RLS

Juan Manuel Cabrera

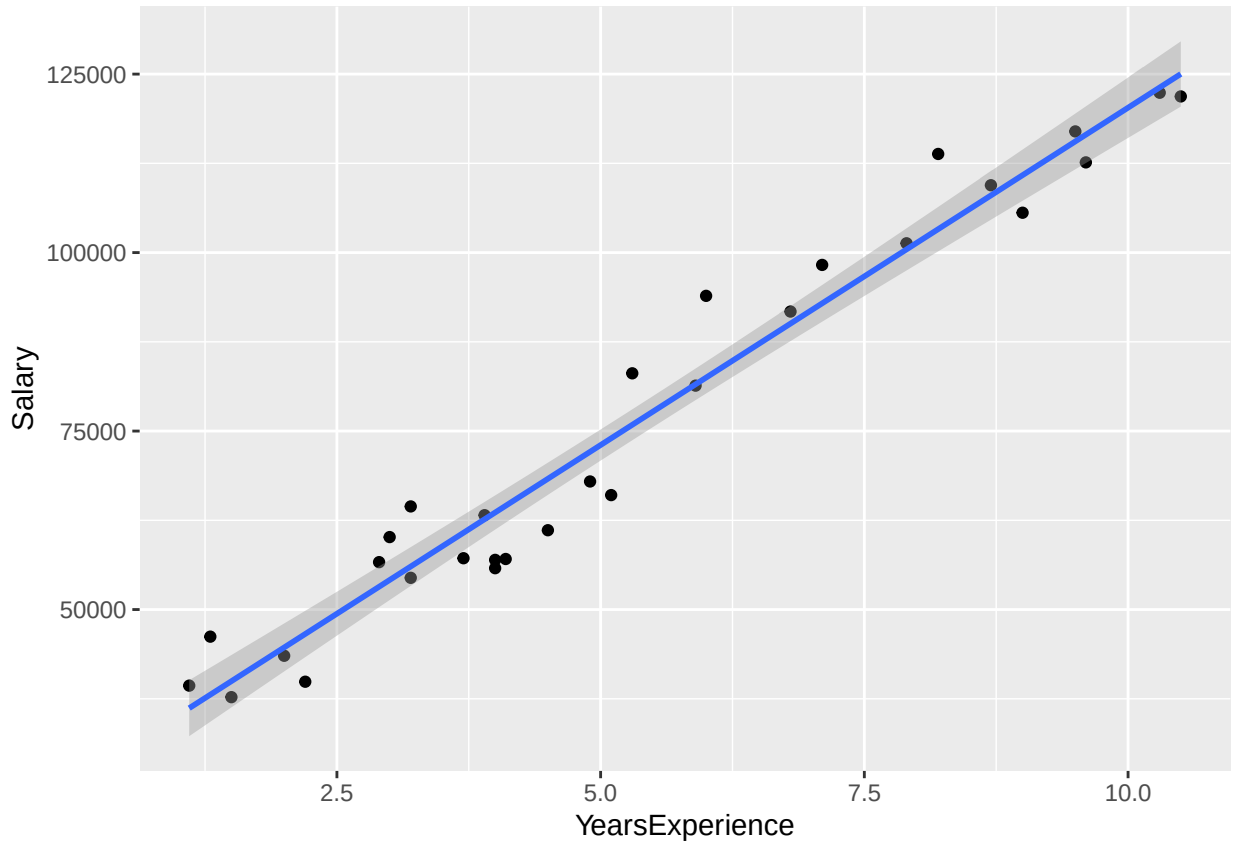
2023-07-29

```
#Librarias
library(caTools)
library(ggplot2)
```

```
#Importar dataset
salary <- read.csv('Datas/Salary_Data.csv')
```

```
#Gráfico de dispersión
ggplot(salary, aes(YearsExperience, Salary))+
  geom_point() +
  stat_smooth(method="lm")
```

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



```r
#Modelado
model <- lm(Salary ~ YearsExperience,
            data = salary)

summary(model)
```

```
##
## Call:
## lm(formula = Salary ~ YearsExperience, data = salary)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7958.0 -4088.5  -459.9  3372.6 11448.0
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      25792.2     2273.1   11.35 5.51e-12 ***
## YearsExperience   9450.0      378.8   24.95  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5788 on 28 degrees of freedom
## Multiple R-squared:  0.957,  Adjusted R-squared:  0.9554
## F-statistic: 622.5 on 1 and 28 DF,  p-value: < 2.2e-16
```

Resultado de los datos:

Ecuación del modelo Y = 25792.2 + 9450*x

p-value inferior al 0.05 (nivel de significación del 5%)

R2 = 0.957

La prueba F: F(1,28) = 622, p < 0.001)

```r
sigma(model)
```

```
## [1] 5788.315
```

Error estándar residual (RSE) o sigma = 5788.315

```r
sigma(model)*100 / mean(salary$Salary)
```

```
## [1] 7.615903
```

Tasa de error de predicción = 7.615903 %

```r
summary(model)$coefficients
```

```
##                  Estimate Std. Error  t value     Pr(>|t|)
## (Intercept)     25792.200  2273.0534 11.34694 5.511950e-12
## YearsExperience  9449.962   378.7546 24.95009 1.143068e-20
```

```
confint(model)
```

```
##                     2.5 %   97.5 %
## (Intercept)     21136.061 30448.34
## YearsExperience  8674.119 10225.81
```

```
#Dividimos el data set en set de training y set de test.
set.seed(123)
split <- sample.split(salary$Salary, SplitRatio = 2/3)
training_set <- subset(salary, split == T)
test_set <- subset(salary, split == F)
```

Prediciendo resultados:

```
y_pred <- predict(model, newdate = test_set)
```

Graficamos

```
ggplot() +
  geom_point(aes(training_set$YearsExperience, training_set$Salary), color='red') +

  geom_point(aes(test_set$YearsExperience, predict(model, test_set)), color='green') +

  geom_line(aes(training_set$YearsExperience, predict(model, training_set)), color='blue')
```