Introducción a los métodos de aprendizaje automático

Universidad Católica del Uruguay

UT02 – Trabajo de Aplicación 2

TA2 Ejercicio 1 Estandarización / normalización

- descargar el dataset “wine” de UCI y abrir con excel
- revisar el problema planteado
- analizar la variable objetivo
- analizar los atributos y sus tipos de datos
- analizar técnicas para estandarizar / normalizar -
  - REMITIR UN DOC a la tarea UT02-TA2 en la webas con las decisiones de normalización / estandarización tomadas

TA2 Ejercicio 2

Utilizando el dataset “wine” , analizar los bloques apropiados para implementar las técnicas identificadas en el Ej. 1.

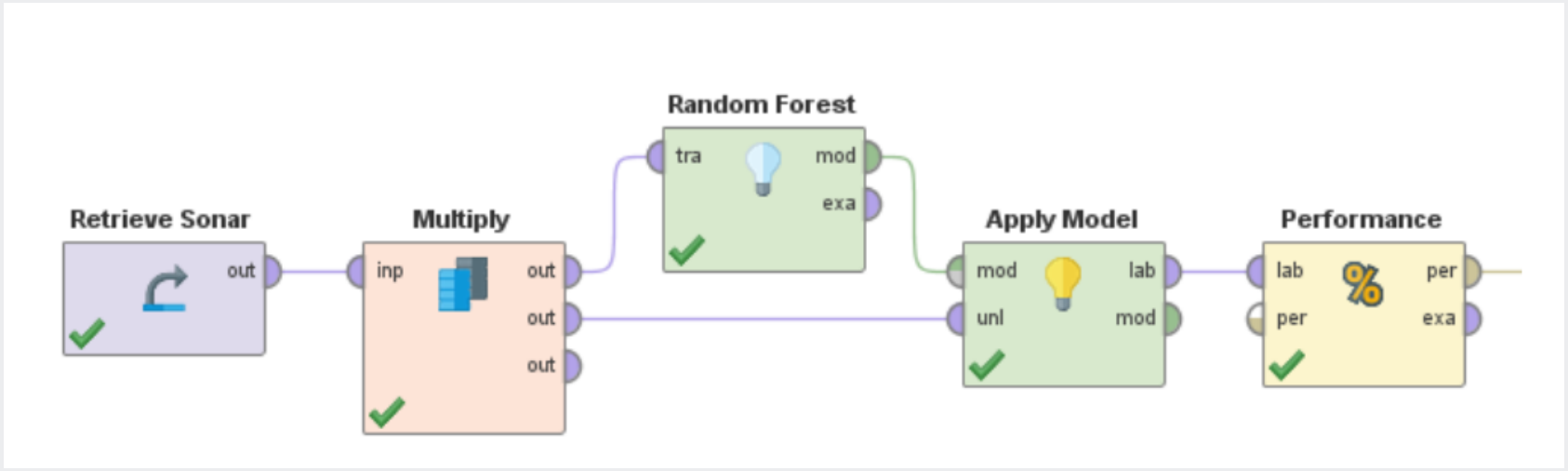
- Identificar los bloques de RM y escribir en un doc la descripción / parámetros de cada un (muy breve!)

