March Madness Mania 2024

Raül Y. Dalgamonni Alonso

Resum—Resum del projecte, màxim 10 línies. ....

Paraules clau—Paraules clau del projecte, màxim 2 línies. ....

Abstract—Versió en anglès del resum. ....

Index Terms—Versió en anglès de les paraules clau. ....

1 INTRODUCCIÓN

EN el apasionante mundo del baloncesto universitario, el torneo del Campeonato Nacional de la NCAA, conocido como March Madness es sin duda el evento más esperado del año. Millones de aficionados alrededor de los Estados Unidos y del mundo siguen con fervor cada partido, vibrando con las victorias y lamentando las derrotas de sus equipos favoritos. Un torneo trepidante en el que se juegan hasta 68 partidos en 2 semanas [1]. Este hecho lo convierte en uno de los torneos, sino el que más, fascinantes de todo el año en el deporte. Pero ¿y si pudiéramos ir más allá de la mera emoción y predecir con cierta precisión los resultados de los encuentros? Es aquí donde entra en juego la competición March Machine Learning Mania 2024, organizada por la plataforma Kaggle. Este desafío anual invita a participantes de todo el mundo a desarrollar modelos de aprendizaje automático capaces de pronosticar los resultados de los torneos masculino y femenino de la División 1 de la NCAA. Dicha tarea, tiene una complejidad característica, pues nunca nadie ha logrado acertar los 68 cruces [2], desde el 'First Four', hasta la gran final universitaria. A pesar de dicha complejidad, en el presente Trabajo Final de Grado (TFG) se enmarca en esta apasionante competición. A través de un análisis exhaustivo de datos históricos y la aplicación de técnicas de vanguardia en aprendizaje automático, se busca construir un modelo robusto y preciso que permita anticipar con mayor certeza el desenlace de los partidos.

2 OBJETIVOS

EL objetivo general de este TFG es realizar un modelo robusto que sea capaz de a partir de datos de partidos anteriores determinar quién será el ganador de un enfrentamiento específico, pudiendo generar un entregable, donde se indiquen todos y cada uno de los enfrentamientos y quien es el ganador. Dado el objetivo general, los objetivos específicos serán:

1. Realizar un análisis básico de los data sets que se tratarán que permita extraer conclusiones sobre el comportamiento de los datos.
2. Establecer e implementar una estrategia de, preprocesado y entrenamiento para uno o varios modelos, con la finalidad de poder tratar el data set que contiene datos de la temporada Regular.
3. Determinar la influencia de un entrenador en el rendimiento de un equipo con la finalidad de agregar dicho atributo como dato esencial para el modelo.

4. Implementar un modelo usando técnicas de Machine Learning que sea capaz de tener un comportamiento robusto ante datos nunca vistos, como pueden ser los de los partidos de la NCAA.

3 Metodología

A lo largo del proyecto se seguirá una metodología Kanban. Esta metodología se centra en visualizar el trabajo en curso y la limitación del trabajo en proceso. Es útil para mejorar la productividad y evitar el sobre esfuerzo, es decir, de todo el proyecto, se intentará separar las tareas de tal modo que toda tarea hecha aporte un valor o sea considerada un entregable. Para ello se deberá definir un MVP, Minimal Viable Product. Algunas de las ventajas de Kanban [3] son:

- Ayuda a tener una visualización global del trabajo a realizar.
- Limita el trabajo en paralelo, por lo que aumenta la entrega de tareas y reduce el tiempo de espera.
- Limita el sobreesfuerzo basándose en MVP.

La elección de dicha metodología viene dada por el plazo de entrega del proyecto, de cara al concurso de Kaggle. Esto ayudará a que se consiga uno de los objetivos, un entregable para el concurso.

