

# Supplement to: A Method to Standardize Application Scores

Joseph Policarpio, Colin Van Orman, MS, Nora Few, PhD, and Ruoqing Zhu, PhD

12/9/2020

## Background

Nearly every medical school admissions office is tasked with evaluating thousands of medical school applications per year – far too many for any individual reviewer to score. Earlier this year, we explored scaling, standardizing, and normalizing medical school application scores from different reviewers. We developed a model and simulated 100 applications each scored by three reviewers. We found that our method of transforming each set of a single reviewer’s scores into a Gaussian distribution before merging with sets of scores provided by other reviewers may provide value in highlighting applications with the highest, or lowest scores provided by a single reviewer. Our method may help admissions offices to reduce inter-reviewer cognitive biases and allow for a method of standardization beyond a typical scoring rubric.

In a preliminary literature review, Pubmed search terms “Search (((school[Title]) AND ((admissions[Title]) OR admission[Title])) AND scor\*[Title/Abstract]) AND ((standard\*[Title/Abstract]) OR normal\*[Title/Abstract])” yielded 21 results, none of which included discussion on reviewing or scoring processes, nor a normalization of such scores. Further Pubmed search terms "(secondary application[Title/Abstract]) OR supplemental application[Title/Abstract] AND admission\*[Title/Abstract]" yielded no results that mentioned the selection process.

Here we propose a method of scaling, standardizing and normalizing scores from various reviewers.

## Simulate Three Reviewers: Harsh, Typical, and Lenient

We will first assume a panel of three reviewers of varying strictness. A harsh, who grades on a more difficult scale than the average reviewer and thus provides lower scores of applications, is considered here to be a Hawk. A reviewer who grades on a more lenient scale and thus provides higher scores is considered here to be a Dove. An average reviewer here is given the name Average Jane.

The scores provided by each of the reviewers in the panel will be modeled using the R programming language. In the first example, to demonstrate proof of concept, we will have each reviewer score 100 applications.

Assume the possible range of scores to be from 0 to 50.

First we load the required RStudio computing packages, then set N, the number of applications to 100.

```
library(timeDate)
library(timeSeries)
library(fBasics)
library(fGarch)
```

```
N = 100
```

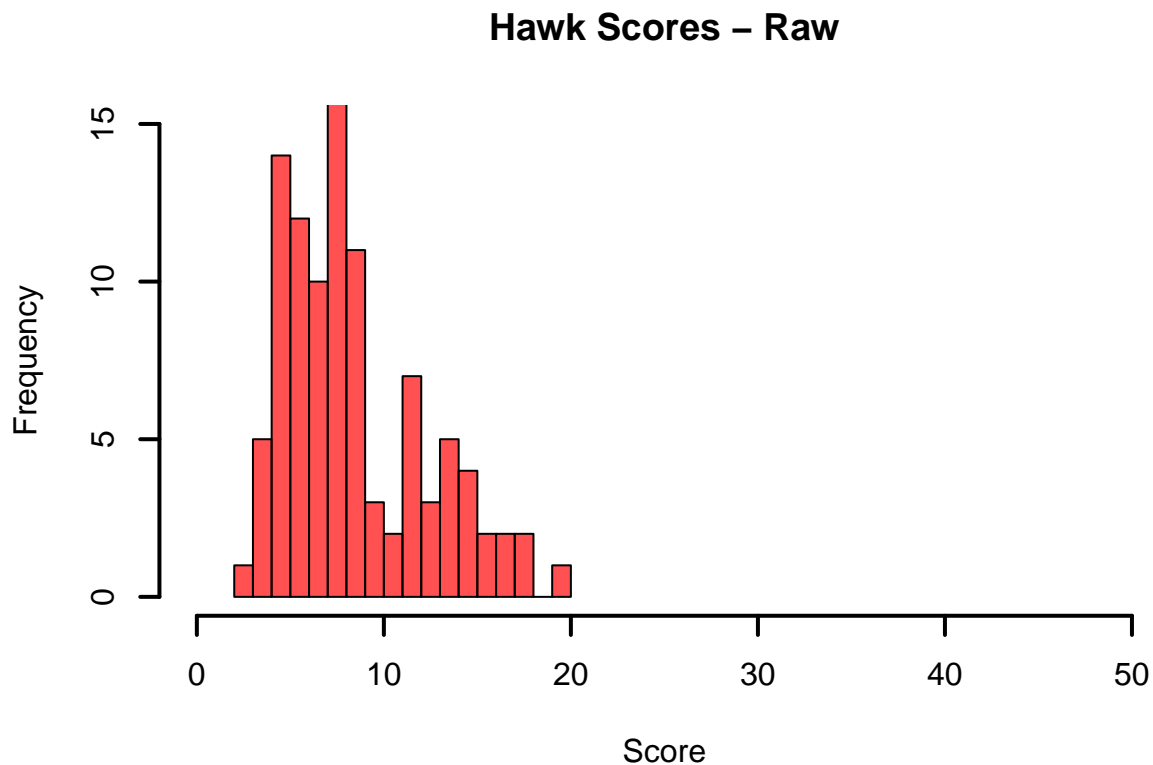
```
a = 175 # transparency value of histogram color, out of 255
```

We then model the scores provided by each of the three reviewers. The means and standard deviations are selected arbitrarily here.

Hawk (red): Right skewed distribution with mean of 9 and standard deviation of 4. The scores are then displayed on a histogram.

```
set.seed(1)
hawk1 = rsnorm(N, mean = 9, sd = 4, xi = 3)
hawk = data.frame(Raw_Hawk = as.vector(round(hawk1)))
hawk$Raw_Hawk_Sorted = sort(hawk$Raw_Hawk)
ymax = 15 # max(hawk$Raw_Hawk)

hawk_hist = hist(hawk$Raw_Hawk_Sorted, breaks = seq(min(hawk$Raw_Hawk)-1,
  max(hawk$Raw_Hawk), by=1), col=rgb(255,0,0,a, maxColorValue = 255), lwd=2,
  xlim=range(0,50), ylim=c(0,15), main="Hawk Scores - Raw", xlab="Score",
  ylab="Frequency")
```

