The Naïve Bayes Algorithm

Maria-Florina Balcan

02/05/2018

Probabilistic Approach to Learning

Instead of learning $\mathrm{F}: \mathrm{X} \rightarrow \mathrm{Y}$, learn P(Y|X).

Can design algorithms that learn functions with uncertain outcomes (e.g., predicting tomorrow’s stock price) **and that incorporate prior knowledge to guide learning** (e.g., a bias that tomorrow’s stock price is likely to be similar to today’s price).

Bayes Rule

Bayes, Thomas (1763) An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London*, 53:370-418

Bayes Rule: $\mathbf{P(A|B) = \dfrac{P(B|A)P(A)}{P(B)}}$

$P(A)$ prior

$P(A|B)$ posterior

…by no means merely a curious speculation in the doctrine of chances, but necessary to be solved in order to a sure foundation for all our reasonings concerning past facts, and what is likely to be hereafter…. necessary to be considered by any that would give a clear account of the strength of *analogical* or *inductive reasoning…*

Naïve Bayes in a Nutshell, Y, X_i Discrete

Pick the most probably Y for $X^{new} = (X_1^{new}, X_2^{new}, \ldots, X_n^{new})$

$$Y^{new} = \mathrm{argmax}_{y_k} P(Y = y_k | X_1^{new}, \ldots, X_n^{new})$$

Bayes Rule: $P(Y = y_k | X_1, \ldots, X_n) = \dfrac{P(Y = y_k) P(X_1, \ldots, X_n | Y = y_k)}{P(X_1, \ldots, X_n)}$

Main problem: learning P(X|Y) might require more data than we have…

Example:

n=100 attributes

Number of rows in this table? $2^{100} \sim 100^{10} \sim 10^{30}$

Number of people on Earth? 10^{9}

Fraction of rows with 0 training examples: 0.9999

Naïve Bayes algorithms assume Conditional Independence

X_i and X_j are conditionally independent given Y, for all $i \neq j$

$$P(X_1, X_2, \dots, X_n | Y) = \prod_i P(X_i | Y)$$

If $X_1, \dots, X_n, Y$ are all Boolean, how many parameters do we need to describe $P(X_1, X_2, \dots, X_n | Y)$?

- Without the conditional independence assumption: $2(2^n - 1)$
- With conditional independence assumption: $2n$

Naïve Bayes in a Nutshell

Bayes Rule: $P(Y = y_k | X_1, \ldots, X_n) = \dfrac{P(Y = y_k) P(X_1, \ldots, X_n | Y = y_k)}{P(X_1, \ldots, X_n)}$

If X_i and X_j are conditionally independent given Y, for all $i \neq j$

$$P(Y = y_k | X_1, \ldots, X_n) = \frac{P(Y = y_k) \prod_i P(X_i | Y = y_k)}{P(X_1, \ldots, X_n)}$$

So, to pick the most probably Y for $X^{new} = (X_1^{new}, X_2^{new}, \ldots, X_n^{new})$

$$Y^{new} = \operatorname{argmax}_{y_k} P(Y = y_k) \prod_i P(X_i^{new} | Y = y_k)$$

Naïve Bayes: discrete X_i

Training phase (input: training examples)

- For each value y_k, estimate $\pi_k = P(Y = y_k)$; get $\widehat{\pi_k}$
- For each value x_{ij} of attribute X_i estimate $\theta_{i,j,k} = P(X_i = x_{ij} | Y = y_k)$; get $\widehat{\theta_{i,j,k}}$

Testing phase:

- Classify $X^{new} = (X_1^{new}, X_2^{new}, \ldots, X_n^{new})$

$$Y^{new} = \operatorname{argmax}_{y_k} \widehat{\pi_k} \prod_i \widehat{\theta_{i,new,k}}$$

[Ideal rule: $Y^{new} = \operatorname{argmax}_{y_k} P(Y = y_k) \prod_i P(X_i^{new} | Y = y_k)$]

Estimating parameters Y, X_i discrete

Maximum Likelihood Estimation

- For each value y_k, get $\widehat{\pi_k} = \widehat{P}(Y = y_k) = \frac{\#D(Y=y_k)}{|D|}$

number of training data points with $Y = y_k$ divided by the total number of training data points

- For each value x_{ij} of attribute X_i estimate $\theta_{i,j,k} = P(X_i = x_{ij} | Y = y_k)$;

get $\widehat{\theta_{i,j,k}} = \widehat{P}(X_i = x_{ij} | Y = y_k) = \frac{\#D\left(X_i = x_{ij} \wedge Y = y_k \right)}{\#D(Y = y_k)}$

The fraction of datapoints with $Y = y_k$ that also have $X = x_{i,j}$.

