STAT 210
Applied Statistics and Data Analysis
Comparing Two Populations

Joaquin Ortega

Fall 2020

Example 1

Example 1

In this example, we want to compare two experimental fish groups: the groups were fed with different diets, and we want to compare the results.

The outcome we measure is a composite measure of weight and length.

We read the data of the two samples into `R` and set each in vectors (matrix/table of one column).

To be able to read the data, we need to have the data sets in the working directory, or else give the path to the data files in the `scan` command below.

To know which is our working directory, we use the function `getwd()`. To change the working directory, use `setwd()`.

Example 1

```r
getwd()
```

```
## [1] "/Users/ortegaj/Stat210/2-ComparingPopulations"
```

```r
fish1 = matrix(scan("data/diet1.1"), ncol=1, byrow=T)
fish2 = matrix(scan("data/diet1.2"), ncol=1, byrow=T)
```

Compute the mean (average) of each of the samples:

```r
mean(fish1)
mean(fish2)
```

```
## [1] 49.63333
## [1] 49.05
```

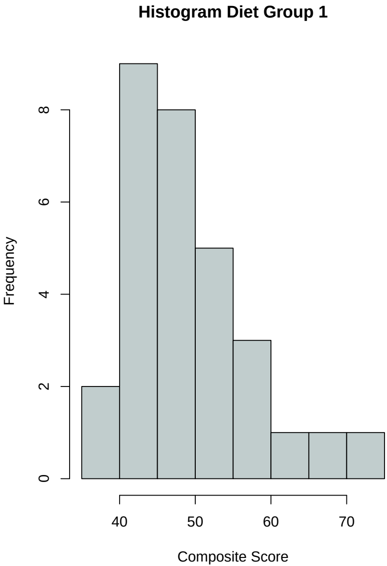
The means are almost equal, but means do not tell the whole story.

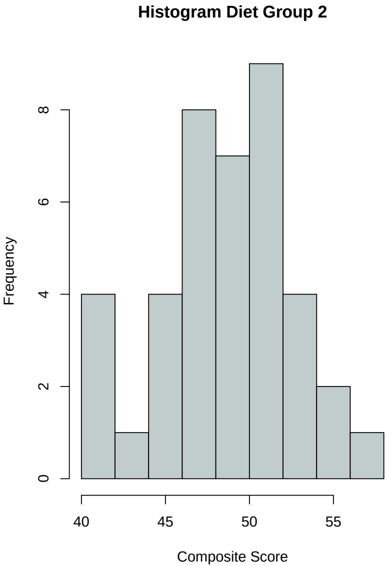
Histograms

To make a histogram, we divide the range of values into ‘bins’, which are usually all the same size. Then we count how many observations fall in each bin and draw a bar graph with this data.

Histograms give an idea of the distribution of the data, but their appearance depends on the number of bins and the value of the *anchor* (starting point of the division).

Histograms

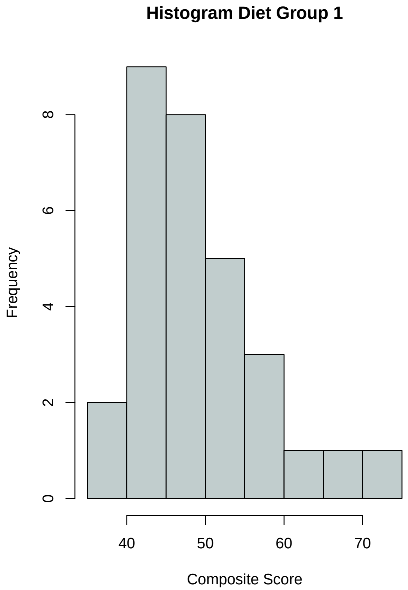

Histogram Diet Group 1

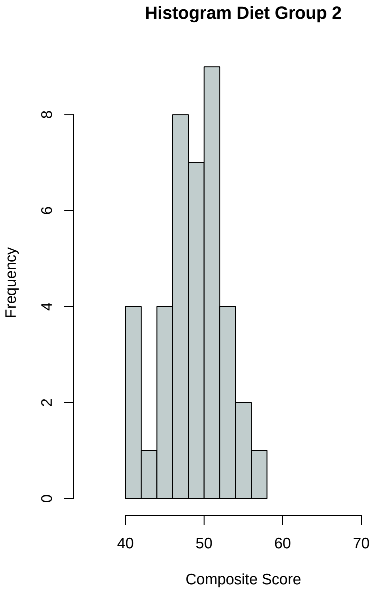

Histogram Diet Group 2

Histograms

A problem with the previous graph, if we want to compare the two samples is that the ranges in the x axes are not the same. This gives the wrong idea about their spread.

Histograms


Histogram Diet Group 1
Frequency
0
2
4
6
8
40
50
60
70
Composite Score


Histogram Diet Group 2
Frequency
0
2
4
6
8
40
50
60
70
Composite Score

Histograms

To further improve the comparison, we plot the graphs in a single column.

