

Entregable WEKA

Laura Rodríguez Navas
rodrigueznava@posgrado.uimp.es

Mayo 2020

Preparación de datos

Consideramos la base de datos Prostate definida sobre 12600 variables predictivas (todas numéricas) y una variable clase binaria {Tumor, Normal}. Está formada por 136 registros y en ella no existen valores desconocidos. Pero está ordenada en función de la variable clase. Como consecuencia, tenemos que aleatorizar la base de datos. Para ello se aplica un filtro a nivel de registro, concretamente de tipo no supervisado llamado *Randomize*. Usando los parámetros por defecto después de aplicar el filtro comprobaremos que la base de datos se ha aleatorizado.

Analizando un poco la base de datos vemos que el número de registros disponibles (136) es bajo en comparación con el número de atributos (12601). Así, vamos a tratar con un problema de alta dimensionalidad y poca muestra.

Para la clasificación se utilizarán los clasificadores Naive Bayes y J48 (C4.5). El clasificador Naive Bayes es muy buen clasificador en problemas con un número elevado de variables predictoras, como en este caso.

Clasificación

Primero se usan los clasificadores Naive Bayes y J48 (C4.5) en una validación cruzada de 5 carpetas (5cv) sobre el conjunto inicial de la base de datos, obteniendo así una medida inicial que se intentará mejorar.

Se consideran dos parámetros de rendimiento para la evaluación de los resultados de las clasificaciones. Los parámetros son los registros clasificados correctamente (Accuracy) y los registros no clasificados correctamente (Error Rate).

Después de realizar la primera clasificación obtenemos los siguientes resultados:

Clasificador	Acc. en %	ERR en %
Naive Bayes	51.1471	44.8529
J48 (C4.5)	80.1471	19.8529

Inicialmente y sin la aplicación de algún tipo mejora, podemos observar que el algoritmo J48 (C4.5) nos da una clasificación aceptable y clasifica mucho mejor en comparación con el algoritmo Naive Bayes. El algoritmo Naive Bayes no hace una buena clasificación. Se puede pensar que se debe a la alta presencia de variables relevantes y redundantes en la base de datos, ya que sabemos que Naive Bayes es un algoritmo muy sensible a este caso.

Mejoras

En este apartado se muestran mejoras que se basan en técnicas de discretización, de selección de variables y una combinación de ambas.

Discretización

Como la variable clase juega un papel importante y los clasificadores que utilizamos pueden ser entrenados de manera muy eficiente con aprendizaje supervisado, vamos a utilizar una técnica de discretización supervisada. Concretamente, probamos un método supervisado a nivel de atributo basado en la entropía, llamado MLDP. Como resultado, los valores de Accuracy y Error Rate después de esta discretización son:

Clasificador	Antes Disc.		Después Disc.	
	Acc. in %	ERR in %	Acc. in %	ERR in %
Naives Bayes	51.1471	44.8529	76.4706	23.5294
J48 (C4.5)	80.1471	19.8529	86.7647	13.2353

Observamos que es una buena discretización para los dos algoritmos. Como nos ha dado un buen resultado, vamos a combinar esta discretización con técnicas de selección de variables del siguiente apartado.

Selección de variables

Sabemos que la base de datos contiene muchos atributos y que eso crea un problema, así que para una estimación honesta primero tenemos que reducir el número de atributos de la base de datos. Por ejemplo, para quedarnos con los 200 más representativos y la variable clase, vamos a realizar una selección univariada de tipo Ranker (ver Figura 1).

Hecha la selección de los 200 atributos más representativos de la base de datos, donde el primer atributo más representativo es el 6185 y el último es el 6776, borraremos el resto de los atributos usando un filtro de registro de tipo no supervisado llamado *Remove*.

