Práctica

# Regrecion Lineal

---

Luis Eduardo Galindo Amaya (1274895)

| | |
|---|---|
| Asignatura | Estadística Avanzada |
| Docente | Olivia Mendoza Duarte |
| Fecha | 2022-09-15 |

# Regrecion Lineal

Luis Eduardo Galindo Amaya (1274895)

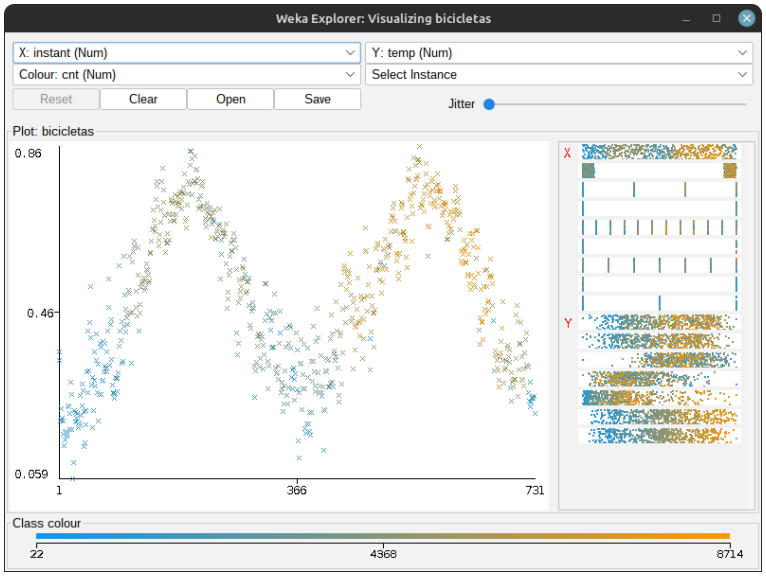
2022-09-15

## Bicis Por Dia - Temp

prompt

```
=== Classifier model (full training set) ===

Linear regression on temp
173.71 * temp + 279.95

Predicting 0 if attribute value is missing.
Time taken to build model: 0 seconds

=== Cross-validation ===
=== Summary ===

Correlation coefficient                  0.1338
Mean absolute error                    182.9304
Root mean squared error                209.1587
Relative absolute error                 99.9184 %
Root relative squared error             98.979  %
Total Number of Instances              731
```

captura

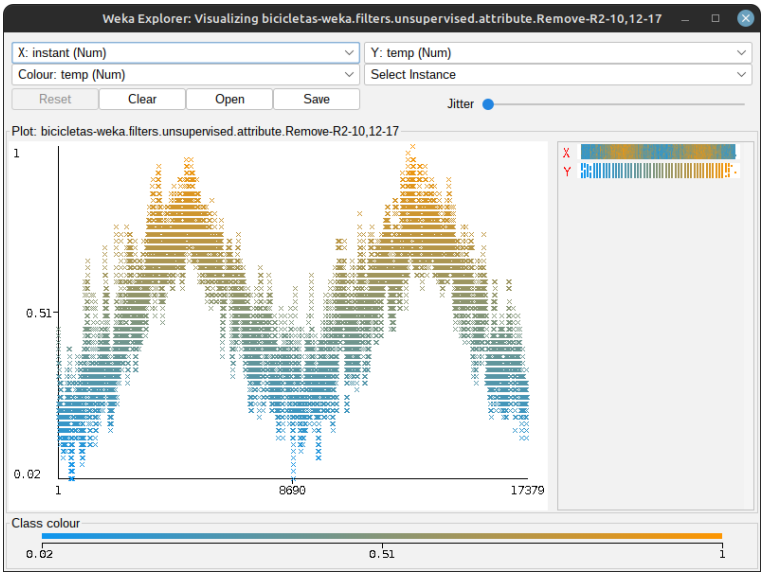



# Bicis por Hora - Temp

Prompt

```
=== Classifier model (full training set) ===

Linear regression on temp

3548.1 * temp + 6926.64

Predicting 0 if attribute value is missing.
Time taken to build model: 0 seconds

=== Cross-validation ===
=== Summary ===
Correlation coefficient                  0.1355
Mean absolute error                   4340.2185
Root mean squared error               4970.6098
Relative absolute error                 99.8908 %
Root relative squared error             99.0728 %
Total Number of Instances           17379
```

Captura

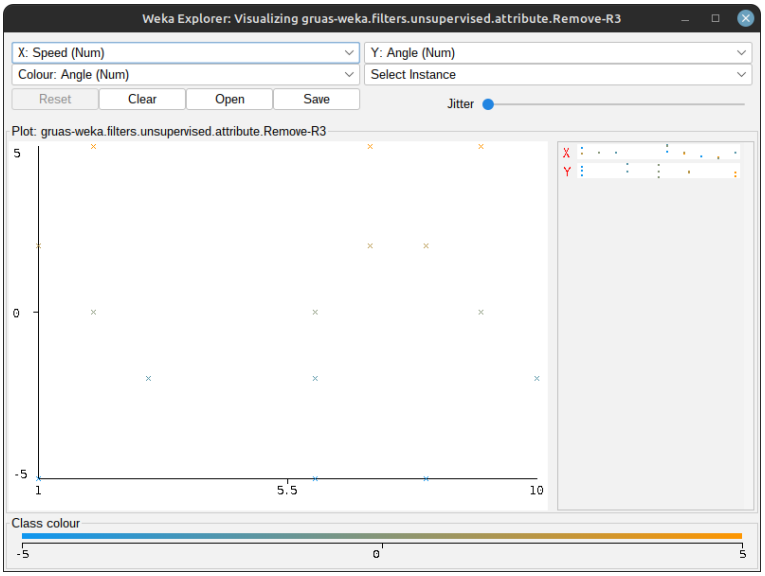



# Gruas - Angle

prompt

```
=== Classifier model (full training set) ===

Linear regression on Speed

0.07 * Speed - 0.38

Predicting 0 if attribute value is missing.
Time taken to build model: 0 seconds

=== Cross-validation ===
=== Summary ===

Correlation coefficient                  -0.6982
Mean absolute error                       3.3747
Root mean squared error                   4.0533
Relative absolute error                 111.6984 %
Root relative squared error             111.0355 %
Total Number of Instances                15
```

Captura




# Código para regrecion linel simple en R

```
## Eje x
x = c(1:10)
## Eje Y
y = c(34, 36, 19, 20, 22, 20, 19, 13, 15, 16)
n = length(y)

if( length(y) != length(x) ) {
    stop("Los arreglos son de simenciones diferentes")
}

## multiplicacion de elementos
xy = x*y

## elementos al cuadrado
xp2 = x^2

## pendiente
m = (n * sum(xy) - sum(x)*sum(y))/(n*sum(xp2) - sum(xp2))

## ordenada de origen
b = (sum(y) - m*sum(x))/n

## termina
print(sprintf("%fx + %f", m, b))
```