

# Aprendizaje Automático, Level up!

Inteligencia Artificial e Ingeniería del Conocimiento

---

Constantino Antonio García Martínez

Universidad San Pablo Ceu

## Bibliografía

- Russell, Stuart J., and Peter Norvig. Artificial intelligence: a modern approach. Pearson, 2016.

# El conjunto de datos de Titanic

En esta presentación, mejoraremos nuestras habilidades estudiando un nuevo problema.

## Code Example: Titanic

Intentemos predecir si un pasajero sobrevivió al desastre del Titanic utilizando datos como la clase del billete, el sexo del pasajero, la edad, etc. **¿Es este un problema de regresión o clasificación?**

# El conjunto de datos de Titanic

En esta presentación, mejoraremos nuestras habilidades estudiando un nuevo problema.

## Code Example: Titanic

Intentemos predecir si un pasajero sobrevivió al desastre del Titanic utilizando datos como la clase del billete, el sexo del pasajero, la edad, etc. **¿Es este un problema de regresión o clasificación?**

Con algunos ajustes menores, podremos reutilizar la receta de aprendizaje automático de la presentación anterior. Solo necesitamos tener en cuenta:

1. Necesitamos un modelo de clasificación. Utilizaremos un modelo de regresión logística. (**¡Advertencia! A pesar del nombre, un regresor logístico es un clasificador!**)
2. Necesitamos una métrica para medir el rendimiento en un problema de clasificación. Una métrica razonable **es la precisión**:

# El conjunto de datos de Titanic

En esta presentación, mejoraremos nuestras habilidades estudiando un nuevo problema.

## Code Example: Titanic

Intentemos predecir si un pasajero sobrevivió al desastre del Titanic utilizando datos como la clase del billete, el sexo del pasajero, la edad, etc. **¿Es este un problema de regresión o clasificación?**

Con algunos ajustes menores, podremos reutilizar la receta de aprendizaje automático de la presentación anterior. Solo necesitamos tener en cuenta:

1. Necesitamos un modelo de clasificación. Utilizaremos un modelo de regresión logística. **(¡Advertencia! A pesar del nombre, un regresor logístico es un clasificador!)**
2. Necesitamos una métrica para medir el rendimiento en un problema de clasificación. Una métrica razonable **es la precisión**:  
Precisión: es la proporción de instancias predichas correctamente con respecto al total de instancias.

## El conjunto de datos de Titanic

Code Example: El conjunto de datos de Titanic

## Un problema de ML: pasos

Usemos el conjunto de datos de Titanic para intentar mejorar algunos puntos en nuestra Receta de ML:

### Receta de ML (III)

1. Ingeniería de características y Preprocesamiento.
2. Elegir un modelo. (¡debería ser un modelo de clasificación!)
3. Dividir los datos en conjuntos de entrenamiento y prueba.
4. Entrenar el modelo en el conjunto de entrenamiento para tratar de maximizar el rendimiento.
5. Medir el rendimiento real en el conjunto de prueba.

## Ingeniería y selección de características

---

# Ingeniería y selección de características

- **Ingeniería de características:** El proceso de crear nuevas características o transformar las existentes para mejorar el poder predictivo.
  - Ejemplo: En la clasificación de imágenes, necesitaríamos extraer características estadísticas como la intensidad media de los píxeles para usarlas como entrada de los clasificadores.
  - Ejemplo: Combinar las columnas `the number of siblings` y `parents` para crear la variable `FamilySize`.

Raw data:  
pixel grid



---

Better features: <b>clock</b> hands' coordinates	$\{x1: 0.7, y1: 0.7\}$	$\{x1: 0.0, y1: 1.0\}$
	$\{x2: 0.5, y2: 0.0\}$	$\{x2: -0.38, y2: 0.32\}$

---

Even better features: angles of clock hands	theta1: 45 theta2: 0	theta1: 90 theta2: 140
---	-------------------------	---------------------------

