Predicción de eclipses asistida por Machine Learning

Juan Sebastián Martínez Arévalo[1], Emmanuel Arias Polanco[1], and Juan Gabriel Suárez Cadena[1]

[1]Departamento de Física, Universidad Nacional de Colombia, Bogotá D.C., Colombia

***Resumen*—Reportamos el proceso de implementación de un algoritmo de Machine Learning (ML) para la determinación de eclipses solares o lunares, dada la separación aparente del sol y la luna en el cielo usando los modelos VSOP87 y ELP-2000/82 implementados con la librería `PyEphem`. Se reporta el rendimiento de los algoritmos K-Nearest Neighbors (K-NN), Logistic Regression (LR), Random Forest Classifier (RFC), y Multi-Layer Perceptron Classifier (MLP). Se obtuvo un rendimiento ligeramente superior con el método K-NN (F1 score = 0.93717), considerando además el sesgo de muestras que presenta el problema, donde solo el 0.6 % del total de eventos, corresponden a eventos verdaderos (tránsitos identificados como eclipses según los modelos VSOP87 y ELP-2000/82); en este contexto, el algoritmo K-NN obtuvo el menor número de falsos positivos/negativos respecto a los otros métodos. Se discute la posibilidad de mejora del algoritmo al entrenar un modelo ML que aprenda a determinar separaciones en el cielo usando los datos utilizados en el presente trabajo.**

1. Introducción

Desde los inicios de nuestra historia como forma de vida más primitiva, convivimos con este fascinante y, en principio, extraño fenómeno que, a lo largo de la historia, ha dado lugar a distintas interpretaciones según la cultura y mitología desde la que se interpretase. Hoy, entendemos su naturaleza desde la ciencia, y los usamos para la observación astronómica, desde el simple hobby hasta la investigación científica, como se hizo en 1919, cuando se sometió a prueba por primera vez la Teoría de la Relatividad General de Einstein, aprovechando la intervención de un eclipse solar para comprobar las predicciones hechas por la relatividad general sobre la curvatura de la luz debida a objetos masivos.

Es por esto que, históricamente, se ha tenido interés por predecir estos eventos. Hoy, con la teoría física de la que disponemos, somos capaces de predecir -aunque no analíticamente- la ocurrencia de este y otros fenómenos relacionados con las dinámicas debidas a la interacción gravitacional.

Por otro lado, el actual desarrollo de las técnicas computaciones, y especialmente el Aprendizaje Automático, o Machine Learning (ML), ha demostrado un gran poder y la existencia gran aliado a la hora de realizar cálculos pesados computacionalmente, en tiempos racionales, como resulta de gran interés en ciertas ramas de la computación, tales como los gráficos generados por computadora.

Dada la importancia de esta rama, en su artículo, Frincu y Sferdian, 2021 discuten sobre la generación de un conjunto de datos de entrenamiento para un algoritmo de Machine Learning, a partir de datos de la Administración Nacional de Aeronáutica y el Espacio (NASA) y de datos meteorológicos de tránsito solar y lunar, comprendiendo inicialmente un conjunto de datos que va desde el 1999 a.C, hasta el 2100 de nuestros tiempos. Con estos datos, y el uso de la librería `PyEphem`, se crea un nuevo conjunto de datos que comprende los valores diarios, desde el primero de enero de 1950, hasta el 31 de diciembre de 2100, de la mejor separación sol-luna en el cielo terrestre para la identificación de eclipses. En el artículo original, los datos entre 1950/01/01 y 2019/12/31 (YYYY/MM/DD), son usados como datos de entrenamiento, y el resto del conjunto corresponderá a datos de prueba.

Una vez obtenido este conjunto de datos, el problema de predicción se reduce a la determinación de un valor mínimo como criterio de descarte para la clasificación de los eventos dada la separación calculada. La correcta determinación de este parámetro de criterio, resulta determinante a la hora de identificar los eclipses, en especial los anulares, y, en general, eclipses lunares, donde la sombra sobre la luna está mucho más dispersa. Es por esto que este parámetro debe ser elegido en base a un análisis de todo el conjunto de prueba, y es aquí donde emerge la posibilidad de un algoritmo de Machine Learning que determine internamente este criterio de clasificación.