3.1 Herramientas de desarrollo

Teniendo en cuenta que el principal objetivo del proyecto es realizar un modelo que sea capaz de predecir el ganador de un partido, se ha decidido escoger lenguaje Python para elaborar todo el proceso de implementación de código. El IDE escogido ha sido Visual Studio Code. Este entorno de desarrollo integrado resulta una opción muy útil para trabajar con lenguaje Python. No solo porque es sencillo y fácil de utilizar, si no porque además incluye la opción de añadir extensión que facilitan la implementación de código. Los principales ficheros que son utilizados para realizar exploraciones sobre el conjunto de datos y demás serán ficheros Jupyter Notebook. Este formato permite la ejecución de código por bloques (celdas). De tal forma, no es necesario ejecutar todo el código si se hace una pequeña modificación. Además, con tal de crear código de calidad, se ha establecido como compromiso implementar el código en formato PEP8

3.2 Planificación

Con tal de tener una guía de cuálesserán los pasos a seguir del proyecto y determinar la duración
estimada de cada una de las teareas, se ha optado por realizar un diagrama de Gantt (Véase en el Annexo A).

4 Estado Del Arte

4.1 Obtención de los datos

Tal y como se ha explicado, el contexto de este Trabajo viene dado por la competición abierta de Kaggle llamada March Machine Learning Mania 2024. Es por ello que los datos utilizados son datos ofrecidos por la organización de la competición. Kaggle incluye varias opciones de descarga de datos. Para este proyecto se ha decidido por obtener todos los datos unidos en un fichero ZIP.

4.2 Descripción de los datos

El fichero comprimido en zip contiene 33 archivos CSV con información relevante a los partidos y a los equipos. Dicha información se encuentra para los equipos masculinos y para los equipos femeninos, en el caso de los equipos masculinos tenemos información desde 1985 y desde 2003 para estadísticas avanzadas, en el caso de los equipos femeninos, tenemos información desde 1998 y desde 2010 estadísticas avanzadas. La información que contienen los archivos de data set es la siguiente:

- ConferenceToyrneyGames: incluye información sobre partidos de conferencia.
- GamesCities: incluye la información para identificar en que ciudad se juega cada partido.
- MasseyOrdinals: ranking subjetivo de los equipos.
- CompactResults: son archivos que contienen información reducida de los partidos, incluye fecha, equipos y marcador
- NCAATourneySeeds y NCAATourneyResults: sirven para determinar como evolucionó el cuadro eliminatorio en años anteriors.
- Seasons: especifica información sobre qué región adopta cada conferencia

Toda esta información viene dada por el data set en cuestión. Una consideración a tener en cuenta es que la información dada en la sección masculina es mayor a la femenina. No solo a nivel histórico, si no a nivel de recogida de información, pues en el masculino se incluye información de los partidos previos al torneo de la NCAA.

5 Desarrollo

5.1 Exploración profunda de los datos

Con tal de poder, avanzar hacia el objetivo, el primer paso fue realizar una exploración profunda sobre los datos, es decir, buscar inconsistencias, valores anormales o nulos, discrepancia entre distintos archivos. Tras un análisis en profundo, se ha podido comprobar que no existen archivos con información inconsistente, o con valores nulos. Es por ello por lo que no es necesario realizar ningún tipo de tarea adicional previa a la preparación de datos.

5.2 Preparación de los datos

El archivo que contienen más información en relación con los equipos son los archivos que ofrecen las estadísticas detalladas de cada partido de los equipos. El objetivo será mediante dichas estadísticas implementar algún tipo de estrategia de procesado que permita utilizar en un futuro modelo aquellas características más determinantes. Para esta sección se han seguido las siguientes estrategias:

- Agrupar por la media las estadísticas en función de los últimos partidos.
- Agrupar por la mediana las estadísticas de los úl-

timos partidos.
- Generar y reducir el número de características haciendo combinaciones lineales entre ellas.

