



universidad  
de león



# Escuela de Ingenierías Industrial, Informática y Aeroespacial

## MÁSTER EN INGENIERÍA AERONÁUTICA

Trabajo de Fin de Máster

APLICACIÓN DE *MACHINE LEARNING* A LA GESTIÓN  
AEROPORTUARIA

MACHINE LEARNING APPLICATION TO AIRPORT  
MANAGEMENT

Autora: Sandra Victorero Huerta

Tutor: Miguel Ángel Conde González

(Febrero, 2022)

**UNIVERSIDAD DE LEÓN**  
**Escuela de Ingenierías Industrial, Informática y**  
**Aeroespacial**

**MÁSTER EN INGENIERÍA AERONÁUTICA**  
**Trabajo de Fin de Máster**

**ALUMNA:** Sandra Victorero Huerta

**TUTOR:** Miguel Ángel Conde González

**TÍTULO:** Aplicación de *machine learning* a la gestión aeroportuaria

**TITLE:** Machine learning application to airport management

**CONVOCATORIA:** Febrero, 2022

**RESUMEN:**

El fuerte crecimiento pronosticado para los niveles de tráfico aéreo, unido a los retos introducidos por la pandemia del COVID-19, la preocupación medioambiental, los cambios en las regulaciones y en las necesidades y comportamientos de los pasajeros, y el auge de la digitalización y la importancia del dato, representan un desafío a los aeropuertos y su gestión, que deben ser capaces de adaptarse para mantenerse eficientes y sostenibles a largo plazo, proporcionando la capacidad necesaria para albergar las operaciones. En este trabajo se realiza un estudio de las soluciones y beneficios que aporta el uso del aprendizaje automático para este fin, ayudando en la gestión eficiente de los medios aeroportuarios para reducir los costes de operación, y reduciendo los tiempos de procesamiento y de espera en cola en los distintos procesos para mejorar la experiencia de los pasajeros e incrementar ingresos. Como caso práctico de estudio se plantea la predicción del número de controles de seguridad requeridos mediante la construcción de un modelo que aplica las fórmulas de planificación y dimensionamiento de equipos establecidas en el ADRM de IATA e implementa técnicas de aprendizaje supervisado mediante clasificación con los algoritmos de Random Forest, K-NN y SVM. Tras el entrenamiento del modelo, se evalúan las métricas más relevantes en este tipo de problemas: exactitud, precisión, *recall*, y *F1-score*. Los tres métodos proporcionan unos resultados muy aceptables, con una exactitud superior al 90% en todos los casos, siendo el Random Forest y el K-NN los

mejores modelos, con unos resultados casi idénticos, una exactitud superior al 92% y el resto de las métricas por encima del 91%; mientras que el SVM está ligeramente por debajo en exactitud, con un 90,36%, y unos resultados algo más bajos en términos de precisión, pero manteniéndose por encima del 81%, lo que hace que el F1-score también baje, aunque está por encima del 85%.

**ABSTRACT:**

The significant predicted growth for air traffic levels, coupled with the challenges introduced by the COVID-19 pandemic, environmental concerns, changes in regulations and passenger needs and behaviours, and the rise of digitalisation and the importance of data, represent a challenge to airports and their management, who must be able to adapt to remain efficient and sustainable in the long term, providing the necessary capacity to host operations. This work studies the solutions and benefits of using machine learning for this purpose, helping in the efficient asset management of airports to reduce operating costs, and reducing processing and queuing times in the different processes to improve the passenger experience and increase revenues. As a practical case study, the prediction of the number of required security controls is proposed through the construction of a model that applies the planning and equipment sizing formulas established in the IATA ADRM and implements supervised learning techniques through classification with Random Forest, K-NN and SVM algorithms. After training the model, the most relevant metrics in this type of problem are evaluated: accuracy, precision, recall, and F1-score. The three methods provide very acceptable results, with an accuracy above 90% in all cases. Random Forest and K-NN are the best models, with almost identical results, an accuracy above 92% and the rest of the metrics above 91%; SVM is slightly below in accuracy, with 90.36%, and it offers lower results in terms of precision, but still above 81%, which means that the F1-score also drops, although it is above 85%.

**Palabras clave:** *machine learning*, aprendizaje automático, *asset management*, gestión de recursos, *airport* , aeropuerto

**Firma del alumno:****VºBº Tutor/es:**

---

# Índice de contenidos

Índice de contenidos .....	4
Índice de figuras .....	6
Índice de tablas.....	8
Glosario.....	9
1. Introducción .....	10
1.1 Objetivos del trabajo .....	10
1.2 Breve reseña histórica .....	12
1.3 Motivación .....	15
1.4 Estructura del trabajo .....	17
2. Conceptos de <i>machine learning</i> .....	20
2.1 Aprendizaje supervisado.....	21
2.2 Aprendizaje no supervisado.....	27
2.3 Aprendizaje por refuerzo .....	31
2.4 Aprendizaje profundo y redes neuronales .....	32
3. Estado del arte .....	36
3.1 Revisión sistemática de la literatura.....	36
3.2 Aplicaciones instaladas actualmente en los aeropuertos del mundo .....	85
3.3 Normativa .....	90
4. Concepto <i>smart airports</i> .....	92
4.1 Seguridad .....	93
4.2 Experiencia del pasajero .....	94
4.3 Sostenibilidad y resiliencia.....	94
5. Aplicación práctica .....	96
5.1 Descripción del caso de estudio .....	96
5.2 Análisis y procesamiento de los datos.....	96

---

5.3	Construcción del modelo .....	107
5.4	Entrenamiento y evaluación del modelo.....	115
6.	Resultados .....	118
7.	Conclusiones y recomendaciones .....	120
	Bibliografía.....	122
	Anexo 1: Código desarrollado .....	132

# Índice de figuras

Figura 1.1: El impacto del COVID-19 en números ( <i>Fuente: [1]</i> ) .....	10
Figura 1.2: Recuperación del tráfico aéreo en las distintas crisis ( <i>Fuente: [2]</i> ) .....	11
Figura 1.3: Evolución tecnológica de los aeropuertos ( <i>Fuente: elaboración propia</i> ).....	14
Figura 1.4: Inversión anual de los aeropuertos en IT en todo el mundo ( <i>Fuente: [4]</i> ) .....	14
Figura 1.5: Prioridades de inversión de los aeropuertos para 2023 ( <i>Fuente: [5]</i> ) .....	15
Figura 1.6: Diagrama Gantt para la elaboración de este TFM ( <i>Fuente: elaboración propia</i> ) .....	19
Figura 2.1: Ramas dentro de la IA ( <i>Fuente: [6]</i> ).....	20
Figura 2.2: Ejemplo de función logística ( <i>Fuente: [7]</i> ).....	22
Figura 2.3: Esquema de árbol de decisión ( <i>Fuente: [9]</i> ).....	22
Figura 2.4: Ejemplo de regresión lineal simple ( <i>Fuente: [10]</i> ) .....	23
Figura 2.5: Bases del SVM para clasificación ( <i>Fuente: [12]</i> ).....	25
Figura 2.6: Esquema de funcionamiento de métodos <i>ensemble</i> ( <i>Fuente: [13]</i> ).....	26
Figura 2.7: Ejemplo algoritmo <i>k-means</i> ( <i>Fuente: [16]</i> ).....	28
Figura 2.8: Ejemplo de agrupamiento jerárquico ( <i>Fuente: [17]</i> ).....	28
Figura 2.9: Ejemplo agrupamiento <i>Fuzzy C-Means</i> ( <i>Fuente: [18]</i> ) .....	29
Figura 2.10: Técnica de análisis de componentes principales ( <i>Fuente: [20]</i> ).....	30
Figura 2.11: Esquema aprendizaje por refuerzo ( <i>Fuente: [21]</i> ) .....	31
Figura 2.12: Estructura común de una red neuronal ( <i>Fuente: [22]</i> ) .....	32
Figura 2.13: Red neuronal de base radial ( <i>Fuente: [24]</i> ).....	34
Figura 2.14: Redes neuronales VS Aprendizaje profundo ( <i>Fuente: [25]</i> ).....	35
Figura 3.1: Definición del protocolo de revisión ( <i>Fuente: elaboración propia</i> ) .....	37
Figura 3.2: Número de resultados por base de datos ( <i>Fuente: elaboración propia</i> ) .....	44
Figura 3.3: Porcentaje de trabajos aceptados y descartados ( <i>Fuente: elaboración propia</i> ) .....	45
Figura 3.4: Diagrama modelo PRISMA de los resultados ( <i>Fuente: elaboración propia</i> ) .....	46
Figura 3.5: Trabajos seleccionados por campo de estudio ( <i>Fuente: elaboración propia</i> ) ..	55
Figura 3.6: Evolución anual del número de trabajos ( <i>Fuente: elaboración propia</i> ).....	84

---

Figura 3.7: Mejora en los procesos IATA OneID (Fuente: [80]).....	91
Figura 4.1: Concepto de <i>smart airport</i> (Fuente: [84]) .....	93
Figura 5.1: Resultados de datos faltantes y duplicados (Fuente: <i>elaboración propia</i> ) .....	99
Figura 5.2: Pasajeros totales por mes (Fuente: <i>elaboración propia</i> ).....	99
Figura 5.3: Pasajeros/hora máximos por mes (Fuente: <i>elaboración propia</i> ) .....	100
Figura 5.4: Media de pasajeros/hora por mes (Fuente: <i>elaboración propia</i> ) .....	100
Figura 5.5: Pasajeros totales por día de la semana (Fuente: <i>elaboración propia</i> ) .....	101
Figura 5.6: Pasajeros/hora máximos por día de la semana (Fuente: <i>elaboración propia</i> )	102
Figura 5.7: Media de pasajeros/hora por día de la semana (Fuente: <i>elaboración propia</i> ) .....	102
Figura 5.8: Pasajeros/hora máximos (Fuente: <i>elaboración propia</i> ) .....	103
Figura 5.9: Media de pasajeros/hora (Fuente: <i>elaboración propia</i> ) .....	103
Figura 5.10: Datos originales leídos en Python (Fuente: <i>elaboración propia</i> ) .....	104
Figura 5.11: Muestra datos originales (Fuente: <i>elaboración propia</i> ) .....	104
Figura 5.12: Datos leídos tras el procesamiento (Fuente: <i>elaboración propia</i> ) .....	106
Figura 5.13: Niveles de servicio definidos por IATA (Fuente: [85]) .....	108
Figura 5.14: Datos leídos tras añadir columna de “Controles” (Fuente: <i>elaboración propia</i> ) .....	109
Figura 5.15: Hiperparámetros por defecto Random Forest (Fuente: [86]).....	111
Figura 5.16: Importancia de las características (Fuente: <i>elaboración propia</i> ) .....	112
Figura 5.17: Hiperparámetros por defecto K-NN (Fuente: [88]) .....	113
Figura 5.18: Accuracy VS valor de K (Fuente: <i>elaboración propia</i> ).....	114
Figura 5.19: Hiperparámetros por defecto SVM (Fuente: [89]) .....	115
Figura 5.20: Resultados evaluación de los modelos (Fuente: <i>elaboración propia</i> ).....	116

---

# Índice de tablas

Tabla 3.1. Definición del objetivo de la SLR ( <i>Fuente: elaboración propia</i> ).....	38
Tabla 3.2. Fuentes de búsqueda para la SLR ( <i>Fuente: elaboración propia</i> ) .....	39
Tabla 3.3. Palabras clave y sinónimos ( <i>Fuente: elaboración propia</i> ) .....	39
Tabla 3.4. Definición de la cadena de búsqueda genérica ( <i>Fuente: elaboración propia</i> ) ...	40
Tabla 3.5. Definición de los criterios de inclusión ( <i>Fuente: elaboración propia</i> ) .....	40
Tabla 3.6: Definición de los criterios de exclusión ( <i>Fuente: elaboración propia</i> ) .....	41
Tabla 3.7. Definición de los criterios de calidad ( <i>Fuente: elaboración propia</i> ) .....	42
Tabla 3.8. Cadena de búsqueda en IEEE Digital Library ( <i>Fuente: elaboración propia</i> ) .....	43
Tabla 3.9. Cadena de búsqueda en Science Direct ( <i>Fuente: elaboración propia</i> ).....	43
Tabla 3.10. Cadena de búsqueda en Springer Link ( <i>Fuente: elaboración propia</i> ) .....	43
Tabla 3.11. Cadena de búsqueda en Scopus ( <i>Fuente: elaboración propia</i> ) .....	43
Tabla 3.12. Número de resultados en cada base de datos ( <i>Fuente: elaboración propia</i> ) ..	44
Tabla 3.13: Resultados obtenidos en la SLR ( <i>Fuente: elaboración propia</i> ) .....	48
Tabla 3.14: Resumen trabajos de “predicción de demoras” ( <i>Fuente: elaboración propia</i> )	61
Tabla 3.15: Resumen trabajos de “meteorología” ( <i>Fuente: elaboración propia</i> ).....	66
Tabla 3.16: Resumen trabajos de “rodaje de las aeronaves” ( <i>Fuente: elaboración propia</i> ) .....	69
Tabla 3.17: Resumen trabajos de “edificio terminal” ( <i>Fuente: elaboración propia</i> ) .....	72
Tabla 3.18: Resumen trabajos de “pasajeros” ( <i>Fuente: elaboración propia</i> ) .....	74
Tabla 3.19: Resumen trabajos de “operaciones” ( <i>Fuente: elaboración propia</i> ).....	76
Tabla 3.20: Resumen trabajos de “tratamiento de equipajes” ( <i>Fuente: elaboración propia</i> ) .....	77
Tabla 3.21: Resumen trabajos de “ruido” ( <i>Fuente: elaboración propia</i> ) .....	79
Tabla 3.22: Resumen trabajos de “general” ( <i>Fuente: elaboración propia</i> ).....	80
Tabla 3.23: Resumen trabajos de “mantenimiento de campo de vuelos” ( <i>Fuente: elaboración propia</i> ) .....	80
Tabla 3.24: Ejemplos de biométrica en aeropuertos ( <i>Fuente: elaboración propia</i> ) .....	86
Tabla 5.1: Resultados evaluación de los modelos ( <i>Fuente: elaboración propia</i> ) .....	117



# Glosario

IATA: International Air Transport Association

ACI: Airports Council International

EASA: European Union Aviation Safety Agency

CAGR: Compound Annual Growth Rate

ML: Machine Learning

IA: Inteligencia Artificial

RF: Random Forest

K-NN: K-nearest neighbour

SVM: Support Vector Machine

NBC: Naïve Bayes Classifier

PCA: Principal Component Analysis

SVD: Singular Value Decomposition

DL: Deep Learning

ANN: Artificial Neural Network

CNN: Convolutional Neural Network

LSTM: Long Short-Term Memory

SVR: Support Vector Regressor

SVC: Support Vector Classifier

IoT: Internet of Things

RMSE: Root Mean Square Error

MAE: Mean Absolute Error

ATC: Air Traffic Control

A-CDM: Airport Collaborative Decision Making

RVR: Runway Visual Range

LVP: Low Visibility Procedure

OACI: Organización de Aviación Civil Internacional

ADRM: Airport Development Reference Manual

# 1. Introducción

## 1.1 OBJETIVOS DEL TRABAJO

Hoy en día, los aeropuertos cumplen múltiples funciones diferentes y acogen a una gran variedad de partes interesadas, que a menudo tienen objetivos divergentes. A esto se une el gran dinamismo de la industria del transporte aéreo, con constantes cambios regulatorios, de tendencias y ciclos, y de preferencias y necesidades de los usuarios.

Hasta el cierre del año 2019, el tráfico aéreo se encontraba en una década de crecimiento en la que los aeropuertos se centraban en aumentar su capacidad y eficiencia para dar respuesta a una demanda en continuo aumento. En ese momento, surgió la pandemia del COVID-19 que supuso una crisis mundial sin precedentes para el sector, trayendo consigo la mayor caída de los niveles de tráfico en la historia de la aviación. Este impacto se puede apreciar en la Figura 1.1 que sigue.

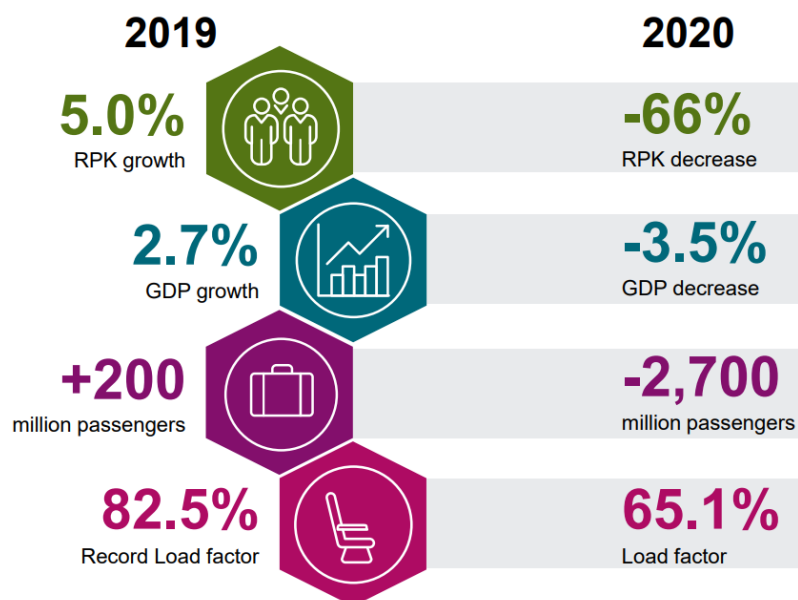


Figura 1.1: El impacto del COVID-19 en números (Fuente: [1])

A lo largo de su evolución, el transporte aéreo ha sufrido otras crisis importantes, como los atentados del 11S o la crisis económica de 2008, y en todas ha demostrado su capacidad de recuperación y resiliencia. Aun así, el impacto del COVID-19 está suponiendo un

verdadero desafío para todos los involucrados (operadores aeroportuarios, líneas aéreas, fabricantes de aeronaves, etc.) tanto por su golpe inmediato como por su duración en el tiempo e incertidumbre. La Figura 1.2 a continuación muestra esta situación.

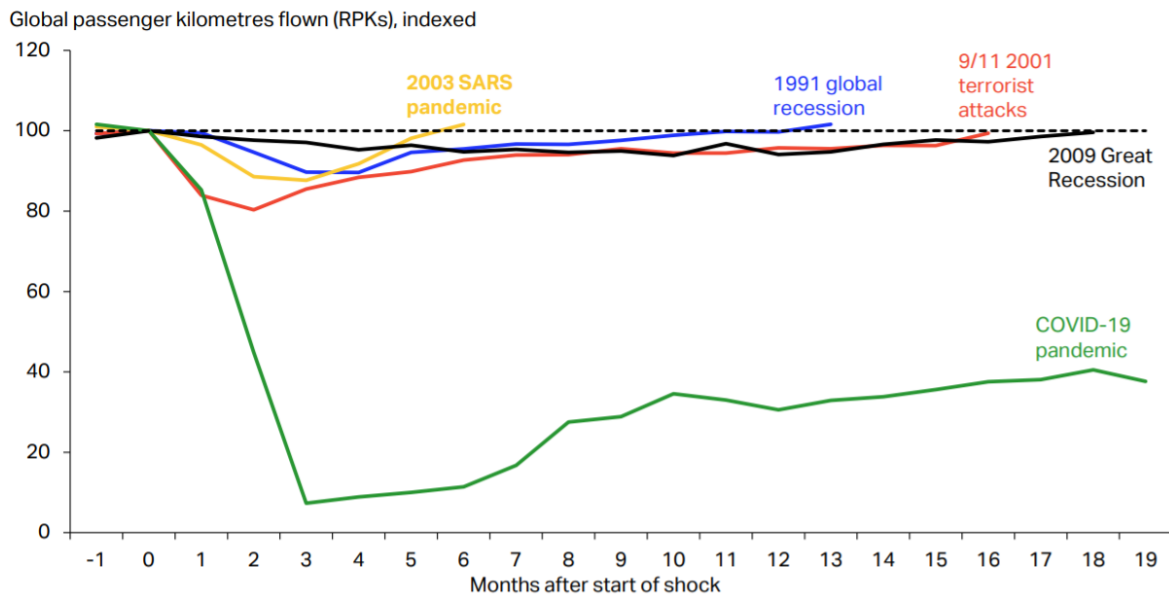


Figura 1.2: Recuperación del tráfico aéreo en las distintas crisis (Fuente: [2])

En esta coyuntura, no es posible establecer con certeza cuál será la evolución en los próximos años, y estará ligada estrechamente a la evolución de la pandemia y las restricciones de viaje impuestas por los distintos gobiernos. No obstante, tanto los principales organismos internacionales (IATA, ACI, EASA, Eurocontrol) como los mayores fabricantes de aeronaves (Airbus y Boeing) pronostican en sus estudios que entre 2023 y 2025 se recuperarán los niveles de tráfico de antes de la pandemia, para continuar con un crecimiento CAGR aproximado del 3,9% hasta 2040.

