Pre-Procesado

MSc. Felipe Meza-Obando

Aprendizaje Automático
Programa de Ciencia de Datos

January, 2025

1

Pre-procesado

2

Algunos métodos…

1. Metodología de Diseño.
2. Pre-procesado.
3. Preparación de los datos.
4. Algunas Tareas del Pre-procesado.
5. Análisis exploratorio de los datos (EDA).
6. Valores faltantes.
7. Outliers.
8. Datos no-balanceados.
9. Transformación de datos.
10. Reducción de dimensiones.

3

Datos – Un buen dato!

1. Precisión.
2. Completitud.
3. Consistencia.
4. Relevancia.
5. Actualización.
6. Accesibilidad.
7. Fiabilidad.

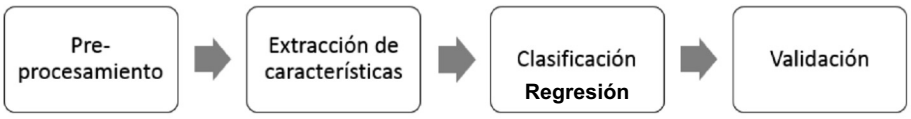
4

Datos – Un mal dato!

1. Imprecisión
2. Incompletitud
3. Inconsistencia
4. Irrelevancia
5. Obsolescencia
6. Dificultad de Acceso
7. No Fiabilidad

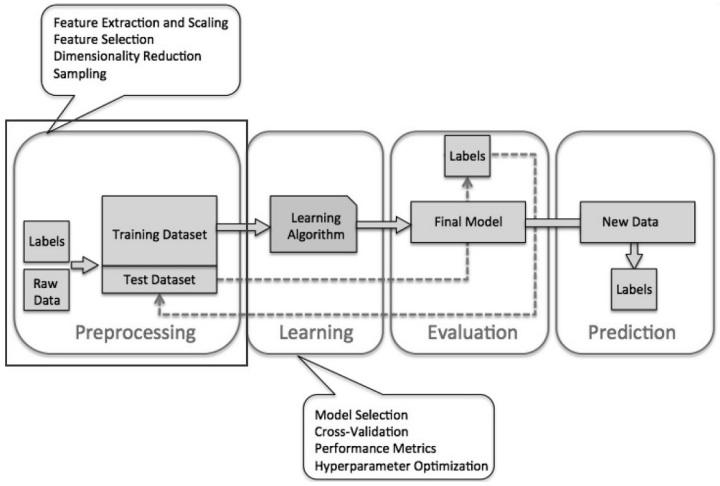
5

Introducción al Aprendizaje Automático



6

Metodología de Diseño



7

Pre-procesado

- Consiste en identificar partes o componentes del conjunto de datos que sean **incompletas**, imprecisas, incorrectas o irrelevantes, de manera tal que puedan ser **reemplazadas**, modificadas o removidas.
- Puede implicar también la **transformación** de los datos "puros" a otros formatos que faciliten su manejo por parte de los algoritmos de minería de datos.
- Incluye también la **reducción** de los datos a menores dimensiones para agilizar su procesamiento.
- En inglés varios **términos** se refieren a tales tareas: data preparation, cleaning, pre-processing, cleansing, wrangling.

8

Preparación de los datos

- En las metodologías de diseño generalmente en las **primeras etapas**, corresponde llevar a cabo las tareas de selección de datos, pre-procesado o transformación.
- En python, se recurre al uso de **librerías** como pandas que resultan ser muy buenas para las tareas asociadas a la preparación de los datos.
- Las labores de preparación no son un componente integral de los algoritmos de aprendizaje, sin embargo, puede tomar un **tiempo** considerable dependiendo del conjunto de datos a analizar (80%-90% del proceso), por lo que se debe prestar especial atención.

9

Algunas Tareas del Pre-procesado

- Estandarizar, Normalizar.
- Análisis exploratorio de los datos.
- Valores faltantes.
- Outliers.
- Datos no-balanceados.
- Transformación de datos.

