10-301/10-601 Introduction to Machine Learning

Machine Learning Department
School of Computer Science
Carnegie Mellon University

MLE/MAP
+
Naïve Bayes

Matt Gormley

Lecture 17
Mar. 20, 2023

Reminders

- Lecture 18: this Friday; Recitation on Wednesday
- Homework 6: Learning Theory / Generative Models
  - Out: Fri, Mar. 17
  - Due: Fri, Mar. 24 at 11:59pm
  - IMPORTANT: only 2 grace/late days permitted
- Exam 2 (Thu, Mar 30)
- Exam 3 (Tue, May 2)

MAP ESTIMATION

MLE vs. MAP

Suppose we have data $\mathcal{D} = \{x^{(i)}\}_{i=1}^{N}$

Principle of maximum likelihood estimation (MLE): Choose the parameters that maximize the likelihood of the data.

$$\boldsymbol{\theta}^{\text{MLE}} = \text{argmax}_{\boldsymbol{\theta}}\, p(\mathcal{D}|\boldsymbol{\theta}) = \text{argmax}_{\boldsymbol{\theta}} \prod_{i=1}^{N} p\left(\mathbf{x}^{(i)}|\boldsymbol{\theta}\right)$$

Maximum Likelihood Estimate (MLE)

Principle of maximum *a posteriori* (MAP) Estimation: Choose the parameters that maximize the posterior of the parameters given the data.

$$\boldsymbol{\theta}^{\text{MLE}} = \text{argmax}_{\boldsymbol{\theta}}\, p(\boldsymbol{\theta}|\mathcal{D}) = \text{argmax}_{\boldsymbol{\theta}}\, f(\boldsymbol{\theta}) \prod_{i=1}^{N} p\left(\mathbf{x}^{(i)}|\boldsymbol{\theta}\right)$$

Maximum *a posteriori* (MAP) estimate

MLE vs. MAP

Suppose we have data $\mathcal{D} = \{x^{(i)}\}_{i=1}^{N}$

Principle of maxi **MLE):**

Choose the paran hood of the data.

$$\boldsymbol{\theta}^{\text{MLE}} = \text{argma} \quad \boldsymbol{\theta})$$

Maximum Likelihood Estimate (MLE)

Important!

Usually the parameters are **continuous,** so the prior is a probability **density** function

Principle of maximum *a posteriori* (MAP) Es mation:

Choose the parameters that maximize the osterior of the parameters given the data.

$$\boldsymbol{\theta}^{\text{MLE}} = \text{argmax}_{\boldsymbol{\theta}}\, p(\boldsymbol{\theta}|\mathcal{D}) = \text{argmax}_{\boldsymbol{\theta}}\, \overbrace{f(\boldsymbol{\theta})}^{\text{Prior}} \prod_{i=1}^{N} p(\mathbf{x}^{(i)}|\boldsymbol{\theta})$$

Maximum *a posteriori* (MAP) estimate

The MAP Estimation Objective

MLE: $p(\mathcal{D} \mid \boldsymbol{\theta})$

MAP: $$\overbrace{p(\boldsymbol{\theta} \mid \mathcal{D})}^{\text{posterior}} = \frac{\overbrace{p(\mathcal{D} \mid \boldsymbol{\theta})}^{\text{likelihood}}\overbrace{p(\boldsymbol{\theta})}^{\text{prior}}}{\underbrace{p(\mathcal{D})}_{\substack{\text{not a function of } \theta \\ \int_{\boldsymbol{\theta}'} p(\mathcal{D} \mid \boldsymbol{\theta}')p(\boldsymbol{\theta}')d\boldsymbol{\theta}'}}}$$

Bayes Rule

$$\begin{aligned}
\boldsymbol{\theta}_{MAP} &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, p(\boldsymbol{\theta} \mid \mathcal{D}) \\
&= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \frac{p(\mathcal{D} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{D})} \\
&= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, p(\mathcal{D} \mid \boldsymbol{\theta})p(\boldsymbol{\theta}) \\
&= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \underbrace{\log p(\mathcal{D} \mid \boldsymbol{\theta}) + \log p(\boldsymbol{\theta})}_{\ell_{MAP}(\boldsymbol{\theta})}
\end{aligned}$$

Recipe for Closed-form MLE

1. Assume data was generated iid from some model, i.e., write the *generative story*

   $x^{(i)} \sim p(x|\boldsymbol{\theta})$

2. Write the log-likelihood

   $\ell(\boldsymbol{\theta}) = \log p(x^{(1)}|\boldsymbol{\theta}) + \ldots + \log p(x^{(N)}|\boldsymbol{\theta})$

3. Compute partial derivatives, i.e., the gradient

   $\partial\ell(\boldsymbol{\theta})/\partial\theta_1 = \ldots$

   …

   $\partial\ell(\boldsymbol{\theta})/\partial\theta_M = \ldots$

4. Set derivatives equal to zero and solve for $\boldsymbol{\theta}$

   $\partial\ell(\boldsymbol{\theta})/\partial\theta_m = 0$ for all $m \in \{1, \ldots, M\}$

   $\boldsymbol{\theta}^{MLE}$ = solution to system of M equations and M variables

5. Compute the second derivative and check that $\ell(\boldsymbol{\theta})$ is concave down at $\boldsymbol{\theta}^{MLE}$

Recipe for Closed-form MAP

1. Assume data was generated iid from some model, i.e., write the *generative story*

   $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$ and then for all i: $x^{(i)} \sim p(x|\boldsymbol{\theta})$

2. Write the log posterior

   $\ell_{MAP}(\boldsymbol{\theta}) = \log p(\boldsymbol{\theta}) + \log p(x^{(1)}|\boldsymbol{\theta}) + \ldots + \log p(x^{(N)}|\boldsymbol{\theta})$

3. Compute partial derivatives, i.e., the gradient

   $\partial \ell_{MAP}(\boldsymbol{\theta})/\partial \theta_1 = \ldots$

   $\ldots$

   $\partial \ell_{MAP}(\boldsymbol{\theta})/\partial \theta_M = \ldots$

4. Set derivatives to equal zero and solve for $\boldsymbol{\theta}$

   $\partial \ell_{MAP}(\boldsymbol{\theta})/\partial \theta_m = 0$ for all $m \in \{1, \ldots, M\}$

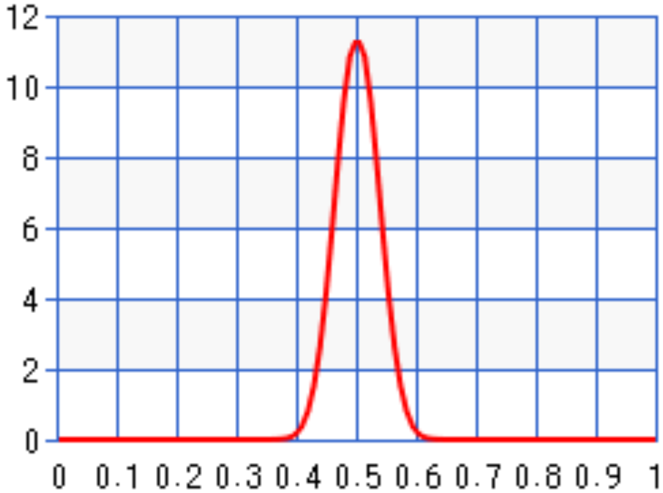
   $\boldsymbol{\theta}^{MAP}$ = solution to system of M equations and M variables

5. Compute the second derivative and check that $\ell(\boldsymbol{\theta})$ is concave down at $\boldsymbol{\theta}^{MAP}$

The Prior Distribution

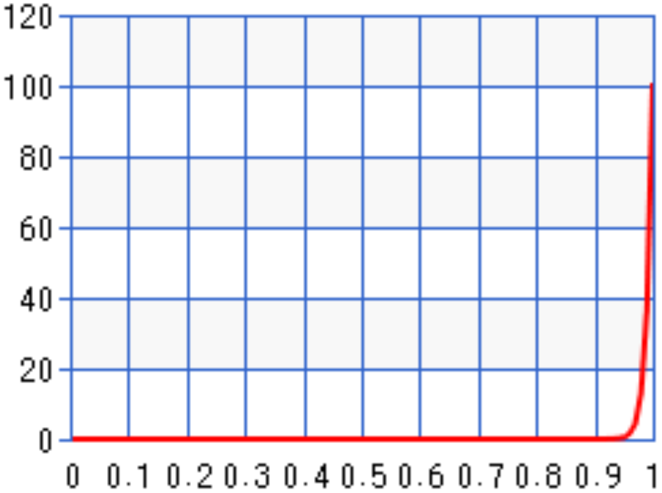
- The prior distribution encodes domain knowledge about the problem.
- Question: Why do we use the Beta distribution as the prior for the Bernoulli?
- **Reason #1:** It has the right support, i.e. [0,1].

Example: Beta prior "fair coin"



$$f(\phi \mid \alpha = 101, \beta = 101)$$

Example: Beta prior "unfair coin"



$$f(\phi \mid \alpha = 101, \beta = 1)$$

The Prior Distribution

- The prior distribution encodes domain knowledge about the problem.
- Question: Why do we use the Beta distribution as the prior for the Bernoulli?
- **Reason #2:** The Beta is a conjugate prior for the Bernoulli.
- **Definition:** A distribution is the **conjugate prior** of a likelihood if the form of the posterior is the same as the form of the prior.

Posterior p(θ | D)	Likelihood p(D | θ)	Prior p(θ)	Conjugate?
Beta	Bernoulli	Beta	yes
Dirichlet	Multinomial	Multinomial	yes
Gaussian	Guassian	Guassian	yes
Gamma	Exponential	Gamma	yes
??	Multinomial	Logistic Normal	no

MLE of Bernoulli Model

$$N_1 = \#(x^{(i)} = 1)$$
$$N_0 = \#(x^{(i)} = 0)$$

1. Model: $\mathbf{x}^{(i)} \sim \text{Bernoulli}(\phi)$ for $i = 1, \ldots, N$
2. Log-posterior:

$$\begin{aligned} \ell_{\text{MLE}}(\phi) &= \log p(\mathcal{D} \mid \phi) \\ &= \log \left(\phi^{N_1} (1-\phi)^{N_0} \right) \\ &= N_1 \log(\phi) + N_0 \log(1-\phi) \end{aligned}$$

3. Derivative: $\dfrac{\partial \ell_{\text{MLE}}(\phi)}{\partial \phi} = \dfrac{N_1}{\phi} - \dfrac{N_0}{1-\phi}$
4. Set to zero and solve: $\phi_{\text{MLE}} = \dfrac{N_1}{N_1 + N_0} = \dfrac{N_1}{N}$

MAP of Beta-Bernoulli Model

1. Model: $\phi \sim$ Beta(α, β)

   $\mathbf{x}^{(i)} \sim$ Bernoulli(ϕ) for $i = 1, \ldots, N$

MAP of Beta-Bernoulli Model

1. Model: $\phi \sim \text{Beta}(\alpha, \beta)$

   $\mathbf{x}^{(i)} \sim \text{Bernoulli}(\phi)$ for $i = 1, \ldots, N$

$$N_1 = \#(x^{(i)} = 1)$$
$$N_0 = \#(x^{(i)} = 0)$$

2. Log-posterior:

$$\begin{aligned}
\ell_{\text{MAP}}(\phi) &= \log\left[p(\mathcal{D} \mid \phi) f(\phi \mid \alpha, \beta)\right] \\
&= \log\left[\left(\phi^{N_1}(1-\phi)^{N_0}\right)\left(\frac{1}{B(\alpha,\beta)}\phi^{(\alpha-1)}(1-\phi)^{(\beta-1)}\right)\right] \\
&= \log\left[\phi^{(N_1+\alpha-1)}(1-\phi)^{(N_0+\beta-1)}\frac{1}{B(\alpha,\beta)}\right] \\
&= (N_1+\alpha-1)\log(\phi) + (N_0+\beta-1)\log(1-\phi) - \log B(\alpha,\beta) \\
&= N_1' \log(\phi) + N_0' \log(1-\phi) - \log B(\alpha,\beta)
\end{aligned}$$

$$N_1' = N_1 + \alpha - 1$$
$$N_0' = N_0 + \beta - 1$$

3. Derivative: $\dfrac{\partial \ell_{\text{MAP}}(\phi)}{\partial \phi} = \dfrac{N_1'}{\phi} - \dfrac{N_0'}{1-\phi}$

4. Set to zero and solve: $\phi_{\text{MAP}} = \dfrac{N_1'}{N_1' + N_0'} = \dfrac{N_1 + \alpha - 1}{N_1 + \alpha - 1 + N_0 + \beta - 1}$

MAP of Beta-Bernoulli Model

Example 1 (MLE) Suppose $D = \{8H, 2T\}$

$$\phi_{\text{MLE}} = \frac{8}{10} = 0.8$$

Example 2 (MAP) Same dataset, but $\phi \sim \text{Beta}(\alpha = 101, \beta = 101)$

$$\phi_{\text{MAP}} = \frac{8 + 101 - 1}{8 + 101 - 1 + 2 + 101 - 1} = \frac{108}{108 + 102} \approx 0.5$$

"fair coin" prior

Example 3 (MAP) Same dataset, but $\phi \sim \text{Beta}(\alpha = 101, \beta = 1)$

$$\phi_{\text{MAP}} = \frac{108}{108 + 2} \approx 1.0$$

"unfair coin" prior

Example 4 (MLE) Suppose $D = \{108H, 102T\}$

$$\phi_{\text{MLE}} = \frac{108}{108 + 102} \approx 0.5$$

Takeaways

- One view of what ML is trying to accomplish is **function approximation**
- The principle of **maximum likelihood estimation** provides an alternate view of learning

- **Synthetic data** can help **debug** ML algorithms
- Probability distributions can be used to **model** real data that occurs in the world
  (don't worry we'll make our distributions more interesting soon!)

