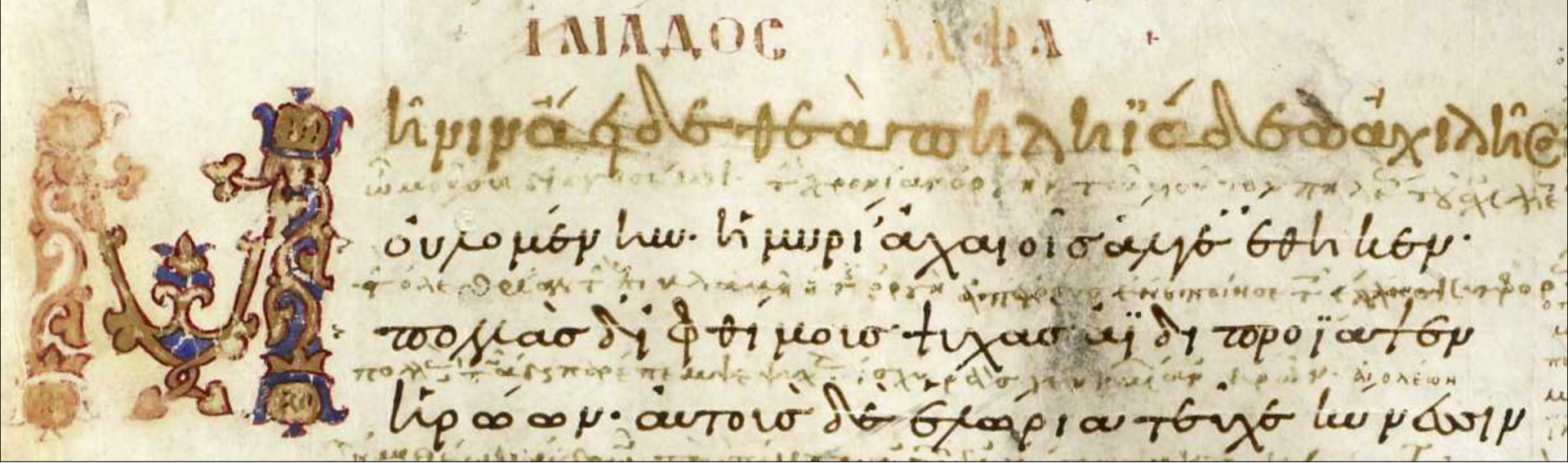




Natural Language Processing

Info 159/259

Lecture 3: Text classification 1 (Jan 26, 2021)

David Bamman, UC Berkeley

Ethics: Bias

- Embeddings for African-American first names are closer to “unpleasant” words than European names

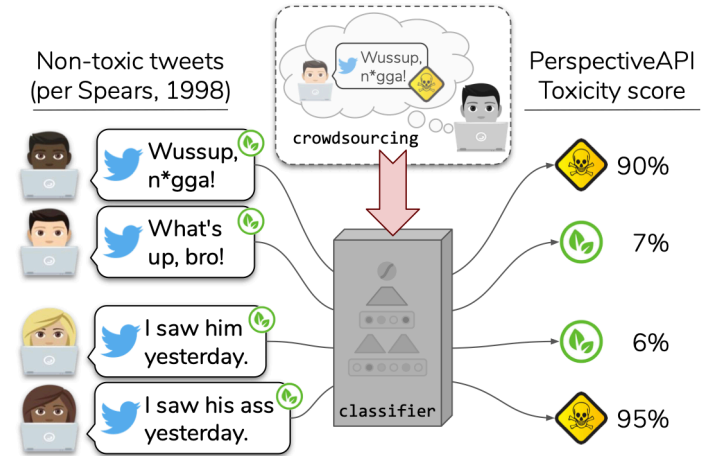
Caliskan et al. 2017, “Semantics derived automatically from language corpora contain human-like biases”

- Sentiment analysis over sentences containing African-American first names are more negative than identical sentences with European names

Kiritchenko and Mohammad (2018), “Examining Gender and Race Bias in Two Hundred Sentiment Analysis Systems”

- Toxicity detection systems score text with African-American English as more offensive

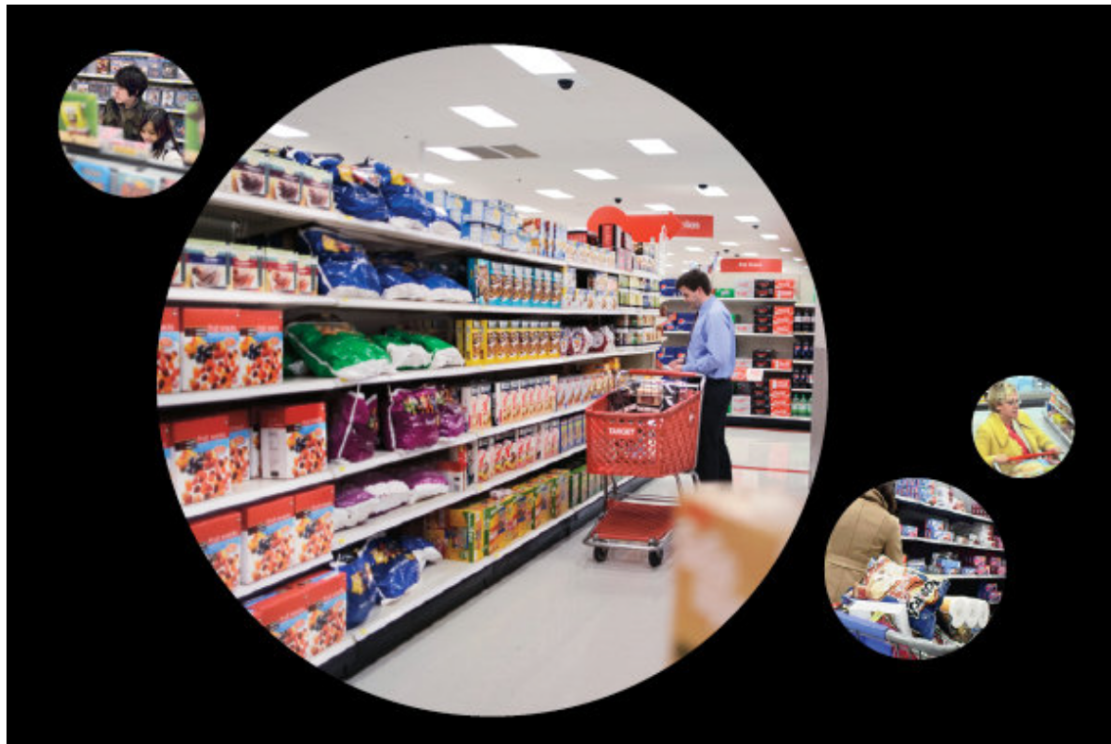
Sap et al. (2019), “The risk of racial bias in hate speech detection”



Privacy

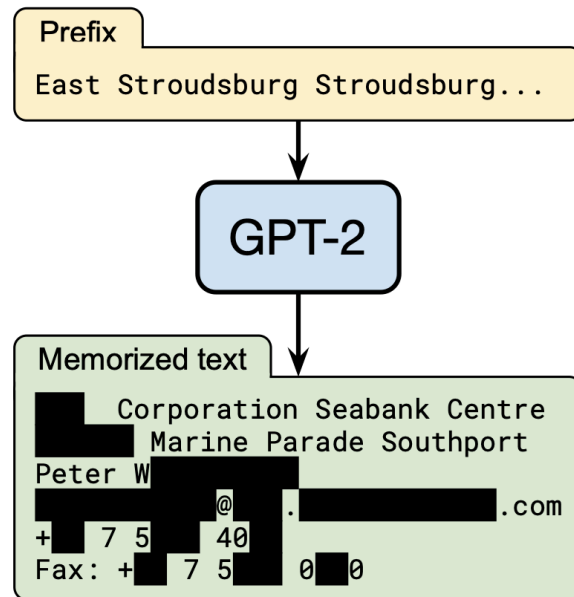
How Companies Learn Your Secrets

By CHARLES DUHIGG FEB. 16, 2012



Privacy

- Large language models (e.g., GPT-3, BERT) can memorize training data, which is recoverable from it.
- Potential violations of confidential data (e.g., Gmail messages) and **contextual integrity** (data being published in a way that violates a user's expectations of use).



Exclusion

- Focus on data from one domain/demographic
- State-of-the-art models perform worse for young (Hovy and Søgaard 2015) and minorities (Blodgett et al. 2016)

Exclusion

	AAE	White-Aligned
<i>langid.py</i>	13.2%	7.6%
Twitter-1	8.4%	5.9%
Twitter-2	24.4%	17.6%

Table 3: Proportion of tweets in AA- and white-aligned corpora classified as non-English by different classifiers. (§4.1)

Parser	AA	Wh.	Difference
SyntaxNet	64.0 (2.5)	80.4 (2.2)	16.3 (3.4)
CoreNLP	50.0 (2.7)	71.0 (2.5)	21.0 (3.7)

Language identification

Dependency parsing

Dual Use

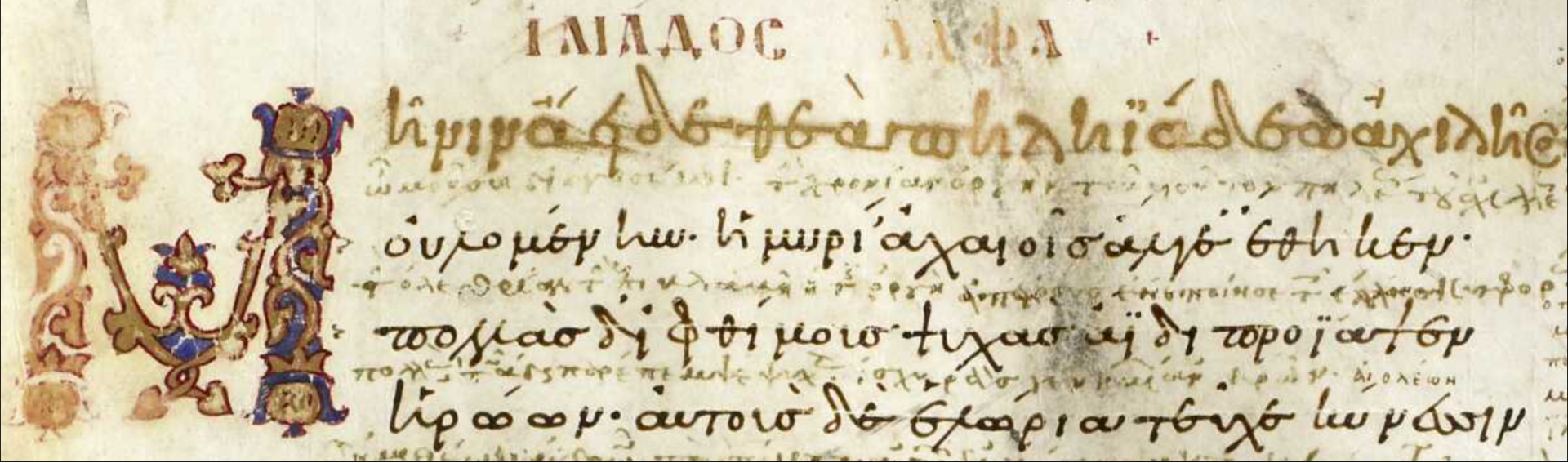
- **Authorship attribution** (author of *Federalist Papers* vs. author of ransom note vs. author of political dissent)
- **Fake review detection** vs. fake review generation
- **Censorship** evasion vs. enabling more robust censorship



ACM Conference on Fairness,
Accountability, and Transparency
(ACM FAccT)



Ethics, Bias, and Fairness. This area includes work that analyzes, detects and mitigates stereotypical bias or offensive wording in language data as well as work discussing ethical concerns about NLP applications.



