

# Implementación de Machine learning para el pronóstico de resultados en los torneos de Basketball de la división 1 de la NCAA.

Especialización en Análítica y Ciencia de Datos.

Universidad de Antioquia.

Presentado por: Felipe Ramírez Vargas

Cédula: 1.152.217.130

email: [feramirezva@unal.edu.co](mailto:feramirezva@unal.edu.co)

GitHub: <https://github.com/FelipeRam22/MonografiaUDEA>

Asesor: Carmen Elena Patiño.

## 1. Descripción del problema.

El proyecto a ejecutar es una de las competencias con incentivo económico que ofrece la plataforma Kaggle.com denominado "March Machine Learning Mania 2023" (<https://www.kaggle.com/competitions/march-machine-learning-mania-2023>). Por medio de los dataset suministrados se buscará generar una predicción utilizando metodologías de Machine learning para determinar la probabilidad de ganar un equipo a otro en la competencia National Collegiate Athletic - NCAA.

El problema a analizar se debe abordar a través de una metodología supervisada de regresión toda vez que, se busca determinar un resultado numérico el cual denota la probabilidad de que un equipo A le gane a un equipo B.

El día 17 de mayo de 2023, se sostuvo la primera reunión con la asesora de la monografía, que para este caso es la profesora Carmen Elena Patiño.

## 2. Artículos relacionados.

A continuación conforme a las instrucciones que se tienen para el segundo entregable de la asignatura seminario, se relacionan algunos artículos donde se implementaron metodologías de machine learning útiles en el desarrollo del modelo, a su vez, se relacionan desarrollos de códigos abiertos en la plataforma kaggle para el problema, entre ellos el ganador de la competencia.

- Ganador absoluto competencia kaggle:  
<https://www.kaggle.com/competitions/march-machine-learning-mania-2023/discussion/399553> usuario Rusty B
- Tercer puesto competencia kaggle:  
<https://www.kaggle.com/code/tihonby/march-madness-3th-place-solution/notebook> usuario Tihonby
- Código abierto plataforma kaggle  
<https://www.kaggle.com/code/theovieli/t-s-that-time-of-the-year-again> usuario Theo Viel

## 3. Análisis y limpieza de datos.

Para generar el análisis y determinar un modelo predictivo adecuado, se utilizaron las bases de datos suministradas por la plataforma kaggle, que si bien en principio relacionan una gran cantidad, se utilizaron las que mayor relevancia tienen y aportan al modelo, sin embargo, posteriormente para el segundo semestre de la especialización se utilizaran datos de casas de apuestas y las bases de datos que no se utilizaron para perfeccionar y ajustar los modelos.

Las bases de datos utilizadas fueron:

- MNCAATourneySeeds.csv
- WNCAATourneySeeds.csv
- MRegularSeasonCompactResults.csv
- WRegularSeasonCompactResults.csv
- WNCAATourneyCompactResults.csv
- MNCAATourneyCompactResults.csv

En el primer entregable de la asignatura, se

relaciona que en primera instancia se iban a analizar únicamente los datos de los hombres por practicidad, sin embargo, en este punto de la asignatura, se generó el ejercicio tanto para hombres como para mujeres.