Volvemos a clasificar y vemos que mejora mucho la clasificación de los dos algoritmos si la comparamos con la clasificación inicial, y también mejoran después de la discretización. Los resultados se observan en la siguiente tabla:

Clasificador	Inicial		Ranker	
	Acc. in %	ERR in %	Acc. in %	ERR in %
Naives Bayes	51.1471	44.8529	90.4412	9.5588
J48 (C4.5)	80.1471	19.8529	91.1765	8.8235

Concretamente, la mejora de Naive Bayes es muy significativa. Eso nos demuestra que, al reducir el número de atributos, hemos reducido el número de variables relevantes y redundantes de la base de datos; ya que como hemos comentado anteriormente, Naive Bayes es un algoritmo muy sensible en este caso.

```

Attribute selection output

=== Attribute Selection on all input data ===

Search Method:
  Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 12601 Class):
  Gain Ratio feature evaluator

Ranked attributes:
0.463      6185 37639_at
0.3389     8965 37720_at
0.3385     4365 41468_at
0.3229     8850 37366_at
0.3085     5757 36491_at
0.3039     10138 41288_at
0.2934      299 31538_at
0.2925     6866 39756_g_at
0.2914     12148 575_s_at
0.2834     6865 39755_at
0.2829     8528 36204_at
0.2829     4438 41661_at
0.2754     2839 37068_at
0.2744     8747 37023_at
0.2744     8593 36628_at
0.2744     3394 38581_at
0.2684     3649 39315_at
0.2672     8306 35742_at
0.2658     12351 338_at
0.2658     5847 36821_at
0.2658     6534 38706_at
0.2658     6605 39016_r_at
0.2658     7622 40000_at

```

Figura 1: Selección de variables univariada de tipo Ranker.

A continuación, lanzaremos dos selecciones de variables multivariadas de tipo Filter y Wrapper.

Primero, lanzamos la selección de variables multivariada de tipo Filter, que se caracteriza por una técnica de búsqueda CFS (Correlation Feature Subset), y una función de evaluación en selección hacia delante (forward). En este caso la selección de atributos se ve ampliamente reducida a 18 atributos (ver Figura 2).

Después de quedarnos con los 18 atributos más representativos y al volver a clasificar obtenemos:

Clasificador	Ranker		Filter	
	Acc. in %	ERR in %	Acc. in %	ERR in %
Naives Bayes	90.4412	9.5588	95.5882	4.4118
J48 (C4.5)	91.1765	8.8235	91.1765	8.8235

Otra vez más, el algoritmo Naives Bayes mejora al reducir el número de atributos. En cambio, no hay mejora del algoritmo J48 (C4.5). Podemos decir que el algoritmo J48 (C4.5) no es tan sensible a la reducción de atributos redundantes o irrelevantes, y aunque sigamos reduciendo el número de atributos no habrá mejoras. Lo comprobaremos en la aplicación de la selección multivariada de tipo Wrapper que describimos a continuación.

Finalmente, lanzamos una selección multivariada de tipo Wrapper para cada clasificador, con la función de evaluación en selección hacia delante (forward) y obtenemos los resultados:

- Naive Bayes. La selección de variables se reduce a 5: {1, 6, 11, 12, 18}.

Clasificador	Filter		Wrapper	
	Acc. in %	ERR in %	Acc. in %	ERR in %
Naives Bayes	95.5882	4.4118	97.7941	2.2059

- J48 (C4.5). La selección de atributos se reduce a 2: {4, 6}.

Clasificador	Filter		Wrapper	
	Acc. in %	ERR in %	Acc. in %	ERR in %
J48 (C4.5)	91.1765	8.8235	91.1765	8.8235

Como conclusión, nos quedamos con la última clasificación ya que es la mejor para los dos algoritmos, los valores de Accuracy son los más altos y los valores de Error Rate los más bajos. Trabajar con variables redundantes o irrelevantes nos puede perjudicar en la capacidad predictiva de nuestro algoritmo. Por ese motivo, algoritmos sensibles a esto, como Naive Bayes, las técnicas de selección de variables producen muchas mejoras en las clasificaciones al reducir el número de este tipo de atributos.

```

Attribute selection output

=== Attribute Selection on all input data ===

Search Method:
  Greedy Stepwise (forwards).
  Start set: no attributes
  Merit of best subset found: 0.595

Attribute Subset Evaluator (supervised, Class (nominal): 201 Class):
  CFS Subset Evaluator
  Including locally predictive attributes

Selected attributes: 1,3,5,6,10,13,16,18,31,43,51,58,67,83,98,101,123,141 : 18
37639_at
41468_at
36491_at
41288_at
39755_at
37068_at
38581_at
35742_at
1664_at
37502_at
34602_at
32598_at
39608_at
35082_at
36691_at
41346_at
35127_at
496_s_at

```

Figura 2: Selección de variables multivariada de tipo Filter.