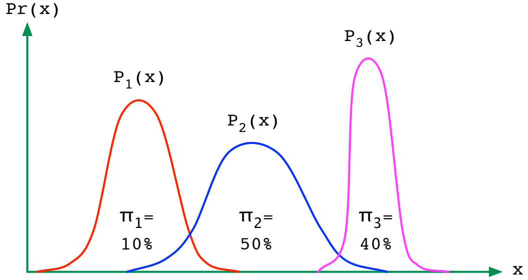
Classification with generative models II

ECE407

Recall: generative model framework



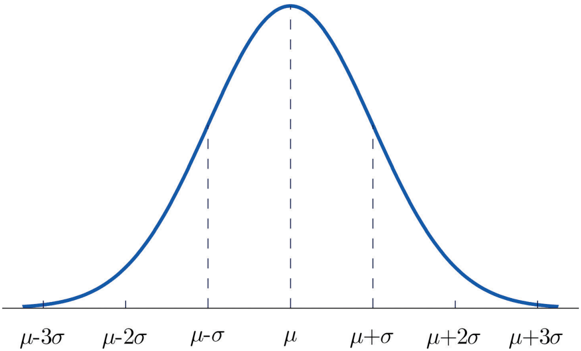
Labels $\mathcal{Y} = \{1, 2, \ldots, k\}$, density $\Pr(x) = \pi_1 P_1(x) + \cdots + \pi_k P_k(x)$.

where $P_1(x) = \Pr(x|Y=1), \ldots, P_k(x) = \Pr(x|Y=k)$

Approximate each P_j with a simple, parametric distribution:

- Product distributions.
  Assume coordinates are independent: naive Bayes.
- Multivariate Gaussians.
  Linear and quadratic discriminant analysis.
- More general graphical models.

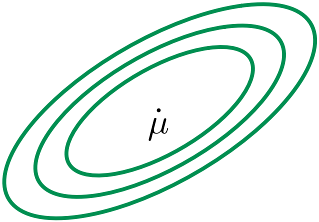
The univariate Gaussian



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

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

The multivariate Gaussian

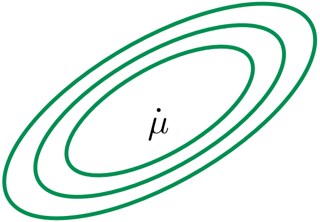


$N(\mu, \Sigma)$: Gaussian in $\mathbb{R}^p$

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

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

The multivariate Gaussian



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

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

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

Let $X = (X_1, X_2, \ldots, X_p)$ be a random draw from $N(\mu, \Sigma)$.

- $\mathbb{E}X = \mu$. That is, $\mathbb{E}X_i = \mu_i$ for all $1 \le i \le p$.
- $\mathbb{E}(X-\mu)(X-\mu)^T = \Sigma$. That is, for all $1 \le i, j \le p$,

$$\text{cov}(X_i, X_j) = \mathbb{E}(X_i - \mu_i)(X_j - \mu_j) = \Sigma_{ij}$$

In particular, $\text{var}(X_i) = \mathbb{E}(X_i - \mu_i)^2 = \Sigma_{ii}$.

Special case: spherical Gaussian

The X_i are independent and all have the same variance σ^2. Thus

$$\Sigma = \sigma^2 I_p$$

Special case: spherical Gaussian

The X_i are independent and all have the same variance σ^2. Thus

$$\Sigma = \sigma^2 I_p$$

Simplified density:

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

Special case: spherical Gaussian

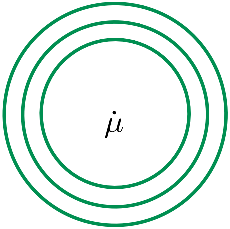
The X_i are independent and all have the same variance σ^2. Thus

$$\Sigma = \sigma^2 I_p$$

Simplified density:

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

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



Special case: spherical Gaussian

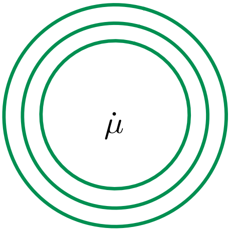
The X_i are independent and all have the same variance σ^2. Thus

$$\Sigma = \sigma^2 I_p$$

Simplified density:

