



UNIVERSIDAD DE LA REPÚBLICA
FACULTAD DE CIENCIAS
INSTITUTO DE FÍSICA
DEPARTAMENTO DE ASTRONOMÍA



INICIACIÓN A LA INVESTIGACIÓN

Reducción de datos de los meteoros detectados por la Red BOCOSUR (Bólidos del Cono SuR)

Nombre
Lucía Velasco

Tutor: Gonzalo Tancredi, Cotutor: Manuel Caldas
(Departamento de Astronomía, UdelAR)

11 de septiembre de 2024

Agradecimientos

Agradezco especialmente a mis tutores Gonzalo Tancredi y Manuel Caldas que me han invitado a participar de la red BOCOSUR y realizar el proyecto de Iniciación a la Investigación en el marco del mismo. Me han enseñado cómo la Astronomía se construye en grupo y la importancia de la perseverancia frente a obstáculos imprevistos en una investigación.

Este trabajo no habría sido posible sin el apoyo financiero proporcionado por CSIC y la Embajada de los Estados Unidos en Uruguay. El apoyo de estas instituciones hizo posible que realizara las pasantías dentro del proyecto BOCOSUR que enriquecieron este trabajo.

Gracias a los amigos que encontré en Facultad de Ciencias, por nunca escatimar en ánimos o risas, los celebro siempre. Gracias Matías, Lucas, Valeria, Belén, Alfredo, Mauro, Rodrigo y Nicolás.

Agradezco a Juan Pedro Ballestrino, Cecilia Deandraya y Cristian Uviedo creadores de AUTOBOL y su tutor Dr. Ing. Ignacio Ramírez por la disposición a responder preguntas y la ayuda brindada a la hora de adaptar el programa a las nuevas cámaras de la red BOCOSUR.

Gracias a los estudiantes de la Lic. en Física y Lic. en Astronomía que colaboraron con gran entusiasmo en la clasificación “manual” de videos: Paula, Sofía, Eyarúa, Inés, Sebastián e Ignacio. Su aporte fue invaluable.

Muchas gracias a mi familia, en especial a mi madre con quien supimos buscar estrellas fugaces para pedir deseos cuando era niña. Hoy le diría que son meteoros.

Gracias queridos Martin, Diana y Pepi.

Índice

Agradecimientos	1
Resumen	3
1. Introducción	4
1.1. Motivación	4
1.2. Objetivos	5
1.3. Estructura de la monografía	6
2. Terminología de la ciencia de meteoros	7
3. Redes de detección de meteoros	11
3.1. <i>Machine-learning</i> en redes de detección de meteoros	13
4. La red BOCOSUR	15
4.1. Hardware de las estaciones BOCOSUR	17
4.2. Software de las estaciones BOCOSUR	18
4.3. Operación de la red BOCOSUR	20
4.4. Mantenimiento remoto de las estaciones	20
4.5. Mantenimiento in situ de las estaciones	22
5. Base de datos de BOCOSUR	24
5.1. Clasificación “manual” de videos	25
6. Clasificación basada en aprendizaje supervisado	27
6.1. <i>Machine learning</i> en Python	28
7. Antecedentes: AUTOBOL	30
8. Clasificador automático para videos ASI	35
8.1. AUTOBOL en la etapa ASI	35
8.2. Modificaciones a AUTOBOL	37
8.3. Características o <i>features</i> del nuevo clasificador	45
8.4. Nuevo clasificador: entrenamiento y modelo final	49
9. Bólidros encontrados con el nuevo clasificador	53
9.1. Eventos multi-estación	55
9.2. Lluvias de meteoros mayores	55
10. Análisis de un caso multi-estación	58
11. Conclusiones y futuros pasos	65
Bibliografía	67

Resumen

El estudio de los meteoros, especialmente los bólidos, ha ganado relevancia dentro de la ciencia planetaria, proporcionando información sobre la composición, origen y evolución del sistema solar. Además, estas investigaciones ayudan a mejorar modelos predictivos sobre el impacto de meteoroides, esenciales para la seguridad planetaria.

En la actualidad, existen varias redes de detección óptica de meteoros. Sin embargo, la gran mayoría se localizan en el hemisferio norte. Aunque existen algunas contribuciones en el hemisferio sur, principalmente en Australia (DFN) y Brasil (BRAMON), la cobertura que proporcionan en este hemisferio es limitada. Una adición al esfuerzo de estas redes es la red uruguaya BOCOSUR, que comenzó su despliegue en 2019.

BOCOSUR cuenta con 20 estaciones de detección, distribuidas en instituciones de educación media y en Facultad de Ciencias (UdelaR), logrando una cobertura total en el país. En 2022 se realizó una actualización en todas las estaciones, pasando de sistemas de video analógicos de baja resolución a cámaras digitales de alta resolución (de WAT-902H2 ULTIMATE a ZWO ASI 178MM).

Presentamos el funcionamiento general de la red BOCOSUR y contribuciones al *pipeline* de reducción de los datos de la misma. Hemos modificado AUTOBOL, un algoritmo de clasificación basado en *machine learning* diseñado para los videos de las cámaras anteriores. El nuevo clasificador identifica objetos luminosos en movimiento y extrae características de su desplazamiento y curva de luz. El modelo final (basado en *Random Forests*) fue entrenado con una muestra compuesta de 290 ejemplos de “Bólidos” y cerca del doble de “No Bólidos”, obtenidos de experiencias de clasificación “manual” como parte de una iniciativa de ciencia ciudadana. Se hallaron 6.7k “Bólidos” en 118k videos (de las cámaras actuales). De esta forma, se redujo a menos de un 6 % la cantidad de videos a ser verificados por inspección visual.

En base a la clasificación automática, encontramos 3k eventos multi-estación (combinando el valor de la etiqueta de evento y una medida de su intensidad). Las observaciones de un mismo bólido desde distintas estaciones, permiten la determinación de su trayectoria y posible caída, computación de radiante y velocidad pre-atmosférica, que se utilizan para calcular la órbita heliocéntrica del meteoroides antes de impactar con la atmósfera terrestre. Como ejemplo, presentamos el análisis de uno de estos eventos, detectado en 4 estaciones, en mayo de 2023. Estimamos que es de origen cometario y no está asociado a ninguna de las lluvias de meteoros mayores observables desde el hemisferio sur.

1. Introducción

1.1. Motivación

El estudio de meteoros, y en particular de los meteoros brillantes conocidos como bólidos, ha ganado una relevancia creciente en la ciencia planetaria en los últimos años. Estos fenómenos, que se producen cuando objetos interplanetarios ingresan a la atmósfera terrestre a gran velocidad produciendo un destello luminoso, ofrecen una ventana única para entender la composición, origen y evolución del sistema solar.

Los meteoroides, al atravesar la atmósfera, se calientan y fragmentan debido a la fricción con los gases de la misma, liberando energía en forma de luz y calor (que llamamos meteoros). Este proceso no solo crea espectáculos luminosos impresionantes, sino que también proporciona valiosa información sobre la naturaleza de los cuerpos progenitores (principalmente cometas y asteroides). El estudio de los meteoros permite determinar la composición química y física de los meteoroides, que son representativos de los materiales primordiales del sistema solar.

Los asteroides, al igual que los cometas, son la población remanente de planetesimales, pequeños cuerpos primordiales a partir de los cuales se formaron los planetas. A lo largo del cinturón de asteroides, existen zonas donde las perturbaciones gravitacionales resonantes por parte de los planetas, principalmente Júpiter y Saturno, crean zonas dinámicamente caóticas. Cuando las colisiones entre asteroides cerca de los límites de estas zonas envían fragmentos hacia ellas, las excentricidades orbitales aumentan rápidamente y los fragmentos cruzan las órbitas de otros planetas, incluida la Tierra ([Council, 1998](#)). De esta manera, la recuperación de los fragmentos que llegan a la superficie terrestre, es decir, meteoritos, posibilita el análisis de materiales que datan de las primeras etapas de la formación del sistema solar.

La importancia del estudio de meteoros no se restringe al hallazgo de meteoritos. El análisis continuo en el tiempo ha logrado identificar que hay dos tipos de eventos: unos son periódicos y otros esporádicos. De acuerdo a las definiciones de [American Meteor Society \(2023\)](#) los eventos periódicos se asocian típicamente a una lluvia de meteoros, la cual ocurre cuando la órbita de la Tierra intersecta la órbita de una corriente meteórica (grupo de meteoroides, en general de origen cometario, que viajan juntos en órbita heliocéntrica).

Además, el análisis de bólidos tiene implicaciones importantes para la seguridad planetaria. Al mejorar nuestra comprensión de cómo se desintegran los meteoroides en la atmósfera, es posible desarrollar mejores modelos para predecir los posibles daños que podrían causar los grandes impactos de meteoritos. Esto es fundamental para evaluar los riesgos asociados a eventos de impacto y para el desarrollo de estrategias de mitigación. Un ejemplo destacado de estos esfuerzos es el proyecto DART (Double Asteroid Redirection Test) de la NASA, que se centró en probar la tecnología necesaria para desviar asteroides potencialmente peligrosos mediante el impacto cinético ([NASA, 2024a](#)).

Una gran ventaja que tiene el estudio de meteoros, frente a otras áreas de la astronomía, es que puede realizarse desde tierra, con equipos de fotografía y video relativamente accesibles. Por lo tanto, se han desplegado diversas redes de detección y monitoreo de estos fenómenos en el mundo (ver sección 3). Sin embargo, en el hemisferio sur hay menos cobertura de observación debido a la menor densidad de instalaciones de monitoreo y a la falta de una infraestructura histórica robusta en comparación con el hemisferio norte. Este déficit histórico hace que cualquier dato recopilado en el hemisferio sur

sea especialmente valioso para completar el panorama global de la distribución y frecuencia de los meteoros.

Una contribución a las redes de detección óptica de meteoros es la red uruguaya BOCOSUR. Esta red consiste actualmente de 20 estaciones equipadas con cámaras monocromáticas de alta resolución *all-sky* distribuidas homogéneamente en el territorio uruguayo (Caldas et al., 2024).

Uno de los principales retos que enfrentan las redes de detección de meteoros, como consecuencia de su crecimiento, es el análisis de un volumen de datos considerablemente elevado. Para manejar y analizar eficientemente estos grandes conjuntos de datos, se ha recurrido cada vez más a la automatización de procesos mediante técnicas de machine learning. Estas tecnologías permiten la clasificación automática de meteoros, la identificación de patrones en los datos y la predicción de eventos futuros. Por ejemplo, algoritmos de aprendizaje automático se utilizan para diferenciar entre meteoros y otros fenómenos, mejorar la precisión de las trayectorias calculadas y extraer características relevantes de las imágenes y videos capturados (ver apartado 3.1). Esta automatización no solo optimiza el procesamiento de datos, sino que también abre nuevas posibilidades para la investigación científica, permitiendo a los científicos centrarse en el análisis e interpretación de los resultados.

En resumen, el estudio de bólidos no solo amplía nuestro conocimiento del sistema solar, sino que también proporciona información crucial para la seguridad planetaria.

1.2. Objetivos

Este trabajo se inscribe en el esfuerzo continuo por mejorar nuestra comprensión de estos fenómenos, en particular procura contribuir al desarrollo del pipeline de manejo y procesamiento de datos de la red BOCOSUR, y a la obtención de sus primeros resultados científicos.

A continuación establecemos los objetivos específicos del presente trabajo:

- Contribuir al mantenimiento y operación de las estaciones BOCOSUR.
- Generar una base de datos de videos de todas las estaciones. Realizar una clasificación “manual” para obtener una base de datos de bólidos y no bólidos para la posterior fase de entrenamiento y validación de un modelo de *machine learning*.
- De ser posible, entrenar, validar y correr el clasificador AUTOBOL sobre la base de datos de videos. En el caso de que la adaptabilidad del código sea limitada por las características de los videos de alta resolución, realizar las modificaciones necesarias para cumplir el objetivo adecuado en esta etapa de la red BOCOSUR.
- A partir de los videos clasificados como bólidos iniciar una búsqueda de eventos multi-estación y hacer un estudio fotométrico y astrométrico de un caso real con implementación existente como la aplicación de Post Procesamiento (ver Caldas et al. (2024)).

1.3. Estructura de la monografía

El informe está organizado en diez secciones, donde la presente Introducción corresponde a la primera.

La segunda sección presenta la terminología elemental de la ciencia de meteoros que emplearemos en este trabajo, así como una descripción de la física involucrada en el fenómeno.

La tercera sección desarrolla en mayor profundidad la distribución de redes de detección de meteoros a nivel mundial.

La sección 4 consiste en una presentación de la red BOCOSUR, descripción del hardware y software de las estaciones de monitoreo y los datos que se generan a partir de la infraestructura actual. Asimismo, exponemos cómo se ha contribuido al mantenimiento remoto e in situ de las estaciones.

La sección 5 describe la construcción de una base de datos de la red BOCOSUR y los detalles de la clasificación “manual” inicialmente aplicada.

En la sección 6 presentamos las nociones básicas de aprendizaje supervisado para la formulación de un clasificador propio.

En las secciones 7 y 8 documentamos la adaptación de un clasificador basado en machine learning ya existente llamado AUTOBOL (creado con el fin de clasificar videos de la red BOCOSUR con un sistema de adquisición anterior). Además, incluimos una búsqueda temporal de eventos candidatos a pertenecer a las lluvias de meteoros mayores observables desde el hemisferio sur y una búsqueda de bólidos detectados desde varias estaciones simultáneamente (multi-estación).

La sección 9 resume los resultados obtenidos: los bólidos predichos por el nuevo clasificador y el análisis astrométrico y fotométrico de un evento multi-estación.

Por último, la sección 10 contiene las conclusiones del presente trabajo, y una descripción de los futuros pasos de la red BOCOSUR.

2. Terminología de la ciencia de meteoros

Partimos de las definiciones básicas de la física de meteoros aprobadas por el consenso de la Unión Astronómica Internacional (IAU) ([Unión Astronómica Internacional, 2017](#)):

1. **Meteoroide:** Es un objeto natural sólido de un tamaño aproximado entre 30 micrómetros y 1 metro que se mueve en el espacio interplanetario o proviene de él.
2. **Meteoro:** Es la luz y los fenómenos físicos asociados (calor, choque, ionización) que resultan de la entrada a alta velocidad de un objeto sólido del espacio en una atmósfera gaseosa.
3. **Meteorito:** Es cualquier objeto natural sólido que sobrevivió la fase de meteoro en una atmósfera gaseosa sin ser completamente vaporizado.
4. **Bólide:** Un meteoro más brillante que la magnitud visual absoluta¹ de -4 .
5. **Superbólide:** Un **meteoro** más brillante que la magnitud visual absoluta de -17 .
6. **Estela de meteoro:** Es la luz o la ionización que queda a lo largo de la trayectoria del meteoro después de que ha pasado.
7. **Corriente de meteoroides:** meteoroides con órbitas similares y de origen común.
8. **Lluvia de meteoros:** Es un grupo de meteoros producidos por meteoroides de la misma corriente de meteoroides.

La ablación, un proceso fundamental en este contexto, se refiere a la pérdida de masa de un meteoroides en cualquier forma y fase. Esto incluye la pérdida de masa sólida como fragmentos, líquida como gotas y gaseosa como vapor caliente. Esta última siempre forma parte de la etapa final y es responsable del fenómeno luminoso observado (meteoro). La radiación emitida por un meteoro se debe principalmente a la excitación de átomos metálicos, especialmente hierro, que alcanzan temperaturas de entre $3000K$ y $5000K$. Esta radiación constituye más del 90 % de la luz observada en los meteoros. Mientras el meteoroides se desplaza por la atmósfera y sufre ablación, su velocidad disminuye gradualmente ([Ceplecha et al., 1998](#)).

Un meteoroides en la atmósfera se convierte en un meteorito después de que cesa la ablación y el objeto continúa su vuelo oscuro hasta el suelo ([Unión Astronómica Internacional, 2017](#)).

En la figura 1 se ilustran los conceptos listados con anterioridad.

¹Se define la magnitud absoluta de un meteoro como aquella que presenta a 100 km de altura.

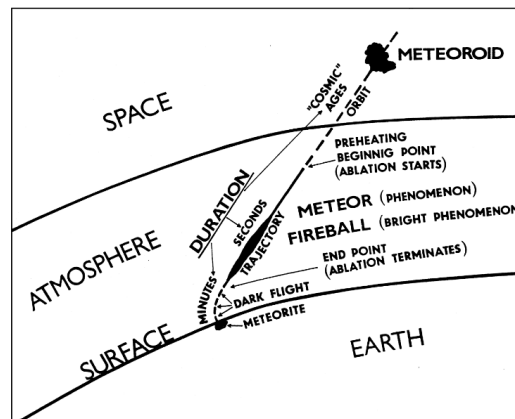


Figura 1: Terminología básica del estudio de meteoros (Fuente: [Ceplecha et al. \(1998\)](#)).

En lo que respecta al origen de los meteoroides y de las corrientes de meteoroides, se entiende actualmente que los principales cuerpos progenitores son asteroides o cometas. Como explica [Council \(1998\)](#), los asteroides, al igual que los cometas, son la población remanente de planetesimales, pequeños cuerpos primordiales a partir de los cuales se acumularon los planetas. Generalmente, los asteroides considerados son planetesimales relicto formados dentro y más allá del cinturón de asteroides, que se encuentra entre 2.2 y 3.2 UA del Sol, extendiéndose hasta los troyanos de Júpiter. Aquellos formados en ubicaciones más distantes suelen considerarse cometas.

Un aspecto clave que distingue a cometas y asteroides es su composición. Las definiciones históricas de asteroides se basan en la presencia o ausencia de “actividad cometaria”, que implica la presencia de compuestos volátiles, especialmente hielo de agua, cerca o en la superficie del objeto. “Watson et al. (1963) demostraron que el hielo de agua sublima rápidamente en períodos de tiempo cortos en comparación con la edad del sistema solar hasta la distancia de Júpiter” ([Council, 1998](#)). Por lo tanto, se espera que los objetos formados por una mezcla primitiva de hielos y materiales refractarios, situados a distancias asteroidales del Sol, hayan perdido sus hielos superficiales. En contraste, los objetos desviados hacia el sistema solar interior desde ubicaciones mucho más remotas, como el Cinturón de Kuiper, el Disco Disperso y la Nube de Oort, conservan estos volátiles cerca de sus superficies y muestran actividad “cometaria”, como la presencia de coma y colas, hasta que han residido en el sistema solar interior durante varios miles de años.

De acuerdo a lo mencionado anteriormente, las agrupaciones conocidas como corrientes de meteoroides se generan cuando asteroides o cometas se fragmentan y dispersan manteniendo una misma órbita, que puede ser la del cuerpo progenitor dependiendo de las interacciones gravitatorias que experimenta la agrupación. Si la órbita de dicha corriente intersecta la órbita de la Tierra, se desata un flujo de ingreso de esos meteoroides a la atmósfera terrestre. Esto ocasiona una lluvia de meteoros. Las lluvias de meteoros son fenómenos periódicos, dado que la Tierra intercepta la órbita de la corriente que los origina siempre en la misma época. [Porubčan et al. \(2004\)](#) explica que aunque la mayoría de las corrientes de meteoroides se asocian a los “escombros” de cometas (como Perseidas, Orionidas y Eta Aquaridas, etc.), se está ampliando el conocimiento de corrientes y lluvias asociadas a asteroides, donde un fuerte ejemplo son las Gemínidas (relacionado con el asteroide 3200 Phaethon).

Dado que los asteroides y los cometas tienen estructuras y composiciones distintas, implica que los mecanismos de formación de las corrientes de meteoroides también varían. La colisión de asteroi-

des puede producir la formación de una corriente de meteoroides, mientras que el mecanismo en los cometas está ligado a la sublimación de los hielos a medida que el núcleo cometario se acerca al Sol ([Whipple, 1951](#)). De esta forma, escapa gas que arrastra consigo a los meteoroides lejos del núcleo. Este proceso se observa principalmente cerca del perihelio de la órbita cometaria. Debido a que la velocidad de eyección es mucho menor que la velocidad orbital, los cambios en la energía y el momento angular por unidad de masa son mínimos, resultando en órbitas de meteoroides que son muy similares.

Debido a que los meteoroides asociados a cometas son principalmente partículas de polvo, los meteoritos suelen tener como cuerpo progenitor a fragmentos de asteroides. Existen diferentes tipos de asteroides y dan lugar a una gran variedad de análogos meteoritos ². Los meteoroides que no se asocian a ninguna corriente dan lugar a **meteoros esporádicos**, en contraposición de la periodicidad de los meteoros que pertenecen a lluvias. Los meteoros esporádicos de origen asteroidal, por su composición, tienen una mayor probabilidad de resultar en la caída de un meteorito.

([Ceplecha et al., 1998](#), p. 329-334) explica que, sin importar el origen de los meteoroides, existen límites en las velocidades que pueden alcanzar en colisión con la atmósfera terrestre. El límite inferior es la velocidad de escape de la Tierra 11.2km/s mientras que el límite superior corresponde a la suma de 42.5km/s de velocidad parabólica en el perihelio de la Tierra y la velocidad de la Tierra en el perihelio (30.3km/s), es decir, 72.8km/s .

Realizando una descripción física del fenómeno, [Ceplecha et al. \(1998\)](#) distingue entre 4 tipos de interacciones de un meteoroides con la atmósfera, que dependen de la masa del objeto y en menor medida, también de su velocidad:

1. Meteoros

Los meteoros típicos son causados por meteoroides de $\approx 0.01\text{mm}$. El tamaño exacto depende de la velocidad. A medida que estos cuerpos entran en la atmósfera, aumenta su temperatura rápidamente. Cuando la temperatura superficial alcanza aproximadamente 2200 K , generalmente entre 80 y 90 km de altura, el material del meteoroides comienza a sublimarse, llenando su entorno con vapores calientes. La mayor parte de la luz del meteorito proviene de la radiación de átomos metálicos poco excitados (temperaturas entre 3000K y 5000K). Durante este proceso de ablación, el meteoroides se desacelera muy poco.

2. Bóolidos y Caídas de Meteoritos

Ocurren cuando el meteoroides tiene un tamaño mayor a 20 cm . La ablación no desintegra al objeto por completo en el tiempo que tarda alcanzar una velocidad límite de $\approx 3\text{km/s}$. A esta velocidad, la energía no es suficiente y cesa el fenómeno luminoso. Los restos del meteoroides continúan desplazándose a velocidades y temperaturas menores (formando una costra de fusión). El vuelo oscuro puede llegar a durar minutos hasta dar lugar a un meteorito. Por ejemplo, un meteoroides de 20 cm que ingresa a velocidad de 15 km/s , corresponde a un bólido de magnitud -8 y a un remanente meteorito de $\approx 10\text{g}$.

3. Impactos Explosivos

Es el caso más extremo. El impactor corresponde a un meteoroides denso de tamaño mayor a varios metros. De esta manera, la desaceleración es mínima y el cuerpo impacta la superficie terrestre a velocidades de varios kilómetros por segundo. La emisión de luz por los vapores

²En la tabla 4.1 de [Council \(1998\)](#) “*Common Asteroid Types*” se presenta un esquema que condensa la relación entre tipos de asteroides y tipos de meteoritos.

del meteoroides continúa hasta el contacto con la Tierra, donde se forma un cráter de impacto (explosivo).

4. Partículas de Polvo Meteórico

Si el tamaño del meteoroides es menor a unas pocas centésimas de milímetro, se desacelera a menos de unos pocos kilómetros por segundo en la atmósfera y no alcanza a producir un fenómeno luminoso (se detectan generalmente por radar). Estas partículas de polvo meteórico se desplazan lentamente a través de la atmósfera sin cambios significativos hasta llegar a la superficie.

La dirección de la que proviene un meteoroides se denomina tradicionalmente **radiante**. Este término se originó de la observación de trayectorias casi paralelas de meteoroides pertenecientes a una misma corriente. Como resultado, las trayectorias aparentes de los meteoros que forman parte de la misma lluvia, parecen irradiar desde un único punto en el cielo, conocido como el radiante de la lluvia (puesto que la dirección está dada por la velocidad relativa meteoroides-Tierra).

Los meteoroides también pueden experimentar fragmentación en su descenso por la atmósfera. Este es un fenómeno que introduce complejidades adicionales en la observación de meteoros. Hay dos tipos principales de fragmentación: continua y súbita. La fragmentación continua es el proceso gradual de pérdida de masa del meteoroides mientras atraviesa la atmósfera. En contraste, la fragmentación súbita ocurre abruptamente en un punto específico, dividiendo el meteoroides en varios fragmentos grandes (Ceplecha & Revelle, 2005).

