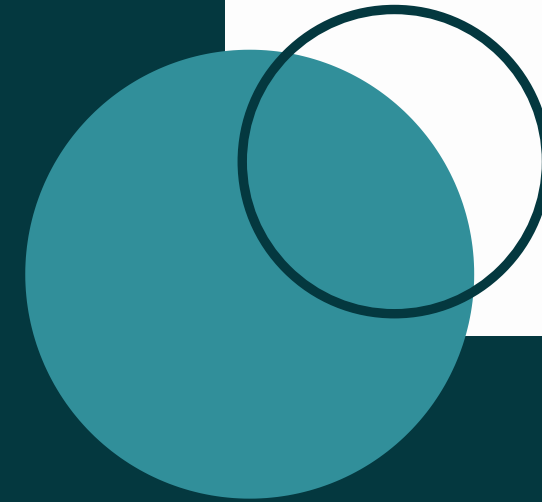


TÉCNICAS DE MINERÍA DE DATOS

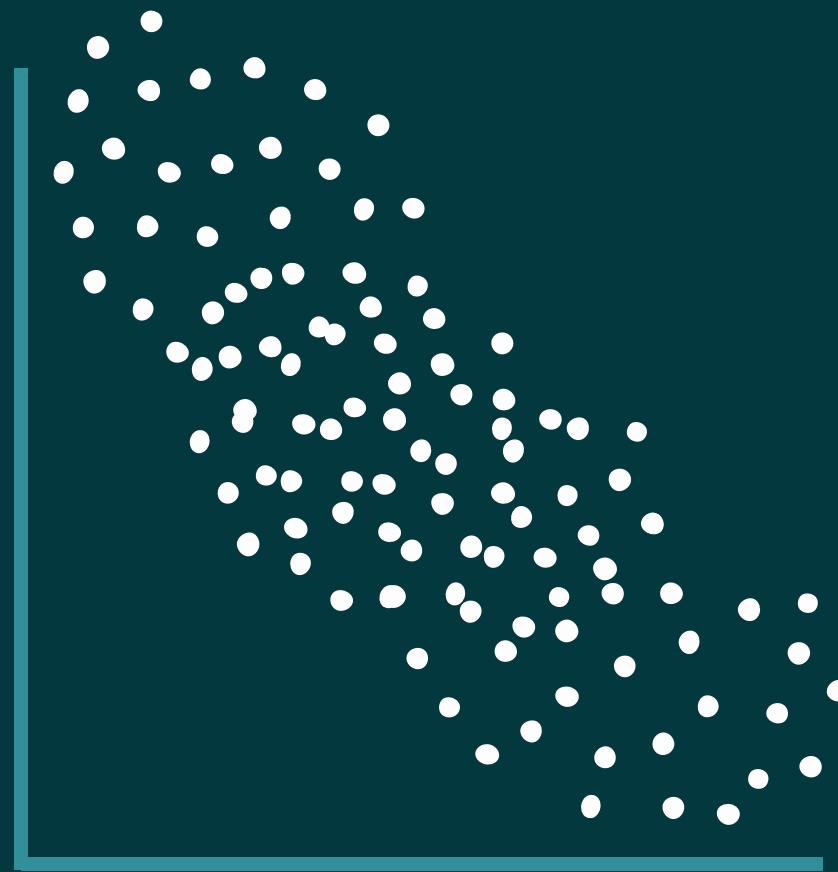
Regresión Lineal

1851448 Muñoz Barrientos Regina
1941592 Lagos Martínez José Alejandro
1849202 Domínguez Victorino Cesar Oswaldo
1793775 Rodríguez Guerrero Luisa Victoria

Equipo 1
Grupo 002



REGRESIÓN LINEAL SIMPLE



$$r = \sqrt{1 - \frac{\sum(Y - \hat{Y})^2}{\sum(Y - \bar{Y})^2}}$$

$$SSE = \sum (Y - \hat{Y})^2$$

Variación de Y alrededor de la recta.

$$SST = \sum (Y - \bar{Y})^2$$

Variación de Y alrededor de la media

$$y = \beta_0 + \beta_1 x + \varepsilon$$

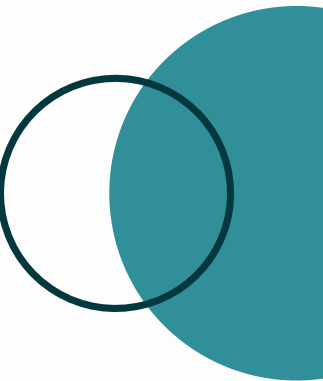
LA FORMULA DE MINIMOS CUADRADOS QUEDA:

$$S(\beta_0, \beta_1) = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2 = \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2$$

SE DERIVA:

$$\frac{\partial S}{\partial \beta_0} = -2 \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = 0$$

$$\frac{\partial S}{\partial \beta_1} = -2 \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) x_i = 0$$

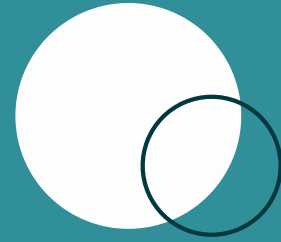


REGRESIÓN LINEAL SIMPLE

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n x_i y_i - \frac{(\sum_{i=1}^n x_i)(\sum_{i=1}^n y_i)}{n}}{\sum_{i=1}^n x_i^2 - \frac{(\sum_{i=1}^n x_i)^2}{n}}$$

En la ecuación general de la recta de regresión, B1 es la pendiente y B0 el valor de la variable dependiente Y para la que X = 0.

