EDA LA-1

2022-06-23

1.Reading the data from the dataset

```
data=read.csv("C:/Users/Ananya/OneDrive/Documents/Rstudio/R/COVID-19 India Statewise Vaccine Data.csv")
```

2.To view all objects

```
ls()
```

```
## [1] "data"
```

3.Using ls() for pattern matching

```
ls(data)
```

```
## [1] "Dose.1"                 "Dose.2"
## [3] "Population"             "State.UTs"
## [5] "Total.Vaccination.Doses"
```

4.Finding the class of dataset

```
class(data)
```

```
## [1] "data.frame"
```

5.Finding the structure of dataset

```
str(data)
```

```
## 'data.frame':    36 obs. of  5 variables:
##  $ State.UTs              : chr  "Andaman and Nicobar" "Andhra Pradesh"
"Arunachal Pradesh" "Assam" ...
##  $ Total.Vaccination.Doses: int  624118 84043879 1576467 42130902
116997161 1989833 35836458 1313783 30381117 2558857 ...
##  $ Dose.1                 : int  311046 40596914 851479 22470609 62060111
1086102 18693176 727305 16492260 1349270 ...
##  $ Dose.2                 : int  313072 43446965 724988 19660293 54937050
903731 17143282 586478 13888857 1209587 ...
##  $ Population             : int  399001 91702478 1711947 35998752
128500364 1158040 32199722 773997 19301096 1521992 ...
```

6.Selectind the element present in 3rd row and 3rd column

```
data[3,3]
```

```
## [1] 851479
```

7.Selecting 3rd row and displaying columns 1 to 4

```
data[3,1:4]
```

```
##           State.UTs Total.Vaccination.Doses Dose.1 Dose.2
## 3 Arunachal Pradesh                 1576467 851479 724988
```

8.Selecting the first column alone

```
data[,1]
```

```
##  [1] "Andaman and Nicobar"
##  [2] "Andhra Pradesh"
##  [3] "Arunachal Pradesh"
##  [4] "Assam"
##  [5] "Bihar"
##  [6] "Chandigarh"
##  [7] "Chhattisgarh"
##  [8] "Dadra and Nagar Haveli and Daman and Diu"
##  [9] "Delhi"
## [10] "Goa"
## [11] "Gujarat"
## [12] "Haryana"
## [13] "Himachal Pradesh"
## [14] "Jammu and Kashmir"
## [15] "Jharkhand"
## [16] "Karnataka"
## [17] "Kerala"
## [18] "Ladakh"
## [19] "Lakshadweep"
## [20] "Madhya Pradesh"
## [21] "Maharashtra"
## [22] "Manipur"
## [23] "Meghalaya"
## [24] "Mizoram"
## [25] "Nagaland"
## [26] "Odisha"
## [27] "Puducherry"
## [28] "Punjab"
## [29] "Rajasthan"
## [30] "Sikkim"
## [31] "Tamil Nadu"
## [32] "Telangana"
## [33] "Tripura"
## [34] "Uttar Pradesh"
## [35] "Uttarakhand"
## [36] "West Bengal"
```

9.Displaying specific rows and all columns

```
data[c(1,3),]
```

```
##             State.UTs Total.Vaccination.Doses Dose.1 Dose.2 Population
## 1 Andaman and Nicobar                  624118 311046 313072     399001
## 3   Arunachal Pradesh                 1576467 851479 724988    1711947
```

10.Displaying specific rows and all columns except 4th column

```
data[c(1,3),-4]
```

```
##             State.UTs Total.Vaccination.Doses Dose.1 Population
## 1 Andaman and Nicobar                  624118 311046     399001
## 3   Arunachal Pradesh                 1576467 851479    1711947
```

11.Fetching column data by its name

```
data[c(1,3),"State.UTs"]
```

```
## [1] "Andaman and Nicobar" "Arunachal Pradesh"
```

12.Sorting the column

```
sort(data$State.UTs)
```

```
##  [1] "Andaman and Nicobar"
##  [2] "Andhra Pradesh"
##  [3] "Arunachal Pradesh"
##  [4] "Assam"
##  [5] "Bihar"
##  [6] "Chandigarh"
##  [7] "Chhattisgarh"
##  [8] "Dadra and Nagar Haveli and Daman and Diu"
##  [9] "Delhi"
## [10] "Goa"
## [11] "Gujarat"
## [12] "Haryana"
## [13] "Himachal Pradesh"
## [14] "Jammu and Kashmir"
## [15] "Jharkhand"
## [16] "Karnataka"
## [17] "Kerala"
## [18] "Ladakh"
## [19] "Lakshadweep"
## [20] "Madhya Pradesh"
## [21] "Maharashtra"
## [22] "Manipur"
## [23] "Meghalaya"
## [24] "Mizoram"
## [25] "Nagaland"
## [26] "Odisha"
## [27] "Puducherry"
## [28] "Punjab"
## [29] "Rajasthan"
## [30] "Sikkim"
## [31] "Tamil Nadu"
## [32] "Telangana"
## [33] "Tripura"
## [34] "Uttar Pradesh"
## [35] "Uttarakhand"
## [36] "West Bengal"
```

13.Ordering a particular column in a dataset

```
order(data$Population)
```

```
##  [1] 19 18  1 30  8  6 24 10 27  3 25 22 23 33 13 35 14  9 12 28  7 17  4
32 15
## [26] 26 16 11 29 31 20  2 36 21  5 34
```

14.Determine the rank of particular column

```
rank(data$Dose.1)
```

```
##  [1]  3 27  8 23 33 10 19  5 18 12 28 21 15 17 20 29 24  2  1 31 35 13 11
6  7
## [26] 26  9 22 30  4 32 25 14 36 16 34
```

15.Displaying first six rows of the dataset

```
head(data)
```

```
##             State.UTs Total.Vaccination.Doses   Dose.1   Dose.2 Population
## 1 Andaman and Nicobar                  624118   311046   313072     399001
## 2      Andhra Pradesh                84043879 40596914 43446965   91702478
## 3   Arunachal Pradesh                 1576467   851479   724988    1711947
## 4               Assam                42130902 22470609 19660293   35998752
## 5               Bihar               116997161 62060111 54937050  128500364
## 6          Chandigarh                 1989833  1086102   903731    1158040
```

16.Displaying the last six rows of the dataset

```
tail(data)
```

```
##        State.UTs Total.Vaccination.Doses    Dose.1    Dose.2 Population
## 31    Tamil Nadu               102131951  54996643  47135308   83697770
## 32     Telangana                57832105  29452703  28379402   38157311
## 33       Tripura                 4894221   2644217   2250004    4184959
## 34 Uttar Pradesh               287827071 153151985 134675086  231502578
## 35   Uttarakhand                15936251   8140263   7795988   11700099
## 36   West Bengal               126778443  67031629  59746814  100896618
```

###17.Displaying first 3 rows in a dataset

```
head(data,n=3)
```

```
##             State.UTs Total.Vaccination.Doses   Dose.1   Dose.2 Population
## 1 Andaman and Nicobar                  624118   311046   313072     399001
## 2      Andhra Pradesh                84043879 40596914 43446965   91702478
## 3   Arunachal Pradesh                 1576467   851479   724988    1711947
```

18.Summary of dataset

```
summary(data)
```

```
##   State.UTs         Total.Vaccination.Doses     Dose.1
##  Length:36          Min.   :   112063       Min.   :    56773
##  Class :character   1st Qu.:  1886492       1st Qu.:  1037732
##  Mode  :character   Median : 33108788       Median : 17592718
```

```
##                     Mean   : 48212343       Mean   : 25379940
##                     3rd Qu.: 86911628       3rd Qu.: 42740998
##                     Max.   :287827071       Max.   :153151985
##      Dose.2            Population
##  Min.   :    55290   Min.   :    66001
##  1st Qu.:   859045   1st Qu.:  1695473
##  Median : 14597482   Median : 24100882
##  Mean   : 22832403   Mean   : 39718608
##  3rd Qu.: 43747444   3rd Qu.: 69799860
##  Max.   :134675086   Max.   :231502578
```

19.To display column names

```
colnames(data)
```

```
## [1] "State.UTs"               "Total.Vaccination.Doses"
## [3] "Dose.1"                  "Dose.2"
## [5] "Population"
```

20.To display row names

```
row.names(data)
```

```
##  [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10" "11" "12" "13" "14"
"15"
## [16] "16" "17" "18" "19" "20" "21" "22" "23" "24" "25" "26" "27" "28" "29"
"30"
## [31] "31" "32" "33" "34" "35" "36"
```

21.To display both row and column names

```
dimnames(data)
```

```
## [[1]]
##  [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10" "11" "12" "13" "14"
"15"
## [16] "16" "17" "18" "19" "20" "21" "22" "23" "24" "25" "26" "27" "28" "29"
"30"
## [31] "31" "32" "33" "34" "35" "36"
##
## [[2]]
## [1] "State.UTs"               "Total.Vaccination.Doses"
## [3] "Dose.1"                  "Dose.2"
## [5] "Population"
```

22.To find the transpose of the data frame

```
t(data)
```

```
##                          [,1]                  [,2]
## State.UTs                "Andaman and Nicobar" "Andhra Pradesh"
## Total.Vaccination.Doses "   624118"            " 84043879"
## Dose.1                   "   311046"            " 40596914"
## Dose.2                   "   313072"            " 43446965"
## Population               "   399001"            " 91702478"
```

```
##                          [,3]                [,4]        [,5]
## State.UTs                "Arunachal Pradesh" "Assam"     "Bihar"
## Total.Vaccination.Doses "  1576467"          " 42130902" "116997161"
## Dose.1                  "   851479"          " 22470609" " 62060111"
## Dose.2                  "   724988"          " 19660293" " 54937050"
## Population              "  1711947"          " 35998752" "128500364"
##                          [,6]         [,7]
## State.UTs                "Chandigarh" "Chhattisgarh"
## Total.Vaccination.Doses "  1989833"   " 35836458"
## Dose.1                  "  1086102"   " 18693176"
## Dose.2                  "   903731"   " 17143282"
## Population              "  1158040"   " 32199722"
##                          [,8]                                      [,9]
## State.UTs                "Dadra and Nagar Haveli and Daman and Diu" "Delhi"
## Total.Vaccination.Doses "  1313783"                                 "
30381117"
## Dose.1                  "   727305"                                 "
16492260"
## Dose.2                  "   586478"                                 "
13888857"
## Population              "   773997"                                 "
19301096"
##                          [,10]       [,11]       [,12]       [,13]
## State.UTs                "Goa"       "Gujarat"   "Haryana"   "Himachal
Pradesh"
## Total.Vaccination.Doses "  2558857" " 97449266" " 40194002" " 11714543"
## Dose.1                  "  1349270" " 49173252" " 21896657" "  6014888"
## Dose.2                  "  1209587" " 48276014" " 18297345" "  5699655"
## Population              "  1521992" " 70400153" " 28900667" "  7503010"
##                          [,14]               [,15]       [,16]       [,17]
## State.UTs                "Jammu and Kashmir" "Jharkhand" "Karnataka"
"Kerala"
## Total.Vaccination.Doses " 20123295"         " 36465011" " 98646532" "
50506342"
## Dose.1                  "  9930317"         " 21158905" " 49856747" "
26961645"
## Dose.2                  " 10192978"         " 15306106" " 48789785" "
23544697"
## Population              " 14999397"         " 40100376" " 69599762" "
34698876"
##                          [,18]       [,19]         [,20]
## State.UTs                "Ladakh"    "Lakshadweep" "Madhya Pradesh"
## Total.Vaccination.Doses "   402228" "   112063"   "106980921"
## Dose.1                  "   218773" "    56773"   " 53932164"
## Dose.2                  "   183455" "    55290"   " 53048757"
## Population              "   290492" "    66001"   " 85002417"
##                          [,21]         [,22]       [,23]       [,24]
## State.UTs                "Maharashtra" "Manipur"   "Meghalaya" "Mizoram"
## Total.Vaccination.Doses "153958201"   "  2616582" "  2330226" "  1428976"
## Dose.1                  " 84358055"   "  1447454" "  1324555" "   780905"
```

```
## Dose.2                   " 69600146"   "  1169128" "  1005671" "   648071"
## Population               "124904071"   "  3436948" "  3772103" "  1308967"
##                          [,25]       [,26]       [,27]        [,28]
## State.UTs                "Nagaland"  "Odisha"    "Puducherry" "Punjab"
## Total.Vaccination.Doses "  1502881" " 59927514" "  1554319"  " 40322736"
## Dose.1                   "   833294" " 31200338" "   892623"  " 22180522"
## Dose.2                   "   669587" " 28727176" "   661696"  " 18142214"
## Population               "  2073074" " 47099270" "  1646050"  " 30501026"
##                          [,29]       [,30]       [,31]        [,32]
## State.UTs                "Rajasthan" "Sikkim"    "Tamil Nadu" "Telangana"
## Total.Vaccination.Doses " 95514875" "  1041215" "102131951"  " 57832105"
## Dose.1                   " 50865992" "   538261" " 54996643"  " 29452703"
## Dose.2                   " 44648883" "   502954" " 47135308"  " 28379402"
## Population               " 79502477" "   658019" " 83697770"  " 38157311"
##                          [,33]       [,34]           [,35]         [,36]
## State.UTs                "Tripura"   "Uttar Pradesh" "Uttarakhand" "West
Bengal"
## Total.Vaccination.Doses "  4894221" "287827071"     " 15936251"
"126778443"
## Dose.1                   "  2644217" "153151985"     "  8140263"   "
67031629"
## Dose.2                   "  2250004" "134675086"     "  7795988"   "
59746814"
## Population               "  4184959" "231502578"     " 11700099"
"100896618"
```

23.Ordering the rows based on column name

```
with(data,order(State.UTs))
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
24 25
## [26] 26 27 28 29 30 31 32 33 34 35 36
```

24.Converting dataframe to a matrix

```
as.matrix(data)
```

```
##       State.UTs                                    Total.Vaccination.Doses
##  [1,] "Andaman and Nicobar"                        "   624118"
##  [2,] "Andhra Pradesh"                             " 84043879"
##  [3,] "Arunachal Pradesh"                          "  1576467"
##  [4,] "Assam"                                      " 42130902"
##  [5,] "Bihar"                                      "116997161"
##  [6,] "Chandigarh"                                 "  1989833"
##  [7,] "Chhattisgarh"                               " 35836458"
##  [8,] "Dadra and Nagar Haveli and Daman and Diu"   "  1313783"
##  [9,] "Delhi"                                      " 30381117"
## [10,] "Goa"                                        "  2558857"
## [11,] "Gujarat"                                    " 97449266"
## [12,] "Haryana"                                    " 40194002"
## [13,] "Himachal Pradesh"                           " 11714543"
```

```
## [14,] "Jammu and Kashmir"                          " 20123295"
## [15,] "Jharkhand"                                  " 36465011"
## [16,] "Karnataka"                                  " 98646532"
## [17,] "Kerala"                                     " 50506342"
## [18,] "Ladakh"                                     "   402228"
## [19,] "Lakshadweep"                                "   112063"
## [20,] "Madhya Pradesh"                             "106980921"
## [21,] "Maharashtra"                                "153958201"
## [22,] "Manipur"                                    "  2616582"
## [23,] "Meghalaya"                                  "  2330226"
## [24,] "Mizoram"                                    "  1428976"
## [25,] "Nagaland"                                   "  1502881"
## [26,] "Odisha"                                     " 59927514"
## [27,] "Puducherry"                                 "  1554319"
## [28,] "Punjab"                                     " 40322736"
## [29,] "Rajasthan"                                  " 95514875"
## [30,] "Sikkim"                                     "  1041215"
## [31,] "Tamil Nadu"                                 "102131951"
## [32,] "Telangana"                                  " 57832105"
## [33,] "Tripura"                                    "  4894221"
## [34,] "Uttar Pradesh"                              "287827071"
## [35,] "Uttarakhand"                                " 15936251"
## [36,] "West Bengal"                                "126778443"
##       Dose.1      Dose.2      Population
##  [1,] "   311046" "   313072" "   399001"
##  [2,] " 40596914" " 43446965" " 91702478"
##  [3,] "   851479" "   724988" "  1711947"
##  [4,] " 22470609" " 19660293" " 35998752"
##  [5,] " 62060111" " 54937050" "128500364"
##  [6,] "  1086102" "   903731" "  1158040"
##  [7,] " 18693176" " 17143282" " 32199722"
##  [8,] "   727305" "   586478" "   773997"
##  [9,] " 16492260" " 13888857" " 19301096"
## [10,] "  1349270" "  1209587" "  1521992"
## [11,] " 49173252" " 48276014" " 70400153"
## [12,] " 21896657" " 18297345" " 28900667"
## [13,] "  6014888" "  5699655" "  7503010"
## [14,] "  9930317" " 10192978" " 14999397"
## [15,] " 21158905" " 15306106" " 40100376"
## [16,] " 49856747" " 48789785" " 69599762"
## [17,] " 26961645" " 23544697" " 34698876"
## [18,] "   218773" "   183455" "   290492"
## [19,] "    56773" "    55290" "    66001"
## [20,] " 53932164" " 53048757" " 85002417"
## [21,] " 84358055" " 69600146" "124904071"
## [22,] "  1447454" "  1169128" "  3436948"
## [23,] "  1324555" "  1005671" "  3772103"
## [24,] "   780905" "   648071" "  1308967"
## [25,] "   833294" "   669587" "  2073074"
## [26,] " 31200338" " 28727176" " 47099270"
```

```
## [27,] "   892623" "   661696" "  1646050"
## [28,] " 22180522" " 18142214" " 30501026"
## [29,] " 50865992" " 44648883" " 79502477"
## [30,] "   538261" "   502954" "   658019"
## [31,] " 54996643" " 47135308" " 83697770"
## [32,] " 29452703" " 28379402" " 38157311"
## [33,] "  2644217" "  2250004" "  4184959"
## [34,] "153151985" "134675086" "231502578"
## [35,] "  8140263" "  7795988" " 11700099"
## [36,] " 67031629" " 59746814" "100896618"
```

