Reproducible Research Project 1

Reading the data and loading library

```
library(ggplot2)
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.5.1
```

```
##
## 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
```

```
act<- read.csv("/users/andrewhu/desktop/activity.csv")
```

What is mean total number of steps taken per day?

```
#Calculate the total number of steps taken per day and return a df
step_sum<- aggregate(act[c("steps")],list(date=act$date),sum,na.rm=T)

#Calculate and report the mean and median of the total number of steps taken per day
mean(step_sum$steps)
```

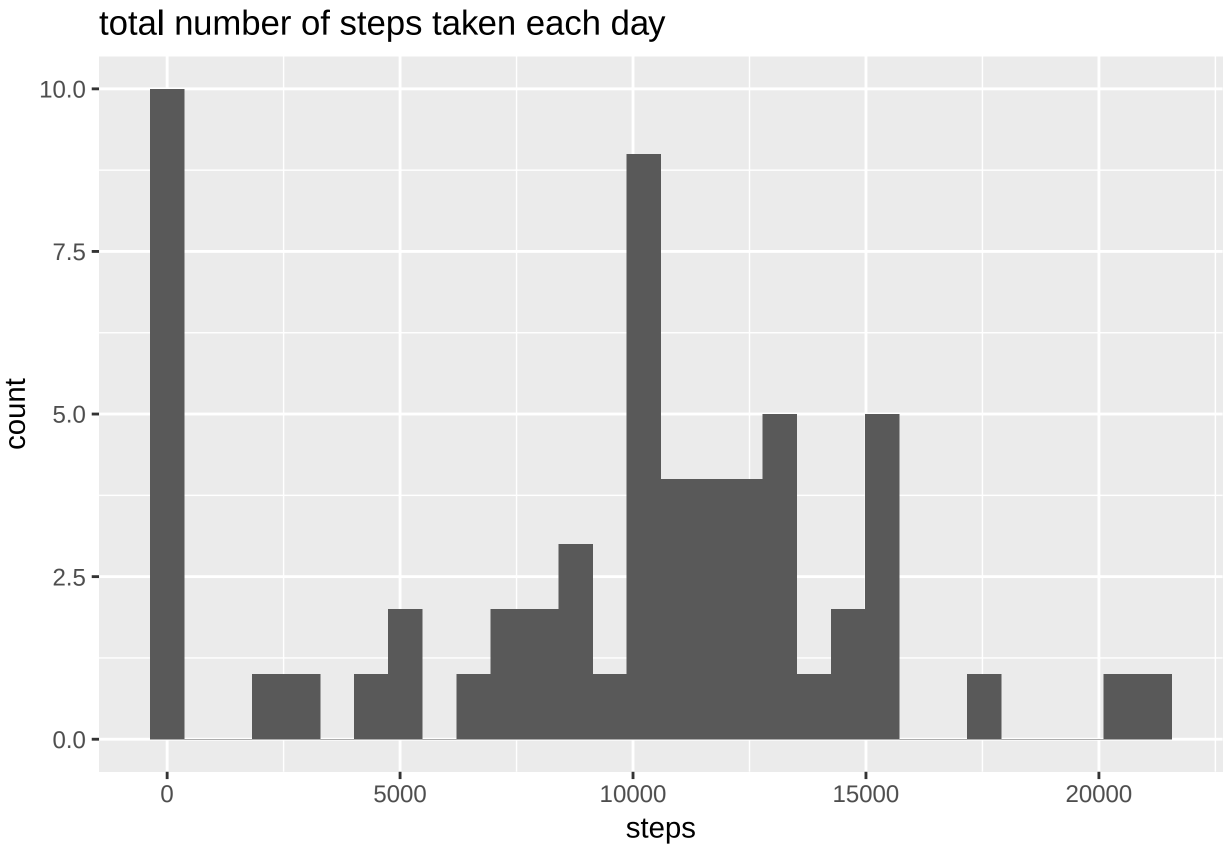
```
## [1] 9354.23
```

```
median(step_sum$steps)
```

