

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
VALPARAÍSO - CHILE



**“ANÁLISIS DE LA DETECCIÓN DE TRÁNSITO DE
EXOPLANETAS MEDIANTE MODELOS DE APRENDIZAJE
PROFUNDO SEGÚN CARACTERÍSTICAS DE LA ESTRELLA
ANFITRIONA”**

TANAY CAMILO GONZÁLEZ PÉREZ

**MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN INFORMÁTICA**

Profesor Guía: Dra. Carolina Saavedra
Profesor Correferente: Dr. Mauricio Araya

Agosto - 2026



CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): ☒ Memoria o trabajo de título ☐ Tesis de Postgrado

Título del trabajo: Análisis de la detección de tránsito de exoplanetas mediante modelos de aprendizaje profundo según características de la estrella anfitriona.

Nombre del candidato(a): Tanay Camilo González Pérez

Carrera / Grado: Ingeniería Civil Informática

Campus: Casa Central Valparaíso

Departamento: Departamento de Informática

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Dra. Carolina Saavedra Ruiz, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución.

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL (marcar una opción)

☒ El trabajo **NO contiene** información que amerite confidencialidad y puede ser publicado de inmediato en repositorio con acceso abierto.

☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (**embargo**) por (**marcar una opción**):

☐ 6 meses ☐ 12 meses ☐ 2 años ☐ 3 años ☐ 5 años ☐ 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 27/07/2026

Firma: 

Estudiante o Candidato(a):

Fecha: 27/07/2026

Firma: 

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

AGRADECIMIENTOS

En esta etapa cúlmine de mi educación universitaria, quiero dedicar este espacio para dar las gracias.

En primer lugar, a mis padres Marly y Raúl, quienes me criaron con amor y se han esforzado durante años para que pueda perseguir mis sueños. A mi hermano Minoah, por darme ánimos cuando me sentí agobiado. A mis abuelos, Jaime, Perla y Sylvia, por entregarme su cariño.

A mis amistades que conservo desde hace años, Ricardo, Carla, Rock, Gabriela, María Paz, Esteban y Yessica, por cada risa y alegría que hemos compartido. También a mis compañeros de la universidad, Fabián, Diego, Martín y Aylin, con quienes he tenido el gusto de coincidir en este tramo formativo de nuestras vidas.

RESUMEN

Resumen—El método del tránsito es la técnica más popular para identificar exoplanetas, y en los últimos años se han producido avances en el uso de redes neuronales para la tarea de *vetting* (la discriminación entre verdaderos planetas y falsos positivos), alcanzando resultados notables. No obstante, la evaluación de estos modelos suele realizarse de manera global. Este trabajo propone comparar el rendimiento de diferentes modelos de aprendizaje profundo a lo largo de las distintas clases espectrales y etapas evolutivas. Los resultados revelan que el rendimiento predictivo está condicionado por la astrofísica de la estrella anfitriona. Mientras que las estrellas tipo G ofrecen un régimen óptimo de detección, la degradación de la relación señal-ruido en estrellas subgigantes y tipo K provoca caídas significativas en la sensibilidad y precisión.

Palabras Clave— Detección de exoplanetas; método del tránsito; parámetros estelares; vetting; aprendizaje profundo.

ABSTRACT

Abstract—The transit method is the most popular technique for identifying exoplanets, and in recent years there have been advancements in the use of neural networks for the task of vetting (distinguishing real planets from false positives), achieving remarkable results. Nevertheless, these models are usually evaluated in aggregate. This work proposes to compare the performance of various deep learning models across different spectral classes and evolutionary stages. Results reveal that predictive performance is conditioned by the host star's astrophysics. While G-type stars provide an optimal detection regime, the degradation of the signal-to-noise ratio in subgiants and K-type stars cause significant drops in recall and precision.

Keywords— Exoplanet detection; transit method; stellar parameters; vetting; deep learning.

GLOSARIO

AUC: Área Bajo la Curva (*Area Under Curve*).

CCD: *Charge-Coupled Device*.

CNN: Red Neuronal Convolucional (*Convolutional Neural Network*).

ED: *Encoder-Decoder*.

FCNN: Red Neuronal Totalmente Conectada (*Fully Connected Neural Network*).

GPU: Unidad de Procesamiento Gráfico (*Graphics Processing Unit*).

LSTM: *Long Short-Term Memory*.

MAST: Archivo Mikulski para Telescopios Espaciales (*Mikulski Archive for Space Telescopes*)

MC: Mapa de Características.

ROC: *Receiver Operating Characteristic*.

RNR: Red Neuronal Recurrente.

SNR: Razón Señal-Ruido (*Signal-to-Noise Ratio*).

TCE: Eventos que Cruzan el Umbral (*Threshold Crossing Event*).

TFP: Tasa de falsos positivos.

ÍNDICE DE CONTENIDOS

RESUMEN	iii
ABSTRACT	iii
INDICE DE FIGURAS	vii
INDICE DE TABLAS	ix
INTRODUCCIÓN	1
CAPÍTULO 1 DEFINICIÓN DEL PROBLEMA	2
1.1 Introducción	2
1.2 Descripción del problema	2
1.3 Objetivos	3
1.3.1 Objetivo general	3
1.3.2 Objetivos específicos	3
CAPÍTULO 2 MARCO CONCEPTUAL	4
2.1 El método del tránsito	4
2.2 Signal-to-Noise Ratio (SNR) en curvas de luz	5
2.3 Aprendizaje de máquina	7
2.4 Aprendizaje profundo	7
2.4.1 Funciones de activación	9
2.4.2 Funciones de pérdida	10
2.4.3 Entrenamiento de una red neuronal	10
2.4.4 Redes neuronales convolucionales	12
2.4.5 Redes Neuronales Recurrentes	13
2.4.6 Long Short-Term Memory	15
2.4.7 Arquitectura Encoder-Decoder	16
2.4.8 Mecanismos de atención	17
2.5 Métricas de evaluación	18
2.6 Fuentes conocidas de falsos positivos	21
2.7 Clasificación de Estrellas	22
2.7.1 Clases espectrales	22
2.7.2 Clases de luminosidad y etapa evolutiva	23
CAPÍTULO 3 PROPUESTA DE SOLUCIÓN	25
3.1 El conjunto de datos	25
3.1.1 Parámetros de tránsito y propiedades estelares	26
3.2 Preprocesamiento de los datos	26

3.2.1 Curvas de luz	27
3.2.2 Centroides	29
3.2.3 Parámetros estelares	30
3.3 Análisis exploratorio	30
3.4 Modelos de aprendizaje profundo	32
3.4.1 AstroNet	32
3.4.2 ExoNet	34
3.4.3 Modelo híbrido de CNN con LSTM y atención aditiva	36
3.4.4 Entrenamiento e hiperparámetros	38
CAPÍTULO 4 VALIDACIÓN DE LA SOLUCIÓN	39
4.1 Tiempos de entrenamiento e inferencia	39
4.2 Resultados generales	40
4.3 Resultados según clase espectral	41
4.3.1 Resultados en las clases espectrales F, G y K	41
4.3.2 Resultados en las clases espectrales B, A y M	44
4.4 Resultados según etapa evolutiva	45
CAPÍTULO 5 CONCLUSIONES	48
5.1 Conclusiones generales	48
5.2 Cumplimiento de objetivos	49
5.3 Trabajo futuro	49
REFERENCIAS BIBLIOGRÁFICAS	51
ANEXOS	55

ÍNDICE DE FIGURAS

1	Esquema de un planeta transitando frente a una estrella.....	5
2	Diagrama de una red neuronal artificial FCNN.....	8
3	Funciones de activación más populares en modelos de aprendizaje profundo	10
4	Ejemplo de la operación de convolución entre dos matrices	12
5	Esquema de la arquitectura AlexNet.....	13
6	Diagrama de una red recurrente de Elman	14
7	Diagrama de una célula de memoria en el modelo LSTM.....	16
8	Diagrama de la arquitectura <i>Encoder-Decoder</i>	17
9	Curva <i>Receiver Operating Characteristic</i>	21
10	Fotometría de un sistema de estrellas binarias eclipsantes	22
11	Primera etapa del preprocesamiento	28
12	Vistas global y local del TCE 9388479-1	29
13	Vistas global y local de las curvas de luz y centroides.....	30
14	Valores promedio para las vistas globales y locales.....	32
15	Diagrama de la arquitectura de AstroNet	33
16	Diagrama de la arquitectura de ExoNet	35
17	Diagrama de la arquitectura del modelo híbrido	37
18	F1-Score de cada modelo según la clase espectral de la estrella anfitriona.....	41
19	<i>Recall</i> de cada modelo según la clase espectral de la estrella anfitriona	42
20	Valores promedio y desviación estándar en las vistas globales para los vectores de desplazamiento de los centroides	43
21	Tasa de falsos positivos de cada modelo según la clase espectral de la estrella anfitriona.....	44
22	F1-Score de cada modelo según la etapa evolutiva de la estrella anfitriona	45
23	<i>Recall</i> de cada modelo según la etapa evolutiva de la estrella anfitriona.....	46
24	Tasa de falsos positivos de cada modelo según la etapa evolutiva de la estrella anfitriona.....	47

ÍNDICE DE TABLAS

1	Matriz de confusión	19
2	Clases espectrales principales del sistema de Harvard	23
3	Clases de luminosidad estelares y sus etapas evolutivas asociadas	24
4	Distribución de las estrellas y TCEs del conjunto por clase espectral	31
5	Distribución de las estrellas y TCEs del conjunto por etapa evolutiva	31
6	Distribución de TCEs en los conjuntos de entrenamiento, evaluación y pruebas	38
7	Resumen de hiperparámetros utilizados para entrenamiento	38
8	Número de parámetros y tiempos de entrenamiento y para cada modelo	39
9	Métricas de evaluación de los modelos sobre el conjunto de prueba completo	40
10	Métricas de evaluación de AstroNet segmentadas por la clase espectral de la estrella anfitriona	56
11	Métricas de evaluación de ExoNet segmentadas por la clase espectral de la estrella anfitriona	56
12	Métricas de evaluación del modelo híbrido segmentadas por la clase espectral de la estrella anfitriona	57
13	Métricas de evaluación de AstroNet segmentadas por la etapa evolutiva de la estrella anfitriona	57
14	Métricas de evaluación de ExoNet segmentadas por la etapa evolutiva de la estrella anfitriona	58
15	Métricas de evaluación del modelo híbrido segmentadas por la etapa evolutiva de la estrella anfitriona	58

INTRODUCCIÓN

Los modelos de aprendizaje automático se proyectan como una fuerte herramienta en el dominio de la astronomía; a medida que los telescopios modernos capturan cada vez una mayor cantidad de datos del universo, surge la necesidad de realizar tareas de análisis y clasificación con menor supervisión humana. Un caso particular es la búsqueda de planetas fuera del sistema solar, también llamados exoplanetas. Desde el descubrimiento del primer exoplaneta, 51 Pegasi b, en 1995, miles de ejemplares han sido identificados, expandiendo significativamente el conocimiento estadístico del universo.

La estrategia más fructífera para esta tarea es el método del tránsito, que consiste en buscar disminuciones periódicas en el brillo observado de una estrella, para luego evaluar si son provocadas por un exoplaneta orbitando a su alrededor, por ruido instrumental, u otros fenómenos astronómicos; un proceso de validación denominado *vetting*. Los modelos de aprendizaje profundo han surgido como una alternativa robusta a la inspección manual ya que ahorran tiempo a los expertos y proporcionan resultados más uniformes. Esta propiedad es deseable, puesto que permite estimar la completitud de los catálogos y derivar estadísticas poblacionales con mayor certeza. En este contexto, un eje poco estudiado en la literatura es cómo varía el rendimiento de estos modelos al inspeccionar señales provenientes de estrellas con distintas características físicas.

El presente trabajo se enfoca en comparar distintos modelos de aprendizaje profundo, con una mirada en profundidad a la efectividad de detección según las características de la estrella anfitriona, a fin de describir su rendimiento a lo largo de distintos segmentos de la población estelar. Para ello, se implementaron tres arquitecturas presentes en el estado del arte y se evaluó su rendimiento utilizando métricas clave de clasificación, segmentando las señales en base a los parámetros del sistema de clasificación de Morgan-Keenan: la clase espectral y la clase de luminosidad (etapa evolutiva).

La estructura del documento se detalla a continuación. En el Capítulo 1 se introduce el contexto del problema a abordar y se plantean los objetivos de la investigación. En el Capítulo 2 se definen los conceptos teóricos relevantes para el trabajo, incluyendo las bases del aprendizaje profundo, las fuentes de falsos positivos y la clasificación estelar. El Capítulo 3 detalla la metodología de la propuesta, describiendo el conjunto de datos, el preprocesamiento y los modelos implementados. En el Capítulo 4 se presentan y analizan los resultados obtenidos. Finalmente, en el Capítulo 5 se exponen las conclusiones y se proponen posibles líneas de trabajo futuro.

CAPÍTULO 1

DEFINICIÓN DEL PROBLEMA

1.1 Introducción

La búsqueda y caracterización de planetas fuera del sistema solar es uno de los campos de interés en la astrofísica contemporánea. Múltiples misiones espaciales han dedicado sus esfuerzos a adquirir datos que permitan detectar la presencia de nuevos exoplanetas y conocer sus características físicas. Para este propósito se han empleado diversas técnicas analíticas, entre las cuales destacan, por el número de descubrimientos logrados, la medición de velocidad radial —que busca desplazamientos en el espectro luminoso de una estrella provocados por la acción gravitacional de un planeta cercano—, y el método del tránsito, que se basa en la detección de oscurecimientos periódicos en la intensidad de la luz emitida por una estrella, provocados por un planeta que la cubre parcialmente durante su órbita.

El telescopio espacial Kepler monitoreó el brillo de más de 150.000 estrellas a lo largo de sus cuatro años activo entre 2009 y 2013 (recibiendo además una extensión de la misión bajo el nombre de K2), con el objetivo principal de buscar exoplanetas similares a la Tierra. El análisis de sus datos ha permitido la confirmación de más de 2500 nuevos exoplanetas [NASA Exoplanet Archive, s.f.].

1.2 Descripción del problema

La inspección en profundidad de los datos adquiridos por Kepler y otras misiones permite generar catálogos planetarios y derivar estadísticas fundamentales como, por ejemplo, la tasa de ocurrencia de planetas similares a la Tierra orbitando estrellas de características análogas al Sol [Catanzarite y Shao, 2011], o de candidatos a planetas en función de la metalicidad de las estrellas [Narang et al., 2018].

Dado que estas estimaciones requieren conocer la completitud de los catálogos —es decir, la cantidad de planetas que no han sido detectados en la muestra ya sea debido a limitaciones instrumentales o sesgos en los métodos de búsqueda— resulta pertinente medir la capacidad de los modelos de aprendizaje profundo para aislar los verdaderos candidatos a planeta de los falsos positivos. No obstante, es común que las métricas de rendimiento en la literatura se calculen de forma global, considerando todas las muestras como equivalentes, sin ahondar en las diferencias de efectividad que los modelos exhiben al ser evaluados en estrellas de distintas naturalezas astrofísicas.

Diversos estudios han llamado la atención sobre la relevancia de las características estelares en las tareas de detección de tránsito y *vetting*. Por ejemplo, los autores de (Sagear, 2020) ocupan una herramienta computacional llamada *ktransit* capaz de inyectar tránsitos simulados en curvas de luz, demostrando que la efectividad de recuperación de las señales se ve afectada por la variabilidad intrínseca de la estrella, estableciendo límites de detección de tránsitos como una función del radio y período orbital del exoplaneta. En [Bryant et al., 2023] se realizó una búsqueda exhaustiva de planetas gigantes orbitando estrellas de baja masa (menos de 0.71 masas solares) logrando identificar exoplanetas que habían sido ignorados en análisis previos.

El propósito central de este trabajo es estudiar el rendimiento de distintas arquitecturas de redes neuronales artificiales para la tarea de *vetting* (clasificación de eventos que cruzan el umbral de detección o *Threshold Crossing Events* en candidatos a planeta y falsos positivos), evaluando específicamente cómo este rendimiento fluctúa frente a variaciones en las propiedades fundamentales de la estrella anfitriona.

1.3 Objetivos

1.3.1 Objetivo general

Comparar el rendimiento de distintos modelos de aprendizaje profundo aplicados a la clasificación automatizada de señales de tránsito exoplanetario (*vetting*) en datos fotométricos, evaluando el efecto que tienen las características fundamentales de la estrella anfitriona en su capacidad predictiva.

1.3.2 Objetivos específicos

- O1. Investigar e implementar arquitecturas de aprendizaje profundo en el estado del arte, aplicadas a la tarea de *vetting* de exoplanetas mediante el método del tránsito.
- O2. Evaluar globalmente cada modelo implementado utilizando métricas estadísticas para tareas de clasificación binaria.
- O3. Comparar el desempeño de los modelos utilizando las propiedades físicas de las estrellas anfitrionas (clase espectral y etapa evolutiva) como ejes principales de segmentación y análisis.

CAPÍTULO 2

MARCO CONCEPTUAL

2.1 El método del tránsito

La idea original detrás del método del tránsito fue propuesta en 1952 por Otto Struve en un breve artículo científico. En él, planteaba que la disminución en la luz percibida de una estrella al ser eclipsada parcialmente por un planeta orbitando a su alrededor debería poder ser detectada con los métodos fotoeléctricos disponibles en su época. Su propuesta no generaría tracción sino hasta 1999 con la primera observación del tránsito de un planeta ya descubierto mediante velocidad radial, HD 209458b, lo que motivó la búsqueda sistemática de tránsitos en otras estrellas [Deeg y Alonso, 2018]. El primer nuevo exoplaneta descubierto exclusivamente con este método llegaría en 2003 por parte del proyecto astronómico OGLE, denominado OGLE-TR-56b [Konacki et al., 2023].

La cantidad física principal utilizada para detectar tránsitos es el flujo F , que corresponde al brillo observado de una estrella (o del sistema estrella-planeta) por unidad de tiempo. El flujo se mide utilizando sensores electrónicos llamados *Charge-Coupled Devices* o CCDs, los cuales capturan fotones y los convierten en señales eléctricas. El fotómetro de la misión Kepler estaba compuesto por 42 CCDs, cada uno con una resolución de 1024 x 2200 píxeles [Koch et al., 2010]. Para estimar el flujo de una estrella en particular se aplica una “máscara de apertura” centrada en el objetivo y se suman los valores de los píxeles en su interior.

Bajo las suposiciones de que el flujo emitido directamente desde el planeta es despreciable, y de que tanto el planeta como la estrella poseen forma esférica, la variación relativa del flujo δ es proporcional a la razón entre las áreas proyectadas de ambos cuerpos, definida mediante el radio del planeta R_p y el radio de la estrella R_s [Deeg y Alonso, 2018].

$$\delta = \frac{\Delta F}{F} \approx \left(\frac{R_p}{R_s} \right)^2$$

Donde $\Delta F = F_{\text{sin tránsito}} - F_{\text{con tránsito}}$ y δ es la profundidad del tránsito, parámetro que representa la disminución relativa del brillo percibido de la estrella y que se suele medir en partes por millón (ppm). Mientras mayor sea δ , más notoria es la disminución de flujo en la denominada “curva de luz”, una serie temporal obtenida al concatenar las mediciones de flujo en el tiempo. La Figura 1 ilustra el fenómeno físico de un exoplaneta transitando frente a su estrella y el efecto resultante en la curva de luz captada por el telescopio.

