

# MAT 2001

## Statistics



Embedded Lab e-Record

### Assessment – 1

L1+L2

WINTER SEMESTER 2019–20

#### Exp 1: Measures of Central Tendency

January 6, 2020

submitted by

SHARADINDU ADHIKARI

19BCE2105

## Question 1

### Problem:

From the following data of the marks obtained by 60 students of a class, calculate the arithmetic mean and median of the distribution.

Marks	20	30	40	50	60	70
No. of students	8	12	20	10	6	4

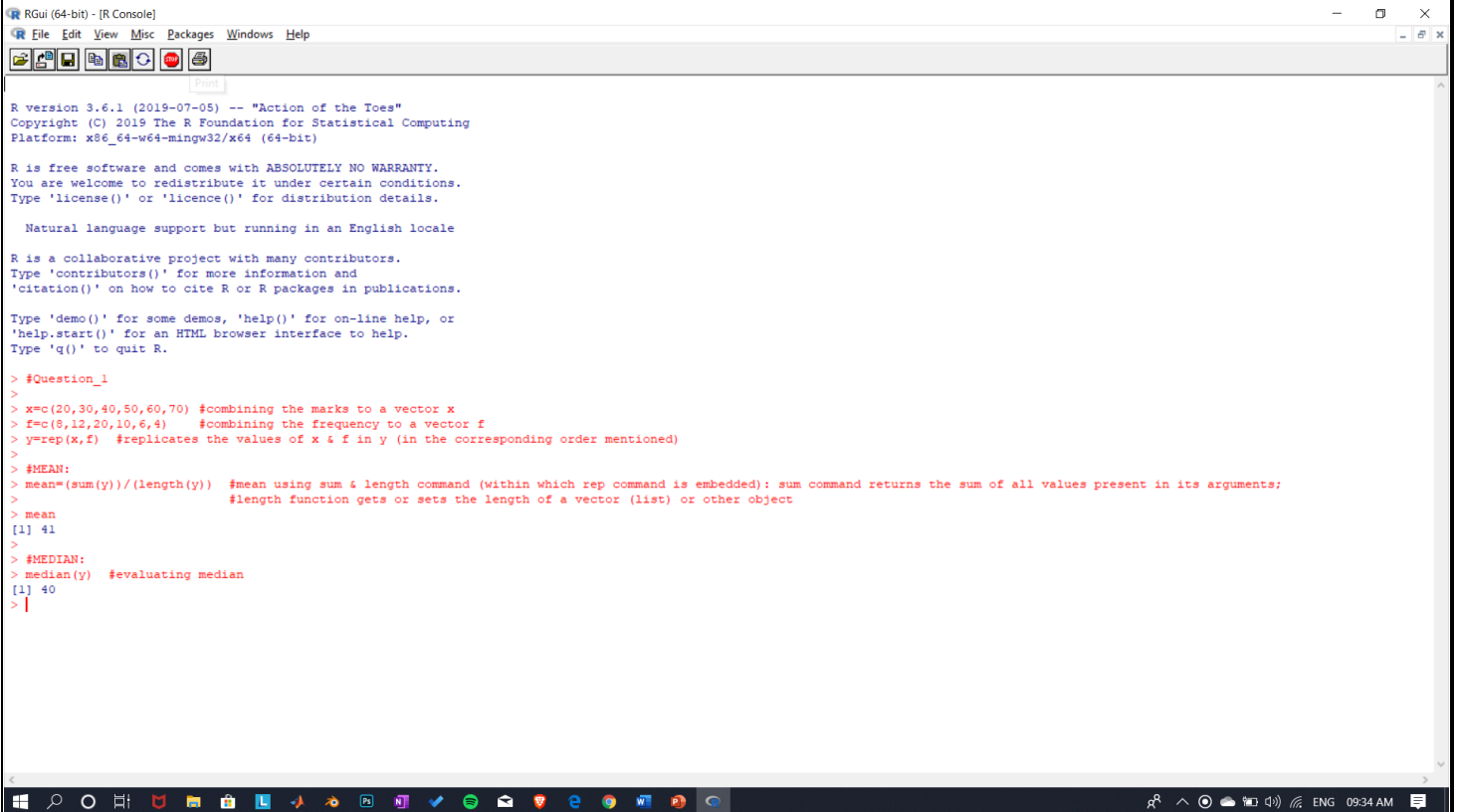
### Code & Input in R console:

```
> #Question_1
>
> x=c(20,30,40,50,60,70) #combining the marks to a vector x
> f=c(8,12,20,10,6,4)    #combining the frequency to a vector f
> y=rep(x,f) #replicates the values of x & f in y (in the
corresponding order mentioned)
>
> #MEAN:
> mean=(sum(y))/(length(y)) #mean using sum & length command
(within which rep command is embedded inside y): sum command
returns the sum of all values present in its arguments; length
function gets or sets the length of a vector (list) or other object
> mean
[1] 41
>
> #MEDIAN:
> median(y) #evaluating median
[1] 40
```

