ggplot2_plotly

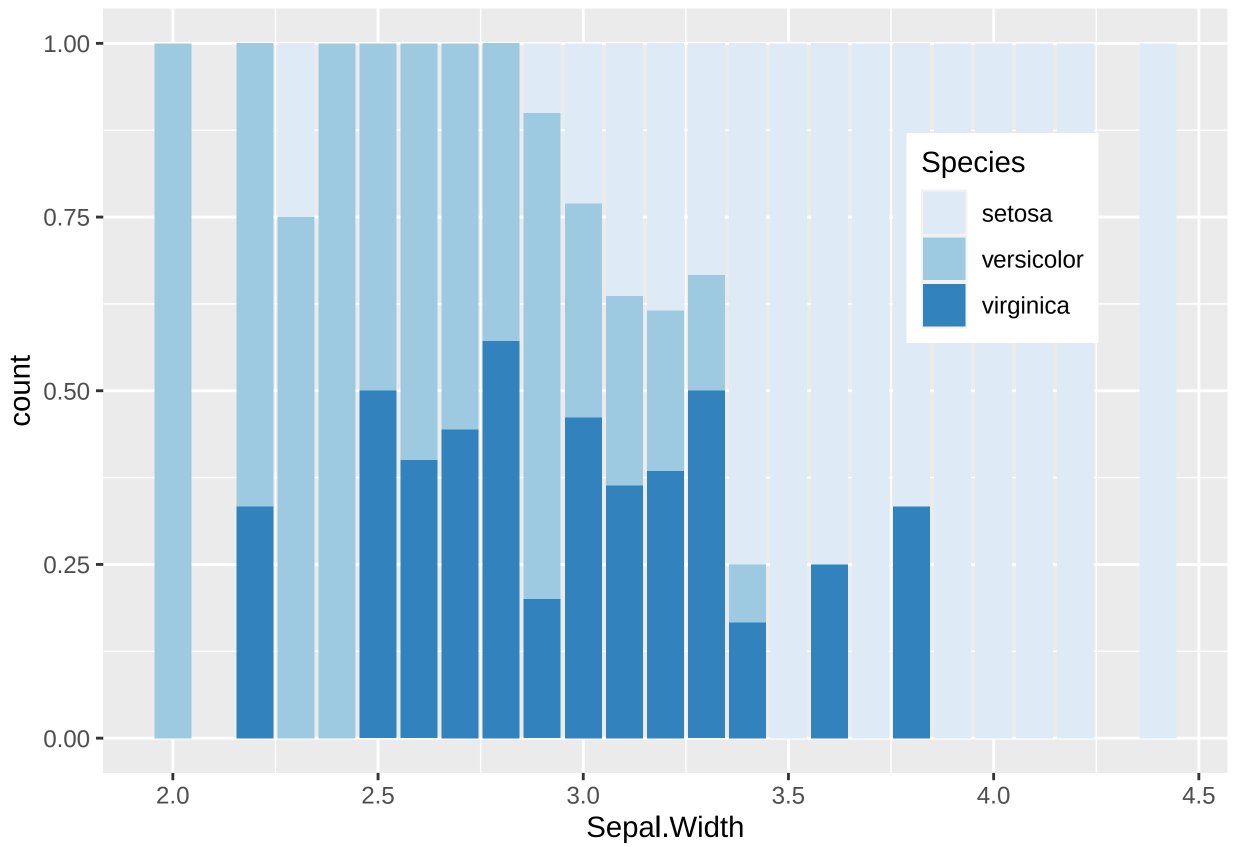
Katarzyna Tokarczuk

2024-12-28