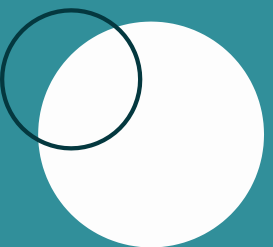


REGRESIÓN LINEAL MÚLTIPLE



Un modelo de regresión donde interviene más de una variable regresora, se llama modelo de regresión múltiple, el cual maneja el siguiente modelo:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \varepsilon$$





LA FORMULA DE MINIMOS CUADRADOS QUEDA:

$$S(\beta_0, \beta_1, \dots, \beta_k) = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (y_i - \beta_0 - \sum_{j=1}^k \beta_j x_{ij})^2$$

SE DERIVA:

$$\frac{\delta S}{\delta \beta_0} = -2 \sum_{i=1}^n \left(y_i - \widehat{\beta}_0 - \sum_{j=1}^k \widehat{\beta}_j x_{ij} \right) = 0$$

$$\frac{\delta S}{\delta \beta_j} = -2 \sum_{i=1}^n \left(y_i - \widehat{\beta}_0 - \sum_{j=1}^k \widehat{\beta}_j x_{ij} \right) x_{ij} = 0$$



”

ECUACIÓN DE MINIMOS CUADRADOS:

$$n\widehat{\beta}_0 + \widehat{\beta}_1 \sum_{i=1}^n x_{i1} + \widehat{\beta}_2 \sum_{i=1}^n x_{i2} + \cdots + \widehat{\beta}_k \sum_{i=1}^n x_{ik} = \sum_{i=1}^n y_i$$

$$\widehat{\beta}_0 \sum_{i=1}^n x_{i1} + \widehat{\beta}_1 \sum_{i=1}^n x_{i1}^2 + \widehat{\beta}_2 \sum_{i=1}^n x_{i1}x_{i2} + \cdots + \widehat{\beta}_k \sum_{i=1}^n x_{i1}x_{ik} = \sum_{i=1}^n x_{i1}y_i$$

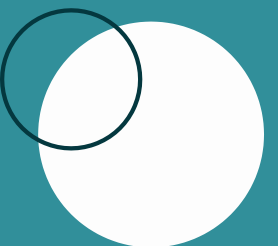
$$\vdots$$

$$\widehat{\beta}_0 \sum_{i=1}^n x_{ik} + \widehat{\beta}_1 \sum_{i=1}^n x_{ik}x_{i1} + \widehat{\beta}_2 \sum_{i=1}^n x_{ik}x_{i2} + \cdots + \widehat{\beta}_k \sum_{i=1}^n x_{ik}^2 = \sum_{i=1}^n x_{ik}y_i$$

ECUACIÓN MATRICIAL DEL MODELO:

$$y = x\beta + \varepsilon$$

“





LOS VALORES ESTÁN DADOS POR:

”

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad X = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix} \quad \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} \quad e = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$

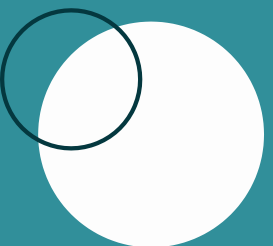
ECUACIÓN DE MÍNIMOS
CUADRADOS:

$$X'X\hat{\beta} = X'y$$

ESTIMADOR DE MÍNIMOS
CUADRADOS:

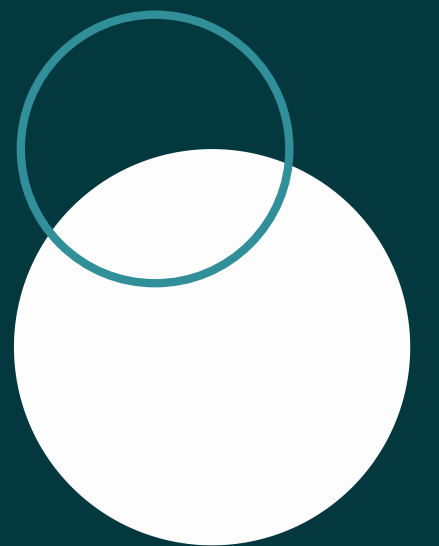
$$\hat{\beta} = (X'X)^{-1}X'y$$

“



MULTICOLINEALIDAD

La multicolinealidad es la relación de dependencia lineal fuerte entre más de dos variables explicativas en una regresión múltiple.

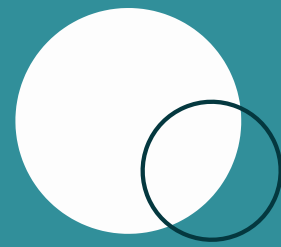


¿COMO MEDIRLA?

Obtenemos el factor de inflación de varianza para el j-ésimo coeficiente de la regresión.

$$VIF_j = \frac{1}{1 - R_j^2}$$

Los factores $VIF > 10$ presentan problemas de multicolinealidad.