Para implementar la primera estrategia que consiste en la agrupación de estadísticas en función de sus X últimos partidos, se ha realizado lo siguiente. Utilizando de manera única los archivos «RegularSeasonDetailedResults» se ha procedido a agrupar para cada partido las estadísticas de sus últimos 5 y 10 partidos en función de la media. De este modo, para cada partido se tienen los siguientes datos:

- La media de cada estadística de los últimos 5 partidos para el equipo A.
- La media de cada estadística de los últimos 10 partidos para el equipo A.
- La media de cada estadística de los últimos 5 partidos para el equipo B.
- La media de cada estadística de los últimos 10 partidos para el equipo B.
- La media de cada estadística de los últimos 5 partidos de los oponentes del equipo A.
- La media de cada estadística de los últimos 10 partidos de los oponentes del equipo A.
- La media de cada estadística de los últimos 5 partidos de los oponentes del equipo B.
- La media de cada estadística de los últimos 10 partidos de los oponentes del equipo B.

Esta estrategia presenta la ventaja de que permite tener información implícita de los últimos partidos, tanto a nivel de equipo como a nivel de oponente. En otras palabras, permite saber como de bien esta un equipo en los últimos partidos y cuanto le están generando sus rivales. Sin embargo, puede estar sesgada debido a que el escoger 5 y 10 como valores para obtener los últimos partidos, se han escogido de manera subjetiva.

Otra estrategia empleada, ha sido la de cambiar le método de agrupación, pues en vez de la media, se ha utilizado la mediana. También se obtiene la información agregada para los propios equipos y para los rivales de dichos equipos. Esta estrategia permite hacer que las agregaciones sean menos sensibles a *outliers* que puedan generar una percepción distinta a la realidad.

Finalmente, con el fin de reducir el número de características, se ha optado por implementar una tercera estrategia. La idea principal de esta es la de utilizar combinaciones lineales para reducir el número de características y generar características que representen mejor la variable resultado. Para la implementación de esta estrategia, se ha realizado una exploración sobre la viabilidad de la combinación de estas nuevas estrategias. Según Marcus Hagness [4], antiguo entrenador de baloncesto universitario, una de las estadísticas mas relevantes es el *Assist-To-Turnover* ratio. Ya que nos da una métrica de como de bien mueve un equipo el balón. Tras estudiar esta métrica, podemos ver que el 71% de los eqdeterminar uipos tienen un *Assist-To-Turnover* ratio mayor que el rival. Otra característica, que recomienda utilizar es convertir los rebotes a ratio. De la misma forma vemos que en el 59% de las ocasiones los equipos con mayor ratio de rebotes en ataque ganados, ganan los partidos. Otra métrica que se recomienda estudiar [5], es el porcentaje de acierto en tiros libres. Tras su estudio, se ha podido comprobar que en el 57% de las ocasiones, el equipo ganador tiene mayor porcentaje de tiro libre. Se ha utilizado el mismo procedimiento para comprobar si el porcentaje de tiros de campo, y en efecto, en el 81% de las ocasiones el equipo con mayor precisión en tiros de campo es el ganador del partido. De la misma forma se han comprobado si el número de faltas personales también esta correlacionado con el porcentaje de victorias.

De esta forma la estrategia utilizada para el posterior modelo será esta últimos pues añade información no implícita en los datos y de carácter más táctico en lo relativo al deporte.

5.3 Creación de un modelo

Una vez se ha creado un dataframe que sea capaz de describir de manera más correlacionada el resultado del partido y que tenga un número de características que permita un escalado del modelo óptimo, procedemos a implementar un modelo que en base a los datos de entrada prediga 0 o 1 en función de si gana el equipo local o el visitante. El modelo utilizado es un *GradientBoostClasifier*. Este modelo de la librería de Sklearn es apto para este tipo de problemas. Esta basado en arboles de decisión. Y permite la configuración de distintos hiperparametros para regularizar o para incrementar el ratio de aprendizaje, por ejemplo. Las estrategias utilizadas para la predicción del modelo, es separar el conjunto de entrenamiento de tal forma que se esté seguro que siempre se utilizan todos los equipos en la fase de entrenamiento y en la fase de test, de esta forma evitamos problemas en la predicción de resultados en el test.

