Will V. Denzel

Joshua Bennett *Washington State University*

In this Article we will discuss who is the better artor between Will Smith and Denzel Washington.

Keywords: T-tests, Histogram, Data Integrity, ScatterPlot, correlation tables

December 14, 2020

```
library(devtools);
```

```
## Loading required package: usethis
```

```
library(humanVerseWSU);

path.github = "https://raw.githubusercontent.com/MonteShaffer/humanVerseWSU/master/";

include.me = paste0(path.github, "misc/functions-nlp.R");
source_url( include.me );
```

```
## SHA-1 hash of file is 704afa69d52215d315cb5f59cdc020b0bbfd0b13
```

```
## Warning: package 'tm' was built under R version 4.0.3
```

```
## Loading required package: NLP
```

```
## Warning: package 'NLP' was built under R version 4.0.3
```

```
## Warning: package 'SentimentAnalysis' was built under R version 4.0.3
```

```
##
## Attaching package: 'SentimentAnalysis'
```

```
## The following object is masked from 'package:base':
##
##     write
```

```
include.me = paste0(path.github, "misc/functions-nlp-str.R");
source_url( include.me );
```

```
## SHA-1 hash of file is 6bdb234fa84eea995969dc29d6ff2a78f3982131
```

```
include.me = paste0(path.github, "misc/functions-nlp-stack.R");
source_url( include.me );
```

```
## SHA-1 hash of file is 034efbce0405954198545f8798e119b77a4809c9
```

```
include.me = paste0(path.github, "misc/functions-nlp-pos.R");
source_url( include.me );
```

```
## SHA-1 hash of file is d8c8cf01c8ead1b6d4228891aa52bac77084a6e7
```

```
## Warning: package 'openNLP' was built under R version 4.0.3
```

```
include.me = paste0(path.github, "humanVerseWSU/R/functions-encryption.R");
source_url( include.me );
```

```
## SHA-1 hash of file is da71dde620bed33db055778b752eefb476f7bf6b
```

```
include.me = paste0(path.github,"misc/functions-project-measure.R");
source_url( include.me);
```

```
## SHA-1 hash of file is 091aa1c443f262dce181395047d037a756331a65
```

```
## Warning: package 'Hmisc' was built under R version 4.0.3
```

```
## Loading required package: lattice
```

```
## Loading required package: survival
```

```
## Loading required package: Formula
```

```
## Loading required package: ggplot2
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following object is masked from 'package:NLP':
##
##     annotate
```

```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
path.to.nascent = "C:/Users/Alexander Nevsky/Dropbox/WSU-419/Fall 2020/__student_access__/unit_02_confi

folder.nlp = "nlp/";
path.to.nlp = paste0(path.to.nascent, folder.nlp);


###### UPDATES TO dataframe subset function ######
# inflation adjustments for NA ... and improvements on subsetting
include.me = paste0(path.github, "humanVerseWSU/R/functions-dataframe.R");
source_url( include.me );
```

```
## SHA-1 hash of file is 1149cbf3e865f692b50d4d1983e6364dc56ce62d
```

```
include.me = paste0(path.github, "humanVerseWSU/R/functions-inflation.R");
source_url( include.me );
```

```
## SHA-1 hash of file is b6d29327e3fe030ca132b135f4a89b6fc6a61a66
```

1 (IMDB) Custom library

This is a large dataset I harvested in September. It will allow us to explore more comprehensively the relationships of various features of the movie database. It is large (about 50MB), so installing may take some time if you are on a slow internet connection.

This dataset will be the source you will use on your final exam to answer the question posed earlier in the semester about Will Smith and Denzel Washington. You now have more analytics skills and with the new dataset there are more features you can extract.

```
# library(devtools);
# install_github("MonteShaffer/imdb/imdb"); # choose #3 to humanVerseWSU
# detach(package:imdb);
library(imdb);
packageVersion("imdb");  # '0.1.1'
```

```
## [1] '0.1.1'
```

```
# ?loadDataIMDB
```

1.1 Load data

Once this is run, a lot of memory will be required to read in the 23 compressed files.

```
install_github("MonteShaffer/imdb/imdb")
```

```
## Skipping install of 'imdb' from a github remote, the SHA1 (b29c6691) has not changed since last insta
##   Use 'force = TRUE' to force installation
```

```
imdb::loadDataIMDB();
names(imdb.data);
```

```
##  [1] "all.movies.creatives"         "all.movies.companies"
##  [3] "all.movies.actors.characters" "all.actors.rank"
##  [5] "all.actors.movies"            "all.actors.info"
##  [7] "moviecount.byyear"            "actors"
##  [9] "glue"                         "headliners"
## [11] "movies"                       "movies.df"
```

```
humanVerseWSU::loadInflationData();
```

Create Dataframe for the top gross and best reviews movies

```
library(KernSmooth)
```

```
## Warning: package 'KernSmooth' was built under R version 4.0.3
```

```
## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009
```

```
normalDiagnosticPlot = function(x,  normalityTest=TRUE,
                                    showDensity=TRUE,
                                    showNormal=TRUE,
                                    showSDs=FALSE,
```

```
                                    showAxis=TRUE
                                  )
  {
  xx = na.omit(x);
  x.stats = doStatsSummary(x);
  # x.table = table(x);

  # library(KernSmooth); # install.packages("KernSmooth", dependencies=TRUE);
  # bin.count = dpih(xx);
  # mybreaks = 100 * bin.count;

  mxlim = c(x.stats$mean - 3.5 * x.stats$sd ,
            x.stats$mean + 3.5 * x.stats$sd );
  h = hist(xx, breaks="Sturges", plot=F);
  mylim = c(0, max(h$counts));

  myMain = paste0(  "Histogram (mean: ",
                  round(x.stats$mean,digits=3),
                  ", sd: ",
                  round(x.stats$sd,digits=3),
                  ")"
                  );



mxlab = "";
  if(normalityTest)
    {
    isNormal = NULL;
    if(x.stats$shapiro.is.normal$`0.10`) { isNormal = 0.10; }
    if(x.stats$shapiro.is.normal$`0.05`) { isNormal = 0.05; }
    if(x.stats$shapiro.is.normal$`0.01`) { isNormal = 0.01; }

    isNormalResult = FALSE;
    if(!is.null(isNormal)) { isNormalResult = TRUE;}
    if(is.null(isNormal)) { isNormal = 0.05;}

    mxlab = paste0("Shapiro Normality test at (alpha = ",
                isNormal, ") is ... ",isNormalResult);
    }


### Histogram
  hist(xx, breaks="Sturges",  xlim=mxlim, ylim=mylim,
      xlab=mxlab, xaxt='n', main=myMain);

  if(showDensity)
    {
    par(new=T); # overlay
  ### Density Plot (remember first reading?)
    plot( density(xx, kernel="epanechnikov") ,
            xlim=mxlim,
            main="",
```

```
            xlab="",
            ylab="",
            xaxt='n',
            yaxt='n'
        );
    }


  if(showNormal)
    {
    par(new=T); # overlay
  ### Normal Curve
    xt = seq(-3.5,3.5, length=100);
            yt = dnorm(xt);

    plot( xt, yt,
          type="l",
          lwd=2,
          col = "red",
          axes=F,
          xlab="",
          ylab=""
        );
    }


  if(showSDs)
    {
  ### vertical lines at sd's of data ...
    abline(v=x.stats$mean,lwd=4,col="blue");
      abline(v=x.stats$mean - 1 * x.stats$sd , col="green",lwd=3);
      abline(v=x.stats$mean + 1 * x.stats$sd , col="green",lwd=3);
      abline(v=x.stats$mean - 2 * x.stats$sd , col="green",lwd=2);
      abline(v=x.stats$mean + 2 * x.stats$sd , col="green",lwd=2);
      abline(v=x.stats$mean - 3 * x.stats$sd , col="green",lwd=1);
      abline(v=x.stats$mean + 3 * x.stats$sd , col="green",lwd=1);
    }


  if(showAxis)
    {
  ### axis labels showing the ability to use expression
    axis(1, at = -3:3, labels = c( expression("-3"~hat(sigma) ), expression("-2"~sigma ), expression("-
        #axis(1, at = -3:3, labels = c("-3s", "-2s", "-1s", "hat(mu)", "1s", "2s", "3s"))
    }


  }
```

Denzels revenue is also much more stable, much lower highs

```
library(dplyr)
```