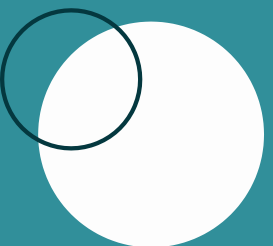
PRUEBA DE SIGNIFICANCIA



H0: $B1 = 0$, la regresión no es significativa

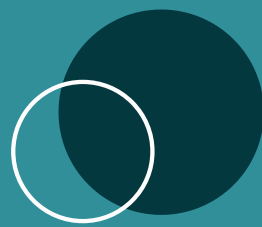
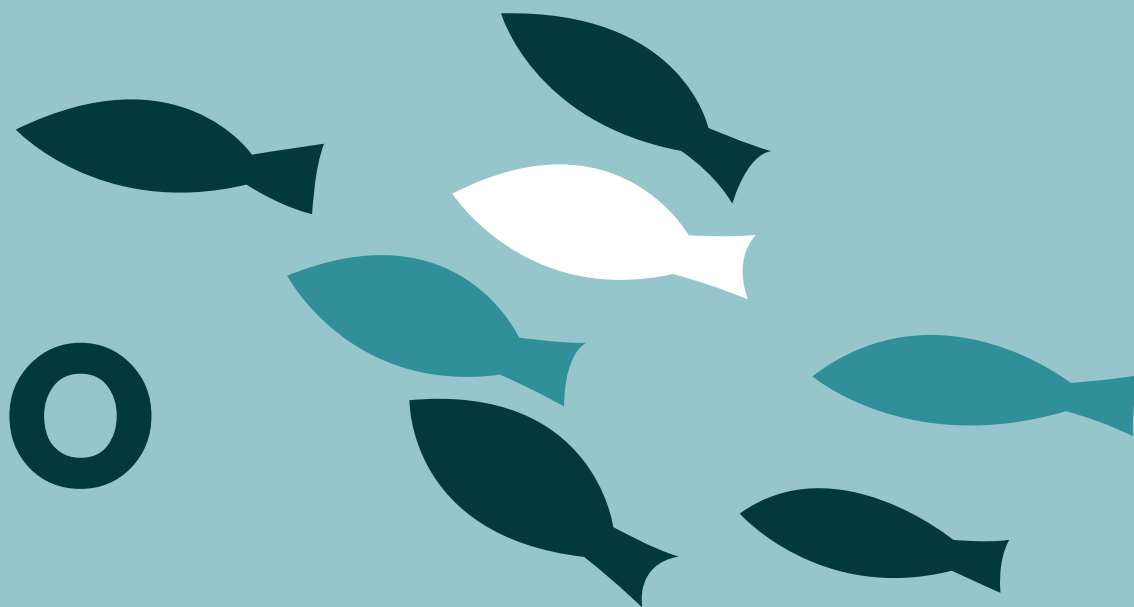
H1: $B1 \neq 0$, la regresión es significativa

Si la regresión es significativa entonces el modelo si se ajusta al comportamiento de los datos.






EJEMPLO



FISH MARKET DATASET

Predecir el peso de los peces
de la especie "Perch".





IMPORTAR LIBRERIAS Y DATOS

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt # Visualización
import seaborn as sns #Visualización
from statsmodels.stats.anova import anova_lm
from statsmodels.formula.api import ols
df = pd.read_csv("Fish.csv")
```

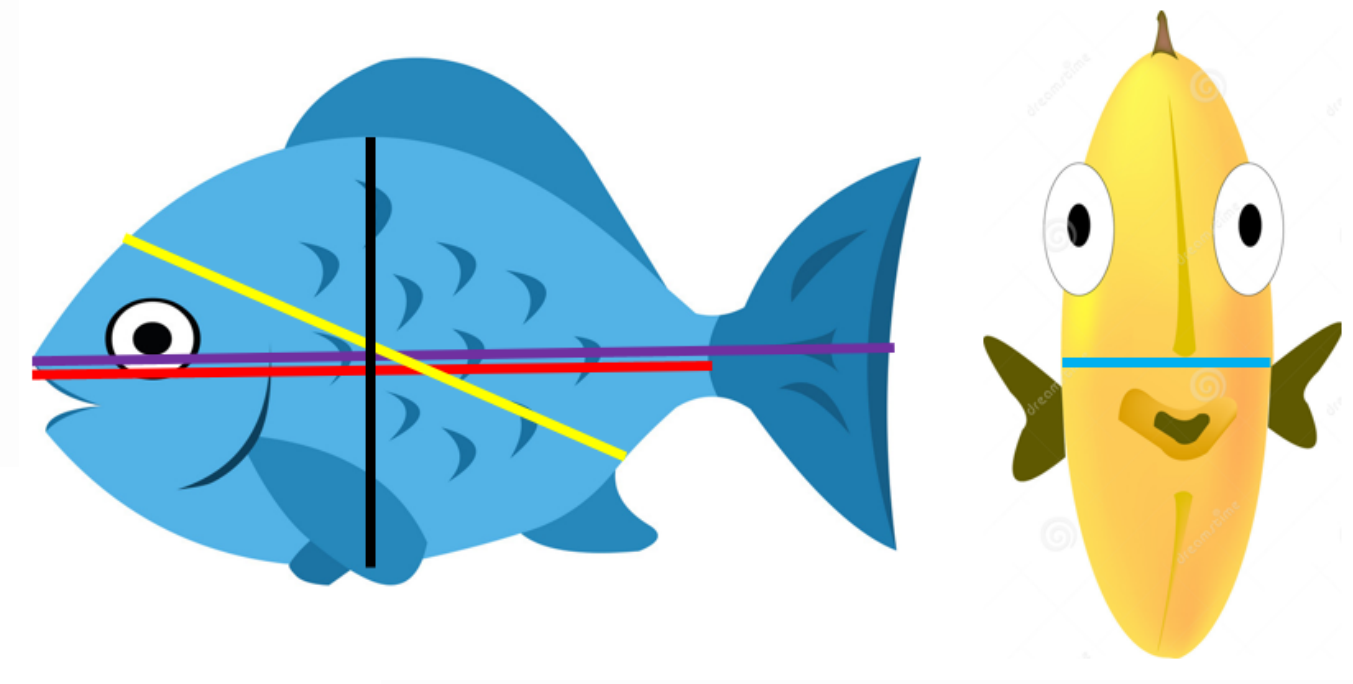
DATASET



