RR_project1

Yazad Jal

Downloading the zipfile from the web and saving it in the working directory

```
fileurl <- "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip"
destfile <- paste0(getwd(),"/","activity_data.zip")
download.file(fileurl, destfile, method = "curl", quiet = TRUE)
```

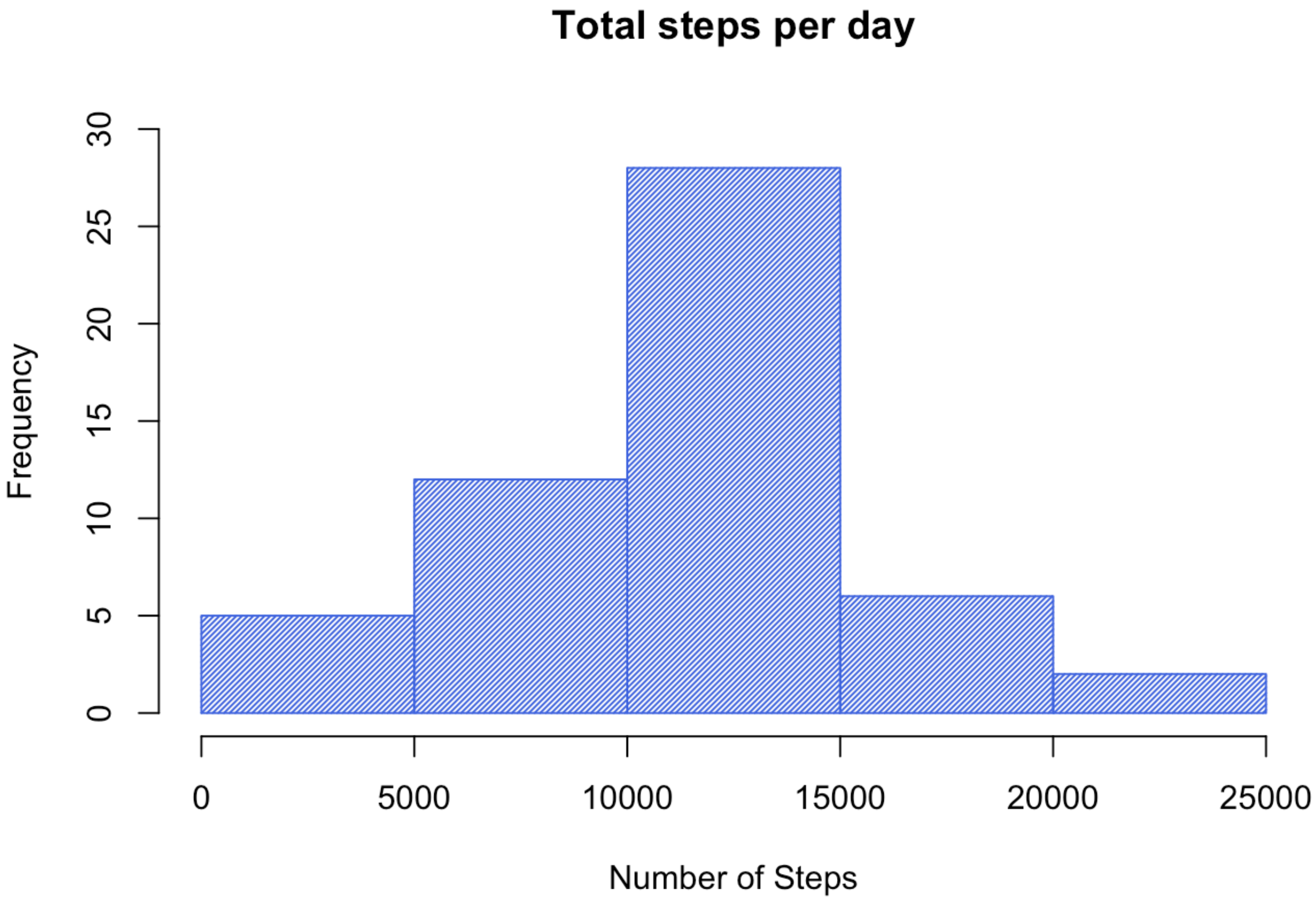
Unziping the downloaded file and getting a feel for the data:

```
unzip("activity_data.zip")
activity <- read.csv("activity.csv")
str(activity)
```

```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 1 1 1 1 1 1 1 1 1 1
...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

To calculate total number steps per day, I created a new data frame called totalsteps and then made a histogram.

