Sanjay Chauhan - 919 **M.Sc. Statistics – IV (2024-25)**

Practical – 1

Statistical Quality Control-I (Control Chart on Variables)

Q.1) Draw $\bar{x}$, R and S charts for the diameter in the dataset pistonrings for the first 25 samples. Explain your findings.

```
> library(qcc)

> data(package="qcc")

> data("pistonrings")

> head(pistonrings)

  diameter sample trial
1   74.030      1  TRUE
2   74.002      1  TRUE
3   74.019      1  TRUE
4   73.992      1  TRUE
5   74.008      1  TRUE
6   73.995      2  TRUE

> pistonrings

    diameter sample trial
1     74.030      1  TRUE
2     74.002      1  TRUE
3     74.019      1  TRUE
4     73.992      1  TRUE
5     74.008      1  TRUE
6     73.995      2  TRUE
7     73.992      2  TRUE
8     74.001      2  TRUE
9     74.011      2  TRUE
10    74.004      2  TRUE
11    73.988      3  TRUE
12    74.024      3  TRUE
```

13	74.021	3	TRUE
14	74.005	3	TRUE
15	74.002	3	TRUE
16	74.002	4	TRUE
17	73.996	4	TRUE
18	73.993	4	TRUE
19	74.015	4	TRUE
20	74.009	4	TRUE
21	73.992	5	TRUE
22	74.007	5	TRUE
23	74.015	5	TRUE
24	73.989	5	TRUE
25	74.014	5	TRUE
26	74.009	6	TRUE
27	73.994	6	TRUE
28	73.997	6	TRUE
29	73.985	6	TRUE
30	73.993	6	TRUE
31	73.995	7	TRUE
32	74.006	7	TRUE
33	73.994	7	TRUE
34	74.000	7	TRUE
35	74.005	7	TRUE
36	73.985	8	TRUE
37	74.003	8	TRUE
38	73.993	8	TRUE
39	74.015	8	TRUE
40	73.988	8	TRUE
41	74.008	9	TRUE
42	73.995	9	TRUE
43	74.009	9	TRUE

44	74.005	9	TRUE
45	74.004	9	TRUE
46	73.998	10	TRUE
47	74.000	10	TRUE
48	73.990	10	TRUE
49	74.007	10	TRUE
50	73.995	10	TRUE
51	73.994	11	TRUE
52	73.998	11	TRUE
53	73.994	11	TRUE
54	73.995	11	TRUE
55	73.990	11	TRUE
56	74.004	12	TRUE
57	74.000	12	TRUE
58	74.007	12	TRUE
59	74.000	12	TRUE
60	73.996	12	TRUE
61	73.983	13	TRUE
62	74.002	13	TRUE
63	73.998	13	TRUE
64	73.997	13	TRUE
65	74.012	13	TRUE
66	74.006	14	TRUE
67	73.967	14	TRUE
68	73.994	14	TRUE
69	74.000	14	TRUE
70	73.984	14	TRUE
71	74.012	15	TRUE
72	74.014	15	TRUE
73	73.998	15	TRUE
74	73.999	15	TRUE

75	74.007	15	TRUE
76	74.000	16	TRUE
77	73.984	16	TRUE
78	74.005	16	TRUE
79	73.998	16	TRUE
80	73.996	16	TRUE
81	73.994	17	TRUE
82	74.012	17	TRUE
83	73.986	17	TRUE
84	74.005	17	TRUE
85	74.007	17	TRUE
86	74.006	18	TRUE
87	74.010	18	TRUE
88	74.018	18	TRUE
89	74.003	18	TRUE
90	74.000	18	TRUE
91	73.984	19	TRUE
92	74.002	19	TRUE
93	74.003	19	TRUE
94	74.005	19	TRUE
95	73.997	19	TRUE
96	74.000	20	TRUE
97	74.010	20	TRUE
98	74.013	20	TRUE
99	74.020	20	TRUE
100	74.003	20	TRUE
101	73.988	21	TRUE
102	74.001	21	TRUE
103	74.009	21	TRUE
104	74.005	21	TRUE
105	73.996	21	TRUE

106	74.004	22	TRUE
107	73.999	22	TRUE
108	73.990	22	TRUE
109	74.006	22	TRUE
110	74.009	22	TRUE
111	74.010	23	TRUE
112	73.989	23	TRUE
113	73.990	23	TRUE
114	74.009	23	TRUE
115	74.014	23	TRUE
116	74.015	24	TRUE
117	74.008	24	TRUE
118	73.993	24	TRUE
119	74.000	24	TRUE
120	74.010	24	TRUE
121	73.982	25	TRUE
122	73.984	25	TRUE
123	73.995	25	TRUE
124	74.017	25	TRUE
125	74.013	25	TRUE
126	74.012	26	FALSE
127	74.015	26	FALSE
128	74.030	26	FALSE
129	73.986	26	FALSE
130	74.000	26	FALSE
131	73.995	27	FALSE
132	74.010	27	FALSE
133	73.990	27	FALSE
134	74.015	27	FALSE
135	74.001	27	FALSE
136	73.987	28	FALSE

137	73.999	28	FALSE
138	73.985	28	FALSE
139	74.000	28	FALSE
140	73.990	28	FALSE
141	74.008	29	FALSE
142	74.010	29	FALSE
143	74.003	29	FALSE
144	73.991	29	FALSE
145	74.006	29	FALSE
146	74.003	30	FALSE
147	74.000	30	FALSE
148	74.001	30	FALSE
149	73.986	30	FALSE
150	73.997	30	FALSE
151	73.994	31	FALSE
152	74.003	31	FALSE
153	74.015	31	FALSE
154	74.020	31	FALSE
155	74.004	31	FALSE
156	74.008	32	FALSE
157	74.002	32	FALSE
158	74.018	32	FALSE
159	73.995	32	FALSE
160	74.005	32	FALSE
161	74.001	33	FALSE
162	74.004	33	FALSE
163	73.990	33	FALSE
164	73.996	33	FALSE
165	73.998	33	FALSE
166	74.015	34	FALSE
167	74.000	34	FALSE

168	74.016	34	FALSE
169	74.025	34	FALSE
170	74.000	34	FALSE
171	74.030	35	FALSE
172	74.005	35	FALSE
173	74.000	35	FALSE
174	74.016	35	FALSE
175	74.012	35	FALSE
176	74.001	36	FALSE
177	73.990	36	FALSE
178	73.995	36	FALSE
179	74.010	36	FALSE
180	74.024	36	FALSE
181	74.015	37	FALSE
182	74.020	37	FALSE
183	74.024	37	FALSE
184	74.005	37	FALSE
185	74.019	37	FALSE
186	74.035	38	FALSE
187	74.010	38	FALSE
188	74.012	38	FALSE
189	74.015	38	FALSE
190	74.026	38	FALSE
191	74.017	39	FALSE
192	74.013	39	FALSE
193	74.036	39	FALSE
194	74.025	39	FALSE
195	74.026	39	FALSE
196	74.010	40	FALSE
197	74.005	40	FALSE
198	74.029	40	FALSE

```
199   74.000     40 FALSE
200   74.020     40 FALSE
> attach(pistonrings)
> diameter
  [1] 74.030 74.002 74.019 73.992 74.008 73.995 73.992 74.001 74.011 74.004 73.988
 [12] 74.024 74.021 74.005 74.002 74.002 73.996 73.993 74.015 74.009 73.992 74.007
 [23] 74.015 73.989 74.014 74.009 73.994 73.997 73.985 73.993 73.995 74.006 73.994
 [34] 74.000 74.005 73.985 74.003 73.993 74.015 73.988 74.008 73.995 74.009 74.005
 [45] 74.004 73.998 74.000 73.990 74.007 73.995 73.994 73.998 73.994 73.995 73.990
 [56] 74.004 74.000 74.007 74.000 73.996 73.983 74.002 73.998 73.997 74.012 74.006
 [67] 73.967 73.994 74.000 73.984 74.012 74.014 73.998 73.999 74.007 74.000 73.984
 [78] 74.005 73.998 73.996 73.994 74.012 73.986 74.005 74.007 74.006 74.010 74.018
 [89] 74.003 74.000 73.984 74.002 74.003 74.005 73.997 74.000 74.010 74.013 74.020
[100] 74.003 73.988 74.001 74.009 74.005 73.996 74.004 73.999 73.990 74.006 74.009
[111] 74.010 73.989 73.990 74.009 74.014 74.015 74.008 73.993 74.000 74.010 73.982
[122] 73.984 73.995 74.017 74.013 74.012 74.015 74.030 73.986 74.000 73.995 74.010
[133] 73.990 74.015 74.001 73.987 73.999 73.985 74.000 73.990 74.008 74.010 74.003
[144] 73.991 74.006 74.003 74.000 74.001 73.986 73.997 73.994 74.003 74.015 74.020
[155] 74.004 74.008 74.002 74.018 73.995 74.005 74.001 74.004 73.990 73.996 73.998
[166] 74.015 74.000 74.016 74.025 74.000 74.030 74.005 74.000 74.016 74.012 74.001
[177] 73.990 73.995 74.010 74.024 74.015 74.020 74.024 74.005 74.019 74.035 74.010
[188] 74.012 74.015 74.026 74.017 74.013 74.036 74.025 74.026 74.010 74.005 74.029
[199] 74.000 74.020
> piston<-qcc.groups(diameter,sample)
> piston
    [,1]   [,2]   [,3]   [,4]   [,5]
1  74.030 74.002 74.019 73.992 74.008
2  73.995 73.992 74.001 74.011 74.004
3  73.988 74.024 74.021 74.005 74.002
4  74.002 73.996 73.993 74.015 74.009
5  73.992 74.007 74.015 73.989 74.014
```

