DSC 140A

Probabilistic Modeling & Machine Learning

Lecture 13 | Part 1

Bayes with Multiple Features

Recap

- **Bayes Classifier:** predict y that maximizes $\mathbb{P}(Y = y \mid X = x)$

- **Alternatively:** predict y that maximizes

$$p_X(x \mid Y = y)\mathbb{P}(Y = y)$$

- We must estimate these probabilities/densities.

Example: NBA Players

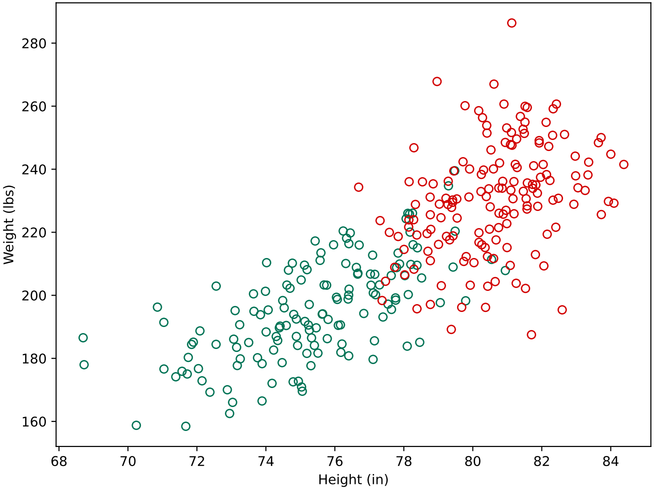
- **Guard** and **Forward** are two positions in basketball.

- Forwards tend to be larger than guards.

Example: NBA Players

- Suppose we have a data set of n NBA players:
  - X_1: the player's height
  - X_2: the player's weight
  - Y: the player's position (1 = guard, 0 = forward)

- **Given:** a new player's height and weight, predict their position.


280
260
240
220
200
180
160
Weight (lbs)
68
70
72
74
76
78
80
82
84
Height (in)

Bayes in ≥ 2 Dimensions

- With one feature, Bayes said to pick y maximizing:

$$p_X(x \mid Y = y)\mathbb{P}(Y = y)$$

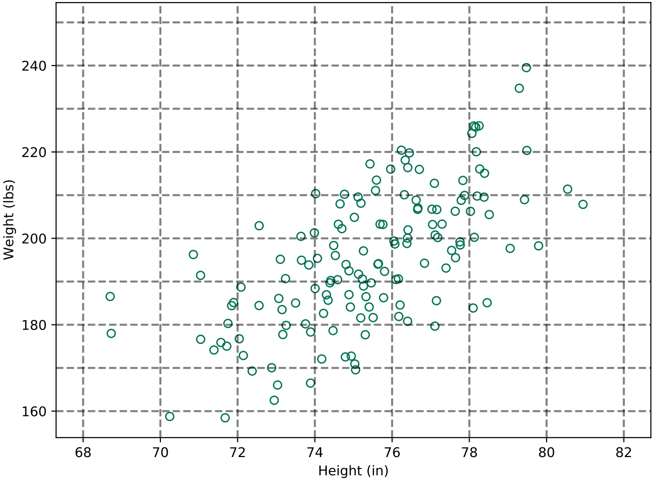
- With k features, pick y maximizing:

$$p_{\vec{X}}(\vec{x} \mid Y = y)\mathbb{P}(Y = y)$$

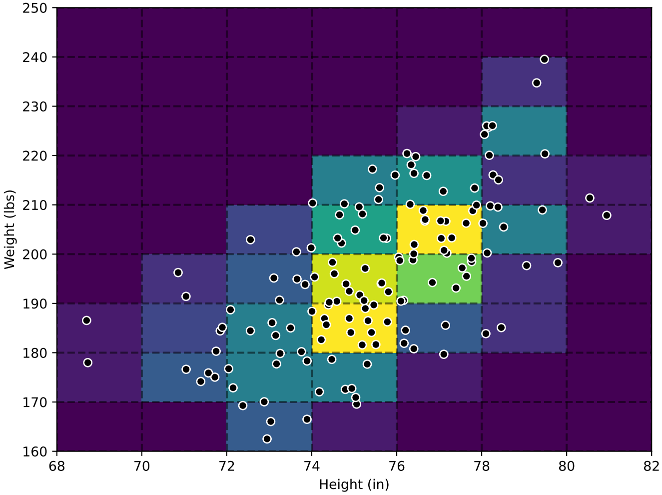
- $\vec{x}$ is the **feature vector**. Here: $(\text{height}, \text{weight})^T$

- We need to estimate density $p(\vec{x} \mid Y = y)$ for each class.