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

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



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

Special case: diagonal Gaussian

The X_i are independent, with variances σ_i^2. Thus

$$\Sigma = \text{diag}(\sigma_1^2, \ldots, \sigma_p^2)$$

Special case: diagonal Gaussian

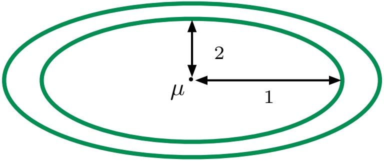
The X_i are independent, with variances σ_i^2. Thus

$$\Sigma = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_p^2)$$

Simplified density:

$$p(x) = \frac{1}{(2\pi)^{p/2}\sigma_1 \cdots \sigma_p} \exp\left(-\sum_{i=1}^{p} \frac{(x_i - \mu_i)^2}{2\sigma_i^2}\right)$$

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



Special case: diagonal Gaussian

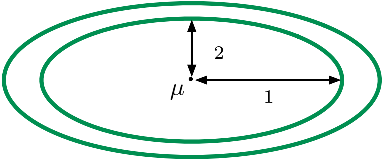
The X_i are independent, with variances σ_i^2. Thus

$$\Sigma = \text{diag}(\sigma_1^2, \ldots, \sigma_p^2)$$

Simplified density:

$$p(x) = \frac{1}{(2\pi)^{p/2}\sigma_1 \cdots \sigma_p} \exp\left(- \sum_{i=1}^{p} \frac{(x_i - \mu_i)^2}{2\sigma_i^2} \right)$$

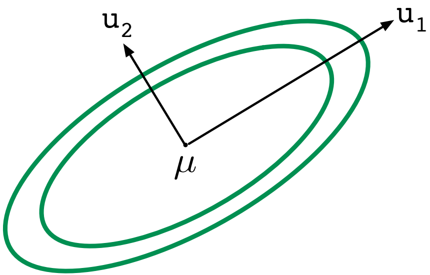
Contours of equal density are axis-aligned ellipsoids centered at μ:



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

The general Gaussian

$N(\mu, \Sigma)$:

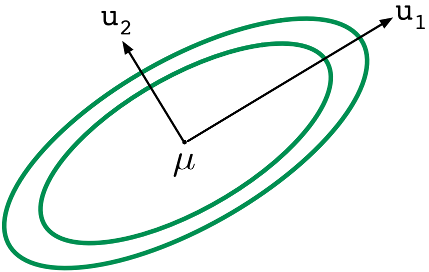


Eigendecomposition of Σ:

- Eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$
- Corresponding eigenvectors $u_1, \ldots, u_p$

The general Gaussian

$N(\mu, \Sigma)$:



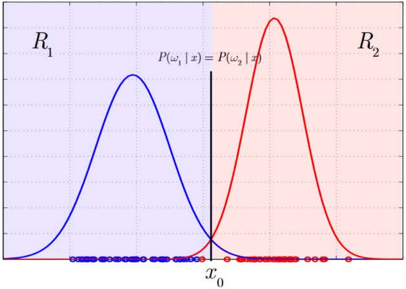
Eigendecomposition of Σ:

- Eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$
- Corresponding eigenvectors $u_1, \ldots, u_p$

$N(\mu, \Sigma)$ is simply a rotated version of $N(\mu, \text{diag}(\lambda_1, \ldots, \lambda_p))$.

Binary classification with Gaussian generative model

Example: Two classes: ω_1 and ω_2 with $N(\mu_i,\sigma_i)$, respectively



- *Red and blue dots are the training data. Estimate the mean and variance of distributions of each class.*
- *Decision threshold is x_o assuming that class prob are equal $\pi_1 = \pi_2$)*

Binary classification with Gaussian generative model

Estimate class probabilities π_1, π_2 and fit a Gaussian to each class:

$$P_1 = N(\mu_1, \Sigma_1), P_2 = N(\mu_2, \Sigma_2)$$

E.g. If data points $x^{(1)}, \ldots, x^{(m)} \in \mathbb{R}^p$ are class 1:

$$\mu_1 \; = \; \frac{1}{m}\left(x^{(1)} + \cdots + x^{(m)}\right) \quad \text{and} \quad \Sigma_1 \; = \; \frac{1}{m}\sum_{i=1}^{m}(x^{(i)} - \mu_1)(x^{(i)} - \mu_1)^T$$