Natural Language Processing

Info 159/259

Lecture 3: Text classification 1 (Jan 26, 2021)

David Bamman, UC Berkeley



Classification

A mapping h from input data x (drawn from instance space $\mathcal{X}$) to a label (or labels) y from some enumerable output space $\mathcal{Y}$

$\mathcal{X}$ = set of all documents

$\mathcal{Y} = \{\text{english, mandarin, greek, ...}\}$

x = a single document

y = ancient greek



Classification

$$h(x) = y$$

$h(\mu\eta\eta\nu\nu\ \acute{\alpha}\epsilon\iota\delta\epsilon\ \theta\epsilon\acute{\alpha}) = \text{ancient grc}$



Classification

Let $h(x)$ be the “true” mapping.
We never know it. How do we
find the best $\hat{h}(x)$ to
approximate it?

One option: rule based

if x has characters in
unicode point range 0370-03FF:
 $\hat{h}(x) = \text{greek}$



Classification

Supervised learning

Given training data in the form
of $\langle x, y \rangle$ pairs, learn $\hat{h}(x)$

Text categorization problems

task	$\mathcal{X}$	$\mathcal{Y}$
language ID	text	{english, mandarin, greek, ...}
spam classification	email	{spam, not spam}
authorship attribution	text	{jk rowling, james joyce, ...}
genre classification	novel	{detective, romance, gothic, ...}
sentiment analysis	text	{positive, negative, neutral, mixed}

Sentiment analysis

- Document-level SA: is the entire text **positive** or **negative** (or both/neither) with respect to an implicit target?
- Movie reviews [Pang et al. 2002, Turney 2002]

Training data

positive

“... is a film which still causes real, not figurative, chills to run along my spine, and it is certainly the bravest and most ambitious fruit of Coppola's genius”

Roger Ebert, Apocalypse Now

- “I hated this movie. Hated hated hated hated hated this movie. Hated it. Hated every simpering stupid vacant audience-insulting moment of it. Hated the sensibility that thought anyone would like it.”

negative

Roger Ebert, North

Sentiment analysis

- Is the text positive or negative (or both/neither) with respect to an explicit target **within the text**?

Feature: **picture**

Positive: 12

- Overall this is a good camera with a really good picture clarity.
- The pictures are absolutely amazing - the camera captures the minutest of details.
- After nearly 800 pictures I have found that this camera takes incredible pictures.

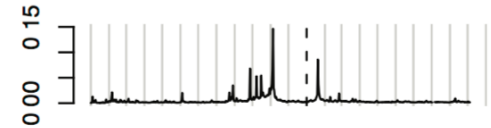
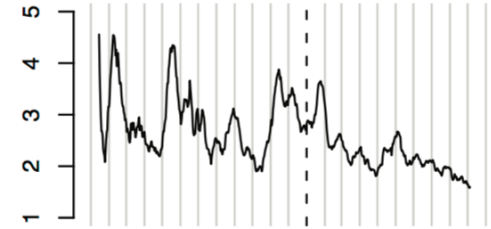
...

Negative: 2

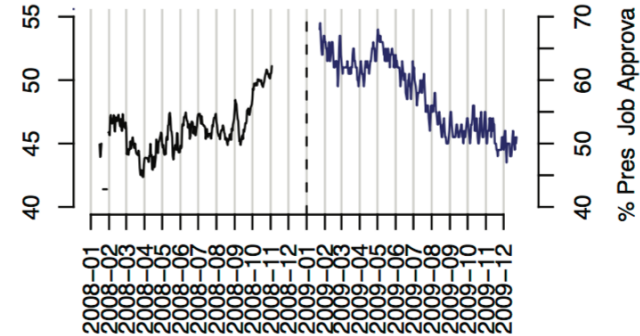
- The pictures come out hazy if your hands shake even for a moment during the entire process of taking a picture.
- Focusing on a display rack about 20 feet away in a brightly lit room during day time, pictures produced by this camera were blurry and in a shade of orange.

Hu and Liu (2004), "Mining and Summarizing Customer Reviews"

Twitter sentiment →



Job approval polls →



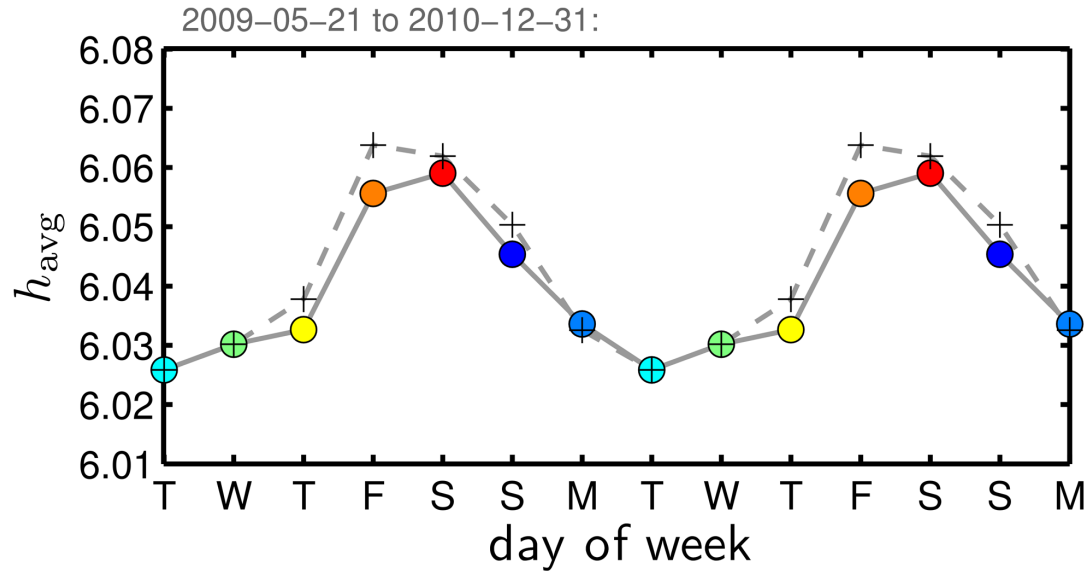
O'Connor et al (2010), "From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series"

Figure 9: The sentiment ratio for *obama* (15-day window), and fraction of all Twitter messages containing *obama* (day-by-day, no smoothing), compared to election polls (2008) and job approval polls (2009).

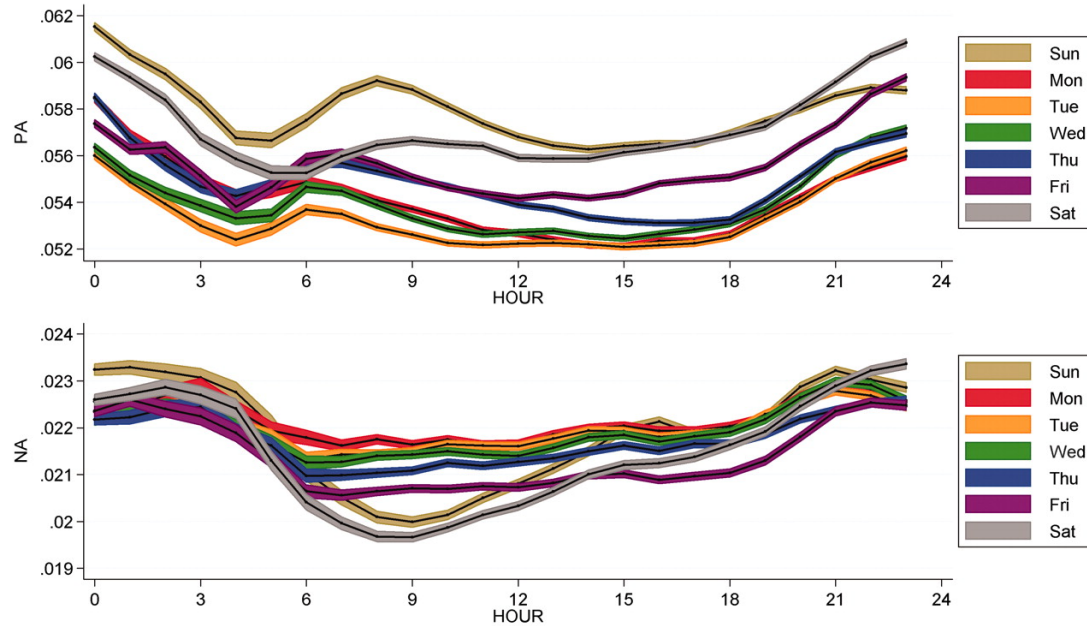
Sentiment as tone

- No longer the speaker's attitude with respect to some particular target, but rather the positive/negative **tone** that is evinced.

Sentiment as tone



Sentiment as tone



Golder and Macy (2011), "Diurnal and Seasonal Mood Vary with Work, Sleep, and Daylength Across Diverse Cultures," *Science*. Positive affect (PA) and negative affect (NA) measured with LIWC.

