EECS 391
Intro to AI

Learning from Examples

L16 Tue Nov 9

Classifying Uncertain Data

- Consider the credit risk data again.
- Suppose now we want to ***learn*** the best classification P(D | J,M) from the data?
- Instead of a yes or no answer want some estimate of how strongly we *believe* a loan applicant is a credit risk.
- This might be useful if we want some flexibility in adjusting our decision criteria.
  - Eg, suppose we're willing to take more risk if times are good.
  - Or, if we want to examine case we believe are higher risks more carefully.

Predicting credit risk

<2 years at current job?	missed payments?	defaulted?
N	N	N
Y	N	Y
N	N	N
N	N	N
N	Y	Y
Y	N	N
N	Y	N
N	Y	Y
Y	N	N
Y	N	N
⋮	⋮	⋮

Pick your poison: Mushrooms

- Or suppose we wanted to know how *likely* a mushroom was safe to eat?
- One approach is to consult a guide, and go by the listed criteria, but does that allow us to place any certainty on the decision?
- (btw: never eat wild mushrooms without an expert guide, it really is a serious risk)

"Death Cap"

Mushroom data

	EDIBLE?	CAP-SHAPE	CAP-SURFACE	...
1	edible	flat	fibrous	...
2	poisonous	convex	smooth	...
3	edible	flat	fibrous	...
4	edible	convex	scaly	...
5	poisonous	convex	smooth	...
6	edible	convex	fibrous	...
7	poisonous	flat	scaly	...
8	poisonous	flat	scaly	...
9	poisonous	convex	fibrous	...
10	poisonous	convex	fibrous	...
11	poisonous	flat	smooth	...
12	edible	convex	smooth	...
13	poisonous	knobbed	scaly	...
14	poisonous	flat	smooth	...
15	poisonous	flat	fibrous	...
	⋮	⋮	⋮	⋱

Bayesian classification for more complex models

- Recall the class conditional probability:

$$\begin{aligned} p(C_k|\mathbf{x}) &= \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})} \\ &= \frac{p(\mathbf{x}|C_k)p(C_k)}{\sum_k p(\mathbf{x}|C_k)p(C_k)} \end{aligned}$$

- How do we define the data likelihood, $p(\mathbf{x}|C_k)$

  ie the probability of $\mathbf{x}$ given class C_k

Defining a probabilistic classification model

- How would we define credit risk problem?
  - Class:

    C_1 = "defaulted"

    C_2 = "didn't default"
  - Data:

    $\mathbf{x}$ = { "<2 years", "missed payments" }
  - Prior (from data):

    $p(C_1) = 3/10$; $p(C_2) = 7/10$;
  - Likelihood:

    $p(x_1, x_2 | C_1) = ?$

    $p(x_1, x_2 | C_2) = ?$
  - How would we determine these?

Predicting credit risk

<2 years at current job?	missed payments?	defaulted?
N	N	N
Y	N	Y
N	N	N
N	N	N
N	Y	Y
Y	N	N
N	Y	N
N	Y	Y
Y	N	N
Y	N	N
⋮	⋮	⋮

Defining a probabilistic model by counting

- The "prior" is obtained by counting number of classes in the data:

$$p(C_k = k) = \frac{\text{Count}(C_k = k)}{\#\ \text{records}}$$

- The likelihood is obtained the same way:

$$p(\mathbf{x} = \mathbf{v} | C_k) = \frac{\text{Count}(\mathbf{x} = \mathbf{v} \wedge C_k = k)}{\text{Count}(C_k = k)}$$

$$p(x_1 = v_1, \ldots, x_N = v_N | C_k = k) = \frac{\text{Count}(x_1 = v_1, \ldots \wedge x_N = v_N, \wedge C_k = k)}{\text{Count}(C_k = k)}$$

- This is the maximum likelihood estimate (MLE) of the probabilities

Defining a probabilistic classification model

- Determining the likelihood:

  $p(x_1, x_2 | C_1) = ?$

  $p(x_1, x_2 | C_2) = ?$

- Simple approach: look at counts in data

x_1 <2 years at current job?	x_2 missed payments?	C_1 did default	C_2 did not default
N	N		
N	Y		
Y	N		
Y	Y		

Predicting credit risk

<2 years at current job?	missed payments?	defaulted?
N	N	N
Y	N	Y
N	N	N
N	N	N
N	Y	Y
Y	N	N
N	Y	N
N	Y	Y
Y	N	N
Y	N	N
⋮	⋮	⋮

Defining a probabilistic classification model

- Determining the likelihood:

  $p(x_1, x_2 | C_1) = ?$

  $p(x_1, x_2 | C_2) = ?$

- Simple approach: look at counts in data

x_1 <2 years at current job?	x_2 missed payments?	C_1 did default	C_2 did not default
N	N	0/3	3/3
N	Y		
Y	N		
Y	Y		

Predicting credit risk

<2 years at current job?	missed payments?	defaulted?
N	N	N
Y	N	Y
N	N	N
N	N	N
N	Y	Y
Y	N	N
N	Y	N
N	Y	Y
Y	N	N
Y	N	N
⋮	⋮	⋮

Defining a probabilistic classification model

- Determining the likelihood:

  $p(x_1, x_2 | C_1) = ?$

  $p(x_1, x_2 | C_2) = ?$

- Simple approach: look at counts in data

x_1 <2 years at current job?	x_2 missed payments?	C_1 did default	C_2 did not default
N	N	0/3	3/3
N	Y	2/3	1/3
Y	N		
Y	Y		

Predicting credit risk

<2 years at current job?	missed payments?	defaulted?
N	N	N
Y	N	Y
N	N	N
N	N	N
N	Y	Y
Y	N	N
N	Y	N
N	Y	Y
Y	N	N
Y	N	N
⋮	⋮	⋮

Defining a probabilistic classification model

- Determining the likelihood:

  $p(x_1, x_2 | C_1) = ?$

  $p(x_1, x_2 | C_2) = ?$

- Simple approach: look at counts in data

x_1 <2 years at current job?	x_2 missed payments?	C_1 did default	C_2 did not default
N	N	0/3	3/3
N	Y	2/3	1/3
Y	N	1/4	3/4
Y	Y		

Predicting credit risk

<2 years at current job?	missed payments?	defaulted?
N	N	N
Y	N	Y
N	N	N
N	N	N
N	Y	Y
Y	N	N
N	Y	N
N	Y	Y
Y	N	N
Y	N	N
⋮	⋮	⋮

Defining a probabilistic classification model

- Determining the likelihood:

  $p(x_1, x_2 | C_1) = ?$

  $p(x_1, x_2 | C_2) = ?$

- Simple approach: look at counts in data

x_1 <2 years at current job?	x_2 missed payments?	C_1 did default	C_2 did not default
N	N	0/3	3/3
N	Y	2/3	1/3
Y	N	1/4	3/4
Y	Y	0/0	0/0

Predicting credit risk

<2 years at current job?	missed payments?	defaulted?
N	N	N
Y	N	Y
N	N	N
N	N	N
N	Y	Y
Y	N	N
N	Y	N
N	Y	Y
Y	N	N
Y	N	N
⋮	⋮	⋮

Defining a probabilistic classification model

- Determining the likelihood:

  $p(x_1, x_2 | C_1) = ?$

  $p(x_1, x_2 | C_2) = ?$

- Simple approach: look at counts in data

x_1 <2 years at current job?	x_2 missed payments?	C_1 did default	C_2 did not default
N	N	0/3	3/3
N	Y	2/3	1/3
Y	N	1/4	3/4
Y	Y	0/0	0/0

What do we do about these?

