Clasificación basada en ensembles

UCO ONLINE

Clasificación basada en ensembles

UNIVERSIDAD Ð CÓRDOBA

1. Introducción

Hasta el momento, prácticamente todos los métodos estudiados para resolver problemas de clasificación se basaban en la creación de un único modelo que diese solución al problema. Sin embargo, los métodos denominados ensembles o multi-clasificadores se basan en la creación y combinación de varios clasificadores, de modo que la salida o predicción final para un patrón dado, se basa en la combinación de las salidas de varios modelos más simples.

Cuando en lecciones anteriores estudiábamos, por ejemplo, el enfoque *one-vs-rest* para clasificadores lineales en escenarios multi-clase, realmente estábamos construyendo lo que puede considerarse un modelo de ensemble. En ese caso, estábamos construyendo k clasificadores binarios, y la salida final del clasificador venía dada por la combinación de las salidas de dichos k modelos.

La principal idea u objetivo detrás del aprendizaje basado en ensembles, es que, teniendo varios modelos (en principio, más simples que si entrenásemos un único modelo), de algún modo diversos entre sí, cada uno de ellos podría estar más centrado en una zona diferente del espacio de entrada. De este modo, la combinación de las salidas de estos clasificadores se espera que sea mejor que las predicciones de cada uno de ellos por separado, ya que se reduciría la probabilidad de un único clasificador produciendo una salida errónea para un patrón concreto. Sin embargo, habría que construir el ensemble con cuidado ya que, si por ejemplo incluimos varios clasificadores mucho más débiles o pobres en cuanto a sus predicciones, podríamos obtener el efecto contrario.

Como estudiaremos a lo largo de la lección, la diversidad en los distintos métodos del ensemble se puede conseguir de distintos modos: utilizando distintos clasificadores base, utilizando distintos patrones de entrenamiento, o atributos de entrada en cada uno de ellos, utilizando distintos parámetros para cada uno de los modelos, etc. A lo largo de la lección estudiaremos algunos paradigmas de aprendizaje basado en ensembles genéricos y más utilizados (Bagging, Boosting, y Stacking); así como múltiples métodos de combinación de las salidas de los distintos clasificadores.

Cabe destacar que los métodos basados en ensembles son mucho más amplios a los que se estudiarán en esta sección, y que están sujetos a infinidad de variaciones. Según las necesidades de un problema específico, se pueden utilizar o combinar distintos métodos de generación de diversidad o de combinación de las salidas.

2. Bagging

Bagging es un paradigma estándar y genérico de clasificación basada en ensembles, donde por lo general se utiliza un único método de aprendizaje para todos los modelos a combinar en el ensemble. La diversidad en cada uno de ellos se alcanza por medio de seleccionar subconjuntos de los datos de entrenamiento para cada uno de ellos. La decisión final, por lo general, viene dada por el voto mayoritario entre todos los clasificadores.

La selección o muestreo de instancias para cada uno de los clasificadores se suele hacer siguiendo uno de dos tipos de muestreo distintos:

- Sin reemplazo. Cada clasificador incluye un subconjunto de los patrones de entrenamiento seleccionados aleatoriamente sin repetición. Por lo general, de entre 66-80% de los datos originales, aunque no tiene por qué seguir esta regla.
- Con reemplazo. El muestreo de los patrones de entrenamiento se hace de tal forma que un mismo patrón pueda seleccionarse más de una vez. Por lo general, el muestreo se suele hacer del mismo tamaño, y se espera que en torno al 63% de los datos muestreados sean únicos, y el resto repetidos.

Sobre cada uno de los muestreos, se construye un clasificador, cada uno de los cuales producirá una salida. El tamaño del ensemble (es decir, número de clasificadores base), es una cuestión recurrente en la literatura, y que también dependerá de cada problema específico. Por lo general, un ensemble de 10 clasificadores suele ser un tamaño aceptable, pero de nuevo, va a depender siempre de la complejidad o características de nuestro problema concreto.

Una vez teniendo las salidas de los n clasificadores para un nuevo patrón, la predicción final se produce por voto mayoritario; es decir, la clase con más votos entre los clasificadores base es la predicha finalmente.

