VIT®

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Creating tables and Analyzing Data regarding most runs scored in in International Cricket

Team Members:

Sricharan Adapa (21BCE6072)

Aim: To create tables and analyze data in R Studio.

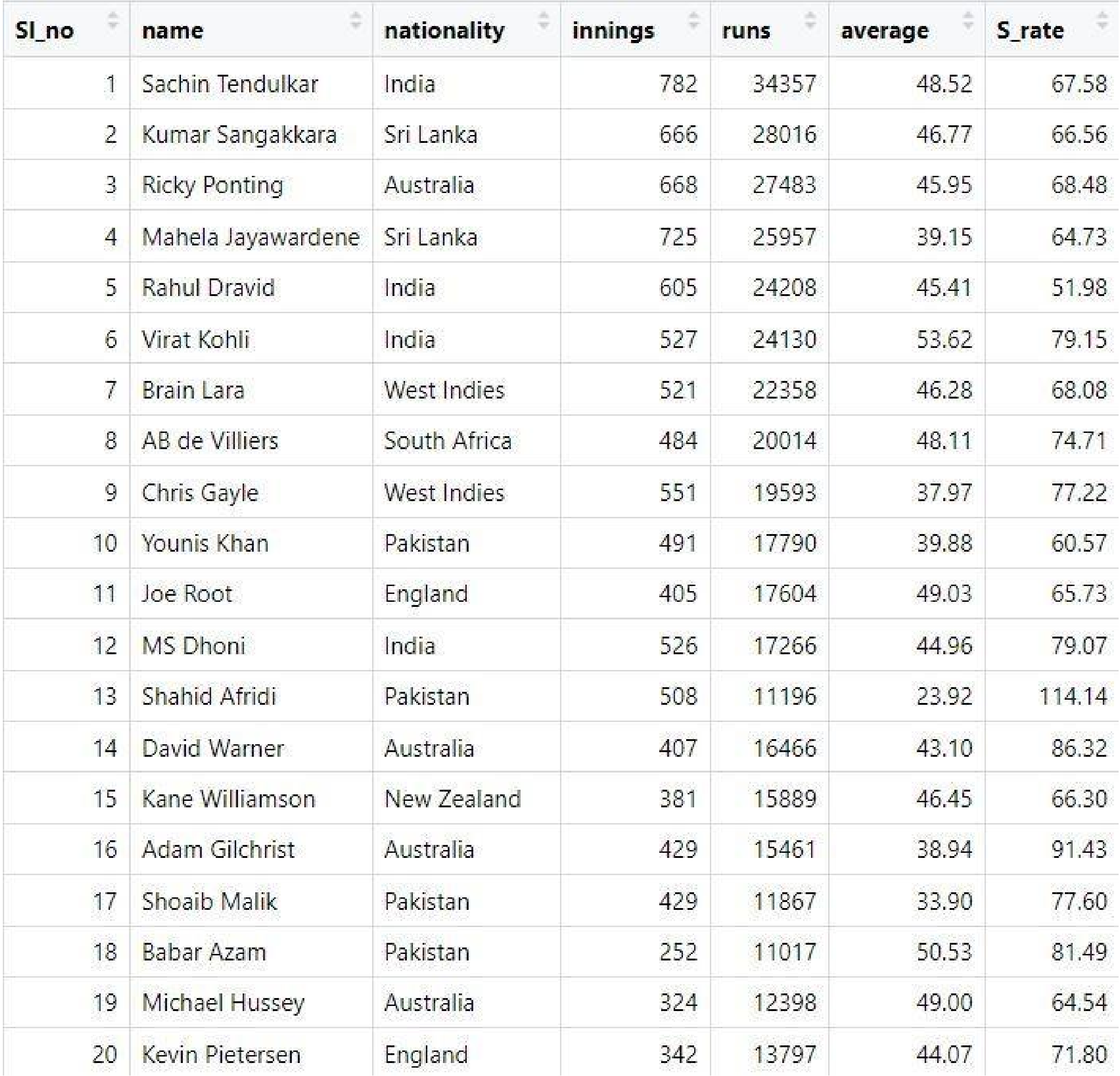
Dataset used: The data was taken from Kaggle.com. This dataset contain the Most Runs in International cricket in (ODI, Test , T20) and the all information about the batsman like strike rate,average and other information of the batsman.

