Activity monitoring

Jeff Kian

Thursday, May 14, 2015

### Load and Process the data

##### Load the data

Steps taken:  
1. Set the working directory  
2. Initialise the relevant libraries e.g (dplyr)  
3. Read the data set into the actvity data

setwd("C:/Users/jeffk\_000/Dropbox/Work Files/BI/Information Management/07. Data Scientist Toolbox/05. Reproducible Research/Assignment 1/repdata-data-activity")  
library(dplyr)  
library(sqldf)

## Warning: package 'RSQLite' was built under R version 3.1.3

library(data.table)  
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.1.3

ActData<-read.csv("activity.csv")

##### Process the data

Approach

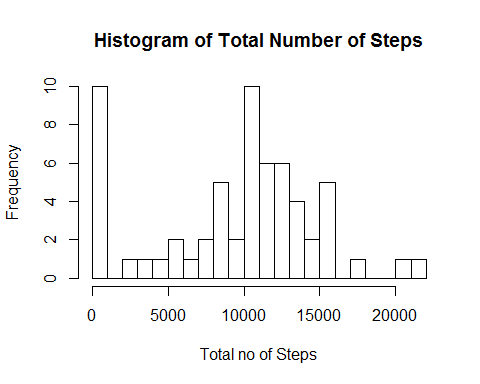
1. For each day calculate the total number of steps for that day
2. Use the group function from dplyr to group the data and assign to Aggregate Activity data table (AggActData)
3. Use the summarise function to summarise steps by date and don't consider NA records
4. Add a weekday indicator onto this summary table

AggActData<-ActData %>%   
group\_by(date) %>%   
summarise(TotalSteps=sum(steps,na.rm=TRUE))  
AggActData$day <- weekdays(as.Date(AggActData$date))

#### Question 1: What is mean total number of steps taken per day?

1. Calculate the total nuber of steps per day
2. Create a Histogram of total number of steps

hist(AggActData$TotalSteps,  
 breaks=20,  
 xlab="Total no of Steps",  
 main="Histogram of Total Number of Steps")



The median of the Total Steps for all days is

median(AggActData$TotalSteps)

## [1] 10395

The mean of the total steps for all days is

mean(AggActData$TotalSteps)

## [1] 9354.23

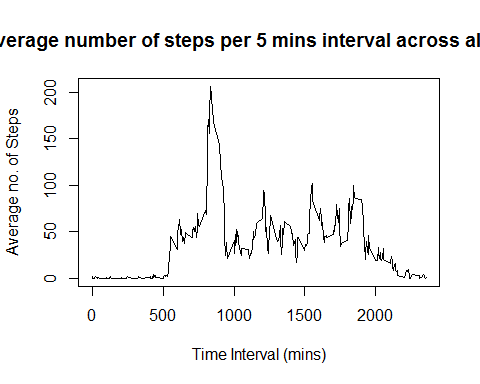
#### Question 2: What is the average daily activity pattern?

Q 2.1 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)

Approach

* For each interval calculate the total number of steps for that interval
* Use the group function from dplyr to group the data and assign to Aggregate Activity data table (AggActData2)
* Use the summarise function to summarise steps by interval and don't consider NA records
* Plot a line graph of interval against Average Number of Steps

AggActData2<-ActData %>%   
group\_by(interval) %>%   
summarise(AveSteps=mean(steps,na.rm=TRUE))  
  
plot(AggActData2$interval, AggActData2$AveSteps,type="l", xlab="Time Interval (mins)",ylab="Average no. of Steps",main="Average number of steps per 5 mins interval across all days")



Q 2.2 Show which 5 mins interval i.e. which time of day has the most steps

Approach

* Identify which time interval has the highest number of steps.
* Find the index of the row number where the number of steps is the maximum number of steps
* Use that row number to identify the interval

AggActData2[which(AggActData2$AveSteps==max(AggActData2$AveSteps)),1]

## Source: local data frame [1 x 1]  
##   
## interval  
## 1 835

#### Question 3: Inputting Missing Values

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

Approach

* Firstly identify which records is.na return true the code will generate a vector with logical values indicating if the rows have an NA value
* Use the which function to identify the row numbers
* Use the length function to count the number of entries in the NA vector, this will indicate the number of records with NA values

The total number of observations that have NA values is:

length(which(is.na(ActData$steps)=="TRUE"))

## [1] 2304

3.2 Create a new dataset that has the missing values filled in

Approach  
- Using the sqldf library join the Activity data (Steps by interval by day ActData)to the previously calculated Average Steps per interval (AggActData2 that excluded NA)  
- Using the mutate function from dplyr create a new column AveStepsFixed  
- Embed conditional statement that evaluates whether the steps in the original activity data for that interval were empty  
- If they were NA replace with the calculated Average steps for that interval Else use the existing steps for that interval

ActData2<-sqldf('select ActData.steps,ActData.date,ActData.interval,AggActData2.AveSteps from ActData join AggActData2 using (interval) ')

