

Capitulo BUAP de la Sociedad Mexicana de Inteligencia Artificial

Impartido por Act. Luis Ángel Alcántara Rosas

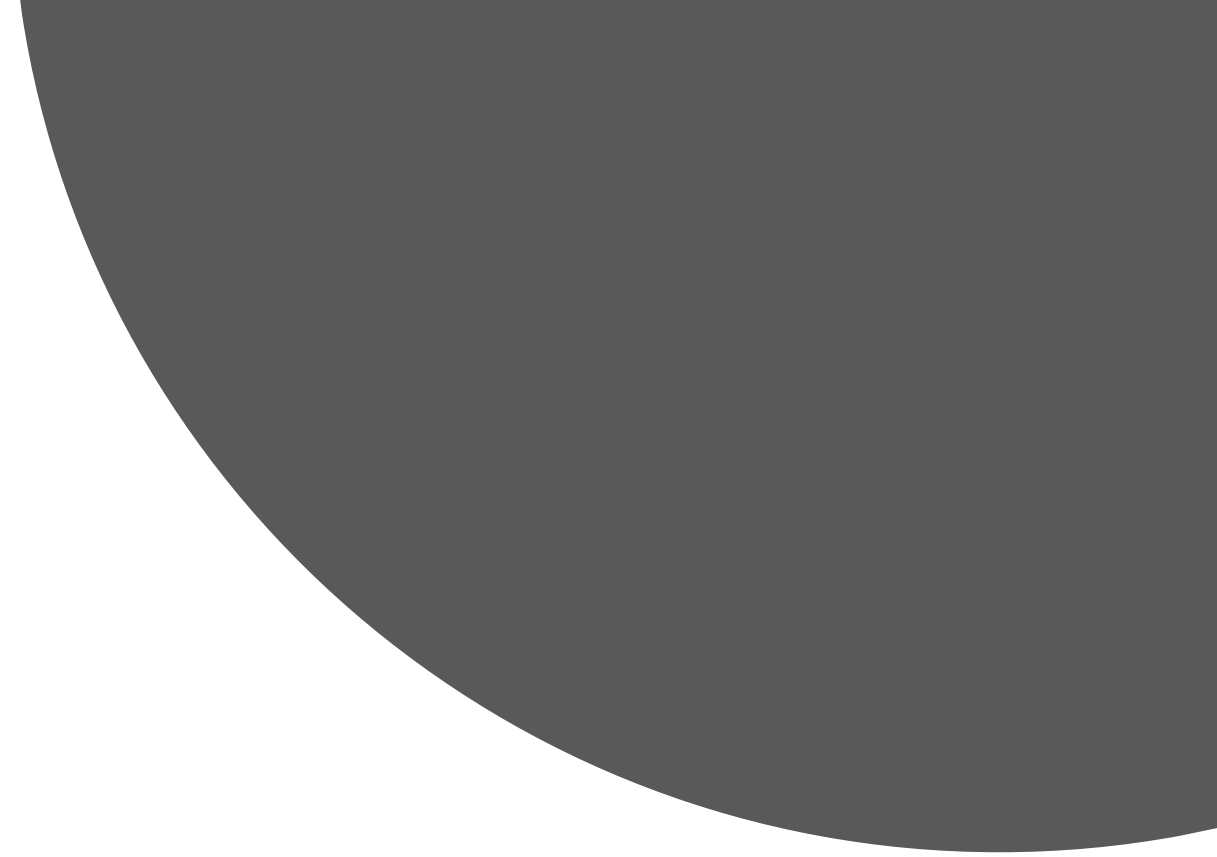
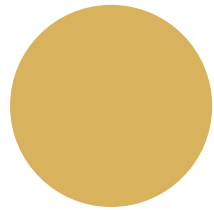
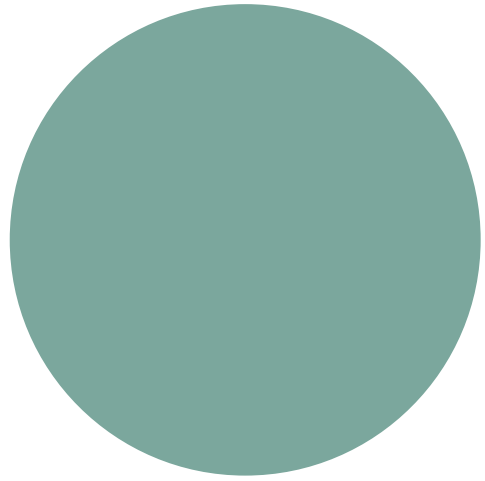
Taller de Machine Learning con



Enfoque a Aprendizaje Supervisado

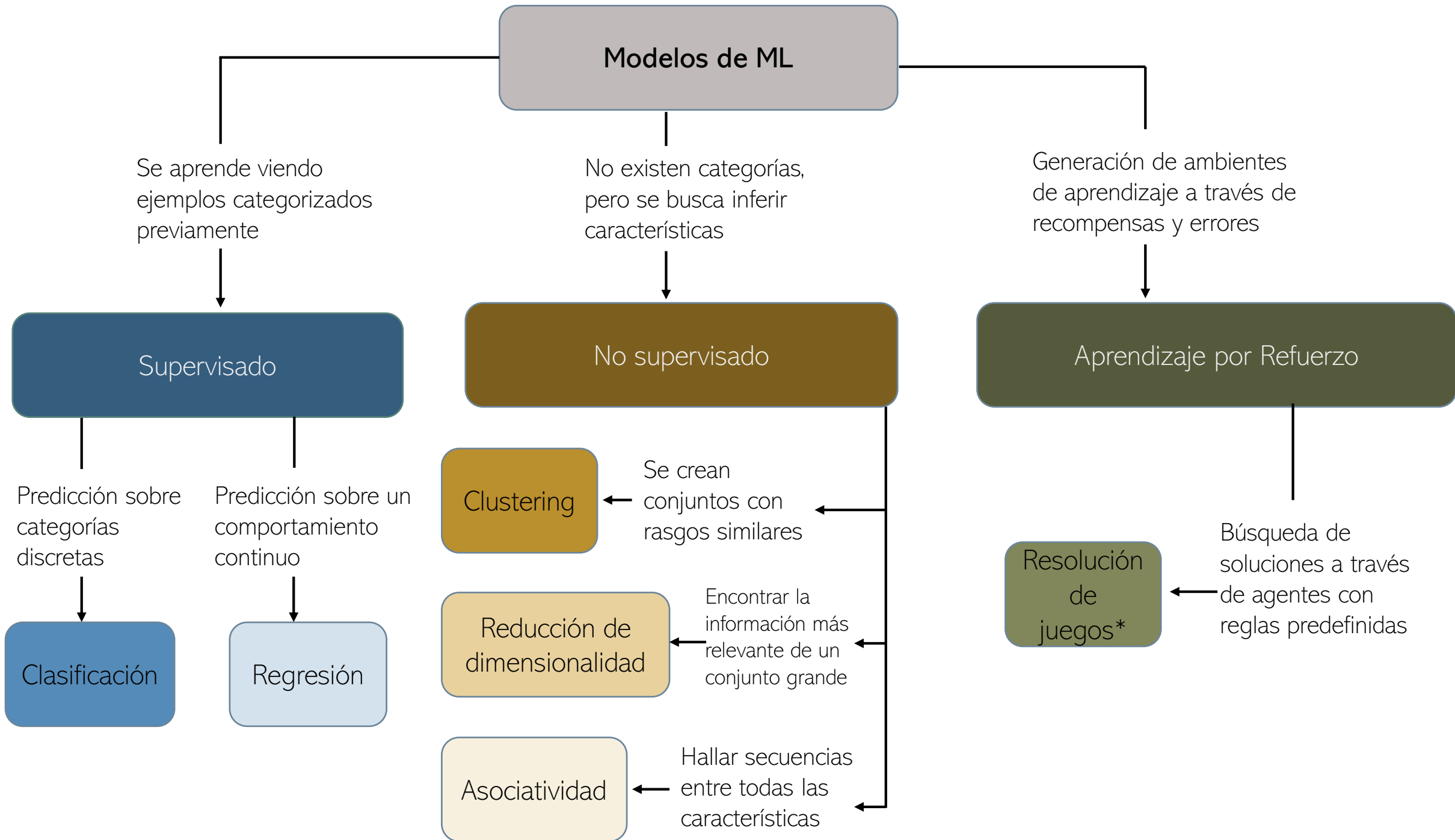
Índice

- **Bloque 1:** Conceptos básicos de Machine Learning
- **Bloque 2:** Selección de datos
- **Bloque 3:** Regresiones
- **Bloque 4:** K-Nearest Neighbors (K-NN)
- **Bloque 5:** Árboles de decisión
- **Bloque 6:** Comparación de modelos
- **Bloque 7*:** Introducción a las Redes Neuronales



Conceptos básicos de Machine Learning

BLOQUE 1





Supervised

- Linear Regression
- Neural Networks
- Decision Trees

Unsupervised

- K-Means
- PCA

Reinforcement Learning

- Q-Learning
- SARSA
- Deep Q Learning

Supervised Learning

Vector de Entrada (Inputs)

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

Dados por una base de datos

Parámetros del modelo

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_n \end{bmatrix}$$

Establecidos dentro del modelo
(Jugaremos con esto)

Modelo de ML elegido

$$f(\mathbf{x}, \boldsymbol{\theta}) = \bar{y} - \varepsilon = y$$

↓
Pronóstico realizado por el modelo

↓
Error entre pronóstico y valores reales

↓
Valores reales

Supervised Learning

```
graph TD; A[Supervised Learning] --> B[Clasificación (y discreta)]; A --> C[Regresión (y continua)]; B --> D[Grados de Riesgo Crediticio]; C --> E[Precios diarios del petróleo];
```

Clasificación (y discreta)

Grados de Riesgo Crediticio

1. Alto
2. Medio
3. Bajo



Regresión (y continua)

Precios diarios del petróleo

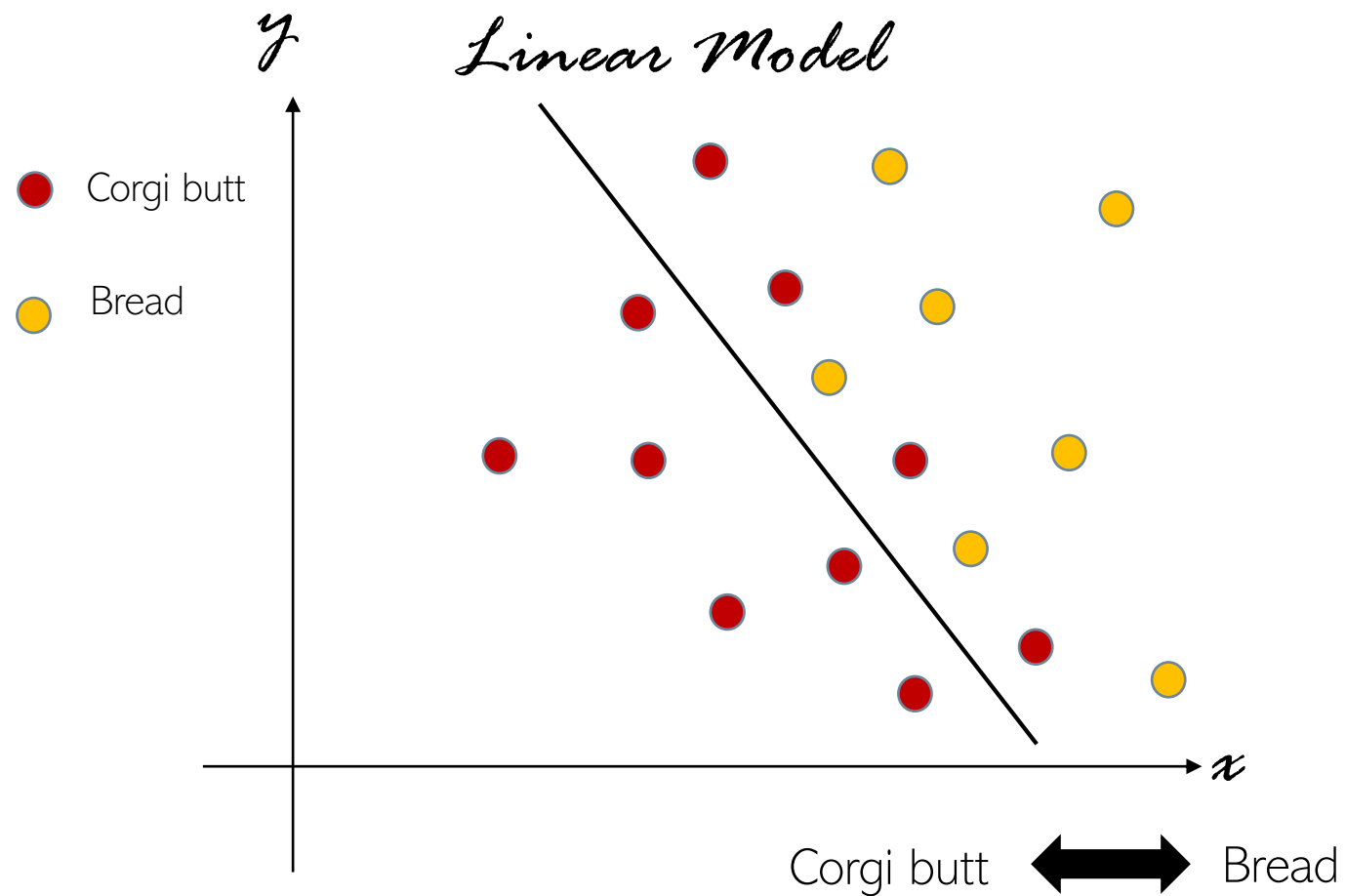
\$40.50
\$42.31
\$45.78
.....



