



Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Estado de México

Inteligencia artificial avanzada para la ciencia de datos I

**Momento de Retroalimentación: Módulo 2 Análisis y Reporte sobre el
desempeño del modelo. (Portafolio Análisis)**

Grupo 101

Gilberto André García Gaytán - A01753176

Profesor:

Jorge Adolfo Ramírez Uresti

Momento de Retroalimentación: Módulo 2 Análisis y Reporte sobre el desempeño del modelo. (Portafolio Análisis)

- Reporte: Análisis del Modelo de Random Forest en el Dataset de Jugadores de Fútbol	3
- Introducción	3
- Justificación de la Selección del Dataset	3
- Separación y Evaluación del Modelo (Train/Test/Validation)	3
- Muestra del Conjunto de Entrenamiento:	4
- Muestra del Conjunto de Validación:	4
- Muestra del Conjunto de Prueba:	5
- Gráfica de dispersión de (Train/Test/Validation)	6
- Beneficios de la Separación de Conjuntos:	6
- Comparación de Métricas de Desempeño para Diferentes Configuraciones de Modelo	6
- Diagnóstico del Grado de Sesgo (Bias) y Varianza	7
- Diagnóstico del Nivel de Ajuste del Modelo	8
- Técnicas de Regularización o Ajuste de Parámetros	8
- 1. Diagnóstico y explicación del grado de sesgo:	8
- 2. Diagnóstico y explicación del grado de varianza:	8
- 3. Diagnóstico y explicación del nivel de ajuste del modelo:	9
- 4. Uso de técnicas de regularización o ajuste de parámetros:	9
- Gráficas, explicación y análisis	10
- Ajuste de parámetros y técnicas de regularización:	13
- Conclusión	14

Reporte: Análisis del Modelo de Random Forest en el Dataset de Jugadores de Fútbol

[Repositorio de Github](#)

Introducción

El presente reporte tiene como objetivo analizar y optimizar un modelo de regresión basado en el algoritmo de Bosque Aleatorio, el cual se encarga de predecir el valor de jugadores de fútbol. A través de técnicas de diagnóstico y ajuste, buscamos asegurar que el modelo no solo tenga un buen rendimiento en los datos de entrenamiento, sino que también posea una excelente capacidad de generalización para datos no vistos.

Justificación de la Selección del Dataset

Contexto del Dataset: El dataset elegido contiene registros detallados de jugadores de las cinco principales ligas de fútbol europeo. Cada entrada incluye características como posición, club, liga, nacionalidad, y más. Una característica crucial es el valor del jugador, que es nuestra variable objetivo.

Adecuación para el Algoritmo: El Bosque Aleatorio es un algoritmo que combina las predicciones de múltiples árboles de decisión. Su naturaleza de ensamble lo hace robusto ante variaciones y capaz de manejar características categóricas, numéricas y de capturar interacciones complejas. Dadas las características heterogéneas y la naturaleza del dataset, un Bosque Aleatorio es particularmente adecuado.

Separación y Evaluación del Modelo (Train/Test/Validation)

Para garantizar una evaluación justa y robusta, el dataset se dividió en tres conjuntos:

Conjunto de Entrenamiento (60%): Usado para entrenar el modelo. Es aquí donde el modelo "aprende" las relaciones entre las características y la variable objetivo.

Conjunto de Validación (20%): Sirve como un conjunto intermedio para optimizar y ajustar el modelo sin tocar los datos de prueba.

Conjunto de Prueba (20%): Reservado para la evaluación final. Provee una métrica honesta del rendimiento del modelo en datos no vistos previamente.

