

RR_Assessment1

Loading and preprocessing the data

Loading the data

```
DS<-read.csv("activity.csv")
head(DS)
```

```
##   steps      date interval
## 1    NA 2012-10-01         0
## 2    NA 2012-10-01         5
## 3    NA 2012-10-01        10
## 4    NA 2012-10-01        15
## 5    NA 2012-10-01        20
## 6    NA 2012-10-01        25
```

Preprocessing the data

New variables:

- steps_repared - based on assignment requirements, new data will be put in the place of NAs
- day - weekday or weekend (based on date)
- ind - indicator (1 - data available, 0 - data - not available)

```
DS[,4]=DS[,1]

for (i in 1:length(DS[,1]))
{
  day=weekdays(as.Date(DS[i,2]))
  if(day=="subota" || day=="nedjelja")
    DS[i,5]="weekend"
  else DS[i,5]="weekday"
}

for (i in 1:length(DS[,1])){if (is.na(DS[i,1])) {DS[i,6]=0} else {DS[i,6]=1}}

colnames(DS)=c("steps","date","interval","steps_repared", "day","ind")
```

Grouping data - new datasets:

- SBD1 - steps_by_date_1 - grouping by date (only available data (ind=1))
- SBI1 - steps_by_interval_1 - grouping by interval (only available data (ind=1))

```
SBD1<-aggregate(subset(DS,ind==1)$steps,list(subset(DS,ind==1)$date),sum)
colnames(SBD1)<-c("date","steps")
SBI1<-aggregate(subset(DS,ind==1)$steps,list(subset(DS,ind==1)$interval),mean)
colnames(SBI1)<-c("interval","steps")
```

Analysis

A1. Total number of steps taken per day

A1.1. Total number of steps taken per day

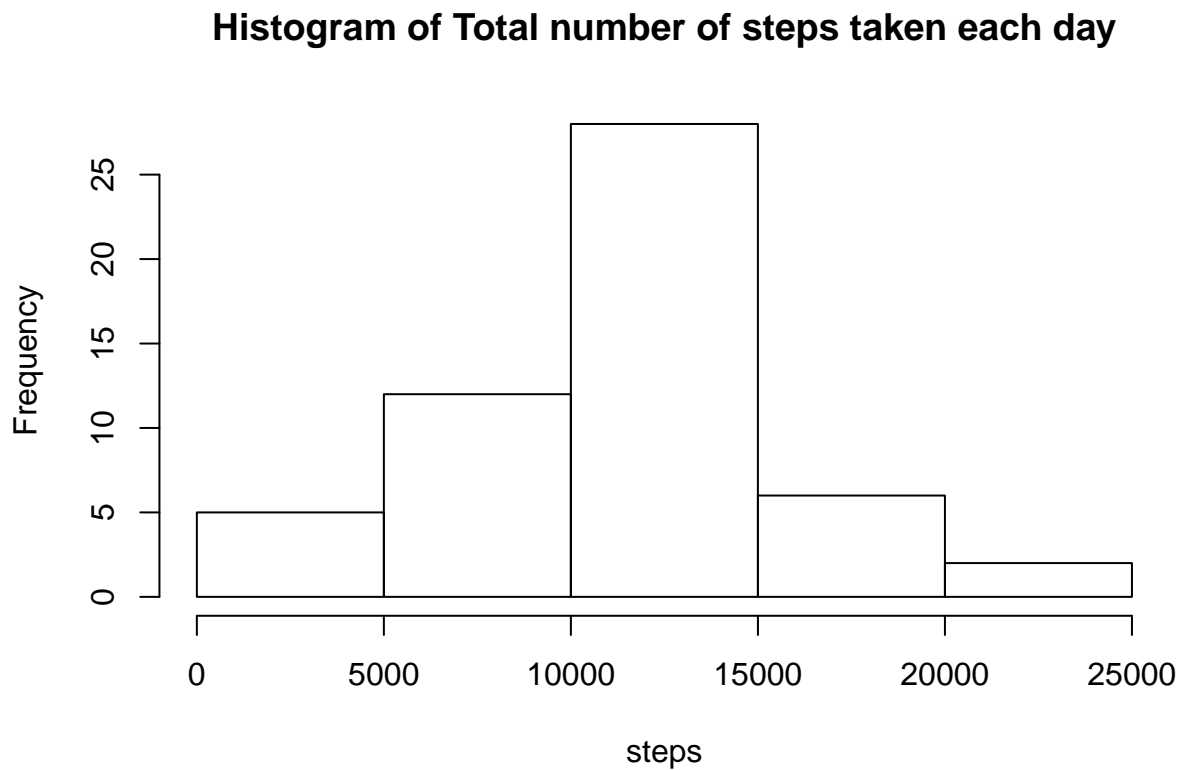
Calculated in preprocessing data

```
head(SBD1)
```

```
##      date steps
## 1 2012-10-02   126
## 2 2012-10-03 11352
## 3 2012-10-04 12116
## 4 2012-10-05 13294
## 5 2012-10-06 15420
## 6 2012-10-07 11015
```

