HW2_Lin

zhengzhi lin

2019.9.5

P3

For solo workers, version control helps them track every single step they made, and help them remember
every decision they've made. For group workers, version control simply makes cooperating easier for each
other, because by version control, they can know the progress on each one and the team can adjust to make
things more efficient.

P4

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.5.1
```

```
##
## 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
```

```r
url <- "https://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/Sensory.dat"
operator <- read.table(url,fill = TRUE)
op_dat <- as.matrix(operator[-c(1:2),])
for (i in 1:10) {
  t <- op_dat[3*i-1,1:5]
  t <- c(i,t)
  op_dat[3*i-1,] <- t
  m <- op_dat[3*i,1:5]
  m <- c(i,m)
  op_dat[3*i,] <- m

}
op_dat <- as.data.frame(op_dat)
names(op_dat) <- c('item','operator1','operator2','operator3','operator4','operator5')
op_dat[,2:6] <- op_dat[,2:6] %>% mutate_if(is.factor, as.character)
op_dat[,2:6] <- op_dat[,2:6] %>% mutate_if(is.character, as.numeric)
op_dat
```

```
##      item operator1 operator2 operator3 operator4 operator5
## 3      1       4.3       4.9       3.3       5.3       4.4
## 4      1       4.3       4.5       4.0       5.5       3.3
## 5      1       4.1       5.3       3.4       5.7       4.7
## 6      2       6.0       5.3       4.5       5.9       4.7
## 7      2       4.9       6.3       4.2       5.5       4.9
## 8      2       6.0       5.9       4.7       6.3       4.6
## 9      3       2.4       2.5       2.3       3.1       2.4
## 10     3       3.9       3.0       2.8       2.7       1.3
## 11     3       1.9       3.9       2.6       4.6       2.2
## 12     4       7.4       8.2       6.4       6.8       6.0
## 13     4       7.1       7.9       5.9       7.3       6.1
## 14     4       6.4       7.1       6.9       7.0       6.7
## 15     5       5.7       6.3       5.4       6.1       5.9
## 16     5       5.8       5.7       5.4       6.2       6.5
## 17     5       5.8       6.0       6.1       7.0       4.9
## 18     6       2.2       2.4       1.7       3.4       1.7
## 19     6       3.0       1.8       2.1       4.0       1.7
## 20     6       2.1       3.3       1.1       3.3       2.1
## 21     7       1.2       1.5       1.2       0.9       0.7
## 22     7       1.3       2.4       0.8       1.2       1.3
## 23     7       0.9       3.1       1.1       1.9       1.6
## 24     8       4.2       4.8       4.5       4.6       3.2
## 25     8       3.0       4.5       4.7       4.9       4.6
## 26     8       4.8       4.8       4.7       4.8       4.3
## 27     9       8.0       8.6       9.0       9.4       8.8
## 28     9       9.0       7.7       6.7       9.0       7.9
## 29     9       8.9       9.2       8.1       9.1       7.6
## 30    10       5.0       4.8       3.9       5.5       3.8
## 31    10       5.4       5.0       3.4       4.9       4.6
## 32    10       2.8       5.2       4.1       3.9       5.5
```

```r
mean_op_table <- op_dat %>%
  group_by(item) %>%
  summarize(
    mean1 = mean(operator1),
    mean2 = mean(operator2),
    mean3 = mean(operator3),
    mean4 = mean(operator4),
    mean5 = mean(operator5)
  )


url <- "https://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/LongJumpData.dat"
olympic <- read.table(url,fill = TRUE)
olympic <- as.matrix(olympic)
oly_dat <- rbind(olympic[-1,1:2],olympic[-1,3:4],olympic[-1,5:6],olympic[-1,7:8])
oly_dat <- as.data.frame(oly_dat)
names(oly_dat) <- c("year","Long Jump")
#oly_table <- oly_dat %>% summarize(mean=mean(`Long Jump`),var=var(`Long Jump`))
oly_dat <- oly_dat[-c(23,24),]
oly_dat <- oly_dat %>% mutate_if(is.factor,as.character) %>% mutate_if(is.character,as.numeric)
oly_dat
```

```
##    year Long Jump
## 1    -4    249.75
## 2     0    282.88
## 3     4    289.00
## 4     8    294.50
## 5    12    299.25
## 6    20    281.50
## 7    24    293.13
## 8    28    304.75
## 9    32    300.75
## 10   36    317.31
## 11   48    308.00
## 12   52    298.00
## 13   56    308.25
## 14   60    319.75
## 15   64    317.75
## 16   68    350.50
## 17   72    324.50
## 18   76    328.50
## 19   80    336.25
## 20   84    336.25
## 21   88    343.25
## 22   92    342.50
```

