

Universidad de Guadalajara



Thompson Rivers University Seminar

Introduction to Machine Learning

Dr. Carlos Villaseñor

Day 1 (Dr. Carlos Villaseñor)

- Exploring data with Pandas
- Introduction to machine learning (ML)
- Paradigms of learning
- Prototypical problems

Rest

- Linear Regression
- Regression performance metrics
- Scaling data and Polynomial regression
- Overfit and underfit

Rest

- Nonlinear regression (DT, SVM, KNN, MLP)
- Hyperparameter search

Day 2 (Dr. Carlos Villaseñor)

- Classification problem
- Logistic regression
- Nonlinear classification (DT, SVM, KNN, MLP)

Rest

- Classification metrics
 - Confusion matrix
 - Classification report

Rest

- Non supervised learning
 - Clustering techniques (Kmeans, Spectral clustering, DBSCAN)
 - Silhouette score
 - Dimensionality reduction (PCA, t-SNE)

Day 3 (Dr. Carlos Villaseñor)

- Introduction to neural networks
 - Biological neuron
 - Artificial neuron
 - Perceptron algorithm

Rest

- Gradient Descent
- Linear Neurons and Logistic neurons (MSE and BCE)
- One-layer Network (Softmax and CCE)
- How to model the last layer of a net

Rest

- Multi-layer Perceptron
- Keras/TensorFlow framework

Day 4 (Dr. Javier Gómez)

- Introduction to digital image processing
- Introduction to convolutional neural networks

Day 5 (Dr. José Hernández)

- Practice machine learning with a real problem



<https://github.com/Dr-Carlos-Villasenor/TRSeminar.git>

Guidelines of the course



- The first to days of the course we are going to avoid the mathematical detail of the ML models, instead we are going to focus on the ML practice.
- In the third day, we are going to explore neural networks with some mathematical detail.
- I've included some random facts about Mexico!!.



Libraries to learn



<https://pandas.pydata.org/>



<https://scikit-learn.org/>

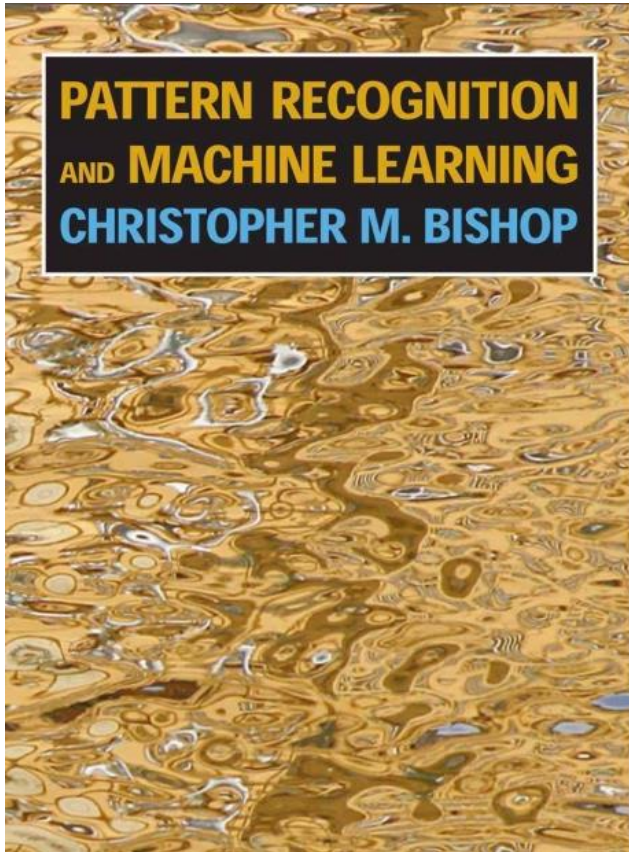


TensorFlow

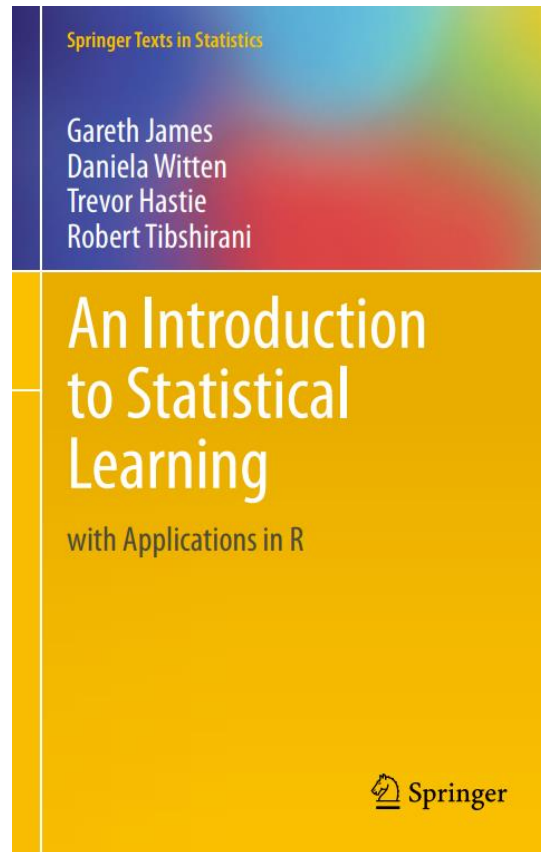
<https://www.tensorflow.org/>



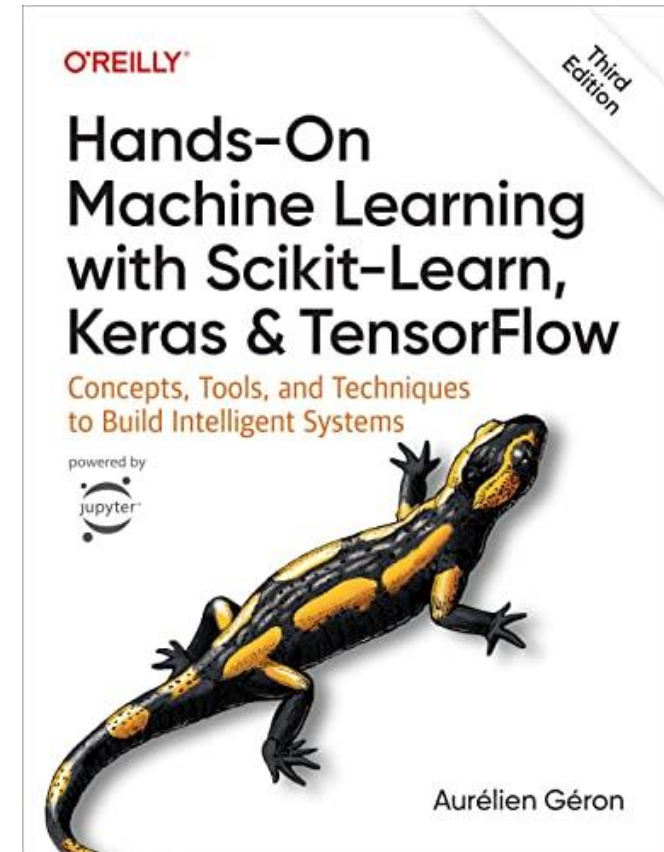
What to read after this lectures?



Bishop, C. M., & Nasrabadi, N. M. (2006). Pattern recognition and machine learning (Vol. 4, No. 4, p. 738). New York: springer.

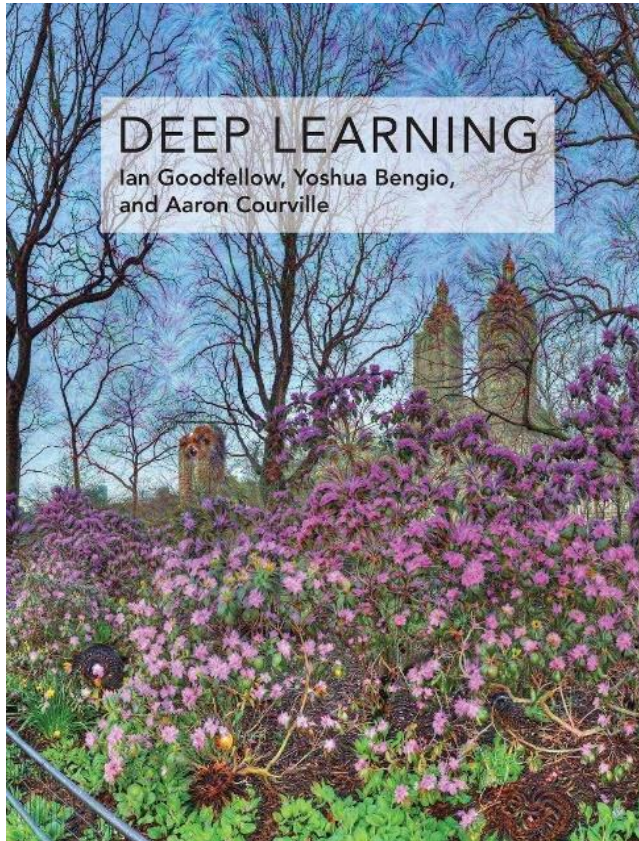


James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.

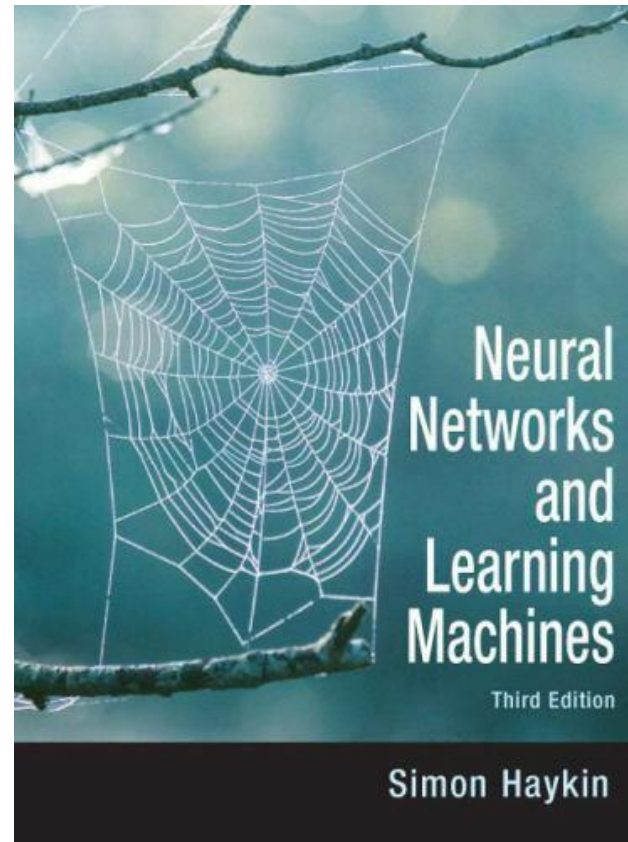


Géron, A. (2022). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. " O'Reilly Media, Inc."..

