Data Integrity: A Study

Joshua Bennett *Washington State University*

In this article we discuss a Data Set to try to expose data that isn't in line with the norm, and extrapolate the possible reasons for the data inaccuracies.

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

December 13, 2020

1 Introduction

Data Integrity is a critical part of Data Analysis. It is what ensures recoverability and searchability, as well as traceability and connectivity. Protecting the validity and accuracy of the data also increases stability and performance while improving reusability and maintainability. Data Integrity can be compromised in several different ways. Physical compromise of the device, bugs/viruses, transfer errors(unintended alterations to the data), and Human error(malicious or not). This report will be looking at mainly the last two categories of different ways it can compromised. Transfer errors, and Human error.

This is an example of outliers in the data[1]

2 Methods of finding issues in the Data

There are many different methods of looking for Data Integrity issues. This report will focus on two different methods for discovering outliers in the data set. First, we will use the Virtuvian Man data, which was collected from over 60,000 men and over 1000 Women. Using this data we can compare the averages that were found from using that data to compare to the Measure data set. It will also take into account extreme examples, such as the NBA and compare that data against the given data set.

3 Research Question: What is my primary question

My primary research question is are there outliers in the data that might not have Data Integrity?

3.1 What is my secondary question

Given the Vitruvian man data, do the arm spans and height correlations differ enough to show possible issues?

3.2 What is my other secondary question

Are there discrepancies on both sides?

4 Data Description

This data was collected by hand by around 30 students. It was taken from the family and friends of the students in the 419 Multivariate Statistics course at Washington State University. The data was recorded in early September during the Covid-19 Pandemic and because of this was harder than normal to acquire reliable data. The data gathered is focusing on bodily proportions but other data was taken as well, such as eye color and age. This data was chosen due to the ability for it to be taken easily and efficiently by the student in the class. The method of Acquiring the data was through a handout that was created by the students and could be given to a subject to fill out with descriptions of their proportions. It was then put into an excel document and transferred to a .txt file to be accessed and analyzed in RStudio.

Very brief introduction to the data, how it was collected, and so on. Remember that everything is covered (who, what, when, where, why, how, so what, and so on). Reference the section in the Appendix with greater detail about the data provenance. This section should be about two paragraphs, and the Appendix should have more information.

4.1 Summary of Sample

This is a summary of the Data Set that was used for this analysis[2]

4.2 Summary Statistics of Data

Using the Viruvian man Data, We created a Proportion Data Frame that we used to convert the original Measure Data to give us a general example of what a persons proportions should be. This is the covariate being analyzed divided by the height of the person. This gives us a number that we can compare to the Viruvian man data and if they are off by a significant margin we know there is a discrepancy. First we took the proportion data and created a plot of the height and headheight categories. Showing us the original anomalies, then created another plot that showed the arm span and height categories.

A correlation Table was then created using someone suspected of having fraudulent data and comparing it to someone that has data that closely matches the Viruvian man data. This gave us a SD of .07 for the suspect data and a SD of .02 for the matching data. As well as a mean of 1.0 for the matching data and a 1.2 for the suspect data. There was also a .64 significance for the suspect data.

The second Correlation Table was then created using a different sample of possible fraudulent data and compared it to the same example we compared during the first Correlation table. The results for this were 1.0 for the mean of the matching data and a .8 mean for the suspect data. the Standard Deviation was .02 for for the correct data and a Standard Deviation of .29 for the suspect data. with a -.37 significance for the possible fraudulent data.

Two Welch Sample T-Tests were then done. The first one was the suspected arm span fraudulent data used in the first correlation table against the whole Data Set which gave us t = -8.5806, df = 10.062, p-value = 6.086e-06 with a 95 percent confidence interval: -0.0145928 0.4023104.

The other T-test was the other sample that was found whose data didn't match the Vivruvian man data for arm span. These were the opposite end of the spectrum though, much too small compared to the body. the values returned from the T-test were t = 2.0355, df = 9.0791, p-value = 0.07202 with a 95 percent confidence interval: -0.02054352 0.39448094.

