cricketr: A R package for analyzing performances of cricketers

Tinniam V Ganesh

Friday, July 03, 2015

Yet all experience is an arch wherethro'
Gleams that untravell'd world whose margin fades
For ever and forever when I move.
How dull it is to pause, to make an end,
To rust unburnish'd, not to shine in use!

`Ulysses by Alfred Tennyson`

Introduction

This is an introductory post in which I introduce a cricketing package **'cricketr'** whicj I have created. This package was a natural culmination to many earlier posts on cricketers and my completing 10 modules of an absorbing topics in Data Science Specialization, from John Hopkins University at Coursera. The thought of creating this package struck me some time back, and I have finally been able to bring this to fruition.

So here it is. My R package **'cricketr!!!'**

This package uses the statistics info available in ESPN Cricinfo Statsguru. The current version of this package supports all formats of the game including Test, ODI and Twenty20 versions.

You should be able to install the package from GitHub and use the many functions available in the package. Please mindful of the ESPN Cricinfo Terms of Use

Take a look at my short video tutorial on my R package cricketr on Youtube - R package cricketr - A short tutorial

Do check out my interactive Shiny app implementation using the cricketr package - Sixer - R package cricketr's new Shiny avatar

The cricketr package

The cricketr package has several functions that perform several different analyses on both batsman and bowlers. The package has function that plot percentage frequency runs or wickets, runs likelihood for a batsman, relative run/strike rates of batsman and relative performance/economy rate for bowlers are available.

Other interesting functions include batting performance moving average, forecast and a function to check whether the batsmans in in-form or out-of-form.

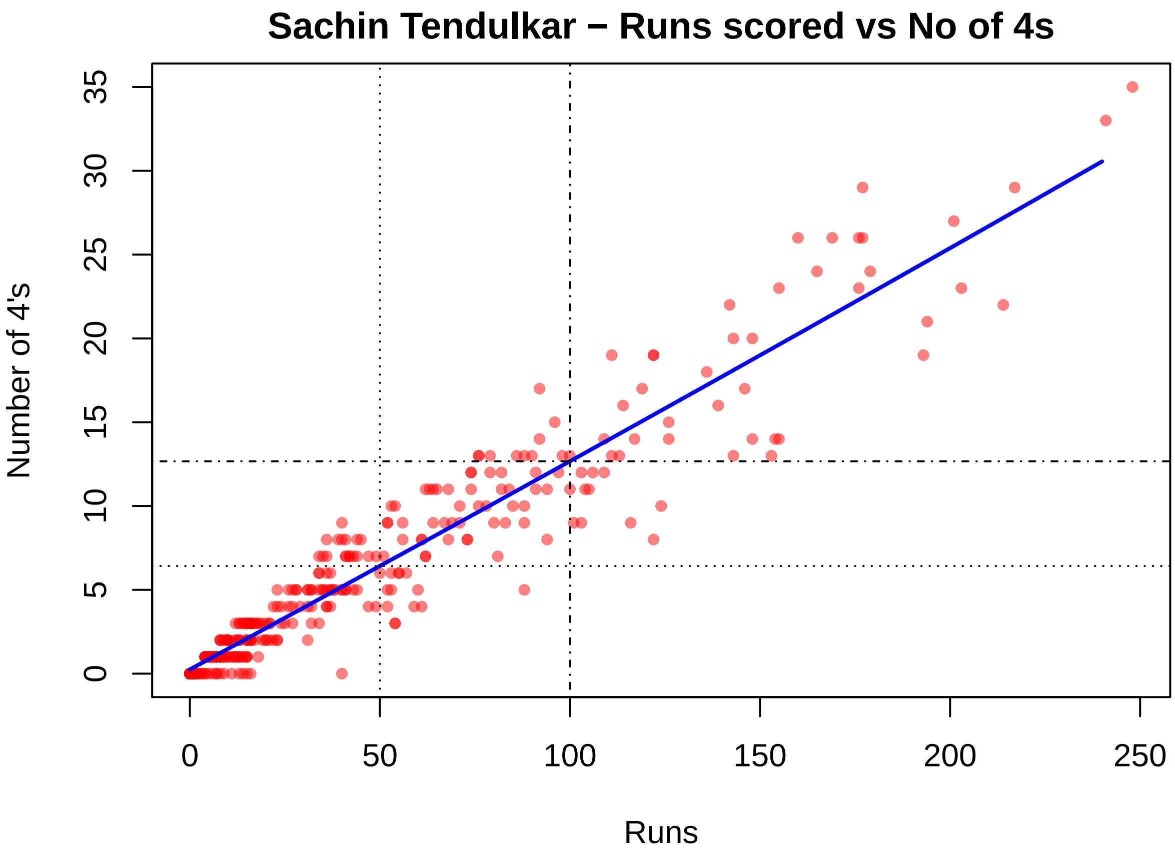
The data for a particular player can be obtained with the getPlayerData() function. To do you will need to go to ESPN CricInfo Player and type in the name of the player for e.g Ricky Ponting, Sachin Tendulkar etc. This will bring up a page which have the profile number for the player e.g. for Sachin Tendulkar this would be http://www.espncricinfo.com/india/content/player/35320.html. Hence, Sachin's profile is 35320. This can be used to get the data for Tendulkar as shown below

The cricketr package is now available from **CRAN!!!** You should be able to install directly with

```
if (!require("cricketr")){
    install.packages("cricketr",lib = "c:/test")
}
library(cricketr)
```

The cricketr package includes some pre-packaged sample (.csv) files. You can use these sample to test functions as shown below

```
# Retrieve the file path of a data file installed with cricketr
pathToFile <- system.file("data", "tendulkar.csv", package = "cricketr")
batsman4s(pathToFile, "Sachin Tendulkar")
```



```
# The general format is pkg-function(pathToFile,par1,...)
#batsman4s(<path-To-File>,"Sachin Tendulkar")
```

Alternatively, the cricketr package can be installed from GitHub with

```
if (!require("cricketr")){
  library(devtools)
  install_github("tvganesh/cricketr")
}
library(cricketr)
```

The pre-packaged files can be accessed as shown above. To get the data of any player use the function getPlayerData()

```r
tendulkar <- getPlayerData(35320,dir="..",file="tendulkar.csv",type="batting",homeOrAway=c(1,2),
                           result=c(1,2,4))
```

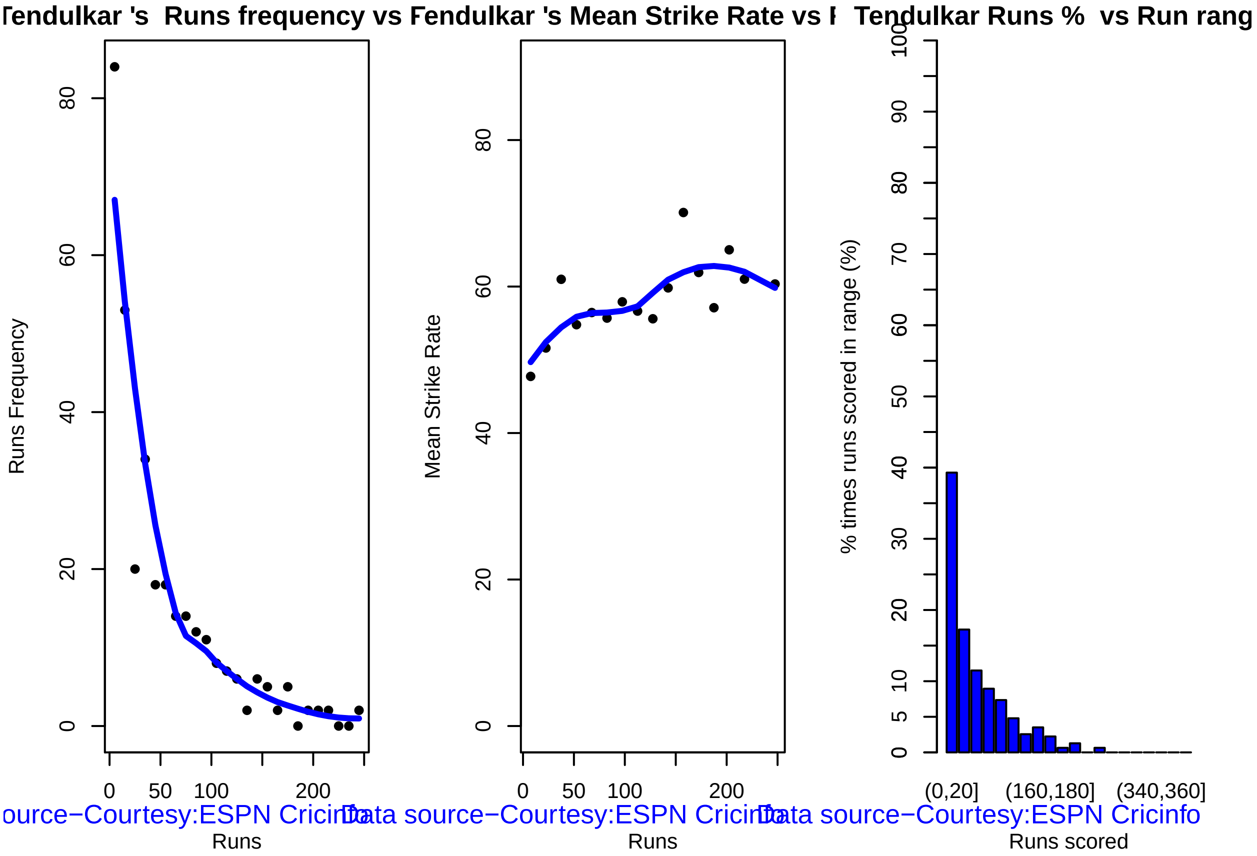
Important Note This needs to be done only once for a player. This function stores the player's data in a CSV file (for e.g. tendulkar.csv as above) which can then be reused for all other functions. Once we have the data for the players many analyses can be done. This post will use the stored CSV file obtained with a prior getPlayerData for all subsequent analyses

Sachin Tendulkar's performance - Basic Analyses

The 3 plots below provide the following for Tendulkar

1. Frequency percentage of runs in each run range over the whole career
2. Mean Strike Rate for runs scored in the given range
3. A histogram of runs frequency percentages in runs ranges

```r
par(mfrow=c(1,3))
par(mar=c(4,4,2,2))
batsmanRunsFreqPerf("./tendulkar.csv","Tendulkar")
batsmanMeanStrikeRate("./tendulkar.csv","Tendulkar")
batsmanRunsRanges("./tendulkar.csv","Tendulkar")
```

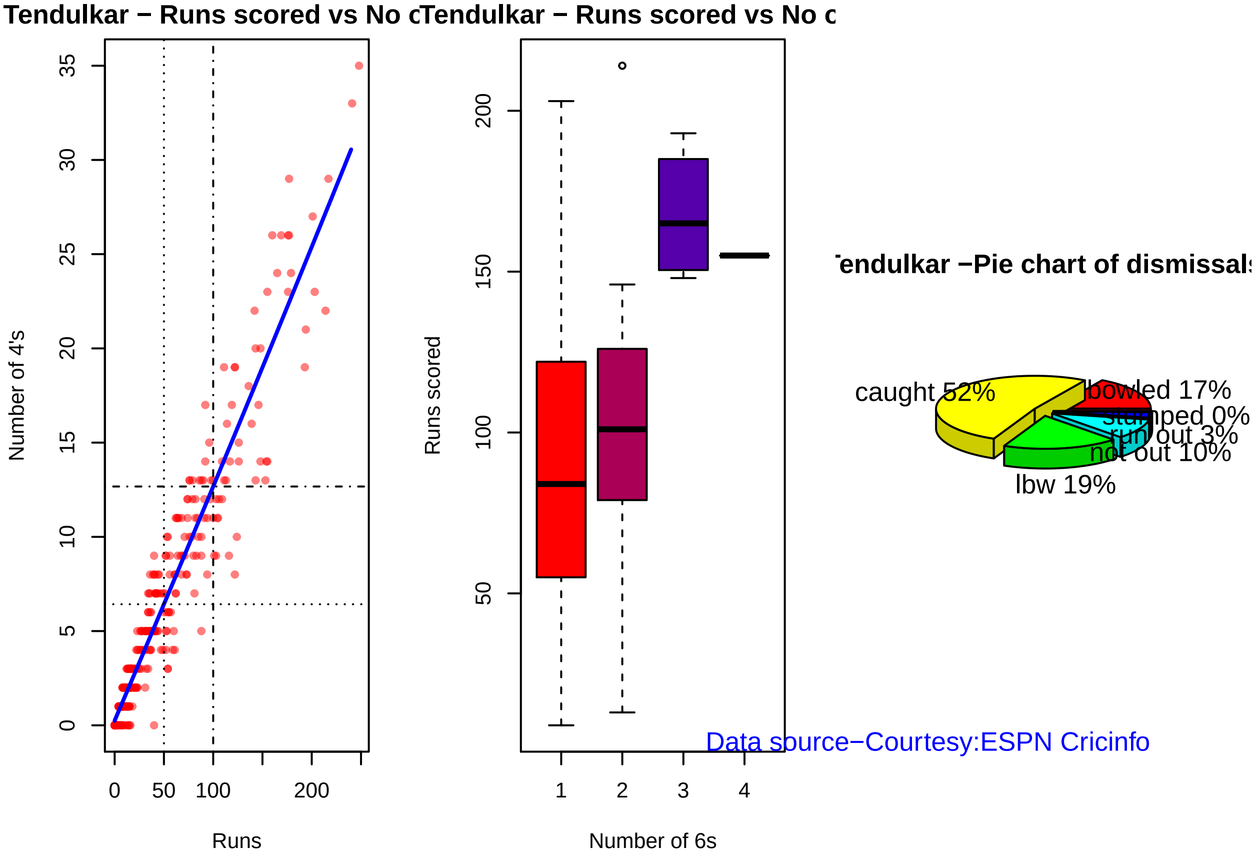


```
dev.off()
```

```
## null device
##           1
```

More analyses

```
par(mfrow=c(1,3))
par(mar=c(4,4,2,2))
batsman4s("./tendulkar.csv","Tendulkar")
batsman6s("./tendulkar.csv","Tendulkar")
batsmanDismissals("./tendulkar.csv","Tendulkar")
```



