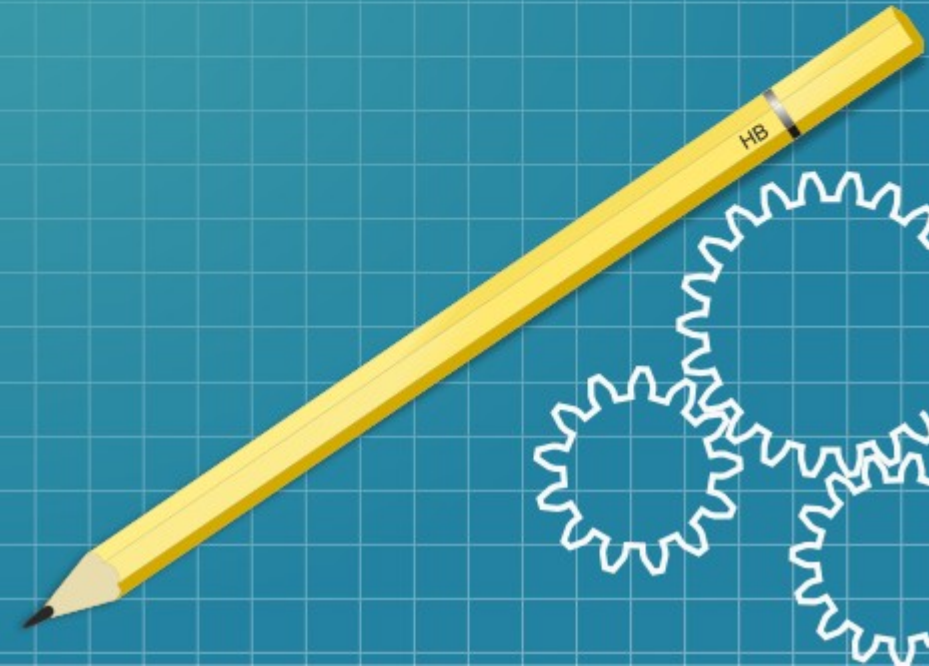


CORRELACIÓN Y REGRESIONES



MENÚ



*CONCEPTO DE CORRELACIÓN

*REGRESIONES:

@ LINEAL

@ CUADRÁTICA

@ POLINOMIAL

@ CASOS RELACIONADOS: LOGARÍTMICA Y EXPONENCIAL

@ DOS O MÁS VARIABLES

@ REGRESIONES CON ESTEROIDES (CON DESCENSO DEL GRADIENTE)

@ CUANDO CONOCEMOS LA RELACIÓN FUNCIONAL...

