Naïve Bayes with Large Vocabularies

Shannon Quinn

(with thanks to William Cohen and Aarti Singh of CMU)

Naïve Bayes: A primer

- Anyone remember how this works?

Classification

Goal: Construct a **predictor** $f : X \to Y$ to minimize a risk (performance measure) $R(f)$

Sports
Science
News

Features, X

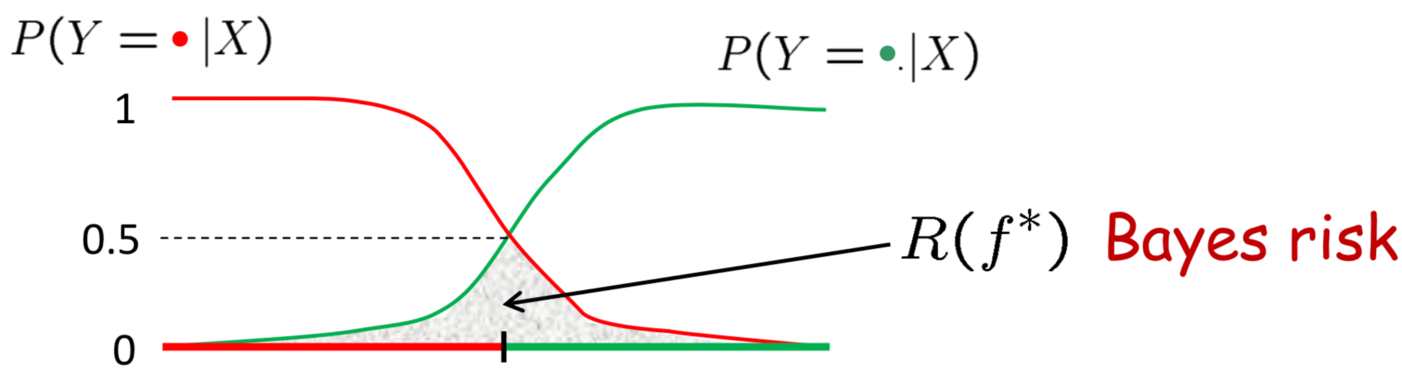
Labels, Y

$$R(f) = P(f(X) \neq Y)$$

Probability of Error

Optimal Classification

Optimal predictor: $f^* = \arg\min_f P(f(X) \neq Y)$
(Bayes classifier)



$$f^*(x) = \arg\max_{Y=y} P(Y = y|X = x)$$

- Even the optimal classifier makes mistakes R(f*) > 0
- Optimal classifier depends on **unknown** distribution P_{XY}

Bayes Rule

- Anyone remember?

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$$

$$P(Y=y|X=x) \;=\; \frac{P(X=x|Y=y)P(Y=y)}{P(X=x)}$$

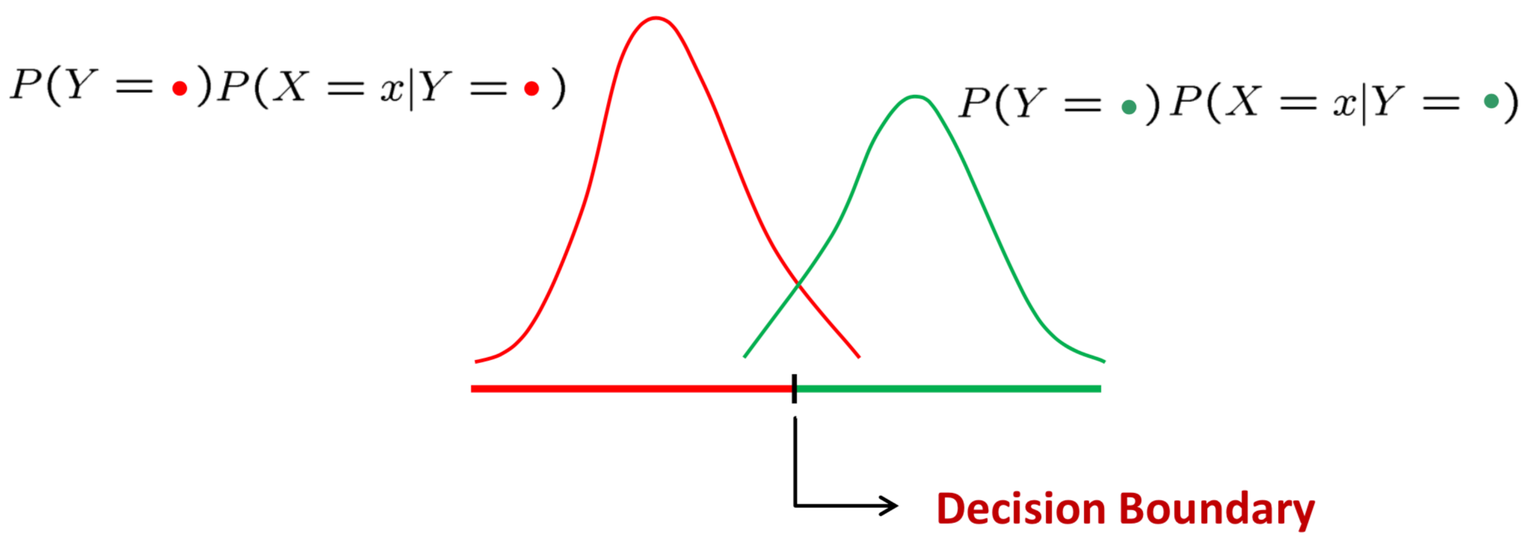
Optimal classifier:

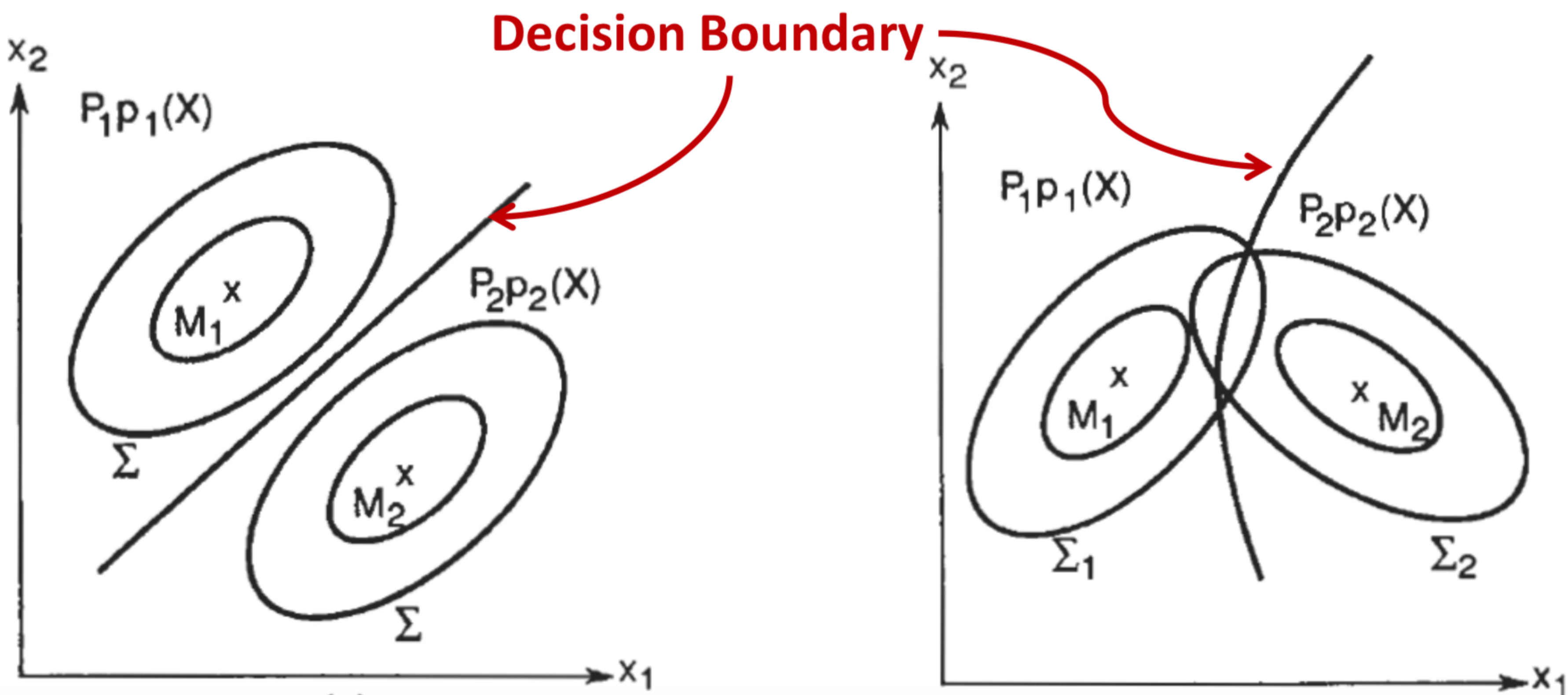
$$\begin{aligned} f^*(x) &= \arg\max_{Y=y} P(Y=y|X=x) \\ &= \arg\max_{Y=y} \underbrace{P(X=x|Y=y)}_{\text{Class conditional density}}\underbrace{P(Y=y)}_{\text{Class prior}} \end{aligned}$$

Decision Boundaries

- Gaussian class conditional densities (1-dimension/feature)

$$P(X = x|Y = y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x-\mu_y)^2}{2\sigma_y^2}\right)$$




Decision Boundary
x_2
$P_1p_1(X)$
M_1
$P_2p_2(X)$
M_2
Σ
Σ
x_1
x_2
$P_1p_1(X)$
$P_2p_2(X)$
M_1
M_2
Σ_1
Σ_2
x_1

Learning the Optimal Classifier

Task: Predict whether or not a picnic spot is enjoyable

Training Data: $X = (X_1 \quad X_2 \quad X_3 \quad \ldots \quad \ldots \quad X_d) \quad Y$

n rows

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

Lets learn P(Y|X) – how many parameters?