2. Métodos

Los datos y métodos utilizados para resolver el problema son tomados del trabajo de Frincu y Sferdian, 2021. Los datos de separación son determinados al calcular, para un determinado día, la mejor separación sol-luna (según cada caso; para un eclipse solar, donde los astros están cercanos en el cielo; o un eclipse lunar, donde están en posiciones opuestas) obtenida mediante datos meteorológicos de tránsito solar y lunar en el cielo terrestre (http://www.timeanddate.com/eclipse/list.html). El cálculo de estos datos está basado en los modelos VSOP87 y ELP-2000/82 implementados en la librería `PyEphem`.

El problema de clasificación puede ser abordado por distintos enfoques, expresado en forma de distintos métodos de implementación, como se describirá posteriormente. Primero, veamos que la distribución del conjunto de datos (figura 2.1), está fuertemente representada por los eventos falsos (no eclipses), los cuales representan aproximadamente el 99.4 % del total en cada caso. Es por esto que nuestro problema puede presentar cierto sesgo a la hora de analizar las métricas de rendimiento, dado que, con la distribución que se tiene para los eventos en el conjunto de entrada, los errores al identificar eventos verdaderos (eclipses) no aportan mucho a las métricas de rendimiento general, porque este subconjunto es, a su vez, poco representativo respecto al total. Luego, las métricas que se usen estarán, en general, fuertemente representadas por el rendimiento del modelo al identificar correctamente eventos falsos, lo cual no representa un punto de interés, como sí la identificación de eventos verdaderos. Este efecto se podrá notar también cuando se presenten los resultados en las matrices de confusión.

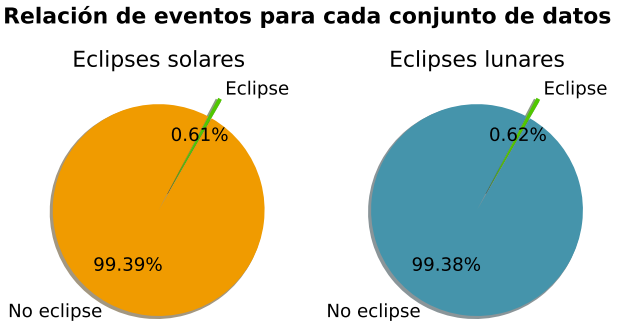


Figura 2.1: Relación entre eventos verdaderos (eclipses) y falsos (no eclipses) para los distintos conjuntos de datos (eclipses solares y eclipses lunares)

Para lidiar con este problema, se hace uso del algoritmo de *N vecinos más cercanos*, o *k-Nearest Neighbors (k-NN)*. Este algoritmo es un clasificador supervisado que, dado un dato de entrada, determina la salida más probable según el número de datos que rodeen al dato de entrada. En nuestro caso, dado un valor de separación, el algoritmo determina la etiquetas con las que se identificaron los 7 vecinos más cercanos (los 7 valores de separación más parecidos al valor de entrada) durante el entrenamiento del modelo. Este modelo representa una forma de lidiar con el problema de representación del conjunto de datos verdaderos respecto al total de eventos.

Para comparar el rendimiento y el porcentaje de aciertos respecto a otros modelos, haremos también la implementación de modelos basados en Regresión Logística (Logistic Regression), Bosques Aleatorios (Random Forest), y un modelo sencillo basado en Redes Neuronales con múltiples capas de neuronas (Multi-Layer Perceptron).

De estos métodos, dado que los algoritmos de Regresión Logística (LR), y Bosque Aleatorio (RF), tienen un funcionamiento en gran medida compatible con la naturaleza del problema, en donde se determina de forma iterativa un criterio de selección para los eventos, es de esperarse también un buen rendimiento en estos casos.

