RepData_PeerAssessment1

Mostafa Alaa

10 Jan 2016

Contents

`{r, echo=FALSE,results='hide'} locale_original <- Sys.getlocale( category = "LC_TIME" ) Sys.setlocale("LC_TIME", "English")` Load Needed Libraries

```
library(sqldf)
library(lattice)
library(ggplot2)
```

Loading and preprocessing the data

```
unzip("activity.zip")
Activity_Data <- read.csv("activity.csv", colClasses = c("numeric", "Date", "factor"))
Activity_Data$Month = as.numeric(format(Activity_Data$date, "%m"))
```

What is mean total number of steps taken per day?

1. Make a histogram of the total number of steps taken each day

```
activity_Data_NaExcluded <- na.exclude(Activity_Data)

rownames(activity_Data_NaExcluded) <- 1:nrow(activity_Data_NaExcluded)

g <- ggplot(activity_Data_NaExcluded, aes(date, steps))

g +  geom_bar(stat = "identity", colour = "violetred4", fill = "violetred4", width = 0.6) + facet_grid(
ggtitle("Histogram of Total Number of Steps Taken Each Day")
```

2. Calculate and report the mean and median total number of steps taken per day

- Mean total number of steps taken per day: “‘{r}

totalSteps <- aggregate(activity_Data_NaExcluded$steps, list(Date = activity_{D}ata_{N}aExcluded$date), FUN = “sum”)

mean(totalSteps$x)

“‘ - Median total number of steps taken per day:

```
median(totalSteps$x)
```

What is the average daily activity pattern?

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)

```
average_Steps <- aggregate(activity_Data_NaExcluded$steps, list(interval = as.numeric(as.character(acti

ggplot(average_Steps, aes(interval, x)) + geom_line(color = "wheat3", size = 0.7) + labs(title = "Time
```

2. Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?

```
names(average_Steps)[2] <- "avgSteps"
average_Steps[average_Steps$avgSteps == max(average_Steps$avgSteps), ]
```

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.

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

```
sum(is.na(Activity_Data))
```

2. 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.

My strategy is Replacing all NA steps values with the average value in the same inteval

3. Create a new dataset that is equal to the original dataset but with the missing data filled in.

```
Activity_Data_Replaced_NA <- sqldf(c("update Activity_Data set steps = (select average_Steps.avgSteps f
head(Activity_Data_Replaced_NA)

sum(is.na(Activity_Data_Replaced_NA))
```

4. 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.

```
g <- ggplot(Activity_Data_Replaced_NA, aes(date, steps))

g +  geom_bar(stat = "identity", colour = "violetred4", fill = "violetred4", width = 0.7) + facet_grid(
```

5. 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?

- Mean total number of steps taken per day: “‘{r} average_Steps_without_NA <- sqldf(“select date, sum(steps) steps from Activity_Data_Replaced_NA group by date order by date”)

mean.default(average_Steps_without_NA$steps)

“‘

- Median of total number of steps taken per day: “‘{r}

median(average_Steps_without_NA$steps)

`There is a different from the first input Data in the median`{r} mean(average_Steps_without_NA$steps) - mean(totalSteps$x)

median(average_Steps_without_NA$steps) - median(totalSteps$x)

“‘

Are there differences in activity patterns between weekdays and weekends?

1. 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.

```
Activity_Data_Replaced_NA$"dayweek" <- weekdays(Activity_Data_Replaced_NA$date)

Activity_Data_Replaced_NA <- sqldf(c("update Activity_Data_Replaced_NA set dayweek = 'weekend' where day

Activity_Data_Replaced_NA <- sqldf(c("update Activity_Data_Replaced_NA set dayweek = 'weekday' where 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).

```
Grouped_Activity_weekday <- aggregate(data = Activity_Data_Replaced_NA,Activity_Data_Replaced_NA$steps,
                      FUN = "mean")

xyplot(data = Grouped_Activity_weekday,x ~ interval | dayweek,
       layout = c(1, 2), type = "l",
       xlab = "Interval", ylab = "Number of steps")
```