



ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

EDUCACIÓN
PROFESIONAL

Diplomado en Programación y Aplicaciones de Python

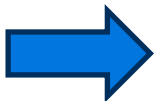
Aplicaciones en Ciencia de Datos e
Inteligencia Artificial

Francisco Pérez Galarce





Fechas de evaluaciones



Fecha	Actividad/Evaluación
29-10-24	Introducción al aprendizaje de máquina: exploración y procesamiento de datos con Python Actividad 1 (No evaluada)
05-11-24	Aprendizaje supervisado con Python : regresiones Actividad 2 (Evaluada)
12-11-24	Aprendizaje supervisado con Python : naive Bayes y métricas de evaluación Control 1
19-11-24	Aprendizaje supervisado con Python : decision tree, random forest Mini Proyecto 1
26-11-24	Aprendizaje supervisado con Python : KNN, SVM Actividad 3 (No Evaluada)
03-12-24	Aprendizaje no supervisado con Python: K-means y reducción de dimensionalidad Actividad 4 (Evaluada) – Control 2
10-12-24	Redes Neuronales I Mini Proyecto 2
17-12-24	Redes Neuronales II Prueba Final



Objetivos

- Identificar modelos de regresión tales como: regresión lineal simple, regresión polinomial, regresión Lasso y regresión Ridge.
- Aplicar (implementar) a través de Python los modelos vistos en clases.



ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

EDUCACIÓN
PROFESIONAL

Aprendizaje supervisado

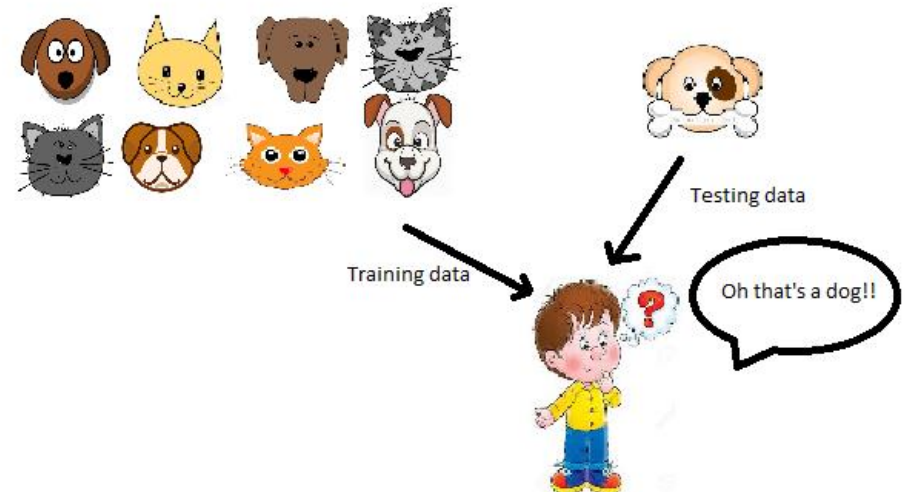
Aprendizaje supervisado

Aquellos algoritmos y modelos que aprenden desde una base de datos con pares del tipo **input/output**.

$$\mathcal{D} = \left\{ \overbrace{(x_1, y_1)}^{\text{Input}} \overbrace{, \dots, (x_N, y_N)}^{\text{Output}} \right\}$$

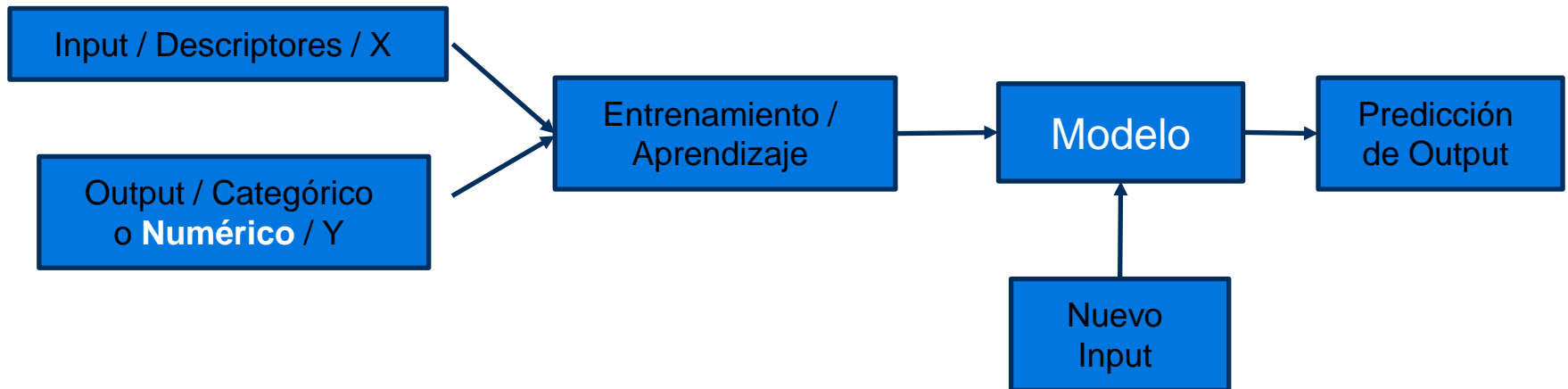
Input: edad, profesión, género, nivel de deuda, etc.

Output: comprará o no un nuevo producto



Aprendizaje supervisado

Aprender una función (Clasificador/**Regresor**) que permita obtener los valores del **output** por medio del **input**.





Algunas aplicaciones

Predicción de demanda

Rendimiento académico

Días de hospitalización

Pronóstico de nivel de
contaminación



Contenidos

- Regresión lineal simple y múltiple
- Regresión polinomial
- Regresiones con penalización



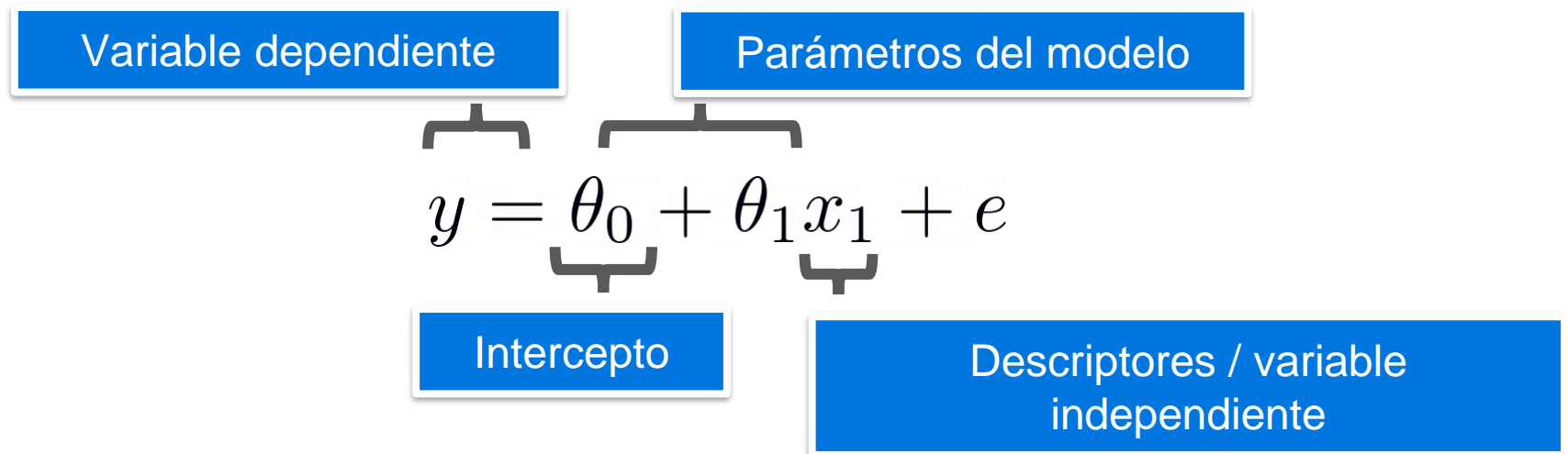
ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

