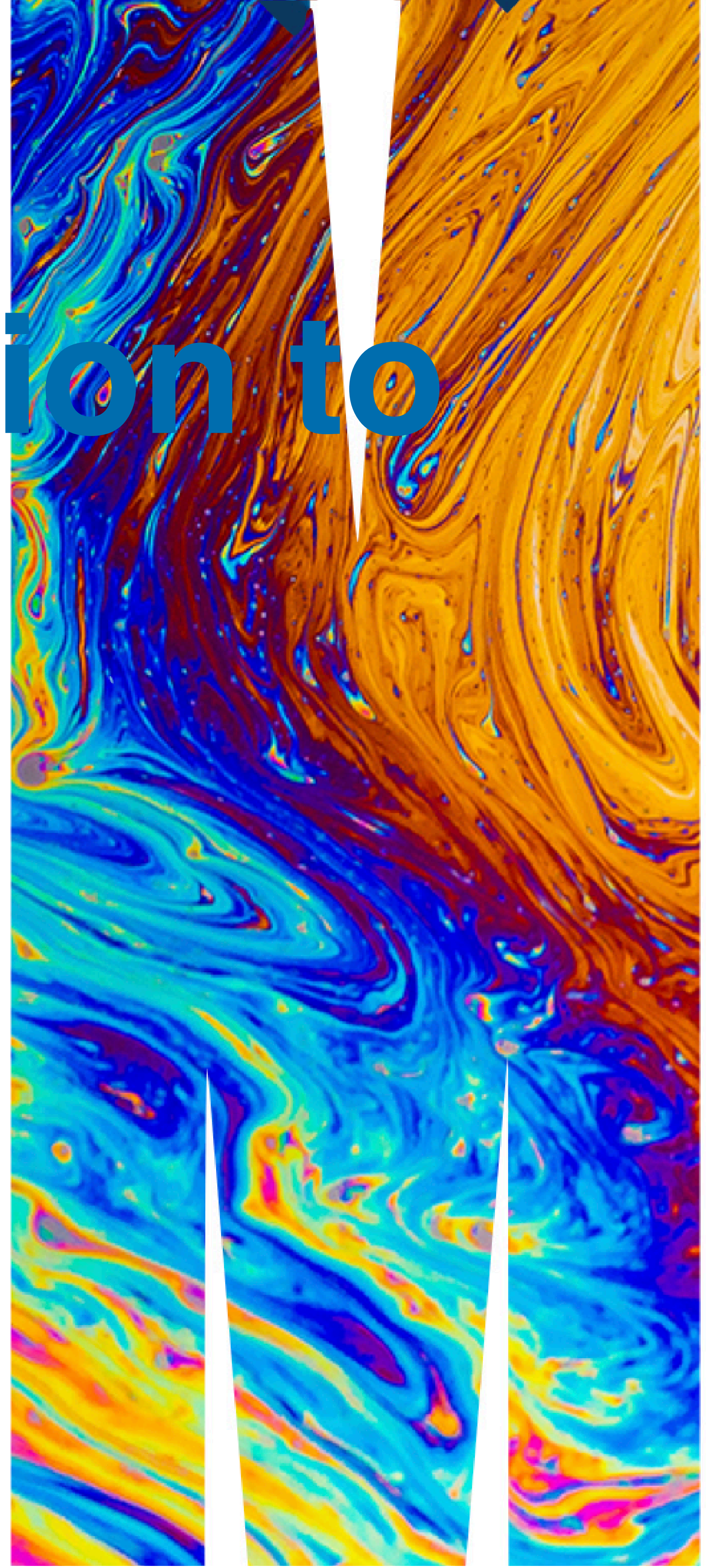
ETC3250/5250 Introduction to Machine Learning

Week 4: Logistic regression and discriminant analysis

Professor Di Cook

etc3250.clayton-x@monash.edu

Department of Econometrics and Business Statistics



Overview

We will cover:

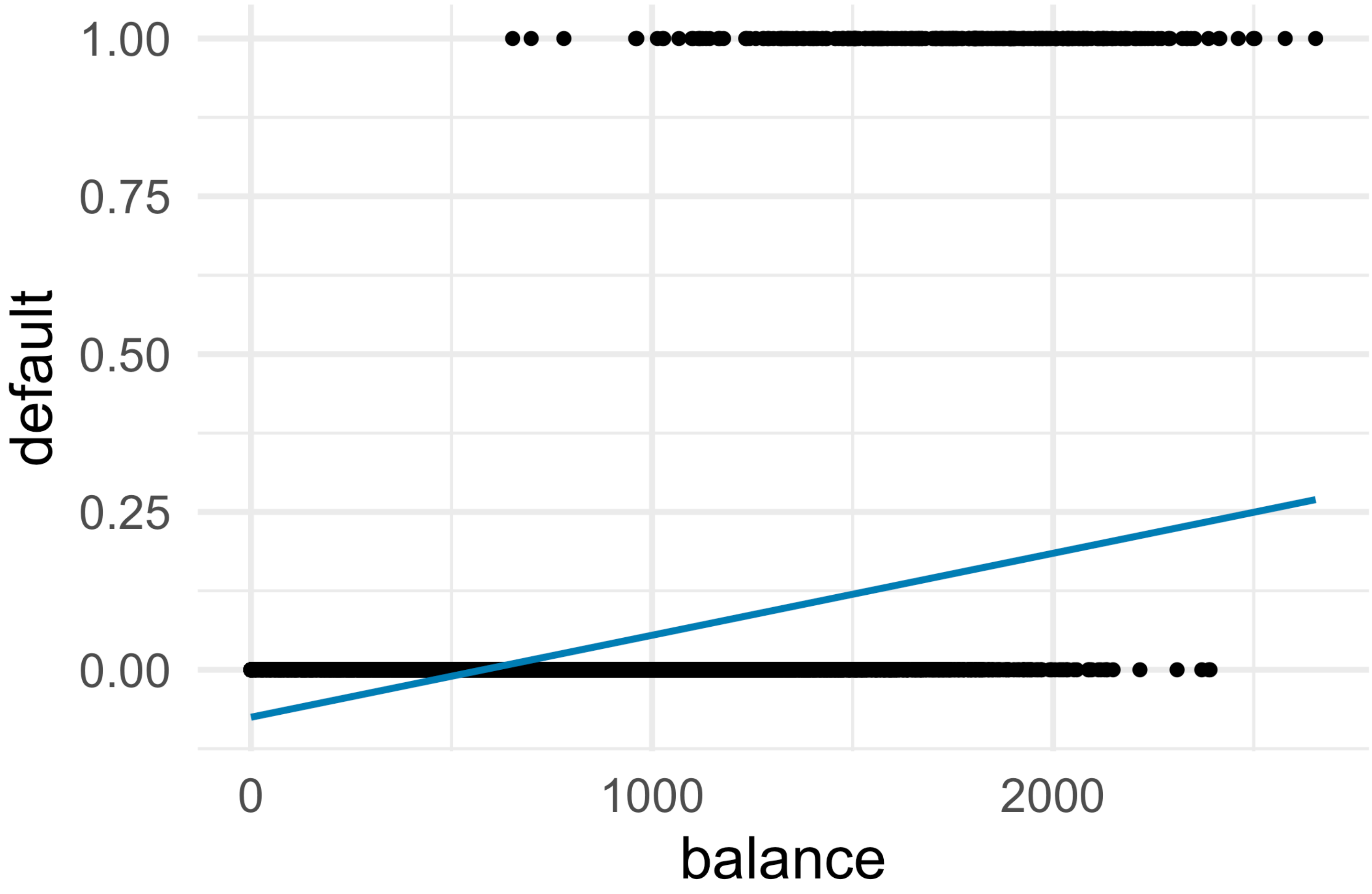
- Fitting a categorical response using logistic curves
- Multivariate summary statistics
- Linear discriminant analysis, assuming samples are elliptically shaped and equal in size
- Quadratic discriminant analysis, assuming samples are elliptically shaped and different in size
- Discriminant space: making a low-dimensional visual summary

Logistic regression

When linear regression is not appropriate

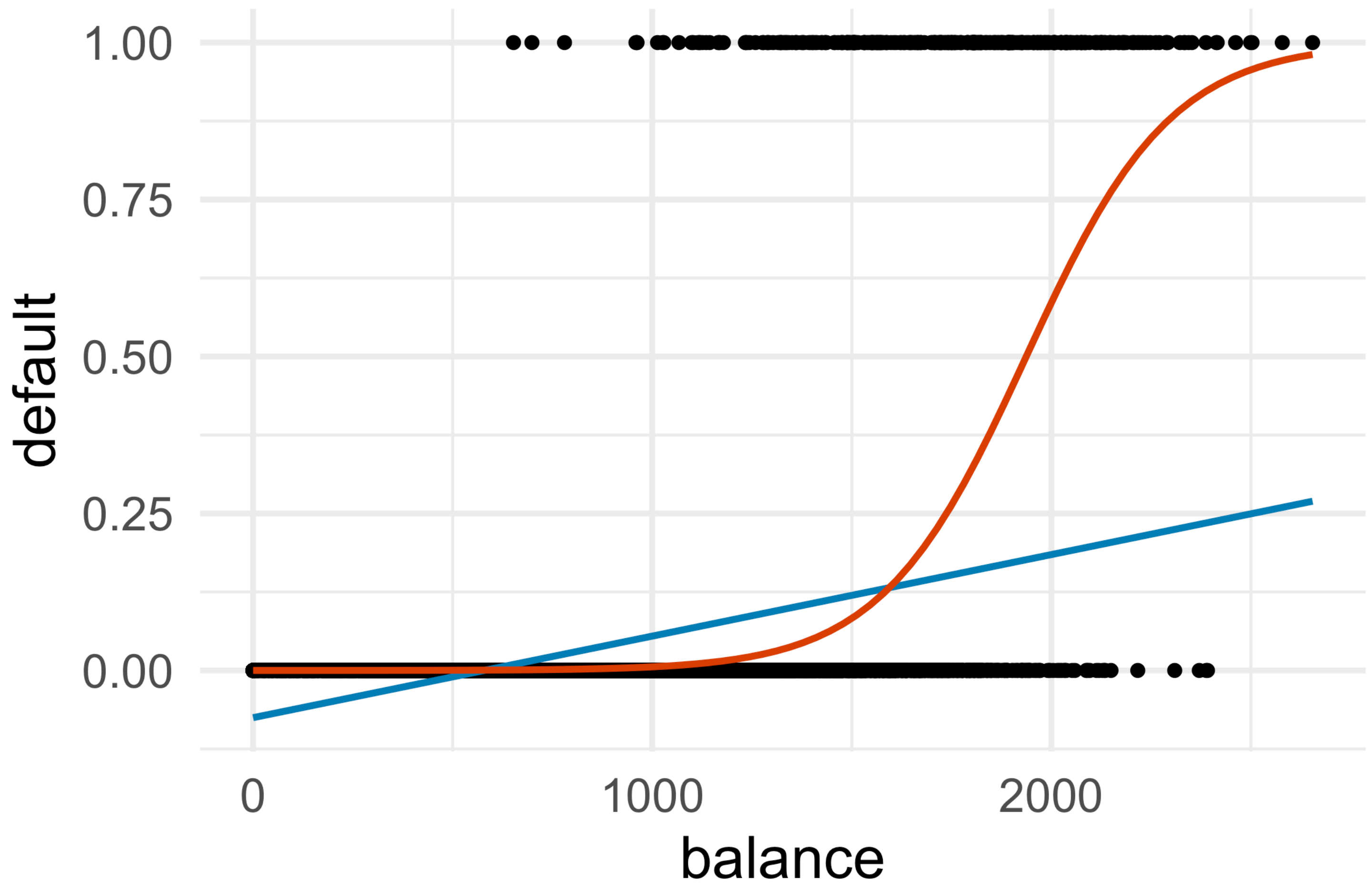
Consider the following data `Default` in the ISLR R package (textbook) which looks at the default status based on credit balance.

```r
library(ISLR)
data(Default)
simcredit <- Default |>
  mutate(default_bin = ifelse(default=="Yes",
```



Why is a linear model less than ideal for this data?

