Statistical Theory, Conditional Probability

Vaibhav Walvekar

```
# Load some helpful libraries
library(tidyverse)
```

If a baseball team scores X runs, what is the probability it will win the game?

Baseball is a played between two teams who take turns batting and fielding. A run is scored when a player advances around the bases and returns to home plate. More information about the dataset can be found at http://www.retrosheet.org/.

```
#Reading column names for the data present in GL2010, GL2010, GL2011, GL2012, GL2013
colNames <- read.csv("cnames.txt", header=TRUE)
#Creating a null df
baseballData <- NULL
#Looping to read dat from all the GL .txt files in sequence from 2010 to 2013
for (year in seq(2010,2013,by=1)){
  #Constructing the name of the file by using year dynimically to read from working directory
  mypath <- paste('GL',year,'.TXT',sep='')
  # cat(mypath,'\n')
  #Reading each GL .txt filesand binding them row by row and
  #using Name column from colNames df for header names
  baseballData <- rbind(baseballData,read.csv(mypath,
  col.names=colNames$Name))
  #Converting to a table df
  baseballData <- tbl_df(baseballData)
}
# baseballData
```

Selecting relevant columns and to create a new data frame to store the data for analysis.

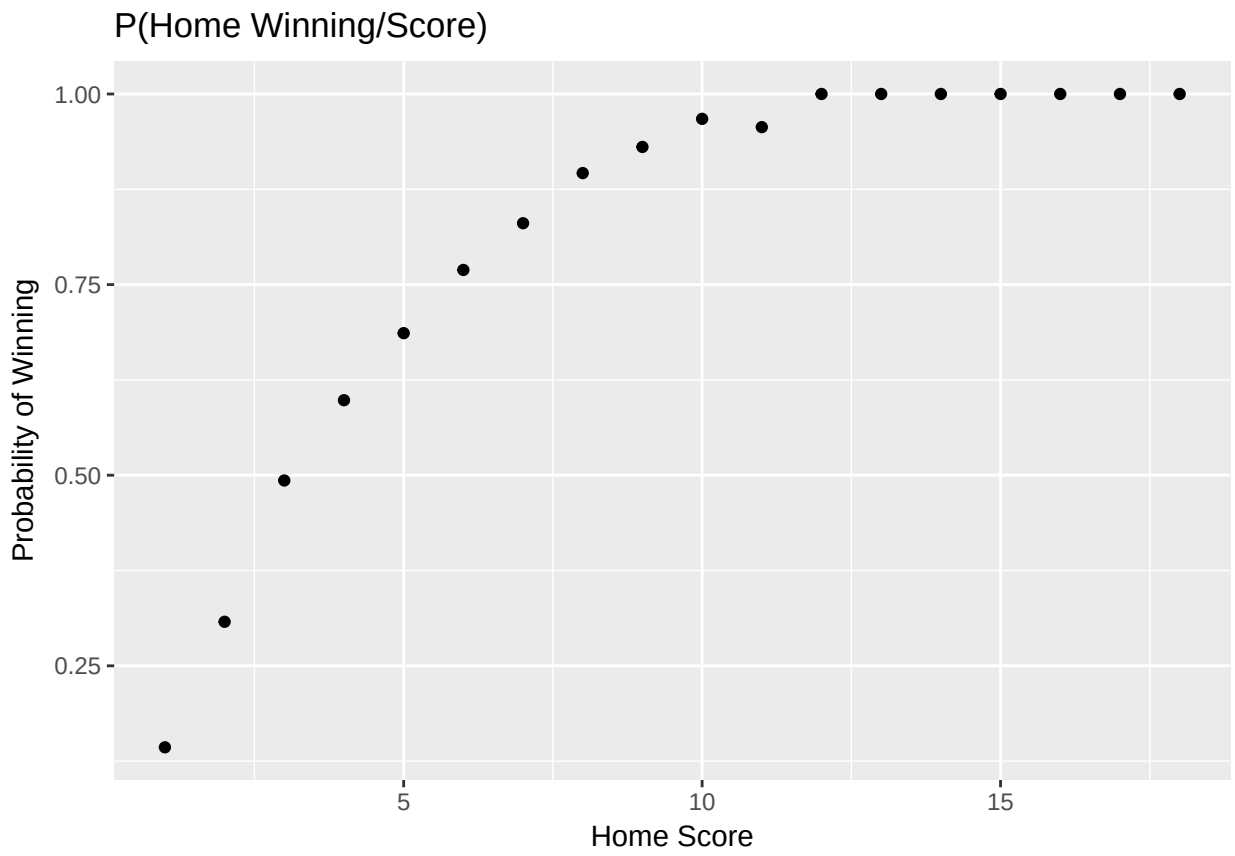
- Date
- Home
- Visitor
- HomeLeague
- VisitorLeague
- HomeScore
- VisitorScore

```
baseballData_final <- baseballData[,c('Date','Home','Visitor','HomeLeague','VisitorLeague','HomeScore',
```