Clasificadores



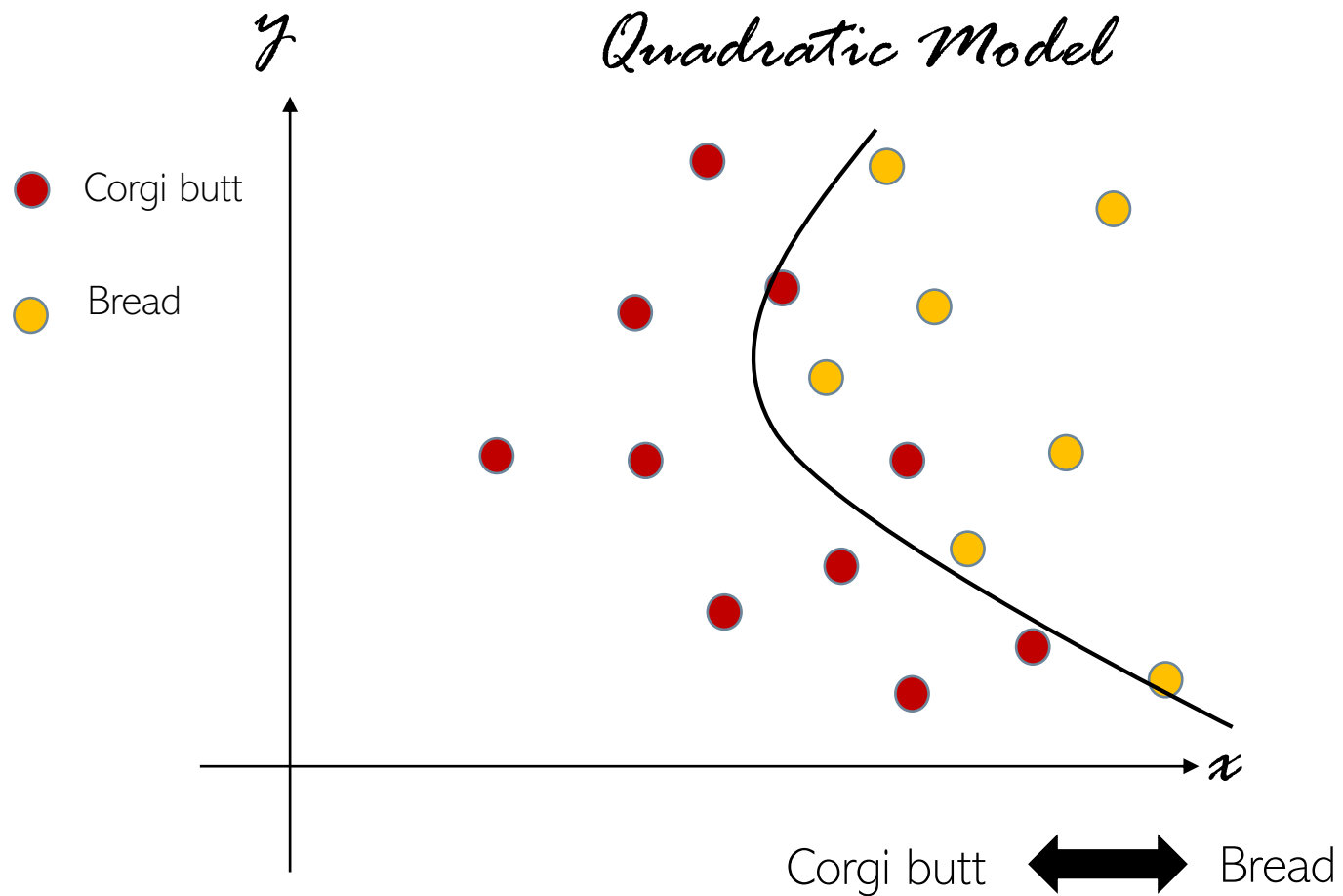
Corgi butt vs. Bread



Clasificadores



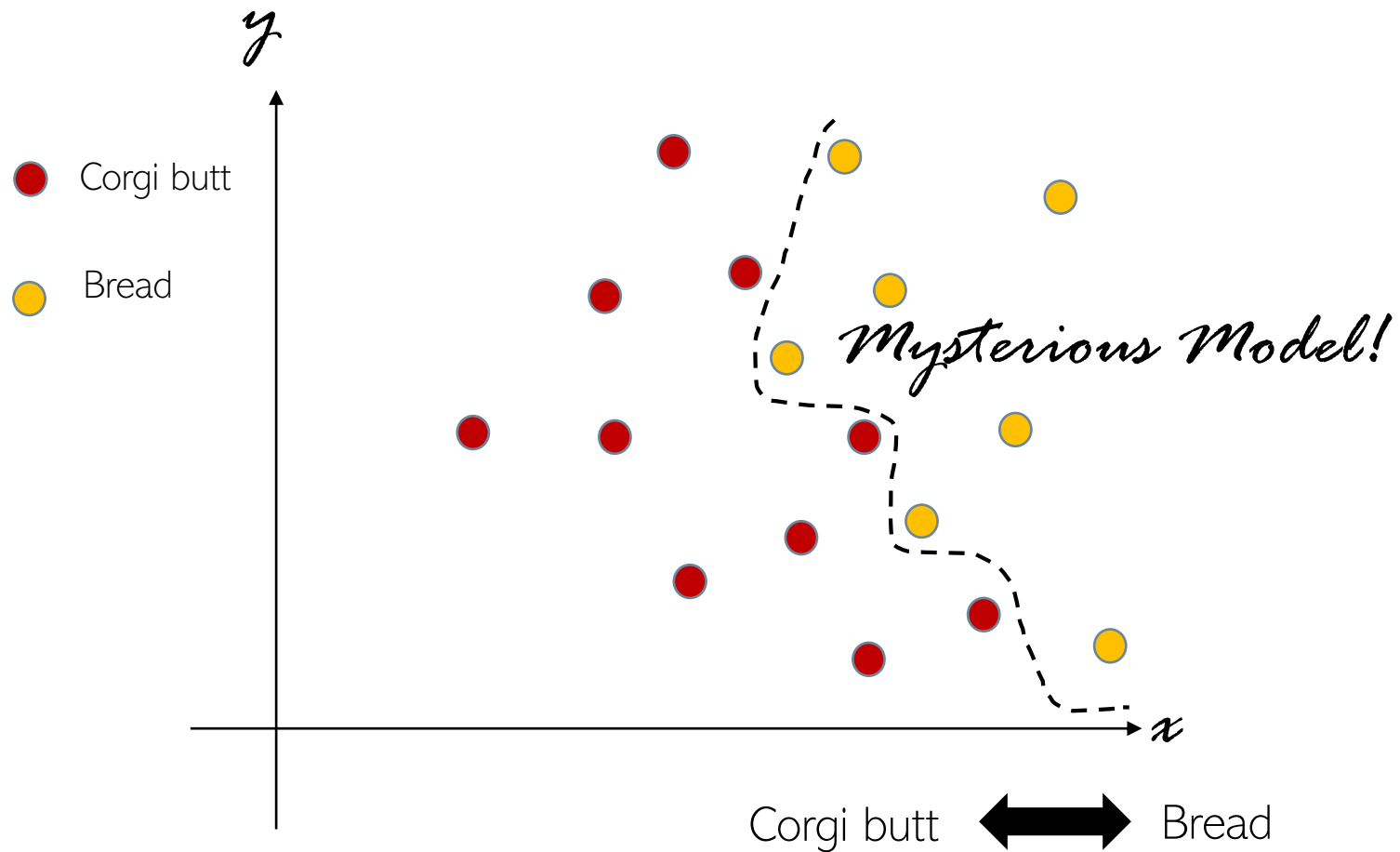
Corgi butt vs. Bread



Clasificadores



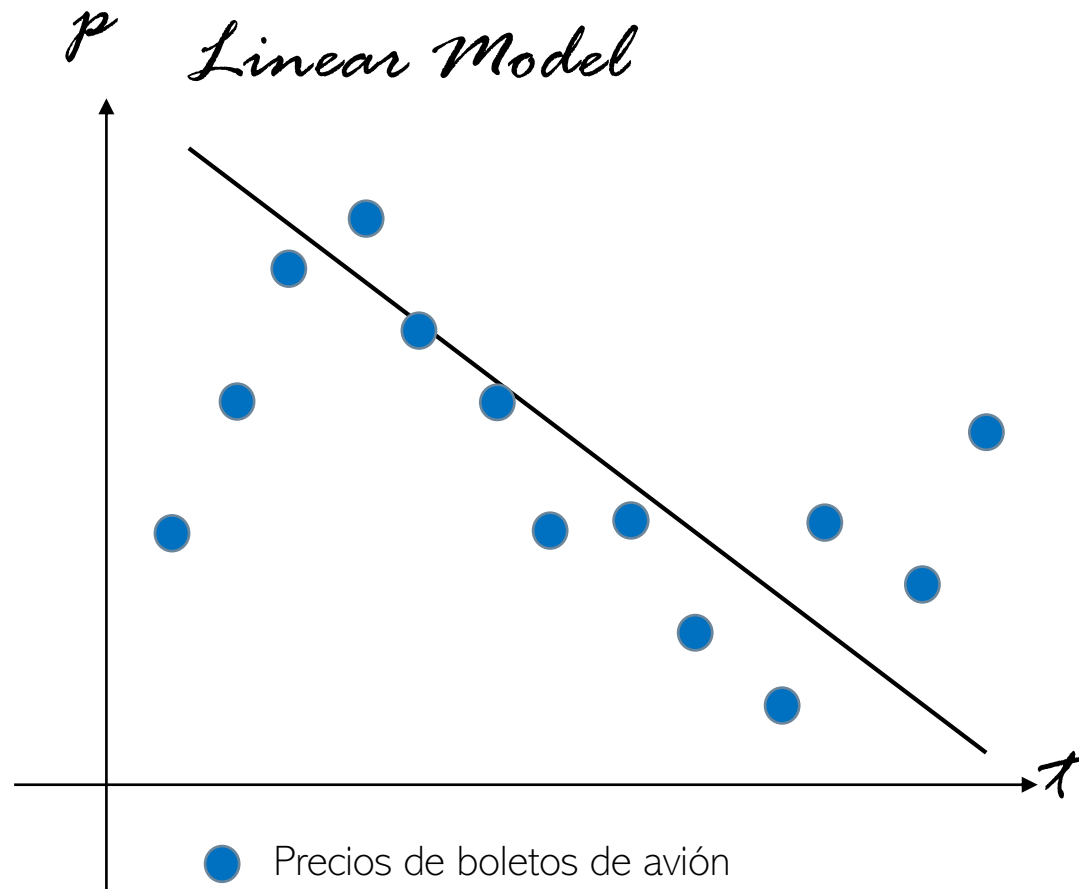
Corgi butt vs. Bread



Regresores



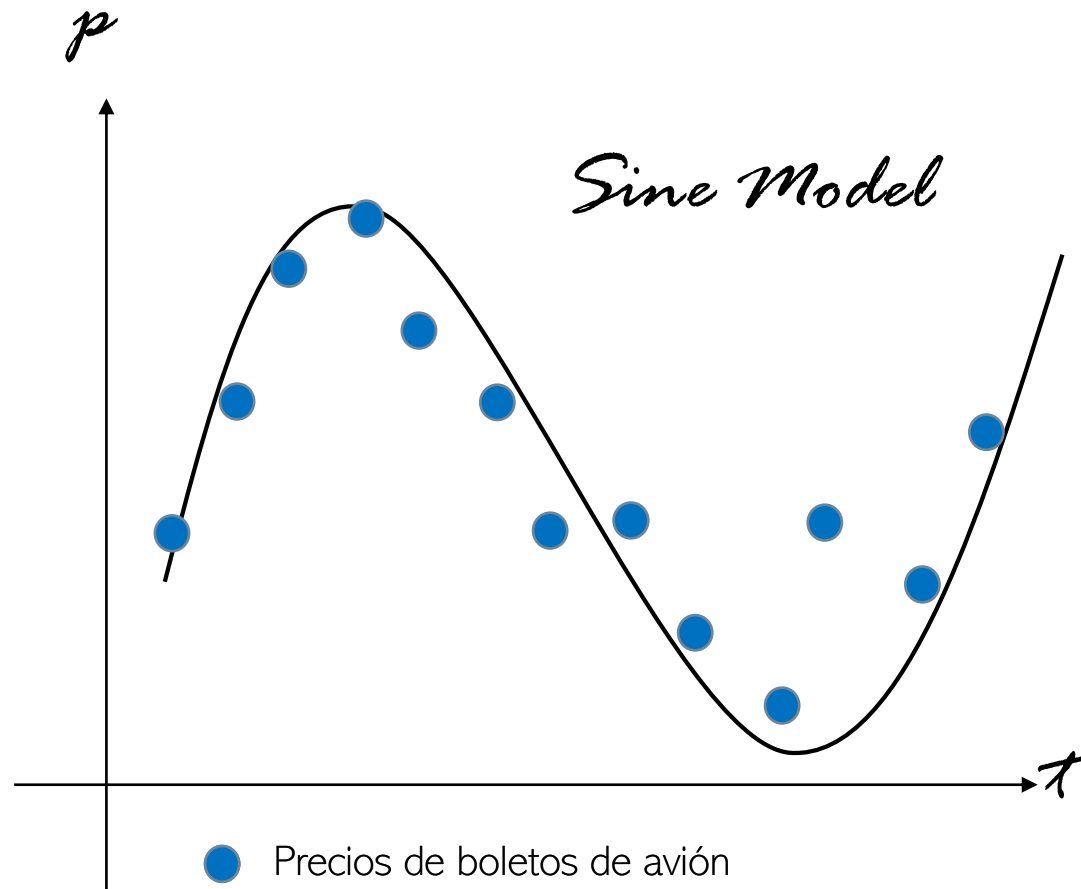
Cantidad de pasajeros a través del tiempo



Regresores



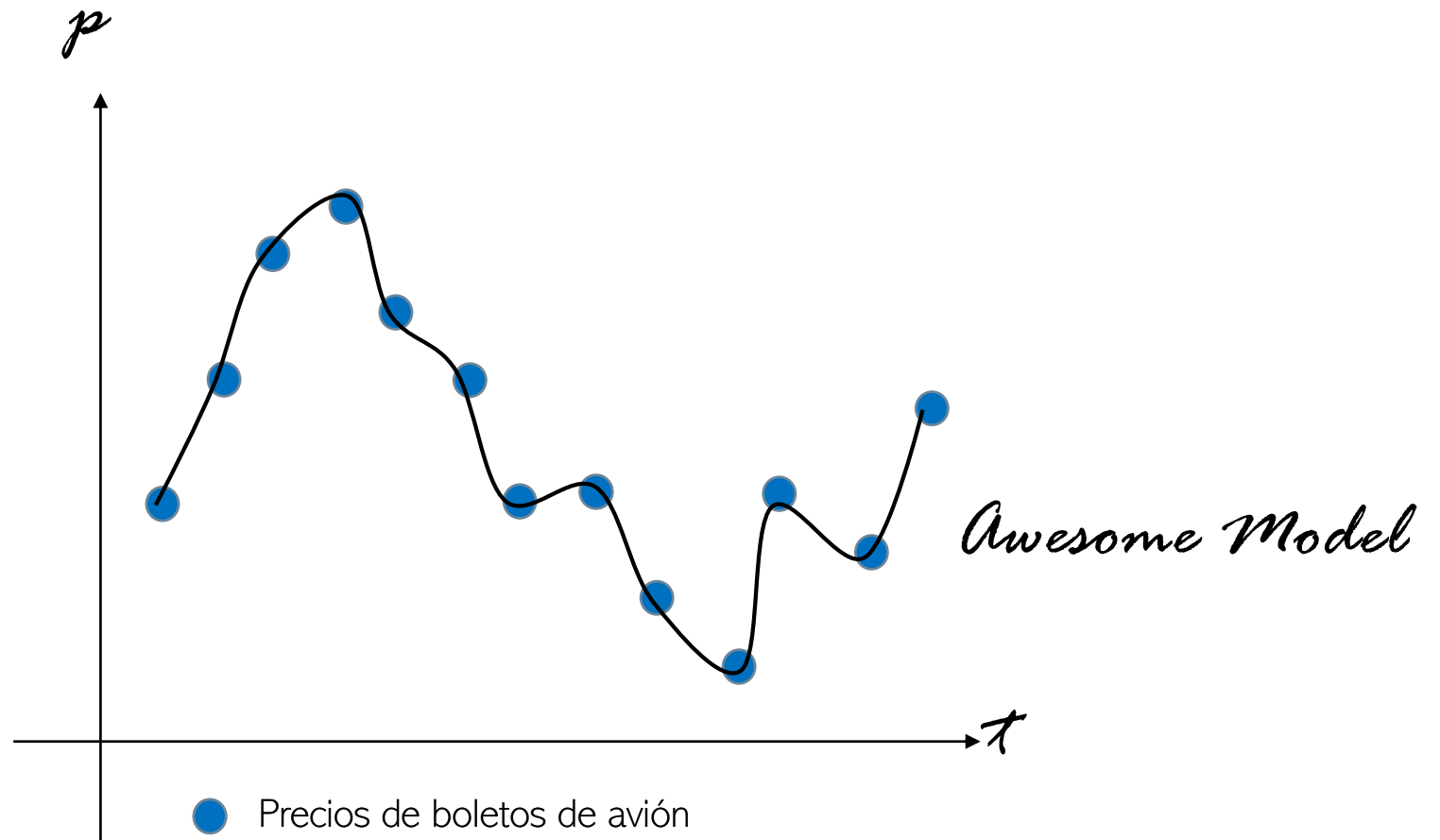
Cantidad de pasajeros a través del tiempo



