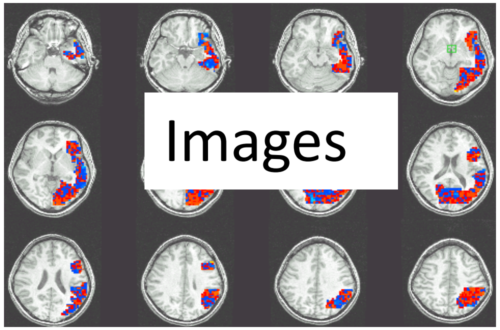
MLE/MAP for learning distributions

Aarti Singh & Geoff Gordon

Machine Learning 10-701
Feb 3, 2021

Notion of "Features aka Attributes"

Input $X \in \mathcal{X}$



How to represent inputs mathematically?

Document vector X

- frequency of words (length of document = size of vocabulary), also known as **Bag-of-words** approach

remember to wake up when class ends
= wake ends to class remember up when

Misses out context!!

- list of n-grams (n-tuples of words)

Image X

- intensity/value at each pixel
- fourier transform values
- SIFT
- Deep representation

Distribution of Inputs

Input $X \in \mathcal{X}$

Discrete Probability Distribution P(X) = P(X=x)

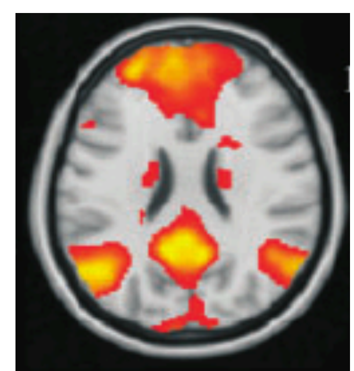
e.g. P(head) = ½, P(word x in text) = p_x

Probabilities in a distribution sum to 1

$\sum_x P(X=x) = 1$ P(tail) = 1 – p(head), $\sum_x p_x = 1$

Continuous Probability density p(x) P(a<=X<=b) = $\int_a^b p(x)dx$

e.g. p(brain activity)



Probability density integrate to 1

$$\int p(x)dx = 1$$

Distributions in Supervised tasks

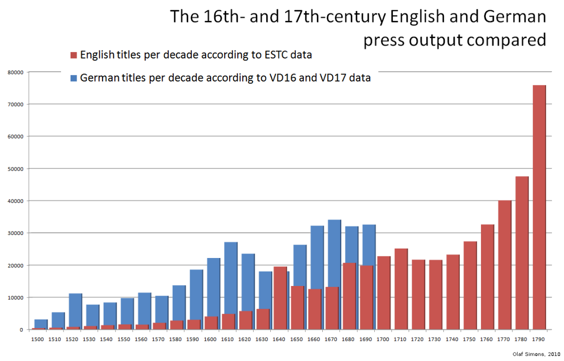
Input $X \in \mathcal{X}$

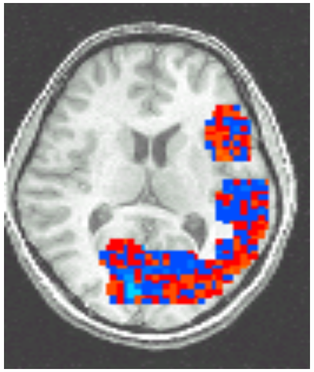
- Distribution learning also arises in supervised learning tasks e.g. classification

P(Y= y) Distribution of class labels

P(X = x | Y = y) Distribution of words in 'news' documents

Distribution of brain activity under 'stress'





Olaf simons'10

P(Y = y|X = x) Distribution of topics given document

How to learn parameters from data? MLE

(Discrete case)

Learning parameters in distributions

$$P(Y = {\color{red}\bullet}) = \theta \qquad P(Y = {\color{green}\bullet}) = 1 - \theta$$

Learning θ is equivalent to learning probability of head in coin flip.

- How do you learn that?

Data = 

Answer: 3/5

- Why??

Bernoulli distribution

Data, D = 

- Parameter θ : P(Heads) = θ, P(Tails) = 1-θ
- Flips are **i.i.d.**:
  - **Independent** events
  - **Identically distributed** according to Bernoulli distribution

<u>Choose θ that maximizes the probability of observed data aka Likelihood</u>

Maximum Likelihood Estimation (MLE)

Choose θ that maximizes the probability of observed data (aka likelihood)

$$\widehat{\theta}_{MLE} = \arg\max_{\theta} P(D \mid \theta)$$

MLE of probability of head:

$$\widehat{\theta}_{MLE} = \frac{\alpha_H}{\alpha_H + \alpha_T}$$ **= 3/5**

"Frequency of heads"

Derivation

$$\widehat{\theta}_{MLE} = \arg\max_{\theta} P(D \mid \theta)$$

Multinomial distribution

Data, D = rolls of a dice

- $P(1) = p_1$, $P(2) = p_2$, …, $P(6) = p_6$ $p_1 + \ldots + p_6 = 1$
- Rolls are **i.i.d.**:
  - **Independent** events
  - **Identically distributed** according to Multinomial(θ) distribution where

$$\theta = \{p_1, p_2, \ldots, p_6\}$$

Choose θ that maximizes the probability of observed data aka "Likelihood"

Maximum Likelihood Estimation (MLE)

Choose θ that maximizes the probability of observed data

$$\widehat{\theta}_{MLE} = \arg\max_{\theta} P(D \mid \theta)$$

MLE of probability of rolls:

$$\widehat{\theta}_{MLE} = \hat{p}_{1,MLE}, \ldots, \hat{p}_{6,MLE}$$

$$\hat{p}_{y,MLE} = \frac{\alpha_y}{\sum_y \alpha_y}$$

α_y ← Rolls that turn up y

$\sum_y \alpha_y$ ← Total number of rolls

"Frequency of roll y"