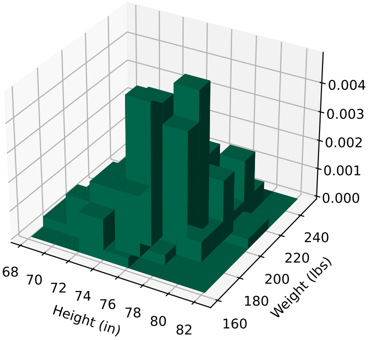
Estimating with Histograms



Estimating with Histograms



Estimating with Histograms



Predicting with Histograms

To predict the class of an input $\vec{x}$:

1. Use histograms to estimate $p_{\vec{X}}(\vec{x} \mid Y = y)$ for each class separately.

2. Predict the class y maximizing

$$p_{\vec{X}}(\vec{x} \mid Y = y)\mathbb{P}(Y = y)$$

Histogram Estimators

- Histogram density estimators are very flexible.
- But suffer heavily from **curse of dimensionality.**
- Not feasible for estimating density in more than a few dimensions.

Today

- **Last time:** we saw the **parametric** approach to density estimation.
  - Pick a parametric distribution (e.g., Gaussian)
  - Find parameters by maximizing likelihood

- We saw how to do this for one-dimensional data.

- **Today:** multidimensional data.

In particular...

- **Today:** multivariate Gaussian density estimation.
- That is: fitting multivariate Gaussians to data with maximum likelihood.

DSC 140A

Probabilistic Modeling & Machine Learning

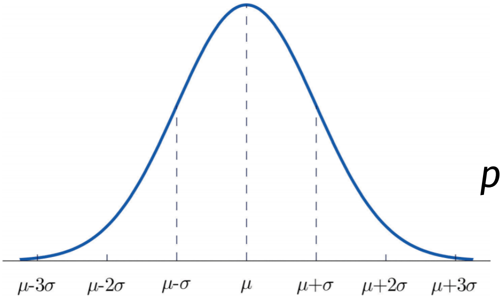
Lecture 13 | Part 2

Multivariate Gaussians

Multivariate Gaussians

- In 1 dimension, a Gaussian seemed to describe distribution of heights.
- Does a **multivariate** Gaussian describe distribution of heights and weights?

“Deriving” Multivariate Gaussians



$$X \sim \mathcal{N}(\mu, \sigma^2)$$

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(x-\mu)^2/\sigma^2}$$

Setting #1

- Suppose we have d independent random variables $X_1, \ldots, X_d$.

- Assume that each is Gaussian; different mean, but **same** variance:

$$X_1 \sim \mathcal{N}(\mu_1, \sigma^2), \quad X_2 \sim \mathcal{N}(\mu_2, \sigma^2), \ldots, \quad X_d \sim \mathcal{N}(\mu_d, \sigma^2).$$

Setting #1

- What is the **joint density** $p(x_1, x_2, \dots, x_d)$?

- Since we assumed $X_1, \dots, X_d$ are independent:

$$p(x_1, x_2, \dots, x_d) = p(x_1)p(x_2)\cdots p(x_d)$$

$$= \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(x-\mu_1)^2/\sigma^2}\right) \cdot \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(x-\mu_2)^2/\sigma^2}\right) \cdots \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(x-\mu_d)^2/\sigma^2}\right)$$

Setting #1

- What is the **joint density** $p(x_1, x_2, \ldots, x_d)$?
- Since we assumed $X_1, \ldots, X_d$ are independent:

$$p(x_1, x_2, \ldots, x_d) = p(x_1)p(x_2)\cdots p(x_d)$$

$$= \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(x-\mu_1)^2/\sigma^2}\right) \cdot \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(x-\mu_2)^2/\sigma^2}\right) \cdots \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(x-\mu_d)^2/\sigma^2}\right)$$

$$= \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{(x_1-\mu_1)^2 + (x_2-\mu_2)^2 + \ldots + (x_d-\mu_d)^2}{2\sigma^2}\right)$$

Setting #1

- What is the **joint density** $p(x_1, x_2, \ldots, x_d)$?