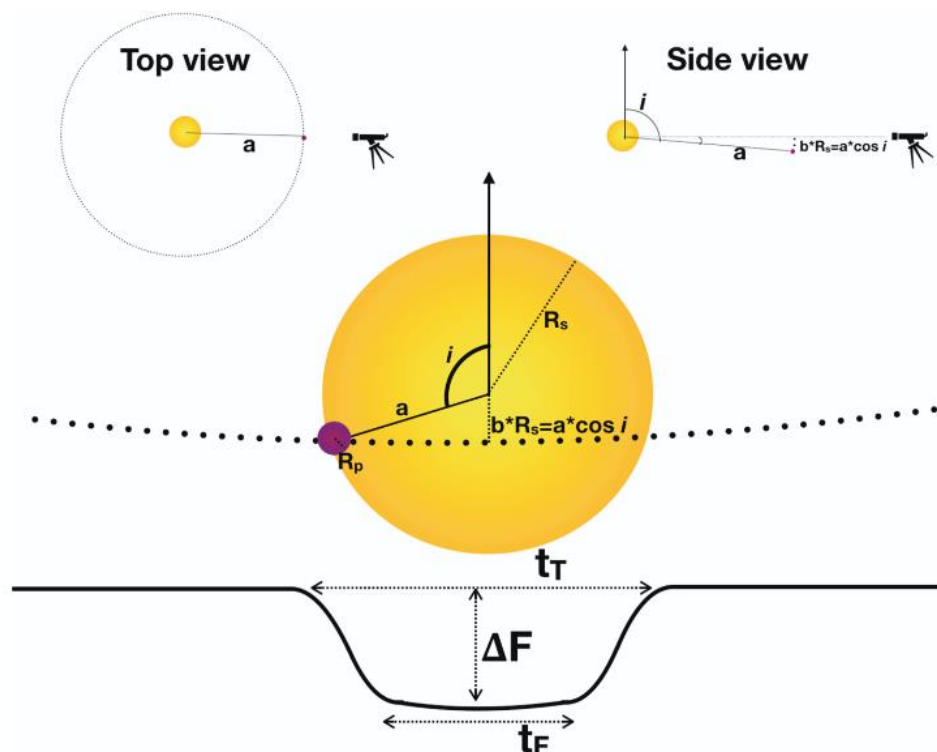


Figura 1: Esquema de un planeta transitando frente a una estrella desde distintos puntos de vista. La curva de luz ilustra la variación del flujo de la estrella.

Fuente: [Deeg y Alonso, 2018].

Para automatizar la detección de estos eventos en cientos de miles de curvas de luz, se han empleado múltiples algoritmos. El más estandarizado en la astronomía moderna es el algoritmo *Box Least Squares* [Kovács et al., 2002]. Este método ajusta una forma geométrica de “caja” (pulso rectangular) a la curva iterando a través de una grilla de parámetros del tránsito: período orbital, profundidad, duración y época (el tiempo del primer tránsito detectado). El algoritmo selecciona aquella configuración que maximiza el Residuo de la Señal (SR, por sus siglas en inglés, *Signal Residue*), una métrica que representa la proporción de la varianza en los datos que logra ser explicada por el modelo de tránsito propuesto.

2.2 Signal-to-Noise Ratio (SNR) en curvas de luz

Otra cantidad central en la detección de tránsitos y en el posterior proceso de *vetting* es la razón Señal-Ruido o SNR (*Signal-to-Noise Ratio*), que describe la intensidad de la señal de interés comparada con las diversas fuentes de ruido presentes en la medición. En este contexto, la “señal” es la disminución del flujo provocada por el planeta, y el “ruido” aglomera a diversas fuentes de perturbación, como la propia variabilidad intrínseca de la

estrella, los fotones capturados del fondo estelar, el ruido de lectura térmico en los sensores CCD y (en el caso de telescopios terrestres) las variaciones atmosféricas.

Una relación fundamental planteada en [Kipping, 2023] entre el ruido de una señal y la detectabilidad del tránsito es la siguiente:

$$SNR \propto \delta \sqrt{D}$$

Donde D es la duración del tránsito, la cantidad de puntos ocurridos durante el tránsito, o alguna medida similar, y la constante de proporcionalidad depende del tratamiento específico del ruido. [Kipping, 2023] llama la atención sobre la simplificación implícita que realiza el algoritmo *Box Least Squares* al modelar los tránsitos con una forma de pulso estrictamente rectangular, destacando la relevancia de estimar la SNR mediante formulaciones más apropiadas, especialmente cuando el objetivo analítico es calcular tasas de ocurrencia planetaria rigurosas.

En el caso específico de la misión Kepler, el método oficial para cuantificar el ruido fue descrito en [Christiansen et al., 2012], y se fundamenta en el concepto de la Precisión Fotométrica Diferencial Combinada (CDPP, por sus siglas en inglés). De esta forma, la SNR para un candidato detectado por Kepler se formula como:

$$SNR = \sqrt{N_{tr}} * \frac{\delta}{CDPP_{eff}}$$

Donde N_{tr} es la cantidad media de tránsitos observados para una señal con un período dado, y el valor $CDPP_{eff}$ correspondiente a la duración de ese tránsito es un parámetro precalculado y publicado por NASA para cada estrella objetivo.

La importancia crítica de la SNR radica en su impacto directo sobre la interpretabilidad de una curva de luz. A mayor SNR, el perfil geométrico de un tránsito periódico es más fácil de reconocer e inspeccionar (tanto por astrónomos como por modelos de aprendizaje profundo). Por el contrario, en escenarios de baja SNR, perturbaciones estelares genuinas pueden quedar ocultas, o bien anomalías instrumentales pueden imitar artificialmente la forma de un tránsito.

Un matiz fundamental en la fotometría de tránsitos es que la forma de este es una proyección directa de las propiedades físicas del sistema. El ejemplo más evidente es la proporción entre los radios de la estrella y el planeta (R_p/R_s), que determina la profundidad δ . Más allá de los radios, se ha demostrado que, bajo supuestos razonables, la densidad estelar puede ser estimada utilizando exclusivamente parámetros extraídos de la fotometría [Seager y Mallén-Ornelas, 2003], en particular: ΔF , el período orbital P , y los tiempos t_T y t_F demarcados en la Figura 1.

Bajo estas condiciones, resulta ineludible considerar el efecto que tienen las propiedades estelares sobre el ruido y, por ende, sobre la SNR. La variabilidad natural de la estrella actúa como fuente de ruido durante la detección. Un estudio aplicado a 34,000 estrellas de la

secuencia principal observadas por Kepler encontró que las estrellas de períodos de rotación más cortos y menor temperatura contienen mayor amplitud en su variabilidad fotométrica ligada a manchas estelares [McQuillan, 2014]. Del mismo modo, [Bastien et al., 2013] documentaron una correlación observacional entre la gravedad superficial de una estrella ($\log g$) y las variaciones a corto plazo en su flujo, estableciendo que las propiedades de la estrella anfitriona impactan en el nivel de dificultad en las tareas de detección y *vetting*.

2.3 Aprendizaje de máquina

El aprendizaje de máquina (ML) es un dominio de la computación cuya idea central es enseñarle a un algoritmo a realizar una tarea particular, sin entregarle las reglas específicas para completarla correctamente. Algunas tareas típicas son la clasificación, detección de anomalías y predicción de valores.

El aprendizaje supervisado utiliza un conjunto de pares de entrada y salida o etiqueta (x, y), comúnmente llamado conjunto de datos (*dataset*), para enseñarle a un modelo a producir las salidas deseadas. Cada elemento de entrada o salida suele consistir en un vector unidimensional que representa valores numéricos o categóricos.

Típicamente el *dataset* es dividido en tres partes; una de ellas es utilizada para entrenar el modelo mediante algún algoritmo, otra para medir el aprendizaje durante el entrenamiento y la última para evaluar el rendimiento del modelo al terminar. Dichos subconjuntos reciben los nombres de conjunto de entrenamiento (*train set*), validación (*validation set*) y prueba (*test set*) respectivamente. Luego de ser entrenado, el modelo puede recibir nuevas entradas x' y generar las etiquetas y' correspondientes. Este proceso es conocido como inferencia.

2.4 Aprendizaje profundo

El aprendizaje profundo o *Deep Learning* es un subconjunto de los métodos de aprendizaje de máquina y su propiedad distintiva es el uso de redes neuronales artificiales. Una neurona artificial es una unidad de cómputo que toma un vector de entrada x y realiza una operación matemática de la forma:

$$f(x) = g \left(\sum_{i=1}^d w_i x_i - b \right)$$

Donde la función g es denominada función de activación, los parámetros w_i son llamados pesos de conexión entre la neurona y aquellas otras de las que recibe información, y b es llamado sesgo (*bias*) o umbral de excitación. Estos parámetros son ajustados durante el entrenamiento, modificando el comportamiento de la red. Posteriormente, el resultado $f(x)$ es propagado a otras neuronas.

En un modelo simple, las neuronas son dispuestas en capas densas, donde cada neurona está conectada con todas las de la capa siguiente y la información fluye en un solo sentido sin generar ciclos en el grafo, arquitectura conocida como red neuronal totalmente conectada (*Fully Connected Neural Network* o FCNN), ilustrada en la [Figura 2](#). La primera capa de una red es denominada capa de entrada (*input layer*) y recibe los datos, la última capa es denominada capa de salida (*output layer*) y produce las predicciones, y todas las capas internas son llamadas capas ocultas (*hidden layers*).

Cabe destacar que, gracias a la naturaleza vectorial de las operaciones en la red, los cálculos de cada capa se pueden paralelizar de forma eficiente mediante multiplicación de matrices. Otro gran potencial de las redes neuronales es su capacidad para trabajar con datos no estructurados, como pueden ser series de tiempo o imágenes. Al utilizar, por ejemplo, una curva de luz para clasificar un tránsito detectado en candidato a planeta o falso positivo, un modelo de ML clásico necesitaría la extracción previa de características numéricas o categóricas, como por ejemplo la profundidad del tránsito δ , el valor de SNR y el flujo F . En cambio, una red neuronal apropiada puede trabajar directamente con la propia serie de tiempo de valores (t, F_t) .

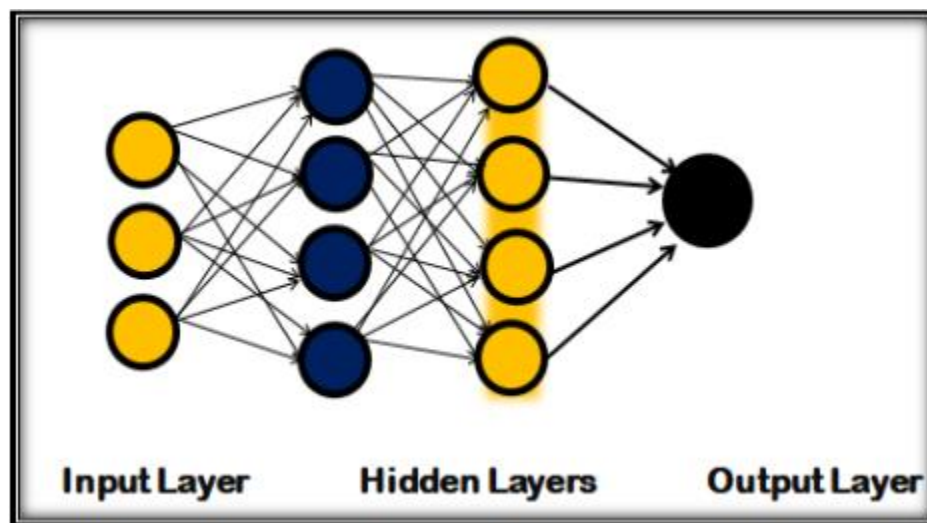


Figura 2: Diagrama de una red neuronal artificial FCNN con dos capas ocultas de cuatro neuronas cada una.

Fuente: [\[Qamar y Zardari, 2023\]](#).

2.4.1 Funciones de activación

Las funciones de activación suelen ser no-lineales, permitiendo a la red aprender representaciones complejas e interpretar conjuntos de datos no lineales y de alta dimensionalidad [Sharma et al., 2020]. Algunas de las más utilizadas son las siguientes:

- **Sigmoide:** La función sigmoide convierte la entrada a una salida en el intervalo $[0,1]$, siendo útil para producir salidas interpretables como probabilidades. Se suele representar como σ y está definida por:

$$\sigma(x) = \frac{1}{(1 + e^{-x})}$$

- **Tangente hiperbólica:** La función tangente hiperbólica es similar a la sigmoide, pero es simétrica alrededor del origen, permitiendo entregar salidas de distinto signo, en el intervalo $(-1,1)$. Se define como:

$$\tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$$

- **Rectified Linear Unit (ReLU):** La función más popular para capas ocultas en redes profundas. Entrega una salida lineal excepto para valores negativos, los cuales son llevados a cero. Permite que algunas neuronas se “apaguen” durante la operación, reduciendo la complejidad efectiva del modelo. Se define mediante la operación:

$$ReLU(x) = \max(0, x)$$

- **Leaky ReLU:** Versión mejorada de ReLU, diseñada para evitar problemas de la ReLU clásica que aparecen durante el entrenamiento (como el decaimiento de neuronas o *dying ReLU*). En lugar de llevar valores negativos a cero, mantiene una pequeña componente lineal mediante un parámetro α , típicamente igual a 0.01. Se define como:

$$LReLU(x) = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases}$$

El comportamiento de estas funciones está ilustrado en la [Figura 3](#).

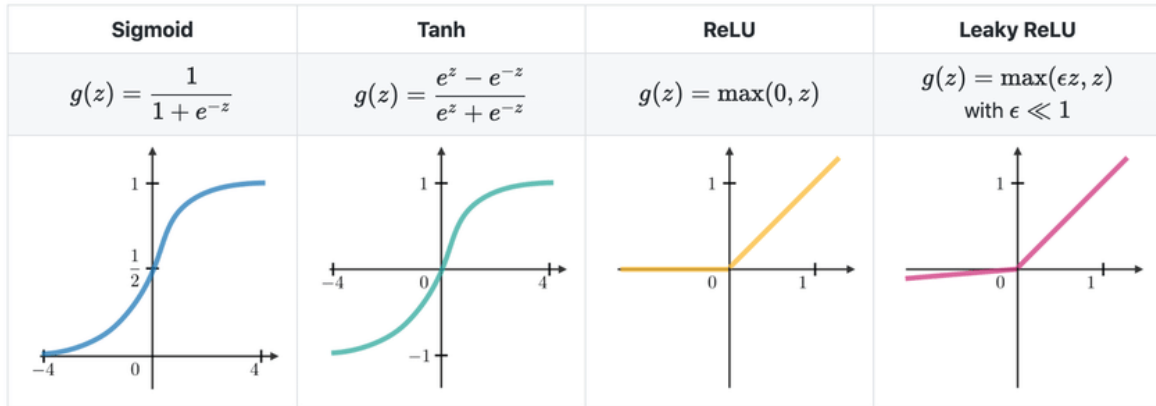


Figura 3: Funciones de activación más populares en modelos de aprendizaje profundo. De izquierda a derecha: función Sigmoide, función Tangente hiperbólica, función Rectified Linear Unit y función Leaky Rectified Linear Unit. Fuente: [Loaiza, 2020].

2.4.2 Funciones de pérdida

El entrenamiento de una red neuronal utiliza los ejemplos en el *dataset* para ajustar los parámetros de la red. Para ello, se debe poder medir la calidad de una predicción $\hat{y}$ ante una entrada x . Esto se consigue con una función de pérdida o costo, comúnmente denotada por L , cuya definición depende del tipo de problema.

En el caso de clasificación binaria, donde las etiquetas son valores 0 o 1, la clase con etiqueta 1 es denominada clase positiva y las salidas de la red corresponden a valores en el intervalo $[0,1]$ indicando la probabilidad de pertenecer a la clase positiva. En estos casos se suele utilizar la función de Entropía Cruzada Binaria (*Binary Cross-Entropy*), que compara la probabilidad predicha p_i con la etiqueta real y_i . Para un conjunto de N ejemplos, se tiene la siguiente ecuación:

$$L = -\frac{1}{N} \sum_{i=1}^N y_i \cdot \log(p_i) + (1 - y_i) \cdot \log(1 - p_i)$$

Esta función puede ser adaptada para la tarea de clasificación de múltiples categorías, recibiendo el nombre de Entropía Cruzada Categórica. Otra función de pérdida común es el Error Cuadrático Medio (*Mean Squared Error*), utilizado frecuentemente en problemas de regresión.

2.4.3 Entrenamiento de una red neuronal

Una vez definida la arquitectura del modelo en términos de la cantidad de capas, número de neuronas por capa, disposición de las conexiones entre neuronas y sus respectivas funciones de activación, el entrenamiento requiere de varios pasos, los cuales describiremos a continuación:

1. **Inicialización de los parámetros:** todos los parámetros (pesos de conexión y sesgos) han de ser inicializados con algún valor. Una estrategia sencilla es la inicialización aleatoria, que utiliza una distribución uniforme o normal. Varias estrategias han sido propuestas. Por ejemplo, la comúnmente llamada inicialización de Xavier [Glorot y Bengio, 2010] utiliza una distribución parametrizada con el número de conexiones entrantes y salientes para cada capa.
2. **Propagación hacia adelante (forward pass):** Los casos de ejemplo son introducidos a la red, y la información se propaga por las capas hasta generar la salida. Los datos suelen ser ingresados en lotes (*batches*), y luego de cada lote se calcula la función de pérdida.
3. **Propagación hacia atrás (backpropagation):** En esta etapa se calcula la derivada parcial de la función de pérdida respecto a cada parámetro, para así ajustarlos en la dirección que la minimice. En este cálculo se introduce la tasa de aprendizaje que define cuánto se modifican los parámetros en cada paso.

Los pasos de propagación hacia adelante y hacia atrás se repiten hasta alcanzar un criterio de parada. En general se suele definir un número de épocas de entrenamiento, y por cada época se utiliza el conjunto de entrenamiento completo. Alternativamente, se puede detener el proceso cuando no se encuentran mayores cambios en la función de pérdida en varias iteraciones seguidas, indicando que el algoritmo ha logrado la convergencia a un mínimo, al menos local, de la función de pérdida.

Todos los parámetros que no forman parte de la red pero que influyen en el proceso de entrenamiento, como el tamaño de los lotes, número de épocas y tasa de aprendizaje son denominados hiperparámetros, y suelen ser ajustados mediante sintonización de parámetros.

A continuación, definiremos brevemente otros conceptos relacionados al entrenamiento.

- **Sobreajuste (*overfitting*):** Se habla de sobreajuste cuando el modelo optimiza sus parámetros de sobremano para el conjunto de entrenamiento, lo que resulta en una menor capacidad de generalización a nuevos datos.
- **Aumento de datos:** Durante el entrenamiento se suelen utilizar técnicas para diversificar sintéticamente el conjunto de datos, ayudando a reducir el sobreajuste. Algunos ejemplos son la introducción de ruido aleatorio, o la aplicación de reflexiones u otras transformaciones morfológicas a los datos.
- **Optimizadores:** Los optimizadores modifican la tasa de aprendizaje dinámicamente para cada parámetro, ayudando a mejorar la convergencia. El optimizador más popular es llamado Adam, que utiliza estimaciones del primer y segundo momento del gradiente de la función de pérdida [Kingma y Ba, 2014].

2.4.4 Redes neuronales convolucionales

Las redes neuronales convolucionales (CNN por sus siglas en inglés) fueron diseñadas originalmente para tareas de reconocimiento visual, como clasificación de imágenes o caracteres. A través de la operación de convolución, una entrada en 2D se procesa mediante un *kernel* de menor tamaño, computando el producto punto entre los elementos correspondientes de la entrada y los del kernel. El cálculo se repite moviendo de posición el kernel hasta haber cubierto todos los elementos de la entrada. El resultado de esta operación es un mapa de características (MC), también llamado mapa de activación.

