Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Detección de actividades sospechosas de buques en tiempo real mediante datos AIS

Grado en Ingeniería Mecánica

ALUMNO: Ignacio Hernández de Armijo Jáudenes

DIRECTOR: Miguel Rodelgo Lacruz

CURSO ACADÉMICO: 2021-2022

Universidade Vigo

Centro Universitario de la Defensa
en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Detección de actividades sospechosas de buques en tiempo real mediante datos AIS

Grado en Ingeniería Mecánica

Intensificación en Tecnología Naval

Cuerpo General

Universidade Vigo

RESUMEN

A raíz del continuo cambio y adaptación de la Armada española a las exigencias de las nuevas tecnologías (*Armada 4.0*), el Almirante Jefe del Estado Mayor de la Armada (AJEMA), en 2019, propuso la apertura de un proyecto de investigación para ser realizado conjuntamente por el Centro Universitario de la Defensa (CUD) y el Centro de Operaciones y Vigilancia de Acción Marítima (COVAM). El objetivo de dicho proyecto es la aplicación de la inteligencia artificial y el procesado de grandes flujos de información (*Big Data*) para mejorar el Conocimiento del Entorno Marítimo (CEM).

Enmarcado en este proyecto, este Trabajo de Fin de Grado es propuesto para el desarrollo de un algoritmo que permita detectar actividades sospechosas de buques en tiempo real (online), a partir de datos indirectos de los mensajes AIS como la cinemática, zona de actividad o el tipo de barco y datos registrales de los buques. Para ello, se analiza una base de datos enriquecida recogida durante 15 días de mayo de 2021. El análisis se realiza para identificar indicadores de comportamiento de actividades sospechosas de ilegalidad y establecer umbrales para desarrollar un algoritmo capaz de detectar dichos buques. Los resultados muestran que es posible detectar en tiempo real posibles actividades sospechosas.

PALABRAS CLAVE

AIS, COVAM, Tiempo real, Analítica, CEM

AGRADECIMIENTOS

Quiero comenzar estas palabras de agradecimiento dirigiéndome a mi familia, que han sido el apoyo firme sobre el que todo marino debe sostenerse durante estos cinco años de carrera. A su vez, quiero agradecérselo a Julia, por todos los sacrificios que ha hecho por apoyarme y por aguantar todas las penas y crisis que he tenido, en especial realizando este TFG.

También, quiero agradecérselo a mi tutor Dr. Ing. D. Miguel Rodelgo Lacruz, por su inmensurable paciencia hacia mis crisis de nerviosismo y tozudez. Es gracias a él y su atenta mentorización, que este trabajo ha salido adelante.

Por último, dado que este trabajo pone fin a mi estancia en la Escuela Naval Militar, quiero agradecer a todos mis amigos y compañeros de la promoción 422-152, junto a los cuales, he pasado maravillosos cinco años de mi vida, definitivos para mi formación moral y profesional. Únicamente gracias a ellos, recordaré esta etapa de mi carrera con nostalgia cuando esté destinado en la Flota. Quiero hacer una mención especial a mis amigos Jesús, Emilio y Paco, por ser mi apoyo del día a día. Ellos han hecho que saque lo mejor de mí estos dos últimos años en los que hemos vivido juntos, los cuales han sido los cimientos de una amistad para toda la vida.

CONTENIDO

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

1 INTRODUCCIÓN Y OBJETIVOS

1.1 Contexto y motivación

Es evidente e inequívoco que el mundo se encuentra en la cuarta revolución industrial o *Industria 4.0* (I4.0). La sociedad, y todo lo que la rodea, tiene una tendencia a una transformación sin precedentes, ya que la mayoría de empresas empiezan a incluir, entre otros, sofisticados sistemas software, potentes procesadores o novedosas tecnologías de comunicación.

La calidad de vida ha dado un salto significativo hacia el futuro debido a que, todo lo que rodea a la sociedad, está conectado entre sí (personas, datos, máquinas…) y ahora, la industria también. Ha surgido un nuevo concepto de consumo que implica nuevas formas de fabricar. Algunas de las técnicas colaborativas utilizadas para lograr este avance I4.0 son: robótica colaborativa, realidad aumentada, *Big Data*, impresión 3D, visión artificial, internet de las cosas (IoT) o inteligencia artificial.

Los países han aplicado distintas políticas en relación a la I4.0 con el objetivo de alcanzar el éxito de la aplicación a través de la productividad, calidad y eficiencia [1]. El coste de las nuevas y sofisticadas tecnologías ha ido disminuyendo en los últimos años y su accesibilidad ha posibilitado que cada vez más empresas se sumen a la transformación progresivamente. En España también se está llevando a cabo de forma gradual dicha trasformación industrial y tecnológica.

Las importantes compañías Accenture y Oxford Economics han realizado un estudio para comparar el grado de digitalización de los países. Para dictaminarlo, el estudio se ha basado en la digitalización del entorno de trabajo, el conocimiento de las personas, los equipos tecnológicos utilizados (infraestructuras) y la influencia socioeconómica. Una vez realizado el estudio, han publicado un gráfico (véase Figura 1-1) de dicho índice de digitalización.

En dicho gráfico, España se encuentra en la undécima posición del ranking global, aunque la transformación escalonada ha ocurrido de forma más tardía que en el resto de Europa debido a la lenta aplicación de las políticas adaptativas.

Figura 1-1 Índice de digitalización de países del año 2020 [1]

La Armada española, como en cualquier otra empresa, también se ha unido al cambio iniciando un proceso de transformación digital y creando la *Armada 4.0*. Este proceso se ha visto apoyado por el Centro de Sistemas y Tecnologías de Información y Comunicaciones de la Secretaría de Estado de Defensa (CESTIC). El principal precursor de este cambio fue el predecesor del actual Almirante Jefe del Estado Mayor de la Armada (AJEMA), el Almirante General Teodoro López Calderón (véase Figura 1-2), el cual, y junto al Ministerio de Defensa, creó la Oficina de Innovación de la Armada (OFINAR) y además, implementó el Plan de Acción del Ministerio de Defensa para la Transformación Digital [2] en octubre de 2020. Un objetivo a destacar en este plan es el de la "gestión corporativa inteligente del conocimiento, la información y los datos".

Figura 1-2 Almirante General Teodoro López Calderón [3]

En aplicación del análisis de información y datos, la Armada dispone del Centro de Operaciones y Vigilancia de Acción Marítima (COVAM), con base en Cartagena. Siguiendo las líneas del AJEMA en este ámbito, las nuevas tecnologías deben ser explotadas al máximo de manera eficiente y eficaz para contribuir a las operaciones *Maritime Situational Awareness (MSA)* o Conocimiento del Entorno Marítimo (CEM). Es por esa razón por la que, en octubre de 2019, el AJEMA propuso la apertura de un

proyecto de investigación llamado Conocimiento del Entorno Marítimo *Artificial Intelligence* (CEMAI) para ser realizado conjuntamente por el Centro Universitario de la Defensa (CUD) y el COVAM. El objetivo de este proyecto es la aplicación de la inteligencia artificial para solucionar problemas de información y detectar situaciones anómalas de algunas embarcaciones [4], así como la implicación de los futuros Oficiales de la Armada en investigaciones de nuevas tecnologías. En paralelo, se ha lanzado el proyecto, SIRENA (acrónimo de Sistema de Inteligencia artificial para el Reconocimiento del EntorNo mArítimo) como complemento al CEMAI. Este trabajo se encuadra dentro de ambos proyectos asociados.

La principal fuente de información en el entorno marítimo y en los proyectos mencionados es el *Automatic Identification System (AIS),* que permite un continuo flujo de datos (posicionamiento e información) en tiempo real de todo tipo de embarcaciones. El principal problema es el hecho de trabajar con una ingente cantidad de datos (*Big Data)* en tiempo real que presentan una gran dificultad para operar con ellos de forma manual. Se puede añadir también que existe una falta de calidad de datos proporcionados por el AIS. Un ejemplo es la gran cantidad de buques con número MMSI *(Maritime Mobile Service Identity)* erróneo [5]. La información más fiable que proporciona el AIS es la dinámica: la posición, el rumbo y la velocidad del buque. Por ello, se requiere de tecnologías novedosas con el propósito de procesar, fundamentalmente, esta información dinámica del flujo de datos AIS en tiempo real para detectar actividades ilegales de tráfico.

1.2 Objetivos

El objetivo principal del presente trabajo es identificar actividades sospechosas de los buques en tiempo real mediante datos AIS. Concretamente, se desarrollará un algoritmo que permita identificar los buques que cometan actividades sospechosas de tráfico ilegal, a partir de datos indirectos como la cinemática, zona de actividad, el tipo de barco o datos registrales. Para su evaluación se empleará un histórico de mensajes AIS existente en el CUD. También, se propondrá la manera de implementarlo en tiempo real con el objetivo de detectar una actividad sospechosa en el momento que se realiza.

Para conseguir este objetivo global, se plantean los siguientes objetivos específicos:

- Revisión del estado del arte de tratamientos de datos, herramientas de procesamiento y trabajos existentes en relación al objetivo principal y la importancia de su empleo para el Conocimiento del Entorno Marítimo en el contexto del *Big Data* marítimo.
- Desarrollo del algoritmo y selección de umbrales mediante análisis de los datos.
- Análisis de los resultados obtenidos para estudiar posibles mejoras.

Además, otros objetivos complementarios planteados son los siguientes:

- Familiarización con la manipulación de *Big Data* y con las herramientas disponibles para ello.
- Ampliar conocimientos de programación trabajando con el lenguaje Python.
- Exponer conocimientos adquiridos durante el grado en asignaturas como Informática para la ingeniería.
- Adquisición de las competencias requeridas en [6].

1.3 Estructura de la memoria

Una vez contextualizado el trabajo y habiendo establecido los objetivos, se describe la organización de la estructura de esta memoria:

- En el Capítulo 1 se contextualiza el ámbito en el que se desarrolla este trabajo, justificando los diferentes hechos que motivan el proyecto, y se establecen los objetivos a alcanzar.
- En el Capítulo 2 se realiza una revisión de estado del arte, describiendo los conceptos teóricos de relevancia necesarios para la comprensión de este trabajo, así como las diferentes herramientas y tecnologías necesarias para proceder con el desarrollo. Para ello, primero se introducen los conceptos de *Big Data*, el análisis de datos masivos y la inteligencia artificial aplicada a este ámbito. A continuación, se realiza una introducción al Conocimiento del Entorno Marítimo y se profundiza este concepto, concretando la importancia del COVAM en este campo como entidad competente de la Armada. Se describen también los conceptos de la mensajería AIS, que forma parte del *Big Data* marítimo, y de las actividades ilegales. Posteriormente, se describen las tecnologías involucradas en el trabajo, y se cierra el capítulo con un breve resumen de las conclusiones sobre trabajos de índole similar.
- En el Capítulo 3 se desarrolla el algoritmo seleccionando los umbrales y los campos que se añaden en el mismo, atendiendo a la revisión del capítulo anterior. A continuación, se muestran los resultados obtenidos y se analizan posibles mejoras.
- En el Capítulo 4 se realiza una reflexión sobre el desarrollo del TFG para obtener las conclusiones. Además, se presentan las posibles líneas futuras.
- Para cerrar la memoria del trabajo, se incluyen la bibliografía y siete anexos.

2 ESTADO DEL ARTE

2.1 Industria 4.0

Las exigencias procedentes de la industria 4.0 han hecho que se requieran tecnologías y planteamientos de trabajo que se adapten a la digitalización de todos los medios disponibles. Es por ello que surgen nuevos conceptos 4.0 que son necesarios para entender la adaptación a la I4.0. A continuación, se procede a desarrollar los conceptos generales y específicos necesarios para la comprensión y realización de este trabajo.

2.1.1 Big Data

Conforme fueron surgiendo nuevas tecnologías a principios del presente siglo, el crecimiento del volumen de datos utilizados se fue multiplicando. En este momento surgió el concepto de *Big Data* para referirse a la inmensa cantidad de datos generados que las aplicaciones de *Software* de procesamiento de datos, que tradicionalmente se venían usando, no son capaces de capturar, procesar y poner en valor en un tiempo razonable.

Cuando surgió el concepto de *Big Data* se asoció con un modelo de magnitudes para su comprensión profunda. Este modelo fue llamado modelo de las 3 V: Volumen, Velocidad y Variedad. Más tarde se fueron añadiendo magnitudes conforme fueron surgiendo para dar lugar a modelos más específicos: el modelo pasó a ser el modelo de las 5 V o de las 7 V. Todos los modelos son igual de válidos, ya que, explican a grandes rasgos el concepto de *Big Data*. A continuación se detalla el modelo más reciente llamado el modelo de las 10 V (véase Figura 2-1) [7]:

1. Volumen: es la cantidad masiva de datos generados que crecen con el tiempo. Las aplicaciones y arquitectura construida que los soportan, deben crecer paralelamente al *Big Data.*
2. Velocidad: se entiende como el elevado ritmo al que se reciben, se almacenan y se procesan los datos. Muchas tecnologías funcionan en tiempo real, por lo que, los datos requieren una evaluación y actuación inmediata.
3. Variedad: se refiere a la diversidad de datos procedentes de numerosas fuentes y que se encuentran en distinto formato.
4. Variabilidad: en un entorno tan turbulento como el del *Big Data*, la información varía conforme avanza el tiempo.
5. Veracidad: conocer la fiabilidad de la información recogida es importante para obtener datos de calidad. Puede dar una ventaja competitiva en la explotación de los datos.

6. Validez: responde a cuán limpios se encuentran los datos en relación con el uso que se les pretende dar. Una gran parte del proceso debe dedicarse a limpiar y considerar los datos antes de su análisis.
7. Vulnerabilidad: la seguridad de la información es de las magnitudes más relevantes y que son necesarias reforzar. Muy poca información del universo digital se encuentra protegida.
8. Volatilidad: existen muchos datos que, con el tiempo, caduca su validez. El almacenamiento y el posible procesado de estos datos causan una pérdida de eficiencia, por lo que, es necesario establecer el tiempo que tarda un de ficheros de datos en ser obsoleto.
9. Visualización: se refiere al uso de gráficos para interpretar la cantidad de datos que se pueden representar.
10. Valor: todas las magnitudes cobran sentido cuando se consiguen datos de valor que representan factores clave, que son particulares para cada objetivo.

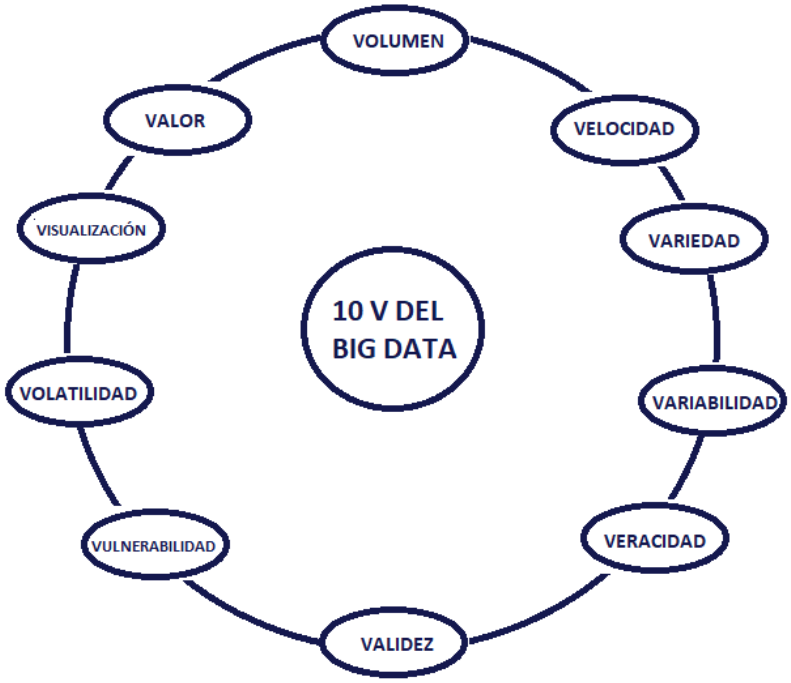

Figura 2-1 Las 10 V del Big Data

Otras nociones que deben conocerse al hablar de *Big Data* son los distintos tipos de estructura (véase Figura 2-2) que pueden presentar los datos con relación a la "Variedad" [8]:

- Datos estructurados: son los que tradicionalmente se han usado en el tratamiento de datos. Se encuentran ordenados y definidos en formato y longitud.
- Datos no estructurados: se trata de datos en su forma original que no poseen un formato definido. Carecen de valor si no se ordenan, identifican y almacenan de manera correcta.
- Datos semiestructurados: estructura ligeramente ordenada, pero que no es lo suficientemente regular para gestionarla con la misma facilidad que los datos estructurados. Posee ciertos patrones que relacionan la información. Es el caso de los formatos XML, JSON o HTML.

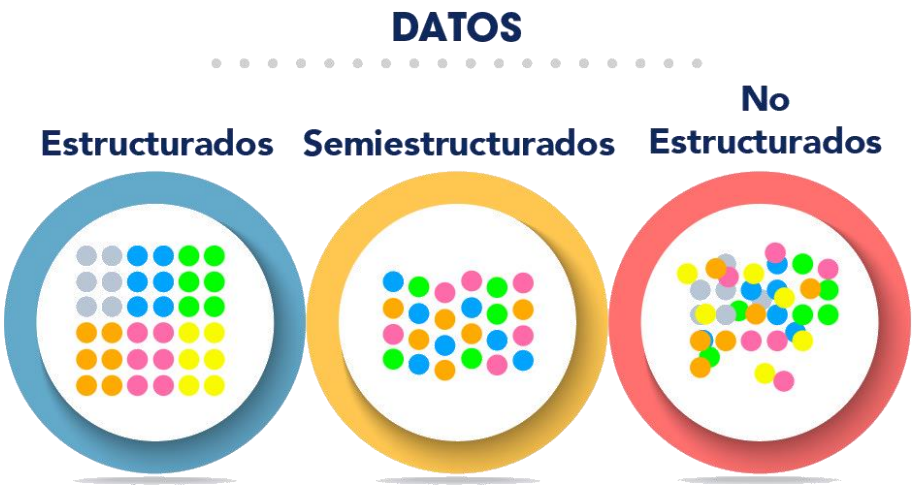

Figura 2-2 Tipos de estructuras de datos [9]

Una vez definidas las características principales y los tipos de datos que existen, se procede a explicar, de manera generalizada, las etapas de la gestión de los datos [8]:

1. Captación de la información: existen numerosos métodos y plataformas para capturar la información. Algunos ejemplos de los métodos más utilizados son: *Web Scraping* (técnica para extraer información de sitios web), gestión de información a través de aplicaciones informáticas creadas para ese propósito o servicios como *Apache Kafka* (específicamente diseñado para recopilar y agregar grandes volúmenes de datos).
2. Almacenamiento: una vez se ha captado la información, ha de ser almacenada. Existen diferentes opciones de almacenamiento dependiendo del uso que se le vaya a dar y al tipo de información. Algunos ejemplos son optar por elementos dispares como hojas de cálculo para información estructurada tradicional o sistemas NoSQL (que permiten el almacenamiento de información no estructura de forma flexible y rápida).
3. Tratamiento: el tratamiento de la información almacenada dependerá del uso y del tipo de información. Hay un amplio abanico de posibilidades, desde tratamientos sencillos a sistemas predictivos complejos. De manera generalizada, se puede extraer conocimiento y buscar patrones repetitivos de los datos a través de la estadística y aplicar el aprendizaje automático (con el objetivo de generalizar comportamientos en base a ejemplos que contrasten el entrenamiento).
4. Puesta en valor: sin un análisis y un tratamiento adecuado, los datos no garantizan conocimiento. El valor se encuentra en la relación entre los datos. Esta relación se encuentra en extrayendo patrones de comportamiento mediante inteligencia artificial.

2.1.2 Analítica de datos

El tratamiento y puesta en valor del *Big Data* lleva implícito el análisis completo mediante un proceso de descubrimiento que requiere conocimientos humanos e implementación tecnológica de aprendizaje automático (analítica inteligente). El objetivo principal de la analítica del *Big Data* es identificar patrones, tomar decisiones y predecir comportamientos. Sin embargo, no siempre se pueden usar los modelos estadísticos directamente.

En [10] se define como el análisis del *Big Data* a la combinación de sistemas de alta tecnología y de matemáticas que analizan la información y le dan un significado valioso. Algunas técnicas que se pueden utilizar para lograr este objetivo son: aprendizaje automático, analítica predictiva, análisis de textos, la minería de datos y métodos estadísticos; siendo estos dos últimos objeto de estudio de este trabajo.

La minería de datos es una técnica de análisis que se define como el proceso de detectar la información procesable del *Big Data* empleando análisis matemático para deducir los patrones y tendencias existentes. Estas deducciones se pueden recopilar y definir en un modelo de minería de datos que persigue los objetivos descritos del análisis de datos [11].

La estadística se basa en la predicción de una variable que se asocia a una observación. El objetivo de los métodos estadísticos en el *Big Data* es encontrar la relación matemática que relacione las predicciones y las respuestas de la forma más precisa posible [12].

Existen cuatro tipos de análisis de *Big Data*, los cuales con complementarios son inclusivos con respecto al anterior (véase Figura 2-3):

1. Análisis descriptivo: explica con la ayuda de gráficos o informes lo que ha sucedido, pero no lo que sucederá en el futuro.
2. Análisis de diagnóstico: muy unido al análisis descriptivo. Este tipo de análisis trata de interpretar los datos e intenta diagnosticar las causas de los resultados obtenidos.
3. Análisis predictivo: como su propio indica, analiza los datos para predecir lo que podría suceder. Se apoya en los dos análisis anteriores para detectar tendencias futuras.
4. Análisis prescriptivo: es una evolución del análisis predictivo. Basado en procesos de automatización que además de analizar y predecir, aconseja cómo proceder en el futuro recomendando acciones.

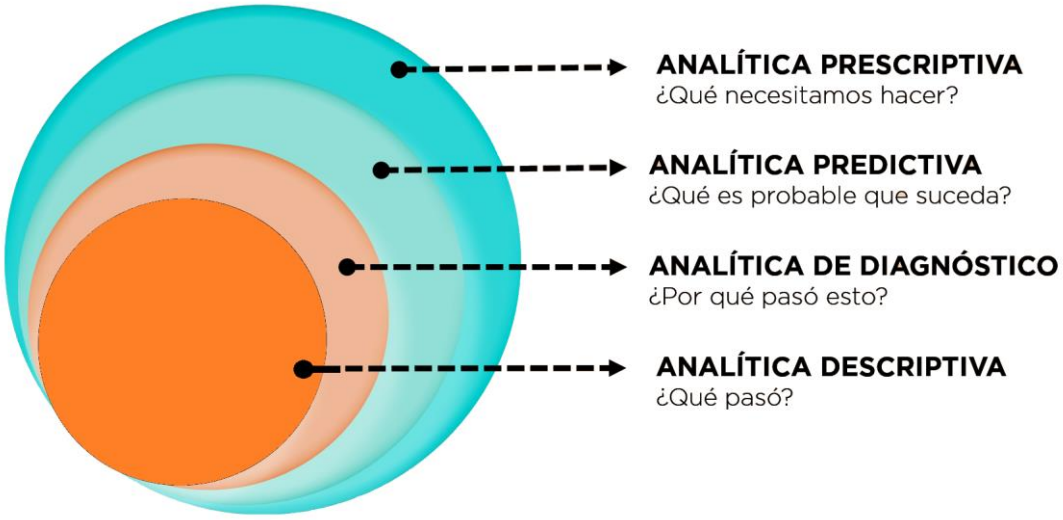

Figura 2-3 Tipos de análisis inclusivos [13]

La Inteligencia Artificial (IA) es la combinación de algoritmos creados con el objetivo de crear máquinas que imiten las mismas capacidades cognitivas del ser humano [14]. Los algoritmos, a través de la IA, se volverán más eficientes si se le proporcionan más datos.

Algunos de los principales beneficios que la IA proporciona son:

- Observación de patrones: la IA detecta patrones en los datos que, para los seres humanos, pueden pasar desapercibidos.
- Detección de anomalías: la IA analiza los datos y detecta datos anómalos y atípicos como aquellos que se desvían de la tendencia de datos.
- Predicciones: una vez reconocidos los patrones, la IA es capaz de predecir futuros resultados basados en estos patrones.

Además, existen unos subconjuntos dentro de la IA, que vienen dados intrínsecamente en su naturaleza. Son el aprendizaje automático o *machine learning* y el aprendizaje profundo o *deep learning* (véase Figura 2-4).

El aprendizaje automático es un subconjunto de la IA que utiliza métodos estadísticos para que, a partir de un resultado, generar un programa que haga que los resultados se cumplan, invirtiéndose así el paradigma de programar antes de obtener resultados; mientras que el aprendizaje profundo es un subconjunto dentro del aprendizaje automático que utiliza redes neuronales multicapa [15].

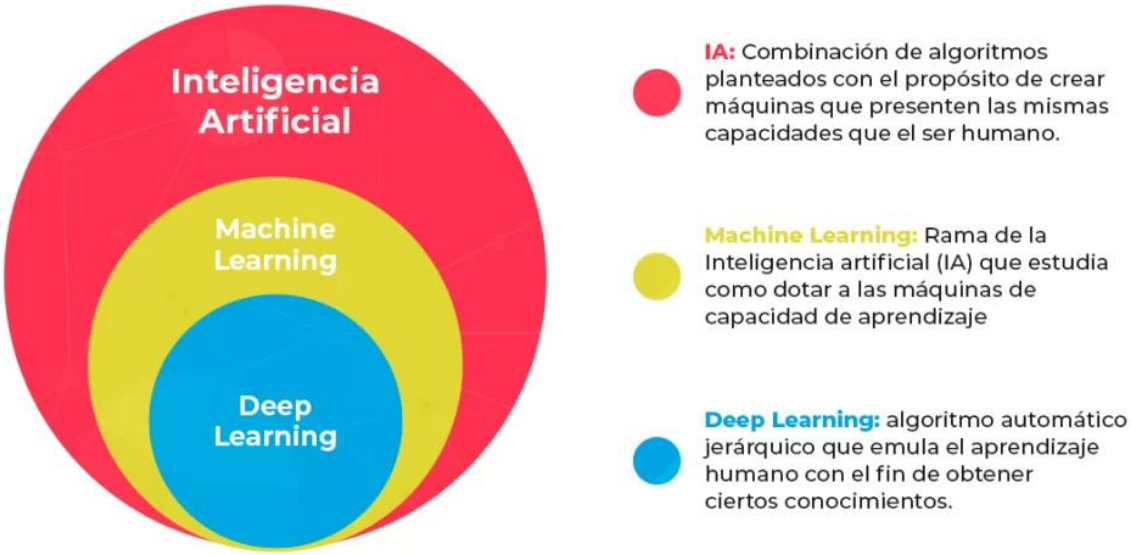

Figura 2-4 Inteligencia artificial, *Machine learning* y *Deep learning* [16]

Los tipos de implementación de aprendizaje automático pueden clasificarse en tres categorías [17]:

- Aprendizaje supervisado: se caracteriza por requerir de bases de datos ya etiquetadas para que las máquinas puedan generar una solución, es decir, el sistema informático es capaz de etiquetar nuevos datos basándose en un histórico de datos ya etiquetados.
- Aprendizaje no supervisado: se caracteriza por no requerir datos etiquetados. La técnica más usada es el *Clustering*, agrupa y segmenta datos en función de características similares.
- Aprendizaje reforzado: tiene como objetivo construir modelos optimizadores en base a resultados obtenidos anteriormente, utilizando redes neuronales multicapa.

La operativa habitual de la analítica de datos combina distintas técnicas. En la primera fase se emplea minería de datos para analizar la calidad de los mismos y extraer características relevantes que luego son transformadas o empleadas por métodos estadísticos para la detección de patrones o como paso previo al aprendizaje automático para que este, obtenga una buena precisión y exhaustividad.

