# Lecture 24 – More Naive Bayes

DSC 40A, Spring 2023

# Announcements

- Midterm 2 review session is tonight from 7-9pm in **FAH 1301**.
  - That's the big room where Midterm 1 review was held.
  - No groupwork, no attendance.
  - Come to ask questions about the mock exam posted on the course website.
  - You should do the exam on your own beforehand.
- Homework 7 is due **tomorrow at 11:59pm**. This is the **last homework!**

# Midterm 2 is Monday during lecture

- You may use an unlimited number of handwritten note sheets for Midterm 2 (and Final Part 2). Start working on this now as you study!
- No calculators.
- Leave all answers **unsimplified** in terms of permutations, combinations, factorials, exponents, etc.
- Assigned seats will be posted on Campuswire.
- We will not answer questions during the exam. State your assumptions if anything is unclear.

# Midterm 2 is Monday during lecture

- The exam will definitely include short-answer questions such as multiple choice or filling in the numerical answer to a probability or combinatorics question. Short-answer questions will be graded on correctness only, so you don't need to show your work or provide explanation for these questions.

?

- The exam may also include long-answer homework-style questions, which would require explanation and be graded with partial credit.
- Midterm 2 covers all material that was not covered on Midterm 1. Clustering is in scope, but the vast majority will be probability and combinatorics. This week's lectures are also in scope.

# Agenda

- Naive Bayes with smoothing.
- Application — text classification.

## Naive Bayes with smoothing

# Recap: Naive Bayes classifier

healthy/unhealthy biological markers

- We want to predict a class, given certain features.

ripe/unripe softness, color, variety

- Using Bayes' theorem, we write

ripe

$$P(\text{class}|\text{features}) = \frac{P(\text{class}) \cdot \boxed{P(\text{features}|\text{class})}}{P(\text{features})}$$

ripe product

- For each class, we compute the numerator using the **naive assumption of conditional independence of features given the class**.

- We estimate each term in the numerator based on the training data.

- We predict the class with the largest numerator.
  - Works if we have multiple classes, too!

# Example: avocados

| color | softness | variety | ripeness |
|---|---|---|---|
| bright green | firm | Zutano | unripe |
| green-black | medium | Hass | ripe |
| purple-black | firm | Hass | ripe |
| green-black | medium | Hass | unripe |
| purple-black | soft | Hass | ripe |
| bright green | firm | Zutano | unripe |
| green-black | soft | Zutano | ripe |
| purple-black | soft | Hass | ripe |
| green-black | soft | Zutano | ripe |
| green-black | firm | Hass | unripe |
| purple-black | medium | Hass | ripe |

proportional to

features

You have a soft green-black Hass avocado. Based on this data, would you predict that your avocado is ripe or unripe?

$$P(\text{ripe} \mid \text{soft}, gb, \text{Hass}) \propto P(\text{ripe}) \cdot \boxed{p(\text{soft}, gb, \text{Hass} \mid \text{ripe})}$$
$$= P(\text{ripe}) \cdot P(\text{soft} \mid \text{ripe}) \cdot P(gb \mid \text{ripe}) \cdot P(\text{Hass} \mid \text{ripe})$$
$$= 7/11 \cdot 4/7 \cdot 3/7 \cdot 5/7$$

$$P(\text{unripe} \mid \text{soft}, gb, \text{Hass}) \propto P(\text{unripe}) \cdot p(\text{soft}, gb, \text{Hass} \mid \text{unripe})$$
$$= P(\text{unripe}) \cdot P(\text{soft} \mid \text{unripe}) \cdot P(gb \mid \text{unripe}) \cdot P(\text{Hass} \mid \text{unripe})$$
$$= 4/11 \cdot \boxed{0/4} \cdot 2/4 \cdot 2/4$$

makes whole prob. = 0

# Uh oh...

- There are no soft unripe avocados in the data set.
- The estimate $\boxed{P(\text{soft}|\text{unripe}) \approx \frac{\#\text{ soft unripe avocados}}{\#\text{ unripe avocados}}}$ is 0.

  ← 0

  ← 4
- The estimated numerator, $P(\text{unripe}) \cdot P(\text{soft, green-black, Hass}|\text{unripe}) = P(\text{unripe}) \cdot P(\text{soft}|\text{unripe}) \cdot P(\text{green-black}|\text{unripe}) \cdot P(\text{Hass}|\text{unripe})$, is also 0.
- But just because there isn't a soft unripe avocado in the data set, doesn't mean that it's impossible for one to exist!
- **Idea:** Adjust the numerators and denominators of our estimate so that they're never 0.