```
totalsteps<-aggregate(steps~date,data=activity,sum,na.rm=TRUE)
hist(totalsteps$steps,
        col = "royalblue", border = "royalblue", density = 50,
        xlab = "Number of Steps", main = "Total steps per day",
        ylim = c(0,30))
```



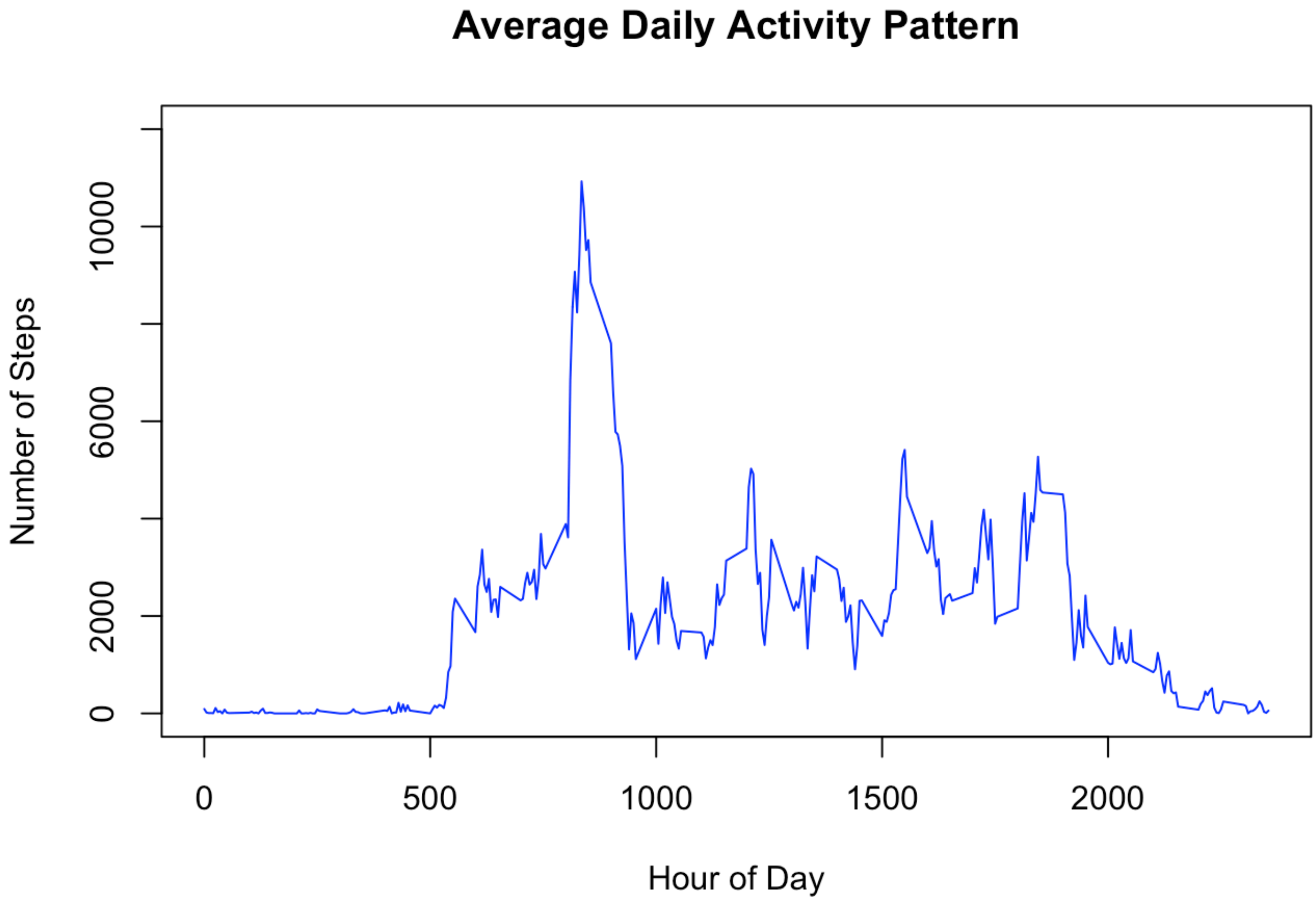
Calculating the mean and median of the total number of steps taken per day