25.Converting dataframe to list

```
as.list(data)
```

```
## $State.UTs
##  [1] "Andaman and Nicobar"
##  [2] "Andhra Pradesh"
##  [3] "Arunachal Pradesh"
##  [4] "Assam"
##  [5] "Bihar"
##  [6] "Chandigarh"
##  [7] "Chhattisgarh"
##  [8] "Dadra and Nagar Haveli and Daman and Diu"
##  [9] "Delhi"
## [10] "Goa"
## [11] "Gujarat"
## [12] "Haryana"
## [13] "Himachal Pradesh"
## [14] "Jammu and Kashmir"
## [15] "Jharkhand"
## [16] "Karnataka"
## [17] "Kerala"
## [18] "Ladakh"
## [19] "Lakshadweep"
## [20] "Madhya Pradesh"
## [21] "Maharashtra"
## [22] "Manipur"
## [23] "Meghalaya"
## [24] "Mizoram"
## [25] "Nagaland"
## [26] "Odisha"
## [27] "Puducherry"
## [28] "Punjab"
## [29] "Rajasthan"
## [30] "Sikkim"
## [31] "Tamil Nadu"
## [32] "Telangana"
## [33] "Tripura"
## [34] "Uttar Pradesh"
## [35] "Uttarakhand"
```

```
## [36] "West Bengal"
##
## $Total.Vaccination.Doses
##  [1]    624118  84043879   1576467  42130902 116997161   1989833  35836458
##  [8]   1313783  30381117   2558857  97449266  40194002  11714543  20123295
## [15]  36465011  98646532  50506342    402228    112063 106980921 153958201
## [22]   2616582   2330226   1428976   1502881  59927514   1554319  40322736
## [29]  95514875   1041215 102131951  57832105   4894221 287827071  15936251
## [36] 126778443
##
## $Dose.1
##  [1]    311046  40596914    851479  22470609  62060111   1086102  18693176
##  [8]    727305  16492260   1349270  49173252  21896657   6014888   9930317
## [15]  21158905  49856747  26961645    218773     56773  53932164  84358055
## [22]   1447454   1324555    780905    833294  31200338    892623  22180522
## [29]  50865992    538261  54996643  29452703   2644217 153151985   8140263
## [36]  67031629
##
## $Dose.2
##  [1]    313072  43446965    724988  19660293  54937050    903731  17143282
##  [8]    586478  13888857   1209587  48276014  18297345   5699655  10192978
## [15]  15306106  48789785  23544697    183455     55290  53048757  69600146
## [22]   1169128   1005671    648071    669587  28727176    661696  18142214
## [29]  44648883    502954  47135308  28379402   2250004 134675086   7795988
## [36]  59746814
##
## $Population
##  [1]    399001  91702478   1711947  35998752 128500364   1158040  32199722
##  [8]    773997  19301096   1521992  70400153  28900667   7503010  14999397
## [15]  40100376  69599762  34698876    290492     66001  85002417 124904071
## [22]   3436948   3772103   1308967   2073074  47099270   1646050  30501026
## [29]  79502477    658019  83697770  38157311   4184959 231502578  11700099
## [36] 100896618
```

26.To get the 20% quantile of a given column

```
quantile(data$Population,0.2)
```

```
##     20%
## 1521992
```

27.To pick 20%,50%, and 80% quantile

```
quantile(data$Population,c(0.2,0.5,0.8))
```

```
##      20%      50%      80%
##  1521992 24100882 79502477
```

28.Trying quantiles for non numeric order data

```
quantile(data$Population, c(0.5,0.75,0.25))
```

```
##      50%      75%      25%
## 24100882 69799860  1695473
```

29.To determine the cumulative sum of data

```
cumsum(data$Dose.1)
```

```
##  [1]    311046  40907960  41759439  64230048 126290159 127376261 146069437
##  [8] 146796742 163289002 164638272 213811524 235708181 241723069 251653386
## [15] 272812291 322669038 349630683 349849456 349906229 403838393 488196448
## [22] 489643902 490968457 491749362 492582656 523782994 524675617 546856139
## [29] 597722131 598260392 653257035 682709738 685353955 838505940 846646203
## [36] 913677832
```

30.To get the cumulative maximum value

```
cummax(data$Dose.1)
```

```
##  [1]    311046  40596914  40596914  40596914  62060111  62060111  62060111
##  [8]  62060111  62060111  62060111  62060111  62060111  62060111  62060111
## [15]  62060111  62060111  62060111  62060111  62060111  62060111  84358055
## [22]  84358055  84358055  84358055  84358055  84358055  84358055  84358055
## [29]  84358055  84358055  84358055  84358055  84358055 153151985 153151985
## [36] 153151985
```

31.To get the minimum cumulative sum

```
cummin(data$Dose.1)
```

```
##  [1] 311046 311046 311046 311046 311046 311046 311046 311046 311046 311046
## [11] 311046 311046 311046 311046 311046 311046 311046 218773  56773  56773
## [21]  56773  56773  56773  56773  56773  56773  56773  56773  56773  56773
## [31]  56773  56773  56773  56773  56773  56773
```

32.To get the cumulative product of the sample

```
cumprod(data$Dose.2)
```

```
##  [1]  3.130720e+05  1.360203e+13  9.861307e+18  1.938762e+26  1.065099e+34
##  [6]  9.625626e+39  1.650148e+47  9.677756e+52  1.344130e+60  1.625842e+66
## [11]  7.848916e+73  1.436143e+81  8.185521e+87  8.343484e+94 1.277063e+102
## [16] 6.230761e+109 1.467014e+117 2.691310e+122 1.488025e+127 7.893789e+134
## [21] 5.494089e+142 6.423293e+148 6.459720e+154 4.186357e+160 2.803130e+166
## [26] 8.052601e+173 5.328374e+179 9.666850e+186 4.316141e+194 2.170820e+200
## [31] 1.023223e+208 2.903845e+215 6.533663e+221 8.799216e+229 6.859859e+236
## [36] 4.098547e+244
```

33.Trying cumulative command on a vector of character data

```
cummax("State.UTs")
```

```
## Warning: NAs introduced by coercion
```

```
## [1] NA
```

34.To generate sequences of values

```
seq(data$Dose.2)
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
24 25
## [26] 26 27 28 29 30 31 32 33 34 35 36
```

35.By using along we are generating a vector

```
seq(along=data$Dose.2)
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
24 25
## [26] 26 27 28 29 30 31 32 33 34 35 36
```

36.To get the mean of column 2

```
mean(data[,2])
```

```
## [1] 48212343
```

37.To get the sum of each column

```
colSums(data[,c(2,3)])
```

```
## Total.Vaccination.Doses                  Dose.1
##              1735644345               913677832
```

38.To get the mean of each row

```
rowMeans(data[,c(3,4)])
```

```
##  [1]    312059.0  42021939.5    788233.5  21065451.0  58498580.5
994916.5
##  [7]  17918229.0    656891.5  15190558.5   1279428.5  48724633.0
20097001.0
## [13]   5857271.5  10061647.5  18232505.5  49323266.0  25253171.0
201114.0
## [19]     56031.5  53490460.5  76979100.5   1308291.0   1165113.0
714488.0
## [25]    751440.5  29963757.0    777159.5  20161368.0  47757437.5
520607.5
## [31]  51065975.5  28916052.5   2447110.5 143913535.5   7968125.5
63389221.5
```

39. Apply command

```
apply(data[,c(3,4)],2,median)
```

```
##   Dose.1   Dose.2
## 17592718 14597482
```

40.Max of a column

```
max(data$Population)
```

```
## [1] 231502578
```

41.lapply which is used to apply for all columns

```
lapply(data,mean,na.rm=TRUE)
```

```
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA

## $State.UTs
## [1] NA
##
## $Total.Vaccination.Doses
## [1] 48212343
##
## $Dose.1
## [1] 25379940
##
## $Dose.2
## [1] 22832403
##
## $Population
## [1] 39718608
```

42.sapply same as lapply in a prettier form

```
sapply(data,mean,na.rm=TRUE)
```

```
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA

##               State.UTs Total.Vaccination.Doses                  Dose.1
##                      NA                48212343                25379940
##                  Dose.2              Population
##                22832403                39718608
```

43.Creating list of column names

```
colnames=names(data)
colnames
```

```
## [1] "State.UTs"               "Total.Vaccination.Doses"
## [3] "Dose.1"                  "Dose.2"
## [5] "Population"
```

44.Creating contingency table of dataframe

```
data.tab = with(data,table(colnames[1],colnames[2]))
data.tab
```

```
##
##             Total.Vaccination.Doses
##   State.UTs                       1
```

45.Display the table as proportions of the row tables

```
prop.table(data$Dose.2)
```

```
##  [1] 3.808817e-04 5.285734e-02 8.820165e-04 2.391861e-02 6.683612e-02
##  [6] 1.099474e-03 2.085642e-02 7.135060e-04 1.689711e-02 1.471577e-03
## [11] 5.873234e-02 2.226045e-02 6.934169e-03 1.240072e-02 1.862133e-02
```

```
## [16] 5.935739e-02 2.864435e-02 2.231904e-04 6.726551e-05 6.453883e-02
## [21] 8.467516e-02 1.422355e-03 1.223494e-03 7.884397e-04 8.146159e-04
## [26] 3.494933e-02 8.050158e-04 2.207172e-02 5.431959e-02 6.118911e-04
## [31] 5.734456e-02 3.452623e-02 2.737343e-03 1.638450e-01 9.484557e-03
## [36] 7.268765e-02
```

46.Create a stem and leaf plot for the dataframe

```
stem(data$Population)
```

```
##
##   The decimal point is 7 digit(s) to the right of the |
##
##    0 | 000111122223448259
##    2 | 912568
##    4 | 07
##    6 | 00
##    8 | 0452
##   10 | 1
##   12 | 59
##   14 |
##   16 |
##   18 |
##   20 |
##   22 | 2
```

47.Loading dplyr package to use predefined functions to perform on dataset

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.1.3
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

48.Adding a column to the dataset

```
data %>% mutate(newcol = NA)
```

```
##                                   State.UTs Total.Vaccination.Doses
Dose.1
## 1                       Andaman and Nicobar                  624118
311046
## 2                            Andhra Pradesh                84043879
40596914
## 3                         Arunachal Pradesh                 1576467
```

```
851479
## 4                                    Assam                42130902
22470609
## 5                                    Bihar               116997161
62060111
## 6                                Chandigarh                1989833
1086102
## 7                              Chhattisgarh                35836458
18693176
## 8  Dadra and Nagar Haveli and Daman and Diu                1313783
727305
## 9                                    Delhi                30381117
16492260
## 10                                     Goa                 2558857
1349270
## 11                                 Gujarat                97449266
49173252
## 12                                 Haryana                40194002
21896657
## 13                          Himachal Pradesh                11714543
6014888
## 14                         Jammu and Kashmir                20123295
9930317
## 15                                 Jharkhand                36465011
21158905
## 16                                 Karnataka                98646532
49856747
## 17                                    Kerala                50506342
26961645
## 18                                    Ladakh                  402228
218773
## 19                                Lakshadweep                  112063
56773
## 20                             Madhya Pradesh               106980921
53932164
## 21                                Maharashtra               153958201
84358055
## 22                                   Manipur                 2616582
1447454
## 23                                 Meghalaya                 2330226
1324555
## 24                                   Mizoram                 1428976
780905
## 25                                  Nagaland                 1502881
833294
## 26                                    Odisha                59927514
31200338
## 27                               Puducherry                 1554319
892623
## 28                                    Punjab                40322736
```

```
22180522
## 29                                  Rajasthan                 95514875
50865992
## 30                                     Sikkim                  1041215
538261
## 31                                 Tamil Nadu                102131951
54996643
## 32                                  Telangana                 57832105
29452703
## 33                                    Tripura                  4894221
2644217
## 34                               Uttar Pradesh                287827071
153151985
## 35                                Uttarakhand                 15936251
8140263
## 36                                West Bengal                126778443
67031629
##       Dose.2 Population newcol
## 1     313072     399001     NA
## 2   43446965   91702478     NA
## 3     724988    1711947     NA
## 4   19660293   35998752     NA
## 5   54937050  128500364     NA
## 6     903731    1158040     NA
## 7   17143282   32199722     NA
## 8     586478     773997     NA
## 9   13888857   19301096     NA
## 10   1209587    1521992     NA
## 11  48276014   70400153     NA
## 12  18297345   28900667     NA
## 13   5699655    7503010     NA
## 14  10192978   14999397     NA
## 15  15306106   40100376     NA
## 16  48789785   69599762     NA
## 17  23544697   34698876     NA
## 18    183455     290492     NA
## 19     55290      66001     NA
## 20  53048757   85002417     NA
## 21  69600146  124904071     NA
## 22   1169128    3436948     NA
## 23   1005671    3772103     NA
## 24    648071    1308967     NA
## 25    669587    2073074     NA
## 26  28727176   47099270     NA
## 27    661696    1646050     NA
## 28  18142214   30501026     NA
## 29  44648883   79502477     NA
## 30    502954     658019     NA
## 31  47135308   83697770     NA
## 32  28379402   38157311     NA
```

```
## 33   2250004    4184959     NA
## 34 134675086  231502578     NA
## 35   7795988   11700099     NA
## 36  59746814  100896618     NA
```

49.Renaming the first column as states

```
data = data %>% rename(states = "State.UTs")
data
```

```
##                                     states Total.Vaccination.Doses
Dose.1
## 1                       Andaman and Nicobar                  624118
311046
## 2                           Andhra Pradesh                84043879
40596914
## 3                        Arunachal Pradesh                 1576467
851479
## 4                                    Assam                42130902
22470609
## 5                                    Bihar               116997161
62060111
## 6                                Chandigarh                 1989833
1086102
## 7                              Chhattisgarh                35836458
18693176
## 8  Dadra and Nagar Haveli and Daman and Diu                 1313783
727305
## 9                                    Delhi                30381117
16492260
## 10                                     Goa                 2558857
1349270
## 11                                 Gujarat                97449266
49173252
## 12                                 Haryana                40194002
21896657
## 13                          Himachal Pradesh                11714543
6014888
## 14                          Jammu and Kashmir                20123295
9930317
## 15                                Jharkhand                36465011
21158905
## 16                                Karnataka                98646532
49856747
## 17                                   Kerala                50506342
26961645
## 18                                   Ladakh                  402228
218773
## 19                               Lakshadweep                  112063
56773
## 20                          Madhya Pradesh               106980921
```

```
53932164
## 21                              Maharashtra                153958201
84358055
## 22                                  Manipur                  2616582
1447454
## 23                                Meghalaya                  2330226
1324555
## 24                                  Mizoram                  1428976
780905
## 25                                 Nagaland                  1502881
833294
## 26                                    Odisha                 59927514
31200338
## 27                               Puducherry                  1554319
892623
## 28                                    Punjab                 40322736
22180522
## 29                                Rajasthan                  95514875
50865992
## 30                                    Sikkim                  1041215
538261
## 31                               Tamil Nadu                 102131951
54996643
## 32                                Telangana                  57832105
29452703
## 33                                  Tripura                  4894221
2644217
## 34                            Uttar Pradesh                 287827071
153151985
## 35                              Uttarakhand                  15936251
8140263
## 36                              West Bengal                 126778443
67031629
##       Dose.2 Population
## 1     313072     399001
## 2   43446965   91702478
## 3     724988    1711947
## 4   19660293   35998752
## 5   54937050  128500364
## 6     903731    1158040
## 7   17143282   32199722
## 8     586478     773997
## 9   13888857   19301096
## 10   1209587    1521992
## 11  48276014   70400153
## 12  18297345   28900667
## 13   5699655    7503010
## 14  10192978   14999397
## 15  15306106   40100376
## 16  48789785   69599762
```

```
## 17  23544697   34698876
## 18    183455     290492
## 19     55290      66001
## 20  53048757   85002417
## 21  69600146  124904071
## 22   1169128    3436948
## 23   1005671    3772103
## 24    648071    1308967
## 25    669587    2073074
## 26  28727176   47099270
## 27    661696    1646050
## 28  18142214   30501026
## 29  44648883   79502477
## 30    502954     658019
## 31  47135308   83697770
## 32  28379402   38157311
## 33   2250004    4184959
## 34 134675086  231502578
## 35   7795988   11700099
## 36  59746814  100896618
```

50.Getting maximum value in Dose.1 column

```
max(data$Dose.1)
```

```
## [1] 153151985
```

51.Reordering columns in a dataset

```
data %>% select(Dose.2,Dose.1,Population)
```

```
##       Dose.2    Dose.1 Population
## 1     313072    311046     399001
## 2   43446965  40596914   91702478
## 3     724988    851479    1711947
## 4   19660293  22470609   35998752
## 5   54937050  62060111  128500364
## 6     903731   1086102    1158040
## 7   17143282  18693176   32199722
## 8     586478    727305     773997
## 9   13888857  16492260   19301096
## 10   1209587   1349270    1521992
## 11  48276014  49173252   70400153
## 12  18297345  21896657   28900667
## 13   5699655   6014888    7503010
## 14  10192978   9930317   14999397
## 15  15306106  21158905   40100376
## 16  48789785  49856747   69599762
## 17  23544697  26961645   34698876
## 18    183455    218773     290492
## 19     55290     56773      66001
## 20  53048757  53932164   85002417
```

```
## 21  69600146  84358055  124904071
## 22   1169128   1447454    3436948
## 23   1005671   1324555    3772103
## 24    648071    780905    1308967
## 25    669587    833294    2073074
## 26  28727176  31200338   47099270
## 27    661696    892623    1646050
## 28  18142214  22180522   30501026
## 29  44648883  50865992   79502477
## 30    502954    538261     658019
## 31  47135308  54996643   83697770
## 32  28379402  29452703   38157311
## 33   2250004   2644217    4184959
## 34 134675086 153151985  231502578
## 35   7795988   8140263   11700099
## 36  59746814  67031629  100896618
```

52.Getting a subset of a dataframe

```
data[data$bookft == "Population" &
 data$reviews >= 1 &
 data$reviews <= 20,
 c("Dose.1", "Dose.2")]
```

```
## [1] Dose.1 Dose.2
## <0 rows> (or 0-length row.names)
```

