

julia

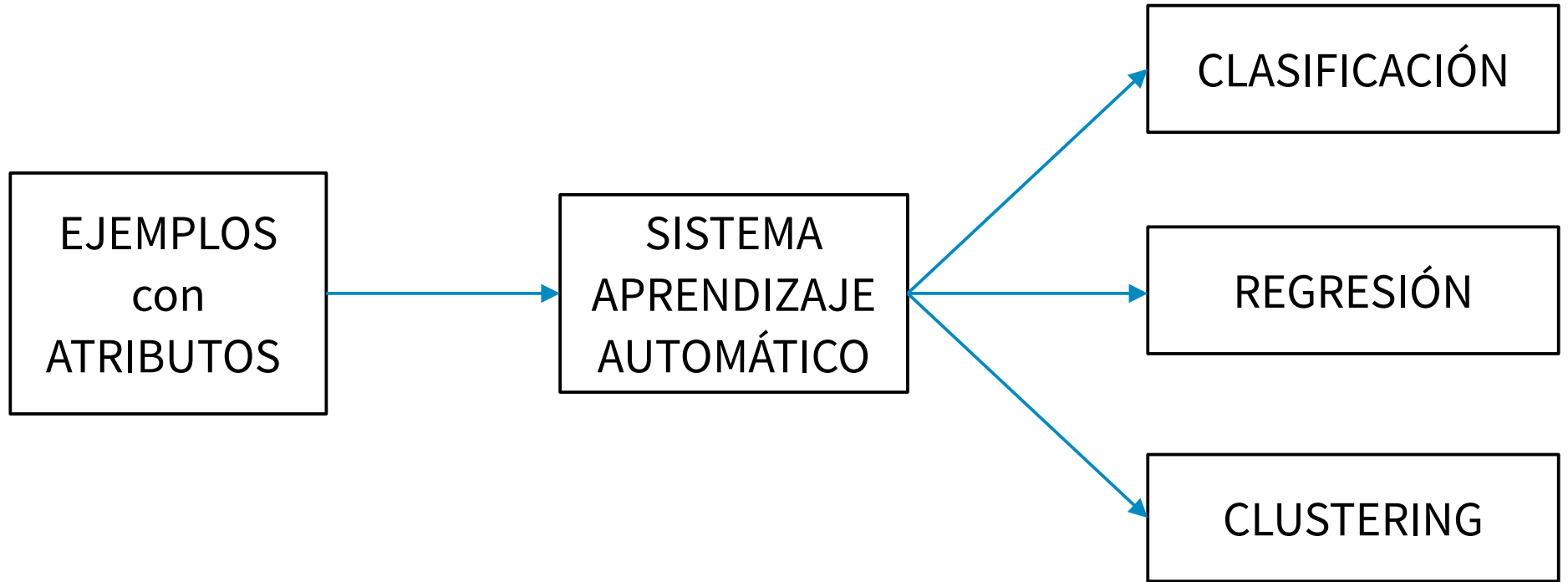
UVa

Escuela
Ingeniería
Informática
Valladolid
UVa

HOUR
OF
CODE

1 DIC 2021

Planteamiento General



Tratamientos de la ENTRADA

- **Organización** normalmente en matrices bidimensionales:
 - **último índice** es el número de **ejemplo/muestra**
(en python es el primero)
 - Específicamente, matrices 4D en imágenes ó 5D en vídeo
- **Selección** de muestras de aprendizaje/test estratificada, si es para clasificación:
 - **Resorte** (Hold-Out)
 - **Validación cruzada** (K-Fold)
- **Normalización** de atributos:
 - **Escalado** [0,1]
 - **Estandarización** (media=cero, desviación estándar=1)

DATAFRAMES

- Equivalente a **Pandas**, pero más orientado a la **Ciencia de Datos**.
- Manejo de los datos de manera **tabulada**.
- **Acceso** más sencillo que en Pandas (matriz **bidimensional**).
- **Selección** y **conversión** a una matriz de manera inmediata.
- **Almacenamiento** en los formatos más **estándar**:
 - CSV, Excel, ods
 - HDF5, JSON
 - JLD2 (propio): no requiere biblioteca C de HDF5

