## Modelos de regresión y predicción

### Regresión lineal de primer orden

Como vimos en el contenido teórico, la regresión es un modelo matemático que se utiliza para ajustar relaciones entre variables, es decir, para establecer mediante una o varias ecuaciones la relación entre variables dependientes (que son las salidas de un proceso o sistema) y las variables independientes (que son las entradas de este).

Para este ejemplo, haremos uso del grupo de datos *cars*, que contiene información sobre la velocidad de los automóviles y su distancia al frenar. Este conjunto de datos está conformado por dos variables, la velocidad en millas por hora y la distancia en pies.

Para dar un vistazo a este *dataset,* utilizamos la función *View(),* que se utiliza para visualizar un grupo de datos en forma de hoja de datos o tabla. La instrucción a realizar es la siguiente:

```
> view(cars)
```

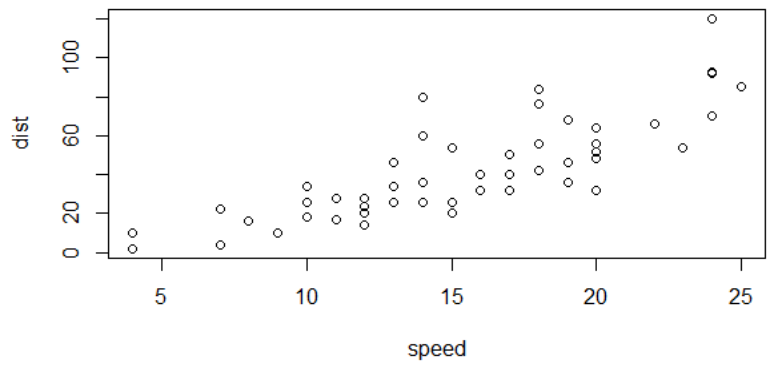
| | speed | dist |
|---|---|---|
| 1 | 4 | 2 |
| 2 | 4 | 10 |
| 3 | 7 | 4 |
| 4 | 7 | 22 |
| 5 | 8 | 16 |
| 6 | 9 | 10 |
| 7 | 10 | 18 |
| 8 | 10 | 26 |
| 9 | 10 | 34 |
| 10 | 11 | 17 |
| 11 | 11 | 28 |

Showing 1 to 12 of 50 entries, 2 total columns

Como puedes ver, es un grupo de datos que cuenta con 50 observaciones y 2 variables.

A continuación, grafica este grupo de datos para ver cómo están distribuidos. Para eso, haremos uso de la función plot.

```
> plot(dist ~ speed, data = cars)
```

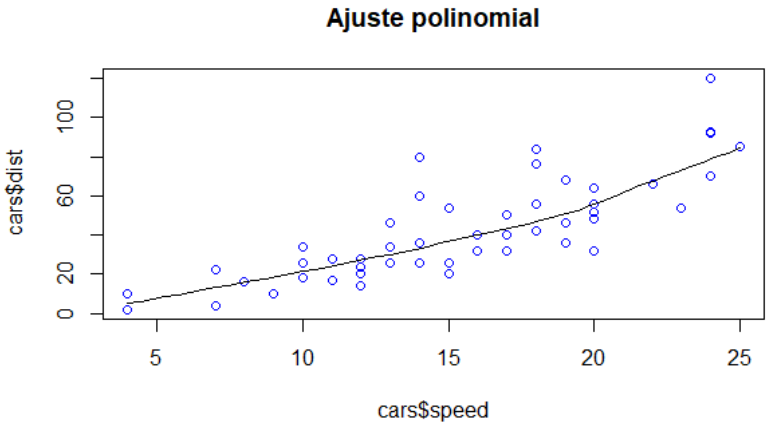


Es necesario calcular el coeficiente de correlación para ver si existe una relación lineal entre los datos:

```
> cor(cars$dist, cars$speed)
[1] 0.8068949
```