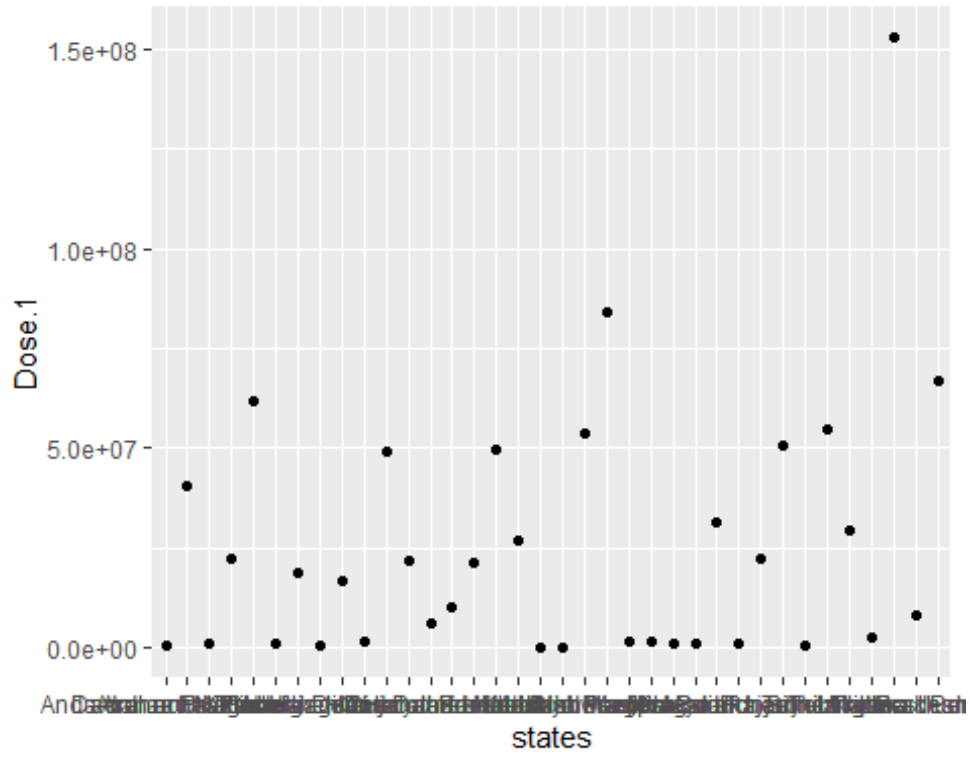
53.Loading ggplot2 library

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.1.3
```

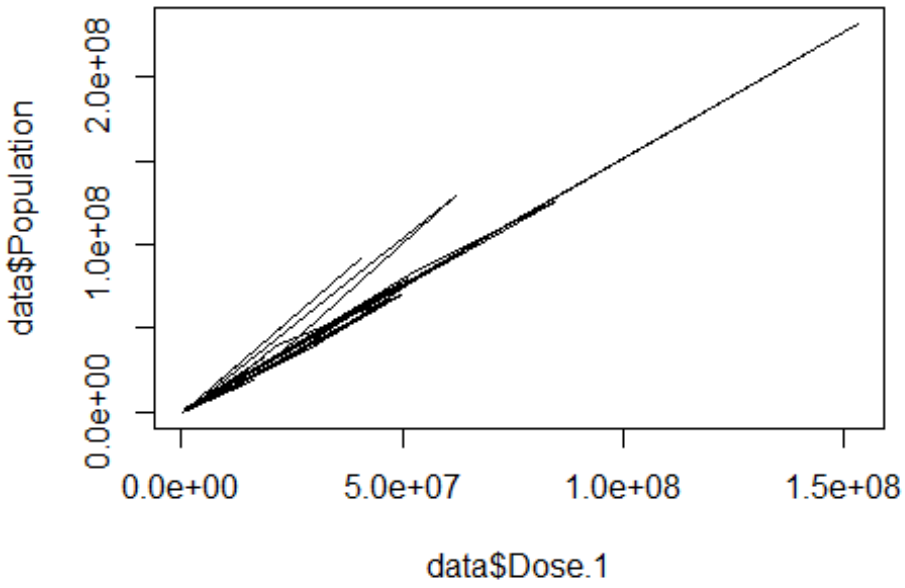
54.To make a scatter plot where x is states and y is Dose.1

```
ggplot(data,aes(x=states,y=Dose.1))+geom_point()
```



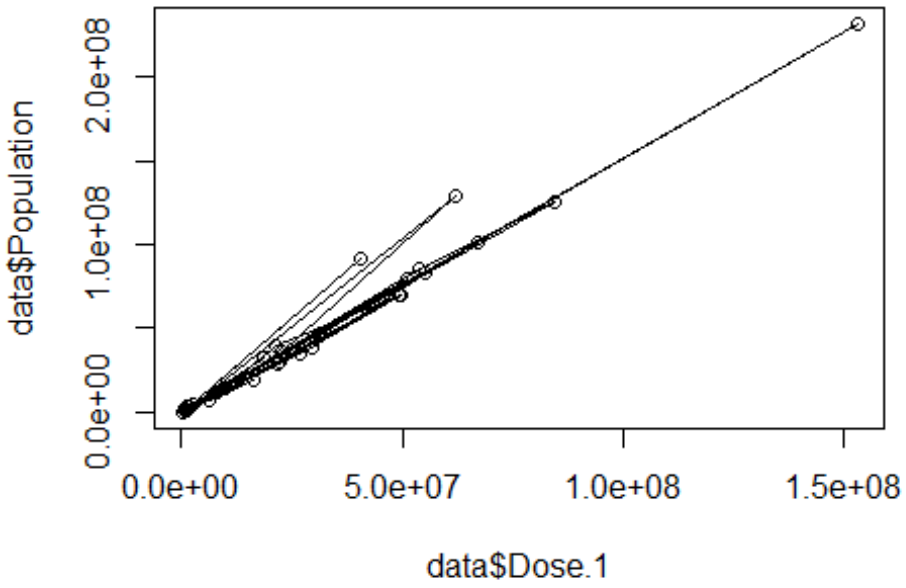
55.To make line graph

```
plot(data$Dose.1,data$Population,type="l")
```



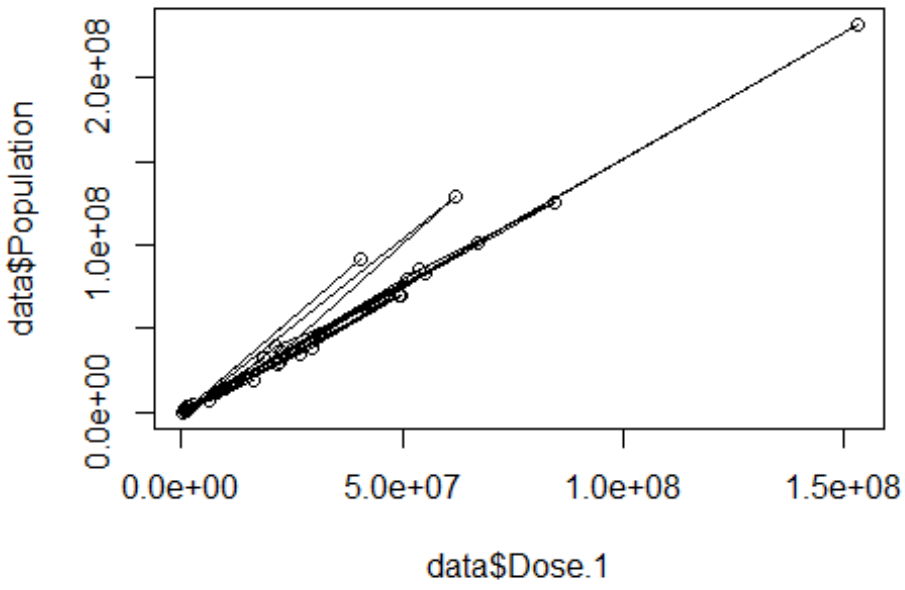
56.Plotting line graphs along with points

```
plot(data$Dose.1,data$Population,type="l")
points(data$Dose.1,data$Population)
```

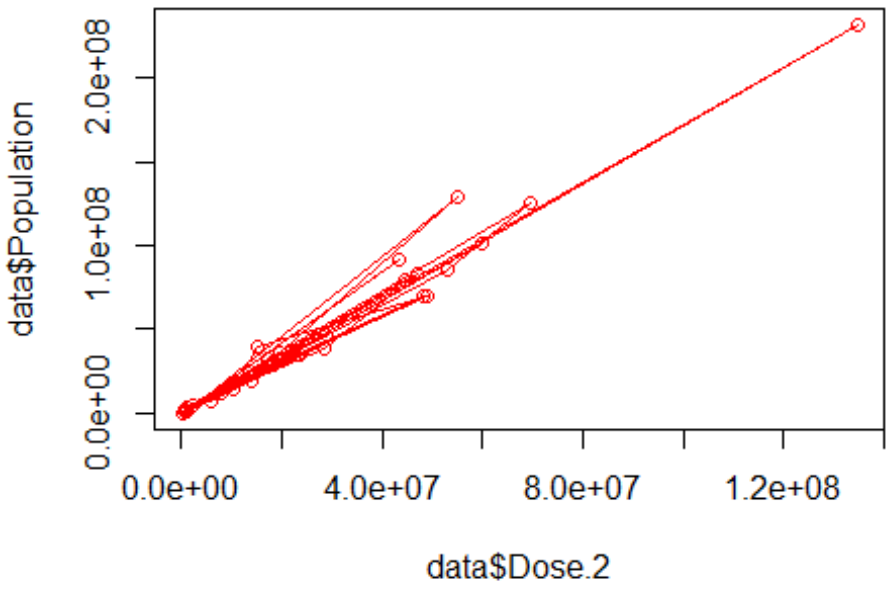


57.Comparing dose1 and dose2 data by plotting line point graph

```
plot(data$Dose.1,data$Population,type="l")
points(data$Dose.1,data$Population)
```

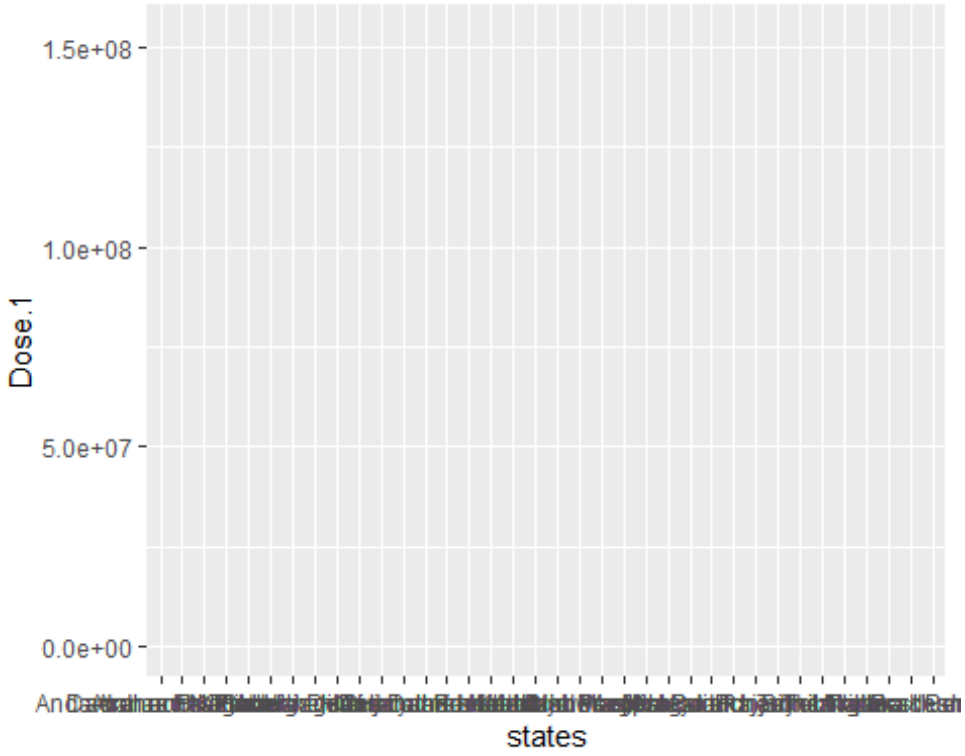



```
plot(data$Dose.2,data$Population,type="l",col="red")
points(data$Dose.2,data$Population,col="red")
```

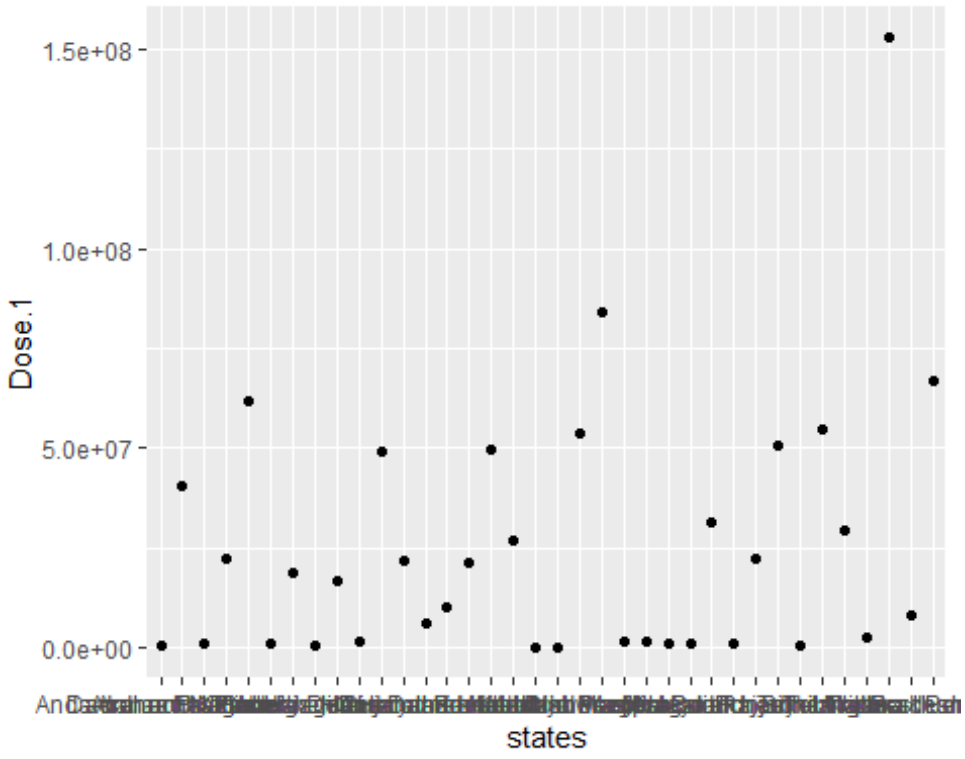


58.Comparing geom_line and geom_point

```
ggplot(data,aes(x=states,y=Dose.1))+geom_line()

## geom_path: Each group consists of only one observation. Do you need to
adjust
## the group aesthetic?
```

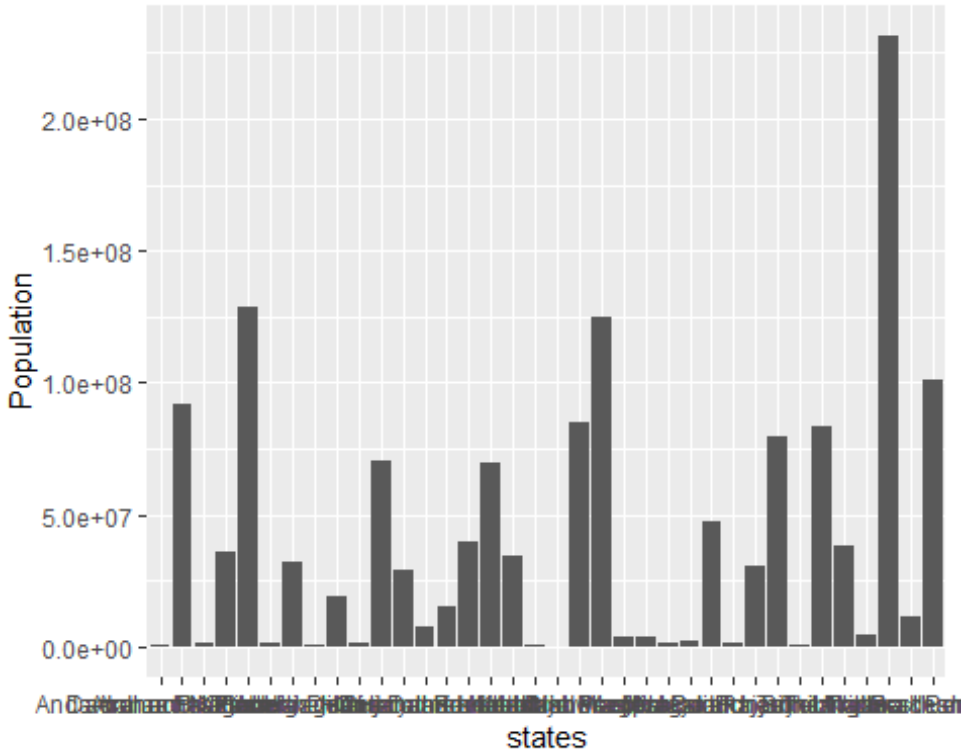



```
ggplot(data,aes(x=states,y=Dose.1))+geom_line()+geom_point()

## geom_path: Each group consists of only one observation. Do you need to
adjust
## the group aesthetic?
```



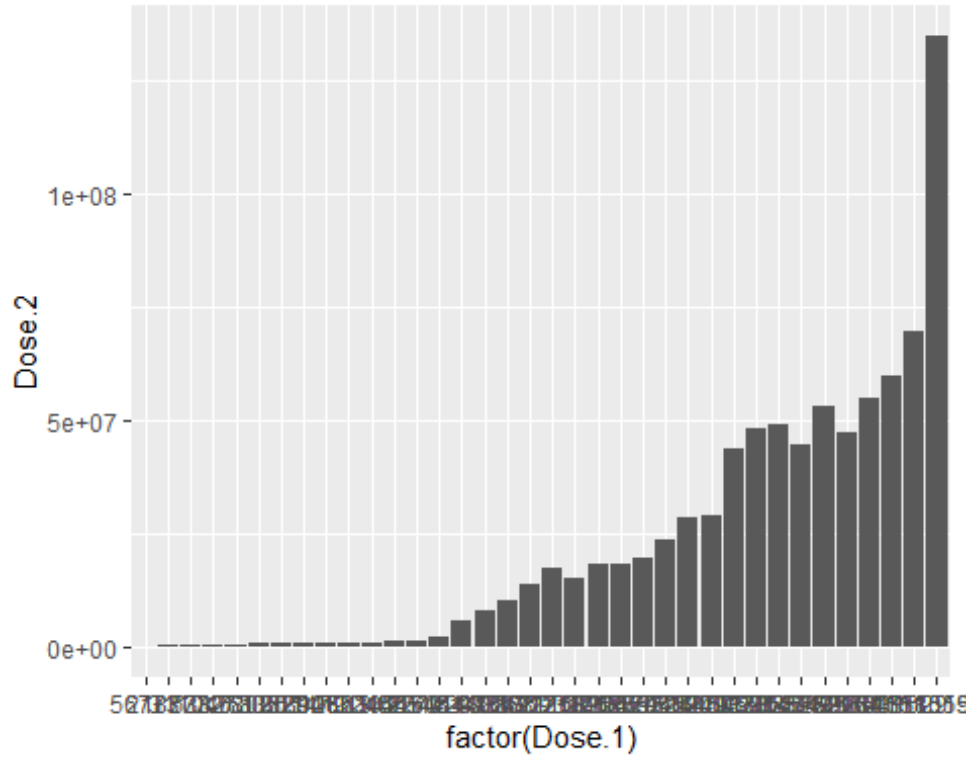
59.Bar graph of data frame

