INFO 2950: Intro to Data Science

Lecture 20
2023-11-06

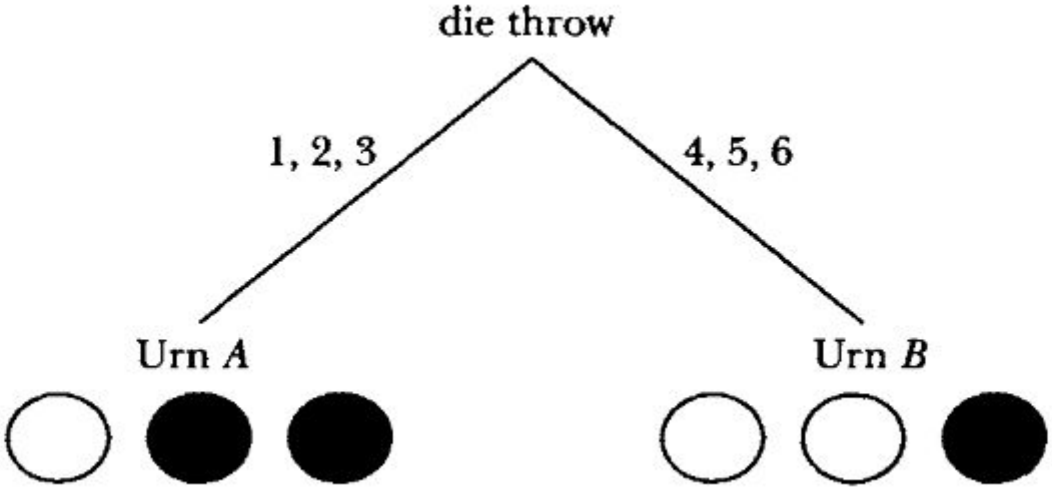
Agenda

1. Bayes Review
2. Text Analysis
3. Log probability
4. Naive Bayes Classifier

What is Bayes' rule for?

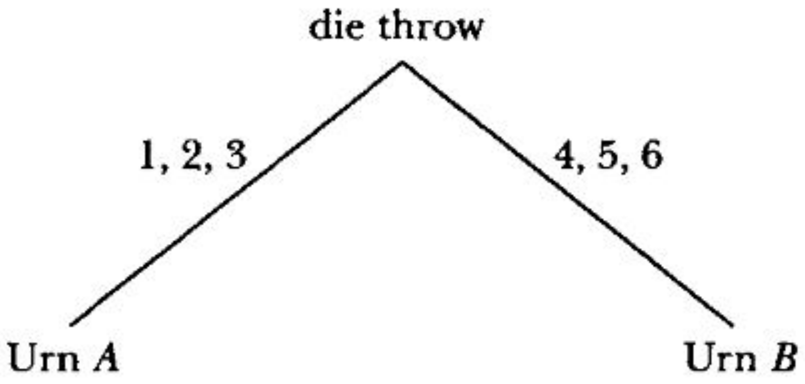
- Lots of teaching examples involve drawing “balls from urns” because it’s an easy way to explain the concept
- But, there are lots of places you might use Bayes’ rule in data science jobs, e.g.: estimating probabilities about spam filters detecting spam emails, given they are spam (or not spam)
- This is why lots of data science interviews test for things like understanding Bayes’ rule!

Balls in Urns!



https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

We secretly throw a die!

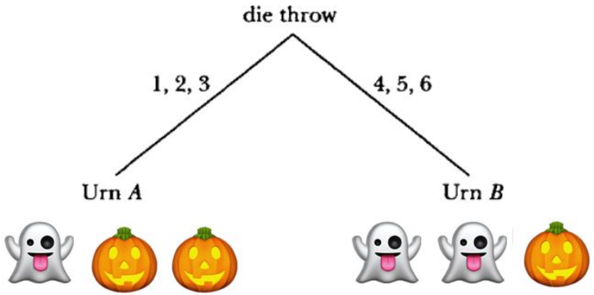


https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

We won't show you the die roll

- But, we draw a single element from the "urn" that was decided based on the die roll
- **What is the probability that what we drew came from Urn A?**

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$



https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

How did you think about the probability?

- Calculation using Bayes' rule?

- Intuition without Bayes’ rule?

Interview question alert!

What is the probability that you roll two fair dice and the sum of their faces is 7?

@Allison inspired by u I've been asking this as an introductory interview question

😆 2

Everyone ive asked so far has taken at least like 15 secs to answer and done it by brute force

😮

Interview question alert!

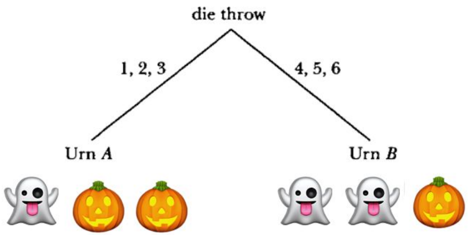
Brute force method:

- [1,6],[2,5],[3,4],[4,3],[5,2],[6,1] → 6 ways to roll a 7
- 6*6 = 36 total ways to roll two dice
- 6/36 = ⅙ probability of rolling a 7

Non-brute-force method:

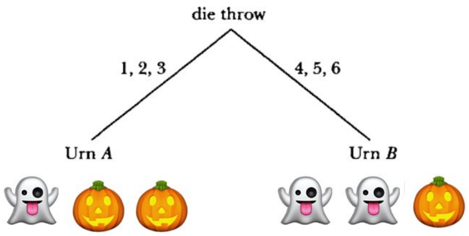
Roll one die. No matter what it lands on, there is exactly one roll by the second die such that the sum equals 7

❤️ 2



If the first draw was a ghost

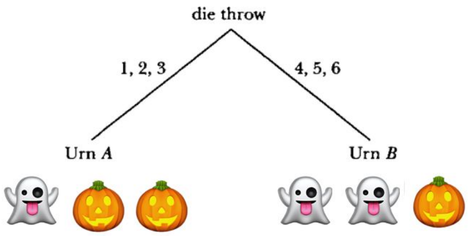
- Calculation using Bayes' rule
  - P(A|ghost) = P(ghost, A) / P(ghost)



If the first draw was a ghost

- Calculation using Bayes' rule
  - P(A|ghost) = P(ghost, A) / P(ghost)

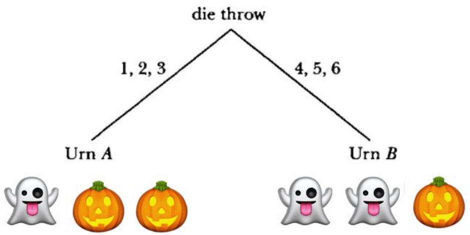
  Conditional Joint / Marginal



If the first draw was a ghost

- Calculation using Bayes' rule
  - P(A|ghost) = P(ghost, A) / P(ghost)
  - P(ghost, A) = joint probability

    = ½ * ⅓ = 1 / 6



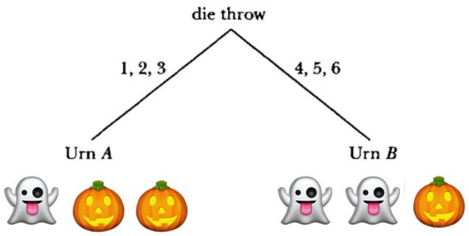
If the first draw was a ghost

- Calculation using Bayes' rule
  - P(A|ghost) = **P(ghost, A)** / P(ghost)
  - **P(ghost, A)** = joint probability

= ½ * ⅓ = 1 / 6

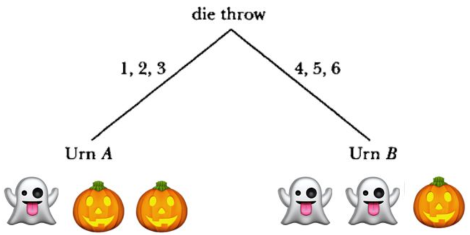
Prob of rolling 1, 2, 3

Within Urn A, getting a ghost



If the first draw was a ghost

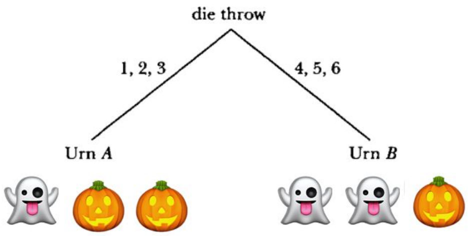
- Calculation using Bayes' rule
  - P(A|ghost) = P(ghost, A) / P(ghost)
  - P(ghost, A) = joint probability = 1 / 6
  - P(ghost) = 1/2



If the first draw was a ghost

- Calculation using Bayes' rule
  - P(A|ghost) = P(ghost, A) / **P(ghost)**
  - P(ghost, A) = joint probability = 1 / 6
  - **P(ghost)** = 1/2

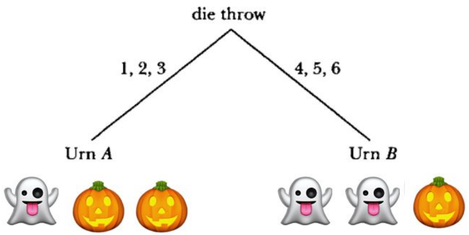
Across the urns, there are 6 items to draw, and 3 of them are ghosts



If the first draw was a ghost

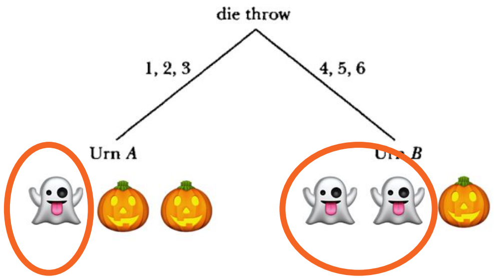
- Calculation using Bayes' rule
  - P(A|ghost) = P(ghost, A) / P(ghost)
  - P(ghost, A) = joint probability = 1 / 6
  - P(ghost) = 1/2

  - P(A|ghost) = ⅙ / ½ = ⅓



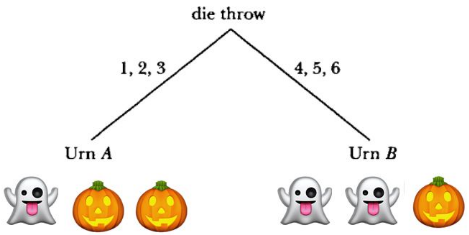
If the first draw was a ghost

- Calculation using Bayes' rule
  - P(A|ghost) = P(ghost, A) / P(ghost)
  - P(A|ghost) = $\frac{1}{6}$ / $\frac{1}{2}$ = **$\frac{1}{3}$**
- Intuition without Bayes' rule
  - ?