Otro de los retos a los que se enfrentan actualmente los aeropuertos es la necesidad de conciliar su desarrollo con la protección al medioambiente, tanto desde el punto de vista puramente de las operaciones aéreas como desde los métodos de gestión, que tienen que buscar el equilibrio entre la economía de recursos y la prestación de servicios eficientes y sostenibles.

A todo esto hay que añadir el cambio que se está produciendo en las necesidades, la mentalidad y el comportamiento de los pasajeros, que buscan soluciones más flexibles y ágiles, con mayor compromiso con el medio ambiente, y que ofrezcan mayor digitalización y conectividad.

En este contexto, para mantenerse competitivos y sostenibles, los aeropuertos tienen que ser capaces de adaptarse al ritmo de los cambios. Así, el principal objetivo de este trabajo es estudiar y comprobar las soluciones que aporta el aprendizaje automático o *machine learning* a este fin, convirtiendo la incertidumbre en oportunidad y contribuyendo a dar respuesta a los nuevos retos a los que se enfrenta el sector.

## 1.2 BREVE RESEÑA HISTÓRICA

A lo largo de su historia, los aeropuertos han experimentado un cambio significativo. Los primeros aeródromos servían al único propósito de permitir la operación de las aeronaves, estando formados por áreas despejadas para su operación y las instalaciones auxiliares mínimas requeridas. Estas construcciones fueron evolucionando hacia sistemas más complejos con dos zonas diferenciadas: el lado aire, centrado en albergar las operaciones de las aeronaves con niveles de seguridad cada vez más estrictos; y el lado tierra, cuyo centro es el edificio terminal visto como un centro de procesamiento e intercambio de pasajeros. A partir de ahí, el siguiente paso son los aeropuertos tal y como se conocen hoy en día, grandes sistemas que además de cumplir su función principal de transporte, son centro de numerosas actividades comerciales, económicas y de servicios.

En la evolución de los aeropuertos desde el punto de vista tecnológico, se distinguen cuatro periodos:

- Aeropuerto 1.0:

En este primer periodo, el aeropuerto está enfocado en garantizar la seguridad de las operaciones de las aeronaves. Se proporcionan servicios esenciales como la entrega de carga, el embarque y desembarque de los pasajeros, la seguridad y accesos, la entrega y retirada de equipajes, etc., pero sin centrarse en la experiencia de los pasajeros, ofreciendo únicamente unas comodidades mínimas.

La mayoría de los procesos son realizados manualmente por el personal y no existe mayor colaboración ni transferencia de datos entre servicios y partes interesadas.

- **Aeropuerto 2.0:**

También conocido como aeropuerto ágil. Implanta las soluciones tecnológicas digitales en todas las unidades de negocio y entornos operativos. Utiliza un sistema de gestión unificado y compartido que destaca por la colaboración a través de una red única y con sistemas habilitados, como la telefonía IP y la videovigilancia. Todo ello permite una mayor colaboración y transferencia de datos y una mayor eficiencia.

Se incorporan sistemas digitales de información y orientación en la terminal, proporcionando una mejor experiencia al pasajero.

- **Aeropuerto 3.0:**

Este es el periodo de digitalización [3]. Las tecnologías digitales se utilizan ampliamente para la mayoría de los procesos aeroportuarios y para añadir valor más allá de los requisitos operativos básicos. Esto incluye, por ejemplo, el comercio electrónico, el autoservicio en facturación y entrega de equipajes, los escáneres de tarjetas de embarque a través del móvil, los escáneres de cuerpo entero y de tomografía computarizada en el control de seguridad, y los servicios basados en la localización. Los datos electrónicos se capturan y utilizan para la toma de decisiones y se comparten ampliamente con las principales partes interesadas. Varios de los sistemas están interconectados, como los de gestión del propio aeropuerto con los servicios de asistencia en tierra y las aerolíneas para facilitar la gestión de los equipajes.

- **Aeropuerto 4.0:**

Se conoce también como aeropuerto inteligente o *smart airport*. Es la etapa de transformación digital del aeropuerto, en la que se crea valor a partir de los datos que se recogen y comparten con los principales interesados, permitiendo su uso en tiempo real y guardando registros históricos para un mayor conocimiento de la operativa.

En este aeropuerto, los sistemas y procesos están integrados, por tanto, en un ecosistema digital más amplio que conecta a las principales partes interesadas. Por ejemplo, una aplicación sería que los tiempos de espera en fila en el control de seguridad se muestren en los servicios de trenes del aeropuerto, o que el número de pasajeros que entran en la

terminal se recuente y comparta con el control de seguridad o el de fronteras para apoyar la asignación de recursos.

La Figura 1.3 muestra un resumen de estos cuatro periodos de evolución tecnológica en los aeropuertos.

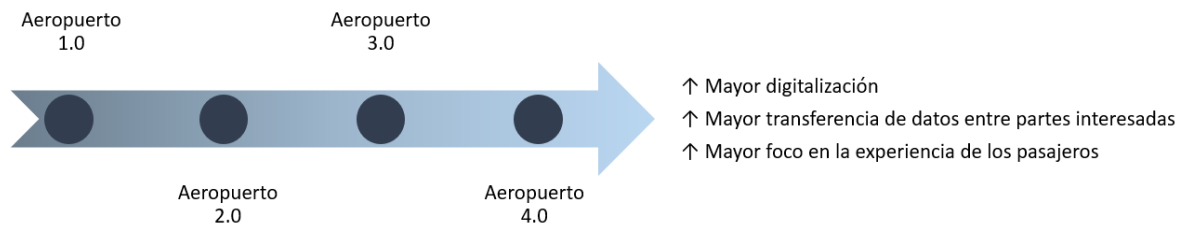
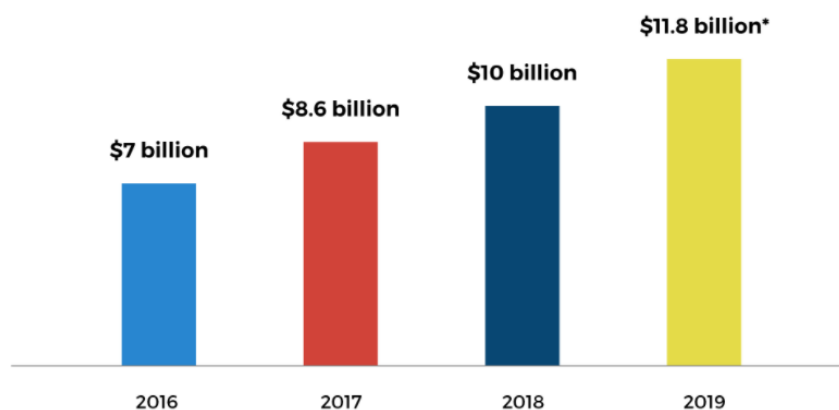


Figura 1.3: Evolución tecnológica de los aeropuertos (*Fuente: elaboración propia*)

Por otra parte, la Figura 1.4 presenta la evolución de la inversión anual de los aeropuertos en IT en todo el mundo. En ella se puede comprobar la tendencia creciente, siguiendo la evolución descrita en los párrafos anteriores.

### Annual Airport Spending on Information Technology Worldwide



Source: SITA Air Transport IT Insights 2019 report.

\*Planned. Based on a survey of 101 senior airport information technology executives done by a market research firm, with answers weighted to reflect the worldwide industry.

Figura 1.4: Inversión anual de los aeropuertos en IT en todo el mundo (*Fuente: [4]*)

### 1.3 MOTIVACIÓN

La transformación digital es un hecho en todos los sectores de las economías desarrolladas. En un mundo cada vez más digitalizado y globalizado, con las personas y las “cosas” cada vez más conectadas, generando datos en cada momento, la importancia de estos datos y la implantación de procesos *data-driven* son fundamentales para muchas organizaciones que necesitan sacar partido a esa información para generar ventajas estratégicas en un entorno cambiante y competitivo.

En este marco, los aeropuertos no son una excepción y están experimentando una gran transformación de la mano de la ciencia de datos. La Figura 1.5 muestra las principales prioridades de inversión de los aeropuertos para el año 2023.

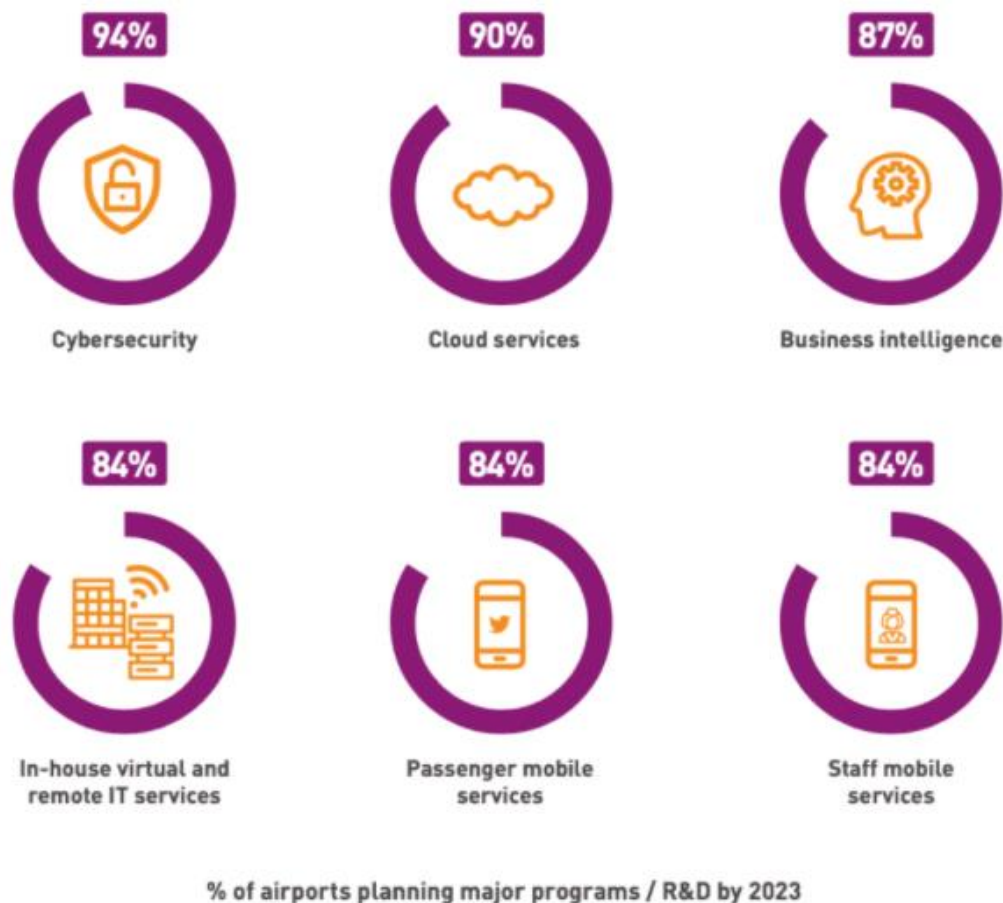


Figura 1.5: Prioridades de inversión de los aeropuertos para 2023 (Fuente: [5])

Las principales motivaciones para esta transformación son las siguientes:

- El volumen de tráfico aéreo aumenta, por lo que los aeropuertos necesitan mejores formas de procesar a los pasajeros, con sistemas de gestión más eficientes.
- La importancia de los datos es cada vez mayor. Los aeropuertos pueden procesar y trabajar con estos datos para optimizar sus procesos e instalaciones y detectar oportunidades de obtener mayores beneficios.
- La experiencia de los pasajeros es un factor clave para que estos elijan volver al aeropuerto. La reducción de los tiempos de espera y la agilización de procesos son fundamentales para mejorar su satisfacción y potenciar la competitividad del aeropuerto.
- Los ingresos no aeronáuticos suponen un porcentaje muy importante en las cuentas de la mayor parte de los aeropuertos comerciales. Y a su vez, una buena oferta comercial supone una mejor experiencia del pasajero.
- Las soluciones tanto en términos de operatividad como del negocio aeroportuario tienen que pasar necesariamente por ser sostenibles a corto, medio y largo plazo, medioambiental y financieramente.
- En esta constante situación de cambio y evolución, las infraestructuras tienen que ser flexibles y escalables, de tal manera que los cambios previstos o imprevistos en la demanda tengan un impacto contenido en la economía del aeropuerto, por ejemplo, mediante soluciones que permitan un uso flexible del espacio.
- Las instalaciones deben ser capaces de adaptarse a los cambios en la legislación, la evolución de los comportamientos de los pasajeros y la evolución de la tecnología.
- Como infraestructuras críticas, los aeropuertos deben garantizar la seguridad en todo momento.
- Los gestores tienen que fomentar la excelencia operativa, con procesos ágiles, y reducir la incertidumbre en sus decisiones.

Con estos retos, las ventajas que puede introducir la aplicación de *machine learning* son múltiples. Entre ellas destacan:



- Permite aprender de la gran cantidad de datos que se tienen disponibles y generar modelos analíticos para su estudio, pudiendo detectar patrones y predecir tendencias y necesidades.
- Extrae y presenta los datos adecuados para la toma de decisiones, reduciendo la incertidumbre en las mismas.
- Proporciona un mayor conocimiento de la operativa en tiempo real, detectando y eliminando cuellos de botella.
- Ofrece un mayor conocimiento de los hábitos, preferencias y comportamientos de los flujos de pasajeros, permitiendo brindar una mejor experiencia mediante el análisis de preferencias y la oferta de productos personalizados.
- Incrementa los niveles de seguridad.
- Reduce el error humano.
- Automatiza tareas que aportan poco valor y optimiza procesos logísticos.
- Favorece la detección y aplicación de acciones preventivas.
- Ayuda en el desarrollo del *e-commerce*.
- Mejora la percepción del pasajero con respecto a la imagen del aeropuerto y, por ende, de la ciudad que representa.
- Permite al gestor realizar análisis financieros sólidos y basados en datos reales y representativos de su aeropuerto.

Por tanto, con todo ello queda justificada la motivación del estudio planteado en este trabajo.

#### 1.4 ESTRUCTURA DEL TRABAJO

Para cumplir con los objetivos establecidos, tras este primer capítulo de Introducción, se incluyen en el capítulo 2 los conceptos básicos de aprendizaje automático. Nótese que no se pretende incluir un análisis muy profundo sobre los conceptos matemáticos de los distintos algoritmos, sino más bien una explicación de su funcionamiento práctico que sienta las bases para el desarrollo del resto del trabajo.

Posteriormente, el capítulo 3 presenta uno de los puntos de mayor relevancia en este trabajo, que es la revisión sistemática de la literatura para dar respuesta a las preguntas de

investigación planteadas. Esto se completa, además, con los resultados de un estudio acerca de las aplicaciones instaladas actualmente en los aeropuertos del mundo, y una referencia a la normativa de aplicación.

En el capítulo 4 se profundiza en el concepto de *smart airport*, por ser el periodo en el que se enmarca este documento dentro de la evolución tecnológica de los aeropuertos.

Otro de los puntos más relevantes de este trabajo se incluye en el capítulo 5 con el desarrollo del caso práctico de estudio. En él se plantea la solución que se quiere ofrecer, el análisis y procesamiento de los datos disponibles de partida, la construcción del modelo de ML y el entrenamiento y evaluación del modelo desarrollado; para posteriormente realizar un análisis más profundo de los resultados obtenidos en el capítulo 6.

Por su parte, el capítulo 7 incluye las principales conclusiones obtenidas, así como las recomendaciones y posibles mejoras futuras que se podrían implementar a partir de este TFM.

Tras esto, y para finalizar, se incluye el listado de referencias bibliográficas empleadas y un anexo con el código Python desarrollado.



## 2. Conceptos de *machine learning*

El aprendizaje automático o *machine learning* (ML) se define como una disciplina científica dentro de la inteligencia artificial (IA) mediante la cual un sistema es capaz de aprender a partir de datos, en lugar de mediante programación explícita. Como se describe en [6], el ML hace uso de una serie de algoritmos que aprenden iterativamente de los datos para obtener información y hacer predicciones en función de la misma.

Hoy en día, se ha generalizado el uso de distintos términos cuando se trata el tema de *big data* y análisis de datos. La Figura 2.1 se incluye para contextualizar dónde se enmarca el aprendizaje automático.

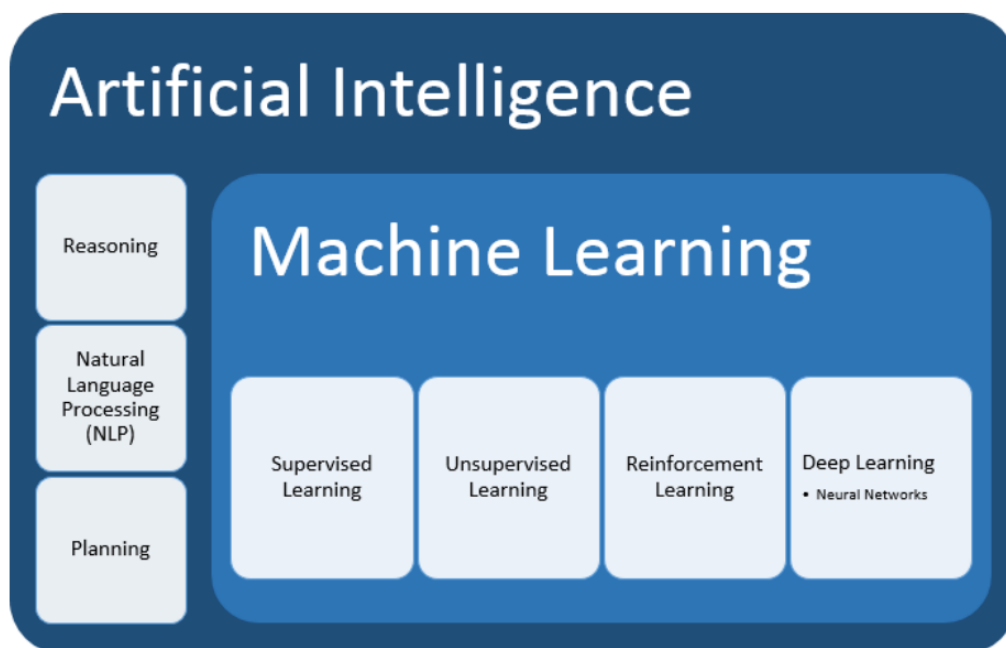


Figura 2.1: Ramas dentro de la IA (Fuente: [6])

Es decir, inteligencia artificial es la categoría general que incluye el ML, el razonamiento, el procesamiento del lenguaje natural y la planificación.

Por su parte, existen múltiples técnicas de aprendizaje automático, y su clasificación y definiciones varían entre las distintas fuentes consultadas. En general, en este trabajo se opta por la siguiente clasificación:

- Técnicas supervisadas (*supervised learning*)
- Técnicas no supervisadas (*unsupervised learning*)
- Técnicas de refuerzo (*reinforcement learning*)
- Aprendizaje profundo (*deep learning*)

La aplicación de una u otra técnica depende del problema que se quiere abordar y los recursos disponibles.

No obstante, nótese que la clasificación que se incluye en los siguientes apartados no es única y se ha elegido así para tratar de organizar la información encontrada en diversas fuentes de una forma esquemática y concisa, con el fin de construir una base sobre la que desarrollar el resto del trabajo.

## 2.1 APRENDIZAJE SUPERVISADO

En el aprendizaje supervisado se parte de un conjunto establecido de datos etiquetados en los que se conoce la relación entre la entrada y la salida (variable objetivo), por lo que el aprendizaje se basa en construir un modelo que explique esa relación, de tal manera que cuando se le apliquen las variables de entrada, este sea capaz de asignarles la etiqueta de salida adecuada. Es decir, en este caso el algoritmo se entrena con datos históricos y así aprende esa relación entrada-salida, por lo que al introducirle un nuevo valor de entrada, es capaz de predecir el valor de salida correspondiente.

Dependiendo del tipo de variable objetivo, se tienen dos subcategorías de aprendizaje supervisado:

- **Clasificación:** la variable objetivo es una categoría. Existen numerosos métodos de clasificación, aunque los más simples y comunes son la regresión logística y los árboles de decisión.
  - **Regresión logística (*Logistic regression*):** se trata de un algoritmo que utiliza la función logística para predecir la probabilidad de ocurrencia de un acontecimiento, por lo que su resultado es un número entre 0 y 1 que indica esa probabilidad. Se utiliza en problemas de clasificación binarios (de dos clases), en los que la variable objetivo es binaria (ocurre o no ocurre).

La Figura 2.2 muestra un ejemplo de función logística.

