Reproductible Research : Projet Assignment Week1

Elise VI NHU BA

24 janvier 2016

Loading and preprocessing the data

Show any code that is needed to

1. Load the data (i.e. read.csv())

```
data<-read.csv("./Data/activity.csv")
```

2. Process/transform the data (if necessary) into a format suitable for your analysis

```
dt<-read.table(file = "./Data/activity.csv", header = TRUE, sep = ",")
```

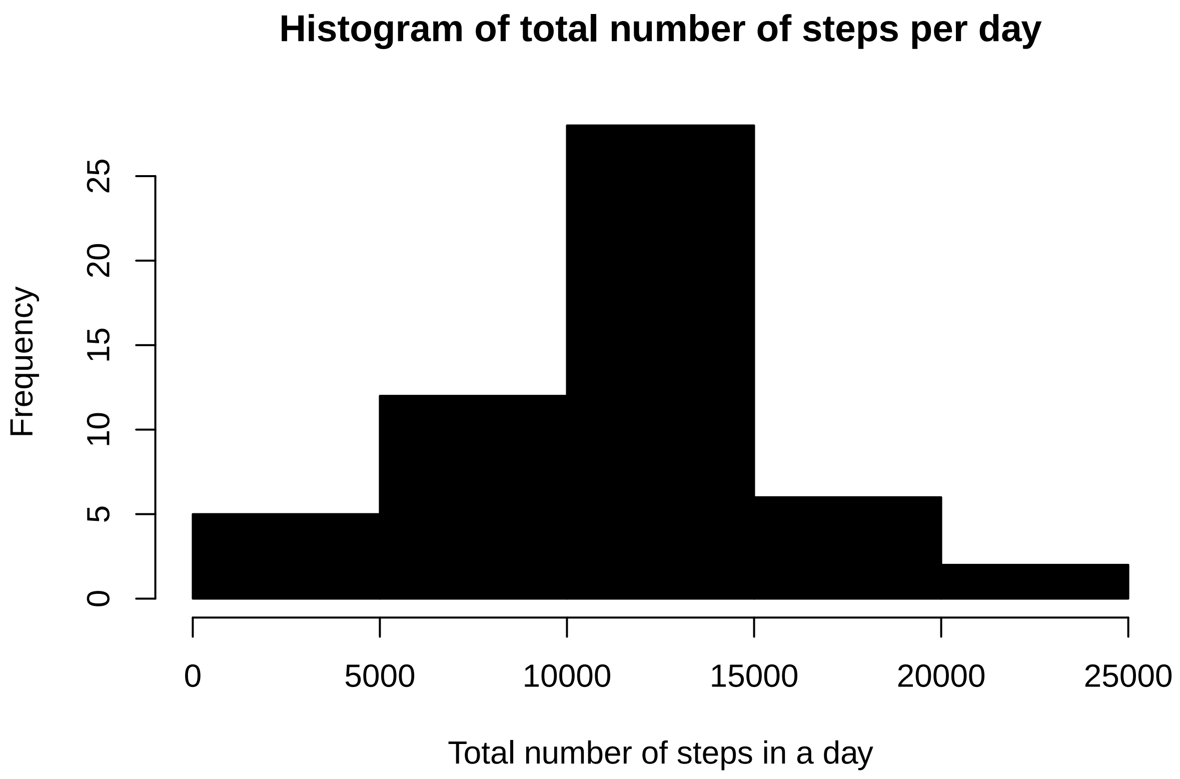
What is mean total number of steps taken per day?

For this part of the assignment, you can ignore the missing values in the dataset.

```
dt_wo_NA <- na.omit(dt)
```

1. 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

```
StepPerDay<- aggregate(steps ~ date, dt_wo_NA, sum)
hist(StepPerDay$steps, col=1, main="Histogram of total number of steps per day",
     xlab="Total number of steps in a day")
```



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

```
mean(StepPerDay$steps)
```

```
## [1] 10766.19
```

