HW2_Wei_Yanran_Problems 4-7

Yanran Wei

September 12, 2017

Problem 4

Version control can assist in: 1). **Bcakup.** If any server dies, and these system were collaborating via Version Control Systems, any of the repositories can be copied back up to the server to restore it. Every clone is a full backup of all the data. 2). **Revision History.** Version Control Systems have a simple databse that keeps all the changes to files under revision control. 3). **Collaboration.** People need to collaborate with developers on other systems. Version Control Systems can solve this problem. These systems have a single server that contains all the versioned files, and people can check out files from that central place.

Problem 5

```
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
```

```
library(tidyr)
library(readr)
```

Part A. Sensory Data

```
url<-"http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/Sensory.dat"
   Sensory_raw<-read.table(url, header=F, skip=1, fill=T, stringsAsFactors = F)
   Sensory_tidy<-Sensory_raw[-1,]
   Sensory_tidy_a<-filter(.data = Sensory_tidy,V1 %in% 1:10) %>%
                   rename(Item=V1,V1=V2,V2=V3,V3=V4,V4=V5,V5=V6)
   Sensory_tidy_b<-filter(.data = Sensory_tidy,!(V1 %in% 1:10)) %>%
                   mutate(Item=rep(as.character(1:10),each=2)) %>%
                   mutate(V1=as.numeric(V1)) %>%
                   select(c(Item,V1:V5))
   Sensory_tidy<-bind_rows(Sensory_tidy_a,Sensory_tidy_b)
   colnames(Sensory_tidy)<-c("Item",paste("Person",1:5,sep="_"))
   Sensory_tidy<-Sensory_tidy %>%
       gather(Person,value,Person_1:Person_5) %>%
       mutate(Person = gsub("Person_","",Person)) %>%
       arrange(Item)
```

```r
knitr::kable(summary(Sensory_tidy), caption="Sensory data summary")
```

Table 1: Sensory data summary

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
NA	NA	Mean :4.657
NA	NA	3rd Qu.:6.000
NA	NA	Max. :9.400

Part B. Long Jump data

```r
url<-"http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/LongJumpData.dat"
    LongJump_raw<-read.table(url, header=F, skip=1, fill=T, stringsAsFactors = F)
    colnames(LongJump_raw)<-rep(c("V1","V2"),4)
    LongJump_tidy<-rbind(LongJump_raw[,1:2],LongJump_raw[,3:4],
                             LongJump_raw[,5:6],LongJump_raw[,7:8])
    LongJump_tidy<-LongJump_tidy %>%
        filter(!(is.na(V1))) %>%
        mutate(YearCode=V1, Year=V1+1900, dist=V2) %>%
        select(-V1,-V2)
```

```r
knitr::kable(summary(LongJump_tidy), caption="Long Jump data summary")
```

Table 2: Long Jump data summary

YearCode	Year	dist
Min. :-4.00	Min. :1896	Min. :249.8
1st Qu.:21.00	1st Qu.:1921	1st Qu.:295.4
Median :50.00	Median :1950	Median :308.1
Mean :45.45	Mean :1945	Mean :310.3
3rd Qu.:71.00	3rd Qu.:1971	3rd Qu.:327.5
Max. :92.00	Max. :1992	Max. :350.5

Part C. Brain vs Body data

```r
url<-"http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BrainandBodyWeight.dat"
    BrainBody_raw<-read.table(url, header=F, skip=1, fill=T, stringsAsFactors = F)
    colnames(BrainBody_raw)<-rep(c("Brain","Body"),3)
    BrainBody_tidy<-rbind(BrainBody_raw[,1:2],BrainBody_raw[,3:4],
                             BrainBody_raw[,5:6])
    BrainBody_tidy<-BrainBody_tidy %>%
        filter(!(is.na(Brain)))
```

```r
knitr::kable(summary(BrainBody_tidy), caption="Brain/Body weight data summary")
```

Table 3: Brain/Body weight data summary

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. Tomato data

```
url<-"http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/tomato.dat"
    Tomato_raw<-read.table(url, header=F, skip=2, fill=T, stringsAsFactors = F, comment.char = "")
    Tomato_tidy<-Tomato_raw %>%
        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") %>%
        mutate(C10000_3=gsub(",","",C10000_3)) %>%
        gather(Clone,value,C10000_1:C30000_3) %>%
        mutate(Variety=V1, Clone=gsub("C","",Clone)) %>%
        mutate(Variety=gsub("\\\\#"," ",Variety)) %>%
        separate(Clone,into = c("Clone","Replicate")) %>%
        select(-V1,Variety,Clone,value) %>%
        arrange(Variety)
```

```
knitr::kable(summary(Tomato_tidy), caption="Tomato data summary")
```

