MODELO PREDICTIVO

El objetivo de este apartado es tomar el output de todas las transformaciones realizadas y explicadas anteriormente y obtener un modelo que sea capaz de predecir qué equipo va a ganar un partido. Para ello, se toma como input el csv input\_extendido que es el resultado de todas las transformaciones realizadas y explicadas anteriormente.

El desarrollo del modelo se divide en cinco partes:

* **Parte I:** Exploración, limpieza, construcción/transformación de variables
* **Parte II:** Selección de variables.
* **Parte III:** Selección de Modelos.
* **Parte IV:** Construcción de modelos.
* **Parte V:** Evaluación del modelo.

**Parte I**

**Exploración, limpieza, construcción/transformación de variables**

En esta sección se realiza la preparación de los datos, limpieza del dataframe por valores nulos, renombrado de variables, cálculo de nuevas variables, descarte de variables que no forman parte de análisis, exploración gráfica y otros procesos. Haciendo uso de las librerías de *Pandas* y *Numpy* de *Python.*

En primer lugar, se importa el csv input extendido en un dataframe. Este dataframe inicial contempla 6220 filas y 180 columnas (variables). Después de un proceso de limpieza por valores nulos derivados del proceso de extracción y de cálculos de divisiones de denominadores iguales a 0, se ha obtenido finalmente 5809 (son menos hay que calcularlo si se considera necesario) filas para trabajar.

Seguidamente se pasa al **cálculo de variables compuestas** donde destaca, entre alguno de los cálculos, la creación de la variable ***“target”*** (objeto de estudio de valores binarios) a partir de la diferencia de los puntos marcados por el equipo local y el equipo visitante, de tal forma que, valores igual a 1 para indicar cuando el equipo local gana y 0 para el equipo visitante cuando pierde.

Entre las 180 variables que se han importado del .csv resaltan 2 variables categóricas que contienen la información de la división y la conferencia a la que pertenece el equipo local y el equipo visitante. Estas variables categóricas deben ser transformadas en variables dicotómicas (*dummy´s).* Esta transformación se ha realizado mediante el enfoque de *One Hot encoding* con funciones de la librería de *Pandas.* Dado que existen dos conferencias y seis divisiones en la NBA, han resultado 14 variables dicotómicas que sustituyen en cuanto a información a las dos variables originales.

Una vez identificadas, calculadas, transformadas, y organizadas todas las variables en diversas listas (variables continuas y dicotómicas) se decide filtrar las variables que *a priori* no aportan ningún valor al modelo, muchas derivan del proceso de extracción de datos y otras que, si bien tienen mucha información asociada, se han descartado por otras variables que en información ya las sustituye.

Básicamente un proceso de *“Garbage in, garbage out”* que nos permite trabajar específicamente con las variables de interés para el modelo.

Este proceso de selección se divide entre variables previas al partido y variables del partido se la siguiente forma:

**(se puede hacer una especie de cuadro o algo más visual)**

**Variables previas al partido:**

**REVISAR NOTEBOOK, YA TODO ESTO ESTA ESCRITO**

**Variables durante el partido:**

**REVISAR NOTEBOOK, YA TODO ESTO ESTA ESCRITO**

**Parte II**

**Selección de variables**

En esta sección se realiza la selección definitiva de variables para el modelo desde distintos enfoques que dependen del tipo de variable. Inicialmente se han separado las variables en diferentes listas; continuas y dicotómicas, así como, la separación del conjunto de entrenamiento y test para la evaluación del modelo final.

Este proceso de separación de los datos para el proceso de selección de variables se realiza para evitar extraer información del total conjunto de datos, ya que, posteriormente una parte de estos datos se deben usar para evaluar el modelo. Comúnmente en aprendizaje automático a este fenómeno se le conoce como “fuga de datos” (*data leakege*).

Haciendo uso de la librería de ***sklearn*** de Python se ha desarrollado todo el proceso de aprendizaje automático.

Es importante resaltar que, durante la fase de análisis exploratorio (Parte I) se realizaron matrices de correlación sobre las variables continuas previas y durante el partido (por separado) sobre *“target”* para evaluar el posible grado de relación lineal que tienen las variables entre sí y la variable objetivo.

En este análisis de correlación no se han descartado variables porque en este estudio no se pretende realizar un análisis de causalidad entre variables explicativas y la variable objetivo. Este proceso de selección de variables tiene sentido en esos casos porque se pretende cuantificar la atribución de una variable respecto a otra y de esta forma el nivel de importancia.

En el caso de modelajes predictivos es un error excluir variables por presencia de altos niveles de correlación entre variables explicativas (excepto perfecta colinealidad entre independientes) porque se contribuye al deterioro de la precisión del modelo, esto se debe porque las variables en conjunto se complementan y comprenden más información que cada una de estas por separado.

Del mismo modo, el descarte de variables por escasez de correlación es debatible, depende en muchos casos el modelo a utilizar y de la relación establecida a priori entre la variable y la objetivo. El coeficiente de correlación de Pearson, por ejemplo, es una medida de la relación "lineal" entre dos variables, en este contexto, se puede decir que no necesariamente las variables tienen que estar relacionadas linealmente.

