CSE 446
Naïve Bayes

Administrative

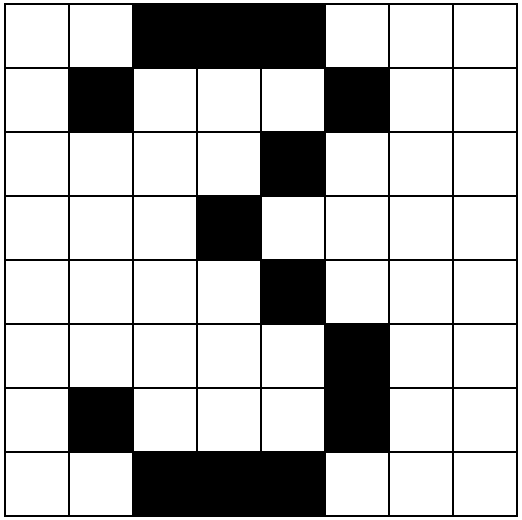
- Homework 1 due this **Friday**
- Homework 2 is out **today**
  - Programming portion out later this week

Naïve Bayes

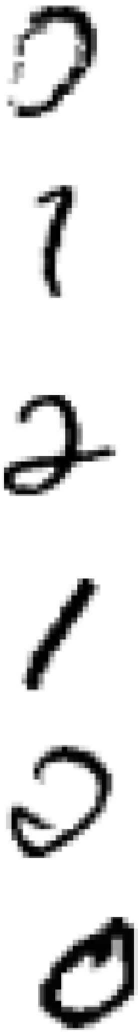
- [see lecture notes]

A Digit Recognizer

- Input: pixel grids



- Output: a digit 0-9



Naïve Bayes for Digits (Binary Inputs)

- Problem setup:
  - One feature F_{ij} for each grid position <i,j>
  - Possible feature values are on / off, based on whether intensity is more or less than 0.5 in underlying image
  - Each input maps to a feature vector, e.g.

$$\rightarrow \langle F_{0,0} = 0 \;\; F_{0,1} = 0 \;\; F_{0,2} = 1 \;\; F_{0,3} = 1 \;\; F_{0,4} = 0 \;\; \ldots F_{15,15} = 0 \rangle$$

  - Here: lots of features, each is binary valued
- Naïve Bayes model:

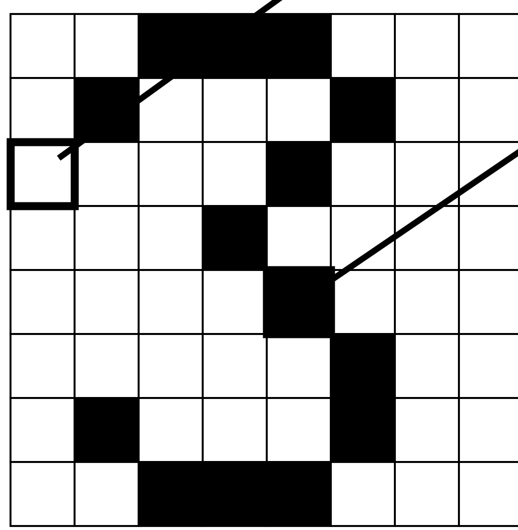
$$P(Y|F_{0,0} \ldots F_{15,15}) \propto P(Y) \prod_{i,j} P(F_{i,j}|Y)$$

- Are the features independent given class?
- What do we need to learn?

