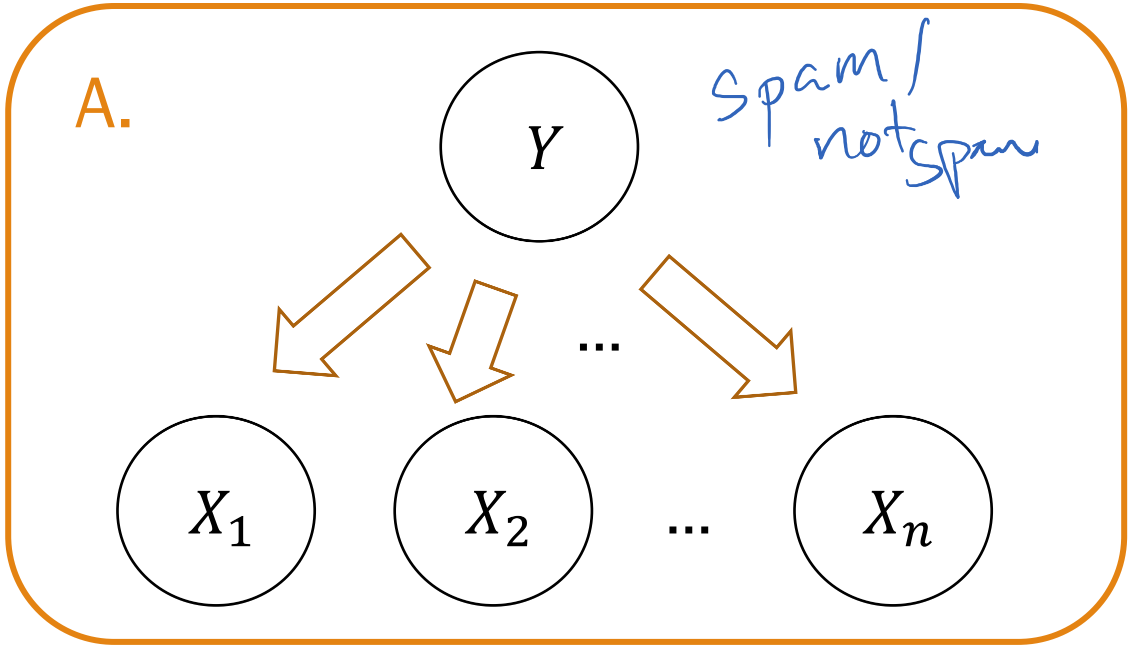
(by yourself)