Modelling binary responses



Orange line (logistic model fit) is similar to computing a running average of the 0s/1s. It’s much better than the linear fit, because it remains between 0 and 1, and can be interpreted as proportion of 1s.

What is a logistic function?

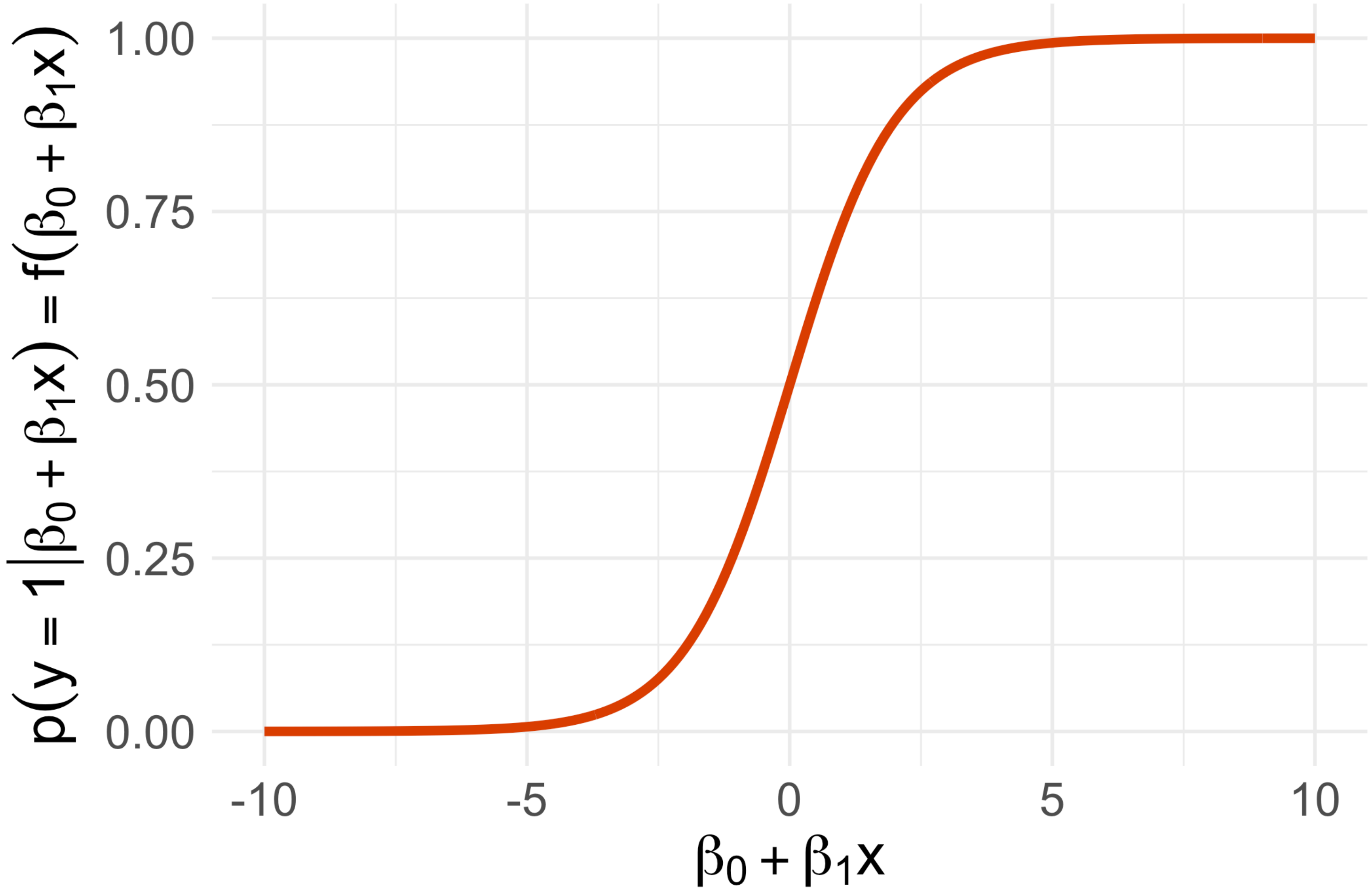
The logistic function

Instead of predicting the outcome directly, we instead predict the probability of being class 1, given the (linear combination of) predictors, using the logistic function.

$$p(y = 1|\beta_0 + \beta_1 x) = f(x)$$

where

$$f(x) = \frac{e^{\beta_0+\beta_1 x}}{1 + e^{\beta_0+\beta_1 x}}$$



Logistic function

Transform the function:

$$y = \frac{e^{\beta_0+\beta_1 x}}{1 + e^{\beta_0+\beta_1 x}}$$

$$\longrightarrow y = \frac{1}{1/e^{\beta_0+\beta_1 x}+1}$$

$$\longrightarrow 1/y = 1/e^{\beta_0+\beta_1 x} + 1$$

$$\longrightarrow 1/y - 1 = 1/e^{\beta_0+\beta_1 x}$$

$$\longrightarrow \frac{1}{1/y-1} = e^{\beta_0+\beta_1 x}$$

$$\longrightarrow \frac{y}{1-y} = e^{\beta_0+\beta_1 x}$$

$$\longrightarrow \log_e \frac{y}{1-y} = \beta_0 + \beta_1 x$$

Transforming the response $\log_e \frac{y}{1-y}$ makes it possible to use a linear model fit.

> The left-hand side, $\log_e \frac{y}{1-y}$, is known as the log-odds ratio or logit.

The logistic regression model

The fitted model, where $P(Y=0|X) = 1 - P(Y=1|X)$, is then written as:

$$\log_e \frac{P(Y=1|X)}{1-P(Y=1|X)} = \beta_0 + \beta_1 X$$

When there are more than two categories:

- the formula can be extended, using dummy variables.
- follows from the above, extended to provide probabilities for each level/category, and the last category is 1-sum of the probabilities of other categories.
- the sum of all probabilities has to be 1.

Connection to generalised linear models

- To model **binary data**, we need to link our **predictors** to our response using a *link function*. Another way to think about it is that we will transform Y, to convert it to a proportion, and then build the linear model on the transformed response.
- There are many different types of link functions we could use, but for a binary response we typically use the logistic link function.

Interpretation

- **Linear regression**
  - β_1 gives the average change in Y associated with a one-unit increase in X
- **Logistic regression**
  - Because the model is not linear in X, β_1 does not correspond to the change in response associated with a one-unit increase in X.
  - However, increasing X by one unit changes the log odds by β_1, or equivalently it multiplies the odds by e^{β_1}

Maximum Likelihood Estimation

Given the logistic $p(x_i) = \frac{1}{e^{-(\beta_0+\beta_1 x_i)}+1}$ choose parameters β_0, β_1 to maximize the likelihood:

$$l_n(\beta_0, \beta_1) = \prod_{i=1}^{n} p(x_i)^{y_i}(1 - p(x_i))^{1-y_i}.$$

It is more convenient to maximize the *log-likelihood*:

$$\begin{aligned}\log l_n(\beta_0, \beta_1) &= \sum_{i=1}^{n} \big(y_i \log p(x_i) + (1 - y_i) \log(1 - p(x_i))\big) \\ &= \sum_{i=1}^{n} \big(y_i(\beta_0 + \beta_1 x_i) - \log{(1 + e^{\beta_0+\beta_1 x_i})}\big)\end{aligned}$$

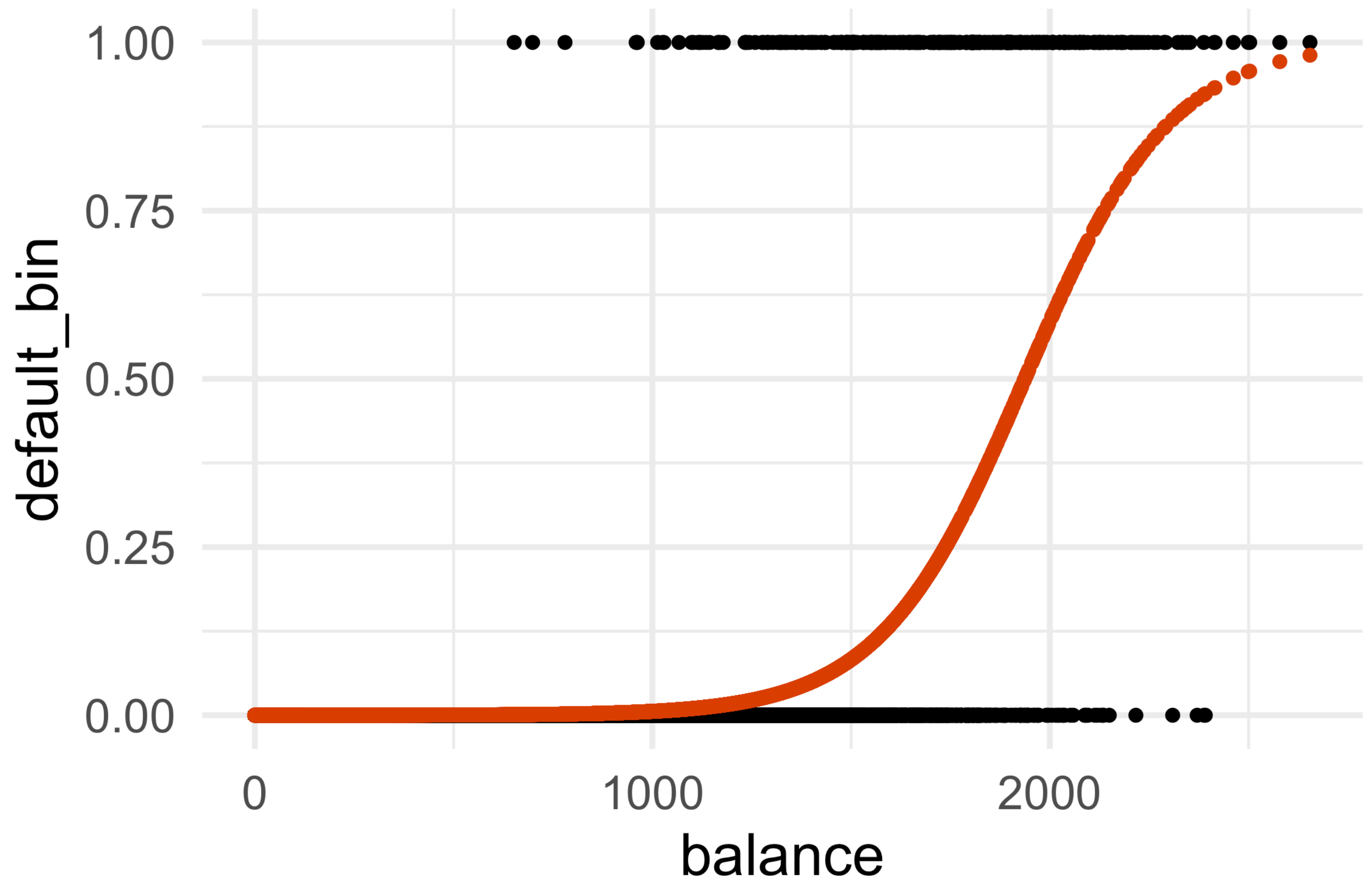
Making predictions

With estimates from the model fit, $\hat{\beta}_0, \hat{\beta}_1$, we can predict the **probability of belonging to class 1** using:

$$p(y = 1|\hat{\beta}_0 + \hat{\beta}_1 x) = \frac{e^{\hat{\beta}_0+\hat{\beta}_1 x}}{1 + e^{\hat{\beta}_0+\hat{\beta}_1 x}}$$

Round to 0 or 1 for class prediction.

```r
fit <- glm(default~balance,
           data=simcredit, family="binomial")
simcredit_fit <- augment(fit, simcredit,
                         type.predict="respons
```



Orange points are fitted values, $\hat{y}_i$. Black points are observed response, y_i (either 0 or 1).

