Lecture 18:

$$\text{LDA} \longrightarrow \Pr\{Y=k \mid X=x\} \approx \frac{\underbrace{\hat{f}_k(x)}_{N(\mu_k,\,\sigma^2)}\;\hat{\pi}_k}{\Pr\{X=x\}}$$

$$\hat{\pi}_k = \frac{\#\{i : y_i = k\}}{n}$$

(an estimate for $\Pr\{y=k\}$)

$$\frac{1}{\sqrt{2\pi}\,\hat{\sigma}}\; e^{-\frac{(x-\hat{\mu}_k)^2}{2\hat{\sigma}^2}}$$

(an estimate for $\Pr\{X=x \mid y=k\}$)

At prediction time, given a new input x:

$$\hat{y}(x) = \underset{k}{\operatorname{argmax}}\ \Pr\{y=k \mid X=x\}$$

$$= \underset{k}{\arg\max} \frac{\hat{f}_k(x) \cdot \hat{\pi}_k}{\cancel{P(X=x)}}$$

the same for every class k

$$= \underset{k}{\arg\max} \ \hat{f}_k(x) \ \hat{\pi}_k$$

$$\frac{1}{\sqrt{2\pi}\hat{\sigma}} e^{-\frac{(x-\hat{\mu}_k)^2}{2\hat{\sigma}^2}}$$

$$= \underset{k}{\arg\max} \left\{ \log \hat{f}_k(x) + \log \hat{\pi}_k \right\}$$

$$= \underset{k}{\arg\max} \left\{ \log \frac{1}{\sqrt{2\pi}\hat{\sigma}} - \frac{(x-\hat{\mu}_k)^2}{2\hat{\sigma}^2} + \log \hat{\pi}_k \right\}$$

Because of the assumption that the variances were the same over all the classes, we can remove this quadratic term $\frac{x^2}{2\hat{\sigma}^2}$

$$= \underset{k}{\arg\max} \left\{ \cancel{\log \frac{1}{\sqrt{2\pi}\hat{\sigma}}} \cancel{+ \frac{x^2}{2\hat{\sigma}^2}} - \frac{\hat{\mu}_k^2}{2\hat{\sigma}^2} + \frac{x\hat{\mu}_k}{\hat{\sigma}^2} + \log \hat{\pi}_k \right\}$$

do not depend on the class k

$$= \underset{k}{\arg\max} \left\{ \underbrace{-\frac{\hat{\mu}_k^2}{2\hat{\sigma}^2} + \frac{x\hat{\mu}_k}{\hat{\sigma}^2} + \log \hat{\pi}_k}_{\text{linear in } x} \right\}$$

- Since the decision rule is linearly dependent on x, this classifier is called "linear" discriminant analysis.

Extension to the multi-dimensional setting:

$$\text{Data} = \{ (x_i, y_i) \}_{i=1}^{n} \quad \begin{cases} y_i \in \{1, \dots, K\} \\ x_i \in \mathbb{R}^p \end{cases}$$

$$\Pr\{ y = k \mid X = x \}$$

$$= \frac{\underbrace{\Pr\{ X = x \mid y = k \}}_{f_k(x) \sim \mathcal{N}(\vec{\mu}_k, \Sigma)} \cdot \underbrace{\Pr\{ y = k \}}_{\pi_k}}{\Pr\{ X = x \}}$$

the same over all the classes

AAW

Multi-variate Gaussian Distribution

$(\mathcal{N}(\mu, \Sigma))$

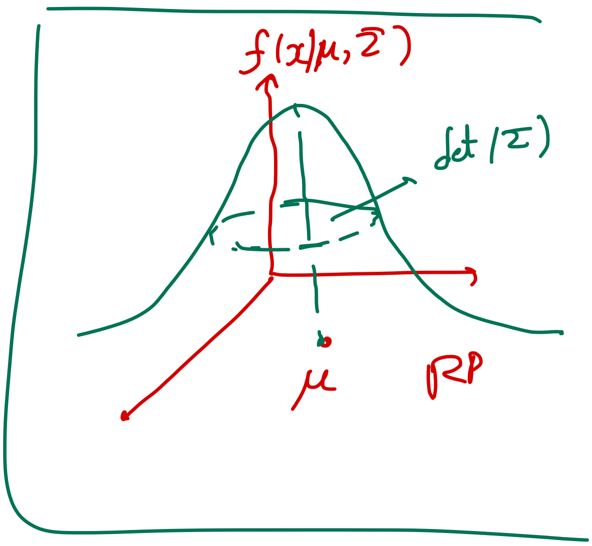
$\mu \in \mathbb{R}^p$, Σ: $p \times p$ matrix

$$f(x \mid \underset{=}{\mu}, \underset{=}{\Sigma})$$

mean (μ), covariance matrix (Σ)

