Lecture 25 – Precision and Recall

DSC 40A, Winter 2024

Announcements

- **Midterm 2 is Wednesday 3/13** during lecture.

- I'm travelling from next Tuesday to Saturday, Prof. Gal Mishne will proctor the midterm on 3/13.

- ~~Next~~ This week we have two review sessions, one is Monday discussion (for Midterm 2), one is Friday lecture (for Final)
  - Zhenduo (TA) and tutors will lead both sessions.

  - I've asked TA/tutor to move their OH to Monday/Tuesday.

Announcements

- Final is on March 22, Final part I/Part II is replacable with midterm 1/midterm 2, respectively.
  - See more announcements about final on Course Website/Campuswire.

  - Final will be more multiple choice/fill in the blank style

- Please fill out Student Evaluation of Teaching:
  - https://academicaffairs.ucsd.edu/Modules/Evals?e11210304

  - If at least 80% of the enrolled students fill out this survey, everyone in this class will get 0.5% extra credit on their final grade.

About Midterm 2

- You'll be allowed an unlimited number of handwritten note sheets for Midterm 2. Start studying and preparing your notes now!
  - Has to be handwritten, no printed notes.

- Midterm 2 covers lecture 13-24. Clustering is included, but the vast majority will be **probability and combinatorics**.

- No calculators.
  - There will be some numerical calculations, but no very hard ones.

- Assigned seats ~~will be~~ has been posted on Campuswire.

- We will not answer questions during the exam. State your assumptions if anything is unclear.

Midterm 2 Preparation Strategy

- One useful strategy is attributing complicated real-world problems into known models.
  - Example: rolling a die

- Unlike Part I of this course which is mostly proof, in Part II we have done lots of examples in lecture, make sure you understand them. If not, please ask questions in OH/Campuswire.
  - You will see something similar in the exam.

- Everything I covered in the lecture 13-24 is possible to appear in the midterm.

Emphasize on Probability & Combinatorics

Agenda

- Recap: Text classification with Naive Bayes
- Measuring quality of classification

Text classification

Recap: Naive Bayes for spam classification

- To classify an email, we'll use Bayes' theorem to calculate the probability of it belonging to each class:

$$P(\mathrm{spam} \mid \mathrm{features}) = \frac{P(\mathrm{spam}) \cdot \boxed{P(\mathrm{features} \mid \mathrm{spam})}}{P(\mathrm{features})}$$

$$P(\mathrm{ham} \mid \mathrm{features}) = \frac{P(\mathrm{ham}) \cdot \boxed{P(\mathrm{features} \mid \mathrm{ham})}}{P(\mathrm{features})}$$

conditional

- We'll find the larger probability by comparing numerators, and predict that class.

Independence

- To compute the numerator, we make the naive assumption that the features are conditionally independent given the class.

Concrete example

- Dictionary: “prince”, “money”, “free”, and “just”.

- Dataset of 5 emails (red are spam, green are ham):

  Training Dataset

  - “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.”

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

Email1 Email2 Email3 Email4 Email5

Center.”

Concrete example

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

Email1 Email2 Email3 Email4 Email5

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

$$P(\text{Spam} \mid \text{features})$$

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

$$= \frac{2}{5} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{0}{2} = 0$$

$$P(\text{Ham} \mid \text{features})$$

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

$$= \frac{3}{5} \cdot \frac{1}{3} \cdot \frac{3}{3} \cdot \frac{0}{3} \cdot \frac{0}{3} = 0$$

Smoothing

+1 to top
+2 to bottom

- **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”?

$$P(\text{Spam} \mid \text{features})$$

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

$$\frac{2}{5} \cdot \frac{2}{4} \cdot \frac{2}{4} \cdot \frac{2}{4} \cdot \frac{0+1}{2+2} =$$

$$P(\text{Ham} \mid \text{features})$$

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

$$= \frac{3}{5} \cdot \frac{1+1}{3+2} \cdot \frac{3+1}{3+2} \cdot \frac{0+1}{3+2} \cdot \frac{0+1}{3+2} = \cancel{0}\ \frac{1}{125}$$

Center.”

$$\begin{array}{r} \text{Prince} \\ \text{money} \\ \text{free} \\ \text{just} \end{array} \begin{bmatrix}1\\1\\0\\0\end{bmatrix} \begin{bmatrix}0\\0\\1\\0\end{bmatrix} \begin{bmatrix}0\\0\\1\\0\end{bmatrix} \begin{bmatrix}0\\0\\1\\0\end{bmatrix} \begin{bmatrix}1\\0\\1\\0\end{bmatrix}$$

