Bios 6301: Assignment 6

Jiaheng Yu

Due Tuesday, 24 October, 1:00 PM

$5^{n=day}$ points taken off for each day late.

40 points total. Submit a single knitr file (named `homework6.rmd`), along with a valid PDF output file. Inside the file, clearly indicate which parts of your responses go with which problems (you may use the original homework document as a template). Add your name as `author` to the file's metadata section. Raw R code/output or word processor files are not acceptable.

Failure to name file `homework6.rmd` or include author name may result in 5 points taken off.

Question 1

16 points

Obtain a copy of the football-values lecture. Save the five 2021 CSV files in your working directory.

Modify the code to create a function. This function will create dollar values given information (as arguments) about a league setup. It will return a data.frame and write this data.frame to a CSV file. The final data.frame should contain the columns 'PlayerName', 'pos', 'points', 'value' and be ordered by value descendingly. Do not round dollar values.

Note that the returned data.frame should have `sum(posReq)*nTeams` rows.

Define the function as such (10 points):

```
# path: directory path to input files
# file: name of the output file; it should be written to path
# nTeams: number of teams in league
# cap: money available to each team
# posReq: number of starters for each position
# points: point allocation for each category

# original values, nTeams=12, cap=200

ffvalues <- function(path, file='outfile.csv', nTeams=12, cap=200, posReq=c(qb=1, rb=2, wr=3, te=1, k=1
                     points=c(fg=4, xpt=1, pass_yds=1/25, pass_tds=4, pass_ints=-2,
                              rush_yds=1/10, rush_tds=6, fumbles=-2, rec_yds=1/20, rec_tds=6)) {
  ## read in CSV files
  k <- read.csv('proj_k21.csv', header=TRUE, stringsAsFactors=FALSE)
  qb <- read.csv('proj_qb21.csv', stringsAsFactors=FALSE)
  rb <- read.csv('proj_rb21.csv')
  te <- read.csv('proj_te21.csv')
  wr <- read.csv('proj_wr21.csv')

  # what columns are present in these files?
  cols <- unique(c(names(k), names(qb), names(rb), names(te), names(wr)))
```

```
# add a column specifying the position (aka the source file)
k[,'pos'] <- 'k'
qb[,'pos'] <- 'qb'
rb[,'pos'] <- 'rb'
te[,'pos'] <- 'te'
wr[,'pos'] <- 'wr'
cols <- c(cols, 'pos')

# adds addl columns not found in each file, with values set to 0
k[,setdiff(cols, names(k))] <- 0
qb[,setdiff(cols, names(qb))] <- 0
rb[,setdiff(cols, names(rb))] <- 0
te[,setdiff(cols, names(te))] <- 0
wr[,setdiff(cols, names(wr))] <- 0

# combine data.frames by row, using consistent column order
df <- rbind(k[,cols], qb[,cols], rb[,cols], te[,cols], wr[,cols])

## calculate dollar values

# first calculate fantasy points, point values specified in the function arguments
df[,'p_fg'] <- df[,'fg']*points['fg']
df[,'p_xpt'] <- df[,'xpt']*points['xpt']
df[,'p_pass_yds'] <- df[,'pass_yds']*points['pass_yds']
df[,'p_pass_tds'] <- df[,'pass_tds']*points['pass_tds']
df[,'p_pass_ints'] <- df[,'pass_ints']*points['pass_ints']
df[,'p_rush_yds'] <- df[,'rush_yds']*points['rush_yds']
df[,'p_rush_tds'] <- df[,'rush_tds']*points['rush_tds']
df[,'p_fumbles'] <- df[,'fumbles']*points['fumbles']
df[,'p_rec_yds'] <- df[,'rec_yds']*points['rec_yds']
df[,'p_rec_tds'] <- df[,'rec_tds']*points['rec_tds']

# total fantasy points for each player, sum of point columns
df[,'points'] <- rowSums(df[,grep("^p_", names(df))])

# now can convert to dollar values

df2 <- df[order(df[,'points'], decreasing=TRUE),]

k.ix <- which(df2[,'pos']=='k')
qb.ix <- which(df2[,'pos']=='qb')
rb.ix <- which(df2[,'pos']=='rb')
te.ix <- which(df2[,'pos']=='te')
wr.ix <- which(df2[,'pos']=='wr')

# calculate marginal points by subtracting "baseline" player's points
# every 12th-best player's marginal points = 0
# posReq=c(qb=1, rb=2, wr=3, te=1, k=1)

if(posReq['k']==0) {
df2[k.ix, 'marg'] <- -1 # just the kickers for now
df2[qb.ix, 'marg'] <- df2[qb.ix,'points'] - df2[qb.ix[nTeams*posReq['qb']],'points']
df2[rb.ix, 'marg'] <- df2[rb.ix,'points'] - df2[rb.ix[nTeams*posReq['rb']],'points']
```

```
  df2[te.ix, 'marg'] <- df2[te.ix,'points'] - df2[te.ix[nTeams*posReq['te']],'points']
  df2[wr.ix, 'marg'] <- df2[wr.ix,'points'] - df2[wr.ix[nTeams*posReq['wr']],'points']
} else {
  df2[k.ix, 'marg'] <- df2[k.ix,'points'] - df2[k.ix[nTeams*posReq['k']],'points']
  df2[qb.ix, 'marg'] <- df2[qb.ix,'points'] - df2[qb.ix[nTeams*posReq['qb']],'points']
  df2[rb.ix, 'marg'] <- df2[rb.ix,'points'] - df2[rb.ix[nTeams*posReq['rb']],'points']
  df2[te.ix, 'marg'] <- df2[te.ix,'points'] - df2[te.ix[nTeams*posReq['te']],'points']
  df2[wr.ix, 'marg'] <- df2[wr.ix,'points'] - df2[wr.ix[nTeams*posReq['wr']],'points']
}

  # create a new data.frame subset by non-negative marginal points
  df3 <- df2[df2[,'marg'] >= 0,]

  # re-order by marginal points
  df3 <- df3[order(df3[,'marg'], decreasing=TRUE),]

  rownames(df3) <- NULL

  # calculation for player value
  df3[,'value'] <- (nTeams*cap-nrow(df3)) * df3[,'marg'] / sum(df3[,'marg']) + 1

  # create a data.frame with more interesting columns
  df4 <- df3[,c('PlayerName','pos','points','value')]

  ## save dollar values as CSV file
  # columns for new file = 'PlayerName', 'pos', 'points', 'value'
  write.csv(df4, file=file)

  ## return data.frame with dollar values
  df4
  #nrow(sum(posReq)*nTeams)
}
```