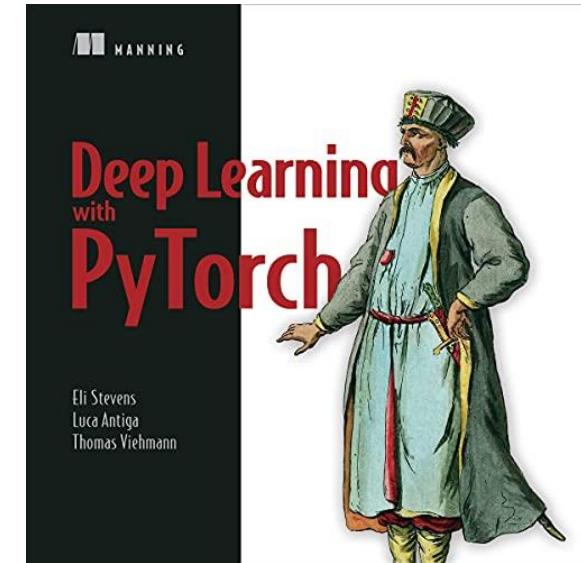
What to read after this lectures?



Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.



Haykin, S. (2009). *Neural networks and learning machines*, 3/E. Pearson Education India.



Stevens, E., Antiga, L., & Viehmann, T. (2020). *Deep learning with PyTorch*. Manning Publications.

Introducción a pandas



Pandas (**P**anel **d**ata) is a Python library focused on Data manipulation and analysis. Pandas offers a data structure (DataFrame) and Time series.



Let's get hands on! Open https://github.com/Dr-Carlos-Villasenor/TRSeminar/blob/main/TRS01_Pandas.ipynb



Jericallas!

Jericalla is a Mexican dessert from the city of Guadalajara, it is made of egg, milk, vanilla, cinnamon and sugar.

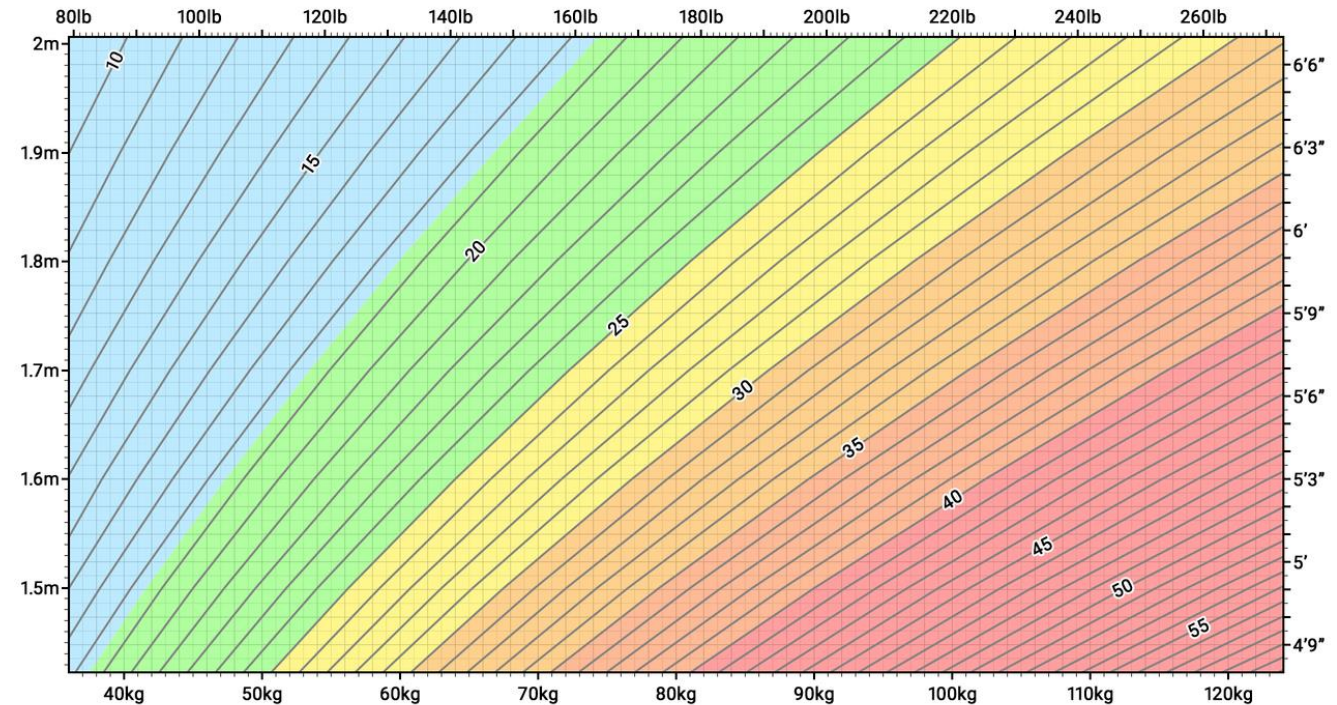


Ask for one: “Quiero una jericalla”

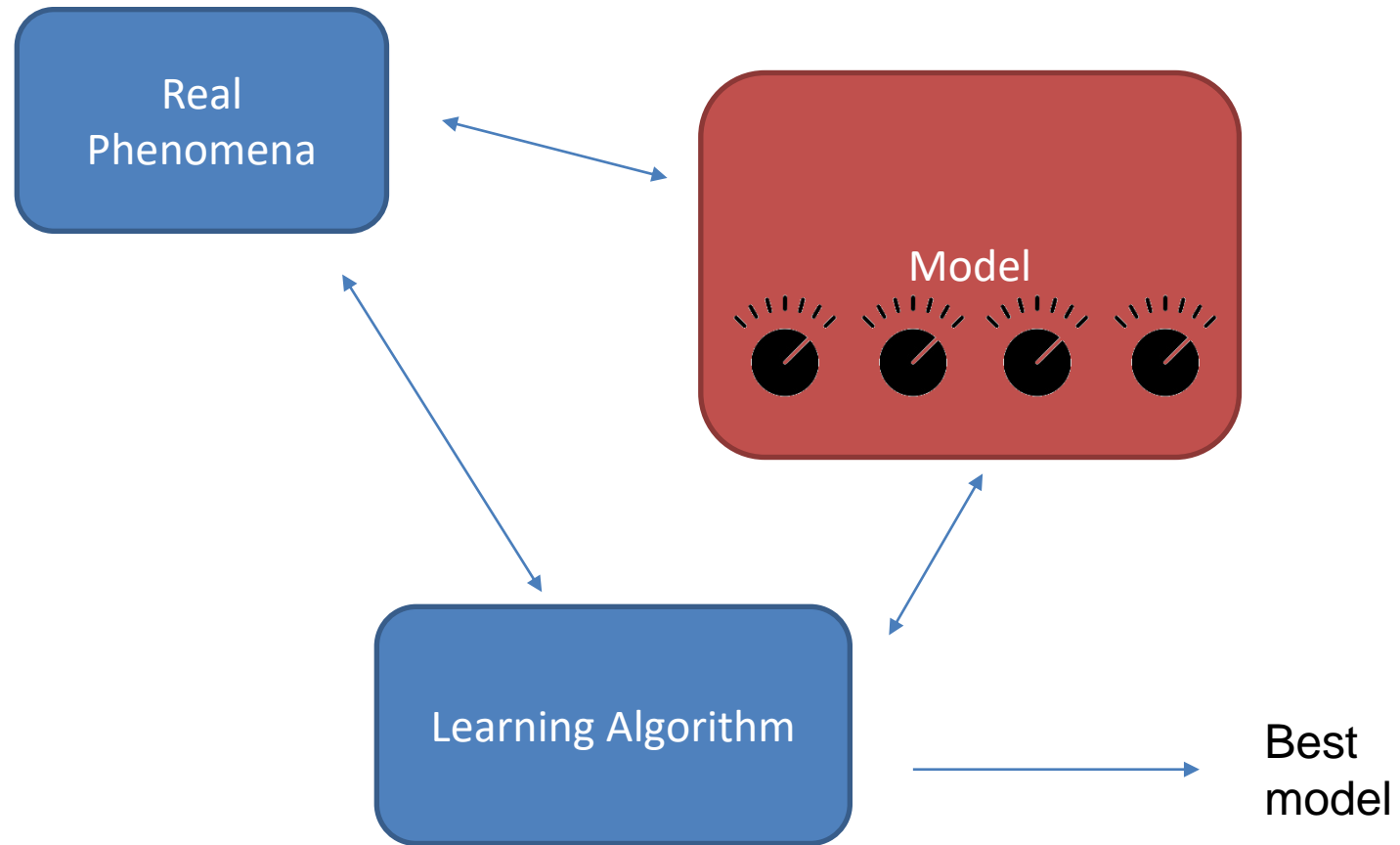
What is learning?

- Develop a program that predicts if a person is overweight

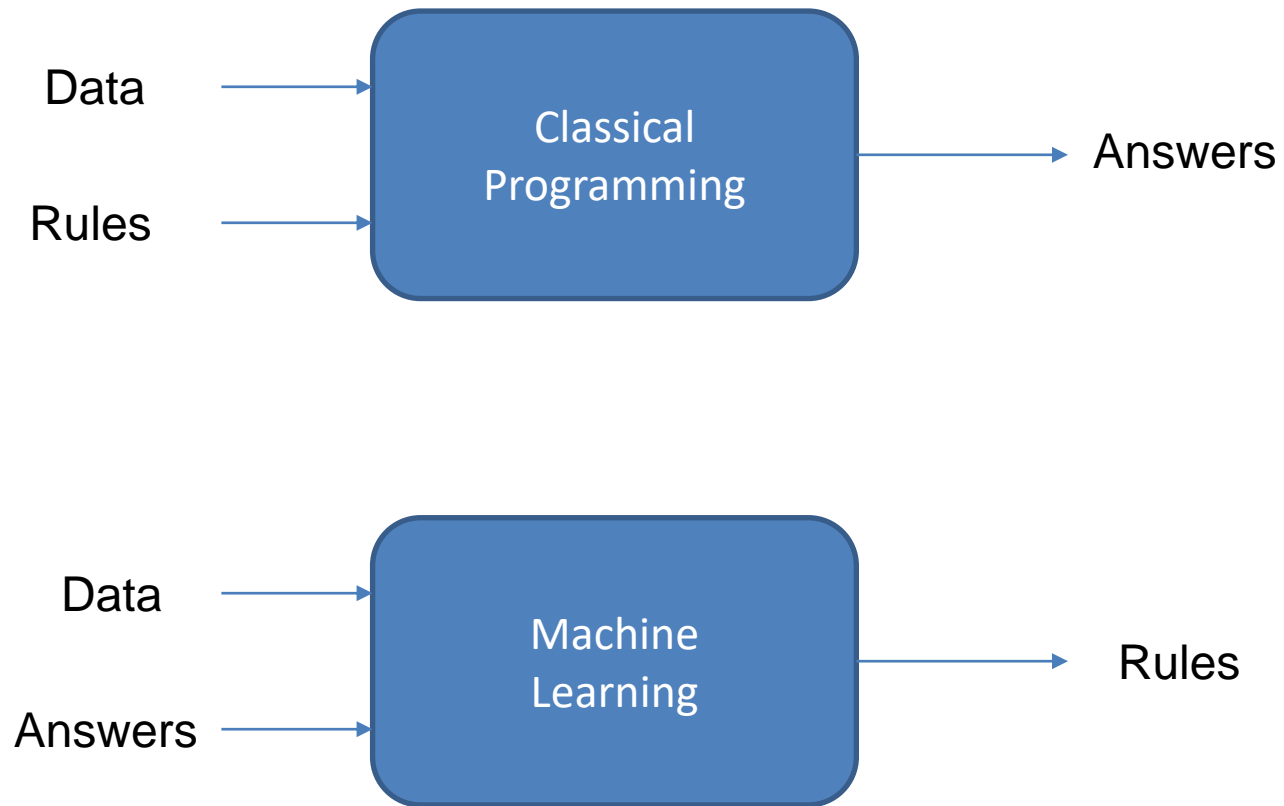
```
def overweight(mass_kg, height_m):  
    bmi = mass_kg / height_m**2  
    if bmi < 25:  
        return True  
    else:  
        return False
```



What is learning?