Email 1 Email 2 Email 3 Email 4 Email 5

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

# of spam containing “just”

# of spam not containing “just”

Modifications and extensions

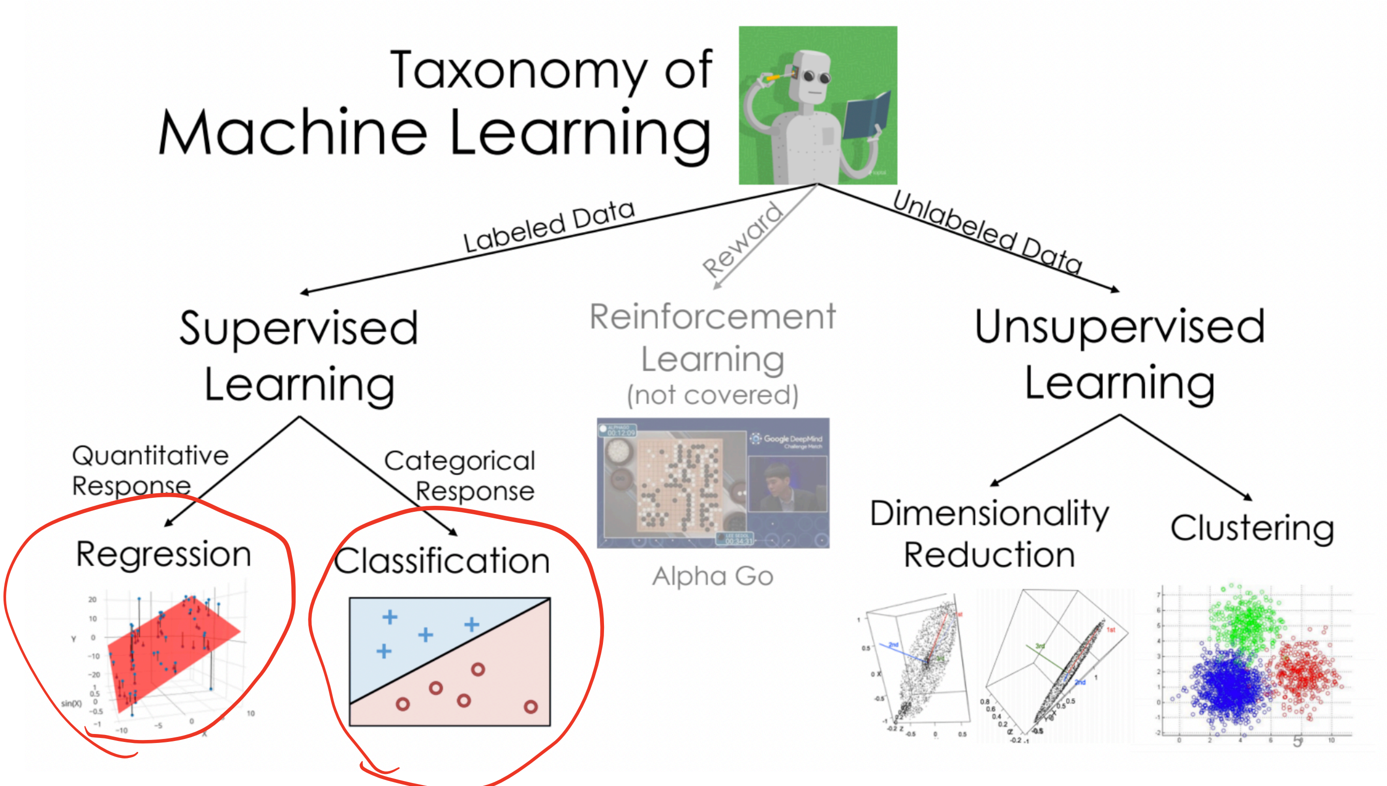
- **Idea:** Use pairs (or longer sequences) of words rather than individual words as features.
  - This better captures the dependencies between words.
  - It also leads to a much larger space of features, increasing the complexity of the algorithm.

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.

Measuring quality of classification



1

---

[1]taken from Joseph Gonzalez at UC Berkeley

Classification problems

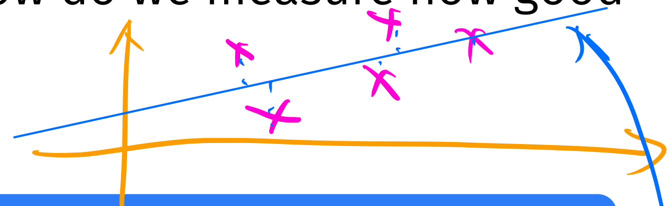
- In the classification problem, we make predictions based on data (called **training data**) for which we know the value of the **categorical** response variable.