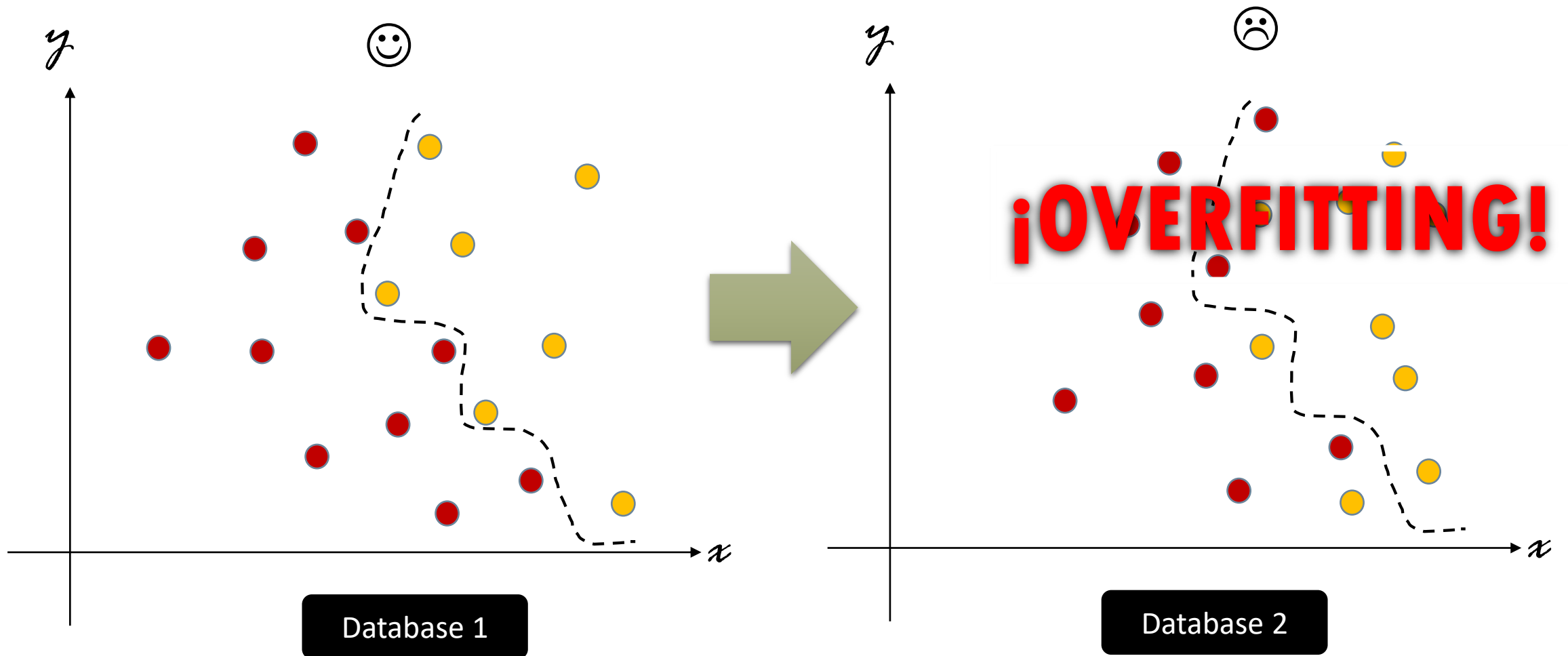
Regresores



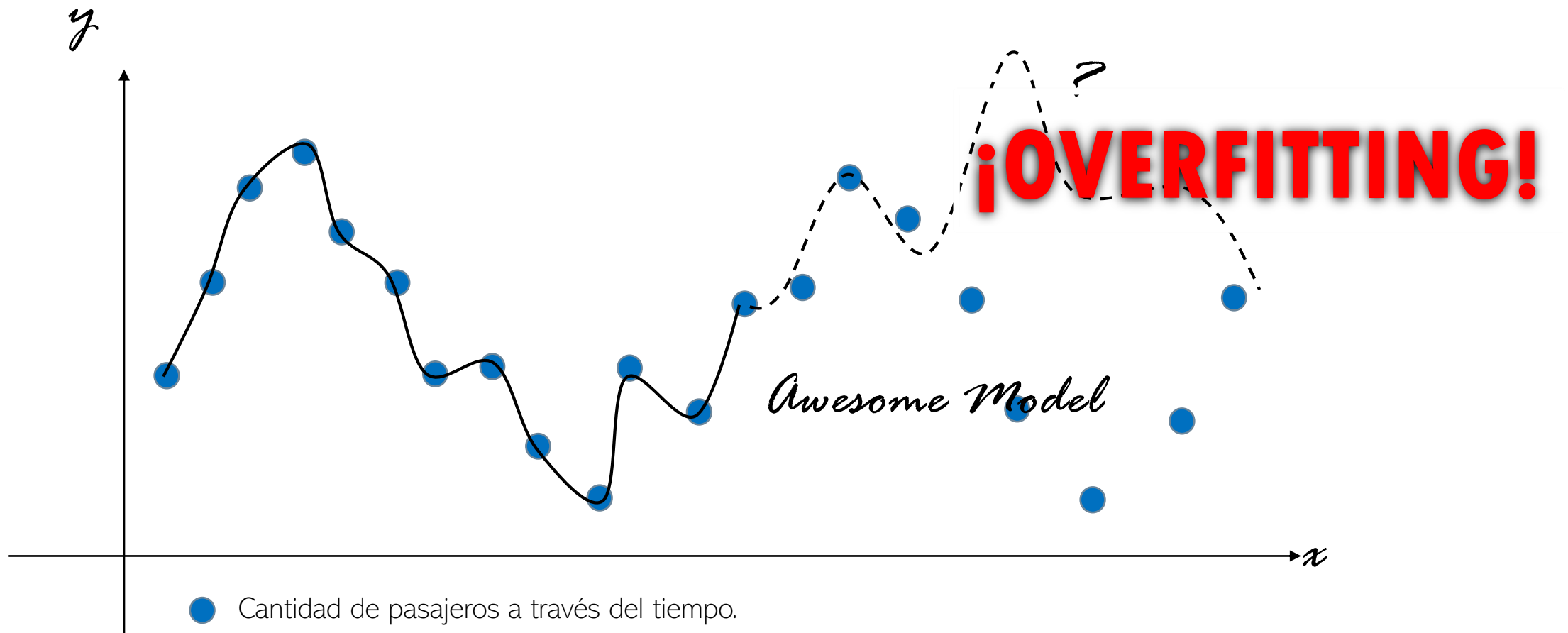
Cantidad de pasajeros a través del tiempo



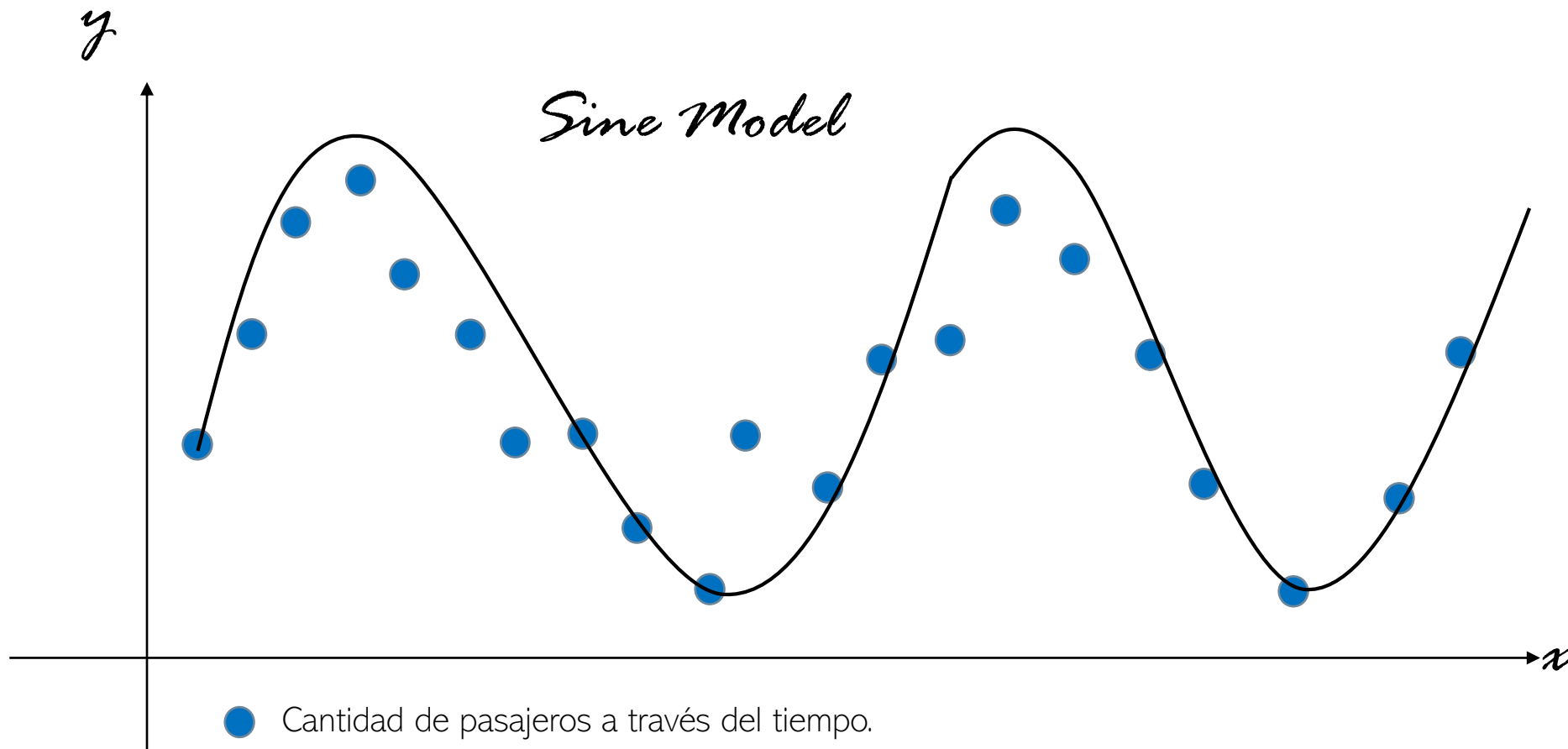
Clasificadores



Regresores



Regresores





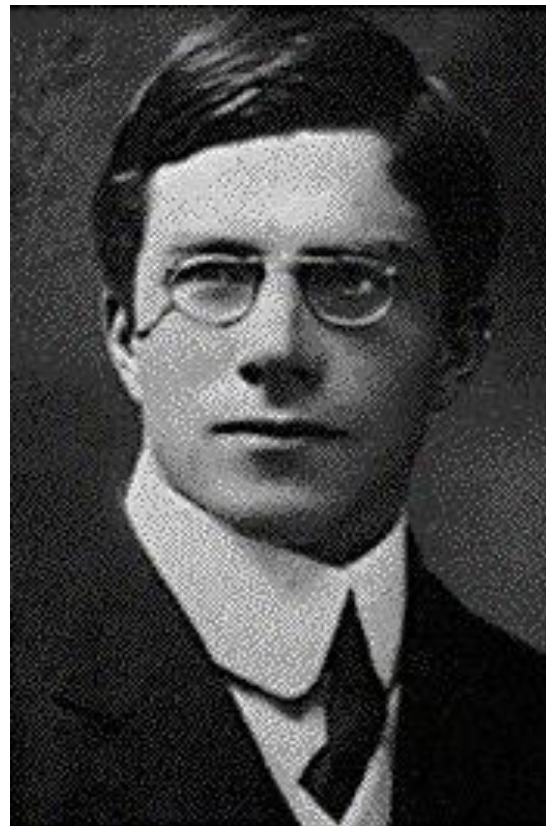
Selección de datos

BLOQUE 2

Caso Iris (1936)



Edgar Anderson



Ronald Fischer

Caso Iris



Iris setosa

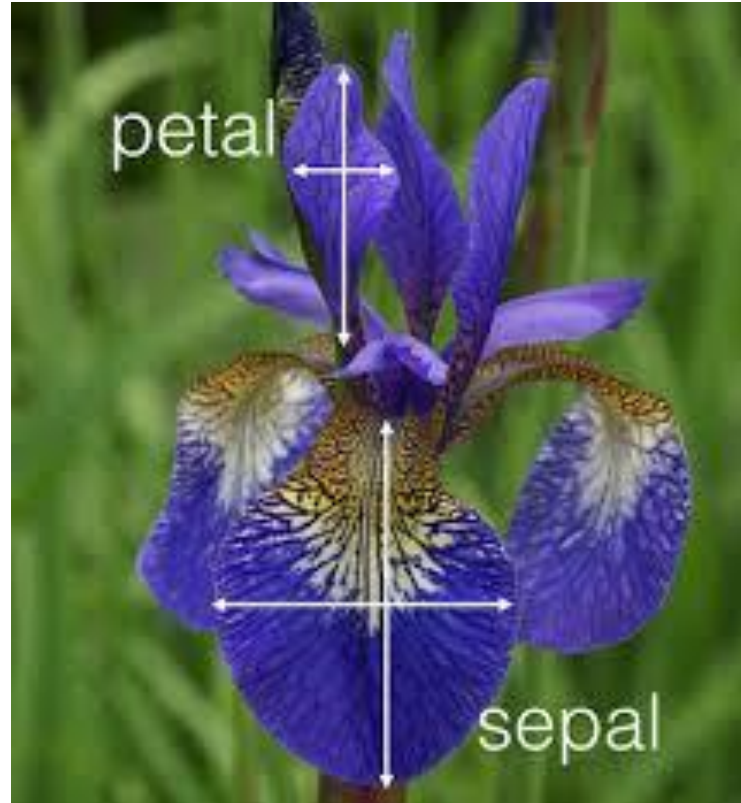


Iris versicolor



Iris virginica

Caso Iris



¡Abrir su plataforma R!