3. Resultados

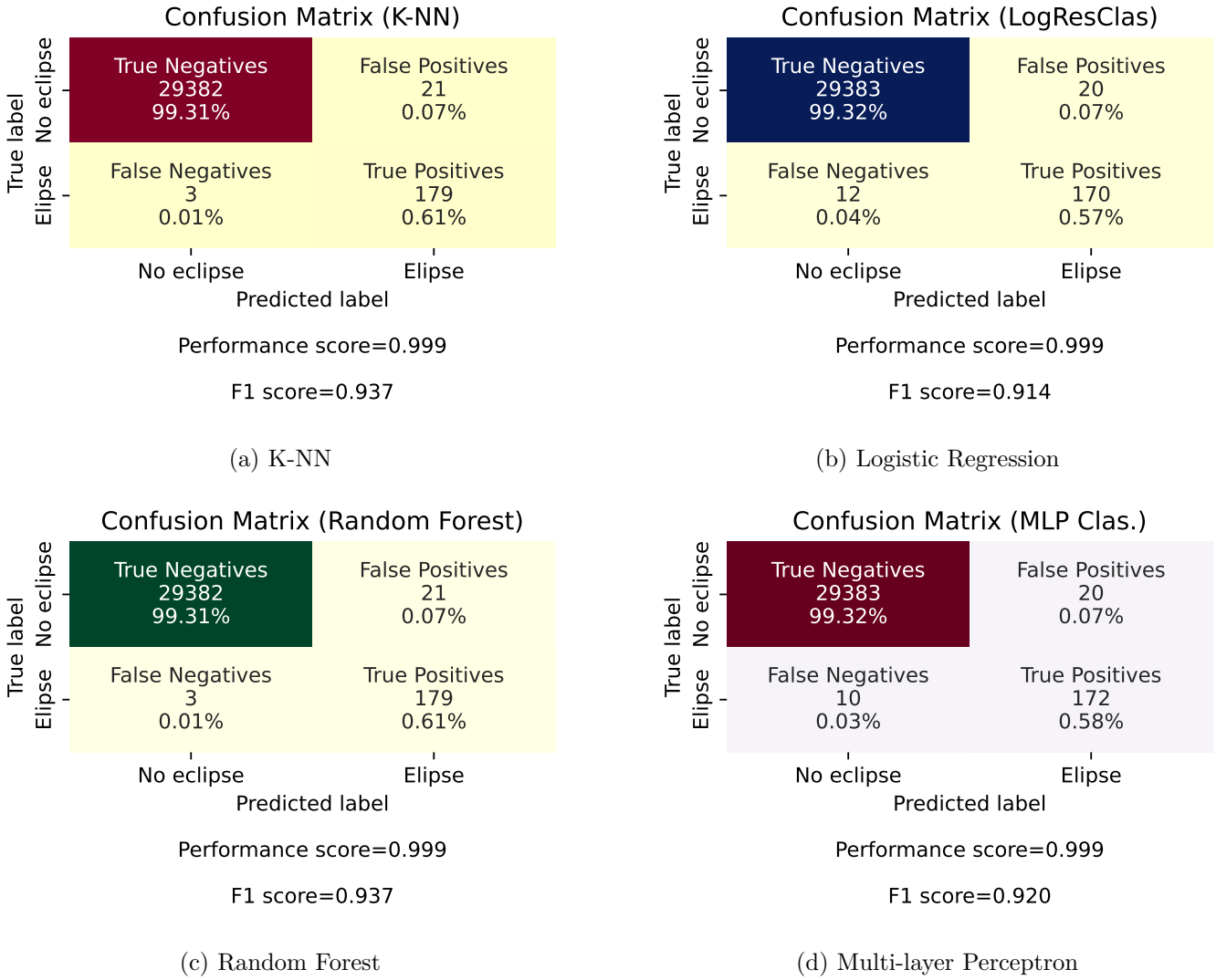
Usando los métodos `KNeighborsClassifier`, `LogisticRegression`, `RandomForestClassifier` y `MLPClassifier`, de la librería `scikit-learn` de Python, se realizó la implementación de los siguientes modelos con los parámetros especificados en la tabla 3.1 [1].

Para la discusión inicial, hacemos el entrenamiento con solo eclipses solares; posteriormente, veremos el rendimiento de estos mismos modelos al predecir eclipses solares y lunares. Con esto, nuestro conjunto final de datos (`solar-eclipses-classif.csv`), consiste de un conjunto de 55152 muestras, de las cuales 337 corresponden a registros de eclipses solares, y las restantes 54815 muestras corresponden a eventos de no eclipses. Estos datos corresponden a los registros diarios de separación sol-luna en el cielo terrestre desde 1950/01/01 hasta 2100/12/31.