Sublety 1: Violation of the Naïve Bayes Assumption

- Usually features are not conditionally independent given the label

$$\mathrm{P}(\mathrm{X}_1, \mathrm{X}_2, \ldots, \mathrm{X}_\mathrm{n} | \mathrm{Y}) \neq \prod_{\mathrm{i}} \mathrm{P}(\mathrm{X}_\mathrm{i} | \mathrm{Y})$$

- Nonetheless, NB is widely used:
  - NB often performs well, even when assumption is violated
  - [Domingos & Pazzani ’96] discuss some conditions for good performance

Subtlety 2: Need to use MAP

$$Y^{new} = \mathrm{argmax}_{y_k}\ \hat{P}(Y = y_k | X_1, \dots, X_n) = \mathrm{argmax}_{y_k}\ \hat{P}(Y = y_k) \prod_i \hat{P}(X_i^{new} | Y = y_k)$$

Note: If we never see a certain combination $X_i = a$ and $Y = b$ in our training data, then on any new example with $X_i = a$ we will predict a zero probability of $Y = b$

E.g., if we never see a training instance where $X_1 = a$ and $Y = b$?

$$\text{e.g., } Y = \text{SpamEmail}, X = \text{"Earn"} \qquad \hat{P}(X_1 = a | Y = b) = 0$$

- Thus no matter what the values $X_2^{new}, \dots, X_n^{new}$ take, we get

$$\hat{P}(Y = b | X_1^{new} = a, X_2^{new}, \dots, X_n^{new}) = 0$$

- Solution: use MAP estimate!!!!

Estimating parameters Y, X_i discrete

Maximum A Posteriori Estimation

K - number of distinct values label can take; l determines the strength of this smoothing; assume the hallucinated examples are spread evenly over the possible values of Y; so, number of hallucinated examples is lK.

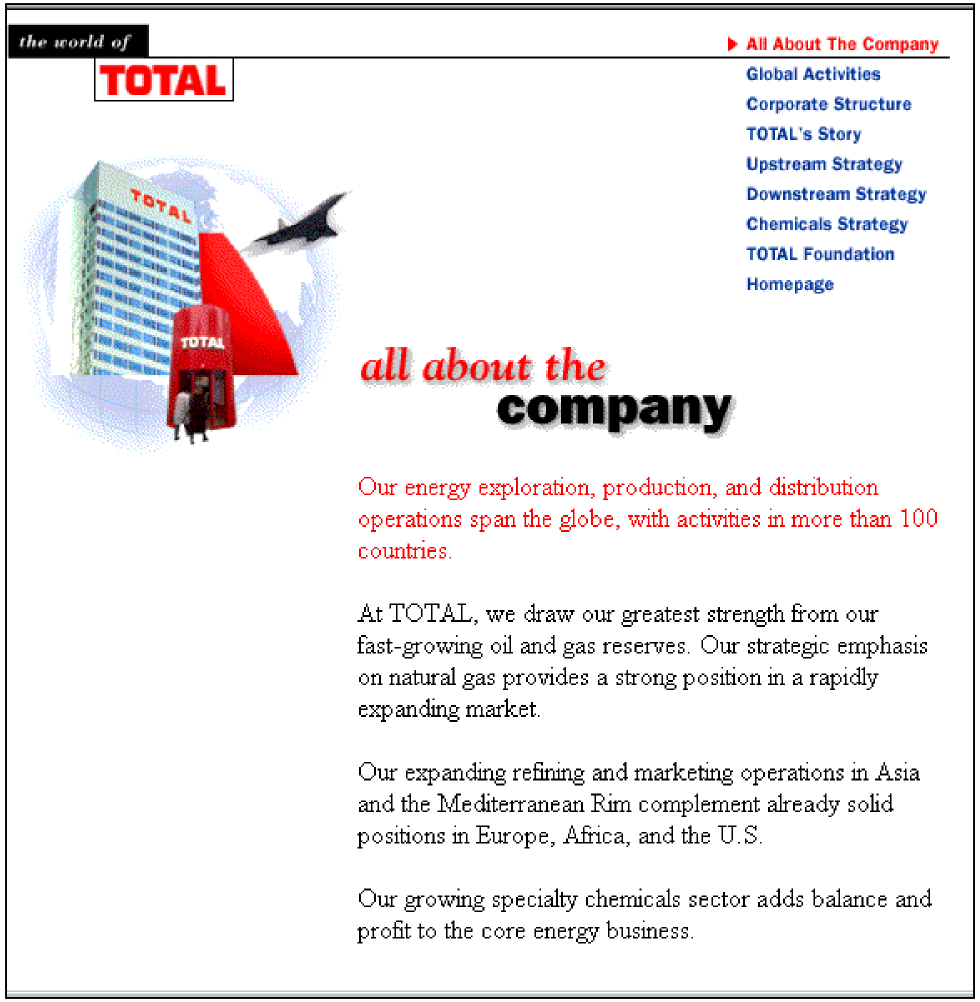
- For each value y_k, get $\widehat{\pi_k} = \widehat{P}(Y = y_k) = \frac{\#D(Y=y_k)+l}{|D|+lK}$

- For each value x_{ij} of attribute X_i estimate $\theta_{i,j,k} = P(X_i = x_{ij}|Y = y_k)$;

get $\widehat{\theta_{i,j,k}} = \widehat{P}(X_i = x_{ij}|Y = y_k) = \frac{\#D(\,X_i=x_{ij} \wedge Y=y_k)+l}{\#D(Y=y_k)+lJ}$

J - number of distinct values that feature i can take; l determines the strength of this smoothing; assume the hallucinated examples are spread evenly over the possible values of X_i; so, number of hallucinated examples is lJ.

