Machine Learning

4771

Instructor: Itsik Pe'er

Reminder: Ensembles

Many weak classifiers → a powerful one




(Classification) models

Parametric

Estimate parameters of the distribution of the data

Non-parametric

Reason about data assuming unknown distribution

(Classification) models

Parametric

Estimate parameters of the distribution of the data

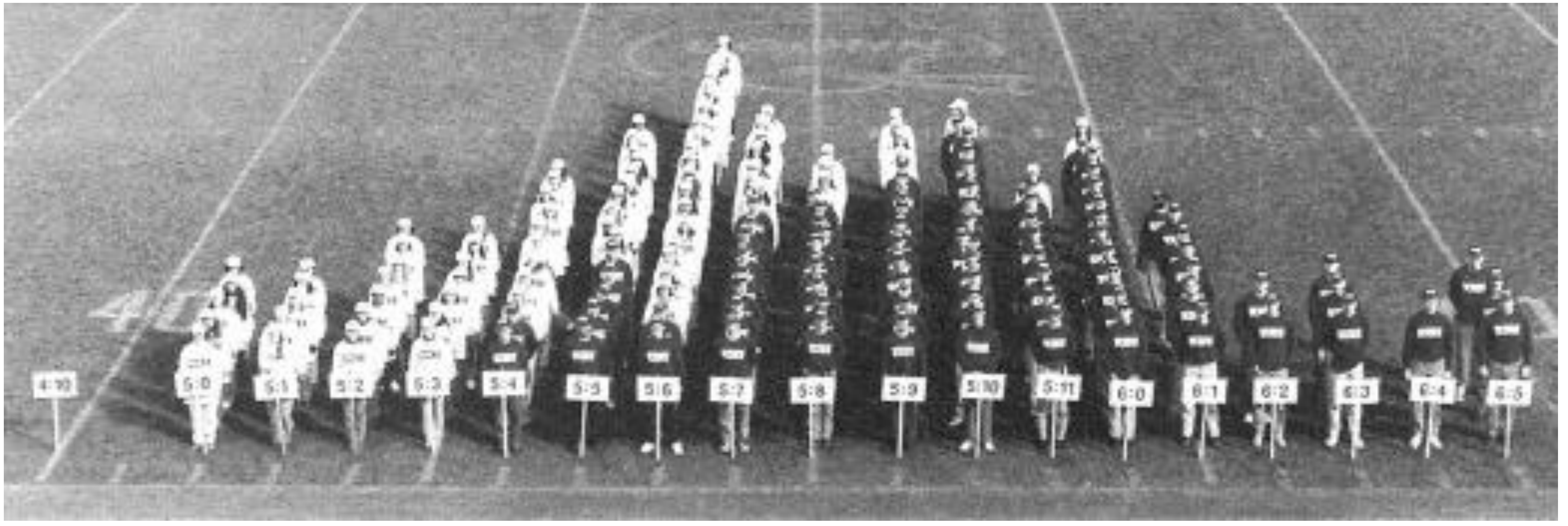
- Logistic regression
- Least squares regression

Non-parametric

Reason about data assuming unknown distribution

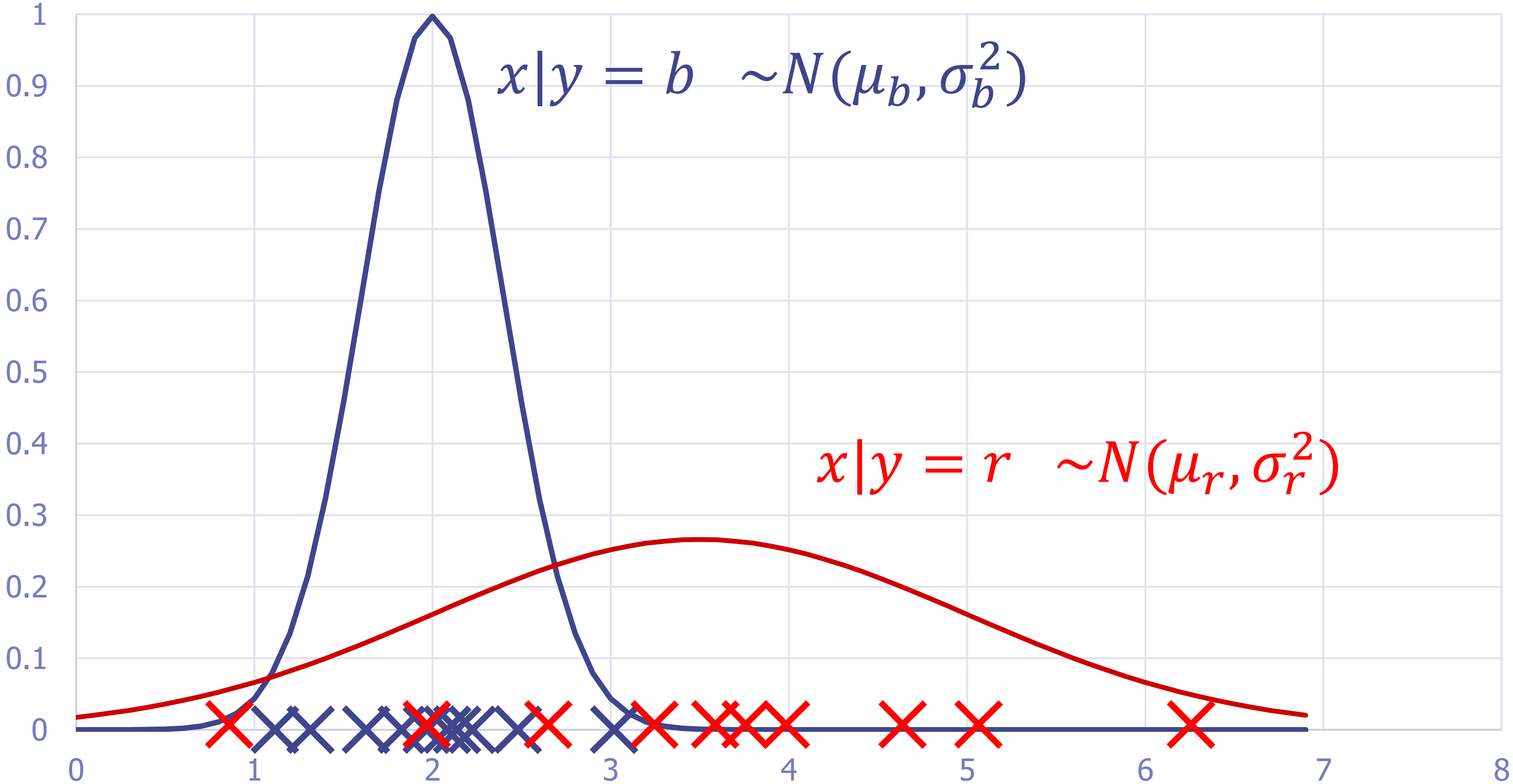
- Nearest neighbors
- Decision trees
- SVM
- RBF regression