Learning Objectives

MLE / MAP

You should be able to…

1. Recall probability basics, including but not limited to: discrete and continuous random variables, probability mass functions, probability density functions, events vs. random variables, expectation and variance, joint probability distributions, marginal probabilities, conditional probabilities, independence, conditional independence
2. Describe common probability distributions such as the Beta, Dirichlet, Multinomial, Categorical, Gaussian, Exponential, etc.
3. State the principle of maximum likelihood estimation and explain what it tries to accomplish
4. State the principle of maximum a posteriori estimation and explain why we use it
5. Derive the MLE or MAP parameters of a simple model in closed form

NAÏVE BAYES

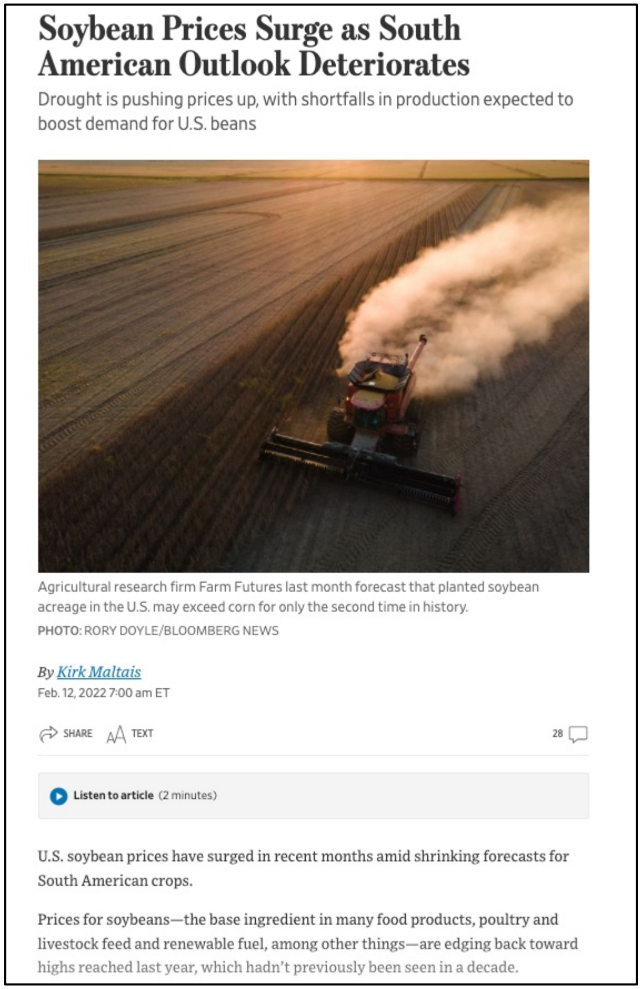
Naïve Bayes

- Why are we talking about Naïve Bayes?
  - It's **just another decision function** that fits into our "big picture" recipe from last time
  - But it's our first **example of a Bayesian Network** and provides a *clearer* picture of **probabilistic learning**
  - Just like the other Bayes Nets we'll see, it **admits a closed form solution** for MLE and MAP
  - So learning is **extremely efficient** (just counting)

Fake News Detector

Today's Goal: To define a generative model of emails of two different classes (e.g. real vs. fake news)

The Economist



Soybean Prices Surge as South American Outlook Deteriorates

Drought is pushing prices up, with shortfalls in production expected to boost demand for U.S. beans

Agricultural research firm Farm Futures last month forecast that planted soybean acreage in the U.S. may exceed corn for only the second time in history.

PHOTO: RORY DOYLE/BLOOMBERG NEWS

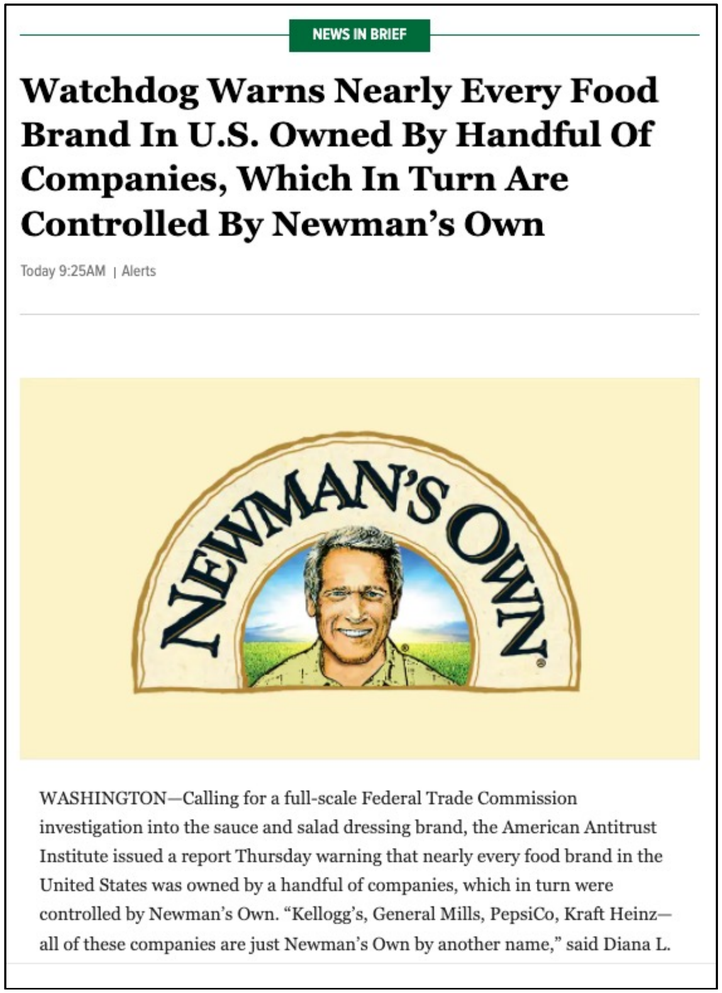
By Kirk Maltais

Feb. 12, 2022 7:00 am ET

U.S. soybean prices have surged in recent months amid shrinking forecasts for South American crops.

Prices for soybeans—the base ingredient in many food products, poultry and livestock feed and renewable fuel, among other things—are edging back toward highs reached last year, which hadn't previously been seen in a decade.

The Onion



NEWS IN BRIEF

Watchdog Warns Nearly Every Food Brand In U.S. Owned By Handful Of Companies, Which In Turn Are Controlled By Newman's Own

Today 9:25AM | Alerts

WASHINGTON—Calling for a full-scale Federal Trade Commission investigation into the sauce and salad dressing brand, the American Antitrust Institute issued a report Thursday warning that nearly every food brand in the United States was owned by a handful of companies, which in turn were controlled by Newman's Own. "Kellogg's, General Mills, PepsiCo, Kraft Heinz—all of these companies are just Newman's Own by another name," said Diana L.

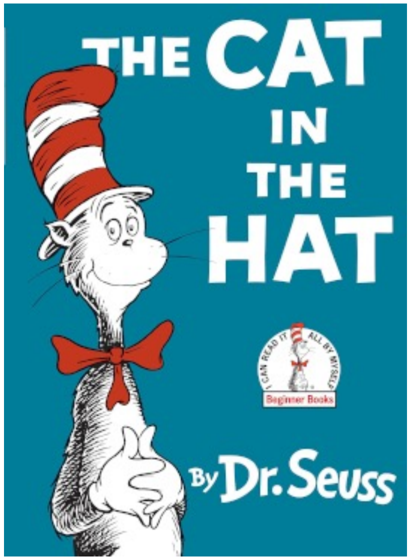
Bag-of-Words Model

x_1 ("hat")	x_2 ("cat")	x_3 ("dog")	x_4 ("fish")	x_5 ("mom")	x_6 ("dad")	y (Dr. Seuss)

Bag-of-Words Model

x_1 (“hat”)	x_2 (“cat”)	x_3 (“dog”)	x_4 (“fish”)	x_5 (“mom”)	x_6 (“dad”)	y (Dr. Seuss)
1	1	0	0	0	0	1

The Cat in the Hat
(by Dr. Seuss)

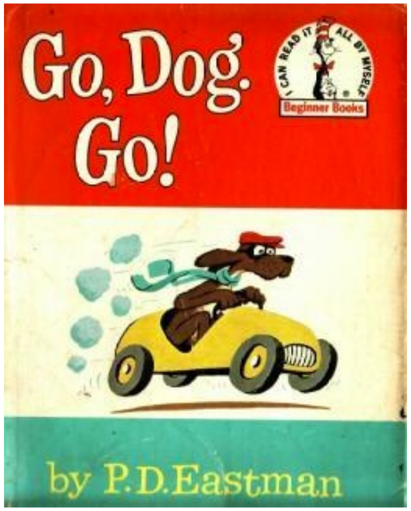


Source: https://en.wikipedia.org/wiki/The_Cat_in_the_Hat#/media/File:The_Cat_in_the_Hat.png

Bag-of-Words Model

x_1 (“hat”)	x_2 (“cat”)	x_3 (“dog”)	x_4 (“fish”)	x_5 (“mom”)	x_6 (“dad”)	y (Dr. Seuss)
1	1	0	0	0	0	1
0	0	1	0	0	0	0

Go, Dog. Go!
(by P. D. Eastman)

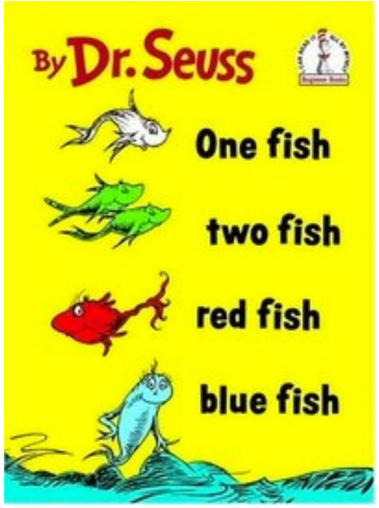


Source: https://en.wikipedia.org/wiki/Go,_Dog._Go!#/media/File:Go_Dog_Go.jpg

Bag-of-Words Model

x_1 ("hat")	x_2 ("cat")	x_3 ("dog")	x_4 ("fish")	x_5 ("mom")	x_6 ("dad")	y (Dr. Seuss)
1	1	0	0	0	0	1
0	0	1	0	0	0	0
0	0	0	1	0	0	1

One Fish, Two Fish, Red Fish, Blue Fish (by Dr. Seuss)