How to learn parameters from data? MLE

(Continuous case)

d-dim Gaussian distribution

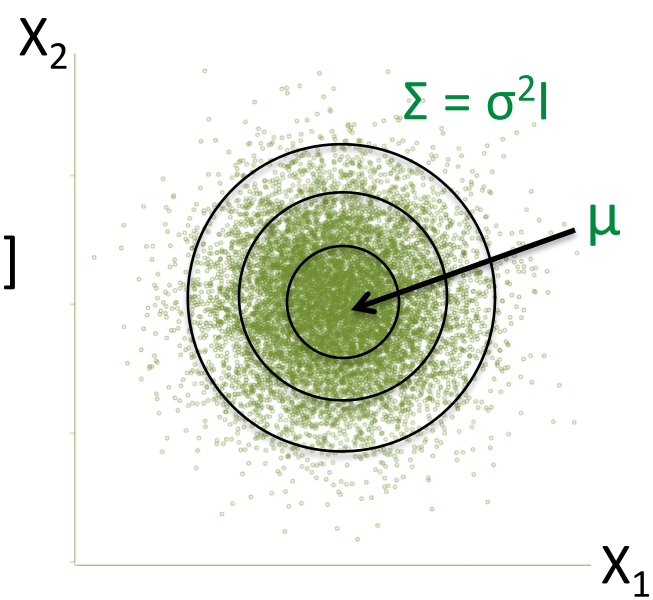
X is Gaussian N(μ, Σ)

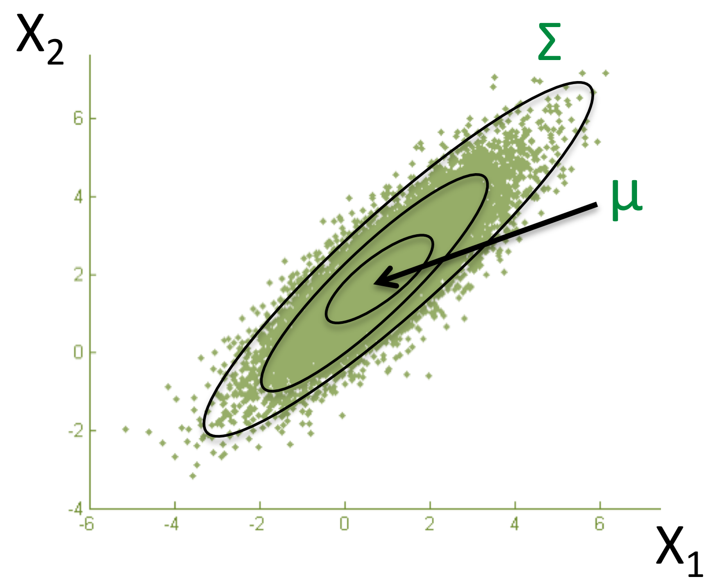
μ is d-dim vector, Σ is dxd dim matrix

$$P(X=x|\mu,\Sigma)=\frac{1}{\sqrt{(2\pi)^d|\Sigma|}}\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right),$$

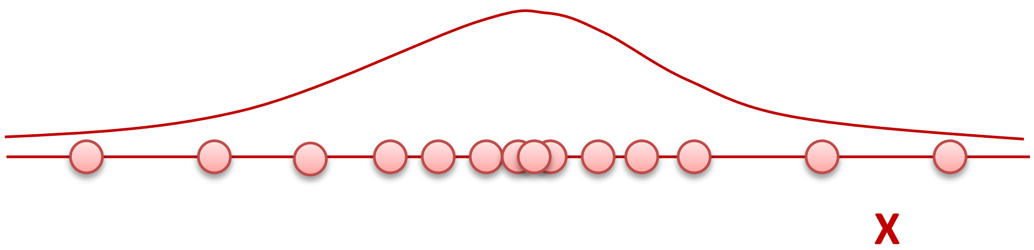
d=2

$X = [X_1; X_2]$





Gaussian distribution

Data, D = 

- Parameters: μ – mean, σ^2 - variance

- Data are **i.i.d.**:
  - **Independent** events
  - **Identically distributed** according to Gaussian distribution

Maximum Likelihood Estimation (MLE)

Choose $\theta = (\mu, \sigma^2)$ that maximizes the probability of observed data

$$\widehat{\theta}_{MLE} = \arg\max_{\theta} P(D \mid \theta)$$

$$= \arg\max_{\theta} \prod_{i=1}^{n} P(X_i|\theta)$$

Independent draws

Maximum Likelihood Estimation (MLE)

Choose $\theta = (\mu, \sigma^2)$ that maximizes the probability of observed data

$$\hat{\theta}_{MLE} = \arg\max_{\theta} P(D \mid \theta)$$

$$= \arg\max_{\theta} \prod_{i=1}^{n} P(X_i|\theta)$$

Independent draws

$$= \arg\max_{\theta} \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(X_i-\mu)^2/2\sigma^2}$$

Identically distributed

Maximum Likelihood Estimation (MLE)

Choose $\theta = (\mu, \sigma^2)$ that maximizes the probability of observed data

$$\hat{\theta}_{MLE} = \arg\max_{\theta} P(D \mid \theta)$$

$$= \arg\max_{\theta} \prod_{i=1}^{n} P(X_i|\theta)$$ Independent draws

$$= \arg\max_{\theta} \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(X_i-\mu)^2/2\sigma^2}$$ Identically distributed

$$= \arg \max_{\theta=(\mu,\sigma^2)} \underbrace{\frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\sum_{i=1}^{n}(X_i-\mu)^2/2\sigma^2}}_{J(\theta)}$$

MLE for Gaussian mean

MLE for Gaussian mean and variance

$$\widehat{\mu}_{MLE} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\widehat{\sigma}^2_{MLE} = \frac{1}{n}\sum_{i=1}^{n} (x_i - \widehat{\mu})^2$$

<u>Self exercise:</u>
Derive MLE of variance?

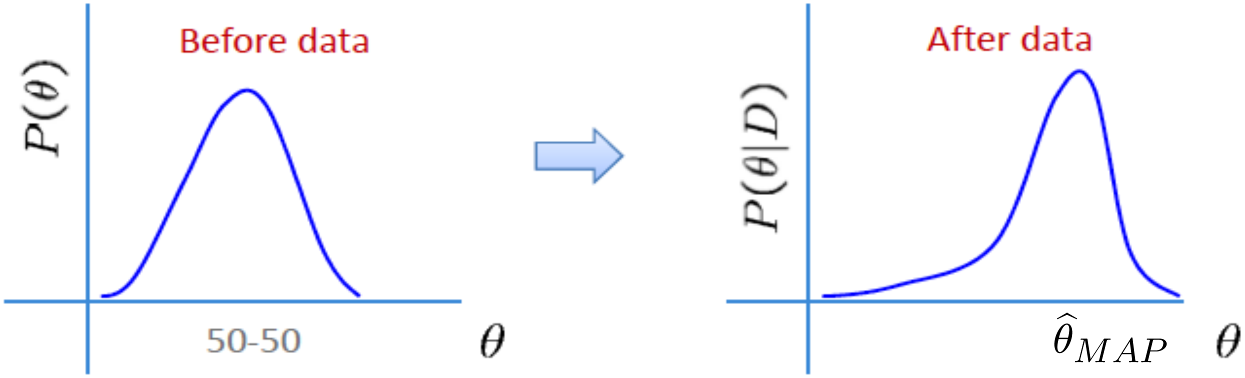
Is the MLE of mean unbiased?
Is the MLE of variance unbiased?
How can you make it unbiased?

d-dimensional versions?