Considering only games between two teams in the National League, computing the conditional probability of the team winning given X runs scored, for $X = 0, ..., 10$. Doing this separately for Home and Visitor teams.

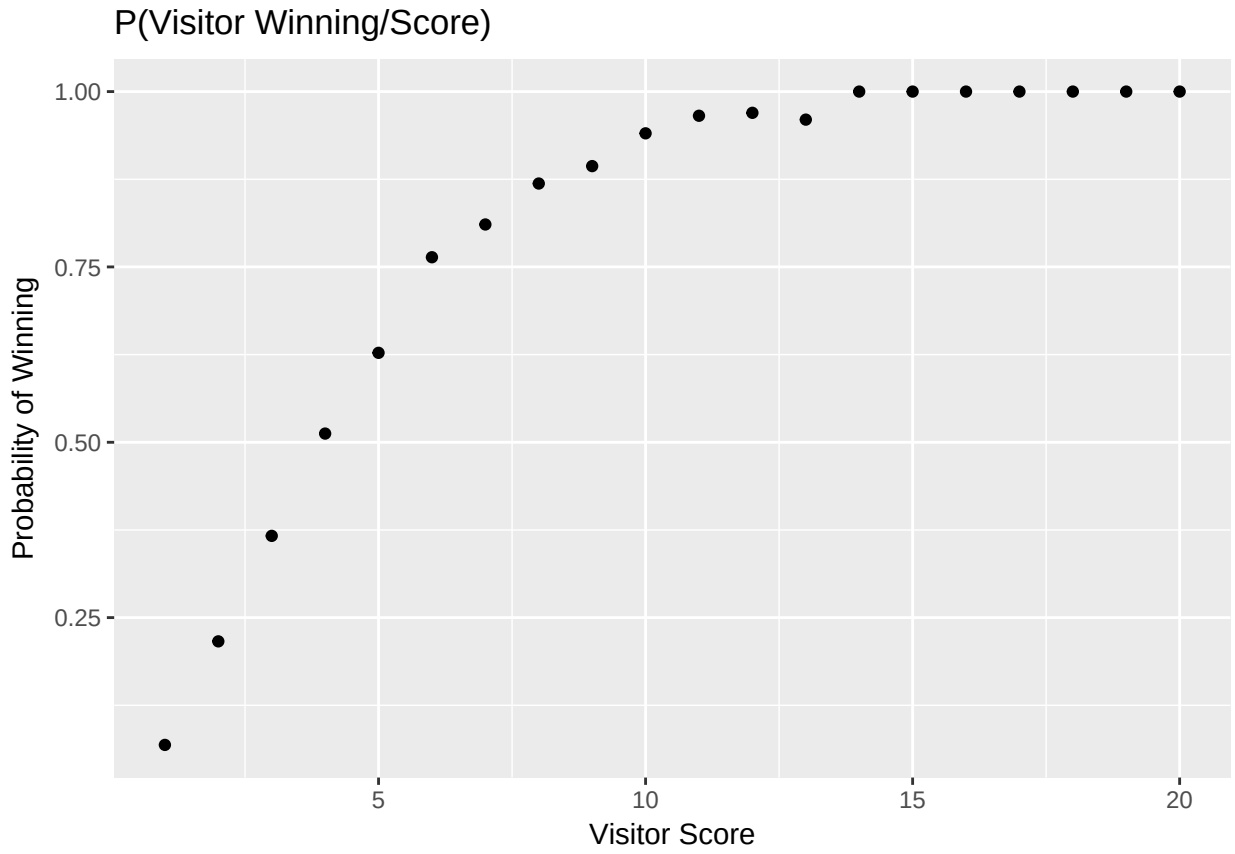
- Design a visualization that shows your results.
- Discuss what you find.

```r
baseballData_filtered <- filter(baseballData_final, baseballData_final$VisitorLeague == "NL", baseballDa
#Considering home win
baseballData_filtered$ResultforHome <- ifelse(baseballData_filtered$HomeScore>baseballData_filtered$Visi
# Calculating total matches for each score by Home team
by_HomeScore = group_by(baseballData_filtered,HomeScore)
sum_hs = summarize(by_HomeScore, totalcount = n())

# Calculating total matches for each score by Home team when they won
by_HomeScore_ResultforHome = group_by(baseballData_filtered,HomeScore,ResultforHome)
sum_hs_rsh = filter(summarize(by_HomeScore_ResultforHome, count = n()),ResultforHome == "Won")

# Merging the above two df, to calculate probabilty
final_hs_rsh = merge(sum_hs_rsh, sum_hs, by= "HomeScore")
# Adding neew column  for probability
final_hs_rsh <- mutate(final_hs_rsh,Probability = count/totalcount)
#Plotting probabilty of win for each score
p3 <- ggplot(final_hs_rsh, aes(x = HomeScore, y = Probability))