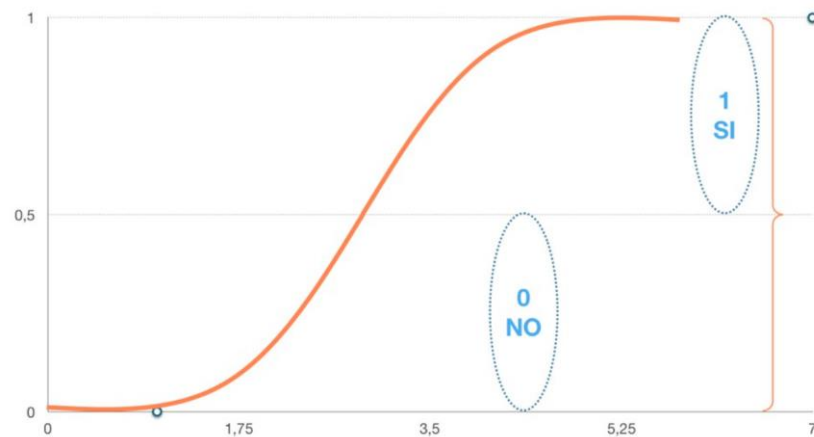


Figura 2.2: Ejemplo de función logística (Fuente: [7])

- **Árboles de decisión (Decision trees):** se trata de una técnica que representa los sucesos que pueden derivar de la toma de una decisión. Partiendo de un nodo inicial, construye los caminos que se pueden recorrer en función de la decisión que se tome en cada nodo. El algoritmo crea el esquema de árbol con los datos de entrenamiento de tal manera que establece un diagrama de construcciones lógicas que posteriormente aplica a los datos de entrada que se introduzcan para tomar las decisiones correctas.

En la Figura 2.3 se presenta un ejemplo de esquema.

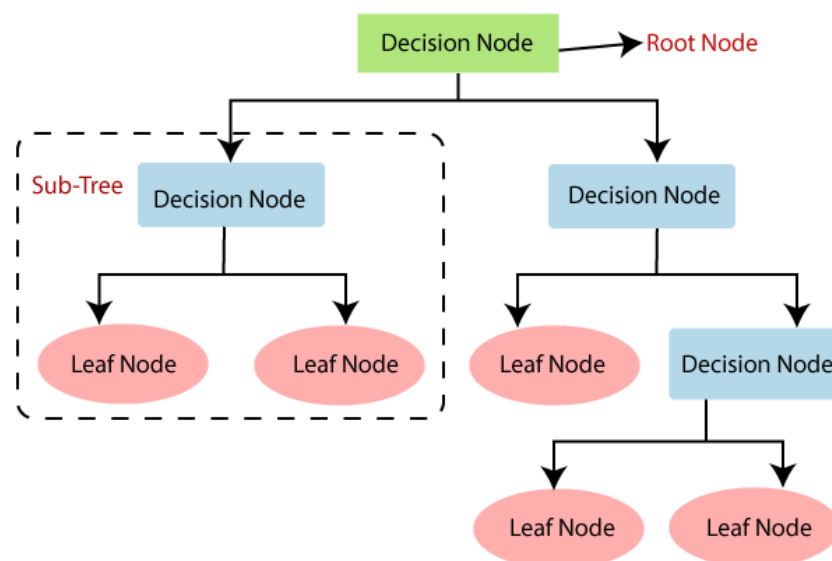


Figura 2.3: Esquema de árbol de decisión (Fuente: [9])

Los árboles de decisión se utilizan sobre todo en clasificación, pero también se emplean en problemas de regresión.

- **Regresión:** la variable objetivo es una variable numérica. La más habitual es la regresión lineal, aunque también se utiliza ampliamente la polinomial.
  - **Regresión lineal (*Linear regression*):** modela la relación entre una variable objetivo (dependiente) y una serie de variables explicativas (independientes). La Figura 2.4 muestra un ejemplo gráfico.

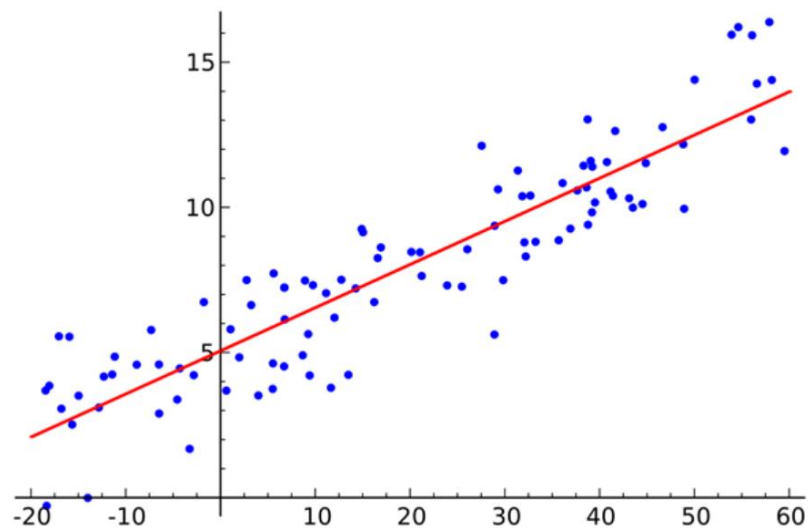


Figura 2.4: Ejemplo de regresión lineal simple (*Fuente: [10]*)

Cabe señalar que la clasificación anterior de las distintas técnicas ha de servir únicamente como una base, ya que en la realidad las relaciones son más complejas, por ejemplo, existen métodos incluidos a priori como clasificación que también pueden ser utilizados para regresión, como los árboles de decisión.

Además de los expuestos anteriormente, entre los algoritmos más usados de aprendizaje supervisado se tienen los siguientes:

- **Bosque aleatorio (*Random Forest, RF*):**

Es un algoritmo que se puede emplear tanto en clasificación como en regresión. Se trata fundamentalmente de una combinación de árboles de decisión, los cuales se obtienen tras un proceso de entrenamiento utilizando una muestra aleatoria diferente para cada uno de

ellos. Se obtiene, por tanto, una predicción distinta en cada caso. La predicción final resultante se obtiene como aquella opción reportada por una mayor cantidad de árboles, lo cual permite obtener un modelo con menor varianza, según se explica en [11].

En general, a mayor número de árboles, más robusto es el bosque y mayor será su precisión.

Entre sus ventajas se tiene la posibilidad de emplear conjuntos de datos con una elevada dimensionalidad, así como de estimar datos faltantes y mantener la precisión cuando falta una alta proporción de datos.

Por el contrario, uno de sus inconvenientes es que no tiene la misma capacidad explicativa que los árboles de decisión y, en algunas ocasiones, se tiene poco control sobre lo que hace el modelo. Además, no produce buenos modelos cuando los conjuntos de datos son pequeños y no son muy buenos en la extrapolación.

- **K- vecino más próximo (*K-nearest neighbour*, K-NN):**

El K-NN es principalmente un algoritmo de clasificación, aunque en ciertos casos también puede usarse en problemas de regresión. Tiene como peculiaridad que no se genera un modelo con el entrenamiento de los datos, sino que el aprendizaje se produce en el mismo entrenamiento, mediante la memorización de las instancias.

Su funcionamiento se basa en calcular la distancia del nuevo dato al resto de datos del conjunto y seleccionar los k elementos más próximos (este k se elige como una variable de control, cada conjunto de datos tendrá su número óptimo). Así, el nuevo dato se clasifica en el grupo en el que tenga un mayor número de k vecinos más próximos.

Para medir estas distancias entre elementos, se pueden aplicar varias opciones: distancia euclidiana, distancia de Manhattan y distancia de Minkowski, entre otras.

Su principal ventaja es que es sencillo de implementar. Sin embargo, requiere un uso muy grande de memoria y recursos de procesamiento, por lo que se utiliza sobre todo cuando los conjuntos de datos son pequeños.

- **Máquinas de vectores de soporte (*Support Vector Machines*, SVM):**

Las máquinas de vectores de soporte también se emplean tanto en clasificación como en regresión. En el caso de la clasificación, SVM genera un hiperplano óptimo de separación de clases a partir de los datos de entrenamiento. Este hiperplano óptimo será el que



maximice el margen de separación entre las clases, siendo los vectores de soporte los puntos de  $n$  dimensiones que definen ese margen máximo. Esto se explica en la Figura 2.5.

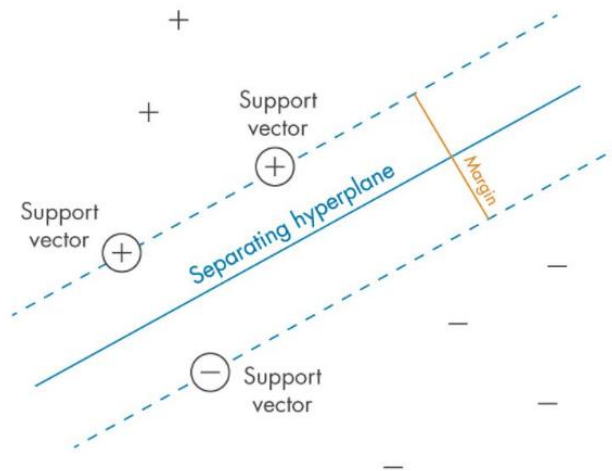


Figura 2.5: Bases del SVM para clasificación (Fuente: [12])

Por su parte, para problemas de regresión, el algoritmo se basa en el mismo principio y busca el hiperplano que mejor modele la tendencia de los datos de entrenamiento y, a partir del mismo, realizar las predicciones para los nuevos datos de entrada.

- **Clasificador Bayesiano ingenuo (Naïve Bayes Classifier, NBC):**

Se trata de un algoritmo de clasificación basado en la probabilidad Bayesiana que asume que todas las variables o características son independientes entre sí. Permite calcular la probabilidad “posterior” de un evento dadas las probabilidades de eventos “anteriores”, basándose en la ecuación 2.1.

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \quad (2.1)$$

Donde:

$P(c|x)$  se conoce como probabilidad a posteriori o probabilidad condicional y se refiere a la probabilidad de que se dé la clase  $c$  si  $x$  (nueva entrada de datos) es verdadero.

$P(x|c)$  es la probabilidad de ocurrencia de  $x$  si  $c$  ocurre, es decir, representa el grado de presunción de que  $x$  pertenezca a  $c$ .

$P(c)$  es la probabilidad total de que ocurra  $c$  o probabilidad total de la clase, es decir, todos los posibles escenarios en los que puede ocurrir  $c$ .

$P(x)$  es la probabilidad de ocurrencia de  $x$  basada en eventos pasados también conocida como evidencia.

Como ventajas, destacan su simplicidad y rapidez; mientras que su mayor inconveniente es la simplificación de que las características son independientes, ya que pocas veces se dará esta condición en la realidad.

Una de sus aplicaciones más conocida es como filtro de correos no deseados.

- **Métodos *Ensemble* (Conjuntos de clasificadores):**

Como su nombre indica, consisten en combinar múltiples algoritmos de aprendizaje con el fin de mejorar los resultados que se obtendrían con la aplicación de esos algoritmos de manera individual. Se aplican tanto en clasificación como en regresión.

Su esquema de funcionamiento se resume en la Figura 2.6.

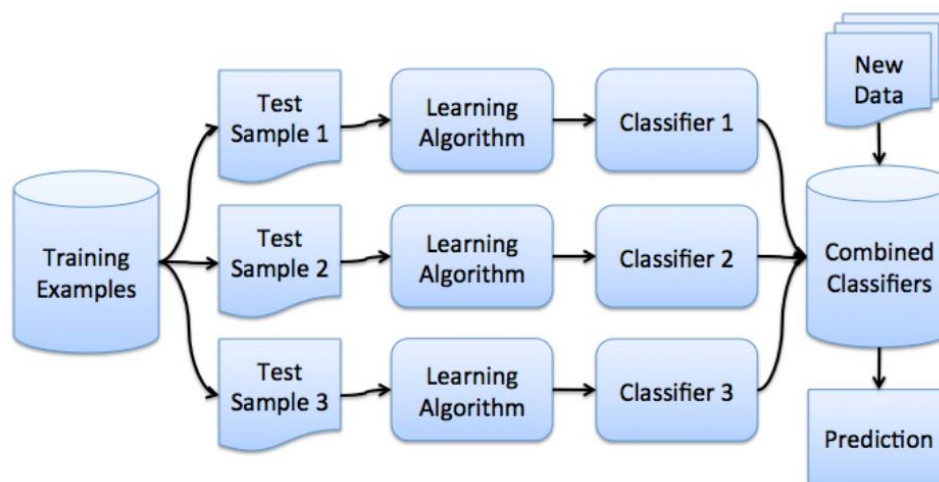


Figura 2.6: Esquema de funcionamiento de métodos *ensemble* (Fuente: [13])

Los métodos de combinación más comunes son: *bagging* y *boosting*.

Por una parte, la idea básica del *bagging* (o *bootstrap aggregation*) es, según [14], remuestrear los datos y calcular las predicciones sobre el nuevo conjunto de datos remuestreados, es decir, se extraen muestras diferentes del conjunto de entrenamiento (muestras *bootstrap*) y se utilizan como conjunto de entrenamiento.

El *bagging* es más eficiente sobre algoritmos no lineales como los árboles de decisión o las redes neuronales. El Random Forest es un *ensemble* de árboles de decisión basado en *bagging*.

Por el contrario, según [15], el *boosting* consiste en variar los pesos de los datos en el conjunto de entrenamiento para generar distintos modelos, pero haciendo uso siempre del conjunto de entrenamiento completo. Se basa en la teoría de *weak learners* (conjunto de aprendizaje que es al menos mejor que la predicción aleatoria, es decir, mayor que 50%). Los métodos de *boosting* más conocidos son: AdaBoost (*Adaptive Boosting*), GBDT (*Gradient Boosting Decision Tree*) o árboles de decisión potenciados por gradiente, xGBoosting (*eXtreme Gradient Boosting*), y LightGBM (*Light Gradient Boosting Machine*).

## 2.2 APRENDIZAJE NO SUPERVISADO

En contraste con las técnicas anteriores, en el aprendizaje no supervisado los datos del conjunto de entrenamiento no están etiquetados y no se tiene una variable objeto predeterminada, es decir, el modelo analiza los datos, estudia su estructura y encuentra patrones para proporcionar resultados útiles. Por tanto, en el aprendizaje no supervisado no existe el entrenamiento del modelo, sino que el algoritmo genera sus propias decisiones en base a los datos de entrada.

Las técnicas de aprendizaje no supervisado más empleadas son la agrupación o clusterización y la reducción de la dimensionalidad:

- **Clusterización:** mediante este método se procesan los datos de entrada y se identifican grupos (o clústeres) a partir de los mismos, es decir, el algoritmo encuentra un patrón y agrupa los datos en base a este, de tal manera que los datos dentro un grupo son similares o al menos más similares entre sí que con los datos de otro grupo. Esta técnica permite ajustar la granularidad del grupo y definir el número de grupos que se deben identificar.

Existen distintos tipos:

- **Agrupamiento exclusivo:** cada dato únicamente puede pertenecer a un grupo. El ejemplo más representativo es el algoritmo *k-means*, que divide

los datos en  $k$  grupos de modo que cada dato pertenece al grupo cuya media es más cercana. La Figura 2.7 muestra un ejemplo de este algoritmo.

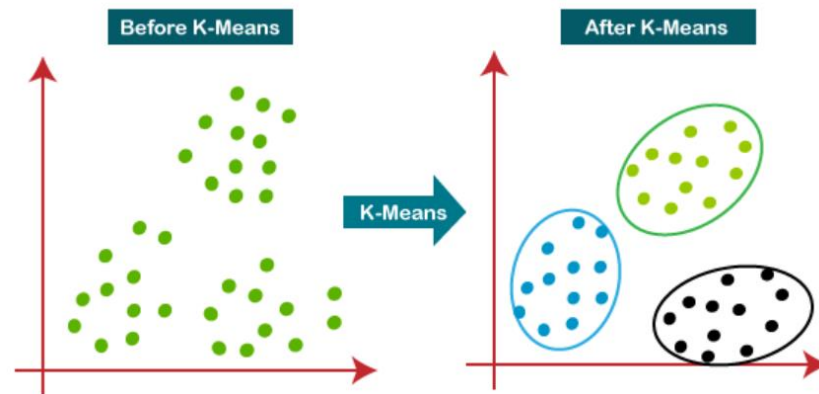


Figura 2.7: Ejemplo algoritmo  $k$ -means (Fuente: [16])

- **Agrupamiento aglomerativo:** cada dato es un grupo y, mediante iteraciones, las uniones de los dos grupos más cercanos van reduciendo el número de grupos. El ejemplo más común es el agrupamiento jerárquico, el cual va construyendo una jerarquía de grupos en cada unión, tal y como se presenta en la Figura 2.8.

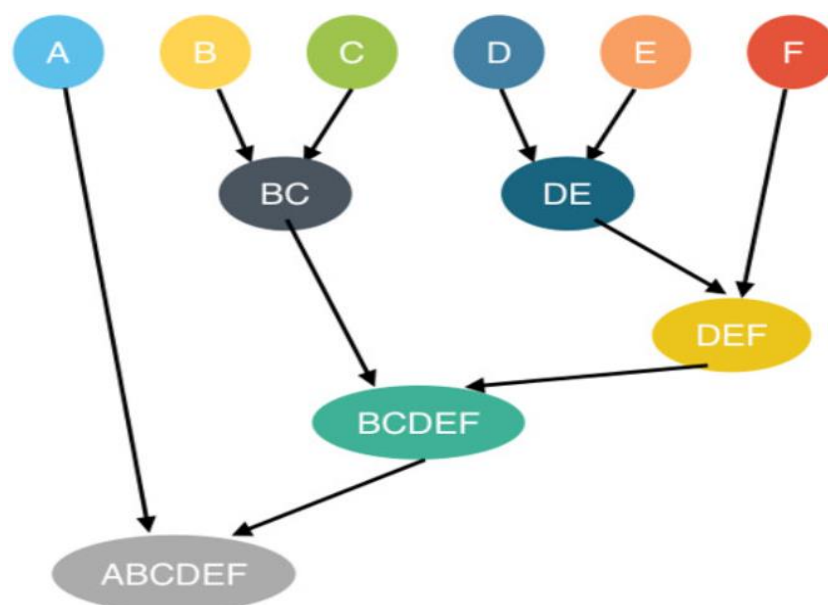


Figura 2.8: Ejemplo de agrupamiento jerárquico (Fuente: [17])

- **Solapamiento:** cada dato puede pertenecer a distintos grupos con diferentes grados de pertenencia (similitud). Un ejemplo es el algoritmo *Fuzzy C-Means*, que utiliza la lógica difusa para asignar un grado de pertenencia a cada dato dentro de cada grupo. La Figura 2.9 muestra un gráfico de ejemplo.

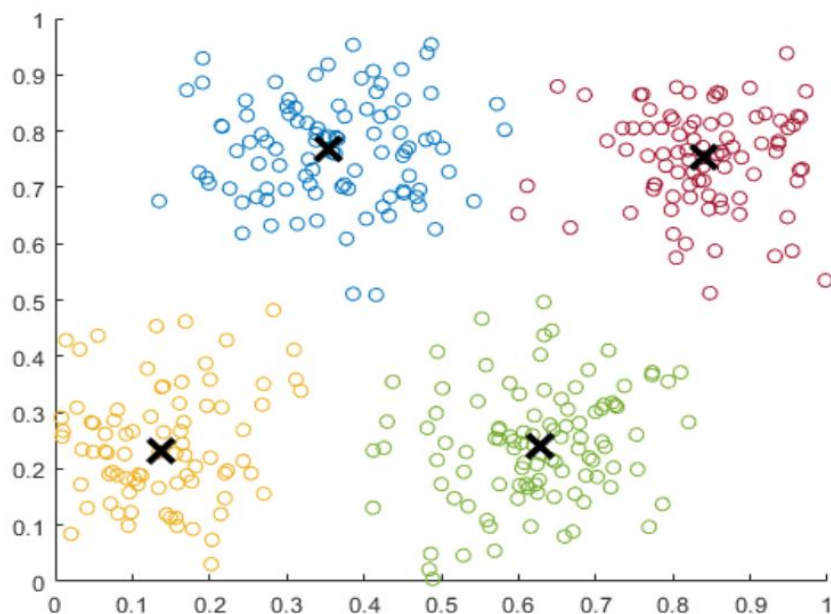


Figura 2.9: Ejemplo agrupamiento *Fuzzy C-Means* (Fuente: [18])

- **Agrupamiento probabilístico:** los grupos se crean haciendo uso de la distribución de probabilidad. Un ejemplo muy usado es el algoritmo EM (*Expectation – Maximization*) o de maximización de la esperanza.
- **Reducción de la dimensionalidad:** se utiliza cuando los conjuntos de datos presentan un gran número de variables aleatorias (también llamadas características). Esta técnica permite reducir este número para obtener un nuevo conjunto que representa esos mismos datos, pero con menor dimensionalidad (menor número de características). Esto se consigue, normalmente, aplicando uno de los siguientes métodos:

- **Selección de variables:** se eligen las variables más significativas del conjunto de datos como aquellas que poseen mayor relevancia en la predicción del resultado final.
- **Análisis de componentes principales (*Principal Component Analysis, PCA*):** [19] explica que esta técnica consiste en estudiar las relaciones entre las variables correlacionadas originales mediante la transformación del conjunto original de variables en otro conjunto de nuevas variables no correlacionadas entre sí, denominado conjunto de componentes principales. Estas nuevas variables son combinaciones lineales de las anteriores y se van construyendo según el orden de importancia en cuanto a la variabilidad total que recogen de la muestra. En la Figura 2.10 se recoge una explicación esquemática de esta técnica.