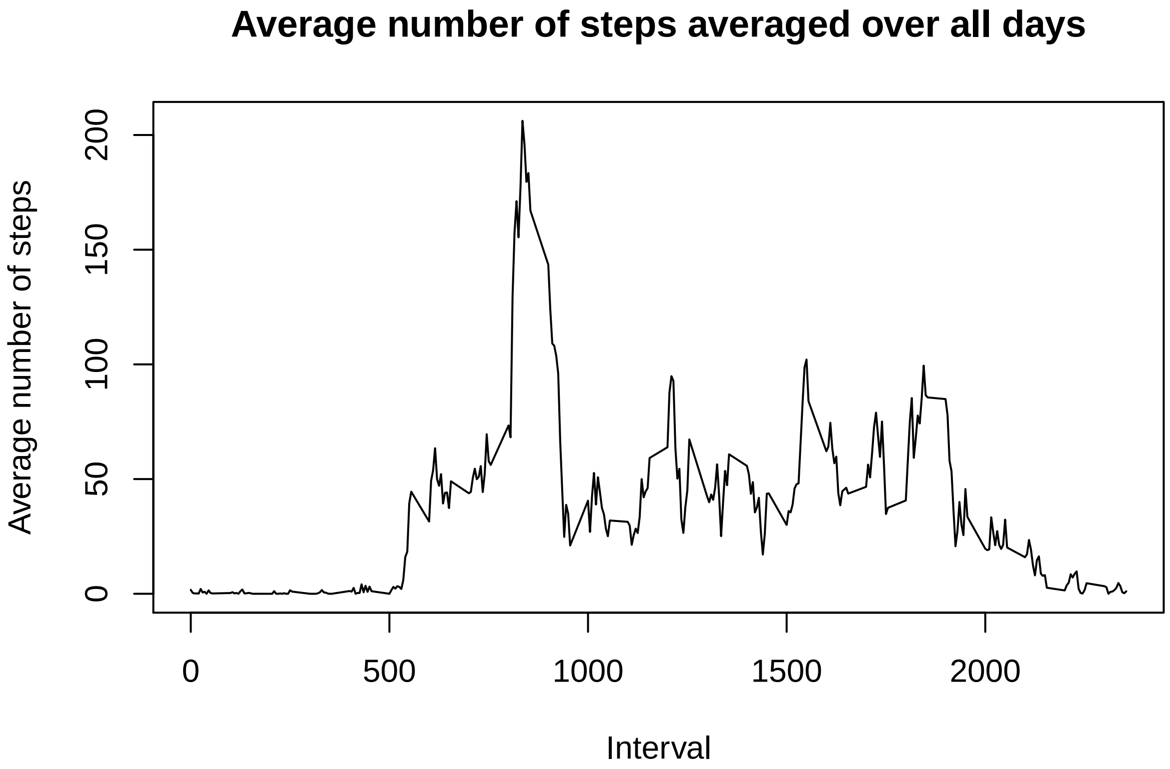
```
median(StepPerDay$steps)
```

```
## [1] 10765
```

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)

```
StepMeanPerInterval <- aggregate(steps ~ interval, dt_wo_NA, mean)
plot(StepMeanPerInterval$interval, StepMeanPerInterval$steps, type='l', col=1,
     main="Average number of steps averaged over all days", xlab="Interval",
     ylab="Average number of steps")
```



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

```
StepMeanPerInterval$interval[which.max(StepMeanPerInterval$steps)]
```

```
## [1] 835
```

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)

```
  dt_NA <- dt[is.na(dt)]
  length(dt_NA)
```

```
## [1] 2304
```

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.

I decide to use the mean for that 5-minute interval because for each there is at less one value ok.