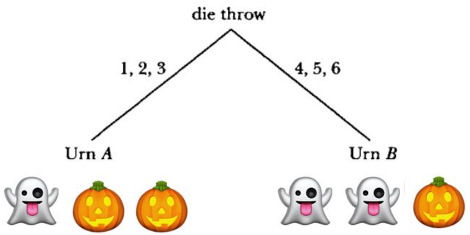
If the first draw was a ghost

- Calculation using Bayes' rule
  - P(A|ghost) = P(ghost, A) / P(ghost)
  - P(A|ghost) = ⅙ / ½ = ⅓
- Intuition without Bayes' rule
  - All 6 items are equally likely to be drawn before the die is thrown! (This is called our *prior*)
  - ⅓ of the ghosts overall are in Urn A



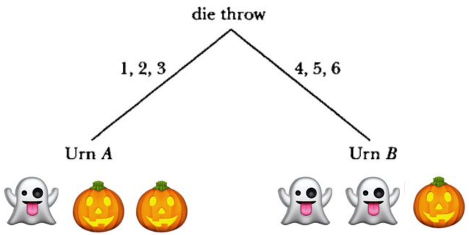
If the first draw was a pumpkin

- Calculation using Bayes' rule
  - P(A|pumpkin) = P(pumpkin, A) / P(pumpkin)
  - P(pumpkin, A) = joint probability = ?
  - P(pumpkin) = ?



If the first draw was a pumpkin

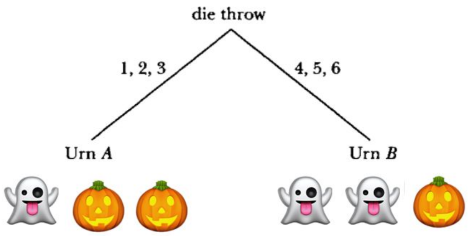
- Calculation using Bayes' rule
  - P(A|pumpkin) = P(pumpkin, A) / P(pumpkin)
  - P(pumpkin, A) = 2/6 = **1/3**
  - P(pumpkin) = **1/2**



If the first draw was a pumpkin

- Calculation using Bayes' rule
  - P(A|pumpkin) = P(pumpkin, A) / P(pumpkin)
  - P(pumpkin, A) = 2/6 = **1/3**
  - P(pumpkin) = **1/2**

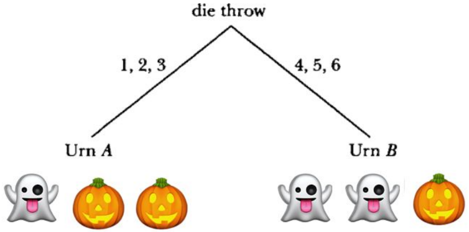
  - Joint probability: Pr(rolling a 1,2,3) * Pr(drawing a pumpkin from urn A) = ½ * ⅔ = 2/6 = **⅓**
  - Marginal: Pr(pumpkin) = 3 pumpkins / 6 items across both urns = **½**



If the first draw was a pumpkin

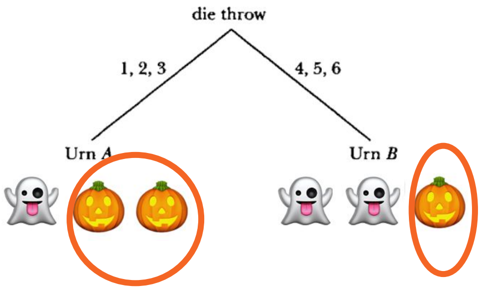
- Calculation using Bayes' rule
  - P(A|pumpkin) = P(pumpkin, A) / P(pumpkin)
  - P(pumpkin, A) = 2/6 = **1/3**
  - P(pumpkin) = **1/2**

  - P(A|pumpkin) = ⅓ / ½ = **⅔**



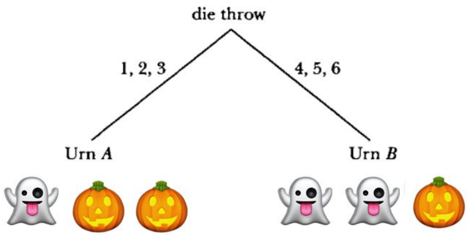
If the first draw was a pumpkin

- Calculation using Bayes' rule
  - P(A|pumpkin) = P(pumpkin, A) / P(pumpkin)
  - P(A|pumpkin) = ⅓ / ½ = ⅔
- Intuition without Bayes' rule
  - ?



If the first draw was a pumpkin

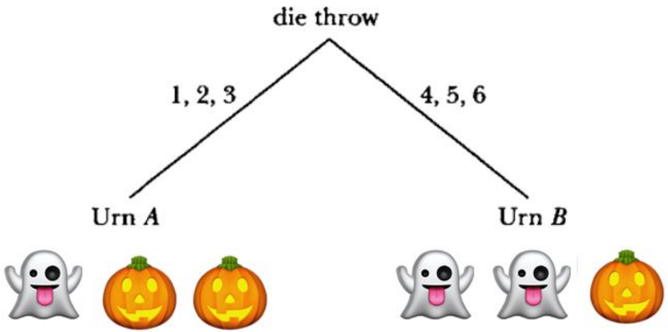
- Calculation using Bayes' rule
  - P(A|pumpkin) = P(pumpkin, A) / P(pumpkin)
  - P(A|pumpkin) = ⅓ / ½ = ⅔
- Intuition without Bayes' rule
  - All 6 items are equally likely to be drawn before the die is thrown! (This is called our *prior*)
  - ⅔ of the pumpkins are in Urn A



Let's assume our first draw was a pumpkin

- Now we replace the pumpkin in the same urn we drew it out of (i.e., we drew *with replacement*)
- And now, we take a second draw from the **same urn** as before

Let's assume our first draw was a pumpkin



- Now we replace the pumpkin in the same urn we drew it out of (i.e., we drew *with replacement*)
- And now, we take a **second** draw from the **same urn** as before
- **Question: what is the probability that what we drew came from Urn A?**

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

How did you think about the probability?

- Calculation using Bayes’ rule?

- Intuition without Bayes’ rule?

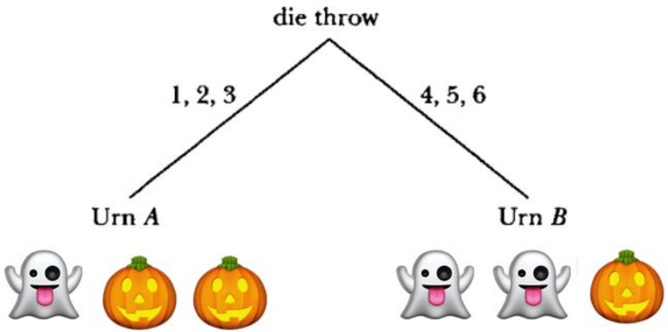
How did you think about the probability?

- Calculation using Bayes’ rule?

- Intuition without Bayes’ rule?

Bayes' rule

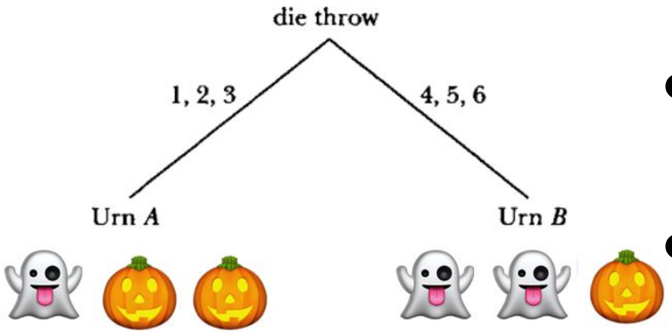
$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$



https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Bayes' rule

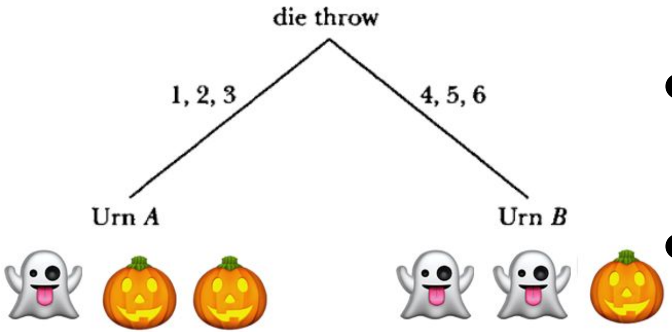
$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$



- First draw from urns was a pumpkin. It is returned to original urn **with replacement**
- And now, we take a second draw from the **same urn**, and get a pumpkin
- What is the **probability we drew from Urn A**?

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Bayes' rule $P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$

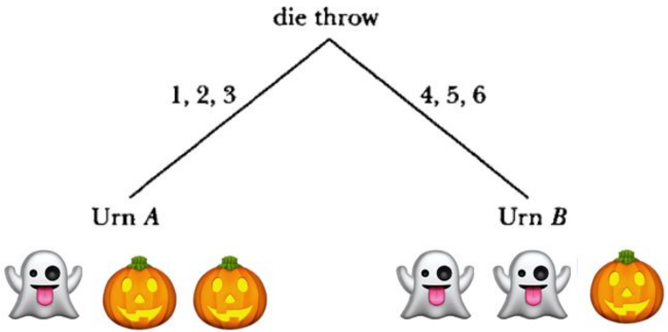


- First draw from urns was a pumpkin. It is returned to original urn **with replacement**
- And now, we take a second draw from the **same urn**, and get a pumpkin
- What is the **probability we drew from Urn A**?
- How do we define A and B if we use Bayes' rule?

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Bayes’ rule

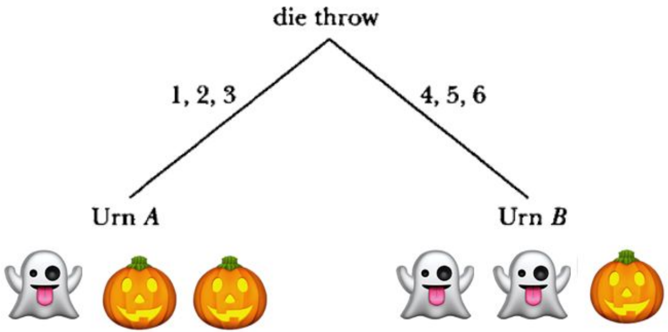
$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$



A = drawing from urn A
B = 1st draw 🎃 and 2nd draw 🎃

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Bayes' rule: solve!