Asimismo, en el apartado anterior (Parte I) se han descartado variables que pretenden aportar la misma información en el modelo. En cualquier caso, de requerirse para modelos futuros, se ha dejado en el notebook **funciones\_modelo.ipynb** la función construida para Factor Inflación de la Varianza (VIF) que permite de forma automática, descartar variables explicativas dependiendo del nivel de correlación establecido.

A continuación, separamos el proceso de selección de variables como se ha comentado en el inicio de este apartado:

1. **Selección de Variables Dicotómicas:**

Después del proceso de transformación de variables categóricas comentado en la primera sección y sumado a las variables dicotómicas generadas en previos procesos del estudio, se han obtenido en total 26 variables dummy´s que se reparten en partes iguales entre el equipo local y visitante.

|  |  |
| --- | --- |
| played\_previous\_date | Conf\_Este |
| played\_two\_days\_ago | Div\_Atlantic |
| played\_three\_days\_ago | Div\_Central |
| played\_prorrogue\_previous\_date | Div\_Northwest |
| played\_prorrogue\_two\_days\_ago | Div\_Pacific |
| played\_prorrogue\_three\_days\_ago | Div\_Southeast |
|  | Div\_Southwest |

Estas variables se han calculado para enriquecer la complejidad del modelo, sin embargo, como es posible sospechar, muchas de estas pueden que no sean relevantes para predecir el resultado de un juego, y su inclusión en el modelo solo contribuyen a la generación de “ruido” (perturbaciones sobre las predicciones).

Por lo tanto, es necesario conocer cuáles de estas variables son las que más información aportan o afectan el comportamiento de la variable objetivo. Entre las distintas técnicas y conceptos estadísticos que existen se ha decidido utilizar la **prueba de independencia de chi-cuadrado** y la medida de **información mutua.**

La **prueba de independencia de ji-cuadrado** (chi-cuadrado) para variables categóricas contrasta la hipótesis nula de que las variables dentro de la prueba son independientes a través de la distribución que presenta una frente a la otra mediante tablas de contingencia.

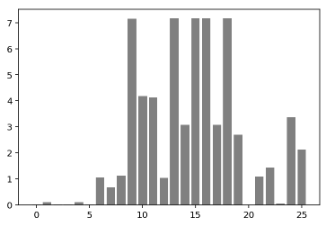
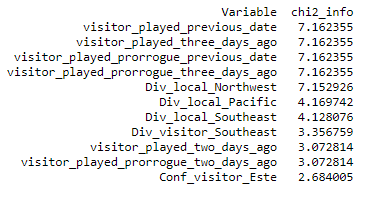
Mientras que, la **información mutua** (*mutual information*) es una cantidad que mide la relación entre dos variables aleatorias que se muestrean simultáneamente. En particular, mide cuánta información se comunica, en promedio, en una variable aleatoria sobre otra.

Un teorema importante de la teoría de la información dice que la información mutua entre dos variables es 0 si y solo si las dos variables son estadísticamente independientes.

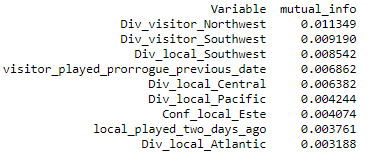
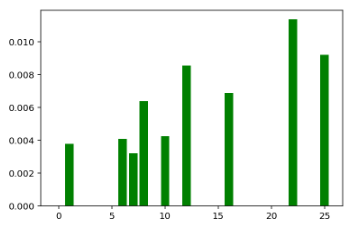
En efecto, el mismo resultado que evaluamos cuando se realiza la hipótesis nula de una prueba de chi-cuadrado y se espera rechazar cuando se realiza selección de variables categóricas, por ejemplo.

Más allá del criterio empírico del proceso estadístico, luego de evaluar los resultados de ambos enfoques, se decide utilizar las variables seleccionadas por el cálculo de la información mutua, a priori, parece que las variables seleccionadas por este método tienen mayor importancia en cuanto al criterio del deporte del baloncesto respecta.

Las variables dummy´s seleccionadas por chi-cuadrado son las siguientes:



Las variables *dummy´s* seleccionadas por *información mutua* son las siguientes:

1. **Selección de Variables Continuas:**

Derivado del proceso de filtrado de *“Garbage in, garbage out”* en el apartado de análisis exploratorio (Parte I) se obtuvieron 53 variables continuas. Al igual que el proceso de selección anterior, se utilizaron diversos métodos para descartar las variables no relevantes para predecir el resultado de un juego, entre las que destacan: *Genetic Algorithms, Permutation Importance y Random Forest Feature Importance.*

En entornos de aprendizaje automático podemos citar a Joaquín Amat Rodrigo para definir que los algoritmos genéticos son “métodos de optimización heurística que, entre otras aplicaciones, pueden emplearse para encontrar la combinación de variables que consigue maximizar la capacidad predictiva de un modelo”.