```
ggplot(data,aes(x=states,y=Population))+geom_col()
```



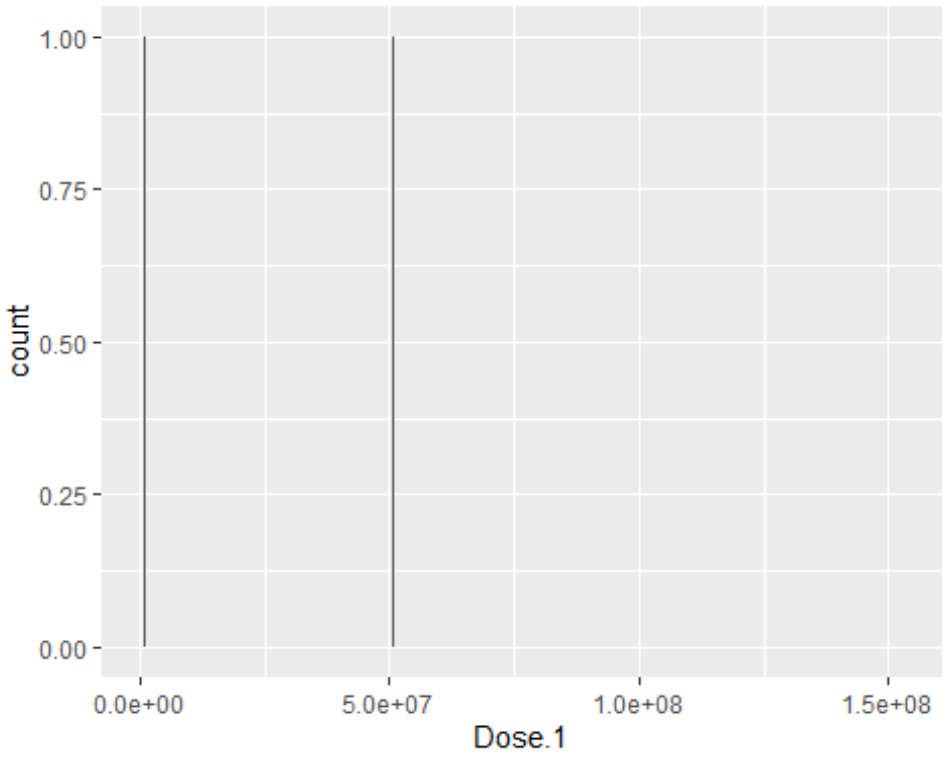
60.Converting x coordinate into factor and then plotting bar graph so that it is discrete

```
ggplot(data,aes(x=factor(Dose.1),y=Dose.2))+geom_col()
```



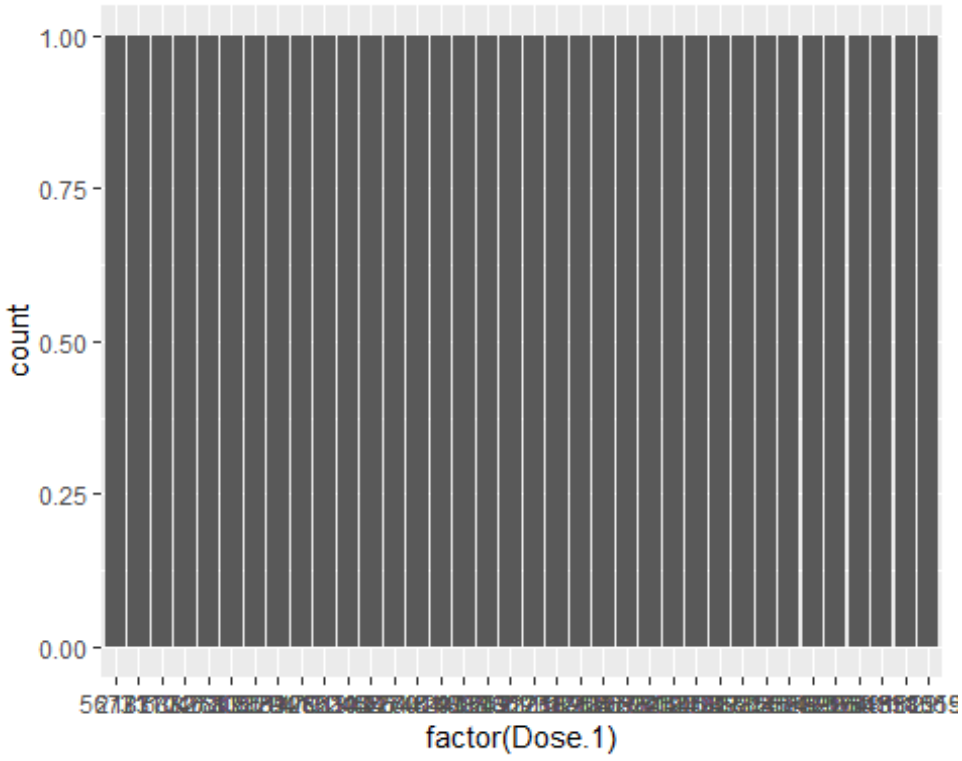
61.To plot bar graph

```
ggplot(data,aes(x=Dose.1))+geom_bar()
```



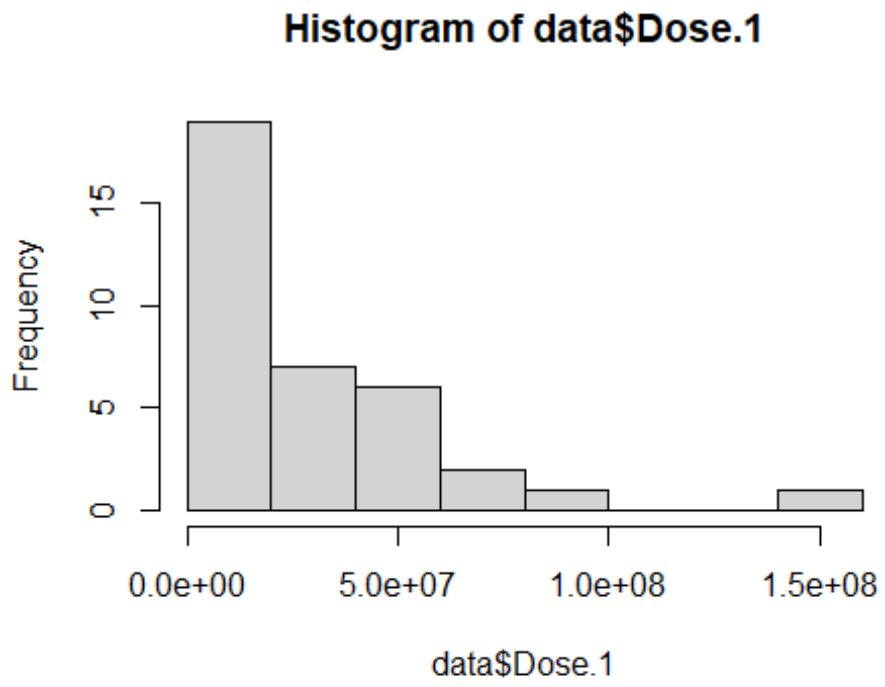
62.To plot bar graph using factor

```
ggplot(data,aes(x=factor(Dose.1)))+geom_bar()
```



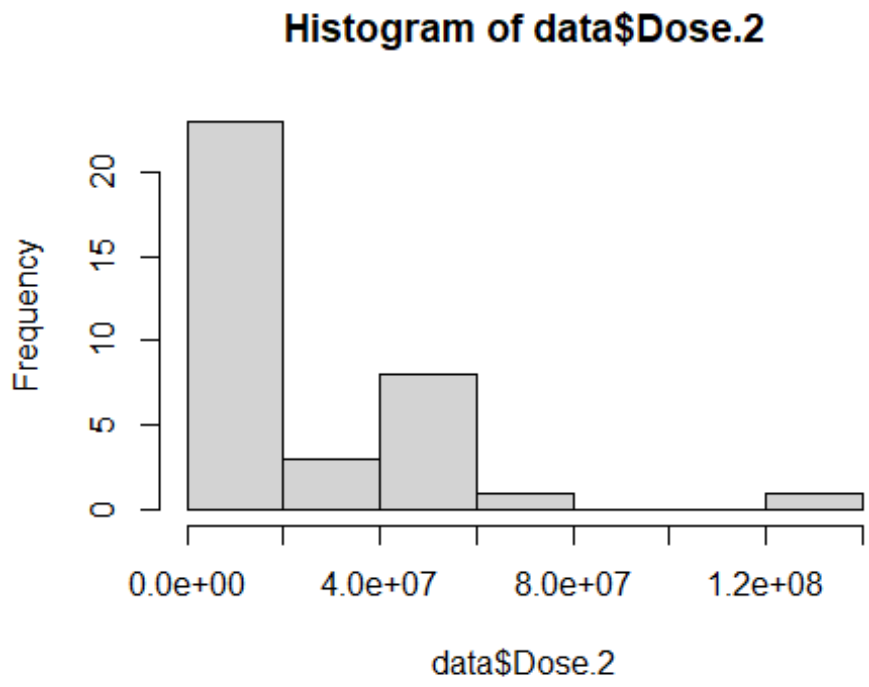
63.Plotting Histogram

```
hist(data$Dose.1)
```



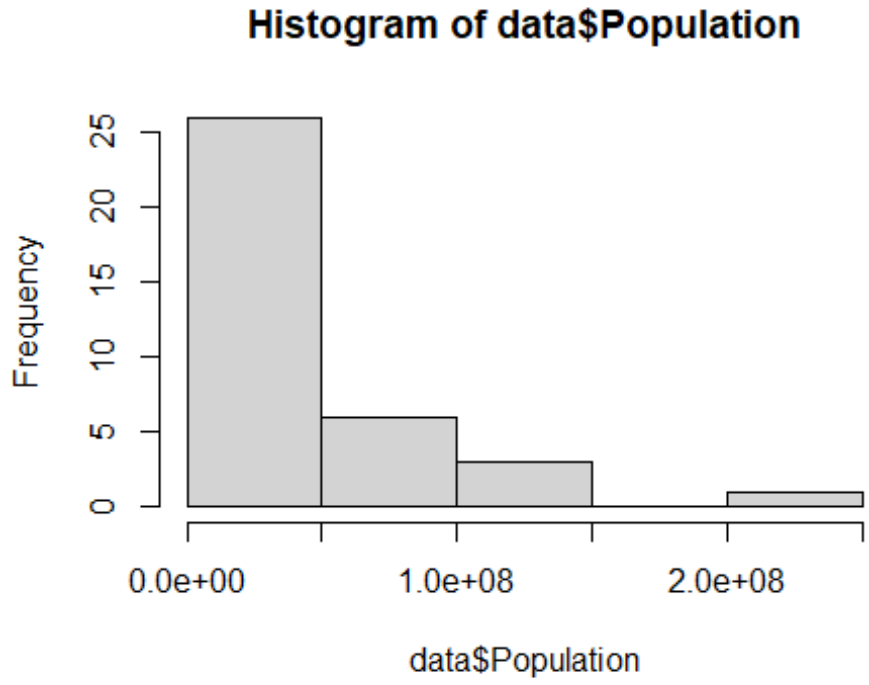
64.Plotting histogram of dose2

```
hist(data$Dose.2)
```



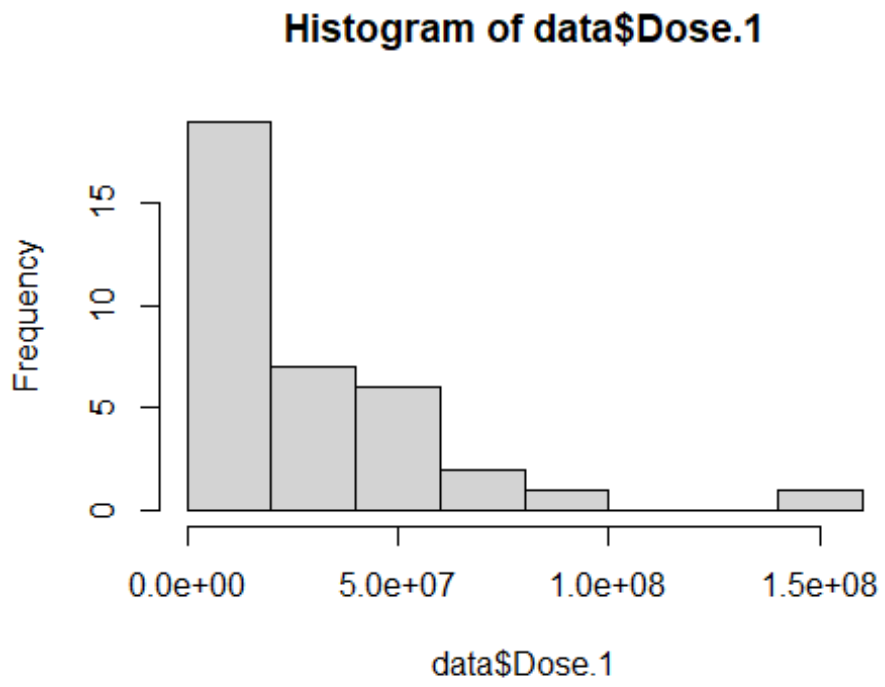
65.Plotting histogram of population

```
hist(data$Population)
```



66.Plotting histogram with specifying approximate number of bins with breaks

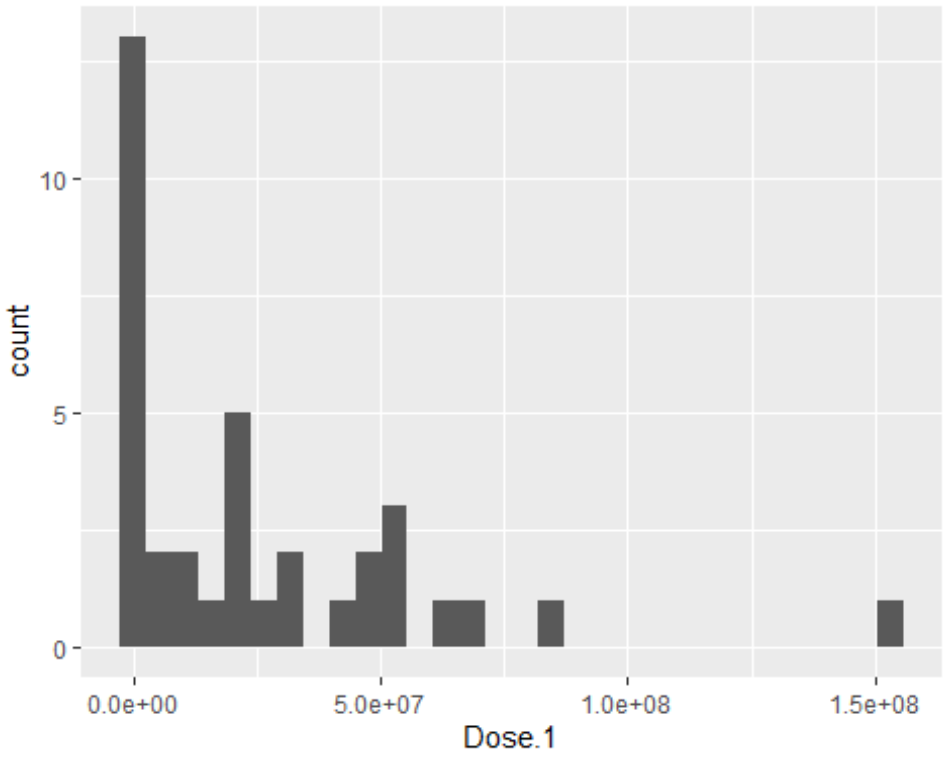
```
hist(data$Dose.1,breaks=10)
```



67.Using ggplot to plot histogram

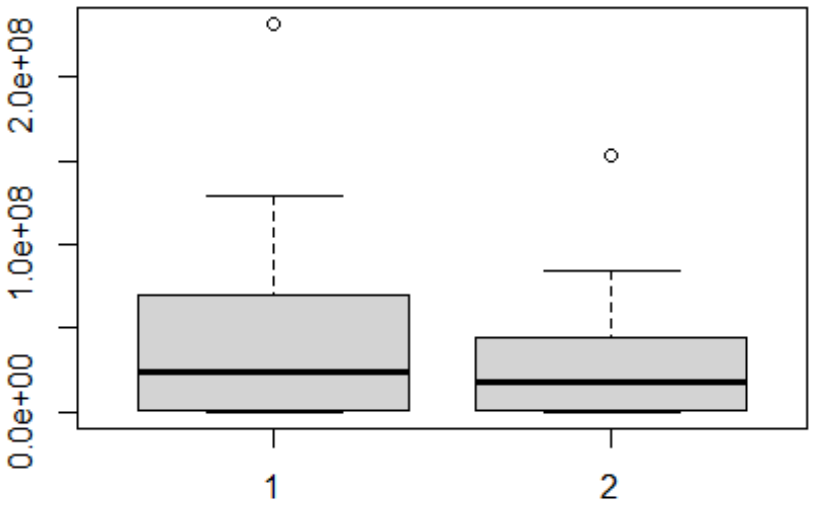
```
ggplot(data,aes(x=Dose.1))+geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



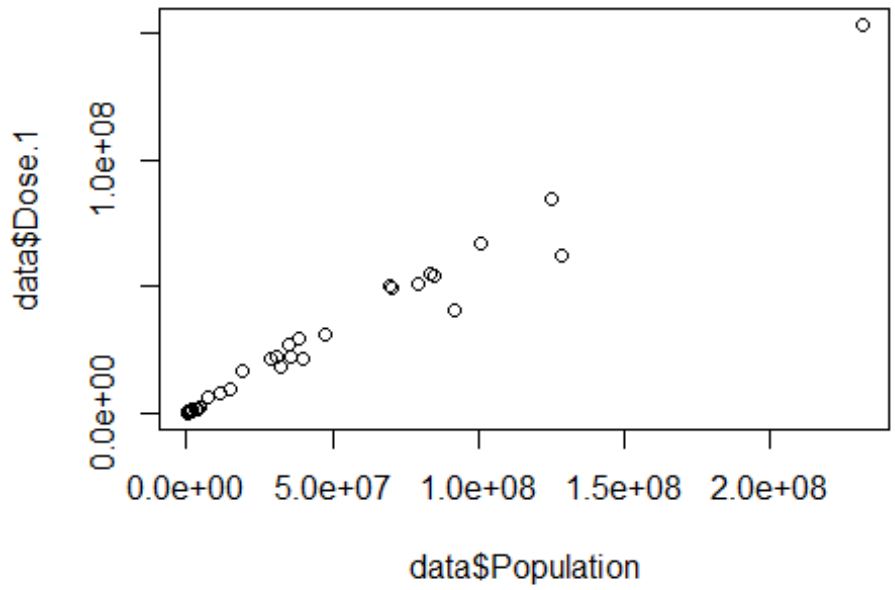
69.Creating boxplot

```
boxplot(data$Population,data$Dose.1)
```



