

# Árboles de Regresión

jerf

17/6/2021

## Árboles de Regresión

```
dataset <- read.csv("Position_Salaries.csv")
dataset = dataset[, 2:3]
dataset
```

```
##      Level  Salary
## 1         1   45000
## 2         2   50000
## 3         3   60000
## 4         4   80000
## 5         5  110000
## 6         6  150000
## 7         7  200000
## 8         8  300000
## 9         9  500000
## 10        10 1000000
```

## Ajustar modelo de Regresión con el Conjunto de Datos

```
library("rpart")

## Warning: package 'rpart' was built under R version 4.0.5
regression = rpart(formula = Salary ~.,
                   data = dataset,
                   control = rpart.control(minsplit = 1))
```

## Haciendo predicciones con el Árbol de Regresión

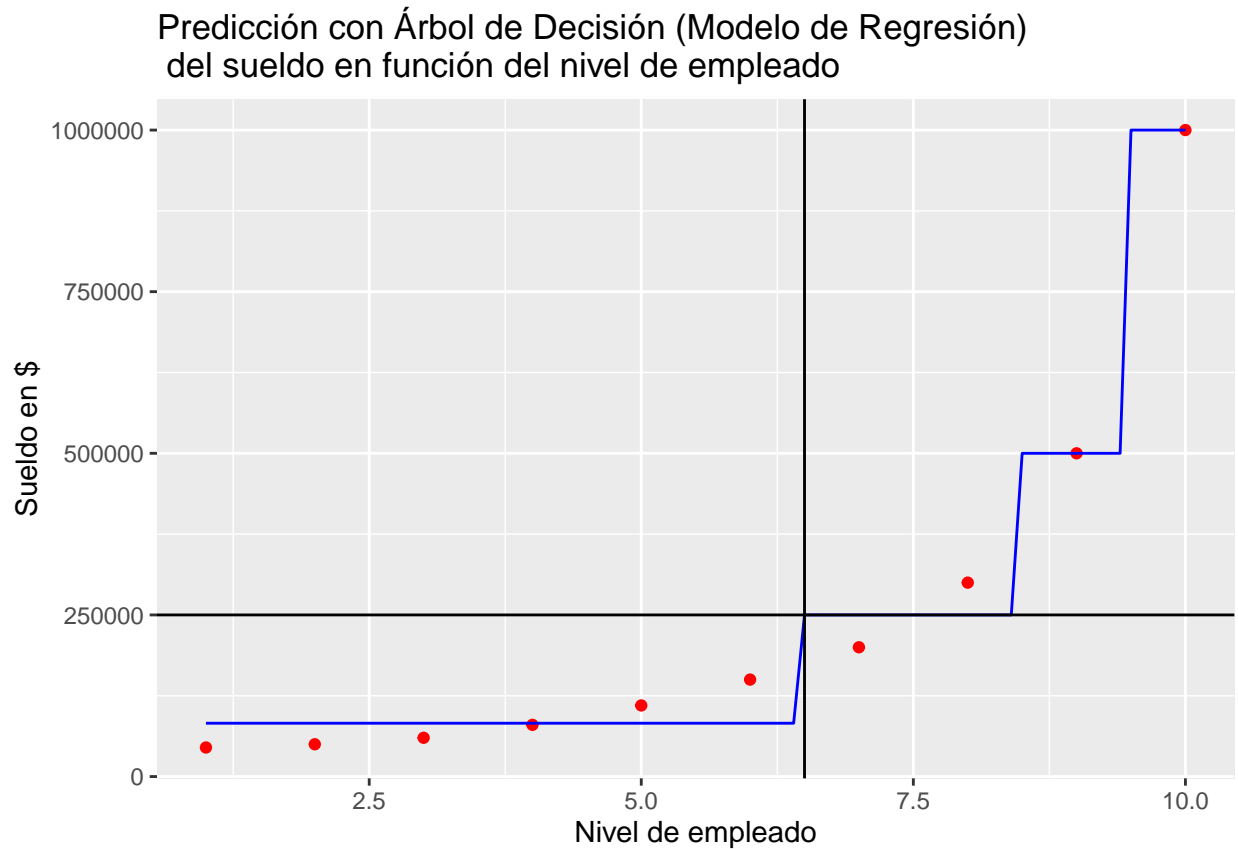
```
y_pred = predict(regression, newdata = data.frame(Level = 6.5))
y_pred

##      1
## 250000
```

## Visualización del modelo Polinómico

```
library(ggplot2)
X_grid = seq(min(dataset$Level), max(dataset$Level), 0.1)
```

```
ggplot() +
  geom_point(aes(x = dataset$Level, y = dataset$Salary),
             colour = "red") +
  geom_line(aes(x = X_grid, y = predict(regression, newdata = data.frame(Level = X_grid))),
            colour = "blue")+
  geom_vline(xintercept = 6.5) +
  geom_hline(yintercept = predict(regression, newdata = data.frame(Level = 6.5))) +
  xlab("Nivel de empleado") +
  ylab("Sueldo en $") +
  ggtitle("Predicción con Árbol de Decisión (Modelo de Regresión) \n del sueldo en función del nivel de
```



Al querer predecir, solo toma el más cercano por ello. Por eso al nivel de empleado 6.5 se le asigna el valor del nodo donde lo clasifica. Este algoritmo se mejora con la regresión de bosques aleatorios.