Table creation:

Rcode

```
Sl_no=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)
name=c("Sachin Tendulkar","Kumar Sangakkara","Ricky Ponting","Mahela Jayawardene",
    "Rahul Dravid","Virat Kohli","Brain Lara","AB de Villiers","Chris Gayle","Younis Khan",
    "Joe Root", "MS Dhoni","Shahid Afridi","David Warner","Kane Williamson","Adam Gilchrist",
    "Shoaib Malik","Babar Azam","Michael Hussey","Kevin Pietersen" )
nationality=c(0,1,2,1,0,0,3,4,3,5,6,0,5,2,7,2,5,5,2,6)
innings=c(782,666,668,725,605,527,521,484,551,491,405,526,508,407,381,429,429,252,324,342)
runs=c(34357,28016,27483,25957,24208,24130,22358,20014,19593,
     17790,17604,17266,11196,16466,15889,15461,11867,11017,12398,13797)

average=c(48.52,46.77,45.95,39.15,45.41,53.62,46.28,48.11,37.97,39.88,49.03,44.96,23.92,43.10,46
.45,38.94,33.90,50.53,49.00,44.07)

S_rate=c(67.58,66.56,68.48,64.73,51.98,79.15,68.08,74.71,77.22,60.57,65.73,79.07,114.14,86.32,66.
30,91.43,77.60,81.49,64.54,71.80)

Bat=data.frame(Sl_no,name,nationality,innings,runs,average,S_rate)

Bat

Bat$nationality=factor(Bat$nationality,labels=c("India","Sri Lanka","Australia","West Indies","South
Africa","Pakistan","England","New Zealand"))
```

Output:



Sl_no	name	nationality	innings	runs	average	S_rate
1	Sachin Tendulkar	India	782	34357	48.52	67.58
2	Kumar Sangakkara	Sri Lanka	666	28016	46.77	66.56
3	Ricky Ponting	Australia	668	27483	45.95	68.48
4	Mahela Jayawardene	Sri Lanka	725	25957	39.15	64.73
5	Rahul Dravid	India	605	24208	45.41	51.98
6	Virat Kohli	India	527	24130	53.62	79.15
7	Brain Lara	West Indies	521	22358	46.28	68.08
8	AB de Villiers	South Africa	484	20014	48.11	74.71
9	Chris Gayle	West Indies	551	19593	37.97	77.22
10	Younis Khan	Pakistan	491	17790	39.88	60.57
11	Joe Root	England	405	17604	49.03	65.73
12	MS Dhoni	India	526	17266	44.96	79.07
13	Shahid Afridi	Pakistan	508	11196	23.92	114.14
14	David Warner	Australia	407	16466	43.10	86.32
15	Kane Williamson	New Zealand	381	15889	46.45	66.30
16	Adam Gilchrist	Australia	429	15461	38.94	91.43
17	Shoaib Malik	Pakistan	429	11867	33.90	77.60
18	Babar Azam	Pakistan	252	11017	50.53	81.49
19	Michael Hussey	Australia	324	12398	49.00	64.54
20	Kevin Pietersen	England	342	13797	44.07	71.80

S_rate → Strike Rate

Data Frames

The function data.frame() creates data frames, tightly coupled collections of variables which share many of the properties of matrices and of lists, used as the fundamental data structure by most of R's modeling software.

Factors

The function factor is used to encode a vector as a factor (the terms ‘category’ and ‘enumerated type’ are also used for factors). If argument ordered is TRUE, the factor levels are assumed to be ordered. For compatibility with S there is also a function ordered.

