

Aprendizaje No Supervisado (UL).

Investigación Corta #3

Carlos Brenes-Jiménez.

Julio 2020

Curso: Reconocimiento de Patrones.

Agenda

1. Definiciones
2. Aprendizaje No Supervisado vs. Supervisado
3. Bibliotecas Utilizadas
4. Fortalezas y Debilidades
5. Retos
6. Algoritmos y Aplicaciones en cuaderno de Jupyter Lab
7. Referencias

Modelo de Aprendizaje:

- Aquel que aprende de los datos de entrenamiento (Experiencia) para mejorar su tasa de error (Desempeño) en una tarea.
- El criterio de éxito final es qué tan bien su experiencia generaliza a algo nuevo (error de generalización).
- La entrada vectorial \bar{X} es transformada en una salida vectorial \bar{Y} .

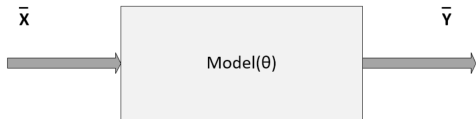


Figure 1: Modelo genérico [1]

Aprendizaje No Supervisado:

- Las etiquetas no están disponibles.
- La tarea del agente de *Int.Artificial* no está bien definida.
- Su desempeño no puede ser tan claramente medido.
- Los problemas no están claramente definidos y tienen complejidad mayor.

Aprendizaje No Supervisado vs. Supervisado.

Table 1: Comparación de Aprendizajes. [3]

Parámetros	No Supervisado	Supervisado
Datos	No Etiquetados	Etiquetados
Implementacion Computacional	Compleja	Simple
Exactitud/Confiabilidad	Menor	Mayor

Bibliotecas utilizadas [4]:

- **NumPy:** Procesamiento de arreglos multi-D, matrices y Alg Lineal.
- **Pandas:** Datos relacionales/etiquetados (Serial-1D y DF-2D).
- **SciPy:** Alg.Lineal, Proc.Imágenes/Señales, Sol/Ecuaciones
- **Scikit-learn:** Algos(NSup/Sup), Clasif/Regres, P-Procesado.
- **Theano:** Compilador de Optimización, Soporta(CPUs/GPUs).
- **TensorFlow:** Const/Modelos(ML-DNA), H/W:GPU/CPU/TPUs.
- **Keras:** Soporte a DNN/CNN/RNN, Imágenes/Texto.
- **PyTorch:** Visión por Computador, ML, NLP.
- **Torch:** Cálculo de tensores, H/W(GPUs-PreProc).
- **Matplotlib:** visualización de datos 2D/Fig (varios formatos)

Fortalezas:

- Efectivo en problemas de patrones(ocultos) / Conj.Datos desconocidos y variables.
- Aprende la estructura subyacente de los datos/Entrenamiento.
- Representa los datos con un conjunto de parámetros menor.
- Etiqueta automáticamente datos no etiquetados.
- Las propiedades espectrales de las clases cambian con el tiempo, por lo tanto puede detectar nuevos patrones en datos futuros.

Debilidades:

- No hay información precisa de la clasificación (datos no etiquetados)
- El usuario necesita invertir tiempo en interpretar y etiquetar las clases provenientes del proceso de clasificación.

Retos:

- **Sobre-ajuste (Overfitting)**
- **Valores Atípicos (Outliers)**
- **Desplazamiento de datos (Data drift)**



1 Patel, Ankur A (2019)

Hands-On Unsupervised Learning Using Python: How to Build Applied Machine Learning Solutions from Unlabeled Data

O'Reilly Media.



2 Guru99. (2020)

Unsupervised Machine Learning: What is, Algorithms, Examples

[https://www.guru99.com/unsupervised-machine-learning.html#10.](https://www.guru99.com/unsupervised-machine-learning.html#10)



3 Bonaccorso, Giuseppe and Fandango, Armando and Shanmugamani, Rajalingappa (2018)

Python: Advanced Guide to Artificial Intelligence: Expert machine learning systems and intelligent agents using Python
Packt Publishing Ltd.



4 upgrad.com (2020)

Top 9 Python Libraries for Machine Learning in 2020
<https://www.upgrad.com/blog/top-python-libraries-for-machine-learning/>.