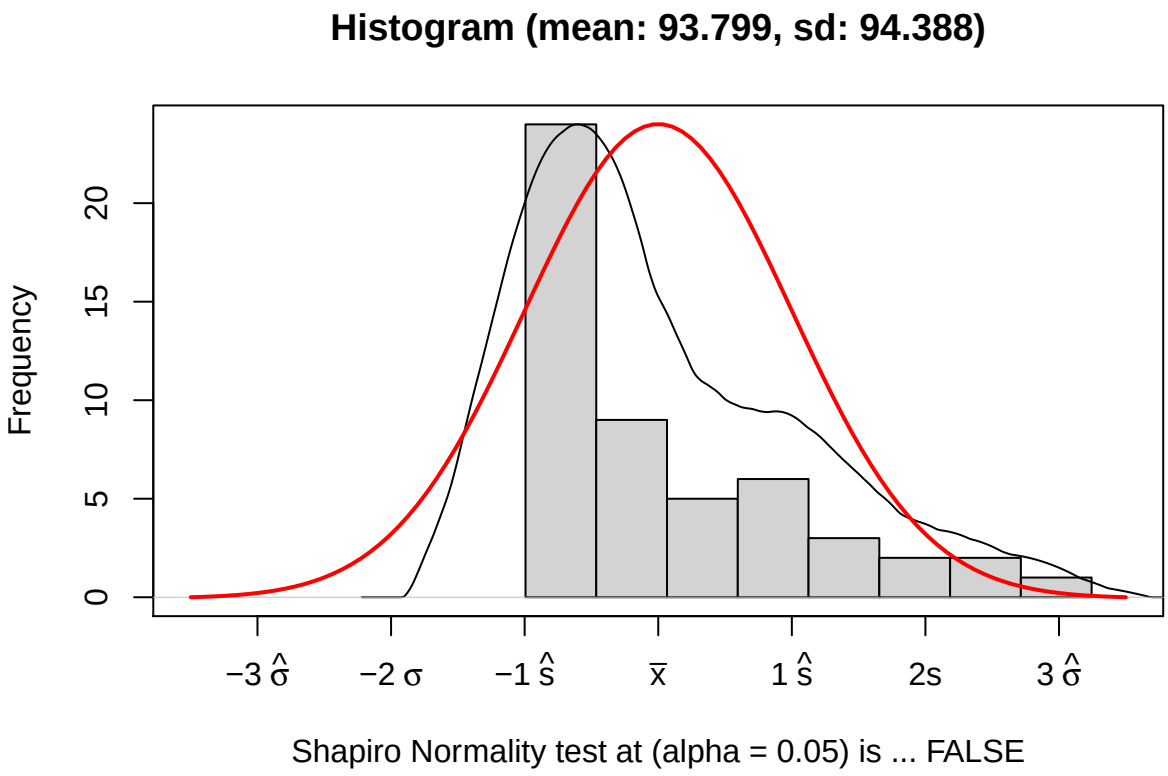
```
##
```

```
## Attaching package: 'dplyr'

## The following objects are masked from 'package:Hmisc':
##
##     src, summarize

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

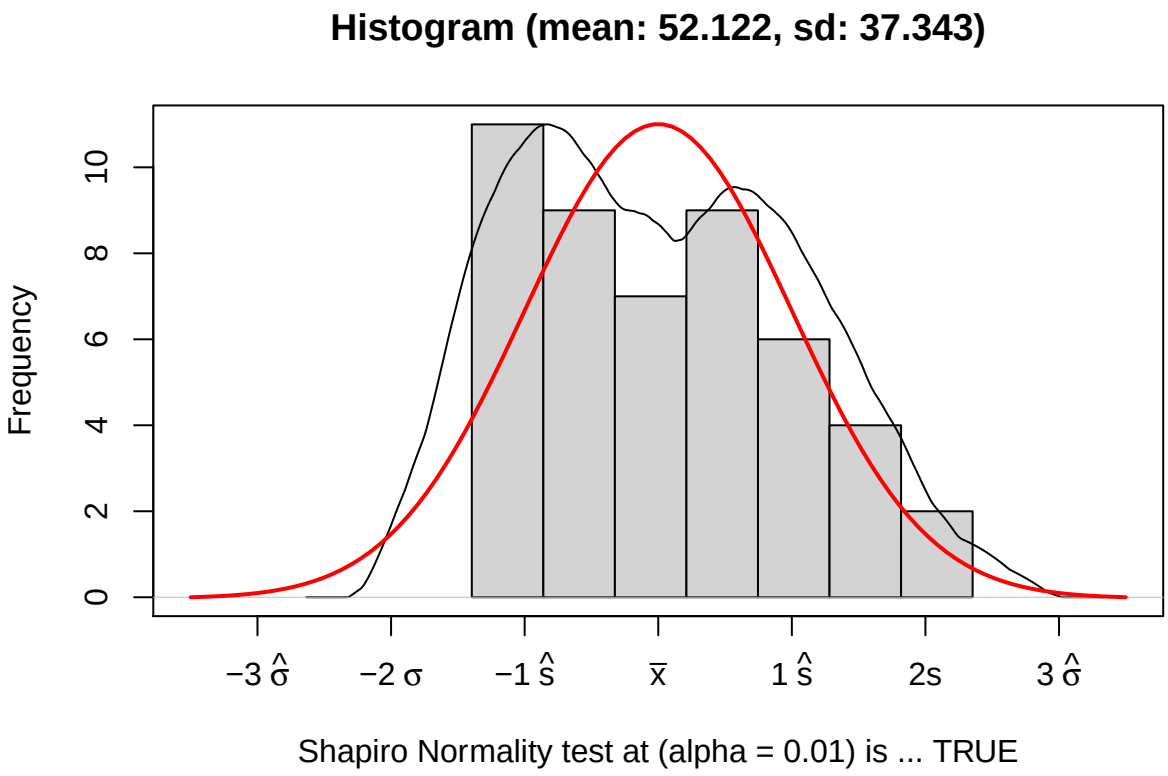
```
library(ggplot2)
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 4.0.3

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

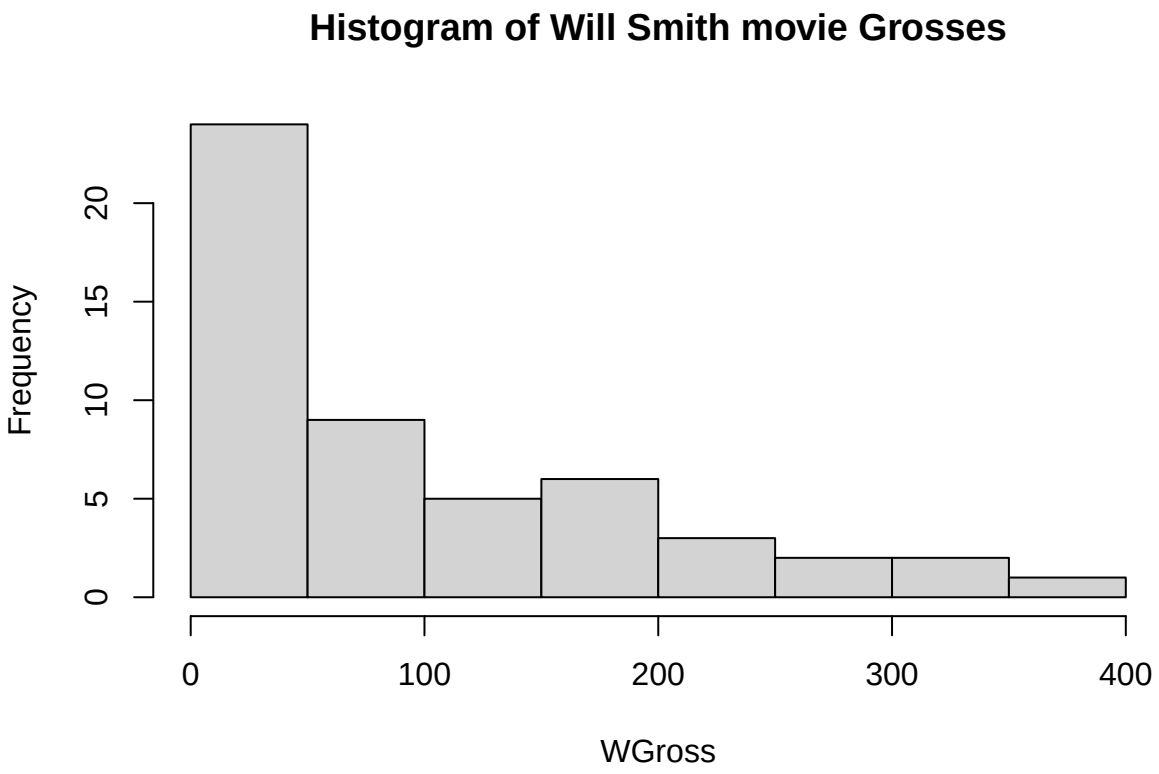
```
will.nmid = "nm0000226";
will.movies = IMDB.getMoviesForPerson(will.nmid);
will.n = nrow(will.movies);


denzel.nmid = "nm0000243";
denzel.movies = IMDB.getMoviesForPerson(denzel.nmid);
denzel.n = nrow(denzel.movies);
#will = IMDB.searchPersonName("Will* Smith*");
#denzel = IMDB.searchPersonName("Denzel* Washington")

myWill.df = will.movies

WillRatings = myWill.df$metacritic


myDenzel.df = denzel.movies
denzelTop = myDenzel.df

DMovieRatings = myDenzel.df$metacritic


WGross = will.movies$millions
DGross = denzel.movies$millions

normalDiagnosticPlot(WGross)
```