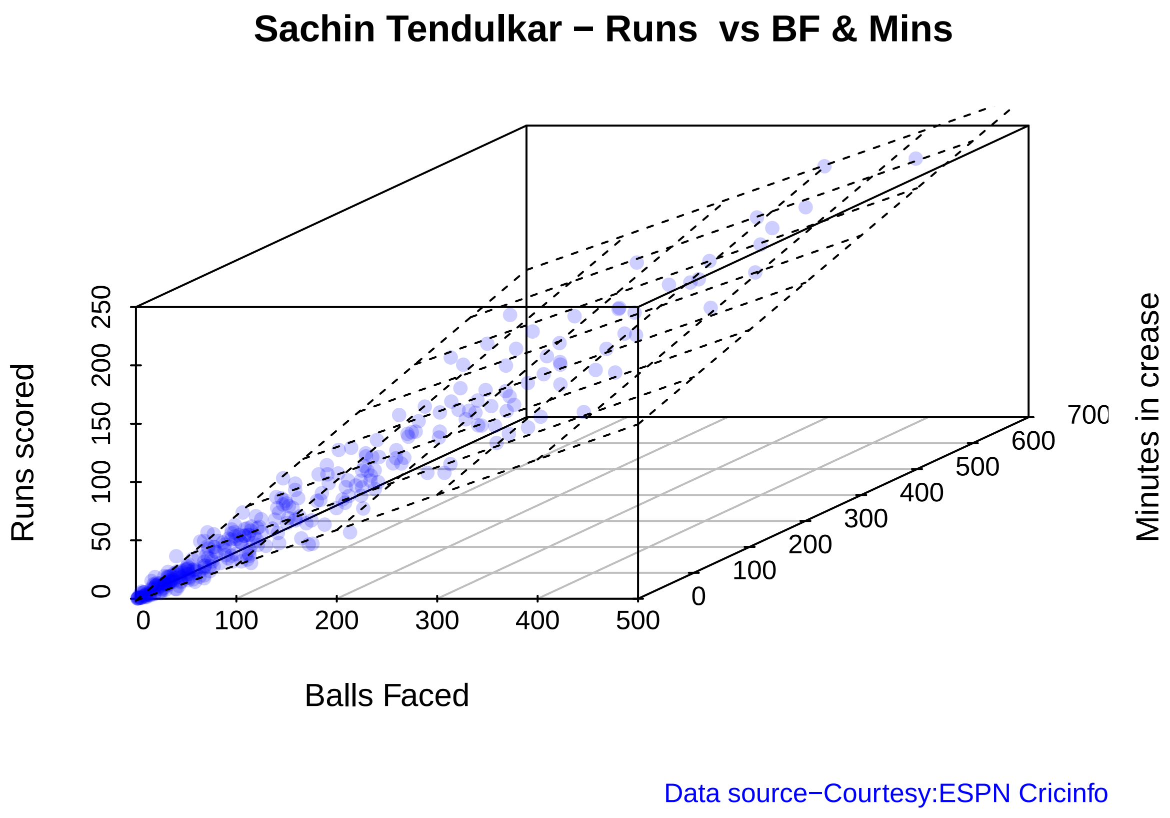
```
dev.off()
```

```
## null device
##           1
```

3D scatter plot and prediction plane

The plots below show the 3D scatter plot of Sachin Runs versus Balls Faced and Minutes at crease. A linear regression model is then fitted between Runs and Balls Faced + Minutes at crease

```
battingPerf3d("./tendulkar.csv","Sachin Tendulkar")
```



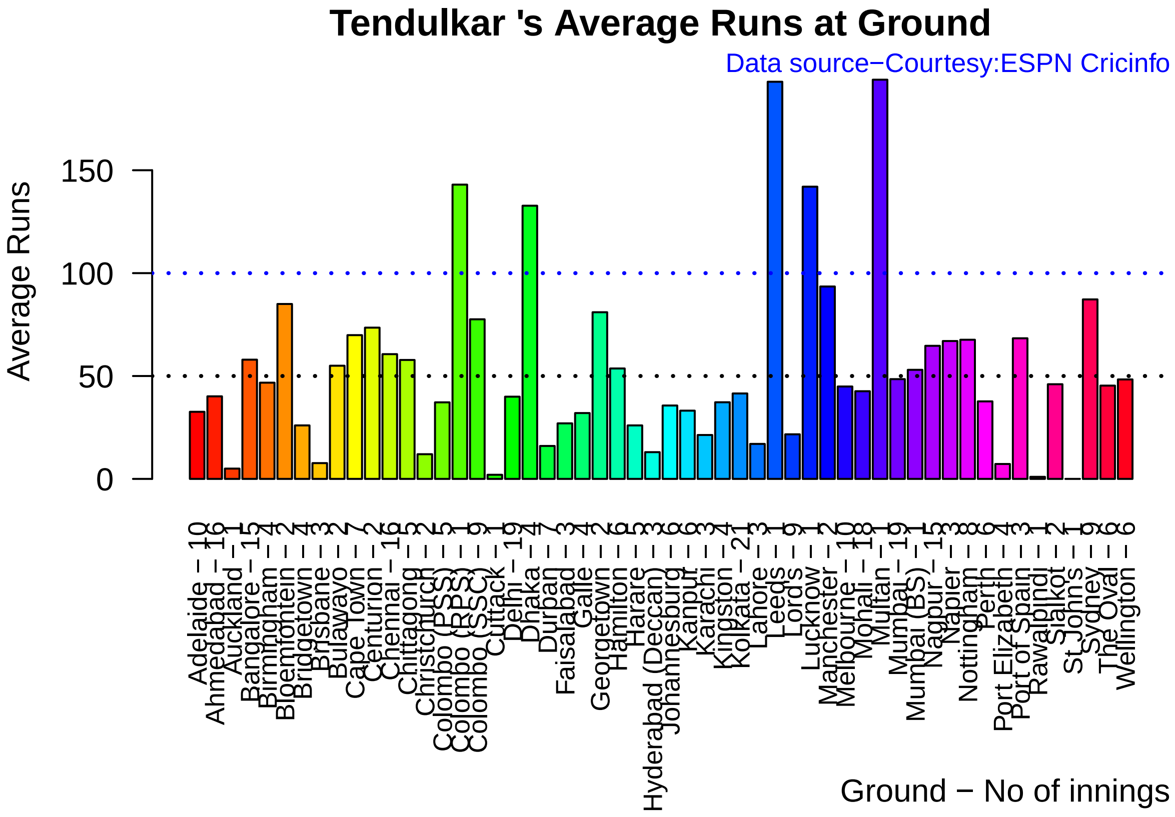
Sachin Tendulkar − Runs vs BF & Mins

Data source−Courtesy:ESPN Cricinfo

Average runs at different venues

The plot below gives the average runs scored by Tendulkar at different grounds. The plot also the number of innings at each ground as a label at x-axis. It can be seen Tendulkar did great in Colombo (SSC), Melbourne overseas and Mumbai, Mohali and Bangalore at home

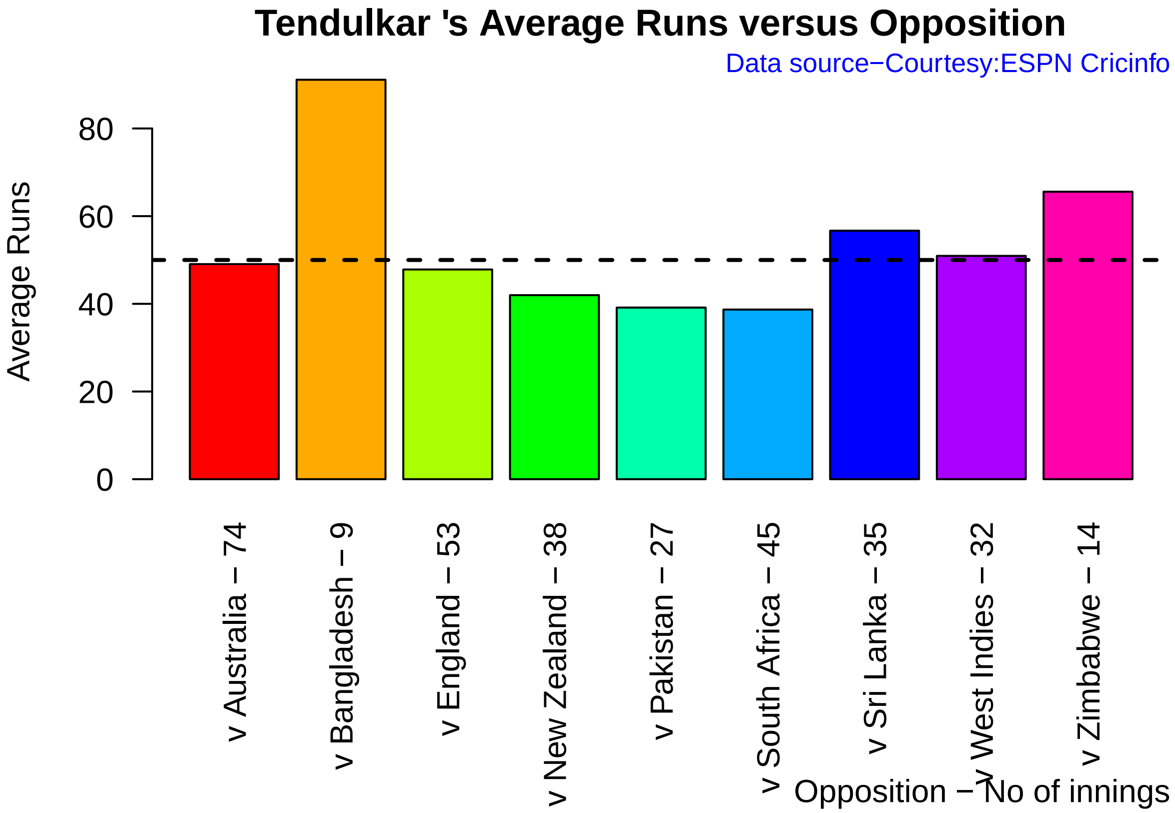
```
batsmanAvgRunsGround("./tendulkar.csv","Tendulkar")
```



Average runs against different opposing teams

This plot computes the average runs scored by Tendulkar against different countries. The x-axis also gives the number of innings against each team

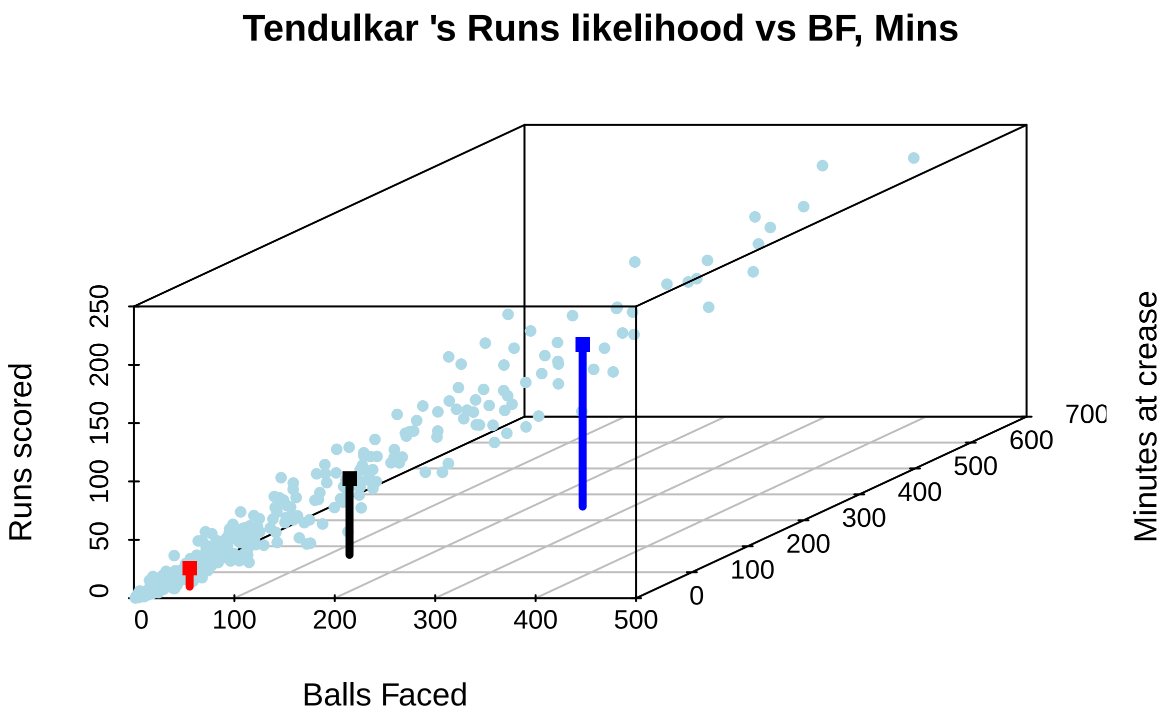
```
batsmanAvgRunsOpposition("./tendulkar.csv","Tendulkar")
```



Highest Runs Likelihood

The plot below shows the Runs Likelihood for a batsman. For this the performance of Sachin is plotted as a 3D scatter plot with Runs versus Balls Faced + Minutes at crease. K-Means. The centroids of 3 clusters are conputed and plotted. In this plot Sachin Tendulkar's highest tendencies are computed and plotted using K-Means

```r
batsmanRunsLikelihood("./tendulkar.csv","Tendulkar")
```



Data source−Courtesy:ESPN Cricinfo

```
## Summary of  Tendulkar 's runs scoring likelihood
## **************************************************
##
## There is a 16.51 % likelihood that Tendulkar  will make  139 Runs in  251 balls over 353  Minutes
## There is a 58.41 % likelihood that Tendulkar  will make  16 Runs in  31 balls over  44  Minutes
## There is a 25.08 % likelihood that Tendulkar  will make  66 Runs in  122 balls over 167  Minutes
```

A look at the Top 4 batsman - Tendulkar, Kallis, Ponting and Sangakkara

The batsmen with the most hundreds in test cricket are

1. Sachin Tendulkar :**Average:53.78,100's - 51, 50's - 68**
2. Jacques Kallis : **Average: 55.47, 100's - 45, 50's - 58**
3. Ricky Ponting : **Average: 51.85, 100's - 41 , 50's - 62**
4. Kumara Sangakarra: **Average: 58.04 ,100's - 38 , 50's - 52**

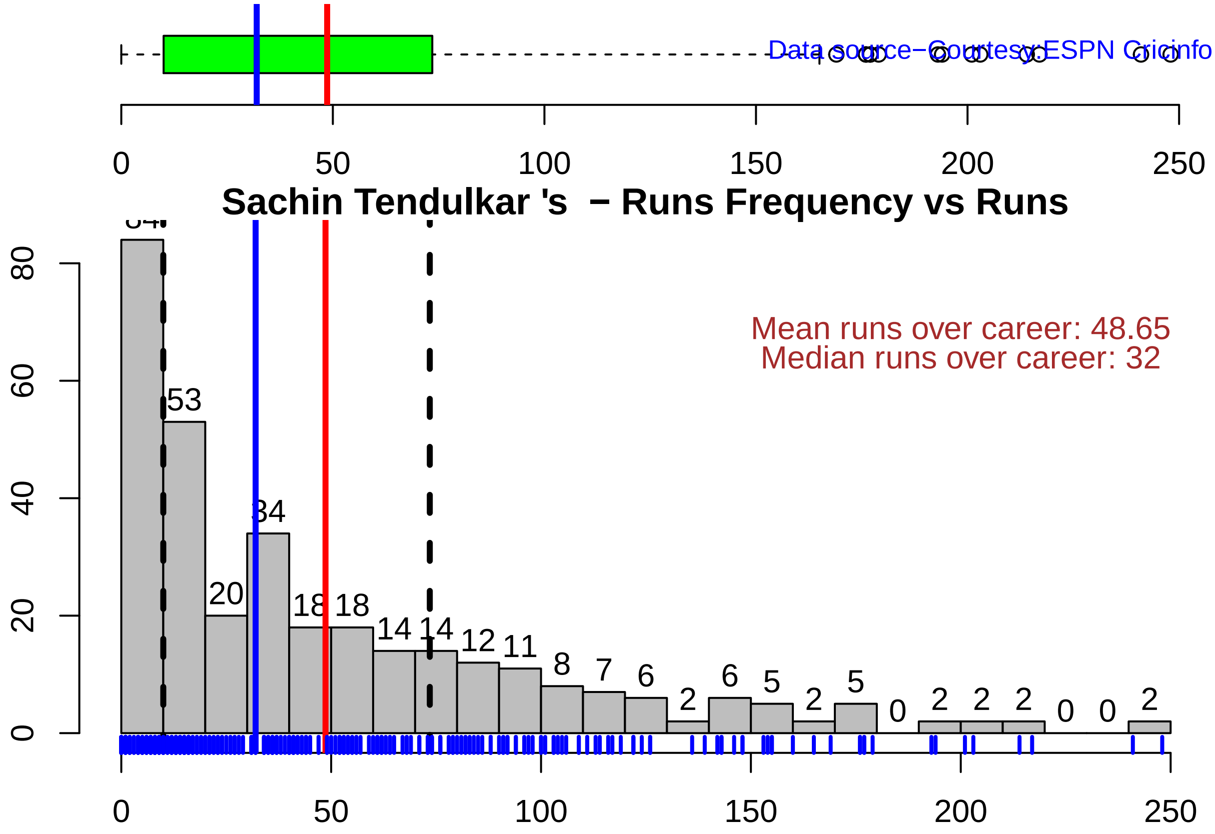
in that order.

