**Aplicación de Modelos de Machine Learning a Búsquedas de Materia Oscura con Datos del Satélite Fermi-LAT de la NASA**

*Machine Learning Applications to Dark Matter Searches with Data of the NASA Satellite Fermi-LAT*

Autor: Marta Canino Romero

Tutor: Viviana Gammaldi

# **Introducción**

El presente Trabajo de Fin de Grado (TFG) aborda el análisis de la materia oscura (DM, por sus siglas en inglés), una de las incógnitas más fascinantes en el ámbito de la física moderna. A pesar de no haber sido detectada directamente en experimentos de laboratorio, su influencia gravitacional es evidente a escalas astrofísicas. Dentro del catálogo 4FGL de Fermi-LAT, que documenta fuentes puntuales de rayos gamma, aproximadamente un tercio de las fuentes identificadas carecen de una asociación clara con fenómenos astrofísicos conocidos.

El proyecto parte de la hipótesis de que algunas de estas fuentes no identificadas (unIDs) podrían ser evidencia indirecta de la existencia de materia oscura, manifestada en forma de rayos gamma provenientes de la aniquilación de partículas masivas débilmente interactuantes (WIMPs).

En estudios previos, se emplearon técnicas de clasificación binaria basadas en Machine Learning (ML) con redes neuronales artificiales (ANN) para intentar distinguir entre fuentes astrofísicas y posibles fuentes de materia oscura. El presente trabajo tiene como objetivo **replicar este análisis**, evaluar el desempeño del modelo base y **explorar el uso de técnicas de ML alternativas**, con el fin de superar las limitaciones actuales y mejorar los resultados obtenidos.

El desarrollo del proyecto incluye la comprensión y adaptación del código base proporcionado, la replicación de los resultados previos y la implementación de un nuevo modelo de ML.

Enlace al repositorio de GitHub:

<https://github.com/martacanirome4/DarkMatter_ML_TFG>

Tabla de contenido

[Introducción 1](#_Toc63272188)

[Metodología y Justificación 3](#_Toc90610928)

[Aplicación de Metodologías Ágiles y Herramientas 4](#_Toc1315008444)

[Planificación Inicial del Proyecto 5](#_Toc1262216049)

[Épicas y Objetivos Generales 6](#_Toc496117946)

[Planificación en Sprints 7](#_Toc1645660979)

[Cronograma Visual 15](#_Toc1328273005)

[Desarrollo del Modelo 16](#_Toc693832024)

[1. El problema 17](#_Toc1151332067)

[2. Fuente de Datos 17](#_Toc2000310239)

[3. Preparación de los Datos 18](#_Toc881773218)

[4. Análisis Explorativo de los Datos (EDA) 18](#_Toc236665176)

[5. Ingeniería de Características 18](#_Toc1931516825)

[6. Selección del Modelo 18](#_Toc1756917778)

[7. Entrenamiento del Modelo 18](#_Toc1266440271)

[8. Evaluación del Modelo 19](#_Toc757979517)

[9. Optimización del Modelo 19](#_Toc568944651)

[10. Despliegue del Modelo 19](#_Toc279632911)

[Anexos 20](#_Toc1801951643)

[Tablero Kanban y Burndown Chart 21](#_Toc874689910)

[Código fuente y dependencias 21](#_Toc1214695057)

[Bibliografía 21](#_Toc1249041748)

# Metodología y Justificación

A la hora de llevar a cabo la planificación para el desarrollo del presente Trabajo de Fin de Grado, se decidió optar por basar está en las metodologías denominadas Agile, típicas de los proyectos de desarrollo software y del sector tecnológico.

En Machine Learning, el proceso iterativo denominado **ETL** (Extract, Transform, and Load) abarca las fases de definición del Problema, extracción de los datos, preparación de los datos, desarrollo y evaluación del modelo y despliegue del modelo. El objetivo final ha sido extrapolar este proceso al definido en las fases de planificación del proyecto implementando también técnicas *agile*. Al tratarse el proceso ETL de uno iterativo, los Sprints, que también se consideran iterativos me permiten adaptarlo.

Al ser este trabajo uno que se asemeja más al de un trabajo de investigación científica e investigación con Aprendizaje Automático, y aunque inicialmente pudo parecer poco ajustable esta metodología a la situación a la que nos enfrentábamos, se encontró que el enfoque Agile podía traer muchos beneficios, siendo el principal de ellos el poder afrontar un problema que podía parecer una “caja negra”.

Al aplicar esta metodología al inicio de la planificación del proyecto, se definieron primero cuatro “épicas” o etapas que abarcaría el desarrollo de esta, para, dentro de cada una de estas épicas, establecer una serie de “Sprints”, o periodos de aproximadamente 2 ó 3 semanas en los que se detalla de manera más concreta qué tareas se pretendía llevar a cabo para esta fase, y con qué objetivo (detallado en el apartado de Épicas y Objetivos Generales).

# Aplicación de Metodologías Ágiles y Herramientas

Para garantizar una organización eficiente y un progreso constante, el desarrollo del TFG seguirá un enfoque basado en metodologías ágiles, específicamente adaptando elementos de SCRUM y eXtreme Programming (XP). Estas metodologías, ampliamente utilizadas en proyectos de desarrollo de software, son ideales para proyectos exploratorios y adaptativos como este.