El primer intento de selección de variables se basó en este método, utilizando básicamente adaptaciones de códigos suministrados en una asignatura durante el Máster y aplicaciones técnicas encontradas en la web. La maximización de la capacidad predictiva del modelo se fundamentó en la medida del índice de impureza de Gini.

Según O’Relly Machine Learning Book etc etc **(cita).**“Dada una muestra, la impureza de Gini mide la probabilidad de una clasificación errónea si una etiqueta se elige aleatoriamente usando la distribución de probabilidad de la rama. El índice alcanza su mínimo (0.0) cuando todas las muestras de un nodo se clasifican en una sola categoría”.

El segundo proceso de selección de variables fue el *Random Forest Feature Importance* usando adaptaciones del código propuesto por la librería de *sklearn* en la web. Según la web este algoritmo es un “meta-estimador que ajusta un número de clasificadores de árbol de decisión en varias submuestras del conjunto de datos y utiliza el promedio para mejorar la precisión predictiva y controlar el sobreajuste”.

El criterio de selección de importancia de variables por defecto de la librería de *sklearn* es basado en la medida del índice de impureza de Gini (*Gini impurity index*).

La importancia de las variables basadas en impurezas se fundamenta en el criterio de “cuanto más alto el índice, más importante es la variable”. Y esta importancia se calcula como la “reducción total (normalizada) del índice de cada variable”.

Finalizado el proceso de selección anterior, se decide realizar un último proceso de selección llamado “*Permutation Importance*” derivado de las recomendaciones de la propia librería de *sklearn* por las carencias que tiene el algoritmo anterior sobre el criterio de selección basado en la medida de importancia de Gini; en la web indican que este proceso de selección puede ser “engañoso” para variables de alta cardinalidad (sobrestimación de su importancia), asimismo, demuestran cómo el algoritmo tiende a sobreestimar la importancia de variables que no tienen relación alguna con la variable objetivo.

La “importancia de permutación” de una variable se calcula de la siguiente manera:

Primero, se evalúa una métrica de línea base (baseline metric), definida por una puntuación (scoring) en un conjunto de datos (diferente a la línea base). A continuación, se permuta una columna de una variable del conjunto de validación y se evalúa nuevamente la métrica. La importancia de la permutación de cada variable se define como la diferencia entre la métrica de línea base y la métrica obtenida de permutar la columna de la variable.

Cada uno de los métodos mencionados anteriormente arrojaron resultados diferentes, sin embargo, a priori parece existir un patrón entre el número optimo de variables necesarias para maximizar la métrica con la que se estuviese evaluando cada método. Este hallazgo es importante, y es la razón por el cual se ha decidido contrastar diferentes metodologías para la selección de variables. Si bien cada método le dio más o menos importancia a algún grupo de variables en concreto, todas resultaron arrojar entre 22 y 27 variables esencialmente importantes para maximizar el desempeño del modelo.

Se decidió elegir finalmente las variables por el criterio de “importancia de permutación”; específicamente por mayor conocimiento del cálculo de la métrica de selección (comparado con el complejo proceso del proceso del Algoritmo Genérito) y porque este proceso es recomendado por la librería de *sklearn* por encima del proceso del *Random Forest Feature Importance.*

Las variables continuas más importantes seleccionadas por cada uno de los métodos seleccionados anteriormente son las siguientes:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Nota:**

El proceso de selección de variables forma parte del proceso de optimización de parámetros durante la construcción y evaluación del modelo, en este contexto, no existen garantías de uso exclusivo de las variables seleccionadas en esta sección. La totalidad de estas variables se utilizan como base para la construcción del modelo, luego durante el proceso de optimización de parámetros y evaluación del modelo se determina si la inclusión de variables adicionales mejora o no el desempeño del modelo.

Una vez realizado el diagnostico de las curvas de aprendizaje de los modelos y de las métricas de evaluación, se encontró (derivado de muchas pruebas de ensayo y error) que la inclusión de la totalidad de las variables del conjunto de entrenamiento contribuiría a la mejora de la complejidad del modelo sin incidir en el detrimento de las métricas de evaluación.

Según el costo de oportunidad de “alto sesgo” y “baja varianza” *(high bias and low variance)* analizado en las curvas de aprendizaje, se prefirió incrementar la complejidad de los modelos (mayor varianza) y en efecto, reducir el sesgo.

Cuanto menos sesgado sea un modelo (magnitud del error), mayor será su capacidad para ajustar bien los datos (cuidando el riesgo del *overfitting*). Cuanto mayor es la capacidad del modelo para ajustarse bien a los datos, mayor es la varianza (medida de dispersión entre modelos respecto al modelo poblacional.

Finalmente, con esta decisión hemos optimizado dos modelos muy diferentes. El primero con un alto sesgo (baja varianza), y el segundo con alta varianza (bajo sesgo). Por lo que el primero tiene una mayor capacidad de generalización (replicar el comportamiento aprendido en el conjunto de entrenamiento en el conjunto de validación) y el segundo, un modelo más complejo que se ajusta al patrón observado en el conjunto de entrenamiento.

**Parte II**

**Selección de Modelos**