What is learning?



What is learning?



- Tom Mitchell definition:

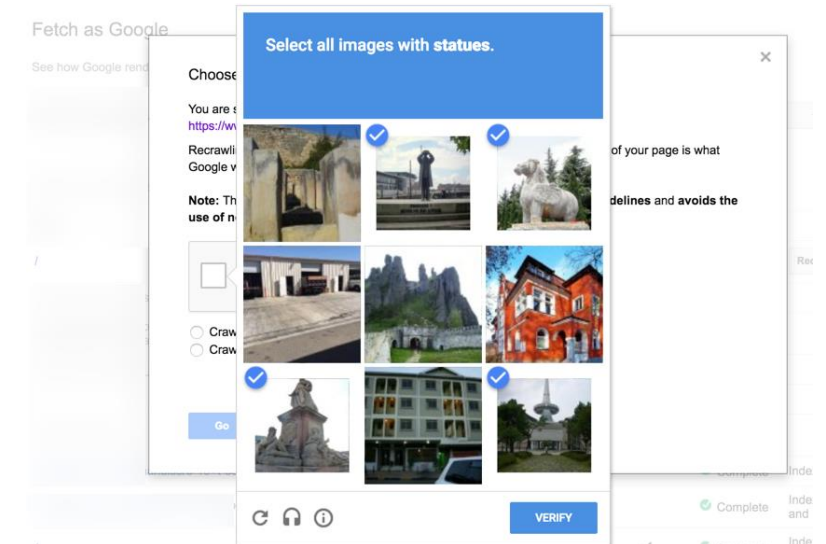
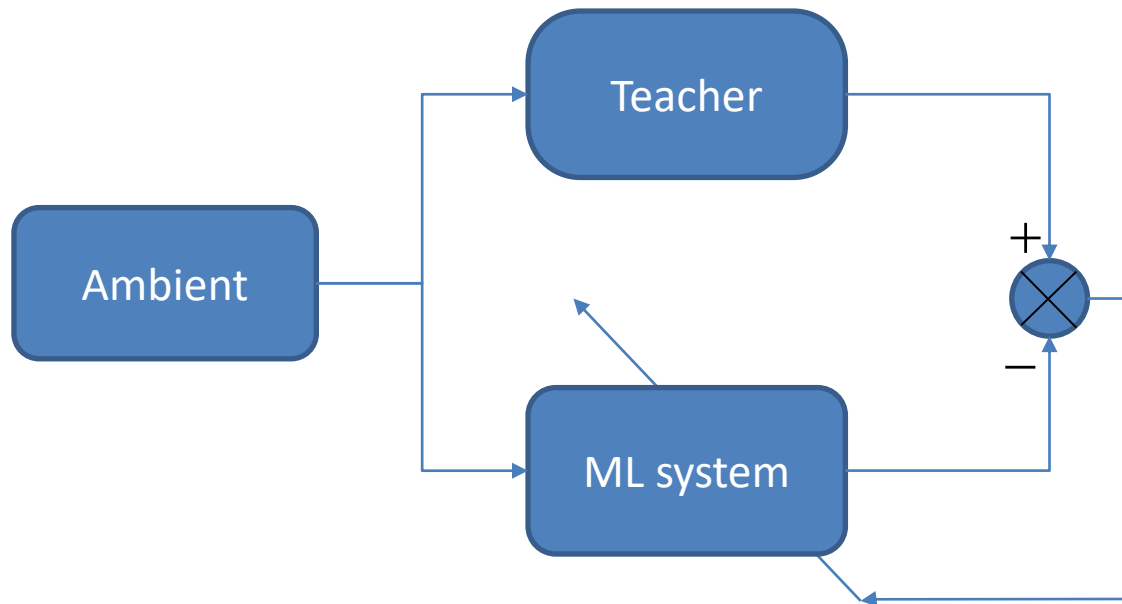
“A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P , if its performance at tasks in T , as measured by P , improves with experience E .”



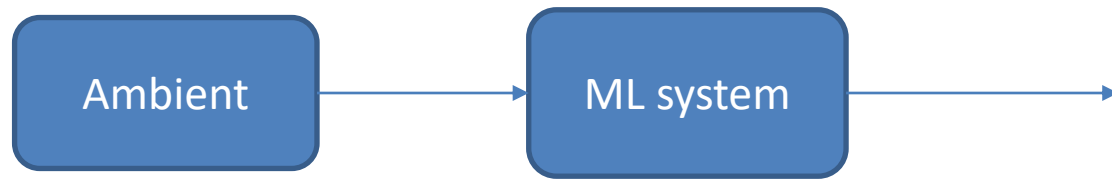
Tom M. Mitchell
Professor at the Carnegie Mellon
University



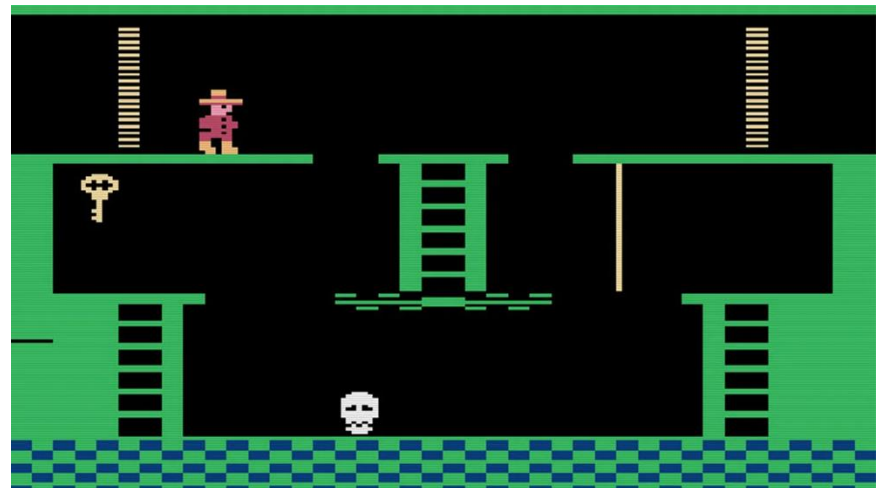
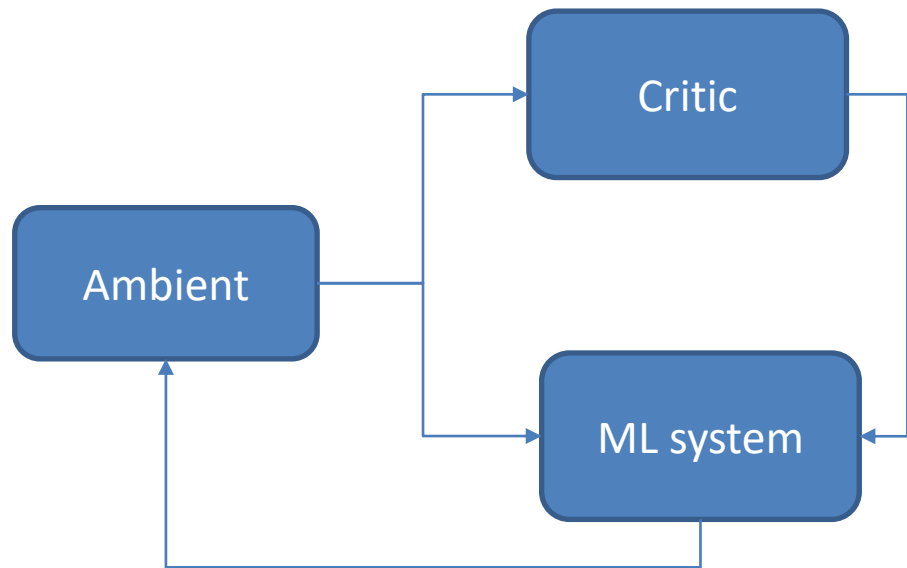
Supervised learning



Non-supervised learning



Reinforcement learning



Prototypical problems



		Wished output	
		Categorical	Continuous
Paradigm	Supervised	Classification	Regression
	Non-supervised	Clustering	Dimensionality reduction



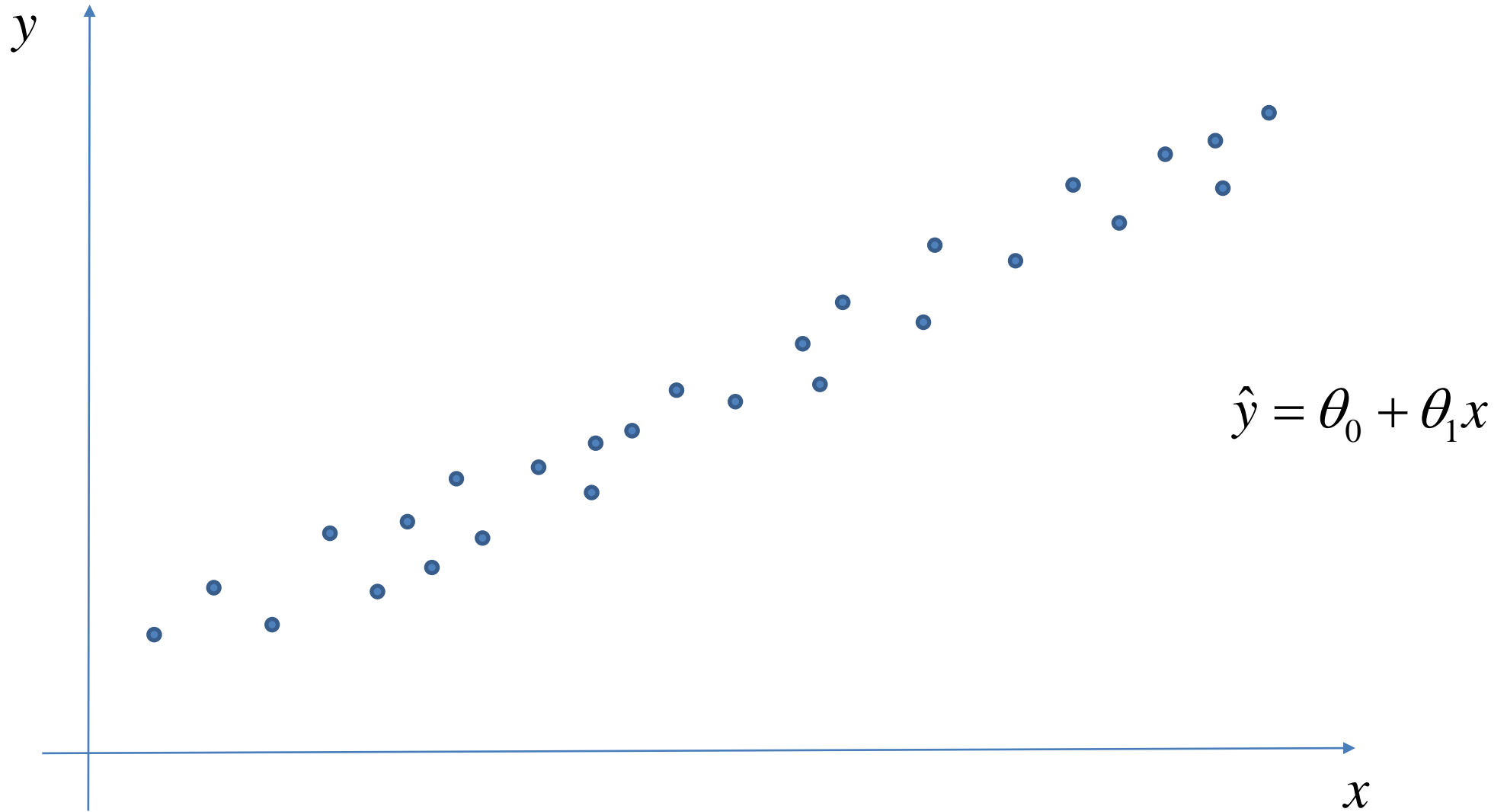
Tejuino

Tejuino is a cold fermented beverage made from corn and popularly consumed in the Mexican states of Jalisco, Colima, Nayarit and Oaxaca. Tejuino is usually made from corn dough, the same kind used for tortillas.