1. Call `x1 <- ffvalues('.')`

```
x1 <- ffvalues('.')
```

```
## Warning in file(file, "rt"): cannot open file 'proj_k21.csv': No such file or
## directory
```

```
## Error in file(file, "rt"): cannot open the connection
```

```
# nrow(x1) # = 96
# sum(posReq)*nTeams) = 8 * 12 = 96
# row number matches what it's supposed to be
```

   a. How many players are worth more than $20? (1 point)

```
length(x1['PlayerName'][which(x1['value'] > 20),]) # 41 players
```

```
## Error in eval(expr, envir, enclos): object 'x1' not found
```

b. Who is 15th most valuable running back (rb)? (1 point)

```
x1['PlayerName'][which(x1[,'pos']=='rb')[15],] # David Montgomery
```

```
## Error in eval(expr, envir, enclos): object 'x1' not found
```

2. Call `x2 <- ffvalues(getwd(), '16team.csv', nTeams=16, cap=150)`

```
x2 <- ffvalues(getwd(), '16team.csv', nTeams=16, cap=150)
```

```
## Warning in file(file, "rt"): cannot open file 'proj_k21.csv': No such file or
## directory
```

```
## Error in file(file, "rt"): cannot open the connection
```

```
#sum(posReq)*nTeams) = 8 * 16 = 128 rows
```

a. How many players are worth more than $20? (1 point)

```
length(x2['PlayerName'][which(x2['value'] > 20),]) # 46 players
```

```
## Error in eval(expr, envir, enclos): object 'x2' not found
```

b. How many wide receivers (wr) are in the top 40? (1 point)

```
nrow(x2[which(x2[1:40,]['pos']=='wr'),]) # 8 wide receivers
```

```
## Error in eval(expr, envir, enclos): object 'x2' not found
```