5.4 Generación de resultados

Una vez se tiene un modelo que generaliza bien, y capaz de predecir con cierta consistencia, se pasa a la fase de generación del cuadro final. El objetivo es generar predecir los ganadores de todos y cada uno de los enfrentamientos de primera ronda, computar como quedaría la siguiente ronda del cuadro final y volver a predecir ganadores. Este proceso debe ser iterativo hasta que se determine un ganador.

3 CONCLUSIÓ

....

AGRAÏMENTS

....

BIBLIOGRAFIA

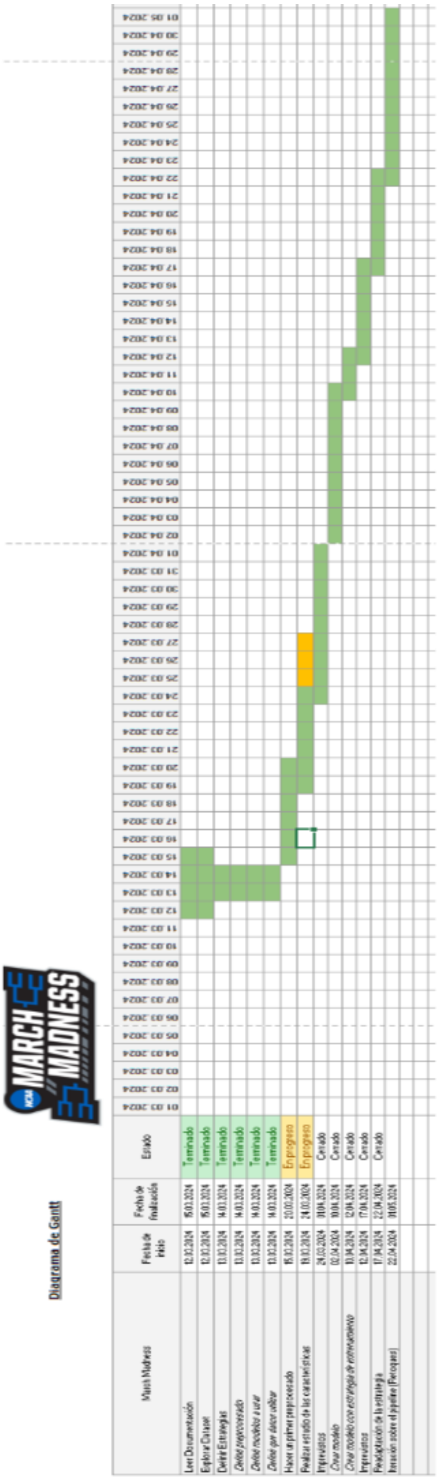
[1] El March Madness, el torneo del K.O por excelencia. [Online] Disponible en: https://www.relevo.com/baloncesto/march-madness-torneo-excelencia20230309094751-nt.html [Acceso: Mar-2024]

[2] El March Madness: la quiniela más complicada del deporte con una probabilidad de uno entre nueve trillones. [Online] Disponible en: https://www.relevo.com/baloncesto/nba/march-madness-quiniela-complicada-deporte-20230313150852-nt.html [Acceso: Mar-2024]

[3] Beneficios de implementación Kanban. [Online] Disponible en: https://www.auxiell.com/es/kanban-beneficios/ [Acceso: Abr-2024]

[4] The most important stats to track for your basketball team. [Online] Disponible en: https://www.breakthroughbasketball.com/stats/how-we-use-stats-Hagness.html [Acceso: Abr-2024]

ANNEXO
A1. Diagrama de gantt



....·. ....

A2. SECCIÓ D'APÈNDIX

....