Workouts

Kayla Catalfamo

2017-10-10

Description

This dataset was made based on my active weekly workouts. The workouts occurred four - five times a week focusing on different body parts of the body. Every workout had four sets containing ten reps. Each rep may have contained different weights (lbs) due to the fact that I like to increase the weight with each rep. So the weight (lbs) that were recorded in the dataset is the average weights used overall in each workout. The variables of the dataset are as followed;

```
Day - "The actual date that the workout was performed"
Week - "The week # in the year"
Workout Day - "What body category was focused on"
# of Workouts - "The amount of different exercises that were done"
Cardio - "Total number of calories that are burned in a cardio session" (calories)
Weight - "The amount of pounds I weighed for that day" (lbs)
```

The other variables are the types of exercises that were done at each workout day.

Reading in the Data

The packages that are required to run the code and functions are;

```
"googlesheets"
"devtools"
"plyr"
"reshape2"
"ggplot2"
```

This grouping of r code will run the data into R from the google drive and make it a data frame.

```
gs_ls()
```

```
## Auto-refreshing stale OAuth token.
```

```
## # A tibble: 1 x 10
##                 sheet_title        author  perm version             updated
##                       <chr>         <chr> <chr>   <chr>              <dttm>
## 1                 Workouts k.m.catalfam<U+0085>    rw     new 2017-10-10 05:21:00
## # ... with 5 more variables: sheet_key <chr>, ws_feed <chr>,
## #   alternate <chr>, self <chr>, alt_key <chr>
```

```
workout <- gs_title("Workouts")
```

```
## Sheet successfully identified: "Workouts"
```

```
workouts <- gs_read(ss=workout, wo = "Workouts for Each Day", header = T)
```

```
## Accessing worksheet titled 'Sheet1'.
```

```
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   Day = col_character(),
##   `Workout Day` = col_character(),
##   `Good Mornings` = col_character(),
##   `Cable Reverse Fly` = col_double(),
##   Weight = col_double()
## )
```

```
## See spec(...) for full column specifications.
```

```
workouts <- as.data.frame(workouts)

# workouts <- read.table("Workouts-Sheet1", header = TRUE)
```

Extra Coding

This section sets up coding to revise the dataframe in the meantime while more data is being collected throughout the weeks. Descriptions of what is being done is stated underneath the coding in comment form.

```
colnames(workouts) <- gsub(' ', '_', colnames(workouts))
  # puts _ where spaces are within the variable names


avg_cal <- ddply(workouts, .(Workout_Day), summarize,
                Avg_Calories_Burned = mean(Calories_Burned, na.rm = TRUE))
  # subset of the data just looking at the workout day and the average calories burned
    # for each of those workout days


workouts2 <- workouts[,-c(4:6, 33)]
  # subset the original data frame
  # this is getting rid of columns 4 through 6 and 33
  # this dataframe will be used when wanting to look at the average weights (lbs) for
    # each week to see the overall progress

#colnames(workouts2)[35] <- 'Body_Fat'
  # changes the column name of Body Fat %

workouts.melt <- melt(workouts2,
                      id.vars = c('Day', 'Workout_Day', 'Calories_Burned', 'Weight',
                                  'Week'))
  # the melt function is converting the data.frame workouts2 into a molten data frame
    # where the variables for Day, Workout Day, Calories Burned, Weight, Body Fat, and
    # Week into numeric id vectors.

workouts.final <- workouts.melt[-which(is.na(workouts.melt$value) == TRUE),]
  # remove na value-ed rows
head(workouts.final)
```

```
##           Day Workout_Day Calories_Burned Weight Week
## 1  9/12/2017         Leg             654    144   37
## 5  9/19/2017         Leg             448    145   38
## 8  9/26/2017         Leg             510    145   39
## 11 10/3/2017         Leg             561    145   40
## 14 9/12/2017         Leg             654    144   37
## 18 9/19/2017         Leg             448    145   38
##                               variable value
## 1  Barbell_Squats_(weights_lbs/per_set)    75
## 5  Barbell_Squats_(weights_lbs/per_set)   110
## 8  Barbell_Squats_(weights_lbs/per_set)   115
## 11 Barbell_Squats_(weights_lbs/per_set)   130
## 14                   Smith_Calf_Raises   150
## 18                   Smith_Calf_Raises   140
```

```
  # looks at the first couple rows of the new data.frame
```

```
## for ggplot looking at calories over time
workouts3 <- workouts2[, -c(4:32)]
workout.cal <- melt(workouts3,
                      id.vars = c('Day', 'Workout_Day', 'Calories_Burned', 'Weight',
                                  'Week'))
head(workout.cal)
```

```
##         Day    Workout_Day Calories_Burned Weight Week      variable value
## 1 9/12/2017            Leg             654  144.0   37 Assisted_Dips    NA
## 2 9/13/2017      Shoulders             366  145.2   37 Assisted_Dips    NA
## 3 9/14/2017           Arms             489  145.0   37 Assisted_Dips    NA
## 4 9/18/2017           Back             464  145.0   38 Assisted_Dips    NA
## 5 9/19/2017            Leg             448  145.0   38 Assisted_Dips    NA
## 6 9/21/2017 Shoulders/Arms             594  144.0   38 Assisted_Dips    NA
```