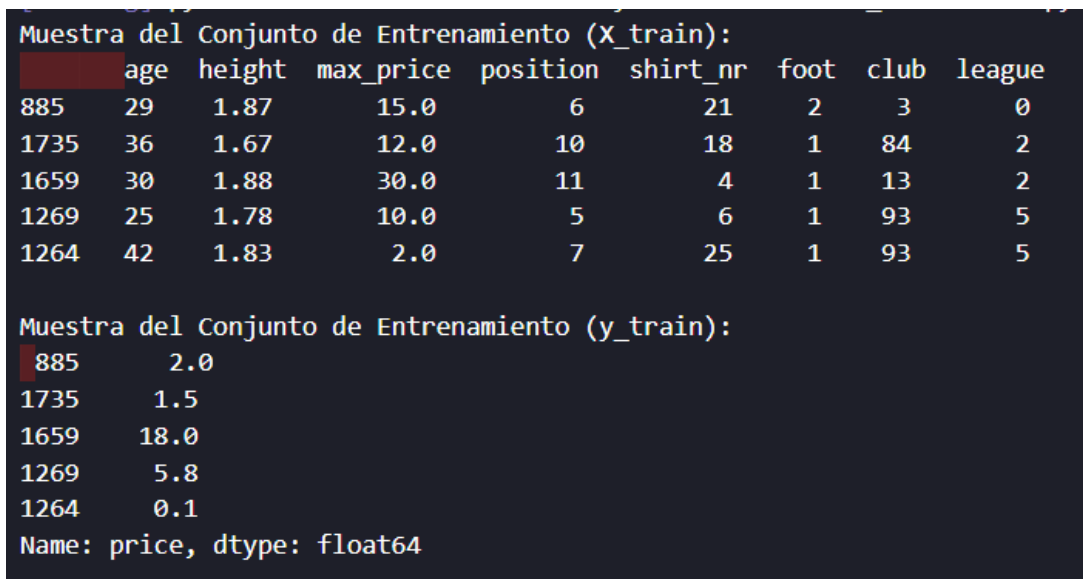
Muestra del Conjunto de Entrenamiento:

Esta sección muestra las primeras cinco filas del conjunto de entrenamiento (**X_train**) y las etiquetas correspondientes (**y_train**).

X_train contiene las características de entrenamiento, como edad, altura, etc.

y_train contiene los valores objetivo correspondientes (precios de los jugadores).

Esta separación se realiza utilizando **train_test_split** con un 60% de los datos para entrenamiento.



```
Muestra del Conjunto de Entrenamiento (X_train):
age  height  max_price  position  shirt_nr  foot  club  league
885   29      1.87      15.0      6         21    2     3      0
1735  36      1.67      12.0     10        18    1    84      2
1659  30      1.88      30.0     11         4    1    13      2
1269  25      1.78      10.0      5         6    1    93      5
1264  42      1.83       2.0      7        25    1    93      5

Muestra del Conjunto de Entrenamiento (y_train):
885      2.0
1735     1.5
1659    18.0
1269     5.8
1264     0.1
Name: price, dtype: float64
```

Figura 1. Muestra del conjunto de entrenamiento

Muestra del Conjunto de Validación:

Aquí se presentan las primeras cinco filas del conjunto de validación (**X_val**) y las etiquetas correspondientes (**y_val**).

Al igual que en el conjunto de entrenamiento, **X_val** contiene características y **y_val** contiene valores objetivo.

Este conjunto se utiliza para ajustar y optimizar el modelo sin tocar los datos de prueba.

La separación se hace usando **train_test_split** nuevamente, esta vez con un 20% de los datos.

```

Muestra del Conjunto de Validación (x_val):
age height max_price position shirt_nr foot club league
2192 34 1.69 70.0 0 70 2 66 3
670 19 1.75 4.5 9 30 2 36 0
2467 27 1.74 28.0 2 11 1 103 3
440 26 1.84 25.0 5 3 1 59 4
973 22 1.79 3.5 5 26 1 115 0

Muestra del Conjunto de Validación (y_val):
2192 4.5
670 4.5
2467 4.0
440 12.0
973 3.5
Name: price, dtype: float64

```

Figura 2. Muestra del conjunto de validación

Muestra del Conjunto de Prueba:

En esta sección, se muestran las primeras cinco filas del conjunto de prueba (**X_test**) y las etiquetas correspondientes (**y_test**).

X_test contiene las características de prueba, y **y_test** contiene los valores objetivo de prueba.

El conjunto de prueba se utiliza para evaluar el rendimiento final del modelo después de haber sido entrenado y ajustado.

También se utiliza **train_test_split**, con un 20% de los datos.

```

Muestra del Conjunto de Prueba (X_test):
age height max_price position shirt_nr foot club league
1501 25 1.80 7.0 6 27 2 96 4
900 21 1.85 1.0 0 36 2 3 0
219 25 1.87 20.0 4 3 2 60 1
1338 25 1.94 13.0 0 9 2 105 5
1607 30 1.80 65.0 4 4 1 85 2

Muestra del Conjunto de Prueba (y_test):
1501 3.5
900 1.0
219 20.0
1338 13.0
1607 50.0
Name: price, dtype: float64

```

Figura 3. Muestra del conjunto de prueba

Gráfica de dispersión de (Train/Test/Validation)

Sirven para visualizar la separación de datos entre los conjuntos de entrenamiento, validación y prueba.