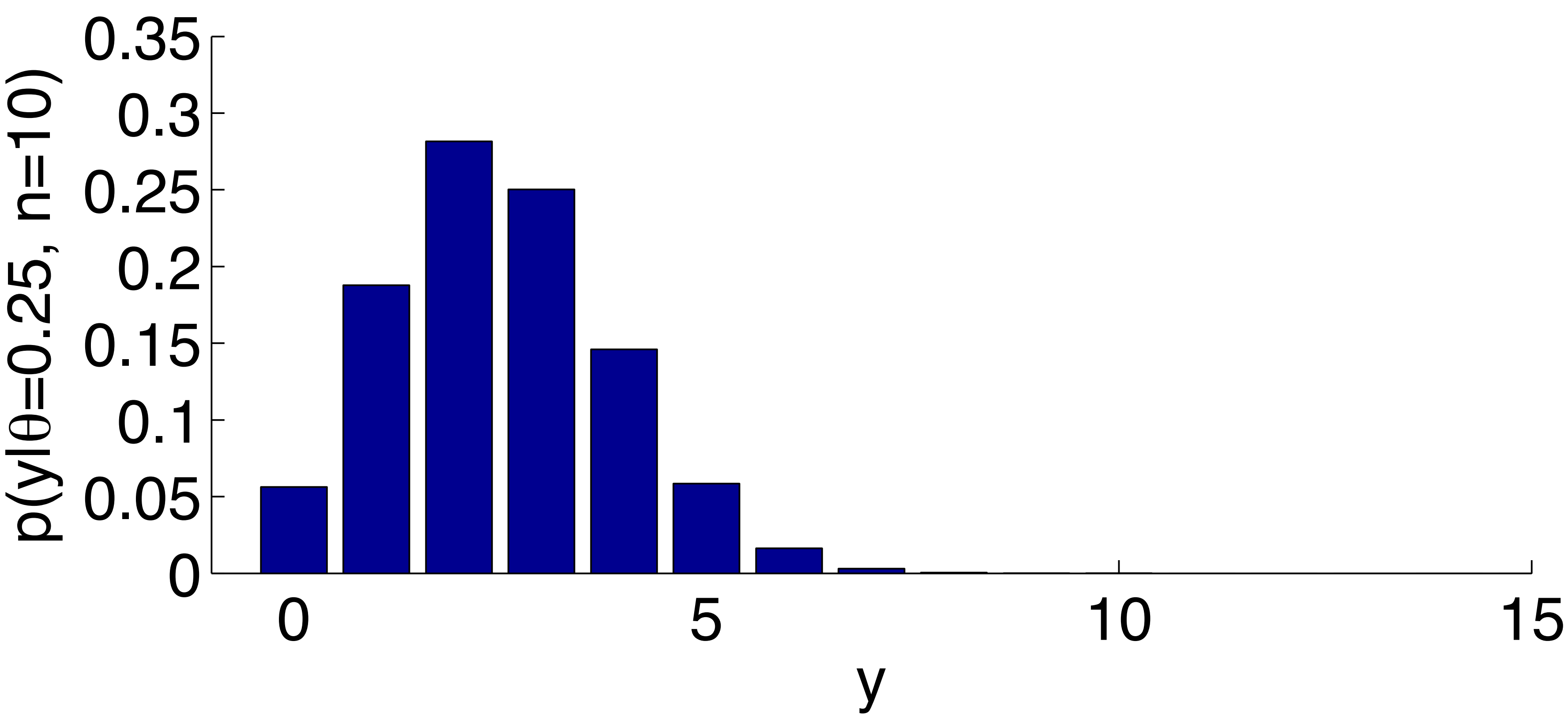
Predicting credit risk

<2 years at current job?	missed payments?	defaulted?
N	N	N
Y	N	Y
N	N	N
N	N	N
N	Y	Y
Y	N	N
N	Y	N
N	Y	Y
Y	N	N
Y	N	N
⋮	⋮	⋮

Being (proper) Bayesians: Recall our coin-flipping example

- In Bernoulli trials, each sample is either 1 (e.g. heads) with probability θ, or 0 (tails) with probability $1 - \theta$.
- The *binomial distribution* specifies probability of total #heads, y, out of n trials:

$$p(y|\theta, n) = \binom{n}{y} \theta^y (1-\theta)^{n-y}$$



Applying Bayes' rule

- Given n trials with k heads, what do we know about θ?
- We can apply Bayes' rule to see how our knowledge changes as we acquire new observations:

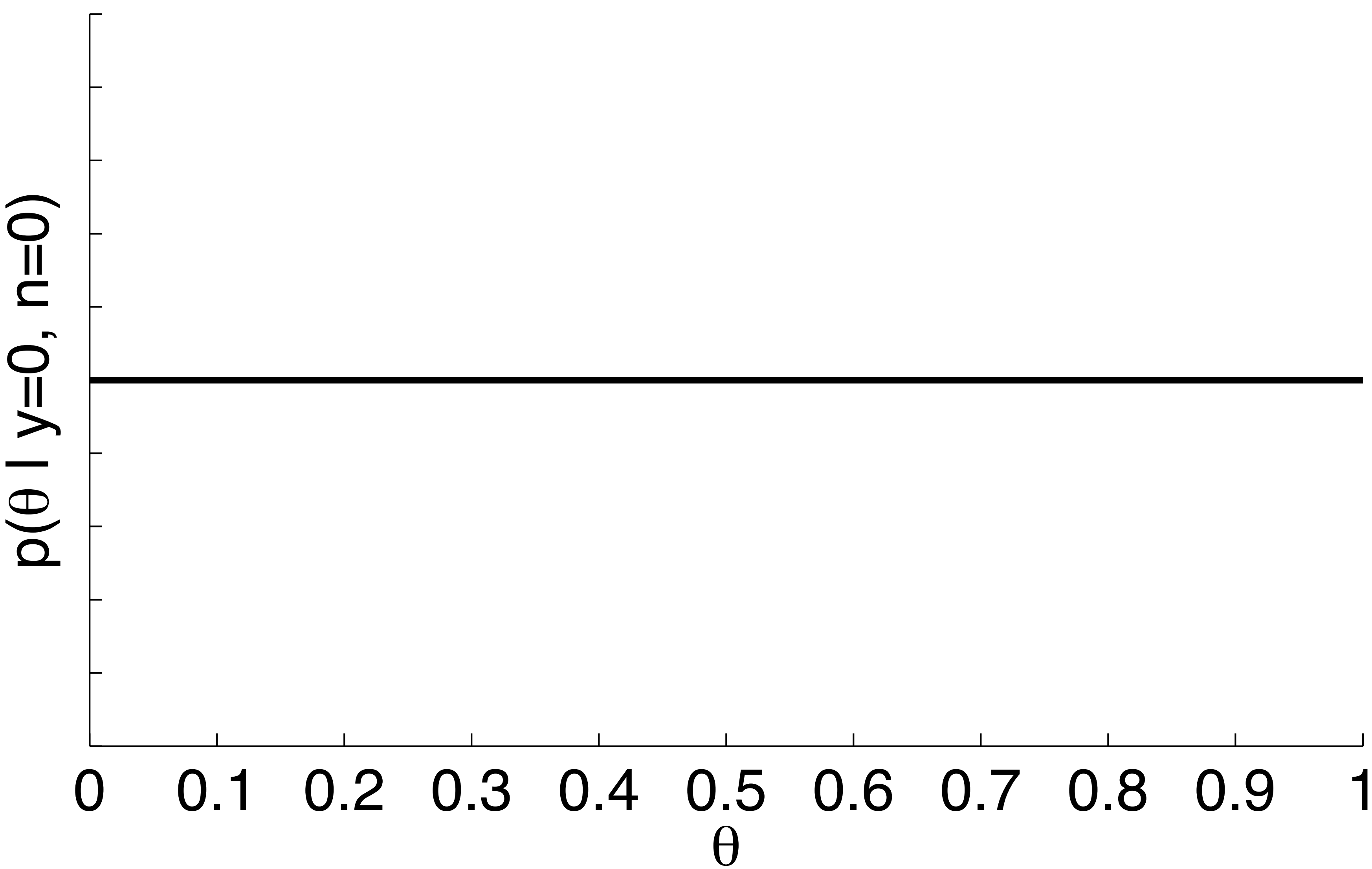
$$\underbrace{p(\theta|y,n)}_{\text{posterior}} = \frac{\overbrace{p(y|\theta,n)}^{\text{likelihood}}\overbrace{p(\theta|n)}^{\text{prior}}}{\underbrace{p(y|n)}_{\text{normalizing constant}}} \quad p(y|n) = \int p(y|\theta,n)p(\theta|n)d\theta$$

- We know the likelihood, what about the prior?
- Uniform on [0, 1] is a reasonable assumption, i.e. "we don't know anything".
- What is the form of the posterior?
- In this case, the posterior is just proportional to the likelihood:

$$p(\theta|y,n) \propto \binom{n}{y}\theta^y(1-\theta)^{n-y}$$

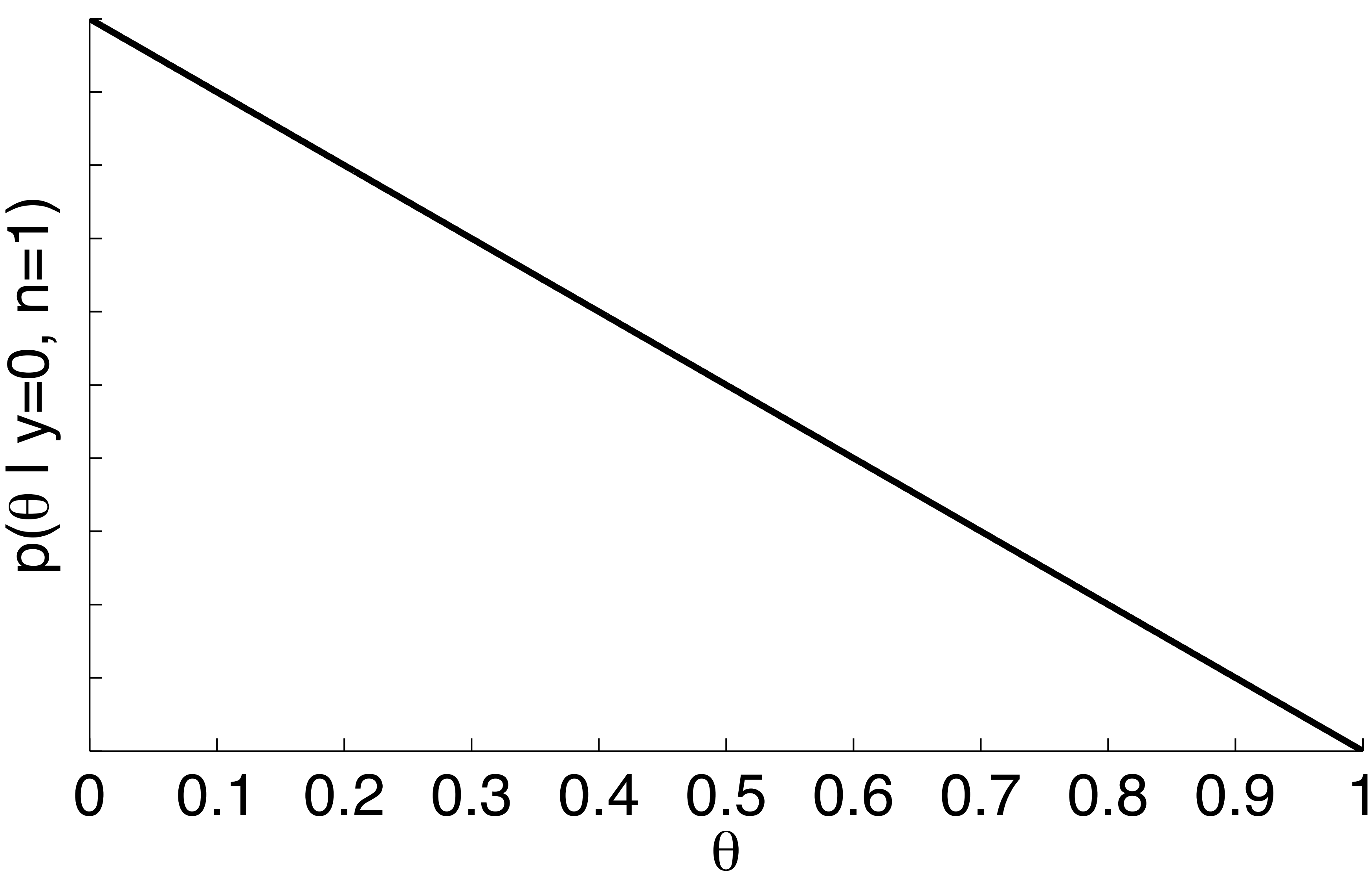
Evaluating the posterior

- What do we know initially, before observing any trials?



