Coding Grace and Dublin R

A Taste of R

Kevin O'Brien

Overview

- Dice Rolls : `sample()` and `runif()` functions.
- The Monty Hall Problem :
- Gambler's Ruin : using `plot()` command.
- Monte Carlo and the Gambler's Fallacy.
- p-values.

1 Gambler's Ruin

Consider a gambler who starts with an initial fortune of 1 and then on each successive gamble either wins or loses independent of the past with probabilities p and q = 1-p respectively. Suppose the gambler has a starting kitty of A. This gamblers places bets with the Banker, who has an initial fortune B. We will look at the game from the perspective of the gambler only. The Banker is, by convention, the richer of the two.

- Probability of successful gamble for gambler : p
- Probability of unsuccessful gamble for gambler : q (where q = 1 - p)
- Ratio of success probability to failure success: s = p=q
- Conventionally the game is biased in favour of the Banker (i.e.$q > p$ and $s < 1$)

Let Rn denote the Gamblers total fortune after the $n-$th gamble. If the Gambler wins the first game, his wealth becomes Rn = A + 1. If he loses the first gamble, his wealth becomes Rn = A - 1. The entire sum of money at stake is the Jackpot i.e. A + B. The game ends when the Gambler wins the Jackpot (Rn = A + B) or loses everything (Rn = 0).

1.1 1.1 Simulation a Single Gamble

To simulate one single bet, compute a single random number between 0 and 1.

```
runif(1)
```

Lets assume that the game is biased in favour of the Banker p = 0.45 , q = 0.55. If the number is less than 0.45, the gamble wins. Otherwise the Banker wins.

```
> runif(1)
[1] 0.1251274
>#Gambler Loses
>
> runif(1)
[1] 0.754075
>#Gambler wins
>
> runif(1)
[1] 0.2132148
>#Gambler Loses
>
> runif(1)
[1] 0.8306269
```

Let A be the Gambler's Kitty at the start of the gambling. Let B be the Banker's Wealth. The probability of A winning a gamble is p. The vector Rn records the gambler's worth on an ongoing basis. At the start, The first value is A.

```
A=20;B=100;p=0.47
Rn=c(A)
probval = runif(1)
if (probval < p)
{
A = A+1; B =B-1
}else{A=A-1;B=B+1}
#Save the values from each bet
R=c(R,A)
```

Should the Gambler win the entire jackpot (A+B). The game would also cease. We include a break statement to stop the loop if the gambler wins the entire jackpot. A break statement will stop a loop if a certain logical condition is met.

```
A=20;B=100;p=0.47
Rec=c(A)
Total=A+B
while(A>0)
{
ProbVal=runif(1)
if(ProbVal <= p)
{
A = A+1; B =B-1
}else{A=A-1;B=B+1}
Rec=c(Rec,A)
if(A==Total){break}
}
```

We can construct a plot to depict the gambler's ongoing fortunes in the game.

```
length(Rec)
plot(Rec,type="l",col="red")
abline(h=0)
abline(v=0)
abline(h=A,col="red")
abline(h=Total,col="green")
```

2 Using the sample() command

```
sample(1:6,4)
sample(1:6,10)
sample(1:6,10,replace=TRUE)
```

```
> sample(1:6,4)
[1] 6 3 2 1
>
> sample(1:6,10)
Error in sample(1:6, 10) :
  cannot take a sample larger than the population when 'replace = FALSE'
>
> sample(1:6,10,replace=TRUE)
 [1] 6 1 2 6 5 5 5 1 3 4
```

3 The Monty Hall Problem

Imagine that the set of Monty Hall's game show Let's Make a Deal has three closed doors. Behind one of these doors is a car; behind the other two are goats. The contestant does not know where the car is, but Monty Hall does.

The contestant picks a door and Monty opens one of the remaining doors, one he knows doesn't hide the car. If the contestant has already chosen the correct door, Monty is equally likely to open either of the two remaining doors.

After Monty has shown a goat behind the door that he opens, the contestant is always given the option to switch doors. What is the probability of winning the car if she stays with her first choice? What if she decides to switch?