El efecto esperado gracias a Bagging es el que se muestra en la Figura 1. Dados 3 clasificadores o modelos distintos en el ejemplo, cada uno de ellos se entrena con un subconjunto de los datos de entrenamiento; en la Figura, suponemos que se utilizan para entrenar cada modelo los patrones en color. Cada modelo produciría una frontera de decisión, que al combinarlas, daría una frontera similar a la que se observa abajo a la derecha. De este modo, se espera que la frontera obtenida finalmente gracias a la combinación de los distintos modelos tenga un mejor rendimiento en test que cada uno de los modelos base por separado.

Nótese que en Bagging, todos los clasificadores pueden construirse en paralelo, ya que son independientes unos de otros.

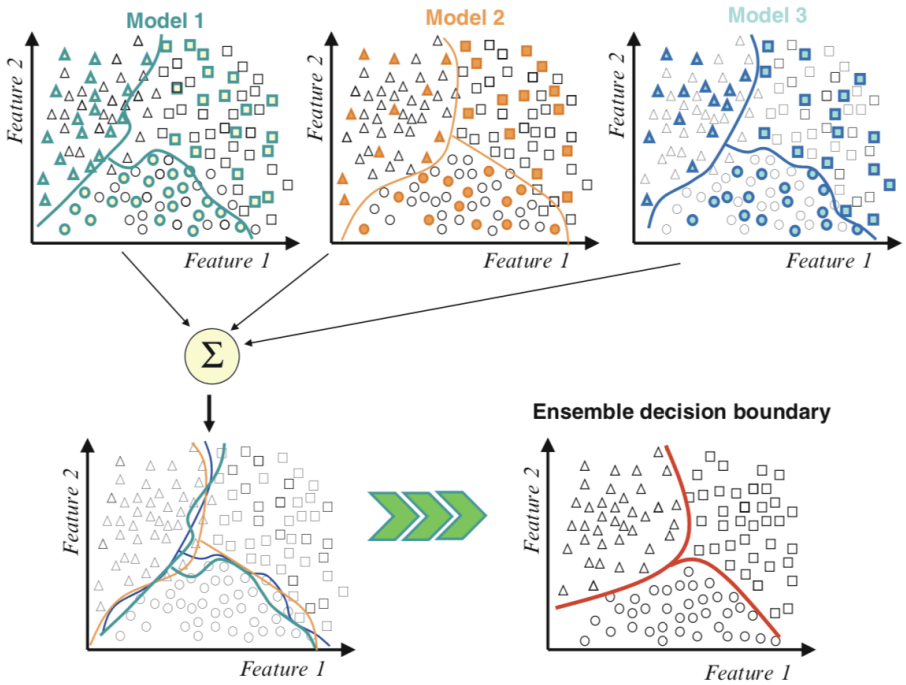

Figura 1. Ejemplo gráfico del objetivo y funcionamiento de Bagging.[1]

3. Boosting

Los métodos de Boosting siguen un paradigma distinto al visto anteriormente. Al contrario que en Bagging, donde los distintos modelos base podrían entrenarse de forma paralela, Boosting sigue un proceso iterativo, que se ilustra en la Figura 2.

En primer lugar, Boosting entrena un primer clasificador tal y como lo haría en cualquier escenario. Una vez entrenado, se identifican aquellos patrones que se predicen incorrectamente por el modelo, y se les incrementa el peso o importancia de estos patrones para la siguiente iteración (mientras que los pesos del resto se reducen por lo general). El siguiente modelo se entrenará sobre el conjunto de datos modificado, donde existen patrones con una mayor importancia a otros. Al generar este segundo modelo, se repite el mismo proceso anterior: se identifican los patrones clasificados incorrectamente por este clasificador, se les aumenta el peso o importancia, y se vuelve a entrenar otro clasificador. Este proceso se repite hasta obtener el número de clasificadores deseado.