The following plots take a closer at their performances. The box plots show the mean (red line) and median (blue line). The two ends of the boxplot display the 25th and 75th percentile.

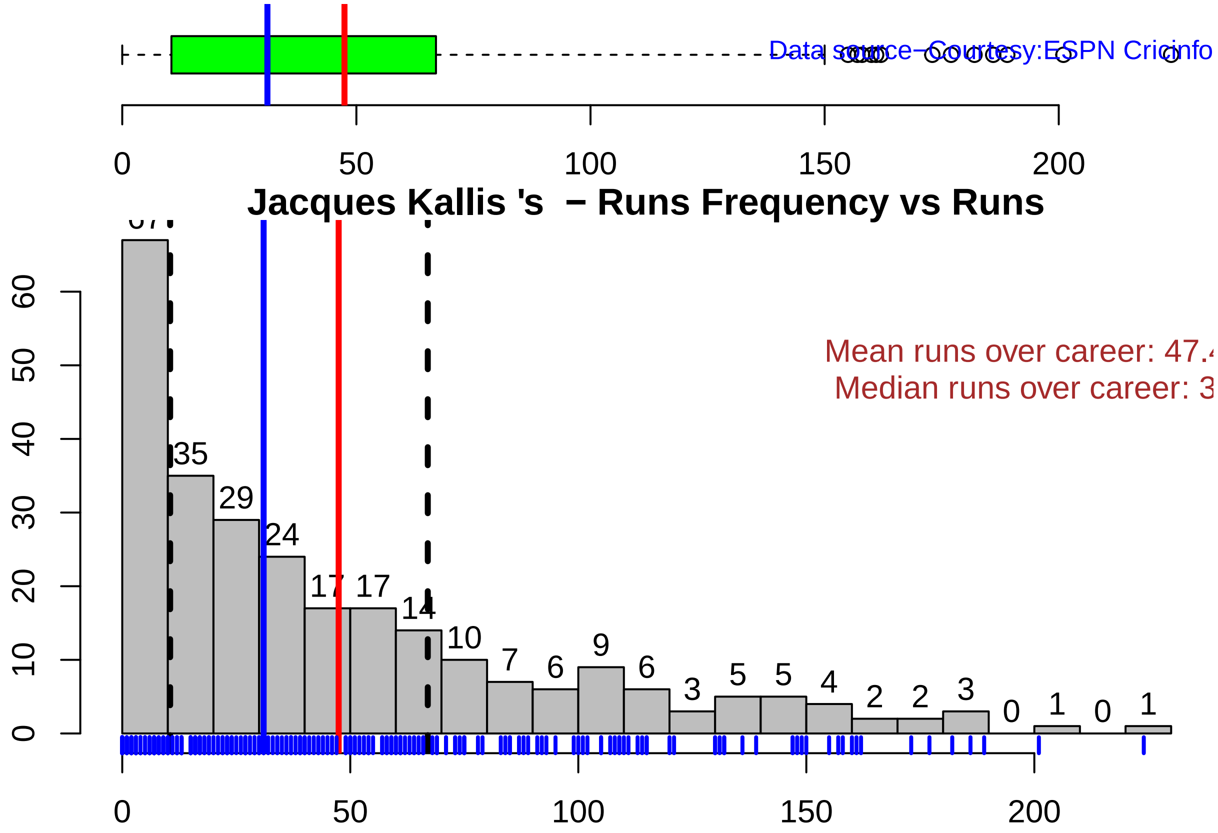
Box Histogram Plot

This plot shows a combined boxplot of the Runs ranges and a histogram of the Runs Frequency

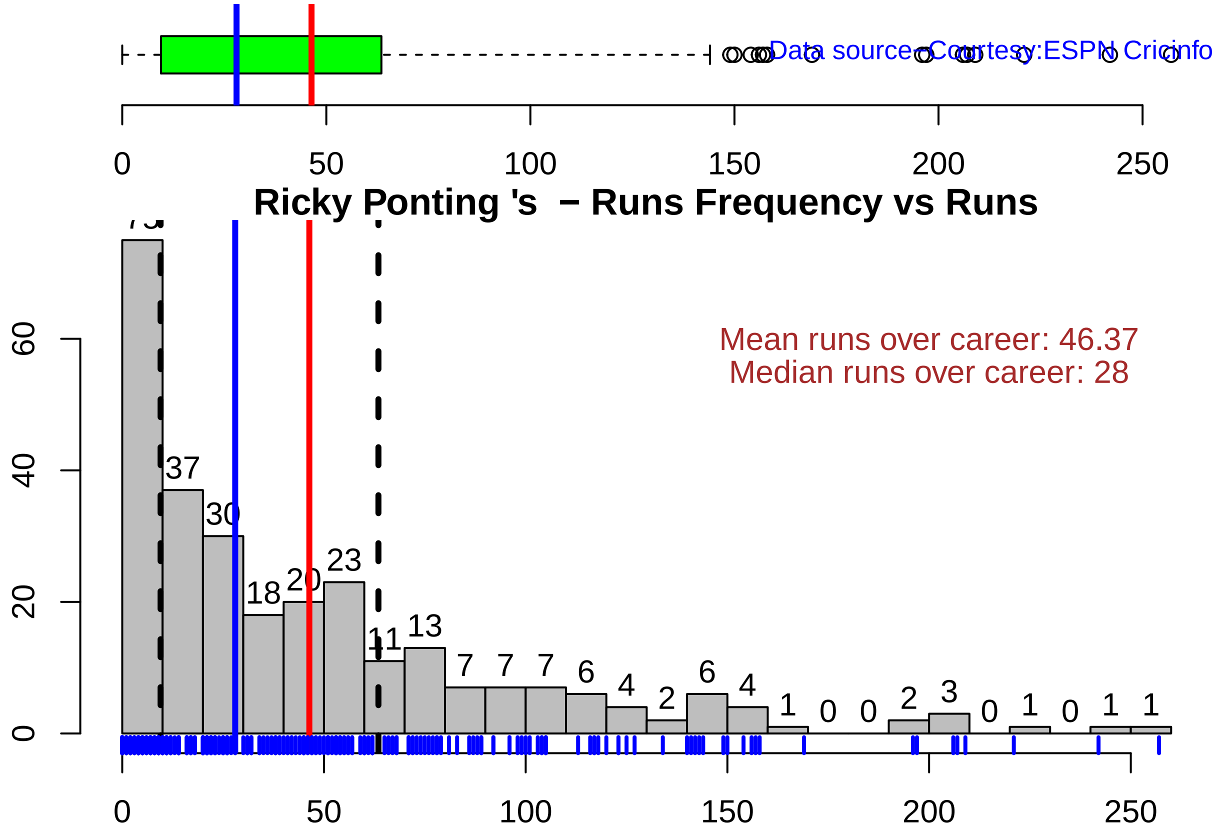
```
batsmanPerfBoxHist("./tendulkar.csv","Sachin Tendulkar")
```



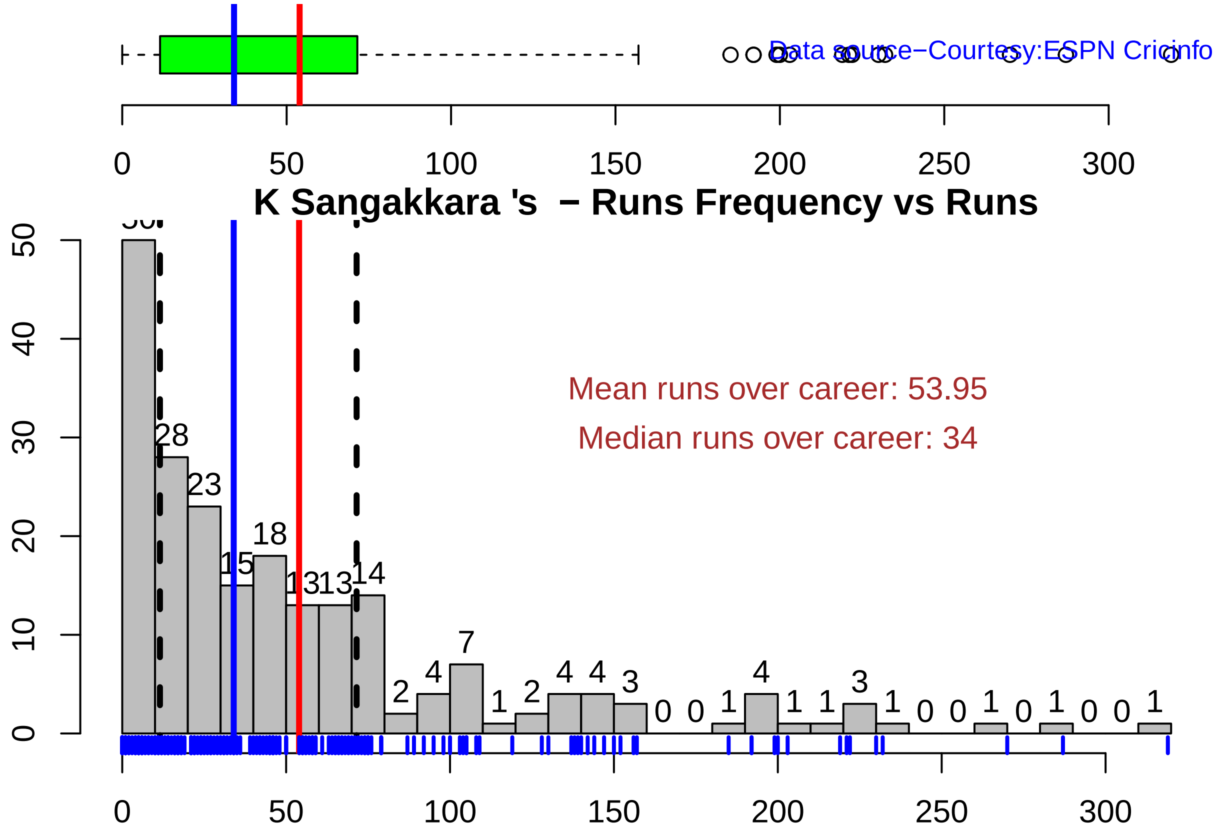
```
batsmanPerfBoxHist("./kallis.csv","Jacques Kallis")
```



```
batsmanPerfBoxHist("./ponting.csv","Ricky Ponting")
```



```r
batsmanPerfBoxHist("./sangakkara.csv","K Sangakkara")
```



Contribution to won and lost matches

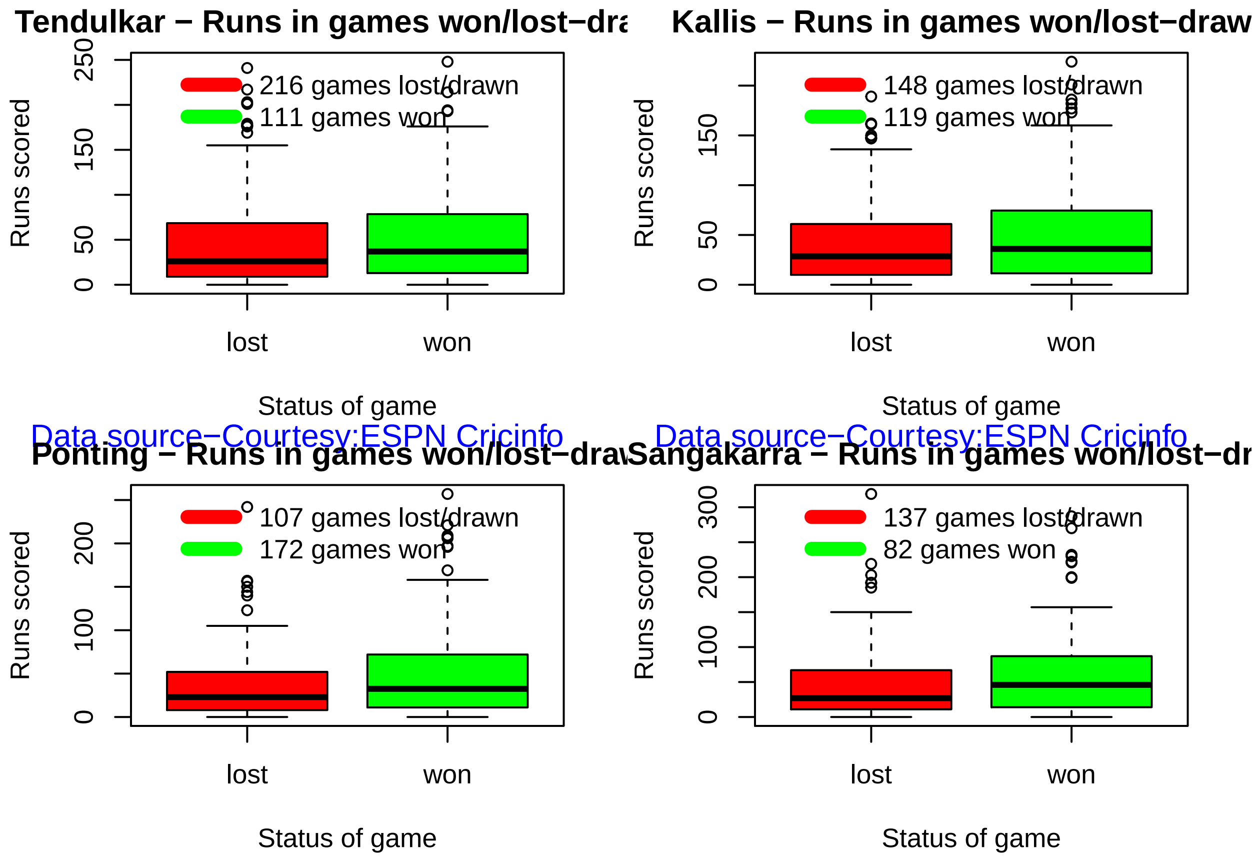
The plot below shows the contribution of Tendulkar, Kallis, Ponting and Sangakarra in matches won and lost. The plots show the range of runs scored as a boxplot (25th & 75th percentile) and the mean scored. The total matches won and lost are also printed in the plot.