Example Distributions

$P(Y)$

1	0.1
2	0.1
3	0.1
4	0.1
5	0.1
6	0.1
7	0.1
8	0.1
9	0.1
0	0.1



$P(F_{3,1} = on|Y)$

1	0.01
2	0.05
3	0.05
4	0.30
5	0.80
6	0.90
7	0.05
8	0.60
9	0.50
0	0.80

$P(F_{5,5} = on|Y)$

1	0.05
2	0.01
3	0.90
4	0.80
5	0.90
6	0.90
7	0.25
8	0.85
9	0.60
0	0.80

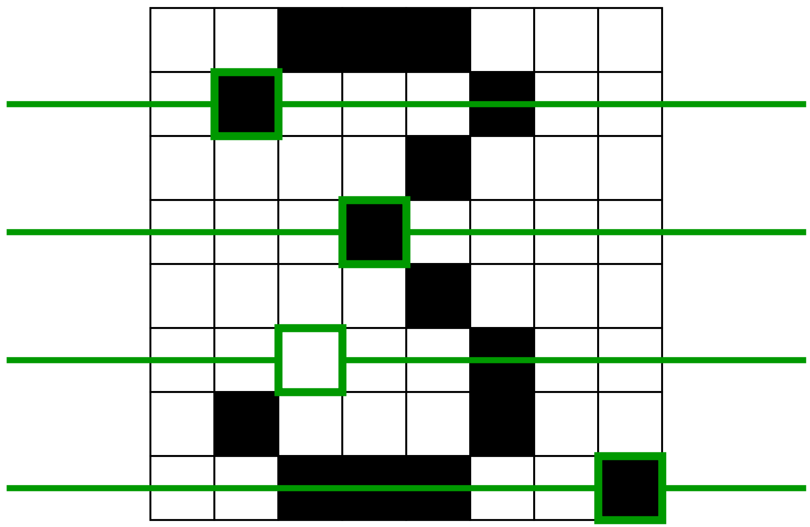
Subtleties of NB Classifier

$P(\text{features}, C = 2)$

$P(\text{features}, C = 3)$

$P(C = 2) = 0.1$

$P(C = 3) = 0.1$



$P(\text{on}|C = 2) = 0.8$

$P(\text{on}|C = 3) = 0.8$

$P(\text{on}|C = 2) = 0.1$

$P(\text{on}|C = 3) = 0.9$

$P(\text{off}|C = 2) = 0.1$

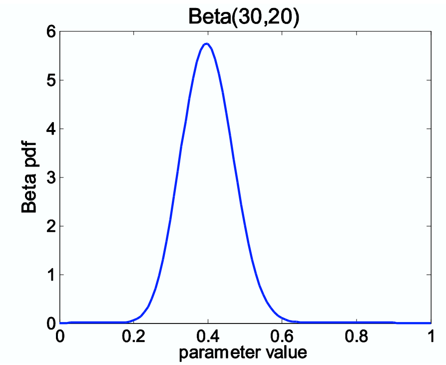
$P(\text{off}|C = 3) = 0.7$

$P(\text{on}|C = 2) = 0.01$

$P(\text{on}|C = 3) = 0.0$

2 wins!!

For Binary Features: We already know the answer!



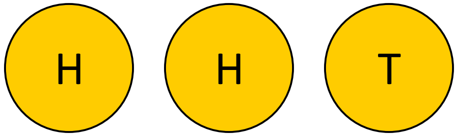
$$P(\theta \mid \mathcal{D}) = \frac{\theta^{\beta_H+\alpha_H-1}(1-\theta)^{\beta_T+\alpha_T-1}}{B(\beta_H+\alpha_H, \beta_T+\alpha_T)} \sim Beta(\beta_H+\alpha_H, \beta_T+\alpha_T)$$

- MAP: use most likely parameter

$$\widehat{\theta} = \arg\max_{\theta} P(\theta \mid \mathcal{D}) = \frac{\alpha_H + \beta_H - 1}{\alpha_H + \beta_H + \alpha_T + \beta_T - 2}$$

- Beta prior equivalent to extra observations for each feature
- As $N \rightarrow \infty$, prior is "forgotten"
- **But, for small sample size, prior is important!**

Multinomials: Laplace Smoothing



- Laplace's estimate:
  - Pretend you saw every outcome k extra times

$$P_{LAP,k}(x) = \frac{c(x) + k}{N + k|X|}$$

$$P_{LAP,0}(X) = \left\langle \frac{2}{3}, \frac{1}{3} \right\rangle$$

$$P_{LAP,1}(X) = \left\langle \frac{3}{5}, \frac{2}{5} \right\rangle$$

$$P_{LAP,100}(X) = \left\langle \frac{102}{203}, \frac{101}{203} \right\rangle$$

  - What's Laplace with k = 0?
  - k is the strength of the prior
  - Can derive this as a MAP estimate for multinomial with *Dirichlet priors*
- Laplace for conditionals:
  - Smooth each condition independently:

$$P_{LAP,k}(x|y) = \frac{c(x, y) + k}{c(y) + k|X|}$$

Text classification

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

- What about the features **X**?
  - The text!

Features **X** are entire document – X_i for i^{th} 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
```

NB for Text classification

- P(**X**|Y) is huge
  - Article at least 1000 words, $\mathbf{X}=\{X_1,\ldots,X_{1000}\}$
  - X_i represents i^{th} word in document, i.e., the domain of X_i is entire vocabulary, e.g., Webster Dictionary (or more), 10,000 words, etc.

- NB assumption helps a lot
  - $P(X_i=x_i|Y=y)$ is just the probability of observing word x_i at position i in a document on topic y

$$h_{NB}(\mathbf{x}) = \arg\max_y P(y) \prod_{i=1}^{LengthDoc} P(x_i|y)$$

Bag of words model

- Typical additional assumption –
  - **Position in document doesn't matter**:
    - $P(X_i=x_i|Y=y) = P(X_j=x_i|Y=y)$ (all position have the same distribution)
  - "Bag of words" model – order of words on the page ignored
  - Sounds really silly, but often works very well!

$$P(y) \prod_{i=1}^{LengthDoc} P(x_i|y)$$

When the lecture is over, remember to wake up the person sitting next to you in the lecture room.

Bag of words model

- Typical additional assumption –
  - **Position in document doesn't matter**:
    - $P(X_i=x_i|Y=y) = P(X_j=x_i|Y=y)$ (all position have the same distribution)
  - "Bag of words" model – order of words on the page ignored
  - Sounds really silly, but often works very well!

$$P(y) \prod_{i=1}^{LengthDoc} P(x_i|y)$$

in is lecture lecture next over person remember room sitting the the the to to up wake when you

Bag of Words Approach

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

NB with Bag of Words for text classification

- Learning phase:
  - Prior P(Y)
    - Count how many documents from each topic (prior)
  - $P(X_i|Y)$
    - For each topic, count how many times you saw word in documents of this topic, divide by total # of words for that topic
- Test phase:
  - For each document
    - Use naïve Bayes decision rule

$$h_{NB}(\mathbf{x}) = \arg\max_y P(y) \prod_{i=1}^{LengthDoc} P(x_i|y)$$

Continuous Features

- [see lecture notes]

What you need to know about Naïve Bayes

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

Bayesian Networks Intro

- [see lecture notes]