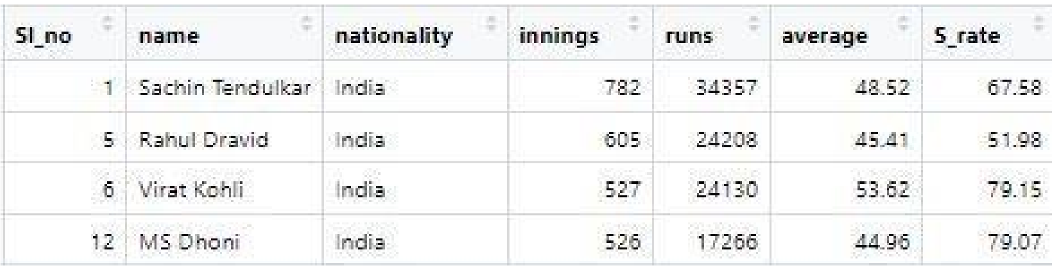
Tables based on nationality

1) India

Rcode:

```
Team_Ind=subset(Bat,Bat$nationality=='India')
```

Output:



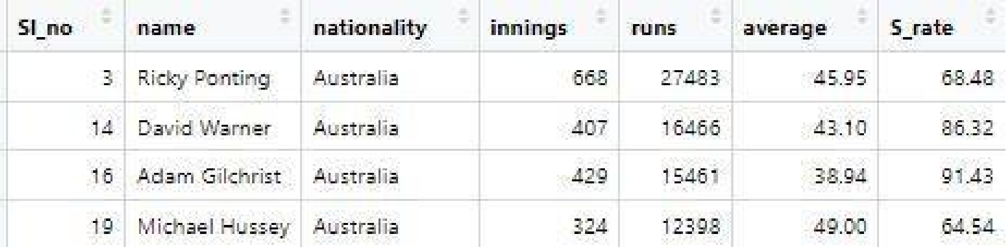
Sl_no	name	nationality	innings	runs	average	S_rate
1	Sachin Tendulkar	India	782	34357	48.52	67.58
5	Rahul Dravid	India	605	24208	45.41	51.98
6	Virat Kohli	India	527	24130	53.62	79.15
12	MS Dhoni	India	526	17266	44.96	79.07

2) Australia

Rcode:

```
Team_Aus=subset(Bat,Bat$nationality=='Australia')
```

Output:



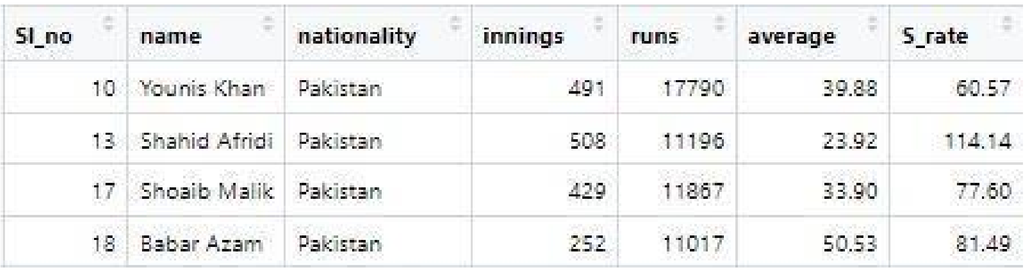
Sl_no	name	nationality	innings	runs	average	S_rate
3	Ricky Ponting	Australia	668	27483	45.95	68.48
14	David Warner	Australia	407	16466	43.10	86.32
16	Adam Gilchrist	Australia	429	15461	38.94	91.43
19	Michael Hussey	Australia	324	12398	49.00	64.54

3) Pakistan

Rcode:

```
Team_PAK=subset(Bat,Bat$nationality=='Pakistan')
```

Output:



Sl_no	name	nationality	innings	runs	average	S_rate
10	Younis Khan	Pakistan	491	17790	39.88	60.57
13	Shahid Afridi	Pakistan	508	11196	23.92	114.14
17	Shoaib Malik	Pakistan	429	11867	33.90	77.60
18	Babar Azam	Pakistan	252	11017	50.53	81.49

<u>Rcode for the rest of the teams:</u>

Team_SL=subset(Bat,Bat$nationality=='Sri Lanka')

Team_NZ=subset(Bat,Bat$nationality=='New Zealand')

Team_WI=subset(Bat,Bat$nationality=='West Indies')

Team_RSA=subset(Bat,Bat$nationality=='South Africa')

Team_Eng=subset(Bat,Bat$nationality=='England')

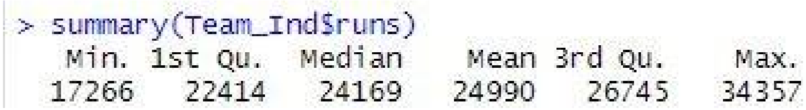
Subset

Return subsets of vectors, matrices or data frames which meet conditions.