Sentiment Dictionaries

- General Inquirer (1966)
- MPQA subjectivity lexicon (Wilson et al. 2005)
http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/
- LIWC (Linguistic Inquiry and Word Count, Pennebaker 2015)
- AFINN (Nielsen 2011)
- NRC Word-Emotion Association Lexicon (EmoLex), Mohammad and Turney 2013

pos	neg
unlimited	lag
prudent	contortions
superb	fright
closeness	lonely
impeccably	tenuously
fast-paced	plebeian
treat	mortification
destined	outrage
blessing	allegations
steadfastly	disoriented

LIWC

- 73 separate lexicons designed for applications social psychology

Positive Emotion	Negative Emotion	Insight	Inhibition	Family	Negate
appreciat*	anger*	aware*	avoid*	brother*	aren't
comfort*	bore*	believe	careful*	cousin*	cannot
great	cry	decid*	hesitat*	daughter*	didn't
happy	despair*	feel	limit*	family	neither
interest	fail*	figur*	oppos*	father*	never
joy*	fear	know	prevent*	grandf*	no
perfect*	griev*	knew	reluctan*	grandm*	nobod*
please*	hate*	means	safe*	husband	none
safe*	panic*	notice*	stop	mom	nor
terrific	suffers	recogni*	stubborn*	mother	nothing
value	terrify	sense	wait	niece*	nowhere
wow*	violent*	think	wary	wife	without

Why is SA hard?

- Sentiment is a measure of a speaker's private state, which is unobservable.
- Sometimes words are a good indicator of sentiment (love, amazing, hate, terrible); many times it requires deep world + contextual knowledge

“*Valentine's Day* is being marketed as a Date Movie. I think it's more of a First-Date Movie. If your date likes it, do not date that person again. And if you like it, there may not be a second date.”

Roger Ebert, *Valentine's Day*



Classification

Supervised learning

Given training data in the form of $\langle x, y \rangle$ pairs, learn $\hat{h}(x)$

x	y
loved it!	positive
terrible movie	negative
not too shabby	positive

$$\hat{h}(x)$$

- The classification function that we want to learn has two different components:
 - the formal structure of the learning method (what's the relationship between the input and output?) → Naive Bayes, logistic regression, convolutional neural network, etc.
 - the **representation** of the data

Representation for SA

- Only positive/negative words in MPQA
- Only words in isolation (bag of words)
- Conjunctions of words (sequential, skip ngrams, other non-linear combinations)
- Higher-order linguistic structure (e.g., syntax)

“... is a film which still causes real, not figurative, chills to run along my spine, and it is certainly the **bravest** and most **ambitious** fruit of Coppola's **genius**”

Roger Ebert, Apocalypse Now

“I **hated** this movie. **Hated hated hated hated hated** this movie. **Hated** it. **Hated** every simpering **stupid** vacant audience-**insulting** moment of it. **Hated** the sensibility that thought anyone would **like** it.”

Roger Ebert, North

Bag of words

Representation of text only
as the counts of words that
it contains

	Apocalypse now	North
the	1	1
of	0	0
hate	0	9
genius	1	0
bravest	1	0
stupid	0	1
like	0	1
...		

Naive Bayes

- Given access to $\langle x, y \rangle$ pairs in training data, we can train a model to estimate the class probabilities for a new review.
- With a bag of words representation (in which each word is independent of the other), we can use Naive Bayes
- Probabilistic model; not as accurate as other models (see next two classes) but fast to train and the foundation for many other probabilistic techniques.

Random variable

- A variable that can take values within a fixed set (discrete) or within some range (continuous).

$$X \in \{1, 2, 3, 4, 5, 6\}$$

$$X \in \{the, a, dog, cat, runs, to, store\}$$

$$P(X = x)$$

Probability that the random variable X takes the value x (e.g., 1)

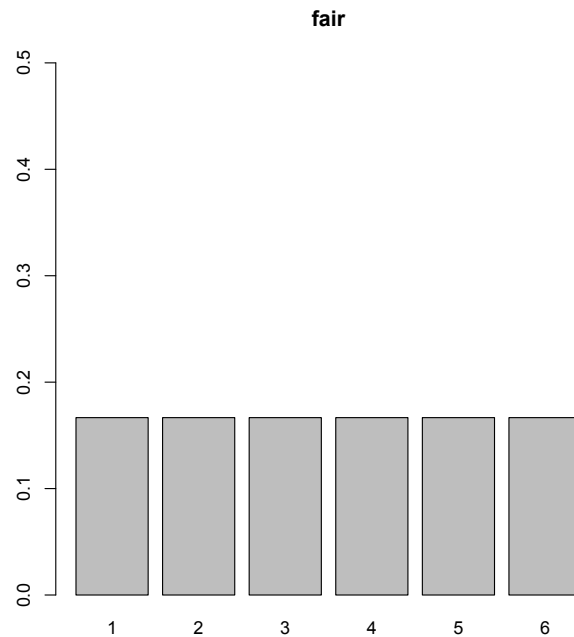
$$X \in \{1, 2, 3, 4, 5, 6\}$$

Two conditions:

1. Between 0 and 1: $0 \leq P(X = x) \leq 1$
2. Sum of all probabilities = 1 $\sum_x P(X = x) = 1$

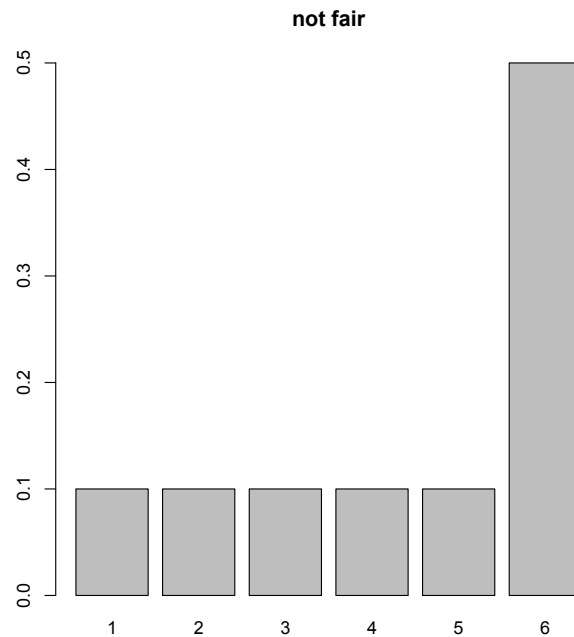
Fair dice

$$X \in \{1, 2, 3, 4, 5, 6\}$$



Weighted dice

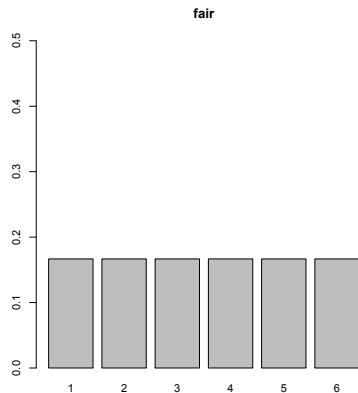
$$X \in \{1, 2, 3, 4, 5, 6\}$$



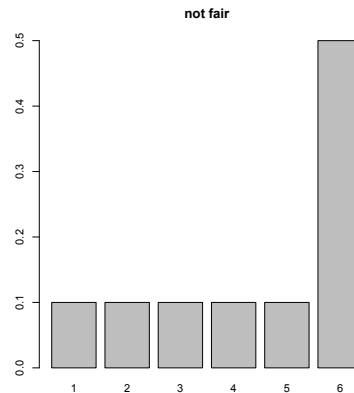
Inference

$$X \in \{1, 2, 3, 4, 5, 6\}$$

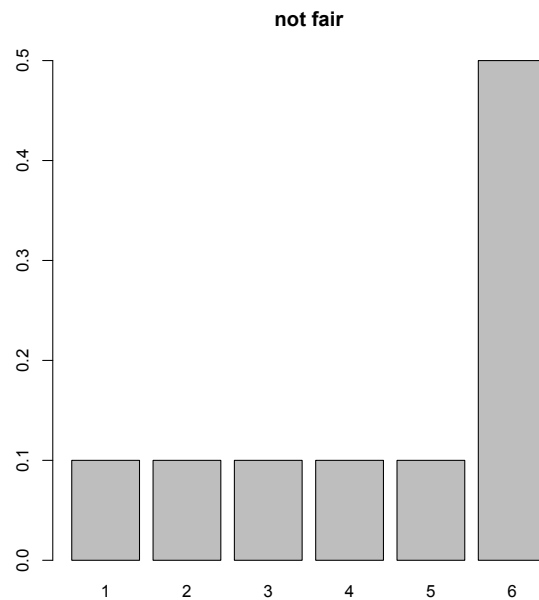
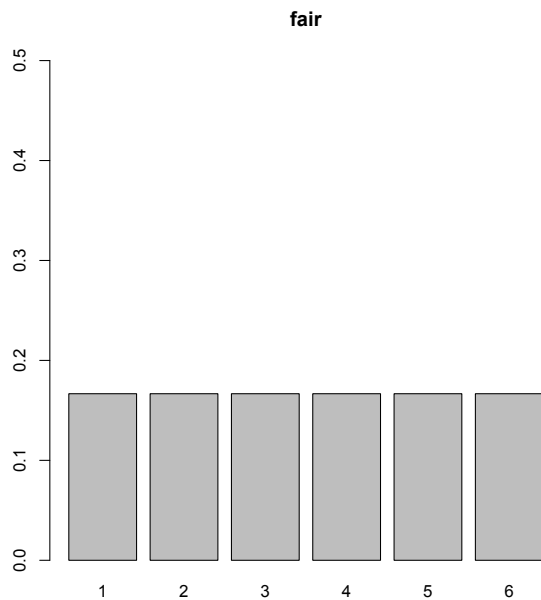
We want to *infer* the probability distribution that generated the data we see.



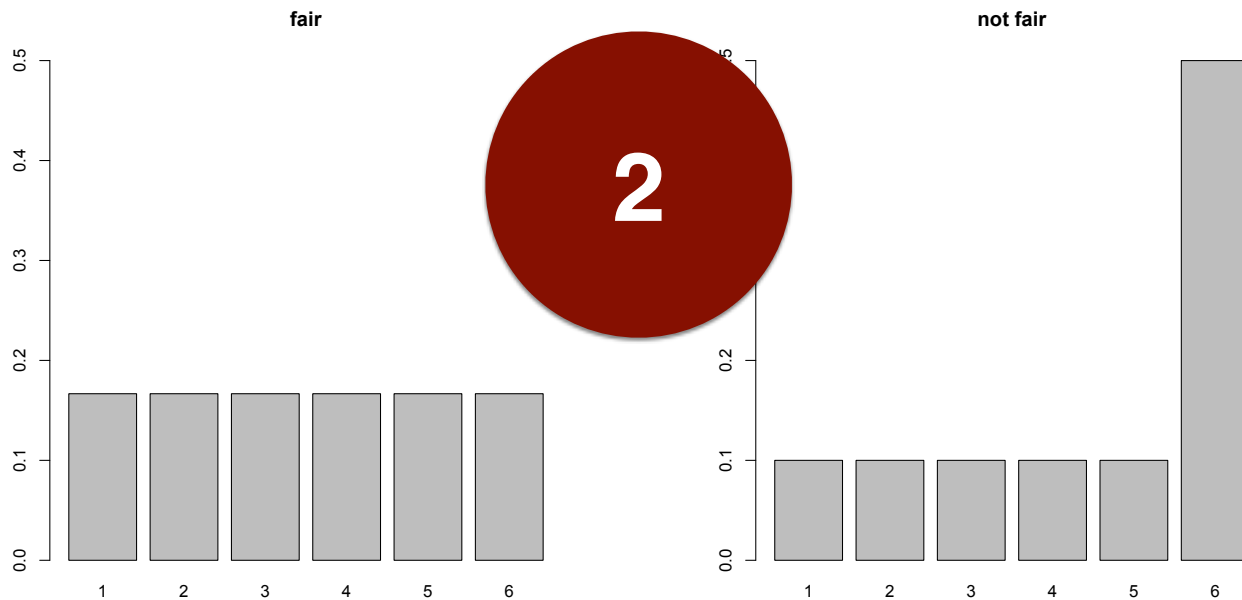
?