Estas técnicas se pueden clasificar según el momento en el que se haga el procesamiento de datos [18]:

- Análisis *batch*: también llamado análisis offline. En este caso, el modelo se entrena accediendo a todos los datos disponibles. Si se reciben nuevos datos, el modelo debe ser reentrenado.
- Análisis online: el modelo se entrena o se procesa incrementalmente, actualizándose cada vez que se recibe una muestra (en tiempo real). Son sistemas adecuados para datos que cambian con rapidez y también cuando no es posible reentrenar el modelo o consultar los datos históricos con cada nueva muestra. Una alta tasa de aprendizaje, llevará a olvidar más

rápidamente los datos más antiguos y viceversa. Este tipo de análisis es de aplicable en el desarrollo de este trabajo.

2.2 Conocimiento del Entorno Marítimo (CEM)

El sector marítimo es uno de los principales motores de la industria. Este antiquísimo sector se ha ido desarrollando a lo largo de los siglos y, en la actualidad, cuenta con numerosas innovaciones que permiten la automatización de la creación de la imagen de la situación marítima en tiempo real. Es necesario tener una imagen precisa de lo que sucede en la mar ayudándose de estas innovadoras tecnologías.

Por otro lado, es fundamental para la seguridad del mercado mundial proteger el comercio marítimo de los ataques o la explotación por parte de los terroristas. La seguridad marítima tiene también una importante dimensión de defensa. Es impensable que un ejército convencional pueda sostener un conflicto armado sin la capacidad de transportar activos significativos mediante la mar. Por eso resulta vital disponer de una imagen de todos los aspectos de las actividades marítimas. Esta imagen ha de ser precisa, en tiempo real y seleccionada cuando sea necesario. Es por ello que a continuación se describe el concepto de CEM y sus principales componentes.

2.2.1 Concepto de CEM

El concepto de CEM surge en 2005 a partir de la Nueva Estrategia de Seguridad Marítima de los EE.UU. a raíz de los atentados del 11 de septiembre de 2001. En el *National Plan to Achieve Maritime Domain Awareness*, se define como "el conocimiento efectivo de todo lo asociado al dominio marítimo que podría afectar la protección, la seguridad, la economía o el medioambiente de los Estados Unidos" [19]. También, incluye que el CEM es el elemento fundamental que permite a los dirigentes de todos los niveles tomar decisiones eficaces y actuar con prontitud contra una amplia gama de amenazas a la seguridad de los EE.UU., sus intereses, aliados y amigos [20].

La Armada española define el CEM o *Maritime Situational Awareness* (MSA) como la acción de "fusionar y analizar la información que se recibe de un elevado número de fuentes, obteniendo una imagen precisa de todo lo que sucede en los espacios marítimos de interés nacional" [21]. En periodos de paz, tener un control marítimo real, contundente y efectivo es un factor disuasivo para cualquier agresor potencial a los intereses nacionales.

El CEM apoya a la industria marítima, a los gobiernos y a las organizaciones internacionales con técnicas de aprendizaje automático y de *Big Data,* analizando así los datos de tráfico de buques disponibles a través del AIS. Uno de los retos más importantes de la ampliación del MSA computacional a los regímenes de *Big Data* es la integración de los algoritmos centrales de aprendizaje con los modos de almacenamiento de *Big Data* y análisis de datos.

2.2.2 Centro de Operaciones y Vigilancia de Acción Marítima (COVAM)

El entorno marítimo de interés nacional se organizaba en Zonas Marítimas, Sectores Navales, Provincias Marítimas y Distritos Marítimos. Estas estructuras daban origen a la estructura territorial de la Armada española. En 2004 se crea la Fuerza de Acción Marítima (FAM) (véase Figura 2-5) suprimiendo las estructuras citadas. El objetivo y la razón de su creación es "concebir el espacio marítimo español de forma global como una entidad geoestratégica única, proporcionando la organización del conjunto de la Fuerza Naval sin sujeción a condicionantes territoriales y atendiendo exclusivamente a la función que desempeñan sus unidades, con independencia de su base de despliegue"[22]. La FAM está directamente relacionada con el CEM, de hecho, es su principal cometido.

Figura 2-5 Escudo de la Fuerza de Acción Marítima [23]

Uno de los principales órganos de este cometido es el Centro de Operaciones y Vigilancia de Acción Marítima (COVAM) que nace en 2005 de la mano de la FAM bajo el mando del Almirante de Acción Marítima (ALMART). La situación geográfica de España, rodeada de mar y siendo la puerta de acceso al Mar Mediterráneo, requiere un centro de operaciones centralizado que vigile y tenga una presentación de la situación marítima o *Recognized Maritime Picture* (RMP).

La misión principal del COVAM es "la ayuda de la conducción de las operaciones de los buques que integran la FAM, y de los buques transferidos al Mando de Vigilancia y Seguridad Marítima (MVSM)" [24]. También da apoyo a otras agencias nacionales "contribuyendo a la protección de los recursos naturales, del medioambiente, y de la contaminación marina, la vigilancia de pesca, la lucha contra la inmigración ilegal, el tráfico ilícito de estupefacientes y el contrabando, la vigilancia y protección del patrimonio arqueológico subacuático, la cooperación en búsqueda, salvamento y rescate o el apoyo en tareas de rescate de submarinos". El producto final en tiempo real que produce el COVAM, tras analizar y conocer de manera efectiva todo lo asociado al dominio marítimo en aguas de interés para España como las aguas territoriales, la Zona Económica Exclusiva (ZEE), el Índico, el Golfo de Guinea o el Mar Negro, es la ya citada RMP. El COVAM pone la RMP a disposición de todos los buques de la Armada y de las agencias nacionales que la soliciten [21].

El COVAM está situado en Cartagena, en el edificio de Capitanía General. Dispone de medios de Mando y Control nacionales, de la Unión Europea y de la OTAN necesarios para coordinar las operaciones de Seguridad Marítima. Su plantilla, compuesta por 30 personas de la Armada, está siempre disponible, las 24 horas del día y los 365 días del año. También permite la participación física de personal de otros organismos de la Administración Marítima en el proceso de fusionar y analizar los datos.

La sala principal del COVAM (véase Figura 2-6), cuenta con aplicaciones informáticas que permiten fusionar toda la información recibida para analizarla y evaluarla. Para integrar toda la información en tiempo real, dispone del Sistema Integrado de Vigilancia y Conocimiento del Entorno Marítimo (SIVICEMAR), que permite analizar el comportamiento de los buques que navegan por las zonas de interés nacional y detecta comportamientos anómalos [25]. Concretamente, este sistema permite el seguimiento, en aguas territoriales españolas, de todas las embarcaciones de uso civil, de las Fuerzas y Cuerpos de Seguridad del Estado y de las militares.

Figura 2-6 Sala de situación del COVAM [24]

Los medios que dispone el COVAM, tanto de personal como materiales, están distribuidos de forma permanente en tres "capas" relacionadas, pero diferenciadas por el nivel de seguridad que manejan [26]:

1. En la primera capa se agrupa la información sin clasificación de seguridad y está diseñada para su obtención, fusión, análisis y distribución.
2. En la segunda se agrupa toda la información que es sensible por intereses industriales, comerciales o mediáticos, aunque no tenga clasificación de seguridad.
3. En la tercera capa se agrupa aquella información clasificada y de acceso únicamente militar.

Asimismo, la Armada española ha puesto en marcha, en noviembre de 2021, un programa de modernización y mejoras técnicas, operativas de formación y de infraestructura del COVAM, incluyendo mejoras de estandarización y normalización. La aplicación de las mejoras se estima en un plazo de 14 meses. Las principales actuaciones de modernización serán [27]:

- El desarrollo de Software a medida a partir de productos *Commercial Off-The-Shelf* (COTS)/*Government Off-The-Shelf* (GOTS).
- La arquitectura de red.
- Adquisición de productos Hardware y Software COTS.
- Hardware a medida: adquisición de cajas fusionadoras de transmisión de datos
- Contratación de trabajos de infraestructura

La principal fuente de información de la que se vale el COVAM es el *Automatic Identification System* (AIS). El correcto funcionamiento y análisis de este sistema es de vital importancia para levantar la presentación de la situación marítima (RMP) con claridad y fiabilidad. Otras fuentes de información son los propios buques de la Armada española que se encuentran navegando que informan al COVAM de la situación marítima a través del Entorno Colaborativo Marítimo de la Armada (ENCOMAR); también son fuentes de información la base de datos registral de los buques, los datos meteorológicos y el análisis de fuentes abiertas (OSINT), entre otras.

2.2.3 Automatic Identification System (AIS)

La organización Marítima Internacional (OMI) exige, a través del convenio *Safety Of Life At Sea* (SOLAS) (Capítulo V – Regla 19) [28], el uso de AIS a buques, que realicen viajes internacionales, de

más de 300 toneladas. No obstante, se recomienda su uso en todos los casos y es obligatorio también para los buques de pesca en atención a las atribuciones delegadas en las Comunidades autónomas [29].

Uno de los objetivos principales del presente trabajo es definir anomalías en base a comportamientos de los buques basándose en datos AIS, por lo que la importancia del estudio de estos datos y mensajes resulta evidente, ya que posteriormente se trataran estos datos con el objetivo de identificar parámetros (en tiempo real) que sean indicadores de estas anomalías. Además, para cumplir con dicho objetivo, se utiliza una base de datos enriquecida, con campos adicionales, por lo que a continuación se procede a explicar los aspectos básicos de este sistema y su relación con el trabajo.

2.2.3.1 Descripción del AIS

El AIS es una ayuda a la navegación, de vital importancia dada su funcionalidad y la gran cantidad de información que proporciona. Trata de resolver la dificultad de identificar a los buques, especialmente cuando no están a la vista (de noche, sectores ciegos del radar de navegación o a distancia larga) proporcionando datos básicos (véase Tabla 2-1) como el nombre, el número de identificación de llamada (MMSI), la posición, el rumbo, la velocidad, el destino, la actividad que realiza (*Navigation status*), el tipo de buque, sus dimensiones y otros datos complementarios. Estos datos son enviados en forma de mensaje por VHF o incluso por satélite (menos común).

Campo	Descripción
MMSI	Número MMSI del buque (identificador AIS)
TIME STAND	Fecha y hora (UTC) de la emisión
LATITUDE	Latitud geográfica (WGS84)
LONGITUDE	Longitud geográfica (WGS84)
COG	Rumbo sobre fondo (grados)
SOG	Velocidad sobre fondo (nudos)
HEADING	Rumbo del buque (grados). Un valor de 511 indica que no hay datos del rumbo
NAVIGATION STATUS	Estado de navegación de acuerdo con especificación AIS
IMO	Número IMO del buque
NAME	Nombre del buque
CALL SIGN	Distintivo de llamada del buque
TYPE	Tipo de buque de acuerdo con especificación AIS
A	Distancia en metros de la antena a la proa del buque
B	Distancia en metros de la antena a la popa del buque
C	Distancia en metros de la antena al costado de babor del buque
D	Distancia en metros de la antena al costado de estribor del buque
DRAUGHT	Calado del buque en metros
DEST	Puerto de destino
ETA	Tiempo estimado de llegada en formato completo fecha/hora
SRC	Fuente (Source) de los datos AIS: Terrestre (TER) o Satélite (SAT)
ZONE	Nombre de la zona donde se encuentra el buque
ECA	Indica si el barco se encuentra en zona ECA/SECA (Zona de control de emisiones)

Tabla 2-1 Campos de un mensaje AIS [30]

Un parámetro interesante es el MMSI, que se emplea como un identificador del buque que es único y específico para cada barco. Por lo que un MMSI duplicado se considera un error o anomalía.

Los datos del buque se muestran en un *chartplotter* o incluso en un software de navegación con capacidad de datos AIS. Aparte de toda la información que transmite, el AIS también hace cálculos cinemáticos para evitar el abordaje como la distancia mínima de paso (CPA) o el tiempo hasta la misma (TCPA).

Las principales aplicaciones del AIS son:

- Ser utilizado por las autoridades marítimas como fuente de información.
- Dar apoyo como herramienta complementaria a dispositivos de control del tráfico marítimo o *Vessel Traffic Service* (VTS) y a tareas de salvamento y rescate (SAR).
- Complementar el radar, pudiendo trabajar de manera conjunta e integrada.
- Evitar los abordajes en la mar facilitando intercambio de información (véase Figura 2-7).
- Identificación de balizas y marcas de navegación

Es importante destacar que el AIS no debe confundirse con un radar a la hora de evitar un abordaje, ya que no proporciona un crudo radar como tal. Y siempre ha de ser un complemento del radar, nunca un sustituto [29].

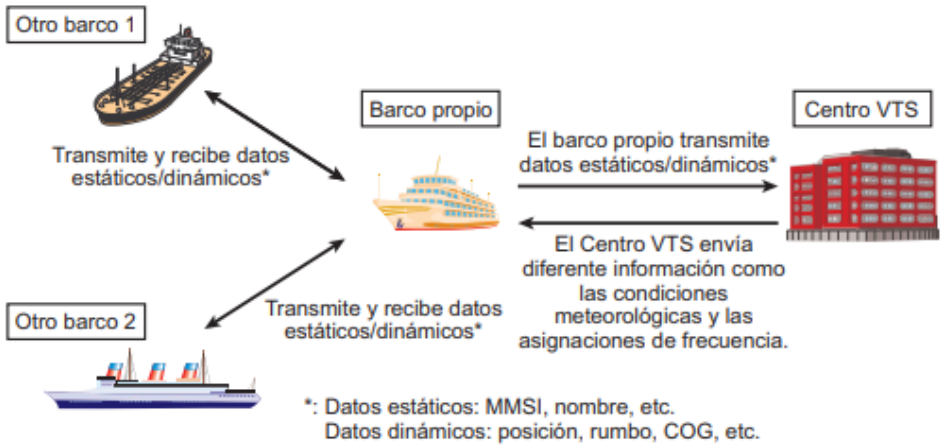

Figura 2-7 Flujo de información AIS [31]

Los datos e información que transmite el AIS, se clasifican en cuatro grupos:

- Información estática: indica datos básicos del buque que no cambian normalmente con el tiempo, tan solo cambia cuando se modifica alguna de las características principales del buque. Estos datos son el nombre, MMSI, distintivo de llamada, IMO, dimensiones y tipo de barco. Se emite cada seis minutos.
- Información dinámica: indica los datos importantes para la navegación relativos a la posición del barco: posición, rumbo efectivo, rumbo verdadero y velocidad. Se emite en un intervalo de 2 a 10 segundos.
- Información relativa al viaje: son los datos que se introducen manualmente: destino, ETA, si lleva carga peligrosa, su plan de ruta y el estado de navegación. Esta información se puede suprimir por motivos de seguridad. Se emite cada seis minutos.
- Textos y mensajes: mensajes relativos a la seguridad para alertar de peligros.

2.2.3.2 Calidad datos AIS

Anteriormente mencionado, el principal problema de los datos AIS es la falta de calidad y fiabilidad. A continuación se procede a clasificar los datos según su fiabilidad en base a la comparación y contraste de trabajos previos.

Los autores Frank Heymann, Thoralf Noack y Pawel Banyś, del Centro Aeroespacial alemán eV, han realizado un análisis de plausibilidad de los parámetros AIS relacionado con la navegación basado en series temporales [32]. Los resultados expuestos en dicho trabajo ponen de manifiesto que la información dinámica es altamente fiable:

- La posición recibida de los barcos es el parámetro más fiable, ya que, el satélite GPS contiene un error de pocas decenas de yardas. Por lo que la información adicional dinámica se contrasta con la calculada por la posición (véase Figura 2-8).
- El análisis detallado del COG (*Course Over Ground*), campo dinámico de un mensaje AIS, muestra que el 95% de los valores recibidos son fiables, con un error de ±3 grados.
- De la misma manera que el SOG (*Speed Over Ground*) muestra también que el 95% de los datos AIS recibidos son fiables, con error de ±0.3 nudos.

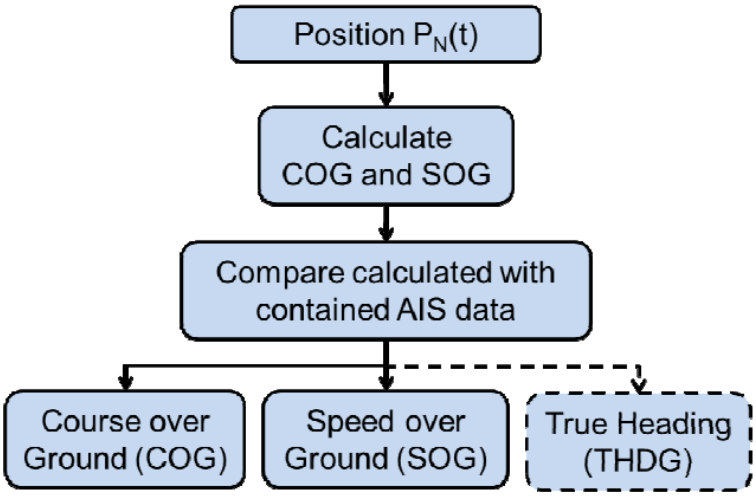

Figura 2-8 Estrategia del análisis de plausibilidad de [32]

Los autores Ties Emmens, Chintan Amrit, Asad Abdi y Mayukh Ghosh, de las universidades de Amsterdam y Twente (Países Bajos), han realizado un estudio de las promesas y los peligros de los datos AIS [33]. En dicho trabajo describen numerosas veces un término, el ruido, para definir la cantidad de información sobreabundante que dificulta el procesamiento de la información realmente válida y fiable. Además, se estudia la información que raramente es fiable, la información relativa al viaje e información estática:

- El 35% de los buques no actualizan el puerto de destino durante su derrota. Porcentaje muy elevado para considerar fiable este parámetro. Además, el puerto de destino, es un campo que se introduce manualmente (escribiendo) al no existir una lista estandarizada de puertos. Como consecuencia, el análisis automatizado de este campo se dificulta.
- Además, el 53% de las transmisiones, carecen de ETA (tiempo estimado de llegada en formato completo fecha/hora)
- El 30% de los buques transmiten un valor nulo del calado.

Como conclusión del trabajo, destaca que los resultados estáticos y relacionados con el viaje contienen porcentajes elevados de ruido (información no válida y no fiable). La principal causa es que son datos que han de ser introducidos manualmente, a pesar de que según regula la OMI: "Es trabajo del oficial de guardia: antes de la salida o de la llegada, preparar el puente".

Como consecuencia de esta clasificación, para el desarrollo del presente trabajo, la información dinámica se considera fiable, poniendo en segundo lugar la información estática y relativa al viaje.

2.2.4 Big Data Marítimo

La base de datos con la se trabaja para la realización de este trabajo está compuesta por mensajes AIS enriquecidos con campos añadidos (Anexo I: Campos de la base de datos). Los datos AIS forman parte del *Big Data* Marítimo (véase Figura 2-9) que atienden al modelo de Las 10 V del Big Data. En la línea de lo establecido en [34]:

- El Volumen de los datos marítimos tienen un crecimiento imparable. Aplicaciones como *MarineTraffic* utilizan bases de datos con 18 millones de entradas mensuales en relación a los buques y puertos, y 800 millones de posiciones de buques registradas mensualmente.
- Los centros de operaciones y vigilancia marítima, como el COVAM, requieren información en tiempo real para generar mapas en vivo (RMP). Esta información requiere un procesamiento inmediato: requiere Velocidad.
- El *Big Data* marítimo tiene intrínseca la Variedad, ya que, los datos se recogen mediante decenas de dispositivos diferentes y en distintos formatos. Algunos ejemplos son las fuentes de datos alternativos como bases de datos registrales (aplicadas en el presente trabajo), OSINT, información meteorológica, etc.
- Es posible que se notifiquen datos que no sean coherentes y que existan ambigüedades entre las fuentes. Para lograr la Veracidad del *Big Data* Marítimo se necesitan métodos de evaluación de la calidad como la deduplicación, la desambiguación y procesos de limpieza de datos específicos.
- Los inmensos datos marítimos pueden utilizarse para el desarrollo de grandes aplicaciones; sin embargo, estos datos deben cobrar un Valor para que cobren sentido. Los patrones de las rutas y trayectorias de los buques pueden permitir la detección de anomalías y aumentar la seguridad marítima.

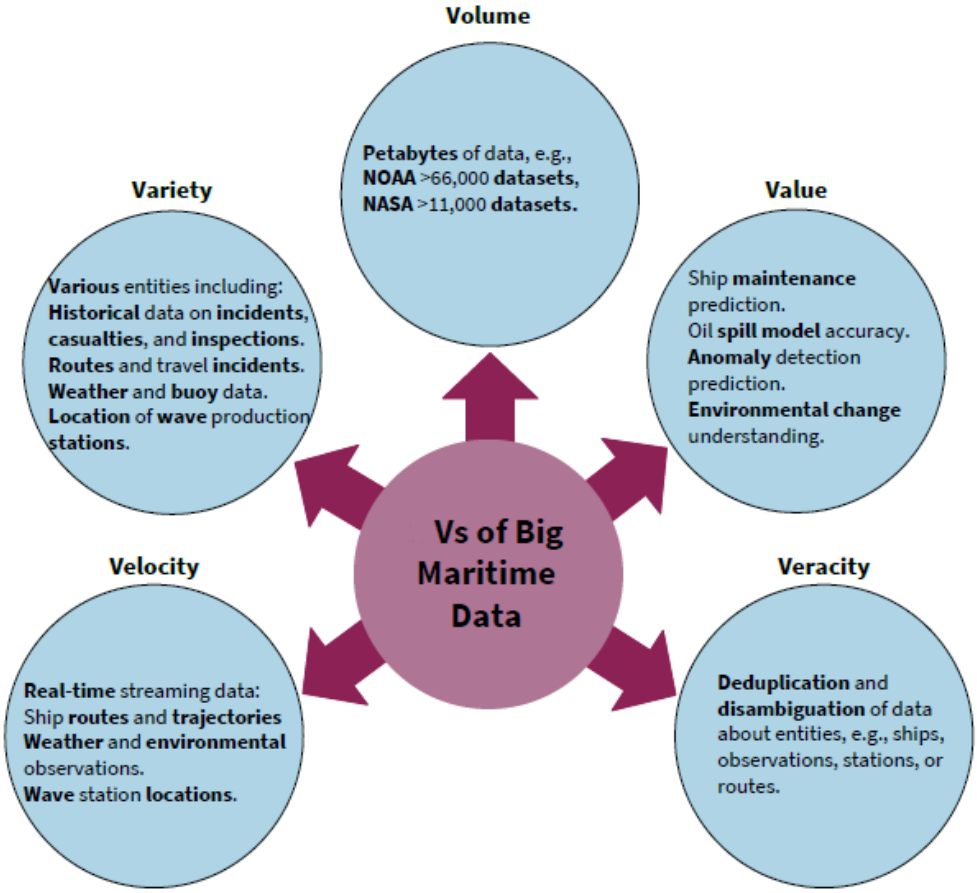

Figura 2-9 Las V del *Big Data* Marítimo [34]

Debido a que los datos del *Big Data* marítimo cambian con rapidez y debido a su extenso volumen (decenas de millones al día) que hace que no sea posible consultar los datos históricos para cada mensaje recibido, se decide que los algoritmos que se desarrollen, deben ser en tiempo real, con análisis online.

2.2.5 Actividades ilegales

La mayoría del tráfico ilegal en España, a excepción de las pateras y las lanchas semirrígidas en el STROG, se produce a través de barcos mercantes, la mayoría con cambio de bandera. Cualquier sustancia o producto regulado, puede ser susceptible de ser traficado.

Las sustancias estupefacientes o drogas son partícipes del mayor tráfico ilegal del mundo y es por ello, que se requiere un control estricto de mercancía. Es de gran interés nacional detectar e interceptar cualquier tipo de actividad sospechosa que se relacione con este tráfico. El 70% de la droga que tiene destino en Europa es transportada por vía marítima (véase Figura 2-10) y generalmente por buques mercantes desde puertos como Venezuela o Bolivia.

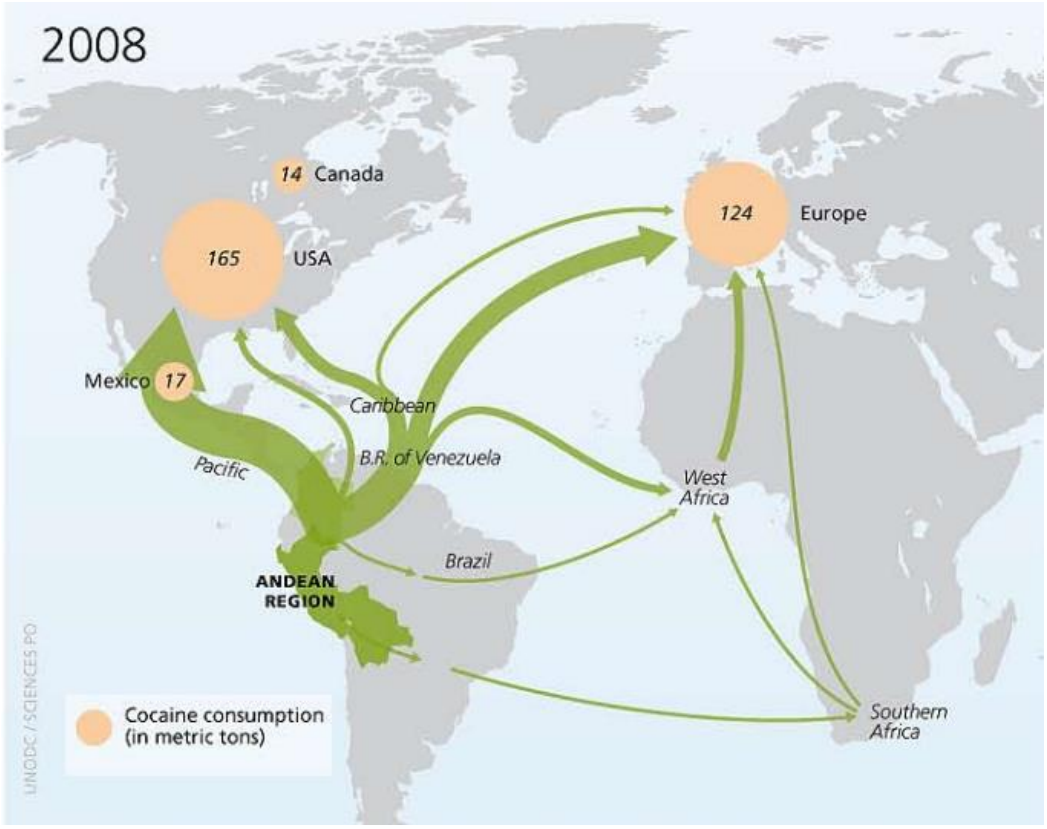

Figura 2-10 Mapa mundial del tráfico de cocaína en 2008 [35]

Existen diferentes métodos para la importación ilegal de estas sustancias, dependiendo del volumen del alijo. Sin embargo, es sabido que uno de los métodos más utilizados es transportar la droga oculta entre la carga legal de un mercante (véase Figura 2-12) u oculta en compartimentos de difícil acceso del buque, para descargar la droga cerca de la costa en planeadoras o embarcaciones rápidas que navegan hasta las playas: "Un método es transportar la droga oculta entre la carga legal o en compartimentos de difícil acceso del buque y descargar la droga cerca de la costa en planeadoras o embarcaciones rápidas que navegarán hasta las playas, donde la droga será cargada en vehículos hacia algún almacén." Con la aplicación de este método, los narcotraficantes evitan ser detectados en los controles aduaneros de mercantes de la mercancía [36].

El método usado por los narcotraficantes, recientemente explicado, se puede aplicar al tráfico ilegal de cualquier sustancia. Por lo que uno de los patrones más característicos de anomalías en un mercante es que salga de la ruta convencional de tráfico y realice patrones de movimientos que no son habituales en este tipo de buques.

Figura 2-11 Fotografías de incautación de cocaína oculta en falsos plátanos en Algeciras [37]

CONFIDENCIAL

CONFIDENCIAL

2.3 Arquitectura y tecnologías involucradas

En el presente trabajo se utilizan numerosas tecnologías de apoyo para lograr los objetivos. Es por ello que, a continuación, se describe la arquitectura del sistema y las principales tecnologías que se han involucrado en el desarrollo del trabajo.