Fitting credit data in R

We can use the `glm` function in R to fit a logistic regression model. The `glm` function can support many response types, so we specify `family="binomial"` to let R know that our response is *binary*.

```r
fit <- glm(default~balance,
           data=simcredit, family="binomial")
simcredit_fit <- augment(fit, simcredit,
                         type.predict="respons
```

Same calculation but written in `tidymodels` style

```r
logistic_mod <- logistic_reg() |>
  set_engine("glm") |>
  set_mode("classification") |>
  translate()

logistic_fit <-
  logistic_mod |>
  fit(default ~ balance,
      data = simcredit)
```

Examine the fit

```r
tidy(logistic_fit)
```

```
# A tibble: 2 × 5
  term        estimate std.error statistic   p.value
  <chr>          <dbl>     <dbl>     <dbl>     <dbl>
1 (Intercept) -10.7     0.361        -29.5 3.62e-191
2 balance       0.00550 0.000220      25.0 1.98e-137
```

```r
glance(logistic_fit)
```

```
# A tibble: 1 × 8
  null.deviance df.null logLik   AIC   BIC deviance
          <dbl>   <int>  <dbl> <dbl> <dbl>    <dbl>
1         2921.    9999  -798. 1600. 1615.    1596.
# ℹ 2 more variables: df.residual <int>, nobs <int>
```

Parameter estimates

$\hat{\beta}_0$ = -10.65

$\hat{\beta}_1$ = 0.01

Can you write out the model?

Model fit summary

Null model deviance 2920.6 (error for model with no predictors)

Model deviance 1596.5 (error from fitted model)

How good is the model?

Check the model performance

```r
simcredit_fit <- augment(logistic_fit, simcred
simcredit_fit |>
  count(default, .pred_class) |>
  group_by(default) |>
  mutate(Accuracy = n[.pred_class==default]/su
  pivot_wider(names_from = ".pred_class", valu
  select(default, No, Yes, Accuracy)
```

```
# A tibble: 2 × 4
# Groups:   default [2]
  default    No   Yes Accuracy
  <fct>   <int> <int>    <dbl>
1 No       9625    42    0.996
2 Yes       233   100    0.300
```

Compute the balanced accuracy.

Unbalanced data set, with very different performance on each class.

How good is this model?

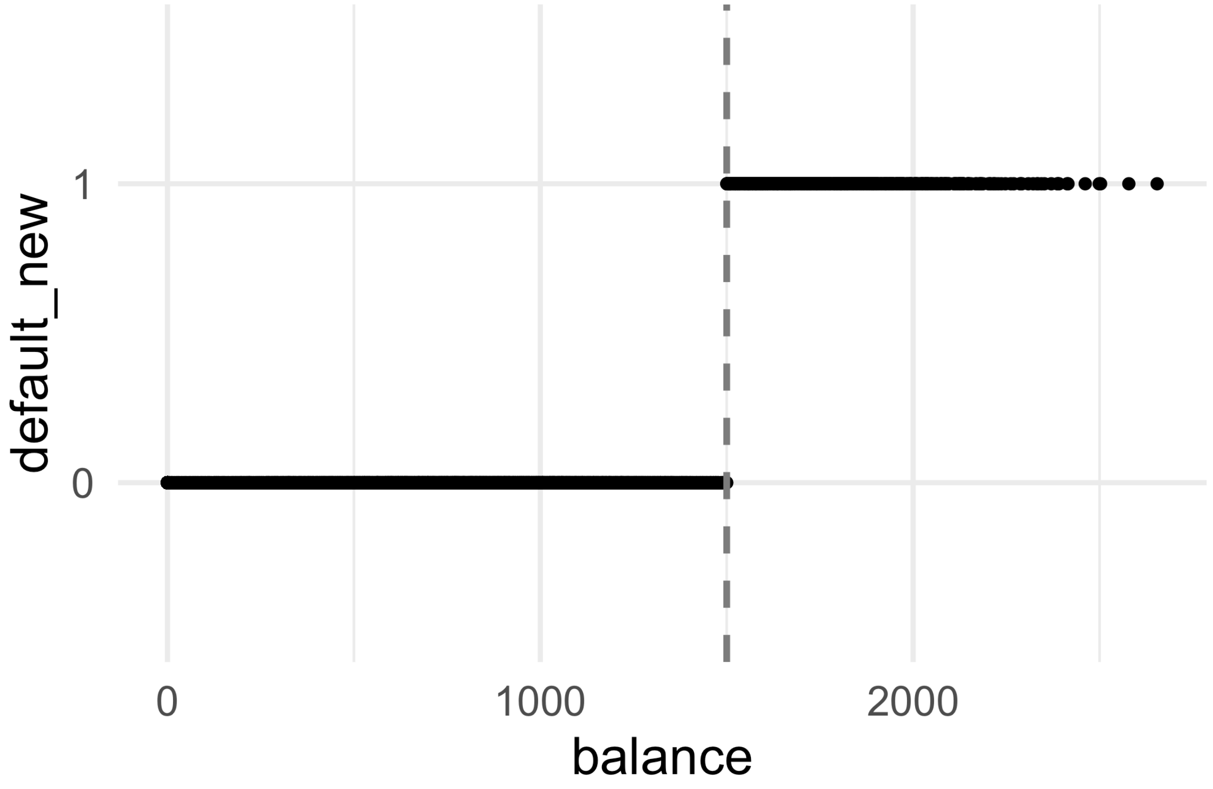
- Explains about half of the variation in the response, which would normally be reasonable.
- Gets most of the smaller but important class wrong.
- Not a very useful model.

A warning for using GLMs!

> Logistic regression model fitting fails when the data is *perfectly* separated.

MLE fit will try and fit a step-wise function to this graph, pushing coefficients sizes towards infinity and produce large standard errors.

Pay attention to warnings!



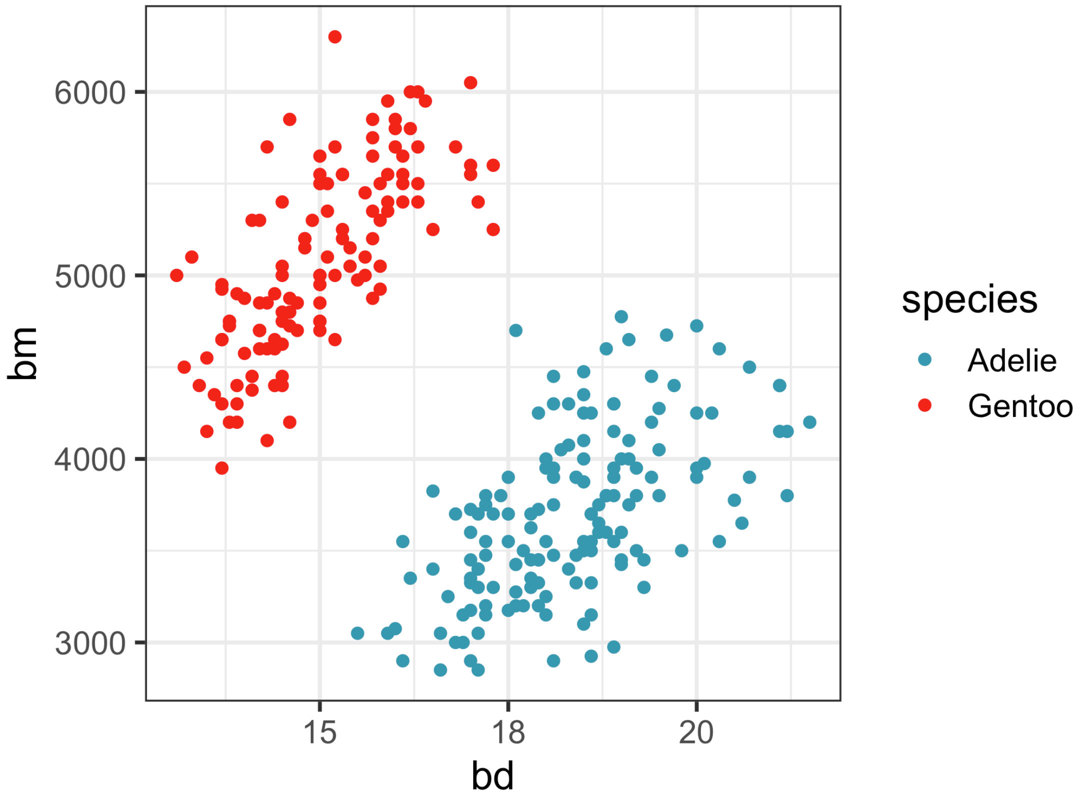
```r
logistic_fit <-
  logistic_mod |>
  fit(default_new ~ balance,
      data = simcredit)
```

```
Warning: glm.fit: algorithm did not converge

Warning: glm.fit: fitted probabilities numerically 0 or 1
occurred
```

Discriminant Analysis

Linear Discriminant Analysis

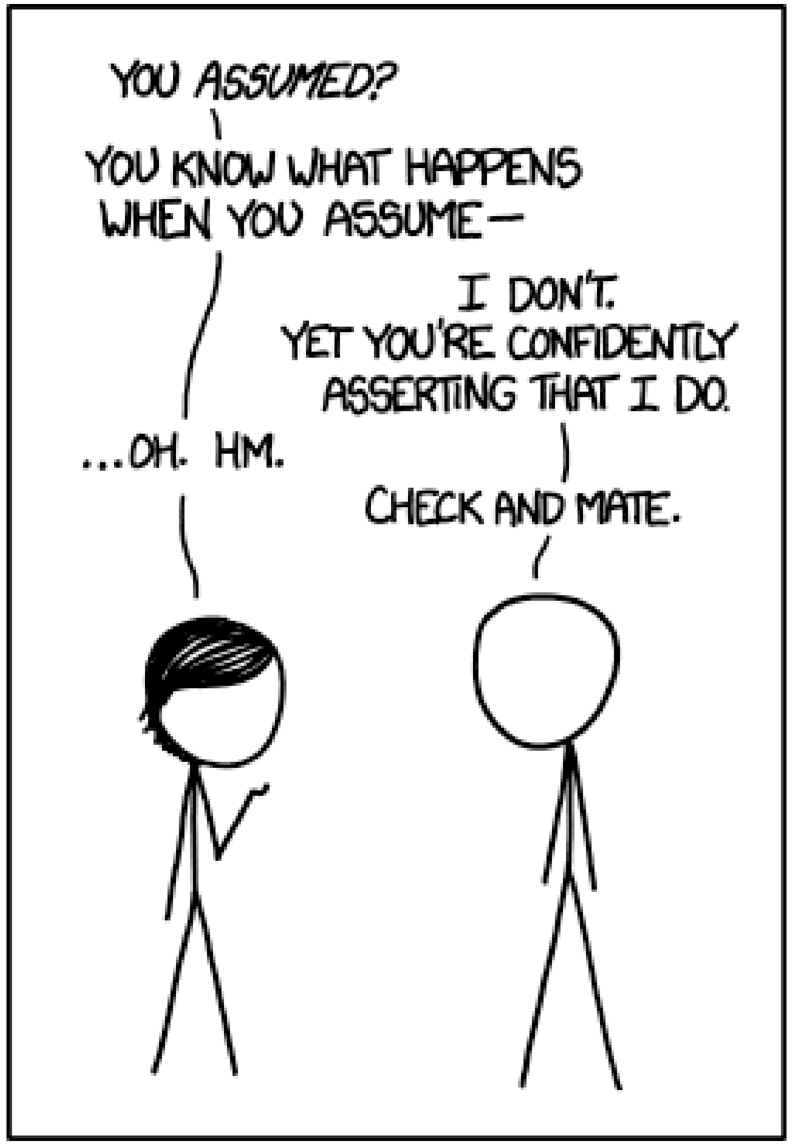


Where would you draw a line to create a boundary separating Adelie and Gentoo penguins?

