

Homework Assignment 2 PSTAT 131

Quinlan Wilson and Jack Guo (both 131)

November 06, 2025

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.5
## v forcats    1.0.1      v stringr   1.5.2
## v ggplot2    4.0.0      v tibble    3.3.0
## v lubridate  1.9.4      v tidyr     1.3.1
## v purrr      1.1.0
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

## Warning: package 'ISLR' was built under R version 4.5.2

## Warning: package 'ROCR' was built under R version 4.5.2

##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##      select
```

Linear Regression

(1.)

```
glimpse(Auto)
```

```
## Rows: 392
## Columns: 9
## $ mpg      <dbl> 18, 15, 18, 16, 17, 15, 14, 14, 14, 15, 15, 14, 15, 14, 2~
## $ cylinders <dbl> 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 4, 6, 6, 6, 4, ~
## $ displacement <dbl> 307, 350, 318, 304, 302, 429, 454, 440, 455, 390, 383, 34~
## $ horsepower <dbl> 130, 165, 150, 150, 140, 198, 220, 215, 225, 190, 170, 16~
## $ weight      <dbl> 3504, 3693, 3436, 3433, 3449, 4341, 4354, 4312, 4425, 385~
## $ acceleration <dbl> 12.0, 11.5, 11.0, 12.0, 10.5, 10.0, 9.0, 8.5, 10.0, 8.5, ~
## $ year        <dbl> 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 7~
## $ origin      <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 1, 1, 1, 3, ~
## $ name        <fct> chevrolet chevelle malibu, buick skylark 320, plymouth sa~
```

```
Auto_clean <- Auto %>%
  dplyr::select(-contains("name")) %>%
  mutate(origin = as.factor(origin))

lmod <- lm(mpg ~ ., data = Auto_clean)
summary(lmod)
```

```
##
## Call:
## lm(formula = mpg ~ ., data = Auto_clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -9.009 -2.078 -0.098  1.986 13.361
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.80e+01  4.68e+00  -3.84  0.00014 ***
## cylinders    -4.90e-01  3.21e-01  -1.52  0.12821
## displacement  2.40e-02  7.65e-03   3.13  0.00186 **
## horsepower   -1.82e-02  1.37e-02  -1.33  0.18549
## weight       -6.71e-03  6.55e-04 -10.24 < 2e-16 ***
## acceleration  7.91e-02  9.82e-02   0.81  0.42110
## year          7.77e-01  5.18e-02  15.01 < 2e-16 ***
## origin2        2.63e+00  5.66e-01   4.64  4.7e-06 ***
## origin3        2.85e+00  5.53e-01   5.16  3.9e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.31 on 383 degrees of freedom
## Multiple R-squared:  0.824, Adjusted R-squared:  0.821
## F-statistic: 224 on 8 and 383 DF, p-value: <2e-16
```

With a 0.01 threshold we can reject the null hypothesis that there is no linear association between mpg and displacement, weight, year, and origin. While failing to reject that cylinders, horsepower, and acceleration have no linear association with mpg.

(2.)

```
training_MSE <- mean((Auto$mpg - predict(lmod, Auto_clean))^2)
```

The training MSE is 10.6821 and we cannot calculate the test MSE as the training set is the whole dataset so we have nothing to compare with.

(3.)

```
car <- data.frame(
  cylinders = 4,
  displacement = 133,
```

```

horsepower = 117,
weight = 3250,
acceleration = 29,
year = 97,
origin = factor(2)
)

car_pre_mpg <- predict(lmod, newdata = car)

```

The gas mileage predicted for an European car with 4 cylinders, displacement 133, horsepower of 117, weight of 3250, acceleration of 29, built in the year 1997 is 39.6351.

(4.)

```
coef(lmod)
```

##	(Intercept)	cylinders	displacement	horsepower	weight	acceleration
##	-17.95460	-0.48971	0.02398	-0.01818	-0.00671	0.07910
##	year	origin2	origin3			
##	0.77703	2.63000	2.85323			

The difference in mpg between a Japanese car and the mpg of an American car is 2.85323 and the difference between a European car and an American car is 2.63000.

(5.)

The change in mpg associated with a 30 unit increase in horsepower is -0.5455.