2.3.1 Arquitectura del sistema

El sistema del proyecto (véase Figura 2-13) utiliza *Apache Kafka* y la librería de *Java, AISLib,* para la ingesta de los datos AIS. Los datos se procesan en tiempo real en *Java* que almacena cierta información de estado en una base de datos *Redis*. Las etapas de este procesado son:

- *Parsing & Selection:* procesado básico y selección de las tramas. Se extrae la información útil.
- *Filtering:* eliminación de duplicados y tramas incorrectas. Es un tipo de procesado específico del tipo.
- *Tagging:* etiquetado de eventos AIS con información extra (datos registrales como cambios de bandera, zona en la que se encuentra, etc.). Se realiza con la ayuda de *Redis* (almacenamiento en memoria de una base de datos con información de estado de los datos).
- *Logging:* registro de salida a fichero.

Los datos se almacenan ya filtrados y enriquecidos en *Elastic* para consultarlos con un cuadro de mando con *Kibana*.

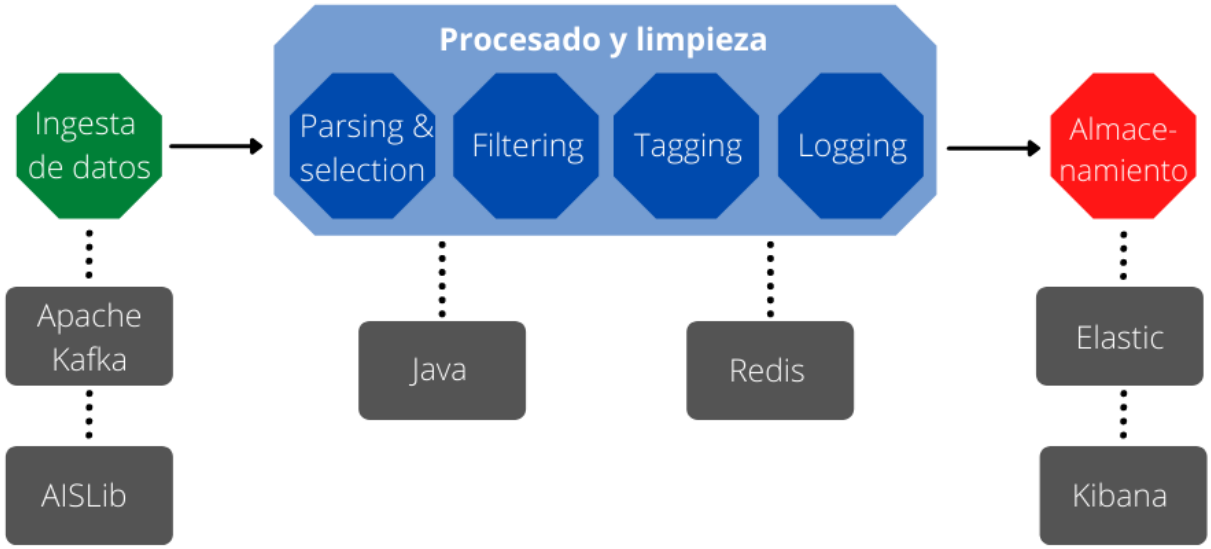

Figura 2-13 Arquitectura del sistema

Para el desarrollo del presente trabajo, se parte del conjunto de datos filtrados y enriquecidos que se encuentran almacenados en *Elastic* (dataset) de manera offline, pero limitando el análisis a técnicas online, de forma que el algoritmo, una vez validado, pueda ser implementado en la etapa de procesado en tiempo real. Las consultas requeridas deben ser implementables a *Redis* o una base de datos de capacidad moderada, pero no es viable que se hagan consultas a *Elastic* con cada muestra debido al tiempo que conllevaría y al gran volumen de muestras. Por eso es necesario que no se hagan consultas históricas para cada barco si se requiere que sea en tiempo real.

Para el análisis se usará Python y sus librerías, apoyándose en Kepler GL para la visualización de los resultados espaciales.

2.3.2 Motor de búsqueda Elasticsearch

Elasticsearch es una herramienta de código abierto utilizada como un motor de búsqueda y análisis de grandes volúmenes de datos. Se entiende búsqueda elástica como la capacidad de escalar los datos de manera ágil y centralizada [38]. Cuenta con una gran variedad de clientes de conexión y se adapta a sus necesidades como con plataformas como Python.

Una de las funciones interesantes de esta herramienta, que es de provecho para este trabajo, es que permite realizar consultas complejas de datos (incluyendo filtros, ordenaciones y agrupaciones). La búsqueda se ajusta a los requerimientos del usuario y es de gran utilidad para un análisis ágil y eficaz, concretamente es usado en el presente trabajo para consultar en una gran base de datos y almacenarlos a través de Pandas en un dataframe.

Para almacenar datos también se usa *Redis (Remote Dictionary Server)*, que es un almacén de datos en memoria de código abierto que se utiliza como base de datos, caché, agente de mensajes y cola. La característica principal de *Redis* es que es muy rápido, ofrece tiempos de respuesta inferiores al milisegundo, lo que permite que se realicen millones de solicitudes por segundo para aplicaciones en tiempo real. *Redis* es una opción muy habitual en aplicaciones de almacenamiento en caché, análisis en tiempo real o datos geoespaciales [39].

2.3.3 Conjunto de datos disponible

El conjunto de datos disponible contiene todos los mensajes AIS transmitidos en zonas de interés nacional durante 15 días: desde el 18 de mayo del 2021 hasta el 1 de junio del 2021. Los 55 campos de la base de datos utilizada (Anexo I: Campos de la base de datos) difieren de los de un mensaje AIS

(Campos de un mensaje AIS [30]) que cuenta con 22 campos. Las modificaciones se realizan para reducir el tamaño y hacer que las consultas sean más eficientes y en tiempo real: se han omitido los campos redundantes que, a priori, carecen de valor y se han añadido campos que dan más valor a los datos.

A continuación se describen los campos que se utilizan en el presente trabajo y que son no comunes a los de un mensaje AIS convencional, que anteriormente se han explicado (Campos de un mensaje AIS [30])

Se ha realizado una tabla de los campos relevantes no comunes a los mensajes AIS (véase Tabla 2-2), incluyendo una breve descripción de los mismos. Posteriormente, se describen más detalladamente para hacer posible la comprensión de la base de datos utilizada.

Campo	Descripción
CELDAS H3	Identificador de celda que utiliza un sistema de cuadrícula
MACRO	Indica la zona en la que es transmitido el mensaje AIS
ES_*tipo de barco*	Booleano de confirmación de tipo de buque
CAMBIOS DE BANDERA	Booleano que indica si un buque ha cambiado de bandera
PARADO	Booleano que indica si un buque está parado
DETRO ZEE	Booleano que indica si un buque está dentro de la Zona Económica Exclusiva
PROMEDIADO COG W1	Valor precalculado que indica el suavizado exponencial de la variación del COG

Tabla 2-2 Descripción de los campos novedosos y relevantes de la base de datos

Los campos "celda H3" corresponden a un identificador de celda que utiliza un sistema de cuadrícula, desarrollado por Uber, ideal para analizar grandes conjuntos de datos espaciales, dividiendo la Tierra en celdas de cuadrículas identificables (véase Figura 2-14) [40]. H3 define sus índices mediante el formato hexadecimal (16 bits). Existen diferentes resoluciones de las divisiones, en la base de datos se han almacenado las celdas de tres resoluciones diferentes: H3_9 (0.1905 km²), H3_7 (5.1612 km²) y H3_6 (36.1291 km²). Estos campos se usarán también para la representación geoespacial de los datos a través de Kepler GL.

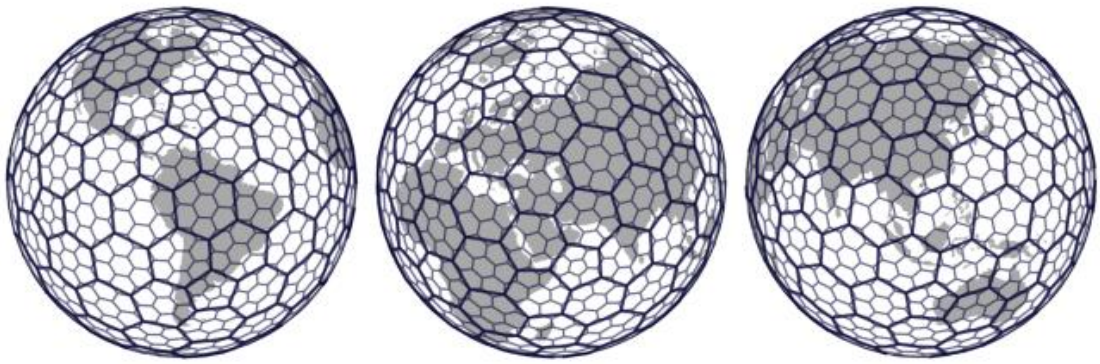

Figura 2-14 División del globo en celdas H3 [40]

El campo "macro" indica la zona en la que es transmitido el mensaje AIS correspondiente. Las opciones disponibles se corresponden con las zonas de interés nacional, incluyendo mar territorial, ZEE (véase Figura 2-15) y otras zonas de interés:

- El Océano Atlántico y Mar Cantábrico.

- Las Islas Canarias.
- Estrecho de Gibraltar.
- El golfo de Adén.
- El golfo de Guinea.
- El Mar Negro.
- El Mar Rojo.
- El Mar Mediterráneo.
- El Mediterráneo Oriental.

Este campo es de gran ayuda para filtrar los datos según su zona de acción. Existen comportamientos característicos de determinadas zonas. Por ejemplo, una zona muy interesante es el Estrecho de Gibraltar (STROG), ya que, se produce la combinación de barcos que acceden o salen en tránsito del Mediterráneo, con zonas de pesca muy concurridas y además aguas territoriales extranjeras, tanto de Marruecos como de Gibraltar.

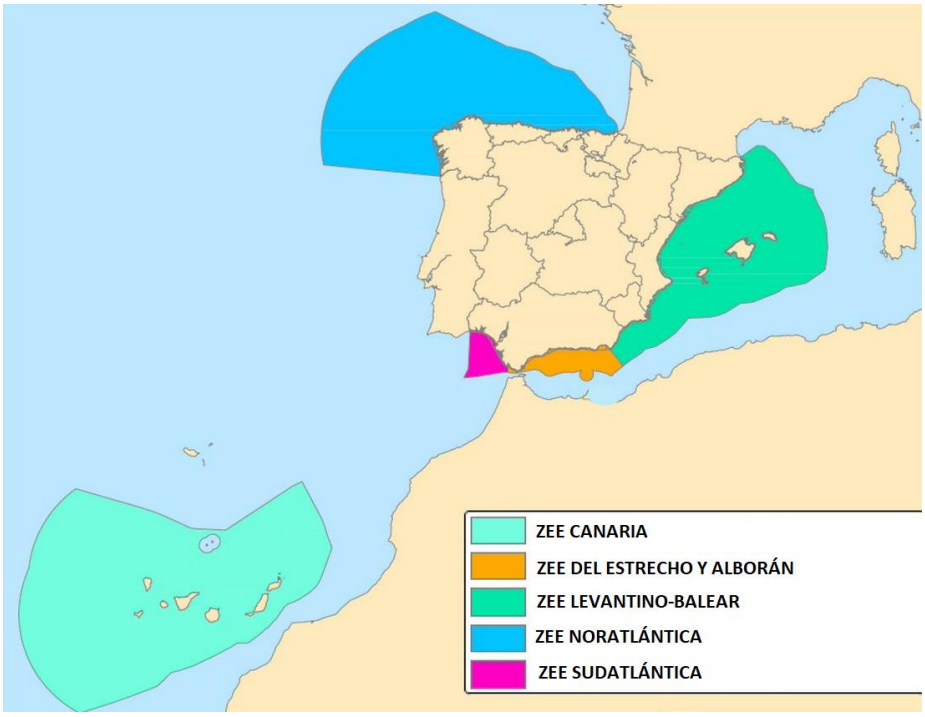

Figura 2-15 ZZEE de España [41]

Otro campo interesante es "ES_(*tipo de barco*)" que indica si es, o no, el tipo de barco del campo. El motivo de la adición de este campo es la necesidad de contrastar el campo tipo_buque_AIS. A excepción de los demás, el último campo en vez de indicar el tipo de barco indica si es, o no, un barco de pabellón español. La base de datos contiene 5 booleanos de esta índole:

- es_pesquero.
- es_carguero.
- es_militar.
- es_recreo.
- es_español.

El campo "cambio de bandera" (booleano) indica si un barco ha cambiado de bandera desde su registro. A priori, puede parecer un dato que carece de valor, pero es un indicativo de actividad sospechosa, ya que, cuando un barco cambia de bandera, lo suele hacer por temas fiscales.

"Parado" (boolenao) es un campo cuya utilidad es evitar realizar consultas pesadas cuando se requiere consultar si está parado en movimiento. Un barco se considera parado, en este campo, cuando su velocidad es menor de dos nudos, ya que, por el efecto de las corrientes y viento, un barco puede abatir y, estando parado, tener una velocidad de 1 nudo. Si se considera parado únicamente con 0 nudos, saldrían resultados atípicos. Por ello, más adelante, se analiza cuál es umbral de velocidad para determinar que un barco está parado.

De igual manera, el campo "Dentro ZEE" es útil para evitar realizar búsquedas pesadas, agilizando el proceso y delimitando la búsqueda a la zona de interés nacional.

El campo "promediado cog" es un valor calculado en la etapa de procesado en tiempo real. Su objetivo es ver cuánto ha variado el COG (la diferencia entre muestras) y promediar este valor. Para evitar realizar consultas inviables en tiempo real, en lugar de hacer un promediado convencional se calcula una media móvil exponencial considerando el tiempo que ha pasado desde la última muestra. De esta forma, un valor de, por ejemplo, 50 indica que el barco ha variado el rumbo aproximadamente (no exactamente, debido a que la media es exponencial) 50 grados en las últimas 6 horas, lo que hace que sea un campo muy interesante, ya que, si un barco en tránsito realiza muchos cambios de rumbo, podría ser un indicativo de anomalías y actividades sospechosas, como se analiza más adelante.

2.3.4 Python

Python es un lenguaje de programación interpretado, multiparadigma y multiplataforma que está desarrollado bajo una licencia de código abierto, de libre uso y distribución. Se caracteriza por ser un lenguaje de programación simple y ágil, que permite la programación orientada a objetos, programación funcional e imperativa. Además, Python (véase Figura 2-16) es de tipado dinámico, ya que, una variable puede tomar valores distintos si así lo desea el programador.

Figura 2-16 Logotipo Python [42]

Principalmente, es usado en *Big Data*, Inteligencia artificial, Data Science, Frameworks de pruebas y desarrollo web. Incluye una extensa biblioteca que ofrece una amplia gama de aplicaciones [43]. Es el recurso principal para desarrollo de herramientas que requieran análisis, tratado y procesamiento de datos (véase Figura 2-17). Además, junto con JAVA, es el lenguaje utilizado en trabajos del proyecto CEMAI/SIRENA que precenden al presente. Es por ello que se emplea dicho lenguaje para el desarrollo de este trabajo.

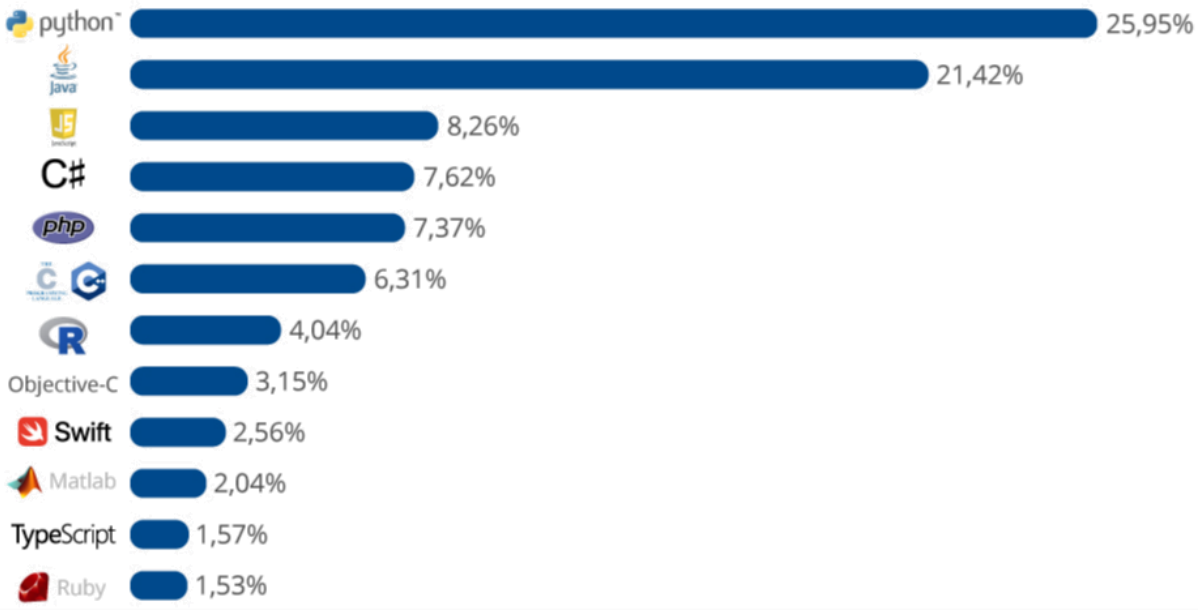

Figura 2-17 Porcentaje del uso de lenguajes de programación más populares del mundo [44]

Algunas de las características que aporta Python para el análisis de datos son [45]:

- Multifunción: permite analizar datos en cualquier tipo de entorno.
- Compatibilidad: Python es altamente compatible con otras plataformas de todo tipo.
- Tiene una rápida integración con aplicaciones con bases de datos no relacional.

Como se ha descrito anteriormente, Python contiene numerosas librerías utilizadas como herramientas de apoyo. Por lo que a continuación se describen las librerías más empleadas.

2.3.4.1 Librería NumPy

NumPy (*Numerical Python*) es una librería de Pyhton que es usada para aplicar computación científica. Está especializada en tratar grandes volúmenes de datos mediante el cálculo numérico y análisis de los mismos. Contiene una herramienta llamada *arrays* que permite presentar colecciones de datos en varias dimensiones, además de sus respectivas funciones para su manipulación [46].

2.3.4.2 Librería Pandas

La librería Pandas es empleada para el manejo y análisis de estructura de datos. Sus principales características son [47]:

- Nuevas estructuras de datos basadas en los *arrays* de NumPy, pero con novedosas funciones.
- Alta compatibilidad con ficheros CSV, Excel y bases de datos SQL. Permite su lectura y edición.
- Permite acceder o consultar datos a través de índices.
- Contiene funciones para ordenar, dividir y combinar conjuntos de datos.
- Permite manejar datos de series temporales.

Pandas contempla tres estructuras de agrupamiento de datos: series (una dimensión), DataFrame (dos dimensiones (tablas)) y panel (tres dimensiones (cubos)).

2.3.4.3 Librería Matplotlib

Librería cuya función principal es la creación de gráficos (véase Figura 2-18) en dos dimensiones. Permite crear, manipular y personalizar numerosos tipos de gráficos, los más destacados son:

- Diagramas de dispersión o puntos.
- Diagramas de líneas
- Diagramas de áreas.
- Diagrama de barras.
- Histogramas.
- Diagramas de sectores.
- Diagramas de caja y bigotes.
- Diagramas de violín.

Matplotlib permite la combinación de todos ellos y la representación de distintos datos, pero de igual objeto, en el mismo gráfico.

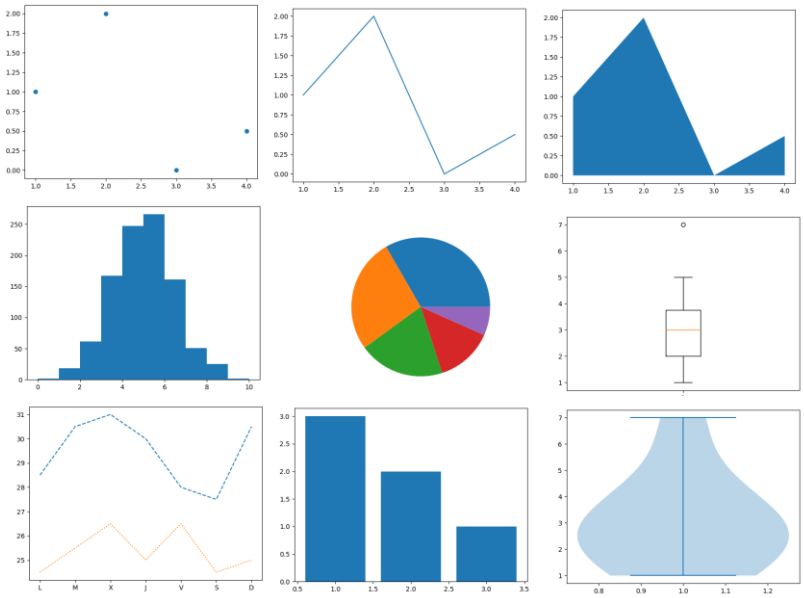

Figura 2-18 Gráficos de la librería Matplotlib

2.3.5 Jupyter Notebook

Jupyter Notebook es una aplicación web de código abierto que sirve a modo de puente constante entre el código y los textos explicativos. Así, los usuarios pueden crear y compartir en tiempo real código, ecuaciones, visualizaciones, etc. El nombre de *Jupyter* (véase Figura 2-19) proviene de la unión de los tres lenguajes de programación principales en los que se basa la aplicación: Julia, Pyhton y R. Aunque también es compatible con numerosos lenguajes.

Figura 2-19 Logotipo de Jupyter [48]

Esta aplicación web permite crear y compartir archivos web en formato JSON. El programa se ejecuta en un servidor del CUD y es accesible desde la aplicación web en cualquier navegador estándar. Es requisito para utilizar esta herramienta instalar previamente en el sistema el *software* de *Jupyter Notebook*. En este caso, al acceder remotamente a los servidores del CUD, no es necesario instalar nada en el sistema debido a que ya está instalado en el servidor al que se accede.

2.3.6 Kepler GL

Kepler GL es una herramienta de código abierto desarrollada por Uber y Mapbox que está compuesta por una plataforma cuyo objetivo es construir visualizaciones de información geográfica con excelentes resultados gráficos (véase Figura 2-20) y permitiendo una gran interacción del usuario. Permite trabajar con grandes volúmenes de datos que pueden estar agrupados en frameworks [49].

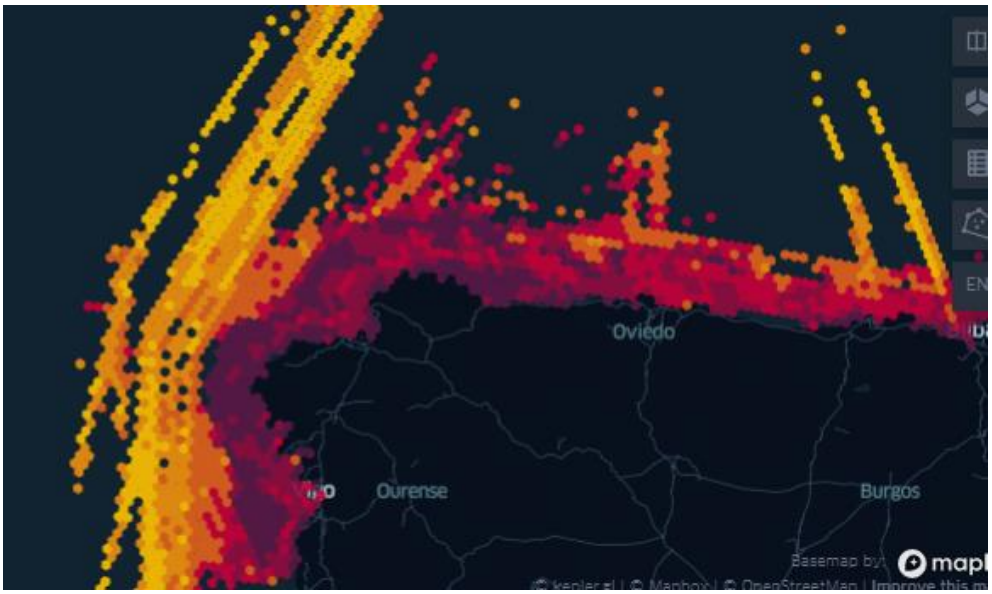

Figura 2-20 Representación Kepler de Mapbox

La característica fundamental de Kepler es que permite la fácil integración con otras plataformas como Python. Además, permite añadir capas de información, aplicarles filtros, exportar y compartir datos y mapas. Puede representar millones de puntos y es compatible con agregaciones espaciales, como por ejemplo las celdas geográficas.

Es una herramienta que permite, de forma rápida y eficaz, analizar datos geográficos de manera visual.

2.4 Trabajos relacionados

Los datos AIS han motivado numerosos estudios y proyectos de investigación. La información que proporciona el AIS es considerada, en lo que respecta a la detección de patrones de comportamiento y anomalías, la fuente válida principal, ya que, representan el mayor porcentaje de datos transmitidos en la mar. Existen numerosos trabajos relacionados en la literatura, a continuación se muestran algunos ejemplos.

El autor Adri Fluit, de la Administración Danesa de Seguridad Marítima, ha realizado un informe sobre la calidad de la información AIS de los mensajes estáticos [5]. Dicho trabajo tiene como objetivo estructurar los datos del *Big Data* marítimo y eliminar el ruido excedente, buscando así, la eficiencia y eficacia del almacenamiento y procesado de los datos AIS. Para ello, busca un sistema de almacenamiento de todos los datos dinámicos, evitando la duplicación y almacenando, únicamente una vez a menos que cambien, los datos estáticos y los relacionados con el viaje. Como resultado, elabora listas de buques que transmiten datos falsos, buques que transmiten MMSI asignados a otros buques y estadísticas que indican la mejora de calidad de los datos transmitidos por los buques.

Los autores de numerosos centros y universidades de China, han realizado un artículo sobre la Identificación de los tipos de buques pesqueros y análisis de las actividades estacionales en el norte del Mar de China Meridional a partir de los datos del AIS [50]. En dicho trabajo se identifica los diferentes tipos de pesqueros: con redes de enmalle, de arrastre y pesca con cerquero. Para ello analiza los patrones de comportamiento de los distintos tipos de pesqueros y crean un algoritmo de aprendizaje supervisado, *Light Gradient Boosting Machine (LightGBM),* para entrenarlo a través de dichos patrones de comportamiento. Los parámetros que analiza (véase Figura 2-21) para la detección son la media, el cuartil superior e inferior, el STD (desviación estándar) y el coeficiente de dispersión para la velocidad, el rumbo, las variaciones de latitud y longitud y el desplazamiento. De los resultados obtenidos (precisión de 0.9562 y exhaustividad de 0.9578) se puede concluir que es posible identificar la actividad de un buque con precisión a partir de datos cinemáticos. Para ello, el algoritmo considera los parámetros más importantes, y en este orden, a los siguientes: la STD de la velocidad segmentada ([0, 0.6] nudos), la STD de la velocidad global, la media de la velocidad global, la media de la velocidad segmentada ([0, 0.6] nudos) y la STD del desplazamiento segmentado ([0.3, 1.3] metros).