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

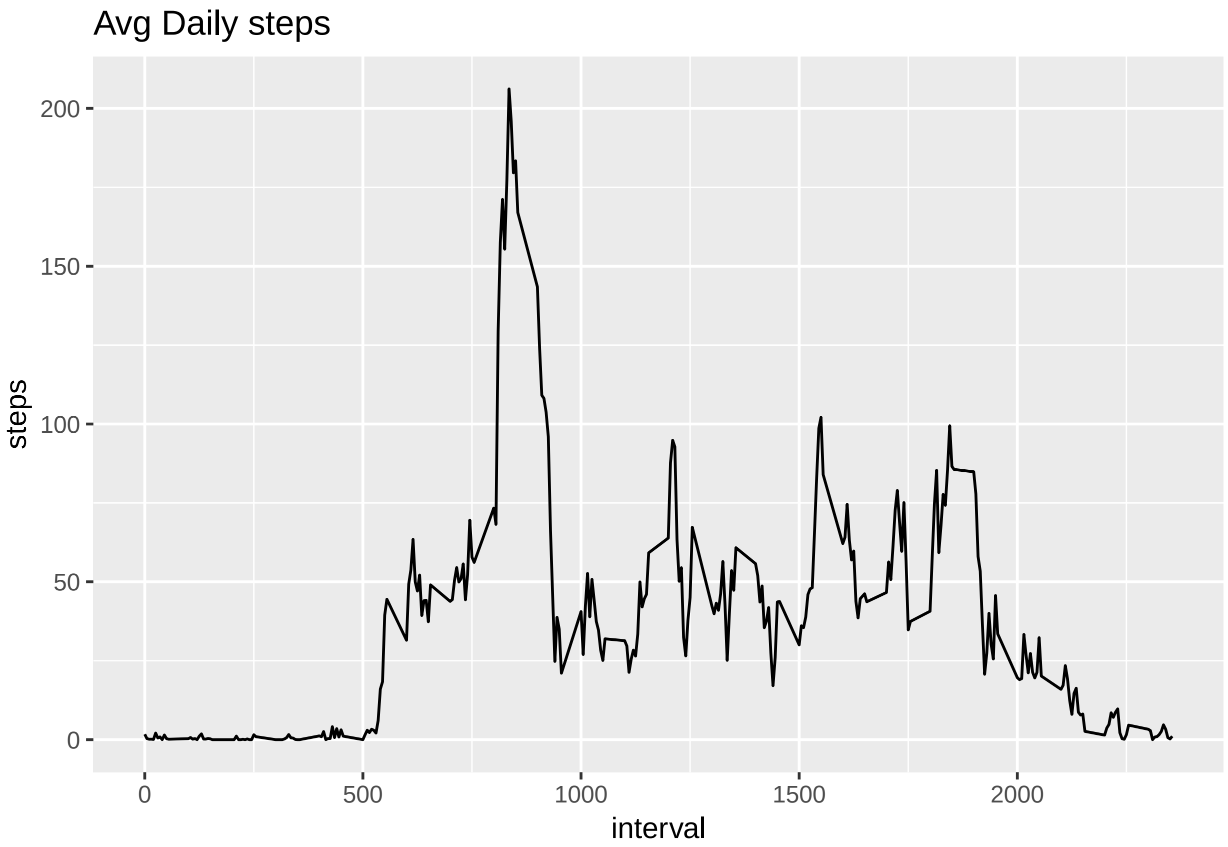
```
#Histogram of the total number of steps taken each day
g<- ggplot(step_sum, aes(steps))
g + geom_histogram()+labs(title= "total number of steps taken each day")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



What is the average daily activity pattern? Make a time series plot (i.e. type="l") of the 5-minute interval (x-axis)
and the average number of steps taken, averaged across all days (y-axis)

```
time_series <- with(act, tapply(steps, interval, sum, na.rm=T))
interval_df <- aggregate(act[c("steps")], list(interval= act$interval), mean, na.rm=T )
##ggplot
g<- ggplot(interval_df, aes(interval,steps) )
g + geom_line()+labs(title="Avg Daily steps", x="interval", y="steps")
```



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

```
max_interval <- which.max(time_series )
names(max_interval)
```

```
## [1] "835"
```

Imputing missing values

Calculate and report the total number of missing values in the dataset