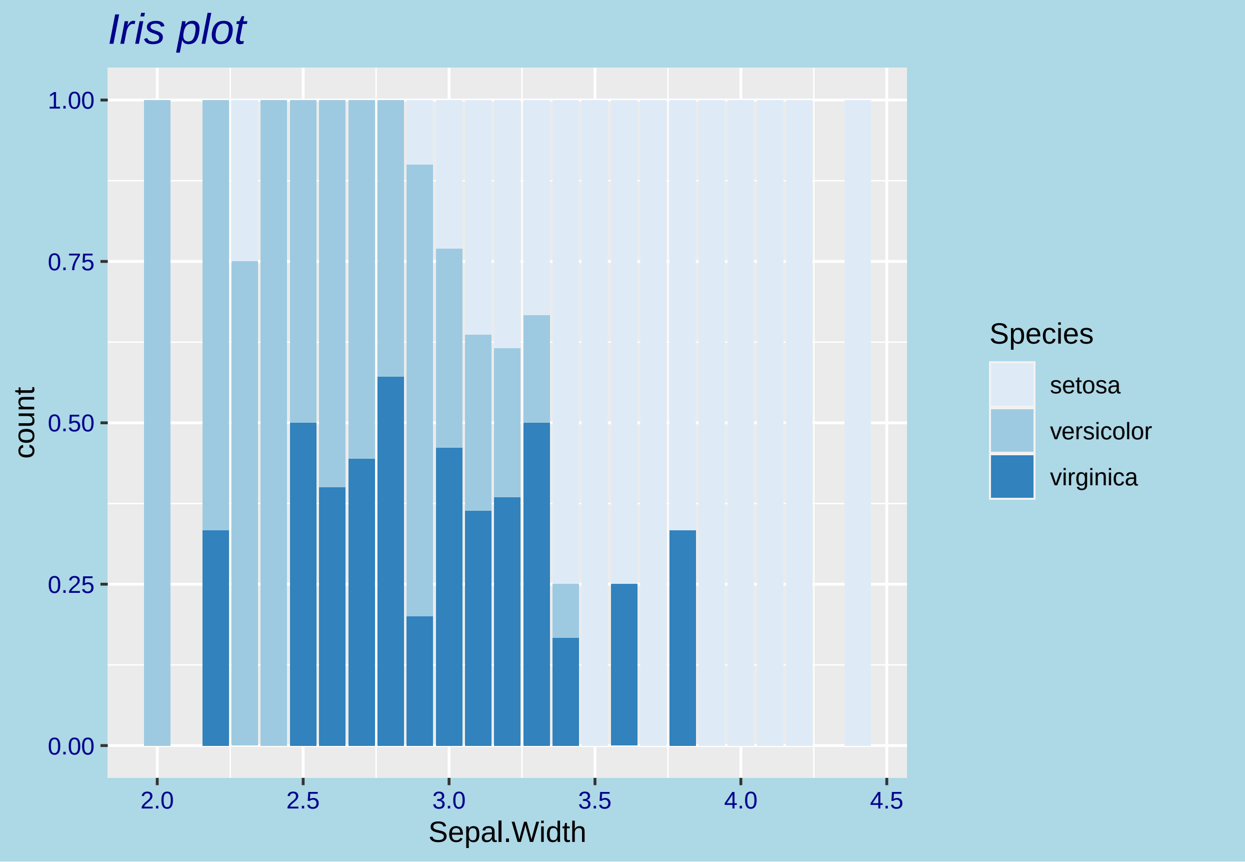
```
library('ggplot2')
library('tidyverse')
library('ggthemes')
library('plotly')
```