Estos procesos afectan significativamente la dinámica de los meteoroides. La fragmentación continua fragmenta el meteoroides en pequeñas partículas, reduciendo gradualmente su masa. La fragmentación súbita, también conocida como fragmentación discreta, altera drásticamente la trayectoria del meteoroides al dividirlo de manera repentina. Comprender estos fenómenos es esencial para mejorar los modelos de movimiento, pérdida de masa y radiación de los meteoroides durante su entrada en la atmósfera terrestre (Ceplecha & Revelle, 2005).

En Ceplecha et al. (1998) se presentan los pilares de la física de meteoros, estableciendo las ecuaciones de movimiento del meteoroides considerando los distintos procesos que experimenta como precalentamiento, ablación, vuelo oscuro e impacto. El modelo más simple, que describe un meteoroides que no se fragmenta (*single non-fragmenting body*), es representado por tres ecuaciones: una de arrastre o *drag* del objeto, una de pérdida de masa (ablación) y otra de altura (totalmente geométrica). Estos modelos teóricos fueron refinándose para incluir la descripción de procesos de fragmentación, como en el trabajo de Ceplecha & Revelle (2005).

En resumen, las características de la colisión de un meteoroides con la atmósfera terrestre, hacen posible estudiar al fenómeno a partir de distintas técnicas. Generalmente se realizan detecciones ópticas (fotografía o video), se emplea radar para estudiar meteoros poco brillantes o partículas de polvo meteórico y en el caso de impactos se pueden incluir detecciones sísmicas. En la siguiente sección profundizaremos acerca de los métodos empleados por las principales redes de detección de meteoros.

3. Redes de detección de meteoros

Las redes de detección de meteoros son sistemas diseñados para el monitoreo y estudio de los meteoroides y sus interacciones con la atmósfera terrestre. Estas redes utilizan una variedad de instrumentos, incluyendo cámaras ópticas y de radar, para detectar y analizar la trayectoria, velocidad y composición de los meteoros. A continuación se describen algunas de las principales redes de detección de meteoros en el mundo, con énfasis particular en las redes ópticas presentadas en la siguiente tabla:

Nombre de la red	Fuente	Ubicación	Desde	Estaciones
European Fireball Network (EN)	Spurný et al. (2017)	Europa	1963	16+
Southern Ontario Meteor Network	Weryk et al. (2008)	Canadá	2004	5+
Spanish Meteor Network (SPMN)	Trigo-Rodríguez et al. (2004)	España	2004	4+
Desert Fireball Network (DFN)	Devillepoix (2020)	Australia	2007	50+
NASA All-Sky Fireball Network	NASA (2024b)	EE. UU.	2008	17
FRIPON	Colas et al. (2020)	Francia y otros	2014	100+
PRISMA	Gardiol et al. (2020)	Italia	2016	50+
Global Meteor Network (GMN)	Roggemans et al. (2024)	Internacional	2017	450+
Brazilian Meteor Observation Network (BRAMON)	Amaral et al. (2018)	Brasil	2014	30+

Tabla 1: Principales redes de detección *all-sky* de meteoros en el mundo. La columna fuente hace referencia al documento de donde se extrae la fecha de fundación de cada red y la cantidad de estaciones que la compone.

La *European Fireball Network* (EN) es una de las redes de detección de meteoros más antiguas y extensas de Europa, centrada principalmente en la observación de bólidos y la recuperación de meteoritos. En 1998 ya contaban con más de 34 cámaras *all-sky*, desplegadas con un espaciamiento de $\approx 100\text{km}$ y cubriendo un área de alrededor de $10\text{e}6\text{km}$ en la República Checa, Eslovaquia, Alemania, así como partes de Bélgica, Suiza y Austria ([Oberst, 1998](#)). En publicaciones posteriores, como [Spurný et al. \(2017\)](#) reportan una actualización de los equipos utilizados en 16 de sus estaciones llamados DAFO o *Digital Autonomous Fireball Observatory*. De esta manera, transicionaron a un sistema de cámaras digitales que les permite realizar detecciones más sofisticadas. Las contribuciones de la red abarcan desde la detección de bólidos y recolección de meteoritos (desde sus orígenes) hasta el análisis de variaciones del flujo de lluvias de meteoros.

La red canadiense *Southern Ontario Meteor Network* ha desarrollado una red automatizada de sistemas de video CCD *all-sky* para detectar meteoroides de tamaño mediano a grande que se desintegran sobre el sur de Ontario, Canadá. El sistema consta de 7 estaciones ([Group, 2024](#)), con una distancia máxima entre ellas de 180 km. El objetivo principal de la red *all-sky* es actuar como el disparador principal para la Red de Meteoros del Sur de Ontario, que incluye instrumentos de radar, infrasonido y ópticos. El objetivo secundario de la red es proporcionar órbitas y datos atmosféricos de cualquier meteoroides, caracterizar los aumentos en los meteoroides grandes de corrientes específicas y estudiar la distribución orbital general y las características físicas de los meteoroides del tamaño de

un centímetro (Weryk et al., 2008).

Trigo-Rodríguez et al. (2004) describen los inicios de la red española SPMN. En el comienzo del despliegue de dicha red, a finales de 2004, contaban con 4 estaciones. Cada estación cuenta con dos cámaras que operan durante 30 segundos en exposiciones alternas, para evitar la pérdida de cobertura en el tiempo que tarda en descargarse la imagen tomada. Esperan que el desarrollo de la red permita recuperar meteoritos en regiones de la Península Ibérica y el norte de África.

La red australiana DFN (*Desert Fireball Network*), establecida en 2007 en colaboración internacional, es la primera red de detección de meteoros en el hemisferio sur. Consta de cuatro observatorios *all-sky* autónomos en Australia Occidental. Ha llevado a la recuperación exitosa de dos caídas de meteoritos. Hasta la fecha, la Red ha registrado más de 550 bólidos, permitiendo la determinación de precisas órbitas y trayectorias para alrededor de 150 de ellos. Además de las caídas recuperadas, se estima que otros doce bólidos han resultado en caídas de meteoritos (Bland et al., 2012). Esta red presenta una iniciativa de ciencia ciudadana, llamada “Fireballs in the Sky” (Day et al., 2017). Amplía el alcance del proyecto al reclutar observadores ciudadanos en todo el mundo para hacer observaciones visuales de bólidos y registrar y enviar sus observaciones a través de una aplicación de realidad aumentada gratuita para dispositivos móviles. El proyecto se encuentra actualmente en pausa mientras se expande de la red australiana DFN al *Global Fireball Observatory*.

A mediados de 2008, MEO (*Meteoroid Environment Office* de NASA) estableció la Red de Bólidos All Sky de la NASA (NASA AllSkyFireballNetwork), una red de cámaras de meteoros en el sur de Estados Unidos. Los objetivos de esta red de video son establecer la distribución de velocidad de los meteoroides del tamaño de centímetros, determinar qué fuentes esporádicas producen partículas grandes, determinar órbitas (de baja precisión) para meteoros brillantes, intentar descubrir el tamaño a partir del cual las lluvias comienzan a dominar el flujo de meteoroides, monitorear la actividad de las principales lluvias de meteoros y ayudar en la localización de caídas de meteoritos (Cooke & Moser, 2012).

La red *all-sky* FRIPON (*Fireball Recovery and InterPlanetary Observation Network*), localizada principalmente en Francia y Europa, se dedica a la detección y recuperación de meteoritos. FRIPON utiliza más de 100 cámaras para monitorear el cielo y localizar bólidos, con el objetivo de recuperar meteoritos (Colas et al., 2020).

La red PRISMA (*Prima Rete per la Sorveglianza sistematica di Meteore e Atmosfera*) es una red italiana *all-sky* que se inspira en los esfuerzos de FRIPON y busca lograr la misma cobertura en Italia. Los principales objetivos de PRISMA son la detección de bólidos y recuperación de meteoritos (Gardioli et al., 2016).

La *Global Meteor Network* (GMN) es una red colaborativa internacional que utiliza cámaras *all-sky* para la detección de meteoros, con el objetivo de aumentar la cobertura global y proporcionar datos en tiempo real. Fundada y coordinada por Denis Vida y otros investigadores, la red ha implementado un sistema estandarizado para la recopilación y el análisis de datos meteóricos. Este sistema incluye software de código abierto y cámaras de bajo costo, lo que facilita la participación de astrónomos aficionados y profesionales de todo el mundo (Vida et al., 2021).

Otra gran contribución a la detección de meteoros en el hemisferio sur es la red brasileña BRA-

MON. El trabajo de [Amaral et al. \(2018\)](#) presenta a la red y la historia de la misma. Comienza como un conjunto de intentos de construir una red propulsada por astrónomos *amateurs*, actualmente sigue siendo mantenida por voluntarios. BRAMON comenzó a operar en 2014 con 9 estaciones y creció a 23 en 2015. A finales de 2016, la red consistía en 39 estaciones repartidas en 16 estados, ubicadas en la región sureste de Brasil. En la región nororiental las estaciones se concentran cerca de la costa, ya que el interior de esta zona tiene un clima semiárido. Tienen como objetivo determinar radiantes de lluvias de meteoros.

Por el momento, hemos mencionado únicamente redes de detección ópticas de meteoros. Como explica [Ceplecha et al. \(1998\)](#), los meteoroides generan una estela de gas ionizado llamada “incandescencia meteórica”. Esta incandescencia puede reflejar las ondas electromagnéticas emitidas por un radar. La detección de meteoros con esta tecnología no se ve afectada por condiciones atmosféricas (con la excepción de tormentas eléctricas) o por la luz solar. Suele emplearse para detectar meteoroides muy pequeños que, con el método óptico, no sería sencillo.

La *Canadian Meteor Orbit Radar* (CMOR) es una red de radar operativa desde finales de 2001, diseñada para monitorear y estudiar la actividad meteórica. CMOR consiste en tres estaciones de radar que operan a una frecuencia de 29.85MHz , ubicado cerca de Tavistock, Ontario. Esta red de radar tiene una mayor área de recolección efectiva en comparación con el *Advanced Meteor Orbit Radar* (AMOR) de Nueva Zelanda, pero es menos sensible. Esta característica hace que CMOR sea más adecuada para el estudio de corrientes de meteoros, ya que estas tienden a ser relativamente más ricas en partículas más grandes en comparación con el complejo de meteoroides esporádicos ([Jones et al., 2005](#)).

Las redes descritas anteriormente dejan en evidencia la falta de cobertura de estos fenómenos en el hemisferio sur. Por otro lado, el desarrollo en el hemisferio norte, con una gran variedad de objetivos en la ciencia de meteoros, sirve de referencia para futuros proyectos en regiones aún no monitoreadas.

3.1. *Machine-learning* en redes de detección de meteoros

Como explica [Peña-Asensio et al. \(2023\)](#) el aumento de estaciones de detección de meteoros ha incrementado enormemente los datos de video e imágenes disponibles, lo que ha llevado a muchas redes a adoptar la automatización para gestionar eficientemente estos volúmenes de datos. De esta manera, emplean *machine learning* tanto para el seguimiento de un meteorito en una imagen (denominado usualmente como *tracking*), así como para la detección o también la clasificación de observaciones generadas.

Dentro de las técnicas de *machine learning* empleadas en el procesamiento de imágenes, se suele implementar *deep learning* para problemas complejos y bases de datos muy grandes. Esta técnica se caracteriza por el uso de redes neuronales profundas, inspiradas en el cerebro humano [Amr \(2020\)](#).

Además, en problemas de *deep learning* es común aplicar *transfer learning* o aprendizaje por transferencia, que consiste en utilizar un modelo previamente entrenado en una tarea específica como punto de partida para entrenar un modelo en una nueva tarea relacionada. Esta metodología es especialmente útil cuando se dispone de un conjunto de datos limitado para la nueva tarea, pero se tiene acceso a un modelo robusto entrenado en una gran cantidad de datos para una tarea similar ([Peña-Asensio et al., 2023](#)).

En la siguiente tabla, se resume qué redes utilizan *transfer learning*, con qué propósito (*Goal*) y los valores de algunas métricas de evaluación (que serán desarrolladas en la sección 6) de los modelos de aprendizaje automático desarrollados.

Article	Meteor network	# samples	Goal	Transfer learning	Model	F1 score	Accuracy
Ganju et al. (2023)	CAMS	19,152	Detection	BiLSTM	-	0.91	0.89
Anghel et al. (2023)	MOROI	9,858	Detection	-	CNN, GB and RF	0.989	0.998
Sennlaub et al. (2022a)	AllSky7 FNG	20,000	Detection	ImageNET	CNN: GRU and SVM	-	0.991
Cecil & Campbell-Brown (2020)	CAMO	50,745	Tracking	No	CNN: multi-layer and max pool	0.997	0.998
Gural (2019)	CAMS	200,361	Tracking	MeteorNet	RNN and LSTM	0.974	0.981
Marsola & Lorenna (2019)	EXOSS	400	Detection	ImageNET	VGG16	-	0.8435
Vítek & Nasyrova (2019)	MAIA	-	Tracking	No	DPT	-	-
Galindo & Lorenna (2018)	EXOSS	1,660	Detection	Fashion-MNIST	-	0.94	0.960

Tabla 2: Algunas redes de detección de meteoros que emplean *machine learning* y para qué propósito. Fuente: tabla 1 de (Peña-Asensio et al., 2023, p. 4).

La necesidad de implementar técnicas sofisticadas de aprendizaje automático surgen cuando los datos a analizar superan la capacidad humana de inspección manual. En la tabla 2 se explicita el número de muestras utilizadas en el entrenamiento del modelo de *machine-learning* de cada red.

El trabajo de Sennlaub et al. (2022b), que se centra en la clasificación de meteoros en videos de la red alemana AllSky7, es un gran ejemplo de los distintos usos que se le puede dar al aprendizaje automático en la ciencia de meteoros. Cabe destacar que, para hallar *clusters* en los datos de las distintas clases (categorías de lo que se detecta en los videos como meteoros, animales, etc.) y seleccionar características que permitan distinguir entre las mismas, recurren al uso de redes neuronales. Calculan qué tan bien se pueden agrupar los “no-meteoros” (sin usar la información de las etiquetas durante el entrenamiento) empleando una red neuronal convolucional (CNN) como un extractor de características.

A medida de que se expandan las redes actuales y surjan otras nuevas, la cantidad de datos creados hará que sea ineludible el uso de estas técnicas.

4. La red BOCOSUR

La red nacional de detección de bólidos BOCOSUR es un proyecto liderado por el Departamento de Astronomía de Facultad de Ciencias, y está compuesta por 20 estaciones con cámaras *all-sky* instaladas en centros de educación media de todo el país, así como por infraestructura alojada en Facultad de Ciencias, UDELAR (Estación 1).

La red BOCOSUR se desplegó con el fin de contribuir a la detección de meteoros y bólidos en el hemisferio sur, proporcionando cobertura total en el territorio uruguayo. Esta red tiene como objetivo estudiar meteoros brillantes esporádicos, al igual que aquellos asociados a corrientes de meteoroides.

Las estaciones se encuentran situadas a lo largo del país, con una separación de aproximadamente 120 km entre sí. La densidad de la red permite la observación de eventos multi-estación, estos son necesarios para la determinación de trayectoria y órbita del meteoróide. Los principales requerimientos de una estación de BOCOSUR son el acceso a la red eléctrica y a Internet. Además, se priorizó la búsqueda de ubicaciones que albergaran una comunidad activa de docentes apasionados por la Astronomía y dispuestos a participar en el proyecto, colaborando con el mantenimiento de las estaciones. La numeración y ubicación de las estaciones se encuentran señaladas en la tabla 3.

La instalación completa de la red tuvo lugar entre abril de 2020 y febrero de 2023. Se comenzó por el despliegue de la red en la región sur del país y cuando se encontraban instaladas las primeras 6 estaciones, durante el año 2022, el proyecto estuvo marcado por la actualización de los sistemas de adquisición utilizados.

Se realizó una transición de sistemas de video analógicos de baja resolución basados en una cámara **WAT-902H2 ULTIMATE** (que de ahora en más llamaremos *WATEC*), a cámaras digitales de alta resolución **ZWO ASI 178MM** (que llamaremos *ASI*). En la tabla 4 ³ se ilustran las principales diferencias entre los dos modelos de cámaras utilizados, que apoyan la decisión de la actualización del equipo.

³El valor de SNR relativo y tamaño de los videos corresponde a un resultado experimental del proyecto PAIE edición 2020 “Calibración astrométrica y fotométrica para cámaras all-sky ” <https://www.estudiantes.csic.edu.uy/2022/04/27/calibracion-astrometrica-y-fotometrica-para-camaras-all-sky/>. Los costos refieren al momento de evaluación de su adquisición.

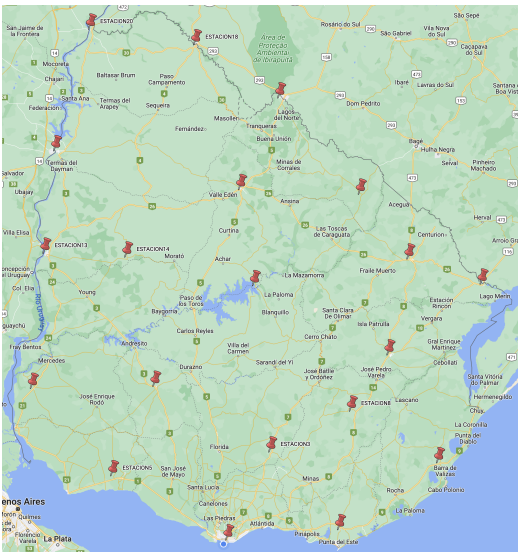


Figura 2: Distribución espacial de las 20 estaciones de la red BOCOSUR.

Número	Institución	Ciudad	Departamento
1	Facultad de Ciencias, Udelar	Montevideo	Montevideo
2	Liceo 1 “Monseñor Mariano Soler”	San Carlos	Maldonado
3	Liceo “Ramón Goday”	Casupá	Florida
4	Liceo “José Aldunate Ferreira”	Castillos	Rocha
5	Liceo “Agustín Urbano Indart Curuchet”	Rosario	Colonia
6	Liceo 1 “Dr. Roberto Taruselli”	Dolores	Soriano
7	Liceo 1 “Carlos Brignoni Mosquera”	Trinidad	Flores
8	Estación Agraria, UTU	Pirarajá	Lavalleja
9	Liceo 1 “Dr. Nilo L. Goyoaga”	Treinta y Tres	Treinta y Tres
10	Liceo “Lucio Gabino Núñez”	San Gregorio de Polanco	Tacuarembó
11	Liceo 1 “Juana de Ibarbourou”	Melo	Cerro Largo
12	Liceo “Dr. Aníbal Acosta Estapé”	Río Branco	Cerro Largo
13	Liceo 1 “Química Farmacéutica Elida Heinzen”	Paysandú	Paysandú
14	Liceo de Guichón	Guichón	Paysandú
15	Liceo 1 “Ildefonso Pablo Estéves”	Tacuarembó	Tacuarembó
16	Liceo de Vichadero	Vichadero	Rivera
17	Liceo 1 “Instituto Politécnico Osimani y Llerena”	Salto	Salto
18	Liceo 2	Artigas	Artigas
19	Liceo 5	Rivera	Rivera
20	Liceo 1 “Escribano Diego Carlos Muguruza”	Bella Unión	Artigas

Tabla 3: Denominación y ubicación de las estaciones de BOCOSUR.

Características	Watec 902 H2 Ultimate	ZWO ASI 178MM
Número máximo de pixeles	720x576	3096x2080
Diámetro del círculo efectivo	576	2080
# Píxeles por hemisferio	2.61e5	3.40e6
Escala (arcmin)	16.9	4.7
<i>fps</i> (1/s)	25	30
Exposición (ms)	16.7	33.3
SNR	x1	x6-7
Tamaño video (5s)	40 MB	600 MB
Hardware captura	Capturadora	USB
Costo aproximado	USD 460	USD 350

Tabla 4: Comparación de las cámaras ZWO ASI 178MM y Watec 902 H2 Ultimate. Además de las especificaciones de las cámaras se aclara el dispositivo de captura y un estimado de los costos.

[Caldas \(2012\)](#) describe en gran detalle el proceso de diseño de las estaciones BOCOSUR, sus antecedentes, al igual que el programa de desarrollo propio para la detección de meteoros.

La construcción e instalación de las estaciones también fue llevada a cabo por miembros de BOCOSUR, incluyendo la contribución de esta autora en el marco de las pasantías realizadas asociadas al proyecto (con principal énfasis en el período comprendido entre abril de 2023 y el presente).

A continuación, desarrollaremos en mayor profundidad el hardware y software actual de las estaciones. Asimismo, describiremos el funcionamiento de las mismas y las actividades de mantenimiento realizadas (en los apartados 4.4 y 4.5), que son parte del presente trabajo.

4.1. Hardware de las estaciones BOCOSUR

Una estación, en su descripción elemental, consiste de una caja estanca que protege a la cámara y la miniPC que la controla. En la presentación de la red BOCOSUR, [Caldas et al. \(2024\)](#) describe los componentes principales del hardware (representados en la figura 3 con sus respectivas referencias en la tabla 4). La configuración óptica de las estaciones incluye una lente ojo de pez de $2.5mm$, que tiene un campo de visión (FOV) de $\approx 170^\circ$, cubierta por un domo de policarbonato con revestimiento duro. La cámara se conecta a una nano PC local (CPU i5-8279U, 2.40 GHz, 16 GB de RAM) vía USB 3.0. Cuentan también con calentador y enfriador (ventilador) de CCTV. Además, en el exterior de la caja se sitúa un receptor GPS conectado a la PC, proporcionando lecturas precisas de ubicación y tiempo. Las últimas adiciones al modelo incluyen un módulo de monitoreo de temperatura, humedad y calentador basado en una placa Arduino Nano y un sensor DHT11. Las estaciones están conectadas a la red de internet del instituto mediante Ethernet y tienen la posibilidad de conexión Wi-Fi.

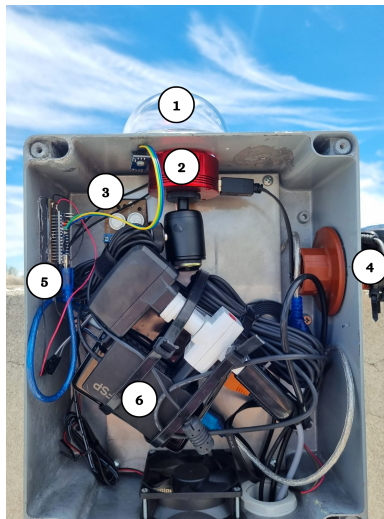


Figura 3: Componentes básicos de una estación BOCOSUR.

Referencia	Componente	Descripción
1	Domo	Policarbonato
2	Cámara	Conectada a la mini PC
3	Sistema de calentamiento	Regulador de temperatura conectado a una resistencia
4	GPS	Permite recuperar ubicación y tiempo precisos
5	Arduino	Registra temperatura y humedad
6	mini PC	Conectado por cable a la red de Ethernet

Figura 4: Tabla de componentes ilustrados en la figura 3.

A pesar de que no se observa en la imagen anterior, un elemento importante que fue incorporado en la mayoría de las estaciones es el *Smart Switch Sonoff*, un dispositivo de automatización que permite encender y apagar la estación conectados desde cualquier lugar utilizando una aplicación móvil. Esta característica permite reiniciar la estación de forma segura y remota.

El diseño de la estación presenta 4 orificios en la caja estanca para la cámara y el domo, ventilación, el cableado y ventilador. Todos los orificios, menos el de la cámara, se encuentran asegurados por una malla que previene el ingreso de polvo o insectos. Una vez cerrada la caja, es reforzado su aislamiento con la aplicación de silicona neutra y poliuretano expandido.

4.2. Software de las estaciones BOCOSUR

Dentro de cada estación se encuentra una miniPC cuyo sistema operativo actualmente es Windows 10. Hay 5 principales programas y códigos que se ejecutan en una de ellas y a continuación describimos cuál es su función.

- **bolidosGUI:** detección de meteoros.
- **status_estacion.py:** reporte del estado o *status* de la estación. Es parte del conjunto de *scripts* llamado Sentinela.

- **Anydesk y TeamViewer:** escritorio remoto.
- **FileZilla:** transferencia de archivos.

El programa más importante que debe estar ejecutándose es *bolidosGUI*, la aplicación de detección de meteoros desarrollada por Manuel Caldas en MATLAB en el marco de su tesis de maestría (Caldas, 2012) y que en Caldas et al. (2024) reporta algunas actualizaciones. El algoritmo de detección registra movimiento considerando la diferencia entre dos fotogramas consecutivos. En la diferencia de imagen, se calcula el número ($nDiff$) de píxeles que tienen un brillo por encima de un cierto umbral (thr), y su distancia media a su centroide (rm). Se activa un “evento” (grabación) mientras $nDiff$ supere un umbral de cantidad de puntos (evitando la detección de objetos luminosos con muy pequeño desplazamiento) y rm sea menor que cierto umbral de distancia (evitando la detección de objetos extendidos en movimiento). El algoritmo continúa adquiriendo fotogramas en el búfer de memoria, hasta que el video llega a una longitud ajustable (20s actualmente).