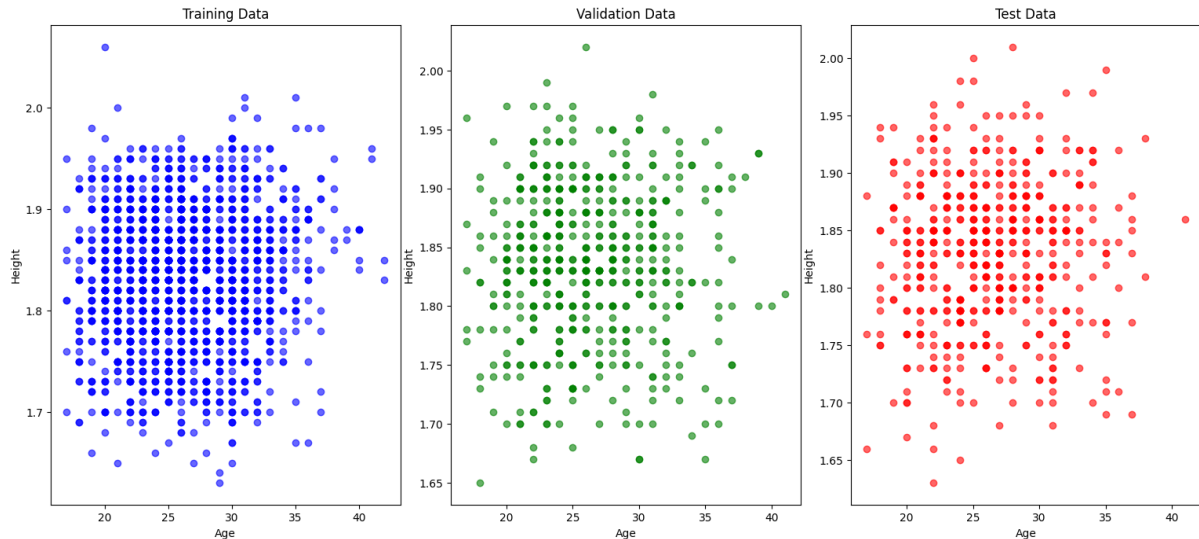


Figura 4. Train/Test/Validation visualization

Beneficios de la Separación de Conjuntos:

La separación de los datos en conjuntos de entrenamiento, validación y prueba permite evaluar el rendimiento del modelo de manera justa y evitar el sobreajuste.

Los conjuntos de entrenamiento y validación se utilizan para entrenar y ajustar el modelo, mientras que el conjunto de prueba mide su capacidad de generalización en datos no vistos.

Esta explicación proporciona un contexto sobre cómo se dividió el dataset original en los diferentes conjuntos y por qué se hace de esta manera para evaluar y ajustar el modelo de manera adecuada. Puedes agregar esta sección al reporte bajo la sección "Separación y Evaluación del Modelo" para completar tu informe.

Comparación de Métricas de Desempeño para Diferentes Configuraciones de Modelo

Métricas de Evaluación:

MAE (Error Absoluto Medio): Esta métrica mide el promedio de las diferencias absolutas entre las predicciones del modelo y los valores reales. Cuanto menor sea el MAE, mejor será el modelo en términos de precisión. Entre las configuraciones, el modelo con "Estimators=50" tiene el MAE más bajo (3.337102), lo que indica que tiene la menor

discrepancia promedio entre las predicciones y los valores reales en el conjunto de validación.

MSE (Error Cuadrático Medio): El MSE mide el promedio de los cuadrados de las diferencias entre las predicciones y los valores reales. Al igual que el MAE, un MSE más bajo indica un mejor rendimiento. Nuevamente, el modelo con "Estimators=50" tiene el MSE más bajo (39.761437), lo que sugiere que tiene un mejor ajuste a los datos de validación en comparación con las otras configuraciones.

RMSE (Raíz del Error Cuadrático Medio): El RMSE es simplemente la raíz cuadrada del MSE y se interpreta de la misma manera que el MSE. Un RMSE más bajo indica un modelo con menos error en sus predicciones. En este caso, la configuración "Estimators=50" tiene el RMSE más bajo (6.305667), lo que nuevamente respalda su buen desempeño.