6 74.009 73.994 73.997 73.985 73.993

7 73.995 74.006 73.994 74.000 74.005

8 73.985 74.003 73.993 74.015 73.988

9 74.008 73.995 74.009 74.005 74.004

10 73.998 74.000 73.990 74.007 73.995

11 73.994 73.998 73.994 73.995 73.990

12 74.004 74.000 74.007 74.000 73.996

13 73.983 74.002 73.998 73.997 74.012

14 74.006 73.967 73.994 74.000 73.984

15 74.012 74.014 73.998 73.999 74.007

16 74.000 73.984 74.005 73.998 73.996

17 73.994 74.012 73.986 74.005 74.007

18 74.006 74.010 74.018 74.003 74.000

19 73.984 74.002 74.003 74.005 73.997

20 74.000 74.010 74.013 74.020 74.003

21 73.988 74.001 74.009 74.005 73.996

22 74.004 73.999 73.990 74.006 74.009

23 74.010 73.989 73.990 74.009 74.014

24 74.015 74.008 73.993 74.000 74.010

25 73.982 73.984 73.995 74.017 74.013

26 74.012 74.015 74.030 73.986 74.000

27 73.995 74.010 73.990 74.015 74.001

28 73.987 73.999 73.985 74.000 73.990

29 74.008 74.010 74.003 73.991 74.006

30 74.003 74.000 74.001 73.986 73.997

31 73.994 74.003 74.015 74.020 74.004

32 74.008 74.002 74.018 73.995 74.005

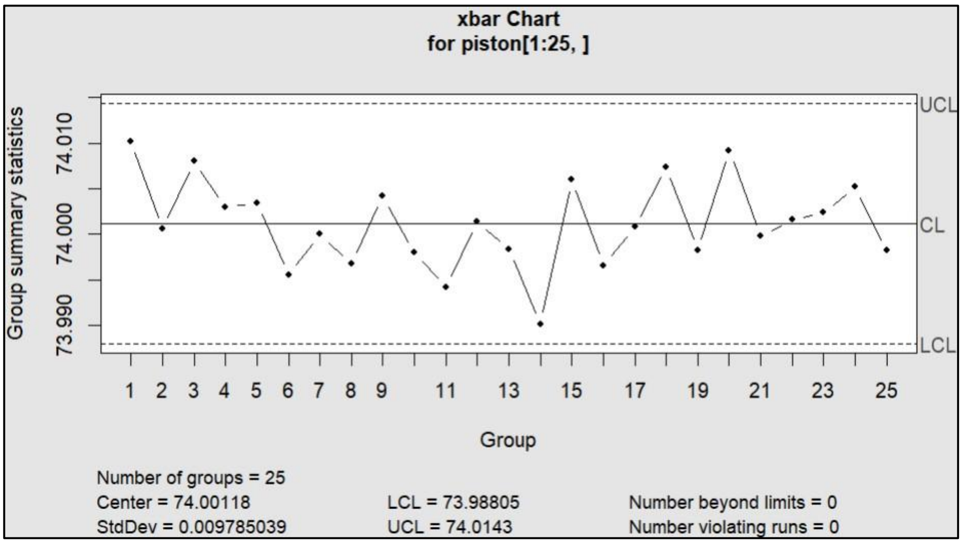
33 74.001 74.004 73.990 73.996 73.998

34 74.015 74.000 74.016 74.025 74.000

35 74.030 74.005 74.000 74.016 74.012

36 74.001 73.990 73.995 74.010 74.024

```
37 74.015 74.020 74.024 74.005 74.019
38 74.035 74.010 74.012 74.015 74.026
39 74.017 74.013 74.036 74.025 74.026
40 74.010 74.005 74.029 74.000 74.020
> qcc(piston[1:25,],type="xbar")
List of 11
 $ call      : language qcc(data = piston[1:25, ], type = "xbar")
 $ type      : chr "xbar"
 $ data.name : chr "piston[1:25, ]"
 $ data      : num [1:25, 1:5] 74 74 74 74 74 ...
  ..- attr(*, "dimnames")=List of 2
 $ statistics: Named num [1:25] 74 74 74 74 74 ...
  ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
 $ sizes     : Named int [1:25] 5 5 5 5 5 5 5 5 5 5 ...
  ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
 $ center    : num 74
 $ std.dev   : num 0.00979
 $ nsigmas   : num 3
 $ limits    : num [1, 1:2] 74 74
  ..- attr(*, "dimnames")=List of 2
 $ violations:List of 2
 - attr(*, "class")= chr "qcc"
```



> qcc(piston[1:25,],type="R")

List of 11

 $ call   : language qcc(data = piston[1:25,], type = "R")

 $ type   : chr "R"

 $ data.name : chr "piston[1:25,]"

 $ data   : num [1:25, 1:5] 74 74 74 74 74 ...

  ..- attr(*, "dimnames")=List of 2

 $ statistics: Named num [1:25] 0.038 0.019 0.036 0.022 0.026 ...

  ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...

 $ sizes   : Named int [1:25] 5 5 5 5 5 5 5 5 5 5 ...

  ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...

 $ center   : num 0.0228

 $ std.dev   : num 0.00979

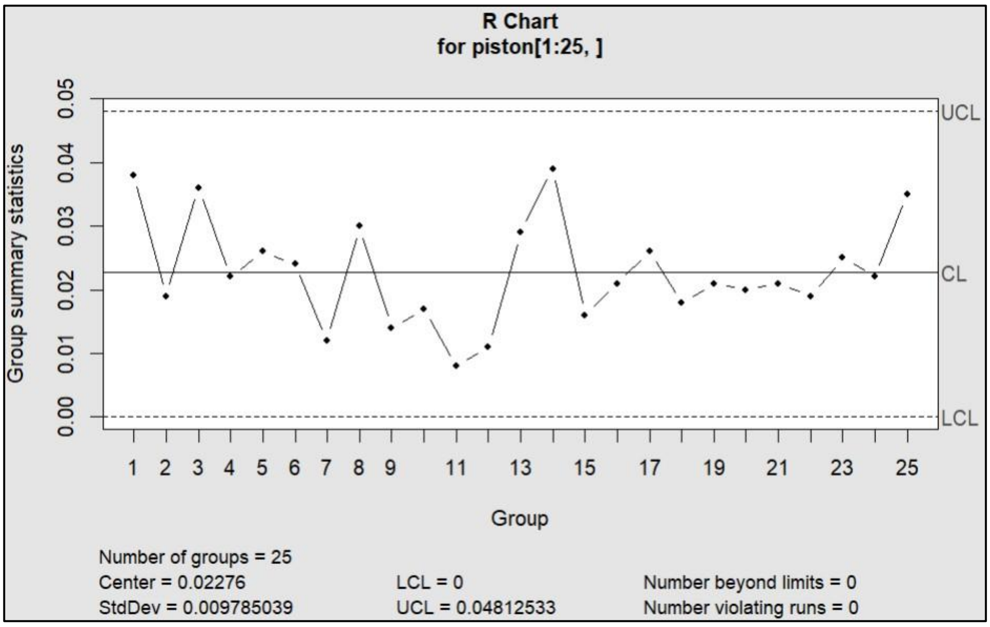
 $ nsigmas   : num 3

 $ limits   : num [1, 1:2] 0 0.0481

  ..- attr(*, "dimnames")=List of 2

 $ violations:List of 2

 - attr(*, "class")= chr "qcc"



> qcc(piston[1:25,],type="S")

List of 11

 $ call   : language qcc(data = piston[1:25,], type = "S")

$ type   : chr "S"

$ data.name : chr "piston[1:25,]"

$ data   : num [1:25, 1:5] 74 74 74 74 74 ...

 ..- attr(*, "dimnames")=List of 2

$ statistics: Named num [1:25] 0.01477 0.0075 0.01475 0.00908 0.01222 ...

 ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...

$ sizes   : Named int [1:25] 5 5 5 5 5 5 5 5 5 5 ...

 ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...

$ center   : num 0.00924

$ std.dev   : num 0.00983

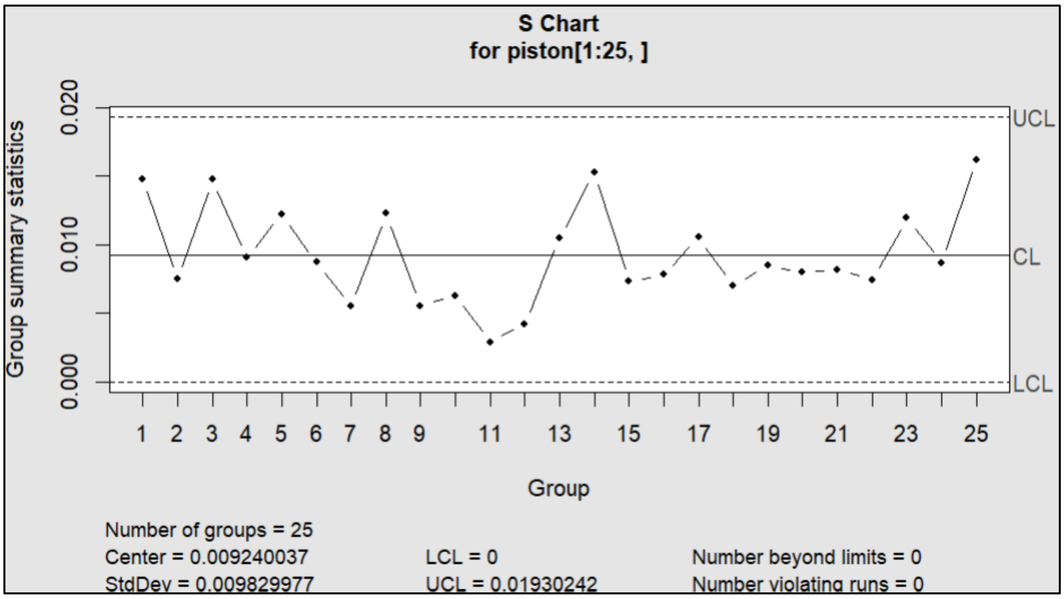
$ nsigmas   : num 3

$ limits   : num [1, 1:2] 0 0.0193

 ..- attr(*, "dimnames")=List of 2

$ violations:List of 2

- attr(*, "class")= chr "qcc"

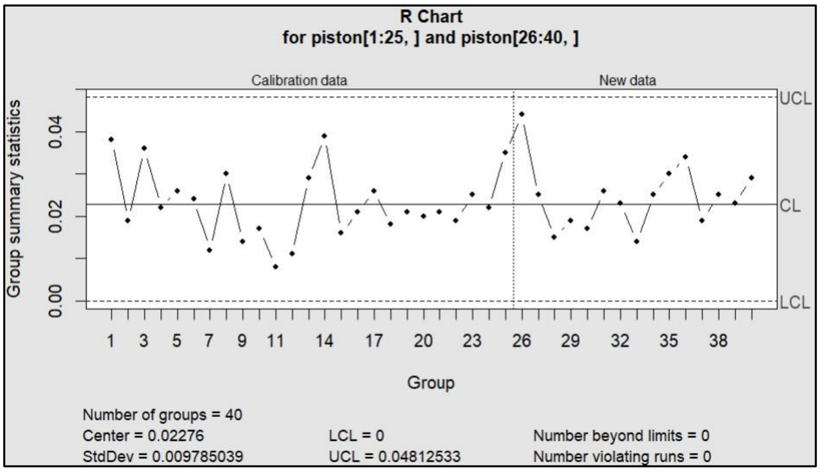


<u>Q.2)</u> Draw $\overline{x}$, R and S charts for the diameter in the dataset pistonrings for the last 15 samples after establishing control limits for the first 25 samples. Explain your findings.