Bag of Words Approach



the world of

TOTAL

▸ All About The Company

Global Activities

Corporate Structure

TOTAL's Story

Upstream Strategy

Downstream Strategy

Chemicals Strategy

TOTAL Foundation

Homepage

all about the **company**

Our energy exploration, production, and distribution operations span the globe, with activities in more than 100 countries.

At TOTAL, we draw our greatest strength from our fast-growing oil and gas reserves. Our strategic emphasis on natural gas provides a strong position in a rapidly expanding market.

Our expanding refining and marketing operations in Asia and the Mediterranean Rim complement already solid positions in Europe, Africa, and the U.S.

Our growing specialty chemicals sector adds balance and profit to the core energy business.

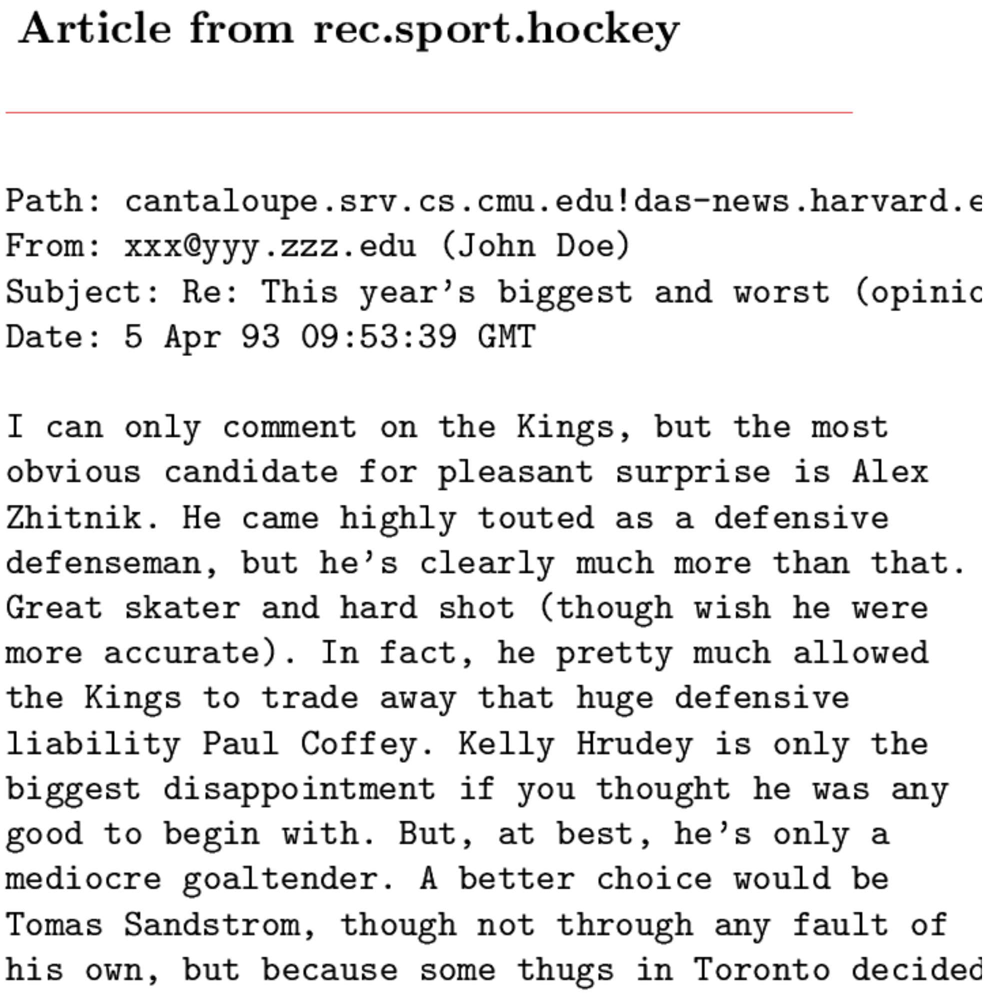
aardvark	0
about	2
all	2
Africa	1
apple	0
anxious	0
...	
gas	1
...	
oil	1
…	
Zaire	0

Case Study: Text Classification

- Classify e-mails
  - $Y = \{\text{Spam, NotSpam}\}$
- Classify news articles
  - $Y =$ what is the topic of the article?
- Classify webpages
  - $Y = \{\text{student, professor, project, …}\}$

- What about the features X?
  - The text!

