

Reproducible Research - Project 1

```
rm(list=ls())  
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2      v purrr   0.3.4  
## v tibble  3.0.4      v dplyr  1.0.2  
## v tidyr   1.1.2      v stringr 1.4.0  
## v readr   1.4.0      v forcats 0.5.0
```

```
## Warning: package 'tibble' was built under R version 4.0.3
```

```
## Warning: package 'readr' was built under R version 4.0.3
```

```
## -- Conflicts ----- tidyverse_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag()     masks stats::lag()
```

```
library(lubridate)
```

```
## Warning: package 'lubridate' was built under R version 4.0.3
```

```
##  
## Attaching package: 'lubridate'
```

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

```
activity <- read_csv('activity.csv')
```

```
##  
## -- Column specification -----  
## cols(  
##   steps = col_double(),  
##   date = col_date(format = ""),  
##   interval = col_double()  
## )
```

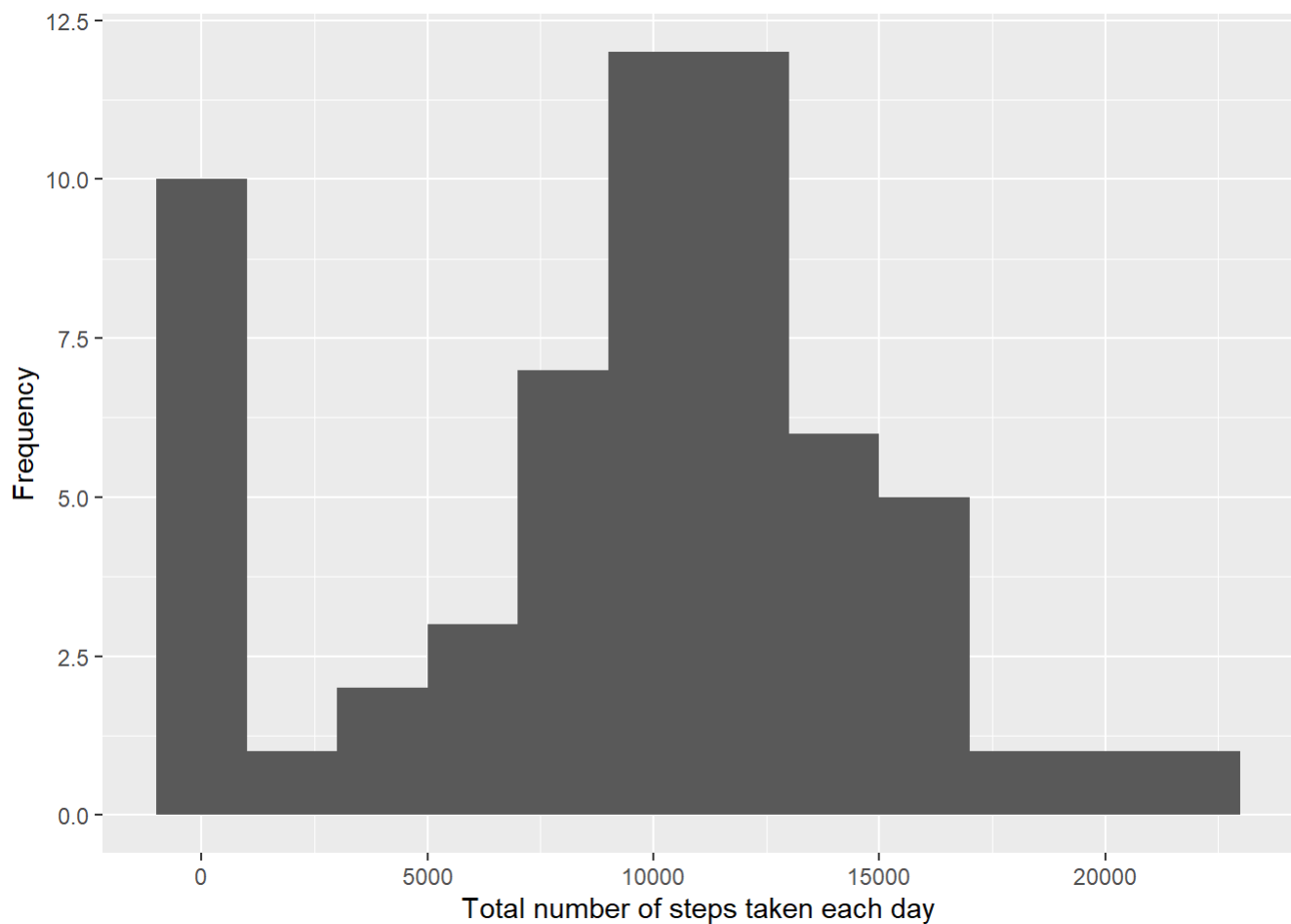
```
activity$interval_num <- as.integer(activity$interval)
```