  smoothing

# Smoothing


- **Without** smoothing:

add to 1

$$P(\text{soft}|\text{unripe}) \approx \frac{\#\text{ soft unripe}}{\#\text{ soft unripe} + \#\text{ medium unripe} + \#\text{ firm unripe}}$$

$$P(\text{medium}|\text{unripe}) \approx \frac{\#\text{ medium unripe}}{\#\text{ soft unripe} + \#\text{ medium unripe} + \#\text{ firm unripe}}$$

$$P(\text{firm}|\text{unripe}) \approx \frac{\#\text{ firm unripe}}{\#\text{ soft unripe} + \#\text{ medium unripe} + \#\text{ firm unripe}}$$

- **With** smoothing:

still add to 1

add 1 to top, 3 to bottom

$$P(\text{soft}|\text{unripe}) \approx \frac{\#\text{ soft unripe} + 1}{\#\text{ soft unripe} + 1 + \#\text{ medium unripe} + 1 + \#\text{ firm unripe} + 1}$$

$$P(\text{medium}|\text{unripe}) \approx \frac{\#\text{ medium unripe} + 1}{\#\text{ soft unripe} + 1 + \#\text{ medium unripe} + 1 + \#\text{ firm unripe} + 1}$$

$$P(\text{firm}|\text{unripe}) \approx \frac{\#\text{ firm unripe} + 1}{\#\text{ soft unripe} + 1 + \#\text{ medium unripe} + 1 + \#\text{ firm unripe} + 1}$$

because 3 levels of firmness

- When smoothing, we add 1 to the count of every group whenever we're estimating a conditional probability.

# Example: avocados, with smoothing

| color | softness | variety | ripeness |
|---|---|---|---|
| bright green | firm | Zutano | unripe |
| green-black | medium | Hass | ripe |
| purple-black | firm | Hass | ripe |
| green-black | medium | Hass | unripe |
| purple-black | soft | Hass | ripe |
| bright green | firm | Zutano | unripe |
| green-black | soft | Zutano | ripe |
| purple-black | soft | Hass | ripe |
| green-black | soft | Zutano | ripe |
| green-black | firm | Hass | unripe |
| purple-black | medium | Hass | ripe |

You have a soft green-black Hass avocado. Using Naive Bayes, **with smoothing**, would you predict that your avocado is ripe or unripe?

without smoothing: $\frac{4}{7}$

with smoothing: $\frac{5}{10}$

P(Hass|ripe) without smoothing: $\frac{\#\text{ripe Hass}}{\#\text{ripe Hass} + \#\text{ripe Zutano}}$

$$P(\text{ripe}|\text{soft},gb,\text{Hass}) \propto P(\text{ripe}) \cdot P(\text{soft},gb,\text{Hass}|\text{ripe})$$
$$= P(\text{ripe}) \cdot \boxed{P(\text{soft}|\text{ripe})} \cdot P(gb|\text{ripe}) \cdot \boxed{P(\text{Hass}|\text{ripe})}$$
$$= 7/11 \cdot 5/10 \cdot 4/10 \cdot 6/9$$

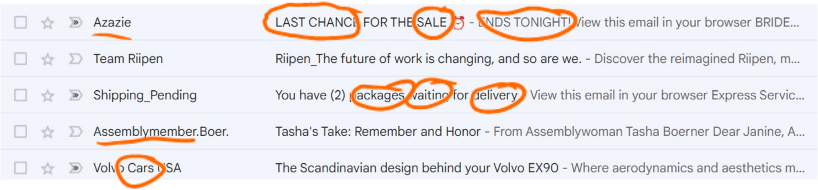
$$P(\text{unripe}|\text{soft},gb,\text{Hass}) \propto P(\text{unripe}) \cdot P(\text{soft},gb,\text{Hass}|\text{unripe})$$
$$= P(\text{unripe}) \cdot P(\text{soft}|\text{unripe}) \cdot P(gb|\text{unripe}) \cdot P(\text{Hass}|\text{unripe})$$
$$= 4/11 \cdot 1/7 \cdot 3/7 \cdot 3/6$$

# Text classification

# Text classification

- Text classification problems include:
  - Sentiment analysis (e.g. positive and negative customer reviews).
  - Determining genre (news articles, blog posts, etc.).
  - Spam filtering.

# Spam filtering




- **Our goal:** given the body of an email, determine whether it's **spam** or **ham** (not spam).

- **Question:** How do we come up with features?

words

# Features

**Idea:**

- Choose a **dictionary** of $d$ words.
- Represent each email with a **feature vector** $\vec{x}$:

$$\vec{x} = \begin{bmatrix} x^{(1)} \\ x^{(2)} \\ \dots \\ x^{(d)} \end{bmatrix}$$

→ prince

  where
  - $x^{(i)} = 1$ if word $i$ is present in the email, and
  - $x^{(i)} = 0$ otherwise.

This is called the **bag-of-words** model. This model ignores the frequency and meaning of words.