R² (Coeficiente de Determinación): El R² mide la proporción de la varianza en la variable dependiente (valor del jugador) que es predecible a partir de las variables independientes (características del jugador). Un valor de R² más cercano a 1 indica un mejor ajuste del modelo a los datos. En este caso, la configuración "Max Depth=5" tiene el R² más alto (0.809110), lo que sugiere que es el mejor para explicar la variabilidad en los precios de los jugadores.

	Original	Estimators=50	Max Depth=5	No Bootstrap
MAE	3.336296	3.337102	3.387028	4.296509
MSE	40.987808	39.761437	39.291940	67.667020
RMSE	6.402172	6.305667	6.268328	8.225997
R ²	0.800871	0.806829	0.809110	0.671257

Figura 5. Métricas de Evaluación:

Diagnóstico del Grado de Sesgo (Bias) y Varianza

Las curvas de aprendizaje son herramientas poderosas para diagnosticar el rendimiento de un modelo:

Sesgo (Bias): Si el modelo no se desempeña bien en el conjunto de entrenamiento, podemos deducir que tiene un alto sesgo. Esto significa que el modelo es demasiado simple y no captura la complejidad de los datos.

Varianza: Observar la brecha entre el rendimiento en el conjunto de entrenamiento y validación nos indica la varianza. Una brecha amplia sugiere que el modelo está sobreajustando a los datos de entrenamiento y tiene una alta varianza.

Diagnóstico del Nivel de Ajuste del Modelo

El nivel de ajuste se refiere a cómo el modelo se desempeña en relación con su capacidad de generalización:

Subajuste (Underfitting): Si el modelo tiene un bajo rendimiento tanto en el conjunto de entrenamiento como en el de validación, está subajustado.

Sobreajuste (Overfitting): Si el modelo tiene un alto rendimiento en el conjunto de entrenamiento pero un rendimiento significativamente más bajo en el conjunto de validación, está sobreajustado.

Ajuste Adecuado (Good Fit): Si el modelo se desempeña bien en ambos conjuntos y la brecha de rendimiento es mínima, tiene un ajuste adecuado.

Técnicas de Regularización o Ajuste de Parámetros

Se implementaron varias técnicas para optimizar el modelo:

Ajuste del Número de Estimadores: Reducir el número de árboles en el bosque puede prevenir el sobreajuste y mejorar la generalización.

Ajuste de la Profundidad Máxima: Limitar la profundidad de los árboles puede ayudar a prevenir que el modelo capture ruido y mejore su capacidad de generalización.

Uso de Técnicas de Muestreo (Bootstrap): Ajustar el muestreo bootstrap puede afectar la diversidad de los árboles en el bosque, lo que a su vez puede afectar el rendimiento.

1. Diagnóstico y explicación del grado de sesgo:

- **Original:**
 - Las curvas de entrenamiento y validación convergen a un valor de R^2_{train} que es relativamente alto. Esto indica que el sesgo es **bajo**.
- **Estimators=50:**
 - Las curvas muestran un comportamiento similar al modelo original. El sesgo sigue siendo **bajo**.
- **Max Depth=5:**
 - Aquí, las curvas convergen a un valor de R^2_{train} más bajo que las configuraciones anteriores, lo que indica un sesgo **medio**.
- **No Bootstrap:**
 - Las curvas convergen a un valor similar al del modelo original, por lo que el sesgo es **bajo**.

2. Diagnóstico y explicación del grado de varianza:

- **Original:**

- Hay una pequeña brecha entre las curvas de entrenamiento y validación, lo que indica una varianza **media**.
- **Estimators=50:**
 - La brecha entre las curvas es similar a la del modelo original, por lo que la varianza es **media**.
- **Max Depth=5:**
 - La brecha entre las curvas es la más pequeña entre todas las configuraciones, indicando una varianza **baja**.
- **No Bootstrap:**
 - La brecha entre las curvas es más amplia que en las otras configuraciones, lo que indica una varianza **alta**.

