NAÏVE BAYES

David Kauchak
CS159 Fall 2020

1

Admin

Assignment 6a

Assignment 6b

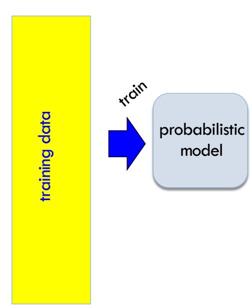
2

Machine Learning is…

Machine learning is about predicting the future based on the past.
-- Hal Daume III

3

Probabilistic Modeling



Model the data with a probabilistic model

specifically, learn *p(features, label)*

p(features, label) tells us how likely these features and this example are

4

An example: classifying fruit

Training data

examples	label
red, round, leaf, 3oz, …	apple
green, round, no leaf, 4oz, …	apple
yellow, curved, no leaf, 4oz, …	banana
green, curved, no leaf, 5oz, …	banana

train → probabilistic model: p(*features*, *label*)

5

Probabilistic models

Probabilistic models define a *probability distribution* over features and labels:

yellow, curved, no leaf, 6oz, banana → probabilistic model: p(*features*, *label*) → 0.004

6

Probabilistic model vs. classifier

Probabilistic model:

yellow, curved, no leaf, 6oz, banana → probabilistic model: p(*features*, *label*) → 0.004

Classifier:

yellow, curved, no leaf, 6oz → probabilistic model: p(*features*, *label*) → banana

7

Probabilistic models: classification

Probabilistic models define a *probability distribution* over features and labels:

yellow, curved, no leaf, 6oz, banana → probabilistic model: p(*features*, *label*) → 0.004

Given an unlabeled example: yellow, curved, no leaf, 6oz predict the label

How do we use a probabilistic model for classification/prediction?

8

Probabilistic models

Probabilistic models define a *probability distribution* over features and labels:

yellow, curved, no leaf, 6oz, banana → probabilistic model: p(*features*, *label*) → **0.004**

yellow, curved, no leaf, 6oz, apple → probabilistic model: p(*features*, *label*) → 0.00002

For each label, ask for the probability under the model
Pick the label with the highest probability

9

Probabilistic model vs. classifier

Probabilistic model:

yellow, curved, no leaf, 6oz, banana → probabilistic model: p(*features*, *label*) → 0.004

Classifier:

yellow, curved, no leaf, 6oz → probabilistic model: p(*features*, *label*) → banana

Why probabilistic models?

10

Probabilistic models

Probabilities are nice to work with
- range between 0 and 1
- can combine them in a well understood way
- lots of mathematical background/theory

Provide a strong, well-founded groundwork
- Allow us to make clear decisions about things like smoothing
- Tend to be much less "heuristic"
- Models have very clear meanings

11

Probabilistic models: big questions

1. Which model do we use, i.e. how do we calculate p(*feature, label*)?
2. How do train the model, i.e. how to we we estimate the probabilities for the model?
3. How do we deal with overfitting (i.e. smoothing)?

12

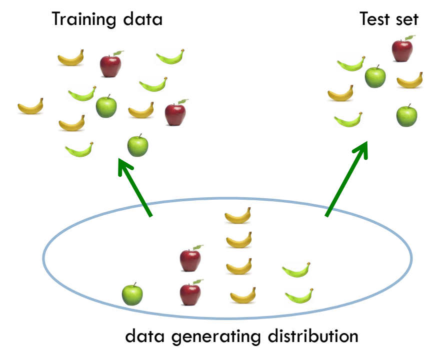
Basic steps for probabilistic modeling

Probabilistic models

Step 1: pick a model	Which model do we use, i.e. how do we calculate p(*feature, label*)?
Step 2: figure out how to estimate the probabilities for the model	How do train the model, i.e. how to we we estimate the probabilities for the model?
Step 3 (optional): deal with overfitting	How do we deal with overfitting?

13

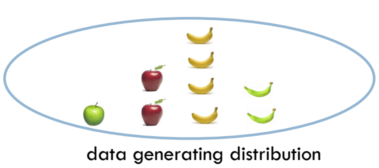
What was the data generating distribution?



