*Modelos de Ensamble y Boosting Models*

***Conceptos básicos***

Los métodos de “ensamble de modelos” o “métodos combinados” intentan ayudar a mejorar el rendimiento de los modelos de Machine Learning al mejorar su precisión. Este es un proceso mediante el cual se construyen estratégicamente varios modelos de Machine Learning para resolver un problema particular.

Veamos un ejemplo:

Supongamos que queremos invertir en una empresa, pero no estamos seguro de su posible rendimiento, por lo tanto buscamos varios expertos financieros para que nos aconsejen si el precio de la acción aumentará en más de un 5% anual.

Los resultados fueron los siguientes:

Texto

Descripción generada automáticamente con confianza baja

Evaluemos los resultados obtenidos:

* **Empleado de la empresa:**Esta persona conoce la funcionalidad interna de la empresa y tiene información sobre la funcionalidad de la misma, pero carece de una perspectiva más amplia sobre cómo están innovando los competidores y cómo está evolucionando la tecnología. En el pasado, ha tenido razón 70% de veces.
* **Asesor financiero de la empresa:** Esta persona tiene una perspectiva más amplia sobre cómo la estrategia de las empresas será ajustada en este entorno competitivo, sin embargo, carece de una visión sobre cómo las políticas internas de la empresa. En el paso ha tenido razón un 75% de las veces.
* **Operador del mercado de valores:** Esta persona ha observado el precio de las acciones de la empresa en los últimos años y conoce las tendencias de estacionalidad y el rendimiento del mercado en general, pero también conoce que las acciones pueden variar con el tiempo. En el pasado ha tenido razón 65% de veces.
* **Empleado de un competidor:** Esta persona conoce la funcionalidad interna de las firmas competidoras y está consciente de ciertos cambios que aún no se han implementado, pero por otra parte carece de conocimiento sobre la empresa enfocada y de los factores externos que pueden relacionar el crecimiento del competidor. En el pasado, ha tenido razón el 60% de las veces.

Como podemos observar del ejemplo anterior, cada experto tiene su propia opinión respecto del tema y nos ofrece una respuesta a la pregunta solicitada. La idea justamente es encontrar la decisión más adecuada en base a las diversas miradas de los expertos.

Ahora bien, ¿Cuál es la relación entre el ejemplo propuesto y los modelos de ensamble? El ensamblado es el arte de combinar un conjunto de diversos de modelos para improvisar sobre la estabilidad y el poder predictivo del modelo es decir, las “diferentes opiniones” de los expertos en nuestro ejemplo.

Obviamente estos modelos, pueden ser diferentes entre sí por una variedad de razones:

* Puede haber diferencia en la población de datos.
* Puede haber una técnica de modelado diferente utilizada.
* Puede haber una hipótesis diferente.

Volviendo al ejemplo anterior, serían las diferentes opiniones de nuestros expertos financieros.

En consecuencia, ¿En qué consisten los algoritmos ensamblados? De manera sencilla podría decirse que son algoritmos formados por algoritmos más simples. Estos algoritmos simples se unen para formar un algoritmo más potente, siguiendo la premisa: “La unión hace la fuerza”.

Aunque hay diversas formas de ensamblar o unir algoritmos débiles para formar otros, las más usadas y populares son el bagging y el boosting.

Cada tipo de algoritmo tiene unas ventajas y unos inconvenientes y pueden ser usados convenientemente según sea nuestra problemática a resolver.

***Algoritmos de bagging***

Los métodos de bagging son métodos donde los algoritmos simples son usados en paralelo. El principal objetivo de los métodos en paralelo es el de aprovecharse de la independencia que hay entre los algoritmos simples, ya que el error se puede reducir bastante al promediar las salidas de los modelos simples.

Podríamos pensarlo como como si quisiéramos resolver un problema entre varias personas independientes unas de otras, y finalmente damos por bueno lo que eligiese la mayoría de las personas.

**Importante:** El principal objetivo intrínseco de los algoritmos de bagging es el de la reducción de la varianza.

Diagrama

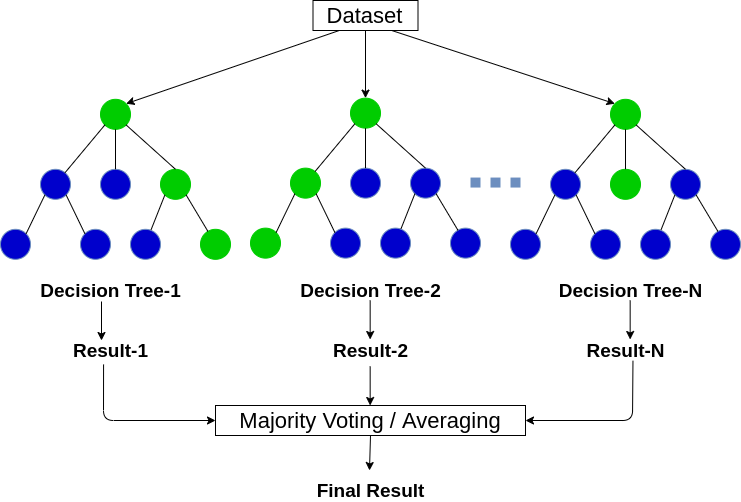
Descripción generada automáticamente

Bagging proviene del concepto de agregación de bootstrap. Un forma de reducir la varianza de las estimaciones es promediando estimaciones de distintos modelos o algoritmos.