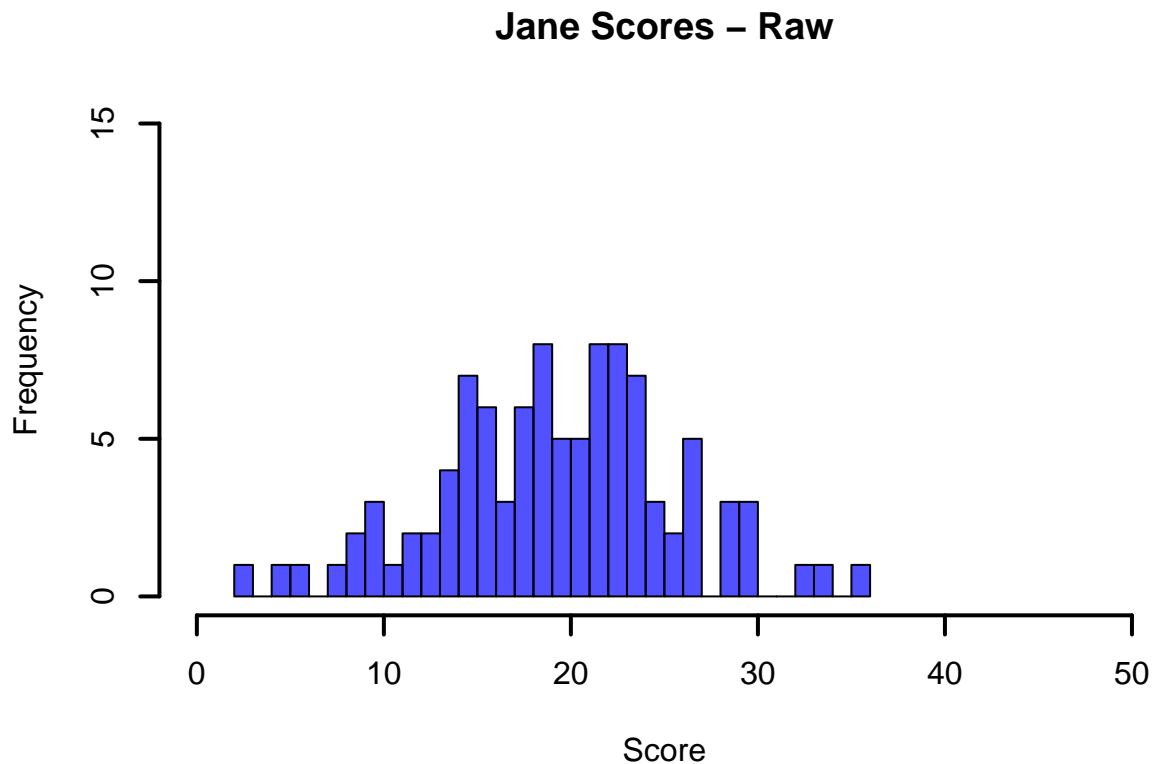


Jane (violet): Normal distribution with mean of 19 and standard deviation of 7. The scores are then displayed on a histogram.

```
set.seed(1)
jane1 = rnorm(N, 19, 7)
jane = data.frame(Raw_Jane = as.vector(round(jane1)))
```

```
jane$Raw_Jane_Sorted = sort(jane$Raw_Jane)

jane_hist = hist(jane$Raw_Jane_Sorted, breaks = seq(min(jane$Raw_Jane)-1,
  max(jane$Raw_Jane), by=1), col=rgb(0,0,255,a, maxColorValue = 255), lwd=2,
  xlim=range(0,50), ylim=c(0, ymax), main="Jane Scores - Raw", xlab="Score",
  ylab="Frequency")
```

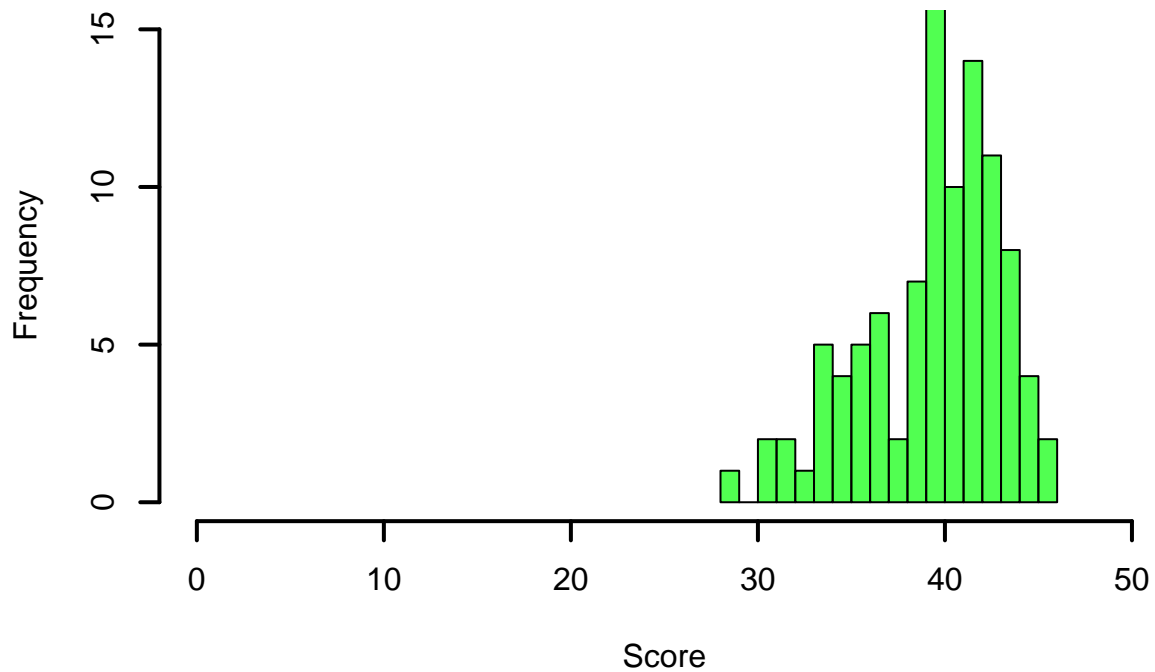


Dove (green): Left skewed distribution with mean of 40 and standard deviation of 4. The scores are then displayed on a histogram.

```
set.seed(1)
dove1 = rsnorm(N, mean = 40, sd = 4, xi = 0.5)
dove = data.frame(Raw_Dove = as.vector(round(dove1)))
dove$Raw_Dove_Sorted = sort(dove$Raw_Dove)

dove_hist = hist(dove$Raw_Dove_Sorted, breaks = seq(min(dove$Raw_Dove)-1,
  max(dove$Raw_Dove), by=1), col=rgb(0,255,0,a, maxColorValue = 255), lwd=2,
  xlim=range(0,50), ylim=c(0,ymax), main="Dove Scores - Raw", xlab="Score",
  ylab="Frequency")
```

## Dove Scores – Raw



A table showing the first 20 application scores is displayed.

```
mydata = data.frame(App_No = c(1:N),
                     Hawk_raw = hawk$Raw_Hawk_Sorted,
                     Jane_raw = jane$Raw_Jane_Sorted,
                     Dove_raw = dove$Raw_Dove_Sorted
                     )
head(mydata, n=20)
```

##	App_No	Hawk_raw	Jane_raw	Dove_raw
## 1	1	3	3	29
## 2	2	4	5	31
## 3	3	4	6	31
## 4	4	4	8	32
## 5	5	4	9	32
## 6	6	4	9	33
## 7	7	5	10	34
## 8	8	5	10	34
## 9	9	5	10	34
## 10	10	5	11	34
## 11	11	5	12	34
## 12	12	5	12	35
## 13	13	5	13	35
## 14	14	5	13	35
## 15	15	5	14	35
## 16	16	5	14	36

