2301 DSCI353-353m-453: LE2: Classification, Statistics, and Deep Learning

Profs: R. H. French, L. S. Bruckman, P. Leu, K. Davis, S. Cirlos

TAs: W. Oltjen, K. Hernandez, M. Li, M. Li, D. Colvin

01 February, 2023

Contents

2.1.1 LE2, 9 points.

2.1.1.1 Code style (1 point) Details

- Due

  - By Midnight of due date
- The grading is done on how you show your thinking,
  - explain yourself and
  - show your R code and
  - the output you got from your code.
- Code style is important
  - Follow Rstudio code diagnostics notices
  - And the Google R Style Guide

To be done as an Rmd file,

- where you turn in
  - the Rmd file and
  - the compiled pdf showing your work.
  - and the R script of IntroR.R

You will want to produce a report type format

- (html and pdf type document) to turn in.
- And not an ioslides or beamer (slide type) compiled output.
  - These are presentation formats, and can be fussy

Also are you backing up your git repo

- in a second and third location,
- to avoid corruption problems?

2.1.2 LE2a: Classification (3 Points)

2.1.2.1 2a-1. Classification of the Weekly dataset

- This question should be answered using the Weekly data set,
  - which is part of the ISLR2 R package.

This data is similar in nature to the Smarket data

- from this Chapter's lab, From ISLR2, page 171
  - 4.7Lab: Classification Methods,
  - 4.7.1 The Stock Market Data
- except that it contains
  - 1,089 weekly returns
  - for 21 years,
  - from the beginning of 1990 to the end of 2010.

2.1.2.1.1 (a) EDA: Exploratory data analysis Create a matrix of scatterplots for the Weekly data set.
Create a correlation matrix of the Weekly data set for all continuous variables (ones that are not categorical)

```
library(ISLR2)
names(Weekly)
```

```
## [1] "Year"      "Lag1"      "Lag2"      "Lag3"      "Lag4"      "Lag5"
## [7] "Volume"    "Today"     "Direction"
```

```
dim(Weekly)
```

```
## [1] 1089    9
```

```
summary(Weekly)
```

```
##       Year           Lag1               Lag2               Lag3         
##  Min.   :1990   Min.   :-18.1950   Min.   :-18.1950   Min.   :-18.1950  
##  1st Qu.:1995   1st Qu.: -1.1540   1st Qu.: -1.1540   1st Qu.: -1.1580  
##  Median :2000   Median :  0.2410   Median :  0.2410   Median :  0.2410  
##  Mean   :2000   Mean   :  0.1506   Mean   :  0.1511   Mean   :  0.1472  
##  3rd Qu.:2005   3rd Qu.:  1.4050   3rd Qu.:  1.4090   3rd Qu.:  1.4090  
##  Max.   :2010   Max.   : 12.0260   Max.   : 12.0260   Max.   : 12.0260  
##       Lag4               Lag5              Volume            Today         
##  Min.   :-18.1950   Min.   :-18.1950   Min.   :0.08747   Min.   :-18.1950  
##  1st Qu.: -1.1580   1st Qu.: -1.1660   1st Qu.:0.33202   1st Qu.: -1.1540  
##  Median :  0.2380   Median :  0.2340   Median :1.00268   Median :  0.2410  
##  Mean   :  0.1458   Mean   :  0.1399   Mean   :1.57462   Mean   :  0.1499  
##  3rd Qu.:  1.4090   3rd Qu.:  1.4050   3rd Qu.:2.05373   3rd Qu.:  1.4050  
##  Max.   : 12.0260   Max.   : 12.0260   Max.   :9.32821   Max.   : 12.0260  
##  Direction
##  Down:484
##  Up  :605
##
##
##
##
```

```
sapply(Weekly, class)
```

```
##      Year      Lag1      Lag2      Lag3      Lag4      Lag5    Volume     Today 
## "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" 
## Direction 
##  "factor"
```

Where do you see a strong positive correlation?

Answer: There is fairly strong positive correlation between Year and Volume.

2.1.2.1.2 (b) Use the full data set to train a logistic regression model

- with Direction as the response
- and the five lag variables plus Volume as predictors.

Use the summary function to print the results.

Do any of the predictors appear to be statistically significant?

If so, which ones?

Answer: Lag2 is statistically significant. $\Pr(>|z|) = 0.03$

2.1.2.1.3 (c) Compute:

- the confusion matrix and
- overall fraction of correct predictions.