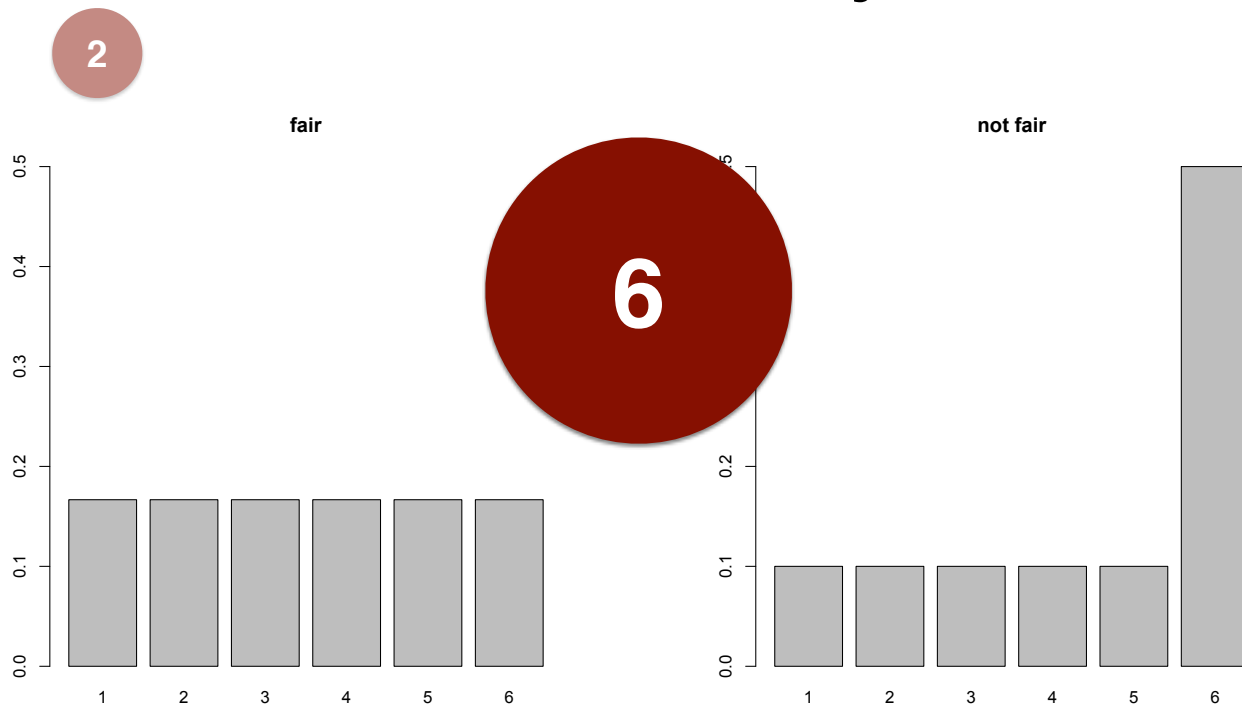
Probability



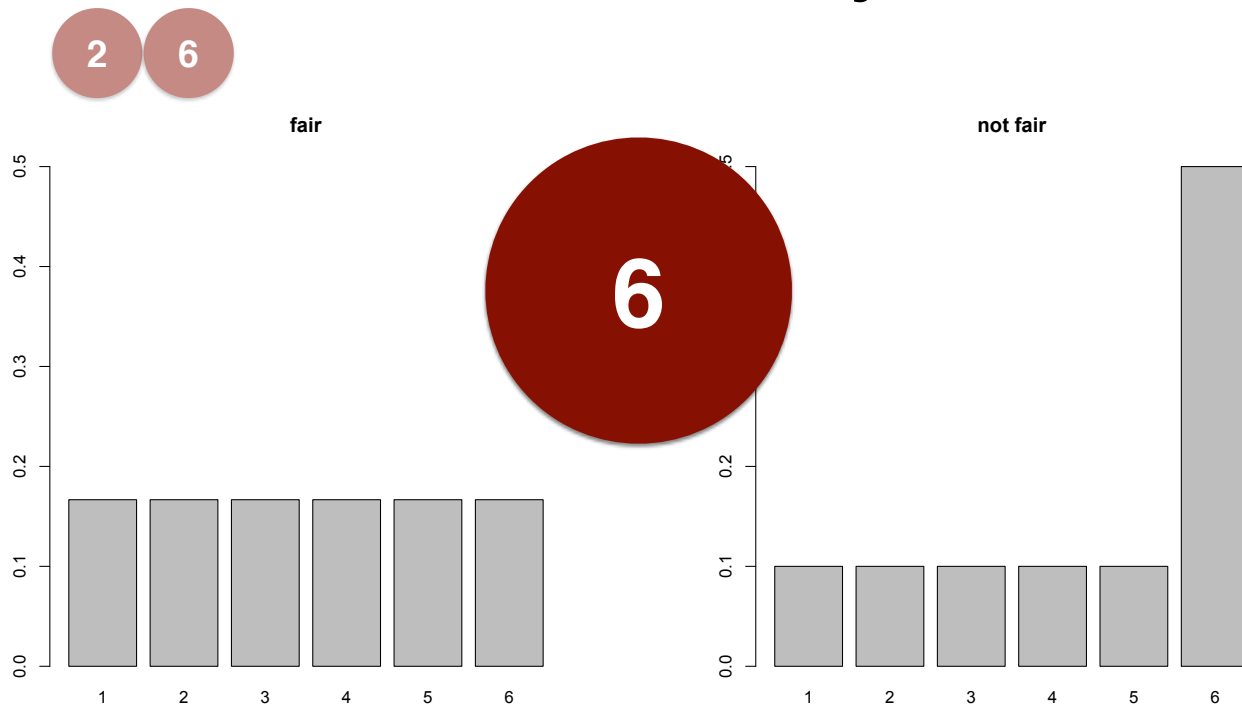
Probability



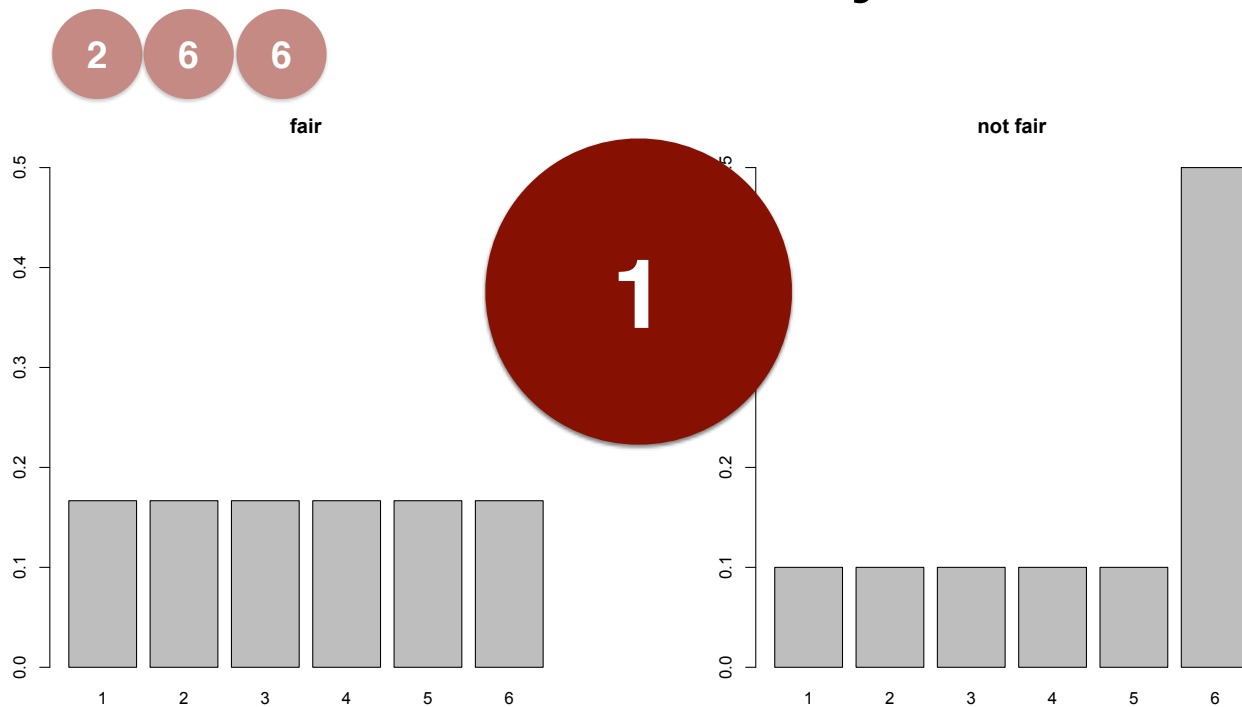
Probability



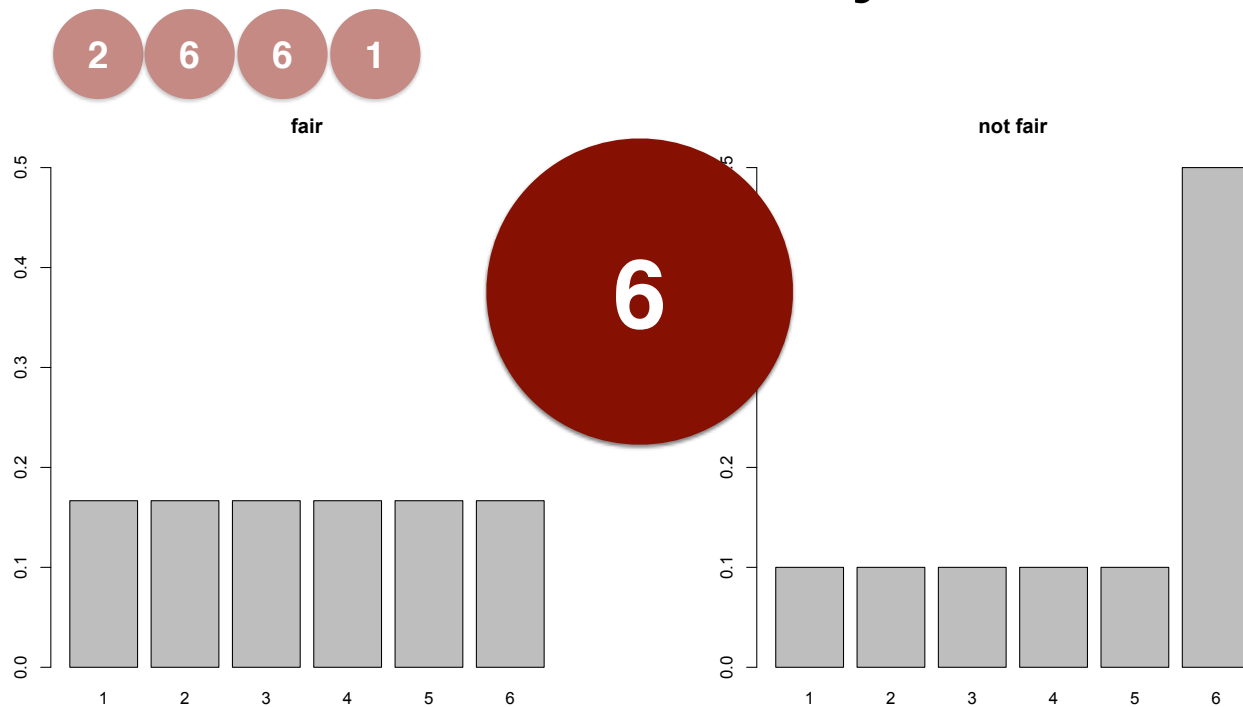
Probability



Probability



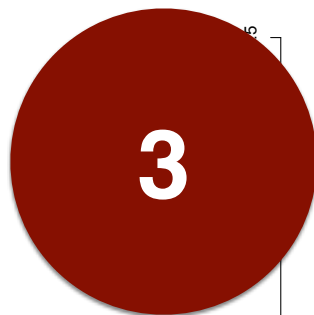
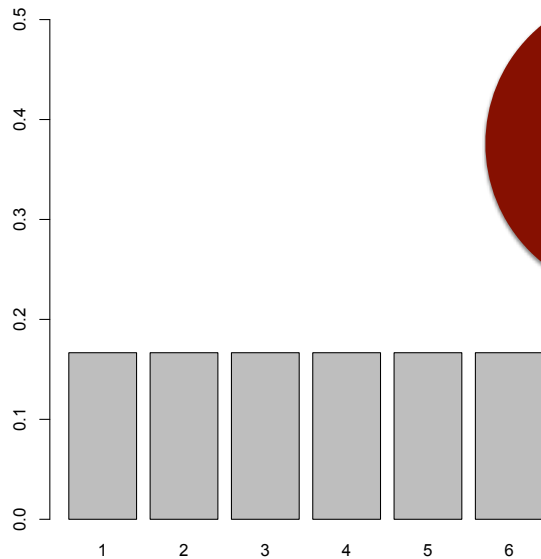
Probability



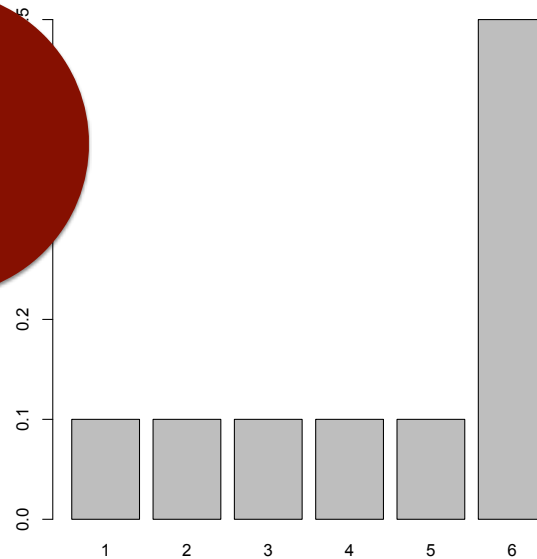
Probability



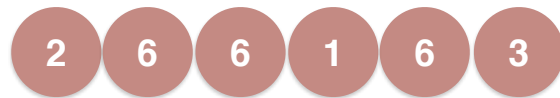
fair



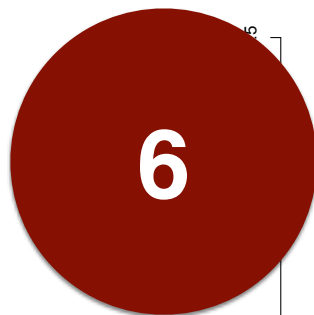
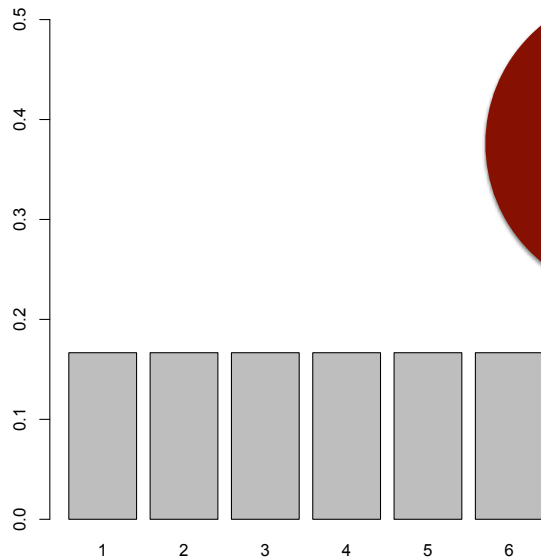
not fair



