Reproducible Research - Project 2

Synopsis

Data Processing

Include the code to import the csv file and transform the data so that it can be visualized.

```
library(ggplot2)
```

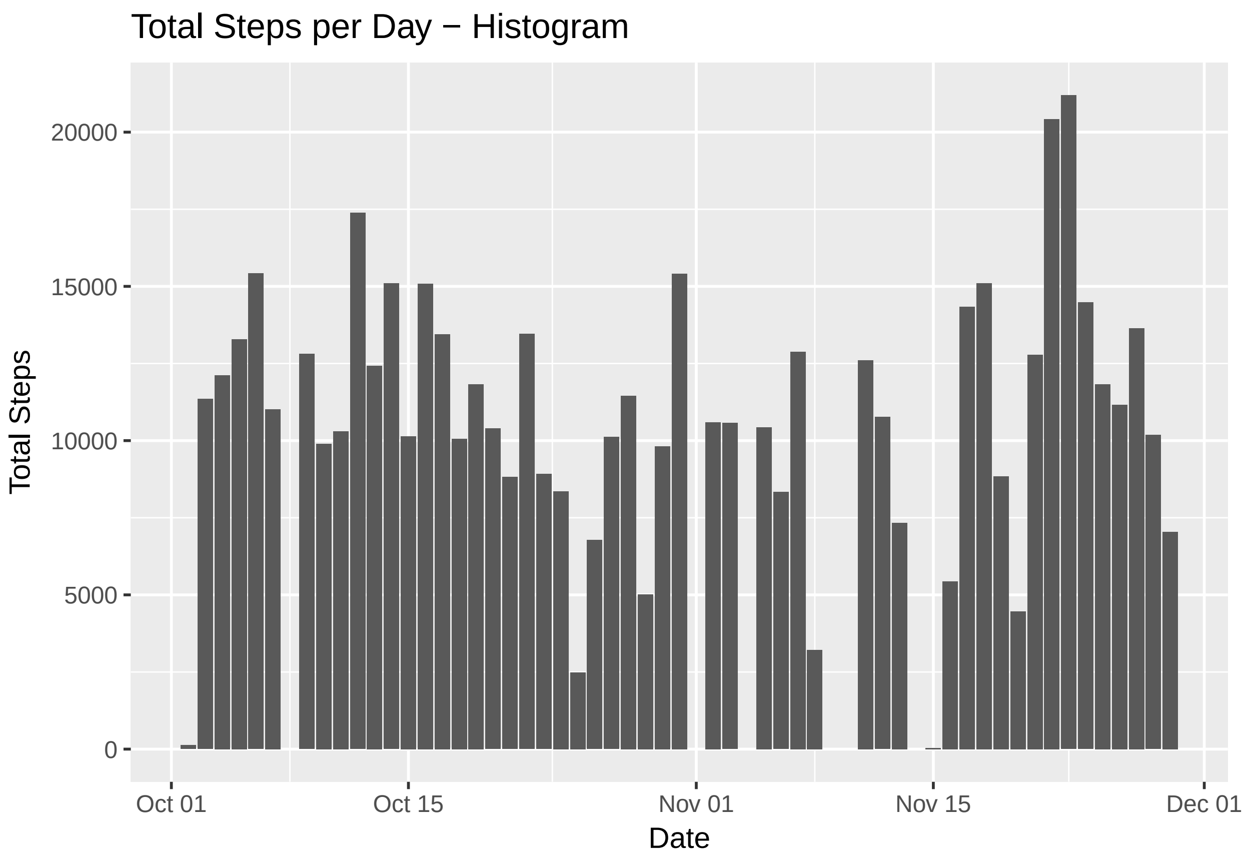
```
## Warning: package 'ggplot2' was built under R version 3.3.3
```

```
#Code for reading in the data set
activity<-read.csv("activity.csv")

#remove N/A steps
hact<-activity[is.na(activity["steps"])==FALSE,]

#Fix Date Field
#aggregate the steps per date:
ag<-aggregate(x=hact[1],by =list(hact$date),FUN="sum")
names(ag)[1]="dt"
ag[1]<-as.Date(ag$dt, format="%Y-%m-%d")

#Now figure out the average number of steps for 5 minute intervals
#per day
fvmin<-aggregate(x=hact[1],by =list(hact$date),FUN="mean")
names(fvmin)[1]="date"
#now aggregate on interval
intag<-aggregate(x=hact[1],by =list(hact$interval),FUN="mean")
names(intag)[1]="interval"
#find interval where the max is
```

