Fourth Industrial Summer School

Advanced Machine Learning

Generative Models

Session Objectives

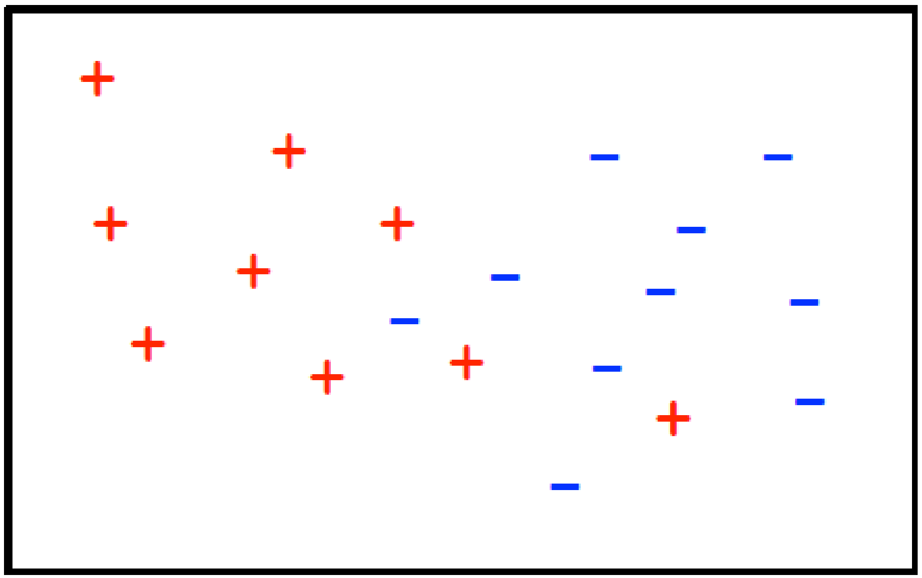
- ✓ Generative approach
- ✓ One dimensional modeling
- ✓ Two Dimensional modeling
- ✓ Multivariate Gaussians

Generative Classifiers

Introduction

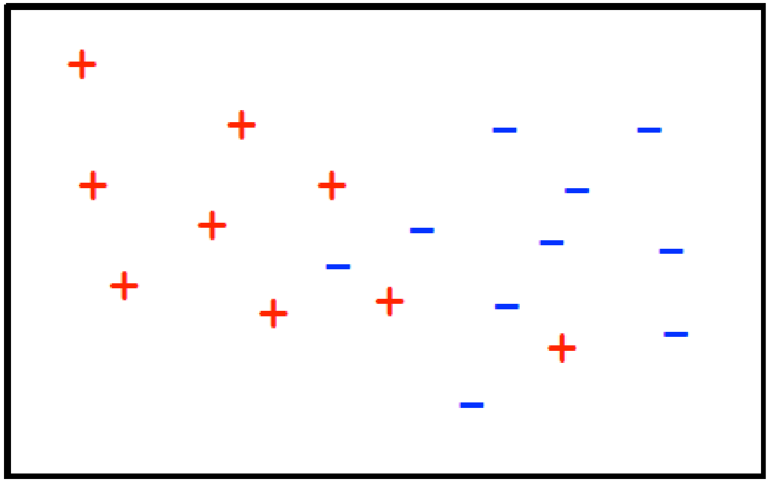
The generative approach to classification

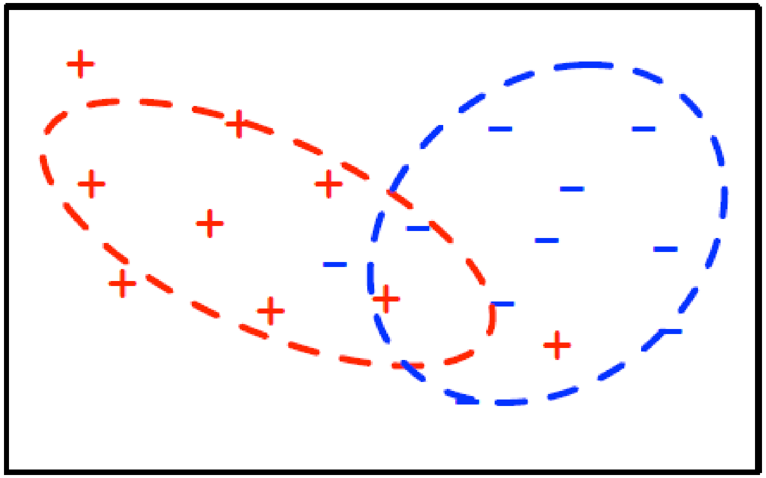
- The learning process:
  - Fit a probability distribution to each class, individually



The generative approach to classification

- The learning process:
  - Fit a probability distribution to each class, individually