<u>Measures of central tendency for runs of each team</u>

1) India
   Rcode:
   summary(Team_Ind$runs)

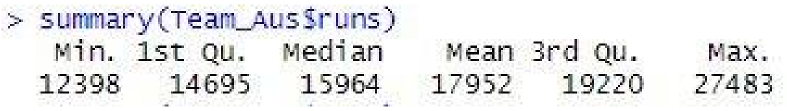
   Output:



```
> summary(Team_Ind$runs)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  17266   22414   24169   24990   26745   34357 
```

2) Australia
   Rcode:
   summary(Team_Aus$runs)

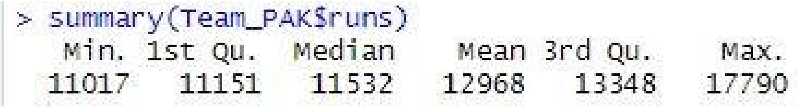
   Output:



```
> summary(Team_Aus$runs)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  12398   14695   15964   17952   19220   27483 
```

3) Pakistan
   Rcode:
   summary(Team_PAK$runs)

   Output:



```
> summary(Team_PAK$runs)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  11017   11151   11532   12968   13348   17790 
```

Rcode for the remaining teams:

summary(Team_SL$runs)

summary(Team_NZ$runs)

summary(Team_WI$runs)

summary(Team_RSA$runs)

summary(Team_Eng$runs)

Summary

The summary() function will run a quick statistical summary of a data frame, calculating mean, median and quartile values for continuous variables

Correlation between the teams:

Between India and Australia , between Australia and Pakistan ,and between India and Pakistan

a1=Team_Ind$runs

a2=Team_Aus$runs

a3=Team_PAK$runs

cor(a1,a2) cor(a2,a3)

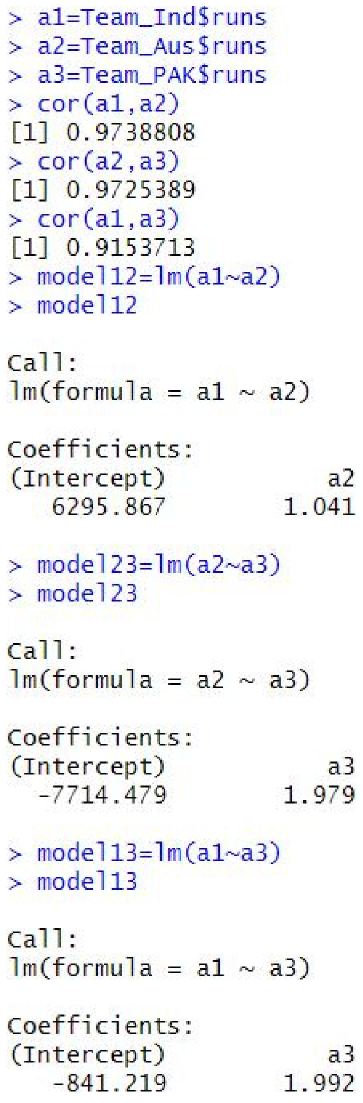
cor(a1,a3)

model12=lm(a1~a2)

model23=lm(a2~a3)

model13=lm(a1~a3)

Output:



```
> a1=Team_Ind$runs
> a2=Team_Aus$runs
> a3=Team_PAK$runs
> cor(a1,a2)
[1] 0.9738808
> cor(a2,a3)
[1] 0.9725389
> cor(a1,a3)
[1] 0.9153713
> model12=lm(a1~a2)
> model12

Call:
lm(formula = a1 ~ a2)

Coefficients:
(Intercept)           a2
   6295.867        1.041

> model23=lm(a2~a3)
> model23

Call:
lm(formula = a2 ~ a3)

Coefficients:
(Intercept)           a3
  -7714.479        1.979

> model13=lm(a1~a3)
> model13

Call:
lm(formula = a1 ~ a3)

Coefficients:
(Intercept)           a3
   -841.219        1.992
```

Correlation

Correlation quantifies the strength of the linear relationship between a pair of variables, whereas regression expresses the relationship in the form of an equation.

