Lying with Statistics Project

Contents

Diploma PROOF :)

1 Data manipulation

1.1 Cleans up the RMD output and files by ensuring they don't fall off the page.

1.1.1 A lot of lying with statistics is appearing credible. Thus, we used the "knitr" and "readr" libraries to clean up the graph outputs and ensure there is no run off text in the pdf. This gives the document a clean and clear feel. We used StackOverflow to figure out how to do this and include it in all of our markdowns.

```r
options(tinytex.verbose = TRUE)
options(digits = 5)

#Load Libraries
Libraries <- c("knitr", "readr")
for (p in Libraries) {
  library(p, character.only = TRUE)
}

opts_chunk$set(fig.align='center',
               external=TRUE,
               echo=TRUE,
               warning=FALSE,
               fig.pos='H',
               tidy.opts=list(width.cutoff=60),
               tidy=TRUE,
               warning = FALSE
)
```

1.2 Install and load all required packages (Don't install in rmd)

1.2.1 No functions can be installed in Rmd, but the code is included for reproducibility. tidyverse and dplyr are critical for data manipulation. plotrix helped to create a 3D piechart.

```r
# install.packages('dplyr') install.packages('tidyverse')
# install.packages('knitr') install.packages('readr')
# install.packages('plotrix')

library("tidyverse")
library("dplyr")
library("knitr")
library("readr")
library("plotrix")
```

1.3 Read in files based on type, headers, and with specific NA values accounted for

1.3.1 The data used was from a longitudinal study of 674 children over 4 years. Each file was read in based on type of file while setting certain NA strings to NA. This data cleaning was critical in the results, because otherwise those numbers would significantly alter the results.

```
setwd("./Data")
w1_child <- read.csv("w1_child.csv", header = TRUE, na.strings = c("9",
    "8", "98", "99"))
w2_child <- read.table("w2_child.dat", header = TRUE, , na.strings = c("-999"))
w3_child <- read.csv("w3_child.csv", header = TRUE, na.strings = c("9",
    "8", "98", "99"))
w4_child <- read.csv("w4_child.csv", header = FALSE, na.strings = c("9",
    "8", "98", "99"))
names_w4_child <- read.table("names_w4_child.txt")
educinc <- read.csv("educinc.csv", header = TRUE, na.strings = c("."))
```

1.4 Add transposed variable names to w4_child based off of the provided text file

1.4.1 Adding variable names was critical to accurately tracking data and ensuring results are not misconstrued by mislabeling.

```
names(w4_child) <- t(names_w4_child)
```

1.5 Make all headers into lower case for easier merging

```
names(w1_child) <- tolower(names(w1_child))
names(w2_child) <- tolower(names(w2_child))
names(w3_child) <- tolower(names(w3_child))
names(w4_child) <- tolower(names(w4_child))
names(educinc) <- tolower(names(educinc))
```

1.6 Merge files by famid and select specified variables

1.6.1 Although there was a lot more data, selecting which variables were applicable lay at the crux of our data analysis. Other extraneous variables would only serve to clutter the data set and possibly cause inaccurate filtering of NA values down the road. In this analysis, we focused on five variables: "Child's use of alcohol, tobacco, or other drugs", "Child's attachment to teachers", "Child's attachment to school", "Child's peer competence", "Child's perceived discrimination". Demographic information such as gender, number of siblings, etc. was also included for further inspection.

```r
w1234 <- (list(w1_child, w2_child, w3_child, w4_child, educinc) %>%
    reduce(full_join, by = "famid")) %>%
    dplyr::select(famid, c01cohort, c01gender, c01school, c01sibli,
        contains("atts"), contains("pcmp"), contains("attt"),
        contains("dscr"), contains("atod"), fameduc, income,
        c01sibli, contains("edex"))
```

1.7 Reverse code for pcmp 1 and 2

1.7.1 The vairable child's peer competence had an inverse score, in comparison to the other scores. This meant that a low peer competence score was a negative thing. Since the study directors ordered the other variables with a high score as negative e.g., 7 for increased percived discrimination, we needed to recode the variables to ensure accruate correlation and analysis results.

```r
pcmp_01_02_cols <- c(grep("pcmp01", names(w1234)), grep("pcmp02",
    names(w1234)))
w1234[, pcmp_01_02_cols] <- 5 - w1234[, pcmp_01_02_cols]
```

