Reproducible Research Course Project No. 1

By Zahiruddin B. Zahidanishah

Published on Fri, Jun 05 2020, 17:18:52.

Introduction.

This assignment use data from a personal activity monitoring device such as Fitbit, Nike Fuelband or Jawbone Up. These type of devices are part of the "quantified self" movement - a group of enthusiasts who make measurements about themselves regularly to improve their health and to find pattern in their behavior.

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.

As part of the reproducible research criteria, this documents is produced in a single R markdown document using Rstudio and can be processed by knitr and converted to a HTML file for final submission.

Loading and preprocessing the data

Load the required packages for the entire assessment and set the global option with echo equals to TRUE so that code is visible to any anyone reading this markdown file.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(knitr)
opts_chunk$set(echo=TRUE)
```

The data for this assignment is save in a csv file name activity.csv.

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

The following shows the detail main summary and structure of the collected data:-

```
summary(activity)
```

```
##      steps            date              interval
##  Min.   :  0.00   Length:17568       Min.   :   0.0
##  1st Qu.:  0.00   Class :character   1st Qu.: 588.8
##  Median :  0.00   Mode  :character   Median :1177.5
##  Mean   : 37.38                      Mean   :1177.5
##  3rd Qu.: 12.00                      3rd Qu.:1766.2
##  Max.   :806.00                      Max.   :2355.0
##  NA's   :2304
```

```
str(activity)
```

```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : chr  "2012-10-01" "2012-10-01" "2012-10-01" "2012-10-01" ...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

Based on the above, the variables included in the datasets are as follow:-

1. **steps**: *Number of steps in a 5-minute interval (missing values are coded as NA)*
2. **date**: *The date on which the measurement was taken in YYYY-MM-DD format*
3. **interval**: *Identifier for the 5-minute interval in which measurement was taken*

What is mean total number of steps taken per day?

Based on the given datasets, the total daily number of steps from October to November are as detail below:-

