courseraRR

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

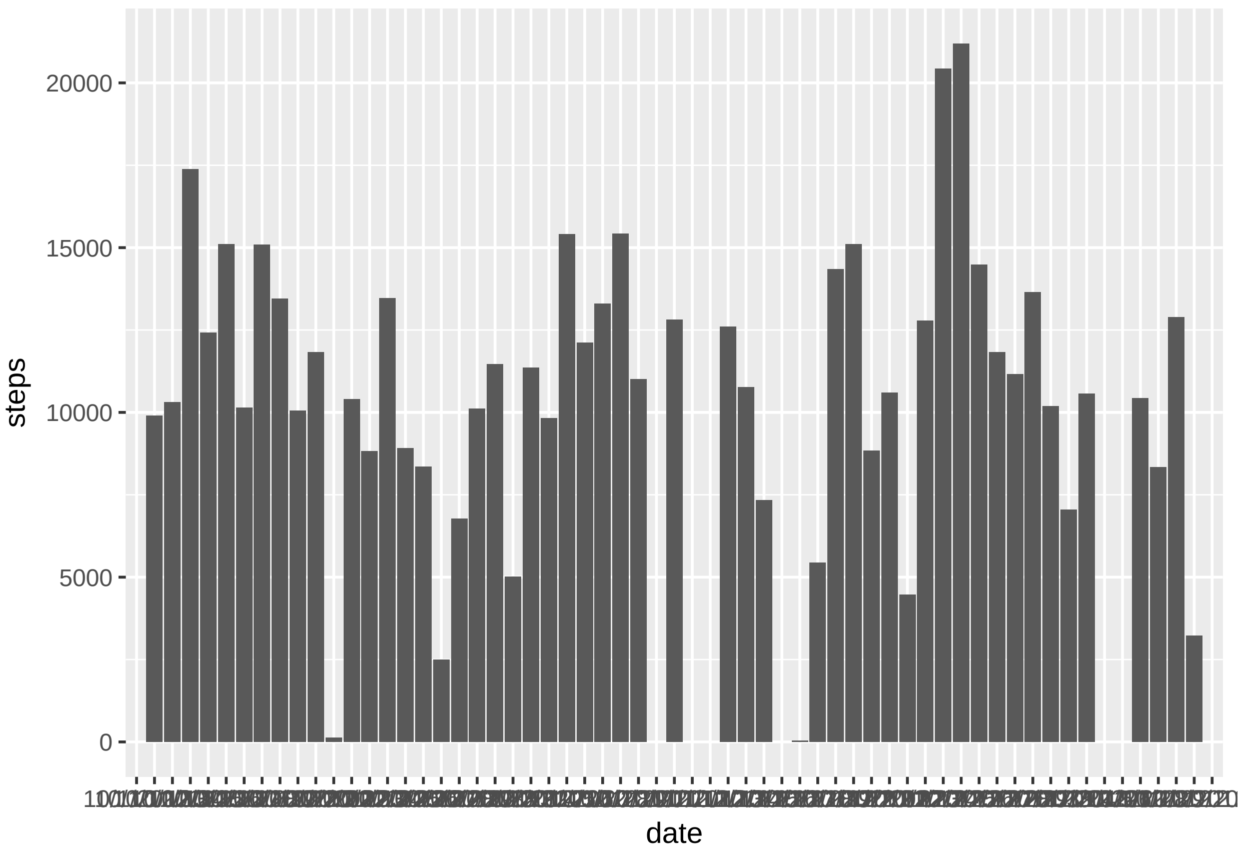
```
install.packages('ggplot2')
```

```
## Installing package into 'C:/Users/jmclaughlin/Documents/R/win-library/3.3'
## (as 'lib' is unspecified)
```