p3 + geom_point() +labs(x = "Home Score",
                                    y="Probability of Winning",
                                    title = "P(Home Winning/Score)")
```




```r
#Considering visitor win
baseballData_filtered$ResultforVisitor <- ifelse(baseballData_filtered$HomeScore<baseballData_filtered$V

# Calculating total matches for each score by Visitor team
by_VisitorScore = group_by(baseballData_filtered,VisitorScore)
```

```
sum_vs = summarize(by_VisitorScore, totalcount = n())

# Calculating total matches for each score by Visitor team when they won
by_VisitorScore_ResultforVisitor = group_by(baseballData_filtered,VisitorScore,ResultforVisitor)
sum_vs_rsv = filter(summarize(by_VisitorScore_ResultforVisitor, count = n()),ResultforVisitor == "Won")

# Merging the above two df, to calculate probabilty
final_vs_rsv = merge(sum_vs_rsv, sum_vs, by= "VisitorScore")
# Adding new column  for probability
final_vs_rsv <- mutate(final_vs_rsv,Probability = count/totalcount)
#Plotting probabilty of win for each score
p4 <- ggplot(final_vs_rsv, aes(x = VisitorScore, y = Probability))
p4 + geom_point() +labs(x = "Visitor Score",
                                    y="Probability of Winning",
                                    title = "P(Visitor Winning/Score)")
```

