Ensamble Secuencial de Modelos de Aprendizaje Automático Supervisado

Introducción:

En esta sección se dedican varios párrafos para dar el contexto en el que se desarrolla el trabajo presentado. Normalmente se va desde lo más general a lo más preciso, para englobar el trabajo dentro de un ámbito concreto. Hay que recordar que se debe referenciar la información que se consiga de fuentes bibliográficas (artículos, libros, capítulos de libros, páginas web, apuntes, etc.). Para ello se añade el ítem correspondiente en el apartado de referencias, dotándolo de un número, y se añade la referencia al final de la frase o párrafo correspondiente [1]. Si se realiza en Microsoft Word o LibreOffice, se puede hacer uso de *referencias cruzadas* para actualizar la numeración de las referencias de forma automática en todo el documento.

Después vienen un par de párrafos para dar un poco más de detalle sobre el trabajo realizado. Hay que indicar la problemática o el objetivo que se marca, y cómo se ha enfocado la solución propuesta en este trabajo. Finalmente, el último párrafo se dedica a comentar la estructura del documento por secciones, como el que sigue.

En este trabajo se da una posible solución al problema de crear un ensamble de modelos de aprendizaje automático supervisado. Para ello, algunos conceptos básicos que convendría repasar son:

* Aprendizaje supervisado: consiste en algoritmos que necesitan la variable objetivo para entrenar y predecir sus valores, con lo que podemos medirlos y ver cómo de buenos son. Por tanto, son contrarios al aprendizaje por refuerzo, en el que no se sabe la solución antes de empezar.
* Durante el curso se ha trabajado con Naive Bayes, KNN, árboles de decisión y redes neuronales. Todos ellos han sido probados en este trabajo, además de algunos otros, con intención de entender porqué o no son buenos para ser ensamblados.

Para ello, hemos utilizado una técnica similar a GradientBoosting (potenciación del gradiente) descrita en el pseudocódigo del documento introductorio, evaluando los modelos con evaluación cruzada, como se pedía en la propuesta.

Estructura y Decisiones de Diseño:

El notebook comienza con todos los imports, seguido de celdas de documentación. La celda 5 tiene la implementación del meta-modelo, todo lo que necesita para funcionar.

**Decisiones de diseño:**

* El formato de los datos es csv y se almacena (en memoria) en un objeto de tipo DataFrame de Pandas.
* El meta-modelo puede aceptar tanto tareas de regresión como de clasificación.
* El meta-modelo acepta o no normalización.
* Se inicializa la predicción inicial como la media de la variable objetivo, ya que eso garantiza un mejor resultado en menos iteraciones que inicializarla a 0 en el mayor de los casos.
* Se usa validación cruzada para la evaluación del ensamble.
* Cada modelo entrena con datos aleatorios.
* La métrica usada tanto para la parada temprana como para la validación cruzada es el coeficiente de determinación, *R*2.
* Para la parada temprana (early stopping) hemos definido un valor umbral para considerar si la mejora de la predicción del modelo actual respecto del anterior es útil, en vez de comprobar que no haya mejora, ya que, si no, con los pocos datos que tenemos y las iteraciones que este trabajo exige no se aplicaría nunca. De hecho, tenemos que trabajar con valores bastante elevados del umbral y con una paciencia baja para que pare. Esta evaluación se realiza con los datos de prueba no utilizados en el entrenamiento de cada modelo.
* El ensamble soporta regresores de redes neuronales.
* El ensamble soporta tareas de regresión y de clasificación
* El ensamble hereda de BaseEstimator y RegressorMixin para poder utilizar diversos métodos de scikit learn que se salen del alcance de este trabajo.

Experimentación:

Para la experimentación se han usado los csv originales para las tareas de regresión y los mismos pero cambiando el valor de la variable objetivo a binaria. Se han utilizado los regresores DecisionTreeRegressor (DTR), el obligatorio, para el que se han obtenido los siguientes valores experimentando con su hipérparámetro max\_depth y los hiperparámetros n\_estimators, learning rate y sample\_size, tal como se exige en el trabajo. Además, y con objeto de reutilizar esta tabla, el otro regresor seleccionado es HistGradientBoostingRegressor (HGBR), que es un ensamble parecido al que estamos realizando (con GradientBoosting) solo que es exclusivo para árboles, pero estrictamente es un regresor diferente, por eso es analizado en este trabajo. Estos resultados se han obtenido con el csv de parkinsons, ya que los resultados son mejores que los obtenidos con house\_prices. Además, para todos ellos se han utilizado 10 pliegues en la validación cruzada, así como un épsilon de 0.0002 y una paciencia de 6:

SE OBSERVA QUE LOS MEJORES PARÁMETROS PARA

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| n\_estimators | Learning rate | Sample size | max depth | R2 DTR | R2 HGBR |
| 1 | 0.01 | 0.75 | 5 | 0.004100127685379129 (0.1 s) |  |
| 50 | 0.01 | 0.75 | 5 | 0.3903660490816846 (1.5 s) |  |
| 100 | 0.01 | 0.75 | 5 | 0.599086564902081 (3.3 s) |  |
| 150 | 0.01 | 0.75 | 5 | 0.7088736826203287 (5.5 s) |  |
| 200 | 0.01 | 0.75 | 5 | 0.7716728458728124 (8.9 s) |  |
| 250 | 0.01 | 0.75 | 5 | 0.8058932571320918 (11.8 s) |  |
| 300 | 0.01 | 0.75 | 5 | 0.8278481114782436 (23.4 s) |  |
| 1 | 0.05 | 0.75 | 5 | 0.04741486275217653 (0.0 s) | 0.06967745237753957 (1.5 s) |
| 50 | 0.05 | 0.75 | 5 | 0.8073874656748916 (1.1 s) | 0.9128288415849394 (20.8 s) |
| 100 | 0.05 | 0.75 | 5 | 0.8569389911796504 (5.2 s) |  |
| 150 | 0.05 | 0.75 | 5 | 0.8725326144810763 (6.2 s) |  |
| 200 | 0.05 | 0.75 | 5 | 0.8718035153459839 (8.9 s) |  |
| 250 | 0.05 | 0.75 | 5 | 0.8747888736233855 (12.2 s) |  |
| 300 | 0.05 | 0.75 | 5 | 0.8765984376955676 (22.5 s) |  |
| 1 | 0.1 | 0.75 | 5 |  |  |
| 50 | 0.1 | 0.75 | 5 |  |  |
| 100 | 0.1 | 0.75 | 5 |  |  |
| 150 | 0.1 | 0.75 | 5 |  |  |
| 200 | 0.1 | 0.75 | 5 |  |  |
| 250 | 0.1 | 0.75 | 5 |  |  |
| 300 | 0.1 | 0.75 | 5 |  |  |
| 1 | 0.25 | 0.75 | 5 |  |  |
| 50 | 0.25 | 0.75 | 5 |  |  |
| 100 | 0.25 | 0.75 | 5 |  |  |
| 150 | 0.25 | 0.75 | 5 |  |  |
| 200 | 0.25 | 0.75 | 5 |  |  |
| 250 | 0.25 | 0.75 | 5 |  |  |
| 300 | 0.25 | 0.75 | 5 |  |  |
| 1 | 0.01 | 0.9 | 5 |  |  |
| 50 | 0.01 | 0.9 | 5 |  |  |
| 100 | 0.01 | 0.9 | 5 |  |  |
| 150 | 0.01 | 0.9 | 5 |  |  |
| 200 | 0.01 | 0.9 | 5 |  |  |
| 250 | 0.01 | 0.9 | 5 |  |  |
| 300 | 0.01 | 0.9 | 5 |  |  |
| 1 | 0.05 | 0.9 | 5 |  |  |
| 50 | 0.05 | 0.9 | 5 |  |  |
| 100 | 0.05 | 0.9 | 5 |  |  |
| 150 | 0.05 | 0.9 | 5 |  |  |
| 200 | 0.05 | 0.9 | 5 |  |  |
| 250 | 0.05 | 0.9 | 5 |  |  |
| 300 | 0.05 | 0.9 | 5 |  |  |
| 1 | 0.1 | 0.9 | 5 |  |  |
| 50 | 0.1 | 0.9 | 5 |  |  |
| 100 | 0.1 | 0.9 | 5 |  |  |
| 150 | 0.1 | 0.9 | 5 |  |  |
| 200 | 0.1 | 0.9 | 5 |  |  |
| 250 | 0.1 | 0.9 | 5 |  |  |
| 300 | 0.1 | 0.9 | 5 |  |  |
| 1 | 0.25 | 0.9 | 5 |  |  |
| 50 | 0.25 | 0.9 | 5 |  |  |
| 100 | 0.25 | 0.9 | 5 |  |  |
| 150 | 0.25 | 0.9 | 5 |  |  |
| 200 | 0.25 | 0.9 | 5 |  |  |
| 250 | 0.25 | 0.9 | 5 |  |  |
| 300 | 0.25 | 0.9 | 5 |  |  |
| 1 | 0.01 | 0.75 | 3 |  |  |
| 50 | 0.01 | 0.75 | 3 |  |  |
| 100 | 0.01 | 0.75 | 3 |  |  |
| 150 | 0.01 | 0.75 | 3 |  |  |
| 200 | 0.01 | 0.75 | 3 |  |  |
| 250 | 0.01 | 0.75 | 3 |  |  |
| 300 | 0.01 | 0.75 | 3 |  |  |
