

STK2100- Machine Learning and Statistical Methods for Prediction and Classification

Spring 2026

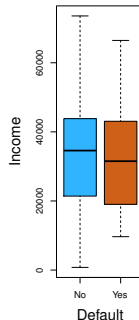
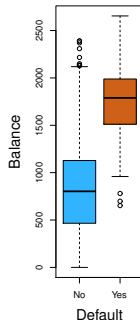
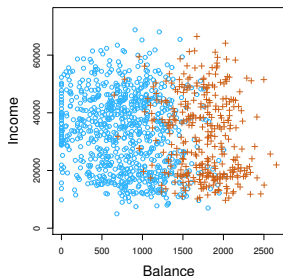
Linear models for classification

Covers Sections 4.1, 4.2, 4.3 (except 4.3.3) and 4.4
(except 4.4.3)

Ingrid Hobæk Haff
Department of Mathematics
University of Oslo

Example: Credit default data

- Response: Will the customer default on the credit payment
Yes/No (Default)
- Explanatory variables:
 - Annual income (`income`)
 - Monthly credit card balance (`balance`)
 - Is a student Yes/No (`student`)
- We want to predict whether a customer will default on his/her credit card based on the income and credit balance



Classification

- What characterises **classification** is that the response Y is qualitative/categorical
- The data are of the type $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$,
 $y_i \in \{1, \dots, K\}$
- Recall that the solution that minimises the 0 – 1 loss
 $L(y, k) = I(y \neq k)$ is the **Bayes classifier**
 $\hat{Y}(\mathbf{x}) = \arg \max_k \Pr(Y = k | \mathbf{X} = \mathbf{x})$
- A probability is a quantitative measure, and methods for analyzing probabilities very similar to methods for regression
- Chapter 4 treats linear regression for classification, logistic regression and linear and quadratic discriminant analysis
- Later chapters introduce others methods for classification, namely generalised additive models, decision trees, boosting, random forests and neural nets

Why not use linear regression?

- Assume first that $K = 2$, such as in the credit default example
- Define the quantitative variable Z as $Z = 1$ if Y is in the second class and $Z = 0$ when Y is in the first class
- Then we can use a linear regression model with Z as the response

$$Z_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_p x_{ip} + \varepsilon$$

- The prediction for Z with corresponding covariate vector $\mathbf{x}$ is then $\hat{z} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \dots + \hat{\beta}_p x_p$
- Based on this prediction, we can make the classification

$$\hat{y} = \begin{cases} \text{Class 2} & \text{if } \hat{z} > \tau; \\ \text{Class 1} & \text{otherwise} \end{cases},$$

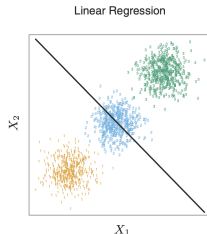
where τ is a chosen threshold

Why not use linear regression?

- However, this approach has some weaknesses:
 - It is ad hoc
 - How do we interpret $\hat{z} < 0$ or $\hat{z} > 1$?
 - It is not obvious how to generalise to $K > 2$, i.e. multiple classes, as illustrated below
- Consider an example with $K = 3$:

$$Y = \begin{cases} 1 & \text{if stroke} \\ 2 & \text{if overdose} \\ 3 & \text{if epileptic seizures} \end{cases}$$

The figure shows the result when fitting a linear regression model with numeric response



- **Masking**, i.e ignoring a class, can occur

Example with multiple classes

- Further, we get a different answer if we switch the labels, for instance to

$$Y = \begin{cases} 1 & \text{if epileptic seizures} \\ 2 & \text{if stroke} \\ 3 & \text{if overdose} \end{cases}$$

- It is also difficult to convert a categorical variable to a quantitative variable
- An alternative is to define the dummy variables

$$\begin{cases} Y_1 = I(\text{if stroke}) \\ Y_2 = I(\text{if overdose}) \\ Y_3 = I(\text{if epileptic seizures}) \end{cases}$$

so that $Y = \arg \max_j Y_j$

- Then we regress Y_1 , Y_2 and Y_3 based on $\mathbf{x}$, and classify as $\hat{Y} = \arg \max_{j \in \{1,2,3\}} (\hat{Y}_1(\mathbf{x}), \hat{Y}_2(\mathbf{x}), \hat{Y}_3(\mathbf{x}))$
- A better option is to construct methods directly tuned towards classification