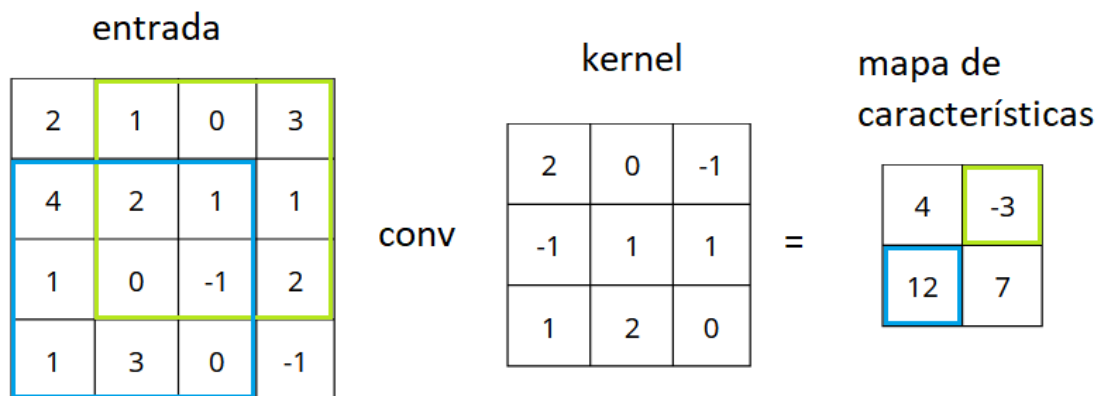


Figura 4: Ejemplo de la operación de convolución entre dos matrices. La matriz de entrada y el kernel tienen dimensiones 4x4 y 3x3 respectivamente. El FM producido tiene dimensiones 2x2. Fuente: Elaboración propia.

En el caso de entradas de mayor dimensión, como un tensor de dimensiones $N \times M \times D$ (equivalente a D matrices de tamaño $N \times M$) se han de utilizar D kernels componiendo un filtro, cada kernel siendo operado con uno de los canales de entrada y sumando las salidas para obtener un resultado de dos dimensiones. Más aún, distintos filtros pueden ser aplicados a la entrada, y entrenados para reconocer distintos patrones. En el caso de procesar imágenes, distintos filtros pueden identificar bordes, formas o texturas.

Las dimensiones del mapa de características obtenido dependen de varios factores además del tamaño del kernel. La cantidad de espacios que se mueve el kernel en cada paso de la convolución es llamada salto (*stride*), definiendo qué tan sobrepuesta está la información entre elementos adyacentes del MC. Con un salto de valor alto, las dimensiones del MC son mucho menores a las de la entrada. El *padding* es una extensión a las dimensiones de una entrada con nuevos valores o ceros, permitiendo modificar el tamaño de la salida. Se puede apreciar un ejemplo en la Figura 4, donde no hay *padding* y el salto es de 1.

Otro concepto fundamental en las redes convolucionales es el de *pooling*. Una capa de *pooling* aplica una reducción de las dimensiones del mapa de características entrante intentando retener la información valiosa, generalmente mediante un pequeño kernel que implementa la función $\max()$ que conserva el elemento mayor o $\text{avg}()$ que conserva la media entre los elementos. Esta reducción de dimensiones disminuye la complejidad computacional y el riesgo de sobreajuste.

En general, las capas convolucionales son combinadas con capas de *pooling*, y se añade posteriormente un bloque de una o más capas densas para generar la salida final. En la [Figura 5](#) se presenta el diseño de la red convolucional AlexNet, famosa por demostrar la eficiencia de las redes neuronales entrenadas con unidades de procesamiento gráfico (GPUs).

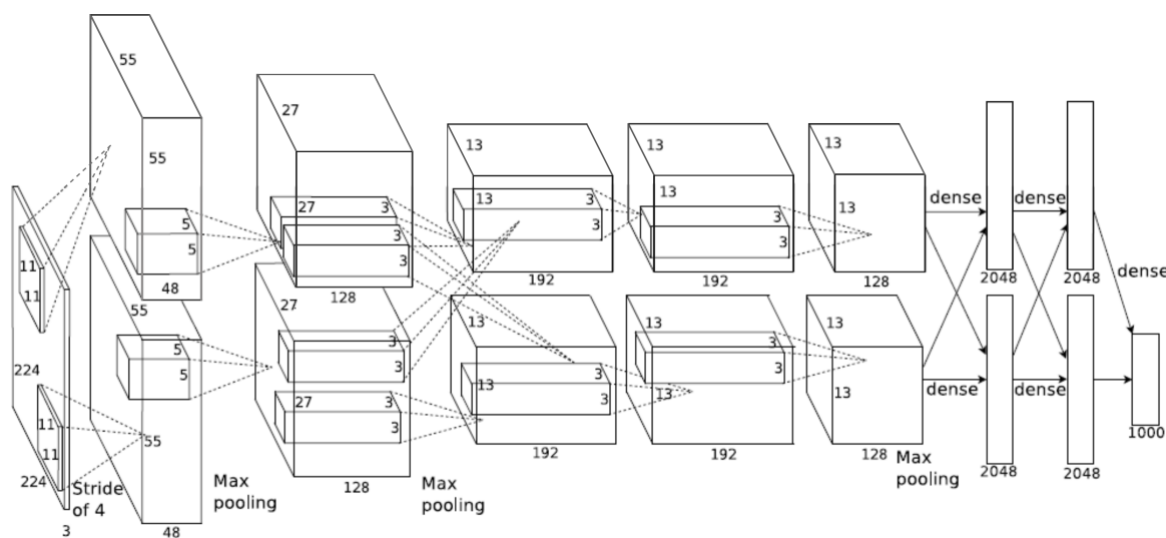


Figura 5: Esquema de la arquitectura AlexNet, utilizado para clasificar imágenes del conjunto ImageNet. Cuenta con 5 capas convolucionales y 3 capas FC. Luego de las capas ocultas 1, 2 y 5 se aplica *pooling* mediante la función $\max()$. Fuente: [\[Krizhevsky et al., 2012\]](#).

2.4.5 Redes Neuronales Recurrentes

Hasta el momento, se han descrito únicamente arquitecturas de redes neuronales carentes de memoria o contexto. En otras palabras, la predicción y' para una entrada x' es independiente de las observaciones previas procesadas por la red.

Las redes neuronales recurrentes (RNR) fueron diseñadas con el propósito de procesar series de tiempo y secuencias de elementos donde el orden cronológico o secuencial resulta crítico. El modelo básico propuesto por Jeffrey Elman se define formalmente mediante las siguientes ecuaciones:

$$y_t = \sigma_y (Wh_t + b_y)$$

$$h_t = \sigma_h (Ux_t + Vh_{t-1} + b_h)$$

Donde x_t , y_t , h_t y h_{t-1} representan vectores y $t \in \{1, 2, \dots, T\}$ denota el paso temporal. El vector y_t es la salida para el instante t , la cual depende directamente del estado oculto h_t . A su vez, el estado oculto h_t depende del estado oculto previo h_{t-1} . Las matrices de pesos W , U y V , junto con los vectores de los sesgos b_h y b_y constituyen los parámetros entrenables de la red, mientras que σ_y y σ_h corresponden a las funciones de activación de cada capa.

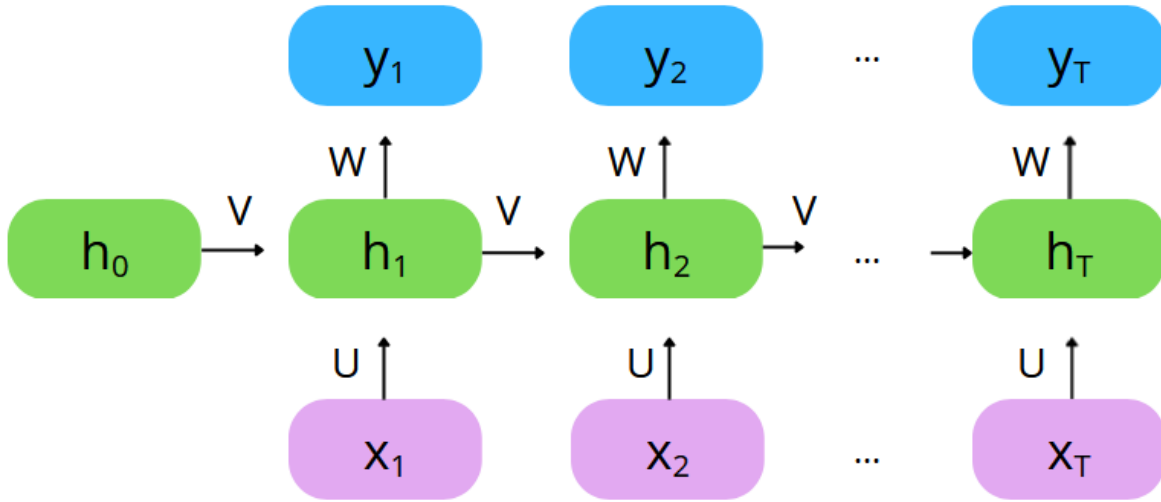


Figura 6: Diagrama de una red recurrente de Elman “desenrollada” en el tiempo.
Fuente: Elaboración propia.

La célula mantiene estos parámetros constantes a través del tiempo (pesos compartidos), lo que otorga a las redes recurrentes la capacidad de operar sobre secuencias de longitud T variable, iterando recursivamente sobre los vectores de entrada x_t . El comportamiento de una RNR se puede visualizar graficando sus iteraciones como una secuencia lineal de capas, un proceso conocido como “desenrollar” la red. Un ejemplo de esto se incluye en la [Figura 6](#). El estado h_t se calcula evaluando conjuntamente el elemento de entrada actual x_t y la memoria del pasado h_{t-1} . En el inicio, $t=1$, el estado de memoria contiene exclusivamente información de x_1 , mientras que en $t=T$, el vector h_T encapsula teóricamente información de toda la secuencia introducida.

En la configuración general, las secuencias de entrada y salida tienen la misma longitud ($T_x = T_y$), generando respuestas de forma sincrónica a medida que se desenrolla la red. No obstante, esta arquitectura es flexible: permite construir modelos muchos-a-uno (conservando únicamente la predicción final y_T), o modelos uno-a-muchos (utilizando la salida y_{t-1} como alimentación de la entrada en t , reemplazando observaciones futuras).

Una limitación importante de este modelo radica en su dificultad para modelar dependencias a largo plazo, es decir, relaciones o patrones entre observaciones distantes dentro de la secuencia. El motivo se encuentra en el proceso de entrenamiento: al aplicar el algoritmo de propagación hacia atrás en el tiempo (*backpropagation through time*), las señales de error sufren multiplicaciones sucesivas, provocando que los gradientes crezcan descontroladamente (gradientes explosivos) o decaigan exponencialmente hacia cero (gradientes desvanecientes). Como resultado de esta inestabilidad, la red pierde su capacidad de ajustar los parámetros adecuadamente [Salehinejad et al., 2017]. Para mitigar estos problemas, se han propuesto diversas arquitecturas avanzadas, siendo la más destacada la red *Long Short-Term Memory*.

2.4.6 Long Short-Term Memory

Propuesta originalmente en 1997 por Sepp Hochreiter y Jürgen Schmidhuber, la arquitectura LSTM aborda el problema de la memoria incorporando el concepto de célula de memoria C, cuyos flujos de información son regulados por un sistema de compuertas (*gates*). El diseño original contaba con dos mecanismos de control, la puerta de entrada (*input gate*) encargada de filtrar información entrante irrelevante, y la puerta de salida (*output gate*), que regula qué proporción del estado de memoria interno se transmite a las capas posteriores [Hochreiter y Schmidhuber, 1997]. Una tercera puerta fue introducida en [Gers et al., 2000], denominada la puerta de olvido (*forget gate*), la cual dota a la célula con la capacidad de “olvidar” información de la memoria que ha dejado de ser pertinente. La configuración completa se observa en la Figura 7.

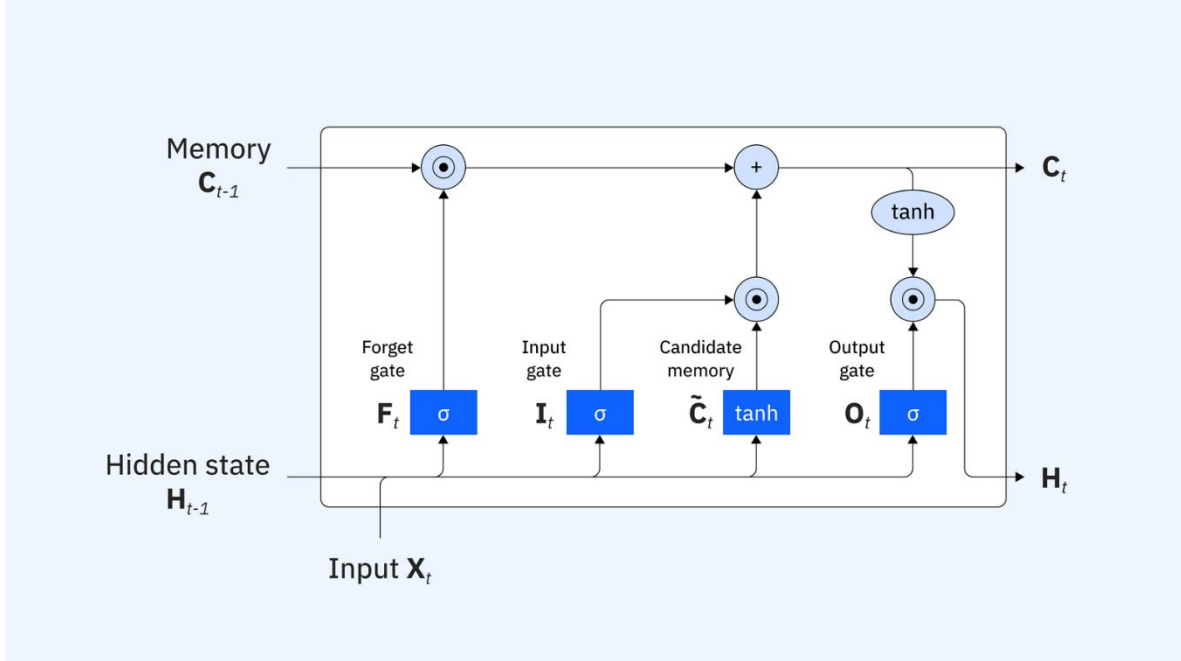


Figura 7: Diagrama de una célula de memoria en el modelo LSTM.

Fuente: [Noble, s.f.].

El comportamiento de la red queda definido matemáticamente por el siguiente conjunto de operaciones para cada paso temporal t :

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$

$$\tilde{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$$

$$h_t = o_t \odot \sigma_h(C_t)$$

En las ecuaciones, los vectores f_t , i_t y o_t representan las activaciones de las puertas de olvido, entrada y salida, respectivamente. El término $\tilde{C}_t$ es el vector candidato a memoria en el instante t , el cual es modulado por la puerta de entrada para actualizar el estado de memoria C_t . Finalmente, el símbolo $\odot$ denota el producto de Hadamard (multiplicación elemento a elemento).

2.4.7 Arquitectura Encoder-Decoder

Una propiedad deseable para las redes recurrentes es la capacidad para ingerir una secuencia de longitud T_x y generar una salida de longitud T_y distinta ($T_x \neq T_y$). Esta disparidad surge naturalmente en áreas como el procesamiento de lenguaje natural (por ejemplo, en la traducción automática, donde una oración en un idioma no comparte la misma cantidad de palabras que su traducción). Los sistemas diseñados para abordar esta tarea se agrupan bajo el término general de modelos Seq2Seq (*sequence-to-sequence*).

La arquitectura *Encoder-Decoder* permite el mapeo Seq2Seq acoplando dos redes recurrentes independientes, típicamente bloques LSTM. El codificador (*Encoder*) procesa iterativamente la secuencia de entrada completa, y su estado oculto final h_T se define como el vector de contexto C , el cual tiene el propósito de encapsular la información de toda la observación. A continuación, el decodificador (*Decoder*) emplea este contexto C como semilla o estado inicial para generar de forma autónoma la secuencia de salida, calculando sus propios estados ocultos s_t . La dinámica de transferencia entre ambos componentes se detalla en la [Figura 8](#).

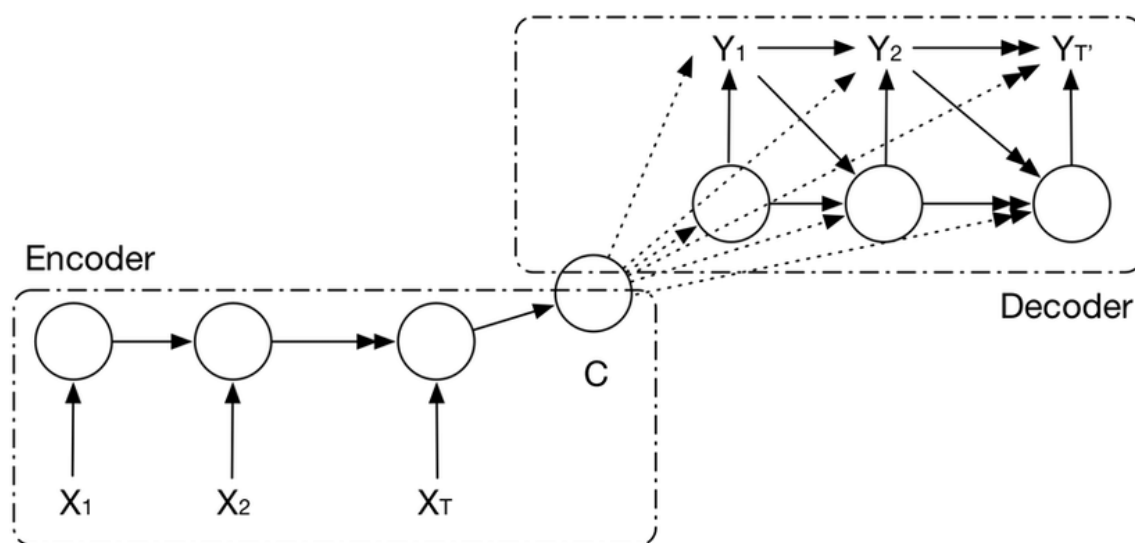


Figura 8: Diagrama de la arquitectura *Encoder-Decoder*. Cada componente es una red neuronal recurrente. El *Encoder* genera el contexto C utilizado por el *Decoder*. Fuente: [\[Peng et al., 2018\]](#)

2.4.8 Mecanismos de atención

Los autores de [\[Bahdanau et al., 2014\]](#) identificaron un cuello de botella en la arquitectura clásica *Encoder-Decoder*: la dependencia de un vector C de tamaño fijo. Su estudio planteó que comprimir secuencias extensas en un vector de dimensionalidad rígida degrada el rendimiento de la red, particularmente al evaluar datos más largos que los provistos

durante el entrenamiento. Para aliviar el problema, propusieron extraer todos los estados ocultos generados por el codificador (h_j) para formar un conjunto de “anotaciones”, e introdujeron un modelo de alineación capaz de calcular un contexto dinámico c_t para cada paso del decodificador, valiéndose de su estado previo s_{t-1} . Este avance permite a la red “prestar atención” selectivamente en distintos fragmentos de la entrada original según sea necesario.

