# Lab 6

Apurva Shah 705595011

2022-05-15

## Contents



```
## Date last run: 2022-05-15

## Hello World!
```

Note that included data sets were made by processing data obtained from MLB and the NHL.

## Examples

### Binomial Model

Imagine a baseball team, call them the Chattanooga P-Values. This upcoming season, this imaginary team will play 40 home games, and, for each home game, will have the same probability of winning, 70%.

The binomial distribution can be used here to model the number of season home game wins.

```
xdomain <- I(0:40)

hg_win_prop <- dbinom(xdomain, size=40, prob=0.70)
hg_win_prop
```

```
##  [1] 1.215767e-21 1.134715e-19 5.162955e-18 1.525940e-16 3.293487e-15 5.533059e-14
##  [7] 7.531108e-13 8.535256e-12 8.215184e-11 6.815560e-10 4.929921e-09 3.137223e-08
## [13] 1.769045e-07 8.890585e-07 4.000763e-06 1.618087e-05 5.899274e-05 1.943290e-04
## [19] 5.793884e-04 1.565365e-03 3.835144e-03 8.522543e-03 1.717422e-02 3.136161e-02
## [25] 5.183378e-02 7.740510e-02 1.041992e-01 1.260681e-01 1.365738e-01 1.318644e-01
## [31] 1.128173e-01 8.491625e-02 5.572629e-02 3.152194e-02 1.514289e-02 6.057157e-03
## [37] 1.962968e-03 4.951630e-04 9.121424e-05 1.091452e-05 6.366806e-07
```

```r
par(mfrow=c(1,1), lend=1, cex=0.65)
plot(xdomain, hg_win_prop, type="h", lwd=3,
     xlab="Number of Home Game Wins", ylab="Probability")
```

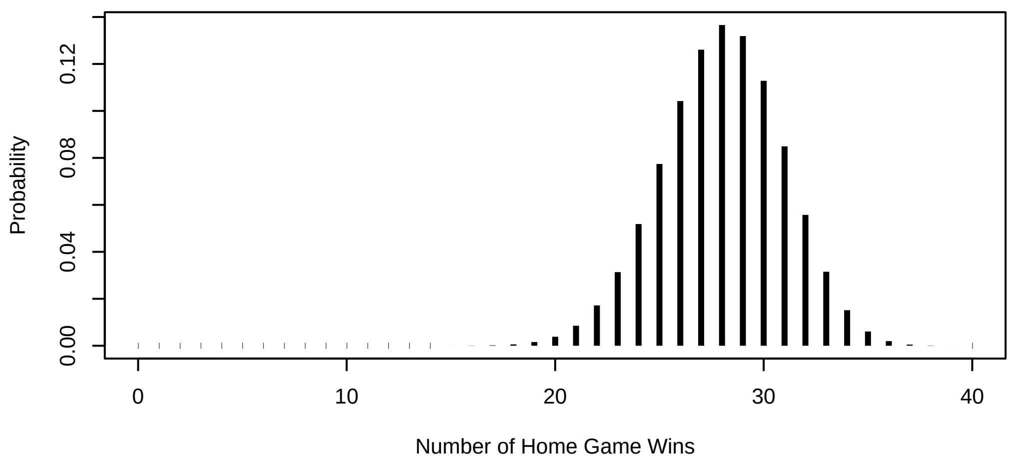



Figure 1: Distribution of Home Game Wins.

The expectation by definition of binomial PMF is $n \cdot p = 28$

Using the general definition for the expectation of a PMF, $\sum_i X_i \cdot \Pr[X_i] = 28$.

Same answer.

What's the probability team will win 30 or more home games?

```r
sum( dbinom(I(30:40), size=40, prob=0.70) )
```

```
## [1] 0.3087427
```

Using the cumulative R function:

```r
1 - pbinom(29, size=40, prob=0.70)
```

```
## [1] 0.3087427
```

What's the probability team will lose half or more of their home games?

```r
sum( dbinom(I(0:20), size=40, prob=0.70) )
```

```
## [1] 0.006254504
```

Using the cumulative R function

```r
pbinom(20, size=40, prob=0.70)
```

```
## [1] 0.006254504
```

## Normal Model

The normal, or Gaussian probability distribution is a PDF — its domain is over the continuum of the real numbers.