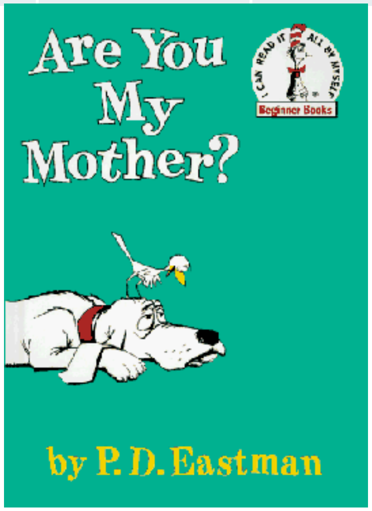


Source: https://en.wikipedia.org/wiki/One_Fish,_Two_Fish,_Red_Fish,_Blue_Fish#/media/File:One_Fish_Two_Fish_Red_Fish_Blue_Fish_(cover_art).jpg

Bag-of-Words Model

x_1 ("hat")	x_2 ("cat")	x_3 ("dog")	x_4 ("fish")	x_5 ("mom")	x_6 ("dad")	y (Dr. Seuss)
1	1	0	0	0	0	1
0	0	1	0	0	0	0
0	0	0	1	0	0	1
0	0	0	0	1	0	0

Are You My Mother?
(by P. D. Eastman)



Source: https://en.wikipedia.org/wiki/Are_You_My_Mother%3F#/media/File:Areyoumymother.gif

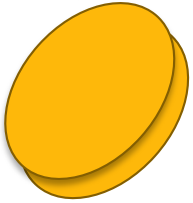
Naive Bayes: Model

Whiteboard

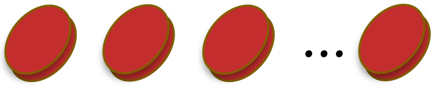
- Generating synthetic "labeled documents"
- Definition of model
- Naive Bayes assumption
- Counting # of parameters with / without NB assumption

Model 1: Bernoulli Naïve Bayes

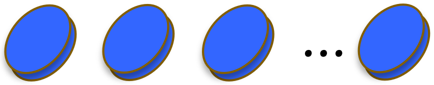
Flip weighted coin



If HEADS, flip
each red coin



If TAILS, flip
each blue coin



y	x_1	x_2	x_3	…	x_M
0	1	0	1	…	1
1	0	1	0	…	1
1	1	1	1	…	1
0	0	0	1	…	1
0	1	0	1	…	0
1	1	0	1	…	0

We can **generate** data in this fashion. Though in practice we never would since our data is **given.**

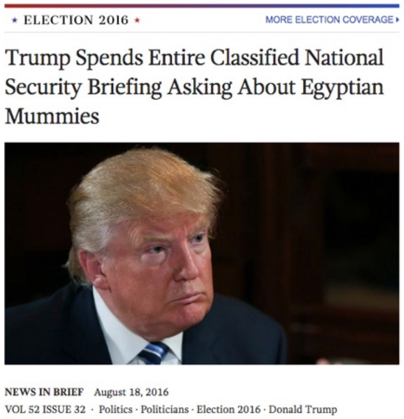
Instead, this provides an explanation of **how** the data was generated (albeit a terrible one).

Each red coin corresponds to an x_m

What's wrong with the Naïve Bayes Assumption?

The features might not be independent!!

- Example 1:
  - If a document contains the word "Donald", it's extremely likely to contain the word "Trump"
  - These are not independent!



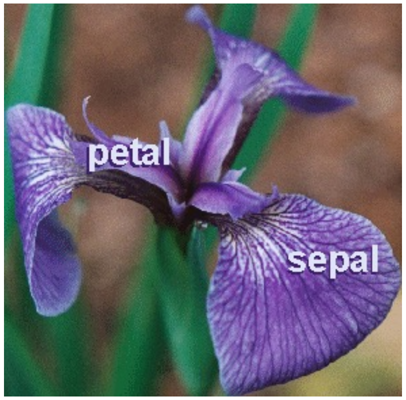
⋆ ELECTION 2016 ⋆ MORE ELECTION COVERAGE ▸

Trump Spends Entire Classified National Security Briefing Asking About Egyptian Mummies

NEWS IN BRIEF August 18, 2016

VOL 52 ISSUE 32 · Politics · Politicians · Election 2016 · Donald Trump

- Example 2:
  - If the petal width is very high, the petal length is also likely to be very high



Q&A

Q: Why would we use Naïve Bayes? Isn’t it too Naïve?

A: Naïve Bayes has one **key advantage** over methods like Perceptron, Logistic Regression, Neural Nets:

Training is lightning fast!

While other methods require slow iterative training procedures that might require hundreds of epochs, Naïve Bayes computes its parameters in closed form by counting.

Naïve Bayes: Learning from Data

Whiteboard

- Data likelihood
- MLE for Naive Bayes
- Example: MLE for Naïve Bayes with Two Features
- MAP for Naive Bayes

Recipe for Closed-form MLE

1. Assume data was generated iid from some model, i.e., write the *generative story*

   $x^{(i)} \sim p(x|\boldsymbol{\theta})$

2. Write the log-likelihood

   $\ell(\boldsymbol{\theta}) = \log p(x^{(1)}|\boldsymbol{\theta}) + \ldots + \log p(x^{(N)}|\boldsymbol{\theta})$

3. Compute partial derivatives, i.e., the gradient

   $\partial\ell(\boldsymbol{\theta})/\partial\theta_1 = \ldots$

   $\ldots$

   $\partial\ell(\boldsymbol{\theta})/\partial\theta_M = \ldots$

4. Set derivatives equal to zero and solve for $\boldsymbol{\theta}$

   $\partial\ell(\boldsymbol{\theta})/\partial\theta_m = 0$ for all $m \in \{1, \ldots, M\}$

   $\boldsymbol{\theta}^{MLE}$ = solution to system of M equations and M variables

5. Compute the second derivative and check that $\ell(\boldsymbol{\theta})$ is concave down at $\boldsymbol{\theta}^{MLE}$

BERNOULLI NAÏVE BAYES

Model 1: Bernoulli Naïve Bayes

Data: Binary feature vectors, Binary labels

$$\mathbf{x} \in \{0,1\}^M \qquad y \in \{0,1\}$$

Generative Story:

$$y \sim \text{Bernoulli}(\phi)$$
$$x_1 \sim \text{Bernoulli}(\theta_{y,1})$$
$$x_2 \sim \text{Bernoulli}(\theta_{y,2})$$
$$\vdots$$
$$x_M \sim \text{Bernoulli}(\theta_{y,M})$$

Model:

$$\begin{aligned} p_{\phi,\boldsymbol{\theta}}(\boldsymbol{x}, y) &= p_{\phi,\boldsymbol{\theta}}(x_1, \ldots, x_M, y) \\ &= p_\phi(y) \prod_{m=1}^{M} p_{\boldsymbol{\theta}}(x_m|y) \\ &= \Big[(\phi)^y (1-\phi)^{(1-y)} \\ &\quad \prod_{m=1}^{M} (\theta_{y,m})^{x_m} (1-\theta_{y,m})^{(1-x_m)}\Big] \end{aligned}$$

Model 1: Bernoulli Naïve Bayes

Maximum Likelihood Estimation

Training: Find the **class-conditional** MLE parameters

Count Variables:

$$N_{y=1} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 1)$$

$$N_{y=0} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0)$$

$$N_{y=0,x_m=1} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0 \wedge x_m^{(i)} = 1)$$

$$\ldots$$

Maximum Likelihood Estimators:

$$\phi = \frac{N_{y=1}}{N}$$

$$\theta_{0,m} = \frac{N_{y=0,x_m=1}}{N_{y=0}}$$

$$\theta_{1,m} = \frac{N_{y=1,x_m=1}}{N_{y=1}}$$

$$\forall m \in \{1, \ldots, M\}$$

Model 1: Bernoulli Naïve Bayes

Maximum Likelihood Estimation

Training: Find the **class-conditional** MLE parameters

Count Variables:

$$N_{y=1} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 1)$$

$$N_{y=0} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0)$$

$$N_{y=0,x_m=1} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0 \wedge x_m^{(i)} = 1)$$

$$\ldots$$

Maximum Likelihood Estimators:

$$\phi = \frac{N_{y=1}}{N}$$

$$\theta_{0,m} = \frac{N_{y=0,x_m=1}}{N_{y=0}}$$

$$\theta_{1,m} = \frac{N_{y=1,x_m=1}}{N_{y=1}}$$

$$\forall m \in \{1, \ldots, M\}$$

Data:

y	x_1	x_2	x_3	…	x_M
0	1	0	1	…	1
1	0	1	0	…	1
1	0	1	1	…	1
0	0	0	1	…	1
0	1	0	1	…	0
1	1	0	1	…	0

Question 1:

What is the MLE of ϕ?

(A) 0/6 (B) 1/6 (C) 2/6 (D) 3/6
(E) 4/6 (F) 5/6 (G) 6/6 (H) None of the above

Model 1: Bernoulli Naïve Bayes

Maximum Likelihood Estimation

Training: Find the **class-conditional** MLE parameters

Count Variables:

$$N_{y=1} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 1)$$

$$N_{y=0} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0)$$

$$N_{y=0,x_m=1} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0 \wedge x_m^{(i)} = 1)$$

$$\ldots$$

Maximum Likelihood Estimators:

$$\phi = \frac{N_{y=1}}{N}$$

$$\theta_{0,m} = \frac{N_{y=0,x_m=1}}{N_{y=0}}$$

$$\theta_{1,m} = \frac{N_{y=1,x_m=1}}{N_{y=1}}$$

$$\forall m \in \{1, \ldots, M\}$$

Data:

y	x_1	x_2	x_3	…	x_M
0	1	0	1	…	1
1	0	1	0	…	1
1	0	1	1	…	1
0	0	0	1	…	1
0	1	0	1	…	0
1	1	0	1	…	0

Question 2:

What is the MLE of $\theta_{0,1}$?

(A) 0/6 (B) 1/6 (C) 2/6 (D) 3/6

(E) 4/6 (F) 5/6 (G) 6/6 (H) None of the above

Model 1: Bernoulli Naïve Bayes

Maximum Likelihood Estimation

Training: Find the **class-conditional** MLE parameters

Count Variables:

$$N_{y=1} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 1)$$

$$N_{y=0} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0)$$

$$N_{y=0,x_m=1} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0 \wedge x_m^{(i)} = 1)$$

$$\ldots$$

Maximum Likelihood Estimators:

$$\phi = \frac{N_{y=1}}{N}$$

$$\theta_{0,m} = \frac{N_{y=0,x_m=1}}{N_{y=0}}$$

$$\theta_{1,m} = \frac{N_{y=1,x_m=1}}{N_{y=1}}$$

$$\forall m \in \{1, \ldots, M\}$$

MLE for Naïve Bayes is a splendid learning algorithm for when you have say billions of training examples and hundreds of millions of features!

You only need one pass through the data to perform some counting.

MAP ESTIMATION FOR BERNOULLI NAÏVE BAYES

MLE

What does maximizing likelihood accomplish?

- There is only a finite amount of probability mass (i.e. sum-to-one constraint)
- MLE tries to allocate **as much** probability mass **as possible** to the things we have observed…

  … **at the expense** of the things we have **not** observed

A Shortcoming of MLE

For Naïve Bayes, suppose we **never** observe the word "`unicorn`" in a real news article.