```
mn <- mean(totalsteps$steps)
md <- median(totalsteps$steps)
```

Mean = 10766.19 and median is 10765

Five minute intervals and plot

```
fivemin <- aggregate(steps~interval, data = activity, sum, na.rm=TRUE)
plot(fivemin$interval, fivemin$steps, type = "l", col = "blue",
     xlab="Hour of Day", ylab="Number of Steps",
     main="Average Daily Activity Pattern", ylim = c(0,12000))
```



Max 5 min interval

```
max <- fivemin[which.max(fivemin$steps),]$interval
max2 <-paste0(0, max)
```

0835 is when the max exercise happens.

Imputing using impute function from Hmisc package

```
library(Hmisc, quietly = TRUE)
activityImputed <- activity
activityImputed$steps <- impute(activity$steps, fun=mean)
totalsteps2<-aggregate(steps~date,data=activityImputed,sum,na.rm=TRUE)
```

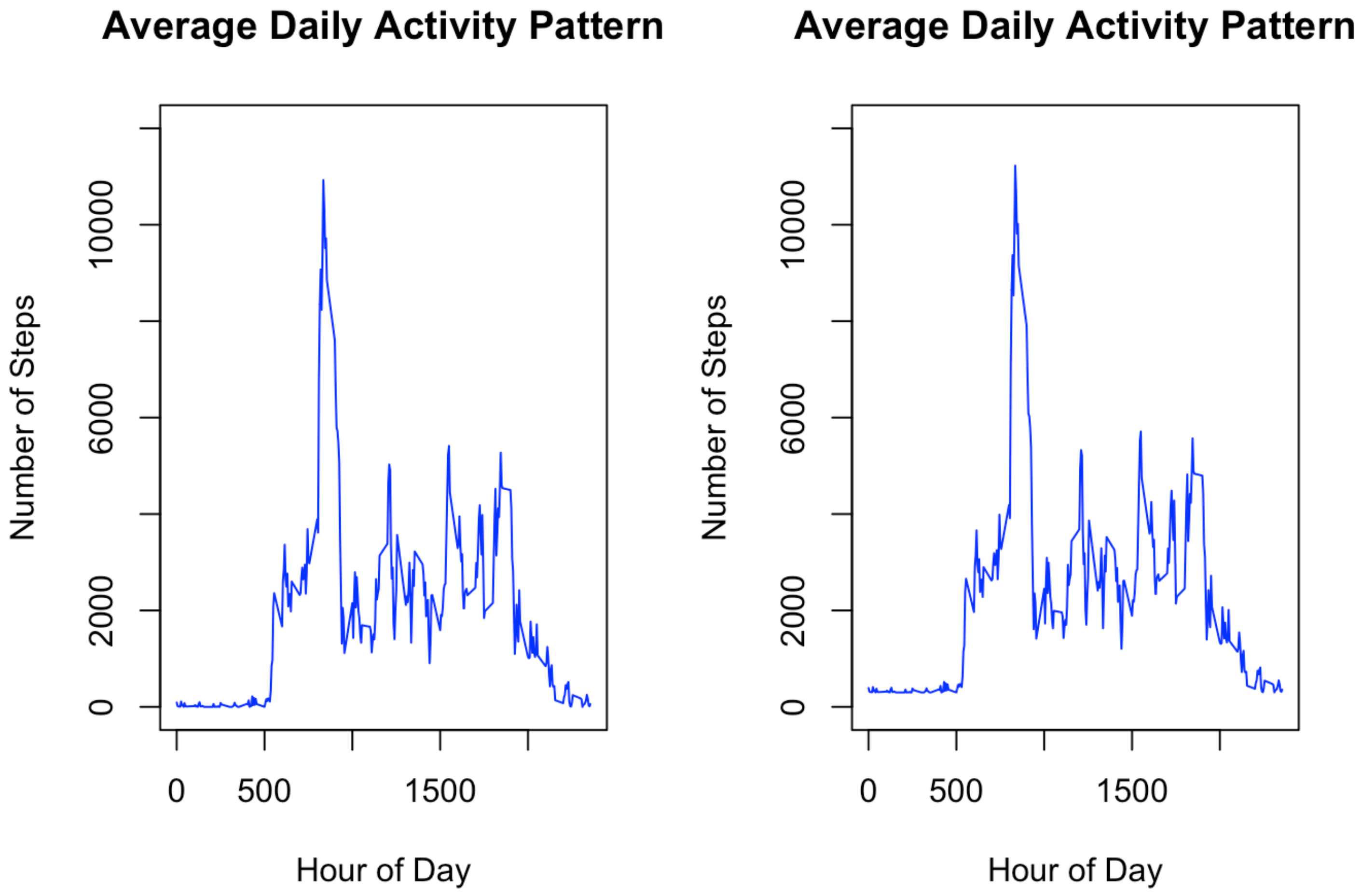
New mean and Median