> qcc(piston[1:25,],type="R",newdata = piston[26:40,])

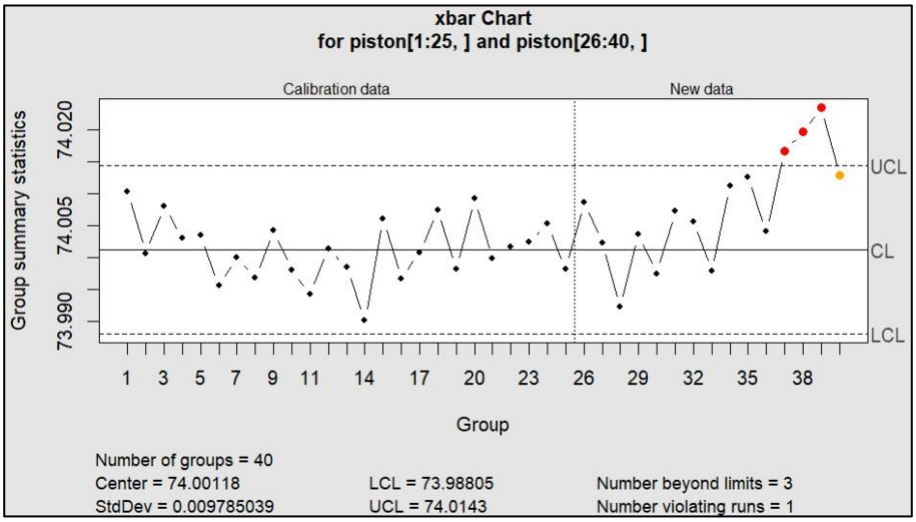
List of 15

```
$ call        : language qcc(data = piston[1:25, ], type = "R", newdata = piston[26:40, ])
$ type        : chr "R"
$ data.name   : chr "piston[1:25, ]"
$ data        : num [1:25, 1:5] 74 74 74 74 74 ...
 ..- attr(*, "dimnames")=List of 2
$ statistics  : Named num [1:25] 0.038 0.019 0.036 0.022 0.026 ...
 ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
$ sizes       : Named int [1:25] 5 5 5 5 5 5 5 5 5 5 ...
 ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
$ center      : num 0.0228
$ std.dev     : num 0.00979
$ newstats    : Named num [1:15] 0.044 0.025 0.015 0.019 0.017 ...
 ..- attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
$ newdata     : num [1:15, 1:5] 74 74 74 74 74 ...
 ..- attr(*, "dimnames")=List of 2
$ newsizes    : Named int [1:15] 5 5 5 5 5 5 5 5 5 5 ...
 ..- attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
$ newdata.name: chr "piston[26:40, ]"
$ nsigmas     : num 3
$ limits      : num [1, 1:2] 0 0.0481
 ..- attr(*, "dimnames")=List of 2
$ violations  :List of 2
- attr(*, "class")= chr "qcc"
```



> qcc(piston[1:25,],type="xbar",newdata = piston[26:40,]) **#Assignable cost in the points not in control**

```
List of 15
 $ call        : language qcc(data = piston[1:25, ], type = "xbar", newdata = piston[26:40, ])
 $ type        : chr "xbar"
 $ data.name   : chr "piston[1:25, ]"
 $ data        : num [1:25, 1:5] 74 74 74 74 74 ...
  ..- attr(*, "dimnames")=List of 2
 $ statistics  : Named num [1:25] 74 74 74 74 74 ...
  ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
 $ sizes       : Named int [1:25] 5 5 5 5 5 5 5 5 5 5 ...
  ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
 $ center      : num 74
 $ std.dev     : num 0.00979
 $ newstats    : Named num [1:15] 74 74 74 74 74 ...
  ..- attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
 $ newdata     : num [1:15, 1:5] 74 74 74 74 74 ...
  ..- attr(*, "dimnames")=List of 2
 $ newsizes    : Named int [1:15] 5 5 5 5 5 5 5 5 5 5 ...
  ..- attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
 $ newdata.name: chr "piston[26:40, ]"
 $ nsigmas     : num 3
 $ limits      : num [1, 1:2] 74 74
  ..- attr(*, "dimnames")=List of 2
 $ violations  :List of 2
 - attr(*, "class")= chr "qcc"
```




```
> qcc(piston[1:25,],type="S",newdata = piston[26:40,])
```

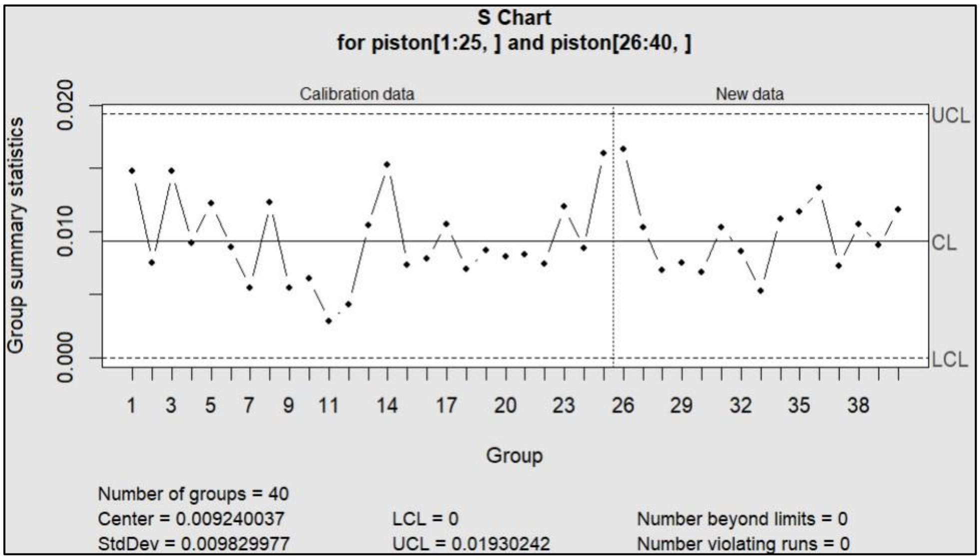
List of 15

```
 $ call       : language qcc(data = piston[1:25, ], type = "S", newdata = piston[26:40, ])
 $ type       : chr "S"
 $ data.name  : chr "piston[1:25, ]"
 $ data       : num [1:25, 1:5] 74 74 74 74 74 ...
  ..- attr(*, "dimnames")=List of 2
 $ statistics : Named num [1:25] 0.01477 0.0075 0.01475 0.00908 0.01222 ...
  ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
 $ sizes      : Named int [1:25] 5 5 5 5 5 5 5 5 5 5 ...
  ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
 $ center     : num 0.00924
 $ std.dev    : num 0.00983
 $ newstats   : Named num [1:15] 0.01655 0.01033 0.00691 0.0075 0.00673 ...
  ..- attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
 $ newdata    : num [1:15, 1:5] 74 74 74 74 74 ...
  ..- attr(*, "dimnames")=List of 2
 $ newsizes   : Named int [1:15] 5 5 5 5 5 5 5 5 5 5 ...
  ..- attr(*, "names")= chr [1:15] "26" "27" "28" "29" ...
 $ newdata.name: chr "piston[26:40, ]"
 $ nsigmas    : num 3
 $ limits     : num [1, 1:2] 0 0.0193
```

..- attr(*, "dimnames")=List of 2

$ violations :List of 2

- attr(*, "class")= chr "qcc"



Sample no 37,38,39 is out of control limits sample 40 is expected to violate the run. While the R chart and S chart is in control.

Q.3)

Construct a control chart for mean and range for the following data, samples of 5 beingtaken every hour:

Sample No.	1	2	3	4	5	6	7	8	9	10	11	12
Observations	42	42	19	36	42	51	60	18	15	69	64	61
	65	45	24	54	51	74	60	20	30	109	90	78
	75	68	80	69	57	75	72	27	39	113	93	94
	78	72	81	77	59	78	95	42	62	118	109	109
	87	90	81	84	80	132	138	60	84	153	112	136

Check if the process is in control. If not, revise the limits for the future.

> data1<-que_2

> data1

Observation Sample

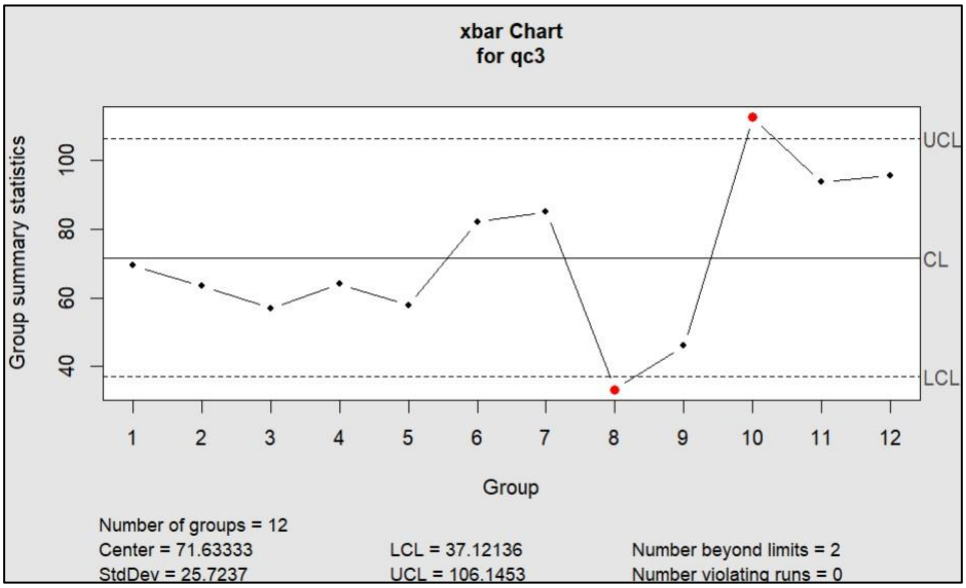
1 42 1

2 65 1

3	75	1
4	78	1
5	87	1
6	42	2
7	45	2
8	68	2
9	72	2
10	90	2
11	19	3
12	24	3
13	80	3
14	81	3
15	81	3
16	36	4
17	54	4
18	69	4
19	77	4
20	84	4
21	42	5
22	51	5
23	57	5
24	59	5
25	80	5
26	51	6
27	74	6
28	75	6
29	78	6
30	132	6
31	60	7
32	60	7
33	72	7

```
34        95    7
35       138     7
36        18    8
37        20    8
38        27    8
39        42    8
40        60    8
41        15    9
42        30    9
43        39    9
44        62    9
45        84    9
46        69    10
47       109     10
48       113     10
49       118     10
50       153     10
51        64    11
52        90    11
53        93    11
54       109     11
55       112     11
56        61    12
57        78    12
58        94    12
59       109     12
60       136     12
> qc3<-qcc.groups(data1$Observation,  data1$Sample)
> qc3
  [,1] [,2] [,3] [,4] [,5]
1   42   65   75   78   87
```

```
2   42  45  68  72  90
3   19  24  80  81  81
4   36  54  69  77  84
5   42  51  57  59  80
6   51  74  75  78 132
7   60  60  72  95 138
8   18  20  27  42  60
9   15  30  39  62  84
10  69 109 113 118 153
11  64  90  93 109 112
12  61  78  94 109 136
> qcc(qc3, type="xbar")
List of 11
 $ call      : language qcc(data = qc3, type = "xbar")
 $ type      : chr "xbar"
 $ data.name : chr "qc3"
 $ data      : int [1:12, 1:5] 42 42 19 36 42 51 60 18 15 69 ...
  ..- attr(*, "dimnames")=List of 2
 $ statistics: Named num [1:12] 69.4 63.4 57 64 57.8 ...
  ..- attr(*, "names")= chr [1:12] "1" "2" "3" "4" ...
 $ sizes     : Named int [1:12] 5 5 5 5 5 5 5 5 5 5 ...
  ..- attr(*, "names")= chr [1:12] "1" "2" "3" "4" ...
 $ center    : num 71.6
 $ std.dev   : num 25.7
 $ nsigmas   : num 3
 $ limits    : num [1, 1:2] 37.1 106.1
  ..- attr(*, "dimnames")=List of 2
 $ violations:List of 2
 - attr(*, "class")= chr "qcc"
```



The x-bar chart shows two points (Groups 8 and 10) beyond the control limits, indicating potential issues. The next step is to remove these points and create the x-bar chart again to check for stability.