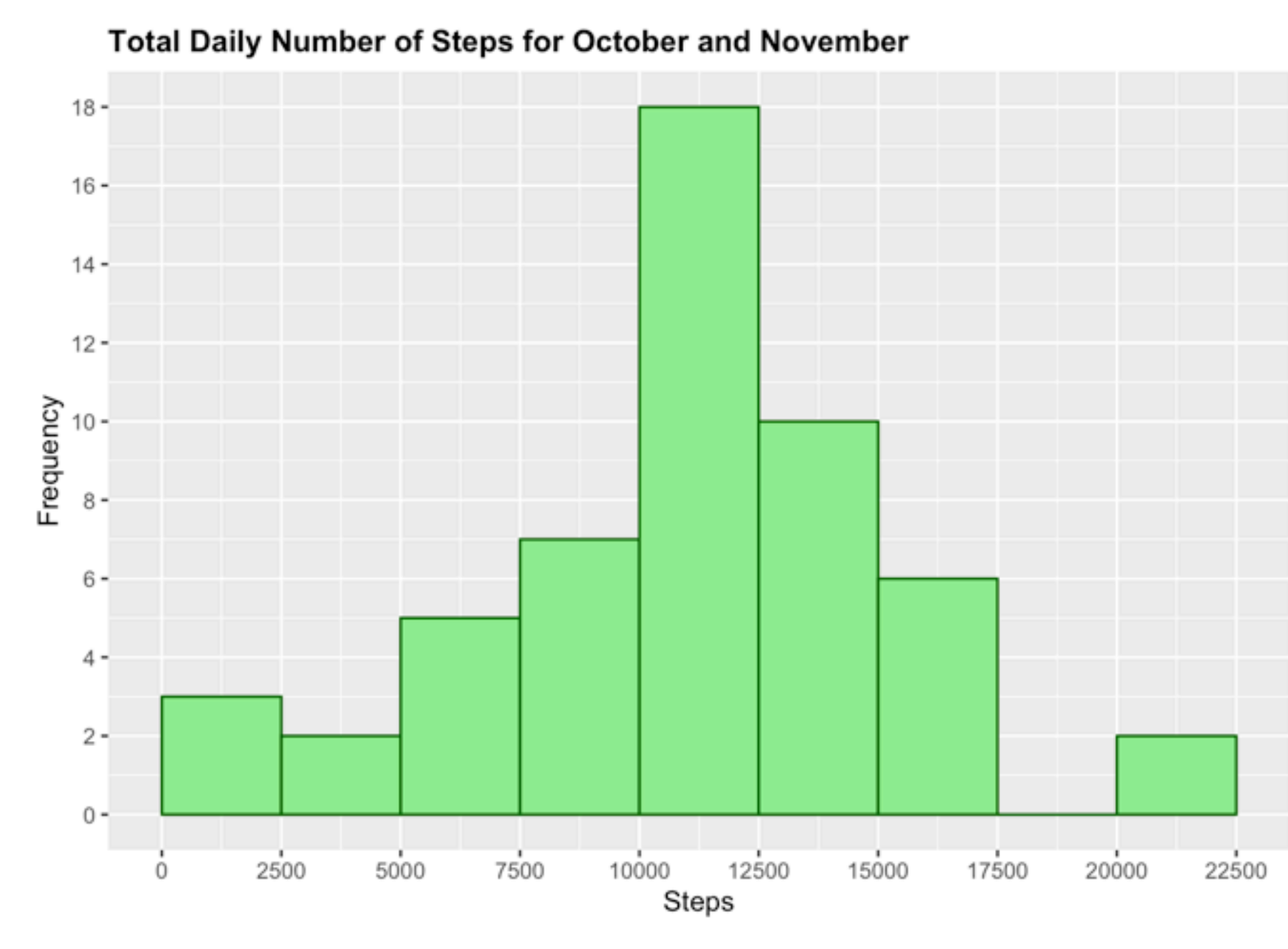
```
StepsPerDay <- aggregate(activity$steps, list(activity$date), FUN=sum)
colnames(StepsPerDay) <- c("Date", "Steps")
StepsPerDay
```

```
##          Date Steps
## 1  2012-10-01    NA
## 2  2012-10-02   126
## 3  2012-10-03 11352
## 4  2012-10-04 12116
## 5  2012-10-05 13294
## 6  2012-10-06 15420
## 7  2012-10-07 11015
## 8  2012-10-08    NA
## 9  2012-10-09 12811
## 10 2012-10-10  9900
## 11 2012-10-11 10304
## 12 2012-10-12 17382
## 13 2012-10-13 12426
## 14 2012-10-14 15098
## 15 2012-10-15 10139
## 16 2012-10-16 15084
## 17 2012-10-17 13452
## 18 2012-10-18 10056
## 19 2012-10-19 11829
## 20 2012-10-20 10395
## 21 2012-10-21  8821
## 22 2012-10-22 13460
## 23 2012-10-23  8918
## 24 2012-10-24  8355
## 25 2012-10-25  2492
## 26 2012-10-26  6778
## 27 2012-10-27 10119
## 28 2012-10-28 11458
## 29 2012-10-29  5018
## 30 2012-10-30  9819
## 31 2012-10-31 15414
## 32 2012-11-01    NA
## 33 2012-11-02 10600
## 34 2012-11-03 10571
## 35 2012-11-04    NA
## 36 2012-11-05 10439
## 37 2012-11-06  8334
## 38 2012-11-07 12883
## 39 2012-11-08  3219
## 40 2012-11-09    NA
## 41 2012-11-10    NA
## 42 2012-11-11 12608
## 43 2012-11-12 10765
## 44 2012-11-13  7336
## 45 2012-11-14    NA
## 46 2012-11-15    41
## 47 2012-11-16  5441
## 48 2012-11-17 14339
## 49 2012-11-18 15110
## 50 2012-11-19  8841
## 51 2012-11-20  4472
## 52 2012-11-21 12787
## 53 2012-11-22 20427
## 54 2012-11-23 21194
## 55 2012-11-24 14478
## 56 2012-11-25 11834
## 57 2012-11-26 11162
## 58 2012-11-27 13646
## 59 2012-11-28 10183
## 60 2012-11-29  7047
## 61 2012-11-30    NA
```

The following shows the histogram plot for the total daily number of steps for October and November:-

```
g <- ggplot(StepsPerDay, aes(Steps))
g+geom_histogram(boundary=0, binwidth=2500, col="darkgreen", fill="lightgreen")+ggtitle("Total Daily Number of St
eps for October and November")+xlab("Steps")+ylab("Frequency")+theme(plot.title = element_text(face="bold", size=
12))+scale_x_continuous(breaks=seq(0,25000,2500))+scale_y_continuous(breaks=seq(0,18,2))
```

```
## Warning: Removed 8 rows containing non-finite values (stat_bin).
```