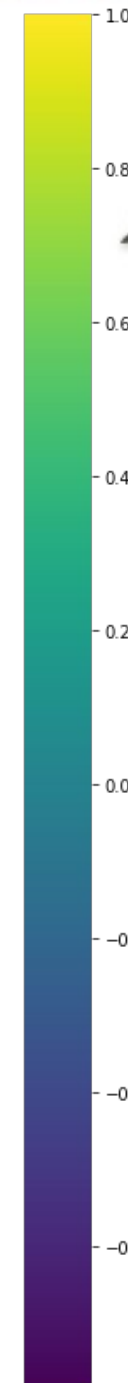
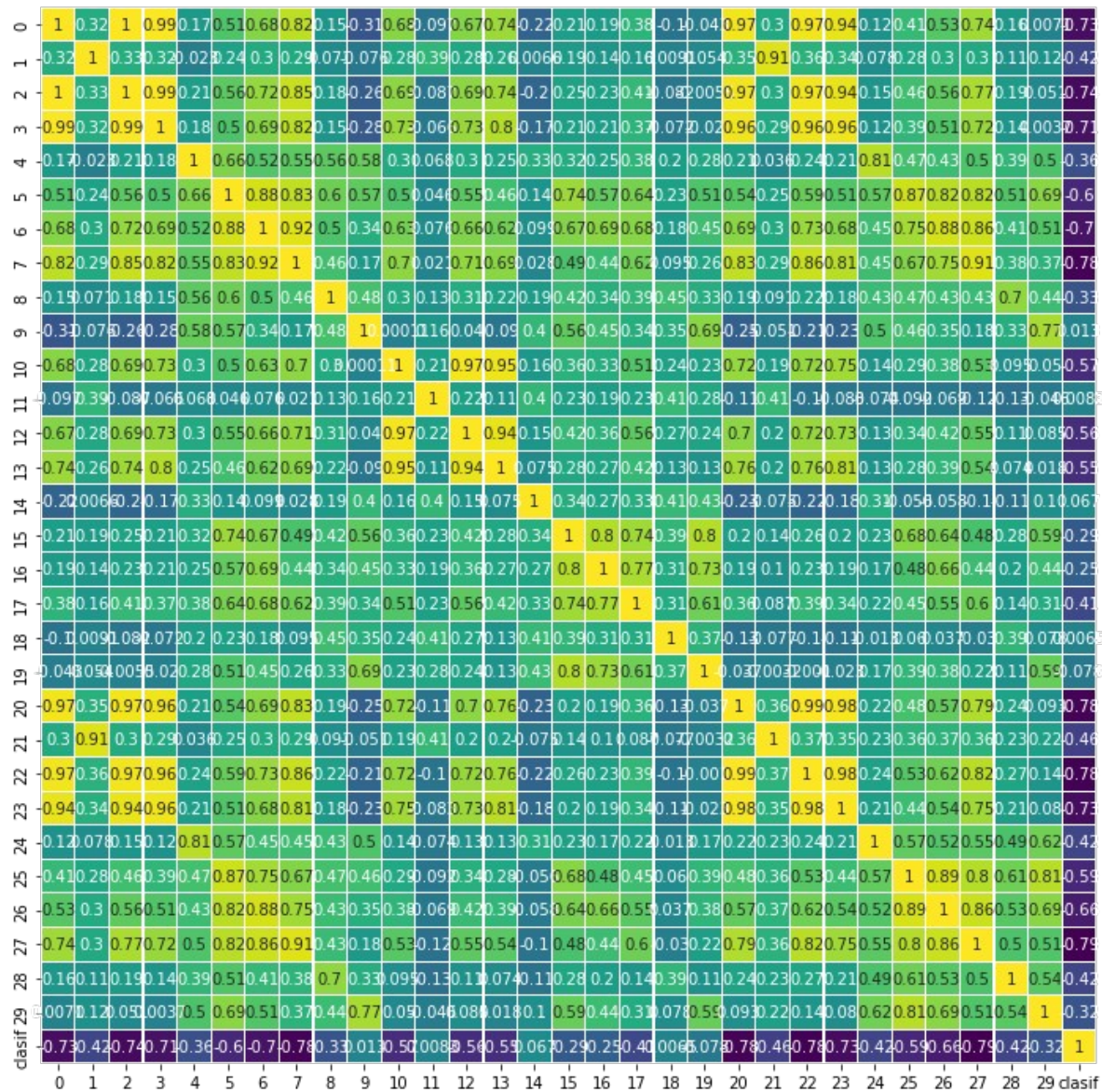
@ REGRESIÓN LINEAL BAYESIANA (la próxima clase)

CORRELACIÓN

- Dos o más variables están correlacionadas, cuando un cambio en una de ellas se refleja en cambios en otra
- No confundir con Relación de Causalidad, ya que la Estadística no puede asegurar la misma
- Hay varios coeficientes que tratan de medir la correlación, el más conocido es el de Pearson

$$r = \cos(\alpha) = \frac{\sum_{i=1}^N (x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^N (y_i - \bar{y})^2}}$$

Correlación de Pearson de las características



REGRESIONES



- Supongamos que hay una relación lineal entre dos variables

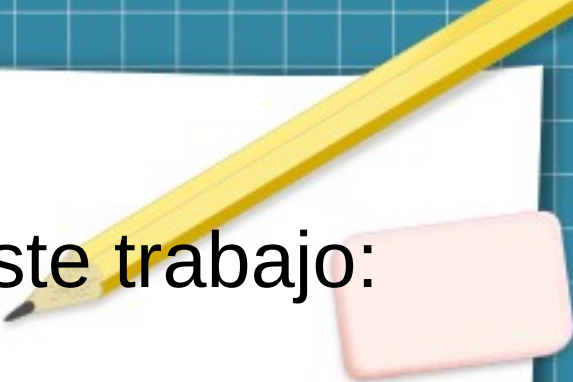
$$Y = a * X + b$$


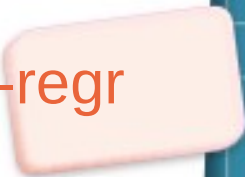
- X es la variable *independiente* o *explicativa* o *regresora*
- Y es la variable *dependiente* (de X), también *explicada* o *regresando*
- A partir de un conjunto de datos (x_i, y_i) quiero conocer los parámetros **a** y **b**