Features X are entire document - X_i for ith word in article



Article from rec.sport.hockey

```
Path: cantaloupe.srv.cs.cmu.edu!das-news.harvard.e
From: xxx@yyy.zzz.edu (John Doe)
Subject: Re: This year's biggest and worst (opinic
Date: 5 Apr 93 09:53:39 GMT

I can only comment on the Kings, but the most
obvious candidate for pleasant surprise is Alex
Zhitnik. He came highly touted as a defensive
defenseman, but he's clearly much more than that.
Great skater and hard shot (though wish he were
more accurate). In fact, he pretty much allowed
the Kings to trade away that huge defensive
liability Paul Coffey. Kelly Hrudey is only the
biggest disappointment if you thought he was any
good to begin with. But, at best, he's only a
mediocre goaltender. A better choice would be
Tomas Sandstrom, though not through any fault of
his own, but because some thugs in Toronto decided
```

Naïve Bayes for Text Classification

- What are the features: X_i represents ith word in document.
  - the domain of X_i is entire vocabulary, e.g., Webster Dictionary, 10,000 words
- E.g., if article has 1000 words, $X = \{X_1, \ldots, X_{1000}\}$, then domain of X has size 10000^{1000}.
- $P(X|Y)$ is huge!

- **Naïve Bayes assumption helps a lot!**
  - Meaning of naïve Bayes assumption: the word in position i is independent of all the other words in the document given the label y

Naïve Bayes for Text Classification

- **Naïve Bayes assumption helps a lot!**
  - $P(X_i = x_i | Y = y)$ is just the probability of observing word x_i at the ith position in a document on topic y.
  - Assume X_i is independent of all other words in document given the label y: $P(X_i = x_i | Y = y, X_{-i}) = P(X_i = x_i | Y = y)$.

$$h_{NB}(x) = \arg\max_{y} P(y) \prod_{i=1}^{\text{lengthDoc}} P(X_i = x_i | y)$$

  - For each label y, have 1000 distributions of size 10000 to estimate.
  - This is 10000×1000 items, which is big but much less than 10000^{1000}…

Bag of Words Model

- Typical additional assumption – **Position in document doesn't matter**:

$$P(X_i = x_i \mid Y = y) = P(X_k = x_i \mid Y = y)$$

the probability distributions of words are the same at each position: $P_i = P_j$ for all i, j.

- **"Bag of Words"** model – order of words in the document is ignored

- Now, only 10000 quantities $P(x_i|y)$ to estimate for each label y (the 10000 possible values that x_i can be) plus the prior.

$$h_{NB}(x) = \arg\max_{y} P(y) \prod_{i=1}^{1000} P(x_i|y)$$

Bag of Words model

- Typical additional assumption – **Position in document doesn't matter**:

$$P(X_i = x_i \mid Y = y) = P(X_k = x_i \mid Y = y)$$

- **"Bag of Words"** model – order of words on the page ignored
- Sounds silly but often works very well

A piece of text like "When the lecture is over, remember to take your bag" would look to this algorithm the same as if we just sorted the words alphabetically*"bag is lecture over remember take the to When your"*

Bag of Words model

- Typical additional assumption – **Position in document doesn't matter**:

$$P(X_i = x_i \mid Y = y) = P(X_k = x_i \mid Y = y)$$

- **"Bag of Words"** model – order of words on the page ignored

Can simplify further:

$$\prod_{i=1}^{\mathrm{lengthDoc}} P(x_i|y) = \prod_{w=1}^{W} P(w|y)^{\mathrm{count}(w)}$$

Bag of Words representation

- Since we are assuming the order of words doesn't matter, an alternative representation of document is as vector of counts:
  - $\mathrm{x}^{(\mathrm{j})} =$ number of occurrences of word j in document x.
  - Typical document: $[0\ 0\ 1\ 0\ 0\ 3\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 2\ 0\ 0 \ldots]$
  - Called "bag of words" or "term vector" or "vector space model" representation

Naïve Bayes with Bag of Words for text classification

- Learning phase
  - Class Prior $\mathrm{P(Y)}$
  - $\mathrm{P(X_i|Y)}$
- Test phase:
  - For each document
    - Use naïve Bayes decision rule