Bayesian Classification



If distribution of each cluster known/inferable:
Can be used for classification

Bayesian Classification



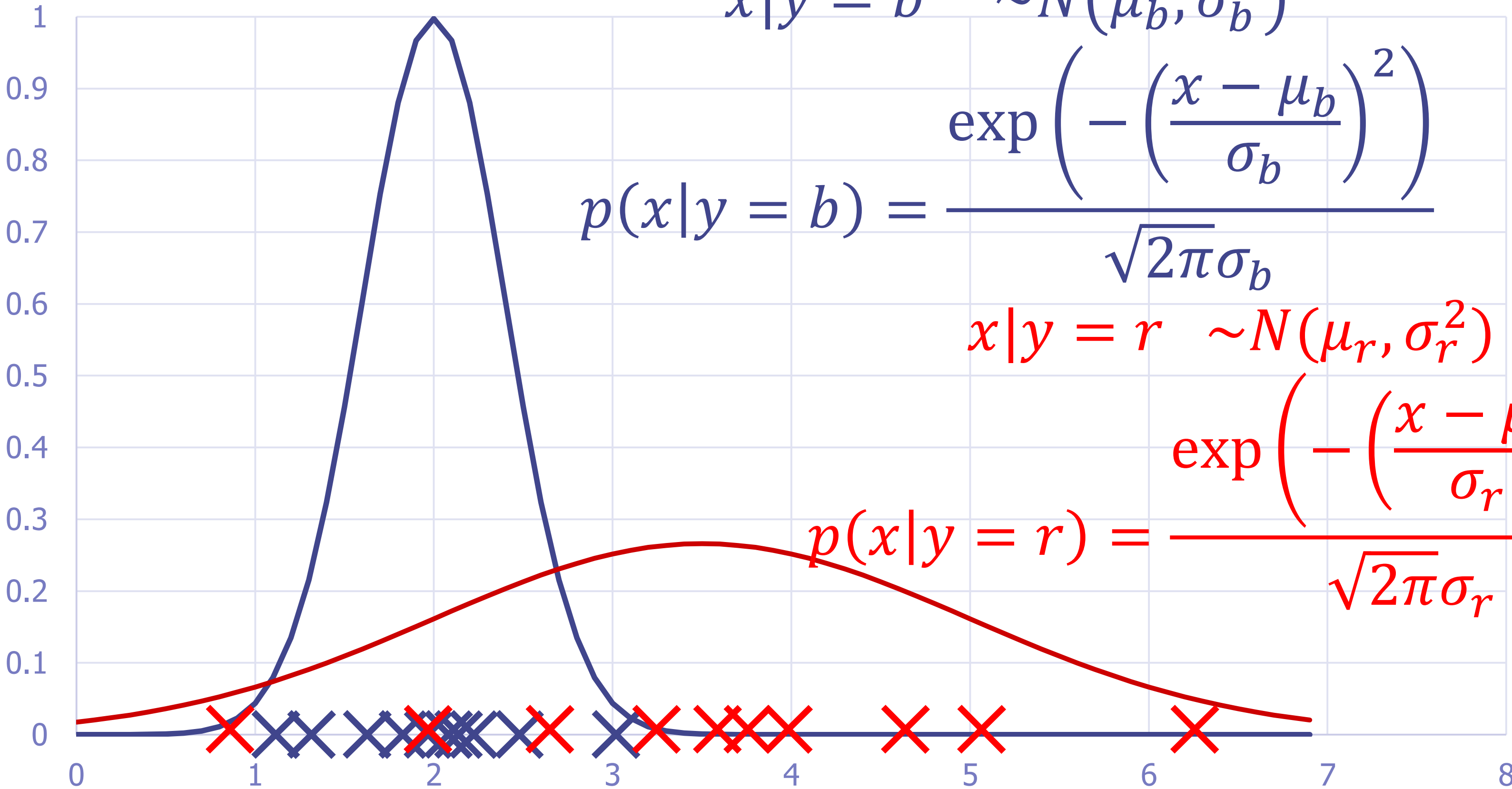
Bayesian Classification

$$x|y = b \quad \sim N\left(\mu_b, \sigma_b^2\right)$$

$$p(x|y = b) = \frac{\exp\left(-\left(\frac{x-\mu_b}{\sigma_b}\right)^2\right)}{\sqrt{2\pi}\sigma_b}$$

$$x|y = r \quad \sim N\left(\mu_r, \sigma_r^2\right)$$

$$p(x|y = r) = \frac{\exp\left(-\left(\frac{x-\mu_r}{\sigma_r}\right)^2\right)}{\sqrt{2\pi}\sigma_r}$$