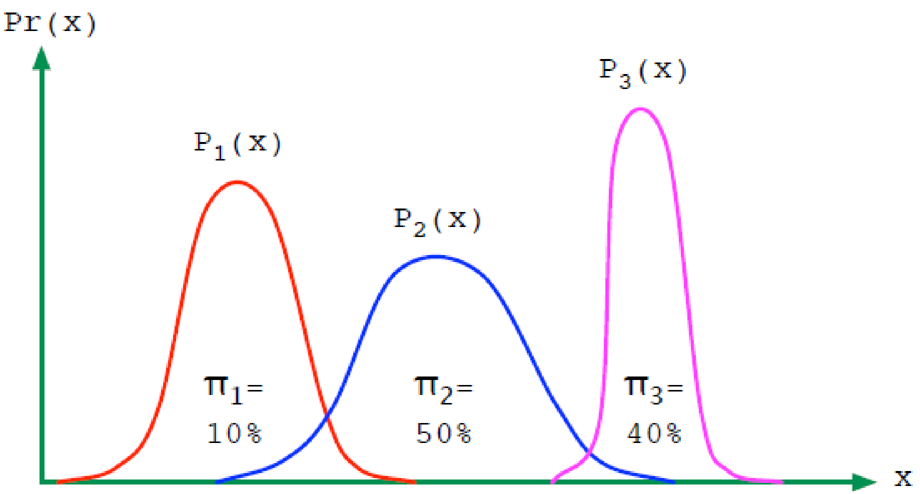
- To classify a new point:
  - Which of these distributions was it most likely to have come from?

Generative models

Example:
Data space $\mathcal{X} = \mathbb{R}$
Classes/labels $\mathcal{Y} = \{1, 2, 3\}$

For each class j, we have:

- the probability of that class, $\pi_j = \Pr(y = j)$
- the distribution of data in that class, $P_j(x)$



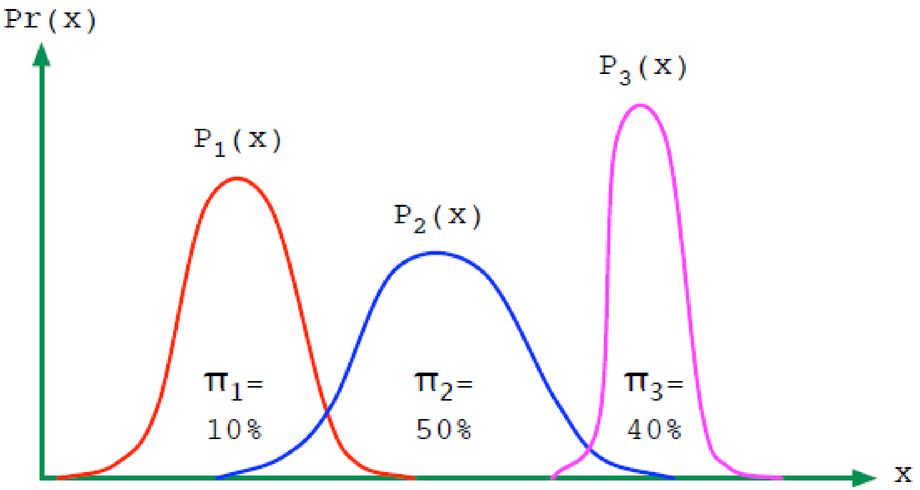
Overall **joint distribution**: $\Pr(x, y) = \Pr(y)\Pr(x|y) = \pi_y P_y(x).$

Generative models

Example:
Data space $\mathcal{X} = \mathbb{R}$
Classes/labels $\mathcal{Y} = \{1, 2, 3\}$

For each class j, we have:

- the probability of that class, $\pi_j = \Pr(y = j)$
- the distribution of data in that class, $P_j(x)$



Overall **joint distribution**: $\Pr(x, y) = \Pr(y)\Pr(x|y) = \pi_y P_y(x)$.

To classify a new x: pick the label y with largest $\Pr(x, y)$

Generative Classifiers

One Dimensional Modeling

The generative approach to classification

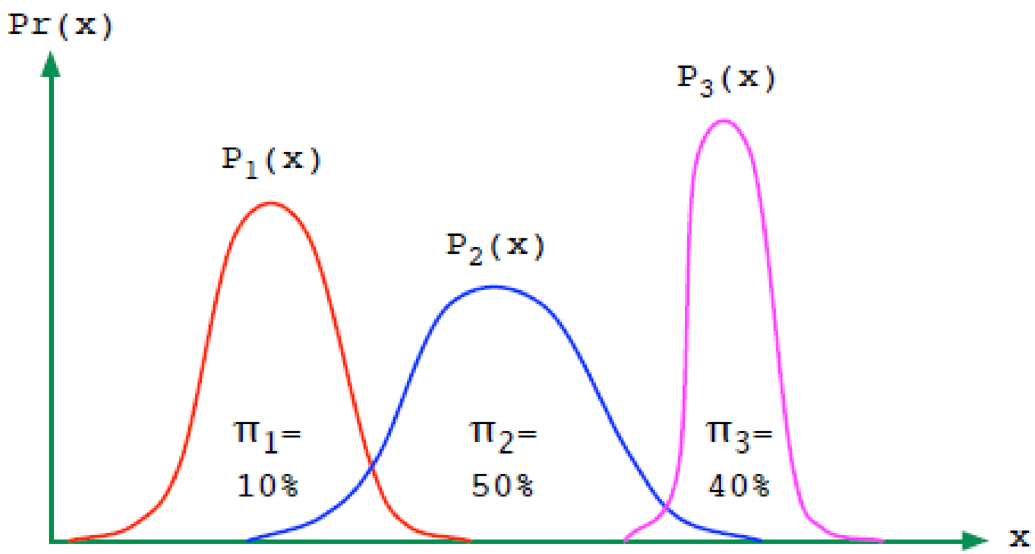
- Running Case Study
  - IRIS dataset
  - 4 Features

    ['petal_length', 'petal_width', 'sepal_length', 'sepal_width']
  - 3 Categories (Species)

    [Iris setosa, Iris virginica, and Iris versicolor]

  - A total of 150 samples, divided into train and test sets

Image: https://en.wikipedia.org/wiki/Iris_flower_data_set



For any data point $x \in \mathcal{X}$ and any candidate label j,

$$\Pr(y=j|x) = \frac{\Pr(y=j)\Pr(x|y=j)}{\Pr(x)} = \frac{\pi_j P_j(x)}{\Pr(x)}$$

Optimal prediction: the class j with largest $\pi_j P_j(x)$.

