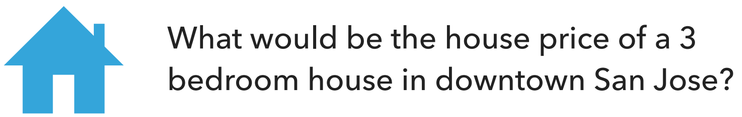
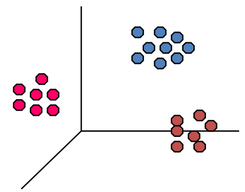
CMPE 297, INSTRUCTOR: JORJETA JETCHEVA

NAIVE BAYES & CLASSIFICATION

MACHINE LEARNING OVERVIEW

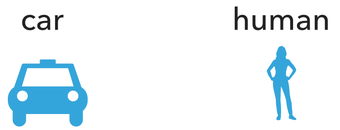
- "A scientific field is best defined by the central question it studies. The field of Machine Learning seeks to answer the question: How can we build computer systems that automatically improve with experience, and what are the fundamental laws that govern all learning processes?", Tom Mitchell (CMU)
- Common types of ML problems:
  - Classification (predict a label/category)
  - Regression (predict a value)
  - Clustering (group similar items together)



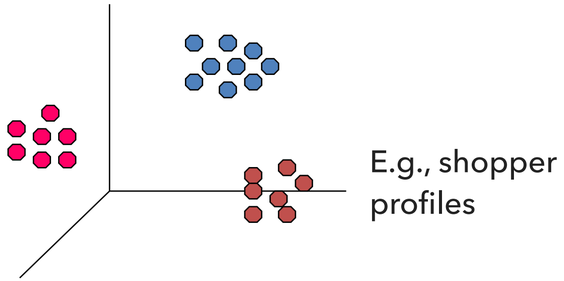


MAJOR TYPES OF MACHINE LEARNING

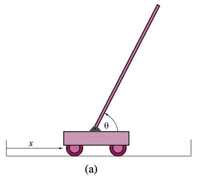
Main types of machine learning approaches based on input feedback:



- Supervised learning: the agent observes (is trained on) input-output pairs and learns a function that maps from input to output
  - E.g., the agent is trained on a number of images labeled as either "car" or "human" (this is the supervisory signal) and the resulting model can be used to label new images that the agent hasn't seen before
  - Note that supervised learning involves labeled examples (often labeled by humans)



- Unsupervised learning: the agent learns patterns from the input (data) without any explicit feedback
  - E.g., clustering users into "likely to buy more stuff if they get a coupon for 20% off" vs. "likely to buy more stuff if they get a buy second item of the same kind for 40% off"



- Reinforcement learning: the agent is trained with the use of rewards and punishments in response to its actions
  - Figures out how to alter its actions in the future in order to maximize rewards and/or minimize punishment

SUPERVISED LEARNING

More formally, the task of supervised learning can be described as follows:

- Given a training set of N example input-output pairs

  $(x_1,y_1), (x_2,y_2),\ldots, (x_N,y_N)$

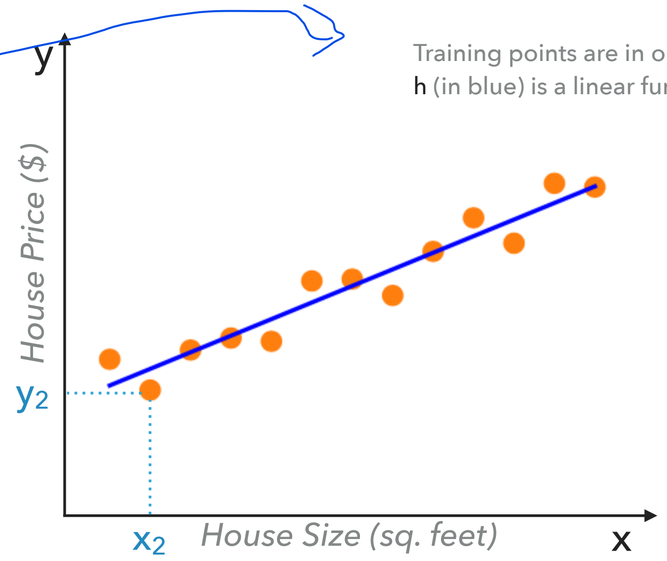
  where each pair was generated by an unknown function $y = \mathbf{f}(x)$,

  we need to discover a function **h** that approximates the true function **f**.