Table 4: Tomato data summary

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

Problem 6

```
library(swirl)
```

```
##
## | Hi! I see that you have some variables saved in your workspace. To keep
## | things running smoothly, I recommend you clean up before starting swirl.
##
## | Type ls() to see a list of the variables in your workspace. Then, type
## | rm(list=ls()) to clear your workspace.
##
## | Type swirl() when you are ready to begin.
```

```
# Path to data
.datapath <- file.path(path.package('swirl'), 'Courses',
```

```r
                      'R_Programming_E', 'Looking_at_Data',
                      'plant-data.txt')
# Read in data
plants <- read.csv(.datapath, strip.white=TRUE, na.strings="")

# Remove annoying columns
.cols2rm <- c('Accepted.Symbol', 'Synonym.Symbol')
plants <- plants[, !(names(plants) %in% .cols2rm)]

# Make names pretty
names(plants) <- c('Scientific_Name', 'Duration', 'Active_Growth_Period',
                   'Foliage_Color', 'pH_Min', 'pH_Max',
                   'Precip_Min', 'Precip_Max',
                   'Shade_Tolerance', 'Temp_Min_F')

# Delete rows with NA value of Foliage_Color, pH_Min, pH_Max
plantsN<-plants[apply(plants,1,function(x)!any(is.na(x))),,drop=F]

# Select columns of Foliage_Color, pH_Min, pH_Max
plantsS<-select(plantsN,Foliage_Color,pH_Min,pH_Max)

# Linear Regression between Foliage_Color and pH_Median
plantsS<-plantsS %>%
mutate(pH_Median=(pH_Min+pH_Max)/2,Color=as.numeric(Foliage_Color))
lm<-lm(formula=Color~pH_Median,data=plantsS)
knitr::kable(summary(lm)$coefficients)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	2.5098534	0.3314904	7.571422	0.0000000
pH_Median	0.0581931	0.0535966	1.085761	0.2779075

```r
aov1<-aov(Color~pH_Median,data=plantsS)
summary(aov1)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## pH_Median     1    0.8  0.8084   1.179  0.278
## Residuals   811  556.1  0.6857
```

Problem 7

a. Load databasets

```r
# Load dataset into R
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```r
setwd("C:/Users/Echo/Downloads")
    Car_Gebreken_raw <- read.csv("Open_Data_RDW__Gebreken.csv",stringsAsFactors = F, nrows=200, header=
    Car_Geconstat_raw <- read.csv("Open_Data_RDW__Geconstateerde_Gebreken.csv", stringsAsFactors = F, n
    Car_Person_raw <- read.csv("Personenauto_basisdata.csv",stringsAsFactors = F, nrows=200, header=T)

    Car_Gebreken_raw.colclass <- sapply(Car_Gebreken_raw,class)
    Car_Geconstat_raw.colclass <- sapply(Car_Geconstat_raw,class)
    Car_Person_raw.colclass <- sapply(Car_Person_raw,class)

    print("Gebreken")
```

```
## [1] "Gebreken"
```

```r
    print(Car_Gebreken_raw.colclass)
```

```
##    Gebrek.identificatie     Ingangsdatum.gebrek        Einddatum.gebrek
##             "character"               "integer"               "integer"
## Gebrek.paragraaf.nummer   Gebrek.artikel.nummer     Gebrek.omschrijving
##               "integer"             "character"             "character"
```

```r
    print("Geconstat")
```

```
## [1] "Geconstat"
```

```r
    print(Car_Geconstat_raw.colclass)
```

```
##                          Kenteken Soort.erkenning.keuringsinstantie
##                       "character"                       "character"
## Meld.datum.door.keuringsinstantie  Meld.tijd.door.keuringsinstantie
##                         "integer"                         "integer"
##              Gebrek.identificatie      Soort.erkenning.omschrijving
##                       "character"                       "character"
##     Aantal.gebreken.geconstateerd
##                         "integer"
```

```r
    print("Personen")
```

```
## [1] "Personen"
```

```r
    print(Car_Person_raw.colclass)
```

```
##                          Kenteken                     Voertuigsoort
##                       "character"                       "character"
##                              Merk                   Handelsbenaming
##                       "character"                       "character"
##              Datum.tenaamstelling                         Bruto.BPM
##                       "character"                         "integer"
##                    Cilinderinhoud              Massa.ledig.voertuig
##                         "integer"                         "integer"
## Toegestane.maximum.massa.voertuig            Datum.eerste.toelating
##                         "integer"                       "character"
##    Datum.eerste.afgifte.Nederland                    Catalogusprijs
##                       "character"                         "integer"
##                     WAM.verzekerd
##                       "character"
```

b. Merge three tables