14

Step 1: picking a model

What we're really trying to do is model the data generating distribution, that is how likely the feature/label combinations are



15

Some math

$$p(features, label) = p(x_1, x_2, \ldots, x_m, y)$$

$$= p(y)p(x_1, x_2, \ldots, x_m \mid y)$$

What rule?

16

Some math

$$
\begin{aligned}
p(features, label) &= p(x_1, x_2, \ldots, x_m, y) \\
&= p(y)p(x_1, x_2, \ldots, x_m \mid y) \\
&= p(y)p(x_1 \mid y)p(x_2, \ldots, x_m \mid y, x_1) \\
&= p(y)p(x_1 \mid y)p(x_2 \mid y, x_1)p(x_3, \ldots, x_m \mid y, x_1, x_2) \\
&= p(y)\prod_{j=1}^{m} p(x_i \mid y, x_1, \ldots, x_{i-1})
\end{aligned}
$$

17

Step 1: pick a model

$$p(features, label) = p(y)\prod_{j=1}^{m} p(x_i \mid y, x_1, \ldots, x_{i-1})$$

So, far we have made NO assumptions about the data

$$p(x_m \mid y, x_1, x_2, \ldots, x_{m-1})$$

How many entries would the probability distribution table have if we tried to represent all possible values and we had 7000 binary features?

18

Full distribution tables

x_1	x_2	x_3	…	y	p( )
0	0	0	…	0	*
0	0	0	…	1	*
1	0	0	…	0	*
1	0	0	…	1	*
0	1	0	…	0	*
0	1	0	…	1	*
			…		

All possible combination of features!

Table size: 2^{7000} = ?

19

2^{7000}

1621696755662202026466665085478377095191112430363743256235982084151527023162702352987080237879
4460004651996019099530984538652557892546513204107022110253564658647431585227076599373340842842
7224200122818782600729310826170431944842663920777841250999968601694360066600112098175792966787
8196255237700655294757256678055809293844627218640216108862600816097132874749204352087401101862
6908423275017246052311293955235059054544214554772509509096507889478094683592939574112569473438
6191215296848474344406741204174020887540371869421701550220735398381224299258743537536161041593
4359455766656170179090417259702533652666268202180849389281269970952857089069637557541434487608
8248369941993802415197514510125127043829087280919538476302857811854024099958895964192277601255
3604911562403499947144160905730842429313962119953679373012944795600248333570738998392029910322
3465980389530690429801740098017325210691307971242016963397230218353007589784519525848553710885
8195631737000743805167411189134617501484521767984296782842287373127422122022517597535994839257
0298779077063553347902449354353866605125910795672914312162977887848185522928196541766009803989
9799168140474938421574351580260381151068286406789730483829220346042775765507377656754750702714
4662263487685709621261074762705203049488907208978593689047063428548531668665657327174660658185
6090664849508012761754614572161769555751992117507514067775104496728590822558547771447242334900
7640263217608921135525612411945387026802990440018385850576719369689759366121356888838680023840
9325673807775018914703049621509969838539752071549396339237202875920415172949370790977853625108
3200928396048072379548870695466216880446521124930762900919907177423550391351174415329737479300
8995583051888413533479846411368000499940373724560035428811232632821866113106455077289922996946
9156018580839820741704606832124388152026099584696588161375826382921029547343888832163627122302
9212297953848683554835357106034077891774170263636562027269554375177807413134551018100094688094
0781122057380335371124632958916237089580476224595091825301636909236240671411644331656159828058
3720783439888562390892028440902553829376

Any problems with this?

20

Full distribution tables

x_1	x_2	x_3	…	y	p( )
0	0	0	…	0	*
0	0	0	…	1	*
1	0	0	…	0	*
1	0	0	…	1	*
0	1	0	…	0	*
0	1	0	…	1	*
			…		

- Storing a table of that size is impossible!
- How are we supposed to learn/estimate each entry in the table?

21

Step 1: pick a model

$$p(features, label) = p(y)\prod_{j=1}^{m} p(x_i \mid y, x_1, \dots, x_{i-1})$$

So, far we have made NO assumptions about the data

Model selection involves making assumptions about the data

We've done this before, n-gram language model, parsing, etc.

