LAB 9

STAT 131a

November 6, 2019

Welcome to the lab 9! In this lab, we will learn how to perform PCA and implement multiple linear regressions in R.

A simple PCA example

This is a simple example for PCA from R Bloggers. The iris dataset is perhaps the best known dataset for classification. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant.

```
# Load data
data(iris)
head(iris, 3)
```

```
##   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
```

There are five variables in the dataset `iris`, where `Sepal.Length`, `Sepal.Width`, `Petal.Length` and `Petal.Width` are continuous and `Species` is categorical.

```
names(iris)
```

```
## [1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"
## [5] "Species"
```

Apply PCA to the four continuous variables

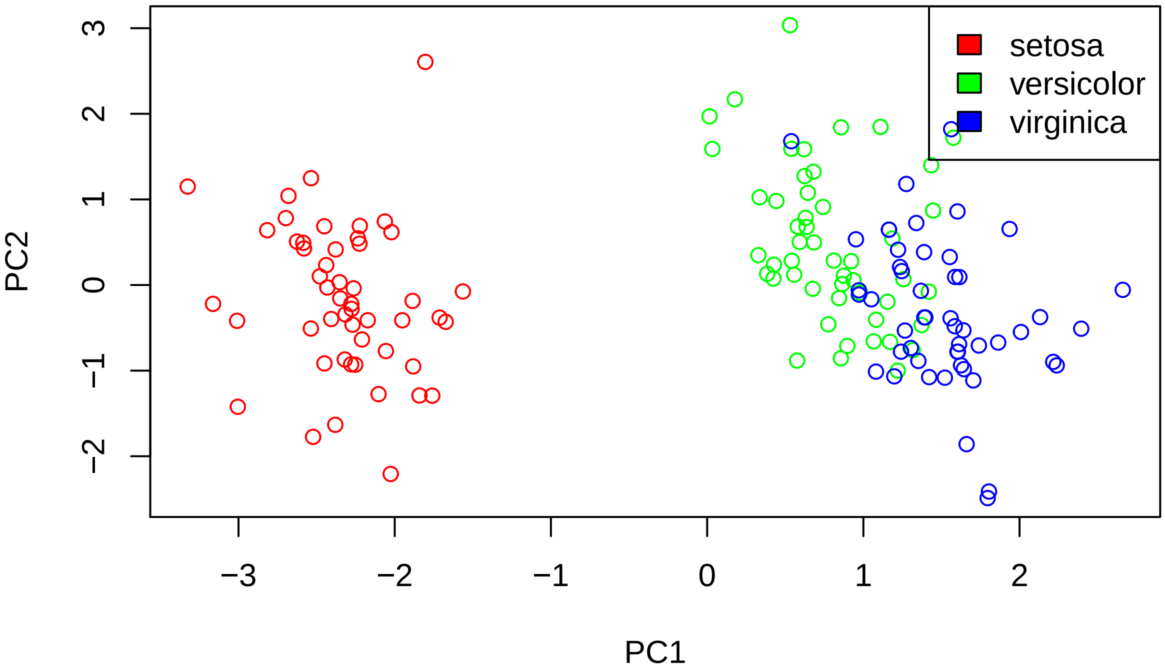
```
# log transform on the four continuous variable
log.ir <- log(iris[, 1:4])
# the Species variable
ir.species <- iris[, 5]

# apply PCA - scale. = TRUE is highly advisable, but default is FALSE.
# center and scale is used to standardize the variables prior to the application of PCA.
ir.pca <- prcomp(log.ir,
                 center = TRUE,
                 scale. = TRUE)
```

Plot the PC1 and PC2.

```
colorvec <- c("red", "green", "blue")
names(colorvec) = unique(ir.species)

plot(ir.pca$x[, 1], ir.pca$x[, 2], col = unname(colorvec[ir.species]), xlab = "PC1", ylab = "PC2")
legend("topright", legend = names(colorvec), fill = unname(colorvec))
```



Lets add a few NA values to the data.

```
iris[2,3]= NA
iris[5,2]= NA
iris[10,1] = NA
```

Apply PCA as before, but make change to code to address the missing values.

```
log.ir <- log(iris[, 1:4])
ir.species <- iris[, 5]

#ir.pca <- prcomp(log.ir,
#                  center = TRUE,
#                  scale. = TRUE)
```

Multiple Regression

Diamond Price

This is a very large data set showing various factors of over 50,000 diamonds including price, cut, color, clarity, etc. We are interested in the prediction of the `price` and how different factors influence the diamond price.