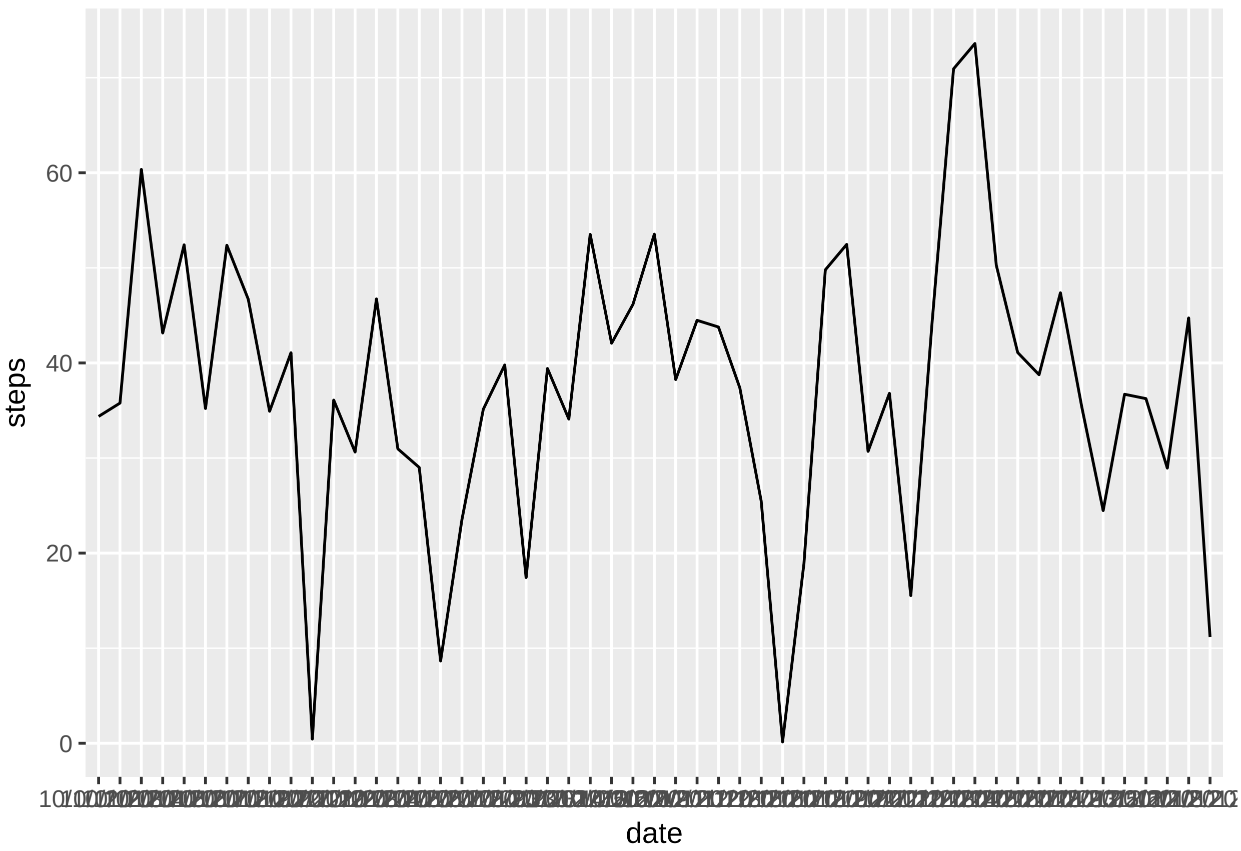
```
## package 'ggplot2' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\jmclaughlin\AppData\Local\Temp\2\RtmpWGq2Pm\downloaded_packages
```

```
library('ggplot2')

activity <- read.csv("activity.csv", sep = ",", header = TRUE)

ggplot(data = activity, aes(date, steps)) + stat_summary(fun.y = sum, geom = "bar")
```

```
## Warning: Removed 2304 rows containing non-finite values (stat_summary).
```



```
meandays <- aggregate(steps ~ date, activity,mean)
mediandays <- aggregate(steps ~ date, activity,median)

ggplot(data = meandays, aes(date, steps, group = 1)) + stat_summary(fun.y = sum, geom = "line")
```



```
avg_interval <- aggregate(steps ~ interval, activity, mean)
subset(avg_interval, steps==max(avg_interval$steps))
```

```
##     interval    steps
## 104      835 206.1698
```

```
##input missing data

sum(is.na(activity))
```

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

```
activity2<- activity
nas<- is.na(activity2$steps)
avg_interval<- tapply(activity2$steps, activity2$interval, mean, na.rm=TRUE, simplify = TRUE)
activity2$steps[nas] <- avg_interval[as.character(activity2$interval[nas])]
names(activity2)
```