Cabe destacar que **bolidoGUI** fue desarrollado para operar durante la noche. Un parámetro ajustable es la elevación solar a partir de la cual puede generar eventos. Asimismo, la aplicación controla los parámetros de la cámara. En el modo actual de las cámaras ASI, se aplica *binning 2x*, lo que altera los valores presentados en la tabla 4. Con esta configuración, el tamaño de los fotogramas es de 1548x1040 píxeles, la tasa de fotogramas (*frame rate*) es de 16 *fps* y el tiempo de exposición aumenta a 62.5 ms.

La aplicación de detección genera tres archivos para cada evento detectado. El principal archivo es un video (.avi) de máxima calidad, con un nombre de archivo que incluye el ID de la estación (X) y la marca de tiempo del primer fotograma “*Station_X_yyyy-mm-dd-HH-MM-SS*”. Además, se crea un archivo de metadatos de MATLAB (.mat), con nombre análogo “*Event_X_yyyy-mm-dd-HH-MM-SS*”, que incluye la marca de tiempo del primer fotograma, la hora absoluta de cada fotograma y la diferencia entre los tiempos GPS y CPU. Por último, se añade una fila a una hoja de cálculo (.xlsx) en la que se registran los eventos de una noche y facilita las actividades de clasificación “manual” que son desarrolladas en la sección 5.

Además, durante la noche, **bolidosGUI** toma una imagen o *snapshot* del cielo cada tres minutos. Esta imagen es transferida al servidor central en Facultad de Ciencias a través de un *script* local (*status_estacion.py*). En adición a ese reporte visual, este código envía información sobre si **bolidosGUI** está operativa, la conectividad con el GPS, la diferencia horaria entre el GPS y la hora interna del PC, la temperatura, la humedad dentro de la estación y si el calentador del domo está encendido o apagado, al igual que el espacio libre en el disco.

Luego, **Anydesk** es el escritorio remoto que utilizamos para acceder a la miniPC de las estaciones. Cuenta con una licencia *Standard* que permite el acceso a la totalidad de estaciones de forma rápida y segura. Como acción de contingencia, hemos instalado de forma complementaria el análogo **TeamViewer**, dado que encontramos inconveniencias puntuales con el primero (bloqueo del acceso por este medio). Los problemas de conexión mediante **Anydesk** requirieron visitar las estaciones en cuestión para solucionarlos.

Actualmente, el mecanismo responsable de transferencia de archivos al servidor local es **Filezilla** que se basa en el protocolo FTP.

4.3. Operación de la red BOCOSUR

Decimos que las estaciones se encuentran totalmente operativas cuando el hardware y software funcionan de manera adecuada. Podemos asegurar su correcto funcionamiento a partir de la verificación por conexión remota y la información registrada por el grupo de códigos llamado Sentinela.

Sentinela consiste de 4 códigos, donde **status_estacion.py** es el único que corre en una estación, el resto (**bolidosEstado.sh**, **bolidosImágenesWeb.sh**, y **bolidosGifAnimadoNoche.sh**) son ejecutados en el servidor local localizado en Facultad de Ciencias. Ofrecen una comunicación entre las estaciones, el servidor local y la página web del proyecto BOCOSUR ⁴. El primero toma los resúmenes recolectados de las estaciones para mostrar en la página web, el segundo toma los *snapshots* creados en cada estación y agrega información para su presentación en la página web y el tercero se ejecuta todos los días al mediodía UT para generar un gif animado a partir de los *snapshots* de la noche anterior.

Por otra parte, el servidor local también recibe y ordena los archivos generados por cada evento detectado en las estaciones. En el apartado 5 se describen los datos disponibles.

Los problemas en la transmisión de datos estación-servidor-página web suelen radicar en el funcionamiento de la primera. Las estaciones requieren un monitoreo periódico, tanto remoto como in situ (visitas). El desarrollo de estas actividades de mantenimiento de la red están alineadas con el primer objetivo específico planteado para este trabajo.

4.4. Mantenimiento remoto de las estaciones

A pesar de que las estaciones fueron diseñadas para tener gran autonomía, el mantenimiento remoto mediante **Anydesk** o **TeamViewer** es una práctica regular que incluye las siguientes tareas.

- Actualizaciones del sistema (Windows) y otros programas:
Como cualquier PC, una actividad básica de mantenimiento implica corroborar que el sistema se encuentre en las condiciones necesarias para que el software implementado pueda operar correctamente. Esto involucra la actualización de drivers, donde son de particular importancia los que facilitan la comunicación entre PC y cámara.
- Corroboración del funcionamiento de componentes de la estación: Debemos comprobar principalmente que la cámara (desde el software de los fabricantes de la cámara *ZWO ASI Studio* ⁵ o la aplicación de detección), el GPS (desde un monitor serial) y la placa Arduino (desde **status_estacion.py**).
- Minimizar la diferencia entre tiempo PC y tiempo GPS: Sincronizamos regularmente la hora a la de un servidor NTP asociado a la infraestructura de Secundaria.
- Verificación del funcionamiento adecuado de **bolidosGUI** y **Sentinela**:
Ambos programas están configurados para ejecutarse automáticamente al inicio del sistema. Esta medida asegura la continuidad de la operación ante las interrupciones frecuentes del suministro eléctrico que afectan a las estaciones. No obstante, en reiteradas ocasiones hemos enfrentado inconveniencias con el funcionamiento de **bolidosGUI**. La causa principal es la creación continua de eventos (cientos por noche) que provocan el eventual fallo de la aplicación. La creación de esta

⁴Ver pestañas “LA ÚLTIMA NOCHE” y “ESTADO DE LAS ESTACIONES” <http://bolidos.astronomia.edu.uy>

⁵<https://astronomy-imaging-camera.com/news/what-is-asistudio-and-what-can-it-bring-us/>

cantidad de eventos puede tener distintos orígenes, algunos asociados a la desconfiguración de parámetros regulables de la aplicación (exposición y ganancia de la cámara o umbrales como los descritos en el apartado 4.2) o la presencia de objetos en movimiento como animales o lluvia que no son descartados por el algoritmo de detección. En estos casos, suele resolverse el problema al reiniciar la aplicación.

- Eliminación de eventos espurios previa a la transferencia al servidor:
Un video de un evento detectado tiene un peso de $\approx 30MB$. En una noche típica se crean entre 30 y 50 eventos, esto corresponde a $\approx 1.5GB$. Cuando algún desperfecto de la operación de la aplicación **boldosGUI** desata la creación descontrolada de eventos, se satura el disco duro de almacenaje de la PC (discos de 223 GB o 476 GB dependiendo de la estación). Entonces, descartamos los videos de aquellas noches que superan un peso de $\approx 5GB$, conteniendo más de 170 videos con alta probabilidad de ser espurios.
- Planificación de la transferencia de archivos:
Este es un proceso que podría automatizarse en otras condiciones. Sin embargo, puesto que las estaciones se encuentran conectadas a Ethernet de centros educativos, es necesario programar la transferencia de archivos en horarios y fechas que contemplen el calendario de actividades de la institución. Entonces, buscamos que el consumo de ancho de banda no perjudique el desarrollo de sus actividades. De la misma manera, se evita saturar al servidor receptor. Por lo tanto, las transferencias de archivos se realizan de forma fraccionada durante el transcurso de las noches o fines de semana.

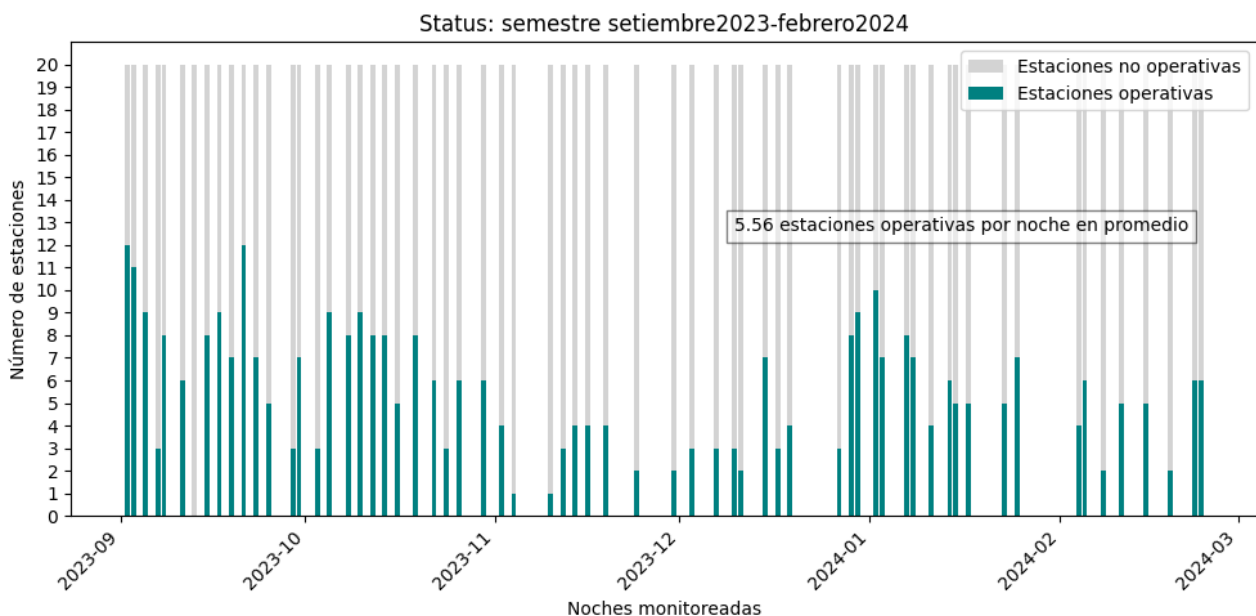


Figura 5: Estado de las estaciones en el semestre con inicio en setiembre 2023 y febrero de 2024.

El mayor obstáculo del monitoreo remoto de las estaciones es la intermitencia de la conexión a internet. Sin una buena conectividad, no es posible acceder vía **Anydesk** o **TeamViewer**. En la figura 5 evaluamos el estado de las estaciones en 64 noches comprendidas entre setiembre de 2023 y febrero de 2024. Considerando como operativas aquellas estaciones con conexión por el escritorio remoto donde la aplicación de detección se encuentra funcionando de manera adecuada, es decir, aquellas

donde no se requiere intervención más allá de las medidas preventivas listadas en las tareas anteriores. Encontramos que $\approx 5.6\%$ de las estaciones se encuentran operativas por noche.

Si los problemas radican en la aplicación de detección, la situación es rápidamente corregible pero, si depende de la conectividad de la estación, está fuera de nuestro control directo.

4.5. Mantenimiento in situ de las estaciones

El mantenimiento físico de las estaciones es una tarea que preserva la infraestructura de la red. La caja estanca ha permitido mantener los equipos aislados de condiciones meteorológicas adversas, pero de todas formas hemos encontrado diversas razones por las que una estación requiere de una visita.

Los principales motivos refieren a la conexión de Ethernet. En su defecto, la miniPC cuenta con un puerto RJ45 donde aseguramos la conexión por cable pero en varias ocasiones se ha estropeado. Por lo tanto, recurrimos a conectar un adaptador USB a RJ45. En estaciones donde el cableado expuesto a la intemperie es largo, hemos encontrado fracturas en el mismo que interrumpen la óptima comunicación.

Por otro lado, un componente que requiere ser cambiado periódicamente es el domo de policarbonato que protege la cámara de la estación. En la figura 6 podemos apreciar los efectos de la fotodegradación que, no actúan solo sobre el domo, sino también sobre la cámara que tiene originalmente un color rojo intenso. El estado del domo se puede evaluar directamente mediante la previsualización en **boldosGUI**.



Figura 6: Visita a la estación 8 (14/12/23).



Figura 7: Visita a la estación 9 (9/5/2024).

Imágenes tomadas en salidas de mantenimiento de las estaciones con motivo principal de recuperar conexión vía Anydesk.

Como se menciona en la Sección 4.2, la configuración de Anydesk ha experimentado modificaciones que, en ciertas ocasiones, resultan en el bloqueo del acceso desatendido, es decir, aquel que no requiere la autorización del usuario remoto. Por ejemplo en diciembre de 2023, esa fue la causa de la visita a la estación 8 o en mayo de 2024 ocurrió para las estaciones 9 y 12. Para evitar este inconveniente en el futuro, se ha incorporado recientemente **TeamViewer** a modo de respaldo.

En la siguiente tabla se reportan las 20 visitas de mantenimiento de las estaciones 1 a la 14, realizadas en el marco de las pasantías del proyecto BOCOSUR y este trabajo por la autora.

Estación	Fecha de visitas de mantenimiento
1	5/4/2024
2	9/9/2021, 25/03/2022, 13/12/2023
3	11/02/2022
4	10/09/2021, 13/12/2023
5	30/9/2021, 1/5/2022
6	1/10/2021, 27/7/2023, 31/5/2024
7	16/12/2021
8	14/12/2023
9	14/12/2023, 9/5/2024
11	9/5/2024
12	10/5/2024
13	2/12/2022
14	2/12/2022

Tabla 5: Fechas de visitas de mantenimiento para cada estación.

5. Base de datos de BOCOSUR

Según explicamos en los apartados 4.2 y 4.3, la detección de un evento genera archivos de video y de metadatos. Estos son transferidos al servidor central respetando una estructura ordenada que separa los archivos por estación y fecha de creación. Optamos por usar la fecha en el calendario juliano y así creamos directorios con los primeros 6 dígitos del día juliano y subdirectorios con los otros dos. Por ejemplo, el video “*Station_3_2022-10-29-01-20-23.avi*” se guarda en la ruta *3/24598/81* ya que el día juliano del evento es *2459881.68082*. De esta manera, hemos creado una base de datos de la red BOCOSUR que se mantiene en constante crecimiento.

Actualmente, se encuentran almacenados en el servidor $\approx 292k$ videos generados por las cámaras WATEC (de las primeras 6 estaciones) y $\approx 204k$ ⁶ por las cámaras actuales ASI (de las 20 estaciones). De esos aproximadamente $\approx 204k$ archivos, la adición de los últimos aproximadamente $\approx 113k$ corresponde a una de las tareas realizadas por la autora durante las pasantías del proyecto BOCOSUR y este trabajo, específicamente desde abril de 2023. Dentro de estas tareas, la autora se encargó de la transferencia de dichos archivos a la base de datos. Puesto que las estaciones han estado operativas por tiempos distintos, hay un desbalance entre los datos disponibles. La distribución por estación de los videos ASI se representa en la figura 8.

Las cantidades de videos reportadas corresponden casi en su totalidad a videos sin analizar, es decir, desconocemos si son videos de meteoros u otros eventos. Al inicio de la red BOCOSUR, la clasificación de videos se realizaba mediante inspección visual. Sin embargo, con el despliegue de más estaciones y el crecimiento de la base de datos, esta tarea se volvió insostenible.

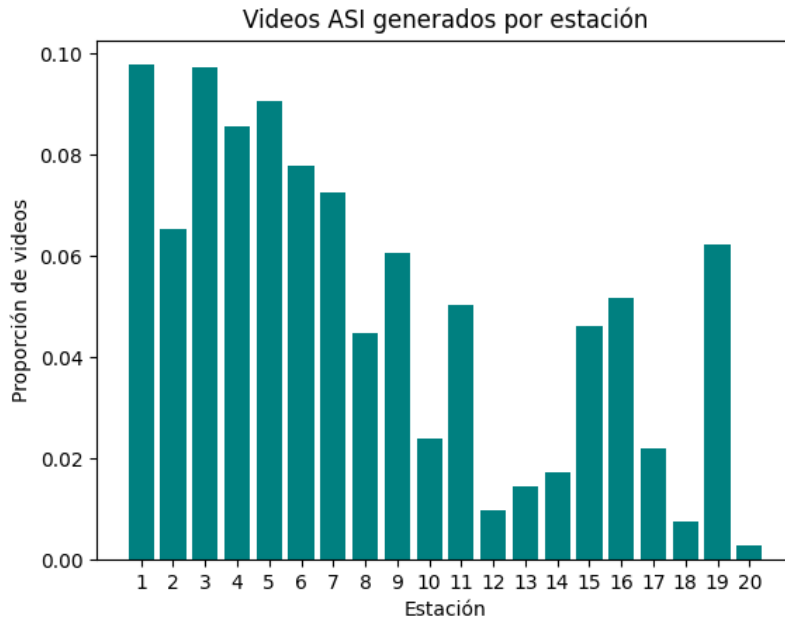


Figura 8: Distribución de videos ASI en el servidor local.

⁶Si de esos $\approx 204k$ videos eliminamos los de aquellas noches que superan los $5GB$ de peso, el número de videos ASI desciende a $\approx 118K$.

5.1. Clasificación “manual” de videos

En la clasificación “manual” de videos distinguimos entre varias categorías de eventos: meteoros, aviones, animales, focos y reflejos, nubes, lluvia, otros y nada. Cada categoría presenta características que las separa del resto, como la periodicidad de las luces de un avión o el movimiento aleatorio de un animal. En este contexto, la categoría “nada” corresponde a videos donde no se reconoce el origen que disparó la grabación. En la figura 9 mostramos algunos ejemplos de las categorías mencionadas.

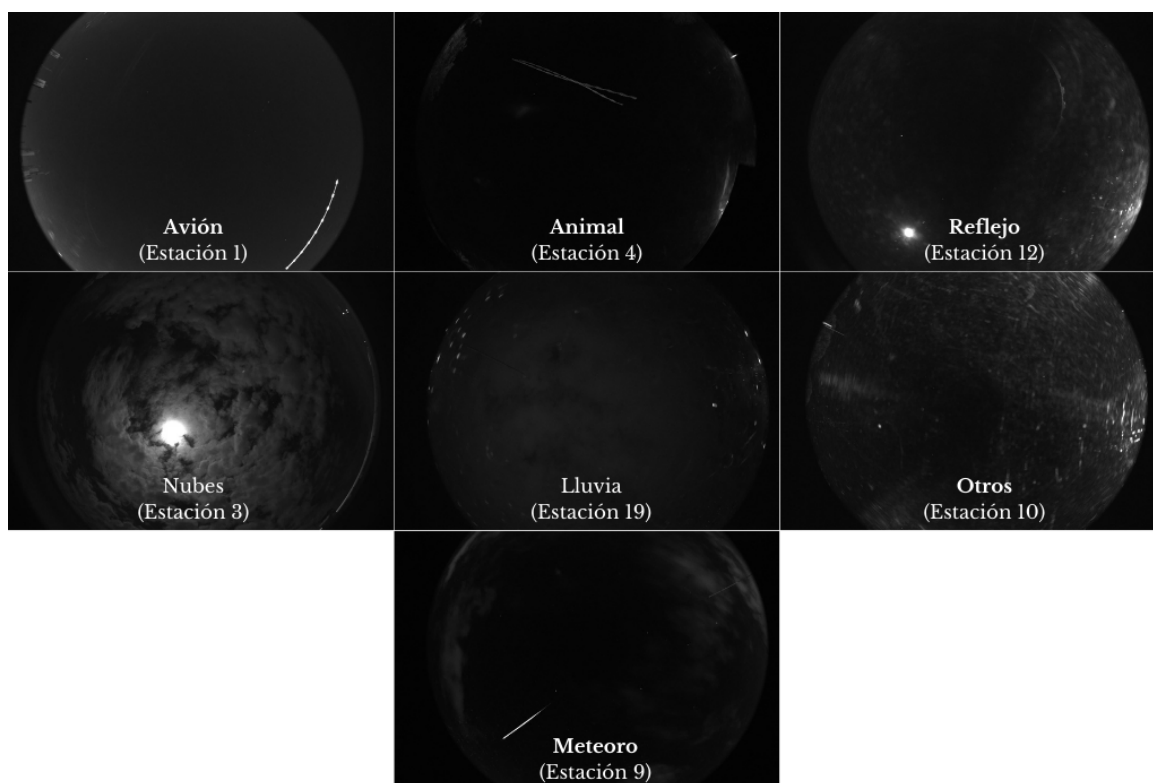


Figura 9: Ejemplos de las categorías de clasificación “manual”. Las imágenes fueron generadas mediante el apilamiento (*stacking*) de todos los fotogramas del video. Posteriormente, los valores de los píxeles son normalizados para ajustar los niveles de brillo. La categoría “otros” se representa con un ejemplo de telarañas sobre el domo de la estación 10, la categoría “animales” con el movimiento de dos aves en la estación 4 y en la categoría “meteoro” se muestra un bólido brillante detectado en la estación 9.

La tarea de clasificación se nutrió de la contribución de estudiantes y docentes de Secundaria, así como estudiantes de la Lic. en Física y la Lic. en Astronomía. Estas clasificaciones proceden de resultados los **Talleres BOCOSUR** ⁷. Estos talleres virtuales, en sus 4 ediciones, se desarrollaron con el fin de divulgar el proyecto BOCOSUR, presentar las nociones básicas de la física de meteoros y realizar actividades prácticas de clasificación de videos de la red.

Nos inspiramos también en esfuerzos de unir a la ciencia y la comunidad de un país en proyectos de Ciencia ciudadana, como es el caso de la red italiana ASTRALÉ (Delvecchio et al., 2021) ⁸. En este

⁷Por más información acerca de los contenidos y ediciones del taller visitar <http://bolidos.astronomia.edu.uy/taller-bocosur/>.

⁸<https://astrale.astronomiamo.it/>

proyecto involucran a usuarios voluntarios a monitorear el cielo en búsqueda de meteoros y controlar la calidad de aire a partir de datos generados por la red *all-sky*.

En la etapa de clasificación “manual” seleccionamos de manera aleatoria en fecha y estación $\approx 5k$ videos ASI. Estos fueron clasificados por los participantes de los Talleres BOCOSUR, siendo revisados posteriormente por esta autora para corroborar los resultados. En particular, de los videos mencionados, clasificamos como bólidos a 187 de ellos (despreciando así a meteoros poco brillantes).

Habiendo hallado 187 bólidos en $\approx 5k$ videos aleatorios, esperamos encontrar al menos $\approx 7.6k$ bólidos en la base de datos ASI.

La relación entre videos clasificados y bólidos hallados es evidencia de lo poco eficiente que es la clasificación “manual” en la etapa actual de la red. Con el fin de mejorar este rendimiento, los videos clasificados de esta forma sirven para crear una muestra de entrenamiento de un clasificador automático.

6. Clasificación basada en aprendizaje supervisado

Cuando hablamos de *machine learning* debemos distinguir entre aprendizaje supervisado y no supervisado. Ambas técnicas pueden ser empleadas para realizar tareas de clasificación pero lo hacen con enfoques diferentes. Amr (2020) explica que el aprendizaje supervisado se basa en la disponibilidad de un conjunto de datos etiquetados, donde cada entrada (o ejemplo) incluye tanto las características (*features*) como la etiqueta (*label*), es decir a qué clase pertenece. En contraposición, la clasificación o *clustering* en el contexto de aprendizaje no supervisado parte de datos no etiquetados. Un algoritmo de *clustering* busca agrupar muestras de manera que las pertenecientes al mismo grupo sean lo más similares entre sí, mientras que las de diferentes grupos sean lo más disímiles posible.

Amr (2020) compara el aprendizaje supervisado con la experiencia académica en la que al estudiante le presentan una serie de ejercicios de ejemplo para practicar, con las respuestas correctas al final de cada uno y así validar su desempeño mientras aprende. Finalmente, en el examen, es evaluado sin material para consultar.

En el aprendizaje supervisado aplicado a la clasificación es necesario conocer previamente qué características son útiles para discernir entre clases y tener ejemplos de las distintas clases para entrenar un modelo que pueda asignar etiquetas tras calcular los valores de las características de nuevos datos.

Uno de los algoritmos más simples y versátiles dentro del aprendizaje supervisado son los árboles de decisión, “es básicamente un set de condiciones” (Amr, 2020, p. 30). Un árbol de decisión clasifica instancias creando particiones recursivamente en el espacio de instancias. Si tenemos los datos en un formato de tabla o matriz, donde cada fila contiene las características de un objeto, a dicha fila le llamamos instancia.

Un árbol de decisión consiste en nodos organizados que emergen de una raíz, tal como un árbol con sus ramas, nodos y hojas. Los nodos internos dividen el espacio de instancias según los valores de las características (*features*), mientras que las hojas se asignan a una clase (o contienen vectores de probabilidad de pertenencia a una clase Maimon & Rokach (2014)).

Existen varias métricas de evaluación en el aprendizaje supervisado. Estas expresan relaciones entre las siguientes cantidades (Learners, 2024):

- **TP** (de *True Positive* o verdadero positivo) – Son los valores que el algoritmo clasifica como positivos y que realmente son positivos.
- **TN** (de *True Negative* o verdadero negativo) – Son valores que el algoritmo clasifica como negativos (0 en este caso) y que realmente son negativos.
- **FP** (de *False Positive* o falso positivo) – Son los valores que el algoritmo clasifica como positivo cuando realmente son negativos.
- **FN** (de *False Negative* o falso negativo) – Son los valores que el algoritmo clasifica como negativo cuando realmente son positivos.