En este caso, el valor es positivo y alto, lo que indica que sí existe una relación lineal directa entre los datos (distancia con respecto a la velocidad). Ahora, realizarás un ajuste automático utilizando la función *scatter.smooth()*, la cual realiza un ajuste polinomial determinado por uno o más predictores numéricos para obtener lo siguiente:

```
> scatter.smooth(x=cars$speed, y=cars$dist, col = "blue", main="Ajuste polinomial")
```




Con ese gráfico podrás ver que se puede realizar un ajuste, sin embargo, el modelo polinomial podría ser muy complicado al contar con numerosos coeficientes. Debido a la alta correlación obtenida, realizarás un ajuste de un modelo simple de primer orden, el cual puede tipificarse con la siguiente ecuación:

$$Y= mx + b$$

En este caso ***x*** es la variable independiente; ***y*** es la variable dependiente; ***m*** es el coeficiente de pendiente y ***b*** es una constante también (de intersección con el eje *y*). Representado de la siguiente forma:

$$Y= b1\ x + b0$$

Ahora usarás la función *lm()*, la cual se utiliza para ajustar modelos lineales de la siguiente manera:

```
> ajuste <- lm(formula = dist ~ speed, data = cars)
> print(ajuste)

Call:
lm(formula = dist ~ speed, data = cars)

Coefficients:
(Intercept)        speed
    -17.579        3.932
```

Donde ***m*** = 3.932 y ***b*** = -17.579

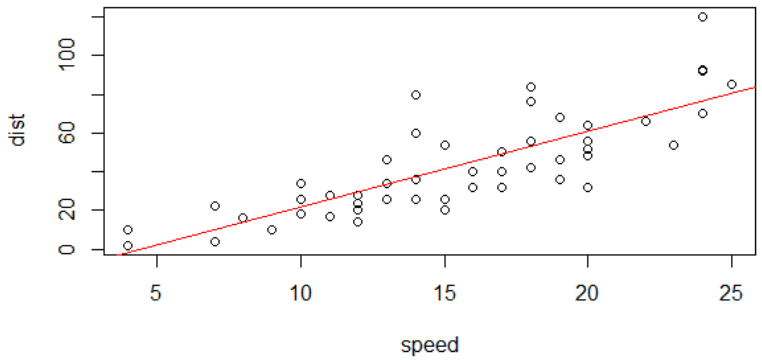
Por tanto, el modelo de regresión quedaría como lo siguiente:

**Y = 3.932 X - 17.579**

A continuación, graficarás de nuevo los datos y agregarás la aproximación del modelo de regresión lineal de la siguiente forma:

```
> plot(dist ~ speed, data = cars)
> abline(ajuste, col = "red")
```

Lo que da como resultado la siguiente gráfica:

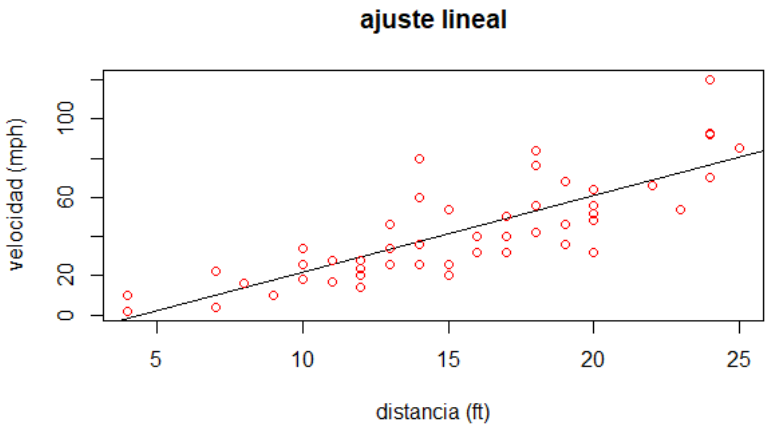



Otra forma de realizar el graficado de la aproximación es la que se muestra en el siguiente conjunto de instrucciones:

```
> x <- cars[,1]
> y <- cars[,2]

> plot(x, y, col = "red", main = "ajuste lineal", abline(lm(formula = dist ~ speed, dat
a = cars)), xlab = "distancia (ft)", ylab = "velocidad (mph)" )
```

