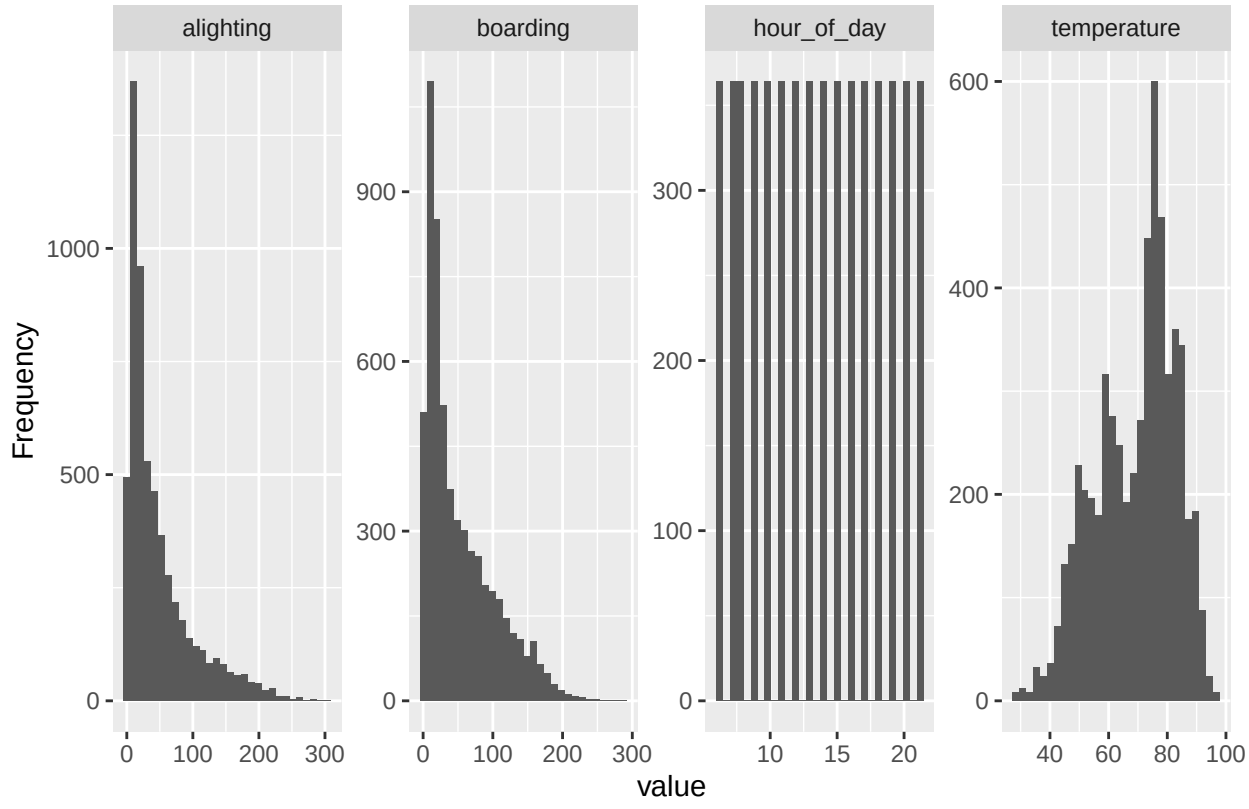
```
library(gridExtra)
library(ggplot2)
# library(dplyr)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
## v purrr   0.3.