Max A Posteriori (MAP) estimation

Can we bring in prior knowledge if data is not enough?

- Assume a prior (before seeing data D) distribution P(θ) for parameters θ



- Choose value that maximizes a posterior distribution P(θ|D) of parameters θ

$$\widehat{\theta}_{MAP} = \arg\max_{\theta} P(\theta \mid D) = \arg\max_{\theta} P(D \mid \theta) P(\theta)$$

How to choose prior distribution?

- $P(\theta)$
  - Prior knowledge about domain e.g. unbiased coin $P(\theta) = 1/2$
  - A mathematically convenient form e.g. "conjugate" prior

    If $P(\theta)$ is conjugate prior for $P(D|\theta)$, then Posterior has same form as prior

    Posterior = Likelihood x Prior

    $$P(\theta|D) = P(D|\theta) \times P(\theta)$$

e.g.	Beta	Bernoulli	Beta	θ = bias
	Gaussian	Gaussian	Gaussian	θ = mean μ (known Σ)
	inv-Wishart	Gaussian	inv-Wishart	θ = cov matrix Σ (known μ)

MAP estimation for Bernoulli r.v.

Choose θ that maximizes a posterior probability

$$\begin{aligned}\widehat{\theta}_{MAP} &= \arg\max_{\theta} \quad P(\theta \mid D) \\ &= \arg\max_{\theta} \quad P(D \mid \theta)P(\theta)\end{aligned}$$

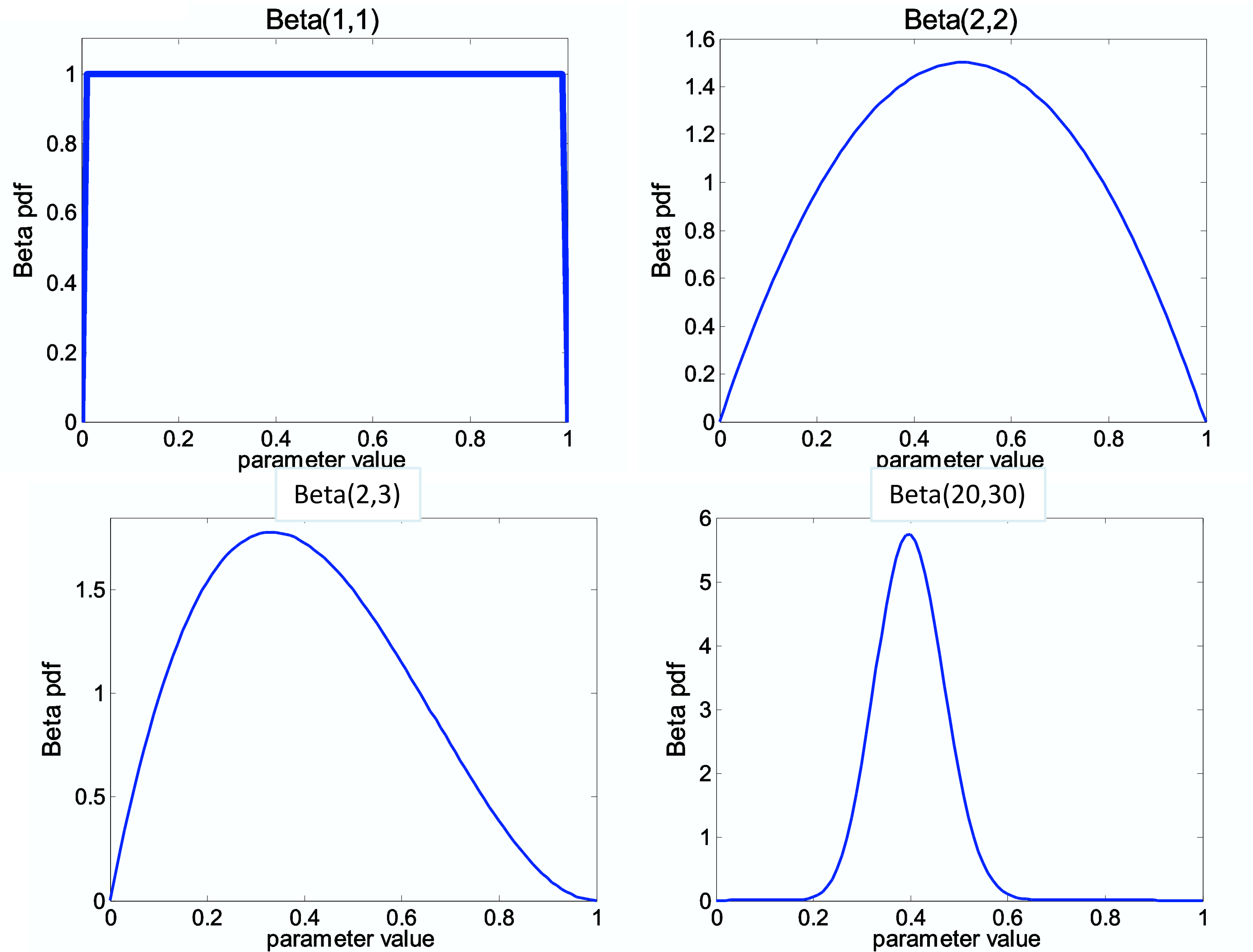
MAP estimate of probability of head (using Beta conjugate prior):

$$P(\theta) \sim Beta(\beta_H, \beta_T)$$

Beta distribution

$Beta(\beta_H, \beta_T)$

More concentrated as values of β_H, β_T increase



MAP estimation for Bernoulli r.v.