All the players have scored more in the matches they won than the matches they lost. Ricky Ponting is the only batsman who seems to have more matches won to his credit than others. This could also be because he was a member of strong Australian team

For the 2 functions below you will have to use the getPlayerDataSp() function. I have commented this as I already have these files

```
#tendulkarsp <- getPlayerDataSp(35320,tdir=".",tfile="tendulkarsp.csv",ttype="batting")
#kallissp <- getPlayerDataSp(45789,tdir=".",tfile="kallissp.csv",ttype="batting")
#pontingsp <- getPlayerDataSp(7133,tdir=".",tfile="pontingsp.csv",ttype="batting")
#sangakkarasp <- getPlayerDataSp(50710,tdir=".",tfile="sangakkarasp.csv",ttype="batting")
```

```
par(mfrow=c(2,2))
par(mar=c(4,4,2,2))
batsmanContributionWonLost("tendulkarsp.csv","Tendulkar")
batsmanContributionWonLost("kallissp.csv","Kallis")
batsmanContributionWonLost("pontingsp.csv","Ponting")
batsmanContributionWonLost("sangakkarasp.csv","Sangakarra")
```



```
dev.off()
```

```
## null device 
##           1
```

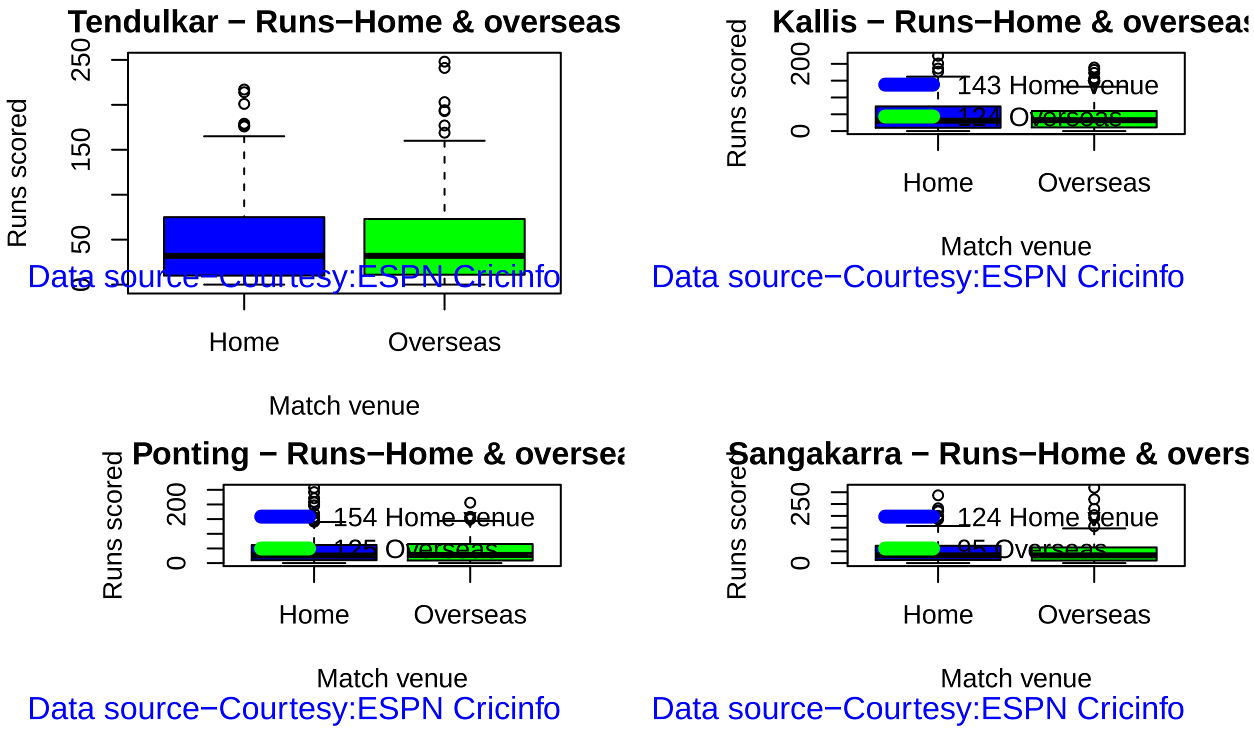
Performance at home and overseas

From the plot below it can be seen

Tendulkar has more matches overseas than at home and his performace is consistent in all venues at home or abroad. Ponting has lesser innings than Tendulkar and has an equally good performance at home and overseas.Kallis and Sangakkara's performance abroad is lower than the performance at home.

This function also requires the use of getPlayerDataSp() as shown above

```
par(mfrow=c(2,2))
par(mar=c(4,4,2,2))
batsmanPerfHomeAway("tendulkarsp.csv","Tendulkar")
batsmanPerfHomeAway("kallissp.csv","Kallis")
batsmanPerfHomeAway("pontingsp.csv","Ponting")
batsmanPerfHomeAway("sangakkarasp.csv","Sangakarra")
```



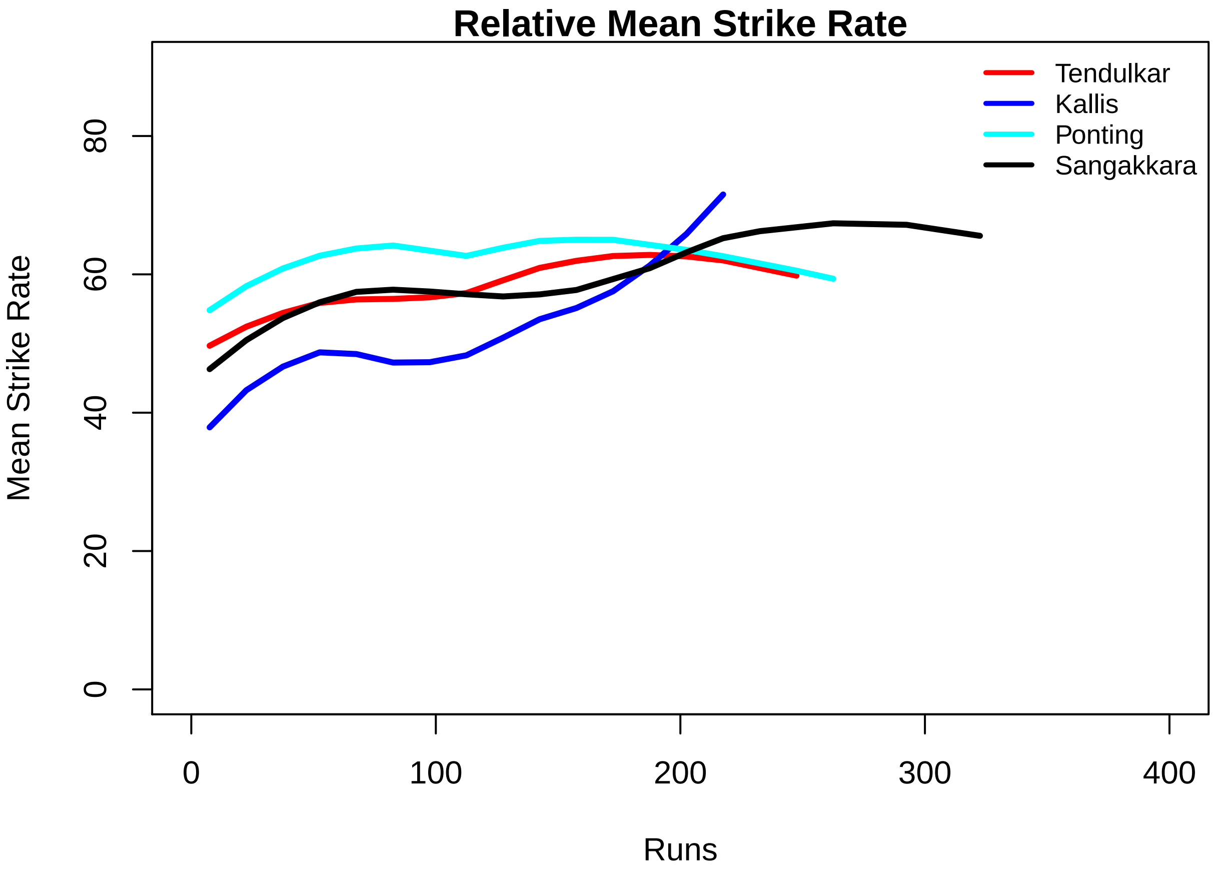
```
dev.off()
```

```
## null device
##           1
```

Relative Mean Strike Rate plot

The plot below compares the Mean Strike Rate of the batsman for each of the runs ranges of 10 and plots them. The plot indicate the following Range 0 - 50 Runs - Ponting leads followed by Tendulkar Range 50 -100 Runs - Ponting followed by Sangakkara Range 100 - 150 - Ponting and then Tendulkar

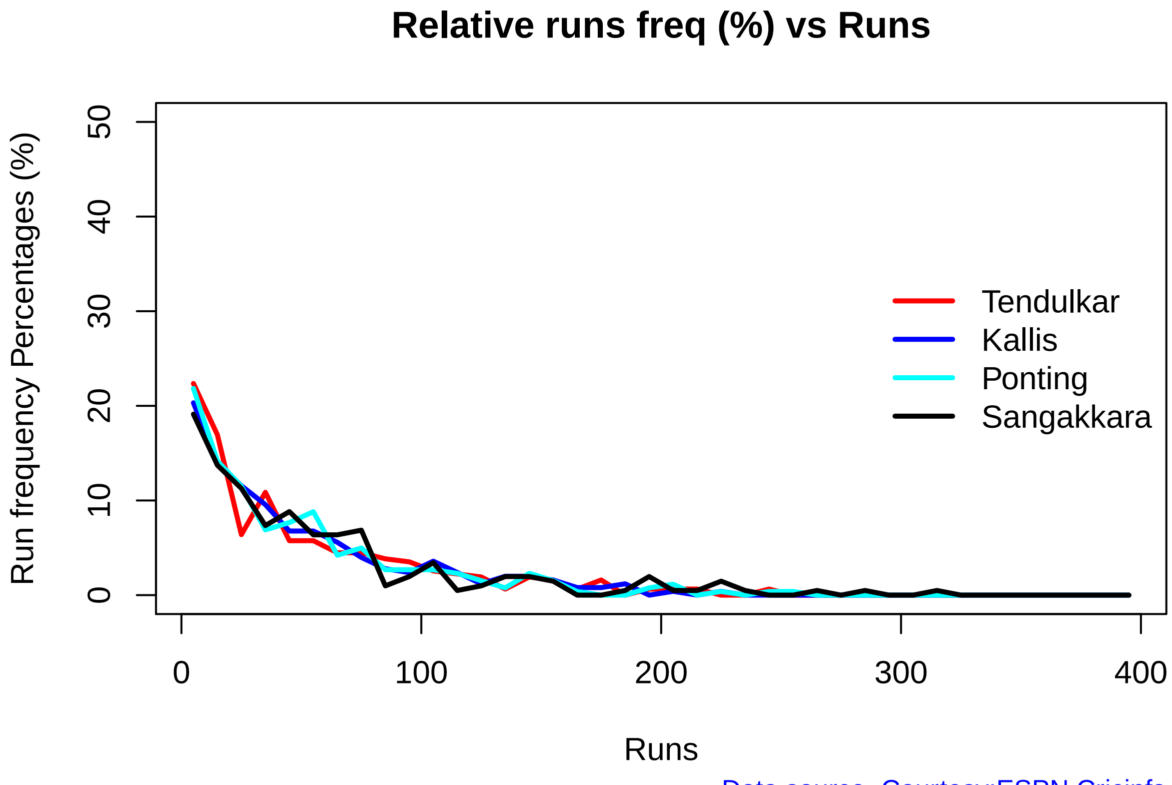
```
frames <- list("./tendulkar.csv","./kallis.csv","ponting.csv","sangakkara.csv")
names <- list("Tendulkar","Kallis","Ponting","Sangakkara")
relativeBatsmanSR(frames,names)
```



Relative Runs Frequency plot

The plot below gives the relative Runs Frequency Percetages for each 10 run bucket. The plot below show Sangakkara leads followed by Ponting

```
frames <- list("./tendulkar.csv","./kallis.csv","ponting.csv","sangakkara.csv")
names <- list("Tendulkar","Kallis","Ponting","Sangakkara")
relativeRunsFreqPerf(frames,names)
```

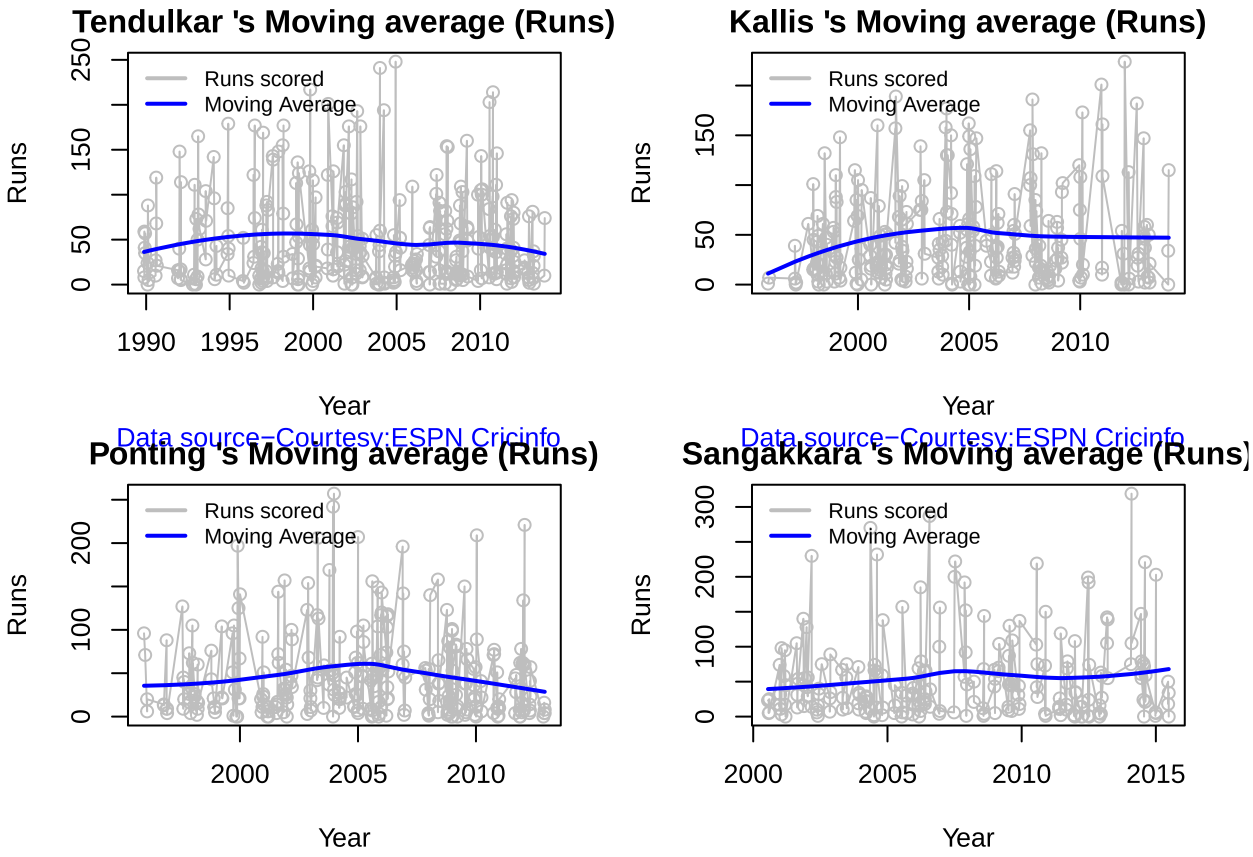


Data source−Courtesy:ESPN Cricinfo

Moving Average of runs in career

Take a look at the Moving Average across the career of the Top 4. Clearly . Kallis and Sangakkara have a few more years of great batting ahead. They seem to average on 50. . Tendulkar and Ponting definitely show a slump in the later years

```
par(mfrow=c(2,2))
par(mar=c(4,4,2,2))
batsmanMovingAverage("./tendulkar.csv","Tendulkar")
batsmanMovingAverage("./kallis.csv","Kallis")
batsmanMovingAverage("./ponting.csv","Ponting")
batsmanMovingAverage("./sangakkara.csv","Sangakkara")
```



```r
dev.off()
```

```
## null device
##           1
```