[1] Fuente: https://machinelearningmastery.com/how-ensemble-learning-works/

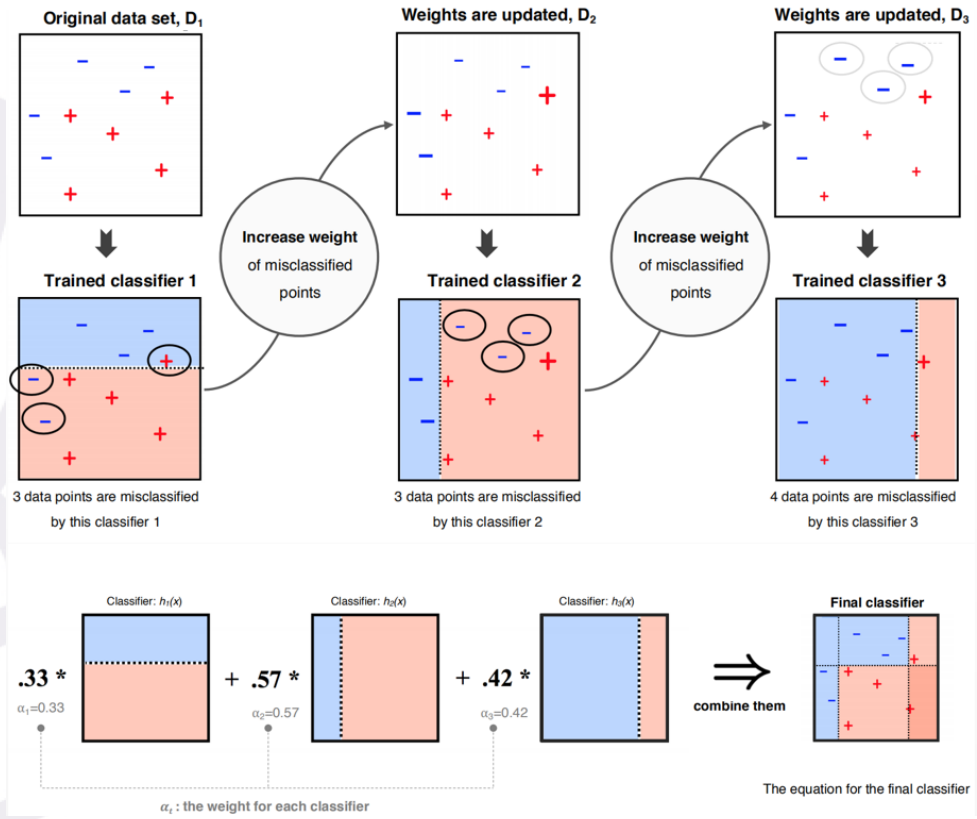

Figura 2. Ejemplo de funcionamiento de Boosting.[2]

Con este proceso, lo que se persigue es que en cada iteración los modelos se enfoquen más en patrones que los modelos anteriores no fueron capaces de clasificar correctamente, de modo que, al combinar las predicciones de todos ellos, el ensemble tenga información centrada en distintos fragmentos del espacio de entrada.

La clasificación final del ensemble vendrá dada por la combinación de salidas de todos los clasificadores. Además, por lo general, en Boosting la salida de los distintos clasificadores base suele estar ponderada por un valor identificativo de la calidad de dicho modelo. De este modo, si por el proceso de pesado de los patrones se obtienen modelos con un peor rendimiento predictivo, tendrán también menos peso en la clasificación final.

[2] Fuente: https://www.codenong.com/cs107125018/

4. Stacking

El último de los 3 paradigmas más extendidos para la generación de ensembles es el apilamiento o stacking. Los métodos de stacking se basan principalmente, en definir otro proceso para la combinación, o para tener en cuenta las salidas, de los distintos clasificadores base.

Los métodos de stacking se dividen en dos fases. En la primera, se construyen n métodos que, como siempre, sean diversos entre sí. En la segunda fase, se entrena un nuevo clasificador (solo uno), donde en este caso se tomará como entrada las salidas de los clasificadores anteriores. Existen dos formas principales de construir el conjunto de datos para el clasificador de la segunda fase:

1. Utilizar únicamente las salidas de los n clasificadores como atributos de entrada. Es decir, el conjunto de datos tendrá, para cada patrón, n atributos (correspondientes con la salida de los modelos anteriores), y el valor de clase original asociado a dicho patrón.
2. Concatenar las salidas de los n clasificadores como atributos de entrada. En este caso, los patrones de entrada para el segundo clasificador siguen conservando sus características originales, pero además, se les añaden las salidas de los clasificadores anteriores.

En otro métodos de combinación de las salidas de los clasificadores base del ensemble se utiliza únicamente una función simple (media, voto mayoritario, etc.) para obtener la predicción final. La idea tras el método de stacking es construir un modelo algo más complejo que sea capaz de aprender de dichas salidas y modelar mejor el atributo de clase en base a los clasificadores base.