Let's take a rest!

Linear Regression



Let's get hands on! Open https://github.com/Dr-Carlos-Villasenor/TRSeminar/blob/main/TRS02_Linear_regression.ipynb



Consider a supervised dataset

$$x^{(i)} \in \mathbb{R}^n, y^{(i)} \in \mathbb{R}$$

$$D = \left\{ \left(x^{(1)}, y^{(1)} \right), \left(x^{(2)}, y^{(2)} \right), \dots, \left(x^{(p)}, y^{(p)} \right) \right\}$$

- Mean Absolute Error (MAE)

$$\text{MAE}(y, \hat{y}) = \frac{1}{p} \sum_{i=1}^p |y^{(i)} - \hat{y}^{(i)}|$$

- Median Absolute Error (MedAE)

$$\text{MedAE}(y, \hat{y}) = \text{median}\left(\left|y^{(1)} - \hat{y}^{(1)}\right|, \dots, \left|y^{(p)} - \hat{y}^{(p)}\right|\right)$$

- Mean Squared Error (MSE)

$$\text{MAE}(y, \hat{y}) = \frac{1}{p} \sum_{i=1}^p \left(y^{(i)} - \hat{y}^{(i)}\right)^2$$

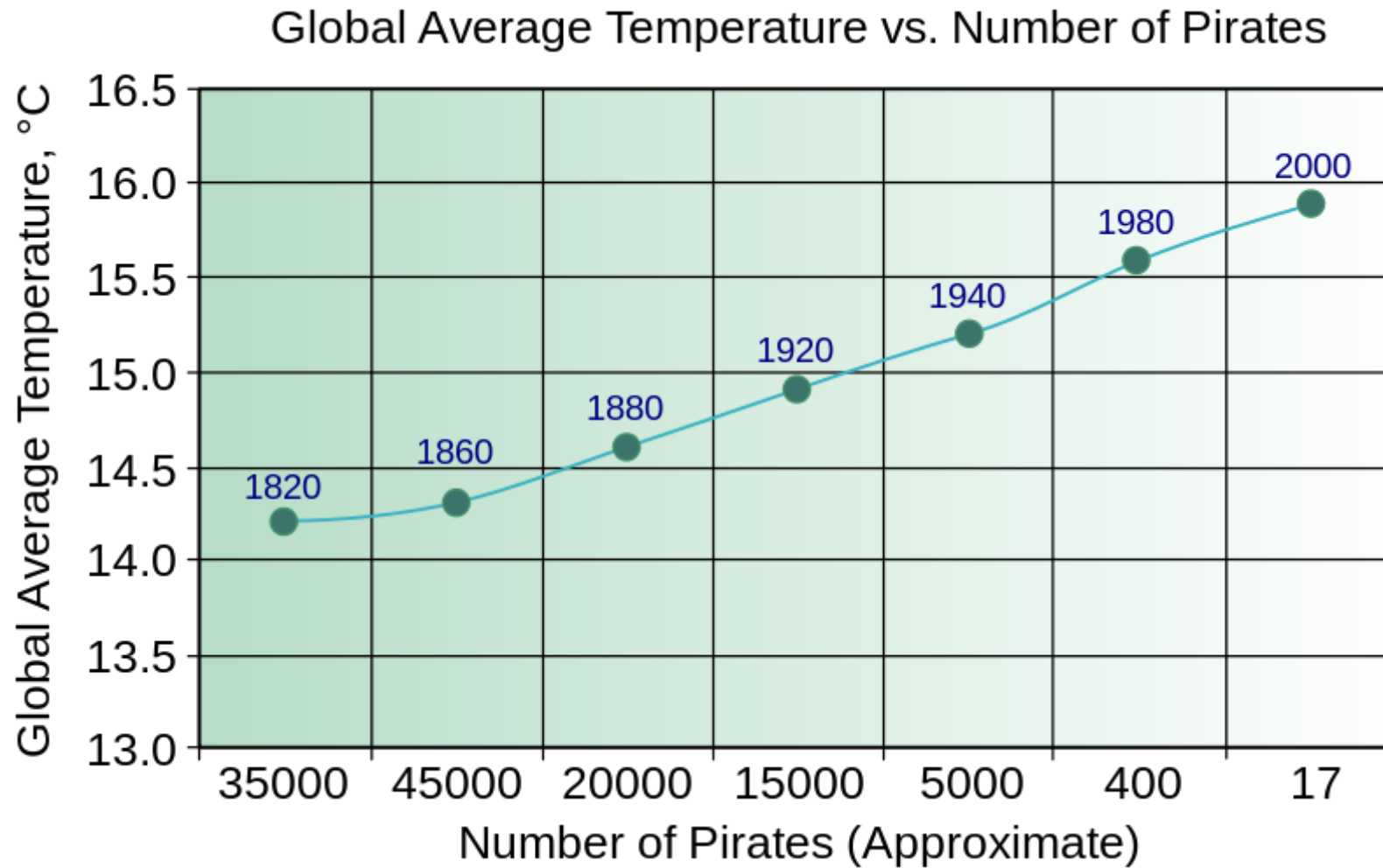
- Explained Variance Score (EVS)

$$\text{EVS}(y, \hat{y}) = 1 - \frac{\text{var}(y - \hat{y})}{\text{var}(y)}$$

- R2-score

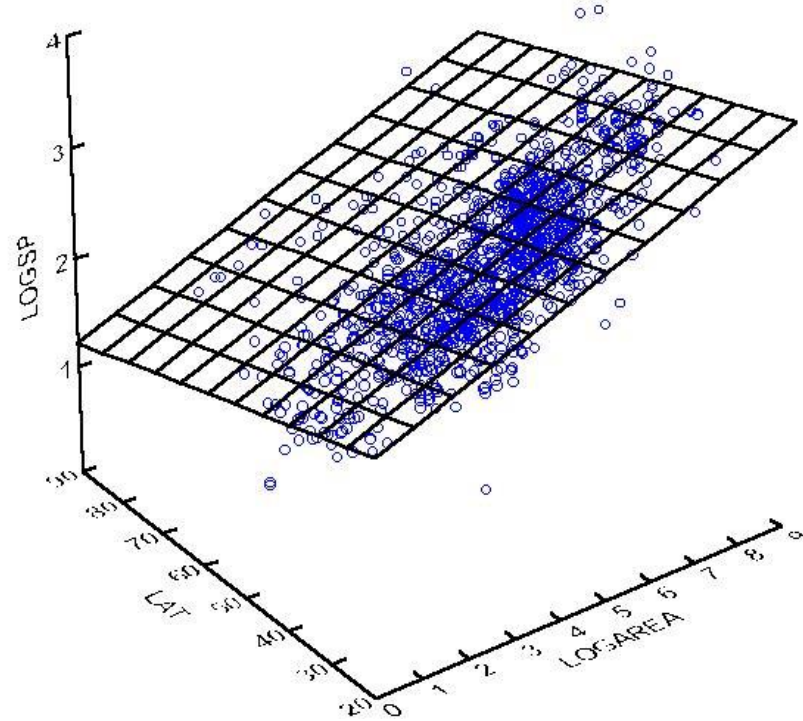
$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^p (y^{(i)} - \hat{y}^{(i)})^2}{\sum_{i=1}^p (y^{(i)} - \bar{y})^2}$$

Correlation vs causality



Linear regression

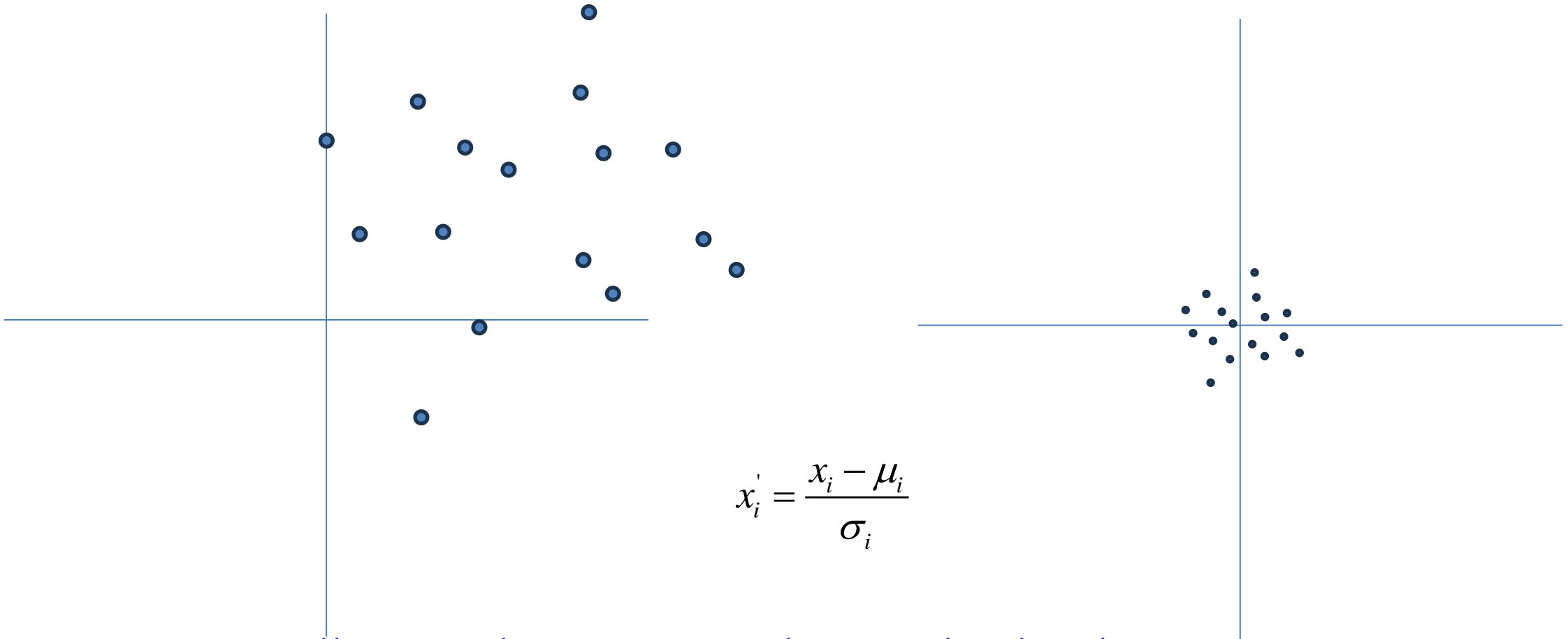
$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$



$$\text{Loss}(y, \hat{y}) = \frac{1}{2p} \sum_{i=1}^p (y^{(i)} - \hat{y}^{(i)})^2$$