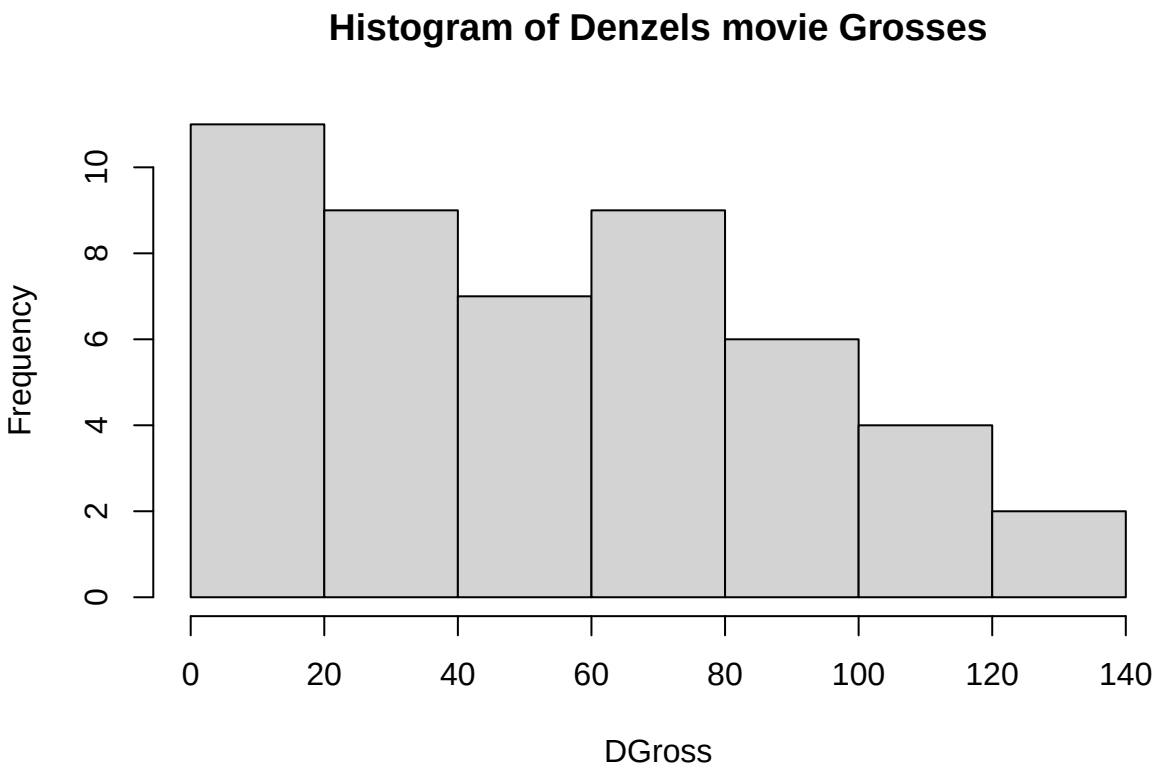

```
normalDiagnosticPlot(DGross)
```




```
hist(WGross, main = "Histogram of Will Smith movie Grosses")
```



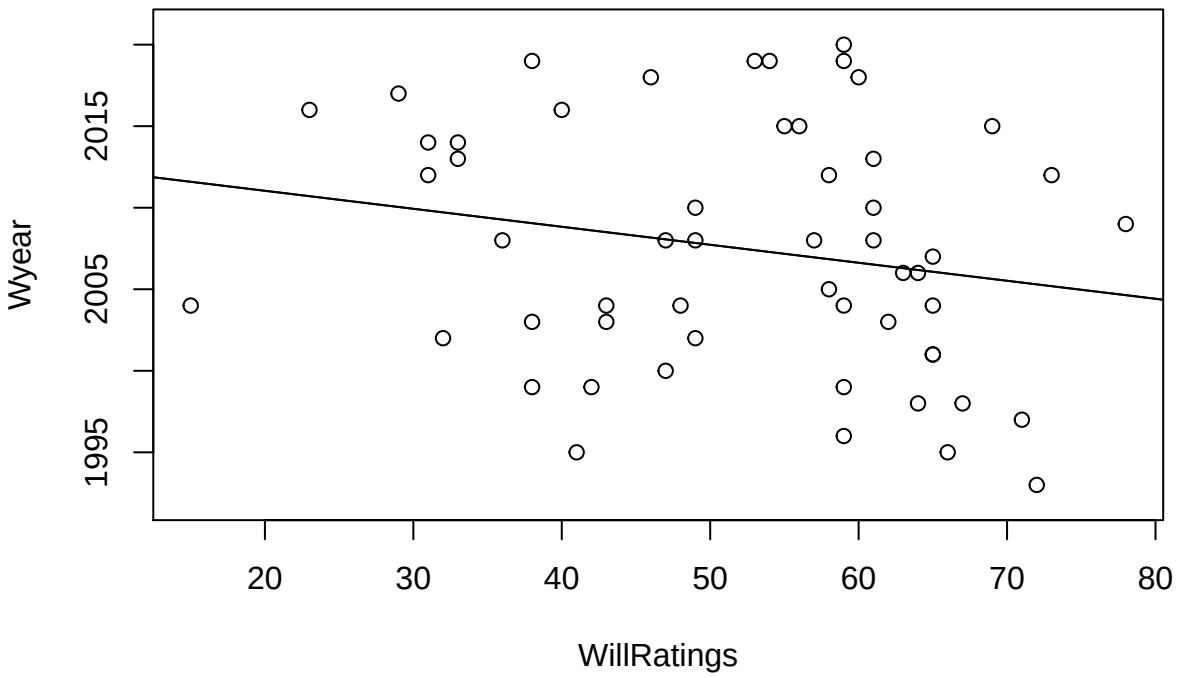

```
hist(DGross, main = "Histogram of Denzels movie Grosses")
```



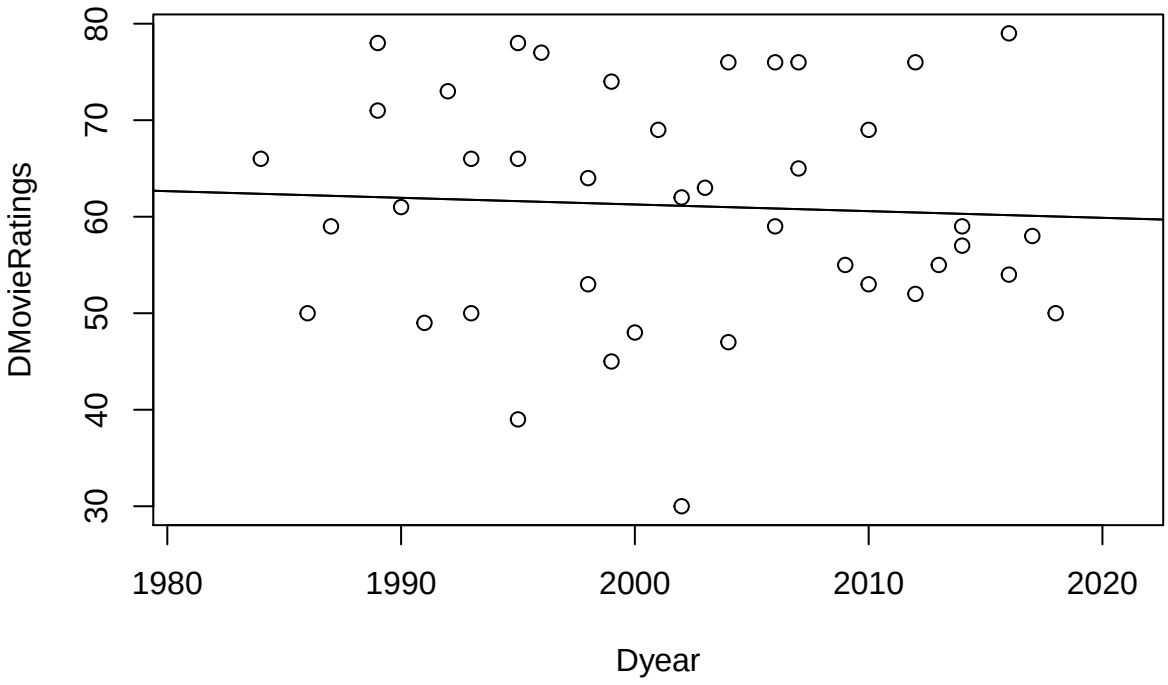

```
t.test(WGross, DGross)
```

```
##
##  Welch Two Sample t-test
##
## data:  WGross and DGross
## t = 2.9442, df = 67.651, p-value = 0.004434
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  13.42761 69.92714
## sample estimates:
## mean of x mean of y
##  93.79904  52.12167
```

```
Wyear = myWill.df$year
Dyear = myDenzel.df$year

plot(WillRatings, Wyear)
reg.n = lm(Wyear ~ WillRatings)
abline(reg.n)

abline(reg.n)
```