In this case, what is the MLE of the following quantity?

p(x$_{\texttt{unicorn}}$ | y=real) =

Recall: $$\theta_{k,0} = \frac{\sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0 \wedge x_k^{(i)} = 1)}{\sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0)}$$

Now suppose we observe the word "`unicorn`" at test time. What is the posterior probability that the article was a real article?

$$p(y = real|\mathbf{x}) = \frac{p(\mathbf{x}|y = real)p(y = real)}{p(\mathbf{x})}$$

Recipe for Closed-form MAP Estimation

1. Assume data was generated i.i.d. from some model (i.e. write the generative story)

   $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$ and then for all i: $x^{(i)} \sim p(x|\boldsymbol{\theta})$

2. Write log-likelihood

   $\ell_{MAP}(\boldsymbol{\theta}) = \log p(\boldsymbol{\theta}) + \log p(x^{(1)}|\boldsymbol{\theta}) + \ldots + \log p(x^{(N)}|\boldsymbol{\theta})$

3. Compute partial derivatives (i.e. gradient)

   $\partial \ell_{MAP}(\boldsymbol{\theta})/\partial \theta_1 = \ldots$

   $\partial \ell_{MAP}(\boldsymbol{\theta})/\partial \theta_2 = \ldots$

   …

   $\partial \ell_{MAP}(\boldsymbol{\theta})/\partial \theta_M = \ldots$

4. Set derivatives to zero and solve for $\boldsymbol{\theta}$

   $\partial \ell_{MAP}(\boldsymbol{\theta})/\partial \theta_m = 0$ for all $m \in \{1, \ldots, M\}$

   $\boldsymbol{\theta}^{MAP}$ = solution to system of M equations and M variables

5. Compute the second derivative and check that $\ell(\boldsymbol{\theta})$ is concave down at $\boldsymbol{\theta}^{MAP}$

Model 1: Bernoulli Naïve Bayes

MAP Estimation (Beta Prior)

1. Generative Story:

The parameters are drawn once for the entire dataset.

```
$\phi \sim$ Beta$(\alpha', \beta')$
**for** $m \in \{1, \ldots, M\}$**:**
    **for** $y \in \{0, 1\}$**:**
        $\theta_{m,y} \sim$ Beta$(\alpha, \beta)$
**for** $i \in \{1, \ldots, N\}$**:**
    $y^{(i)} \sim$ Bernoulli$(\phi)$
    **for** $m \in \{1, \ldots, M\}$**:**
        $x_m^{(i)} \sim$ Bernoulli$(\theta_{y^{(i)},m})$
```

$$N_{y=1} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 1)$$

$$N_{y=0} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0)$$

$$N_{y=0,x_m=1} = \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0 \wedge x_m^{(i)} = 1)$$

$$\ldots$$

2. Likelihood:

$$\ell_{MAP}(\phi, \boldsymbol{\theta})$$

$$= \log \left[p(\phi, \boldsymbol{\theta} | \alpha', \beta', \alpha, \beta) p(\mathcal{D} | \phi, \boldsymbol{\theta}) \right]$$

$$= \log \left[\left(p(\phi | \alpha', \beta') \prod_{m=1}^{M} p(\theta_{0,m} | \alpha, \beta) \right) \left(\prod_{i=1}^{N} p(\mathbf{x}^{(i)}, y^{(i)} | \phi, \boldsymbol{\theta}) \right) \right]$$

3. MAP Estimators: $(\phi^{MAP}, \boldsymbol{\theta}^{MAP}) = \underset{\phi, \boldsymbol{\theta}}{\operatorname{argmax}}\, \ell_{MAP}(\phi, \boldsymbol{\theta})$

Take derivatives, set to zero and solve…

$$\phi = \frac{(\alpha' - 1) + N_{y=1}}{(\alpha' - 1) + (\beta' - 1) + N}$$

$$\theta_{0,m} = \frac{(\alpha - 1) + N_{y=0,x_m=1}}{(\alpha - 1) + (\beta - 1) + N_{y=0}}$$

$$\theta_{1,m} = \frac{(\alpha - 1) + N_{y=1,x_m=1}}{(\alpha - 1) + (\beta - 1) + N_{y=1}}$$

$$\forall m \in \{1, \ldots, M\}$$

Model 1: Bernoulli Naïve Bayes

MAP Estimation (Beta Prior)

1. Generative Story:

The parameters are drawn once for the entire dataset.

```
$\phi \sim$ Beta$(\alpha', \beta')$
for $m \in \{1, \ldots, M\}$:
    for $y \in \{0, 1\}$:
        $\theta_{m,y} \sim$ Beta$(\alpha, \beta)$
for $i \in \{1, \ldots, N\}$:
    $y^{(i)} \sim$ Bernoulli$(\phi)$
```

2. Likelihood:

$$\ell_{MAP}(\phi, \boldsymbol{\theta})$$

$$= \log\left[p(\phi, \boldsymbol{\theta} | \alpha', \beta', \alpha, \beta) p(\mathcal{D} | \phi, \boldsymbol{\theta})\right]$$

$$= \log\left[\left(p(\phi|\alpha', \beta') \prod_{m=1}^{M} p(\theta_{0,m}|\alpha, \beta)\right)\left(\prod_{i=1}^{N} p(\mathbf{x}^{(i)}, y^{(i)}|\phi, \boldsymbol{\theta})\right)\right]$$

3. MAP Estimators: $(\phi^{MAP}, \boldsymbol{\theta}^{MAP}) = \underset{\phi, \boldsymbol{\theta}}{\text{argmax}}\, \ell_{MAP}(\phi, \boldsymbol{\theta})$

Take derivatives, set to zero and solve…

$$\phi = \frac{(\alpha' - 1) + N_{y=1}}{(\alpha' - 1) + (\beta' - 1) + N}$$

$$\theta_{0,m} = \frac{(\alpha - 1) + N_{y=0, x_m=1}}{(\alpha - 1) + (\beta - 1) + N_{y=0}}$$

$$\theta_{1,m} = \frac{(\alpha - 1) + N_{y=1, x_m=1}}{(\alpha - 1) + (\beta - 1) + N_{y=1}}$$

$$\forall m \in \{1, \ldots, M\}$$

A common choice for the class prior:

α' = 1 and β' = 1

Since Beta(1,1) = Uniform(0,1)

THE NAÏVE BAYES FRAMEWORK

Many NB Models

There are many Naïve Bayes models!

1. **Bernoulli** Naïve Bayes:
   - for **binary features**
2. **Multinomial** Naïve Bayes:
   - for **integer features**
3. **Gaussian** Naïve Bayes:
   - for **continuous features**
4. **Multi-class** Naïve Bayes:
   - for classification problems with > 2 classes
   - **event model** could be any of Bernoulli, Gaussian, Multinomial, depending on features

Model 2: Multinomial Naïve Bayes

Support: Option 1: Integer vector (word IDs)

$\mathbf{x} = [x_1, x_2, \dots, x_M]$ where $x_m \in \{1, \dots, K\}$ a word id.

Generative Story:

```
for $i \in \{1, \dots, N\}$:
  $y^{(i)} \sim$ Bernoulli($\phi$)
  for $j \in \{1, \dots, M_i\}$:
    $x_j^{(i)} \sim$ Multinomial($\boldsymbol{\theta}_{y^{(i)}}, 1$)
```

Model:

$$p_{\phi,\boldsymbol{\theta}}(\boldsymbol{x}, y) = p_\phi(y) \prod_{k=1}^{K} p_{\boldsymbol{\theta}_k}(x_k | y)$$

$$= (\phi)^y (1-\phi)^{(1-y)} \prod_{j=1}^{M_i} \theta_{y,x_j}$$

Model 3: Gaussian Naïve Bayes

Support: $\mathbf{x} \in \mathbb{R}^K$

Model: Product of **prior** and the event model

$$p(\boldsymbol{x}, y) = p(x_1, \ldots, x_K, y)$$

$$= p(y) \prod_{k=1}^{K} p(x_k | y)$$

Gaussian Naive Bayes assumes that $p(x_k|y)$ is given by a Normal distribution.

Model 3: Gaussian Naïve Bayes

Support: $\mathbf{x} \in \mathbb{R}^K$

Model:

- Binary label
  - $Y \sim \text{Bernoulli}(\pi)$
  - $\hat{\pi} = {}^{N_{Y=1}}/_{N}$
    - N = # of data points
    - $N_{Y=1}$ = # of data points with label 1
- Real-valued features
  - $X_d | Y = y \sim \text{Gaussian}\left(\mu_{d,y}, \sigma^2_{d,y}\right)$
  - $\hat{\mu}_{d,y} = \frac{1}{N_{Y=y}} \sum_{n: y^{(n)}=y} x_d^{(n)}$
  - $\hat{\sigma}^2_{d,y} = \frac{1}{N_{Y=y}} \sum_{n: y^{(n)}=y} \left(x_d^{(n)} - \hat{\mu}_{d,y}\right)^2$
    - $N_{Y=y}$ = # of data points with label y

Model 4: Multiclass Naïve Bayes

Model:

The only change is that we permit y to range over C classes.

$$\begin{aligned} p(\boldsymbol{x}, y) &= p(x_1, \ldots, x_K, y) \\ &= p(y) \prod_{k=1}^{K} p(x_k | y) \end{aligned}$$

Now, $y \sim$ Multinomial$(\phi, 1)$ and we have a separate conditional distribution $p(x_k|y)$ for each of the C classes.

Model 4': Multiclass Gaussian Naïve Bayes

Support: $\mathbf{x} \in \mathbb{R}^K$

Model:

- Discrete label (Y can take on one of M possible values)
  - $Y \sim \text{Categorical}(\pi_1, \dots, \pi_M)$
  - $\hat{\pi}_m = {}^{N_{Y=m}}/_{N}$
    - N = # of data points
    - $N_{Y=m}$ = # of data points with label m
- Real-valued features
  - $X_d | Y = y \sim \text{Gaussian}\left(\mu_{d,y}, \sigma_{d,y}^2\right)$
  - $\hat{\mu}_{d,y} = \frac{1}{N_{Y=y}} \sum_{n: y^{(n)} = y} x_d^{(n)}$
  - $\hat{\sigma}_{d,y}^2 = \frac{1}{N_{Y=y}} \sum_{n: y^{(n)} = y} \left(x_d^{(n)} - \hat{\mu}_{d,y}\right)^2$
    - $N_{Y=y}$ = # of data points with label y

Generic Naïve Bayes Model

Support: Depends on the choice of **event model,** $P(X_k|Y)$

Model: Product of **prior** and the event model

$$P(\mathbf{X}, Y) = P(Y) \prod_{k=1}^{K} P(X_k|Y)$$

Training: Find the **class-conditional** MLE parameters

For $P(Y)$, we find the MLE using all the data. For each $P(X_k|Y)$ we condition on the data with the corresponding

Classification: Find the class that maximizes the posterior

$$\hat{y} = \underset{y}{\operatorname{argmax}}\, p(y|\mathbf{x})$$

Generic Naïve Bayes Model

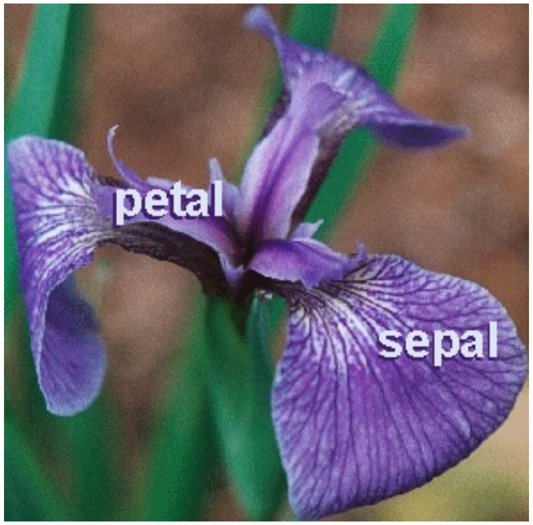
Classification:

$$\hat{y} = \underset{y}{\operatorname{argmax}}\, p(y|\mathbf{x}) \quad \text{(posterior)}$$

$$= \underset{y}{\operatorname{argmax}}\, \frac{p(\mathbf{x}|y)p(y)}{p(x)} \quad \text{(by Bayes' rule)}$$

$$= \underset{y}{\operatorname{argmax}}\, p(\mathbf{x}|y)p(y)$$

VISUALIZING GAUSSIAN NAÏVE BAYES


petal
sepal

Fisher Iris Dataset

Fisher (1936) used 150 measurements of flowers from 3 different species: Iris setosa (0), Iris virginica (1), Iris versicolor (2) collected by Anderson (1936)

Species	Sepal Length	Sepal Width	Petal Length	Petal Width
0	4.3	3.0	1.1	0.1
0	4.9	3.6	1.4	0.1
0	5.3	3.7	1.5	0.2
1	4.9	2.4	3.3	1.0
1	5.7	2.8	4.1	1.3
1	6.3	3.3	4.7	1.6
1	6.7	3.0	5.0	1.7

Full dataset: https://en.wikipedia.org/wiki/Iris_flower_data_set

Iris Data (2 classes)