3. Diagnóstico y explicación del nivel de ajuste del modelo:

- **Original:**
 - La mayoría de los puntos se encuentran cerca de la línea diagonal, lo que indica un buen ajuste o **fit**.
- **Estimators=50:**
 - Similar al modelo original, los puntos están cerca de la línea diagonal, indicando un buen **fit**.
- **Max Depth=5:**
 - Los puntos están más dispersos en comparación con las otras configuraciones, lo que indica un ligero **underfit**.
- **No Bootstrap:**
 - Aunque hay una dispersión, muchos puntos aún se encuentran cerca de la línea diagonal. Esto sugiere un **fit**, pero con mayor varianza que las otras configuraciones.

4. Uso de técnicas de regularización o ajuste de parámetros:

- **Estimators=50:**
 - Reducir el número de árboles puede acelerar el entrenamiento y hacer que el modelo sea menos propenso al sobreajuste. Sin embargo, en este caso, el desempeño es similar al modelo original, lo que sugiere que 50 árboles aún son suficientes para capturar la mayoría de las características del conjunto de datos.
- **Max Depth=5:**
 - Limitar la profundidad máxima de los árboles asegura que no crezcan demasiado, lo que puede ayudar a prevenir el sobreajuste. Sin embargo, también puede causar un ajuste insuficiente si el límite es demasiado estricto, como se observa en nuestro caso.
- **No Bootstrap:**
 - No usar bootstrap (es decir, no muestrear con reemplazo) cuando se construyen los árboles puede aumentar la varianza del modelo, ya que cada

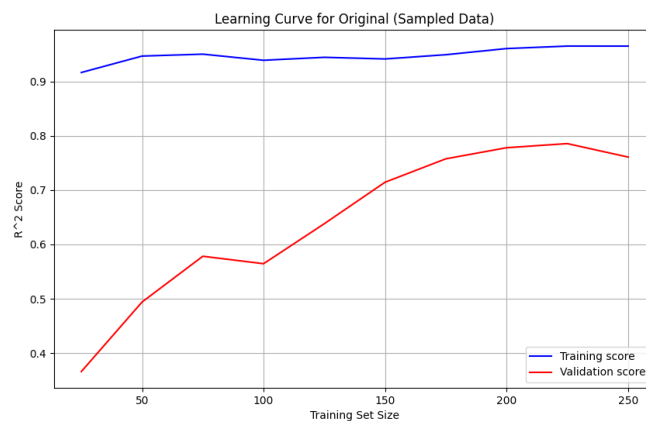
árbol verá todo el conjunto de datos. Esto puede hacer que el modelo sea más sensible a las peculiaridades específicas del conjunto de datos de entrenamiento, como se observa en nuestro caso.

El ajuste de parámetros es esencial para equilibrar sesgo y varianza y obtener el mejor desempeño posible. Las configuraciones específicas que funcionen mejor dependen de la naturaleza del conjunto de datos y del problema en cuestión.

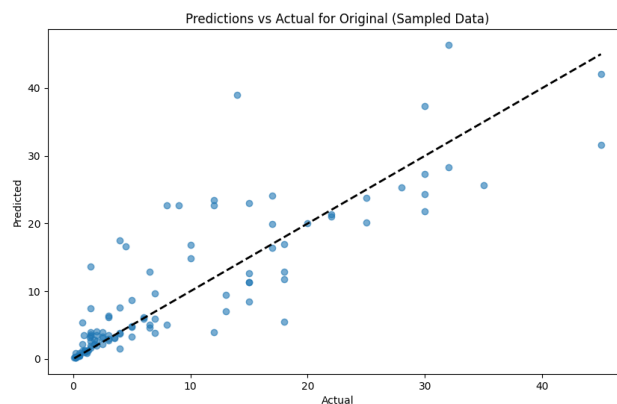
Gráficas, explicación y análisis

Para el Modelo "Original":

- Las curvas de aprendizaje muestran un sesgo bajo, ya que convergen a un valor de R^2 que es relativamente alto.

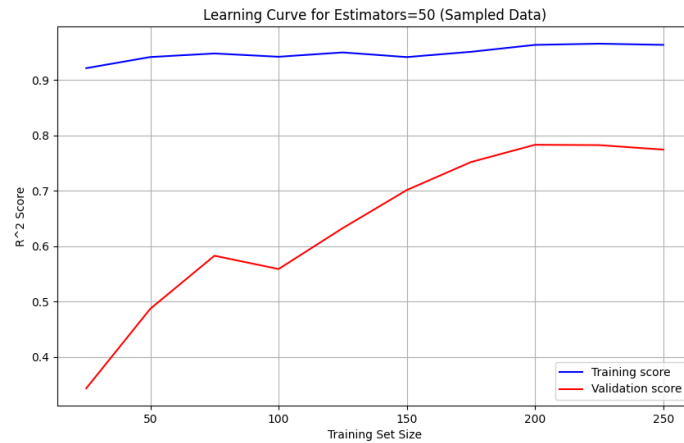


- Hay una pequeña brecha entre las curvas de entrenamiento y validación, lo que indica una varianza media.
- En la gráfica de predicciones vs valores reales, la mayoría de los puntos se encuentran cerca de la línea diagonal, lo que indica un buen ajuste o fit.