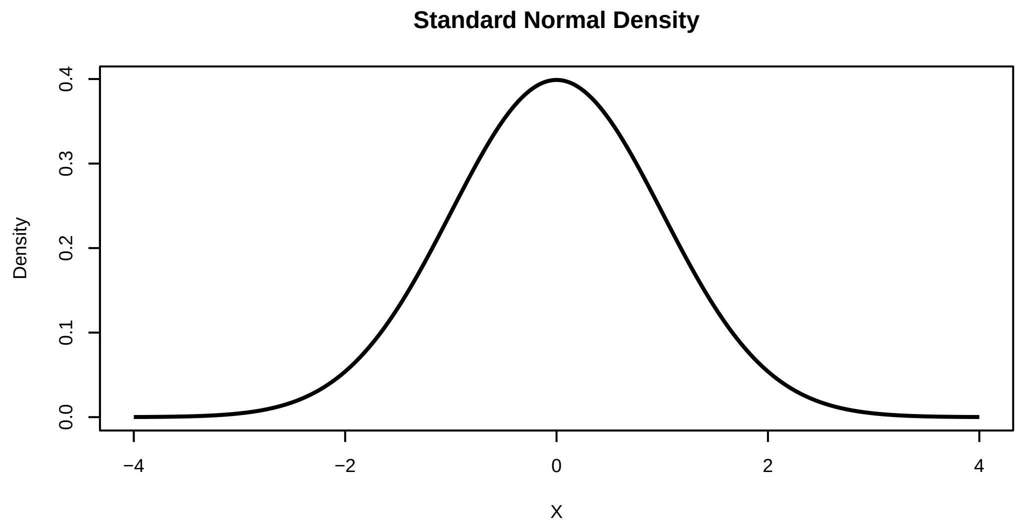
A normal distribution is uniquely defined by two parameters, the mean (the expectation) and the standard deviation (or the variance).

We'll use a path to show density.

```r
xdom <- seq(-4, 4, length=500)

xdensity <- dnorm(xdom, 0, 1)

par(mfrow=c(1,1), lend=1, cex=0.60)
plot(xdom, xdensity, type="l", lwd=2,
     xlab="X", ylab="Density", main="Standard Normal Density")
```




## The Normal Approximation to The Binomial Model

The normal model is rather unique as it is the limiting distribution of many estimators, along with other distributions.

The normal model can be used to model the binomial model.

Let's illustrate an example.

If X is normally distributed, the probability that X will be one or more standard deviation greater than the mean is

```
1 - pnorm(1, 0, 1)
```

```
## [1] 0.1586553
```

For increasing binomial sample size (i.e., number of trials), were going to calculate the probability of each respective binomial random variable being more than one standard deviation from the mean.

```
p_success <- 0.5

xtrialsTry <- seq(5, 2000, by=5)

pout_vec <- numeric(length(xtrialsTry))

for(i in 1:length(xtrialsTry)) {
  xthis_numTrials <- xtrialsTry[ i ]
  xthis_mean <- p_success * xthis_numTrials
  xthis_sd <- sqrt( (1 - p_success) * p_success * xthis_numTrials )
  xdom <- I(0:xthis_numTrials)

  xdom_prob <- xdom[ xdom > (xthis_mean + 1 * xthis_sd) ]
  pout_vec[ i ] <- sum(dbinom(xdom_prob, size=xthis_numTrials, prob=p_success))

}
```

```r
par(mfrow=c(1,1), lend=1, cex=0.60)
plot(xtrialsTry, pout_vec, type="l", lwd=1,
     xlab="Sample size", ylab="Probability X > mu + 1*sd",
     main="Normal Approximation of Binomial Model, Example")

abline(h=1 - pnorm(1, 0, 1), lwd=2, col="#22BB5577")
```