Para obtener la agregación de las salidas de cada modelo simple e independiente, bagging puede usar la votación para los métodos de clasificación y el promedio para los métodos de regresión.

**Entonces …. ¿Cuál es el algoritmo más popular de Bagging?**

Random Forest!



***Algoritmos de bosting***

En los algoritmos de boosting, los modelos simples son utilizados secuencialmente, es decir, cada modelo simple va delante o detrás de otro modelo simple.

El principal objetivo de los métodos secuenciales es el de aprovecharse de la dependencia entre los modelos simples.

El rendimiento general puede ser mejorado haciendo que un modelo simple posterior, le de más importancia a los errores cometidos por un modelo simple previo. Sería como si nosotros al resolver un problema aprovechásemos nuestro conocimiento de los errores de otros para mejorar en algo intentando no cometerlos nosotros.

Diagrama

Descripción generada automáticamente

Las predicciones de cada modelo simple se combinan por medio de una votación (para problemas de clasificación) o por medio de una suma ponderada (para problemas de regresión) para producir la predicción final.

La diferencia con el bagging es que en el boosting los algoritmos no se entrenan independientemente, sino que se ponderan según los errores de los anteriores.

**Adaboost:**

Este algoritmo entrena de forma secuencial un conjunto de aprendices débiles a partir de un algoritmo base común. Todos los aprendices son entrenados con el mismo conjunto de datos pero éstos van recibiendo pesos que dependen de los errores cometidos por cada aprendiz.

Así, inicialmente todos las muestras reciben un peso inicial wi de 1/n (suponiendo que haya n muestras). El primer aprendiz es entrenado y se estima su tasa de error.

Suponiendo que estemos trabajando en un problema de clasificación, esta tasa de error (en general, para el aprendiz j-ésimo) sería:

Imagen que contiene Texto

Descripción generada automáticamente

Si no ha habido muestras mal clasificadas, esta tasa de error será 0. Por el contrario, si todas las muestras han sido mal clasificadas, esta tasa será 1.

**Gradient Boosting:**

El aumento de gradiente es un tipo de algoritmo de aumento. Se basa en la intuición de que el mejor modelo siguiente posible, cuando se combina con modelos anteriores, minimiza el error de predicción general. La idea clave es establecer los resultados objetivo para este próximo modelo con el fin de minimizar el error. El aumento de gradiente se puede utilizar tanto para clasificación como para regresión.

Diagrama, Forma

Descripción generada automáticamente

**XGBoost:**

Extreme Gradient Boosting, es uno de los algoritmos de machine learning de tipo supervisado más usados en la actualidad. Este algoritmo se caracteriza por obtener buenos resultados de predicción con relativamente poco esfuerzo, en muchos casos equiparables o mejores que los devueltos por modelos más complejos computacionalmente, en particular para problemas con datos heterogéneos.

Se basa en el principio de “boosting”, que como bien sabemos la idea principal consiste en generar múltiples modelos de predicción “débiles” secuencialmente y que cada uno de estos tome los resultados del modelo anterior, para generar un modelo más “fuerte”, con mejor poder predictivo y mayor estabilidad en sus resultados.

Para conseguir un modelo más fuerte, se emplea un algoritmo de optimización, este caso Gradient Descent (descenso de gradiente). Durante el entrenamiento, los parámetros de cada modelo débil son ajustados iterativamente tratando de encontrar el mínimo de una función objetivo, que puede ser la proporción de error en la clasificación, el área bajo la curva (AUC), la raíz del error cuadrático medio (RMSE), etc.

Cada modelo es comparado con el anterior. Si un nuevo modelo tiene mejores resultados, entonces se toma este como base para realizar nuevas modificaciones. Si, por el contrario, tiene peores resultados, se regresa al mejor modelo anterior y se modifica ese de una manera diferente.

Este proceso se repite hasta llegar a un punto en el que la diferencia entre modelos consecutivos es insignificante, lo cual nos indica que hemos encontrado el mejor modelo posible, o cuando se llega al número de iteraciones máximas definido por el usuario.

**LightGBM:**

Utiliza la técnica Gradient Boosting. Con este método los árboles se construyen de manera secuencial y cada uno que se agrega aporta su granito de arena para refinar la predicción anterior. Es decir, se comienza con un valor constante y cada árbol nuevo se entrena para predecir el error en la suma de todas las predicciones de los árboles anteriores.

Una vez terminado el proceso, las predicciones se calculan sumando los resultados de todos los árboles que se construyeron. El efecto que tiene esto es que cada vez que se agrega un árbol nuevo se le presta atención a las muestras en las que el modelo está funcionando peor y se trabaja para mejorar ese aspecto.