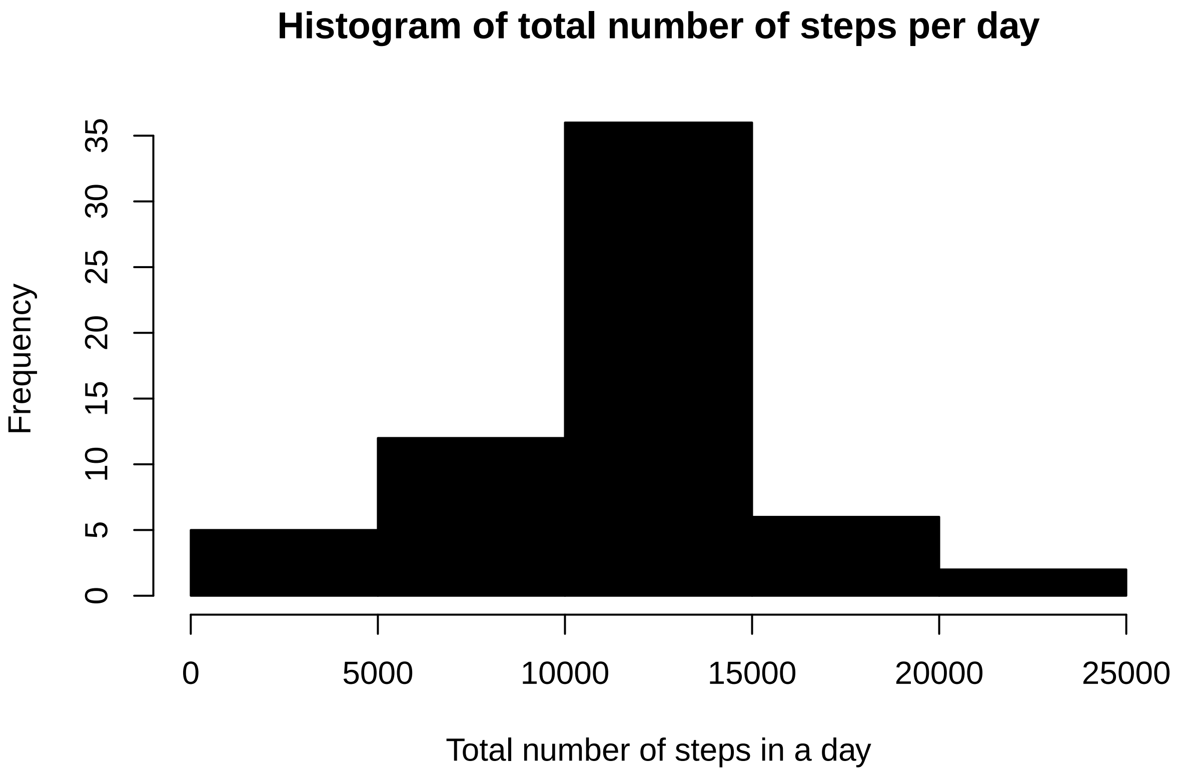
3. Create a new dataset that is equal to the original dataset but with the missing data filled in.

```
dt_Modified_NA<-dt;
MeanStepPerInterval<-aggregate(steps ~ interval, dt, mean) ;
for (i in 1:nrow(dt_Modified_NA)){
    if (is.na(dt_Modified_NA$steps[i])){
      inter <- dt_Modified_NA$interval[i];
      row_id <- which(MeanStepPerInterval$interval == inter);
      if(length(row_id)>0){
        steps_val <- MeanStepPerInterval[row_id,2];
        dt_Modified_NA$steps[i] <- steps_val;
      }else{dt_Modified_NA$steps[i] <- 0}

    }
  }
```

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.

```
StepPerDay_ModifiedNA<- aggregate(steps ~ date, dt_Modified_NA, sum)
hist(StepPerDay_ModifiedNA$steps, col=1, main="Histogram of total number of steps per day",
     xlab="Total number of steps in a day")
```



```
mean(StepPerDay_ModifiedNA$steps)
```

```
## [1] 10766.19
```

```
median(StepPerDay_ModifiedNA$steps)
```

```
## [1] 10766.19
```

Do these values differ from the estimates from the first part of the assignment? The mean doesn't differ and the median is equal to the mean.

What is the impact of imputing missing data on the estimates of the total daily number of steps? It errase the "numerical" effect and it is more realist than NA estimation.

Are there differences in activity patterns between weekdays and weekends?