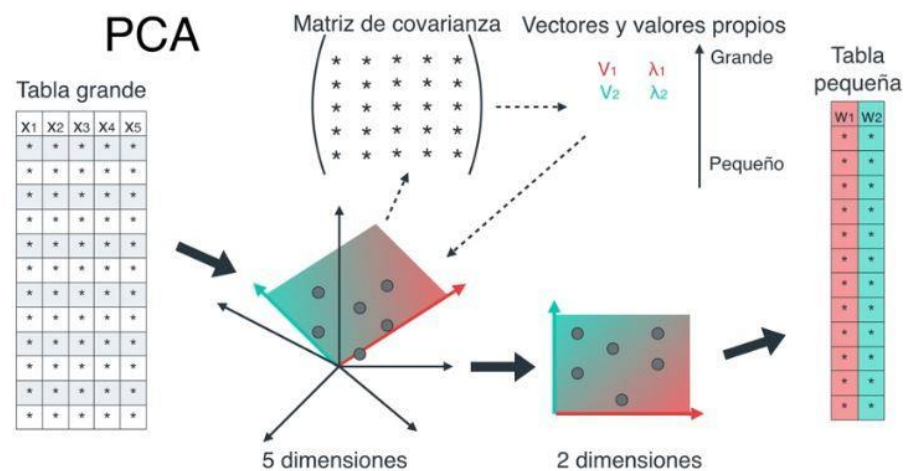


Figura 2.10: Técnica de análisis de componentes principales (*Fuente: [20]*)

- **Descomposición de valores singulares (*Singular Value Decomposition, SVD*):** consiste en aplicar el método de los valores singulares a la matriz formada por m filas de muestras y n columnas de características con el fin de reducir la dimensión de los datos. El concepto básico se resume en reducir el conjunto de datos mediante la descomposición de la matriz en sus valores singulares, obteniendo unos datos con menor dimensión que

reflejan más claramente su estructura y relaciones, y los ordena de mayor a menor variación, siendo posible ignorar aquellos cuya variación se encuentra por debajo de un determinado umbral, con lo que se pueden reducir asegurando que se mantienen las principales relaciones de interés. Es quizá de los tres métodos el que tiene un uso menos extendido.

## 2.3 APRENDIZAJE POR REFUERZO

En cuanto al aprendizaje por refuerzo, se trata de un modelo de aprendizaje conductual apoyado en el Proceso de Decisión de Markov. Consiste en que el agente (modelo) aprenda qué acciones escoger basándose en la prueba y error, mediante un sistema de “recompensas” y “castigos”, dentro de un ambiente determinado (entorno donde interactúa el agente, que contiene las limitaciones y reglas).

Es muy útil cuando se conoce el resultado al que se quiere llegar, pero se desconoce el camino a seguir para conseguirlo. En ningún caso se le indica al agente qué acciones tomar, sino que va experimentando mediante prueba y error qué acciones que toma maximizan la recompensa, y no solo de forma inmediata sino también a lo largo de todo el camino. Es decir, el agente es capaz de aprender de las experiencias y refinar su estrategia hasta conseguir la óptima.

La Figura 2.11 muestra un esquema de este proceso.

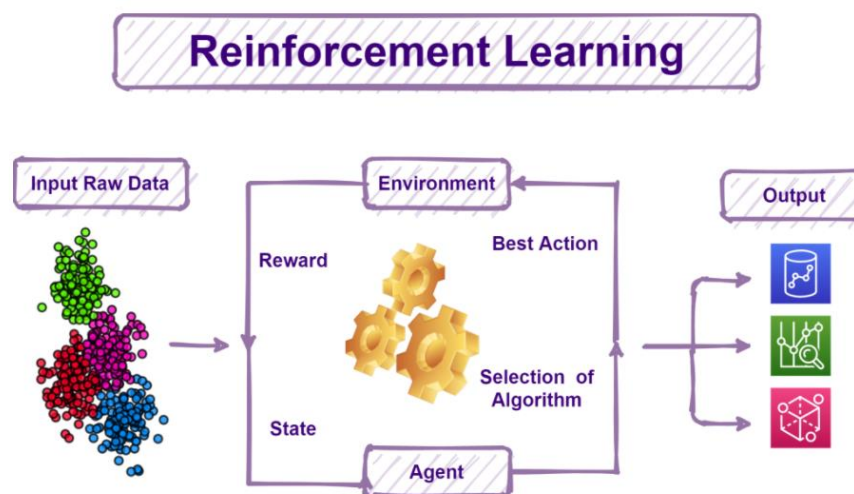


Figura 2.11: Esquema aprendizaje por refuerzo (Fuente: [21])

Algunas fuentes clasifican esta técnica dentro de un tipo particular de aprendizaje supervisado. Sin embargo, una característica diferenciadora es que más que hacer predicciones o clasificaciones para la toma de decisiones, el *reinforcement learning* es capaz de generar estrategias de manera automática.

Es muy utilizado en robótica, navegación autónoma, y personalización de opciones y recomendaciones.

## 2.4 APRENDIZAJE PROFUNDO Y REDES NEURONALES

Finalmente, el aprendizaje profundo o *deep learning* (DL) está diseñado para emular el funcionamiento del cerebro humano con el fin de que las máquinas sean capaces de tratar problemas abstractos y/o poco definidos. Para ello, incorpora redes neuronales artificiales en capas sucesivas para aprender de los datos de una manera iterativa.

Las redes neuronales artificiales (ANN – *Artificial Neural Networks*) son algoritmos que se basan en cómo funcionan las conexiones de las neuronas en el cerebro. Están constituidas por neuronas artificiales (o nodos) agrupadas en capas y conectadas entre sí.

La estructura más común de una red neuronal es la mostrada en la Figura 2.12.

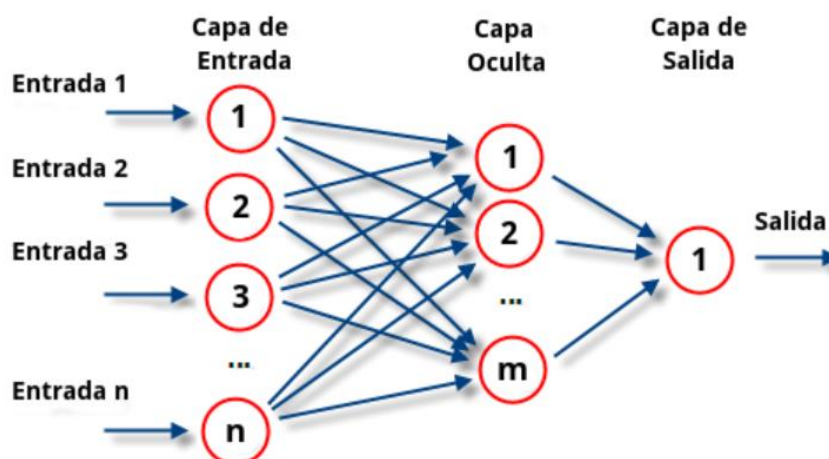


Figura 2.12: Estructura común de una red neuronal (Fuente: [22])

La primera capa (capa de entrada) está formada por un número de nodos igual a la cantidad de datos que se introducen. Cada uno de los nodos de la red tiene asignado un peso



(ponderación) con el que modifica la entrada que recibe, dando lugar a una salida que formará parte de la entrada a los nodos de la siguiente capa. De esta manera, se tendrán unos enlaces más fuertes o más débiles. En la salida de los nodos se puede establecer también una función (sigmoidea o logística) que establezca un umbral que no se puede superar para propagarse al siguiente nodo (función de activación).

En conclusión, la entrada a cada nodo de la capa oculta estará formada por el sumatorio de las salidas de los nodos de la capa anterior teniendo en consideración su peso. Para los nodos de la capa de salida se repite el proceso desde la capa oculta.

Las ANN siempre tienen una capa de entrada y una capa de salida, pudiendo tener ninguna o varias capas ocultas. Cuanto mayor sea el número de capas, más complejas son las redes y también más complejas son las funciones que pueden resolver.

Dependiendo de este número de capas, las redes neuronales se clasifican en:

- Red neuronal monocapa – perceptrón simple: sin capas ocultas
- Red neuronal multicapa – perceptrón multicapa: con capas ocultas

Además, en base al tipo de conexiones las redes pueden ser:

- Convolutiva (CNN, *Convolutional Neural Network*): cada nodo no tiene conexión con todos los nodos de la capa siguiente, sino que solo con un grupo (se especializa).
- No recurrentes: las conexiones van en un único sentido (de entrada hacia salida).
- Recurrentes: las neuronas se pueden retroalimentar (alimentación en ambos sentidos) con otras de la misma capa o de capas distintas, lo que les permite tener memoria. Dentro de este grupo, la arquitectura LSTM (*Long Short-Term Memory*) es una de las más conocidas y usadas.
- De base radial: la salida se calcula en función de la distancia a un punto llamado centro, de tal manera que es una combinación lineal de las funciones de activación radiales utilizadas por las neuronas individuales, [23].

La Figura 2.13 muestra un esquema de este funcionamiento.

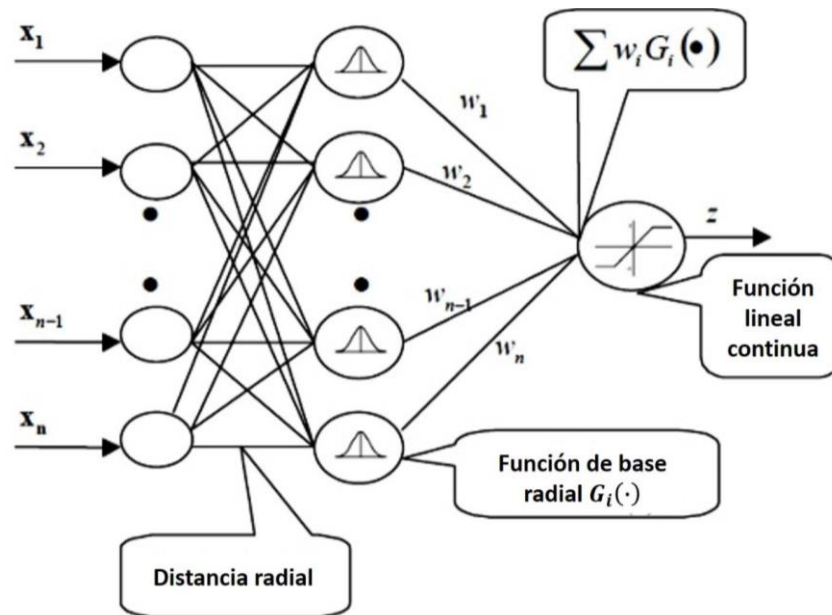


Figura 2.13: Red neuronal de base radial (Fuente: [24])

En un primer momento, las predicciones que realiza la red (las salidas) son aleatorias, pero va aprendiendo mediante el entrenamiento. Al introducirle registros para los que se conoce el resultado, la red va proporcionando salidas que, a su vez, se comparan con los resultados conocidos y en base al error cometido, se van ajustando las ponderaciones de los nodos. A medida que se va entrenando, la red se va haciendo más precisa y mejorando sus predicciones hasta que se consiguen los resultados buscados. Este método de entrenamiento se conoce como propagación hacia atrás o *backpropagation*.

Las ANN se utilizan ampliamente para problemas de clasificación y regresión dentro del aprendizaje supervisado. Sin embargo, a partir de este punto, el concepto de aprendizaje profundo va más allá.

Se habla de aprendizaje profundo cuando se tiene una estructura de redes neuronales en numerosas capas y el tratamiento de los datos se realiza de forma jerárquica, de tal manera que según se va avanzando en las sucesivas capas se tienen representaciones de los datos cada vez más significativas y el sistema es capaz de aprender elementos cada vez más complejos.

Esta diferencia entre lo que se denomina redes neuronales y el concepto de aprendizaje profundo se explica gráficamente en la Figura 2.14.

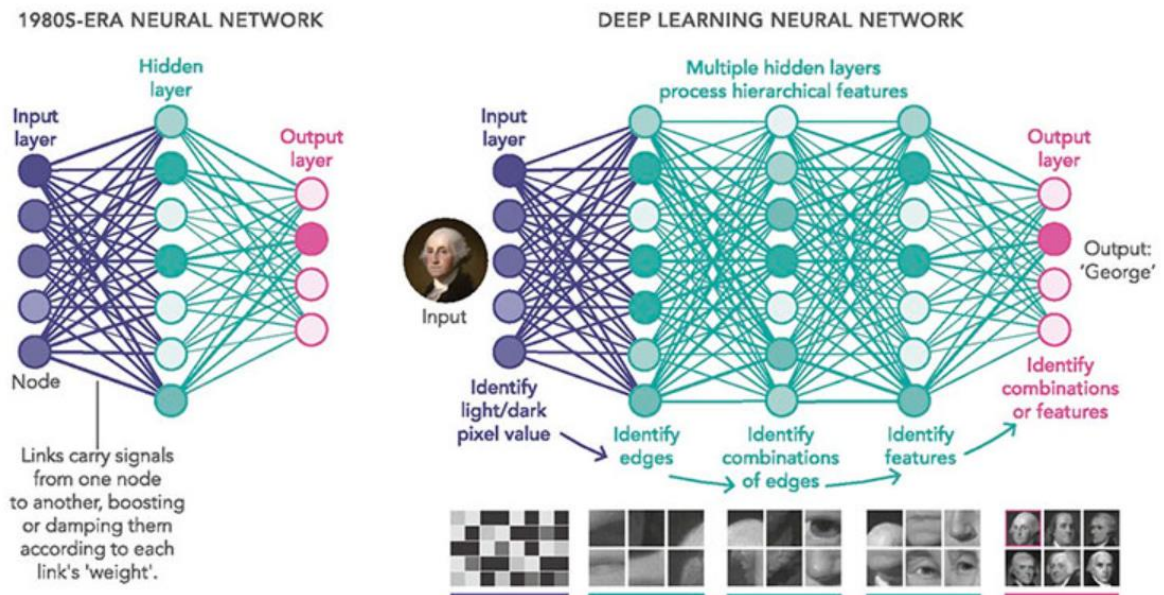


Figura 2.14: Redes neuronales VS Aprendizaje profundo (Fuente: [25])

El *deep learning* es muy efectivo en la identificación de patrones, y su uso está muy extendido en el mantenimiento predictivo, reconocimiento de imágenes, reconocimiento facial y de voz, prevención del fraude y pronóstico de enfermedades.

## 3. Estado del arte

### 3.1 REVISIÓN SISTEMÁTICA DE LA LITERATURA

Con el objetivo de identificar y analizar los estudios existentes relevantes en el ámbito de la aplicación de técnicas de *machine learning* y *business intelligence* en la gestión operativa de infraestructuras aeroportuarias, se lleva a cabo una revisión sistemática de la literatura (*Systematic Literature Review, SLR*).

Se opta por realizar una SLR en lugar de otros métodos o tipos de revisión porque se parte de una pregunta de investigación bien definida y, además, se quiere analizar en mayor profundidad cuáles son los distintos subcampos dentro del ámbito aeroportuario en los que más se está estudiando la aplicación de ML, así como cuáles son las técnicas más empleadas para cada subcampo. Y, para ello, la SLR aparece como el método que puede proporcionar unos mejores resultados por seguir un proceso sistemático preestablecido, con estrategias que limitan los sesgos y errores aleatorios.

El paso previo a la realización de esta revisión es hacer una búsqueda de otras posibles revisiones que se hubieran llevado a cabo para el mismo tema. La conclusión es que hasta el momento del inicio de este trabajo, no se ha llevado a cabo ninguna revisión sistemática de la literatura con el mismo objetivo que la que se propone. Por tanto, queda confirmada la necesidad de realizar esta SLR.

En la elaboración de la SLR se hace uso de la herramienta online “Parsifal” (para apoyar el desarrollo de las distintas fases de la revisión), así como de la aplicación “Zotero” (para la gestión de las publicaciones encontradas).

Las fases seguidas en el desarrollo de esta revisión son las siguientes:

1. Planificación de la revisión
2. Ejecución de la revisión
3. Realización del informe de la revisión

En los siguientes epígrafes se incluye una descripción más detallada de estas fases y los principales resultados obtenidos en cada una de ellas.

### 3.1.1 Planificación de la revisión

El primer paso en el desarrollo de la SLR es la elaboración de un protocolo de revisión, el cual queda definido en el diagrama de la Figura 3.1.

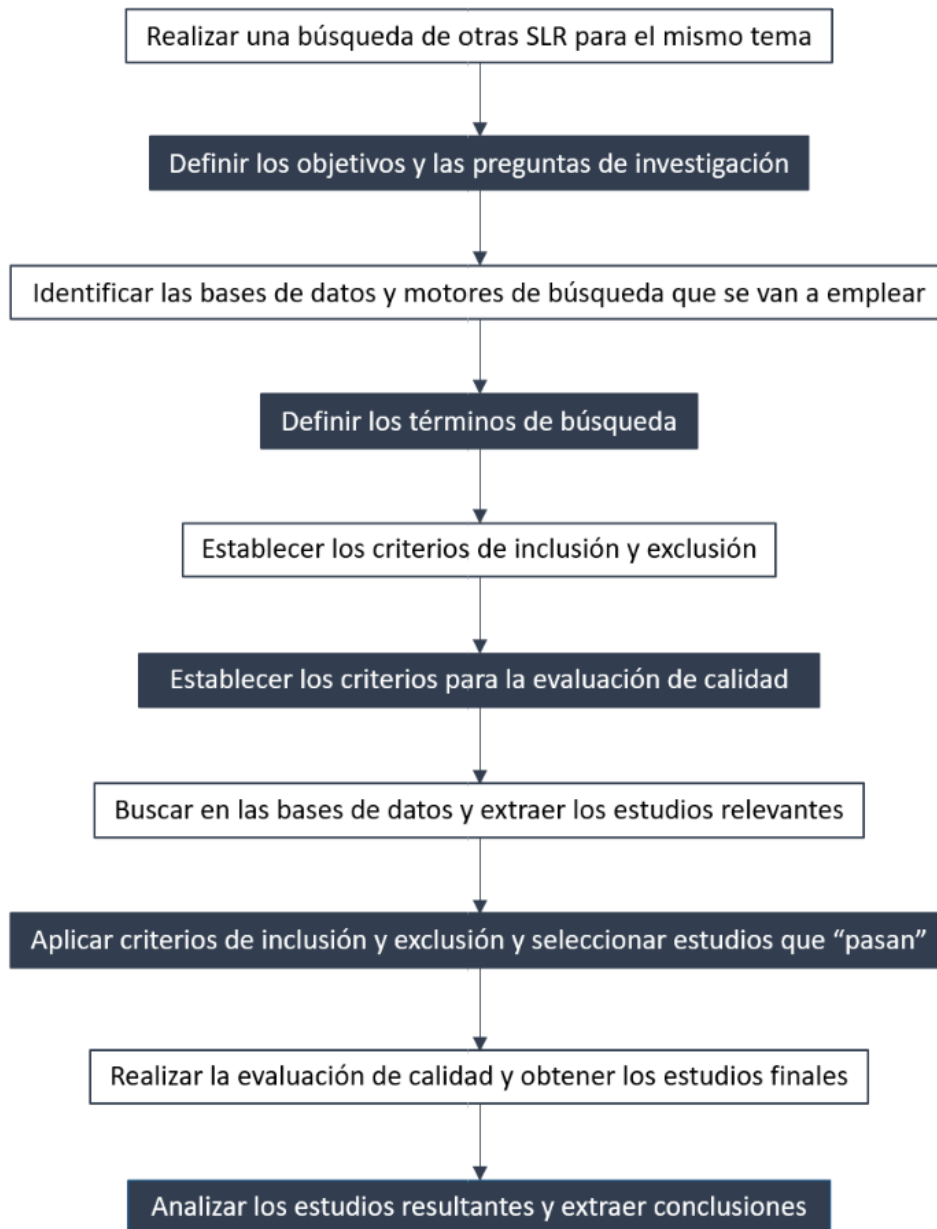


Figura 3.1: Definición del protocolo de revisión (*Fuente: elaboración propia*)

Como se ha comentado, la búsqueda de otras SLR confirma que no se ha llevado a cabo ninguna revisión sistemática de la literatura con el mismo objetivo que la que se propone.

Se sigue, por tanto, con los siguientes pasos. La Tabla 3.1 presenta el objetivo de la SLR.

Tabla 3.1. Definición del objetivo de la SLR (*Fuente: elaboración propia*)

Objetivo
<i>El objetivo de esta revisión es realizar una búsqueda y posterior análisis de los estudios existentes relevantes en el ámbito de la aplicación de técnicas de machine learning y business intelligence en la gestión operativa de infraestructuras aeroportuarias.</i>

Tras establecer el objetivo, se definen las siguientes preguntas de investigación:

- **PI-1: ¿Qué aplicaciones de *machine learning* se están usando en la gestión operativa de los aeropuertos?**
- **PI-2: ¿Cómo puede el *machine learning* apoyar la toma de decisiones para mejorar la eficiencia operativa y la gestión de recursos en los aeropuertos?**

A partir de las preguntas de investigación, se emplea el método PICOC para definir el ámbito de la revisión:

- **Población (P):** *Machine learning* y *business intelligence* en la gestión operativa aeroportuaria.
- **Intervención (I):** Aplicaciones de *machine learning* en los aeropuertos.
- **Comparación (C):** Sin comparación.
- **Resultados (O – Outputs):** Propuestas de uso de *machine learning* y *business intelligence* para mejorar la eficiencia operativa en los aeropuertos.
- **Contexto (C):** Contextos relacionados con el uso de *machine learning* en los aeropuertos.