1.8 Compute averages based on sets of columns for variable sets and place average into a new variable

1.8.1 In this analysis, we investigated averages to ensure broader conclusions. We chose not to look closely at the nuances of each data piece. We sought macro over micro conclusions.

```r
w1234$c01attt <- rowMeans(w1234[c(grep("c01attt", names(w1234)))],
    na.rm = TRUE)
w1234$c04attt <- rowMeans(w1234[c(grep("c04attt", names(w1234)))],
    na.rm = TRUE)

w1234$c01pcmp <- rowMeans(w1234[c(grep("c01pcmp", names(w1234)))],
    na.rm = TRUE)
w1234$c04pcmp <- rowMeans(w1234[c(grep("c04pcmp", names(w1234)))],
    na.rm = TRUE)

w1234$c01dscr <- rowMeans(w1234[c("c01dscr07", "c01dscr08", "c01dscr09",
    "c01dscr10")], na.rm = TRUE)
w1234$c04dscr <- rowMeans(w1234[c("c04dscr07", "c04dscr08", "c04dscr09",
    "c04dscr10")], na.rm = TRUE)

w1234$c01atts <- rowMeans(w1234[c("c01atts03", "c01atts07", "c01atts08",
    "c01atts10")], na.rm = TRUE)
w1234$c02atts <- rowMeans(w1234[c("c02atts03", "c02atts07", "c02atts08",
    "c02atts10")], na.rm = TRUE)
w1234$c03atts <- rowMeans(w1234[c("c03atts03", "c03atts07", "c03atts08",
    "c03atts10")], na.rm = TRUE)
w1234$c04atts <- rowMeans(w1234[c("c01atts03", "c04atts07", "c04atts08",
```

```r
    "c04atts10")], na.rm = TRUE)

w1234$c01atod <- rowSums(w1234[c("c01atod01", "c01atod02", "c01atod03",
    "c01atod04", "c01atod05", "c01atod06", "c01atod07", "c01atod08",
    "c01atod09")], na.rm = TRUE)
w1234$c04atod <- rowSums(w1234[c("c04atod01", "c04atod02", "c04atod03",
    "c04atod04", "c04atod05", "c04atod06", "c04atod07", "c04atod08",
    "c04atod09")], na.rm = TRUE)
```

1.9 Create difference scores between Waves 1 and 4

1.9.1 Looking across a 4 year study provided an invaluable look at children's changing views. Thus, we chose to catelog and statistically explore these changes across this large time span. We utilized absolute value to compare changes broadly, with less interest in the change's final direction, but the magnititude the of the score differences.

```r
score_variables <- c("atts", "pcmp", "attt", "dscr", "atod")
w1234[paste("difference_", score_variables, sep = "")] <- w1234[paste("c01",
    score_variables, sep = "")] - w1234[paste("c04", score_variables,
    sep = "")]
ave_abs_change <- abs(colMeans(w1234[, c("difference_atts", "difference_pcmp",
    "difference_attt", "difference_dscr", "difference_atod")],
    na.rm = TRUE))

atts <- abs(mean(w1234$c01atts - w1234$c04atts, na.rm = TRUE))
pcmp <- abs(mean(w1234$c01pcmp - w1234$c04pcmp, na.rm = TRUE))
attt <- abs(mean(w1234$c01attt - w1234$c04attt, na.rm = TRUE))
dscr <- abs(mean(w1234$c01dscr - w1234$c04dscr, na.rm = TRUE))
atod <- abs(mean(w1234$c01atod - w1234$c04atod, na.rm = TRUE))
```

1.10 Function to generate a random vector of colors in order from

1.10.1 We coded this function to return a generic color vector with length of "num". When graphing bargraphs and histograms, manually inputting each color of each bar e.g., c("blue", "red", "yellow") consumed extra space and effort. After research online, we found a cool method that creates a vector of pretty in order colors that we wielded.