Including Plots

Histogram for total number of steps

```
ggplot(data=ag, aes(x=dt,y=steps))+geom_bar(stat = "identity")+labs(title = "Total Steps per Day - Histo
```



Mean and Median Steps per day

```
mn<-mean(ag$steps)
md<-median(ag$steps)
paste("The mean number of total steps per day is",mn)
```

```
## [1] "The mean number of total steps per day is 10766.1886792453"
```

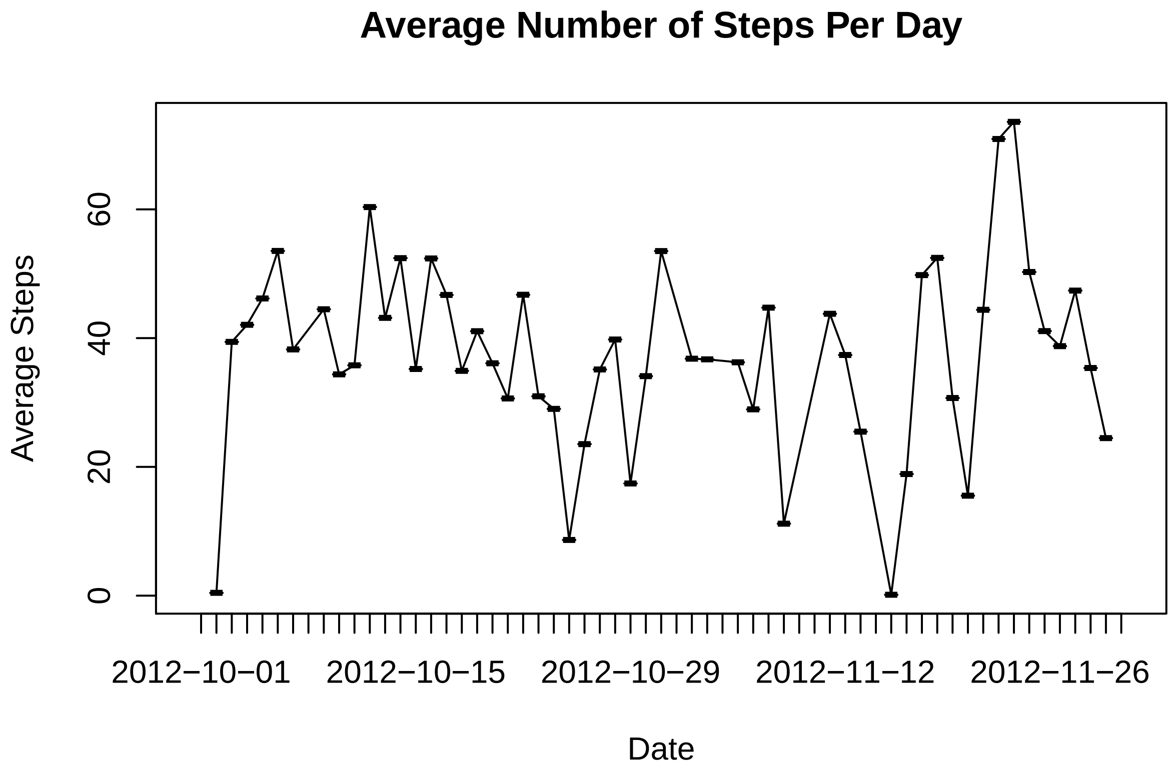
```
paste("The median number of total steps per day is",md)
```

```
## [1] "The median number of total steps per day is 10765"
```

Results

What is the average daily activity pattern?

```
plot(x=fvmin$date,y=fvmin$steps,type="l",main="Average Number of Steps Per Day",xlab="Date", ylab="Avera
lines(x=fvmin$date,y=fvmin$steps,type="l")
```



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

```
mx<-intag[intag$steps==max(intag$steps),1]

paste("The maximum number of average steps was at interval",mx)
```

```
## [1] "The maximum number of average steps was at interval 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.