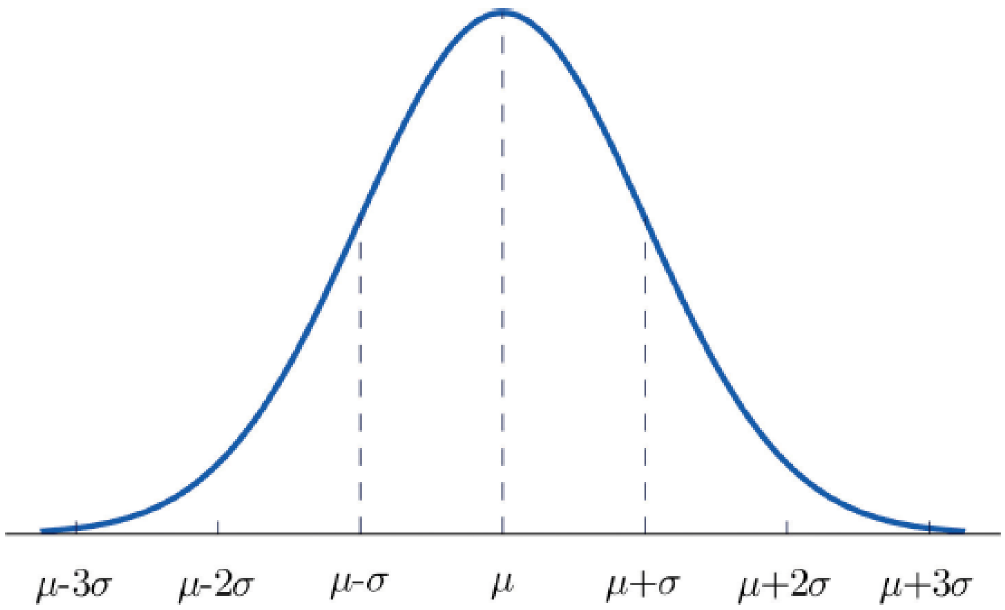
Case Study-Fitting a generative model

- Training set of 105 samples
  - Species-0: 33, Species-1: 34, Species-2: 38
  - For each sample, we have four features

- Class weights:
  - $\Pi_0 = 33/105 = 0.31$, $\Pi_1 = 34/105 = 0.32$, $\Pi_2 = 38/105 = 0.37$,

- Need distributions P_1; P_2; P_3, one per class.
  - Base these on a single feature: 'petal_length'.