Here we compute the limit references for the *x* axis

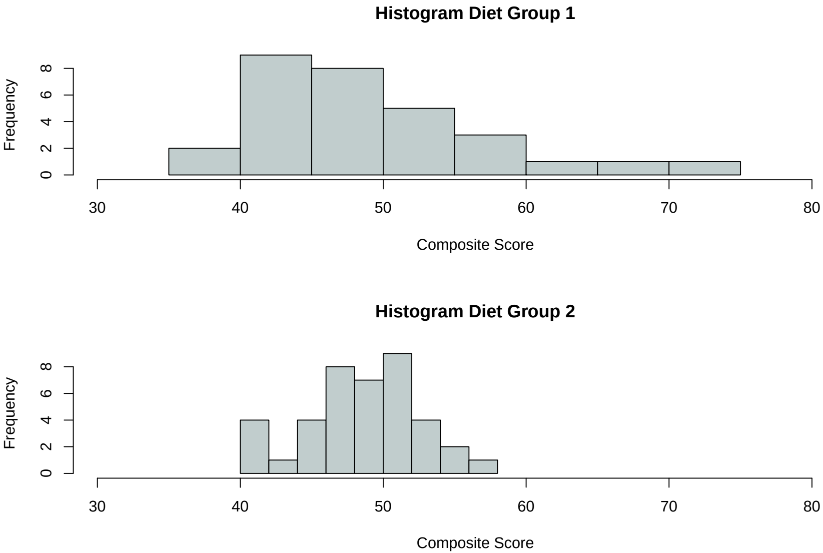
```
(minval = 0.80*min(fish1, fish2))
```

```
## [1] 30.4
```

```
(maxval = 1.10*max(fish1, fish2))
```

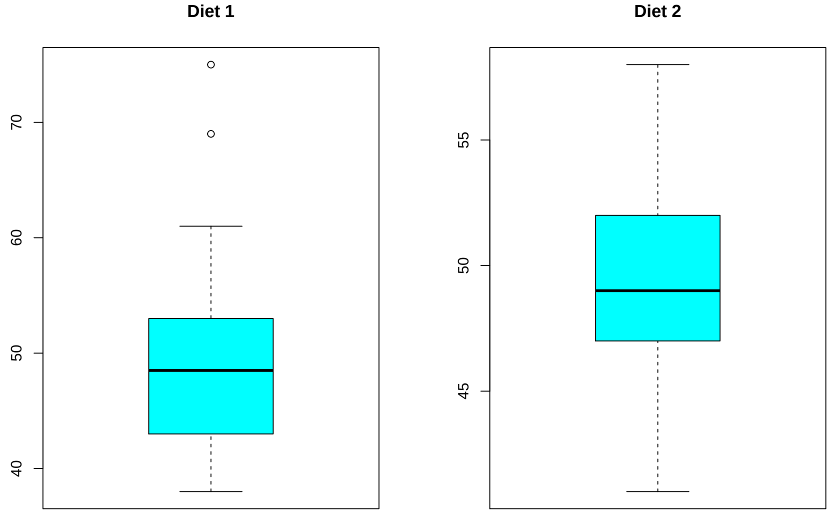
```
## [1] 82.5
```

Histograms


Histogram Diet Group 1
Frequency
0
2
4
6
8
30
40
50
60
70
80
Composite Score
Histogram Diet Group 2
Frequency
0
2
4
6
8
30
40
50
60
70
80
Composite Score

Boxplots

Another way to compare distributions is the use of boxplots.



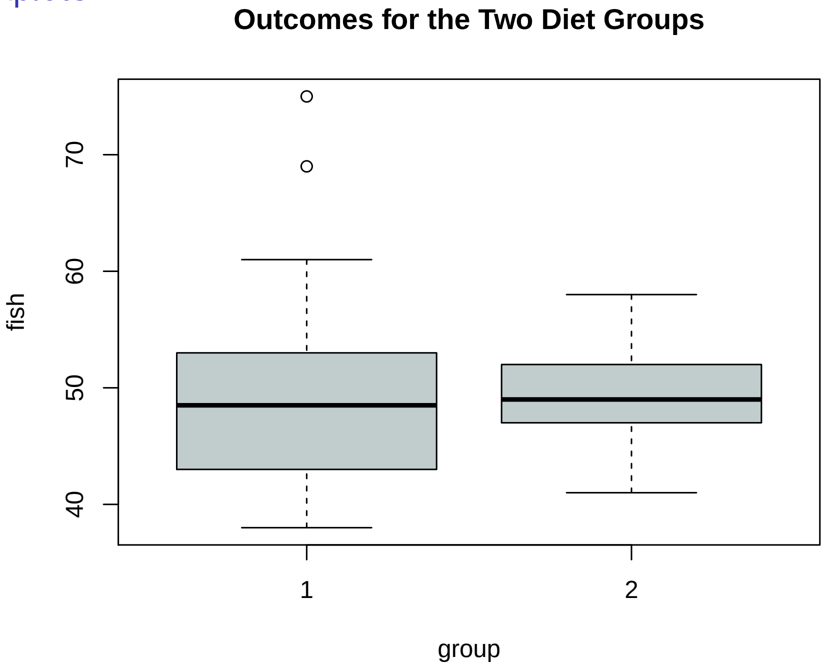
Boxplots

In this case, the *y* scales are different, and we cannot compare the populations.

Display both populations in one graph.

```r
par(mfrow=c(1,1))
fish = c(fish1, fish2)
n1 = length(fish1); n2 = length(fish2)
group = c(rep(1, n1), rep(2, n2))
fish.df <- data.frame(fish=fish, group = group)
boxplot(fish ~ group, data = fish.df,
        main = "Outcomes for the Two Diet Groups",
        col = 'azure3')
```

Boxplots



Outliers

Outliers or atypical values are data points that differ significantly from the other points in the sample.