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::combine() masks gridExtra::combine()
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

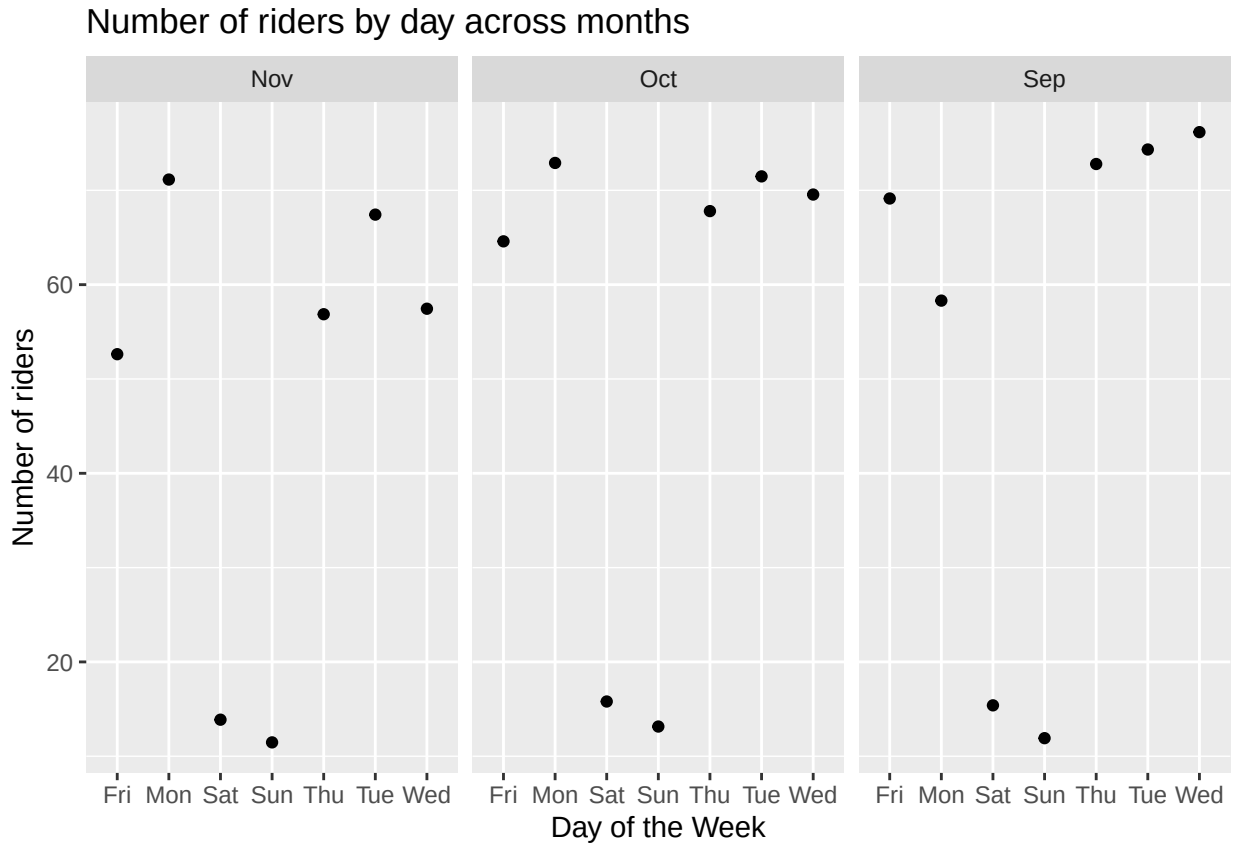
```
library(DataExplorer)

