Memoria Machine Learning Project

Yago García Marqués

26 de septiembre de 2021

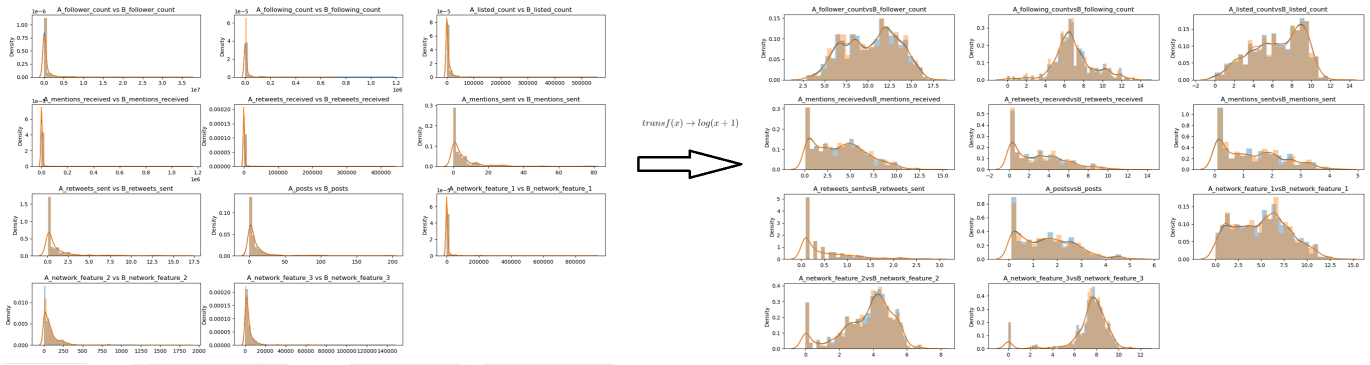
1. Introducción

El dataset proviene de la web Kaggle de una competición con enlace (`https://www.kaggle.com/c/predict-who-is-more-influential-in-a-social-network/`). Contiene 22 columnas con características de dos elementos (A B) de una red que compararemos en cada fila, el objetivo es construir un modelo que nos diga cual de los dos elementos es mas influyente en la red. Para ello contaremos con la columna 'Choice' que data de la opinión humana en una encuesta sobre cual de los dos elementos es mas influyente, si la salida es '1', A es mas influyente, si es '0', B lo será. A partir de la construcción del modelo no solo buscamos la predicción binaria, si no que evaluaremos cuales son las características mas importantes para que un elemento sea influyente en dicha red.

2. Depuración de los datos

En primer lugar nuestro dataset está pulido en cuanto a NA's y datos mal recogidos, además que hay 6 variables que no podemos conocer si están bien o no ya que desconocemos que significan (Network features). Realizamos una visualización general de los datos en formato tabla y con gráficos, observamos que es necesario tratar la muestra puesto que las variables difieren en unidades y por tanto en variabilidad, además como queremos plantear en un futuro un modelo de predicción, también nos interesa que nuestras variables tengan una forma razonablemente normal, por ello tras probar diversas transformaciones de las que hablaré en la presentación por encima, escojo la función logística sumando una unidad por variable:

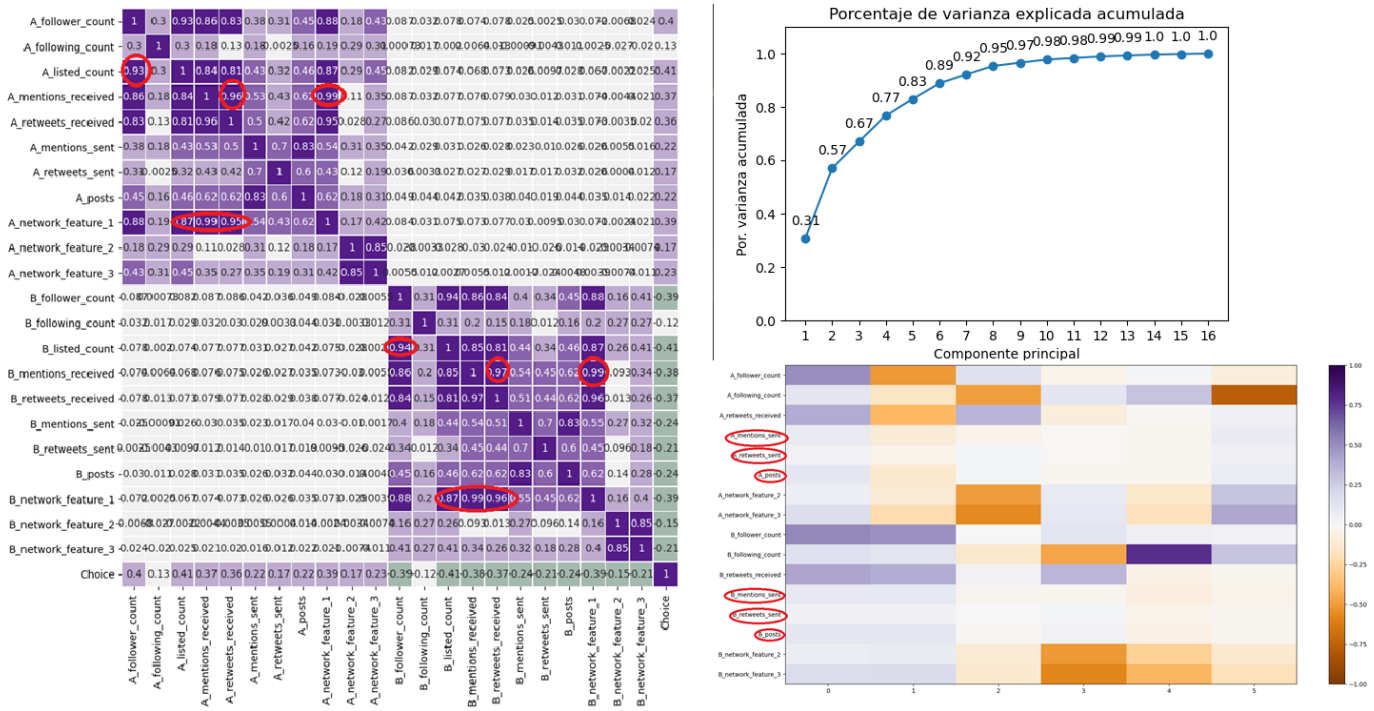
$$transf(x) \rightarrow log(x+1)$$



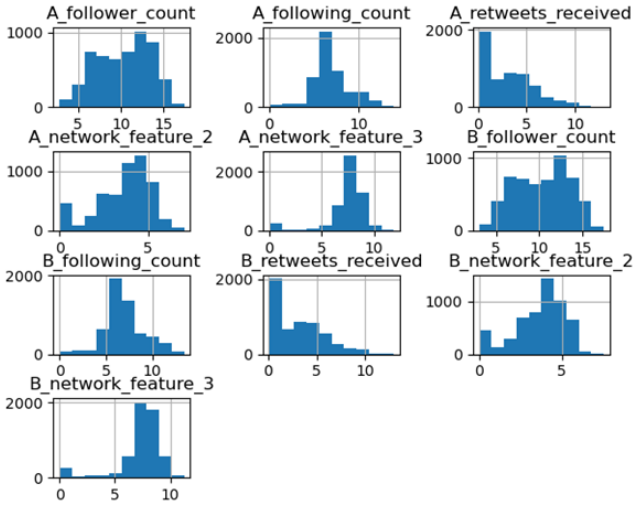
Decido no realizar una limpieza de los puntos outliers por los siguientes motivos, en primer lugar uno de los métodos de clasificación de la solución es un árbol de decisión (robusto ante outliers), realizaremos una regresión que efectivamente no es tan robusta ante outliers, sin embargo vamos a usar un método mas adelante que va a paliar este problema y por último, este trabajon data de una red social en la que nos encontraremos outliers con bastante frecuencia de los usuarios mas 'influencers' que queremos clasificar y tener en cuenta.

3. Analisis dimensional

Procedemos a evaluar cuantas de nuestras variables son útiles en nuestro modelo, para ello realizaremos un análisis de correlación lineal (no queremos variables fuertemente relacionadas de forma lineal), eliminaremos las variables A network feature, A counted list, A mentions recived, B network feature, B counted list, B mentions recived. A continuación realizaremos un análisis por componentes principales donde evaluaremos si existen algunas variables que no son importantes en cuanto a variabilidad se refiere, concluimos escogiendo 6 componentes que las variables A mentions, A retweets sent, A posts, B mentions, B retweets sent, B posts son prescindibles y decidimos eliminarlas:



Por tanto nuestro dataset al final de todo el proceso de depuración tendrá la siguiente forma:



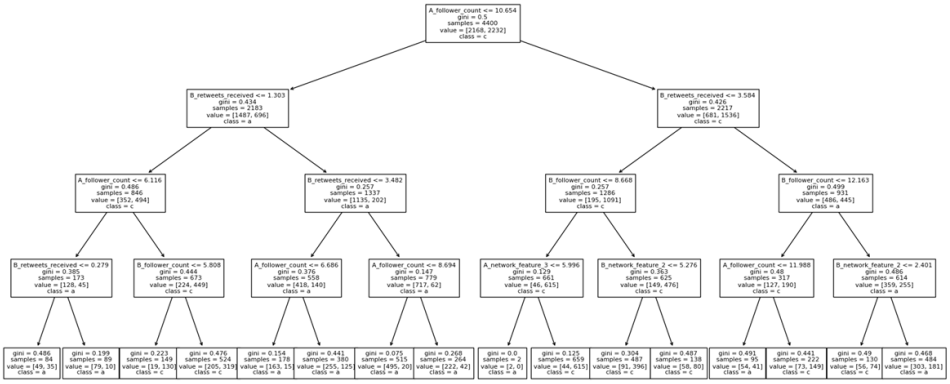
4. Modelos de predicción

Para realizar los modelos vamos a separar nuestra muestra en entreno y validación usando la herramienta de Sklearn. En primer lugar realizaremos un modelo de regresión lineal, el modelo predice de una manera aceptable sin realizarle ningún cambio al predeterminado de Sklearn, solo añadimos una cross validation de 10 muestras para saber si el modelo es estable con la muestra de entrenamiento, además de el podemos observar que las variables con mayor poder de decisión son A follower Count, A network feature 2, B follower Count, B network feature 2:

$$Log(\frac{\pi}{1-\pi}) = 0{,}344325(Afollowercount) - 0{,}005928(Afollowingcount) + 0{,}165128(Aretweetsreceived) + \\ + 0{,}399651(Anetworkfeature2) - 0{,}165024(Anetworkfeature3) - 0{,}274651(Bfollowercount) + \\ + 0{,}006213(Bfollowingcount) - 0{,}235519(Bretweetsreceived) - 0{,}378489(Bnetworkfeature2) + \\ + 0{,}131130(Bnetworkfeature3) - 0{,}2999178$$

Private Score	Public Score
0.80358	0.80263

Probamos con un árbol de decisión predeterminado iterando sobre la profundidad del arbol para conseguir el mejor acierto sin sobre ajuste, los resultados predictivos son parecidos salvo que este método se decanta por las variables A follower Count, A retweets received, B follower Count, B retweets received:

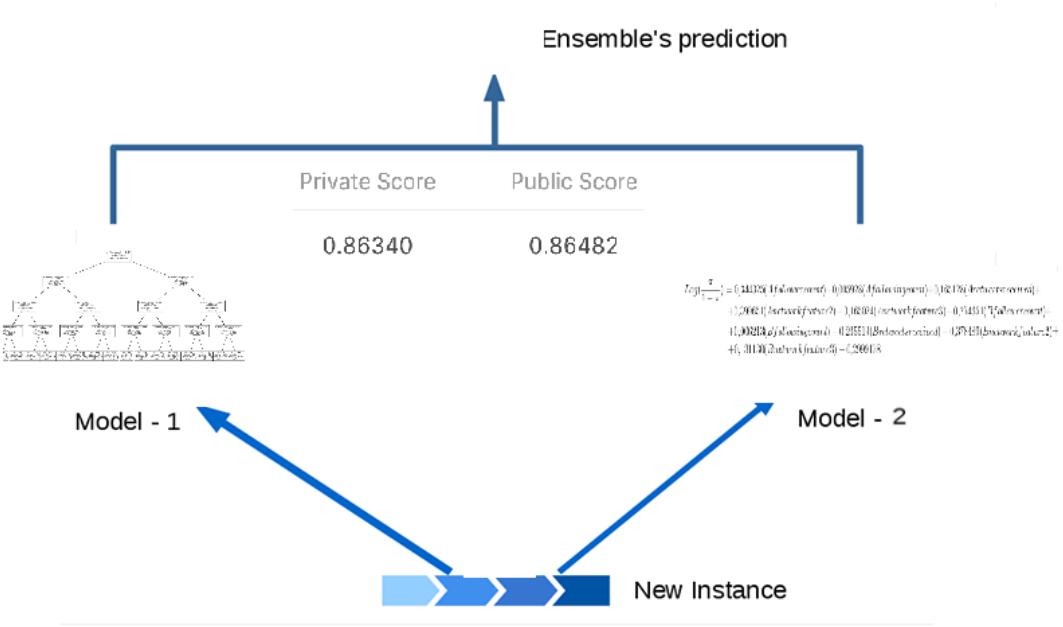


	Score	Feature
0	0.3005	A_follower_count
1	0.2495	B_retweets_received
2	0.1668	B_follower_count
3	0.0980	A_retweets_received
4	0.0393	B_network_feature_2
5	0.0388	B_following_count
6	0.0351	A_network_feature_2
7	0.0295	A_following_count
8	0.0282	A_network_feature_3
9	0.0144	B_network_feature_3