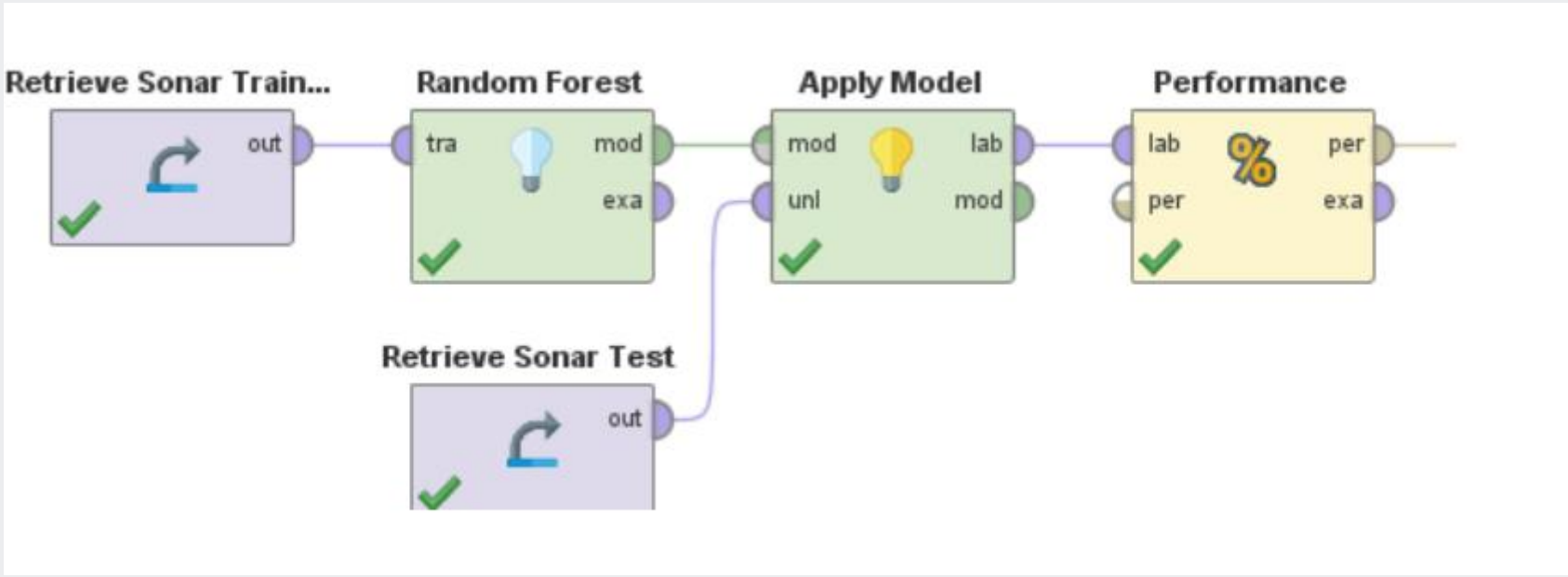
- Armar al menos 2 líneas de proceso para comparar con y sin estandarización / normalización

Training vs. Test Error

Universidad Católica del Uruguay

- Queremos saber qué tan bien se comportará nuesto modelo!
- Lo hacemos midiendo la exactitud, para seleccionar los mejores algoritmos y luego ajustar los parámetros para lograr mejores resultados.
- Validación del modelo:
  - Error de entrenamiento (training error)
  - Error de test (test error)