```

algae <- read.table("algaeBloom.txt", col.names=
  c('season','size','speed','mxPH','mnO2','Cl','NO3','NH4',
    'oPO4','PO4','Chla','a1','a2','a3','a4','a5','a6','a7'),
  na = "XXXXXXX")

algae.transformed <- algae %>% mutate_at(vars(4:11), funs(log(.)))
algae.transformed <- algae.transformed %>%
  mutate_at(vars(4:11),funs(ifelse(is.na(.),median(.,na.rm=TRUE),.)))
# a1 == 0 means low
algae.transformed <- algae.transformed %>% mutate(a1 = factor(as.integer(a1 > 5), levels = c(0, 1)))

calc_error_rate <- function(predicted.value, true.value) {
  return(mean(true.value != predicted.value))
}

set.seed(1)
test.indices = sample(1:nrow(algae.transformed), 50)
algae.train=algae.transformed[-test.indices,]
algae.test=algae.transformed[test.indices,]

```

Algae Classification using Discriminant Analysis

(1.)

```
lda_mod <- lda(a1 ~ ., data = algae.transformed, CV = T)

lda_class <- lda_mod$class

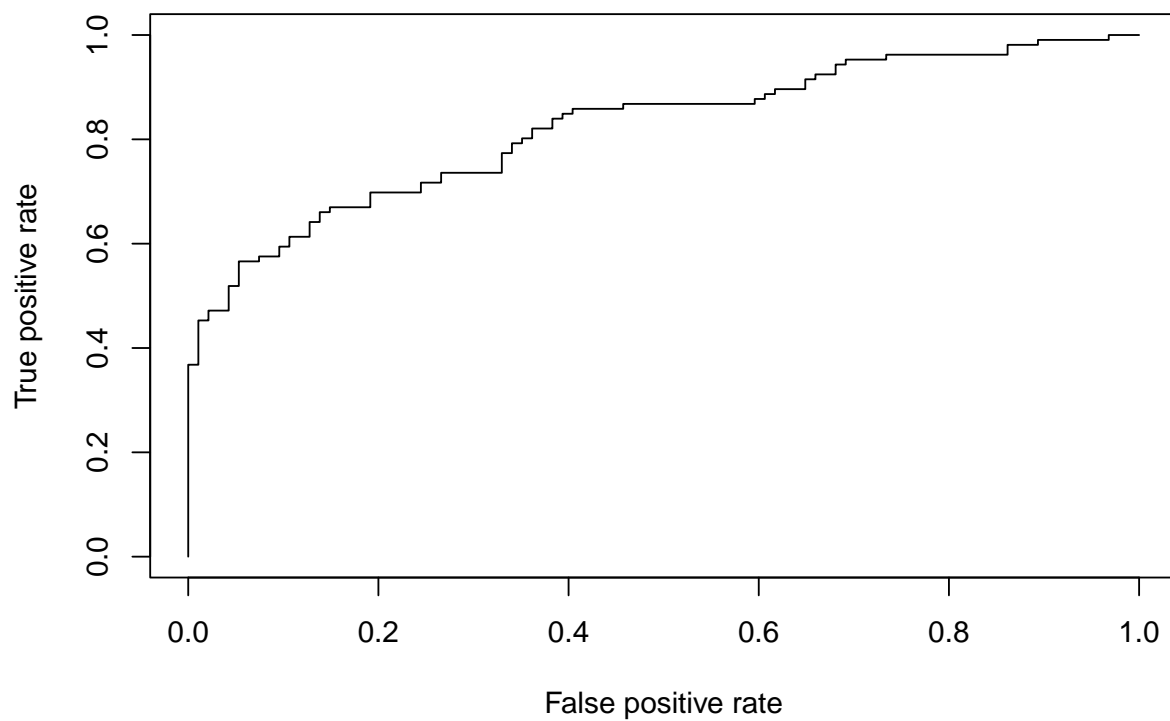
lda_error <- calc_error_rate(lda_class, algae.train$a1)
lda_post <- lda_mod$posterior

if (ncol(lda_post) == 1) {
  lda_post <- cbind("0" = 1 - lda_post, "1" = lda_post)
}

lda_probs <- lda_post[, "1"]

pred <- prediction(lda_mod$posterior[, "1"], algae.transformed$a1)
perf <- performance(pred, 'tpr', 'fpr')

plot(perf)
```



```
pred_lda <- prediction(lda_probs, algae.transformed$a1)
performance(pred_lda, "auc")@y.values[[1]]
```