```
colnames(Car_Gebreken_raw)[1]<-'identification'
colnames(Car_Geconstat_raw)[5]<-'identification'
merge2<-merge(Car_Geconstat_raw,Car_Person_raw,by="Kenteken",all=TRUE)
merge3<-merge(Car_Gebreken_raw,merge2,by="identification",all=TRUE)
```

c. Clean the data and remove NA

```
# Took the first 7 columns which is useful to questions and shrink the dataset
mergec<-merge3[1:7]
colnames(mergec)<-c("defect_code","begin_date","end_date","make","model","defect_description","license
# Remove NA
mergec2<-mergec[apply(mergec,1,function(x)!any(is.na(x))),,drop=F]
```

d. How many DIFFERENT makes and models of cars

```
Mergedd<-filter(mergec2,end_date>=20170101)
makes <- n_distinct(Mergedd$make)
models <- n_distinct(Mergedd$model)
makes
```

```
## [1] 11
```

```
models
```

```
## [1] 27
```

e. 5 most frequent defects and make/models

```
library(sqldf)
```

```
## Loading required package: gsubfn
```

```
## Loading required package: proto
```

```
## Loading required package: RSQLite
```

```
test2<-sqldf("select defect_description,make, count (*) as count from Mergedd group by defect_descripti
test2
```

```
##                                             defect_description make count
## 1      Werking/toestand verplicht licht/retroreflector 5.*.55   10    31
## 2 Ruitenwisser/-sproeier werkt niet goed/niet aanwezig 5.*.43    9    18
## 3                             Band onvoldoende profiel 5.*.27    6    13
## 4  Uitlaatsysteem niet gasdicht/ondeugdelijk bevestigd 5.*.11    3    11
## 5                                  Afstelling dimlicht 5.*.56   10    10
```

f. Relationship between number of defects and make

```
test3<-sqldf("select make, count (*) as count from Mergedd group by defect_description ORDER BY count DE
lm2<-lm(count~make,data=test3)
knitr::kable(summary(lm2)$coefficients)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.8974987	2.5165679	0.356636	0.7231933
make	0.3960119	0.3407779	1.162082	0.2519223

```
aov2<-aov(formula = count ~ make, data = test3)
summary(aov2)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## make         1   41.7   41.70    1.35  0.252
## Residuals   41 1266.1   30.88
```

g. Relationship between number of defects and model

```
test4<-sqldf("select model, count (*) as count from Mergedd group by defect_description ORDER BY count
lm3<-lm(count~model,data=test4)
knitr::kable(summary(lm3)$coefficients)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	6.000000	1.910770	3.1400955	0.0063253
model5.*.16	-1.000000	3.309551	-0.3021558	0.7664256
model5.*.18	-4.000000	3.309551	-1.2086233	0.2443670
model5.*.19	-4.333333	2.466793	-1.7566665	0.0980922
model5.*.20	-3.000000	3.309551	-0.9064675	0.3781378
model5.*.26	-5.000000	3.309551	-1.5107791	0.1503413
model5.*.27	1.333333	2.466793	0.5405128	0.5962882
model5.*.28	-3.500000	2.702237	-1.2952233	0.2136207
model5.*.29	-5.000000	2.340206	-2.1365644	0.0484273
model5.*.3	-5.000000	2.702237	-1.8503190	0.0828219
model5.*.31	-4.000000	2.260854	-1.7692434	0.0959074
model5.*.37	-5.000000	3.309551	-1.5107791	0.1503413
model5.*.38	-3.500000	2.702237	-1.2952233	0.2136207
model5.*.39	-4.500000	2.702237	-1.6652871	0.1153113
model5.*.41	-5.000000	3.309551	-1.5107791	0.1503413
model5.*.42	-5.000000	3.309551	-1.5107791	0.1503413
model5.*.43	12.000000	3.309551	3.6258700	0.0022716
model5.*.44	-4.000000	3.309551	-1.2086233	0.2443670
model5.*.46	-4.000000	3.309551	-1.2086233	0.2443670
model5.*.51	-2.000000	3.309551	-0.6043117	0.5541097
model5.*.53	-3.000000	3.309551	-0.9064675	0.3781378
model5.*.55	25.000000	3.309551	7.5538957	0.0000012
model5.*.56	4.000000	3.309551	1.2086233	0.2443670
model5.*.69	-5.000000	3.309551	-1.5107791	0.1503413
model5.*.71	-4.000000	3.309551	-1.2086233	0.2443670
model5.*.9	-5.000000	3.309551	-1.5107791	0.1503413
model5.5.31	-5.000000	3.309551	-1.5107791	0.1503413

```
aov3<-aov(formula = count ~ model, data = test4)
summary(aov3)
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## model        26 1190.9   45.81   6.273 0.000191 ***
## Residuals    16  116.8    7.30
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

h. this workflow can be more efficient by using more frequent and practila operators.