10

Normalizar, Estandarizar

- **Normalizar** (1) es llevar los datos a una nueva escala en un rango entre 0 y 1. Recomendado en casos donde los datos tengan múltiples escalas y donde los algoritmos sean sensibles a la escala.
- **Estandarizar** (2) consiste en llevar la distribución de los datos a una media de 0 y una desviación estándar de 1. Recomendado en casos donde el algoritmo es sensible a una distribución normal.

$$z = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1)$$

$$z = \frac{x - \mu}{\sigma} \quad (2)$$

11

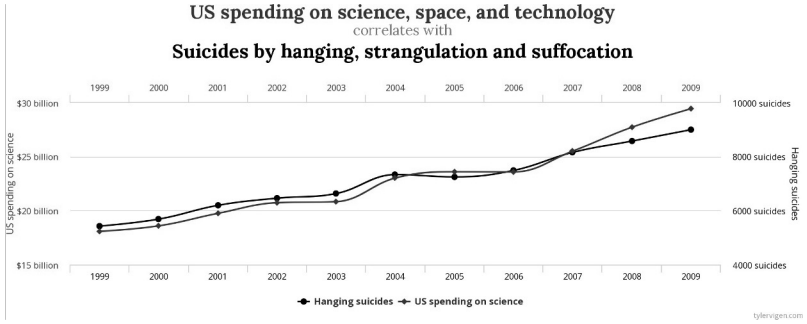
Análisis exploratorio de los datos (EDA)

- Es la práctica del uso de métodos cuantitativos y **visuales** para comprender mejor un conjunto de datos sin tener que asumir hechos.
- Arrojar el conjunto de datos a un algoritmo y esperar los mejores resultados, **NO** es la mejor estrategia.
- Usualmente se lleva cabo una o varias de las siguientes actividades:
  - **Visualización** de un resumen estadístico del conjunto de datos.
  - **Exploración** visual de cualquier relación que pueda tener cada atributo con la clase que nos interesa predecir.
  - Mediante diagramas de dispersión **observar** cualquier tipo de agrupamiento que se pueda presentar en los datos.

12

Análisis exploratorio de los datos (EDA)

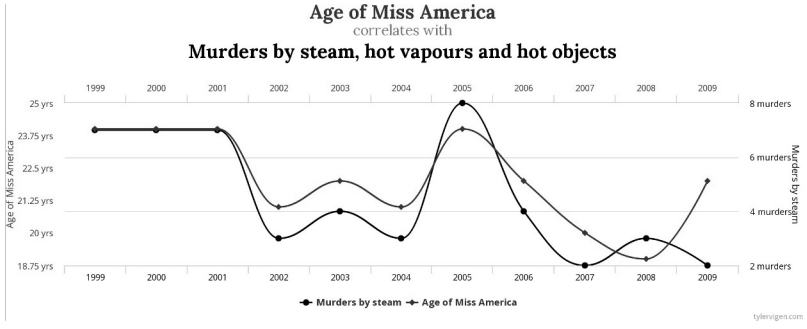
Es importante entender bien los datos no solo estadísticamente sino también su naturaleza a través de la exploración profunda.



13

Análisis exploratorio de los datos (EDA)

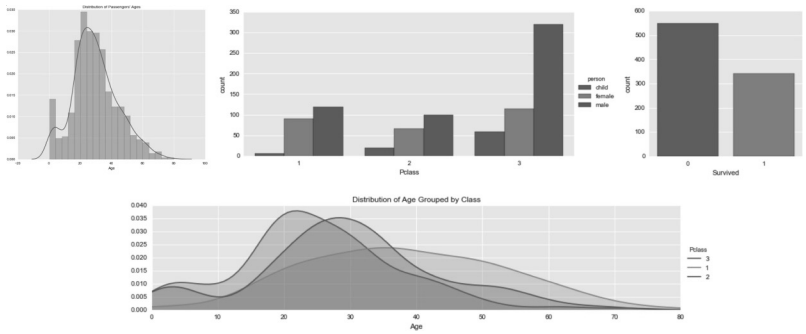
Es importante entender bien los datos no solo estadísticamente sino también su naturaleza a través de la exploración profunda.