```
mn2 <- mean(totalsteps2$steps)
median(totalsteps2$steps)
```

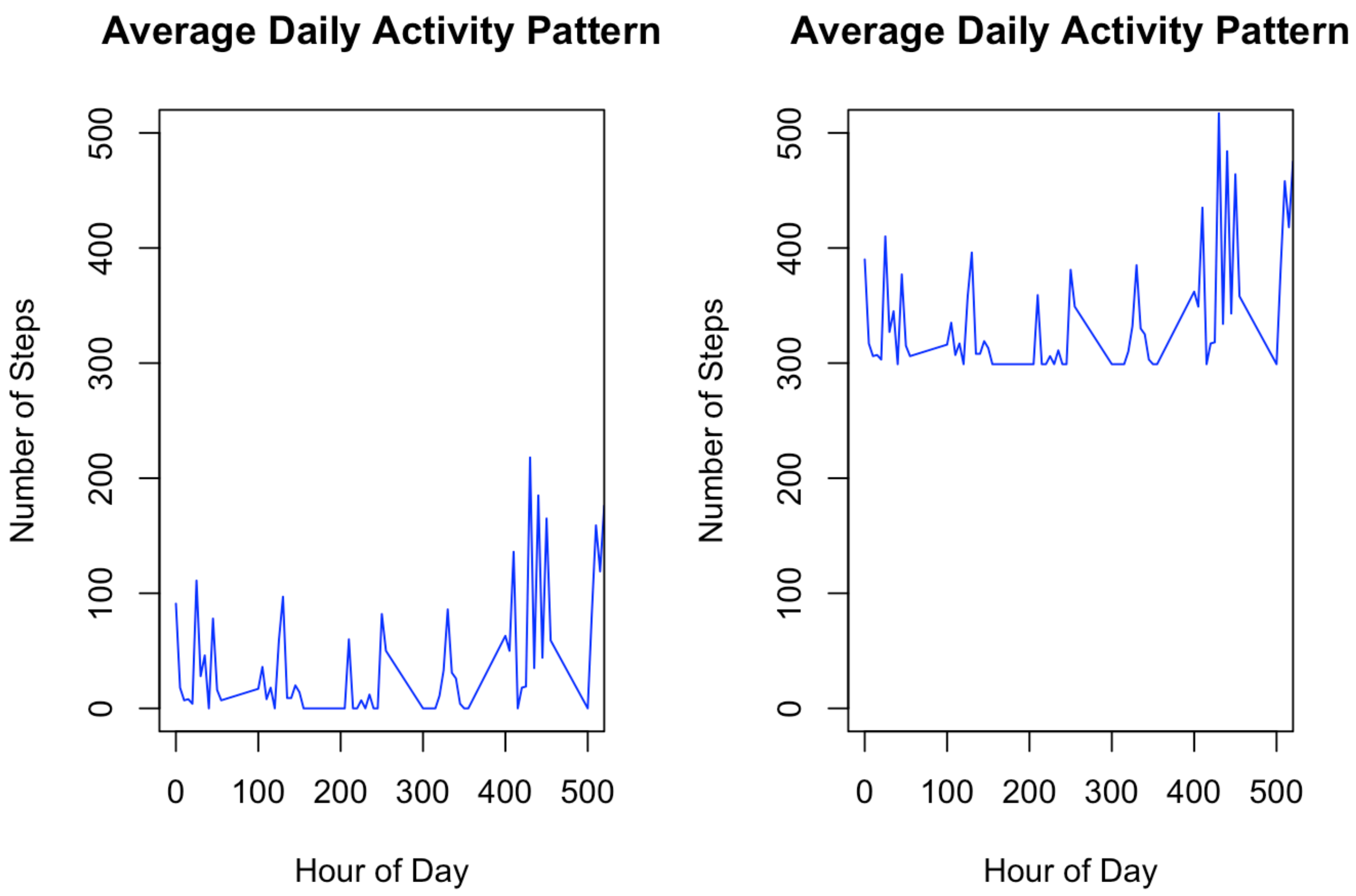
```
## [1] 10766.19
```

the new mean is 1.076618910^{4}