Una vez en este punto, se eligen las bases de datos y motores de búsqueda que se van a emplear. En este caso, los principales criterios para la elección son: la calidad y relevancia en el ámbito científico, la disponibilidad y la capacidad de acceso a la información de manera gratuita. Además, para mantener la eficiencia en el proceso, se elige un número no muy elevado de fuentes de búsqueda, considerando que 4 pueden dar unos resultados de calidad. Con todo esto, la Tabla 3.2 establece las fuentes elegidas.

Tabla 3.2. Fuentes de búsqueda para la SLR (*Fuente: elaboración propia*)

Fuentes de búsqueda
IEEE Digital Library. <a href="http://ieeexplore.ieee.org">http://ieeexplore.ieee.org</a>
Science@Direct. <a href="http://www.sciencedirect.com">http://www.sciencedirect.com</a>
Scopus. <a href="http://www.scopus.com">http://www.scopus.com</a>
Springer Link. <a href="http://link.springer.com">http://link.springer.com</a>

El siguiente paso es la definición de las palabras clave y sus sinónimos, la cual se incluye en la Tabla 3.3. Parsifal permite tanto definir las como importarlas del método PICOC. En este caso, se utiliza una combinación de ambas opciones.

Tabla 3.3. Palabras clave y sinónimos (*Fuente: elaboración propia*)

Palabras clave	Sinónimos	Relacionado con
<i>Aeropuerto</i>	<i>Airport</i>	<i>Intervención</i>
<i>Business Intelligence</i>	-	<i>Población</i>
<i>Eficiencia</i>	<i>Efficiency</i>	<i>Resultados</i>
<i>Gestión</i>	<i>Management</i>	<i>Población</i>
<i>Gestión de recursos</i>	<i>Resources allocation</i>	<i>Población</i>
<i>Machine learning</i>	-	<i>Población</i>
<i>Operación</i>	<i>Operation, operational, operativa</i>	<i>Población</i>
<i>Toma de decisiones</i>	<i>Decision making</i>	<i>Población</i>

Con todo esto, se define una cadena de búsqueda genérica, que posteriormente se adapta a los requisitos de cada fuente de búsqueda utilizada. Esta cadena genérica se incluye en la Tabla 3.4.

Tabla 3.4. Definición de la cadena de búsqueda genérica (*Fuente: elaboración propia*)

<b>Cadena de búsqueda genérica</b>
<i>("machine learning" OR "business intelligence") AND ("eficiencia operativa" OR "operational efficiency" OR "operacion" OR "operation" OR "gestion de recursos" OR "resources allocation" OR "toma de decisiones" OR "decision making" OR "gestion" OR "management") AND ("aeropuerto" OR "airport")</i>

Una vez llegados a este punto, se definen los criterios de inclusión y exclusión, los cuales se aplican a los trabajos encontrados con el fin de definir cuáles son relevantes para obtener respuestas a las preguntas de investigación definidas. A la hora de establecer estos criterios, se tiene en cuenta que por el gran volumen de estudios que devuelven las búsquedas, estos han de poder aplicarse sin leer los trabajos en detalle, únicamente con el título y el resumen.

La Tabla 3.5 incluye los criterios de inclusión definidos, mientras que la Tabla 3.6 muestra los criterios de exclusión.

Tabla 3.5. Definición de los criterios de inclusión (*Fuente: elaboración propia*)

<b>Criterios de inclusión</b>
<ul style="list-style-type: none"><li>▪ <i>CI-1: La publicación es posterior a 2010 (inclusive)</i></li><li>▪ <i>CI-2: La publicación está en español o en inglés</i></li><li>▪ <i>CI-3: La publicación está publicada en revistas, libros, conferencias, workshops, o en las webs de organismos y empresas relevantes del sector</i></li><li>▪ <i>CI-4: La publicación trata sobre la aplicación de machine learning o business intelligence en los aeropuertos</i></li><li>▪ <i>CI-5: La versión completa de la publicación está disponible a través de las suscripciones de las que disponemos</i></li></ul>



Tabla 3.6: Definición de los criterios de exclusión (*Fuente: elaboración propia*)

Criterios de exclusión
<ul style="list-style-type: none"><li>▪ <i>CE-1: La publicación es anterior a 2010</i></li><li>▪ <i>CE-2: La publicación no está en español o en inglés</i></li><li>▪ <i>CE-3: La publicación no está publicada en revistas, libros, conferencias, workshops, o en las webs de organismos y empresas relevantes del sector</i></li><li>▪ <i>CE-4: La publicación no trata sobre la aplicación de machine learning o business intelligence en los aeropuertos</i></li><li>▪ <i>CE-5: La versión completa de la publicación no está disponible a través de las suscripciones de las que disponemos</i></li></ul>

La justificación de elegir el año 2010 como límite de antigüedad de las publicaciones viene dada fundamentalmente porque se trata de una disciplina cuya aplicación a la gestión aeroportuaria es muy reciente, teniendo su mayor crecimiento en los últimos cinco años, y, además, los conceptos aplicables evolucionan a un ritmo acelerado, por lo que se considera que publicaciones anteriores a ese año tienen escaso interés para este trabajo. Por otra parte, como idiomas de las publicaciones se limitan a español e inglés debido, por una parte, a que el español permite identificar publicaciones de estudios que se estén llevando a cabo a nivel nacional, mientras que la mayor parte de las publicaciones internacionales están redactadas en inglés. Además, incorporar otros idiomas de los que no se tiene un conocimiento adecuado para realizar la búsqueda y el análisis podría distorsionar los resultados.

El resto de los criterios se establecen con el fin de que los resultados obtenidos cumplan unas condiciones de mínimas de relevancia y accesibilidad.

Una vez establecidas estas pautas, se definen los criterios de calidad que se aplican a los trabajos que han pasado el filtro de los criterios de inclusión y exclusión. Para ello, se proponen una serie de preguntas de calidad, cuyas respuestas se evalúan asignándoles un valor de una escala predefinida. Estas preguntas de calidad se recogen en la Tabla 3.7.

Tabla 3.7. Definición de los criterios de calidad (*Fuente: elaboración propia*)

Criterios de calidad
<ul style="list-style-type: none"><li>▪ CC-1: ¿Incluye una descripción del contexto en el que se realizó el estudio?</li><li>▪ CC-2: ¿Incluye un número adecuado de referencias bibliográficas?</li><li>▪ CC-3: ¿Incluye un apartado con las conclusiones del estudio?</li><li>▪ CC4: ¿Responde a todas las preguntas de investigación adecuadamente?</li></ul>

Puntuación de las respuestas:

- SI: 1 punto
- Parcialmente: 0,5 puntos
- NO: 0 puntos

Por tanto, la puntuación máxima que cada trabajo puede obtener es de 4 puntos. Se establece una puntuación de corte de 3 puntos, de tal manera que los trabajos seleccionados son los que obtengan una puntuación superior. La justificación de esta decisión de puntuación mínima viene dada para garantizar que no “pase” ningún estudio cuya respuesta sea “NO” a alguna de las preguntas de calidad.

### 3.1.2 Ejecución de la revisión

Una vez terminada la planificación de la revisión, se procede con su ejecución.

Cabe señalar que, por simplicidad en el proceso, los criterios de inclusión y exclusión CI-1 y CE-1, CI-2 y CE-2, y CI-5 y CE-5 se aplican directamente al realizar las búsquedas en las bases de datos, ya que estas permiten aplicar estos filtros. Con esto se ahorra una cantidad de esfuerzo y tiempo importante.

La cadena de búsqueda genérica que se ha definido se tiene que adaptar ligeramente al formato de cada base de datos. No obstante, se garantiza que las consultas son equivalentes para que los resultados obtenidos sean comparables. Se incluyen en las Tabla 3.8, Tabla 3.9, Tabla 3.10 y Tabla 3.11 las cadenas de búsqueda empleadas en cada una de las cuatro fuentes establecidas.

Tabla 3.8. Cadena de búsqueda en IEEE Digital Library (*Fuente: elaboración propia*)

**Cadena de búsqueda IEEE Digital Library**

("machine learning" OR "business intelligence") AND ("eficiencia operativa" OR "operational efficiency" OR "operacion" OR "operation" OR "gestion de recursos" OR "resources allocation" OR "toma de decisiones" OR "decision making" OR "gestion" OR "management") AND ("aeropuerto" OR "airport")

Tabla 3.9. Cadena de búsqueda en Science Direct (*Fuente: elaboración propia*)

**Cadena de búsqueda Science Direct**

("machine learning" OR "business intelligence") AND ("operational efficiency" OR "operation" OR "resources allocation" OR "decision making" OR "management") AND ("airport")

Tabla 3.10. Cadena de búsqueda en Springer Link (*Fuente: elaboración propia*)

**Cadena de búsqueda Springer Link**

("machine learning" OR "business intelligence") AND ("eficiencia" OR "efficiency" OR "operacion" OR "operation" OR "operational" OR "operativa" OR "gestion de recursos" OR "resources allocation" OR "toma de decisiones" OR "decision making" OR "gestion" OR "management") AND ("aeropuerto" OR "airport")

Tabla 3.11. Cadena de búsqueda en Scopus (*Fuente: elaboración propia*)

**Cadena de búsqueda Scopus**

( "machine learning" OR "business intelligence" ) AND ( "eficiencia operativa" OR "operational efficiency" OR "operacion" OR "operation" OR "gestion de recursos" OR "resources allocation" OR "toma de decisiones" OR "decision making" OR "gestion" OR "management" ) AND ( "aeropuerto" OR "airport" ) AND ( LIMIT-TO ( OA , "all" ) ) AND ( LIMIT-TO ( PUBYEAR , 2021 ) OR LIMIT-TO ( PUBYEAR , 2020 ) OR LIMIT-TO ( PUBYEAR , 2019 ) OR LIMIT-TO ( PUBYEAR , 2018 ) OR LIMIT-TO ( PUBYEAR , 2017 ) OR LIMIT-TO ( PUBYEAR , 2016 ) OR LIMIT-TO ( PUBYEAR , 2015 ) OR LIMIT-TO (

### Cadena de búsqueda Scopus

*PUBYEAR, 2014) OR LIMIT-TO (PUBYEAR, 2013) OR LIMIT-TO (PUBYEAR, 2012) OR LIMIT-TO (PUBYEAR, 2011) OR LIMIT-TO (PUBYEAR, 2010)) AND (LIMIT-TO (LANGUAGE, "English") OR LIMIT-TO (LANGUAGE, "Spanish"))*

Los resultados de la búsqueda en cada base de datos se exportan a los archivos .csv o BibTeX correspondientes y, a partir de estos, se importan en Parsifal.

El número total de resultados de partida obtenido es de 3011 trabajos, siendo el reparto por bases de datos el mostrado en la Tabla 3.12 y, gráficamente, en la Figura 3.2.

Tabla 3.12. Número de resultados en cada base de datos (*Fuente: elaboración propia*)

Fuente	Número de trabajos
<i>IEEE Digital Library</i>	49
<i>Science Direct</i>	114
<i>Scopus</i>	907
<i>Springer Link</i>	1941

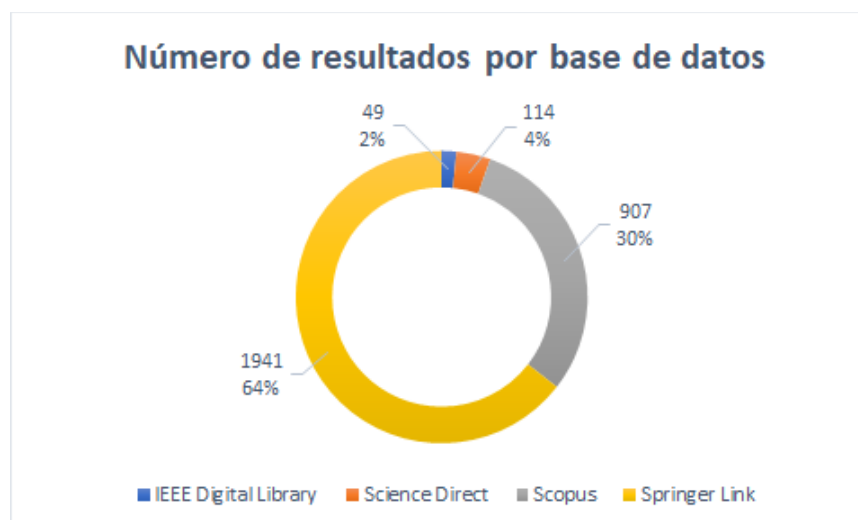


Figura 3.2. Número de resultados por base de datos (*Fuente: elaboración propia*)

Por otra parte, se realiza una búsqueda manual de comprobación con el fin de detectar posibles estudios que no hayan sido detectados durante las búsquedas en las bases de datos. En esta se encuentra 1 estudio adicional, que se añade a la lista de encontrados. A partir de estos resultados de base, se emplea la opción de Parsifal de *Find Duplicates* para identificar los trabajos duplicados. En total, se han encontrado 110 duplicados. Una vez eliminados los duplicados, se procede a clasificar los trabajos en base a los criterios de inclusión y exclusión definidos. Esta clasificación conduce a seleccionar un total 142 trabajos que pasan a la siguiente fase de evaluación de calidad, como muestra la Figura 3.3.



Figura 3.3. Porcentaje de trabajos aceptados y descartados (*Fuente: elaboración propia*)

Se procede, por tanto, a aplicar los criterios de calidad definidos a estos 142 trabajos seleccionados. Los resultados de esta evaluación llevan a seleccionar finalmente 48 trabajos que cumplen con todos los criterios establecidos en esta SLR.

A continuación, la Figura 3.4 incluye un diagrama de los resultados expuestos siguiendo el modelo PRISMA.

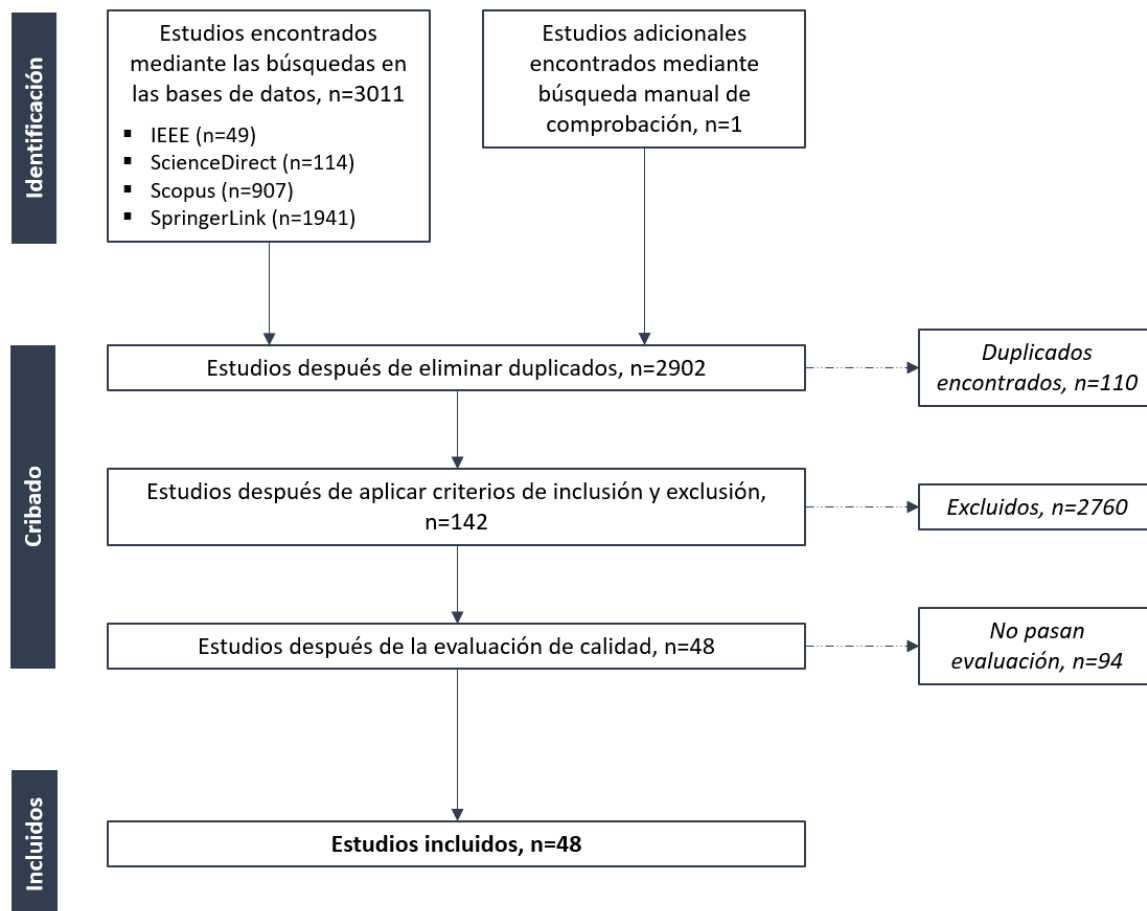


Figura 3.4: Diagrama modelo PRISMA de los resultados (Fuente: elaboración propia)

### 3.1.3 Reporte de resultados de la revisión

Dentro de la temática de estos 48 estudios incluidos, se identifican 10 campos de estudio principales en los que se pueden categorizar:

1. Predicción de demoras: se incluyen dentro de este campo los trabajos que tratan sobre la aplicación de *machine learning* para predecir demoras generales en los aeropuertos o subsistemas de los mismos.
2. Meteorología: se incluyen los trabajos acerca del modelado y la predicción de fenómenos meteorológicos que pueden afectar a las operaciones de las aeronaves (meteorología aeronáutica).

3. Rodaje de las aeronaves: abarca tanto herramientas para predecir el tiempo de rodaje como para realizar una gestión inteligente de las velocidades de rodaje de las aeronaves en el área de movimientos.
4. Edificio terminal: dentro de este campo se agrupan los trabajos relativos a los distintos procesos y servicios que se realizan dentro del edificio terminal de pasajeros.
5. Pasajeros: abarca los trabajos enfocados en la predicción del volumen de pasajeros y en el análisis de su comportamiento.
6. Operaciones: dentro de este campo se incluyen los trabajos que tratan sobre aplicaciones dentro del ámbito de las operaciones en los aeropuertos.
7. Tratamiento de equipajes: agrupa los trabajos sobre el uso de *machine learning* en el BHS (*Baggage Handling System*).
8. Ruido: se incluyen los trabajos que proponen el uso de *machine learning* para la gestión del ruido proveniente de la operación de las aeronaves en el aeropuerto.
9. General: este campo comprende los trabajos que proponen un enfoque más general de los aeropuertos.
10. Mantenimiento de campo de vuelos: en este campo se incorporan los trabajos relativos al empleo de técnicas de *machine learning* para apoyar las decisiones de mantenimiento sobre elementos del campo de vuelos.

La Tabla 3.13 incluye el listado de estos 48 trabajos obtenidos, divididos en función de los temas expuestos e incluyendo los datos más relevantes.

El número de datos incluidos en esta tabla se han limitado al tamaño legible de la misma. No obstante, más adelante en el documento se incluyen tablas adicionales para cada campo de estudio que contienen mayor detalle de los trabajos.