Plots

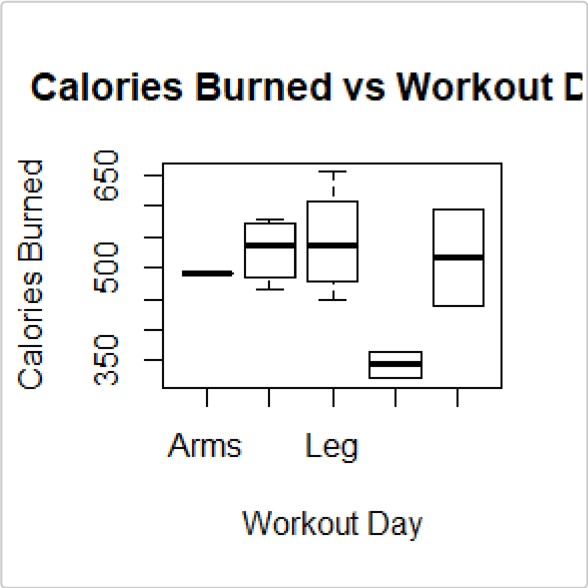
There are two plots depicting relationships within the data set of workouts. The function, workout_boxpt, for the boxplot allows the user to place the “x” and “y” values of their choice into the boxplot. For the example of the boxplot it is looking at the average amount of calories burned for each workout day. Everything in the workout_boxpt is generic enough where when the preferred “x” and “y” variables are chosen, the axis and boxplot title will depict those variables properly.

It is helpful that when creating the boxplot to have “x” as a categorical variable and “y” as numeric.

Boxplot

```
## Boxplot of the avg calories burned for each workout day

workout_boxpt <- function(workouts,x,y)
{
  x.vec <- workouts[[x]]
  y.vec <- workouts[[y]]
  boxplot(y.vec ~ x.vec, na.rm = TRUE,
          xlab = gsub("_", " ", x),
          ylab = gsub("_"," ", y),
          main = paste(gsub("_"," ", y), "vs", gsub("_", " ", x))
  )
}
workout_boxpt(workouts, "Workout_Day", "Calories_Burned")
```



Looking at calories burned for each workout day is desired because there should be a change over time. In each workout day, exercises and the number of exercises are picked at random. The exercises really depend on what I feel like doing that day. So the number of workouts done and the type of exercises will affect the outcome of the total calories burned for that workout day. The variability in calories burned for each workout day is interesting to see because you can see what days I like to favor more than the others. The favored workout day can be seen based on the distribution of the average amount of calories burned for each workout day.

In terms of the data so far, I like to favor “Leg” day. This is accurate with the boxplot because “Leg” day is my favorite workout day and I usually go more aggressively in those exercises than any other workout day. Thus from going more aggressively in those exercises I burn more calories. The amount of calories burned, like I mentioned above, will change based on the different exercises done for that workout day. I don’t like to do the same exercises over and over again, so I will add new workouts into the rotation of exercises. The exercises done in each workout day is really based on how “bored” or “motivated” I am.

It is also captivating to see the different range of calories burned for each exercise day. As of right now both “Back” and “Leg” workout days are accumulating the most calories burned in a workout session where “Shoulders” don’t burn as many calories. In terms of my body that makes sense because my legs and back are actually the strongest parts where my arms are average and my shoulders are the weakest. My back and legs are the strongest due to years of being a catcher in softball, jumper in track and field, and lifting on and off. My arms were stronger during my years of softball and track and field, but my durability has decreased just from lack of endurance on that muscle group. There are actually many reasons why my shoulders are the weakest, but the main reason is because tendinitis and from never working on those muscles during a workout.