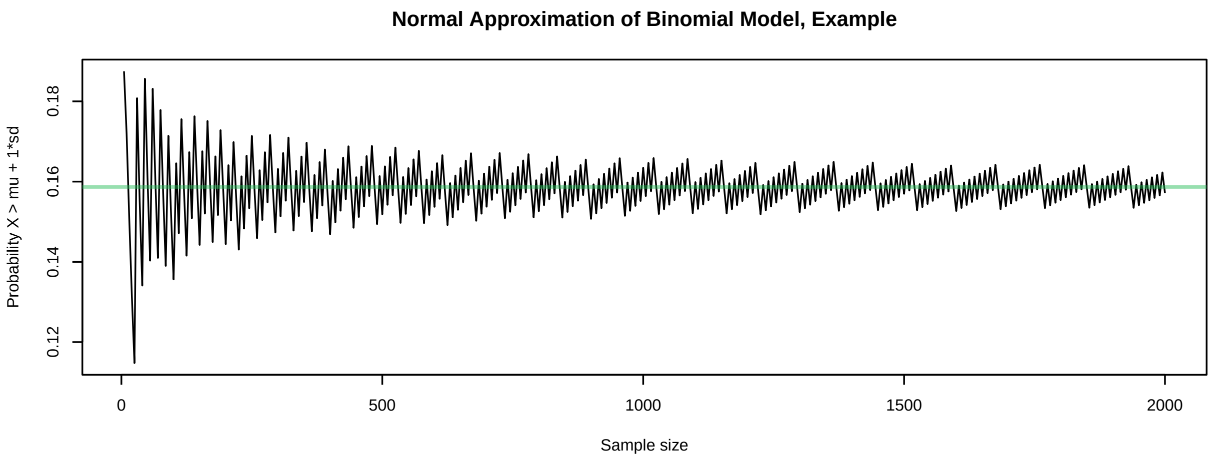



Figure 2: Binomial probability that number of successes will be greater than 1 standard deviation above the mean for increasing number of binomial trials. Grean line shows probability under normal distribution

To make the convergence more pronounced:

```r
p_success <- 0.5

xtrialsTry <- 1 * 2^(I(2:15))

pout_vec <- numeric(length(xtrialsTry))

for(i in 1:length(xtrialsTry)) {
  xthis_numTrials <- xtrialsTry[ i ]
  xthis_mean <- p_success * xthis_numTrials
  xthis_sd <- sqrt( (1 - p_success) * p_success * xthis_numTrials )
  xdom <- I(0:xthis_numTrials)

  xdom_prob <- xdom[ xdom > (xthis_mean + 1 * xthis_sd) ]
  pout_vec[ i ] <- sum(dbinom(xdom_prob, size=xthis_numTrials, prob=p_success))

}
```

```r
par(mfrow=c(1,1), lend=1, cex=0.60)
plot(xtrialsTry, pout_vec, type="l", lwd=1,
     xlab="Sample size", ylab="Probability X > mu + 1*sd",
```

```r
    main="Normal Approximation of Binomial Model, Example")

abline(h=1 - pnorm(1, 0, 1), lwd=2, col="#22BB5577")
```

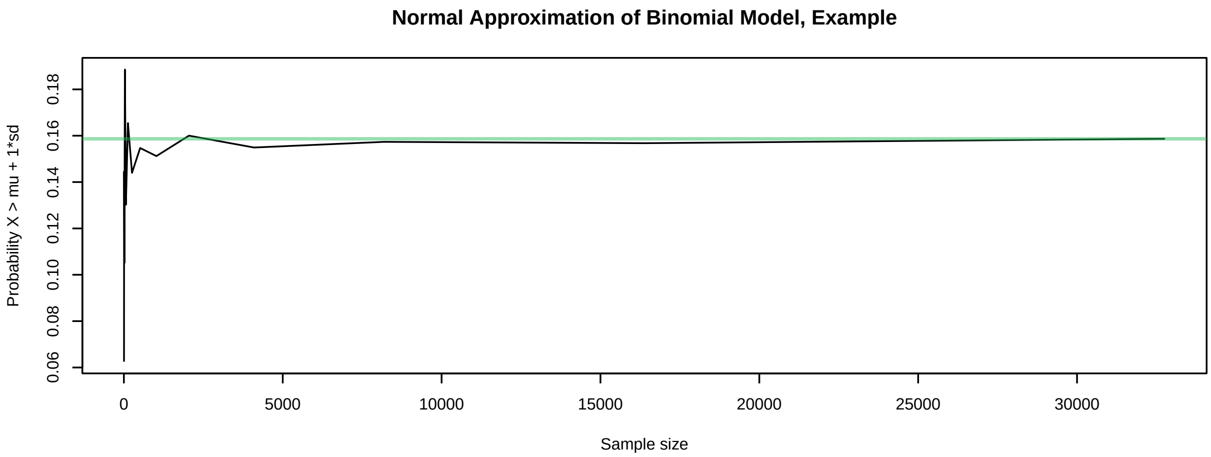



Figure 3: Binomial probability that number of successes will be greater than 1 standard deviation above the mean for increasing number of binomial trials. Grean line shows probability under normal distribution

