assignment_2_solution

Sahir Khan

August 11, 2024

```r
# Q1a.Chest of coins with 20 gold, 30 silver and 50 bronze coins.
# You randomly draw 10 coins from this chest.
chest <- c(rep("G",times=20),rep("S",times=30),rep("B",times=50))
q1a<-sample(chest,10,replace=T)
print(q1a)
```

```
##  [1] "B" "S" "S" "B" "B" "G" "G" "B" "B" "B"
```

```r
# Q1b.In a surgical procedure, the chances of success and failure are 90% and 10% respectively.
# Generate a sample space for the next 10 surgical procedures performed.
surgery <- c(T,F)
q1b<-sample(surgery,10,replace=T,prob = c(0.9,0.1))
print(q1b)
```

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

```r
# Q2.A room has n people, and each has an equal chance of being born on any of the 365
# days of the year. (For simplicity, we'll ignore leap years). What is the probability
# that two people in the room have the same birthday?
N<-23 #No.of people in room
probability1 <- 1-(choose(365,N)*factorial(N))/((365)^N)
iterations<-1000 #no.of simulations
sum=0
for(val in 1:iterations){
  birthdays <- sample(365, N, replace = TRUE)
  sum <- sum + as.integer(any(duplicated(birthdays)))
}
probability_simulated <- sum/iterations
print(paste("Probability that two people in the room have the same birthday = ",probability_simulated))
```

```
## [1] "Probability that two people in the room have the same birthday =  0.508"
```

```r
# Q2.A room has n people, and each has an equal chance of being born on any of the 365
# days of the year. (For simplicity, we'll ignore leap years). What is the probability
# that two people in the room have the same birthday?
# Function to simulate the birthday problem
birthday_simulation <- function(n, trials = 10000) {
  same_birthday <- 0  # Counter for simulations where two people have the same birthday
```

```r
  for (i in 1:trials) {
    birthdays <- sample(1:365, n, replace = TRUE)  # Generate n random birthdays
    if (length(unique(birthdays)) != n) {  # Check if there are duplicates
      same_birthday <- same_birthday + 1
    }
  }

  # Estimate the probability
  probability <- same_birthday / trials
  return(probability)
}

# Simulate for different values of n
n_values <- c(5, 10, 20, 30, 50, 100)
probabilities <- sapply(n_values, birthday_simulation)

# Display the results
results <- data.frame(n = n_values, Probability = probabilities)
print(results)
```

```
##     n Probability
## 1   5      0.0267
## 2  10      0.1161
## 3  20      0.4010
## 4  30      0.6999
## 5  50      0.9714
## 6 100      1.0000
```

```r
# Q3.suppose the probability of the weather being cloudy is 40%. Also suppose the probability
# of rain on a given day is 20% and that the probability of clouds on a rainy day is 85%.
# If it's cloudy outside on a given day, what is the probability that it will rain that day?
bayesTheorem <- function(pA,pB,pBA){
  pAB <- pBA*pA/pB
  return(pAB)
}
pCloud <- 0.4
pRain <- 0.2
pCloudyRain <- 0.85
pRainCloud <- bayesTheorem(pRain, pCloud, pCloudyRain)
print(paste("Probabilty of rain given cloudy day",pRainCloud))
```

```
## [1] "Probabilty of rain given cloudy day 0.425"
```

```r
# Q4.The iris dataset is a built-in dataset in R that contains measurements on 4 different
# attributes (in centimeters) for 150 flowers from 3 different species.
dat <- iris
# Print first few rows of this dataset
print(head(dat))
```

```
##   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
```

```r
# Find the structure of this dataset
print(str(dat))
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
## NULL
```

```r
# Find the range of the data regarding the sepal length of flowers.
print(range(dat$Sepal.Length))
```

```
## [1] 4.3 7.9
```

```r
# Find the mean of the sepal length.
print(mean(dat$Sepal.Length))
```

```
## [1] 5.843333
```

```r
# Find the median of the sepal length.
print(median(dat$Sepal.Length))
```

```
## [1] 5.8
```

```r
# Find the first and the third quartiles and hence the interquartile range.
print(quantile(dat$Sepal.Length, 0.25))
```

```
## 25%
## 5.1
```

```r
print(quantile(dat$Sepal.Length, 0.75))
```

```
## 75%
## 6.4
```

```r
print(IQR(dat$Sepal.Length))
```

```
## [1] 1.3
```

```r
# Find the standard deviation and variance.
print(lapply(dat[, 1:4], sd))
```

```
## $Sepal.Length
## [1] 0.8280661
##
## $Sepal.Width
## [1] 0.4358663
##
## $Petal.Length
## [1] 1.765298
##
## $Petal.Width
## [1] 0.7622377
```

```r
# Try doing the above exercises for sepal.width, petal.length and petal.width.
summary(iris)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
##        Species
##  setosa    :50
##  versicolor:50
##  virginica :50
##
##
##
```

```r
# Q5.So we create a user function to calculate mode of a data set in R.
# This function takes the vector as input and gives the mode value as output.
mode <- function(v){
  u<-unique(v)
  u[which.max(tabulate(match(v,u)))]
}
v<-c(2,1,2,3,1,2,3,4,1,5,5,3,2)
m<-mode(v)
print(paste("Mode=",m))
```

```
## [1] "Mode= 2"
```