### Output:

```
> mean: 41
> median: 40
```

## Screenshot of Code:



```
RGui (64-bit) - [R Console]
File Edit View Misc Packages Windows Help

R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> #Question_1
>
> x=c(20,30,40,50,60,70) #combining the marks to a vector x
> f=c(8,12,20,10,6,4)    #combining the frequency to a vector f
> y=rep(x,f) #replicates the values of x & f in y (in the corresponding order mentioned)
>
> #MEAN:
> mean=(sum(y))/(length(y)) #mean using sum & length command (within which rep command is embedded): sum command returns the sum of all values present in its arguments;
>                                #length function gets or sets the length of a vector (list) or other object
> mean
[1] 41
>
> #MEDIAN:
> median(y) #evaluating median
[1] 40
> |
```

## Question 2

### Problem:

From the following data of weight of 122 persons, determine the mean, median & modal weight.

Weight (in lbs)	100-110	110-120	120-130	130-140	140-150	150-160	160-170	170-180
No. of persons	4	6	20	32	33	17	8	2

### Code & Input in R console:

```
> #Question_2
>
> mid=seq(105,175,10)      #mid function compute a vector of
midpoints between values in a numeric vector;
>                          #seq command represents a set of related
numbers, events, movements, or items that follow each other in a
particular order
> mid #printing the mid function
[1] 105 115 125 135 145 155 165 175
> f=c(4,6,20,32,33,17,8,2) #combining the frequency to a vector f
> fr.distr=data.frame(mid,f) #command to make frequency
distribution table
> fr.distr #printing the table
  mid  f
1 105  4
2 115  6
3 125 20
4 135 32
5 145 33
6 155 17
7 165  8
8 175  2
>
> #MEAN:
> mean=(sum(mid*f))/sum(f) #applying the formula of mean using
mid function
> mean #printing the mean value
[1] 139.5082
>
```

```

> #MEDIAN:
> cl=cumsum(f) #command to calculate cumulative frequency of given
data
> cl #printing the same
[1] 4 10 30 62 95 112 120 122
> n=sum(f) #to find the sum of frequency
> n #printing the sum
[1] 122
> ml=min(which(cl>=n/2)) #serial number of the median class
> ml
[1] 4
> h=10 #class length of given data
> h
[1] 10
> freq=f[ml] #frequency of the median class
> freq
[1] 32
> c=cl[ml-1] #cumulative frequency of the median class
> c
[1] 30
> l=mid[ml]-h/2 #lower class boundary
> l
[1] 130
> median=l+(((n/2)-c)/freq)*h #formula of median
> median #printing its value
[1] 139.6875
>
> #MODE:
> mx=which((f==max(f))) #serial number of modal class
> mx
[1] 5
> fm=f[mx] #frequency of the modal class
> fm
[1] 33
> f1=f[mx-1] #frequency of the pre modal class
> f1
[1] 32
> f2=f[mx+1] #frequency of the post modal class
> f2
[1] 17
> l=mid[mx]-h/2 #lower class boundary
> l
[1] 140
> mode=l+((fm-f1)/(2*fm-f1-f2))*h #formula of mode
> mode #printing its value
[1] 140.5882