A1.2. Histogram Histogram of the total number of steps taken each day

```
hist(SBD1$steps,main="Histogram of Total number of steps taken each day",xlab="steps")
```



```
mean(SBD1$steps)
```

A1.3.1. Mean of the total number of steps taken per day

```
## [1] 10766
```

```
median(SBD1$steps)
```

A1.3.2. Median of the total number of steps taken per day

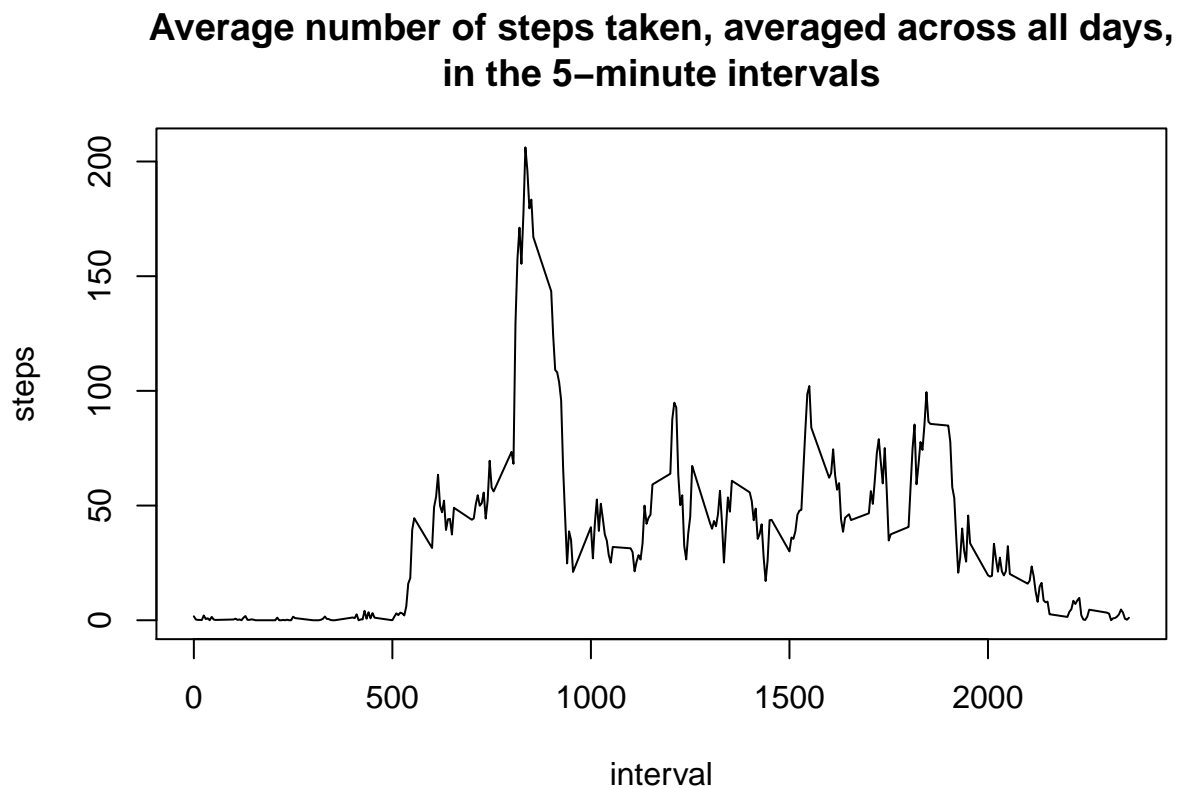
```
## [1] 10765
```

A2. Average daily activity pattern

A2.1. Time series plot

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)

```
plot(SBI1,type="l",main="Average number of steps taken, averaged across all days,\n in the 5-minute intervals")
```



A2.2. The 5-minute interval, on average across all the days, that contains the maximum number of steps.

```
SBI1[SBI1[,2]==max(SBI1[,2]),]
```

```
##      interval steps  
## 104         835 206.2
```

A3. Imputing missing values

A3.1. Total number of missing values in the dataset

```
sum(is.na(DS$steps))
```

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

A3.2.Strategy for filling in all of the missing values in the dataset: the mean for that 5-minute interval.