These assumptions allow us to represent the data more compactly and to estimate the parameters of the model

22

Naïve Bayes assumption

$$p(features, label) = p(y)\prod_{j=1}^{m} p(x_i \mid y, x_1, \dots, x_{i-1})$$

$$p(x_i \mid y, x_1, x_2, \dots, x_{i-1}) = p(x_i \mid y)$$

What does this assume?

23

Naïve Bayes assumption

$$p(features, label) = p(y)\prod_{j=1}^{m} p(x_i \mid y, x_1, \dots, x_{i-1})$$

$$p(x_i \mid y, x_1, x_2, \dots, x_{i-1}) = p(x_i \mid y)$$

Assumes feature i is independent of the the other features *given the label*

Is this true for text, say, with unigram features?

24

Naïve Bayes assumption

$$p(x_i \mid y, x_1, x_2, \ldots, x_{i-1}) = p(x_i \mid y)$$

For most applications, this is not true!

For example, the fact that "San" occurs will probably make it *more likely* that "Francisco" occurs

However, this is often a reasonable approximation:

$$p(x_i \mid y, x_1, x_2, \ldots, x_{i-1}) \approx p(x_i \mid y)$$

25

Naïve Bayes model

$$p(features, label) = p(y)\prod_{j=1}^{m} p(x_j \mid y, x_1, \ldots, x_{j-1})$$

$$= p(y)\prod_{j=1}^{m} p(x_j \mid y)$$ naïve Bayes assumption

$p(x_i \mid y)$ is the probability of a particular feature value given the label

How do we model this?

- for binary features (e.g., "banana" occurs in the text)
- for discrete features (e.g., "banana" occurs x_i times)
- for real valued features (e.g, the text contains x_i proportion of verbs)

26

p(x|y)

Binary features (aka, Bernoulli Naïve Bayes) :

$$p(x_j \mid y) = \begin{cases} \theta_j & if\ x_i = 1 \\ 1-\theta_j & otherwise \end{cases}$$

biased coin toss!

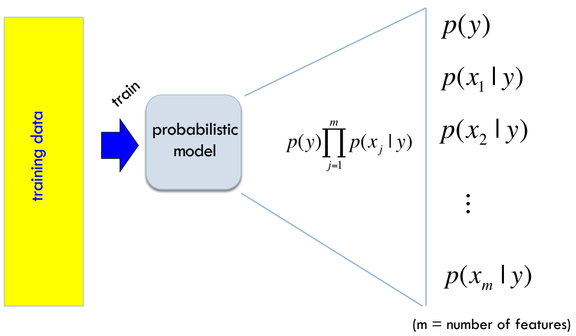
27

Basic steps for probabilistic modeling

	Probabilistic models
Step 1: pick a model	Which model do we use, i.e. how do we calculate p(*feature, label*)?
Step 2: figure out how to estimate the probabilities for the model	How do train the model, i.e. how to we we estimate the probabilities for the model?
Step 3 (optional): deal with overfitting	How do we deal with overfitting?

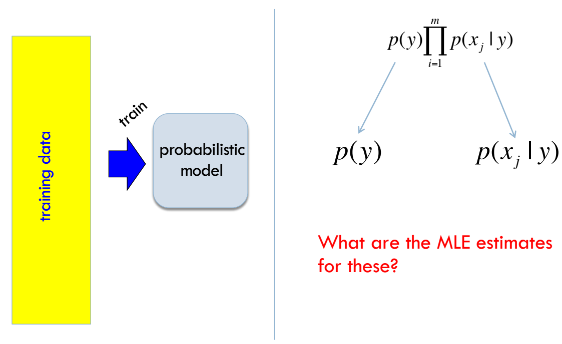
28

Obtaining probabilities



29

MLE estimation for Bernoulli NB



What are the MLE estimates for these?

30

Maximum likelihood estimates

$$p(y) = \frac{count(y)}{n} \qquad \frac{\text{number of examples with label}}{\text{total number of examples}}$$

$$p(x_j \mid y) = \frac{count(x_j, y)}{count(y)} \qquad \frac{\text{number of examples with the label with feature}}{\text{number of examples with label}}$$

What does training a NB model then involve?
How difficult is this to calculate?

31

Text classification

$$p(y) = \frac{count(y)}{n}$$

$$p(w_j \mid y) = \frac{count(w_j, y)}{count(y)}$$

Unigram features:
w_j, whether or not word w_j occurs in the text

What are these counts for text classification with unigram features?