A continuación, se analiza las bases de datos, se genera una descripción estadística y se concatenan las mismas con la información tanto para mujeres como para hombres, a continuación con miras a esbozar los planteamientos anteriormente mencionados, se plasman algunas gráficas y tablas que lo denotan.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 307855 entries, 0 to 307854
Data columns (total 7 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Season                307855 non-null int64
1   DayNum               307855 non-null int64
2   WTeamID              307855 non-null int64
3   WScore               307855 non-null int64
4   LTeamID              307855 non-null int64
5   LScore               307855 non-null int64
6   Diferencia Puntaje   307855 non-null int64
dtypes: int64(7)
memory usage: 16.4 MB
```

Tabla 1. Información Data Frame.

	Season	DayNum	WTeamID	WScore	LTeamID	LScore
count	307855.000000	307855.000000	307855.000000	307855.000000	307855.000000	307855.000000
mean	2007.565016	73.131594	2105.995917	74.739241	2103.407003	61.740881
std	10.118320	34.481003	987.829906	11.917605	990.160767	11.607099
min	1985.000000	0.000000	1101.000000	30.000000	1101.000000	11.000000
25%	2000.000000	43.000000	1261.000000	66.000000	1254.000000	54.000000
50%	2009.000000	75.000000	1413.000000	74.000000	1408.000000	61.000000
75%	2016.000000	103.000000	3245.000000	82.000000	3245.000000	69.000000
max	2023.000000	132.000000	3477.000000	186.000000	3477.000000	150.000000

Tabla 2. Medidas estadísticas.

Los datasets no cuentan con datos nulos.

#### 4. Procesamiento y modelado.

En primera instancia se podría pensar en que, teniendo el puntaje de los partidos jugados por cada equipo, se podría analizar la media de la diferencia de los partidos cuando se gana y cuando se pierde, esto genera una variable de importancia para cada equipo. A continuación se realiza un ejemplo en aras de demostrar la relevancia de este dato.

Media diferencia Equipo A Gana = 12 puntos  
 Media diferencia Equipo B Gana = 7 puntos  
 Media diferencia Equipo A Pierde = 2 puntos  
 Media diferencia Equipo B Pierde = 5 puntos

De esta manera en un enfrentamiento entre el equipo A y el B, el equipo A ganaría en principio toda vez que, normalmente cuando gana hace más puntos que el equipo B y cuando pierde recibe menos puntos que este.

A su vez, se puede condensar por equipo la cantidad de partidas ganadas, y generar una proporción de la misma frente al total de partidas.

De manera análoga, se puede generar una característica que delimite:

Promedio Diferencia = ((Partidas Ganadas\*Media diferencia Part Ganadas) - (Partidas pérdidas\*Media diferencia Partidas pérdidas))/(Partidas ganadas +Partidas pérdidas)

Esta medida es muy relevante porque entre mayor sea la magnitud (positivamente) más posibilidades tiene un equipo de ganarle a otro. Solo con este dato, se podría generar una regresión y una distribución de probabilidad e inferir si un equipo puede ganarle a otro.

Season	TeamID	Partidas Ganadas	Partidas Perdidas	Media Diferencia Puntaje Partidas ganadas	Media Diferencia Puntaje perdidas	Proporcion_ganadas	Promedio diferencia
1985	1102	5.0	19.0	10.000000	9.947368	0.208333	-5.791667
1985	1103	9.0	14.0	7.555556	9.857143	0.391304	-3.043478
1985	1104	21.0	9.0	13.190476	4.777778	0.700000	7.800000
1985	1106	10.0	14.0	9.500000	13.285714	0.416667	-3.791667
1985	1108	19.0	6.0	13.842105	10.666667	0.760000	7.960000
...	...	...	...	...	...	...	...
2023	3473	1.0	24.0	9.000000	18.500000	0.040000	-17.400000
2023	3474	5.0	21.0	10.200000	20.523810	0.192308	-14.615385
2023	3475	9.0	17.0	13.000000	12.176471	0.346154	-3.461538
2023	3476	8.0	20.0	10.125000	12.800000	0.285714	-6.250000
2023	3477	13.0	19.0	10.538462	15.631579	0.406250	-5.000000

Tabla 3. Promedio diferencia

A su vez, se consolida un dataframe que disgregue tanto para el equipo ganador como para el perdedor la proporción de partidas ganadas como el promedio de diferencia.

Season	DayNum	TeamIDa	Scorea	TeamIDb	Scoreb	SeedA	SeedB	Proporcion_ganadasa	Promedio_diferenciala	Proporcion_ganadasb	Promedio_diferencialb	
0	2015	137	3116	57	3321	55	10	7	0.566667	4.900000	0.741935	9.419355
1	2015	137	3124	77	3322	96	2	15	0.909091	20.666667	0.517241	-3.275862
2	2015	137	3143	78	3455	66	4	13	0.718750	7.343750	0.878788	13.424242
3	2015	137	3173	78	3235	66	7	10	0.808452	11.935484	0.600000	6.333333
4	2015	137	3177	79	3278	72	9	8	0.787879	16.909091	0.718750	4.583750

Tabla 4. Dataset consolidado

La plataforma kaggle, suministra un dataset para generar la validación y comparar resultados. A esta se le asignan los valores de promedio diferencia y proporción ganadas, como se muestra a continuación:

```
df_test.head()
```

	ID	PreD	Season	TeamIDA	TeamIDB	SeedA	SeedB	Proportion_ganadasA	Promedio diferenciaA	Proportion_ganadasB	Promedio diferenciaB
0	2023_1101_1102	0.5	2023	1101	1102	1	1	0.346154	-3.692308	0.437500	-0.125000
1	2023_1101_1103	0.5	2023	1101	1103	1	1	0.346154	-3.692308	0.645161	5.838710
2	2023_1101_1104	0.5	2023	1101	1104	1	1	0.346154	-3.692308	0.652941	13.676471
3	2023_1101_1105	0.5	2023	1101	1105	1	1	0.346154	-3.692308	0.400000	-3.066667
4	2023_1101_1106	0.5	2023	1101	1106	1	1	0.346154	-3.692308	0.233333	-10.033333

Tabla 5. Data test

Análogamente, como medida preliminar se puede hacer una diferencia entre el promedio de diferencia y si este valor es mayor que 0 quiere decir que en primera instancia el equipo A le gana al B, así generamos una nueva casilla.

De esta manera, se buscará generar una predicción mediante regresión lineal o logística. Sin embargo de manera preliminar, se buscará generar la normalización de la data para que los datos se encuentren en la misma magnitud

```
def minmax(caracteristica, df_train, df_val, df_test=None):
    min_ = df_train[caracteristica].min()
    max_ = df_train[caracteristica].max()

    df_train[caracteristica] = (df_train[caracteristica] - min_) / (max_ - min_)
    df_val[caracteristica] = (df_val[caracteristica] - min_) / (max_ - min_)
```

Análogamente para que se pueda hacer una buena validación de la data y el modelo se utilizará la metodología K-fold, la cual permitirá por medio de una validación cruzada analizar toda la data y que el modelo sea más robusto.

De esta manera, se genera una regresión en aras de calcular la probabilidad de que un equipo A le gane a un equipo B.

Para la regresión se utiliza la metodología de elastic Net, la cual puede ser útil por su

condición de penalización con los factores L1 y L2.

```
if mode == "reg":
    model = ElasticNet(alpha=1, l1_ratio=0.5)
else:
    model = LogisticRegression(C=1)

model.fit(df_train[caracteristica], df_train[target])
```

Es de esta manera que se obtienen los siguientes resultados para las temporadas 2016-2022.

```
pred_tests = kfold(dftor, df_test, plot=False, verbose=1, mode="cls")

Validating on season 2016
-> Scored 0.212

Validating on season 2017
-> Scored 0.203

Validating on season 2018
-> Scored 0.212

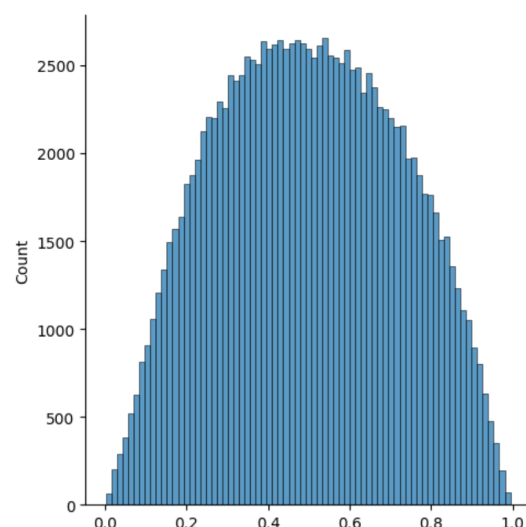
Validating on season 2019
-> Scored 0.195