Que dan como resultado la gráfica que se muestra a continuación:



Con la función *summary()* podrás obtener un resumen de la estadística del ajuste que se realizó con la regresión lineal:

```
> summary(ajuste)

Call:
lm(formula = dist ~ speed, data = cars)

Residuals:
    Min      1Q  Median      3Q     Max
-29.069  -9.525  -2.272   9.215  43.201

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -17.5791     6.7584  -2.601   0.0123 *
speed         3.9324     0.4155   9.464 1.49e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 15.38 on 48 degrees of freedom
Multiple R-squared:  0.6511,    Adjusted R-squared:  0.6438
F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

A partir del modelo de regresión lineal podrás predecir la velocidad con base en nuevas distancias, por ejemplo, si desearás conocer la velocidad para una distancia de 22:

```
> nuevo_ajuste <- lm(formula = y ~ x)
> nuevo_dato <- data.frame(x = 22)
> predict(nuevo_ajuste, nuevo_dato)
       1
68.9339
```

**Ahora es tu turno de practicar con el siguiente ejercicio:**

**Ejercicio:**

Investiga en bases de datos de tu comunidad, región o país los datos de niveles de concentración de ozono en el aire promediados por mes del año anterior, para que sean contrastados con los meses del año donde se presentan.

a) Realiza el modelo de inferencia de primer orden.
b) Grafica la aproximación lineal obtenida.
c) Estima el nivel de concentración de ozono que te tiene exactamente a la mitad de los meses de junio y julio.

## Regresión lineal de segundo orden

Las regresiones lineales de segundo orden se utilizan cuando la relación que guardan los datos con aquellos con los que se desea construir el modelo no parece a una línea recta. No obstante, se sigue tratando de datos que guardan una relación lineal. En esos casos, con un modelo de segundo orden se podrían obtener mejores resultados en la estimación.
El modelo de segundo orden puede expresarse con la siguiente ecuación:

$$y = b0 + b1X + b2X^2$$

Para realizar esta aproximación en R es necesario que realices el cálculo del cuadrado del valor de ***x*** antes de agregarlo a la ecuación, por ejemplo:

```
> ajuste2 <- lm(formula = dist ~ speed + I(speed^2), data = cars)
> ajuste2

Call:
lm(formula = dist ~ speed + I(speed^2), data = cars)

Coefficients:
(Intercept)        speed   I(speed^2)
    2.47014      0.91329      0.09996
```

Donde:

**b0 = 2.47014**
**b1 = 0.91329**
**b2 = 0.09996**

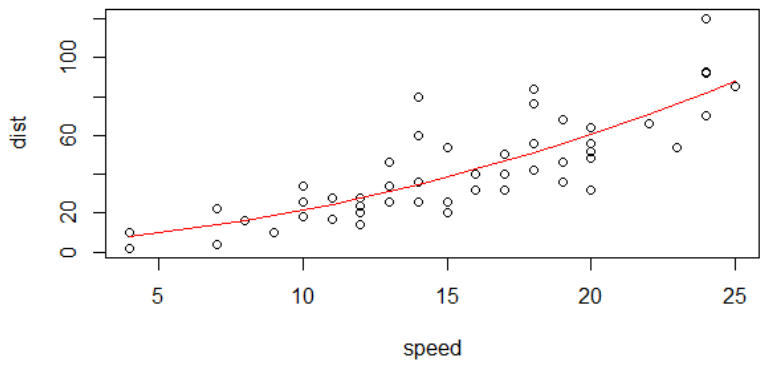
Por tanto, el modelo de predicción quedaría como lo siguiente:

$$y = 2.47014 + 0.91329\ X + 0.09996\ X^2$$

Para graficar el nuevo ajuste tendrás que realizar las siguientes instrucciones:

```
> plot(dist ~ speed, data = cars)
> lines(cars$speed, predict(ajuste2), col="red")
```

Con lo que se obtiene la gráfica que se muestra a continuación:



## Regresión lineal de tercer orden

Las regresiones lineales pueden seguir creciendo en orden cuando se desean ajustar modelos de gráficas que no se pueden tipificar con líneas rectas. Estos modelos pueden crecer en orden y número de coeficientes, pero esto no siempre significa que la estimación va a mejorar, por lo que la recomendación es incrementar el orden. Sin embargo, cuando el error de estimación ya no se reduce de forma considerable es cuando vale la pena apostar por un modelo más sencillo.

El modelo de tercer orden puede expresarse con la siguiente ecuación:

$$y = b0 + b1X + b2X^2 + b3X^3$$

Para realizar esta aproximación en R es necesario que realices el cálculo del cuadrado del valor de ***x***, así como el valor de **x** al cubo antes de agregarlo a la ecuación.

Las instrucciones de código para esto se pueden realizar de la siguiente forma:

```
> ajuste3 <- lm(formula = dist ~ speed + I(speed^2) + I(speed^3), data = cars)
> ajuste3

Call:
lm(formula = dist ~ speed + I(speed^2) + I(speed^3), data = cars)

Coefficients:
(Intercept)        speed   I(speed^2)   I(speed^3)
  -19.50505      6.80111     -0.34966      0.01025
```

Ahora tenemos lo siguiente:

**b0 = -19.50505**
**b1 = 6.80111**
**b2 = -0.34966**
**b3 = 0.01025**

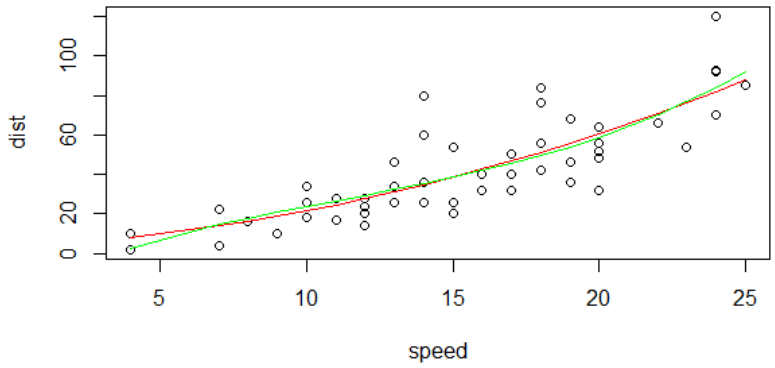
Es decir, que el modelo quedaría como lo siguiente:

$$y = -19.50505 + 6.80111\ X - 0.34966\ X^2 + 0.01025\ X^3$$

Para agregar la nueva aproximación de tercer orden a la gráfica actual basta con que introduzcas las nuevas líneas:

```
> lines(cars$speed, predict(ajuste3), col="green")
```

Ahora obtenemos la gráfica siguiente:




La estadística del tercer ajuste es la siguiente:

```
> summary(ajuste3)

Call:
lm(formula = dist ~ speed + I(speed^2) + I(speed^3), data = cars)

Residuals:
    Min      1Q  Median      3Q     Max
-26.670  -9.601  -2.231   7.075  44.691

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -19.50505   28.40530  -0.687    0.496
speed         6.80111    6.80113   1.000    0.323
I(speed^2)   -0.34966    0.49988  -0.699    0.488
I(speed^3)    0.01025    0.01130   0.907    0.369

Residual standard error: 15.2 on 46 degrees of freedom
Multiple R-squared:  0.6732,    Adjusted R-squared:  0.6519
F-statistic: 31.58 on 3 and 46 DF,  p-value: 3.074e-11
```

**Ahora es tu turno para practicar:**

**Ejercicio:**

Realiza los modelos de segundo y tercer orden con los mismos datos que utilizaste para el modelo de inferencia de primer orden:

a) Estima los coeficientes de ambos modelos.
b) Graficas las aproximaciones lineales obtenidas.
c) Compara el error de estimación de los tres modelos (primero, segundo y tercero) y justifica la selección de uno de estos.