

Probability and Statistics (UCS410)

Experiment 2: Descriptive statistics, Sample space, definition of probability

Name: Jasween Kaur Brar

Roll no: 102017187

Sub-group: 3CS8

Q1. (a) Suppose there is a chest of coins with 20 gold, 30 silver and 50 bronze coins. You randomly draw 10 coins from this chest. Write an R code that will give us the sample space for this experiment. (Use of sample(): an in-built function in R)

(b) 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. (Use of prob(): an in-built function in R)

Ans.

(a) Code:

```
a = rep('Gold',20)
b = rep('Silver',30)
c = rep('Bronze',50)
temp = c(a,b,c)
sample(temp,size = 10)
```

Output :

```
> temp = c(a,b,c)
> sample(temp,size = 10)
[1] "Bronze" "Bronze" "Bronze" "Silver" "Gold"   "Silver"
[7] "silver" "Bronze" "Bronze" "Silver"
> |
```

(b) Code :

```
temp = c("S","F")
sample(temp,size = 10,replace=TRUE,prob=c(0.9,0.1))
```

Output :

```
> temp = c("S","F")
> sample(temp,size = 10,replace=TRUE,prob=c(0.9,0.1))
[1] "S" "S" "S" "S" "S" "S" "S" "F" "S" "F"
```

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?

(a) Use an R simulation to estimate this for various n .

(b) Find the smallest value of n for which the probability of a match is greater than .5.

Ans. (a) Code :

```
n = 20
probability = 1
for(i in 1:n)
{
  probability = probability*(365-i)/365
}
probability=1-probability
print("Probability that two people have birthday on the same day :-")
print(probability)
```

Output :

```
[1] "Probability that two people have birthday on the same day :-"
> print(probability)
[1] 0.4436883
>
```

b) Code :

```
count <- 0
probability <- 1 - probability
y <- 1
while(y >= probability)
{
  y <- y * ((365-count)/365)
  count <- count + 1
}
print('Smallest value of n for which probability is greater than 0.5')
print(count)
```

Output :

```
> print('Smallest value of n for which probability is greater than 0.5')
[1] "Smallest value of n for which probability is greater than 0.5"
> print(count)
[1] 21
```

Q3. Write an R function for computing conditional probability. Call this function to do the following problem: 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?

Ans. Code :

```
cloudy <- 0.4
rain <- 0.2
intersection <- 0.85
probability <- intersection*rain/cloudy
print('The probability that it will rain the day it's cloudy outside :- ')
print(probability)
```

Output :

```
[1] "The probability that it will rain\nthe day it's cloudy outside :-  
"  
> print(probability)  
[1] 0.425  
> |
```

Q4. The iris dataset is a built-in dataset in R that contains measurements on 4 different attributes (in centimetres) for 150 flowers from 3 different species. Load this dataset and do the following:

- (a) Print the first few rows of this dataset.
- (b) Find the structure of this dataset.
- (c) Find the range of the data regarding the sepal length of flowers.
- (d) Find the mean of the sepal length.
- (e) Find the median of the sepal length.
- (f) Find the first and the third quartiles and hence the interquartile range.
- (g) Find the standard deviation and variance.
- (h) Try doing the above exercises for sepal.width, petal.length and petal.width.
- (i) Use the built-in function summary on the dataset Iris.

Ans.

```
data(iris)
```

(a)

```
head(iris)
```

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

(b)

```
str(iris)
```

```
> str(iris)
'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 ...
> |
```

(c)

```
range(iris$Sepal.Length)
```

```
> range(iris$Sepal.Length)
[1] 4.3 7.9
> |
```

(d)

```
mean(iris$Sepal.Length)
```

```
> mean(iris$Sepal.Length)
[1] 5.843333
|
```

(e)

```
median(iris$Sepal.Length)
```

```
> median(iris$Sepal.Length)
[1] 5.8
>
```

(f)

```
quantile(iris$Sepal.Length, c(0.25,0.75))  
quartile = quantile(iris$Sepal.Length, c(0.75)) - quantile(iris$Sepal.Length, c(0.25))  
print('Interquartile Range :- ')  
print(quartile)
```

```
> quantile(iris$Sepal.Length, c(0.25,0.75))  
25% 75%  
5.1 6.4  
> quartile = quantile(iris$Sepal.Length, c(0.75)) - quantile(iris$Sepal.Length, c(0.25))  
> print('Interquartile Range :- ')  
[1] "Interquartile Range :- "  
> print(quartile)  
75%  
1.3
```

(g)

```
sd(iris$Sepal.Length)
```

```
> sd(iris$Sepal.Length)  
[1] 0.8280661
```

(h)

```
var(iris$Sepal.Length)
```

```
> var(iris$Sepal.Length)  
[1] 0.6856935
```

```
range(iris$Petal.Length)
```

```
> range(iris$Petal.Length)  
[1] 1.0 6.9
```

```
range(iris$Petal.Width)
```

```
> range(iris$Petal.width)
[1] 0.1 2.5
```

(i)

```
summary(iris)
```

```
> 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
```

Q5. R does not have a standard in-built function to calculate mode. 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.

Ans.

```
install.packages("DescTools")
library("DescTools")

x1<- c(1,1,1,2,3,4,5,5,5,5)

Mode(x1)
```

```
> library("DescTools")
> x1<- c(1,1,1,2,3,4,5,5,5,5)
> Mode(x1)
[1] 5
attr(,"freq")
[1] 4
> |
```

```
x1 <- c(11, 18, 11, 19, 21, 19, 46, 21, 19)
mode <- function(v)
{
  uniq <- unique(v)
  uniq[which.max(tabulate(match(v, uniq)))]
}
result <- mode(x1)
print("Mode is :- ")
print(result)
```

```
> result <- mode(x1)
> print("Mode is :- ")
[1] "Mode is :- "
> print(result)
[1] 19
> |
```