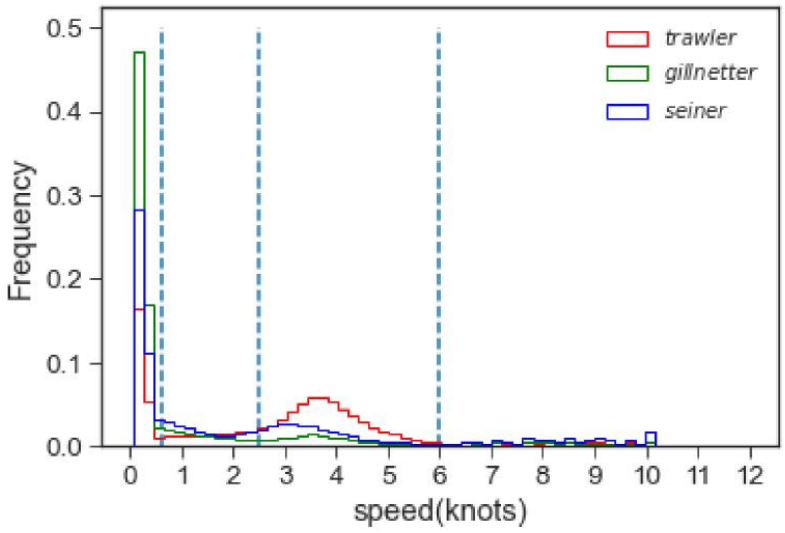

Figura 2-21 Ejemplo de análisis de parámetros [50]

Autores italianos del Centro Común de Investigación de la Comisión europea, han realizado un artículo de conferencia titulado "Descubrimiento de las actividades de los buques en el mar mediante datos AIS: Cartografía de la huella pesquera" [51]. El resultado de dicho trabajo es la presentación de un algoritmo encargado de la extracción de datos sobre la actividad de buques mediante aprendizaje no

supervisado. En particular, se consideran las zonas de pesca explotando los datos AIS históricos emitidos por los buques pesqueros. El algoritmo detecta y agrupa (DBSCAN) los puntos en los que, probablemente, los barcos estén pescando. Permitiendo construir un mapa de las zonas de pesca en un marco temporal (véase Figura 2-22).

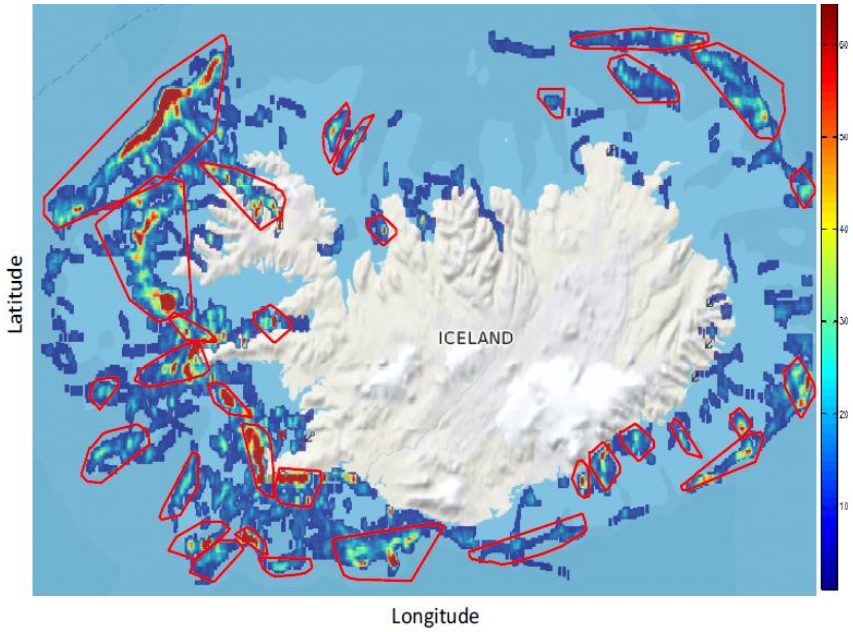

Figura 2-22 Detección de zonas de pesca [51]

3 DESARROLLO

En este capítulo se describe el método utilizado para identificar los buques que cometen actividades sospechosas de tráfico ilegal, a partir de datos indirectos como la cinemática, zona de actividad, el tipo de barco o datos registrales. En primer lugar, se expone cómo se lleva a cabo el acceso a la base de datos del CUD y las herramientas utilizadas. Una vez establecido el entorno de trabajo, se procede al desarrollo del objetivo global y específicos.

3.1 Configuración del entorno

Para poder manejar los datos, se procede al acceso a los servidores del CUD, los cuales tienen disponible la base de datos AIS a utilizar. Una vez se accede a la base de datos se procede a realizar consultas mediante el Motor de búsqueda *Elasticsearch*.

3.1.1 Instalación de Putty y acceso al servidor

PuTTY es un *software* de código abierto diseñado para utilizar el protocolo SSH en *Windows*. El principal uso que se le da es para conectarse de forma remota a otro equipo, en este caso, al servidor del proyecto del CUD. La interfaz de este programa es muy simple (véase Figura 3-1). Contiene un menú de configuración en el que se introduce la dirección IP y el puerto del equipo/servidor al que se desea conectar. Estos dos parámetros se han de conocer previamente a la configuración. También es necesario configurar el apartado de *SSH/tunnels* para que redireccione correctamente.

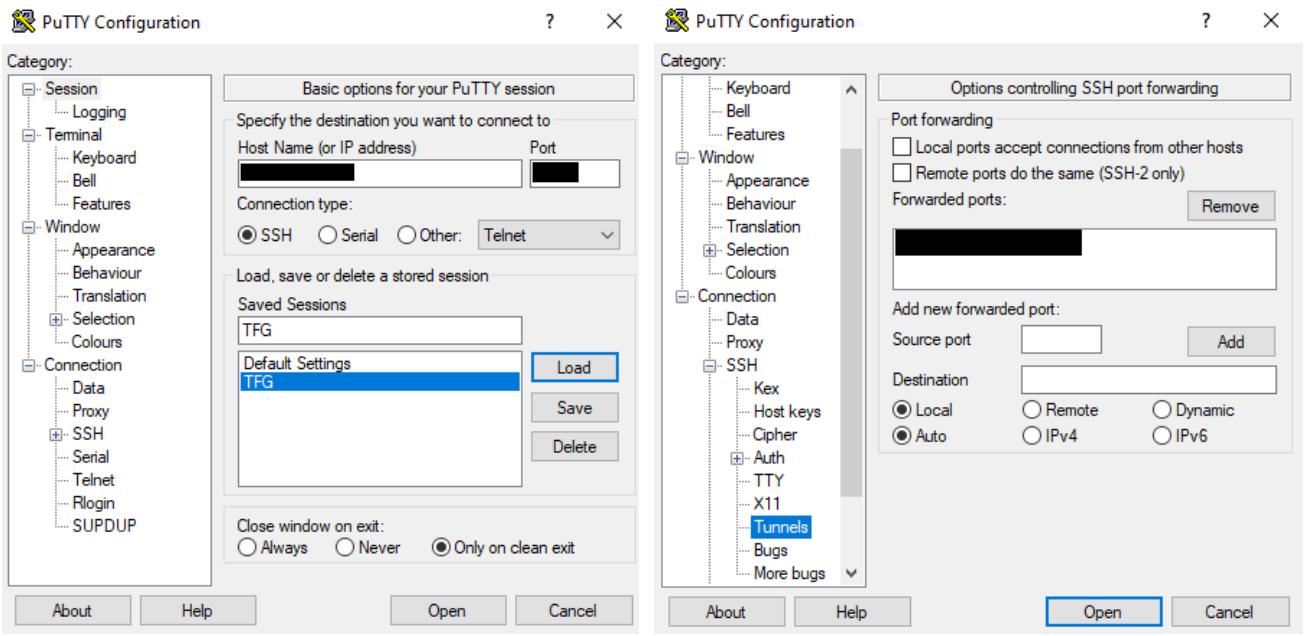

Figura 3-1 Menú de configuración de PuTTY

Antes de establecer la conexión, el servidor del CUD pide un usuario y contraseña para acceder a él (véase Figura 3-2). Una vez establecido el enlace, se accede al servidor a través de *localhost* desde el navegador. Al acceder al localhost, la conexión se redirige al servidor de *Jupyter Notebook* que está instalado en el servidor que tiene almacenada la base de datos a utilizar.

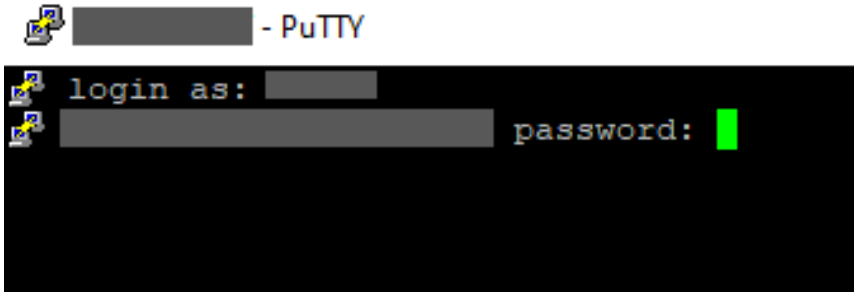

Figura 3-2 Inicio de sesión a los servidores del CUD

3.1.2 Consultas a través de Elasticsearch

Para hacer las consultas a la base de datos se utiliza el Motor de búsqueda *Elasticsearch* a través de un *script* de Python. En dicho *script* se escribe el código deseado haciendo llamadas a la base de datos mediante Elastic, y así, personalizar las consultas: filtrarlas por zona, tipo de barco, características, etc. Es decir, se ajusta la consulta mediante la restricción de cada uno de los campos de la base de datos (Anexo I: Campos de la base de datos) que se desee. Esta consulta se ordena de manera personalizada y se almacena en un dataframe a través de la Librería Pandas. El dataframe se puede modificar haciendo los cálculos que se deseen y añadiendo nuevos campos. El dataframe resultante de cada consulta es usado para el análisis y es manipulado por las aplicaciones descritas en 0.

A continuación se añade un ejemplo de una consulta simple y se explica para su comprensión:

```
“results = elastic.search(index=_index_name, body={"query": { "bool": { "must":
                                                    [{"term": { "mmsi":"CONFIDENCIAL"}},
                                                    {"term": { "dentro_ZEE": True}},
                                                    {"range": {"sog": {"gt": 2}}},
                                                    {"term": { "tipo_buque_AIS": "mercante"}}]}},
                                                  "sort": [{"@timestamp": {"order": "asc"}}],
                                                  "size": 10000})
df = Select.from_dict(results).to_pandas()”.
```

Primero se hace una llamada al motor de búsqueda *Elastic* y, a continuación, en esa llamada, se introducen la personalización deseada. En este caso, se está pidiendo a *Elastic* que devuelva todos los mensajes de la base de datos que cumplan con las siguientes características:

- Todos los mensajes devueltos por *Elastic* deben ("must") tener en el campo MMSI relleno con el número "CONFIDENCIAL", es decir, se está realizando una consulta de un único barco con ese número MMSI.
- Los mensajes transmitidos por el barco, deben tener el booleano "dentro_ZEE" con valor “True”, es decir, se solicita que los mensajes estén transmitidos dentro de la Zona Económmica Exclusiva de España.
- Todos los mensajes deben tener el campo "sog" por encima "gt" de 2 nudos, es decir, que la velocidad transmitida por el buque para cada mensaje sea mayor que 2.
- Además, el campo "tipo_buque_AIS" debe estar relleno con la palabra "mercante", es decir, el buque debe transmitir que es un mercante.
- Por último, se requiere que los resultados de la consulta sean devueltos de manera ordenada ("order") de menor a mayor ("asc") según la etiqueta de la fecha y hora "@timestamp" que tengan.

Para finalizar esta consulta, se almacenan resultados en un dataframe de la librería de pandas. Como se ha explicado antes, este dataframe se puede modificar haciendo los cálculos que se deseen y añadiendo nuevos campos para el análisis de los resultados.

Hay varias maneras de realizar una consulta en *Elastic*, algunas de las utilizadas en el trabajo son:

- Consultas realizando *scroll*, ya que, con la configuración estándar, *Elastic* limita el número de resultados a 10000. Para ello, se añaden los resultados de la consulta al dataframe uno a uno con el fin de que no haya límite máximo de posibles resultados.
- Consultas con agregación. Este tipo de consultas son de gran utilidad. Los resultados que devuelve la consulta no son los mensajes AIS propiamente almacenados, si no que devuelve una lista de los mensajes agregados al parámetro que se desee. Un ejemplo son las consultas por agregación de celdas, que permiten recibir la celda y cantidad de mensajes que se reciben en cada una (especificándola) o la agregación por MMSI, que agrupa los mensajes por MMSI señalando el número de MMSI y la cantidad mensajes que se reciben con las características implicadas. Es de gran utilidad cuando se desea hacer consultas barco a barco, para ello es necesario agrupar antes con agregación los barcos por MMSI para posteriormente realizar consultas buque a buque.

Es de vital importancia, realizar consultas de calidad que contengan información valiosa, ya que, una consulta con información redundante, empaña la validez de la información realmente valiosa. Es por ello, que los parámetros que se seleccionan para filtrar los datos han de ser seleccionados con sentido.

3.2 Análisis y desarrollo del algoritmo

Para la búsqueda de buques con los patrones sospechosos que se indican en 2.2.5, primero se realiza una consulta a los datos, agrupando los buques de manera separada con consultas con agregación y mediante su número MMSI. Es decir, primero se consulta la lista de MMSI con las características que a continuación se explican para poder realizar una consulta para cada MMSI y así, analizar el barco. Se realiza la consulta filtrando los datos de la siguiente manera:

- Han de ser barcos mercantes ("tipo_buque_AIS"), ya que, como se ha mencionado anteriormente, parte del tráfico ilegal se realiza a través de ellos.
- La velocidad (SOG) ha de ser mayor de 2 nudos debido a que, los buques con una velocidad igual o menor de 2 nudos podrían considerarse buques parados, y tenerlos en cuenta, desvirtúa la consulta, como se ha explicado anteriormente.
- Los buques deben estar dentro de la Zona Económica Exclusiva (dentro_ZEE) de España para acotar, en cierta manera, el globo a las zonas de interés, pues las anomalías que interesan son las anomalías que ocurren dentro de las zonas de interés nacional.

Antes de continuar, es conveniente explicar lo que se considera un buque parado. El principal problema es que los buques parados generan mucho ruido en la base de datos, ya que la gran mayoría de los barcos fondeados, o incluso atracados, siguen transmitiendo mensajes AIS. Además, se sabe que con frecuencia los buques fondean o se paran antes de entrar en puerto para ajustar la hora de llegada acordada con el puerto. Si estos mensajes se tienen en cuenta a la hora de analizar los datos, se hará un análisis con datos redundantes, lo que puede provocar que sea erróneo. Además, un buque con las máquinas paradas, o incluso fondeado, se mueve por el efecto de las corrientes y del viento, lo que en términos marinos se denomina derivar o abatir. Debido a este fenómeno, el COVAM indica que parado se puede considerar a los barcos cuya velocidad sea inferior a dos nudos estrictamente, siendo este el criterio para calcular el booleano "parado" en el etiquetado online. A continuación se analiza si este umbral se ha dispuesto correctamente.

Se ha realizado un histograma de la velocidad (SOG) de todos los mensajes de todos los buques de la base de datos (véase Figura 3-3) en todas las zonas con el objetivo de definir el umbral de lo que se considera un buque parado. Como se puede apreciar, se reciben muchos más mensajes a 0, 1 y 2 nudos. Por ello, se decide considerar buques parados también a los que vayan a una velocidad de 2 nudos. También, se destaca que el campo del SOG de la base de datos se ha almacenado como un valor entero. Por lo tanto, un buque que tenga una velocidad de 0, 1 o 2 nudos (menor que 3 nudos) se considera parado.

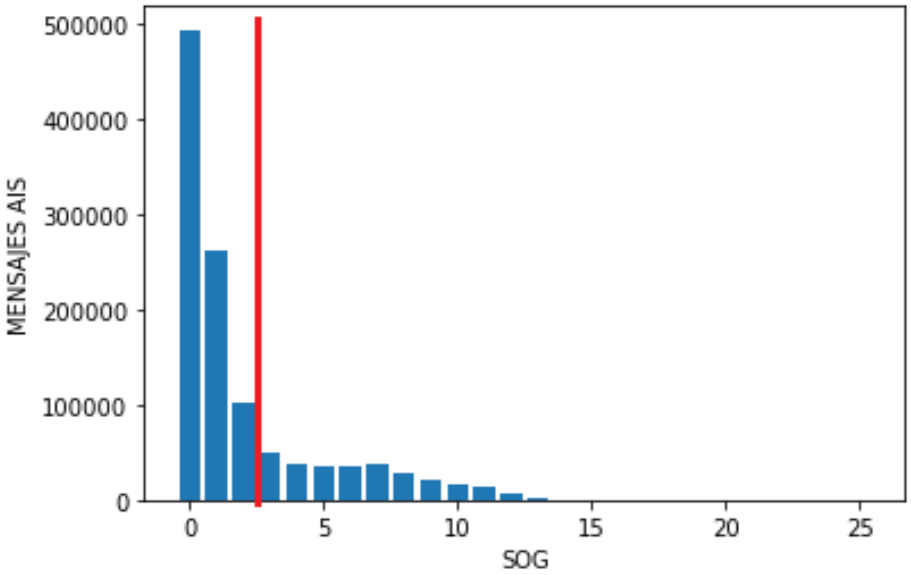

Figura 3-3 Velocidad indicada por todos los mensajes de la base de datos

La idea es analizar el número de celdas repetidas en los mensajes AIS de cada barco, ya que, podría ser un indicativo de buques con comportamiento anómalo. Esto es debido a que para un barco mercante, cuya función es transportar mercancía de un puerto a otro, recorrer muchas veces una misma celda podría ser sospechoso. La valoración del algoritmo se hará cualitativamente analizando los resultados caso a caso debido a que los datos no están etiquetados según si los buques son anómalos o no.

Es conveniente aclarar que la zona del Estrecho es una zona peculiar, ya que coinciden muchos puertos de diferentes países en una zona muy pequeña. Además, es la entrada y salida del Mediterráneo, por lo que muchos barcos repetirán numerosas veces la celda por la que transitan. Como se ve más adelante, esto hace que el algoritmo detecte muchos barcos sospechosos que no lo son realmente. Además, se ha detectado una zona de espera en la que muchos buques se quedan a la deriva (con las máquinas paradas) esperando a entrar en Algeciras, uno de los puertos más concurridos de España (véase Figura 3-4). Destacar que todas las derrotas se visualizan a través de *Kepler GL*. Se anexa un ejemplo (Anexo II: Visualización derrotas de los buques).

Figura 3-4 Zona de espera para el puerto de Algeciras

En principio, el algoritmo a desarrollar no en cuenta estos buques, ya que se consideran parados, sin embargo, algunos de grandes proporciones no están completamente parados en la zona de espera. Estos se encuentran a una velocidad que no consuma demasiado combustible (3-5 nudos) pero sin llegar a

estar parados por seguridad. Si un buque de grandes proporciones para las máquinas para quedarse a la deriva, tardará un periodo considerable en volver a ponerse en marcha debido al gran desplazamiento de este, siendo muy peligroso debido al elevado tiempo de reacción ante cualquier situación de peligro. Además, el tipo de derrota que siguen en este periodo de espera es muy característica: en una zona muy reducida, de 3 o 4 celdas H3-6, hacen derrotas sin salirse de esta área (véase Figura 3-5). Sin embargo, el Estrecho también es uno de los escenarios donde más tráfico ilícito se realiza.

Figura 3-5 Ejemplo derrota de espera

Además, como se ha mencionado anteriormente, parte de los buques que realizan actividades sospechosas o que ya han sido incautados, han realizado cambios de bandera con el objetivo de evadir la justicia del país de su pabellón. Según [52], haciendo referencia a los barcos que han cambiado de bandera: "a menudo se desconfía de las mercancías y tripulantes que llegan a los puertos cuando el registro es abierto. Productos de contrabando, drogas, ruptura de embargos, prostitución y delincuentes varios son capturados en este tipo de barcos". Por otro lado, las principales banderas de conveniencia son las siguientes (véase Figura 3-6).

Figura 3-6 Principales banderas de conveniencia [52]

Una vez se realiza la consulta filtrada y agrupada para cada buque, se procede a la consulta personalizada con el objetivo de identificar los buques que pasan por una celda H3-6 más de cierto número de veces en los 15 días de recogida de datos. Se podría considerar un indicativo de anomalías debido a que, un buque mercante con una velocidad media de 8 nudos y con una velocidad normalizada de transmisión de mensajes AIS, transmite una media de 5 mensajes AIS cuando transita por cada celda H3-6 (véase Figura 3-7). Por lo que se podría considerar normal que un buque transite al menos 10 veces por la misma celda, ya que, un barco puede hacer varias veces la misma ruta en un corto periodo, sin embargo, es necesario añadir un margen de respeto.

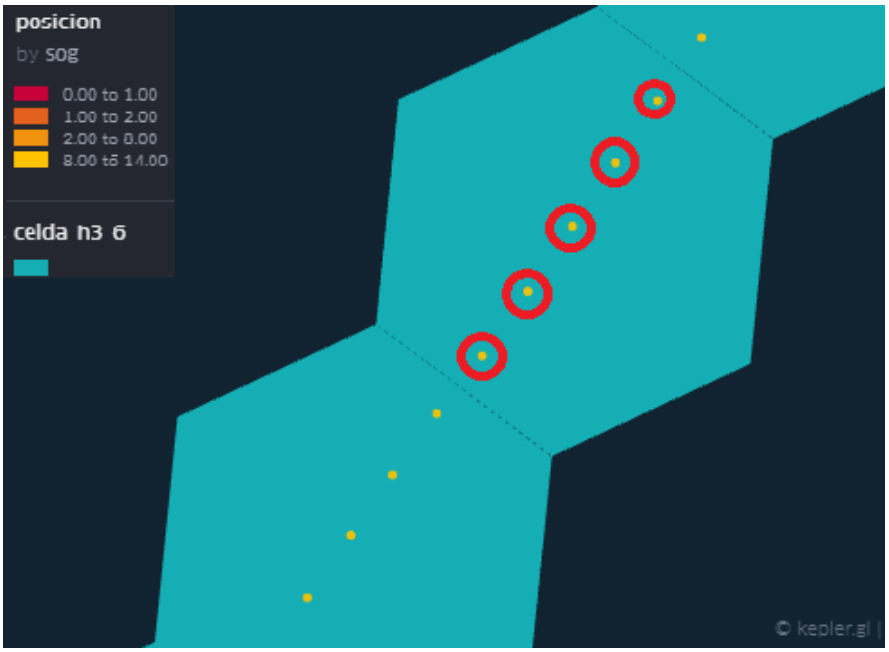

Figura 3-7 Tránsito normal de un buque mercante por una celda H3-6

Es necesario añadir un margen de repeticiones de mensajes transmitidos por barco en cada celda debido a un posible cambio de rumbo no anómalo o tránsito por la diagonal del hexágono. Para ello se realiza un gráfico (véase Figura 3-8) de los barcos anómalos (alarmas) que saldrían escogiendo los diferentes márgenes y con el objetivo de elegir el margen adecuado. Los gráficos se realizan teniendo en cuenta la implementación del tiempo real, que más adelante se explica.

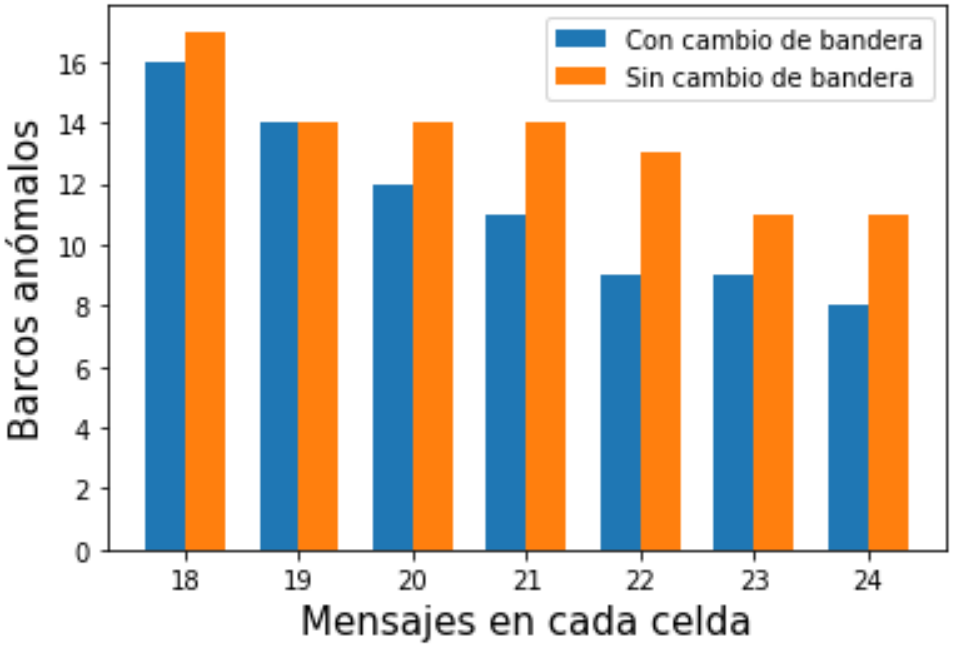

Figura 3-8 Cambios de bandera de barcos anómalos en función de las repeticiones en cada celda

Como se puede apreciar, el umbral con mejores resultados es aquel en el que los barcos, que no estén parados, repiten más de 19 mensajes en una celda H3-6, ya que la proporción de barcos con cambio de bandera (lo cual es un indicio de actividad sospechosa) es la más alta para este umbral, que detectaría 28 barcos anómalos en los 15 días, es decir, casi 2 buques diarios de media, lo que es razonable. Por lo que se considerarán anómalos a los buques mercantes, que no estén parados, que transmitan más de 19 mensajes AIS en una misma celda H3-6.

Para la implementación en tiempo real de esta búsqueda, se propone el almacenamiento de una lista de las celdas de los últimos mensajes AIS. Cuando en esta lista, se repita más de 19 veces una misma celda, el comportamiento del barco se considera anómalo. Para definir el umbral de cuantas listas se almacenan, se analizan los resultados realizando un gráfico del número de anomalías en función del histórico almacenado (véase Figura 3-9). También se compara con los resultados de una consulta histórica consultando todas las celdas (Anexo III: Obtención lista barcos anómalos mediante consulta histórica). Como es obvio, el número de anomalías aumenta al aumentar la profundidad del histórico, pero a partir de 150, el incremento no es significativo. Por ello, se considera la opción más eficiente almacenar 150 celdas debido a que, con ese volumen, es posible detectar muchos de los barcos que se detectarían haciendo una consulta histórica. Dado que la base de datos de 15 días dispone de 156.914 barcos diferentes, se almacenan 150 celdas y cada celda ocupa 64 bits, implementar esta lista supone un espacio de almacenamiento de 188.29 MB. Este tamaño es aceptable para que sea implantado en *Redis* durante el procesado de los datos, ya que el espacio de almacenamiento no es grande y las consultas a esta lista serían rápidas.

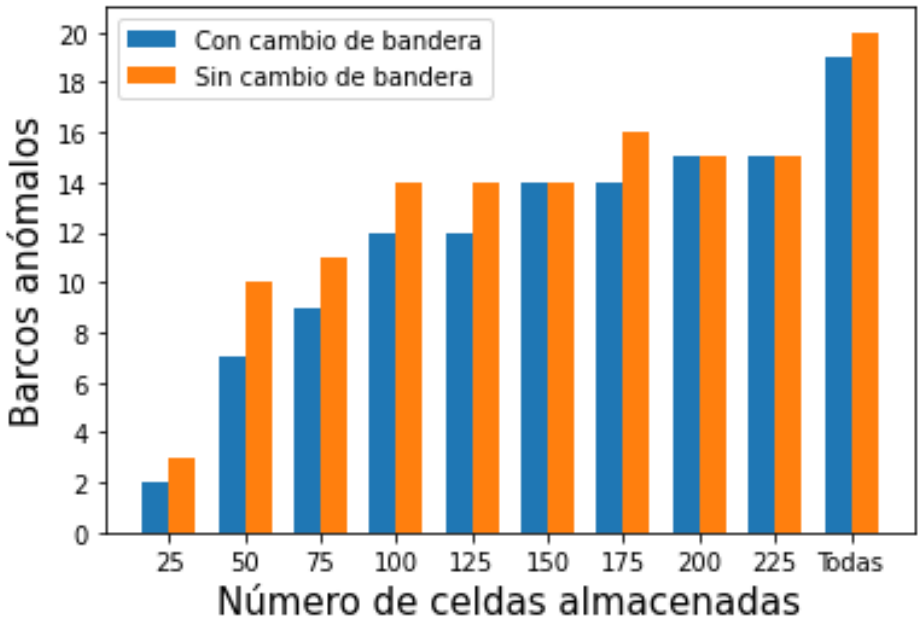

Figura 3-9 Cambios de bandera de barcos anómalos en función de las celdas almacenadas

3.3 Resultados

Mediante las restricciones y filtros descritos en el apartado anterior y aplicando la implantación en tiempo real, durante los 15 días de recogida de datos, se obtiene una lista de 28 mercantes que aparentemente tienen comportamiento anómalo dentro de la ZEE de España (véase Tabla 3-1) (Anexo IV: Obtención lista barcos anómalos mediante almacenamiento de celdas).