```

### Output:

```

> mean: 139.5082
> median: 139.6875
> mode: 140.5882

```

## Screenshot of Code:

```
RGui (64-bit) - [R Console]
File Edit View Misc Packages Windows Help
[Icons] [Stop current computation]

> #Question_2
>
> mid=seq(105,175,10) #mid function compute a vector of midpoints between values in a numeric vector;
> #seq command represents a set of related numbers, events, movements, or items that follow each other in a particular order
> mid #printing the mid function
[1] 105 115 125 135 145 155 165 175
> f=c(4,6,20,32,33,17,8,2) #combining the frequency to a vector f
> fr.distr=data.frame(mid,f) #command to make frequency distribution table
> fr.distr #printing the table
  mid f
1 105 4
2 115 6
3 125 20
4 135 32
5 145 33
6 155 17
7 165 8
8 175 2
>
> #MEAN:
> mean=(sum(mid*f))/sum(f) #applying the formula of mean using mid function
> mean #printing the mean value
[1] 139.5082
>
> #MEDIAN:
> cl=cumsum(f) #command to calculate cumulative frequency of given data
> cl #printing the same
[1] 4 10 30 62 95 112 120 122
> n=sum(f) #to find the sum of frequency
> n #printing the sum
[1] 122
> ml=min(which(cl>=n/2)) #serial number of the median class
> ml
[1] 4
> h=10 #class length of given data
> h
[1] 10
> freq=f[ml] #frequency of the median class
> freq
[1] 32
> c=c[ml-1] #cumulative frequency of the median class
> c
[1] 30
> l=mid[ml]-h/2 #lower class boundary
> l
[1] 130
> median=l+(((n/2)-c)/freq)*h #formula of median
> median #printing its value
[1] 139.6875
>
> #MODE:
> mx=which(f==max(f)) #serial number of modal class
> mx
[1] 5
> fm=f[mx] #frequency of the modal class
> fm
[1] 33
> fl=f[m-1] #frequency of the pre modal class
Error: object 'm' not found
> fl=f[mx-1] #frequency of the pre modal class
> fl
[1] 32
> f2=f[mx+1] #frequency of the post modal class
> f2
[1] 17
> l=mid[mx]-h/2 #lower class boundary
> l
[1] 140
> mode=l+((fm-fl)/(2*fm-fl-f2))*h #formula of mode
> mode #printing its value
[1] 140.5882
> |
```

### Question 3

#### Problem:

From the following data of wages (in Rupees) of workers, compute:

23460, 34850, 43956, 34535, 23784, 23785, 23456, 43569, 23758, 34535, 43864, 23679, 34535, 43762, 34873, 23764, 23784, 86345, 34850, 47645

- (a) Find the mean and median wage.
- (b) Find modal wage of all workers.
- (c) Find median wage of workers under 25000 and above 35000.

#### Frequency distribution table:

Wages	No. of workers
23460	1
34850	2
43956	1
34535	3
23784	2
23785	1
23456	1
43569	1
23758	1
43864	1
23679	1
43762	1
34873	1
23764	1
86345	1
47645	1

### Code & Input in R console:

```
> #Question_3
>
> #Part(a):
>
> #MEAN:
>
x=c(23460,34850,43956,34535,23784,23785,23456,43569,23758,43864,2
3679,43762,34873,23764,86345,47645) #combining the wages to a
vector x
> f=c(1,2,1,3,2,1,1,1,1,1,1,1,1,1,1) #combining the frequency
to a vector f
> y=rep(x,f) #replicates the values of x & f in y (in the
corresponding order mentioned)
> mean=(sum(y))/(length(y)) #formula of mean
> mean #printing its value
[1] 35339.45
>
> #MEDIAN:
> median(y) #printing its value
[1] 34535
>
> #Part(b):
>
> #MODE:
> z=c(23460, 34850, 43956, 34535, 23784, 23785, 23456, 43569,
23758, 34535, 43864, 23679, 34535, 43762,34873, 23764, 23784,
86345, 34850, 47645) #combining the wages to a vector z
> xr=table(z) #tabulating the wages of vector z
> mode=which(xr==max(xr)) #formula of mode
> mode #printing its value
34535
8
>
> #Part(c):
>
> #MEDIAN under 25000:
> a=z[z<25000] #setting the limit till which we're calculating
the median
> ma=median(a) #calculating the median
> ma #printing its value
[1] 23761
>
> #MEDIAN above 35000:
> b=z[z>35000] #limiting the range of median calculation
> mb=median(b)
> mb #printing median value
[1] 43910
```