Choose θ that maximizes a posterior probability

$$\widehat{\theta}_{MAP} = \arg\max_{\theta} \; P(\theta \mid D)$$
$$= \arg\max_{\theta} \; P(D \mid \theta) P(\theta)$$

MAP estimate of probability of head (using Beta conjugate prior):

$$P(\theta) \sim Beta(\beta_H, \beta_T)$$

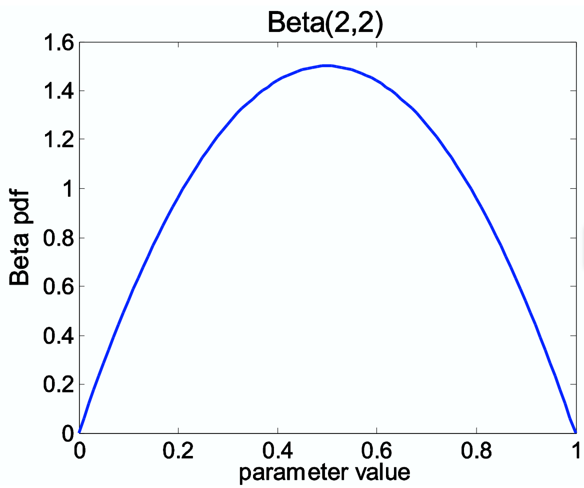
$$P(\theta|D) \sim Beta(\beta_H + \alpha_H, \beta_T + \alpha_T)$$

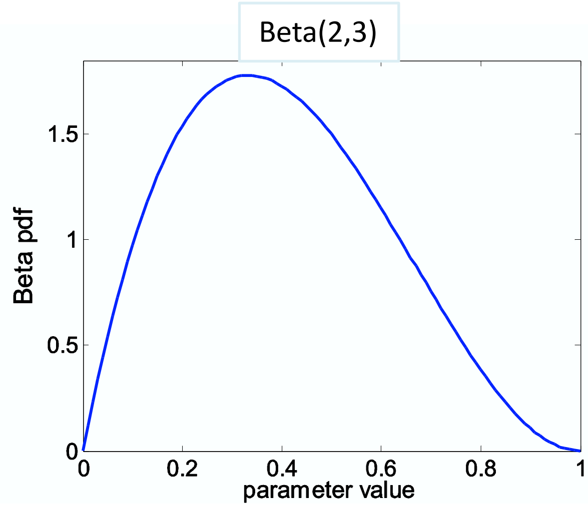
Count of H/T simply get added to parameters

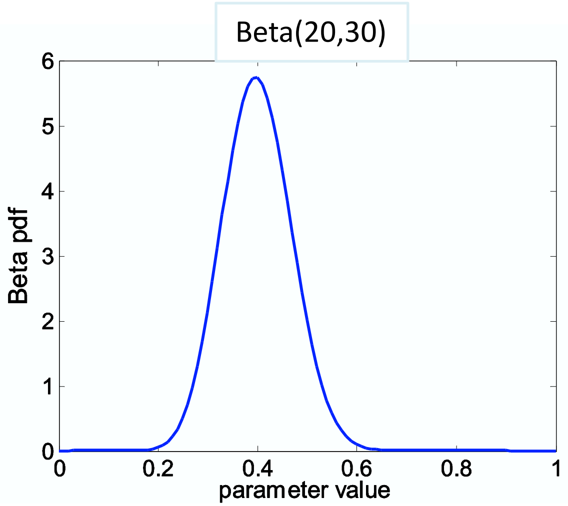
Beta conjugate prior

$$P(\theta) \sim Beta(\beta_H, \beta_T)$$

$$P(\theta|D) \sim Beta(\beta_H + \alpha_H, \beta_T + \alpha_T)$$







After observing 1 Tail

After observing 18 Heads and 28 Tails

As $n = \alpha_H + \alpha_T$ increases, posterior distribution becomes more concentrated

MAP estimation for Bernoulli r.v.

Choose θ that maximizes a posterior probability

$$\begin{aligned}\widehat{\theta}_{MAP} &= \arg\max_{\theta} \quad P(\theta \mid D) \\ &= \arg\max_{\theta} \quad P(D \mid \theta)P(\theta)\end{aligned}$$

MAP estimate of probability of head:

$$P(\theta) \sim Beta(\beta_H, \beta_T)$$

Count of H/T simply get added to parameters

$$P(\theta|D) \sim Beta(\beta_H + \alpha_H, \beta_T + \alpha_T)$$

$$\widehat{\theta}_{MAP} = \frac{\alpha_H + \beta_H - 1}{\alpha_H + \beta_H + \alpha_T + \beta_T - 2}$$

Mode of Beta distribution

Equivalent to adding extra coin flips (β_H - 1 heads, β_T - 1 tails)

As we get more data, effect of prior is "washed out"

MAP estimation for Gaussian r.v.

Parameters $\theta = (\mu,\sigma^2)$

- Mean μ (known σ^2): Gaussian prior $P(\mu) = N(\eta,\lambda^2)$

$$P(\mu \mid \eta, \lambda) = \frac{1}{\lambda\sqrt{2\pi}} e^{\frac{-(\mu-\eta)^2}{2\lambda^2}}$$

$$\widehat{\mu}_{MAP} = \frac{\frac{1}{\sigma^2}\sum_{i=1}^{n} x_i + \frac{\eta}{\lambda^2}}{\frac{n}{\sigma^2} + \frac{1}{\lambda^2}} \qquad \widehat{\mu}_{MLE} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

As we get more data, effect of prior is "washed out"

- Variance σ^2 (known μ): inv-Wishart Distribution

MLE vs. MAP

- Maximum Likelihood estimation (MLE)

  Choose value that maximizes the probability of observed data

$$\widehat{\theta}_{MLE} = \arg\max_{\theta} P(D|\theta)$$

- Maximum *a posteriori* (MAP) estimation

  Choose value that is most probable given observed data and prior belief

$$\begin{aligned}\widehat{\theta}_{MAP} &= \arg\max_{\theta} P(\theta|D) \\ &= \arg\max_{\theta} P(D|\theta)P(\theta)\end{aligned}$$

When is MAP same as MLE?