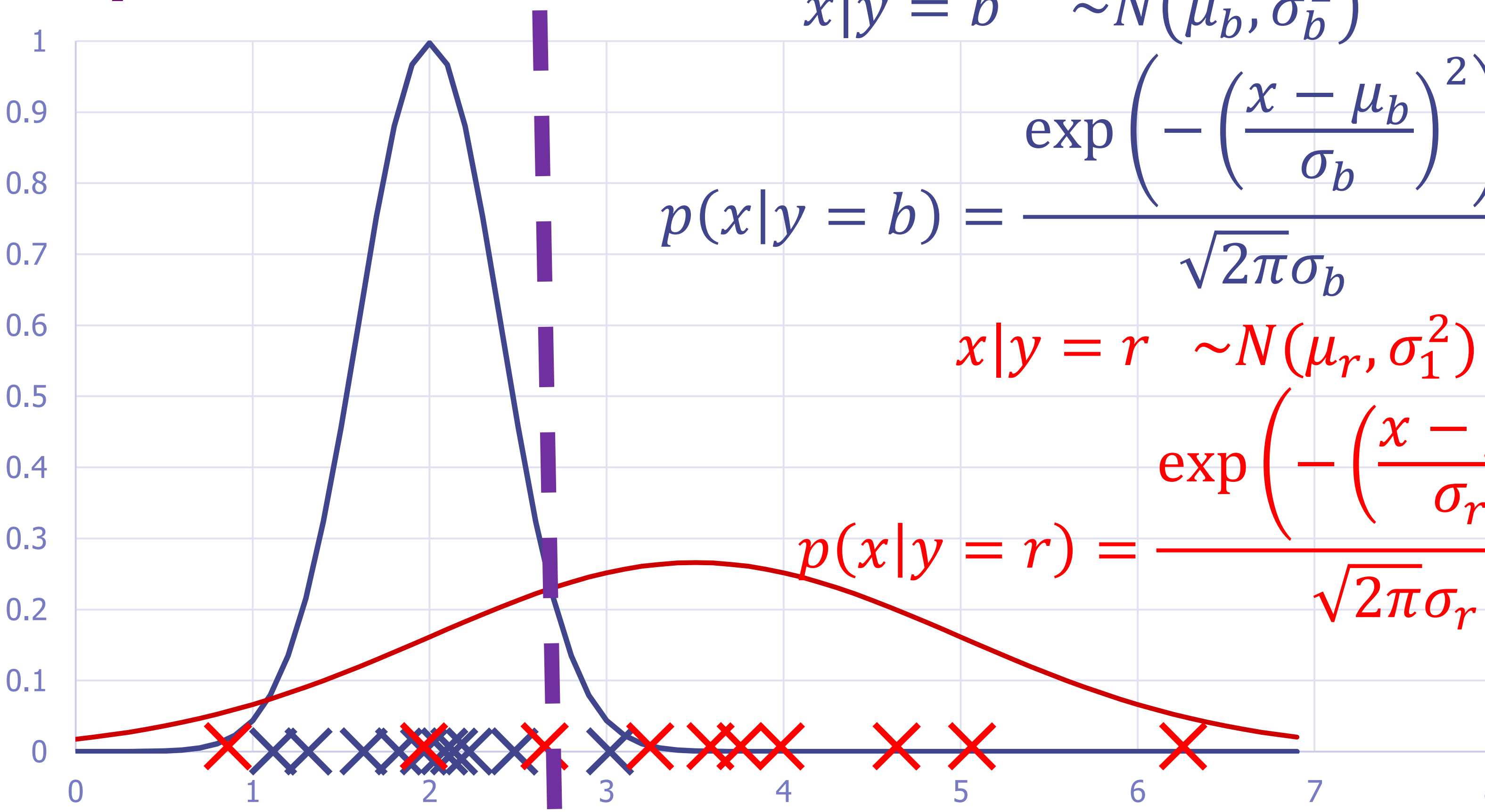
Bayesian Classification

$$x|y=b \quad \sim N\left(\mu_b,\sigma_b^2\right)$$

$$p(x|y=b)=\frac{\exp\left(-\left(\frac{x-\mu_b}{\sigma_b}\right)^2\right)}{\sqrt{2\pi}\sigma_b}$$

$$x|y=r \quad \sim N\left(\mu_r,\sigma_1^2\right)$$

$$p(x|y=r)=\frac{\exp\left(-\left(\frac{x-\mu_r}{\sigma_r}\right)^2\right)}{\sqrt{2\pi}\sigma_r}$$



$$p(y_i=b|x_i)=\frac{p(x_i|y_i=b)p(y_i=b)}{\sum_{y=\{b,r\}}p(x_i|y_i=y)p(y_i=y)}$$

If uniform prior: $p(y_i=b|x_i)=Cp(x_i|y_i=b)$ so max likelihood

High Dimensional Data

- Text classification: simplest model

$\in \begin{cases} religion \\ politics \end{cases}$

- 10^5-10^6 words in English
- Each document is D=10^5 dimensional binary vector $\vec{x}_i$
- Each dimension is a word, set to 1 if word in the document

Dim1: "we" = 1
Dim2: "hello" = 0
Dim3: "people" = 1
Dim4: "justice" = 1
...

$$p(\vec{x}) = p\left(\vec{x}(1), \vec{x}(2), \dots, \vec{x}(D)\right)$$

High Dimensional Data

- Text classification: simplest model

$\in \begin{cases} religion \\ politics \end{cases}$

- 10^5-10^6 words in English
- Each document is D=10^5 dimensional binary vector $\vec{x}_i$
- Each dimension is a word, set to 1 if word in the document

Dim1: "we" = 1
Dim2: "hello" = 0
Dim3: "people" = 1
Dim4: "justice" = 1
...

$$p(\vec{x}) = p\left(\vec{x}(1), \vec{x}(2), \dots, \vec{x}(D)\right)$$

- Each 1 dimensional $\vec{x}(d)$ is a Bernoulli variable
- $\vec{x}$ is multivariate Bernoulli

High Dimensional Data

$\in \begin{cases} religion \\ politics \end{cases}$

- Text classification: simplest model
- 10^5-10^6 words in English
- Each document is D=10^5 dimensional binary vector $\vec{x}_i$
- Each dimension is a word, set to 1 if word in the document

Dim1: "we" = 1
Dim2: "hello" = 0
Dim3: "people" = 1
Dim4: "justice" = 1
...