#detach(package:plyr,unload=TRUE)

metro <- read.csv("capmetro_UT.csv")

# Looking at some distributions

ncols <- dplyr::select_if(metro, is.numeric)
plot_histogram(ncols)
```



```r
# Diving into riders analysis

library(tidyverse)

r1 = metro %>%
               group_by(hour_of_day) %>%
                 summarize(riders = mean(boarding))

r2 = metro %>%
              group_by(hour_of_day) %>%
                 summarize(riders = mean(alighting))

plot1 = ggplot(r1) + geom_line(aes(x=hour_of_day, y=riders)) +
                                    ggtitle("Number of people boarding")

plot2 = ggplot(r2) + geom_line(aes(x=hour_of_day, y=riders)) +
                                    ggtitle("Number of people alighting")

# grid.arrange(r1, r2, nrow = 1)

r3 = metro %>%
             group_by(day_of_week,month) %>%
               summarize(riders = mean(boarding))
```

```
## `summarise()` has grouped output by 'day_of_week'. You can override using the
## `.groups` argument.
```

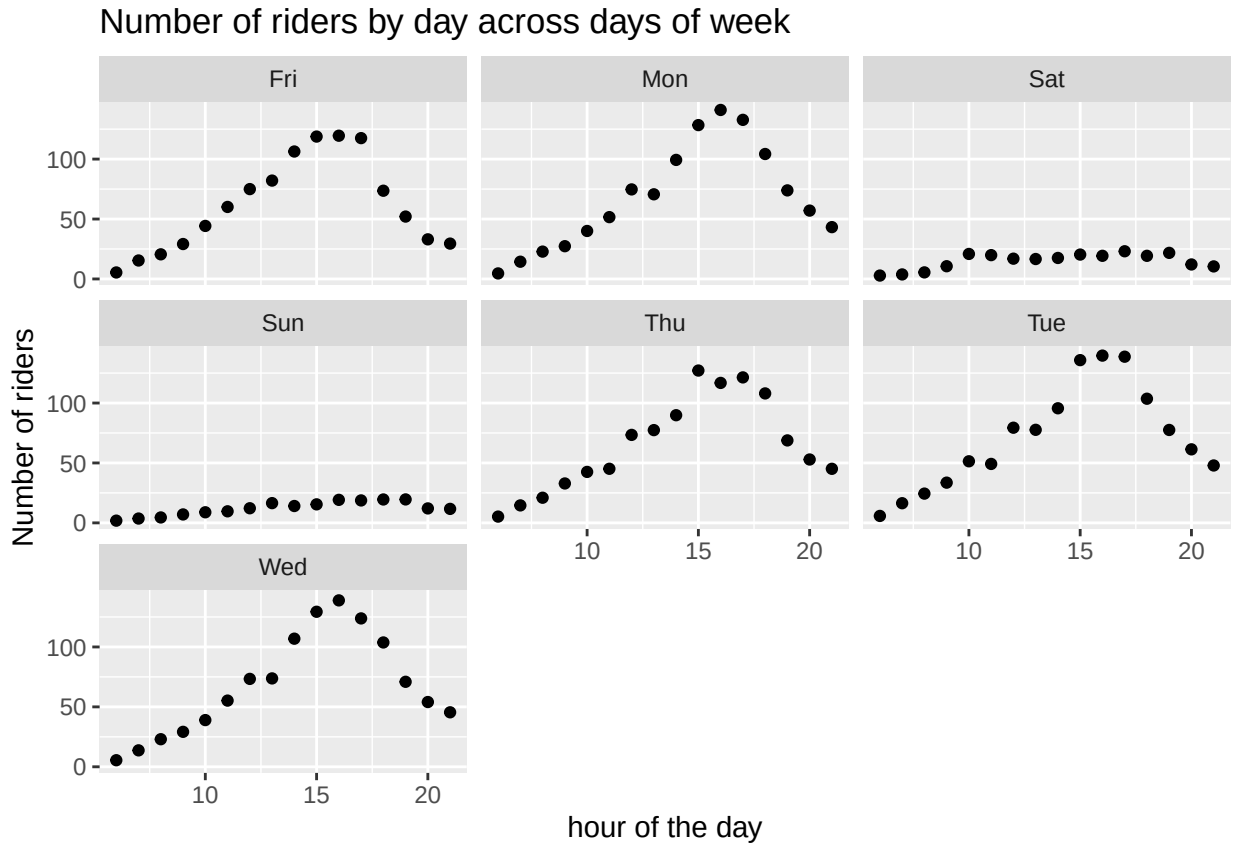
```r
ggplot(r3) + geom_point(aes(x=day_of_week, y=riders)) + facet_wrap(~month) +
             labs(x="Day of the Week", y="Number of riders",
                  title="Number of riders by day across months")
```




```
r4 = metro %>%
              group_by(hour_of_day, day_of_week) %>%
                summarize(riders = mean(boarding))
```