14

Análisis exploratorio de los datos (EDA)

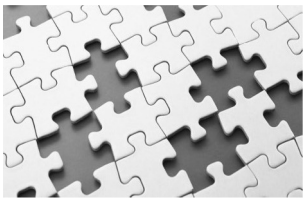
Ejemplo de EDA con conjunto de datos TITANIC



15

Manejo de Valores Faltantes en Datos

En el análisis de datos, los valores faltantes son un problema común que puede afectar la calidad de los modelos predictivos. No existe un **método universal** para tratarlos, ya que la mejor estrategia dependerá de la naturaleza del conjunto de datos, la cantidad de valores ausentes y el impacto que puedan tener en los análisis o modelos.



16

Manejo de Valores Faltantes en Datos

1. Eliminar Instancias (Filas) con Valores Faltantes

Se eliminan todas las filas que contienen valores ausentes en al menos una columna.

•Cuándo aplicarlo:

- Cuando la cantidad de valores faltantes es pequeña y no afecta significativamente la muestra.
- Cuando eliminar las instancias no genera un sesgo en el análisis o modelo.

```
df_cleaned = df.dropna()  # Elimina filas con valores NaN
```

17

Manejo de Valores Faltantes en Datos

2. Eliminar Atributos (Columnas) con Valores Faltantes

Se eliminan las columnas que contienen valores faltantes.

•Cuándo aplicarlo:

- Cuando una columna tiene demasiados valores faltantes (> 50%) y su eliminación no afecta la interpretación del modelo.
- Cuando la variable no es relevante o no aporta información útil.

```
df_reduced = df.dropna(axis=1)  # Elimina columnas con valores NaN
```

18

Manejo de Valores Faltantes en Datos

3. Imputación con la Media del Atributo Faltante

Se reemplazan los valores faltantes con el promedio de la columna.

•Cuándo aplicarlo:

- Cuando la variable tiene una distribución normal y los valores extremos no afectan significativamente la media.
- Cuando la cantidad de datos faltantes es moderada.

```
df['columna'] = df['columna'].fillna(df['columna'].mean())
```

19

Manejo de Valores Faltantes en Datos

4. Imputación con la Mediana del Atributo Faltante

Se reemplazan los valores faltantes con la mediana de la columna.

•Cuándo aplicarlo:

- Cuando la variable tiene una distribución sesgada y la media podría verse afectada por valores atípicos.
- Cuando los valores extremos tienen una alta variabilidad.

```
df['columna'] = df['columna'].fillna(df['columna'].median())
```

20

Manejo de Valores Faltantes en Datos

5. Imputación con la Moda del Atributo Faltante

Se reemplazan los valores faltantes con el valor más frecuente en la columna.

•**Cuándo aplicarlo:**

- Cuando se trabaja con variables categóricas.
- Cuando la mayoría de los valores de la columna son iguales y reemplazar los valores faltantes con la moda no altera significativamente la distribución.

```
df['columna'] = df['columna'].fillna(df['columna'].mode()[0])
```

21

Manejo de Valores Faltantes en Datos

6. Usar Regresión Lineal para Estimar los Valores Faltantes

Se entrena un modelo de **regresión** para predecir los valores faltantes utilizando otras variables del dataset como entrada.