Scaling data

Standard scaler

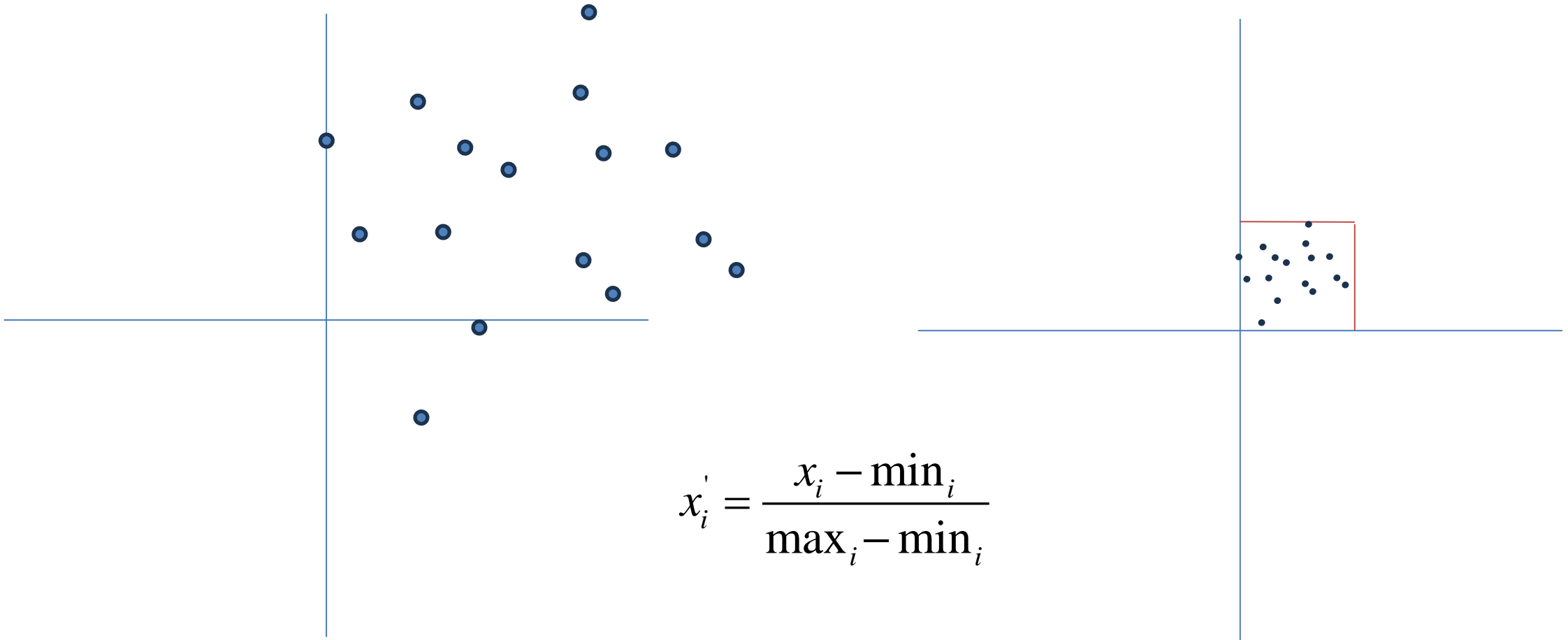


$$x'_i = \frac{x_i - \mu_i}{\sigma_i}$$

Let's get hands on! Open https://github.com/Dr-Carlos-Villasenor/TRSeminar/blob/main/TRS03_Scaling_data.ipynb

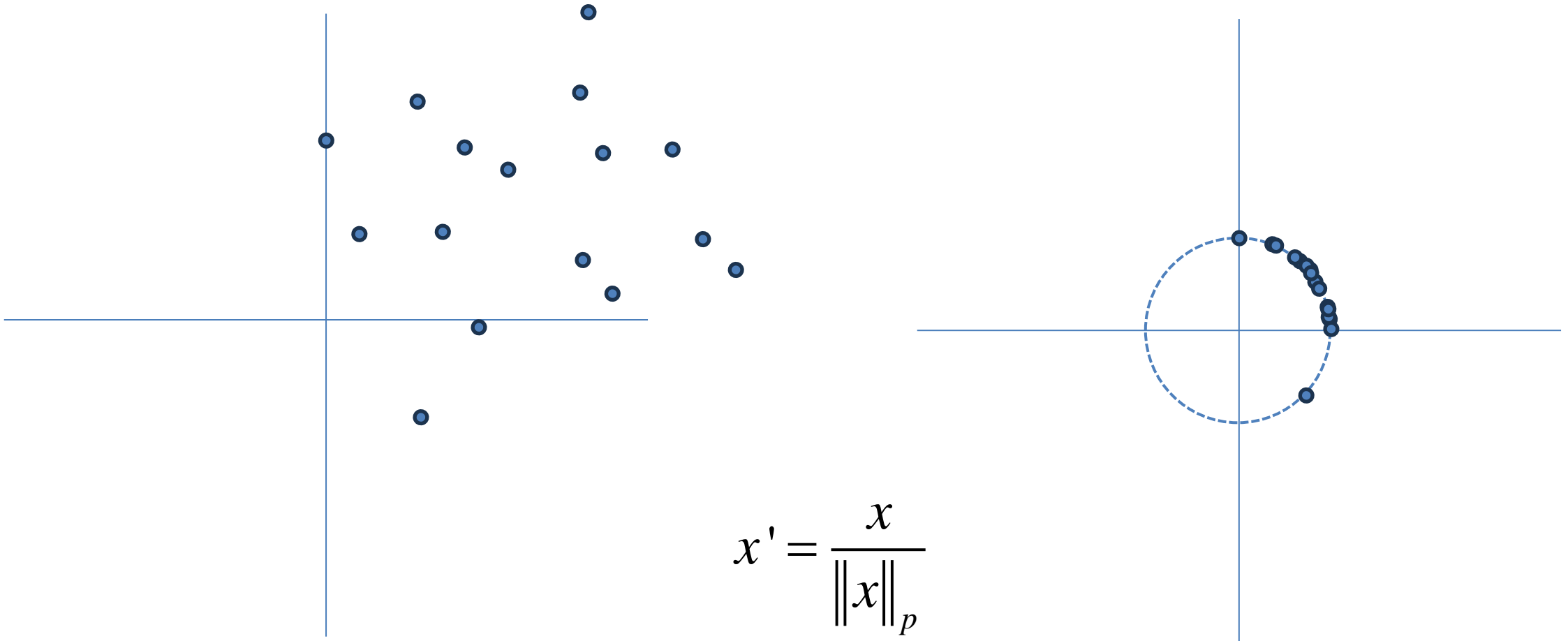
Scaling data

MinMax scaler



Scaling data

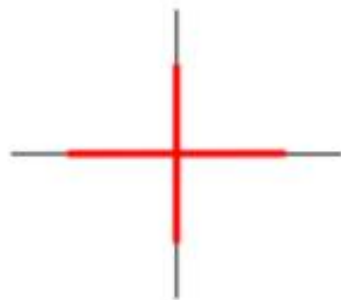
Normalizing scaler



$$x' = \frac{x}{\|x\|_p}$$

Scaling data

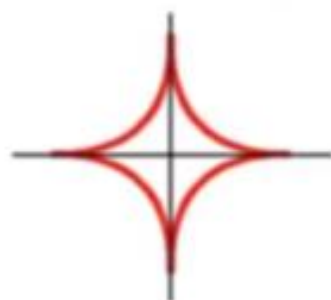
$$\|x\|_p = \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$$