EDUCACIÓN
PROFESIONAL

Regresión lineal simple

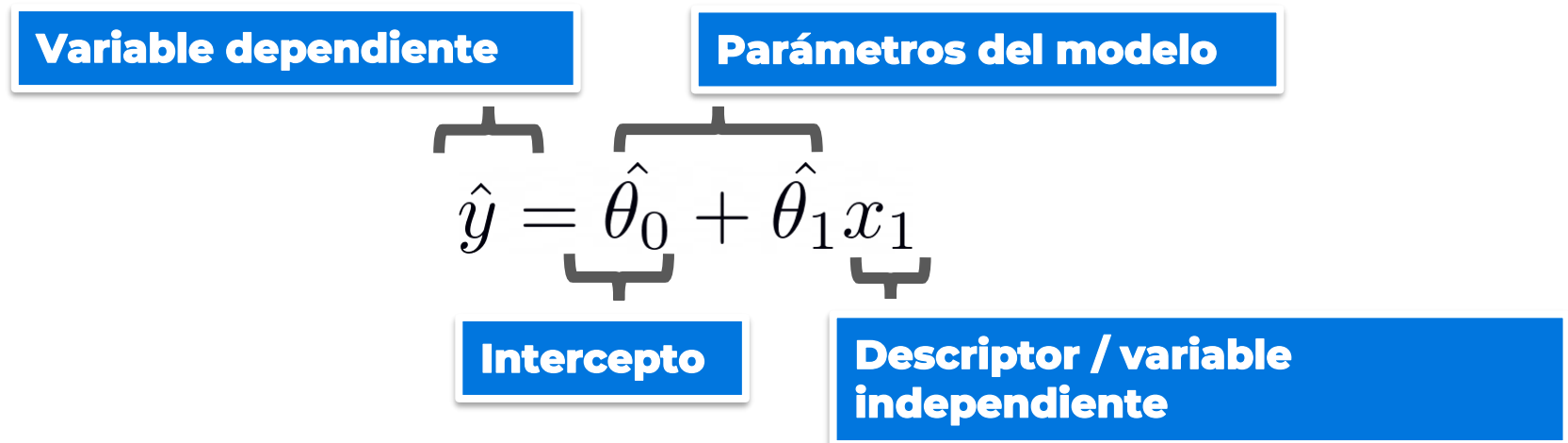


Regresión lineal simple

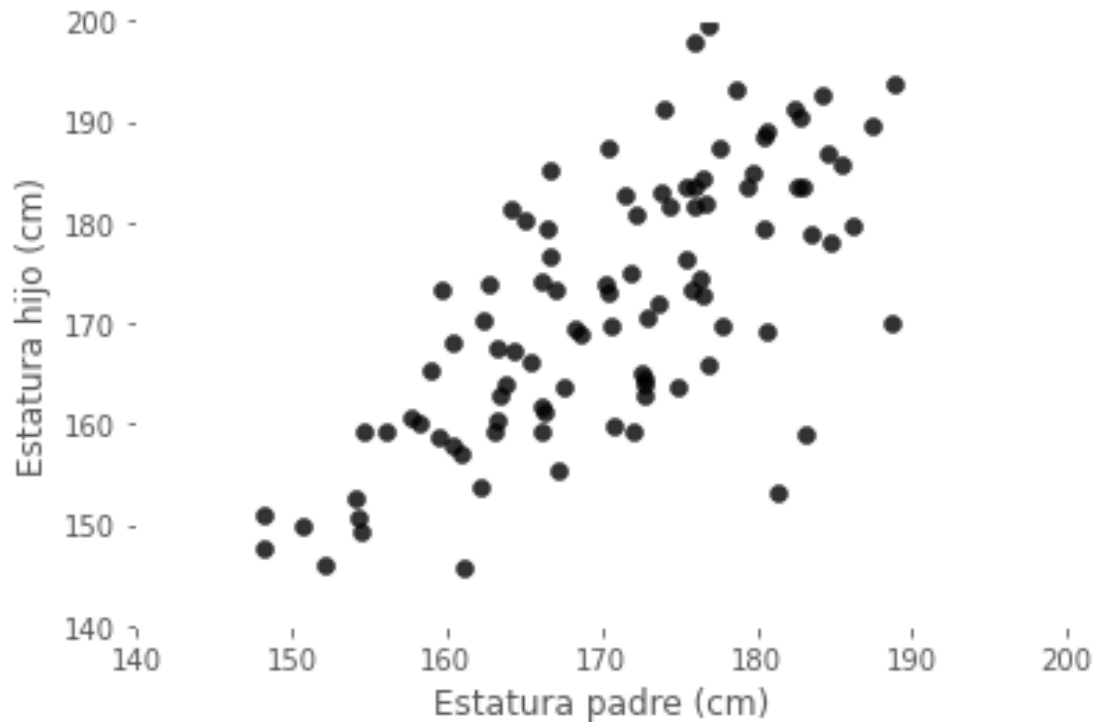




Regresión lineal simple



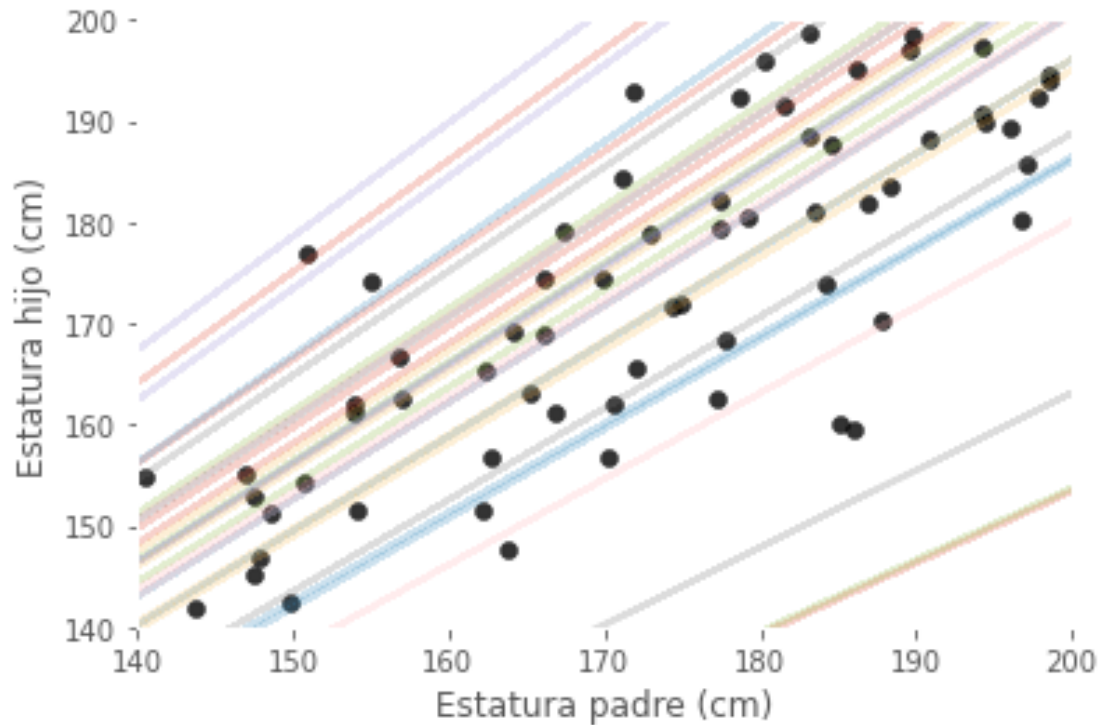
Regresión lineal simple



$$\hat{y} = \underbrace{\hat{\theta}_0 + \hat{\theta}_1 x_1}_{\text{¿Valores óptimos?}}$$