Aleatorización



Aleatorizar los datos permite eliminar sesgos de los curadores de datos originales y genera un campo fértil para encontrar patrones. **NOTA: NO EN TODOS LOS CASOS ES LO MEJOR!!!**

Balance de datos



Se tiene un problema de clasificación binaria. Se clasifica 0 cuando hay fracaso o 1 cuando hay éxito. En el 95% de la base de datos hay 1's y para el 5% restante 0's. El modelo de ML registra un éxito del 95%.... ¿Qué está pasando aquí?

Train & test datasets

Al proceso anterior comúnmente se le llama **preprocesamiento** y en algunas ocasiones puede ser más exhaustivo. Una vez teniendo una base de datos adecuada se procede a ajustar el modelo de ML

Train



Se proceden a estimar los parámetros θ del modelo de ML $f(x, \theta)$ para disminuir el error ε .
(Posteriormente se verán métricas específicas para esto)

Test



Se verifica que con los parámetros θ encontrados la variación de \bar{y}_{train} respecto a \bar{y}_{test} sea “*acceptable*”. En caso contrario, entrenar nuevamente, cambiar de modelo de ML o replantear el problema.

Train & test datasets

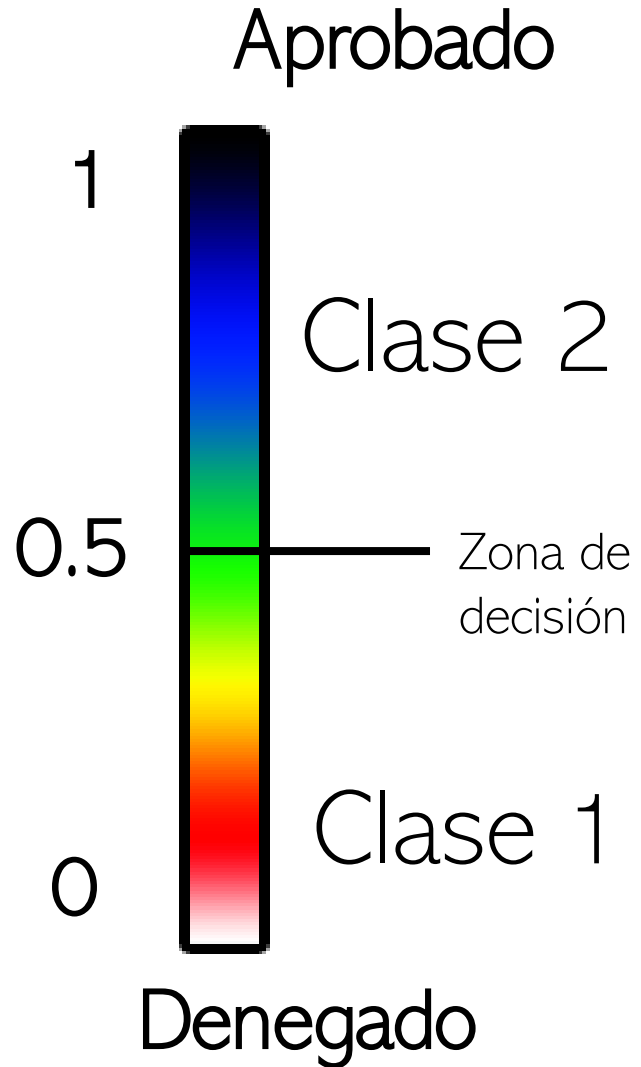


Aspectos importantes de la selección de datos:

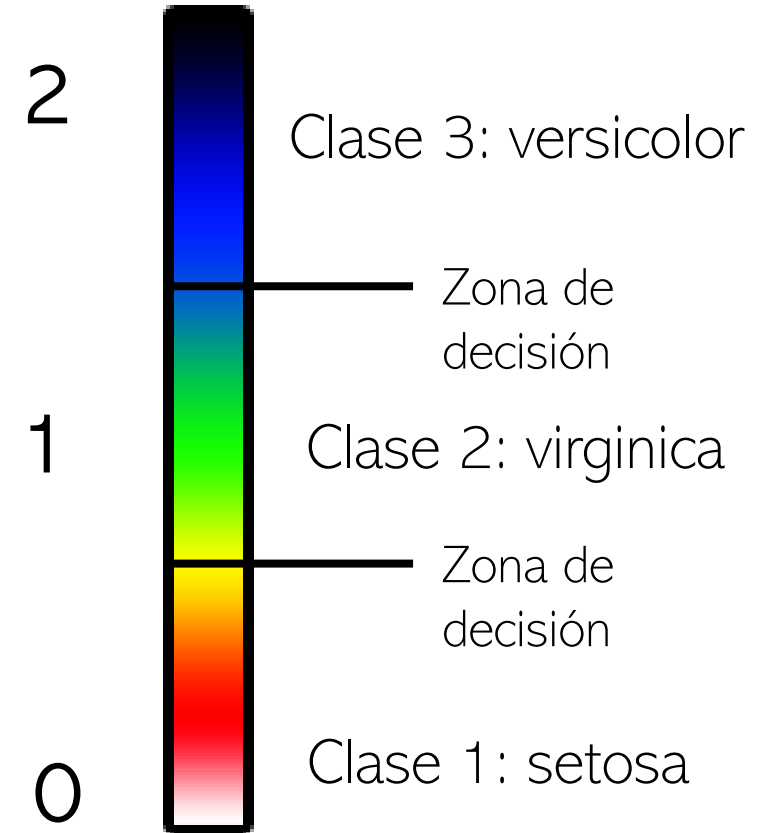
- Deben ser independientes, tanto el conjunto de train como de test.
- Tomar en cuenta siempre el contexto de cada problema. En especial ver que en casos de regresión no se puede separar tan arbitrariamente la información.
- A veces es conveniente tener un tercer conjunto de *validación* para confirmar el desempeño observado en test.
- Si se tiene la oportunidad de tener más datos, jamás desperdiciarla.

Etiquetas

Binario



Multiclase



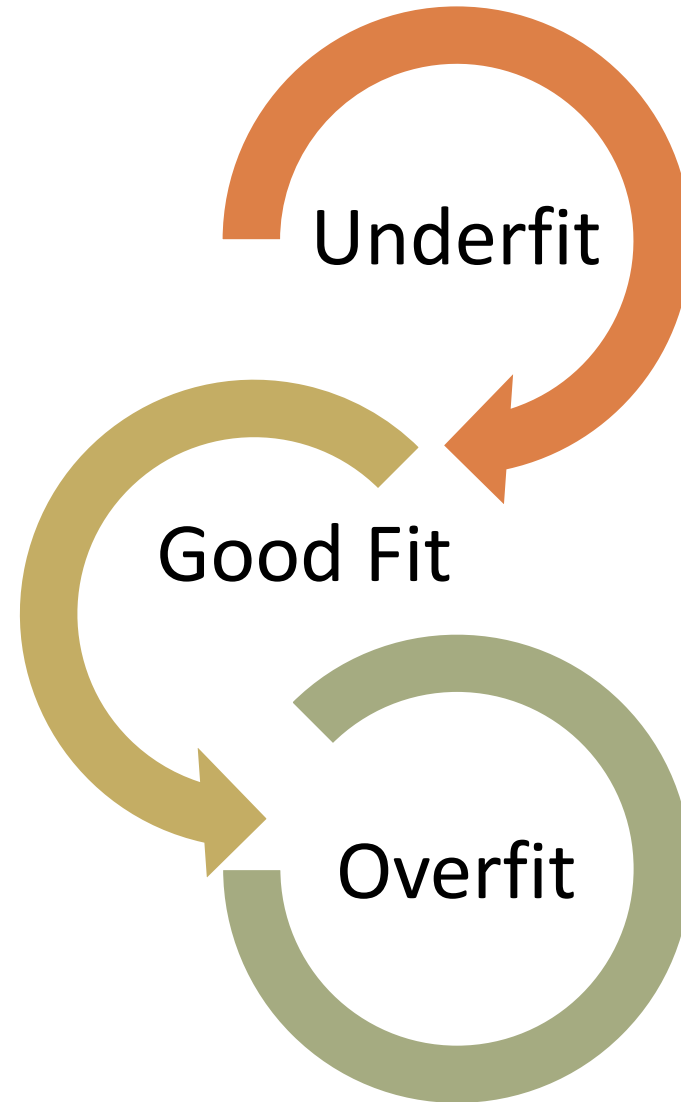
El paradigma del Clasificador Aleatorio



Si se tiene un clasificador binario, y se desea tener un sistema de clasificación, se puede tratar de modelar mediante el lanzamiento de una moneda. En el 50% de los casos nos dará una categoría e igualmente con 50% de probabilidad se generará la otra. **Un clasificador apropiado debería ser mejor que un lanzamiento de moneda.**

Underfit vs Overfit

Problema que se
nota durante train



Underfit

Good Fit

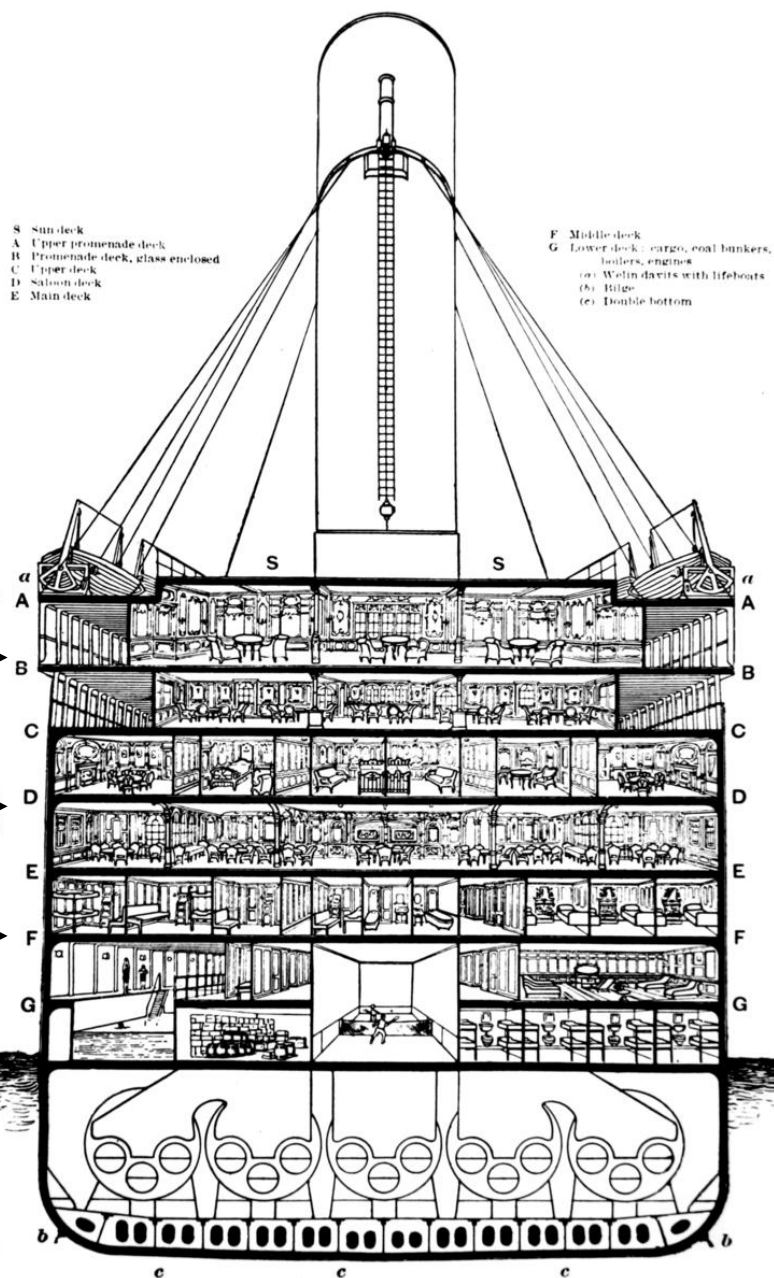
Overfit

Problema que se
nota durante test

Titanic Database



- RMS Titanic: Construido de 1909 a 1912 y hundido el 15 de abril de 1912.
- Sus zonas estaban divididas en 3 clases según el status económico de los pasajeros.
- Hizo 3 paradas antes de viajar de Europa a Nueva York en Estados Unidos:
 1. Southampton. (Gran Bretaña)
 2. Cherburgo (Francia)
 3. Queenstown (Irlanda)
- El estricto protocolo establecía que:
“mujeres y niños primero”
- El barco estaba construido con ciertas estructuras que dividían cabinas y zonas.



Cubierta B: 1° y
2° clase.

Cubierta D: 2° y
3° clase.

Cubierta F: 2° y
3° clase.

Sala de
Maquinas

Cubierta A: 1° clase.

Cubierta C: 1°,
2° y 3° clase

Cubierta E: 3°
clase.

Cubierta G: 2° y
3° clase.

Titanic Database



Titanic Database

survival - Survival (0 = No; 1 = Yes)

Pclass - Passenger Class (1 = 1st; 2 = 2nd; 3 = 3rd)

Name - Name

sex - Sex

age - Age

sibsp - Number of Siblings/Spouses Aboard

parch - Number of Parents/Children Aboard

ticket - Ticket Number

fare - Passenger Fare

cabin - Cabin

embarked - Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)

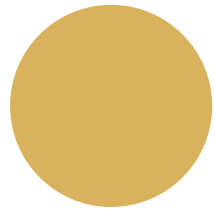
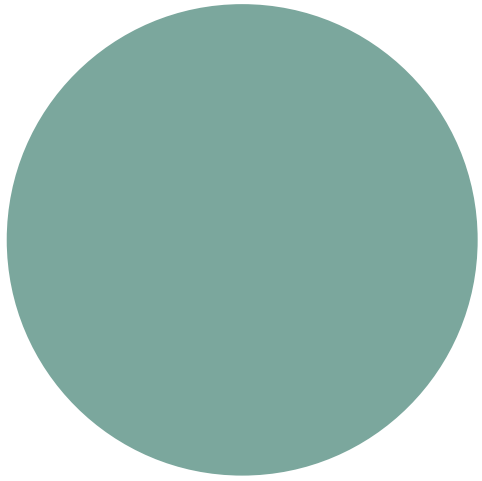
Titanic Database

Algo extra de contexto....

La base de datos a utilizar viene de un popular concurso online de la página Kaggle. Tiene dividida su base en 2: train y test. Se espera que con la base de train se *entrene y pruebe* el modelo de ML para que posteriormente con la base de test se califique en el concurso.

El reto....