## MLB 2021 Season — Are Some Teams Actually Better than Others?

Suppose a friend says they've been to many MLB games, and they believe that there's no difference between the teams, the outcome of the game is pure chance, and that the probability the home team will win is always 50%.

The experiment that follows uses empiric probabilities, and requires some creative thinking.

```r
## Read in our data
xdf <- read.csv("MLB_team_2021.csv", header=TRUE)
head(xdf, n=6)
```

```
##       date gameID                  team VorH bat_runs bat_homeRuns bat_strikeOuts
## 1 20210401 634615   Los Angeles Dodgers    V        5            0              6
## 2 20210401 634615      Colorado Rockies    H        8            0              4
## 3 20210401 634618  Arizona Diamondbacks    V        7            4             12
## 4 20210401 634618      San Diego Padres    H        8            2             10
## 5 20210401 634622        Atlanta Braves    V        2            1             10
## 6 20210401 634622 Philadelphia Phillies    H        3            0             13
##   bat_baseOnBalls pitch_runs pitch_homeRuns pitch_strikeOuts pitch_baseOnBalls
## 1               8          8              0                4                 3
## 2               3          5              0                6                 8
## 3               1          8              2               10                 5
## 4               5          7              4               12                 1
## 5               2          3              0               13                 4
## 6               4          2              1               10                 2
```

Let's look at the distribution of total home game wins for each of the thirty MLB teams.

```r
WorL <- xdf[ , "bat_runs"] > xdf[ , "pitch_runs"]

xdf_HT <- xdf[ xdf[ , "VorH"] == "H", ]
dim(xdf_HT)
```

```
## [1] 2429   12
```

```r
xWinTH <- WorL[ xdf[ , "VorH"] == "H" ]
xWinTH
```

```
##    [1]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE
##   [15]  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
##   [29] FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
##   [43] FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE
##   [57] FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE
##   [71] FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE
##   [85] FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE
##   [99]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE
##  [113]  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE
##  [127] FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE
```

```
##  [141]  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
##  [155]  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE
##  [169]  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE
##  [183] FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE
##  [197] FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE
##  [211] FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE
##  [225] FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE
##  [239] FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE
##  [253]  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
##  [267]  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE
##  [281] FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE
##  [295]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE
##  [309] FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE
##  [323] FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
##  [337] FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
##  [351] FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
##  [365]  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE
##  [379]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
##  [393] FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE
##  [407]  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE
##  [421]  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE
##  [435] FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE
##  [449] FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE
##  [463] FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE
##  [477]  TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##  [491] FALSE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE
##  [505]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE
##  [519]  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE
##  [533]  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE
##  [547]  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
##  [561] FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
##  [575] FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE
##  [589]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE
##  [603] FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE
##  [617]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE
##  [631]  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE
##  [645] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
##  [659]  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE
##  [673] FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##  [687] FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
##  [701]  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE
##  [715] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
##  [729] FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
##  [743] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
##  [757] FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
##  [771]  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
##  [785]  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE FALSE
##  [799]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
##  [813]  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
##  [827]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE
##  [841] FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
##  [855]  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
##  [869] FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE
##  [883] FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE
##  [897]  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
##  [911]  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE  TRUE
##  [925]  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE
##  [939] FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE
##  [953]  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
```

```
##  [967] FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE
##  [981] FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##  [995]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE
## [1009]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE
## [1023] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE
## [1037]  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
## [1051]  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE
## [1065] FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE
## [1079] FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE
## [1093]  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE
## [1107]  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE
## [1121] FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
## [1135]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
## [1149]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE
## [1163]  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
## [1177]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE
## [1191]  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE
## [1205]  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE
## [1219] FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE
## [1233] FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE
## [1247]  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
## [1261]  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE
## [1275]  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE
## [1289] FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE
## [1303]  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE
## [1317]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE
## [1331]  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
## [1345]  TRUE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE
## [1359] FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE
## [1373] FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE
## [1387]  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE
## [1401] FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE
## [1415]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
## [1429] FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE
## [1443]  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE
## [1457]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE
## [1471]  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE
## [1485]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## [1499] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE
## [1513] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE
## [1527]  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE
## [1541]  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE
## [1555]  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
## [1569]  TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE FALSE
## [1583]  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE
## [1597] FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE
## [1611]  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE
## [1625] FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE
## [1639]  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
## [1653] FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
## [1667] FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE
## [1681]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE
## [1695]  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE FALSE
## [1709] FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE
## [1723] FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE
## [1737] FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE
## [1751] FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
## [1765] FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE
## [1779] FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

