Text Classification

Sandeep Soni

09/25/2023

Class logistics

- Hw6 due

Hw6 review

What are the similarities and differences between KMeans clustering and topic modeling?

Hw6 review

	KMeans	LDA
Similarities	Word lists are likely to be very similar	
Differences	KMeans surfaces words that are more frequent	LDA surfaces words that are less frequent and more cohesive to a topic

Question for the day

"How to predict a label for given text?"

Agenda

- Classification problem
- Naive Bayes
- Training/test setup

Classification

- Input: document (e.g., email)
- Output: label (e.g., spam/ham)

Formal task

Formal task

- x is input (e.g., an email)
  - $x \in \mathcal{X}$ (e.g., set of emails)

Formal task

- x is input (e.g., an email)
  - $x \in \mathcal{X}$ (e.g., set of emails)
- y is output (e.g., spam)
  - $y \in \mathcal{Y}$ (e.g., {spam, ham})

Formal task

- x is input (e.g., an email)
  - $x \in \mathcal{X}$ (e.g., set of emails)
- y is output (e.g., spam)
  - $y \in \mathcal{Y}$ (e.g., {spam, ham})
- $y = h(x)$
  - h maps instances to labels

Classification

- The true mapping function h is not known to us so we want to find $\hat{h}$ that's a closest approximation

Rule based classification

- $\hat{h}$ ("I wish to discuss personal investment business matters with you so as to be able to learn of the available investment opportunities in your region or country.")

$\hat{h}$	if email contains phrase "investment opportunities" then spam

Supervised learning

- Learn $\hat{h}$ from training data given in the form of $<x, y>$ pairs

Classification Problems

Task	x	y
Language ID	text	{english, mandarin, hindi, …}
spam classification	email	{spam, ham}
party affiliation	speech	{republican, democrat}
sentiment analysis	text	{positive, negative, mixed, neutral}
music genre	lyric	{rock, pop, jazz, rap,…}

$\hat{h}(x)$

- We still need to resolve:
  - How to learn this mapping function? (e.g., which method to use, how to optimize, etc)
  - How to represent the input to this function?

Naive Bayes

- One simple yet quite effective classification method is Naive Bayes
  - Similar to LDA, it's a generative model
  - We'll represent input text as a bag of words vector

Refresher: Chain rule

- If x and y are random variables, the joint probability can be factorized using chain rule

$$P(x, y) = P(y)P(x \mid y)$$

$$P(x, y) = P(x)P(y \mid x)$$

Refresher: Chain rule

- If there are three variables, then chain rule gives:

$$P(x, y, z) = P(x)P(y \mid x)P(z \mid x, y)$$

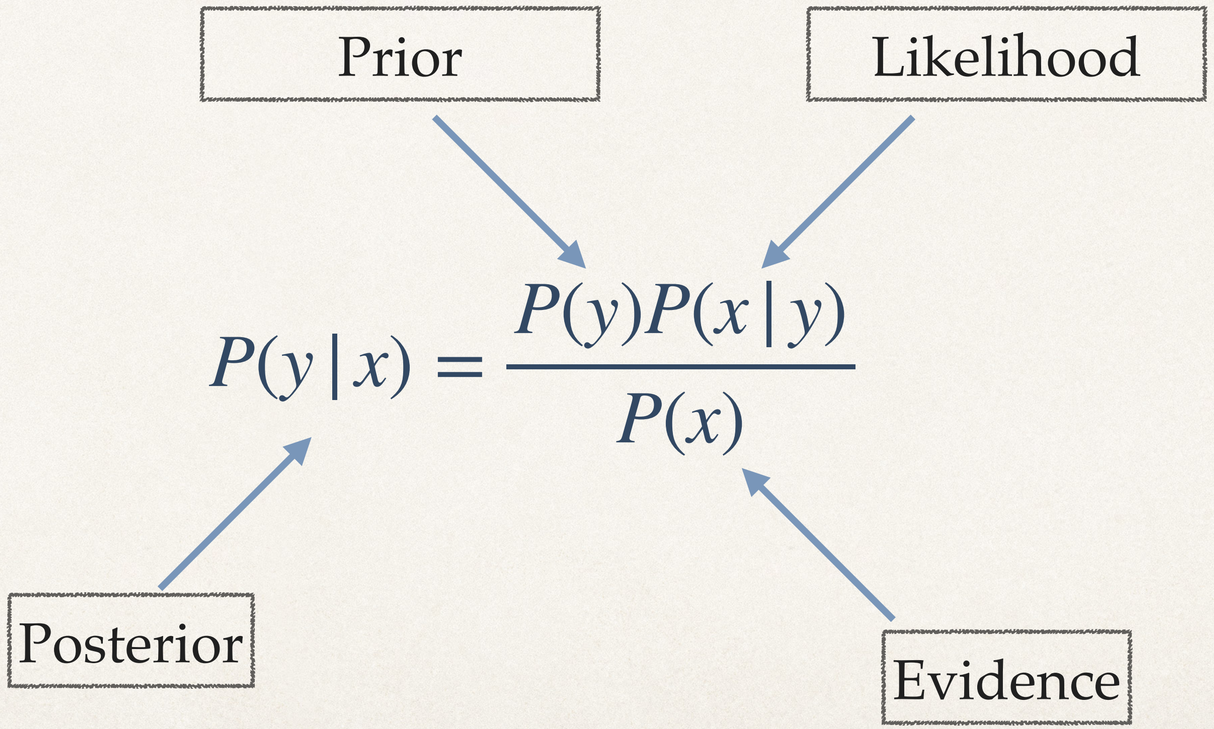
- Just like two variables, permutations are equivalent

Refresher: Chain rule

- In general, for multiple variables, chain rule is:

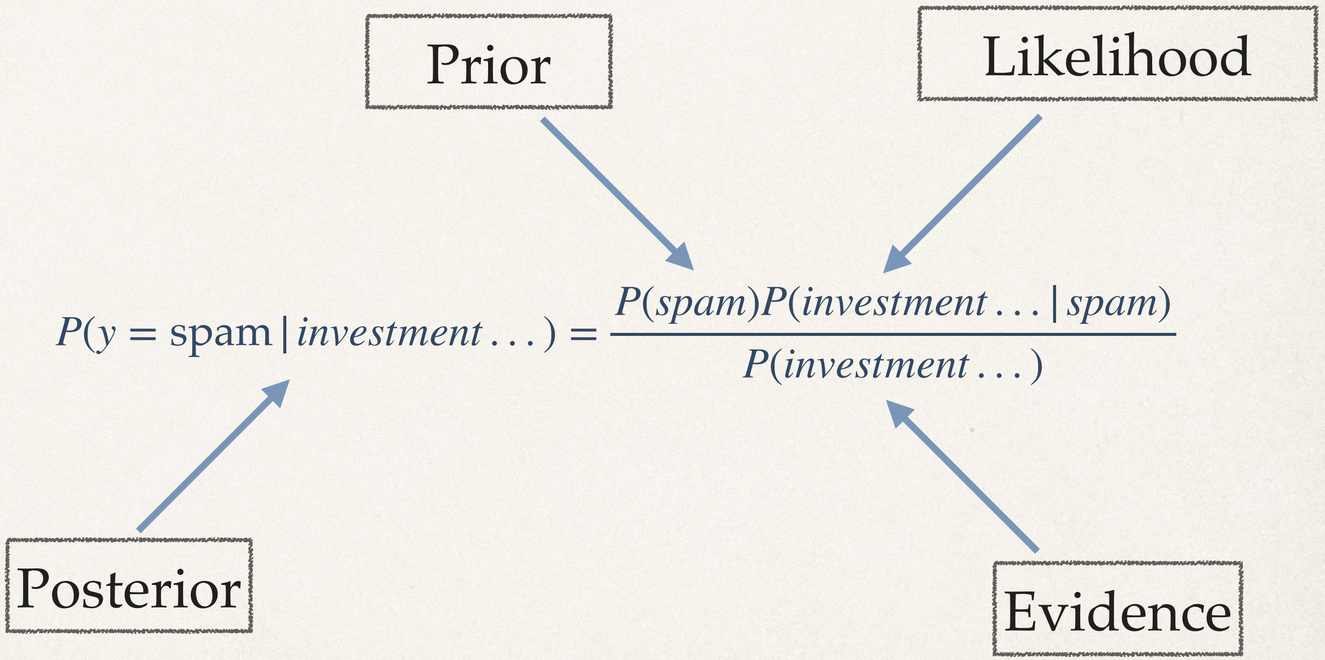
$$P(y, x_1, x_2, \ldots, x_n) = P(y)P(x_1 \mid y)P(x_2 \mid x_1, y)\ldots P(x_n \mid x_{n-1}, \cdots, y)$$