Therefore, the **mean** of steps take each day are as follow:-

```
mean(StepsPerDay$Steps, na.rm=TRUE)
```

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

And the **median** of steps take each day are as follow:-

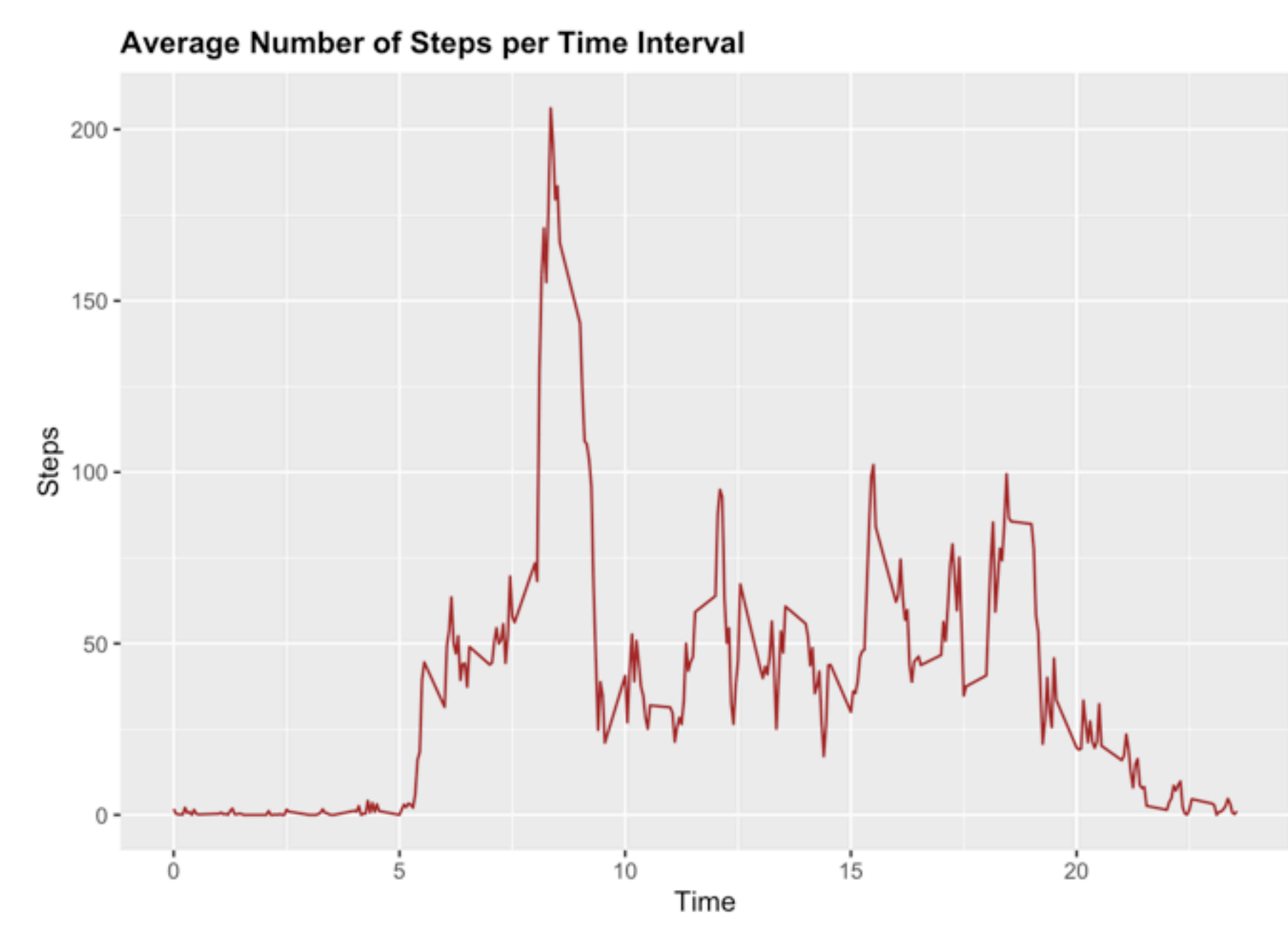
```
median(StepsPerDay$Steps, na.rm=TRUE)
```

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

What is the average daily activity pattern?