Cor

cor computes the correlation of x and y if these are vectors. If x and y are matrices then the correlations between the columns of x and the columns of y are computed.

Fitting linear models

lm is used to fit linear models, including multivariate ones. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance.

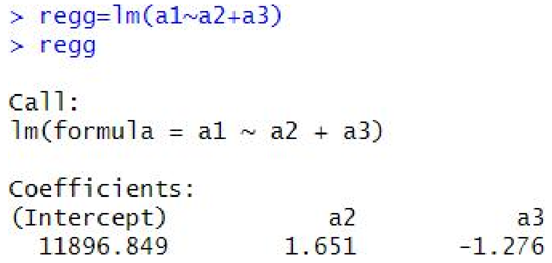
Multiple regression between the teams:

Rcode:

regg=lm(a1~a2+a3)

regg

Output:



```
> regg=lm(a1~a2+a3)
> regg

Call:
lm(formula = a1 ~ a2 + a3)

Coefficients:
(Intercept)           a2           a3  
  11896.849        1.651       -1.276  
```

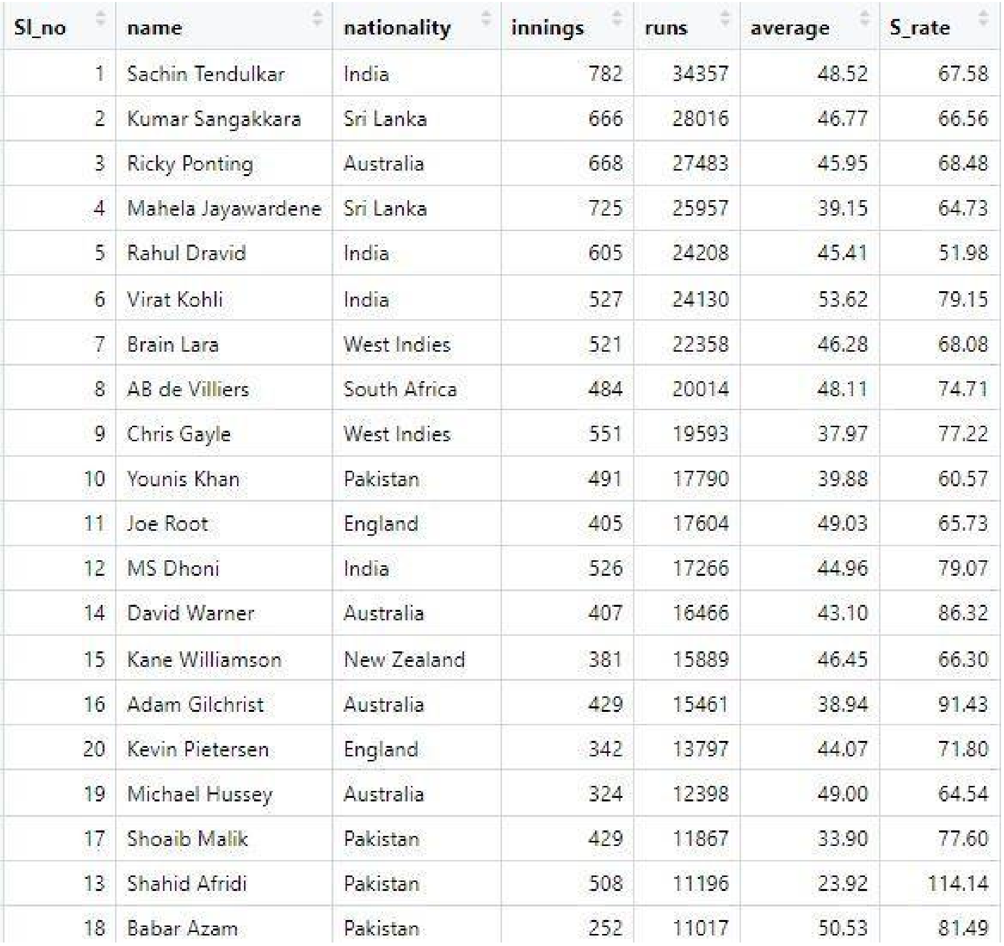
Sorting of players in decreasing order of the runs scored by them