**¿Valores
óptimos?**

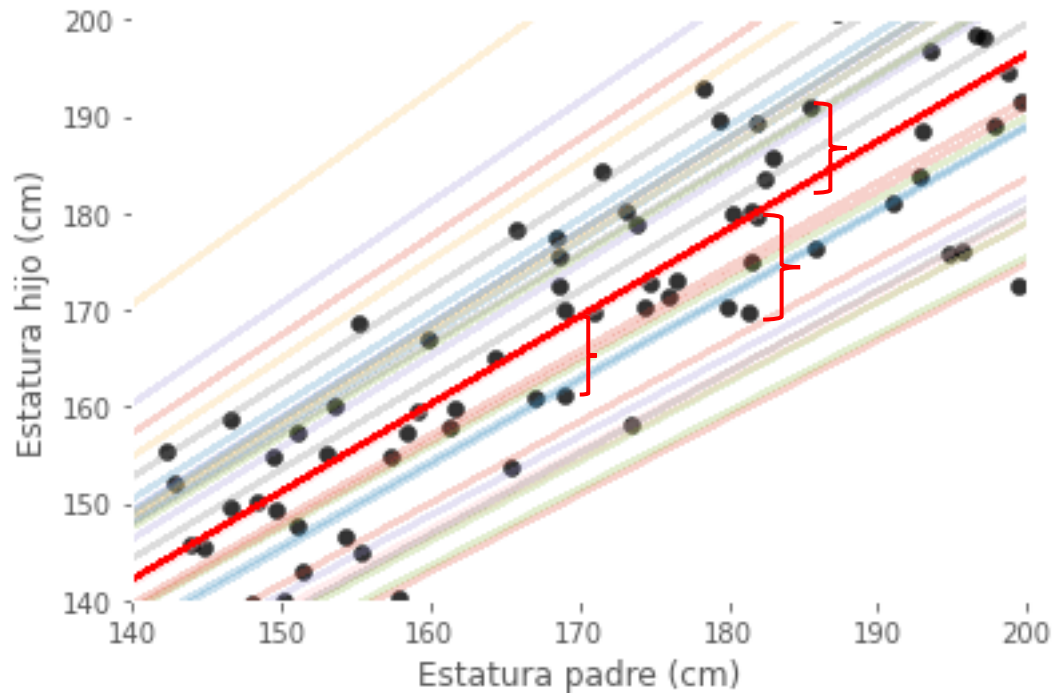
Regresión lineal simple



$$\hat{y} = \underbrace{\hat{\theta}_0 + \hat{\theta}_1 x_1}$$

**¿Valores
óptimos?**

Obtención de los parámetros



$$\frac{1}{m} \sum_{i=1}^m (\theta^T \cdot \mathbf{x}^i - y^i)^2$$

Obtención de los parámetros

Predicción

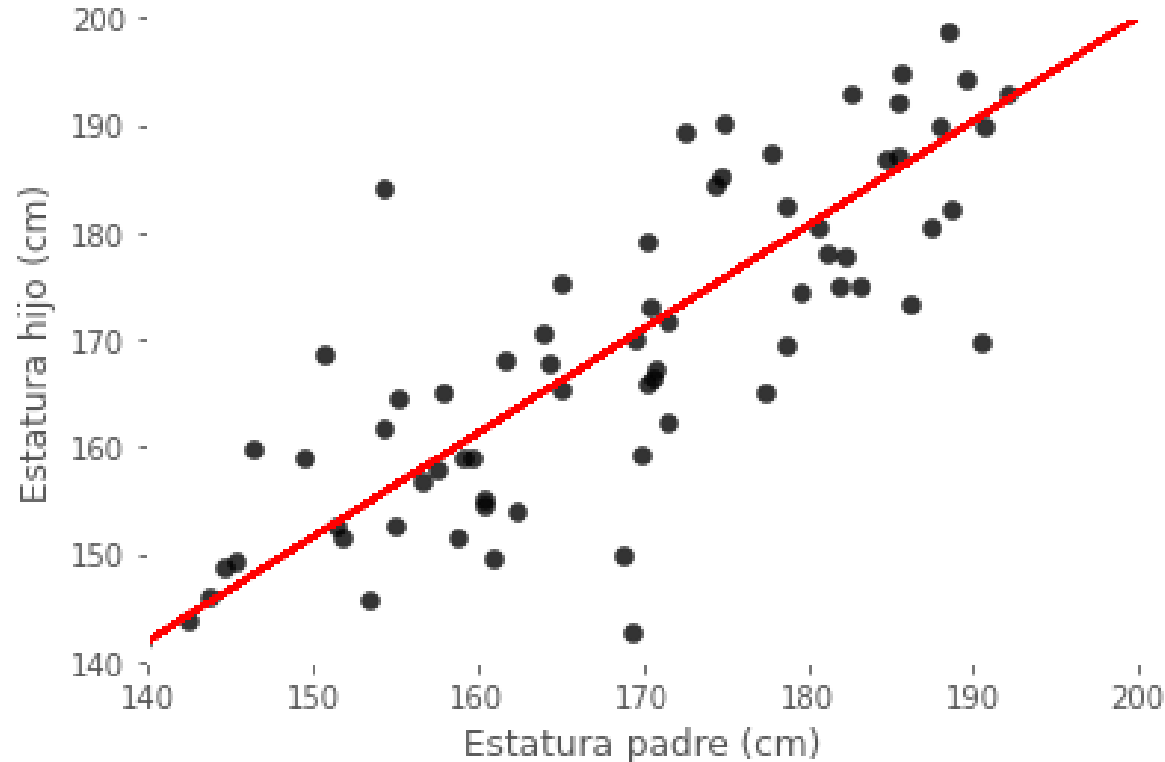
$$\text{MSE}(\mathbf{X}, h_{\theta}) = \frac{1}{m} \sum_{i=1}^m \overbrace{(\theta^T \cdot \mathbf{x}^i - y^i)^2} \quad \left. \vphantom{\sum_{i=1}^m} \right\} \text{Mínimo MSE}$$

Solución cerrada

$$\hat{\theta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

Obtención de los parámetros

$$\hat{\theta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$





Regresión lineal múltiple

$$\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x_1 + \dots + \hat{\theta}_i x_i + \dots + \hat{\theta}_n x_n$$

$$\hat{y} = \hat{\theta}^T \cdot \mathbf{x} = h_{\hat{\theta}}(\mathbf{x})$$

\hat{y} = predicción

n = número de descriptores

$\hat{\theta}_j$ = j -ésimo parámetro del modelo

(incluyendo el intercepto θ_0 y los pesos que ponderan los descriptores $\theta_1, \theta_2, \dots, \theta_n$)



Supuestos



Relación lineal



No hay multicolinealidad



Distribución normal



Homocedasticidad



ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

EDUCACIÓN
PROFESIONAL

Regresión Polinomial



Regresión Polinomial

$$y = \theta_0 + \overbrace{\theta_1 x_1} + \overbrace{\theta_2 x_1^2} + \dots + \overbrace{\theta_1 x_1^{p-1}} + \overbrace{\theta_1 x_1^p} + e$$

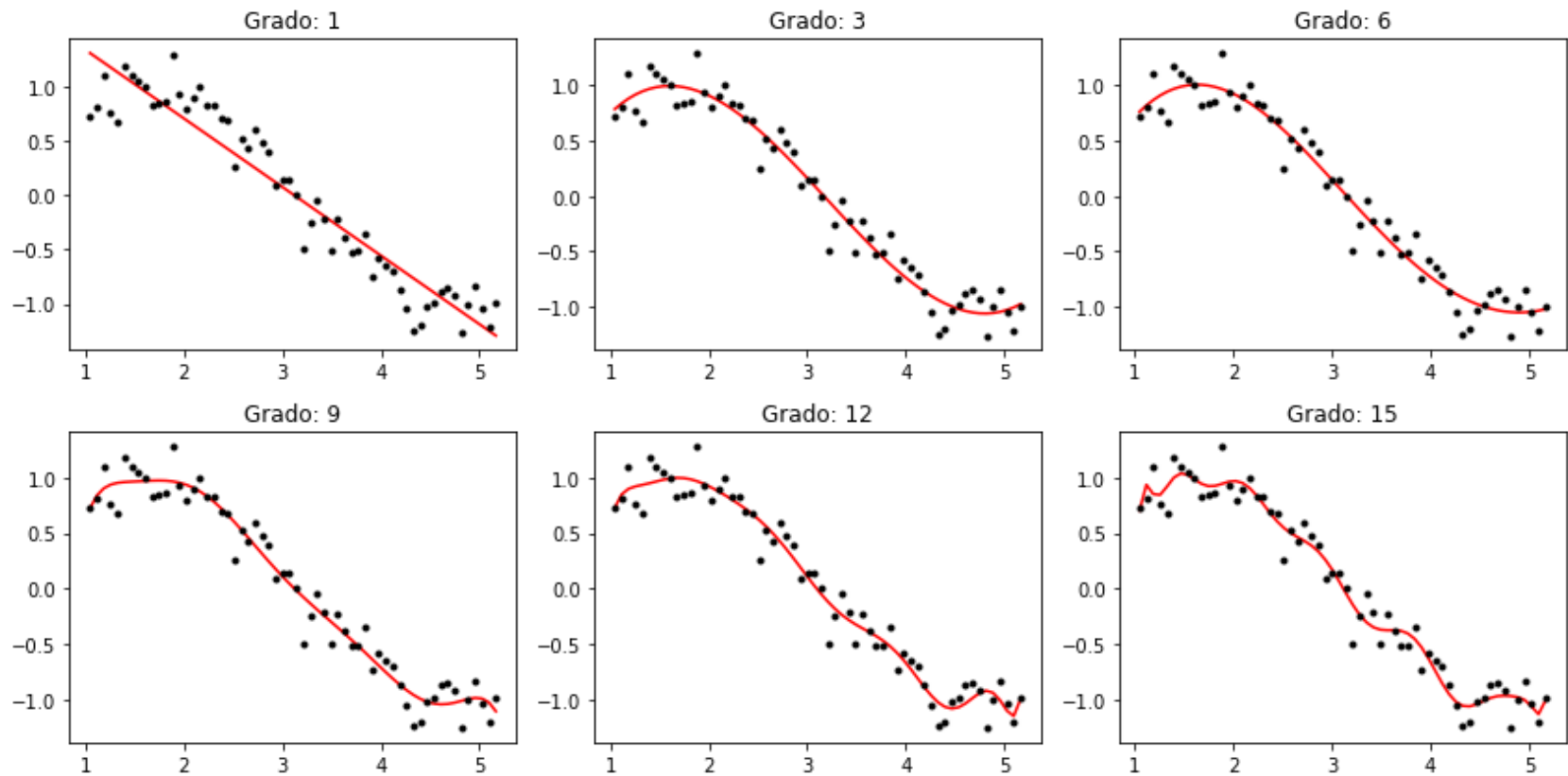


Regresión Polinomial

Modelo sigue siendo lineal en los parámetros, sin embargo, se puede aumentar su expresividad por medio de una transformación (polinomial al input).