$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

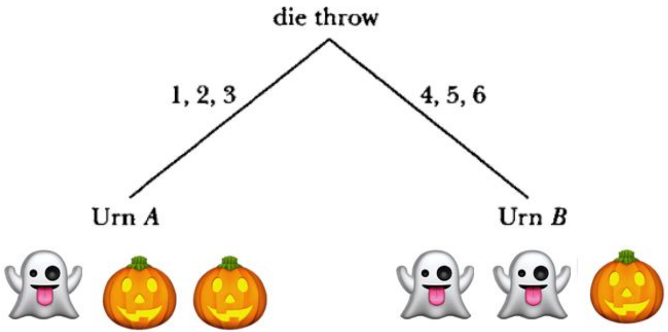
A = drawing from urn A
B = 1st draw 🎃 and 2nd draw 🎃

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Bayes' rule

P(A,B) = ⅔ * ⅔ =

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$



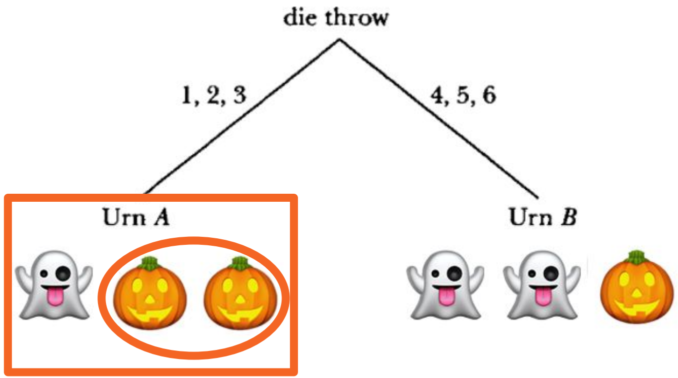
A = drawing from urn A
B = 1st draw 🎃 and 2nd draw 🎃

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Bayes' rule

P(A,B) = ⅔ * ⅔ =

$$P(A \mid B) = \frac{\boxed{P(B \mid A) \cdot P(A)}}{P(B)}$$

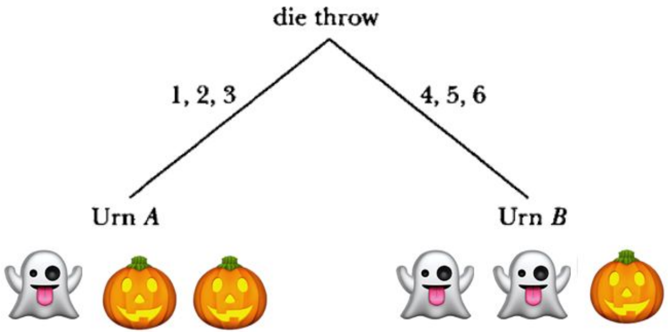


P(1st draw pumpkin, 2nd draw pumpkin, from Urn A) = ⅔ * ⅔ = 4/9

A = drawing from urn A
B = 1st draw 🎃 and 2nd draw 🎃

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Bayes' rule

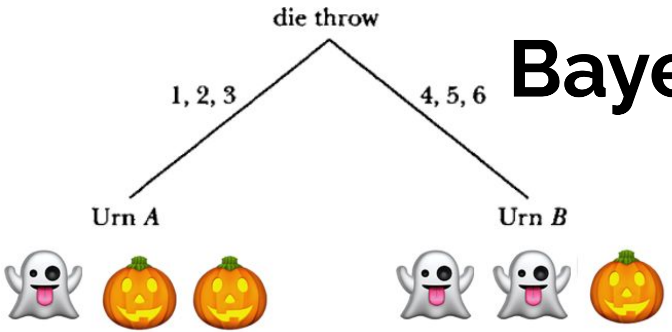


$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

$$= \tfrac{2}{3} * \tfrac{2}{3} + \tfrac{1}{3} * \tfrac{1}{3}$$

A = drawing from urn A
B = 1st draw 🎃 and 2nd draw 🎃

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179



Bayes' rule

- P(1st draw pumpkin, 2nd draw pumpkin, Urn A) = $\frac{2}{3} * \frac{2}{3}$ = 4/9

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

$= \frac{2}{3} * \frac{2}{3} + \frac{1}{3} * \frac{1}{3}$

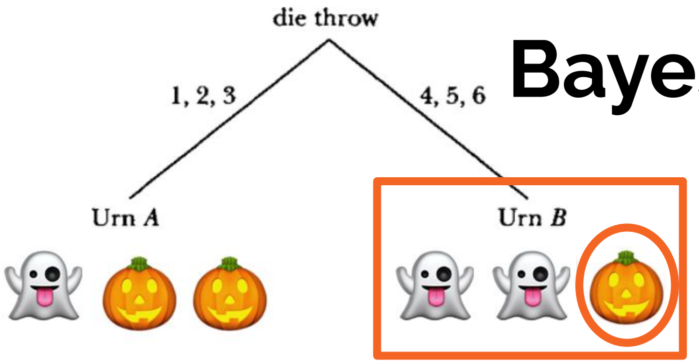
= P(B,A)+P(B,not-A)

A = drawing from urn A

B = 1st draw 🎃 and 2nd draw 🎃

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Bayes' rule



- P(1st draw pumpkin, 2nd draw pumpkin, Urn A) = ⅔ * ⅔ = 4/9
- P(1st draw pumpkin, 2nd draw pumpkin, Urn B) = ⅓ * ⅓ = 1/9

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

= ⅔ * ⅔ + ⅓ * ⅓

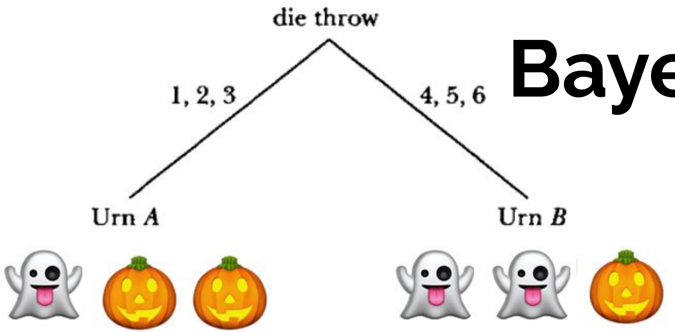
= P(B,A)+P(B,not-A)

A = drawing from urn A

B = 1st draw 🎃 and 2nd draw 🎃

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Bayes' rule



- P(1st draw pumpkin, 2nd draw pumpkin, Urn A) = ⅔ * ⅔ = 4/9
- P(1st draw pumpkin, 2nd draw pumpkin, Urn B) = ⅓ * ⅓ = 1/9
- P(1st draw pumpkin, 2nd draw pumpkin) = P(1st draw pumpkin, 2nd draw ghost, urn A) + P(1st draw pumpkin, 2nd draw ghost, urn B) = 5/9

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

= ⅔ * ⅔ + ⅓ * ⅓

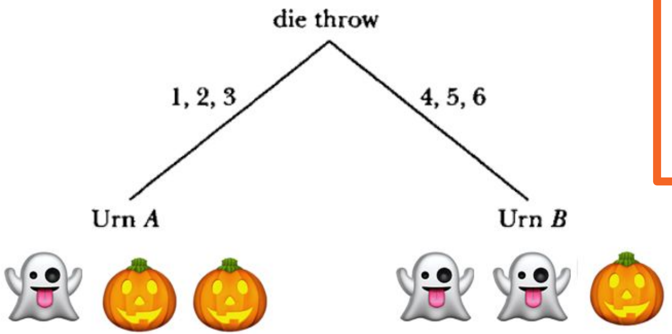
= P(B,A)+P(B,not-A)

A = drawing from urn A

B = 1st draw 🎃 and 2nd draw 🎃

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Bayes' rule



$$P(A,B) = \tfrac{2}{3} * \tfrac{2}{3} =$$

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

$$P(B) = \tfrac{2}{3} * \tfrac{2}{3} + \tfrac{1}{3} * \tfrac{1}{3}$$

$$= (4/9) \,/\, (5/9)$$
$$= 4/5$$

A = drawing from urn A
B = 1st draw 🎃 and 2nd draw 🎃

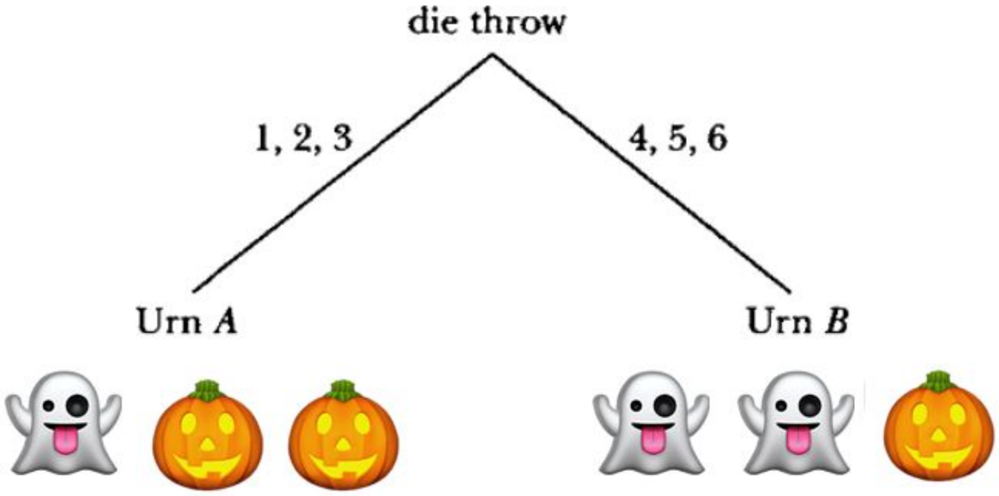
https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

How did you think about the probability?

- Calculation using Bayes' rule?

- Intuition without Bayes' rule?

Can we express the “posterior belief” after the 1st roll?

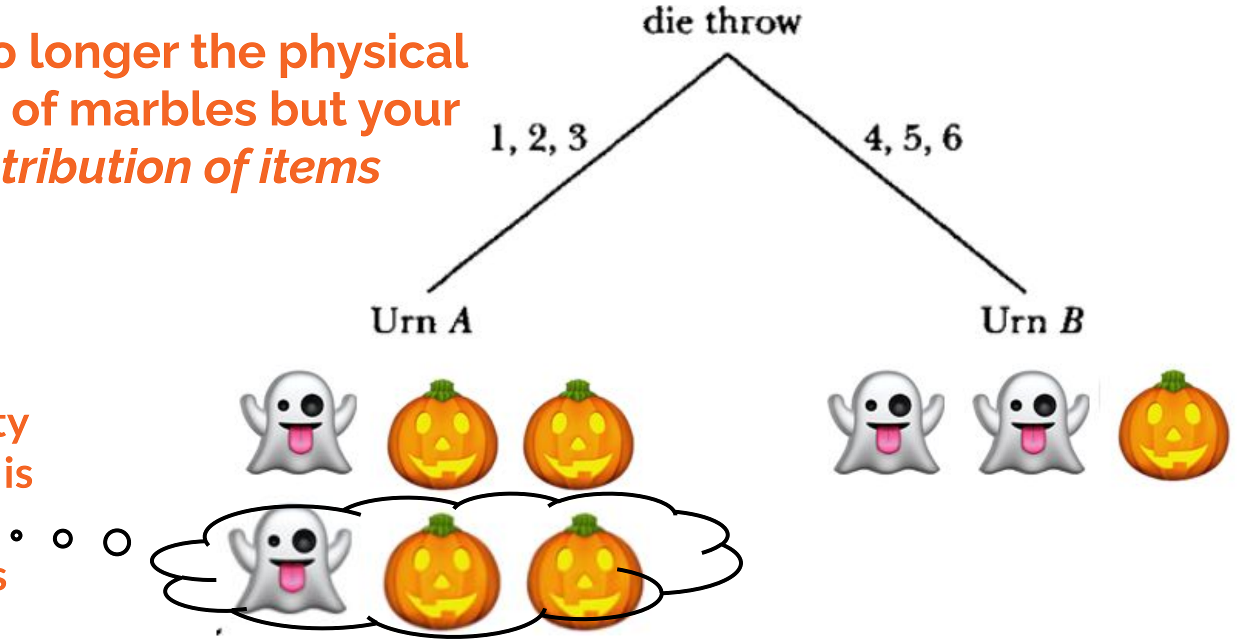


My beliefs before the 2nd draw are that the probability that Urn A is used is ⅔ (i.e. it’s twice as likely that Urn A is used as Urn B).