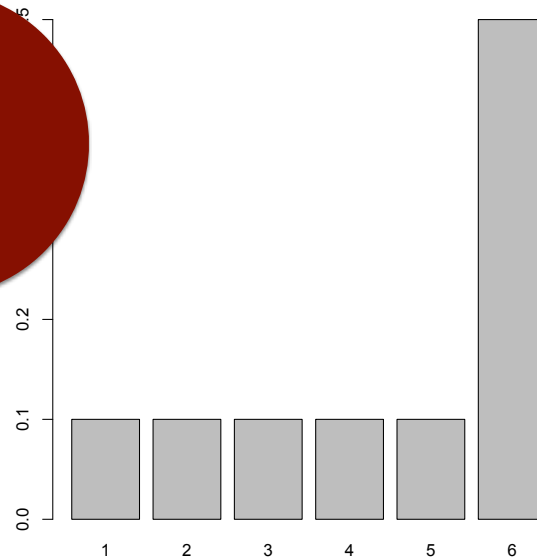
Probability



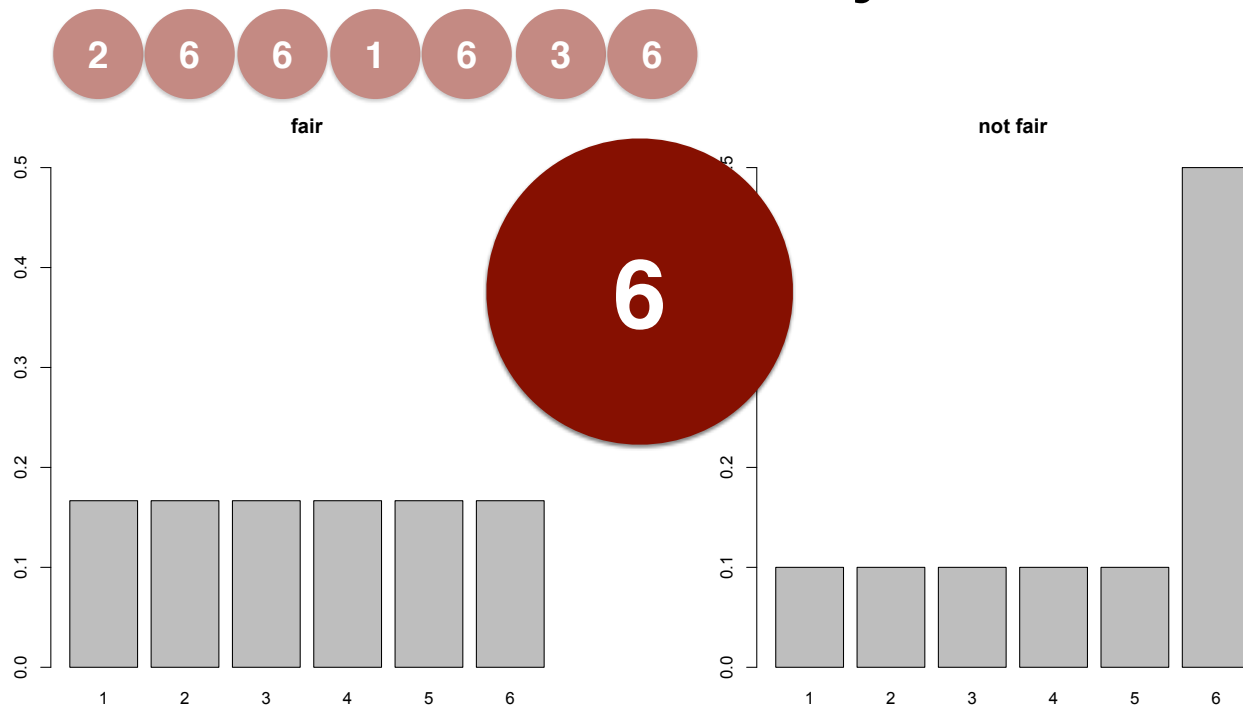
fair



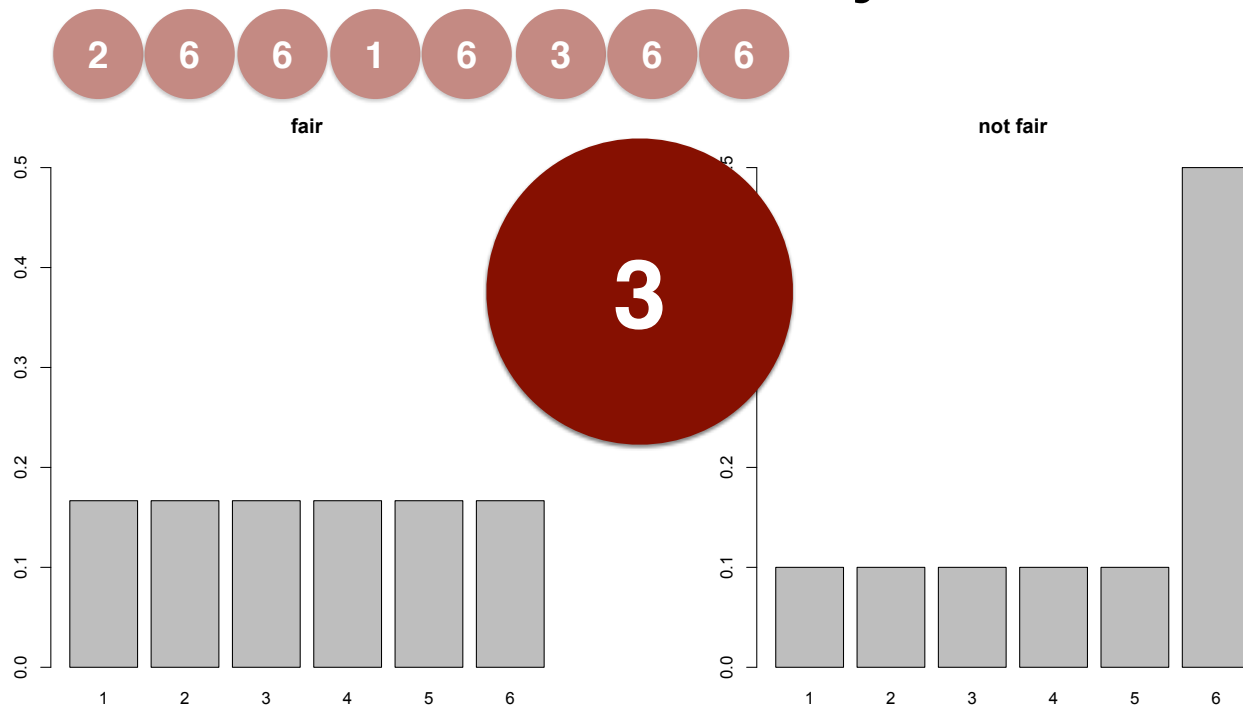
not fair



Probability



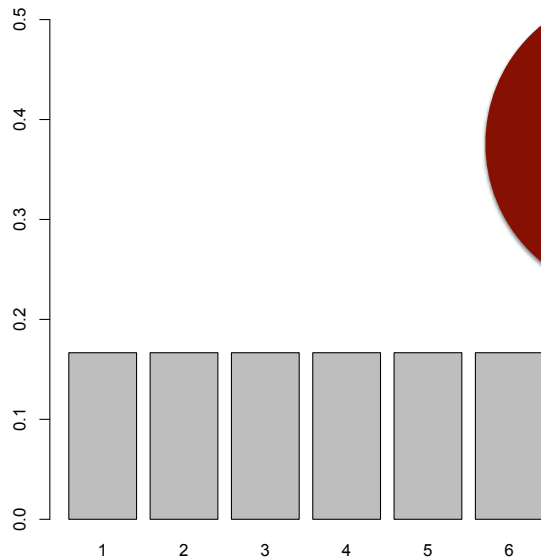
Probability



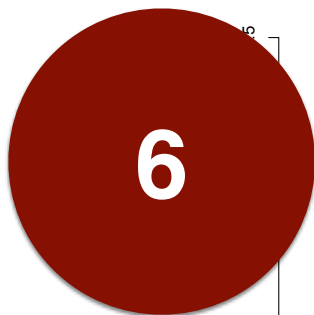
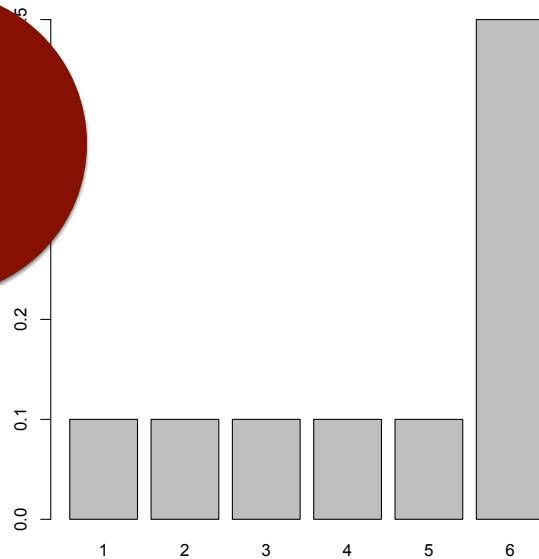
Probability



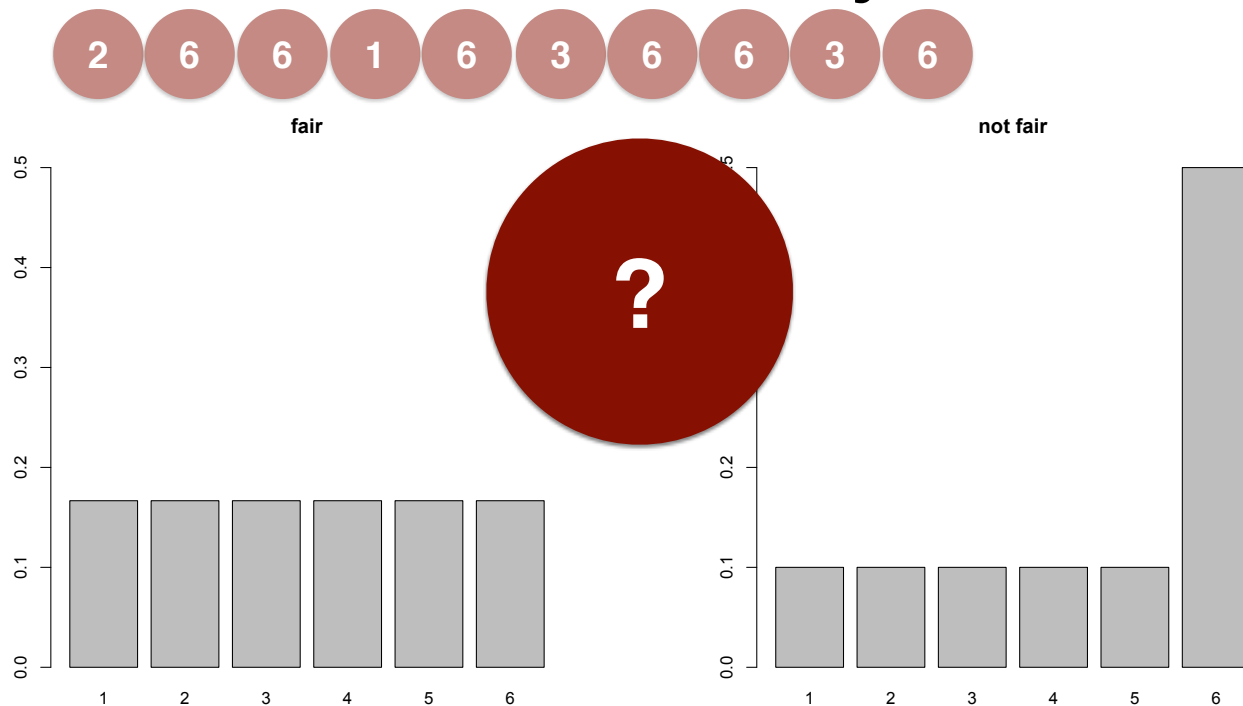
fair



not fair



Probability



Independence

- Two random variables are independent if:

$$P(A, B) = P(A) \times P(B)$$

- In general:

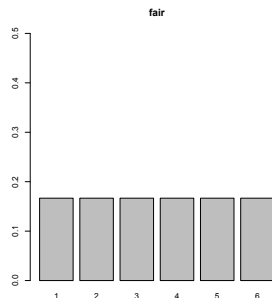
$$P(x_1, \dots, x_n) = \prod_{i=1}^N P(x_i)$$