$$\mathrm{h_{NB}(x) = \arg\max_{y} P(y) \prod_{i=1}^{1000} P(x_i|y)}$$

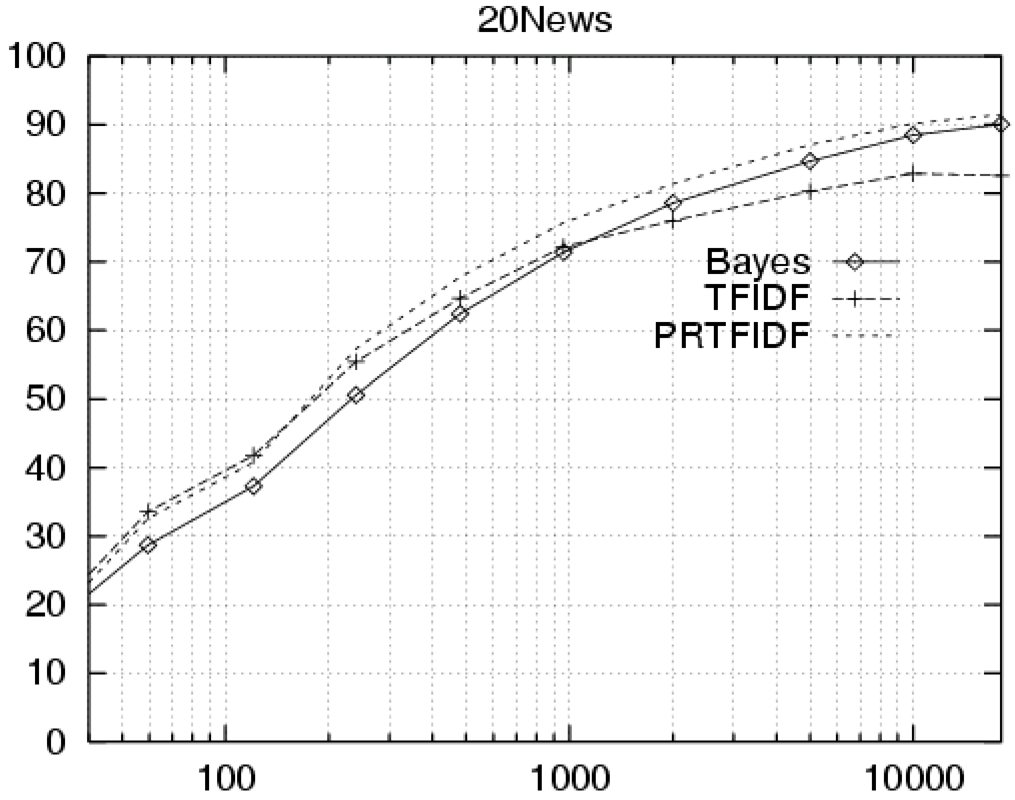
Twenty news groups results

- Given 1000 training documents from each group, learn to classify new documents according to which newsgroup it came from

comp.graphics, comp.os.ms-windows.misc, comp.sys.ibm.pc.hardware, comp.sys.max.hardware, comp.windows.x,
misc.forsale, rec.autos, rec.motorcycles, rec.sport.baseball, rec.sport.hockey
alt.atheism, soc.religion.christian, talk.religion.misc, talk.politics.mideast, talk.politics.misc, talk.politics.guns,
sci.space, sci.crypt, sci.electronics, sci.med

- Naïve Bayes: 89% classification accuracy

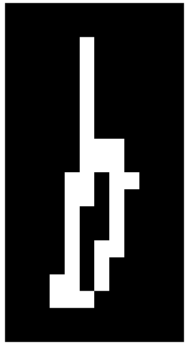
Learning curve for twenty news groups

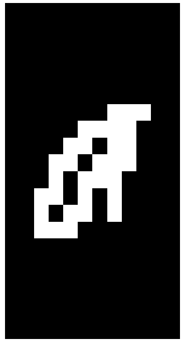


Accuracy vs Training set size (1/3 withheld for test)

What if features are continuous?

- E.g., character recognition: X_i is intensity at ith pixel





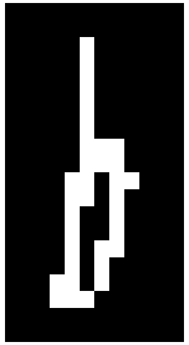
- Gaussian Naïve Bayes (GNB):

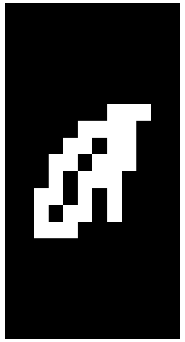
$$P(X_i = x | Y = y_k) = \frac{1}{\sigma_{ik}\sqrt{2\pi}} e^{-\frac{(x-\mu_{ik})^2}{2\sigma_{ik}^2}}$$

distribution of feature X_i is Gaussian with a mean and variance that can depend on the label y_k and which feature X_i it is

What if features are continuous?

- E.g., character recognition: X_i is intensity at ith pixel





- Gaussian Naïve Bayes (GNB):

$$P(X_i = x|Y = y_k) = \frac{1}{\sigma_{ik}\sqrt{2\pi}}\ e^{-\frac{(x-\mu_{ik})^2}{2\sigma_{ik}^2}}$$

- Different mean and variance for each class k and each pixel i.
- Sometimes assume variance:
  - Is independent of Y (i.e., just have σ_i)
  - Or independent of X (i.e., just have σ_k)
  - Or both (i.e., just have σ)

Estimating parameters: Y discrete, X_i continuous

- Maximum likelihood estimates:

$$\boxed{\hat{\mu}_{\mathrm{MLE}} = \frac{1}{N}\sum_{j=1}^{N} x_j}$$

$$\hat{\mu}_{ik} = \frac{1}{\sum_j \delta(Y^j = y_k)} \sum_j X_i^j \delta\left(Y^j = y_k\right)$$