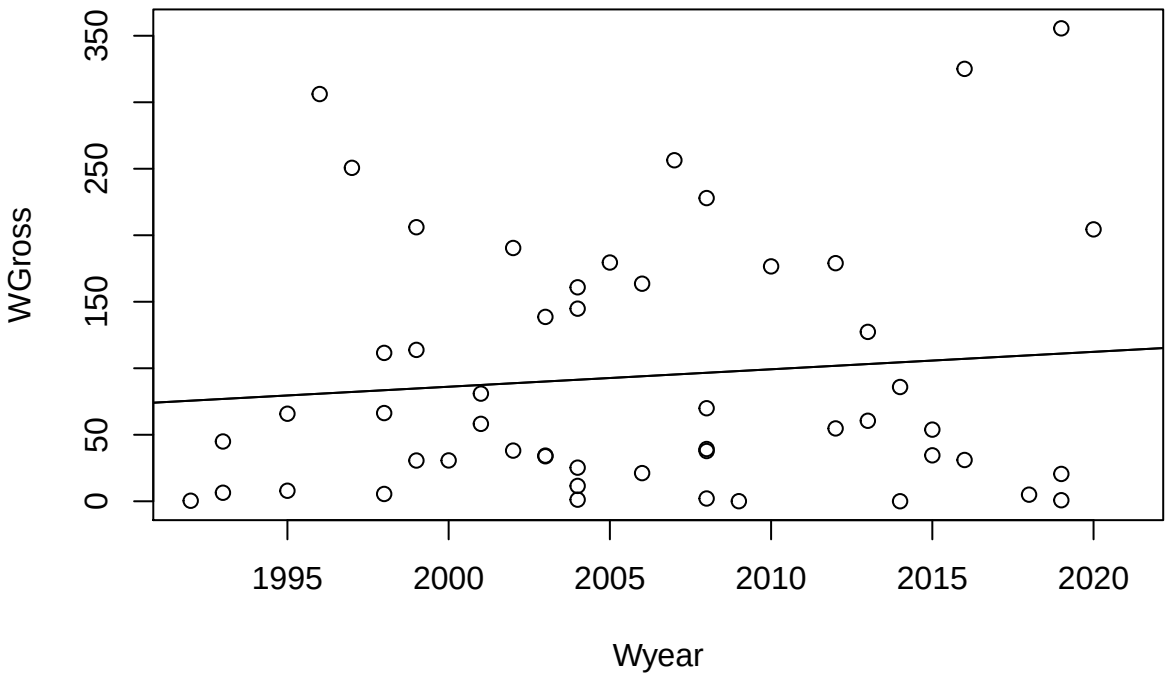

```
plot1 = plot(Dyear, DMovieRatings)
reg.n = lm(DMovieRatings ~ Dyear)
abline(reg.n)

abline(reg.n)
```



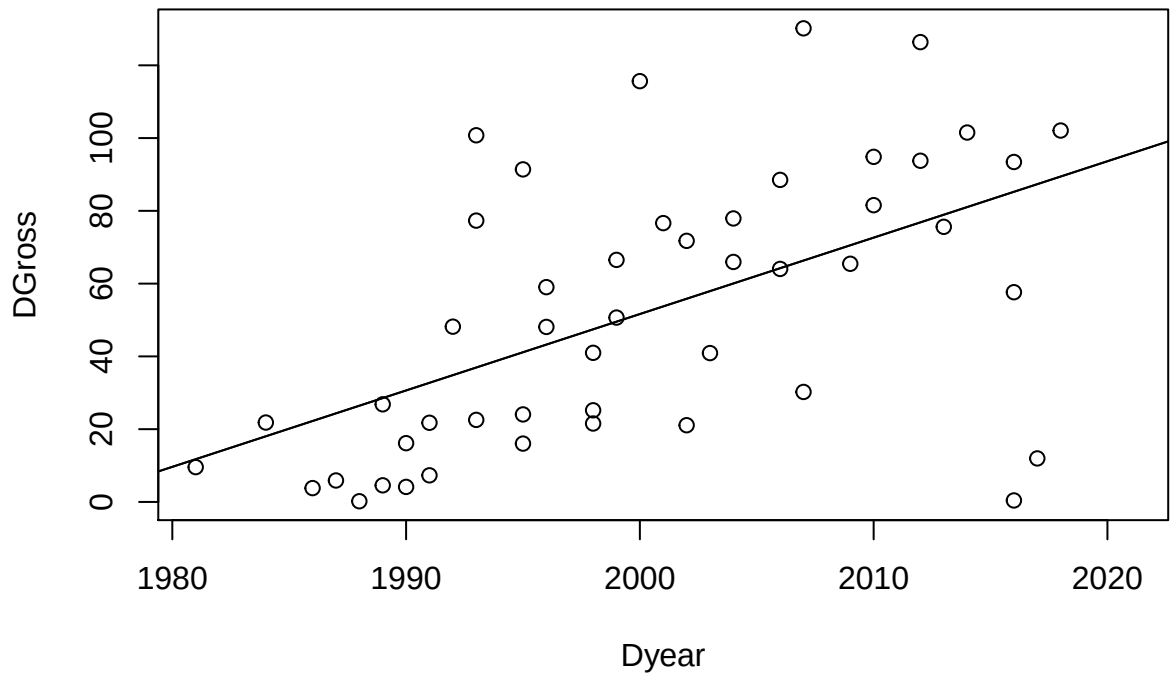

```
plot2 = plot(Wyear, WGross)
reg.n = lm(WGross ~ Wyear, col = "red")
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'col' will be disregarded
```

```
abline(reg.n)

abline(reg.n)
```




```
plot(Dyear, DGross)
reg.n = lm(DGross ~ Dyear)
abline(reg.n)

abline(reg.n)
```




```
summary(will.movies)
```

```
##      ttid               nmid                rank            year
##  Length:111         Length:111         Min.   :  1.0   Min.   :1992
##  Class :character   Class :character   1st Qu.: 28.5   1st Qu.:2002
##  Mode  :character   Mode  :character   Median : 56.0   Median :2007
##                                        Mean   : 56.0   Mean   :2007
##                                        3rd Qu.: 83.5   3rd Qu.:2014
##                                        Max.   :111.0   Max.   :2021
##                                                        NA's   :35
##     title              genre              rated              minutes
##  Length:111         Length:111         Length:111         Min.   : 52.0
##  Class :character   Class :character   Class :character   1st Qu.: 93.5
##  Mode  :character   Mode  :character   Mode  :character   Median :105.0
##                                                           Mean   :106.3
##                                                           3rd Qu.:118.0
##                                                           Max.   :157.0
##                                                           NA's   :40
##     ratings        metacritic        votes           millions
##  Min.   :2.300   Min.   :15.00   Min.   :    34   Min.   :  0.02
##  1st Qu.:5.700   1st Qu.:41.50   1st Qu.: 11735   1st Qu.: 24.24
##  Median :6.300   Median :56.00   Median : 56408   Median : 56.48
##  Mean   :6.228   Mean   :51.98   Mean   :131488   Mean   : 93.80
##  3rd Qu.:6.875   3rd Qu.:62.50   3rd Qu.:206926   3rd Qu.:161.54
##  Max.   :8.600   Max.   :78.00   Max.   :675160   Max.   :355.56
##  NA's   :37      NA's   :56      NA's   :45       NA's   :59
```

```
##   paragraph
##  Length:111
##  Class :character
##  Mode  :character
##
##
##
##
```