```
## [1] 0.8232
```

(2.)

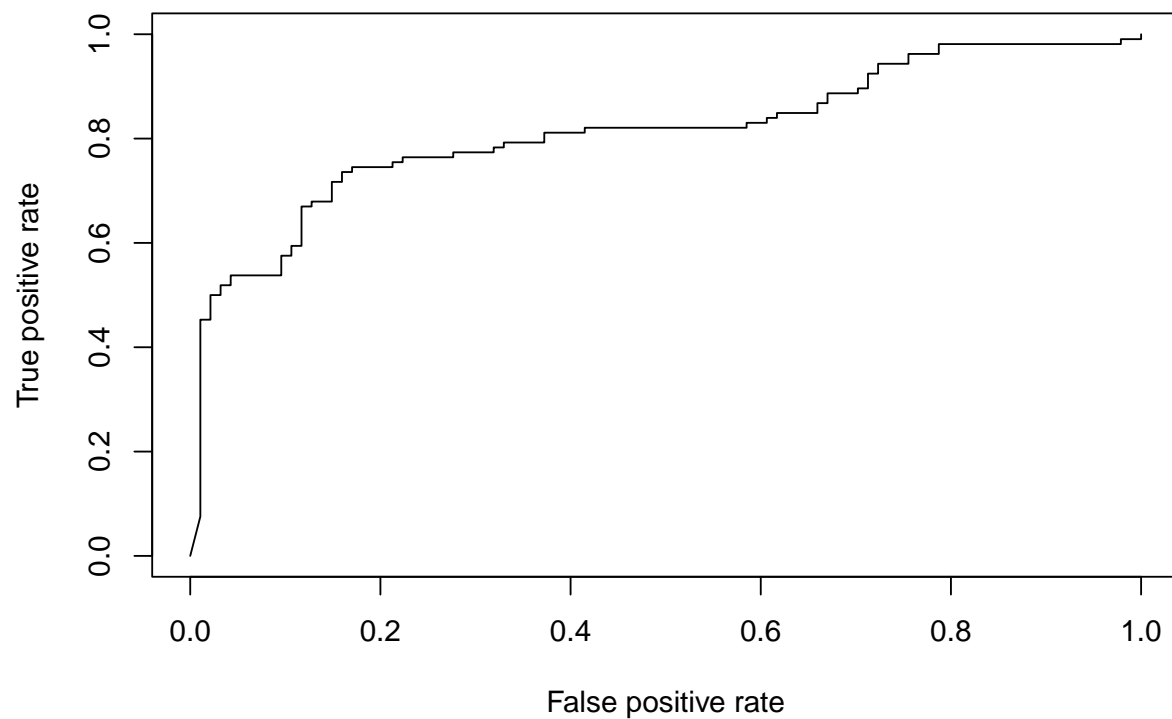
```
qda_mod <- qda(a1 ~ ., data = algae.transformed, CV = T)
qda_class <- qda_mod$class
qda_post <- qda_mod$posterior

if (ncol(qda_post) == 1) qda_post <- cbind("0" = 1 - qda_post, "1" = qda_post)

qda_probs <- qda_post[, "1"]

pred_qda <- prediction(qda_probs, algae.transformed$a1)
perf_qda <- performance(pred_qda, "tpr", "fpr")
auc_qda <- performance(pred_qda, "auc")@y.values[[1]]

plot(perf_qda)
```



```
auc_qda
```

```
## [1] 0.8131
```

LDA has a higher area under the ROC curve between the two models meaning it performs better at separating the two classes. LDA has the advantage of lower variance but in the case of different contrivances more bias is introduced.

Fundamentals of the bootstrap

(1.)

$$\left(1 - \frac{1}{n}\right)^n$$

(2.)

```
(1-1/1000)^1000
```

```
## [1] 0.3677
```

(3.)

```
n <- 1000  
sample <- sample(1:n, size = n, replace = T)  
missing <- n - length(unique(sample))  
ratio <- missing / n  
ratio
```

```
## [1] 0.371
```