Future Runs forecast

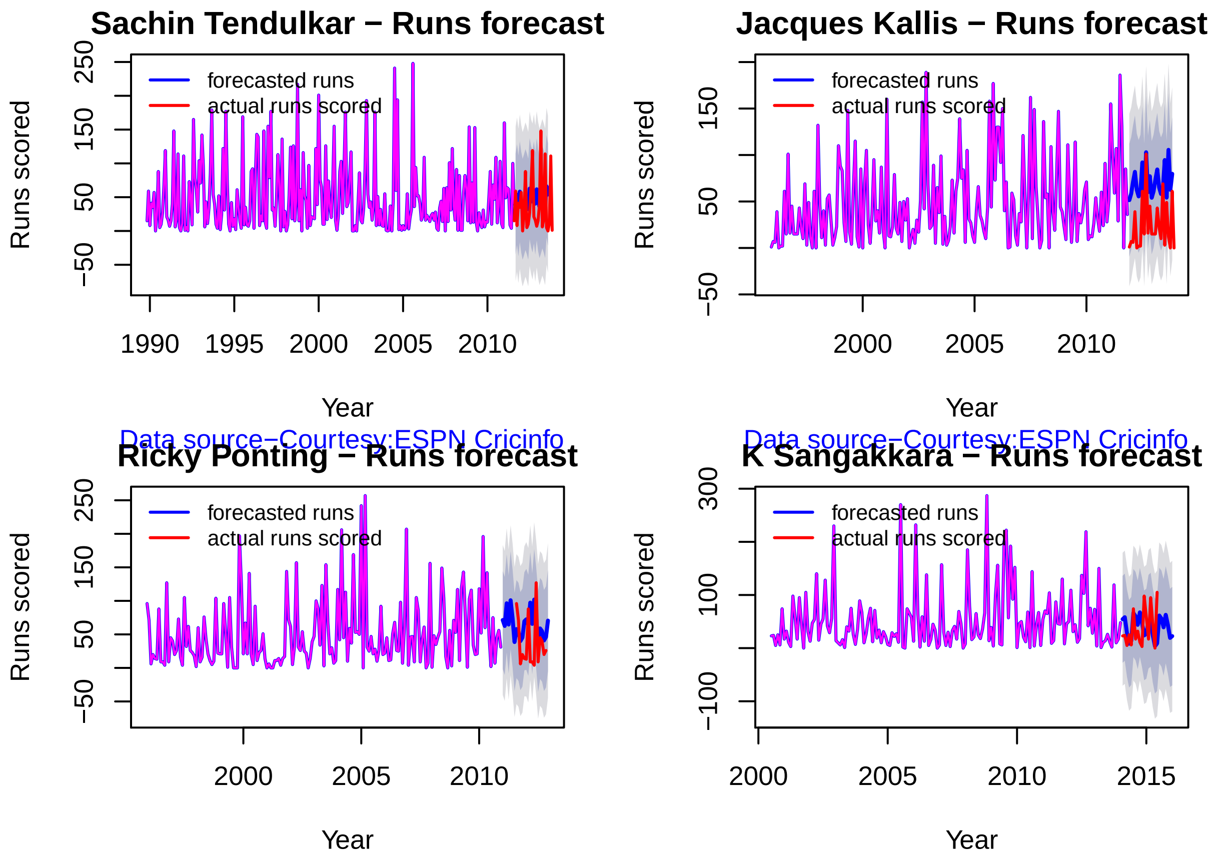
Here are plots that forecast how the batsman will perform in future. In this case 90% of the career runs trend is uses as the training set. the remaining 10% is the test set.

A Holt-Winters forecating model is used to forecast future performance based on the 90% training set. The forecated runs trend is plotted. The test set is also plotted to see how close the forecast and the actual matches

Take a look at the runs forecasted for the batsman below.

- Tendulkar's forecasted performance seems to tally with his actual performance with an average of 50
- Kallis the forecasted runs are higher than the actual runs he scored
- Ponting seems to have a good run in the future
- Sangakkara has a decent run in the future averaging 50 runs

```r
par(mfrow=c(2,2))
par(mar=c(4,4,2,2))
batsmanPerfForecast("./tendulkar.csv","Sachin Tendulkar")
batsmanPerfForecast("./kallis.csv","Jacques Kallis")
batsmanPerfForecast("./ponting.csv","Ricky Ponting")
batsmanPerfForecast("./sangakkara.csv","K Sangakkara")
```



```
dev.off()
```

```
## null device
##           1
```

Check Batsman In-Form or Out-of-Form

The below computation uses Null Hypothesis testing and p-value to determine if the batsman is in-form or out-of-form. For this 90% of the career runs is chosen as the population and the mean computed. The last 10% is chosen to be the sample set and the sample Mean and the sample Standard Deviation are caculated.

The Null Hypothesis (H0) assumes that the batsman continues to stay in-form where the sample mean is within 95% confidence interval of population mean The Alternative (Ha) assumes that the batsman is out of form the sample mean is beyond the 95% confidence interval of the population mean.

A significance value of 0.05 is chosen and p-value us computed If p-value >= .05 - Batsman In-Form If p-value < 0.05 - Batsman Out-of-Form

Note Ideally the p-value should be done for a population that follows the Normal Distribution. But the runs population is usually left skewed. So some correction may be needed. I will revisit this later

This is done for the Top 4 batsman

```r
checkBatsmanInForm("./tendulkar.csv","Sachin Tendulkar")
```

```
## [1] "**************************** Form status of Sachin Tendulkar ****************************\n\n Po
```

```r
checkBatsmanInForm("./kallis.csv","Jacques Kallis")
```

```
## [1] "**************************** Form status of Jacques Kallis ****************************\n\n Popu
```

```r
checkBatsmanInForm("./ponting.csv","Ricky Ponting")
```

```
## [1] "**************************** Form status of Ricky Ponting ****************************\n\n Popul
```

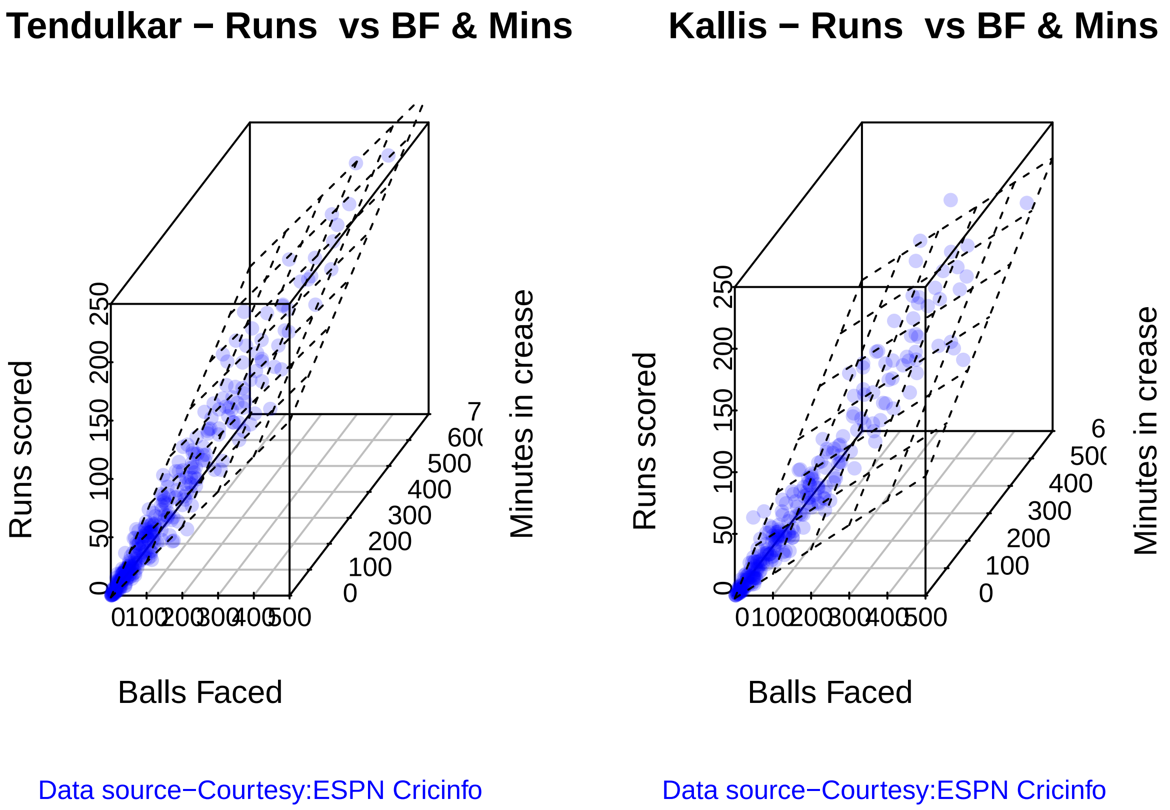
```r
checkBatsmanInForm("./sangakkara.csv","K Sangakkara")
```

```
## [1] "**************************** Form status of K Sangakkara ****************************\n\n Popula
```

3D plot of Runs vs Balls Faced and Minutes at Crease

The plot is a scatter plot of Runs vs Balls faced and Minutes at Crease. A prediction plane is fitted

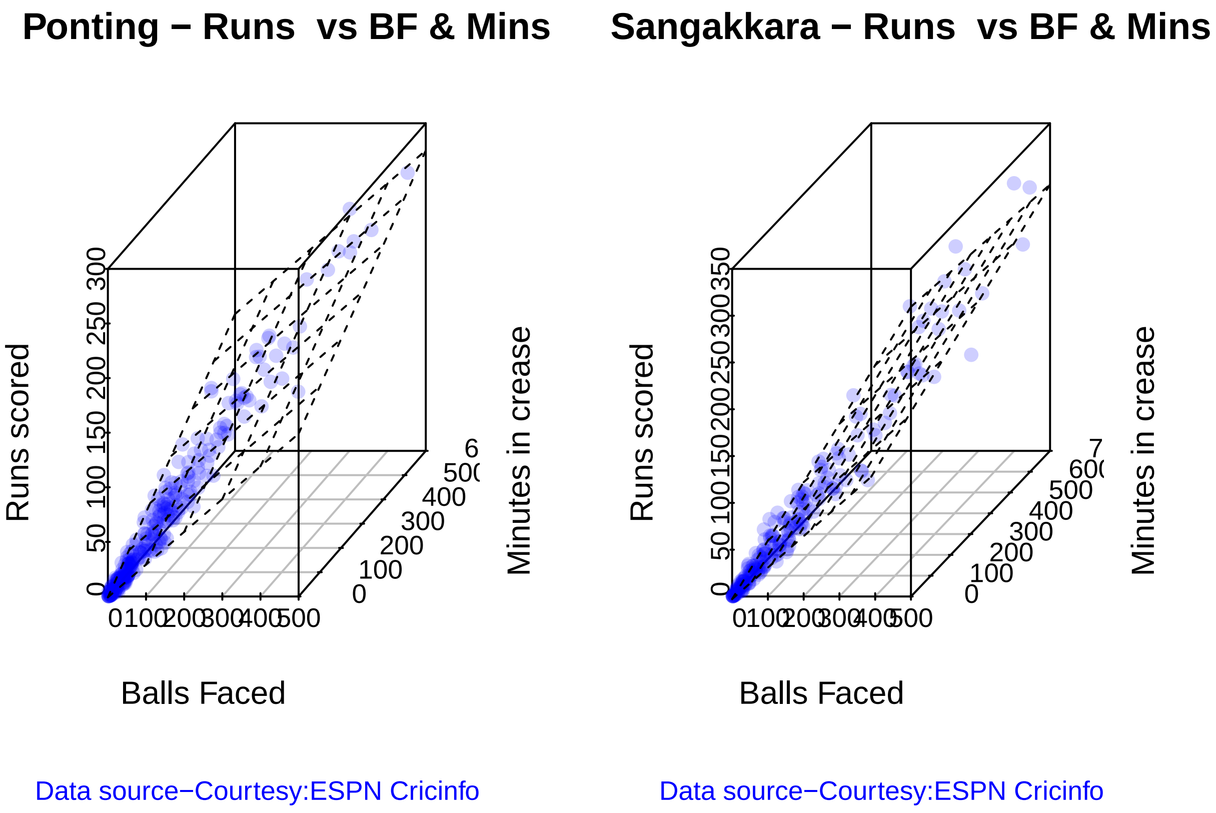
```r
par(mfrow=c(1,2))
par(mar=c(4,4,2,2))
battingPerf3d("./tendulkar.csv","Tendulkar")
battingPerf3d("./kallis.csv","Kallis")
```



Data source−Courtesy:ESPN Cricinfo

Data source−Courtesy:ESPN Cricinfo

```
par(mfrow=c(1,2))
par(mar=c(4,4,2,2))
battingPerf3d("./ponting.csv","Ponting")
battingPerf3d("./sangakkara.csv","Sangakkara")
```



```
dev.off()
```

```
## null device
##           1
```

Predicting Runs given Balls Faced and Minutes at Crease

A multi-variate regression plane is fitted between Runs and Balls faced +Minutes at crease.

```
BF <- seq( 10, 400,length=15)
Mins <- seq(30,600,length=15)
newDF <- data.frame(BF,Mins)
tendulkar <- batsmanRunsPredict("./tendulkar.csv","Tendulkar",newdataframe=newDF)
kallis <- batsmanRunsPredict("./kallis.csv","Kallis",newdataframe=newDF)
ponting <- batsmanRunsPredict("./ponting.csv","Ponting",newdataframe=newDF)
sangakkara <- batsmanRunsPredict("./sangakkara.csv","Sangakkara",newdataframe=newDF)
```

The fitted model is then used to predict the runs that the batsmen will score for a given Balls faced and Minutes at crease. It can be seen Ponting has the will score the highest for a given Balls Faced and Minutes at crease.