```
## `summarise()` has grouped output by 'hour_of_day'. You can override using the
## `.groups` argument.
```

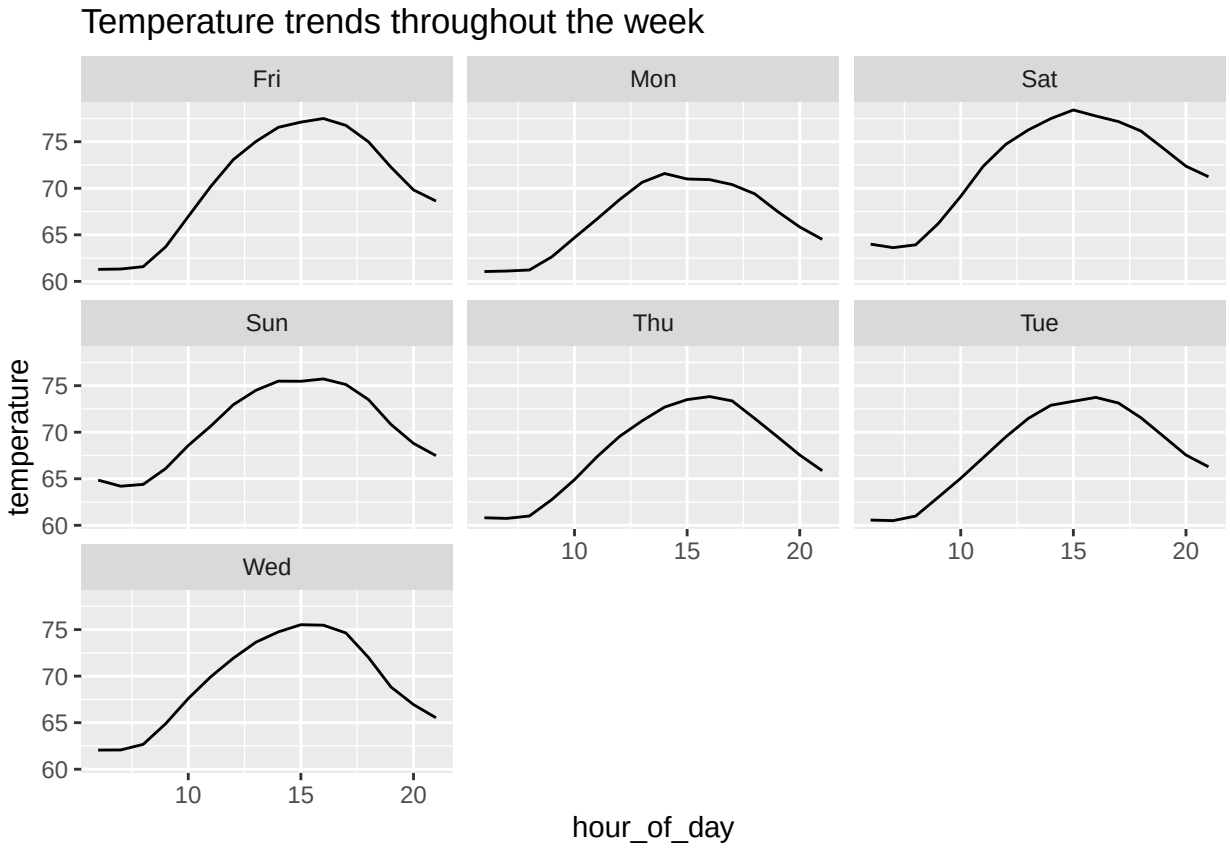
```
ggplot(r4) + geom_point(aes(x=hour_of_day, y=riders)) +
              facet_wrap(~day_of_week) + labs(x="hour of the day",
  y="Number of riders", title="Number of riders by day across days of week")
```




```
r5 = metro %>%
        group_by(day_of_week, hour_of_day) %>%
            summarize(temperature = mean(temperature))
```