SCRUM será el marco principal para la organización y planificación del proyecto. Las actividades del TFG se estructurarán en ciclos iterativos denominados sprints, que tendrán una duración aproximada de dos semanas. Cada sprint incluirá fases clave como:

* **Definición de User Stories**: Se generarán descripciones claras y concisas de las tareas basadas en los objetivos específicos del proyecto, como comprender el código base, replicar resultados y probar nuevos modelos. Cada user story incluirá criterios de aceptación para evaluar si la tarea se completó satisfactoriamente.
* **Planificación**: Antes de iniciar cada sprint, se seleccionarán y priorizarán las user stories que se completarán durante el período, en función de su impacto en el proyecto y la disponibilidad de tiempo.
* **Seguimiento**: Se revisará semanalmente el estado del tablero Kanban (gestionado con Excel) para ajustar el progreso, identificar bloqueos y replanificar si es necesario.
* **Revisión y Retrospectiv**a: Al final de cada sprint, se evaluará el cumplimiento de las tareas planificadas y se reflexionará sobre los aspectos positivos y las áreas de mejora, documentando aprendizajes para aplicar en futuros ciclos.

Además, se realizarán reuniones semanales con el Product Owner. En estas reuniones se revisará el cumplimiento de las tareas, se priorizarán nuevas actividades y se analizarán posibles mejoras o ajustes en los objetivos.

Con este enfoque metodológico, se espera optimizar el desarrollo del proyecto y alcanzar resultados significativos que contribuyan al estudio de la materia oscura.

# Planificación Inicial del Proyecto

## Épicas y Objetivos Generales

**Épica 1 (EPIC-01): Definición del Alcance y Organización del Proyecto**

* **Objetivo:** Definir el marco teórico, establecer el marco de trabajo, definir los objetivos y organizar el backlog.
* **Historias de usuario:**
  + *Como desarrollador, quiero definir el alcance y objetivos del proyecto para tener una dirección clara.*
  + *Como investigador, quiero organizar y priorizar las tareas en un backlog para gestionar el proyecto de forma eficiente.*
  + *Como analista, quiero definir indicadores de éxito para evaluar la solución final.*

**Épica 2 (EPIC-02) Validación y Comprensión del Modelo Existente (UNN)**

* **Objetivo:** Ejecutar, comprender y documentar el código fuente del modelo UNN, que sirve de base de referencia.
* **Historias de usuario:**
  + *Como desarrollador, quiero ejecutar el código del modelo UNN para confirmar su funcionamiento y obtener resultados de referencia.*
  + *Como investigador, quiero revisar y comprender la lógica, variables y parámetros del código para facilitar futuras modificaciones.*
  + *Como analista, quiero documentar y comentar las funciones y resultados obtenidos para tener una base comparativa.*

**Épica 3 (EPIC-03): Desarrollo del Nuevo Modelo de Detección de Anomalías**

* **Objetivo:** Diseñar, implementar, entrenar y optimizar una solución de detección de anomalías.
* **Historias de usuario:**
  + *Como desarrollador, quiero diseñar la arquitectura del nuevo modelo para asegurar que cumple con los requerimientos técnicos.*
  + *Como investigador, quiero realizar spikes experimentales y evaluar diferentes técnicas para elegir la más adecuada.*
  + *Como analista, quiero entrenar y validar el modelo con métodos de validación cruzada para garantizar resultados fiables.*

**Épica 4 (EPIC-04): Integración, Análisis Comparativo y Documentación Final**

* **Objetivo:** Integrar los resultados de ambos modelos (UNN y detección de anomalías), realizar un análisis comparativo y redactar el informe final del TFG.
* **Historias de usuario:**
  + *Como autor, quiero integrar y comparar los resultados de ambos enfoques para extraer conclusiones sólidas y elaborar un informe final claro y coherente.*

## Planificación en Sprints

Cada sprint se identifica con un ID y las tareas tienen su propio identificador siguiendo el formato:

**TSK-[EPIC ID]-[SPRINT ID]-[Número de tarea]**

Sprint 1 (Semanas 1-2) – EPIC-01: Definición del Alcance y Organización del Proyecto

* **Tareas:**
  + **TSK-EP1-S1-01**: "Definir el alcance y objetivos del proyecto"

*Como desarrollador, quiero definir claramente el alcance y objetivos para tener una dirección precisa.*

* + **TSK-EP1-S1-02**: "Crear el backlog y priorizar tareas”

*Como investigador, quiero organizar todas las tareas en un backlog priorizado para gestionar el proyecto de forma eficiente.*

* + **TSK-EP1-S1-03**: "Revisión de literatura sobre el modelo UNN"

*Como investigador, quiero recopilar y sintetizar información sobre UNN para fundamentar mi proyecto.*

* + **TSK-EP1-S1-04**: "Revisión de literatura sobre técnicas de detección de anomalías”

*Como investigador, quiero conocer diferentes enfoques para la detección de anomalías para seleccionar el más adecuado.*

Sprint 2 (Semanas 3-4) – EPIC-02: Validación y Comprensión del Modelo Existente (UNN)

* **Tareas:**
  + **TSK-EP2-S2-01**: “Ejecución del código fuente del modelo UNN"

*Como desarrollador, quiero ejecutar el código del modelo UNN para confirmar su funcionamiento y obtener resultados de referencia.*

* + **TSK-EP2-S2-02:** "Revisión del código fuente para entender la lógica, variables y parámetros"

Como investigador, quiero analizar el código para comprender su funcionamiento interno y arquitectura.

* + **TSK-EP2-S2-03:** "Añadir comentarios y documentar funciones, variables y parámetros"

Como desarrollador, quiero agregar comentarios claros en el código para facilitar su mantenimiento.