Example 1

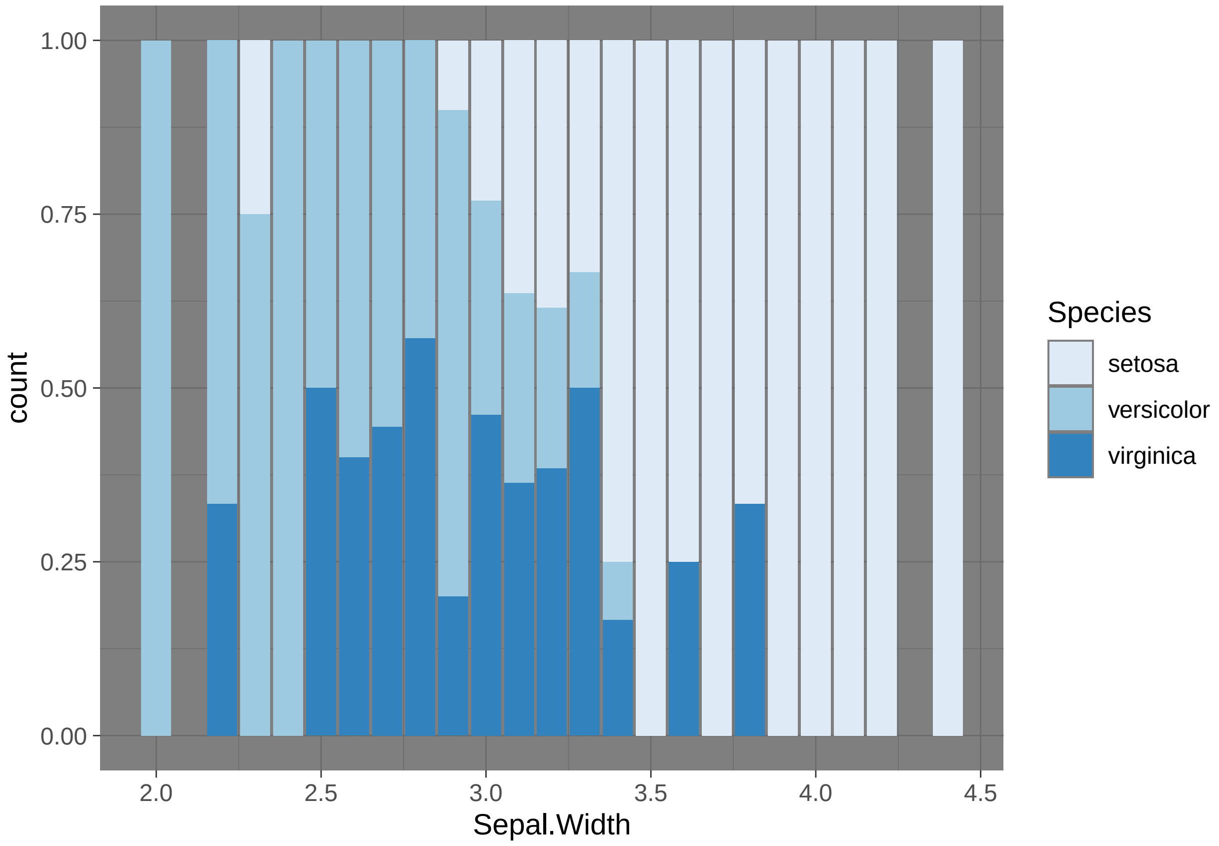
```
#a)
ir_pl<-ggplot(iris, aes(Sepal.Width, fill = Species)) +
  geom_bar(position = "fill")+
  scale_fill_brewer()
ir_pl+
  theme(legend.position=c(0.8, 0.7))
```