kth class

jth training image

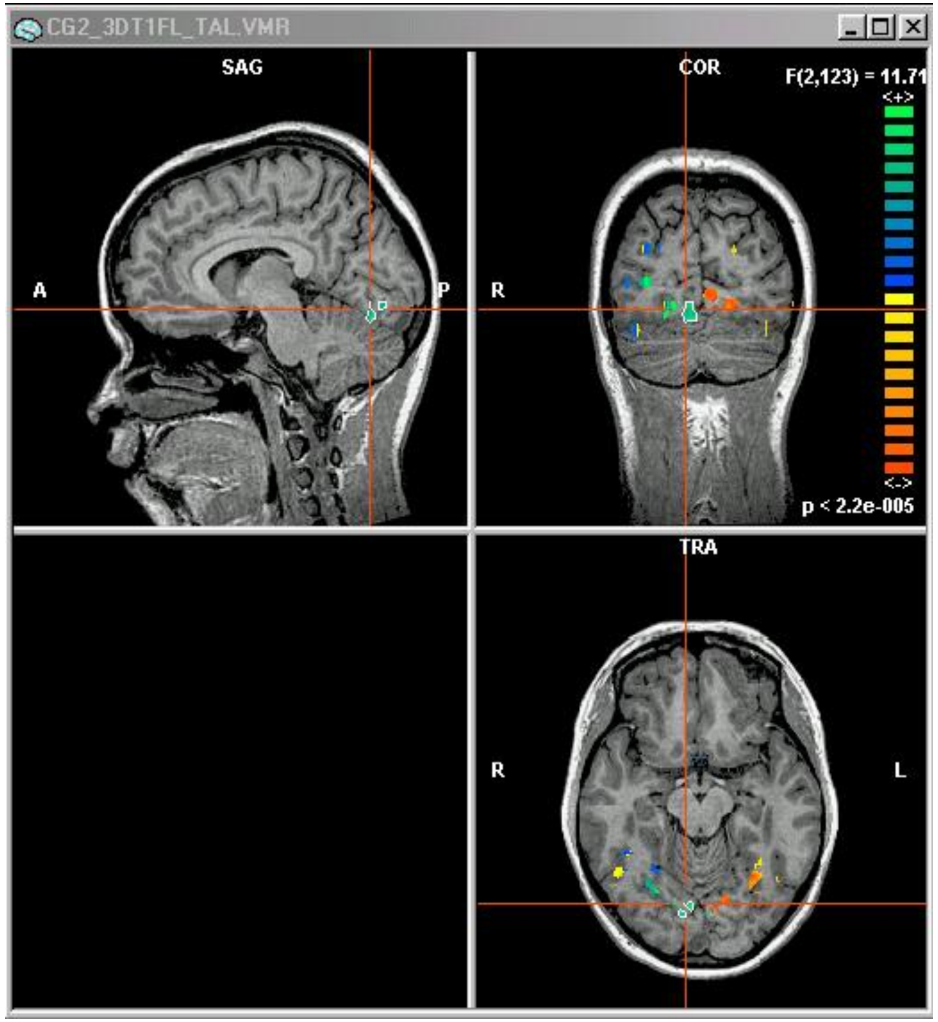
ith pixel in jth training image

$$\boxed{\hat{\sigma}^2_{\mathrm{unbiased}} = \frac{1}{N-1}\sum_{j=1}^{N} \left(x_j - \hat{\mu}\right)^2}$$

$$\hat{\sigma}^2_{ik} = \frac{1}{\sum_j \delta(Y^j = y_k) - 1} \sum_j (X_i^j - \hat{\mu}_{ik})^2 \delta\left(Y^j = y_k\right)$$

Example: GNB for classifying mental states

[Mitchell et al.]



- Classify a person’s cognitive state, based on brain image
  - reading a sentence or viewing a picture?
  - reading the word describing a “Tool” or “Building”?
  - reading the word describing a “Person” or an “Animal”?
- Training: Patients were shown words of different categories and then a measurement was done to see what parts of the brain responded.

Example: GNB for classifying mental states

[Mitchell et al.]

