Project1

Justin

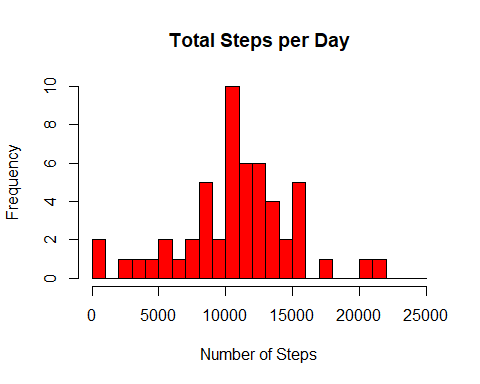
6/24/2020

# load csv   
activityData <- read.csv("activity.csv", header=TRUE)  
  
# load necessary libraries for later analysis   
library(ggplot2)

What is mean total number of steps taken per day? For this part of the assignment, you can ignore the missing values in the dataset.

Calculate the total number of steps taken per day If you do not understand the difference between a histogram and a barplot, research the difference between them. Make a histogram of the total number of steps taken each day Calculate and report the mean and median of the total number of steps taken per day

# Begin by calculating total # of steps taken per day and ignore NA  
StepsPerDay <- aggregate(steps ~ date, activityData, FUN=sum, na.rm = TRUE)  
  
# histogram of the total number of steps taken per day filled in with red  
hist(StepsPerDay$steps,  
 breaks = seq(from=0, to=25000, by=1000),  
 col = "red",  
 main = "Total Steps per Day",  
 xlab = "Number of Steps")

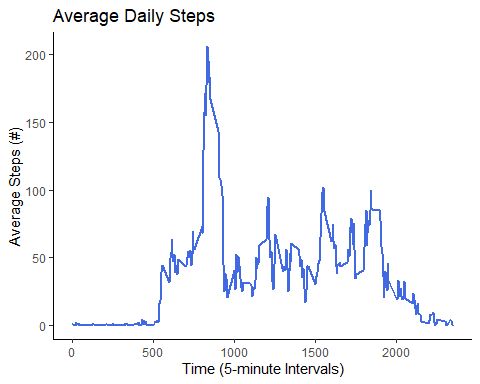


# Calculate and report the mean and median of total steps taken per day  
Steps\_mean <- mean(StepsPerDay$steps, na.rm = TRUE)  
Steps\_median <- median(StepsPerDay$steps, na.rm = TRUE)

Answer: mean is 10766.19 and median is 10765.

What is the average daily activity pattern? Make a time series plot (i.e. type = “l”) of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis) Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?

# This figure will be created using ggplot2 and with a classic theme so no gridlines will be behind the data line. The data line will be blue and x and y axes labeled accordingly along with a brief title.   
library(ggplot2)  
Mean\_StepsPer5min <- aggregate(steps ~ interval, activityData, mean)  
ggplot(data = Mean\_StepsPer5min, aes(x = interval, y = steps)) +  
 geom\_line(color="royalblue", size=1) +  
 ggtitle("Average Daily Steps") +  
 xlab("Time (5-minute Intervals)") +  
 ylab("Average Steps (#)") +  
 theme(plot.title = element\_text(hjust = 0.5))+  
 theme\_classic() +   
 scale\_colour\_hue(name = "Steps")



# This script will find the point where the max point is and return its value   
GreatestStepsPer5min <- Mean\_StepsPer5min[which.max(Mean\_StepsPer5min$steps),]

Answer: Greatest steps per 5 min was at interval 835 and was 206.17 steps. The average activity is large spikes early in the morning with a smaller spike late afternoon. Most activity on the early end of the day.

Next Question:

Imputing missing values Note that there are a number of days/intervals where there are missing values (coded as NA). The presence of missing days may introduce bias into some calculations or summaries of the data.

Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs)

Devise a strategy for filling in all of the missing values in the dataset. The strategy does not need to be sophisticated. For example, you could use the mean/median for that day, or the mean for that 5-minute interval, etc.

Create a new dataset that is equal to the original dataset but with the missing data filled in. Make a histogram of the total number of steps taken each day and Calculate and report the mean and median total number of steps taken per day. Do these values differ from the estimates from the first part of the assignment? What is the impact of imputing missing data on the estimates of the total daily number of steps?