32

Text classification

$$p(y) = \frac{count(y)}{n}$$ number of texts with label / total number of texts

$$p(w_j \mid y) = \frac{count(w_j, y)}{count(y)}$$ number of texts with the label with word w_j / number of texts with label

33

Naïve Bayes classification

yellow, curved, no leaf, 6oz, banana → NB Model p(features, label) → 0.004

$$p(y)\prod_{j=1}^{m} p(x_j \mid y)$$

Given an unlabeled example: yellow, curved, no leaf, 6oz predict the label

How do we use a probabilistic model for classification/prediction?

34

NB classification

probabilistic model: p(features, label1)

yellow, curved, no leaf, 6oz, banana → $p(y=1)\prod_{j=1}^{m} p(x_j \mid y=1)$ →

yellow, curved, no leaf, 6oz, apple → $p(y=2)\prod_{j=1}^{m} p(x_j \mid y=2)$ →

pick largest

label = $\operatorname{argmax}_{y \in labels} p(y)\prod_{j=1}^{m} p(x_j \mid y)$

35

NB classification

probabilistic model: p(features, label1)

yellow, curved, no leaf, 6oz, banana → $p(y=1)\prod_{j=1}^{m} p(x_j \mid y=1)$ →

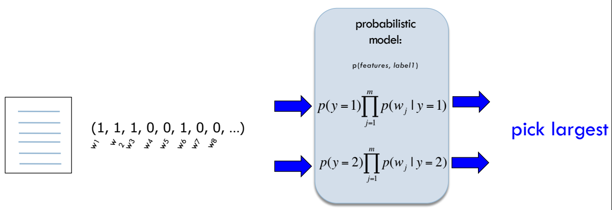
yellow, curved, no leaf, 6oz, apple → $p(y=2)\prod_{j=1}^{m} p(x_j \mid y=2)$ →

pick largest

Notice that each label has its own separate set of parameters, i.e. $p(x_j \mid y)$

36

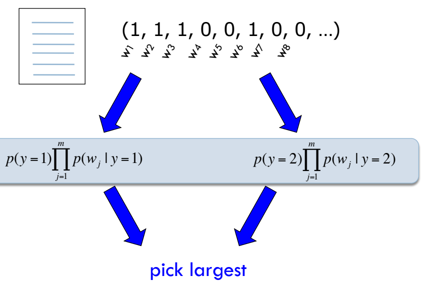
Bernoulli NB for text classification



How good is this model for *text* classification?

37

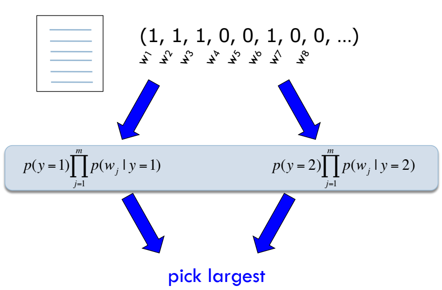
Bernoulli NB for text classification



For text classification, what is this computation? Does it make sense?

38

Bernoulli NB for text classification



Each word that occurs, contributes $p(w_i \mid y)$

Each word that does NOT occur, contributes $1 - p(w_i \mid y)$

39

Generative Story

To classify with a model, we're given an example and we obtain the probability

We can also ask how a given model would *generate* an example

This is the "generative story" for a model

Looking at the generative story can help understand the model

We also can use generative stories to help develop a model

40

Bernoulli NB generative story

$$p(y)\prod_{j=1}^{m} p(x_j \mid y)$$

What is the generative story for the NB model?

41

Bernoulli NB generative story

$$p(y)\prod_{j=1}^{m} p(x_j \mid y)$$

1. Pick a label according to p(y)
   - roll a biased, num_labels-sided die
2. For each feature:
   - Flip a *biased* coin:
     - if heads, include the feature
     - if tails, don't include the feature

What does this mean for text classification, assuming unigram features?

