Project

KK Setshedi

01/10/2020

Loading library and reading data

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------------------

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.1
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ------------------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

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

Reprocessing of data

Change date format into date

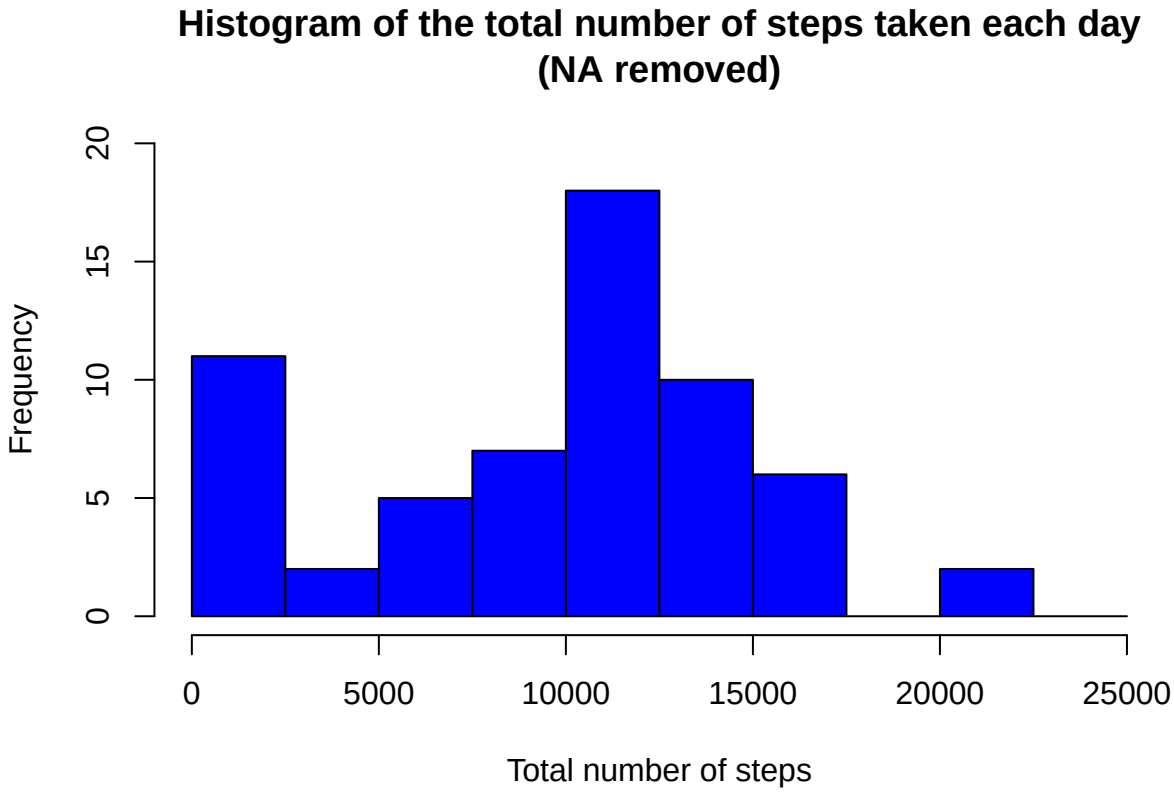
```
data$date <-  as.Date(data$date)
```

Histogram without NA values

Histogram of total number of steps taken on each day is shown below:

```
sum_steps<-aggregate(data$steps,by=list(data$date),FUN=sum,na.rm=TRUE)

hist(sum_steps$x,
      breaks=seq(from=0, to=25000, by=2500),
      col="Blue",
      xlab="Total number of steps",
      ylim=c(0, 20),
      main="Histogram of the total number of steps taken each day\n(NA removed)")
```



Mean and Median of Steps

```
mean_steps <- mean(sum_steps$x)
median_steps <-  median(sum_steps$x)
```