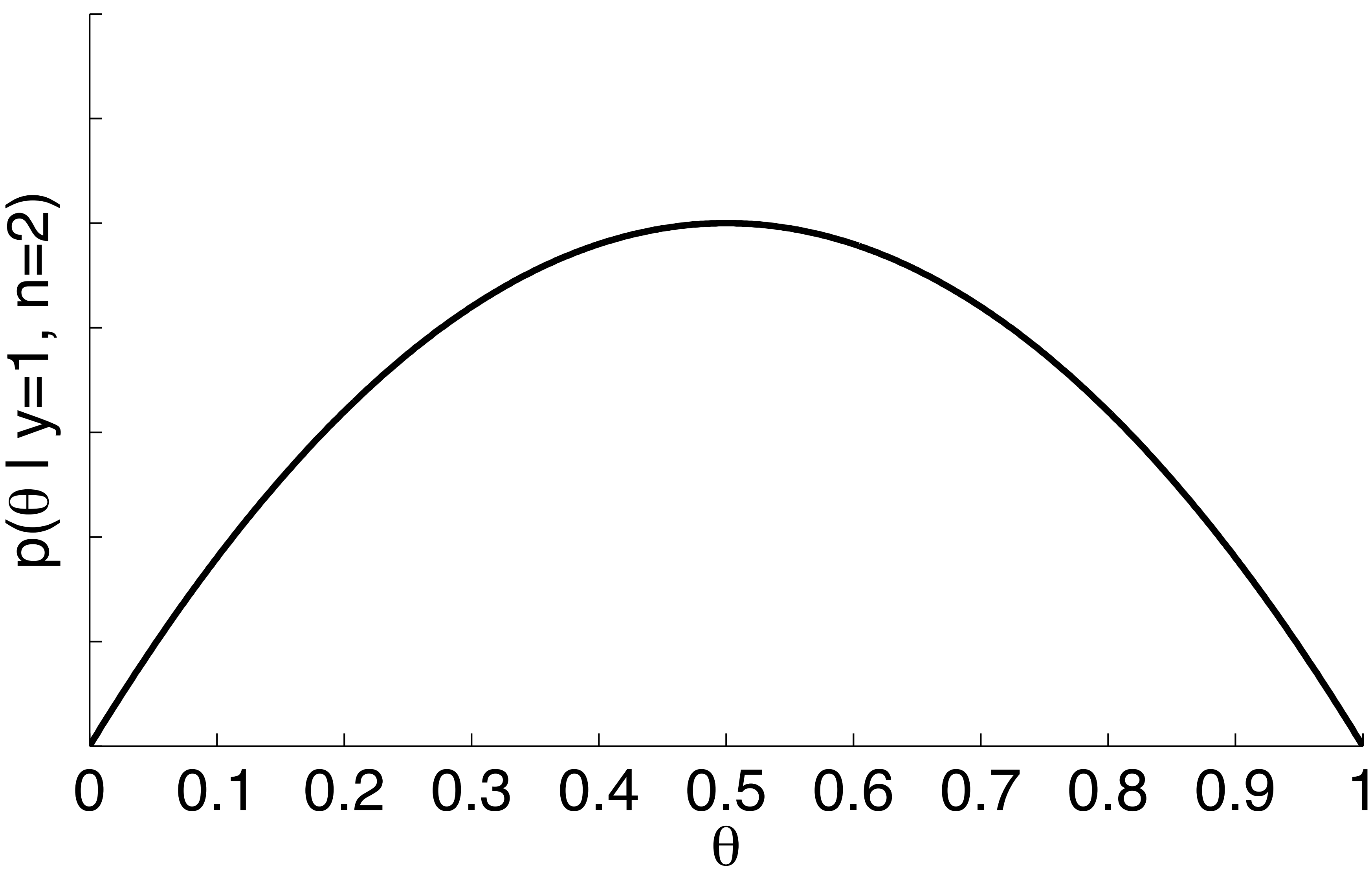
Coin tossing

- What is our belief about θ after observing one “tail” ?



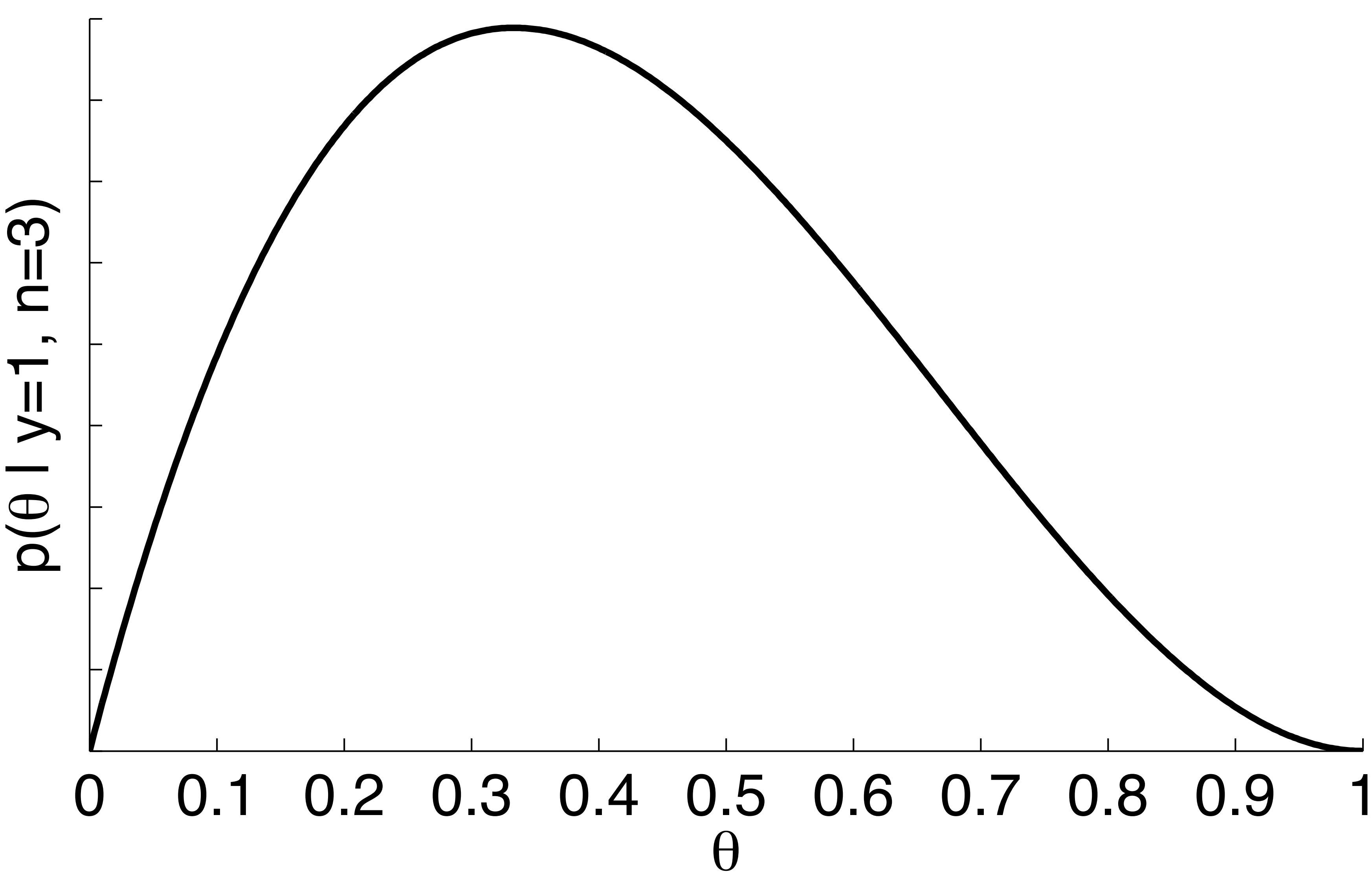
Coin tossing

- Now after two trials we observe 1 head and 1 tail.