42

Bernoulli NB generative story

$$p(y)\prod_{j=1}^{m} p(w_j \mid y)$$

1. Pick a label according to p(y)
   - roll a biased, num_labels-sided die
2. For each word in your vocabulary:
   - Flip a *biased* coin:
     - if heads, include the word in the text
     - if tails, don't include the word

43

Bernoulli NB

$$p(y)\prod_{j=1}^{m} p(x_j \mid y)$$

Pros/cons?

44

Bernoulli NB

Pros

- Easy to implement
- Fast!
- Can be done on large data sets

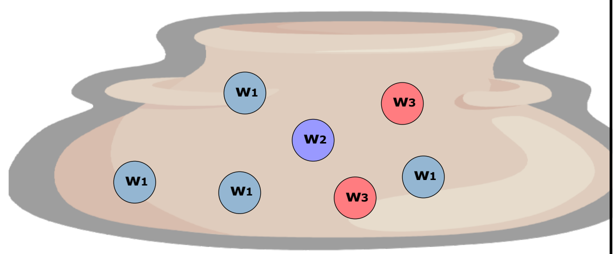
Cons

- Naïve Bayes assumption is generally not true
- Performance isn't as good as other models
- For text classification (and other sparse feature domains) the $p(x_i=0|y)$ can be problematic

45

Another generative story

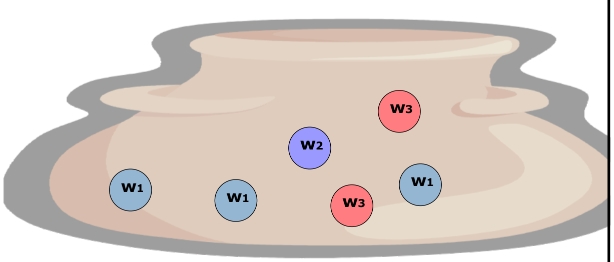
Randomly draw words from a "bag of words" until document length is reached