One way to think about this problem is to consider the sample space, which Monty alters by opening one of the doors that has a goat behind it. In doing so, he effectively removes one of the two losing doors from the sample space. We will assume that there is a winning door and that the two remaining doors, A and B, both have goats behind them. There are three options:

The contestant first chooses the door with the car behind it. She is then shown either door A or door B, which reveals a goat. If she changes her choice of doors, she loses. If she stays with her original choice, she wins. The contestant first chooses door A. She is then shown door B, which has a goat behind it. If she switches to the remaining door, she wins the car. Otherwise, she loses. The contestant first chooses door B. She is then is shown door A, which has a goat behind it. If she switches to the remaining door, she wins the car. Otherwise, she loses. Each of the above three options has a 1/3 probability of occurring, because the contestant is equally likely to begin by choosing any one of the three doors. In two of the above options, the contestant wins the car if she switches doors; in only one of the options does she win if she does not switch doors. When she switches, she wins the car twice (the number of favorable outcomes) out of three possible options (the sample space). Thus the probability of winning the car is 2/3 if she switches doors, which means that she should always switch doors - unless she wants to become a goatherd.

This result of 2/3 may seem counterintuitive to many of us because we may believe that the probability of winning the car should be 1/2 once Monty has shown that the car is not behind door A or door B. Many people reason that since there are two doors left, one of which must conceal the car, the probability of winning must be 1/2. This would mean that switching doors would not make a difference. As we've shown above through the three different options, however, this is not the case.

One way to convince yourself that 2/3 is the correct probability is to do a simulation with a friend. Have your friend impersonate Monty Hall and you be the contestant. Keep track of how often you win the car by switching doors and by not switching doors.

The objective of this report is to complete at least 20 of the 24 tests set out in the project outline. The ultimate goal is to have garnered a wider knowledge in both statistics and the tools used to implement them, in this case R & LaTeX. To achieve this, I will be using 3 different datasets to better represent the results, and to provide variety for the multiple different outcomes. They are:

1.

The (%) change in Real GDP growth throughout the world from 2008 until 2017 (predicted). This will be a CSV file read into the document.

2.

Using the MASS library, a study on the population of women who were at least 21 years old, of Pima Indian heritage and living near Phoenix, Arizona, who was tested for diabetes and their health measured.

3.

Again using the MASS library, a data frame with 17 observations on the boiling point of water in the Alps and the respective barometric pressure in inches of mercury.

I will be detailing these datasets throughout the next few pages. I have also added a notation beside every test (iii); clicking on it will return you to the Table of Contents, for quicker navigation. Now, to get started:

Relevant commands for dataset 1: (RealGDPgrowth.csv)

Assigning Columns:

TTE = RealGDP[,1], where *TTE*=2008;

TTN = RealGDP[,2], where *TTN*=2009;

TTT = RealGDP[,3], where *TTT*=2010;

TTEl = RealGDP[,3], where *TTEl*=2011;

TTTw = RealGDP[,4], where *TTTw*=2012;

TTTh = RealGDP[,5], where *TTTh*=2013;

TTF = RealGDP[,6], where *TTF*=2014;

TTFi = RealGDP[,7], where *TTFi*=2015;

TTS = RealGDP[,8], where *TTS*=2016;

TTSe = RealGDP[,9], where *TTSe*=2017;

Assigning Rows:

Wor = c(1.3, -2.3, 3.9, 2.6, 2.1, 2.3, 2.9, 2.9, 2.9, 3),
where *Wor*= World.

NAm = c(-0.2, -3, 2.5, 1.9, 2.2, 2.1, 2.4, 2.3, 2.3, 2.4),
where *NAm*= North America.

WE = c(0.1, -4.2, 2.3, 1.7, -0.1, 0.2, 1.1, 1.4, 1.4, 1.4),
where *WE*= Western Europe.

TE = c(4.5, -5.6, 3.4, 3.9, 2.6, 2.6, 3.3, 3.7, 3.8, 4.1),
where *TE*= Transition Economies.

AAJ = c(2.8, 0.7, 6.9, 3.6, 4, 4.1, 4.6, 4.3, 4.1, 4.1),
where *AAJ*= Australasia (including Japan.)

LA = c(3.9, -1.9, 5.9, 4.3, 3, 3.6, 4, 3.8, 3.9, 3.9),
where *LA*= Latin America.

MENAf = c(4.4, 1.3, 5.2, 2.5, 3.6, 3.3, 4.4, 4.7, 5, 5.1),
where *MENAf*= Middle East & North Africa.

SSA = c(4.8, 1.2, 4.5, 4.6, 4, 4.5, 5, 5, 5.4, 5.7),
where *SSA*= Sub-Saharan Africa.

