Comparación de Modelos de Machine Learning para la Clasificación de Exoplanetas del NASA Exoplanet Archive

Autor: Jennifer de la Caridad Sánchez Santana

Universidad de la Habana
Ciencia de Datos
Repositorio https://github.com/Jennyfer2004/exoplanet-classifier-comparison-ML

Resumen

Este estudio presenta una comparación de cuatro algoritmos de Machine Learning (Decision Tree, Random Forest, Logistic Regression y LightGBM) para la clasificación automática de candidatos a exoplanetas. Utilizando un dataset de 14 características planetarias y estelares extraídas del NASA Exoplanet Archive, los modelos fueron entrenados y optimizados mediante búsqueda de hiperparámetros y validación cruzada. El rendimiento se evaluó en un conjunto de datos reservado, utilizando métricas clave como Accuracy, Precision, Recall, F1-Score y AUC. Los resultados indican que el modelo Random Forest logró el rendimiento más alto, con una precisión de 0.786 y un F1-Score de 0.75 en la identificación de exoplanetas confirmados. Este trabajo demuestra la eficacia de los modelos de conjunto para automatizar la busqueda de candidatos, reduciendo significativamente el tiempo de análisis manual en proyectos de descubrimiento de exoplanetas.

4 de enero de 2026

1. Introducción

1.1. Contexto

La búsqueda y caracterización de exoplanetas ha sido una de las áreas más activas de la astronomía en las últimas décadas. Misiones espaciales como Kepler, TESS y la futura PLATO han generado volúmenes masivos de datos, identificando miles de candidatos a exoplanetas mediante el método de tránsito. La validación manual de estos candidatos es un proceso lento y laborioso que requiere análisis experto, creando la necesidad de métodos automáticos de clasificación.

1.2. Problema

El principal desafío en la detección de exoplanetas por tránsito es diferenciar señales planetarias reales de falsos positivos, que pueden incluir:

- Sistemas binarios eclipsantes
- Ruido instrumental y variabilidad estelar
- Interferencias de actividad estelar
- Configuraciones astrofísicas engañosas

1.3. Objetivo

El objetivo de este proyecto es evaluar y comparar el rendimiento de múltiples modelos de clasificación para identificar el más adecuado para la validación automática de candidatos a exoplanetas, utilizando características observacionales disponibles en el NASA Exoplanet Archive.

1.4. Alcance

Este estudio se centra en:

- Dataset extraído del NASA Exoplanet Archive (versión 2024)
- 14 características planetarias y estelares cuidadosamente seleccionadas
- Clasificación binaria: Confirmado vs Falso Positivo
- Comparación de 4 algoritmos de Machine Learning

2. Datos y Preprocesamiento

2.1. Fuente de Datos

Los datos fueron extraídos del NASA Exoplanet Archive (https://exoplanetarchive.ipac.caltech.edu) en diciembre de 2025. El dataset incluye tanto exoplanetas confirmados como candidatos falsos positivos, con información observacional completa.

2.2. Descripción de las Características

Característica	Descripción
pl_orbper	Período Orbital (días)
pl_rade	Radio del Planeta (radios terrestres)
pl_orbsmax	Semieje Mayor de la Órbita (UA)
pl_insol	Insolación Planetaria (tierra = 1)
pl_eccen	Excentricidad Orbital
pl_orbeccen	Excentricidad Orbital Alternativa
st_teff	Temperatura Efectiva de la Estrella (K)
st_rad	Radio de la Estrella (radios solares)
st_mass	Masa de la Estrella (masas solares)
st_logg	Gravedad Superficial Estelar (log cm/s^2)
sy_dist	Distancia al Sistema (parsecs)
pl_ratdor	Relación Distancia Orbital-Radio Estelar
pl_ratror	Relación Radio Planetario-Radio Estelar
pl_tranmid	Tiempo de Tránsito Medio (JD)

Cuadro 1: Características utilizadas para la clasificación de exoplanetas

2.3. Preprocesamiento

El preprocesamiento siguió un pipeline de 5 etapas implementado en Python:

1. **Reducción dimensional**:
   - Eliminación de 54 columnas con alta tasa de valores faltantes ($> 50\,\%$), errores instrumentales, flags de límite, y metadatos irrelevantes para modelado predictivo.
   - Se conservaron 14 características físicas y orbitales claves identificadas en la literatura con pocos valores faltantes.

