Parte VI

Combinación de clasificadores

Capítulo 15

Combinación de clasificadores

En la actualidad, el estado del arte en minería de datos combina dos aproximaciones diferentes. La primera consiste en el desarrollo de nuevos algoritmos (lo cual ocurre muy ocasionalmente), o bien en el ajuste fino de los parámetros de algún algoritmo conocido, habitualmente para adaptarlo a la naturaleza concreta de unos datos o del problema a resolver. En este sentido, los avances en el área de minería de datos están asegurados por el llamado *no free lunch theorem* [31], el cual postula que no existe un algoritmo *a priori* que sea superior al resto para cualquier conjunto de datos.

La segunda aproximación para crear mejores modelos consiste en combinar clasificadores más o menos sencillos para crear uno mucho más complejo, de forma que la decisión tomada sea una combinación de cientos o miles de decisiones parciales. Obviamente, los clasificadores usados como base para construir el clasificador combinado deben ser lo más diversos posible, con la esperanza de que los errores cometidos por

un clasificador base concreto sean minoritarios con respecto al resto, de forma que los errores puntuales no alteren una decisión basada en la opinión correcta de la mayoría. Una revisión reciente de los trabajos en esta área de investigación puede encontrarse en [32].

Existen dos opciones para la construcción de clasificadores combinados. La primera, que combina clasificadores trabajando en paralelo, es cuando todos los clasificadores base utilizan el mismo modelo o algoritmo (p. ej. árboles de decisión), así que será necesario manipular el conjunto de entrenamiento original para generar diferentes clasificadores base y poder tomar una decisión conjunta a partir de la decisión parcial de cada uno de ellos. La segunda opción consiste en combinar clasificadores base muy diferentes (p. ej. árboles de decisión y redes neuronales) de forma secuencial, de forma que cada clasificador utilice los resultados de un clasificador anterior, intentando capturar alguna característica clave de la naturaleza de los datos o del problema a resolver.

15.1. Combinación paralela de clasificadores base similares

Como se ha comentado, una opción es generar una gran cantidad de clasificadores base construidos a partir de la alteración del conjunto de entrenamiento original. Todos estos clasificadores base son muy parecidos, al estar basados en ligeras variaciones del conjunto de entrenamiento, pero no son idénticos, proporcionando diversidad al clasificador combinado, el cual combina (mediante un esquema de votación) las clasificaciones parciales para tomar una decisión.

Existen dos técnicas básicas para la creación del clasificador combinado, en función de cómo se genera el conjunto de entrenamiento de cada clasificador parcial y del peso asignado a cada uno de ellos en la votación final, llamadas *bagging* y *boosting*.

15.1.1. Bagging

La idea básica del *bagging* es utilizar el conjunto de entrenamiento original para generar centenares o miles de conjuntos similares usando muestreo con reemplazo. Es decir, de un conjunto de N elementos se pueden escoger $N' \leq N$ al azar, existiendo la posibilidad de escoger un mismo elemento más de una vez (de ahí «con reemplazo»). Normalmente $N' = N$, por lo que existirán elementos repetidos.

Aunque no es tan habitual, los conjuntos generados también pueden usar una dimensionalidad $d' \leq d$, de forma que no todas las mismas variables disponibles existan en los conjuntos generados. Esto puede ayudar a reducir el grado de colinealidad entre variables, haciendo emerger variables relevantes que pueden quedar siempre descartadas en frente de otra variable.

Una vez se han construido los conjuntos de entrenamiento parciales, se construye un clasificador para cada uno de ellos, siendo los árboles de decisión la opción más típica. De hecho, teniendo en cuenta la naturaleza del clasificador combinado, no es necesario crear clasificadores base muy precisos, ya que el objetivo es que los errores cometidos por cada clasificador base queden minimizados en frente de la decisión de la mayoría. Por lo tanto, en el caso de árboles de decisión, lo que es habitual es crear clasificadores más pequeños (es decir, limitados en

profundidad) reduciendo el coste computacional de todo el conjunto.

En el caso del *bagging*, la decisión final se toma por mayoría, dando el mismo peso a todas las decisiones parciales. Es decir, la clase resultante del clasificador combinado es aquella que aparece más veces entre las decisiones parciales tomadas por los clasificadores base. La figura 15.1 representa el proceso de creación del clasificador combinado.