El vector c_t se calcula como una suma ponderada de todas las anotaciones:

$$c_t = \sum_{j=1}^{T_x} \alpha_{tj} h_j$$

La ponderación α_{tj} asignada a la anotación h_j se normaliza mediante una función *softmax*, asegurando que la suma de los valores α_{tj} sea igual a uno.

$$\alpha_{tj} = \frac{\exp(e_{tj})}{\sum_{k=1}^{T_x} \exp(e_{tk})}$$

Donde las puntuaciones de alineación e_{tj} están dadas por:

$$e_{tj} = a(s_{t-1}, h_j)$$

El modelo de alineación a cuantifica la relevancia entre el instante actual de la generación y cada elemento original de la secuencia. En la práctica, se implementa mediante una pequeña red neuronal totalmente conectada que se optimiza en conjunto con el modelo principal. Al arreglo completo α_{tj} para un paso temporal t se le denomina matriz o vector de atención.

A raíz de esta formulación base, la literatura ha adoptado múltiples variantes según la naturaleza del problema. Entre ellas, el concepto de Transformadores (*Transformers*) introducido en [Vaswani et al., 2017] y fundamentado en el paradigma de auto-atención (*self-attention*), prescindiendo de la recurrencia, se ha consolidado como un pilar fundamental de las redes neuronales para procesamiento de lenguaje.

2.5 Métricas de evaluación

Las métricas de evaluación permiten cuantificar el rendimiento de un modelo de aprendizaje automático, facilitando el proceso de optimización de hiperparámetros y habilitando la comparación con arquitecturas alternativas. La tarea de *vetting* de tránsitos de exoplanetas se plantea como un problema de clasificación binaria, donde la clase positiva corresponde a un candidato a planeta y la clase negativa a un falso positivo o no-

candidato. A continuación, se describen los conceptos básicos y las métricas más utilizadas para evaluar el rendimiento de clasificadores binarios descritos en [Rainio et al., 2024].

La matriz de confusión es un artefacto que permite visualizar la distribución de los resultados de las predicciones frente a la realidad. Está formada por dos filas y dos columnas que contrastan los valores verdaderos (etiquetas reales) y las predicciones entregadas por el modelo. Esta configuración se detalla en la Tabla 1.

Tabla 1: Matriz de confusión.

		Predicción	
		Negativo	Positivo
Etiqueta real	Negativo	TN	FP
	Positivo	FN	TP

Las cuatro casillas de la matriz contienen los siguientes valores:

- **Verdaderos negativos (TN del inglés *true negative*)**: Número de casos negativos predichos como negativos.
- **Falsos positivos (FP del inglés *false positive*)**: Número de casos negativos predichos como positivos.
- **Falsos negativos (FN del inglés *false negative*)**: Número de casos positivos predichos como negativos.
- **Verdaderos positivos (TP del inglés *true positive*)**: Número de casos positivos predichos como positivos.

A partir de estos cuatro valores se definen las métricas fundamentales que describen el comportamiento estadístico del modelo:

- **Exactitud (*Accuracy*)**: Representa la proporción global de predicciones correctas entre todos los casos evaluados.

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

- **Sensibilidad (*Recall*)**: Mide la proporción de casos positivos reales que el modelo logró identificar correctamente. También se le conoce como tasa de verdaderos positivos, en inglés *True Positive Rate*.

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

- **Precisión (*Precision*):** Cuantifica la proporción de predicciones positivas que son correctas en la realidad, penalizando la generación de falsas alarmas.

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

- **F1-Score:** Corresponde a la media armónica entre precisión y sensibilidad. Dado que esta métrica evalúa el desempeño concentrado exclusivamente en la clase positiva, resulta particularmente robusta para escenarios donde el conjunto de datos presenta un alto desbalance de clases.
- **Especificidad (*Specificity*):** Determina la proporción de casos negativos reales que fueron identificados correctamente. También es conocida como la tasa de verdaderos negativos (TVN), en inglés *True Negative Rate*. A partir de esta métrica se define también la tasa de falsos positivos (TFP), en inglés *False Positive Rate*, que indica la fracción de casos reales negativos clasificados erróneamente como positivos.

$$Specificity = \frac{TN}{TN + FP}, \quad TFP = 1 - Specificity$$

La relevancia de cada métrica depende del contexto del problema. En el dominio de la búsqueda exoplanetaria, producir catálogos saturados de falsos positivos requiere una exhaustiva inspección humana posterior (priorizando la precisión), mientras que clasificar erróneamente un tránsito débil puede significar la pérdida de un candidato a planeta (priorizando la sensibilidad).

Adicionalmente, tanto la sensibilidad como la especificidad dependen del umbral de clasificación (la probabilidad mínima requerida para asignar una etiqueta positiva). La curva ROC (*Receiver Operating Characteristic*) ilustra el comportamiento global del clasificador al variar este umbral en el intervalo [0,1], graficando la relación entre la tasa de verdaderos positivos y la tasa de falsos positivos. El área bajo esta curva, denominada AUC (*Area Under Curve*), representa la probabilidad estadística de que el modelo asigne un mayor puntaje a un caso positivo aleatorio que a un caso negativo aleatorio [Vujovic, 2011]. Mientras más se acerque el valor de AUC a 1.0, mayor es la capacidad discriminativa del modelo, independientemente del umbral de clasificación que se utilice. La Figura 9 contiene un ejemplo de la curva ROC y la AUC.

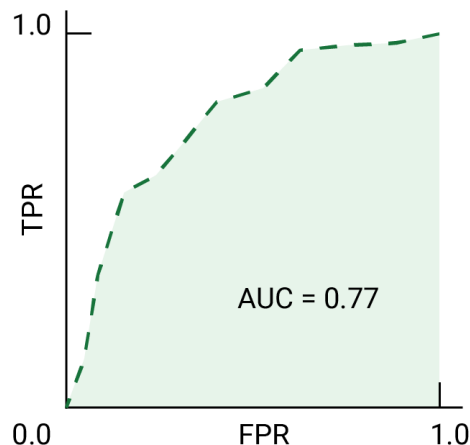


Figura 9: Curva *Receiver Operating Characteristic* de un clasificador binario con AUC = 0.77. Fuente: [Google, s.f.]

2.6 Fuentes conocidas de falsos positivos

La detección de exoplanetas mediante el método del tránsito enfrenta el desafío de que múltiples configuraciones astrofísicas y anomalías estocásticas pueden imitar la morfología de un tránsito genuino en una curva de luz, produciendo un falso positivo en la etapa de *vetting*. Estas fuentes se clasifican usualmente en dos familias: ruido instrumental y escenarios astrofísicos.

El ruido instrumental incluye anomalías de lectura térmica en los sensores CCD, problemas de enfoque del telescopio o el paso de rayos cósmicos. Por su parte, la variabilidad intrínseca de la estrella anfitriona (debida a procesos magnéticos como las manchas estelares en rotación) actúa como una fuente constante de ruido que puede generar disminuciones de flujo periódicas cuando la rotación de la estrella coincide con las escalas temporales de la búsqueda.

Dentro de los escenarios astrofísicos, la fuente dominante de falsos positivos corresponde a sistemas de estrellas Binarias Eclipsantes, formados por dos estrellas orbitando un centro de masa común en un plano orientado hacia la línea de visión de la Tierra. El eclipse primario se produce cuando la estrella de menor temperatura cruza frente a la estrella más brillante, generando una caída en el flujo percibido. Posteriormente, un segundo eclipse de menor profundidad ocurre cuando las posiciones se invierten, ocultando la estrella menos brillante detrás de la principal [Southworth, 2012]. La fotometría de este fenómeno se ilustra en la Figura 10.

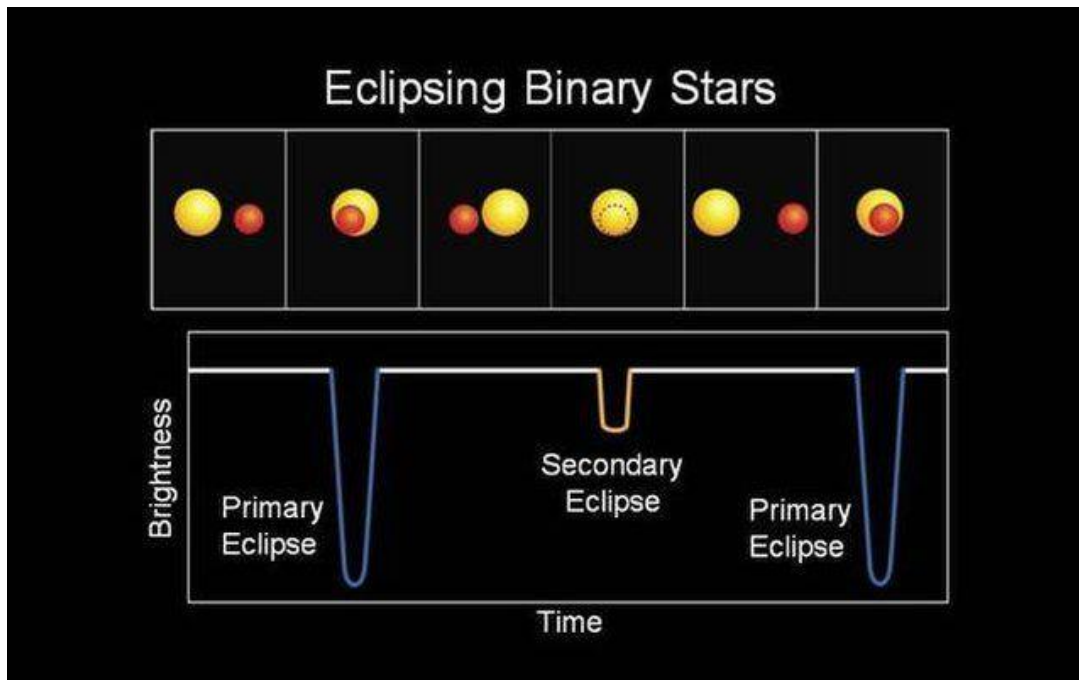


Figura 10: Fotometría de un sistema de estrellas binarias eclipsantes. Los eclipses primarios y secundarios producen bajas periódicas en el brillo percibido del sistema. Fuente: [NASA, 2013].

Existen variaciones como las binarias eclipsantes de fondo (*Background Eclipsing Binaries*), donde un par binario distante y tenue se encuentra visualmente alineado en el mismo sector del cielo que la estrella objetivo, combinando los flujos de luz en los mismos píxeles.

2.7 Clasificación de Estrellas

El rendimiento de un modelo de *vetting* se ve alterado por las propiedades físicas de la estrella que origina la curva de luz observada. Para catalogar de forma sistemática la población estelar analizada, el método de clasificación estelar moderno corresponde al sistema Morgan-Keenan (MK), que combina la clasificación del tipo espectral de Harvard y la clasificación por luminosidad propuesta por Edith Kellman. A continuación, describiremos brevemente estos sistemas.

2.7.1 Clases espectrales

El esquema unidimensional de Harvard clasifica a las estrellas en base a su temperatura efectiva superficial en un conjunto de 7 clases principales dispuestas en orden de temperatura descendente. Cada clase se identifica con una letra del alfabeto y posee una

cromaticidad distintiva, asociada al estado de ionización de los elementos en su fotosfera. La [Tabla 2](#) contiene información sobre las clases y divisiones.

Tabla 2: Clases espectrales principales del sistema de Harvard.

Clase espectral	Rango de temperatura superficial (K)	Cromaticidad
O	≥ 33.000	Azul
B	10.000 – 33.000	Azul-Blanca
A	7.300 – 10.000	Blanca
F	6.000 – 7.300	Amarilla-Blanca
G	5.300 – 6.000	Amarilla
K	3.900 – 5.300	Naranja
M	2.300 – 3.900	Roja

Un estudio realizado sobre las primeras observaciones del telescopio Kepler encontró una tendencia incremental de variabilidad fotométrica al disminuir la temperatura, principalmente debida a su mayor actividad magnética [\[Basri et al., 2010\]](#).

2.7.2 Clases de luminosidad y etapa evolutiva

Dado que estrellas con la misma temperatura superficial pueden presentar grandes diferencias en su tamaño y emisión de energía total, la clase de luminosidad actúa como una segunda dimensión en el sistema Morgan-Keenan. Esta clasificación segmenta a las estrellas según sus líneas de absorción espectral, relacionadas con la gravedad superficial de la estrella, un fenómeno estrechamente correlacionado con la gravedad superficial de la estrella ($\log g$).

Estas clases de luminosidad se denotan mediante números romanos y se asocian directamente a las etapas del ciclo evolutivo de una estrella. A medida que una estrella agota su suministro de hidrógeno en el núcleo, abandona la secuencia principal y se expande, alterando su clasificación de luminosidad. Un resumen simplificado de este esquema (omitiendo subdivisiones menores) se encuentra en la [Tabla 3](#).

Cabe destacar que las estrellas pertenecientes a la secuencia principal (Clase V) mantienen el hidrógeno ardiendo en su núcleo en un equilibrio entre la gravedad superficial, la presión del gas y la radiación, resultando en radios más compactos y mayores densidades superficiales comparadas a las gigantes evolutivas. Estas proporciones geométricas

impactan en la profundidad del tránsito δ , determinando qué tan detectable es la señal de un exoplaneta transitando a su alrededor.

Tabla 3: Clases de luminosidad estelares y sus etapas evolutivas asociadas.

Clase de luminosidad	Etapas evolutivas
I	Supergigantes
II	Gigantes brillantes
III	Gigantes
IV	Subgigantes
V	Secuencia principal

CAPÍTULO 3

PROPUESTA DE SOLUCIÓN

En este capítulo se detallan la metodología experimental y los componentes metodológicos de la solución propuesta. En primer lugar, se presenta una descripción cualitativa y cuantitativa del conjunto de datos empleado para entrenamiento y evaluación de los modelos. Posteriormente, se describe el preprocesamiento de las curvas de luz y series temporales de centroides. Finalmente, se exponen detalladamente las arquitecturas de aprendizaje profundo implementadas, junto con la configuración de su entrenamiento y sus respectivos hiperparámetros.

3.1 El conjunto de datos

Los datos utilizados en este trabajo proceden de la misión astronómica Kepler de la NASA. Lanzado en 2009, el telescopio espacial Kepler capturó observaciones fotométricas de alta precisión para más de 150.000 estrellas durante sus cuatro años de operación principal, con el propósito de detectar planetas del tamaño de la Tierra en la zona de habitabilidad de sus estrellas anfitrionas. Dado este enfoque, la selección de objetivos de la misión priorizó estrellas similares al Sol (clases espectrales F, G y K).

La recolección de observaciones de Kepler se estructuró en segmentos temporales de aproximadamente 90 días, denominados *quarters* o cuartos de observación. Al final de cada cuarto, el observatorio realizaba una maniobra de rotación para mantener sus paneles solares dirigidos al Sol. La misión completó un total de 17 cuartos (Q1-Q17). Dado que no todos los objetivos fueron observados de manera ininterrumpida durante el ciclo completo de la misión, la cobertura temporal y número de cuartos de observación disponibles varían entre estrellas. Adicionalmente, aunque algunos de los objetivos seleccionados resultaron ser sistemas estelares múltiples o binarias eclipsantes [Slawson et al., 2011], para los propósitos de este trabajo se asumirá que cada identificador único de Kepler (Kepler ID) representa a una estrella individual.

El conjunto de datos específico empleado en esta investigación corresponde al catálogo de *Threshold-Crossing Events (TCEs) Q1-Q17 Data Release 24* (NASA Exoplanet Science Institute, s.f.). Un TCE se define como una atenuación periódica en el flujo de una estrella, consistente con la marca de un exoplaneta transitando a su alrededor [Coughlin et al., 2016]. Cada TCE está asociado a una estrella objetivo, cuyas curvas de luz y datos crudos se encuentran disponibles en el portal Mikulski Archive for Space Telescopes (MAST).

3.1.1 Parámetros de tránsito y propiedades estelares

De las más de 100 columnas del catálogo Q1-Q17 *TCE Data Release 24*, se han conservado siete: los identificadores de la estrella y el TCE, sus parámetros principales (período, duración, época y profundidad del tránsito) y la etiqueta que clasifica candidatos a planeta y no candidatos. Estas etiquetas fueron producidas combinando las disposiciones determinadas por el *Threshold Crossing Event Review Team* (integrado por científicos de la misión Kepler) con otros diagnósticos, clasificando cada TCE como candidato a planeta (CP), falso positivo astrofísico o fenómeno no transitorio. El detalle se puede encontrar en [Catanzarite, 2015]. Para manejar la tarea como un problema de clasificación binaria, se unificaron las etiquetas de falso positivo astrofísico y fenómeno no transitorio en NCP (no candidato a planeta). Además, se descartaron 4630 TCEs con la etiqueta *unknown*.

Los datos fueron cruzados con el Catálogo de Propiedades Estelares Gaia-Kepler [Berger et al., 2015] que provee estimaciones de alta calidad de las propiedades físicas de cada estrella. Seis parámetros estelares fueron incorporados: temperatura efectiva, densidad, radio, gravedad superficial, masa y metalicidad. Durante el proceso de enlazar ambos conjuntos se eliminaron 1,162 filas que no contaban con datos disponibles, cerca de 7.3% del total.

Dos curvas de luz se descartaron durante el preprocesamiento por inconsistencias insalvables en sus vectores de desplazamiento de centroides. Finalmente, el conjunto de datos contiene 14,573 TCEs, de los cuales 11,157 están etiquetados como NCP y 3,416 como CP, con una razón de casi 3.3 negativos a positivos. Los TCEs pertenecen a un total de 9,153 estrellas objetivo.

El detalle de las columnas del conjunto de datos final se encuentra en el Anexo A.

3.2 Preprocesamiento de los datos

Las series temporales descargadas desde el portal MAST para cada estrella objetivo se encuentran segmentadas por cuartos de observación. Estas curvas representan el producto del *pipeline* de procesamiento científico de Kepler [Jenkins et al., 2010] que se encarga de calibrar los píxeles, realizar la fotometría de apertura y corregir errores sistemáticos instrumentales, entregando mediciones fotométricas regulares con una cadencia temporal de 29.4 minutos.

Para utilizar estas series de tiempo como entrada en los modelos de aprendizaje profundo, es necesario ejecutar algunos pasos adicionales de preprocesamiento. El flujo presentado a continuación fue propuesto por los autores de la red convolucional AstroNet [Shallue y Vanderburg, 2018] y extendido posteriormente con ligeras modificaciones [Ansdell et al., 2018] para la arquitectura ExoNet.

3.2.1 Curvas de luz

El procesamiento fotométrico inicia con la remoción de la variabilidad de baja frecuencia presente en las curvas. Para lograr esto sin deformar la señal de interés, se enmascaran temporalmente las observaciones que coinciden con todos los TCEs conocidos de la estrella y se interpolan linealmente. Sobre esta curva enmascarada, se ajusta iterativamente una *B-spline*¹ cúbica variando el espaciado entre nodos y calculando el criterio de información bayesiano (BIC por sus siglas en inglés) en toda la curva. El espaciado variable permite ajustar *B-splines* a curvas de luz con distinto nivel de variabilidad intrínseca.

Una vez obtenida la mejor B-spline, que minimiza el BIC, el flujo original se divide por ésta para aplanar la serie temporal. En seguida se descartan los valores atípicos que exceden 3 desviaciones estándar interpolándolos linealmente y el ajuste de la *spline* se repite. Este ciclo se itera hasta que no se encuentren nuevos valores atípicos, o se alcance el límite de iteraciones. Finalmente, los segmentos aplanados de todos los cuartos se concatenan para obtener la curva completa. Esta serie continua se pliega en fase (*phase-folding*) alrededor del período del TCE, de manera que los eventos de tránsito quedan sobrepuestos y alineados en el centro del vector temporal.

La [Figura 11](#) ilustra este flujo aplicado al objetivo Kepler ID 9388479 durante su décimo cuarto de observación. En la curva de luz aplanada se pueden identificar disminuciones periódicas en el flujo alcanzando distintas profundidades; en efecto, esta estrella cuenta con dos TCEs identificados como 9388479-1 y 9388479-2.

