23: Naïve Bayes

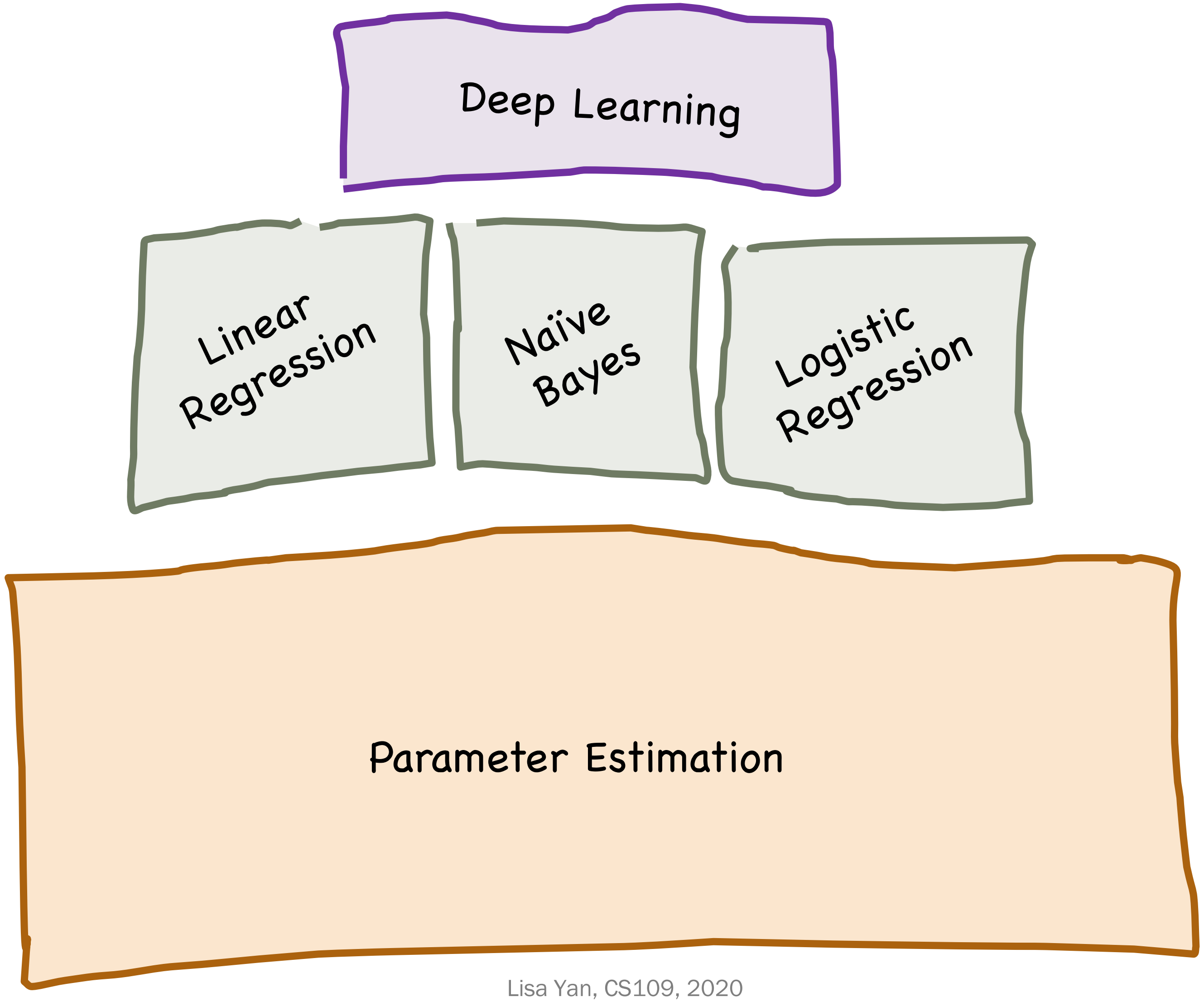
Lisa Yan

May 29, 2020

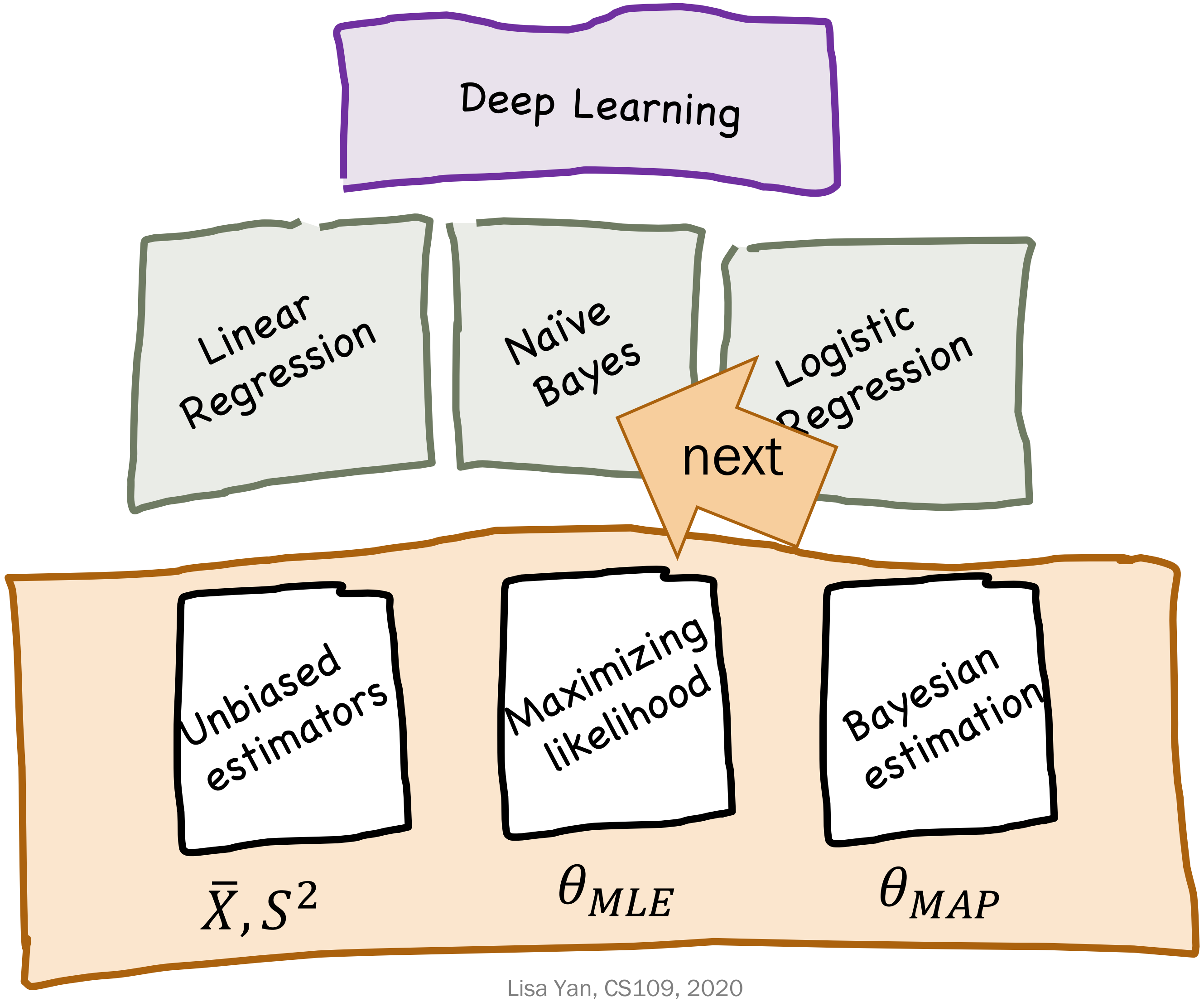
Quick slide reference

Intro: Machine Learning

Our path from here