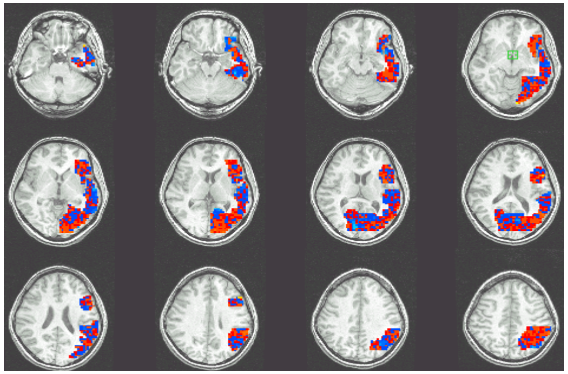
Bayes and Naïve Bayes Classifier

Aarti Singh & Geoff Gordon

Machine Learning 10-701
Feb 3, 2021

Classification

Goal: Construct **prediction rule** $f : \mathcal{X} \rightarrow \mathcal{Y}$



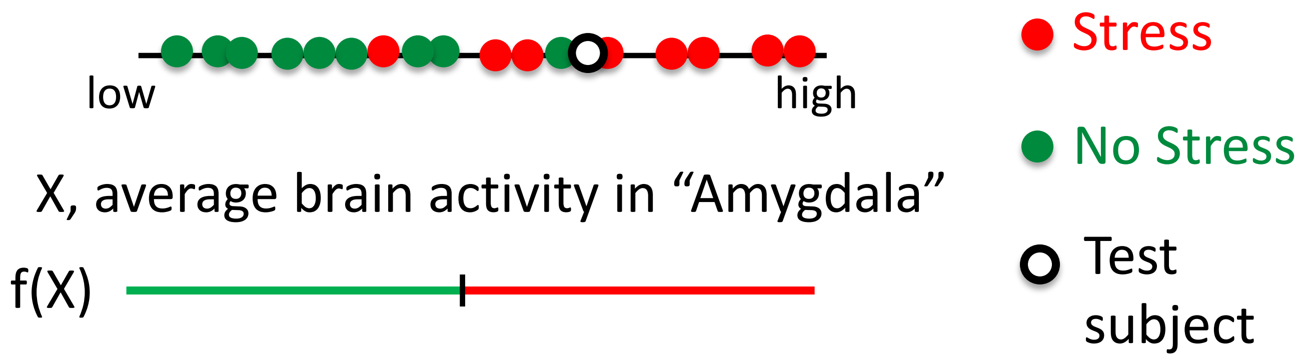
High Stress
Moderate Stress
Low Stress

Input feature vector, X **Label, Y**

In general: label Y can belong to more than two classes
X is multi-dimensional (many features represent an input)

But lets start with a simple case:
label Y is binary (either "Stress" or "No Stress")
X is average brain activity in the "Amygdala"

Binary Classification



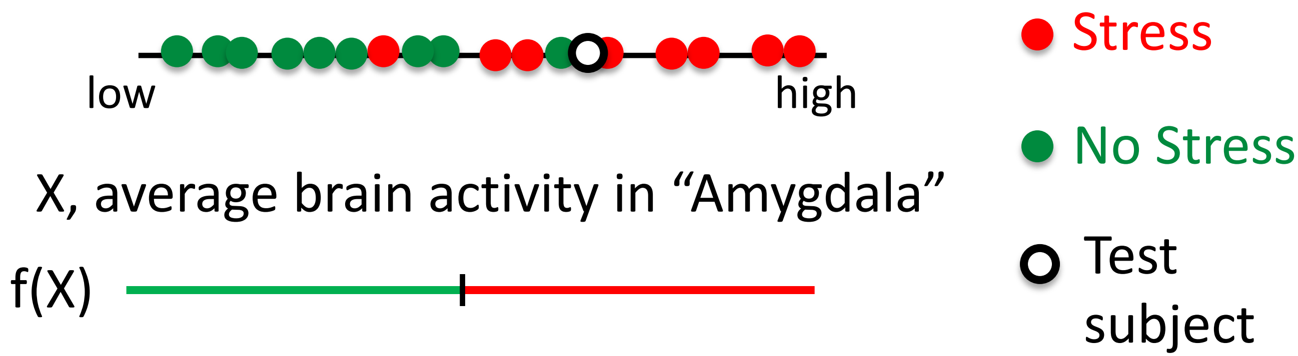
Model X and Y as random variables with joint distribution P_{XY}

Training data $\{X_i, Y_i\}^n_{i=1}$ ~ iid (independent and identically distributed) samples from P_{XY}

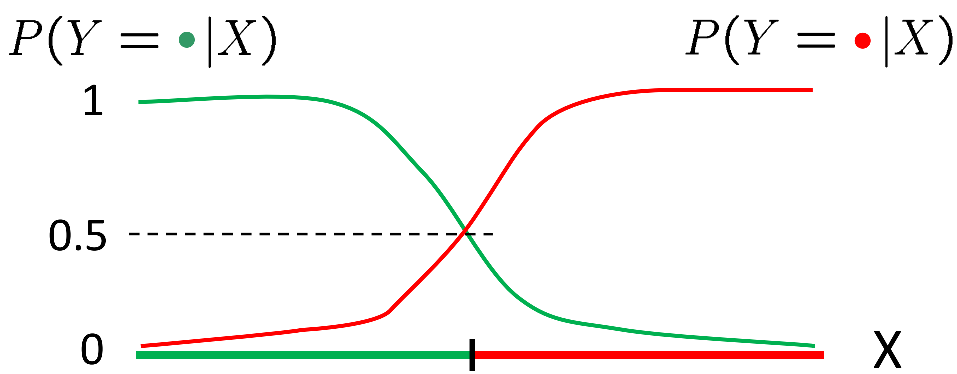
Test data $\{X,Y\}$ ~ iid sample from P_{XY}

Training and test data are independent draws from **same** distribution

Bayes Optimal Classifier



Model X and Y as random variables



For a given X, f(X) = label Y which is more likely

f(X) = $\arg\max_y P(Y = y|X = x)$

Optimality of Bayes Classifier

Bayes Rule

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

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

To see this, recall:

P(X,Y) = P(X|Y) P(Y)

P(Y,X) = P(Y|X) P(X)

Thomas Bayes

Bayes Classifier

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

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

Bayes classifier:

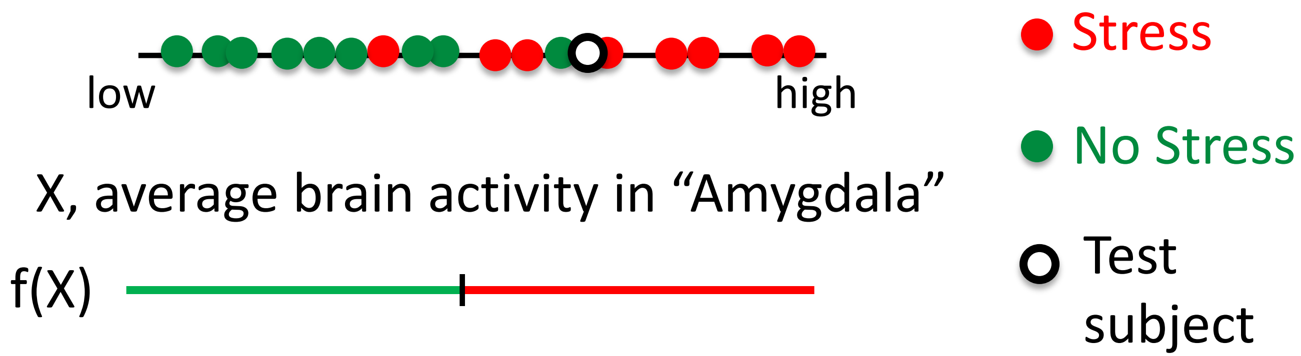
$$f(X) = \arg\max_{Y=y} P(Y = y|X = x)$$

$$= \arg\max_{Y=y} \underbrace{P(X = x|Y = y)}_{\text{Class conditional Distribution of features}} \underbrace{P(Y = y)}_{\text{Distribution of class}}$$

Class conditional Distribution of features

Distribution of class

Bayes Classifier



$$f(X) = \arg\max_{Y=y} \underbrace{P(X = x|Y = y)}_{\text{Class conditional Distribution of features}} \underbrace{P(Y = y)}_{\text{Class distribution}}$$

Class conditional Distribution of features

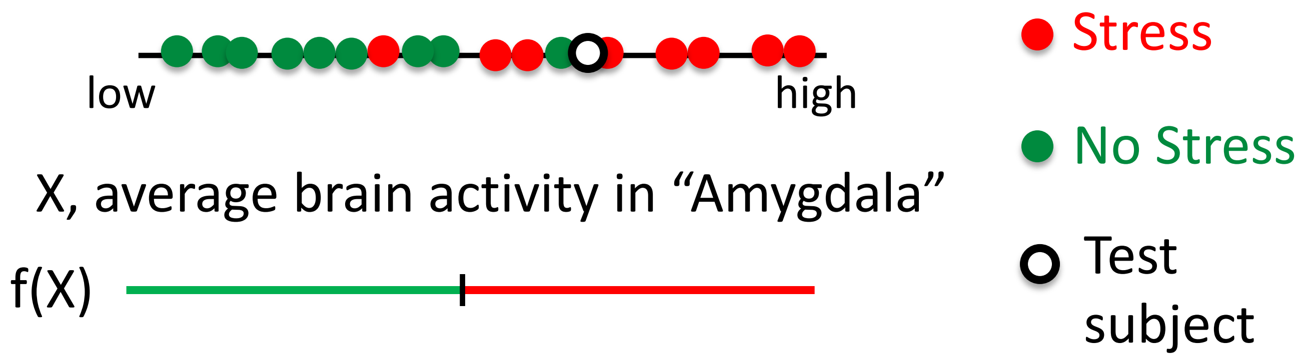
Class distribution

We can now consider appropriate distribution models for the two terms:

Class distribution P(Y=y)

Class conditional distribution of features P(X=x|Y=y)

Modeling class distribution



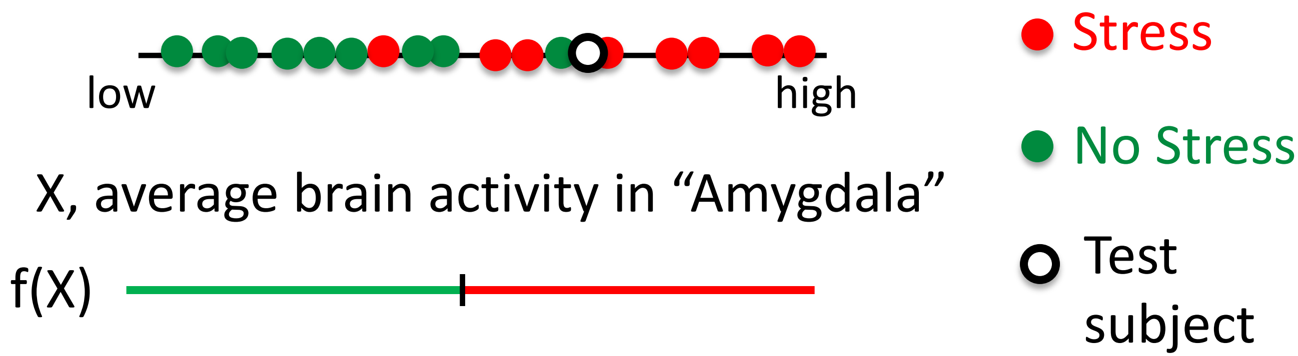
Modeling Class distribution P(Y=y) = Bernoulli(θ)

$P(Y = \text{●}_{red}) = \theta$ $\qquad$ $P(Y = \text{●}_{green}) = 1 - \theta$

Like a coin flip

- How do we model multiple (>2) classes?

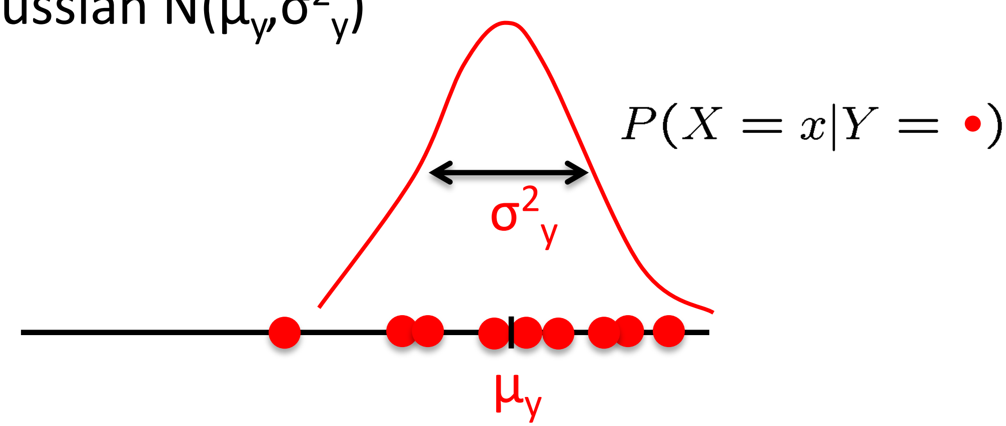
Modeling class conditional distribution of features



Modeling class conditional distribution of feature P(X=x|Y=y)

➢ What distribution would you use?