MMSI						
209188000	210163000	224094450	224121630	224133940	224226000	224237000
224388570	224405560	224503230	224810000	225410000	225977980	225985074
225986632	229610000	244060083	247318900	255806271	311005800	CONFIDENCIAL
314464000	370801000	431076000	CONFIDENCIAL	538001857	CONFIDENCIAL	636019366

Tabla 3-1 Mercantes sospechosos detectados

A continuación se analiza cada uno de estos barcos con el objetivo de detectar los que se comporten de manera sospechosa.

3.3.1 Buques sin cambios de bandera

Todos los buques de la lista que no han realizado cambio de bandera (14), tienen un comportamiento que, a priori, no parece sospechoso. La razón por la que se han detectado como buques con supuesto comportamiento sospechoso, es debido a que la gran mayoría son mercantes que realizan el mismo tránsito repetidamente con el objetivo de transportar mercancía entre dos puertos. Un ejemplo es el buque con número MMSI 247318900 (véase Figura 3-10):

IMO: **9471068**
Name: **EUROCARGO CAGLIARI**
Vessel Type - Generic: **Cargo - Hazard A (Major)**
Vessel Type - Detailed: **Ro-Ro Cargo**
Status: **Active**
MMSI: **247318900**
Call Sign: **ICMR**
Flag: **Italy [IT]**
Gross Tonnage: **32850**
Summer DWT: **10780 t**
Length Overall x Breadth Extreme: **200.63 x 26.5 m**
Year Built: **2012**
Home Port: **PALERMO**

Figura 3-10 Datos y fotografía del mercante Eurocargo Cagliari [53]

Se trata de un mercante italiano que transporta vehículos y que realiza continuamente el tránsito Livorno-Barcelona-Valencia. La derrota recogida por la base de datos (véase Figura 3-11) muestra que el buque recorre siempre la misma ruta y es por ello, que el algoritmo lo detecta como anómalo, ya que, transmite más de 19 mensajes AIS por una misma celda H3-6.

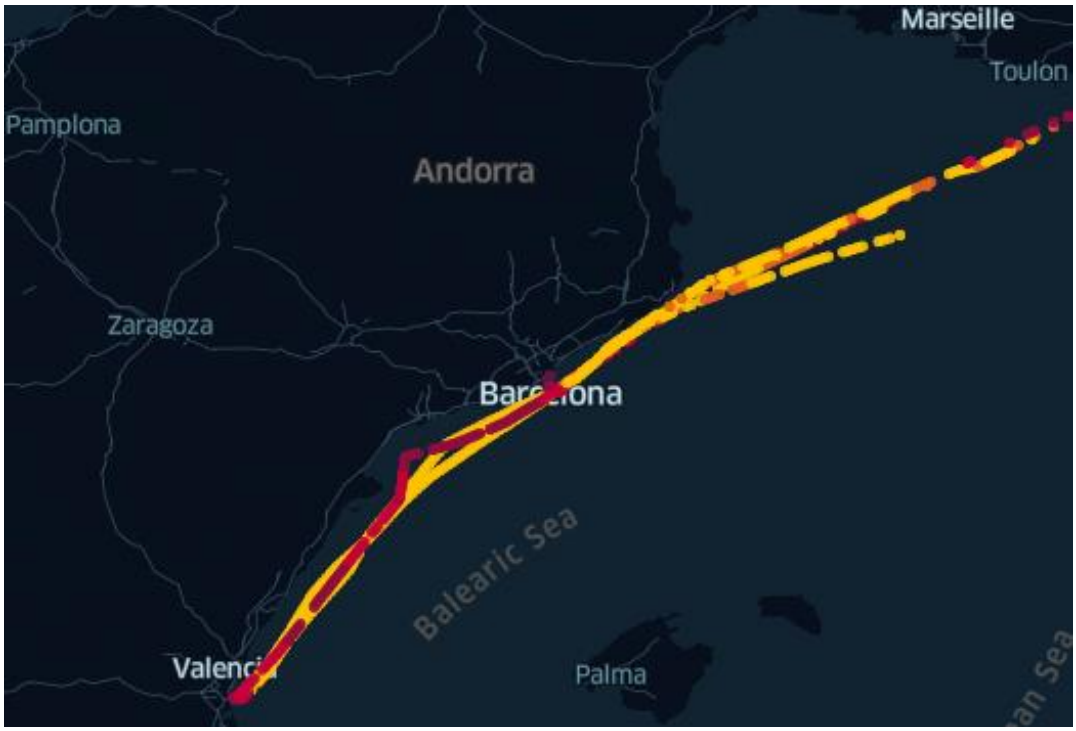

Figura 3-11 Derrota del mercante Eurocargo Cagliari

La gran mayoría de los buques de la lista que no han realizado cambio de bandera, son de este tipo, por lo que, no se consideran sospechosos.

3.3.2 Buques con cambios de bandera

A continuación, se seleccionan los buques que ha detectado el algoritmo y han cambiado de bandera, obteniéndose la Tabla 3-2.

MMSI						
209188000	210163000	224405560	224810000	225410000	229610000	255806271
CONFIDENCIAL	314464000	370801000	CONFIDENCIAL	538001857	CONFIDENCIAL	636019366

Tabla 3-2 Mercantes sospechosos detectados con cambio de bandera

3.3.2.1 Buques considerados no sospechosos

El buque con MMSI 209188000 se llama Miramar Express y es un buque con el mismo carácter que el Eurocargo Cagliari. Realiza siempre tránsitos entre Algeciras y Tánger (véase Figura 3-12) y es por ello que el algoritmo lo detecta anómalo. Además, existe una noticia en la que se notifica de unas reparaciones a las que fue sometido y se explica su cometido: transportar vehículos. Por lo que no se considera sospechoso [54].

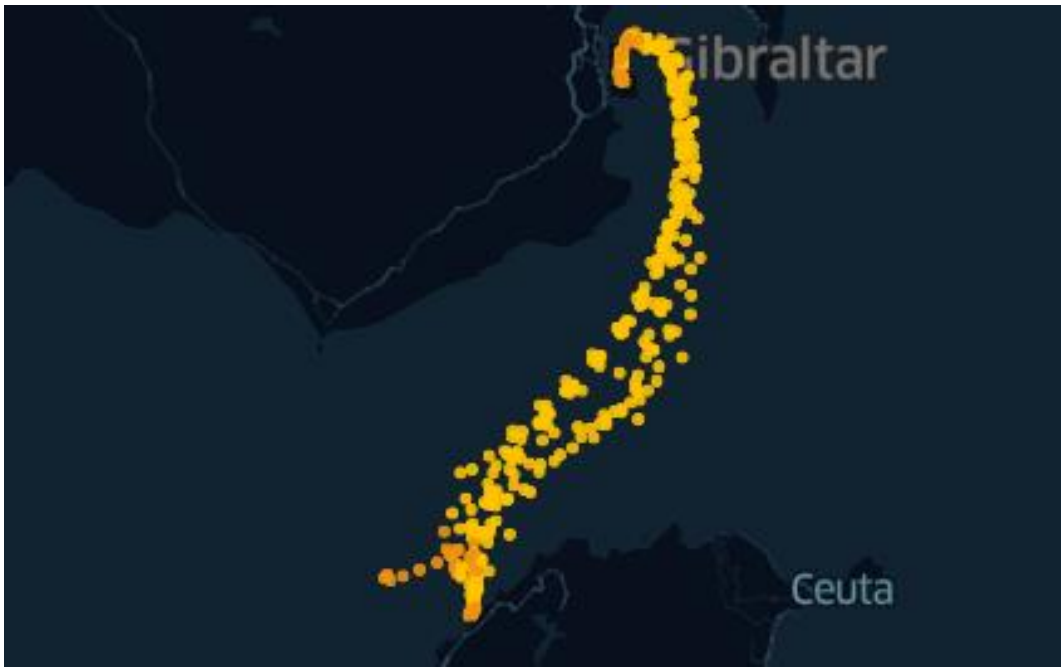

Figura 3-12 Derrota del mercante Miramar Express

El buque Maestro Sun con MMSI 538001857 es un ferry que recorre exactamente la misma ruta Algeciras-Tánger que el Miramar Express.

De igual manera, el mercante con MMSI 225410000 llamado Festivo, aunque este realiza rutas de Algeciras a Ceuta (véase Figura 3-13). Por lo que ninguno de los dos se considera realmente sospechoso.

Figura 3-13 Derrota del mercante Festivo

El buque GSL Susan con MMSI 636019366 es un mercante libanés de gran tamaño: 264 metros. El algoritmo detecta este buque como sospechoso porque al recalar en el puerto de Algeciras, se mantuvo en la zona de espera explicada previamente (véase Figura 3-14), por lo que no se considera como sospechoso. Cabe destacar que recientemente este buque tuvo que suspender sus servicios en Ucrania y cambiar de derrota debido al conflicto de Ucrania y Rusia [55].

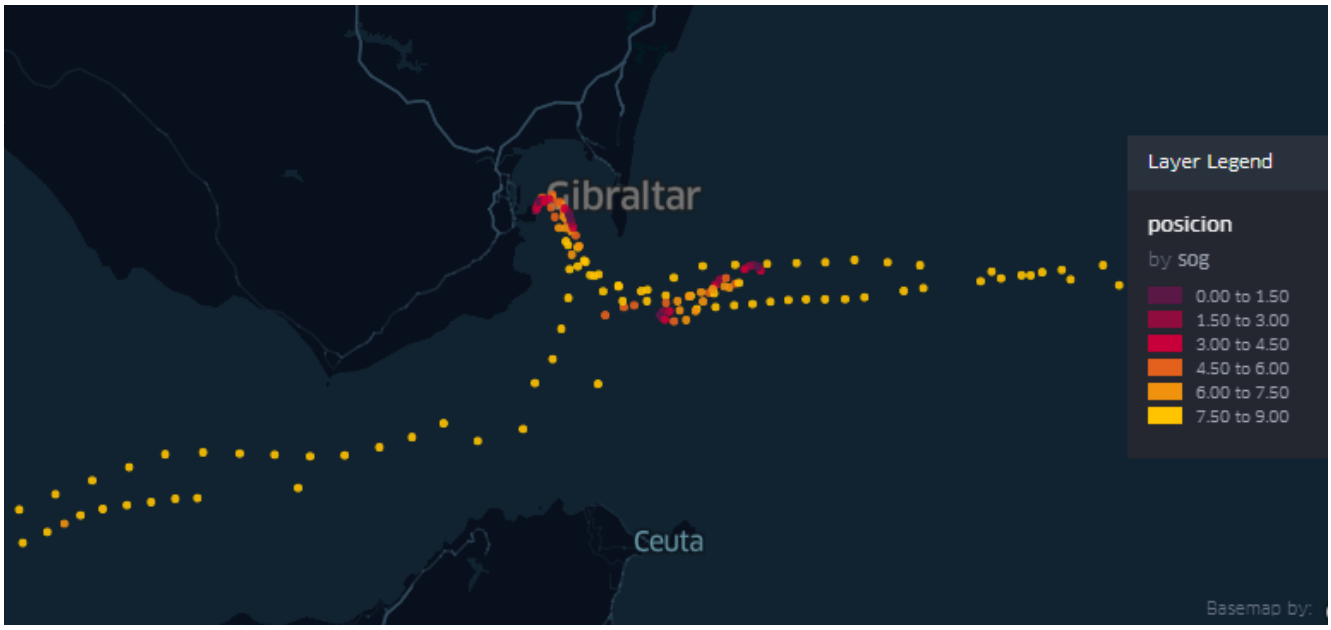

Figura 3-14 Derrota del GSL Susan en zona de espera

Ocurre lo mismo con el X Press Monte Blanco (MMSI 229610000), por lo que tampoco es considerado sospechoso (véase Figura 3-15). Es el mismo caso que el buque Northstart Glory con MMSI 370801000. Además, existe una noticia en la que se informa que en diciembre de 2021, este buque quedó completamente inoperativo por un temporal en Darnelos y tuvo que ser rescatado por las autoridades del lugar [56].

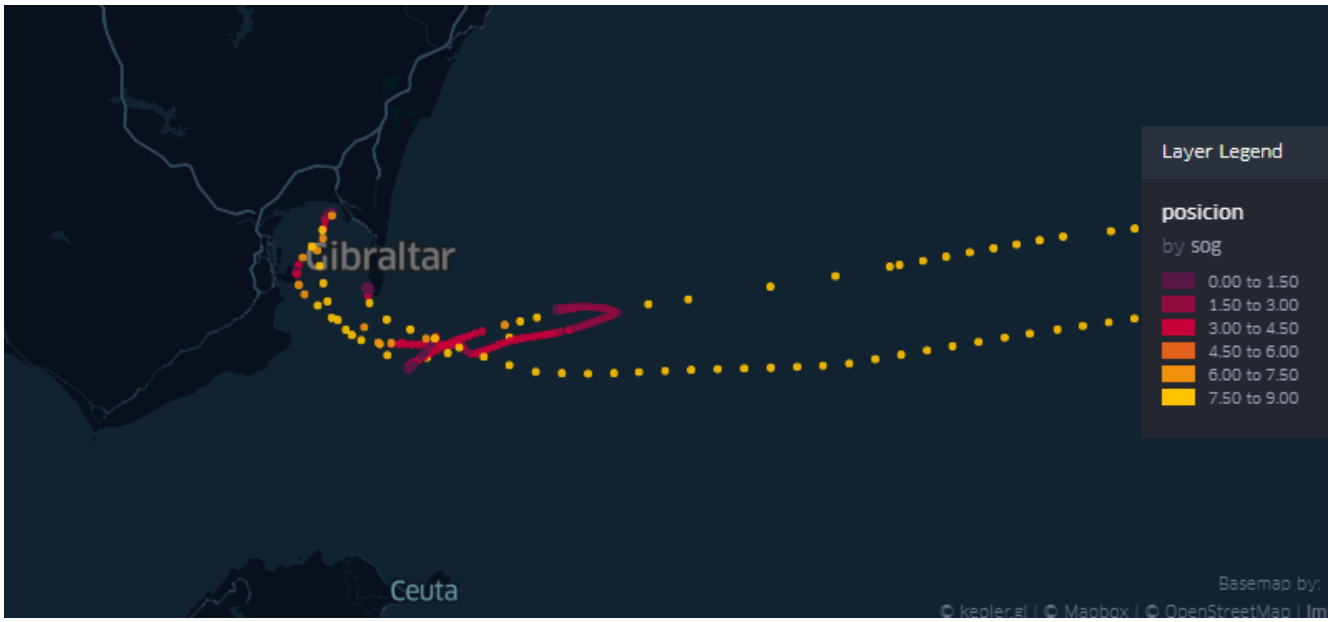

Figura 3-15 Derrota del X Press Monte Blanco en zona de espera

De igual manera, el buque Wilson Aviero con MMSI 314464000 con destino al puerto de Bilbao, se queda parado o casi parado (3 nudos) a la espera de su entrada (véase Figura 3-16). Es por ello que el algoritmo lo detecta como sospechoso cuando realmente no lo es.

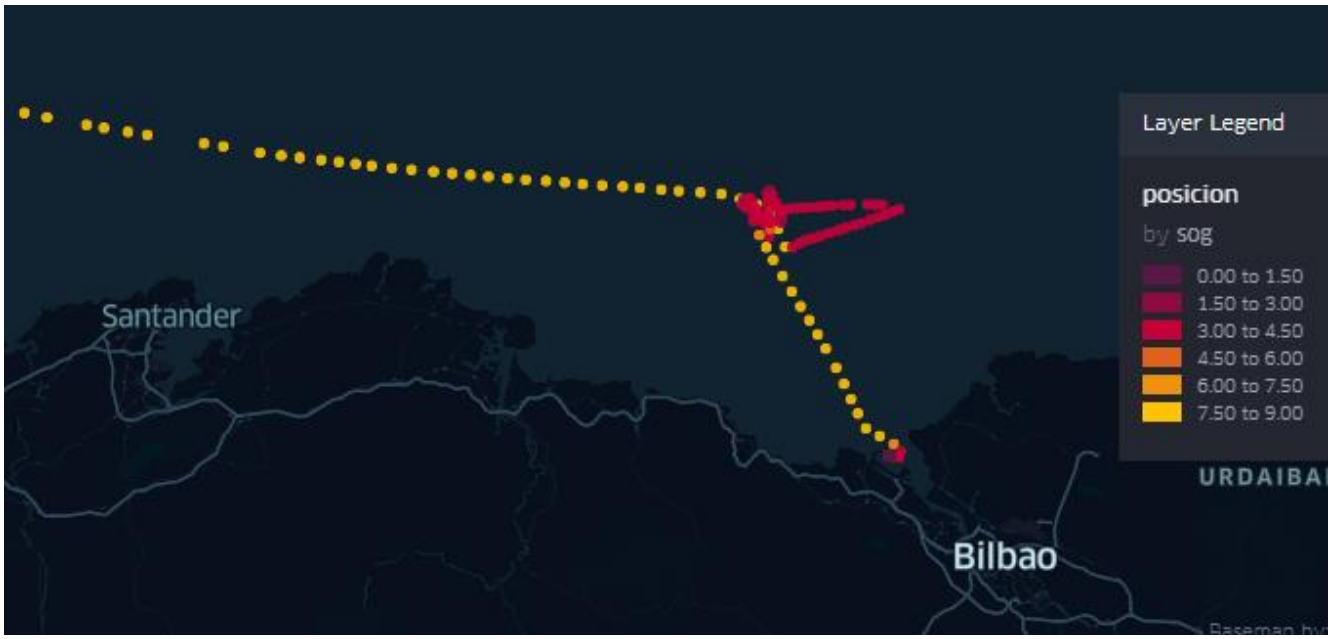

Figura 3-16 Derrota del Wilson Aviero en zona de espera

A su vez, el buque RS Lisa (MMSI 210163000) realiza continuamente la ruta Cádiz-Las Palmas-Tenerife-Lanzarote-Fuerteventura. Por lo que tampoco se considera un barco con comportamiento sospechoso.

Figura 3-17 Derrota del mercante RS Lisa

Otro buque de la lista con cambios de bandera con el mismo comportamiento no sospechoso es el Ofiusa Nova (MMSI: 224405560) (véase Figura 3-18). Su misión es transportar vehículos desde Ibiza a Formentera.

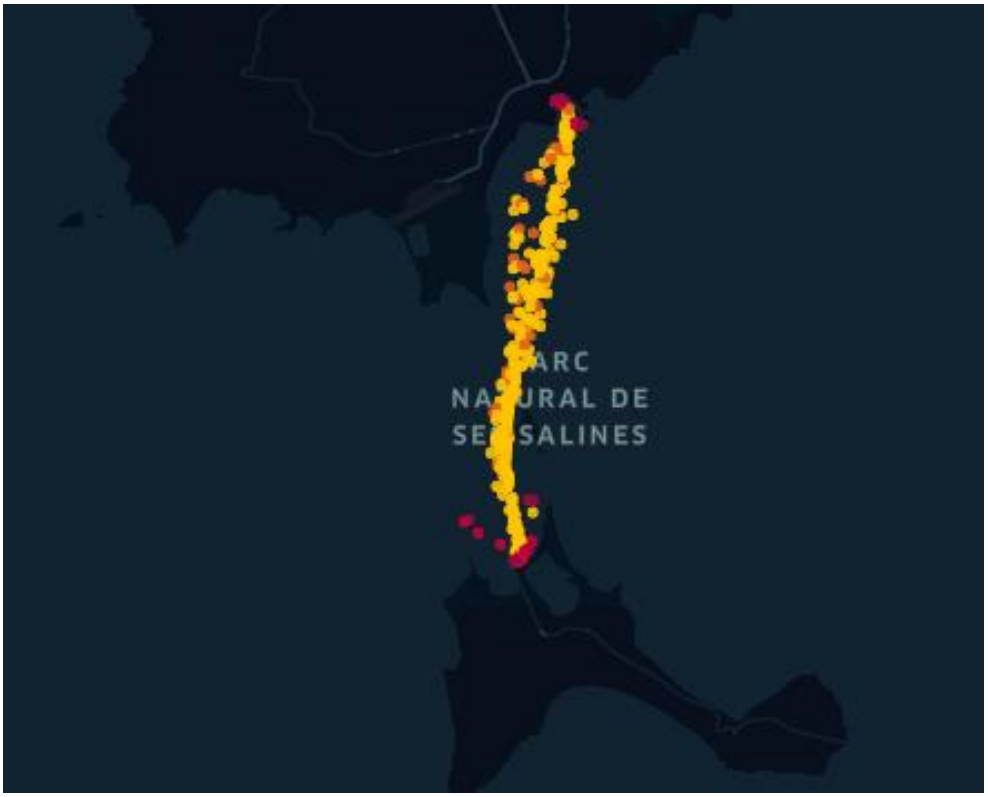

Figura 3-18 Derrota del mercante Ofiusa Nova

Cabe destacar que, según [57], este buque colapsó la línea Ibiza-Formentera durante varias horas por un camión que quedó atrapado mientras desembarcaba al ceder la rampa del buque en septiembre del 2021 (véaase Figura 3-19).

Figura 3-19 Camión atrapado al desembarcar del Ofiusa Nova [57]

El buque con MMSI 224810000 es similar al anterior, también realiza la ruta Ibiza-Formentera continuamente.

3.3.2.2 Buques considerados sospechosos

El buque **B1 (CONFIDENCIAL)** (véase Figura 3-20), actualmente con bandera de Liberia, con MMSI **CONFIDENCIAL** ha sido detectado por el algoritmo como sospechoso.

CONFIDENCIAL

Figura 3-20 Datos y fotografía del mercante B1 [58]

Este buque se construyó en el 2010 y se registró con el nombre de **CONFIDENCIAL** y armador **CONFIDENCIAL**. Un año más tarde, en 2011, el buque cambió de nombre, bandera y armador al actual: pasó a llamarse **B1**, su bandera ser Liberia y el armador es **CONFIDENCIAL** [59].

Como se puede leer en [60] y en [61], Liberia es uno de los países con más presencia de tráfico ilegal. Tanto de armas, de personas, de diamantes o de droga.

La derrota general del buque durante los 15 días de recogida de datos es la mostrada en la Figura 3-21.

Figura 3-21 Derrota general buque sospechoso B1

A continuación se procede a analizar esta derrota con detalle. El buque **B1** sale del puerto de Casablanca el día 19 de mayo de 2021 con destino al puerto de Algeciras. Al llegar a las inmediaciones de Algeciras al día siguiente, espera con cierta velocidad (4 nudos) para entrar en el puerto (véase Figura 3-22). Esta espera es ligeramente sospechosa debido a que ocurre a unas 20 millas al noreste de la zona de espera habitual. Aunque no quiera decir que se corresponda con actividades ilícitas.

Figura 3-22 Espera del buque B1 antes de su entrada en Algeciras

Tras su recalada en el puerto de Algeciras, el 24 de mayo sale hacia el Mar de Alborán, donde se queda parado en la misma zona en la que lo hizo anteriormente (véase Figura 3-23). Transcurridas 24 horas se dirige al puerto de Tánger. Este comportamiento puede ser extraño, aunque podría tener sentido si lo que busca es ahorrarse pagar las tasas por estar un día más atracado en puerto.

Figura 3-23 Espera del buque B1 antes de su entrada en Tánger

Tras 9 horas atracado en el puerto de Tánger, sale de puerto y se vuelve a dirigir de nuevo a la misma zona, esta vez más al noreste que la anterior. Aquí permanece unas horas, luego vuelve a reanudar su ruta para del Mediterráneo (véase Figura 3-24).

Figura 3-24 Espera del buque B1 tras su salida de Tánger

Hay un momento que se pierde la señal AIS mientras está saliendo del Mediterráneo, sin embargo, se vuelve a recuperar a las pocas horas al norte de Casablanca (véase Figura 3-25).

Figura 3-25 Recuperación de señal AIS del buque B1 tras su pérdida

Cabe destacar que es peligroso realizar transbordos en la mar estando parado debido a que no se tiene control de la plataforma del buque. La velocidad mínima de seguridad depende del tamaño de los buques que se aproximan, pero oscila entre los 3 y 4 nudos. El buque **B1** es considerado sospechoso debido a que no es habitual sus repetidos pasos y estancias por la costa sur española. Lo más sospechoso es su salida de Tánger para ir a una zona relativamente cercana a costa, unas 12 millas, totalmente apartada de su ruta al Atlántico. Como se ha visto antes, una técnica utilizada es la del transbordo de mercancía mediante encuentros en la mar con embarcaciones pequeñas y rápidas. Este buque detectado por el algoritmo es sospechoso de dicha actividad.

Otro buque que ha detectado el algoritmo, es el mercante **B2 (CONFIDENCIAL)** con MMSI **CONFIDENCIAL**. La bandera del buque es la de Hong Kong (Figura 3-26).

CONFIDENCIAL

Figura 3-26 Datos y fotografía del mercante B2 [62]

El buque se construyó en 2008 por el astillero **CONFIDENCIAL**. Se desconoce su bandera anterior, pero se sabe por la base de datos que tuvo al menos un cambio de bandera. Actualmente, pertenece a la empresa de **CONFIDENCIAL** [63].

La derrota general del buque durante los 15 días de recogida de datos es la mostrada en la Figura 3-27.

Figura 3-27 Derrota general buque sospechoso B2

Se procede a analizar esta derrota con detalle. La base de datos recoge al buque **B2** desde que empieza a transmitir en el Mar de Alborán con rumbo hacia el Estrecho el día 18 de mayo de 2021 con destino al puerto de Tánger. Al salir del puerto, se dirige a 13 millas al sur de la costa de Marbella, y se queda prácticamente parado (3 nudos) durante unas horas. A continuación, vuelve a moverse para entrar en el puerto de Algeciras (véase Figura 3-28). Este comportamiento es sospechoso debido a que la zona en la que "espera", se encuentra sospechosamente lejos de la zona de espera para los buques que van a entrar en el puerto de Algeciras, en concreto 37 millas. De nuevo, este comportamiento podría estar ligado a tráfico ilegal.

Figura 3-28 Derrota del B2 Nile entre Tánger y Algeciras

Después de salir de Algeciras, el buque no vuelve a realizar comportamientos sospechosos. Se dirige a Barcelona y, más tarde, a Italia, como se puede apreciar en Figura 3-27.

El algoritmo ha detectado otro mercante, se llama **B3 (CONFIDENCIAL)** con MMSI **CONFIDENCIAL** y actualmente posee la bandera de Barbados (véase Figura 3-29).

CONFIDENCIAL

Figura 3-29 Datos y fotografía del mercante B3 [64]

El buque se construyó en 1998 y se registró con el nombre de **CONFIDENCIAL**, bandera de Países Bajos y armador **CONFIDENCIAL**. En 2004 cambió de nombre, bandera y armador, pasó a llamarse **B3**, tener bandera de Barbados y pertenecer al armador **CONFIDENCIAL**. Finalmente, en 2011, cambió de armador al actual, llamado **CONFIDENCIAL** [65]. Se puede apreciar, que es un buque que ha cambiado de manera excesiva en el registro, acción que puede llegar a ser sospechosa.

La derrota del **B3** registrada por la base datos es la mostrada en la Figura 3-30.