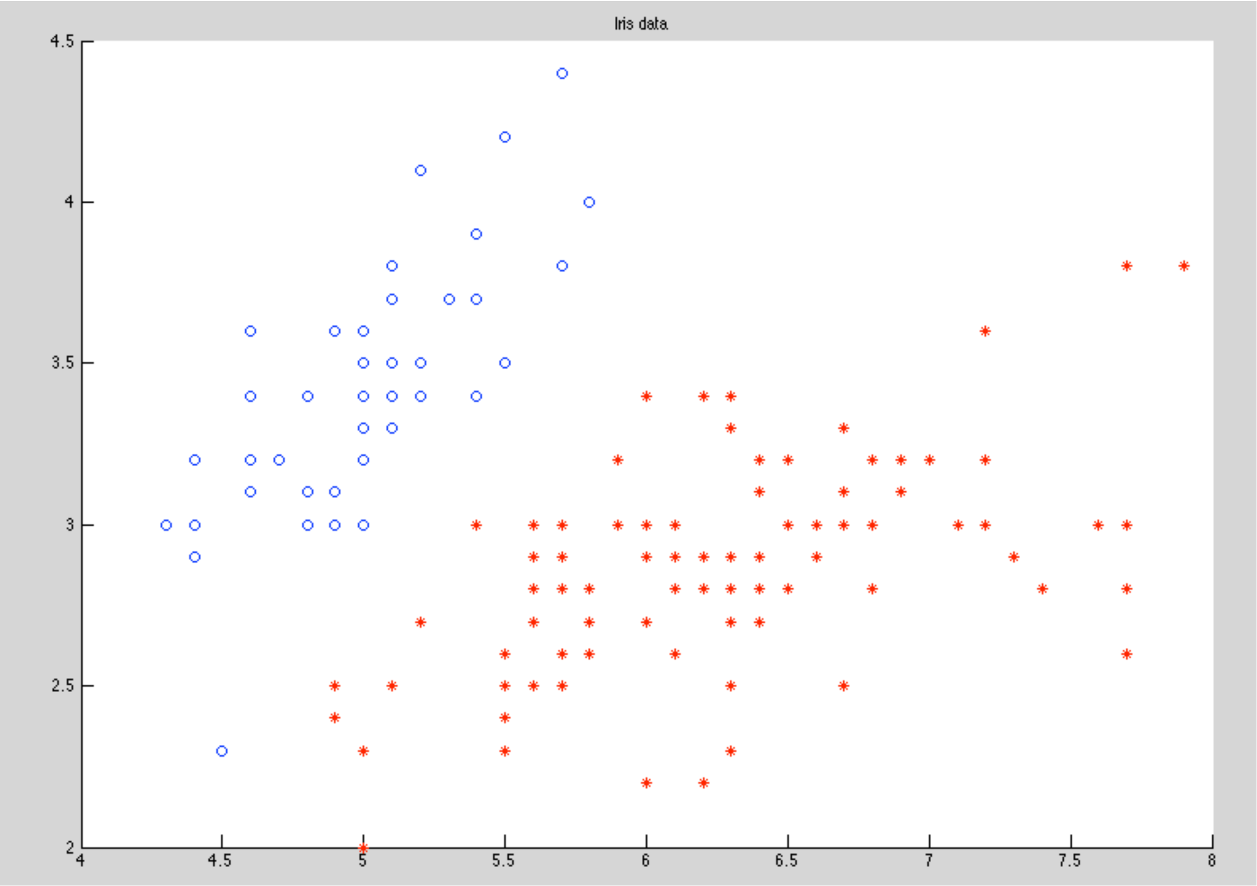


Figure from William Cohen

Iris Data (2 classes)

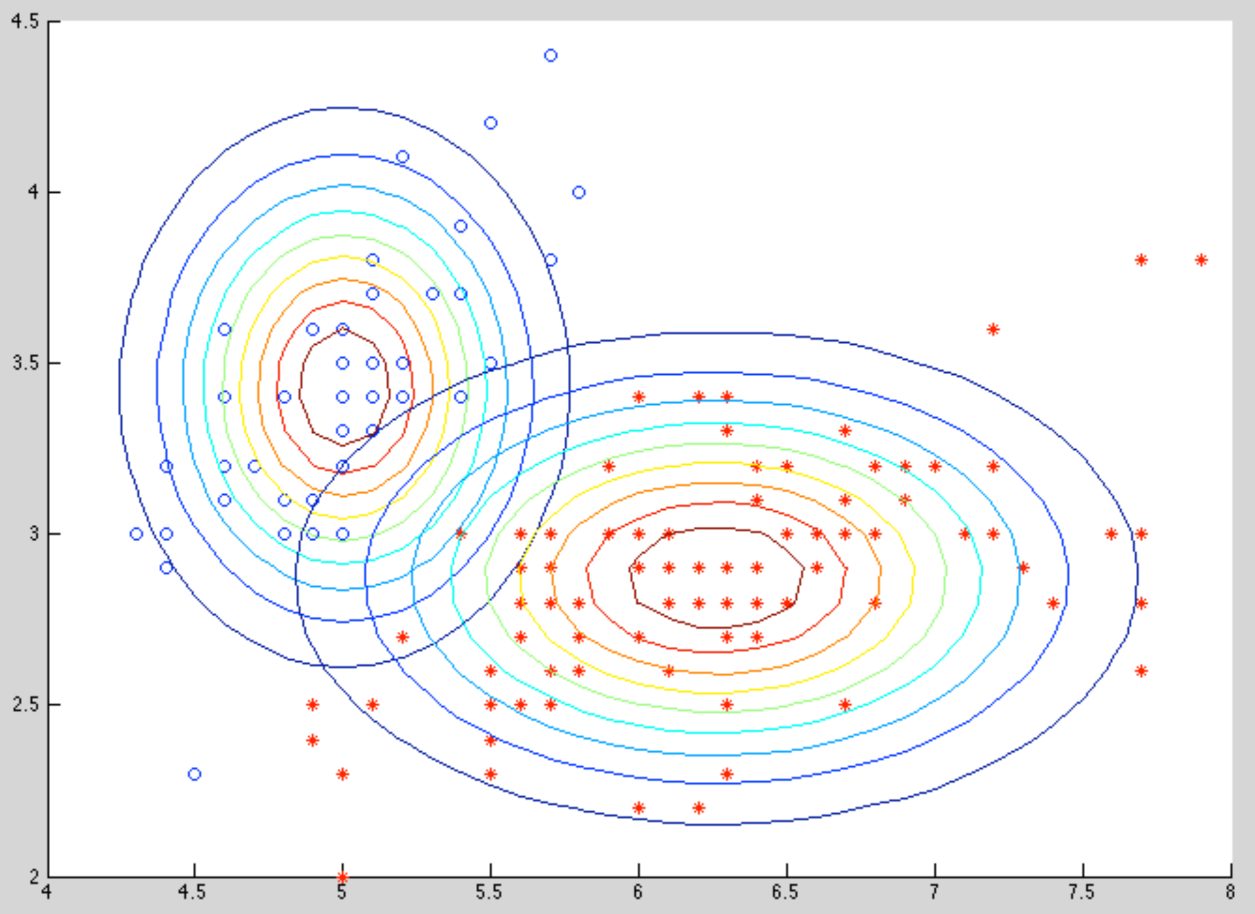
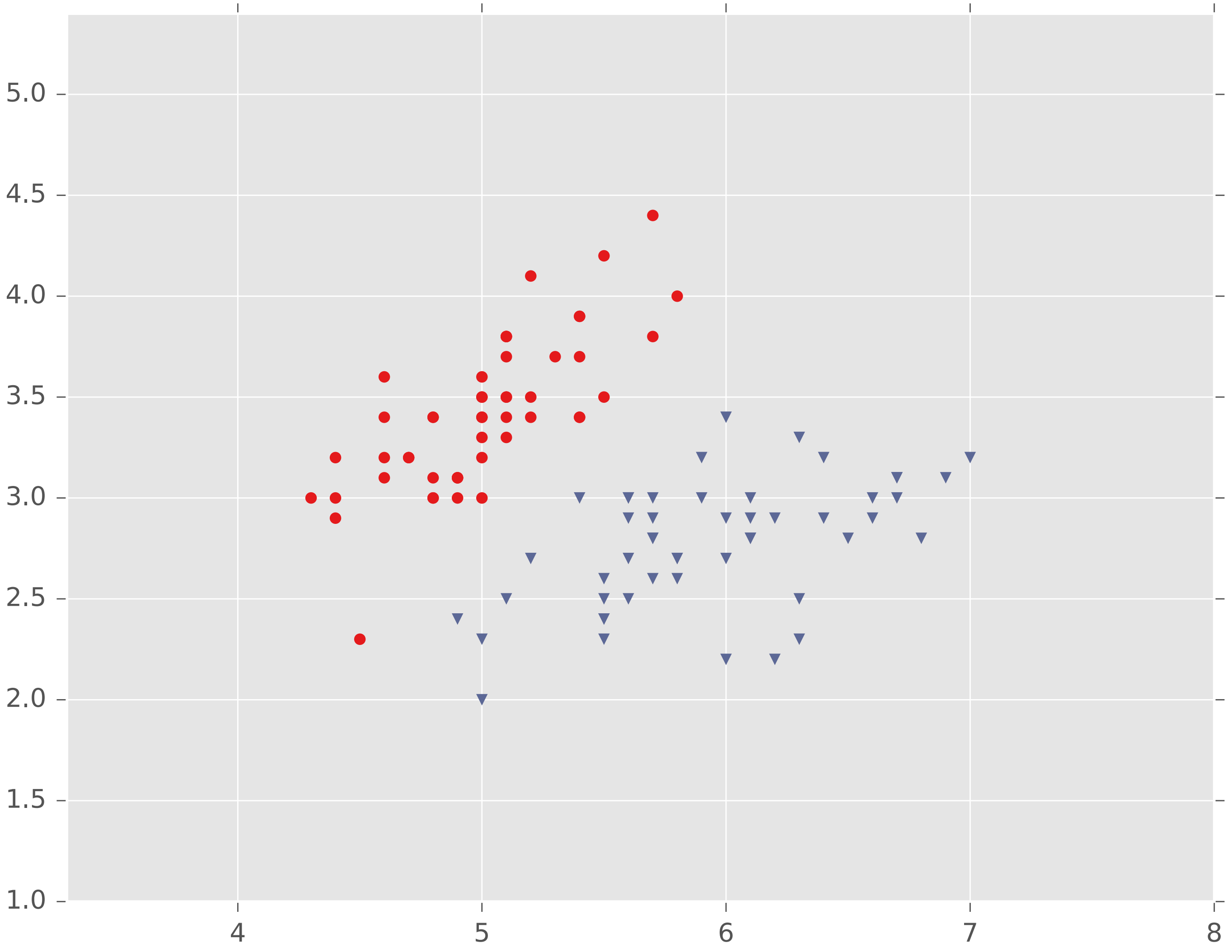