Outliers can occur by chance, but they frequently indicate either measurement error or a heavy-tailed distribution.

An outlier can cause problems in statistical calculations and usually deserve careful consideration.

Robust statistical methods are useful to mitigate the effect of outliers in statistical analyses.

In the following examples, we consider the effect of (artificial) outliers in the analysis of our samples.

Outliers

Order the sample from the minimum to the maximum

```r
fish1s = sort(fish1); fish2s = sort(fish2)
```

The means do not change

```r
mean(fish1s); mean(fish2s)
```

```
## [1] 49.63333
## [1] 49.05
```

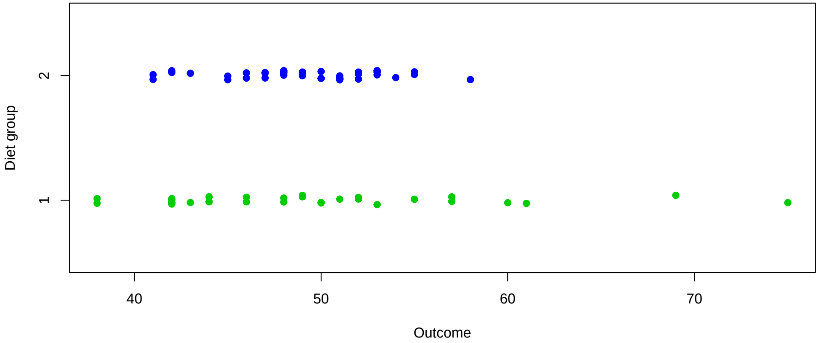
Here we observe the data

```
##  [1] 38 38 42 42 42 42 42 43 44 44 44 46 46 48 48
## [16] 49 49 50 50 51 52 52 53 55 57 57 60 61 69 75
```

```
##  [1] 41 41 42 42 43 45 45 46 46 47 47 47 48 48 48
## [16] 48 48 49 49 49 49 50 50 50 51 51 51 51 51 52
## [31] 52 52 52 53 53 53 54 55 55 58
```

Outliers

```r
with(fish.df, plot(fish, jitter(group, factor=0.2),
     xlab='Outcome', ylab='Diet group', ylim = c(0.5,2.5),
     pch=19, col = group +2, yaxp = c(1,2,1)))
```



Outliers

```
fish1x = fish1s
fish2x = fish2s
fish2x[n2] = 1000  # artificial outlier set at
    # the end of the second sample
```

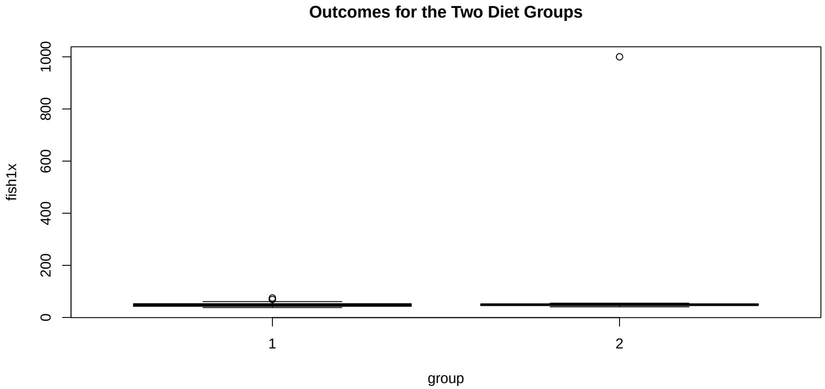
```
mean(fish1x)
mean(fish2x)
```

```
## [1] 49.63333
## [1] 72.6
```

The mean for the second sample has changed considerably.

Outliers

```
par(mfrow=c(1,1))
fish1x = c(fish1x, fish2x)
boxplot(fish1x ~ group, main="Outcomes for the Two Diet Groups")
```



Example 2

Example 2

In this example, we again have two experimental groups that correspond to two methods for improving reading comprehension. The outcome is the score on a reading test.

We have scores before and after the methods are applied.

Read the data and put it into 2 matrices of 2 columns each

```
group1 = matrix(scan("data/comp.1"), ncol=2, byrow=T)
group2 = matrix(scan("data/comp.2"), ncol=2, byrow=T)
```

Build up vectors for each variable

```
group1.pre = group1[,1];
group1.post = group1[,2];
group2.pre = group2[,1];
group2.post = group2[,2]
```

Example 2

Examine the data

```r
sort(group1.post);sort(group2.post)
```

```
##   [1] 64 65 67 69 70 71 71 73 73 73 74 74 74 74 74 75 75 75 75 75 75
##  [22] 76 76 76 76 77 77 77 77 77 77 77 77 77 77 78 78 78 78 78 79 79
##  [43] 79 79 79 79 79 80 80 80 80 80 81 81 81 81 81 81 81 81 81 81 82
##  [64] 82 82 82 83 83 83 83 83 83 84 84 84 84 84 84 85 85 85 86 86 86
##  [85] 86 87 87 87 87 87 87 88 88 89 89 90 91 91 91 94
```

