10.2 Exercise

Decker, Reuben

2023-11-05

```
library(readxl)

library(broom)

library(kableExtra)
```

Pulling in Thoracic Surgery Data

```
## 1. Fit a Logistic Regression Model to Thoracic Surgery Binary Dataset

## a. For this problem, you will be working with the thoracic surgery data set from th
e University of California Irvine machine learning repository. This dataset contains i
nformation on life expectancy in lung cancer patients after surgery. The underlying th
oracic surgery data is in ARFF format. This is a text-based format with information on
each of the attributes. You can load this data using a package such as foreign or by c
utting and pasting the data section into a CSV file.


## Download the Excel file

lung_cancer_data <- read_excel("C:\\Users\\Reuben Decker\\Downloads\\Lung Cancer Data.
xlsx")



## Seeing the Excel file

head(lung_cancer_data)

# A tibble: 6 × 17
  DGN TYPES {DGN3,DGN2,DG…¹  PRE4  PRE5 PRE6 TYPES {PRZ2,PRZ…² PRE7  PRE8  PRE9
  <chr>                     <dbl> <dbl> <chr>                  <chr> <chr> <chr>
1 DGN2                       2.88  2.16 PRZ1                   F     F     F
2 DGN3                       3.4   1.88 PRZ0                   F     F     F
3 DGN3                       2.76  2.08 PRZ1                   F     F     F
4 DGN3                       3.68  3.04 PRZ0                   F     F     F
5 DGN3                       2.44  0.96 PRZ2                   F     T     F
6 DGN3                       2.48  1.88 PRZ1                   F     F     F
# i abbreviated names: ¹`DGN TYPES {DGN3,DGN2,DGN4,DGN6,DGN5,DGN8,DGN1}`,
#   ²`PRE6 TYPES {PRZ2,PRZ1,PRZ0}`
```

```
# ℹ 10 more variables: PRE10 <chr>, PRE11 <chr>,
#   `PRE14 TYPES {OC11,OC14,OC12,OC13}` <chr>, PRE17 <chr>, PRE19 <chr>,
#   PRE25 <chr>, PRE30 <chr>, PRE32 <chr>, AGE <dbl>,
#   `PATIENT SURVIVED 1 YEAR` <chr>
```

1st Binary Logistic Regression Model

```
## Assignment Instructions:

## i. Fit a binary logistic regression model to the data set that predicts whether or
not the patient survived for one year (the Risk1Y variable) after the surgery. Use the
glm() function to perform the logistic regression. See Generalized Linear Models for a
n example. Include a summary using the summary() function in your results.


## Converting F & T to 0 & 1 for binary modeling

lung_cancer_data$'PATIENT SURVIVED 1 YEAR'<- ifelse(lung_cancer_data$'PATIENT SURVIVED
1 YEAR'== "T", 1, 0)


## Fitting a logistic regression model

logistic_model <- glm(`PATIENT SURVIVED 1 YEAR` ~ `DGN TYPES {DGN3,DGN2,DGN4,DGN6,DGN5
,DGN8,DGN1}` + PRE4 + PRE5 + `PRE6 TYPES {PRZ2,PRZ1,PRZ0}` + PRE7 + PRE8 + PRE9 + PRE1
0 + PRE11 + `PRE14 TYPES {OC11,OC14,OC12,OC13}` + PRE17 + PRE19 + PRE25 + PRE30 + PRE3
2 + AGE, data = lung_cancer_data, family = binomial)


## Extract model coefficients & other info out of the model

model_summary <- tidy(logistic_model)


## Simple and nicely formatted summary table

model_summary %>%
  kable() %>%
  kable_styling(full_width = FALSE)
```

term	estimate	std.error	statistic	p.value
(Intercept)	-16.5516983	2399.5452397	-0.0068978	0.9944964
`DGN TYPES {DGN3,DGN2,DGN4,DGN6,DGN5,DGN8,DGN1}`DGN2	14.7362756	2399.5447796	0.0061413	0.9951000
`DGN TYPES {DGN3,DGN2,DGN4,DGN6,DGN5,DGN8,DGN1}`DGN3	14.1805519	2399.5447590	0.0059097	0.9952848

term	estimate	std.error	statistic	p.value
DGN TYPES {DGN3,DGN2,DGN4,DGN6,DGN5,DGN8,DGN1}DGN4	14.6083288	2399.5447886	0.0060880	0.9951425
DGN TYPES {DGN3,DGN2,DGN4,DGN6,DGN5,DGN8,DGN1}DGN5	16.3813211	2399.5448205	0.0068268	0.9945530
DGN TYPES {DGN3,DGN2,DGN4,DGN6,DGN5,DGN8,DGN1}DGN6	0.4088535	2673.0490909	0.0001530	0.9998780
DGN TYPES {DGN3,DGN2,DGN4,DGN6,DGN5,DGN8,DGN1}DGN8	18.0328623	2399.5452109	0.0075151	0.9940039
PRE4	-0.2272448	0.1849113	-1.2289396	0.2190945
PRE5	-0.0303035	0.0178581	-1.6969035	0.0897149
PRE6 TYPES {PRZ2,PRZ1,PRZ0}PRZ1	-0.4427150	0.5199082	-0.8515253	0.3944776
PRE6 TYPES {PRZ2,PRZ1,PRZ0}PRZ2	-0.2937007	0.7906902	-0.3714486	0.7103035
PRE7T	0.7153410	0.5555597	1.2876042	0.1978838
PRE8T	0.1743366	0.3891858	0.4479521	0.6541878
PRE9T	1.3682164	0.4867680	2.8108182	0.0049416
PRE10T	0.5769579	0.4825698	1.1955946	0.2318548
PRE11T	0.5161808	0.3964803	1.3019079	0.1929479
PRE14 TYPES {OC11,OC14,OC12,OC13}OC12	0.4393639	0.3300915	1.3310364	0.1831770
PRE14 TYPES {OC11,OC14,OC12,OC13}OC13	1.1792074	0.6165465	1.9126011	0.0557991
PRE14 TYPES {OC11,OC14,OC12,OC13}OC14	1.6529730	0.6093624	2.7126272	0.0066752
PRE17T	0.9265934	0.4444619	2.0847532	0.0370917
PRE19T	-14.6553784	1653.5410538	-0.0088630	0.9929284