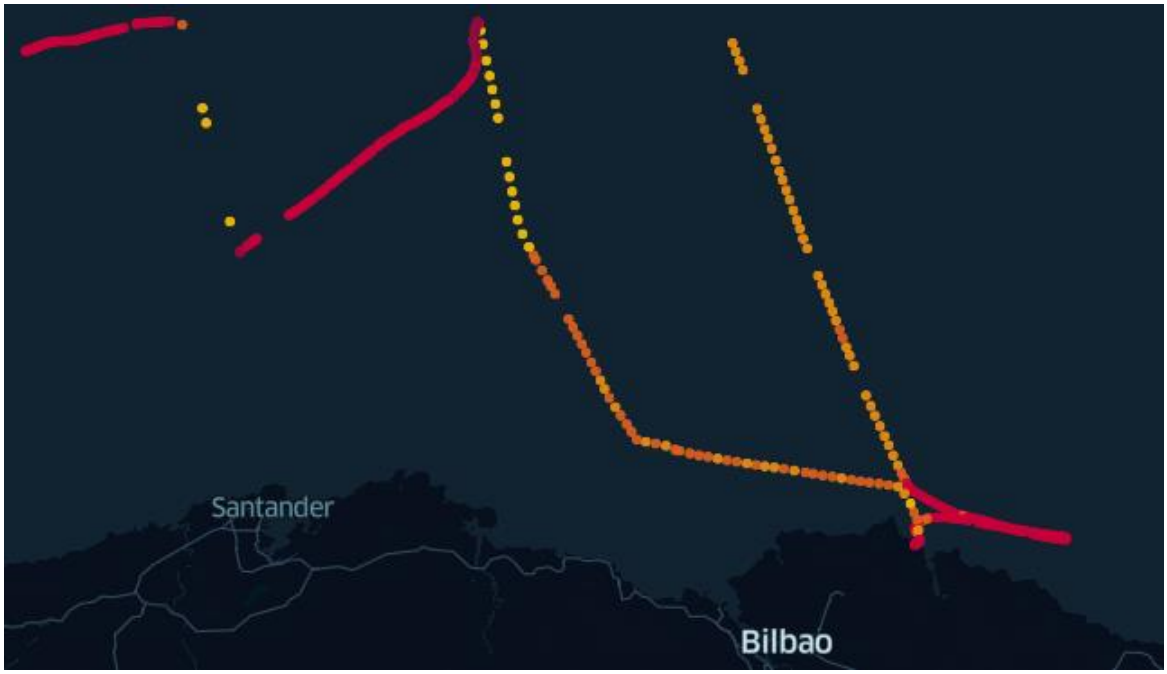

Figura 3-30 Derrota del mercante B3

En dicha derrota se aprecia como el buque se aproxima al puerto de Almika en Bilbao el 25 de mayo de 2021. Antes de llegar a puerto, se desvía 14 millas de la derrota para recorrer la costa vasca a 6 millas de la misma. Se desconoce el puerto de origen. Este comportamiento es considerado anómalo. Sin embargo, al salir de puerto, procede a incorporarse a unas de las rutas de tráfico mercante habitual, y, estando a 32 millas de costa, se para por completo y se queda a la deriva a una velocidad de 2-3 nudos

durante unas horas, luego vuelve a ponerse en marcha y procede a realizar lo mismo, parándose y dejando de transmitir. Por ello, estos comportamientos podrían estar ligados al tráfico ilegal.

3.4 Mejoras del algoritmo

Como se ha visto, muchos buques realizan muchas veces la misma ruta debido a que son ferrys o mercantes con una única ruta predeterminada entre dos puertos, por lo que hacen saltar la alarma sin ser anómalos. En este apartado, se trata de mejorar la discriminación del algoritmo incorporando el análisis del rumbo (COG) y de la velocidad (SOG).

Por otro lado, se realiza una comparación de los buques, que se han detectado a través del algoritmo, que realmente sean sospechosos y los que no lo son. Al representarlos se obtiene que 3 barcos (10.71%) de los 28 son sospechosos realmente. Además, si se comparan con los buques con cambio de bandera, se obtiene que los 3 barcos sospechosos representan el 21.42% de los 14 totales (véase Figura 3-31).

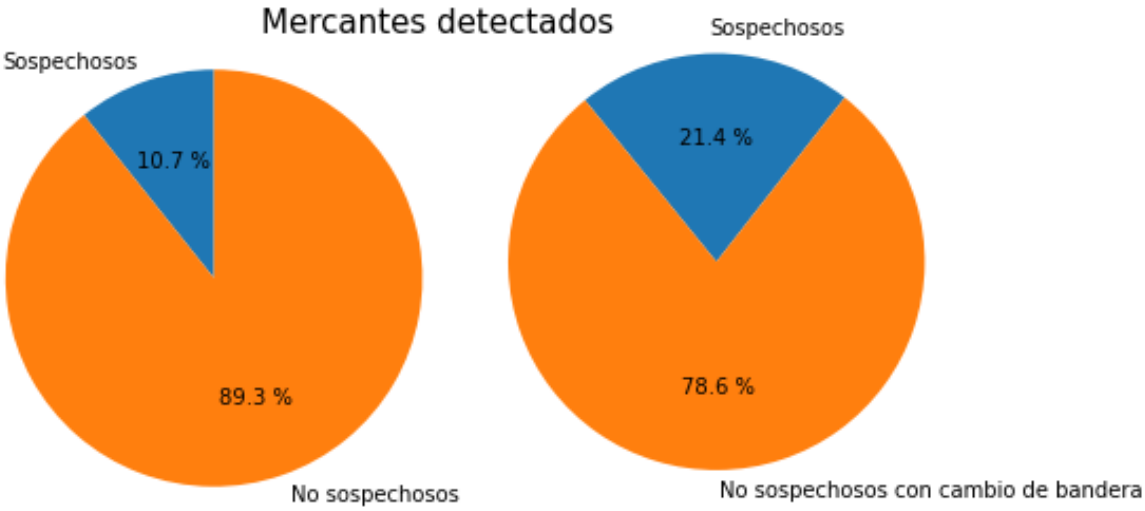

Figura 3-31 Comparación de los barcos sospechosos con los detectados

Por los resultados obtenidos, se puede indicar que el algoritmo es un buen indicador de actividades sospechosas, pero está abierto a mejoras.

Para solucionar este problema se analiza el “promediado_COG_W1” con el objetivo de discriminar en las consultas a los buques que hagan siempre la misma ruta. Este campo, como se explicó anteriormente en 2.3.3, contiene una media móvil exponencial de la variación del rumbo (COG). La siguiente gráfica muestra el “promediado_COG_W1” en función del tiempo de un buque que hace muchos cambios de rumbo. Se puede apreciar que existen valores muy altos, en concreto el pico llega a 70. El resultado se contrasta con la derrota del buque (véase Figura 3-32).

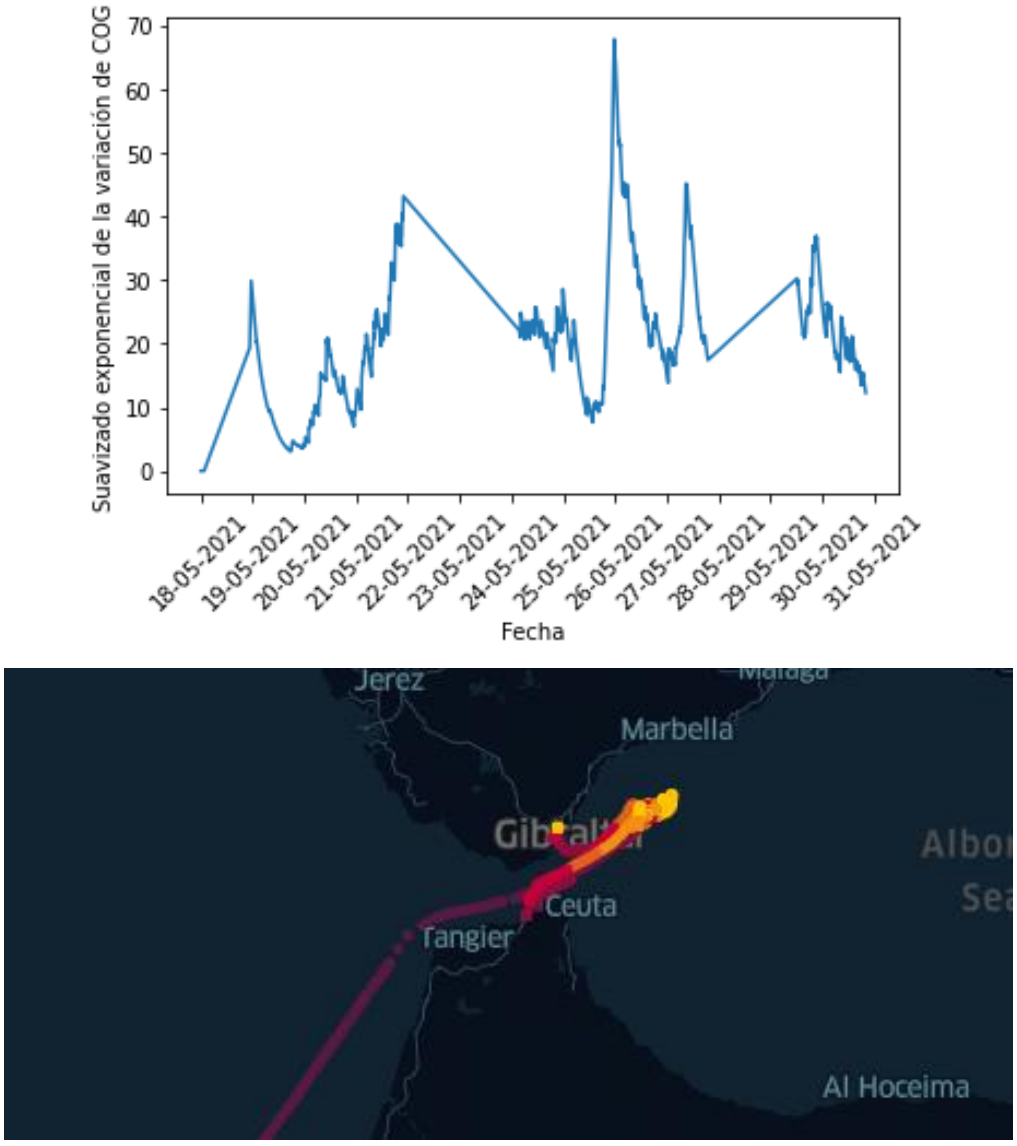

Figura 3-32 Representación del "promediado_COG_W1" en función del tiempo y derrota de un barco sospechoso con muchos cambio de rumbo

Sin embargo, si se comprueba la misma gráfica para un buque que sigue una ruta de tránsito desde el Atlántico hasta el Mediterráneo, se aprecia como el buque lleva un rumbo constante, por lo que el valor máximo del "promediado_COG_W1" es 14. El resultado se contrasta con la derrota del buque (véase Figura 3-33).

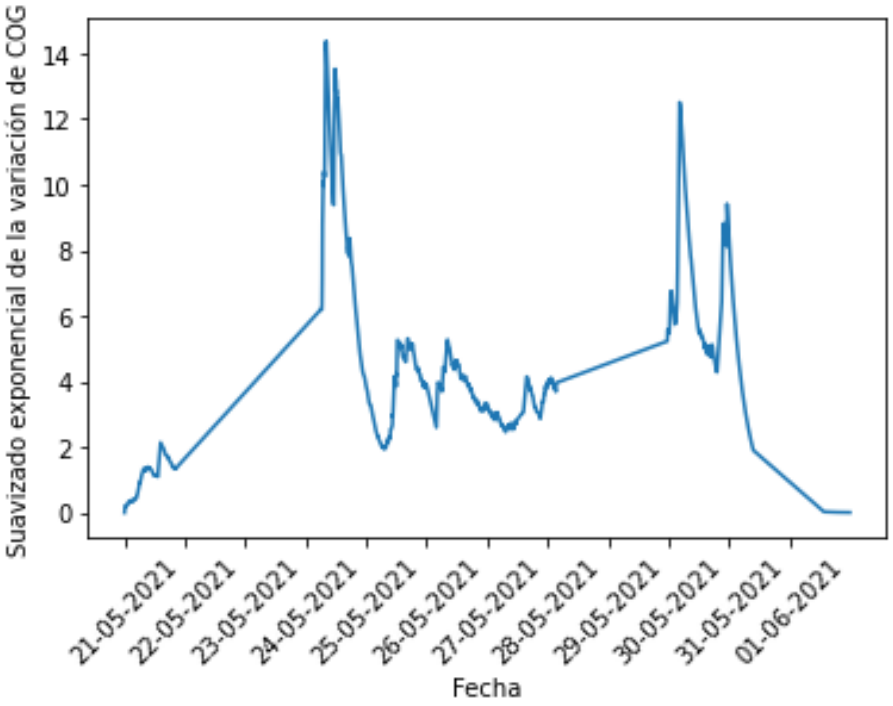

Figura 3-33 Representación del "promediado_COG_W1" en función del tiempo y derrota de un barco con pocos cambio de rumbo

Para analizar el "promediado_COG_W1", se realiza un gráfico representando el número de alarmas (barcos anómalos) que se producirían si se incorpora un umbral sobre el "promediado_COW_W1" en el análisis. En concreto, se evalúan diferentes umbrales de promediado. De manera que, por ejemplo, un umbral de 50 representa los barcos que, además de repetir 19 mensajes en una misma celda, tienen un "promediado_COG_W1" máximo por encima de 50 (véase Figura 3-34) (Anexo V: Obtención lista barcos anómalos usando promediado COG).

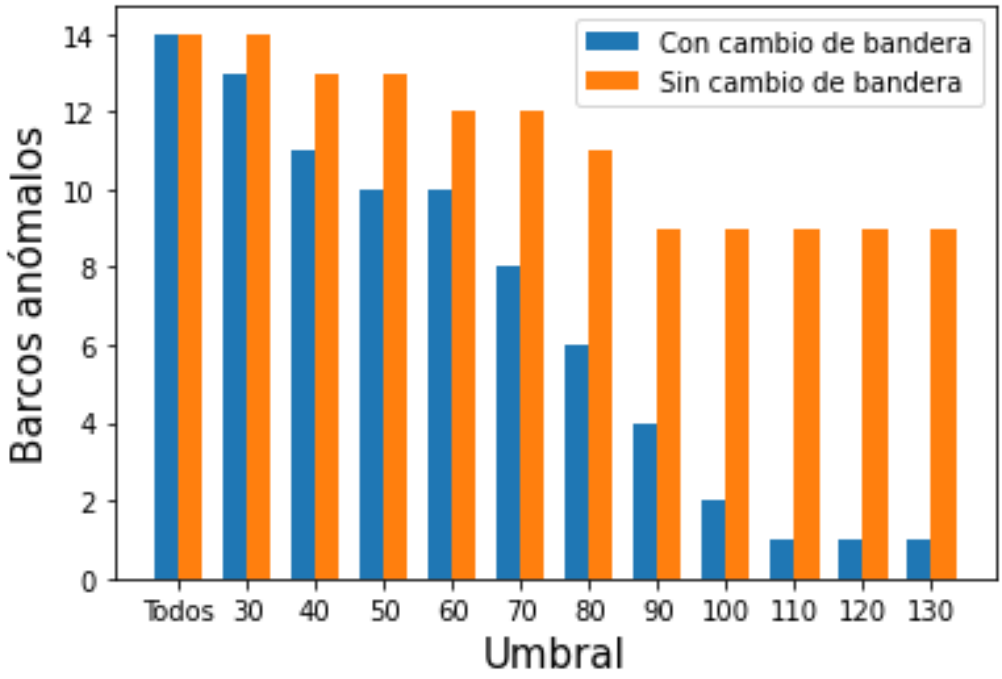

Figura 3-34 Cambios de bandera de barcos anómalos en función del máximo "promediado_COG_W1" y las repeticiones en cada celda

Los resultados de esta gráfica llaman la atención, debido a que se puede apreciar que los buques que más han cambiado de rumbo son los buques sin cambios de bandera. Para poder llegar a una justificación se procede a analizar más a fondo estos resultados.

Para empezar el análisis profundo del "promediado_COG_W1" (Anexo VI: Análisis de parámetros indicadores de actividades sospechosas de todos los mercantes) se realiza una gráfica de dispersión en la se representan todos los buques mercantes de la base de datos que se encuentran dentro de la ZEE y no están parados. Esta representación se realiza en función de su "promediado_COG" máximo y del número máximo de mensajes transmitidos en una celda (véase Figura 3-35).

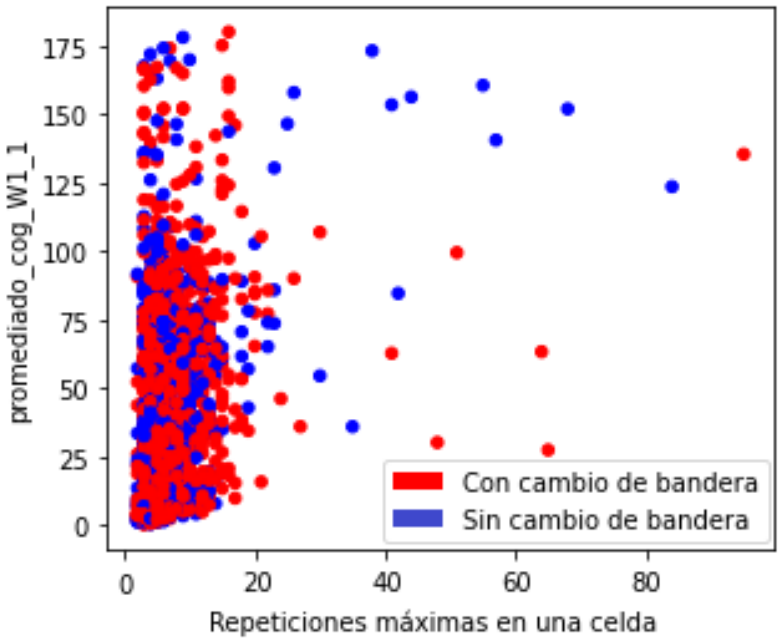

Figura 3-35 Buques mercantes dentro de la ZEE en función de su "promediado_COG" máximo y del número máximo de mensajes transmitidos en una celda

Como se puede visualizar, los buques mercantes con mayores máximos en el "promediado_COG_W1" son los que no han realizado cambio de bandera. Se procede a visualizar una gráfica parecida, esta vez representando todos los buques mercantes de la base de datos que se encuentran dentro de la ZEE y no están parados, en función del número máximo de mensajes transmitidos en una celda y el "promediado_COG" máximo en la celda más repetida. Es decir, en el eje x representa el número máximo de mensajes que se transmiten en una misma celda y el eje y representa el "promediado_COG_W1" máximo que se recibe en la celda más repetida, diferenciándose de la anterior representación que representaba el "promediado_COG_W1" máximo de los barcos en todo el histórico (véase Figura 3-36).

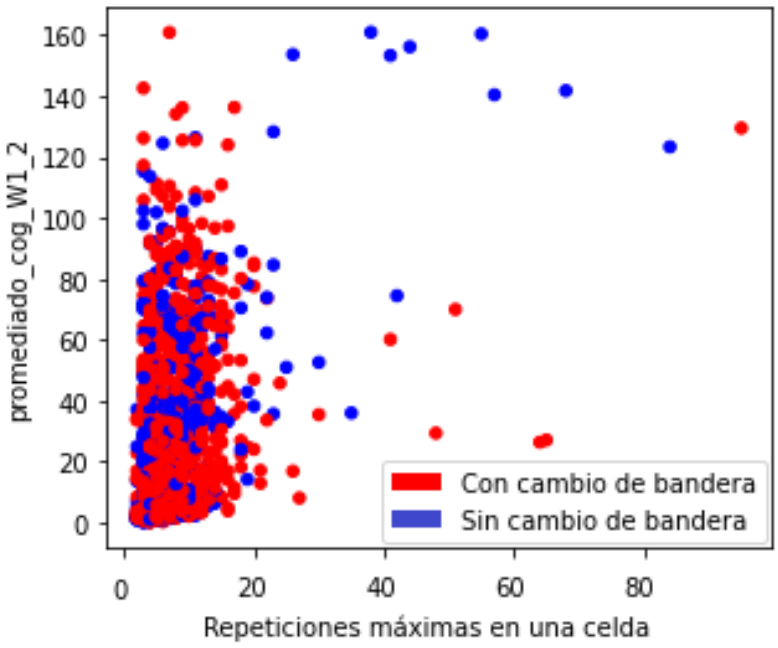

Figura 3-36 Buques mercantes dentro de la ZEE en función del número máximo de mensajes transmitidos en una celda y el "promediado_COG" máximo en esa celda

Para esta gráfica ocurre lo mismo que la anterior, por lo que a la vista de estos resultados obtenidos, es difícil utilizar el promediado_COG para distinguir a los buques que realizan siempre la misma ruta. La razón lógica por la que los buques mercantes con cambio de bandera tienen valores más pequeños de promediado COG, podría ser que estos buques, al ser más internacionales por cambiar de bandera, son más grandes. Es decir, un buque que cambia de pabellón, suele ser un buque que realiza viajes transoceánicos, por lo tanto, más grande que la media. Al ser buques grandes, realizan menos cambios de rumbo, ya que son menos maniobrables y tienen un gran desplazamiento.

A continuación, se procede a comparar los buques sospechosos con el resto de los buques obtenidos por el algoritmo con el objetivo de encontrar algún parámetro característico de los buques sospechosos que se diferencie con el resto (Anexo VII: Análisis de parámetros indicadores de actividades sospechosas de los buques detectados). Para ello primero se realiza de nuevo el análisis del promediado_COG_W1. Se realiza una gráfica en la se representan todos los buques detectados por el algoritmo en función de su "promediado_COG" máximo y del número máximo de mensajes transmitidos en una celda (véase Figura 3-37).

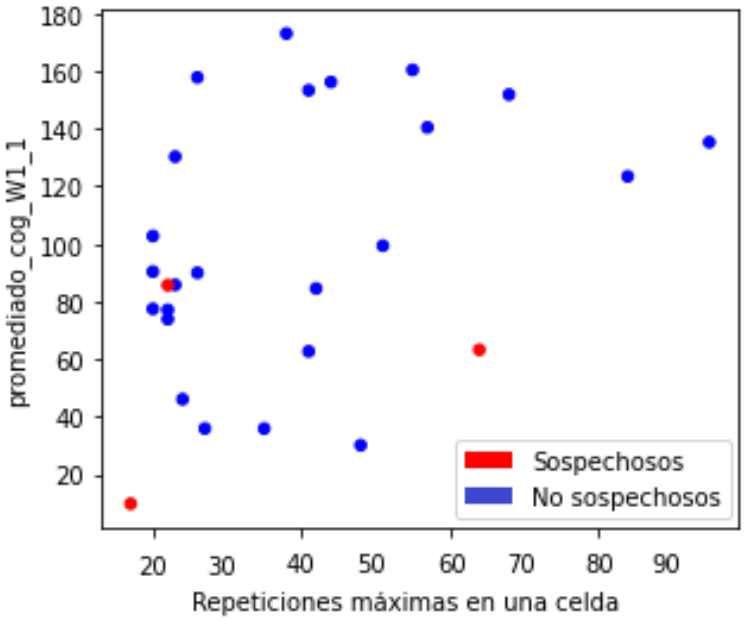

Figura 3-37 Buques detectados en función de su "promediado_COG" máximo y del número máximo de mensajes transmitidos en una celda

Se visualiza que no existe relación en esta gráfica para diferenciar a los buques sospechosos de los que no. Se procede a visualizar una gráfica parecida, esta vez representando los buques detectados por el algoritmo en función del número máximo de mensajes transmitidos en una celda y el "promediado_COG" máximo en esa celda (véase Figura 3-38).

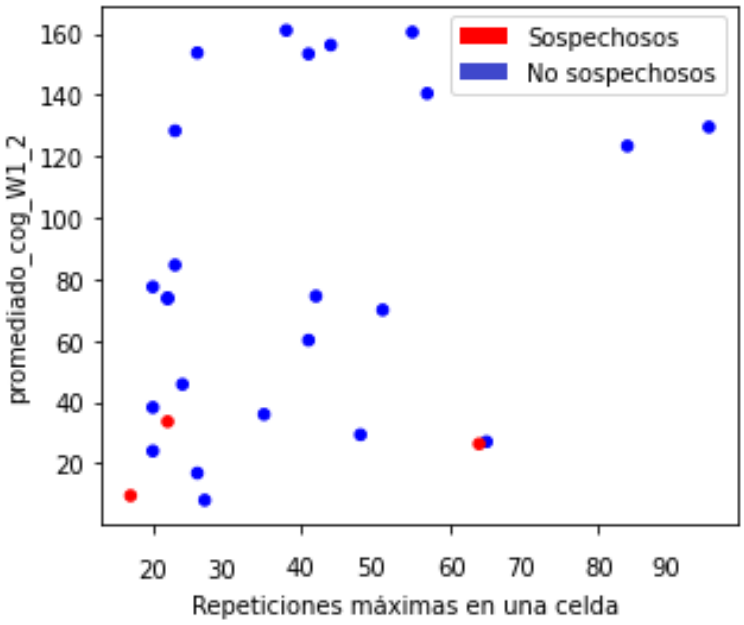

Figura 3-38 Buques detectados en función del número máximo de mensajes transmitidos en una celda y el "promediado_COG" máximo en esa celda

Con esta gráfica de dispersión, se puede ver que el "promediado_COG_W1" máximo de la celda que más se repite en los buques sospechosos tiene un valor más pequeño que el resto al no sobrepasar el valor de 40. Sin embargo, tampoco es un diferenciador de actividades sospechosas.

A continuación, se procede a analizar la velocidad de los buques en relación con los barcos sospechosos. Se realiza una gráfica representando cada buque detectado en función de la velocidad media y del número máximo de mensajes transmitidos en una celda (véase Figura 3-39) con el objetivo de analizar si es un parámetro característico que haga que se diferencien los buques sospechosos del resto.

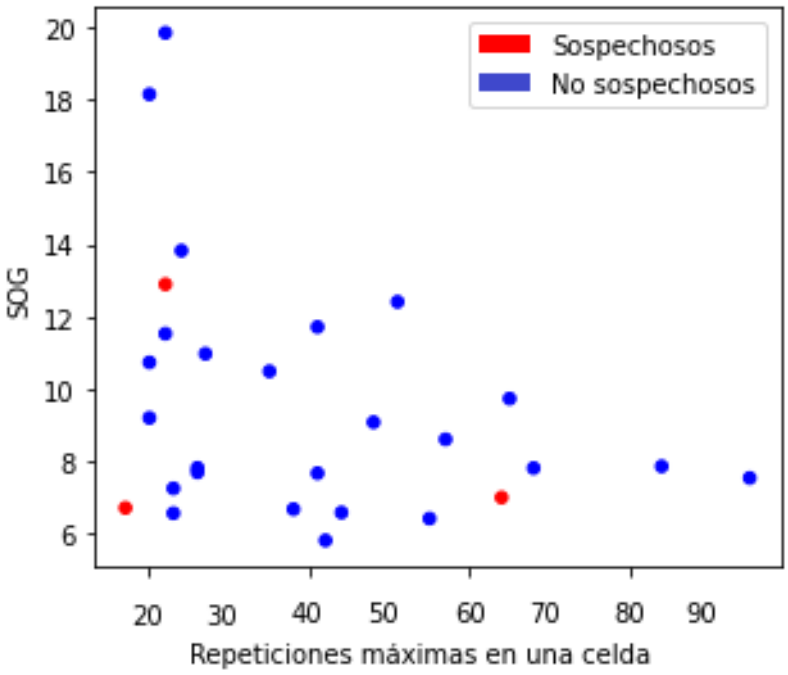

Figura 3-39 Buques detectados en función de su velocidad media y del número máximo de mensajes transmitidos en una celda

Se visualiza que no existe relación en esta gráfica para diferenciar a los buques sospechosos de los que no. Se procede a visualizar una gráfica parecida, esta vez representando los buques detectados por el algoritmo en función del número máximo de mensajes transmitidos en una celda y la velocidad media en esa celda (véase Figura 3-40).