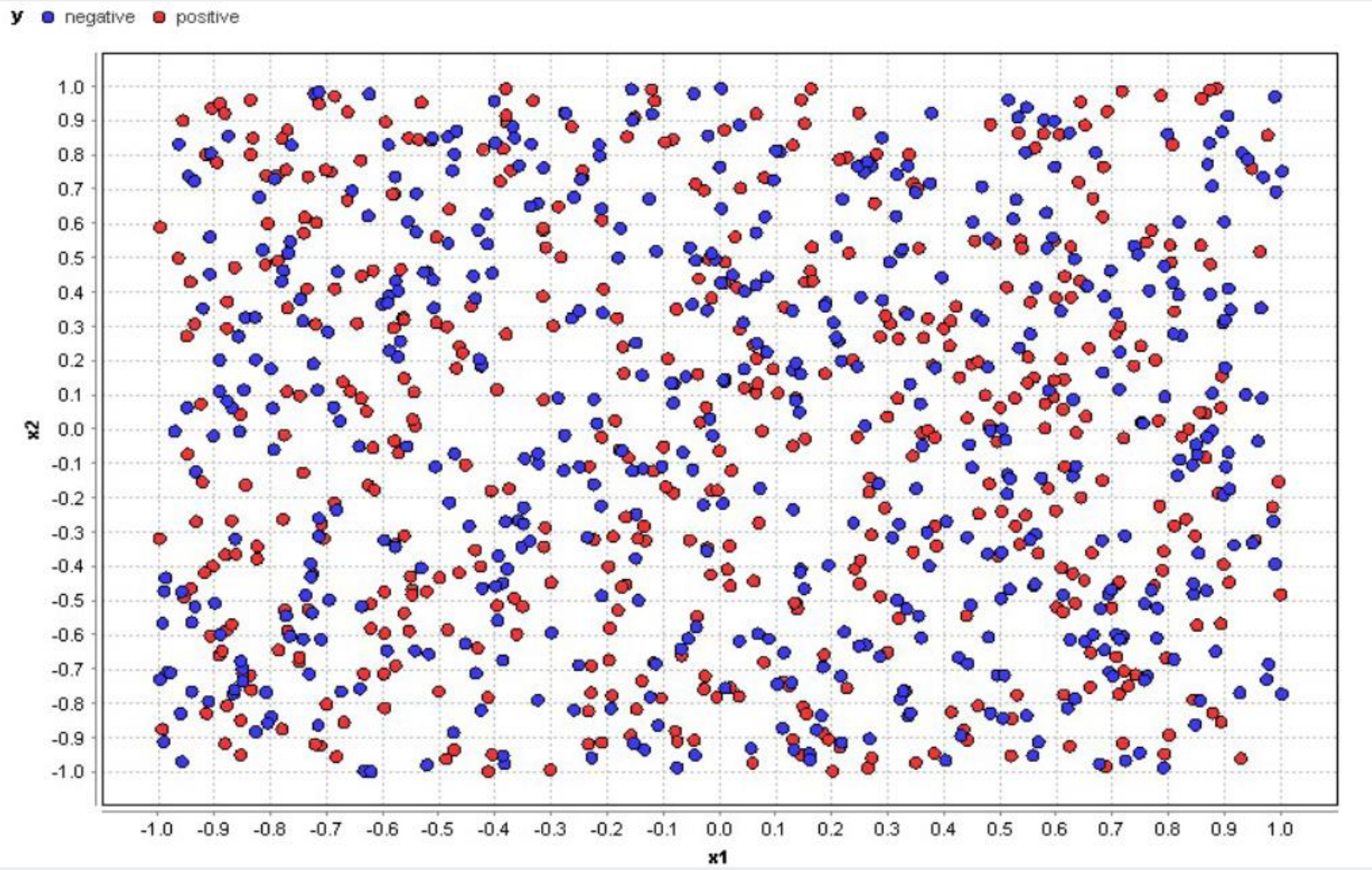

Random Forest
tra
mod
exa
Retrieve Sonar
out
Multiply
inp
out
out
out
Apply Model
mod
lab
unl
mod
Performance
lab
per
per
exa


Retrieve Sonar Train...
out
Random Forest
tra
mod
exa
Apply Model
mod
lab
unl
mod
Performance
lab
per
per
exa
Retrieve Sonar Test
out

