Prática 1 - Regressão Linear Simples

A seguir dados de 32 automóveis e 11 variáveis da base mtcars do pacote ggplot2.

```
require(ggplot2)
data("mtcars")
dim(mtcars)
```

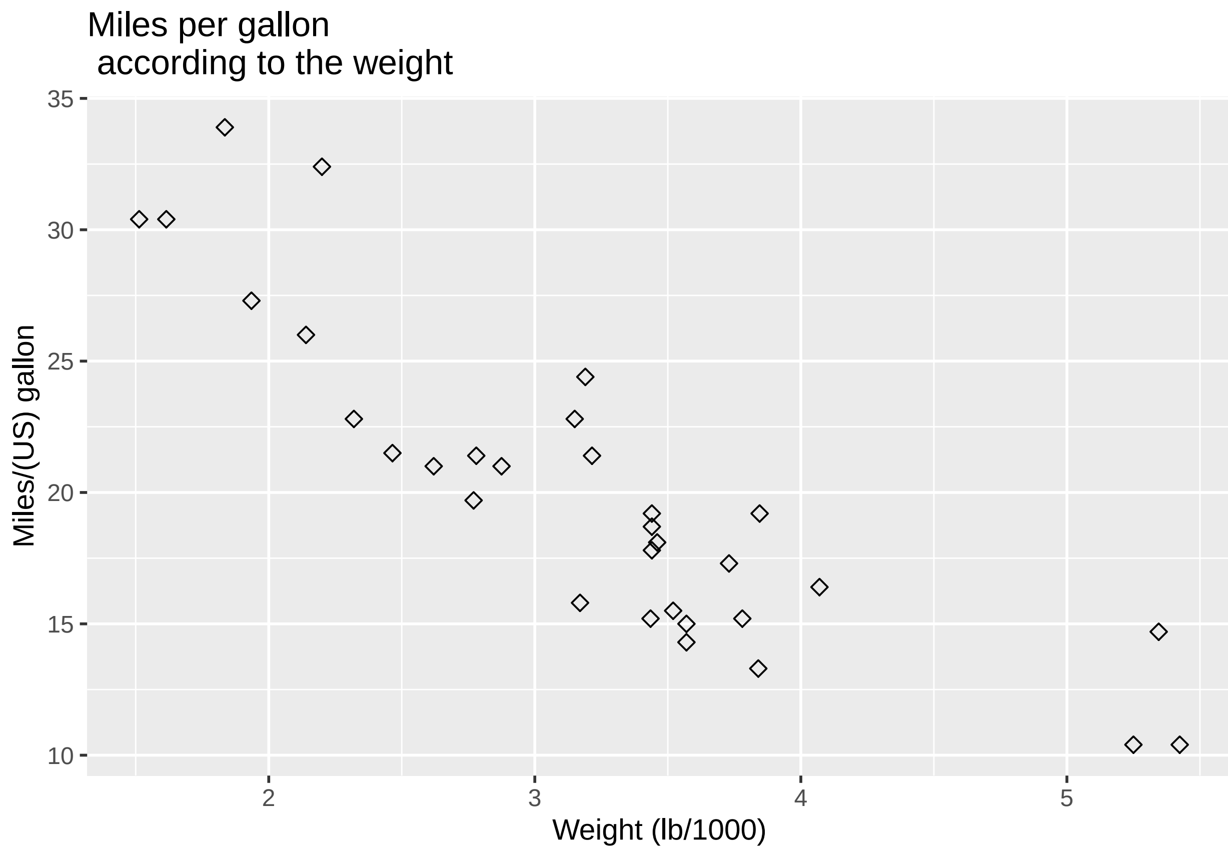
```
## [1] 32 11
```

```
head(mtcars)
```