- Naïve Bayes: assumes each word is independent

$$p(\vec{x}) = p\big(\vec{x}(1), \vec{x}(2), \dots, \vec{x}(D)\big) = \prod_{d=1}^{D} p\big(\vec{x}(d)\big)$$

$$= \prod_{d=1}^{D} \vec{\theta}(d)^{\vec{x}(d)} \left(1 - \vec{\theta}(d)\right)^{1-\vec{x}(d)}$$

Text: Naïve Bayes

- Maximum likelihood: assume we have several IID vectors
- Have N documents, each a 100,000 dimension binary vector
- Each dimension is a word, set to 1 if word in the document

			$\vec{x}_1$	$\vec{x}_2$	$\vec{x}_3$	$\vec{x}_4$
Dim1:	**“the”**	**=**	**1**	**0**	**1**	**1**
Dim2:	**“hello”**	**=**	**0**	**1**	**0**	**1**
Dim3:	**“and”**	**=**	**1**	**1**	**0**	**1**
Dim4:	**“happy”**	**=**	**1**	**0**	**0**	**1**

- Likelihood=

Text: Naïve Bayes

- Maximum likelihood: assume we have several IID vectors
- Have N documents, each a 100,000 dimension binary vector
- Each dimension is a word, set to 1 if word in the document

			$\vec{x}_1$	$\vec{x}_2$	$\vec{x}_3$	$\vec{x}_4$
Dim1:	**“the”**	**=**	**1**	**0**	**1**	**1**
Dim2:	**“hello”**	**=**	**0**	**1**	**0**	**1**
Dim3:	**“and”**	**=**	**1**	**1**	**0**	**1**
Dim4:	**“happy”**	**=**	**1**	**0**	**0**	**1**

- Likelihood= $\prod_{i=1}^{N} p\left(\vec{x}_i \middle| \vec{\theta}\right) = \prod_{i=1}^{N} \prod_{d=1}^{D} \vec{\theta}(d)^{\vec{x}_i(d)} \left(1 - \vec{\theta}(d)\right)^{\left(1-\vec{x}_i(d)\right)}$
- Max likelihood solution:

Text: Naïve Bayes

- Likelihood$= \prod_{i=1}^{N} p\left(\vec{x}_i \middle| \vec{\theta}\right) = \prod_{i=1}^{N} \prod_{d=1}^{D} \vec{\theta}(d)^{\vec{x}_i(d)} \left(1 - \vec{\theta}(d)\right)^{\left(1 - \vec{x}_i(d)\right)}$
- Max likelihood solution: for each word d count $\vec{\theta}(d) = \frac{N_d}{N}$
- Assuming beta-prior:$p\left(\vec{\theta}(d)\right) \sim Beta(1,1)$

posterior: $p\left(\vec{\theta}(d) | data\right) \sim Beta(N_d + 1, (N - N_d) + 1)$

- $\mathrm{EAP}(\vec{\theta}(d)) = \frac{N_d + 1}{N + 2}$

Text: Naïve Bayes

- Likelihood$= \prod_{i=1}^{N} p\left(\vec{x}_i \middle| \vec{\theta}\right) = \prod_{i=1}^{N} \prod_{d=1}^{D} \vec{\theta}(d)^{\vec{x}_i(d)} \left(1 - \vec{\theta}(d)\right)^{\left(1-\vec{x}_i(d)\right)}$
- Max likelihood solution: for each word d count $\vec{\theta}(d) = \frac{N_d}{N}$
- Assuming (conjugate) beta-prior:$p\left(\vec{\theta}(d)\right) \sim Beta(\alpha, \beta)$

posterior: $p\left(\vec{\theta}(d)|data\right) \sim Beta(N_d + \alpha, (N - N_d) + \beta)$

- $\mathrm{EAP}(\vec{\theta}(d)) = \frac{N_d+\alpha}{N+\alpha+\beta}$
- To classify new document $\vec{x}_{new}$, build two models $\vec{\theta}_{religion}, \vec{\theta}_{politics}$

  Compare: $prediction = argmax_{y\in\{religion,politics\}} p\left(\vec{x}_{new} \middle| \vec{\theta}_y\right)$

Text: Naïve Bayes

- Likelihood= $\prod_{i=1}^{N} p\left(\vec{x}_i \middle| \vec{\theta}\right) = \prod_{i=1}^{N} \prod_{d=1}^{D} \vec{\theta}(d)^{\vec{x}_i(d)} \left(1-\vec{\theta}(d)\right)^{\left(1-\vec{x}_i(d)\right)}$
- Max likelihood solution: for each word d count $\vec{\theta}(d) = \frac{N_d}{N}$
- Assuming (conjugate) beta-prior: $p\left(\vec{\theta}(d)\right) \sim Beta(\alpha, \beta)$

posterior: $p\left(\vec{\theta}(d) | data\right) \sim Beta(N_d + \alpha, (N - N_d) + \beta)$

- $\mathrm{EAP}(\vec{\theta}(d)) = \frac{N_d+\alpha}{N+\alpha+\beta}$
- To classify new document $\vec{x}_{new}$, build two models $\vec{\theta}_{religion}, \vec{\theta}_{politics}$