## 17	17	5	14	36
## 18	18	5	14	36
## 19	19	5	15	36
## 20	20	5	15	36

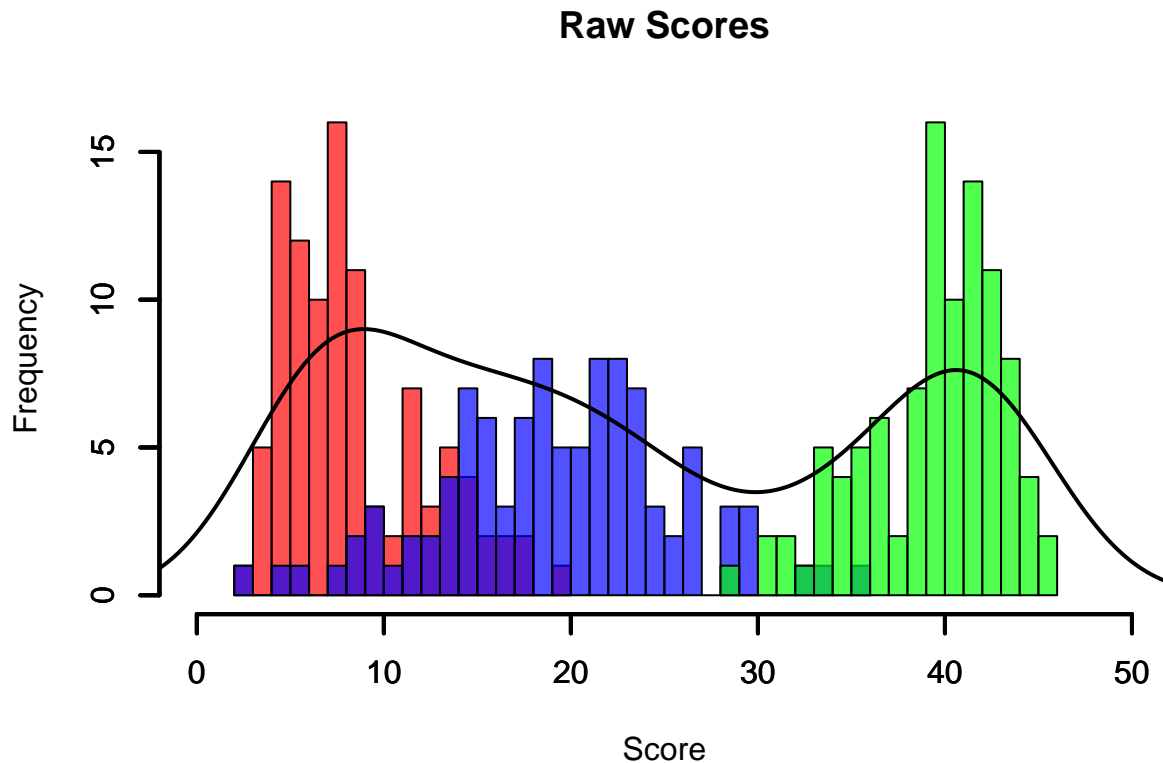
All three histograms are then displayed together. The kernel density of all scores is shown with the black line.

```
ymax = max(c(hawk_hist$counts, jane_hist$counts, dove_hist$counts))

hist(hawk$Raw_Hawk_Sorted, breaks = seq(min(hawk$Raw_Hawk)-1, max(hawk$Raw_Hawk), by=1),
     col=rgb(255,0,0,a, maxColorValue = 255), lwd=2, xlim=range(0,50),
     ylim=range(0,ymax), main=NULL, xlab=NULL, ylab=NULL)
par(new=TRUE)
hist(jane$Raw_Jane_Sorted, breaks = seq(min(jane$Raw_Jane)-1, max(jane$Raw_Jane), by=1),
     col=rgb(0,0,255,a, maxColorValue = 255), lwd=2, xlim=range(0,50),
     ylim=range(0,ymax), main=NULL, xlab=NULL, ylab=NULL)
par(new=TRUE)
hist(dove$Raw_Dove_Sorted, breaks = seq(min(dove$Raw_Dove)-1, max(dove$Raw_Dove), by=1),
     col=rgb(0,255,0,a, maxColorValue = 255), lwd=2, xlim=range(0,50),
     ylim=range(0,ymax), main="Raw Scores", xlab="Score", ylab="Frequency")

scale_factor = N*3
x = seq(0, 40, length.out = 3*N)
all13 = sort(c(hawk$Raw_Hawk, jane$Raw_Jane, dove$Raw_Dove))

y2 = density(all13)
y5 = scale_factor*y2$y
lines(y2$x, y5, col = "black", lwd = 2)
```



## Applying our Method $f(x) = Q(x/(n+1))$

The scores are first ranked from 1 to the total number of applications reviewed, then transformed to a normal distribution using the quantile function and  $x/(n+1)$  copula.

```
hawk$rank = rank(hawk$Raw_Hawk)
jane$rank = rank(jane$Raw_Jane)
dove$rank = rank(dove$Raw_Dove)

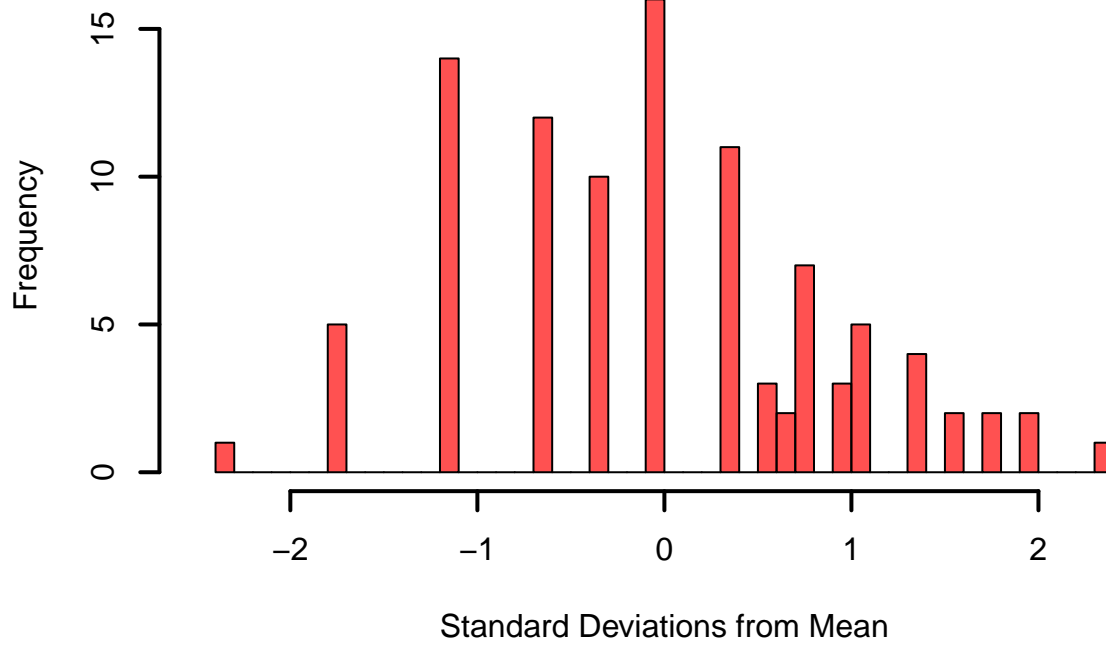
hawk$normal = qnorm(hawk$rank/(1+N))
jane$normal = qnorm(jane$rank/(1+N))
dove$normal = qnorm(dove$rank/(1+N))

nbreak = max(cbind(length(seq(min(hawk$normal), max(hawk$normal), by=0.1)),
                      length(seq(min(jane$normal), max(jane$normal), by=0.1)),
                      length(seq(min(dove$normal), max(dove$normal), by=0.1))))
```

The standardized scores are then displayed on individual histograms and again on a combined histogram.