Al analizar el desempeño de un modelo de aprendizaje supervisado se suele usar la representación llamada **matriz de confusión**:

$$\begin{pmatrix} \text{TN} & \text{FP} \\ \text{FN} & \text{TP} \end{pmatrix}$$

Dentro de las métricas más conocidas en clasificación binaria tenemos *accuracy*, *recall*, *precision* y *f-score*. A continuación se las define de acuerdo a ([Amr, 2020](#), pp. 175-176).

La métrica *accuracy* se define como el porcentaje de predicciones correctas realizadas por un modelo en comparación con el total de predicciones realizadas. Se calcula utilizando la siguiente fórmula:

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

La métrica de *recall*, también llamada tasa de verdaderos positivos, se usa para determinar cuántos de los valores positivos son clasificados correctamente.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (1)$$

De las muestras que fueron asignadas a la clase positiva, el porcentaje de ellas que realmente son positivas es la *precision* de dicha clase.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

Para resumir las dos métricas anteriores en un solo número, se puede utilizar *F1-score*. Muestra un equilibrio entre el peso que se le da a *recall* y *precision*.

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precisión} + \text{Recall}} \quad (3)$$

La métrica F-beta es una generalización de la métrica F1 que permite dar diferente peso a *precision* o *recall*. Se define como:

$$F_{\beta} = (1 + \beta^2) \cdot \frac{\text{Precisión} \cdot \text{Recall}}{(\beta^2 \cdot \text{Precisión}) + \text{Recall}} \quad (4)$$

El parámetro β determina el peso relativo de la *precision* y el *recall*. Valores de $\beta < 1$ hacen que la métrica F-beta dé más peso a la *precision*, mientras que valores de $\beta > 1$ hacen que se dé más peso al *recall*. Esto permite ajustar la métrica F-beta según las necesidades específicas del problema, ya sea que se prefiera minimizar los falsos positivos (dando más peso a la *precision*) o minimizar los falsos negativos (dando más peso al *recall*), como requiere el problema de clasificación de meteoros.

Evaluar un modelo de clases desbalanceadas en base a la *accuracy* no es recomendable ya que es la tasa de aciertos sin discriminar de qué clase. Las métricas mencionadas se suelen utilizar de forma conjunta, como se observa en la tabla 2.

6.1. *Machine learning* en Python

Hoy en día, existen numerosos recursos disponibles para aplicar técnicas de aprendizaje automático en Python. Una de las librerías más populares y accesibles es *scikit-learn* ([Amr, 2020](#)), el cual proporciona una amplia gama de algoritmos y herramientas para el aprendizaje supervisado y no

supervisado. Dentro de estos algoritmos, los árboles de decisión destacan por su capacidad para clasificación y regresión, siendo rápidos, fáciles de entender e interpretar. Además, se han desarrollado métodos avanzados como **Random Forest** dentro de esa librería, que combinan múltiples árboles de decisión para mejorar la precisión y robustez de los modelos.

Random Forest ([Scikit-learn, 2024](#)), desarrollado por Leo Breiman, es un método de ensamblaje que crea un “bosque” de árboles de decisión entrenados sobre diferentes subconjuntos de datos y características. Cada árbol en el bosque vota por una clase y la clase con más votos se convierte en la predicción del modelo. Este enfoque ayuda a reducir el sobreajuste y mejora la precisión del modelo en datos no vistos.

Otro algoritmo de aprendizaje automático ampliamente utilizado, con implementación en Python, es **XGBoost** (abreviatura de *Extreme Gradient Boosting*). Se basa en la construcción y combinación de múltiples árboles de decisión. Este es un algoritmo de potenciación de gradiente que construye árboles de decisión de manera secuencial. Cada nuevo árbol corrige los errores de los árboles anteriores, mejorando gradualmente la precisión del modelo. **XGBoost** también incorpora técnicas avanzadas como la regularización, que ayuda a prevenir el sobreajuste y mejora la generalización del modelo ([Chen & Guestrin, 2016](#)).

Las principales diferencias entre **Random Forest** y **XGBoost** radican en su construcción y optimización. Mientras que **Random Forest** construye árboles de manera independiente, **XGBoost** construye árboles de manera secuencial, optimizando los errores de los árboles anteriores. Además, **XGBoost** es conocido por su eficiencia y rendimiento debido a su capacidad de paralelización y optimización interna, lo que lo hace ideal para grandes conjuntos de datos.

7. Antecedentes: AUTOBOL

El actual algoritmo de detección de meteoros de la red BOCOSUR resulta en una gran cantidad de videos espurios. Durante la fase WATEC del proyecto fue desarrollado un clasificador automático de aprendizaje supervisado llamado AUTOBOL por Juan Pedro Ballestrino, Cecilia Deandraya y Cristian Uviedo ([Ballestrino et al., 2022](#)) en el marco de su trabajo de fin de carrera de Ingeniería eléctrica (Fing, UdelaR). AUTOBOL es un clasificador binario, es decir, distingue entre dos clases. La clase positiva son los “Bóolidos” y la clase negativa los “No Bóolidos”.

A rasgos generales, el clasificador AUTOBOL es un conjunto de códigos escritos en Python que se basa en árboles de decisión, más específicamente emplea la librería **XGBoost** (Extreme Gradient Boosting) de **scikit-learn** descrita en la sección anterior.

Los autores eligieron **XGBoost** puesto que “La idea general detrás de esto es que se pondrá énfasis en las instancias que son difíciles de predecir correctamente (casos “difíciles”) durante el aprendizaje, de modo que el modelo aprenda de los errores del pasado.” ([Ballestrino et al., 2022](#), p. 20).

La estructura del clasificador AUTOBOL se compone de 4 módulos de trabajo. Estos son listados, en el orden en que se suelen usar, de acuerdo a ([Ballestrino et al., 2022](#), p. 60-68).

- **Mask:** Se utiliza para crear la máscara de una estación, removiendo de los fotogramas de un video aquellos objetos que se encuentran en el horizonte, como edificaciones y árboles. Devuelve una imagen que se traduce en una matriz de ceros y unos.
- **Predict verbose:** El módulo tiene como objetivo obtener las características de los videos indicados en una ruta especificada. De esta manera, permite crear nuevos datos de entrenamiento. Crea un archivo “Clasificacion.csv” que es el *input* del siguiente módulo.
- **Train:** El módulo es utilizado para re-entrenar un modelo, para mejorar la eficiencia del algoritmo o para entrenar sobre otra base de datos. Genera un archivo donde se guarda el modelo “model.bin” y un archivo de texto con los parámetros del modelo.
- **Predict:** El módulo tiene como objetivo clasificar los videos que se encuentran en la ruta especificada entre las clases “Bóolido” y “No Bóolido”. Requiere de un modelo para predecir y devuelve un archivo “Predicciones.csv” con la clase asignada y probabilidad de pertenecer a la clase positiva de cada video del directorio seleccionado.
- **Predict localize:** A la vez de predecir, también devuelve una imagen con el evento identificado encerrado en un cuadrado. Es útil para reconocer qué se detecta en un video.

AUTOBOL está comprendido por cuatro códigos que apuntan a la modularidad y reusabilidad del mismo:

- **autobol.py:** *Script* principal que orquesta la ejecución del proyecto.
- **functions.py:** Contempla los distintos usos del clasificador, es decir, los distintos módulos de trabajo.
- **algoritmos.py:** Contiene todas las funciones referentes a la extracción de características de los videos y la creación de las máscaras.

- **abio.py**: Enfocado en las funciones de guardado de archivos.

Este algoritmo de aprendizaje supervisado extrae características del desplazamiento de un objeto luminoso y de su curva de luz. La determinación de estas características recae en el seguimiento o *tracking* del objeto a lo largo de los fotogramas del video. En particular, por cada fotograma o *frame* se selecciona un punto representativo del evento.

En resumen, se elige como candidato al punto (x,y) más luminoso del *frame* habiendo aplicado una resta de la imagen fondo (considerado como el percentil 30 de cada pixel del array de frames) y además restando la máscara que caracteriza a cada estación. Esta máscara se crea de forma interactiva, excluyendo árboles, focos o edificaciones en el horizonte propio de cada locación.

En cada *frame* calculan el *centro de masas* donde el rol de la masa lo tiene el brillo del objeto, de esta manera distinguen entre el centro geométrico del evento (*centroide*) y el centro ponderado por el brillo. Esos puntos candidatos sufren varios filtros para ser considerados pertenecientes a la trayectoria del evento, es decir, *inliers*.

Dentro de los filtros, resaltamos el uso del algoritmo RANSAC (*Random Sample Consensus*), que se utiliza para encontrar el modelo de circunferencia que mejor se ajuste a un conjunto de puntos dados. La distorsión de las lentes ojo de pez hace que los objetos rectilíneos se representen como curvas. De esta forma, la traza de un meteoro (casi rectilínea) se observa como un arco de circunferencia (ejemplificación en la figura 5.8 de (Ballestrino et al., 2022, p. 34)).

El algoritmo RANSAC funciona seleccionando aleatoriamente subconjuntos de datos (tres puntos en esta aplicación) y ajustando un modelo a estos puntos. Luego, evalúa la calidad del modelo midiendo la pérdida (**loss** de la tabla 6). Se define **loss** como la suma de las diferencias entre los datos y el modelo ajustado. Los puntos que tienen una pérdida menor que un umbral específico (determinado por la mediana de la desviación absoluta multiplicada por un parámetro residual) se consideran *inliers*, mientras que los puntos con una mayor pérdida se descartan como *outliers*. Se itera este proceso múltiples veces, eligiendo finalmente el modelo con la menor pérdida acumulada y el mayor número de *inliers*.

Respecto a la determinación de la curva de luz del objeto, se contempla la reconstrucción de brillo en regiones saturadas (superan la intensidad de 255 ADUs⁹ por píxel). Esto es particularmente útil en grabaciones de meteoros brillantes o bólidos. Luego de determinar los *inliers* de la trayectoria con el algoritmo RANSAC, se reconstruye el brillo en los píxeles saturados (Ballestrino et al., 2022, p. 31).

Un paso muy importante a la hora de crear un clasificador de este estilo es elegir las características (*features*) que participan de las decisiones para distinguir entre las clases. El clasificador AUTOBOL extrae 19 características por video. La descripción de cada una de ellas se encuentra en la tabla 6.

El algoritmo de machine learning cuantifica la importancia de cada una de las características calculadas (ver figura 10). A partir de esto, descartan el uso de *var_vel*, *var_fwhm* y *mean_fwhm* como información para reconocer “Bólidos” y “No Bólidos”. Por otro lado, obtienen que la característica o *feature* más importante fue la intensidad mínima del evento o *min_int* (ver figura 6.2 de (Ballestrino et al., 2022, p. 45)).

⁹De la sigla en inglés *Analog to Digital Units*.

Feature	Descripción
Video	Nombre que identifica el video que se procesó
N	Cantidad de puntos del evento
dispersion	Medida de dispersión de los puntos.
<i>max_vel</i>	Velocidad instantánea máxima del evento.
<i>min_vel</i>	Velocidad instantánea mínima del evento.
<i>velocidad_media</i>	Velocidad instantánea media del evento.
<i>var_vel</i>	Varianza de la velocidad instantánea.
<i>mean_intensidad</i>	Intensidad media del evento.
<i>max_int</i>	Intensidad máxima del evento.
<i>min_int</i>	Intensidad mínima del evento.
<i>var_int</i>	Varianza de la intensidad.
<i>mean_fwhm</i>	FWHM ¹⁰ media del evento.
<i>var_fwhm</i>	Varianza del FWHM.
<i>sensibilidad_i1</i>	Sensibilidad de la intensidad normalizada del evento.
<i>sensibilidad_i2</i>	Sensibilidad de la intensidad sin normalizar.
<i>sensibilidad_v1</i>	Sensibilidad de la velocidad normalizada del evento.
<i>sensibilidad_v2</i>	Sensibilidad de la velocidad sin normalizar.
<i>sensibilidad_fwhm1</i>	Sensibilidad del FWHM normalizada del evento.
<i>sensibilidad_fwhm2</i>	Sensibilidad del FWHM sin normalizar.
recorrido	Desplazamiento del evento.
loss	Indica qué tan bien se ajusta el evento al modelo de un arco.

Tabla 6: Descripción de las características extraídas de un evento. Fuente: tabla 5.2 (Ballestrino et al., 2022, p. 36).

AUTOBOL fue entrenado y validado con videos de las estaciones 1, 2 y 3 durante la fase WATEC del proyecto BOCOSUR. La clasificación binaria es particularmente compleja cuando existe un desequilibrio entre las clases, o sea, cuando las clases dentro del conjunto de datos no están representadas de manera equitativa. Esto ocurre en los videos de la base de datos de BOCOSUR. En el trabajo de Ballestrino et al. (2022) entrenaron el modelo con 388 videos, de los cuales 27 son “Bóolidos” y 361 “No Bóolidos”.

AUTOBOL, contemplando el desequilibrio de clases, utiliza métricas que permiten maximizar el *recall* como el F2-score y realizan una búsqueda de hiper-parámetros ¹¹ de **XGBoost** (ver tabla 6.1

¹¹Los hiperparámetros de un algoritmo son parámetros que configuran el comportamiento del modelo antes de iniciar el proceso de aprendizaje. A diferencia de los parámetros del modelo, que se aprenden durante el entrenamiento, los hiperparámetros se establecen previamente y determinan aspectos como la estructura del modelo, la tasa de aprendizaje, el número de árboles, y la profundidad de los árboles, etc. La búsqueda de los hiperparámetros adecuados para el

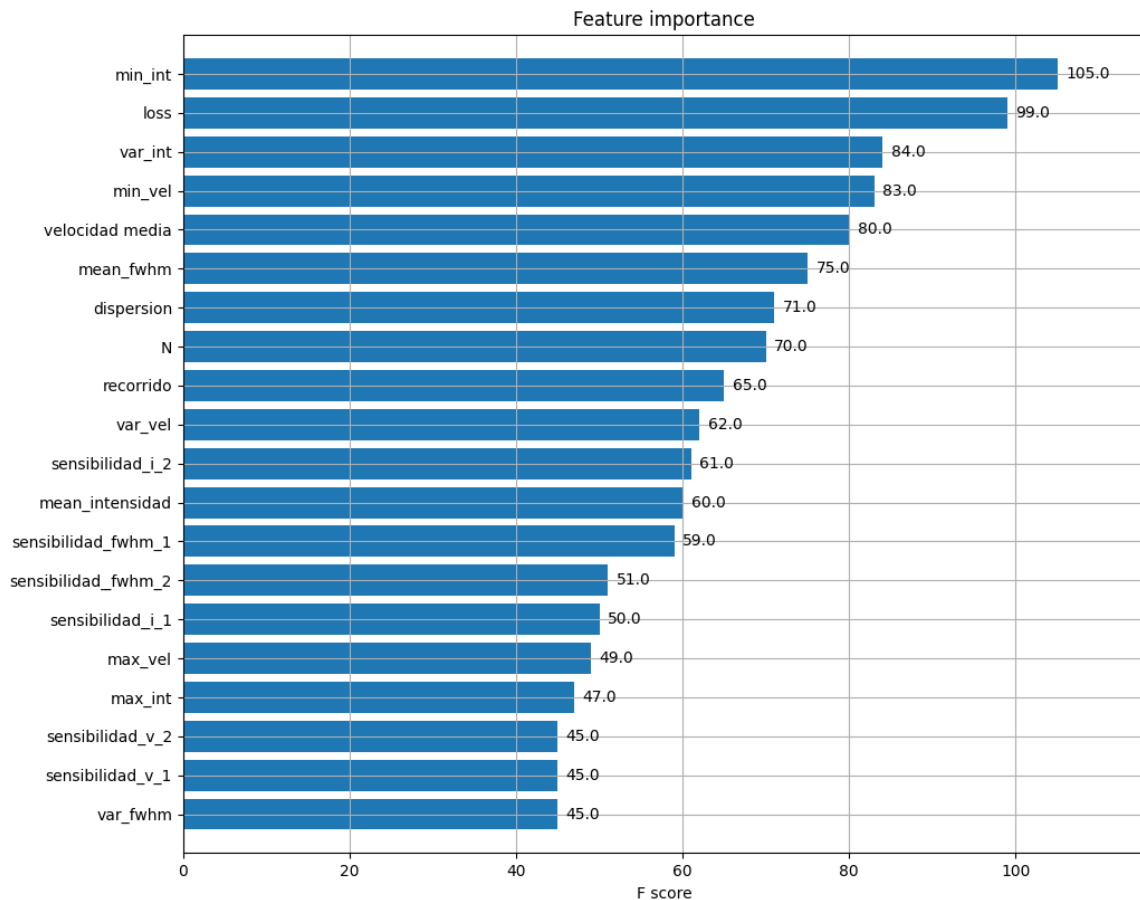


Figura 10: Importancia de las características de la librería XGBoost en el modelo final de AUTOBOL. Fuente: figura 6.2 de (Ballestrino et al., 2022, p. 45).

(Ballestrino et al., 2022, p. 43)). Su modelo final, evaluado con las métricas descritas en la sección 6, tiene los siguientes valores:

Métrica	Valor
Recall	92.6 %
Precision	83.3 %
F2-Score	90.5 %

Tabla 7: Modelo generado por AUTOBOL con 27 bólicos y 361 no bólicos. Fuente: tabla 6.2 de (Ballestrino et al., 2022, p. 44).

Los resultados de la tabla 7 sugieren cómo se debe comportar el modelo, la verdadera prueba consiste en aplicarlo sobre datos que no hayan participado del proceso de entrenamiento. Realizan una evaluación en campo donde poseen 17 bólicos y 210 no bólicos a los que le aplican el modelo para predecir su etiqueta (*label*). La matriz de confusión en su evaluación de campo (extraída de la figura

problema es crucial para mejorar el rendimiento del modelo, ya que influyen en su capacidad para generalizar a nuevos datos y evitar problemas como el sobreajuste (*overfitting*) o el subajuste (*underfitting*).

6.6 de (Ballestrino et al., 2022, p. 48)) fue:

$$\begin{pmatrix} 208 & 2 \\ 1 & 16 \end{pmatrix}$$

Tales resultados son muy prometedores en lo que refiere a los eventos de las cámaras WATEC. Al referirse a la incorporación de las nuevas cámaras (ASI) sostienen “Si bien el modelo actual debería clasificar los videos generados sería recomendable relevar una cantidad significativa de datos -que incluyan las estaciones nuevas- y determinar un nuevo modelo de clasificación, que se puede realizar con las herramientas que ofrece el software actualmente” (Ballestrino et al., 2022, p. 53).

8. Clasificador automático para videos ASI

8.1. AUTOBOL en la etapa ASI

En un primer intento de abordar el problema de clasificación de videos ASI buscamos crear un nuevo modelo realizando menores modificaciones a los códigos AUTOBOL.

Los videos tomados por las cámaras ASI presentan algunas diferencias frente a las cámaras WATEC, como el *fps* y la resolución (tabla 4). El tamaño de los fotogramas configurado en la aplicación de detección es $(640 \times 480)px$ para los videos de la cámara WATEC y $(1548 \times 1040)px$ en las cámara ASI.

Encontramos que estos valores estaban fijos en el código por lo que lo modificamos con el fin de que fuesen parámetros a ser extraídos directamente de los *frames* de los videos a analizar.

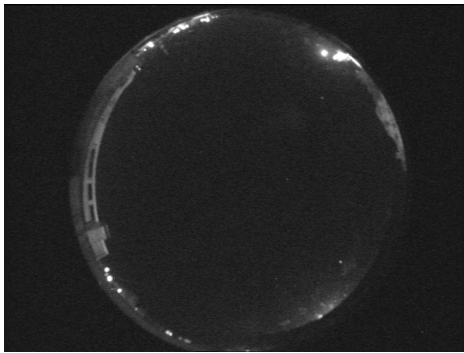


Figura 11: *Frame* individual de un video tomado con la cámara WATEC en la estación 3.



Figura 12: *Frame* individual de un video tomado con la cámara ASI en la estación 3.

Comparación de los *frames* de dos videos de la estación 3, con las cámaras WATEC y ASI.

En los videos de las cámaras ASI, solo una parte del cielo es capturado y observamos la circunferencia del horizonte cortada. Por el contrario, con las cámaras WATEC se puede distinguir el horizonte por completo, incluyendo un margen superior e inferior. La principal consecuencia de este cambio es que, al seguir un evento en un video con la función de *tracking* original de AUTOBOL, es fácil exceder los límites de la imagen al considerar una ventana centrada en un objeto cercano al horizonte. Para solucionar este problema, aplicamos una técnica conocida como **clamping**, que restringe las coordenadas de la ventana dentro de los límites de la imagen.

Para el entrenamiento de un modelo de videos con cámaras ASI consideramos 136 “Bólidos” (que eran un conjunto de meteoros y bólidos propiamente dichos) y 1336 “No Bólidos”. Estas muestras incluyen ejemplos de las 20 estaciones de la red BOCOSUR y una gran variedad de eventos de eventos de ambas clases. Se representaron dentro de los “No Bólidos” aviones, animales diversos, reflejos de focos en el domo, nubes, gotas de lluvia, el “titilar” de las estrellas, etc. Dentro de la muestra de “Bólidos” elegimos videos de meteoros, bólidos cortos y largos y ejemplos con presencia de fragmentación. Tomamos como punto de partida la proporción de bólidos y no bólidos de la muestra de entrenamiento de los creadores de AUTOBOL: aproximadamente 1 bólido cada 14 no bólidos. Sin embargo, afirman que “La causa principal de la alta tasa de falsos negativos es consecuencia del gran desbalance del

dataset” (Ballestrino et al., 2022, p. 45) por lo que buscamos manejar este desbalance con un número más elevado de meteoros y bólidos.

El primer indicio de que algo no estaba funcionando correctamente fue la variabilidad de los valores de las métricas establecidas en la sección 6 al cambiar levemente el porcentaje de la muestra se emplearía en entrenamiento y testeo del modelo.

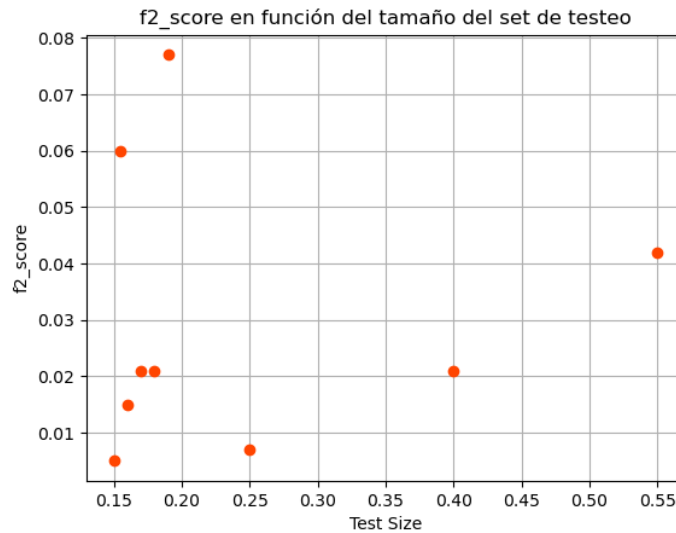


Figura 13: Variabilidad del $f2$ -score en función del tamaño del set de testeo.

Inicialmente sospechamos que este error yacía sobre lo desbalanceado de las clases e incluso que estábamos sobre-exigiendo al clasificador con una muestra de la clase positiva que contenía gran diversidad de meteoros. Asimismo, sopesamos si debíamos establecer un modelo por estación. Recurrimos a depurar la muestra de “Bólidos”, conteniendo únicamente bólidos (brillantes, largos, algunos presentaban fragmentación). Esos distintos modelos demostraron un desempeño deficiente en las predicciones. En consecuencia, consultamos con los desarrolladores de AUTOBOL y sugirieron que revisáramos la validez de la función *tracking* en el contexto de los videos ASI. Presentamos un ejemplo de lo que se detecta en los videos de bólidos en las figuras 14 y 15.

La función *tracking* presenta muchos problemas que se ejemplifican en las figuras recién mencionadas. En ciertos *frames* la función seleccionó estrellas como puntos pertenecientes al evento a pesar de que constituyen puntos muy alejados entre sí y en ninguno de los filtros existentes fueron considerados *outliers*. En los videos de las cámaras ASI, al poseer una sensibilidad mayor, es posible identificar estrellas en *frames* individuales.