Sistemas de Aprendizaje Automáticos

Sistemas Lineales

Basado en Reglas

Árboles de Decisión

Deep Forest

Id3 J48

Bayesianos

Naive-Bayes

Redes Bayesianas

RLRMult

RLog

SVM

RED NEURONAL

MLP

RBF

SOM

CNN

LSTM

RNN

GRU

OneR

JRIP

Instancias

IBk

LWL

Clustering

K-medias

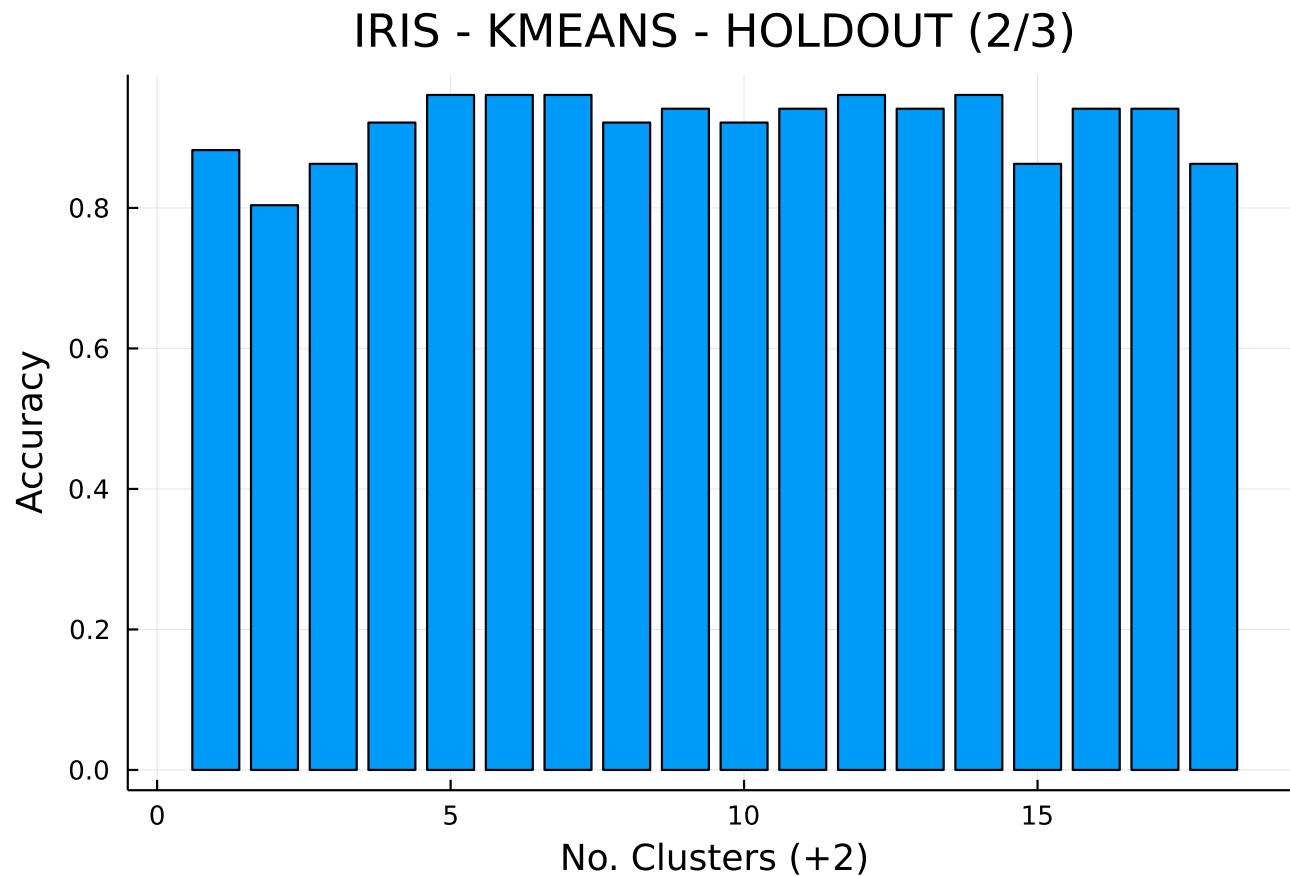
Jerárquicos

DBScan

Clustering

- Dataset: IRIS (MLDataUtils)
- ESTANDARIZACIÓN (StatsBase)
- CODIFICADO DE ETIQUETAS en ENTEROS
- HOLD-OUT (2/3 y 1/3) estratificado (MLDataUtils.stratifiedobs)
- Kmedias (Clustering)
- Clasificación (mode) y Predicción (Distances).
- Tasa de aciertos (mean)
-