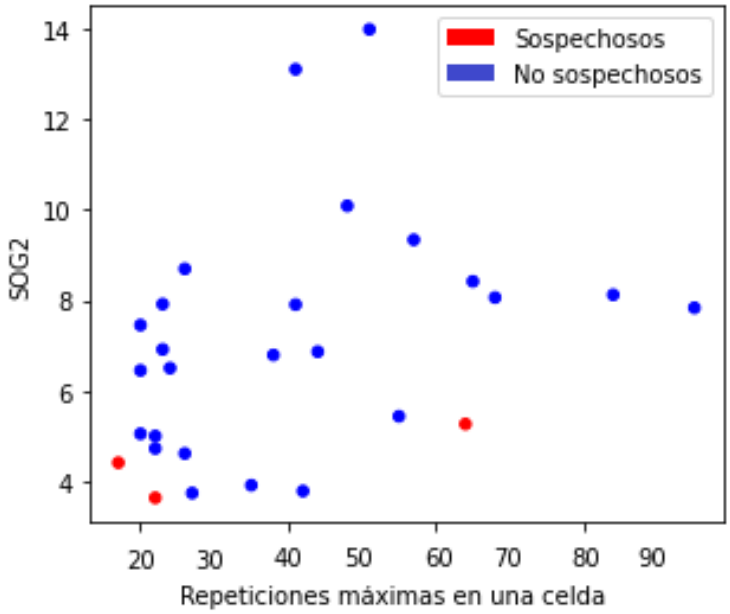

Figura 3-40 Buques detectados en función del número máximo de mensajes transmitidos en una celda y la velocidad media en esa celda

Se aprecia que dos de los buques sospechosos se encuentran en la esquina inferior izquierda, lo que significa que un buque que transmite pocos mensajes a una velocidad media baja tiene más posibilidad de ser sospechoso. Sin embargo, no se considera un buen parámetro indicador, debido a que no tiene justificación y puede ser fruto de la casualidad.

Con los resultados obtenidos, no se ha detectado ningún parámetro más que permita distinguir los buques sospechosos. La razón podría ser que la etiqueta de buque sospechoso, se pone a juicio del autor, basándose en su conocimiento marítimo y habiendo revisado el estado del arte. Por lo que no es sencillo implementar el etiquetado de forma automática.

4 CONCLUSIONES Y LÍNEAS FUTURAS

La adaptación de las nuevas tecnologías 4.0 al ámbito del Conocimiento del Entorno Marítimo es necesaria. Las oportunidades que ofrece la I4.0 han de ser explotadas con eficiencia. La Armada se ha visto necesitada de herramientas *Big Data* que faciliten la gestión de la información tan turbulenta como es la del entorno marítimo. Para ello, con este TFG se ha pretendido contribuir a esta adaptación.

La principal conclusión extraída es que es posible la explotación del *Big Data* marítimo para ampliar el CEM. Más concretamente, se ha demostrado que es viable la aplicación de nuevas tecnologías para obtener resultados de forma automática y en tiempo real. Teniendo en cuenta que el presente trabajo ha incluido información obtenida por el AIS, es importante poner en perspectiva la gran cantidad de opciones que pueden existir con la extrapolación de esta adaptación para un centro de gestión de información como es el COVAM, que recibe información de múltiples fuentes de información.

4.1 Revisión de los objetivos

Finalizado el trabajo, se puede concluir que el objetivo principal de identificar actividades sospechosas de los buques en tiempo real mediante datos AIS, concretamente, desarrollando un algoritmo que indicar actividades sospechosas, a partir de datos indirectos como la cinemática, zona de actividad, el tipo de barco o datos registrales, se ha cumplido satisfactoriamente.

El resultado principal obtenido es que, analizando cada uno de ellos dentro de la Zona Económica Exclusiva en el periodo de 15 días (desde el 18 de mayo al 01 de junio de 2021), de los 28 barcos detectados por el algoritmo, 3 de ellos resultan ser sospechosos, lo que supone un 10.71% del total de barcos detectados. Destacar que los 3 barcos, han realizado cambios de bandera, por lo que representan un 21.42% de barcos detectados por el algoritmo que han realizado cambios de bandera, 14 en total. Además, se ha demostrado que dicho algoritmo es implementable en tiempo real sin necesidad de consultar todo el histórico. Por otro lado, se ha demostrado que la repetición de celdas puede ser un indicador de actividades sospechosas. Por el contrario, la velocidad media o el promedio del COG de los buques, no parecen relevantes para distinguir los barcos realmente sospechosos.

Adicionalmente, para alcanzar el objetivo principal, se han cumplido los objetivos específicos y complementarios que se han planteado. Para el éxito del trabajo, se ha llevado a cabo una revisión del estado del arte y trabajos relacionados que ha posibilitado el desarrollo del algoritmo. También se ha realizado un análisis de los campos que puedan ser indicadores de actividades sospechosas, así como el de los umbrales a seleccionar, justificando así, su empleo. Resaltar que, como resultado de analizar la posibilidad de discriminar los buques que realizan continuamente la misma ruta, no se esperaba obtener que los buques con cambio de bandera cambian menos de rumbo que los que si lo han hecho.

A título personal, el desarrollo de este trabajo me ha sido de gran utilidad para familiarizarme con herramientas de tecnología *Big Data* empleando lenguaje de programación Python.

4.2 Líneas futuras

Una vez demostrada la posibilidad de detectar actividades sospechosas de buques en tiempo real, quedan abiertas una infinidad de oportunidades mediante la explotación del entorno marítimo. Es por ello, que se proponen las siguientes líneas futuras como propuestas de continuación de este trabajo:

- Implementación de un algoritmo aprendizaje automático supervisado para la detección de actividades sospechosas de buques en tiempo real mediante datos AIS. Para ello, sería de gran utilidad disponer de datos etiquetados.
- Analizar la posibilidad de discriminar los buques que hagan siempre la misma ruta en tiempo real mediante el análisis de las celdas visitadas y la secuencia de las mismas.
- Analizar la manera de detectar y calcular la zona habitual de un buque en tiempo real a partir de los mensajes AIS mediante estadísticos online (por ejemplo, mediante centroides con su dispersión).
- Análisis profundo de los comportamientos que realizan los buques incautados e incorporación de otros indicadores (zona habitual, etc.) con el objetivo de poder detectar más patrones de comportamiento.
- Detección y análisis de las zonas de espera de los buques antes de entrar en puerto mediante un algoritmo de agrupamiento.

5 BIBLIOGRAFÍA

[1] «Comparativa de la Industria 4.0 entre España y el resto de países | Atria», *ATRIA Innovation*, 7 de julio de 2020. [En línea]. Disponible en: https://www.atriainnovation.com/comparativa-industria-4-0-espana-y-resto-paises/ (accedido 19 de enero de 2022).

[2] Ministerio de Defensa, «Plan de acción del Ministerio de Defensa para la transformación digital». [En línea]. Disponible en: https://publicaciones.defensa.gob.es/media/downloadable/files/links/t/r/transformaci_n_digial_minisdef.pdf (accedido 19 de enero 2022).

[3] «cv_jemad_tlc_es.pdf». [En línea]. Disponible en: https://emad.defensa.gob.es/Galerias/emad/files/cv_jemad_tlc_es.pdf (accedido 19 de enero de 2022).

[4] Centro de Operaciones y Vigilancia de Acción Marítima, «REF F_DOCUMENTO COVAM ANOMALIAS PATRONES .pdf».

[5] «efficiensea_wp4_13.pdf». [En línea]. Disponible en: http://efficiensea.org/files/mainoutputs/wp4/efficiensea_wp4_13.pdf (accedido 29 de enero de 2022).

[6] «Guia docente 2021_22 Centro Universitario de la Defensa de la Escuela Naval Militar de Marín». [En línea]. Disponible en: https://secretaria.uvigo.gal/docnet-nuevo/guia_docent/index.php?ensenyament=P52G381V01&assignatura=P52G381V01991&fitxa_apartat=1&idioma=cast&idioma_assig=cast&idioma_assig=cast (accedido 5 de marzo de 2022).

[7] «Las 10 V´s del Big Data | Grupo Novatech». [En línea]. Disponible en: https://www.grupo-novatech.com/las-10-vs-del-big-data/ (accedido 8 de febrero de 2022).

[8] «Qué es Big Data | Universidad Complutense de Madrid», *Master Big Data*, 21 de febrero de 2018.[En línea]. Disponible en: https://www.masterbigdataucm.com/que-es-big-data/ (accedido 8 de febrero de 2022).

[9] «Data Warehouse y Data Lake: ¿Qué son? | Salesforce - Blog de Salesforce».[En línea]. Disponible en: https://www.salesforce.com/mx/blog/2020/10/data-warehouse-y-data-lake.html (accedido 8 de febrero de 2022).

[10] «¿Qué es Big Data Analytics y para qué sirve?», *IMNOVATION*.[En línea]. Disponible en: https://www.imnovation-hub.com/es/transformacion-digital/que-es-el-big-data-analytics-la-datificacion-de-la-sociedad/ (accedido 23 de febrero de 2022).

[11] Minewiskan, «Conceptos de minería de datos». [En línea]. Disponible en: https://docs.microsoft.com/es-es/analysis-services/data-mining/data-mining-concepts (accedido 23 de febrero de 2022).

[12] por I. S. Education, «Influencia del Big Data en los modelos estadísticos», *Blog de Tecnología - IMF Smart Education*, 23 de septiembre de 2020. [En línea]. Disponible en: https://blogs.imf-

formacion.com/blog/tecnologia/influencia-big-data-en-modelos-estadisticos-202009/ (accedido 23 de febrero de 2022).

[13] «LOS 4 TIPOS DE DATA ANALYTICS». [En línea]. Disponible en: https://www.veconinter.com/blog/post/los-4-tipos-de-data-analytics/209 (accedido 23 de febrero de 2022).

[14] «¿Qué es la Inteligencia Artificial? - Iberdrola». [En línea]. Disponible en: https://www.iberdrola.com/innovacion/que-es-inteligencia-artificial (accedido 10 de febrero de 2022).

[15] 21a2514a1e, «Conociendo las Redes Neuronales con Diego Calvo», *Bravent*, 12 de junio de 2019. [En línea]. Disponible en: https://www.bravent.net/conociendo-las-redes-neuronales-con-diego-calvo/ (accedido 10 de febrero de 2022).

[16] «Conceptos-IA-Machine-Learning-y-Deep-Learning-980x490.png.webp (980×490)». [En línea]. Disponible en: https://www.masterdatascienceucm.com/wp-content/uploads/2020/12/Conceptos-IA-Machine-Learning-y-Deep-Learning-980x490.png.webp (accedido 10 de febrero de 2022).

[17] «¿Qué es el Machine Learning? | Aprendizaje Automático | UCM», *Máster Data Science*, 18 de diciembre de 2020. [En línea]. Disponible en: https://www.masterdatascienceucm.com/que-es-machine-learning/ (accedido 10 de febrero de 2022).

[18] «Aprendizaje batch y online | Interactive Chaos». [En línea]. Disponible en: https://interactivechaos.com/es/manual/tutorial-de-machine-learning/aprendizaje-batch-y-online (accedido 19 de febrero de 2022).

[19] «Menares - EL MDA Y SU RELACIÓN CON LA ESTRATEGIA MARÍTIMA.pdf». [En línea]. Disponible en: https://revistamarina.cl/revistas/2010/6/arenas.pdf (accedido 26 de enero de 2022).

[20] «HSPD_MDAPlan_0.pdf». [En línea]. Disponible en: https://www.dhs.gov/sites/default/files/publications/HSPD_MDAPlan_0.pdf (accedido 26 de enero de 2022).

[21] «ENCOMAR - Entorno Colaborativo Marítimo de la Armada». [En línea]. Disponible en: https://encomar.covam.es/ (accedido 26 de enero de 2022).

[22] A. Española, «Fuerza de Acción Marítima - Presentación - Organización - Armada Española - Ministerio de Defensa - Gobierno de España». [En línea]. Disponible en: https://armada.defensa.gob.es/ArmadaPortal/page/Portal/ArmadaEspannola/conocenosorganizacion/prefLang-es/03Flota--04FAM (accedido 27 de enero de 2022).

[23] A. Española, «Otros - Escudos Oficiales de la Armada - Galería Fotográfica - Armada Española - Ministerio de Defensa - Gobierno de España». [En línea]. Disponible en: https://armada.defensa.gob.es/ArmadaPortal/page/Portal/ArmadaEspannola/multimediagaleria/prefLang-es/21escudosoficiales--03otros-es?_pageNum=4&paramNo= (accedido 27 de enero de 2022).

[24] N. Gibraltar, «El Centro de Operaciones y Vigilancia de Acción Marítima comienza su modernización», *Noticias Gibraltar*, 15 de enero de 2020. [En línea]. Disponible en: https://noticiasgibraltar.es/campo-gibraltar/defensa/centro-operaciones-y-vigilancia-accion-maritima-comienza-su-modernizacion (accedido 27 de enero de 2022).

[25] Defensa.com, «Centro de operaciones y vigilancia de acción marítima de la Armada - Noticias Defensa En abierto», *Defensa.com*, 8 de septiembre de 2017. [En línea]. Disponible en: https://www.defensa.com/en-abierto/centro-operaciones-vigilancia-accion-maritima-armada (accedido 27 de enero de 2022).

[26] M. de la Gándara García, «EL COVAM DE LA ARMADA AL SERVICIO DE LA COMUNIDAD MARÍTIMA».[En línea]. Disponible en: https://armada.defensa.gob.es/ArmadaPortal/ShowPropertyServlet?nodePath=/BEA%20Repository/Desktops/Portal/ArmadaEspannola/Pages/conocenosespeciales/05actividades/99pirateria/02COVAMIEE/02COVAMIEE-es/argCOVAMIEEES (accedido 27 de enero de 2022).

[27] «programa-CEM.pdf». [En línea]. Disponible en: https://www.defensa.gob.es/Galerias/dgamdocs/programa-CEM.pdf (accedido 26 de enero de 2022).
[28] «International Maritime Organization - 2001 - SOLAS edicin refundida del Convenio Internacion.pdf». [En línea]. Disponible en: http://www.bioscafire.com/upfiles/normativa/solas.pdf (accedido 27 de enero de 2022).
[29] «Seguridad AIS - Para qué sirve». [En línea]. Disponible en: https://www.azimutmarine.es/seguridad-ais (accedido 28 de enero de 2022).
[30] «AIS Dataset - AIS Data - VT Explorer». [En línea]. Disponible en: https://api.vtexplorer.com/docs/response-ais.html (accedido 28 de enero de 2022).
[31] «oes44900a_fa170.pdf». [En línea]. Disponible en: https://www.furunousa.com/-/media/sites/furuno/document_library/documents/manuals/public_manuals/oes44900a_fa170.pdf (accedido 28 de enero de 2022).
[32] F. Heymann, T. Noack, y P. Bany, «Plausibility analysis of navigation related AIS parameter based on time series». [En línea]. Disponible en: https://elib.dlr.de/84166/1/ENC2013_Plausibility_analysis_of_navigation_related_AIS_parameter_based_on_time_series_final.pdf (accedido 19 de febrero de 2022).
[33] T. Emmens, C. Amrit, A. Abdi, y M. Ghosh, «The promises and perils of Automatic Identification System data», *Expert Systems with Applications*, vol. 178. [En línea]. Disponible en: https://n9.cl/9v0o7 (accedido 19 de febrero de 2022).
[34] I. Lytra, M.-E. Vidal, F. Orlandi, y J. Attard, «A big data architecture for managing oceans of data and maritime applications», jun. 2017. [En línea]. Disponible en: https://www.researchgate.net/profile/Fabrizio-Orlandi/publication/322997369_A_big_data_architecture_for_managing_oceans_of_data_and_maritime_applications/links/5b8968164585151fd1402227/A-big-data-architecture-for-managing-oceans-of-data-and-maritime-applications.pdf (accedido 8 de febrero de 2022).
[35] Vereinte Nationen, «The globalization of crime a transnational organi.pdf ». [En línea] Disponible en: https://www.unodc.org/documents/data-and-analysis/tocta/TOCTA_Report_2010_low_res.pdf (accedido 28 febrero de 2022).
[36] O. B. Bergadà, «ESTUDIO DE LAS ACTIVIDADES ILEGALES EN EL TRANSPORTE MARÍTIMO». [En línea]. Disponible en: https://upcommons.upc.edu/bitstream/handle/2099.1/13721/Oriol%20berges.pdf (accedido 28 de febrero de 2022).
[37] «Detalle nota de prensa. Policía Nacional España.». [En línea]. Disponible en: https://www.policia.es/_es/comunicacion_prensa_detalle.php?ID=10002 (accedido 28 de febrero de 2022).
[38] «¿Qué es Elasticsearch? Un motor de búsqueda y análisis», *ITDO Desarrollo web y APPs Barcelona*, 16 de enero de 2020. [En línea]. Disponible en: https://www.itdo.com/blog/que-es-elasticsearch-un-motor-de-busqueda-y-analisis/ (accedido 4 de febrero de 2022).
[39] «Redis: almacén de datos en memoria. Cómo funciona y por qué utilizarlo», *Amazon Web Services, Inc.* [En línea]. Disponible en: https://aws.amazon.com/es/redis/ (accedido 4 de marzo de 2022).
[40] I. Brodsky, «H3: Uber's Hexagonal Hierarchical Spatial Index», *Uber Engineering Blog*, 27 de junio de 2018. [En línea]. Disponible en: https://eng.uber.com/h3/ (accedido 24 de febrero de 2022).
[41] «Legislació Marítima». [En línea]. Disponible en: http://www.catalonia.org/dretmaritim/index_bloque_03_Fin_C2.html (accedido 24 de febrero de 2022).
[42] «Welcome to Python.org», *Python.org*. [En línea]. Disponible en: https://www.python.org/ (accedido 4 de febrero de 2022).

[43] «¿Qué es Python? El lenguaje de programación del 2021», *https://www.crehana.com*. [En línea]. Disponible en: https://www.crehana.com/es/blog/desarrollo-web/que-es-python/ (accedido 1 de febrero de 2022).
[44] «Infografía: Los lenguajes de programación más usados del mundo», *Statista Infografías*. [En línea]. Disponible en: https://es.statista.com/grafico/16580/lenguajes-de-programacion-mas-usados-del-mundo/ (accedido 4 de febrero de 2022).
[45] Israel, «Python vs R para Data Science», *Viewnext*. [En línea]. Disponible en: https://www.viewnext.com/python-vs-r-para-data-science/ (accedido 4 de febrero de 2022).
[46] A. S. Alberca, «La librería Numpy», *Aprende con Alf*. [En línea]. Disponible en: https://aprendeconalf.es/docencia/python/manual/numpy/ (accedido 4 de febrero de 2022).
[47] A. S. Alberca, «La librería Pandas», *Aprende con Alf*. [En línea]. Disponible en: https://aprendeconalf.es/docencia/python/manual/pandas/ (accedido 4 de febrero de 2022).
[48] «Project Jupyter». [En línea]. Disponible en: https://jupyter.org (accedido 22 de febrero de 2022).
[49] «Kepler.gl de Mapbox: mapas interactivos y visualización de datos», *geomapik*, 13 de abril de 2020. [En línea]. Disponible en: http://www.geomapik.com/webmapping-gis/kepler-gl-mapbox-mapas-interactivos-spatial-data/ (accedido 4 de febrero de 2022).
[50] Y. Guan *et al.*, «Identification of Fishing Vessel Types and Analysis of Seasonal Activities in the Northern South China Sea Based on AIS Data: A Case Study of 2018», *Remote Sensing*, vol. 13. [En línea]. Disponible en: https://www.mdpi.com/2072-4292/13/10/1952/pdf (accedido 19 de febrero de 2022).
[51] F. Mazzarella, M. Vespe, D. Damalas, y G. Osio, «Discovering vessel activities at sea using AIS data: Mapping of fishing footprints», en *17th International Conference on Information Fusion.* [En línea]. Disponible en: https://ieeexplore.ieee.org/abstract/document/6916045/authors (accedido 19 de febrero de 2022).
[52] A. Pisabarro, «Banderas de conveniencia, la patria de los océanos», *El Orden Mundial - EOM*, 9 de diciembre de 2018. [En línea]. Disponible en: https://elordenmundial.com/banderas-de-conveniencia-la-patria-de-los-oceanos/ (accedido 14 de marzo de 2022).
[53] «Ship EUROCARGO CAGLIARI (Ro-Ro Cargo) Registered in Italy - Vessel details, Current position and Voyage information - IMO 9471068, MMSI 247318900, Call Sign ICMR | AIS Marine Traffic», *MarineTraffic.com*. [En línea]. Disponible en: https://www.marinetraffic.com/en/ais/details/ships/shipid:281570/mmsi:247318900/imo:9471068/vessel:EUROCARGO CAGLIARI (accedido 2 de marzo de 2022).
[54] «El "Miramar Express", de parada técnica en la Bahía de Algeciras», *El Estrecho Digital*, 16 de febrero de 2019. [En línea]. Disponible en: https://www.elestrechodigital.com/2019/02/17/el-miramar-express-de-parada-tecnica-en-la-bahia-de-algeciras/ (accedido 2 de marzo de 2022).
[55] L. Alonso, «CMA CGM suspende servicios en puertos de Ucrania debido a maniobras militares de Rusia», *PortalPortuario*, 24 de febrero de 2022. [En línea]. Disponible en: https://portalportuario.cl/cma-cgm-suspende-servicios-en-puertos-de-ucrania-debido-a-maniobras-militares-de-rusia/ (accedido 12 de marzo de 2022).
[56] «NORTH STAR GLORY isimli gemi, Çanakkale Boğazı'nda arızalandı», *Deniz Haber*, 27 de diciembre de 2021. [En línea]. Disponible en: https://www.denizhaber.net/north-star-glory-isimli-gemi-canakkale-bogazinda-arizalandi-haber-106012.htm (accedido 12 de marzo de 2022).
[57] «La línea Ibiza-Formentera ha estado casi 4 horas colapsada por un camión atrapado al desembarcar en La Savina», *Noudiari.es*, 16 de septiembre de 2021. [En línea]. Disponible en: https://www.noudiari.es/local-ibiza/la-linea-ibiza-formentera-ha-estado-casi-4-horas-colapsada-por-un-camion-atrapado-en-una-rampa-de-desembarco-en-la-savina/ (accedido 3 de marzo de 2022).
[58] CONFIDENCIAL
[59] CONFIDENCIAL

[60] «África importa el 95% de sus armas para la guerra – Rebelion». [En línea]. Disponible en: https://rebelion.org/africa-importa-el-95-de-sus-armas-para-la-guerra/ (accedido 12 de marzo de 2022).
[61] «Una operación fronteriza en África Occidental pone al descubierto casos de tráfico ilícito de personas, de lingotes de oro y de fármacos falsos». [En línea]. Disponible en: https://www.interpol.int/es/Noticias-y-acontecimientos/Noticias/2019/Una-operacion-fronteriza-en-Africa-Occidental-pone-al-descubierto-casos-de-trafico-ilicito-de-personas-de-lingotes-de-oro-y-de-farmacos-falsos (accedido 12 de marzo de 2022).
[62] CONFIDENCIAL
[63] CONFIDENCIAL
[64] CONFIDENCIAL
[65] CONFIDENCIAL

ANEXO I: CAMPOS DE LA BASE DE DATOS

#	Column	Non-Null Count	Dtype
0	_index	23 non-null	object
1	_type	23 non-null	object
2	_id	23 non-null	object
3	_score	23 non-null	float64
4	@timestamp	23 non-null	int64
5	timecard	23 non-null	object
6	mmsi	23 non-null	object
7	imo	23 non-null	object
8	tipo_mensaje	23 non-null	object
9	eslora	23 non-null	object
10	posicion.lat	23 non-null	float64
11	posicion.lon	23 non-null	float64
12	celda_h3_6	23 non-null	object
13	celda_h3_7	23 non-null	object
14	celda_h3_9	23 non-null	object
15	celda_h3_9_octal	23 non-null	object
16	macro	23 non-null	object
17	micro	23 non-null	object
18	cog	23 non-null	float64
19	sog	23 non-null	float64
20	nombre	23 non-null	object
21	destino	23 non-null	object
22	operador	23 non-null	object
23	bandera	23 non-null	object
24	año	23 non-null	object
25	tipo_buque_AIS	23 non-null	object
26	tipo_buque_SD	23 non-null	object
27	clave_foranea	23 non-null	object
28	es_pesquero	23 non-null	bool
29	es_carguero	23 non-null	bool
30	es_militar	23 non-null	bool
31	es_recreo	23 non-null	bool
32	es_español	23 non-null	bool
33	cambios_de_bandera	23 non-null	bool
34	parado	23 non-null	bool
35	dentro_territorio_español	23 non-null	bool
36	sobre_cableado	23 non-null	bool
37	dentro_ZEE	23 non-null	bool
38	mmsi_duplicado	23 non-null	bool
39	id_sub_ruta	23 non-null	object
40	dst	23 non-null	object
41	promediado_cog_W1	23 non-null	float64
42	promediado_cog_W2	23 non-null	object
43	promediado_sog	23 non-null	object
44	anomalia1	23 non-null	object
45	anomalia3	23 non-null	object
46	anomalia4	23 non-null	object
47	anomalia5	23 non-null	object
48	anomalia6	23 non-null	object
49	anomalia7	23 non-null	object
50	anomalia8	23 non-null	object
51	anomalia9	23 non-null	object
52	anomalia10	23 non-null	object

```
 53  anomalia11                 23 non-null     object
 54  anomalia12                 23 non-null     object
dtypes: bool(11), float64(6), int64(1), object(37)
```

ANEXO II: VISUALIZACIÓN DERROTAS DE LOS BUQUES

```
import sys
from elasticsearch import Elasticsearch, helpers, exceptions
import ssl
import pandas as pd
from pandasticsearch import Select
import time
import h3
from keplergl import KeplerGl

# CONFIGURARACIÓN
# Nombre del indice que se quiere utilizar
_index_name="ais_data_1805_to_0206_def"

# Datos de conexion de Elasticsearch
_host="192.168.16.20"
_port=9200
_http_auth=('cemai_admin', 'cemai4dmin21')
_timeout=30
_size=10000
# Se inicia la conexion con Elasticsearch
context = ssl.create_default_context(cafile="elasticsearch-ca.pem")
context.check_hostname = False
context.verify_mode = ssl.CERT_NONE
elastic = Elasticsearch([{'host': _host, 'port': _port}], http_auth=_http_auth,
scheme="https", ssl_context=context, timeout=_timeout)
#Consulta a la base de datos del barco con MMSI deseado
     results = elastic.search(index=_index_name, body={"query": { "bool": { "must":
                    [{"term":#Se introduce el MMSI deseado{"mmsi": "636092098"}}]}},
                                                          #Se ordenan los mensajes por
                                                                  orden tmeporal
                                                        "sort": [{"mmsi": {"order":
                                              "desc"}}, {"@timestamp": {"order": "asc"}}],
                                                                "size": _size})
#Se almacenan los resultados en un Dataframe
df = Select.from_dict(results).to_pandas()
#Se representa en Kepler
map = KeplerGl(height=400, width=600)
map.add_data(data=df, name="prueba")
map
```