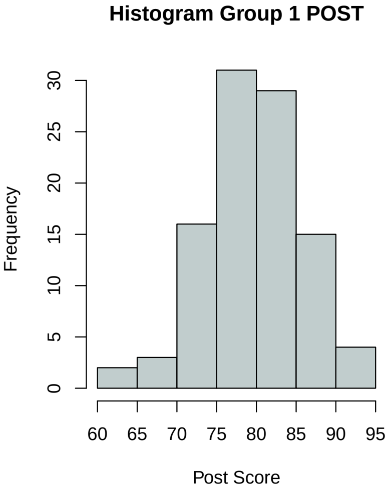
```
##   [1] 67 68 69 70 71 71 72 73 73 73 74 74 74 74 74 74 75 75 75 75 75
##  [22] 75 75 75 76 76 76 76 76 77 77 77 77 77 77 77 78 78 78 78 78 79
##  [43] 79 79 79 79 79 79 79 79 79 79 79 79 79 79 80 80 80 80 80 80 80
##  [64] 80 81 81 81 81 82 82 82 82 82 82 82 83 83 83 83 83 83 83 83 83
##  [85] 83 84 84 84 85 85 85 85 86 86 87 87 87 88 88 90
```

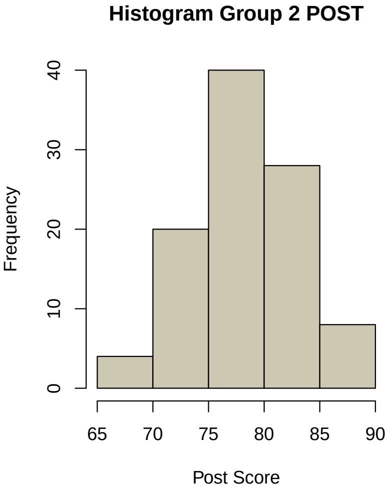
```r
mean(group1.post);mean(group2.post)
```

```
## [1] 80.05
```

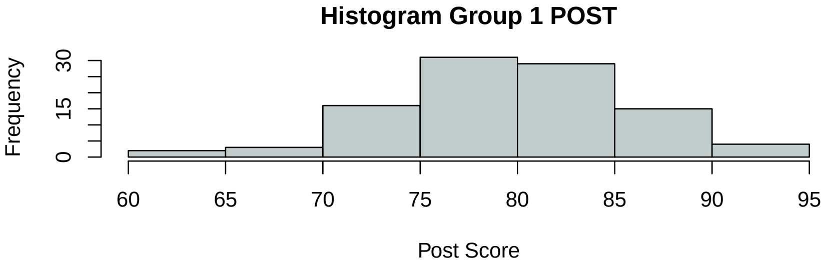
```
## [1] 79.04
```

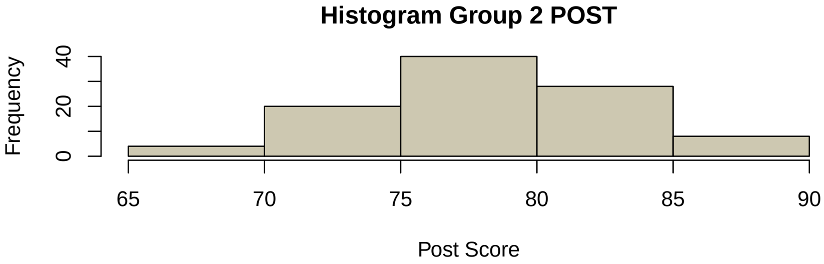
Example 2


Histogram Group 1 POST
Frequency
0
5
10
15
20
25
30
60
65
70
75
80
85
90
95
Post Score


Histogram Group 2 POST
Frequency
0
10
20
30
40
65
70
75
80
85
90
Post Score

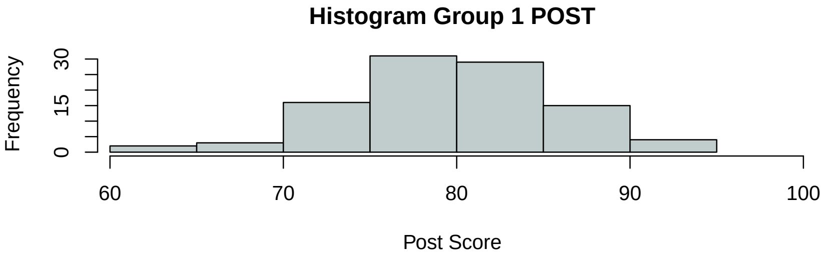
Example 2

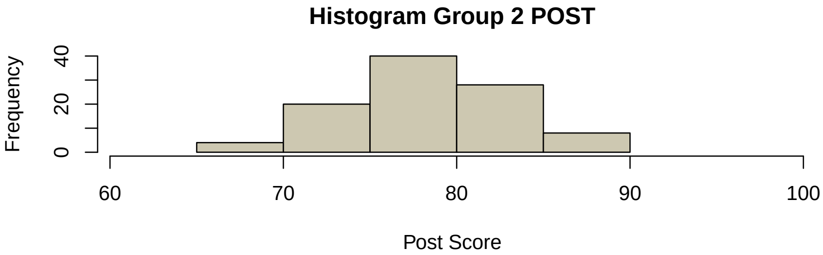

Histogram Group 1 POST
Frequency
0
15
30
60
65
70
75
80
85
90
95
Post Score


Histogram Group 2 POST
Frequency
0
20
40
65
70
75
80
85
90
Post Score

Example 2

```
minval=0.95*min(group1.post,group2.post)
maxval=1.05*max(group1.post,group2.post)
```