## Concrete example

- Dictionary: “prince”, “money”, “free”, and “just”.
- Dataset of 5 emails (red are spam, green are ham):
  - “I am the prince of UCSD and I demand money.”
  - “Tapioca Express: redeem your free Thai Iced Tea!”
  - “DSC 10: free points if you fill out CAPEs!”
  - “Click here to make a tax-free donation to the IRS.”
  - “Free career night at Prince Street Community Center.”

training data

$$
\begin{array}{l|ccccc}
 & \text{email 1} & \text{email 2} & \text{email 3} & \text{email 4} & \text{email 5} \\
\hline
\text{prince} & 1 & 0 & 0 & 0 & 1 \\
\text{money} & 1 & 0 & 0 & 0 & 0 \\
\text{free} & 0 & 1 & 1 & 1 & 1 \\
\text{just} & 0 & 0 & 0 & 0 & 0
\end{array}
$$

# Naive Bayes for spam classification

$$P(\text{class} \mid \text{features}) = \frac{P(\text{class}) \cdot P(\text{features} \mid \text{class})}{P(\text{features})}$$

spam

ham

- To classify an email, we'll use Bayes' theorem to calculate the probability of it belonging to each class:
  - $P(\text{spam} \mid \text{features})$.
  - $P(\text{ham} \mid \text{features})$.
- We'll predict the class with a larger probability.

# Naive Bayes for spam classification

$$P(\text{class} \mid \text{features}) = \frac{P(\text{class}) \cdot P(\text{features} \mid \text{class})}{P(\text{features})}$$

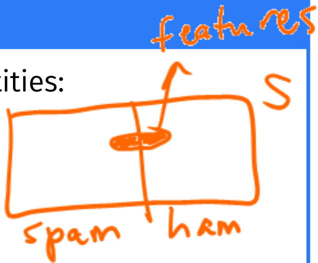
- Note that the formulas for $P(\text{spam} \mid \text{features})$ and $P(\text{ham} \mid \text{features})$ have the same denominator, $P(\text{features})$.
- Thus, we can find the larger probability just by comparing numerators:
  - $P(\text{spam}) \cdot P(\text{features} \mid \text{spam})$.
  - $P(\text{ham}) \cdot P(\text{features} \mid \text{ham})$.

# Naive Bayes for spam classification

**Discussion Question**

We need to determine four quantities:

1. $P$(features | spam). ?
2. $P$(features | ham).
3. $P$(spam). $> = 1$
4. $P$(ham).




Which of these probabilities should add to 1?

a) 1, 2
b) 3, 4
c) Both (a) and (b).
d) Neither (a) nor (b).

# Estimating probabilities with training data

- To estimate $P(\text{spam})$, we compute

$$P(\text{spam}) \approx \frac{\text{\# spam emails in training set}}{\text{\# emails in training set}}$$

- To estimate $P(\text{ham})$, we compute

$$P(\text{ham}) \approx \frac{\text{\# ham emails in training set}}{\text{\# emails in training set}}$$

- What about $P(\text{features} \mid \text{spam})$ and $P(\text{features} \mid \text{ham})$?

# Assumption of conditional independence

- Note that $\boxed{P(\text{features} \mid \text{spam})}$ looks like

money is included

$$P(\underbrace{x^{(1)}}_{\text{prince not included}} = 0, x^{(2)} = 1, \ldots, x^{(d)} = 0 \mid \text{spam})$$

- Recall: the key assumption that the Naive Bayes classifier makes is that **the features are conditionally independent given the class**.

- This means we can estimate $P(\text{features} \mid \text{spam})$ as

$$P(x^{(1)} = 0, x^{(2)} = 1, \ldots, x^{(d)} = 0 \mid \text{spam})$$
$$= \underbrace{P(x^{(1)} = 0 \mid \text{spam})}_{\frac{\#\text{ spam emails without prince}}{\#\text{ spam emails}}} \cdot \underbrace{P(x^{(2)} = 1 \mid \text{spam})} \cdot \ldots \cdot \underbrace{P(x^{(d)} = 0 \mid \text{spam})}$$

# Concrete example

- Dictionary: "prince", "money", "free", and "just".
- Dataset of 5 emails (red are spam, green are ham):
  - "I am the prince of UCSD and I demand money."
  - "Tapioca Express: redeem your free Thai Iced Tea!"
  - "DSC 10: free points if you fill out CAPEs!"
  - "Click here to make a tax-free donation to the IRS."
  - "Free career night at Prince Street Community Center."