- Where are the sample means?
- What is the shape of the sample variance-covariance?

Linear discriminant analysis assumes the distribution of the predictors is a multivariate normal, with the same variance-covariance matrix, separately for each class.

Assumptions underlie LDA


YOU ASSUMED?
YOU KNOW WHAT HAPPENS WHEN YOU ASSUME—
I DON'T. YET YOU'RE CONFIDENTLY ASSERTING THAT I DO.
...OH. HM.
CHECK AND MATE.

Source: https://xkcd.com

- All samples come from normal populations
- with the same population variance-covariance matrix

LDA with $p = 1$ predictors 1/4

If $K = 2$ (two classes labelled A and B) and each group has the *same prior probability*, the LDA rule is to assign the new observation x_0 to class A if

$$x_0 > \frac{\bar{x}_A + \bar{x}_B}{2}$$

- It's a really intuitive rule, eh?
- It also matters which of the two classes is considered to be A!!!
- So maybe easier to think about as "**assign the new observation to the group with the closest mean**".
- How does this rule arise from the assumptions?

Bayes Theorem 2/4

Let $f_k(x)$ be the density function for predictor x for class k. If f is large, the probability that x belongs to class k is large, or if f is small it is unlikely that x belongs to class k.

According to Bayes theorem (for K classes) the probability that x belongs to class k is:

$$P(Y = k|X = x) = p_k(x) = \frac{\pi_k f_k(x)}{\sum_{i=1}^{K} \pi_k f_k(x)}$$

where π_k is the prior probability that an observation comes from class k.

LDA with $p = 1$ predictors 3/4

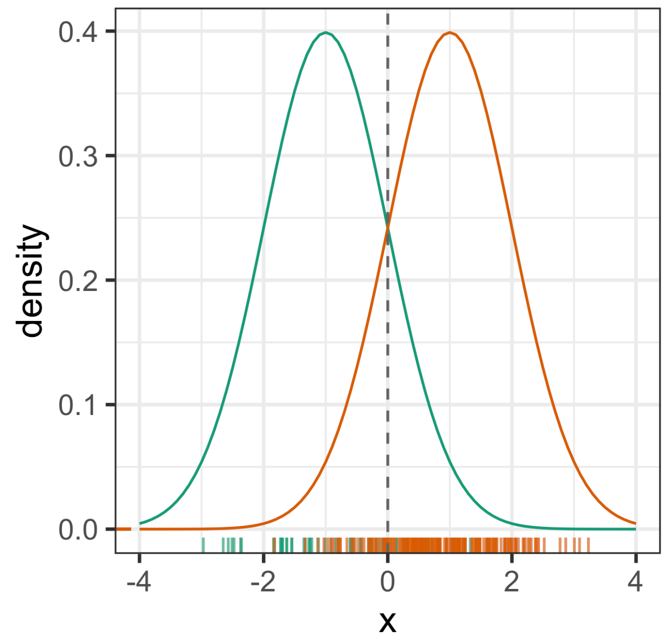
The density function $f_k(x)$ of a univariate normal (Gaussian) is

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp~\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right)$$

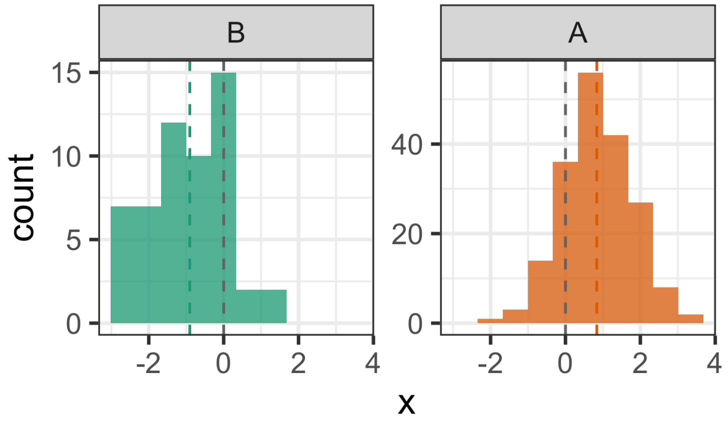
where μ_k and σ_k^2 are the mean and variance parameters for the kth class. We also assume that $\sigma_A^2 = \sigma_B^2 = \dots = \sigma_K^2$; then the conditional probabilities are

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp~\left(-\frac{1}{2\sigma^2}(x-\mu_k)^2\right)}{\sum_{l=1}^K \pi_l \frac{1}{\sqrt{2\pi}\sigma} \exp~\left(-\frac{1}{2\sigma^2}(x-\mu_l)^2\right)}$$

Population



Data



LDA with $p = 1$ predictors 4/4

A simplification of $p_k(x_0)$ yields the discriminant functions, $\delta_k(x_0)$:

$$\delta_k(x_0) = x_0\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + log(\pi_k)$$

from which the LDA rule will assign x_0 to the class k with the largest value.

Let $K = 2$, then the rule reduces to assign x_0 to class A if

$$\frac{\pi_A \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2\sigma^2}(x-\mu_A)^2\right)}{\sum_{l=1}^{2}\pi_l \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2\sigma^2}(x-\mu_l)^2\right)} > \frac{\pi_B \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2\sigma^2}(x-\mu_B)^2\right)}{\sum_{l=1}^{2}\pi_l \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2\sigma^2}(x-\mu_l)^2\right)}$$

$$\longrightarrow \pi_A \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2\sigma^2}(x_0-\mu_A)^2\right) > \pi_B \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2\sigma^2}(x_0-\mu_B)^2\right)$$

$$\longrightarrow \pi_A\exp\left(-\frac{1}{2\sigma^2}(x_0-\mu_A)^2\right) > \pi_B\exp\left(-\frac{1}{2\sigma^2}(x_0-\mu_B)^2\right)$$

$$\longrightarrow \log\pi_A - \frac{1}{2\sigma^2}(x_0-\mu_A)^2 > \log\pi_B - \frac{1}{2\sigma^2}(x_0-\mu_B)^2$$

$$\longrightarrow \log\pi_A - \frac{1}{2\sigma^2}(x_0^2 - 2x_0\mu_A + \mu_A^2) > \log\pi_B - \frac{1}{2\sigma^2}(x_0^2 - 2x_0\mu_B + \mu_B^2)$$

$$\longrightarrow \log\pi_A - \frac{1}{2\sigma^2}(-2x_0\mu_A + \mu_A^2) > \log\pi_B - \frac{1}{2\sigma^2}(-2x_0\mu_B + \mu_B^2)$$

$$\longrightarrow \log\pi_A + \frac{x_0\mu_A}{\sigma^2} - \frac{\mu_A^2}{\sigma^2} > \log\pi_B + \frac{x_0\mu_B}{\sigma^2} - \frac{\mu_B^2}{\sigma^2}$$

$$\longrightarrow \underbrace{x_0\frac{\mu_A}{\sigma^2} - \frac{\mu_A^2}{\sigma^2} + \log\pi_A}_{\text{Discriminant function for class A}} > \underbrace{x_0\frac{\mu_B}{\sigma^2} - \frac{\mu_B^2}{\sigma^2} + \log\pi_B}_{\text{Discriminant function for class B}}$$

Multivariate LDA, $p > 1$

A p-dimensional random variable X has a multivariate Gaussian distribution with mean μ and variance-covariance Σ, we write $X \sim N(\mu, \Sigma)$.

The multivariate normal density function is:

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

with x, μ are p-dimensional vectors, Σ is a $p \times p$ variance-covariance matrix.

Multivariate LDA, $K = 2$

The discriminant functions are:

$$\delta_k(x) = x^\top \Sigma^{-1} \mu_k - \frac{1}{2}\mu_k^\top \Sigma^{-1} \mu_k + \log(\pi_k)$$

and Bayes classifier is assign a new observation x_0 to the class with the highest $\delta_k(x_0)$.

When $K = 2$ and $\pi_A = \pi_B$ this reduces to

Assign observation x_0 to class A if

$$x_0^\top \underbrace{\Sigma^{-1}(\mu_A - \mu_B)}_{\textit{dimension reduction}} > \frac{1}{2}(\mu_A + \mu_B)^\top \underbrace{\Sigma^{-1}(\mu_A - \mu_B)}_{\textit{dimension reduction}}$$

NOTE: Class A and B need to be mapped to the classes in the your data. The class "to the right" on the reduced dimension will correspond to class A in this equation.

Computation

Use sample mean $\bar{x}_k$ to estimate μ_k, and

to estimate Σ use the pooled variance-covariance:

$$S = \frac{n_1 S_1 + n_2 S_2 + \cdots + n_k S_k}{n_1 + n_2 + \cdots + n_k}$$

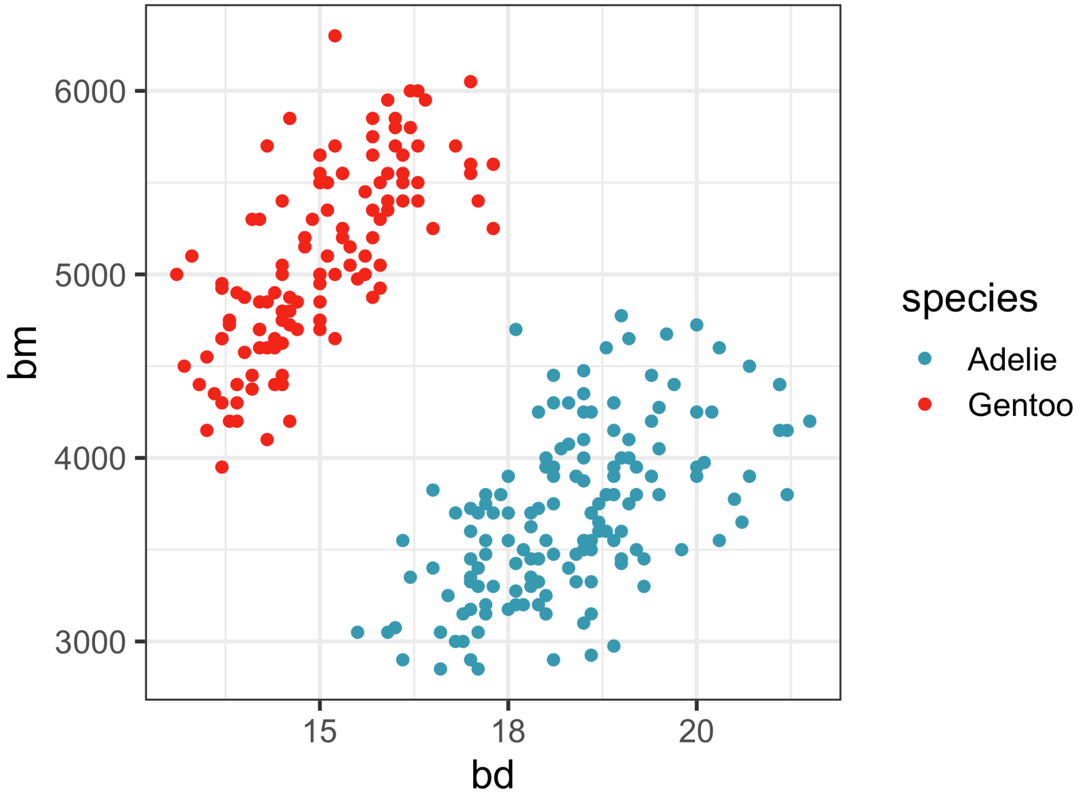
Example: penguins 1/3

Summary statistics

```
# A tibble: 2 × 3
  species    bm    bd
  <fct>   <dbl> <dbl>
1 Adelie  3701.  18.3
2 Gentoo  5076.  15.0
```

```
       bm    bd
bm 210283 321.4
bd    321   1.5
```

```
       bm     bd
bm 254133 355.69
bd    356   0.96
```



```r
library(discrim)
lda_spec <- discrim_linear() |>
  set_mode("classification") |>
  set_engine("MASS", prior = c(0.5, 0.5))
lda_fit <- lda_spec |> 
  fit(species ~ bm + bd, data = p_sub)

lda_fit
```

```
parsnip model object

Call:
lda(species ~ bm + bd, data = data, prior = ~c(0.5, 0.5))

Prior probabilities of groups:
Adelie Gentoo 
   0.5    0.5 

Group means:
         bm bd
Adelie 3701 18
Gentoo 5076 15

Coefficients of linear discriminants:
```

Recommendation: standardise the variables before fitting model, even though it is not necessary for LDA.