Figure from William Cohen

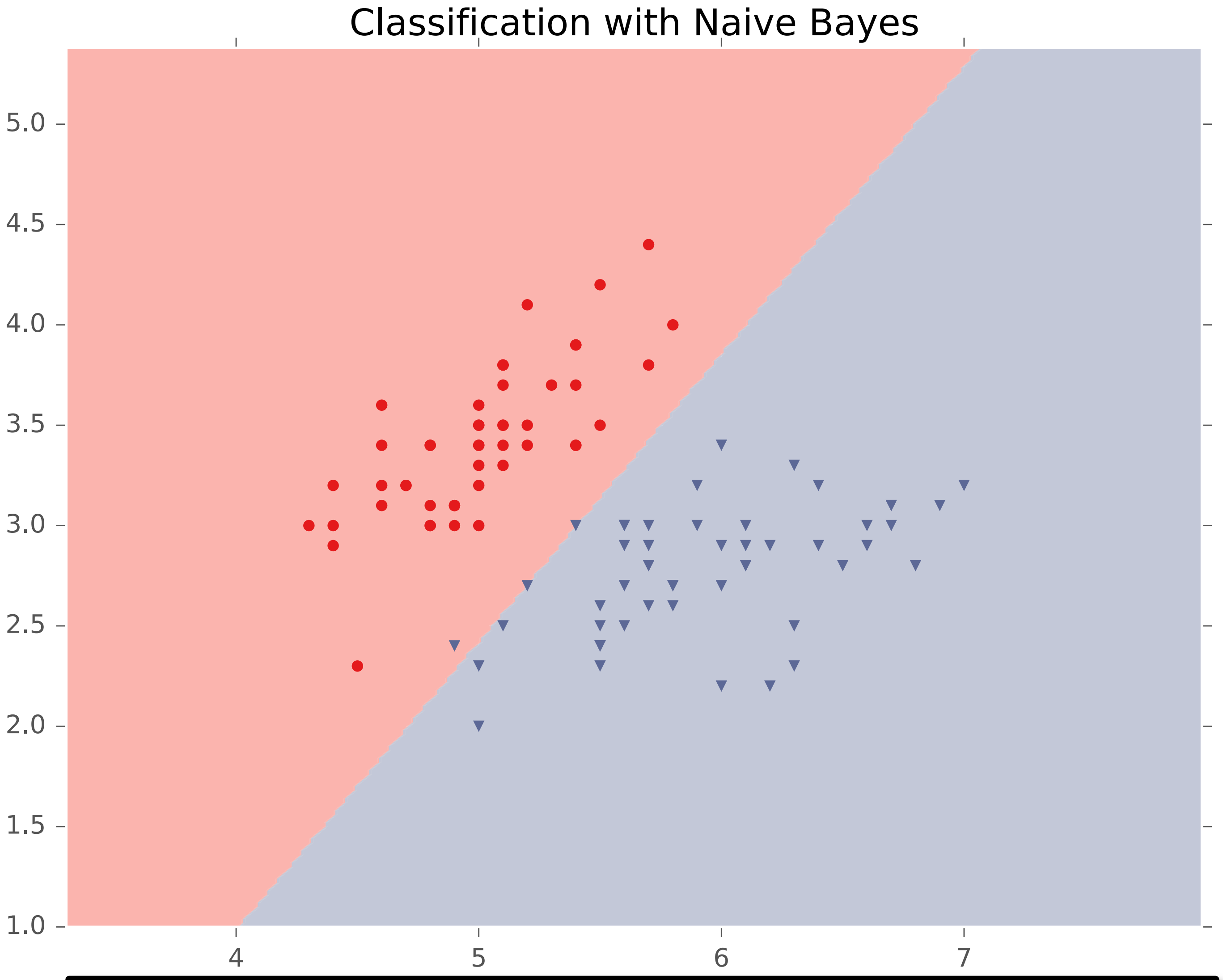
Iris Data (2 classes)

Naïve Bayes has a **linear** decision boundary if variance (sigma) is constant across classes



Iris Data (2 classes)

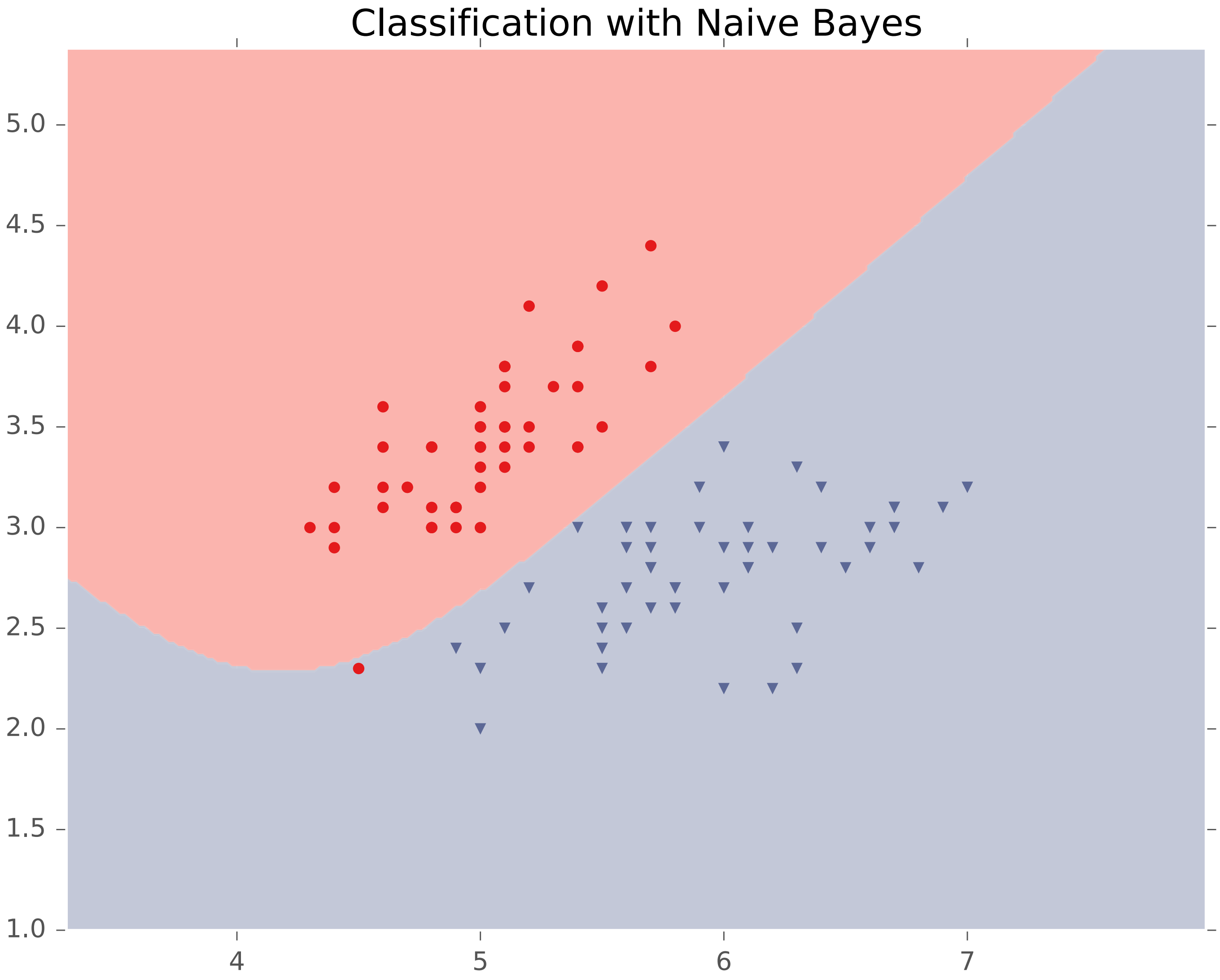
Naïve Bayes has a **linear** decision boundary if variance (sigma) is constant across classes



variance = 1

Iris Data (2 classes)

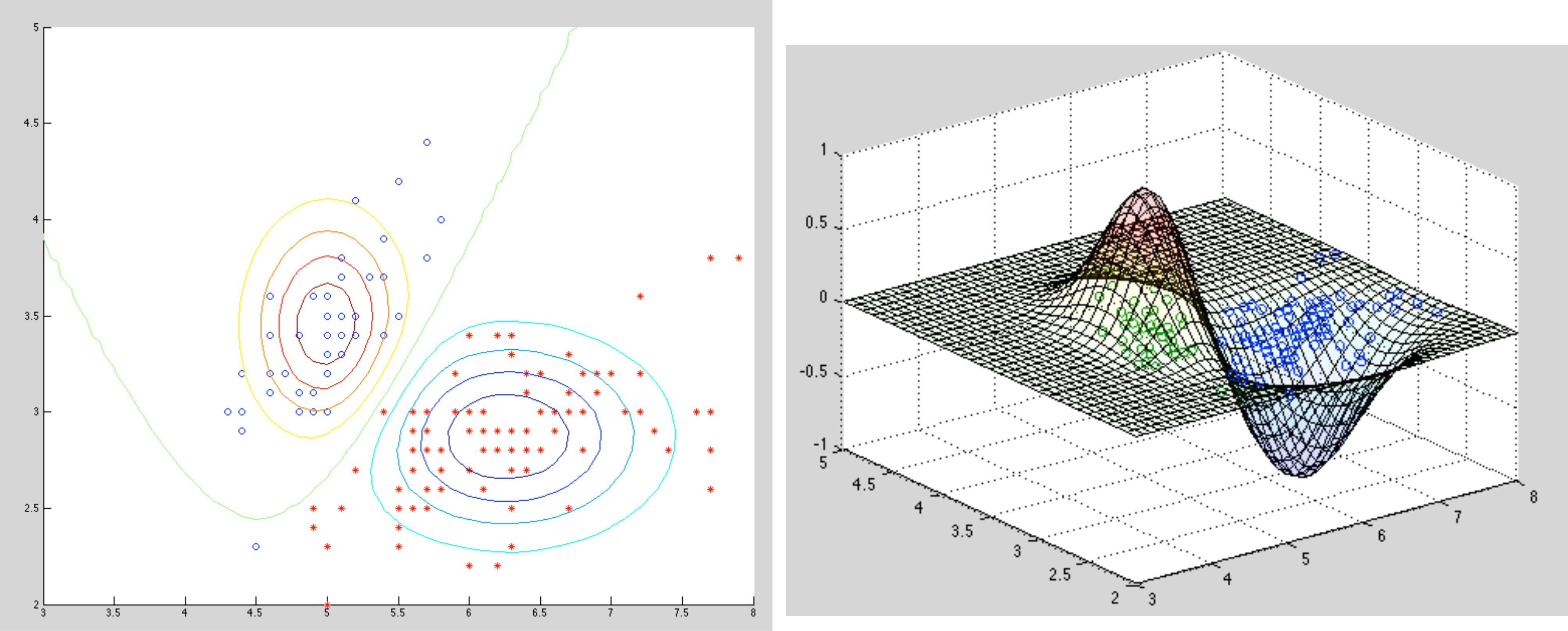
Naïve Bayes can have a **nonlinear** decision boundary if variance (sigma) can vary across classes



variance learned for each class

Iris Data (2 classes)

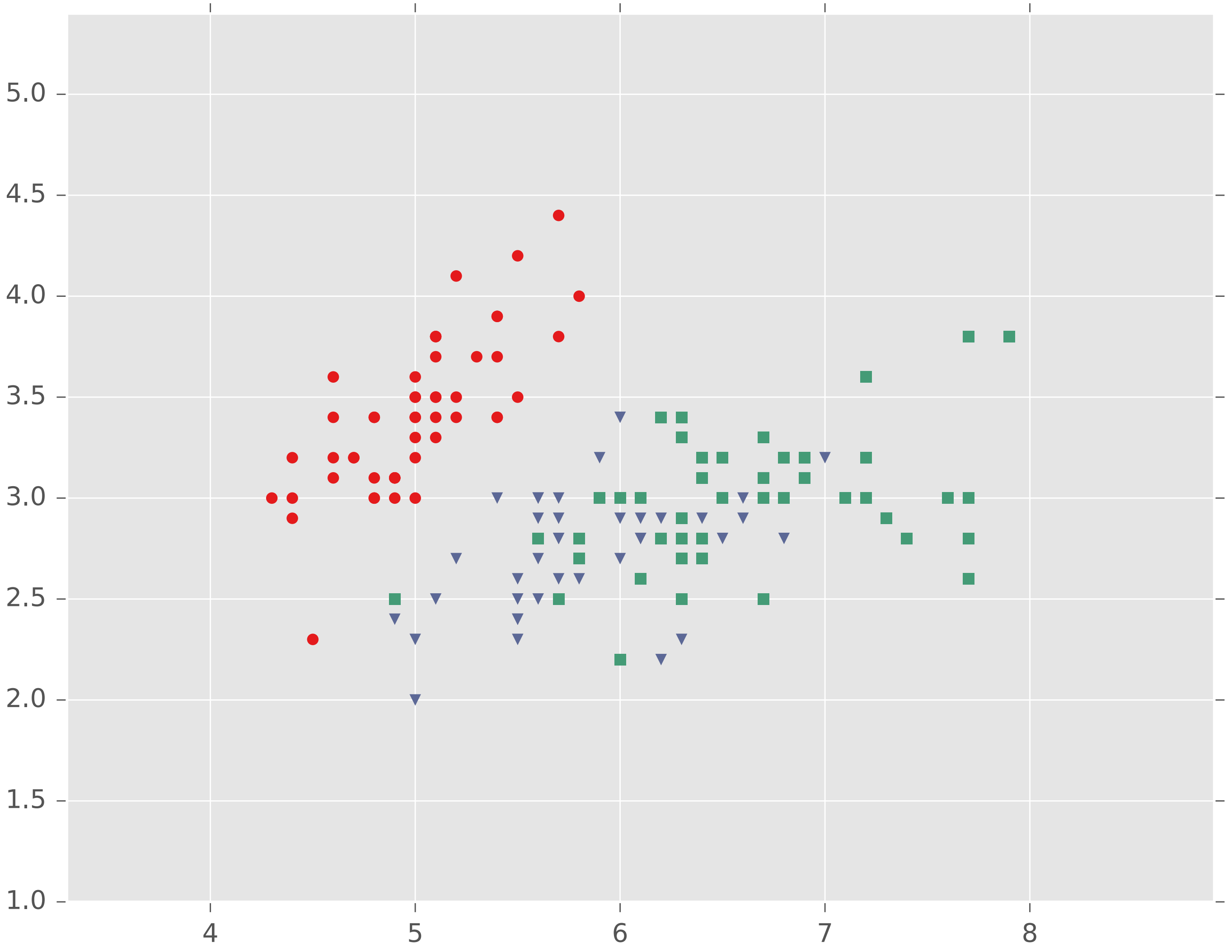
z-axis is the difference of the posterior probabilities: p(y=1 | **x**) – p(y=0 | **x**)