```
print(df.head())  
print(df.shape)
```

	Species	Weight	Length1	Length2	Length3	Height	Width
0	Bream	242.0	23.2	25.4	30.0	11.5200	4.0200
1	Bream	290.0	24.0	26.3	31.2	12.4800	4.3056
2	Bream	340.0	23.9	26.5	31.1	12.3778	4.6961
3	Bream	363.0	26.3	29.0	33.5	12.7300	4.4555
4	Bream	430.0	26.5	29.0	34.0	12.4440	5.1340

(159, 7)



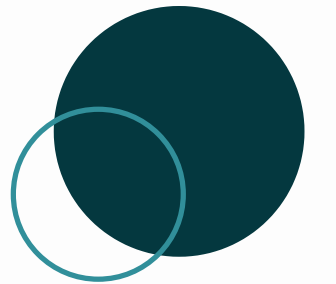
BUSQUEDA DE DATOS NULOS

```
df.isnull().sum()
```

```
Species      0  
Weight       0  
Length1      0  
Length2      0  
Length3      0  
Height       0  
Width        0  
dtype: int64
```



EXTRACCION DE DATOS "PERCH"

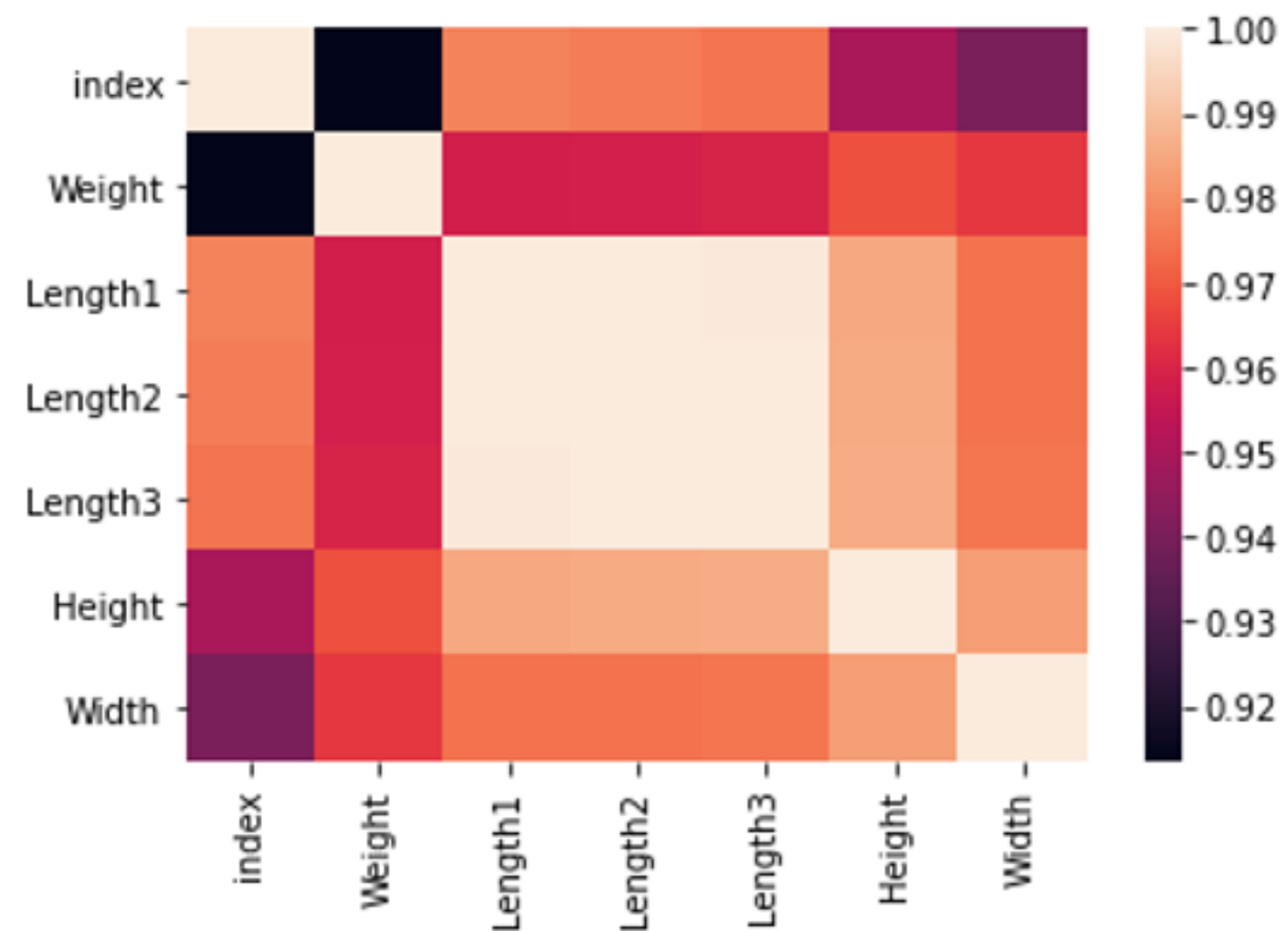


