Lecture 25 – Precision and Recall

DSC 40A, Spring 2023

Announcements

- Homework 7 was due last night, but can still be turned in if you have a slip day remaining. This is the **last homework!**
- Midterm 2 is Monday during lecture.
- Final Exam is **Saturday, June 10**. Given in two separate parts, both of which are optional.
  - Part 1 is 9-9:50am. Can replace Midterm 1.
  - Part 2 is 10-10:50am. Can replace Midterm 2.
- Next week is review only. No new content.

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

- 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:

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

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

breaks up into product

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

- 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):
  - “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.”

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

Concrete example

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

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

$$\begin{bmatrix}1\\0\\0\\1\end{bmatrix}$$

spam

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

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

ham

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

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

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

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

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

$$\begin{bmatrix}1\\0\\0\\1\end{bmatrix}$$

spam ← classify as spam

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

$$\frac{\cancel{2}}{5} \cdot \frac{\cancel{2}}{\cancel{4}} \cdot \frac{\cancel{2}}{\cancel{4}} \cdot \frac{\cancel{2}}{4} \cdot \frac{1}{4} \neq 0 \quad \frac{1}{80}$$

ham

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

$$\approx \frac{3}{5} \cdot \frac{2}{5} \cdot \frac{4}{5} \cdot \frac{1}{5} \cdot \frac{1}{5} \neq 0 \approx \frac{1}{125}$$

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.

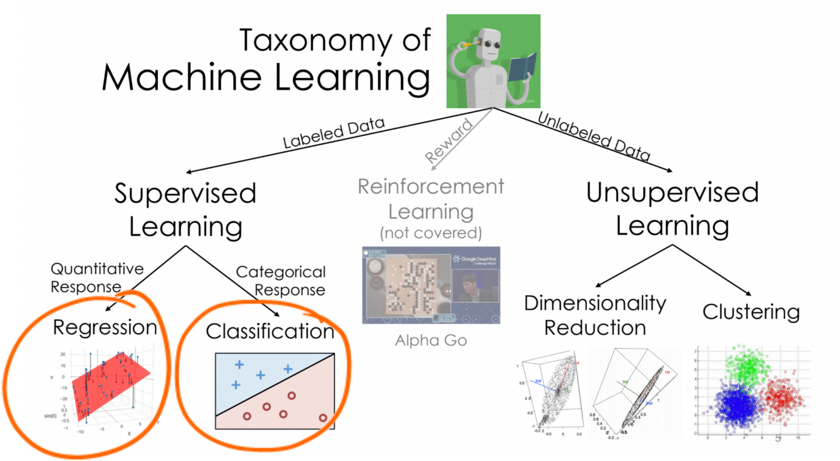
Modifications and extensions

- **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

Taxonomy of machine learning



1

[1]taken from Joseph Gonzalez @ 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?

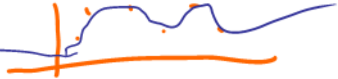
based on distance close vs. far,

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?

no close: just right/wrong

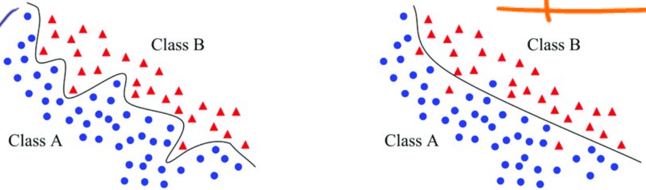


low MSE on training data

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**.

high MSE on new data



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

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

bad

	Actually unhealthy	Actually healthy
Classified as unhealthy		[illegible]
Classified as healthy	[illegible]	

good

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?
- Parameter:

$$P(\text{actually } \underbrace{\textbf{unhealthy}}_{\text{A}} | \underbrace{\text{classified as } \textbf{unhealthy}}_{\text{B}})$$

← want high

- Estimate:

A and B

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

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:

$$P(\text{classified as } \textbf{unhealthy} \mid \text{actually } \textbf{unhealthy}) \leftarrow \text{want it to be high}$$

- Estimate:

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

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: both high

Precision vs. recall

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

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

0

Discussion Question

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

1. What is the precision of this classifier? close to 0 (bad)

2. What is the recall of this classifier?

$$\text{recall} = \frac{\text{few}}{\text{few}+0} = 1 \text{ good}$$

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?

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

stupid

- Compare:
  - Classifier A ($P = 0$, $R = 1$) $\rightarrow \frac{P+R}{2} = \frac{0+1}{2} = 0.5$
  - Classifier B ($P = 0.5$, $R = 0.6$) $\rightarrow \frac{0.5+0.6}{2} \approx 0.55$

smarter

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**

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

(harmonic mean of P + R)

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

recognizes classifier B as better

F-score

← perfect classifier

measured on 0 to 1 scale, like accuracy

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

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

- Higher F-score ⇒ better classifier.

Discussion Question

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

precision = 1
recall = 1

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.

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