The univariate Gaussian

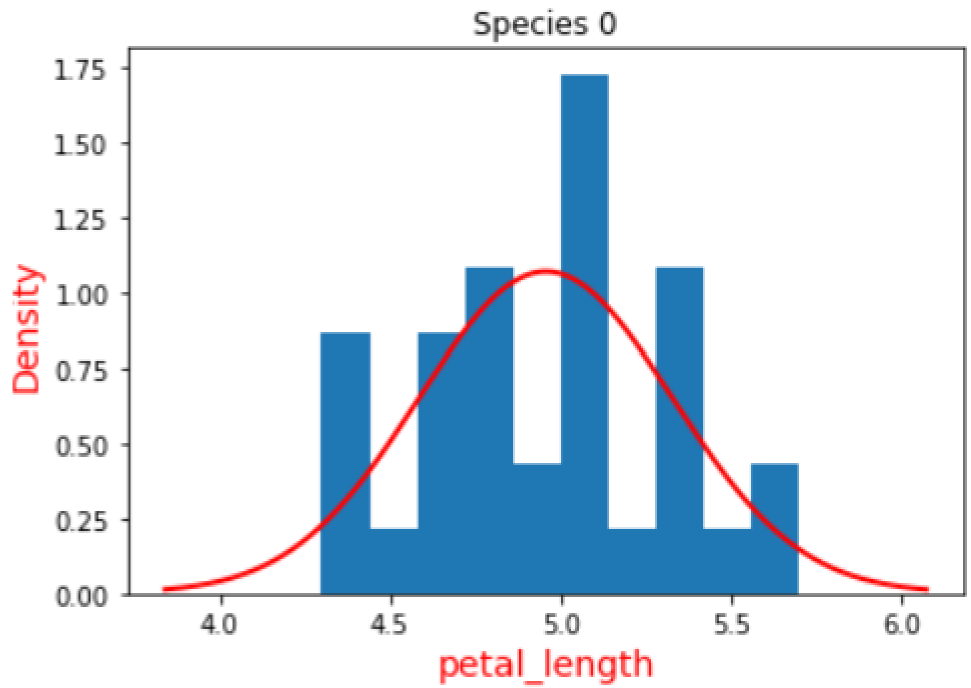


The Gaussian $N(\mu, \sigma^2)$ has mean μ, variance σ^2, and density function

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

Case Study-Distribution for Species-0

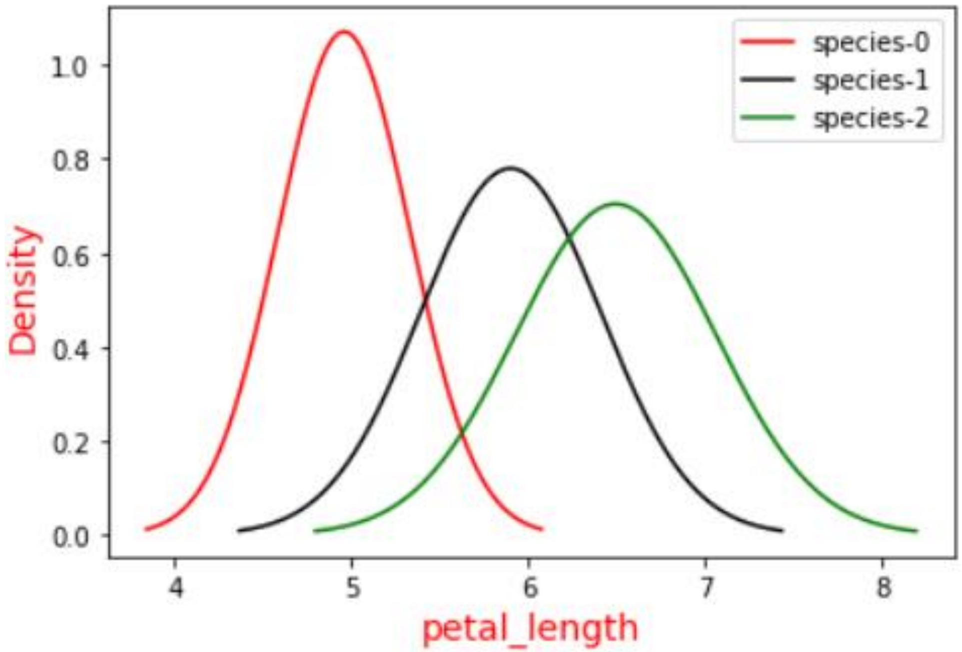
- Feature: 'petal_length'



- Mean μ = 4.96, Standard deviation σ = 0.37 (variance 0.14)

Case Study-Distribution for all the species

- Feature: 'petal_length'



- π_1=0:31, P_1=N(4.96, 0.37)
- π_2=0:32, P_1=N(5.90, 0.51)
- π_3=0:37, P_1=N(6.49, 0.57)

- To classify x: Pick the j with highest $\pi_j P_j$ (x)
- Test error using feature petal_length: 12/45=26.67%

Generative Classifiers

Two Dimensional Modeling

The IRIS prediction problem

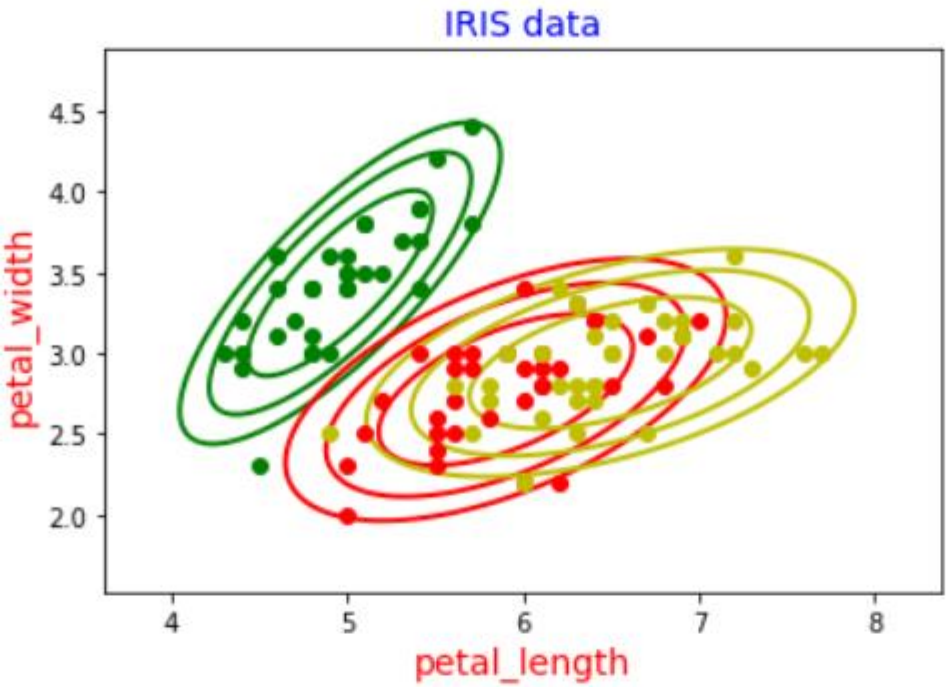
- Which species?

Image: https://en.wikipedia.org/wiki/Iris_flower_data_set

- Using one feature ('petal_length'), error rate is 26.67%.
- What if we use two features?
- This time: 'petal_length' and 'petal_width'.