ANEXO III: OBTENCIÓN LISTA BARCOS ANÓMALOS MEDIANTE CONSULTA HISTÓRICA

```
import sys
from elasticsearch import Elasticsearch
import ssl
import pandas as pd
from pandasticsearch import Select
import numpy as np
from scipy import stats
# CONFIGURARACION
# Nombre del indice que se quiere utilizar
_index_name="ais_data_1805_to_0206_def"
# Datos de conexion de Elasticsearch
_host="192.168.16.20"
_port=9200
_http_auth=('cemai_admin', 'cemai4dmin21')
_timeout=30
_type="MERCANTE"
# Numero de resultados
_size=10000
# Se inicia la conexion con Elasticsearch
context = ssl.create_default_context(cafile="elasticsearch-ca.pem")
context.check_hostname = False
context.verify_mode = ssl.CERT_NONE
elastic = Elasticsearch([{'host': _host, 'port': _port}], http_auth=_http_auth,
scheme="https", ssl_context=context, timeout=_timeout)

#Se realiza la consulta propiamente dicha
results = elastic.search(index=_index_name, body={"size": 0,
                                                  #Se filtra la consulta con los
                                                          parámetros descritos
                                                    "query": {"bool": { "must":
                                                [{"term": { "dentro_ZEE": True}},
                                              {"term": {"tipo_buque_AIS": _type}},
                                                 {"range": {"sog": {"gt": 2}}}]}},
                                                     #Se agrupa los resultados con
                                                          agregación, por MMSI
                                                   "aggs": {"cells": {"composite":
                                                           {"sources": [{ "cells":
                                                 { "terms": { "field": "mmsi"}}}],
                                                             "size" : 10000}}}})
#Se guardan los resultados en un Dataframe de Pandas
df = pd.json_normalize(results['aggregations']['cells']['buckets'])

#Se vuelve a hacer la consulta para los datos que no se hayan consultado, ya que,
Elastic limita el tamaño a 10000 resultados
while len(results['aggregations']['cells']['buckets']):
    after_key = results['aggregations']['cells']['after_key']
    results = elastic.search(index=_index_name, body={"size": 0,
                                                        "query": {"bool": { "must":
                                                    [{"term": { "dentro_ZEE": True}},
                                                  {"term": {"tipo_buque_AIS": _type}},
                                                     {"range": {"sog": {"gt": 2}}}]}},
```

```
                                "aggs": {"cells": {"composite": {"sources": [{ "cells":
                                                    { "terms": { "field": "mmsi"}}}],
                                                                "size" : 10000,
                                                            "after" : after_key}}}})
    #Añadimos los resultados al Dataframe ya creado
    if len(results['aggregations']['cells']['buckets']):
        df2 = pd.json_normalize(results['aggregations']['cells']['buckets'])
        df=df.append(df2,ignore_index=True,sort=False)



suspect=[]
#Se realiza una consulta para cada mmsi y se agrupan los mensajes en todas las
celdas en las que se transmiten
for i in df.index:
    h=df.loc[i,'key.cells']
      results = elastic.search(index=_index_name, body={"query": {"bool": { "must":
                                                        [{"term": { "mmsi":h }},

                                                     {"term": { "dentro_ZEE": True}},

                                                    {"range": {"sog": {"gt": 2}}}]}},
                                                           #Se agrupan los resultados
                                                           con agregación, por celdas
                                                                "aggs": {"cells":
                                                  {"composite": {"sources": [{"cells":
                                                 {"terms": {"field": "celda_h3_6"}}}],
                                                                            "size" :
                                                                          10000}}}})
    #Se almacenan los resultados de la agregación en un Dataframe
    df2 = pd.json_normalize(results['aggregations']['cells']['buckets'])

    #Se almacenan en una lista todos los buques que transmitan más de 19 mensajes
en una celda
    df2["mmsi"]=h
    for p in df2.index:
        if df2.loc[p,"doc_count"]>19:

suspect.append([df2.loc[p,"mmsi"],df2.loc[p,"doc_count"],[df2.loc[p,"key.cells"]]]
)
#La lista generada, se convierte en un Dataframe
suspect=pd.DataFrame(suspect, columns=['mmsi',"doc_count","key.cells"])

#Se eliminan barcos repetidos y se genera lista de barcos anómalos
dfs=pd.DataFrame(suspect, columns=['mmsi'])
list_mmsi=[]
d=0
for i in dfs.index:
    q=dfs.loc[i,"mmsi"]
    for t in list_mmsi:
        if q==t:
            d=1
    if d==0:
        list_mmsi.append([dfs.loc[i,"mmsi"]])
    d=0
#Lista de barcos que hacen saltar la alarma
print(list_mmsi)
```

ANEXO IV: OBTENCIÓN LISTA BARCOS ANÓMALOS MEDIANTE ALMACENAMIENTO DE CELDAS

```
import sys
from elasticsearch import Elasticsearch
import ssl
import pandas as pd
from pandasticsearch import Select
# CONFIGURARACION
# Nombre del indice que se quiere utilizar
_index_name="ais_data_1805_to_0206_def"
# Datos de conexion de Elasticsearch
_host="192.168.16.20"
_port=9200
_http_auth=('cemai_admin', 'cemai4dmin21')
_timeout=30
_type="MERCANTE"
# Numero de resultados
_size=10000
# Se inicia la conexion con Elasticsearch
context = ssl.create_default_context(cafile="elasticsearch-ca.pem")
context.check_hostname = False
context.verify_mode = ssl.CERT_NONE
elastic = Elasticsearch([{'host': _host, 'port': _port}], http_auth=_http_auth,
scheme="https", ssl_context=context, timeout=_timeout)

#Se realiza la consulta propiamente dicha
results = elastic.search(index=_index_name, body={"size": 0,
                                                  #Se filtra la consulta con los
                                                          parámetros descritos
                                                     "query": {"bool": { "must":
                                                [{"term": { "dentro_ZEE": True}},
                                              {"term": {"tipo_buque_AIS": _type}},
                                                 {"range": {"sog": {"gt": 2}}}]}},
                                                      #Se agrupa los resultados con
                                                              agregación, por MMSI
                                    "aggs":{"cells":{"composite":{"sources": [{ "cells":
                                                 { "terms": { "field": "mmsi"}}}],
                                                                "size" : 10000}}}})
#Se guardan los resultados en un Dataframe de Pandas
df = pd.json_normalize(results['aggregations']['cells']['buckets'])

#Se vuelve a hacer la consulta para los datos que no se hayan consultado, ya que,
Elastic limita el tamaño a 10000 resultados
while len(results['aggregations']['cells']['buckets']):
    after_key = results['aggregations']['cells']['after_key']
    results = elastic.search(index=_index_name, body={"size": 0,
                                                        "query": {"bool": { "must":
                                                [{"term": { "dentro_ZEE": True}},
                                              {"term": {"tipo_buque_AIS": _type}},
                                                 {"range": {"sog": {"gt": 2}}}]}},
                                    "aggs":{"cells":{"composite":{"sources": [{ "cells":
                                                 { "terms": { "field": "mmsi"}}}],
                                              "size" : 10000, "after" : after_key}}}})
    #Añadimos los resultados al Dataframe ya creado
```

```
    if len(results['aggregations']['cells']['buckets']):
        df2 = pd.json_normalize(results['aggregations']['cells']['buckets'])
        df=df.append(df2,ignore_index=True,sort=False)


#Se realiza la consulta para cada MMSI obtenidos anteriormente
for i in df.index:
    h=df.loc[i,'key.cells']
    #Se realiza la consulta a la base de datos, se vuelve a introducir los
parámetros de restricción para hacerla ágil
                                    results = elastic.search(index=_index_name, body=
                                {"query": { "bool": { "must": [{"term": { "mmsi":h }},
                                                    {"term": { "dentro_ZEE": True}},
                                                      {"range": {"sog": {"gt": 2}}},
                                              {"term": { "tipo_buque_AIS": "MERCANTE"}}]}},
                                                                #Se ordenan los resultados
                                                             de cada barco en orden temporal
                                                                "sort": [{"mmsi": {"order":
                                                  "desc"}},{"@timestamp": {"order": "asc"}}],
                                                                "size": _size})
    #Se guardan los resultados en un Dataframe
    df2 = Select.from_dict(results).to_pandas()

    #Se almacenan las últimas 150 celdas
    celdas=[]
    s=0
    c=0
    anomalo=False
    for p in df2.index:
        if 2==2:
            if len(celdas)<150:
                celdas.append([df2.loc[p,"celda_h3_6"]])
            else:
                celdas[s]=df2.loc[p,"celda_h3_6"]
                s+=1
            if s==150:
                s=0
              #Se comprueba cada vez que se añade una celda, si se repite más de 19
                                                                            veces
            for t in celdas:
                if celdas.count(t)>19 and c==0:
                    anomalo=True
                    suspect.append([df2.loc[p,"mmsi"],[df2.loc[p,"celda_h3_6"]]])
                      #Una vez se genera una alarma para un barco, ya no se generan
                                        más para el mismo, para agilizar la consulta
                    c=1
#Se guardan la lista de barcos que hacen saltar la alarma en un Dataframe
suspect=pd.DataFrame(suspect, columns=['mmsi',"celda_h3_6"])
#Lista de barcos que hacen saltar la alarma
print(suspect)
```

ANEXO V: OBTENCIÓN LISTA BARCOS ANÓMALOS USANDO PROMEDIADO COG

```
import sys
from elasticsearch import Elasticsearch, helpers, exceptions
import ssl
import pandas as pd
from pandasticsearch import Select
import time
# CONFIGURAR
# Nombre del indice que se quiere utilizar
#="ais_data_1805_to_0206_corregido"
_index_name="ais_data_1805_to_0206_def"
# Datos de conexion de Elasticsearch
_host="192.168.16.20"
_port=9200
_http_auth=('cemai_admin', 'cemai4dmin21')
_timeout=30
_size=10000
_type="MERCANTE"
suspect=[]
# Se inicia la conexion con Elasticsearch
context = ssl.create_default_context(cafile="elasticsearch-ca.pem")
context.check_hostname = False
context.verify_mode = ssl.CERT_NONE
elastic = Elasticsearch([{'host': _host, 'port': _port}], http_auth=_http_auth,
scheme="https", ssl_context=context, timeout=_timeout)
#Se realiza la consulta propiamente dicha
    results = elastic.search(index=_index_name, body={"size": 0,"query": {"bool": {
                                            "must": [{"term": { "dentro_ZEE": True}},
                                                 {"term": {"tipo_buque_AIS": _type}},
                                                    {"range": {"sog": {"gt": 2}}}]}},
                                                        #Se agrupa los resultados con
                                                                agregación, por MMSI
                                                  "aggs": {"cells": {"composite":
                                                          {"sources": [{ "cells":
                                                    { "terms": { "field": "mmsi"}}}],
                                                                 "size" : 10000}}}})

#Se guarda los resultados en un Dataframe de Pandas
df = pd.json_normalize(results['aggregations']['cells']['buckets'])
#Se vuelve a hacer la consulta para los datos que no se hayan consultado, ya que,
Elastic limita el tamaño a 10000 resultados
while len(results['aggregations']['cells']['buckets']):
    after_key = results['aggregations']['cells']['after_key']
      results = elastic.search(index=_index_name, body={"size": 0,"query": {"bool":
                                        { "must": [{"term": { "dentro_ZEE": True}},
                                                 {"term": {"tipo_buque_AIS": _type}},
                                                    {"range": {"sog": {"gt": 2}}}]}},
                                                              "aggs": {"cells":
                                                        {"composite": {"sources": [
                                                                    { "cells": {
                                                  "terms": { "field": "mmsi"}}}],
                                                                 "size" : 10000,
```

```
                                                                    "after" :
                                                                after_key}}}})
    #Añadimos los resultados al Dataframe ya creado
    if len(results['aggregations']['cells']['buckets']):
        df2 = pd.json_normalize(results['aggregations']['cells']['buckets'])
        df=df.append(df2,ignore_index=True,sort=False)
#Se realiza la consulta para cada MMSI obtenidos anteriormente
for i in df.index:
    h=df.loc[i,'key.cells']
    #Se realiza la consulta a la base de datos, se vuelve a introducir los
parámetros de restricción para hacerla ágil
     results = elastic.search(index=_index_name, body={"query": { "bool": { "must":
                                                              [{"term": { "mmsi":h }},
                                                        {"term": { "dentro_ZEE": True}},
                                                          {"range": {"sog": {"gt": 2}}},
                                                {"term": { "tipo_buque_AIS": "MERCANTE"}}]}},
                                                          #Se ordenan los resultados
                                                        de cada barco en orden temporal
                                                            "sort": [{"mmsi": {"order":
                                                                        "desc"}},
                                                                 {"@timestamp":
                                                               {"order": "asc"}}],
                                                                "size": _size})
    #Se guardan los resultados en un Dataframe
    df2 = Select.from_dict(results).to_pandas()
    #Se almacenan las últimas 150 celdas
    celdas=[]
    s=0
    c=0
    anomalo=False
    df2['promediado_cog_W1'] = df2['promediado_cog_W1'].replace(',','.',
regex=True).astype(float)
    for p in df2.index:
            if len(celdas)<150:
                celdas.append([df2.loc[p,"celda_h3_6"]])
            else:
                celdas[s]=df2.loc[p,"celda_h3_6"]
                s+=1
            if s==150:
                s=0
            #Se comprueba cada vez que se añade una celda, si se repite más de 19
veces y si su promediado COG es mayor que el umbral deseado
            for t in celdas:
                if celdas.count(t)>19 and c==0 and
df2["promediado_cog_W1"].max()>50.0:
                    anomalo=True
                    suspect.append([df2.loc[p,"mmsi"],[df2.loc[p,"celda_h3_6"]]])
                    #Una vez se genera una alarma para un barco, ya no se generan
más para el mismo, para agilizar la consulta
                    c=1
#Se guardan la lista de barcos que hacen saltar la alarma en un Dataframe
suspect=pd.DataFrame(suspect, columns=['mmsi',"celda_h3_6"])
print(suspect)
```

ANEXO VI: ANÁLISIS DE PARÁMETROS INDICADORES DE ACTIVIDADES SOSPECHOSAS DE TODOS LOS MERCANTES

```
import sys
from elasticsearch import Elasticsearch
import ssl
import pandas as pd
from pandasticsearch import Select
import time
import matplotlib.pyplot as plt
# CONFIGURAR
# Nombre del indice que se quiere utilizar
_index_name="ais_data_1805_to_0206_def"
# Datos de conexion de Elasticsearch
_host="192.168.16.20"
_port=9200
_http_auth=('cemai_admin', 'cemai4dmin21')
_timeout=30
_type="MERCANTE"
output_list=[]
# Numero de resultados
_size=10000
# Se inicia la conexion con Elasticsearch
context = ssl.create_default_context(cafile="elasticsearch-ca.pem")
context.check_hostname = False
context.verify_mode = ssl.CERT_NONE
elastic = Elasticsearch([{'host': _host, 'port': _port}], http_auth=_http_auth,
scheme="https", ssl_context=context, timeout=_timeout)
results = elastic.search(index=_index_name, body={"size": 0,
                                                  "query": {"bool": { "must": [
                                                      {"term": { "dentro_ZEE":
                                                                 True}},
                                                    {"term": {"tipo_buque_AIS":
                                                                 _type}},
                                                      {"range": {"sog": {"gt":
                                                                 2}}}]}},
                                                "aggs": {"cells": {"composite":
                                                                 {"sources": [
                                                    { "mmsi": { "terms": {
                                                      "field": "mmsi"}}}],
                                                       "size" : 10000}}}})

df = pd.json_normalize(results['aggregations']['cells']['buckets'])
while len(results['aggregations']['cells']['buckets']):
    after_key = results['aggregations']['cells']['after_key']
    results = elastic.search(index=_index_name, body={"size": 0,
                                                      "query": {"bool": { "must":
                                                   [{"term": { "dentro_ZEE": True}},
                                                 {"term": {"tipo_buque_AIS": _type}},
                                                   {"range": {"sog": {"gt": 2}}}]}},
                                                           "aggs": {"cells":
                                                   {"composite": {"sources": [
                                                          { "mmsi": { "terms":
                                                       { "field": "mmsi"}}}],
                                                              "size" : 10000,
```

```
                                                                    "after" : after_key}}}})
    if len(results['aggregations']['cells']['buckets']):
        df2 = pd.json_normalize(results['aggregations']['cells']['buckets'])
        df=df.append(df2,ignore_index=True,sort=False)
for i in df.index:
    if df.loc[i,'doc_count']>20:
        h=df.loc[i,'key.mmsi']
             results = elastic.search(index=_index_name, body={"query": { "bool": {
                                                    "must": [{"term": { "mmsi":h }},
                                                    {"term": { "dentro_ZEE": True}},
                                                      {"range": {"sog": {"gt": 2}}},
                                            {"term": { "tipo_buque_AIS": "MERCANTE"}}]}},
                                                                "sort": [{"mmsi":
                                                               {"order": "desc"}},
                                                                    {"@timestamp":
                                                               {"order": "asc"}}],
                                                                  "size": _size},)
        df2 = Select.from_dict(results).to_pandas()
        df2['promediado_cog_W1'] = df2['promediado_cog_W1'].replace(',','.',
regex=True).astype(float)
        output_list.append([
            df.loc[i, 'key.mmsi'],
            df2.loc[0,'cambios_de_bandera'],
            df2['celda_h3_6'].value_counts().max(),
            df2['promediado_cog_W1'].max(),
            df2.loc[df2['celda_h3_6']==df2['celda_h3_6'].value_counts().idxmax(),
'promediado_cog_W1'].max(),
            df2.loc[0,'eslora']])
output = pd.DataFrame(output_list, columns=["mmsi", "bandera", "repeticiones",
"promediado_cog_W1_1", "promediado_cog_W1_2","eslora"])
output.to_csv("repeticiones_cog.csv", index=False)
print(output.describe())
output.plot.scatter(x='repeticiones', y='promediado_cog_W1_1',
c=output['bandera'], cmap="bwr")
output.plot.scatter(x='repeticiones', y='eslora', c=output['bandera'], cmap="bwr")
output.loc[output['bandera']==True].plot.scatter(x='repeticiones',
y='promediado_cog_W1_1')
plt.xlabel("Repeticiones máximas en una celda")
output.loc[output['bandera']==False].plot.scatter(x='repeticiones',
y='promediado_cog_W1_1')
plt.show()
output.plot.scatter(x='repeticiones', y='promediado_cog_W1_2',
c=output['bandera'], cmap="bwr")
output.loc[output['bandera']==True].plot.scatter(x='repeticiones',
y='promediado_cog_W1_2')
output.loc[output['bandera']==False].plot.scatter(x='repeticiones',
y='promediado_cog_W1_2')
plt.show()
output.loc[output['bandera']==True, 'repeticiones'].hist()
plt.show()
output.loc[output['bandera']==False, 'repeticiones'].hist()
plt.show()
output.loc[output['bandera']==True, 'promediado_cog_W1_1'].hist()
plt.show()
output.loc[output['bandera']==False, 'promediado_cog_W1_1'].hist()
plt.show()
```

```
output.loc[output['bandera']==True, 'promediado_cog_W1_2'].hist()
plt.show()
output.loc[output['bandera']==False, 'promediado_cog_W1_2'].hist()
plt.show()

print("Promediado COG en toda la serie:")
print("Anomalos con cambio de bandera (COG>50, REP>19): ",
      output.loc[(output['bandera']==True) & (output['repeticiones']>19) &
(output['promediado_cog_W1_1']>50), 'mmsi'].count(),
      "/",
      output.loc[output['bandera']==True, 'mmsi'].count()
     )

print("Anomalos sin cambio de bandera: (COG>50, REP>19)",
      output.loc[(output['bandera']==False) & (output['repeticiones']>19) &
(output['promediado_cog_W1_1']>50), 'mmsi'].count(),
      "/",
      output.loc[output['bandera']==False, 'mmsi'].count()
     )

print("Anomalos con cambio de bandera (COG>100, REP>19): ",
      output.loc[(output['bandera']==True) & (output['repeticiones']>19) &
(output['promediado_cog_W1_1']>100), 'mmsi'].count(),
      "/",
      output.loc[output['bandera']==True, 'mmsi'].count()
     )

print("Anomalos sin cambio de bandera: (COG>100, REP>19)",
      output.loc[(output['bandera']==False) & (output['repeticiones']>19) &
(output['promediado_cog_W1_1']>100), 'mmsi'].count(),
      "/",
      output.loc[output['bandera']==False, 'mmsi'].count()
     )

print("Promediado COG en la celda más repetida:")

print("Anomalos con cambio de bandera (COG>50, REP>19): ",
      output.loc[(output['bandera']==True) & (output['repeticiones']>19) &
(output['promediado_cog_W1_2']>50), 'mmsi'].count(),
      "/",
      output.loc[output['bandera']==True, 'mmsi'].count()
     )

print("Anomalos sin cambio de bandera: (COG>50, REP>19)",
      output.loc[(output['bandera']==False) & (output['repeticiones']>19) &
(output['promediado_cog_W1_2']>50), 'mmsi'].count(),
      "/",
      output.loc[output['bandera']==False, 'mmsi'].count()
     )

print("Anomalos con cambio de bandera (COG>100, REP>19): ",
      output.loc[(output['bandera']==True) & (output['repeticiones']>19) &
(output['promediado_cog_W1_2']>100), 'mmsi'].count(),
      "/",
      output.loc[output['bandera']==True, 'mmsi'].count()
     )

print("Anomalos sin cambio de bandera: (COG>100, REP>19)",
```

```
      output.loc[(output['bandera']==False) & (output['repeticiones']>19) &
(output['promediado_cog_W1_2']>100), 'mmsi'].count(),
      "/",
      output.loc[output['bandera']==False, 'mmsi'].count())
```

ANEXO VII: ANÁLISIS DE PARÁMETROS INDICADORES DE ACTIVIDADES SOSPECHOSAS DE LOS BUQUES DETECTADOS

```
import sys
from elasticsearch import Elasticsearch
import ssl
import pandas as pd
from pandasticsearch import Select
import time
import matplotlib.pyplot as plt
# CONFIGURAR
# Nombre del indice que se quiere utilizar
_index_name="ais_data_1805_to_0206_def"
# Datos de conexion de Elasticsearch
_host="192.168.16.20"
_port=9200
_http_auth=('cemai_admin', 'cemai4dmin21')
_timeout=30
_type="MERCANTE"
# Numero de resultados
_size=10000
# Se inicia la conexion con Elasticsearch
context = ssl.create_default_context(cafile="elasticsearch-ca.pem")
context.check_hostname = False
context.verify_mode = ssl.CERT_NONE
elastic = Elasticsearch([{'host': _host, 'port': _port}], http_auth=_http_auth,
scheme="https", ssl_context=context, timeout=_timeout)

detectados=["209188000","210163000","224094450","224121630","224133940","224226000
","224237000","224388570","224405560","224503230","224810000","225410000","2259779
80","225985074","225986632","229610000","244060083","247318900","255806271","31100
5800","CONFIDENCIAL","314464000","370801000","431076000","CONFIDENCIAL","538001857
","CONFIDENCIAL","636019366"]
df=pd.DataFrame()
output_list=[]
for w in detectados:
      results = elastic.search(index=_index_name, body={"size": 0,"query": {"bool":
                                                  { "must": [{"term": { "mmsi": w}},
                                                {"term": {"tipo_buque_AIS": _type}},
                                                   {"range": {"sog": {"gt": 2}}}]}},
                                                                  "aggs": {"cells":
                                                          {"composite": {"sources":
                                        [{ "mmsi": { "terms": { "field": "mmsi"}}}],
                                                                  "size" : 10000}}}})
df=df.append(pd.json_normalize(results['aggregations']['cells']['buckets']),ignore
_index=True)

for i in df.index:
    df.loc[i,"SUSP"]=False
for i in df.index:
    if df.loc[i,'key.mmsi']==CONFIDENCIAL or df.loc[i, 'key.mmsi']==CONFIDENCIAL
or df.loc[i, 'key.mmsi']==CONFIDENCIAL:
        df.loc[i,"SUSP"]=True
for i in df.index:
    if df.loc[i,'doc_count']>20:
```

```
        h=df.loc[i,'key.mmsi']
             results = elastic.search(index=_index_name, body={"query": { "bool": {
                                                    "must": [{"term": { "mmsi":h }},
                                                    {"term": { "dentro_ZEE": True}},
                                                      {"range": {"sog": {"gt": 2}}},
                                               {"term": { "tipo_buque_AIS": _type}}]}},
                                                                "sort": [{"mmsi":
                                                              {"order": "desc"}},
                                                                   {"@timestamp":
                                                              {"order": "asc"}}],
                                                                  "size": _size})
        df2 = Select.from_dict(results).to_pandas()
        df2['promediado_cog_W1'] = df2['promediado_cog_W1'].replace(',','.',
regex=True).astype(float)
        output_list.append([
            df.loc[i, 'key.mmsi'],
            df2.loc[0,'cambios_de_bandera'],
            df2['celda_h3_6'].value_counts().max(),
            len(df2['celda_h3_6'].value_counts()>19),
            df2['promediado_cog_W1'].max(),
            df2.loc[df2['celda_h3_6']==df2['celda_h3_6'].value_counts().idxmax(),
'promediado_cog_W1'].max(),
            df.loc[i,'SUSP']])

output = pd.DataFrame(output_list, columns=["mmsi", "bandera", "repeticiones",
"CEL", "promediado_cog_W1_1", "promediado_cog_W1_2","SUSP"])
output.to_csv("repeticiones_cog.csv", index=False)
print(output.describe())
output.plot.scatter(x='CEL', y='promediado_cog_W1_1', c=output['SUSP'],
cmap="bwr")
output.plot.scatter(x='repeticiones', y='promediado_cog_W1_1', c=output['SUSP'],
cmap="bwr")
output.loc[output['SUSP']==True].plot.scatter(x='repeticiones',
y='promediado_cog_W1_1')
output.loc[output['SUSP']==False].plot.scatter(x='repeticiones',
y='promediado_cog_W1_1')
plt.show()
output.plot.scatter(x='CEL', y='promediado_cog_W1_2', c=output['SUSP'],
cmap="bwr")
output.plot.scatter(x='repeticiones', y='promediado_cog_W1_2', c=output['SUSP'],
cmap="bwr")
output.loc[output['SUSP']==True].plot.scatter(x='repeticiones',
y='promediado_cog_W1_2')
output.loc[output['SUSP']==False].plot.scatter(x='repeticiones',
y='promediado_cog_W1_2')
plt.show()
output.loc[output['SUSP']==True, 'repeticiones'].hist()
plt.show()
output.loc[output['SUSP']==False, 'repeticiones'].hist()
plt.show()
output.loc[output['SUSP']==True, 'promediado_cog_W1_1'].hist()
plt.show()
output.loc[output['SUSP']==False, 'promediado_cog_W1_1'].hist()
plt.show()

output.loc[output['SUSP']==True, 'promediado_cog_W1_2'].hist()
plt.show()
```

```
output.loc[output['SUSP']==False, 'promediado_cog_W1_2'].hist()
plt.show()

print("Promediado COG en toda la serie:")
print("Anomalos SOSPECHOSOS: (COG>50, REP>19): ",
      output.loc[(output['SUSP']==True) & (output['repeticiones']>19) &
(output['promediado_cog_W1_1']>50), 'mmsi'].count(),
      "/",
      output.loc[output['SUSP']==True, 'mmsi'].count()
     )

print("Anomalos NO SOSPECHOSOS: (COG>50, REP>19)",
      output.loc[(output['SUSP']==False) & (output['repeticiones']>19) &
(output['promediado_cog_W1_1']>50), 'mmsi'].count(),
      "/",
      output.loc[output["SUSP"]==False, 'mmsi'].count()
     )

print("Anomalos SOSPECHOSOS (COG>100, REP>19): ",
      output.loc[(output['SUSP']==True) & (output['repeticiones']>19) &
(output['promediado_cog_W1_1']>100), 'mmsi'].count(),
      "/",
      output.loc[output['SUSP']==True, 'mmsi'].count()
     )

print("Anomalos NO SOSPECHOSOS: (COG>100, REP>19)",
      output.loc[(output['SUSP']==False) & (output['repeticiones']>19) &
(output['promediado_cog_W1_1']>100), 'mmsi'].count(),
      "/",
      output.loc[output['SUSP']==False, 'mmsi'].count()
     )

print("Promediado COG en la celda más repetida:")

print("Anomalos SOSPECHOSO (COG>50, REP>19): ",
      output.loc[(output['SUSP']==True) & (output['repeticiones']>19) &
(output['promediado_cog_W1_2']>50), 'mmsi'].count(),
      "/",
      output.loc[output['SUSP']==True, 'mmsi'].count()
     )

print("Anomalos NO SOSPECHOSO: (COG>50, REP>19)",
      output.loc[(output['SUSP']==False) & (output['repeticiones']>19) &
(output['promediado_cog_W1_2']>50), 'mmsi'].count(),
      "/",
      output.loc[output["SUSP"]==False, 'mmsi'].count()
     )

print("Anomalos SOSPECHOSOS: (COG>100, REP>19): ",
      output.loc[(output['SUSP']==True) & (output['repeticiones']>19) &
(output['promediado_cog_W1_2']>100), 'mmsi'].count(),
      "/",
      output.loc[output['SUSP']==True, 'mmsi'].count()
     )

print("Anomalos NO SOSPECHOSOS: (COG>100, REP>19)",
      output.loc[(output['SUSP']==False) & (output['repeticiones']>19) &
(output['promediado_cog_W1_2']>100), 'mmsi'].count(),
```

```
"/",
output.loc[output['SUSP']==False, 'mmsi'].count())
```