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Can we express the “posterior belief” after the 1st roll?

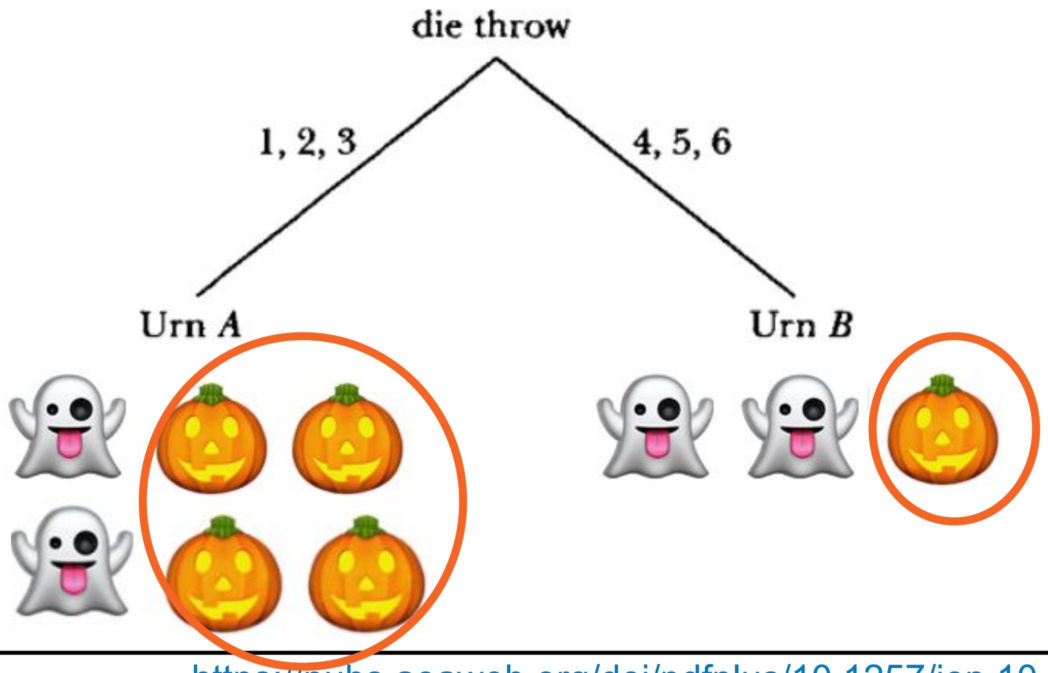
This figure is no longer the physical representation of marbles but your *belief of the distribution of items*



My beliefs before the 2nd draw are that the probability that Urn A is used is ⅔ (i.e. it’s twice as likely that Urn A is used as Urn B).

https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

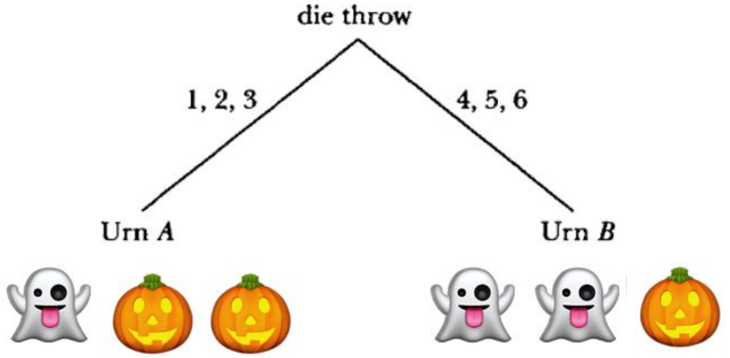
2nd draw a pumpkin: ⅘ prob from A



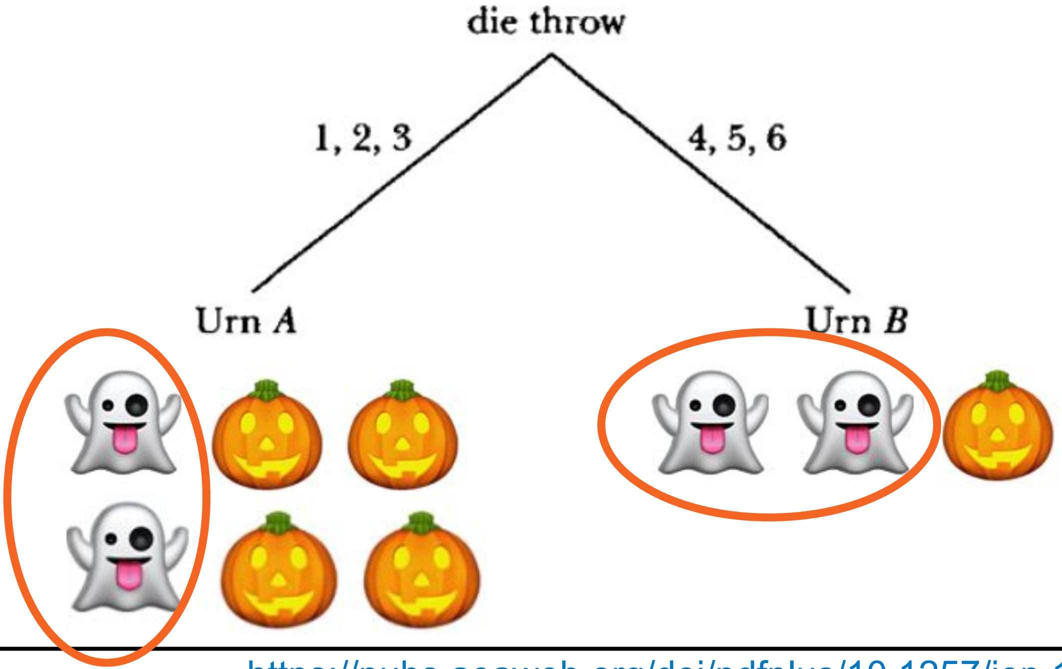
https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

Practice problem for home

- First draw is pumpkin, second draw (with replacement) is ghost. Now, what is the probability that we drew from Urn A?



2nd draw a ghost: ½ prob from A



https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.10.2.179

If the second draw was a ghost

- P(A|1st draw pumpkin, 2nd draw ghost) = P(1st draw pumpkin, 2nd draw ghost, A) / P(1st draw pumpkin, 2nd draw ghost) (Bayes' Rule)
- P(1st draw pumpkin, 2nd draw ghost, A) = ⅔ * ⅓ = 2/9
- P(1st draw pumpkin, 2nd draw ghost, B) = ⅓ * ⅔ = 2/9
- P(1st draw pumpkin, 2nd draw ghost) = P(1st draw pumpkin, 2nd draw ghost, A) + P(1st draw pumpkin, 2nd draw ghost, B) = 4/9 (Marginalizing!)
- P(A|1st draw pumpkin, 2nd draw ghost) = (2/9) / (4/9) = **1/2**

1 min break & attendance

tinyurl.com/yhajjfvk

Now, let's talk about text!

(we promise this will all connect back to Bayes!)

Goal: classify a sentence as positive or negative

A) "I've long been searching for the best camarones diabla and have found that gem here in their camarones endiablados offering."

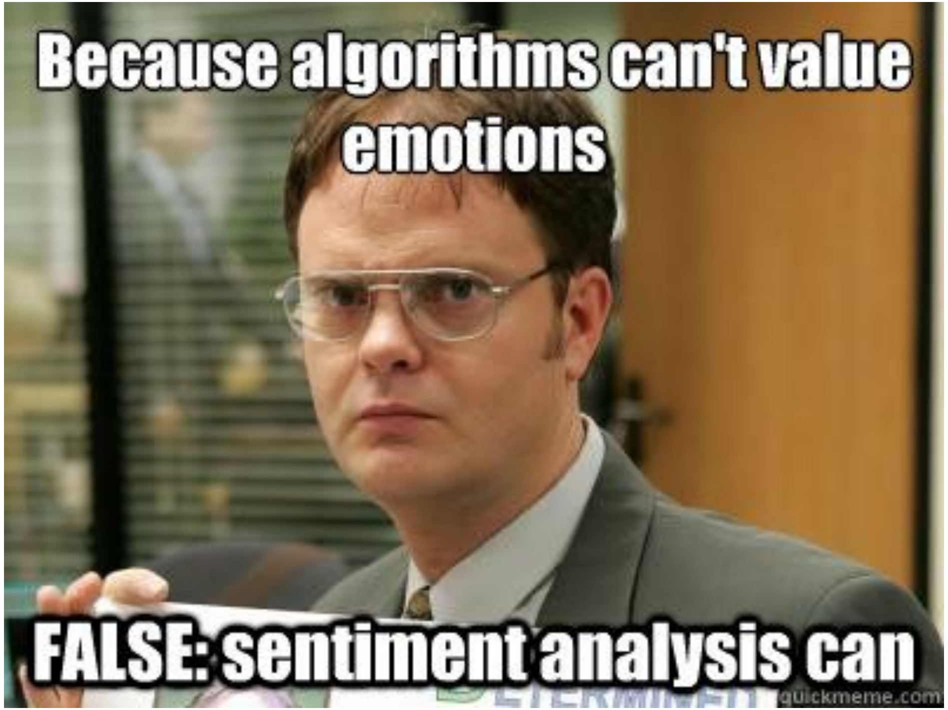
B) "We walked in and nobody greeted us at the entrance and we decided to walk over to one of the open tables."

What do you guess (+ or -)?

Goal: classify a sentence as positive or negative

+ **A)** "I've long been searching for the best camarones diabla and have found that gem here in their camarones endiablados offering." ★★★★★

- **B)** "We walked in and nobody greeted us at the entrance and we decided to walk over to one of the open tables." ★



https://visualist.in/sentiment-analysis-with-textblob/

Apply Bayes' rule

P(★★★★★ | "great food, but service stank")

Apply Bayes' rule

$$P(\star\star\star\star\star \mid \text{"great food, but service stank"}) = \frac{P(\text{"great food, but service stank"} \mid \star\star\star\star\star)\, P(\star\star\star\star\star)}{P(\text{"great food, but service stank"})}$$

Apply Bayes' rule

$$P(★★★★★ \mid \text{"great food, but service stank"}) = \frac{P(\text{"great food, but service stank"} \mid ★★★★★)\, P(★★★★★)}{P(\text{"great food, but service stank"})}$$

Colorless green ideas sleep furiously

Article Talk

From Wikipedia, the free encyclopedia

"Green ideas" redirects here. For the book series, see Green Ideas.