Our path from here



Machine Learning (formally)

Many different forms of "Machine Learning"

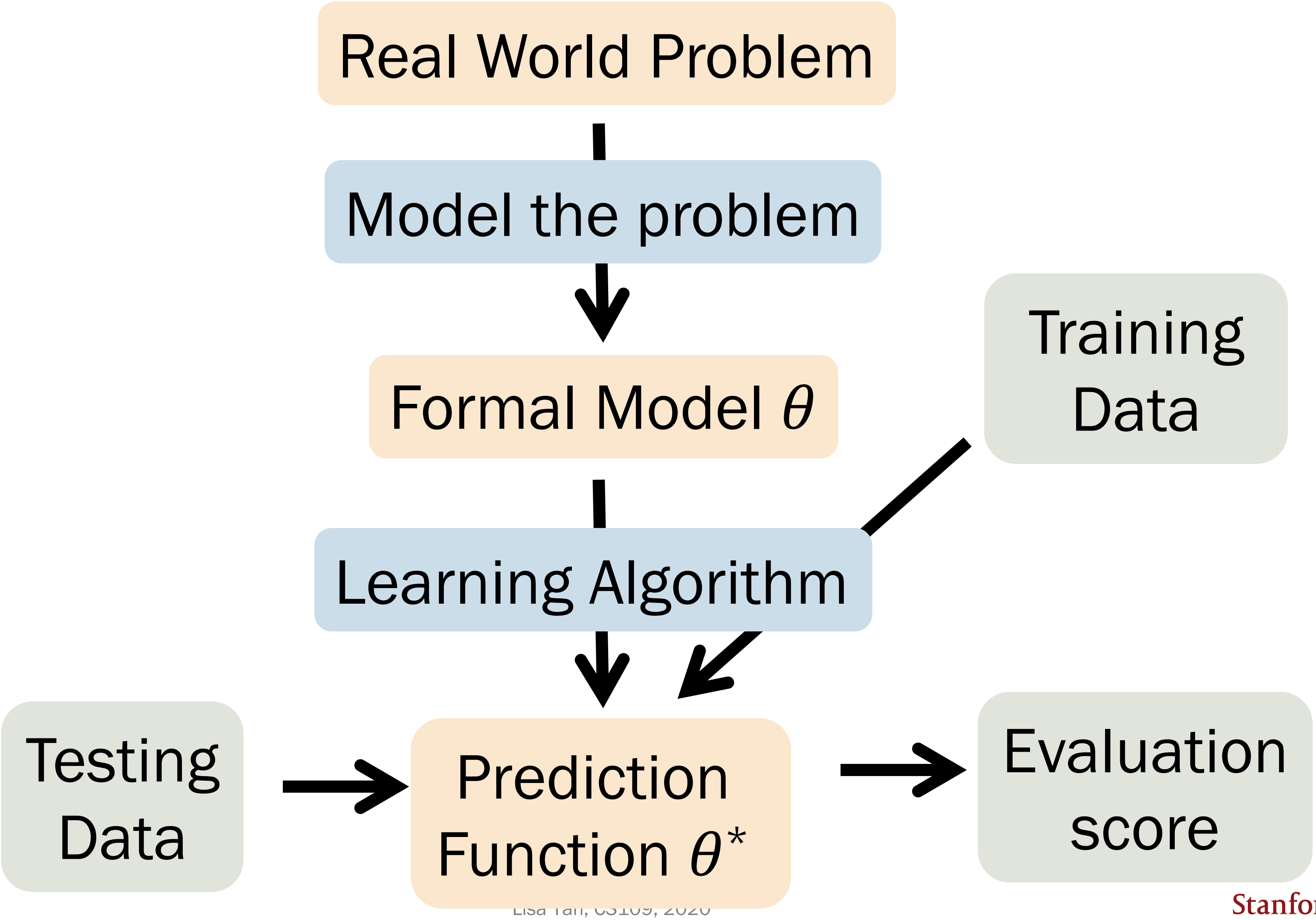
- We focus on the problem of prediction based on observations.