[1] dado que para el modelo de Regresión Logística se usaron los parámetros por defecto en la librería, no se especifican estos en la tabla

Tabla 3.1: Parámetros utilizados para el entrenamiento de cada modelo

K-NN		Random Forest		MLP	
Parámetro	**Valor**	**Parámetro**	**Valor**	**Parámetro**	**Valor**
leaf_size	30	n_estimators	50	hidden_layer_sizes	(5,8,8)
metric	minkowski	max_depth	2	activation	relu
n_neighbors	7	min_samples_split	30	solver	adam
p	2	max_features	None	max_iter	500
weights	uniform	NA	NA	NA	NA



(a) K-NN

(b) Logistic Regression

(c) Random Forest

(d) Multi-layer Perceptron

Figura 3.2: Matrices de confusión para los cuatro modelos implementados en cada caso

Estos datos, como se describe en el artículo original, son separados de la siguiente manera: los datos hasta 2019/12/31 corresponden a datos de entrenamiento, y los datos de esta fecha en adelante (hasta 2100/12/31), corresponden a datos de prueba. En la figura 3.2 se encuentran las matrices de confusión en cada caso luego de entrenar los modelos con este conjunto de datos. En esta figura se puede apreciar el comportamiento mencionado previamente, donde la precisión de cada método está fuertemente ponderada por la cantidad de verdaderos negativos (TN) (los no eclipses en nuestro caso).

Al analizar las matrices de confusión en 3.2, vemos que todos los modelos se comportaron bien, dentro

de los estándares, al predecir los eclipses. Podemos notar que los modelos basados en K-NN y Random Forest (3.2a y 3.2c), arrojaron exactamente los mismos resultados con las configuraciones especificadas. Como mencionamos anteriormente, esperábamos un buen comportamiento de estos modelos, dadas la distribución de datos, en el caso K-NN, y la naturaleza del problema, para el caso RF.

En la tabla 3.2, podemos ver el porcentaje de eclipses reportados como eclipses (TP), que logró cada modelo al hacer predicciones con el conjunto de prueba (en total, en el conjunto de prueba hay 182 eventos reportados como eclipses, y 29403 eventos reportados como no eclipses)

Tabla 3.2: Porcentaje de eclipses solares reportados como eclipses solares por cada algoritmo, respecto al total disponible en el conjunto de prueba

Modelo	**Aciertos**(%)
K-NN	98.4 %
LR	93.4 %
RF	98.1 %
MLP	94.5 %

Con la tabla 3.2 observamos directamente qué tanto resultaron mejores los modelos $K-NN$ y RF para el problema de clasificación. Con estos modelos entrenados, y ayudándonos de la función `get_separation()`, podemos hacer predicciones de eclipses dada una fecha.

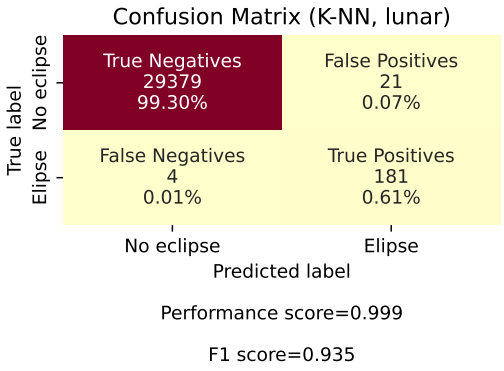
Tabla 3.3: Predicciones hechas por los distintos modelos para un conjunto de fechas dadas. Se muestra la evolución de la identificación para día antes y después de la fecha objetivo. En la tabla, 1 indica en ese día ocurrirá un eclipse, y 0 que no ocurrirá.

Fecha	**Predicción**			
	K-NN	**LR**	**RF**	**MLP**
2023/10/13	0	0	0	0
2023/10/14	1	1	1	1
2023/10/15	0	0	0	0
2024/04/07	0	0	0	0
2024/04/08	1	1	1	1
2024/04/09	0	0	0	0
2036/07/22	0	0	0	0
2036/07/23	1	0	1	0
2036/07/24	0	0	0	0

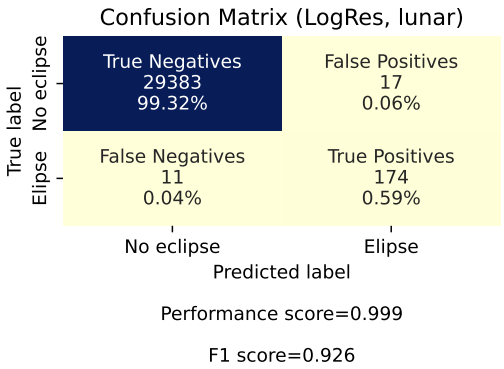
El modelo puede recibir fechas en forma de cadenas de texto al llamar el método `predict` como

```
model_name.predict([[
    get_minimum_separation(
        'YYYY-MM-DD'
        )
    ]])
```