Validating on season 2021
-> Scored 0.217

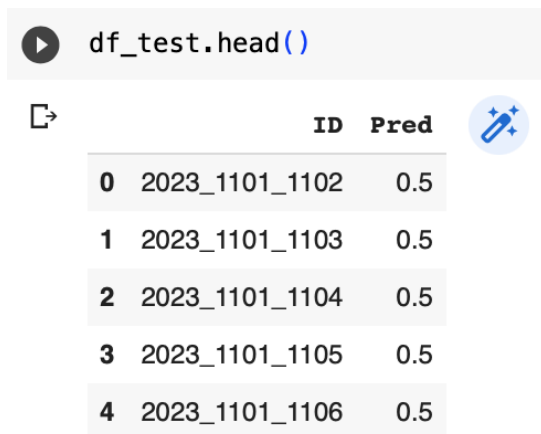
Validating on season 2022
-> Scored 0.218

Local CV is 0.210
```

Seguido de ello, se podrá graficar la distribución de frecuencia que se obtuvo para la predicción de la data. A continuación se plasma dicha gráfica.



Como se había relacionado anteriormente, desde la plataforma kaggle suministran unos datos para que sean validados.

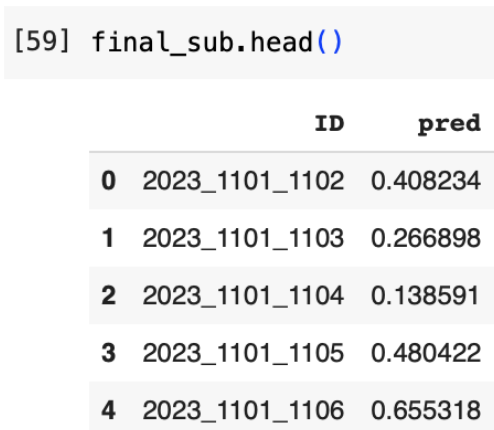


```
df_test.head()
```

	ID	Pred
0	2023_1101_1102	0.5
1	2023_1101_1103	0.5
2	2023_1101_1104	0.5
3	2023_1101_1105	0.5
4	2023_1101_1106	0.5

Tabla 6. data test inicial

Ahora bien, se procederá a generar la predicción.



```
[59] final_sub.head()
```

	ID	pred
0	2023_1101_1102	0.408234
1	2023_1101_1103	0.266898
2	2023_1101_1104	0.138591
3	2023_1101_1105	0.480422
4	2023_1101_1106	0.655318

Tabla 7. Predicción

Se puede percibir claramente que la data test inicial sus datos de predicción eran todos 0.5 ahora luego de la modelación se obtiene la probabilidad real de que el equipo A pueda ganarle al equipo B como se evidencia en la tabla 7.

## 5. Conclusiones.

1. Generar la Normalización de la data es algo fundamental, para que las variables se encuentren en una misma magnitud, permitiendo que el modelo sea adecuado.

2. El modelo que se genera es bueno, las predicciones que realiza son acertadas y son consecuentes con los puntos y partidas ganadas por cada equipo.
3. Para el siguiente semestre se podrá utilizar más datos para optimizar el modelo a implementar, además teniendo en cuenta algunos códigos de la competencia, podría ser útil implementar técnicas de Machine learning como XGboost. Otra alternativa para mejorar el modelo podría ser utilizar datos de casas de apuestas.
4. El modelo es sensible a la cantidad de temporadas utilizadas, es decir se puede utilizar más y menos años, y de esta manera buscar cual es la alternativa más adecuada.

## 6. Referencias.

- Notas de clase de las asignaturas vistas en la Especialización.
- Plataforma Kaggle, (<https://www.kaggle.com/competitions/march-machine-learning-mania-2023>)
- Ganador absoluto competencia kaggle: <https://www.kaggle.com/competitions/march-machine-learning-mania-2023/discussion/399553> usuario Rusty B
- Tercer puesto competencia kaggle: <https://www.kaggle.com/code/tihonby/march-madness-3th-place-solution/notebook> usuario Tihonby
- Código abierto plataforma kaggle <https://www.kaggle.com/code/theoviel/i-t-s-that-time-of-the-year-again> usuario Theo Viel
- XgBoost <https://xgboost.readthedocs.io/en/stable/>