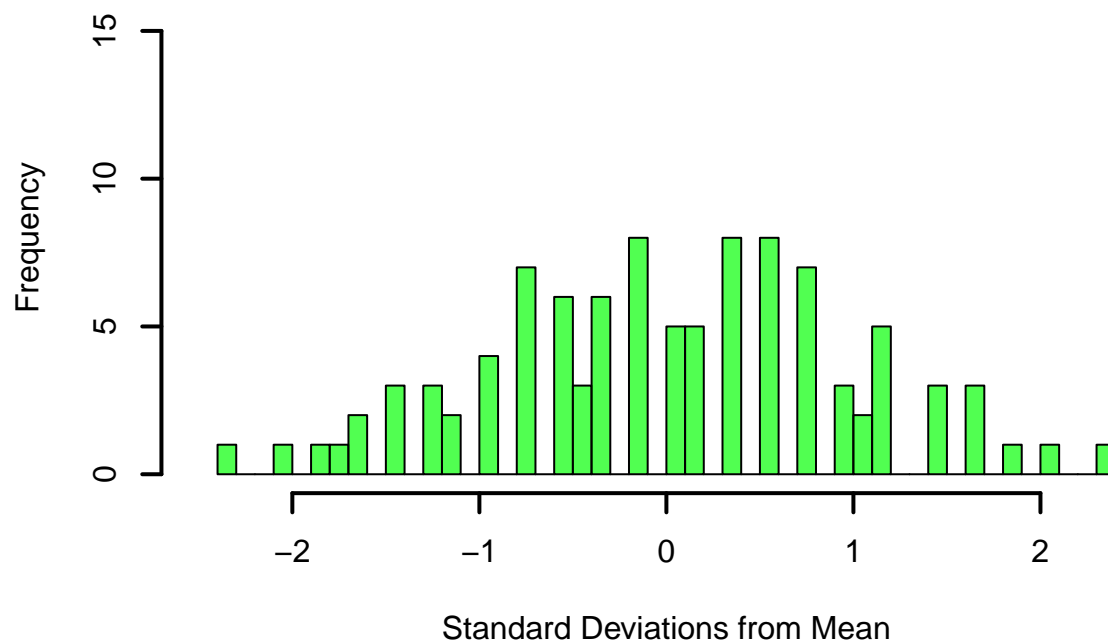
```
hawk_hist2 = hist(hawk$normal, breaks = nbreak,
                  col=rgb(255,0,0,a, maxColorValue = 255), lwd=2,
                  xlim=range(-2.5,2.5), ylim=c(0,ymax), main="Standardized Hawk Scores",
                  xlab="Standard Deviations from Mean",
                  ylab="Frequency")
```

## Standardized Hawk Scores



```
jane_hist2 = hist(jane$normal, breaks = nbreak,  
  col=rgb(0,255,0,a, maxColorValue = 255), lwd=2,  
  xlim=range(-2.5,2.5), ylim=c(0,ymax), main="Standardized Jane Scores",  
  xlab="Standard Deviations from Mean",  
  ylab="Frequency")
```

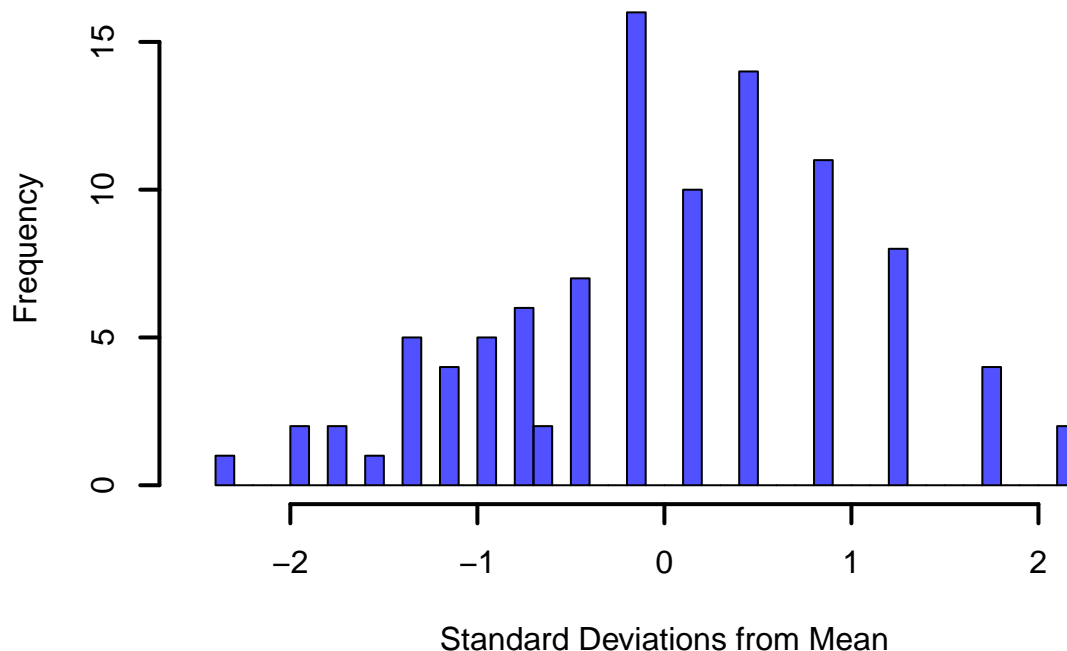
## Standardized Jane Scores



```
dove_hist2 = hist(dove$normal, breaks = nbreak,  
  col=rgb(0,0,255,a, maxColorValue = 255), lwd=2,  
  xlim=range(-2.5,2.5), ylim=c(0,ymax), main="Standardized Dove Scores",  
  xlab="Standard Deviations from Mean",  
  ylab="Frequency")
```