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

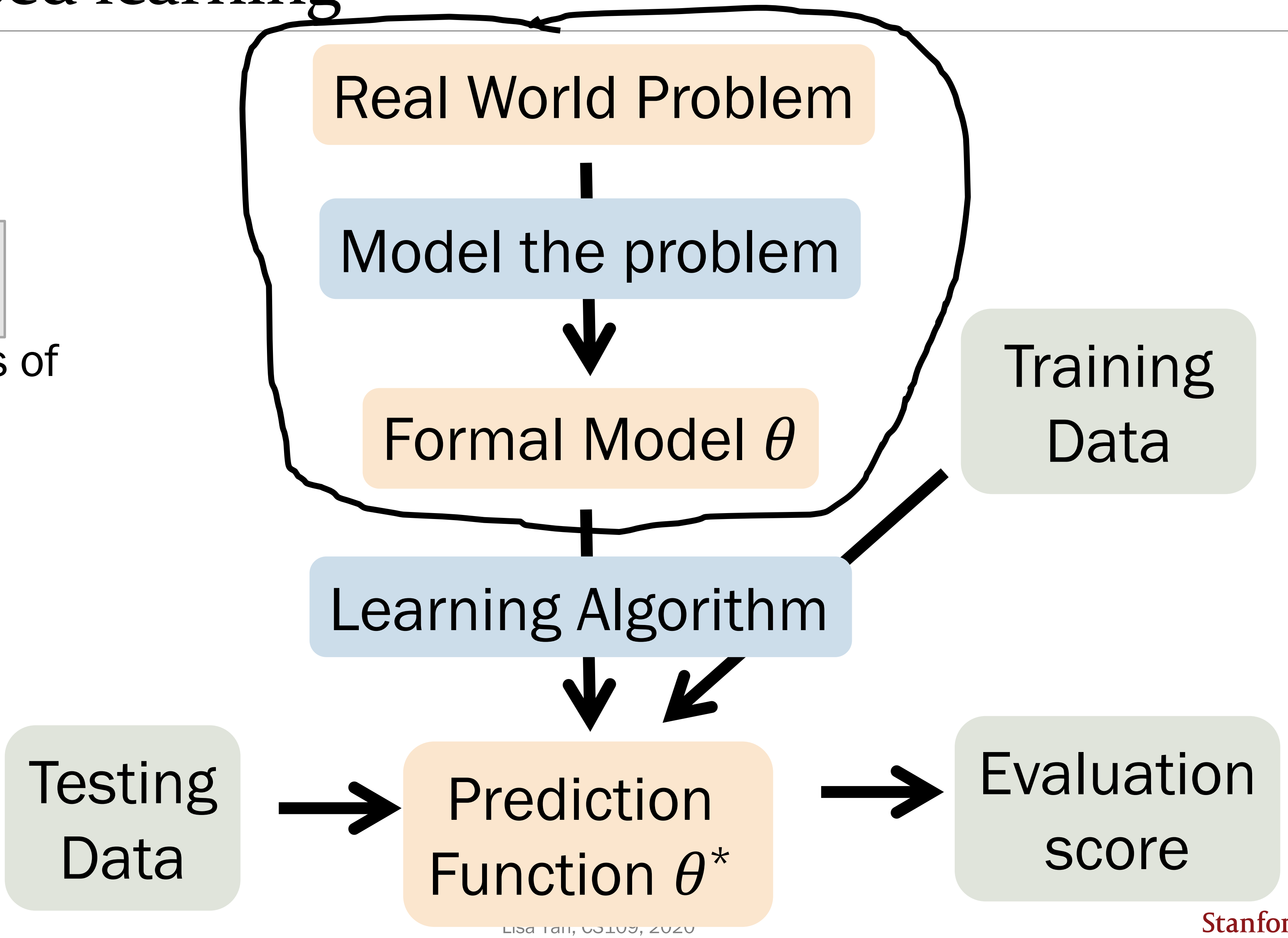
Supervised learning



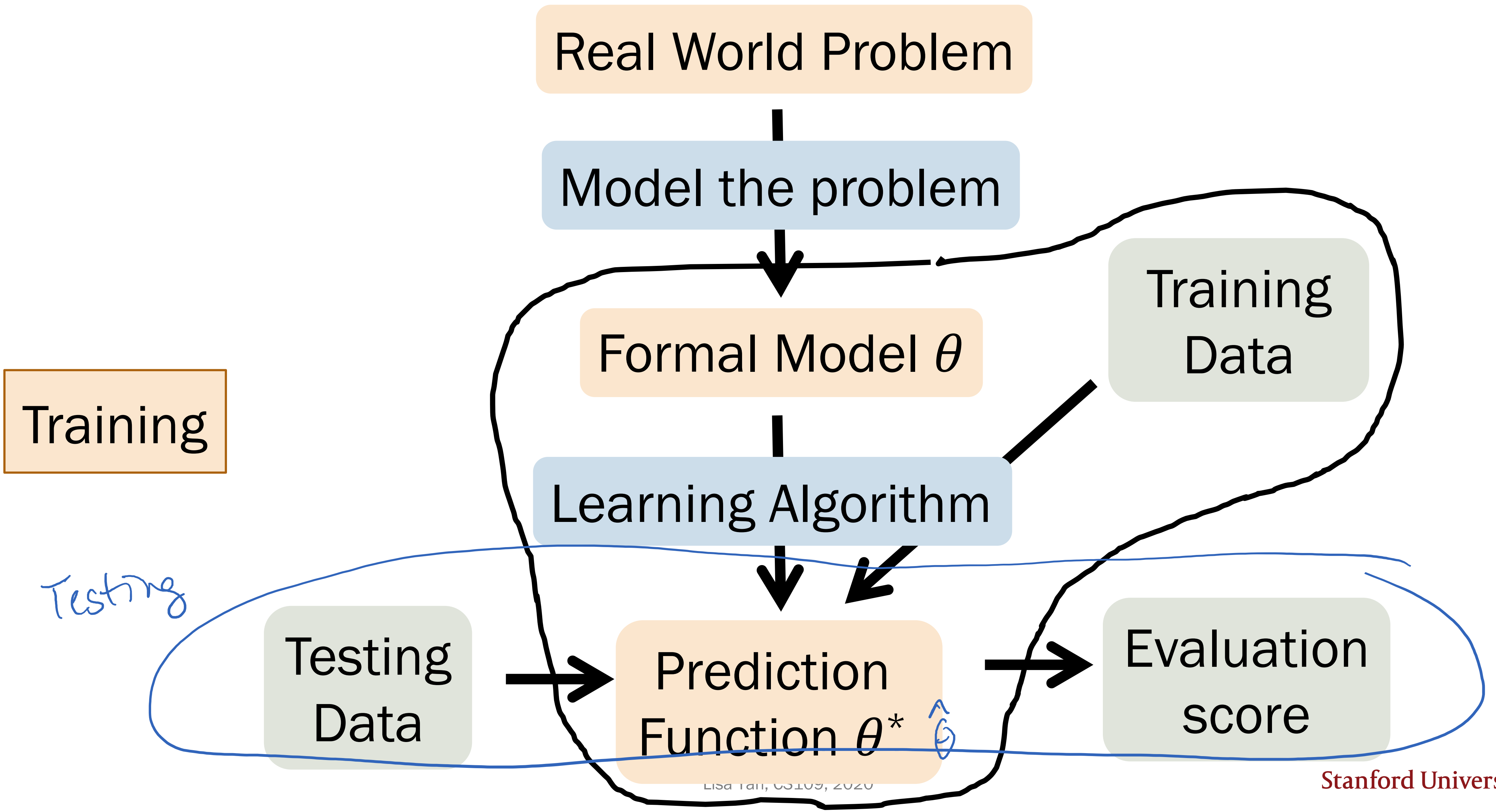
Supervised learning

Modeling

(not the focus of this class)