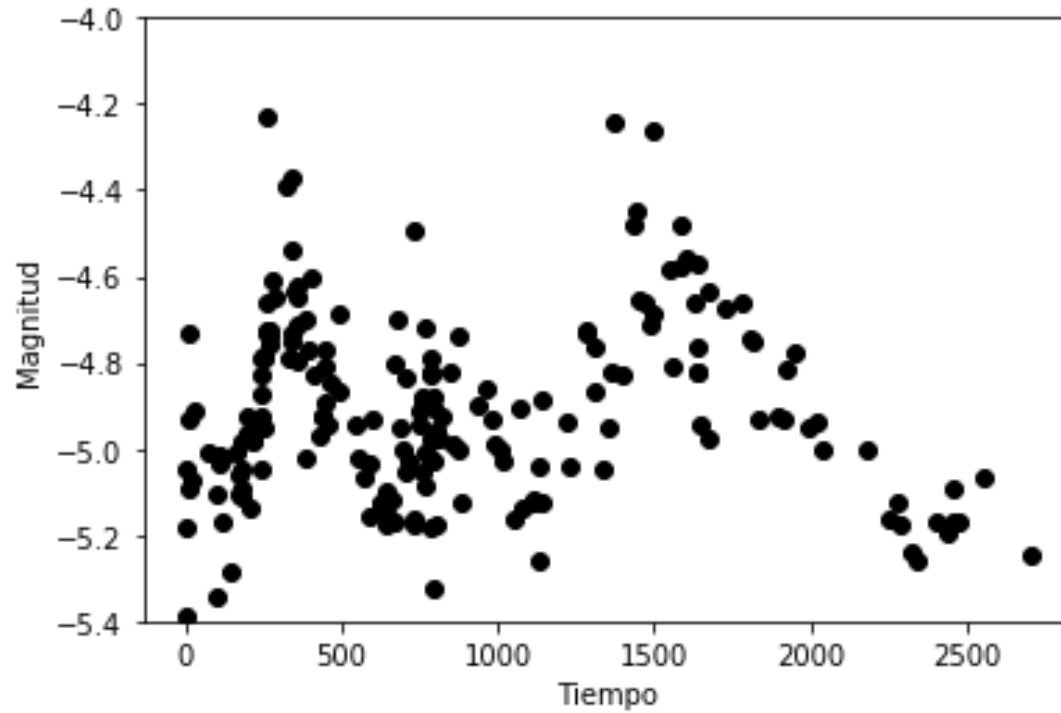
$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \dots + \theta_{p-1} x_1^{p-1} + \theta_p x_1^p + e$$

Regresión Polinomial

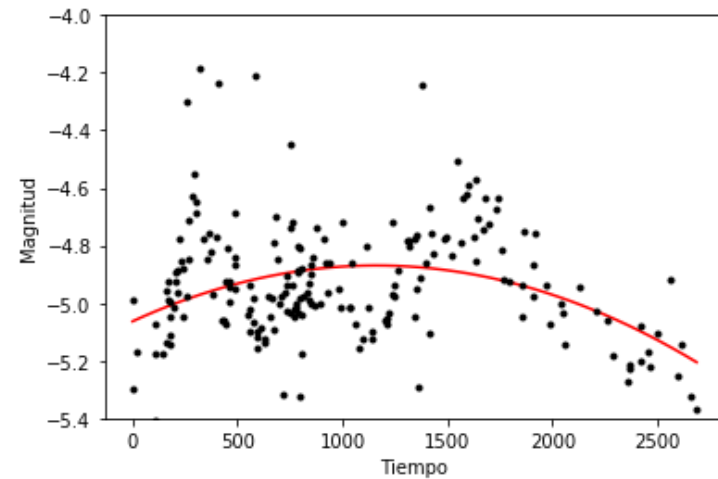
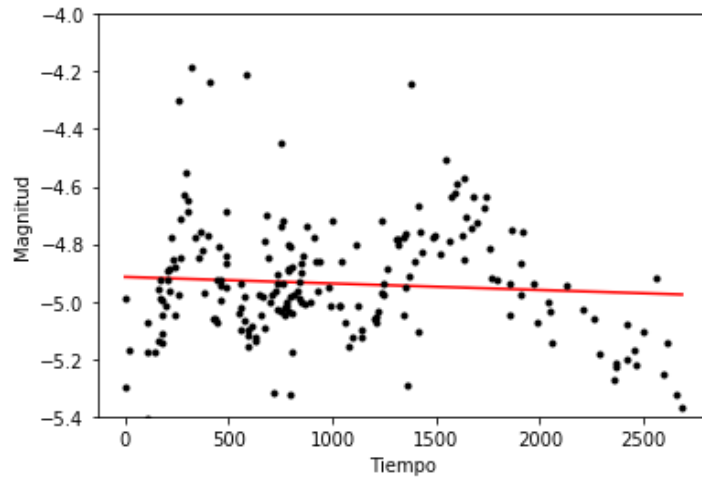