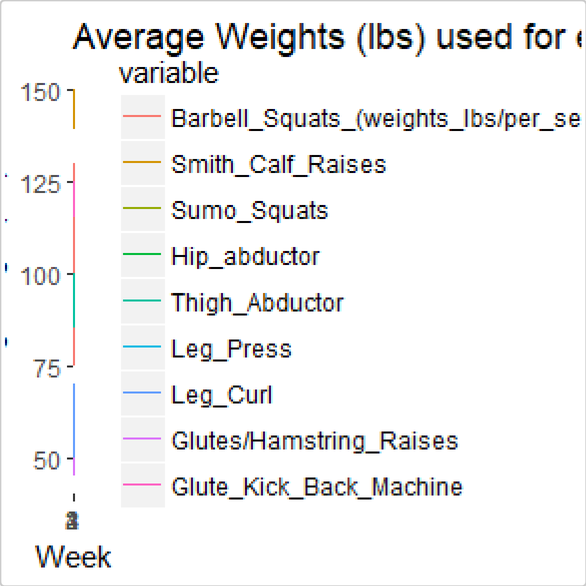
My endurance for each workout day should start to increase which means more weight will be needed or the number of reps will have to increase. This should show a steady increase in calories burned over time and demonstrate a larger distribution of calories burned in each workout day.

Aesthetics Plot

The function, avg_lbs_day, for the ggplot allows for the user to place the preferred data set and workout day to be used. This function will spit out a plot of the workout day wanted to view the progression of the average weights (lbs) being used in each exercise. The week variable is subtracted by 36 to show the 1st week of workouts rather than the 37th week in the year.

```
## function allows to plug in whatever workout day wanted to look at and
  ## create a graph depicting the workouts done for that day over the weeks

avg_lbs_day <- function(data, b)
{
  ggplot(subset(data, Workout_Day == b),
         aes(x = Week - 36, y = value)) +
    xlab("Week") + ylab(" Avg Weight (lbs)") +
    ggtitle("Average Weights (lbs) used for each exercise in Workout Day") +
    geom_line(aes(col = variable))
}
avg_lbs_day(workouts.final, "Leg") # look at whatever workout day wanted
```



In my point of view, I think the aesthetic plot is the most important depiction of the workouts. It allows me to have a visual of my progress in terms of weight lifting and the overall change in strength of the muscles that are being used in each exercise. There might be drops in the average number of weight with some exercises due to how sore, sick, or rushed I am. I have a lot of problems will my knees that can affect the amount of weight I decide to use. Tendinitis and fluid build-up are the two most concerning problems during leg day, but if they are really bad I lower the weight used so I don’t injure myself. It is very important to me to see the overall progress of my workouts because it shows how much I’ve accomplished.

Body Fat Calculations

The body_fat function computes the body fat percentage for a person. Body fat percentage is based on six things; weight in pounds, lean body mass, wrist, waist, hip, and forearm circumferences. The lean body mass needs to be computed before calculating the body fat percentage. The body_fat function also does that computation. All that needs to be read into the function is your weight, wrist, waist, hip, and forearm circumferences and the function will do the calculations. When every desired measurement is called into the function, it will print out what your lean body mass and body fat percentages are.

```
## Bodyfat calculations based on the NAVY body fat calculations

bodyfat <- function(w, wc, wac, hc, fc){
  weight <- w
  # in lbs
  wrist <- wc ; waist <- wac ; hip <- hc ; forearm <- fc
  # wrist, waist, hip, and forearm circumferences in inches
  wrist <- wc/3.14
  waist <- wac * 0.157
  hip <- hc * 0.249
  forearm <- fc * 0.434

  # lean body mass
  LBM <- (
    ((
      ((0.732 * weight) + 8.987) + wrist) - waist) - hip) + forearm

  # body_fat_percent
  fat_percent <- ((weight - LBM) * 100) / weight

  ## want I want to see
  LBM <- round(LBM, digits=2) ; fat_percent <- round(fat_percent, digits = 2)
  c(paste("LBM: ",LBM),
    paste("body fat %: ",fat_percent))
}
bodyfat(145, 6.0, 24, 38, 9.5)
```

```
## [1] "LBM:  107.93"       "body fat %:  25.56"
```

The main focus of this research is calculating my body fat percentage. I want to be able to compare my body fat percentage from the start of working out to my body fat percentage at the end of the semester. It’s hard to see the progress in the body weight tracked because there is a good probability that my weight will increase by the end of the semester. This can be because of an increase in fat and/or increase in muscle mass. Muscle weighs more than fat so as I continue to workout, my muscles are getting bigger and hopefully I’m decreasing the fat on my body. At the end of the semester this research will hopefuly show that I have decreased my body fat percentage.

This function allows me to quickly calculate the body fat percentage without having to find an accurate body fat percentage calculator on the internet, buy calpiers, nor do the calculations out by hand. It would be useful to document the body fat percentage after every few weeks and possibly depict that relationship in a plot or table later on.