## Standardized Dove Scores



```
hawk_hist2 = hist(hawk$normal, breaks = nbreak,
  col=rgb(255,0,0,a, maxColorValue = 255), lwd=2,
  xlim=range(-2.5,2.5), ylim=c(0,ymax), main="All Standardized Scores",
  xlab="Standard Deviations from Mean",
  ylab="Frequency")

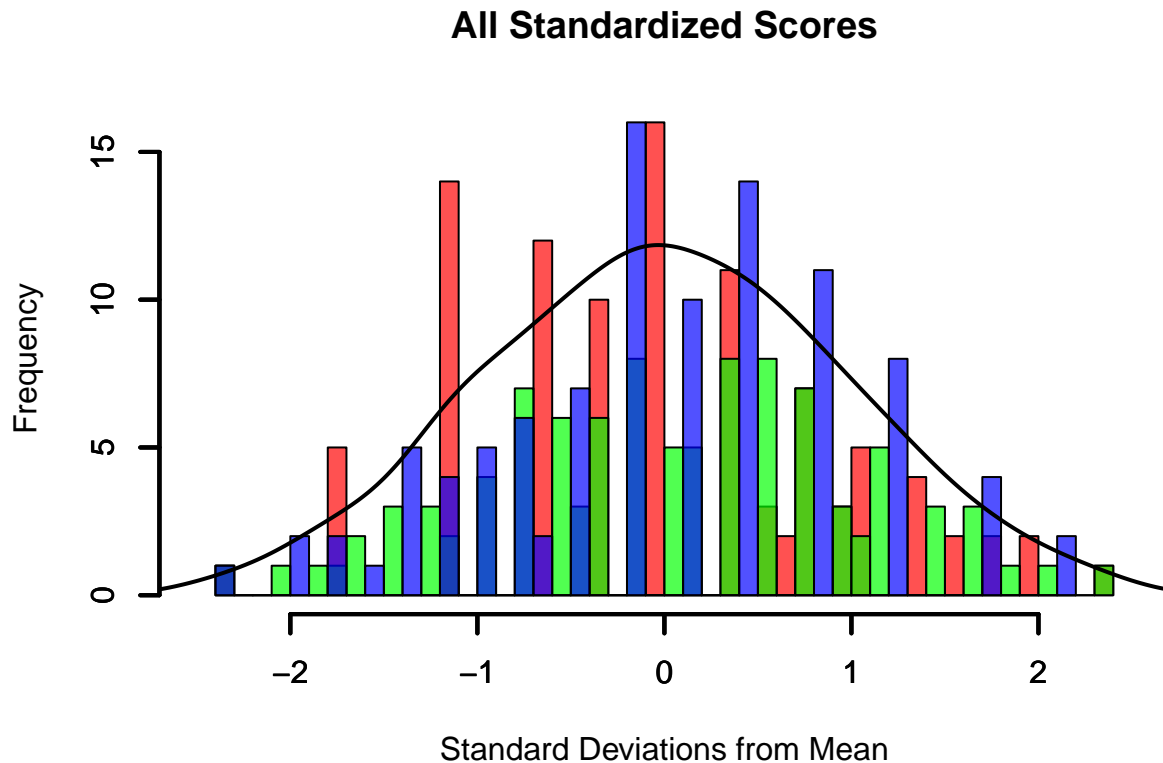
par(new=TRUE)

jane_hist2 = hist(jane$normal, breaks = nbreak,
  col=rgb(0,255,0,a, maxColorValue = 255), lwd=2,
  xlim=range(-2.5,2.5), ylim=c(0,ymax), main=NULL, xlab=NULL, ylab=NULL)

par(new=TRUE)

dove_hist2 = hist(dove$normal, breaks = nbreak,
  col=rgb(0,0,255,a, maxColorValue = 255), lwd=2,
  xlim=range(-2.5,2.5), ylim=c(0,ymax), main=NULL, xlab=NULL, ylab=NULL)

scale_factor2 = N*0.3
all3_2 = sort(c(hawk$normal, jane$normal, dove$normal))
y2 = density(all3_2)
y_scaled2 = scale_factor2*y2$y
lines(y2$x, y_scaled2, col = "black", lwd = 2)
```



## Repeat the simulation with less applications

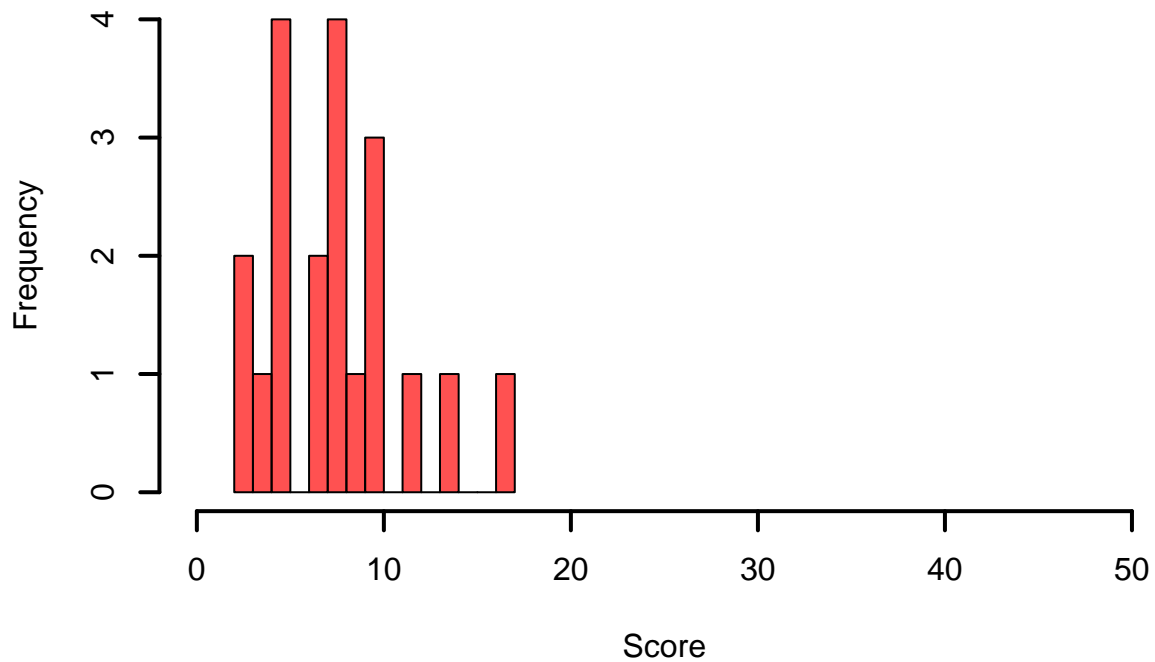
The number of applications is set to 20, and everything is recomputed.