```
# Calculate number of missing items in the dataset

stepNA<-activity[is.na(activity["steps"])==TRUE,]
totStepNA<-NROW(stepNA)
dtNA<-activity[is.na(activity["date"])==TRUE,]
totDtNa<-NROW(dtNA)
intNA<-activity[is.na(activity["interval"])==TRUE,]
totIntNa<-NROW(intNA)
paste("There are",totStepNA, "records that are missing step counts.")
```

```
## [1] "There are 2304 records that are missing step counts."
```

```
paste("There are",totDtNa, "records that are missing dates.")
```

```
## [1] "There are 0 records that are missing dates."
```

```r
paste("There are",totIntNa, "records that are missing intervals.")
```

```
## [1] "There are 0 records that are missing intervals."
```

```r
#Fill in the missing data using the mean step count for the day and interval that the steps are missing

library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.3.2
```

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

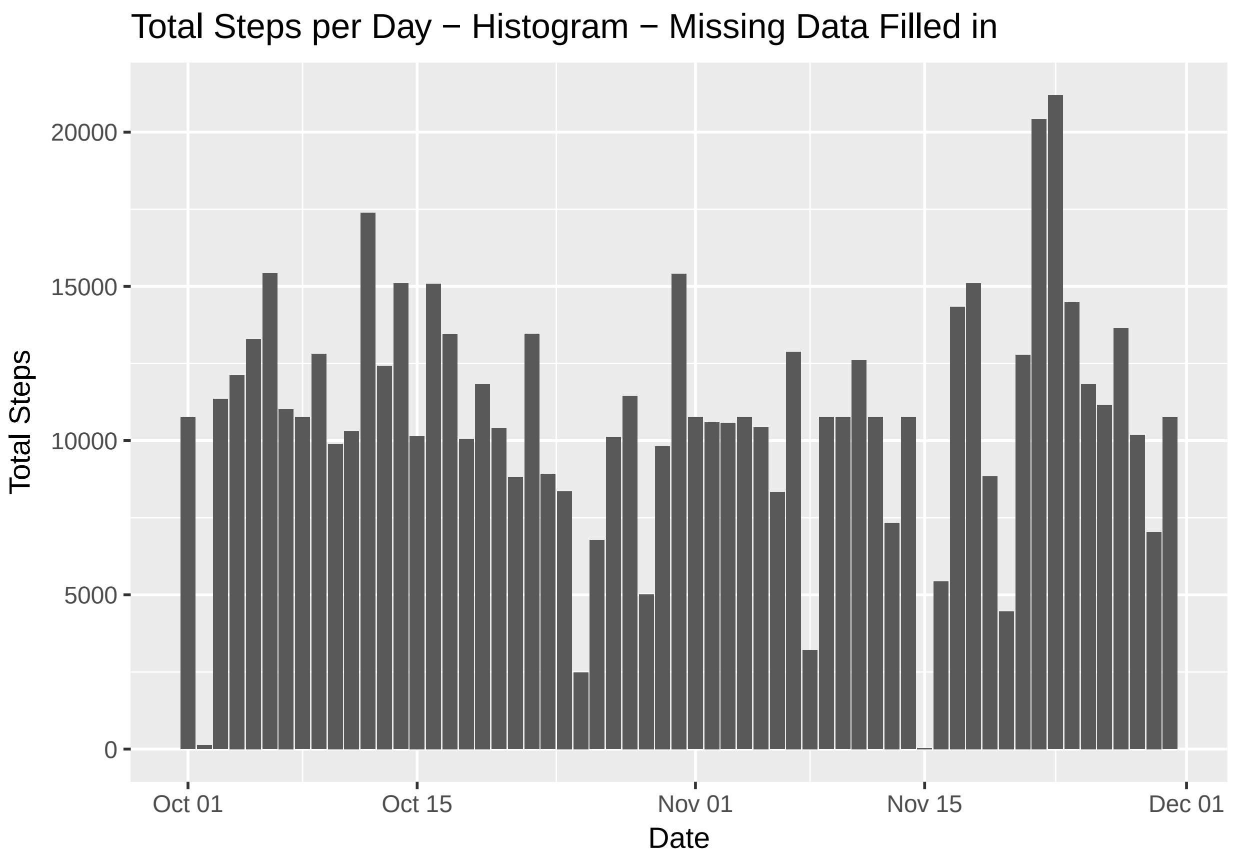
```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