```
## [1793] FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
## [1807]  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE
## [1821]  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE
## [1835]  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE
## [1849]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE
## [1863] FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE
## [1877] FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE
## [1891] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
## [1905] FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE
## [1919] FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
## [1933]  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE
## [1947] FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## [1961] FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE
## [1975] FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE
## [1989] FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE
## [2003]  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE
## [2017]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
## [2031]  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE
## [2045] FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE
## [2059]  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE
## [2073] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE
## [2087]  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE
## [2101] FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE
## [2115]  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE
## [2129]  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE
## [2143] FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE
## [2157]  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE
## [2171]  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE
## [2185]  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE
## [2199] FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE
## [2213]  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE
## [2227] FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE
## [2241] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE
## [2255] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE
## [2269]  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE
## [2283] FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE
## [2297] FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE
## [2311]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE
## [2325]  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
## [2339] FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## [2353]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
## [2367] FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE
## [2381]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
## [2395]  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE
## [2409]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE
## [2423]  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE
```

```r
xagg <- aggregate(xWinTH, by=list(xdf_HT[ , "team"]), sum)

xnumberHGwins <- xagg$x

xbrks <- seq(21.5, 65.5, by=4)
xbrks
```

```
##  [1] 21.5 25.5 29.5 33.5 37.5 41.5 45.5 49.5 53.5 57.5 61.5 65.5
```

```r
par(cex=0.65)
hist(xnumberHGwins, breaks=xbrks, main="Total Home Game Wins for Each Team over MLB 2021 Season")
```

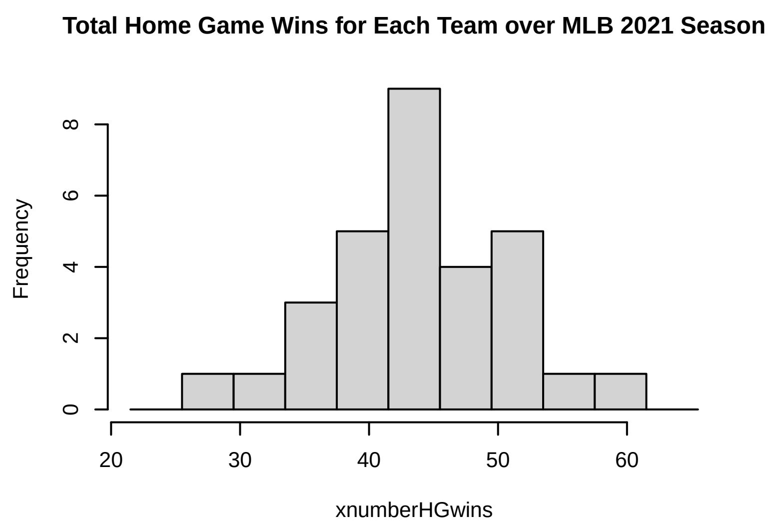



Figure 4: Total Home Games Wins

If all teams are actually the same, we would not expect to see much variation in the number of home game wins between the 30 teams.

What is the observed standard deviation for the 2021 Season?

It is 6.6079889.

So, let's use the binomial model to simulate our friend's claim.