Por qué se debe ignorar / descartar el error de entrenamiento

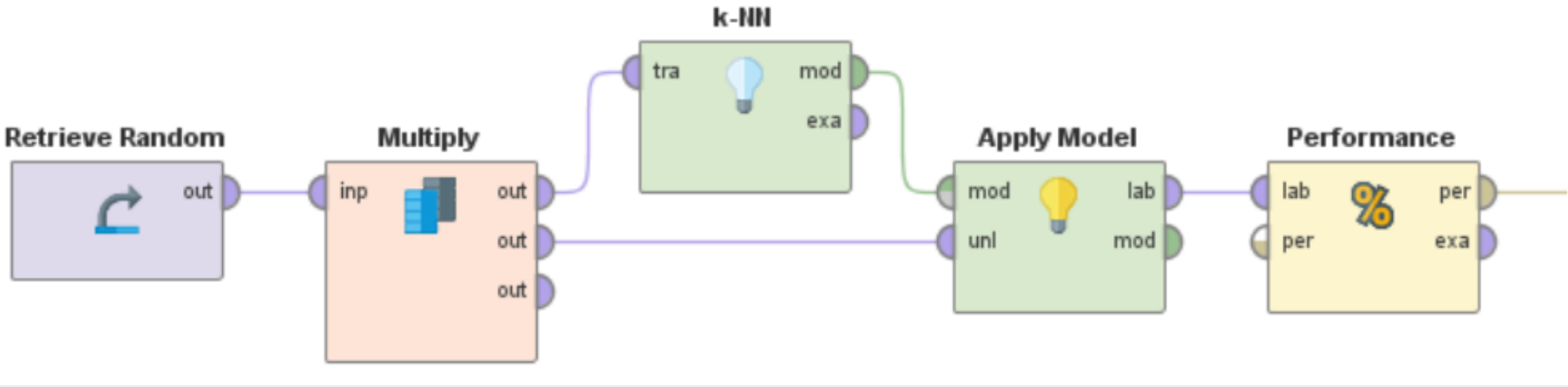
Universidad Católica del Uruguay

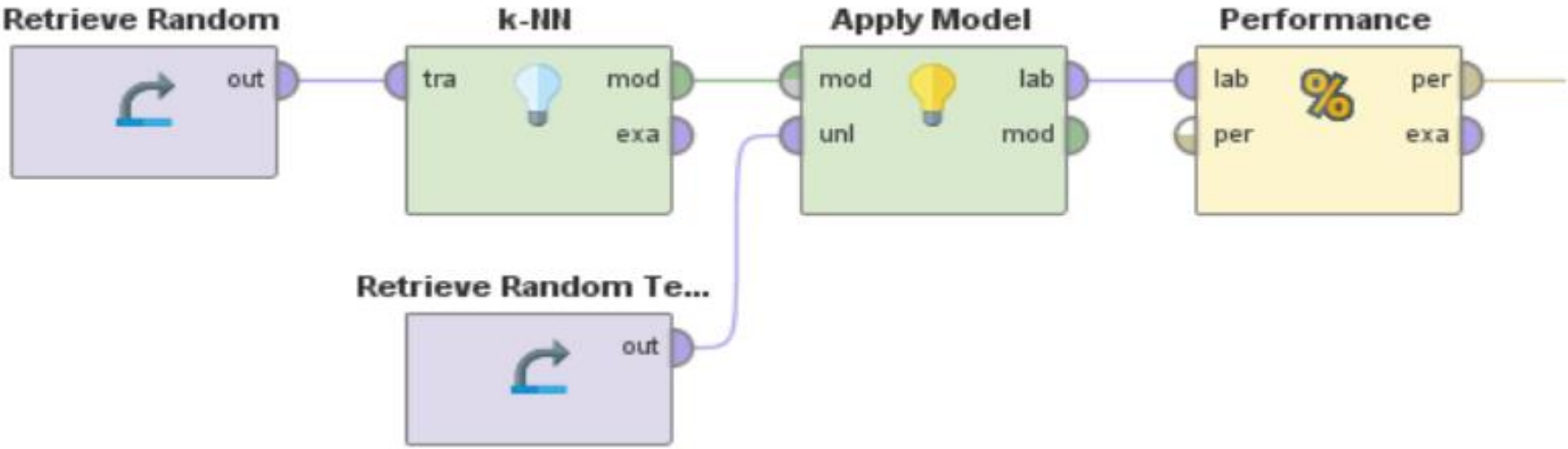
- A pesar de que encontraremos multiples referencias al error de entranamiento en la literature de ML, es una mala práctica
- Puede conducir a errores peligrosos
- Ejemplo con valores bidimensionales (rangos -1 a 1), igual cantidad de ejemplos “positivos” y “negativos”


y
negative
positive
x2
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
-0.1
-0.2
-0.3
-0.4
-0.5
-0.6
-0.7
-0.8
-0.9
-1.0
-1.0 -0.9 -0.8 -0.7 -0.6 -0.5 -0.4 -0.3 -0.2 -0.1 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
x1