## Loading required package: tcltk

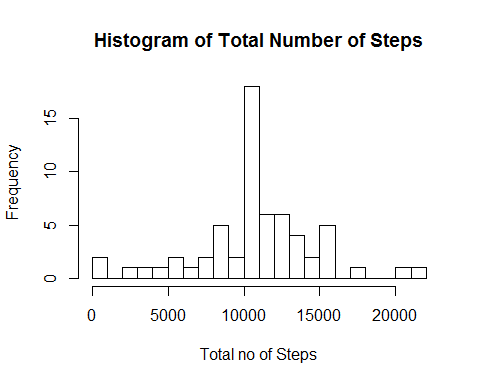
ActDataFixed<-mutate(ActData2,AveStepsFixed=ifelse(is.na(ActData2$steps)==TRUE,AveSteps,steps))  
head(ActDataFixed)

## steps date interval AveSteps AveStepsFixed  
## 1 NA 2012-10-01 0 1.7169811 1.7169811  
## 2 NA 2012-10-01 5 0.3396226 0.3396226  
## 3 NA 2012-10-01 10 0.1320755 0.1320755  
## 4 NA 2012-10-01 15 0.1509434 0.1509434  
## 5 NA 2012-10-01 20 0.0754717 0.0754717  
## 6 NA 2012-10-01 25 2.0943396 2.0943396

3.3 Make a histogram of the total number of steps taken each day

AggActDataFixed<-ActDataFixed %>%   
group\_by(date) %>%   
summarise(TotalSteps=sum(AveStepsFixed,na.rm=TRUE))  
AggActDataFixed$day <- weekdays(as.Date(AggActDataFixed$date))

hist(AggActDataFixed$TotalSteps,  
 breaks=20,  
 xlab="Total no of Steps",  
 main="Histogram of Total Number of Steps")



3.4 Calculate and report the mean and median total number of steps taken per day based on the fixed data

The median of the Total Steps for all days is

median(AggActDataFixed$TotalSteps)

## [1] 10766.19

The mean of the total steps for all days is

mean(AggActDataFixed$TotalSteps)

## [1] 10766.19

The values for the mean and median are different from the original data set. Although calculations on the original data set excluded NA records in calculation of daily summaries. Making the assumption using average values for intervals missing values has increased the mean an median. But not siginificantly

#### Q4. Are there differences in activity patterns between weekdays and weekends?

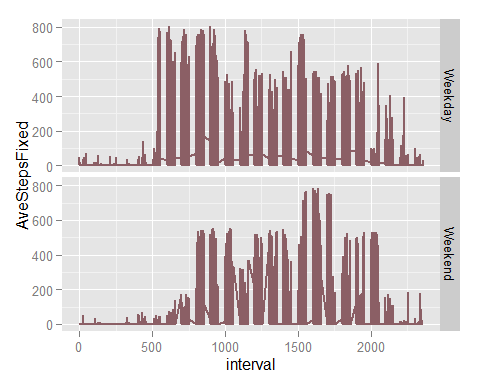
Approach

* Add an indicator to the data set to see if Weekday of weekend
* Plot the average steps per interval on a faceted line plot using ggplot2 and obsere any noticeable patterns between weekend and weekday activity

ActDataFixed$dayofweek <- wday(as.Date(ActDataFixed$date))  
ActDataFixed<-mutate(ActDataFixed,WeekEndInd=ifelse(ActDataFixed$dayofweek ==1 | ActDataFixed$dayofweek ==7,"Weekend","Weekday"))  
head(ActDataFixed,15)

## steps date interval AveSteps AveStepsFixed dayofweek WeekEndInd  
## 1 NA 2012-10-01 0 1.7169811 1.7169811 2 Weekday  
## 2 NA 2012-10-01 5 0.3396226 0.3396226 2 Weekday  
## 3 NA 2012-10-01 10 0.1320755 0.1320755 2 Weekday  
## 4 NA 2012-10-01 15 0.1509434 0.1509434 2 Weekday  
## 5 NA 2012-10-01 20 0.0754717 0.0754717 2 Weekday  
## 6 NA 2012-10-01 25 2.0943396 2.0943396 2 Weekday  
## 7 NA 2012-10-01 30 0.5283019 0.5283019 2 Weekday  
## 8 NA 2012-10-01 35 0.8679245 0.8679245 2 Weekday  
## 9 NA 2012-10-01 40 0.0000000 0.0000000 2 Weekday  
## 10 NA 2012-10-01 45 1.4716981 1.4716981 2 Weekday  
## 11 NA 2012-10-01 50 0.3018868 0.3018868 2 Weekday  
## 12 NA 2012-10-01 55 0.1320755 0.1320755 2 Weekday  
## 13 NA 2012-10-01 100 0.3207547 0.3207547 2 Weekday  
## 14 NA 2012-10-01 105 0.6792453 0.6792453 2 Weekday  
## 15 NA 2012-10-01 110 0.1509434 0.1509434 2 Weekday

ggplot(ActDataFixed, aes(interval,AveStepsFixed))+geom\_line(color="lightpink4", lwd=1)+facet\_grid(WeekEndInd ~ .)



### Observations

Weekdays are charaterised by more activity (steps) during the early portions of the day. This may possibly be attributed to people walking to get to work or to their means of transport to get to work. Weekday activity remains fairly constant during the middle and latter portions with slight spike potentially during lunch periods and tapering off toward the evening. Interesting is that there is not the same actvity as early in the monring when people go to work in the evening when you would expect them to walk home.

Weekends are characterised by minimal actvivity early in the day and morning as it assumed people sleep in on weekends. There is a spike of activity in the late afternoon. With both weekend and weekday actvity minimal during rest , sleep periods.