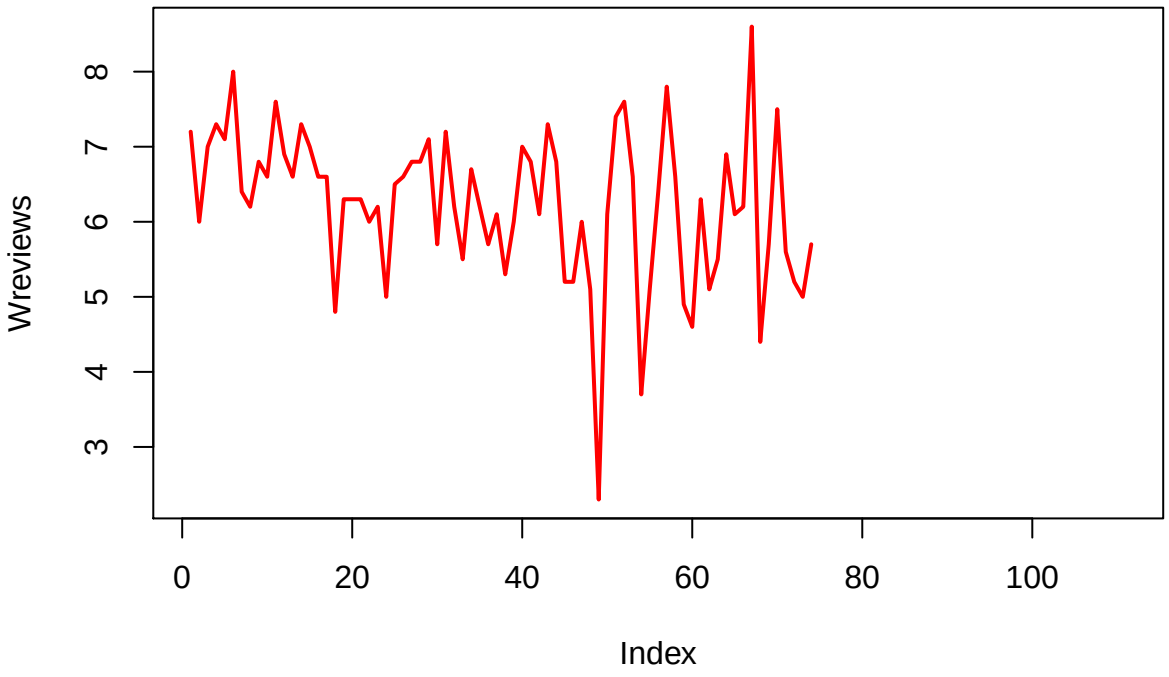
```
summary(denzel.movies)
```

```
##      ttid               nmid                rank         year
##  Length:61          Length:61          Min.   : 1   Min.   :1981
##  Class :character   Class :character   1st Qu.:16   1st Qu.:1994
##  Mode  :character   Mode  :character   Median :31   Median :2001
##                                        Mean   :31   Mean   :2002
##                                        3rd Qu.:46   3rd Qu.:2011
##                                        Max.   :61   Max.   :2021
##                                                     NA's   :2
##     title              genre              rated              minutes
##  Length:61          Length:61          Length:61          Min.   : 60.0
##  Class :character   Class :character   Class :character   1st Qu.:103.5
##  Mode  :character   Mode  :character   Mode  :character   Median :118.0
##                                                           Mean   :117.0
##                                                           3rd Qu.:127.5
##                                                           Max.   :202.0
##                                                           NA's   :6
##     ratings        metacritic        votes           millions
##  Min.   :5.000   Min.   :30.00   Min.   :   330   Min.   :  0.19
##  1st Qu.:6.500   1st Qu.:53.00   1st Qu.: 13118   1st Qu.: 21.43
##  Median :6.850   Median :61.00   Median : 74561   Median : 49.42
##  Mean   :6.815   Mean   :61.15   Mean   :113021   Mean   : 52.12
##  3rd Qu.:7.300   3rd Qu.:71.00   3rd Qu.:183051   3rd Qu.: 78.82
##  Max.   :8.500   Max.   :79.00   Max.   :383980   Max.   :130.16
##  NA's   :7       NA's   :20      NA's   :11       NA's   :13
##   paragraph
##  Length:61
##  Class :character
##  Mode  :character
##
##
##
##
```

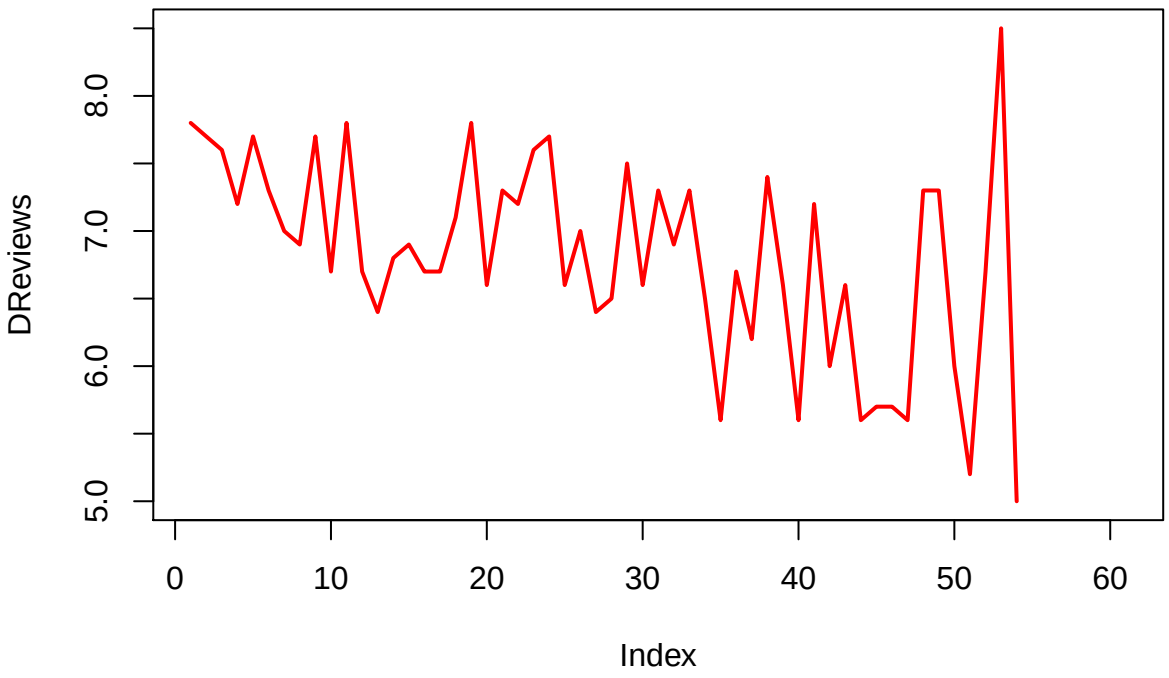
Reviews
You can see that Denzels reviews are much more stable

```
willReviews = will.movies

Wreviews = willReviews$ratings

DenzelReviews = denzel.movies

DReviews = DenzelReviews$ratings
```