Logistic regression

- **Logistic regression** is one of the most used classification models for **binary classification**, i.e. $K = 2$, with the response coded as $Y \in \{0, 1\}$
- Data: $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$
- Logistic regression model:

$$Y \stackrel{ind.}{\sim} \text{Bernoulli}(p(\mathbf{x})) \rightarrow \Pr(Y = y|\mathbf{x}) = p(\mathbf{x})^y (1 - p(\mathbf{x}))^{1-y}$$

with

$$p(\mathbf{x}) = \Pr(Y = 1|\mathbf{x}) = \frac{e^{\mathbf{x}^T \boldsymbol{\beta}}}{1 + e^{\mathbf{x}^T \boldsymbol{\beta}}} = \frac{e^{\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p}}{1 + e^{\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p}}$$

- The logistic regression model is a member of the **generalised linear models** (GLM)

Logistic regression

- The interpretation of the β s in the logistic regression model is related to the **odds** $\Pr(Y = 1|\mathbf{x}) / \Pr(Y = 0|\mathbf{x})$
- We have

$$\text{odds}(\mathbf{x}) = \frac{\Pr(Y = 1|\mathbf{x})}{\Pr(Y = 0|\mathbf{x})} = \left(\frac{p(\mathbf{x})}{1 - p(\mathbf{x})} \right) = \frac{\frac{\exp(\mathbf{x}^T \beta)}{1 + \exp(\mathbf{x}^T \beta)}}{\frac{1}{1 + \exp(\mathbf{x}^T \beta)}} = \exp(\mathbf{x}^T \beta)$$

- For the values $\mathbf{x}_* = (x_{*1}, \dots, x_{*p})^T$ and $\mathbf{x} = (x_1, \dots, x_p)^T$ of the covariates, the **odds ratio** becomes

$$\frac{\text{odds}(\mathbf{x}_*)}{\text{odds}(\mathbf{x})} = \frac{\exp(\mathbf{x}_*^T \beta)}{\exp(\mathbf{x}^T \beta)} = \exp((\mathbf{x}_* - \mathbf{x})^T \beta)$$

- If $x_{*j} = x_j + 1$ and $x_{*k} = x_k$, for $k \neq j$, then

$$\frac{\text{odds}(\mathbf{x}_*)}{\text{odds}(\mathbf{x})} = e^{\beta_j}$$

- Thus, e^{β_j} is the relative effect on the odds of one unit increase in the j -th covariate when the other covariates remain the same
- There is no such interpretation if linear regression is used

Logistic regression and estimation

- The logistic regression model is fitted by maximum likelihood, i.e. $\hat{\beta} = \arg \max_{\beta} l(\beta)$, where $l(\beta)$ is the log-likelihood function
- The log-likelihood function is given by

$$\begin{aligned} l(\beta) &= \log \prod_{i=1}^N p(\mathbf{x}_i)^{y_i} (1 - p(\mathbf{x}_i))^{1-y_i} = \sum_{i=1}^N \left(y_i \log \left(\frac{p(\mathbf{x}_i)}{1 - p(\mathbf{x}_i)} \right) + \log(1 - p(\mathbf{x}_i)) \right) \\ &= \sum_{i=1}^N \left(y_i \mathbf{x}_i^T \beta - \log(1 + e^{\mathbf{x}_i^T \beta}) \right) \end{aligned}$$

- The solution is found by solving $\partial l(\beta) / \partial \beta = \mathbf{0}$, with

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i=1}^N (y_i - p(\mathbf{x}_i)) \mathbf{x}_i = \mathbf{X}^T (\mathbf{y} - \mathbf{p}),$$

where $\mathbf{p} = (p(\mathbf{x}_1), \dots, p(\mathbf{x}_N))^T$

Logistic regression and estimation

- Further,

$$\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T} = - \sum_{i=1}^N p(\mathbf{x}_i)(1 - p(\mathbf{x}_i)) \mathbf{x}_i \mathbf{x}_i^T = -\mathbf{X}^T \mathbf{W} \mathbf{X},$$

where $\mathbf{W}$ is the diagonal matrix with diagonal elements $p(\mathbf{x}_i)(1 - p(\mathbf{x}_i))$