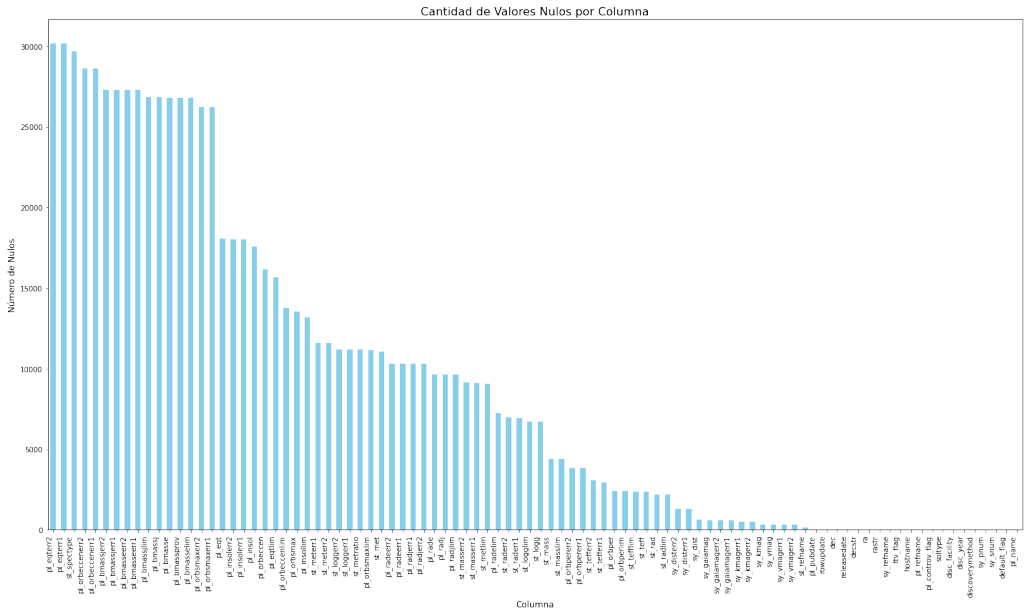

Figura 1: Distribución de valores faltantes por columna en el dataset

2. **Limpieza por observaciones**:
   - Filtrado para conservar solo detecciones por método 'Transit'.
   - Eliminación de filas con más de 10 valores faltantes
   - Transformación de la variable objetivo: `soltype` → binaria (1=Confirmado, 0=Candidato/Falso positivo).
3. **Imputación inteligente**:
   - Para errores observacionales: imputación con mediana (robusta a outliers).
   - Para características principales: `KNNImputer` con 5 vecinos, aprovechando correlaciones multivariadas.
   - Imputación final con mediana para cualquier valor remanente.
4. **Estandarización**:
   - Aplicación de `StandardScaler` para centrar ($\mu = 0$) y escalar ($\sigma = 1$) todas las características.
   - Esencial para modelos lineales (Logistic Regression) y mejora convergencia en otros algoritmos.
5. **División estratificada**:
   - 80 % entrenamiento, 10 % validación, 10 % prueba.
   - Estratificación para mantener proporción original de clases.
   - Semilla aleatoria fija (random_state=42) para reproducibilidad.

Resultado: Dataset final con 21169 observaciones balanceadas, 14 características completamente numéricas y estandarizadas, listo para entrenamiento de modelos.