5 Key Findings

Using the methods described in the Summary Statistics of Data section, we can see that although there is a small sample size the P test for the Suspect(long) arm span data is well below 5%. Using this we can reject the Null Hypothesis. This makes the test extremely credible. There was something wrong about this data. If you multiply these values by 2.54 you don't get a value anywhere close to what you would expect if it was a conversion error between centimeters and inches. The Arm Span is for this does not match up with expected values.

The same method was used for the person who recorded the arm spans that are much too small compared to the Viruvian man averages. There was a P-test of .06 for this example which is above the 5% to reject the null hypothesis, this is due to a variety of factors but if you reduce it to just the 3 values from this persons data that were wrong, the p-test looks almost exactly the same as the long arm span data. (I.E. significantly below 5%).

The histograms are the most visual way to see the discrepancies in the data. Specifically, the histograms with the arm span error vs the dataset, It is a stark difference where the majority of the Data set is in the 60's and the majority of the error data is in the 90's.

The plot charts also clearly show the outliers.

6 Conclusion

The Covariate data for Arm span compared were clearly manufactured data from multiple users. The data is obviously not from a conversion error as clearly seen from the P values as well as the data from the correlation tables, even the means given by the T-tests show how different the values are compared to what they should have been.There is a possibility of one of the pieces in their data being this far off the average, but even then it is unlikely, let alone 3 times or the entire recorded data by that recorder. I am sure that there were more values in the data set that were fabricated. I will continue to look for potential outliers that would point towards a breach in the Data Integrity of the Data Set.

7 Appendix

7.1 Data Provenance

7.1.1 Data Collection and organization

The beginning of the Data Provenance would be the method of collection collected. For my Data Collection personally, it was collected through my immediate family, my Wife, my Daughter, and my Son. I then created and sent out a Handout with all the necessary data for whoever is sent it to. They then filled out their data and sent it back to me via email. This precaution was due to the Covid-19 Pandemic. After all the data was received, it was given to our instructor. He then took that data and combined it into a much larger Data Set that we could Analyze with much more efficiency. The functions that are used to manipulate the data are located at the top of this document, as well as in the Functions folder on GitHub. I then organized the data by using the Viruvian Man Data, changing my main Data Frame from the measure.DF to Proportions.DF. This took only columns that were affected by height and divided them by the height. This gave me numbers that could be checked against the averages found by the Viruvian Man Data. I also cached the final.measure data set to a local hard drive to cut down on access times. The original version of measure.txt had a significant number of missing values and the Inches and CMs weren't converted. I had originally created a method called convert_cm_in that took that data set and changed the data from CM to inches. We were then provided a document that did that already, so the method was unneeded.

7.1.2 Description of the data

The Final.Measure Data set was made up of:
the name of the data Collector, which is the student in the class who is collecting the data.
The Height, which in my case was in inches, from the ground to the top of the subjects head.
The Head Height, which is the height of the head from chin to top of the head.
The head Circumference, which is the measure around the subjects head.
The Arm Span, Which is the measurement from tip of finger to tip of finger with the subjects arms outstreched.
The Floor to Navel measurement, distance from the Floor to the subjects navel.
The Hand Length, The length from the bottom of the palm to tip of finger.
The hand Width, The length of the hand at its widest point.
The Hand to Elbow, the length from the elbow to tip of the finger.
The Reach, how far the subject can reach towards the sky.
The foot Length, The length of the subjects foot.
The knee pit to floor, the length from the kneee pit to the floor.
The hip to floor, the length from your hip to the ground.
The armpit to floor, length from subjects armpit to the ground.
The Dominant writing hand, The hand the subject uses to write.
The Dominant Eye, The subjects dominant Eye, use the ocular dominance test to find this.
The Eye color, color of the subjects eye.
The dominant method of swinging a club, Is the subject a right or left handed swing.
The Age, The subjects Age.
The Gender, the subjects Gender

the Ethnicity, the subjects Ethnicity.
The Quality, the Quality of the subjects measurements.
The minutes, length of time to record the information.
The notes, any additional information the subject or collector decided to add.