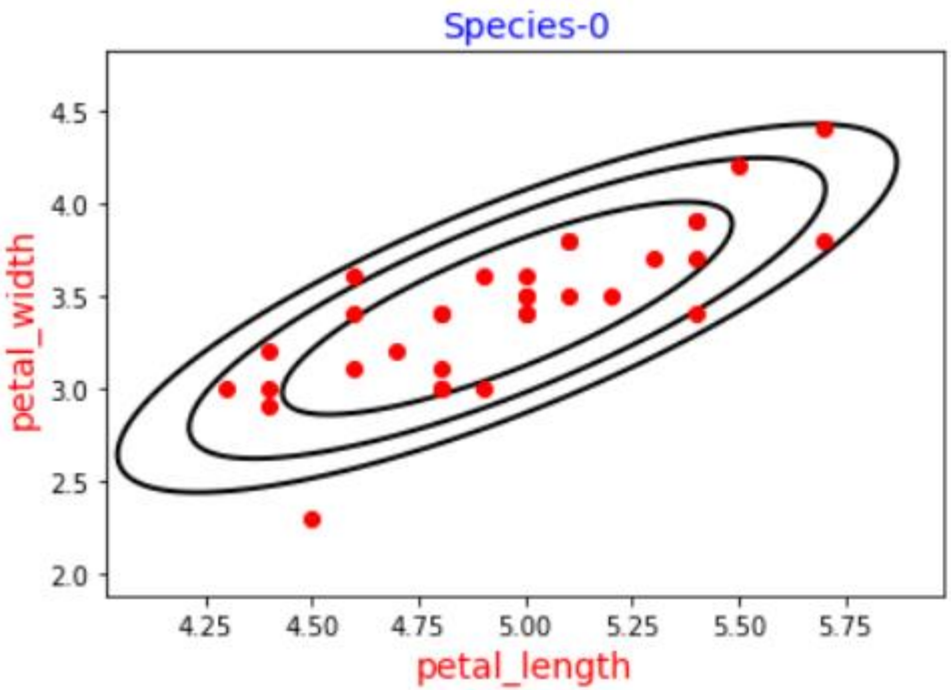
Why it helps to add features

- Better **separation** between the classes!



- Error rate drops from 26.67% to 22.22%.

The bivariate Gaussian



- Model species-0 by a bivariate Gaussian, parametrized by:

  mean $\mu = \begin{pmatrix} 4.96 \\ 3.43 \end{pmatrix}$ and covariance matrix $\Sigma = \begin{bmatrix} 0.14 & 0.12 \\ 0.12 & 0.17 \end{bmatrix}$

Dependence between two random variables

Suppose X_1 has mean μ_1 and X_2 has mean μ_2.

Can measure dependence between them by their **covariance**:

- $\mathrm{cov}(X_1, X_2) = \mathbb{E}[(X_1 - \mu_1)(X_2 - \mu_2)] = \mathbb{E}[X_1 X_2] - \mu_1 \mu_2$
- Maximized when $X_1 = X_2$, in which case it is $\mathrm{var}(X_1)$.
- It is at most $\mathrm{std}(X_1)\mathrm{std}(X_2)$.

The bivariate (2-d) Gaussian

A distribution over $(x_1, x_2) \in \mathbb{R}^2$, parametrized by:

- **Mean** $(\mu_1, \mu_2) \in \mathbb{R}^2$, where $\mu_1 = \mathbb{E}(X_1)$ and $\mu_2 = \mathbb{E}(X_2)$
- **Covariance matrix** $\Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}$ where $\left\{ \begin{array}{c} \Sigma_{11} = \mathrm{var}(X_1) \\ \Sigma_{22} = \mathrm{var}(X_2) \\ \Sigma_{12} = \Sigma_{21} = \mathrm{cov}(X_1, X_2) \end{array} \right\}$

- Density is highest at the mean, falls of in ellipsoidal contours.

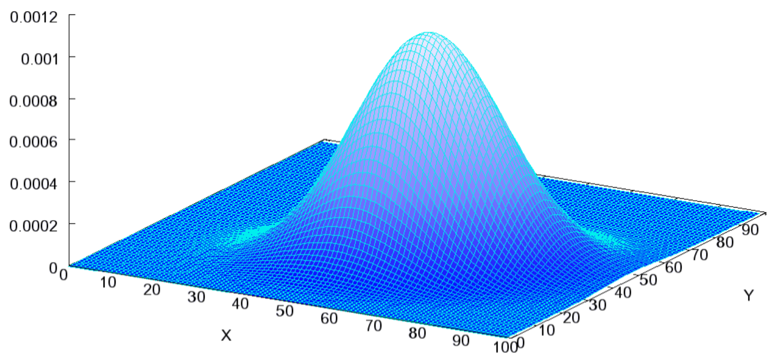


Image:https://en.wikipedia.org/wiki/Multivariate_normal_distribution#/media/File:Multivariate_Gaussian.png