```
##                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

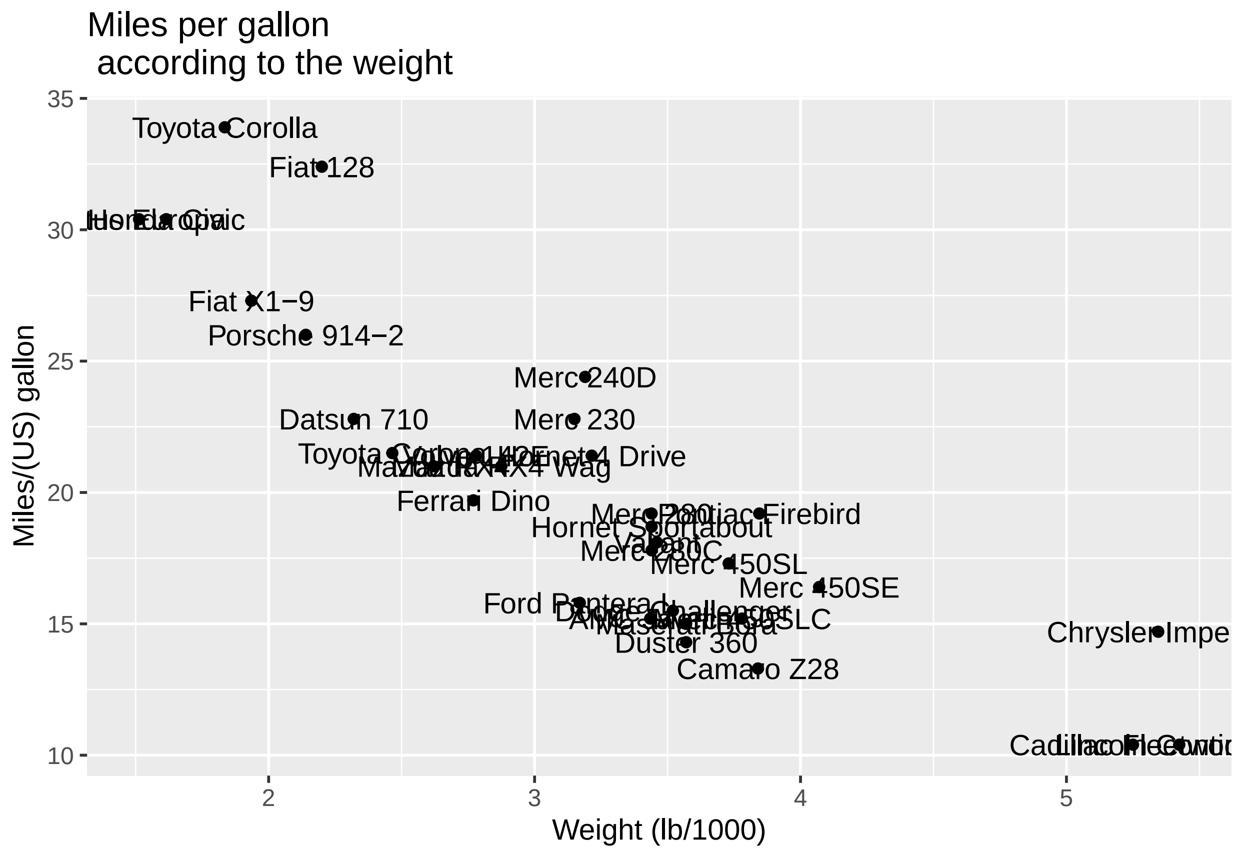
Na prática de hoje iremos utilizar regressão linear simples para analisar os dados da eficiência (milhas por galão) e o peso do carro.