Rcode:

df_sorted =Bat[order(Bat$runs, decreasing = TRUE),]

df_sorted

Output:



Sl_no	name	nationality	innings	runs	average	S_rate
1	Sachin Tendulkar	India	782	34357	48.52	67.58
2	Kumar Sangakkara	Sri Lanka	666	28016	46.77	66.56
3	Ricky Ponting	Australia	668	27483	45.95	68.48
4	Mahela Jayawardene	Sri Lanka	725	25957	39.15	64.73
5	Rahul Dravid	India	605	24208	45.41	51.98
6	Virat Kohli	India	527	24130	53.62	79.15
7	Brain Lara	West Indies	521	22358	46.28	68.08
8	AB de Villiers	South Africa	484	20014	48.11	74.71
9	Chris Gayle	West Indies	551	19593	37.97	77.22
10	Younis Khan	Pakistan	491	17790	39.88	60.57
11	Joe Root	England	405	17604	49.03	65.73
12	MS Dhoni	India	526	17266	44.96	79.07
14	David Warner	Australia	407	16466	43.10	86.32
15	Kane Williamson	New Zealand	381	15889	46.45	66.30
16	Adam Gilchrist	Australia	429	15461	38.94	91.43
20	Kevin Pietersen	England	342	13797	44.07	71.80
19	Michael Hussey	Australia	324	12398	49.00	64.54
17	Shoaib Malik	Pakistan	429	11867	33.90	77.60
13	Shahid Afridi	Pakistan	508	11196	23.92	114.14
18	Babar Azam	Pakistan	252	11017	50.53	81.49

Ordering Permutation

order returns a permutation which rearranges its first argument into ascending or descending order, breaking ties by further arguments. sort.list does the same, using only one argument.

Highest average by a batsman

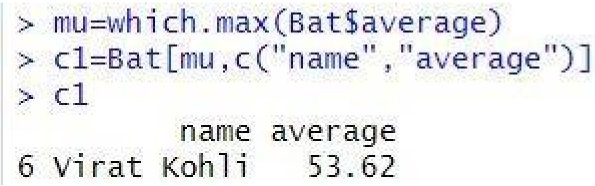
Rcode:

mu=which.max(Bat$average)

c1=Bat[mu,c("name","average")]

c1

Output:



```
> mu=which.max(Bat$average)
> c1=Bat[mu,c("name","average")]
> c1
          name average
6 Virat Kohli    53.62
```

Highest strike rate by a batsman

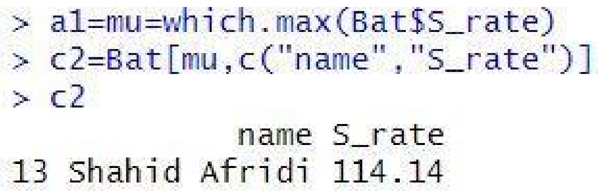
Rcode:

a1=mu=which.max(Bat$S_rate)

c2=Bat[mu,c("name","S_rate")]

c2

Output:



```
> a1=mu=which.max(Bat$S_rate)
> c2=Bat[mu,c("name","S_rate")]
> c2
            name S_rate
13 Shahid Afridi 114.14
```