Refresher: Bayes Theorem



Refresher: Bayes Theorem

- For spam classification:
- Prior how probable is to see the spam label
- Likelihood how likely are these words in the email for spam
- Evidence how probable are these words in the email
- Posterior how probable is spam label on this email



Refresher: Bayes Theorem

- We can express the marginal probability in the denominator as:

$$P(y \mid x) = \frac{P(y)P(x \mid y)}{\sum_{y'} P(x, y')}$$

$$P(y \mid x) = \frac{P(y)P(x \mid y)}{\sum_{y'} P(y')P(x \mid y')}$$

Towards classification

- One way to learn the mapping $\hat{h}(x)$ is to learn the posterior distribution $P(y|x)$
- x is text, so not just one random variable but a bunch of variables, so we have to estimate $P(y|x_1, x_2, x_3, \ldots, x_n)$

Towards classification

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

Towards classification

- Joint probability in the numerator can be factorized using chain rule

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

Continue to apply chain rule to the numerator (denominator is ignored for now)

$$P(y \mid x_1, x_2, \ldots, x_n) \propto P(y)P(x_1, x_2, \ldots, x_n \mid y)$$
$$\propto P(y)P(x_1 \mid y)P(x_2, \ldots, x_n \mid y, x_1)$$
$$\ldots$$
$$\propto P(y)P(x_1 \mid y)P(x_2 \mid x_1, y)P(x_3 \mid x_2, x_1, y)\ldots P(x_n \mid x_{n-1}, \ldots, x_1, y)$$

This is intractable!

Continue to apply chain rule to the numerator (denominator is ignored for now)

$$P(y \mid x_1, x_2, \ldots, x_n) \propto P(y)P(x_1, x_2, \ldots, x_n \mid y)$$
$$\propto P(y)P(x_1 \mid y)P(x_2, \ldots, x_n \mid y, x_1)$$
$$\ldots$$
$$\propto P(y)P(x_1 \mid y)P(x_2 \mid x_1, y)P(x_3 \mid x_2, x_1, y)\ldots P(x_n \mid x_{n-1}, \ldots, x_1, y)$$

This is intractable!

Naive Bayes

- Naive conditional independence assumption

$$P(x_i \mid x_{i-1}, x_{i-2}, \ldots, y) = P(x_i \mid y)$$

- Given the category, the words (features) are independent of each other
- Under naive Bayes, P(“prince”|spam) = P(“prince”|spam, “kenyan”)

Now we can rewrite the posterior probability as:

$$P(y \mid x_1, x_2, \ldots, x_n) \propto P(y)P(x_1 \mid y)P(x_2 \mid y)P(x_3 \mid y)\ldots P(x_n \mid y)$$

$$\propto P(y)\prod_{i=1}^{n} P(x_i \mid y)$$

This is tractable!

We can estimate these probabilities by simply counting the instances in training data