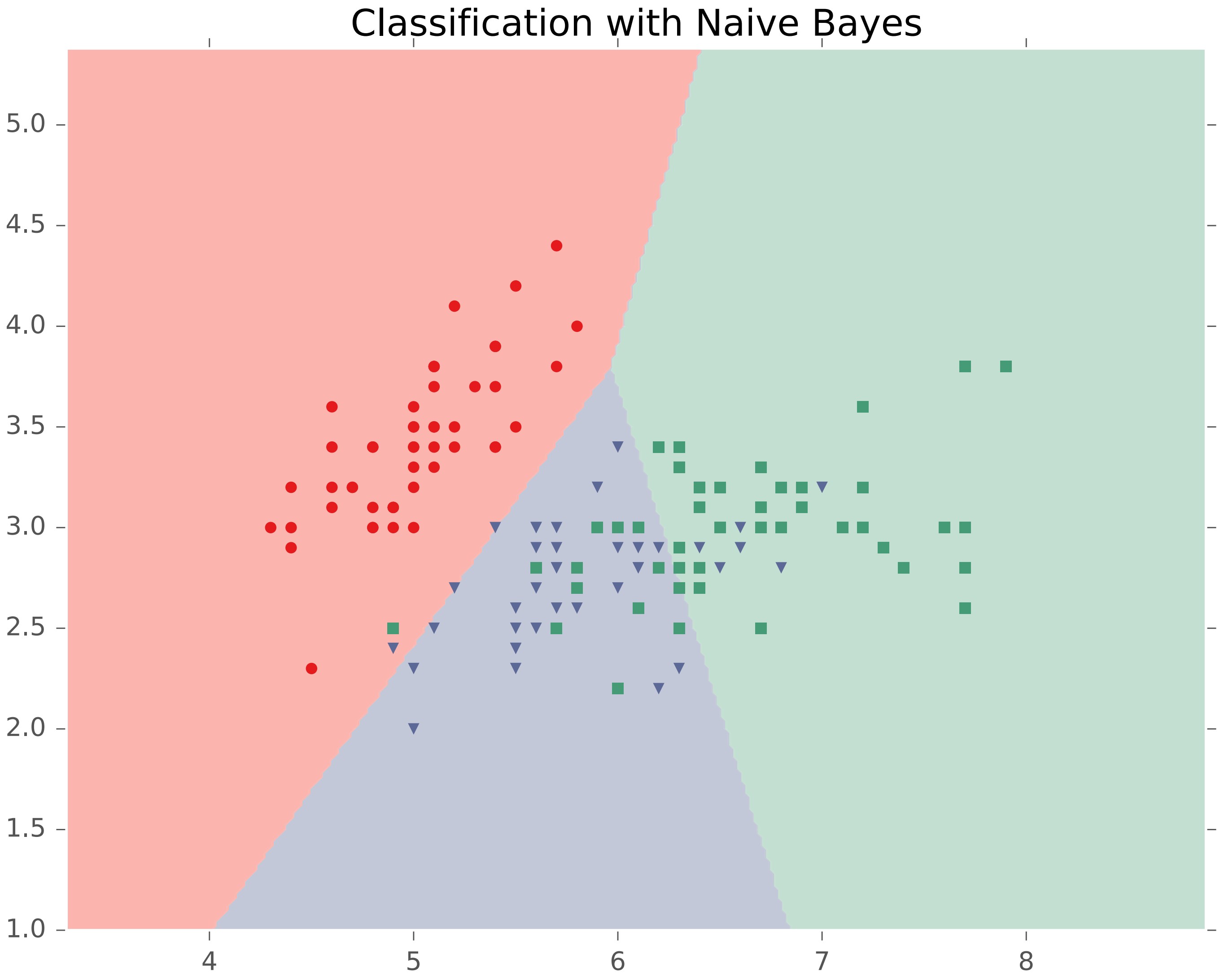
Figures from William Cohen

variance learned for each class

Iris Data (3 classes)

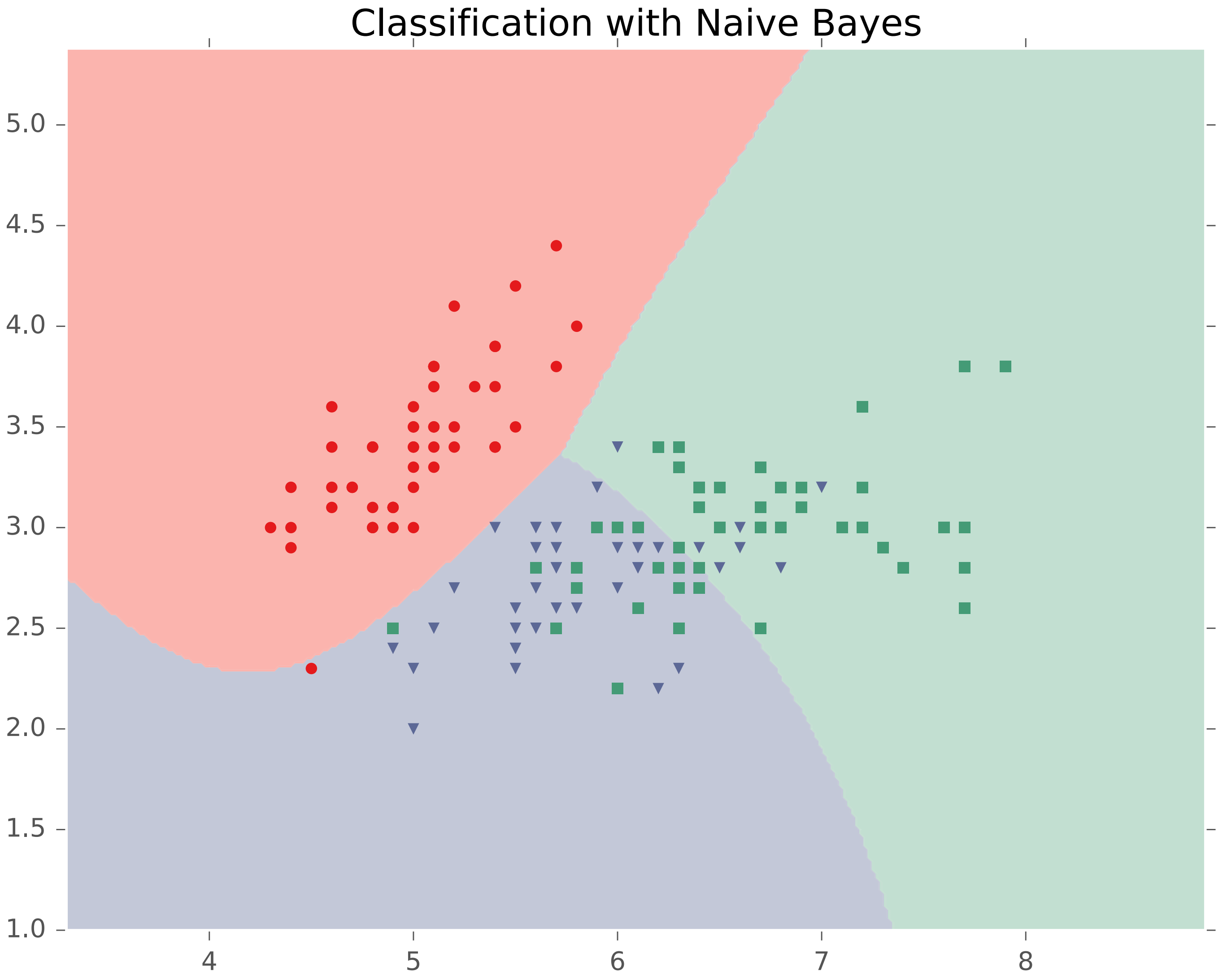


Iris Data (3 classes)



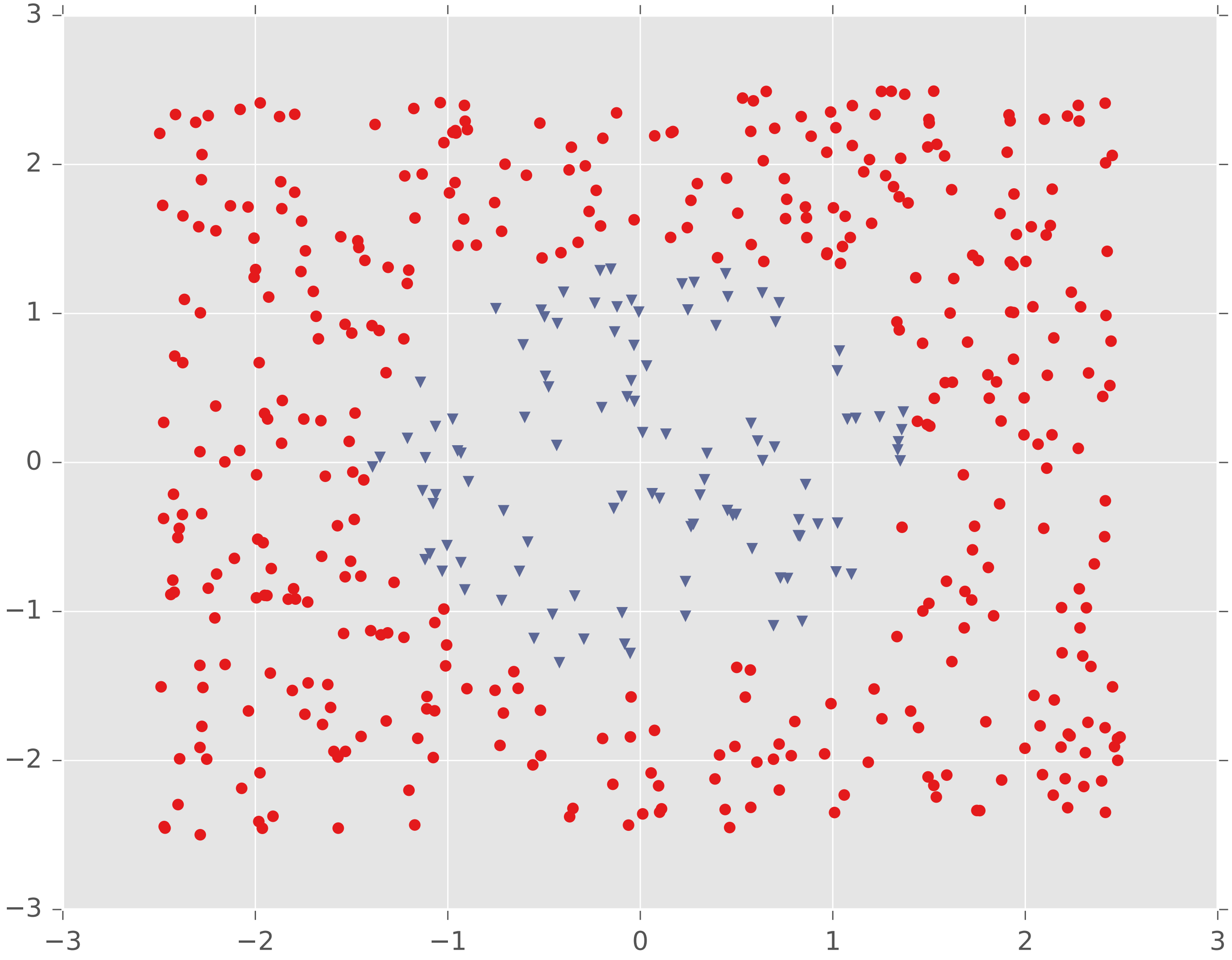
variance = 1

Iris Data (3 classes)