•**Cuándo aplicarlo:**

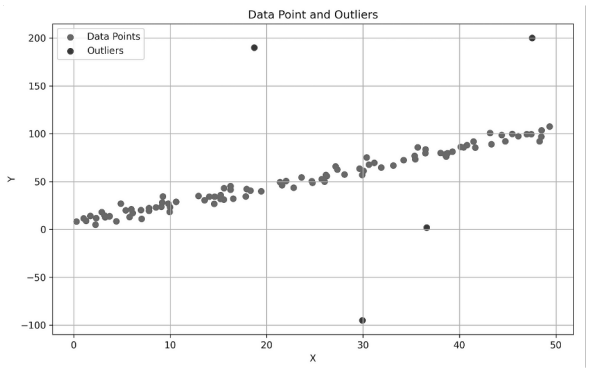
- Cuando hay una relación entre la variable con valores faltantes y otras variables en el dataset.

```
reg = LinearRegression().fit(df_train[['Color', 'Tamaño']], df_train['Precio'])
```

22

Outliers

Los *outliers* (valores atípicos) son observaciones que se desvían significativamente del resto de los datos. Pueden deberse a errores de medición, ruido en los datos o eventos genuinamente inusuales.



23

Outliers

¿Cuándo eliminar los *outliers*?

- Cuando afectan negativamente el desempeño de un modelo.
- Cuando se identifican como errores de medición o entrada de datos.
- Cuando distorsionan las métricas estadísticas clave (*media*, *desviación estándar*, etc.).

24

Outliers

¿Cuándo conservar los *outliers*?

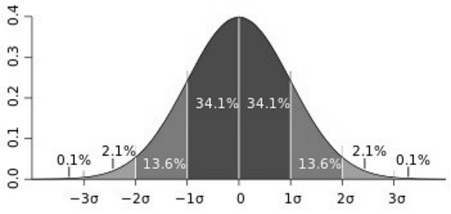
•Cuando representan fenómenos reales (ej. ventas extraordinarias, condiciones climáticas extremas).
•Cuando pueden ser relevantes para detectar patrones anómalos (*fraude, fallos en máquinas, enfermedades raras*).

25

Métodos para Identificar y Remover Outliers

1. Usando Desviación Estándar

Si los datos siguen una distribución normal, los valores fuera de **3 desviaciones estándar** se pueden considerar *outliers*.



26

Métodos para Identificar y Remover Outliers

2. Usando Percentiles (Cuartiles)

Este método se basa en los valores del **primer (Q1) y tercer cuartil (Q3)**, y calcula el **Rango Intercuartil (IQR)**. Se considera *outlier* cualquier dato que esté **fuera de 1.5 veces el IQR**.

27

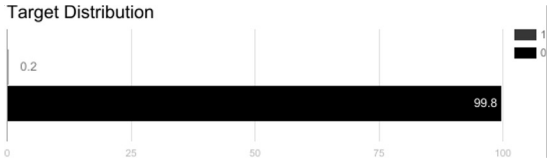
Outliers

No existe un **método universal** para tratar *outliers*. Se debe considerar la **naturaleza de los datos** antes de eliminarlos. Métodos como **desviación estándar** y **percentiles** son útiles para identificar y remover valores extremos.

28

Datos No Balanceados: Desafíos y Estrategias

En muchos problemas de clasificación, los datos pueden estar desbalanceados, lo que significa que una o más clases tienen muchas más instancias que las otras. Esto puede causar que los modelos de Machine Learning tengan **sesgo hacia la clase mayoritaria**, reduciendo la capacidad de detectar correctamente la clase minoritaria.



En estos casos, **no es recomendable evaluar el modelo solo con precisión (accuracy)**, ya que puede ser engañosa.

29

Datos No Balanceados: Desafíos y Estrategias

En lugar de usar solo *accuracy*, es recomendable emplear métricas que evalúen el desempeño en cada clase:

•**Precision / Especificidad**: Indica cuántos de los elementos clasificados como positivos son realmente positivos.

•**Recall / Sensibilidad**: Mide qué proporción de los positivos reales fueron identificados correctamente.

•**F1-Score**: Es la media armónica de *Precision* y *Recall*, útil cuando se busca un balance entre ambas métricas.