which.max

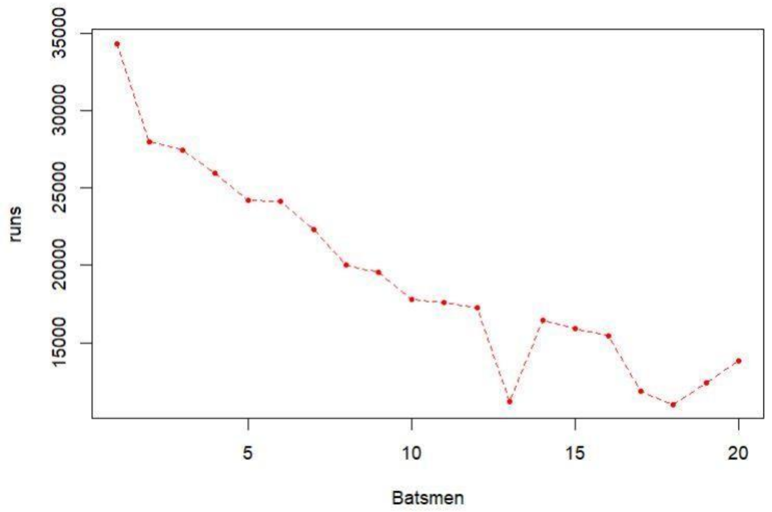
Determines the location, i.e., index of the maximum of a numeric (or logical) vector.

Plots

Rcode:

```
plot(runs,xlab = "Batsmen",type="o", pch=20,lty=2,col="red")
```

Output:



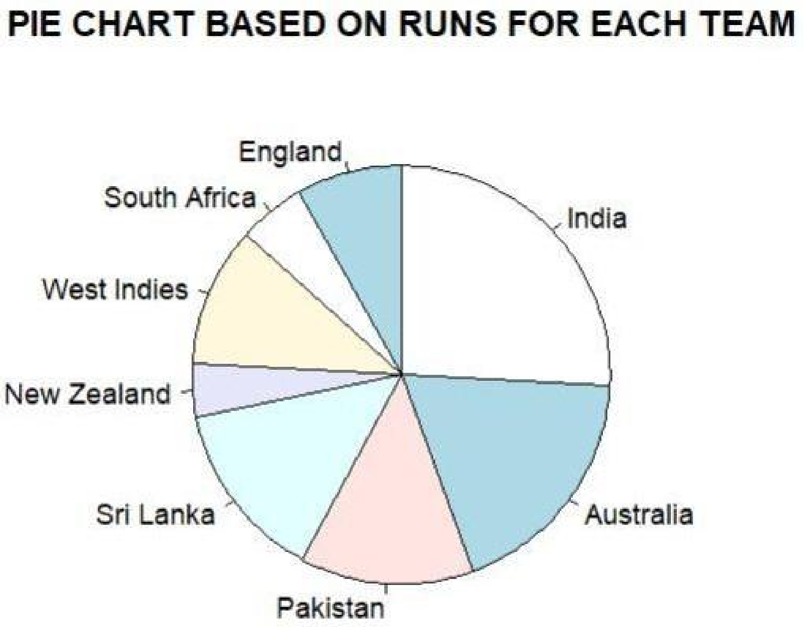
Line chart

It connects series of points by drawing line segments between them.

Rcode:

```
ind1=sum(Team_Ind$runs)

aust=sum(Team_Aus$runs)

paki=sum(Team_PAK$runs)

sri=sum(Team_SL$runs)

nz=sum(Team_NZ$runs)

wi=sum(Team_WI$runs)

sa=sum(Team_RSA$runs)

eng=sum(Team_Eng$runs)

x2=c(ind1,aust,paki,sri,nz,wi,sa,eng)

x3=c("India","Australia","Pakistan","Sri Lanka","New Zealand","West Indies","South
Africa","England")

pie(x2,labels=x3,clockwise = TRUE,main = "PIE CHART BASED ON RUNS FOR EACH TEAM")
```

Output:



Pie Chart

R Programming language has numerous libraries to create charts and graphs. A pie-chart is a representation of values as slices of a circle with different colors. The slices are labeled and the numbers corresponding to each slice is also represented in the chart. In R the pie chart is created using the pie() function which takes positive numbers as a vector input. The additional parameters are used to control labels, color, title etc.