Any prediction where the probability is greater than .5 should be "Up"; Otherwise it should be "Down"

Explain what the confusion matrix is telling you

- about the types of mistakes made by logistic regression.

What percentage of predictions are correct?

Answer: Correct predictions: 56.1%. It is making a lot of mistakes predicting the market will go up, when it is actually going down. The model is a lot more optimistic about the market then it should be.

2.1.2.1.4 (d) Now fit the logistic regression model

- using a training data period from 1990 to 2008,
- with Lag2 as the only predictor.

Compute

- the confusion matrix and
- the overall fraction of correct predictions
- for the held out (testing) data
  - (that is, the data from 2009 and 2010).

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

How well does this model perform?
How often is the prediction correct?
- Explain your answer and reasoning

Answer: The prediction is only right 62.5% of the time.

```
library(MASS)
```

2.1.2.1.5 (e) Repeat (d) using LDA.

```
##
## Attaching package: 'MASS'
```

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

```
## The following object is masked from 'package:ISLR2':
##
##     Boston
```

How well does this model perform?

- Explain your answer and reasoning
- And compare to Logistic Regression Model

Answer: This model is the same as the logistic regression model. It is correct 62.5% of the time.

```
library(class)
```

2.1.2.1.6 (f) (g) Repeat (d) using KNN with K = 1. How well does this model perform?

- Explain your answer and reasoning

Answer: It is right 50% of the time.

2.1.2.1.7 (h) Which of these methods

- appears to provide the best results on this data?

Answer: Logistic regression and linear discriminant analysis seems to be do the best.

2.1.2.2 2a-2. Writing Functions

- This problem involves writing functions.

(a) Create a new function, Power(),

that allows you to pass any two numbers,

- x and a,
- and prints out the value of x^a.

You can do this by beginning your function with the line

- `Power = function (x , a ) {`

You should be able to call your function by entering,

- for instance,
- `Power(3 ,8)` on the command line.

This should output the value of 3^8,

- namely, 6,561.

```
Power <- function(x, a) {

}

Power(3,8)
```

```
## NULL
```

2.1.2.2.1 (b) Using the Power() function that you just wrote,

- compute 10^3, 8^17, and 131^3.

2.1.2.2.2 (c) Now create a new function, Power2(),

- that actually returns the result x^a as an R object,
- rather than simply printing it to the screen.

That is, if you store the value x^a

- in an object called result within your function,
- then you can simply return() this result,
- using the following line:
  - `return ( result )`

The line above should be

- the last line in your function,
- before the } symbol.

```
Power2 <- function(x, a) {

}
```

2.1.2.2.3 (d) Now using the Power2() function,

- create a plot of $f(x) = x^2$.
- use ggplot2

The x-axis should display

- a range of integers from 1 to 10,

and the y-axis should

- display x^2.

Label the axes appropriately,

- and use an appropriate title for the figure.

Consider displaying either

- the x-axis, the y-axis,
- or both on the log-scale.

You can do this by using

- log="x", log="y", or log="xy"
  - as arguments to the plot() function.
- but do it using ggplot2.

2.1.2.2.4 (e) Create a function, PlotPower(),

- that allows you to create a plot
- of x against x^a
- for a fixed a
- and for a range of values of x.

For instance, if you call

- `PlotPower (1:10 ,3)`
- then a plot should be created with
  - an x-axis taking on values 1, 2, . . . , 10,
  - and a y-axis taking on values 13 , 23 , . . . , 103 .

```
PlotPower <- function(x, a) {

}

PlotPower(1:10,3)
```

```
## NULL
```

2.1.2.3 2a-3. Combined exercise

- Recall question 2a-1(c),

- where we constructed a confusion matrix
- using the full dataset and logistic regression
- with all five variables as predictors.

Now that you have some practice with functions,

- write a function that uses the predicted values and results from 2a-1(c) that
- prints a confusion matrix,
- and prints the percentage of correct predictions.

2.1.3 LE2b: Linear and Multiple Regression Concept Questions (2pts)

2.1.3.1 2b-1 linear regression

- Why is linear regression important to understand? Select all that apply:

1. The linear model is often correct
2. Linear regression is very extensible and can be used to capture nonlinear effects
3. Simple methods can outperform more complex ones if the data are noisy
4. Understanding simpler methods sheds light on more complex ones

Answer :

2. The feature vectors could be X, X^2, log(X), etc.
3. Complex ones may start to capture noise and overfit the data; less bias, but more variance
4. Good basis for understanding mroe complex regression.