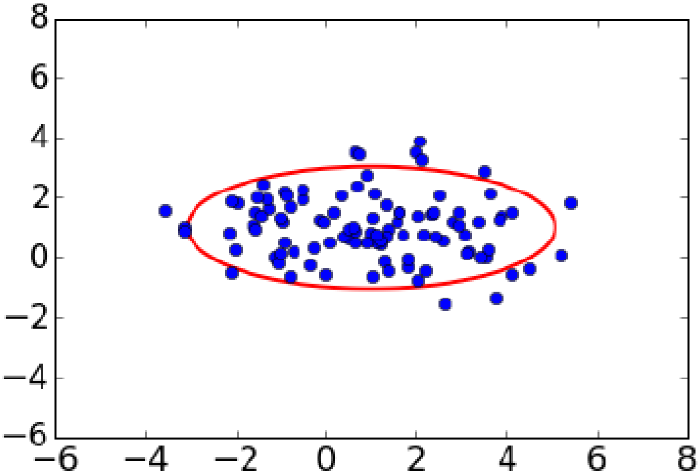
Density of the bivariate Gaussian

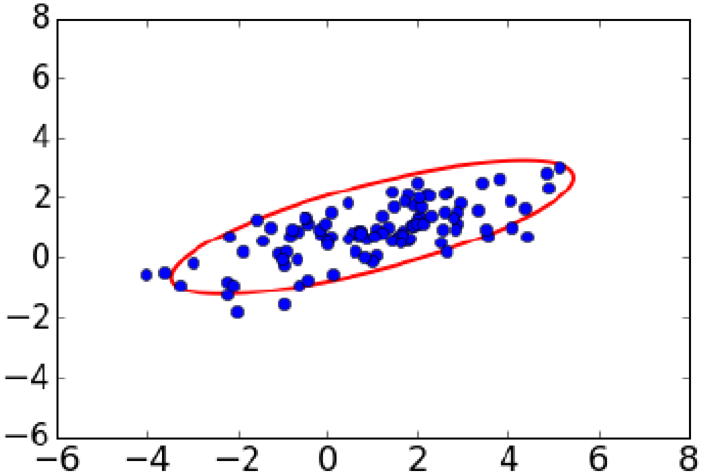
- **Mean** $(\mu_1, \mu_2) \in \mathbb{R}^2$, where $\mu_1 = \mathbb{E}(X_1)$ and $\mu_2 = \mathbb{E}(X_2)$
- **Covariance matrix** $\Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}$

Density $p(x_1, x_2) = \dfrac{1}{2\pi|\Sigma|^{1/2}} \exp\left(-\dfrac{1}{2} \begin{bmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{bmatrix}^T \Sigma^{-1} \begin{bmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{bmatrix}\right)$

Bivariate Gaussian: examples

- In either case, the mean is (1, 1)



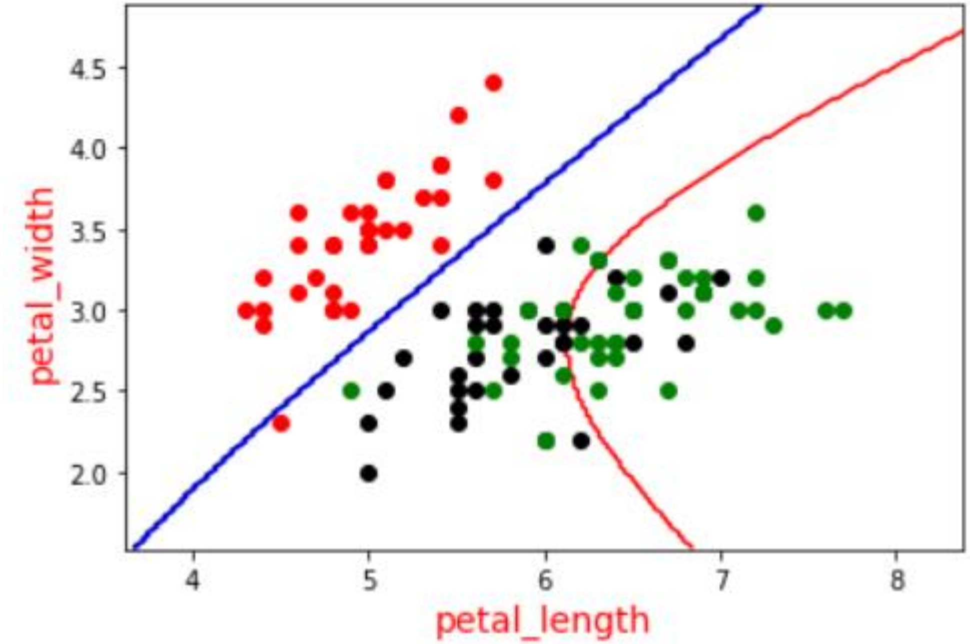


$$\Sigma = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} 4 & 1.5 \\ 1.5 & 1 \end{bmatrix}$$

The decision boundary

- Go from 1 to 2 features: error rate drops from 26.67% to 22.22%.

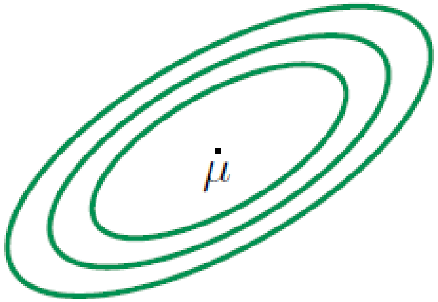


- What kind of function is this?
- Can we use more features?

Generative Classifiers

Multivariate Gaussians

The multivariate Gaussian



$N(\mu, \Sigma)$: Gaussian in $\mathbb{R}^d$

- mean: $\mu \in \mathbb{R}^d$
- covariance: $d \times d$ matrix Σ

Generates points $X = (X_1, X_2, \ldots, X_d)$.

- μ is the vector of coordinate-wise means:

$$\mu_1 = \mathbb{E}X_1, \ \mu_2 = \mathbb{E}X_2, \ldots, \ \mu_d = \mathbb{E}X_d.$$

- Σ is a matrix containing all pairwise covariances:

$$\Sigma_{ij} = \Sigma_{ji} = \text{cov}(X_i, X_j) \quad \text{if } i \neq j$$
$$\Sigma_{ii} = \text{var}(X_i)$$

Density $\quad p(x) = \dfrac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\dfrac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$

Special case: Independent features

- Suppose the X_i are independent, and $\mathrm{var}(X_i) = \sigma_i^2$
- What is the covariance matrix Σ , and what is its inverse Σ^{-1}?