$p \rightarrow 0$

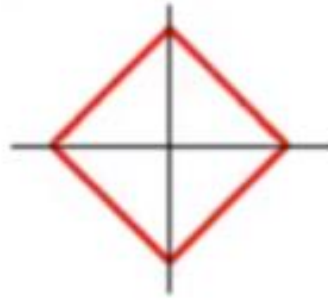
$$\|x\|_0 = (\sum_i |x_i|^0)^\infty$$

The number of non-zero
parameters



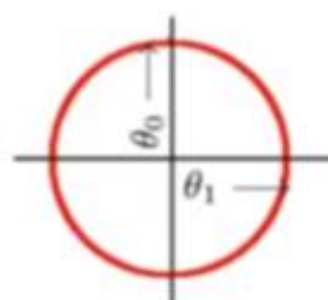
$p = 0.5$

$$\|x\|_{0.5} = (\sum_i |x_i|^{0.5})^2$$



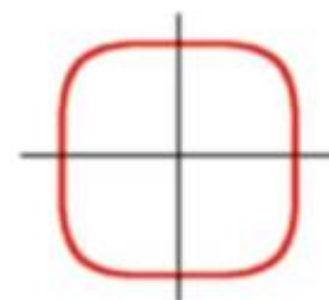
$p = 1$

$$\|x\|_1 = (\sum_i |x_i|^1)^1$$



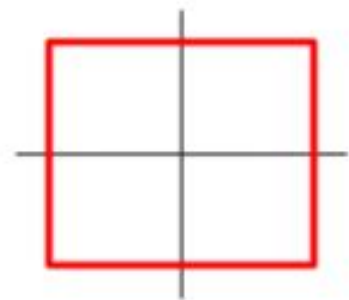
$p = 2$

$$\|x\|_2 = (\sum_i |x_i|^2)^{1/2}$$



$p = 4$

$$\|x\|_4 = (\sum_i |x_i|^4)^{1/4}$$

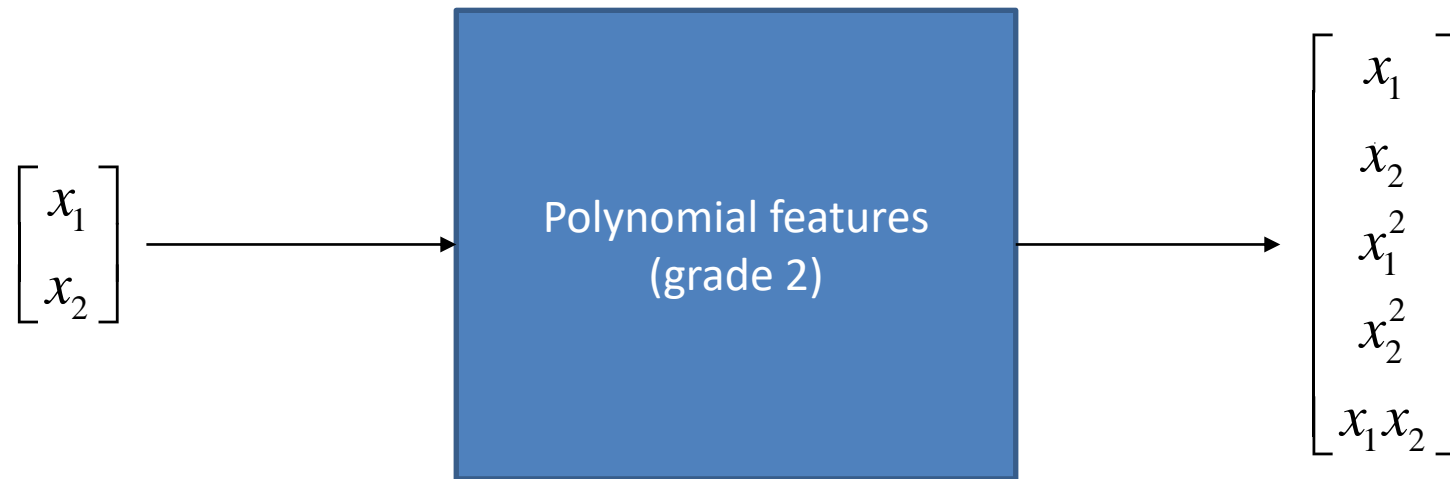


$p \rightarrow \infty$

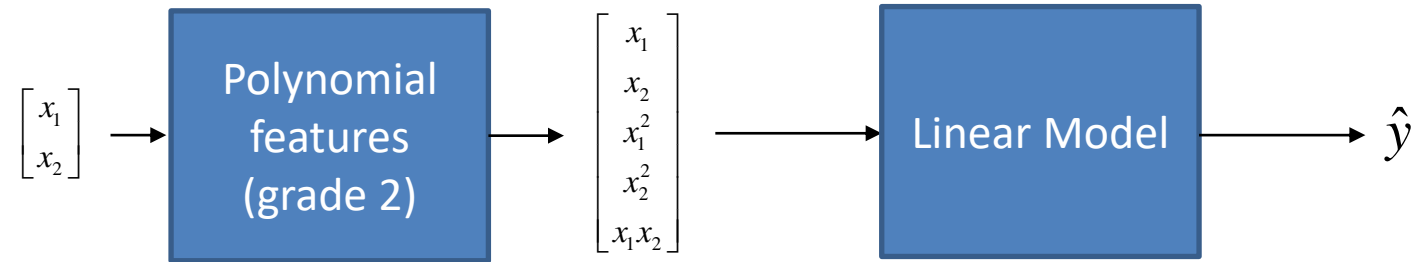
$$\|x\|_\infty = (\sum_i |x_i|^\infty)^0$$

The size of the
largest parameter

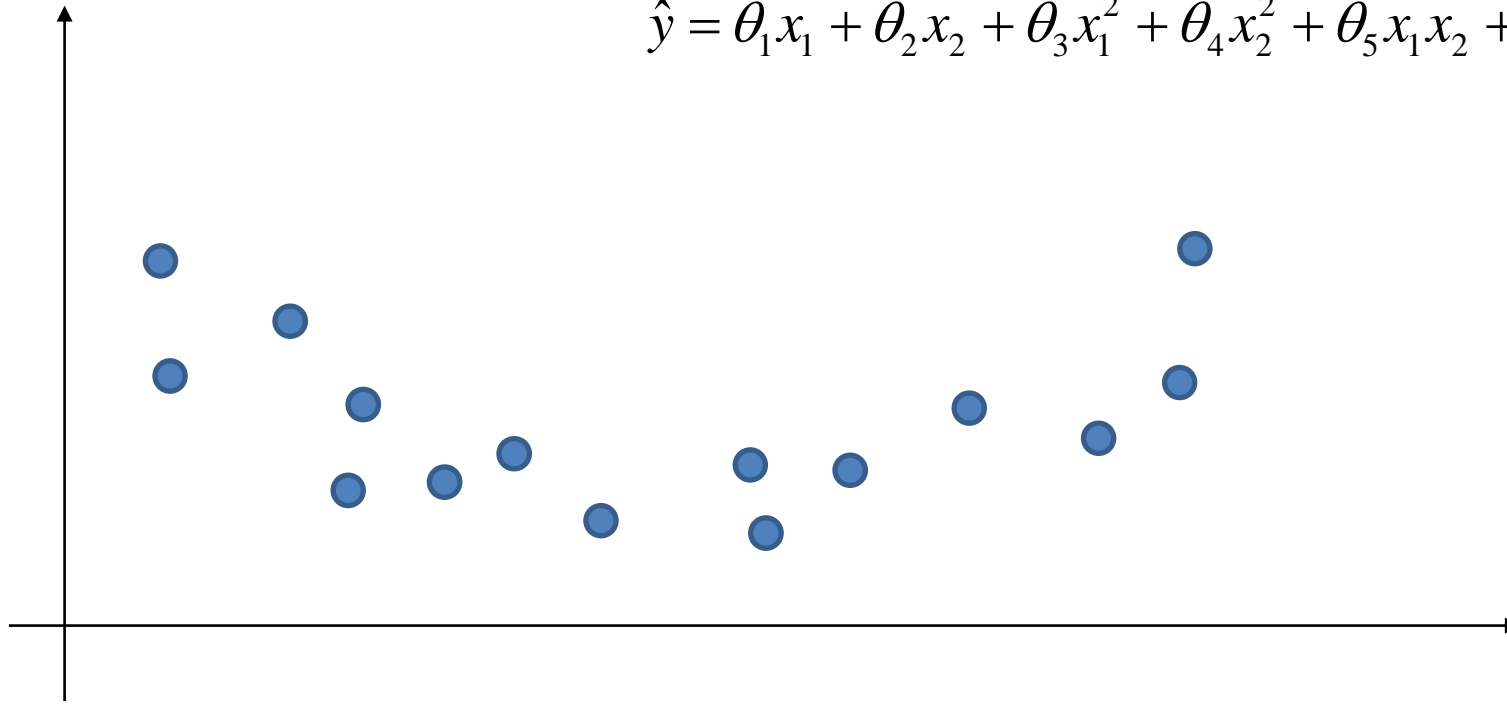
Polynomial Regression



Polynomial Regression



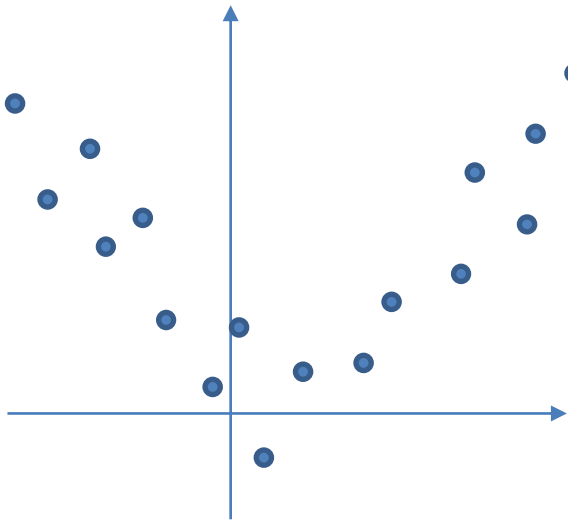
$$\hat{y} = \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2 + \theta_0$$



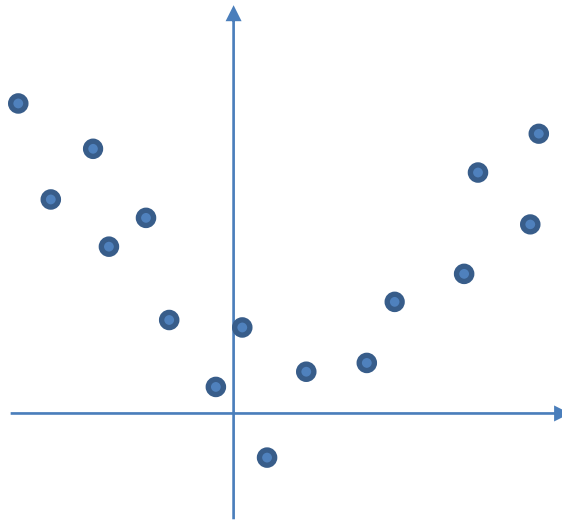
Let's get hands on! Open https://github.com/Dr-Carlos-Villasenor/TRSeminar/blob/main/TRS04_Poly_reg.ipynb