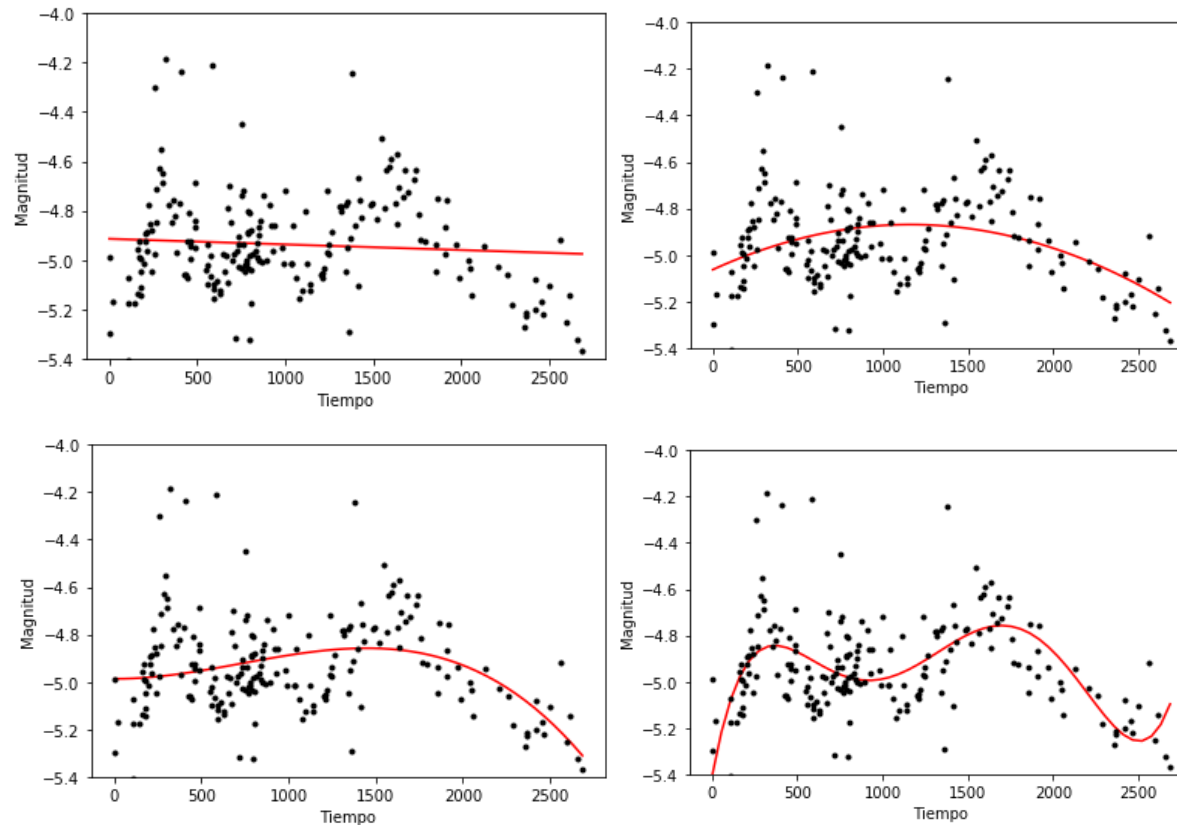
Datos



Polinomio grado 2



Polinomio grado n





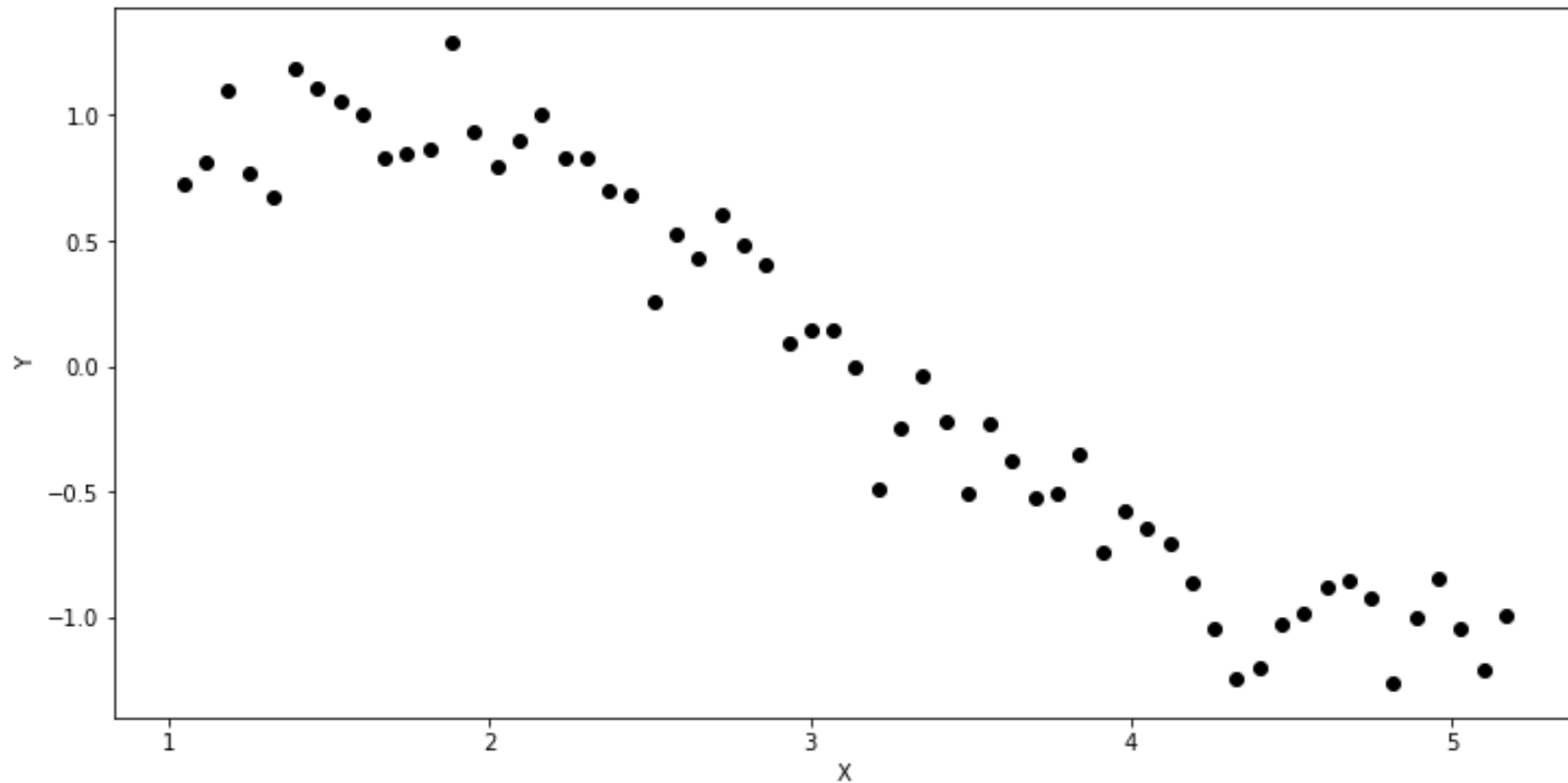
ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