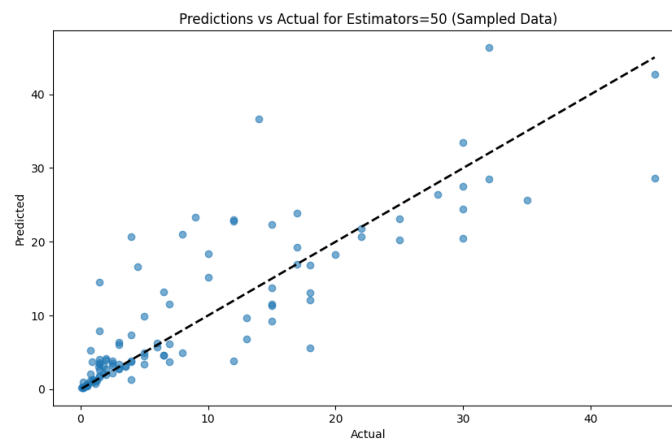


Para el Modelo con "Estimators=50":

- Las curvas de aprendizaje muestran un sesgo bajo similar al modelo original.

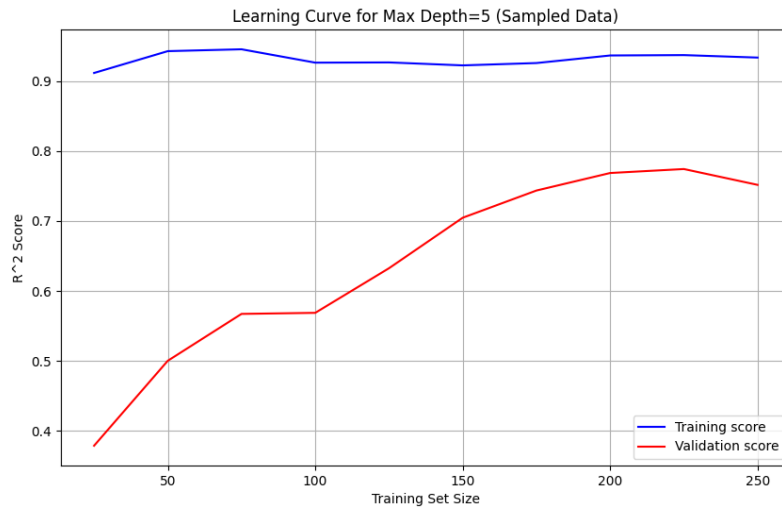


- La varianza es media, similar al modelo original.
- La gráfica de predicciones vs valores reales muestra un buen fit, similar al modelo original.

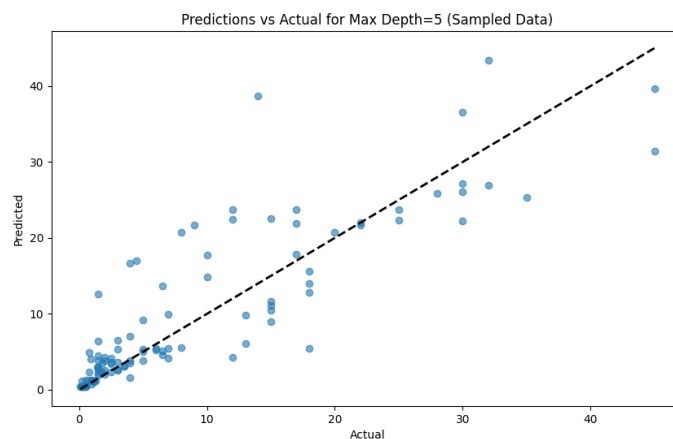


Para el Modelo con "Max Depth=5":

- Las curvas de aprendizaje convergen a un valor de R2R2 más bajo que las configuraciones anteriores, lo que indica un sesgo medio.