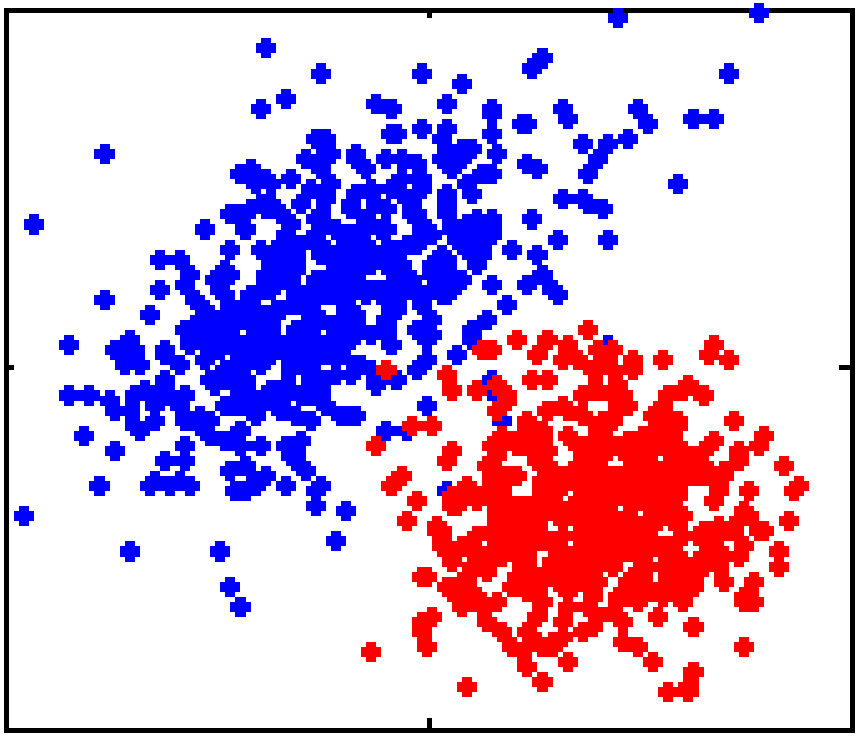
Compare: $prediction = argmax_{y \in \{religion, politics\}} \log p\left(\vec{x}_{new} \middle| \vec{\theta}_y\right) =$

$$argmax_y \sum_{d=1}^{D} \left(\vec{x}_{new}(d) \log \vec{\theta}_y(d) + \left(1 - \vec{x}_{new}(d)\right) \log\left(1 - \vec{\theta}_y(d)\right)\right)$$

$$= argmax_y \sum_{d=1}^{D} \vec{x}_{new}(d) \log \frac{\vec{\theta}_y(d)}{1 - \vec{\theta}_y(d)}$$

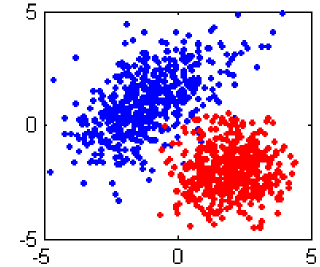
Handling Dependencies: Two 2D Gaussians

Height



pitch

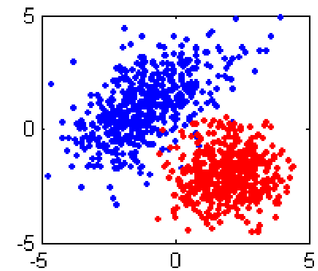
Classification with Gaussians



- Have two classes, each with their own Gaussian:

$$\{(x_1, y_1), \ldots, (x_N, y_N)\} \quad x \in \boldsymbol{R}^D, y \in \{0,1\}$$

Classification with Gaussians



- Have two classes, each with their own Gaussian:

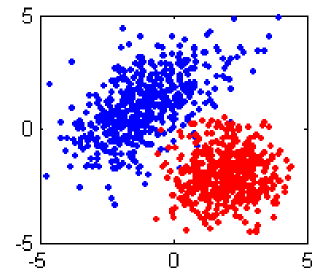
$$\{(x_1, y_1), \ldots, (x_N, y_N)\} \quad x \in \boldsymbol{R}^D, y \in \{0,1\}$$

- Given parameters $\theta = \{\alpha, \mu_0, \Sigma_0\,, \mu_1, \Sigma_1\}$ we can generate iid data from $p(x, y|\theta) = p(y|\theta)p(x|y, \theta)$ by:

  1) flipping a coin to get y via Bernoulli $p(y|\theta) = \alpha^y(1-\alpha)^{1-y}$
  2) sampling an x from y'th Gaussian $p(x|y, \theta) = N\left(\mu_y\,, \Sigma_y\right)$

- Recover parameters from data using maximum likelihood $l(\theta)$

Classification with Gaussians



- Have two classes, each with their own Gaussian:

$$\{(x_1, y_1), \ldots, (x_N, y_N)\} \quad x \in \boldsymbol{R}^D, y \in \{0,1\}$$

- Given parameters $\theta = \{\alpha, \mu_0, \Sigma_0\,, \mu_1, \Sigma_1\}$ we can generate iid data from $p(x, y|\theta) = p(y|\theta)p(x|y, \theta)$ by:

  1) flipping a coin to get y via Bernoulli $p(y|\theta) = \alpha^y(1-\alpha)^{1-y}$
  2) sampling an x from y'th Gaussian $p(x|y, \theta) = N\left(\mu_y\,, \Sigma_y\right)$

- Recover parameters from data using maximum likelihood $l(\theta)$

$$\log p(data|\theta) = \sum_{i=1}^{N} \log p(x_i, y_i|\theta) = \sum_{i=1}^{N} \log p(y_i|\theta) + \sum_{i=1}^{N} p(x_i|y_i, \theta)$$

$$= \sum_{i=1}^{N} \log p(y_i|\alpha) + \sum_{y_i \in 0}^{N} p(x_i|\mu_0, \Sigma_0) + \sum_{y_i \in 1}^{N} p(x_i|\mu_1, \Sigma_1)$$