En la Figura 3 se trata de ejemplificar el funcionamiento del stacking. Como se observa, en este caso se entrenan 3 clasificadores (que pueden ser métodos de distinto tipo, distintos parámetros, distintos subconjuntos de entrenamiento, etc.); cada uno de ellos produce sus predicciones de "primer nivel". A partir de estas predicciones de primer nivel (y considerando, o no, los atributos originales), se entrena un meta-clasificador o clasificador de segundo nivel, que proporcionará la salida final del ensemble. La principal diferencia con otros métodos de ensemble es la existencia de este meta-clasificador en lugar de otro método más sencillo de combinación de las salidas.

Los clasificadores del primer nivel podrían generarse en paralelo, mientras que el meta-clasificador tendría que generarse necesariamente una vez que los del primer nivel ya se han construido.

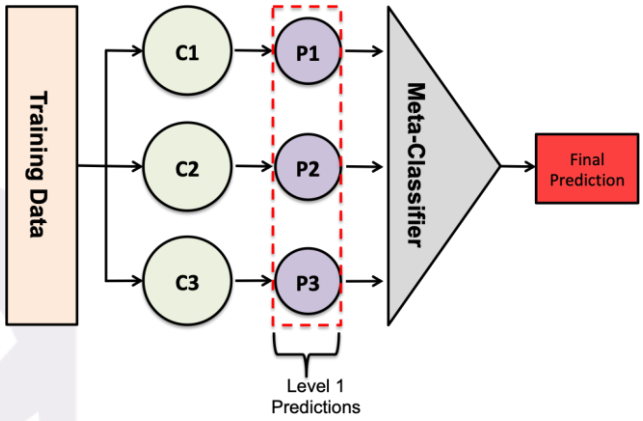

Figura 3. Ejemplo de funcionamiento de stacking.[3]

5. Otros tipos de combinación

En esta sección, se estudiarán distintos tipos de combinación de las salidas de los clasificadores base, y otros métodos de generar clasificadores basados en ensemble.

5.1. Voto mayoritario

El método de voto simple o voto mayoritario es el más extendido en la literatura para generar la predicción final de los clasificadores en el ensemble. De hecho, es el utilizado por lo general en los métodos de Bagging, y en el método Random Forest visto anteriormente, entre otros muchos.

El voto simple es un método sencillo donde, dadas n predicciones de los n clasificadores del ensemble para un patrón dado, la clasificación final será la clase mayoritaria de entre las n predicciones.

Supongamos el ejemplo de la Tabla 1, donde el ensemble tiene 10 clasificadores base. Como vemos, 6 de ellos predicen la clase A, 3 de ellos la clase B, y 1 la clase C. Por tanto, el ensemble predice que el patrón sería de la clase A, que es la clase mayoritaria entre las predicciones.

Tabla 1. Ejemplo de voto mayoritario.

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	Final
A	A	B	A	C	B	B	A	A	A	A

[3] Fuente: https://towardsdatascience.com/stacking-classifiers-for-higher-predictive-performance-566f963e4840

5.2. Combinación de probabilidades

Si los clasificadores proporcionan probabilidades en lugar de únicamente la asignación a una clase, el ensemble podría combinar dichas predicciones mediante cualquier método, como por ejemplo, la media.

En la Tabla 2 ilustramos un ejemplo de combinación de probabilidades para producir la salida final del ensemble. Supongamos las salidas de los 10 clasificadores, donde se muestra la probabilidad de pertenencia a la clase positiva; si consideramos un umbral de 0.5, podemos analizar que de los 10 clasificadores, 6 predecirían la clase positiva (p >= 0.5), y 4 de ellos la clase negativa (p <= 0.4). Sin embargo, en este caso no estamos combinando las clases categóricas sino su probabilidad predicha de pertenencia a la clase. De este modo, tendrían más peso en la predicción final aquellos modelos que predigan una clase con "más convicción" (es decir, probabilidad más cercana a 0 o a 1; cuanto más cercana a 0.5, más indeciso está el clasificador). Al combinar estas salidas, por ejemplo con la media, obtenemos un valor de probabilidad de pertenencia a la clase positiva de 0.39, que considerando el umbral de 0.5 significaría que lo clasifica en la clase negativa.