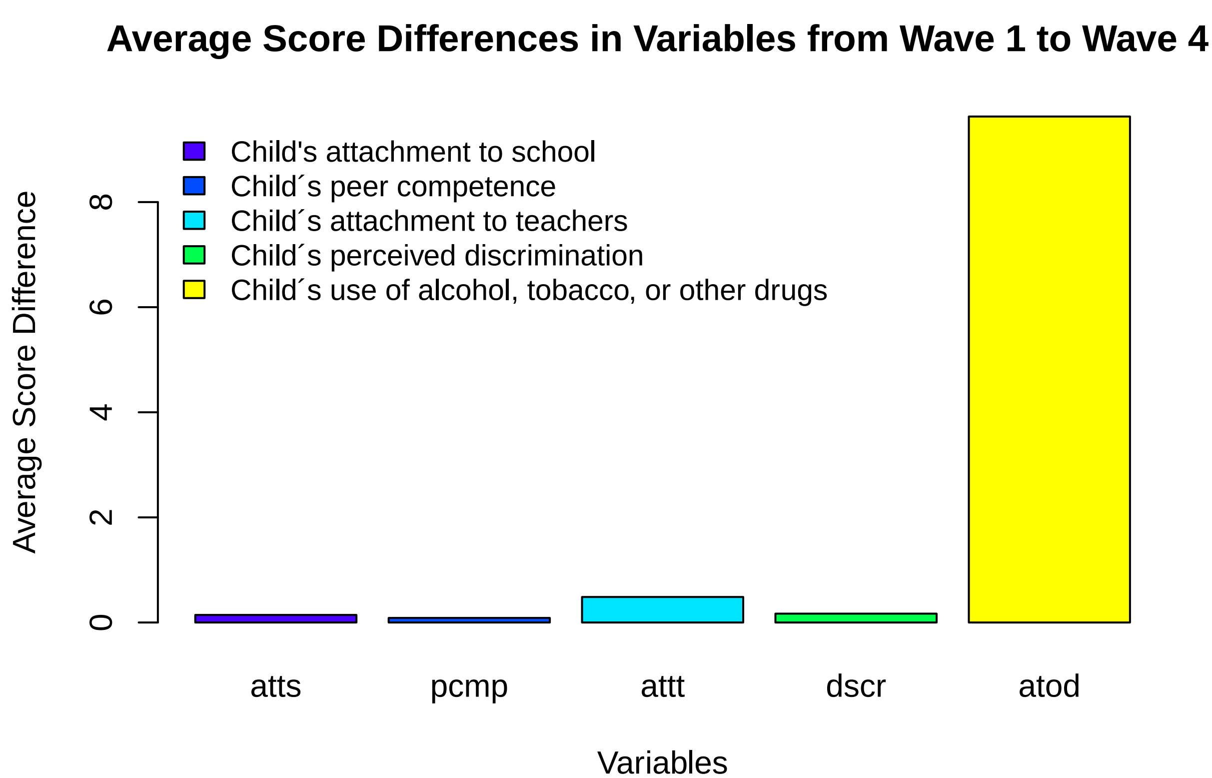
```r
color_vector <- function(num) {
    return(topo.colors(num))
}
```

2 A fair figure depicting data

2.0.1 We decided to graph the average score differences for the five chosen variables longitutionally. This figure nicely sums up the major changes in a clear and effective format. The title, legend, x and y axes acurratedy portrays the results. The labels lack any misleadning information and the scales lie within appropriate ranges for the data. Even the colors fail to persuade.

```
average_change <- c(atts, pcmp, attt, dscr, atod)
barplot(average_change, main = "Average Score Differences in Variables from Wave 1 to Wave 4 ",
    xlab = "Variables", ylab = "Average Score Difference", names.arg = c("atts",
        "pcmp", "attt", "dscr", "atod"), col = color_vector(5))

legend("topleft", c("Child's attachment to school", "Child’s peer competence",
    "Child’s attachment to teachers", "Child’s perceived discrimination",
    "Child’s use of alcohol, tobacco, or other drugs"), cex = 0.9,
    bty = "n", fill = color_vector(5))
```



3 Conducting a statistical test such as t-squared test on girls vs boys

3.1 Recode girls and boys

3.1.1 We found the intersection of gender on child development to pose some questions. Did gender significantly correlate with developmental factors such as attachment to school, peer competence, attachment to teachers, percieved discrimination, and substance usage? Does gender play a role in socialization i.e., peer competence and attachment to teachers more than substance usage, which many consider to be less associated with gender?

```
w1234$c01gender[w1234$c01gender == 1] = "girl"
w1234$c01gender[w1234$c01gender == 2] = "boy"
```