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

transpose (T)



properties:

$$X \sim \mathcal{N}(\mu, \Sigma)$$

$$X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{pmatrix} \in \mathbb{R}^p$$

$$E[X] = \mu, \quad \underbrace{E[(X-\mu)(X-\mu)^T] = \Sigma}_{\forall\, r,s \in \{1,\dots,p\}}$$

$$E[(X_r - \mu_r)(X_s - \mu_s)] \overset{?}{=} \Sigma_{r,s}$$

$$X = \begin{pmatrix} X_1 \\ \vdots \\ X_r \\ \vdots \\ X_s \\ \vdots \\ X_p \end{pmatrix}$$

$E[X_r] = \mu_r$

$E[X_s] = \mu_s$

$$E[(X_r-\mu_r)(X_s-\mu_s)]$$

$$= \Sigma_{r,s}$$

$$\underbrace{\begin{array}{c|ccc} & & & s \\ \hline r & & & \Sigma_{rs} \\ & & & \end{array}}_{\text{matrix } \Sigma}\ p\times p$$

$$\pi_k = Pr\{y=k\} \xrightarrow{\text{estimator}} \hat{\pi}_k = \frac{\overbrace{\#\{i : y_i = k\}}^{n_k}}{\text{total number of data points}}$$

$$f_k(x) = N(\mu_k, \Sigma)$$

$$\hat{\mu}_k = \frac{\sum_{i:y_i=k} x_i}{n_k}$$

$n_k \rightarrow$ number of data points with label k

$$\hat{\Sigma} = \begin{array}{c} \\ r \end{array} \begin{array}{c} s \\ \begin{pmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \hat{\Sigma}_{rs} & - & \cdot \\ - & - & - & \end{pmatrix}_{p \times p} \end{array}$$

$$\hat{\Sigma}_{rs} = \frac{\sum_{k=1}^{K} \sum_{i:\, y_i = k} (x_{i,r} - \hat{\mu}_{k,r})(x_{i,s} - \hat{\mu}_{k,s})}{n-K}$$

$$\boxed{\begin{array}{l} \text{we estimate } \hat{\mu}_k \text{ and } \hat{\Sigma} \text{ and } \hat{\pi}_k \\ \text{from data} \\ \qquad \Rightarrow \text{an estimate for} \\ \qquad \text{the conditional probabilities} \end{array}}$$

$$-\ \Pr\{ y = k \mid X = x \} = \frac{\Pr\{X = x \mid y = k\} \Pr\{y = k\}}{P(X = x)}$$

$$\overset{\text{estimated}}{\approx} \frac{\hat{f}(x \mid \hat{\mu}_k, \hat{\Sigma})\ \hat{\pi}_k}{P(X = x)}$$