# Ingeniería y selección de características

- **Ingeniería de características:** El proceso de crear nuevas características o transformar las existentes para mejorar el poder predictivo.
  - Ejemplo: En la clasificación de imágenes, necesitaríamos extraer características estadísticas como la intensidad media de los píxeles para usarlas como entrada de los clasificadores.
  - Ejemplo: Combinar las columnas `the number of siblings` y `parents` para crear la variable `FamilySize`.
- **Selección de características:** Reducir el número de características para evitar el sobreajuste, reducir la complejidad y mejorar la generalización.
  - **Maldición de la dimensionalidad** (curse of dimensionality): a medida que aumenta el número de características, el rendimiento del modelo puede deteriorarse a menos que se maneje adecuadamente.
  - Métodos:
    - *Conocimiento experto*: Usar un conocimiento previo para seleccionar manualmente las características relevantes.
    - *Métodos de filtro*: Utilizar pruebas estadísticas (p. ej., correlación) para seleccionar las características relevantes.
    - *Métodos de envoltura*: Seleccionar características en función del rendimiento del modelo (p. ej., Eliminación recursiva de características).
    - *Métodos integrados*: La selección de características está integrada en el modelo (p. ej., regularización Lasso).

## Ingeniería y selección de características

**Code Example: El conjunto de datos de Titanic: Ingeniería y selección de características**

## Preprocesamiento de características

---

## Preprocesamiento de características

El preprocesamiento de características es esencial para preparar los datos para modelos de aprendizaje automático. Las técnicas clave incluyen:

- **Imputación:** Manejo de valores faltantes para evitar la pérdida de datos.
- **Codificación Categórica:** Convertir **variables categóricas** en formato numérico para la compatibilidad del modelo.
- **Estandarización:** Escalar **características numéricas** para tener una media de 0 y un ancho aproximado de 1, mejorando la convergencia del modelo.

### Exercise: Características Categóricas vs. Numéricas

¿Qué es una característica categórica? ¿Qué es una característica numérica?

Clasifica cada característica del conjunto de datos de Titanic en una de estas categorías.

## Preprocesamiento de características

---

### Imputación

## Imputación

La imputación es el proceso de reemplazar los valores faltantes en un conjunto de datos (NaNs en Python y NAs en R). Los métodos comunes incluyen:

- **Imputación por media:** Reemplazar los valores faltantes con la media de los datos disponibles.
- **Imputación por mediana:** Usar el valor de la mediana para el reemplazo, especialmente útil para distribuciones sesgadas.
- **Imputación por moda:** Rellenar los valores faltantes con la categoría más frecuente en datos categóricos.
- **K-Veinos Más Cercanos (K-Nearest Neighbors, KNN):** Utilizar el promedio de los K datos más cercanos para imputar los valores faltantes.

## Preprocesamiento de características

---

### Codificación Categórica

## Codificación Categórica

La codificación categórica transforma las variables categóricas en representaciones numéricas. Un método común es la **Codificación One-Hot**.

Original	Categoría A	Categoría B	Categoría C
Rojo	1	0	0
Verde	0	1	0
Azul	0	0	1

**Cuadro 1:** Ejemplo de Codificación One-Hot para una variable de color

## Estandarización

La estandarización es el proceso de escalar las características para tener una media de 0 y un ancho aproximado de 1 (esta es una definición no rigurosa). Los métodos comunes incluyen:

- **Estandarización Z-Score o Escalado Estándar:** Reescalar los datos en función de la media y la desviación estándar.
- **Escalado Min-Max:** Escalar las características a un rango de [0, 1].
- **Escalado Robusto:** Escalar las características según la mediana y el rango intercuartílico, haciéndolo resistente a los valores atípicos.

## Preprocesamiento de características

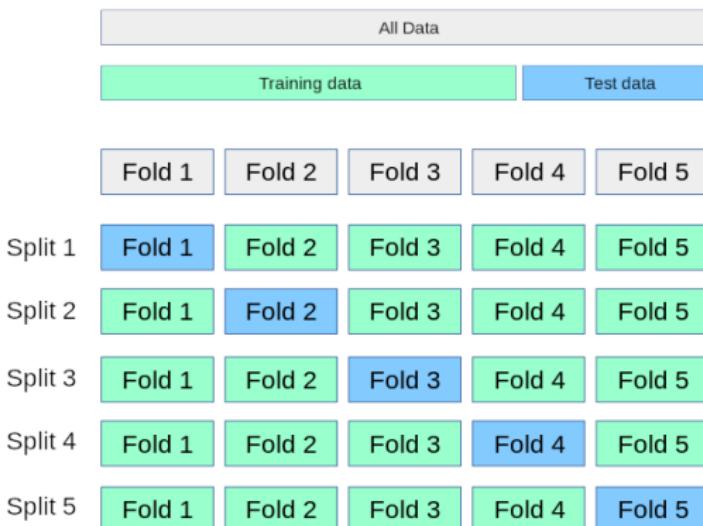
Code Example: Ejemplo de Titanic: preprocesamiento de características

