PRÁCTICA 9. APRENDIZAJE AUTOMÁTICO

Raquel Blanco Morago

Celia Calvo González

GRUPO 16

# Descripción del conjunto de datos

## Clustering: Seeds

Atributos:

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Tipo** | **Valores válidos** |
| Área | REAL |  |
| Perímetro | REAL |  |
| Compacidad | REAL |  |
| Longitud de la semilla | REAL |  |
| Anchura de la semilla | REAL |  |
| Coeficiente de asimetría | REAL |  |
| Longitud del surco de la semilla | REAL |  |
| Tipo de semilla (salida) | NOMINAL | 1 -> Kama, 2 -> Rosa, 3 -> Canadian |

Describe las peculiaridades de semillas para resolver el problema de clustering, con el que podemos agrupar y sacar conclusiones.

URL: <https://archive.ics.uci.edu/ml/datasets/seeds>

## Clasificación: Car Evaluation

Atributos:

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Tipo** | **Valores válidos** |
| Coste | NOMINAL | alto, bajo, mediano, muy\_alto |
| Tamaño | NOMINAL | alto, bajo, mediano, muy\_alto |
| Puertas | INTEGER |  |
| Capacidad | NOMINAL | 1,2,3,4, más |
| Tamaño maletero | NOMINAL | grande, mediano, pequeño |
| Seguridad | NOMINAL | alto, bajo, mediano |
| Evaluacion (salida) | NOMINAL | aceptable, bueno, inaceptable, muy bueno |

Describe las características de vehículos para resolver el problema de clasificación, con el que se obtiene como salida una evaluación del coche, que indica cómo de bueno o de malo es el vehículo.

URL: <https://archive.ics.uci.edu/ml/datasets/car+evaluation>

## Regresión: AUTO MPG

URL: <https://archive.ics.uci.edu/ml/datasets/Auto+MPG>

Atributos:

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Descripción** | **Tipo** |
| MPG (salida) |  | REAL |
| Cilindradas |  | REAL |
| Desplazamiento |  | REAL |
| Caballos |  | REAL |
| Peso |  | REAL |
| Aceleración |  | REAL |
| Década fabricación |  | REAL |
| Origen |  | REAL |
| Nombre |  | NOMINAL |

Describe la especificación de automóviles en cuanto a su consuma y sus características. Es un buen data set para tartar el problema de regresión.

# APARTADO 2. CLASIFICACIÓN

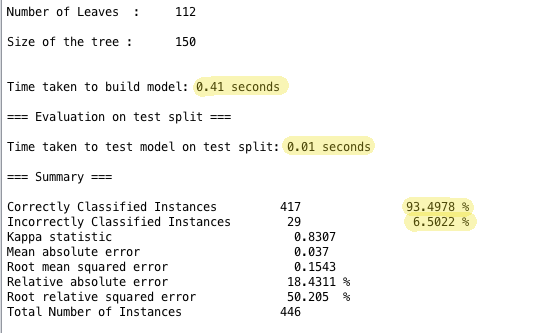
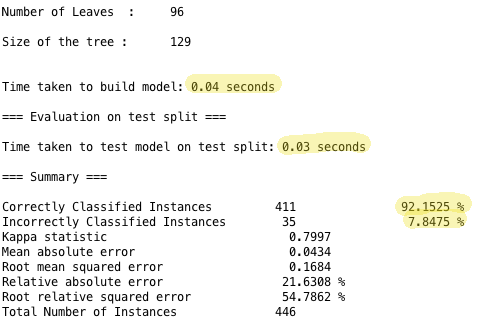
### 1

No hay que normalizar las variables porque todas son nominales, tampoco estandarizar, por la misma razón.

### 2

Ejecutamos J48 con el Percentage split del 66%, con poda 0.35 y minNumObj = 3.

Después de probar varias configuraciones, hemos llegado a la conclusion de que no vale la pena ramificar en una nueva hoja cuando van a clasificarse menos de 3 instancias.



**Percentage Split con Poda=0.35 minNumObj =3.**  **Percentage Split con Poda=0.8 minNumObj =3**

### 3

Ejecutamos J48 con el Training set, si poda. Tras probar varias configuraciones, hemos visto que es mejor utilizando nimNumObj = 2 o 3. Hemos probado también con minNumObj = 5 y el error aumenta considerablemente.