The mean steps are 9354.2295082 and the median steps are 10395

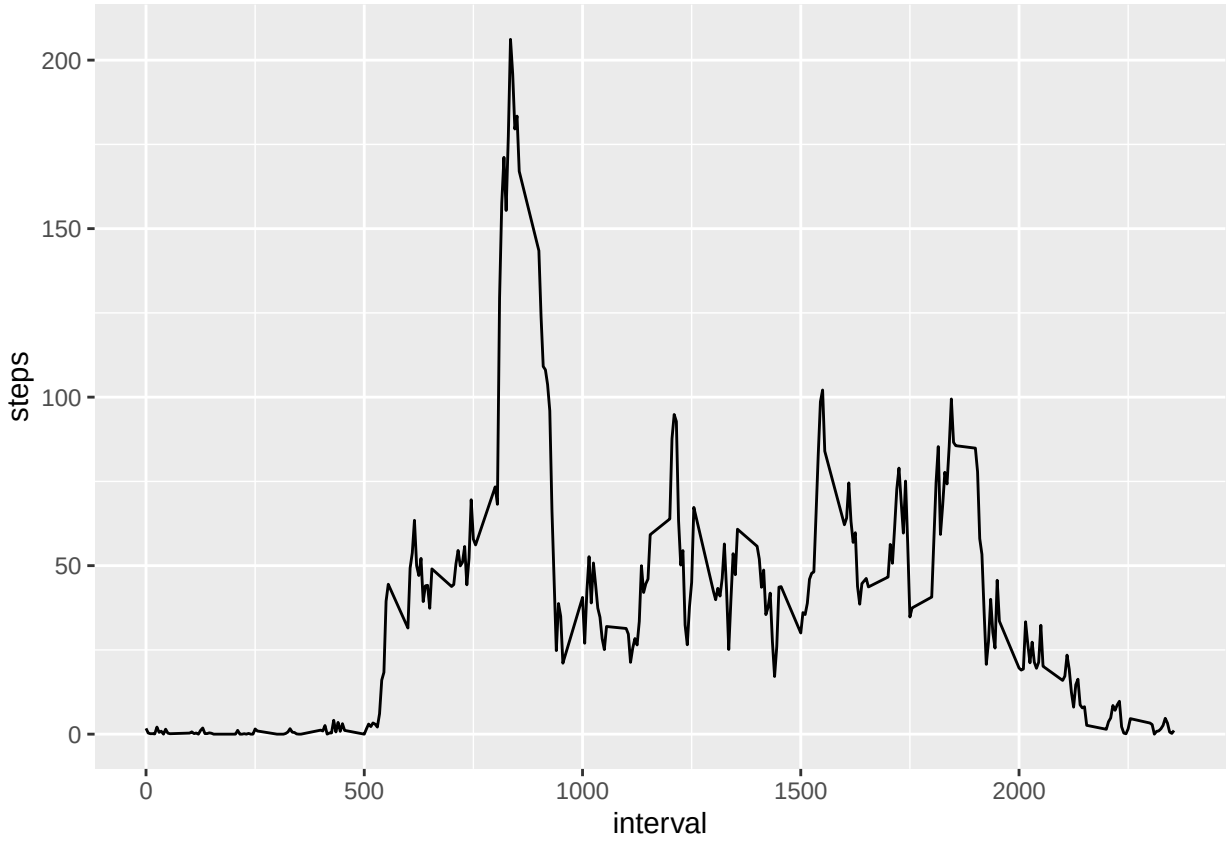
Time Series Plot

Time series plot of the average number of steps taken

```
avg_steps <- aggregate(data$steps, by=list(data$interval), FUN=mean, na.rm=T)

colnames(avg_steps) <- c("interval","steps")

ggplot(aes(x=interval, y=steps), data=avg_steps)+geom_line()
```



Maximum Average 5 minute interal

```
fiveminave <- avg_steps[avg_steps$steps==max(avg_steps$steps), 1]
```

The 5-minute interval that, on average, contains the maximum number of steps is 835

Imputing NA

Here is cosw to describe and show a strategy for imputing missing data

Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NA's)

```
missing <- sum(is.na(data$steps))
```

The total number of missing values in the dataset is 2304

Replace NA values with the mean of the steps

```
data$steps[is.na(data$steps)] <- mean(data$steps, na.rm = T)
```