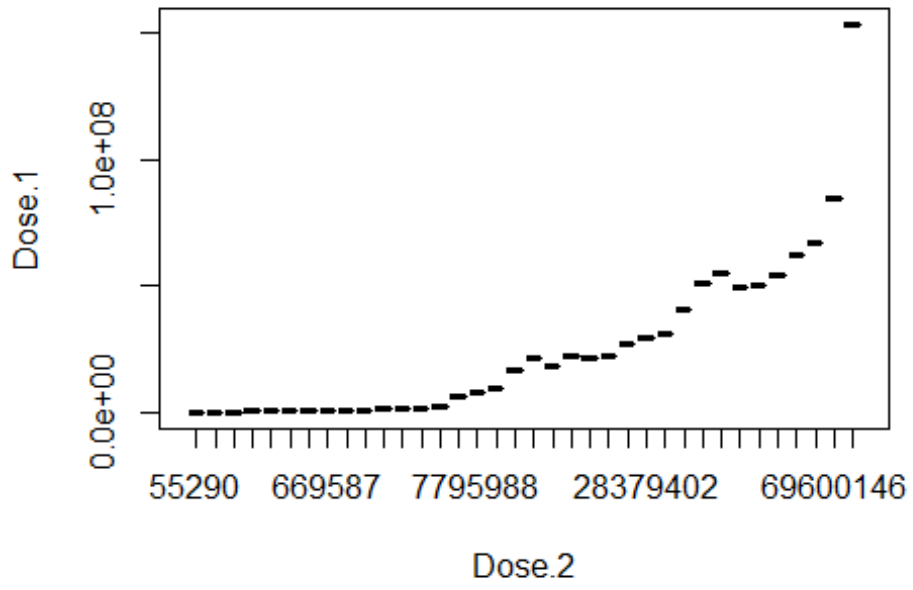
70.Plotting for population and Dose1

```
plot(data$Population,data$Dose.1)
```



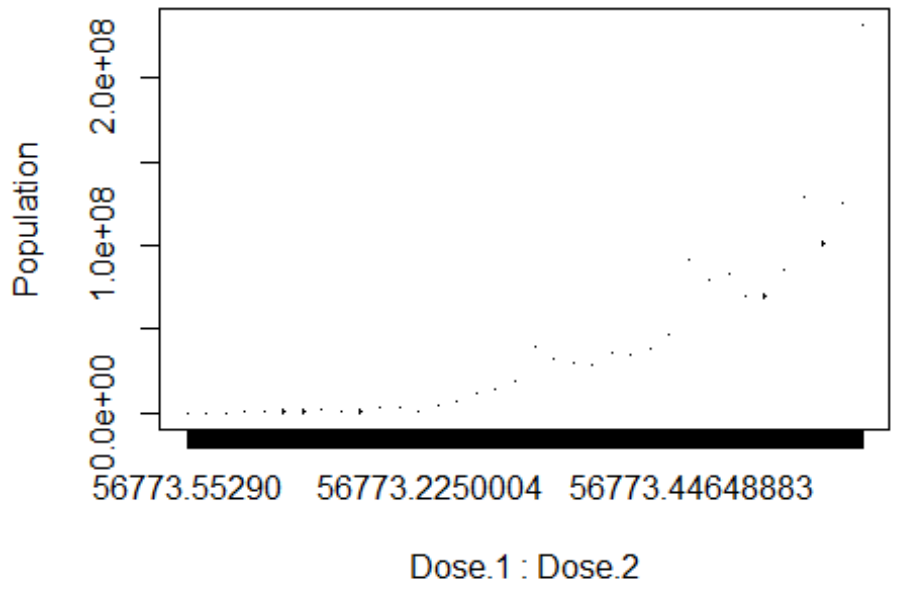
71.Plotting box plot with 2 variables in x-axis

```
boxplot(Dose.1~Dose.2,data=data)
```



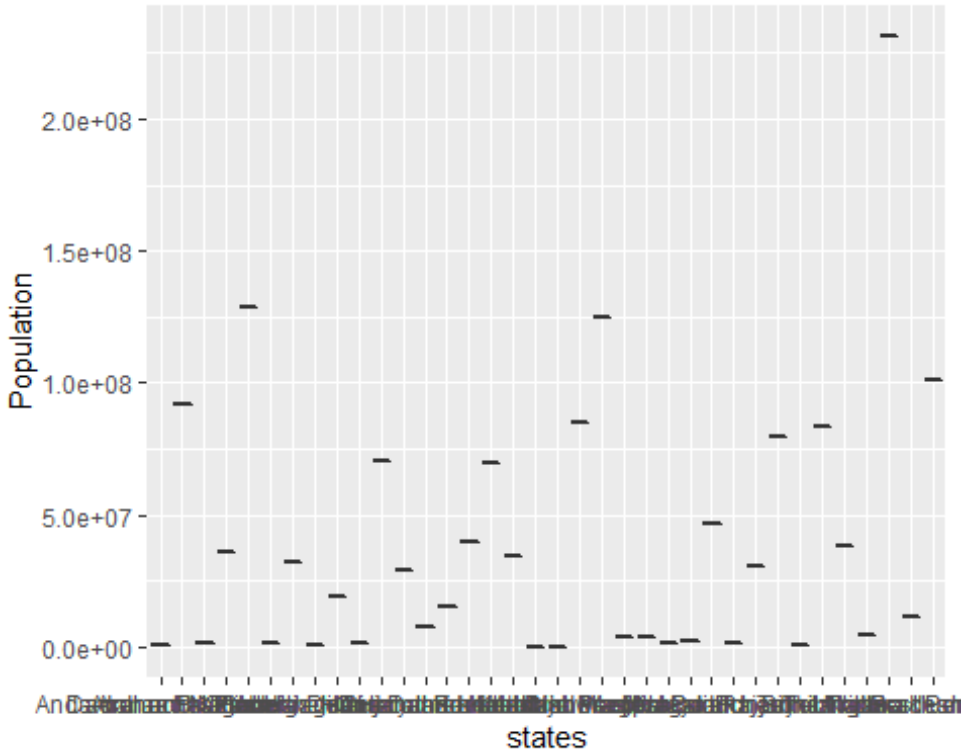
72.Plotting boxplot by putting interaction of two variables on x-axis

```
boxplot(Population~Dose.1+Dose.2,data=data)
```



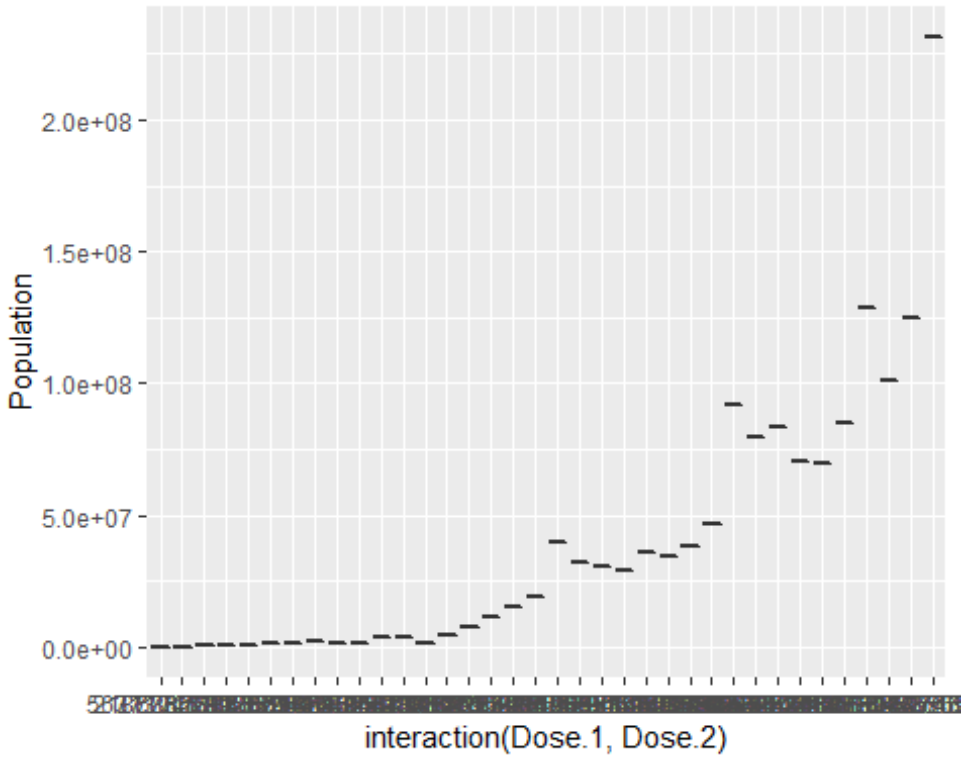
73.Plotting boxplot using ggplot2 package

```
ggplot(data,aes(x=states,y=Population))+geom_boxplot()
```



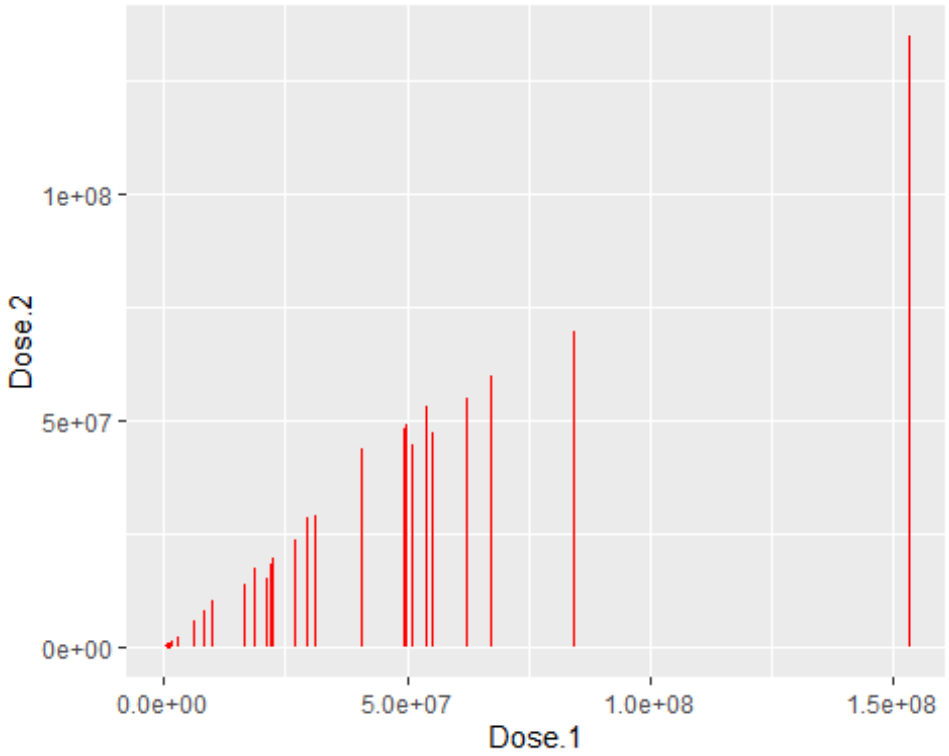
74.Plotting boxplot using ggplot for multiple variables

```
ggplot(data,aes(x=interaction(Dose.1,Dose.2),y=Population,states))+geom_boxpl
ot()
```



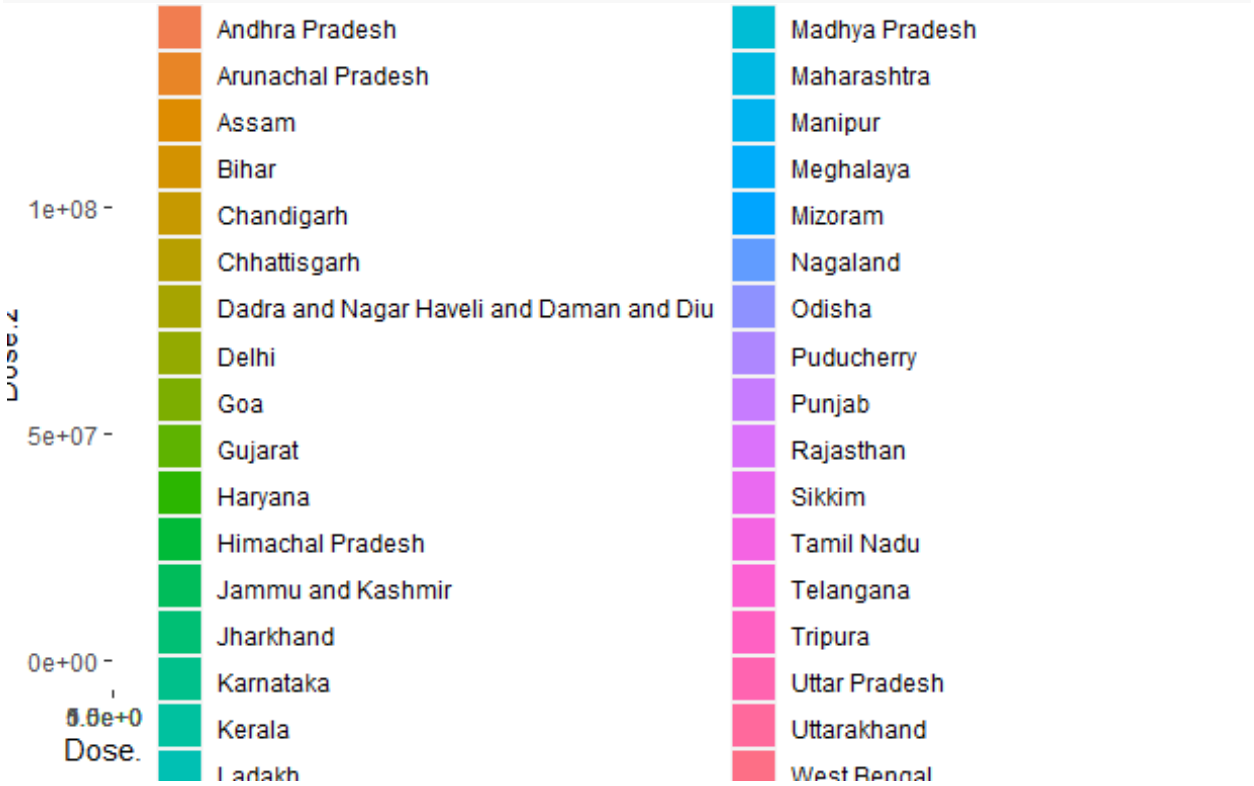
75.Plotting a boxplot using color fill

```
ggplot(data,aes(x=Dose.1,y=Dose.2))+geom_col(fill="lightblue",colour="red")
```



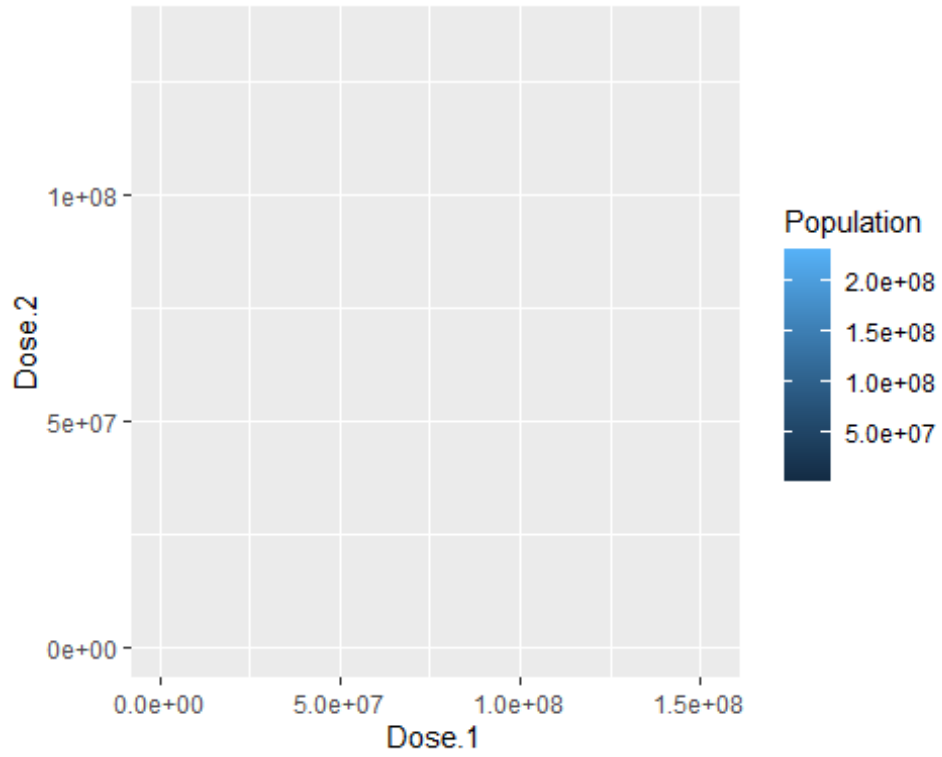
76.Using mapcultivator to fill color into bargraph

```
ggplot(data,aes(x=Dose.1,y=Dose.2,fill = states))+geom_col(position="dodge")
```



77.Using mapcultivator to fill color into bargraph for population

```
ggplot(data,aes(x=Dose.1,y=Dose.2,fill =
Population))+geom_col(position="dodge")
```

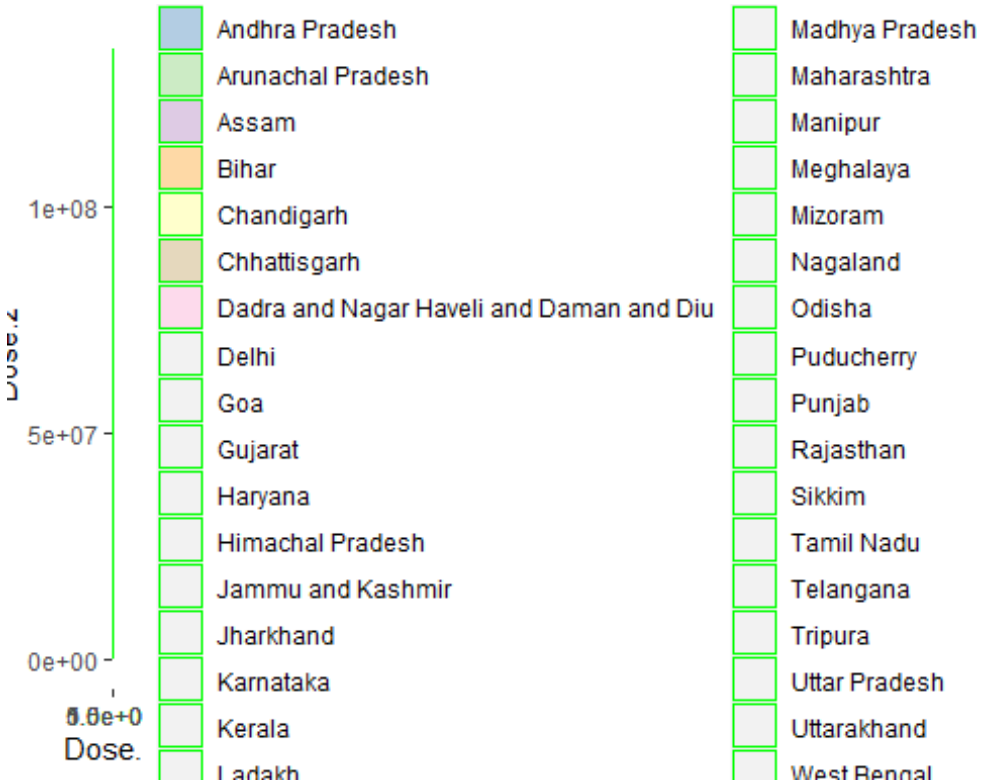


78.Usin

scale_fill_brewer() to the boxplot for changing colors

```
ggplot(data,aes(x=Dose.1,y=Dose.2,fill=states))+geom_col(position="dodge",col
our = "green")+scale_fill_brewer(palette = "Pastel1")
```

```
## Warning in RColorBrewer::brewer.pal(n, pal): n too large, allowed maximum
for palette Pastel1 is 9
## Returning the palette you asked for with that many colors
```