Colorless green ideas sleep furiously was composed by Noam Chomsky in his 1957 book *Syntactic Structures* as an example of a sentence that is grammatically well-formed, but semantically nonsensical. The sentence was originally used in his 1955 thesis *The Logical Structure of Linguistic Theory* and in his 1956 paper "Three Models for the Description of Language".[1]:116

https://en.wikipedia.org/wiki/Colorless_green_ideas_sleep_furiously

But it must be recognized that the notion of "probability of a sentence" is an entirely useless one, under any known interpretation of this term.

https://en.wikipedia.org/wiki/Noam_Chomsky

Colorless green ideas learn furiously: Chomsky and the two cultures of statistical learning

Peter Norvig

First published: 09 August 2012 | https://doi.org/10.1111/j.1740-9713.2012.00590.x | Citations: 4

Abstract

Language recognition programs use massive databases of words, and statistical correlations between those words, to translate or to recognise speech. But correlation is not causation. Do these statistical data-dredgings give any insight into how language works? Or are they a mere big-number trick, useful but adding nothing to understanding? One who holds the latter view is the theorist of language Noam Chomsky. **Peter Norvig** disagrees.

https://rss.onlinelibrary.wiley.com/doi/full/10.1111/j.1740-9713.2012.00590.x

Clearly, it is inaccurate to say that statistical models (and probabilistic models) have achieved limited success; rather they have achieved a dominant (although not exclusive) position.

https://en.wikipedia.org/wiki/Peter_Norvig

Assigning probability to a sentence

P("great food, but service stank" | ★★★★★)

Assigning probability to a sentence

P("great", "food", "but", "service", "stank" | ★★★★★)

Split a string into *tokens*

"the food was great but the service was bad"

Split a string into *tokens*

"the food was great but the service was bad"

["the", "food", "was", "great", "but", "the", "service", "was", "bad"]

We tokenize the string into 9 word <u>tokens</u>

Split a string into *tokens*

"the food was great but the service was bad"

["**the**", "food", "**was**", "great", "but", "**the**", "service", "**was**", "bad"]

{'the': **2**, 'was': **2**, 'food': 1, 'great': 1, 'but': 1, 'service': 1, 'bad': 1}

The string contains 7 distinct word <u>types</u>

Assigning probability to a sentence

P("great food, but service stank" | ★★★★★) =

P("great" | ★★★★★) x

P("food" | ★★★★★) x

P("but" | ★★★★★) x

P("service" | ★★★★★) x

P("stank" | ★★★★★)

Assigning probability to a sentence

Approximate the probability of a sentence by multiplying the probability of each word

P("great food, but service stank" | ★★★★★) =

P("great" | ★★★★★) x

P("food" | ★★★★★) x

P("but" | ★★★★★) x

P("service" | ★★★★★) x

P("stank" | ★★★★★)

Naïve assumption

or

"bag of words" assumption

Assigning probability to a sentence

P("great food, but service stank" | ★★★★★) =

P("great" | ★★★★★) x

P("food" | ★★★★★) x

P("but" | ★★★★★) x

P("service" | ★★★★★) x

P("stank" | ★★★★★)

Bag of words: order doesn't matter

P("great food, but service stank" | ★★★★★) =
P("great service, but food stank" | ★★★★★)

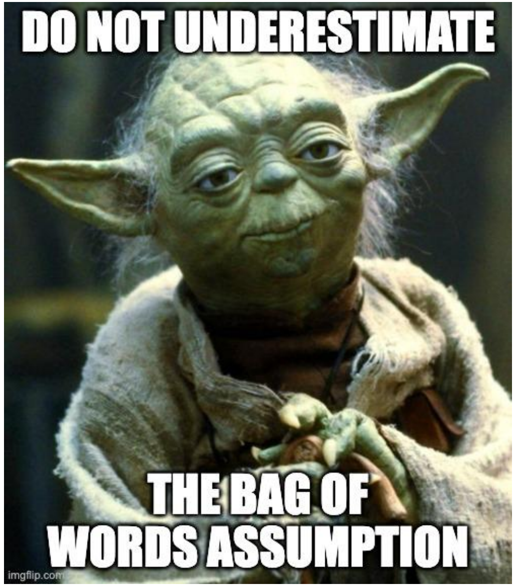
Bag of words: order doesn't matter

P("great food, but service stank" | ★★★★★) =

P("great service, but food stank" | ★★★★★) =

P("stank service food but great" | ★★★★★)

Bag of words: order doesn't matter



P("great food, but service stank" | ★★★★★) =

P("great service, but food stank" | ★★★★★) =

P("stank service food but great" | ★★★★★)

https://imgflip.com/memegenerator/Star-Wars-Yoda

Which source are we sampling?

https://www.walmart.com/ip/M-M-s-Milk-Chocolate-Candy-Full-Size-1-69-oz-Bag/38253480 https://www.amazon.com/Tootsie-Roll-Assorted-Pops-100/dp/B003TRYMHS

Which source are we sampling?

rude worst horrible rude bad dirty disgusting waited worst worse

fantastic best perfect incredible gem BEST perfect delicious LOVE

Multiplying Probabilities

- If P(E) is the probability of event E occurring, and P(F) is the probability of event F occurring:
- E and F are independent events
- **What is the probability that both E and F occur?**

Multiplying Probabilities

- If P(E) is the probability of event E occurring, and P(F) is the probability of event F occurring:
- E and F are independent events
- What is the probability that both E and F occur?
  - P(E and F) = P(E,F) = **P(E) * P(F)**

Probability

- If P(E) is the probability of event E occurring, then we know:

$$\boxed{} \leq \mathrm{P}(E) \leq \boxed{}$$

Probability

- If P(E) is the probability of event E occurring, then we know:

$$0 \leq \mathrm{P}(E) \leq 1$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

In probability theory and computer science, a **log probability** is simply a logarithm of a probability.

https://en.wikipedia.org/wiki/Log_probability

Log Probability

- If P(E) is the probability of event E occurring, then we know:

$$0 \le \mathrm{P}(E) \le 1$$

$$\boxed{\quad\quad} \le \log \mathrm{P}(E) \le \boxed{\quad\quad}$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

- If P(E) is the probability of event E occurring, then we know:

$$0 \leq \mathrm{P}(E) \leq 1$$
$$-\infty \leq \log \mathrm{P}(E) \leq 0$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

- If P(E) is the probability of event E occurring, and P(F) is the probability of event F occurring:

$$\log(\mathrm{P}(E) \cdot \mathrm{P}(F)) = \boxed{} + \boxed{}$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

- If P(E) is the probability of event E occurring, and P(F) is the probability of event F occurring:

$$\log(\mathrm{P}(E) \cdot \mathrm{P}(F)) = \log \mathrm{P}(E) + \log \mathrm{P}(F)$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

- If we already know the values of P(E) and P(F), why would we want to take the logarithm?

$$\log(\mathrm{P}(E) \cdot \mathrm{P}(F)) = \log \mathrm{P}(E) + \log \mathrm{P}(F)$$

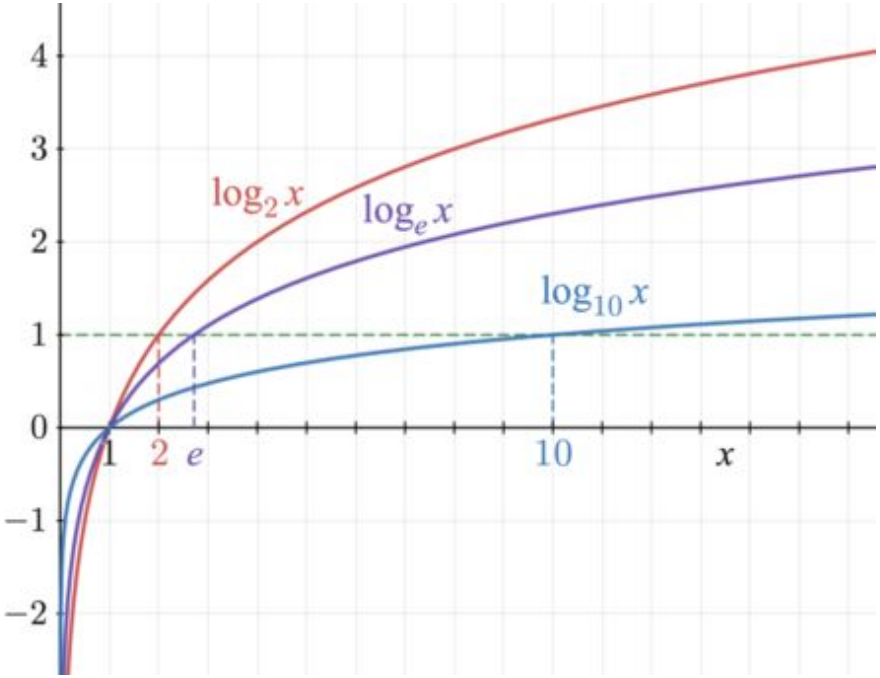
https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

One in	Probability	Log_{10}	Log_e
10	0.1	-1	-2.3
100	0.01	-2	-4.6
1,000	0.001	-3	-6.9
10,000	0.0001	-4	-9.2
100,000	0.00001	-5	-11.5

Logs are much more understandable when probabilities are small!