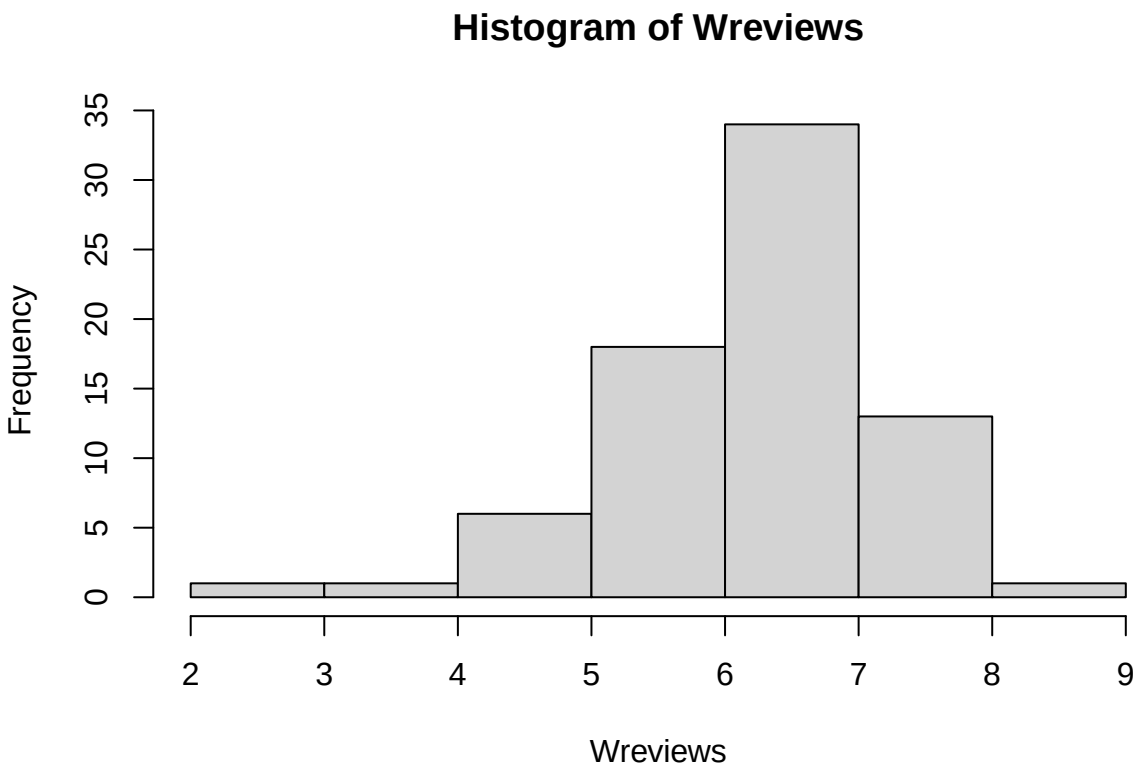
```r
plot(Wreviews, type = "l", lwd = 2, col = "red")
```



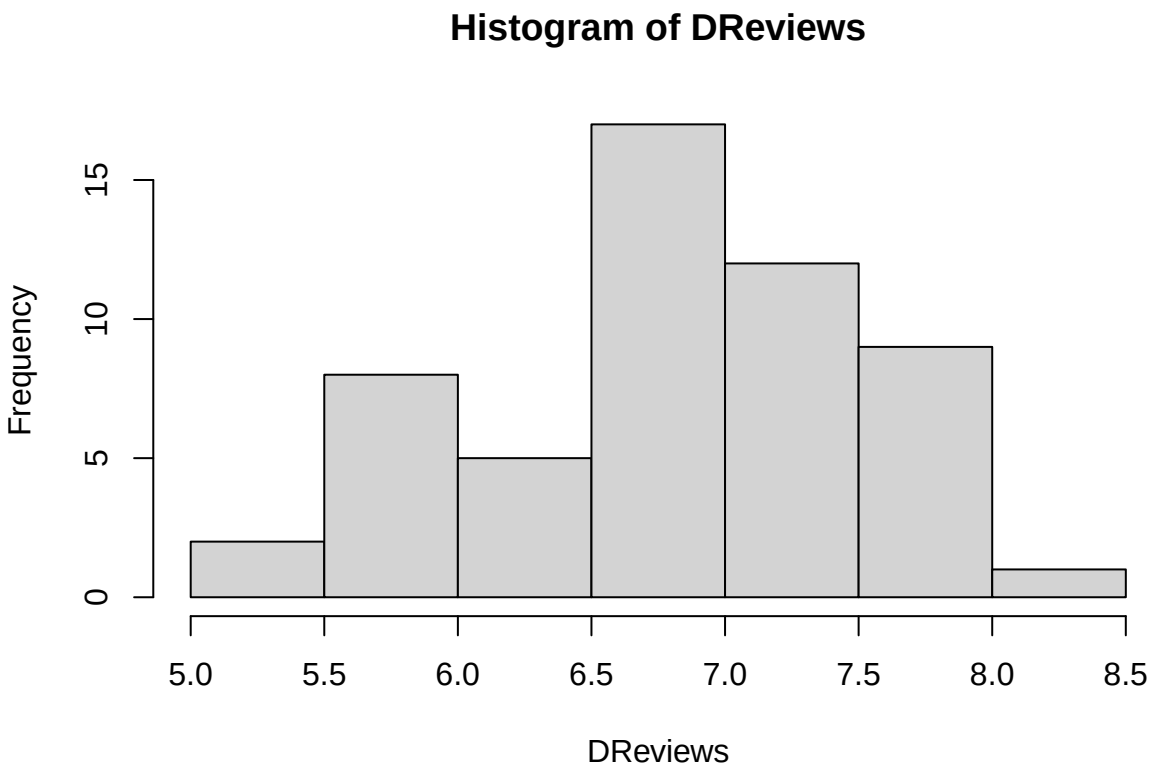

```r
plot(DReviews, type = "l", lwd = 2, col = "red")
```