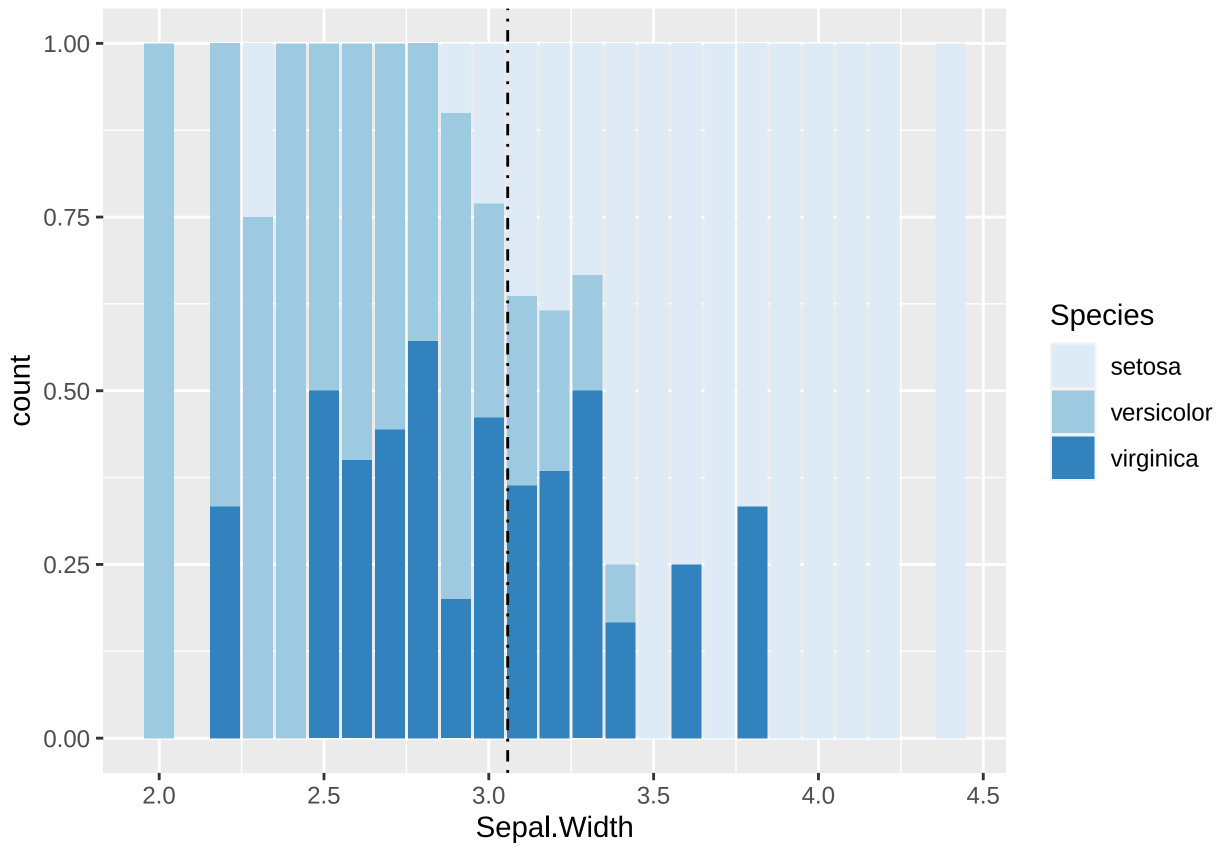
```r
#b)
ir_pl+
  ggtitle("Iris plot")+
  theme(
    rect = element_rect(fill = "light blue"),
    legend.key = element_rect(color = NA),
    axis.text=element_text(color="dark blue"),
    plot.title=element_text(size=16, face="italic", color="dark blue"),
    legend.margin=margin(10, 30, 20, 20, "pt")
  )
```



```
#c)
ir_pl+
  theme_dark()
```