N = 20

```
set.seed(1)
hawk1 = rsnorm(N, mean = 9, sd = 4, xi = 3)
hawk = data.frame(Raw_Hawk = as.vector(round(hawk1)))
hawk$Raw_Hawk_Sorted = sort(hawk$Raw_Hawk)
hawk_hist = hist(hawk$Raw_Hawk_Sorted, breaks = seq(min(hawk$Raw_Hawk)-1,
  max(hawk$Raw_Hawk)), plot = FALSE)
ymax = max(hawk_hist$counts)

hawk_hist = hist(hawk$Raw_Hawk_Sorted, breaks = seq(min(hawk$Raw_Hawk)-1,
  max(hawk$Raw_Hawk), by=1), col=rgb(255,0,0,a, maxColorValue = 255), lwd=2,
  xlim=range(0,50), ylim=c(0,ymax), main="Hawk Scores - Raw", xlab="Score",
  ylab="Frequency")
```

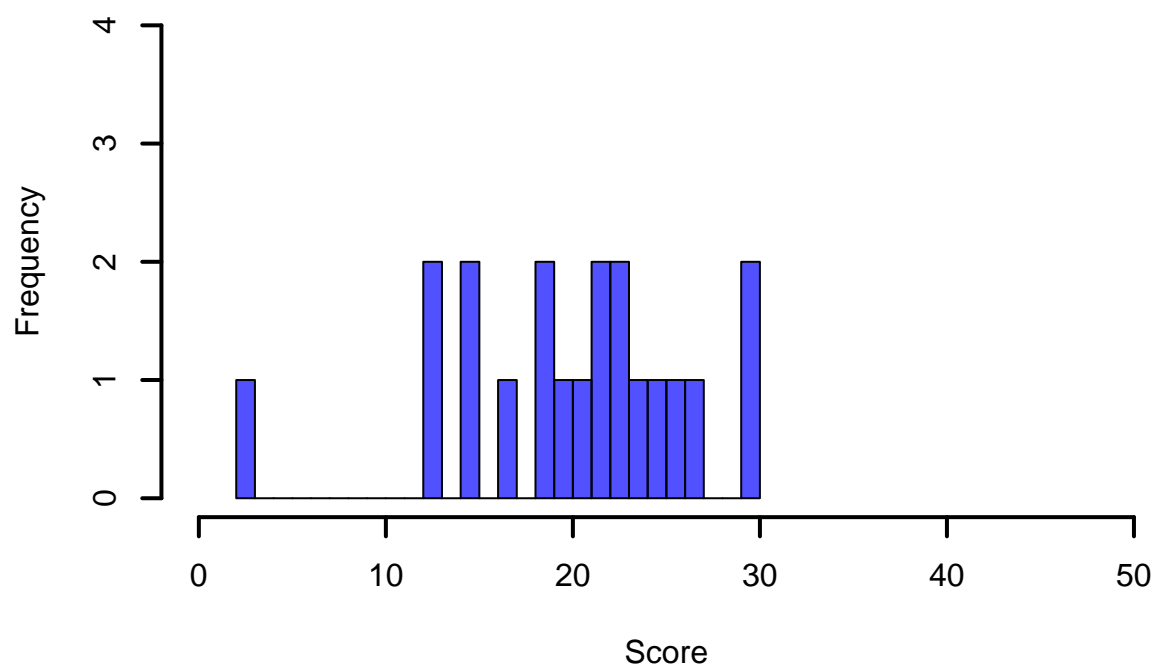
## Hawk Scores – Raw



```
set.seed(1)
jane1 = rnorm(N, 19, 7)
jane = data.frame(Raw_Jane = as.vector(round(jane1)))
jane$Raw_Jane_Sorted = sort(jane$Raw_Jane)

jane_hist = hist(jane$Raw_Jane_Sorted, breaks = seq(min(jane$Raw_Jane)-1,
  max(jane$Raw_Jane), by=1), col=rgb(0,0,255,a, maxColorValue = 255), lwd=2,
  xlim=range(0,50), ylim=c(0, ymax), main="Jane Scores - Raw", xlab="Score",
  ylab="Frequency")
```

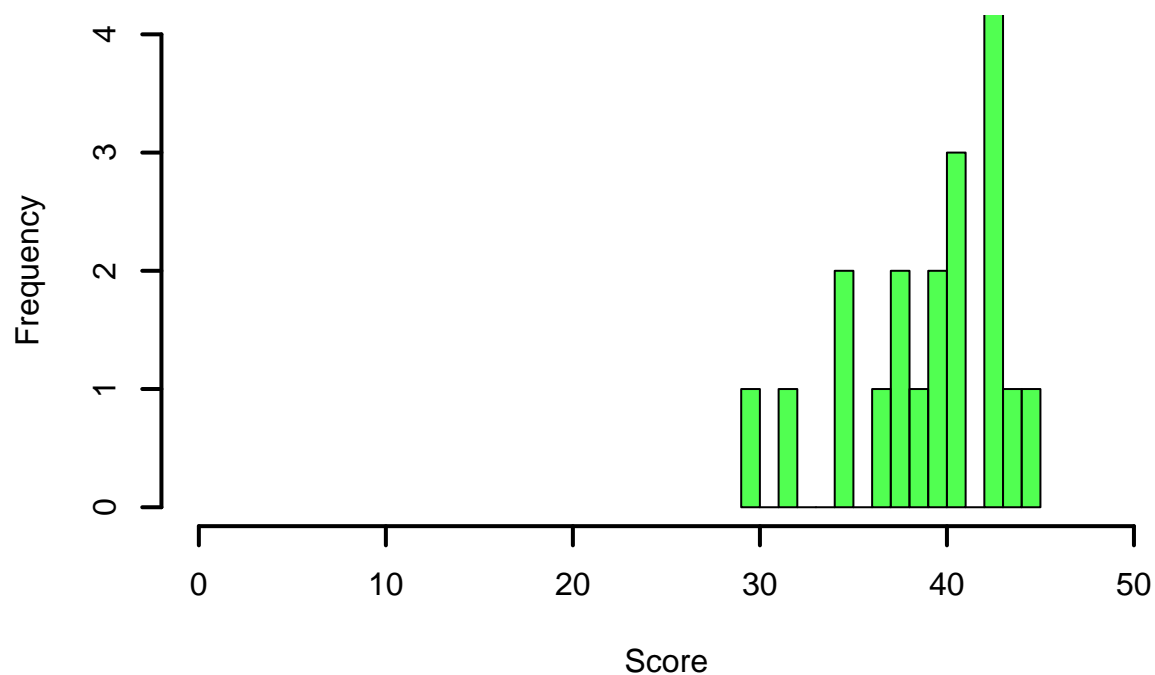
## Jane Scores – Raw



```
set.seed(1)
dove1 = rsnorm(N, mean = 40, sd = 4, xi = 0.5)
dove = data.frame(Raw_Dove = as.vector(round(dove1)))
dove$Raw_Dove_Sorted = sort(dove$Raw_Dove)

dove_hist = hist(dove$Raw_Dove_Sorted, breaks = seq(min(dove$Raw_Dove)-1,
  max(dove$Raw_Dove), by=1), col=rgb(0,255,0,a, maxColorValue = 255), lwd=2,
  xlim=range(0,50), ylim=c(0,ymax), main="Dove Scores - Raw", xlab="Score",
  ylab="Frequency")
```

## Dove Scores – Raw



```
mydata = data.frame(App_No = c(1:N),
                     Hawk_raw = hawk$Raw_Hawk_Sorted,
                     Jane_raw = jane$Raw_Jane_Sorted,
                     Dove_raw = dove$Raw_Dove_Sorted
                     )
head(mydata, n=20)
```

##	App_No	Hawk_raw	Jane_raw	Dove_raw
## 1	1	3	3	30
## 2	2	3	13	32
## 3	3	4	13	35
## 4	4	5	15	35
## 5	5	5	15	37
## 6	6	5	17	38
## 7	7	5	19	38
## 8	8	7	19	39
## 9	9	7	20	40
## 10	10	8	21	40
## 11	11	8	22	41
## 12	12	8	22	41
## 13	13	8	23	41
## 14	14	9	23	43
## 15	15	10	24	43
## 16	16	10	25	43
## 17	17	10	26	43

```
## 18      18      12      27      43
## 19      19      14      30      44
## 20      20      17      30      45
```