Total Number of Steps Per Day

```
total_steps <- activity %>%  
  group_by(date) %>%  
  summarize(daily_steps = sum(steps, na.rm=TRUE))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ggplot(data=total_steps, mapping=aes(daily_steps)) +  
  geom_histogram(binwidth=2000)+  
  xlab("Total number of steps taken each day") +  
  ylab("Frequency")
```



Mean and median total number of steps taken per day

```
mean_daily_total <- mean(total_steps$daily_steps)  
median_daily_total <- median(total_steps$daily_steps)  
print(paste("Mean Daily Steps:", as.character(mean_daily_total)))
```

```
## [1] "Mean Daily Steps: 9354.22950819672"
```

```
print(paste("Median Daily Steps:", as.character(median_daily_total)))
```

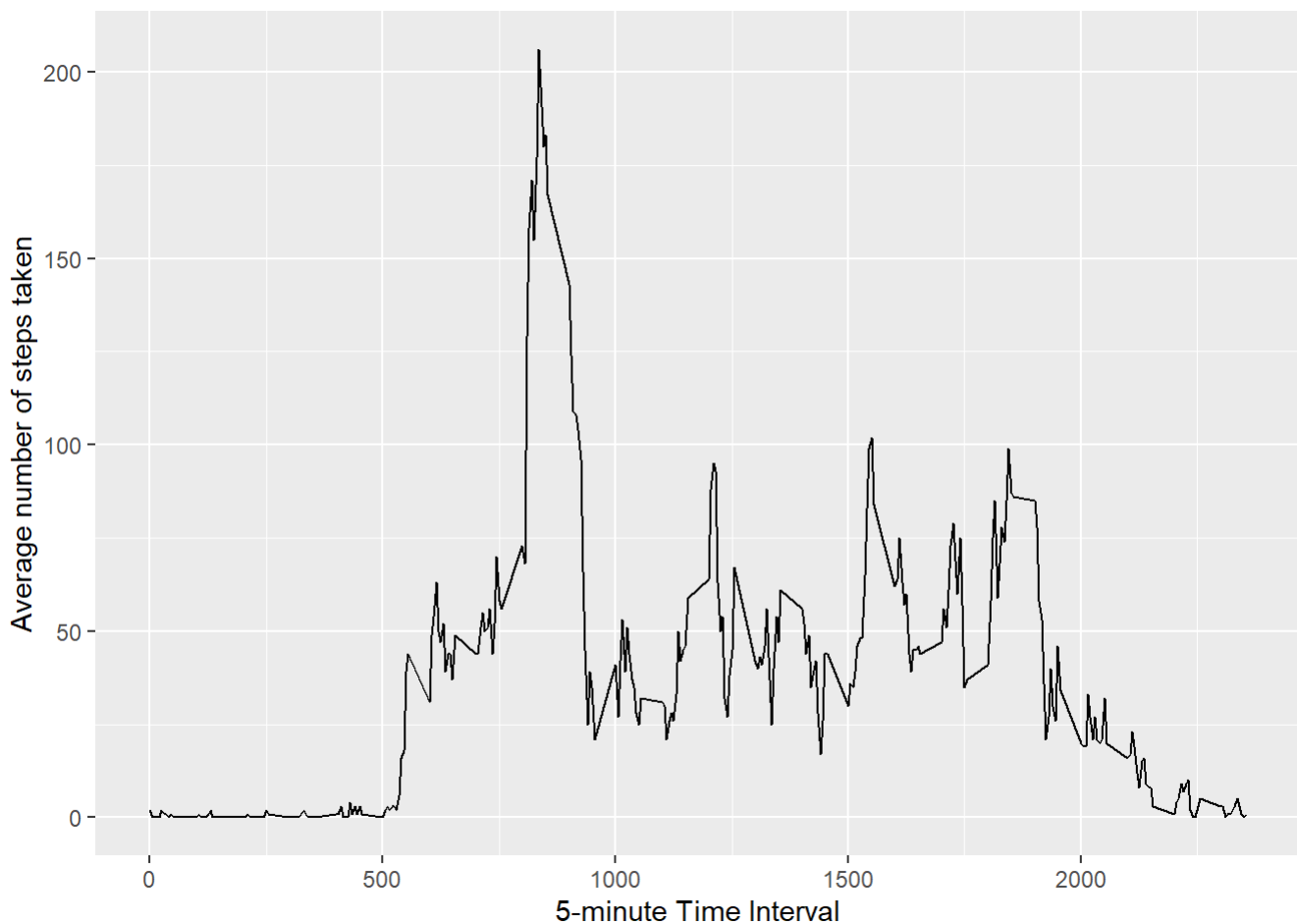
```
## [1] "Median Daily Steps: 10395"
```