- The likelihood equations have to be solved numerically, typically with the **Newton-Raphson** algorithm
- Let $\boldsymbol{\beta}^{(t)}$ be the estimate after t steps, then the update in step $t + 1$ is

$$\begin{aligned}\boldsymbol{\beta}^{(t+1)} &= \boldsymbol{\beta}^{(t)} - \left(\frac{\partial^2 l(\boldsymbol{\beta}^{(t)})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T} \right)^{-1} \frac{\partial l(\boldsymbol{\beta}^{(t)})}{\partial \boldsymbol{\beta}} \\ &= \boldsymbol{\beta}^{(t)} + (\mathbf{X}^T \mathbf{W}^{(t)} \mathbf{X})^{-1} \mathbf{X}^T (\mathbf{y} - \mathbf{p}^{(t)}) \\ &= (\mathbf{X}^T \mathbf{W}^{(t)} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W}^{(t)} \mathbf{X} \boldsymbol{\beta}^{(t)} + (\mathbf{X}^T \mathbf{W}^{(t)} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W}^{(t)} (\mathbf{W}^{(t)})^{-1} (\mathbf{y} - \mathbf{p}^{(t)}) \\ &= (\mathbf{X}^T \mathbf{W}^{(t)} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W}^{(t)} (\mathbf{X} \boldsymbol{\beta}^{(t)} + (\mathbf{W}^{(t)})^{-1} (\mathbf{y} - \mathbf{p}^{(t)})) \\ &= (\mathbf{X}^T \mathbf{W}^{(t)} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W}^{(t)} \mathbf{z}^{(t)}\end{aligned}$$

Logistic regression and inference

- Here $\mathbf{z}^{(t)} = \mathbf{X}\boldsymbol{\beta}^{(t)} + (\mathbf{W}^{(t)})^{-1}(\mathbf{y} - \mathbf{p}^{(t)})$ is denoted the **adjusted response**
- The Newton-Raphson therefore corresponds to an **iteratively reweighted least squares** (IRLS) algorithm, typically performed with the start value $\boldsymbol{\beta}^{(0)}$ until some pre-specified stopping criterion
- Under certain regularity conditions, for large N , the ML estimator has the property

$$\hat{\boldsymbol{\beta}} \stackrel{approx}{\sim} N(\boldsymbol{\beta}, \mathbf{J}^{-1}), \quad \text{with } \mathbf{J} = -E_{Y|\mathbf{X}} (\partial^2 l(\boldsymbol{\beta}) / (\partial \boldsymbol{\beta}^T \partial \boldsymbol{\beta})),$$

so that $\hat{\boldsymbol{\beta}} \stackrel{approx}{\sim} N(\boldsymbol{\beta}, \hat{\mathbf{J}}^{-1})$, with $\hat{\mathbf{J}}$ the observed Fisher information matrix given by

$$\hat{\mathbf{J}} = -\frac{\partial^2 l(\hat{\boldsymbol{\beta}})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T}$$

- This can be used to construct confidence intervals and test hypotheses

Logistic regression and class boundary

- A reasonable classification rule is

$$\hat{y} = \begin{cases} 1 & \text{if } \Pr(Y = 1|\mathbf{x}) > 0.5; \\ 0 & \text{else} \end{cases}$$

- Then, we have:

$$\begin{aligned} \Pr(Y = 1|\mathbf{x}) = \frac{e^{\mathbf{x}^T \beta}}{1 + e^{\mathbf{x}^T \beta}} > 0.5 &\iff e^{\mathbf{x}^T \beta} > 0.5(1 + e^{\mathbf{x}^T \beta}) \\ &\iff e^{\mathbf{x}^T \beta} > 1 \\ &\iff \mathbf{x}^T \beta > 0 \end{aligned}$$