* + **TSK-EP2-S2-04:** "Documentar los resultados obtenidos del modelo UNN"

Como analista, quiero registrar el comportamiento y las métricas del modelo como base de comparación.

Sprint 3 (Semanas 5-6) - EPIC 03 (Parte I): Diseño y Prototipado del Nuevo Modelo

* **Tareas:**
  + **TSK-EP3-S3-01: "Revisar y definir requisitos técnicos para el nuevo modelo"**

*Como investigador, quiero identificar los requisitos funcionales y técnicos (por ejemplo, rendimiento, escalabilidad) que debe cumplir la solución, para orientar el diseño.*

**Subtareas:**

* + - Investigar técnicas actuales de detección de anomalías (autoencoders, Isolation Forest, clustering, etc.).
    - Listar ventajas y desventajas de cada técnica en relación a los datos disponibles.

* + **TSK-EP3-S3-02: "Diseñar la arquitectura preliminar del modelo"**

*Como desarrollador, quiero crear un diagrama de flujo y definir módulos (preprocesamiento, núcleo de detección, salida), para tener una base estructurada de la solución.*

**Subtareas:**

* + - Elaborar un diagrama de flujo de la solución.
    - Especificar las funciones principales y la interacción entre módulos.
  + **TSK-EP3-S3-03: "Implementar la estructura base del prototipo"**

*Como desarrollador, quiero crear el esqueleto del código del modelo que integre la arquitectura definida, para poder realizar pruebas preliminares.*

**Subtareas:**

* + - Configurar el entorno de desarrollo y crear repositorio.
    - Codificar módulos básicos (preprocesamiento, modelo de detección, post-procesamiento).
  + **TSK-EP3-S3-04: “Realizar pruebas rápidas (spike) sobre la técnica seleccionada"**

Como investigador, quiero ejecutar pruebas de concepto con fragmentos de código para validar la técnica elegida, para confirmar su viabilidad en nuestro contexto.

Sprint 4 (S4, Semanas 7-8) - EPIC 03 (Parte II): Entrenamiento y Pruebas Iniciales del Nuevo Modelo

* **Tareas:**
  + **TSK-EP3-S4-01*: "Preparar y dividir el conjunto de datos"***

*Como analista, quiero segmentar los datos en conjuntos de entrenamiento y validación, para disponer de datos estructurados para el proceso de entrenamiento.*

**Subtareas:**

* + - Realizar limpieza y normalización de datos.
    - Establecer criterios de división (porcentaje para entrenamiento vs. Validación).
  + **TSK-EP2-S4-02: "Configurar el entorno de experimentación para el entrenamiento"**

*Como desarrollador, quiero definir la configuración (librerías, parámetros iniciales, etc.) para entrenar el prototipo, para asegurar consistencia en las pruebas.*

**Subtareas:**

* + - Ajustar configuración de software.
    - Establecer parámetros iniciales y sedes para experimentos reproducibles.
  + **TSK-EP2-S4-03: “Entrenar el prototipo del modelo de detección de anomalías”**

*Como desarrollador, quiero ejecutar el entrenamiento del modelo con el conjunto de datos preparado, para obtener resultados preliminares.*

**Subtareas:**

* + - Ejecutar el entrenamiento y monitorizar el proceso (pérdida, métricas, etc.).
    - Guardar *checkpoints* del modelo.
  + **TSK-EP2-S4-04: “Evaluar resultados preliminares y ajustar parámetros básicos"**

*Como analista, quiero analizar las métricas de rendimiento obtenidas y ajustara parámetros clave, para mejorar la capacidad del modelo en detectar anomalías.*

**Subtareas:**

* + - Comparar resultados con los objetivos definidos.
    - Probar ajustes de parámetros (tasa de aprendizaje, número de capas, etc.) en ciclos cortos.

Sprint 5 (S5, Semanas 9-10) - EPIC 03 (Parte III): Optimización y Validación Intensiva del Modelo de Detección de Anomalías

* **Tareas:**
  + **TSK-EP3-S5-01: "Optimizar y ajustar los hiperparámetros del modelo"**

*Como desarrollador, quiero afinar el modelo ajustando hiperparámetros para mejorar la precisión y robustez en la detección de anomalías.*

**Subtareas:**

* + - Realizar una búsqueda de hiperparámetros utilizando técnicas como grid search o random search.
    - Probar variaciones en la tasa de aprendizaje, número de capas y unidades, regularización, etc.
    - Registrar los resultados y compararlos con el desempeño actual.
  + **TSK-EP3-S5-02: "Validar el modelo mediante validación cruzada"**

*Como analista, quiero aplicar validación cruzada en el conjunto de datos reservado para asegurar la consistencia y fiabilidad de los resultados.*

**Subtareas:**

* + - Dividir el set de datos en varios folds (por ejemplo, 5 o 10) para la validación.
    - Ejecutar el proceso de validación cruzada y recopilar métricas (precisión, recall, F1-score, etc.).
    - Analizar la variabilidad entre folds y determinar la estabilidad del modelo.
  + **TSK-EP3-S5-03: "Evaluar el modelo en un set de datos reservado”**

*Como analista, quiero probar el modelo optimizado en un conjunto de datos que no se utilizó durante el entrenamiento, para medir su rendimiento en condiciones reales.*

**Subtareas:**

* + - Separar un set de prueba independiente.
    - Calcular métricas de desempeño y comparar con las obtenidas durante la validación cruzada.
  + **TSK-EP3-S5-04: "Documentar mejoras, ajustes y resultados de validación”**

*Como investigador, quiero registrar detalladamente todos los ajustes realizados y los resultados obtenidos para garantizar la trazabilidad y facilitar futuras iteraciones.*

**Subtareas:**

* + - Redactar un informe técnico con los cambios realizados y sus efectos en el desempeño.
    - Crear gráficos y tablas comparativas de las métricas antes y después de la optimización.