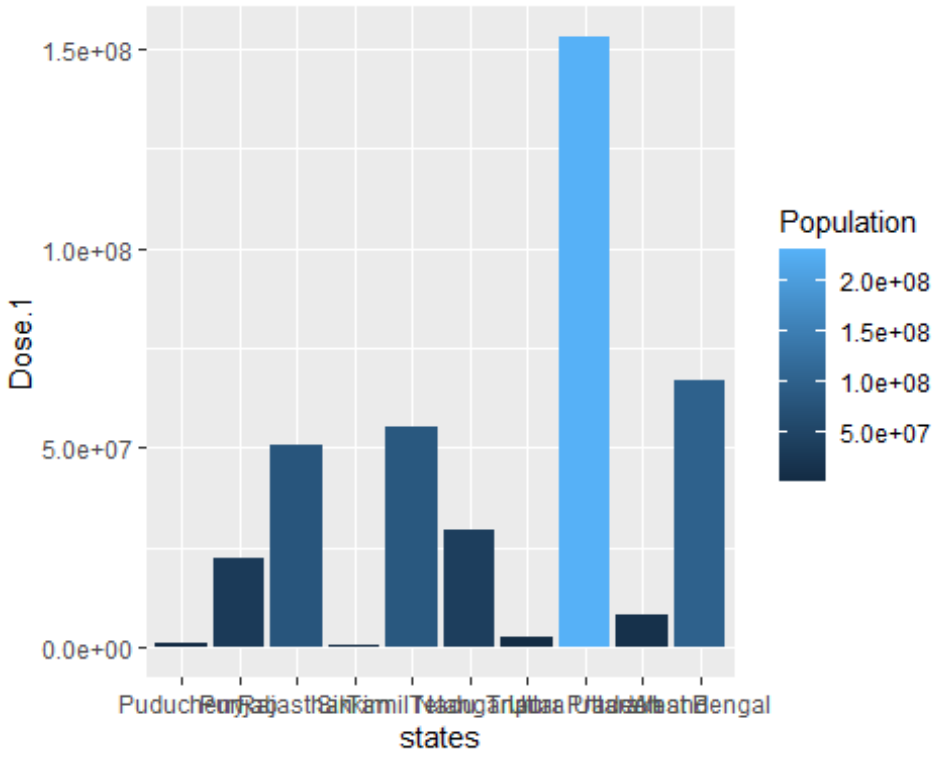
79.Rearranging and slicing

```
ans = data %>%
  arrange(desc(data)) %>%
  slice(1:10)
ans
```

```
##           states Total.Vaccination.Doses    Dose.1    Dose.2 Population
## 1    West Bengal               126778443  67031629  59746814  100896618
## 2    Uttarakhand                15936251   8140263   7795988   11700099
## 3  Uttar Pradesh               287827071 153151985 134675086  231502578
## 4        Tripura                 4894221   2644217   2250004    4184959
## 5      Telangana                57832105  29452703  28379402   38157311
## 6     Tamil Nadu               102131951  54996643  47135308   83697770
## 7         Sikkim                 1041215    538261    502954     658019
## 8      Rajasthan                95514875  50865992  44648883   79502477
## 9         Punjab                40322736  22180522  18142214   30501026
## 10    Puducherry                 1554319    892623    661696    1646050
```

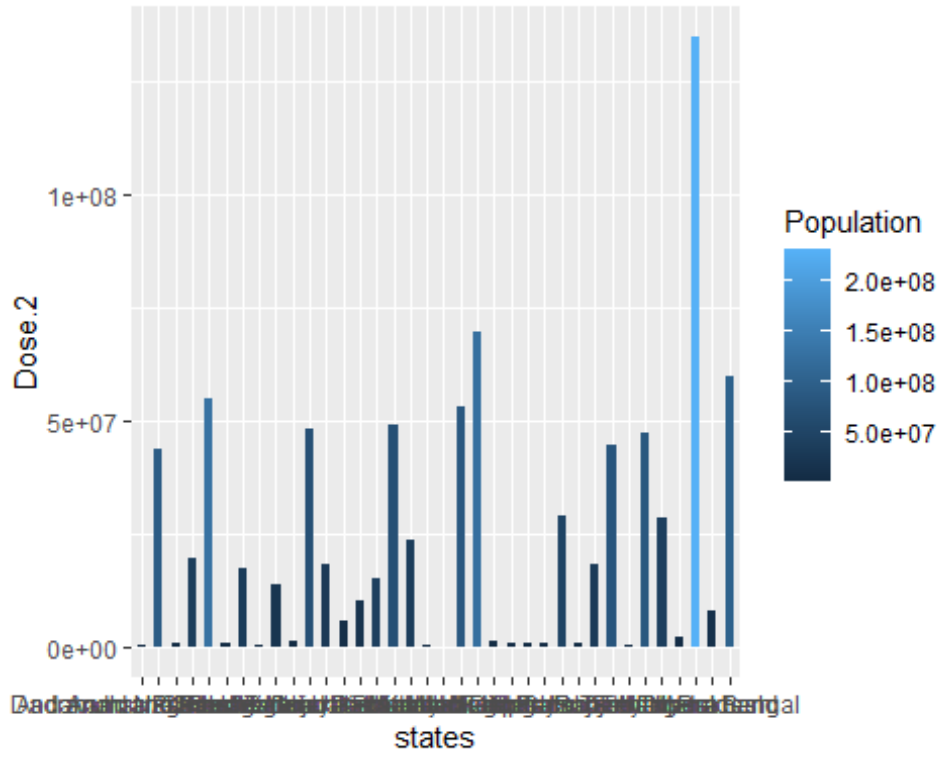
80.Then plotting the above data

```
ggplot(ans,aes(x=states,y=Dose.1,fill=Population))+geom_col()
```



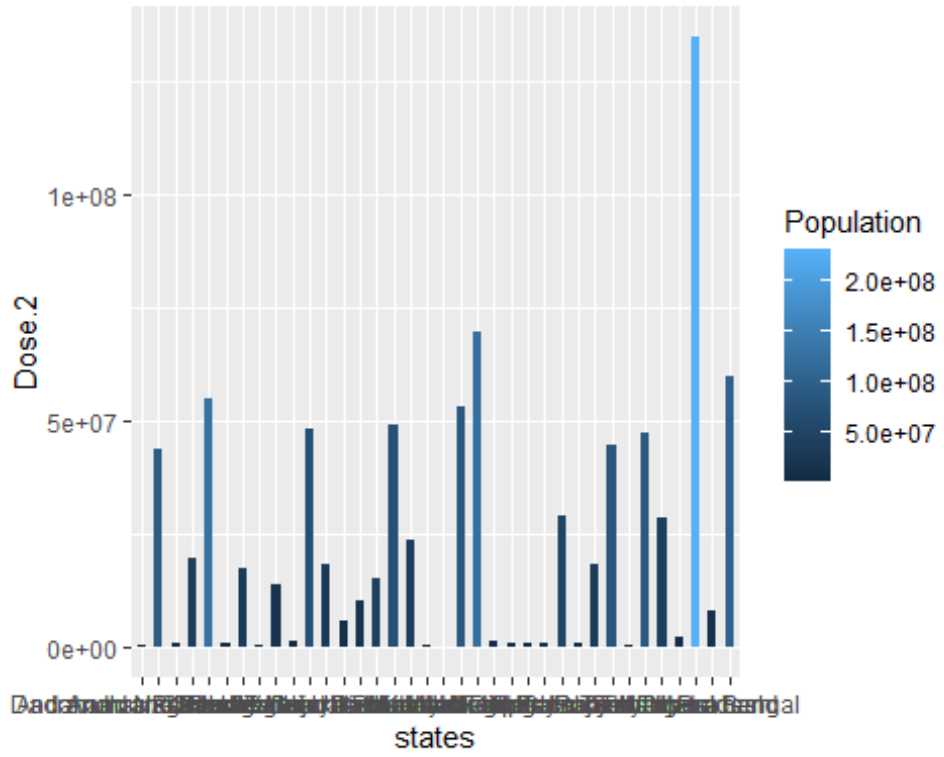
81.Plotting bar graph without spaces between bars

```
ggplot(data,aes(x=states,y=Dose.2,fill=Population))+geom_col(width=0.5,positi
on = "dodge")
```



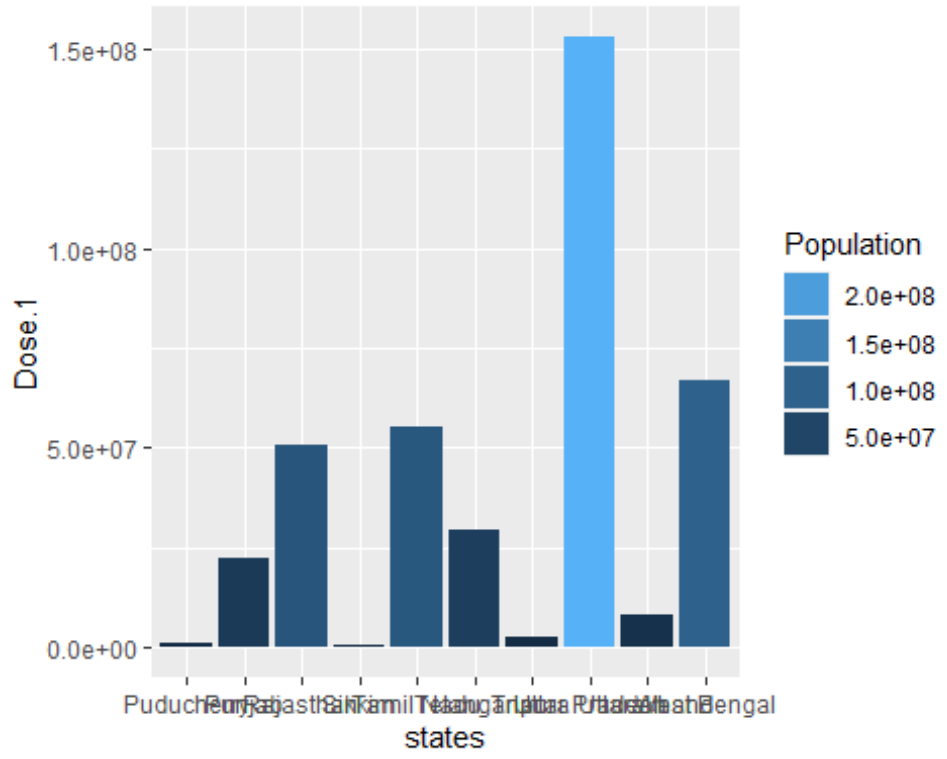
82.Plotting bar graph with spaces between bars

```
ggplot(data,aes(x=states,y=Dose.2,fill=Population))+geom_col(width=0.5,positi
on = position_dodge(0.7))
```



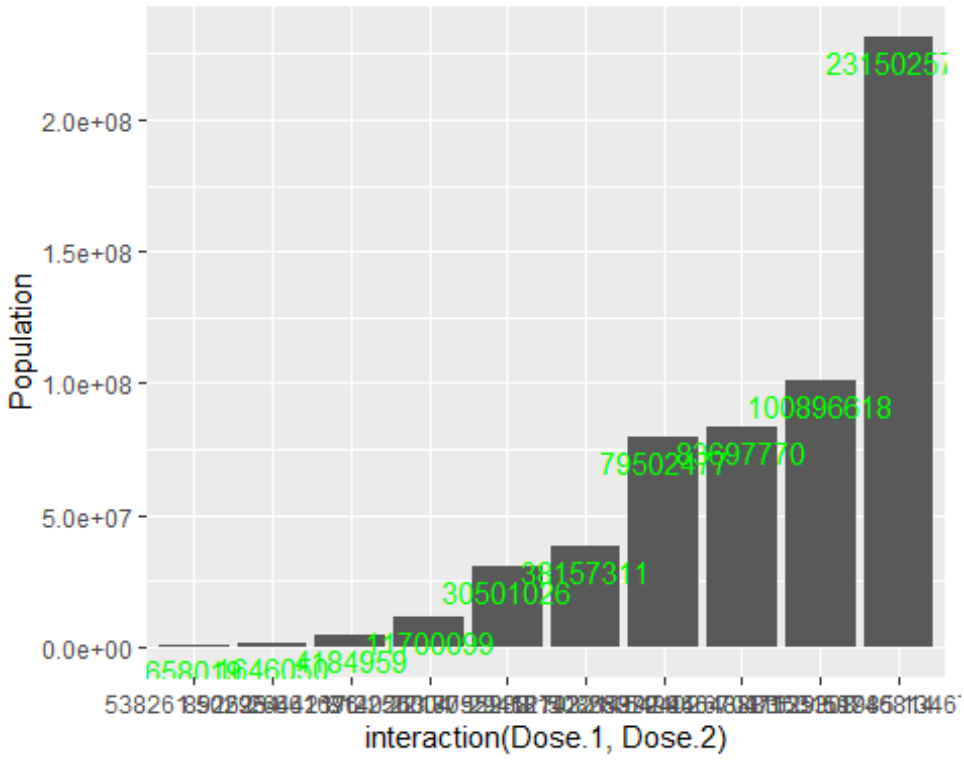
83Reversing the stacking order of the graph

```
ggplot(ans,aes(x=states,y=Dose.1,fill=Population))+geom_col()+guides(fill=gui
de_legend(reverse=TRUE))
```



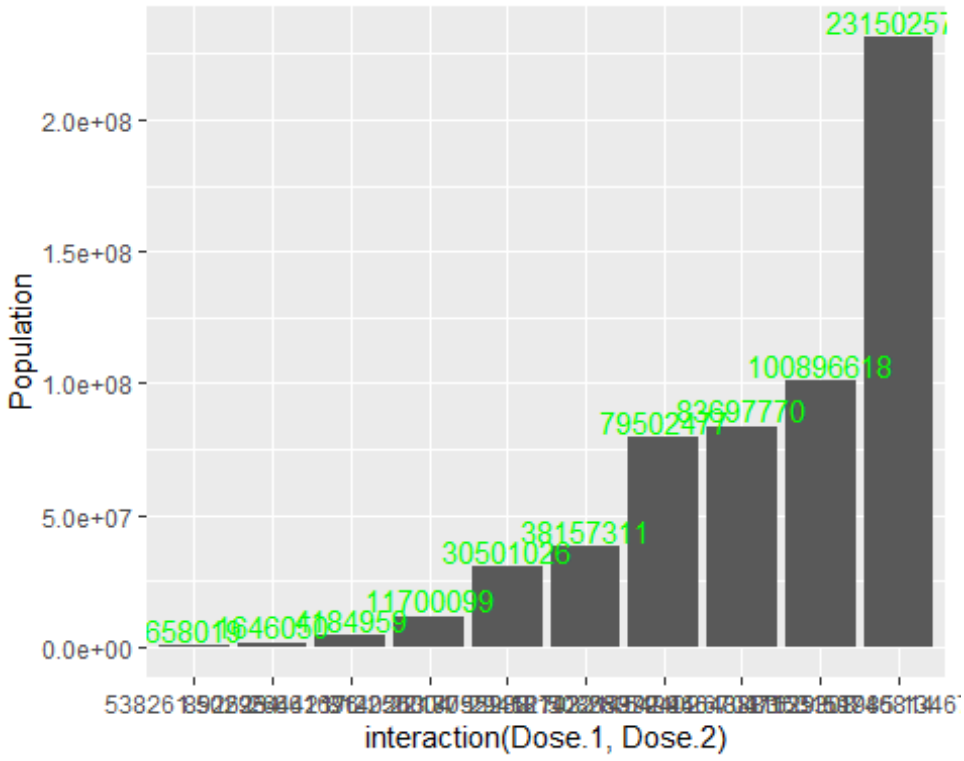
84.Adding text to x and y axis

```
ggplot(ans, aes(x = interaction(Dose.1, Dose.2), y = Population)) +
 geom_col() +
 geom_text(aes(label = Population), vjust = 1.5, colour = "green")
```



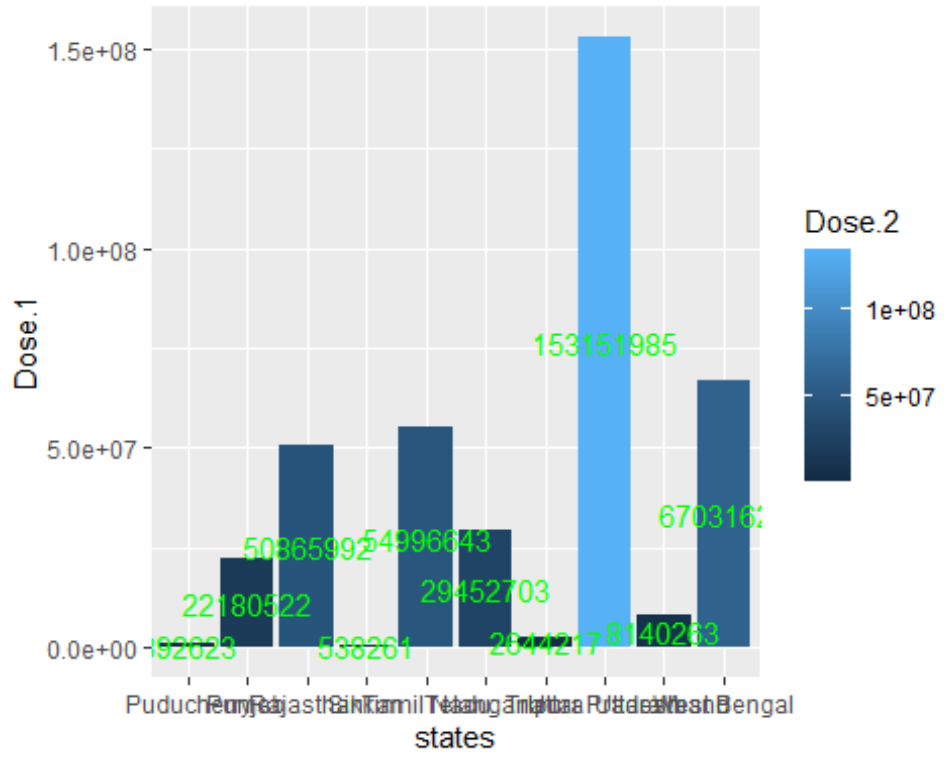
85.Adding text to x and y axis above the bars

```
ggplot(ans, aes(x = interaction(Dose.1, Dose.2), y = Population)) +
 geom_col() +
 geom_text(aes(label = Population), vjust = -0.2, colour = "green")
```