Classification with Gaussians

- Max Likelihood can be done separately for the 3 terms

$$l(\theta) = \sum_{i=1}^{N} \log p(y_i|\alpha) + \sum_{y_i \in 0}^{N} p(x_i|\mu_0, \Sigma_0) + \sum_{y_i \in 1}^{N} p(x_i|\mu_1, \Sigma_1)$$

Classification with Gaussians

- Max Likelihood can be done separately for the 3 terms

$l(\theta) = \sum_{i=1}^{N} \log p(y_i|\alpha) + \sum_{y_i \in 0}^{N} p(x_i|\mu_0, \Sigma_0) + \sum_{y_i \in 1}^{N} p(x_i|\mu_1, \Sigma_1)$

- Count # of pos & neg examples (class prior): $\alpha = \frac{N_1}{N_0 + N_1}$
- Get mean & cov of negatives and mean & cov of positives:

$$\mu_0 = \bar{x}\Big|_{y_i=0} = \frac{1}{N_0}\sum_{y_i=0} x_i \quad \Sigma_0 = \frac{1}{N_0}\sum_{y_i=0}(x_i - \mu_0)(x_i - \mu_0)^T$$

$$\mu_1 = \bar{x}\Big|_{y_i=1} = \frac{1}{N_1}\sum_{y_i=1} x_i \quad \Sigma_1 = \frac{1}{N_1}\sum_{y_i=1}(x_i - \mu_1)(x_i - \mu_1)^T$$

Classification with Gaussians

- Max Likelihood can be done separately for the 3 terms

$l(\theta) = \sum_{i=1}^{N} \log p(y_i|\alpha) + \sum_{y_i \in 0}^{N} p(x_i|\mu_0, \Sigma_0) + \sum_{y_i \in 1}^{N} p(x_i|\mu_1, \Sigma_1)$

- Count # of pos & neg examples (class prior): $\alpha = \frac{N_1}{N_0+N_1}$
- Get mean & cov of negatives and mean & cov of positives:

$$\mu_0 = \bar{x}_0 = \bar{x}\Big|_{y_i=0} = \frac{1}{N_0}\sum_{y_i=0} x_i \quad \Sigma_0 = \frac{1}{N_0}\sum_{y_i=0}(x_i-\mu_0)(x_i-\mu_0)^T$$

$$\mu_1 = \bar{x}_1 = \bar{x}\Big|_{y_i=1} = \frac{1}{N_1}\sum_{y_i=1} x_i \quad \Sigma_1 = \frac{1}{N_1}\sum_{y_i=1}(x_i-\mu_1)(x_i-\mu_1)^T$$

Posterior μ_y if (conjugate) prior is $N(\mu_p, \Sigma_p)$ and known Σ_y:

$\mu_y \sim N(\mu_{post}, \Sigma_{post})$

where : $\mu_{post} = \Sigma_{post}\left(\Sigma_p^{-1} + N\Sigma_y \bar{x}_y\right), \Sigma_{post} = \left(\Sigma_p^{-1} + N\Sigma_y^{-1}\right)^{-1}$

Classification with Gaussians

- Max Likelihood can be done separately for the 3 terms

$$l(\theta) = \sum_{i=1}^{N} \log p(y_i|\alpha) + \sum_{y_i \in 0}^{N} p(x_i|\mu_0, \Sigma_0) + \sum_{y_i \in 1}^{N} p(x_i|\mu_1, \Sigma_1)$$

- Given (x,y) pair, can now compute likelihood $p(x,y)$
- Bayesian classification:
  - Without x, can compute prior guess for y: $p(y)$
  - Give me x, want y, I need posterior $p(y|x)$
  - Bayes optimal decision: $\hat{y} = argmax_{y \in \{0,1\}} p(y|x)$
  - Optimal if we have true probability

Deciding between Gaussians

$$p(y=1|x) = \frac{p(x, y=1)}{p(x, y=0) + p(x, y=1)}$$
$$= \frac{\alpha N(x|\mu_1, \Sigma_1)}{(1-\alpha)N(x|\mu_0, \Sigma_0) + \alpha N(x|\mu_1, \Sigma_1)}$$

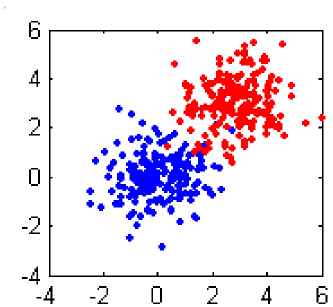
Mahalanobis Distance

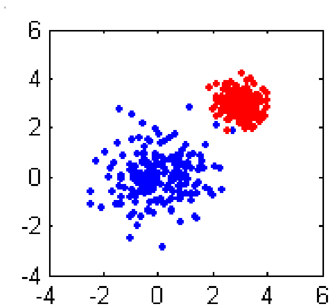
$$p(y=1|x) = \frac{p(x,y=1)}{p(x,y=0)+p(x,y=1)}$$

$$= \frac{\alpha N(x|\mu_1,\Sigma_1)}{(1-\alpha)N(x|\mu_0,\Sigma_0)+\alpha N(x|\mu_1,\Sigma_1)}$$

$$\log p(y_{new}=y|x_{new}) = C - \underbrace{\left(x_{new}-\mu_y\right)^T \Sigma_y^{-1}\left(x_{new}-\mu_y\right)}_{\text{Mahalanobis Distance}\left(x_{new},\mu_y\right)}$$

$$C = C_{\alpha,x_{new}} + C_{\alpha,\Sigma_y}$$





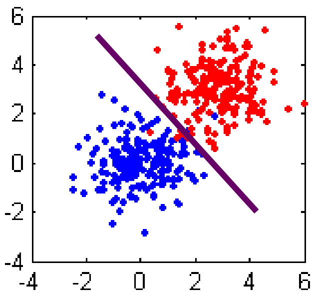
Linear or Quadratic Decisions

- Example cases, plotting decision boundary when = 0.5

$$p(y=1|x) = \frac{p(x,y=1)}{p(x,y=0)+p(x,y=1)}$$
$$= \frac{\alpha N(x|\mu_1,\Sigma_1)}{(1-\alpha)N(x|\mu_0,\Sigma_0)+\alpha N(x|\mu_1,\Sigma_1)}$$