Con lo aprendido en este curso, diseña tu propio modelo de ML y somételo a concurso en Kaggle!!!



Regresiones

BLOQUE 3

Regresión Lineal

Dentro del paradigma de aprendizaje supervisado se tiene la siguiente estructura:

$$f(\mathbf{x}, \boldsymbol{\theta}) = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n$$

Recordar: El vector \mathbf{x} es dado por la base de datos y $\boldsymbol{\theta}$ son los parámetros a ajustar.

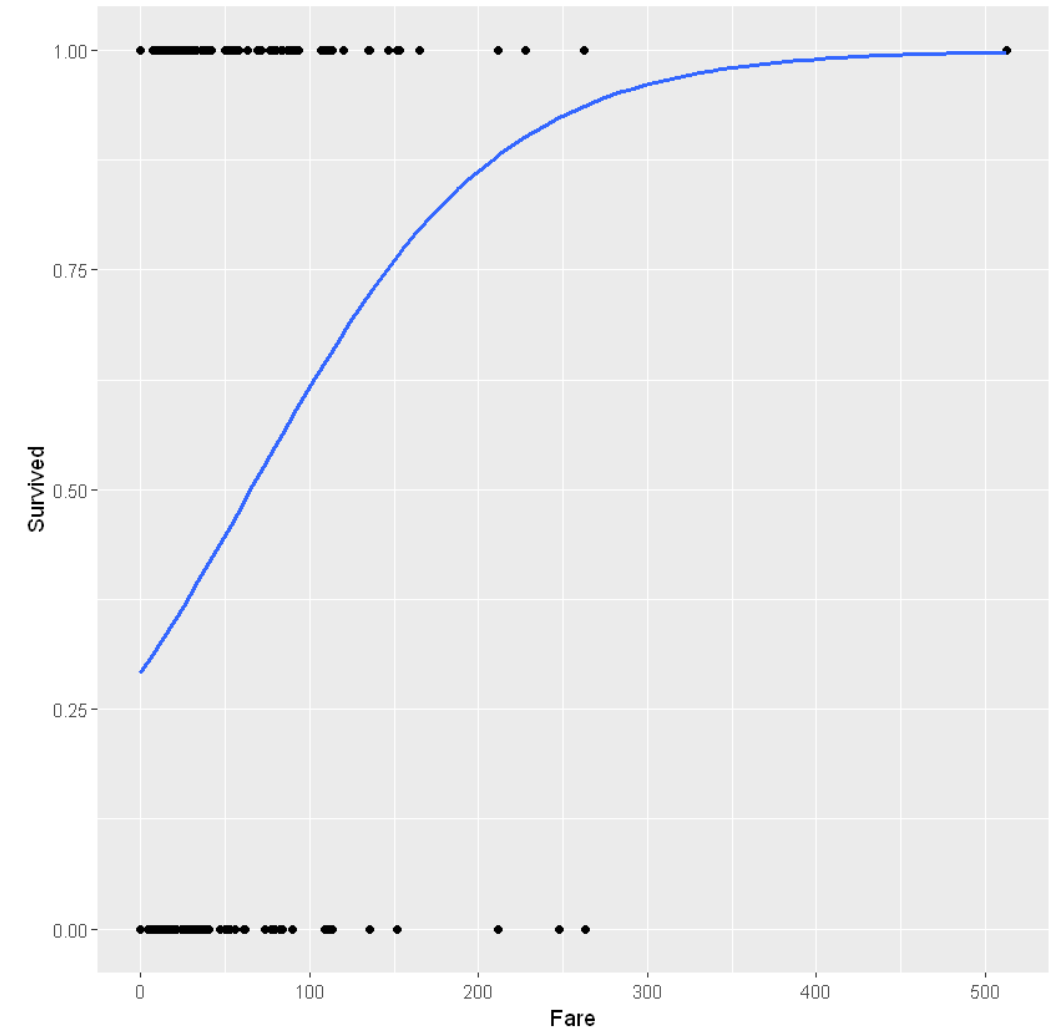
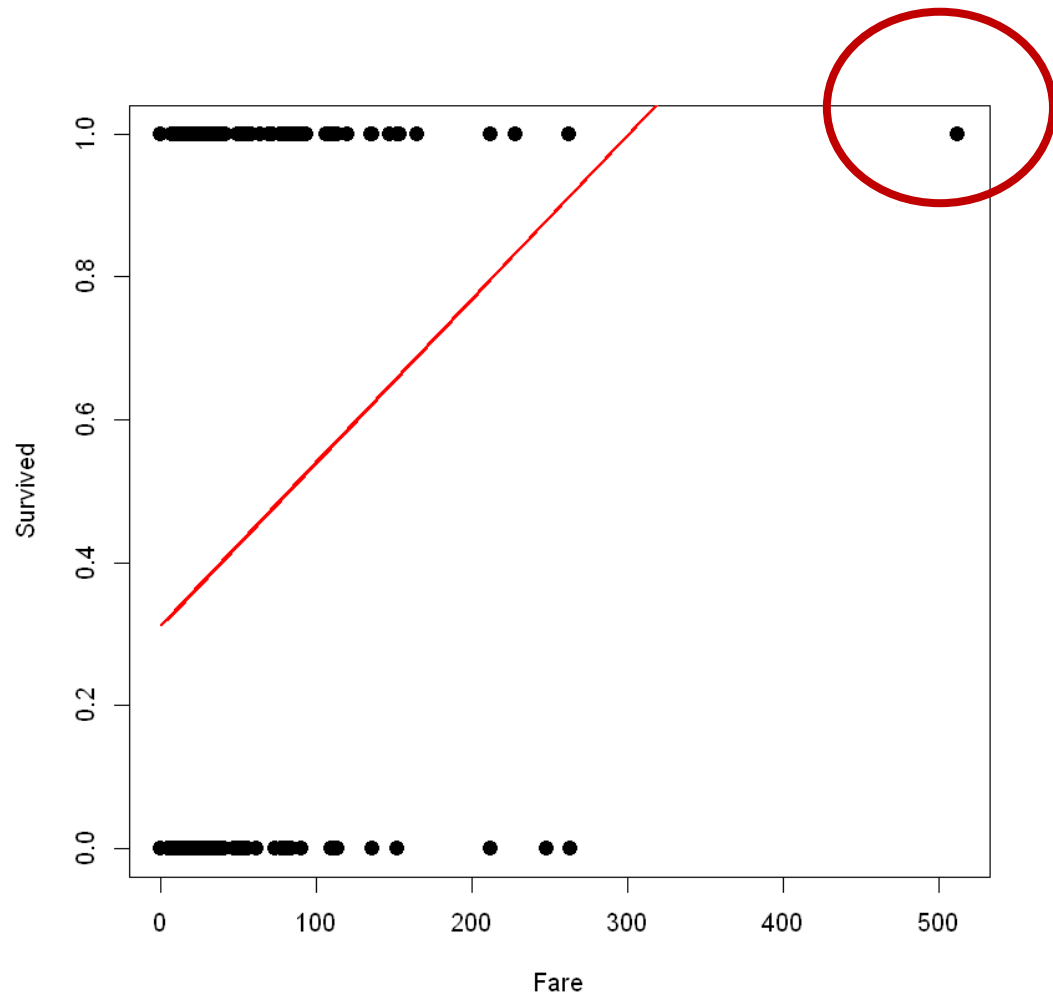
Regresión Logística

Un modelo lineal no se encuentra acotado en un rango entre 0 y 1 que le permita actuar como modelo de probabilidad. Generalmente para clasificación binaria se utiliza una variación de este modelo conocido como regresión logística y dentro del paradigma de aprendizaje supervisado se tiene la siguiente forma:

$$f(\mathbf{x}, \boldsymbol{\theta}) = \frac{1}{1 + e^{-(\theta_0 + \theta_1 x_1 + \dots + \theta_n x_n)}}$$

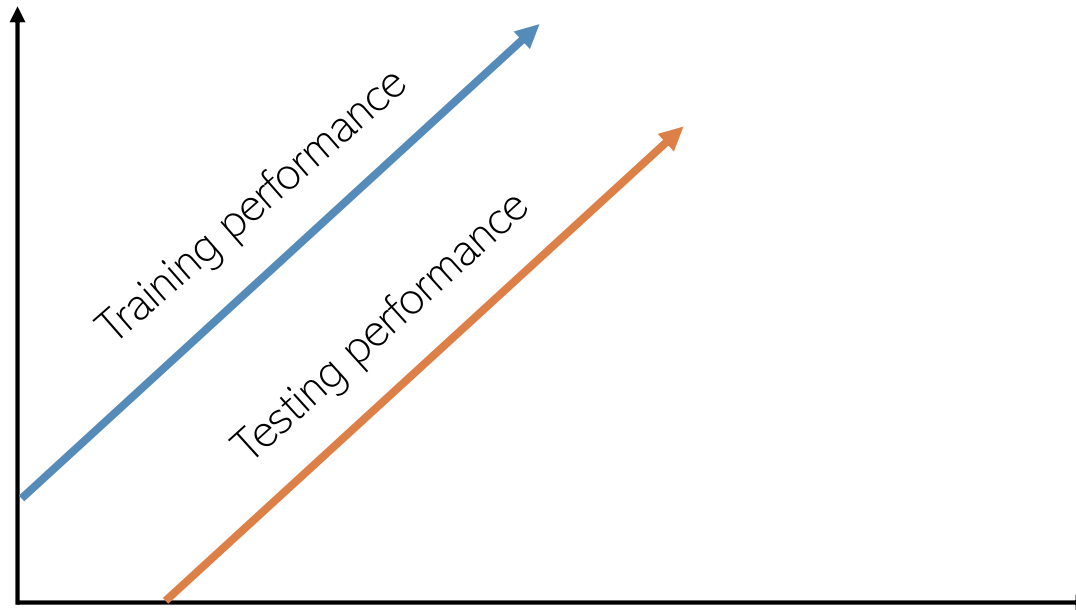
Para un problema de clasificación usando regresiones son de interés las probabilidades. Deseamos que $f(\mathbf{x}, \boldsymbol{\theta})$ nos de una probabilidad de que un elemento de \mathbf{x} pertenece a una clase. En la clasificación binaria la clase **C1** puede ser dada por 0 y la **C2** por 1

R. Logística vs R. Lineal



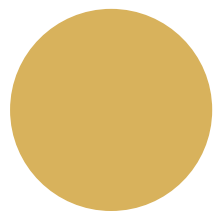
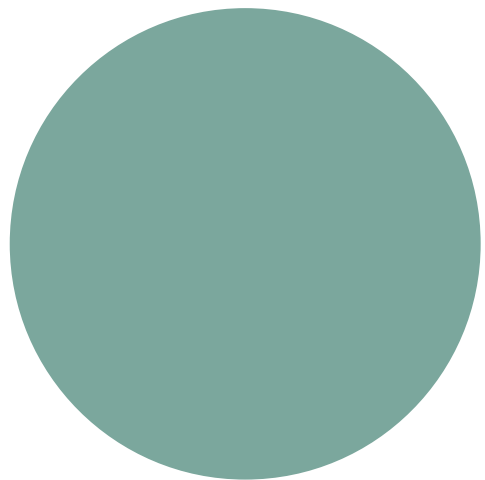
Train & Test performance

Generalmente existe una diferencia en el desempeño de la base de entrenamiento y prueba. Es decir, el modelo de ML al entrenarse tendrá mejores resultados en todas sus métricas que al probarse.



Ventajas y desventajas

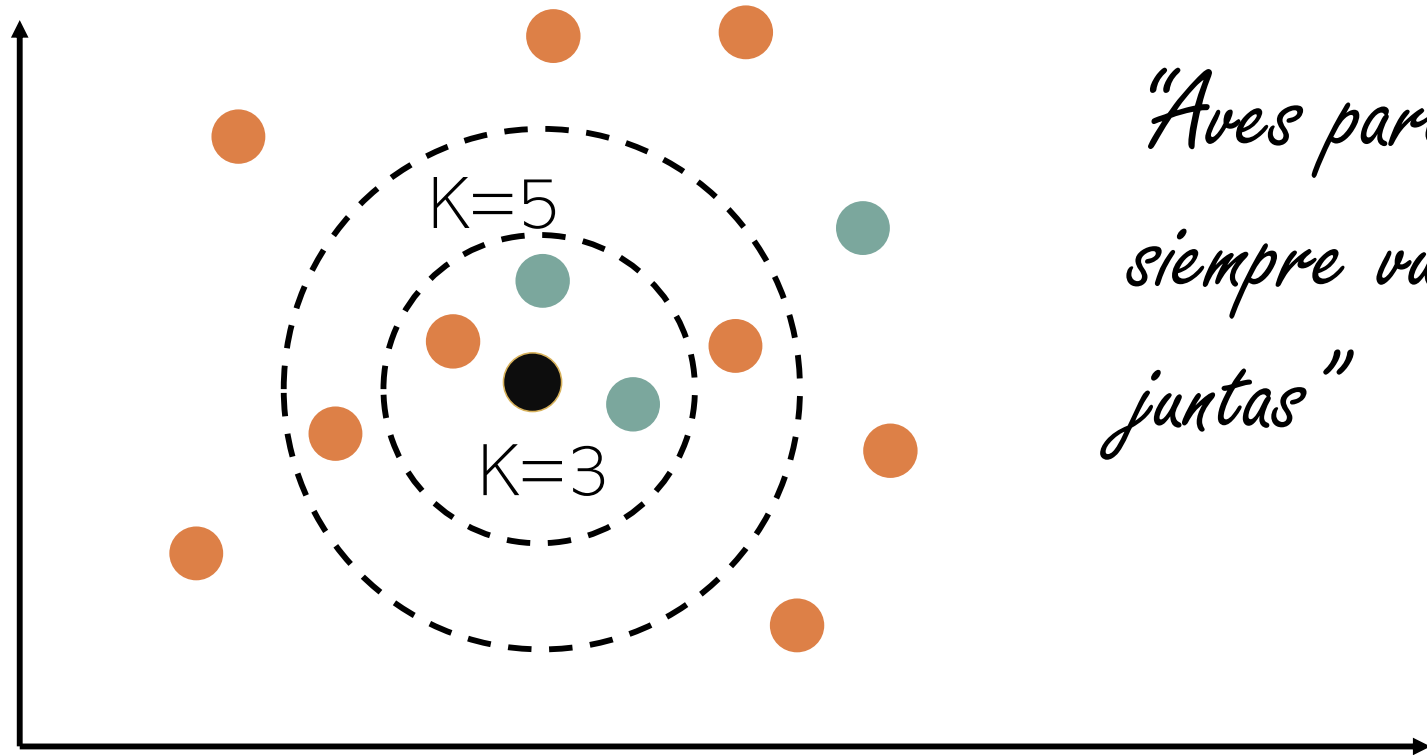
- Sencillo y rápido.
- La dimensionalidad no es un problema para el algoritmo.
- Prácticamente no hay *hiperparámetros* que cambiar
- Únicamente puede utilizarse para clasificación binaria.
- No va a ser el más potente disponible.