```
perch=df[df['Species']=='Perch']  
perch=perch.drop("Species",axis=1)  
perch=perch.reset_index()  
perch=perch.drop("index",axis=1)
```


CORRELOGRAMA

```
corr = perch.corr(method='pearson')  
sns.heatmap(corr, xticklabels = corr.columns.values, yticklabels=corr.columns.values)
```

<matplotlib.axes._subplots.AxesSubplot at 0x7fcd6edb2950>

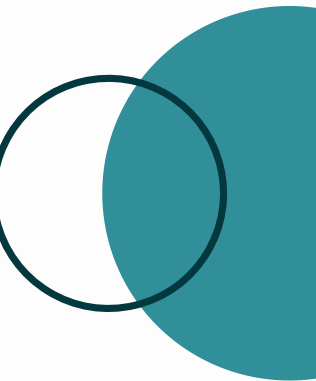
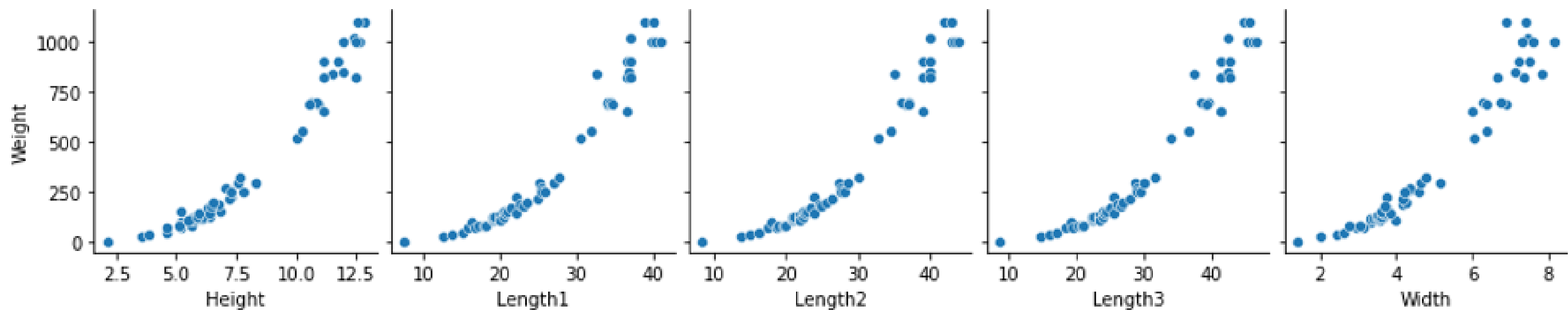


DISPERSIÓN

WEIGHT VS OTRAS VARIABLES

```
sns.pairplot(perch, x_vars=['Height', 'Length1', 'Length2', 'Length3', 'Width'], y_vars='Weight')
```

<seaborn.axisgrid.PairGrid at 0x7fcd6ec65410>



```
from sklearn.feature_selection import RFE
from sklearn.linear_model import LinearRegression
import statsmodels.api as sm
from statsmodels.stats.outliers_influence import variance_inflation_factor
```

```
# Fit the model
y = perch["Weight"]
x1 = perch["Length1"]
x2 = perch["Length2"]
x3 = perch["Length3"]
x4 = perch["Height"]
x5 = perch["Width"]
d = {"x1": x1, "x2": x2, "x3": x3, "x4": x4, "x5": x5}
X=pd.DataFrame(d)
```

```
def build_model(X,y):
    X = sm.add_constant(X) #Adding the constant
    lm = sm.OLS(y,X).fit() # fitting the model
    print(lm.summary()) # model summary
    return X
```

```
def checkVIF(X):
    vif = pd.DataFrame()
    vif['Features'] = X.columns
    vif['VIF'] = [variance_inflation_factor(X.values, i) for i in range(X.shape[1])]
    vif['VIF'] = round(vif['VIF'], 2)
    vif = vif.sort_values(by = "VIF", ascending = False)
    return(vif)
```

DECLARACION DE VARIABLES Y DEFINICIONES



MODELO

```
[141] model1=build_model(X,y)
```

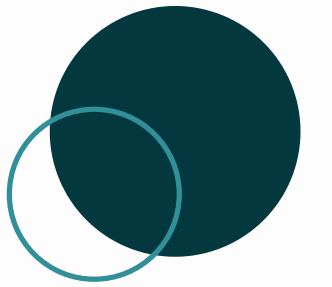
OLS Regression Results

Dep. Variable:	Weight	R-squared:	0.943			
Model:	OLS	Adj. R-squared:	0.937			
Method:	Least Squares	F-statistic:	165.2			
Date:	Wed, 08 Sep 2021	Prob (F-statistic):	7.60e-30			
Time:	19:48:50	Log-Likelihood:	-326.45			
No. Observations:	56	AIC:	664.9			
Df Residuals:	50	BIC:	677.0			
Df Model:	5					
Covariance Type:	nonrobust					
=====						
	coef	std err	t	P> t	[0.025	0.975]

const	-556.5865	60.663	-9.175	0.000	-678.431	-434.741
x1	-3.1302	57.880	-0.054	0.957	-119.386	113.126
x2	-38.5019	88.552	-0.435	0.666	-216.364	139.361
x3	42.9174	60.088	0.714	0.478	-77.773	163.608
x4	65.6555	29.998	2.189	0.033	5.402	125.909
x5	64.9023	36.765	1.765	0.084	-8.943	138.748
=====						
Omnibus:	21.244	Durbin-Watson:	0.500			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	29.351			
Skew:	1.415	Prob(JB):	4.23e-07			
Kurtosis:	5.137	Cond. No.	495.			
=====						

Ecuación del modelo:

$$\hat{y} = -556.58 - 3.13x_1 - 38.5x_2 + 42.91x_3 + 65.65x_4 + 64.9x_5$$