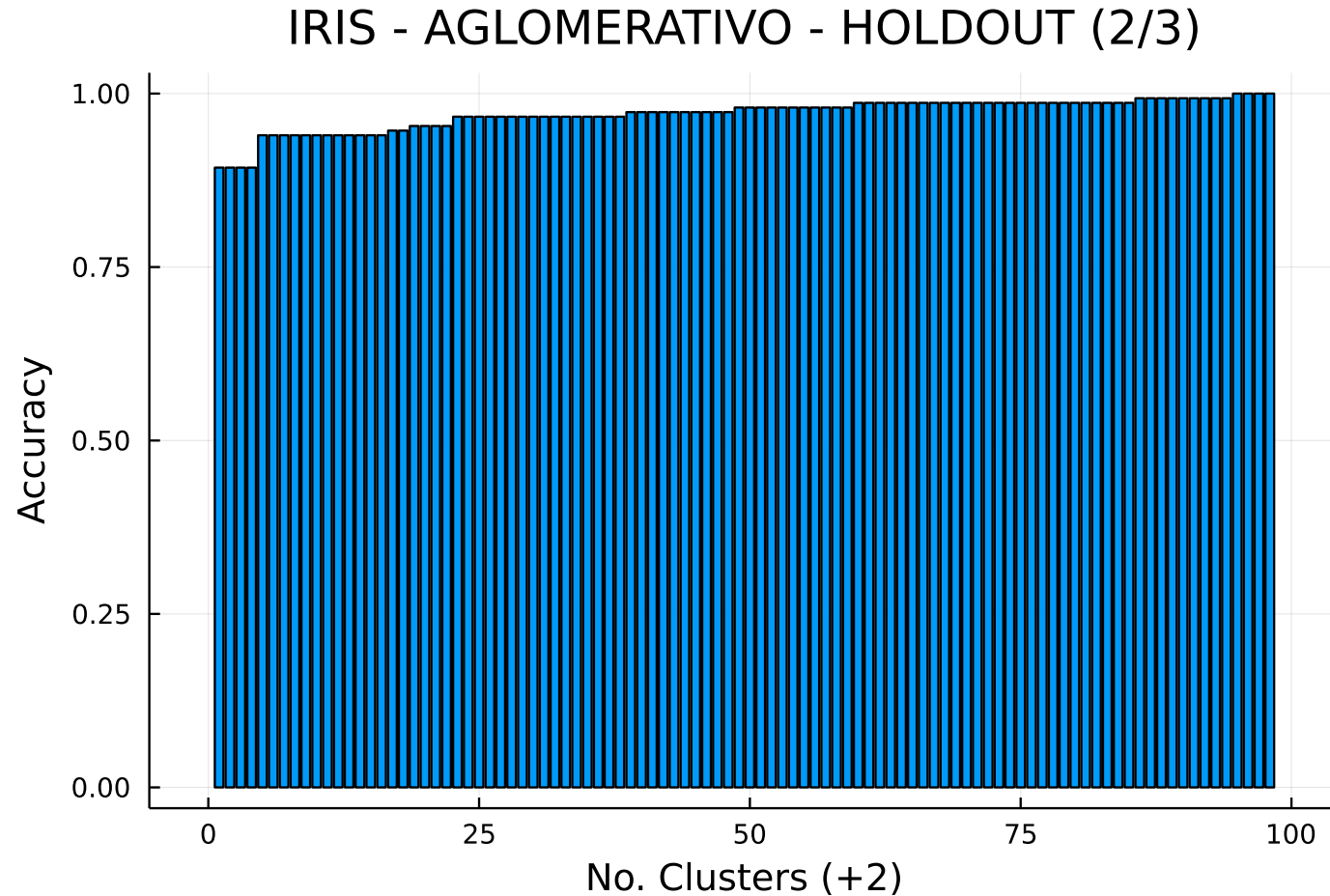
IRIS - KMEANS - HOLDOUT (2/3)



Clustering AGLOMERATIVO

- No hay división de muestras para entrenamiento/test
- `Clustering.hclust(D, :ward)`, D es la matriz de distancias entre muestras
- `cutree(Dendogram, k=n_cluster)`
- Calcular la tasa de aciertos entre la moda del cluster al que pertenece cada muestra y la clasificación deseada
- Variar el número de clusters entre 3 y 100.