Sprint 6 (S6, Semanas 11-12) – EPIC-04 (Parte I): Integración y Análisis Preliminar

* **Tareas:**
  + **TSK-EP3-S6-01: "Integración de los resultados del modelo UNN y el modelo de detección de anomalías"**

***Como desarrollador, quiero consolidar los resultados de ambos modelos para facilitar un análisis comparativo integral.***

**Subtareas:**

* + - **Consolidar los outputs y métricas obtenidas de modelo UNN (documentadas en EPIC-02).**
    - **Organizar los resultados del nuevo modelo optimizado (del Sprint 5) en un formato comparable.**
  + **TSK-EP3-S6-02: "Realización de un análisis preliminar comparativo de ambas soluciones"**

***Como analista, quiero comparar las dos soluciones en términos de rendimiento, precisión y robustez para identificar mejoras y áreas de oportunidad.***

**Subtareas:**

* + - Establecer criterios comparativos (eficiencia, precisión, …)
    - Elaborar gráficos y tablas que muestren las diferencias clave entre ambos modelos.
    - Redactar un resumen con conclusiones preliminares del análisis comparativo.
  + **TSK-EP3-S6-03: "Documentar el proceso de integración y análisis"**

***Como investigador, quiero dejar constancia detallada del proceso de integración análisis para facilitar la revisión y replicación de la solución.***

**Subtareas:**

* + - Redactar un. Apartado específico en la documentación del proyecto que resuma la integración y las comparativas realizadas,
    - Incluir observaciones sobre limitaciones y posibles mejoras detectadas durante el análisis.

Sprint 7 (Semanas 13-14) – EPIC-04 (Parte II): Revisión y Finalización del del Informe

* **Tareas:**
  + **TSK-EP3-S7-01**: "Revisión y ajustes del borrador del informe"

*Como autor, quiero revisar y perfeccionar el borrador para asegurar claridad y coherencia.*

* + **TSK-EP3-S7-02**: "Incorporación de feedback y correcciones finales"

*Como desarrollador, quiero ajustar el informe con base en el feedback recibido para asegurar la calidad del documento.*

Sprint 8 (Semana 15) – EPIC-04 (Parte III): Presentación y Ajustes Finales

* **Tareas:**
  + **TSK-EP4-S8-01: "Revisión final y preparación de la presentación del proyecto"**

***Como desarrollador, quiero crear una presentación clara y concisa que resuma la solución y los hallazgos del proyecto.***

* + **TSK-EP4-S8-02: "Realizar pruebas de contingencia y ajustes de última hora"**

**Como autor, quiero asegurarme de que todos los imprevistos estén resueltos para garantizar una entrega impecable.**

## Cronograma Visual

Escala de tiempo

El contenido generado por IA puede ser incorrecto.

# Desarrollo del Modelo

Para el desarrollo del modelo se ha utilizado la librería de código abierto para Python Scikit-Learn, que permite construir modelos clásicos de ML. Se trata de una *suite* que incluye una variedad de herramientas cubriendo todas las fases del ciclo de desarrollo de un modelo de Machine Learning (preprocesamiento, entrenamiento y pruebas, ajuste de hiperparámetros, …) por lo que permite construir el pipeline completo del modelo de aprendizaje automático solo utilizando su librería.

## El problema

El Fermi Large Area Telescope (Fermi-LAT) ha estudiado el cielo de rayos gamma desde 2008, catalogando miles de fuentes de alta energía.

La gran mayoría de los objetos catalogados son clases conocidas (por ejemplo, púlsares o núcleos galácticos activos), pero un número significativo permanece sin asociar, ya que carecen de contrapartes claras en otras longitudes de onda. Estas fuentes no asociadas y otras características inesperadas en los datos de Fermi-LAT presentan oportunidades para el descubrimiento. Los métodos modernos de aprendizaje automático (ML) se utilizan cada vez más para analizar las observaciones de Fermi-LAT, con el objetivo de clasificar los tipos de fuentes conocidos y señalar fenómenos nuevos o atípicos.

Por otra parte, el estudio de la materia oscura constituye uno de los mayores misterios del universo y del mundo de la astrofísica e incluso de toda la comunidad científica. En base a observaciones del cielo y análisis en el comportamiento de los astros, se ha llegado a la conclusión de que una gran parte del universo debe estar compuesto por un tipo de materia no visible por nuestros ojos, y que no interactúa de ninguna manera con la luz, por ello denominada materia oscura. La única confirmación de la existencia de esta materia se basa en estudios de la fuerza gravitacional entre los astros.

## Fuente de Datos

La fuente de datos para el desarrollo del modelo, por tanto, ha sido directamente el catálogo público del Fermi-LAT. El Telescopio Espacial Fermi-LAT recopila varios tipos de datos de rayos gamma de ata energía. Su catálogo de fuentes enumera miles de fuentes detectadas en el cielo gamma, con información como la posición celestial, la significancia de detección (estadístico TS), el flujo observado y parámetros espectrales.

Para cada fuente del catálogo incluye su espectro de energía y datos sobre su variabilidad, así como curvas de luz (evolución temporal del brillo gama) para fuentes variables.

En resumen, las fuentes de rayos gamma detectadas por Fermi-LAT vienen acompañadas de sus eventos de detección (listas de fotones individuales con tiempo, dirección y energía), espectros y curvas de luz, además de productos derivados como mapas e índices de variabilidad.