```
checkVIF(model1)
```

	Variables	VIF
0	const	27.16
1	x1	1780.08
2	x2	4626.41
3	x3	2376.77
4	x4	54.04
5	x5	30.86

VIF

Como todos los VIFs son altos, hay problemas de multicolinealidad.

REGRESIÓN SIMPLE

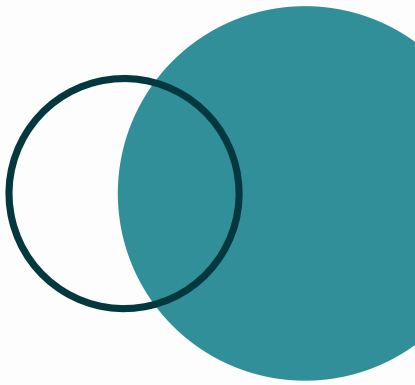
```
d1 = {"x4":x4}
X_1=pd.DataFrame(d1)
mod2=build_model(X_1,y)
```

OLS Regression Results

```
=====
Dep. Variable:          Weight      R-squared:            0.938
Model:                OLS      Adj. R-squared:         0.937
Method:             Least Squares   F-statistic:          815.2
Date:                Wed, 08 Sep 2021   Prob (F-statistic):    2.92e-34
Time:                  20:18:55   Log-Likelihood:       -328.82
No. Observations:          56      AIC:                661.6
Df Residuals:              54      BIC:                665.7
Df Model:                  1
Covariance Type:          nonrobust
=====
```

	coef	std err	t	P> t	[0.025	0.975]
const	-537.3275	34.260	-15.684	0.000	-606.015	-468.640
x4	116.9654	4.096	28.553	0.000	108.752	125.178

```
=====
Omnibus:                11.275   Durbin-Watson:           0.678
Prob(Omnibus):           0.004   Jarque-Bera (JB):        11.319
Skew:                    0.954   Prob(JB):                0.00349
Kurtosis:                 4.099   Cond. No.                 24.8
=====
```



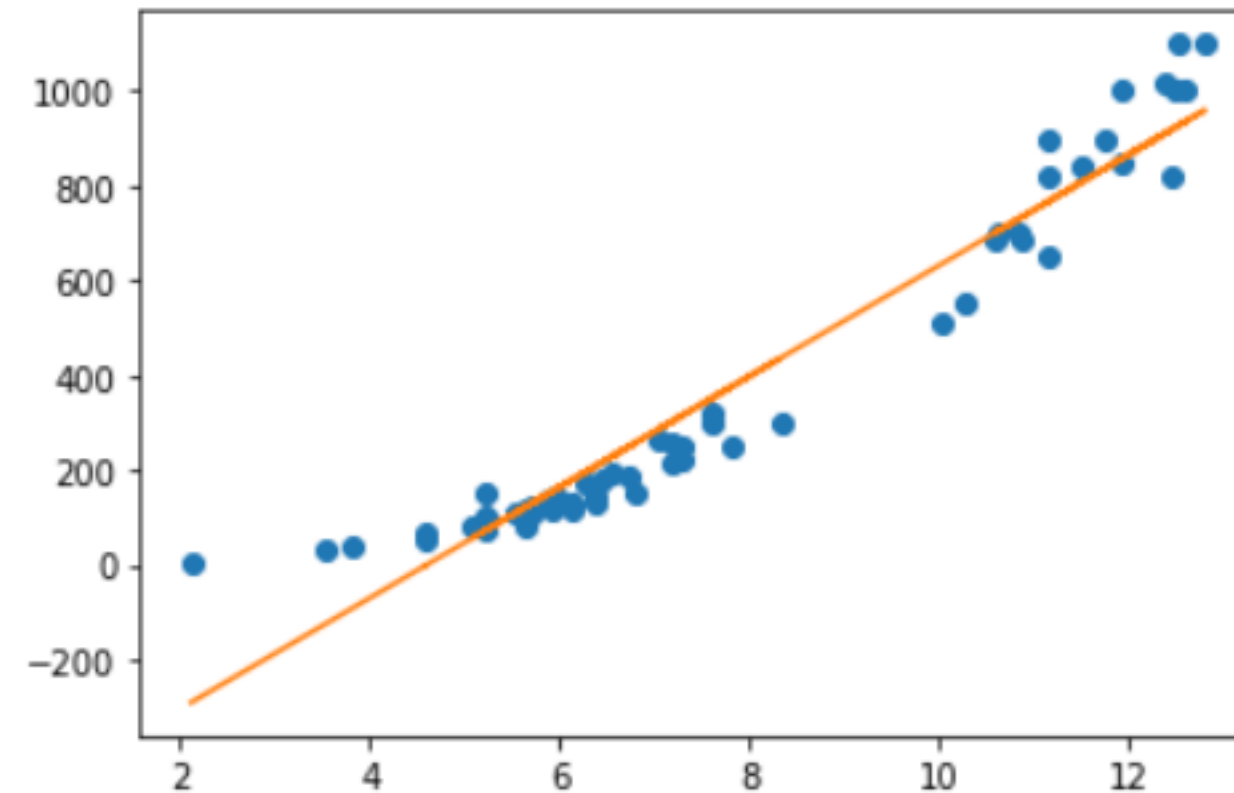
Ecuación del modelo:

$$\hat{y} = -537.32 + 116.96x_4$$

GRÁFICA DEL MODELO