2.1.3.2 2b-2 confidence intervals

- You may want to reread the paragraph on confidence intervals on page 66 of the textbook before trying this question (the distinctions are subtle).

Which of the following are true statements? Select all that apply:

1. A 95% confidence interval is a random interval that contains the true parameter 95% of the time
2. The true parameter is a random value that has 95% chance of falling in the 95% confidence interval
3. I perform a linear regression and get a 95% confidence interval from 0.4 to 0.5. There is a 95% probability that the true parameter is between 0.4 and 0.5.
4. The true parameter (unknown to me) is 0.5. If I sample data and construct a 95% confidence interval, the interval will contain 0.5 95% of the time.

Answer :

2.1.3.3 2b-3 linear regression and $\beta_1 = b$

- We run a linear regression and the slope estimate is 0.5 with estimated standard error of 0.2.

What is the largest value of b for which we would NOT reject the null hypothesis that $\beta_1 = b$?

- (assume normal approximation to t distribution, and
- that we are using the 5% significance level for a two-sided test;
- need two significant digits of accuracy)

Answer :

2.1.3.4 2b-4 Strength of predictor/response relationship

- Which of the following indicates a fairly strong relationship between X and Y?
  1. $R^2 = 0.9$
  2. The p-value for the null hypothesis $\beta_1 = 0$ is 0.0001
  3. The t-statistic for the null hypothesis $\beta_1 = 0$ is 30

Answer :

2.1.3.5 2b-5 Causal relationships

- Suppose we are interested in learning about a relationship between X1 and Y,
  - which we would ideally like to interpret as causal.

True or False?

The estimate $\hat{\beta}_1$ in a linear regression that controls for many variables (that is, a regression with many predictors in addition to X_1) is usually a more reliable measure of a causal relationship than $\hat{\beta}_1$ from a univariate regression on X_1.

Answer :

Adding many predictors to the model can make it difficult to interpret beta_1.

2.1.3.6 2b-6 Interaction terms

- What is the difference between `lm(y ~ x*z)` and `(y ~ I(x*z))`, when x and z are both numeric variables?
  1. The first one includes an interaction term between x and z, whereas the second uses the product of x and z as a predictor in the model.
     2.The second one includes an interaction term between x and z, whereas the first uses the product of x and z as a predictor in the model.
  2. The first includes only an interaction term for x and z, while the second includes both interaction effects and main effects.
  3. The second includes only an interaction term for x and z, while the first includes both interaction effects and main effects.

Answer : _____

2.1.4 LE2c: Clustering Life Expectancy (3 points)

- Find the best number of clusters in the life.expectancy.1971 dataset

We will use the life.expectancy.1971 dataset

- about life expectancy in several countries in 1971,
- which is part of the cluster.datasets package.

It includes 10 attributes:

- the country were the data has been collected,
- the year of data collection, and
- the life expectancy (remaining) for
  - male and female individuals
  - aged 0 years old, 25, 50, and 75.

As with the previous crime dataset,

- this one also does not specify the membership of our cases to categories.

So again, we will have to decide on the number of clusters by ourselves.

- we will examine how to do so more precisely.
- We will create a function for this purpose.

2.1.4.1 2c-1 EDA of the dataset

- Before we do that, let's do some EDA to discover the dataset we will use.

Let's start by loading and examining to dataset.

Show a summary of the dataset

```
library(cluster.datasets)
data(life.expectancy.1971)
summary(life.expectancy.1971)
```

```
##    country               year            m0              m25
##  Length:31          Min.   :1960   Min.   :34.00   Min.   :29.00
##  Class :character   1st Qu.:1962   1st Qu.:57.50   1st Qu.:42.50
##  Mode  :character   Median :1965   Median :61.00   Median :44.00
##                     Mean   :1964   Mean   :59.61   Mean   :43.48
##                     3rd Qu.:1966   3rd Qu.:65.00   3rd Qu.:46.00
##                     Max.   :1967   Max.   :69.00   Max.   :51.00
##       m50             m75              f0             f25
##  Min.   :13.00   Min.   : 5.000   Min.   :38.00   Min.   :32.00
##  1st Qu.:21.50   1st Qu.: 7.000   1st Qu.:62.00   1st Qu.:46.00
##  Median :23.00   Median : 8.000   Median :66.00   Median :49.00
##  Mean   :22.84   Mean   : 8.387   Mean   :64.19   Mean   :47.52
##  3rd Qu.:24.00   3rd Qu.: 9.000   3rd Qu.:68.00   3rd Qu.:51.00
##  Max.   :30.00   Max.   :14.000   Max.   :75.00   Max.   :54.00
##      f50              f75
##  Length:31          Min.   : 6.00
##  Class :character   1st Qu.: 8.50
##  Mode  :character   Median :10.00
##                     Mean   :10.13
##                     3rd Qu.:11.00
##                     Max.   :19.00
```