¹ Una B-Spline es una función polinómica definida por tramos, definidos mediante un vector de nodos (puntos de separación).

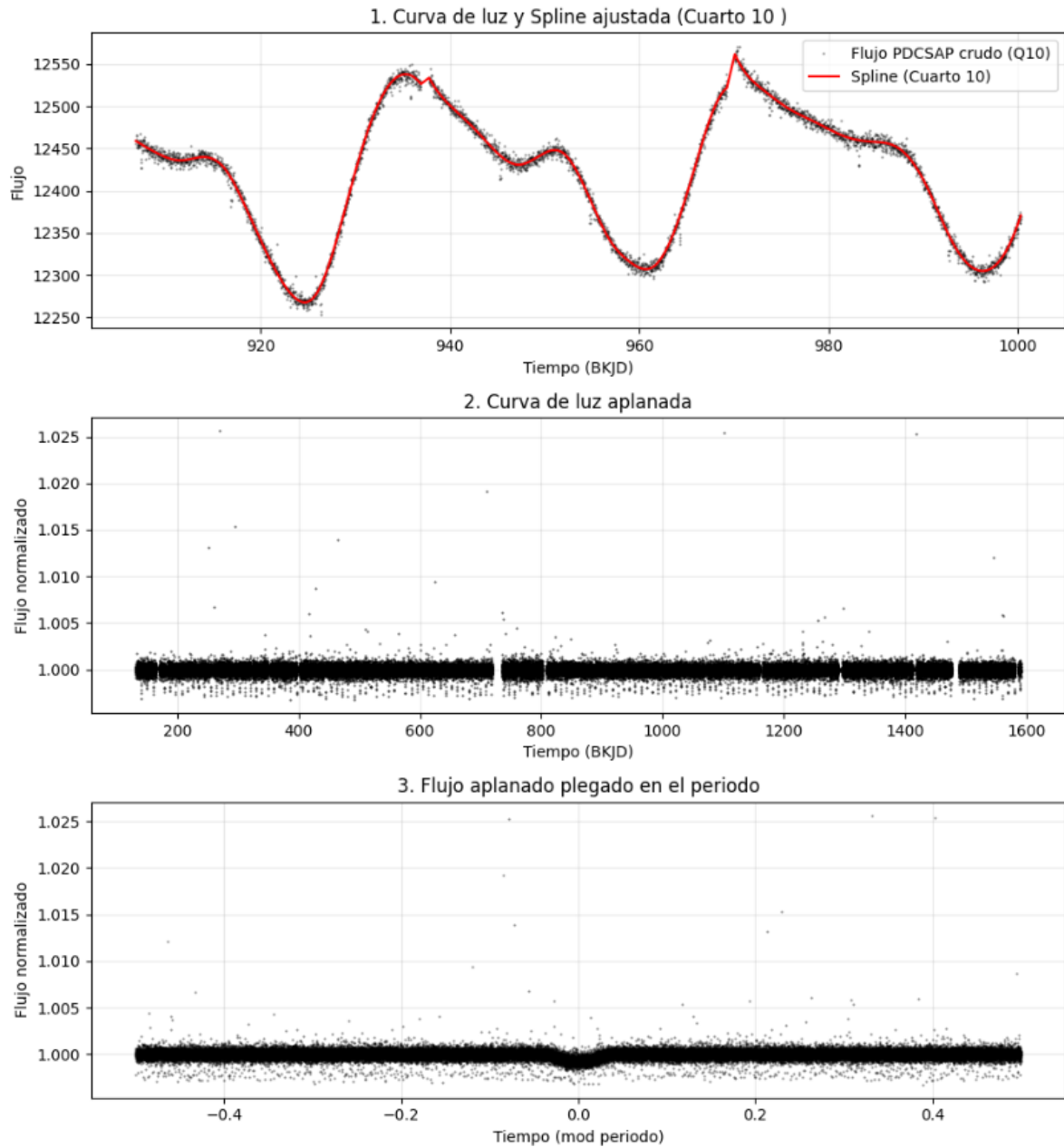


Figura 11: Primera etapa del preprocesamiento de la curva de luz de la estrella con Kepid 9388479. (1) Décimo cuarto de la curva de luz junto a la Spline ajustada. (2) Curva de luz luego de dividir su flujo por el valor de la Spline. (3) Curva de luz aplanada plegada sobre el período del TCE 9388479-1.

Fuente: Elaboración propia.

A partir de esta curva aplanada y plegada se generan dos arreglos o vistas. Los puntos son agrupados en contenedores (*bins*) de ancho δ uniforme y distancia λ entre puntos medios, asignando a cada intervalo la media aritmética de los flujos de luz que contiene.

La vista global abarca la totalidad de la serie temporal y la comprime en un vector de longitud fija de 2001 *bins*. La vista local considera un entorno temporal acotado alrededor del centro del tránsito, equivalente a 8 veces su duración, y agrupa dichos puntos en 201 *bins*, capturando la morfología detallada del tránsito. En ambos casos los vectores son escalados para tener media 0 y profundidad -1. La [Figura 12](#) ejemplifica el resultado de ambas vistas.

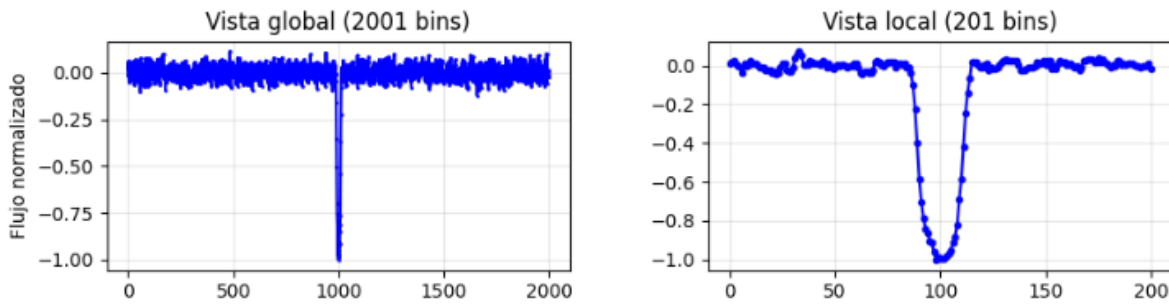


Figura 12: Vistas global y local del TCE 9388479-1, compuestas por 2001 y 201 puntos respectivamente.

Fuente: Elaboración propia.

3.2.2 Centroides

Basado en la arquitectura de ExoNet [\[Ansdell et al., 2018\]](#), se integran las series temporales de los centroides como una entrada adicional. Los centroides registran la variabilidad posicional del objetivo en la imagen capturada por el CCD. Las alteraciones abruptas en el centroide durante un evento de tránsito son indicadores de falsos positivos astrofísicos, pues sugieren que la pérdida de luz no proviene de la estrella objetivo, sino de una binaria eclipsante contigua.

Para procesar esta información, se extraen las coordenadas (x,y) y se calcula la magnitud del desplazamiento $r = \sqrt{(x^2 + y^2)}$. Esta nueva serie de tiempo se somete al mismo proceso de aplanamiento mediante *B-splines*, plegado en fase y agrupamiento en vistas detallado para las curvas de luz. No obstante, la estrategia de normalización es distinta: en lugar de escalar los vectores de los centroides para otorgarles una profundidad fija, se normalizan utilizando la media y la desviación estándar global computadas exclusivamente fuera de los tránsitos para todo el conjunto de entrenamiento. Además, la desviación estándar del centroide se escala con la desviación estándar del flujo de luz, asegurando que las curvas de luz altamente estables estén acompañadas por vectores espaciales

proporcionalmente planos [Ansdell et al., 2018]. El resultado queda ejemplificado en la Figura 13.

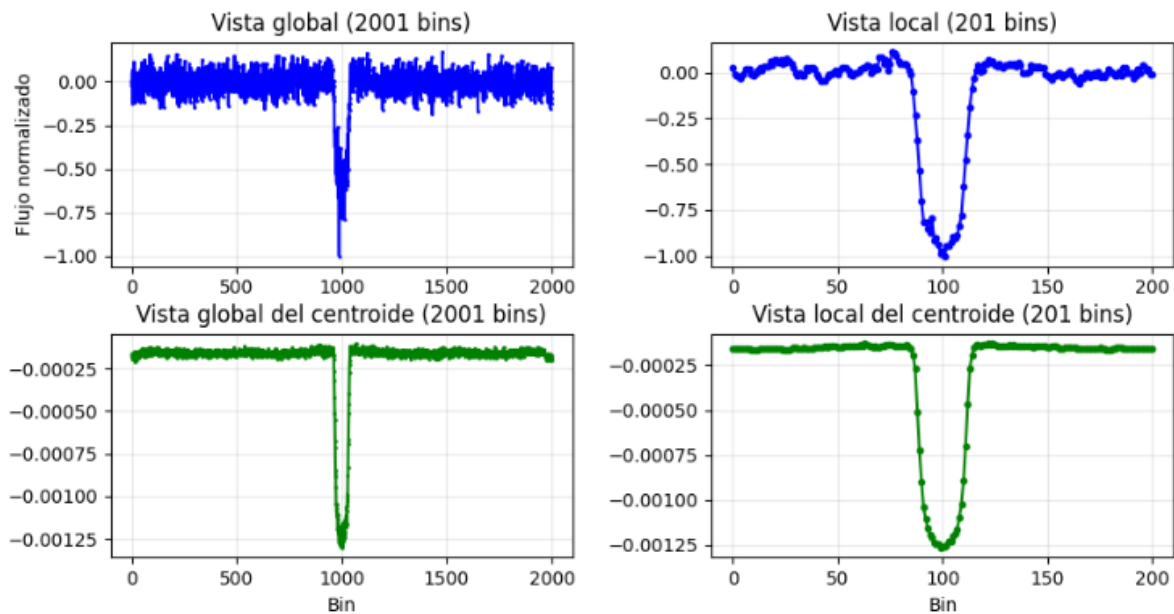


Figura 13: Vistas global y local de las curvas de luz (azul) y centroides (verde). Los ejemplos corresponden al TCE 2997178-1, provocado por efecto de otra estrella captada por el CCD. Fuente: Elaboración propia.

3.2.3 Parámetros estelares

Los parámetros físicos de la estrella anfitriona son normalizados para otorgarles media cero y varianza unitaria. Para prevenir la filtración de información (*data leakage*) durante los experimentos, la media y desviación estándar se calculan exclusivamente sobre la población del conjunto de entrenamiento.

3.3 Análisis exploratorio

La Tabla 4 desglosa la distribución del catálogo según su clase espectral. El perfil del conjunto de datos es consistente con el sesgo observacional de la misión Kepler. Las clases F y G dominan la muestra, con 71,6% de las observaciones. Las estrellas más frías (tipos K y M) agrupan el 16,9% del catálogo, mientras que las estrellas masivas de alta temperatura (tipos A y B) constituyen el 11,5% restante. Se puede identificar un desbalance severo en los extremos del espectro térmico: por ejemplo, la clase B aporta 147 TCEs al conjunto, de los cuales todos fueron clasificados como no candidatos.

Tabla 4: Distribución de las estrellas y TCEs del conjunto por clase espectral.

Clase espectral	Número de estrellas	Porcentaje	Número de TCEs	Candidatos a planeta	No candidatos	Razón NCP a CP
O	-	-	-	-	-	-
B	69	0.75	147	0	147	-
A	985	10.75	2065	13	2052	157.7
F	3490	38.13	6101	773	5328	6.9
G	3065	33.49	4153	1730	2423	1.4
K	1472	16.09	2020	877	1143	1.3
M	72	0.79	87	23	64	2.8

Por otro lado, la Tabla 5 contiene la distribución según estado evolutivo. El estado evolutivo de cada estrella fue designado utilizando la librería *evolstate* [Huber, 2019] que en base a los parámetros de temperatura, radio y gravedad superficial distingue entre tres categorías de estrellas: gigantes (clases I a III), subgigantes (clase IV) y secuencia principal (clase V). La gran mayoría de las estrellas anfitrionas (79.2%) pertenecen a la secuencia principal, seguidas por una fracción considerable de estrellas subgigantes (16.0%) y una minoría marginal de gigantes (4.8%).

Tabla 5: Distribución de las estrellas y TCEs del conjunto por etapa evolutiva.

Estado evolutivo	Cantidad de estrellas	Porcentaje	Cantidad de TCEs	Candidatos a planeta	No candidatos	Proporción NPC a PC
Secuencia principal (clase V)	7248	79.19	11889	2909	8980	3.1
Subgigantes (clase IV)	1467	16.02	2111	470	1641	3.5
Gigantes (clases I-III)	438	4.79	573	37	536	14.5

La visualización de los valores promedio de las series temporales normalizadas se presentan en la Figura 14, mostrando marcadas diferencias entre las clases positiva y negativa. En ambas vistas se observa que la señal agrupada de los candidatos a planeta forma una línea relativamente estable fuera de la zona de eclipse. Los no-candidatos exhiben una variabilidad sustancialmente mayor, y sus tránsitos exhiben perfiles asimétricos o con forma de “V”, característica de las ocultaciones en sistemas binarios eclipsantes.

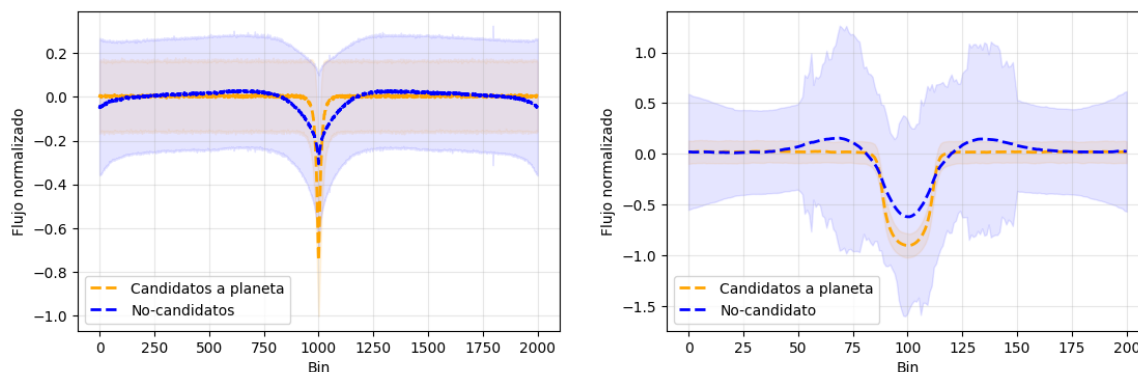


Figura 14: Valores promedio para las vistas global (izquierda) y local (derecha). Las líneas discontinuas indican la media para las clases positiva y negativa, en colores naranja y azul, respectivamente. Las bandas semi traslúcidas corresponden a la desviación estándar de cada valor. Fuente: Elaboración propia.

3.4 Modelos de aprendizaje profundo

En esta sección se describen las tres arquitecturas de redes neuronales para abordar la clasificación de las series de tiempo y la configuración experimental utilizada para su entrenamiento y validación.

3.4.1 AstroNet

AstroNet [Shallue y Vanderburg, 2018] corresponde a una red neuronal convolucional que utiliza las vistas global y local como entradas. Cada entrada es procesada por un bloque de capas convolucionales y de *pooling*, aumentando progresivamente el número de mapas de características para extraer representaciones de la morfología fotométrica. Las características finales extraídas por ambos bloques se concatenan conformando un único vector, el cual ingresa a una red neuronal completamente conectada de cinco capas encargada de realizar la clasificación final. Todas las capas ocultas emplean la función de activación ReLU.

Para mitigar el sobreajuste, el proceso de entrenamiento incorpora un método de aumento de datos, aplicando reflexiones horizontales a las curvas de luz con una probabilidad aleatoria de 0.5. Esta arquitectura, cuyo diagrama completo se aprecia en la Figura 15, servirá como modelo base para las comparaciones, dado que los modelos siguientes representan modificaciones y adiciones a esta arquitectura.



como FC-<número de neuronas>. Fuente: [Shallue y Vanderburg, 2018].

3.4.2 ExoNet

ExoNet es una mejora a AstroNet propuesta en [Ansdell et al., 2018]. La novedad corresponde a la inyección de conocimiento de dominio astrofísico. La arquitectura preserva la extracción de características basada en vistas globales y locales, pero integra dos nuevas entradas a la red: parámetros estelares e información de los centroides de la curva de luz. Las curvas de centroides son ingresadas a los bloques de CNN como un segundo canal en la capa de entrada. Los parámetros estelares (masa, temperatura efectiva, gravedad superficial, metalicidad, radio y densidad media) son ingresados a la red al concatenar los resultados de las capas convolucionales, enriqueciendo el contexto de la red completamente conectada que realiza las clasificaciones. Al igual que AstroNet, emplea la función ReLU en sus capas ocultas.

Los autores mantuvieron el uso de reflexiones horizontales para aumentar los datos, e introdujeron la adición de ruido gaussiano aleatorio en las curvas durante el entrenamiento como táctica adicional de aumento de datos. En este trabajo se optó por omitir este nuevo método a fin de mejorar la comparabilidad entre modelos.

El esquema de la arquitectura ExoNet se puede observar en la Figura 16.

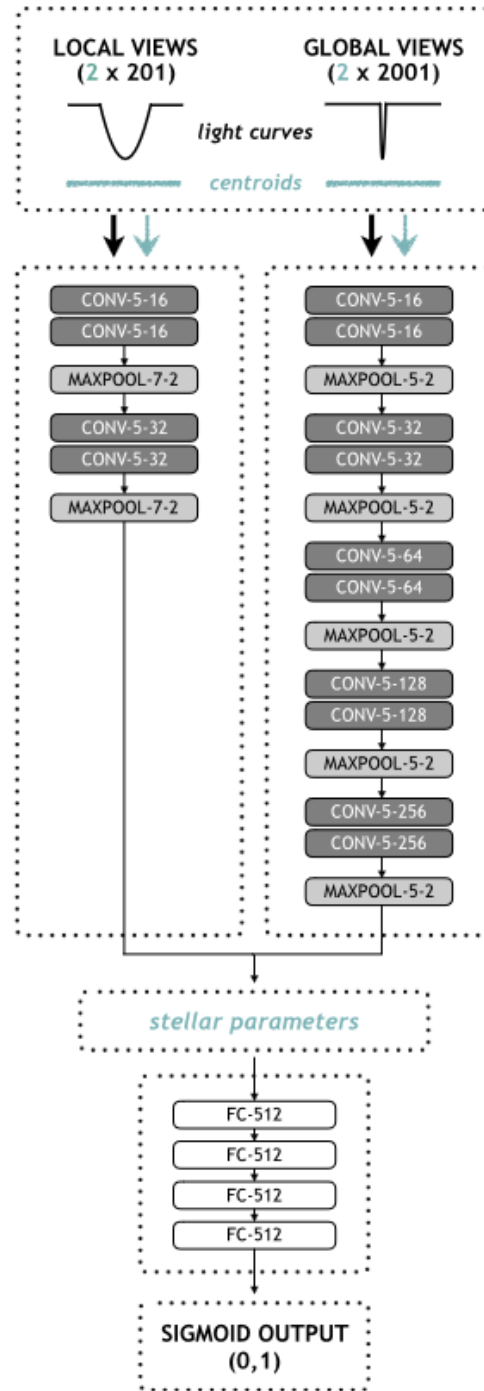


Figura 16: Diagrama de la arquitectura de ExoNet. Las capas convolucionales son denotadas como CONV-
<tamaño del kernel>-<número de FMs>, las capas de *pooling* como MAXPOOL-<largo del kernel>-<stride> y
las capas FC como FC-<número de neuronas>. Fuente: [Ansdell et al., 2018].