```r
url <- 'https://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BrainandBodyWeight.dat'
weight <- read.table(url,fill = TRUE)
weight <- as.matrix(weight[-1,1:6])
weight_dat <- rbind(weight[,1:2],weight[,3:4],weight[,5:6])
weight_dat <- as.data.frame(weight_dat[-nrow(weight_dat),])
names(weight_dat) <- c("Body Wt","Brain Wt")
weight_dat <- weight_dat %>% mutate_if(is.factor,as.character) %>% mutate_if(is.character,as.numeric)
weight_dat
```

```
##       Body Wt Brain Wt
## 1       3.385    44.50
## 2       0.480    15.50
## 3       1.350     8.10
## 4     465.000   423.00
## 5      36.330   119.50
## 6      27.660   115.00
## 7      14.830    98.20
## 8       1.040     5.50
## 9       4.190    58.00
## 10      0.425     6.40
## 11      0.101     4.00
## 12      0.920     5.70
## 13      1.000     6.60
## 14      0.005     0.10
## 15      0.060     1.00
## 16      3.500    10.80
## 17      2.000    12.30
## 18      1.700     6.30
## 19   2547.000  4603.00
## 20      0.023     0.30
```

```
## 21  187.100   419.00
## 22  521.000   655.00
## 23    0.785     3.50
## 24   10.000   115.00
## 25    3.300    25.60
## 26    0.200     5.00
## 27    1.410    17.50
## 28  529.000   680.00
## 29  207.000   406.00
## 30   85.000   325.00
## 31    0.750    12.30
## 32   62.000  1320.00
## 33 6654.000  5712.00
## 34    3.500     3.90
## 35    6.800   179.00
## 36   35.000    56.00
## 37    4.050    17.00
## 38    0.120     1.00
## 39    0.023     0.40
## 40    0.010     0.30
## 41    1.400    12.50
## 42  250.000   490.00
## 43    2.500    12.10
## 44   55.500   175.00
## 45  100.000   157.00
## 46   52.160   440.00
## 47   10.550   179.50
## 48    0.550     2.40
## 49   60.000    81.00
## 50    3.600    21.00
## 51    4.288    39.20
## 52    0.280     1.90
## 53    0.075     1.20
## 54    0.122     3.00
## 55    0.048     0.33
## 56  192.000   180.00
## 57    3.000    25.00
## 58  160.000   169.00
## 59    0.900     2.60
## 60    1.620    11.40
## 61    0.104     2.50
## 62    4.235    50.40
```

```r
url <- 'https://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/tomato.dat'
tomato <- read.csv(url,sep = '\t')
tomato < - tomato %>% mutate_if(is.factor,as.character)
```

```
## Warning in Ops.factor(left): '-' not meaningful for factors
```

```
## Warning in Ops.factor(left, right): '<' not meaningful for factors
```

```
##      X.this.needs.reformatting.to.read.into.Splus
## [1,]                                            NA
```

```
## [2,]                                         NA
## [3,]                                         NA
```

```r
size <- paste(tomato[1,])
size <- strsplit(size," ")
size <- size[[1]][size[[1]]!=""]
size <- as.numeric(size[1:3])
size <- as.vector(size)

ife <- paste(tomato[2,])
ife <- strsplit(ife,' ')
ife <- ife[[1]][ife[[1]]!=""]
ife[2:4] <- strsplit(ife[2:4],',')
name <- as.matrix(rep(ife[[1]],3))
ife <- rbind(ife[[2]],ife[[3]],ife[[4]])
ife <- ife %>%
  cbind(name) %>%
  cbind(c(1000,2000,3000))

pusa <- paste(tomato[3,])
pusa <- strsplit(pusa,' ')
pusa <- pusa[[1]][pusa[[1]]!=""]
pusa[2:4] <- strsplit(pusa[2:4],',')
name <- as.matrix(rep(pusa[[1]],3))
pusa <- rbind(pusa[[2]],pusa[[3]],pusa[[4]])
pusa <- pusa %>%
  cbind(name) %>%
  cbind(c(1000,2000,3000))
tomato_dat <- rbind(pusa,ife)
tomato_dat <- tomato_dat %>% as.data.frame()
colnames(tomato_dat) <- c('1st','2nd','3rd','name','plant density')
tomato_dat <- tomato_dat[,c("name","plant density",'1st','2nd','3rd')]

tomato_dat[,2:5] <- tomato_dat[,2:5] %>%
  mutate_if(is.factor,as.character) %>%
  mutate_if(is.character,as.numeric)

tomato_dat
```

```
##              name plant density  1st  2nd  3rd
## 1 PusaEarlyDwarf          1000  8.1  8.6 10.1
## 2 PusaEarlyDwarf          2000 12.7 13.7 11.5
## 3 PusaEarlyDwarf          3000 14.4 15.4 13.7
## 4         Ife\\#1          1000 16.1 15.3 17.5
## 5         Ife\\#1          2000 16.6 19.2 18.5
## 6         Ife\\#1          3000 20.8 18.0 21.0
```