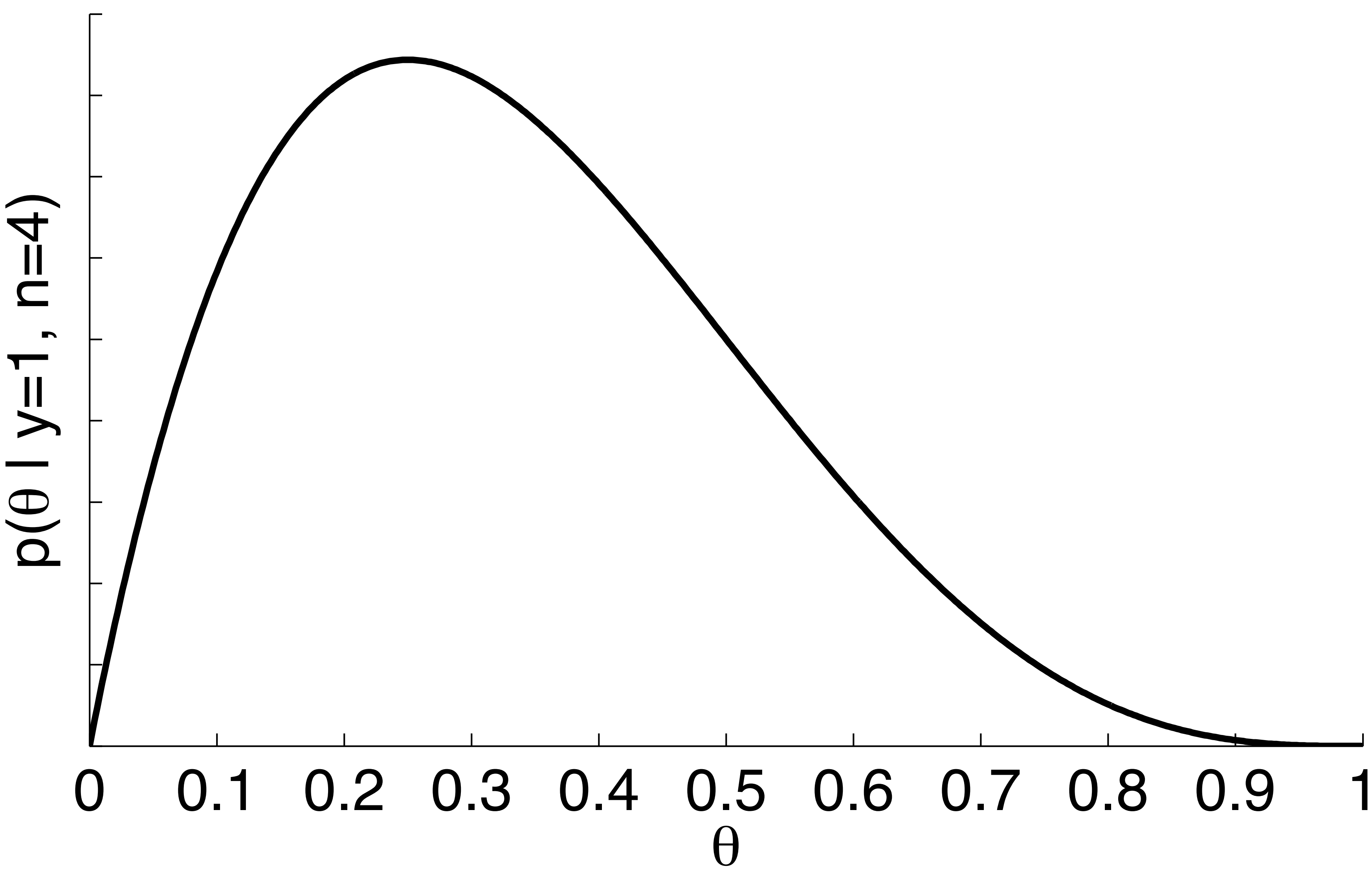
Coin tossing

- 3 trials: 1 head and 2 tails.