Supervised learning



Model and dataset

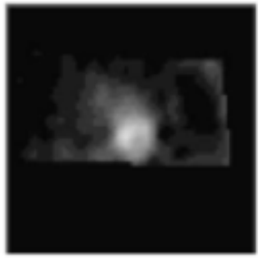
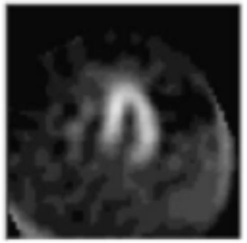
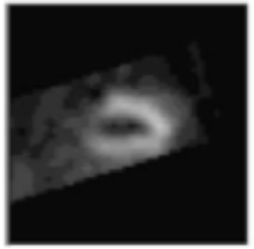
Many different forms of "Machine Learning"

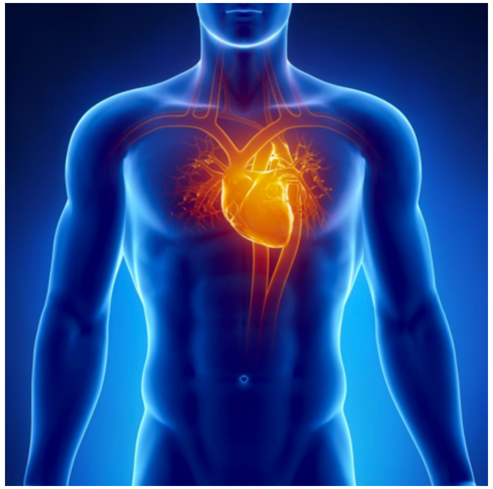
- We focus on the problem of prediction based on observations.

Goal	Based on observed $\boldsymbol{X}$, predict unseen Y
• Features	Vector $\boldsymbol{X}$ of m observed variables $\boldsymbol{X} = (X_1, X_2, \dots, X_m)$
• Output	Variable Y (also called class label if discrete)
Model	$\hat{Y} = g(\boldsymbol{X})$, a function of observations $\boldsymbol{X}$

Training data

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



binary vector

	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 ←

Training data notation

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

m-dimensional observation n datapoints, generated i.i.d.

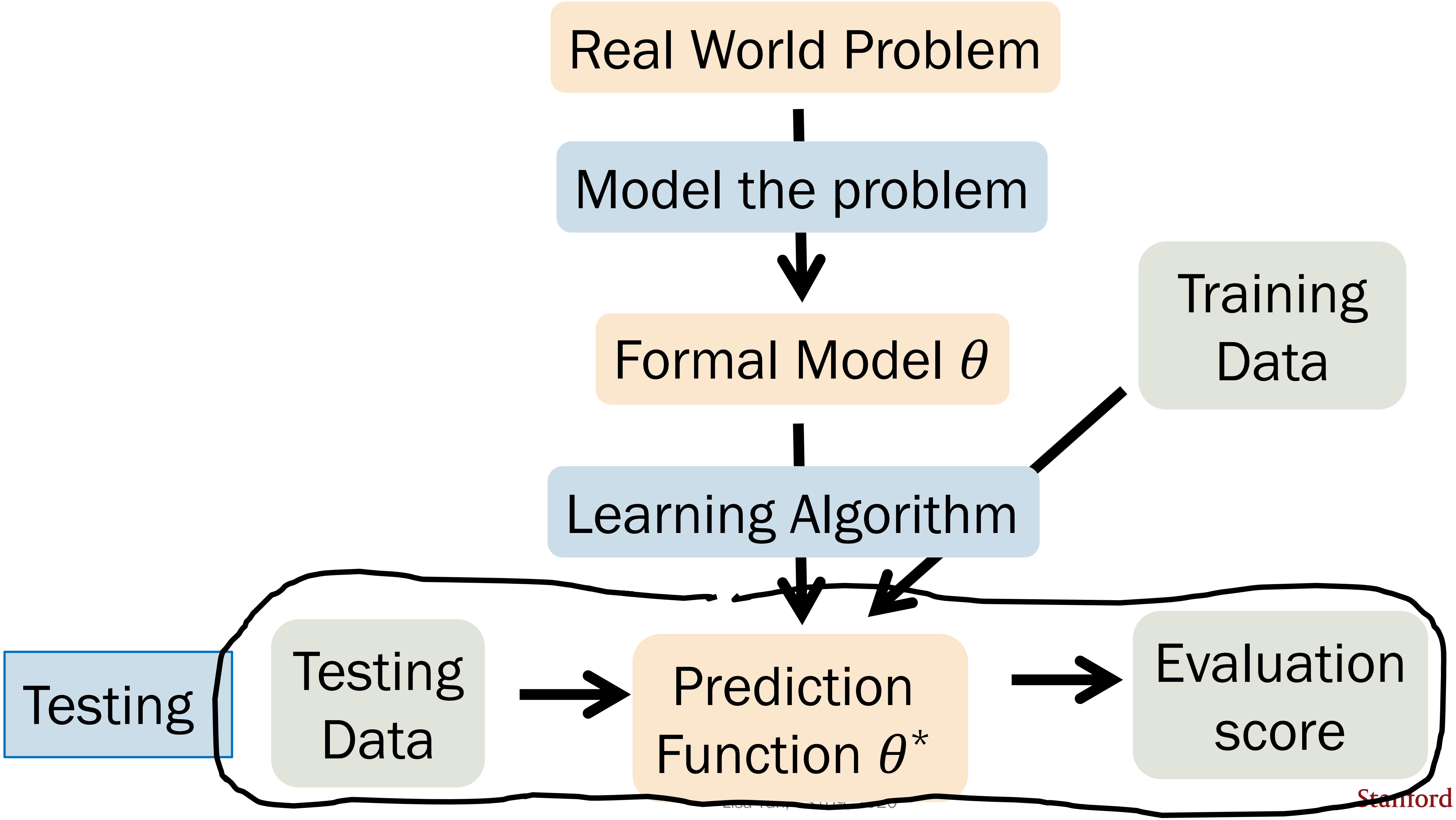
1-dimensional output

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

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

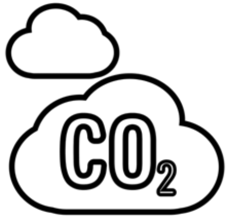
Many different forms of "Machine Learning"