```
sum(is.na(act))
```

```
## [1] 2304
```

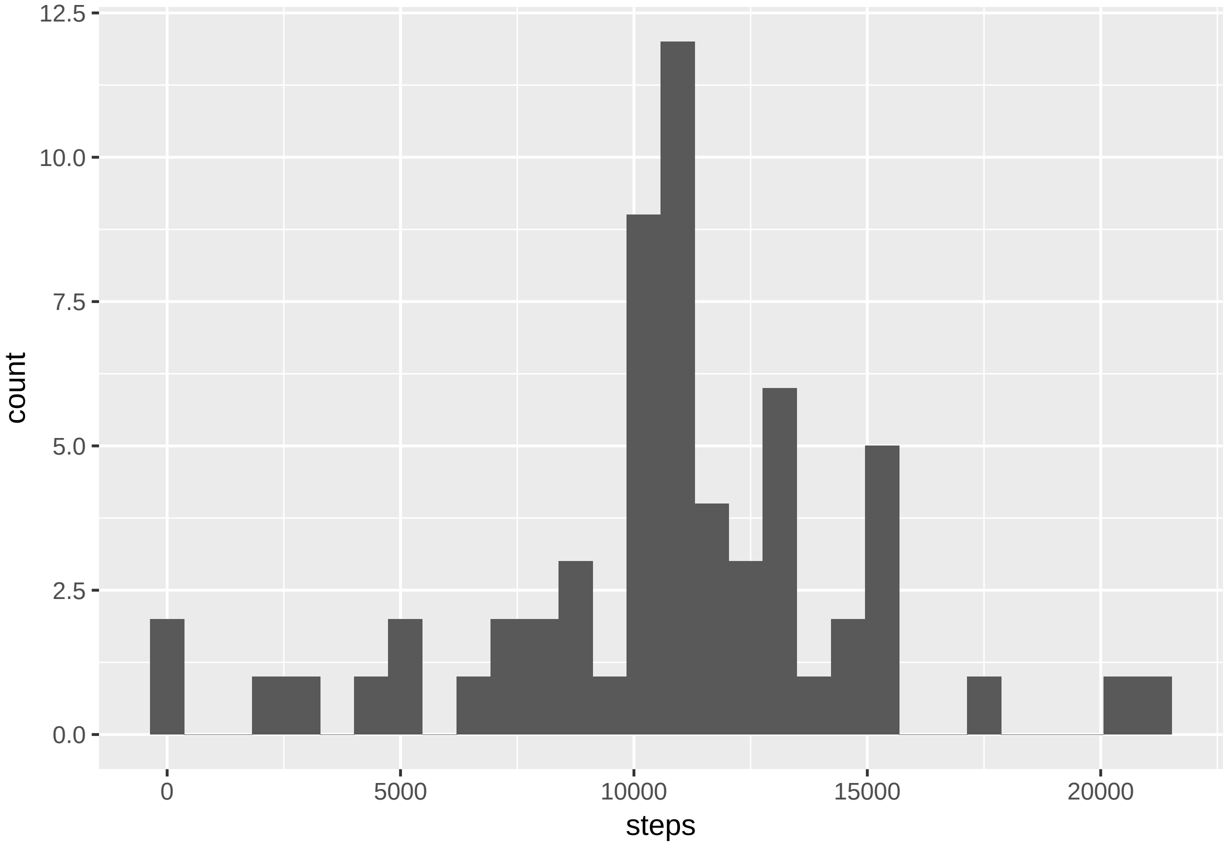
Devise a strategy for filling in all of the missing values in the dataset. The strategy does not need to be sophisticated. For example, you could use the mean/median for that day, or the mean for that 5-minute interval, etc. Create a new dataset that is equal to the original dataset but with the missing data filled in

```
act2<- act
act2$steps = ifelse(is.na(act2$steps), mean(act2$steps, na.rm=TRUE), act2$steps)
act2$interval = ifelse(is.na(act2$interval), mean(act2$interval, na.rm=TRUE), act2$interval)
```

Make a histogram of the total number of steps taken each day and Calculate and report the mean and median total number of steps taken per day. Do these values differ from the estimates from the first part of the assignment? What is the impact of imputing missing data on the estimates of the total daily number of steps?