IMAGEN DE LOS DOS

### 4

* **Matriz de confusión Percentage Split**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **c** | **d** | **Recall (Verdaderos positivos)** | **Precisión** |  |
| 323 | 19 | 0 | 0 | 0,944 | 0,988 | **a** |
| 4 | 83 | 0 | 0 | 0,954 | 0,755 | **b** |
| 0 | 5 | 3 | 0 | 0,375 | 0,429 | **c** |
| 0 | 3 | 4 | 2 | 0,222 | 1 | **d** |

Analizando la precisión de las 4 clases, la probabilidad de clasificar coches en las clase a y d es casi del 100% por lo que podríamos confiar en caso de que nos claseifique un coche en alguna de estas clases. En cambio, en b y c, no podemos asegurarnos de que se vaya a hacer una buena clasificación, por lo que tendríamos que consultar a un experto o añadir más instancias.

Analizando los verdaderos positivos, vemos que en las clases a, b tiene un alto porcentaje de acertar positivos, pero en las clases c y d se aciertan muy pocos verdaderos positivos, la mayoría de los positivos no los acierta.

Todos los que ha predicho que iban a ser de la clase d los ha acertado (1), pero ha dado por positivos muchos que no pertenecían a la clase (0,22). Predice que pertenecen a la clase d más de los que deberian, para evitar esto se necesitaría especificar más las características para ser de la clase d.

Puede que una solución a esta mala predicción a la clasificación en las clases c y d sea la introducción de nuevas atributos que describan a estas clases, sin que puedan llegar a afectar a las clases a y b, que sí que realizan una buena predicción. También la introducción de nuevas instancias de las clases c y d, ya que se tienen muy pocas en este data set.

Esta sería la marena en la que se comportaría con la llegada de nuevos individuos.

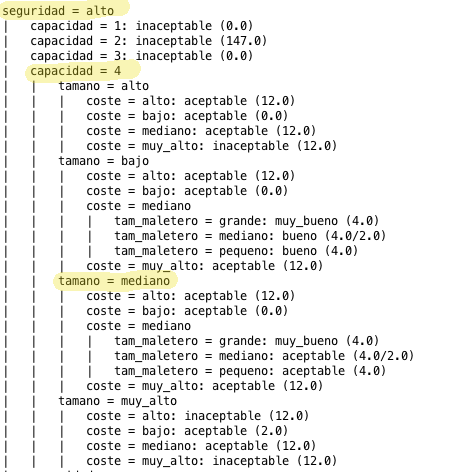
* **Matriz de confusión Training set**:

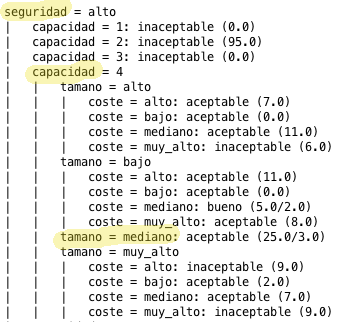
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **c** | **d** | **Recall (Verdaderos positivos)** | **Precisión** |  |
| 947 | 17 | 1 | 0 | 0,981 | 0,993 | **a** |
| 7 | 288 | 1 | 1 | 0,970 | 0,932 | **b** |
| 0 | 2 | 20 | 1 | 0,870 | 0,833 | **c** |
| 0 | 2 | 2 | 22 | 0,846 | 0.975 | **d** |

En este caso, la matriz de confusión nos indica que el entrenamiento del árbol funciona muy bien ya que el número de verdaderos positivos es casi del 100% y el de precisón también.

La mayoría de los individuos los clasifica en la clase que le corresponde.

### 5

Sobreaprende más el training set porque no valida los datos, solo hace el árbol para los individuos que tiene.



ÁRBOL PERCENTAGE SPLIT ÁRBOL TRAINING SET

Observando el árbol que se produce, podemos ver como con el training set predice mejor la clase muy\_bueno, en cambio, con el percentage split se predice bastante mal la clase muy\_bueno, tiene más fallos.

Podemos ver que en la clase muy\_bueno, muchos de los individuos no se clasifican bien en esta clase, por lo que debemos buscar en el árbol de percentage split hojas en las cuales la clasificación en alguna clase tenga un mayor número de fallos, correspondiendo estos fallos a los de la clase muy\_bueno. Estos fallos se deben a que utilizando training set, aplica mayor número de filtros, por lo tanto, hay menor número de errores.

Viendo los resultados generales, es más fiable utilizer training set por el hecho de que tiene mayor número de individuos con los que hace el árbol, pero si en el percentage split utilizaramos validación cruzada éste sería major ya que nos ofrecería una visión más acertada de lo que pasaría en una situación real. Esto es debido a que el percentage split realiza varios árboles de training, los cuales son distintos unos de otros, que se asemeja al número de individuos que se utilizan en un training set solo pero añade la mejora de que valida esos árboles como si de un caso real se tratara.