> #new x_bar

> qcc(qc3[-c(8,10),],type="xbar", title="New x bar chart")

List of 11

$ call : language qcc(data = qc3[-c(8, 10),], type = "xbar", title = "New x bar chart")

$ type : chr "xbar"

$ data.name : chr "qc3[-c(8, 10),]"

$ data : int [1:10, 1:5] 42 42 19 36 42 51 60 15 64 61 ...

..- attr(*, "dimnames")=List of 2

$ statistics: Named num [1:10] 69.4 63.4 57 64 57.8 82 85 46 93.6 95.6

..- attr(*, "names")= chr [1:10] "1" "2" "3" "4" ...

$ sizes : Named int [1:10] 5 5 5 5 5 5 5 5 5 5

..- attr(*, "names")= chr [1:10] "1" "2" "3" "4" ...

$ center : num 71.4

$ std.dev : num 25.5

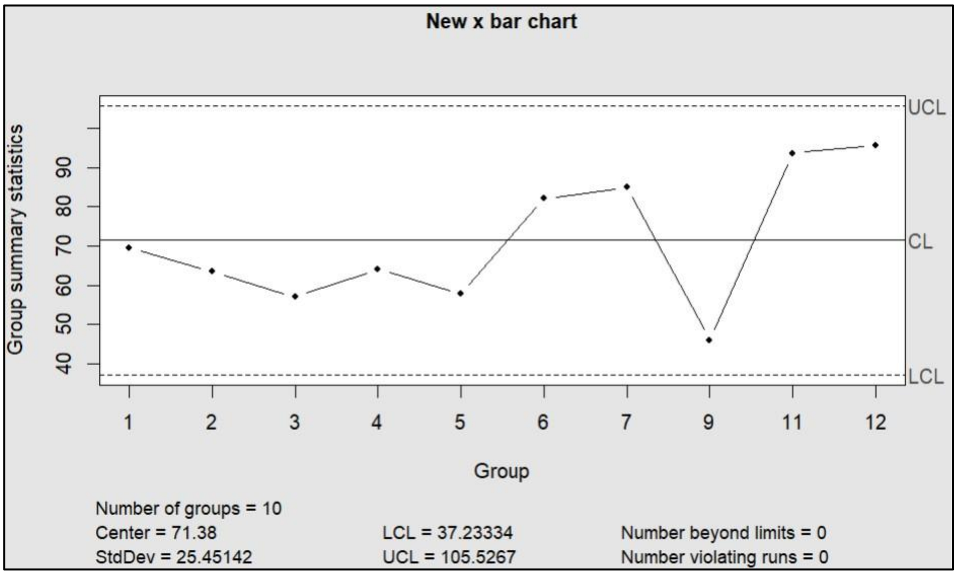
$ nsigmas : num 3

$ limits : num [1, 1:2] 37.2 105.5

..- attr(*, "dimnames")=List of 2

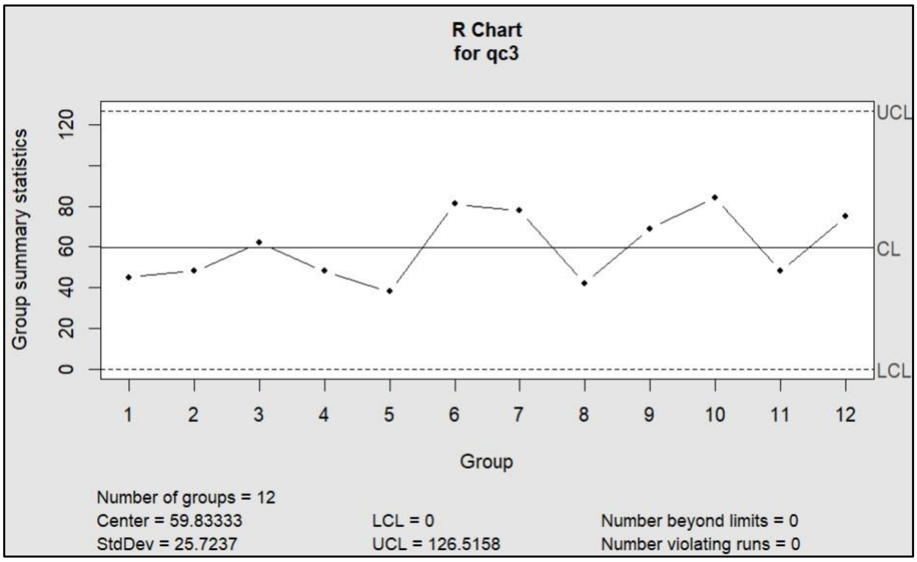
$ violations:List of 2

- attr(*, "class")= chr "qcc"

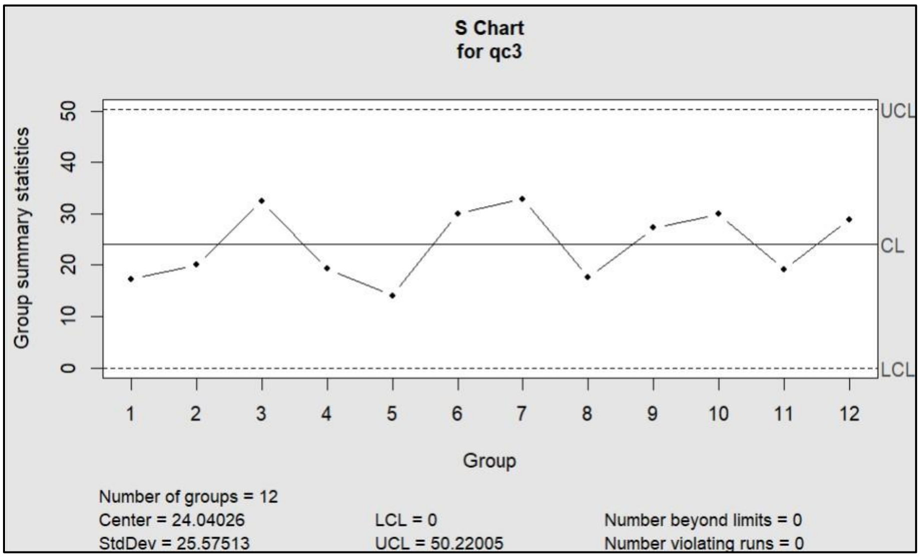



```
> qcc(qc3, type="R")

List of 11
 $ call      : language qcc(data = qc3, type = "R")
 $ type      : chr "R"
 $ data.name : chr "qc3"
 $ data      : int [1:12, 1:5] 42 42 19 36 42 51 60 18 15 69 ...
  ..- attr(*, "dimnames")=List of 2
 $ statistics: Named int [1:12] 45 48 62 48 38 81 78 42 69 84 ...
  ..- attr(*, "names")= chr [1:12] "1" "2" "3" "4" ...
 $ sizes     : Named int [1:12] 5 5 5 5 5 5 5 5 5 5 ...
  ..- attr(*, "names")= chr [1:12] "1" "2" "3" "4" ...
 $ center    : num 59.8
 $ std.dev   : num 25.7
 $ nsigmas   : num 3
 $ limits    : num [1, 1:2] 0 127
  ..- attr(*, "dimnames")=List of 2
 $ violations:List of 2
 - attr(*, "class")= chr "qcc"
```




```
> qcc(qc3, type="S")
List of 11
 $ call      : language qcc(data = qc3, type = "S")
 $ type      : chr "S"
 $ data.name : chr "qc3"
 $ data      : int [1:12, 1:5] 42 42 19 36 42 51 60 18 15 69 ...
  ..- attr(*, "dimnames")=List of 2
 $ statistics: Named num [1:12] 17.2 20 32.5 19.2 14.1 ...
  ..- attr(*, "names")= chr [1:12] "1" "2" "3" "4" ...
 $ sizes     : Named int [1:12] 5 5 5 5 5 5 5 5 5 5 ...
  ..- attr(*, "names")= chr [1:12] "1" "2" "3" "4" ...
 $ center    : num 24
 $ std.dev   : num 25.6
 $ nsigmas   : num 3
 $ limits    : num [1, 1:2] 0 50.2
  ..- attr(*, "dimnames")=List of 2
 $ violations:List of 2
 - attr(*, "class")= chr "qcc"
```



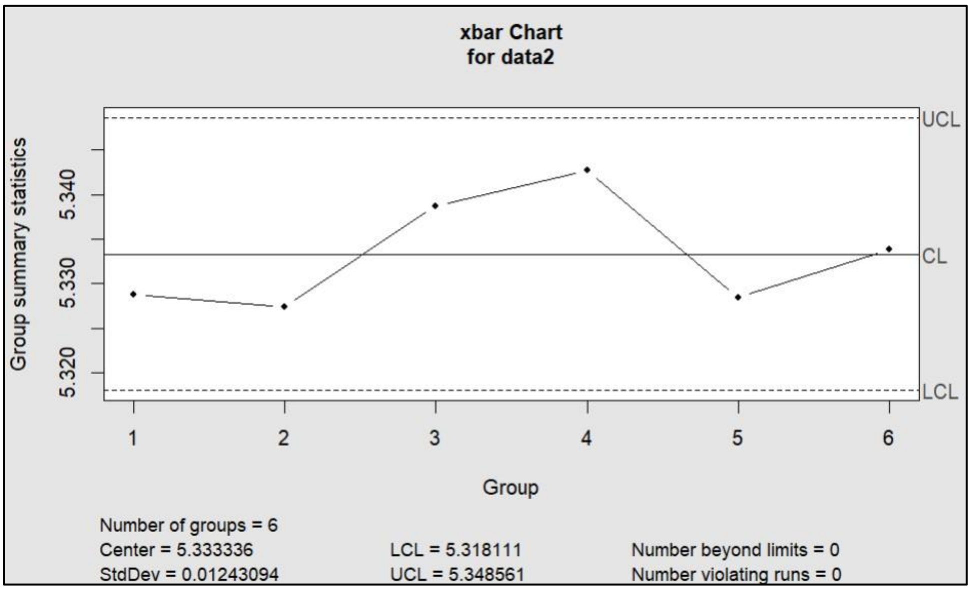
Q.4)

PH-parts, a plastic injection company, produces high-precision vaccination syringes. The inner barrel diameter, a critical quality metric, need to be monitored using SPC tools like X-bar and R-charts. Operators record six samples hourly, and data from six hours of production as follows. State your findings.