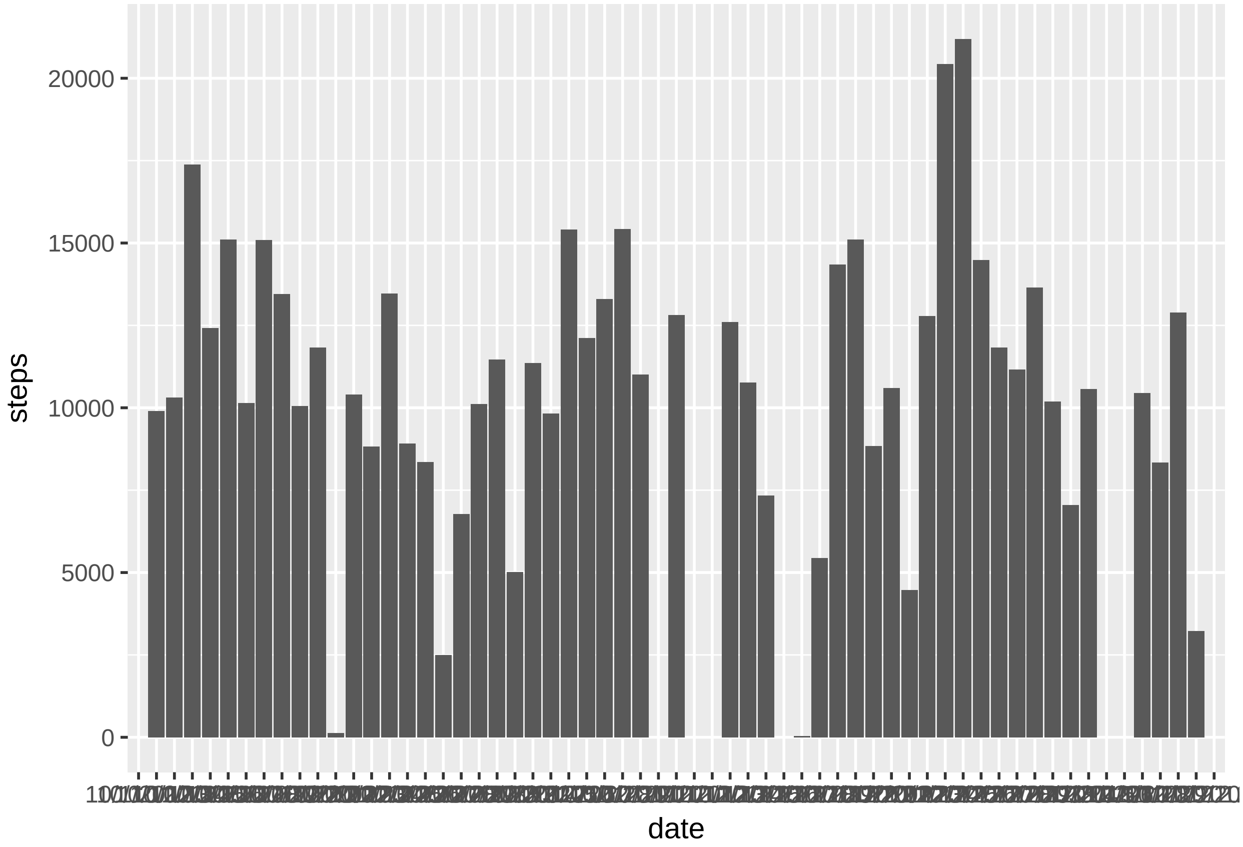
```
## [1] "steps"    "date"     "interval"
```

```
sum(is.na(activity2))
```

```
## [1] 0
```

```
ggplot(data = activity, aes(date, steps)) + stat_summary(fun.y = sum, geom = "bar")
```

```
## Warning: Removed 2304 rows containing non-finite values (stat_summary).
```