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

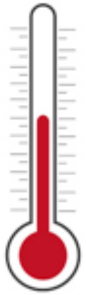
Goal	Based on observed $\boldsymbol{X}$, predict unseen Y
• Features	Vector $\boldsymbol{X}$ of m observed variables $\boldsymbol{X} = (X_1, X_2, \dots, X_m)$
• Output	Variable Y (also called class label if discrete)
Model	$\hat{Y} = g(\boldsymbol{X})$, a function of observations $\boldsymbol{X}$
• Regression	prediction when Y is continuous
• Classification	prediction when Y is discrete

Regression: Predicting real numbers

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



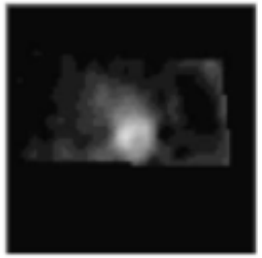
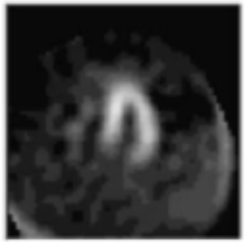
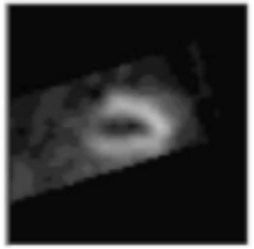




	CO2 levels	Sea level	…	Feature m	Output (Global Land-Ocean temperature)
Year 1	338.8	0	…	1	0.26
Year 2	340.0	1	…	0	0.32
…			⋮		⋮
Year n	340.76	0	…	1	0.14

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

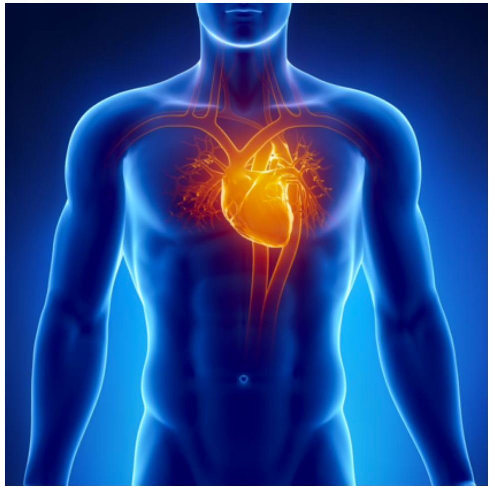
$$\hat{Y} = 1$$

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

Our focus today!

Classification: Example datasets

Heart



Ancestry

23andMe

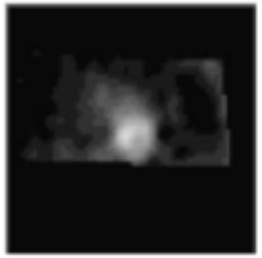
NETFLIX

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 $\boldsymbol{X}$

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

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

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

(Bayes' Theorem)

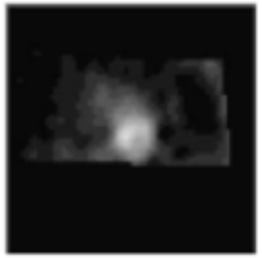
$$= \arg\max_{y=\{0,1\}} P(\boldsymbol{X}|Y)P(Y)$$

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

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

Training: Estimate parameters

$\boldsymbol{X} = (X_1)$



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

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

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

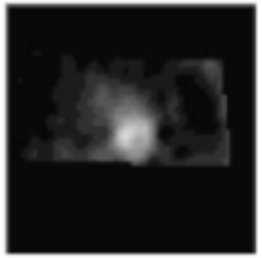
	$\hat{P}(\boldsymbol{X}\|Y=0)$	$\hat{P}(\boldsymbol{X}\|Y=1)$
$X_1 = 0$	θ_1	θ_3
$X_1 = 1$	θ_2	θ_4

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
Patient 1	1	0
Patient 2	1	1
	⋮	⋮
Patient n	0	1

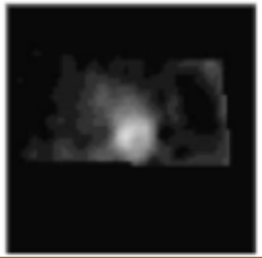
	$\hat{P}(\boldsymbol{X}\|Y=0)$	$\hat{P}(\boldsymbol{X}\|Y=1)$
$X_1 = 0$	θ_1	θ_3
$X_1 = 1$	θ_2 $=1-\theta_1$	θ_4 $=1-\theta_3$

$\boldsymbol{X}|Y=0$ 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

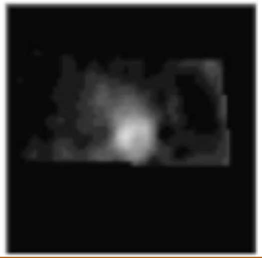
MLE

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

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

Just count!

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



Count:	# datapoints
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

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

MLE

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

Just count!

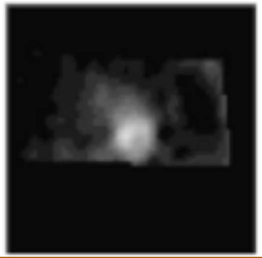
MAP

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

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. $0.052 < 0.5 \Rightarrow \hat{Y} = 1$
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?

Testing

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

(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
$Y = 1$	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)

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

Testing

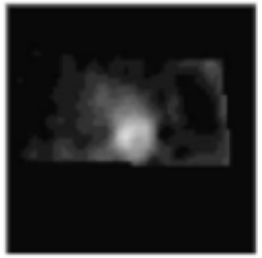
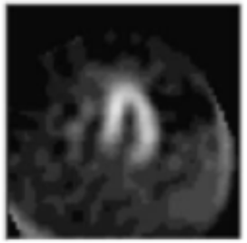
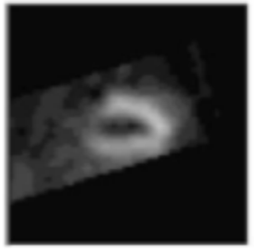
Given an observation $\boldsymbol{X} = (X_1, X_2, \ldots, X_m)$, predict

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

Naïve Bayes Classifier

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

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

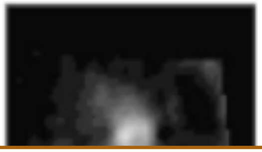
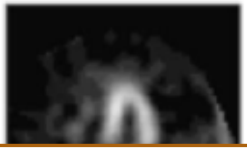
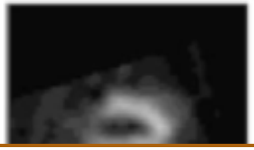
  …  

	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)

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


Count:

	# datapoints
$X_1 = 0, X_2 = 0, \ldots, X_{299} = 0, X_{300} = 0, Y = 0$:	0
$X_1 = 0, X_2 = 0, \ldots, X_{299} = 0, X_{300} = 1, Y = 0$:	0
$X_1 = 0, X_2 = 0, \ldots, X_{299} = 1, X_{300} = 0, Y = 0$:	1
…	
$X_1 = 0, X_2 = 0, \ldots, X_{299} = 0, X_{300} = 0, Y = 1$:	2
$X_1 = 0, X_2 = 0, \ldots, X_{299} = 0, X_{300} = 1, Y = 1$:	1
$X_1 = 0, X_2 = 0, \ldots, X_{299} = 1, X_{300} = 0, Y = 1$:	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)

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

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

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

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 $\boldsymbol{x}$
- $\boldsymbol{X} = (X_1, X_2, \dots, X_{300})$: 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

2^{300}

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

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

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

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

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

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)

$\hat{P}(\boldsymbol{X}|Y) \rightarrow \hat{P}(X_1, X_2, \ldots, X_m|Y)$

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

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

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

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

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

Assumption:

$X_1, \dots, X_m$ are conditionally independent given Y.

$$\hat{P}(X|Y) = \hat{P}(X_1, X_2, \dots, X_{300} | Y)$$
$$= \prod_{i=1}^{m} \hat{P}(X_i|Y)$$

Naïve Bayes Assumption

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

Naïve Bayes Classifier

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

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

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

Training

for $i = 1, \dots, m$: $\hat{P}(X_i = 1|Y = 0)$, $\hat{P}(X_i = 1|Y = 1)$

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

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

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

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

Use MLE or Laplace (MAP)

Testing

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

(for numeric stability)

23: Naïve Bayes (live)

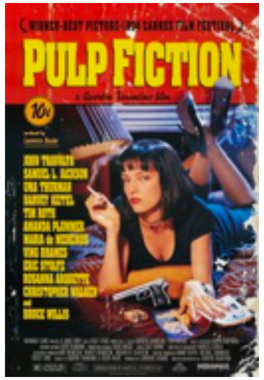
Lisa Yan

May 29, 2020

Classification terminology check

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

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



		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: like movie
0: dislike movie

Classification terminology check

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

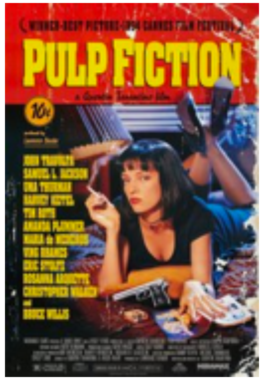
i: i-th user
j: movie j

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

Movie 1

Movie 2

…



Movie m

Output

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

1. $x^{(1)}$ example i: ~~…~~ i
2. $y^{(1)}$ label for example/observation i
3. $(x^{(2)}, y^{(2)})$
4. $x_2^{(n)}$

Multinomial MLE and MAP

Review

Model: Multinomial with m outcomes: p_i probability of outcome i

Observe: n_i = # of trials with outcome i. Total of $\sum_{i=1}^{m} n_i$ trials

MLE

$$p_i = \frac{n_i}{\sum_{i=1}^{m} n_i}$$

MAP with Laplace smoothing (Laplace estimate)

$$p_i = \frac{n_i + 1}{\sum_{i=1}^{m} n_i + m}$$

“Brute Force Bayes” classifier

(strawman)
Review

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

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

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

Testing

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

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

NETFLIX

and Learn

Brute Force Bayes for TV shows

(strawman)
Review

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"

Brute Force Bayes for TV shows

(strawman)
Review

1. What probabilities do we need to estimate? (training)

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

$\boldsymbol{X} = (X_1, X_2)$ binary vector
$Y \in \{0,1\}$

2. How would we estimate $\hat{P}(X_1 = 0, X_2 = 1|Y = 0)$? (training)

3. If $\boldsymbol{X} = (X_1, X_2, \dots, X_m)$ (binary vector of m features), how many probabilities do we need to estimate?

Brute Force Bayes for TV shows

(strawman)
Review

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

$\boldsymbol{X} = (X_1, X_2)$ binary vector
$Y \in \{0,1\}$

1. What probabilities do we need to estimate?

   $2 + 2^3$

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

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

2. How would we estimate $\hat{P}(X_1 = 0, X_2 = 1 | Y = 0)$?

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

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

3. If $\boldsymbol{X} = (X_1, X_2, \dots, X_m)$ (binary vector of m features), how many probabilities do we need to estimate?

   $2^{m+1} + 2$

The Naïve Bayes assumption

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

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

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

"Brute Force Bayes"

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

Naïve Bayes Assumption

Assumption:

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

Naïve Bayes Model is a Bayesian Network

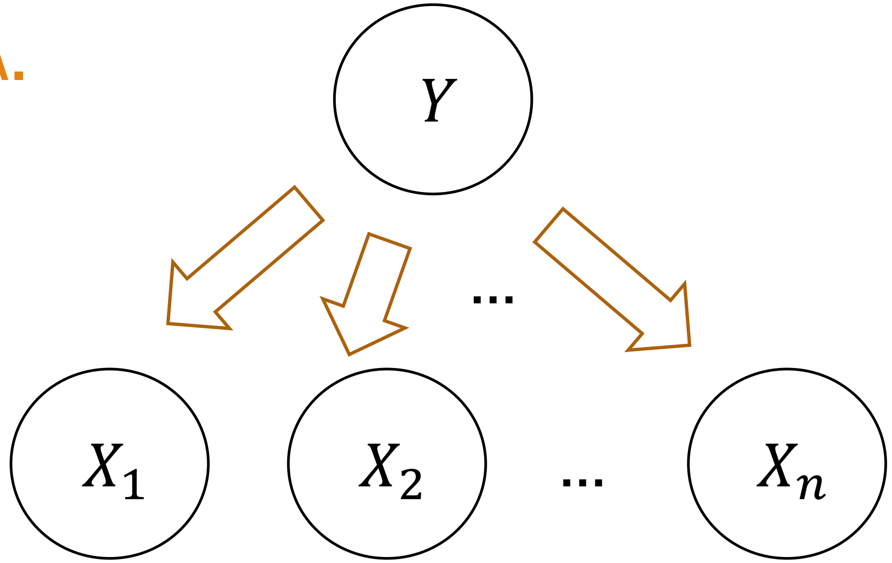
Naïve Bayes Assumption

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

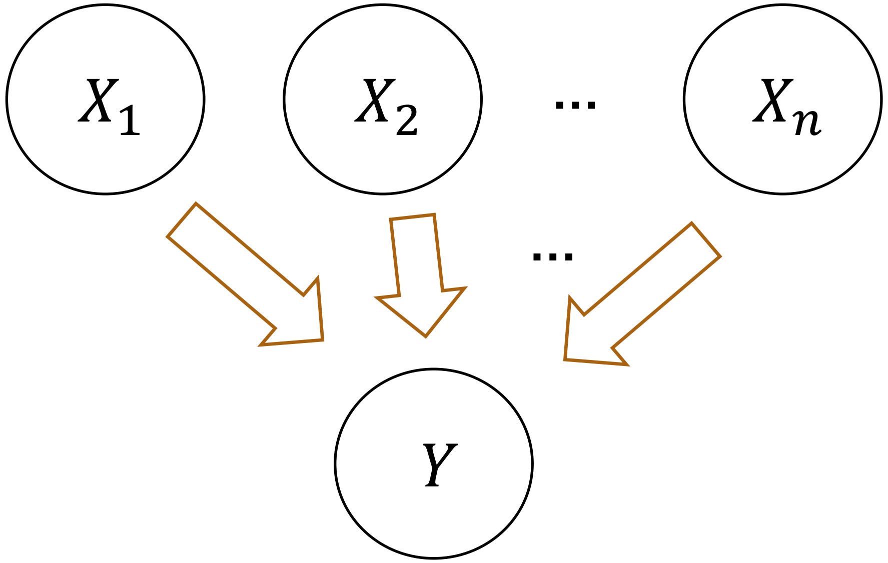
$X_1, \ldots, X_m$ are conditionally independent given Y

Which Bayesian Network encodes this conditional independence?

A.



B.

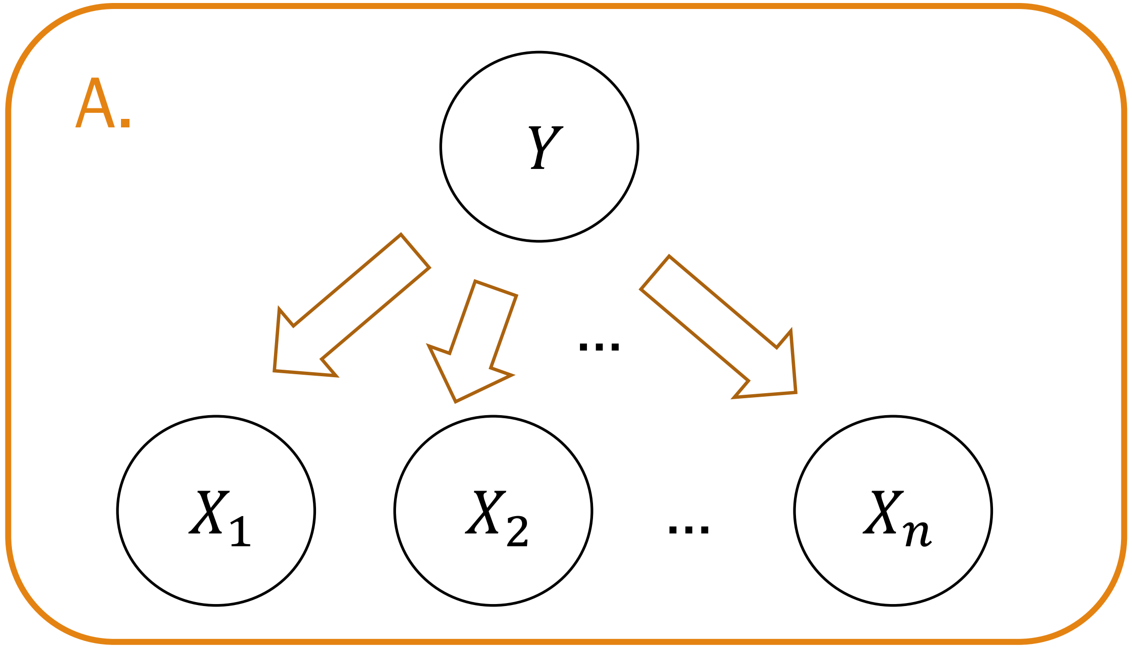


Naïve Bayes Model is a Bayesian Network

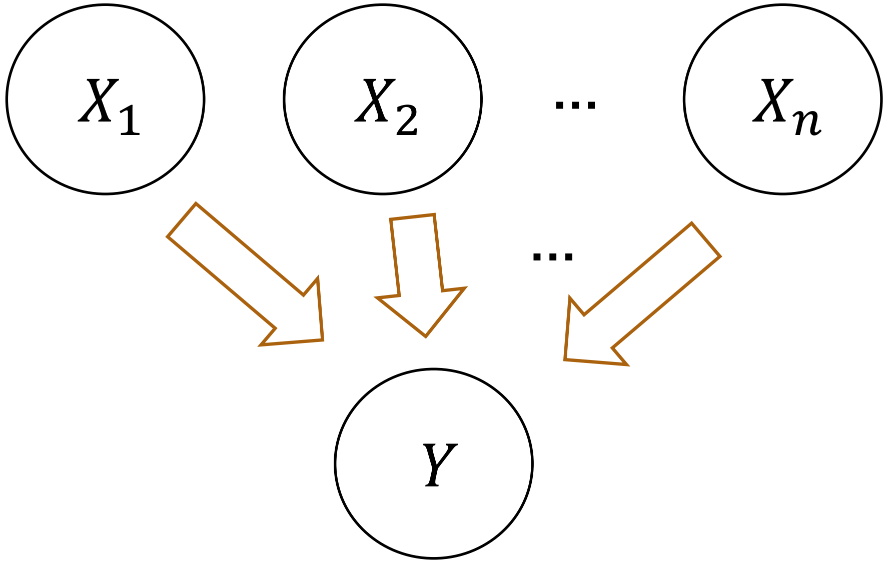
Naïve Bayes Assumption

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

Which Bayesian Network encodes this conditional independence?



B.



X_i are conditionally independent given parent Y

NETFLIX

and Learn

Naïve Bayes for TV shows

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"

Naïve Bayes for TV shows

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

$\boldsymbol{X} = (X_1, X_2)$ binary vector

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

1. What probabilities do we need to estimate?

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

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

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

   2 + 4·m

2. How would we estimate $\hat{P}(X_1 = 0, X_2 = 1|Y = 0)$?

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

3. If $\boldsymbol{X} = (X_1, X_2, \ldots, X_m)$ (binary vector of m features), how many probabilities do we need to estimate?