```
mtcars$cyl <- as.factor(mtcars$cyl)

ggplot(mtcars, aes(x=wt, y=mpg)) +
  geom_point(size=2, shape=23) +
  labs(title="Miles per gallon \n according to the weight",
       x="Weight (lb/1000)", y = "Miles/(US) gallon")
```



Outra opção de visualização dos dados.

```
ggplot(mtcars, aes(x=wt, y=mpg)) +
  geom_point() +
  geom_text(label=rownames(mtcars))+
    labs(title="Miles per gallon \n according to the weight",
       x="Weight (lb/1000)", y = "Miles/(US) gallon")
```



Modelo de regressão linear simples.

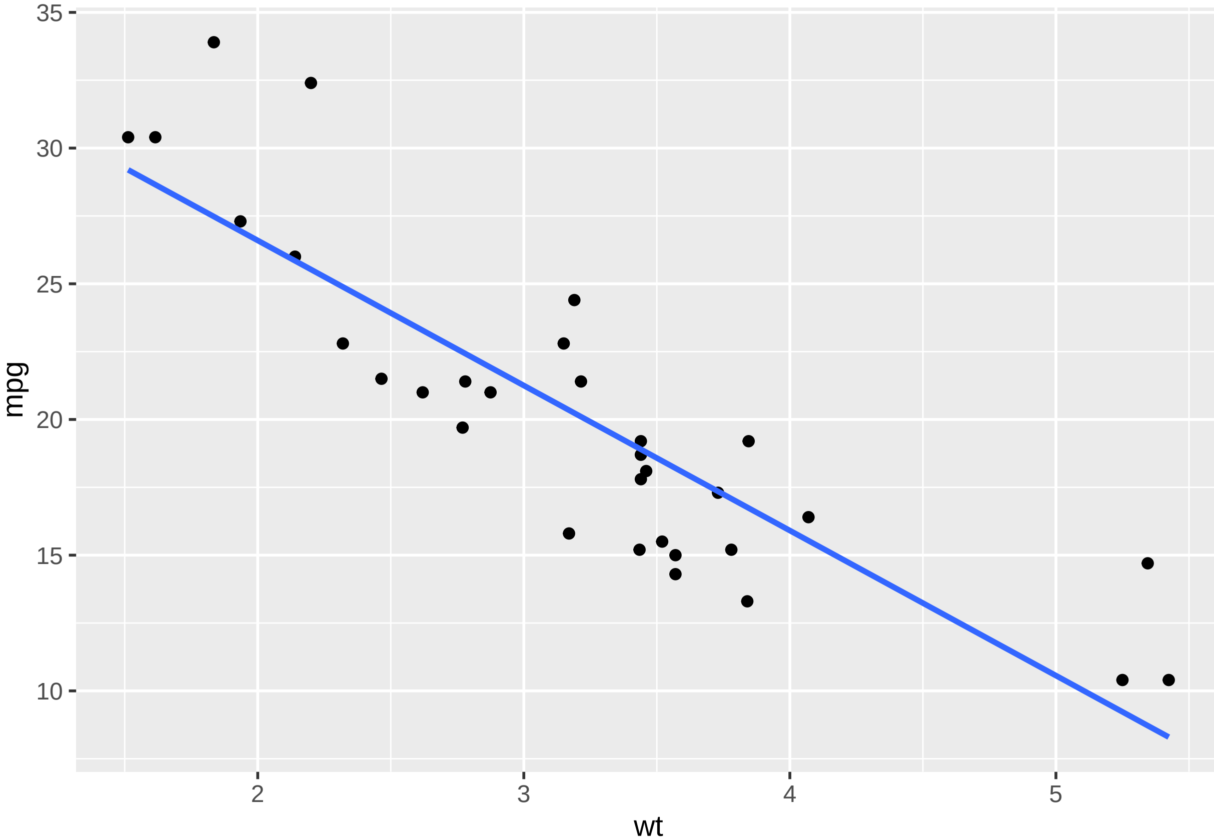
$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