- Example classification problems:
  - Deciding whether a patient has kidney disease.
  - Identifying handwritten digits.
  - Determining whether an avocado is ripe.
  - Predicting whether credit card activity is fraudulent.

Assessing the quality of a classifier

- Naive Bayes is one classification algorithm, or **classifier**, but there are many others.

- Is Naive Bayes any good? How do we measure how good of a job a classifier does?



MSE

Discussion Question

Think back to regression (supervised learning with a quantitative response variable). How did we measure the quality of our predictions? Can we adopt a similar strategy?

base on distance close vs. far

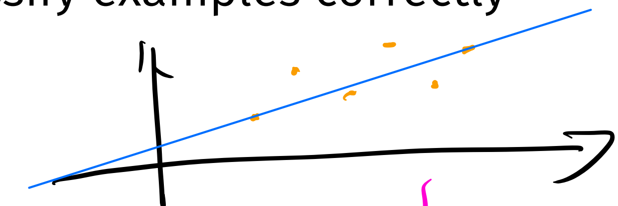
classification: right vs. wrong

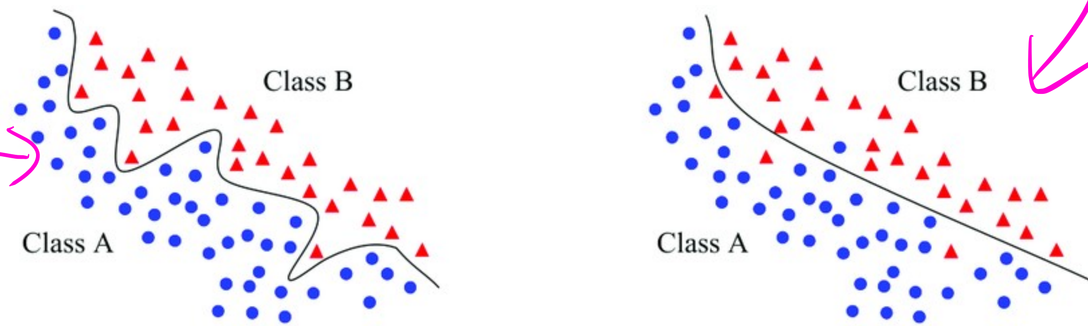
Unseen data

- A natural way to measure the quality of our classifications is to see how often we predict the right category.

- We want to make good predictions on **unseen data**. So we'll measure how often we classify examples correctly for a new set of **test data**.

- This avoids **overfitting**.





2

[2] Park, Hochong Son, Joo-Hiuk. (2021). Machine Learning Techniques for THz Imaging and Time-Domain Spectroscopy. Sensors. 21. 1186. 10.3390/s21041186.

Accuracy

- Classification **accuracy** is the proportion of examples in the test set that are correctly classified.

- Accuracy is measured on a 0 to 1 scale.

Accuracy

- We can think of accuracy as an estimate for the probability of making a correct classification on an unseen example.
- Parameter:

$$P(\text{successful classification})$$

- Estimate:

$$\text{accuracy} = \frac{\text{\# correctly classified examples in test set}}{\text{size of test set}}$$

Imbalanced classes

Alagille syndrome is a rare genetic condition that affects 1 in 40,000 people. We want to classify people as having this condition (**unhealthy**) or not having this condition (**healthy**).

Discussion Question

Consider a classifier that classifies everyone as **healthy**.

1. What is the accuracy of this classifier?

   $\frac{39{,}999}{40{,}000} \to 99\%$

2. What are the ethical repercussions of using this classifier?

High accuracy is not enough

- We want to avoid overdiagnosis (telling someone they have the condition when they don't).
- We also want to avoid underdiagnosis (telling someone they're healthy when they're not).
- It's easy to avoid either one of these. It's hard to avoid both of these simultaneously, yet a good classifier should do exactly that.