term	estimate	std.error	statistic	p.value
PRE25T	-0.0978945	1.0033145	-0.0975711	0.9222729
PRE30T	1.0839970	0.4990305	2.1722061	0.0298401
PRE32T	-13.9832946	1645.3138921	-0.0084989	0.9932190
AGE	-0.0095057	0.0180990	-0.5252038	0.5994415

```
## ii. According to the summary, which variables had the greatest effect on the surviv
al rate?


## DGN TYPES {DGN3,DGN2,DGN4,DGN6,DGN5,DGN8,DGN1}DGN5: This variable has an estimated
coefficient with a large positive value, indicating a strong positive effect on the lo
g-odds of survival.


## DGN TYPES {DGN3,DGN2,DGN4,DGN6,DGN5,DGN8,DGN1}DGN8: This variable also has a large
positive coefficient, suggesting a strong positive effect on the log-odds of survival.


## PRE9T: This variable has a positive coefficient and a relatively low p-value, indic
ating a significant positive effect on survival.


## PRE14 TYPES {OC11,OC14,OC12,OC13}OC14: This variable has a positive coefficient and
a low p-value, indicating a significant positive effect on survival.


## PRE6 TYPES {PRZ2,PRZ1,PRZ0}PRZ1: This variable has a negative coefficient and a low
p-value, suggesting a significant negative effect on survival.


## This is all based on that positive coefficients indicate an increase in the log-odd
s of survival, while negative coefficients indicate a decrease. The magnitude of the c
oefficients indicates the strength of the effect.

## iii. To compute the accuracy of your model, use the dataset to predict the outcome
variable. The percent of correct predictions is the accuracy of your model. What is th
e accuracy of your model?


## Making predictions using the logistic regression model

predictions <- predict(logistic_model, newdata = lung_cancer_data, type = "response")


## Converting predicted probabilities to binary predictions (0 or 1)

predicted_classes <- ifelse(predictions > 0.5, 1, 0)
```

```
## Comparing predicted classes to the actual outcomes

correct_predictions <- sum(predicted_classes == lung_cancer_data$`PATIENT SURVIVED 1 Y
EAR`)


## Calculating accuracy

accuracy <- correct_predictions / nrow(lung_cancer_data)


accuracy

[1] 0.8361702

## This means that 83.6% of the predictions made by my model, matched the actual outco
mes in the dataset.
```

2nd Binary Logisic Regression Model

```
## 2. Fit a Logistic Regression Model

## a. Fit a logistic regression model to the binary-classifier-data.csv dataset


## b. The dataset (found in binary-classifier-data.csv) contains three variables; labe
l, x, and y. The label variable is either 0 or 1 and is the output we want to predict
using the x and y variables.


## Download the Excel file

binary_classifier_data <- read_excel("C:\\Users\\Reuben Decker\\Downloads\\binary_clas
sifier_data.xlsx")


## Seeing the Excel file

head(binary_classifier_data)

# A tibble: 6 × 3

  label     x     y

  <dbl> <dbl> <dbl>

1     0  70.9  83.2

2     0  75.0  87.9

3     0  73.8  92.2

4     0  66.4  81.1

5     0  69.1  84.5

6     0  72.2  86.4

## Converting the label column into factor
```

```
binary_classifier_data$label <- as.factor(binary_classifier_data$label)



## Converting the x column into factor

## binary_classifier_data$x <- as.factor(binary_classifier_data$x)


## Converting the y column into factor

## binary_classifier_data$y <- as.factor(binary_classifier_data$y)


## Fitting a logistic regression model for x variable

logistic_model_x <- glm(label ~ x + y, data = binary_classifier_data, family = binomia
l)


## Extract model coefficients & other info out of the model

model_summary <- tidy(logistic_model_x)


## Simple and nicely formatted summary table

model_summary %>%

  kable() %>%

  kable_styling(full_width = FALSE)
```

term	estimate	std.error	statistic	p.value
(Intercept)	0.4248088	0.1172235	3.623921	0.0002902
x	-0.0025709	0.0018225	-1.410625	0.1583551
y	-0.0079555	0.0018689	-4.256869	0.0000207

```
## Making predictions using the logistic regression model

predictions <- predict(logistic_model_x, newdata = binary_classifier_data, type = "res
ponse")


## Converting predicted probabilities to binary predictions (0 or 1)

predicted_classes <- ifelse(predictions > 0.5, 1, 0)


## Comparing predicted classes to the actual outcomes

correct_predictions <- sum(predicted_classes == binary_classifier_data$label)
```

```
## Calculating accuracy

accuracy <- correct_predictions / nrow(binary_classifier_data)


accuracy

[1] 0.5834446

## This means that 58.3% of the predictions made by my model, matched the actual outco
mes in the dataset.
```