- Hence, the boundary between the classes is linear

Logistic regression and regularisation

- Just like the linear regression model, the logistic regression model can be **regularised** with a ridge penalty

$$\hat{\beta} = \arg \min_{\beta} \left(- \sum_{i=1}^N \left(y_i \mathbf{x}_i^T \beta - \log \left(1 + e^{\mathbf{x}_i^T \beta} \right) \right) + \lambda \sum_{j=1}^p \beta_j^2 \right)$$

or with a Lasso penalty

$$\hat{\beta} = \arg \min_{\beta} \left(- \sum_{i=1}^N \left(y_i \mathbf{x}_i^T \beta - \log \left(1 + e^{\mathbf{x}_i^T \beta} \right) \right) + \lambda \sum_{j=1}^p |\beta_j| \right)$$

- This can for instance be done in the `glmnet` package in R

Example: Credit default data

See R code

Multinomial regression

- The logistic regression model can be expanded to a **multinomial logit regression** model for $K > 2$ classes
- This model is given by

$$\Pr(Y = k|\mathbf{x}) = \frac{\exp(\mathbf{x}^T \beta_k)}{\sum_{l=1}^K \exp(\mathbf{x}^T \beta_l)}, \quad k = 1, \dots, K-1$$
$$\Pr(Y = K|\mathbf{x}) = \frac{1}{\sum_{l=1}^K \exp(\mathbf{x}^T \beta_l)},$$

- This means that class K is used as a reference, just like class 0 in logistic regression
- The parameters $\beta_1, \dots, \beta_{K-1}$ are fitted by maximum likelihood
- The resulting estimators have similar properties as in logistic regression

Example: Alligator data

- This is a data set on food choice of 219 alligators in four Florida lakes
- Response: Primary food choice of alligator (`food`)
- Explanatory variables:
 - Lake (`lake`)
 - Size of the alligator (`size`)

Lake	Size	Primary Food Choice				
	(meters)	Fish	Invertebrate	Reptile	Bird	Other
Hancock	≤ 2.3	23	4	2	2	8
	> 2.3	7	0	1	3	5
Ocklawaha	≤ 2.3	5	11	1	0	3
	> 2.3	13	8	6	1	0
Trafford	≤ 2.3	5	11	2	1	5
	> 2.3	8	7	6	3	5
George	≤ 2.3	16	19	1	2	3
	> 2.3	17	1	0	1	3

See R code

Linear discriminant analysis (LDA)

- An alternative to specifying the probabilities $\Pr(Y = k|\mathbf{X} = \mathbf{x})$ directly, is to instead specify the distributions of $\mathbf{X}|Y = k$, $k = 1, \dots, K$, as well as the marginal class probabilities $\pi_k = \Pr(Y = k)$
- The probabilities $\Pr(Y = k|\mathbf{X} = \mathbf{x})$ are the obtained with Bayes' theorem:

$$\Pr(Y = k|\mathbf{X} = \mathbf{x}) = \frac{\pi_k p(\mathbf{x}|Y = k)}{p(\mathbf{x})} = \frac{\pi_k p_k(\mathbf{x})}{\sum_{l=1}^K \pi_l p_l(\mathbf{x})}$$

- In **linear discriminant analysis** (LDA), one assumes that $[\mathbf{X}|Y = k] \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma})$, i.e. the covariate vectors in different classes follow normal distributions with different mean vectors $\boldsymbol{\mu}_k$, but a common covariance matrix $\boldsymbol{\Sigma}$
- This means that

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

LDA and decision boundaries

- The classification rule is given by

$$\begin{aligned}\hat{y} &= \arg \max_k \Pr(Y = k | \mathbf{X} = \mathbf{x}) \\&= \arg \max_k \frac{\pi_k p_k(\mathbf{x})}{\sum_{l=1}^K \pi_l p_l(\mathbf{x})} \\&= \arg \max_k \pi_k p_k(\mathbf{x}) \\&= \arg \max_k \left(\pi_k \frac{1}{(2\pi)^{p/2} |\Sigma|^{1/2}} \exp \left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right) \right) \\&= \arg \max_k \left(\log(\pi_k) - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right) \\&= \arg \max_k \left(\log(\pi_k) - \frac{1}{2} \mathbf{x}^T \Sigma^{-1} \mathbf{x} + \boldsymbol{\mu}_k^T \Sigma^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_k^T \Sigma^{-1} \boldsymbol{\mu}_k \right) \\&= \arg \max_k \left(\boldsymbol{\mu}_k^T \Sigma^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_k^T \Sigma^{-1} \boldsymbol{\mu}_k + \log(\pi_k) \right) = \arg \max_k \delta_k(\mathbf{x})\end{aligned}$$

