Instructions

Introduction

It is now possible to collect a large amount of data about personal movement using activity monitoring devices such as a Fitbit, Nike Fuelband, or Jawbone Up. These type of devices are part of the “quantified self” movement – a group of enthusiasts who take measurements about themselves regularly to improve their health, to find patterns in their behavior, or because they are tech geeks. But these data remain under-utilized both because the raw data are hard to obtain and there is a lack of statistical methods and software for processing and interpreting the data.

This assignment makes use of data from a personal activity monitoring device. This device collects data at 5 minute intervals through out the day. The data consists of two months of data from an anonymous individual collected during the months of October and November, 2012 and include the number of steps taken in 5 minute intervals each day.

There are 9 questions to answer. 8 Question will be answered question by question. The 9-th question is answered by this document itself.

Question 1

Code for reading in the dataset and/or processing the data

**if**(!file.exists('activity.csv')){

unzip('activity.zip')

}

DF <- read.csv('activity.csv')

str(DF)

## 'data.frame': 17568 obs. of 3 variables:

## $ steps : int NA NA NA NA NA NA NA NA NA NA ...

## $ date : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 1 1 1 1 1 1 1 1 1 1 ...

## $ interval: int 0 5 10 15 20 25 30 35 40 45 ...

Question 2

Histogram of the total number of steps taken each day

*# Create the sums of steps per date*

DFsteps <- tapply(DF$steps, DF$date, FUN=sum, na.rm=TRUE)

*# Perform histogram of steps per day*

**library**(ggplot2)

qplot(DFsteps, binwidth=1000, xlab="total number of steps taken each day")

Question 3

Mean and median number of steps taken each day

*# Create mean and median of steps per day*

stepsMean <- mean(DFsteps, na.rm=TRUE)

stepsMedian <- median(DFsteps, na.rm=TRUE)

*# Output mean and median*

stepsMean

## [1] 9354.23

stepsMedian

## [1] 10395

Question 4

Time series plot of the average number of steps taken

**library**(ggplot2)

*# Create the means by intervals*

averages <- aggregate(x=list(steps=DF$steps), by=list(interval=DF$interval),FUN=mean, na.rm=TRUE)

ggplot(data=averages, aes(x=interval, y=steps)) +

geom\_line() +

ggtitle("Time Series: average number of steps") +

xlab("5-minute interval") +

ylab("average number of steps taken")

Question 5

The 5-minute interval that, on average, contains the maximum number of steps

averages[which.max(averages$steps),]

## interval steps

## 104 835 206.1698

Question 6

Code to describe and show a strategy for imputing missing data

Idea: Replace the NA by the mean of the corresponding interval.

*# copy of data frame*

DF2 <- DF

*# add column for copleating index*

DF2$CI <- "original"

*# number of rows to check*

l <- nrow(DF2)

*# numbers of NAs*

length(which(is.na(DF2$steps)))

## [1] 2304

*# replace NAs by corresponing mean of the same interval --> complete data frame DF2*

**for** (i **in** 1:l) {

**if** (is.na(DF2[i,1])) {

DF2[i,1] <- averages[averages$interval == DF2[i,3],2]

DF2[i,4] <- "completed"

}

}

*# numbers of NAs / completed (control)*

length(which(is.na(DF2$steps)))

## [1] 0

length(which(DF2$CI=="completed"))

## [1] 2304

*# Recreate the sums of steps per date*

DFsteps2 <- tapply(DF2$steps, DF2$date, FUN=sum, na.rm=TRUE )

*# Recreate the mean and median of steps per date*

stepsMean2 <- mean(DFsteps2)

stepsMedian2 <- median(DFsteps2)

c(stepsMean2, stepsMean)

## [1] 10766.19 9354.23

c(stepsMedian2, stepsMedian)

## [1] 10766.19 10395.00

We see, that the completation of the data frame did strongly change mean and median of the steps per date. What did also is the distribution of the sum of steps per date, as we will see in the next section:

Question 7

Histogram of the total number of steps taken each day after missing values are imputed

*# Preparation environment*

**library**(ggplot2)

**library**(gridExtra)

**require**(gridExtra)

*# Perform histogram of steps per day*

plot1 <- qplot(DFsteps,

binwidth=1000,

ylim=c(0,15),

main="original",

xlab="total number of steps taken each day")

plot2 <- qplot(DFsteps2,

binwidth=1000,

ylim=c(0,15),

main="completed",

xlab="total number of steps taken each day")

*# Plotting 2 plot in grid*

grid.arrange(plot1, plot2, ncol=2)

Question 8

Panel plot comparing the average number of steps taken per 5-minute interval across weekdays and weekends

**library**(ggplot2)

**library**(gridExtra)

*# Formatting and expanding DF2 by $WD (Weekday in German) an $WDG (WeekDayGroup)*

DF2[,2] <- as.Date(DF2[,2])

DF2$WD <- weekdays(DF2[,2])

DF2$WDG <- "week" *# default = "week"*

*# Filling in the WeekDayGroup in German*

**for** (i **in** 1:l) {

**if** (DF2[i,5] == "Samstag" | DF2[i,5] == "Sonntag") {

DF2[i,6] <- "weekend"

}

}

DF2[,6] <- as.factor(DF2[,6])

DF2w <-subset(DF2,DF2[,6]=="week")

DF2we <-subset(DF2,DF2[,6]=="weekend")

*# Recreate the means by intervals*

averagesW <- aggregate(steps ~ interval, DF2w, FUN=mean)

averagesWe <- aggregate(steps ~ interval, DF2we, FUN=mean)

*# prepare the plots*

plot1 <- ggplot(data=averagesW, aes(x=interval, y=steps)) +

geom\_line() +

ylim(0, 250) +

ggtitle("Weekdays") +

xlab("5-minute interval") +

ylab("average number of steps taken")

plot2 <- ggplot(data=averagesWe, aes(x=interval, y=steps)) +

geom\_line() +

ylim(0, 250) +

ggtitle("Weekend Days") +

xlab("5-minute interval") +

ylab("average number of steps taken")

*# use the library "gridExtra"*

**require**(gridExtra)

*# plot*

grid.arrange(plot1, plot2, ncol=2)

Question 9

All of the R code needed to reproduce the results (numbers, plots, etc.) in the report

The underlying R Markdown document contains all of the R code needed to reproduce the report.