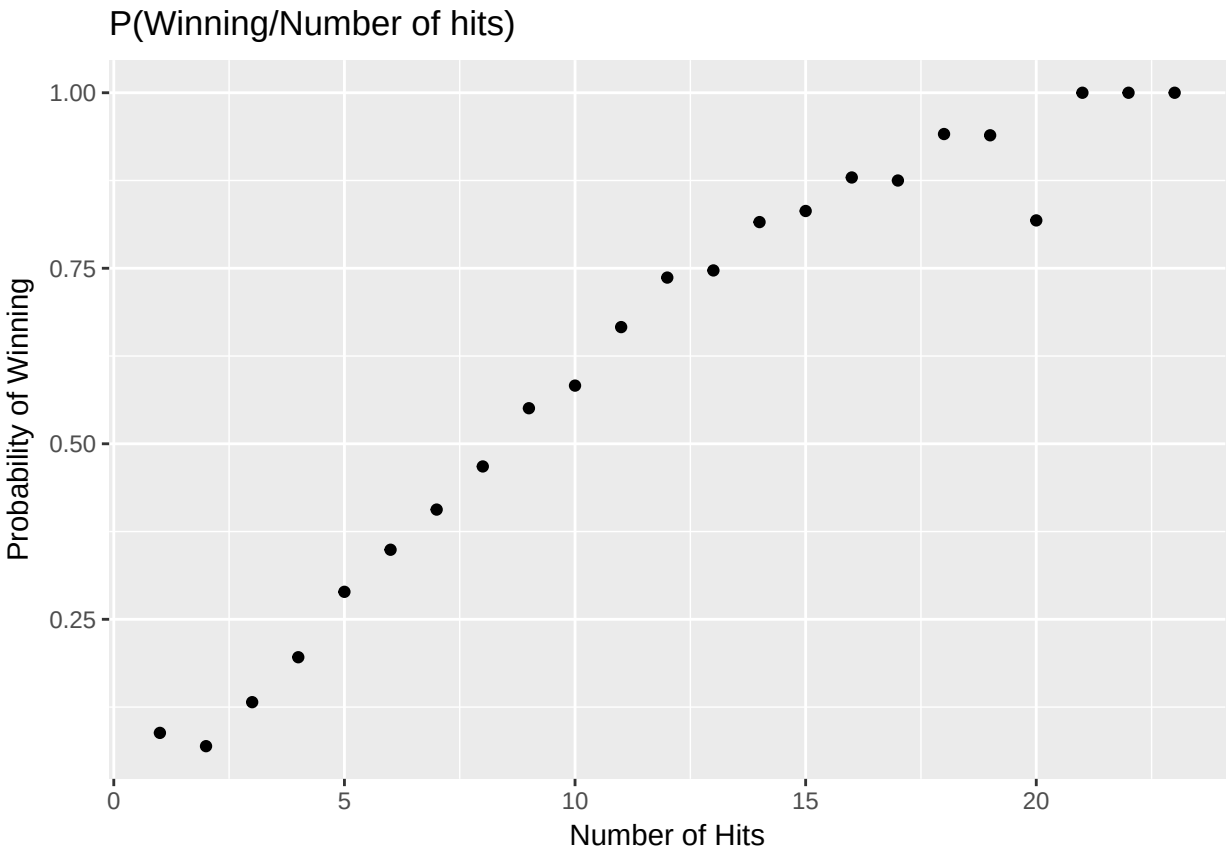


From both the above graphs for Home team or the Visitor we observe that once the team scores above 14 runs, they have a probability of 1 for winning the match. A score of above 6 runs, gives almost a 75% chance of winning the match.

Repeating the above problem, but now considering the probability of winning given the number of hits.

```
#Including Hhits and Vhits in df
baseballData_final_hits <- baseballData[,c('Date','Home','Visitor','HomeLeague','VisitorLeague','HomeSc
```

```r
#Filtering only NL games
baseballData_filtered_hits <- filter(baseballData_final_hits, baseballData_final$VisitorLeague == "NL",
#Adding a new column, which consists of the number of hits scored by the team winning the match
baseballData_filtered_hits$HitsInWinningCause <- ifelse(baseballData_filtered_hits$HomeScore>baseballDa
                                                        , baseballData_filtered_hits$Hhits, baseballData_fil

# Calculating total matches for each hit by Home team
by_Hhits = group_by(baseballData_filtered_hits,Hhits)
sum_Hhits = summarize(by_Hhits, Hhitscount = n())

# Calculating total matches for each hit by Visitor team
by_Vhits = group_by(baseballData_filtered_hits,Vhits)
sum_Vhits = summarize(by_Vhits, Vhitscount = n())

# Merging the above two df, to get total count of matches in which each number of hit were scored
Total_Hhits_Vhits = merge(sum_Hhits, sum_Vhits, by.x= "Hhits",by.y = "Vhits")

# Calculating matches for each hit for winning cause
by_HitsInWinningCause = group_by(baseballData_filtered_hits,HitsInWinningCause)
sum_HitsInWinningCause = summarize(by_HitsInWinningCause, hitscount_winning = n())

# Merging the above two df, to calculate probability
Total_hits_winning_cause = merge(sum_HitsInWinningCause, Total_Hhits_Vhits, by.x= "HitsInWinningCause",

# Adding new column  for probability
Total_hits_winning_cause <- mutate(Total_hits_winning_cause,Probability = hitscount_winning/(Hhitscount
#Plotting probabilty of win for each score
p5 <- ggplot(Total_hits_winning_cause, aes(x = HitsInWinningCause, y = Probability))
p5 + geom_point() +labs(x = "Number of Hits",
                                    y="Probability of Winning",
                                    title = "P(Winning/Number of hits)")
```



From the above graphic we can see that as the number of hits increase for a team, the probability of that team winning the match goes up. One abnormal observation is that when number of hits is equal to 20, the probability of team winning goes down a bit considering lower values (15-19) of number of hits.