4 Compare differences across genders by creating a t-test for genders function

4.0.1 The t-test provided a reliable method of discerning the correlation between gender groups and these variables. Both the p-value and t-value provide insight into the statistical significance and level of results. The function t_test_gender provided an easy shorthand to perform this analysis with the child's gender.

```
t_test_gender <- function(score_var) {
    return(t.test(score_var ~ c01gender, paired = FALSE, data = w1234))
}

tatod <- t_test_gender(w1234$difference_atod)
tattt <- t_test_gender(w1234$difference_attt)
tatts <- t_test_gender(w1234$difference_atts)
tdscr <- t_test_gender(w1234$difference_dscr)
tpcmp <- t_test_gender(w1234$difference_pcmp)

tatod
```

```
##
##  Welch Two Sample t-test
##
## data:  score_var by c01gender
## t = -0.122, df = 669, p-value = 0.9
## alternative hypothesis: true difference in means between group boy and group girl is not equal to 0
## 95 percent confidence interval:
##  -0.87090  0.76865
## sample estimates:
##  mean in group boy mean in group girl
##            -9.6409            -9.5898
```

```
tattt
```

```
##
##  Welch Two Sample t-test
##
## data:  score_var by c01gender
## t = 1.32, df = 581, p-value = 0.19
## alternative hypothesis: true difference in means between group boy and group girl is not equal to 0
## 95 percent confidence interval:
##  -0.043818  0.221895
## sample estimates:
##  mean in group boy mean in group girl
##            0.52987            0.44083
```

```
tatts
```

```
##
##  Welch Two Sample t-test
##
## data:  score_var by c01gender
## t = 0.055, df = 666, p-value = 0.96
## alternative hypothesis: true difference in means between group boy and group girl is not equal to 0
## 95 percent confidence interval:
##  -0.087517  0.092561
## sample estimates:
##  mean in group boy mean in group girl
##            0.14484            0.14232
```

```
tdscr
```

```
##
##  Welch Two Sample t-test
##
## data:  score_var by c01gender
## t = 0.0432, df = 572, p-value = 0.97
## alternative hypothesis: true difference in means between group boy and group girl is not equal to 0
## 95 percent confidence interval:
##  -0.077353  0.080834
## sample estimates:
##  mean in group boy mean in group girl
##            0.16956            0.16782
```

```
tpcmp
```

```
##
##  Welch Two Sample t-test
##
## data:  score_var by c01gender
## t = 1.19, df = 579, p-value = 0.23
## alternative hypothesis: true difference in means between group boy and group girl is not equal to 0
## 95 percent confidence interval:
##  -0.031859  0.130578
```

```
## sample estimates:
##  mean in group boy mean in group girl
##          0.111693           0.062333
```

5 Display statistical analysis results of gender variances across score differences

5.1 Function to extract p values from t-test result text

5.1.1 This function offered a clean and clear shorthand for extracting important data from the t-test results. We later graphed these results, avoiding repition and clear delineage to where to this data came from.

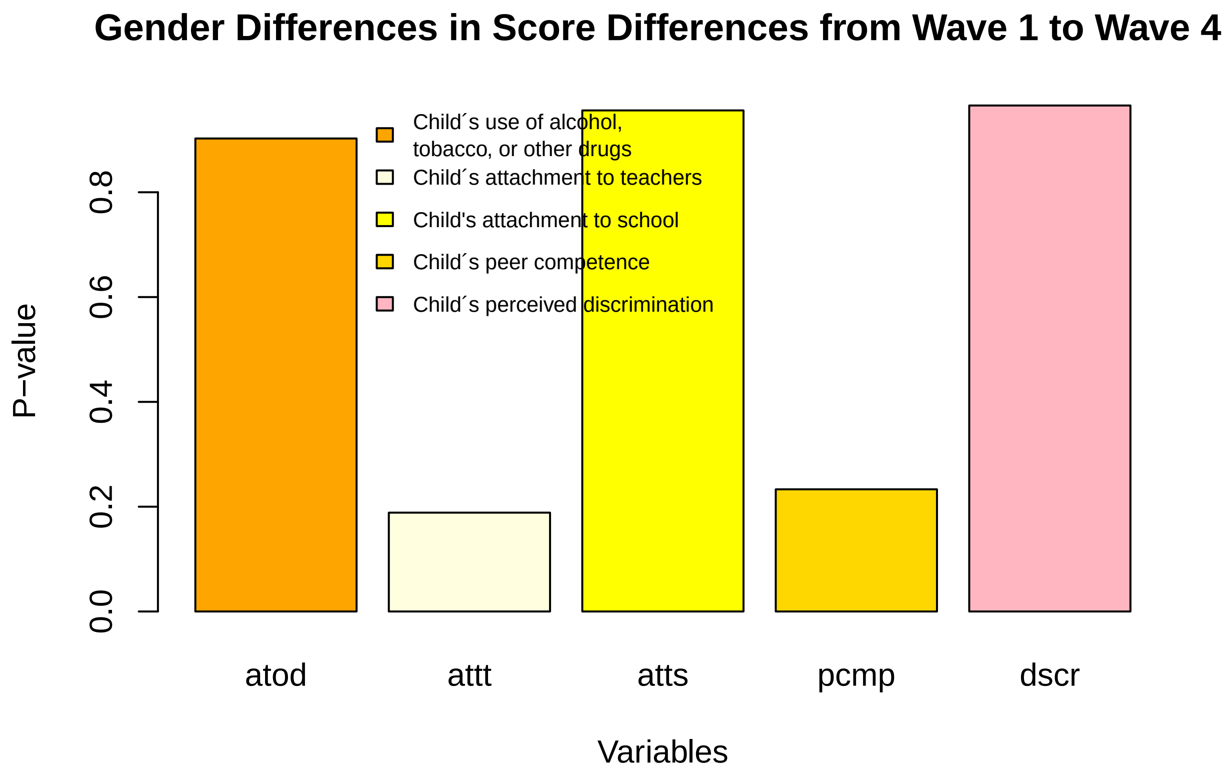
```
extract_pval <- function(ttest) {
    return(ttest$p.value)
}
```