3. Call:

```
x3 <- ffvalues('.', 'qbheavy.csv', posReq=c(qb=2, rb=2, wr=3, te=1, k=0),
        points=c(fg=0, xpt=0, pass_yds=1/25, pass_tds=6, pass_ints=-2,
                rush_yds=1/10, rush_tds=6, fumbles=-2, rec_yds=1/20, rec_tds=6))
```

```
## Warning in file(file, "rt"): cannot open file 'proj_k21.csv': No such file or
## directory
```

```
## Error in file(file, "rt"): cannot open the connection
```

a. How many players are worth more than $20? (1 point)

```
length(x3['PlayerName'][which(x3['value'] > 20),]) # 43 players
```

```
## Error in eval(expr, envir, enclos): object 'x3' not found
```

b. How many quarterbacks (qb) are in the top 30? (1 point)

```
nrow(x3[which(x3[1:30,]['pos']=='qb'),]) # 13 quarterbacks
```

```
## Error in eval(expr, envir, enclos): object 'x3' not found
```

Question 2

24 points

Import the HAART dataset (`haart.csv`) from the GitHub repository into R, and perform the following manipulations: (4 points each)

```
haart <- read.table("/Users/KatietheWise/Desktop/2022_Fall/StatComp/Homework/haart.csv", header=TRUE, s
```

```
## Warning in file(file, "rt"): cannot open file
## '/Users/KatietheWise/Desktop/2022_Fall/StatComp/Homework/haart.csv': No such
## file or directory
```

```
## Error in file(file, "rt"): cannot open the connection
```

1. Convert date columns into a usable (for analysis) format. Use the `table` command to display the counts of the year from `init.date`.

```
haart[,'last.visit'] <- as.POSIXct(haart[,'last.visit'], format="%m/%d/%y")
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```
haart[,'init.date'] <- as.POSIXct(haart[,'init.date'], format="%m/%d/%y")
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```
haart[,'date.death'] <- as.POSIXct(haart[,'date.death'], format="%m/%d/%y")
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```
# make a new column with the year
haart['init_year'] <- strftime(haart[,'init.date'], "%Y")
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```
table(haart['init_year'])
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

2. Create an indicator variable (one which takes the values 0 or 1 only) to represent death within 1 year of the initial visit. How many observations died in year 1?

```
#death_indic <-
# time difference, either 1 year or 12 months

death_indic <-c()
for (i in 1:(nrow(haart))){
  if (!is.na(haart[,'date.death'][i])){ # there's a death date
    if (difftime(haart['date.death'][i,],haart['init.date'][i,], units='days') <= 365.25){
```

```
      death_indic[i] <- 1 # died within a year of initial visit
    } else {
      death_indic[i] <- 0 # died, but more than a year after initial visit
      }
  } else if (is.na(haart[,'date.death'][i])){ # there's no death date
      death_indic[i] <- 0
  }
}
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```
sum(death_indic) # 92 deaths occurred within the first year following the initial visit
```

```
## [1] 0
```

```
# append to the dataframe
haart[,'death_indic'] <- death_indic
```

```
## Error: object 'haart' not found
```

3. Use the `init.date`, `last.visit` and `death.date` columns to calculate a followup time (in days), which is the difference between the first and either the last visit or a death event (whichever comes first). If these times are longer than 1 year, censor them (this means if the value is above 365, set followup to 365). Print the quantile for this new variable.

```
#haart

follow_up <- c()
for (i in 1:(nrow(haart))){
  if (!is.na(haart['date.death'][i,] > haart['last.visit'][i,])){ # both values present, death date is
    difference <- difftime(haart['date.death'][i,],haart['init.date'][i,], units='days')
    if (difference > 365) {
      difference <- 365
    }
    follow_up[i] <- difference
  } else if ((is.na(haart['date.death'][i,]) && (!is.na(haart['last.visit'][i,])))){ # no death date, o
    difference <- difftime(haart['last.visit'][i,],haart['init.date'][i,], units='days')
    if (difference > 365) {
      difference <- 365
    }
    follow_up[i] <- difference
  } else if ((is.na(haart['last.visit'][i,]) && (!is.na(haart['date.death'][i,])))){ # no last visit, o
    difference <- difftime(haart['date.death'][i,],haart['init.date'][i,], units='days')
    if (difference > 365) {
      difference <- 365
    } else if ((is.na(haart['last.visit'][i,]) && (is.na(haart['date.death'][i,])))) { # both NA
      difference <- NA
    }
    follow_up[i] <- difference
  }
}
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```
follow_up
```

```
## NULL
```

```
quantile(follow_up) # usually it's more than a year between initial visit and last visit / death
```

```
##   0%  25%  50%  75% 100%
##   NA   NA   NA   NA   NA
```

4. Create another indicator variable representing loss to followup; this means the observation is not known to be dead but does not have any followup visits after the first year. How many records are lost-to-followup?