3.4.3 Modelo híbrido de CNN con LSTM y atención aditiva

Introducido recientemente en [Hernández-Carnerero et al., 2025], este modelo evalúa el impacto de modelar las dependencias secuenciales del tránsito mediante el acoplamiento de celdas LSTM y mecanismos de atención. Dado que procesar una curva de luz completa mediante redes recurrentes resulta poco eficiente, los autores retuvieron las vistas locales y globales, aplicando bloques convolucionales al comienzo de la red para extraer sus características. La innovación se introduce después de los bloques convolucionales, procesando los mapas finales con células LSTM.

Tras ensayos empíricos con múltiples configuraciones de arquitectura, encontraron que el modelo con mayor rendimiento para el conjunto de Kepler incorpora tanto capas LSTM como el mecanismo de atención. El bloque de la vista global consiste en tres capas LSTM con 500, 200 y 100 células de memoria, respectivamente, mientras que la vista local contiene tres capas con 200, 100 y 50 células. Tanto las capas convolucionales como densas utilizan la función de activación ReLU, mientras que las capas LSTM utilizan la función tangente hiperbólica. Esta arquitectura se puede observar en la [Figura 17].

El mecanismo de atención utilizado es una versión adaptada de la atención de Bahdanau. Debido a que la tarea de *vetting* corresponde a la inferencia de una única etiqueta y no a la generación de una secuencia, se descartó el componente decodificador, reduciendo las operaciones matriciales.

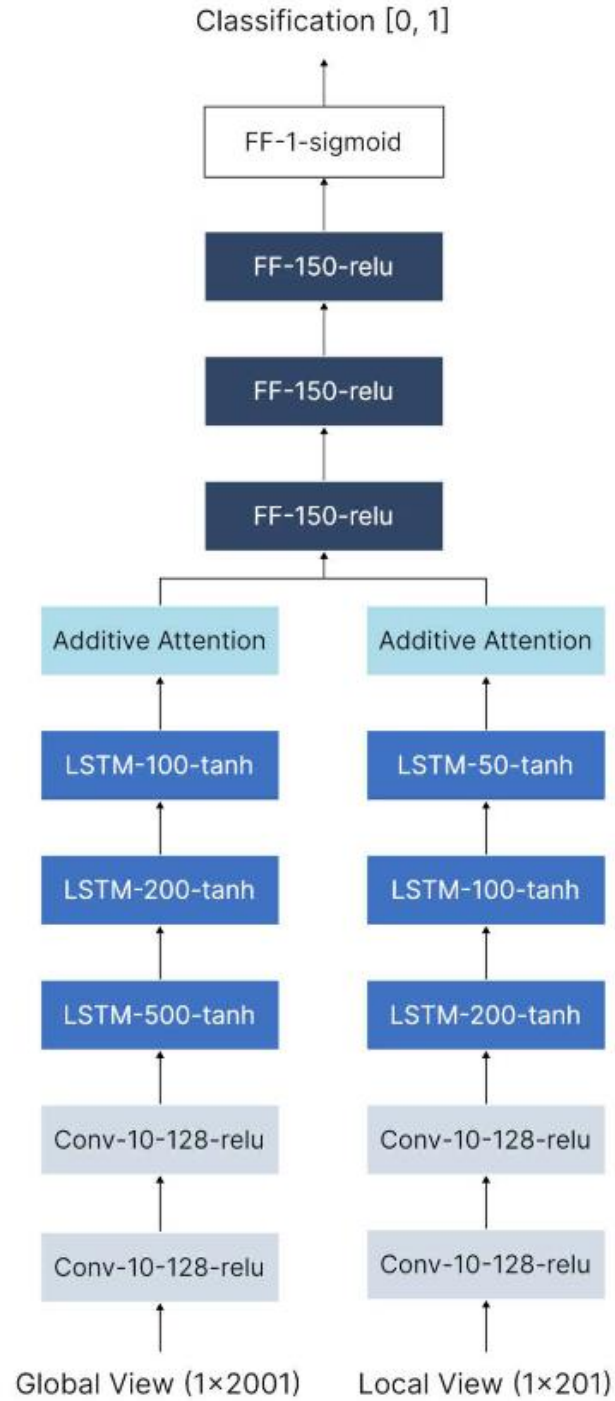


Figura 17: Diagrama de la arquitectura del modelo híbrido. Las capas convolucionales son denotadas como Conv-<tamaño del kernel>-<número de FM>-<activación>, las capas LSTM como LSTM-<número de células>-<activación>. Fuente: [Hernández-Carnerero et al., 2025].

3.4.4 Entrenamiento e hiperparámetros

Para el entrenamiento y la evaluación posterior de los clasificadores, el conjunto de datos fue particionado en subconjuntos de entrenamiento (*train*), validación (*validation*) y prueba (*test*). La segmentación se organizó agrupando los identificadores de las estrellas (Kepid). Esta precaución asegura que las señales de tránsito pertenecientes a un mismo sistema queden alojadas en el mismo subconjunto, evitando que los algoritmos reconozcan variabilidades propias de la estrella al momento de realizar inferencia sobre el segmento de pruebas. La distribución de los TCEs, desglosados por clase espectral y estado evolutivo de la estrella anfitriona se encuentra en la [Tabla 6](#).

Para realizar las predicciones, se entrenaron 10 instancias de cada modelo con el propósito de reportar las métricas de evaluación mediante la media y desviación estándar. Esta estrategia busca considerar la variabilidad producida por la inicialización aleatoria de los parámetros.

Tabla 6: Distribución de TCEs en los conjuntos de entrenamiento, evaluación y pruebas.

	Clase espectral						Etapa evolutiva		
	B	A	F	G	K	M	Secuencia principal	Subgigantes	Gigantes
Train	120	1617	4910	3314	1650	71	9502	1713	467
Val	18	245	577	398	200	5	1185	209	49
Test	9	203	614	441	170	11	1202	189	57

Todos los modelos emplean la función de Entropía Cruzada Binaria como función de pérdida durante el entrenamiento. Los hiperparámetros utilizados corresponden a los reportados por sus respectivos autores, y se encuentran en la [Tabla 7](#). Excepcionalmente, el modelo híbrido utiliza una tasa de aprendizaje variable; igual a $1e-4$ durante las primeras 25 épocas de entrenamiento, y reducido a $1e-5$ durante las 25 épocas restantes. Para los modelos AstroNet y ExoNet se implementó aumento de datos en el proceso de entrenamiento mediante reflexiones horizontales a las curvas de flujo y desplazamiento de centroides, con una probabilidad aleatoria de 0.5.

Tabla 7: Resumen de hiperparámetros utilizados para entrenamiento.

Número de épocas	Tamaño de lote	Tasa de aprendizaje	Parámetros de optimizador Adam
50	64	$1e-5$	$\beta_1=0.9$, $\beta_2=0.999$, $\epsilon=1e-8$

CAPÍTULO 4

VALIDACIÓN DE LA SOLUCIÓN

4.1 Tiempos de entrenamiento e inferencia

Los modelos fueron implementados utilizando la librería PyTorch y entrenados en una unidad de procesamiento gráfico *NVIDIA RTX 3070* con 8 GB de memoria VRAM. AstroNet y ExoNet cuentan con cerca de 10 millones de parámetros, mientras que el modelo híbrido posee cerca de 3 millones de parámetros, debido a la cantidad reducida de capas convolucionales.

En la [Tabla 8](#) se presentan la media aritmética de los tiempos de entrenamiento e inferencia de cada modelo. Al medir los tiempos de inferencia se descartaron las etapas de carga de datos a la GPU. Los modelos AstroNet y ExoNet tienen períodos de entrenamiento acotados, con tiempos totales cercanos a 4 y 5 minutos respectivamente. Por otro lado, el modelo híbrido necesita una cantidad de tiempo más de 10 veces mayor por época, resultando en un tiempo de ejecución de aproximadamente 50 minutos para entrenar el modelo.

En la etapa de inferencia, AstroNet y ExoNet consiguen un muy alto rendimiento, procesando cerca de 3000 y 2000 secuencias de entrada por segundo, respectivamente. El modelo híbrido requiere más tiempo para esta tarea, consiguiendo procesar cerca de 500 secuencias por segundo. Aunque el tiempo total sigue siendo razonable para la red recurrente, esta diferencia sí puede ser significativa en caso de realizar inferencia sobre grandes conjuntos de datos.

Tabla 8: Número de parámetros y tiempos de entrenamiento e inferencia para cada modelo.

Modelo	Tiempo medio de entrenamiento por época (s)	Tiempo medio de inferencia por entrada (ms)
AstroNet	4.86	0.32
ExoNet	5.72	0.42
Modelo híbrido	60.2	2.11

4.2 Resultados generales

Las métricas de rendimiento medio de cada modelo, evaluadas sobre el conjunto de pruebas completo, se detallan en la Tabla 9. En general, las tres arquitecturas exhiben un desempeño altamente competitivo con márgenes estrechos de diferencia. El F1-Score muestra una tendencia progresiva, donde la red híbrida alcanza el mejor desempeño medio (0.902 ± 0.016) posicionándose levemente por encima de los otros modelos. Evaluando la precisión (*precision*), que indica la fracción de predicciones positivas que resultaron correctas, todos los modelos rondan una media de 0.89, mostrando una sólida capacidad de rechazar falsas alarmas, con el modelo híbrido obteniendo el mejor desempeño (0.890 ± 0.010).

En el caso de la sensibilidad (*recall*), la red híbrida logra una mejora de 0.01 puntos en su media respecto a AstroNet (0.914 vs 0.904), lo que en términos prácticos indica que es capaz de detectar un tránsito planetario genuino adicional entre cada 100 evaluados. El área bajo la curva ROC (AUC) es casi idéntica en cada caso, con menos de un 0.2% de diferencia entre cada par de modelos, atestiguando la capacidad general de separación de clases de las tres arquitecturas.

Tabla 9: Métricas de evaluación de los modelos sobre el conjunto de prueba completo.

Modelo	F1-Score	Precision	Recall	AUC
AstroNet	0.895 ± 0.007	0.885 ± 0.009	0.904 ± 0.018	0.986 ± 0.001
ExoNet	0.898 ± 0.007	0.887 ± 0.009	0.910 ± 0.012	0.987 ± 0.001
Híbrido	0.902 ± 0.016	0.890 ± 0.010	0.914 ± 0.024	0.985 ± 0.003

A pesar de que el modelo híbrido presenta las medias más altas en F1-Score y sensibilidad, se debe destacar el traslape estadístico entre los resultados. La diferencia de F1-Score entre el modelo híbrido y AstroNet es de sólo 0.007, un margen inferior a la desviación estándar del modelo híbrido (± 0.016). Esto indica que un modelo puramente convolucional bien inicializado puede obtener resultados iguales a la red híbrida.

La baja dispersión en todas las métricas (reflejada en la desviación estándar) de AstroNet y ExoNet demuestra que son arquitecturas altamente estables. El modelo híbrido, en contraste, exhibe una mayor dispersión. Esto sugiere que la introducción de mecanismos más complejos genera una superficie de la función de pérdida más compleja de optimizar. Por otro lado, también sugiere que con una optimización de hiperparámetros exhaustiva el desempeño de la red híbrida puede ser considerablemente superior al modelo base.

4.3 Resultados según clase espectral

A continuación, se comparan las métricas de evaluación obtenidas por cada modelo al segmentar las estrellas del conjunto de prueba por su clase espectral. Las tablas 10, 11 y 12 del anexo contienen el detalle de estas métricas. Los resultados se presentan mediante gráficos de barras con barras de error. Se han omitido las clases B y M debido a su bajo número de TCEs en el conjunto de prueba (9 y 11 respectivamente), y la clase A debido a su baja cantidad de TCEs positivos (apenas 3 casos). El desempeño de los algoritmos sobre estas categorías será discutido más adelante.

4.3.1 Resultados en las clases espectrales F, G y K

La Figura 18 muestra los valores medios y desviación estándar de la métrica F1-Score. Se puede observar que el mejor valor para las tres arquitecturas se ubica en las estrellas de tipo G (análogas al Sol), en las cuales todos los modelos superan un desempeño medio de 0.910 en esta métrica, con el modelo híbrido alcanzando el máximo de 0.923 ± 0.026 . Las otras clases F y K presentan ligeras disminuciones entre 0.03 y 0.05 en cada modelo.

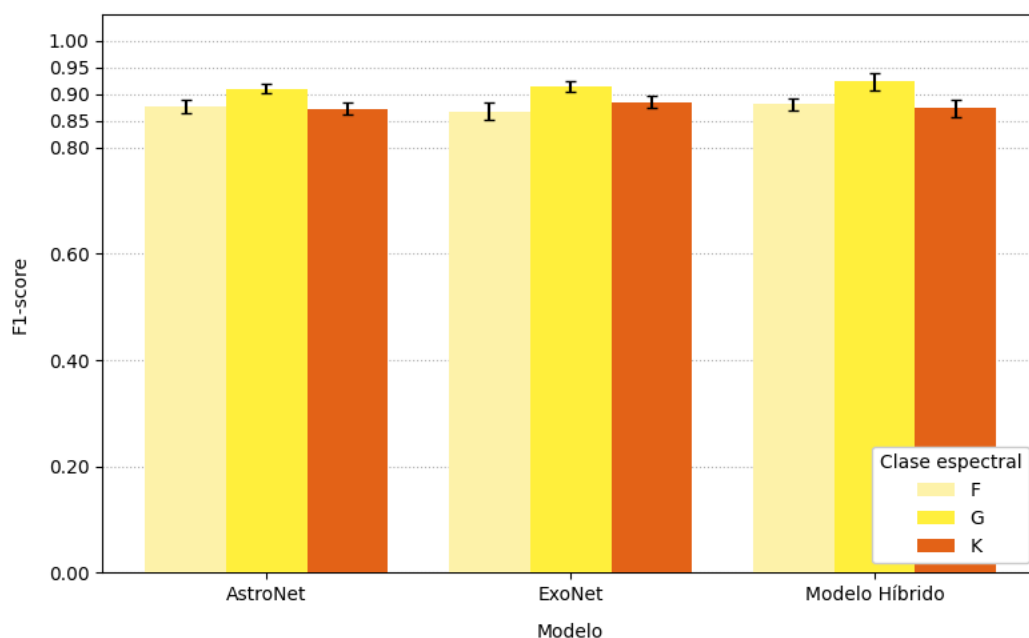


Figura 18: F1-Score de cada modelo según la clase espectral de la estrella anfitriona. Fuente: Elaboración propia.

En el análisis de *recall* entre las distintas clases, tanto AstroNet como el modelo híbrido muestran una tendencia a la baja a medida que disminuye la temperatura de la estrella anfitriona, observable en la [Figura 19](#). El modelo híbrido consigue el valor medio más alto para las estrellas de la clase F (0.930), pero también sufre la mayor caída, resultando en un *recall* reducido a 0.924 y 0.875 en las clases G y K, respectivamente. Esto sugiere que las capas LSTM combinadas con atención son más efectivas para modelar las disminuciones sutiles en el flujo que el modelo base, pero tiende a fallar al clasificar señales de estrellas frías que poseen mayor ruido estelar debido a su variabilidad.

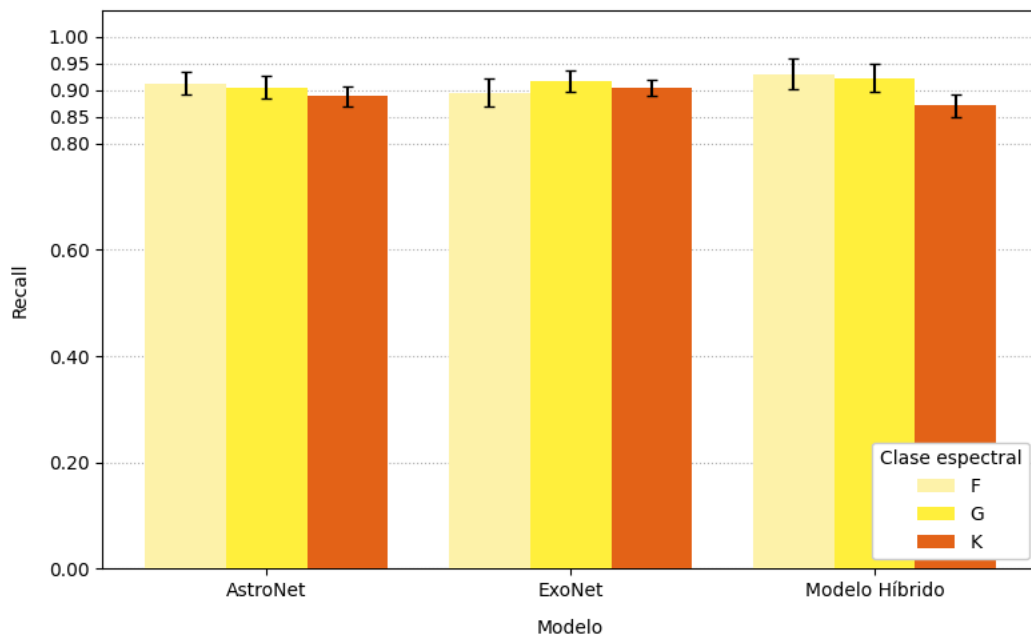


Figura 19: *Recall* de cada modelo según la clase espectral de la estrella anfitriona. Fuente: Elaboración propia.

Llama la atención que ExoNet obtuvo un excelente *recall* en la clase G, pero peor que los otros modelos en la clase F, lo que sugiere que las entradas adicionales ayudan al modelo en algunas clases y lo perjudican en otras. Para ahondar en esta situación, se graficaron los valores promedio y la desviación estándar para las vistas globales de desplazamiento de los centroides en los TCEs de estrellas en las clases F y G. El resultado se puede observar en la [Figura 20](#). La clase G presenta una alta varianza en los valores esta vista, mientras que en la clase F la serie es estable y se pueden identificar fácilmente las desviaciones producidas por estrellas binarias eclipsantes. El origen de esta discrepancia no es evidente. Es posible que las series de alta varianza en la clase G ayuden a evitar el sobreajuste del modelo, forzando a las capas convolucionales a extraer dependencias más robustas. Otra posibilidad consiste en que la mayor razón de casos negativos a positivos de la clase F (6.9, comparado a 1.5 de

la clase G) conduzca a ExoNet a predecir más etiquetas negativas en estrellas de este rango de temperatura.