Con la función `get_separation()` o `get_separation_lunar()` según corresponda. Para el caso de eclipses lunares, estos métodos de predicción, entrenados con los datos de eclipses solares, resultan igual de útiles a la hora de predecir eclipses lunares; esto se debe a que cualquiera de las funciones `get_separation()` o `get_separation_lunar()`, devuelven un valor de separación que representa un mínimo, calculado de forma distinta. Este comportamiento nos confirma que el poder del algoritmo reside en clasificar los eventos dada la separación adecuada. En la figura 3.3 se muestra la matriz de confusión para algoritmos basados en KNN y LogisticRegression, sin re entrenar y evaluando con los datos de prueba para eclipses lunares.



(a) K-NN



(b) Logistic Regression

Figura 3.3: Matrices de confusión para conjunto de datos con eclipses lunares (no re-entrenamiento)

En la figura 3.3, podemos apreciar nuevamente que

el modelo basado en K-NN se comporta mejor a la hora de identificar falsos positivos (FP) y falsos negativos (FN), respecto al modelo de regresión logística.

El total de eclipses en el conjunto empleado para las matrices en la figura 3.3, es de 185 eclipses lunares. La tabla 3.4 muestra el comportamiento de los modelos para una fecha específica (esta fecha fue escogida de forma aleatoria y no representa ningún valor interesante más que uno para mostrar el comportamiento de los modelos).

Tabla 3.4: Predicción de eclipse lunar hecha por los modelos para días alrededor de una fecha específica

Fecha	**Predicción**	
	K-NN	**LR**
2023/05/04	0	0
2023/05/05	1	1
2023/05/06	0	0

4. Conclusiones

El algoritmo de K vecinos más próximos (K-NN), mostró, al igual que el algoritmo de bosques aleatorios, un muy buen desempeño a la hora de predecir eclipses dado el valor de la separación sol-luna para ambos conjuntos de datos: eclipses solares y eclipses lunares. Al mismo tiempo, ambos algoritmos pudieron lidiar bien con el problema de representación en el conjunto de datos de entrenamiento; esta mejora puede ser observada en las figuras 3.2 y las tablas 3.2 y 3.3, respecto a los demás algoritmos implementados (regresión logística y multi-layer perceptron classifier).

El modelo implementado, como se aprecia en la prueba con datos de eclipses lunares en modelos entrenados con eclipses solares, muestra una fuerte dependencia del valor de separación más que en la detección de patrones en el tiempo. Es por esto que este modelo de predicción basado en predicción mediante el ingreso de la separación sol-luna, aunque presenta una ayuda al determinar un criterio interno basado en todos los datos de entrenamiento para decidir si existirá o no un eclipse; consideramos, está aún incompleto, en cuanto que su verdadero potencial está en encontrar patrones en las fechas ingresadas para predecir la ocurrencia de eclipses. Esta posibilidad representaría una implementación implícita de los modelos VSOP87 y ELP-2000/82 mediante un algoritmo de Machine Learning, además de un algoritmo completo de predicción de la separación sol-luna que podría funcionar en conjunción con el algoritmo actualmente implementado, y representaría a su vez una mejora computacional, dados los tiempo de cómputo necesarios para obtener los datos sintéticos en el intervalo deseado (aproximadamente 40 minutos).

Para efectos del trabajo actual, y el trabajo desarrollado en Frincu y Sferdian, 2021, se muestra el correcto funcionamiento del modelo para su propósito inicial, sirviendo de punto de partida la posterior implementación de un algoritmo completo de predicción basado en fechas. Las implementaciones hechas en el presente trabajo pueden ser encontradas en el repositorio “Eclipse-Prediction-by-ML”, 2023.

Referencias

Frincu, M., & Sferdian, M. (2021). When old meets new: evaluating numerical and Machine Learning based eclipse prediction methods. *Romanian Astronomical Journal*, *31*(2), 133-152. http://irep.ntu.ac.uk/id/eprint/44130/

Eclipse-Prediction-by-ML. (2023). https://github.com/CosmeticMichu/Eclipse-Prediction-by-ML