K Nearest Neighbors (K-NN)

BLOQUE 4

Funcionamiento



$$D(\mathbf{x}, \mathbf{y}) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$$

Ventajas y desventajas

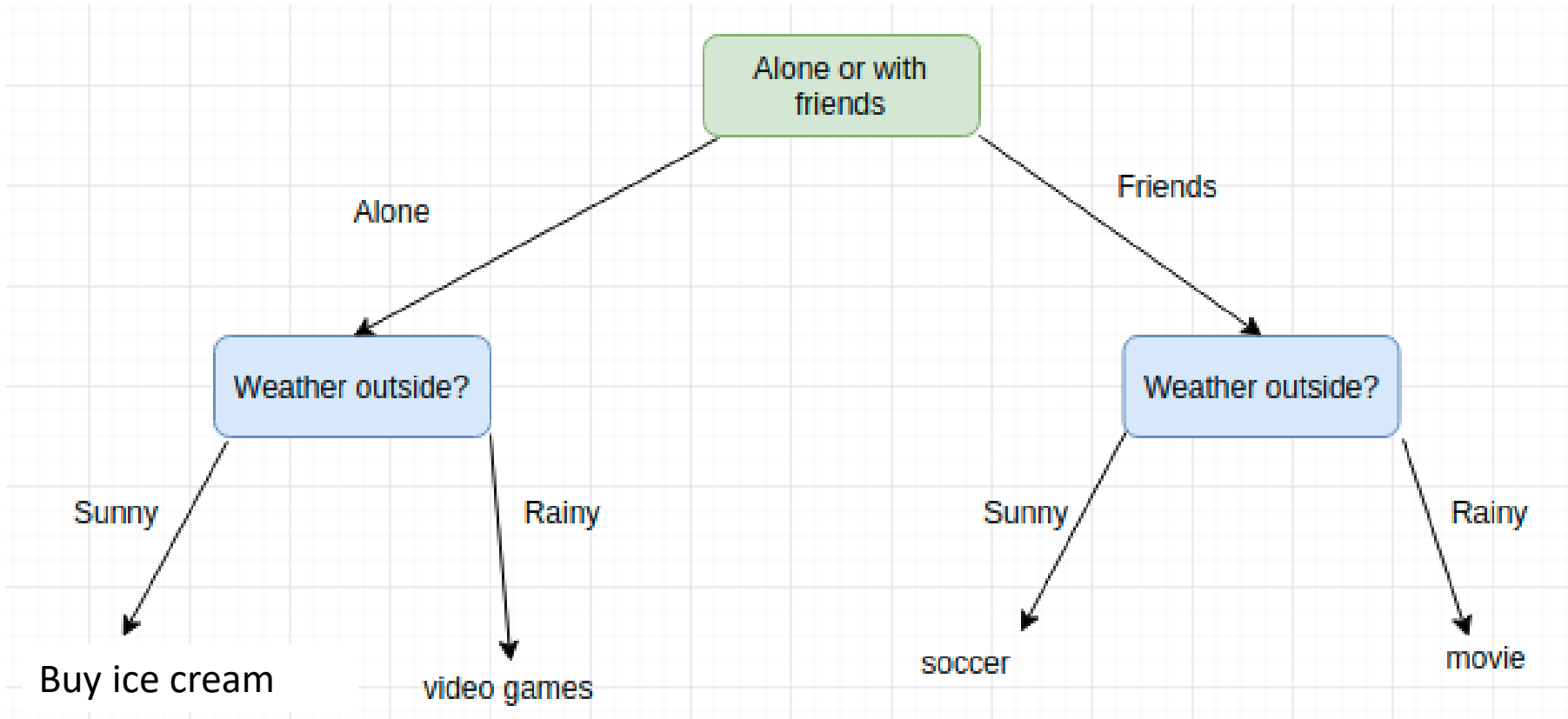
- No es necesario establecer más *hiperparámetros* que la k o el tipo de **distancia** que usar.
- En bases de datos cortas y con poca dimensionalidad es veloz.
- Si la dimensionalidad es alta (x_i) consume más memoria se hace costoso computacionalmente para guardar las distancias.
- En algunas ocasiones es necesario escalar los datos para que la distancia euclidiana pueda hacer mejor su trabajo.
- Hay que explorar para ver qué k es mejor



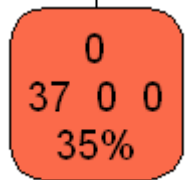
Árboles de decisión

BLOQUE 5

Funcionamiento



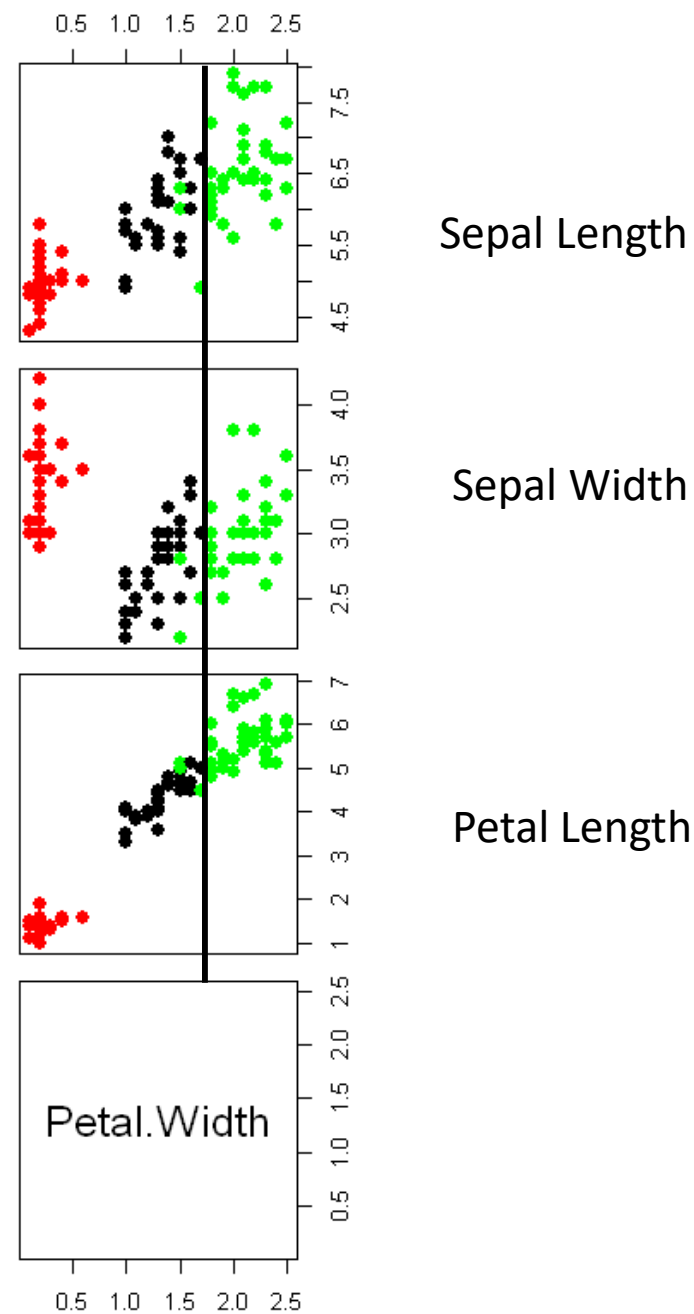
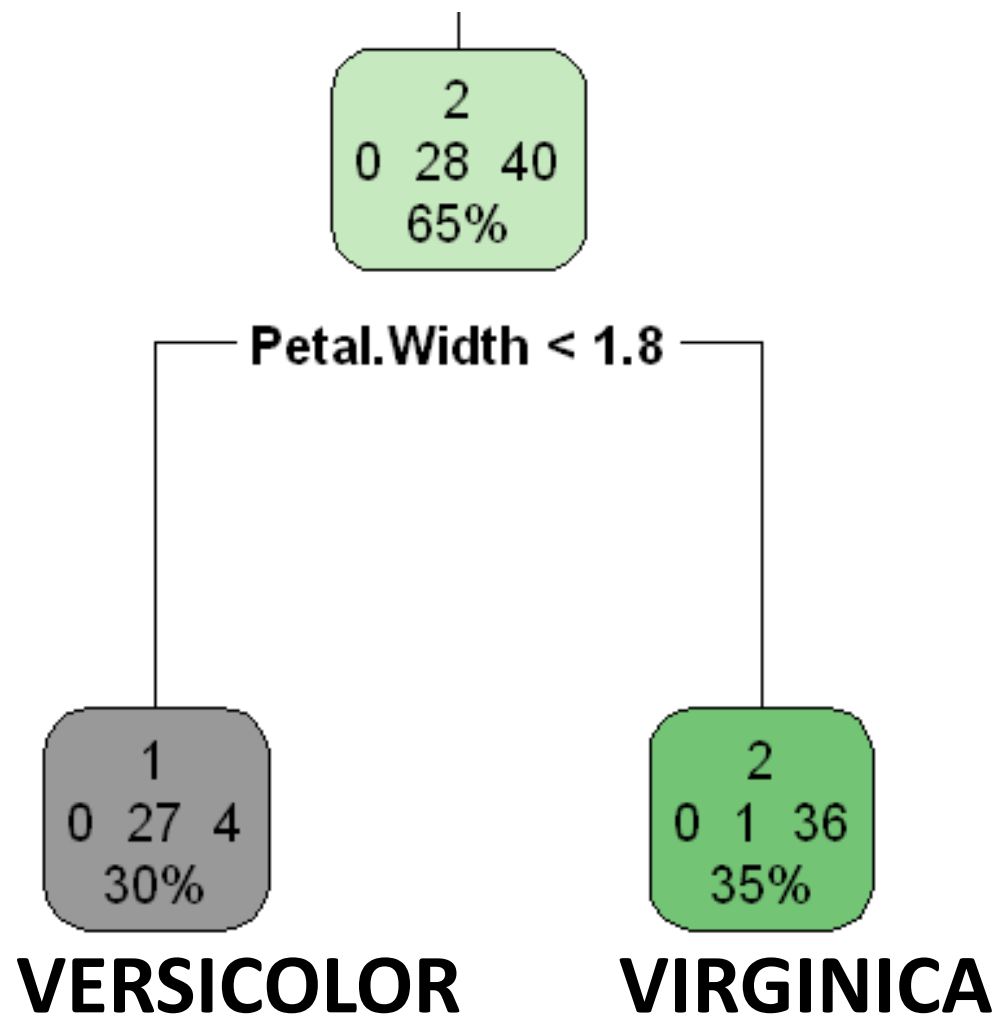
Funcionamiento



SETOSA



Funcionamiento



Ventajas y desventajas

- Funcionamiento fácil de entender.
- Muy visual
- Aunque no se use como mecanismo final de predicción puede usarse para ver relaciones entre las variables
- Ante pequeñas variaciones en la base de datos, puede sufrir grandes diferencias.
- Ante más categorías (dimensionalidad) se vuelve más difícil de interpretar.
- El más básico de su familia de algoritmos, el más potente actualmente se llama XGBOOST

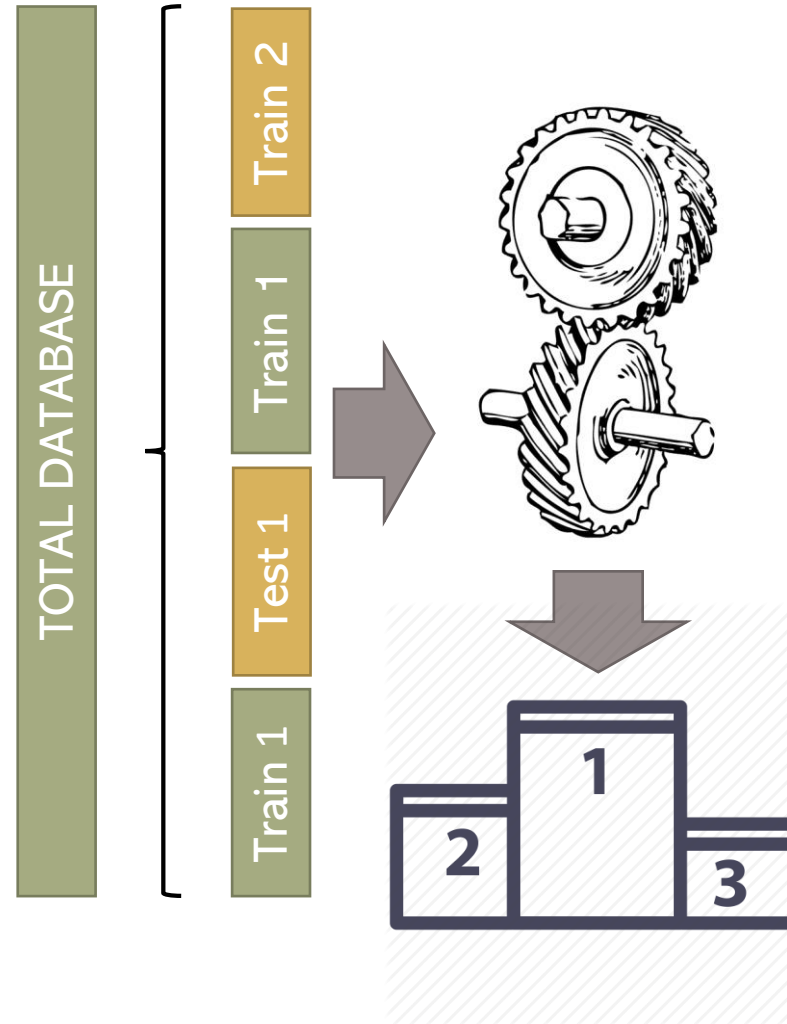


Comparación de Modelos

BLOQUE 6

Validación Cruzada

Una técnica para ver si el modelo de predicción que se usa es lo suficientemente robusto y sus resultados no son producto de *suerte* es la validación cruzada. También se le llega a conocer como *estimación de rotación*. Básicamente es intercambiar roles en una base de datos de *train* y *test* para ver si el método (no sólo el algoritmo de ML) es consistente y también sirve como herramienta de comparación al elegir un algoritmo.