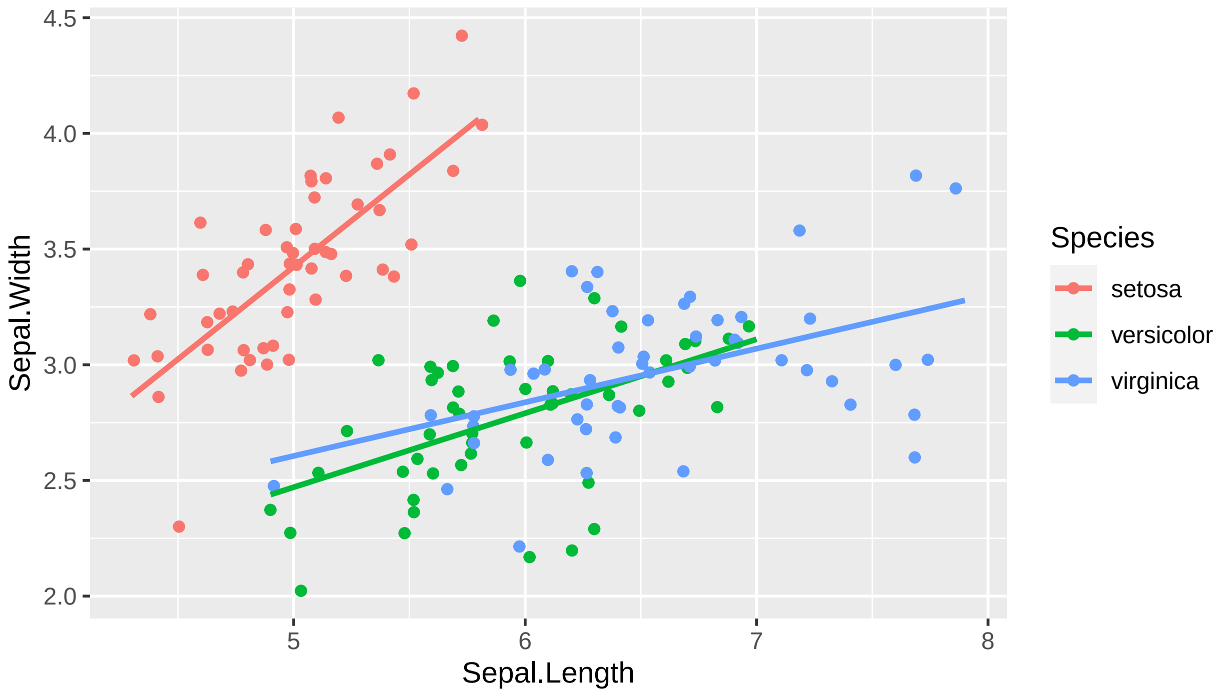


```
#d)
global_mean<-mean(iris$Sepal.Width)
ir_pl+
  geom_vline(xintercept=global_mean, color="black", linetype=4)
```



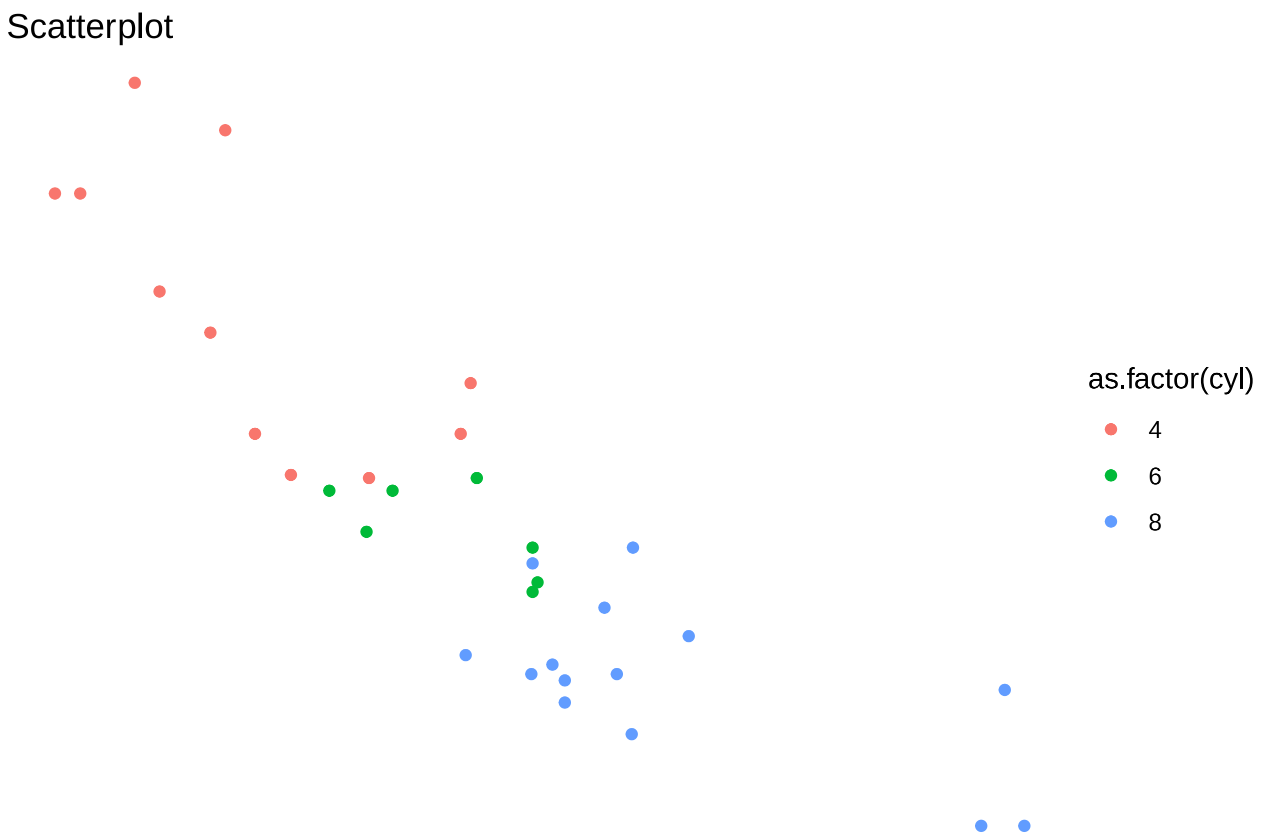
```
#e)
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, color = Species))+
  geom_jitter() +
  geom_smooth(method = "lm", se = FALSE) +
  coord_fixed(ratio=1)
```