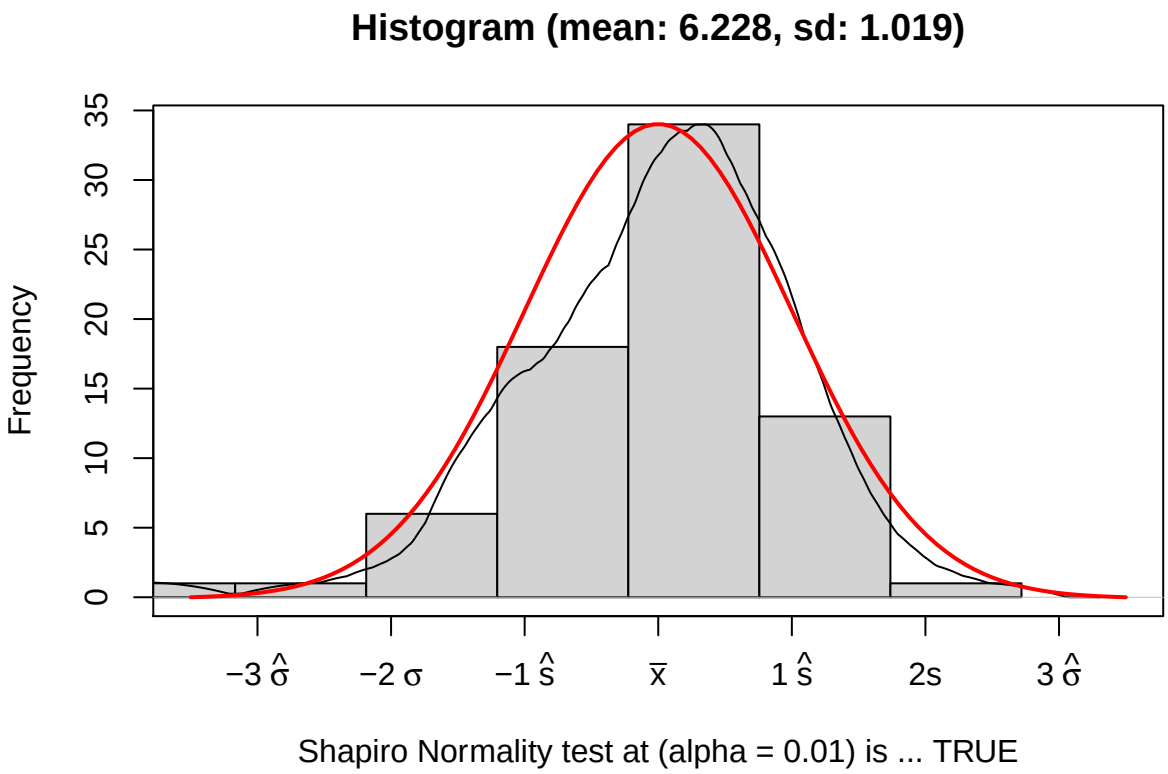

```
hist(Wreviews)
```



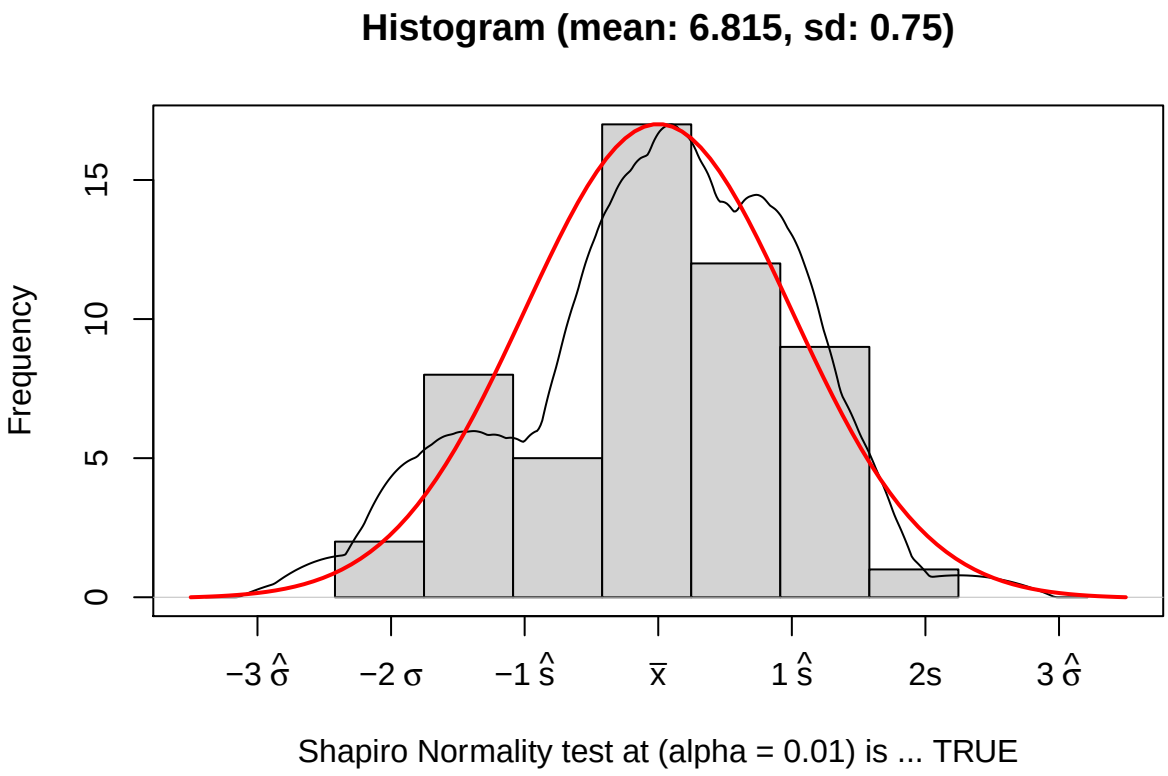

```
hist(DReviews)
```




```
normalDiagnosticPlot(willReviews$ratings)
```




```
normalDiagnosticPlot(DenzelReviews$ratings)
```




```
t.test(Wreviews, DReviews)
```

```
##
##  Welch Two Sample t-test
##
## data:  Wreviews and DReviews
## t = -3.7499, df = 125.99, p-value = 0.0002684
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.8959184 -0.2769545
## sample estimates:
## mean of x mean of y
##  6.228378  6.814815
```

Popularity vs year of movie (when were they popular)

1.1.1 Tables of Descriptive Statistics and Correlations

1.1.2 Tables of Descriptive Statistics and Correlations

```
library(gtsummary)
```

```
## Warning: package 'gtsummary' was built under R version 4.0.3
```

Table 1: will smith correlation

		M	SD	1	2
1	**metacritic**	51.4	14.06	1	
2	**millions**	100.5	95.05	.18	1
3	**year**	2006.3	7.16	-.30*	.08

Notes: Pearson pairwise correlations are reported;
a two-side test was performed to report correlation significance.

$^{\dagger}p < .10$ $^{*}p < .05$ $^{**}p < .01$ $^{***}p < .001$

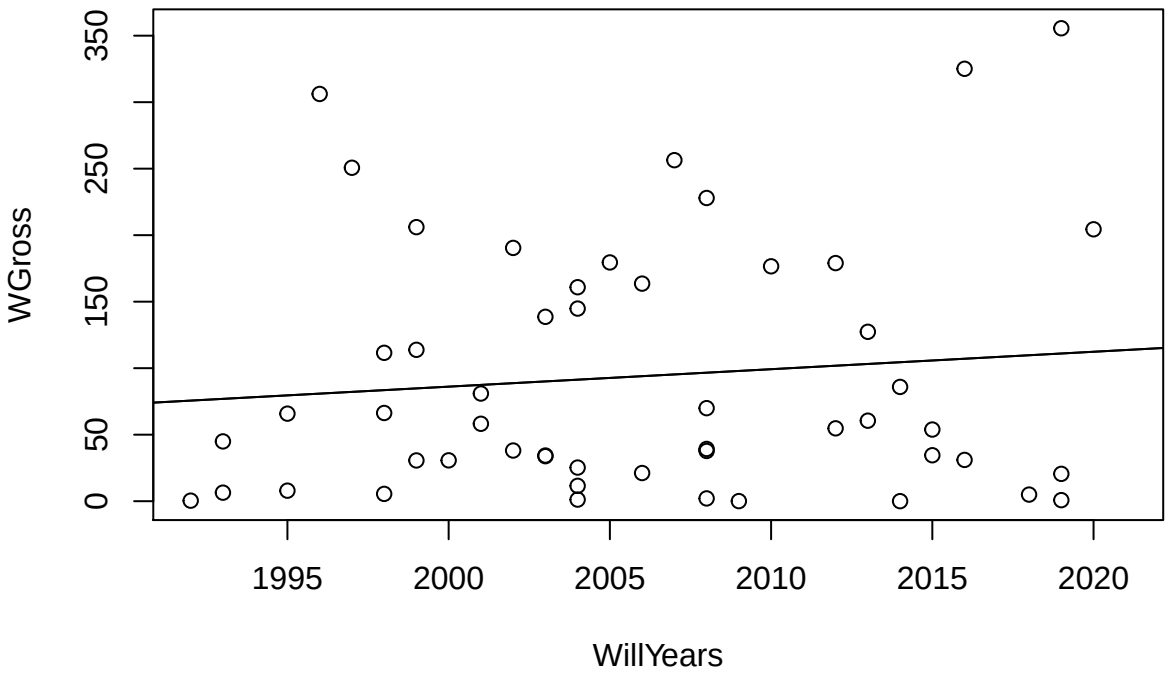
Table 2: Denzel Washington correlation

		M	SD	1	2
1	**metacritic**	51.4	14.06	1	
2	**millions**	100.5	95.05	.18	1
3	**year**	2006.3	7.16	-.30*	.08

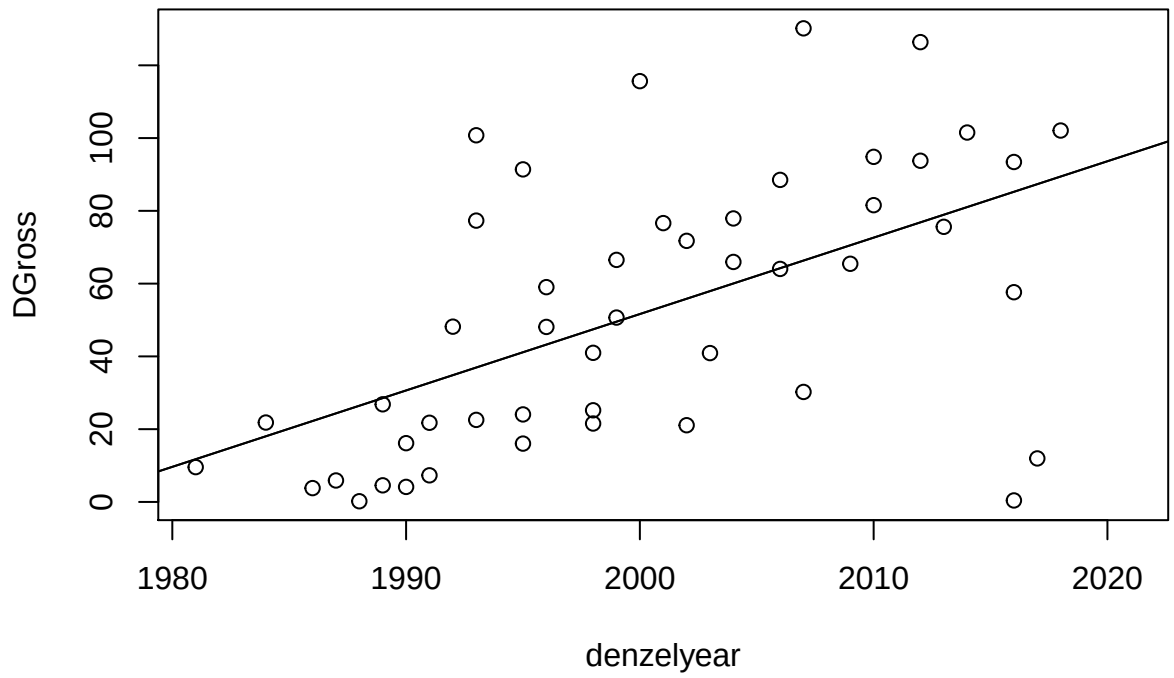
Notes: Pearson pairwise correlations are reported;
a two-side test was performed to report correlation significance.