Ponting is followed by Tendulkar who has Sangakkara close on his heels and finally we have Kallis. This is intuitive as we have already seen that Ponting has a highest strike rate.

```
batsmen <-cbind(round(tendulkar$Runs),round(kallis$Runs),round(ponting$Runs),round(sangakkara$Runs))
colnames(batsmen) <- c("Tendulkar","Kallis","Ponting","Sangakkara")
newDF <- data.frame(round(newDF$BF),round(newDF$Mins))
colnames(newDF) <- c("BallsFaced","MinsAtCrease")
predictedRuns <- cbind(newDF,batsmen)
predictedRuns
```

```
##    BallsFaced MinsAtCrease Tendulkar Kallis Ponting Sangakkara
## 1          10           30         7      6       9          2
## 2          38           71        23     20      25         18
## 3          66          111        39     34      42         34
## 4          94          152        54     48      59         50
## 5         121          193        70     62      76         66
## 6         149          234        86     76      93         82
## 7         177          274       102     90     110         98
## 8         205          315       118    104     127        114
## 9         233          356       134    118     144        130
## 10        261          396       150    132     161        146
## 11        289          437       165    146     178        162
## 12        316          478       181    159     194        178
## 13        344          519       197    173     211        194
## 14        372          559       213    187     228        210
## 15        400          600       229    201     245        226
```

Analysis of Top 3 wicket takers

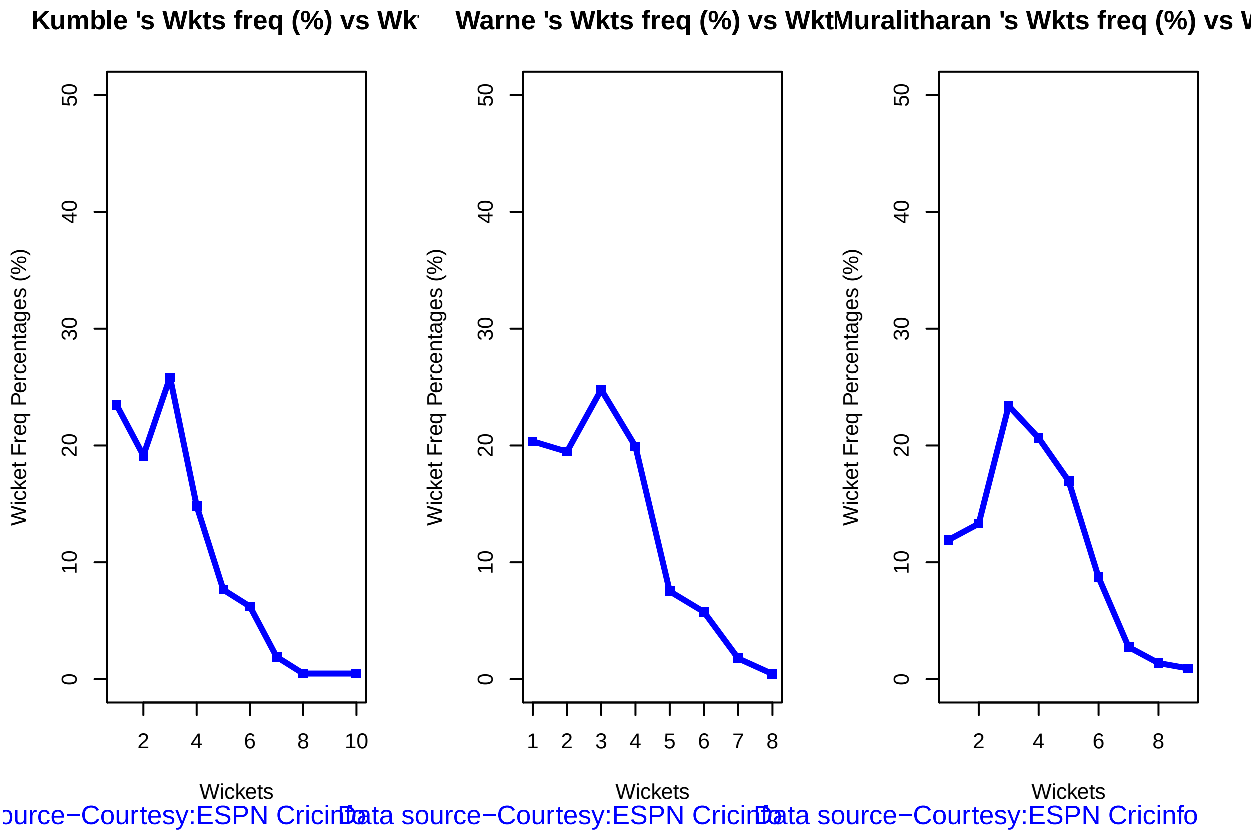
The top 3 wicket takes in test history are 1. M Muralitharan:Wickets: 800, Average = 22.72, Economy Rate - 2.47 2. Shane Warne: Wickets: 708, Average = 25.41, Economy Rate - 2.65 3. Anil Kumble: Wickets: 619, Average = 29.65, Economy Rate - 2.69

How do Anil Kumble, Shane Warne and M Muralitharan compare with one another with respect to wickets taken and the Economy Rate. The next set of plots compute and plot precisely these analyses.

Wicket Frequency Plot

This plot below computes the percentage frequency of number of wickets taken for e.g 1 wicket x%, 2 wickets y% etc and plots them as a continuous line

```
par(mfrow=c(1,3))
par(mar=c(4,4,2,2))
bowlerWktsFreqPercent("./kumble.csv","Kumble")
bowlerWktsFreqPercent("./warne.csv","Warne")
bowlerWktsFreqPercent("./murali.csv","Muralitharan")
```

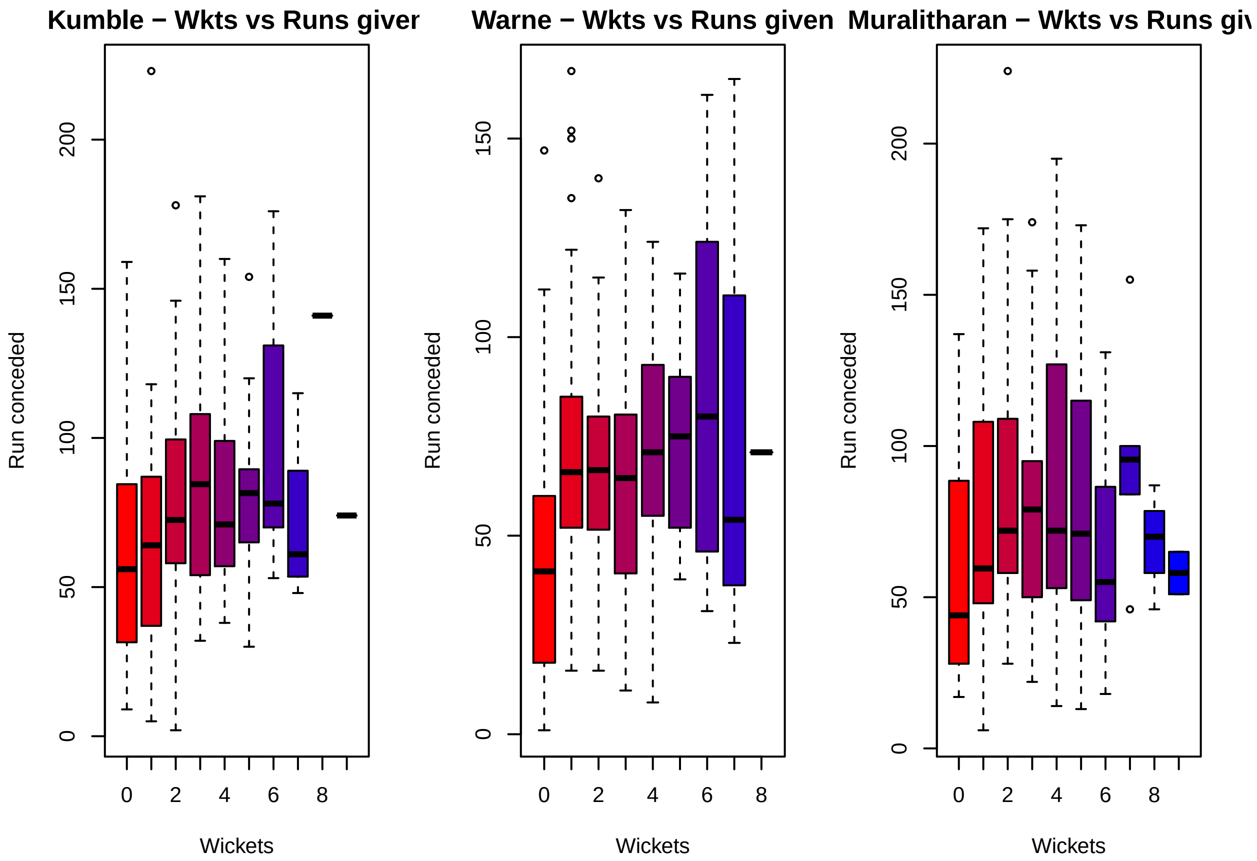


```
dev.off()
```

```
## null device
##           1
```

Wickets Runs plot

```
par(mfrow=c(1,3))
par(mar=c(4,4,2,2))
bowlerWktsRunsPlot("./kumble.csv","Kumble")
bowlerWktsRunsPlot("./warne.csv","Warne")
bowlerWktsRunsPlot("./murali.csv","Muralitharan")
```



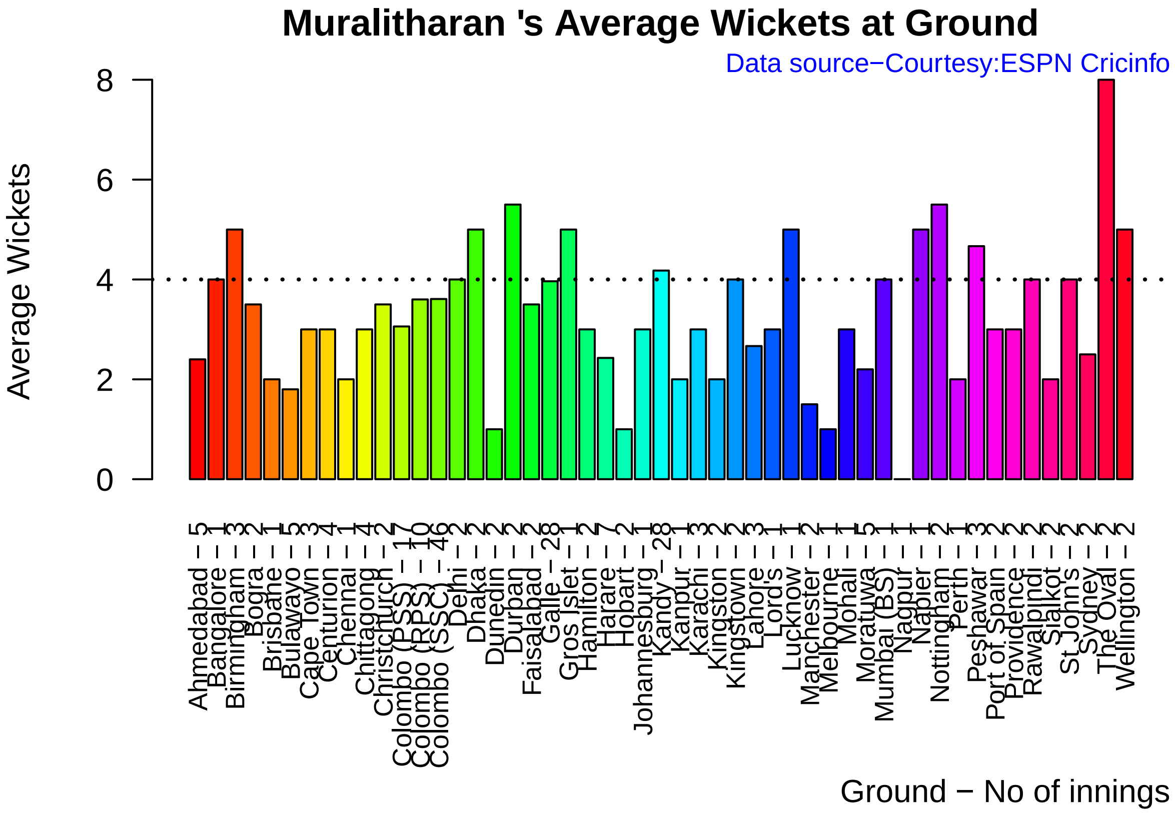
```
dev.off()
```

```
## null device
##           1
```

Average wickets at different venues

The plot gives the average wickets taken by Muralitharan at different venues. Muralitharan has taken an average of 8 and 6 wickets at Oval & Wellington respectively in 2 different innings. His best performances are at Kandy and Colombo (SSC)

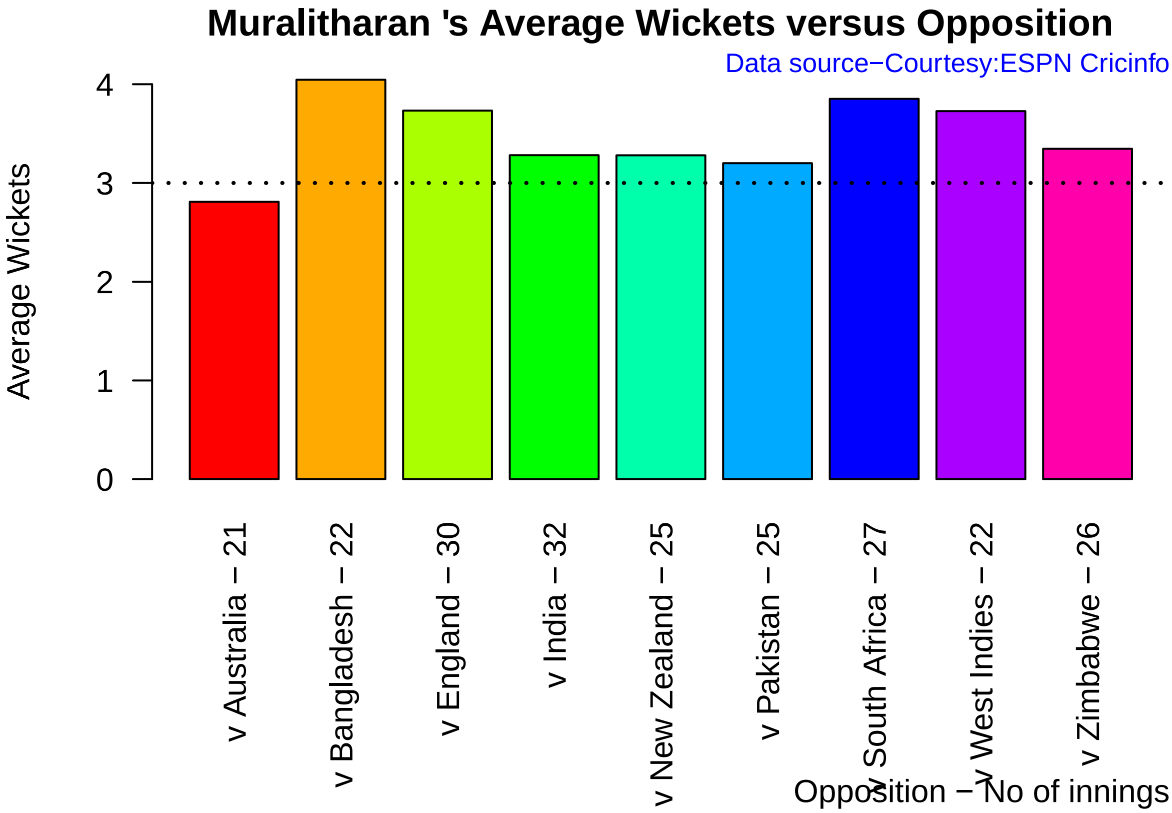
```
bowlerAvgWktsGround("./murali.csv","Muralitharan")
```