EDUCACIÓN
PROFESIONAL

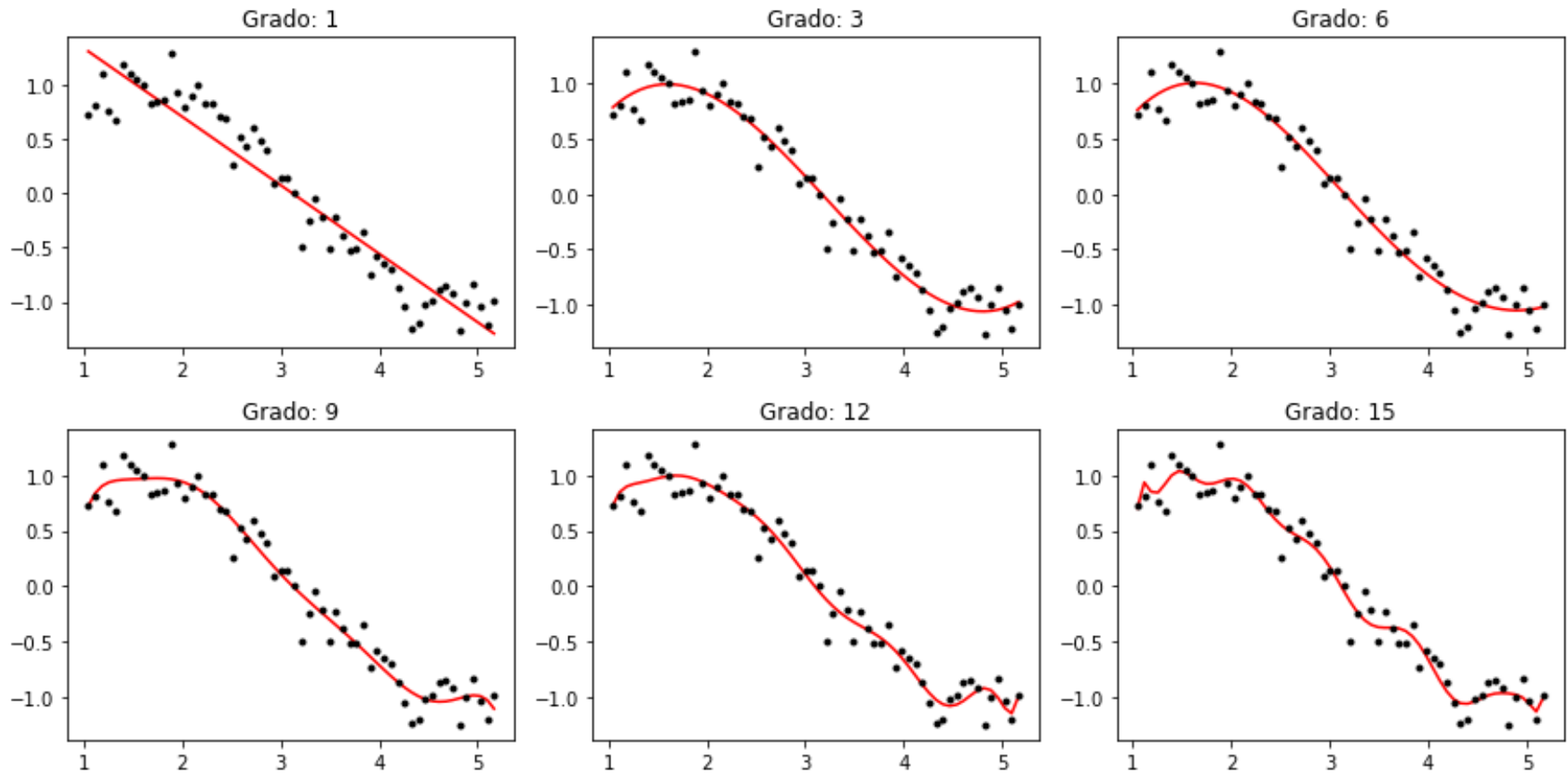
Regresión con penalización



Complejidad en regresiones



Complejidad en regresiones



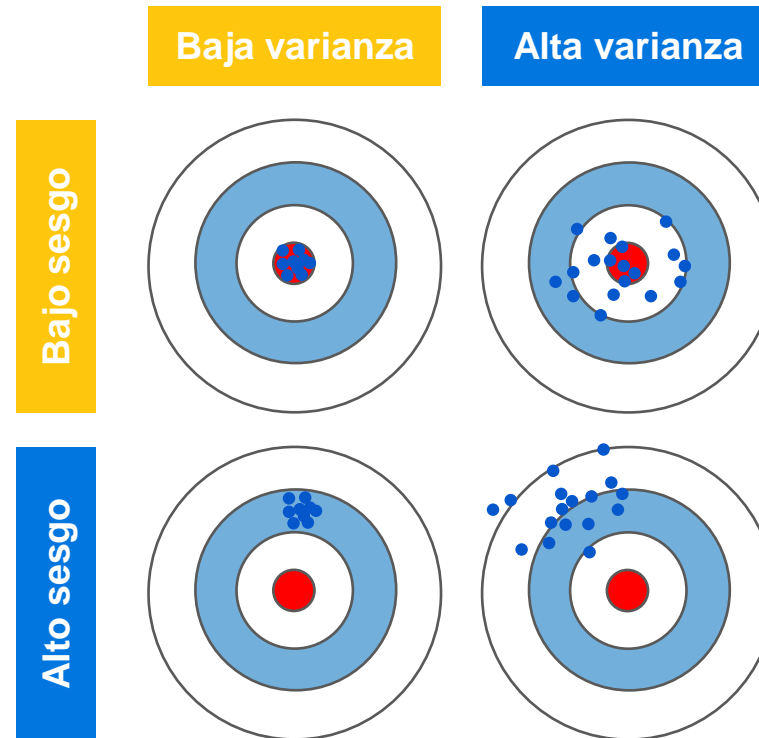


ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

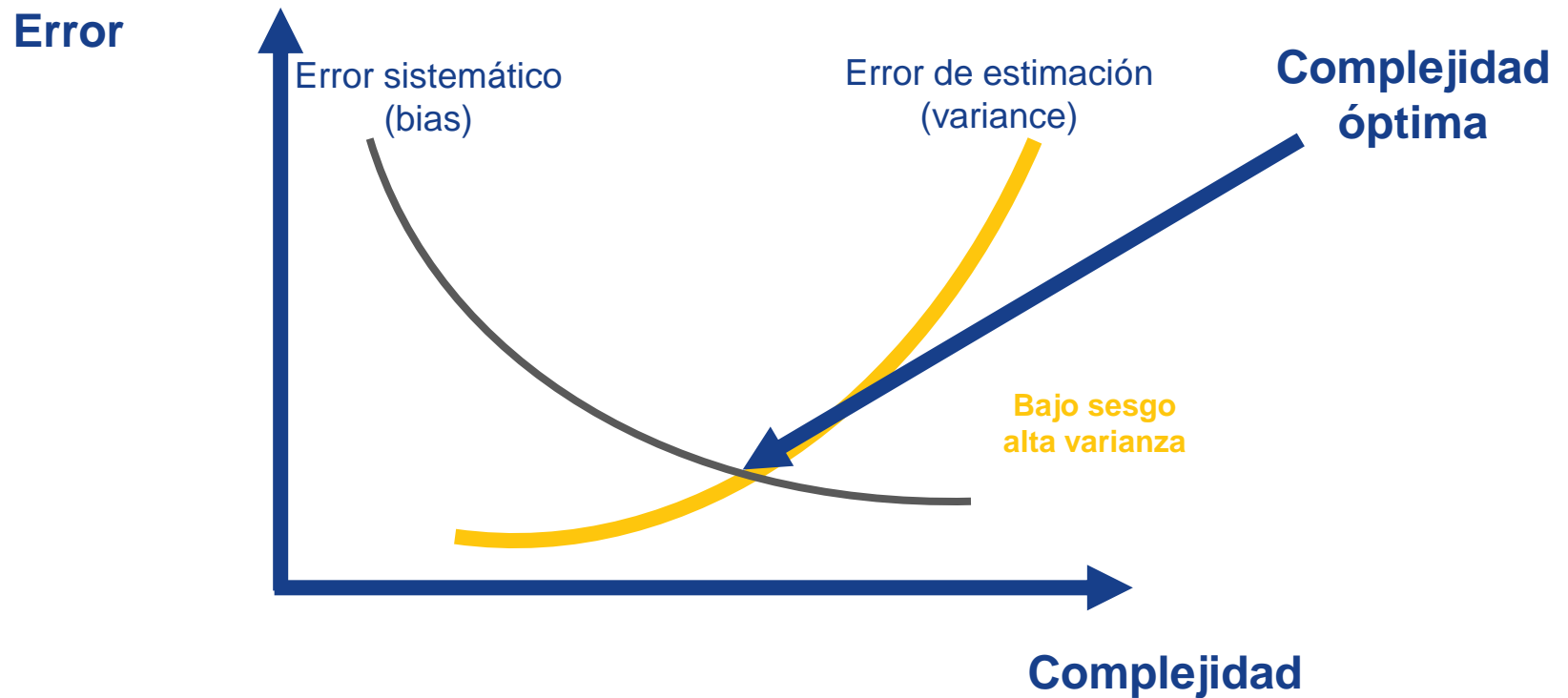
EDUCACIÓN
PROFESIONAL

Sesgo y varianza en modelos

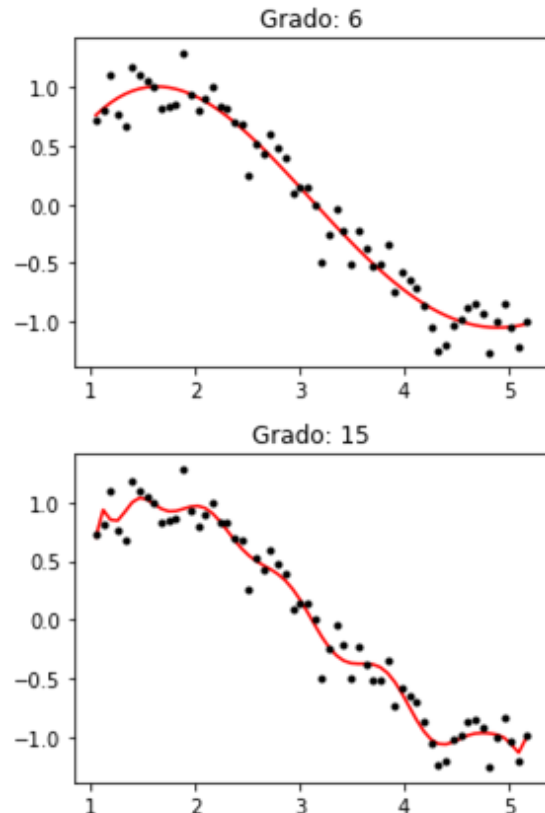
Sesgo-varianza en modelos



Sesgo-varianza en modelos



Complejidad en regresiones



Cuando se ajusta un modelo altamente flexible, se debe tener cuidado con sobreajustarse a los datos de entrenamiento.

Es decir, se debe evitar que el modelo aprenda patrones leves del input, debido a que ellos pueden ser solo producto de ruido y no de los patrones de interés.

Regresiones con penalización

- L1 - Lasso

$$\min \quad \frac{1}{m} \sum_{i=1}^m (\theta^T \cdot \mathbf{x}^i - y^i)^2$$

sujeto a:

$$\sum_{n=1}^N |\theta_n| \leq C$$

- L2 - Ridge

$$\min \quad \frac{1}{m} \sum_{i=1}^m (\theta^T \cdot \mathbf{x}^i - y^i)^2$$

sujeto a:

$$\sum_{n=1}^N \theta_n^2 \leq C$$

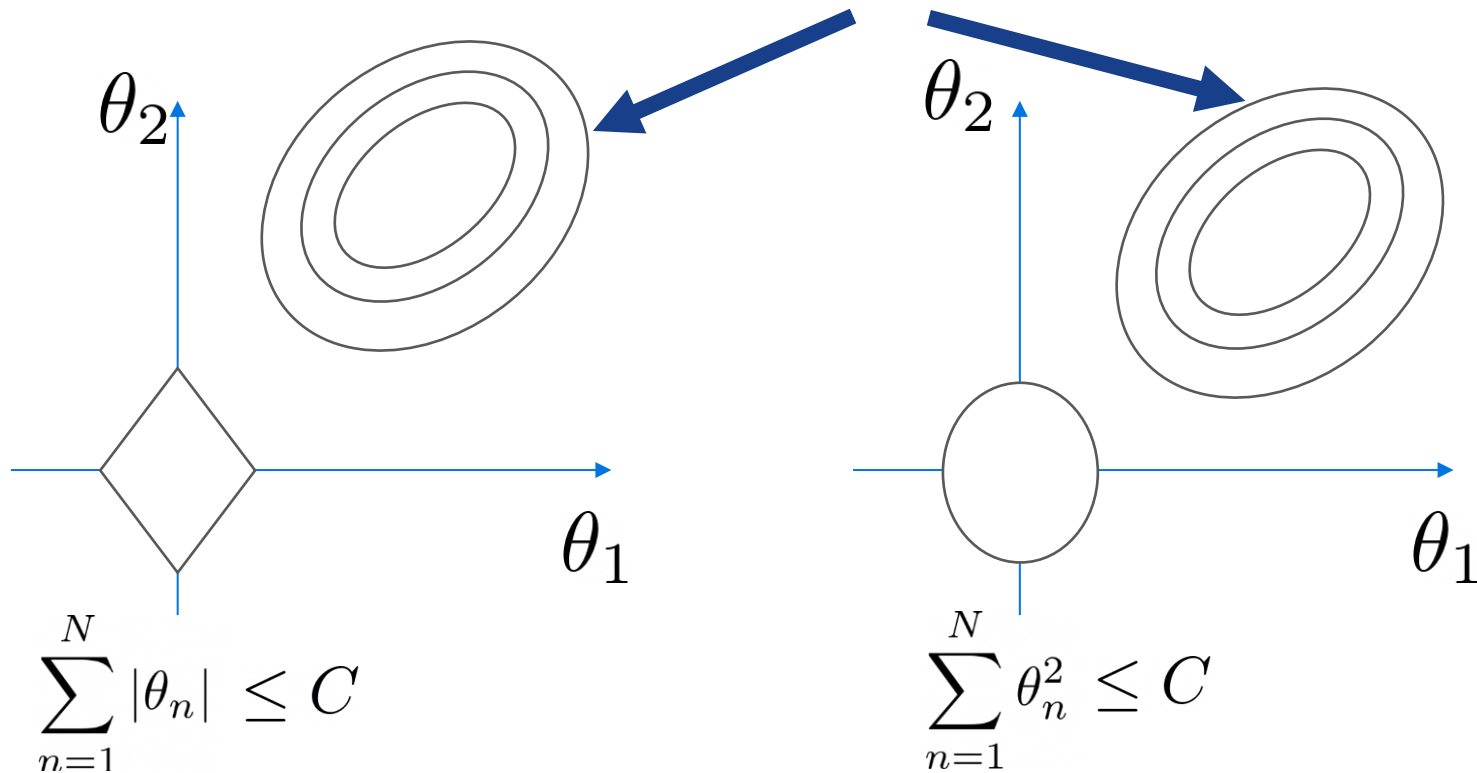
Regresiones con penalización

Predicción

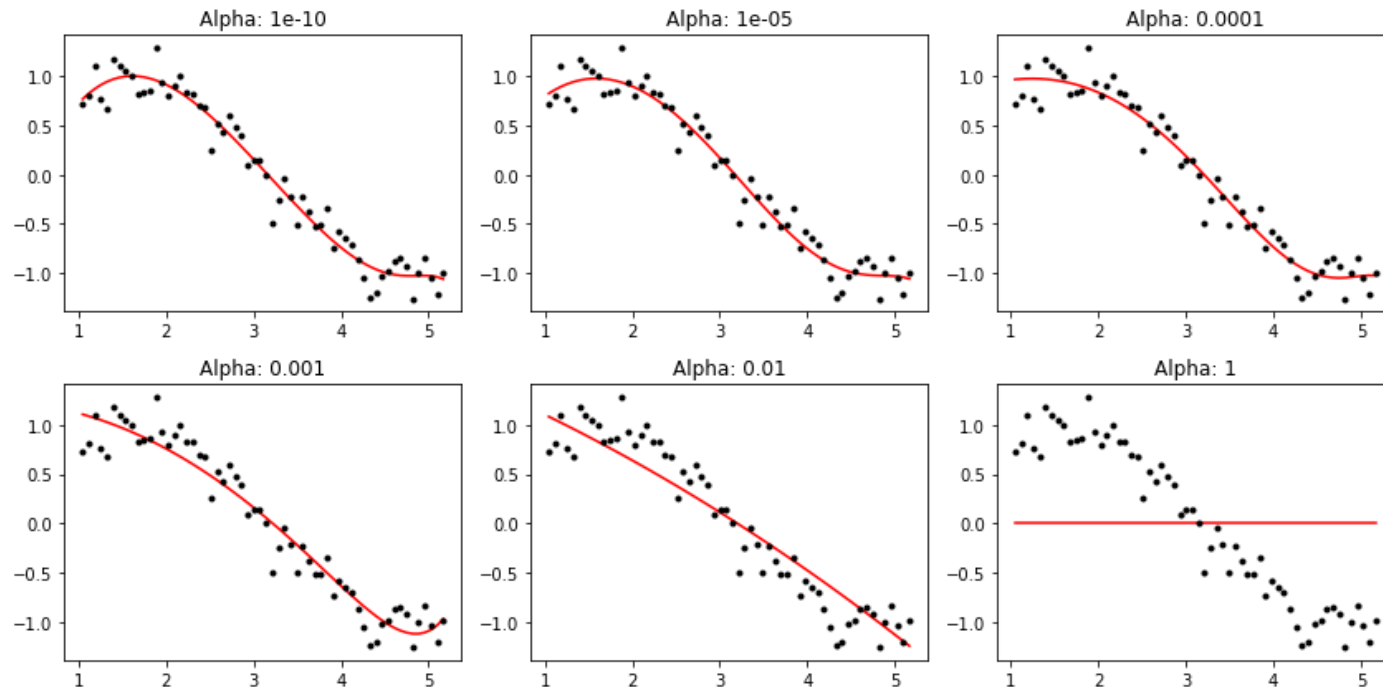
Minimizar $\left\{ \underbrace{\frac{1}{m} \sum_{i=1}^m (\theta^T \cdot \mathbf{x}^i - y^i)^2}_{\text{Mínimo MSE}} + \underbrace{\lambda f(\theta)}_{\text{Penalización}} \right\}$

Ejemplos $\sum_{n=1}^N |\theta_n|$ y $\sum_{n=1}^N \theta_n^2$

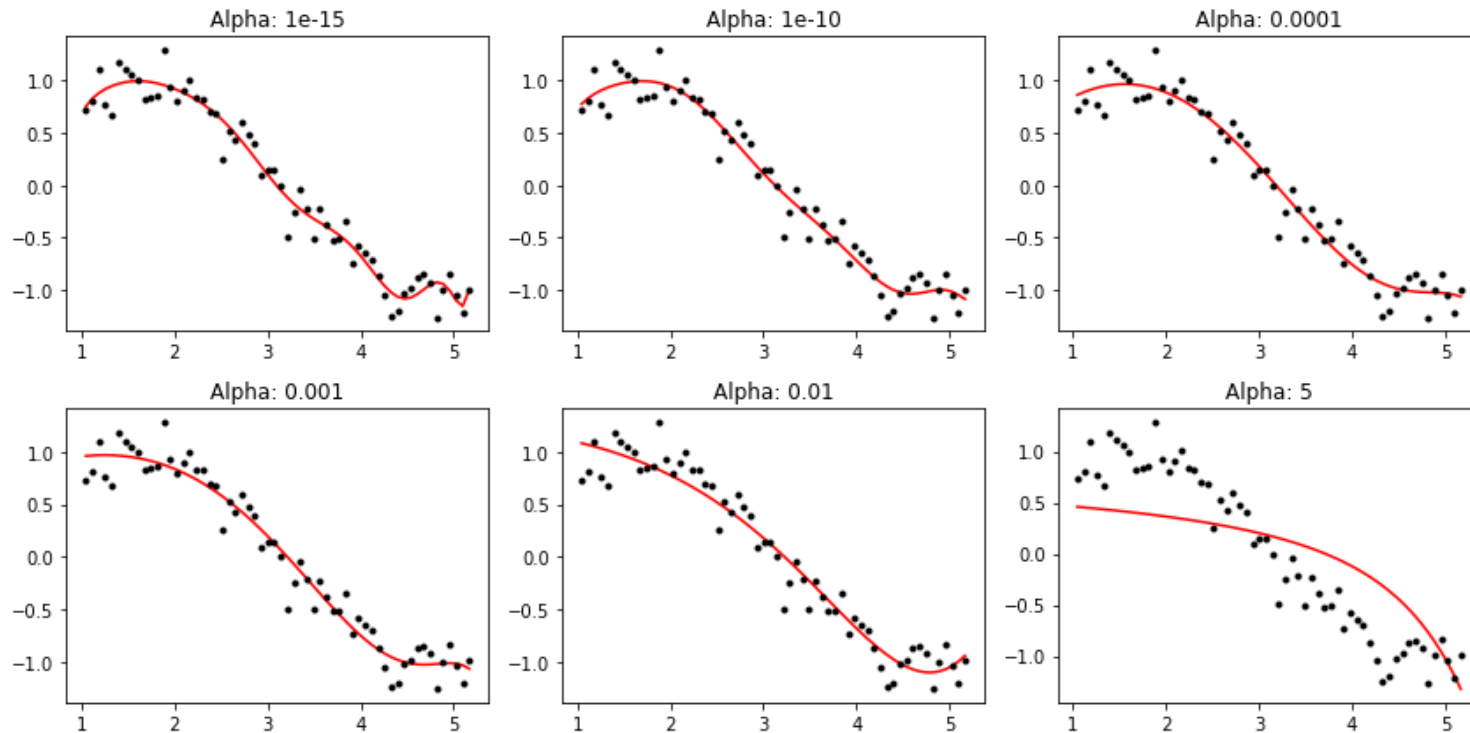
$$\frac{1}{m} \sum_{i=1}^m (\theta^T \cdot \mathbf{x}^i - y^i)^2$$