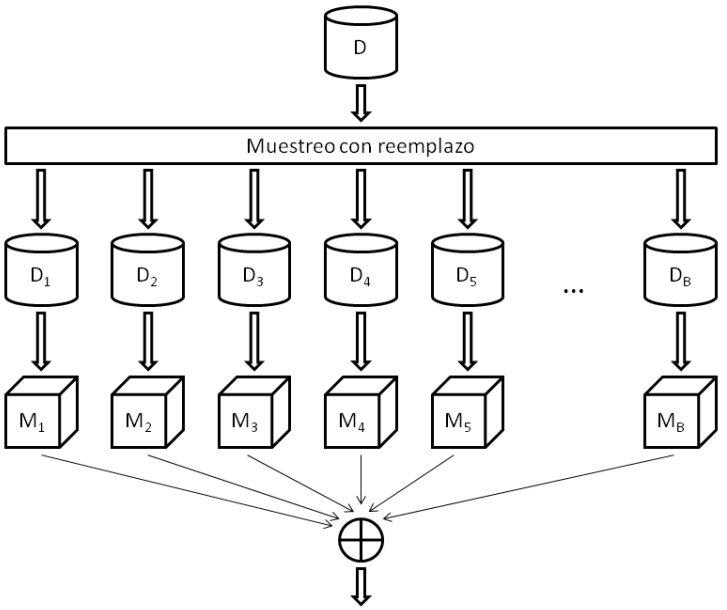

Figura 15.1. Diagrama de un clasificador combinado basado en *bagging*

Una manera de medir el error cometido por el clasificador combinado se conoce como *out-of-bag*, dado que el error estimado es el promedio de todos los errores parciales cometidos por cada clasificador base para todos aquellos elementos no usados en el conjunto de entrenamiento usado para construir-

lo. De esta manera no es necesario separar los datos de entrada en un conjunto de entrenamiento y otro de test, sino que se usan todos para construir el clasificador combinado. No obstante, si se dispone de suficientes datos, es siempre recomendable utilizar un conjunto de test para validar el clasificador combinado.

Random forests

Cuando los clasificadores base son árboles de decisión y se utiliza un muestreo tanto de los elementos del conjunto original de entrenamiento como de sus variables, el clasificador combinado se conoce como *Random forest* [3], dado que se trata precisamente de un conjunto (o *bosque*) de árboles que han sido creados mediante un proceso aleatorio (por lo que respecta al conjunto de entrenamiento usado para cada uno de ellos).

La práctica habitual consiste en generar versiones diferentes del conjunto de entrenamiento usando muestreo con reemplazo, tal y como define el método de *bagging*. Entonces, durante el proceso de construcción de cada árbol de decisión se selecciona aleatoriamente un subconjunto de las variables del conjunto de datos, dando opciones a variables que normalmente quedarían eclipsadas por otras que tuvieran mayor relevancia. Este procedimiento permite medir la importancia relativa de cada variable, estimando el error cometido por el clasificador combinado cuando se altera dicha variable, permutando aleatoriamente sus valores en el conjunto de test. Este error se mide para cada uno de los clasificadores parciales, promediando el error cometido en todos ellos para cada una de las variables. El porcentaje de error se compara al estimado con el conjunto de test sin dicha permutación aleatoria, de forma

que es posible medir el impacto de dicha variable, dado que si el error aumenta cuando se permuta una variable, esto quiere decir que la variable es relevante para el problema que se está resolviendo.

Se recuerda al lector que en el material adicional a este libro se puede encontrar un ejemplo completo donde se muestra el proceso de creación de un *random forest* usando Jupyter y R.

15.1.2. Boosting

La idea del *boosting* es ligeramente diferente. Se parte también de clasificadores base muy sencillos (también llamados débiles) de los cuales se supone que, al menos, cometen menos errores que aciertos, es decir, que funcionan ligeramente mejor que un clasificador aleatorio.

Se empieza construyendo un primer clasificador base usando el conjunto de entrenamiento original. Como el clasificador es débil (por ejemplo, un árbol de decisión de profundidad limitada), se cometen unos cuantos errores para el conjunto de entrenamiento. Entonces, para construir el siguiente clasificador base, lo que se hace es ponderar los elementos del conjunto de entrenamiento, de forma que aquellos que han sido clasificados erróneamente por el clasificador anterior tengan más peso y, de una manera u otra, tengan mayor peso también en el proceso de creación del siguiente clasificador base. El clasificador combinado es la combinación de la predicción del primer clasificador base más la del segundo, ponderadas de acuerdo a algún esquema de pesos que tenga en cuenta la precisión de cada clasificador. Este clasificador combinado también cometerá unos errores que se usarán para dar más peso a aquellos elementos erróneamente clasificados, construyendo un tercer clasificador combinado, etc.