	Sample1	Sample2	Sample3	Sample4	Sample5	Sample6
Hour1	5.3314	5.3399	5.3244	5.3363	5.3228	5.3181
Hour2	5.3240	5.3214	5.3142	5.3237	5.342	5.3392
Hour3	5.3263	5.3404	5.3136	5.3565	5.3387	5.357
Hour4	5.3553	5.3600	5.3171	5.3319	5.3446	5.3474
Hour5	5.3379	5.3264	5.3150	5.3134	5.3375	5.3407
Hour6	5.3432	5.3352	5.3238	5.3463	5.334	5.3205

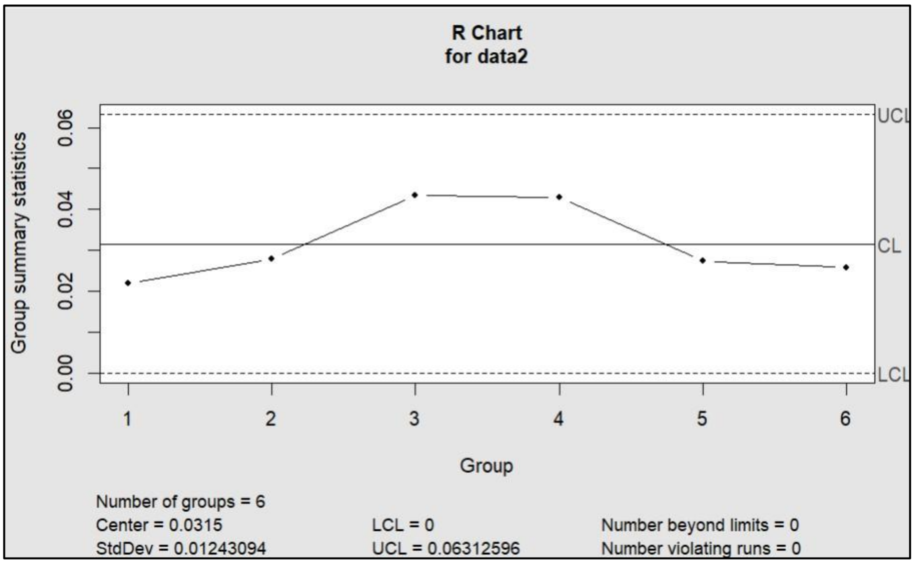
```
> data2<-que_4
> data2
      V1     V2     V3     V4     V5     V6
1 5.3314 5.3399 5.3244 5.3363 5.3228 5.3181
2 5.3240 5.3214 5.3142 5.3237 5.3420 5.3392
3 5.3263 5.3404 5.3136 5.3565 5.3387 5.3570
4 5.3553 5.3600 5.3171 5.3319 5.3446 5.3474
5 5.3379 5.3264 5.3150 5.3134 5.3375 5.3407
6 5.3432 5.3352 5.3238 5.3463 5.3340 5.3205
```

```
> qcc(data2, type="xbar")
List of 11
 $ call      : language qcc(data = data2, type = "xbar")
 $ type      : chr "xbar"
 $ data.name : chr "data2"
 $ data      : num [1:6, 1:6] 5.33 5.32 5.33 5.36 5.34 ...
  ..- attr(*, "dimnames")=List of 2
 $ statistics: Named num [1:6] 5.33 5.33 5.34 5.34 5.33 ...
  ..- attr(*, "names")= chr [1:6] "1" "2" "3" "4" ...
 $ sizes     : int [1:6] 6 6 6 6 6 6
 $ center    : num 5.33
 $ std.dev   : num 0.0124
 $ nsigmas   : num 3
 $ limits    : num [1, 1:2] 5.32 5.35
  ..- attr(*, "dimnames")=List of 2
 $ violations:List of 2
 - attr(*, "class")= chr "qcc"
```



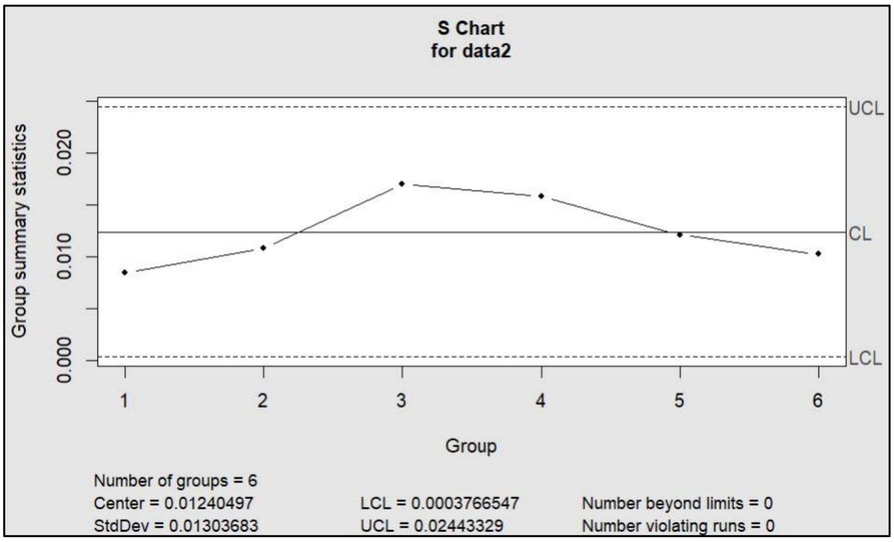

```
> qcc(data2, type="R")
List of 11
 $ call      : language qcc(data = data2, type = "R")
 $ type      : chr "R"
```

```
$ data.name : chr "data2"
$ data      : num [1:6, 1:6] 5.33 5.32 5.33 5.36 5.34 ...
 ..- attr(*, "dimnames")=List of 2
$ statistics: Named num [1:6] 0.0218 0.0278 0.0434 0.0429 0.0273 ...
 ..- attr(*, "names")= chr [1:6] "1" "2" "3" "4" ...
$ sizes     : int [1:6] 6 6 6 6 6 6
$ center    : num 0.0315
$ std.dev   : num 0.0124
$ nsigmas   : num 3
$ limits    : num [1, 1:2] 0 0.0631
 ..- attr(*, "dimnames")=List of 2
$ violations:List of 2
- attr(*, "class")= chr "qcc"
```




```
> qcc(data2, type="S")
List of 11
$ call      : language qcc(data = data2, type = "S")
$ type      : chr "S"
$ data.name : chr "data2"
$ data      : num [1:6, 1:6] 5.33 5.32 5.33 5.36 5.34 ...
 ..- attr(*, "dimnames")=List of 2
$ statistics: Named num [1:6] 0.00844 0.01084 0.01696 0.01585 0.01211 ...
```

```
..- attr(*, "names")= chr [1:6] "1" "2" "3" "4" ...
$ sizes     : int [1:6] 6 6 6 6 6 6
$ center    : num 0.0124
$ std.dev   : num 0.013
$ nsigmas   : num 3
$ limits    : num [1, 1:2] 0.000377 0.024433
..- attr(*, "dimnames")=List of 2
$ violations:List of 2
- attr(*, "class")= chr "qcc"
```



Q.5)

The following are mean and ranges of 20 samples of size 5 each. The data pertain to the overall length of a fragmentation bomb base manufactured during the war by the American store camp.

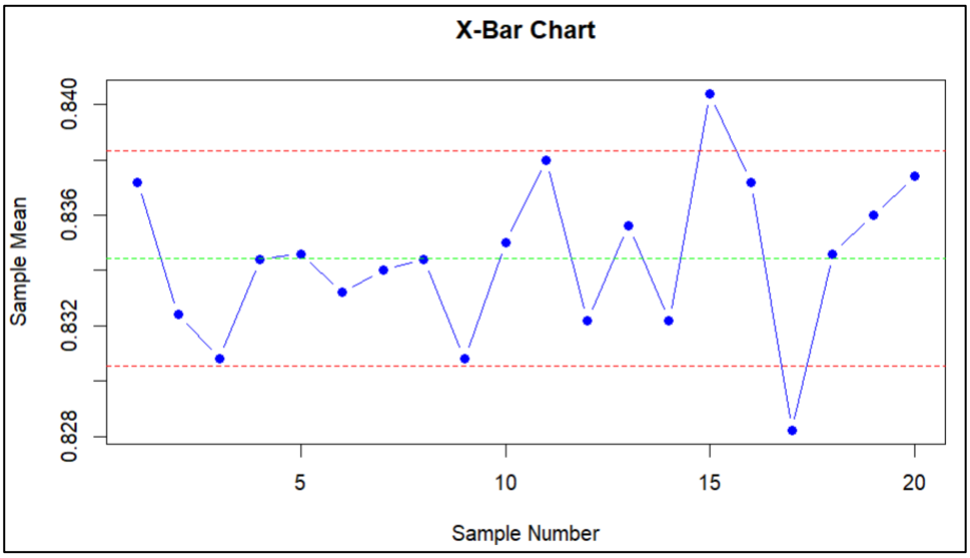
Group	1	2	3	4	5	6	7	8	9	10
Mean	0.8372	0.8324	0.8308	0.8344	0.8346	0.8332	0.834	0.8344	0.8308	0.835
Range	0.01	0.009	0.008	0.004	0.005	0.011	0.009	0.003	0.002	0.006
Group	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
Mean	0.838	0.8322	0.8356	0.8322	0.8404	0.8372	0.8282	0.8346	0.836	0.8374
Range	0.006	0.002	0.013	0.005	0.008	0.011	0.006	0.006	0.004	0.006

From these data, obtain control limits for X and R charts to control the length of bomb bases to be produced in future.

```
> tdata<-Q5.SQC1
> head(tdata)
```