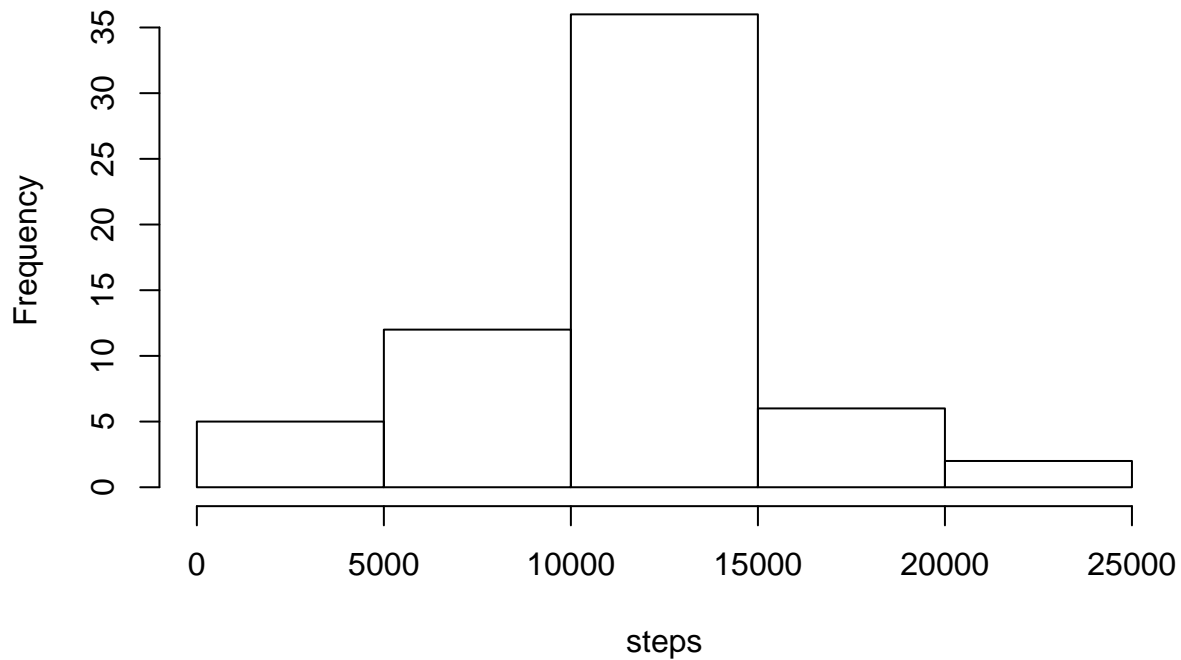
A3.3. New dataset-variable (steps_repared) that is equal to the original dataset but with the missing data filled in.

```
for (i in c(which(DS$ind==0)) )  
  DS[i,]$steps_repared=SBI1[which(SBI1[,1]==DS[i,]$interval),2]
```

A3.4. Histogram of the total number of steps taken each day

```
hist(aggregate(DS$steps_repared, list(date=DS$date), sum)[,2],  
     main="Histogram of the total number of steps taken each day (repared)",xlab="steps")
```

Histogram of the total number of steps taken each day (repared)



Mean and median total number of steps taken per day.

```
mean(aggregate(DS$steps_repared, list(date=DS$date), sum)[,2])
```

```
## [1] 10766
```

```
mean(SBD1[,2])
```

```
## [1] 10766
```

The values for the mean do not differ from the estimates from the first part of the assignment.

```
median(aggregate(DS$steps_repared, list(date=DS$date), sum)[,2])
```

```
## [1] 10766
```

```
median(SBD1[,2])
```

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

The values for the median differ from the estimates from the first part of the assignment.

The impact of imputing missing data on the estimates of the total daily number of steps is larger median.

A4. The differences in activity patterns between weekdays and weekends.

A4.1 New factor variable in the dataset with two levels – “weekday” and “weekend” indicating whether a given date is a weekday or weekend day.

```
DS$day<-as.factor(DS$day)
```

A4.2. 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).

```
set1<-subset(aggregate(steps_repaired~interval+day, data=DS, mean),day=="weekday")
set2<-subset(aggregate(steps_repaired~interval+day, data=DS, mean),day=="weekend")

par(mfrow=c(2, 1))
plot(set1[,1],set1[,3],type="l",xlab="interval",ylab="steps (repaired)",main="Weekdays",bg="gray",ylim=c(0,250))
plot(set2[,1],set2[,3],type="l",xlab="interval",ylab="steps (repaired)",main="Weekend",bg="gray",ylim=c(0,250))
```