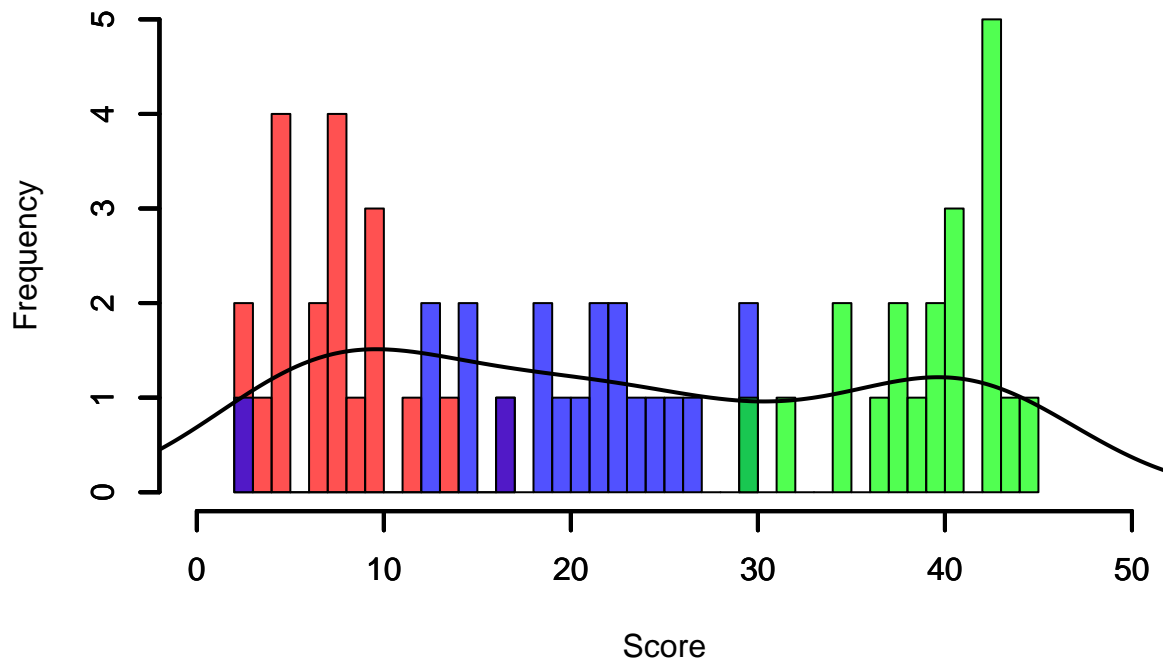
```
ymax = max(c(hawk_hist$counts, jane_hist$counts, dove_hist$counts))

hist(hawk$Raw_Hawk_Sorted, breaks = seq(min(hawk$Raw_Hawk)-1,
                                         max(hawk$Raw_Hawk), by=1),
     col=rgb(255,0,0,a, maxColorValue = 255), lwd=2, xlim=range(0,50),
     ylim=range(0,ymax), main=NULL, xlab=NULL, ylab=NULL)
par(new=TRUE)
hist(jane$Raw_Jane_Sorted, breaks = seq(min(jane$Raw_Jane)-1,
                                         max(jane$Raw_Jane), by=1),
     col=rgb(0,0,255,a, maxColorValue = 255), lwd=2, xlim=range(0,50),
     ylim=range(0,ymax), main=NULL, xlab=NULL, ylab=NULL)
par(new=TRUE)
hist(dove$Raw_Dove_Sorted, breaks = seq(min(dove$Raw_Dove)-1,
                                         max(dove$Raw_Dove), by=1),
     col=rgb(0,255,0,a, maxColorValue = 255), lwd=2, xlim=range(0,50),
     ylim=range(0,ymax), main="Raw Scores", xlab="Score", ylab="Frequency")

scale_factor = N*3
x = seq(0, 40, length.out = 3*N)
all3 = sort(c(hawk$Raw_Hawk, jane$Raw_Jane, dove$Raw_Dove))

y2 = density(all3)
y5 = scale_factor*y2$y
lines(y2$x, y5, col = "black", lwd = 2)
```

## Raw Scores



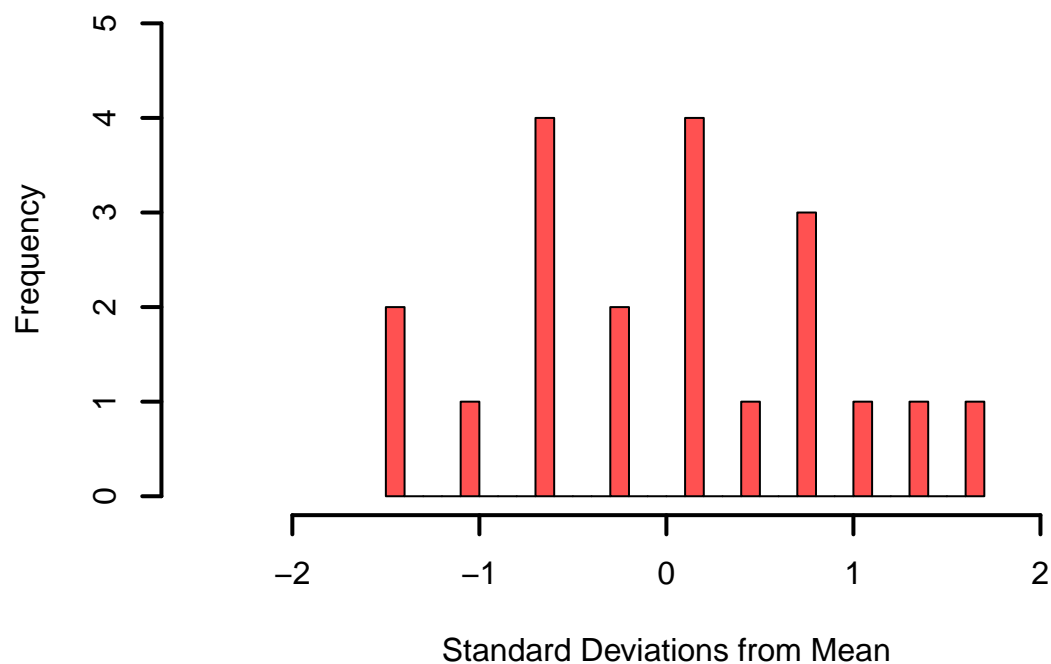
```
hawk$rank = rank(hawk$Raw_Hawk)
jane$rank = rank(jane$Raw_Jane)
dove$rank = rank(dove$Raw_Dove)

hawk$normal = qnorm(hawk$rank/(1+N))
jane$normal = qnorm(jane$rank/(1+N))
dove$normal = qnorm(dove$rank/(1+N))

nbreak = max(cbind(length(seq(min(hawk$normal), max(hawk$normal), by=0.1)),
                      length(seq(min(jane$normal), max(jane$normal), by=0.1)),
                      length(seq(min(dove$normal), max(dove$normal), by=0.1))))
```

```
hawk_hist2 = hist(hawk$normal, breaks = nbreak,
                  col=rgb(255,0,0,a, maxColorValue = 255), lwd=2,
                  xlim=range(-2.5,2.5), ylim=c(0,ymax), main="Standardized Hawk Scores",
                  xlab="Standard Deviations from Mean",
                  ylab="Frequency")
```