```
ajuste.geral = lm(mpg~wt,data=mtcars)
summary(ajuste.geral)
```

```
##
## Call:
## lm(formula = mpg ~ wt, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5432 -2.3647 -0.1252  1.4096  6.8727
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  37.2851     1.8776  19.858  < 2e-16 ***
## wt           -5.3445     0.5591  -9.559 1.29e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.046 on 30 degrees of freedom
## Multiple R-squared:  0.7528, Adjusted R-squared:  0.7446
## F-statistic: 91.38 on 1 and 30 DF,  p-value: 1.294e-10
```

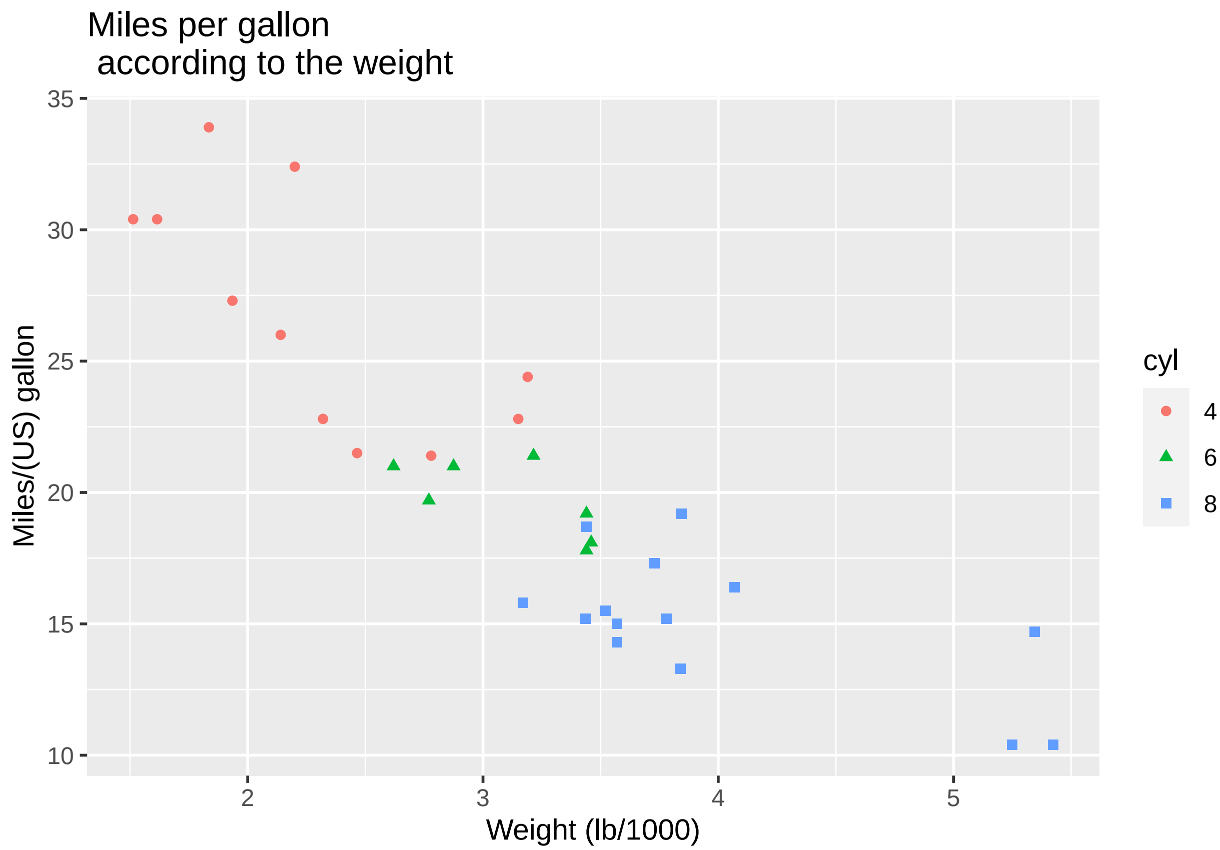
```
ggplot(mtcars, aes(x=wt, y=mpg)) +
  geom_point()+
  geom_smooth(method=lm, se=FALSE)
```

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



O ajuste parece adequado? E se olharmos os dados por grupo (cilindrada)? Algum padrão diferente fica evidente?

```r
ggplot(mtcars, aes(x=wt, y=mpg, shape=cyl, color=cyl)) +
  geom_point()+
    labs(title="Miles per gallon \n according to the weight",
       x="Weight (lb/1000)", y = "Miles/(US) gallon")
```



Vamos ajustar uma regressão para cada grupo de cilindrada. Os ajustes são similares?

