“DATOS y PRE-PROCESADO”

Reconocimiento de Patrones

MAESTRÍA EN ELECTRÓNICA

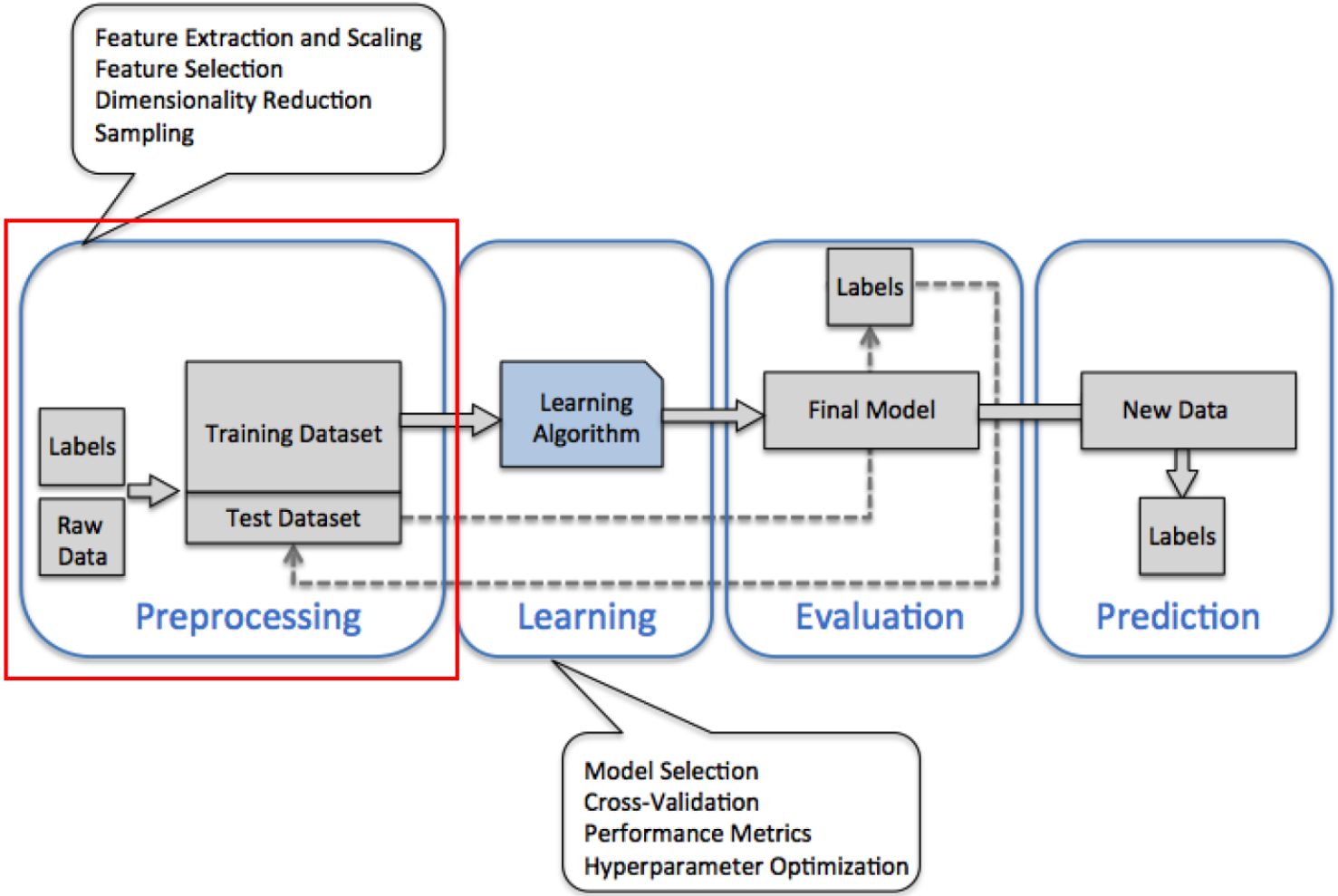
Profesor: MSc. Felipe Meza

June 3, 2021

Agenda

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.

Metodología de Diseño



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.

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 prepraración 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.

Algunas Tareas del Pre-procesado

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

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.