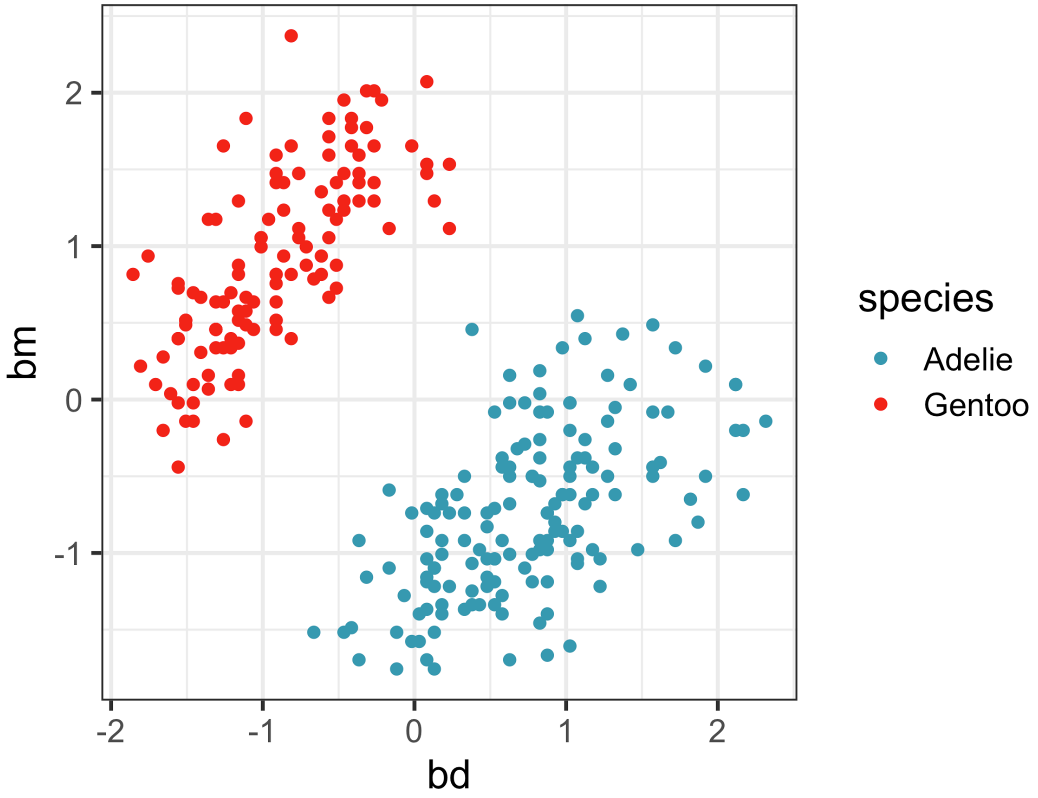
Example: penguins 2/3

Summary statistics

```
# A tibble: 2 × 3
  species     bm     bd
  <fct>    <dbl>  <dbl>
1 Adelie  -0.739  0.750
2 Gentoo   0.907 -0.921
```

```
     bm   bd
bm 0.30 0.19
bd 0.19 0.37
```

```
     bm   bd
bm 0.36 0.21
bd 0.21 0.24
```



```r
library(discrim)
lda_spec <- discrim_linear() |>
  set_mode("classification") |>
  set_engine("MASS", prior = c(0.5, 0.5))
lda_fit <- lda_spec |>
  fit(species ~ bm + bd, data = p_sub)

lda_fit
```

```
parsnip model object

Call:
lda(species ~ bm + bd, data = data, prior = ~c(0.5, 0.5))

Prior probabilities of groups:
Adelie Gentoo
   0.5    0.5

Group means:
          bm    bd
Adelie -0.74  0.75
Gentoo  0.91 -0.92

Coefficients of linear discriminants:
```

Example: penguins 3/3

$$S^{-1}(\bar{x}_A - \bar{x}_B)$$

```r
S1 <- cov(p_sub[p_sub$species == "Adelie",-1])
S2 <- cov(p_sub[p_sub$species == "Gentoo",-1])
Sp <- (S1+S2)/2
Sp
```

```
     bm  bd
bm 0.33 0.2
bd 0.20 0.3
```

```r
Spinv <- solve(Sp)
Spinv
```

```
     bm   bd
bm  5.1 -3.4
bd -3.4  5.6
```

```r
m1 <- as.matrix(lda_fit$fit$means[1,], ncol=1)
m1
```

```
    [,1]
bm -0.74
bd  0.75
```

```r
m2 <- as.matrix(lda_fit$fit$means[2,], ncol=1)
m2
```

```
    [,1]
bm  0.91
bd -0.92
```

```r
Spinv %*% (m1-m2)
```

```
   [,1]
bm  -14
bd   15
```

$$x_0 S^{-1}(\bar{x}_A - \bar{x}_B) > \frac{\bar{x}_A + \bar{x}_B}{2} S^{-1}(\bar{x}_A - \bar{x}_B)$$

```r
(m1 + m2)/2
```

```
     [,1]
bm  0.084
bd -0.085
```

```r
matrix((m1 + m2)/2, ncol=2) %*% Spinv %*% (m1-m2)
```

```
     [,1]
[1,] -2.4
```

If x_0 is -0.68, 0.93, what species is it?

```r
as.matrix(p_sub[1,-1]) %*% Spinv %*% (m1-m2)
```

```
     [,1]
[1,]   23
```

Is Adelie class A or is Gentoo class A?

Check by plugging in the means

```r
t(m1) %*% Spinv %*% (m1-m2)
```

```
     [,1]
[1,]   21
```

```r
predict(lda_fit, p_sub[1,-1])$.pred_class
```

```
[1] Adelie
Levels: Adelie Gentoo
```

Dimension reduction

Dimension reduction via LDA

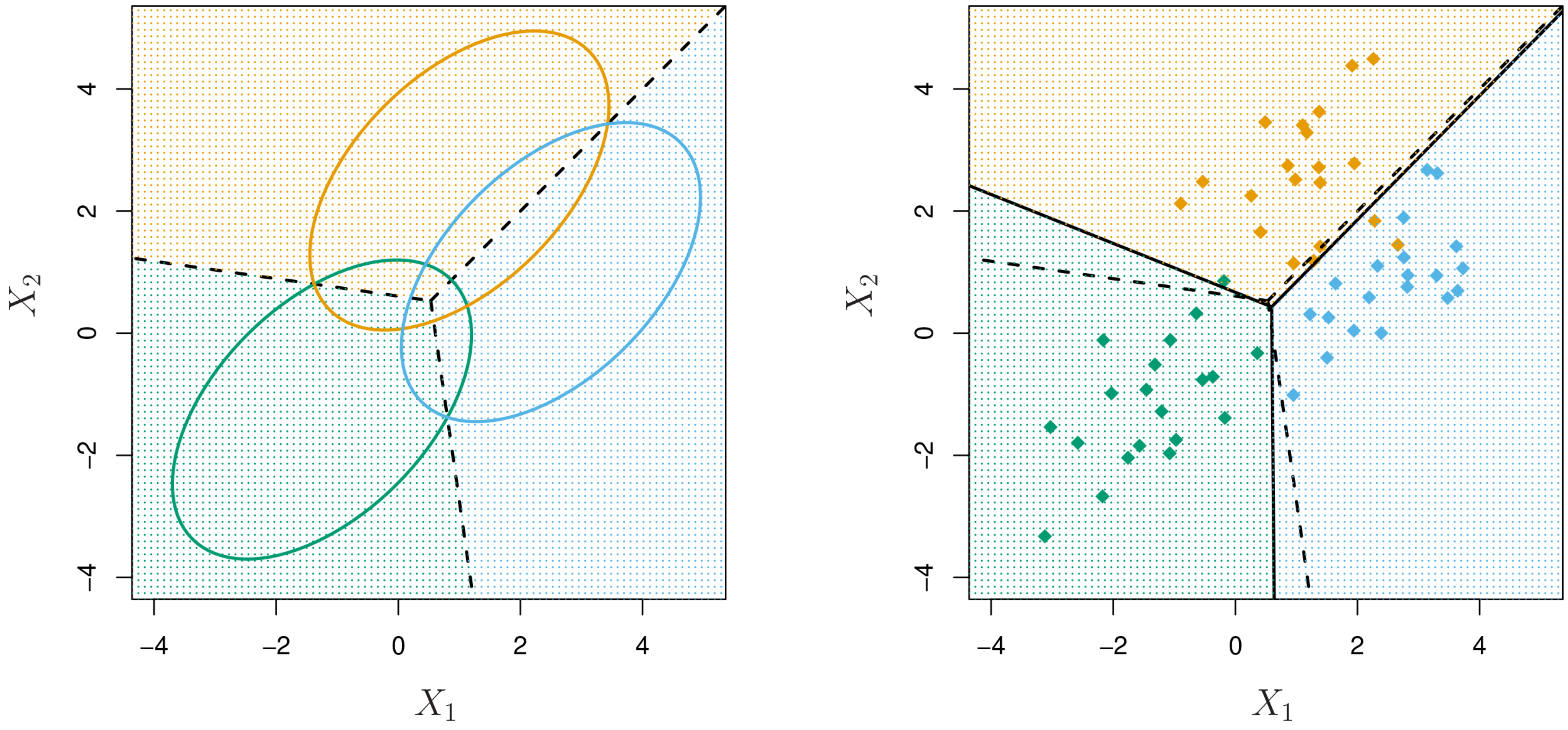
Discriminant space: LDA also provides a low-dimensional projection of the p-dimensional space, where the groups are the most separated. For $K = 2$, this is

$$\Sigma^{-1}(\mu_A - \mu_B)$$

The distance between means relative to the variance-covariance, ie Mahalanobis distance.

Discriminant space

The dashed lines are the Bayes decision boundaries. Ellipses that contain 95% of the probability for each of the three classes are shown. Solid line corresponds to the class boundaries from the LDA model fit to the sample.



(Chapter4/4.6.pdf)

Discriminant space: using sample statistics

> Discriminant space: is the low-dimensional space where the class means are the furthest apart relative to the common variance-covariance.

The discriminant space is provided by the eigenvectors after making an eigen-decomposition of $W^{-1}B$, where

$$B = \frac{1}{K}\sum_{i=1}^{K}(\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^\top \quad \text{and} \quad W = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{n_k}\sum_{i=1}^{n_k}(x_i - \bar{x}_k)(x_i - \bar{x}_k)^\top$$

Note W is the (unweighted) pooled variance-covariance matrix.