Es decir, en cada etapa se construye un clasificador nuevo que combina una serie de decisiones sucesivas que tienden a enfocarse en aquellos elementos más difíciles de clasificar, intentando que cada etapa corrija los errores de la anterior, pero sin estropear las decisiones correctas ya tomadas. Para ello, la secuencia de pesos de los sucesivos clasificadores parciales suele ser descendiente, de forma que el procedimiento se para cuando el siguiente clasificador ya no aporta nada respecto al anterior. Este esquema, mostrado en la figura 15.2, permite múltiples variaciones, siendo la más conocida la descrita por Freund and Schapire en 1996 [8], llamada AdaBoost.

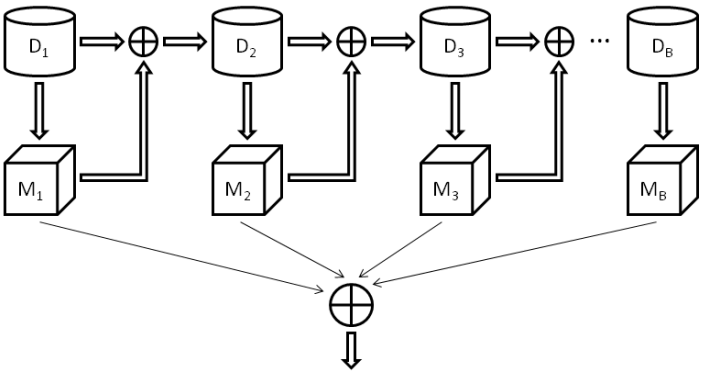

Figura 15.2. Diagrama de un clasificador combinado basado en *boosting*

Es importante destacar que el proceso de creación del clasificador combinado es secuencial, dado que para realizar una nueva iteración y obtener un nuevo clasificador combinado es necesario haber evaluado los clasificadores base anteriores. No obstante, una vez se ha alcanzado el clasificador combinado final, la evaluación se realiza también en paralelo, ya que en una primera etapa se genera la decisión parcial para cada uno

de los clasificadores base y en la segunda se obtiene la clasificación final ponderando todas la decisiones parciales.

Uno de los problemas de *boosting* es la posibilidad de sobreentrenar el clasificador combinado, por lo que es necesario disponer de un conjunto de test e ir evaluando en cada etapa el resultado obtenido sobre el mismo, deteniendo el proceso cuando el error en el conjunto de test empieza a aumentar.

En el material adicional a este libro se puede encontrar un ejemplo completo donde se muestra el proceso de aplicación de *boosting* con árboles de decisión usando Jupyter y R.

15.2. Combinación secuencial de clasificadores base diferentes

En este caso el objetivo es construir un clasificador que combina clasificadores base muy diferentes, con el objetivo de incrementar la diversidad de predicciones e intentar aprovechar todo el conocimiento explícito que se tenga sobre el problema a resolver o los datos disponibles. Por ejemplo, si se utiliza un árbol de decisión como clasificador, se pueden construir unos cuantos clasificadores que funcionen como una etapa previa, de forma que alimenten el árbol de decisión con información extraída de los datos u otras decisiones parciales.

Se trata, entonces, de un proceso secuencial, dado que se generan unas cuantas decisiones parciales que posteriormente son usadas para tomar la decisión final. En función de la información que recibe el clasificador combinado de los clasificadores parciales, se distinguen dos métodos, llamados *stacking* y *cascading*.

Estas técnicas son adecuadas cuando la naturaleza del problema a resolver también presenta una estructura secuencial.

Por ejemplo, en el diagnóstico no invasivo de tumores cerebrales mediante el uso de espectroscopia de resonancia magnética [17], en lugar de intentar predecir el tipo de tumor detectado directamente (lo cual es muy complicado porque existen muchos tipos de tumores diferentes y de diferente grado de malignidad), lo habitual es establecer una secuencia de decisiones parciales intentando atacar el problema desde una primera decisión muy sencilla (p. ej. establecer simplemente si se trata de un tumor o no), seguida de una segunda decisión que determina si se trata de un tumor benigno o maligno, seguida de una tercera que determina el grado de malignidad. Es decir, en cada paso se utiliza la decisión tomada anteriormente, empezando por una primera decisión sencilla que se va refinando en una secuencia de decisiones cada vez más específicas.