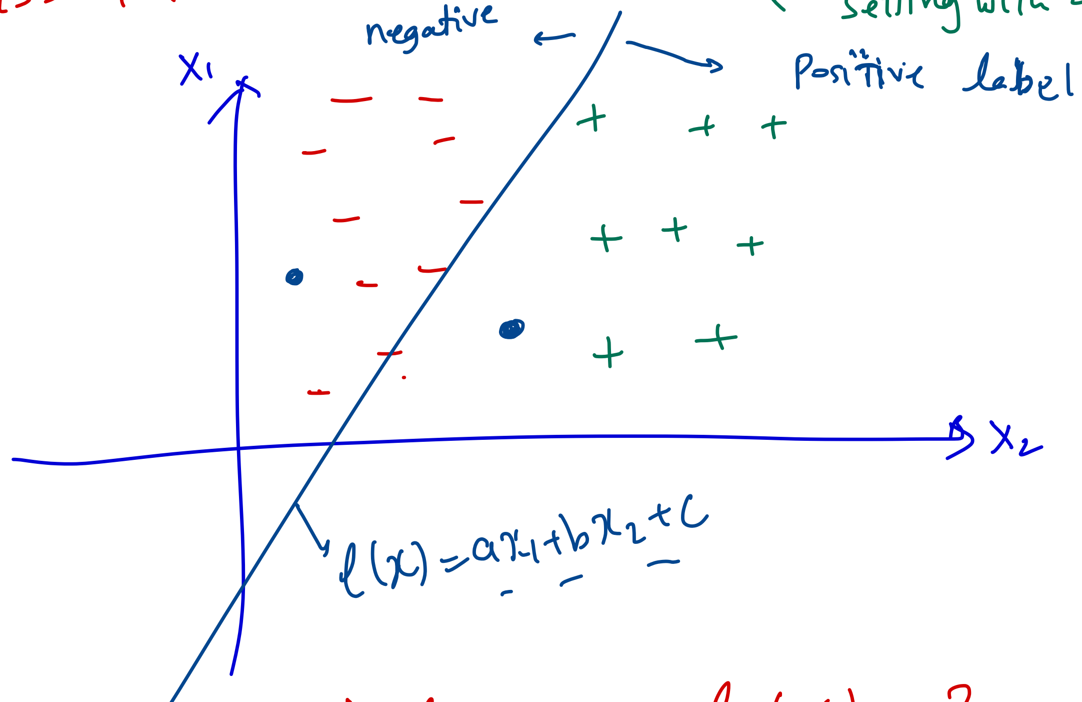
$\Rightarrow$ p.

- Prediction of the label at a new data point x:

$$\hat{y}(x) = \arg\max_{k} \left\{ \hat{f}(x \mid \hat{\mu}_k, \hat{\Sigma})\, \hat{\pi}_k \right\}$$

$$= \arg\max_{k} \left\{ \underbrace{\overbrace{x^T \hat{\Sigma}^{-1} \hat{\mu}_k - \frac{1}{2} \hat{\mu}_k^T \hat{\Sigma}^{-1} \hat{\mu}_k + \log \hat{\pi}_k}^{\ell_k(x)}}_{\ell_k(x) \longrightarrow \text{linear in } x} \right\}$$