```
lost <- c()
for (i in 1:(nrow(haart))){
  if ((is.na(haart['date.death'][i,]) && (!is.na(haart['last.visit'][i,])))){ # no death date, only las
    difference <- difftime(haart['last.visit'][i,],haart['init.date'][i,], units='days')
    if (difference < 365) {
      lost[i] <- 0 # followed up within the first year but not after, lost to follow-up
    } else if (difference > 365) {
      lost[i] <- 1 # followed up after the first year
    }
  } else {
    lost[i] <- 1 # they did follow-up at some point
  }
}
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```
lost # 0 = lost to follow-up
```

```
## NULL
```

```
length(which(lost == 0)) # 173
```

```
## [1] 0
```

```
# append to dataframe

haart[,'loss_to_followup'] <- lost
```

```
## Error: object 'haart' not found
```

5. Recall our work in class, which separated the `init.reg` field into a set of indicator variables, one for each unique drug. Create these fields and append them to the database as new columns. Which drug regimen are found over 100 times?

```r
#haart[,'init.reg']

# create indicator variables

#head(haart[,'init.reg'])
#table(haart[,'init.reg'])

init.reg <- as.character(haart[,'init.reg'])
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```r
(haart[['init.reg_list']] <- strsplit(init.reg, ","))
```

```
## Error in eval(expr, envir, enclos): object 'init.reg' not found
```

```r
unlist(haart$init.reg_list)[seq(50)]
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```r
(all_drugs <- unique(unlist(haart$init.reg_list))) # all unique drugs
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```r
reg_drugs <- matrix(FALSE, nrow=nrow(haart), ncol=length(all_drugs))
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```r
for(i in seq_along(all_drugs)) {
  reg_drugs[,i] <- sapply(haart$init.reg_list, function(x) all_drugs[i] %in% x)
}
```

```
## Error in eval(expr, envir, enclos): object 'all_drugs' not found
```

```r
reg_drugs <- data.frame(reg_drugs)
```

```
## Error in eval(expr, envir, enclos): object 'reg_drugs' not found
```

```r
names(reg_drugs) <- all_drugs
```

```
## Error in eval(expr, envir, enclos): object 'all_drugs' not found
```

```r
# append to the haart database as new columns
haart_merged <- cbind(haart, reg_drugs)
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```
# which combination(s) are found over 100 times?

haart[,'init.reg'] <- as.factor(haart[,'init.reg'])
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

```
#str(haart[,'init.reg']) #47 different combinations
#table(haart[,'init.reg'])
which(table(haart[,'init.reg']) > 100) # 3TC,AZT,EFV & 3TC,AZT,NVP
```

```
## Error in eval(expr, envir, enclos): object 'haart' not found
```