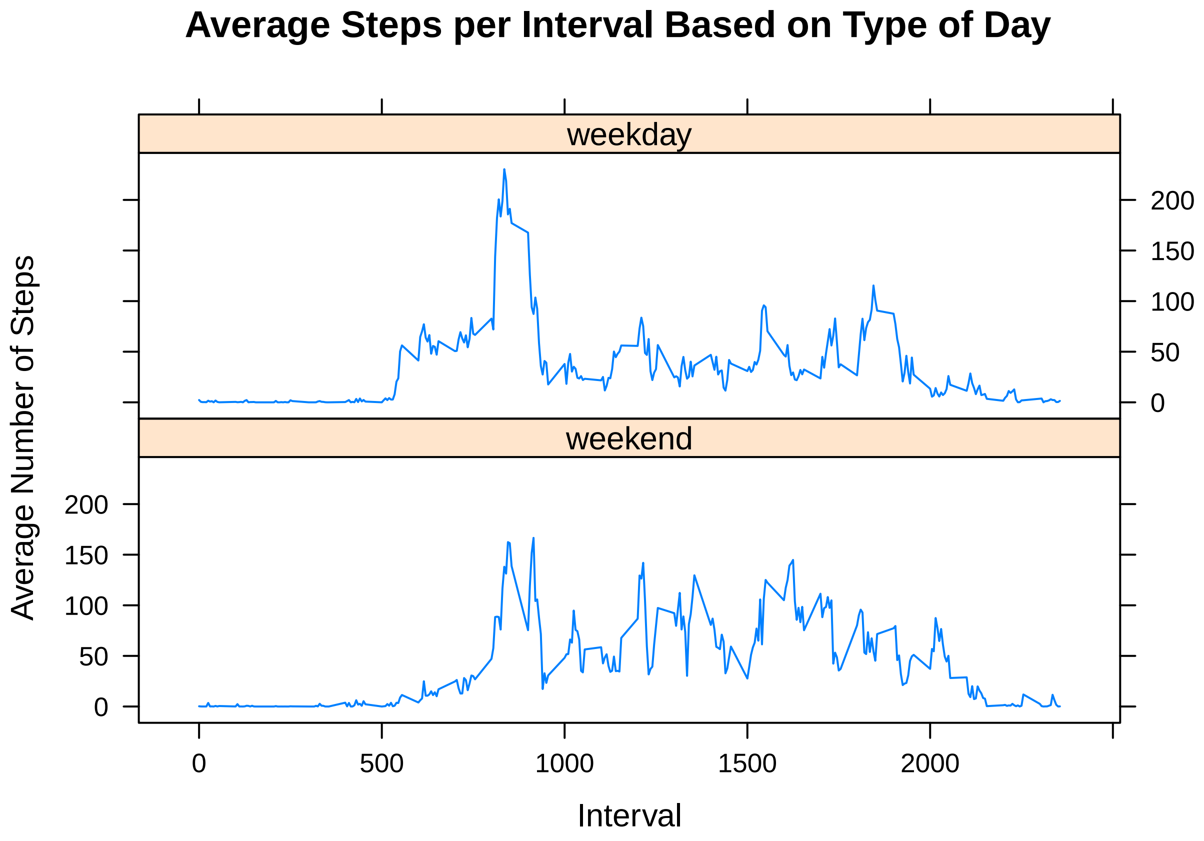


```
activity2$date <- as.Date(activity2$date, format = "%m/%d/%Y")

weekdays1 <- c('Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday')

activity2$wDay <- factor((weekdays(activity2$date) %in% weekdays1),
                    levels=c(FALSE, TRUE), labels=c('weekend', 'weekday'))

head(activity2)
```

```
##       steps       date interval    wDay
## 1 1.7169811 2012-10-01        0 weekday
## 2 0.3396226 2012-10-01        5 weekday
## 3 0.1320755 2012-10-01       10 weekday
## 4 0.1509434 2012-10-01       15 weekday
## 5 0.0754717 2012-10-01       20 weekday
## 6 2.0943396 2012-10-01       25 weekday
```

```
library(lattice)
```

```
avg <- aggregate(steps ~ interval+wDay,activity2, mean)

xyplot(steps~interval|wDay, data=avg, type="l",  layout = c(1,2),
       main="Average Steps per Interval Based on Type of Day",
       ylab="Average Number of Steps", xlab="Interval")
```



Including Plots

You can also embed plots, for example:

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