No únicamente puede ser utilizado para valorar la efectividad de un modelo respecto a otro, sino una metodología entera (conseguir insumos, usar técnicas de reducción de dimensionalidad, limpieza de datos, transformación de datos, etc)

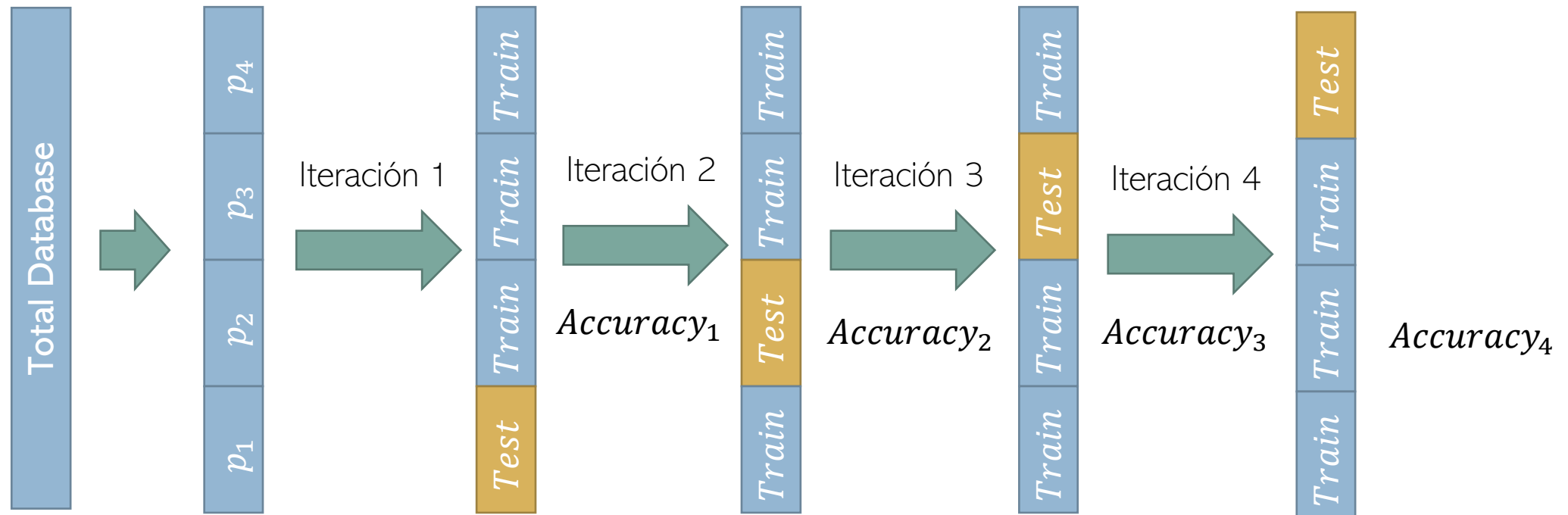
k-Fold Cross Validation

En vez de dividir en un 70% para *train* y 30% para *test*, se hace una partición casi uniforme de **k** dobleces. Por ejemplo, imaginemos que **k=4** y una base de **N=400** datos.

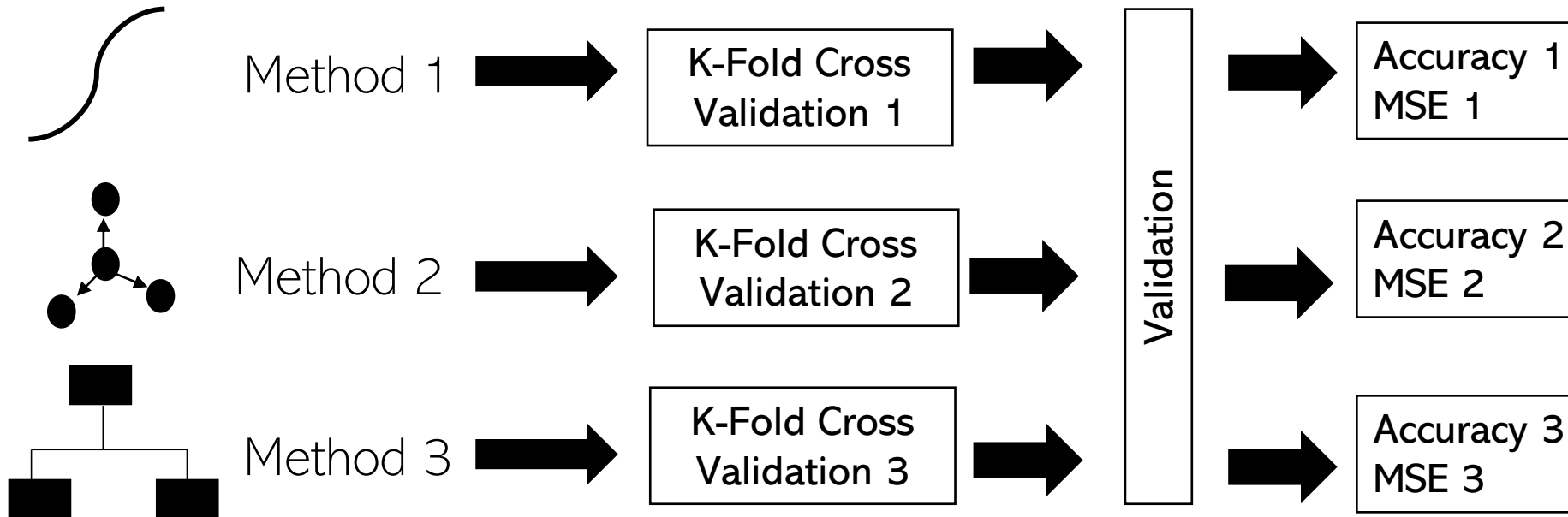
$$Size(p_i) = \frac{N}{k}$$

$$Size(p_i) = \frac{400}{4} = 100$$

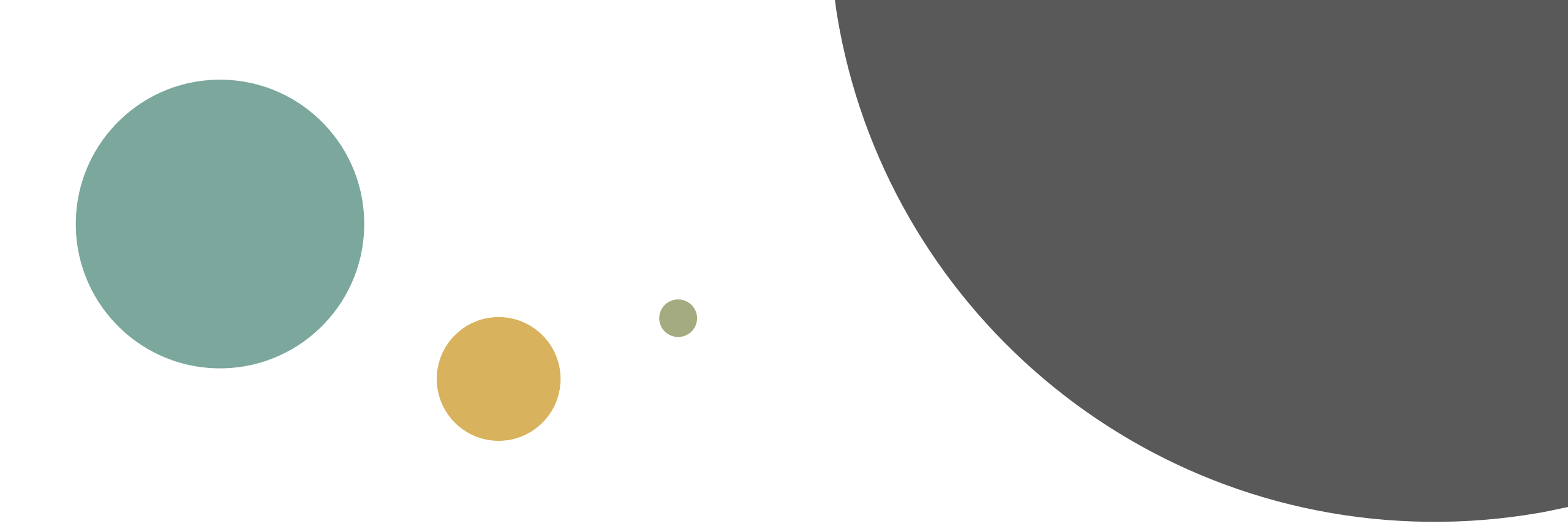
Calcular al final los promedios para Accuracy. En este espacio también se pueden sacar intervalos de confianza y algún otro estadístico deseado.



Comparación Final



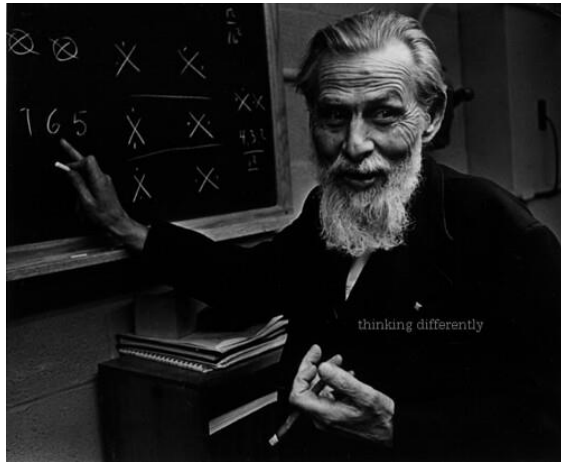
Finalmente se hará una comparación para decidir qué metodología u algoritmo es mejor para resolver un problema. Tomando en cuenta los resultados finales de cada uno



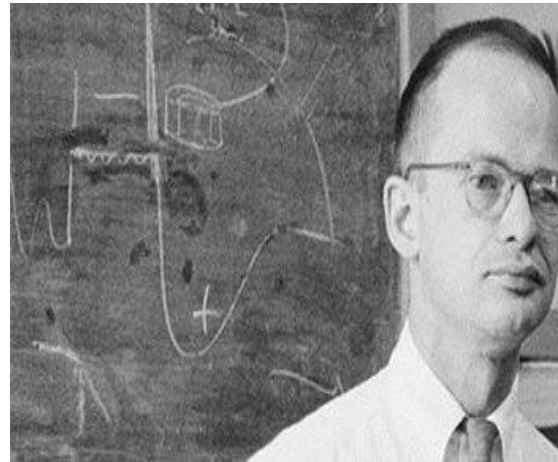
Introducción a las Redes Neuronales

BLOQUE 7

Algo de historia...



Warren McCulloch



Walter Pitts

1943

Creación de un modelo computacional para redes neuronales



Frank Rosenblatt

1958

Creación del Perceptrón

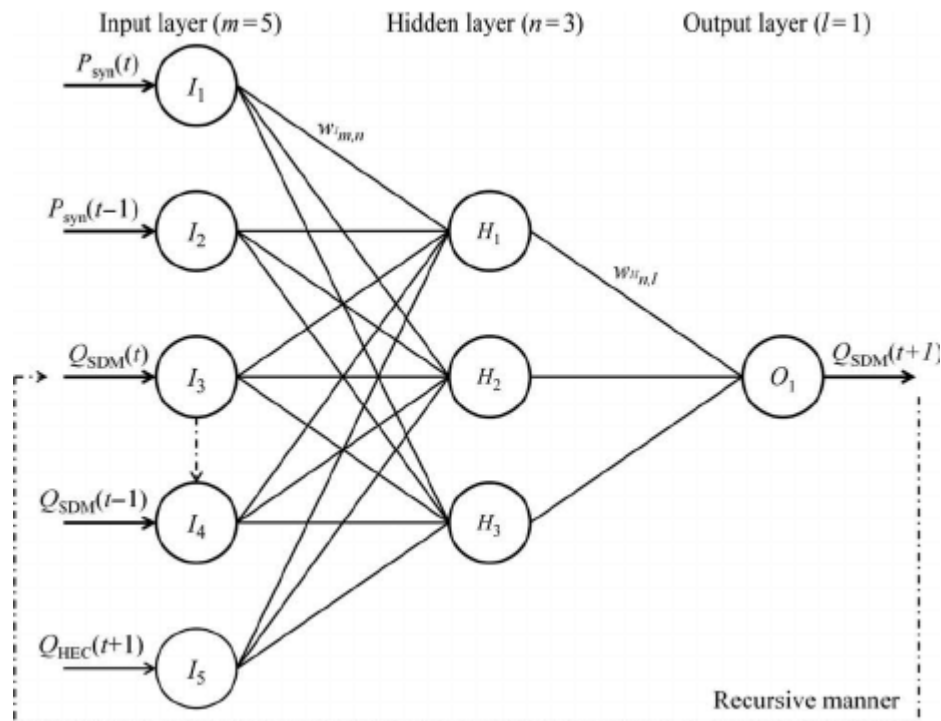


Marvin Minsky

1969

Era oscura

Algo de historia...



1982

Backpropagation

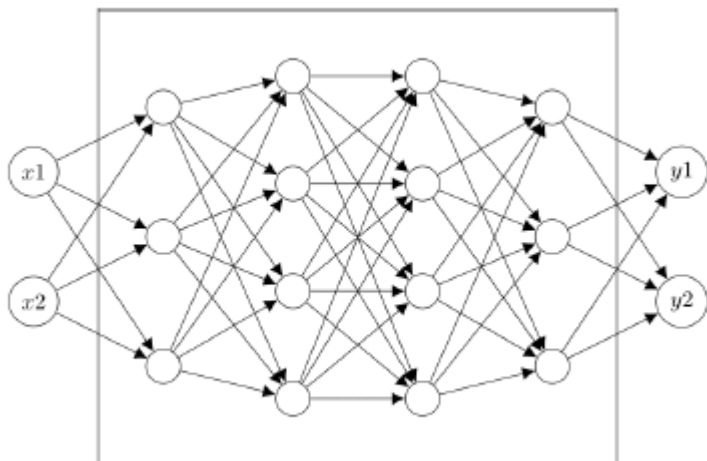


2010

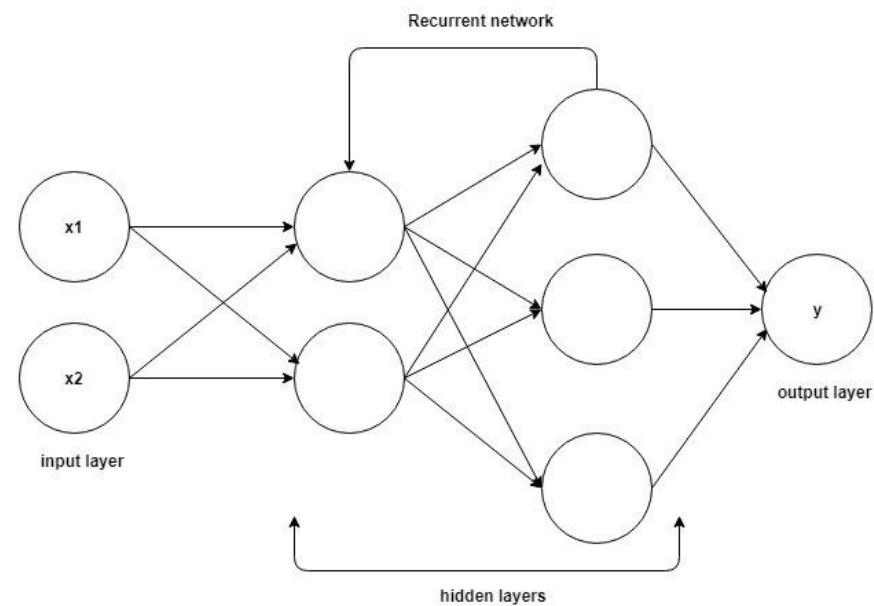
GPU

Tipos de Redes

Feedforward



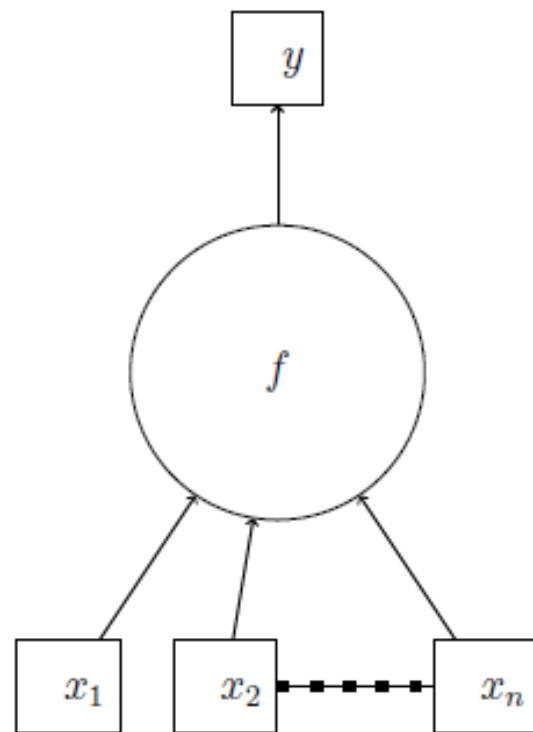
Recurrentes



La Neurona



(a) Neurona biológica



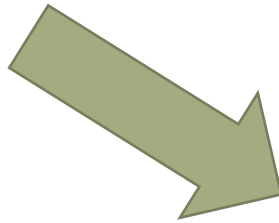
(b) Neurona artificial

Figura 2.6: Tipos de neuronas

Rompiendo la linearización

$$\begin{aligned}y(\mathbf{x}, \mathbf{w}) &= w_0 + w_1x_1 + \dots + w_nx_n \\ &= w_0 + \sum_{i=1}^n w_ix_i\end{aligned}$$