   $4 \cdot m + 2$ vs $2 \cdot 2^m + 2$

Ex 1. Naïve Bayes Classifier (MLE)

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

Training

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

Use MLE or Laplace (MAP)

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

Testing

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

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 \ X_1	0	1
0	3	10
1	4	13

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

Training: Naïve Bayes for TV shows (MLE)

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

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

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

user $(X_1=1, X_2=0, Y=0)$

Predict Y: "likes Pokémon"

Y \ X_1	0	1
0	3	10
1	4	13

Y \ X_2	0	1	
0	5	8	← 13
1	7	10	← 17

Training data counts

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	$\hat{P}(X_1=0|Y=0) = 3/13$	$10/13$
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 \ 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 \ X_1	0	1
0	0.23	0.77
1	0.24	0.76

(from last slide)

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

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

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 \ X_1	0	1
0	0.23	0.77
1	0.24	0.76

Y \ 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)

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

Training

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

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

Use MLE or Laplace (MAP)

Testing

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

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 \ X_1	0	1
0	0.23	0.77
1	0.24	0.76

Y \ 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 $\hat{Y} = 1$

Interlude for jokes/announcements

Announcements

Problem Set 6

Out:	later today
Due:	Wednesday 6/10
Covers:	through next Wed.

No late days or on-time bonus

What topics do you want to see next week?
https://forms.gle/AZy7R7CNkNsLZKq2A

Interesting probability news

Paradoxes of Probability & Statistical Strangeness

- Simpson's Paradox
- Base rate fallacy
- Will Rogers Paradox
- Berkson's Paradox
- Multiple comparisons fallacy

https://scitechdaily.com/paradoxes-of-probability-statistical-strangeness/

CS109 Current Events Spreadsheet

Ex 2. Naïve Bayes Classifier (MAP)

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

Training

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

Use MLE or **Laplace (MAP)**

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

Testing

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

(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 \ 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 $\hat{P}(X_i|Y)$?

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

A. $\frac{\#(X_i=x, Y=y)}{\#(Y=y)}$

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

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

D. other

$X_i = 0, Y = y$

$X_i = 1, Y = y$

Note: $\hat{P}(X_i = x | Y = y), \forall i = 1, \ldots, m$ separate estimation problems

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 $\hat{P}(X_i|Y)$?

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

A. $\frac{\#(X_i=x, Y=y)}{\#(Y=y)}$

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

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

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

Training data

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

$0.27 = \frac{3+1}{13+2}$, $0.73 = \frac{10+1}{15}$

$0.26 = \frac{5}{19}$, $0.74 = \frac{14}{19}$

$0.40 = \frac{6}{15}$

In practice:

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

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

-dim features — size of our training set

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

Training

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

… $0|Y = 0)$, Laplace (MAP)

$\hat{P}$ …

What changes are necessary?

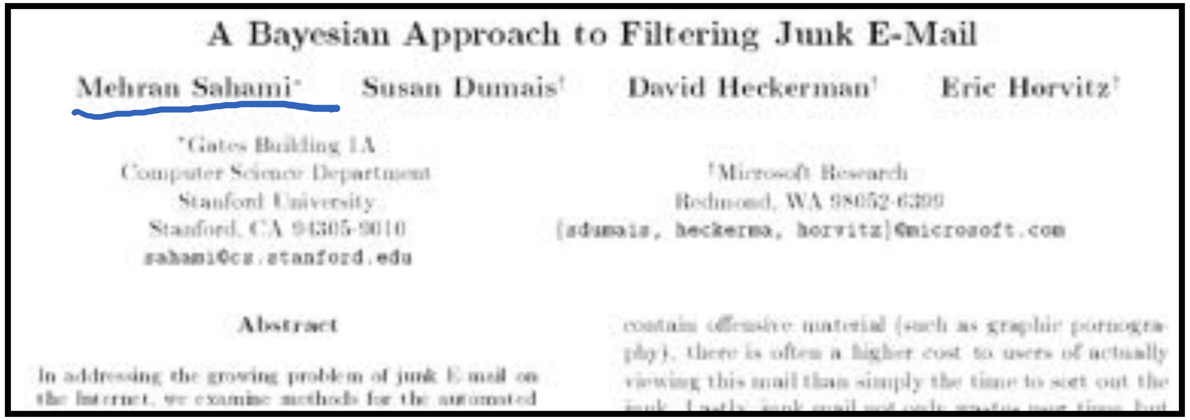
Testing

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

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 junk.

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

$X_i = 1$ if word i appeared in document

Output $Y = 1$ if email is spam

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

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

$\boldsymbol{x}^{(i)} = \left(x_1^{(i)}, x_2^{(i)}, \ldots, x_m^{(i)}\right)$ 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 $\hat{P}(X_i|Y)$?

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

Training: Naïve Bayes Email classification

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

$\boldsymbol{x}^{(i)} = \left(x_1^{(i)}, x_2^{(i)}, \dots, x_m^{(i)}\right)$ 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 $\hat{P}(X_i|Y)$?

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)

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

Training

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

Use MLE or **Laplace (MAP)**

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

Testing

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

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?

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

Testing: Naïve Bayes Email classification

For a new email:

- Generate $\boldsymbol{X} = (X_1, X_2, \ldots, X_m)$
- 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?

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

Will probably lead to underflow!

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

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

Training

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

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

Use sums of log-probabilities for numerical stability.

$$\log \left[\left[\prod_{i=1}^{m} \hat{P}(X_i|Y) \right] \hat{P}(Y) \right]$$

Testing

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

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:

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

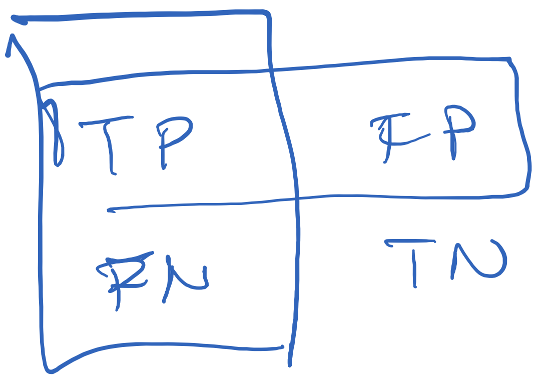
$$\textbf{recall} = \frac{(\text{\# correctly predicted class } Y)}{(\text{\# real class } Y \text{ messages})}$$

	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%

wiki: precision & recall

classifier: $\hat{Y} = 1$, $\hat{Y} = 0$

"true labels": $Y = 1$, $Y = 0$



precision: $\dfrac{TP}{TP + FP}$

recall: $\dfrac{TP}{TP + FN}$