7.1.3 Data Collection Handout

7.2 Functions and devtools setup

Below is the necessary functions and libraries required to run the code referenced in this document.

```
library(devtools);       # required for source_url
path.humanVerseWSU = "https://raw.githubusercontent.com/MonteShaffer/humanVerseWSU/"
source_url( paste0(path.humanVerseWSU,"master/misc/functions-project-measure.R") );
```

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

```
convert_cm_in = function(cm){
  cm_inch = round( cm / 2.54, 2 )
  return(paste(cm_inch, "cm"))
}


VirtuvianMAn = function(input, input2)
  {
    return (input/input2)
  }
```

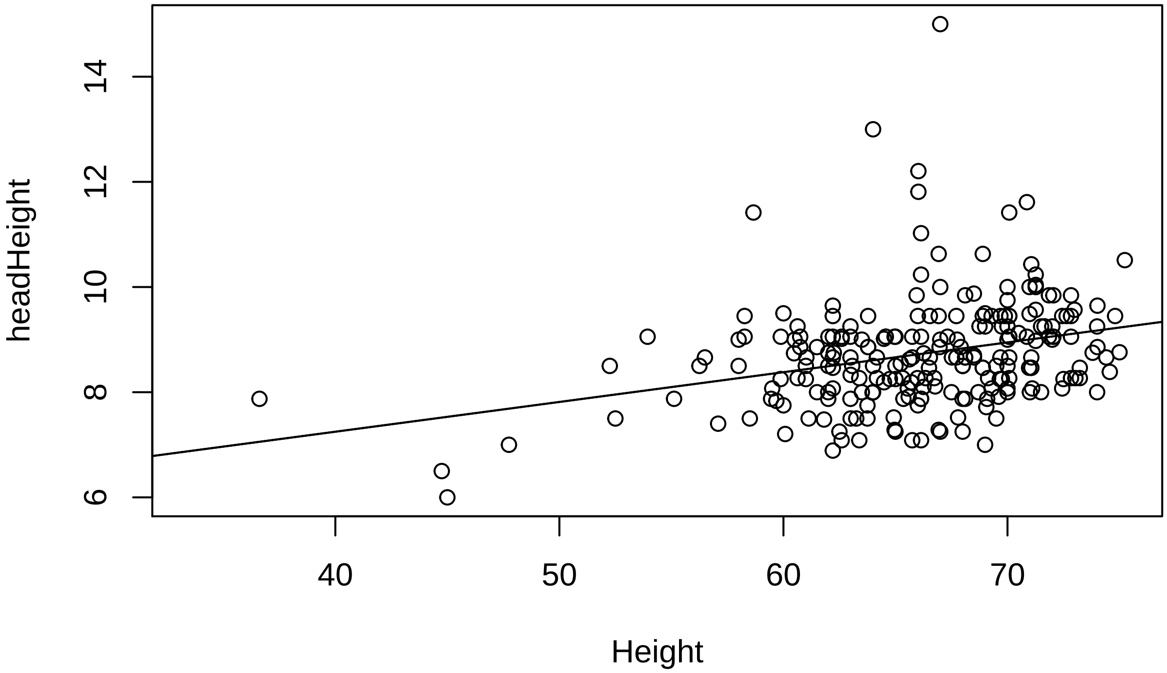
7.3 Retrieve the Data Set

```
path.project = "C:/Users/Galac/Desktop/git419/Stats419_FALL2020/Proj1/"

path.to.data = "C:/Users/Galac/Desktop/stats 419/";
measure = utils::read.csv( paste0(path.to.data, "final.measure.txt"), header=TRUE, quote="", sep="|");

measure.df <- measure

measureNAremoved= na.omit(measure.df)
```