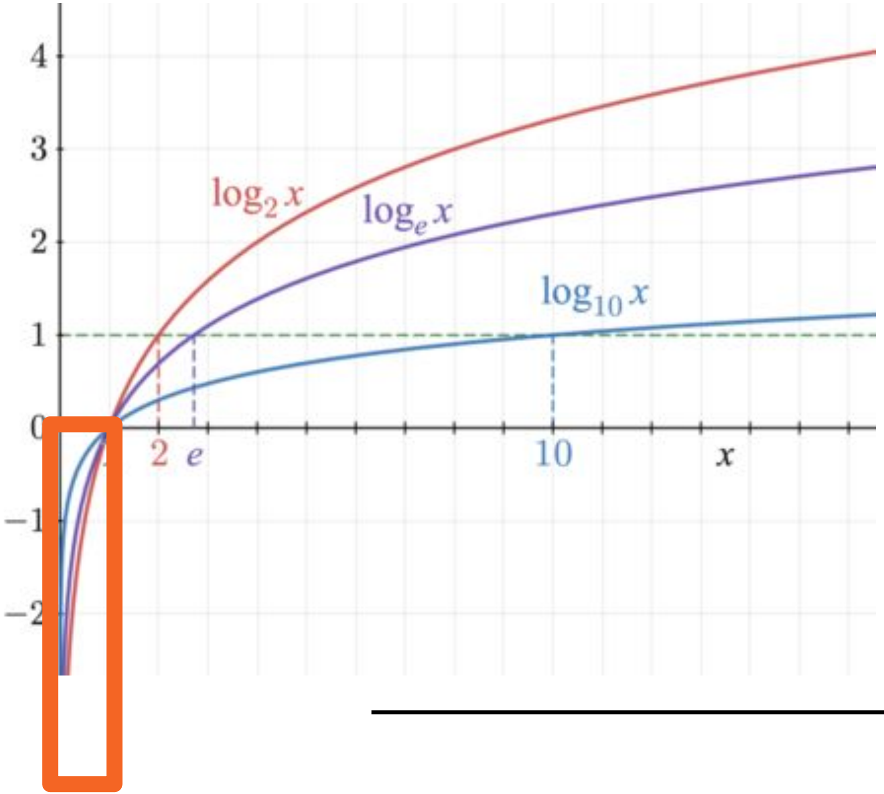
Log Probability



- Before, we've talked about log transforms as "squishing" big numbers

https://en.wikipedia.org/wiki/Logarithm

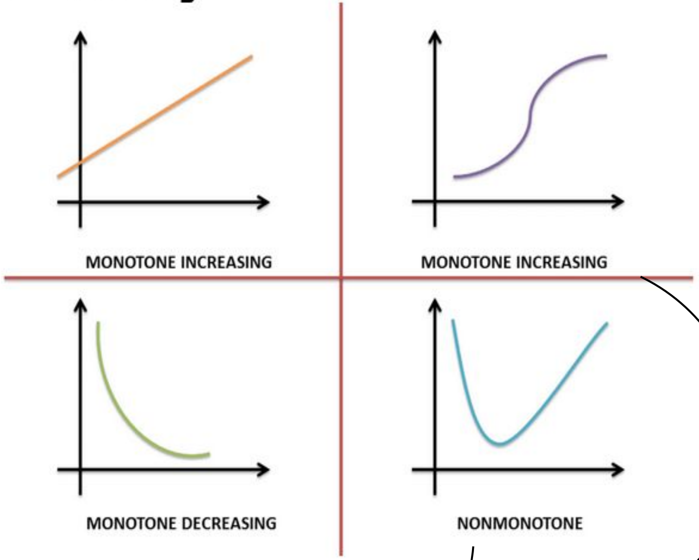
Log Probability



- Before, we've talked about log transforms as "squishing" big numbers
- But now we're constrained to log(P(E)) where 0 < P(E) < 1, where we're actually *not* squishing numbers

https://en.wikipedia.org/wiki/Logarithm

Monotonicity



https://discover.hubpages.com/education/Differences-Between-Spearman-Rank-Correlation-and-Pearson-Correlation 30

Log Probability

- Is the logarithm function (base *e*)
  - **A monotonic function?**
  - **Increasing? Decreasing?**

Log Probability

- Is the logarithm function (base *e*)
    - A monotonic function? **Yes**
    - Increasing? Decreasing? **Increasing**

Log Probability

- Why is it convenient that ln (log base *e*) is a monotonic function?
    - The input point at which *a*b* is maximized is the same input point at which log(a*b) is maximized

Log Probability

- Why is it convenient that ln (log base *e*) is a monotonic function?
    - The input point at which *a*b* is maximized is the same input point at which log(a*b) is maximized
        - This is called the **argmax** because we're sorting by a *function* of the point, not the point itself

Log Probability

- The key is in what we derived before:

$$\log(\mathrm{P}(E) \cdot \mathrm{P}(F)) = \log \mathrm{P}(E) + \log \mathrm{P}(F)$$

Log Probability

- The key is in what we derived before:

$$\log(\mathrm{P}(E) \cdot \mathrm{P}(F)) = \log \mathrm{P}(E) + \log \mathrm{P}(F)$$

$$\log \prod_i \mathrm{P}(E_i) = \sum_i \log \mathrm{P}(E_i)$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

- The key is in what we derived before:

What happens if you multiply a bunch of tiny numbers close to 0?

$$\log \prod_i \mathrm{P}(E_i) = \sum_i \log \mathrm{P}(E_i)$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

- The key is in what we derived before:

What happens if you multiply a bunch of tiny numbers close to 0? **Unless you have a really special computer, you'll get 0**

$$\log \prod_i \mathrm{P}(E_i) = \sum_i \log \mathrm{P}(E_i)$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

- The key is in what we derived before:

The smallest decimal Python can express is 2.225e-308

$$\log \prod_i \mathrm{P}(E_i) = \sum_i \log \mathrm{P}(E_i)$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

- The key is in what we derived before:

Do we run into small number issues if we do sums instead?

$$\log \prod_i \mathrm{P}(E_i) = \sum_i \log \mathrm{P}(E_i)$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

- The key is in what we derived before:

No issues! log(2.225e-308) = -307.652, easy to store in Python

$$\log \prod_i \mathrm{P}(E_i) = \sum_i \log \mathrm{P}(E_i)$$

https://chrispiech.github.io/probabilityForComputerScientists/en/part1/log_probabilities/

Log Probability

Simulate small probabilities

```
>>> import numpy as np

>>> r = np.random.random_sample((100,)) * 0.000001
```

https://math.stackexchange.com/questions/892832/why-we-consider-log-likelihood-instead-of-likelihood-in-gaussian-distribution

Log Probability

```
>>> import numpy as np

>>> r = np.random.random_sample((100,)) * 0.000001

>>> np.product(r)
```

What will we get?

https://math.stackexchange.com/questions/892832/why-we-consider-log-likelihood-instead-of-likelihood-in-gaussian-distribution

Log Probability

```
>>> import numpy as np

>>> r = np.random.random_sample((100,)) * 0.000001

>>> np.product(r)
0.0
```

Not the actual product – too small for Python!

https://math.stackexchange.com/questions/892832/why-we-consider-log-likelihood-instead-of-likelihood-in-gaussian-distribution

Log Probability

```
>>> import numpy as np

>>> r = np.random.random_sample((100,)) * 0.000001

>>> np.product(r)
0.0

>>> log_r = [np.log(x) for x in r]
```

Let’s try taking logs now

https://math.stackexchange.com/questions/892832/why-we-consider-log-likelihood-instead-of-likelihood-in-gaussian-distribution

Log Probability

```
>>> import numpy as np

>>> r = np.random.random_sample((100,)) * 0.000001

>>> np.product(r)
0.0

>>> log_r = [np.log(x) for x in r]

>>> np.sum(log_r)
```

Do we expect a weird number now?

https://math.stackexchange.com/questions/892832/why-we-consider-log-likelihood-instead-of-likelihood-in-gaussian-distribution

Log Probability

```
>>> import numpy as np

>>> r = np.random.random_sample((100,)) * 0.000001

>>> np.product(r)
0.0

>>> log_r = [np.log(x) for x in r]

>>> np.sum(log_r)
-1472.245511811776
```

No Python issues here!

https://math.stackexchange.com/questions/892832/why-we-consider-log-likelihood-instead-of-likelihood-in-gaussian-distribution

Log Probability Takeaways

- If you’re using a computer to help you calculate the product of probabilities, you should nearly always just use log probabilities instead
    - They are harder for humans to interpret…
    - but will allow for more accurate computation
    - and allow for finding the same maximizing points due to monotonicity

1 min break (stare at this table)

One in	Probability	Log_{10}	Log_e
10	0.1	-1	-2.3
100	0.01	-2	-4.6
1,000	0.001	-3	-6.9
10,000	0.0001	-4	-9.2
100,000	0.00001	-5	-11.5

Log prob quiz!

What probability has $\log_e$ **-2.3**?

One in ______

What is the $\log_e$ of **One in 100,000**?

______

Log prob quiz!

What probability has $\log_e$ **-2.3**?

One in __**10**___

What is the $\log_e$ of **One in 100,000**?

__**-11.5**____

Classifying text: sports or not?

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

Training data

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

Training data

Want to classify new data

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports
"A very close game"	

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

In math: P(____ | _____)?

Want to classify new data

Text	Tag
"A very close game"	Sports / Not Sports?

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

In math: P(Sports | “A very close game”)

Want to classify new data

Text	Tag
"A very close game"	Sports / Not Sports?

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

In math: P(Sports | “A very close game”)

Notice: we aren’t outputting a binary estimate of Sports or Not Sports – we’re outputting an estimate of the probability that our tag is Sports

Want to classify new data

Text	Tag
"A very close game"	Sports / Not Sports?

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

In math: P(Sports | “A very close game”)

“Sports” here is a binary variable: it can take either value 0 or 1. We can, for example, write a subset probability: P(Sports=1 | “A very close game”)

Want to classify new data

Text	Tag
"A very close game"	Sports / Not Sports?

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

How do we use training data to calculate P(Sports | "A very close game")?

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

First, let's apply Bayes' theorem:

$$P(\text{Sports} \mid \text{“A very close game”}) = \frac{P(\text{“A very close game”} \mid \text{Sports}) * P(\text{Sports})}{P(\text{“A very close game”})}$$

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

First, let's apply Bayes' theorem:

$$P(\text{Sports} \mid \text{"A very close game"}) = \frac{P(\text{"A very close game"} \mid \text{Sports}) * P(\text{Sports})}{P(\text{"A very close game"})}$$

Remember we're being naive today... let's assume all the words are *independent* (Yoda)

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

Assuming words are independent:

$$P(\text{Sports} \mid \text{“A”, “very”, “close”, game”}) = \frac{P(\text{“A”, “very”, “close”, “game”} \mid \text{Sports}) * P(\text{Sports})}{P(\text{“A”, “very”, “close”, “game”})}$$

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

$$P(\text{Sports} \mid \text{“A”, “very”, “close”, game”}) = \frac{P(\text{“A”, “very”, “close”, “game”} \mid \text{Sports}) * P(\text{Sports})}{P(\text{“A”, “very”, “close”, “game”})}$$

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

We can reduce further using the “chain rule”

$$P(\text{Sports} \mid \text{“A”, “very”, “close”, game”}) = \frac{\boxed{P(\text{“A”, “very”, “close”, “game”} \mid \text{Sports})} * P(\text{Sports})}{P(\text{“A”, “very”, “close”, “game”})}$$

$$P(y|x_1, \ldots, x_n) = \frac{\boxed{P(x_1|y)P(x_2|y)\ldots P(x_n|y)}P(y)}{P(x_1)P(x_2)\ldots P(x_n)}$$

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports | “A very close game”) =

$$\left[\frac{P(\text{“A”}|\text{Sports})*P(\text{“very”}|\text{Sports})*P(\text{“close”}|\text{Sports})*P(\text{“game”}|\text{Sports}) * P(\text{Sports})}{P(\text{“A”}, \text{“very”}, \text{“close”}, \text{“game”})} \right]$$

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports | "A very close game") =

$$\left[\frac{P(\text{"A"}|\text{Sports}) * P(\text{"very"}|\text{Sports}) * P(\text{"close"}|\text{Sports}) * P(\text{"game"}|\text{Sports}) * P(\text{Sports})}{P(\text{"A"}, \text{"very"}, \text{"close"}, \text{"game"})} \right]$$

We can reduce this slightly more!