Daily Activity Pattern - Average Steps per Interval

```
daily_activity_pattern <- activity %>%  
  group_by(interval) %>%  
  summarize(interval_steps = round(mean(steps, na.rm=TRUE)),0)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ggplot(data=daily_activity_pattern, mapping=aes(x=interval, y=interval_steps)) +  
  geom_line()+  
  xlab("5-minute Time Interval") +  
  ylab("Average number of steps taken")
```



```
max_interval_steps <- daily_activity_pattern %>%  
  summarize(m = max(interval_steps, na.rm=TRUE))  
  
print(paste0("The 5 minute interval with the most steps is: ", max_interval_steps[[1]]))
```

```
## [1] "The 5 minute interval with the most steps is: 206"
```

Imputation of missing values

The mean of each time interval is used to fill in missing values.

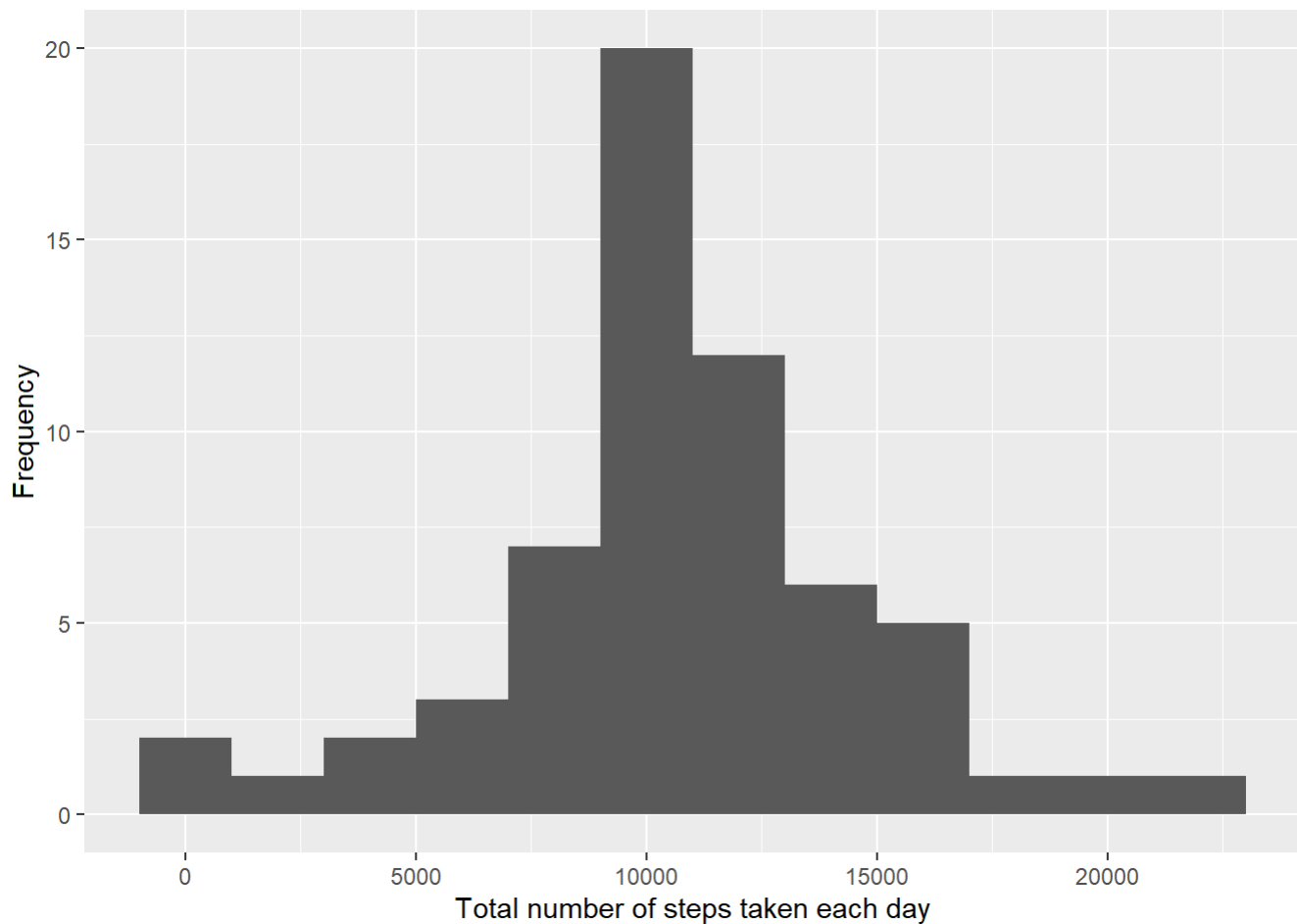
```
activity2 <- activity %>%  
  mutate(steps =  
    case_when(  
      is.na(steps) ~ daily_activity_pattern$interval_steps[match(activity$interval,  
        daily_activity_pattern$interval)],  
      TRUE ~ as.numeric(steps)))
```