6. The dataset `haart2.csv` contains a few additional observations for the same study. Import these and append them to your master dataset (if you were smart about how you coded the previous steps, cleaning the additional observations should be easy!). Show the first five records and the last five records of the complete (and clean) data set.

```
haart2 <- read.table("/Users/KatietheWise/Desktop/2022_Fall/StatComp/Homework/haart2.csv", header=TRUE,
```

```
## Warning in file(file, "rt"): cannot open file
## '/Users/KatietheWise/Desktop/2022_Fall/StatComp/Homework/haart2.csv': No such
## file or directory
```

```
## Error in file(file, "rt"): cannot open the connection
```

```
# convert to usable date formats
haart2[,'last.visit'] <- as.POSIXct(haart2[,'last.visit'], format="%m/%d/%y")
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
haart2[,'init.date'] <- as.POSIXct(haart2[,'init.date'], format="%m/%d/%y")
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
haart2[,'date.death'] <- as.POSIXct(haart2[,'date.death'], format="%m/%d/%y")
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
# make a new column with the year
haart2['init_year'] <- strftime(haart2[,'init.date'], "%Y")
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
# death indicator
death_indic2 <-c()
for (i in 1:(nrow(haart2))){
  if (!is.na(haart2[,'date.death'][i])){ # there's a death date
    if (difftime(haart2['date.death'][i,],haart2['init.date'][i,], units='days') <= 365.25){
      death_indic2[i] <- 1
    } else {
      death_indic2[i] <- 0
      }
  } else if (is.na(haart2[,'date.death'][i])){
      death_indic2[i] <- 0
  }
}
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
# append to the dataframe
haart2[,'death_indic'] <- death_indic2
```

```
## Error: object 'haart2' not found
```

```
# loss to followup
lost2 <- c()
for (i in 1:(nrow(haart2))){
  if ((is.na(haart2['date.death'][i,]) && (!is.na(haart2['last.visit'][i,])))){ # no death date, only l
    difference <- difftime(haart2['last.visit'][i,],haart2['init.date'][i,], units='days')
    if (difference < 365) {
      lost2[i] <- 0 # followed up within the first year but not after, lost to follow-up
    } else if (difference > 365) {
      lost2[i] <- 1 # followed up after the first year
    }
  } else {
    lost2[i] <- 1 # they did follow-up at some point
  }
}
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
# append to dataframe
haart2[,'loss_to_followup'] <- lost2
```

```
## Error: object 'haart2' not found
```

```
# drug combos
init.reg2 <- as.character(haart2[,'init.reg'])
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
(haart2[['init.reg_list']] <- strsplit(init.reg2, ","))
```

```
## Error in eval(expr, envir, enclos): object 'init.reg2' not found
```

```
unlist(haart2$init.reg_list)[seq(50)]
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
(all_drugs2 <- unique(unlist(haart2$init.reg_list))) # all unique drugs
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
reg_drugs2 <- matrix(FALSE, nrow=nrow(haart2), ncol=length(all_drugs2))
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
for(i in seq_along(all_drugs2)) {
  reg_drugs2[,i] <- sapply(haart2$init.reg_list, function(x) all_drugs2[i] %in% x)
}
```

```
## Error in eval(expr, envir, enclos): object 'all_drugs2' not found
```

```
reg_drugs2 <- data.frame(reg_drugs2)
```

```
## Error in eval(expr, envir, enclos): object 'reg_drugs2' not found
```

```
names(reg_drugs2) <- all_drugs2
```

```
## Error in eval(expr, envir, enclos): object 'all_drugs2' not found
```

```
head(reg_drugs2)
```

```
## Error in eval(expr, envir, enclos): object 'reg_drugs2' not found
```

```
# append to the haart database as new columns
#haart2_merged <- cbind(haart2, reg_drugs2)

# noticed an issue here with empty headers
orig_names <- c(names(reg_drugs))
```

```
## Error in eval(expr, envir, enclos): object 'reg_drugs' not found
```

```
add_names <- c(names(reg_drugs2))
```

```
## Error in eval(expr, envir, enclos): object 'reg_drugs2' not found
```

```
to_add <- c(setdiff(orig_names, add_names))
```

```
## Error in eval(expr, envir, enclos): object 'orig_names' not found
```

```
# this part is overly complicated but I ran out of ideas
df1 <- data.frame('ABC' = c(FALSE, FALSE, FALSE, FALSE),
                  'IDV' = c(FALSE, FALSE, FALSE, FALSE),
                  'LPV' = c(FALSE, FALSE, FALSE, FALSE),
                  'RTV' = c(FALSE, FALSE, FALSE, FALSE),
                  'SQV' = c(FALSE, FALSE, FALSE, FALSE),
                  'FTC' = c(FALSE, FALSE, FALSE, FALSE),
                  'TDF' = c(FALSE, FALSE, FALSE, FALSE),
                  'DDC' = c(FALSE, FALSE, FALSE, FALSE),
                  'NFV' = c(FALSE, FALSE, FALSE, FALSE),
                  'T20' = c(FALSE, FALSE, FALSE, FALSE),
                  'ATV' = c(FALSE, FALSE, FALSE, FALSE),
                  'FPV' = c(FALSE, FALSE, FALSE, FALSE)
)

new_df <- cbind(reg_drugs2, df1)
```

```
## Error in eval(expr, envir, enclos): object 'reg_drugs2' not found
```

```
haart2_merged <- cbind(haart2, new_df)
```

```
## Error in eval(expr, envir, enclos): object 'haart2' not found
```

```
# not in 100% the right order, but rbind should take care of that
# now merge haart_merged and haart2_merged
haart_final <- rbind(haart_merged, haart2_merged)
```

```
## Error in eval(expr, envir, enclos): object 'haart_merged' not found
```

```
# first five records
haart_final[1:5,]
```

```
## Error in eval(expr, envir, enclos): object 'haart_final' not found
```

```
# last five records
haart_final[1000:1004,]
```

```
## Error in eval(expr, envir, enclos): object 'haart_final' not found
```