Homework2

Jae Hoon Cho

October 2, 2018

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

## 1. To make graph

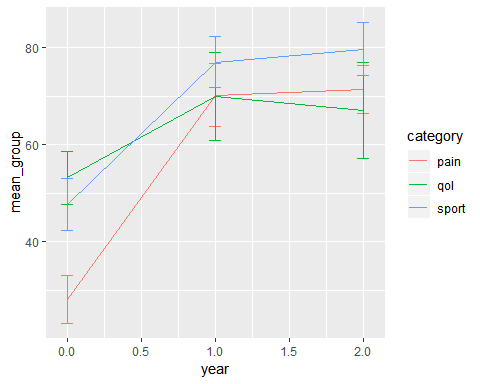
library(tidyr)  
library(ggplot2)  
lab2 <- read.csv("lab2.csv")  
lab2 <- lab2 %>% gather('base\_sport', 'base\_qol', 'base\_pain','first\_sport', 'first\_qol', 'first\_pain','second\_sport', 'second\_qol', 'second\_pain' ,key="index", value="value" )  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

lab2 <- lab2 %>% mutate (category = ifelse (index=="base\_sport"|index=="first\_sport"|index=="second\_sport", "sport", ifelse(index=="base\_qol"|index=="first\_qol"|index=="second\_qol", "qol", "pain")))   
lab2 <- lab2 %>% mutate (year = ifelse (index=="base\_sport"|index=="base\_qol"|index=="base\_pain", 0, ifelse(index=="first\_qol"|index=="first\_sport"|index=="first\_pain", 1, 2)))   
  
summary\_lab <- lab2 %>% group\_by (category, year) %>% summarize (mean\_group=mean(value), sd\_group=sd(value))  
  
summary\_lab <- summary\_lab %>% mutate(y\_max=mean\_group+sd\_group, y\_min=mean\_group-sd\_group)  
  
ggplot(summary\_lab, aes(x=year, y=mean\_group, group=category, color=category)) +   
 geom\_line() + geom\_errorbar(aes(ymax=y\_max, ymin=y\_min), width=0.08, size=0.05)



## 2. To make 2 tables as one

# To make expenditure data tidy  
expenditure <- read.csv("expenditures (1).csv")  
expenditure <- gather(data=expenditure, 'X1991\_\_Total.Health.Spending','X1992\_\_Total.Health.Spending','X1993\_\_Total.Health.Spending','X1994\_\_Total.Health.Spending','X1995\_\_Total.Health.Spending','X1996\_\_Total.Health.Spending', 'X1997\_\_Total.Health.Spending', 'X1998\_\_Total.Health.Spending','X1999\_\_Total.Health.Spending','X2000\_\_Total.Health.Spending',  
 'X2001\_\_Total.Health.Spending','X2002\_\_Total.Health.Spending','X2003\_\_Total.Health.Spending','X2004\_\_Total.Health.Spending','X2005\_\_Total.Health.Spending','X2006\_\_Total.Health.Spending','X2007\_\_Total.Health.Spending', 'X2008\_\_Total.Health.Spending', 'X2009\_\_Total.Health.Spending','X2010\_\_Total.Health.Spending',  
 'X2011\_\_Total.Health.Spending','X2012\_\_Total.Health.Spending','X2013\_\_Total.Health.Spending','X2014\_\_Total.Health.Spending', key="index", value="value")   
library (stringr)  
expenditure$year <- str\_sub (expenditure$index,2,5)  
expenditure$year <- as.numeric(expenditure$year)  
expenditure <- expenditure %>% select (Location, value, year)  
expenditure <- expenditure [c(1,3,2)]  
expenditure <- expenditure %>% filter (!is.na(value))  
expenditure$Location <- as.character (expenditure$Location)

## Including Plots

You can also embed plots, for example:

# To make coverage date tidy   
coverage <- read.csv("coverage2.csv", header=T)  
  
coverage <- gather(data=coverage, 'X2013\_\_Employer','X2013\_\_Non.Group', 'X2013\_\_Medicaid','X2013\_\_Medicare','X2013\_\_Other.Public','X2013\_\_Uninsured','X2013\_\_Total','X2014\_\_Employer','X2014\_\_Non.Group','X2014\_\_Medicaid','X2014\_\_Medicare','X2014\_\_Other.Public','X2014\_\_Uninsured','X2014\_\_Total','X2015\_\_Employer','X2015\_\_Non.Group','X2015\_\_Medicaid','X2015\_\_Medicare','X2015\_\_Other.Public','X2015\_\_Uninsured','X2015\_\_Total','X2016\_\_Employer','X2016\_\_Non.Group','X2016\_\_Medicaid','X2016\_\_Medicare','X2016\_\_Other.Public','X2016\_\_Uninsured','X2016\_\_Total', key="index", value="value")

## Warning: attributes are not identical across measure variables;  
## they will be dropped

coverage <- coverage %>% mutate (year=ifelse(index %in% c('X2013\_\_Employer','X2013\_\_Non.Group', 'X2013\_\_Medicaid','X2013\_\_Medicare','X2013\_\_Other.Public','X2013\_\_Uninsured','X2013\_\_Total'), 2013,   
 ifelse(index %in% c('X2014\_\_Employer','X2014\_\_Non.Group','X2014\_\_Medicaid','X2014\_\_Medicare','X2014\_\_Other.Public','X2014\_\_Uninsured','X2014\_\_Total'), 2014,  
 ifelse(index %in% c('X2015\_\_Employer','X2015\_\_Non.Group','X2015\_\_Medicaid','X2015\_\_Medicare','X2015\_\_Other.Public','X2015\_\_Uninsured','X2015\_\_Total'), 2015, 2016))))  
coverage <- coverage %>% mutate (coverage\_type=ifelse(index %in% c('X2013\_\_Employer', 'X2014\_\_Employer', 'X2015\_\_Employer', 'X2016\_\_Employer'), "Employer",   
 ifelse(index %in% c('X2013\_\_Non.Group','X2014\_\_Non.Group','X2015\_\_Non.Group','X2016\_\_Non.Group'), "Non.Group",  
 ifelse(index %in% c('X2013\_\_Medicaid','X2014\_\_Medicaid','X2015\_\_Medicaid','X2016\_\_Medicaid'), "Medicaid",  
 ifelse(index %in% c('X2013\_\_Medicare','X2014\_\_Medicare','X2015\_\_Medicare','X2016\_\_Medicare'), "Medicare",  
 ifelse(index %in% c('X2013\_\_Other.Public','X2014\_\_Other.Public','X2015\_\_Other.Public','X2016\_\_Other.Public'), "ohter.Public",  
 ifelse(index %in% c('X2013\_\_Uninsured','X2014\_\_Uninsured','X2015\_\_Uninsured','X2016\_\_Uninsured'), "Uninsured", "Total")))))))  
coverage <- coverage %>% select (Location, value, year, coverage\_type)  
coverage$value <- as.numeric(coverage$value)

## Warning: NAs introduced by coercion

coverage <- coverage %>% filter (!is.na(value))   
coverage <- coverage [c(1,3,2,4)]  
coverage$Location <- as.character(coverage$Location)

# To merge two tables  
expenditure <- rename (expenditure, total\_value=value)  
expenditure$value <- NA  
expenditure$coverage\_type <-NA  
coverage$total\_value <- NA  
coverage <- coverage[c(1,2,4,3,5)]  
expenditure <- expenditure [c(1,2,5,3,4)]  
Total\_table <- bind\_rows(coverage,expenditure)  
View (Total\_table)

Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.