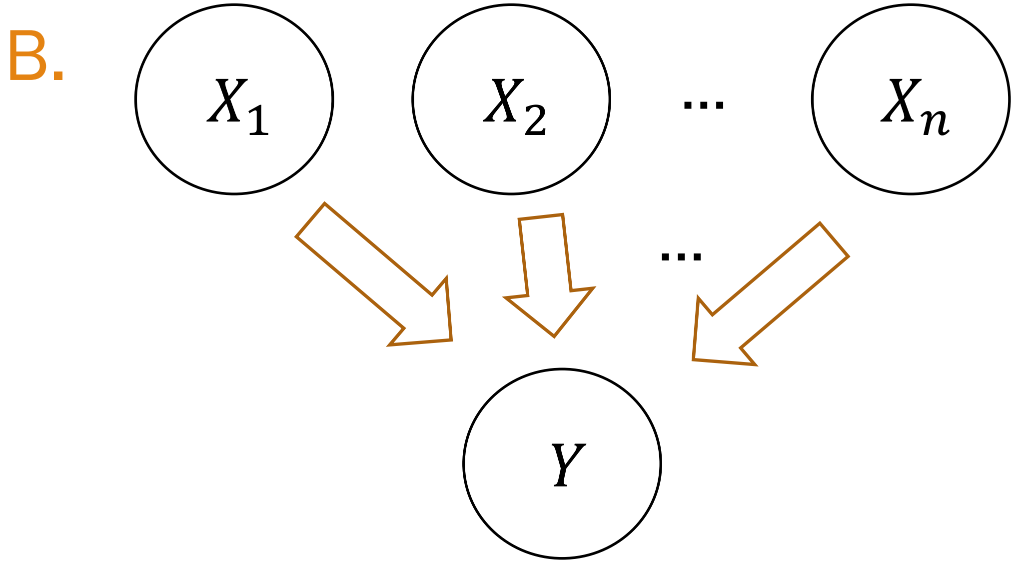
Naïve Bayes Model is a Bayesian Network

Naïve Bayes Assumption

$$P(\mathbf{X}|Y) = \prod_{i=1}^{m} P(X_i|Y) \quad \Rightarrow \quad P(\mathbf{X}, Y) = P(Y)\prod_{j=1}^{m} P\left(X_j|Y\right)$$

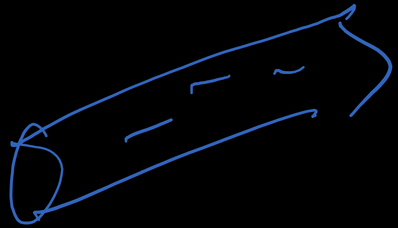
Which Bayesian Network encodes this conditional independence?