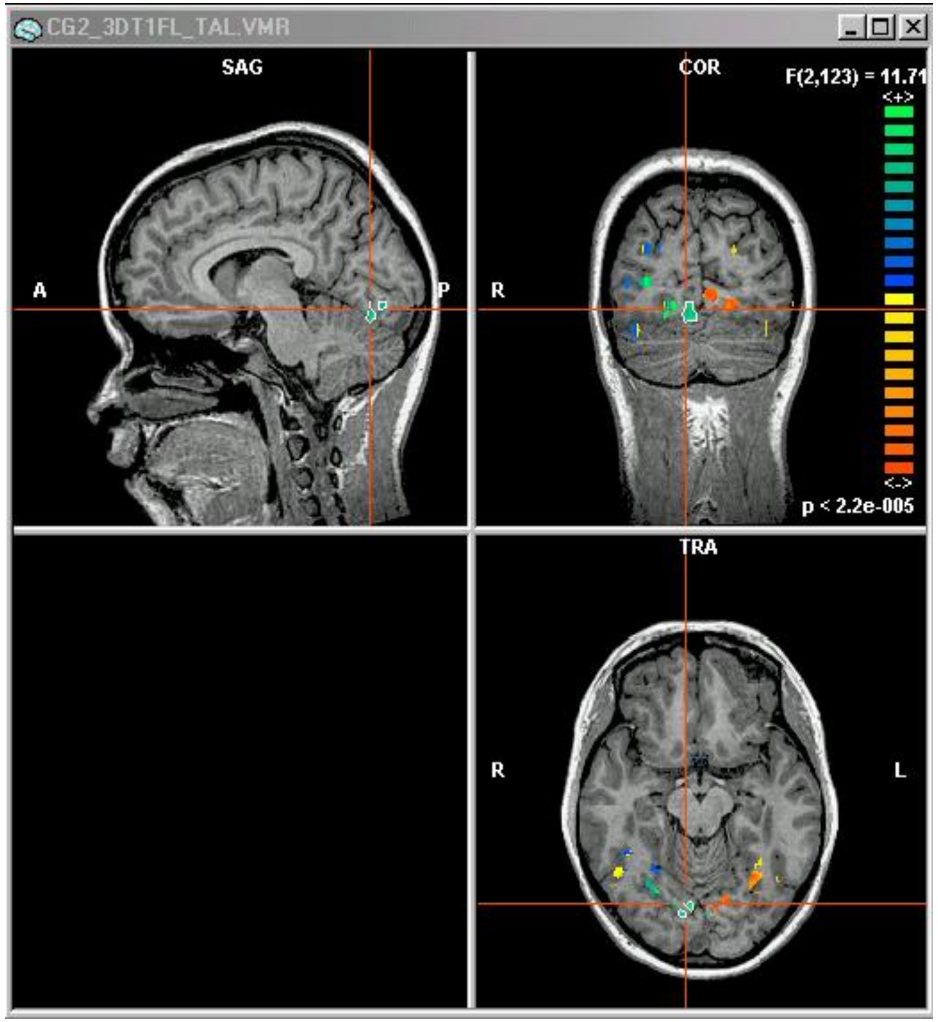
~1mm resolution

~2 images per sec.

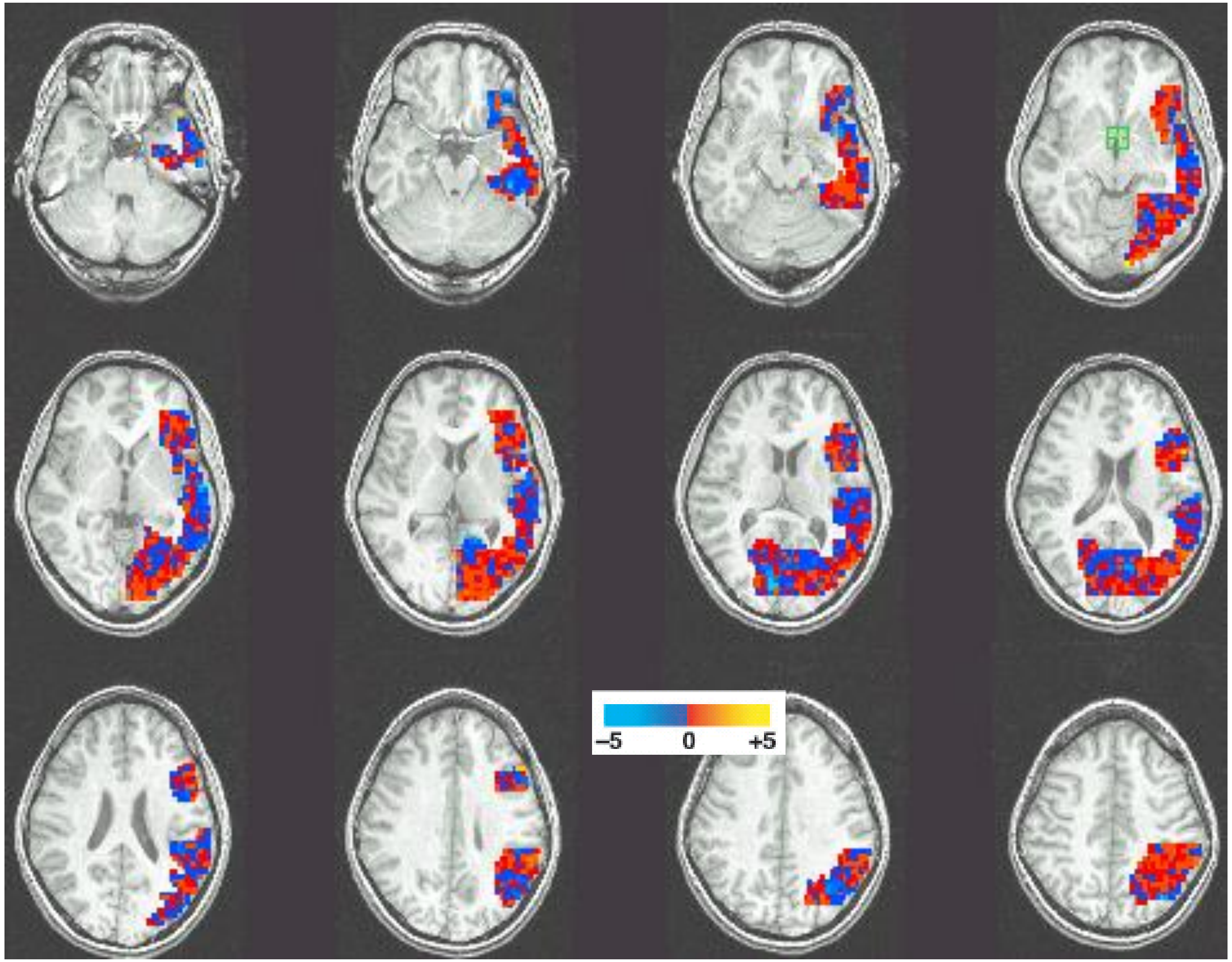
15,000 voxels/image

Non-invasive, save

Measures Blood Oxygen Level Dependent response (BOLD)



Gaussian Naïve Bayes: Learned $\mu_{\text{voxel,word}}$



[Mitchell et al.]