- The classification rule is then in practice to classify to the largest estimated **discriminant function** $\delta_k(\mathbf{x})$

LDA and decision boundaries

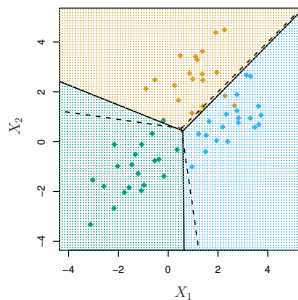
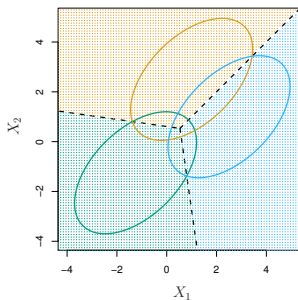
- This means that

$$\delta_k(\mathbf{x}) = \delta_l(\mathbf{x})$$



$$(\boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} - \boldsymbol{\mu}_l^T \boldsymbol{\Sigma}^{-1}) \mathbf{x} = \frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \boldsymbol{\mu}_l^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_l - \log(\pi_k) + \log(\pi_l)$$

- Hence, LDA also leads to linear decision boundaries between classes



LDA and estimation

- The unknown parameters of the LDA model are $\{\pi_k, \mu_k, k = 1, \dots, K\}$ and Σ
- Assuming we have training data $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$, these parameters can be estimated with **method of moments** type estimators given by

$$\hat{\pi}_k = \frac{N_k}{N}, \quad \text{with } N_k = \sum_{i=1}^N I(y_i = k)$$

$$\hat{\mu}_k = \frac{1}{N_k} \sum_{i: y_i = k} \mathbf{x}_i$$

$$\hat{\Sigma} = \frac{1}{N - K} \sum_{k=1}^K \sum_{i: y_i = k} (\mathbf{x}_i - \hat{\mu}_k)(\mathbf{x}_i - \hat{\mu}_k)^T$$

- An alternative is to estimate the parameters with the maximum likelihood estimator

Example: Credit default data

See R code

Quadratic discriminant analysis (QDA)

- In LDA, one makes the assumption that the covariance matrix of $\mathbf{X}|Y = k$ is the same for all classes k , namely Σ
- A more flexible model is obtained by letting each class have its own covariance matrix Σ_k
- This means that

$$p_k(\mathbf{x}) = \frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}} \exp \left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right)$$

QDA and decision boundaries

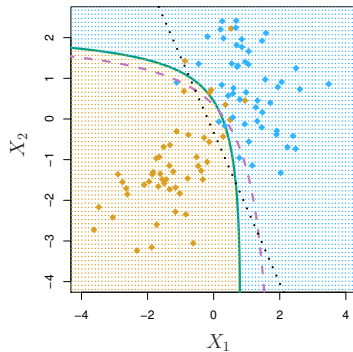
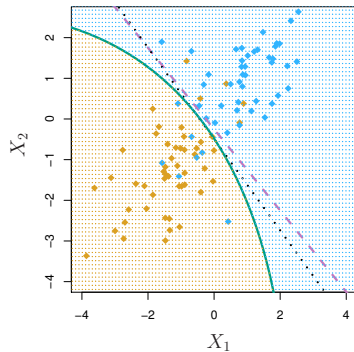
- The classification rule is given by

$$\begin{aligned}\hat{y} &= \arg \max_k \pi_k p_k(\mathbf{x}) \\&= \arg \max_k \left(\pi_k \frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}} \exp \left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right) \right) \\&= \arg \max_k \left(\log(\pi_k) - \frac{1}{2} \log |\Sigma_k| - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right) \\&= \arg \max_k \left(-\frac{1}{2} \mathbf{x}^T \Sigma_k^{-1} \mathbf{x} + \boldsymbol{\mu}_k^T \Sigma_k^{-1} \mathbf{x} \log(\pi_k) - \frac{1}{2} \log |\Sigma_k| \right. \\&\quad \left. - \frac{1}{2} \boldsymbol{\mu}_k^T \Sigma_k^{-1} \boldsymbol{\mu}_k \right) \\&= \arg \max_k \delta_k(\mathbf{x})\end{aligned}$$