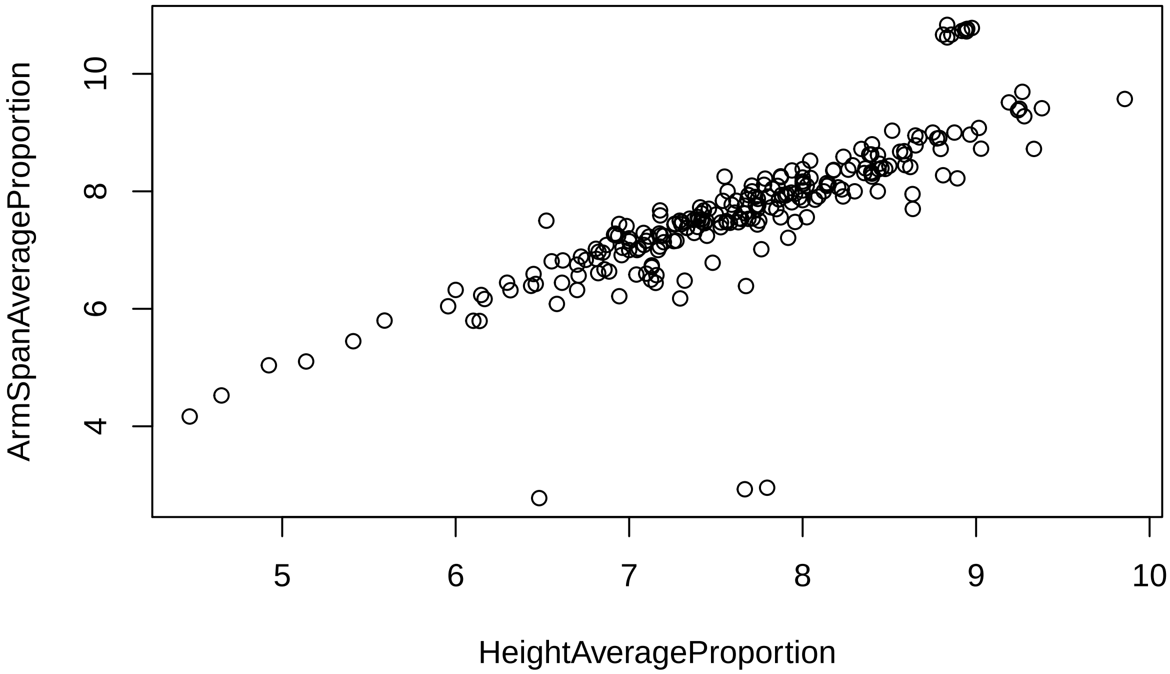
7.4 Set up Proportion Data Frame using Virvuian Man data

```
proportion.df = measure.df

Height = measure.df$height
headHeight = measure.df$head.height

plot(Height, headHeight )
reg.n = lm(headHeight ~ Height)
abline(reg.n)

abline(reg.n)
```