```
ajuste.geral$coef
```

```
## (Intercept)          wt
##   37.285126   -5.344472
```

```
ajuste.cyl4 = lm(mpg~wt,data=mtcars[mtcars$cyl==4,])
ajuste.cyl4$coef
```

```
## (Intercept)          wt
##   39.571196   -5.647025
```

```
ajuste.cyl6 = lm(mpg~wt,data=mtcars[mtcars$cyl==6,])
ajuste.cyl6$coef
```

```
## (Intercept)          wt
##   28.408845   -2.780106
```

```
ajuste.cyl8 = lm(mpg~wt,data=mtcars[mtcars$cyl==8,])
ajuste.cyl8$coef
```

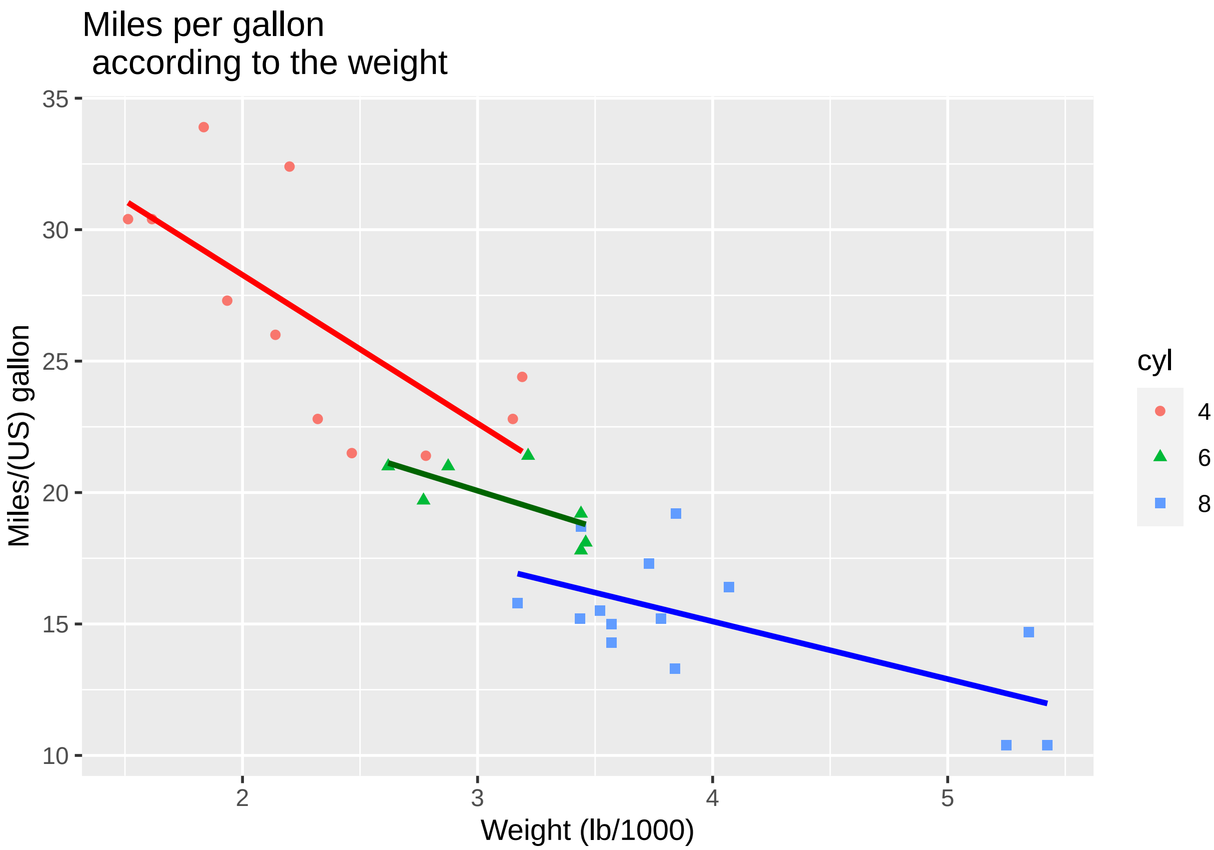
```
## (Intercept)          wt
##   23.868029   -2.192438
```

Nos gráficos:

```
ggplot(mtcars, aes(x=wt, y=mpg)) +
  geom_point(aes(x=wt, y=mpg,shape=cyl, color=cyl),data=mtcars)+
  geom_smooth(method=lm, se=FALSE, data=mtcars[mtcars$cyl==4,],color="red")+
  geom_smooth(method=lm, se=FALSE, data=mtcars[mtcars$cyl==6,],color="darkgreen")+
```

```
  geom_smooth(method=lm, se=FALSE, data=mtcars[mtcars$cyl==8,],color="blue")+
    labs(title="Miles per gallon \n according to the weight",
       x="Weight (lb/1000)", y = "Miles/(US) gallon")
```

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



Vamos comparar a previsão do modelo global e do modelo para o grupo cyl=8. Note como as 2 retas são diferentes e levam a previsões bem diferentes.