Private Score	Public Score
0.81358	0.81263

Tenemos dos métodos de predicción que juntos no se acercan a lo que desearíamos (aunque no distan demasiado), para ganar un poco de potencia predictiva y por tanto mejor puntuación pública, decido mejorar los dos modelos, primero la regresión logística realizando un GridSearch sobre el método de penalty y sobre el parámetro C, además vamos a compensar los outliers que puedan alejar la linea de regresión haciendo un Ada Boost de Sklearn que cambia los pesos de los puntos que clasifica mal para darle una mayor importancia. Además al arbol vamos a hacerle un gradient boosting classifier para conseguir el árbol óptimo en función del error.

A estos dos modelos mejorados los vamos a unir con un ensemble que lo que va a hacer es que a la probabilidad que ambos modelos le den a un punto de pertenecer a una clase de la solución, este le va a dar la media. Con esto ganamos bastante precisión que los modelos por separado y este constituye nuestro modelo final que tiene en cuenta las variables mas importantes de los dos modelos anteriores:

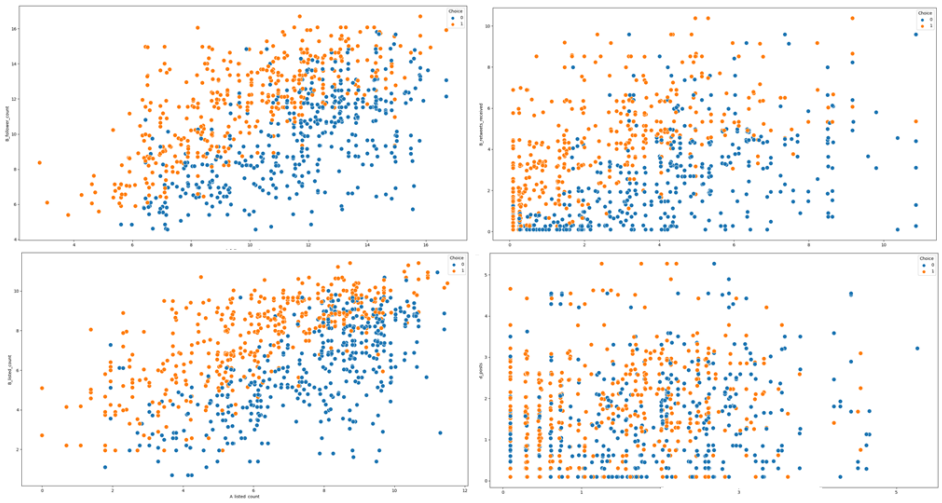


5. Conclusiones:

Hemos realizado un modelo que con mayor o menor acierto vaticina que elemento de la red neuronal tiene mayor influencia entre dos dados, las variables con mayor influencia han sido el numero de seguidores, la característica 2 de la red y los retweets recibidos por la cuenta, siendo estas las caráterísticas mas importantes a la hora de buscar personas influyentes en la red.

6. Análisis del error

En vista que nuestro mejor modelo no es capaz de llegar a una potencia de acierto ni siquiera del 0.9 y que el mejor modelo de Kaggle se queda en torno al 0.88, planteamos dos hipótesis de lo que está sucediendo. La primera que necesitamos mas variables para acabar de predecir de manera fiable lo que sucede en la red de twitter. La segunda que tenemos un lastre en nuestra variable target 'Choice', me explico, la variable target ha sido elegida por el voto de personas evaluando quien a su juicio era una persona mas influyente, pudiendo darse el caso de personas influyentes en un ámbito alejado de la red social y que sin embargo no usen la red con asiduidad o que quizás no les interese. Para ello queremos representar los puntos que erramos en la representación de nuestro data de entrenamiento por ejemplo:



Como podemos observar en estos ejemplos, sobre el eje Y están las variables del punto B y sobre el eje X las del punto A, el color naranja significa A mas influyente y azúl el suceso contrario. Podemos encontrarnos con individuos con menos actividad en la red social, menos seguidores, menos retweets e interacción con la gente que el resto etc etc, esto sucede con todas las variables. Dejo a revisión del proyecto de Kaggle si es posible que esto pueda suceder.