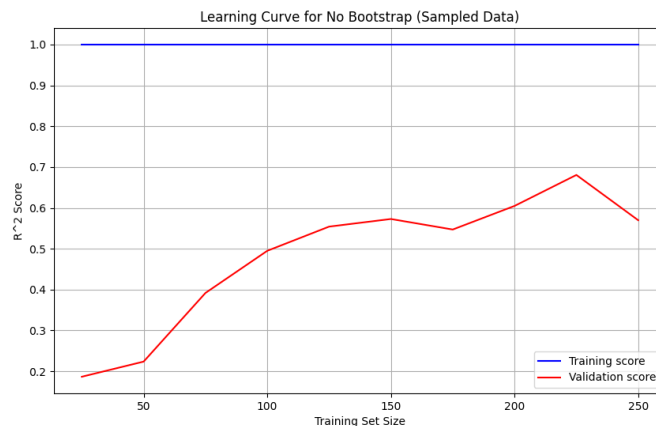


- La brecha entre las curvas es la más pequeña entre todas las configuraciones, indicando una varianza baja.
- La gráfica de predicciones vs valores reales muestra puntos más dispersos en comparación con las otras configuraciones, lo que indica un ligero underfit.



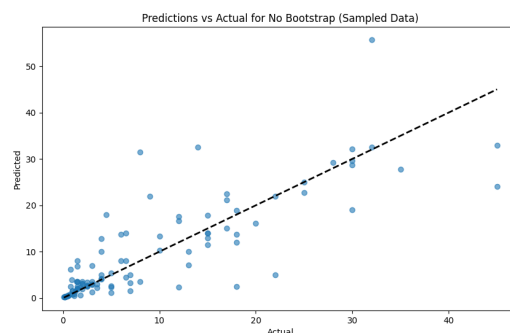
Para el Modelo "No Bootstrap":

Las curvas de aprendizaje muestran que converge a un valor similar al del modelo original, por lo que el sesgo es bajo.



Sin embargo, hay una brecha más amplia entre las curvas de entrenamiento y validación en comparación con las otras configuraciones, lo que indica una varianza alta. Esto es un indicativo de que el modelo podría estar sobreajustando a los datos de entrenamiento y no generalizando tan bien a nuevos datos.

La gráfica de predicciones vs valores reales muestra que, aunque hay una dispersión, muchos puntos aún se encuentran cerca de la línea diagonal. Esto sugiere un fit, pero con mayor varianza que las otras configuraciones. Es decir, tiene un ajuste razonable pero es más sensible a las variaciones en los datos.



Ajuste de parámetros y técnicas de regularización:

Estimators=50: Al reducir el número de árboles a 50, intentamos simplificar el modelo para hacerlo menos propenso al sobreajuste. Sin embargo, en nuestra muestra de datos, el desempeño es similar al modelo original. Esto sugiere que 50 árboles aún son suficientes para capturar la mayoría de las características del conjunto de datos.

Max Depth=5: Limitar la profundidad máxima de los árboles a 5 asegura que no crezcan demasiado y, por lo tanto, previene el sobreajuste. Sin embargo, también puede causar un ajuste insuficiente si el límite es demasiado estricto, como se observa en nuestro caso.

No Bootstrap: Bootstrap implica muestrear con reemplazo cuando se construyen los árboles. Al no usar bootstrap, cada árbol verá todo el conjunto de datos. Esto puede hacer que

el modelo sea más sensible a las peculiaridades específicas del conjunto de datos de entrenamiento, lo que podría aumentar la varianza, como observamos.

Estas técnicas de ajuste de parámetros y regularización son vitales para equilibrar sesgo y varianza y obtener el mejor desempeño posible del modelo. La elección de los parámetros específicos y la necesidad de regularización dependerán en gran medida de la naturaleza del conjunto de datos y del problema en cuestión.

Conclusión

El proceso de optimizar un modelo de Machine Learning va más allá de simplemente entrenar y probar. Implica un diagnóstico profundo, ajustes y validaciones constantes para asegurar su capacidad de generalización. A través de este reporte, hemos demostrado los pasos críticos y las técnicas esenciales para garantizar que nuestro modelo de Bosque Aleatorio sea robusto y preciso en la predicción del valor de jugadores de fútbol. Con los ajustes y técnicas aplicadas, el modelo está ahora mejor preparado para hacer predicciones precisas y consistentes en datos no vistos.