- The decision boundaries of QDA are thus quadratic

QDA - illustration

LDA - black dotted, QDA - green solid, Bayesian - purple dashed



QDA and estimation

- The unknown parameters of the LDA model are $\{\pi_k, \mu_k, \Sigma_k, k = 1, \dots, K\}$
- The method of moments estimators of the π_k s and the μ_k s are the same as in LDA
- The estimators of the covariance matrices are given by

$$\hat{\Sigma}_k = \frac{1}{N_k - 1} \sum_{i: y_i = k} (\mathbf{x}_i - \hat{\mu}_k)(\mathbf{x}_i - \hat{\mu}_k)^T$$

- An alternative is to estimate the parameters with the maximum likelihood estimator

Example: Credit default data

See R code

Computations in LDA or QDA

- A way to avoid costly matrix inversions and computations of determinants is to use the eigen value decompositions $\hat{\Sigma}_k = \mathbf{U}_k \mathbf{D}_k \mathbf{U}_k^T$ of the $\hat{\Sigma}_k$ s, where $\mathbf{U}_k$ is orthonormal and $\mathbf{D}_k$ is diagonal
- This gives $\hat{\Sigma}_k^{-1} = \mathbf{U}_k \mathbf{D}_k^{-1} \mathbf{U}_k^T$, and we can simply compute

$$\begin{aligned} (\mathbf{x} - \hat{\mu}_k)^T \hat{\Sigma}_k^{-1} (\mathbf{x} - \hat{\mu}_k) &= (\mathbf{U}_k^T (\mathbf{x} - \hat{\mu}_k))^T \mathbf{D}_k^{-1} (\mathbf{U}_k^T (\mathbf{x} - \hat{\mu}_k)) \\ \log |\hat{\Sigma}_k| &= \sum_l \log d_{kl}, \end{aligned}$$

where d_{kl} are the eigen values, which are the diagonal elements of $\mathbf{D}_k$

- In LDA, we can sphere the data with the common covariance matrix prior to analysis by letting

$$\mathbf{X}^* = \mathbf{D}^{-1/2} \mathbf{U}^T \mathbf{X}$$

- The common covariance matrix of the $\mathbf{X}^*$ s is then the identity matrix

Regularised discriminant analysis

- Allowing to choose between the common (LDA) and class specific covariance matrices (QDA) creates a form of regularisation:

$$\hat{\Sigma}_k(\alpha) = \alpha \hat{\Sigma}_k + (1 - \alpha) \hat{\Sigma}, \alpha \in [0, 1]$$

- It is also possible to choose between the covariance structure and a shared common variance with no dependence in a similar way:

$$\hat{\Sigma}(\gamma) = \gamma \hat{\Sigma} + (1 - \gamma) \hat{\sigma}^2 \mathbf{I}, \gamma \in [0, 1]$$

- Mixing the three variants of covariance is also an option

Logistic regression vs LDA

- Let $\mathbf{x}_i = (1, \tilde{\mathbf{x}}_i)$ and $\beta = (\beta_0, \tilde{\beta})$
- Then, the log-odds ratio for the logistic regression model is given by

$$\log \frac{\Pr(Y = 1|\mathbf{x})}{\Pr(Y = 0|\mathbf{x})} = \log(e^{\beta_0 + \tilde{\mathbf{x}}^T \tilde{\beta}}) = \beta_0 + \tilde{\mathbf{x}}^T \tilde{\beta}$$