$$
\begin{array}{l}
\text{prince} \\ \text{money} \\ \text{free} \\ \text{just}
\end{array}
\quad
\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}
\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}
\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}
\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}
\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}
$$

email 1 email 2 email 3 email 4 email 5

# Concrete example

- New email to classify: “Download a free copy of the Prince of Persia.”‘

feature vector

$$
\begin{array}{l|ccccc}
 & \text{email 1} & \text{email 2} & \text{email 3} & \text{email 4} & \text{email 5} \\
\hline
\text{prince} & 1 & 0 & 0 & 0 & 1 \\
\text{money} & 1 & 0 & 0 & 0 & 0 \\
\text{free} & 0 & 1 & 1 & 1 & 1 \\
\text{just} & 0 & 0 & 0 & 0 & 0
\end{array}
\qquad
\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}
$$

$$P\left(spam \,\middle|\, \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}\right) \propto P(spam) \cdot P\left(\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} \,\middle|\, spam\right)$$

$$= \underline{P(spam)} \cdot \underline{P(x^{(1)}=1 \mid spam)} \cdot \underline{P(x^{(2)}=0 \mid spam)} \cdot \underline{P(x^{(3)}=1 \mid spam)} \cdot P(x^{(4)}=0 \mid spam)$$

$$= 2/5 \cdot 1/2 \cdot 1/2 \cdot 1/2 \cdot 2/2 = \frac{1}{20}$$

$$P\left(ham \,\middle|\, \begin{bmatrix}1\\0\\1\\0\end{bmatrix}\right) \propto P(ham)\cdot P\left(\begin{bmatrix}1\\0\\1\\0\end{bmatrix} \,\middle|\, ham\right)$$

$$= P(ham)\cdot P(x^{(1)}=1 \mid ham)\cdot P(x^{(2)}=0 \mid ham)$$

$$\cdot P(x^{(3)}=1 \mid ham)\cdot P(x^{(4)}=0 \mid ham)$$

$$= 3/5 \cdot 1/3 \cdot 3/3 \cdot 3/3 \cdot 3/3 = \frac{1}{5}$$

| | email 1 | email 2 | email 3 | email 4 | email 5 |
|---|---|---|---|---|---|
| prince | 1 | 0 | 0 | 0 | 1 |
| money | 1 | 0 | 0 | 0 | 0 |
| free | 0 | 1 | 1 | 1 | 1 |
| just | 0 | 0 | 0 | 0 | 0 |

ham > spam

$$\boxed{\frac{1}{5} > \frac{1}{20}}$$

email will be classified as ham

# Uh oh...

- What happens if we try to classify the email "just what's your price, prince"?

$$
\begin{array}{c|ccccc}
 & \text{email 1} & \text{email 2} & \text{email 3} & \text{email 4} & \text{email 5} \\
\hline
\text{prince} & 1 & 0 & 0 & 0 & 1 \\
\text{money} & 1 & 0 & 0 & 0 & 0 \\
\text{free} & 0 & 1 & 1 & 1 & 1 \\
\text{just} & 0 & 0 & 0 & 0 & 0
\end{array}
\qquad \Rightarrow \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}
$$

$$
P(\text{spam}) \cdot P(x^{(1)}=1 \mid \text{spam}) \cdot P(x^{(2)}=0 \mid \text{spam}) \cdot P(x^{(3)}=0 \mid \text{spam}) \cdot P(x^{(4)}=1 \mid \text{spam}) = 0
$$

same for ham, will have term $P(x^{(4)}=1 \mid \text{ham})=0$

# Smoothing

- **Without** smoothing:

$$P(x^{(i)} = 1 \mid \text{spam}) \approx \frac{\text{\# spam containing word } i}{\text{\# spam containing word } i + \text{\# spam not containing word } i}$$

- **With** smoothing:

$$P(x^{(i)} = 1 \mid \text{spam}) \approx \frac{(\text{\# spam containing word } i) + 1}{(\text{\# spam containing word } i) + 1 + (\text{\# spam not containing word } i) + 1}$$

- When smoothing, we add 1 to the count of every group whenever we're estimating a conditional probability.

# Concrete example with smoothing

- What happens if we try to classify the email “just what’s your price, prince”?

# Modifications and extensions

- **Idea:** Use pairs (or longer sequences) of words rather than individual words as features.
  - This better captures the dependencies between words.
  - It also leads to a much larger space of features, increasing the complexity of the algorithm.

# Modifications and extensions

- **Idea:** Use pairs (or longer sequences) of words rather than individual words as features.
  - This better captures the dependencies between words.
  - It also leads to a much larger space of features, increasing the complexity of the algorithm.
- **Idea:** Instead of recording whether each word appears, record how many times each word appears.
  - This better captures the importance of repeated words.

# Summary

# Summary, next time

- Smoothing gives a way to make better predictions when a feature has never been encountered in the training data.
- The Naive Bayes classifier can be used for text classification, using the bag-of-words model.
- **Next time:** measuring performance of classifiers using precision and recall.