Prior: P(Y = y) for all y **K-1 if K labels**

Likelihood: P(X=x|Y = y) for all x,y **$(2^d - 1)$K if d binary features**

Curse of dimensionality

$2^dK – 1$ (K classes, d binary features)

Need $n >> 2^dK – 1$ number of training data to learn all parameters

Conditional Independence

X is **conditionally independent** of Y given Z:

probability distribution governing X is independent of the value of Y, given the value of Z

$$(\forall x, y, z) P(X = x | Y = y, Z = z) = P(X = x | Z = z)$$

Equivalent to:

$$P(X, Y \mid Z) = P(X \mid Z) P(Y \mid Z)$$

e.g., $P(Thunder | Rain, Lightning) = P(Thunder | Lightning)$

Note: does NOT mean Thunder is independent of Rain

Prediction with Conditional Independence

Predict Lightening

From two **conditionally Independent** features

– Thunder

– Rain

parameters needed to learn likelihood given L

$P(T,R|L)$ **$(2^2-1)2 = 6$**

With conditional independence assumption

$P(T,R|L) = P(T|L)\,P(R|L)$ **$(2-1)2 + (2-1)2 = 4$**

Naïve Bayes Assumption

– Features are independent given class:

$$\begin{aligned} P(X_1, X_2|Y) &= P(X_1|X_2, Y)P(X_2|Y) \\ &= P(X_1|Y)P(X_2|Y) \end{aligned}$$

– More generally:

$$P(X_1 ... X_d|Y) = \prod_{i=1}^{d} P(X_i|Y)$$

Naïve Bayes Classifier

Given:

- Class Prior P(Y)
- *d* conditionally independent features **X** given the class Y
- For each X_i, we have likelihood $P(X_i|Y)$

Decision rule:

$$
\begin{aligned}
f_{NB}(\mathbf{x}) &= \arg\max_y \; P(x_1, \ldots, x_d \mid y) P(y) \\
&= \arg\max_y \prod_{i=1}^{d} P(x_i|y) P(y)
\end{aligned}
$$

Naïve Bayes Algorithm

Training Data $\{(X^{(j)}, Y^{(j)})\}_{j=1}^{n}$ $\quad X^{(j)} = (X_1^{(j)}, \ldots, X_d^{(j)})$

Maximum Likelihood Estimates

– For Class Prior

$$\widehat{P}(y) = \frac{\{\#j : Y^{(j)} = y\}}{n}$$

– For Likelihood

$$\frac{\widehat{P}(x_i, y)}{\widehat{P}(y)} = \frac{\{\#j : X_i^{(j)} = x_i, Y^{(j)} = y\}/n}{\{\#j : Y^{(j)} = y\}/n}$$

NB Prediction for test data $\quad X = (x_1, \ldots, x_d)$

$$Y = \arg\max_y \widehat{P}(y) \prod_{i=1}^{d} \frac{\widehat{P}(x_i, y)}{\widehat{P}(y)}$$

SO, IN OUR CASE…

Bag of Words model

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

Naïve Bayes for documents

Learning phase:

– Class Prior P(Y)

– $P(X_i|Y)$

Test phase:

– For each document

- Use naïve Bayes decision rule

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

SCALING TO LARGE VOCABULARIES: WHY

Numbers (Jeff Dean says) Everyone Should Know

L1 cache reference	0.5 ns	
Branch mispredict	5 ns	
L2 cache reference	7 ns	~= 10x
Mutex lock/unlock	100 ns	
Main memory reference	100 ns	~= 15x
Compress 1K bytes with Zippy	10,000 ns	
Send 2K bytes over 1 Gbps network	20,000 ns	
Read 1 MB sequentially from memory	250,000 ns	
Round trip within same datacenter	500,000 ns	
Disk seek	10,000,000 ns	40x
Read 1 MB sequentially from network	10,000,000 ns	
Read 1 MB sequentially from disk	30,000,000 ns	~= 100,000x
Send packet CA->Netherlands->CA	150,000,000 ns	

Large Vocabularies

- How to implement Naïve Bayes
  - Assuming the event counters do *not* fit in memory
- Possible approaches:
  - Use a database? (or at least a key-value store)

Complexity of Naïve Bayes

- You have a *train* dataset and a *test* dataset
- Initialize an "event counter" (hashtable) C
- For each example *id*, y, $x_1, \ldots, x_d$ in *train:*
  - C("Y=ANY") ++;   C("Y=y") ++
  - For j in $1..d$:
    - C("$Y=y$ ^ $X=x_j$") ++
    - C("$Y=y$ ^ $X=ANY$") ++
- For each example *id*, y, $x_1, \ldots, x_d$ in *test:*
  - For each y' in $dom(Y)$:
    - Compute log Pr$(y', x_1, \ldots, x_d)$ =

$$= \left(\sum_j \log \frac{C(X = x_j \wedge Y = y') + mq_x}{C(X = ANY \wedge Y = y') + m} \right) + \log \frac{C(Y = y') + mq_y}{C(Y = ANY) + m}$$

  - Return the best y'

where:
$q_x = 1/|V|$
$q_y = 1/|dom(Y)|$
$m=1$