E.g. P(X=x|Y=y) = Gaussian $N(\mu_y, \sigma^2_y)$



Gaussian Bayes classifier

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

Class conditional

Distribution of features

Class distribution

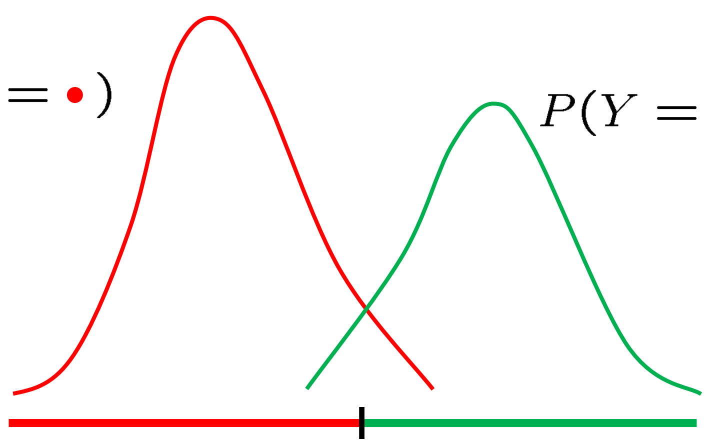
Use MLE/MAP to learn parameters θ, μ_y, Σ_y from data

Gaussian(μ_y, Σ_y)

Bernoulli(θ)

$$P(Y = \bullet)P(X = x|Y = \bullet)$$

$$P(Y = \bullet)P(X = x|Y = \bullet)$$



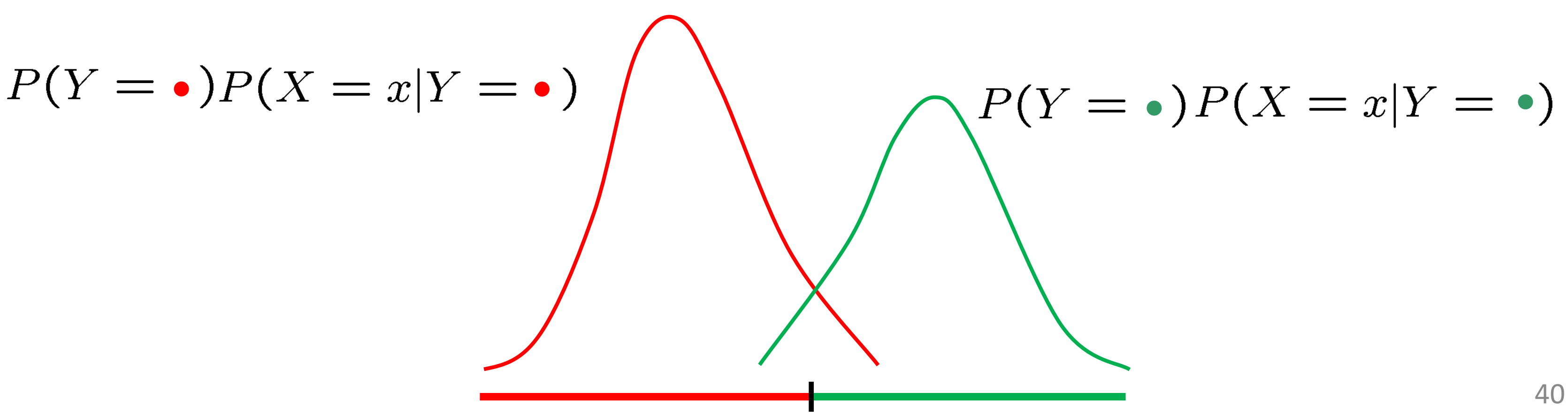
1-dim Gaussian Bayes classifier

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

Class conditional Distribution of features → Gaussian(μ_y, σ^2_y)

Class distribution → Bernoulli(θ)

- What decision boundaries can we get in 1-dim?



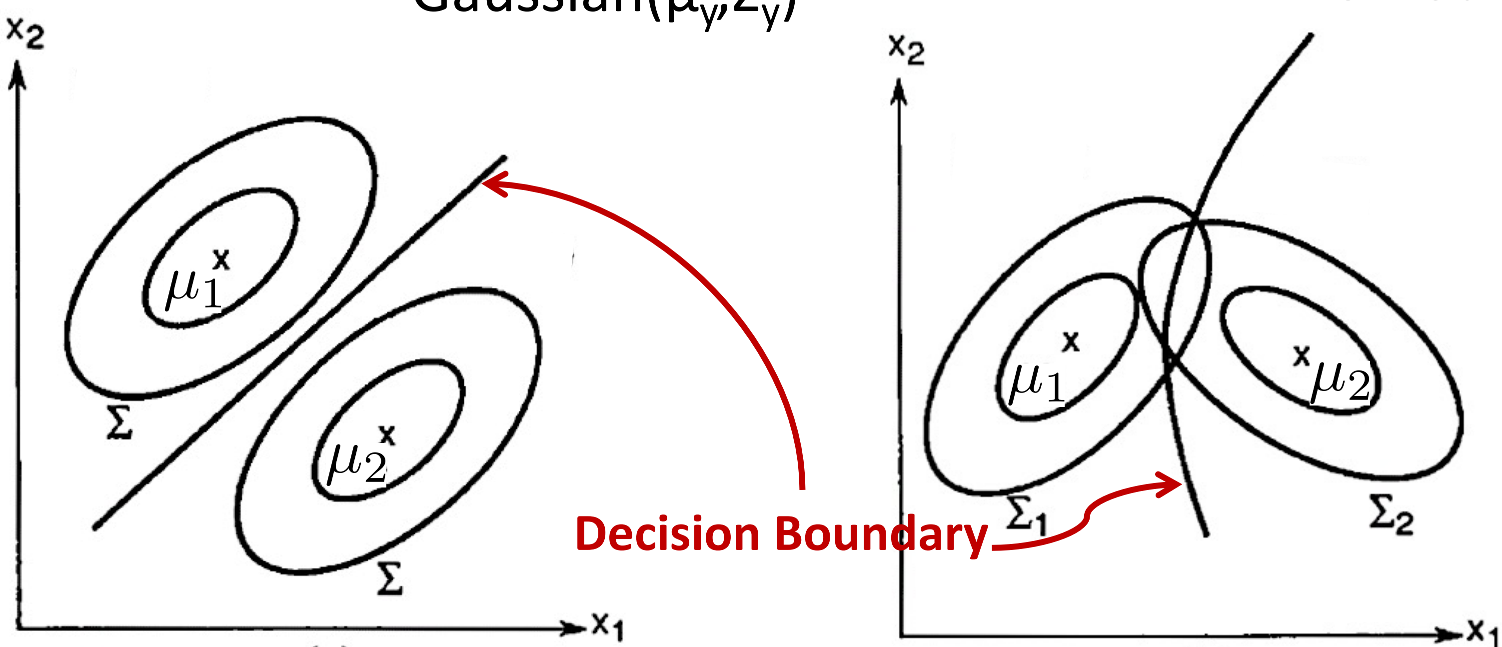
d-dim Gaussian Bayes classifier

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

- What decision boundaries can we get in d-dim?