5.2 Graph p-values

5.2.1 After extracting p-values from the dense wordy responses, we outlined a clear graph to visualize the responses.

```
p_values <- c(extract_pval(tatod), extract_pval(tattt), extract_pval(tatts),
    extract_pval(tpcmp), extract_pval(tdscr))
barplot(p_values, main = "Gender Differences in Score Differences from Wave 1 to Wave 4 ",
    xlab = "Variables", ylab = "P-value", names.arg = c("atod",
        "attt", "atts", "pcmp", "dscr"), col = c("orange", "lightyellow",
        "yellow", "gold", "lightpink"))

legend(1.2, 0.99, c("Child’s use of alcohol,\ntobacco, or other drugs",
    "Child’s attachment to teachers", "Child's attachment to school",
    "Child’s peer competence", "Child’s perceived discrimination"),
    cex = 0.7, bty = "n", fill = c("orange", "lightyellow", "yellow",
        "gold", "lightpink"))
```


Gender Differences in Score Differences from Wave 1 to Wave 4
Child´s use of alcohol, tobacco, or other drugs
Child´s attachment to teachers
Child's attachment to school
Child´s peer competence
Child´s perceived discrimination
P−value
0.0
0.2
0.4
0.6
0.8
atod
attt
atts
pcmp
dscr
Variables

6 Interpretation of results

6.1 The results demonstrate all p values > 0.05. This illuminates that the hypothesis that gender would signfiicantly correlate with differences in scores can likely be rejected. Further, it appears substance use, attachment to school, and percieved discrimination vary due to randomness, not gender. On the other hand, attachment to teachers and child peer competence have much lower p values, falling around 0.2. Albeit not statistically significant, it is important to consider the differences these factors could still play given not fall off p-values. Thus, the idea that gender socialization might be playing a role in these factors might require further investigation.

7 Figure and analysis that provides a distorted version of what we actually would find in the data.

7.1 Code to determine which p_values are stastically significant i.e., <0.05

7.1.1 In this code, although the data is correct, the interpretation is severely skewed. The phrasing intentionally simplies the previous graph and results.

```
p_values[p_values < 0.05] = "yes"
p_values[p_values > 0.05] = "ABSOLUTELY NO CHANGE"
percent_not_significant <- c(length(p_values[p_values > 0.05])/5 *
    100, (length(p_values[p_values < 0.05])/5 * 100) + 0.01)
```

8 Figure that significantly simplies the data

8.0.1 The red color and 3D pie chart serve to dominant the idea the viewer is supposed to percieve. The graph portrays a harsh lack of gender influence on these longitutional variables, when in reality, the story is much more nuanced. We employed the "all-or-nothing" tatic to take 'true' data and favor a narrative.

```
pie3D(percent_not_significant, col = c("red", "blue"), labels = c(percent_not_significant[1],
    "0"), main = "% of Score Difference Variables that Statistically Differed by Gender")

legend(-0.8, 1.05, c("STATISTICALLY NO DIFFERENCE", "yes"), cex = 0.8,
    fill = c("red", "blue"))
```

% of Score Difference Variables that Statistically Differed by Gender