Example 2

Numerical Summaries

```
summary(group1.post)
summary(group2.post)
```

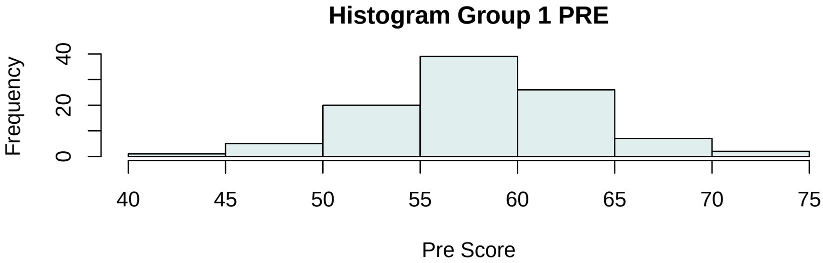
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   64.00   76.75   80.00   80.05   84.00   94.00
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   67.00   76.00   79.00   79.04   82.25   90.00
```

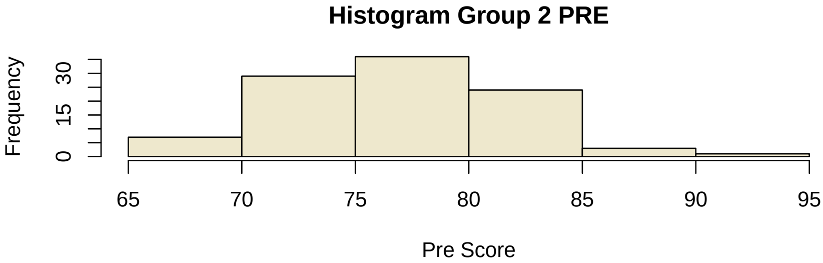
Example 2

- Do we conclude that the post scores of the two methods are similar?
- Can we conclude that the effects of the two methods are the same?

To measure the impact of a method, we also need to look at the baseline

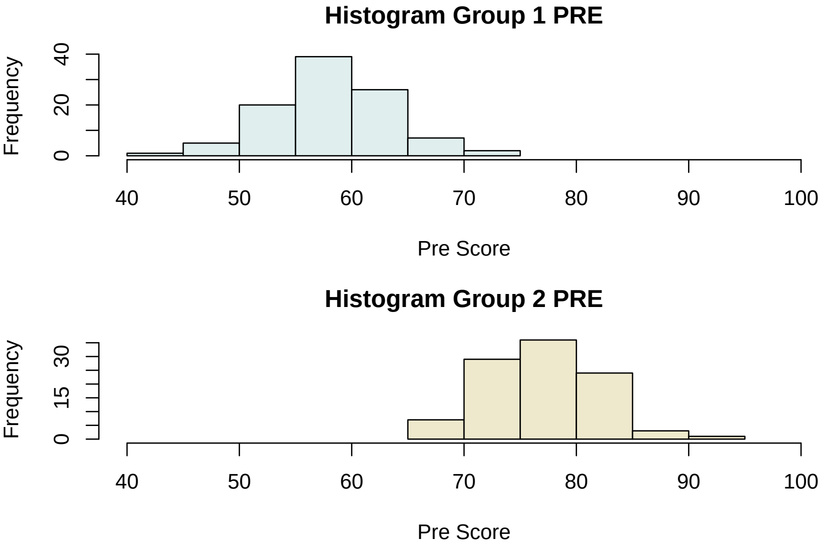
Example 2





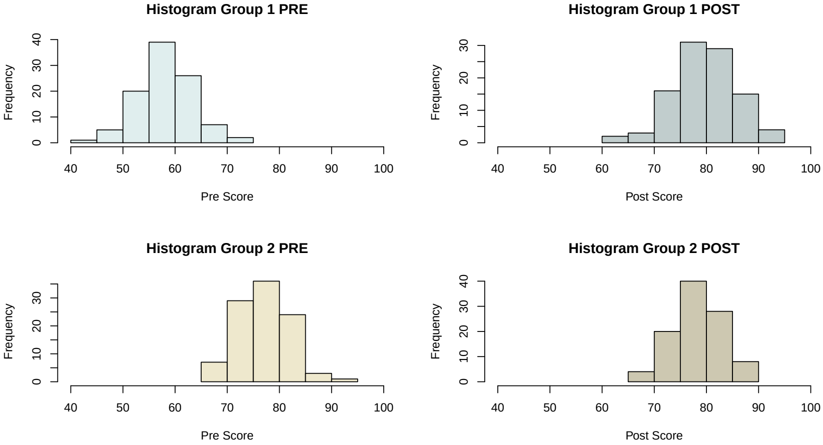
This is a misleading graph!

Example 2



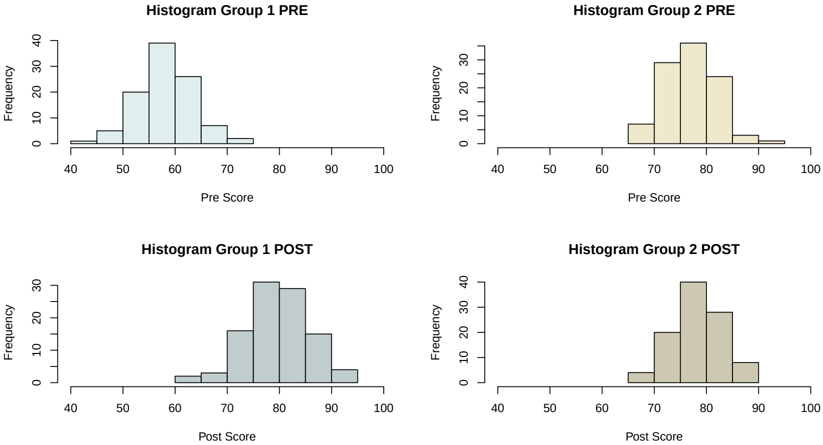
Histogram Group 1 PRE
Frequency
0
20
40
40
50
60
70
80
90
100
Pre Score
Histogram Group 2 PRE
Frequency
0
15
30
40
50
60
70
80
90
100
Pre Score

Example 2


Histogram Group 1 PRE
Frequency
0
10
20
30
40
40
50
60
70
80
90
100
Pre Score
Histogram Group 1 POST
Frequency
0
10
20
30
40
50
60
70
80
90
100
Post Score
Histogram Group 2 PRE
Frequency
0
10
20
30
40
50
60
70
80
90
100
Pre Score
Histogram Group 2 POST
Frequency
0
10
20
30
40
40
50
60
70
80
90
100
Post Score

Example 2

Another perspective:



Example 2

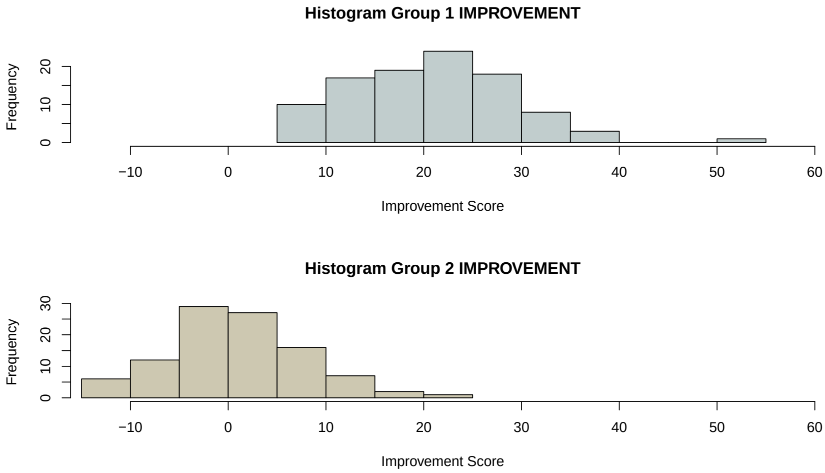
Is post-score the proper way to measure the impact of the methods?

Better way is improvement