Tabla 3.13: Resultados obtenidos en la SLR (Fuente: elaboración propia)

ID	Tema	Título	Autores	Año
1	Predicción de demoras	<i>A Deep Graph-Embedded LSTM Neural Network Approach for Airport Delay Prediction</i>	W. Zeng, J. Li, Z. Quan, and X. Lu	2021
2		<i>A Methodology for Predicting Aggregate Flight Departure Delays in Airports Based on Supervised Learning</i>	B. Ye, B. Liu, Y. Tian, and L. Wan	2020
3		<i>A New Multilevel Input Layer Artificial Neural Network for Predicting Flight Delays at JFK Airport</i>	Sina Khanmohammadi, Salih Tunun, and Yunus Kucuk	2016
4		<i>A system for effectively predicting flight delays based on IoT data</i>	Abdulwahab Aljubairy, Wei Emma Zhang, Ali Shemshadi, Adnan Mahmood, and Quan Z. Sheng	2020
5		<i>Real-Time Investigation of Flight Delays Based on the Internet of Things Data</i>	Abdulwahab Aljubairy, Ali Shemshadi, and Quan Z. Sheng	2016
6		<i>Characterization and prediction of air traffic delays</i>	Juan Jose Rebollo and Hamsa Balakrishnan	2014
7		<i>Development of a predictive model for on-time arrival flight of airliner by discovering correlation between flight and weather data</i>	Noriko Etani	2019



ID	Tema	Título	Autores	Año
8		<i>Flight delay prediction based on deep learning and Levenberg-Marquart algorithm</i>	Maryam Farshchian Yazdi, Seyed Reza Kamel, Seyyed Javad Mahdavi Chabok, and Maryam Kheirabadi	2020
9		<i>Flight Delay Prediction using Airport Situational Awareness Map</i>	Wei Shao, Siyu Tan, Xinhong Hei, Arian Prabowo, Piort Koniusz, Bradley Feest, Sichen Zhao, Jeffrey Chan, and Flora D. Salim	2019
10		<i>Flight Delay Prediction Using Deep Convolutional Neural Network Based on Fusion of Meteorological Data</i>	Jingyi Qu, Ting Zhao, Meng Ye, Jiayi Li, and Chao Liu	2020
11		<i>Ground Delay Program Analytics with Behavioral Cloning and Inverse Reinforcement Learning</i>	Michael Bloem and Nicholas Bambos	2015
12		<i>Predicting Ground Delay Program At An Airport Based on Meteorological Conditions</i>	Avijit Mukherjee, Shon Grabbe, and Banavar Sridhar	2014
13		<i>Prediction of air traffic delays: An agent-based model introducing refined parameter estimation methods</i>	Chunzheng Wang, Minghua Hu, Lei Yang, and Zheng Zhao	2021
14		<i>Predictive Modelling: Flight Delays and Associated Factors, Hartsfield – Jackson Atlanta International Airport</i>	Roberto Henriques and Inês Feiteira	2018

ID	Tema	Título	Autores	Año
15		<i>Prediction of Flight Time Deviation for Lithuanian Airports Using Supervised Machine Learning Model</i>	Pavel Stefanovic, Rokas Strimaitis, and Olga Kurasova	2021
16	Meteorología	<i>Clustering days with similar airport weather conditions</i>	Grabbe S., Srdhar B., and Mukherjee A.	2014
17		<i>Modeling Weather Impact on Ground Delay Programs</i>	Yao Wang and Deepak Kulkarni	2011
18		<i>Prediction of weather impacted airport capacity using ensemble learning</i>	Yao Wang	2011
19		<i>Fog prediction using artificial intelligence: A case study in Wamena Airport</i>	Ristiana Dewi, Prawito, and Hastuadi Harsa	2019
20		<i>Efficient Prediction of Low-Visibility Events at Airports Using Machine-Learning Regression</i>	L. Cornejo-Bueno, C. Casanova-Mateo, J. Sanz-Justo, E. Cerro-Prada, and S. Salcedo-Sanz	2017
21		<i>Persistence Analysis and Prediction of Low-Visibility Events at Valladolid Airport, Spain</i>	Sara Cornejo-Bueno, David Casillas-Pérez, Laura Cornejo-Bueno, Mihaela I. Chidean, Antonio J. Caamaño, Julia Sanz-Justo, Carlos Casanova-Mateo, and Sancho Salcedo-Sanz	2020

ID	Tema	Título	Autores	Año
22		<i>Visibility Forecast for airport operations by LSTM Neural Networks</i>	Tuo Deng, Aijie Cheng, Wei Han, and Hai-Xiang Lin	2019
23	Rodaje de las aeronaves	<i>Machine Learning Techniques for Taxi-out Time Prediction with a Macroscopic Network Topology</i>	Jianan Yin, Yuxin Hu, Yuanyuan Ma, Yan Xu, Ke Han, and Dan Chen	2018
24		<i>Predicting Taxi-Out Time at Congested Airports with Optimization-Based Support Vector Regression Methods</i>	Guan Lian, Yaping Zhang, Jitamitra Desai, Zhiwei Xing, and Xiao Luo	2018
25		<i>Taxi Time Prediction at Charlotte Airport Using Fast-Time Simulation and Machine Learning Techniques</i>	Hanbong Lee, Waqar Malik, Bo Zhang, Balaji Nagarajan, and Yoon C. Jung	2015
26		<i>Taxi-Out Time Prediction for Departures at Charlotte Airport Using Machine Learning Techniques</i>	Hanbong Lee and Waqar Malik	2016
27		<i>Taxiing Speed Intelligent Management of Aircraft Based on DQN for A-SMGCS</i>	Yunfan Zhou, Weilong Liu, Yandong Li, and Bo Jiang	2019
28		<i>Aircraft taxi time prediction: Comparisons and insights</i>	Stefan Ravizza, Jun Chen, Jason A.D. Atkin, Paul Stewart, and Edmund K. Burke	2014
29	Edificio Terminal	<i>Airport Restroom Cleanliness Prediction Using Real Time User Feedback Data</i>	Kilian Ros, Elena Mocanu, and Christin Seifert	2019

ID	Tema	Título	Autores	Año
30	Pasajeros	<i>Hub Airport 4.0 – How Frankfurt Airport Uses Predictive Analytics to Enhance Customer Experience and Drive Operational Excellence</i>	Rolf Felkel, Dieter Steinmann, and Frank Follert	2018
31		<i>Predicting Passenger Flow at Charles De Gaulle Airport Security Checkpoints</i>	Philippe Monmousseau, Gabriel Jarry, Florian Bertosio, Daniel Delahaye, and Marc Houalla	2020
32		<i>Neural Networks trained with WiFi traces to predict airport passenger behavior</i>	Federico Orsini, Massimiliano Gastaldi, Luca Mantecchini, and Riccardo Rossi	2019
33		<i>Queue behavioural patterns for passengers at airport terminals: A machine learning approach</i>	Álvaro Rodríguez-Sanz et al.	2021
34	Pasajeros	<i>A passenger risk assessment method based on 5G-IoT</i>	Weishi Chen, Yifeng Huang, Hao Yang, Jing Li, and Xianfeng Lu	2021
35		<i>Integrating Qualitative Comparative Analysis and Support Vector Machine Methods to Reduce Passengers' Resistance to Biometric E-Gates for Sustainable Airport Operations</i>	Cheong Kim, Francis Joseph Costello, and Kun Chang Lee	2019
36		<i>Forecasting Air Passenger Demand: A Comparison of LSTM and SARIMA</i>	Quang Hung Do, Shih-Kuei Lo, Jeng-Fung Chen, Chi-Luan Le, and Luong Hoang Anh	2020

ID	Tema	Título	Autores	Año
37		<i>Forecasting air passenger traffic flow based on the two-phase learning model</i>	Xinfang Wu, Yong Xiang, Gang Mao, Mingqian Du, Xiuqing Yang, and Xinzhi Zhou	2021
38	Operaciones	<i>Applying Machine Learning Modeling to Enhance Runway Throughput at A Big European Airport</i>	G. Stempfel, I. De Visscher, M. Ellejmi, V. Brossard, A. Bonnefoy, and V. Treve	2020
39		<i>Behavioral Learning of Aircraft Landing Sequencing Using a Society of Probabilistic Finite State Machines</i>	Jiangjun Tang and Hussein A. Abbass	2014
40		<i>Research on Situational Awareness of Airport Operation Based on Petri Nets</i>	Qian Luo, Yanru Chen, Lunyue Chen, Xiao Luo, Huan Xia, Yuanyuan Zhang, and Liangyin Chen	2019
41	Tratamiento de equipajes	<i>Predictive Process Monitoring in Operational Logistics: A Case Study in Aviation</i>	Björn Rafn Gunnarsson, Seppe K. L. M. vandem Broucke, and Jochen De Weerd	2019
42		<i>An Approach for Adaptive Threat Recognition Within 3D Computed Tomography Images for Baggage Security Screening</i>	Qian Wang, Khalid N. Ismail, and Toby P. Breckon	2019
43		<i>Extraction of stable foreground image regions for unattended luggage detection</i>	Grzegorz Szwoch	2016

ID	Tema	Título	Autores	Año
44	Ruido	<i>An Airport Noise Prediction Model Based on Selective Ensemble of LOF-FSVR</i>	Chen H., Deng J., Sun B., and Wang J.	2015
45		<i>Application of intelligent instruments for the monitoring of thrust reverse noise at airports</i>	C. Asensio, M. Ruíz, M. Recuero, G. Moschioni, and M. Tarabini	2013
46	General	<i>A Short Review of the Application of Machine Learning Methods in Smart Airports</i>	Huang Hailong, and Zhu Jinfu	2020
47		<i>The evolution of airports from a network perspective – an analytical concept</i>	Mark Azzam	2015
48	Mantenimiento del campo de vuelos	<i>A Broad-Based Decision-Making Procedure for Runway Friction Decay Analysis in Maintenance Operations</i>	Salvatore Antonio Biancardo, Francesco Abbodanti, Francesca Russo, Rosa Varopalumbo, and Gianluca Dell' Acqua	2020

Gráficamente, la división de los 48 trabajos en estos campos de estudio es la mostrada en la Figura 3.5.

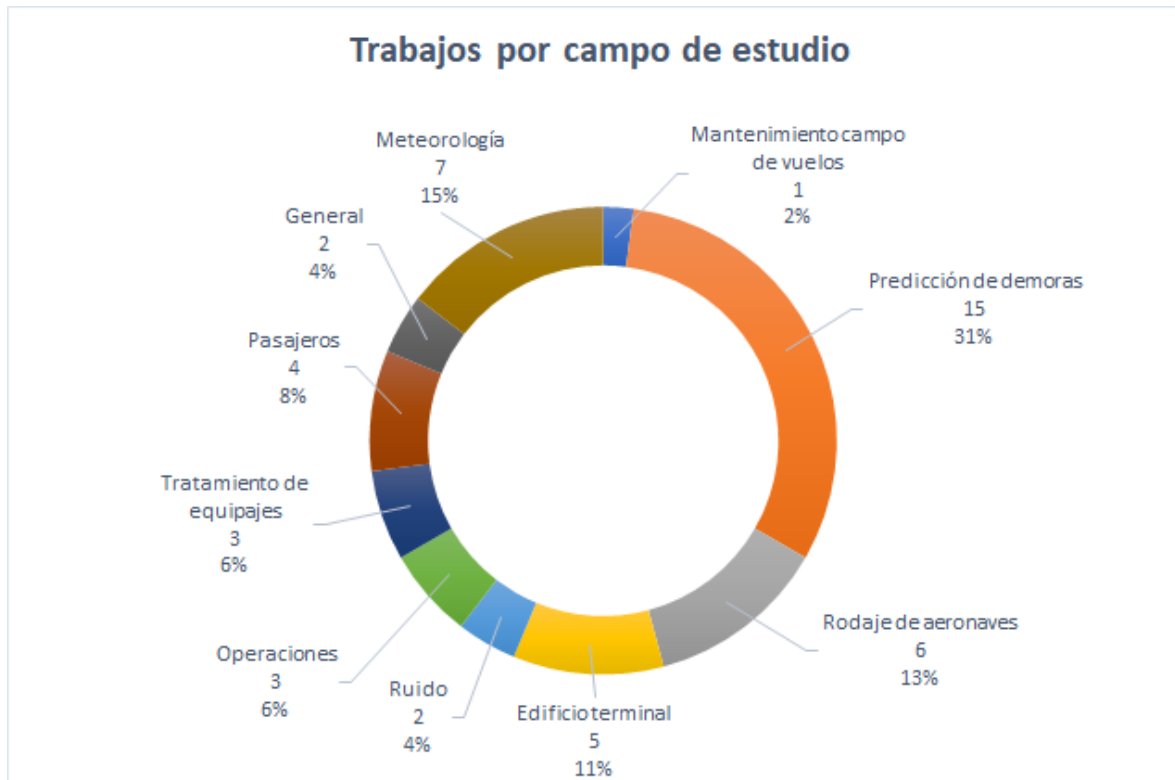


Figura 3.5. Trabajos seleccionados por campo de estudio (*Fuente: elaboración propia*)

Como se puede observar, el campo con mayor número de trabajos publicados es la “predicción de demoras”, con 15 trabajos.

A este respecto, en [29] se propone un modelo de predicción de retrasos basado en una red neuronal profunda para realizar el estudio desde la perspectiva de una red de aeropuertos. En él, los aeropuertos son considerados como nodos de una red de gráficos y se utiliza una red de gráficos dirigida para construir la relación entre los mismos. Para los aeropuertos adyacentes, los pesos de las aristas se miden por la distancia esférica entre ellos, mientras que para los aeropuertos conectados por vuelos se utiliza el número de pares de vuelos entre ellos. Sobre esta base, se construye un núcleo de convolución de difusión para capturar las características de la propagación de los retrasos entre aeropuertos, y se integra además en la red neuronal LSTM de secuencia a secuencia para

establecer un marco de aprendizaje profundo para la predicción de demoras. Este modelo se denomina LSTM profunda embebida en gráficos (DGLSTM). Para la verificación del modelo, se utilizan los datos históricos de retrasos de 325 aeropuertos en Estados Unidos desde 2015 hasta 2018 como el conjunto de entrenamiento del modelo y el conjunto de prueba. Los resultados experimentales sugieren que el método propuesto es superior a los principales métodos existentes en términos de precisión y robustez.

Por su parte, en [30] se propone una metodología para predecir los retrasos agregados en las salidas de los vuelos en los aeropuertos basada en la exploración de métodos de aprendizaje supervisado, en particular, se investigan cuatro de estos métodos: regresión lineal múltiple, máquina de vectores de soporte, árboles extremadamente aleatorios y GBM ligero. Los modelos propuestos se entrenan y validan utilizando datos operativos desde marzo de 2017 a febrero de 2018 para el Aeropuerto Internacional de Nanjing Lukou en China. Los resultados muestran que, para un horizonte de predicción de 1 hora, el modelo GBM ligero ofrece el mejor resultado, con un índice de precisión de 0,8655 y un error absoluto medio de 6,65 minutos, lo que supone 1,83 minutos menos que los resultados de investigaciones anteriores.

Además, en [31] se introduce una nueva estructura ANN, denominada DMP-ANN (*Defect of Modules Prediction – ANN*), con el fin de salvar las dificultades de la ANN con variables nominales. La capa de entrada en la DMP-ANN propuesta consta de varias subcapas en las que una o más neuronas están activas (salida=1) y otras en las que están inactivas (salida=0). Por tanto, el proceso de aprendizaje implica la actualización de los pesos de las neuronas activas. Como caso de estudio, este modelo se aplica para predecir el retraso de los vuelos entrantes en el aeropuerto JFK, donde las neuronas de cada subcapa de la capa de entrada simbolizan las fuentes de retraso en los diferentes niveles del sistema, y la activación de cada neurona representa la posibilidad de ser la fuente de retraso global. Los resultados obtenidos muestran que la DMP-ANN es capaz de superar al método tradicional ANN en cuanto al error de predicción (error medio cuadrático) y al tiempo necesario para entrenar al modelo. No obstante, una de las limitaciones detectadas es la complejidad del método propuesto (a medida que el número de variables aumenta, el número de conexiones también lo hace de manera significativa).



Asimismo, [32] plantea el estudio de las demoras desde una nueva perspectiva, empleando datos obtenidos del IoT en lugar de basarse únicamente en datos históricos. Para ello, desarrolla una herramienta automática que recoge los datos de IoT de varias fuentes, incluyendo datos de vuelo, meteorológicos y de índices de calidad del aire. A partir de estos datos heterogéneos se desarrolla un algoritmo que fusiona las características de las distintas fuentes; y finalmente se adopta la modelización predictiva para estudiar los factores que contribuyen a los retrasos de los vuelos y para predecir los retrasos de los vuelos en el futuro. A este último respecto, se desarrollan dos tipos de modelos: regresión múltiple logística (para clasificar el vuelo en retrasado o en hora) y regresión múltiple lineal (para predecir el tiempo de demora en minutos). Tres de los autores de este trabajo, A. Aljubairy, A. Shemshadi y Q. Z. Sheng ya habían introducido este concepto en otro trabajo anterior, [33].

J. J. Rebollo y H. Balakrishnan proponen en [34] una nueva clase de modelos para predecir los retrasos en el tráfico aéreo. Estos modelos consideran los estados de retraso temporales y espaciales (es decir, de la red) como variables explicativas, y utilizan algoritmos de Random Forest para predecir los retrasos de salida de las siguientes 2 – 24 horas. En su trabajo analizan el rendimiento de los modelos de predicción propuestos tanto en la clasificación de los retrasos por encima o por debajo de un determinado umbral como en la predicción de los valores de retraso. Los modelos se entrenan y validan con datos operativos de 2007 y 2008 de la *Aviation System Performance Metrics (ASPM)* de la FAA, y se evalúan utilizando los 100 enlaces más retrasados del sistema. Los resultados muestran un error de prueba medio del 19% al clasificar los retrasos como superiores o inferiores a 60 minutos, para un horizonte de previsión de 2 horas. Además, tanto los modelos de clasificación como los de regresión resultan bastante robustos ante el aumento del horizonte de previsión: la mediana del error de la prueba de regresión (promediada en los 100 pares de OD) solo aumenta de 19,1 min a 27,4 min cuando el horizonte de previsión pasa de 2 h a 24 h.

Por su parte, N. Etani desarrolla en [35] un modelo predictivo de las demoras en llegadas mediante la correlación de datos de vuelo con datos meteorológicos. Para ello, implementa diferentes algoritmos de aprendizaje automático supervisado. Como resultado, consigue

una predicción de la hora de llegada con una exactitud del 77% utilizando el modelo del clasificador Random Forest.

En [36] se presenta un modelo de predicción de demoras basado en *Deep Learning* que utiliza el algoritmo de Levenberg-Marquart (LM), incluyendo además el diseño de una técnica basada en *autoencoders* para la eliminación del ruido en los datos de demora de partida (SDA – *Stack Denoising Autoencoder*). Además de la estructura completa, SDA-LM, se desarrollan otras dos estructuras adicionales: SAE-LM (*Stack Autoencoder* y LM) y SDA (únicamente SDA, sin LM). Para analizar las tres estructuras, se emplea un conjunto de datos de vuelos en EEUU, los cuales son no balanceados. Para balancear el conjunto de datos se emplea la técnica de submuestreo (*undersampling*). Los resultados concluyen que tanto para el conjunto de datos no balanceados como para los balanceados, la precisión, exactitud, sensibilidad, exhaustividad (*recall*) y valor F del modelo SDA-FM son mejores que los de los modelos SAE-LM y SDA. Con todo, se concluye que la introducción de SDA y de LM, así como el balanceo de los datos, tiene un efecto positivo en la predicción.

El trabajo expuesto en [37] propone la inclusión del mapa de conocimiento situacional del aeropuerto (*situational awareness map*) como factor para la predicción de demoras. Este mapa se refiere a la complejidad del tráfico en el aeropuerto en cuanto a las trayectorias de los vuelos y vehículos en el mismo. Con ello, se desarrolla un modelo de predicción de demoras en salidas partiendo de datos de vuelo, datos meteorológicos y datos del mapa de conocimiento situacional, y aplicando 4 modelos de *machine learning* diferentes: regresor lineal (LR), regresor de vectores de soporte (SVR), perceptrón multicapa (MLP) y LightGBM. Analizando los resultados con diferentes combinaciones de datos y la aplicación de los distintos modelos, se concluye que los mejores resultados en términos de RMSE se obtienen con la inclusión de los datos del mapa de conocimiento situacional y la aplicación de LightGBM.

Por otra parte, [38] presenta dos modelos de predicción de demoras empleando redes neuronales de convolución profunda basadas en la fusión de datos meteorológicos: DCNN (*Dual-channel Convolutional Neural Network*) que se refiere a la estructura de la red ResNet, y SE-DenseNet (*Squeeze and Excitation-Densely Connected Convolutional Network*) que combina las ventajas de DenseNet y SENet. En primer lugar, los datos de vuelo y los

datos meteorológicos se fusionan en el modelo. A continuación, se utilizan los modelos DCNN y SE-DenseNet para extraer automáticamente características basadas en el conjunto de datos fusionados. Por último, se adopta el clasificador Softmax para predecir el nivel de retraso de los vuelos. Para el modelo DCNN propuesto, se diseñan tanto el canal recto como el canal de convolución con el fin de garantizar la transmisión sin pérdidas de la matriz de características y mejorar la permeabilidad de la red profunda. Para el modelo SE-DenseNet, se añade un módulo SE después de la capa de convolución de cada bloque DenseNet, que no sólo puede mejorar la transmisión de la información profunda, sino también lograr la recalibración de características en el proceso de extracción de características. Los resultados indican que, tras considerar las características de la información meteorológica, la precisión del modelo puede mejorarse en un 1% en comparación con el hecho de considerar únicamente la información de vuelo. Además, tanto DCNN como SE-DenseNet pueden mejorar eficazmente la precisión de la predicción, alcanzando el 92,10% y el 93,19%, respectivamente.