| 1 | 0.01 | 0.75 | 10 |  |  |
| 50 | 0.01 | 0.75 | 10 |  |  |
| 100 | 0.01 | 0.75 | 10 |  |  |
| 150 | 0.01 | 0.75 | 10 |  |  |
| 200 | 0.01 | 0.75 | 10 |  |  |
| 250 | 0.01 | 0.75 | 10 |  |  |
| 300 | 0.01 | 0.75 | 10 |  |  |
| 1 | 0.05 | 0.75 | 3 |  |  |
| 50 | 0.05 | 0.75 | 3 |  |  |
| 100 | 0.05 | 0.75 | 3 |  |  |
| 150 | 0.05 | 0.75 | 3 |  |  |
| 200 | 0.05 | 0.75 | 3 |  |  |
| 250 | 0.05 | 0.75 | 3 |  |  |
| 300 | 0.05 | 0.75 | 3 |  |  |
| 1 | 0.05 | 0.75 | 10 |  |  |
| 50 | 0.05 | 0.75 | 10 |  |  |
| 100 | 0.05 | 0.75 | 10 |  |  |
| 150 | 0.05 | 0.75 | 10 |  |  |
| 200 | 0.05 | 0.75 | 10 |  |  |
| 250 | 0.05 | 0.75 | 10 |  |  |
| 300 | 0.05 | 0.75 | 10 |  |  |
| 1 | 0.1 | 0.75 | 3 |  |  |
| 50 | 0.1 | 0.75 | 3 |  |  |
| 100 | 0.1 | 0.75 | 3 |  |  |
| 150 | 0.1 | 0.75 | 3 |  |  |
| 200 | 0.1 | 0.75 | 3 |  |  |
| 250 | 0.1 | 0.75 | 3 |  |  |
| 300 | 0.1 | 0.75 | 3 |  |  |
| 1 | 0.1 | 0.75 | 10 |  |  |
| 50 | 0.1 | 0.75 | 10 |  |  |
| 100 | 0.1 | 0.75 | 10 |  |  |
| 150 | 0.1 | 0.75 | 10 |  |  |
| 200 | 0.1 | 0.75 | 10 |  |  |
| 250 | 0.1 | 0.75 | 10 |  |  |
| 300 | 0.1 | 0.75 | 10 |  |  |
| 1 | 0.25 | 0.75 | 3 |  |  |
| 50 | 0.25 | 0.75 | 3 |  |  |
| 100 | 0.25 | 0.75 | 3 |  |  |
| 150 | 0.25 | 0.75 | 3 |  |  |
| 200 | 0.25 | 0.75 | 3 |  |  |
| 250 | 0.25 | 0.75 | 3 |  |  |
| 300 | 0.25 | 0.75 | 3 |  |  |
| 1 | 0.25 | 0.75 | 10 |  |  |
| 50 | 0.25 | 0.75 | 10 |  |  |
| 100 | 0.25 | 0.75 | 10 |  |  |
| 150 | 0.25 | 0.75 | 10 |  |  |
| 200 | 0.25 | 0.75 | 10 |  |  |
| 250 | 0.25 | 0.75 | 10 |  |  |
| 300 | 0.25 | 0.75 | 10 |  |  |

Se obvian las pruebas con sample size = 0.9 para diferentes max Depth porque se considera que estas son más que suficientes para una amplia cobertura del ensamble de modelos.

Conclusiones:

Podemos observar que, tras unas pruebas exhaustivas, el modelo, utilizando DecissionTreeRegressor alcanza un R2 máximo de X con parámetros. Sin embargo, vemos que a partir de x iteraciones no se alcanza una mejora en el rendimiento

Autodeclaración del uso de IA generativa:

En este proyecto la IA se ha usado exclusivamente para corrección de errores en tiempo de ejecución y para la ayuda puntual de alguna interpretación o la mejora del código existente (más legible y/o optimizado)

Bibliografía:

[ChatGPT](https://chatgpt.com/)

[RegressorMixin — scikit-learn 1.6.1 documentation](https://scikit-learn.org/stable/modules/generated/sklearn.base.RegressorMixin.html)

[cross\_val\_score — scikit-learn 1.6.1 documentation](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.cross_val_score.html)