Los datos del Fermi-LAT son de acceso abierto a través del Fermi Science Support Center (FSSC) de la NASA.

Para la **construcción del dataset** se han utilizado datos experimentales del catálogo 4FGL Fermi-LAT y se ha simulado un conjunto de datos de materia oscura.

Se han analizado variables como la energía pico (E\_peak ) y la curvatura espectral (β), además de introducir características sistemáticas como la **significancia de detección** (σd ) y la **incertidumbre en β**. Además, se ha encontrado que la **significancia de detección** (sigma\_d) mejora la separación entre clases, y que incluir la **incertidumbre en β** ayuda a mejorar la clasificación de fuentes desconocidas.

## Preparación de los Datos

Trabajar con datos de Fermi-LAT en un modelo de Machine Learning enfocado a detección de anomalías requiere convertir los productos científicos en entradas numéricas consistentes, aplicando filtros y normalizaciones adecuados.

## Análisis Explorativo de los Datos (EDA)`

[Comienzo anotaciones teóricas] (Fase **descriptiva)** Consiste en graficar los datos y hacer inferencias estadísticas. También se puede hacer mediante *clustering* (agrupar grupos de datos basado en diferentes características).

* Limpiar datos para eliminar datos irrelevantes
* Eliminar valores extremos (removing *outliers?* Es lo mismo??)
* Eliminar o sustituir valores faltantes
* Cada columna de los datos debe estar en el formato apropiado
* Crear nuevas características (features)
* Identificar cómo dividir los datos en datos de entrenamiento y datos de prueba

(Fase **predictiva**) Se emplean modelos para hacer predicciones sobre los datos, como modelos de regresión (predice valores numéricos) o de clasificación (predice valores categóricos como éxito o fracaso).

Herramientas de visualización de datos: Matplotlib, Seborn, ggplot2 [Fin anotaciones teóricas]

El análisis exploratorio de datos (EDA, Exploratory Data Analysis) es un paso fundamental en cualquier estudio basado en Machine Learning. Su objetivo es comprender la estructura de los datos, identificar patrones, detectar valores atípicos y evaluar la calidad de la información antes de aplicar modelos predictivos.

En esta fase, hemos trabajado con datos provenientes del catálogo 4FGL del telescopio de rayos gamma Fermi-LAT, en busca de características que permitan distinguir entre fuentes astrofísicas identificadas y posibles señales de materia oscura. Para ello, los datos han sido preprocesados y representados mediante distintas visualizaciones.

Objetivos del Análisis Exploratorio:

* Comprender la estructura de los datos: analizar distribuciones, patrones y correlaciones.
* Detectar valores atípicos: identificar posibles errores en la medición o datos inusuales.
* Evaluar la calidad de los datos: verificar inconsistencias en la información.
* Seleccionar y transformar características relevantes: determinar qué variables serán más útiles en el modelo de Machine Learning.

Gráficos generados en el análisis exploratorio de datos

A continuación, se describen los principales gráficos generados y sus conclusiones.

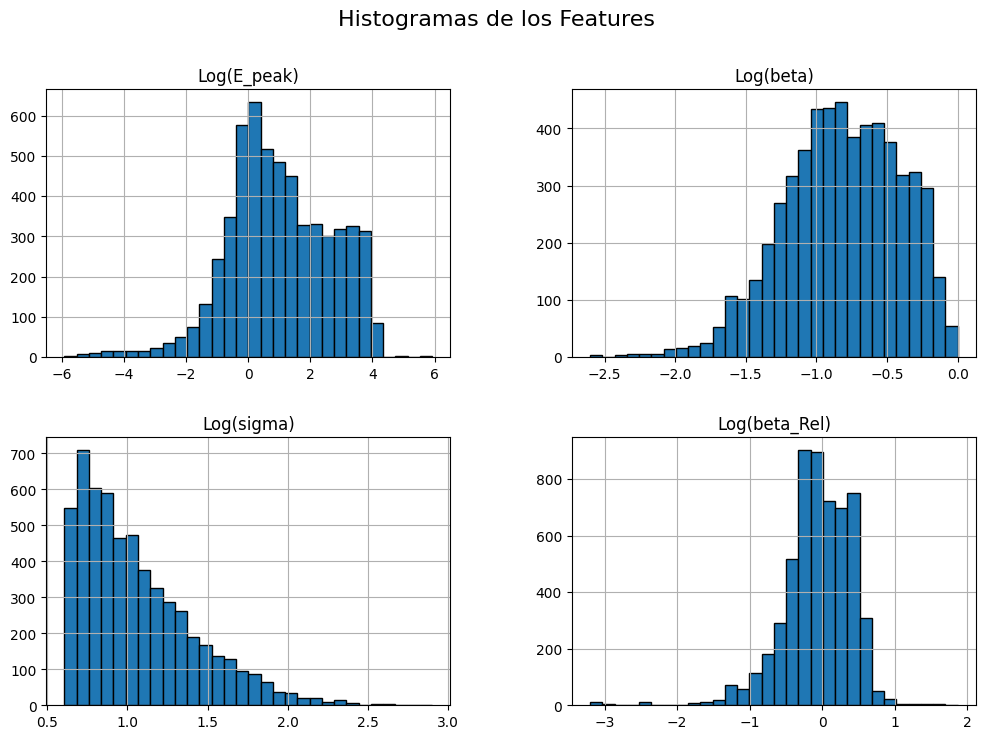
1. Histogramas individuales de cada feature

**Objetivo**: Analizar la distribución de las variables E\_peak, beta, sigma, beta\_Rel y verificar si presentan sesgos o valores extremos.