The time series plot of the average number of steps taken are as follow:-

```
StepsPerTime <- aggregate(steps~interval,data=activity,FUN=mean,na.action=na.omit)
StepsPerTime$time <- StepsPerTime$interval/100
h <- ggplot(StepsPerTime, aes(time, steps))
h+geom_line(col="brown")+ggtitle("Average Number of Steps per Time Interval")+xlab("Time")+ylab("Steps")+theme(pl
ot.title = element_text(face="bold", size=12))
```



The 5 minutes interval on average across all the days which contains the maximum number of steps are as follow:-

```
ST <- tbl_df(StepsPerTime)
```

```
## Warning: `tbl_df()` is deprecated as of dplyr 1.0.0.
## Please use `tibble::as_tibble()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

```
ST %>% select(time, steps) %>% filter(steps==max(ST$steps))
```

```
## # A tibble: 1 x 2
##    time steps
##   <dbl> <dbl>
## 1  8.35  206.
```

Imputing missing values

Based on the total daily number of steps taken in October and November shows that there are a number of days/intervals where there are a missing values (NA). The presence of this missing values may introduce bias to some calculations or summaries of the data.

The total number of missing values in the datasets are as follow:-

```
ACT <- tbl_df(activity)
ACT %>% filter(is.na(steps)) %>% summarize(missing_values = n())
```

```
## # A tibble: 1 x 1
##   missing_values
##            <int>
## 1           2304
```

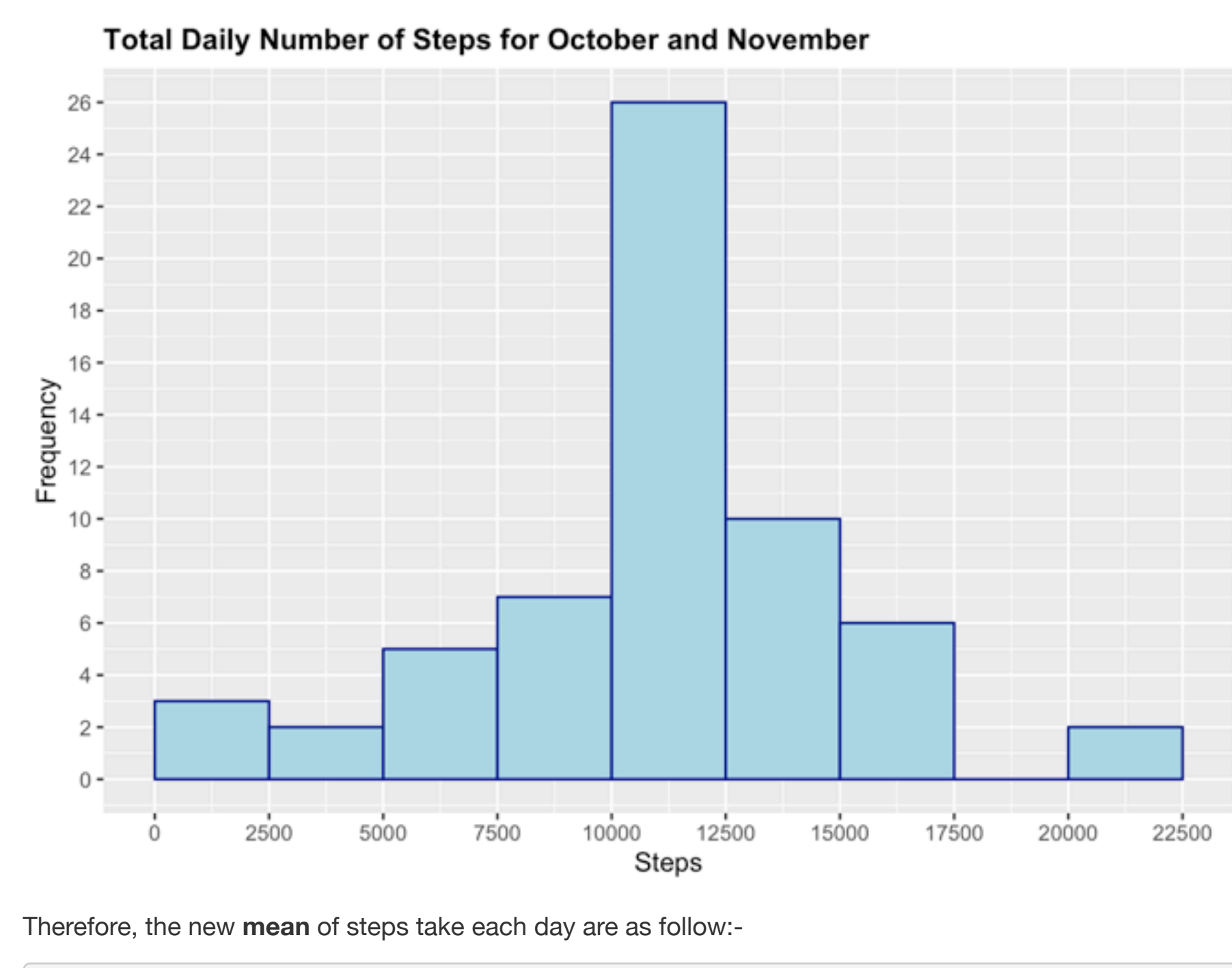
Therefore, the new datasets after imputing the missing values.