```
fivemin2 <- aggregate(steps~interval, data = activityImputed, sum, na.rm=TRUE)
par(mfrow=c(1,2))
plot(fivemin$interval, fivemin$steps, type = "l", col = "blue",
     xlab="Hour of Day", ylab="Number of Steps",
     main="Average Daily Activity Pattern", ylim = c(0,12000))
plot(fivemin2$interval, fivemin2$steps, type = "l", col = "blue",
     xlab="Hour of Day", ylab="Number of Steps",
     main="Average Daily Activity Pattern", ylim = c(0,12000))
```



```
plot(fivemin$interval, fivemin$steps, type = "l", col = "blue",
     xlab="Hour of Day", ylab="Number of Steps",
     main="Average Daily Activity Pattern", ylim = c(0,500),
     xlim=c(0,500))
plot(fivemin2$interval, fivemin2$steps, type = "l", col = "blue",
     xlab="Hour of Day", ylab="Number of Steps",
     main="Average Daily Activity Pattern", ylim = c(0,500),
     xlim = c(0,500))
```



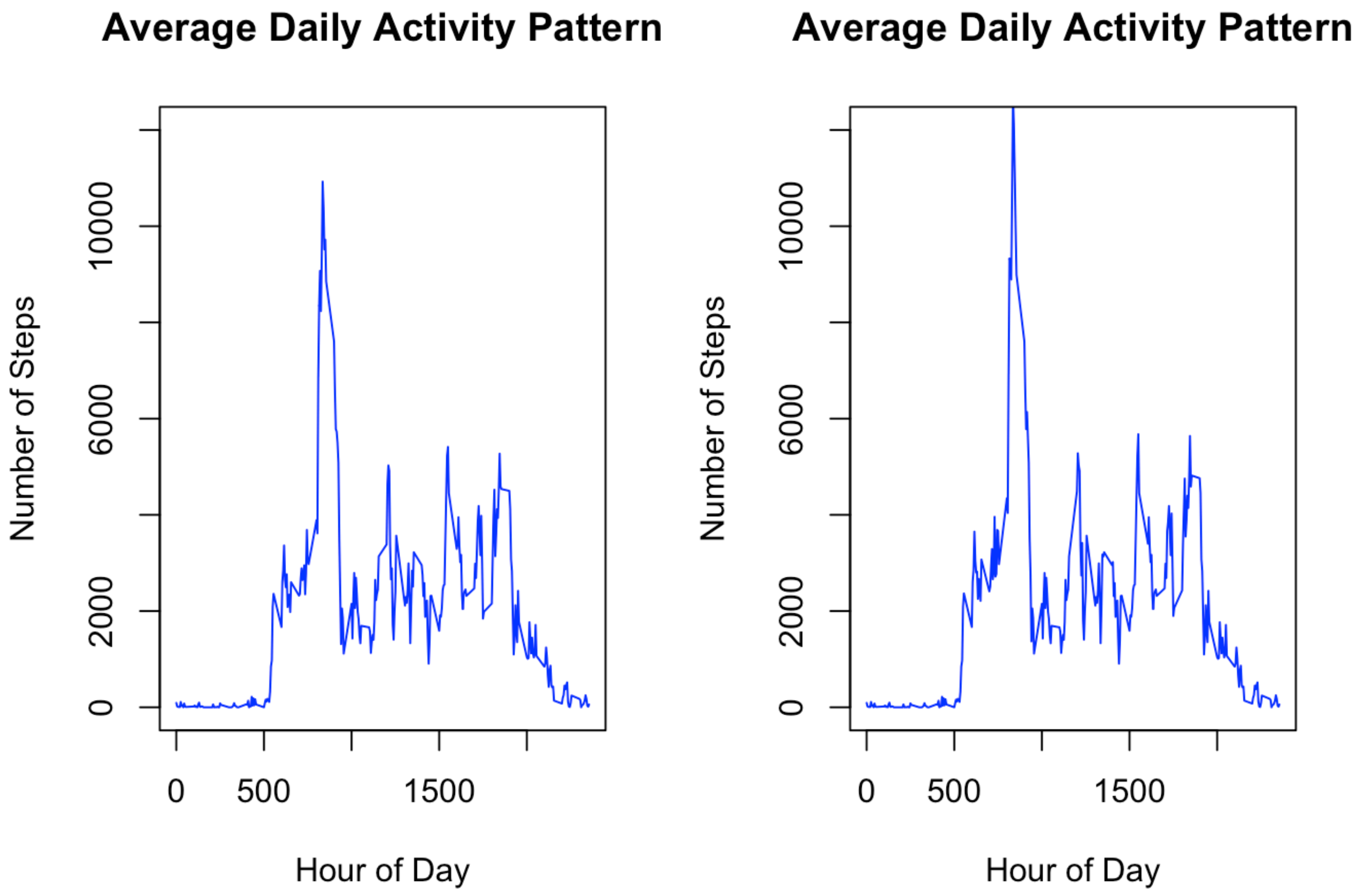
```
par(mfrow=c(1,1))
```

Instead lets impute on using kNN from the package VIM

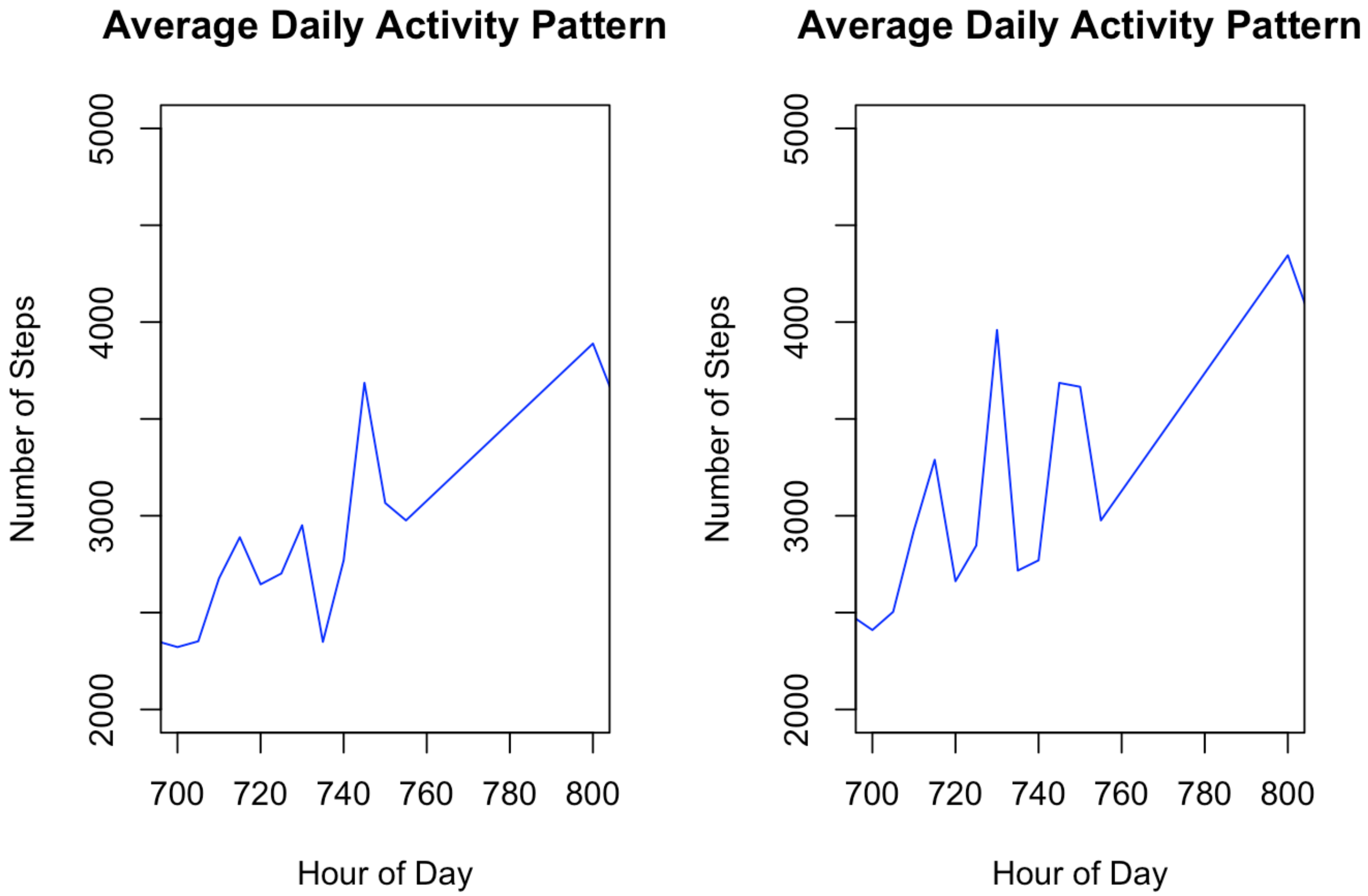
```
library(VIM)
act <- activity
knnact <- kNN(act)
fivemin3 <- aggregate(steps~interval, data = knnact, sum, na.rm=TRUE)
```

replotting