- Since we assumed $X_1, \ldots, X_d$ are independent:

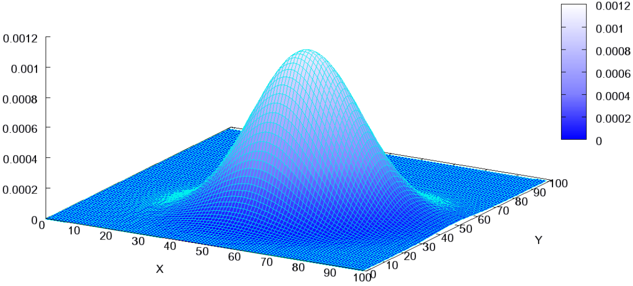
$$p(x_1, x_2, \ldots, x_d) = p(x_1)p(x_2)\cdots p(x_d)$$

$$= \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(x-\mu_1)^2/\sigma^2}\right) \cdot \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(x-\mu_2)^2/\sigma^2}\right) \cdots \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(x-\mu_d)^2/\sigma^2}\right)$$

$$= \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{(x_1 - \mu_1)^2 + (x_2 - \mu_2)^2 + \ldots + (x_d - \mu_d)^2}{2\sigma^2}\right)$$

$$= \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|\vec{x} - \vec{\mu}\|^2}{2\sigma^2}\right)$$

Setting #1

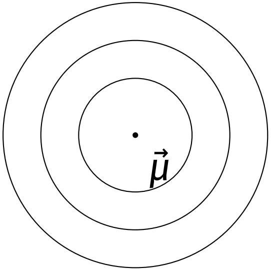


0.0012
0.001
0.0008
0.0006
0.0004
0.0002
0
0
10
20
30
40
50
60
70
80
90
100
X
Y

Setting #1: Spherical Gaussians

$$p(\vec{x}) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{1}{2}\frac{\|\vec{x}-\vec{\mu}\|^2}{\sigma^2}\right)$$

- Contours are (hyper)spheres.
- Every slice through middle gives same Gaussian.



Setting #2

- Still assume $X_1, \dots, X_d$ are independent, Gaussian.
- But they now have different variances:

$$X_1 \sim \mathcal{N}(\mu_1, \sigma_1^2), \quad X_2 \sim \mathcal{N}(\mu_2, \sigma_2^2), \dots, \quad X_d \sim \mathcal{N}(\mu_d, \sigma_d^2).$$

Setting #2

$$p(x_1, x_2, \dots, x_d) = p(x_1)p(x_2)\cdots p(x_d)$$

$$= \left(\frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{1}{2}(x-\mu_1)^2/\sigma_1^2}\right) \cdot \left(\frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-\frac{1}{2}(x-\mu_2)^2/\sigma_2^2}\right) \cdots \left(\frac{1}{\sqrt{2\pi\sigma_d^2}} e^{-\frac{1}{2}(x-\mu_d)^2/\sigma_d^2}\right)$$

Setting #2

$$p(x_1, x_2, \ldots, x_d) = p(x_1)p(x_2)\cdots p(x_d)$$

$$= \left(\frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{1}{2}(x-\mu_1)^2/\sigma_1^2}\right) \cdot \left(\frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-\frac{1}{2}(x-\mu_2)^2/\sigma_2^2}\right) \cdots \left(\frac{1}{\sqrt{2\pi\sigma_d^2}} e^{-\frac{1}{2}(x-\mu_d)^2/\sigma_d^2}\right)$$

$$= \frac{1}{(2\pi)^{d/2}\sigma_1 \cdot \sigma_2 \cdots \sigma_d} \exp\left(-\frac{1}{2}\left[\frac{(x_1-\mu_1)^2}{\sigma_1^2} + \frac{(x_2-\mu_2)^2}{\sigma_2^2} + \ldots + \frac{(x_d-\mu_d)^2}{\sigma_d^2}\right]\right)$$

Setting #2

- Define

$$C = \begin{pmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \sigma_d^2 \end{pmatrix}$$

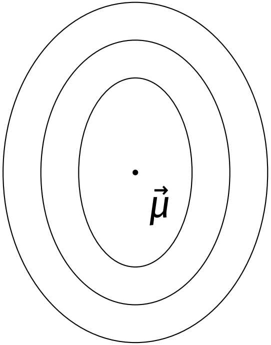
- Then:

$$p(\vec{x}) = \frac{1}{(2\pi)^{d/2}|C|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\vec{x} - \vec{\mu})^T C^{-1} (\vec{x} - \vec{\mu})\right)$$

where $|C|$ is the **determinant** of C.

Setting #2: Axis-Aligned Gaussians

$$p(\vec{x}) = \frac{1}{(2\pi)^{d/2}|C|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\vec{x}-\vec{\mu})^T C^{-1}(\vec{x}-\vec{\mu})\right)$$



- Contours are axis-aligned (hyper)ellipses.
- C is the **covariance matrix**.
  - Diagonal.
  - Entries are variances.