- Information about one random variable (B) gives no information about the value of another (A)

$$P(A) = P(A \mid B) \qquad P(B) = P(B \mid A)$$

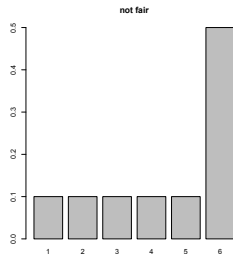
Data Likelihood

$$P(\text{2, 6, 6} \mid \text{fair})$$



$$= .17 \times .17 \times .17$$
$$= 0.004913$$

$$P(\text{2, 6, 6} \mid \text{not fair})$$

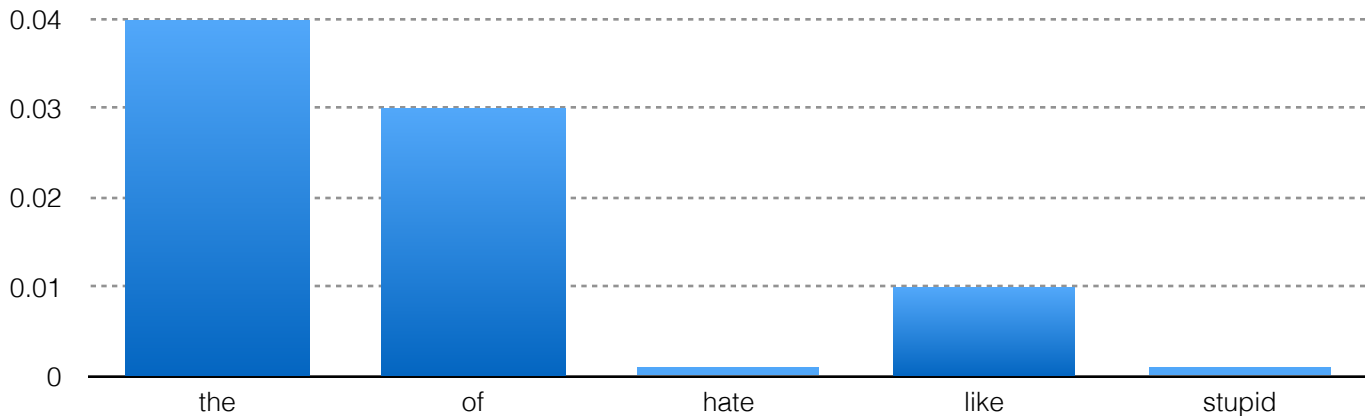


$$= .1 \times .5 \times .5$$
$$= 0.025$$

Data Likelihood

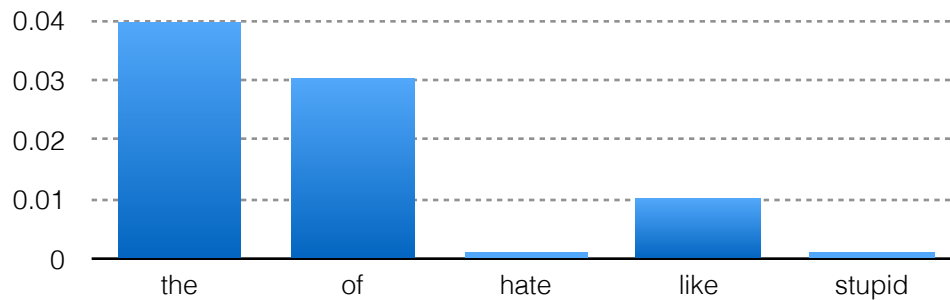
- The likelihood gives us a way of discriminating between possible alternative parameters, but also a strategy for picking a single best parameter among all possibilities