```r
set.seed(777)

nn <- 5000 ### number of simulations

#xsim_range <- integer(nn)
#xsim_max <- integer(nn)
xsim_sd <- integer(nn)
#xsim_IQR <- integer(nn)

for(j in 1:nn) {

    xsim_Win <- rbinom(length(xWinTH), 1, prob=1/2)

    xagg_sim <- aggregate(xsim_Win, by=list(xdf_HT[ , "team"]), sum)
    #xsim_range[j] <- max(xagg_sim[ , "x"]) - min(xagg_sim[ , "x"])
    #xsim_max[j] <- max(xagg_sim[ , "x"])
    xsim_sd[j] <- sd(xagg_sim[ , "x"])
    #xsim_IQR[j] <- IQR(xagg_sim[ , "x"])
}
```

```
par(mfrow=c(1,1), cex=0.65)

hist(xsim_sd, xlim=c(2, 9))
abline(v=sd(xnumberHGwins), lwd=2, col="#33AA33")
```

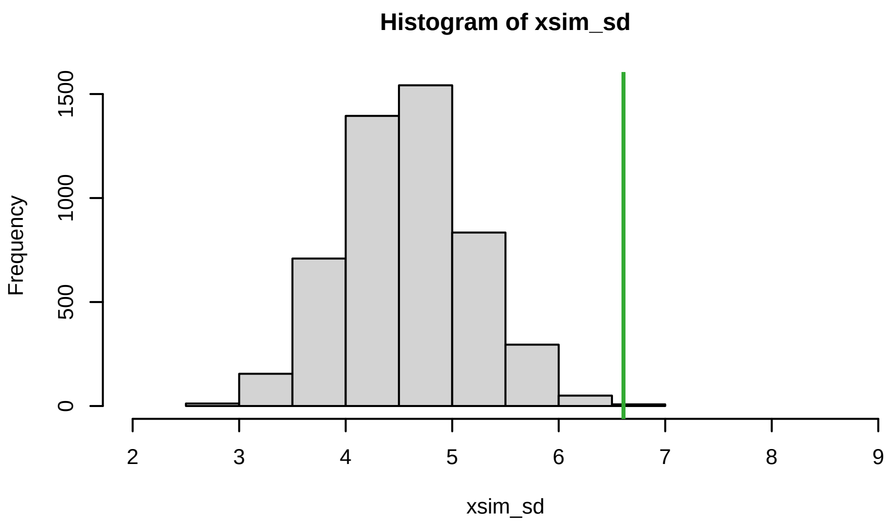



Figure 5: Simulation: Histogram of standard deviation of total home games won assuming our friend is correct

```
sum(xsim_sd >= sd(xagg[ , "x"])) / nn
```

```
## [1] 4e-04
```

## Your Work

Make sure to edit the "author" information in the YAML header near the top to include your name and UID.

Complete/answer the following.

1 — Suppose the Chattanooga P-Values play only 30 home games. Keeping the probability of win at 70%, what is the probability they will lose half or more of their home games? How does this compare with the example we looked at above where they play 40 home games? Comment on the difference.

```
  # Probability that they will lose half or more of their home games. They have a lot   # less of a cha
 sum(dbinom(I(0:15), size=30, prob=0.70))
```

