Rapid Miner outlier detection and normalization tutorial.

Se aplica normalización para luego aplicar el cálculo de las distancias para detectar a los outliers. Siempre se debe normalizar previo al cálculo de los outliers por el hecho de que los diferentes atributos suelen no tener la misma unidad o escala. Para la normalización se utiliza la transformación z.

El operador detect outliers detectará los 10 ejemplos que se alejen más del resto. En el flujo, luego colocamos un filter para remover los 10 outliers antes mencionados.

Ejercicio 2

El dataset posee 13 atributos sin incluir a la variable objetivo. Estos describen distintos elementos de la composición de vinos para luego poder clasificarlos.

La variable objetivo posee tres valores posibles. Estos valores representan a 3 diferentes variantes de viñedos (las plantas de las que se genera el vino son diferentes genéticamente entre sí).

La variable objetivo está relativamente balanceada, no hay una clase que presente un numero muy bajo de ejemplos, por lo que argumentaría que no habría que aplicar ningún tipo de sampling.

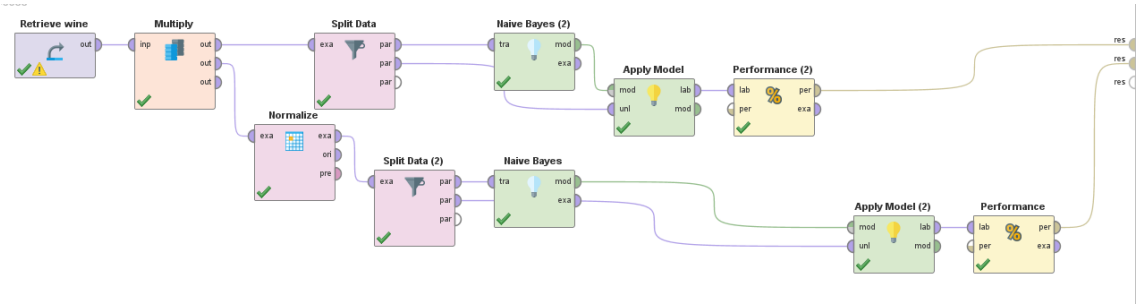
Para algunos atributos, la distribución parece ser normal (ash, alcalinity of ash, proanthocyanins), mientras que algunas otras parecen tener un sesgo hacia la derecha (por ejemplo malic acid y proline), pero no es muy grave tampoco.

Creación del modelo con naive bayes.

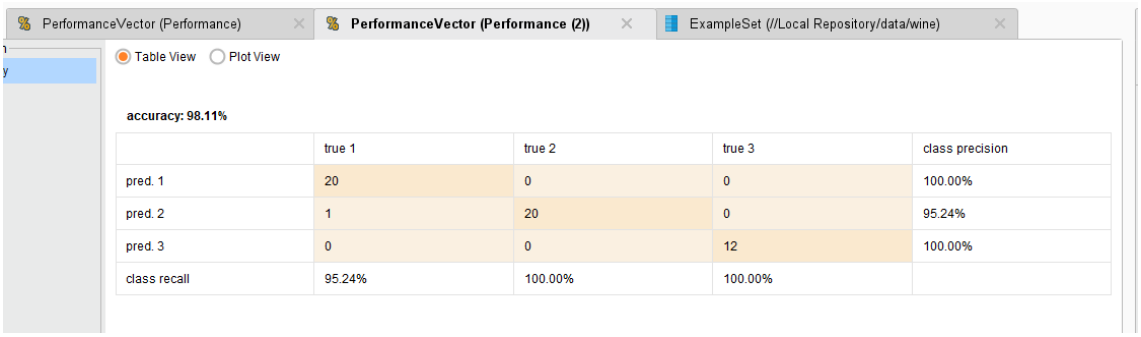
Al visualizar objetivamente la matriz de confusión, los resultados no variaron al aplicar la normalización. Con ambos tipos de normalización (z score y min-max), los resultados fueron los mismos.

Flujo de rapidminer:

El para los Split data, se utilizó la misma seed y la misma proporción (70/30) para que los conjuntos de entrenamiento y testeo sean los mismos para ambas pruebas, con la diferencia que uno estará normalizado y otro no.

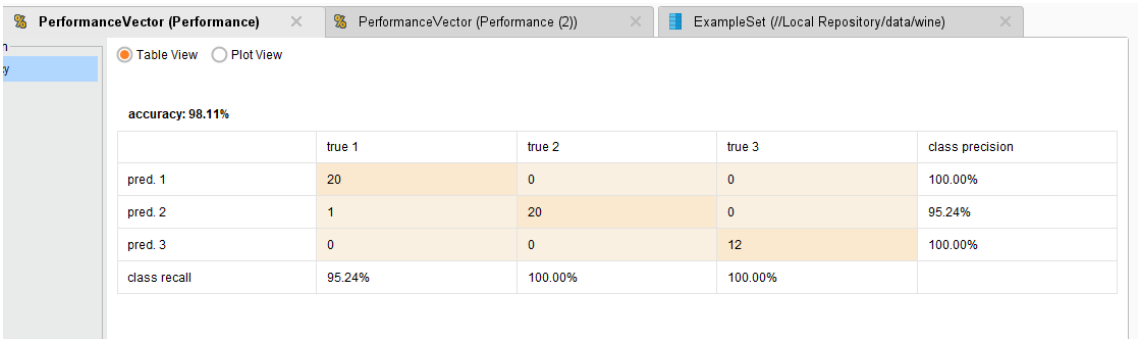


Matriz sin normalizado



	true 1	true 2	true 3	class precision
pred. 1	20	0	0	100.00%
pred. 2	1	20	0	95.24%
pred. 3	0	0	12	100.00%
class recall	95.24%	100.00%	100.00%	

Matriz con normalizado



	true 1	true 2	true 3	class precision
pred. 1	20	0	0	100.00%
pred. 2	1	20	0	95.24%
pred. 3	0	0	12	100.00%
class recall	95.24%	100.00%	100.00%	

Conclusiones

El normalizado no afecta los resultados para este caso en particular. Esto se debe al algoritmo que utilizamos principalmente.