Setting #3: General Gaussians

- We have assumed that $X_1, \dots, X_d$ are independent.

- Now assume that they're not. Define **covariance**:

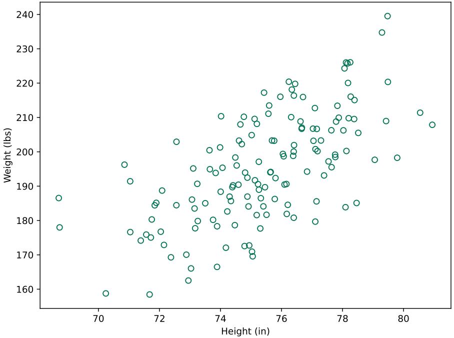
$$\text{Cov}(X_i, X_j) = \mathbb{E}[(X_i - \mu_i)(X_j - \mu_j)]$$

- **Note**:

$$\text{Var}(\text{X}_\text{i}) = \text{Cov}(X_i, X_i)$$

Covariance

- Covariance measures how much two quantities **vary together**.



$$\mathrm{Cov}(X_i, X_j) = \mathbb{E}[(X_i - \mu_i)(X_j - \mu_j)]$$

Setting #3: General Gaussians

- Now the **covariance matrix** has off-diagonal elements:

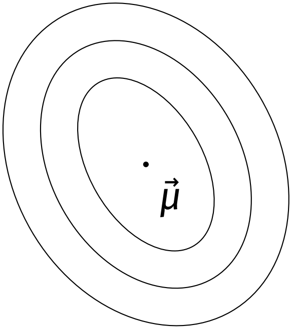
$$C = \begin{pmatrix} \mathrm{Var}(X_1) & \mathrm{Cov}(X_1, X_2) & \cdots & \mathrm{Cov}(X_1, X_d) \\ \mathrm{Cov}(X_2, X_1) & \mathrm{Var}(X_2) & \cdots & \mathrm{Cov}(X_2, X_d) \\ \cdots & \cdots & \cdots & \cdots \\ \mathrm{Cov}(X_d, X_1) & \mathrm{Cov}(X_d, X_2) & \cdots & \mathrm{Var}(X_d) \end{pmatrix}$$

- Since $\mathrm{Cov}(X_i, X_j) = \mathrm{Cov}(X_j, X_i)$, C is symmetric.

Setting #3: General Gaussians

$$p(\vec{x}) = \frac{1}{(2\pi)^{d/2}|C|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\vec{x} - \vec{\mu})^T C^{-1}(\vec{x} - \vec{\mu})\right)$$

Contours are general (hyper)ellipses.
C need not be diagonal.



Overview

- The probability density function for a multivariate Gaussian distribution is:

$$p(\vec{x}) = \frac{1}{(2\pi)^{d/2}|C|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\vec{x} - \vec{\mu})^T C^{-1}(\vec{x} - \vec{\mu})\right)$$

- Here, C is the **covariance matrix**.

Overview

- There are three cases:

1. C is diagonal, with all the same entries.
2. C is diagonal, with different entries.
3. C is not diagonal.

DSC 140A

Probabilistic Modeling & Machine Learning

Lecture 13 | Part 3

Fitting Multivariate Gaussians

Fitting Multivariate Gaussians

- Suppose $\vec{x}^{(1)}, \ldots, \vec{x}^{(n)}$ came from a multivariate Gaussian.
- What were the parameters of that Gaussian?
- We can use the principle of **maximum likelihood**.

What are the parameters?

$$p(\vec{x}) = \frac{1}{(2\pi)^{d/2}|C|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\vec{x}-\vec{\mu})^T C^{-1}(\vec{x}-\vec{\mu})\right)$$

- $\vec{\mu}$: controls Gaussian's location
- C: controls Gaussian's shape

Estimating $\vec{\mu}$

- The maximum likelihood estimator for μ is:

$$\vec{\mu}_{\text{MLE}} = \frac{1}{n} \sum_{i=1}^{n} \vec{x}^{(i)}$$

Estimating C

- First: make assumptions on covariance matrix.
- In order from strict to weak:
  - Spherical: C is diagonal, with all the same entries.
  - Axis-Aligned: C is diagonal, with different entries.
  - General: C is not diagonal.
- The weaker the assumptions, the more parameters to estimate.