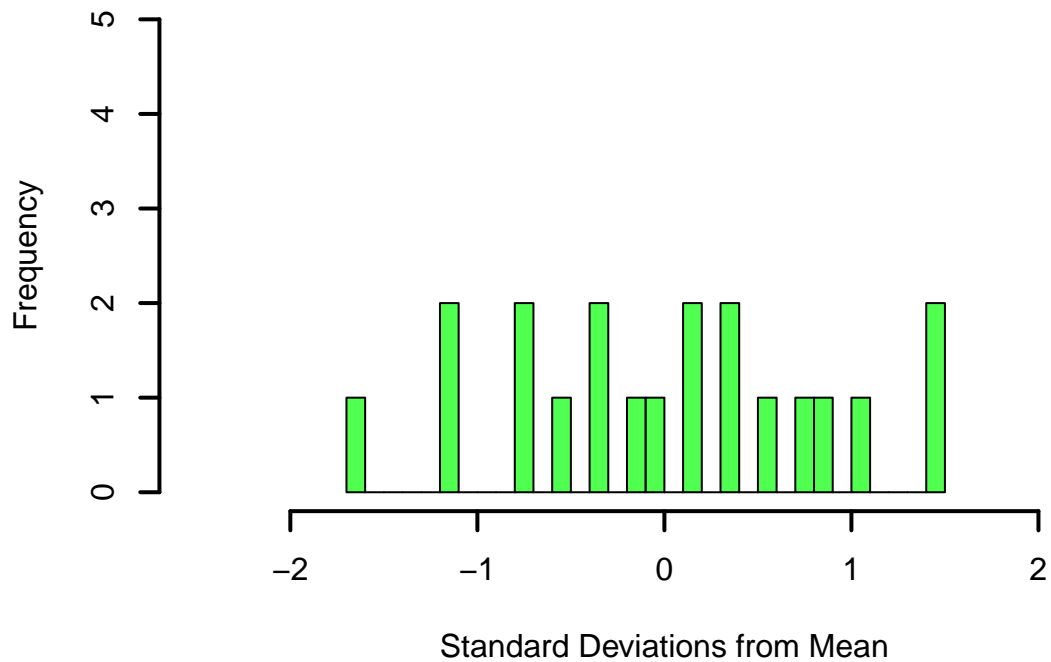
## Standardized Hawk Scores



```
jane_hist2 = hist(jane$normal, breaks = nbreak,  
  col=rgb(0,255,0,a, maxColorValue = 255), lwd=2,  
  xlim=range(-2.5,2.5), ylim=c(0,ymax), main="Standardized Jane Scores",  
  xlab="Standard Deviations from Mean",  
  ylab="Frequency")
```

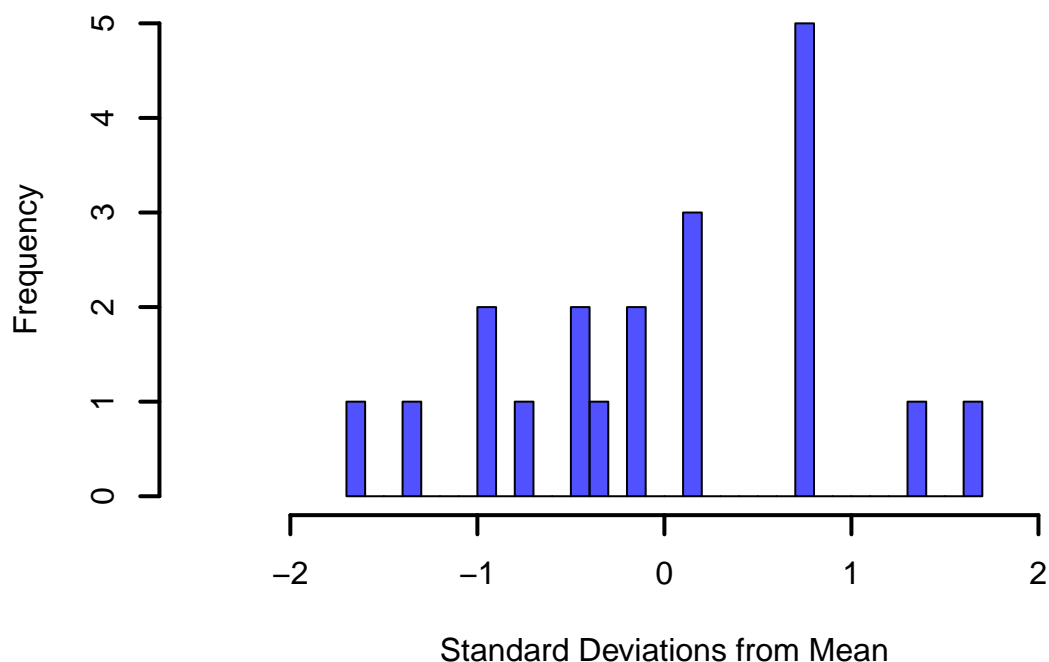


## Standardized Jane Scores



```
dove_hist2 = hist(dove$normal, breaks = nbreak,
  col=rgb(0,0,255,a, maxColorValue = 255), lwd=2,
  xlim=range(-2.5,2.5), ylim=c(0,ymax), main="Standardized Dove Scores",
  xlab="Standard Deviations from Mean",
  ylab="Frequency")
```

## Standardized Dove Scores



```
hawk_hist2 = hist(hawk$normal, breaks = nbreak,
  col=rgb(255,0,0,a, maxColorValue = 255), lwd=2,
xlim=range(-2.5,2.5), ylim=c(0,ymax), main="All Standardized Scores",
xlab="Standard Deviations from Mean",
ylab="Frequency")

par(new=TRUE)

jane_hist2 = hist(jane$normal, breaks = nbreak,
  col=rgb(0,255,0,a, maxColorValue = 255), lwd=2,
xlim=range(-2.5,2.5), ylim=c(0,ymax), main=NULL, xlab=NULL, ylab=NULL)

par(new=TRUE)

dove_hist2 = hist(dove$normal, breaks = nbreak,
  col=rgb(0,0,255,a, maxColorValue = 255), lwd=2,
xlim=range(-2.5,2.5), ylim=c(0,ymax), main=NULL, xlab=NULL, ylab=NULL)

scale_factor2 = N*0.3
all13_2 = sort(c(hawk$normal, jane$normal, dove$normal))
y2 = density(all13_2)
y_scaled2 = scale_factor2*y2$y
lines(y2$x, y_scaled2, col = "black", lwd = 2)
```

### All Standardized Scores

