Training Intervention Analysis

Cian

October 2018

## Load the required libraries

library(infer)  
library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3 v purrr 0.3.4  
## v tibble 3.0.6 v dplyr 1.0.4  
## v tidyr 1.1.2 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(tolerance)

## Read in the training intervention data and view its structure

train.df <- read.csv("Training\_intervention\_data.csv")  
glimpse(train.df)

## Rows: 18  
## Columns: 5  
## $ ID <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,...  
## $ VO2.max\_Pre <dbl> 66.4, 70.9, 64.9, 68.6, 76.7, 75.6, 78.1, 73.1, 74.4, ...  
## $ VO2.max\_Post <dbl> 67.8, 81.7, 70.1, 73.0, 84.5, 78.4, 80.5, 76.0, 78.7, ...  
## $ Squat\_Pre <int> 120, 120, 130, 130, 110, 130, 140, 120, 140, 100, 140,...  
## $ Squat\_Post <int> 140, 150, 160, 160, 140, 160, 170, 140, 170, 130, 180,...

## Summary statistics for pre and post squat

train.df %>% select(Squat\_Pre, Squat\_Post) %>% summary()

## Squat\_Pre Squat\_Post   
## Min. :100 Min. :130.0   
## 1st Qu.:120 1st Qu.:142.5   
## Median :130 Median :160.0   
## Mean :130 Mean :159.4   
## 3rd Qu.:140 3rd Qu.:170.0   
## Max. :160 Max. :190.0

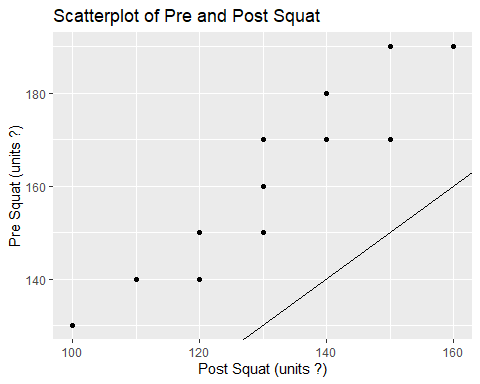
## Mean and standard deviation for pre and post squat

train.df %>% select(Squat\_Pre, Squat\_Post) %>%  
 summarize(Pre\_Mean=mean(Squat\_Pre), Pre\_SD= sd(Squat\_Pre),  
 Post\_Mean=mean(Squat\_Post), Post\_SD= sd(Squat\_Post))

## Pre\_Mean Pre\_SD Post\_Mean Post\_SD  
## 1 130 16.44957 159.4444 18.62074

## Scatterplot of pre and post with line of equality

train.df %>% ggplot(aes(x = Squat\_Pre, y = Squat\_Post)) +  
 geom\_point() +   
 ggtitle("Scatterplot of Pre and Post Squat") +  
 ylab("Pre Squat (units ?)") +  
 xlab("Post Squat (units ?)") +  
 geom\_abline(slope=1, intercept=0)



## Calculate the improvement

train.df <- train.df %>% mutate(Improvement = Squat\_Post-Squat\_Pre) %>%  
 glimpse()

## Rows: 18  
## Columns: 6  
## $ ID <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,...  
## $ VO2.max\_Pre <dbl> 66.4, 70.9, 64.9, 68.6, 76.7, 75.6, 78.1, 73.1, 74.4, ...  
## $ VO2.max\_Post <dbl> 67.8, 81.7, 70.1, 73.0, 84.5, 78.4, 80.5, 76.0, 78.7, ...  
## $ Squat\_Pre <int> 120, 120, 130, 130, 110, 130, 140, 120, 140, 100, 140,...  
## $ Squat\_Post <int> 140, 150, 160, 160, 140, 160, 170, 140, 170, 130, 180,...  
## $ Improvement <int> 20, 30, 30, 30, 30, 30, 30, 20, 30, 30, 40, 40, 20, 30...

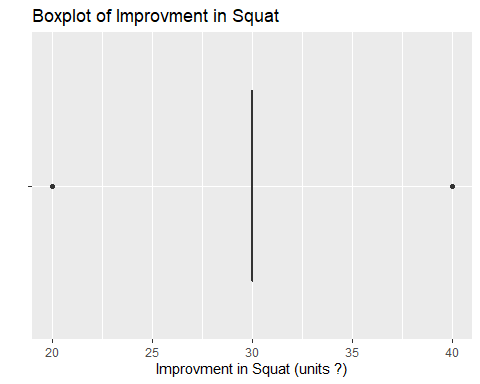
## Mean and standard deviation of improvement

train.df %>% select(Improvement) %>%  
 summarize(Imp\_Mean=mean(Improvement), Imp\_SD= sd(Improvement))