$^{\dagger}p < .10$ $^{*}p < .05$ $^{**}p < .01$ $^{***}p < .001$

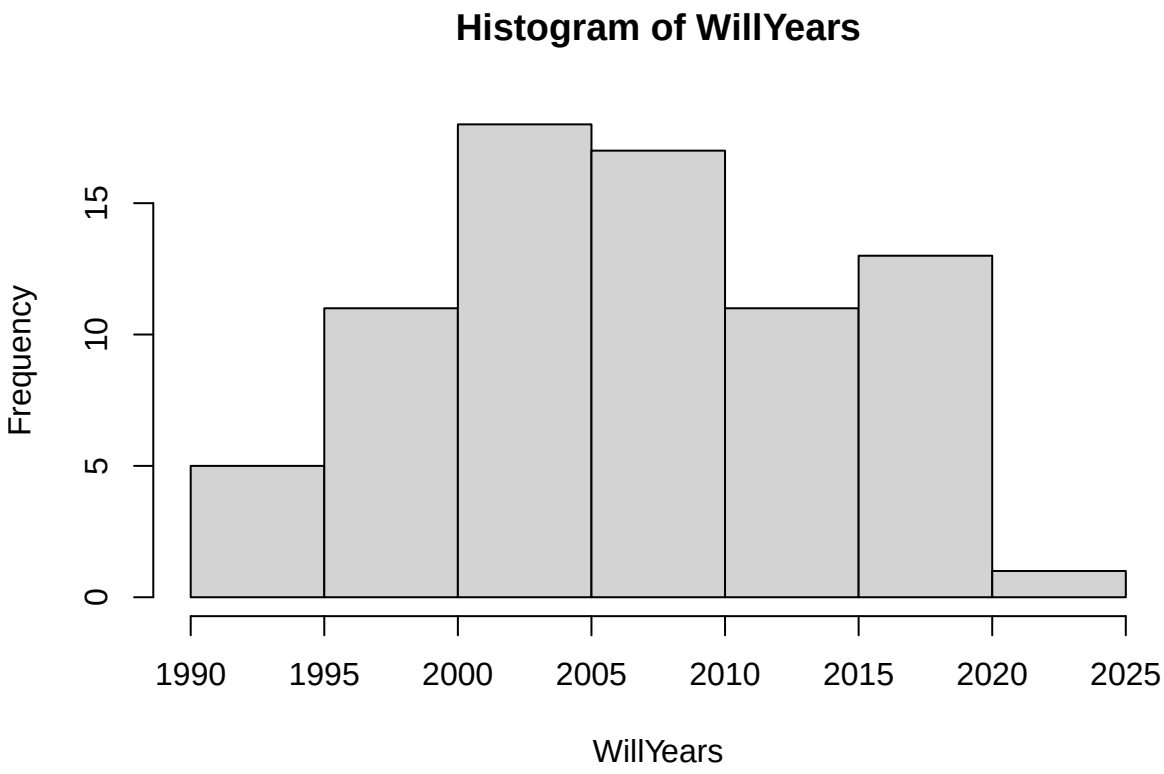
```
library(dplyr)

myWill.df = will.movies

WillYears= myWill.df$year

myDenzel.df = denzel.movies
denzelyear = myDenzel.df$year

WGross = myWill.df$millions
DGross = myDenzel.df$millions

plot(WillYears, WGross)
reg.n = lm(WGross ~ WillYears)
abline(reg.n)

abline(reg.n)
```



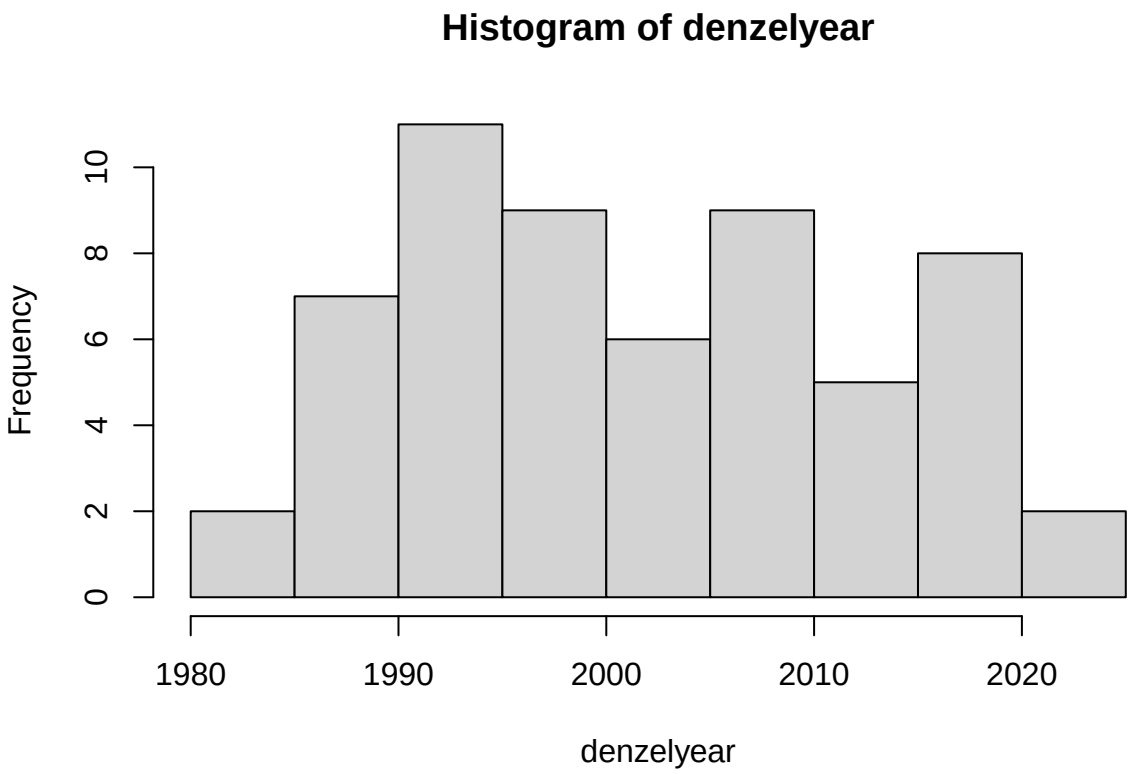

```
plot(denzelyear, DGross)
reg.n = lm(DGross ~ denzelyear)
abline(reg.n)

abline(reg.n)
```




```
hist(WillYears)
```




```
hist(denzelyear)
```




```
total <- merge(myWill.df, myDenzel.df, by="year")
#total

t.test(WGross, DGross)
```

```
##
##  Welch Two Sample t-test
##
## data:  WGross and DGross
## t = 2.9442, df = 67.651, p-value = 0.004434
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  13.42761 69.92714
## sample estimates:
## mean of x mean of y
##  93.79904  52.12167
```

```
#zero = myWill.df[,c("rating","millions")]
summary(myWill.df)
```

```
##      ttid               nmid                rank            year
##  Length:111         Length:111         Min.   :  1.0   Min.   :1992
##  Class :character   Class :character   1st Qu.: 28.5   1st Qu.:2002
##  Mode  :character   Mode  :character   Median : 56.0   Median :2007
##                                        Mean   : 56.0   Mean   :2007
##                                        3rd Qu.: 83.5   3rd Qu.:2014
```

```
##                                       Max.   :111.0   Max.   :2021
##                                                      NA's   :35
##     title               genre               rated              minutes
##  Length:111         Length:111         Length:111         Min.   : 52.0
##  Class :character   Class :character   Class :character   1st Qu.: 93.5
##  Mode  :character   Mode  :character   Mode  :character   Median :105.0
##                                                          Mean   :106.3
##                                                          3rd Qu.:118.0
##                                                          Max.   :157.0
##                                                          NA's   :40
##     ratings         metacritic        votes            millions
##  Min.   :2.300   Min.   :15.00   Min.   :    34   Min.   :  0.02
##  1st Qu.:5.700   1st Qu.:41.50   1st Qu.: 11735   1st Qu.: 24.24
##  Median :6.300   Median :56.00   Median : 56408   Median : 56.48
##  Mean   :6.228   Mean   :51.98   Mean   :131488   Mean   : 93.80
##  3rd Qu.:6.875   3rd Qu.:62.50   3rd Qu.:206926   3rd Qu.:161.54
##  Max.   :8.600   Max.   :78.00   Max.   :675160   Max.   :355.56
##  NA's   :37      NA's   :56      NA's   :45       NA's   :59
##   paragraph
##  Length:111
##  Class :character
##  Mode  :character
##
##
##
##
```

ENDNOTES

TABLE OF CONTENTS