**Observaciones**:

* E\_peak y beta tienen distribuciones sesgadas hacia valores negativos y bajos, lo que confirma que los datos fueron transformados a una escala logarítmica antes de ser usados.
* sigma muestra una distribución más homogénea, lo cual indica que la significación de detección varía en un rango amplio.
* beta\_Rel presenta una distribución con valores negativos y positivos, lo que indica que la incertidumbre relativa en la curvatura espectral puede afectar la clasificación.

**Conclusión**: Los histogramas muestran que los datos han sido preprocesados en una escala logarítmica y que hay variabilidad en sigma y beta\_Rel, lo que puede afectar la clasificación de fuentes.



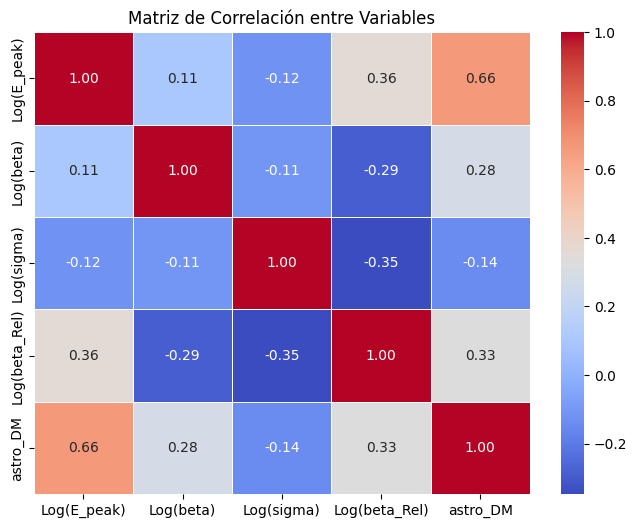
1. Matriz de correlación de las *features*

**Objetivo**: Identificar relaciones entre variables y detectar posibles redundancias.

**Observaciones**:

* E\_peak y astro\_DM tienen una correlación positiva de +0.432, lo que sugiere que la energía pico puede ser un buen predictor para clasificar fuentes.
* beta y astro\_DM tienen una correlación positiva de +0.267, indicando que la curvatura espectral también influye en la clasificación.
* sigma y beta\_Rel están moderadamente correlacionadas (-0.347), lo que indica que fuentes con mayor detección tienden a tener menor incertidumbre en su espectro.

**Conclusión**: El análisis de correlación sugiere que E\_peak y beta son características clave para diferenciar entre fuentes astrofísicas y posibles señales de materia oscura. También indica que sigma y beta\_Rel podrían ser usadas como características adicionales en el modelo.



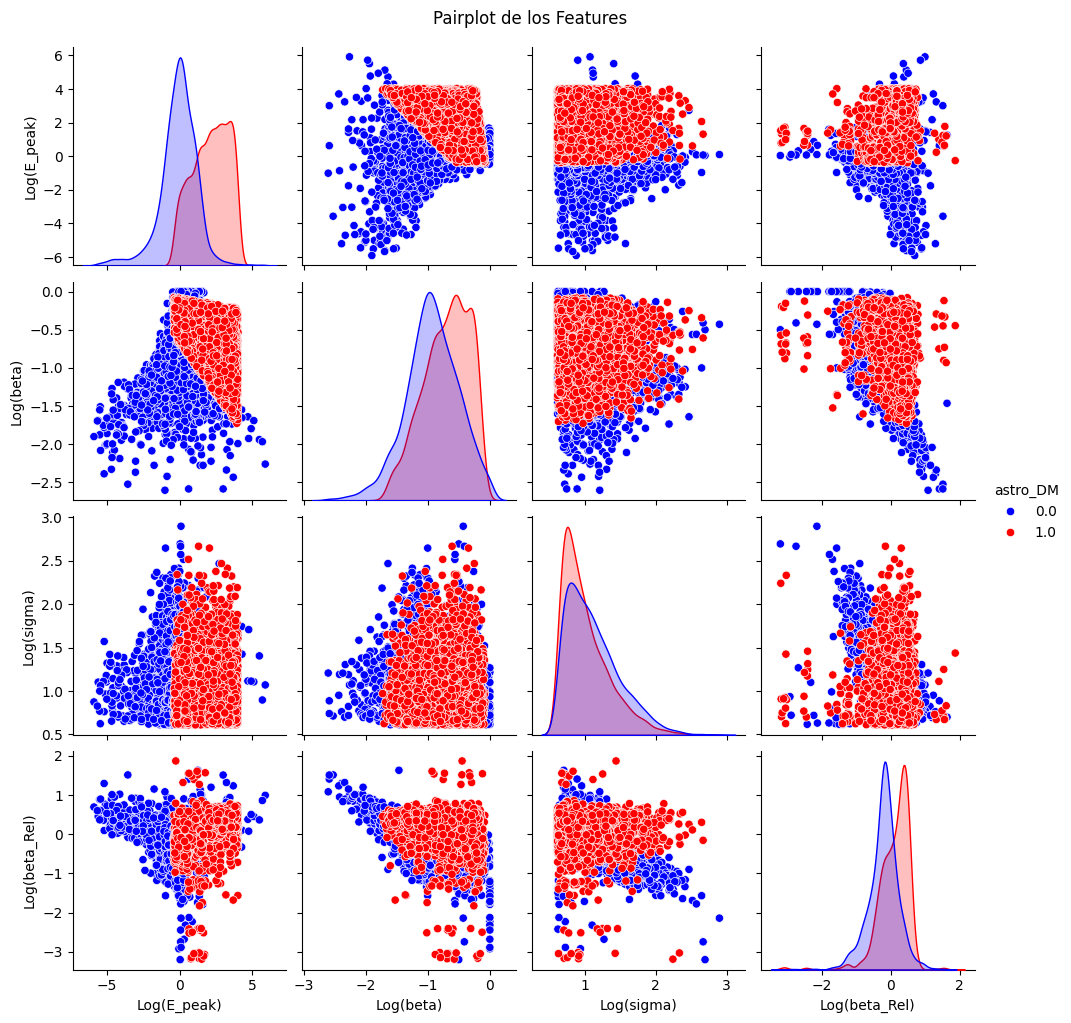
1. Pairplot (gráfica de pares)