```
## [1] 0.01693731
```

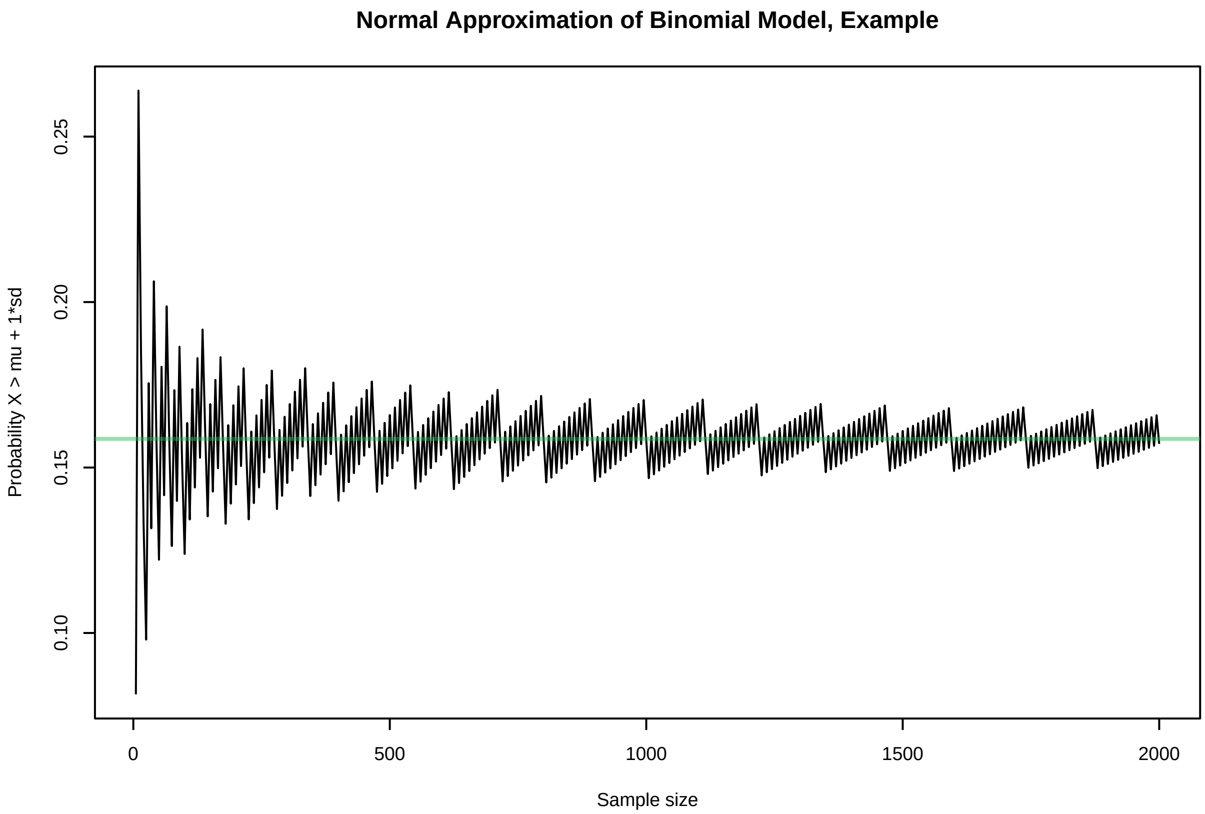
```
  # Compared to the example above.
  0.3087427-sum(dbinom(I(0:15), size=30, prob=0.70))
```

```
## [1] 0.2918054
```

2 — Consider the example where we illustrated the binomial probabilities converging to that produced by the normal distribution. Run this experiment yourself, except change the following: Have the binomial probability of success be only 10% (instead of the 50% we used above), and also look at the probability our respective random variable will be more than 2 standard deviations above the mean. Comment on your results.

```
p_success <- 0.1
xtrialsTry <- seq(5, 2000, by=5)
pout_vec <- numeric(length(xtrialsTry))
for(i in 1:length(xtrialsTry)) {
  xthis_numTrials <- xtrialsTry[i]
  xthis_mean <- p_success * xthis_numTrials
  xthis_sd <- sqrt( (1 - p_success) * p_success * xthis_numTrials )
  xdom <- I(0:xthis_numTrials)
  xdom_prob <- xdom[ xdom > (xthis_mean + 1 * xthis_sd) ]
  pout_vec[ i ] <- sum(dbinom(xdom_prob, size=xthis_numTrials, prob=p_success))
}

par(mfrow=c(1,1), lend=1, cex=0.60)
plot(xtrialsTry, pout_vec, type="l", lwd=1,
     xlab="Sample size", ylab="Probability X > mu + 1*sd",
     main="Normal Approximation of Binomial Model, Example")
abline(h=1 - pnorm(1, 0, 1), lwd=2, col="#22BB5577")
```




3 — Interpret the simulated MLB results from the above Examples Section.

The standard deviation (6.6) indicates that on average, the team is around 6 away from the mean. Now when looking at the binomial model that simulates the friends claim that there is no difference in home game % chance of a win vs non home game. In this simulation, we see that the average standard deviation in this case would be around 4.75 rather than the six that we have. This indicates that the friend is not right in his claim as the % wins does differ more greatly from the mean.

4 — Perform the same analysis we looked at with the MLB data with the NHL data. Are the results more or less convincing? Why?