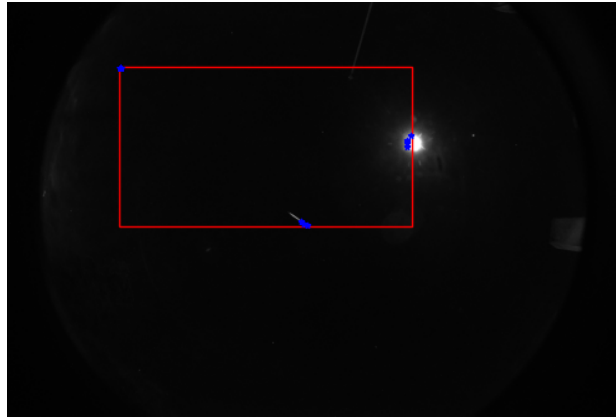


Figura 14: En este ejemplo de un bólido de la estación 7 podemos analizar dos obstáculos en la determinación de la trayectoria en los *frames*: la luna y las estrellas.



Figura 15: Este es un ejemplo todavía más evidente del mal funcionamiento del *tracking* de AUTOBOL en las cámaras ASI, vemos la traza del bólido en la región inferior de la imagen. Sin embargo, ninguno de sus puntos es registrado. Las estrellas son lo más brillante en cada *frame* enmascarado.

En azul se muestran los puntos registrados de la trayectoria del bólido y el rectángulo rojo presenta el tamaño del evento.

8.2. Modificaciones a AUTOBOL

La necesidad de reescribir la función de seguimiento de un evento nos llevó a la siguiente etapa del proyecto. Buscamos crear un clasificador simple pero robusto que se adapte a las características del nuevo sistema de adquisición. Como punto de partida, modificamos la función *tracking* ¹².

El esquema general de la nueva función de seguimiento se encuentra representado en la figura 16. Explicaremos a continuación cada uno de los bloques allí mostrados.

¹²En el apéndice B de funciones implementadas de (Ballestrino et al., 2022, p. 78) presentan un diagrama de flujo que condensa las ideas detrás de la función *tracking* original.

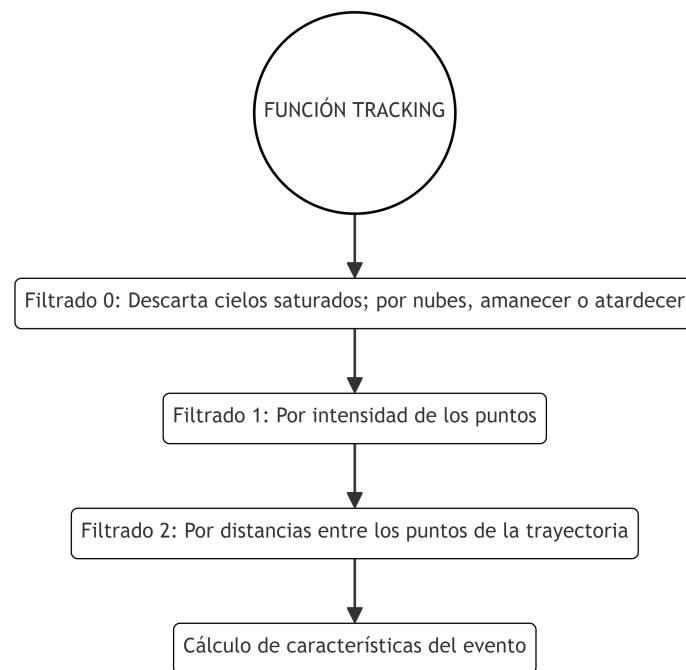


Figura 16: Esquema general de la función *tracking* final.

Agregamos un proceso de filtrado inicial o “filtrado 0”, que se muestra en la figura 16. Durante un análisis de los distintos eventos detectados, identificamos una categoría de videos donde se observa un animal oscuro (generalmente un insecto) con un cielo saturado en el fondo de las imágenes que lo componen. La marca temporal o *time stamp* de estos videos señalaba que eran eventos correspondientes al amanecer o atardecer y en casos aislados, donde se efectuó un desajuste de configuración del equipo como ganancia y exposición de la cámara. Evaluamos distintos casos y calculamos la mediana del fondo de un *frame* del video y observamos que superaban el valor 200, umbral que consideramos para descartar tales videos de un análisis posterior.

Luego, reescribimos el primer bucle de la función de seguimiento (previo a la aplicación de RANSAC y filtros posteriores). La primera prueba consistió en dos cambios fundamentales: trabajar con la diferencia de dos *frames* consecutivos (con la máscara aplicada) y considerar un umbral de distancia para despreciar un punto si se encuentra muy alejado del anterior. La idea detrás de trabajar con la diferencia de los *frames* es, inspirándonos en el algoritmo de detección **boldosGUI**, reconocer los objetos luminosos en movimiento.

La figura 17 muestra el algoritmo preliminar que implementamos, que filtra por intensidad y distancia. Testeamos este algoritmo sobre varios bólidos de ejemplo y notamos que fallaba en eventos donde en los primeros *frames* aún no se observa el bólido ¹³, por lo que al seleccionar el máximo de la imagen diferencia en las primeras iteraciones, se guarda la posición generalmente de una estrella. Entonces, en el siguiente *frame* donde sí “aparece” el bólido, se obtiene una distancia muy grande respecto al punto anterior, superando el umbral de distancia definido y de esa manera todos los puntos de interés son descartados.

¹³Esto es una característica del algoritmo de detección **boldosGUI**, donde los videos generados presentan segundos *extra* antes y después de la grabación del evento detectado.

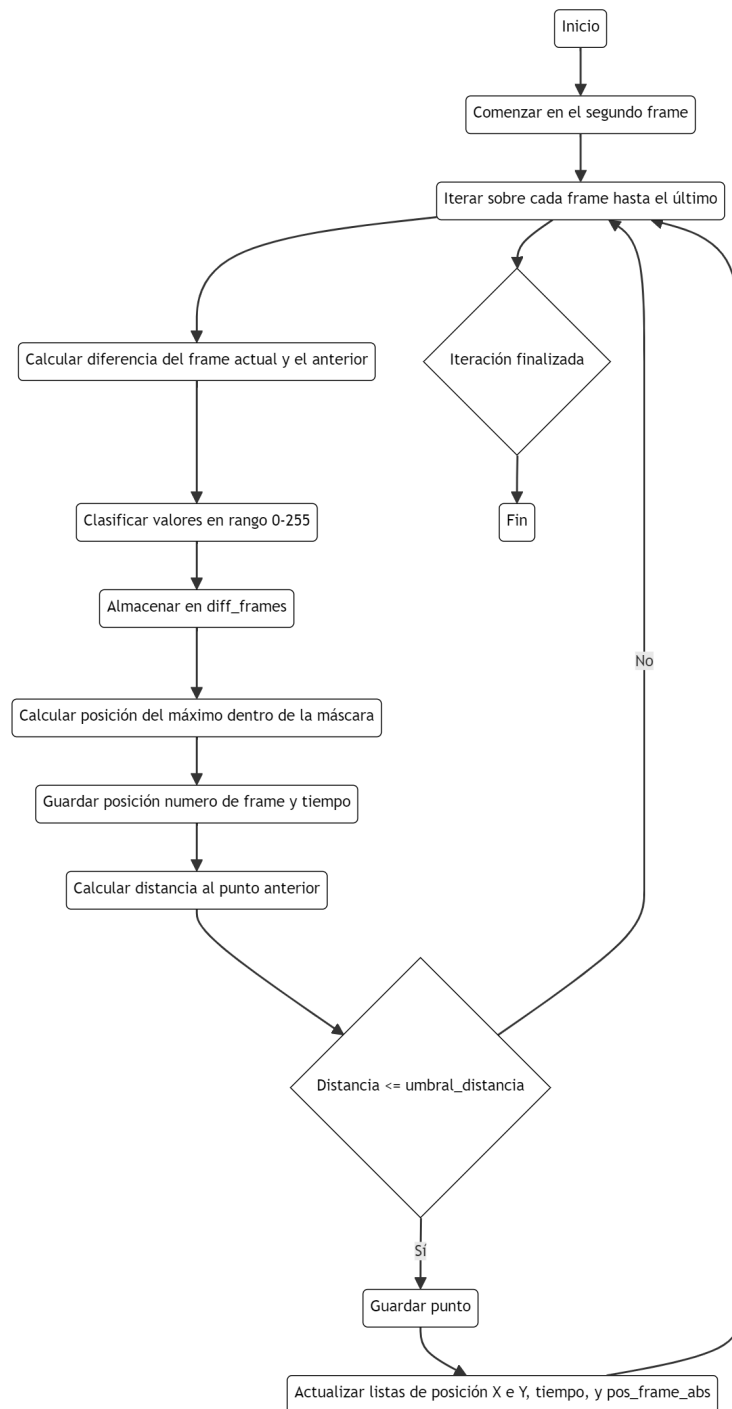


Figura 17: Diagrama de flujo del primer intento de modificación al bucle inicial de la función *tracking*, representando una versión preliminar del código.

Gracias a las pruebas del algoritmo realizadas, fue necesario volver a estructurar el seguimiento de los objetos. Abordando el reemplazo del fragmento de *tracking* anterior (sustitución del bloque representado en 17), nos enfocamos en determinar la presencia de trazas. A la primera parte de este proceso la llamamos “Filtrado 1”. Buscamos un punto representativo de cada *frame* pero en vez de tomar el máximo de imagen diferencia enmascarada, trataremos con todos los puntos de esa imagen

que superen un umbral de brillo dinámico. Los parámetros importantes de esta búsqueda son:

- **thr**: Umbral o *threshold* de brillo dinámico. Se define como el 95 % del máximo de la imagen.
- **knn**: El cuadrado o ventana centrada en el punto de interés (aquel que supera el umbral **thr**) tiene lado $2 * knn$.
- **nC**: Fracción de puntos en la ventana que superan el umbral **thr** respecto al total de puntos o píxeles de la ventana.

Ahora, para que guardemos cierto punto, observamos sus puntos vecinos y evaluamos si hay otros puntos brillantes en esa ventana. Esto nos permite rechazar puntos brillantes aislados y diferenciarlos del caso de una traza (que tiene cierto “grosor”). Los valores de **knn** y **nC** iniciales fueron determinados manualmente mediante pruebas iterativas. Entendemos que son parámetros extremadamente sensibles a las características del video y del sistema de adquisición que, en el caso de un cambio en el futuro de la red BOCOSUR, deberían ser modificados. Estos parámetros buscan un equilibrio entre la sensibilidad y la redundancia en la determinación del evento en la imagen. De haber considerado una ventana muy pequeña sería muy sencillo que los puntos aledaños superen el umbral de brillo o de ser muy grande ingresaría ruido adicional al análisis. Por otro lado, la elección de **nC** inicial o **nC00** (como lo nombramos en el código) resulta de estudiar el brillo de meteoros y bólidos en los primeros frames de la detección, ya que pretendemos extender la determinación de la traza lo más posible, desde el comienzo al final del fenómeno luminoso. Los valores finales fueron $knn = 2$ píxeles y $nC = 0.07$.

El diagrama de flujo correspondiente a la versión final del “Filtrado 1” se encuentra en la figura 18. En el diagrama se representa una comparación del *frame* “actual” con el *frame* situado tres posiciones anteriores. Este cambio fue incorporado al bucle para detectar bólidos que exhiben variaciones de brillo graduales, especialmente cuando se observan de frente. De esta manera, se enfatizan las fluctuaciones de brillo.

Este enfoque es particularmente adecuado para la detección de trazas en videos debido a su capacidad para:

- Enfocarse en pequeñas regiones de alta intensidad.
- Adaptar el umbral de detección en función de la intensidad relativa.
- Resaltar cambios rápidos y significativos entre frames.
- Utilizar un proceso iterativo para mejorar la precisión de detección.

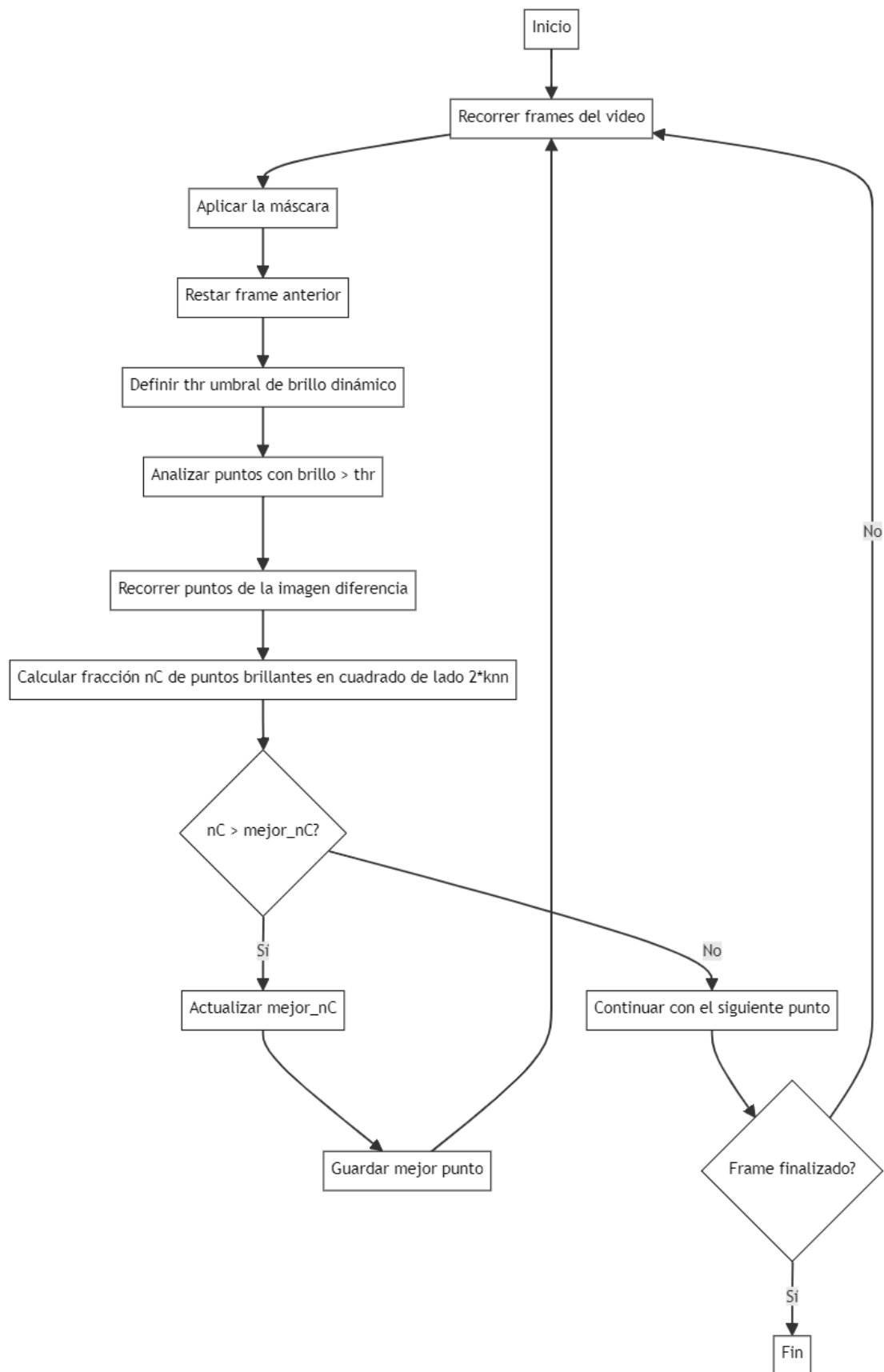


Figura 18: Diagrama de flujo del filtrado por intensidad en la función *tracking* final.

A pesar de que ofrece cierto refinamiento en comparación al razonamiento anterior, un caso común en el que falla este algoritmo es cuando se presenta un video en el que una estrella brillante titila antes de que se observe la traza del meteoro. Se obtienen las coordenadas intercaladas de la estrella y del meteoro. Para resolver este problema agregamos lo que nombramos “Filtrado 2” en el diagrama de la figura 16 y se puede entender en más profundidad en el diagrama de flujo de la figura 19.

Este segundo filtrado se realiza en casos donde existen más de tres puntos registrados en la trayectoria (saliente de “Filtrado 1”). Este bloque de código realiza un filtrado de posiciones detectadas en un video para identificar y eliminar *outliers* mediante un análisis de distancias. Se recorren todas las posiciones para calcular las distancias entre cada posición y las restantes, y se calcula el histograma de estas distancias. Se aplica un criterio de filtrado: si el conteo del *bin* más pequeño del histograma es mayor o igual que el del *bin* más grande, la posición se marca como válida (y guardamos como posición *inlier*).

Remarcamos que este procedimiento nos brinda las posiciones (x,y) del evento, interpretando cada imagen con una representación matricial ¹⁴, el número índice del *frame* donde se guardó un punto y conociendo el valor del *fps* o cuadros por segundo, también tenemos información temporal del evento.

En el código original de la función *tracking* de AUTOBOL existen procesos de filtrado adicionales que optamos por eliminar del nuevo clasificador. Uno de ellos es la aplicación del ajuste de circunferencia basado en RANSAC. Al comprobar la validez de las nuevas modificaciones a la función *tracking*, notamos que en una gran cantidad de eventos obteníamos un vector de posiciones saliente de “Filtrado 2” con más de tres puntos pero al pasar por la función RANSAC este valor decrecía hasta tres.

Como vimos en la sección 7, el algoritmo selecciona aleatoriamente tres puntos para ajustar un modelo de circunferencia. El problema surge cuando todos los puntos menos esos tres son considerados *outliers* debido a varios factores inherentes a su implementación. Primero, el algoritmo selecciona aleatoriamente tres puntos para ajustar un modelo de circunferencia; si estos tres puntos están lo suficientemente cerca o en línea recta, el modelo será muy específico y considerará a todos los demás puntos como *outliers*.

Además, el criterio de *inliers* se basa en el residuo absoluto entre la distancia de cada punto al centro de la circunferencia ajustada y el radio de dicha circunferencia, lo que puede excluir muchos puntos si la distribución es amplia o no circular. El uso de la Mediana de la Desviación Absoluta (MAD) para calcular la desviación de los residuos puede resultar en la clasificación de muchos puntos como *outliers* si los tres puntos iniciales no representan bien la distribución general.

Finalmente, el estricto criterio para la actualización del modelo, que se basa en una menor suma de pérdidas, puede mantener modelos que solo se ajustan bien a los tres puntos seleccionados, excluyendo la mayoría de los demás puntos.

¹⁴El punto (0,0) se encuentra en el vértice superior izquierdo.

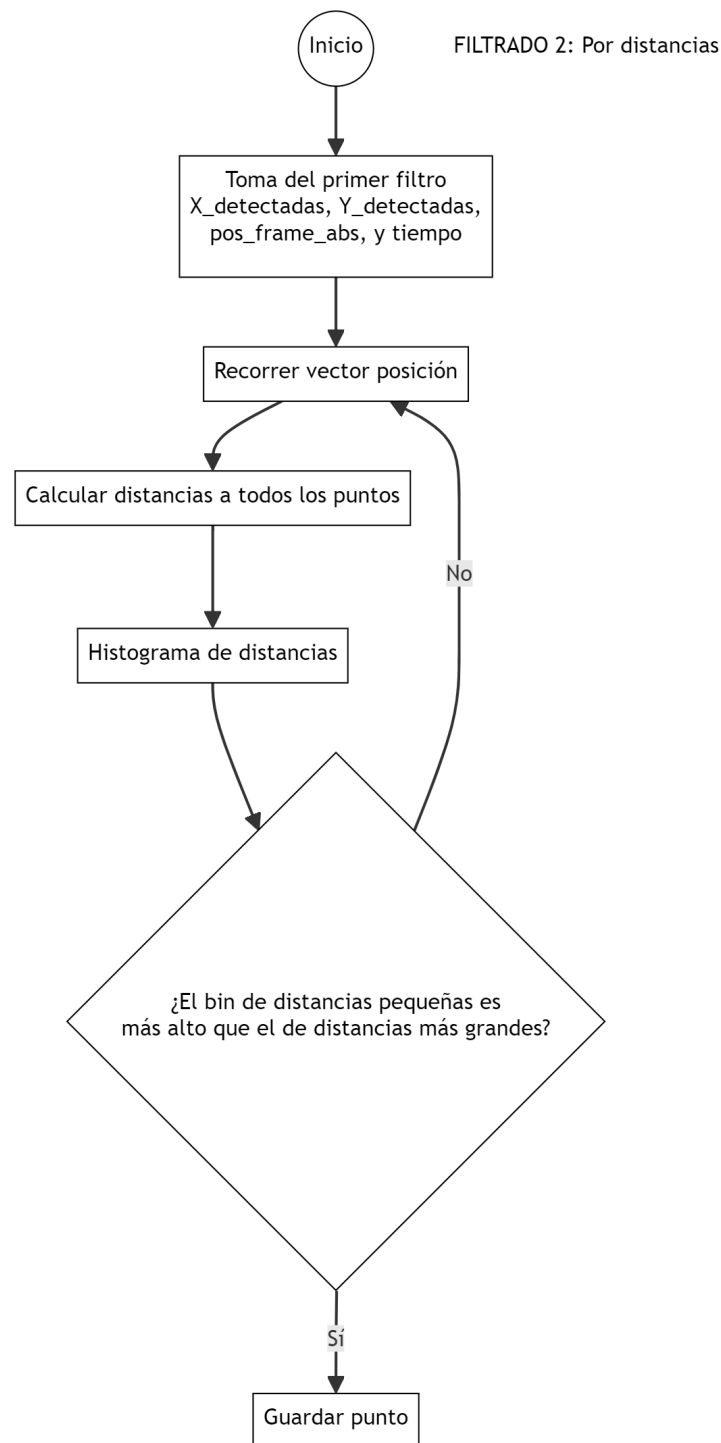


Figura 19: Diagrama de flujo del filtrado por distancias entre los puntos de la trayectoria en la función *tracking*.

Sin dudas, conocer cómo se ajusta la trayectoria de un evento a un arco de circunferencia es información relevante para distinguir entre bólidos y otros eventos. Por lo tanto, realizamos el ajuste guardando como datos el radio de la circunferencia ajustada y cuánto se desvía la trayectoria a ese ajuste pero, sin descartar ningún punto en el proceso. El ajuste es ahora realizado con la librería

Circle-Fit (PyPi, 2024) y en particular con **taubinSVD** que fue implementado en Python en base a los códigos de MATLAB de Nikolai Chernov (Ashkenazi, 2024). Es rápido y eficaz. Hemos notado complicaciones únicamente en casos donde la trayectoria es extremadamente corta y rectilínea, por lo que radio tiende a infinito y devuelve un valor **inf** en Python. El código actual maneja esas excepciones para evitar errores.

Estos cambios impactaron la eficacia de la función *tracking* para nuestro nuevo clasificador. Algunos escenarios desafiantes en los que probamos su desempeño son videos de bólidos en un cielo con Luna, bólidos que no son visibles en los primeros *frames*, bólidos con una trayectoria larga, bólidos en cielo nublado, entre otros.

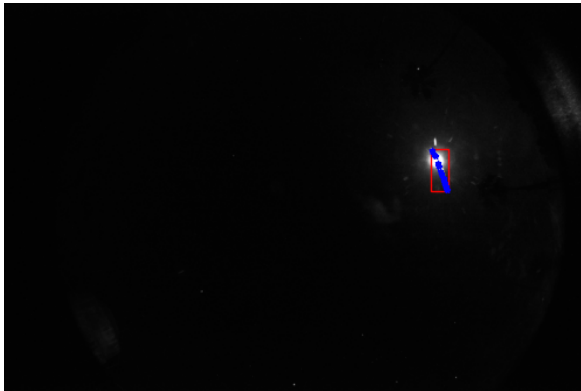


Figura 20: Bólido que pasa frente a la Luna.

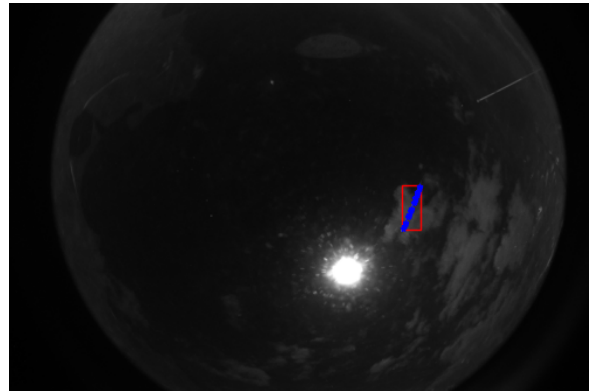


Figura 21: Bólido en un cielo nublado.

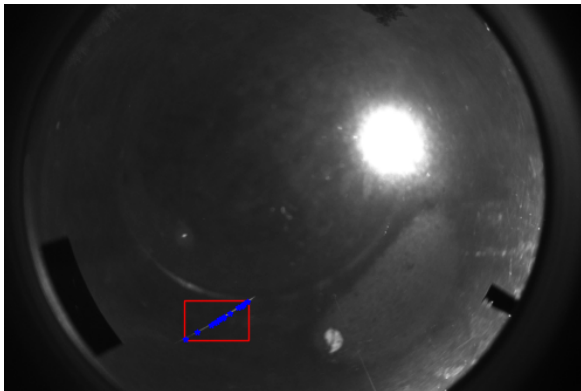


Figura 22: Bólido detectado en Luna llena.