2.1.4.2 2c-2 Cleaning the data: Madagascar, Cameroon, Trinidad, US

- Even without computing the mean and standard deviations for the variables,
  - we can notice that there is quite some variation regarding life expectancy
  - (please refer to the complete output on your screen as well).

A first observation, which is broadly documented,

- is that women have a longer remaining life expectancy
- than men, at all ages.

A country strikes in this list—in Madagascar,

- at the time of data collection,
  - women apparently did not have longer life expectancy
  - than men in their young and old years.

Further, the mean life expectancy at birth

- was only 38
- for both women and men.

This is also the life expectancy of females in Cameroon at that time,

- whereas males were expected to live even a little less (34 years).

Looking at the table,

- we can notice that Trinidad and the US are entered several times,
- as data collection was carried out more than once.

We will therefore discard case 23 (the second entry for Trinidad),

- as well as both cases 24 and 27 (US, data collected in 1966 and 1967)
- because cases 25 and 26 are more specific,
- as they provide estimations for White and Nonwhite individuals.

Let's create a new dataset without these cases

- before we proceed with cluster analysis.

Best done using dplyr and tidyverse commands

2.1.4.3 2c-3 Scale the Data, and Add Some Attributes

- Here we will scale the data.

The importance of scaling data

- has been discussed in the first section of this chapter.

Question: Why do we need to scale this data?

- Answer: Variables that are on a large scale will have a much larger effect on distance between variables on a smaller cale. Data needs to be standardized.

We also add some attributes to the dataset,

- corresponding to the ratio of male life expectancy
  - to female life expectancy at all ages,
- as the difference between male and females
  - would be lost in data scaling (all means will be 0).

Use cbind for this.

When you run your code, you will notice an error.

It happens that attribute f50

- is composed of strings
  - instead of numeric values
- (type mode(life$f50) to check this).

This is a type of problem you might encounter

- when dealing with data you have not prepared yourself
  - (and sometimes even with your data).

The solution is obviously

- to convert the attribute to numeric values
- before being able to compute the ratios.

Use as.numeric, then cbind

We can now repeat our assignment to life.temp with a successful result,

- and scale the data frame (omitting rows 1 and 2:
  - name of country and year of data collection).

We first convert to a data frame

- to get rid of information about mean and standard deviation
- that is contained in the returned object; we then convert to a matrix again.

2.1.4.4 2c-4 Now lets do some External Validation of our Classification

- When examining the iris dataset,
  - we had the correct solution
    - regarding the number of clusters
    - and the classification of cases.

This is not the case here

- we can not tell before running the analyses
  - the number of groups in our data.

We will therefore rely on computational trickery to discover them;

- cluster analysis will be performed iteratively
  - and the clustering solutions will be compared
  - using several indexes for determining the ideal number of clusters.

More information about such indexes can be found in the paper

- Experiments for the number of clusters in k-means, by Chiang and Mirkin, 2007
- (it should also be in your readings).

Here we rely on NbClust() function

- from the NbClust package,
- which we install and load:

The NBClust method includes many methods. Use method = "complete", which uses the maximum distance between two points within in each cluster.

```
library(datasets)
library(NbClust)
library(factoextra)
```

```
## Loading required package: ggplot2
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
data(iris)
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

We simply call the NbClust() function

- specifying the data and clustering algorithm to be used.

By default, the function will perform clustering

- using the Euclidean distance
- and compute all available indexes.

The reader is advised to consult the package help documentation

- for more information about customization.

Question: This shows that how many clusters is the most appropriate solution.

- Answer:

Question: Explain how/what shows this, in the NbClust output.

- Answer:

2.1.4.5 Links