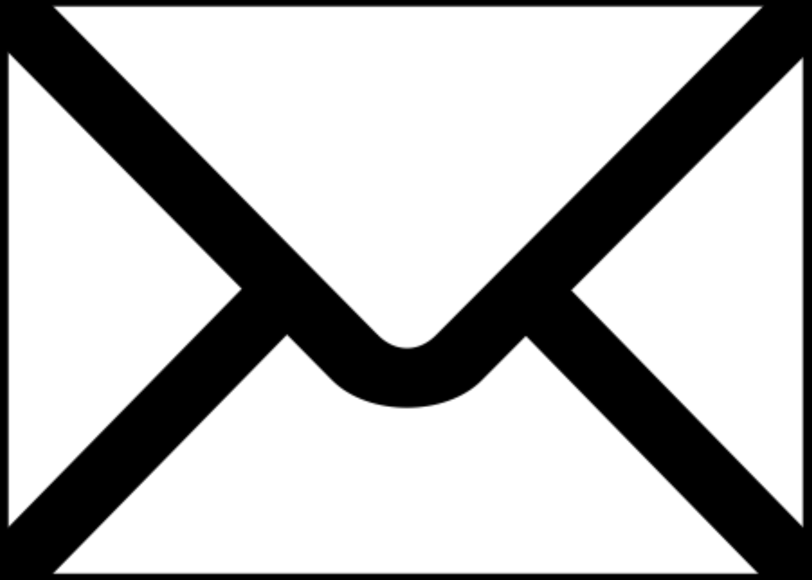




X_i are conditionally independent given parent Y

log

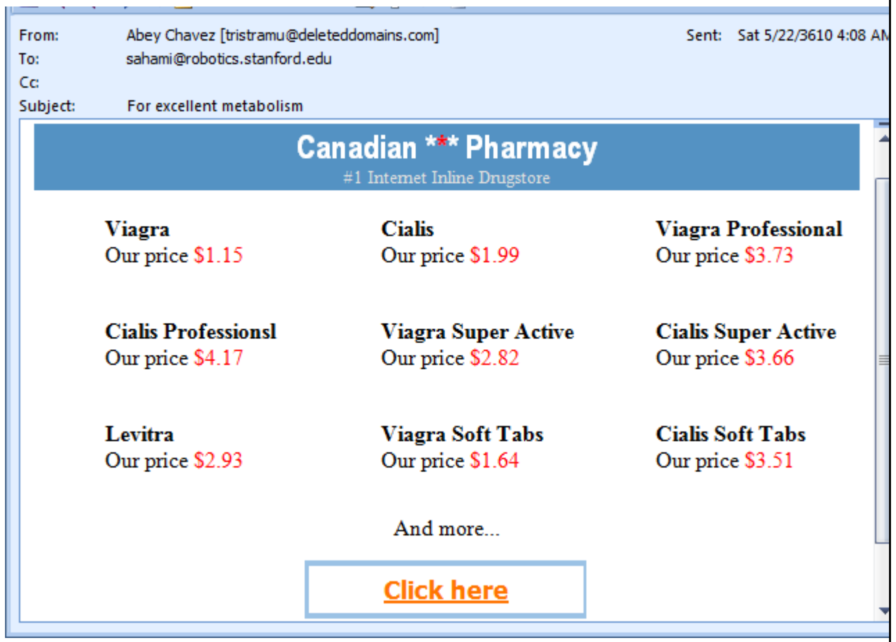




and Learn

naively

What is Bayes doing in my mail server?



Let's get Bayesian on your spam:

```
Content analysis details:   (49.5 hits, 7.0 required)
 0.9 RCVD_IN_PBL            RBL: Received via a relay in Spamhaus PBL
                            [93.40.189.29 listed in zen.spamhaus.org]
 1.5 URIBL_WS_SURBL         Contains an URL listed in the WS SURBL blocklist
                            [URIs: recragas.cn]
 5.0 URIBL_JP_SURBL         Contains an URL listed in the JP SURBL blocklist
                            [URIs: recragas.cn]
 5.0 URIBL_OB_SURBL         Contains an URL listed in the OB SURBL blocklist
                            [URIs: recragas.cn]
 5.0 URIBL_SC_SURBL         Contains an URL listed in the SC SURBL blocklist
                            [URIs: recragas.cn]
 2.0 URIBL_BLACK            Contains an URL listed in the URIBL blacklist
                            [URIs: recragas.cn]
 8.0 BAYES_99               BODY: Bayesian spam probability is 99 to 100%
                            [score: 1.0000]
```

A Bayesian Approach to Filtering Junk E-Mail

Mehran Sahami* Susan Dumais† David Heckerman† Eric Horvitz†

*Gates Building 1A
Computer Science Department
Stanford University
Stanford, CA 94305-9010
sahami@cs.stanford.edu

†Microsoft Research
Redmond, WA 98052-6399
{sdumais, heckerma, horvitz}@microsoft.com

Abstract

In addressing the growing problem of junk E-mail on the Internet, we examine methods for the automated

contain offensive material (such as graphic pornography), there is often a higher cost to users of actually viewing this mail than simply the time to sort out the

Ex 3. Naïve Bayes Classifier (m, n large)

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{i=1}^{m} \hat{P}(X_i|Y) \right) \hat{P}(Y)$$

Training

$\forall i$: $\hat{P}(X_i$ … $= 0)$, Use MLE or

… $0|Y = 0)$, Laplace (MAP)

$\hat{P}$ …

What changes are necessary?

① $\dfrac{\#(X_j = \_\_ \cap Y = \_\_) + 1}{\#(Y = \_\_) + 2}$

② log prob.

Testing

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Email classification

Goal Based on email content $\boldsymbol{X}$, predict if email is spam or not.

Features Consider a lexicon m words (for English: $m \approx 100{,}000$).

$\boldsymbol{X} = (X_1, X_2, \dots, X_m)$, m indicator variables

$X_j = 1$ if word j appeared in document

Output $Y = 1$ if email is spam

Note: m is huge. Make Naïve Bayes assumption: $P(\boldsymbol{X}|\text{spam}) = \prod_{j=1}^{m} P\left(X_j|\text{spam}\right)$

Appearances of words in email are conditionally independent given the email is spam or not

Training: Naïve Bayes Email classification

Train set n previous emails $(\boldsymbol{x}^{(1)}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \ldots, (\boldsymbol{x}^{(n)}, y^{(n)})$

$\boldsymbol{x}^{(i)} = \left(x_1^{(i)}, x_2^{(i)}, \ldots, x_m^{(i)}\right)$ for each word, whether it appears in email i

$y^{(i)} = 1$ if spam, 0 if not spam

Note: m is huge.

Which estimator should we use for $\hat{P}(X_j|Y)$?