- The function **h** is called a *hypothesis*

- The output values, y_i, (that are in our training set) are the ground truth

- Ideally, we want to find **h** such that $h(x_i) = y_i$ though often that's not possible and so we generally look for the best-fit function where $h(x_i)$ is close to y_i.



From Figure 19.1 in Russel & Norvig

Note: what is more important than finding h that fits the training set best, is how well it generalizes to unseen examples

TEXT CLASSIFICATION

- Funny classification example from our textbook

  Jorge Luis Borges (1964) wanted to classify animals into:

  (a) those that belong to the Emperor, (b) embalmed ones, (c) those that are trained, (d) suckling pigs, (e) mermaids, (f) fabulous ones, (g) stray dogs, (h) those that are included in this classification, (i) those that tremble as if they were mad, (j) innumerable ones, (k) those drawn with a very fine camel's hair brush, (l) others, (m) those that have just broken a flower vase, (n) those that resemble flies from a distance.

- Text classification involves assigning a label to a text document (sentence, paragraph, article), e.g.,
  - Topic (e.g., politics, science, etc.)
  - Positive or Negative Sentiment
  - Spam vs. legitimate
  - Fake vs. legitimate
  - Author attribution

NAÏVE BAYES

FREQUENTIST APPROACH TO PROBABILITY

- Probabilities are found after repeated experiments, which help find the true parameters of a probability distribution, e.g., mean and variance

BAYESIAN APPROACH

- In Bayesian statistics, probability measures the degree of belief in a hypothesis and is updated based on new evidence
  - The probability of A, P (A), aka the prior probability of A, is the initial degree of belief in A
    - E.g., P(Meningitis) = 0.00002
  - The probability of A given we know B is true, P (A | B), aka the posterior probability of A given B, is the degree of belief after incorporating news that B is true
    - E.g., P (Meningitis | Stiff Neck): Given patient has stiff neck, what's the probability s/he has meningitis?

JOINT PROBABILITY OF TWO (OR MORE) VARIABLES

- The joint probability of two variables A and B (e.g., denoted P(A,B)) is the probability of a particular value for A to occur at the same time as a particular value of B, e.g.,
  - What is the probability of a person in the general population to be both 7ft tall and 100 pounds (A is height and B is weight)?
  - What is he probability of a person in the general population to be older than 80 and not have dementia (A is age and B is not having dementia)?
- The above applies to more than two variables as well

JOINT PROBABILITY DISTRIBUTION

- Joint probability distribution:
  - Tossing two coins is an event with an outcome that can be either (Coin$_1$=head, Coin$_2$=head), (tail, tail), (head, tail), (tail, head).
  - Each of these four outcomes has an associated probability.
  - The set of these probabilities is the joint probability distribution of the two (coin toss random) variables

Coin$_1$	Coin$_2$	Probability
Head	Head	…
Head	Tail	…
Tail	Head	…
Tail	Tail	…

BAYESIAN CLASSIFICATION

A probabilistic framework for solving classification problems.

Conditional probability of H given E

$$P(H \mid E) = \frac{P(H, E)}{P(E)}$$

Joint probability of H & E

Prior probability of E

Bayes Theorem

$$P(H \mid E) = \frac{P(E \mid H)P(H)}{P(E)}$$

H is also referred to as the Hypothesis, and E as the Evidence

EXAMPLE OF BAYES THEOREM

- Given:
  - A doctor knows that meningitis causes stiff neck 50% of the time — P(*S* | *M*)
  - Prior probability of any patient having meningitis is 1/50,000 — P(*M*)
  - Prior probability of any patient having stiff neck is 1/20 — P(*S*)

- If a patient has stiff neck, what's the probability he/she has meningitis?

$$\mathrm{P}(M \mid S) = \frac{P(S \mid M)P(M)}{P(S)} = \frac{0.5 \times 1/50000}{1/20} = 0.0002$$