```
activity$CompleteSteps <- ifelse(is.na(activity$steps), round(StepsPerTime$steps[match(activity$interval, StepsPe
rTime$interval)],0), activity$steps)
activityFull <- data.frame(steps=activity$CompleteSteps, interval=activity$interval, date=activity$date)
head(activityFull, n=10)
```

```
##    steps interval       date
## 1      2        0 2012-10-01
## 2      0        5 2012-10-01
## 3      0       10 2012-10-01
## 4      0       15 2012-10-01
## 5      0       20 2012-10-01
## 6      2       25 2012-10-01
## 7      1       30 2012-10-01
## 8      1       35 2012-10-01
## 9      0       40 2012-10-01
## 10     1       45 2012-10-01
```

The new histogram plot based on the new datasets are as follow:-

```
StepsPerDayFull <- aggregate(activityFull$steps, list(activityFull$date), FUN=sum)
colnames(StepsPerDayFull) <- c("Date", "Steps")
g <- ggplot(StepsPerDayFull, aes(Steps))
g+geom_histogram(boundary=0, binwidth=2500, col="darkblue", fill="lightblue")+ggtitle("Total Daily Number of Step
s for October and November")+xlab("Steps")+ylab("Frequency")+theme(plot.title = element_text(face="bold", size=12
))+scale_x_continuous(breaks=seq(0,25000,2500))+scale_y_continuous(breaks=seq(0,26,2))
```



Therefore, the new **mean** of steps take each day are as follow:-

```
mean(StepsPerDayFull$Steps)
```

```
## [1] 10765.64
```

And the new **median** of steps take each day are as follow:-

```
median(StepsPerDayFull$Steps)
```

```
## [1] 10762
```

Are there differences in activity patterns between weekdays and weekends?

In order to identify the different activity patterns between weekday and weekends, a new factor variable in the datasets is created indicating weekdays and weekends. The following shows the first 10 rows set of data.

```
activityFull$RealDate <- as.Date(activityFull$date, format = "%Y-%m-%d")
activityFull$weekday <- weekdays(activityFull$RealDate)
activityFull$DayType <- ifelse(activityFull$weekday=='Saturday' | activityFull$weekday=='Sunday', 'Weekends','Wee
kdays')
head(activityFull, n=10)
```

```
##    steps interval       date   RealDate weekday  DayType
## 1      2        0 2012-10-01 2012-10-01  Monday Weekdays
## 2      0        5 2012-10-01 2012-10-01  Monday Weekdays
## 3      0       10 2012-10-01 2012-10-01  Monday Weekdays
## 4      0       15 2012-10-01 2012-10-01  Monday Weekdays
## 5      0       20 2012-10-01 2012-10-01  Monday Weekdays
## 6      2       25 2012-10-01 2012-10-01  Monday Weekdays
## 7      1       30 2012-10-01 2012-10-01  Monday Weekdays
## 8      1       35 2012-10-01 2012-10-01  Monday Weekdays
## 9      0       40 2012-10-01 2012-10-01  Monday Weekdays
## 10     1       45 2012-10-01 2012-10-01  Monday Weekdays
```

Based on this, a new plot to shows both average daily steps on weekdays and weekends are as follow:-

```
StepsPerTimeDT <- aggregate(steps~interval+DayType,data=activityFull,FUN=mean,na.action=na.omit)
StepsPerTimeDT$time <- StepsPerTime$interval/100
j <- ggplot(StepsPerTimeDT, aes(time, steps))
j+geom_line(col="darkred")+ggtitle("Average Steps per Time Interval: Weekdays vs. Weekends")+xlab("Time")+ylab("S
teps")+theme(plot.title = element_text(face="bold", size=12))+facet_grid(DayType ~ .)
```