Triathlon Times

In triathlons, it is common for racers to be placed into age and gender groups. Friends Leo and Mary both completed the Hermosa Beach Triathlon, where Leo competed in the Men, Ages 30 - 34 group while Mary competed in the Women, Ages 25 - 29 group. Leo completed the race in 1:22:28 (4948 seconds), while Mary completed the race in 1:31:53 (5513 seconds). Obviously Leo finished faster, but they are curious about how they did within their respective groups.

Here is some information on the performance of their groups:

- The finishing times of the Men, Ages 30 - 34 group has a mean of 4313 seconds with a standard deviation of 583 seconds.
- The finishing times of the Women, Ages 25 - 29 group has a mean of 5261 seconds with a standard deviation of 807 seconds.
- The distributions of finishing times for both groups are approximately Normal.

Remember: a better performance corresponds to a faster finish.

A short-hand description for these two normal distributions.

We are considering distribution of two categories of triathlons run at the Hermosa Beach Triathlon. One distribution is of the timings taken by all participants in the Mens - age (30 - 34) category. This distribution

has a mean of 4313 seconds and standard deviation of 583 seconds. Second distribution is of the timings taken by all participants in the Womens - age (25 - 29) category. This distribution has a mean of 4948 seconds and standard deviation of 807 seconds. The timings of the completion of race of the participants is the random varibale in both these distributions. Both distbutions are given to be approximately normal, which means that the scores (timings) are symettrical across both sides of the means.
z = X-mu/(sigma) zScoreLeo = (4948-4313)/583
zScoreMary = (5513-5261)/807
Normal distribution for Leo - n1(mean = 4313, sd = 583)
Normal distribution for Mary - n2(mean = 5261, sd = 807)

Calculating Z Score

```
Leo_time <- 4948
Mary_time <- 5513
Men_mean_time <- 4313
Women_mean_time <- 5261
Men_sd <- 583
Women_sd <- 807

zScoreLeo <- (4948-4313)/583
ZScoreMary <- (5513-5261)/807

print(zScoreLeo)
```

```
## [1] 1.089194
```

```
print(ZScoreMary)
```

```
## [1] 0.3122677
```

The Zscore for Leo and Mary are 1.089 and 0.3122 respectively. From the ZScore we can say that, Leo has a completion timing which is 1.089 standard deviations away from the mean of that distribution and Mary has a completion timing which is 0.3122 standard deviations away from the mean of that distribution. In whole, Zscore helps us compare scores from two different distributions, because when we compute zscores, we transform the distribution to a standard normal distribution having mean = 0 and standard deviation = 1. The sign (+ or -) indicates the direction of score from the mean.

Comparing Ranks

As discussed above, by calculating the zscores, we can now make direct comparisons betweent the scores of Leo and Mary in terms of zscores. As Leo is 1.089 standard deviations away from the mean and Mary is only 0.3122 standard deviations from the mean, Leo would have more participants finishing before him than Mary in terms of percentile in respective groups. Thus we can say that Mary ranked better than Leo in their respective groups.

Percentile ranks

```
(1 - pnorm(zScoreLeo))*100
```

```
## [1] 13.80342
```

Leo finished faster than 13.803% of triatheletes in his group. The pnorm function gives the percentile of participants finishing faster than Leo, thus had to subtract the result from 1.

Percentile ranks

```
(1 -pnorm(ZScoreMary))*100
```

```
## [1] 37.74186
```

Mary finished faster than 37.741% of triatheletes in her group. The pnorm function gives the percentile of participants finishing faster than Mary, thus had to subtract the result from 1.

What if distribution is not normal

We have answers to the above question based on the distributions being normal, which means, the scores in both observations were symettrical across the means. The Zscore computed was with the assumption that subtracting mean from the actual score would standard normalize the distrbution with mean as 0 and sd as 1. If the above conditions arent satisfied then we cannot make direct comparisons about scores from two different distributions and thus the answers would change.