CONDITIONAL INDEPENDENCE

- If A & B are conditionally independent given C, then $\Pr(A \cap B \mid C) = \Pr(A \mid C) \Pr(B \mid C)$
  - Given C, knowledge of whether A occurs does not provide information on the likelihood of B occurring

Example: Height and reading skills

- If we don't have the Age column, it looks like height is a predictor of reading skills and vice versa
- Given a certain age, e.g., 3, we can predict reading skills and height independently of each other

	Height (feet)	Reading Skills	Age
1	3	0	3
2	3.5	0	4
3	4.5	1	6
4	5	2	20
5	6	2	43
6	3.8	0	4
7	6.2	2	18
8	5.8	2	71
9	4	1	8
10	3.9	1	7

USING BAYES THEOREM FOR CLASSIFICATION

- Consider each attribute and class label as random variables
- Given a record with attributes $(X_1, X_2, \ldots, X_d)$
  - Goal is to predict class Y_j (to which this record belongs)
  - Specifically, we want to find the value of Y_j that maximizes $P(Y_j \mid X_1, X_2, \ldots, X_d)$
    - I.e., what is the most likely class assignment for this record

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

- Can we estimate $P(Y_j \mid X_1, X_2, \ldots, X_d)$ for all classes directly from data?

HOW DOES NAÏVE BAYES WORK WITH A LIST OF VARIABLES (RATHER THAN JUST ONE)

Applying the Bayes Theorem:

$$P\left(Y_j \middle| X_1 \ldots X_n\right) = \frac{P\left(X_1 \ldots X_n \middle| Y_j\right) P(Y_j)}{P(X_1 \ldots X_n)}$$

Assume conditional independence among attributes X_i, (i.e., when class label is given, the attributes are independent):

$$P(X_1 \ldots X_n \mid Y_j) = P(X_1 \mid Y_j)\, P(X_2 \mid Y_j) \ldots P(X_n \mid Y_j)$$

Classify test sample to class Y_j for which we get max value below among all classes $Y_1 \ldots Y_m$

$$P\left(Y_j \middle| X_1 \ldots X_n\right) = \frac{P\left(Y_j\right) \times \prod_{i=1}^{n} P\left(X_i \middle| Y_j\right)}{P\left(X_1 \ldots X_n\right)}$$

Class (Y_i) for which we get the largest nominator is the most likely class

$P(X_i \mid Y_j)$ for all X_i and Y_j combinations can be computed from the training data!

NAÏVE BAYES CLASSIFICATION EXAMPLE (1)

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

Test record whose class we need to predict:

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?

A: the set of attributes $X_1 \ldots X_n$

M: mammals

N: non-mammals

Predicted class for the test record above is M(ammals) or N(on-mammals)?

$$P(M \mid X_1 \ldots X_n) \quad ? \quad P(N \mid X_1 \ldots X_n)$$

NAÏVE BAYES CLASSIFICATION EXAMPLE (2)

$$P(M \mid X_1 \ldots X_n) = P(M) \times \prod_{i=1}^{n} P(X_i \mid M) = \; ?$$

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

Test record whose class we need to predict:

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?

$X_1 \ldots X_n$: attributes

M: mammals

N: non-mammals

$$P(X_1 \ldots X_n \mid M) = P(X_1 \mid M)\, P(X_2 \mid M) \ldots P(X_n \mid M)$$

$$P(X_1 \ldots X_n \mid M) = \frac{6}{7} \times \frac{6}{7} \times \frac{2}{7} \times \frac{2}{7} = 0.06$$

$$P(M) =$$

$$P(M \mid X_1 \ldots X_n) = P(M) \times \prod_{i=1}^{n} P(X_i \mid M) =$$

NAÏVE BAYES CLASSIFICATION EXAMPLE (3)

$$P(N \mid X_1 \ldots X_n) = P(N) \times \prod_{i=1}^{n} P(X_i \mid N) = \; ?$$

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

Test record whose class we need to predict:

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?