Binary classification with Gaussian generative model

Estimate class probabilities π_1, π_2 and fit a Gaussian to each class:

$$P_1 = N(\mu_1, \Sigma_1), P_2 = N(\mu_2, \Sigma_2)$$

E.g. If data points $x^{(1)}, \ldots, x^{(m)} \in \mathbb{R}^p$ are class 1:

$$\mu_1 \ = \ \frac{1}{m}\left(x^{(1)} + \cdots + x^{(m)}\right) \quad \text{and} \quad \Sigma_1 \ = \ \frac{1}{m}\sum_{i=1}^{m}(x^{(i)} - \mu_1)(x^{(i)} - \mu_1)^T$$

Given a new point x, predict class 1 iff:

$$\pi_1 P_1(x) > \pi_2 P_2(x)$$

Binary classification with Gaussian generative model

Estimate class probabilities π_1, π_2 and fit a Gaussian to each class:

$$P_1 = N(\mu_1, \Sigma_1), P_2 = N(\mu_2, \Sigma_2)$$

If data points $x^{(1)}, \ldots, x^{(m)} \in \mathbb{R}^p$ are class 1, e.g.,:

$$\mu_1 = \frac{1}{m}\left(x^{(1)} + \cdots + x^{(m)}\right) \quad \text{and} \quad \Sigma_1 = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)} - \mu_1)(x^{(i)} - \mu_1)^T$$

π_1, π_2, μ_2 and Σ_2 are estimated from the training data in a similar manner .

Given a new point x, predict class 1 iff:

$$\pi_1 P_1(x) > \pi_2 P_2(x) \quad \Leftrightarrow \quad x^T M x + 2w^T x \geq \theta,$$

where:

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

and θ is a constant depending on the various parameters.

Binary classification with Gaussian generative model

Estimate class probabilities π_1, π_2 and fit a Gaussian to each class:

$$P_1 = N(\mu_1, \Sigma_1), P_2 = N(\mu_2, \Sigma_2)$$

E.g. If data points $x^{(1)}, \ldots, x^{(m)} \in \mathbb{R}^p$ are class 1:

$$\mu_1 = \frac{1}{m}\left(x^{(1)} + \cdots + x^{(m)}\right) \quad \text{and} \quad \Sigma_1 = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)} - \mu_1)(x^{(i)} - \mu_1)^T$$

or divide by (m-1) instead of m

Given a new point x, predict class 1 iff:

$$\pi_1 P_1(x) > \pi_2 P_2(x) \quad \Leftrightarrow \quad x^T M x + 2w^T x \geq \theta,$$

where:

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

and θ is a constant depending on the various parameters.

$\Sigma_1 = \Sigma_2$: linear decision boundary. Otherwise, quadratic boundary.

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

Linear decision boundary: choose class 1 iff

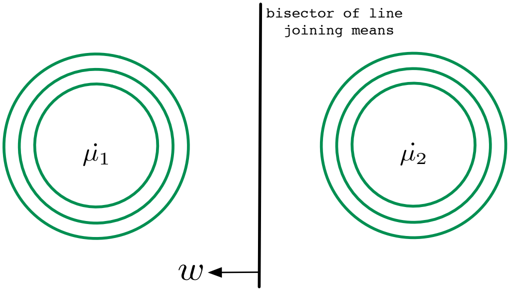
$$x \cdot \underbrace{\Sigma^{-1}(\mu_1 - \mu_2)}_{w} \geq \theta.$$

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

Linear decision boundary: choose class 1 iff

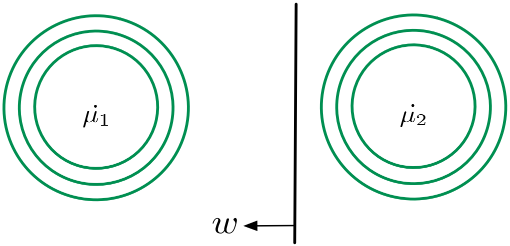
$$x \cdot \underbrace{\Sigma^{-1}(\mu_1 - \mu_2)}_{w} \geq \theta.$$

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

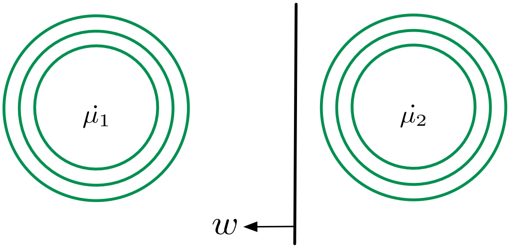


Example 2: Again spherical, but now $\pi_1 > \pi_2$.