```
par(mfrow=c(1,2))
plot(fivemin$interval, fivemin$steps, type = "l", col = "blue",
     xlab="Hour of Day", ylab="Number of Steps",
     main="Average Daily Activity Pattern", ylim = c(0,12000))
plot(fivemin3$interval, fivemin3$steps, type = "l", col = "blue",
     xlab="Hour of Day", ylab="Number of Steps",
     main="Average Daily Activity Pattern", ylim = c(0,12000))
```



```
plot(fivemin$interval, fivemin$steps, type = "l", col = "blue",
     xlab="Hour of Day", ylab="Number of Steps",
     main="Average Daily Activity Pattern", ylim = c(2000,5000),
     xlim=c(700,800))
plot(fivemin3$interval, fivemin3$steps, type = "l", col = "blue",
     xlab="Hour of Day", ylab="Number of Steps",
     main="Average Daily Activity Pattern", ylim = c(2000,5000),
     xlim = c(700,800))
```



```
par(mfrow=c(1,1))
```

Lets check if the mean and median remain the same

```
totalsteps3<-aggregate(steps~date,data=knnact,sum,na.rm=TRUE)
mean(totalsteps3$steps)
```

```
## [1] 9752.393
```

```
median(totalsteps3$steps)
```

```
## [1] 10395
```

new mean is 9752, old mean was 1.076610^{4} – the mean has decreased significantly by