Underfit and overfit

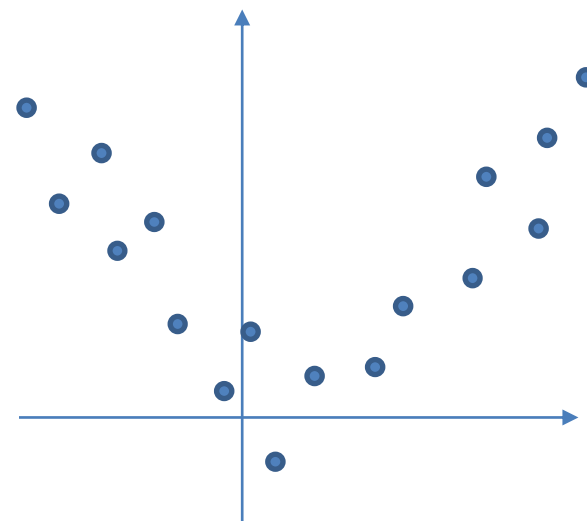
Underfitted



Good generalization

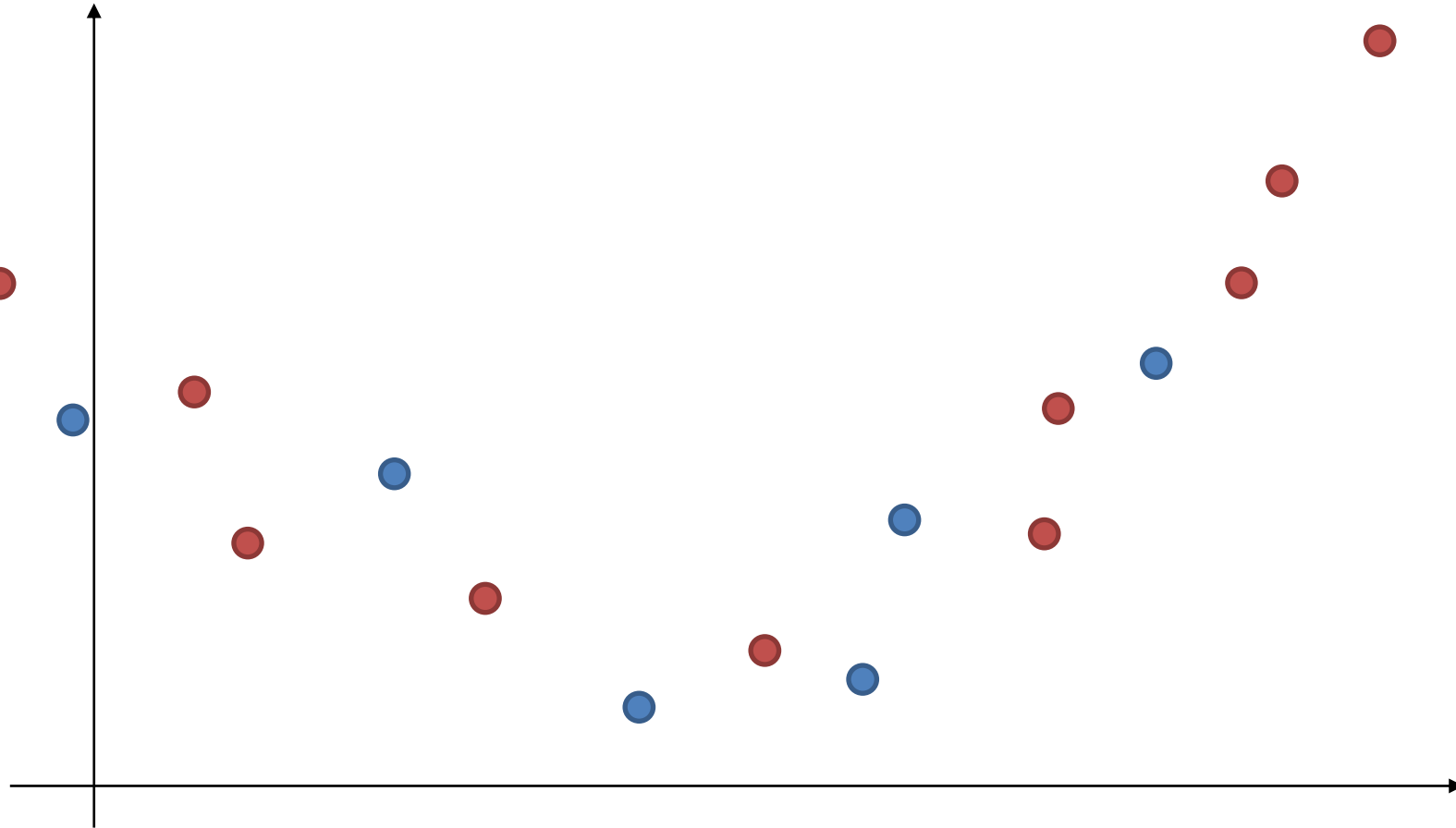


Overfitted



Proving generalization

● Train-set
● Test-set



Train/Test split



Samples	Train-Set	Test-Set
500	75%	25%
2000	80%	20%
10000	90%	10%
100000	98%	2%

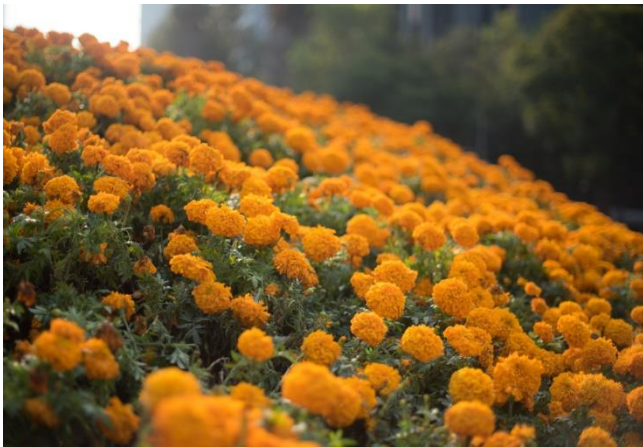


Underfit and overfit



[0-1]	Case 1	Case 2	Case 3	Case 4
Train-set	High	Low	High	Low
Test-set	Low	Low	High	High
	Overfit	Underfit	Good Generalization	Miss-match
	Miss-match			

El día de los muertos



Cempasúchil flowers



Let's take a rest!

Non-linear regression



We will review the following ML models

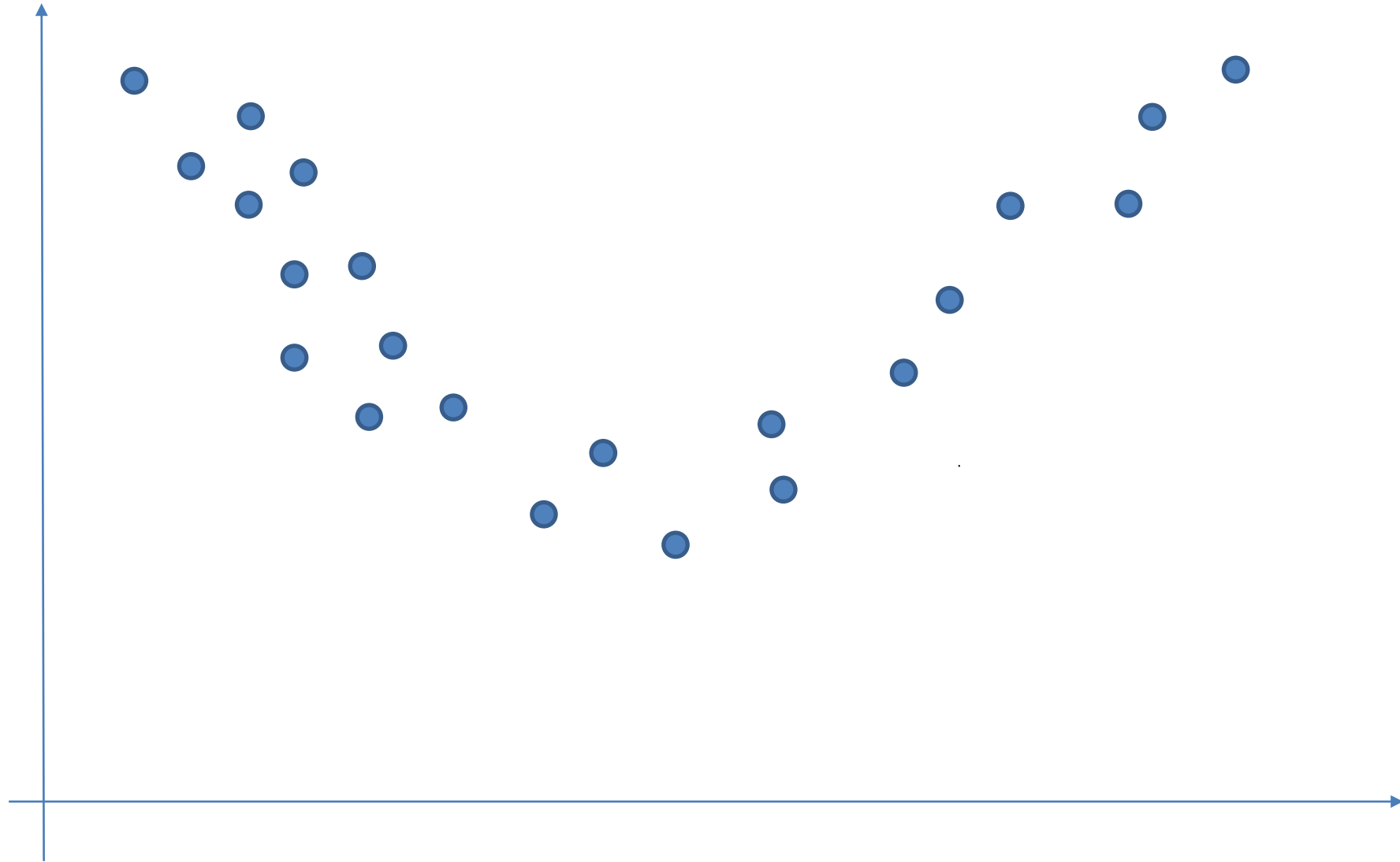
- K Nearest Neighbors (KNN)
- Decision Tree (DT)
- Support Vector Machine (SVM)
- Multi-Layer Perceptron (MLP)

To be a user of these techniques we will experiment with their principal hyper-parameters.

Let's get hands on! Open https://github.com/Dr-Carlos-Villasenor/TRSeminar/blob/main/TRS05_Non_linear_Reg.ipynb



K Nearest Neighbors (KNN)

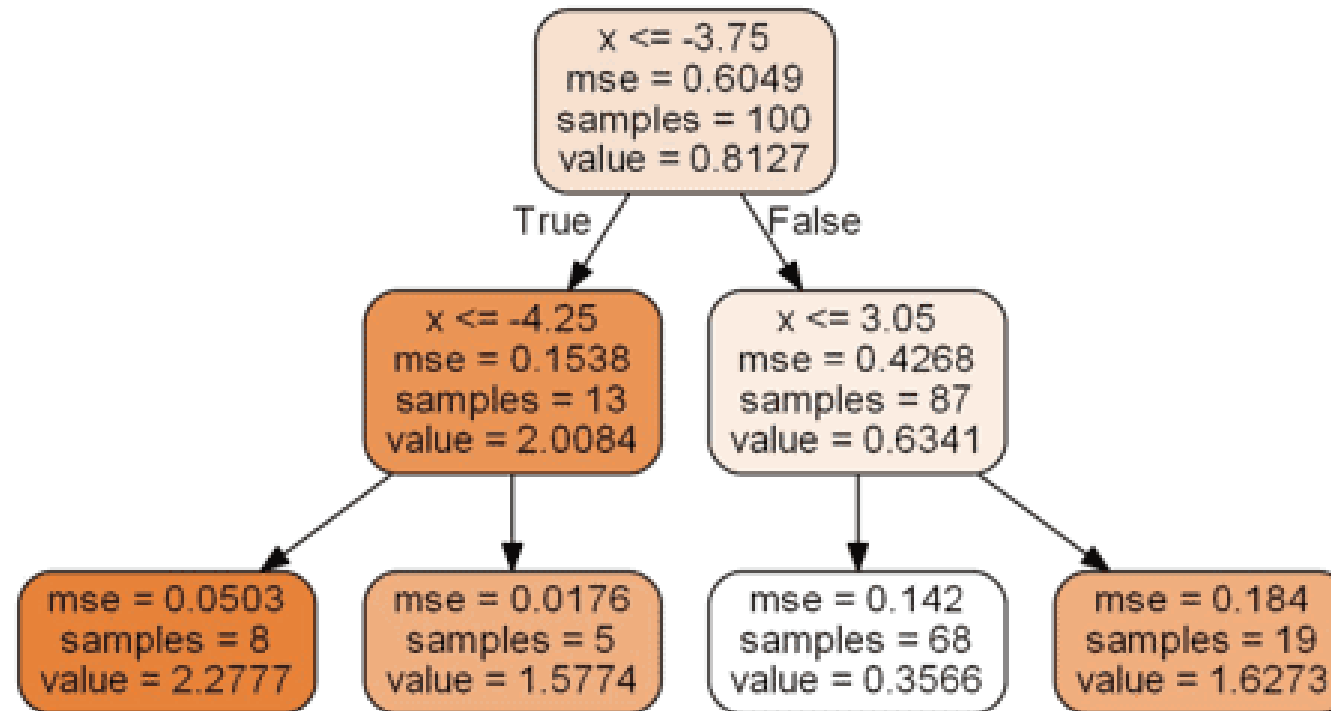


K Nearest Neighbors (KNN)