Special case: Independent features

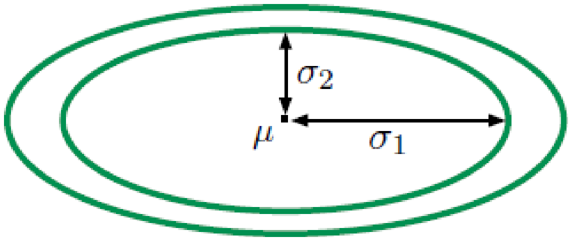
Diagonal Gaussian: the X_i are independent, with variances σ_i^2. Thus

$$\Sigma = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_d^2) \quad \text{(off-diagonal elements zero)}$$

Each X_i is an independent one-dimensional Gaussian $N(\mu_i, \sigma_i^2)$:

$$\Pr(x) = \Pr(x_1)\Pr(x_2)\cdots\Pr(x_d) = \frac{1}{(2\pi)^{d/2}\sigma_1\cdots\sigma_d}\exp\left(-\sum_{i=1}^{d}\frac{(x_i-\mu_i)^2}{2\sigma_i^2}\right)$$

Contours of equal density are **axis-aligned ellipsoids** centered at μ:



How many parameters?

Special case: Spherical Gaussian

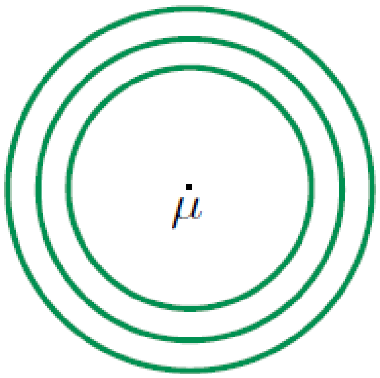
- Suppose the X_i are independent and all have the same variance σ^2

$$\Sigma = \sigma^2 I_d = \mathrm{diag}(\sigma^2, \sigma^2, \ldots, \sigma^2) \quad \text{(diagonal elements } \sigma^2 \text{, rest zero)}$$

Each X_i is an independent univariate Gaussian $N(\mu_i, \sigma^2)$:

$$\Pr(x) = \Pr(x_1)\Pr(x_2)\cdots\Pr(x_d) = \frac{1}{(2\pi)^{d/2}\sigma^d} \exp\left(-\frac{\|x-\mu\|^2}{2\sigma^2}\right)$$

Density at a point depends only on its distance from μ:



How to fit a Gaussian to data

Fit a Gaussian to data points $x^{(1)}, \ldots, x^{(m)} \in \mathbb{R}^d$.

- Empirical mean

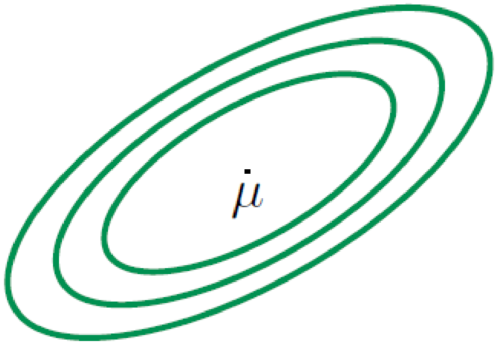
$$\mu = \frac{1}{m}\left(x^{(1)} + \cdots + x^{(m)}\right)$$

- Empirical covariance matrix has i, j entry:

$$\Sigma_{ij} = \left(\frac{1}{m}\sum_{k=1}^{m} x_i^{(k)} x_j^{(k)}\right) - \mu_i \mu_j$$

Classification using multivariate Gaussian

- Going from 1 to 2 features: Test error from 26.67% to 22.22%.
- With all 4 features: Test error rate drops to 4.44%.



$N(\mu, \Sigma)$: Gaussian in $\mathbb{R}^d$

- mean: $\mu \in \mathbb{R}^d$
- covariance: $d \times d$ matrix Σ

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

- What if we work on log domain?