Fitting Spherical Gaussians

- Only one variance parameter: σ^2.
- The density function becomes:

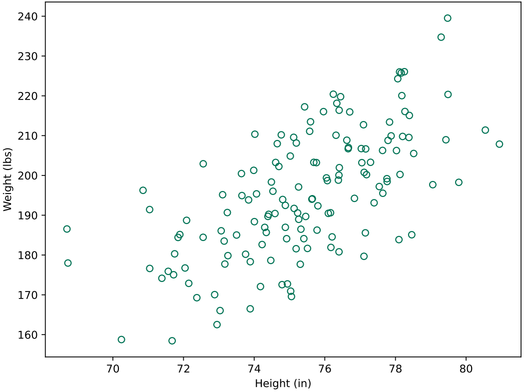
$$p(\vec{x}) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{(\vec{x} - \vec{\mu})^T(\vec{x} - \vec{\mu})}{2\sigma^2}\right)$$

- The maximum likelihood estimator:

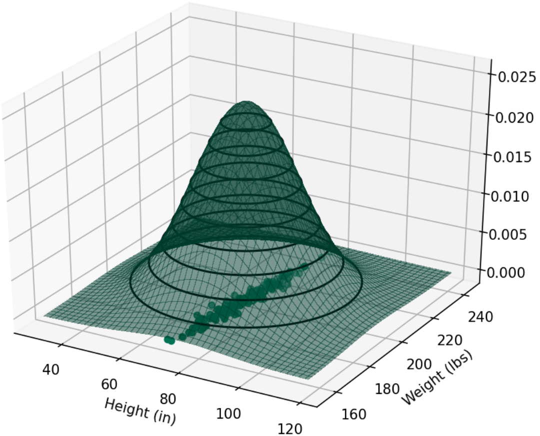
$$\sigma^2_{\text{MLE}} = \frac{1}{n}\sum_{i=1}^{n} \|\vec{x}^{(i)} - \vec{\mu}_{\text{MLE}}\|^2$$

Example: NBA Data

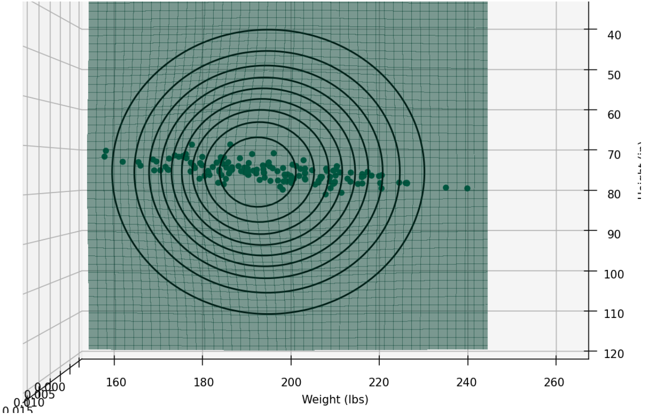
- What if we fit a spherical Gaussian to the NBA data?



Fitting Spherical Gaussians



Fitting Spherical Gaussians



Example: NBA Data

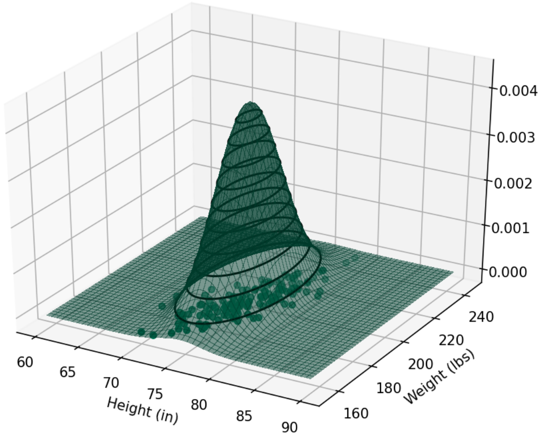
- Spherical Gaussians are not well-suited to this data.
- Perhaps if the data were **standardized...**
- Instead, try axis-aligned Gaussians.