This will lead to what we call linear classification boundaries. (let's consider a binary classification setting with 2-d data)



LDA : $\arg\max \{ \ell_+(x) \,,\, \ell_-(x) \}$

$$\Downarrow$$

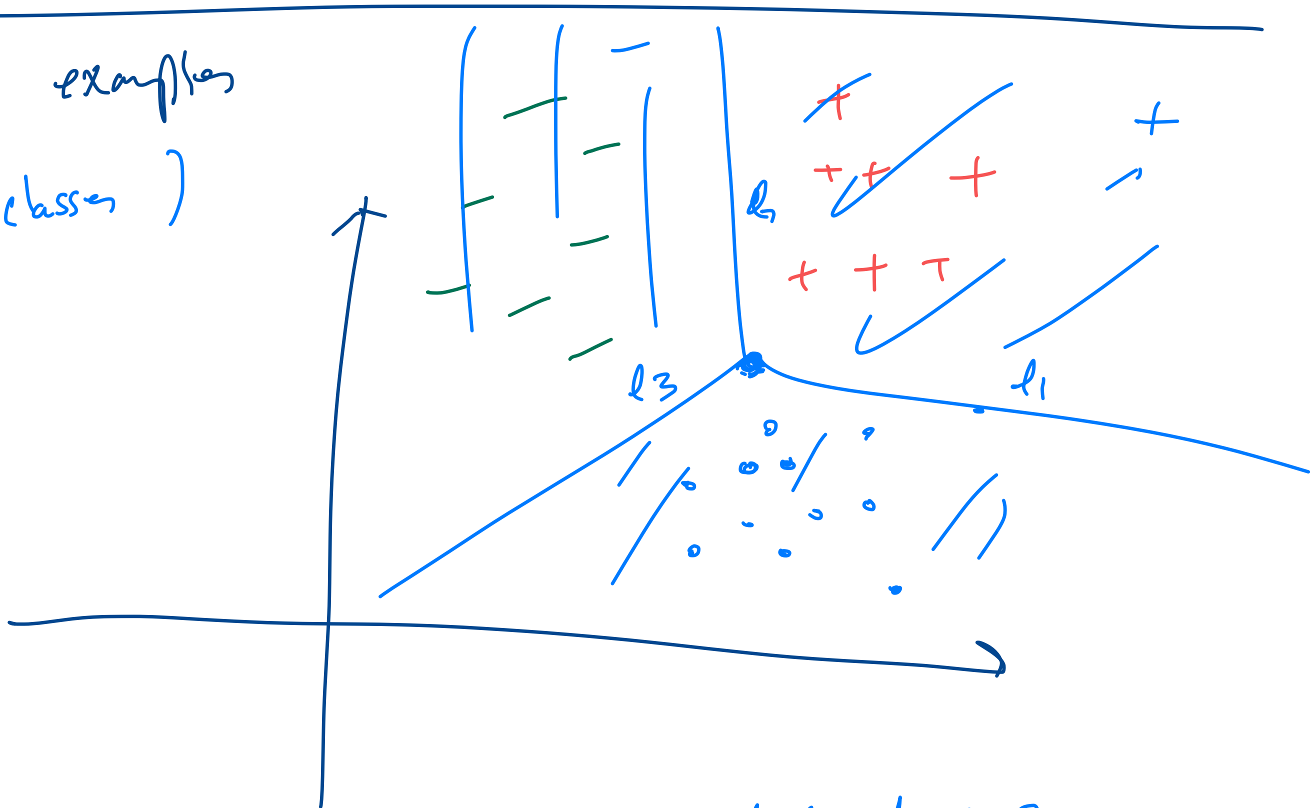
$$\begin{cases} \text{if } \ell_+(x) > \ell_-(x) \Rightarrow \hat{y}(x) = + \\ \text{if } \ell_+(x) \le \ell_-(x) \Rightarrow \hat{y}(x) = - \end{cases}$$