Relevant commands for dataset 2: (Pima.tr)

npreg = Pima.tr[,1]; ***npreg***= number of pregnancies each tested woman has had.

glu = Pima.tr[,2]; ***glu***= plasma glucose concentration of each woman.

bp = Pima.tr[,3]; ***bp***= the blood pressure of the tested women (mm Hg).

skin = Pima.tr[,4]; ***skin***= each woman's triceps skin fold thickness (in mm).

bmi = Pima.tr[,5]; ***bmi***= body mass index, measured by $(\frac{weight(kg)}{height(m)})^2$.

ped = Pima.tr[,6]; ***ped***= diabetes pedigree function (probability of getting diabetes based on genetic history).

age = Pima.tr[,7]; ***age***= age of tested women.

type = Pima.tr[,8]; ***type***= presence of diabetes, as a yes or no answer.

Relevant commands for dataset 3: (forbes)

- ***forbes$bp***, the boiling point (°F)

- ***forbes$pres***, barometric pressure in inches of mercury.

Contents

#2. - #5.

iii #2. Histogram:

A histogram is a graphical procedure used to show the probability distribution of a given variable by grouping the frequencies of observations in a given range of values.

The Test:

Create a histogram. For the purpose of showing what a histogram exemplifies, I will be testing the real GDP Growth for the year 2008.

```
> hist(TTE, col=heat.colors(12), breaks=12,
+ xlim=c(-1,5), col.main="darkorange3", prob=FALSE,
+ xlab=’Annual Rate of Growth’, ylab=
+ ’Frequency of Annual Growth’, main=’Real
+ GDP Growth (2008) w/ Density Curve’,
+ col.lab="darkorange4", cex.lab=1.2)
> lines(density(TTE), lty=4, lwd=2)
```

The Conclusion:

We can interpret this figure as an indication of who was hit hardest by the sudden market crash. While the western world languishes on the left (North America & and Western Europe, respectively), the developing countries are almost unaffected by the beginning of the recession (on the right).

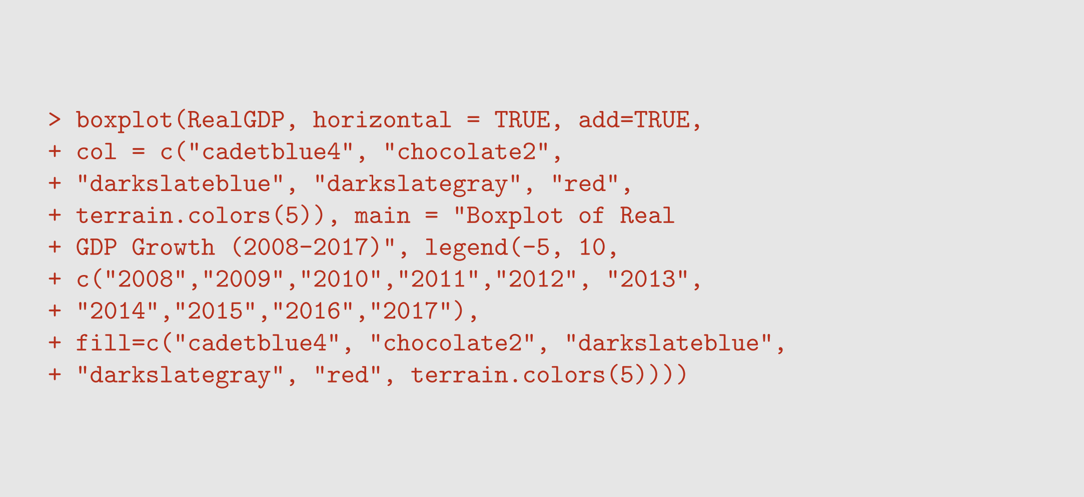
iii #3. Boxplot:

A boxplot is a graph designed to show the distribution of a certain set of data. It is a convenient way of graphically depicting groups of numerical data through their five-number summaries: the IQR, & the sample minimums and maximums. Many boxplots (including this one) use the formula $\pm 1.58 \times (\frac{IQR}{\sqrt{n}})$ as a limit for their samples min.'s & max.'s; the rest are regarded as outliers.

The Test:

Create a boxplot. As with #2., I will be creating a sample test, however this time I will use the boxplot to highlight the difference of overall worldwide real GDP growth from 2008 to 2017.

```
f code
```



```
> boxplot(RealGDP, horizontal = TRUE, add=TRUE,
+ col = c("cadetblue4", "chocolate2",
+ "darkslateblue", "darkslategray", "red",
+ terrain.colors(5)), main = "Boxplot of Real
+ GDP Growth (2008-2017)", legend(-5, 10,
+ c("2008","2009","2010","2011","2012", "2013",
+ "2014","2015","2016","2017"),
+ fill=c("cadetblue4", "chocolate2", "darkslateblue",
+ "darkslategray", "red", terrain.colors(5))))
```