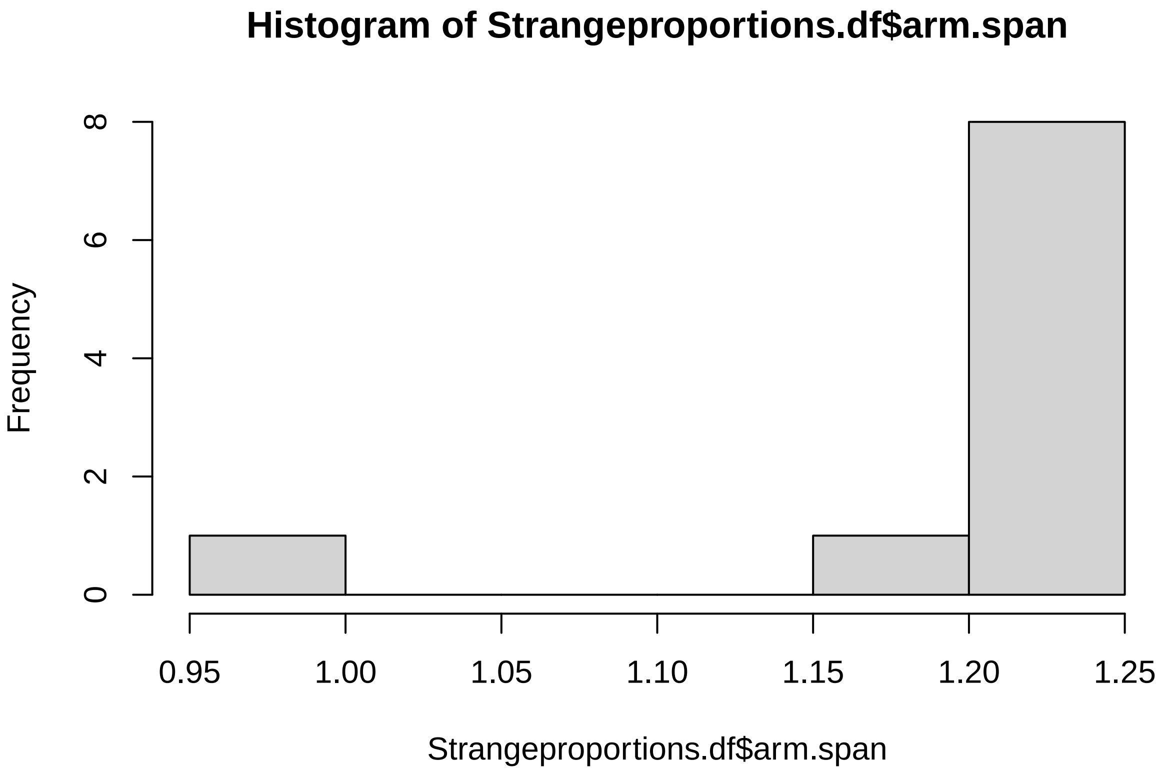
```
proportion.df[,c(3:7, 19:27)] = proportion.df[,c(3:7, 19:27)]/proportion.df$height

proportion.df = measure.df
proportion.df[,c(3:7, 19:27)] = proportion.df[,c(3:7, 19:27)]/proportion.df$height

ArmSpanAverageProportion= VirtuvianMAn(measure.df$arm.span,measure.df$head.height)
HeightAverageProportion = VirtuvianMAn(measure.df$height, measure.df$head.height)

reg.n = lm(ArmSpanAverageProportion ~ HeightAverageProportion)
abline(reg.n)
```



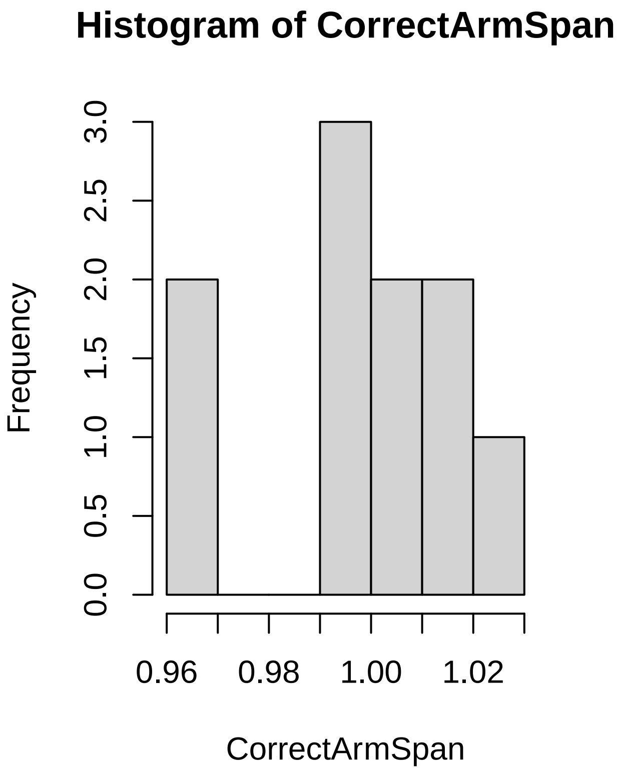
```
plot(HeightAverageProportion, ArmSpanAverageProportion)
```