$X_1 \ldots X_n$: attributes

M: mammals

N: non-mammals

$$P(X_1 \ldots X_n \mid N) = P(X_1 \mid N)\, P(X_2 \mid N) \ldots P(X_n \mid N)$$

$$P(X_1 \ldots X_n \mid N) =$$

$$P(N) =$$

$$P(N \mid X_1 \ldots X_n) = P(N) \times \prod_{i=1}^{n} P(X_i \mid N) =$$

NAÏVE BAYES CLASSIFICATION EXAMPLE (4)

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

Test record whose class we need to predict:

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?

A: the set of attributes $X_1 \ldots X_n$

M: mammals

N: non-mammals

Predicted class for the test record above is M(ammals) or N(on-mammals)?

$$P(M \mid X_1 \ldots X_n) \quad ? \quad P(N \mid X_1 \ldots X_n)$$

NAIVE BAYES CLASSIFICATION FOR TEXT: MULTINOMIAL NAIVE BAYES CLASSIFIER

- We represent a text document as a “bag of words” - only the frequency of appearance of each word matters
- The predicted class is the one with the highest (posterior) probability among the set of classes C, given the document

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} P(c|d)$$

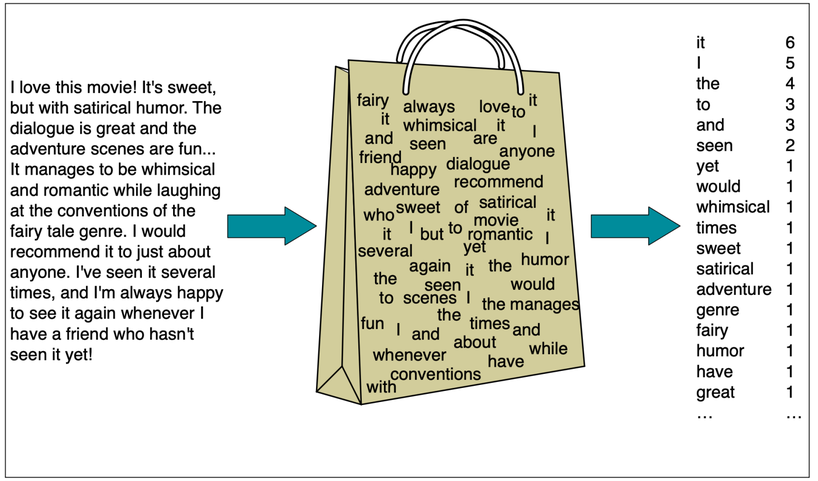


Figure 4.1 Intuition of the multinomial naive Bayes classifier applied to a movie review. The position of the words is ignored (the *bag of words* assumption) and we make use of the frequency of each word.

When we apply Bayes rule, we get:

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} P(c|d) = \underset{c \in C}{\operatorname{argmax}} \frac{P(d|c)P(c)}{P(d)}$$

MULTINOMIAL NAIVE BAYES CLASSIFIER (CONT.)

$$P(w_{1:n}) \approx \prod_{j=1}^{n} P(w_k \mid w_{k-1})$$

Given $$\hat{c} = \underset{c \in C}{\operatorname{argmax}} P(c|d) = \underset{c \in C}{\operatorname{argmax}} \frac{P(d|c)P(c)}{P(d)}$$

Each document is a sequence of words w_i

We assume conditional independence as in our previous Bayes derivation, and arrive at c_{NB}, the class that Naive Bayes (NB) predicts for a document d

$$c_{NB} = \underset{c \in C}{\operatorname{argmax}} P(c) \prod_{i \in positions} P(w_i|c)$$

*A doc is a set of words **w**, one word at each position **i***

Log-based computation to avoid underflow (as we did in n-gram language models)

$$c_{NB} = \underset{c \in C}{\operatorname{argmax}} \log P(c) + \sum_{i \in positions} \log P(w_i|c)$$

NAIVE BAYES AS A GENERATIVE CLASSIFIER

Let's take another look at the high-level Bayes formulation of the model

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} P(c|d) = \underset{c \in C}{\operatorname{argmax}} \frac{P(d|c)P(c)}{P(d)}$$