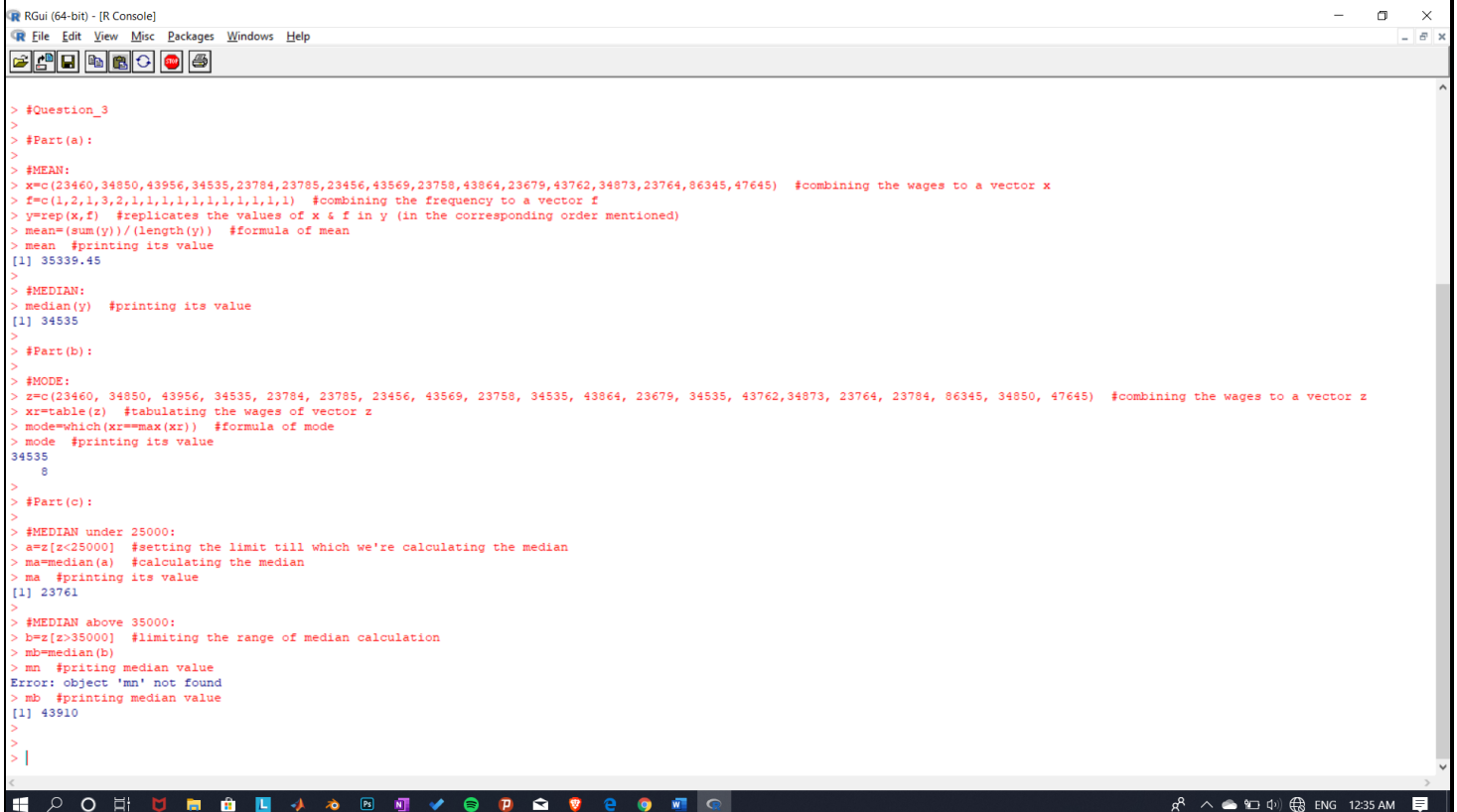
## Output:

```
(a) > mean: 35339.45
      > median: 34535

(b) > mode: 34535
      8

(c) > median under 25000: 23761
      > median above 35000: 43910
```

## Screenshot of Code:



```
RGui (64-bit) - [R Console]
File Edit View Misc Packages Windows Help

> #Question_3
>
> #Part(a):
>
> #MEAN:
> x=c(23460,34850,43956,34535,23784,23785,23456,43569,23758,43864,23679,43762,34873,23764,86345,47645) #combining the wages to a vector x
> f=c(1,2,1,3,2,1,1,1,1,1,1,1,1,1,1,1) #combining the frequency to a vector f
> y=rep(x,f) #replicates the values of x & f in y (in the corresponding order mentioned)
> mean=(sum(y))/(length(y)) #formula of mean
> mean #printing its value
[1] 35339.45
>
> #MEDIAN:
> median(y) #printing its value
[1] 34535
>
> #Part(b):
>
> #MODE:
> z=c(23460, 34850, 43956, 34535, 23784, 23785, 23456, 43569, 23758, 34535, 43864, 23679, 34535, 43762,34873, 23764, 23784, 86345, 34850, 47645) #combining the wages to a vector z
> xr=table(z) #tabulating the wages of vector z
> mode=which(xr==max(xr)) #formula of mode
> mode #printing its value
34535
8
>
> #Part(c):
>
> #MEDIAN under 25000:
> a=z[z<25000] #setting the limit till which we're calculating the median
> ma=median(a) #calculating the median
> ma #printing its value
[1] 23761
>
> #MEDIAN above 35000:
> b=z[z>35000] #limiting the range of median calculation
> mb=median(b)
> mn #printing median value
Error: object 'mn' not found
> mb #printing median value
[1] 43910
>
> |
```

**End**