Word choice as weighted dice

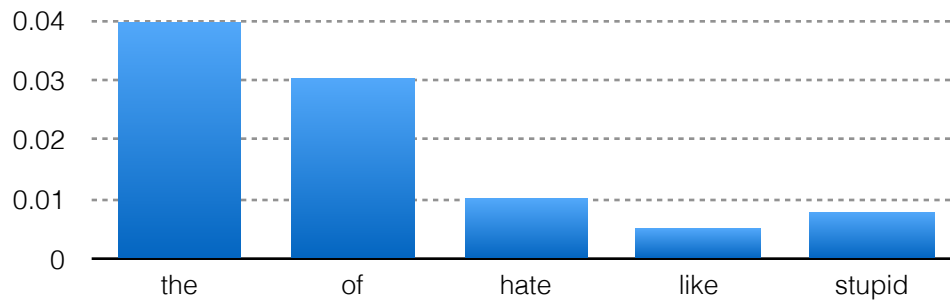


Unigram probability

positive reviews



negative reviews

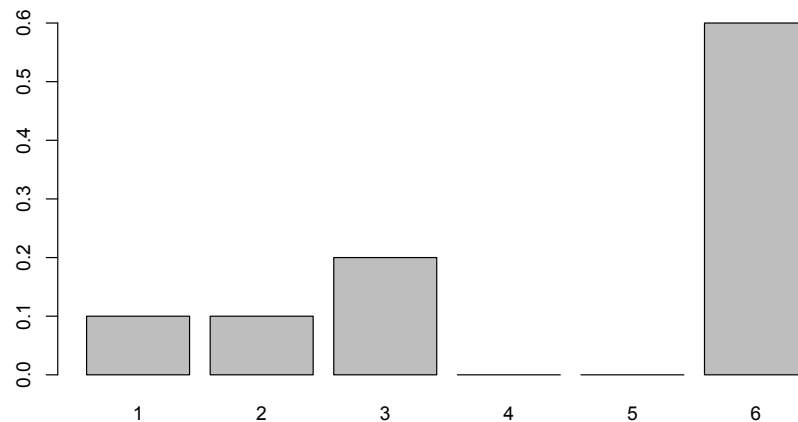


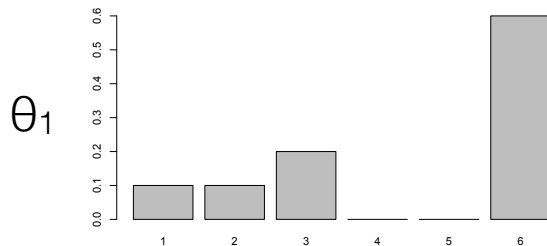
$$P(X = the) = \frac{\#the}{\#total\ words}$$

Maximum Likelihood Estimate

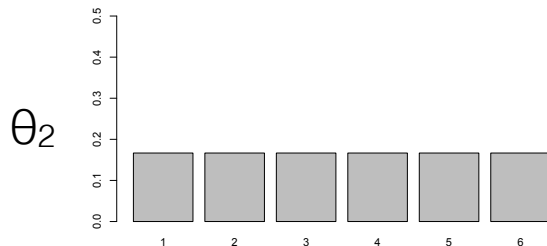
- This is a maximum likelihood estimate for $P(X)$; the parameter values for which the data we observe (X) is **most likely**.

Maximum Likelihood Estimate



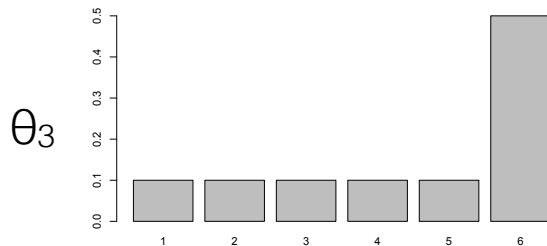


$$P(X \mid \theta_1) = 0.0000311040$$



$$P(X \mid \theta_2) = 0.0000000992$$

(313x less likely)



$$P(X \mid \theta_3) = 0.0000031250$$

(10x less likely)

Conditional Probability

$$P(X = x | Y = y)$$

- Probability that one random variable takes a particular value *given* the fact that a different variable takes another

$$P(X_i = \textit{hate} \mid Y = \oplus)$$

Sentiment analysis

“really really the worst movie ever”

Independence Assumption

really really the worst movie ever



$$P(\text{really, really, the, worst, movie, ever}) = P(\text{really}) \times P(\text{really}) \times P(\text{the}) \dots P(\text{ever})$$

Independence Assumption

really really the worst movie ever



We will assume the features are independent:

$$P(x_1, x_2, x_3, x_4, x_6, x_7 \mid c) = P(x_1 \mid c)P(x_2 \mid c) \dots P(x_7 \mid c)$$

$$P(x_1 \dots x_n \mid c) = \prod_{i=1}^N P(x_i \mid c)$$

A simple classifier

really really the worst movie ever

Y=Positive		Y=Negative	
$P(X=\text{really} \mid Y=\oplus)$	0.0010	$P(X=\text{really} \mid Y=\ominus)$	0.0012
$P(X=\text{really} \mid Y=\oplus)$	0.0010	$P(X=\text{really} \mid Y=\ominus)$	0.0012
$P(X=\text{the} \mid Y=\oplus)$	0.0551	$P(X=\text{the} \mid Y=\ominus)$	0.0518
$P(X=\text{worst} \mid Y=\oplus)$	0.0001	$P(X=\text{worst} \mid Y=\ominus)$	0.0004
$P(X=\text{movie} \mid Y=\oplus)$	0.0032	$P(X=\text{movie} \mid Y=\ominus)$	0.0045
$P(X=\text{ever} \mid Y=\oplus)$	0.0005	$P(X=\text{ever} \mid Y=\ominus)$	0.0005

A simple classifier

really really the worst movie ever

$$P(X = \text{"really really the worst movie ever"} \mid Y = \oplus)$$

$$\begin{aligned} &P(X=\text{really} \mid Y=\oplus) \times P(X=\text{really} \mid Y=\oplus) \times P(X=\text{the} \mid Y=\oplus) \times P(X=\text{worst} \mid \\ &Y=\oplus) \times P(X=\text{movie} \mid Y=\oplus) \times P(X=\text{ever} \mid Y=\oplus) \\ &= 6.00\text{e-}18 \end{aligned}$$

$$P(X = \text{"really really the worst movie ever"} \mid Y = \ominus)$$

$$\begin{aligned} &P(X=\text{really} \mid Y=\ominus) \times P(X=\text{really} \mid Y=\ominus) \times P(X=\text{the} \mid Y=\ominus) \times P(X=\text{worst} \mid \\ &Y=\ominus) \times P(X=\text{movie} \mid Y=\ominus) \times P(X=\text{ever} \mid Y=\ominus) \\ &= 6.20\text{e-}17 \end{aligned}$$

Aside: use logs

- Multiplying lots of small probabilities (all are under 1) can lead to numerical underflow (converging to 0)

$$\log \prod_i x_i = \sum_i \log x_i$$

A simple classifier

- The classifier we just specified is a maximum likelihood classifier, where we compare the **likelihood** of the data under each class and choose the class with the highest likelihood

Likelihood: probability of data
(here, under class y)

$$P(X = x_1 \dots x_n \mid Y = y)$$

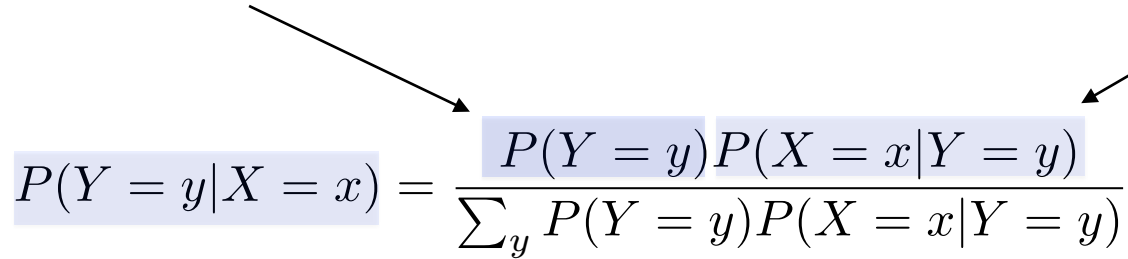
Prior probability of class y

$$P(Y = y)$$

Bayes' Rule

Prior belief that $Y = y$
(before you see any data)