Mahalanobis distance

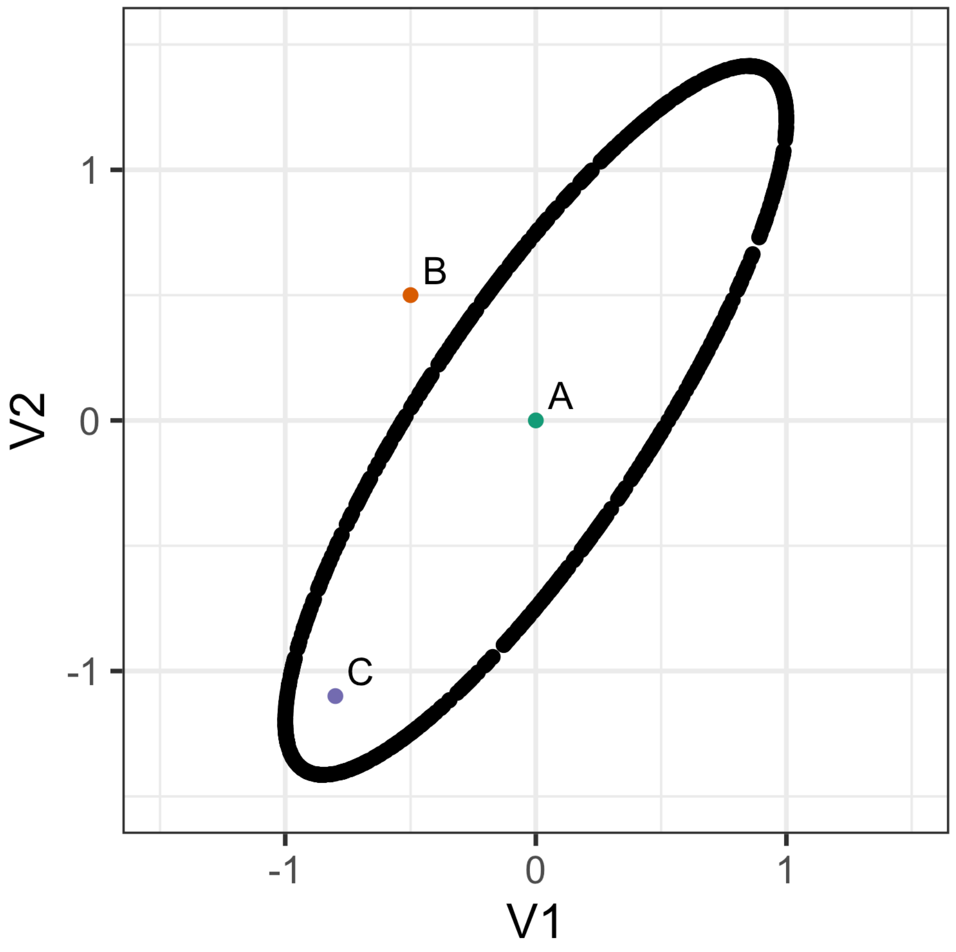
For two p-dimensional vectors, Euclidean distance is

$$d(x, y) = \sqrt{(x-y)^\top (x-y)}$$

and Mahalanobs distance is

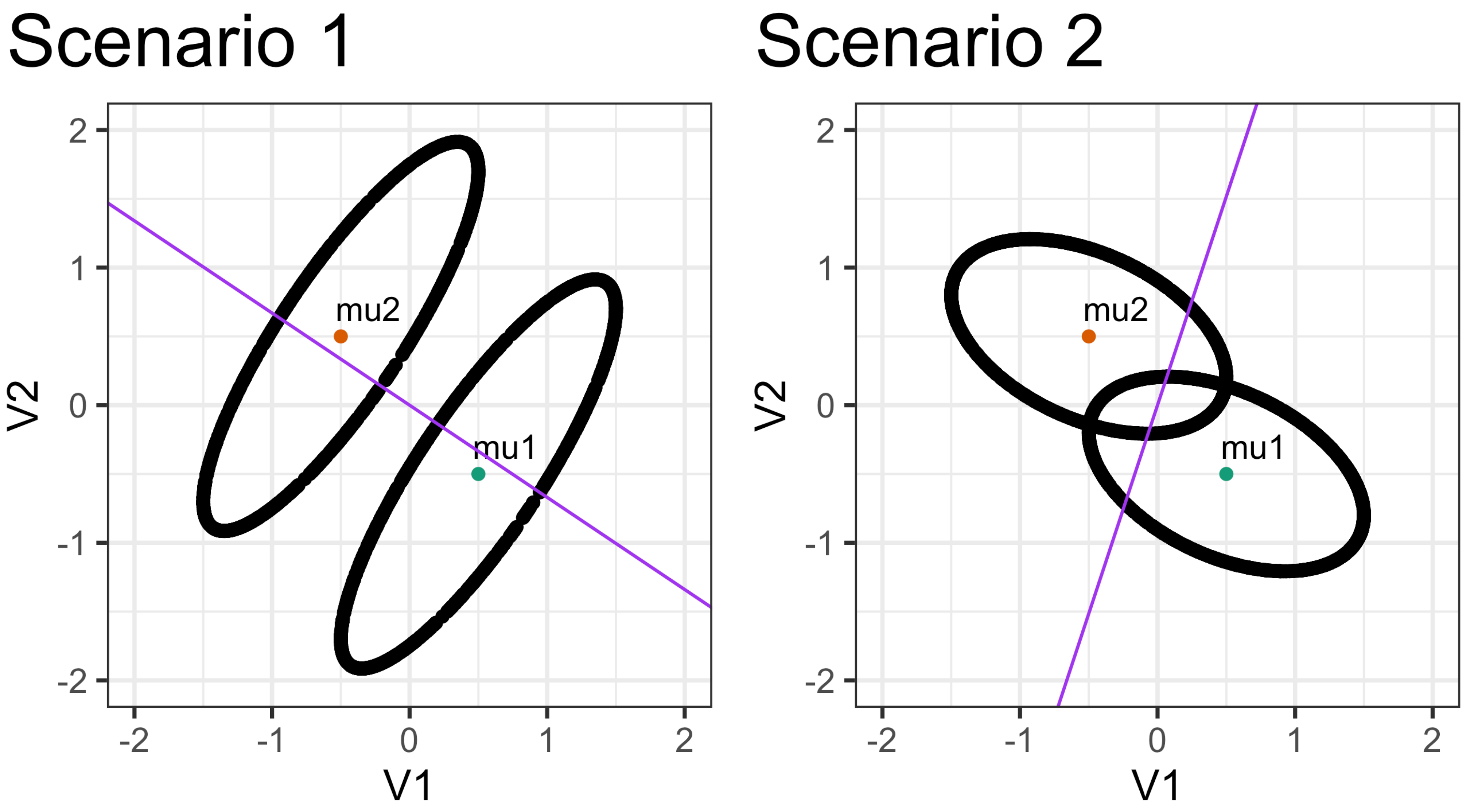
$$d(x, y) = \sqrt{(x-y)^\top \Sigma^{-1} (x-y)}$$

Which points are closest according to Euclidean distance? Which points are closest relative to the variance-covariance?



Discriminant space

In the means of scenarios 1 and 2 are the same, but the variance-covariances are different. The calculated discriminant space is different for different variance-covariances.



Notice: Means for groups are different, and variance-covariance for each group are the same.

Quadratic Discriminant Analysis

If the groups have different variance-covariance matrices, but still come from a normal distribution

Quadratic DA (QDA)

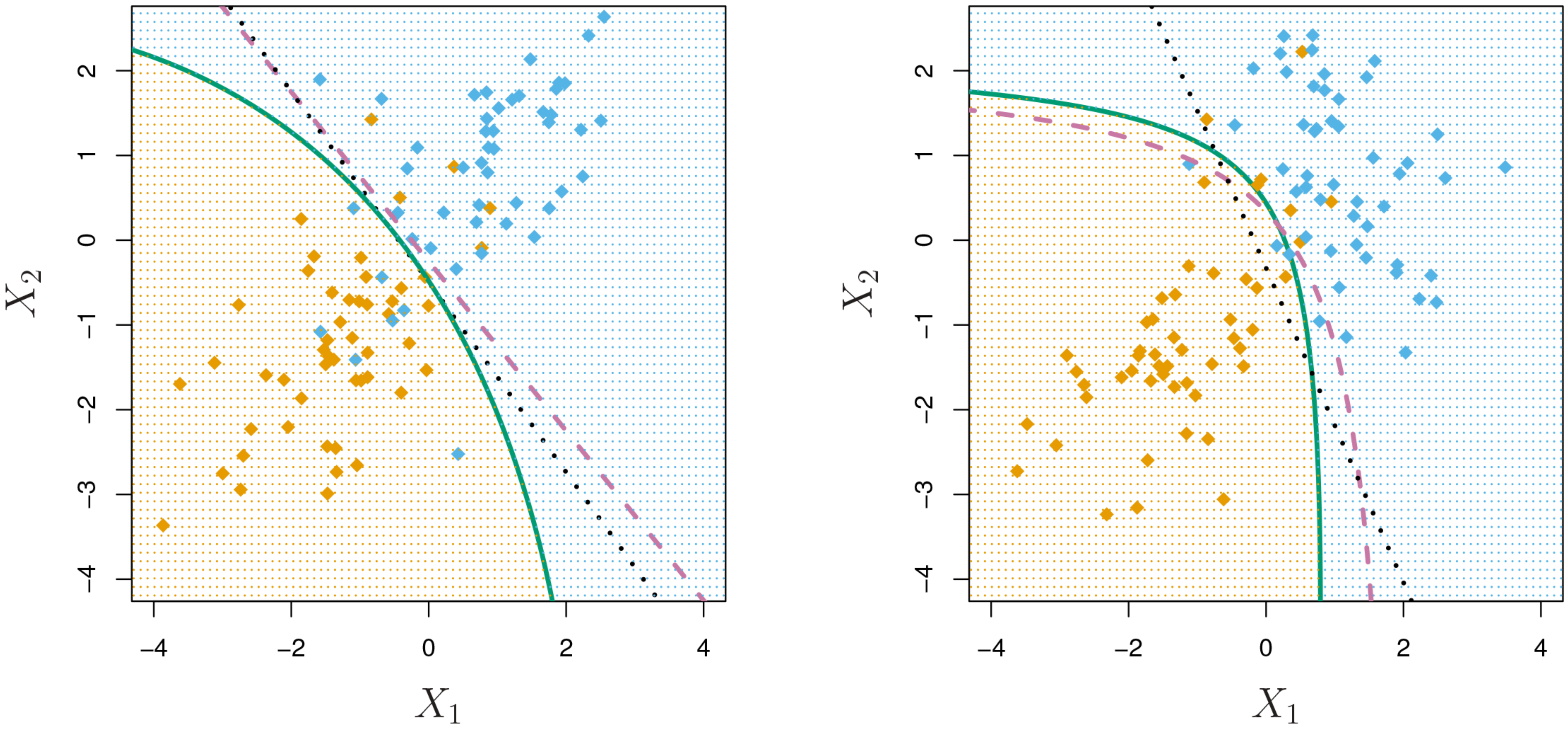
If the variance-covariance matrices for the groups are NOT EQUAL, then the discriminant functions are:

$$\delta_k(x) = x^\top \Sigma_k^{-1} x + x^\top \Sigma_k^{-1} \mu_k - \frac{1}{2}\mu_k^\top \Sigma_k^{-1} \mu_k - \frac{1}{2}\log|\Sigma_k| + \log(\pi_k)$$

where Σ_k is the population variance-covariance for class k, estimated by the sample variance-covariance S_k, and μ_k is the population mean vector for class k, estimated by the sample mean $\bar{x}_k$.