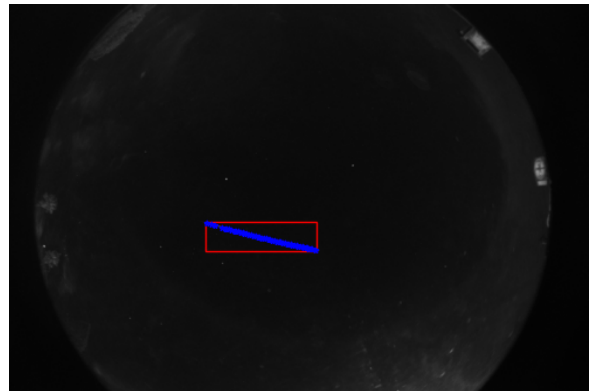


Figura 23: Bólido con una larga trayectoria.

Ejemplos de determinación de trayectoria con la nueva función *tracking* implementada.

Habiendo aplicado las modificaciones mencionadas a la función de seguimiento obtenemos una significativa mejora. Observamos en la figura 20 la trayectoria del bólido que cruza frente a la Luna, el cual no pudimos detectar con implementaciones anteriores (figura 14). La función fue testeada en 35 videos de bólidos y en todos ellos fue exitosa.

En el proceso de clasificación de eventos en los videos, utilizamos la función *predict_verbose* para extraer características significativas de los videos.

Una condición para que un video sea considerado en el modelo de clasificación es que se registren al menos tres puntos en la trayectoria del evento. Este enfoque garantiza que solo los eventos con datos suficientes y consistentes sean utilizados para entrenar y evaluar el modelo, mejorando así la precisión y la robustez de las predicciones. Además, al filtrar eventos no representativos, optimizamos el rendimiento del modelo y evitamos sesgos que podrían derivarse de datos insuficientes o inconsistentes.

8.3. Características o *features* del nuevo clasificador

Mostramos en la tabla 6 las *features* que son extraídas de los eventos en el código original AUTO-BOL. Evaluamos su relevancia en la clasificación de videos ASI. La información se extrae a partir de la posición en función del tiempo del evento y su curva de luz. La función “CurvaDeLuz” (Ballestrino et al., 2022, p. 89) calcula la intensidad del brillo de un objeto en un fotograma específico. Define una ventana en torno a las coordenadas representativas del evento en ese *frame*. Luego, suma los valores de intensidad dentro de esta región en el fotograma actual y en el fondo, devolviendo la resta de dichos valores. Los siguientes ejemplos de “Bólidos” y “No Bólidos” mostraron que pueden ser muy similares sus curvas de luz.

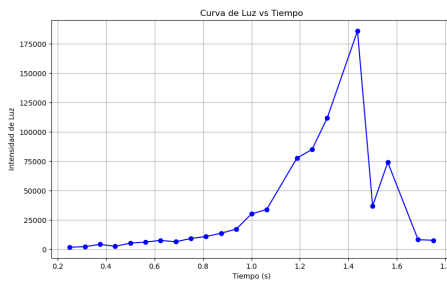


Figura 24: Curva de luz de un bólido que presenta fragmentación registrado por la estación 11.

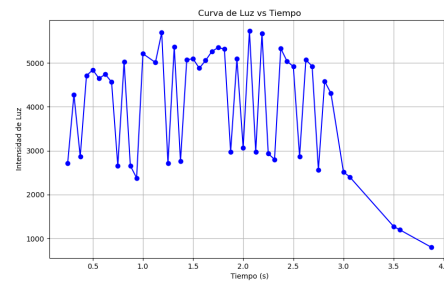


Figura 25: Curva de luz de un bólido de trayectoria larga sin fragmentación, registrado por la estación 15.

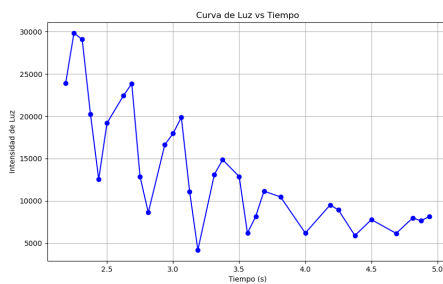


Figura 26: Curva de luz de una araña en movimiento sobre el domo de la estación 11 delante de un cielo completamente cubierto.

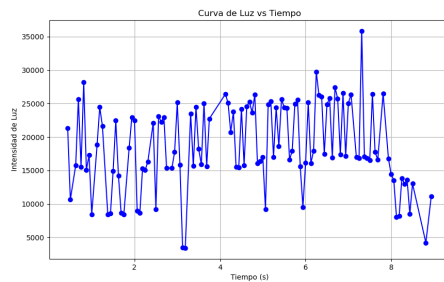


Figura 27: Avión observado en la estación 1.

Ejemplos de curvas de luz obtenidas.

Cuando se considera la curva de luz de un bólido, la representación que probablemente viene a la mente es la mostrada en la figura 24, donde la intensidad del brillo aumenta hasta un máximo,

pudiendo presentar picos debido a la fragmentación del meteoróide. Sin embargo, las curvas de luz de meteoros y bólidos detectados despliegan un abanico de posibles formas. La periodicidad que esperaríamos únicamente en la curva de luz de un avión (ver figura 27) comparte similitudes con la curva de luz de la figura 25 (bólide). Tenemos también que, incluso las curvas de luz de algunos animales, poseen similitudes con las anteriores (ver figura 26).

De las características extraídas originalmente, encontramos que aquellas referidas a la “suavidad” o “sensibilidad” no aportaban a la distinción de eventos. Dicho cálculo no es ideal para medir la suavidad de una curva porque simplemente suma las distancias entre puntos consecutivos (Ballestrino et al., 2022, p. 90). Por razones análogas descartamos el cálculo original de la característica “dispersión”.

Para añadir información de la forma de estas curvas de luz, realizamos un ajuste de segundo orden $y = ax^2 + bx + c$ y guardamos para cada evento a y b .

Las características o *features* que extraemos de cada video ahora son aquellas listadas en la tabla 8. Mantuvimos el cálculo de velocidad original de AUTOBOL.

Feature	Descripción
velocidad media	Velocidad media en la trayectoria
mean_intensidad	Intensidad promedio
N	Número total de puntos registrados en la trayectoria
var_int	Varianza de la intensidad de los puntos
var_vel	Varianza de la velocidad de los puntos
mean_fwhm	Ancho medio a mitad de altura (FWHM) de la intensidad promedio
loss	Error residual del ajuste de circunferencia por taubinSVD de Circle-Fit (PyPi)
radio	Radio estimado por taubinSVD de Circle-Fit (PyPi)
distancia_recorrida	Distancia total recorrida por el objeto
a_coef	Coefficiente 'a' de la ecuación de la parábola ajustada
b_coef	Coefficiente 'b' de la ecuación de la parábola ajustada

Tabla 8: Descripción de las características extraídas con el nuevo clasificador.

Antes de introducir estas características a un algoritmo de *machine learning* decidimos buscar “manualmente” cuáles eran las características más relevantes o el espacio de características que maximiza la separación entre las clases. Implementamos un árbol de decisión rudimentario en la forma de *sentencias if* concatenadas sobre un conjunto de videos pre-clasificado. Un grupo de los “No Bólidos” fueron distinguidos entre las categorías: aviones, animales, lluvia, focos y reflejos y otros. En este ejercicio, consideramos 956 “No Bólidos” y 290 “Bólidos” ¹⁵.

¹⁵Esos 290 “Bólidos” fueron hallados, en parte en las actividades de clasificación “manual” referidas en el apartado 5.1 (Talleres BOCOSUR) y la clasificación adicional de otros $\approx 2k$ videos.

Como mencionamos en la sección 2 existe un límite superior físico en la velocidad de un meteoróide de 72.8km/s (que aproximamos a 100km/s en la siguiente estimación). Ahora, si ese meteoróide se desplazara a 100km de altura de forma horizontal en el cielo, correspondería a una velocidad máxima de 45 deg/s que, considerando las dimensiones de un *frame* de un video ASI, podemos aproximar a 400 px/s .

En el árbol de decisión de prueba tomaremos un límite superior para la velocidad media de 300 px/s ya que todos los videos de “Bólid” seleccionados, en excepción de dos de ellos, cumplen el criterio. Resumimos los pasos en los que filtramos los datos en la figura 30. El procedimiento consistió en realizar gráficas bidimensionales de las *features* de la tabla 8 y reconocer dónde se agrupan los miembros de cada clase (*clusters*). Por ejemplo, la velocidad media puede tomar valores hasta 8000 px/s y *loss* hasta 100. No obstante, la región donde se sitúan los “Bólid” en ese plano está confinada a $\text{velocidad media} < 300\text{ px/s}$ y $\text{loss} < 4$.

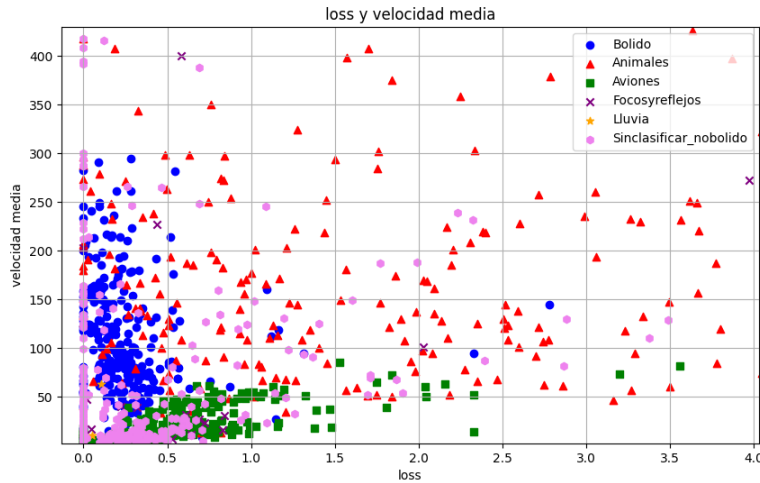


Figura 28: Velocidad media en función de *loss* sobre los datos totales del árbol de decisión de prueba.

Como podemos apreciar en el diagrama del árbol de decisión (figura 30), una misma *feature* puede aparecer en múltiples ocasiones en el proceso de tomar decisiones que dividan las clases. En el caso de la tercera decisión $\text{loss} * N > 2 \ \& \ \text{velocidad media} < 50\text{ px/s}$ (*decisión 3*) buscamos distinguir una región donde se agrupan muchos aviones (objetos que ajustan muy bien su trayectoria a una circunferencia, largos y por lo tanto hay muchos puntos registrados de su movimiento y que se desplazan relativamente lento respecto a los “Bólid”).

Con este análisis del problema logramos observar que *loss* y *velocidad media* son características muy importantes para distinguir entre las clases. Pero, aún más valioso fue explorar cómo características como *mean_fwhm* o *a_coef* y *b_coef* colaboran en la distinción de “Bólid” y “No Bólid”. Los resultados de este ejercicio no fueron lo suficientemente buenos como para desistir de la aplicación de algoritmos de *machine learning*. Por el contrario, resaltan que no existen unas pocas características que discriminen a las dos clases.

En cada “paso” o decisión que se representa en la figura 30 calculamos una matriz de confusión y la

fracción de cada categoría dentro de “No Bóidos” que “sobrevive” al filtro. En el paso final obtuvimos la siguiente matriz de confusión:

$$\begin{bmatrix} 582 & 374 \\ 19 & 271 \end{bmatrix}$$

De esta forma, logramos preservar el 93.5 % de los “Bóidos” a costa de un valor muy alto de falsos positivos: 39.1 % de los “No Bóidos”. Las condiciones del árbol de decisión fueron especialmente inadecuadas para distinguir la clase “Lluvia”, ya que en nodo terminal más profundo que pertenece a la clase positiva aún permanece el 75 % de los ejemplos.

Categoría	Falsos positivos
No Bóidos aleatorio	53.3 %
Animales	27.0 %
Aviones	27.7 %
Focos y reflejos	50.0 %
Lluvia	75.0 %

Tabla 9: Porcentaje de los falsos positivos respecto a la distribución inicial por categoría.

Distribución de falsos positivos por categorías

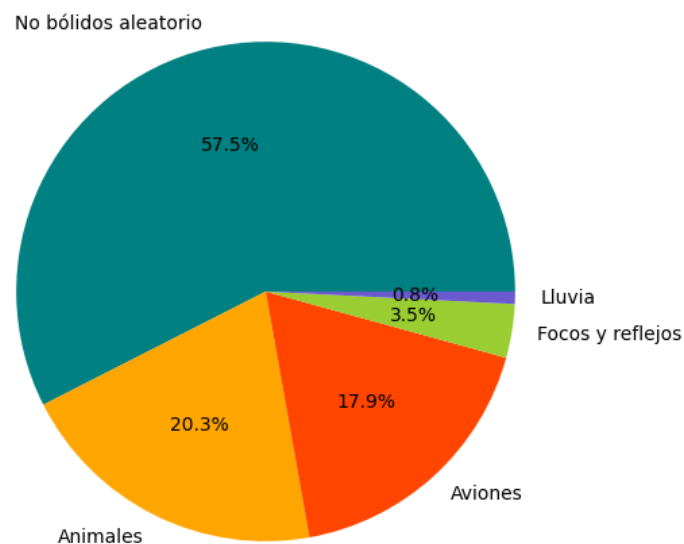


Figura 29: Distribución por categoría de los falsos positivos: gráfico de tarta.

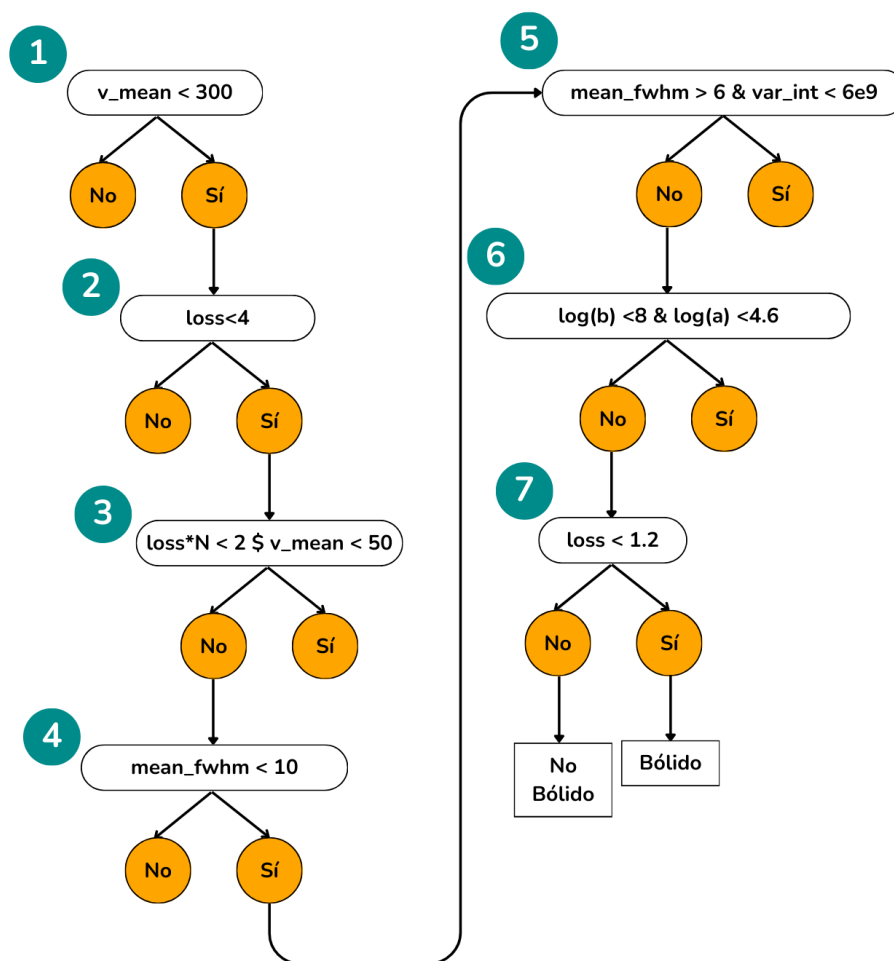


Figura 30: Estudiando el espacio de características: árbol de decisión.

El desempeño del árbol de decisión “manual” proporciona una referencia sobre las expectativas para un clasificador automático. Se espera que este último mantenga un bajo número de falsos negativos sin comprometer la tasa de falsos positivos. Todas las características o *features* de la tabla 8 serán empleadas para crear un modelo de clasificación automática.

8.4. Nuevo clasificador: entrenamiento y modelo final

Seleccionamos un grupo de 290 “Bólidos” y 553 “No Bólidos” que atraviesan el proceso de extracción de características sin valores nulos. Con el objetivo de crear un clasificador robusto y sencillo, editamos la función de entrenamiento de AUTOBOL. En esta sección analizaremos el nuevo módulo de entrenamiento implementado.

El clasificador actual emplea la librería *scikit-learn* para construir, entrenar y evaluar un modelo de clasificación utilizando un *Random Forest*. Aunque AUTOBOL originalmente utiliza *XGBoost*, elegimos este algoritmo por ser fácil de implementar y comprender, por ser más rápido y por su robustez frente al sobreajuste, ya que no contamos con una gran cantidad de “Bólidos” para el entrenamiento. Además, realizamos una búsqueda de hiperparámetros con *GridSearchCV* (de *scikit-learn*) para op-

timizar el rendimiento del modelo. *XGBoost* ofrece mayor precisión que *Random Forest* en conjuntos de datos grandes y complejos, por lo que no descartamos su uso en una futura versión del clasificador cuando se disponga de una muestra mayor de la clase positiva.

El código comienza cargando un archivo CSV resultado de la extracción de características de un grupo de videos donde deben estar las dos clases representadas (columna *label* de etiquetas). Luego, dividimos los datos en características (*features*) y etiquetas (*label*), y manejamos los valores infinitos o *Nan* que pueden surgir principalmente del ajuste de circunferencia a la trayectoria de un *set* de puntos cuasi-rectilíneos. Posteriormente, los datos se dividen en conjuntos de entrenamiento y testeo usando *train_test_split* (también de *scikit-learn*). Por defecto, se separa un 15 % de la muestra de entrenamiento para el testeo o validación del modelo.

Utilizamos el objeto *StandardScaler* (de *scikit-learn*) para normalizar los datos. “Esto convierte las características en una distribución normal estableciendo su media en 0 y su desviación estándar a 1” (Amr, 2020, p. 113). Esta etapa es útil para que el modelo no se vea afectado por escalas diferentes de las características. El escalado se realiza tanto en los datos de entrenamiento como en los de prueba.

Definimos un espacio de búsqueda para los hiperparámetros del *Random Forest* utilizando un diccionario (*param_grid*). Empleamos *GridSearchCV* para realizar una búsqueda exhaustiva con validación cruzada. A continuación, se detalla la importancia de cada uno de los hiperparámetros considerados:

- **n_estimators:** Representa el número de árboles en el bosque. Un mayor número de árboles puede mejorar la precisión del modelo, pero también aumenta el tiempo de cómputo.
- **max_depth:** Indica la profundidad máxima de los árboles. Limitar la profundidad puede prevenir el sobreajuste, permitiendo que el modelo generalice mejor.
- **min_samples_split:** Especifica el número mínimo de muestras requeridas para dividir un nodo interno. Valores más altos pueden hacer que el modelo sea más robusto y menos propenso al sobreajuste.
- **min_samples_leaf:** Define el número mínimo de muestras que debe tener un nodo hoja. Este parámetro ayuda a suavizar el modelo, reduciendo la variabilidad.

Optimizar estos hiperparámetros mediante *GridSearchCV* nos permite encontrar el balance adecuado entre sesgo y varianza, asegurando que el modelo sea tanto preciso como generalizable (Scikit-learn, 2024). En el modelo final encontramos como mejores hiperparámetros a los siguientes:

Hiperparámetro	Valor
max_depth	5
min_samples_leaf	1
min_samples_split	10
n_estimators	100

Tabla 10: Mejores hiperparámetros encontrados mediante *GridSearchCV*.

La métrica de evaluación principal utilizada es el *F2-score*¹⁶ debido a su énfasis en reducir los falsos negativos, siendo definida en la sección 6. Con los mejores hiperparámetros encontrados (tabla 10), entrenamos un nuevo modelo de *Random Forest*. Este modelo se generó utilizando las 11 *features* de la tabla 8.

El rendimiento del modelo final en el conjunto de datos de prueba se evalúa utilizando varias métricas. Los resultados son los siguientes:

Métrica	Valor
<i>Accuracy</i>	0.8957
<i>Precision</i>	0.7973
<i>Recall</i>	0.8939
<i>F2-score</i> (mejor score)	0.8147

Tabla 11: Métricas de evaluación del modelo.

La matriz de confusión para los datos de prueba se muestra a continuación, donde podemos observar que logramos mantener un porcentaje tolerable de falsos negativos (10.6 %) y de falsos positivos (10.3 %).

$$\begin{bmatrix} 130 & 15 \\ 7 & 59 \end{bmatrix}$$

De todas formas, esperamos que el modelo aplicado a nuevos datos (que no participaron del entrenamiento) presente un rendimiento menor a lo ilustrado en los datos de prueba anteriores, en ese contexto realmente evaluaremos su desempeño.

Además, generamos una gráfica que muestra la importancia de las características, proporcionando una medida de cuánto contribuye cada característica o variable a la capacidad predictiva del modelo. La importancia de una característica se calcula como la reducción media de la impureza (*Gini impurity* o *entropy*) proporcionada por esa característica a través de todos los árboles en el bosque. En otras palabras, cuantas más veces una característica se utiliza para dividir los nodos en los árboles y cuanto más profundo sea el árbol en que se utiliza esa característica, mayor será su importancia (Breiman, 2001).

¹⁶Como vimos en la sección 6 la métrica *f-score*, dependiendo del valor de β le puede dar mayor peso a los falsos positivos o falsos negativos. En el caso de *f2-score* enfatiza el rol de los falsos negativos.

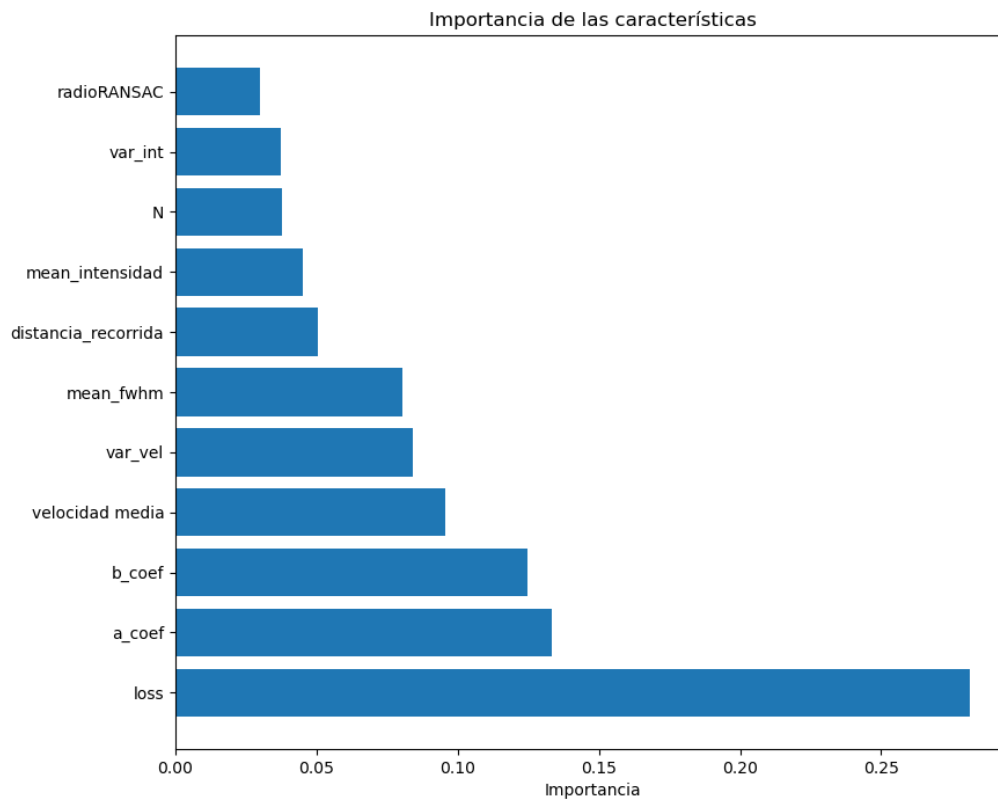


Figura 31: Importancia de las características del modelo final del nuevo clasificador.