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

$$P(y|x_1, ..., x_n) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$$

https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

$$P(y|x_1, ..., x_n) = \frac{\boxed{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}}{P(x_1)P(x_2)...P(x_n)}$$

$$P(y|x_1, ..., x_n) \propto \boxed{P(y)\prod_{i=1}^{n} P(x_i|y)}$$

These are the same; big Pi means multiplying (the same way big Sigma means summing)

https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

$$P(y|x_1, ..., x_n) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$$

alpha here means “proportional to” (e.g., instead “=” which means “equals to”)

https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

$$P(y|x_1, ..., x_n) \propto \boxed{P(y) \prod_{i=1}^{n} P(x_i|y)}$$

This is just the numerator of the above expression! What happened to the denominator?

https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

$$P(y|x_1, ..., x_n) \propto P(y)\prod_{i=1}^{n} P(x_i|y)$$

The goal of calculating $P(y|x_1, ..., x_n)$ is to find the argmax of it!

$$y = argmax_y P(y)\prod_{i=1}^{n} P(x_i|y)$$

https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

$$P(y|x_1,...,x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

$$P(y|x_1,...,x_n) \propto \boxed{P(y)\prod_{i=1}^{n} P(x_i|y)}$$

$$y = argmax_y \boxed{P(y)\prod_{i=1}^{n} P(x_i|y)}$$

https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{\boxed{P(x_1)P(x_2)...P(x_n)}}$$

$$P(y|x_1, ..., x_n) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$$

It turns out the denominator isn't affected by y, so it'll just be a constant that's irrelevant for calculating the argmax!

$$y = argmax_y P(y) \prod_{i=1}^{n} P(x_i|y)$$

https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

What does this mean in words? Let's switch back to our example...

$$y = argmax_y P(y) \prod_{i=1}^{n} P(x_i|y)$$

https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports | “A very close game”) =

$$\left[\frac{P(\text{“A”}|\text{Sports})*P(\text{“very”}|\text{Sports})*P(\text{“close”}|\text{Sports})*P(\text{“game”}|\text{Sports}) * P(\text{Sports})}{P(\text{“A”}, \text{“very”}, \text{“close”}, \text{“game”})}\right]$$

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports | "A very close game") =

$$\left[\frac{P(\text{"A"}|\text{Sports})*P(\text{"very"}|\text{Sports})*P(\text{"close"}|\text{Sports})*P(\text{"game"}|\text{Sports}) * P(\text{Sports})}{P(\text{"A", "very", "close", "game"})}\right]$$

We have y = Sports, and x = "A very close game"

We want to pick the value of y (Sports = 1 or Sports = 0) such that the probability above is maximized.

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports | “A very close game”) =

$$\left[\frac{P(\text{“A”}|\text{Sports})*P(\text{“very”}|\text{Sports})*P(\text{“close”}|\text{Sports})*P(\text{“game”}|\text{Sports}) * P(\text{Sports})}{P(\text{“A”}, \text{“very”}, \text{“close”}, \text{“game”})}\right]$$

P(“A”, “very”, “close”, “game”) is *the same* for Sports and NotSports.

We want to pick the value of y (Sports = 1 or Sports = 0) such that the probability above is maximized.

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports | “A very close game”) =

$$\left[\frac{P(\text{“A”}|\text{Sports})*P(\text{“very”}|\text{Sports})*P(\text{“close”}|\text{Sports})*P(\text{“game”}|\text{Sports}) * P(\text{Sports})}{P(\text{“A”}, \text{“very”}, \text{“close”}, \text{“game”})}\right]$$

$$\propto P(\text{“A”}|\text{Sports})*P(\text{“very”}|\text{Sports})*P(\text{“close”}|\text{Sports})*P(\text{“game”}|\text{Sports}) * P(\text{Sports})$$

alpha-looking thing means “proportional to”

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

To calculate the probability P(Sports | “A very close game”), we need the following ingredients:

- P(“A”|Sports)
- P(“very”|Sports)
- P(“close”|Sports)
- P(“game”|Sports)
- P(Sports)

How do we get P(“word”|Sports)??

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Generate word frequencies!

Training Data

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

Joint Count Data

Word	# in Sports	# in Not sports
“a”		
“very”		
“close”		
“game”		

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Generate word frequencies!

Training Data

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

Joint Count Data

Word	# in Sports	# in Not sports
“a”	2	1
“very”		
“close”		
“game”		

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Generate word frequencies!

Training Data

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

Joint Count Data

Word	# in Sports	# in Not sports
“a”	2	1
“very”	1	0
“close”		
“game”		

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Generate word frequencies!

Training Data

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

Joint Count Data

Word	# in Sports	# in Not sports
“a”	2	1
“very”	1	0
“close”	?	?
“game”		

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Generate word frequencies!

Training Data

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a **close** election"	Not sports

Joint Count Data

Word	# in Sports	# in Not sports
“a”	2	1
“very”	1	0
“close”	**0**	**1**
“game”		

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Generate word frequencies!

Training Data

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

Joint Count Data

Word	# in Sports	# in Not sports
“a”	2	1
“very”	1	0
“close”	0	1
“game”	2	0

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

For P(Sports = 1 | “A very close game”), multiply:

- P(“A”|Sports)
- P(“very”|Sports)
- P(“close”|Sports)
- P(“game”|Sports)
- P(Sports)

For P(Sports = 0 | “A very close game”), multiply:

- P(“A”|Not sports)
- P(“very”|Not sports)
- P(“close”|Not sports)
- P(“game”|Not sports)
- P(Not sports)

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

For P(Sports = 1 | “A very close game”), multiply:

- P(“A”|Sports)
- P(“very”|Sports)
- P(“close”|Sports)
- P(“game”|Sports)
- P(Sports)

For P(Sports = 0 | “A very close game”), multiply:

- P(“A”|Not sports)
- P(“very”|Not sports)
- P(“close”|Not sports)
- P(“game”|Not sports)
- P(Not sports)

Goal: compare which probability is bigger!

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

For P(Sports = 1 | “A very close game”), multiply:

- P(“A”|Sports)
- P(“very”|Sports)
- P(“close”|Sports)
- P(“game”|Sports)
- P(Sports)

For P(Sports = 0 | “A very close game”), multiply:

- P(“A”|Not sports)
- P(“very”|Not sports)
- P(“close”|Not sports)
- P(“game”|Not sports)
- P(Not sports)

Our joint count table gave us these numbers

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

For P(Sports = 1 | "A very close game"), multiply:

- P("A"|Sports)
- P("very"|Sports)
- P("close"|Sports)
- P("game"|Sports)
- P(Sports)

For P(Sports = 0 | "A very close game"), multiply:

- P("A"|Not sports)
- P("very"|Not sports)
- P("close"|Not sports)
- P("game"|Not sports)
- P(Not sports)

But we still don't have these probabilities!

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

P(Sports) = ?

P(Not sports) = ?

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

P(Sports) = 3/5

P(Not sports) = 2/5

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

P(Sports) = 3/5
P(Not sports) = 2/5

 # words in Sports = 11
 # words in Not sports = 9

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

P(Sports) = 3/5
P(Not sports) = 2/5

 # words in Sports = 11
 # words in Not sports = 9

Word	# in Sports	# in Not sports	P(Word|Sports)	P(Word|Not sports)
“a”	2	1	?	?
“very”	1	0	?	?
“close”	0	1	?	?
“game”	2	0	?	?

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

P(Sports) = 3/5
P(Not sports) = 2/5

 # words in Sports = 11
 # words in Not sports = 9

Word	# in Sports	# in Not sports	P(Word|Sports)	P(Word|Not sports)
“a”	2	1	**2/11**	**1/9**
“very”	1	0	**1/11**	**0/9**
“close”	0	1	**0/11**	**1/9**
“game”	2	0	**2/11**	**0/9**

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

We want to multiply…

- P("A"|Sports) = 2/11
- P("very"|Sports) = 1/11
- P("close"|Sports) = 0/11
- P("game"|Sports) = 2/11
- P(Sports) = 3/5

Word	# in Sports	# in Not sports	P(Word|Sports)	P(Word|Not sports)
"a"	2	1	**2/11**	**1/9**
"very"	1	0	**1/11**	**0/9**
"close"	0	1	**0/11**	**1/9**
"game"	2	0	**2/11**	**0/9**

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

We'll just get a 0 probability since "close" never appeared in the Sports=1 training data!

That's not fair, so we have to correct for it somehow…

Word	# in Sports	# in Not sports	P(Word|Sports)	P(Word|Not sports)
"a"	2	1	**2/11**	**1/9**
"very"	1	0	**1/11**	**0/9**
"close"	0	1	**0/11**	**1/9**
"game"	2	0	**2/11**	**0/9**

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Laplace Smoothing: add 1 to every count so nothing is ever zero!

Word	# in Sports	# in Not sports	P(Word|Sports)	P(Word|Not sports)
"a"	2	1	**2/11**	**1/9**
"very"	1	0	**1/11**	**0/9**
"close"	0	1	**0/11**	**1/9**
"game"	2	0	**2/11**	**0/9**

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

With Laplace Smoothing:

Every probability *numerator* increases by *1*

Word	# in Sports	# in Not sports	P(Word|Sports)	P(Word|Not sports)
"a"	2	1	**3/25**	**2/23**
"very"	1	0	**2/25**	**1/23**
"close"	0	1	**1/25**	**2/23**
"game"	2	0	**3/25**	**1/23**

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

With Laplace Smoothing:

Every probability *denominator* increases by the *distinct word count* across all the training data

Word	# in Sports	# in Not sports	P(Word|Sports)	P(Word|Not sports)
“a”	2	1	**3/25**	**2/23**
“very”	1	0	**2/25**	**1/23**
“close”	0	1	**1/25**	**2/23**
“game”	2	0	**3/25**	**1/23**

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

With Laplace Smoothing:

Distinct word count overall = 14
 # words in Sports = 11+14 = 25
 # words in Not sports = 9+14 = 23

Word	# in Sports	# in Not sports	P(Word|Sports)	P(Word|Not sports)
“a”	2	1	3/25	2/23
“very”	1	0	2/25	1/23
“close”	0	1	1/25	2/23
“game”	2	0	3/25	1/23

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

P(Sports) = ⅗
P(Not sports) = ⅖

Word	P(Word|Sports)	P(Word|Not sports)
"a"	3/25	2/23
"very"	2/25	1/23
"close"	1/25	2/23
"game"	3/25	1/23

P(Sports=1 | "A very close game") is the multiplication of:

- P("A"|Sports) = 3/25
- P("very"|Sports) = 2/25
- P("close"|Sports) = 1/25
- P("game"|Sports) = 3/25
- P(Sports) = 3/5

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

P(Sports=1 | "A very close game") is the multiplication of:

- P("A"|Sports) = 3/25
- P("very"|Sports) = 2/25
- P("close"|Sports) = 1/25
- P("game"|Sports) = 3/25
- P(Sports) = 3/5

P(Sports) = ⅗
P(Not sports) = ⅖

Word	P(Word|Sports)	P(Word|Not sports)
"a"	3/25	2/23
"very"	2/25	1/23
"close"	1/25	2/23
"game"	3/25	1/23

We got lucky this time because this is calculable in Python still... 0.000027648

How else can we deal with this P?