# To fill we can replace the missing values with respective 5 minute interval averages. To do so we can use a for loop to find the missing points and then fill in with if statement which will address mean values to fill.   
activityData\_Fill <- activityData  
for(r in 1:nrow(activityData\_Fill)){  
 if (is.na(activityData\_Fill$steps[r])) {  
 repl <- Mean\_StepsPer5min$steps[Mean\_StepsPer5min$interval == activityData\_Fill$interval[r]];  
 activityData\_Fill$steps[r] <- repl;  
 }  
}  
# check to be sure the values are not reporting missing data and the code worked, can also check to be sure by heading data and looking at first few rows.   
colSums(is.na(activityData\_Fill))

## steps date interval   
## 0 0 0

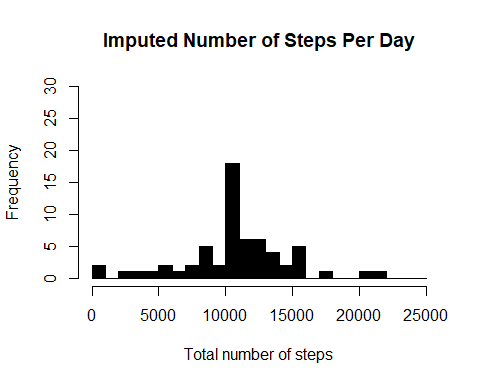
str(activityData\_Fill$steps)

## num [1:17568] 1.717 0.3396 0.1321 0.1509 0.0755 ...

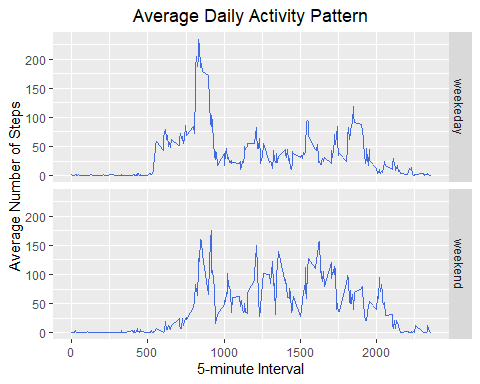
Are there differences in activity patterns between weekdays and weekends? For this part the weekdays() function may be of some help here. Use the dataset with the filled-in missing values for this part.

Create a new factor variable in the dataset with two levels – “weekday” and “weekend” indicating whether a given date is a weekday or weekend day. Make a panel plot containing a time series plot (i.e. type = “l”) of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all weekday days or weekend days (y-axis). See the README file in the GitHub repository to see an example of what this plot should look like using simulated data.

# Make a histogram of the total number of steps taken each day and report mean and median and compare that to original values without na removed.  
  
StepsPer5min <- aggregate(steps ~ date, activityData\_Fill, sum)  
hist(StepsPer5min$steps,  
 main = "Imputed Number of Steps Per Day",  
 breaks = seq(from=0, to=25000, by=1000),  
 col="black",   
 xlab="Total number of steps",   
 ylim=c(0, 30),  
)



MeanStepsFilled <- mean(StepsPer5min$steps, na.rm = TRUE)  
MedianStepsFilled <- median(StepsPer5min$steps, na.rm = TRUE)  
diffMean = MeanStepsFilled - Steps\_mean  
diffMedian = MedianStepsFilled - Steps\_median  
diffTotal = sum(StepsPer5min$steps) - sum(StepsPerDay$steps)  
# There is a difference of 0 in the mean steps of the two dataset. There is a difference of -1.076381110^{4} in the median steps of the two dataset. There is a difference of 8.612950910^{4} in the total steps of the two dataset.  
  
# Are there differences in activity patterns between weekdays and weekends?  
# Create a new factor variable in the dataset with two levels - "weekend" and "weekday"  
DayType <- function(date) {  
 day <- weekdays(date)  
 if (day %in% c('Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday'))  
 return ("weekeday")  
 else if (day %in% c('Saturday', 'Sunday'))  
 return ("weekend")  
 else  
 stop ("Invalid Date Format.")  
}  
activityData$date <- as.Date(activityData$date)  
activityData$day <- sapply(activityData$date, FUN = DayType)  
  
# Make a panel plot containnig a time-series plot of the 5-minute interval  
# and the average number of steps taken across all weekdays or weekends  
meanStepsByDay <- aggregate(steps ~ interval + day, activityData, mean)  
ggplot(data = meanStepsByDay, aes(x = interval, y = steps)) +   
 geom\_line(color = "royalblue") +  
 facet\_grid(day ~ .) +  
 ggtitle("Average Daily Activity Pattern") +  
 xlab("5-minute Interval") +  
 ylab("Average Number of Steps") +  
 theme(plot.title = element\_text(hjust = 0.5))



Answers: New mean is the same as old with 10766.19, but the median changed by 1.19 to now be the same value as the mean. It seems as though the weekday to weekend is different particularly with earlier movement during weekdays and greater peaks in early intervals, but the activity seems to be steady throughout the day on weekends with potentially more area under the curve.