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

```r
stepNA<-stepNA[order(stepNA$interval),]
fillNA<-merge(stepNA,intag,by.x = "interval",by.y = "interval",all=TRUE)
fillNA<-fillNA[c(4,3,1)]
names(fillNA)[1]="steps"
#Now combine with data set
fillNAds<-rbind(hact,fillNA)
#redo the aggregation with the new data set
#aggregate the steps per date:
ag<-aggregate(x=fillNAds[1],by =list(fillNAds$date),FUN="sum")
names(ag)[1]="dt"
ag[1]<-as.Date(ag$dt, format="%Y-%m-%d")

#Now figure out the average number of steps for 5 minute intervals
#per day
fvmin<-aggregate(x=fillNAds[1],by =list(fillNAds$date),FUN="mean")
names(fvmin)[1]="date"
#now aggregate on interval
intag<-aggregate(x=fillNAds[1],by =list(fillNAds$interval),FUN="mean")
names(intag)[1]="interval"
```

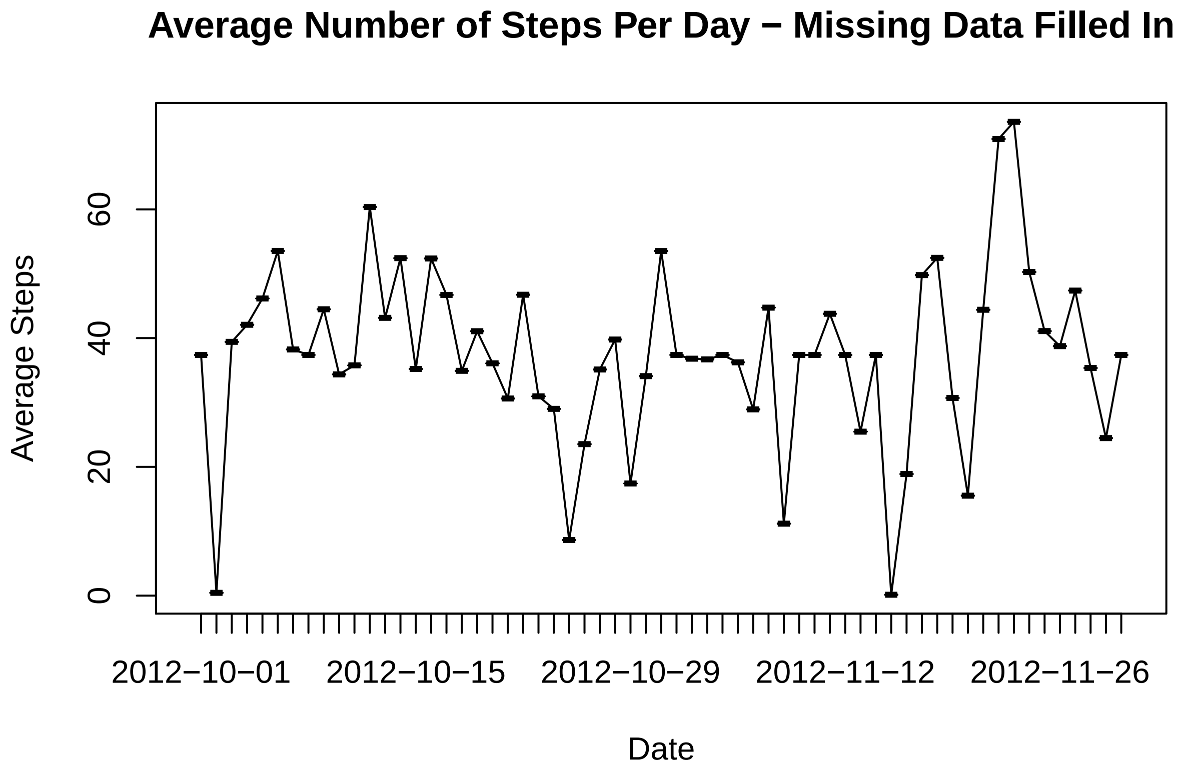
New Histogram with missing data filled in

```r
ggplot(data=ag, aes(x=dt,y=steps))+geom_bar(stat = "identity")+labs(title = "Total Steps per Day - Hist
```



New interval chart with missing data filled in

```
plot(x=fvmin$date,y=fvmin$steps,type="l",main="Average Number of Steps Per Day - Missing Data Filled In
lines(x=fvmin$date,y=fvmin$steps,type="l")
```



Strategy for replacing missing values: - Find the mean values for each interval - Replace the missing values with the mean value for the given interval and day of the missing value. Conclussion - Filling in the missing data did not impact the overall view of the data.