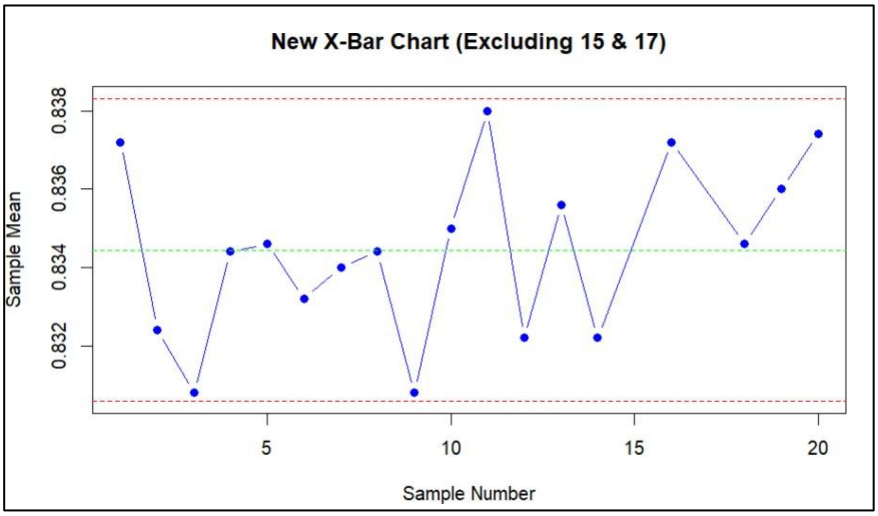
```
  Group   Mean Range
1     1 0.8372 0.010
2     2 0.8324 0.009
3     3 0.8308 0.008
4     4 0.8344 0.004
5     5 0.8346 0.005
6     6 0.8332 0.011
> #Mean of sample means and range
> mean_xbar<-mean(tdata$Mean)
> mean_r<-mean(tdata$Range)
> #control limits for x bar chart
> a2<-0.58 #for n=5
> ucl_xbar<-mean_xbar+a2*mean_r
> lcl_xbar<-mean_xbar-a2*mean_r
> cl_xbar<-mean_xbar
> #control limits for r chart
> d3<-0
> d4<-2.114
> ucl_r<-mean_r*d4
> lcl_r<-mean_r*d3
> cl_r<-mean_r
> mean_s<-mean(tdata$Range)
> #control limits for s chart
> b3<-0.0
> b4<-2.089
> ucl_s<-mean_s*b4
> lcl_s<-mean_s*b3
> cl_s<-mean_s
> new_tdata<-tdata[-c(15,17)]
> plot(tdata$Group, tdata$Mean, type="b", pch=16, col="blue",
+      ylim=c(min(lcl_xbar, min(tdata$Mean)), max(ucl_xbar, max(tdata$Mean))),
```

```
+      xlab="Sample Number", ylab="Sample Mean", main="X-Bar Chart")
> abline(h=ucl_xbar, col="red", lty=2)
> abline(h=lcl_xbar, col="red", lty=2)
> abline(h=cl_xbar, col="green", lty=2)
```



The x-bar chart shows two points (Groups 15 and 17) beyond the control limits, indicating potential issues. The next step is to remove these points and create the x-bar chart again to check for stability.

```
> # New x_bar chart
> mean_xbar<-mean(new_tdata$Mean)
> mean_r<-mean(new_tdata$Range)
> a2<-0.58 #for n=5
> ucl_xbar<-mean_xbar+a2*mean_r
> lcl_xbar<-mean_xbar-a2*mean_r
> cl_xbar<-mean_xbar
> # Plot the new X-bar chart
> plot(new_tdata$Group, new_tdata$Mean, type = "b", pch = 16, col = "blue",
+      ylim = c(min(lcl_xbar, min(new_tdata$Mean)), max(ucl_xbar, max(new_tdata$Mean))),
+      xlab = "Sample Number", ylab = "Sample Mean", main = "New X-Bar Chart (Excluding 15 & 17)")
> abline(h = ucl_xbar, col = "red", lty = 2)
> abline(h = lcl_xbar, col = "red", lty = 2)
> abline(h = cl_xbar, col = "green", lty = 2)
```



> # Plot the R chart

> plot(new_tdata$Group, new_tdata$Range, type = "b", pch = 16, col = "blue",

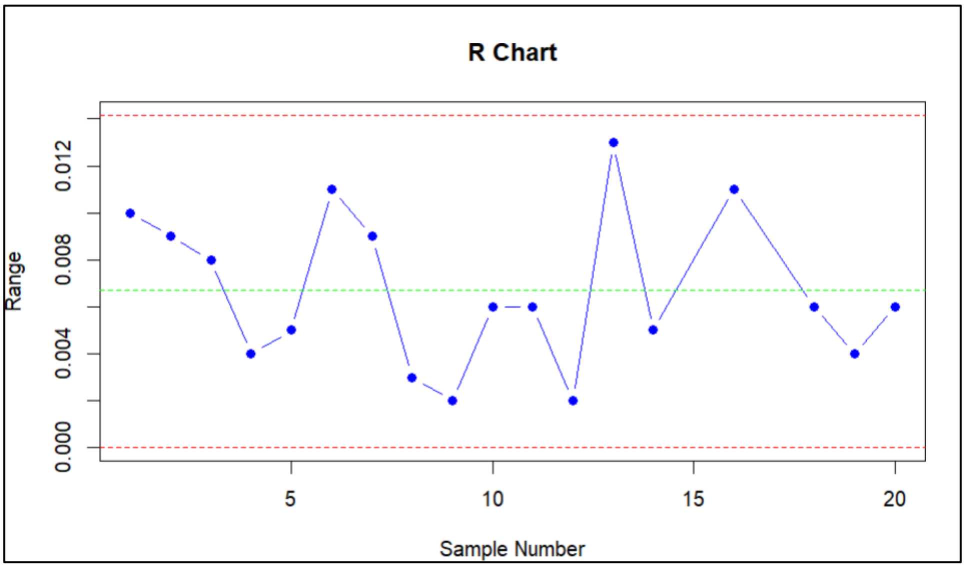
+ ylim = c(0, max(ucl_r, max(new_tdata$Range))),

+ xlab = "Sample Number", ylab = "Range", main = "R Chart")

> abline(h = ucl_r, col = "red", lty = 2)

> abline(h = lcl_r, col = "red", lty = 2)

> abline(h = cl_r, col = "green", lty = 2)

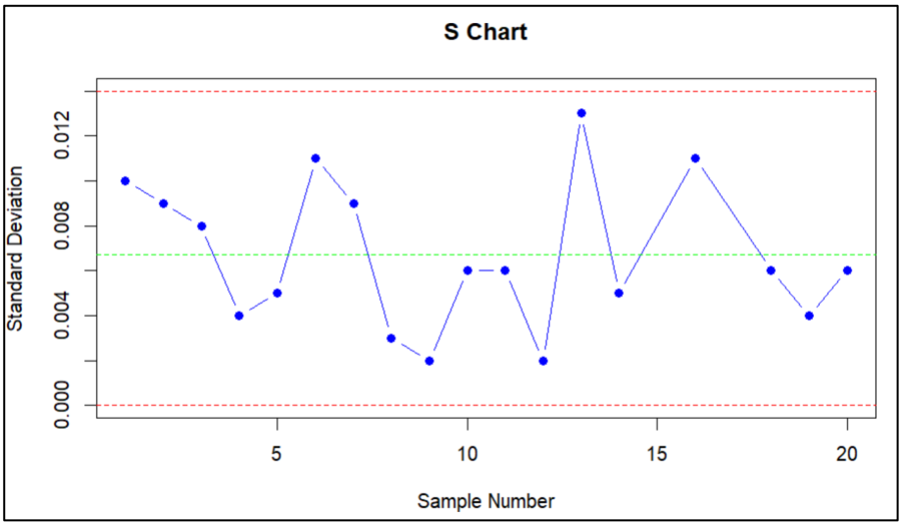


> # Plot the S chart

> plot(new_tdata$Group, new_tdata$Range, type = "b", pch = 16, col = "blue",

+ ylim = c(0, max(ucl_s, max(new_tdata$Range))),

```
+      xlab = "Sample Number", ylab = "Standard Deviation", main = "S Chart")
> abline(h = ucl_s, col = "red", lty = 2)
> abline(h = lcl_s, col = "green", lty = 2)
> abline(h = cl_s, col = "green", lty = 2)
```



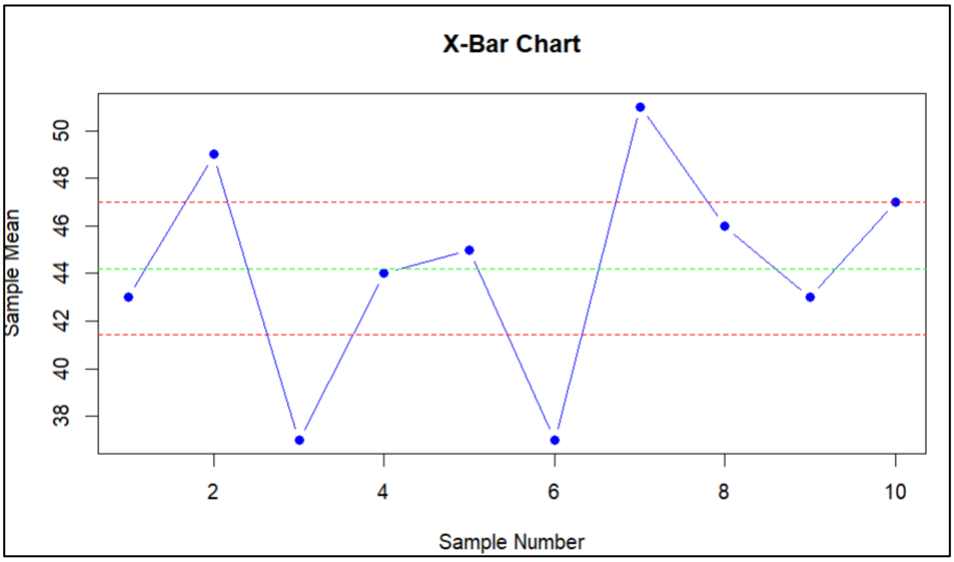
Q.6)

From the following data showing values of sample mean and range for 10 samples eachof 6, calculate the values for central line and the control limits for mean and range chart.Comment on the state of control of the mean chart. Can the process be said to be in a state of control regarding the variations inequality of the manufactured products?