- Naive Bayes is a generative classifier
  - Bayes rule-based formulation above indicates that we can first sample the class based on P(c), then generate the document by sampling words from the class (P(d | c))
  - Model is built of how a class could generate particular input data (as opposed to a discriminative model which learns features to help it differentiate between the classes)
  - Given an observation, Bayes identifies the class most likely to have generated the observation

TRAINING THE NAIVE BAYES CLASSIFIER

To compute the likelihood of a word, given a class **c**,

- We treat all docs in a class **c** as one document that includes all of them (we concatenate the docs together)
- Then we count how many times a word occurs in the concatenated document belonging to the class **c** relative to the number of types of words that occur in the concatenated document

$$\hat{P}(w_i|c) = \frac{count(w_i, c)}{\sum_{w \in V} count(w, c)}$$

Example from sentiment analysis:

$$\hat{P}(\text{“fantastic”}|\text{positive}) = \frac{count(\text{“fantastic”}, \text{positive})}{\sum_{w \in V} count(w, \text{positive})}$$

And we need smoothing to ensure that words that do not appear for docs in a particular class, do not result in 0 overall probability (Laplace works well here):

$$\hat{P}(w_i|c) = \frac{count(w_i, c) + 1}{\sum_{w \in V} (count(w, c) + 1)} = \frac{count(w_i, c) + 1}{\left(\sum_{w \in V} count(w, c)\right) + |V|}$$

*The vocabulary **V** includes all words across all classes*

Note: NB ignores unknown words, i.e., ones not in the vocabulary (removes them from the test set)

OPTIMIZATIONS FOR SENTIMENT ANALYSIS

- Binary (Multinomial) Naive Bayes only looks at how many documents of a particular class a particular word appears in (at least one) but doesn't care about (counting) how many times the word appears
  - Turns out this works well for sentiment analysis
- We concatenate all docs for a class into 1 document as before, but remove all duplicates of each word within each document before we concatenate; from then on, we proceed with the training of the model as before

Four original documents:

- − it was pathetic the worst part was the boxing scenes
- − no plot twists or great scenes
- + and satire and great plot twists
- + great scenes great film

After per-document binarization:

- − it was pathetic the worst part boxing scenes
- − no plot twists or great scenes
- + and satire great plot twists
- + great scenes film

	NB Counts		Binary Counts	
	+	−	+	−
and	2	0	1	0
boxing	0	1	0	1
film	1	0	1	0
great	3	1	2	1
it	0	1	0	1
no	0	1	0	1
or	0	1	0	1
part	0	1	0	1
pathetic	0	1	0	1
plot	1	1	1	1
satire	1	0	1	0
scenes	1	2	1	2
the	0	2	0	1
twists	1	1	1	1
was	0	2	0	1
worst	0	1	0	1

Figure 4.3 An example of binarization for the binary naive Bayes algorithm.

OPTIMIZATIONS FOR SENTIMENT ANALYSIS (CONT.)

- Preserve negations as they are critical to sentiment expression
  - Pre-pend the word NOT_ to every word after a token that negates, e.g.,

    "Didn't like this movie, but I" becomes "Didn't NOT_like NOT_this NOT_movie, but I"

- Use a lexicon of negative and positive words (especially if we don't have enough training data)

OTHER PROMINENT APPLICATIONS OF NAIVE BAYES

- Naive Bayes is an effective model for detecting (email) spam
  - Typically we would add linguistic and non-linguistic features as well, e.g.
    - Email subject is all caps
    - Phrases like “million dollars”, “urgent reply”, etc.
    - Mail server routing path
- Language identification - what language a particular piece of text is in

NAIVE BAYES VS. LANGUAGE MODELS

- Naive Bayes applied to only words (as opposed to non-linguistic features as well)

  where we include all words in the text,

  is the same as a unigram model for each class

- In particular, Naive Bayes can assign a probability to any sentence, conditioned on a class, e.g., example in your book:

  P("I love this fun film" | +) = $0.1 \times 0.1 \times 0.01 \times 0.05 \times 0.1 = 0.0000005$

  P("I love this fun film" | −) = $0.2 \times 0.001 \times 0.01 \times 0.005 \times 0.1 = .0000000010$