STAT 432 Homework-1

Sharvi Tomar (stomar2)

27/08/21

Contents

Question 1 (random number generation and basic statistics)

X1, X2, ..., Xn are i.i.d. $\mathcal{N}(\mu,\sigma^2)$ random variables, where μ=3 and σ=2.

a. Generate a set of `n=100` observations from this distribution. Only display the first 10 observations in your R output. Make sure that you set seed properly in order to replicate the result.

b. What is the statistical formula of the sample mean and sample variance (unbiased estimation)? Type the answer using latex.

c. Calculate the above quantities (in b and c) using R functions. You need to use your own code to calculate these quantities and then match the results with default R functions.

d. Write a new function called mysummarystat that takes the data vector as the input, and output an vector of two elements: the sample mean and variance. Call the function using your data to validate.

Answer:

```
# Generating normally distributed random variable
set.seed(1)
x=rnorm(100, mean=3, sd=2)
x[1:10]
```

```
##  [1] 1.747092 3.367287 1.328743 6.190562 3.659016 1.359063 3.974858 4.476649
##  [9] 4.151563 2.389223
```

Sample mean $\overline{x}$ =

$$\frac{\sum_{i=1}^{n} x_i^2}{n}$$

Sample variance s^2 =

$$\frac{\sum_{i=1}^{n}(x_i-\overline{x})^2}{(n-1)}$$

```r
# Function to calculate sample mean
mymean <- function(y)
  {
    s = 0
    for(el in y)
      s = s + el
    return(s = s / length(y))
  }
```

```r
# Function to calculate sample variance
myvar <- function (y)
 {
    v = 0
    m = mymean(y)
    for(el in y)
      v = v + ((el - m)^2)
    return(v = v / (length(y)-1))
}
```

```r
# Difference of calculated mean with R mean
mymean(x)-mean(x)
```

```
## [1] 0
```

```r
# Difference of calculated variance with R variance
myvar(x)-var(x)
```

```
## [1] -4.440892e-16
```

```r
# Defining 'mysummarystat' function
mysummarystat<-function(y){
  return(c(mean(y),var(y)))
}

# Calling 'mysummarystat' function using x (normally distributed random variable)
mysummarystat(x)
```

```
## [1] 3.217775 3.227048
```

Question 2 (data manipulation, plots and linear model)

Perform the following tasks on the iris dataset. For each question, output necessary information to check that your completed the required operation.

a. Change the class labels of the Species variable from virginica, versicolor, and setosa to Species_1, Species_2 and Species_3, respectively.

b. Change the variable name from Species to Type. Note that for both questions a) and b), you need to change the original variable, not creating a new variable and replacing the old one.

c. Create a boxplot for the variable Petal.Length that shows different boxes for different levels of Type. Adjust chunk options so that the plot is at the center and occupies 60% of the page width.

d. Use a linear model to estimate Petal.Length using all other four covariates. Make sure that the Type variable is specified as a factor. Report the coefficients and the most significant variable. To obtain the most significant variable, you must extract the p-value from the fitted object, instead of reading the value from the R output on your screen. If you do not know how to extract the p-value, use google to search for an answer with relevant keywords. Cite your reference by providing a link to it.

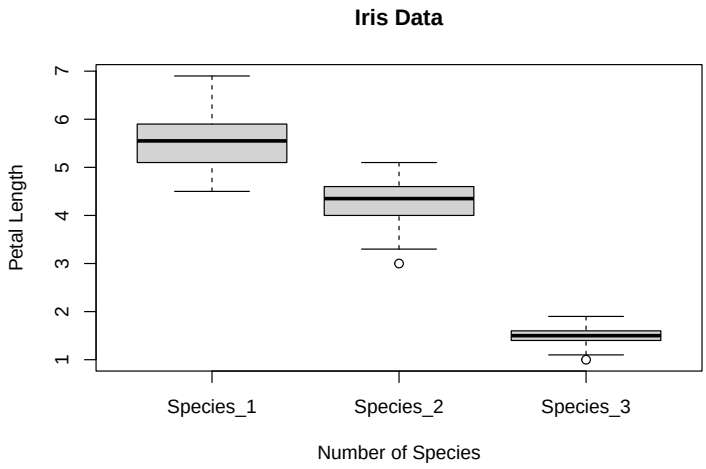
e. Save the iris data into a .csv file, and then read the data from that file back into R. Make sure that the values in this new data is the same to the original one.