En la publicación de [39], se construyen dos modelos de predicción de las decisiones de aplicación del programa de demoras en tierra (GDP – *Ground Delay Program*), que permiten comprender cómo y por qué se toman esas decisiones. En particular, estos modelos son el de clonación conductual y el de aprendizaje por refuerzo inverso. Los datos empleados incluyen métricas de tráfico aéreo reales y programadas, así como condiciones meteorológicas observadas y previstas para los aeropuertos de Newark Liberty International y San Francisco International. Los resultados muestran que el modelo de clonación conductual Random Forest es sustancialmente mejor en la predicción de la aplicación del GDP para estos aeropuertos que el modelo de aprendizaje por refuerzo inverso. No obstante, se concluye que ambos modelos tienen dificultades para predecir la inicialización y la cancelación del GDP. En concreto, se concluye que las características de ambos tipos de modelos sugieren que las decisiones de implantación del GDP son más tácticas que estratégicas, es decir, se toman principalmente en función de las condiciones actuales o de las que se prevén únicamente para las próximas 2 horas.

También en relación con el GDP en los aeropuertos de Newark Liberty International y San Francisco International, en [40] se presentan dos modelos de aprendizaje supervisado,

---

regresión logística y árbol de decisión, para predecir la ocurrencia del mismo en base a las condiciones meteorológicas y a la demanda de tráfico programada. El modelo de regresión logística estima la probabilidad de que se active el GDP durante una determinada hora; mientras que el modelo de árbol de decisión clasifica si es probable o no que el GDP se active durante una determinada hora basándose en las variables de entrada. Los resultados indican que ambos modelos funcionan significativamente mejor que una predicción puramente aleatoria de la ocurrencia del GDP en los dos aeropuertos. El grado de impacto de las distintas variables de entrada en la probabilidad de activación de GDP varía entre los dos aeropuertos: en Newark el factor dominante es la meteorología convectiva en ruta, mientras que para San Francisco los principales impulsores del GDP son la escasa visibilidad y el bajo techo de nubes.

Igualmente relacionado con el GDP, en [41] se propone un método para predecir las demoras de vuelos individuales dentro de la red de tráfico aéreo completa. En concreto, se construye un modelo basado en agentes y varios parámetros cruciales que varían en el tiempo y afectan a los estados de los agentes, obtenidos mediante métodos de minería de datos. Se propone un modelo de probabilidad condicional para determinar si el motivo de la demora es el GDP o razones relacionadas con la compañía aérea. Adicionalmente, se incluyen dos modelos de regresión Random Forest para estimar el tiempo de rotación y el tiempo transcurrido desde la puerta de salida a la puerta de llegada de los vuelos. Los modelos se entrenan y ajustan utilizando los datos de vuelos de 2017 en Estados Unidos. El rendimiento del modelo de predicción de retrasos se prueba durante treinta días con tres tipos de niveles de retraso (bajo, medio y alto), seleccionados aleatoriamente en 2018. Los resultados experimentales muestran que el error absoluto medio en los días de prueba es de 6,8 minutos, y la precisión de la clasificación con un umbral de 15 minutos para un horizonte de previsión de dos horas es del 89,5%. Además, el modelo es robusto al aumento de los intervalos de predicción, y es capaz de predecir los retrasos de los vuelos en una red de tráfico aéreo a gran escala.

Por su parte, en [42] se propone un modelo de predicción de demoras en llegadas para el aeropuerto de Hartsfield – Jackson, Atlanta, aplicando una metodología de descubrimiento de conocimiento en bases de datos (KDD – *Knowledge Discovery in Databases*) y varias

técnicas de minería de datos. Para entrenar el modelo se emplean los datos históricos de vuelo y meteorológicos, información de la aeronave y propagación de las demoras; y para evitar el problema de los datos no balanceados se aplican varias técnicas de muestreo. A partir de ahí, se desarrollan tres modelos de predicción de demoras: Random Forest, árboles de decisión y perceptrón multicapa. Los resultados concluyen que el mejor modelo es el de perceptrón multicapa con un 85% de precisión.

Finalmente, en cuanto a predicción de demoras, los autores de [43] implementan un modelo de aprendizaje automático supervisado para predecir el intervalo de desviación de la hora de los nuevos vuelos en los aeropuertos de Vilnius (VNO), Kaunas (KUN), y Palanga (PLQ). Para ello hacen uso de siete algoritmos diferentes: redes neuronales probabilísticas, perceptrón multicapa, árboles de decisión, Random Forest, *ensemble* de árboles, árboles potenciados por gradientes y máquinas de vectores de soporte. Asimismo, emplean la búsqueda de cuadrícula para encontrar los parámetros que den la mayor precisión a cada algoritmo. Mediante *web scraping* obtienen los datos de vuelo y meteorológicos de las páginas oficiales, y dividen todos los conjuntos de datos en salidas y llegadas para hacer un análisis por separado de ambos segmentos. Los conjuntos de datos son balanceados mediante la técnica SMOTE. Como conclusión, obtienen que la mayor precisión se obtiene utilizando los clasificadores de modelo de árbol, y que el mejor algoritmo de este tipo para predecir es el de árboles potenciados por gradientes.

La Tabla 3.14 presenta un resumen de lo expuesto en los párrafos anteriores.

Tabla 3.14: Resumen trabajos de “predicción de demoras” (Fuente: *elaboración propia*)

ID	Descripción	Método
1	Predicción de retrasos desde la perspectiva de red de aeropuertos	<i>Deep learning</i> basado en LSTM profunda embebida en gráficos (DGLSTM)
2	Predicción de retrasos agregados en salidas	Regresión lineal múltiple, máquina de vectores de soporte, árboles extremadamente aleatorios y GBM ligero

ID	Descripción	Método
3	Predicción de demoras en llegadas	Nueva estructura ANN, denominada DMP-ANN ( <i>Defect of Modules Prediction – ANN</i> )
4	Predicción de demoras de los vuelos basándose en datos de IoT en vez de datos históricos	Regresión múltiple logística (para clasificar el vuelo en retrasado o en hora) y regresión múltiple lineal (para predecir el tiempo de demora en minutos)
5	Predicción de demoras de los vuelos basándose en datos de IoT en vez de datos históricos	Regresión múltiple logística (para clasificar el vuelo en retrasado o en hora) y regresión múltiple lineal (para predecir el tiempo de demora en minutos)
6	Predicción de retrasos en los vuelos	Random Forest
7	Predicción de demoras en llegadas	SVM, <i>Gradient Boosting</i> , Random Forest y AdaBoost
8	Predicción de retrasos en los vuelos	<i>Deep Learning</i> (DL) con el algoritmo de Levenberg-Marquart (LM)
9	Predicción de demoras en salidas utilizando el mapa de conocimiento situacional del aeropuerto	Regresor lineal (LR), regresor de vectores de soporte (SVR), perceptrón multicapa (MLP) y LightGBM
10	Predicción de retrasos en los vuelos	Redes neuronales de convolución profunda: DCNN y SE-DenseNet
11	Predicción de aplicación del programa de demoras en tierra (GDP)	Clonación conductual y aprendizaje por refuerzo inverso
12	Predicción de aplicación del programa de demoras en tierra (GDP)	Regresión logística y árbol de decisión
13	Predicción de aplicación del programa de demoras en tierra (GDP)	Modelo basado en agentes y Random Forest
14	Predicción de demoras en llegadas	Random Forest, árboles de decisión y perceptrón multicapa

---

ID	Descripción	Método
15	Predicción de retrasos en los vuelos	Redes neuronales probabilísticas, perceptrón multicapa, árboles de decisión, Random Forest, <i>ensemble</i> de árboles, árboles potenciados por gradientes y máquinas de vectores de soporte

En cuanto al siguiente tema con mayor número de trabajos obtenido, se tiene que es “meteorología”.

Por una parte, en [44] se propone la aplicación de algoritmos de clasificación y agrupación de minería de datos a vastos archivos de datos meteorológicos y de tráfico aéreo para identificar patrones y decisiones pasadas que puedan, en última instancia, ayudar a la toma de decisiones en el día a día de las operaciones de los aeropuertos. Específicamente, se aplica a Chicago O’ Hare International Airport y Newark Liberty International Airport. Los resultados obtenidos demuestran que es posible utilizar los archivos históricos de meteorología y tráfico aéreo para orientar sobre los tipos de iniciativas de gestión del tráfico que deben aplicarse en respuesta a las condiciones meteorológicas y de tráfico que afectan a un aeropuerto.

Además, en [45] se emplean dos métodos de aprendizaje automático para generar modelos que mapean las condiciones operativas del aeropuerto y la información meteorológica. Estos métodos son: aprendizaje por conjuntos con el modelo BDT (*Bagging Decision Tree*) para clasificar la diferencia entre la duración inicial planificada y la duración final planificada del GDP; y modelo de regresión de redes neuronales para predecir la duración del GDP. Los resultados concluyen que los modelos propuestos ofrecen mejores resultados que el método aplicado actualmente para la activación del GDP.

Uno de los mismos autores que el trabajo anterior, Y. Wang, publica también en [46] un método para evaluar el impacto de la meteorología en la selección de la configuración de pista operativa y la capacidad de los aeropuertos basado en aprendizaje por conjuntos con el modelo BDT. El modelo se desarrolla y valida utilizando los datos del ASPM de la FAA y las previsiones meteorológicas para los aeropuertos de Newark Liberty International, San

---

Francisco International, Chicago O' Hare International y Atlanta International. En el trabajo también desarrolla un modelo basado en SVM con el fin de comparar el rendimiento de ambos métodos. Los resultados experimentales concluyen que el BDT supera en precisión al SVM, y que el modelo propuesto de aprendizaje por conjuntos puede emplearse como un modelo de apoyo a la toma de decisiones de gestión del tráfico aéreo para cumplir con la capacidad aeroportuaria a fin de reducir los costes y aumentar la seguridad.

El resto de los trabajos encontrados relacionados con la meteorología se centran en la predicción de eventos de baja visibilidad y presencia nieblas. Así, por ejemplo, en [47] se incluye un caso de estudio en el aeropuerto de Wamena, Indonesia, en el que se emplean seis algoritmos diferentes para predecir los eventos con presencia de nieblas: Random Forest, aprendizaje profundo, máquinas potenciadas por gradientes, modelo lineal generalizado, árboles extremadamente aleatorios y conjuntos apilados. Para ello se emplea un conjunto de datos sinópticos horarios desde enero de 2015 hasta mayo de 2018, de tal manera que se predicen eventos de niebla/ no niebla partiendo de variables de entrada como temperatura, punto de rocío, humedad relativa, cobertura de nubes, dirección e intensidad del viento, visibilidad y tiempo de las últimas 6 horas anteriores. Los resultados obtenidos muestran que el mejor modelo se obtiene con conjuntos apilados, con una precisión superior al 90% para la previsión de niebla para las próximas 1 – 3 horas.

En [48] se aborda la predicción de eventos de baja visibilidad en los aeropuertos en términos de alcance visual en pista (RVR – *Runway Visual Range*) haciendo uso de algoritmos de redes neuronales (perceptrón multicapa y máquinas de aprendizaje extremo), regresión de vectores de soporte y procesos gaussianos. Para ello se emplean datos reales de variables atmosféricas obtenidos a partir de mediciones locales en el Aeropuerto de Valladolid y en una torre meteorológica cercana. Los resultados obtenidos confirman que los algoritmos propuestos son capaces de predecir correctamente los eventos de baja visibilidad en el aeropuerto, siendo los procesos gaussianos los que ofrecen mejores resultados con una tasa de clasificación correcta superior al 98% cuando el RVR es superior a 1000 metros, y alrededor del 80% por debajo de este umbral. No obstante, el rendimiento de todos los algoritmos se ve negativamente afectado en condiciones extremas de baja visibilidad (RVR inferior a 500 metros). Asimismo, se concluye que el



modelo requiere el uso de una torre meteorológica equipada en las inmediaciones, de la cual no disponen la mayoría de los aeropuertos, por lo que su aplicación real podría estar limitada.

Más adelante, en 2020, los mismos autores junto a otros, publican en [49] un estudio que continúa al anterior para el análisis de persistencia y predicción de los eventos de baja visibilidad en el mismo Aeropuerto de Valladolid. En este caso, plantean diferentes aproximaciones para el análisis de persistencia: para el largo plazo aplican un análisis de fluctuaciones sin tendencia (DFA – *Detrended Fluctuation Analysis*) para estimar las similitudes de las series temporales de RVR a gran escala; mientras que el corto plazo se evalúa mediante un análisis de cadenas de Markov de las series temporales binarias asociadas a los sucesos de baja visibilidad. Y también para la predicción de eventos proponen distintos enfoques: unos derivados del análisis de cadenas de Markov, y otros aplicando técnicas de aprendizaje automático basadas en máquinas de vectores de soporte y máquinas de aprendizaje extremo. Como conclusión se determina que se pueden obtener resultados muy precisos mediante el uso híbrido de operadores de persistencia y técnicas de *machine learning*.

Finalmente, en [50] se propone la predicción de visibilidad en aeropuertos basándose en redes neuronales LSTM, para lo que se emplean datos obtenidos de la administración meteorológica de China para la estación de Pekín desde abril de 2016 hasta diciembre de 2017, con un total de 15.143 datos, de los cuales los primeros 10.000 se utilizan para entrenar el modelo y el resto para la verificación del mismo. Con este fin, se propone una optimización del modelo LSTM para hacerlo más adecuado a las aplicaciones prácticas mediante el ajuste de la función de pérdida y la estructura de la red. Los resultados obtenidos muestran que el modelo LSTM propuesto proporciona un mejor rendimiento en cuanto a la tendencia general y la precisión que los planteados en estudios anteriores basados en ajuste polinómico simple y MLP.

Como resumen de los párrafos anteriores, se incluye en la Tabla 3.15 un esquema de lo descrito.

Tabla 3.15: Resumen trabajos de “meteorología” (Fuente: elaboración propia)

ID	Descripción	Método
16	Identificación de patrones en las condiciones meteorológicas que afectan a las operaciones	Árbol de decisión, regresión logística y árbol logístico
17	Mapeo de las condiciones operativas del aeropuerto y la información meteorológica	<i>Bagging Decision Tree</i> y regresión de redes neuronales
18	Evaluación del impacto de la meteorología en la selección de la configuración de pista operativa y la capacidad de los aeropuertos	<i>Bagging Decision Tree</i> y SVM
19	Predicción de eventos de baja visibilidad y presencia nieblas	Random Forest, <i>deep learning</i> , máquinas potenciadas por gradientes, modelo lineal generalizado, árboles extremadamente aleatorios y conjuntos apilados
20	Predicción de eventos de baja visibilidad	Redes neuronales (perceptrón multicapa y máquinas de aprendizaje extremo), regresión de vectores de soporte y procesos gaussianos
21	Predicción de eventos de baja visibilidad	Cadenas de Markov, máquinas de vectores de soporte y máquinas de aprendizaje extremo
22	Predicción de eventos de baja visibilidad	LSTM

Otro de los campos con mayor estudio encontrado, “rodaje de las aeronaves”, incluye principalmente trabajos enfocados en la predicción del tiempo de rodaje, como por ejemplo, [51], que aplica técnicas de aprendizaje automático para la predicción del tiempo de rodaje en salidas en el aeropuerto de Shanghái mediante un amplio sistema de predictores basado en una topología de red macroscópica desde una visión agregada. Estos

---

predictores se pueden dividir en cuatro categorías: índices de flujo instantáneo en superficie (SIFIs – *Surface Instantaneous Flow Indices*), índices de flujo acumulado en superficie (SCFIs – *Surface Cumulative Flow Indices*), índices de longitud de cola de espera de aeronaves (AQLIs – *Aircraft Queue Length Indices*), e índices de demanda de recursos de franjas horarias (SRDIs – *Slot Resource Demand Indices*). Además, se aplican tres métodos de ML: regresión lineal, máquinas de vectores de soporte y Random Forest; los cuales se formulan utilizando muestras de entrenamiento de un día y de un mes. Los resultados computacionales muestran que el modelo de entrenamiento de RF de un mes supera significativamente a los demás modelos en términos de precisión de la predicción.

Por su parte, en [52] se propone la predicción del tiempo de rodaje en salidas con la aplicación de modelos mejorados de SVR basados en la optimización mediante el algoritmo de inteligencia de enjambre, incluyendo optimización por enjambre de partículas y algoritmo luciérnaga. Como caso de estudio incluye el análisis con datos históricos reales del aeropuerto de Pekín. Además, con fin de realizar una comparativa, se aplican también modelos de regresión lineal generalizada, modelo de regresión Softmax y redes neuronales artificiales. Las medidas de rendimiento muestran que los dos enfoques SVR propuestos, especialmente el método SVR basado en la optimización del algoritmo luciérnaga, no solo presentan las mejores medidas de modelización y tasa de precisión, sino que también pueden lograr un mejor rendimiento predictivo cuando se trata de estados anormales de tiempo de rodaje.

En esta misma temática, en [53] se propone emplear técnicas de ML para evaluar la precisión obtenida con el uso de una herramienta de simulación de eventos discretos en tiempo rápido, LINOS (*Linear Optimized Sequencing*), en la predicción de los tiempos de rodaje y proporcionar estimaciones para la programación de pistas operativas en Charlotte Douglas International Airport. Para ello, se emplean regresión lineal, máquinas de vectores de soporte, K-vecino más próximo y Random Forest. Los resultados obtenidos muestran que K-vecino más próximo y Random Forest proporcionan los mejores resultados en lo que respecta a RMSE y MAE, y que LINOS ofrece mejor rendimiento que la regresión lineal y una precisión similar a SVM.

Un año más tarde, en 2016, dos de los autores del trabajo anterior, H. Lee y W. Malik, aplican en [54] diferentes técnicas para la predicción del tiempo de rodaje en salidas en el mismo aeropuerto de Charlotte Douglas International, entre las que se encuentran: regresión lineal, máquinas de vectores de soporte, K-vecino más próximo, Random Forest y redes neuronales. Para ello, utilizan datos de vuelo reales obtenidos del sistema del aeropuerto. Los resultados de predicción obtenidos muestran que la regresión lineal y el Random Forest proporcionan los mejores rendimientos, pero su precisión en la predicción del tiempo de rodaje para vuelos individuales no es satisfactoria, ya que solo el 65-74% de las salidas se encuentran dentro de los +/- 5 minutos de error en el tiempo de rodaje. Se concluye que este límite en el rendimiento se debe a la complejidad operativa y las incertidumbres del aeropuerto de Charlotte.

Por su parte, [55] incluye un modelo de regulación inteligente de la velocidad de rodaje de las aeronaves basado en el algoritmo DQN (*Deep Q Network*). Este modelo se entrena con datos de conflictos en el esquema de rodadura del aeropuerto obtenidos de simulaciones con el software AirTop. Los resultados experimentales muestran que el DQN es capaz de adoptar diferentes estrategias para la aeronave cuando se enfrenta a un conflicto. Bajo la premisa de garantizar en todo caso el margen de seguridad establecido, se comprueba que el modelo puede reducir el tiempo de espera en rodajes aproximadamente un 60%. Comparado con los datos de simulación del AirTop, el tiempo empleado tanto en la resolución de conflictos como en la ocupación de la longitud de la calle de rodaje disminuyen significativamente.

Por último, [56] pone a prueba los diferentes enfoques utilizados para predecir los tiempos de rodaje en los aeropuertos, en concreto, los siguientes: regresión lineal múltiple, regresión lineal del mínimo cuadrado de la mediana, regresión de vectores de soporte, árboles de decisión, sistemas basados en reglas difusas tipo Mamdani y sistemas basados en reglas difusas tipo TSK. Para ello, hace uso de datos históricos reales de dos aeropuertos *hub* europeos. Este trabajo concluye mediante comparaciones detalladas que un sistema basado en reglas difusas tipo TSK supera a los otros enfoques en términos de precisión de la predicción.

La Tabla 3.16 resume los datos más relevantes de los estudios de este campo.

Tabla 3.16: Resumen trabajos de “rodaje de las aeronaves” (Fuente: elaboración propia)

ID	Descripción	Método
23	Predicción del tiempo de rodaje en salidas	Regresión lineal, SVM y Random Forest
24	Predicción del tiempo de rodaje en salidas	Regresión lineal generalizada, modelo de regresión Softmax y redes neuronales artificiales
25	Evaluación de la precisión de una herramienta de simulación en la predicción de tiempos de rodaje	Regresión lineal, SVM, K-NN y Random Forest
26	Predicción del tiempo de rodaje en salidas	Regresión lineal, SVM, K-NN, Random Forest y redes neuronales
27	Regulación de la velocidad de rodaje	<i>Deep Q Network</i>
28	Predicción del tiempo de rodaje	Regresión lineal múltiple, regresión lineal del mínimo cuadrado de la mediana, SVR, árboles de decisión, sistemas basados en reglas difusas tipo Mamdani y sistemas basados en reglas difusas tipo TSK

El siguiente campo con mayor número de publicaciones es “edificio terminal”. Dentro de este, los trabajos se centran en temas muy variados. Por ejemplo, en [57] se propone el uso de técnicas de *machine learning* para predecir las necesidades de limpieza de los aseos del edificio mediante datos en tiempo de real de satisfacción de los usuarios obtenidos de las máquinas de *feedback* “*happy or not*”. Para ello se desarrollan dos modelos: uno que considera cada baño individual dentro de un mismo núcleo de aseos, y otro que considera el conjunto de baños en un núcleo de aseos. Para el primer modelo, se emplean los algoritmos de Random Forest, máquina de vectores de soporte, AdaBoost y K-vecino más próximo; mientras que para el segundo se hace uso de Random Forest, AdaBoost y K-vecino más próximo únicamente. El conjunto de datos utilizado en el estudio contiene diez

semanas de datos de *feedback* de los usuarios en tiempo real en el período que va desde el lunes 11 de marzo hasta el domingo 19 de mayo de 2019. Los resultados obtenidos muestran que el modelo de mejor rendimiento resulta ser el algoritmo K-vecino más próximo; y, aunque los modelos de predicción combinados superan las líneas de base, la precisión sigue siendo demasiado baja para la aplicación práctica.