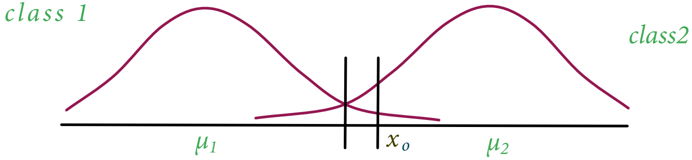
Example 2: Again spherical, but now $\pi_1 > \pi_2$.



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

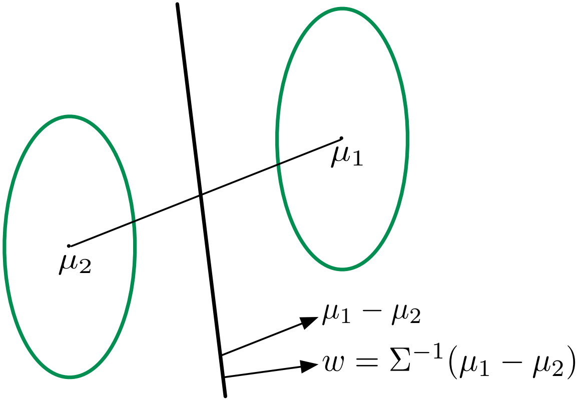


1-D example:

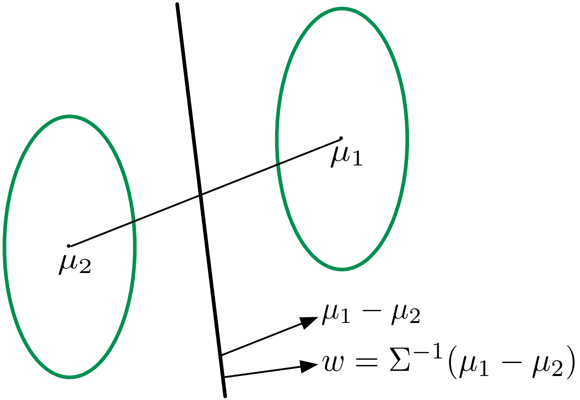


Decision boundary is x_o is closer to the mean of class 2 because $\pi_1 > \pi_2$

Example 3: Non-spherical.



Example 3: Non-spherical.



Rule: $w \cdot x \geq \theta$

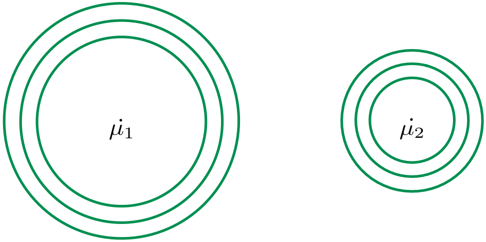
- w, θ dictated by probability model, assuming it is a perfect fit
- Common practice: choose w as above, but fit θ to minimize training/validation error

Different covariances: $\Sigma_1 \neq \Sigma_2$

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

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

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

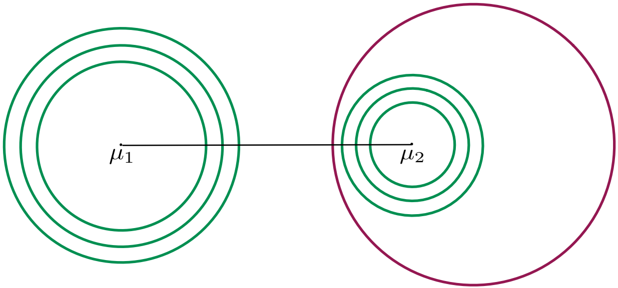


Different covariances: $\Sigma_1 \neq \Sigma_2$

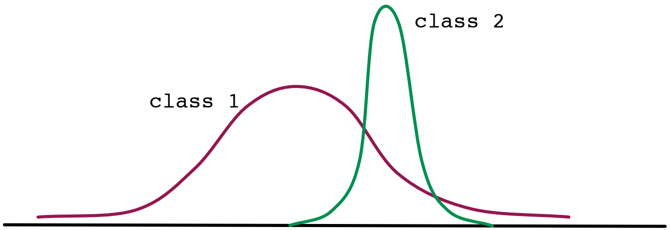
Quadratic boundary: choose class 1 iff $x^T M x + 2w^T x \geq \theta$, where:

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

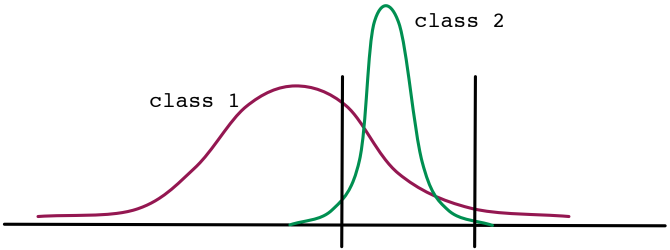
Example 1: $\Sigma_1 = \sigma_1^2 I_p$ and $\Sigma_2 = \sigma_2^2 I_p$ with $\sigma_1 > \sigma_2$



Example 2: Same thing in 1-d. $\mathcal{X} = \mathbb{R}$.

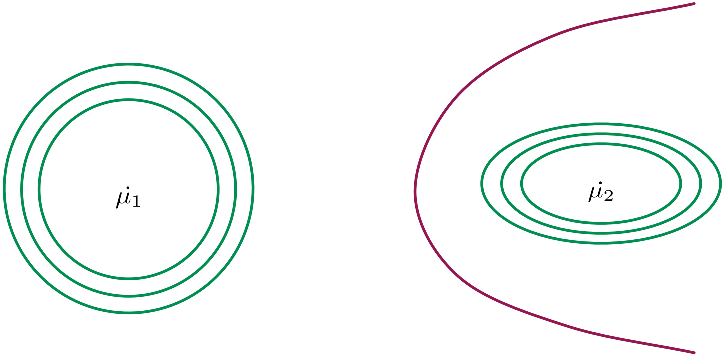


Example 2: 1-D example. $\mathcal{X} = \mathbb{R}$.

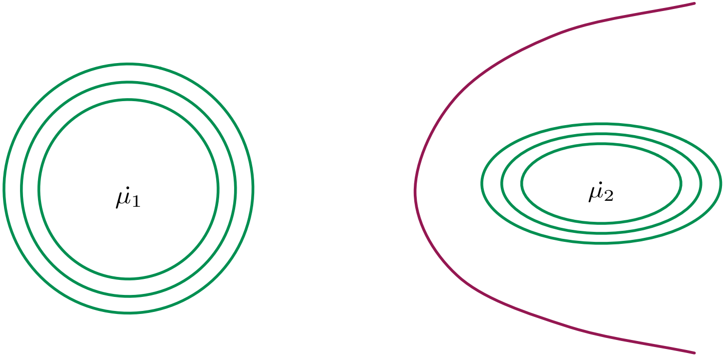


You may have two decision thresholds

Example 3: A parabolic boundary.



Example 3: A parabolic boundary.



Many other possibilities!

Multiclass discriminant analysis

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

Multiclass discriminant analysis

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

Each class has an associated **quadratic** function

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

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

Multiclass discriminant analysis

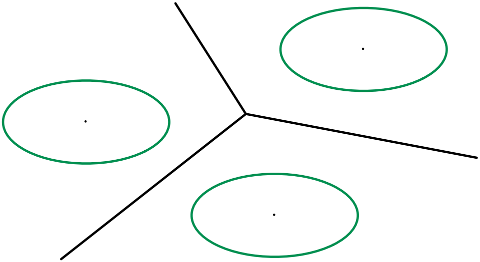
k classes: weights π_j, class-conditional distributions $P_j = N(\mu_j, \Sigma_j)$.

Each class has an associated **quadratic** function

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

To class a point x, solve $\arg\max_j f_j(x)$.

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



Fisher's linear discriminant

A framework for linear classification without Gaussian assumptions.

Fisher's linear discriminant

A framework for linear classification without Gaussian assumptions.

Use only first- and second-order statistics of the classes.