```
imp1 = group1.post - group1.pre
imp2 = group2.post - group2.pre
cbind(imp1, imp2)
```

```
##       imp1 imp2
##  [1,]   17    1
##  [2,]   28   -5
##  [3,]   18    1
##  [4,]   24    2
##  [5,]   30    0
##  [6,]   11    0
##  [7,]   18    7
##  [8,]   17    9
##  [9,]   17    4
## [10,]   26  -11
## [11,]    7    9
## [12,]   29    0
## [13,]   10   -5
## [14,]   27   12
## [15,]   18   -3
## [16,]   32    8
## [17,]   24   11
## [18,]   27    2
```

Example 2


Histogram Group 1 IMPROVEMENT
Frequency
0
10
20
−10
0
10
20
30
40
50
60
Improvement Score
Histogram Group 2 IMPROVEMENT
Frequency
0
10
20
30
−10
0
10
20
30
40
50
60
Improvement Score

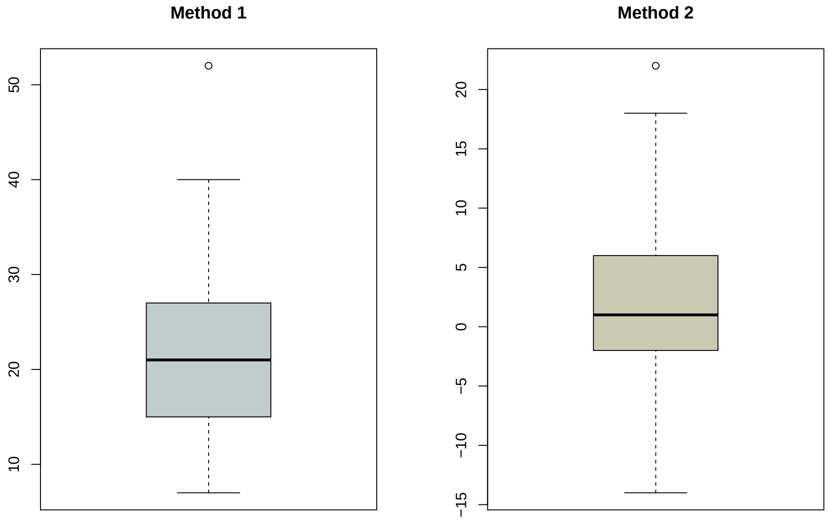
Example 2

Summary statistics