```
plt.plot(x4, y, 'o')  
m, b = np.polyfit(x4, y, 1)  
plt.plot(x4, m*x4 + b)
```

[<matplotlib.lines.Line2D at 0x7fcd656fc410>]



REGRESIÓN POLINOMIAL



```
[153] x4_2=perch["Height"]**2
      d2 = {"x4":x4, "x4_2":x4_2}
      X_2=pd.DataFrame(d2)
      mod3=build_model(X_2,y)
```

OLS Regression Results

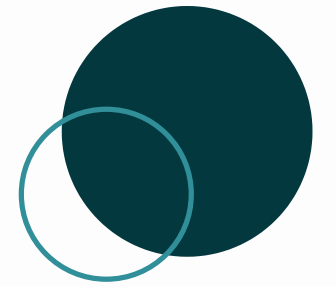
=====						
Dep. Variable:	Weight	R-squared:	0.981			
Model:	OLS	Adj. R-squared:	0.981			
Method:	Least Squares	F-statistic:	1395.			
Date:	Wed, 08 Sep 2021	Prob (F-statistic):	1.47e-46			
Time:	21:16:28	Log-Likelihood:	-295.11			
No. Observations:	56	AIC:	596.2			
Df Residuals:	53	BIC:	602.3			
Df Model:	2					
Covariance Type:	nonrobust					
=====						
	coef	std err	t	P> t	[0.025	0.975]

const	92.4902	59.731	1.548	0.127	-27.314	212.295
x4	-54.1540	15.557	-3.481	0.001	-85.357	-22.952
x4_2	10.2295	0.920	11.118	0.000	8.384	12.075
=====						
Omnibus:	13.550	Durbin-Watson:	1.563			
Prob(Omnibus):	0.001	Jarque-Bera (JB):	34.089			
Skew:	-0.510	Prob(JB):	3.96e-08			
Kurtosis:	6.684	Cond. No.	815.			
=====						

Ecuación del modelo:

$$\hat{y} = 92.49 - 54.15x_4 + 10.22x_4^2$$

REGRESIÓN POLINOMIAL



```
[153] x4_2=perch["Height"]**2
      d2 = {"x4":x4, "x4_2":x4_2}
      X_2=pd.DataFrame(d2)
      mod3=build_model(X_2,y)
```

OLS Regression Results

Dep. Variable:	Weight	R-squared:	0.981			
Model:	OLS	Adj. R-squared:	0.981			
Method:	Least Squares	F-statistic:	1395.			
Date:	Wed, 08 Sep 2021	Prob (F-statistic):	1.47e-46			
Time:	21:16:28	Log-Likelihood:	-295.11			
No. Observations:	56	AIC:	596.2			
Df Residuals:	53	BIC:	602.3			
Df Model:	2					
Covariance Type:	nonrobust					
=====						
	coef	std err	t	P> t	[0.025	0.975]

const	92.4902	59.731	1.548	0.127	-27.314	212.295
x4	-54.1540	15.557	-3.481	0.001	-85.357	-22.952
x4_2	10.2295	0.920	11.118	0.000	8.384	12.075
=====						
Omnibus:	13.550	Durbin-Watson:	1.563			
Prob(Omnibus):	0.001	Jarque-Bera (JB):	34.089			
Skew:	-0.510	Prob(JB):	3.96e-08			
Kurtosis:	6.684	Cond. No.	815.			
=====						

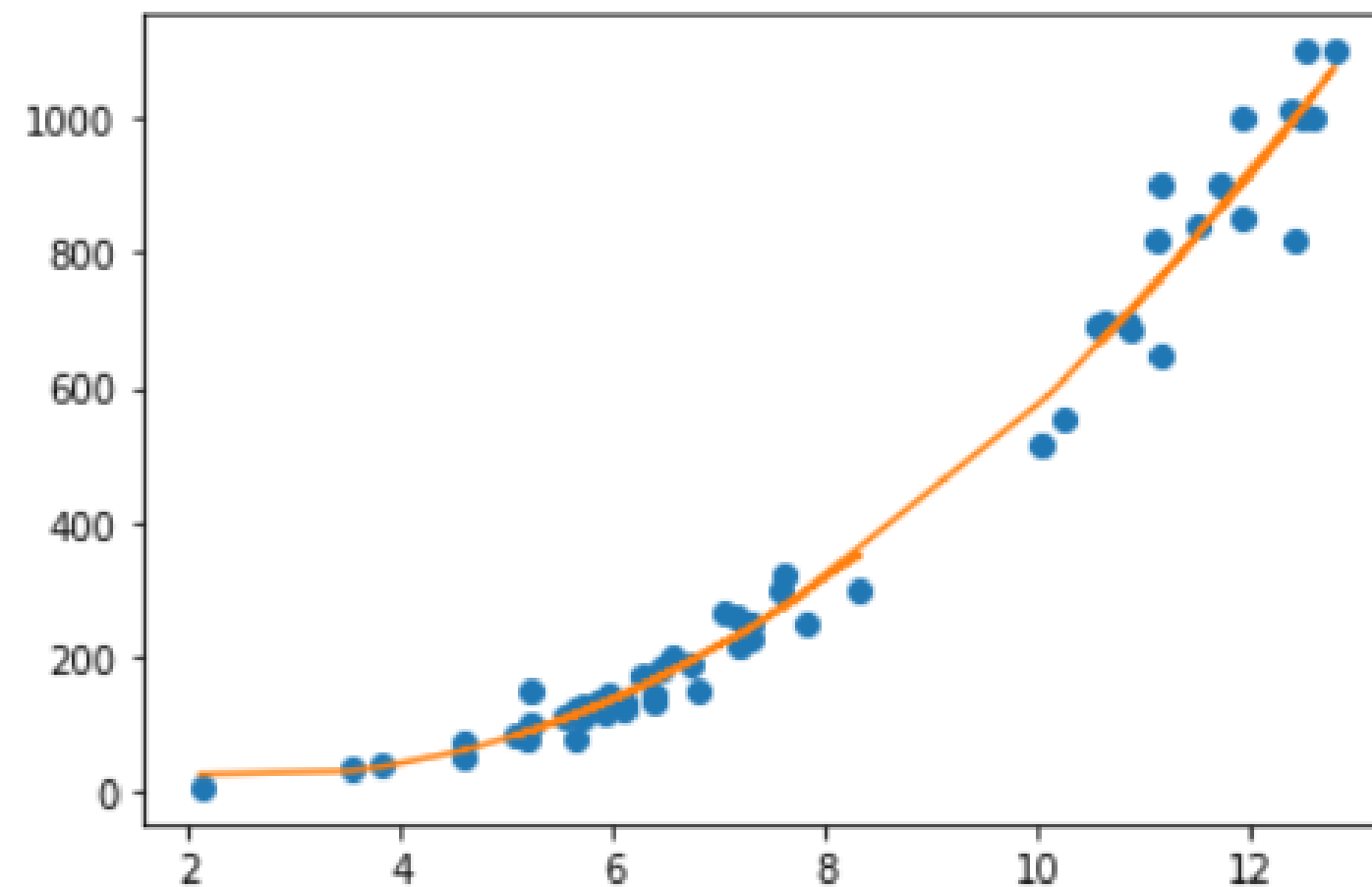
Ecuación del modelo:

$$\hat{y} = 92.49 - 54.15x_4 + 10.22x_4^2$$

GRÁFICA DEL MODELO