Sample No.	1	2	3	4	5	6	7	8	9	10
Mean	43	49	37	44	45	37	51	46	43	47
Range	5	6	5	7	7	4	8	6	4	6

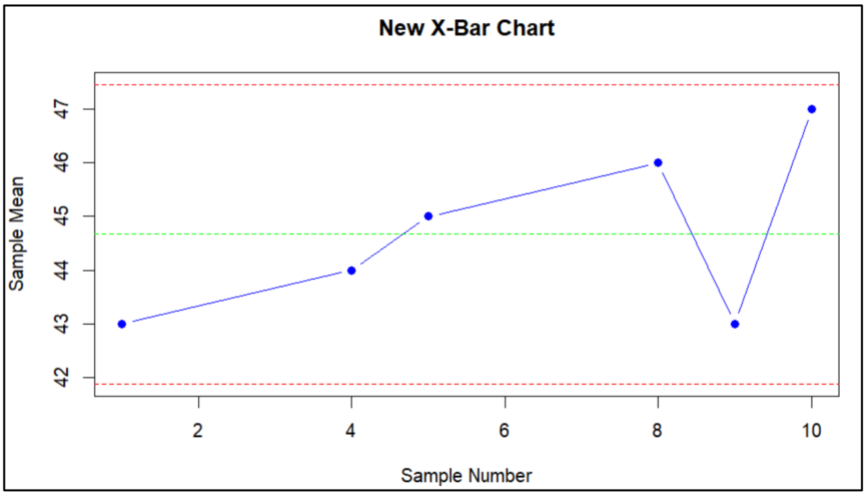
```
> sample_no <- 1:10
> mean_values <- c(43, 49, 37, 44, 45, 37, 51, 46, 43, 47)
> range_values <- c(5, 6, 5, 7, 7, 4, 8, 6, 4, 6)
> sample_size <- 6
> # Mean of sample means and ranges
> mean_xbar <- mean(mean_values)
> mean_r <- mean(range_values)
> # Constants for n = 6
```

```
> a2 <- 0.48
> d3 <- 0.076
> d4 <- 1.924
> b3 <- 0.0
> b4 <- 2.089
> # Control limits for X-bar chart
> ucl_xbar <- mean_xbar + a2 * mean_r
> lcl_xbar <- mean_xbar - a2 * mean_r
> cl_xbar <- mean_xbar
> # Control limits for R chart
> ucl_r <- mean_r * d4
> lcl_r <- mean_r * d3
> cl_r <- mean_r
> # Control limits for S chart
> ucl_s<-mean_s*b4
> lcl_s<-mean_s*b3
> cl_s<-mean_s
> # Plot X-bar Chart
> plot(sample_no, mean_values, type = "b", pch = 16, col = "blue",
+      ylim = c(min(lcl_xbar, min(mean_values)), max(ucl_xbar, max(mean_values))),
+      xlab = "Sample Number", ylab = "Sample Mean", main = "X-Bar Chart")
> abline(h = ucl_xbar, col = "red", lty = 2)
> abline(h = lcl_xbar, col = "red", lty = 2)
> abline(h = cl_xbar, col = "green", lty = 2)
```



The x-bar chart shows four points 2,3,6,7 beyond the control limits, indicating potential issues. The next step is to remove these points and create the x-bar chart again to check for stability.

```
> # Plot New X_bar Chart
> # Remove samples 2, 3, 6, and 7
> cleaned_data <- data6[-c(2, 3, 6, 7), ]
> # Recalculate Mean of sample means and range
> mean_xbar <- mean(cleaned_data$mean)
> mean_r <- mean(cleaned_data$range)
> # Control limits for X-bar chart (recalculated)
> ucl_xbar <- mean_xbar + a2 * mean_r
> lcl_xbar <- mean_xbar - a2 * mean_r
> cl_xbar <- mean_xbar
> # Plot New X-bar Chart
> plot(cleaned_data$`sample no.`, cleaned_data$mean, type = "b", pch = 16, col = "blue",
+     ylim = c(min(lcl_xbar, min(cleaned_data$mean)), max(ucl_xbar, max(cleaned_data$mean))),
+     xlab = "Sample Number", ylab = "Sample Mean", main = "New X-Bar Chart")
> abline(h = ucl_xbar, col = "red", lty = 2)
> abline(h = lcl_xbar, col = "red", lty = 2)
> abline(h = cl_xbar, col = "green", lty = 2)
```



> # Plot R Chart

> plot(sample_no, range_values, type = "b", pch = 16, col = "blue",

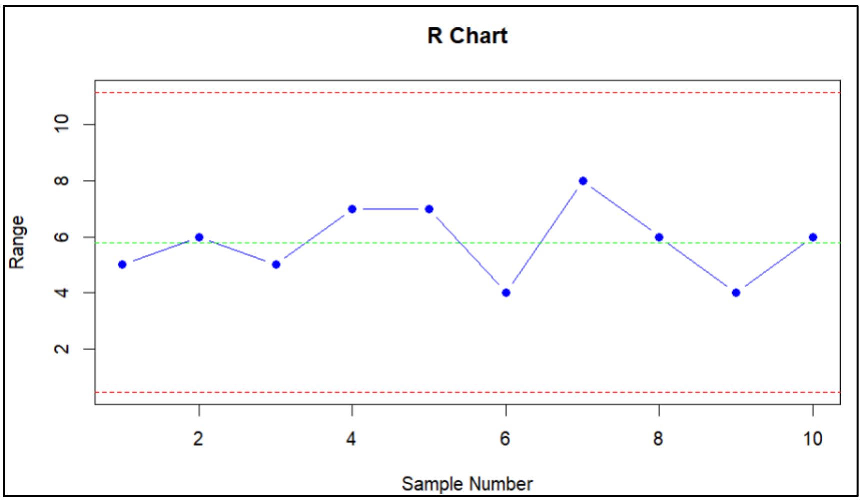
+ ylim = c(min(lcl_r, min(range_values)), max(ucl_r, max(range_values))),

+ xlab = "Sample Number", ylab = "Range", main = "R Chart")

> abline(h = ucl_r, col = "red", lty = 2)

> abline(h = lcl_r, col = "red", lty = 2)

> abline(h = cl_r, col = "green", lty = 2)



> # Plot S-Chart

> plot(sample_no, range_values, type = "b", pch = 16, col = "blue",

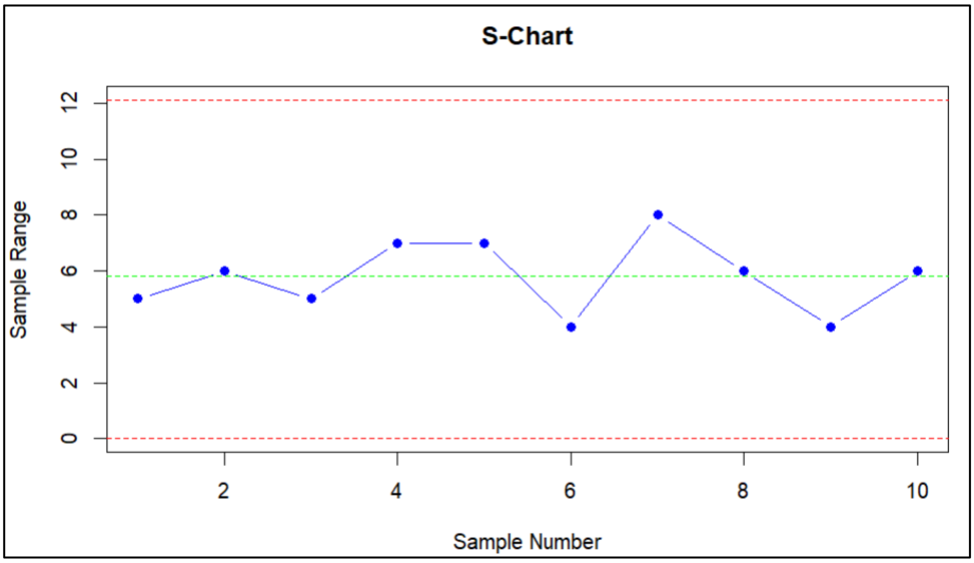
+ ylim = c(min(lcl_s, min(range_values)), max(ucl_s, max(range_values))),

+ xlab = "Sample Number", ylab = "Sample Range", main = "S-Chart")

> abline(h = ucl_s, col = "red", lty = 2)

> abline(h = lcl_s, col = "red", lty = 2)

> abline(h = cl_s, col = "green", lty = 2)



Q.7)

Construct a control chart for mean and range of the following data on the basis of fuses, sample of 4 being taken every hour:

Sample No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Observation	27	30	21	40	51	33	30	35	20	22	34	32	34	28	44
	23	17	44	21	34	30	22	48	34	50	22	48	32	30	32
	36	27	22	29	17	28	18	20	15	45	36	32	28	17	22
	24	32	28	24	10	12	22	47	42	41	44	33	48	23	41

Sample No.	16	17	18	19	20	21	22	23	24	25
Observation	26	38	26	42	30	23	28	25	30	38
	42	40	28	38	32	44	34	29	38	27
	35	51	34	52	39	48	39	40	44	39
	28	32	39	36	45	33	44	33	32	22

Comment on the state of control.

> data7<-que_7

> head(data7)