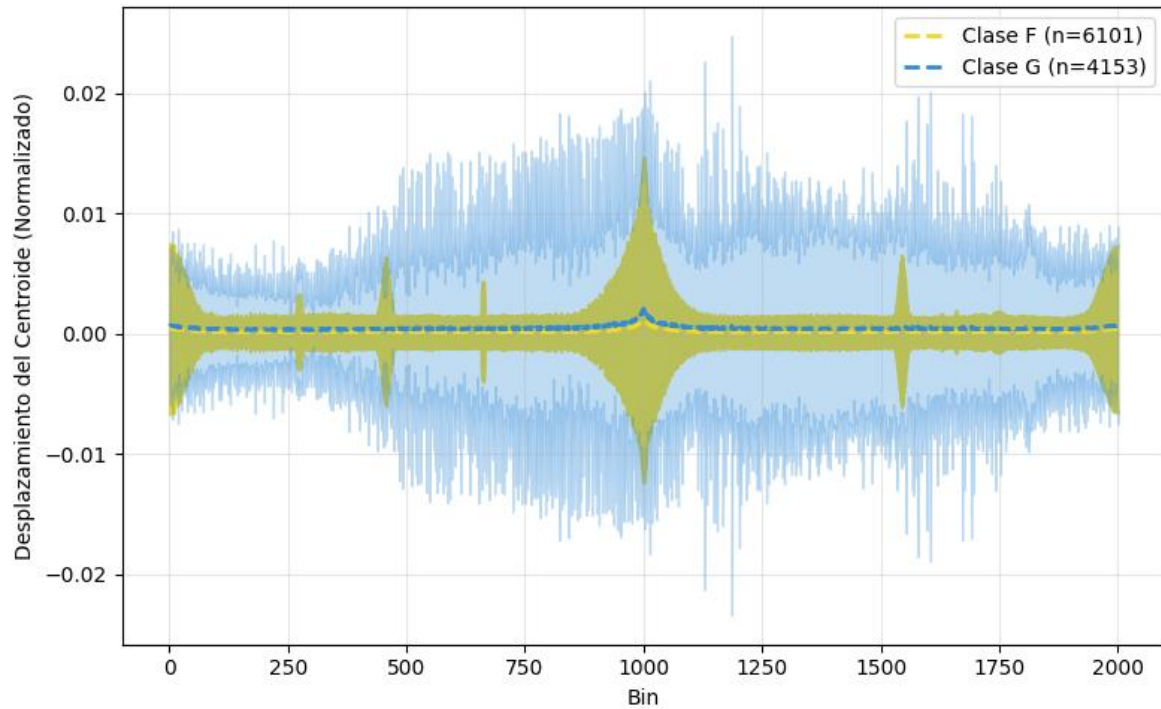


Figura 20: Valores promedio y desviación estándar en las vistas globales para los vectores de desplazamiento de los centroides. Las líneas discontinuas indican la media para TCEs en estrellas de clases F y G, en colores amarillo y azul, respectivamente. Las bandas amplias corresponden a la desviación estándar de estos valores en cada punto. Fuente: Elaboración propia.

La tasa de falsos positivos de cada modelo en estrellas de las distintas clases espectrales se muestra en la [Figura 21](#). Se puede observar que en las estrellas de tipo F todos los modelos son excepcionalmente buenos rechazando el ruido, manteniendo una TFP muy baja (aproximadamente 0.025). Sin embargo, la TFP se dispara en las estrellas de tipo K en todas las arquitecturas, superando un valor medio de 0.10. Este mayor aumento de falsos positivos se condice con la mayor variabilidad esperada en estrellas de menor temperatura, que reduce la relación señal-ruido de las observaciones.

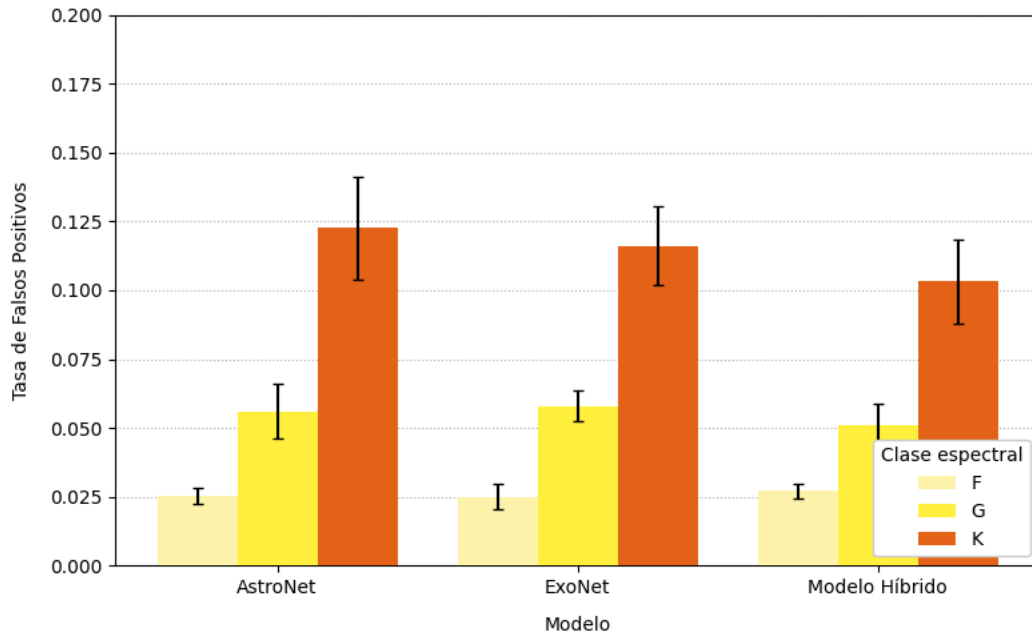


Figura 21: Tasa de falsos positivos de cada modelo según la clase espectral de la estrella anfitriona. Fuente: Elaboración propia.

4.3.2 Resultados en las clases espectrales B, A y M

La clase espectral B carece completamente de TCEs etiquetados como candidatos a planeta en el conjunto de datos. Ante este escenario, todos los modelos evaluados asignaron correctamente la clase negativa a la totalidad de las muestras. Por otro lado, en las clases A y M, el conjunto de pruebas contiene un número marginal de TCEs positivos (3 y 2 respectivamente), correspondientes a los identificadores 5022828-1, 8056665-1, 8056665-2, 9388479-1 y 9388479-2. En una inspección visual de sus curvas de luz ya procesadas se identifican claramente los tránsitos. Más aún, estos eventos se encuentran confirmados como exoplanetas en el catálogo de *Kepler Objects of Interest DR25*.

Todos los modelos clasificaron correctamente estos TCEs. En particular, ya que ExoNet logró identificarlos como positivos con una alta confianza a pesar de pertenecer a una clase espectral minoritaria y dominada por TCEs de la clase negativa, se puede concluir que el modelo es robusto ante el desbalance de clases: cuando la señal fotométrica es clara, la red neuronal prioriza las características morfológicas de la curva de luz por sobre el sesgo estadístico asociado al tipo de estrella.

4.4 Resultados según etapa evolutiva

En esta sección se comparan las métricas de evaluación obtenidas por cada modelo al segmentar las estrellas del conjunto según su etapa evolutiva. El detalle de esta evaluación se puede encontrar en las tablas 13, 14 y 15 del anexo. Se debe considerar que la categoría de estrellas Gigantes ($n = 57$) contiene apenas 3 muestras pertenecientes a la clase positiva (candidatos a planeta), por lo que la relevancia estadística de las métricas de evaluación en esta categoría es menor.

Los resultados exhiben una degradación del rendimiento en la métrica F1-Score a medida que las estrellas evolucionan fuera de la secuencia principal, observable en la Figura 22. En la secuencia principal, el modelo híbrido consigue el mejor desempeño general con un F1-Score de 0.914 ± 0.015 . Al evaluar curvas de luz de subgigantes, el desempeño sufre una caída en esta métrica de aproximadamente 0.06 a 0.07 en todos los modelos. En el segmento de gigantes, se encontró un deterioro muy marcado del desempeño en cada caso, con valores medios de 0.443 para AstroNet, 0.397 para el modelo híbrido y 0.100 para ExoNet (con una muy alta desviación estándar).

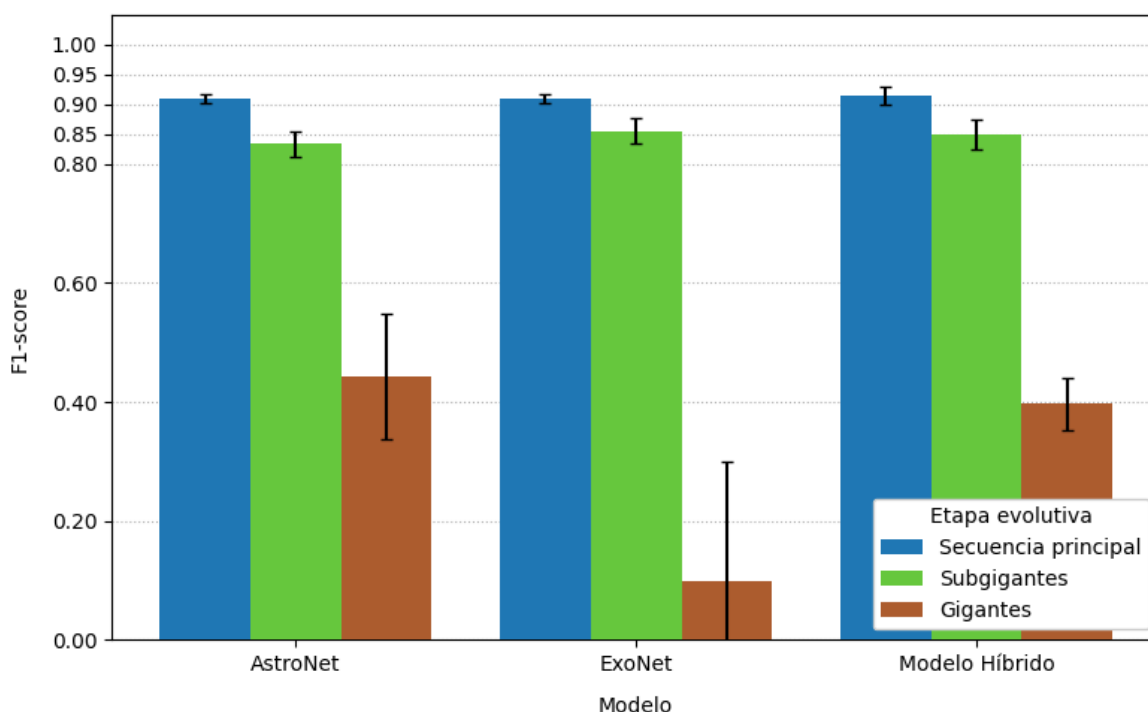


Figura 22: F1-Score de cada modelo según la etapa evolutiva de la estrella anfitriona. Fuente: Elaboración propia.

El *recall* obtenido por cada modelo en este eje de evaluación se puede observar en la [Figura 23](#). En la secuencia principal, AstroNet consiguió una excelente evaluación de 0.915 ± 0.016 , siendo superada por las otras arquitecturas. ExoNet obtuvo el mejor rendimiento en este segmento con un valor de 0.927 ± 0.014 . En subgigantes todos los modelos obtuvieron un *recall* considerablemente inferior y con mayor variabilidad entre instancias. El modelo híbrido se destaca en este segmento, manteniendo un *recall* medio de 0.881, superando a AstroNet (0.852) y ExoNet (0.855). Este margen sugiere que la integración de mecanismos de atención y unidades LSTM permite a la red extraer características geométricas en series con menor relación señal-ruido.

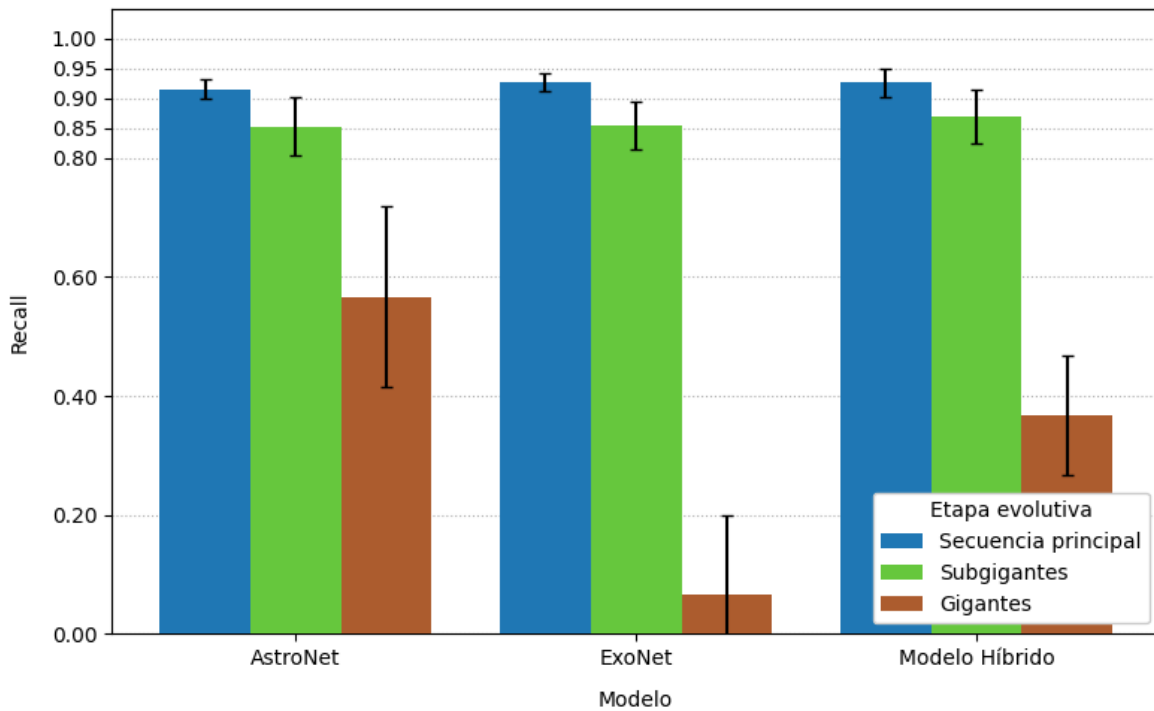


Figura 23: *Recall* de cada modelo según la etapa evolutiva de la estrella anfitriona. (*) El segmento de gigantes contiene sólo 3 TCEs positivos. Fuente: Elaboración propia.

La tasa de falsos positivos de cada modelo en este eje de evaluación se muestra en la [Figura 24](#). Ambos modelos que sólo utilizan las curvas de luz como entrada obtuvieron resultados similares en esta métrica, con una tasa de falsos positivos de poco más de 0.0250 en estrellas de la secuencia principal, y cerca de 0.050 en subgigantes. ExoNet obtiene resultados más uniformes, con una TFP de aproximadamente 0.034 en ambas categorías.

En el segmento de estrellas gigantes, con 57 TCEs en el conjunto de prueba, AstroNet obtuvo la peor evaluación, con una TFP de 0.054 ± 0.006 . El modelo híbrido consigue una

ligera mejora, con un resultado de 0.026 ± 0.012 en esta métrica. En comparación, ExoNet no produjo falsos positivos, lo que sugiere que en señales con una relación señal-ruido muy baja el modelo predice que la señal corresponde a la clase mayoritaria en estrellas de propiedades físicas similares (con una razón de 14.5 entre las clases negativa y positiva).

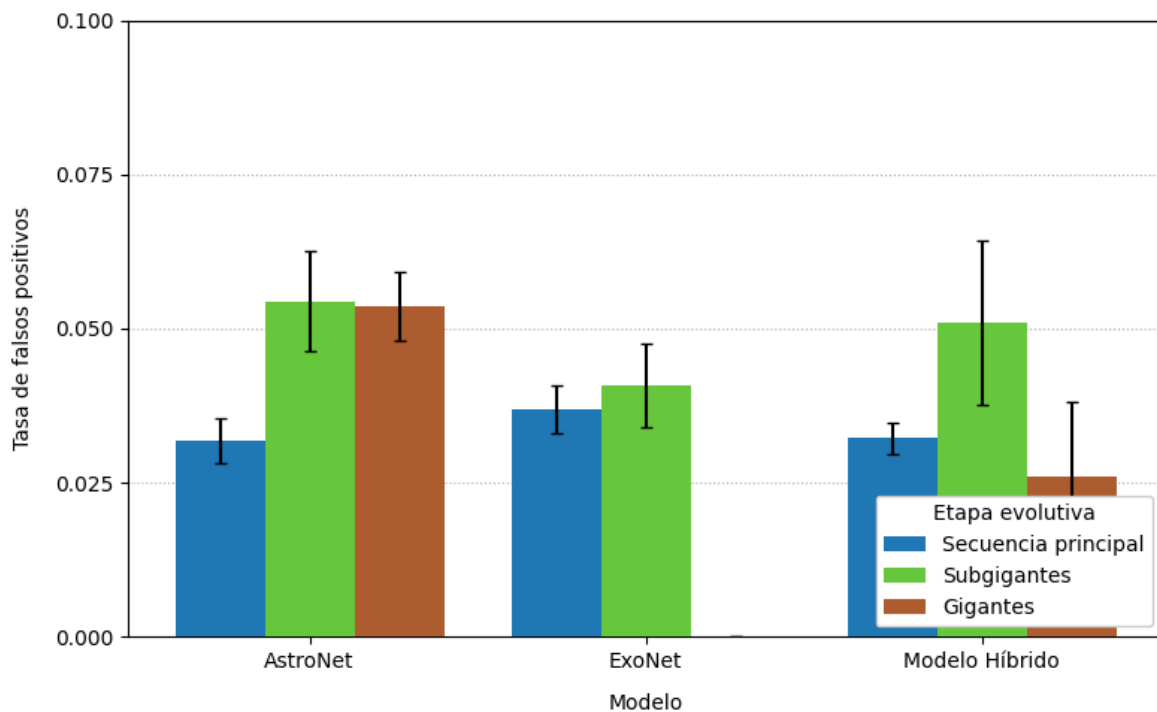


Figura 24: Tasa de falsos positivos de cada modelo según la etapa evolutiva de la estrella anfitriona. Fuente: Elaboración propia.

CAPÍTULO 5

CONCLUSIONES

5.1 Conclusiones generales

En este trabajo de memoria se evaluaron y compararon tres arquitecturas de aprendizaje profundo (AstroNet, ExoNet, y un modelo híbrido basado en LSTM con un mecanismo de atención) aplicadas a la tarea de validación (*vetting*) de exoplanetas. Utilizando un conjunto de 14,573 *Threshold Crossing Events* provenientes de 9,153 estrellas observadas por la misión espacial *Kepler*, se analizó cómo la inyección de conocimiento de dominio y la complejidad arquitectónica impactan la detección de tránsitos. Los experimentos se llevaron a cabo con una tarjeta gráfica RTX 3070 en un entorno Linux. Se analizaron los resultados de 1467 TCEs de prueba segmentados según la según clase espectral y etapa evolutiva de sus estrellas anfitrionas.

En términos comparativos, se concluye que no existe una arquitectura universalmente superior, sino que cada una presenta ventajas según el contexto observacional y computacional. AstroNet destaca como una línea base altamente robusta, estable y rápida de entrenar. ExoNet obtiene los resultados más uniformes a lo largo de la clasificación espectral. Por su parte, el modelo híbrido alcanza los límites más altos de sensibilidad (*recall*) y permite exprimir detecciones en regímenes de menor relación señal-ruido observados en estrellas subgigantes, a costa de una optimización más compleja y mayor tiempo de entrenamiento.

Por otro lado, al analizar los resultados de manera segmentada, se encontró que el rendimiento de las redes neuronales está ligado a la naturaleza física del sistema observado. Tal como dicta la astrofísica, la clasificación en estrellas subgigantes resultó ser significativamente más desafiante que en estrellas en la secuencia principal, con caídas en de *recall* de 5%-10% para todos los modelos debido a la pérdida en la relación señal-ruido por el aumento del radio estelar. De igual manera, al segmentar por clase espectral, las estrellas análogas al Sol (clase G) ofrecieron el mejor rendimiento de predicción, mientras que las estrellas más frías (clase K) presentaron un aumento abrupto en la tasa de falsos positivos, atribuible a su mayor variabilidad fotométrica.

Finalmente, se identificó un efecto variable al incorporar información de los centroides al modelo. En principio, estos datos permiten identificar la presencia de estrellas binarias eclipsantes y posiblemente otros objetos capturados por el CCD, sin embargo, a pesar de que los centroides de las estrellas de clase G presentan una mayor dispersión, ExoNet consiguió mantener una alta capacidad predictiva. En contraste, los centroides estables de las estrellas de clase F penalizaron la sensibilidad del modelo.