Sampling with and without Replacement

In the following situations assume that half of the specified population is male and the other half is female.

From a room of 10 people, compare sampling 2 females with and without replcement.

Sampling with Replacement: To sample out two females in a row, we need to find the probability of choosing one female in and then again choosing one more female with replacement of the first female in the room. As there 10 people and 50% are female, there are a total of 5 females in the room. Thus P(choosing a female) = number of favourable choices/total choices = 5/10 = 1/2, similarly in the second chance, since we are replacing the first selected female back in the rooom, again the P(choosing women second time) = 5/10 = 1/2. Thus the P(sampling two females in a row) = P(choosing a female) * P(choosing women second time) = (1/2) * (1/2) = 1/4 = 0.25.

Sampling without Replacement: Here the P(choosing a female) will remain the same 1/2, but since the chosen women is not replaced in the room, the P(choosing women second time) = number of favourable choices/total choices = 4/9. Therefore, P(sampling two females in a row) = P(choosing a female) * P(choosing women second time) = (1/2) * (4/9) = 2/9 = 0.222.

From a room of 10000 people, compare sampling 2 females with and without replcement.

Sampling with Replacement: To sample out two females in a row, we need to find the probability of choosing one female in and then again choosing one more female with replacement of the first female in the room. As there 10000 people and 50% are female, there are a total of 5000 females in the room. Thus P(choosing a female) = number of favourable choices/total choices = 5000/10000 = 1/2, similarly in the second chance, since we are replacing the first selected female back in the rooom, again the P(choosing women second time) = 5000/10000 = 1/2. Thus the P(sampling two females in a row) = P(choosing a female) * P(choosing women second time) = (1/2) * (1/2) = 1/4 = 0.25.

Sampling without Replacement: Here the P(choosing a female) will remain the same 1/2, but since the chosen women is not replaced in the room, the P(choosing women second time) = number of favourable

choices/total choices = 4999/9999. Therefore, P(sampling two females in a row) = P(choosing a female) * P(choosing women second time) = (1/2) * (4999/9999) = 2222/9999 = 0.249.

Often smaples from large population are considered as independent, explain whether or not this assumption is reasonable.

From above, we can observe that as the population has increased considerably, the probabilty with and without replacement is almost same (0.25 ~ 0.249). Thus as the population size increases, the probability of with replacement is not changed but it affects the probability of without replacement. This is the case because a sample chosen at random in first attempt is highly unlikely to be selected in the second attempt when replaced in the room with population being very large. When the population is small, the sample chosen in second attempt depends very much on what was chosen in first attempt, which is not the case for large paopulations. Thus it is reasonable to assume that a sample chosen from a large population are independent.

Sample Means

You are given the following hypotheses: $H_0 : \mu = 34$, $H_A : \mu > 34$. We know that the sample standard deviation is 10 and the sample size is 65. For what sample mean would the p-value be equal to 0.05? Assume that all conditions necessary for inference are satisfied.

As p value has to be equal to 0.05, that means the sample mean is at the extreme value and any more increase in mean would reject the H_0. Thus for retaining H_0 and having the sample mean at the extreme value in a one tailed distribution, we have to calculate the tcrtitical or zcritcal value. The percentile at the extreme value of the distribution which will retain null hypthosis is 95% as p value is given to be 0.05. Hence, using it we calculate the tcritical or zcritcal. Following which we can calculate sample mean by using this formula : tcritical = (sample_mean - mu)/sample_sd_means sample_sd_means = sample_sd/(n)^0.5

```
n = 65
mu = 34
sample_sd = 10
sample_size = 65
#zcritcal <- qnorm(0.95)
# Degree of freedom = n-1
tcritcal <-  qt(.95, 64)
sample_mean = tcritcal*sample_sd/(n)^0.5 +mu
print(sample_mean)
```

```
## [1] 36.07016
```

Using t statitic as we are unaware of the population variance to use Z statistic. We have to use t test when we only know sample variance. As the sample size is greater than 30, the value of t statistic and z statistic will be almost same. Thus if the sample mean is 36.07016, then we would have the p value as 0.05.