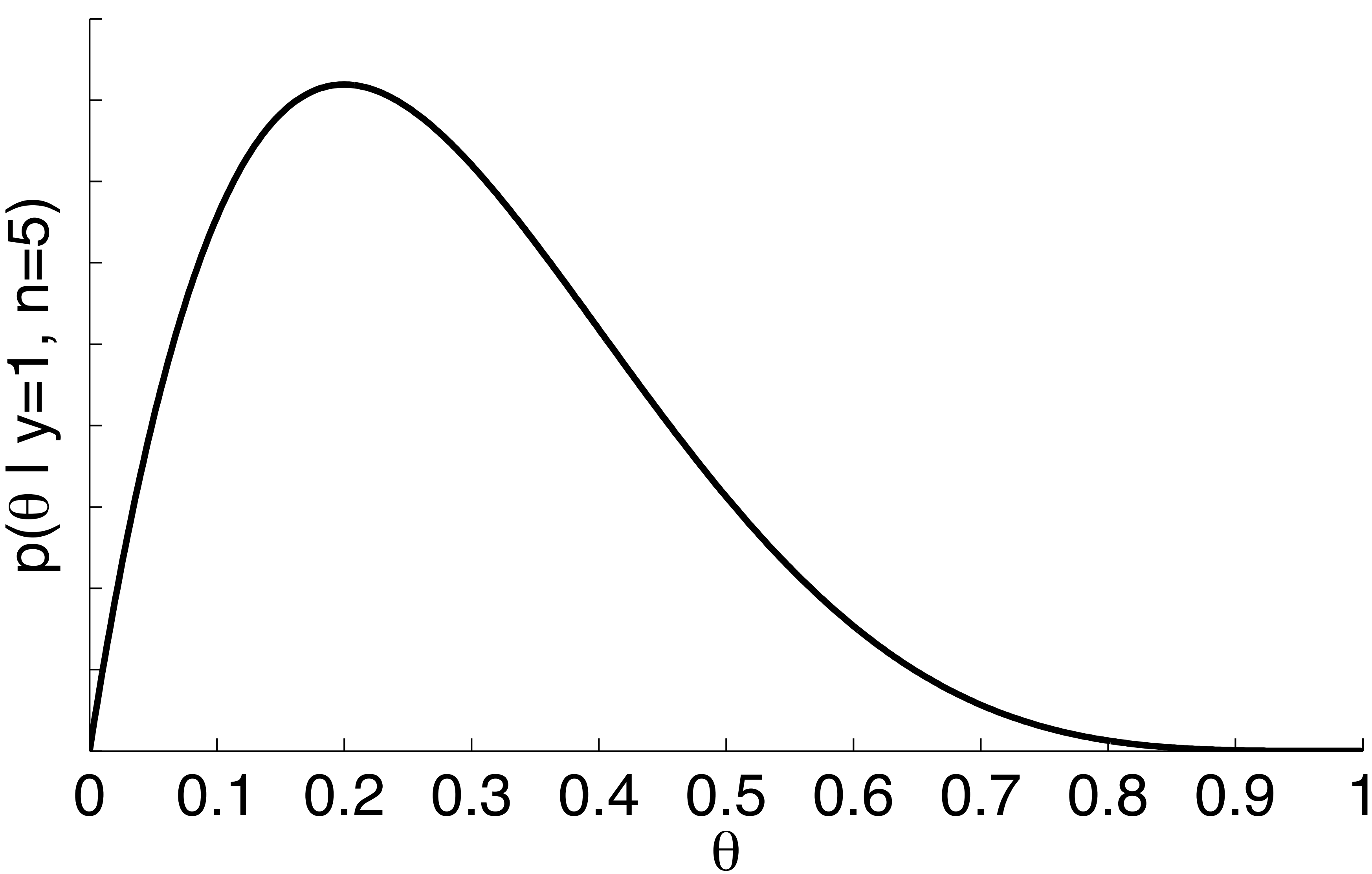
Coin tossing

- 4 trials: 1 head and 3 tails.



Coin tossing

- 5 trials: 1 head and 4 tails.



Evaluating the normalizing constant

- To get proper probability density functions, we need to evaluate p(y|n):

$$p(\theta|y,n) = \frac{p(y|\theta,n)p(\theta|n)}{p(y|n)}$$

- Bayes in his original paper in 1763 showed that:

$$\begin{aligned} p(y|n) &= \int_0^1 p(y|\theta,n)p(\theta|n)d\theta \\ &= \frac{1}{n+1} \end{aligned}$$

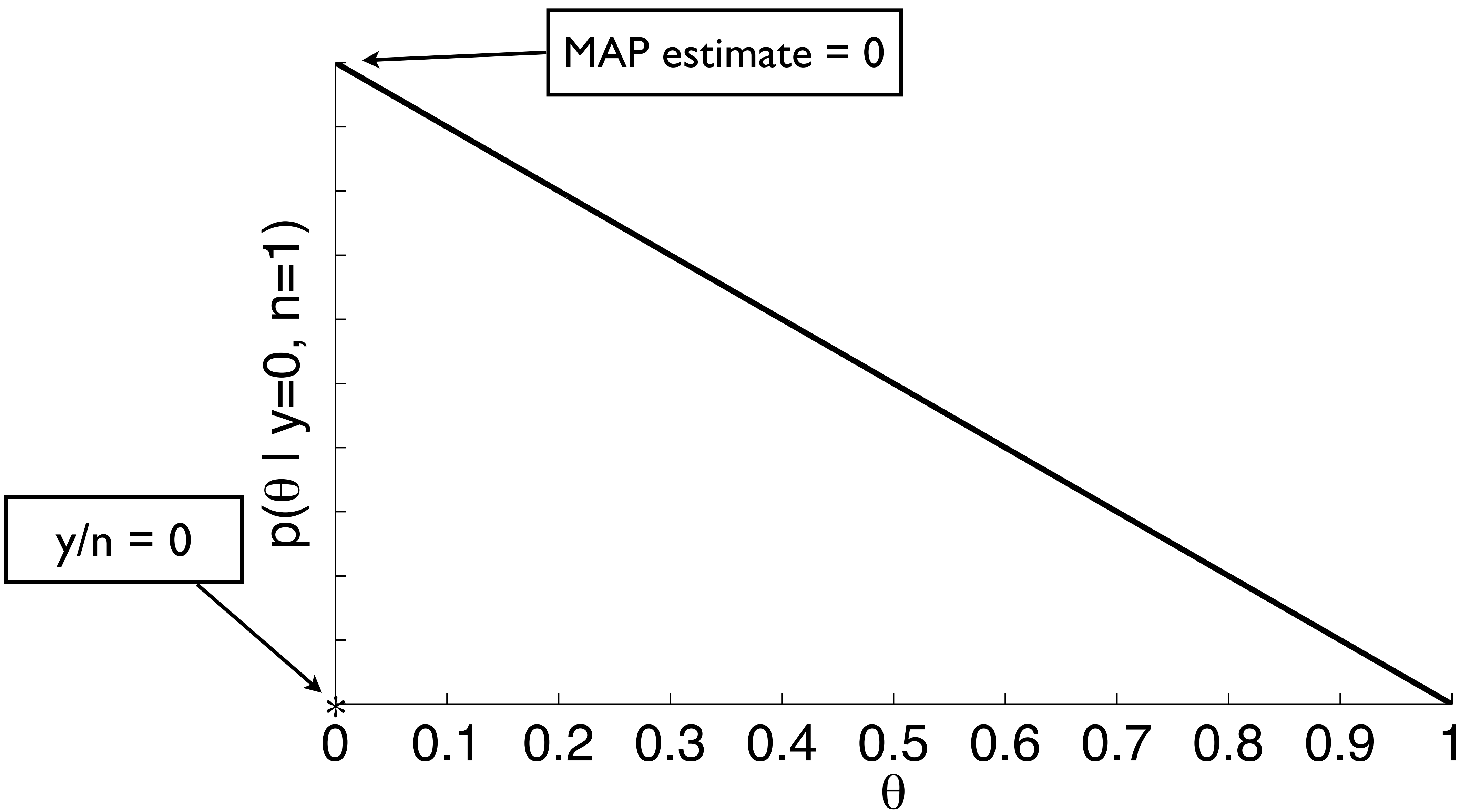
$$\Rightarrow p(\theta|y,n) = \binom{n}{y}\theta^y(1-\theta)^{n-y}(n+1)$$

The ratio estimate

- What about after just one trial: 0 heads and 1 tail?
- MAP and ratio estimate would say 0.

Does this make sense?

- What would a better estimate be?