- For the LDA model for two classes ($K = 2$), we have

$$\begin{aligned} \log \frac{\Pr(Y = 1|\mathbf{x})}{\Pr(Y = 0|\mathbf{x})} &= \log \frac{\pi_1 p_1(\tilde{\mathbf{x}}) / (\pi_1 p_1(\tilde{\mathbf{x}}) + (1 - \pi_1) p_1(\tilde{\mathbf{x}}))}{(1 - \pi_1) p_0(\tilde{\mathbf{x}}) / (\pi_1 p_1(\tilde{\mathbf{x}}) + (1 - \pi_1) p_1(\tilde{\mathbf{x}}))} \\ &= \log(\pi_1) - \log(1 - \pi_1) + \log(p_1(\tilde{\mathbf{x}})) - \log(p_0(\tilde{\mathbf{x}})) \\ &= \log(\pi_1) - \log(1 - \pi_1) + \frac{1}{2} \mu_0^T \Sigma^{-1} \mu_0 - \frac{1}{2} \mu_1^T \Sigma^{-1} \mu_1 \\ &\quad + \tilde{\mathbf{x}}^T (\Sigma^{-1} \mu_1 - \Sigma^{-1} \mu_0) \\ &= \alpha_0 + \tilde{\mathbf{x}}^T \alpha \end{aligned}$$

Logistic regression vs LDA

- Hence, the two models have log odds ratios on the same form
- However, the logistic regression model is fitted based on the conditional model of $Y|\mathbf{X} = \mathbf{x}$, whereas the LDA model is fitted based on the joint model of $(Y, \mathbf{X})$
- Thus, the LDA model uses more information in the estimation
- This results in higher accuracy if the model is correctly specified (though typically not very much higher)
- On the other hand, the logistic regression model is more robust towards model misspecification

Predictive power in classification: confusion matrix, ROC curve and AUC

- From a fitted classification model with binary response ($Y \in \{0, 1\}$), we may obtain the prediction rule:

$$\hat{y}_i = \begin{cases} 1, & \hat{p}_i > \tau \\ 0, & \hat{p}_i \leq \tau \end{cases}$$

with $\hat{p}_i = \hat{\text{Pr}}(Y_i = 1 | \mathbf{x}_i)$

- For a particular cut-off τ , we have

$$\text{true positive count (tp)} = \sum_{i=1}^N I(\hat{y}_i = 1, y_i = 1), \text{ false positive count (fp)} = \sum_{i=1}^N I(\hat{y}_i = 1, y_i = 0)$$

$$\text{false negative count (fn)} = \sum_{i=1}^N I(\hat{y}_i = 0, y_i = 1), \text{ true negative count (tn)} = \sum_{i=1}^N I(\hat{y}_i = 0, y_i = 0)$$

$$\text{true positive rate (tpr)} = \frac{tp}{tp + fn} = \frac{\sum_{i=1}^N I(\hat{y}_i = 1, y_i = 1)}{\sum_{i=1}^N I(y_i = 1)} = \hat{\text{Pr}}(\hat{Y} = 1 | Y = 1)$$

$$\text{false positive rate (fpr)} = \frac{fp}{fp + tn} = \frac{\sum_{i=1}^N I(\hat{y}_i = 1, y_i = 0)}{\sum_{i=1}^N I(y_i = 0)} = \hat{\text{Pr}}(\hat{Y} = 1 | Y = 0)$$

Predictive power in classification: confusion matrix, ROC curve and AUC

- The **confusion matrix** is then given by:

True class	Predicted class	
	0	1
0	tn	fp
1	fn	tp

- A **receiver operating characteristic (ROC) curve** is a plot of tpr as a function of fpr for all possible cut-offs $\tau \in [0, 1]$
- The closer to the upper left corner the ROC curve is, the better the prediction performance

Predictive power in classification: confusion matrix, ROC curve and AUC

- ROC curves from different models might cross each other, making it difficult to assess which model is best
- Further, it is often convenient to summarise the prediction performance with a scalar, for instance in a cross-validation scheme
- One such measure is the area under the ROC curve, denoted AUC
- The AUC is a number on the interval $[0, 1]$, and is 1 for a perfect classifier and 0.5 for random guessing

Example: Credit default data

See R code