Likelihood of the data
given that $Y=y$

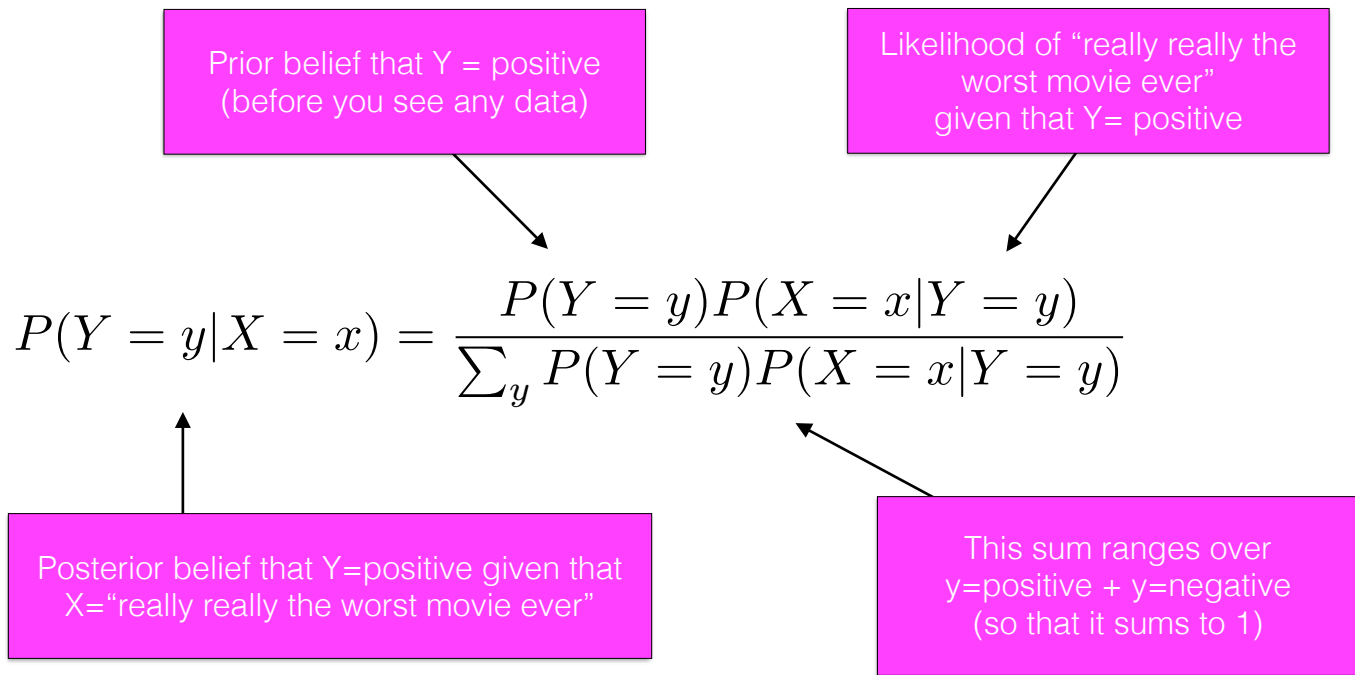


The diagram shows the Bayes' Rule equation with three arrows indicating the components: one from the 'Prior belief' text to the numerator's first term, one from the 'Likelihood' text to the numerator's second term, and one from the 'Posterior belief' text to the entire fraction.

$$P(Y = y|X = x) = \frac{P(Y = y)P(X = x|Y = y)}{\sum_y P(Y = y)P(X = x|Y = y)}$$

Posterior belief that $Y=y$ given that $X=x$

Bayes' Rule



Likelihood: probability of data
(here, under class y)

$$P(X = x_1 \dots x_n \mid Y = y)$$

Prior probability of class y

$$P(Y = y)$$

Posterior belief in the probability of
class y after seeing data

$$P(Y = y \mid X = x_1 \dots x_n)$$

Naive Bayes Classifier

$$\frac{P(Y = \oplus)P(X = \text{“really ...”} \mid Y = \oplus)}{P(Y = \oplus)P(X = \text{“really ...”} \mid Y = \oplus) + P(Y = \ominus)P(X = \text{“really ...”} \mid Y = \ominus)}$$

Let's say $P(Y=\oplus) = P(Y=\ominus) = 0.5$
(i.e., both are equally likely a priori)

$$\frac{0.5 \times (6.00 \times 10^{-18})}{0.5 \times (6.00 \times 10^{-18}) + 0.5 \times (6.2 \times 10^{-17})}$$

$$P(Y = \oplus \mid X = \text{“really ...”}) = 0.088$$

$$P(Y = \ominus \mid X = \text{“really ...”}) = 0.912$$

Naive Bayes Classifier

- To turn probabilities into a classification decisions, we just select the label with the highest posterior probability

$$\hat{y} = \arg \max_{y \in \mathcal{Y}} P(Y | X)$$

$$P(Y = \oplus | X = \text{“really ...”}) = 0.088$$

$$P(Y = \ominus | X = \text{“really ...”}) = 0.912$$

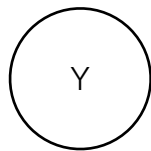
Taxicab Problem

“A cab was involved in a hit and run accident at night. Two cab companies, the Green and the Blue, operate in the city. You are given the following data:

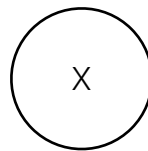
- 85% of the cabs in the city are Green and 15% are Blue.
- A witness identified the cab as Blue. The court tested the reliability of the witness under the same circumstances that existed on the night of the accident and concluded that the witness correctly identified each one of the two colors 80% of the time and failed 20% of the time.

What is the probability that the cab involved in the accident was Blue rather than Green knowing that this witness identified it as Blue?”

(Tversky & Kahneman 1981)



= true color of cab



= reported color of cab

Prior

$$P(Y = \text{Green}) = 0.85$$

$$P(Y = \text{Blue}) = 0.15$$

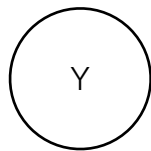
Likelihood

$$P(X = \text{Green} \mid Y = \text{Blue}) = 0.20$$

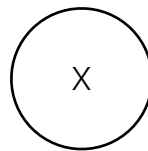
$$P(X = \text{Blue} \mid Y = \text{Blue}) = 0.80$$

$$P(X = \text{Green} \mid Y = \text{Green}) = 0.80$$

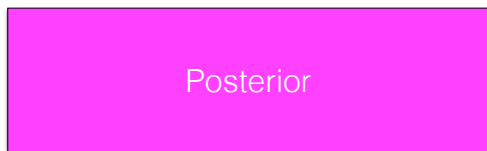
$$P(X = \text{Blue} \mid Y = \text{Green}) = 0.20$$



= true color of cab



= reported color of cab



$$P(Y = \text{Blue} \mid X = \text{Blue})$$

What we care about is this posterior value (the probability that the cab is blue given that the witness said it was blue). We can't measure it directly, but we can plug the prior and likelihood into Bayes' rule to get our answer.

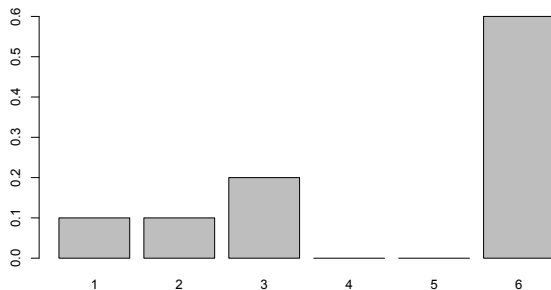
Priors

- Priors can be informed (reflecting expert knowledge) but in practice, but priors in Naive Bayes are often simply estimated from training data

$$P(Y = \oplus) = \frac{\#\oplus}{\#\text{total texts}}$$

Smoothing

- Maximum likelihood estimates can fail miserably when features are never observed with a particular class.



What's the probability of:



Smoothing

- One solution: add a little probability mass to every element.

maximum likelihood
estimate

$$P(x_i | y) = \frac{n_{i,y}}{n_y}$$

$n_{i,y}$ = count of word i in class y
 n_y = number of words in y
 V = size of vocabulary

smoothed estimates

$$P(x_i | y) = \frac{n_{i,y} + a}{n_y + Va}$$

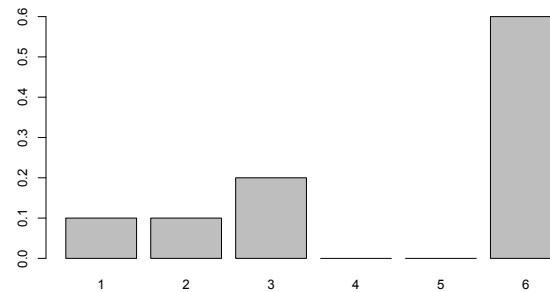
same a for all x_i

$$P(x_i | y) = \frac{n_{i,y} + a_i}{n_y + \sum_{j=1}^V a_j}$$

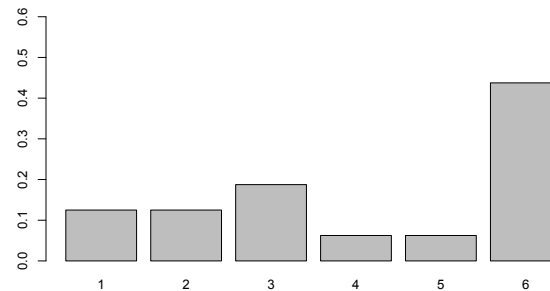
possibly different a for each x_i

Smoothing

MLE

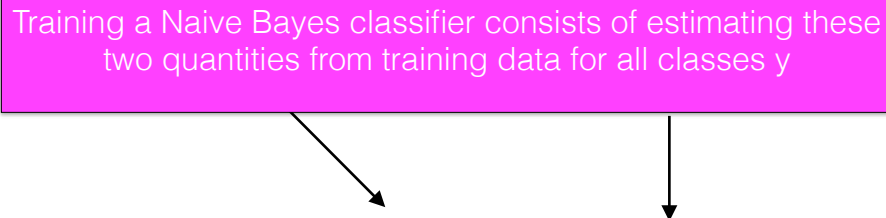


smoothing with $\alpha = 1$



Naive Bayes training

Training a Naive Bayes classifier consists of estimating these two quantities from training data for all classes y



$$P(Y = y|X = x) = \frac{P(Y = y)P(X = x|Y = y)}{\sum_y P(Y = y)P(X = x|Y = y)}$$

At test time, use those estimated probabilities to calculate the posterior probability of each class y and select the class with the highest probability

- Naive Bayes' independence assumption can be killer
- One instance of *hate* makes seeing others much more likely (each mention does contribute the same amount of information)
- We can mitigate this by not reasoning over counts of tokens but by their presence or absence

	Apocalypse now	North
the	1	1
of	0	0
hate	0	9 1
genius	1	0
bravest	1	0
stupid	0	1
like	0	1
...		

Tricks for SA

- Negation in bag of words: add negation marker to all words between negation and end of clause (e.g., comma, period) to create new vocab term [Das and Chen 2001]
 - I do not [like this movie]
 - I do not like_NEG this_NEG movie_NEG

Sentiment Dictionaries

- General Inquirer (1966)
- MPQA subjectivity lexicon (Wilson et al. 2005)
http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/
- LIWC (Linguistic Inquiry and Word Count, Pennebaker 2015)
- AFINN (Nielsen 2011)
- NRC Word-Emotion Association Lexicon (EmoLex), Mohammad and Turney 2013

pos	neg
unlimited	lag
prudent	contortions
superb	fright
closeness	lonely
impeccably	tenuously
fast-paced	plebeian
treat	mortification
destined	outrage
blessing	allegations
steadfastly	disoriented