- En la vida real, esta relación no va a ser perfecta, entonces:

$$y_i = a * x_i + b + r_i$$

- donde r_i es el residuo, es decir, cuánto se aleja el valor medido del valor 'ideal'
- Lo deseable es que todos los r_i sean lo más pequeños posibles
- A partir de cada r_i definimos el **Error Cuadrático Medio** como la suma de los cuadrados de los residuos porque, como a veces serán positivos o negativos, se elevan al cuadrado para compensar esa variación y que no se anule el ECM
- La idea es encontrar los parámetros a y b que relacionan a ambas variables.
- Cómo lo hacemos? Como queremos que ECM sea lo menor posible, buscamos su mínimo
- Este método se llama **“MÍNIMOS CUADRADOS”** y se usa un montón!

- 
- Tenemos varias bibliotecas para hacer este trabajo: Sklearn y Scipy son las principales
 - https://scikit-learn.org/stable/modules/linear_model.html#generalized-linear-models
 - `class sklearn.linear_model.LinearRegression(*, fit_intercept=True, copy_X=True, n_jobs=None, positive=False)`
 - <https://www.statsmodels.org/stable/examples/index.html>
 - `class statsmodels.regression.linear_model.OLS(endog, exog=None, missing='none', hasconst=None, **kwargs)`

- 
- 
- Hay varios tipos de Regresiones lineales:
 - *Regresión Ridge (con penalización L2)
 - https://scikit-learn.org/stable/modules/linear_model.html#ridge-regression-and-classification
 - *Regresión Lasso (con penalización L1)
 - https://scikit-learn.org/stable/modules/linear_model.html#lasso
 - *Elasticnet (penalización L1 y L2)
 - https://scikit-learn.org/stable/modules/linear_model.html#elastic-net
 - *SGD (descenso de gradiente)
 - https://scikit-learn.org/stable/modules/linear_model.html#stochastic-gradient-descent-sgd
 - *Regresiones Robustas
 - https://scikit-learn.org/stable/modules/linear_model.html#robustness-regression-outliers-and-modeling-errors
 - <https://www.statsmodels.org/stable/examples/index.html#robust-regression>
 - * y siguen las firmas...

- Y si buscamos resolver una relación polinomial?

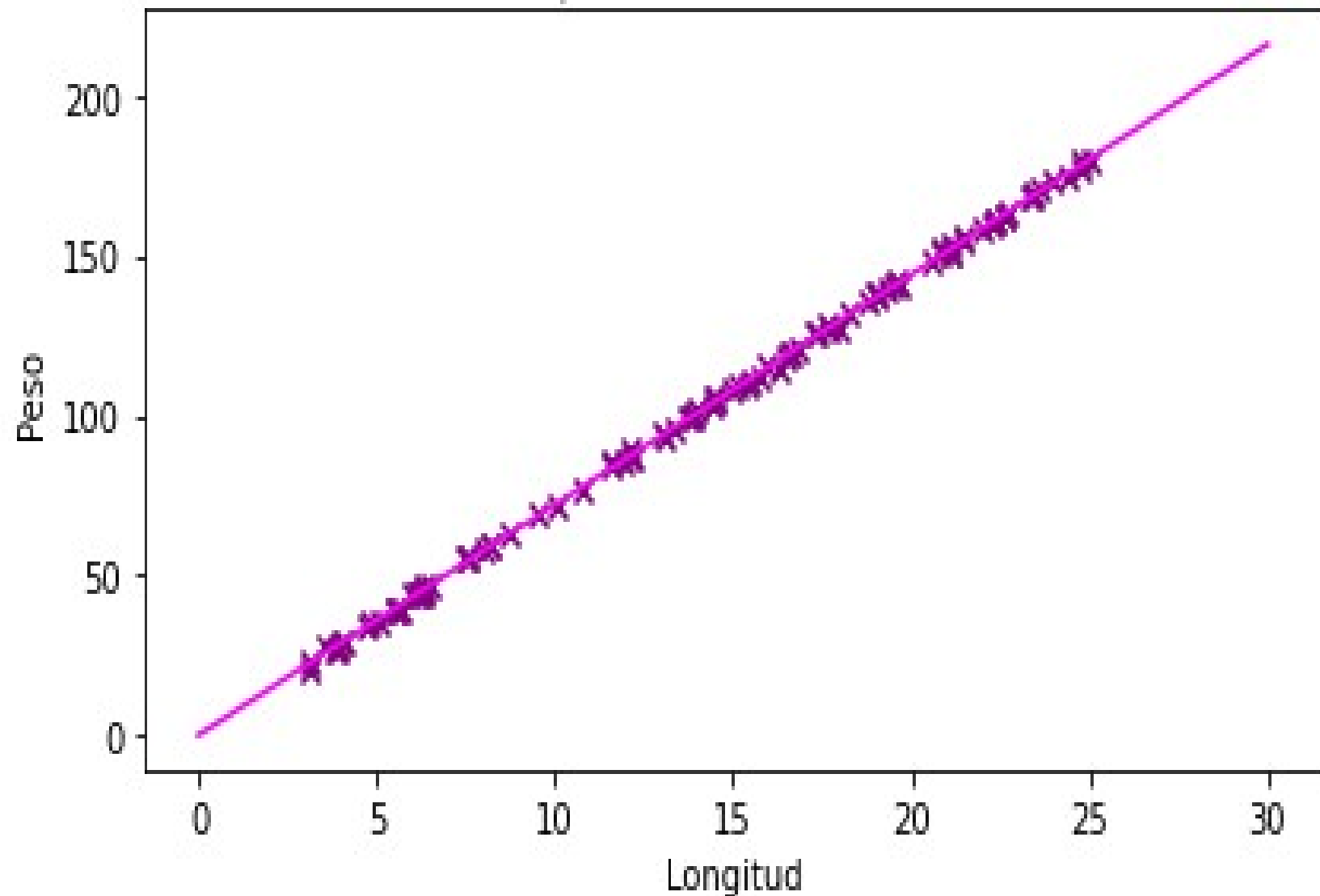
$$Y = a_0 + a_1 * X^{**1} + a_2 * X^{**2} + a_3 * X^{**3} + \dots$$