Guardamos el modelo entrenado y el *scaler* en archivos binarios para uso futuro utilizando funciones de la librería *joblib*. En el nuevo módulo de predicción, se requieren estos archivos para realizar predicciones sobre otro conjunto de datos (también en formato CSV). Se escalan los nuevos datos de la misma manera que en el modelo y a cada conjunto de *features* que identifican a un video en particular, se le asigna una clase con cierta probabilidad.

9. Bóldos encontrados con el nuevo clasificador

Aplicamos el modelo descrito en la sección anterior sobre la totalidad de videos ASI de la red BOCOSUR ¹⁷. De esos 118222 videos, el 43.9 % pasaron satisfactoriamente los filtros de la función *tracking* (su trayectoria tiene al menos tres puntos identificados) y el 5.7 % fueron clasificados como “Bóldos”. Consideramos “Bóldos” a aquellos con una probabilidad de pertenecer a la clase positiva mayor al 50 %.

La distribución de las predicciones por estación se detalla en la tabla 12. La proporción de “Bóldos” detectados por estación, calculada como el cociente entre las instancias predichas positivas y las instancias totales de cada estación, presenta una varianza de menos del 3 %. Sin embargo, sospechamos que la discrepancia entre las estaciones se debe a la sensibilidad del modelo a los datos de entrenamiento. Aunque tomamos “Bóldos” de todas las estaciones, la cantidad de videos pre-clasificados depende fuertemente del tiempo en que cada estación ha estado operativa. Conviene destacar que las primeras tres estaciones se encuentran recolectando datos (con cámaras ASI) desde hace prácticamente dos años. Esto resulta que en estaciones como la 11 se obtenga un porcentaje algo más bajo de clasificaciones positivas (2.3 %).

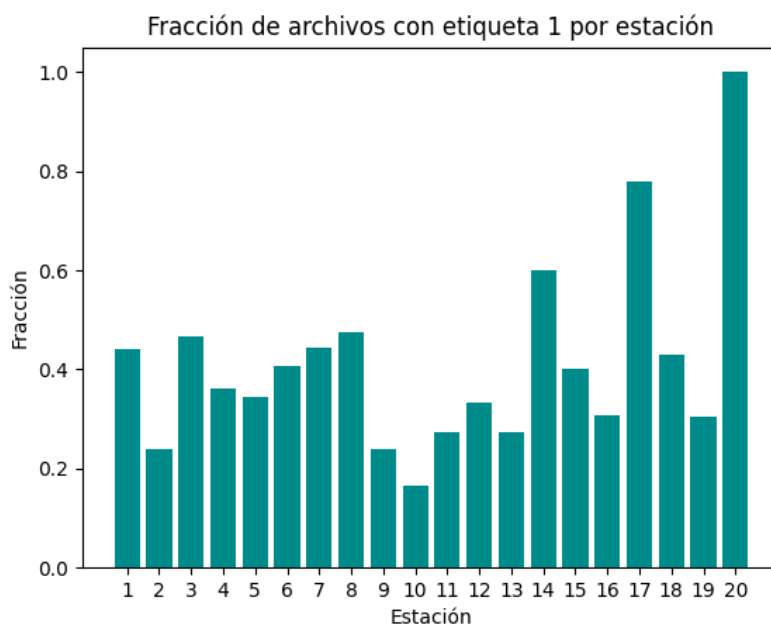


Figura 32: Resultados de la inspección visual de 300 videos aleatoriamente seleccionados que fueron clasificados como “Bóldos”.

Como mencionamos anteriormente no suponemos que los 6771 videos clasificados como “Bóldos” realmente lo sean. El modelo indicaba aproximadamente 10 % de falsos positivos, lo consideramos como una cota inferior del desempeño en campo. Para estimar el porcentaje de falsos positivos seleccionamos una muestra aleatoria de 300 de esos videos clasificados como “Bóldos” y encontramos que 112 eran efectivamente bóldos, o sea 37.3 %. Suponiendo que esta fracción se mantiene al extrapo-

¹⁷Descartamos previamente aquellos generados en noches que superan los 170 videos.

Estación	# videos	Post tracking	“Bólidos”	Noches de observación	Bólidos/noche
1	12138	6421	616	606	1.02
2	7326	3985	430	292	1.47
3	6759	2809	564	402	1.40
4	6172	2540	476	213	2.23
5	12003	5692	739	294	2.51
6	7101	3645	635	335	1.90
7	9101	2910	255	233	1.09
8	7498	3297	454	283	1.60
9	7271	3776	525	233	2.25
10	3523	1456	124	86	1.44
11	7930	1202	179	188	0.95
12	793	452	61	75	0.81
13	2541	1228	138	56	2.46
14	2770	2029	108	68	1.59
15	6406	2631	349	211	1.65
16	5024	2822	327	154	2.12
17	3810	1693	253	230	1.10
18	1309	647	69	41	1.68
19	8223	2493	438	203	2.16
20	524	171	31	20	1.55
TOTAL	118222	51899	6771	4223	1.60

Tabla 12: Resumen de datos analizados por el clasificador distribuidos por estación.

lar, esperamos encontrar alrededor de 2500 bólidos reales. Aunque pueda parecer un valor pequeño, representa un aumento de la muestra de “Bólidos” hallados durante las experiencias de clasificación “manual”, en un factor de 8.

En el marco del estudio del desempeño del clasificador notamos dificultades en el *tracking* de bólidos de corta duración que presentan fragmentación, donde es difícil reconocer la traza de los mismos. Este es un punto de partida a tener en cuenta en una nueva muestra de entrenamiento o en un algoritmo más avanzado a implementarse.

La gráfica de la figura 32 muestra un comportamiento no homogéneo en la capacidad predictiva del clasificador entre las distintas estaciones. Encontramos en cada estación que existen objetos particulares con características similares a las de un “Bólido”, exclusivas de esa estación. Por ejemplo, las aves que se observan en la estación 6, los aviones de la estación 1 o las luciérnagas de la estación 8. Al clasificar manualmente los candidatos a bólidos también obtendremos una base de datos de “No Bóli-

dos” más significativa, ya que desafía lo esperado por el modelo y nos posibilita entrenarlo nuevamente.

A partir de los valores presentados en la tabla 12 y de la inspección visual aleatoria, podemos realizar algunas consideraciones. En la quinta columna tenemos las noches de observación individuales de cada estación. Ahora, considerando que hay fechas donde coinciden tales observaciones, podemos decir que hay 715 noches donde al menos una de las estaciones se encontraba operativa y generó datos efectivos (que fueron transferidos al servidor central). Entonces, si detectamos al menos 2500 bólidos en ese tiempo, podemos estimar que se observan por noche 0.6 bólidos por estación y 3.5 bólidos en toda la red. Esos 3.5 bólidos pueden ser eventos que se observan de forma simultánea desde distintas estaciones, eventos que llamamos multi-estación.

9.1. Eventos multi-estación

Buscamos eventos multi-estación a partir de coincidencias temporales de los eventos. Esta información la extraemos fácilmente del nombre de los videos que indican número de estación, fecha y hora. Contemplando que puede haber variaciones entre la hora PC de las estaciones involucradas, establecemos una tolerancia de 300s en nuestra búsqueda. Dentro de los videos clasificados como “Bólidos” tenemos 1099 eventos candidatos a observaciones multi-estación de un mismo bólido. De esos 1099 solo 34 corresponden a observaciones desde 5 o más estaciones.

De la misma manera que encontramos que el número de falsos positivos es mayor al esperado, tenemos presente que debe haber un número mucho más alto de falsos negativos. Sería particularmente una pérdida no identificar bólidos brillantes que pertenecen a un evento multi-estación. Ampliamos la búsqueda a casos donde en una estación se clasificó un evento como “Bólido” y en una cercanía temporal se encuentra un evento brillante (estimamos esta propiedad a partir de la *feature*: **mean_intensidad**). Establecimos un umbral de brillo donde nos interesan los eventos que cumplen **mean_intensidad** > 7000 en las unidades del problema.

Así, encontramos otros 1851 nuevos candidatos a observaciones multi-estación, además de los 1099 anteriores.

9.2. Lluvias de meteoros mayores

Aplicando un método análogo, emprendimos la búsqueda de eventos candidatos a pertenecer a lluvias de meteoros conocidas en el hemisferio sur. De este modo, restringimos la búsqueda con el fin de computar el radiante y órbita de estos eventos en el futuro, en el caso de que sean efectivamente observaciones de bólidos (esa es la forma definitiva de determinar la pertenencia). Presentamos en la tabla 13 la información pertinente. Estos valores se ven directamente afectados por el número de estaciones operativas durante el año, como podemos apreciar en las figuras 33 y 34. En dichas figuras se muestra la pobre cobertura de la red BOCOSUR durante los primeros meses del año 2022 (aún en proceso de instalación de las estaciones) y en el segundo semestre del 2023 (por un conjunto de dificultades de operación).

La distribución de “Bólidos” durante los años 2022 y 2023 presentan coincidencias con las lluvias de meteoros mayores resumidas en la tabla 13. En ambos años existe muy buena coincidencia de los picos de la gráfica con los máximos de las Lyridas, Eta Aquaridas, Delta Aquaridas, Tauridas del Sur,

Nombre de la lluvia	Code	Inicio	Máximo	Fin	Candidatos 2022	Candidatos 2023
Lyridas	LYR	04-16	04-22	04-25	26	244
Eta Aquaridas	ETA	04-19	05-05	05-28	194	971
Delta Aquaridas	SDA	07-12	07-30	08-23	233	333
Perseidas	PER	07-17	08-13	08-24	224	297
Orionidas	ORI	10-02	10-22	11-07	426	159
Tauridas del Sur	STA	09-10	10-07	11-20	845	294
Leonidas	LEO	11-06	11-17	11-30	434	63
Geminidas	GEM	12-04	12-14	12-17	271	66
Ursidas	URS	12-17	12-22	12-26	193	40

Tabla 13: Lluvias de meteoros visibles desde el hemisferio sur y “Bólidos” candidatos en los años 2022 y 2023. Fuente: *International Meteor Organization*.

Geminidas y Ursidas. Esas coincidencias ofrecen mayor fiabilidad a las predicciones del clasificador automático.

También es necesario estudiar qué eventos no se repiten en esos años. Existen algunos picos de actividad que no coinciden con las lluvias de meteoros mayores. Por ejemplo, el pico del 27 de junio de 2022 coincide con el máximo de la lluvia variable Boötidas de junio (JBO) de ese año. También, a mediados de junio de 2023 hay un pico considerable que requiere de estudio particular para descartar que no sean detecciones espurias. Además, en el primer cuatrimestre de 2023 se reconocen 6 picos con más de 35 meteoros por noche que requieren de un análisis más profundo para determinar si corresponden a lluvias menores o lluvias no identificadas.

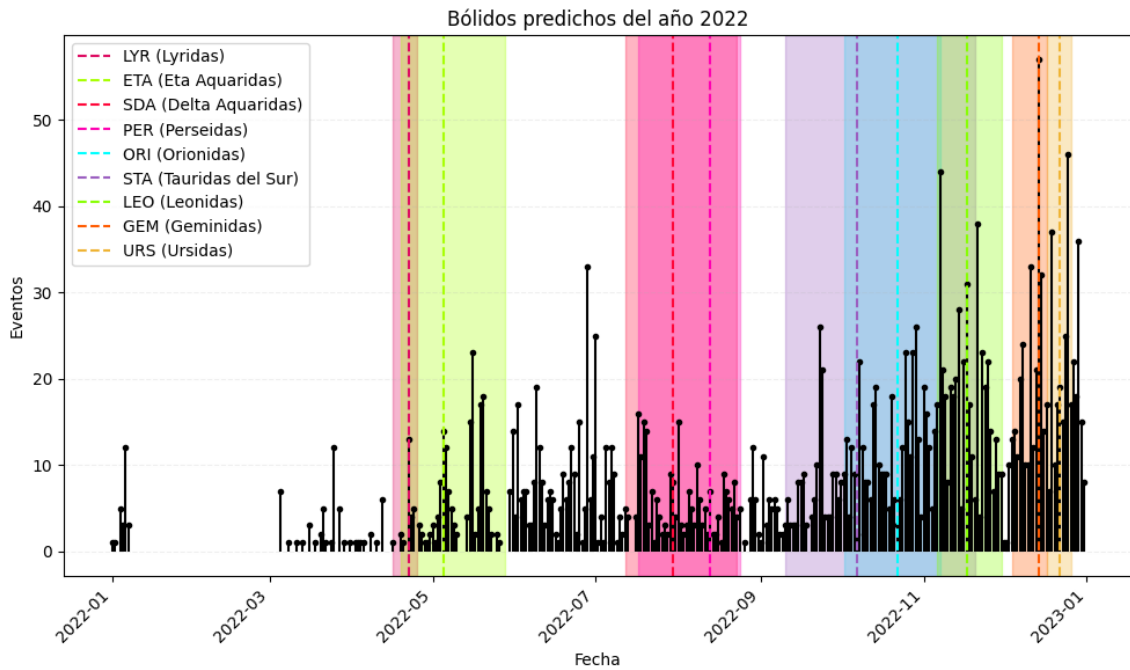


Figura 33: “Bólidos” hallados en el 2022.

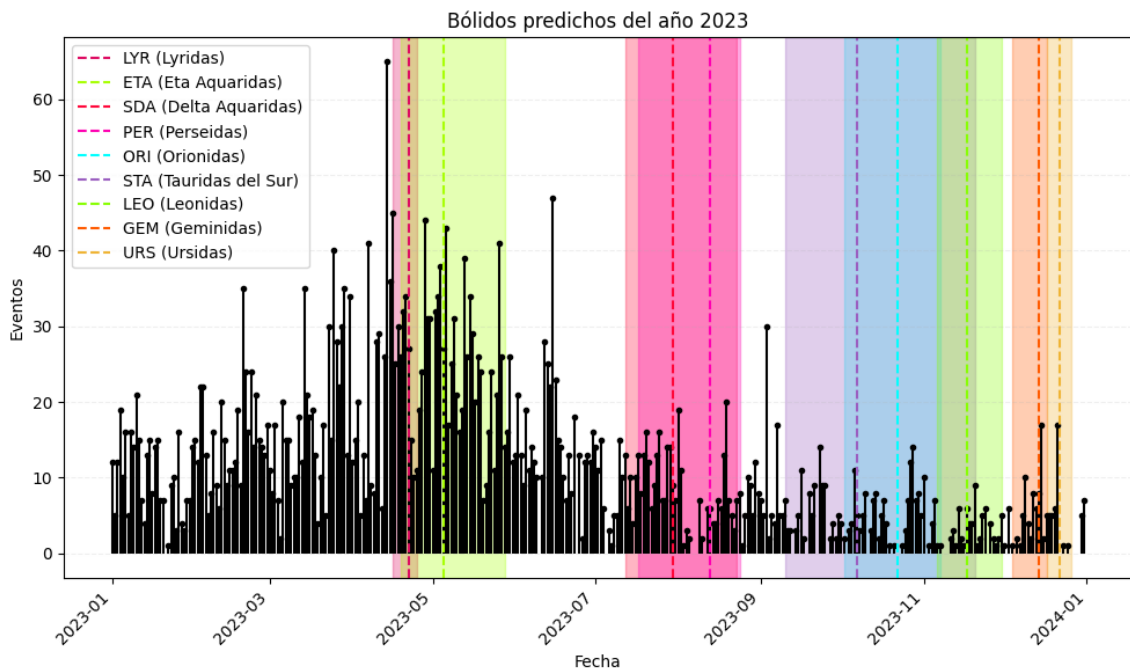


Figura 34: “Bólidos” hallados en el 2023.

“Bólidos” predichos por el clasificador automático en los años 2022 y 2023. Se superpone información de las lluvias de meteoros mayores observables en el hemisferio sur, mostrando su duración con una banda vertical y el máximo de cada una de ellas como una línea vertical punteada.

10. Análisis de un caso multi-estación

A partir del procedimiento explicado en la sección anterior hallamos un evento múltiple registrado en 4 estaciones: 8, 10, 16 y 19. Este bólido se observó el 13 de mayo de 2023 aproximadamente a las 18h : 56m *HLU*. Calculamos la trayectoria, velocidad y radiante del evento utilizando códigos pre-existentes a este trabajo.

En [Caldas et al. \(2024\)](#) describimos el procedimiento de calibración astrométrica y fotométrica de la red BOCOSUR que empleamos en el tratamiento de este caso particular. Estos procedimientos se condensan en una única aplicación de post-procesamiento en MATLAB.

La calibración astrométrica transforma coordenadas de píxeles (x, y) en coordenadas horizontales (z, Az). Se utiliza una función de mapeo que ajusta un modelo no lineal mediante el algoritmo de *Levenberg-Marquardt*¹⁸, minimizando los errores entre las coordenadas calculadas y las de un catálogo estelar. Este proceso se realiza iterativamente, descartando los datos con mayor error hasta alcanzar un umbral predefinido. Las constantes de placa, que incluyen las coordenadas del centro de proyección (COP), los coeficientes de un polinomio que relaciona la distancia desde el centro con el ángulo desde el eje óptico, y los ángulos de rotación y desviación, se determinan durante este procedimiento.

Realizamos la calibración astrométrica para las 4 estaciones del problema, identificando ≈ 200 estrellas en cada una de ellas, obteniendo una resolución promedio de 6 *arcmin/px*.

Estación	#Estrellas	Mean O-C z (arcmin)	Mean O-C Az (arcmin)	Mean O-C total (arcmin)	Resolución (arc- min/px)
8	238	7	4	6	6
10	218	6	3	5	6
16	174	6	3	5	6
19	155	9	5	7	6

Tabla 14: Errores astrométricos y resolución de la solución astrométrica de las estaciones donde se observó el evento multi-estación.

Para la calibración fotométrica, se identifican estrellas en un marco integrado y se realiza fotometría de apertura circular en sus posiciones (x, y). Se ajusta un modelo que incluye el coeficiente de extinción atmosférica y un punto cero, utilizando también el algoritmo de *Levenberg-Marquardt*. Este modelo permite extrapolar la magnitud de objetos brillantes, como bólidos, a partir de la magnitud de las estrellas. Esta extrapolación puede realizarse incluso cuando los bólidos saturan la imagen, haciendo una reconstrucción del perfil de brillo de los mismos ([Caldas et al., 2024](#)).

Una vez que se han completado los pasos anteriores, la aplicación de post-procesamiento permite obtener la trayectoria del bólido en el cielo por cada estación, al igual que mediciones de flujo que se convierten a magnitud aparente.

¹⁸Ver documentación de su implementación en MATLAB en <https://es.mathworks.com/help/optim/ug/least-squares-model-fitting-algorithms.html>.

Obtenemos para cada observación del mismo evento mediciones de tiempo, z , Az y magnitud aparente. Para procesar estos datos y determinar la trayectoria del bólido, dado que aún nos encontramos perfeccionando esta parte del *pipeline*, utilizamos dos códigos de MATLAB basados en métodos de intersección de planos. Uno de ellos implementa el método *MOP* de [Ceplecha \(1987\)](#) (este código fue utilizado en el análisis de un evento multi-estación en el trabajo de [Caldas et al. \(2024\)](#)). El otro método también resuelve la intersección de planos pero con un enfoque ligeramente distinto al *MOP* original y fue desarrollado por G. Tancredi (comunicación personal, 2024). En el resto del trabajo, será denominado como método *Tancredi*.

Como explica [Jansen-Sturgeon et al. \(2020\)](#), existen tres métodos principales para la triangulación de meteoroides, donde *MOP* es el más antiguo y menos preciso, pero es computacionalmente simple. Tiene cuatro pasos clave: la construcción del plano, formación del radiante, determinación de la posición y ajuste de la velocidad.

Primero, se construye un plano para cada sensor o estación: cada sensor captura las observaciones del meteoroides en su campo de visión y dichas observaciones son una serie de líneas de visión o *rays* que se proyectan desde la ubicación del sensor hacia la trayectoria observada del meteoroides en el cielo. Luego, se intersectan estos planos de a pares para determinar la trayectoria en línea recta. A partir de estas rectas, se halla una trayectoria promedio, sobre la que se proyectan las observaciones de cada estación. Las posiciones a lo largo de la trayectoria se determinan como los puntos más cercanos en la línea de trayectoria desde la línea de visión observada.

Finalmente, las velocidades se ajustan mediante un modelo posicional en función del tiempo. [Jansen-Sturgeon et al. \(2020\)](#) remarca que, como indican los autores del procedimiento, esos modelos pueden no ser precisos para trayectorias largas, donde la hipótesis de una trayectoria rectilínea pierde su validez. Para resolver ese problema, desarrolla DTF, un método que resuelve la trayectoria sin partir de esa hipótesis. Esto no nos debe preocupar para el bólido del 2023/05/13 ya que la completitud del fenómeno luminoso tiene una duración menor a $2s$.

A continuación mencionamos brevemente el procedimiento del método alternativo *Tancredi*. En primer lugar, se calculan las posiciones de los observadores en un sistema XYZ utilizando las coordenadas UTM. A partir de las coordenadas (z, Az) en cada *frame*, se obtiene un versor dirección para cada posición del meteoroides en el cielo. Con estos versores, se calcula un versor perpendicular al plano que los contiene. Para cada estación, se resuelve un sistema de ecuaciones homogéneas utilizando el método SVD (*Single Value Decomposition*). Una trayectoria rectilínea en el espacio se define por un versor y un punto en la trayectoria. La trayectoria rectilínea debe cumplir las siguientes condiciones: i) la normal a cada uno de los planos de la visual debe ser perpendicular a la trayectoria; ii) la normal a cada uno de los planos de la visual debe ser perpendicular a una recta que une al observador con un punto de la trayectoria. Esto nos define dos sistemas de ecuaciones, uno homogéneo y otro heterogéneo, donde debemos hallar el versor a lo largo de la trayectoria y un punto de la misma que satisfagan esos sistemas. Estos sistemas se resuelven también con el método SVD. Para la búsqueda de la solución de mínimos cuadrados, se puede dar diferente peso a los datos de cada estación; en la solución que se presenta se optó por asignar pesos según el número de datos (frames) de cada observación, con lo cual los videos más largos, y generalmente los más extensos en términos angulares, tienen más peso.

La intersección de los planos, con el método *Tancredi*, se representa en las figuras 35 y 36. Se muestra los planos de observación y la trayectoria desde dos perspectivas diferentes, una alineada con

la trayectoria.

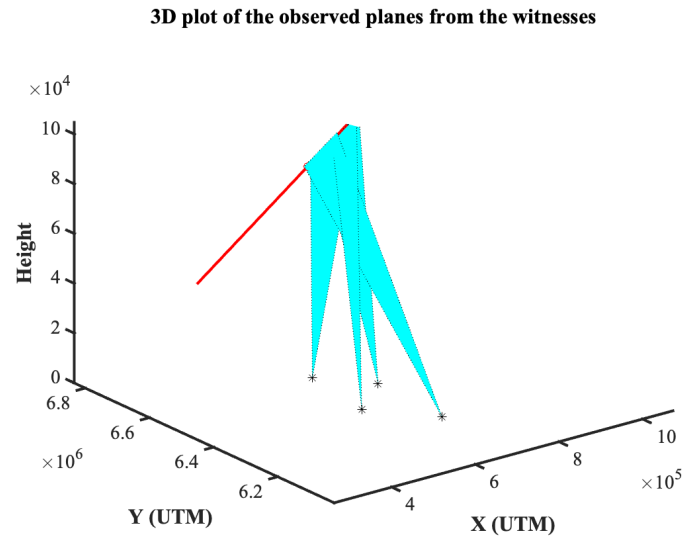


Figura 35: Representación de los planos de observación destacando las perspectivas de cada estación de observación.

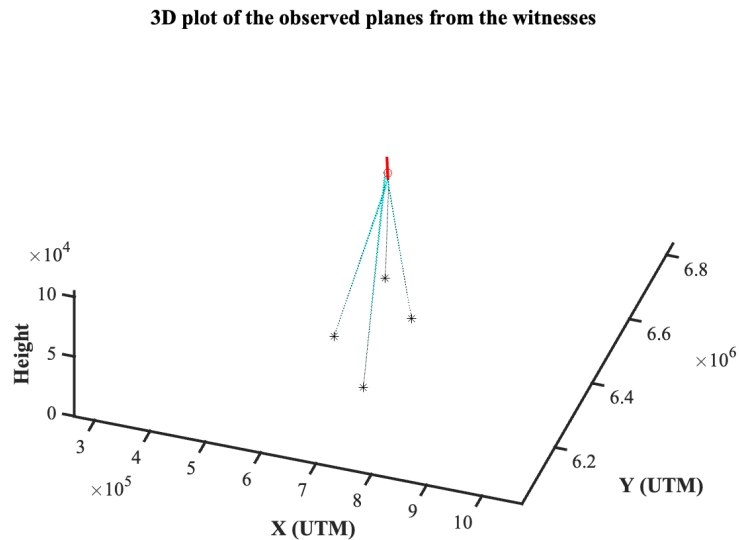


Figura 36: Representación de los planos de observación alineada a la trayectoria.