A. MLE
B. Laplace estimate (MAP)
C. Other MAP estimate
D. Both A and B

(by yourself)

Training: Naïve Bayes Email classification

Train set n previous emails $\left(\boldsymbol{x}^{(1)}, y^{(1)}\right), \left(\boldsymbol{x}^{(2)}, y^{(2)}\right), \ldots, \left(\boldsymbol{x}^{(n)}, y^{(n)}\right)$

$\boldsymbol{x}^{(i)} = \left(x_1^{(i)}, x_2^{(i)}, \ldots, x_m^{(i)}\right)$ for each word, whether it appears in email i

$y^{(i)} = 1$ if spam, 0 if not spam

Note: m is huge.

Which estimator should we use for $\hat{P}\left(X_j|Y\right)$?

A. MLE
B. Laplace estimate (MAP)
C. Other MAP estimate
D. Both A and B

Many words are likely to not appear at all in the training set!

Ex 3. Naïve Bayes Classifier (m, n large)

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Training

$\forall i$: $\hat{P}(X_j = 1|Y = 0), \hat{P}(X_j = 0|Y = 0)$,
$\hat{P}(X_j = 1|Y = 1), \hat{P}(X_j = 0|Y = 0)$,
$\hat{P}(Y = 1), \hat{P}(Y = 0)$

Use MLE or Laplace (MAP)

Testing

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \right.$$

Laplace (MAP) estimates avoid estimating 0 probabilities for events that don't occur in your training data.

Testing: Naïve Bayes Email classification

For a new email:

- Generate $\boldsymbol{X} = (X_1, X_2, \dots, X_m)$
- Classify as spam or not using Naïve Bayes assumption

Note: m is huge.

Suppose train set size n also huge (many labeled emails).

Can we still use the below prediction?

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Testing: Naïve Bayes Email classification

For a new email:

- Generate $\boldsymbol{X} = (X_1, X_2, \dots, X_m)$
- Classify as spam or not using Naïve Bayes assumption

Note: m is huge.

Suppose train set size n also huge (many labeled emails).

Can we still use the below prediction?

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Will probably lead to underflow!

Ex 3. Naïve Bayes Classifier (m, n large)

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\prod_{j=1}^{m} \hat{P}(X_j|Y) \right) \hat{P}(Y)$$

Training

$\forall i$: $\hat{P}(X_j = 1|Y = 0), \hat{P}(X_j = 0|Y =$

$\hat{P}(X_j = 1|Y = 1), \hat{P}(X_j = 0|Y$ P)

$\hat{P}(Y = 1), \hat{P}(Y = 0)$

Use sums of log-probabilities for numerical stability.

Testing

$$\hat{Y} = \arg\max_{y=\{0,1\}} \left(\log \hat{P}(Y) + \sum_{j=1}^{m} \log \hat{P}(X_j|Y) \right)$$

How well does Naïve Bayes perform?

After training, you can test with another set of data, called the **test set**.

- Test set also has known values for Y so we can see how often we were right/wrong in our predictions $\hat{Y}$.

Typical workflow:

- Have a dataset of 1789 emails (1578 spam, 211 ham)
- Train set: First 1538 emails (by time)
- Test set: Next 251 messages

Evaluation criteria on test set:

$$\textbf{precision} = \frac{(\text{\# correctly predicted class } Y)}{(\text{\# predicted class } Y)}$$

$$\textbf{recall} = \frac{(\text{\# correctly predicted class } Y)}{(\text{\# real class } Y \text{ messages})}$$

	Spam		Non-spam	
	Prec.	Recall	Prec.	Recall
Words only	97.1%	94.3%	87.7%	93.4%
Words + addtl features	100%	98.3%	96.2%	100%

What are precision and recall?

Accuracy (# correct)/(# total) sometimes just doesn't cut it.

Precision: Of the emails you predicted as spam, how many are *actually* spam? — Measure of false positives

Recall: Of the emails that are actually spam, how many did you predict? — Measure of false negatives

More on Wikipedia (https://en.wikipedia.org/wiki/Precision_and_recall) and Problem Set 6!