Quadratic DA (QDA)

A quadratic boundary is obtained by relaxing the assumption of equal variance-covariance, and assume that $\Sigma_k \neq \Sigma_l, \quad k \neq l, k, l = 1, \ldots, K$

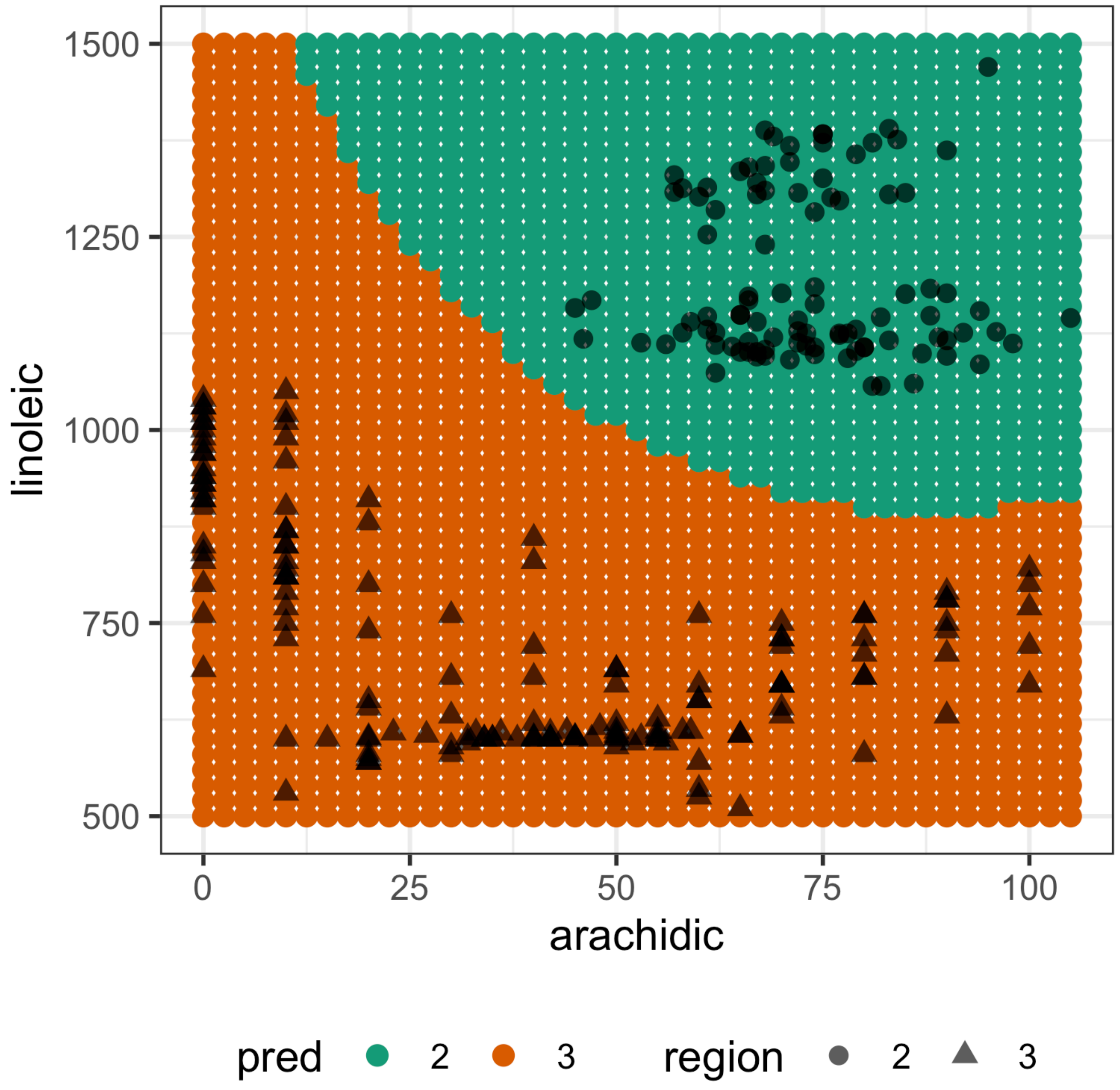


true, LDA, QDA.

(Chapter4/4.9.pdf)

QDA: Olive oils example

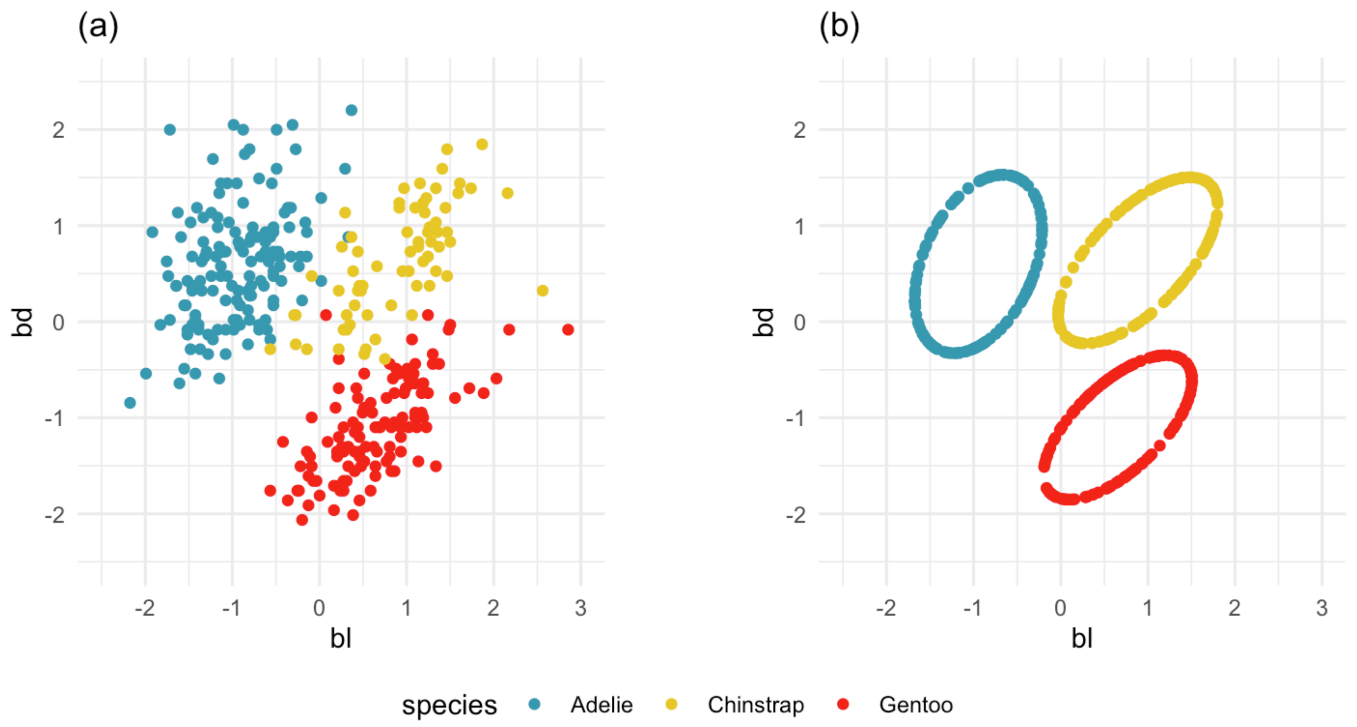
Even if the population is NOT normally distributed, QDA might do reasonably. On this data, region 3 has a “banana-shaped” variance-covariance, and region 2 has two separate clusters. The quadratic boundary though does well to carve the space into neat sections dividing the two regions.



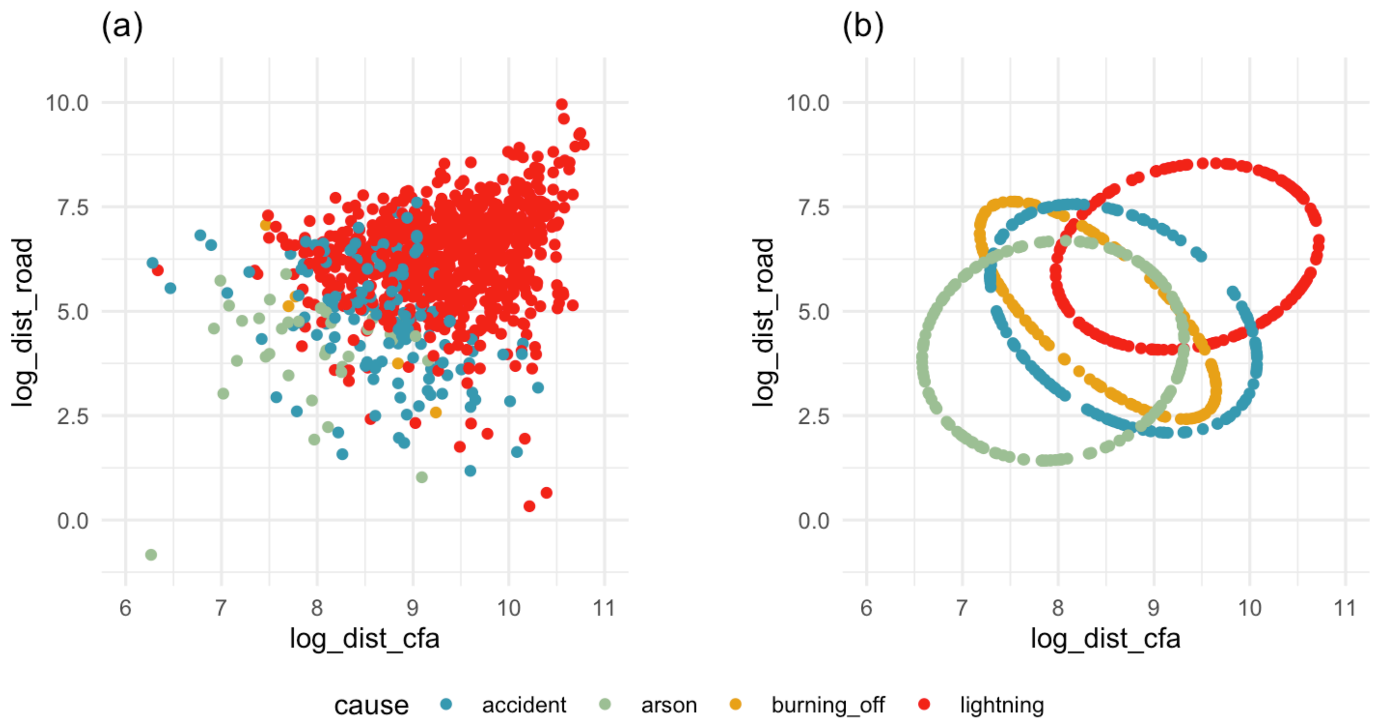
Checking the assumptions for LDA and QDA 1/2

Check the shape of the variability of each group could be considered to be elliptical, and the size is same for LDA but different to use QDA.