Class conditional Distribution of features → Gaussian(μ_y,Σ_y)

Class distribution → Bernoulli(θ)



Decision Boundary of Gaussian Bayes

- Decision boundary is set of points x: P(Y=1|X=x) = P(Y=0|X=x)

Compute the ratio

$$1 = \frac{P(Y=1|X=x)}{P(Y=0|X=x)} = \frac{P(X=x|Y=1)P(Y=1)}{P(X=x|Y=0)P(Y=0)}$$

$$= \sqrt{\frac{|\Sigma_0|}{|\Sigma_1|}} \exp\left(-\frac{(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1)}{2} + \frac{(x-\mu_0)^T\Sigma_0^{-1}(x-\mu_0)}{2}\right)\frac{\theta}{1-\theta}$$

In general, this implies a quadratic equation in x. But if $\Sigma_1 = \Sigma_0$, then quadratic part cancels out and decision boundary is linear.

How many parameters do we need to learn (continuous features)?

Class probability:

$P(Y = y) = p_y$ for all y in H, M, L $\qquad$ p_H, p_M, p_L (sum to 1)

K-1 if K labels

Class conditional distribution of features:

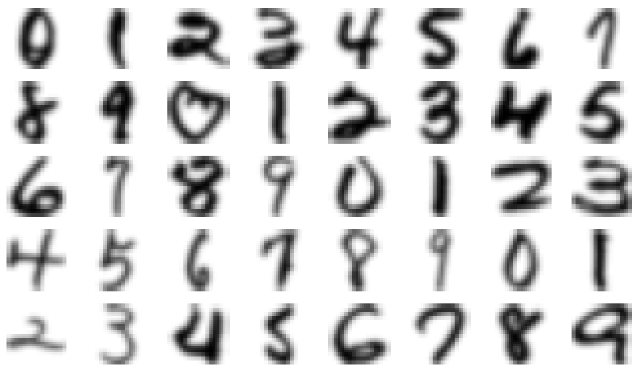
$P(X=x|Y = y) \sim N(\mu_y,\Sigma_y)$ for each y $\qquad$ μ_y – d-dim vector

Σ_y - dxd matrix

Kd + Kd(d+1)/2 = O(Kd²) if d features

Quadratic in dimension d! If d = 256x256 pixels, ~ 13 billion parameters!

How many parameters do we need to learn (discrete features)?



Class probability:

$P(Y = y) = p_y$ for all y in 0, 1, 2, …, 9

$p_0, p_1, \ldots, p_9$ (sum to 1)

K-1 if K labels

Class conditional distribution of (binary) features:

$P(X=x|Y = y)$ ~ For each label y, maintain probability table with 2^d-1 entries

$K(2^d – 1)$ if d binary features

Exponential in dimension d!

What's wrong with too many parameters?

- How many training data needed to learn one parameter (bias of a coin)?

- Need lots of training data to learn the parameters!
  - Training data > number of (independent) parameters

Naïve Bayes Classifier

- Bayes Classifier with additional "naïve" assumption:
  - Features are independent given class:

$$X = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$$

$$P(X_1, X_2|Y) = P(X_1|X_2, Y)P(X_2|Y)$$
$$= P(X_1|Y)P(X_2|Y)$$

  - More generally:

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

$$X = \begin{bmatrix} X_1 \\ X_2 \\ \dots \\ X_d \end{bmatrix}$$

- If conditional independence assumption holds, NB is optimal classifier! But worse otherwise.

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

Naïve Bayes Classifier

- Bayes Classifier with additional "naïve" assumption:
  – Features are independent given class:

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

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

- How many parameters now?

How many parameters do we need to learn (continuous features)?

Class probability:

$P(Y = y) = p_y$ for all y in H, M, L $\quad$ p_H, p_M, p_L (sum to 1)

K-1 if K labels

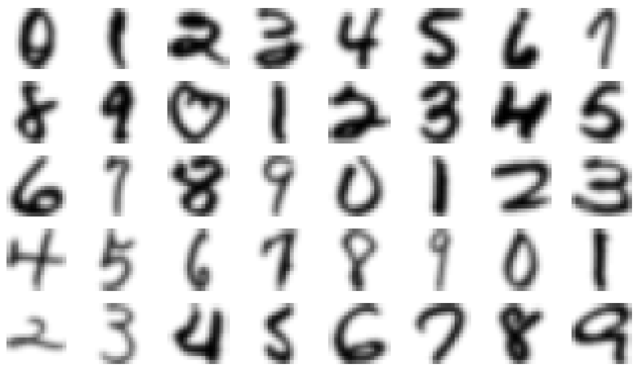
Class conditional distribution of features (using Naïve Bayes assumption):

$P(X_i = x_i | Y = y) \sim N(\mu^{(y)}_i, \sigma^2_i{}^{(y)})$ for each y and each pixel i

2Kd if d features

Linear instead of Quadratic in dimension d!

How many parameters do we need to learn (discrete features)?



Class probability:

$P(Y = y) = p_y$ for all y in 0, 1, 2, …, 9

$p_0, p_1, \ldots, p_9$ (sum to 1)

K-1 if K labels

Class conditional distribution of (binary) features:

$P(X_i = x_i | Y = y)$ – one probability value for each y, pixel i

Kd if d binary features

Linear instead of Exponential in dimension d!

Naïve Bayes Classifier

- Bayes Classifier with additional "naïve" assumption:
  - Features are independent given class:

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

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

- Has fewer parameters, and hence requires fewer training data, even though assumption may be violated in practice

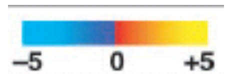
Learned Gaussian Naïve Bayes Model

Means for P(BrainActivity | WordCategory)

[Mitchell et al.03]

Pairwise classification accuracy: 85%

People words



Animal words