## Imp\_Mean Imp\_SD  
## 1 29.44444 6.391375

## Boxplot to visualize the improvement

train.df %>% ggplot(aes(x = "", y = Improvement)) +  
 geom\_boxplot() +   
 ggtitle("Boxplot of Improvment in Squat") +  
 ylab("Improvment in Squat (units ?)") +  
 xlab("") +  
 coord\_flip()



## T-test to get 95% confidence interval for improvement

train.df %>% select(Improvement) %>% t.test()

##   
## One Sample t-test  
##   
## data: .  
## t = 19.545, df = 17, p-value = 4.356e-13  
## alternative hypothesis: true mean is not equal to 0  
## 95 percent confidence interval:  
## 26.26609 32.62280  
## sample estimates:  
## mean of x   
## 29.44444

## Bootstrap to calculate confidence interval for mean of improvement

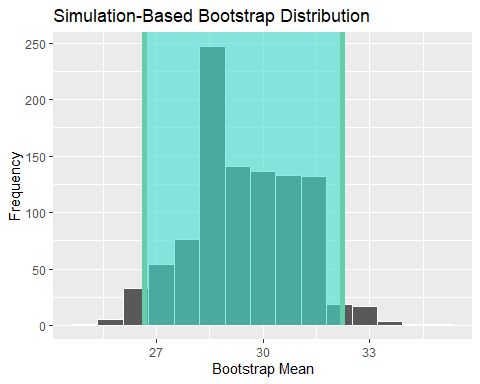
boot <- train.df %>%  
 specify(response = Improvement) %>%  
 generate(reps = 1000, type = "bootstrap") %>%  
 calculate(stat = "mean")  
  
percentile\_ci <- get\_ci(boot)  
round(percentile\_ci,2)

## # A tibble: 1 x 2  
## lower\_ci upper\_ci  
## <dbl> <dbl>  
## 1 26.7 32.2

## Visualize the bootstrap distribution with confidence intervals

boot %>% visualize(endpoints = percentile\_ci, direction = "between") +  
 xlab("Bootstrap Mean") + ylab("Frequency")

## Warning: `visualize()` should no longer be used to plot a confidence interval.  
## Arguments `endpoints`, `endpoints\_color`, and `ci\_fill` are deprecated. Use  
## `shade\_confidence\_interval()` instead.



## Bootstrap to calculate confidence interval for median of improvement

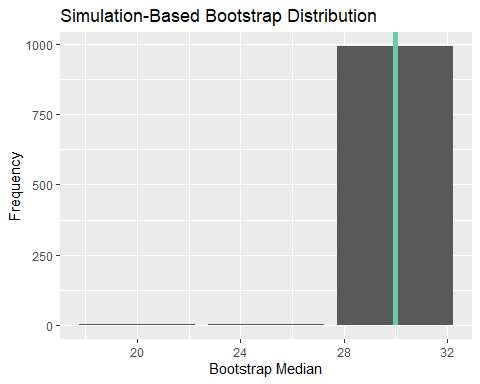
boot.median <- train.df %>%  
 specify(response = Improvement) %>%  
 generate(reps = 1000, type = "bootstrap") %>%  
 calculate(stat = "median")  
  
percentile\_ci\_median <- get\_ci(boot.median)  
round(percentile\_ci\_median,2)

## # A tibble: 1 x 2  
## lower\_ci upper\_ci  
## <dbl> <dbl>  
## 1 30 30

## Visualize the bootstrap distribution with confidence intervals

boot.median %>% visualize(endpoints = percentile\_ci\_median, direction = "between") +  
 xlab("Bootstrap Median") + ylab("Frequency")

## Warning: `visualize()` should no longer be used to plot a confidence interval.  
## Arguments `endpoints`, `endpoints\_color`, and `ci\_fill` are deprecated. Use  
## `shade\_confidence\_interval()` instead.



## Calculate the 95% tolerance interval for improvement

normtol.int(train.df$Improvement, alpha = 0.05, P = 0.95)

## alpha P x.bar 1-sided.lower 1-sided.upper  
## 1 0.05 0.95 29.44444 13.76674 45.12215

## Conclusion

In conclusion the data definitely shows an improvement in squats over the 10 weeks. The variables Pre\_Mean and Post\_Mean clearly show that the average after the 10 weeks was much higher than before. The scatterplot with line of equality clearly display this. The variable Improvement then displays the improvement for each individual. The mean improvement is 29.4 with the standard deviation being 6.4. The 95% confidence interval indicates that improvement lies between 26.27 and 32.62. The 95% bootstrap CI for the mean displayed very similar values of 26.67 and 32.22.