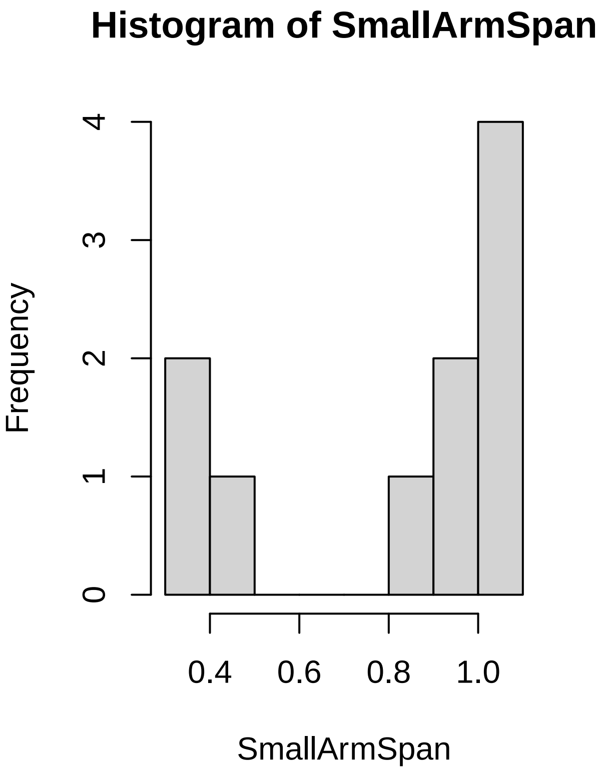


```
reg.n = lm(ArmSpanAverageProportion ~ HeightAverageProportion)

Strangeproportions.df = proportion.df[proportion.df$data_collector == "c51267de031fb6d879a8abf25d260269
Correctproportions.df = proportion.df[proportion.df$data_collector == "fd36e2b3ec59dbd996587454cbb59725

smallerProportions.df = proportion.df[proportion.df$data_collector ==   "5a2f371a934f22dffcf1e994cb6eca4
smallerProportions1.df = smallerProportions.df[-1,]
smallerProportions2.df = smallerProportions1.df[-1,]

Correctproportions1.df = Correctproportions.df[-1,]

hist(Strangeproportions.df$arm.span)
```



```r
par(mfrow = c(1,2))
CorrectArmSpan = Correctproportions1.df$arm.span
SmallArmSpan = smallerProportions2.df$arm.span
hist(CorrectArmSpan)
hist(SmallArmSpan)
```





```
#summary(proportion.df) removed because I show this earlier in the document
```

7.5 *Correlation table for Larger armspan Subset*

Below is the code to generate the summary statistics and save them as a table that you see in Section ??.

```
LargeData.df = proportion.df[proportion.df$data_collector == "c51267de031fb6d879a8abf25d260269", ]
validData.df = proportion.df[proportion.df$data_collector == "fd36e2b3ec59dbd996587454cbb59725",]
heightData1 = validData.df[-1,]
helper = heightData1$arm.span
combined2 = data.frame("goodData" = heightData1, "badData" = LargeData.df)

path.tables = paste0(path.project,"tables/")
file.correlation = paste0(path.tables,"Project-one-correlation.tex");
myData2 = as.matrix(combined2[,c("goodData.arm.span", "badData.arm.span")]);  # numeric values only, on
buildLatexCorrelationTable(myData2,
  rotateTable = TRUE,
  width.table = 0.60, # best for given data ... 0.95 when rotateTable = FALSE
                      # 0.60 when rotateTable = TRUE
  myFile = file.correlation,
  myNames = c("Correct arm span measurements", "Faulty arm span measurements") );
Sys.sleep(2); # in case Knit-PDF doesn't like that I just created the file...
```

Table 1: Descriptive Statistics and Correlation Analysis

		M	SD	1
1	**Correct arm span measurements**	1.0	.02	1
2	**Faulty arm span measurements**	1.2	.07	.64*

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: Descriptive Statistics and Correlation Analysis

		M	SD	1
1	**Correct arm span measurements**	1.0	.02	1
2	**Large Faulty arm span measurements**	.8	.29	-.37