La primera pregunta que se realiza en el nodo raíz es el tipo de seguridad del vehículo, por la cual se lleva a cabo una clasificación bastante importante, separando en tres hojas, de las cuales una es terminal y asigna 437 individuos a la clase inaceptable. Las otras dos ramas continúan la clasificación.

Estas dos otros nodos hijos de la raíz (seguridad alta y media) se pregunta por la capacidad y en ambos, clasifica a los individuos de capacidad =2 como inaceptable, siendo 147 individuos en cada rama.

Con esas pocas preguntas, ya se llevaría más de la mitad de la muestra (473+147+147=731), ya que la muestra tiene 1311 instancias.

La clase que mejor se clasifica es la de inaceptable, que es la que primero se clasifica y de una forma más rápida. A partir de ahí se ramifica más y los nodos terminales como mucho clasifican 12 individuos, a excepción de los de la clase inaceptable, que pueden llegar hasta 16.

Seguridad es la variable más relevante porque genera la clasificación con mayor número de individuos, al igual que capacidad, que también clasifica un gran número de individuos.

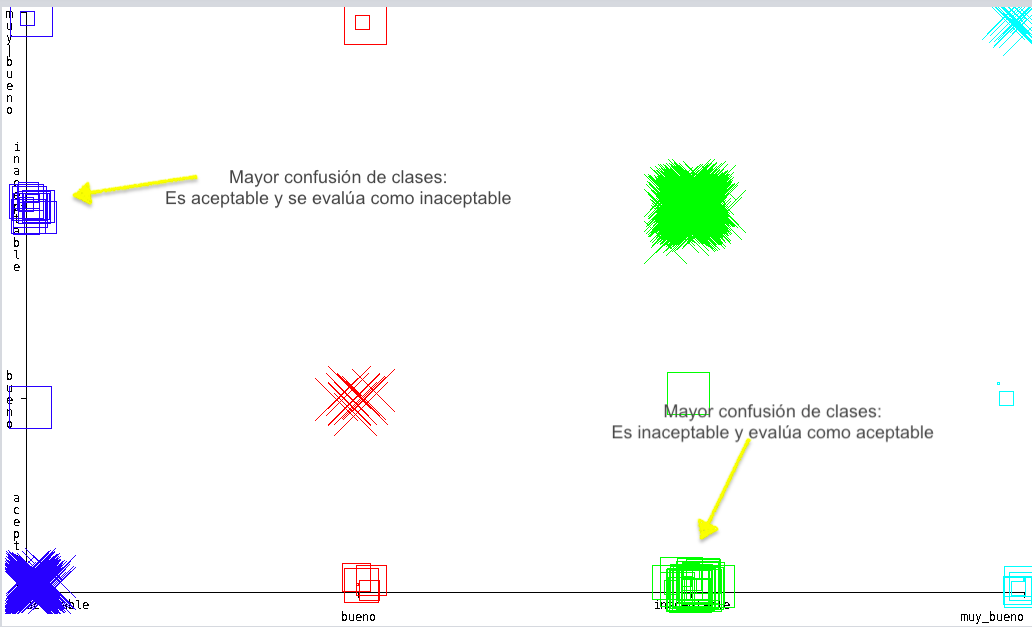
La entropía de todo el árbol es muy baja, ya que las clases de individuos están muy diferenciadas.

El árbol siempre elige variables diferentes a lo largo de las distintas ramas.

Interpretación del árbol:

* Un coche con seguridad baja es inaceptable, eso es obvio.
* Los coches con seguridad media y capacidad menor o igual de dos, también son inaceptables.
* El tamaño del maletero es un poco irrelevante.
* Los mejores coches son los pequeños.
* Un coche grande no es una buena opción, a no ser que sea de coste mediano o alto, bajo o muy alto no.
* El tamaño va relacionado con el coste. Según este data set, es mejor un coche pequeño y caro, que uno grande y caro.

Clases mayormente confundibles:



En las clases que se produce mayor confusión entre sí a la hora de predecir son en aceptable e inaceptable, donde ambas se confunden en algunas ocasiones.

Observando las características de las precicciones de evaluación que se confunden entre sí, son las que están sobre todo la rama de seguridad mediano, capacidad 4 o más y tamaño de maletro mediano. Esto ocurre porque el tamaño del maletero es irrelevante cuando la seguridad es mediana y la capacidad de personas es grande, no aporta nueva información, sólo ramifica.

Con lo cual, el nodo-pregunta seguridad es el más destacable ya que clasifica exactamente un tercio de la muestra. También cabe destacar el nodo-pregunta capacidad, que clasifica gran parte de la clase inaceptable.

El nodo-pregunta menos relevante es el ya dicho tamaño del maletero, que no clasifica bien e incluso sobraría, ya que confunde la gran parte de sus predicciones.