```
ajuste.geral = lm(mpg~wt,data=mtcars)
summary(ajuste.geral)
```

```
##
## Call:
## lm(formula = mpg ~ wt, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5432 -2.3647 -0.1252  1.4096  6.8727
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  37.2851     1.8776  19.858  < 2e-16 ***
## wt           -5.3445     0.5591  -9.559 1.29e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 3.046 on 30 degrees of freedom
## Multiple R-squared:  0.7528, Adjusted R-squared:  0.7446
## F-statistic: 91.38 on 1 and 30 DF,  p-value: 1.294e-10
```

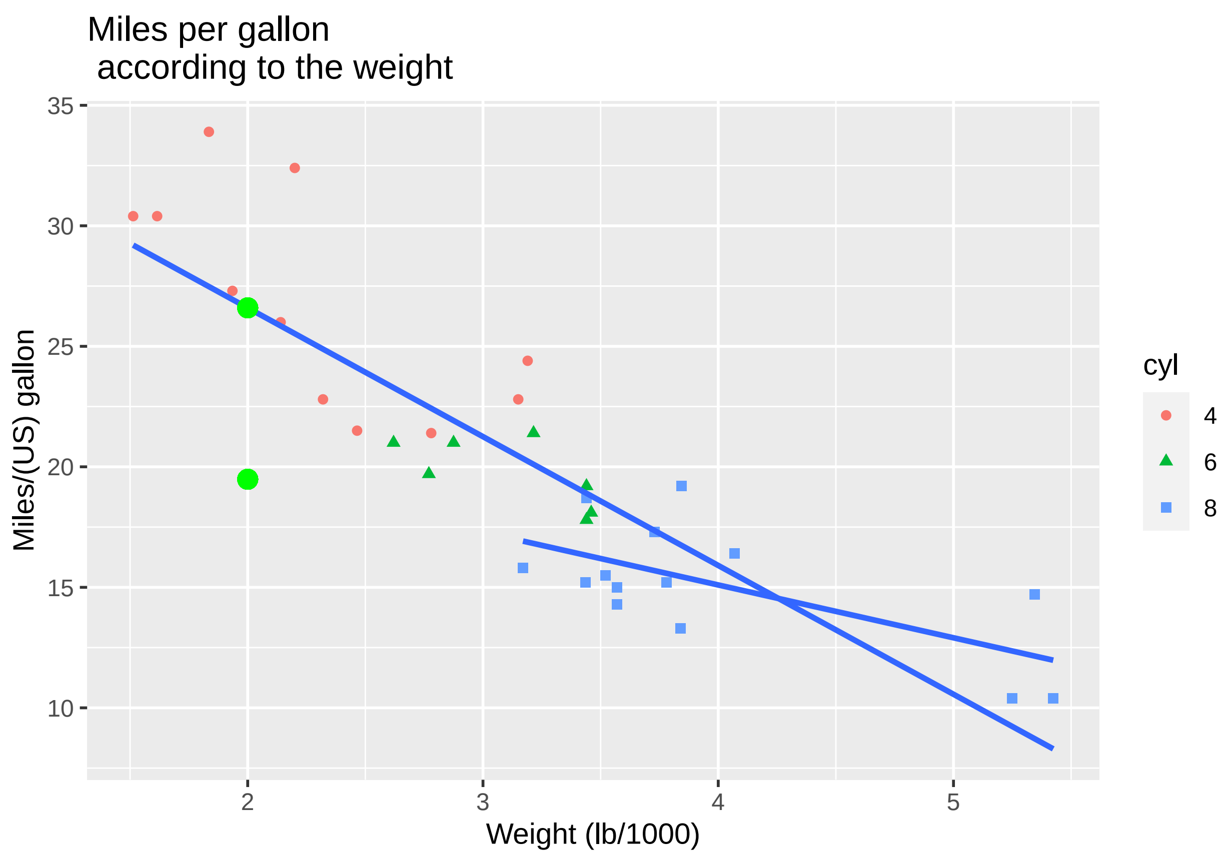
```
new.dt <- data.frame(wt = 2)

pred.g = predict.lm(ajuste.geral,newdata=new.dt)
pred.g
```

```
##        1
## 26.59618
```

```
pred.8 = predict.lm(ajuste.cyl8,newdata=new.dt)
pred.8
```

```
##        1
## 19.48315
```

```
ggplot(mtcars, aes(x=wt, y=mpg)) +
  geom_point(aes(x=wt, y=mpg,shape=cyl, color=cyl),data=mtcars)+
  geom_smooth(method=lm, se=FALSE, data=mtcars)+
  geom_smooth(method=lm, se=FALSE, data=mtcars[mtcars$cyl==8,])+
  geom_point(x=2,y=pred.g,color="green",size=3)+
  geom_point(x=2,y=pred.8,color="green",size=3)+
  labs(title="Miles per gallon \n according to the weight",
       x="Weight (lb/1000)", y = "Miles/(US) gallon")
```



Veremos que podemos fazer um ajuste conjunto com as variáveis wt, cyl no contexto de regressão linear múltipla.