

HW2_McGill_Max

Due Monday September 11, 10 am

Max McGill

2017-09-11

Problem 4

In the classroom, version control can assist in:

- updating and reverting code as necessary for assignments and projects
- maintaining and securing multiple versions of code for comparison, reversion, and conversion
- facilitating collaborative efforts for projects that require such methods

These functionalities benefit a classroom environment that makes heavy use of programming methods by providing the tools necessary to alter code, either alone or collaboratively, in such a way that facilitates an ease of maneuverability between versions of the code developed or used. In turn, this provides access to a host of options for the modification and sharing of code that simplify the classroom experience of tasks.

Problem 5

Part A

```
## Warning: package 'bindrcpp' was built under R version 3.4.1
```

##	Item	Person	value
##	Length:150	Length:150	Min. :0.700
##	Class :character	Class :character	1st Qu.:3.025
##	Mode :character	Mode :character	Median :4.700
##			Mean :4.657
##			3rd Qu.:6.000
##			Max. :9.400

Part B

##	Year	Long Jump
##	Min. :1896	Min. :249.8
##	1st Qu.:1921	1st Qu.:295.4
##	Median :1950	Median :308.1
##	Mean :1945	Mean :310.3
##	3rd Qu.:1971	3rd Qu.:327.5
##	Max. :1992	Max. :350.5

Part C

##	Brain	Body
##	Min. : 0.005	Min. : 0.10

```
## 1st Qu.: 0.600 1st Qu.: 4.25
## Median : 3.342 Median : 17.25
## Mean : 198.790 Mean : 283.13
## 3rd Qu.: 48.203 3rd Qu.: 166.00
## Max. :6654.000 Max. :5712.00
```

Part D

```
## Clone Replicate value
## Length:18 Length:18 Length:18
## Class :character Class :character Class :character
## Mode :character Mode :character Mode :character
## Variety
## Length:18
## Class :character
## Mode :character
```

Problem 6

Problem 7

Appendix 1: R Code

Problem 5

```
#Part A
urla<-"http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/Sensory.dat"
sensoryraw<-read.table(urla, header=F, skip=1, fill=T, stringsAsFactors = F)
sensorytidy<-sensoryraw[-1,]
#The data is recorded in triplicate, resulting in missing sets of item values when read.
#This causes a shift of the values in rows with an integer and those without.
#Create two dataframes by filtering rows with integers in V1.
#Rename columns in the wone with integer values appropriately.
sensorytidy_a<-filter(.data = sensorytidy,V1 %in% 1:10) %>%
  rename(Item=V1,V1=V2,V2=V3,V3=V4,V4=V5,V5=V6)
#In the one without, create a column to represent missing item values.
sensorytidy_b<-filter(.data = sensorytidy,! (V1 %in% 1:10)) %>%
  mutate(Item=rep(as.character(1:10),each=2)) %>%
  mutate(V1=as.numeric(V1)) %>%
  select(c(Item,V1:V5))
#bind the dataframes and name their columns as relevant.
sensorytidy<-bind_rows(sensorytidy_a,sensorytidy_b)
colnames(sensorytidy)<-c("Item",paste("Person",1:5,sep="_"))
#Gather and mutate the person variable to create a numerically valued column.
#Arrange the data by item.
sensorytidy<-sensorytidy %>%
  gather(Person,value,Person_1:Person_5) %>%
  mutate(Person = gsub("Person_", "",Person)) %>%
```

```

    arrange(Item)
summary(sensorytidy)

```

#Part B

```

urlb<-"http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/LongJumpData.dat"
goldraw<-read.table(urlb,header=T,fill=T,stringsAsFactors=F)
#The data are split among multiple columns that must be combined into coherence.
#Create new dataframes for each variable by selecting their respective columns.
#Then gather their columns into new data frames to create a singular column for each.
yeardf<-select(goldraw,"Year","Jump","Long.1","Year.2")
yearvalue<-gather(yeardf,Year:Year.2)
longjumpdf<-select(goldraw,"Long","Year.1","Jump.1","Long.2")
longjumpvalue<-gather(longjumpdf,Long:Long.2)
#Combine these combined columns to create a new dataframe.
goldna<-data.frame(yearvalue$value,longjumpvalue$value)
#Remove the nonexistent values to create the tidy data frame.
goldtidy<-drop\_na(goldna)
#Set the variable names to appropriate values.
colnames(goldtidy)<-c("Year","Long Jump")
summary(goldtidy)

```

#Part C

```

urlc<-"http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BrainandBodyWeight.dat"
brainbodyraw<-read.table(urlc, header=F, skip=1, fill=T, stringsAsFactors = F)
#This data contains two variables spread across multiple columns.
#Rename the columns using repetition to properly designate the variables.
colnames(brainbodyraw)<-rep(c("Brain","Body"),3)
#Create a new dataframe by binding the individual pairs of columns.
brainbodytidy<-rbind(brainbodyraw[,1:2],brainbodyraw[,3:4],
                    brainbodyraw[,5:6])
#Remove nonexistent values.
brainbodytidy<-brainbodytidy %>%
  filter(!(is.na(Brain)))
summary(brainbodytidy)

```

```

urld<-"http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/tomato.dat"
tomatoraw<-read.table(urld, header=F, skip=2, fill=T, stringsAsFactors = F, comment.char =
  "")
#The data is listed in triplicate by commas, contains an extraneous comma in one variable,
#and contains a character hiding the first row of data.
tomatotidy<-tomatoraw %>%
#Separate each collection of observations into appropriate columns.
  separate(V2,into=paste("C10000",1:3,sep="_"),sep=",",remove=T, extra="merge") %>%
  separate(V3,into=paste("C20000",1:3,sep="_"),sep=",",remove=T, extra="merge") %>%
  separate(V4,into=paste("C30000",1:3,sep="_"),sep=",",remove=T, extra="merge") %>%
#Remove the extraneous comma.
  mutate(C10000_3=gsub(""," ",C10000_3)) %>%
#Gather columns and rename the variables as appropriate.
  gather(Clone,value,C10000_1:C30000_3) %>%
  mutate(Variety=V1, Clone=gsub("C","",Clone)) %>%
  mutate(Variety=gsub("\\\\\\\\\\#", " ",Variety)) %>%
#Separate the clone column into variables clone and replicate.
  separate(Clone,into = c("Clone","Replicate")) %>%
#Exclude the V1 column and arrange by variety.

```

```
select(-V1,Variety,Clone,value) %>%  
  arrange(Variety)  
#All variables are still listed as characters.  
summary(tomatotidy)
```

Problem 6

Problem 7