**Objetivo**: Visualizar las relaciones entre todas las variables y observar la separación entre clases (astro\_DM = 0 y 1).

**Observaciones**:

* En la relación E\_peak vs. beta, se observan zonas de solapamiento, lo que indica que ciertas fuentes astrofísicas y posibles señales de materia oscura pueden compartir características espectrales similares.
* Algunas variables muestran patrones de agrupación, lo que sugiere que podrían ser útiles para la clasificación.

**Conclusión**: El espacio de características no es completamente separable solo con E\_peak y beta, lo que confirma la necesidad de agregar más features como sigma y beta\_Rel en los modelos de clasificación.



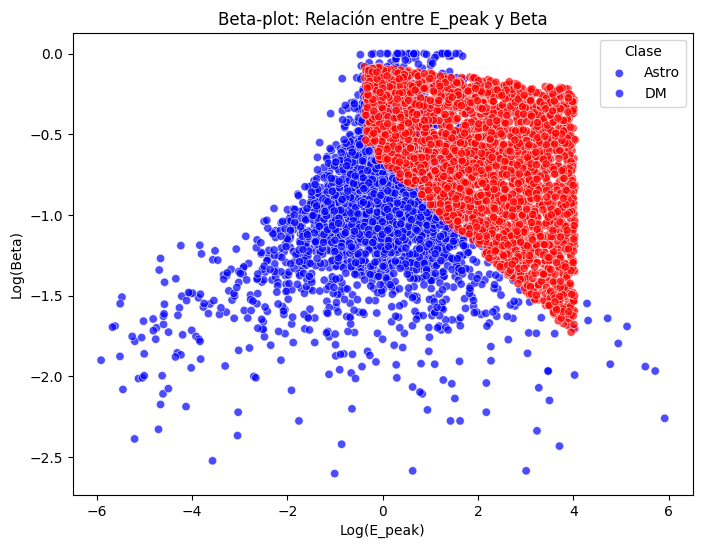
1. Scatter plot: E\_peak vs Beta (Beta-plot)

**Objetivo**: Visualizar cómo se distribuyen las clases (astro\_DM = 0 o 1) en el espacio de E\_peak y beta.

**Observaciones**:

* Se observan zonas con solapamiento entre fuentes astrofísicas y candidatas a materia oscura.
* Algunas fuentes tienen valores de E\_peak y beta bien diferenciados, lo que podría ayudar en la clasificación.

**Conclusión**: El gráfico confirma que hay una degeneración entre algunas fuentes astrofísicas y candidatas a materia oscura, por lo que se necesitan más features para mejorar la clasificación.



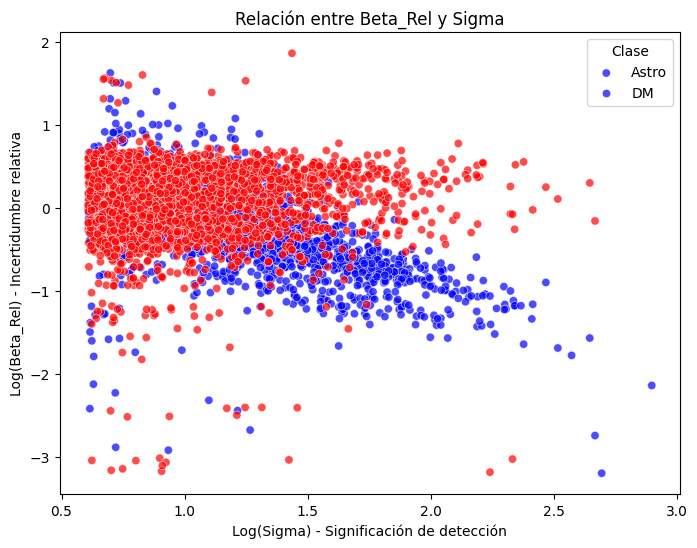
1. Scatter plot: Beta\_rel vs Sigma

**Objetivo**: Explorar la relación entre la incertidumbre relativa en beta y la significación de detección (sigma).

**Observaciones**:

* Las fuentes con mayor sigma tienden a tener menor incertidumbre (beta\_Rel).
* Las fuentes con baja significación de detección (sigma) tienen mayor incertidumbre en beta, lo que puede afectar su clasificación.

**Conclusión**: Fuentes con baja significación de detección tienen mayor incertidumbre espectral, lo que sugiere que sigma podría ser una feature clave para mejorar la clasificación.