Here are some roes of new data set

```
head(data)
```

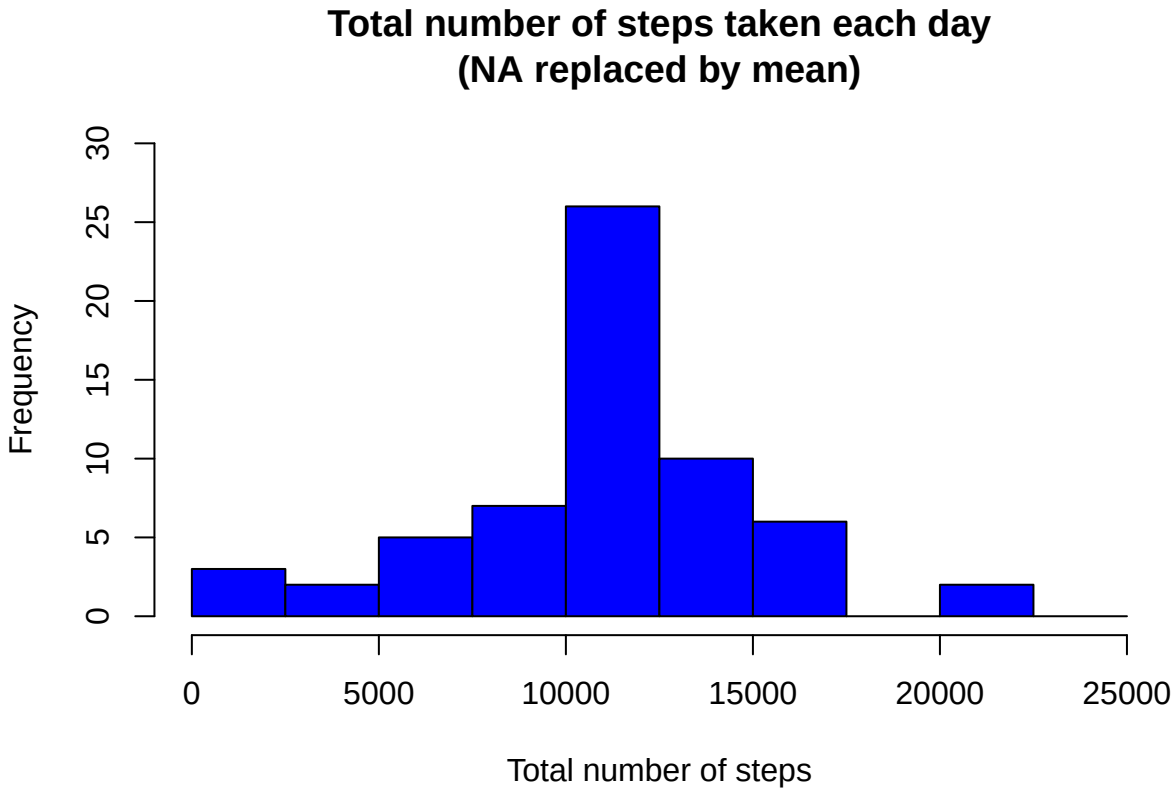
```
##     steps       date interval
## 1 37.3826 2012-10-01        0
## 2 37.3826 2012-10-01        5
## 3 37.3826 2012-10-01       10
## 4 37.3826 2012-10-01       15
## 5 37.3826 2012-10-01       20
## 6 37.3826 2012-10-01       25
```

Histogram with Replaced NA values

Histogram of total number of steps taken on each day is shown below:

```
sum_steps <- aggregate(data$steps, by=list(data$date), FUN=sum, na.rm = T)

hist(sum_steps$x,
      breaks=seq(from=0, to=25000, by=2500),
      col="blue",
      xlab="Total number of steps",
      ylim=c(0, 30),
      main="Total number of steps taken each day\n(NA replaced by mean)"
)
```



Mean and median number of steps taken each day after replacing NA values with mean

```
mean(sum_steps$x)
```

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

```
median(sum_steps$x)
```

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

Difference in activity patterns between weekdays and weekdends

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

```
# convert date into weekdays
data$days = tolower(weekdays(data$date))

# Now categorised days into weekend and weekdays

data$day_type <- ifelse(data$days == "saturday" | data$days== "sunday","weekend","weekday")

# take mean steps taken on weekend or weekday in the intervals
```

```
avg_steps <- aggregate(data$steps, by=list(data$interval, data$day_type), FUN=mean,na.rm=T)

colnames(avg_steps) <- c("interval", "day_type", "steps")

ggplot(aes(x=interval,y=steps), data=avg_steps) +geom_line()+facet_wrap(~avg_steps$day_type)
```