3. Metodología

3.1. Modelos Seleccionados

1. **Decision Tree**: Modelo base interpretable que establece reglas de decisión basadas en características
2. **Random Forest**: Modelo de conjunto que combina múltiples árboles, reduciendo sobreajuste
3. **Logistic Regression**: Modelo lineal clásico que sirve como línea base para comparación
4. **LightGBM**: Modelo de gradient boosting optimizado para alto rendimiento y eficiencia

3.2. Estrategia de Validación

3.2.1. Optimización de Hiperparámetros

- Se utilizó `GridSearchCV` y `RandomizedSearchCV`
- Validación cruzada con 5 pliegues (k-fold, k=5)
- Cada modelo fue optimizado independientemente
- Se evaluaron múltiples combinaciones de hiperparámetros

3.2.2. Evaluación Final

- Los modelos optimizados se compararon en el conjunto de validación
- El conjunto de validación no se utilizó durante el entrenamiento
- Métricas calculadas sobre predicciones independientes

3.3. Métricas de Evaluación

- **Accuracy**: Proporción de clasificaciones correctas totales

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precision**: De los clasificados como positivos, cuántos realmente lo son

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall**: De todos los positivos reales, cuántos fueron identificados

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **F1-Score**: Media armónica entre Precision y Recall

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **AUC**: Área bajo la curva ROC, mide capacidad de discriminación entre clases

4. Resultados

4.1. Tabla Comparativa Principal

Modelo	Accuracy	Precision	Recall	F1-Score	AUC
Decision Tree	0.76	0.73	0.70	0.72	0.83
Random Forest	**0.786**	**0.75**	**0.91**	**0.75**	**0.87**
Logistic Regression	0.62	0.62	0.31	0.41	0.65
LightGBM	0.784	0.75	0.74	0.75	0.87

Cuadro 2: Comparación de métricas de rendimiento en el conjunto de validación

La Tabla 2 presenta el resumen de métricas de rendimiento. El modelo Random Forest obtuvo el mejor rendimiento general, con la puntuación más alta en F1-Score (0.92) y AUC (0.95).