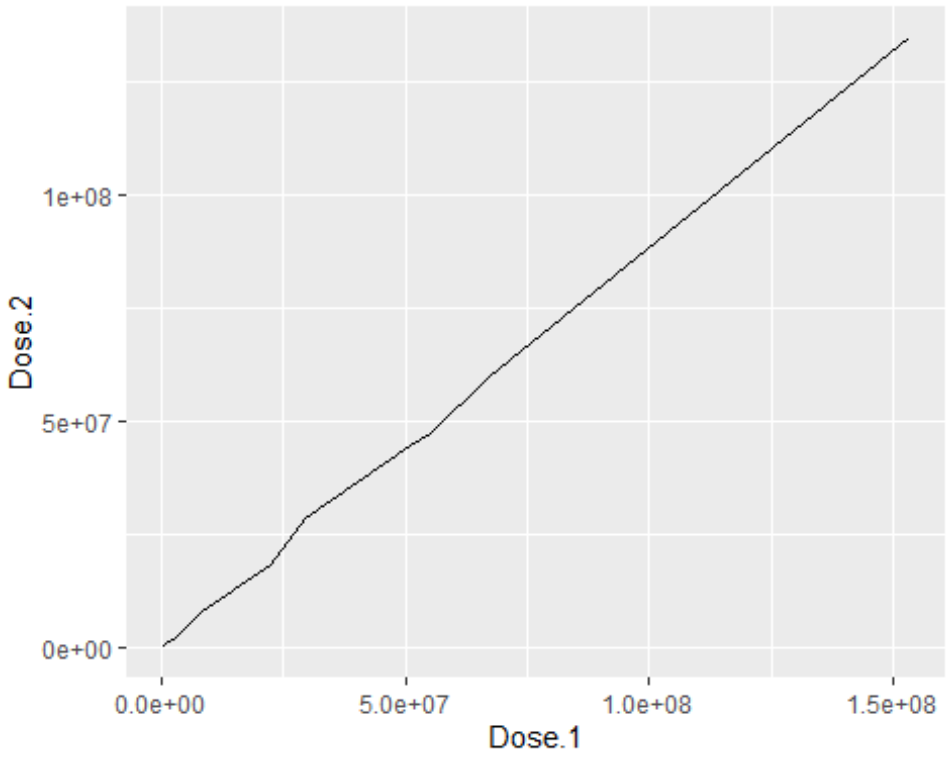
86.Putting the label in the middle of each bar

```
ce <- ans%>%
 group_by(Dose.1) %>%
 mutate(label_y = cumsum(Dose.1) - 0.5 * Dose.1)
ggplot(ce, aes(x = states, y = Dose.1, fill = Dose.2)) +
 geom_col() +
 geom_text(aes(y = label_y, label = Dose.1), colour = "green")
```



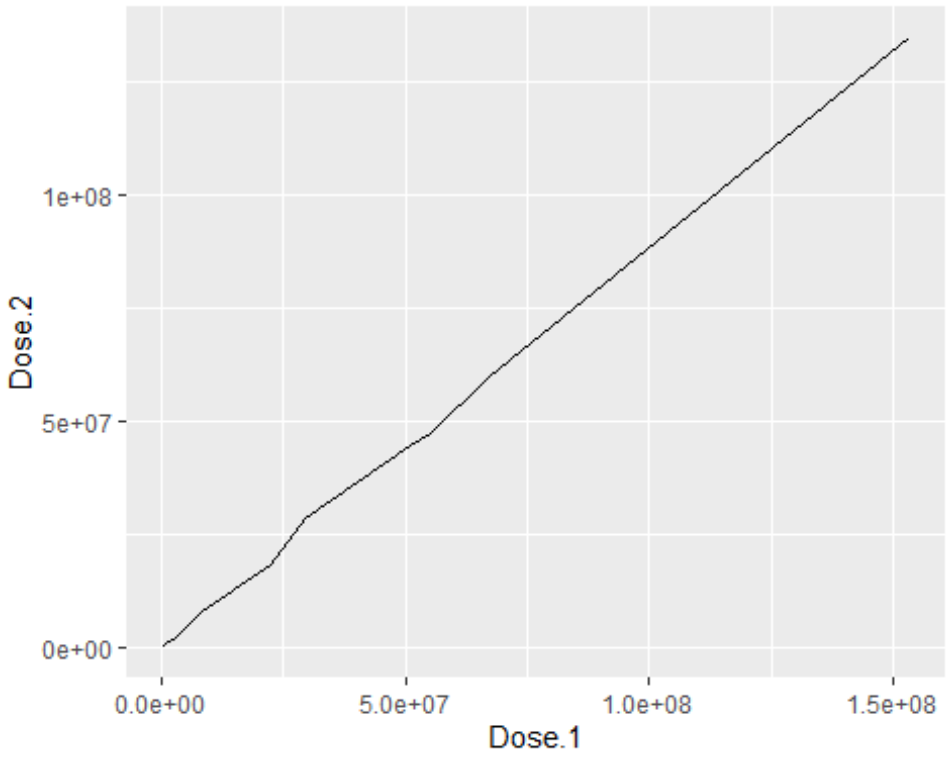
87.Setting the dose 2 column in the range from 0 to max

```
ggplot(ans, aes(x = Dose.1, y = Dose.2)) +
 geom_line() +
 ylim(0, max(ans$Dose.2))
```



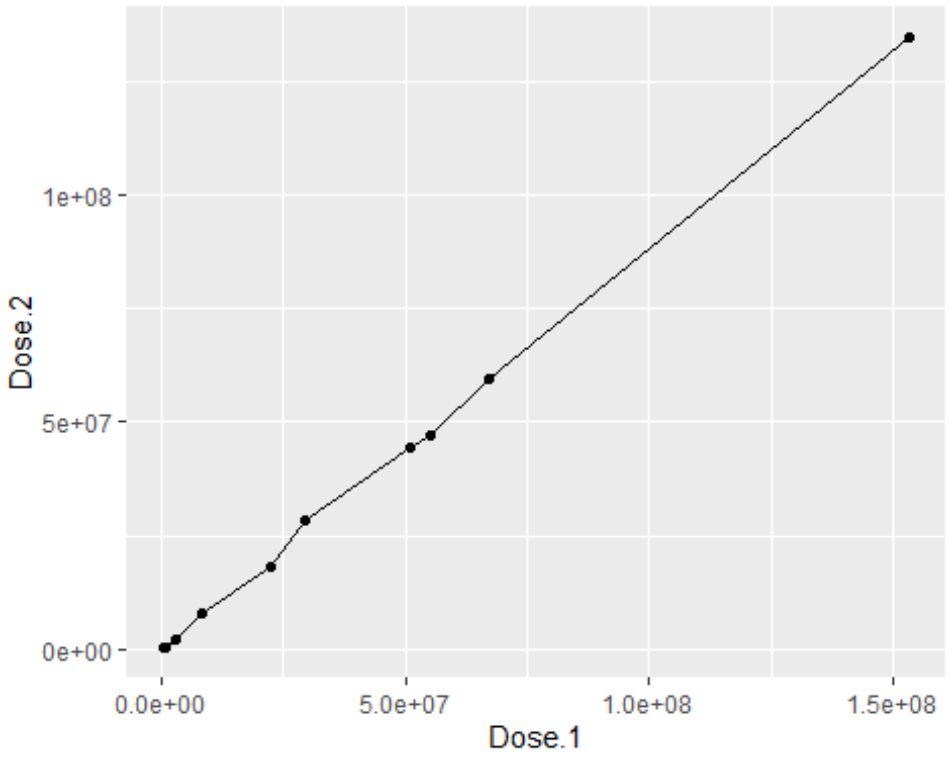
88.Setting the dose 2 column in the range from 0 to max by using expand_limits()

```
ggplot(ans, aes(x = Dose.1, y = Dose.2)) +
 geom_line() +
 expand_limits(y = 0)
```



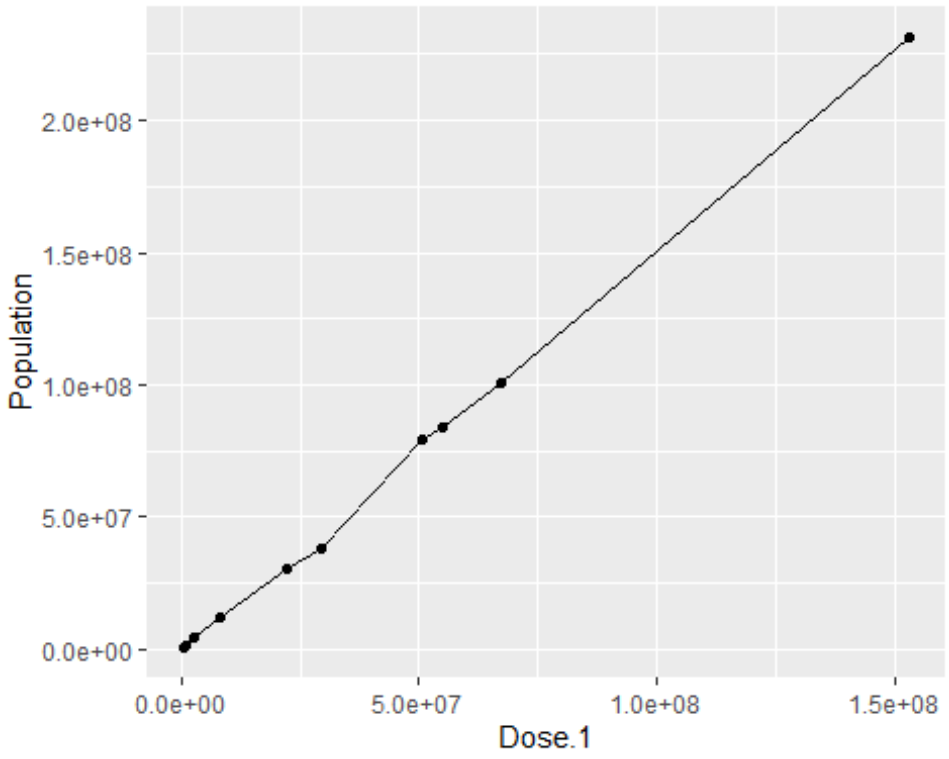
89.Plotting line and point graph for ans data

```
ggplot(ans, aes(x = Dose.1, y = Dose.2)) +
 geom_line() +
 geom_point()
```



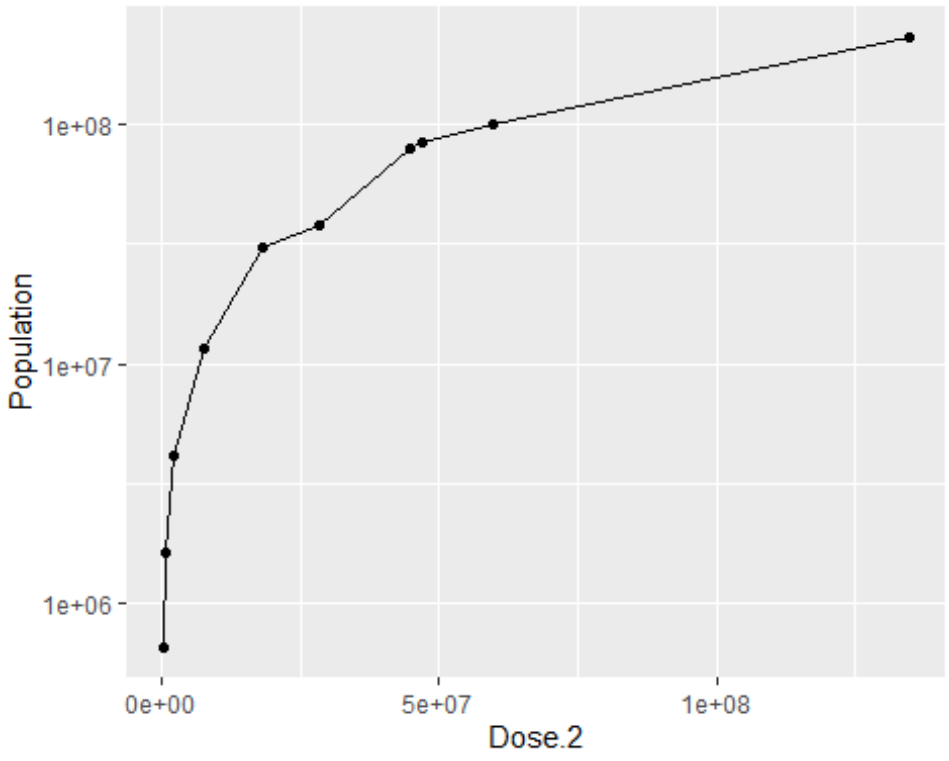
90.Plotting the line and point graph with inconsistent data points

```
ggplot(ans, aes(x = Dose.1, y = Population)) +
 geom_line() +
 geom_point()
```



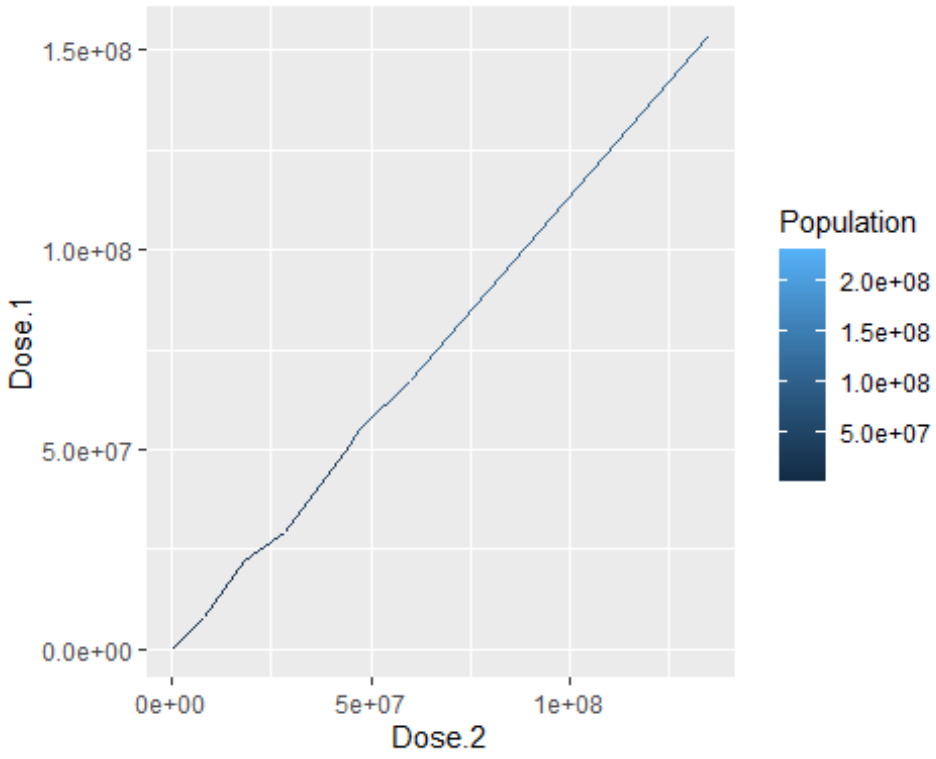
91.Plotting the line and point graph with inconsistent data points and log on y axis

```
ggplot(ans, aes(x = Dose.2, y = Population)) +
 geom_line() +
 geom_point() +
 scale_y_log10()
```



92.Adding mapsup color to line graph

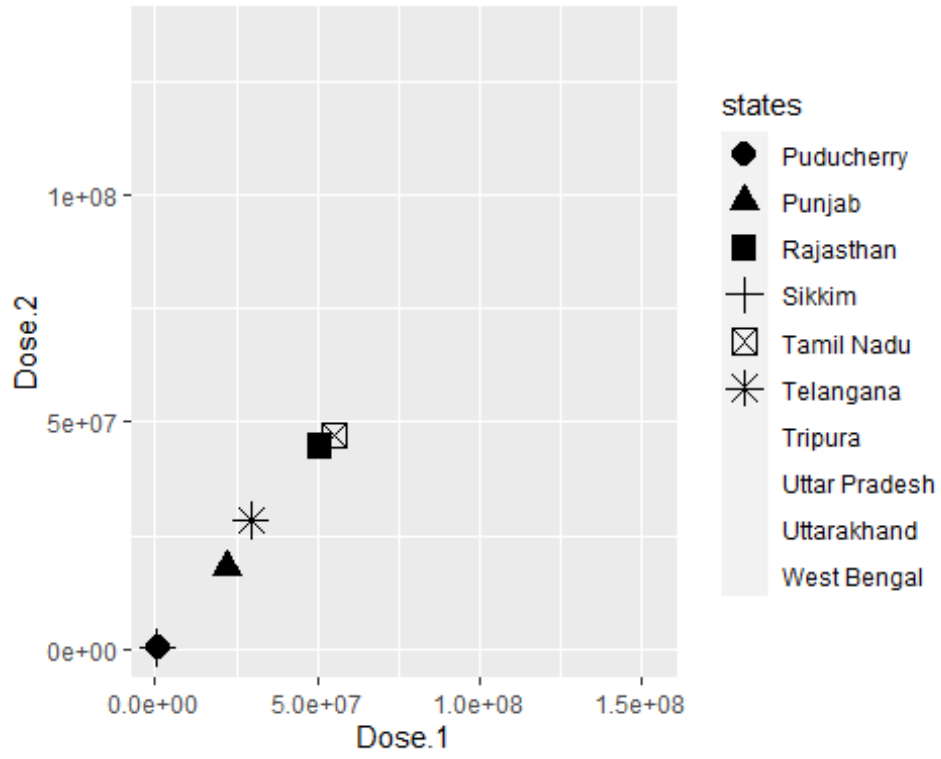
```
ggplot(ans, aes(x = Dose.2, y = Dose.1, colour = Population)) +
 geom_line()
```