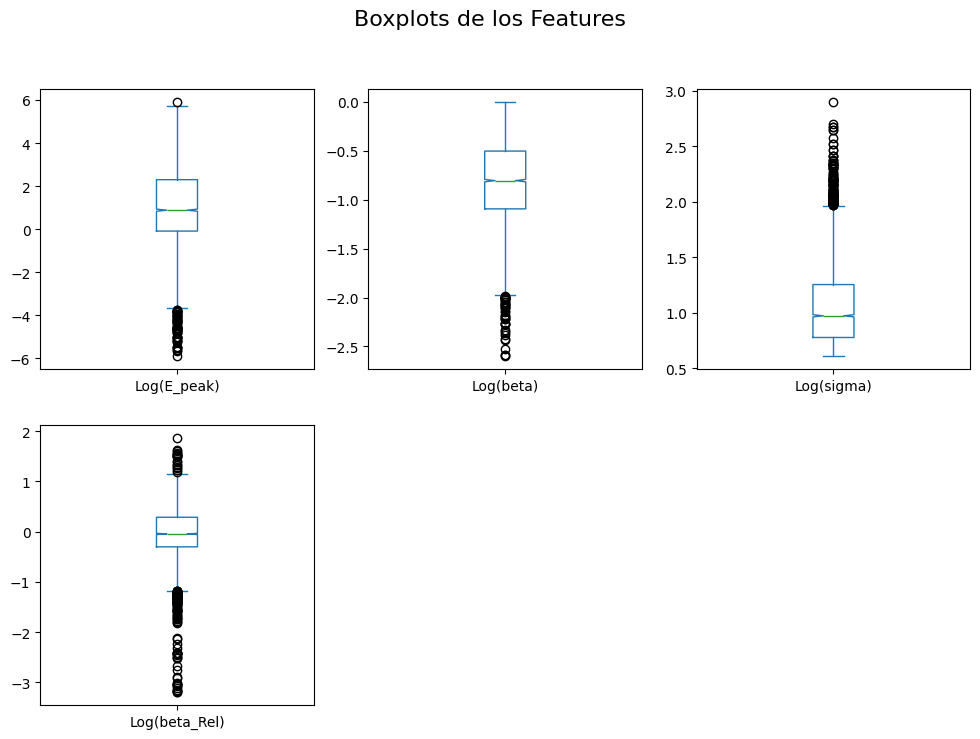
1. Boxplots para identificar valores atípicos

**Objetivo:** Detectar valores extremos que podrían sesgar el modelo.

**Observaciones**:

* Se observan **valores atípicos en E\_peak y beta**, lo que podría indicar fuentes con características inusuales.
* En sigma y beta\_Rel, hay algunos puntos fuera del rango típico, lo que sugiere que algunas fuentes tienen detecciones débiles o incertidumbre alta.

**Conclusión:** Existen valores atípicos que podrían afectar la clasificación. Se recomienda hacer una detección de anomalías (*Anomaly Detection*) para evaluar si estos datos deberían ser tratados de forma especial antes de entrenar el modelo.



## Ingeniería de Características

## Selección del Modelo

## Entrenamiento del Modelo

Feature engineering, cross validation, hyperparameter tuning, …

Herramientas para crear y afinar modelos ML: **NumPy** (computaciones numéricas sobre arrays de datos), **Pandas** (análisis de datos, visualización, limpieza y preparación), **SciPy** (cómputo para optimización, integración y regresión linear) o **Scikit-lern** (suite de clasificación, regresión, clustering y reducción de dimensionalidad, está construido sobre Numpy, SciPy y MatplotLib).

Herramientas para diseñar, entrenar y probar modelos basados en redes neuronales: **TensorFlow** (computación numérica y modelos de gran escala), **Keras** (implementación de ANNs), **Theano** (definición, optimización y evaluación matemática de expresiones relacionadas con arrays), **PyTorch** (visión computacional, NLP y experimentación).

## Evaluación del Modelo

## Optimización del Modelo

## Despliegue del Modelo

# Anexos

## Tablero Kanban y Burndown Chart

* [Excel](https://ceu365-my.sharepoint.com/:x:/g/personal/marta_caninoromero_usp_ceu_es/Ed0fLzeQ9yZHr1Hnzp4SlJwB1y4bLVMcd0qRfn579XWE0g?e=mZMkW4)

## Código fuente y dependencias

* [Repositorio de GitHub](https://github.com/martacanirome4/DarkMatter_ML_TFG)

# Bibliografía

Fermi Large Area Telescope Fourth Source Catalog Data Release 4 (4FGL-DR4)

<https://ar5iv.labs.arxiv.org/html/2307.12546#:~:text=We%20present%20an%20incremental%20version,are%20updated%20for%20all%20sources>

Fermi Gamma-ray Space Telescope Currently Available Data Products

<https://fermi.gsfc.nasa.gov/ssc/data/access/#:~:text=,51>

How and Why to Use Agile for Machine Learning

<https://medium.com/qash/how-and-why-to-use-agile-for-machine-learning-384b030e67b6>

Machine Learning Steps

<https://www.simplilearn.com/tutorials/machine-learning-tutorial/machine-learning-steps>

What is Agile Iteration

<https://www.wrike.com/agile-guide/faq/what-is-agile-iteration/>

Life Cycle of a Machine Learning Project

<https://neptune.ai/blog/life-cycle-of-a-machine-learning-project>

Navigating Through a Machine Learning Project: A Step-by-Step Guide

<https://medium.com/@careerInAI/navigating-through-a-machine-learning-project-a-step-by-step-guide-d673f72a80ff>

Classification of Fermi -LAT sources with deep learning using energy and time spectra

<https://www.researchgate.net/publication/354086181_Classification_of_Fermi_-LAT_sources_with_deep_learning_using_energy_and_time_spectra>