Different types of errors

good: want to maximize

	Actually **unhealthy**	Actually **healthy**
Classified as **unhealthy**	True positive	False positive
Classified as **healthy**	False negative	True negative

bad: want to minimize

True positive means a person is __tru__ly classified as __positive__

False positive — falsely — positive

False negative — falsely — negative

True negative — truly — negative

Avoid overdiagnosis

	Actually **unhealthy**	Actually **healthy**
Classified as **unhealthy**	True positive	False positive
Classified as **healthy**	False negative	True negative

- How often does our prediction of the condition mean a person actually has the condition?

want to maximize ✓

- Parameter:

A B

P(actually **unhealthy**|classified as **unhealthy**)

- Estimate:

A ∩ B or 'A and B'

$$\textbf{precision} = \frac{\text{\# people in test set \textbf{correctly} classified as \textbf{unhealthy}}}{\text{\# people in test set classified as \textbf{unhealthy}}}$$

B

Avoid underdiagnosis

	Actually **unhealthy**	Actually **healthy**
Classified as **unhealthy**	True positive	False positive
Classified as **healthy**	False negative	True negative

- How often do we identify those that actually have the condition?

- Parameter:

want to maximize

$$P(\overset{A}{\text{classified as } \textbf{unhealthy}} \mid \overset{B}{\text{actually } \textbf{unhealthy}})$$

- Estimate:

$$\textbf{recall} = \frac{\overset{A \text{ and } B}{\#\text{ people in test set } \textbf{correctly} \text{ classified as } \textbf{unhealthy}}}{\underset{B}{\#\ \textbf{unhealthy} \text{ people in test set}}}$$

Precision vs. recall

	Actually **unhealthy**	Actually **healthy**
Classified as **unhealthy**	True positive	False positive
Classified as **healthy**	False negative	True negative

- Precision:

$$\textbf{precision} = \frac{\text{\# people in test set \textbf{correctly} classified as \textbf{unhealthy}}}{\text{\# people in test set classified as \textbf{unhealthy}}}$$

$$= \frac{\text{true positives}}{\text{true positives + false positives}}$$

- Recall:

$$\textbf{recall} = \frac{\text{\# people in test set \textbf{correctly} classified as \textbf{unhealthy}}}{\text{\# \textbf{unhealthy} people in test set}}$$

$$= \frac{\text{true positives}}{\text{true positives + false negatives}}$$

goal: maximize both precision & recall

Precision vs. recall

	Actually **unhealthy**	Actually **healthy**
Classified as **unhealthy**	True positive 0	False positive 0
Classified as **healthy**	False negative small	True negative large

Discussion Question

Consider a classifier that classifies everyone as **healthy**.

1. What is the precision of this classifier?

   undefined, but good

2. What is the recall of this classifier?

recall = 0 bad.

Precision vs. recall

	Actually **unhealthy**	Actually **healthy**
Classified as **unhealthy**	True positive few	False positive many
Classified as **healthy**	False negative 0	True negative 0

Discussion Question

Now consider a classifier that classifies everyone as **unhealthy**.

1. What is the precision of this classifier?

$\frac{few}{few+many}$ close to 0 (bad)

2. What is the recall of this classifier?

$\frac{few}{few+0} \Rightarrow 1$ good.

Combining precision and recall

- We want high precision and high recall, but it's hard to have both.
- Let's combine them into a single measurement.
- Does the average of precision and recall work well?

$$\frac{P + R}{2}$$

- Compare:
  - Classifier A ($P = 0$, $R = 1$) $\rightarrow$ 0.5
  - Classifier B ($P = 0.5$, $R = 0.6$) $\rightarrow$ 0.55

Combining precision and recall

- **Key insight:** Two moderate values are better than two extremes. Use the product, which shrinks when either term in the product is small.
- New way of combining precision and recall: **F-score**

$$\frac{2PR}{P+R}$$

- Compare:
  - Classifier A ($P = 0$, $R = 1$) $\longrightarrow \frac{2PR}{P+R} = 0$
  - Classifier B ($P = 0.5$, $R = 0.6$) $\longrightarrow \frac{2PR}{P+R} = \frac{6}{11}$

F-score

- The **F-score** combines the precision and recall of a classifier in a single measurement.

$P=1$
$R=1$

$$\frac{2PR}{P+R} = \frac{2\cdot1\cdot1}{1+1} = 1$$

- Higher F-score $\Rightarrow$ better classifier.

Discussion Question

What would be the F-score of a "perfect classifier"?

Summary

Summary

- Accuracy is a simple way of measuring the quality of a classifier, but it can be misleading when classes are imbalanced.
- Precision and recall are two other ways of measuring the quality of a classifier, but they can be hard to achieve simultaneously.
- The F-score combines precision and recall into a single measurement that assesses the quality of a classifier on a 0 to 1 scale.

$$\frac{2PR}{P+R}$$