Steps Taken Per Day - with Missing Values Imputed with Mean

```
imputed_total_steps <- activity2 %>%  
  group_by(date) %>%  
  summarize(daily_steps = sum(steps, na.rm=TRUE))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ggplot(data=imputed_total_steps, mapping=aes(daily_steps)) +  
  geom_histogram(binwidth=2000)+  
  xlab("Total number of steps taken each day") +  
  ylab("Frequency")
```



When we compare the mean and median total daily steps with the data set which has imputed values for missing data, the mean increases by 1411 steps and the median increases by 367.

```
mean_daily_total_imputed <- mean(imputed_total_steps$daily_steps)
median_daily_total_imputed <- median(imputed_total_steps$daily_steps)

print(paste("Mean Daily Steps:", as.character(mean_daily_total)))
```

```
## [1] "Mean Daily Steps: 9354.22950819672"
```

```
print(paste("Median Daily Steps:", as.character(median_daily_total)))
```

```
## [1] "Median Daily Steps: 10395"
```

```
print(paste("Mean Daily Steps with Missing Values Imputed:", as.character(mean_daily_total_imputed)))
```

```
## [1] "Mean Daily Steps with Missing Values Imputed: 10765.6393442623"
```

```
print(paste("Median Daily Steps with Missing Values Imputed:", as.character(median_daily_total_imputed)))
```

```
## [1] "Median Daily Steps with Missing Values Imputed: 10762"
```

```
print(paste("Difference in Mean Daily Steps with Missing Values Imputed:", as.character(round(mean_daily_total_imputed - mean_daily_total, 0))))
```

```
## [1] "Difference in Mean Daily Steps with Missing Values Imputed: 1411"
```

```
print(paste("Difference in Median Daily Steps with Missing Values Imputed:", as.character(median_daily_total_imputed - median_daily_total, 0)))
```

```
## [1] "Difference in Median Daily Steps with Missing Values Imputed: 367"
```

Comparison of Weekend Vs. Weekday Activity Patterns

```
activity2 <- activity2 %>%
  mutate(dayofweek = wday(date, label=TRUE)) %>%
  mutate(weekday =
    if_else(
      substr(dayofweek, 1,1) == 'S', "weekend",
      "weekday"))

activity2$weekday <- factor(activity2$weekday)

weekend_intervals <- activity2 %>%
  group_by(interval, weekday) %>%
  summarize(interval_steps = round(mean(steps, na.rm=TRUE)),0)
```

```
## `summarise()` regrouping output by 'interval' (override with `.groups` argument)
```

```
ggplot(data=weekend_intervals, mapping=aes(x=interval, y=interval_steps)) +
  geom_line()+
  facet_grid(rows = vars(weekday)) +
  xlab("5-minute Time Interval") +
  ylab("Average number of steps taken")
```