93.Mapping variables to properties of a point

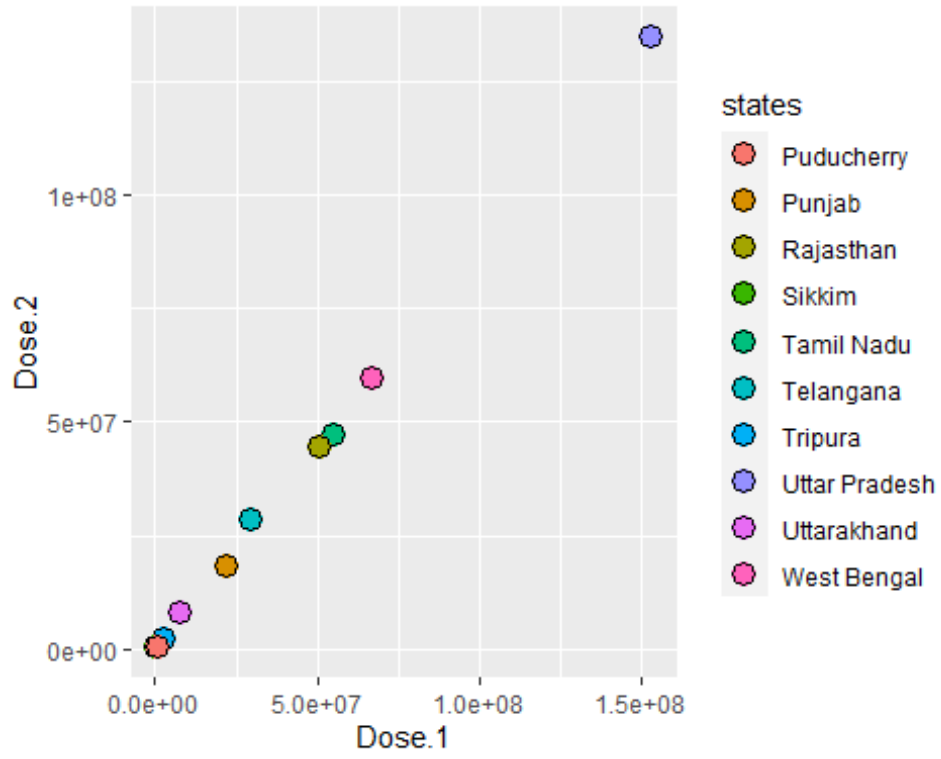
```
ggplot(ans, aes(x = Dose.1, y = Dose.2, shape = states)) +
 geom_line() +
 geom_point(size = 4)
```

```
## Warning: The shape palette can deal with a maximum of 6 discrete values
because
## more than 6 becomes difficult to discriminate; you have 10. Consider
## specifying shapes manually if you must have them.

## geom_path: Each group consists of only one observation. Do you need to
adjust
## the group aesthetic?

## Warning: Removed 4 rows containing missing values (geom_point).
```



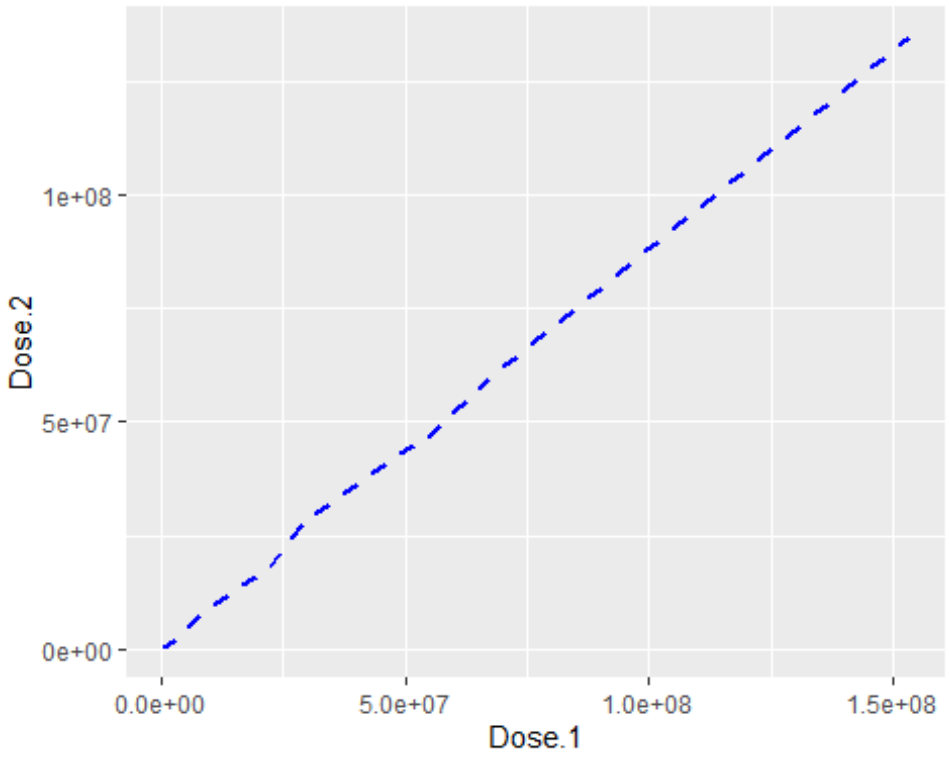
94.Mapping variables to properties of a point of different colors and same shape

```
ggplot(ans, aes(x = Dose.1, y = Dose.2, fill = states)) +
 geom_line() +
 geom_point(size = 4, shape = 21)

## geom_path: Each group consists of only one observation. Do you need to
adjust
## the group aesthetic?
```



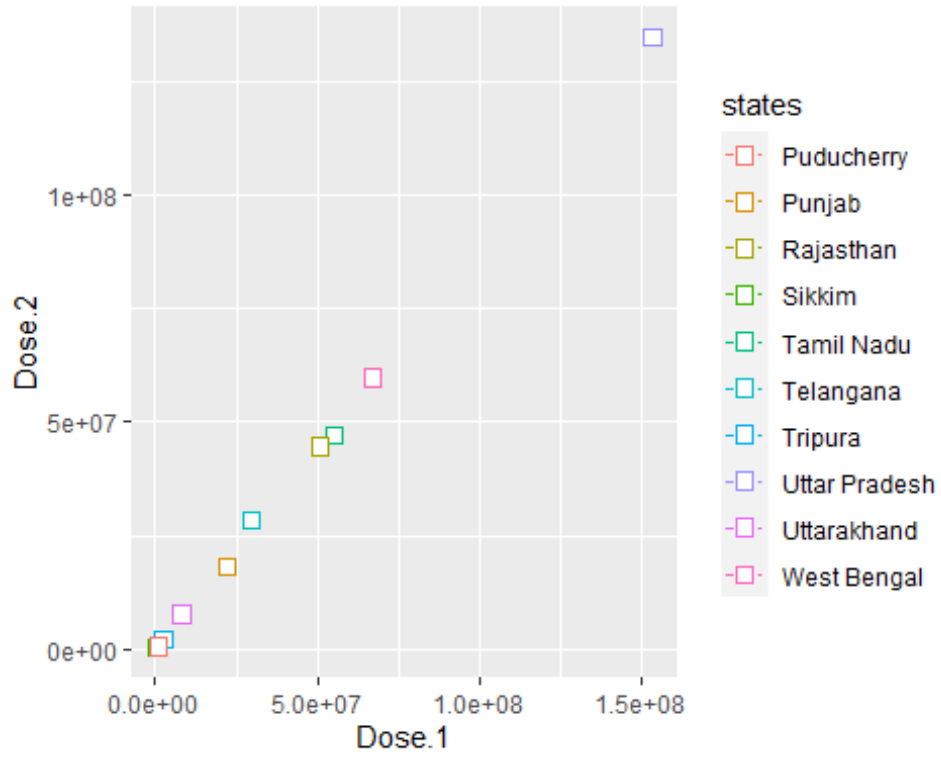
95.Plotting dashed line graph

```
ggplot(ans, aes(x = Dose.1, y = Dose.2)) +
 geom_line(linetype = "dashed", size = 1, colour = "blue")
```



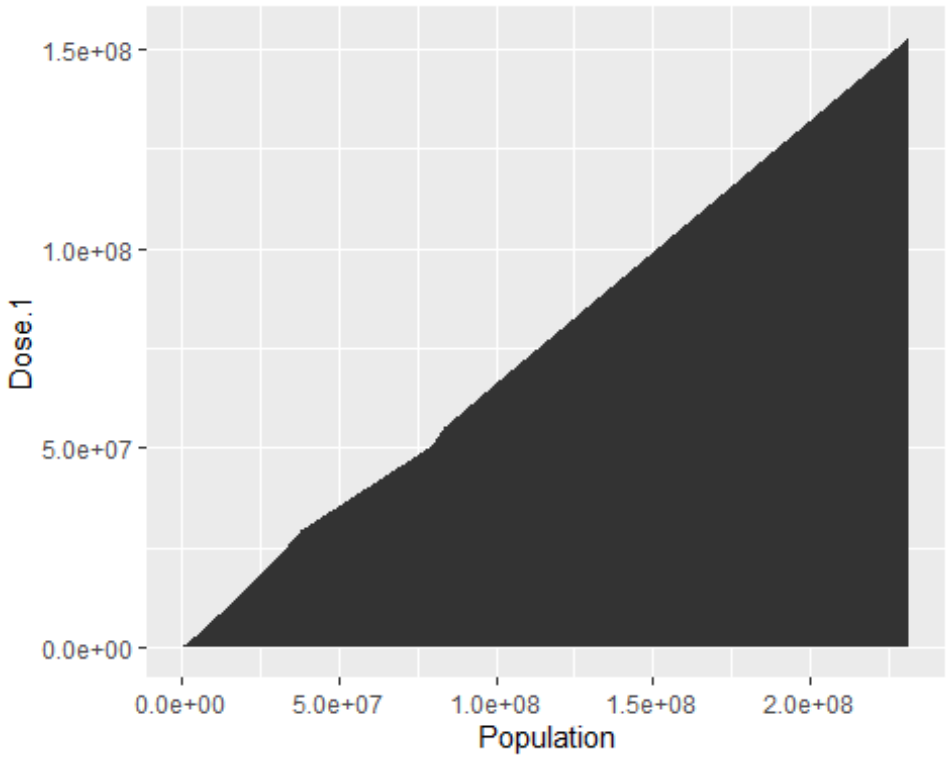
96.Plotting dashed lined raph with points

```
ggplot(ans, aes(x = Dose.1, y = Dose.2, colour = states)) +
 geom_line(linetype = "dashed") +
 geom_point(shape = 22, size = 3, fill = "white")

## geom_path: Each group consists of only one observation. Do you need to
adjust
## the group aesthetic?
```



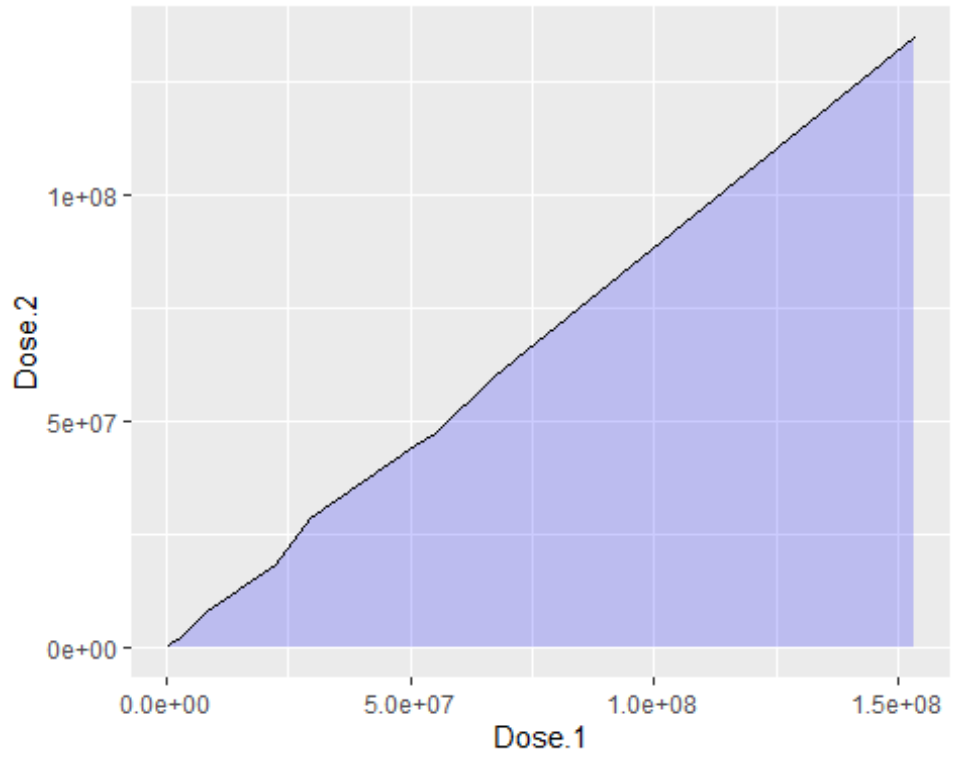
97.Plotting areaplot

```
ggplot(ans, aes(x = Population, y = Dose.1)) +
 geom_area()
```



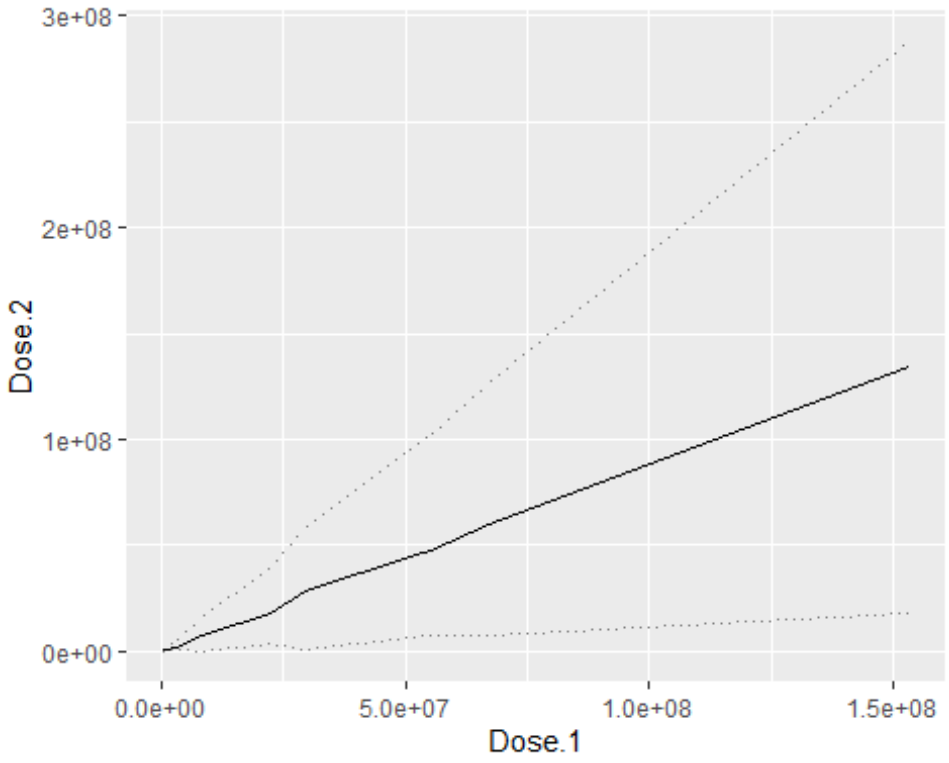
98.Plotting areaplot with colours for better analysis

```
ggplot(ans, aes(x = Dose.1, y = Dose.2)) +
 geom_area(colour = "black", fill = "blue", alpha = .2)
```



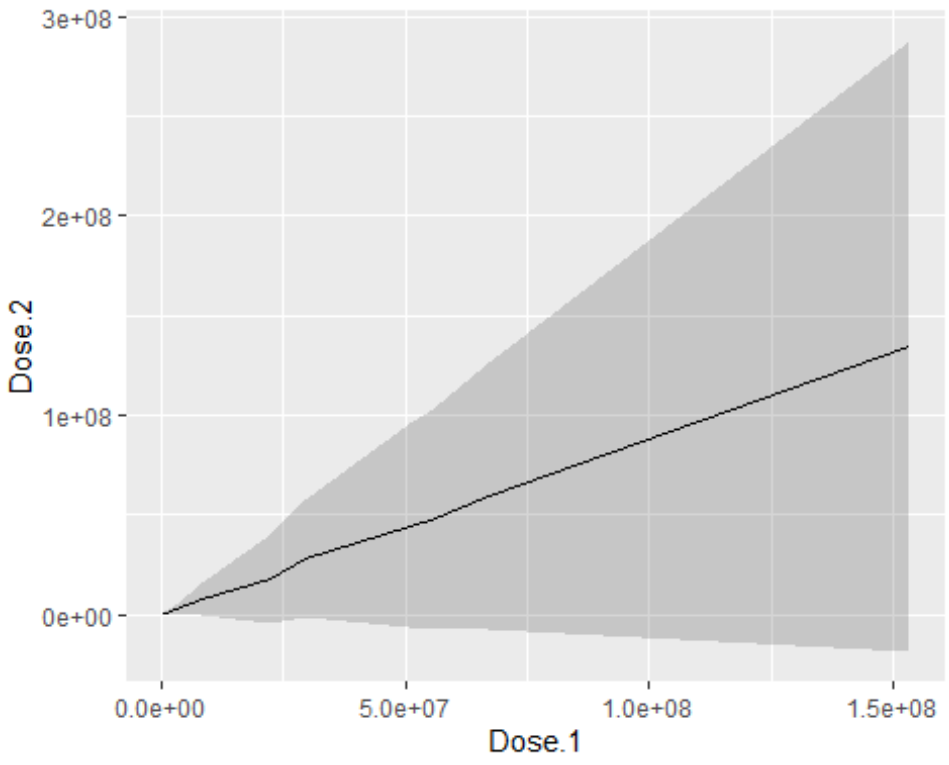
99.Area graph With a dotted line for upper and lower bounds

```
ggplot(ans, aes(x = Dose.1, y = Dose.2)) +
geom_line(aes(y = Dose.1 - Dose.2), colour = "grey50",
linetype = "dotted") +
geom_line(aes(y = Dose.1 + Dose.2), colour = "grey50",
linetype = "dotted") +
geom_line()
```



100.Plotting line graph on top of shaded regions

```
ggplot(ans, aes(x = Dose.1, y = Dose.2)) +
geom_ribbon(aes(ymin = Dose.2 - Dose.1,
ymax = Dose.1 + Dose.2),
alpha = 0.2) +
geom_line()
```


3e+08
2e+08
1e+08
0e+00
Dose.2
0.0e+00
5.0e+07
1.0e+08
1.5e+08
Dose.1