$$\Downarrow$$

$$\boxed{\underbrace{\ell_+(x) - \ell_-(x)}_{\triangleq\, \ell(x) \;=\; ax_1 + bx_2 + c} \;\overset{\hat{y}(x)=+}{\underset{\hat{y}(x)=-}{\gtrless}}\; 0}$$

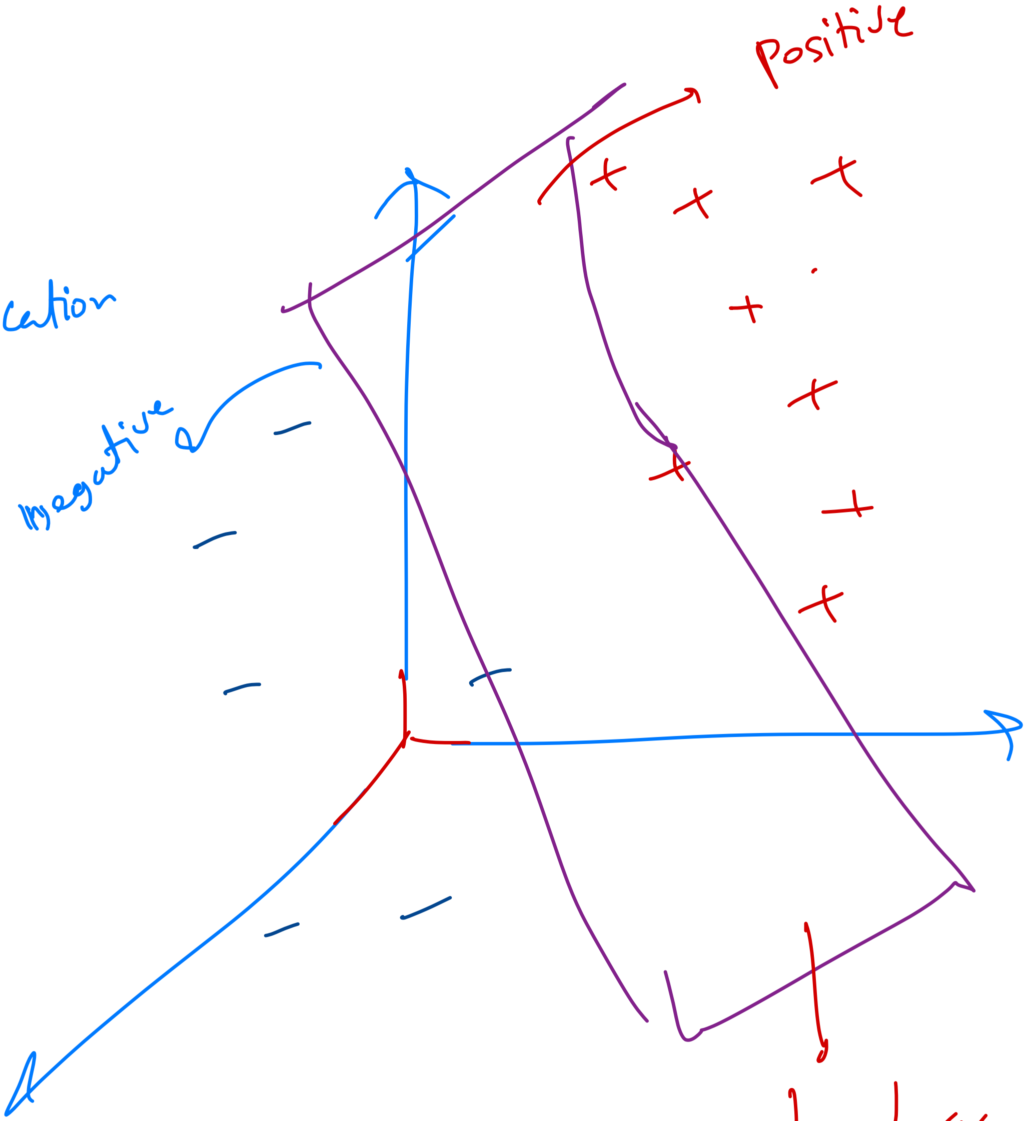
other examples

(3 classes)



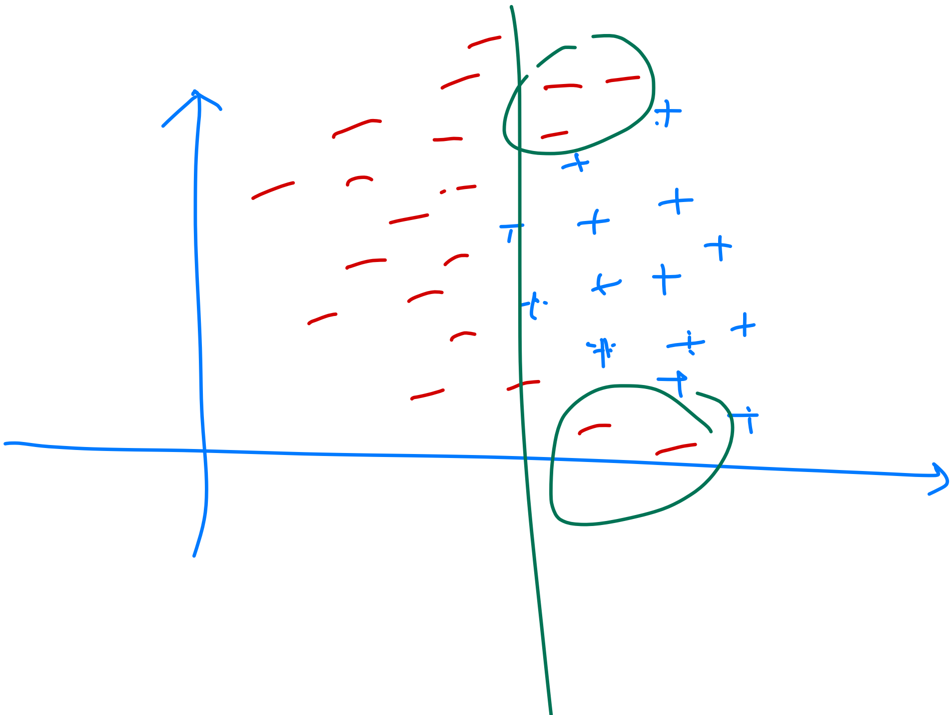
$$\hat{y}(x) = \operatorname{argmax} \{ \ell_+(x), \ell_-(x), \ell_0(x) \}$$