$$P(y = spam) = \frac{\#\text{samples labeled spam}}{\#\text{ samples}}$$

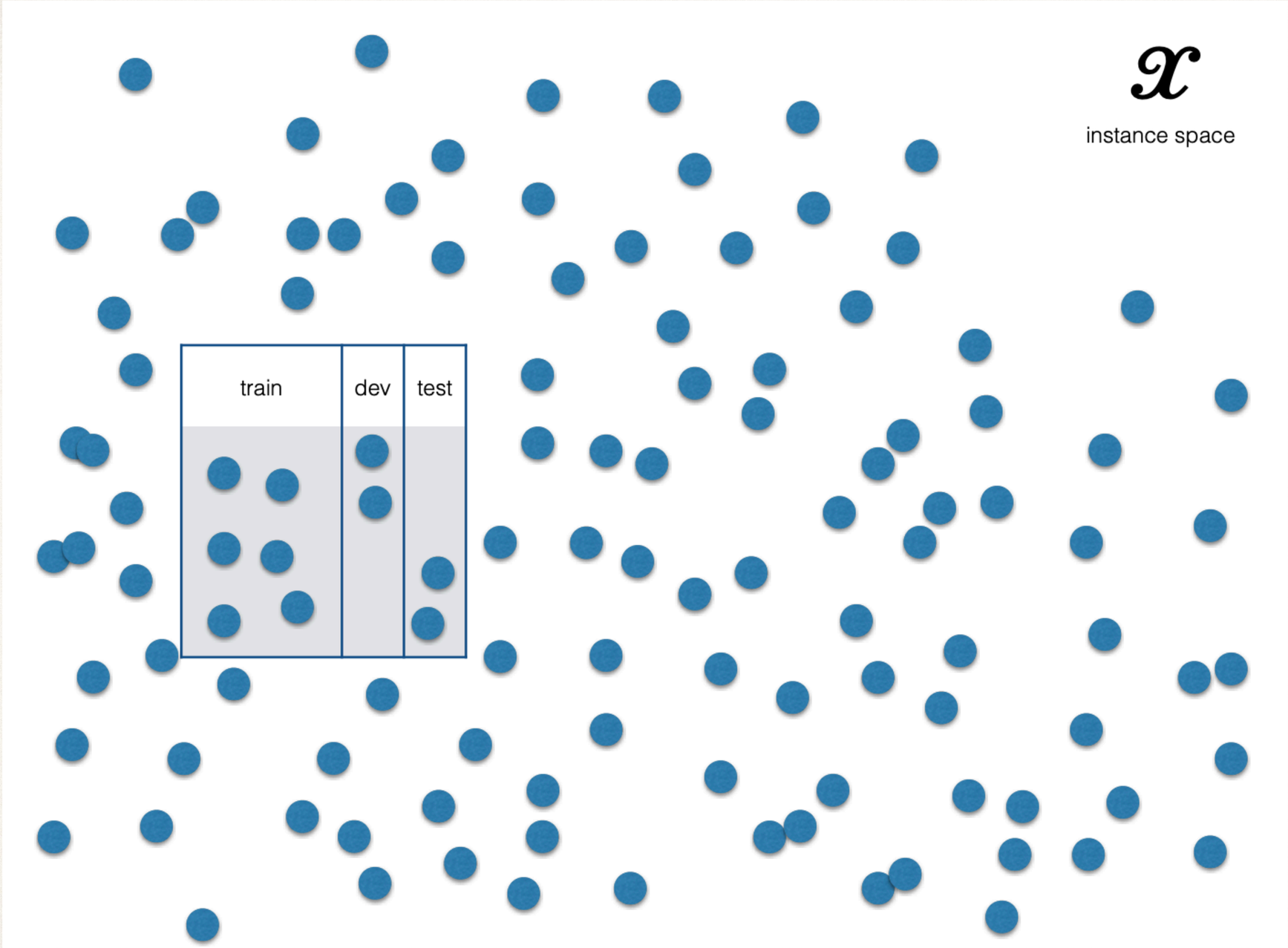
$$P(\text{"kenyan"}|\text{spam}) = \frac{\#\text{samples labeled spam and contain "kenyan"}}{\#\text{samples labeled spam}}$$

Picking the label

- Once you estimate the probabilities from training data, you can pick the label that maximizes the posterior

$$P(y = \text{spam} \mid \text{text}) > P(y = \text{ham} \mid \text{text})$$ Spam

otherwise Ham



Slide from David Bamman

Experiment design

- Training set is to estimate parameters of the model
- Development set is to perform model selection
- Test set for evaluation

Experiment design

- Typically, we use 80% data for training, 10% for model selection and 10% for evaluation
- One should be careful never to use to development or test data to do estimation

In class

- Naive Bayes Demo