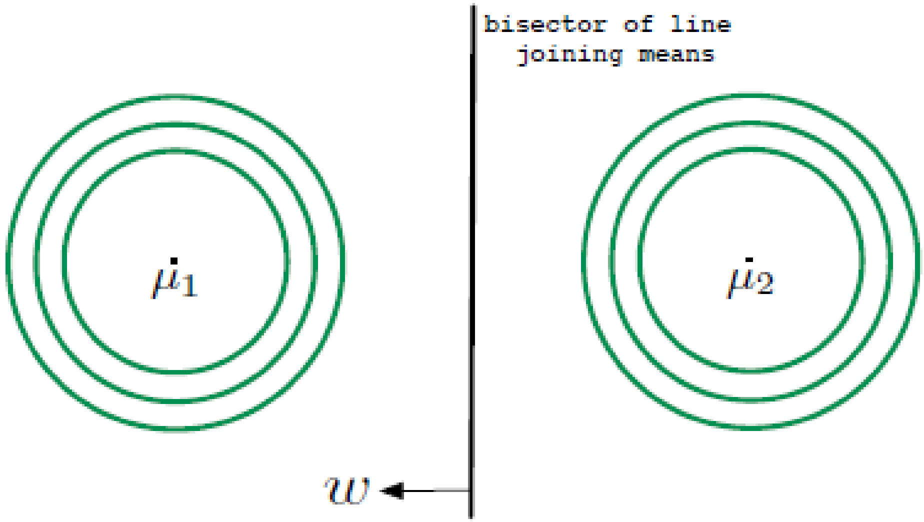
Binary classification with Gaussian

Common covariance: $\Sigma_1 = \Sigma_2 = \Sigma$

Linear decision boundary: choose class 1 if

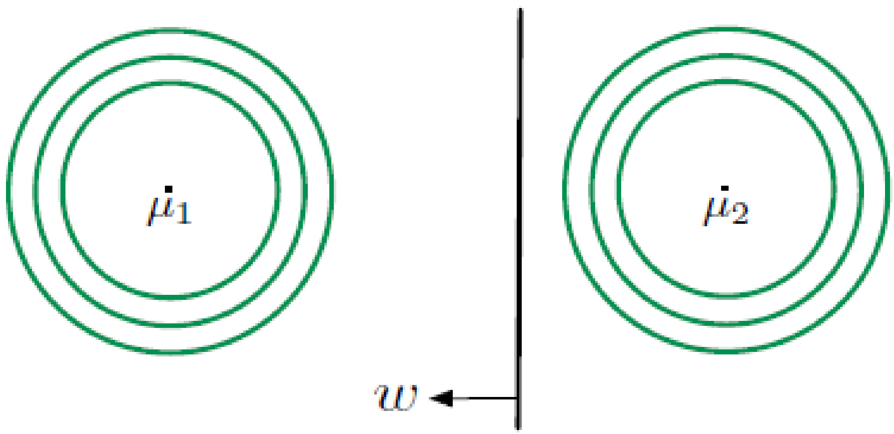
$$x \cdot \underbrace{\Sigma^{-1}(\mu_1 - \mu_2)}_{w} \geq \theta.$$

Example 1: Spherical Gaussians with $\Sigma = I_d$ and $\pi_1 = \pi_2$.



Binary classification with Gaussian

Example 2: Again spherical, but now $\pi_1 > \pi_2$.



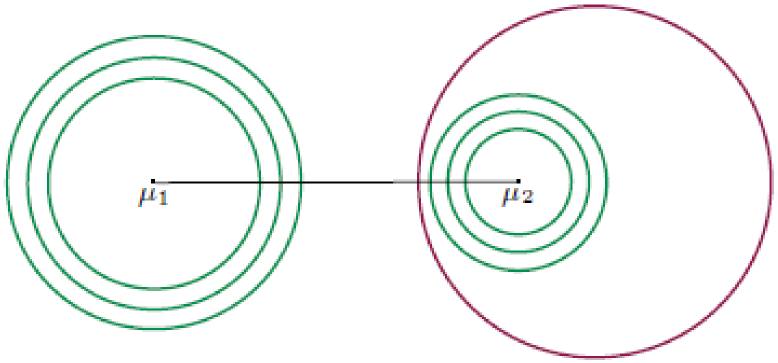
Binary classification with Gaussian

Different covariances: $\Sigma_1 \neq \Sigma_2$

Quadratic boundary: choose class 1 if $x^T M x + 2w^T x \geq \theta$, where:

$$M = \frac{1}{2}(\Sigma_2^{-1} - \Sigma_1^{-1})$$
$$w = \Sigma_1^{-1}\mu_1 - \Sigma_2^{-1}\mu_2$$

Example 1: $\Sigma_1 = \sigma_1^2 I_d$ and $\Sigma_2 = \sigma_2^2 I_d$ with $\sigma_1 > \sigma_2$



Think about 1-d case!

Multiclass discriminant analysis

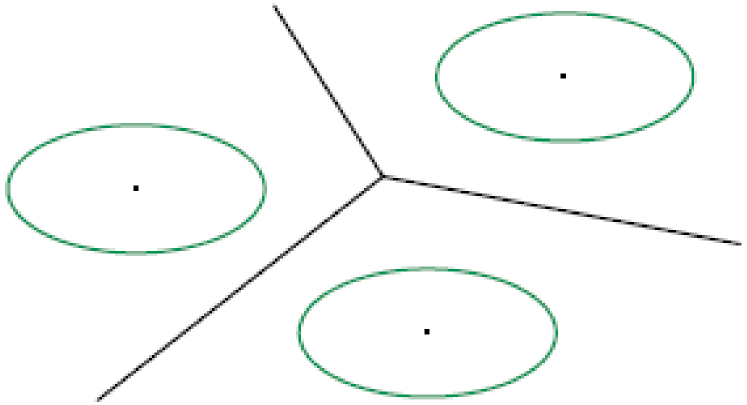
k classes: weights π_j, class-conditional densities $P_j = N(\mu_j, \Sigma_j)$.

Each class has an associated **quadratic** function

$$f_j(x) = \log\left(\pi_j \, P_j(x)\right)$$

To classify point x, pick $\arg\max_j f_j(x)$.

If $\Sigma_1 = \cdots = \Sigma_k$, the boundaries are **linear**.



Some other models and distributions

- Gaussian distribution with multiple components per class
- Bernoulli
- Poisson
- Graphical models

References

- Sanjoy Dasgupta, Machine Learning Fundamentals, UC San Diego
- Andrew Ng, Machine Learning, Stanford University
- Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, Foundations of Machine Learning, second edition, The MIT Press
- Andrew Ng, Machine Learning Yearning, deeplearning.ai