5.2 Cumplimiento de objetivos

La completitud de esta memoria se puede evaluar en base a los objetivos específicos establecidos, los cuales se alinean con el objetivo general descrito en la sección 1.3.1. A continuación se presentan los resultados obtenidos para cada objetivo específico:

- **Investigar e implementar modelos de aprendizaje profundo aplicados a la tarea de *vetting* de exoplanetas con el método del tránsito.**
Este objetivo fue trabajado en la sección 3.4, donde se presentan tres modelos de aprendizaje profundo presentes en la literatura diseñados para la tarea de *vetting*. El modelo base consiste en una arquitectura CNN, y los dos siguientes proponen mejoras basadas en añadir información de la estrella y los centroides (ExoNet) y en la inclusión de arquitecturas de aprendizaje profundo novedosas (Modelo híbrido con LSTM y atención).
- **Evaluar cada modelo implementado utilizando métricas clave para tareas de clasificación binaria.**
En la sección 3.2 se detalla el preprocesamiento aplicado a los datos, la sección 3.4.4 detalla el proceso de entrenamiento para cada modelo, incluyendo los tiempos de entrenamiento e hiperparámetros. Por último, en la sección 4 se presentan los resultados generales, con métricas como *accuracy*, *recall* y tasa de falsos positivos.
- **Comparar los resultados obtenidos utilizando características de las estrellas anfitrionas como eje de análisis.**
Este objetivo fue abordado en el capítulo 4, donde se compararon los resultados de todos los modelos de manera segmentada por clase espectral y etapa evolutiva de las estrellas, considerando los efectos de las modificaciones al modelo base en las predicciones.

5.3 Trabajo futuro

Los resultados obtenidos en esta memoria abren algunas líneas de investigación futuras para perfeccionar las herramientas de *vetting* de exoplanetas.

1. **Aumento de datos mediante simulaciones sintéticas:** Dado el desbalance de datos entre categorías de estrellas en el catálogo de Kepler, consistente con los objetivos de la misión, se podrían utilizar herramientas computacionales para generar curvas de luz artificiales con tránsitos de diversos parámetros, ayudando a entrenar las redes neuronales para identificar planetas en estrellas de clase espectral K o gigantes.

2. **Módulos de atención cruzada para centroides:** Ante el hallazgo sobre el efecto de la varianza en los centroides, se propone incorporar un mecanismo de atención que permita a la red neuronal evaluar dinámicamente la relación señal-ruido en esta serie de tiempo, decidiendo cuándo darle peso a esta información para descartar binarias, y cuándo priorizar las características del canal de flujo.
3. **Interpretabilidad astrofísica (XAI):** Para mitigar el efecto de “caja negra” del aprendizaje profundo, estudios futuros deben implementar técnicas de Inteligencia Artificial Explicable (como mapas de activación), que permitan comprobar visualmente si las redes neuronales están identificando la morfología del tránsito o si están aprendiendo sesgos derivados de la variabilidad de la estrella o la distribución de estrellas en el conjunto de datos (en el caso de utilizar parámetros estelares como entrada).

REFERENCIAS BIBLIOGRÁFICAS

- [Ansdell et al., 2018] Ansdell, M., Ioannou, Y., Osborn, H. P., & Sasdelli, M. (2018). Scientific Domain Knowledge Improves Exoplanet Transit Classification with Deep Learning. *The Astrophysical Journal Letters*.
- [Bahdanau et al., 2014] Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate.
- [Basri et al., 2010] Basri, G., Walkowicz, L. M., Batalha, N., Gilliland, R. L., Jenkins, J., Borucki, W. J., . . . Brown, T. (2010). Photometric Variability in Kepler Target Stars: The Sun Among Stars -- A First Look. *The Astrophysical Journal*.
- [Bastien et al., 2013] Bastien, F. A., Stassun, K. G., Basri, G., & Pepper, J. (2013). An observational correlation between stellar brightness variations and surface gravity. *Nature*.
- [Berger et al., 2015] Berger, T., Huber, D., Sadlers, J., Gaidos, E., Tayar, J., & Kraus, A. (2015). The Gaia-Kepler Stellar Properties Catalog. I. Homogeneous Fundamental Properties for 186,301 Kepler Stars. *The Astronomical Journal*.
- [Bryant et al., 2023] Bryant, E. M., Bayliss, D., & Van Eylen, V. (2023). The occurrence rate of giant planets orbiting low-mass stars with TESS. *Monthly Notices of the Royal Astronomical Society*.
- [Catanzarite, 2015] Catanzarite, J. H. (2015). *Autovetter Planet Candidate Catalog for Q1-Q17 Data Release 24*. NASA AmesResearch Center. Obtenido de <https://exoplanetarchive.ipac.caltech.edu/docs/KSCI-19091-001.pdf>
- [Catanzarite y Shao, 2011] Catanzarite, J., & Shao, M. (2011). The Occurrence Rate of Earth Analog Planets Orbiting Sunlike Stars. *The Astrophysical Journal*.
- [Christiansen et al., 2012] Christiansen, J. L., Jenkins, J. M., Caldwell, D. A., Burke, C. J., Tenenbaum, P., Seader, S., . . . Van Cleve, J. (2012). The Derivation, Properties, and Value of Kepler's Combined Differential Photometric Precision. *Publications of the Astronomical Society of the Pacific*.
- [Coughlin et al., 2016] Coughlin, J. L., Mullally, F., Thompson, S. E., Rowe, J. F., Burke, C. J., Latham, D. W., . . . Rac. (2016). Planetary Candidates Observed by Kepler. VII. The First Fully Uniform Catalog Based on The Entire 48 Month Dataset (Q1-Q17 DR24). *The Astrophysical Journal Supplement Series*.
- [Deeg y Alonso, 2018] Deeg, H. J., & Alonso, R. (2018). Transit Photometry as an Exoplanet Discovery Method. In *Handbook of Exoplanets*. Springer International Publishing.

- [Gers et al., 2000] Gers, F., Schmidhuber, J., & Cummins, F. (2000). Learning to Forget: Continual Prediction with LSTM. *Neural Computation*.
- [Glorot y Bengio, 2010] Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *International Conference on Artificial Intelligence and Statistics*.
- [Google, s.f.] Google. (s.f.). *Clasificación: ROC y AUC*. Google Developers. Recuperado de: <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc?hl=es-419>
- [Hernández-Carnerero et al., 2025] Hernández-Carnerero, À., Sánchez-Marrè, M., & Morales, J. C. (2025). Additive Attention for Vetting Transiting Exoplanet Candidates. *The Astronomical Journal*.
- [Hochreiter et al., 1997] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*.
- [Huber, 2019] Huber, D. (2019). evolstate: Assign simple evolutionary states to stars. Astrophysics Source Code Library, record ascl:1905.003.
- [Jenkins et al., 2010] Jenkins, J. M., Caldwell, D. A., Chandrasekaran, H., Twicken, J. D., Bryson, S. T., Quintana, E. V., . . . Borucki, W. J. (2010). Overview of the Kepler Science Processing Pipeline. *The Astrophysical Journal*.
- [Kaushik et al., 2025] Kaushik, M., Mattoo, A., & Rastogi, R. (2025). Exoplanet Detection: A Detailed Analysis. *International Conference on Big Data Analysis*. Springer.
- [Kingma y Ba, 2014] Kingma, D. P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization. *3rd International Conference for Learning Representations*.
- [Kipping, 2023] Kipping, D. (2023). The SNR of a transit. *Monthly Notices of the Royal Astronomical Society*.
- [Koch et al., 2010] Koch, D. G., Borucki, W. J., Basri, G., Batalha, N. M., Brown, T. M., Caldwell, D., . . . Jen. (2010). KEPLER MISSION DESIGN, REALIZED PHOTOMETRIC PERFORMANCE, AND EARLY SCIENCE. *The Astrophysical Journal Letters*.
- [Konacki et al., 2003] Konacki, M., Torres, G., Jha, S., & Sasselov, D. D. (2003). An extrasolar planet that transits the disk of its parent star. *Nature*.
- [Kovács et al., 2002] Kovács, G., Zucker, S., & Mazeh, T. (2002). A box-fitting algorithm in the search for periodic transits. *Astronomy and Astrophysics*.
- [Krizhevsky et al., 2012] Krizhevsky, A., Hinton, G., & Sutskever, I. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*.

- [Loaiza, 2020] Loaiza, K. (2020). *Deep learning for decision support in dermatology*. Kongens Lyngby: Technical University of Denmark.
- [McQuillan et al., 2014] McQuillan, A., Mazeh, T. & Aigrain, S. (2014). ROTATION PERIODS OF 34,030 KEPLER MAIN-SEQUENCE STARS: THE FULL AUTOCORRELATION SAMPLE. *The Astrophysical Journal Supplement Series*.
- [Narang et al., 2018] Narang, M., Manoj, P., Furlan, E., Mordasini, C., Henning, T., Mathew, B., . . . Sivarani, T. (2018). Properties and Occurrence Rates for Kepler Exoplanet Candidates as a Function of Host Star Metallicity from the DR25 Catalog. *The Astronomical Journal*.
- [NASA, 2013] NASA. (2013). *Light curve of binary star Kepler-16*. NASA Kepler News (Enlace original inactivo). Recuperado a través de Wikimedia Commons, disponible en: es.wikipedia.org/wiki/Archivo:Light_curve_of_binary_star_Kepler-16.jpg
- [NASA Exoplanet Archive, s.f.] NASA Exoplanet Archive. (s.f.). Exoplanet and candidate statistics. Recuperado de: https://exoplanetarchive.ipac.caltech.edu/docs/counts_detail.html
- [NASA Exoplanet Science Institute, s.f.] NASA Exoplanet Science Institute. (s.f.). *Kepler Threshold Crossing Events (TCE) Q1-Q17 DR24*. Recuperado de https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=q1_q17_dr24_tce
- [Noble, s.f.] Noble, J. (s.f.) *What is long short-term memory (LSTM)? IBM*. Recuperado de: <https://www.ibm.com/think/topics/lstm>
- [Peng et al., 2018] Peng, C., Li, Y., Yu, Y., Zhou, Y., & Du, S. (2018). Multi-step-ahead Host Load Prediction with GRU Based Encoder-Decoder in Cloud Computing. *2018 10th International Conference on Knowledge and Smart Technology*.
- [Qamar y Zardari, 2023] Qamar, R., & Zardari, B. A. (2023). Artificial Neural Networks: An Overview. *Mesopotamian journal of Computer Science*.
- [Rainio et al., 2024] Rainio, O., Teuho, J., & Klén, R. (2024). Evaluation metrics and statistical tests for machine learning. *Scientific Reports*.
- [Sagear, 2020] Sagear, S. A. (2020). Upper Limits on Planet Occurrence around Ultracool Dwarfs with K2. *The Astronomical Journal*.
- [Salehinejad et al., 2017] Salehinejad, H., Sankar, S., Barfett, J., Colak, E., & Valaee, S. (2017). Recent Advances in Recurrent Neural Networks.
- [Seager & Mallén-Ornelas, 2003] Seager, S., & Mallén-Ornelas, G. (2003). A Unique Solution of Planet and Star Parameters from an Extrasolar Planet Transit Light Curve. *The Astrophysical Journal*.

- [Shallue & Vanderburg, 2018] Shallue, C. J., & Vanderburg, A. (2018). Identifying Exoplanets with Deep Learning: A Five Planet Resonant Chain around Kepler-80 and an Eighth Planet around Kepler-90. *The Astronomical Journal*.
- [Sharma et al., 2020] Sharma, S., Sharma, S., & Athaiya, A. (2020). ACTIVATION FUNCTIONS IN NEURAL NETWORKS. *International Journal of Engineering Applied Sciences and Technology*.
- [Slawson et al., 2011] Slawson, R. W., Prša, A., Welsh, W. F., Orosz, J. A., Rucker, M., Batalha, N., . . . Koch, D. (2011). Kepler Eclipsing Binary Stars. II. 2165 Eclipsing Binaries in the Second Data Release. *The Astronomical Journal*.
- [Southworth, 2012] Southworth, J. (2012). Eclipsing Binary Stars: the Royal Road to Stellar Astrophysics. *Orbital couples: 'Pas de deux' in the Solar System and the Milky Way*.
- [Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., . . . Polosukhin, I. (2017). Attention Is All You Need. *31st International Conference on Neural Information Processing Systems*.
- [Vujovic, 2021] Vujovic, Z. (2021). Classification Model Evaluation Metrics. *International Journal of Advanced Computer Science and Applications*.

ANEXOS

Descripción de variables del conjunto de datos

A continuación, se detallan las columnas conservadas en el conjunto de datos consolidado para este trabajo, divididas entre los parámetros de entrada del evento de cruce de umbral (TCE) y las propiedades físicas de la estrella anfitriona extraídas de Gaia-Kepler.

Columnas conservadas del catálogo *TCE Q1-Q17 Data Release 24*:

- **Kepid**: Número de identificación de la estrella objetivo. Este ID es único por estrella.
- **tce_pln_num**: Identificador numérico secuencial asignado a cada TCE detectado en una misma estrella objetivo.
- **tce_period**: Período orbital estimado del tránsito, medido en días.
- **tce_time0bk**: Época o tiempo medio del primer tránsito detectado en la serie temporal, medido en Barycentric Julian Day (BJD), con un desfase que sitúa el inicio en el 1 de enero de 2009, 12:00 UTC.
- **tce_duration**: Duración del tránsito observado, medida desde el ingreso hasta el egreso, expresada en horas.
- **tce_depth**: Profundidad del tránsito o fracción del flujo lumínico bloqueado, medida en partes por millón.
- **av_training_label**: Etiqueta de clasificación de referencia del catálogo, reducida a un esquema binario de clasificación: candidato a planeta (CP, clase positiva) o no candidato (NCP, clase negativa).

Parámetros físicos de la estrella anfitriona incorporadas desde el catálogo *Gaia-Kepler Stellar Properties Catalog*:

- **Mass**: Masa estimada de la estrella, expresada en masas solares ($M_{\odot}$).
- **Teff**: Temperatura efectiva de la atmósfera estelar, medida en Kelvin (K).
- **Logg**: Logaritmo en base 10 de la gravedad superficial de la estrella, expresado en unidades cgs ($\log_{10}(g)$ con g en cm/s^2).
- **Fe/H**: Metalicidad estelar, medida como el logaritmo de la abundancia de hierro con respecto al hidrógeno, normalizada en relación con la abundancia solar (dex).
- **Rad**: Radio estimado de la estrella, expresado en radios solares ($R_{\odot}$).
- **Rho**: Densidad estelar media, expresada como el logaritmo en base 10 de la densidad, normalizada en relación con la densidad solar ($\log_{10} \frac{\rho_{\star}}{\rho_{\odot}}$).

Métricas de evaluación de cada modelo según la clase espectral de la estrella

Se han marcado (*) las clases con baja significancia estadística. Las clases B y M contienen un bajo número de datos en el conjunto de prueba (9 y 11 respectivamente). Además, la clase B no posee ejemplos de la clase positiva, lo que invalida las métricas F1-Score, sensibilidad (*recall*) y precisión (*precision*). La clase A contiene solo 3 ejemplos de la clase positiva, versus 200 ejemplos de la clase negativa.

Tabla 10: Métricas de evaluación de AstroNet segmentadas por la clase espectral de la estrella anfitriona

Clase	F1-Score	Accuracy	Recall	TFP	Precision
B*	-	1.000 ± 0.000	-	0.000 ± 0.000	-
A*	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000	0.000 ± 0.000	1.000 ± 0.000
F	0.876 ± 0.013	0.966 ± 0.003	0.912 ± 0.021	0.025 ± 0.003	0.843 ± 0.015
G	0.910 ± 0.009	0.928 ± 0.006	0.905 ± 0.021	0.056 ± 0.010	0.916 ± 0.012
K	0.872 ± 0.011	0.882 ± 0.012	0.858 ± 0.018	0.123 ± 0.019	0.858 ± 0.018
M*	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000	0.000 ± 0.000	1.000 ± 0.000

Tabla 11: Métricas de evaluación de ExoNet segmentadas por la clase espectral de la estrella anfitriona.

Clase	F1-Score	Accuracy	Recall	TFP	Precision
B*	-	1.000 ± 0.000	-	0.000 ± 0.000	-
A*	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000	0.000 ± 0.000	1.000 ± 0.000
F	0.868 ± 0.016	0.964 ± 0.004	0.895 ± 0.026	0.025 ± 0.004	0.843 ± 0.023
G	0.915 ± 0.010	0.932 ± 0.007	0.917 ± 0.021	0.058 ± 0.006	0.914 ± 0.007
K	0.885 ± 0.012	0.894 ± 0.011	0.866 ± 0.015	0.116 ± 0.014	0.866 ± 0.015
M*	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000	0.000 ± 0.000	1.000 ± 0.000

Tabla 12: Métricas de evaluación del modelo híbrido segmentadas por la clase espectral de la estrella anfitriona.

Clase	F1-Score	Accuracy	Recall	TFP	Precision
B *	0.000 ± 0.000	0.997 ± 0.006	-	0.000 ± 0.000	-
A*	0.931 ± 0.124	1.000 ± 0.000	1.000 ± 0.000	0.003 ± 0.000	0.893 ± 0.184
F	0.881 ± 0.012	0.967 ± 0.003	0.930 ± 0.029	0.027 ± 0.003	0.837 ± 0.012
G	0.923 ± 0.017	0.939 ± 0.012	0.923 ± 0.026	0.051 ± 0.008	0.924 ± 0.011
K	0.873 ± 0.017	0.885 ± 0.015	0.871 ± 0.021	0.103 ± 0.015	0.875 ± 0.017
M*	0.967 ± 0.100	0.991 ± 0.027	0.950 ± 0.150	0.000 ± 0.000	1.000 ± 0.000

Métricas de evaluación de cada modelo según la clase espectral de la estrella

Se ha marcado la categoría de gigantes (*) debido a su baja significancia estadística. El conjunto de pruebas contiene sólo 57 ejemplos de este segmento, de los cuales 54 pertenecen a la clase negativa y 3 a la clase positiva.

Tabla 13: Métricas de evaluación de AstroNet segmentadas por la etapa evolutiva de la estrella anfitriona.

Etapas evolutiva	F1-Score	Accuracy	Recall	TFP	Precision
Secuencia principal	0.909 ± 0.008	0.955 ± 0.004	0.915 ± 0.016	0.032 ± 0.004	0.903 ± 0.009
Subgigantes	0.834 ± 0.021	0.925 ± 0.008	0.852 ± 0.049	0.054 ± 0.008	0.818 ± 0.018
Gigantes*	0.443 ± 0.105	0.926 ± 0.011	0.567 ± 0.153	0.054 ± 0.006	0.365 ± 0.081

Tabla 14: Métricas de evaluación de ExoNet segmentadas por la etapa evolutiva de la estrella anfitriona.

Etapa evolutiva	F1-Score	Accuracy	Recall	TFP	Precision
Secuencia principal	0.908 ± 0.08	0.954 ± 0.004	0.927 ± 0.014	0.037 ± 0.004	0.891 ± 0.010
Subgigantes	0.855 ± 0.022	0.936 ± 0.009	0.855 ± 0.040	0.041 ± 0.007	0.857 ± 0.019
Gigantes*	0.100 ± 0.200	0.951 ± 0.007	0.067 ± 0.133	0.000 ± 0.000	0.200 ± 0.400

Tabla 15: Métricas de evaluación del modelo híbrido segmentadas por la etapa evolutiva de la estrella anfitriona.

Etapa evolutiva	F1-Score	Accuracy	Recall	TFP	Precision
Secuencia principal	0.914 ± 0.015	0.957 ± 0.007	0.926 ± 0.024	0.032 ± 0.003	0.903 ± 0.008
Subgigantes	0.849 ± 0.026	0.931 ± 0.011	0.831 ± 0.035	0.051 ± 0.013	0.831 ± 0.035
Gigantes*	0.397 ± 0.043	0.942 ± 0.008	0.457 ± 0.068	0.026 ± 0.012	0.457 ± 0.068