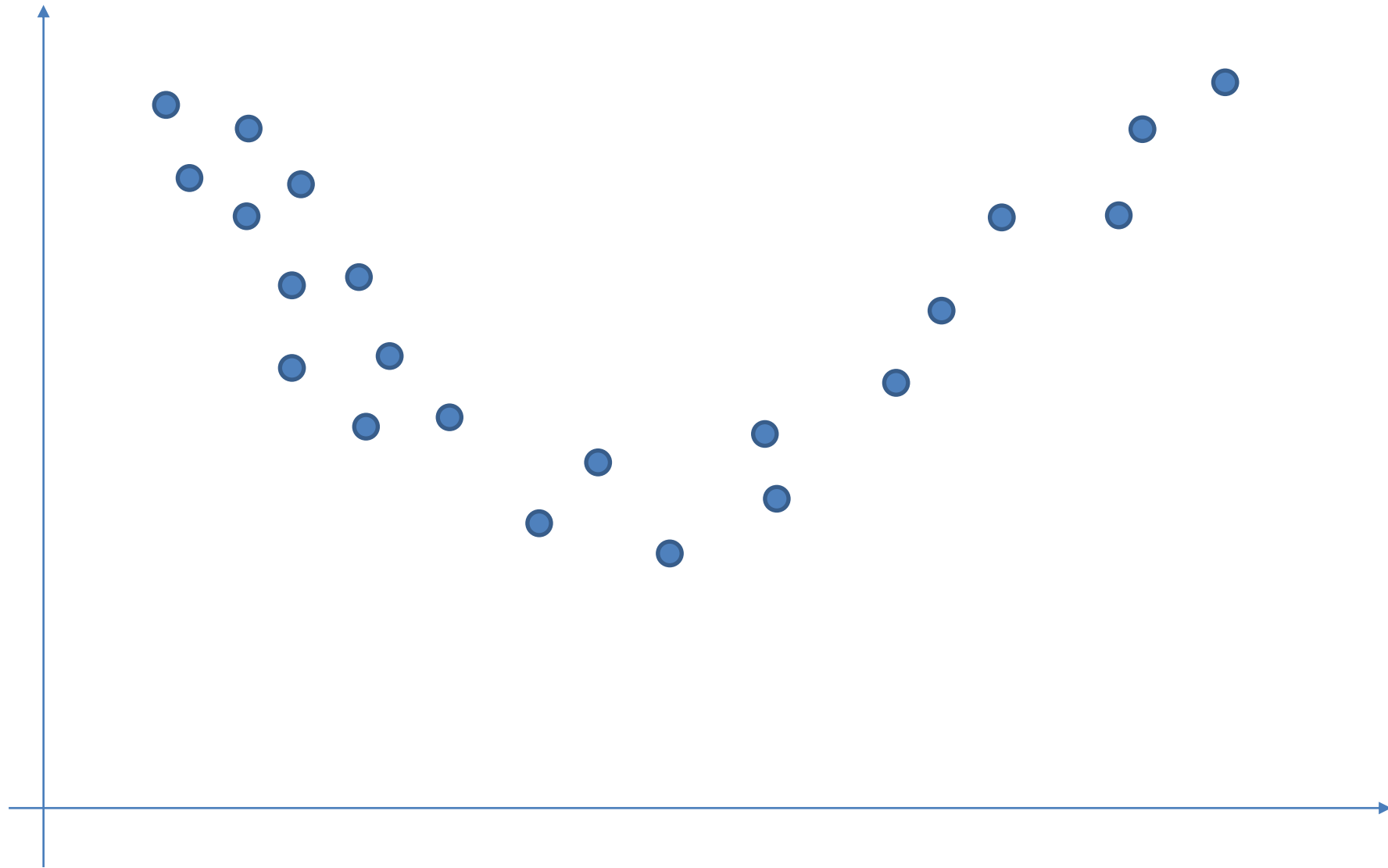


- Hyper-parameters:
 - **n_neighbors** = número de vecinos,
 - **weights**: {'uniform', 'distance'},
 - **algorithm**: {'auto', 'ball_tree', 'kd_tree', 'brute'},
 - **metric**: {'euclidean', 'minkowski'}



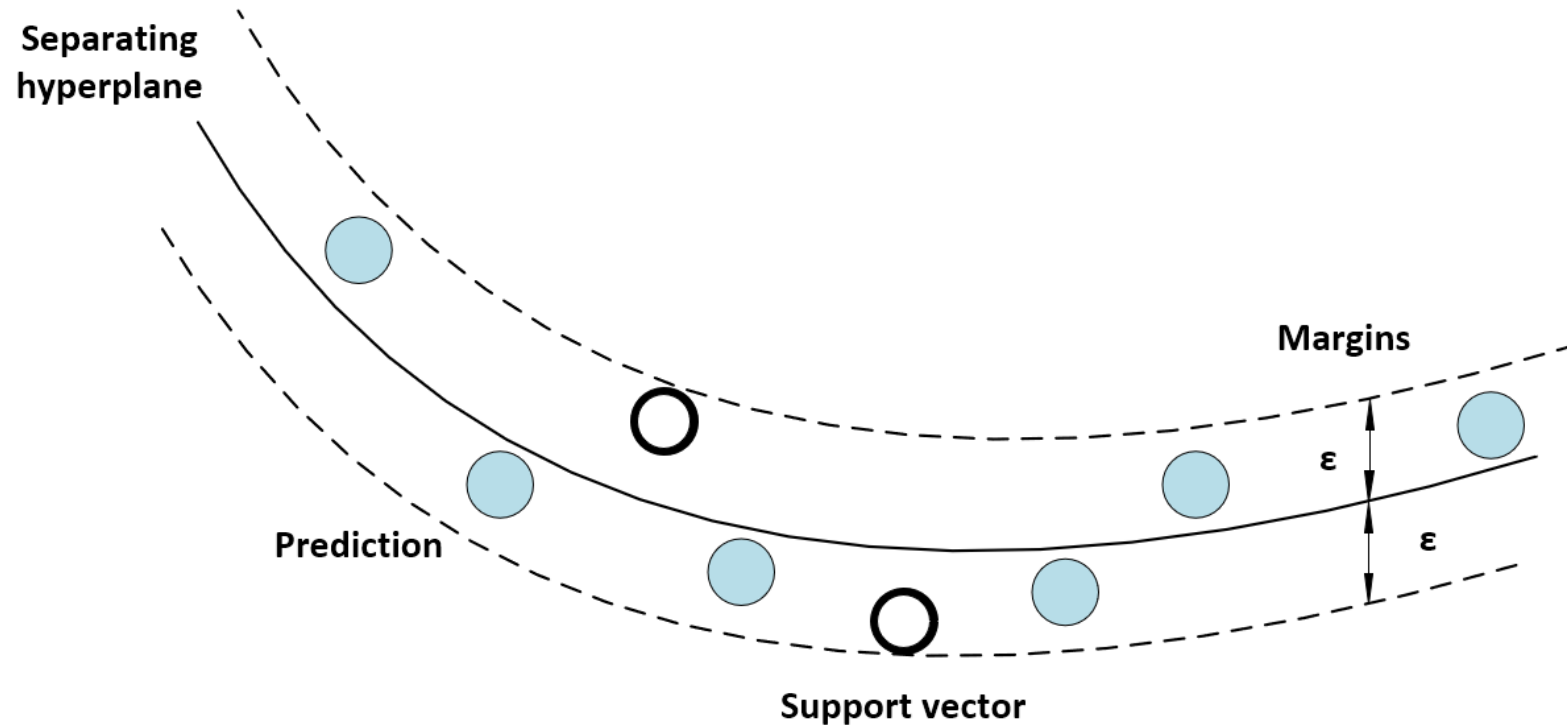


Decision Tree



- Hyper-parameters:
 - `max_depth`
 - `min_samples_split`
 - `min_samples_leaf`

Support Vector Regressor (SVR)



Support Vector Regressor (SVR)



- Hyper-parameters:
 - **kernel:** {'linear', 'poly', 'rbf', 'sigmoid', 'precomputed'}
 - **gamma:** {'scale', 'auto'}
 - **C** : Regularization



Multi-Layer Perceptron(MLP)

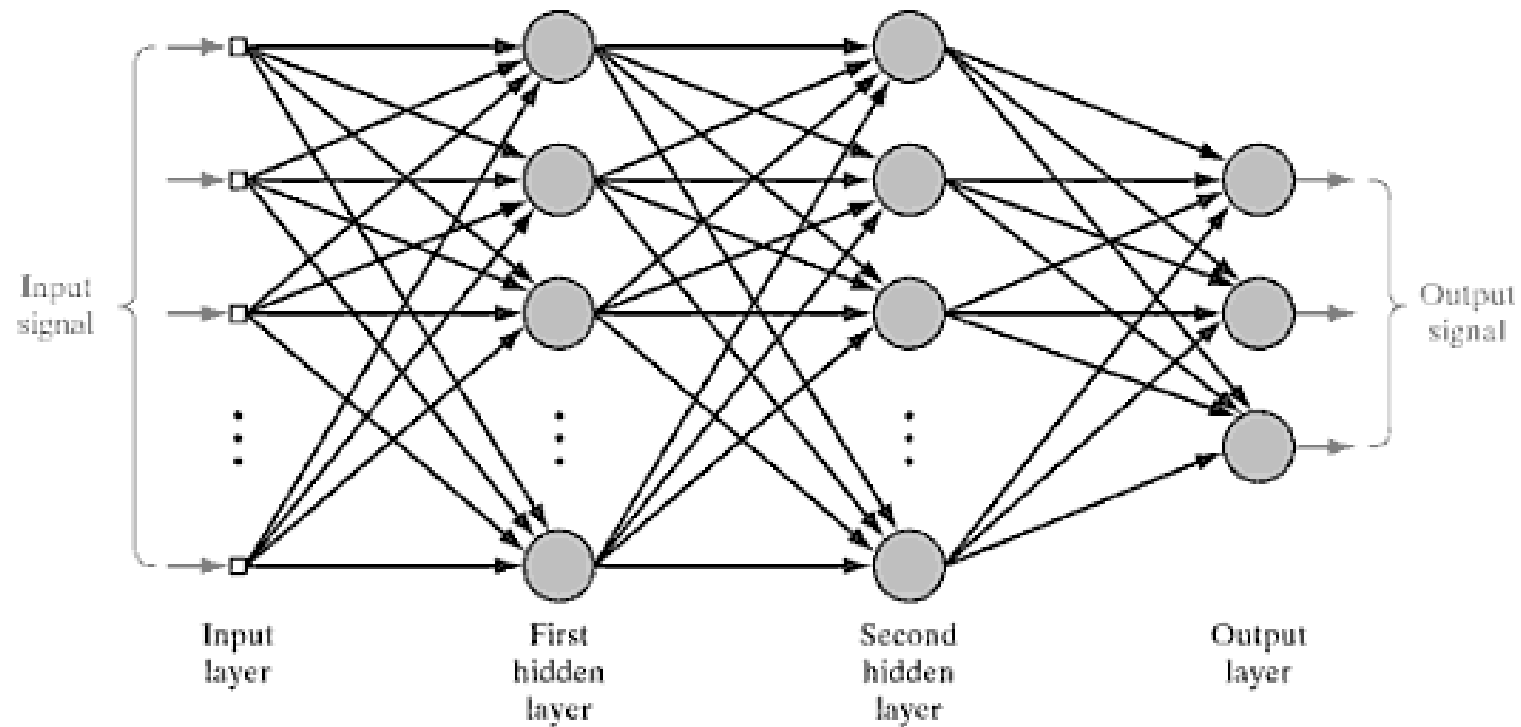


FIGURE 4.1 Architectural graph of a multilayer perceptron with two hidden layers.

Multi-Layer Perceptron (MLP)



- Hyper-parameters:
 - **hidden_layer_sizes** = (100,)
 - **activation**: {'identity', 'logistic', 'tanh', 'relu'}
 - **solver**: {'lbfgs', 'sgd', 'adam'}
 - **learning_rate**: {'constant', 'invscaling', 'adaptive'}
 - **max_iter**: None



Real example of a regressor



Let's get hands on! Open https://github.com/Dr-Carlos-Villasenor/TRSeminar/blob/main/TRS06_Reg_real_example.ipynb