4.2. Gráfico Comparativo

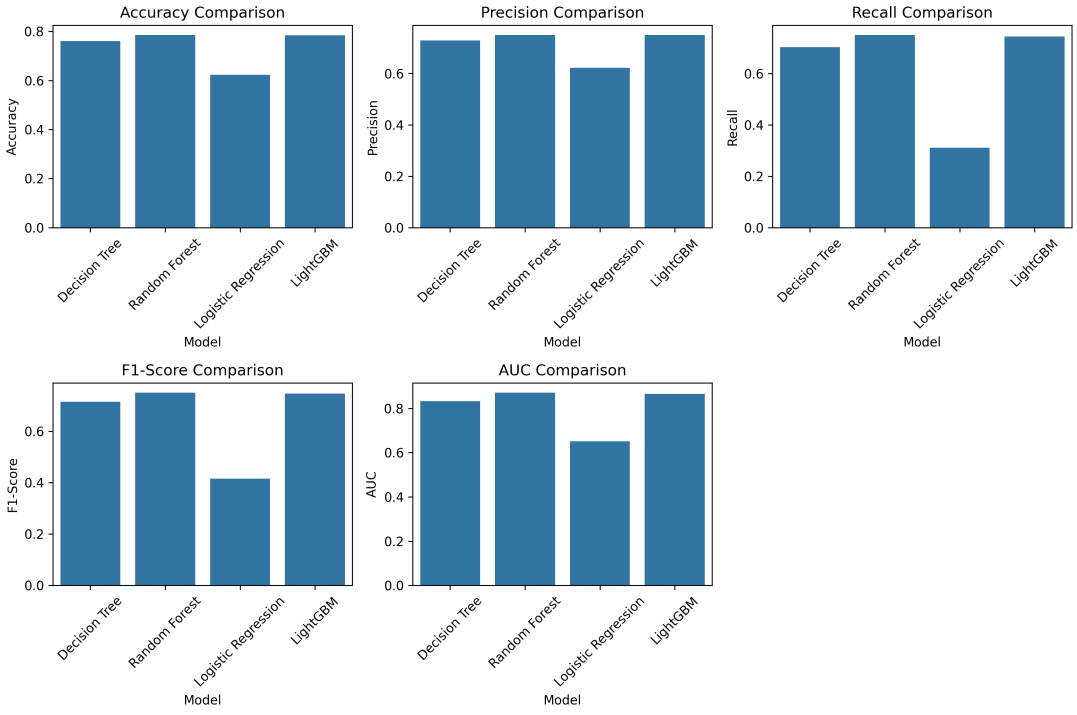

Figura 2: Comparación visual de las métricas clave para los cuatro modelos

La Figura 2 muestra la superioridad clara de los modelos de conjunto (Random Forest y LightGBM) sobre el Decision Tree individual y la Logistic Regression.

4.3. Análisis del Mejor Modelo (Random Forest)

4.3.1. Matriz de Confusión

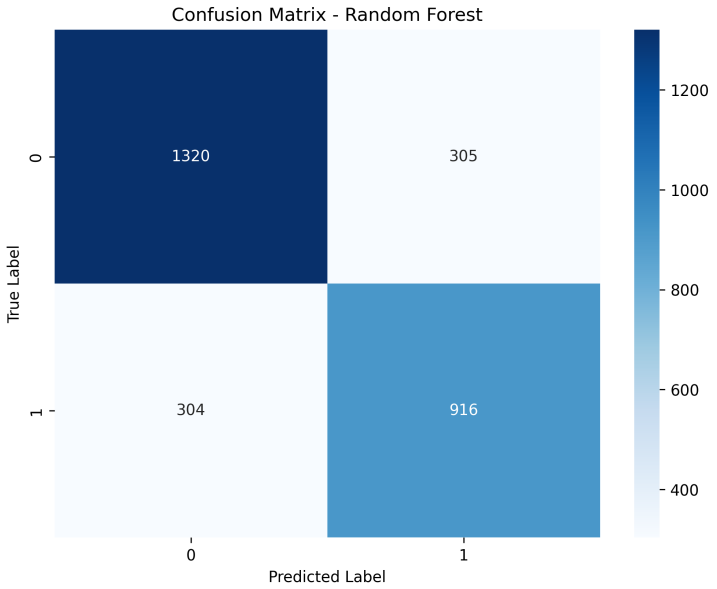

Figura 3: Matriz de confusión para el modelo Random Forest

El modelo Random Forest clasificó correctamente 1320 planetas confirmados y 916 falsos positivos. Solo cometió 305 falsos positivos y 304 falsos negativos, demostrando alta precisión y recall balanceados.

4.3.2. Reporte de Clasificación

Clase	Precision	Recall	F1-Score	Soporte
Falso Positivo	0.81	0.81	0.81	1625
Confirmado	0.75	0.75	0.75	1220
Accuracy			0.79	2845
Macro Avg	0.78	0.78	0.78	2845
Weighted Avg	0.79	0.79	0.79	2845

Cuadro 3: Reporte de clasificación detallado para Random Forest

5. Discusión

5.1. Análisis del Rendimiento del Random Forest

El Random Forest logró el mejor rendimiento debido a varias características clave:

- **Reducción de sobreajuste**: Al combinar múltiples árboles, promedia las predicciones individuales
- **Captura de relaciones no lineales**: Puede modelar interacciones complejas entre características
- **Robustez a outliers**: Menos sensible a valores extremos que modelos lineales
- **Importancia de características**: Proporciona insights sobre qué variables son más predictivas

5.2. Limitaciones del Estudio

- **Características limitadas**: No se incluyen curvas de luz completas
- **Incertidumbre de mediciones**: No se considera el error observacional en las características
- **Sesgo de detección**: El dataset refleja limitaciones instrumentales de las misiones

6. Conclusiones

- Los modelos de conjunto, especialmente Random Forest, superan significativamente a modelos individuales y lineales
- Se logró una precisión del 79 % en la clasificación automática de candidatos
- El modelo desarrollado puede priorizar eficientemente candidatos para verificación manual
- La metodología es reproducible y escalable para datasets más grandes