T-test

Case 1:

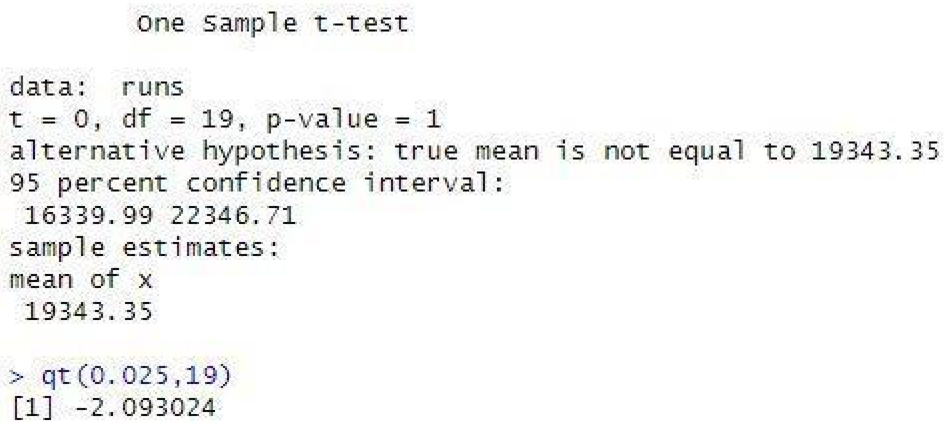
H0: The mean runs scored is 19343.35 H1:

The mean runs scored is not 19343.35

Rcode:

```
t.test(runs,alternative="two.sided",mu=19343.35)qt(0.025,19)
```

Output:



```
        One Sample t-test

data:  runs
t = 0, df = 19, p-value = 1
alternative hypothesis: true mean is not equal to 19343.35
95 percent confidence interval:
 16339.99 22346.71
sample estimates:
mean of x
 19343.35

> qt(0.025,19)
[1] -2.093024
```

Conclusion: Since t(cal)<t(critical), reject null hypothesis at 5% LOS. Thus, the mean runs scored is 19343.35.

Case 2:

H0: The mean runs scored is 9000 H1:

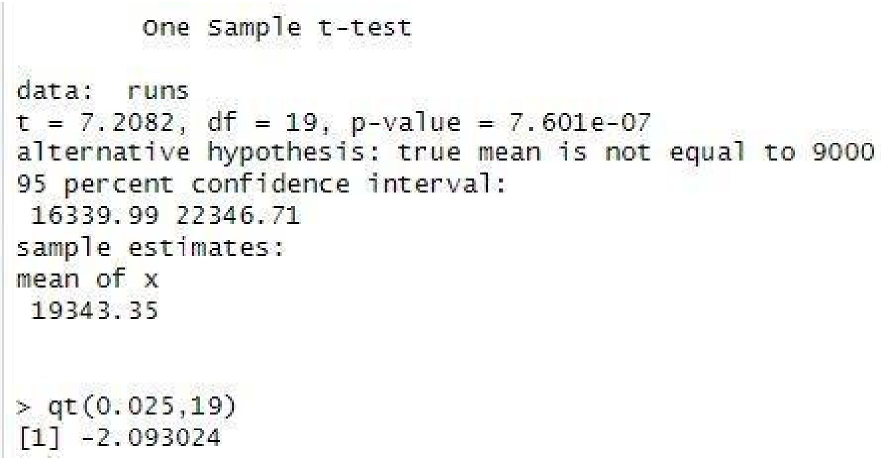
The mean runs scored is not 9000

Rcode:

t.test(runs,alternative="two.sided",mu=9000)

qt(0.025,19)

Output:



```
        One Sample t-test

data:  runs
t = 7.2082, df = 19, p-value = 7.601e-07
alternative hypothesis: true mean is not equal to 9000
95 percent confidence interval:
 16339.99 22346.71
sample estimates:
mean of x
 19343.35


> qt(0.025,19)
[1] -2.093024
```

Conclusion: Since t(cal)>t(critical), reject null hypothesis at 5% LOS. Thus, the mean runs scored is not 9000.

Student’s t-Test

Performs one and two sample t-tests on vectors of data.

The student t Distribution

Density, distribution function, quantile function and random generation for the t distribution with df degrees of freedom (and optional non-centrality parameter ncp).

Conclusion: Hence, tables were successfully created and a dataset analyzed in R Studio.