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Log[P(Sports=1 | “A very close game”)] is the sum of:

P(Sports) = ⅗
P(Not sports) = ⅖

Word	P(Word|Sports)	P(Word|Not sports)
“a”	3/25	2/23
“very”	2/25	1/23
“close”	1/25	2/23
“game”	3/25	1/23

- Log[P(“A”|Sports)] = log(3/25)
- Log[P(“very”|Sports)] = log(2/25)
- Log[P(“close”|Sports)] = log(1/25)
- Log[P(“game”|Sports)] = log(3/25)
- Log[P(Sports)] = log(3/5)

This gives us -10.5

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Log prob quiz!

What probability has $\log_e$ **-2.3**?

One in ______

What is the $\log_e$ of **One in 100,000**?

______

Log prob quiz!

What probability has $\log_e$ **-2.3**?

One in __**10**___

What is the $\log_e$ of **One in 100,000**?

__**-11.5**___

Log[P(Sports=1 | “A very close game”)] is the sum of:

P(Sports) = ⅗
P(Not sports) = ⅖

Word	P(Word|Sports)	P(Word|Not sports)
“a”	3/25	2/23
“very”	2/25	1/23
“close”	1/25	2/23
“game”	3/25	1/23

- Log[P(“A”|Sports)] = log(3/25)
- Log[P(“very”|Sports)] = log(2/25)
- Log[P(“close”|Sports)] = log(1/25)
- Log[P(“game”|Sports)] = log(3/25)
- Log[P(Sports)] = log(3/5)

This gives us -10.5

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Classifying text: sports or not?

For P(Sports = 1 | “A very close game”), multiply:

- P(“A”|Sports)
- P(“very”|Sports)
- P(“close”|Sports)
- P(“game”|Sports)
- P(Sports)

For P(Sports = 0 | “A very close game”), multiply:

- P(“A”|Not sports)
- P(“very”|Not sports)
- P(“close”|Not sports)
- P(“game”|Not sports)
- P(Not sports)

Goal: compare which probability is bigger!

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports = 1 | “A very close game”)

α 0.000027648

P(Sports = 0 | “A very close game”)

α 0.00000572

Goal: compare which probability is bigger!

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports = 1 | “A very close game”)

ɑ 0.000027648

P(Sports = 0 | “A very close game”)

ɑ 0.00000572

Normalize by 0.000027648+0.00000572 = 0.000033368

= 0.000027648 / 0.000033368 ≈ 83%

= 0.00000572 / 0.000033368 ≈ 17%

If we want to compare this without ‘proportional to’...

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports = 1 | “A very close game”)

α 0.000027648

Log[P(Sports = 1 | “A very close game”)]

= -10.5

P(Sports = 0 | “A very close game”)

α 0.00000572

Log[P(Sports = 0 | “A very close game”)]

= -12.07

Goal: compare which probability is bigger!

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports = 1 | “A very close game”)

α 0.000027648

Log[P(Sports = 1 | “A very close game”)]

= -10.5

Better than 1 in 100k

P(Sports = 0 | “A very close game”)

α 0.00000572

Log[P(Sports = 0 | “A very close game”)]

= -12.07

Goal: compare which probability is bigger!

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports = 1 | "A very close game")

α 0.000027648

Log[P(Sports = 1 | "A very close game")]

= -10.5

P(Sports = 0 | "A very close game")

α 0.00000572

Log[P(Sports = 0 | "A very close game")]

= -12.07

Worse than 1 in 100k

Goal: compare which probability is bigger!

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Classifying text: sports or not?

P(Sports = 1 | "A very close game")

= 0.000027648

Log[P(Sports = 1 | "A very close game")]

= -10.5

P(Sports = 0 | "A very close game")

= 0.00000572

Log[P(Sports = 0 | "A very close game")]

= -12.07

It's more likely that "A very close game" is about Sports!

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/, https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

Probability ratios

$$\log\left(\frac{P(Sports = 1|text)}{P(Sports = 0|text)}\right) \quad =$$

$$\log P(Sports = 1|text) - \log P(Sports = 0|text)$$

Probability ratios

$$\log\left(\frac{P(Sports = 1|text)}{P(Sports = 0|text)}\right) =$$

$$\log P(Sports = 1|text) - \log P(Sports = 0|text)$$

= -10.5 - (-12.07) = 1.57

Probability ratios

$$\log\left(\frac{P(Sports = 1|text)}{P(Sports = 0|text)}\right) =$$

$$\log P(Sports = 1|text) - \log P(Sports = 0|text)$$

= -10.5 - (-12.07) = 1.57

Exponentiating gives us how many times more likely this text is Sports is vs. Not sports

Probability ratios

$$\log\left(\frac{P(Sports=1|text)}{P(Sports=0|text)}\right) =$$

$$\log P(Sports=1|text) - \log P(Sports=0|text)$$

= -10.5 - (-12.07) = 1.57

Exponentiating gives us how many times more likely this text is Sports is vs. Not sports: $e^{1.57}$ = 4.8 times more likely

Classifying text: sports or not?

- This process was called **Naive Bayes**
    - Get frequencies of your data units (e.g. word)
    - Calculate probabilities assuming independence of data units
    - Use Bayes' rule to determine what classification has the highest probability

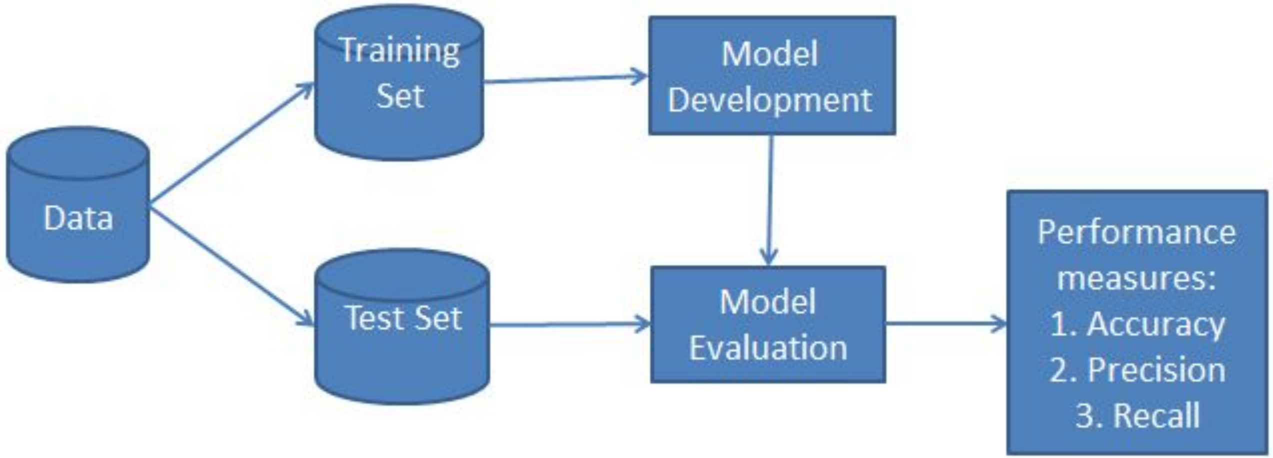
Naive Bayes

- **Naive Bayes allows us to classify our outcomes**
  - These can be binary (Sports / not sports) or multi-category (sport A / sport B / elections)
  - We calculate probabilities based on the frequencies of these categories in our data for each independent input x (whether x_i's are words or other df inputs)

Gaussian Naive Bayes

- **Naive Bayes allows us to classify our outcomes**
  - These can be binary (Sports / not sports) or multi-category (sport A / sport B / elections)
  - We calculate probabilities based on the normal probability density function ~~frequencies of these categories~~ in our data for each independent input x (whether x_i's are words or other df inputs)

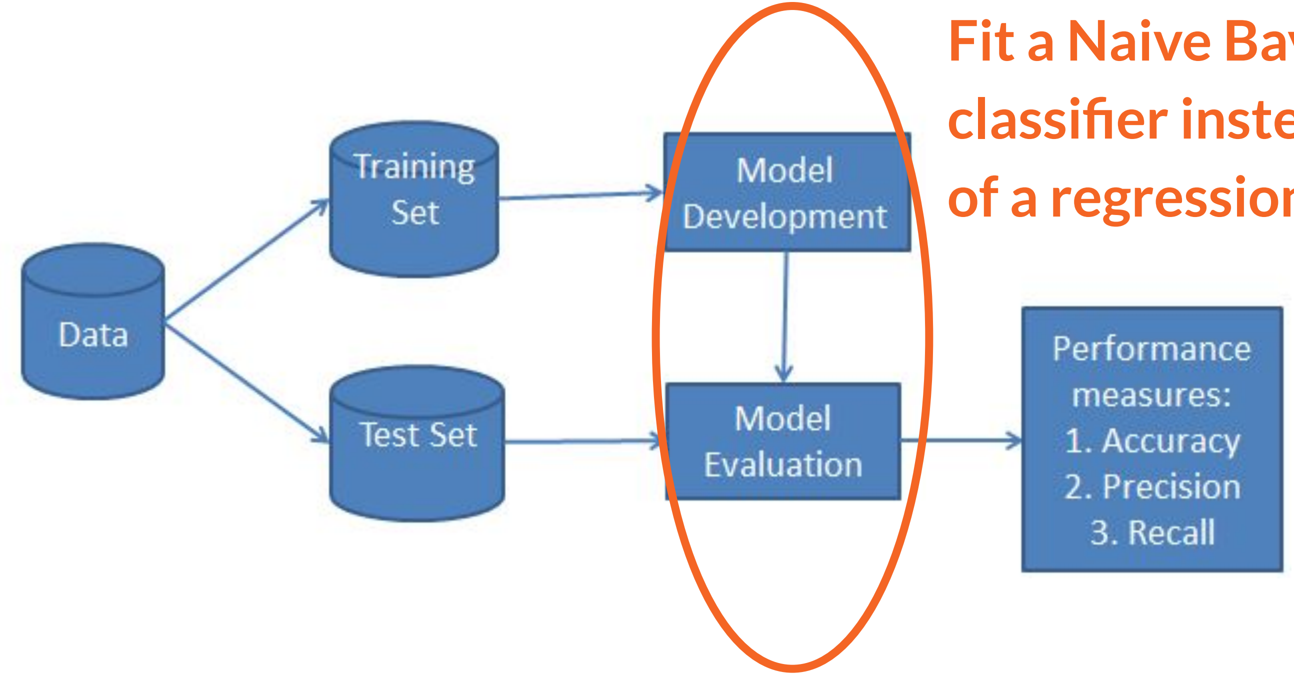
Naive Bayes... as a model?



https://www.datacamp.com/tutorial/naive-bayes-scikit-learn

Naive Bayes... as a model?

Fit a Naive Bayes classifier instead of a regression!



https://www.datacamp.com/tutorial/naive-bayes-scikit-learn

Naive Bayes using scikit learn

```
from sklearn.naive_bayes import GaussianNB

model = GaussianNB()

model.fit(X_train,Y_train)

model.predict(X_test)
```

https://www.datacamp.com/tutorial/naive-bayes-scikit-learn