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



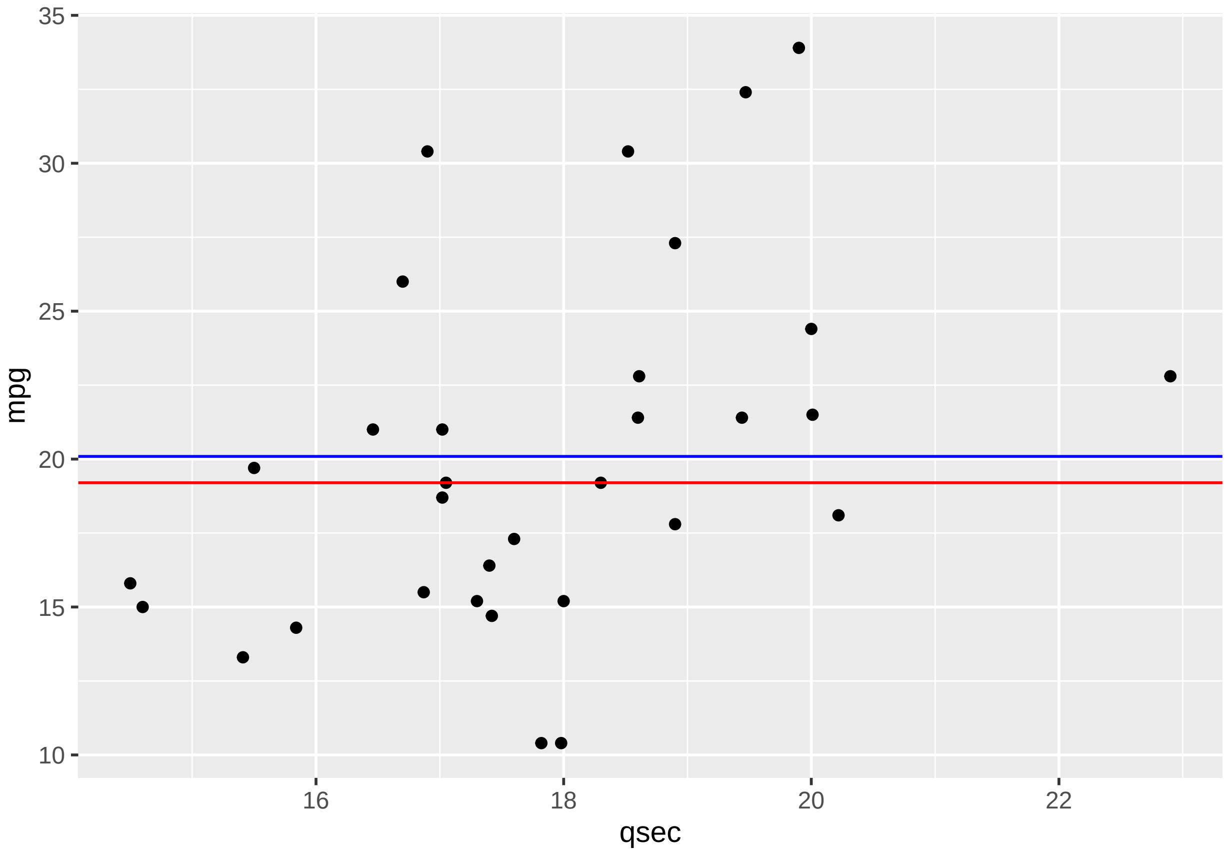
Task 1

```
#a,b)
ggplot(mtcars, aes(x= wt, y = mpg, color = as.factor(cyl))) + # converting a numeric variable to a fact
                                                              # to indicate that the values of the numb
  geom_point() +
  ggtitle("Scatterplot") +
  theme(
    plot.title=element_text(size=14, face="bold.italic", color="red"),
    axis.title.x=element_text(size=14, face="bold", color = "blue"),
    axis.title.y=element_text(size=14, face="bold", color = "#993333"),
    plot.background  = element_rect(fill = "light blue"),
    panel.background = element_rect(fill = "light yellow"),
    panel.border = element_rect(colour = "grey", fill=NA)
  )+
  theme_void()
```



```
#c)
data(mtcars)

ggplot(mtcars, aes(x = qsec, y = mpg)) +
  geom_point() +
  geom_hline(yintercept = median(mtcars$mpg), color = "red", linetype=1) +
  geom_hline(yintercept = mean(mtcars$mpg), color = "blue", linetype = 1)
```