```
#Calculate the total steps per day and return a df.
s_sum2 <- aggregate(act2[c("steps")],list(date=act2$date),sum)
#Making the histogram of total steps
g<- ggplot(s_sum2, aes(steps) )
g + geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
#Mean os steps per day
mean(s_sum2$steps)
```

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

```
#Median of steps per day
median(s_sum2$steps)
```

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

Are there differences in activity patterns between weekdays and weekends?

Create a new factor variable in the dataset with two levels – "weekday" and "weekend" indicating whether a given date is a weekday or weekend day.

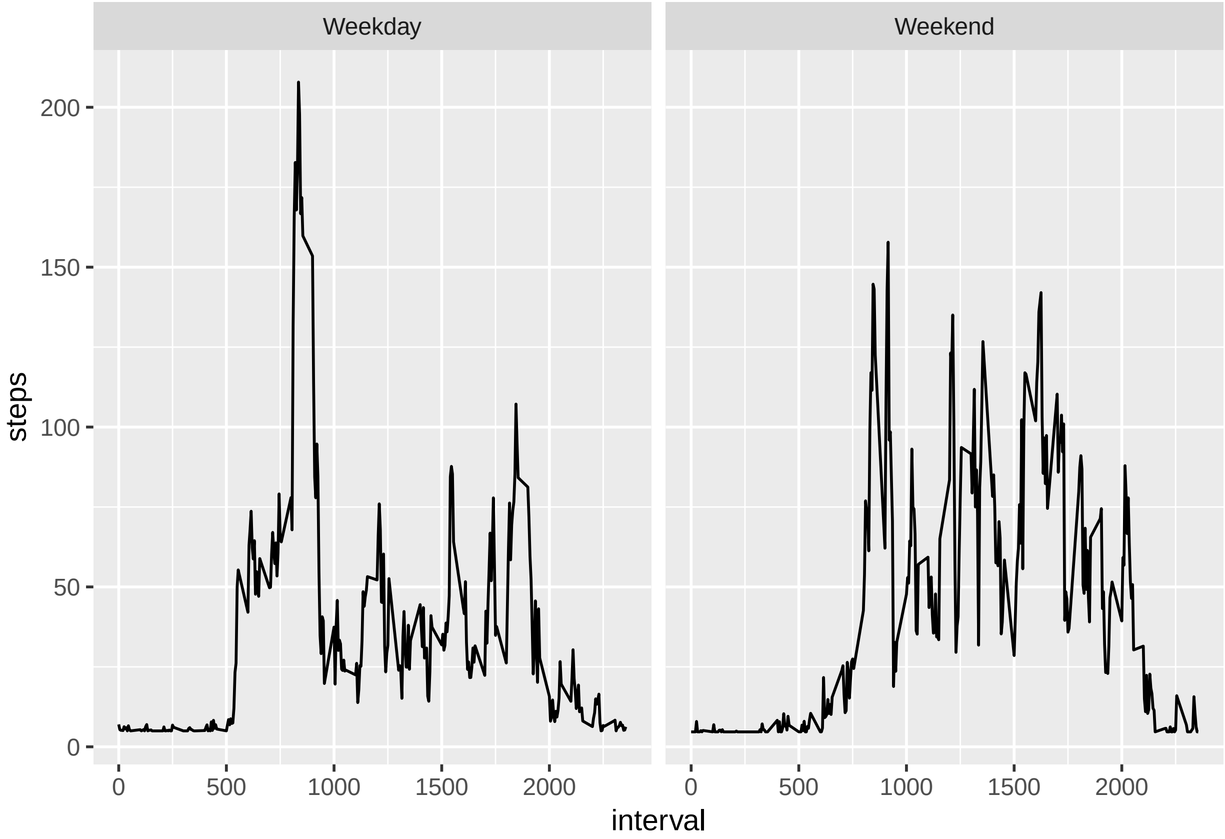
```
act2$date <- as.Date(act2$date, "%Y-%m-%d")

day <- weekdays(act2$date)
daylevel <- vector()
for (i in 1:nrow(act2)) {
    if (day[i] == "Saturday") {
        daylevel[i] <- "Weekend"
    } else if (day[i] == "Sunday") {
        daylevel[i] <- "Weekend"
    } else {
        daylevel[i] <- "Weekday"
    }
}
act2$daylevel <- daylevel
act2$daylevel <- factor(act2$daylevel)

steps_per_day <- aggregate(steps ~ interval +daylevel, data=act2, mean)
names(steps_per_day) <- c("interval", "daylevel", "steps")
```

Make a panel plot containing a time series plot (i.e. `type = "l"`type=“l”) of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all weekday days or weekend days (y-axis). See the README file in the GitHub repository to see an example of what this plot should look like using simulated data.

```
g<- ggplot(steps_per_day, aes(x=interval, y=steps))
g+ geom_line() +facet_wrap(.~ daylevel)
```


Weekday
Weekend
steps
interval