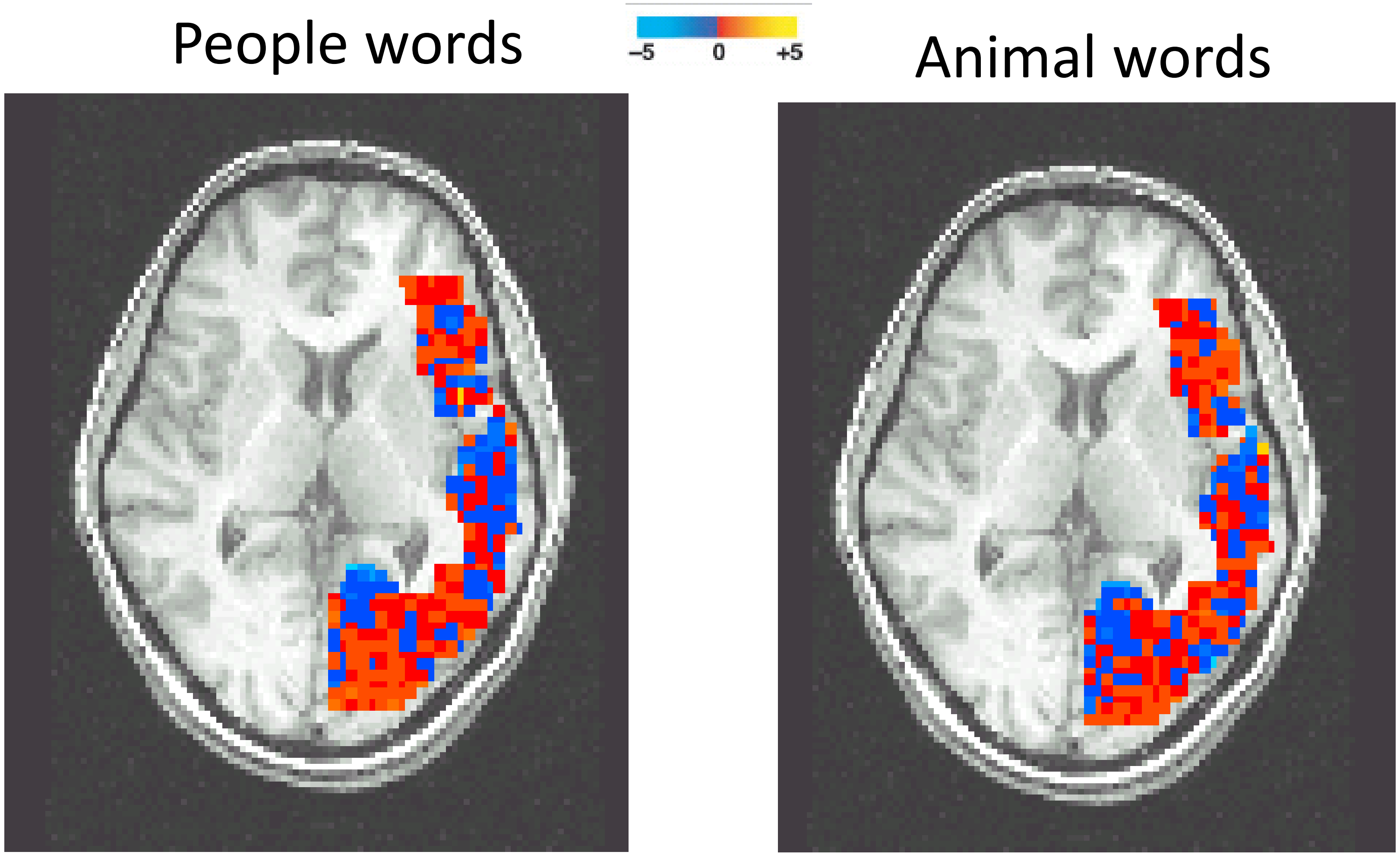
15,000 voxels or features

10 training examples or subjects per class

Learned Naïve Bayes Models –
Means for P(BrainActivity | WordCategory)

[Mitchell et al.]

Pairwise classification accuracy: 85%



People words

Animal words

What you should know

- Naïve Bayes classifier
  - What’s the assumption
  - Why we use it
  - How do we learn it
  - Why is Bayesian estimation important
- Text classification
  - Bag of words model
- Gaussian NB
  - Features are still conditionally independent
  - Each feature has a Gaussian distribution given class