The expected value estimate

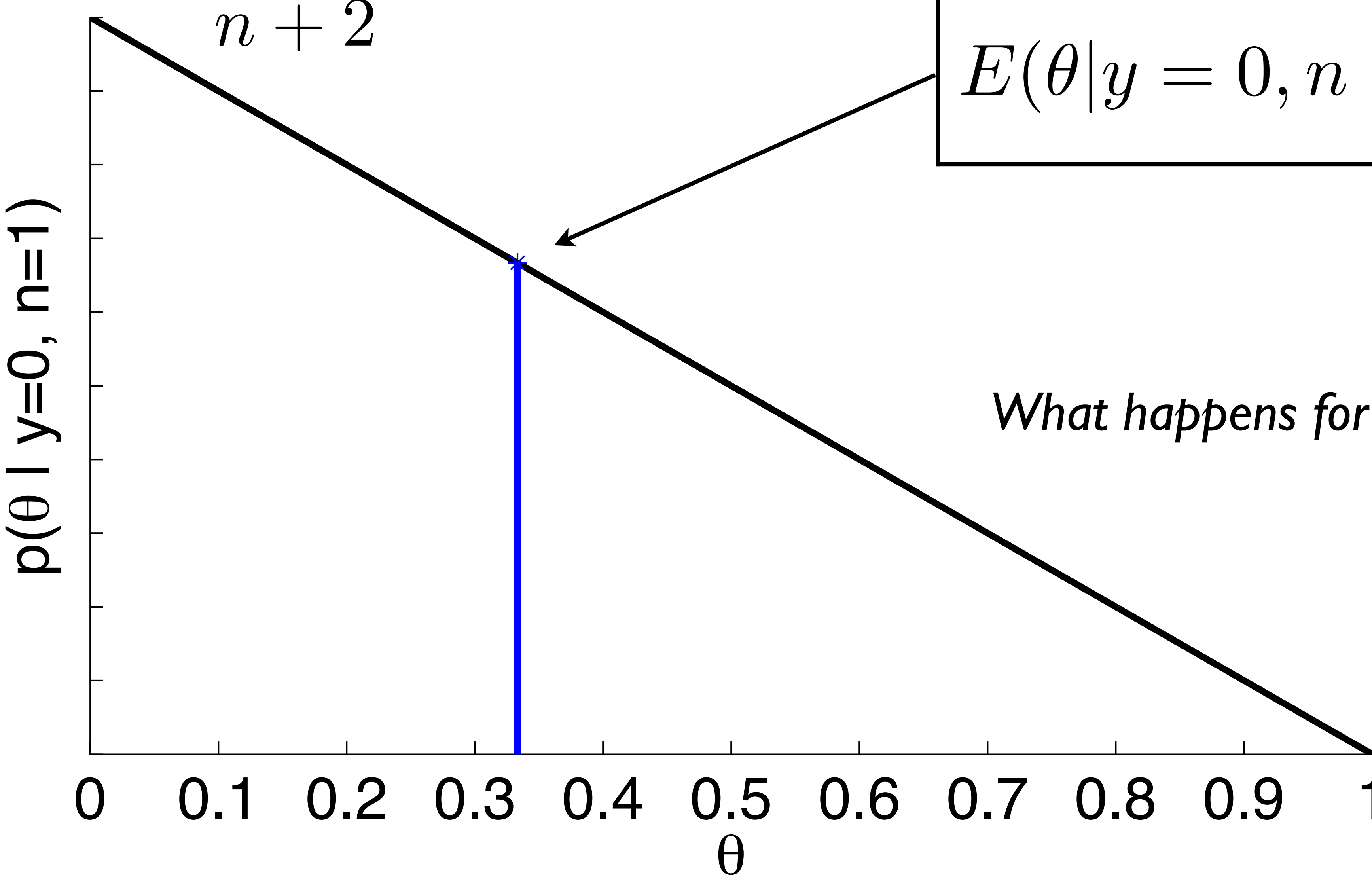
- The expected value of a pdf is:

$$E(\theta|y,n) = \int_0^1 \theta p(\theta|y,n)d\theta$$

$$= \frac{y+1}{n+2}$$

This is called
"smoothing" or
"regularization"

$$E(\theta|y=0,n=1) = \frac{1}{3}$$



What happens for zero trials?

On to the mushrooms!

	EDIBLE?	CAP-SHAPE	CAP-SURFACE	CAP-COLOR	ODOR	STALK-SHAPE	POPULATION	HABITAT	• • •
1	edible	flat	fibrous	red	none	tapering	several	woods	• • •
2	poisonous	convex	smooth	red	foul	tapering	several	paths	• • •
3	edible	flat	fibrous	brown	none	tapering	abundant	grasses	• • •
4	edible	convex	scaly	gray	none	tapering	several	woods	• • •
5	poisonous	convex	smooth	red	foul	tapering	several	woods	• • •
6	edible	convex	fibrous	gray	none	tapering	several	woods	• • •
7	poisonous	flat	scaly	brown	fishy	tapering	several	leaves	• • •
8	poisonous	flat	scaly	brown	spicy	tapering	several	leaves	• • •
9	poisonous	convex	fibrous	yellow	foul	enlarging	several	paths	• • •
10	poisonous	convex	fibrous	yellow	foul	enlarging	several	woods	• • •
11	poisonous	flat	smooth	brown	spicy	tapering	several	woods	• • •
12	edible	convex	smooth	yellow	anise	tapering	several	woods	• • •
13	poisonous	knobbed	scaly	red	foul	tapering	several	leaves	• • •
14	poisonous	flat	smooth	brown	foul	tapering	several	leaves	• • •
15	poisonous	flat	fibrous	gray	foul	enlarging	several	woods	• • •
16	edible	sunken	fibrous	brown	none	enlarging	solitary	urban	• • •
17	poisonous	flat	smooth	brown	foul	tapering	several	woods	• • •
18	poisonous	convex	smooth	white	foul	tapering	scattered	urban	• • •
19	poisonous	flat	scaly	yellow	foul	enlarging	solitary	paths	• • •
20	edible	convex	fibrous	gray	none	tapering	several	woods	• • •
	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •

The scaling problem

$$p(\mathbf{x} = \mathbf{v}|C_k) = \frac{\text{Count}(\mathbf{x} = \mathbf{v} \wedge C_k = k)}{\text{Count}(C_k = k)}$$

$$p(x_1 = v_1, \ldots, x_N = v_N|C_k = k) = \frac{\text{Count}(x_1 = v_1, \ldots \wedge x_N = v_N, \wedge C_k = k)}{\text{Count}(C_k = k)}$$

- The prior is easy enough.
- But for the likelihood, the table is huge!

Mushroom attributes and values

# values	attributes
2	EDIBLE: edible poisonous
6	CAP-SHAPE: bell conical convex flat knobbed sunken
4	CAP-SURFACE: fibrous grooves scaly smooth
10	CAP-COLOR: brown buff cinnamon gray green pink purple red white yellow
2	BRUISES: bruises no
9	ODOR: almond anise creosote fishy foul musty none pungent spicy
2	GILL-ATTACHMENT: attached free
2	GILL-SPACING: close crowded
2	GILL-SIZE: broad narrow
12	GILL-COLOR: black brown buff chocolate gray green orange pink purple red white yellow
2	STALK-SHAPE: enlarging tapering
4	STALK-ROOT: bulbous club equal rooted
4	STALK-SURFACE-ABOVE-RING: fibrous scaly silky smooth
4	STALK-SURFACE-BELOW-RING: fibrous scaly silky smooth
9	STALK-COLOR-ABOVE-RING: brown buff cinnamon gray orange pink red white yellow
9	STALK-COLOR-BELOW-RING: brown buff cinnamon gray orange pink red white yellow
2	VEIL-TYPE: partial universal
4	VEIL-COLOR: brown orange white yellow
3	RING-NUMBER: none one two
5	RING-TYPE: evanescent flaring large none pendant
9	SPORE-PRINT-COLOR: black brown buff chocolate green orange purple white yellow
6	POPULATION: abundant clustered numerous scattered several solitary
7	HABITAT: grasses leaves meadows paths urban waste woods

22 attributes with an average of 5 values!