Fitting Axis-Aligned Gaussians

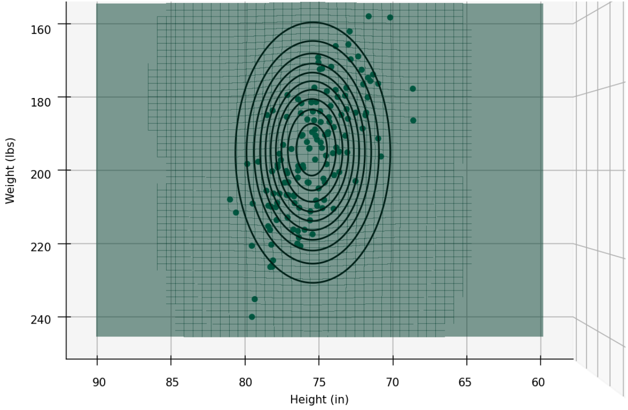
- Variance for each axis: σ_1^2 and σ_2^2.
- Maximum likelihood estimates:

$$\sigma_1^2 = \text{sample variance of heights}$$
$$\sigma_2^2 = \text{sample variance of weights}$$

Fitting Axis-Aligned Gaussians



Fitting Axis-Aligned Gaussians



Example: NBA Data

- Axis-aligned Gaussian does not capture correlation between height and weight.
- Try general Gaussian with full covariance.

Fitting General Gaussians

- Must compute covariance for each pair of dimensions.

- Maximum likelihood estimate for covariance of feature *i* and *j*:

$$C_{ij} = \left(\frac{1}{n} \sum_{k=1}^{n} \vec{x}_i^{(k)} \vec{x}_j^{(k)} \right) - \mu_i \mu_j$$

Computing the Covariance Matrix

Step 1. Make matrix with heights in first column, weights in second:

$$\begin{pmatrix} \text{height } 1 & \text{weight } 1 \\ \text{height } 2 & \text{weight } 2 \\ \dots & \dots \\ \text{height } n & \text{weight } n \end{pmatrix}$$

Computing the Covariance Matrix

Step 2. Subtract sample mean height, mean weight from each column. Call this matrix X:

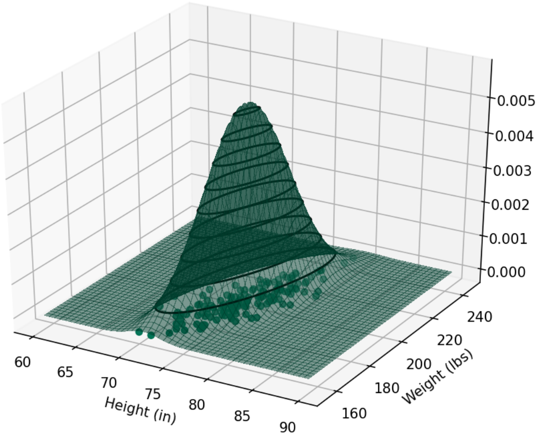
$$X = \begin{pmatrix} \text{height } 1 - \text{mean height} & \text{weight } 1 - \text{mean weight} \\ \text{height } 2 - \text{mean height} & \text{weight } 2 - \text{mean weight} \\ \dots & \dots \\ \text{height } n - \text{mean height} & \text{weight } n - \text{mean weight} \end{pmatrix}$$

Computing the Covariance Matrix

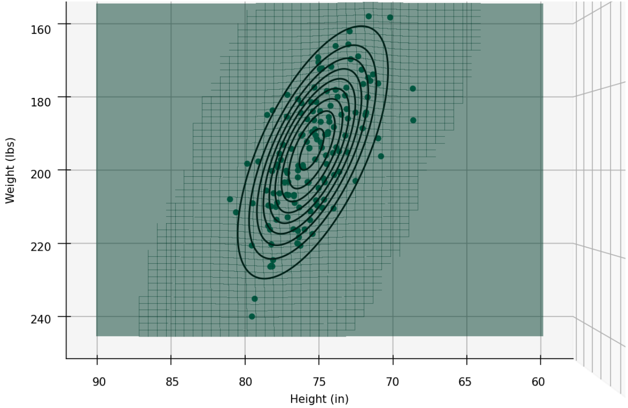
The empirical covariance matrix is then:

$$C = \frac{1}{n} X^T X$$

Fitting General Gaussians



Fitting General Gaussians



Up next...

Making predictions using these fitted Gaussians.

DSC 140A

Probabilistic Modeling & Machine Learning

Lecture 13 | Part 4

Discriminant Analysis

Bayes Classifier with MV Gaussians

1. Fit Gaussian for $p(\vec{X} \mid Y = y)$ for each class, y.

2. For new point, predict y maximizing:

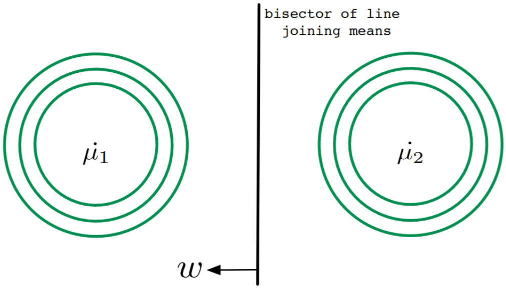
$$p(\vec{X} = \vec{x} \mid Y = y)\mathbb{P}(Y = y)$$

Decision Boundary

- For every point in space, we have a classification.
- The **decision boundary**: surface between different classifications.
  - On one side, prediction is y_1;
  - on the other, prediction is y_2.