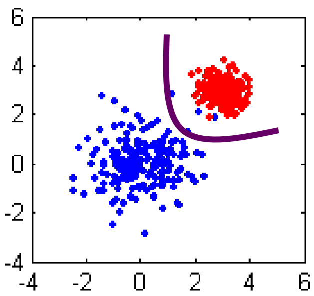
- If covariances are equal:

  linear decision



- If covariances are different:

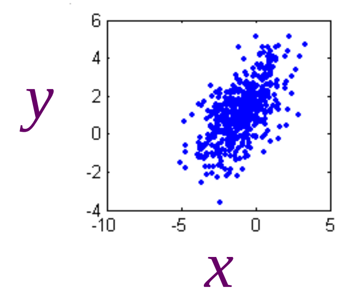
  quadratic decision



Summary

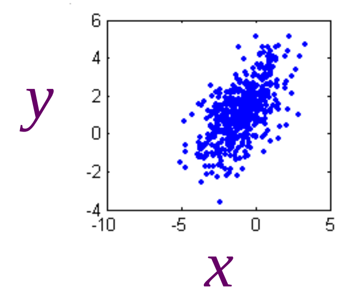
- Naïve Bayes:
  - Assuming independence of features
- Classifying Gaussians:
  - Bayesian
  - Mahalanobis Distance

More Fun with Gaussians



- Have input and output, each Gaussian:

$\{(x_1, y_1), \ldots, (x_N, y_N)\}\ x \in \boldsymbol{R}^{D_x}, y \in \boldsymbol{R}^{D_y}, D = D_x + D_y$



Multiplying Gaussians

- Have input and output, each Gaussian:

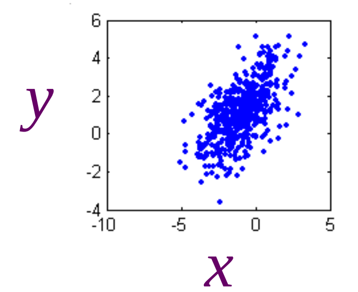
$$\{(x_1, y_1), \ldots, (x_N, y_N)\} \ x \in \boldsymbol{R}^{D_x}, y \in \boldsymbol{R}^{D_y}, D = D_x + D_y$$

concatenate $z_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}$

$$p(z|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(z-\mu)^T \Sigma^{-1} (z-\mu)\right)$$

- Maximum Likelihood

Regression with Gaussians



- Have input and output, each Gaussian:

$\{(x_1, y_1), \ldots, (x_N, y_N)\}\ x \in \boldsymbol{R}^{D_x}, y \in \boldsymbol{R}^{D_y}, D = D_x + D_y$

concatenate $z_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}$

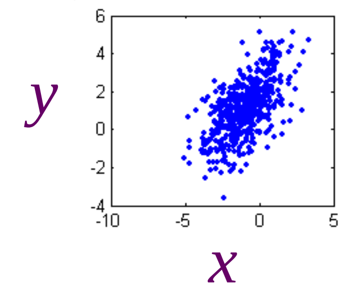
$$p(z|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(z-\mu)^T \Sigma^{-1}(z-\mu)\right)$$

- Maximum Likelihood is as usual for a multivariate Gaussian

$$\mu = \frac{1}{N}\sum_i^N z_i \quad \Sigma = \frac{1}{N}\sum_i^N (z_i - \mu)^T (z_i - \mu)$$

- Bayes optimal decision:

Regression with Gaussians



- Have input and output, each Gaussian:

$\{(x_1, y_1), \dots, (x_N, y_N)\}\ x \in \boldsymbol{R}^{D_x}, y \in \boldsymbol{R}^{D_y}, D = D_x + D_y$

concatenate $z_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}$

$$p(z|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(z-\mu)^T \Sigma^{-1} (z-\mu)\right)$$

- Maximum Likelihood is as usual for a multivariate Gaussian

$$\mu = \frac{1}{N}\sum_{i}^{N} z_i \quad \Sigma = \frac{1}{N}\sum_{i}^{N} (z_i - \mu)^T (z_i - \mu)$$

- Bayes optimal decision: $\hat{y} = argmax_{y \in \boldsymbol{R}}\, p(y|x)$

- Or we can use: $\hat{y} = E_{p(y|x)}\{y\}$

- Have joint, need conditional: $p(y|x) = \frac{p(x,y)}{p(x)} = \frac{p(x,y)}{\int_y p(x,y)}$

Gaussian Marginals/Conditionals

- Conditional & marginal from joint: $p(y|x) = \frac{p(x,y)}{p(x)} = \frac{p(x,y)}{\int_y p(x,y)}$
- Gaussian: $p(z|\mu,\Sigma) = \frac{1}{(2\pi)^{D/2}\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(z-\mu)^T\Sigma^{-1}(z-\mu)\right)$

$p(x,y) =$

Gaussian Marginals/Conditionals

$$p(x,y) = \frac{\exp\left(-\frac{1}{2}\left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}\right)^T \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix}^{-1} \left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}\right)\right)}{(2\pi)^{D/2}\sqrt{|\Sigma|}} =$$