Class 1	Class 2
mean μ_1	mean μ_2
cov Σ_1	cov Σ_2
# pts n_1	# pts n_2

Fisher's linear discriminant

A framework for linear classification without Gaussian assumptions.

Use only first- and second-order statistics of the classes.

Class 1	Class 2
mean μ_1	mean μ_2
cov Σ_1	cov Σ_2
# pts n_1	# pts n_2

A linear classifier projects all data onto a direction w. Choose w so that:

Fisher's linear discriminant

A framework for linear classification without Gaussian assumptions.

Use only first- and second-order statistics of the classes.

Class 1	Class 2
mean μ_1	mean μ_2
cov Σ_1	cov Σ_2
# pts n_1	# pts n_2

A linear classifier projects all data onto a direction w. Choose w so that:

- Projected means are well-separated, i.e. $(w \cdot \mu_1 - w \cdot \mu_2)^2$ is large.

Fisher's linear discriminant

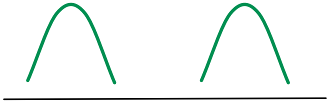
A framework for linear classification without Gaussian assumptions.

Use only first- and second-order statistics of the classes.

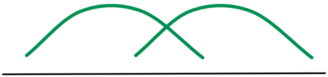
Class 1	Class 2
mean μ_1	mean μ_2
cov Σ_1	cov Σ_2
# pts n_1	# pts n_2

A linear classifier projects all data onto a direction w. Choose w so that:

1. Projected means are well-separated, i.e. $(w \cdot \mu_1 - w \cdot \mu_2)^2$ is large.
2. Projected within-class variance is small.



better than



Fisher LDA (linear discriminant analysis)

Two classes: means μ_1, μ_2; covariances Σ_1, Σ_2; sample sizes n_1, n_2.

Fisher LDA (linear discriminant analysis)

Two classes: means μ_1, μ_2; covariances Σ_1, Σ_2; sample sizes n_1, n_2.

Project data onto direction (unit vector) w.

- Projected means: $w \cdot \mu_1$ and $w \cdot \mu_2$
- Projected variances: $w^T \Sigma_1 w$ and $w^T \Sigma_2 w$
- Average projected variance:

$$\frac{n_1(w^T \Sigma_1 w) + n_2(w^T \Sigma_2 w)}{n_1 + n_2} = w^T \Sigma w,$$

where $\Sigma = (n_1 \Sigma_1 + n_2 \Sigma_2)/(n_1 + n_2)$.

Fisher LDA (linear discriminant analysis)

Two classes: means μ_1, μ_2; covariances Σ_1, Σ_2; sample sizes n_1, n_2.

Project data onto direction (unit vector) w.

- Projected means: $w \cdot \mu_1$ and $w \cdot \mu_2$
- Projected variances: $w^T \Sigma_1 w$ and $w^T \Sigma_2 w$
- Average projected variance:

$$\frac{n_1(w^T \Sigma_1 w) + n_2(w^T \Sigma_2 w)}{n_1 + n_2} = w^T \Sigma w,$$

where $\Sigma = (n_1 \Sigma_1 + n_2 \Sigma_2)/(n_1 + n_2)$.

Find w to maximize $J(w) = \dfrac{(w \cdot \mu_1 - w \cdot \mu_2)^2}{w^T \Sigma w}$

Fisher LDA (linear discriminant analysis)

Two classes: means μ_1, μ_2; covariances Σ_1, Σ_2; sample sizes n_1, n_2.

Project data onto direction (unit vector) w.

- Projected means: $w \cdot \mu_1$ and $w \cdot \mu_2$
- Projected variances: $w^T \Sigma_1 w$ and $w^T \Sigma_2 w$
- Average projected variance:

$$\frac{n_1(w^T \Sigma_1 w) + n_2(w^T \Sigma_2 w)}{n_1 + n_2} = w^T \Sigma w,$$

where $\Sigma = (n_1 \Sigma_1 + n_2 \Sigma_2)/(n_1 + n_2)$.

Find w to maximize $J(w) = \dfrac{(w \cdot \mu_1 - w \cdot \mu_2)^2}{w^T \Sigma w}$

Solution: $w \propto \Sigma^{-1}(\mu_1 - \mu_2)$. Look familiar?