Answer:

```
# Changing the class labels of the Species variable
levels(iris$Species) <- list(Species_1  = "virginica", Species_2 = "versicolor", Species_3 = "setosa")
```

```
# Changing the variable name from Species to Type
colnames(iris)[5]<-"Type"
```

```
# Creating a boxplot for the variable Petal.Length for different levels of Type
boxplot(Petal.Length~Type,data=iris, main="Iris Data",
        xlab="Number of Species", ylab="Petal Length")
```




```
# Checking data type of Type variable
str(iris$Type)
```

```
##  Factor w/ 3 levels "Species_1","Species_2",..: 3 3 3 3 3 3 3 3 3 3 ...
```

```
# Creating a linear model to estimate Petal.Length using all other four covariates.
model= lm(Petal.Length~Sepal.Length+Sepal.Width+Petal.Width+Type, data = iris)
```

```r
# Reporting coefficients
model$coefficients
```

```
##   (Intercept)  Sepal.Length   Sepal.Width   Petal.Width TypeSpecies_2
##     0.8632341     0.6080058    -0.1805236     0.6022215    -0.5108520
## TypeSpecies_3
##    -1.9742229
```

```r
# Generating linear model summary to see statistically significant variables
summary(model)
```

```
##
## Call:
## lm(formula = Petal.Length ~ Sepal.Length + Sepal.Width + Petal.Width +
##     Type, data = iris)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.78396 -0.15708  0.00193  0.14730  0.65418
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.86323    0.30527   2.828  0.00536 **
## Sepal.Length   0.60801    0.05024  12.101  < 2e-16 ***
## Sepal.Width   -0.18052    0.08036  -2.246  0.02619 *
## Petal.Width    0.60222    0.12144   4.959 1.97e-06 ***
## TypeSpecies_2 -0.51085    0.09500  -5.377 2.98e-07 ***
## TypeSpecies_3 -1.97422    0.24480  -8.065 2.60e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2627 on 144 degrees of freedom
## Multiple R-squared:  0.9786, Adjusted R-squared:  0.9778
## F-statistic:  1317 on 5 and 144 DF,  p-value: < 2.2e-16
```

The statistically significant variables (with highest significance codes) as per the model are: "Sepal.Length","Petal.Width","TypeSpecies_2","TypeSpecies_3".

```r
# Reporting p-values of variates
summary(model)$coefficients[,4]
```

```
##   (Intercept)  Sepal.Length   Sepal.Width   Petal.Width TypeSpecies_2
##  5.355202e-03  1.073592e-23  2.619373e-02  1.968679e-06  2.984973e-07
## TypeSpecies_3
##  2.600021e-13
```

```r
# Variable with smallest p-value
pvals=summary(model)$coefficients[,4]
print(min(pvals))
```

```
## [1] 1.073592e-23
```

The most significant variable is the one with the smallest p-value which is "Sepal.Length".

```
# Saving the iris data into a .csv file
write.csv(iris, file = "mydata.csv")
# Reading the data from that file back into R
data=read.csv("mydata.csv",stringsAsFactors=TRUE)
# Checking values in new data same as the original one
head(data)
```

```
##   X Sepal.Length Sepal.Width Petal.Length Petal.Width      Type
## 1 1          5.1         3.5          1.4         0.2 Species_3
## 2 2          4.9         3.0          1.4         0.2 Species_3
## 3 3          4.7         3.2          1.3         0.2 Species_3
## 4 4          4.6         3.1          1.5         0.2 Species_3
## 5 5          5.0         3.6          1.4         0.2 Species_3
## 6 6          5.4         3.9          1.7         0.4 Species_3
```