Average wickets against different opposition

The plot gives the average wickets taken by Muralitharan against different countries. The x-axis also includes the number of innings against each team

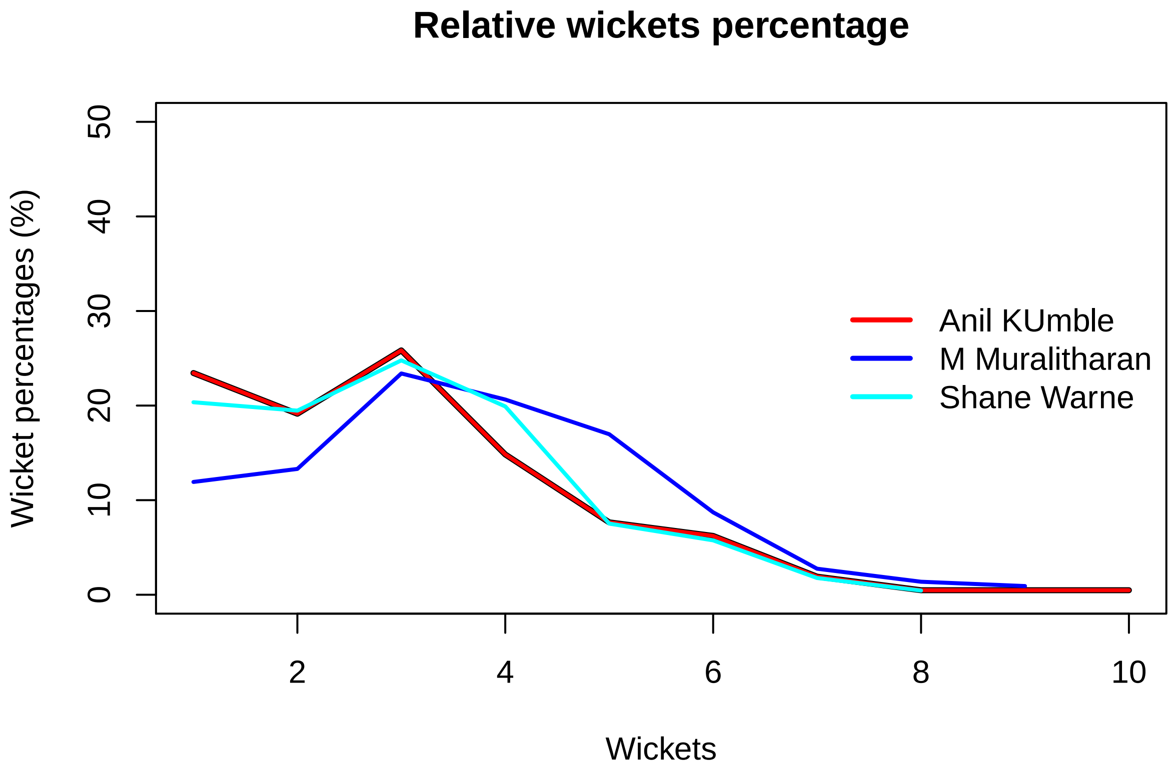
```
bowlerAvgWktsOpposition("./murali.csv","Muralitharan")
```



Relative Wickets Frequency Percentage

The Relative Wickets Percentage plot shows that M Muralitharan has a large percentage of wickets in the 3-8 wicket range

```
frames <- list("./kumble.csv","./murali.csv","warne.csv")
names <- list("Anil KUmble","M Muralitharan","Shane Warne")
relativeBowlingPerf(frames,names)
```

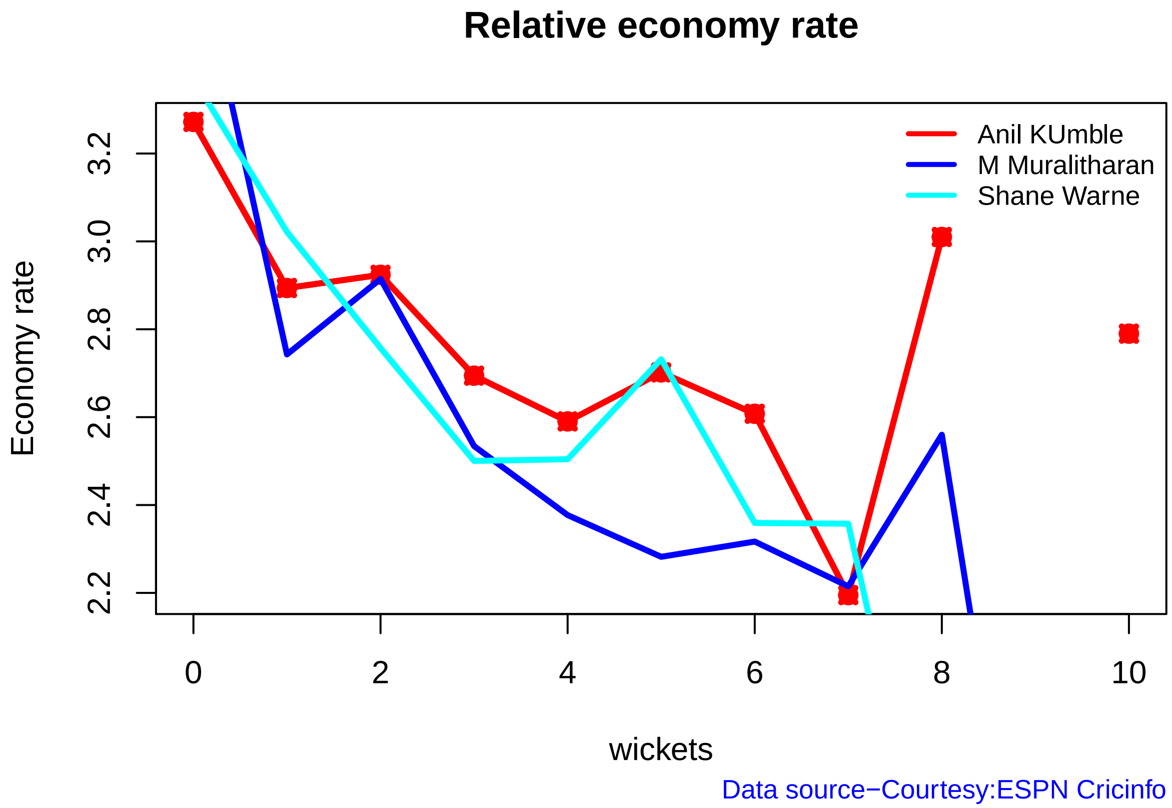


Data source−Courtesy:ESPN Cricinfo

Relative Economy Rate against wickets taken

Clearly from the plot below it can be seen that Muralitharan has the best Economy Rate among the three

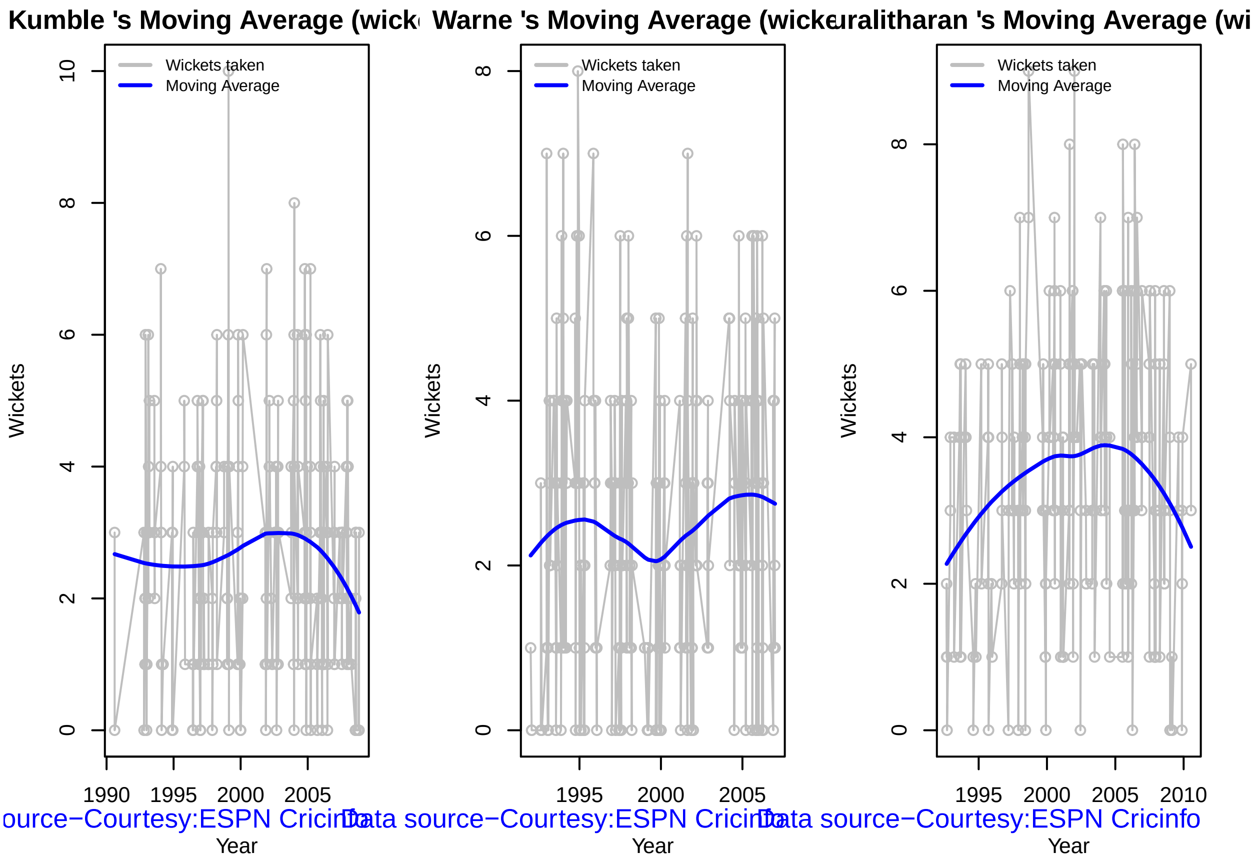
```
frames <- list("./kumble.csv","./murali.csv","warne.csv")
names <- list("Anil KUmble","M Muralitharan","Shane Warne")
relativeBowlingER(frames,names)
```



Wickets taken moving average

From th eplot below it can be see 1. Shane Warne's performance at the time of his retirement was still at a peak of 3 wickets 2. M Muralitharan seems to have become ineffective over time with his peak years being 2004-2006 3. Anil Kumble also seems to slump down and become less effective.

```
par(mfrow=c(1,3))
par(mar=c(4,4,2,2))
bowlerMovingAverage("./kumble.csv","Kumble")
bowlerMovingAverage("./warne.csv","Warne")
bowlerMovingAverage("./murali.csv","Muralitharan")
```



```r
dev.off()
```

```
## null device
##           1
```

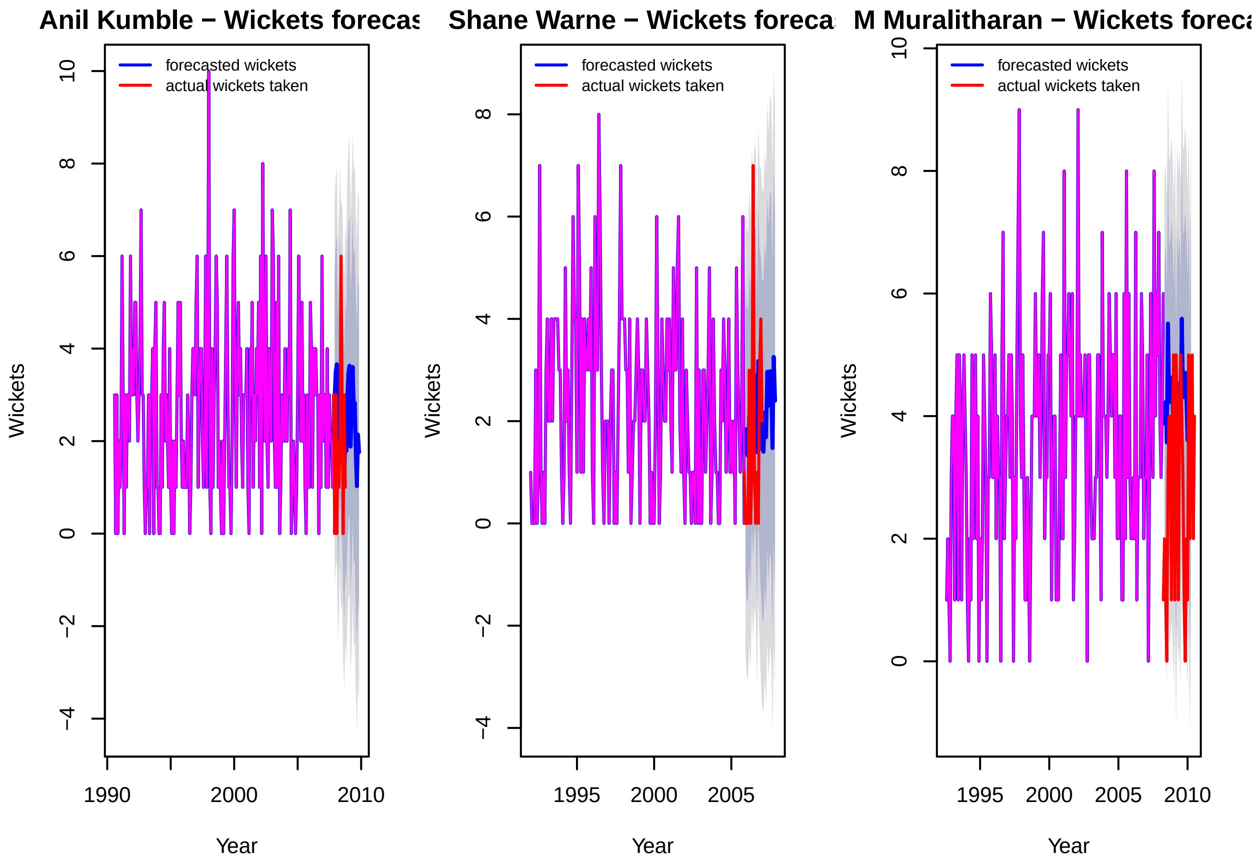
Future Wickets forecast

Here are plots that forecast how the bowler will perform in future. In this case 90% of the career wickets trend is used as the training set. the remaining 10% is the test set.