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

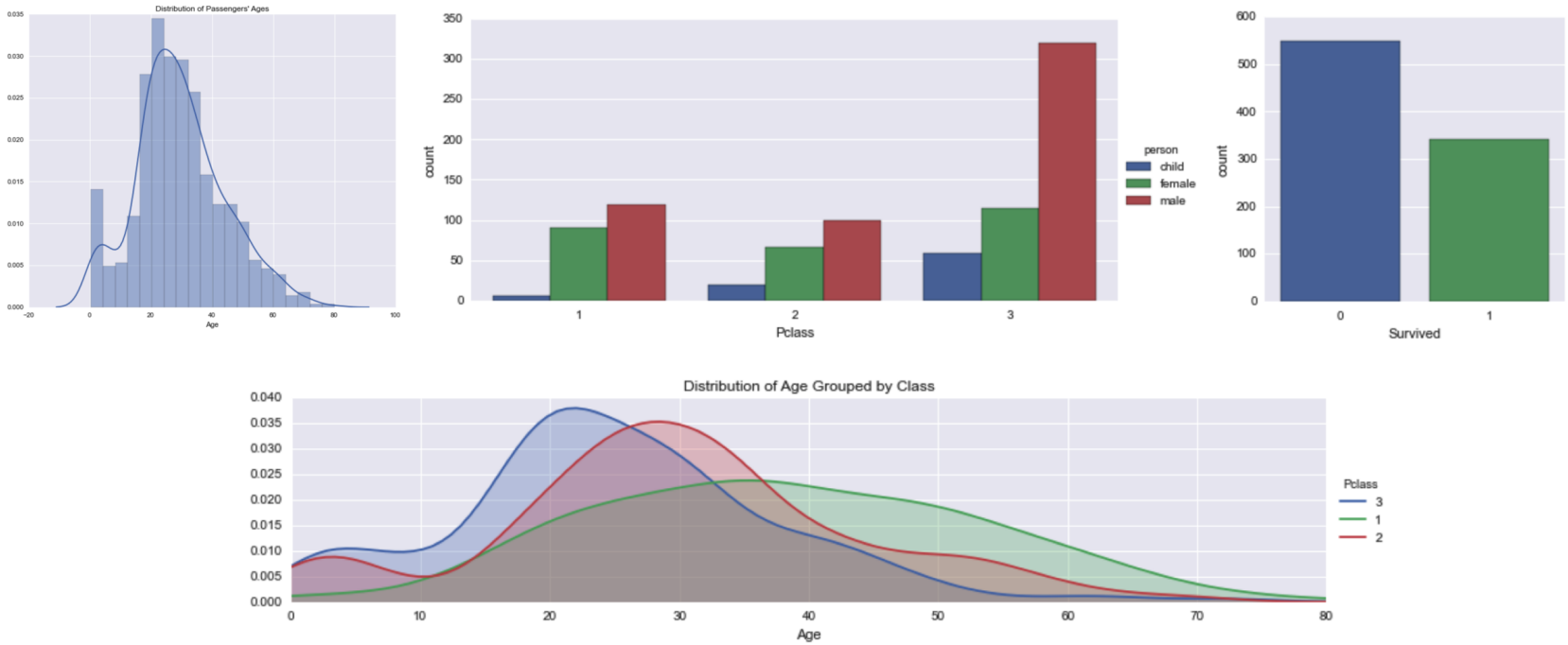
$$z = \frac{x - \mu}{\sigma} \tag{2}$$

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.

Análisis exploratorio de los datos (EDA)

Ejemplo de EDA con conjunto de datos TITANIC



Valores faltantes

- No existe un **método universal** para instancias con valores faltantes.
- Algunas técnicas **comúnes** son:
  - Eliminar instancias.
  - Eliminar atributos.
  - Calcular "media" del atributo faltante.
  - Calcular "mediana" del atributo faltante.
  - Calcular "moda" del atributo faltante.
  - Usar regresión para estimar el valor del atributo faltante.