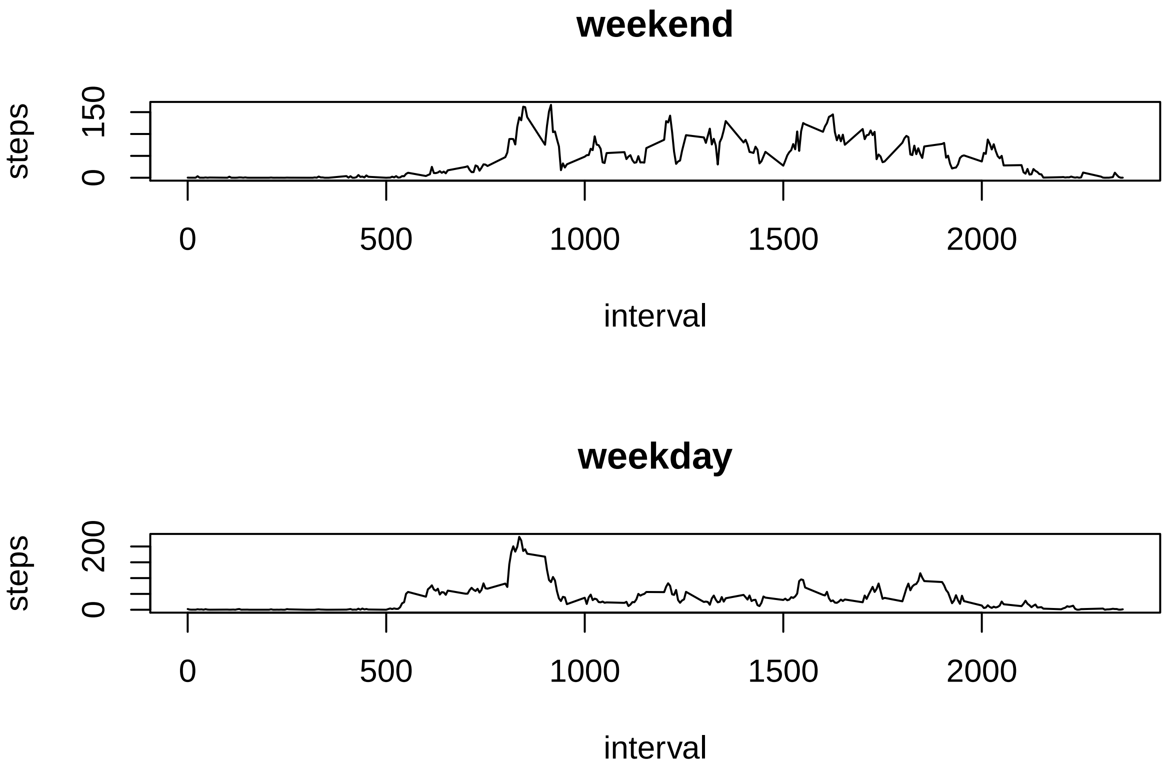
For this part the weekdays() function may be of some help here. Use the dataset with the filled-in missing values for this part.

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.

```
library(knitr)
## ---------------------------------------------------------------------------
dt_WDayEnd<-dt_Modified_NA
# convert date to string
dt_WDayEnd$date <- as.Date(dt_WDayEnd$date, "%Y-%m-%d")
# add a new column indicating day of the week
dt_WDayEnd$day <- weekdays(dt_WDayEnd$date)
# add a new column for "weekday" or "weekend", initialised with "weekday"
dt_WDayEnd$day_type <- c("weekday")
# If day is "samedi" or "dimanche, make day_type as weekend
for (i in 1:nrow(dt_WDayEnd)){
  if (dt_WDayEnd$day[i] == "dimanche" || dt_WDayEnd$day[i] == "samedi"){
    dt_WDayEnd$day_type[i] <- "weekend"
  }
}
# convert day_time to factor
dt_WDayEnd$day_type <- as.factor(dt_WDayEnd$day_type)
StepMeanPerInterval_WDayEnd <- aggregate(steps ~ interval+day_type, dt_WDayEnd, mean)
```

2. 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.

```
par( mfrow = c( 2, 1 ))
for (type in c("weekend", "weekday")) {
    steps.type <- aggregate(steps ~ interval, data = StepMeanPerInterval_WDayEnd , subset = StepMeanPerI
    plot(steps.type, type = "l", main = type)
}
```



weekend

weekday