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

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

7.6 Correlation Table for Small armspan Subset

```r
ShortData.df = proportion.df[proportion.df$data_collector == "5a2f371a934f22dffcf1e994cb6eca40", ]

heightData.df = proportion.df[proportion.df$data_collector == "fd36e2b3ec59dbd996587454cbb59725",]
heightData2 = heightData.df[-1,]
ShortDataFix = ShortData.df[-3, ]
ShortDataFix1 = ShortDataFix[-4, ]
helper = heightData1$arm.span
combined = data.frame("goodData" = heightData2, "badData" = ShortDataFix1)

path.tables = paste0(path.project,"tables/")
file.correlation = paste0(path.tables,"Project-one-correlation-table.tex");
myData = as.matrix(combined[,c("goodData.arm.span", "badData.arm.span")]);  # numeric values only, only

buildLatexCorrelationTable(myData,
  rotateTable = TRUE,
  width.table = 0.60, # best for given data ... 0.95 when rotateTable = FALSE
                      # 0.60 when rotateTable = TRUE
  myFile = file.correlation,
  myNames = c("Correct arm span measurements", "Large Faulty arm span measurements") );
Sys.sleep(2); # in case Knit-PDF doesn't like that I just created the file...
```

7.7 Welch T-Tests

```r
  StrangeData.df = proportion.df[proportion.df$data_collector == "c51267de031fb6d879a8abf25d260269", ]
  StrangerData.df = StrangeData.df$arm.span
  heightData.df = proportion.df[proportion.df$data_collector == "fd36e2b3ec59dbd996587454cbb59725",]
  heightData1 = heightData.df[-1,]
  try= round(StrangerData.df, digits = 3)
  helper = heightData1$arm.span
  combined = data.frame("goodData" = helper, "badData" = try)

  proportions1.df = proportion.df[proportion.df$data_collector != "c51267de031fb6d879a8abf25d260269", ]

  CorrectArmSpan = proportions1.df$arm.span
  IncorrectArmSpan = combined$badData

  t.test(CorrectArmSpan, IncorrectArmSpan)
```

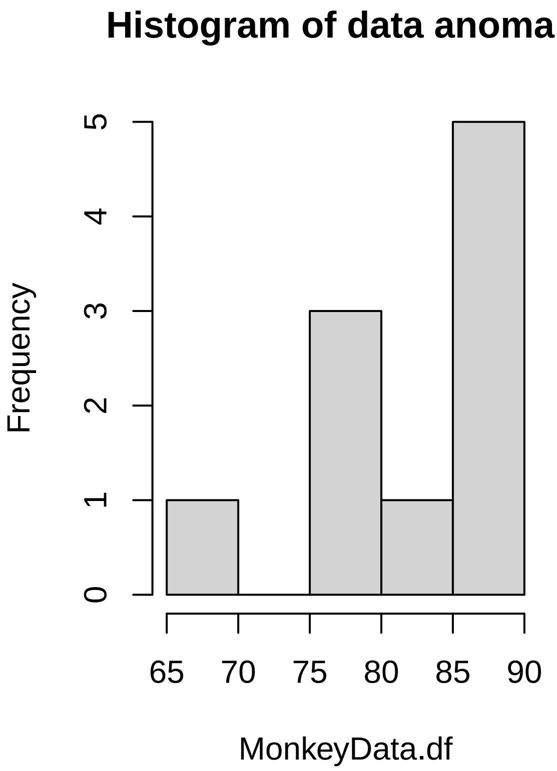
```
##
##  Welch Two Sample t-test
##
## data:  CorrectArmSpan and IncorrectArmSpan
## t = -8.5806, df = 10.062, p-value = 6.086e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2424512 -0.1425588
## sample estimates:
## mean of x mean of y
##  0.991595  1.184100
```