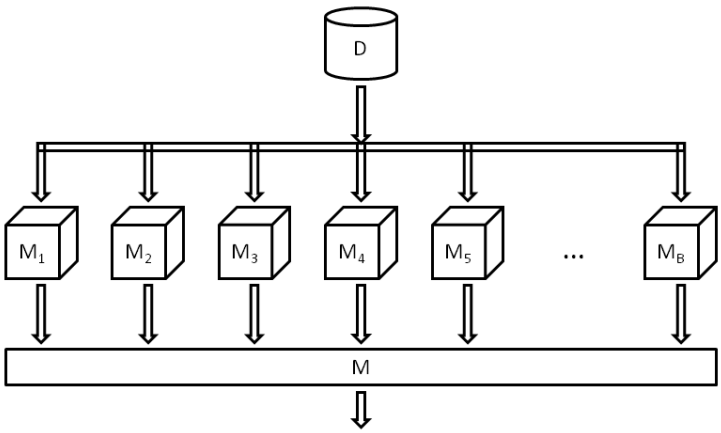

Figura 15.3. Diagrama general de un clasificador combinado basado en *cascading* y *stacking*

En general, se puede pensar en la combinación secuencial como una manera de abordar el problema, intentando resolver primero los casos más sencillos con un clasificador también

sencillo, dejando el resto a una secuencia de clasificadores cada vez más complejos y específicos.

15.2.1. Stacking

La idea básica de *stacking* es construir diferentes clasificadores base de forma que cada uno de ellos genere una decisión parcial para cada elemento de entrada del conjunto de entrenamiento. Entonces se construye un nuevo clasificador usando como datos de entrada todas las predicciones parciales, en lugar de los datos originales de entrada. Este segundo clasificador suele ser un árbol de decisión, una red neuronal sencilla o una regresión logística (en el caso de que el número de clases sea dos).

Aunque no es habitual, esta estructura puede repetirse en diferentes niveles, combinando decisiones de otros clasificadores combinados, de ahí la idea de apilamiento o *stacking*. El problema de siempre es la tendencia a crear un clasificador combinado demasiado específico para el conjunto de entrenamiento que no generalice bien ante nuevos datos.

15.2.2. Cascading

El caso de *cascading* es parecido al de *stacking* pero utilizando no solamente las predicciones parciales de los clasificadores base, sino también los datos originales e incluso otros datos que se hayan podido generar durante la toma de decisiones. La idea básica es alimentar al clasificador combinado con decisiones parciales, así como los motivos que han llevado a tomar dichas decisiones.

Por ejemplo, cuando un árbol de decisión ha asignado una clase a un elemento del conjunto de entrada, dicha clase es el

resultado de una serie de decisiones internas que llevan a una hoja, la cual tiene una profundidad en el árbol, representa una región que contiene un porcentaje de los datos de entrada y se conoce el error cometido al asignar dicha clase como representante de todos los datos de entrada contenidos en la región que representa dicha hoja. O por ejemplo, si uno de los clasificadores base es un algoritmo de *clustering*, se puede usar como información adicional a la clase o clúster asignado a un elemento del conjunto de entrada cada una de las distancias a cada uno de los centroides de los clústeres, como medidas de la fuerza que tiene dicha decisión.

15.3. Resumen

El uso de clasificadores combinados permite, en muchas ocasiones, mejorar la capacidad predictiva de un modelo, siendo posible también la inclusión de conocimiento relativo a la naturaleza del problema a resolver, especialmente con los clasificadores que operan secuencialmente. Su construcción es sencilla, utilizando en muchas ocasiones los mismos algoritmos y técnicas (reemplazo con muestreo, sistemas de votación por mayoría simple, etc.), siendo el caso de *boosting* el más complejo, aunque existen diferentes algoritmos que lo implementan.

No obstante, los clasificadores combinados también presentan una serie de problemas, algunos ya mencionados a lo largo de este libro. El primer punto a tener en cuenta es el coste computacional que puede tener el clasificador combinado, tanto por lo que respecta a su creación (especialmente) como a su ejecución. No obstante, es posible aprovechar la estructura paralela de los clasificadores combinados en caso de disponer de infraestructura tecnológica que permita la paralelización

de procesos, de forma que cada clasificador base se ejecuta en un nodo en paralelo (a la vez) a todos los otros clasificadores base, reduciendo así el tiempo de ejecución necesario.

Un aspecto a tener en cuenta es que no por aumentar el número de clasificadores base involucrados se va a conseguir siempre una mejora de la precisión. Es mucho más importante la diversidad de los clasificadores base que su número. Como la diversidad depende del conjunto de entrenamiento usado para cada clasificador base, es importante que el conjunto de entrenamiento original sea suficientemente grande y diverso y que el proceso de muestreo asegure un buen grado de aleatoriedad.

Por otra parte, la interpretabilidad de los resultados es mucho más complicada, ya que exige la interpretación de cientos o miles de clasificadores base parciales y su posterior combinación. No obstante, es posible medir la importancia relativa de cada variable en el conjunto mediante diferentes técnicas (p. ej. contando cuántas veces aparece cada variable en cada clasificador base con su peso en el clasificador combinado), lo que puede proporcionar cierto conocimiento al respecto.