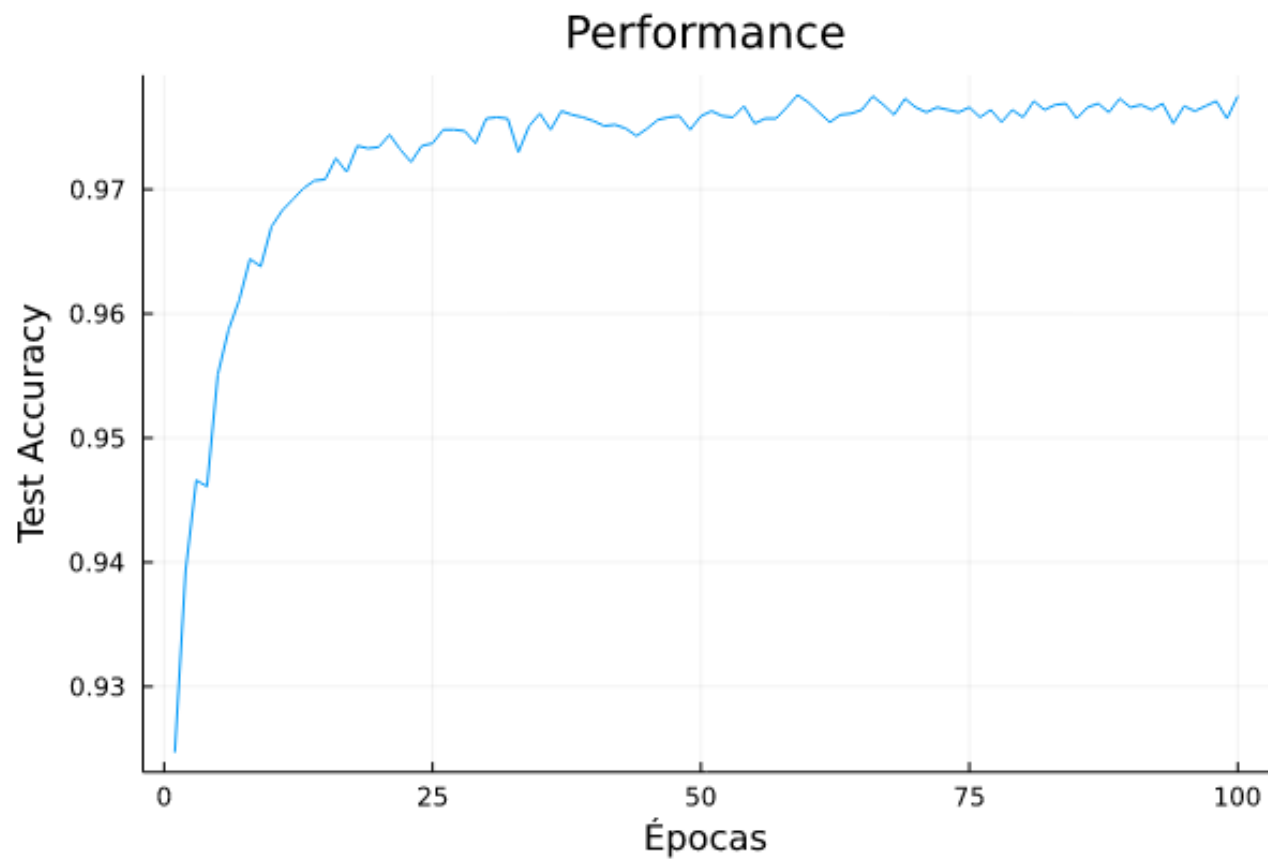
Tasa de aciertos entre moda del cluster / clase



MNIST con MLP (DENSE)

- CARGA DEL DATASET (`MLDatasets.MNIST`)
- Aplanado de las muestras (`reshape`)
- Salida deseada para 10 neuronas (`Flux.onehotbatch`)
- MLP con una capa oculta de 100 neuronas + Función de COSTE
- Función de COSTE (`Flux.Losses.logitcrossentropy`)
- OPTIMIZADOR (`Flux.Optimise.RMSProp`)
- Función ENTRENA: devuelve número de épocas y la secuencia de aciertos en cada época

Evolución Tasa Aciertos



Clasificación de Dígitos Manuscritos (CNN + MLP)

- Dataset: MNIST
- Preparación de muestras (matrices 4D) y salidas (onehot)
- Modelo (Flux): RED CONVOLUTIVA DE 3 CAPAS + DENSE (MLP) DE UNA CAPA OCULTA
- Optimizador (RMSProp)
- Cargador de datos - Definición de LOTES (Flux.Data.DataLoader)
- Función de coste o de pérdida (Flux.Losses.logitcrossentropy)
- Entrenamiento (Flux.train)

Tasa de Aciertos de las Redes Convolucionales