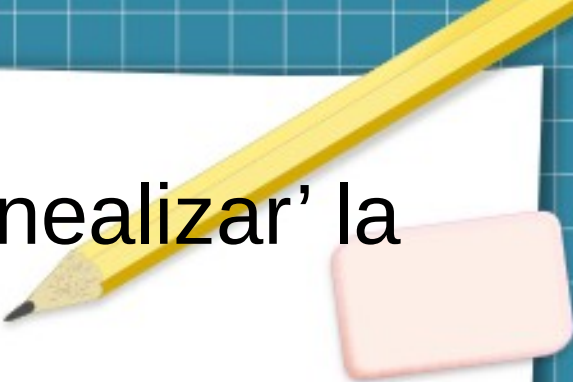
- Podemos aplicar regresión lineal todavía... sobre los coeficientes de la regresión!
- Y también Sklearn lo implementa con PolynomialFeatures
- https://scikit-learn.org/stable/modules/linear_model.html#polynomial-regression-extending-linear-models-with-basis-functions
- ```
class
sklearn.preprocessing.PolynomialFeatures(degree=2,
*, interaction_only=False, include_bias=True, order='C')
```
- Pero después hay que aplicar una regresión lineal...

Ajuste lineal: Peso vs. Longitud

$$\text{Peso} = 7.202 * \text{Longitud}$$

ECM = 0.929 | Coef de Correlación = 0.99979




- 
- Para ciertos problemas podemos 'linealizar' la regresión...
  - $Y = a \cdot \exp(b \cdot X) \rightarrow \ln(Y) = \ln(a) + b \cdot X$
  - Cuando conocemos la relación funcional entre las variables,  $Y = f(X)$ , podemos usar un método de Scipy llamado `curve_fit`: dada la relación funcional calcula los coeficientes

<https://docs.scipy.org/doc/scipy/tutorial/optimize.html>

- [https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve\\_fit.html](https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve_fit.html)



- 
- También podemos aplicar Regresiones No Lineales
  - $Y = a_0 * X^{**} a_1$
  - Y todo esto se puede extender a más de una variable...

### EJEMPLO EN COLAB:

- [https://colab.research.google.com/drive/1EVhJoIM4Ham79Q5Rba5\\_Az4T8C2orO-a](https://colab.research.google.com/drive/1EVhJoIM4Ham79Q5Rba5_Az4T8C2orO-a)



DUDAS O  
CONSULTAS????



GRACIAS POR SU  
ATENCIÓN!!!