Decision Tree Regression

Sasha Ajay Malkani

2025-06-04

## Importing the Dataset

dataset = read.csv('Position\_Salaries.csv')  
  
print(dataset)

## Position Level Salary  
## 1 Business Analyst 1 45000  
## 2 Junior Consultant 2 50000  
## 3 Senior Consultant 3 60000  
## 4 Manager 4 80000  
## 5 Country Manager 5 110000  
## 6 Region Manager 6 150000  
## 7 Partner 7 200000  
## 8 Senior Partner 8 300000  
## 9 C-level 9 500000  
## 10 CEO 10 1000000

## Reset the dataset

dataset = dataset[2:3]  
  
print(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

## Fitting Regression Model to the dataset

#install.packages('rpart')  
library(rpart)  
  
regressor = rpart(formula = Salary ~ .,   
 data = dataset,  
 control = rpart.control(minsplit = 1))  
  
summary(regressor)

## Call:  
## rpart(formula = Salary ~ ., data = dataset, control = rpart.control(minsplit = 1))  
## n= 10   
##   
## CP nsplit rel error xerror xstd  
## 1 0.77638626 0 1.00000000 1.234568 0.7835133  
## 2 0.15496716 1 0.22361374 1.234568 0.7835133  
## 3 0.05217357 2 0.06864658 1.234568 0.7835133  
## 4 0.01000000 3 0.01647301 1.234568 0.7835133  
##   
## Variable importance  
## Level   
## 100   
##   
## Node number 1: 10 observations, complexity param=0.7763863  
## mean=249500, MSE=8.066225e+10   
## left son=2 (8 obs) right son=3 (2 obs)  
## Primary splits:  
## Level < 8.5 to the left, improve=0.7763863, (0 missing)  
##   
## Node number 2: 8 observations, complexity param=0.05217357  
## mean=124375, MSE=6.921484e+09   
## left son=4 (6 obs) right son=5 (2 obs)  
## Primary splits:  
## Level < 6.5 to the left, improve=0.7600316, (0 missing)  
##   
## Node number 3: 2 observations, complexity param=0.1549672  
## mean=750000, MSE=6.25e+10   
## left son=6 (1 obs) right son=7 (1 obs)  
## Primary splits:  
## Level < 9.5 to the left, improve=1, (0 missing)  
##   
## Node number 4: 6 observations  
## mean=82500, MSE=1.38125e+09   
##   
## Node number 5: 2 observations  
## mean=250000, MSE=2.5e+09   
##   
## Node number 6: 1 observations  
## mean=500000, MSE=0   
##   
## Node number 7: 1 observations  
## mean=1000000, MSE=0

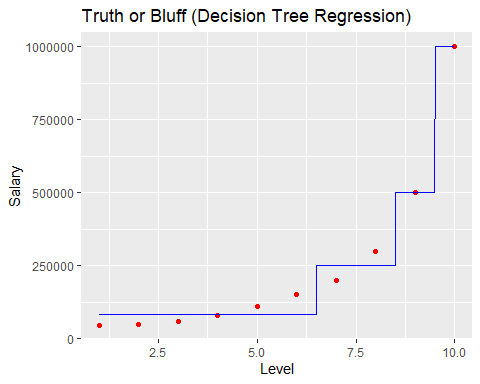
## Predicting a new result

y\_pred = predict(regressor, data.frame(Level = 6.5))  
  
print(y\_pred)

## 1   
## 250000

## Visualizing the Regression Model results (for higher resolution and smoother curve)

# install.packages('ggplot2')  
library(ggplot2)  
x\_grid = seq(min(dataset$Level), max(dataset$Level), 0.01)  
ggplot() +  
 geom\_point(aes(x = dataset$Level, y = dataset$Salary),  
 colour = 'red') +  
 geom\_line(aes(x = x\_grid, y = predict(regressor, newdata = data.frame(Level = x\_grid))),  
 colour = 'blue') +  
 ggtitle('Truth or Bluff (Decision Tree Regression)') +  
 xlab('Level') +  
 ylab('Salary')



Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.