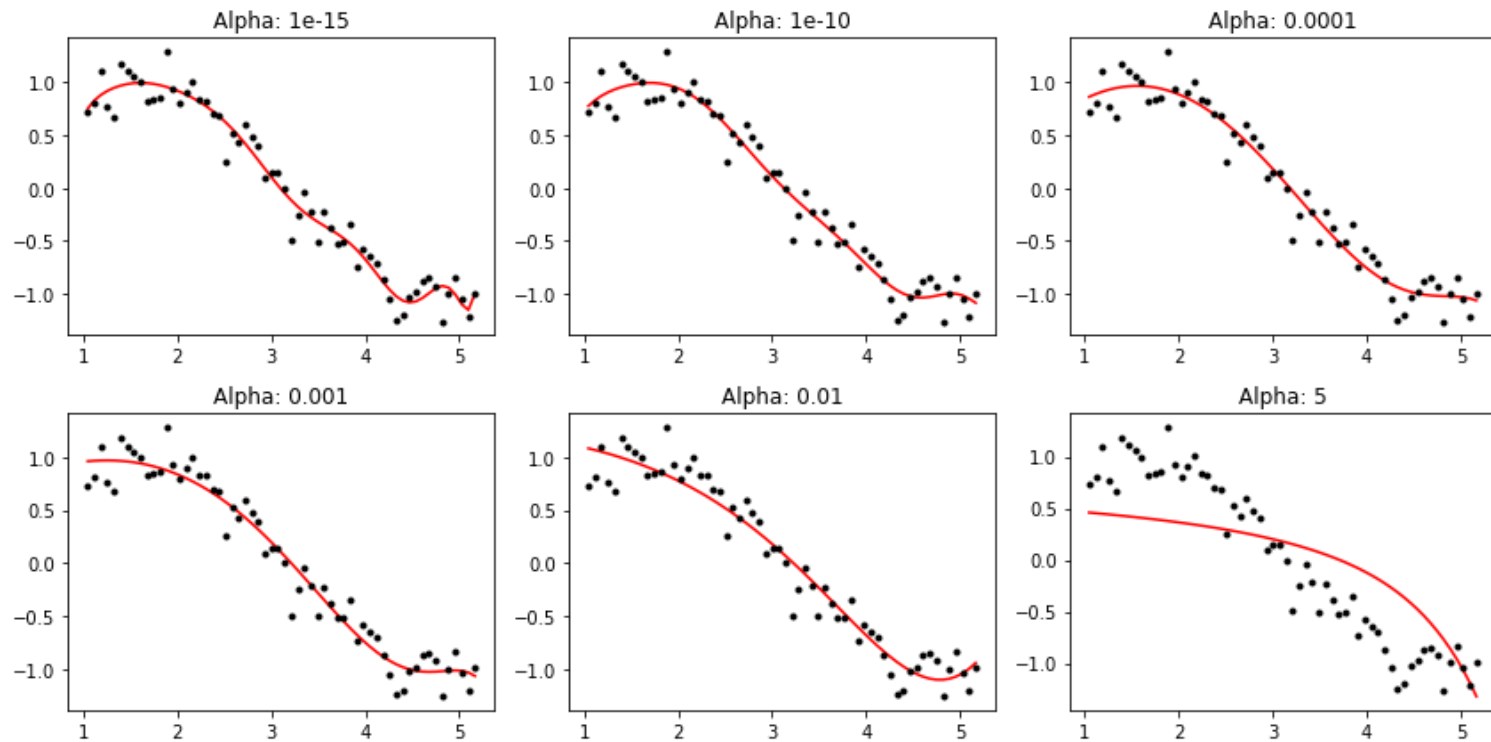
Regularización L1



Regularización L1



Regularización L1





Evaluación de regresiones

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|$$

https://scikit-learn.org/stable/modules/model_evaluation.html

<https://towardsdatascience.com/what-are-the-best-metrics-to-evaluate-your-regression-model-418ca481755b>

www.educacionprofesional.ing.uc.cl

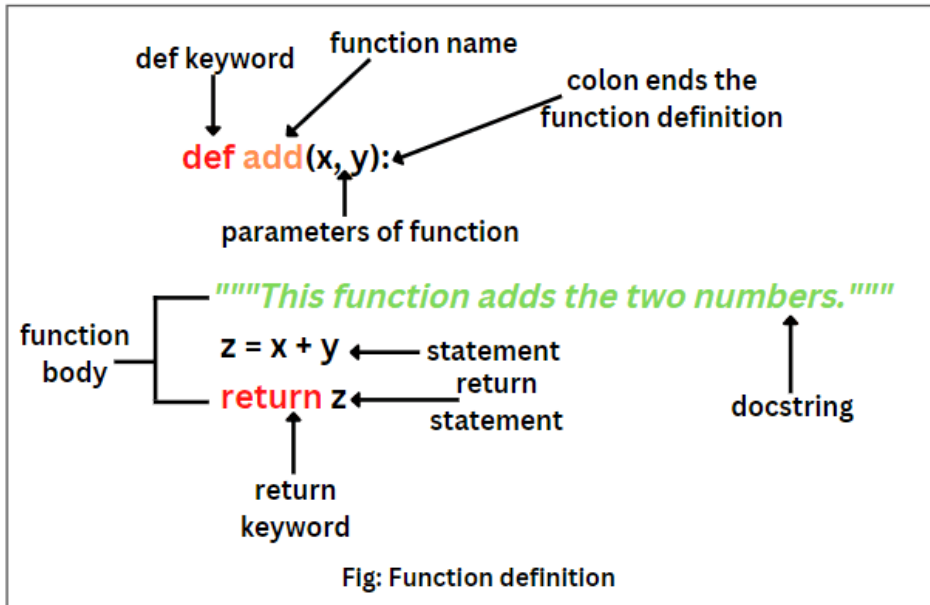


ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

EDUCACIÓN
PROFESIONAL

Implementación Actividad 1

Funciones y clases en Python



```
def sumar_numeros(a, b, c=None):  
    'Esta función calcula la suma de dos o tres números'  
    if c:  
        suma = a+b+c  
    else:  
        suma = a+b  
    return suma  
  
sumar_numeros(1,2)
```

https://www.scientecheasy.com/2022/11/functions-in-python.html/#google_vignette



Funciones y clases en Python

```
class DataPipeline:
    """A simple data pipeline class to load, clean, and process data."""
    def __init__(self, source):
        """
        Parameters:
        - source (str): Path or URL of the data source.
        """
        self.source = source
        self.data = None

    def load_data(self):
        """Load data from the source."""
        self.data = pd.read_csv(self.source)
        print("Data loaded successfully.")

    def clean_data(self):
        """Clean the loaded data by removing unwanted characters."""
        #TODO

    def run_pipeline(self):
        """Run the entire data pipeline in sequence."""
        self.load_data()
        self.clean_data()

pipeline = DataPipeline("path/to/data.csv") # Initialize pipeline with a data source
pipeline.run_pipeline() # Run the complete data pipeline
```



Bibliografía

- McKinney, W. (2012). *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. " O'Reilly Media, Inc."
- Muller, A. C., & Guido, S. (2017). Introduction to machine learning with Python: a guide for data scientists. O'Reilly Media.
- Mitchell, T. (1997). Machine Learning. MacGraw-Hill Companies. Inc., Boston.



ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

EDUCACIÓN
PROFESIONAL

Diplomado en Programación y Aplicaciones de Python

Aplicaciones en Ciencia de Datos e
Inteligencia Artificial

Francisco Pérez Galarce

fjperez10@uc.cl