## **Validación cruzada**

---

## Validación Cruzada (CV)

- **Objetivo:** Evaluar el rendimiento del modelo y asegurar que generaliza bien a datos no vistos.
- **Vs. División de train-test** CV proporciona una estimación más confiable del rendimiento del modelo al reducir el sesgo. Además, permite estimar la incertidumbre en el rendimiento del modelo.
- **Idea clave:** Dividir el conjunto de datos en varias partes (**pliegues o folds**) para entrenar y probar el modelo varias veces.



# Validación Cruzada

- **Tipos de Validación Cruzada:**

- **K-Fold:**

- Divide los datos en  $K$  partes iguales (pliegues).
    - Entrenar en  $K - 1$  pliegues, probar en el pliegue restante.
    - Repetir  $K$  veces, cada vez usando un pliegue diferente para pruebas.

- **Leave-One-Out:** Caso especial donde  $K = N$  (número de muestras).

- **K-Fold Estratificado:** Mantiene la proporción de clases en cada pliegue (usado principalmente para problemas de clasificación).

- **Ventajas:**

- Evaluación más robusta al usar múltiples divisiones de train-test.
  - Reduce el riesgo de sobreajuste o subajuste.

- **Desventajas:**

- Puede ser computacionalmente costoso para conjuntos de datos grandes.

## Validación Cruzada

Code Example: El conjunto de datos de Titanic: Validación Cruzada

Code Exercise: `cross_val_score`

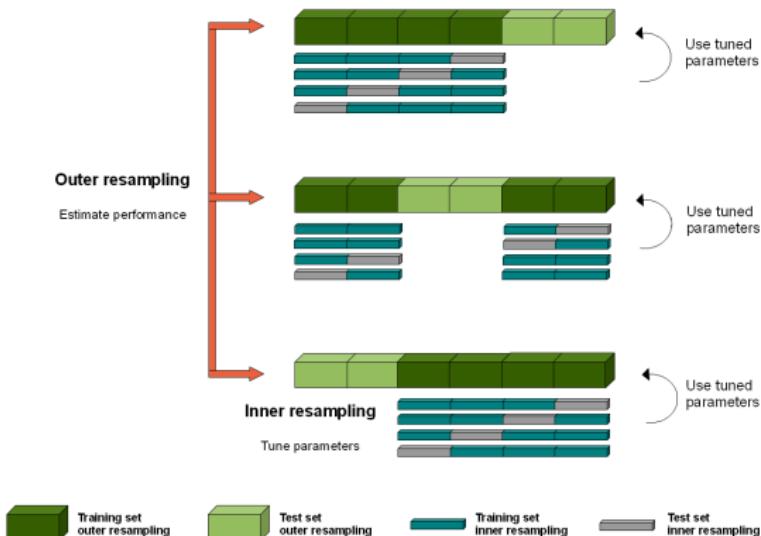
## **Validación cruzada**

---

### **Validación Cruzada Anidada**

# Validación Cruzada Anidada para Selección de Modelos

- **Objetivo:** Proporciona una estimación insesgada del rendimiento del modelo mientras se selecciona el mejor modelo (p. ej., ajuste de hiperparámetros).
- **Idea clave:** Combina dos capas de validación cruzada:
  - **Bucle exterior:** Estima el rendimiento de generalización del modelo.
  - **Bucle interior:** Selecciona el mejor modelo o hiperparámetros.



# Validación Cruzada Anidada para Selección de Modelos

Code Exercise: Model selection with nested CV

Code Exercise: Nested CV with Sklearn

## Proyectos de Investigación

---

- **Estudiar PCA para selección y visualización de características.**

PCA es una herramienta poderosa para reducir la dimensionalidad y puede ayudar a descubrir patrones ocultos en los datos, lo que lo hace útil tanto para seleccionar características esenciales como para visualizar conjuntos de datos de alta dimensionalidad.

- **Estudiar la Eliminación Recursiva de Características (RFE) para la selección de características.**

RFE es un método iterativo que clasifica las características por su poder predictivo, ayudando a refinar los modelos al eliminar las características menos informativas y mejorar la interpretabilidad.