```r
### here's a head start for you
xdf <- read.table( "/Users/apurvashah/Documents/GitHub/stats10/lab6/yourLab/NHL_20202021_game.tsv", sep=
tail(xdf)
```

```
##         date   season            startTime              endTime status VTabbr HTabbr
## 863 20210513 20202021 2021-05-14T00:00:00Z 2021-05-14T02:27:32Z  Final    MIN    STL
## 864 20210514 20202021 2021-05-15T00:00:00Z 2021-05-15T02:21:48Z  Final    TOR    WPG
## 865 20210515 20202021 2021-05-15T19:30:00Z 2021-05-15T21:53:17Z  Final    VAN    EDM
## 866 20210516 20202021 2021-05-17T02:30:00Z 2021-05-17T05:12:09Z  Final    CGY    VAN
## 867 20210518 20202021 2021-05-18T20:00:00Z 2021-05-18T22:39:15Z  Final    CGY    VAN
## 868 20210519 20202021 2021-05-19T19:30:00Z 2021-05-19T22:04:59Z  Final    VAN    CGY
##                      VT                HT periods VTgoals HTgoals VTfinal HTfinal
## 863      Minnesota Wild   St. Louis Blues       3       3       7       3       7
## 864 Toronto Maple Leafs     Winnipeg Jets       3       2       4       2       4
## 865   Vancouver Canucks   Edmonton Oilers       3       4       1       4       1
## 866      Calgary Flames Vancouver Canucks       4       6       5       6       5
```

```
## 867      Calgary Flames Vancouver Canucks       3       2       4       2       4
## 868   Vancouver Canucks    Calgary Flames       3       2       6       2       6
##                                                                         officials
## 863                         Dean Morton;Peter MacDougall;Jesse Marquis;Bryan Pancich
## 864                   Chris Schlenker;Graham Skilliter;Scott Cherrey;David Brisebois
## 865                        Kendrick Nicholson;Brad Meier;Derek Nansen;Kiel Murchison
## 866 Chris Schlenker;Graham Skilliter;Kendrick Nicholson;Derek Nansen;Kiel Murchison
## 867                   Chris Schlenker;Kendrick Nicholson;Derek Nansen;Kiel Murchison
## 868                   Chris Schlenker;Kendrick Nicholson;Derek Nansen;Kiel Murchison
##                              official_type
## 863         Referee;Referee;Linesman;Linesman
## 864         Referee;Referee;Linesman;Linesman
## 865         Referee;Referee;Linesman;Linesman
## 866 Referee;Referee;Referee;Linesman;Linesman
## 867         Referee;Referee;Linesman;Linesman
## 868         Referee;Referee;Linesman;Linesman
```

```r
dim(xdf)
```

```
## [1] 868  16
```

```r
N <- nrow(xdf)

WorL <- xdf[ , "HTfinal"] > xdf[ , "VTfinal"]

sum(xdf[ , "HTfinal"] == xdf[ , "VTfinal"]) ### no ties
```

```
## [1] 0
```

```r
# xagg <- aggregate(WorL, by=list(xdf[ , "HT"]), sum)

sd(xagg[ ,"x"])
```

```
## [1] 6.607989
```

```r
set.seed(777)
nn <- 5000
xsim_sd <- integer(nn)

# for(j in 1:nn) {
#   xsim_Win <- rbinom(length(xagg), 1, prob=1/2)
#   xagg_sim <- aggregate(xsim_Win, by=list(xdf_HT[ , "team"]), sum) #xsim_range[j] <- max(xagg_sim[ ,
#   xsim_sd[j] <- sd(xagg_sim[ , "x"])
#   #xsim_IQR[j] <- IQR(xagg_sim[ , "x"])
# }
#
# par(mfrow=c(1,1), cex=0.65)
# hist(xsim_sd, xlim=c(2, 9))
# abline(v=sd(xnumberHGwins), lwd=2, col="#33AA33")
#
# sum(xsim_sd >= sd(xagg[ , "x"])) / nn
```

This problem is far too hard when you do not teach any of the relevant data formatting specifications in lecture or discussion.