```
## `summarise()` has grouped output by 'day_of_week'. You can override using the
## `.groups` argument.
```

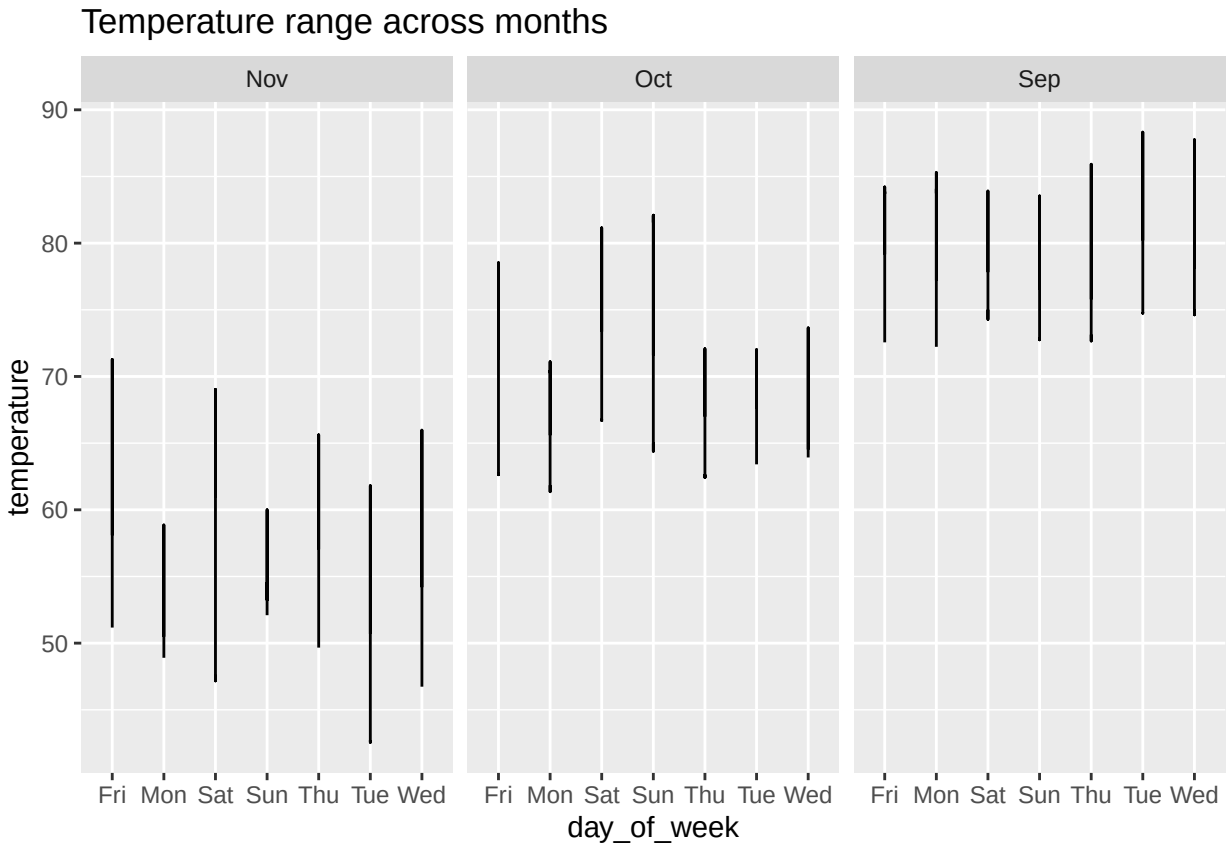
```
ggplot(r5) + geom_line(aes(x=hour_of_day, y=temperature)) +
    ggtitle("Temperature trends throughout the week") + facet_wrap(~day_of_week)
```




```
r6 = metro %>%
        group_by(day_of_week, hour_of_day,month) %>%
          summarize(temperature = mean(temperature))
```

```
## `summarise()` has grouped output by 'day_of_week', 'hour_of_day'. You can
## override using the `.groups` argument.
```

```
ggplot(r6) +
  geom_line(aes(x=day_of_week, y=temperature)) +
      ggtitle("Temperature range across months") +
          facet_wrap(~month)
```


Temperature range across months
Nov
Oct
Sep
90
80
70
60
50
temperature
Fri Mon Sat Sun Thu Tue Wed
day_of_week