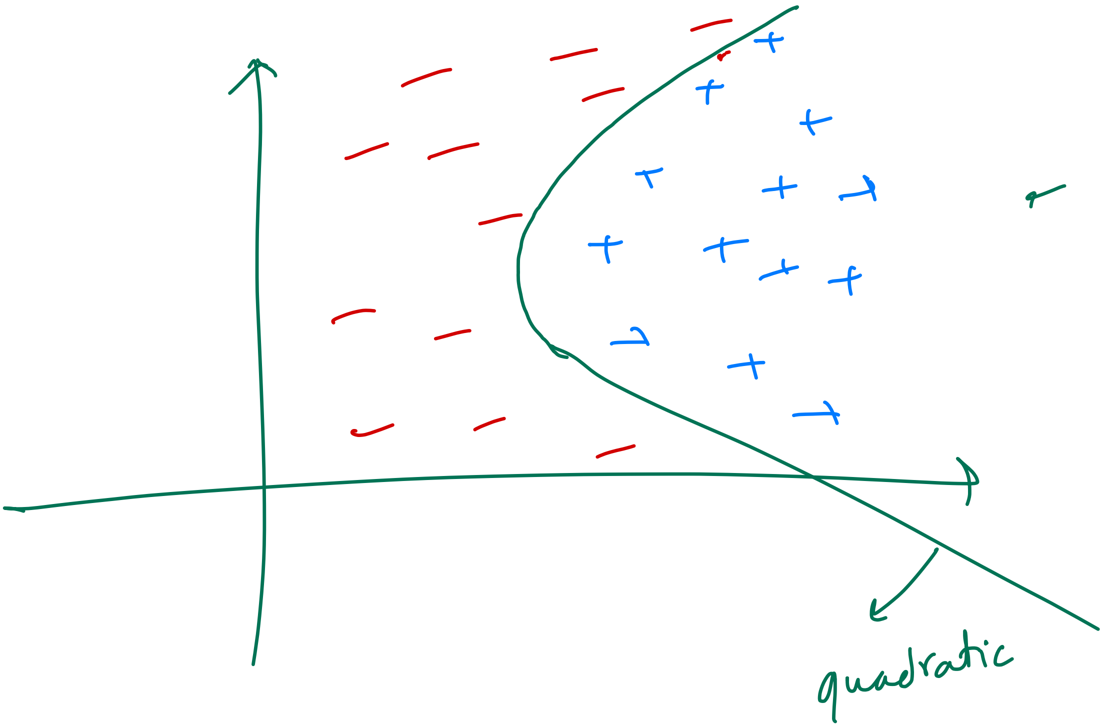
3-dim binary classification



boundary will be a hyperplane

In what cases can LDA be ineffective?





Quadratic Discriminant Analysis:

- Similar to LDA except that the variance can change across the classes.

- for each class k : $f_k(x) = N(\mu_k, \Sigma_k)$

learned using data from class k.

quadratic term

$$\hat{y}_{QDA}(x) = \underset{k}{\operatorname{argmax}} \left\{ \underbrace{-\tfrac{1}{2}(x-\hat{\mu}_k)^T \Sigma_k^{-1} (x-\hat{\mu}_k)} - \tfrac{1}{2}\log\det(\hat{\Sigma}_k) + \log\hat{\pi}_k \right\}$$

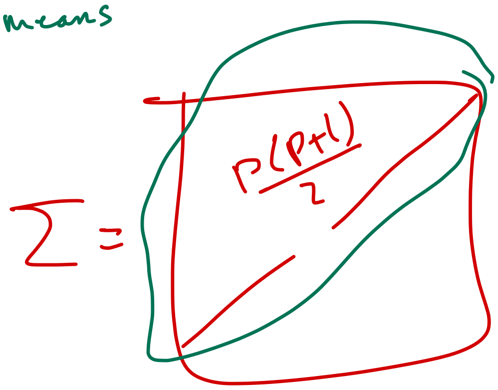
- why QDA is a more complex class than LDA ?

LDA

→ learn the mean vector per class + the covariance matrix.

$$\underbrace{K \cdot p}_{\text{means}} + \frac{p(p+1)}{2} \rightarrow \text{covariance matrix}$$

→ symmetric matrix



QDA

- 1 mean vector per class
- 1 covariance matrix per class

$$K \cdot p + K \frac{p(p+1)}{2}$$

$$Kp + \frac{p(p+1)}{2} \ll Kp + \frac{Kp(p+1)}{2}$$

↓

when p is large

LDA	QDA
→ less parameters → less flexibility / complexity might underfit	might overfit

LDA is a better bet than QDA if there are relatively few training data points. In contrast, QDA is a better choice if the number of training data points is large or if the assumption of a common covariance matrix for all the k classes is off.

(in this case LDA will underfit)

Module 4 : Unsupervised learning:

Supervised learning: $\{(\underline{\underline{x_i}}, \underline{\underline{y_i}})\}_{i=1}^{n} \longrightarrow y_i \approx \underline{\tilde{f}}(x_i)$

unsupervised learning $\{x_i\}_{i=1}^{n}$