A Holt-Winters forecating model is used to forecast future performance based on the 90% training set. The forecated wickets trend is plotted. The test set is also plotted to see how close the forecast and the actual matches

Take a look at the wickets forecasted for the bowlers below. - Shane Warne and Muralitharan have a fairly consistent forecast - Kumble forecast shows a small dip

```r
par(mfrow=c(1,3))
par(mar=c(4,4,2,2))
bowlerPerfForecast("./kumble.csv","Anil Kumble")
bowlerPerfForecast("./warne.csv","Shane Warne")
bowlerPerfForecast("./murali.csv","M Muralitharan")
```



```r
dev.off()
```

```
## null device 
##           1
```

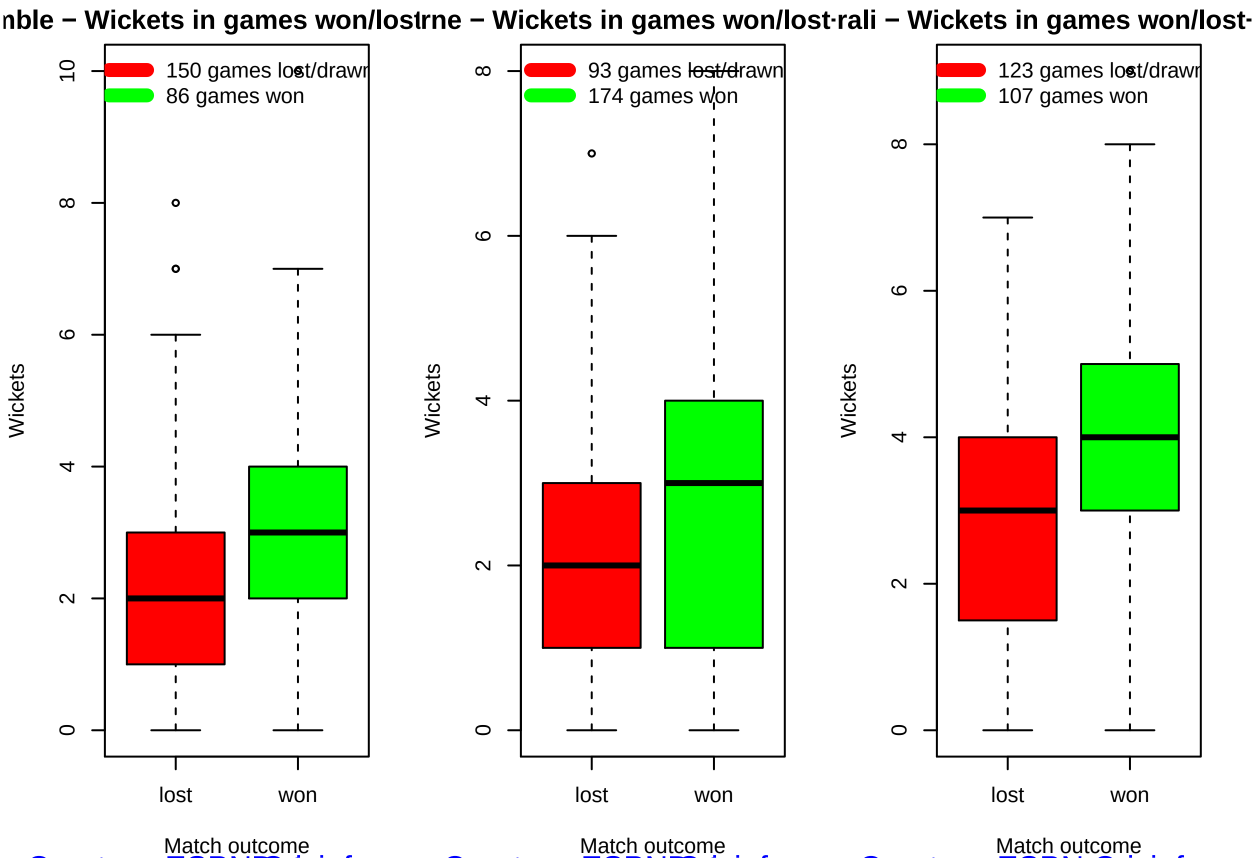
Contribution to matches won and lost

The plot below is extremely interesting 1. Kumble wickets range from 2 to 4 wickets in matches wons with a mean of 3 2. Warne wickets in won matches range from 1 to 4 with more matches won. Clearly there are other bowlers contributing to the wins, possibly the pacers 3. Muralitharan wickets range in winning matches is more than the other 2 and ranges ranges 3 to 5 and clearly had a hand (pun unintended) in Sri Lanka's wins

As discussed above the next 2 charts require the use of getPlayerDataSp()

```r
#kumblesp <- getPlayerDataSp(30176,tdir=".",tfile="kumblesp.csv",ttype="bowling")
#warnesp <- getPlayerDataSp(8166,tdir=".",tfile="warnesp.csv",ttype="bowling")
#muralisp <- getPlayerDataSp(49636,tdir=".",tfile="muralisp.csv",ttype="bowling")
```

```r
par(mfrow=c(1,3))
par(mar=c(4,4,2,2))
bowlerContributionWonLost("kumblesp.csv","Kumble")
bowlerContributionWonLost("warnesp.csv","Warne")
bowlerContributionWonLost("muralisp.csv","Murali")
```



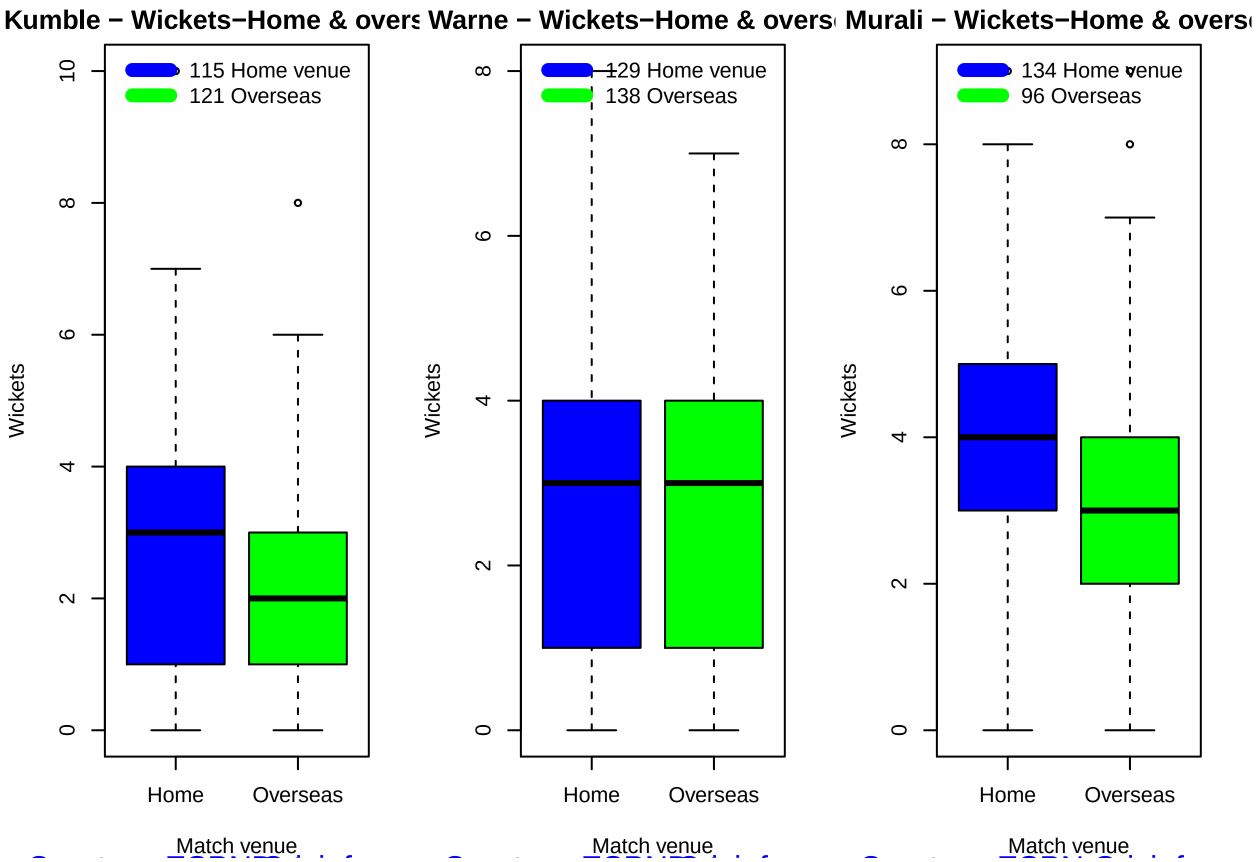
```
dev.off()
```

```
## null device
##           1
```

Performance home and overseas

From the plot below it can be seen that Kumble has played more matches overseas than Warne or Muralitharan. Both Kumble and Warne show an average of 2 wickers overseas, though Kumble has more matches overseas. Murali on the other has an average of 2.5 wickets overseas but a slightly less number of matches than Kumble

```
par(mfrow=c(1,3))
par(mar=c(4,4,2,2))
bowlerPerfHomeAway("kumblesp.csv","Kumble")
bowlerPerfHomeAway("warnesp.csv","Warne")
bowlerPerfHomeAway("muralisp.csv","Murali")
```



```r
dev.off()
```

```
## null device
##           1
```

Check for bowler in-form/out-of-form

The below computation uses Null Hypothesis testing and p-value to determine if the bowler is in-form or out-of-form. For this 90% of the career wickets is chosen as the population and the mean computed. The last 10% is chosen to be the sample set and the sample Mean and the sample Standard Deviation are caculated.

The Null Hypothesis (H0) assumes that the bowler continues to stay in-form where the sample mean is within 95% confidence interval of population mean The Alternative (Ha) assumes that the bowler is out of form the sample mean is beyond the 95% confidence interval of the population mean.

A significance value of 0.05 is chosen and p-value us computed If p-value >= .05 - Batsman In-Form If p-value < 0.05 - Batsman Out-of-Form

Note Ideally the p-value should be done for a population that follows the Normal Distribution. But the runs population is usually left skewed. So some correction may be needed. I will revisit this later

Note: The check for the form status of the bowlers indicate 1. That both Kumble and Muralitharan were out of form. This also shows in the moving average plot 2. Warne is still in great form and could have continued for a few more years. Too bad we didn't see the magic later

```
checkBowlerInForm("./kumble.csv","Anil Kumble")
```

```
## [1] "**************************** Form status of Anil Kumble ****************************\n\n Popula
```

```
checkBowlerInForm("./warne.csv","Shane Warne")
```

```
## [1] "**************************** Form status of Shane Warne ****************************\n\n Popula
```

```
checkBowlerInForm("./murali.csv","M Muralitharan")
```

```
## [1] "**************************** Form status of M Muralitharan ****************************\n\n Popu
```

```
dev.off()
```

```
## null device
##           1
```

Key Findings

The plots above capture some of the capabilities and features of my **cricketr** package. Feel free to install the package and try it out. Please do keep in mind ESPN Cricinfo's Terms of Use.

Here are the main findings from the analysis above

Analysis of Top 4 batsman

The analysis of the Top 4 test batsman Tendulkar, Kallis, Ponting and Sangakkara show the folliwing

1. Sangakkara has the highest average, followed by Tendulkar, Kallis and then Ponting.
2. Ponting has the highest strike rate followed by Tendulkar,Sangakkara and then Kallis
3. The predicted runs for a given Balls faced and Minutes at crease is highest for Ponting, followed by Tendulkar, Sangakkara and Kallis
4. The moving average for Tendulkar and Ponting shows a downward trend while Kallis and Sangakkara retired too soon
5. Tendulkar was out of form about the time of retirement while the rest were in-form. But this result has to be taken along with the moving average plot. Ponting was clearly on the way out.
6. The home and overseas performance indicate that Tendulkar is the clear leader. He has the highest number of matches played overseas and his performance has been consistent. He is followed by Ponting, Kallis and finally Sangakkara

Analysis of Top 3 legs spinners

The analysis of Anil Kumble, Shane Warne and M Muralitharan show the following

1. Muralitharan has the highest wickets and best economy rate followed by Warne and Kumble
2. Muralitharan has higher wickets frequency percentage between 3 to 8 wickets
3. Muralitharan has the best Economy Rate for wickets between 2 to 7

4. The moving average plot shows that the time was up for Kumble and Muralitharan but Warne had a few years ahead
5. The check for form status shows that Muralitharan and Kumble time was over while Warne still in great form
6. Kumble's has more matches abroad than the other 2, yet Kumble averages of 3 wickets at home and 2 wickets overseas liek Warne . Murali has played few matches but has an average of 4 wickets at home and 3 wickets overseas.

Final thoughts

Here are my final thoughts

Batting

Among the 4 batsman Tendulkar, Kallis, Ponting and Sangakkara the clear leader is Tendulkar for the following reasons

1. Tendulkar has the highest test centuries and runs of all time.Tendulkar's average is 2nd to Sangakkara, Tendulkar's predicted runs for a given Balls faced and Minutes at Crease is 2nd and is behind Ponting. Also Tendulkar's performance at home and overseas are consistent throughtout despite the fact that he has a highest number of overseas matches
2. Ponting takes the 2nd spot with the 2nd highest number of centuries, 1st in Strike Rate and 2nd in home and away performance.
3. The 3rd spot goes to Sangakkara, with the highest average, 3rd highest number of centuries, reasonable run frequency percentage in different run ranges. However he has a fewer number of matches overseas and his performance overseas is significantly lower than at home
4. Kallis has the 2nd highest number of centuries but his performance overseas and strike rate are behind others
5. Finally Kallis and Sangakkara had a few good years of batting still left in them (pity they retired!) while Tendulkar and Ponting's time was up

Bowling

Muralitharan leads the way followed closely by Warne and finally Kumble. The reasons are

1. Muralitharan has the highest number of test wickets with the best Wickets percentage and the best Economy Rate. Murali on average gas taken 4 wickets at home and 3 wickets overseas
2. Warne follows Murali in the highest wickets taken, however Warne has less matches overseas than Murali and average 3 wickets home and 2 wickets overseas
3. Kumble has the 3rd highest wickets, with 3 wickets on an average at home and 2 wickets overseas. However Kumble has played more matches overseas than the other two. In that respect his performance is great. Also Kumble has played less matches at home otherwise his numbers would have looked even better.
4. Also while Kumble and Muralitharan's career was on the decline , Warne was going great and had a couple of years ahead.

You can download this analysis at Introducing cricketr

Hope you have fun using the cricketr package as I had in developing it.

Also see my other posts in R

1. A peek into literacy in India: Statistical Learning with R
2. A crime map of India in R - Crimes against women
3. Analyzing cricket's batting legends - Through the mirage with R
4. Masters of Spin: Unraveling the web with R
5. Mirror, mirror . the best batsman of them all?