```r
summary(imp1)
summary(imp2)
```

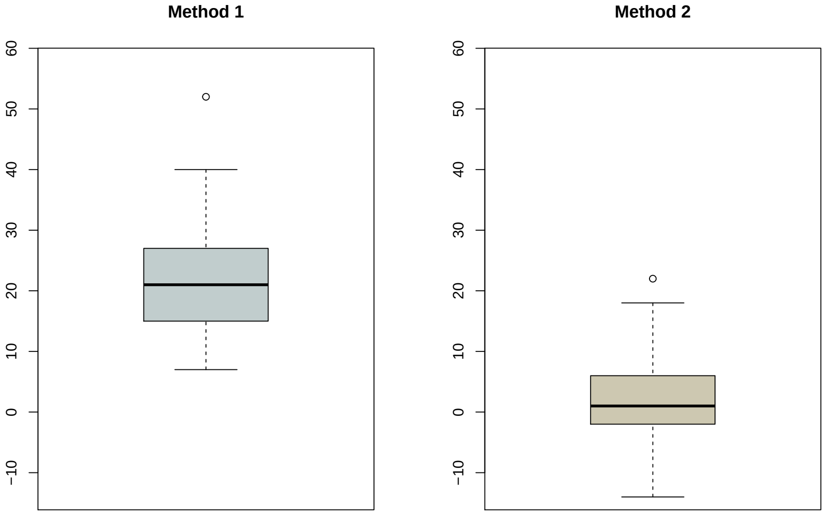
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    7.00   15.00   21.00   21.35   27.00   52.00
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -14.00   -2.00    1.00    1.61    6.00   22.00
```

Example 2



This is also a misleading graph!

Example 2



Example 3

Example 3

Understanding uncertainty in statistical results arising from random sampling

Every time we draw a sample from a population, the result can be different.

Even if the samples come from populations with the same distribution, there will be differences between them.

We explore these differences with a simulation study. We look at the difference between sample means.

From what we have studied about the sampling distribution, we expect these differences to depend on the size of the sample

Example 3

We don't have access to the entire (hypothetical) population. We only sample the population (have a small subset of the population)

```
n1 = n2 = 5
mu1 = mu2 = 60
sd1 = sd2 = 5
samp1 = rnorm(n1, mu1, sd1)
samp2 = rnorm(n2, mu2, sd2)
mean(samp1)
```

```
## [1] 62.46298
```

```
mean(samp2)
```

```
## [1] 59.30772
```

```
mean(samp1) - mean(samp2)
```

```
## [1] 3.155266
```

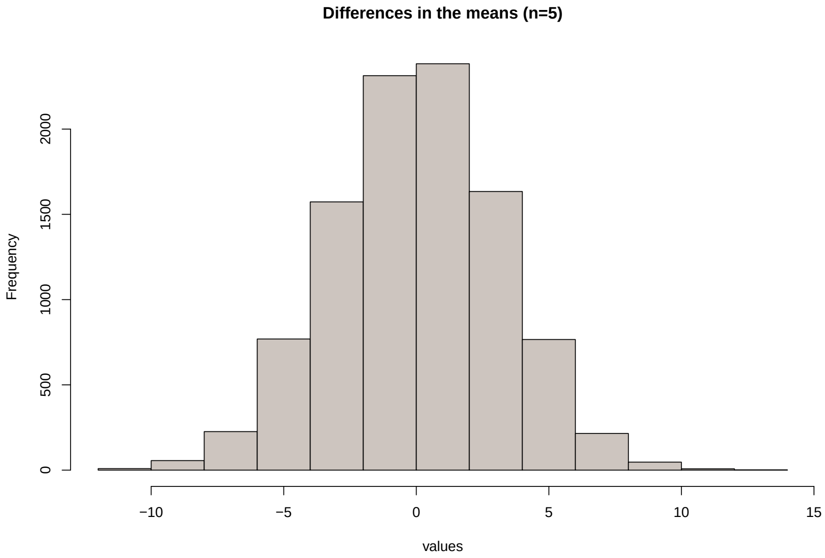
Example 3

To illustrate the sample behavior of the difference in the sample means we will repeat this simulation 10,000 times and plot the results.

The sample size is 5.

```r
n1 = n2 = 5
B = 10000
samp1.mat <- matrix(rnorm(n1*B,mu1,sd1), ncol = n1)
samp2.mat <- matrix(rnorm(n1*B,mu1,sd1), ncol = n2)
mean1.vec <- apply(samp1.mat, 1, mean)
mean2.vec <- apply(samp2.mat, 1, mean)
diff.mean.5 <- mean1.vec - mean2.vec
```

Example 3


Differences in the means (n=5)
Frequency
0
500
1000
1500
2000
−10
−5
0
5
10
15
values

Example 3

```
mean(diff.mean.5); sd(diff.mean.5)
```

```
## [1] 0.007843658
```

```
## [1] 3.166236
```

```
summary(diff.mean.5)
```

```
##       Min.    1st Qu.     Median       Mean    3rd Qu.
## -11.442677  -2.137427   0.055265   0.007844   2.161418
```

```
sort(diff.mean.5)[c(.1*B,.9*B)]
```