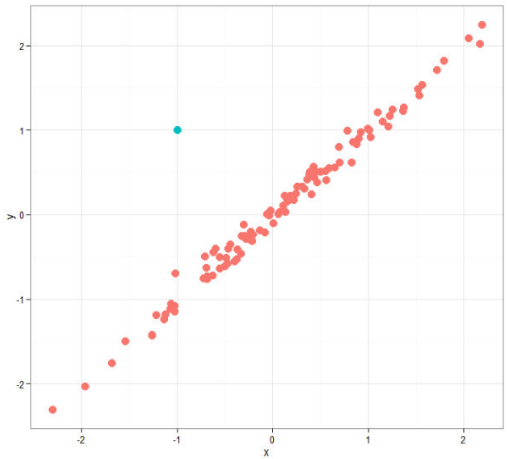
Valores faltantes

pandas

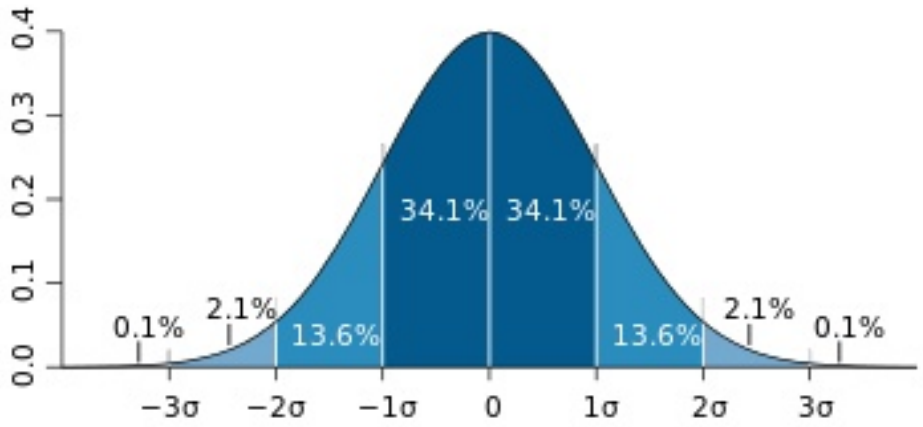
```
# Drop the col where all elements are missing values:
df.dropna(axis=1, how='all')

# Drop the col where any of the elements're missing values
df.dropna(axis=1, how='any')

# Keep only the rows which contain 2 missing values max
df.dropna(thresh=2)

# Fill all missing values with the mean of the column
df.fillna(df.mean())

# Fill any missing value in col 'A' with the col median
df['A'].fillna(df['A'].median())
```

Outliers

- Así como hay situaciones donde es importante mantenerlos en otros casos es necesario **eliminarlos**.
- Se recomienda hacer un análisis previo basado en la **naturaleza** de los datos.
- Algunas técnicas comúnes son:
  - Removerlos usando desviación estándar (`PYTHON`).
  - Removerlos usando percentiles (`pandas`).



Outliers - Desviación Estándar



Regla en distribución normal: Remover datos en $(mean + 2\sigma)$ y $(mean - 2\sigma)$

Regla 68-95-99 ó 3σ

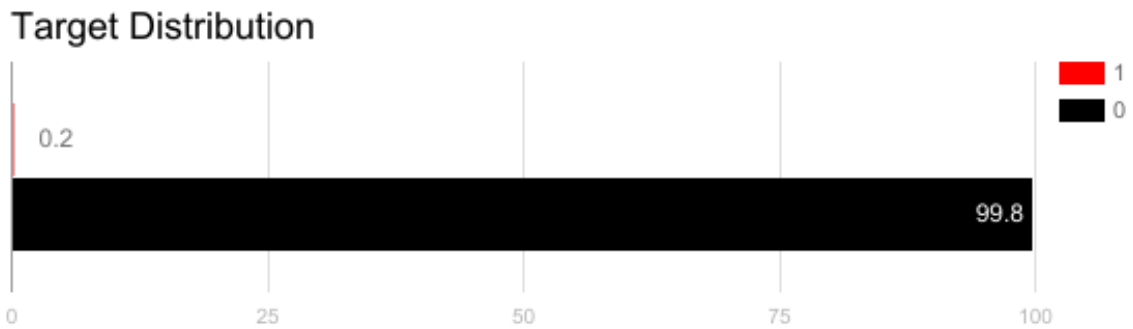
Outliers - Desviación Estándar

PYTHON

```python
import numpy

arr = [10, 386, 479, 491, 501 ... 411, 399, 363, 19, 543]

elements = numpy.array(arr)

mean = numpy.mean(elements, axis=0)
sd = numpy.std(elements, axis=0)

final_list = [x for x in arr if (x > mean - 2 * sd)]
final_list = [x for x in final_list if (x < mean + 2 * sd)]
print(final_list)
```

Datos no-balanceados

- Ocurre cuando una clase de datos en el conjunto, posee una mayoría importante de la cantidad de datos e.g un conjunto de datos de 2 clases, donde: $CLASE1 = 98\%$ y $CLASE2 = 2\%$.



Datos no-balanceados

- Algunas técnicas comúnes son:
  - Usar otras métricas diferentes al porcentaje de exactitud, por ejemplo:
    - Precision/Specificity: cuantas instancias seleccionadas son relevantes.
    - Recall/Sensitivity: cuantas instancias relevantes son seleccionadas.
    - F1 score: media harmónica de "precision" y "recall".
  - Muestreo de datos:
    - sub-muestreo: Eliminar instancias abundantes (sólo si hay suficientes datos).
    - sobre-muestreo: Generar instancias faltantes (mediante métodos de repetición o generación, solo en caso de que sea posible)
  - Descomponer el conjunto de datos en subconjuntos.
  - Hacer clustering de grupo abundante.

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 Categóricos

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

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

One Hot Encode

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`

Questions?

Felipe Meza - fmeza@itcr.ac.cr