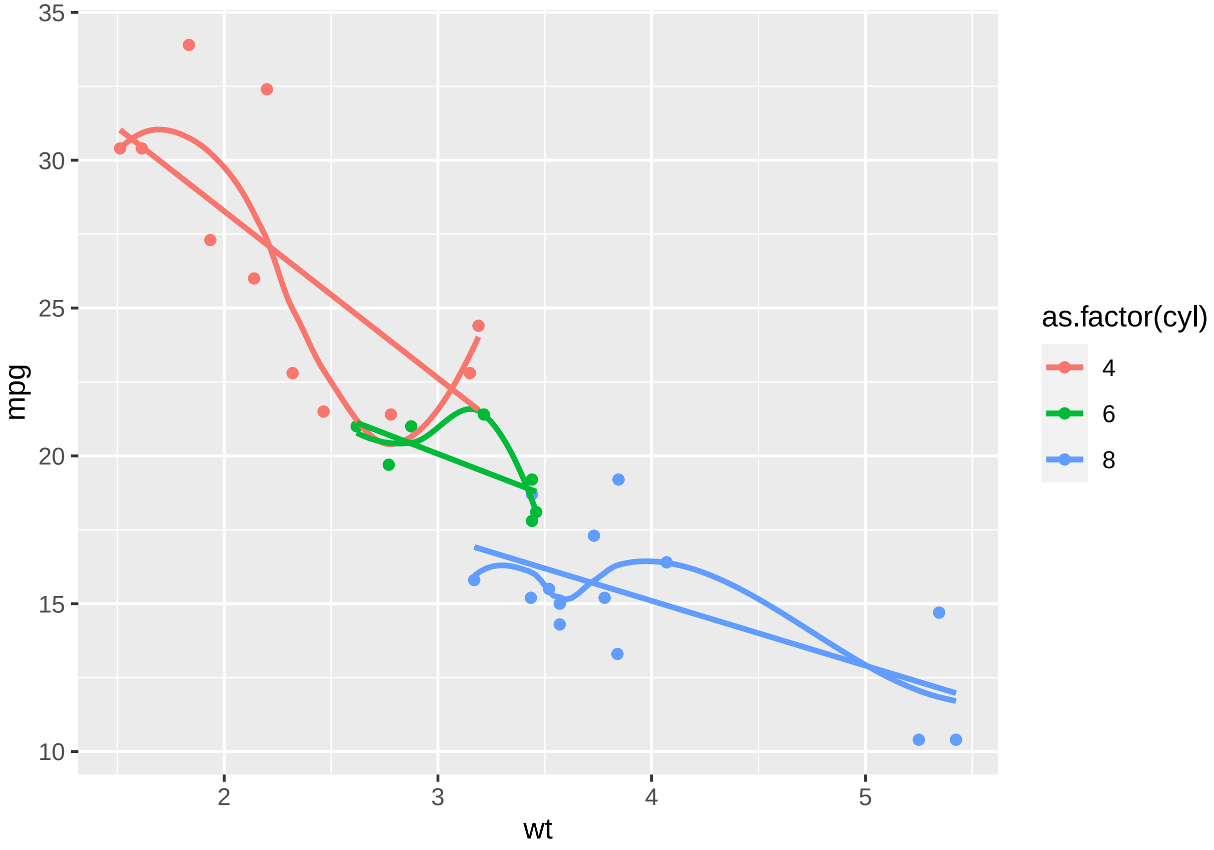


```
#d)
p <- ggplot(mtcars, aes(wt, mpg, group = cyl, color = as.factor(cyl))) +
  geom_point() +
  geom_smooth(method = "loess", se=FALSE) +geom_smooth(method = "lm", se=FALSE)
  labs(title = "Regression line for each group",
       x = "Car Weight (1000 lbs)",
       y = "Miles per Gallon",
       color = "Cylinders")
```

```
## $x
## [1] "Car Weight (1000 lbs)"
##
## $y
## [1] "Miles per Gallon"
##
## $colour
## [1] "Cylinders"
##
## $title
## [1] "Regression line for each group"
##
## attr(,"class")
## [1] "labels"
```

```
p
```

```
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
```