Simplifying with "Naïve" Bayes

- What if we assume the features are independent?

$$\begin{aligned} p(\mathbf{x}|C_k) &= p(x_1, \ldots, x_N|C_k) \\ &= \prod_{n=1}^{N} p(x_n|C_k) \end{aligned}$$

- We know that's not precisely true, but it might make a good approximation.
- Now we only need to specify N different likelihoods:

$$p(x_i = v_i|C_k = k) = \frac{\text{Count}(x_i = v_i \wedge C_k = k)}{\text{Count}(C_k = k)}$$

- Huge savings in number of of parameters

Inference with Naïve Bayes

- Inference is just like before, but with the independence approximation:

$$
\begin{aligned}
p(C_k|\mathbf{x}) &= \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})} \\
&= \frac{p(C_k)\prod_n p(x_n|C_k)}{p(\mathbf{x})} \\
&= \frac{p(C_k)\prod_n p(x_n|C_k)}{\displaystyle\sum_k p(C_k)\prod_n p(x_n|C_k)}
\end{aligned}
$$

- Classification performance is often surprisingly good
- easy to implement

Implementation issues

- If you implement Naïve Bayes naïvely, you'll run into trouble. Why?

$$p(C_k|\mathbf{x}) \quad = \quad \frac{p(C_k)\prod_n p(x_n|C_k)}{\sum_k p(C_k)\prod_n p(x_n|C_k)}$$

- It's never good to compute products of a long list of numbers
- They'll quickly go to zero with machine precision, even using doubles (64 bit)
- Strategy: compute log probabilities

$$\begin{aligned} \log p(C_k|\mathbf{x}) \quad &= \quad \log p(C_k) + \sum_n \log p(x_n|C_k) - \log\left[\sum_k p(C_k)\prod_n p(x_n|C_k)\right] \\ &= \quad \log p(C_k) + \sum_n \log p(x_n|C_k) - \text{constant} \end{aligned}$$

- What about that constant? It still has a product.

Converting back to probabilities

- The only requirement of the denominator is that it normalize the numerator to yield a valid probability distribution.
- We used a log transformation:

$$g_i = \log p_i + \text{constant}$$

- The form of the probability the same for any constant c

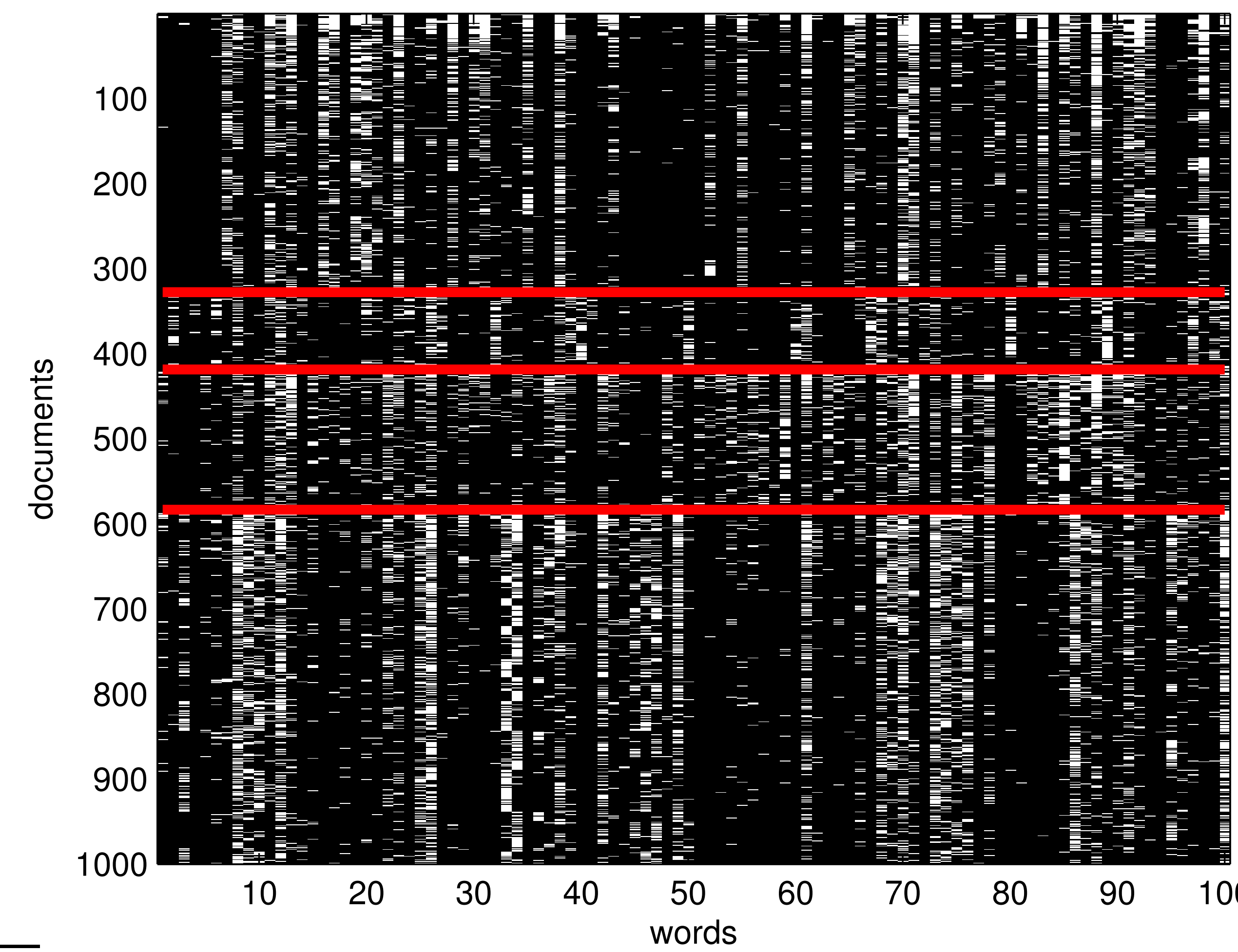
$$\begin{aligned}
\frac{p_i}{\sum_i p_i} &= \frac{e^{g_i}}{\sum_i e^{g_i}} \\
&= \frac{e^c e^{g_i}}{\sum_i e^c e^{g_i}} \\
&= \frac{e^{g_i+c}}{\sum_i e^{g_i+c}}
\end{aligned}$$

- A common choice: choose c so that the log probabilities are shifted to zero:

$$c = -\max_i g_i$$

Text classification with the *bag of words* model

- Each row is a document represented as a bag-of-words vector.
- The different classes are different newsgroups.
- The differences in word frequencies are readily apparent.
- We can use mixture models and naïve Bayes to classify the documents



$$p(C_k|\mathbf{x}) = \frac{p(C_k)\prod_n p(x_n|C_k)}{\sum_k p(C_k)\prod_n p(x_n|C_k)}$$

- We only replace the data likelihood with our bag-of-words model.
- This is a common way to build a spam filter or classify web pages.