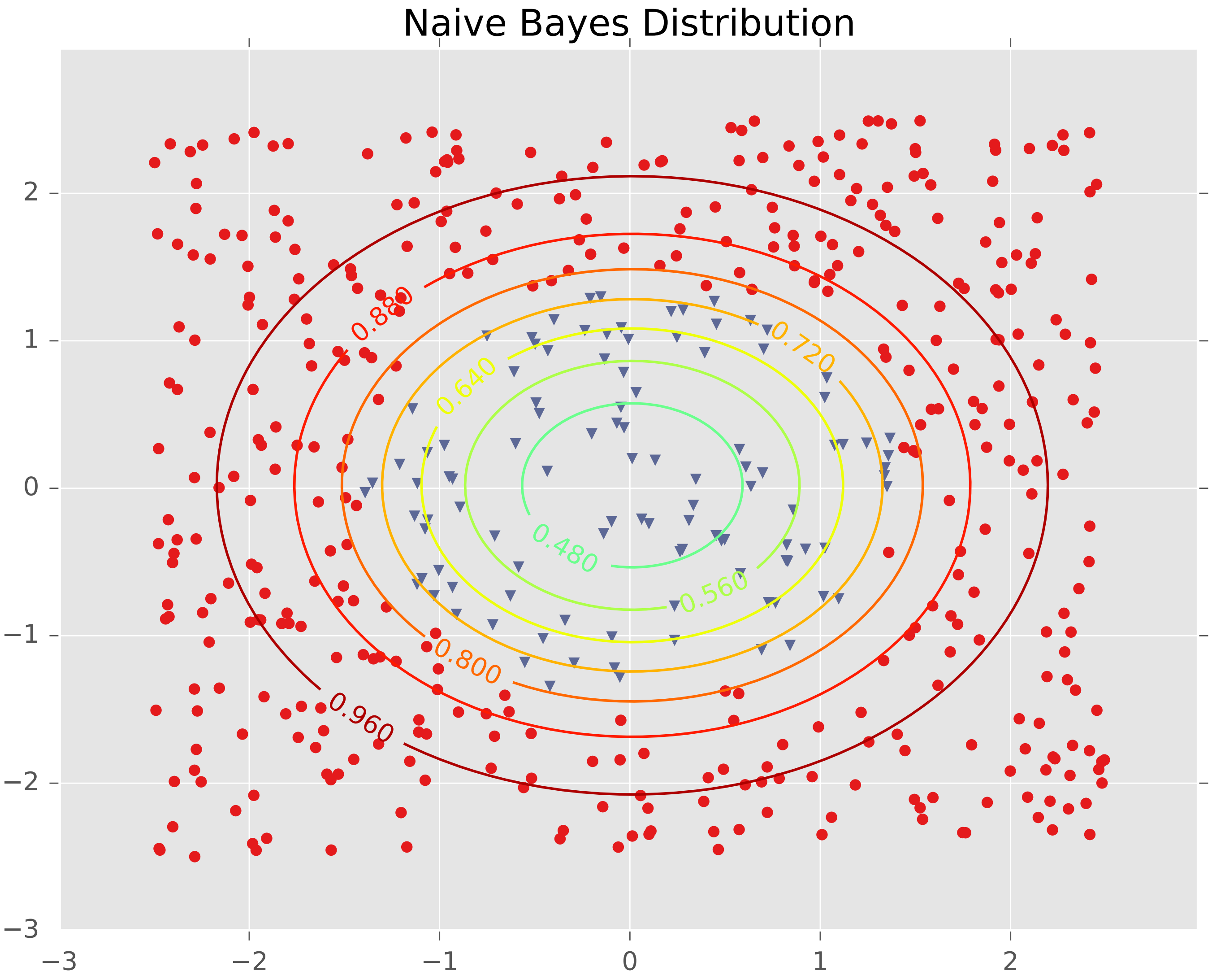


variance learned for each class

One Pocket

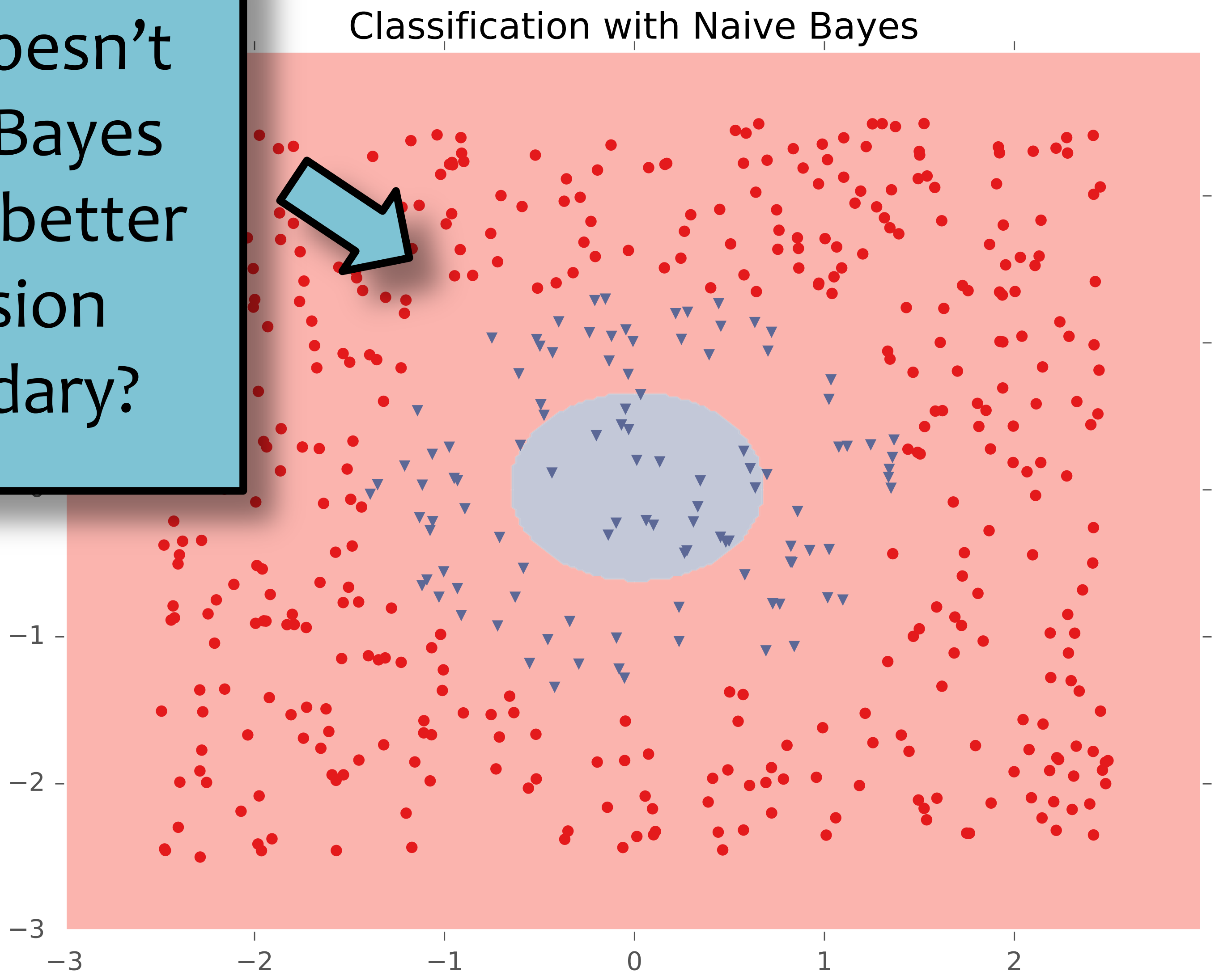


One Pocket



variance learned for each class

One Pocket



Why doesn’t Naïve Bayes learn a better decision boundary?

variance learned for each class

DISCRIMINATIVE AND GENERATIVE CLASSIFIERS

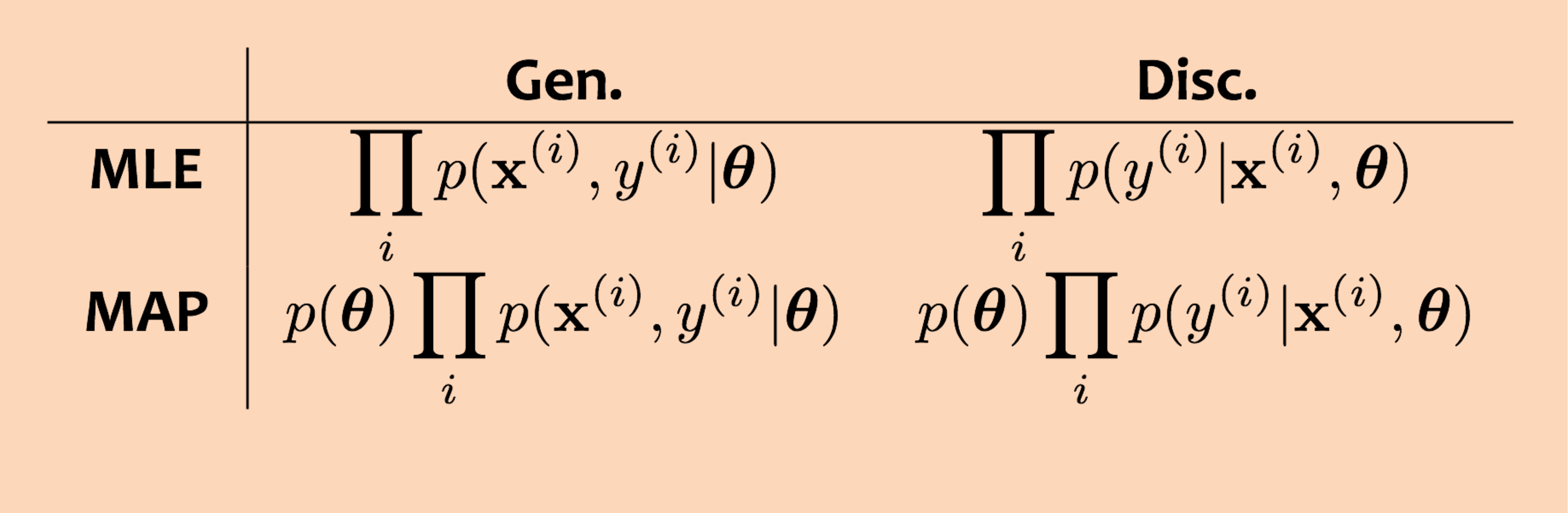
Generative vs. Discriminative

- **Generative Classifiers:**
  - Example: Naïve Bayes
  - Define a joint model of the observations **x** and the labels y: $p(\boldsymbol{x}, y)$
  - Learning maximizes (joint) likelihood
  - Use Bayes' Rule to classify based on the posterior:

$$p(y|\mathbf{x}) = p(\mathbf{x}|y)p(y)/p(\mathbf{x})$$

- **Discriminative Classifiers:**
  - Example: Logistic Regression
  - Directly model the conditional: $p(y|\mathbf{x})$
  - Learning maximizes conditional likelihood

Generative vs. Discriminative



	Gen.	**Disc.**
MLE	$\prod_i p(\mathbf{x}^{(i)}, y^{(i)} \| \boldsymbol{\theta})$	$\prod_i p(y^{(i)} \| \mathbf{x}^{(i)}, \boldsymbol{\theta})$
MAP	$p(\boldsymbol{\theta}) \prod_i p(\mathbf{x}^{(i)}, y^{(i)} \| \boldsymbol{\theta})$	$p(\boldsymbol{\theta}) \prod_i p(y^{(i)} \| \mathbf{x}^{(i)}, \boldsymbol{\theta})$