46

Draw words from a fixed distribution

Selected: w_1



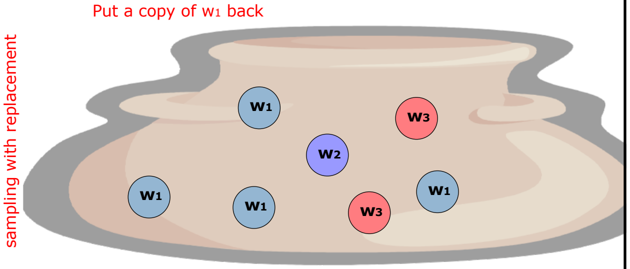
47

Draw words from a fixed distribution

Selected: w_1

Put a copy of w_1 back

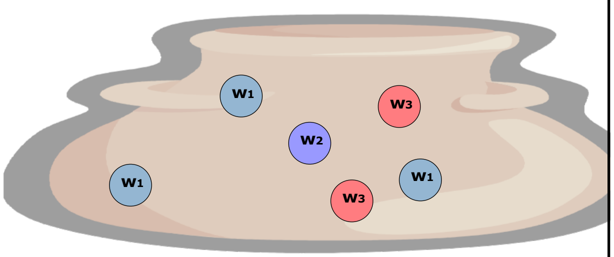
sampling with replacement



48

Draw words from a fixed distribution

Selected: w_1 w_1



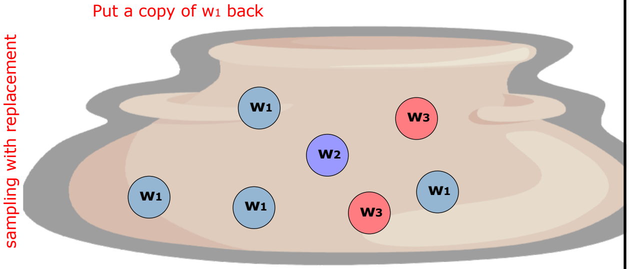
49

Draw words from a fixed distribution

Selected: w_1 w_1

Put a copy of w_1 back

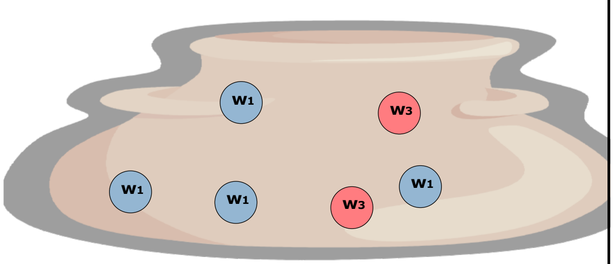
sampling with replacement



50

Draw words from a fixed distribution

Selected: w_1 w_1 w_2



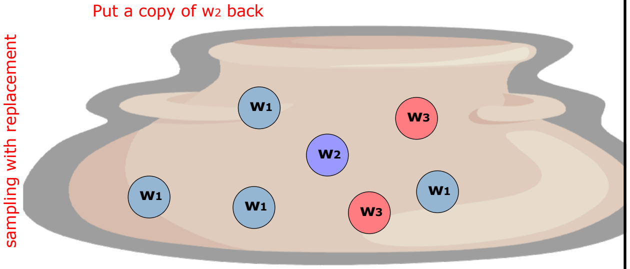
51

Draw words from a fixed distribution

Selected: w_1 w_1 w_2

Put a copy of w_2 back

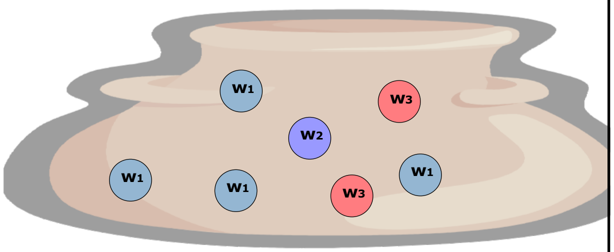
sampling with replacement



52

Draw words from a fixed distribution

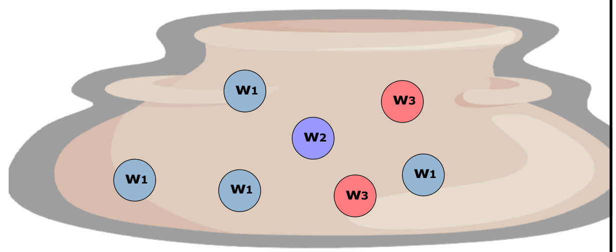
Selected: w1 w1 w2 …



53

Draw words from a fixed distribution

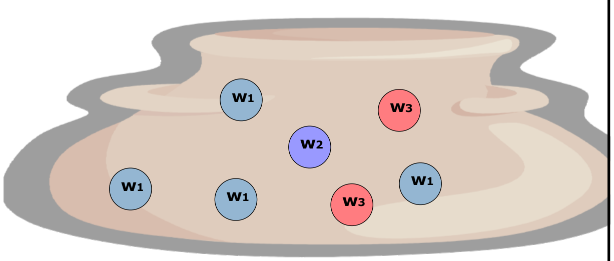
Is this a NB model, i.e. does it assume each individual word occurrence is independent?



54

Draw words from a fixed distribution

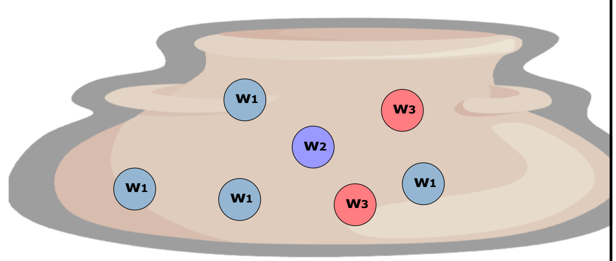
Yes! Doesn't matter what words were drawn previously, still the same probability of getting any particular word



55

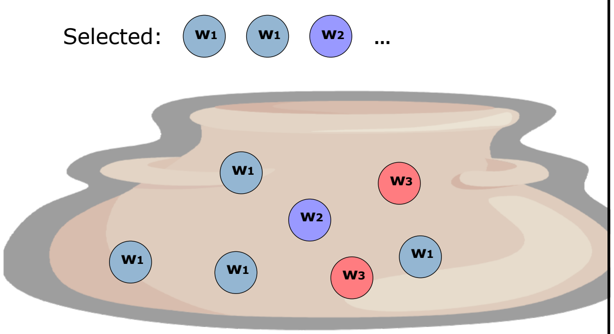
Draw words from a fixed distribution

Does this model handle multiple word occurrences?



56

Draw words from a fixed distribution



57