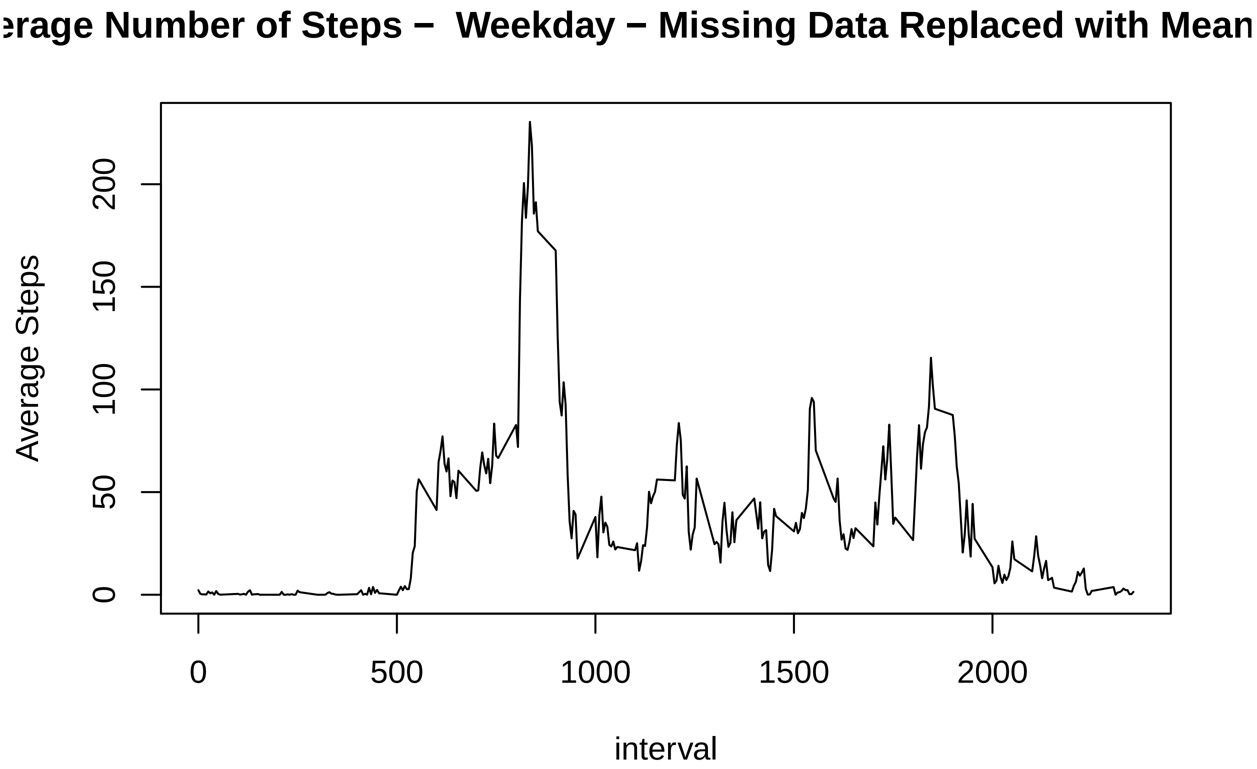
Are there differences in activity patterns between weekdays and weekends? Yes, the weekends show a significant increase in the number of steps.