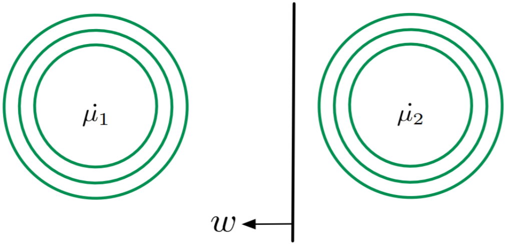
Setting #1

- Assume:
  - classes equally likely: $\mathbb{P}(Y = 1) = \mathbb{P}(Y = 0)$
  - identical covariance matrices



Setting #1

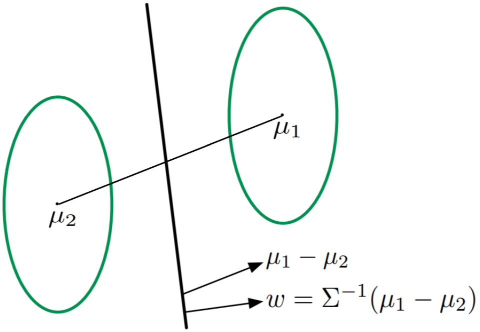
- If $\mathbb{P}(Y = y_1) > \mathbb{P}(Y = y_2)$:



Choose class 1 if $\vec{w} \cdot \frac{(\vec{\mu}_1 - \vec{\mu}_2)}{\sigma^2} \geq \theta$.

Setting #2

- Assume:
  - covariance matrices identical, diagonal
  - that is: axis-aligned Gaussians



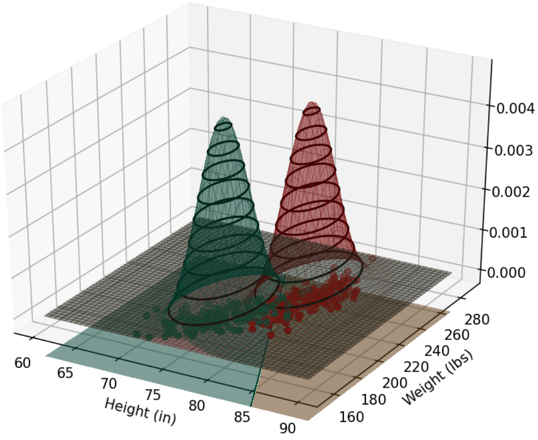
Predict class 1 if $\vec{x} \cdot \vec{w} \geq \theta$.

Example

- Use to predict position given height and weight.
- How do we get one covariance matrix?
- Don't lump data together...
- Instead, compute covariance matrix for each class, perform weighted average:

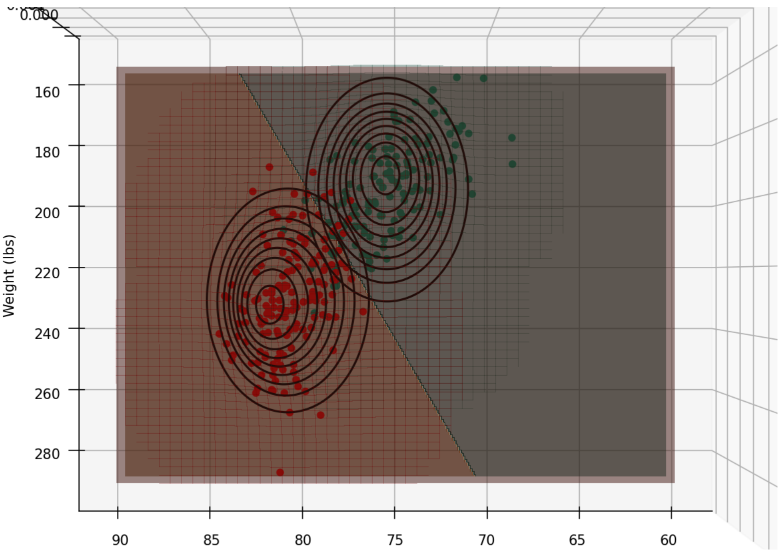
$$C = \frac{n_1 C_1 + n_2 C_2}{n_1 + n_2}$$