- 500 ejemplos positivos y 500 negativos, distribuidos aleatoriamente
- ¿puede construirse un modelo predictivo?
- … lo mejor que se podría obtener sería 50%...

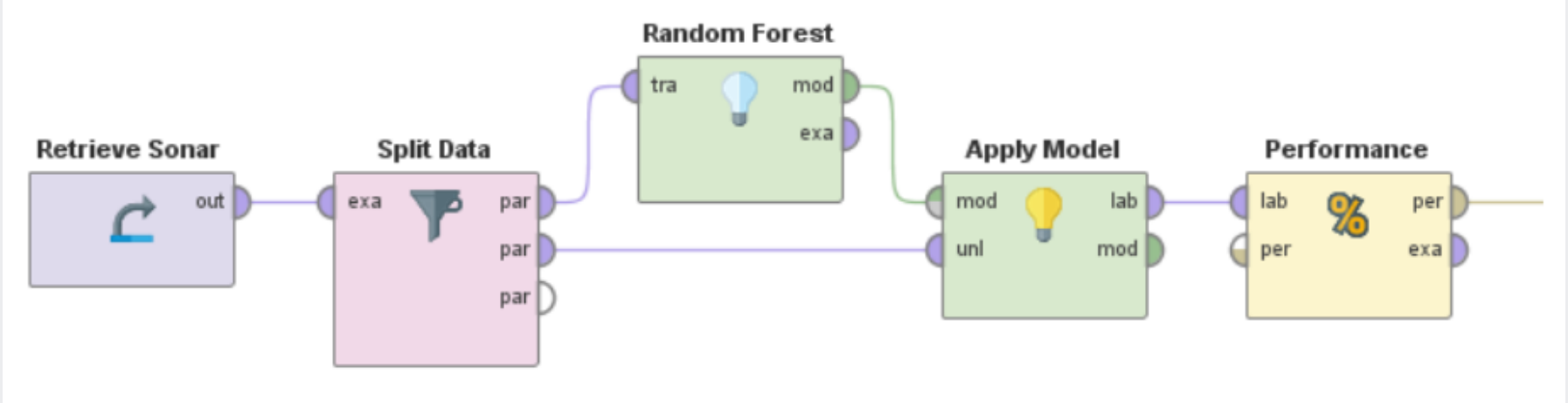
- Si aplicamos k-nn (con n = 1), obtendremos una exactitud de 100%


k-NN
tra
mod
exa
Retrieve Random
out
Multiply
inp
out
out
out
Apply Model
mod
unl
lab
mod
Performance
lab
per
per
exa


Retrieve Random
out
k-NN
tra
mod
exa
Apply Model
mod
unl
lab
mod
Performance
lab
per
per
exa
Retrieve Random Te...
out

Validación utilizando conjuntos hold-out

- Ya vimos que considerar el error de entrenamiento es una mala idea…
- Lo primero a tener en cuenta es que obtener o generar más datos puede ser difícil y costoso!
- Es práctica común entonces utilizar una parte de los datos disponibles para entrenamiento, y otra para test (“hold-out set” – “data split”)



TA2 Ejercicio 3

- Generar un proceso con “split validation” (con parámetros estándar 70/30), para medir el rendimiento en los dos “canales” (con y sin normalización / estandarización)
- utilizar un algoritmo “k-nn” con parámetros estándar
- ejecutar los modelos y registrar los resultados de performance de los dos “canales” en una planilla simple comparativa