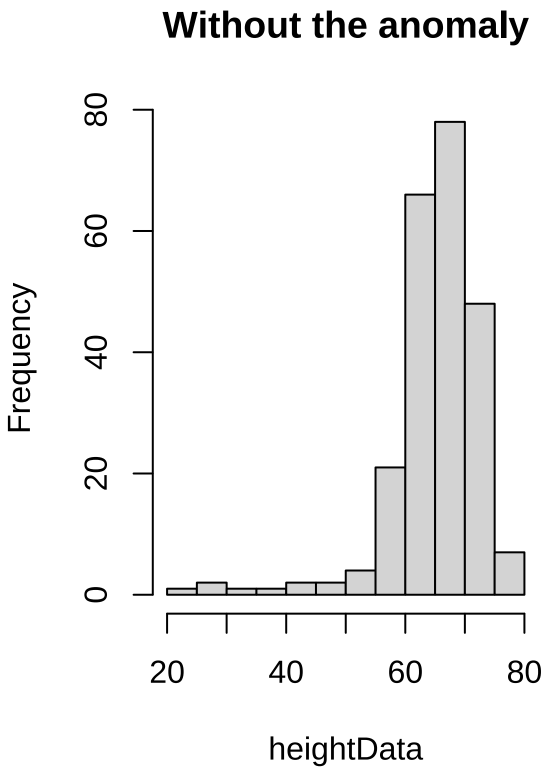
```
  StrangeData.df = proportion.df[proportion.df$data_collector == "5a2f371a934f22dffcf1e994cb6eca40", ]
  StrangeData1.df = StrangeData.df[-1,]
  StrangeData2.df = StrangeData1.df[-4,]
  StrangerData.df = StrangeData2.df$arm.span

  heightData.df = proportion.df[proportion.df$data_collector == "fd36e2b3ec59dbd996587454cbb59725",]
  heightData1 = heightData.df[-1,]
  try= round(StrangerData.df, digits = 3)
  helper = round(heightData1$arm.span, digits = 3)
  combined2 = data.frame("goodData" = helper, "badData" = try)

  proportions2.df = proportion.df[proportion.df$data_collector != "5a2f371a934f22dffcf1e994cb6eca40", ]

  CorrectArmSpan = proportions2.df$arm.span
  IncorrectArmSpan = combined2$badData

  t.test(CorrectArmSpan, IncorrectArmSpan)
```

```
##
##  Welch Two Sample t-test
##
## data:  CorrectArmSpan and IncorrectArmSpan
## t = 2.1027, df = 9.0295, p-value = 0.06473
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.0145928  0.4023104
## sample estimates:
## mean of x mean of y
##  1.007859  0.814000
```

7.8 Final Histograms

```
proportionHist.df = measure.df

proportionHist1.df = proportionHist.df[proportionHist.df$data_collector != "c51267de031fb6d879a8abf25d2


StrangeData.df = measure.df[measure.df$data_collector == "c51267de031fb6d879a8abf25d260269", ]
MonkeyData.df = StrangeData.df$arm.span
par(mfrow = c(1,2))
hist(MonkeyData.df, main = "Histogram of data anomaly")
heightData = proportionHist1.df$arm.span
hist(heightData, main = "Without the anomaly")
```


Histogram of data anomaly
Frequency
0
1
2
3
4
5
65
70
75
80
85
90
MonkeyData.df


Without the anomaly
Frequency
0
20
40
60
80
20
40
60
80
heightData

ENDNOTES

[1] Sometimes when looking for data that was fabricated or altered incorrectly it can be very hard to find.
So finding a good method for searching out the data is probably the most important part. In this graphic we can easily see data that is significanly off the normal path. Both above and below the median. This is a great visual example of failed Data Integrity.

[2] This data is slightly changed from the given data at the start of the Project. I mainly used the proportion DataFrame that I created and I felt it was more fitting that I put the summary of the Proportion data here and not the measure.df

TABLE OF CONTENTS

1 Introduction 1

2 Methods of finding issues in the Data 1

3 Research Question: What is my primary question 1

3.1 What is my secondary question . . . 1

3.2 What is my other secondary question . . . 1

4 Data Description 1

4.1 Summary of Sample . . . 2

4.2 Summary Statistics of Data . . . 2

5 Key Findings 2

6 Conclusion 3

7 Appendix 3

7.1 Data Provenance . . . 3

7.1.1 Data Collection and organization . . . 3

7.1.2 Description of the data . . . 3

7.1.3 Data Collection Handout . . . 4

7.2 Functions and devtools setup . . . 4

7.3 Retrieve the Data Set . . . 4

7.4 Set up Proportion Data Frame using Virvuian Man data . . . 4

7.5 Correlation table for Larger armspan Subset . . . 8

7.6 Correlation Table for Small armspan Subset . . . 11

7.7 Welch T-Tests . . . 11

7.8 Final Histograms . . . 12