GOOD



BAD

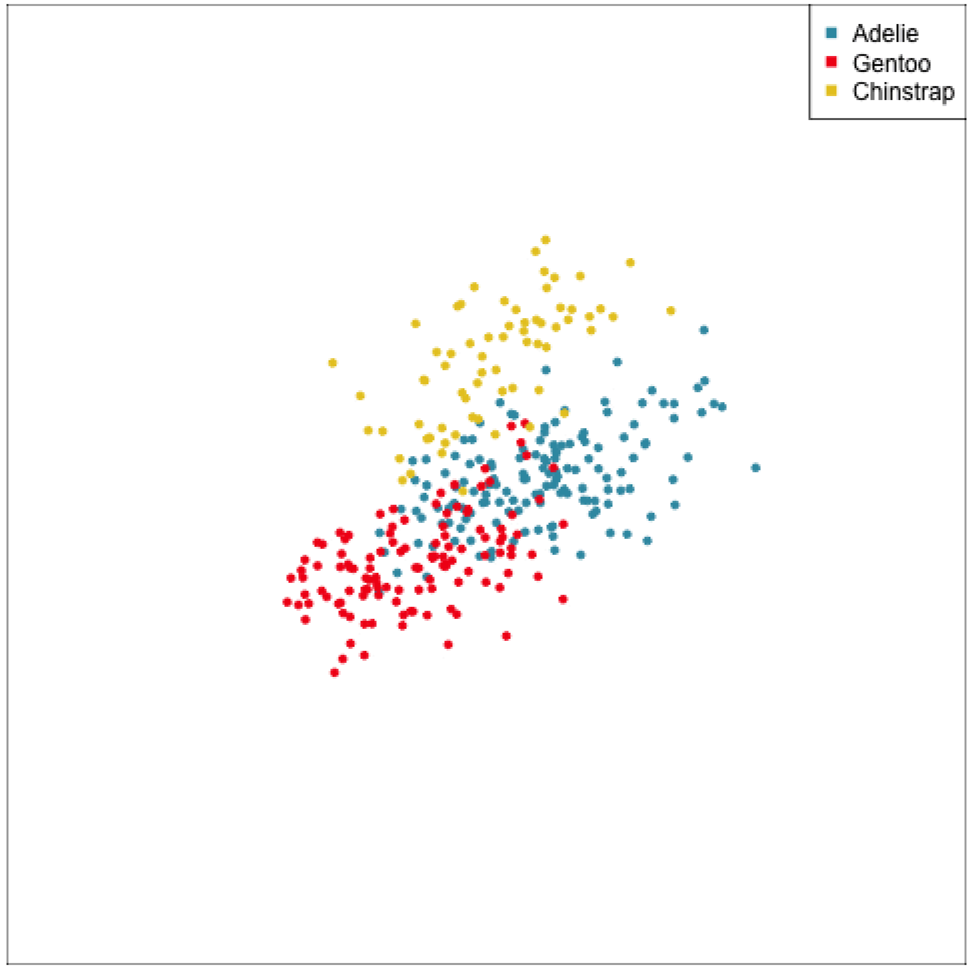


from Cook and Laa (2024)

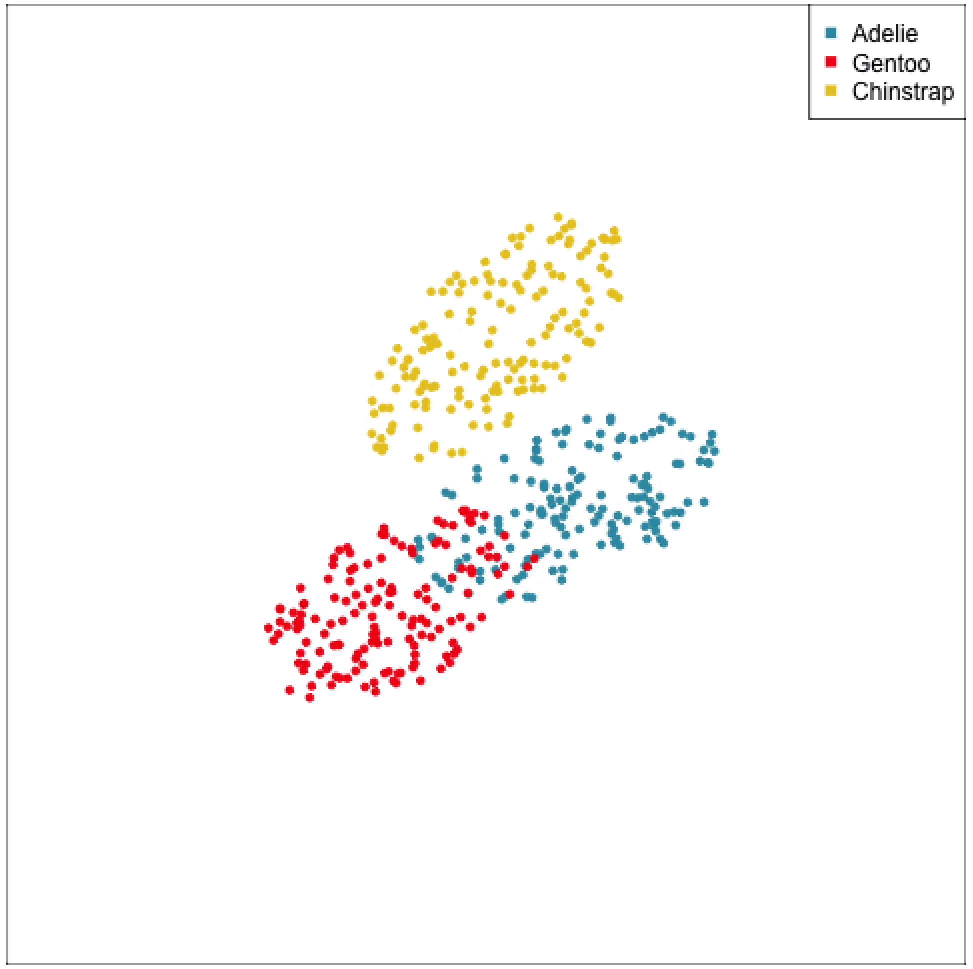
Checking the assumptions for LDA and QDA 2/2

This can also be done for $p > 2$.

DATA



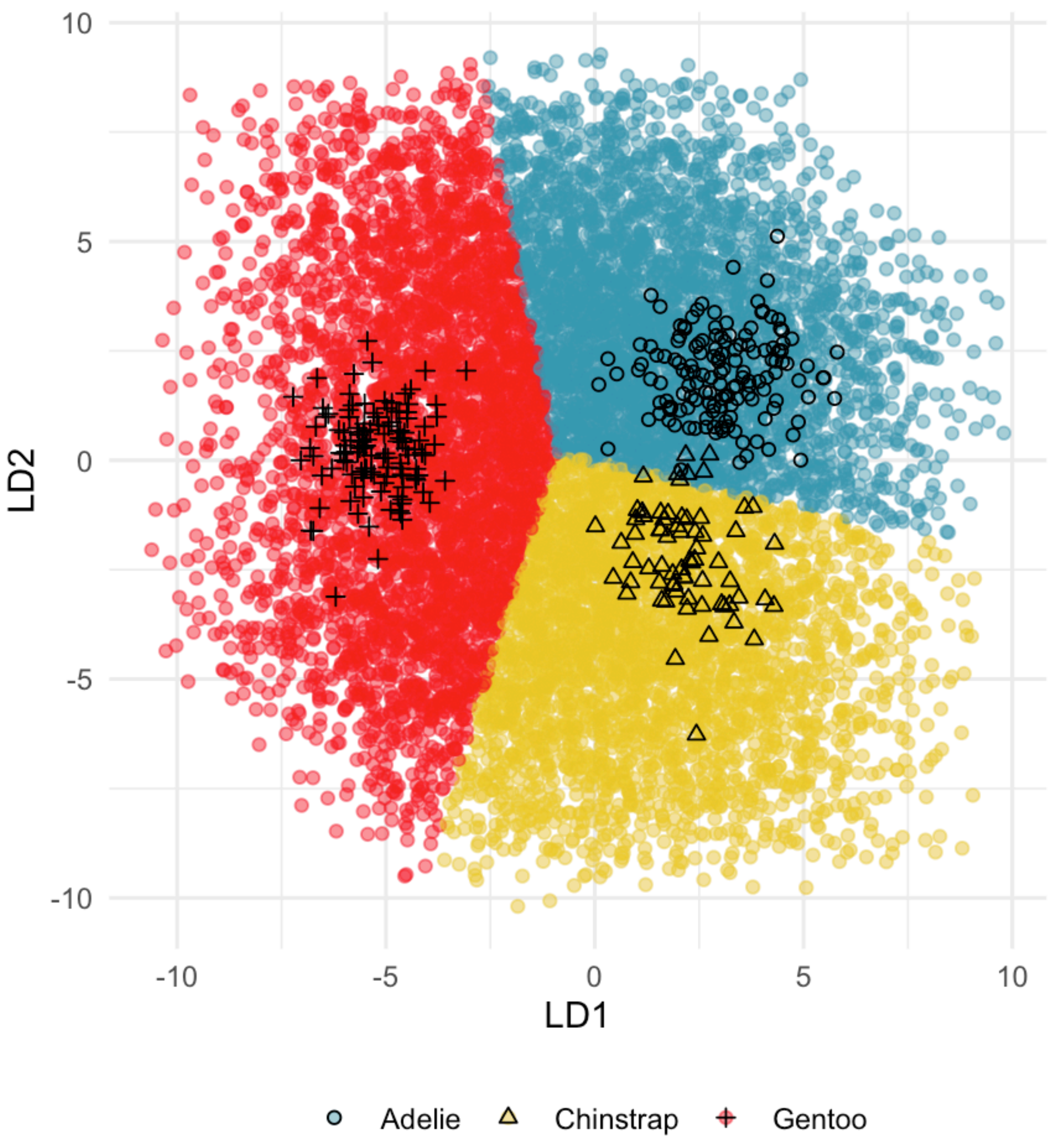
POINTS ON SURFACE OF ELLIPSES



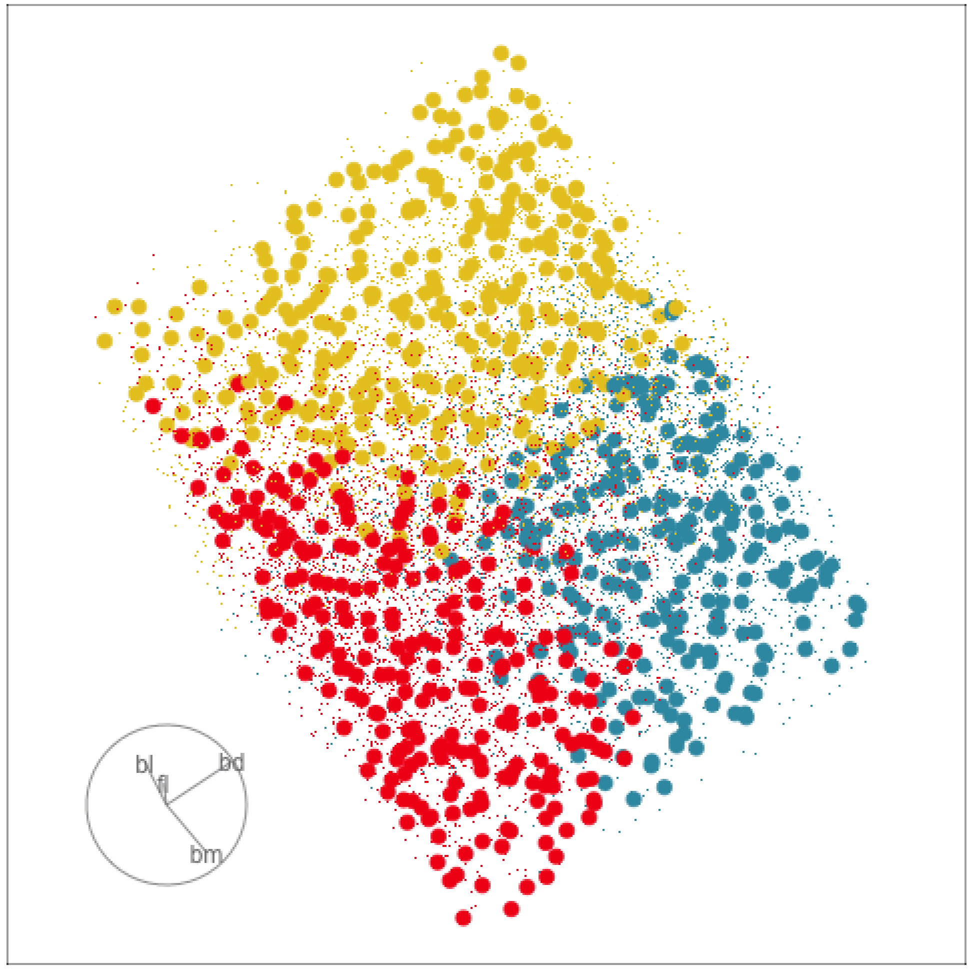
from Cook and Laa (2024)

Plotting the model

Data-in-the-model-space



Model-in-the-data-space



from Cook and Laa (2024)

Next: Trees and forests