Tabla 2. Ejemplo de combinación de probabilidades.

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	Final
0.60	0.05	0.51	0.10	0.15	0.02	0.55	0.55	0.80	0.60	0.39

5.3. Combinación con pesos

Hasta el momento, los dos tipos de combinación de salidas vistos anteriormente le dan el mismo peso en la predicción final a todos los clasificadores por igual, independientemente de su rendimiento. Sin embargo, es posible también dar más peso o prioridad en la clasificación final a aquellos modelos que sabemos que funcionan mejor. Como ya se introdujo en secciones anteriores, Boosting utiliza una combinación con pesos, donde el peso de cada clasificador viene dado por alguna métrica de rendimiento del mismo, dando más peso en la predicción final a aquellos con un mejor rendimiento. Sin embargo, los pesos se podrían obtener de otras formas y no solo en base al rendimiento de los modelos.

Supongamos un ejemplo más sencillo como el que se muestra en la Tabla 3, con 7 clasificadores binarios. Si utilizásemos un voto mayoritario, donde todos los clasificadores tienen el mismo peso en la clasificación final, la predicción para ese patrón sería la clase A (ya que 4 proporcionan la salida A y solo 3 la salida B). Sin embargo, como cada clasificador tiene asociado un peso, la salida predicha por cada uno de ellos se pondera por dicho peso. Así, la probabilidad de que el ensemble

proporcione la salida A será $P(A) = (0.5 + 0.55 + 0.60 + 0.55)/5.0 = 0.44$, mientras que $P(B) = (0.95 + 0.95 + 0.9)/5.0 = 0.56$. Nótese que el valor del denominador la suma de todos los pesos. De este modo, considerando un umbral de 0.5, el ensemble predeciría la clase B (pese a que no es la opción mayoritaria entre las predicciones).

Tabla 3. Ejemplo de combinación con pesos.

Clasificador	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**Final**
Peso	0.5	0.55	0.95	0.95	0.60	0.55	0.90	
Predicción	A	A	B	B	A	A	B	B

La combinación con pesos no tiene por qué ser solo de las clases categóricas, sino que se podrían combinar también las probabilidades predichas de pertenencia a cada clase. En ese caso, habría que multiplicar cada peso por su probabilidad asociada, obteniendo la salida final del ensemble.

5.4. Selección dinámica

Por último, simplemente introduciremos, sin entrar en mucho detalle, los métodos de ensemble basados en selección dinámica de clasificadores. Hasta ahora, para un patrón dado, la salida del ensemble venía dada por una combinación de las salidas de todos los clasificadores base del mismo. Sin embargo, pueden existir escenarios donde es más interesante que un patrón se clasifique únicamente por un subconjunto de los clasificadores base del ensemble, que mejor se ajusten a las características de ese nuevo patrón.

Uno de los enfoques más populares de selección dinámica es, en primer lugar, realizar un agrupamiento o división de los datos de entrenamiento en distintos grupos disjuntos. A partir de estos grupos de patrones, se entrenaría un clasificador distinto para cada conjunto; así, cada clasificador podría ser capaz de distinguir más fácilmente características específicas de dichos nichos o subconjuntos, tratando de clasificar mejor únicamente los patrones de ese grupo.

Cuando llega un nuevo patrón al ensemble, el primer paso sería intentar ajustarlo en uno (o varios) de los grupos de patrones iniciales, en base a la distancia o similitud de los patrones de cada uno de los grupos. Una vez que se asocia ese nuevo patrón a uno de los grupos, la salida del ensemble será, por lo general, la salida del clasificador entrenado para ese grupo específico. También podría, en lugar de escogerse un grupo, escoger varios, y utilizar sus clasificadores asociados para producir salidas y combinarlas en una salida final.

Referencias

[Agg15] Aggarwal, C. C. (2015). Data Classification. Algorithms and Applications. *Chapman and Hall/CRC*.

[Mai10] Maimon, O., & Rokach, L. (Eds.). (2010). Data mining and knowledge discovery handbook, 2nd edition. *Springer*.

[Wit11] Witten, I. H., Frank, E., & Hall, M. A. (2011). Data mining: practical machine learning tools and techniques, 3rd edition. *Morgan Kaufmann*.