Gaussian Marginals/Conditionals

$$p(x,y) = \frac{\exp\left(-\frac{1}{2}\left(\begin{bmatrix}x\\y\end{bmatrix}-\begin{bmatrix}\mu_x\\\mu_y\end{bmatrix}\right)^T \begin{bmatrix}\Sigma_{xx} & \Sigma_{xy}\\\Sigma_{yx} & \Sigma_{yy}\end{bmatrix}^{-1}\left(\begin{bmatrix}x\\y\end{bmatrix}-\begin{bmatrix}\mu_x\\\mu_y\end{bmatrix}\right)\right)}{(2\pi)^{D/2}\sqrt{|\Sigma|}} = F_{xx}F_{xy}F_{yy}$$

$$\begin{bmatrix}\Sigma_{xx} & \Sigma_{xy}\\\Sigma_{yx} & \Sigma_{yy}\end{bmatrix}^{-1} = \begin{bmatrix}\Sigma_{xx}^{-1} & M\\M^T & \Sigma_{yy}^{-1}\end{bmatrix}$$

$$F_{xx} = \frac{\exp\left(-\frac{(x-\mu_x)^T{\Sigma_{xx}}^{-1}(x-\mu_x)}{2}\right)}{(2\pi)^{D_x/2}\sqrt{|\Sigma_{xx}|}}$$

$$F_{yy} = \frac{\exp\left(-\frac{(y-\mu_y)^T{\Sigma_{yy}}^{-1}(y-\mu_y)}{2}\right)}{(2\pi)^{D_y/2}\sqrt{|\Sigma_{yy}|}}$$

Gaussian Marginals/Conditionals

$$p(x,y) = \frac{\exp\left(-\frac{1}{2}\left(\begin{bmatrix}x\\y\end{bmatrix}-\begin{bmatrix}\mu_x\\\mu_y\end{bmatrix}\right)^T\begin{bmatrix}\Sigma_{xx} & \Sigma_{xy}\\\Sigma_{yx} & \Sigma_{yy}\end{bmatrix}^{-1}\left(\begin{bmatrix}x\\y\end{bmatrix}-\begin{bmatrix}\mu_x\\\mu_y\end{bmatrix}\right)\right)}{(2\pi)^{D/2}\sqrt{|\Sigma|}} = F_{xx}F_{xy}F_{yy} \;;$$

$$\begin{bmatrix}\Sigma_{xx} & \Sigma_{xy}\\\Sigma_{yx} & \Sigma_{yy}\end{bmatrix}^{-1} = \begin{bmatrix}\Sigma_{xx}^{-1} & M\\M^T & \Sigma_{yy}^{-1}\end{bmatrix} \quad M = -\Sigma_{xx}^{-1}\Sigma_{xy}\Sigma_{yy}^{-1}$$

$$F_{xx} = \frac{\exp\left(-\frac{(x-\mu_x)^T\Sigma_{xx}{}^{-1}(x-\mu_x)}{2}\right)}{(2\pi)^{Dx/2}\sqrt{|\Sigma_{xx}|}};\; F_{yy} = \frac{\exp\left(-\frac{(y-\mu_y)^T\Sigma_{yy}{}^{-1}(y-\mu_y)}{2}\right)}{(2\pi)^{Dy/2}\sqrt{|\Sigma_{yy}|}}$$

$$F_{xy} = \sqrt{\frac{|\Sigma_{xx}||\Sigma_{yy}|}{|\Sigma|}}\exp\left(-(x-\mu_x)^T M(y-\mu_y)\right)$$

Gaussian Marginals/Conditionals

- Conditional & marginal from joint: $p(y|x) = \frac{p(x,y)}{p(x)} = \frac{p(x,y)}{\int_y p(x,y)}$
- Gaussian: $p(z|\mu,\Sigma) = \frac{1}{(2\pi)^{D/2}\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(z-\mu)^T \Sigma^{-1}(z-\mu)\right)$

$$p(x,y) = \frac{\exp\left(-\frac{1}{2}\left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}\right)^T \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix}^{-1} \left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}\right)\right)}{(2\pi)^{D/2}\sqrt{|\Sigma|}}$$

$$p(x) = \frac{1}{(2\pi)^{D_x/2}\sqrt{|\Sigma_{xx}|}} \exp\left(-\frac{1}{2}(x-\mu_x)^T \Sigma_{xx}^{-1}(x-\mu_x)\right) = N(\mu_x, \Sigma_{xx})$$

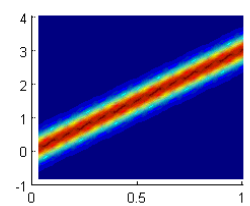
Gaussian Marginals/Conditionals

- Conditional & marginal from joint: $p(y|x) = \frac{p(x,y)}{p(x)} = \frac{p(x,y)}{\int_y p(x,y)}$
- Gaussian: $p(z|\mu,\Sigma) = \frac{1}{(2\pi)^{D/2}\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(z-\mu)^T\Sigma^{-1}(z-\mu)\right)$

$$p(x,y) = \frac{\exp\left(-\frac{1}{2}\left(\begin{bmatrix}x\\y\end{bmatrix} - \begin{bmatrix}\mu_x\\\mu_y\end{bmatrix}\right)^T \begin{bmatrix}\Sigma_{xx} & \Sigma_{xy}\\ \Sigma_{yx} & \Sigma_{yy}\end{bmatrix}^{-1} \left(\begin{bmatrix}x\\y\end{bmatrix} - \begin{bmatrix}\mu_x\\\mu_y\end{bmatrix}\right)\right)}{(2\pi)^{D/2}\sqrt{|\Sigma|}}$$

$$p(x) = \frac{1}{(2\pi)^{D_x/2}\sqrt{|\Sigma_{xx}|}} \exp\left(-\frac{1}{2}(x-\mu_x)^T\Sigma_{xx}^{-1}(x-\mu_x)\right) = N(\mu_x, \Sigma_{xx})$$

$$p(y|x) = N\left(\mu_y + \Sigma_{yx}\Sigma_{xx}^{-1}(x-\mu_x), \Sigma_{yy} - \Sigma_{yx}\Sigma_{xx}^{-1}\Sigma_{xy}\right)$$



- Here argmax is conditional expectation:

$$\hat{y} = \mu_y + \Sigma_{yx}\Sigma_{xx}^{-1}(x-\mu_x)$$