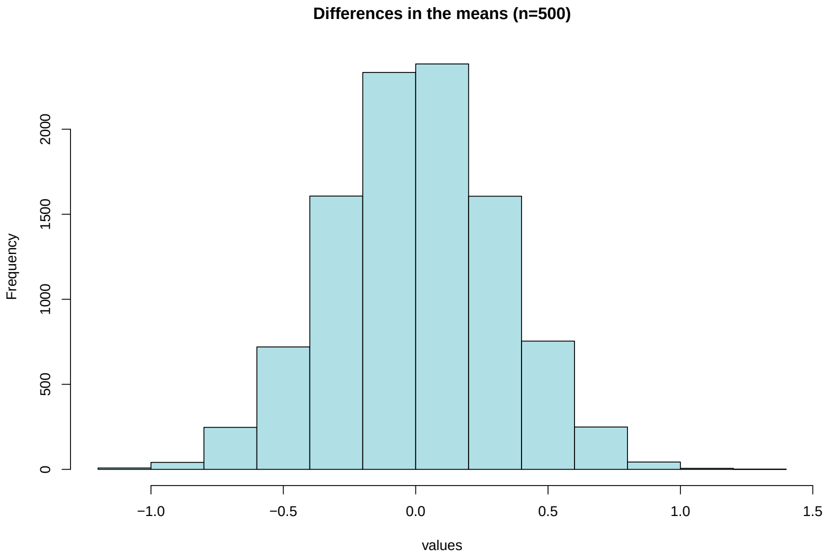
```
## [1] -4.110528  4.069226
```

Example 3

We repeat the simulation with samples of size 500 instead of 5.

```
n1 = n2 = 500
samp1.mat <- matrix(rnorm(n1*B,mu1,sd1), ncol = n1)
samp2.mat <- matrix(rnorm(n1*B,mu1,sd1), ncol = n2)
mean1.vec <- apply(samp1.mat, 1, mean)
mean2.vec <- apply(samp2.mat, 1, mean)
diff.mean.500 <- mean1.vec - mean2.vec
```

Example 3


Differences in the means (n=500)
Frequency
0
500
1000
1500
2000
−1.0
−0.5
0.0
0.5
1.0
1.5
values

Example 3

```
mean(diff.mean.500); sd(diff.mean.500)
```

```
## [1] 0.002059467
```

```
## [1] 0.3160012
```

```
summary(diff.mean.500)
```

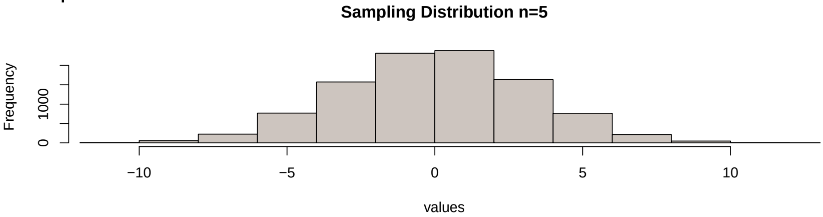
```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -1.160097 -0.213491  0.002368  0.002060  0.216305  1.277359
```

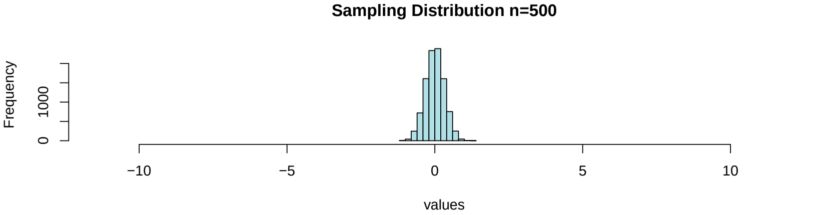
```
sort(diff.mean.500)[c(.1*B,.9*B)]
```

```
## [1] -0.4025577  0.4110778
```

Example 3

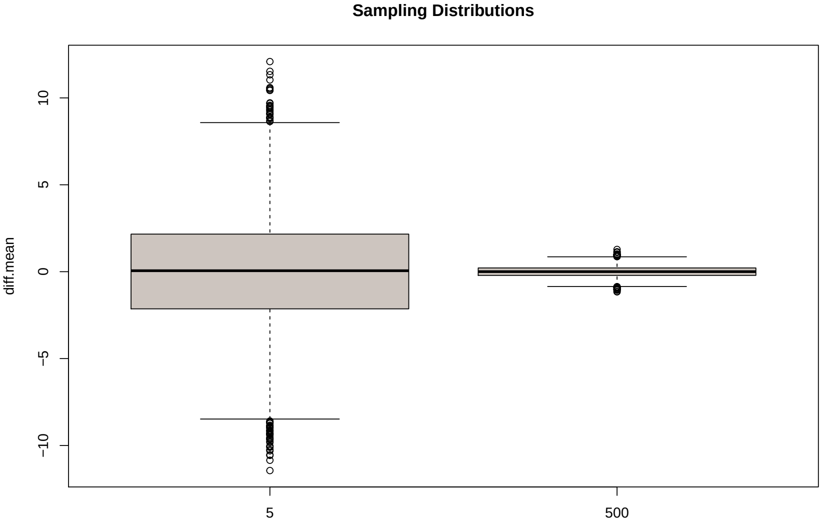
Compare the two results





Example 3

```
diff.mean = c(diff.mean.5, diff.mean.500)
group = c(rep(5, B), rep(500,B))
boxplot(diff.mean~group, col='seashell3', main="Sampling Di
```


Sampling Distributions
diff.mean