30

Datos No Balanceados: Técnicas de Muestreo

Para equilibrar las clases, podemos **modificar la cantidad de instancias de cada clase**:

Submuestreo (Under-sampling): Reduce la cantidad de instancias de la clase mayoritaria eliminando datos. Se usa cuando hay suficientes datos de la clase minoritaria.

```
from imblearn.under_sampling import RandomUnderSampler
X_resampled, y_resampled = RandomUnderSampler().fit_resample(X, y)
```

Sobremuestreo (Over-sampling): Aumenta la cantidad de instancias de la clase minoritaria replicando datos o generando sintéticos.

```
from imblearn.over_sampling import SMOTE
X_resampled, y_resampled = SMOTE().fit_resample(X, y)
```

31

Datos No Balanceados: Técnicas de Muestreo

Aplicar Clustering en el Grupo Mayoritario

•En lugar de eliminar datos directamente, podemos usar **clustering** (ej. K-Means) para agrupar los datos de la clase mayoritaria y seleccionar una muestra representativa.

Si el desbalanceo es severo y no se puede solucionar con muestreo, ***explorar modelos diseñados para datos desbalanceados*** *como* ***XGBoost, Balanced Random Forest ****

32

Transformación de datos

- Ocurre cuando **transformamos** un valor zi en yi mediante una función $f()$ de forma tal que $yi = f(zi)$.
- Se hace con el fin de alinear los datos con alguna suposición estadística, mejorar la interpretación de los datos o bien obtener gráficos de mejor apariencia.
- Técnica muy común: **One Hot Encode**, permite convertir datos categóricos en numéricos (vectores binarios).

Sample	Category	Numerical
1	Human	1
2	Human	1
3	Penguin	2
4	Octopus	3
5	Alien	4
6	Octopus	3
7	Alien	4

Datos Categ´oricos

Sample	Human	Penguin	Octopus	Alien
1	1	0	0	0
2	1	0	0	0
3	0	1	0	0
4	0	0	1	0
5	0	0	0	1
6	0	0	1	0
7	0	0	0	1

Vectores Binarios

$$[0\ \ 0\ \ 0\ \ 1\ \ 0] \times \begin{bmatrix} 17 & 24 & 1 \\ 23 & 5 & 7 \\ 4 & 6 & 13 \\ 10 & 12 & 19 \\ 11 & 18 & 25 \end{bmatrix} = [10\ \ 12\ \ 19]$$

One Hot Encode

33

Clasificación de Datasets por Nivel de Dificultad

La dificultad de un dataset en Machine Learning depende de varios factores como el número de características (*features*), la cantidad de datos, la presencia de ruido y la separabilidad de las clases.

Dataset de Baja Dificultad

•Pocas *features* (≤ 10) y la mayoría son numéricas.
•Bajo número de muestras.
•Clases bien separadas y distribuidas equitativamente.
•Baja cantidad de valores faltantes o atípicos (*outliers*).

34

Clasificación de Datasets por Nivel de Dificultad

Dataset de Mediana Dificultad

•Número moderado de *features*.
•Mayor cantidad de muestras.
•Clases menos separables y/o cierto desbalance de datos.
•Presencia de datos categóricos y algunas variables ruidosas.

Dataset de Alta Dificultad

•Alto número de *features*, a menudo no estructurados (texto, imágenes).
•Gran volumen de datos.
•Clases fuertemente solapadas y datos altamente desbalanceados.
•Alta presencia de ruido, valores atípicos y datos faltantes.

35

Algunas Fuentes de DATASETS

- Kaggle Datasets
  - www.kaggle.com
- UCI Machine Learning Repository
  - https://archive.ics.uci.edu/ml/index.php
- Google's Datasets
  - https://toolbox.google.com/datasetsearch
- Microsoft Datasets
  - https://msropendata.com/
- Awesome Public Datasets
  - https://github.com/awesomedata/awesome-public-datasets

36

Questions?

Felipe Meza / fmeza@itcr.ac.cr

37