Example 2

```
#a)
plot_ly(data = iris, x = ~Sepal.Length, y = ~Petal.Length)
```

```
## No trace type specified:
##   Based on info supplied, a 'scatter' trace seems appropriate.
##   Read more about this trace type -> https://plotly.com/r/reference/#scatter
```

```
## No scatter mode specifed:
##   Setting the mode to markers
##   Read more about this attribute -> https://plotly.com/r/reference/#scatter-mode
```

```
plot_ly(data = iris, x = ~Sepal.Length, y = ~Petal.Length)%>%
  add_markers()
```

```
#b)
iris%>%
  filter(Species =="setosa")%>%
plot_ly(x = ~Sepal.Width)%>%
  add_histogram(nbinsx=6, color = I("darkgreen"), opacity = 0.5)
```

```
#c)
plot_ly(data = iris, x = ~Sepal.Length, y = ~Petal.Length)%>%
  add_markers(marker = list(size = 12,
                            color = 'rgba(255, 182, 193, .9)',
                            line = list(color = "rgba(100, 20, 20, .5)",
                                        width = 5)))%>%
  layout(title = "Scatterplot")
```

Task 2

```
#a)
str(mtcars)
```

```
## ’data.frame’:    32 obs. of  11 variables:
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
##  $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
##  $ qsec: num  16.5 17 18.6 19.4 17 ...
##  $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
##  $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
##  $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
##  $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

```
mtcars%>%
  filter(cyl==4)%>%
  plot_ly(x=~disp, y=~mpg, color=~am) %>%
  add_markers(colors="Set1") %>%
layout(title = "New colors")
```

```
#lub
str(mtcars)
```

```
## ’data.frame’:    32 obs. of  11 variables:
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
##  $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
##  $ qsec: num  16.5 17 18.6 19.4 17 ...
##  $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
##  $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
##  $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
##  $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

```
mtcars%>%
  filter(cyl==4)%>%
  plot_ly(x=~disp, y=~mpg, color=~as.factor(am),colors="Set1") %>%
  add_markers() %>%
  layout(title = "New colors")
```

```
#b)
plot_ly(data = mtcars, x=~disp, y=~mpg )%>%
  add_histogram2d(nbinsx=3, nbinsy=3)
```

```
#c)
plot_ly(data = mtcars, y=~mpg, x=~cyl)%>%
  add_boxplot()%>%
  layout(title = "Boxplot")
```