````
---
title: "represearch"
output: html_document
---

```{r setup, include=TRUE}
knitr::opts_chunk$set(echo = TRUE, warning = FALSE, fig.width = 10, fig.height = 5,
                      fig.keep = 'all' ,fig.path = 'figures\ ', dev = 'png')
```

## R Markdown

activity <- read.csv("activity.csv")
activity$date <- as.POSIXct(activity$date, "%Y%m%d")
day <- weekdays(activity$date)
activity <- cbind(activity, day)
summary(activity)


```{r activity,echo=TRUE}
summary(activity)
```

```{r activity, echo=TRUE}

## Including Plots
g <- ggplot(totalStepsdf, aes(x = Steps)) +
+     geom_histogram(breaks = seq(0, 25000, by = 2500), fill = "#83CAFF", col = "black") +
+     ylim(0, 30) +
+     xlab("Total Steps Taken Per Day") +
+     ylab("Frequency") +
+     ggtitle("Total Number of Steps Taken on a Day")

## The mean of the total number of steps taken per day is:
mean(activityTotalSteps$Steps)

##The median of the total number of steps taken per day is:
median(activityTotalSteps$Steps)

#Calculating the average number of steps taken, averaged across all days by 5-min
intervals.
averageDailyActivity <- aggregate(activity$steps, by = list(activity$interval),
                                  FUN = mean, na.rm = TRUE)

#Changing col names
names(averageDailyActivity) <- c("Interval", "Mean")

#Converting the data set into a dataframe
averageActivitydf <- data.frame(averageDailyActivity)

#Plotting on ggplot2
da <- ggplot(averageActivitydf, mapping = aes(Interval, Mean)) +
  geom_line(col = "blue") +
  xlab("Interval") +
  ylab("Average Number of Steps") +
  ggtitle("Average Number of Steps Per Interval")

  print(da)

sum(is.na(activity$steps))
# Matching the mean of daily activity with the missing values
imputedSteps <- averageDailyActivity$Mean[match(activity$interval,
averageDailyActivity$Interval)]
## Transforming steps in activity if they were missing values with the filled values from
````
````

```
above.
activityImputed <- transform(activity,
                             steps = ifelse(is.na(activity$steps), yes = imputedSteps, no
= activity$steps))

#Forming the new dataset with the imputed missing values.
totalActivityImputed <- aggregate(steps ~ date, activityImputed, sum)
#Changing col names
names(totalActivityImputed) <- c("date", "dailySteps")

#Testing new dataset
sum(is.na(totalActivityImputed$dailySteps))

# Converting the data set into a data frame to be able to use ggplot2
totalImputedStepsdf <- data.frame(totalActivityImputed)

# Plotting a histogram using ggplot2
p <- ggplot(totalImputedStepsdf, aes(x = dailySteps)) +
  geom_histogram(breaks = seq(0, 25000, by = 2500), fill = "#83CAFF", col = "black") +
  ylim(0, 30) +
  xlab("Total Steps Taken Per Day") +
  ylab("Frequency") +
  ggtitle("Total Number of Steps Taken on a Day")
  print(p)

##The mean of the total number of steps taken per day is:

mean(totalActivityImputed$dailySteps)

##The median of the total number of steps taken per day is:

median(totalActivityImputed$dailySteps)

#Updating format of the dates
activity$date <- as.Date(strptime(activity$date, format="%Y-%m-%d"))

# Creating a function that distinguises weekdays from weekends

activity$dayType <- sapply(activity$date, function(x) {
  if(weekdays(x) == "Saturday" | weekdays(x) == "Sunday")
  {y <- "Weekend"}
  else {y <- "Weekday"}
  y
})

#Creating the data set that will be plotted
activityByDay <-  aggregate(steps ~ interval + dayType, activity, mean, na.rm = TRUE)

# Plotting using ggplot2
dayPlot <-  ggplot(activityByDay, aes(x = interval , y = steps, color = dayType)) +
  geom_line() + ggtitle("Average Daily Steps by Day Type") +
  xlab("Interval") +
  ylab("Average Number of Steps") +
  facet_wrap(~dayType, ncol = 1, nrow=2) +
  scale_color_discrete(name = "Day Type")
```

```{r pressure, echo=TRUE}
plot(pressure)
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.