Función lineal

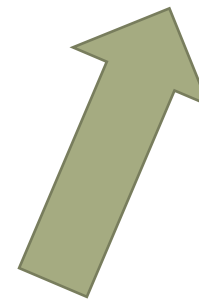


$$f(x) = \frac{1}{1 + e^{-x}}, \quad x \in \mathbb{R}.$$

Sigmoide

$$a = f\left(u + \sum_{i=1}^n w_ix_i\right)$$

Neurona



Capas Ocultas

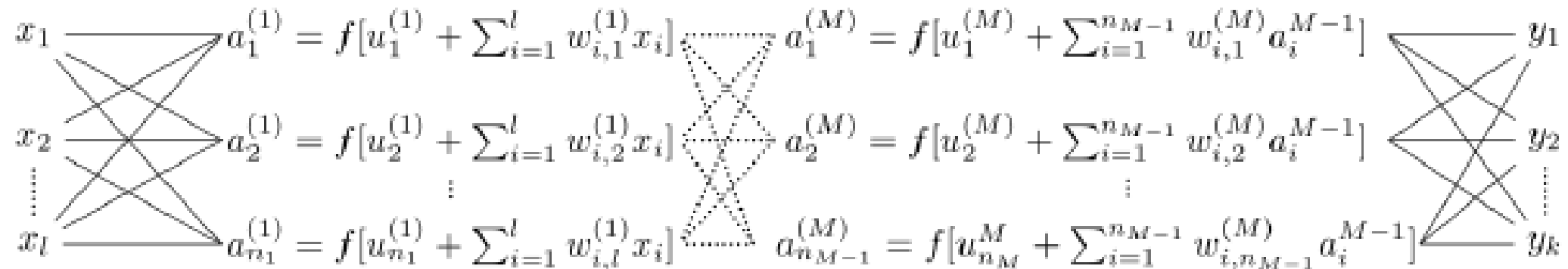


Figura 2.9: Red Neuronal Feedforward con múltiples capas ocultas.

Objetivo de la RNA: Disminuir el ECM (MSE)

$$ECM(\mathbf{w}) = \frac{1}{2} \sum_{t=1}^T ||\mathbf{y}(\mathbf{x}_t, \mathbf{w}) - \mathbf{z}_t||^2$$

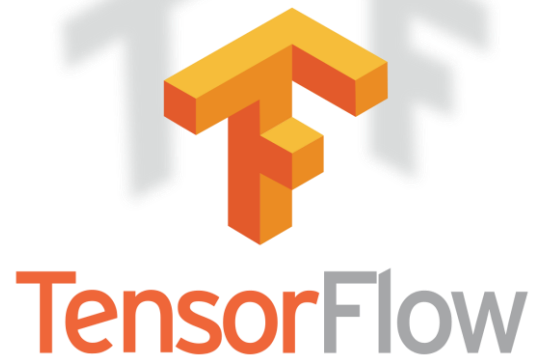
BackPropagation

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \alpha \nabla ECM(\mathbf{w}_t)$$

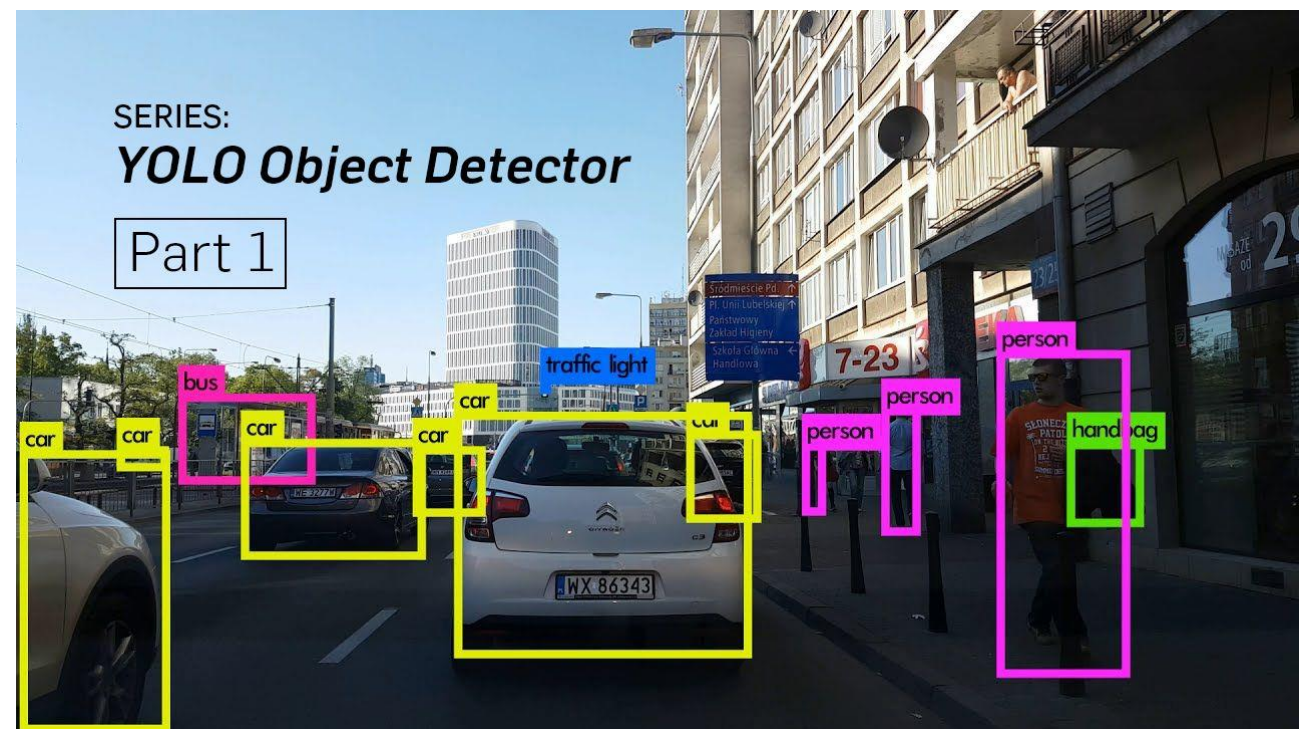
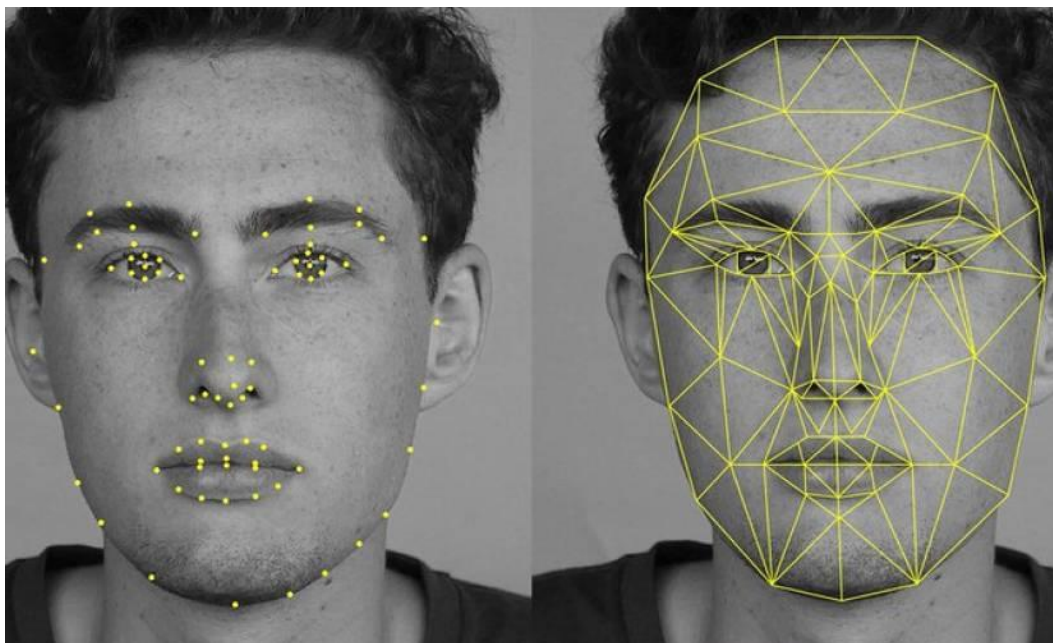
α : Factor de Operaciones

\mathbf{w}_t : Vector de pesos dentro de la RNA

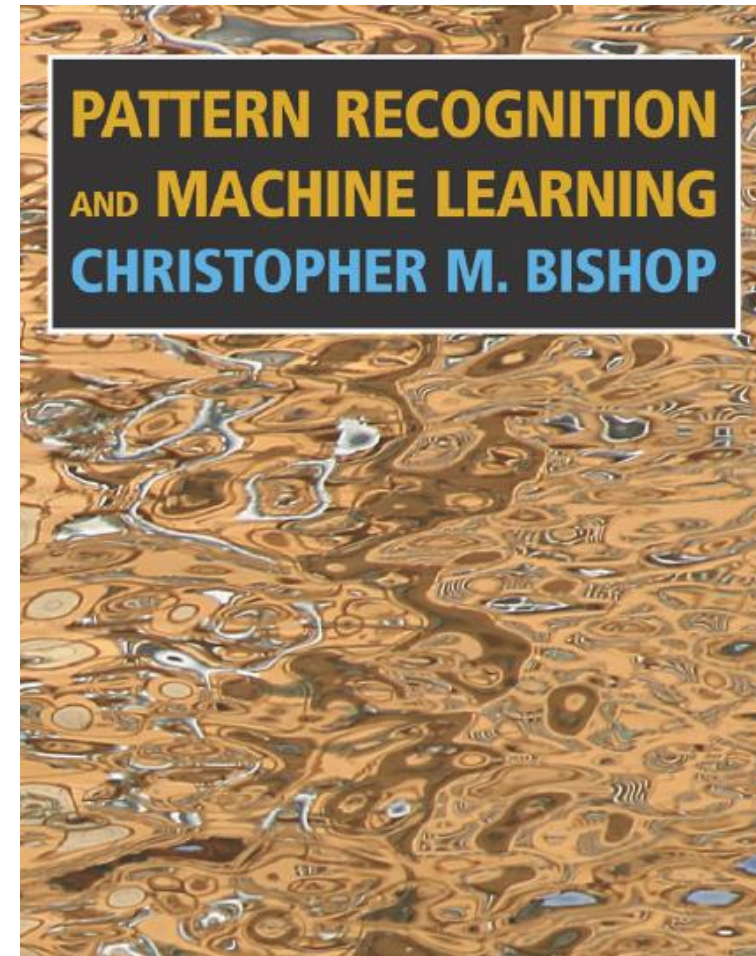
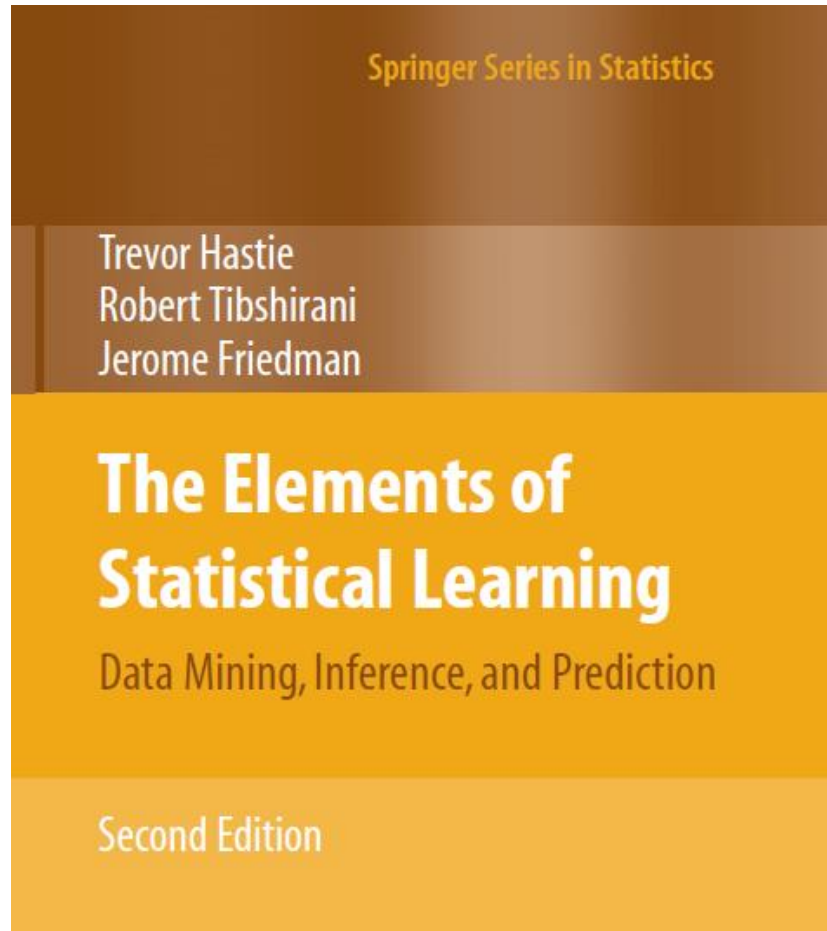
Software para RNAs



Usos dentro de las RNA



Recursos Extra



People with no idea about AI
saying it will take over the world:

My Neural Network:



GRACIAS!!!