- **price**: price in US dollars ($326–$18,823)
- **carat**: weight of the diamond (0.2–5.01)
- **cut**: quality of the cut (Fair, Good, Very Good, Premium, Ideal)
- **color**: diamond colour, from J (worst) to D (best)
- **clarity**: a measurement of how clear the diamond is (I1 (worst), SI1, SI2, VS1, VS2, VVS1, VVS2, IF (best))
- **length.in.mm**: length in mm (0–10.74)
- **width.of.mm**: width in mm (0–58.9)
- **depth.in.mm**: depth in mm (0–31.8)
- **depth**: total depth percentage = z / mean(x, y) = 2 * z / (x + y) (43–79)

- **table**: width of top of diamond relative to widest point (43–95)

```
diamonds <- read.csv("diamonds.csv")
head(diamonds)
```

```
##   carat       cut color clarity depth table price length.in.mm width.of.mm
## 1  0.23     Ideal     E     SI2  61.5    55   326          3.95        3.98
## 2  0.21   Premium     E     SI1  59.8    61   326          3.89        3.84
## 3  0.23      Good     E     VS1  56.9    65   327          4.05        4.07
## 4  0.29   Premium     I     VS2  62.4    58   334          4.20        4.23
## 5  0.31      Good     J     SI2  63.3    58   335          4.34        4.35
## 6  0.24 Very Good     J    VVS2  62.8    57   336          3.94        3.96
##   depth.in.mm
## 1        2.43
## 2        2.31
## 3        2.31
## 4        2.63
## 5        2.75
## 6        2.48
```

Exercise 1: Exploratory Data Analysis before Regression

To understand the relationship between prices and carat, cut, etc. We first do the scatter plot. Create scatter plots between the response variable (price) and all the continuous variables. The function `is.numeric` might be helpful to check whether a variable is numeric. For example,

```
vec1 = 1:10
vec2 = as.character(1:10)
vec1
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```
vec2
```

```
##  [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10"
```

```
# The following line of code would return TRUE
is.numeric(vec1)
```

```
## [1] TRUE
```

```
# The following line of code would return FALSE
is.numeric(vec2)
```

```
## [1] FALSE
```

```
# The following line of code would return TRUE
is.character(vec2)
```

```
## [1] TRUE
```

The function `which` can help you to locate the column indices of the numeric vectors. (In fact, function `which` is a super useful function in R.)

```
# function `which` give the TRUE indices of a logical object
which(c(TRUE, FALSE, TRUE, FALSE, TRUE))
```

```
## [1] 1 3 5
```

```
# Insert your code here, use for loop to loop throught each numerical variable
```

Multiple Regression with Continuous Variable

Exercise 2 Fit the model and Calculate the Statistics

(a) Fit a linear model to price with all the continuous variable as explanatory variables. Print the summary of your model.

```
# Insert you code here, save your model as `fit`
# fit <-
```

(b) Calculate the fitted values.

```
# Insert you code here, save your results as `fitted.value`
# fitted.value <-
```

(c) Calculate the residual, the residual sum of squares (RSS) and the total sum of squares (TSS) using the `fitted.value` from the above chunk.

```
# Insert you code here, save your results as `RSS`
# residual <-
# RSS <-
# RSS
# TSS <-
# TSS
```

(d) Calculate the R-square (R^2) using RSS and TSS. How will you interpret the R^2?

```
# Insert you code here, save your results as `Rsq`
# Rsq <-
```

Exercise 3 Think deeper. Is the model resonable?

(a) Using your fitted model, we can write down the model formula: (the estimated coefficients can be found in the summary chart)

$$earnings = 20849.316+10686.309\cdot carat-203.154\cdot depth-102.446\cdot table-1315.668\cdot length.in.mm+66.322\cdot width.of.mm+41.62$$

How do we interpret this equation? By looking at the p-value, we know that the coefficients we estimated are significant except for `depth.in.mm`. Take `length.in.mm` for example, the coefficient - 1315.668 tells us that if we increase `length.in.mm` by 1, the price of the diamond will decrease 1315.668 dollars in average. How weird is that! Will you consider drop `length.in.mm`, `width.of.mm` and `depth.in.mm` in your model? Why? (HINT: Check the correlation between `length.in.mm`, `width.of.mm`, `depth.in.mm` and `carat`.)

```
# insert your code here
```

(b) Plot the variables versus the residual and calculate their correlation? Can you find any problem in your model by looking at the scatter plots? If you're asked to add some terms to improve the model, what will you do? (HINT: Consider the scatter plot in Exercise 1: are the relationships linear?)

```
# Insert your code here, use for loop to loop thrugth each numerical variable
```

Multiple Regression with Continuous and Categorical Variable

Exercise 4

(a) Fit a linear regression model with explanatory variable `carat`, `depth`, `table`, `clarity`, `color` and `cut`.

```r
levels(diamonds$clarity)
```

```
## [1] "I1"   "IF"   "SI1"  "SI2"  "VS1"  "VS2"  "VVS1" "VVS2"
```

```r
levels(diamonds$color)
```

```
## [1] "D" "E" "F" "G" "H" "I" "J"
```

```r
levels(diamonds$cut)
```

```
## [1] "Fair"      "Good"      "Ideal"     "Premium"   "Very Good"
```

```r
# Insert you code here, save your model as `fit.categorical`
# fit.categorical <-
```

(b) Write the equation when

I. clarity is VS2, color is H and cut is Premium.

```
# repalce ??? by numerical values
price = ??? + ??? * carat + ??? * depth + ??? * table
```

II. clarity is I1, color is D and cut is Fair.

```
# repalce ??? by numerical values
price = ??? + ??? * carat + ??? * depth + ??? * table
```