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

```
#Add weekday column, it will be Y/N value
#convert factor to date field
fillNAds[2]<-as.Date(fillNAds$date, format="%Y-%m-%d")
#get name of day for date
fillNAds[4]<-weekdays(fillNAds$date)
#identify weekdays vs weekends
fillNAds[5]<-fillNAds[4]!="Saturday" & fillNAds[4]!="Sunday"
names(fillNAds)[4:5]=c("Day","IsWD")
wd<-fillNAds[fillNAds$IsWD==TRUE,]
we<-fillNAds[fillNAds$IsWD==FALSE,]
agwd<-aggregate(x=wd[1],by =list(wd$interval),FUN="mean")
agwe<-aggregate(x=we[1],by =list(we$interval),FUN="mean")
names(agwd)[1]="interval"
names(agwe)[1]="interval"
```

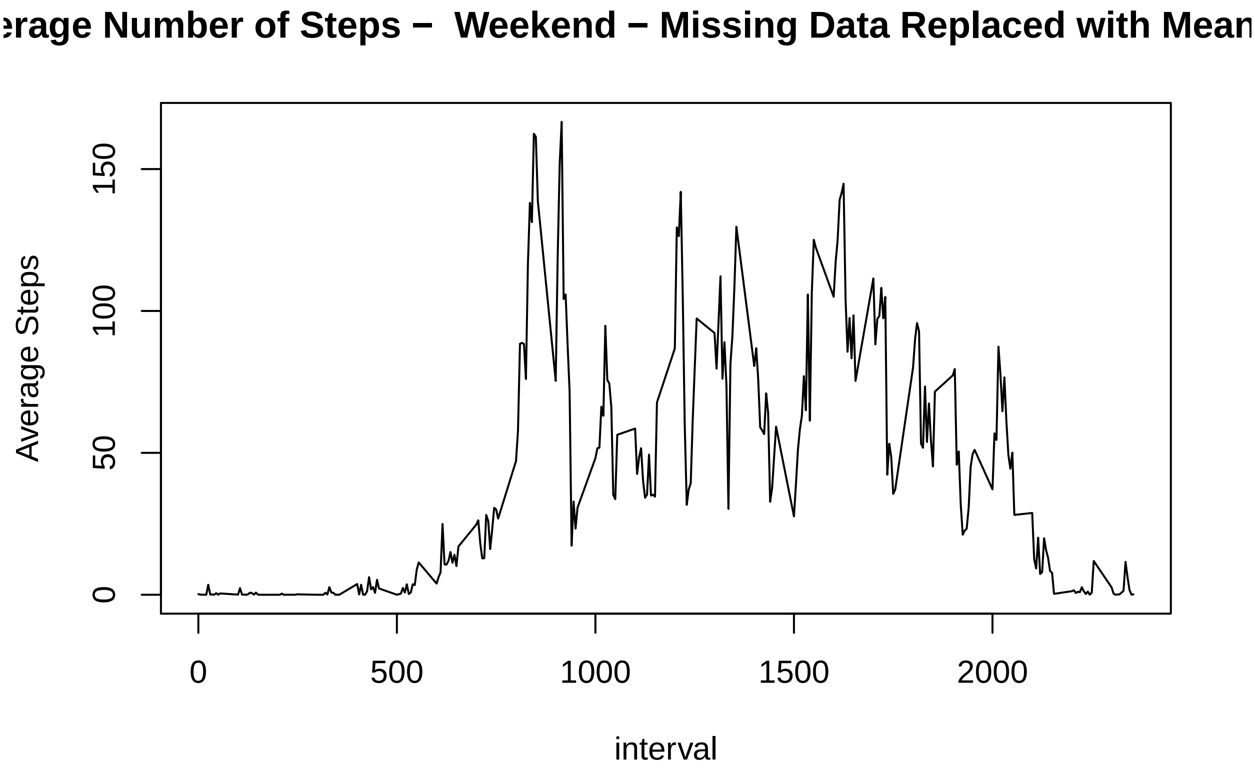
Show average steps on weekdays with the missign data replaced by mean values

```
plot(x=agwd$interval,y=agwd$steps,type="l",main="Average Number of Steps -  Weekday - Missing Data Repla
```



Show average steps on Weekends with replaced missing values

```
plot(x=agwe$interval,y=agwe$steps,type="l",main="Average Number of Steps -  Weekend - Missing Data Repla
```


erage Number of Steps − Weekend − Missing Data Replaced with Mean
Average Steps
0
50
100
150
0
500
1000
1500
2000
interval