Planos de observación con el método *Tancredi*. En celeste se representan los planos, en rojo su intersección (la recta que contiene la trayectoria del meteoróide en la fase luminosa) y en el plano XY (sistema de coordenadas universal transversal de Mercator o UTM) se indica la posición de las estaciones “testigo” del evento.

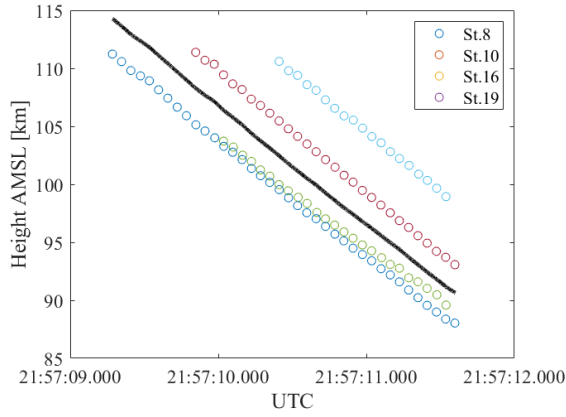


Figura 37: Altura sobre nivel del mar promedio en función del tiempo (UTC).

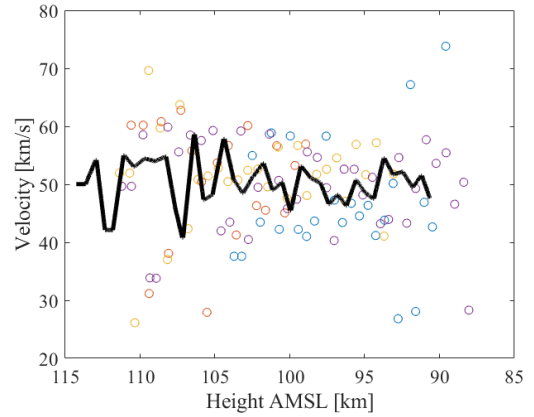


Figura 38: Velocidad en función de la altura sobre nivel del mar promedio.

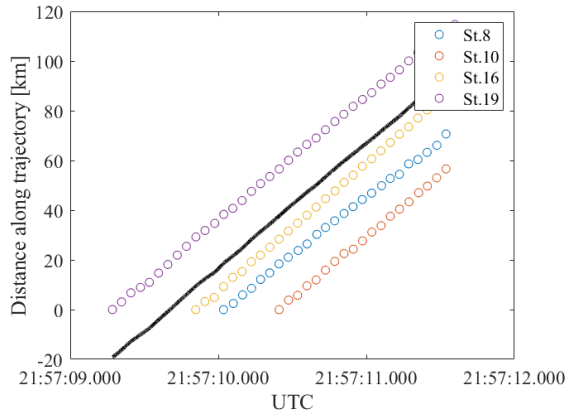


Figura 39: Distancia a lo largo de la trayectoria.

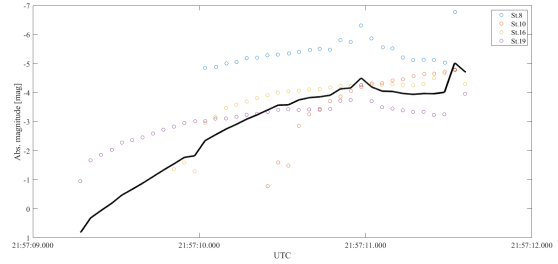


Figura 40: Curva de luz (magnitud absoluta en función del tiempo).

Resultados de la determinación de la trayectoria del bólido en base al método *MOP*.

Según las figuras 37 y 38, el meteoróide viaja de una altura de $\approx 115\text{km}$ a $\approx 90\text{km}$ en su recorrido (o de 104.8km a 84.2 según la implementación del método *Tancredi*), con una velocidad de $\approx 50\text{km/s}$ que no parece mostrar desaceleración considerable. Esto es consistente con la expresión de la desaceleración propuesta por [Halliday et al. \(1996\)](#) para meteoroides a gran altura y tamaño mayor que unos pocos *cm*. Por lo tanto, es adecuado aproximar la velocidad inicial del meteoróide con la velocidad promedio.

Además, de las figuras 37 y 39 vemos que el objeto ingresa de forma rasante, recorriendo $\approx 120\text{km}$ a lo largo de su trayectoria en apenas $\approx 25\text{km}$ de altitud.

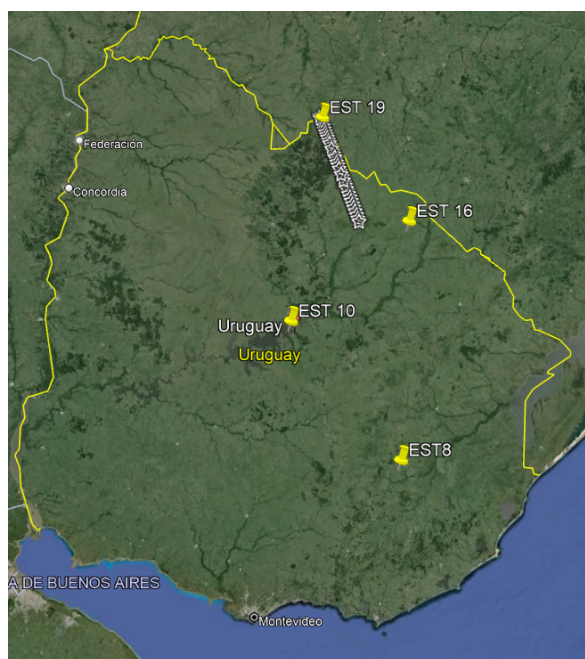


Figura 41: Trayectoria determinada del bólido proyectada sobre la superficie terrestre.

Resultados de la determinación de la trayectoria del bólido en base al método *MOP*.

En la figura 41 se representa la dirección en la que se desplazó el meteoro proyectado sobre la superficie de Uruguay, habiéndose desplazado en dirección aproximada SE-NO. La proyección en la figura 42 puede ser engañosa, dando a entender que la posición de caída es muy cercana a la estación 19.

En la figura 43 es muy llamativo el comportamiento de los datos temporales de la estación 10, notamos un desfase temporal de $\approx 30s$ en contraste a las estaciones 8, 16 y 19. Podemos atribuir este retraso a un desperfecto del funcionamiento del GPS en esa estación. Entonces, el tiempo en la estación 10 para ese evento fue extraído directamente de la hora PC.

En la implementación del método *MOP*, la variación de tiempo entre las estaciones se resuelve mediante una correlación basada en la curva de luz del evento, con el objetivo de sincronizar temporalmente las observaciones de diferentes estaciones. Dado que todas estas curvas corresponden al mismo evento, es posible identificar los máximos en una estación “ancla” (donde el GPS funciona adecuadamente) y buscar esos mismos máximos en las curvas de luz de las demás estaciones. En el caso del método *Tancredi*, la información temporal solo se utiliza para el cálculo de la velocidad para los datos de cada estación; donde se proyecta cada visual de observación sobre la trayectoria, asignándole el tiempo en el video, y se considera la variación relativa de los tiempos para el cálculo de la velocidad en cada estación. Finalmente, para el cálculo de la velocidad promedio se hace un promedio ponderado por el número de observaciones.

Un aporte de la figura 45 es la extrapolación más allá del fenómeno luminoso, implementando un análisis del vuelo oscuro basado en las ecuaciones de (Ceplecha et al., 1998, p. 390-392). El código de Tancredi resuelve dichas ecuaciones de movimiento de un cuerpo sin ablación. La solución de estas

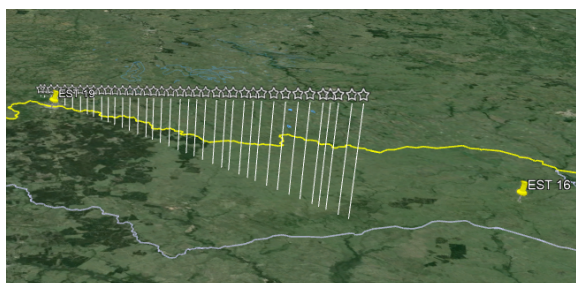


Figura 42: Desplazamiento tridimensional del meteoro. La estación 19 lo observa a una altitud menor ($\approx 85 km$).

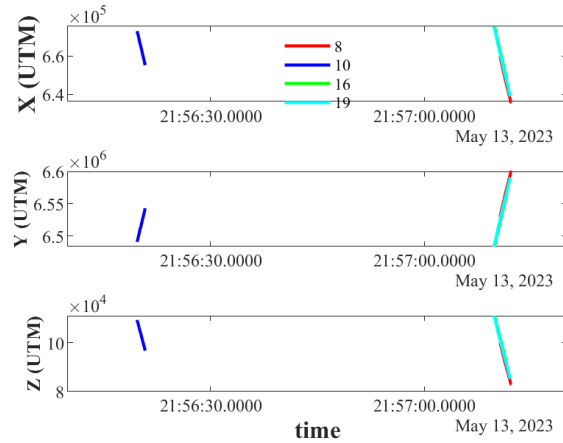


Figura 43: Posición XYZ en función del tiempo para cada estación (método *Tancredi*). Observamos un retraso temporal respecto a las observaciones de la estación 10.

ecuaciones diferenciales requiere como condiciones iniciales a la última velocidad medida durante el fenómeno luminoso y la desaceleración en el punto terminal, la posición del punto terminal y la dirección de la trayectoria.

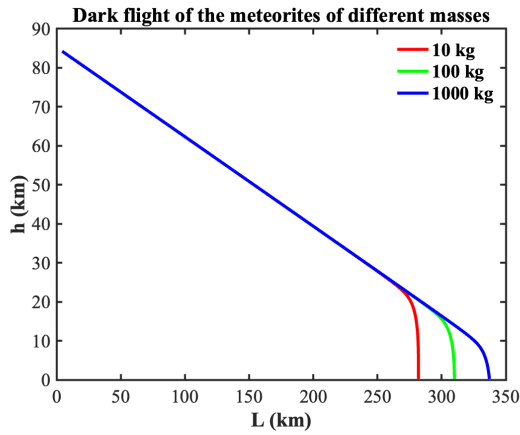


Figura 44: Trayectoria durante el vuelo oscuro para distintos valores de masa del meteoróide. La distancia L se mide en dirección de la proyección de la trayectoria y la superficie.

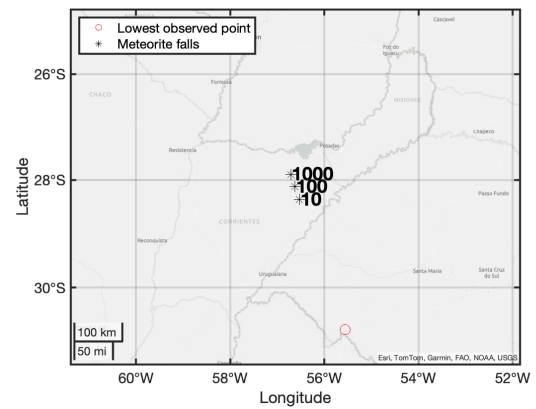


Figura 45: Posibles ubicaciones de caída del meteorito en función de las masas consideradas ($10kg$, $100kg$ y $1000kg$).

Resultados de la solución de las ecuaciones de movimiento del meteoróide en el vuelo oscuro.

Entonces, si la ablación del meteoróide no consumió por completo su masa, el posible meteorito cayó en Argentina. El lugar exacto depende de la masa del objeto.

Con ambas implementaciones se obtiene el cálculo del radiante y velocidad de ingreso del meteoróide a la atmósfera.

Cantidad	MOP	Tancredi
Ascensión Recta (deg)	279.76 ± 0.21	279.9 ± 0.24
Declinación (deg)	-62.56 ± 0.04	-63.95 ± 0.17
Velocidad (km/s)	49.15 ± 0.06	50.3 ± 0.1

Tabla 15: Radiante y velocidad exo-atmosférica del bólido analizado con el método *MOP* y el método alternativo de *Tancredi*. Las incertidumbres se calcularon en base a clones considerando los errores astrométricos de cada estación.

De acuerdo a la tabla 15 no observamos grandes discrepancias entre la aplicación de estos métodos de intersección de planos. Los valores computados no corresponden a las lluvias de meteoros mayores (ver 13) pero aún no podemos afirmar que se trata de un evento esporádico. Es necesario realizar una búsqueda más exhaustiva, considerando incluso otras lluvias no confirmadas al igual que eventos detectados por la red BOCOSUR en fechas similares en ese y otros años.

A su vez, del radiante calculado y la velocidad (corregida por la rotación terrestre) se obtienen, con el código del método *Tancredi*, los elementos orbitales del meteoróide en su órbita heliocéntrica antes de la captura por la Tierra (ver tabla 16).

Elemento Orbital	Valor calculado	Desviación estándar (σ)
a	-52.811	
1/a	-0.019	0.012
e	1.0113	0.0071
i	82.176	0.290
Ω	232.56	0.00
ω	79.41	0.48
M	0.10	179.76

Tabla 16: Cálculo de los elementos orbitales y su desviación estándar como el error asociado a cada medida. Este error se calcula a partir de la iteración con 1000 clones, asumiendo una distribución normal de los errores astrométricos. Para ello, se generan clones de las observaciones según los errores astrométricos medidos. Se produce un nuevo par (z, Az) para cada dato de observación, centrado en la observación real y sumado un número aleatorio con distribución gaussiana de media 0 y desviación estándar igual al error astrométrico en cada coordenada.

Los valores de la tabla 16, ofrecen información sobre el origen del meteoróide. La alta inclinación y el valor de $1/a$, considerando una órbita hiperbólica (o incluso elíptica si tomamos una incertidumbre de 2σ), describen la órbita de un cometa de la Nube de Oort. Esto implica que tiene muy poca probabilidad de ser un cuerpo rocoso y en consecuencia, resultar en la caída de un meteorito.

El sistemático estudio de eventos multi-estación de la red BOCOSUR contribuirá a nuestro entendimiento de meteoros esporádicos así como de los pertenecientes a lluvias.

11. Conclusiones y futuros pasos

Hemos participado activamente en el mantenimiento y operación de la red BOCOSUR, tanto de forma remota como realizando visitas de mantenimiento a las estaciones. En el proceso de construcción de la base de datos de las cámaras actuales de la red, hemos duplicado los datos disponibles en el servidor local mediante la transferencia periódica de archivos.

Para avanzar en la construcción de una base de datos de videos clasificados, inicialmente nos dedicamos a la clasificación manual o inspección visual de videos, seguida de la implementación de un clasificador automático. Se clasificaron manualmente aproximadamente $7k$ videos, identificando 290 bólidos que luego se utilizaron como base de entrenamiento para el clasificador automático.

Adaptamos AUTOBOL, un clasificador automático diseñado específicamente para videos de la red BOCOSUR con el sistema de adquisición anterior. La nueva función de seguimiento implementada resultó efectiva en la identificación de trazas de diversos tamaños, permitiendo la detección de meteoros y bólidos de diferentes características. Sin embargo, observamos un rendimiento limitado en bólidos de corta duración y fragmentación pronunciada, lo cual dificulta su identificación. Esta función de seguimiento permitió una primera depuración significativa de videos espurios, encontrando que aproximadamente el 56 % de los 118k videos analizados no contienen trazas detectables y son probablemente espurios.

El nuevo clasificador desarrollado identificó alrededor del 5.7 % del total de videos como “Bólidos”, lo que equivale a $6.7k$ videos. Estimamos que al menos $2.5k$ de estos son verdaderos positivos, basándonos en un análisis de una submuestra aleatoria de 300 videos. Considerando las noches operativas de las estaciones, calculamos un límite inferior de 0.6 bólidos detectados por noche por estación y 3.5 bólidos por noche en toda la red.

Para mejorar la clasificación de bólidos, planeamos aplicar técnicas más avanzadas de aprendizaje automático que permitan un desempeño robusto en diversas clases de bólidos, incluyendo aquellos con características de fragmentación.

Los videos candidatos a contener bólidos están siendo clasificados manualmente como parte de un proyecto de Ciencia Ciudadana, en colaboración con instituciones de educación media del país. Esta clasificación se realiza a través de una aplicación web accesible, lo cual facilita su uso por parte de los usuarios¹⁹.

Además, hemos identificado aproximadamente $3k$ eventos multi-estación, buscando entre los bólidos detectados y otros eventos muy brillantes. Uno de estos eventos, detectado el 13 de mayo de 2023 a las 18h : 56m HLU por cuatro estaciones, fue analizado en detalle y se estima que podría tener origen cometario. Aunque las características del radiante y la velocidad no coinciden con las lluvias de meteoros conocidas, se necesitan estudios adicionales para determinar su naturaleza como evento esporádico o parte de otra lluvia de meteoros.

Al observar la distribución anual de los bólidos clasificados por el algoritmo de aprendizaje automático, hemos identificado coincidencias con los picos de actividad de las principales lluvias obser-

¹⁹La dirección de la aplicación web es http://astroweb.fisica.edu.uy/users/sign_in. Estamos trabajando en la creación de usuarios bajo demanda.

vadas desde el hemisferio sur. Continuaremos investigando aquellos picos que no están asociados con estas lluvias, con el objetivo de contribuir al estudio de lluvias variables y lluvias menores.

La evolución del pipeline de la red BOCOSUR implicarán una optimización del programa de detección de meteoros que actúa in situ en las estaciones, re-entrenamiento de un modelo de clasificación habiendo aumentado la base de entrenamiento de meteoros y el desarrollo posterior (adaptando implementaciones ya existentes) de programas que permitan la determinación de la trayectoria de forma más precisa, así como la determinación de órbita del meteoróide y descripción del vuelo oscuro para bólidos muy brillantes y candidatos a dar lugar a meteoritos.

Bibliografía

- Amaral L. S., et al., 2018, in Gyssens M., Rault J.-L., eds, International Meteor Conference, Petnica, Serbia. pp 171–175
- American Meteor Society 2023, Glossary of Meteor Terms, <https://www.amsmeteors.org/resources/glossary/>
- Amr T., 2020, Hands-On Machine Learning with scikit-learn and Scientific Python Toolkits: A practical guide to implementing supervised and unsupervised machine learning algorithms in Python. Packt Publishing Ltd
- Anghel S., Nedelcu D., Birlan M., Boaca I., 2023, [Monthly Notices of the Royal Astronomical Society](#), 518, 2810
- Ashkenazi M., 2024, Fitting a Circle to 2D Points, <https://people.cas.uab.edu/~mosya/cl/MATLABcircle.html>
- Ballestrino J. P., Deandraya C., Uviedo C., 2022, AUTOBOL: Detección Automática de Bóolidos para Cámaras AllSky
- Bland P. A., et al., 2012, [Australian Journal of Earth Sciences](#), 59, 177
- Breiman L., 2001, Machine learning, 45, 5
- Caldas J. M., 2012, Tesis para la obtención del título de magíster en física, opción astronomía, Universidad de la República Oriental del Uruguay, Facultad de Ciencias
- Caldas M., Guaimare A., Abraham V., Barrios L., Hernández M., Velasco L., Tancredi G., 2024, BOCOSUR: An all sky network for fireball detection in Uruguay, Under submission to Meteoritics & Planetary Science
- Cecil D., Campbell-Brown M., 2020, [Planetary and Space Science](#), 186, 104920
- Cepplecha Z., 1987, Bulletin of the Astronomical Institutes of Czechoslovakia, 38, 222
- Cepplecha Z., Revelle D. O., 2005, , 40, 35
- Cepplecha Z., Borovička J., Elford W. G., Revelle D. O., Hawkes R. L., Porubčan V., Šimek M., 1998, , 84, 327
- Chen T., Guestrin C., 2016, [arXiv e-prints](#), p. arXiv:1603.02754
- Colas F., et al., 2020, [Astronomy & Astrophysics](#), 644, A53
- Cooke W. J., Moser D. E., 2012, in Proceedings of the International Meteor Conference, 30th IMC, Sibiu, Romania, 2011. pp 9–12
- Council N. R., 1998, Evaluating the Biological Potential in Samples Returned from Planetary Satellites and Small Solar System Bodies: Framework for Decision Making. The National Academies Press, Washington, DC, [doi:10.17226/6281](https://doi.org/10.17226/6281), <https://doi.org/10.17226/6281>
- Day B. H., Bland P., Sayers R., 2017, in AGU Fall Meeting Abstracts. pp ED24B–01

- Delvecchio I., et al., 2021, in European Planetary Science Congress. pp EPSC2021–30, [doi:10.5194/epsc2021-30](https://doi.org/10.5194/epsc2021-30)
- Devillepoix H. e. a., 2020, *Planetary & Space Science*, 191, 105036
- Galindo Y., Lorena A., 2018, in Anais do XV Encontro Nacional de Inteligência Artificial e Computacional (ENIAC 2018). São Paulo, pp 528–537, [doi:10.5753/eniac.2018.4445](https://doi.org/10.5753/eniac.2018.4445), <http://portaldeconteudo.sbc.org.br/index.php/eniac/article/view/4445>
- Ganju S., Hatua A., Jenniskens P., Krishna S., Ren C., Ambardar S., 2023,] 10.48550/ARXIV.2308.02664
- Gardiol D., Cellino A., Di Martino M., 2016, in Roggemans A., Roggemans P., eds, International Meteor Conference Egmond, the Netherlands, 2-5 June 2016. p. 76
- Gardiol D., et al., 2020, *Monthly Notices of the Royal Astronomical Society*, 501, 1215
- Group W. M., 2024, Western Meteor Group’s Southern Ontario Meteor Network, <https://impact.uwo.ca/fireballs/western-meteor-groups-southern-ontario-meteor-network/>
- Gural P., 2019, *Monthly Notices of the Royal Astronomical Society*, 489, 5109
- Halliday I., Griffin A. A., Blackwell A. T., 1996, , 31, 185
- Jansen-Sturgeon T., Sansom E. K., Devillepoix H. A. R., Bland P. A., Towner M. C., Howie R. M., Hartig B. A. D., 2020, , 160, 190
- Jones J., Brown P., Ellis K. J., Webster A. R., Campbell-Brown M., Krzeminski Z., Weryk R. J., 2005, , 53, 413
- Learners T. M., 2024, Métricas de clasificación, <https://www.themachinelearners.com/metricas-de-clasificacion/>
- Maimon O. Z., Rokach L., 2014, Data mining with decision trees: theory and applications. Vol. 81, World scientific
- Marsola T., Lorena A., 2019, in Anais do 14^o Simpósio Brasileiro de Automação Inteligente.
- NASA 2024a, Double Asteroid Redirection Test (DART), <https://www.nasa.gov/planetarydefense/dart>
- NASA 2024b, NASA’s Fireball and Bolide Data, <https://fireballs.ndc.nasa.gov/>
- Oberst J. e. a., 1998, *Meteoritics & Planetary Science*, 33, 49
- Peña-Asensio E., Trigo-Rodríguez J. M., Grèbol-Tomàs P., Regordosa-Avellana D., Rimola A., 2023, , 238, 105802
- Porubčan V., Kornoš L., Williams I. P., 2004, *Earth, Moon, and Planets*, 95, 697
- PyPi 2024, circle-fit 0.1.4, <https://pypi.org/project/circle-fit/>
- Roggemans P., Campbell-Burns P., Kalina M., McIntyre M., Scott J. M., Šegon D., Stano R., Vida D., 2024, eMeteorNews, 9, 56

- Scikit-learn 2024, `sklearn.ensemble.RandomForestClassifier`, `scikit-learn` Documentation, <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>
- Sennlaub R., Hofmann M., Hankey M., Ennes M., Müller T., Kroll P., Mäder P., 2022a, *Monthly Notices of the Royal Astronomical Society*, 516, 811
- Sennlaub R., Hofmann M., Hankey M., Ennes M., Müller T., Kroll P., Mäder P., 2022b, , 516, 811
- Spurný P., Borovička J., Mucke H., Svoreň J., 2017, *Astronomy & Astrophysics*, 605, A68
- Trigo-Rodríguez J. M., et al., 2004, *Earth Moon and Planets*, 95, 553
- Unión Astronómica Internacional 2017, IAU Commission F1 Nomenclature for Fundamental Astronomy, https://www.iau.org/static/science/scientific_bodies/commissions/f1/meteordefinitions_approved.pdf
- Vida D., et al., 2021, *Monthly Notices of the Royal Astronomical Society*, 506, 5046
- Vítek S., Nasyrova M., 2019, *Astronomische Nachrichten*, 340, 646
- Weryk R. J., Brown P. G., Domokos A., Edwards W. N., Krzeminski Z., Nudds S. H., Welch D. L., 2008, *Earth Moon and Planets*, 102, 241
- Whipple F. L., 1951, , 113, 464