A tibble: 6 × 2

observation sample

<dbl> <dbl>

1 27 1

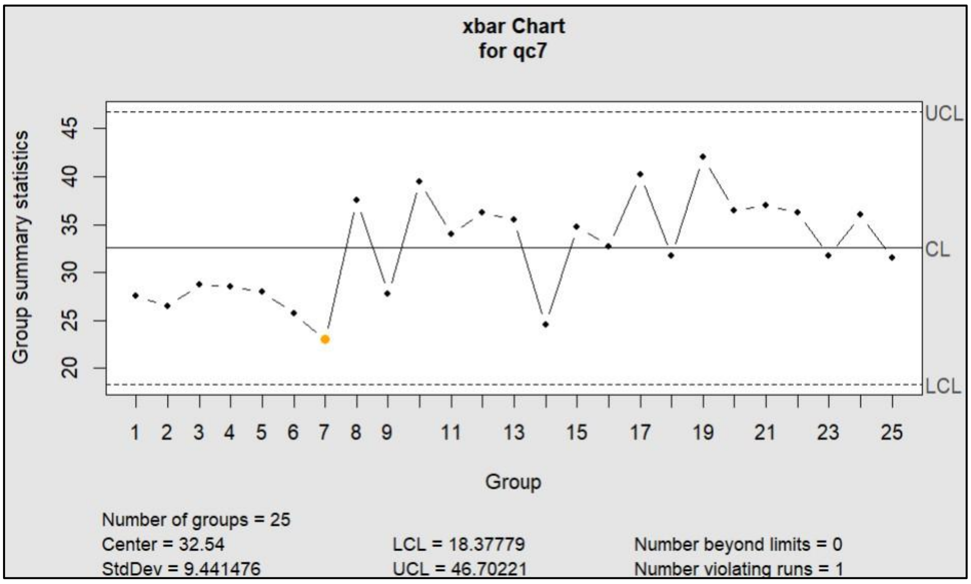
2 23 1

3 36 1

4 24 1

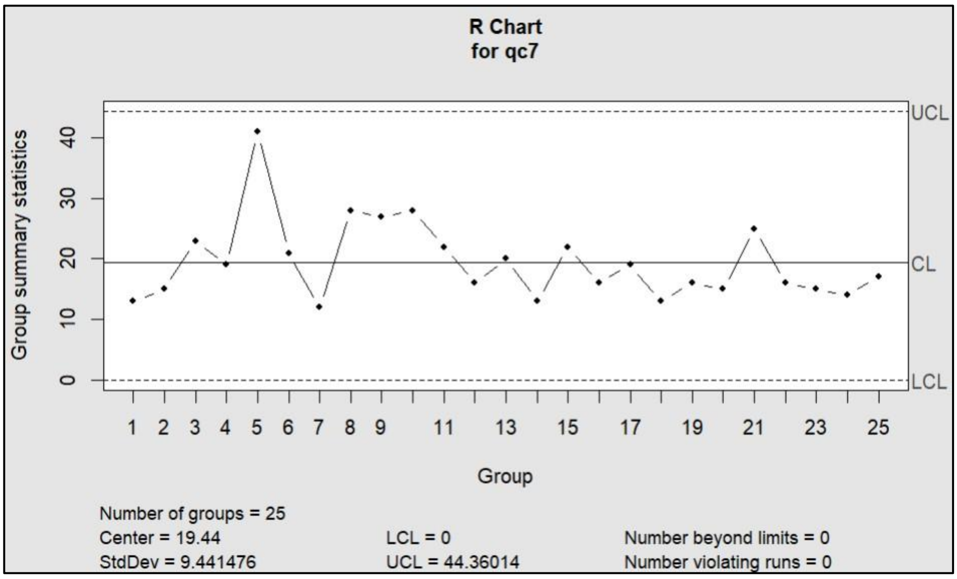
```
5       30    2
6       17    2
> qc7<-qcc.groups(data7$observation,  data7$sample)
> qc7
  [,1] [,2] [,3] [,4]
1   27   23   36   24
2   30   17   27   32
3   21   44   22   28
4   40   21   29   24
5   51   34   17   10
6   33   30   28   12
7   30   22   18   22
8   35   48   20   47
9   20   34   15   42
10   22   50   45   41
11   34   22   36   44
12   32   48   32   33
13   34   32   28   48
14   28   30   17   23
15   44   32   22   41
16   26   42   35   28
17   38   40   51   32
18   26   28   34   39
19   42   38   52   36
20   30   32   39   45
21   23   44   48   33
22   28   34   39   44
23   25   29   40   33
24   30   38   44   32
25   38   27   39   22
> qcc(qc7, type="xbar")
```

```
List of 11
 $ call      : language qcc(data = qc7, type = "xbar")
 $ type      : chr "xbar"
 $ data.name : chr "qc7"
 $ data      : num [1:25, 1:4] 27 30 21 40 51 33 30 35 20 22 ...
  ..- attr(*, "dimnames")=List of 2
 $ statistics: Named num [1:25] 27.5 26.5 28.8 28.5 28 ...
  ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
 $ sizes     : Named int [1:25] 4 4 4 4 4 4 4 4 4 4 ...
  ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
 $ center    : num 32.5
 $ std.dev   : num 9.44
 $ nsigmas   : num 3
 $ limits    : num [1, 1:2] 18.4 46.7
  ..- attr(*, "dimnames")=List of 2
 $ violations:List of 2
 - attr(*, "class")= chr "qcc"
```




```
> qcc(qc7, type="R")
List of 11
 $ call      : language qcc(data = qc7, type = "R")
 $ type      : chr "R"
```

```
$ data.name : chr "qc7"
$ data      : num [1:25, 1:4] 27 30 21 40 51 33 30 35 20 22 ...
 ..- attr(*, "dimnames")=List of 2
$ statistics: Named num [1:25] 13 15 23 19 41 21 12 28 27 28 ...
 ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
$ sizes     : Named int [1:25] 4 4 4 4 4 4 4 4 4 4 ...
 ..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...
$ center    : num 19.4
$ std.dev   : num 9.44
$ nsigmas   : num 3
$ limits    : num [1, 1:2] 0 44.4
 ..- attr(*, "dimnames")=List of 2
$ violations:List of 2
- attr(*, "class")= chr "qcc"
```




```
> qcc(qc7, type="S")
List of 11
$ call      : language qcc(data = qc7, type = "S")
$ type      : chr "S"
$ data.name : chr "qc7"
$ data      : num [1:25, 1:4] 27 30 21 40 51 33 30 35 20 22 ...
 ..- attr(*, "dimnames")=List of 2
```

$ statistics: Named num [1:25] 5.92 6.66 10.63 8.35 18.35 ...

..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...

$ sizes : Named int [1:25] 4 4 4 4 4 4 4 4 4 4 ...

..- attr(*, "names")= chr [1:25] "1" "2" "3" "4" ...

$ center : num 8.71

$ std.dev : num 9.45

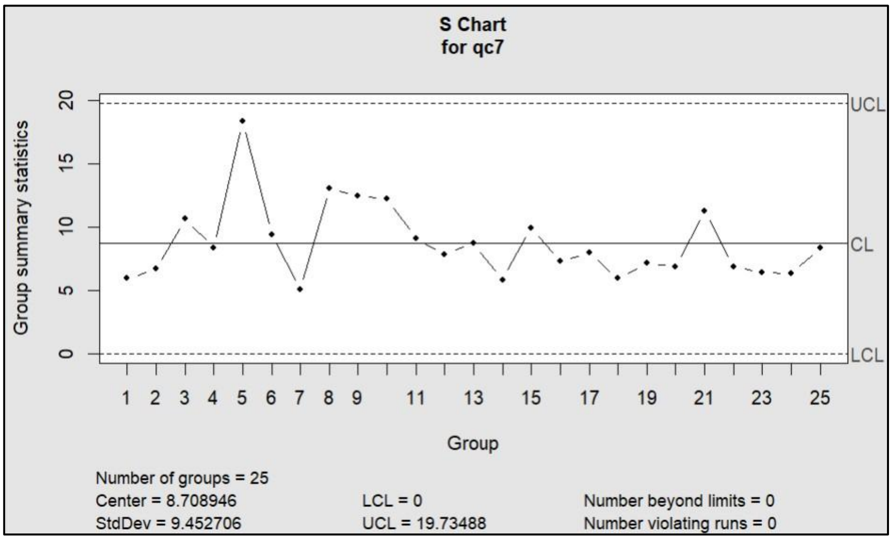
$ nsigmas : num 3

$ limits : num [1, 1:2] 0 19.7

..- attr(*, "dimnames")=List of 2

$ violations:List of 2

- attr(*, "class")= chr "qcc"



<u>Q.8)</u> A machine is set to deliver packets of a given weight. 10 samples of size 5 each were recorded as follows: 15, 17, 15, 18, 17, 14, 18, 15, 17 and 16. The mean range in these samples is 7.4. Construct a mean chart. Are all the sample points lying within the control limits?

```
> sample_means <- c(15, 17, 15, 18, 17, 14, 18, 15, 17, 16)
> mean_range <- 7.4
> sample_size <- 5
> d2 <- 2.326
> #Calculate process standard deviation
```

```
> sigma <- mean_range / d2
> #Calculate control limits
> A2 <- 3 / (d2 * sqrt(sample_size))
> center <- mean(sample_means)
> UCL <- center + A2 * mean_range
> LCL <- center - A2 * mean_range
> #Display calculated control limits
> cat("Center Line (CL):", center, "\n")
Center Line (CL): 16.2
> cat("Upper Control Limit (UCL):", UCL, "\n")
Upper Control Limit (UCL): 20.46833
> cat("Lower Control Limit (LCL):", LCL, "\n")
Lower Control Limit (LCL): 11.93167
> #Construct and display the mean chart
> mean_chart <- qcc(data = sample_means,
+             type = "xbar",
+             sizes = sample_size,
+             center = center,
+             std.dev = sigma)
> print(mean_chart)
List of 11
 $ call     : language qcc(data = sample_means, type = "xbar", sizes = sample_size, center = center, std.dev = sigma)
 $ type     : chr "xbar"
 $ data.name : chr "sample_means"
 $ data     : num [1:10, 1] 15 17 15 18 17 14 18 15 17 16
  ..- attr(*, "dimnames")=List of 2
 $ statistics: Named num [1:10] 15 17 15 18 17 14 18 15 17 16
  ..- attr(*, "names")= chr [1:10] "1" "2" "3" "4" ...
 $ sizes    : num [1:10] 5 5 5 5 5 5 5 5 5 5
 $ center   : num 16.2
```

$ std.dev   : num 3.18

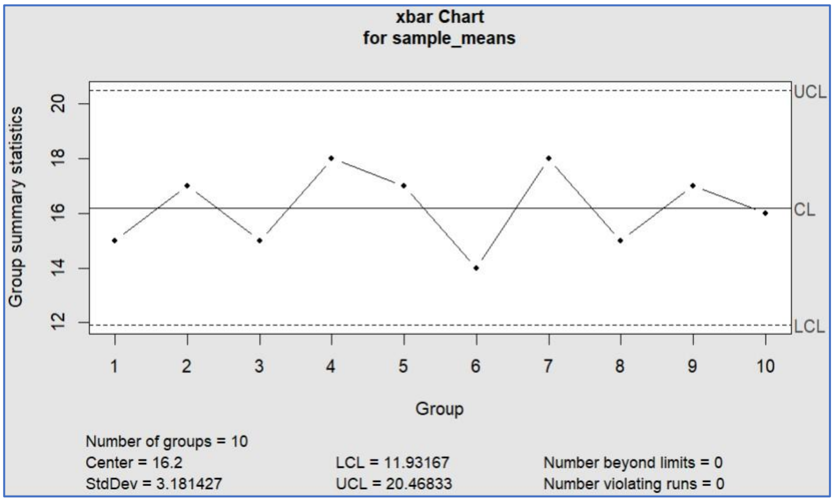
$ nsigmas   : num 3

$ limits   : num [1, 1:2] 11.9 20.5

 ..- attr(*, "dimnames")=List of 2

$ violations:List of 2

- attr(*, "class")= chr "qcc"



<u>Q.9)</u> A drilling machine bores holes with a mean diameter of 0.523 cm and a standard deviation of 0.0032 cm. Calculate 2 – sigma and 3 – sigma upper and lower control limits for means of samples 4 and prepare a control chart.

```
> #Parameters
> mu <- 0.523
> sigma <- 0.0032
> n <- 4
> #Calculate control limits
> sigma_sample <- sigma / sqrt(n)
> ucl_2 <- mu + 2 * sigma_sample
> lcl_2 <- mu - 2 * sigma_sample
> ucl_3 <- mu + 3 * sigma_sample
> lcl_3 <- mu - 3 * sigma_sample
> #Simulate sample means
> set.seed(123)
```

```
> sample_means <- rnorm(100, mean = mu, sd = sigma_sample)
> #Create control chart
> plot(sample_means, type = "o", col = "blue", pch = 16,
+     ylim = c(min(lcl_3, min(sample_means)) - 0.001, max(ucl_3, max(sample_means)) + 0.001),
+     xlab = "Sample Index", ylab = "Sample Mean",
+     main = "Control Chart")
> abline(h = mu, col = "black", lwd = 2, lty = 2)  # Mean line
> abline(h = c(ucl_2, lcl_2), col = "orange", lwd = 2, lty = 2)  # 2-sigma limits
> abline(h = c(ucl_3, lcl_3), col = "red", lwd = 2, lty = 2)  # 3-sigma limits
> legend("topright", legend = c("Mean", "2-sigma Limits", "3-sigma Limits"),
+      col = c("black", "orange", "red"), lty = 2, lwd = 2)
```