Example



0.004
0.003
0.002
0.001
0.000
60
65
70
75
80
85
90
Height (in)
160
180
200
220
240
260
280
Weight (lbs)

Example

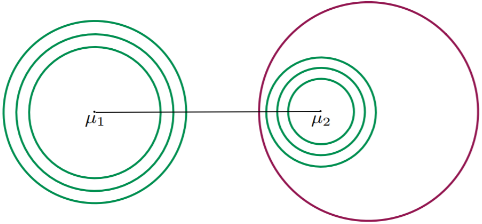


Linear Discriminant Analysis

- When covariance matrices are **equal**, decision boundary is linear.
- This procedure is called **linear discriminant analysis** (LDA).
- True even if the Gaussians have full covariance.

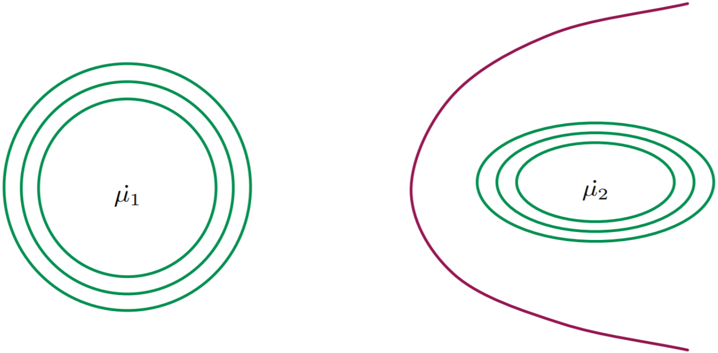
Setting #3

- Assume:
  - covariance matrices C_1, C_2 different, non-diagonal

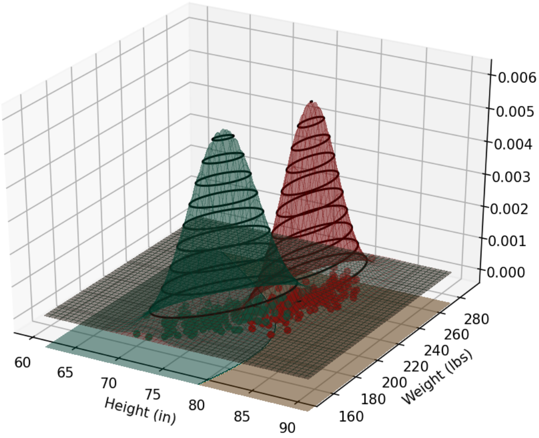


Setting #3

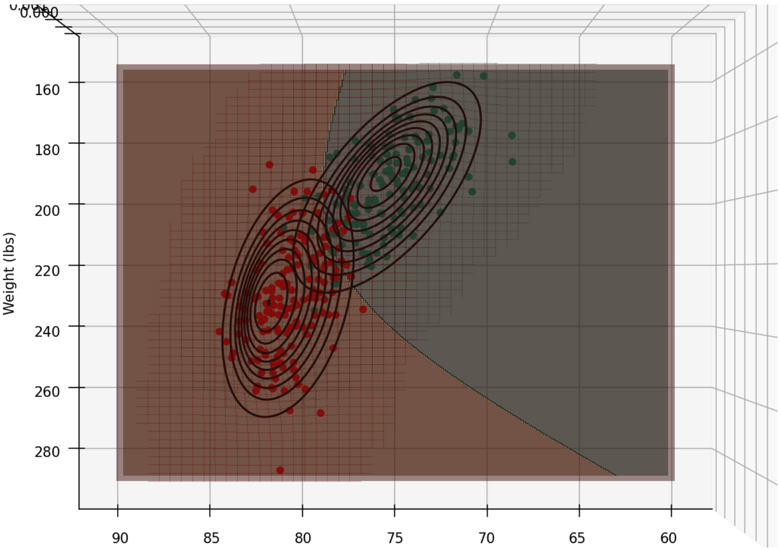
- Assume:
  - covariance matrices C_1, C_2 different, non-diagonal



Example


0.006
0.005
0.004
0.003
0.002
0.001
0.000
60
65
70
75
80
85
90
Height (in)
160
180
200
220
240
260
280
Weight (lbs)

Example



Quadratic Discriminant Analysis

- When covariance matrices are equal, decision boundary is quadratic (ellipsoidal, paraboloidal, hyperboloidal).
- This procedure is called **quadratic discriminant analysis** (QDA).

In practice...

- A full covariance requires estimating $\Theta(d^2)$ parameters; needs more data.
- Gaussian assumption may be a poor match for data.