```
plt.plot(x4, y, 'o')  
a,b,c = np.polyfit(x4, y, 2)  
plt.plot(x4, a*x4*x4 + b*x4 + c)
```

[<matplotlib.lines.Line2D at 0x7fcd657d4a90>]



CÓMO HACER UNA ESTIMACIÓN



Haciendo uso de la ecuación del modelo, se sustituye la x .
Estimar el precio de una percha con altura de 9.5 cm

$$\hat{y} = 92.49 - 54.15x_4 + 10.22x_4^2$$

$$\hat{y} = 92.49 - 54.15(9.5) + 10.22(9.5)^2$$

$$\hat{y} = 501.2379 \text{ g}$$

```
X_2=sm.add_constant(X_2)
y_pred = sm.OLS(y,X_2).fit().predict([1,9.5,9.5**2])
y_pred
```

```
array([501.23791418])
```

PREGUNTAS

- ¿EN LA REGRESIÓN LINEAL SIMPLE, QUÉ DETERMINA EL COEFICIENTE B_1 ?
- ¿QUÉ NOMBRE SE LE ASIGNA A LA(S) VARIABLE(S) INDEPENDIENTE(S) (X'S) EN LOS MODELOS DE REGRESIÓN?
- ¿CUÁNTOS COEFICIENTES B HAY EN UN MODELO DE REGRESIÓN CON TRES VARIABLES PREDICTORAS?
- ¿QUÉ SIGNIFICA UN VIF ALTO?
- ¿ES LA RELACIÓN DE DEPENDENCIA LINEAL FUERTE ENTRE MÁS DE DOS VARIABLES EN REGRESIÓN MULTIPLE?

RESPUESTAS

- **DETERMINA LA PENDIENTE DE LA RECTA EN EL MODELO**
- **VARIABLE(S) PREDICTORA(S)**
- **CUATRO, B_0 , B_1 , B_2 Y B_3**
- **QUE UNA DE LAS VARIABLES REGRESORAS ESTÁ RELACIONADA LINEALMENTE CON LAS DEMÁS**
- **MULTICOLINEALIDAD**

BIBLIOGRAFÍA

- *Apuntes Métodos Estadísticos con MET*. Alejandra Cerda. Sem Febrero-Junio 2021
- Goyal, S. (2019). *Car Price Prediction (Linear Regression - RFE)*. Obtenido de Kaggle: <https://www.kaggle.com/goyalshalini93/car-price-prediction-linear-regression-rfe>
- *How to plot a linear regression line on a scatter plot in Python*. (s.f.). Obtenido de Kite: <https://www.kite.com/python/answers/how-to-plot-a-linear-regression-line-on-a-scatter-plot-in-python>
- *pandas.DataFrame.reset_index*. (s.f.). Obtenido de pandas: https://pandas.pydata.org/pandas-docs/dev/reference/api/pandas.DataFrame.reset_index.html
- Pyae, A. (2019, junio 13). *Fish Market*. Retrieved from Kaggle: <https://www.kaggle.com/aungpyaeap/fish-market>