Siguiendo en la temática de la experiencia del pasajero, en [58] se presentan las distintas soluciones que el operador del aeropuerto, Fraport, está poniendo en marcha en el Aeropuerto de Frankfurt en lo que a análisis predictivo se refiere, tanto para mejorar la experiencia del pasajero como para fortalecer sus distintas unidades de negocio. Entre las soluciones planteadas se incluyen:

- **Análisis del flujo de pasajeros:** se desarrolla una estrategia integral para la gestión avanzada del flujo de pasajeros con el fin de reducir los tiempos de espera en los distintos procesos. Para ello, se mide el flujo de pasajeros a través de la terminal en múltiples puntos del proceso, recogiendo los datos mediante diversas tecnologías de sensores incluyendo soluciones basadas en vídeo y escáneres de códigos de barras de tarjetas de embarque. A partir de esto, se calculan las previsiones de la demanda futura en diferentes puntos del proceso, utilizando modelos para predecir el número de pasajeros en cada vuelo individual, así como las relaciones de transferencia entre todos los vuelos. Finalmente, se aplican tecnologías avanzadas como la simulación del comportamiento basado en agentes para calcular el movimiento esperado de los pasajeros a través de la terminal, lo que permite obtener conclusiones sobre la futura demanda en los distintos puntos del proceso.
- **Reconocimiento de tendencias en *retail*:** en base a un análisis estadístico de los datos históricos, se predicen los ingresos de las tiendas de *retail* con un alto grado de confianza basándose únicamente en dos parámetros: el número de pasajeros y la hora.
- **Optimización del estacionamiento de las aeronaves:** se plantea un prototipo que permite simular el efecto de los planes de estacionamiento de las aeronaves sobre los ingresos de *retail*, basándose en un modelo multivariable que considera el efecto de la ubicación de la puerta de embarque, aerolínea, destino, tiempo de

espera, tiempo de permanencia y otros parámetros en el movimiento de los pasajeros a través del terminal y, por tanto, en los ingresos comerciales del mismo.

- Mejora del cálculo de la hora estimada de puesta de calzos: se desarrolla un modelo de predicción de la hora estimada de puesta de calzos a partir de datos históricos de vigilancia de la aproximación final y de la fase de rodaje.

También relacionado con el flujo de pasajeros en el terminal, en [59] se plantea la predicción a nivel estratégico de los flujos de pasajeros en los controles de seguridad del Aeropuerto Charles De Gaulle utilizando técnicas de aprendizaje automático basadas en redes neuronales LSTM. El conjunto de datos empleado en el estudio comprende datos reales de pasajeros gestionados desde el 1 de febrero de 2017 hasta el 31 de marzo de 2019. Los datos de 2017 y 2018 se utilizan para la fase de entrenamiento, y los datos de 2019 para la fase de validación. Los resultados obtenidos son prometedores, ya que los modelos LSTM superan al actual en todos los puntos de control utilizando las métricas teóricas; y en tres de los ocho puntos de control, los modelos LSTM superan al actual utilizando todas las métricas consideradas.

Por su parte, en [60] los autores estudian la aplicación de tres arquitecturas de redes neuronales para predecir el comportamiento de los pasajeros a través del edificio terminal, desarrollando un caso de estudio real utilizando rastros anónimos de WiFi recogidos en el Aeropuerto de Bolonia para entrenar las redes. Las tres arquitecturas propuestas son: redes neuronales prealimentadas (FNN – *Feedforward Neural Networks*), redes LSTM, y una combinación de ambas. Los resultados muestran que el rendimiento de las FNN es muy bueno en la parte inicial de la secuencia de actividades de los pasajeros, empeorando en el periodo crítico (de 100 a 30 minutos antes de la salida del vuelo); mientras que la red LSTM proporciona predicciones mucho más fiables a corto plazo, y al aumentar el horizonte de predicción, el rendimiento pasa a ser comparable con el del modelo FNN.

Finalmente, Á. Rodríguez-Sanz et al. plantean en [61] un análisis de las pautas de comportamiento en las colas de los mostradores de facturación y de los controles de seguridad, para lo cual se simula el flujo de pasajeros en el Aeropuerto de Palma de Mallorca entre 2014 y 2016 empleando datos de vuelo reales con el fin de obtener la longitud de las colas. Tras realizar un análisis de los datos basado en métodos estadísticos

y de visualización, se implementan algoritmos de Random Forest para desarrollar diferentes modelos de predicción para cada parámetro. Las técnicas de análisis de las características entre las variables dependientes y las variables objetivo (longitud de colas) determinan cuáles son los elementos fundamentales para explicar el comportamiento de las colas y para predecir la longitud de las mismas. Los resultados muestran que los parámetros que más influyen son el promedio de pasajeros programados, la hora, la estacionalidad, el día de la semana y si se trata de un evento especial en el calendario o no. Asimismo, el comportamiento de las colas en los mostradores de facturación es más difícil de capturar que el de los controles de seguridad, debido a su naturaleza estocástica y a su variabilidad. No obstante, en ambos casos la predicción proporciona una precisión razonable.

Un resumen de lo expuesto en los párrafos anteriores para este campo de estudio queda recogido en la Tabla 3.17.

Tabla 3.17: Resumen trabajos de “edificio terminal” (Fuente: elaboración propia)

ID	Descripción	Método
29	Predicción de las necesidades de limpieza de los aseos a partir de datos en tiempo de real de satisfacción de los usuarios	Random Forest, SVM, AdaBoost y K-NN
30	Análisis predictivo para mejorar la experiencia del pasajero y para fortalecer las distintas unidades de negocio del aeropuerto	Simulación del comportamiento basado en agentes
31	Predicción a nivel estratégico de los flujos de pasajeros en los controles de seguridad	LSTM
32	Predicción del comportamiento de los pasajeros a través del edificio terminal	Redes neuronales prealimentadas (FNN), redes LSTM, y una combinación de ambas



---

ID	Descripción	Método
33	Análisis de las pautas de comportamiento en las colas de los mostradores de facturación y de los controles de seguridad	Random Forest

En el siguiente escalón en cuanto a número de estudios obtenidos, se tiene el siguiente campo, “pasajeros”. Los temas en los que se centran los trabajos de esta materia son variados. Por ejemplo, en [62] se propone un nuevo modelo de evaluación de riesgos para el control de seguridad de pasajeros basado, en primer lugar, en obtener datos en tiempo real del estado de los pasajeros mediante cámaras de alta definición, detectores de seguridad de ondas milimétricas, etc.; para, a continuación, junto con datos históricos, aplicar técnicas de ML (tecnologías de análisis de emociones y reconocimiento de expresiones faciales) con el fin de analizar el nivel de riesgo de los pasajeros e integrar un proceso de jerarquía analítica (AHP – *Analytic Hierarchy Process*) mejorado con la teoría de decisión de grupo (GD – *Group Decision*). Así, según el nivel de riesgo, los pasajeros pueden clasificarse en objetivos conocidos, ordinarios y peligrosos. El tratamiento diferenciado de los diferentes objetivos podría ahorrar significativamente el tiempo del control de seguridad y mejorar la experiencia de los pasajeros en el mismo.

Por otro lado, [63] realiza un análisis de las causas que generan resistencia a los pasajeros a hacer uso de las *e-gates* biométricas en los aeropuertos, así como los riesgos y beneficios percibidos por los mismos, las características de los usuarios y los mecanismos de propagación que provocan dicha resistencia. Para ello, con datos de encuestas a 339 usuarios, se genera un modelo basado en máquinas de vectores de soporte para clasificar correctamente las causas de la resistencia y se aplica un análisis comparativo cualitativo de conjuntos crispados (csQCA – *crisp set Qualitative Comparative Analysis*) con el fin de comprender las complejas causas subyacentes. Los resultados obtenidos muestran que la percepción de presencia de riesgos y ausencia de beneficios son las principales causas de la reticencia de los pasajeros a utilizar las *e-gates*. Además, se detecta la información *online* como mecanismo de propagación con impacto en la resistencia a esta tecnología.

Los dos últimos estudios encontrados de este campo se centran en la previsión de demanda de pasajeros. Por una parte, [64] propone un nuevo modelo de previsión de pasajeros basado en una red neuronal profunda (DNN), concretamente, el algoritmo LSTM. Este se aplica a los datos reales del Aeropuerto Internacional de Incheon, Corea del Sur, para demostrar su eficacia y viabilidad. En el estudio también se incluyen predicciones utilizando la metodología SARIMA (*Seasonal Auto-Regressive Integrated Moving Average*) con el fin de comparar ambos métodos. Los resultados experimentales muestran que tanto SARIMA como LSTM proporcionan una previsión precisa y fiable, y tienen una gran capacidad de predicción en el corto plazo; sin embargo, el modelo LSTM muestra un rendimiento superior, especialmente a partir del medio plazo.

Por otra parte, [65] presenta un modelo de predicción basado en un marco de aprendizaje en dos fases. En la fase uno varios predictores (SARIMA, CNN-LSTM y *Seasonal Difference-XGBoost*) tratan las diferentes características de las series temporales en paralelo. En la segunda fase se integran los resultados de predicción de tal manera que las previsiones obtenidas son además entrenadas mediante regresión lineal múltiple y se aplican otras dos estrategias de fusión: el método promedio y el método ponderado. Además, el trabajo compara los principales indicadores de error con los datos reales, y los resultados muestran que el modelo de aprendizaje de dos fases funciona mejor que los modelos de fusión actuales y posee un rendimiento estable.

La Tabla 3.18 resume los resultados explicados para este campo.

Tabla 3.18: Resumen trabajos de “pasajeros” (*Fuente: elaboración propia*)

ID	Descripción	Método
34	Evaluación de riesgos para el control de seguridad de pasajeros	DL. Análisis de emociones y reconocimiento de expresiones faciales
35	Análisis de las causas que generan resistencia a los pasajeros a hacer uso de las <i>e-gates</i> biométricas	SVM
36	Previsión de demanda de pasajeros	LSTM

---

ID	Descripción	Método
37	Previsión de demanda de pasajeros	CNN-LSTM, <i>Seasonal Difference</i> -XGBoost y regresión lineal múltiple

El campo que ocupa el siguiente peldaño en cuanto a número de publicaciones encontradas es “operaciones”.

En primer lugar, [66] presenta la aplicación de ML para el desarrollo de modelos precisos para la predicción del tiempo de ocupación de pista (ROT – *Runway Occupancy Time*). Basados en los regresores de refuerzo de gradiente, estos modelos utilizan la información de vuelo (como el tipo de avión, la compañía aérea y los datos de vuelo) y la información meteorológica para modelar el ROT. El método se desarrolla y evalúa sobre la base de datos de tres años de mediciones de campo recogidas en el aeropuerto de Zúrich. Los resultados muestran que el modelo es capaz de predecir el ROT usando únicamente datos disponibles 10 minutos antes del aterrizaje. Además, el estudio incluye una metodología para convertir las predicciones del ROT en indicadores de distancia que pueden ser utilizados en una herramienta de ATC con el fin de permitir una reducción en las separaciones basadas en el ROT, con lo que se podría incrementar la capacidad de pista en un 10%.

Por otra parte, en [67] se plantea la aplicación de un algoritmo genético para aprender la secuenciación de aeronaves aplicada por los controladores aéreos utilizando una sociedad de máquinas probabilísticas de estado finito (PFSMs – *Probabilistic Finite-State Machines*). Como datos de aplicación, se generan aleatoriamente 200 secuencias únicas de aterrizaje, cada secuencia compuesta por 20 aviones. Cada PFSM aprende un subespacio diferente, de tal manera que el problema de aprendizaje se descompone en un grupo de agentes que necesitan trabajar juntos para aprender el problema global. Se utilizan tres métricas de secuenciación diferentes para la función de aptitud en el algoritmo genético: Levenshtein, Hamming y distancias de posición. Los resultados muestran que el enfoque propuesto es capaz de aprender el mecanismo subyacente que genera las secuencias de aterrizaje aplicadas por los controladores conociendo únicamente información limitada.

Por último, [68] propone un método de conocimiento de la situación basado en redes de Petri (PN – *Petri Nets*) en combinación con redes neuronales artificiales, introduciendo la

teoría de colas y los parámetros perceptivos en las PN existentes. Se propone un algoritmo completo de conocimiento de la situación para realizar una predicción sostenida y precisa mediante la resolución de estimaciones puntuales de la situación macroscópica y microscópica en este modelo, lo que ayuda a abordar algunos retos a los que se enfrentan los aeropuertos de la aviación civil actual. Para evaluar el método y poder compararlo con el A-CDM se emplea un conjunto de datos de operaciones en 2016 del aeropuerto de Hefei Xinqiao, China. Los resultados obtenidos verifican que el modelo y su algoritmo de estimación de la situación tienen un mejor rendimiento que el A-CDM.

Un resumen de las principales características de estos trabajos se incluye en la Tabla 3.19.

Tabla 3.19: Resumen trabajos de “operaciones” (Fuente: elaboración propia)

ID	Descripción	Método
38	Predicción del tiempo de ocupación de pista	Regresores de refuerzo de gradiente
39	Aprendizaje de la secuenciación de aeronaves aplicada por los controladores aéreos	Algoritmo genético, utilizando una sociedad de máquinas probabilísticas de estado finito (PFSMs)
40	Conocimiento de la situación ( <i>situational awareness</i> )	Redes de Petri (PN) en combinación con redes neuronales artificiales

Con el mismo número de estudios se tiene el tema de “tratamiento de equipajes”. A este respecto, en [69] se describe un caso práctico de aplicación de técnicas de monitorización de procesos tanto exploratorias como predictivas para comprender y predecir los tiempos de finalización de un proceso de gestión de equipajes en un aeropuerto. Para ello, se combina un modelo de Random Forest y un modelo de LSTM en un nuevo modelo de predicción apilado. Random Forest se utiliza para proporcionar una predicción de referencia basada en información de características estáticas, mientras que LSTM se usa para actualizar secuencialmente la predicción inicial de referencia a medida que la instancia del proceso comienza a generar actividades. El método se aplica para predecir el tiempo de finalización de la transferencia de maletas en el aeropuerto de Bruselas, obteniendo

unos resultados que concluyen que se supera el rendimiento de otros enfoques empleados en la actualidad.

Por su parte, [70] propone aplicar una metodología de aprendizaje automático adaptativo al problema de reconocimiento de amenazas en el tratamiento de equipajes de un aeropuerto. Esta se compone de 3 fases: un algoritmo de segmentación de imágenes 3D multiescala de tomografía computarizada (CT – *Computerized Tomography*), un clasificador SVM para el reconocimiento del material del objeto, y una estrategia de adaptación a requisitos cambiantes. Para validar la eficacia del enfoque propuesto, se realizan una serie de experimentos con datos reales de imágenes de CT de equipaje. Los resultados obtenidos muestran una probabilidad de detección de en torno al 90% con una probabilidad de falsa alarma inferior al 20%.

Finalmente, G. Szwoch presenta en [71] un algoritmo para detectar equipajes abandonados a partir de imágenes de video. El algoritmo propuesto se basa en la detección de regiones de la imagen que contienen píxeles en primer plano que tienen valores estables en el tiempo y en la comprobación de su correspondencia con los objetos en movimiento detectados. Los objetos estacionarios son examinados por un clasificador para comprobar si representan un equipaje y, posteriormente, se comprueban las condiciones espacio temporales para decidir si los objetos detectados como equipaje están o no desatendidos (abandonados). En esta clasificación de equipaje o no y decisión de si está desatendido o no, se aplica Random Forest y SVM. Los resultados muestran que ambos clasificadores obtienen resultados idénticos en la fase de validación (94,75% de precisión).

La Tabla 3.20 recoge las principales características descritas para estos trabajos.

Tabla 3.20: Resumen trabajos de “tratamiento de equipajes” (*Fuente: elaboración propia*)

ID	Descripción	Método
41	Predicción de los tiempos de finalización de un proceso de gestión de equipajes en un aeropuerto	Combinación de Random Forest y LSTM en un nuevo modelo de predicción apilado
42	Reconocimiento de amenazas en el tratamiento de equipajes	SVM

---

ID	Descripción	Método
43	Detección de equipajes abandonados a partir de imágenes de video	Random Forest y SVM

El siguiente campo de estudio es el “ruido”, con dos trabajos obtenidos.

Por una parte, en [72] se diseña un algoritmo de regresión de vectores de soporte difusos basados en factores de exclusión locales (LOF-FSVR, *Local Outlier Factor based Fuzzy Support Vector Regression*) para predecir el ruido aeroportuario. La base de este algoritmo es SVM, pero se introduce un método de cálculo de pertenencia difusa de cada muestra en base a un factor local de valores atípicos. Además, con el fin de mejorar la fiabilidad y la precisión de las predicciones, se construye un modelo *ensemble* de LOF-FSVR. El estudio emplea datos de mediciones de ruido en tiempo real recogidos de los sistemas de control del ruido a gran escala que se han desarrollado para supervisar el ruido de los aeropuertos. En él se concluye que SVM proporciona buenas capacidades de predicción, pero como los datos de ruido recogidos incluyen valores atípicos, el modelo diseñado de LOF-FSVR resuelve ese problema y mejora las predicciones.

Por otra parte, [73] plantea la aplicación de K-NN para la clasificación de eventos sonoros recogidos por una herramienta de detección de ruido en el aterrizaje. Se tienen dos detectores de ruido, uno de aterrizaje y otro de empuje de reversa. Tras detectar los dos eventos sonoros consecutivos, el sistema los clasifica de forma independiente con el fin de reducir las tasas de identificación de falsos positivos. El proceso de reconocimiento comienza con un análisis de componentes principales (PCA), utilizado para descorrelacionar los datos. Después, se utiliza K-NN para los eventos sonoros de empuje de reversa y un clasificador Parzen para los de aterrizaje. De esta manera, la identificación de la activación de empuje de reversa es positiva si el primer evento se clasifica como aterrizaje y el segundo se clasifica como ruido de empuje de reversa. El estudio se realiza sobre mediciones tomadas en el Aeropuerto Adolfo Suárez Madrid – Barajas. Los resultados muestran unos índices de clasificación bastante buenos, por encima del 90%.

En la Tabla 3.21 se incluye un resumen de lo expuesto para este campo de estudio.

Tabla 3.21: Resumen trabajos de “ruido” (Fuente: elaboración propia)

ID	Descripción	Método
44	Predicción del ruido aeroportuario	SVM modificado: regresión de vectores de soporte difusos
45	Clasificación de eventos sonoros	PCA y K-NN

También con dos publicaciones se tiene el siguiente campo “general”.

La primera, [74], incluye una revisión de las aplicaciones de *machine learning* en los *smart airports*. Según ella, las cuatro mayores áreas de aplicación del ML en los aeropuertos están relacionadas con: los retrasos de los vuelos, la gestión medioambiental, la gestión de las operaciones y la seguridad.

Por su parte, [75] plantea un modelo para analizar la evolución de los aeropuertos de unas categorías a otras a lo largo de un periodo de 29 años. Además de ofrecer una visión general de la evolución, el documento incluye un esquema de evaluación basada en datos que puede ayudar al sector público y a la industria a tomar decisiones estratégicas con conocimiento de causa en materia de infraestructuras de transporte aéreo. Para ello, se utiliza un registro de esos 29 años (1979-2007) de datos de vuelos globales obtenidos de OAG (*Official Airline Guide*) y se calculan las probabilidades de transición condicionales, mostrando los resultados como un gráfico de evolución similar a una cadena de Markov en tiempo discreto. En base a este concepto analítico, se analiza el significado de los aeropuertos desde una perspectiva de red y se establece una nueva taxonomía aeroportuaria. Teniendo en cuenta seis parámetros de la teoría de redes, se identifican doce categorías de aeropuertos diferentes que se extienden siguiendo patrones evolutivos claros. Los resultados muestran que los aeropuertos con una función similar en la red pasan por etapas de desarrollo similares.

La Tabla 3.22 resume las características más relevantes de estos dos trabajos.

Tabla 3.22: Resumen trabajos de “general” (Fuente: elaboración propia)

ID	Descripción	Método
46	<i>Smart airports</i>	-
47	Análisis de la evolución de los aeropuertos de unas categorías a otras a lo largo de un periodo de 29 años	Cadena de Markov

Finalmente, el último campo de estudio es el “mantenimiento del campo de vuelos”, con una publicación. El trabajo incluido en [76] propone el desarrollo de modelos calibrados para predecir la degradación de la fricción en las pistas de vuelo en relación con las cargas de tráfico. El método propuesto permite identificar la posición y la extensión de la zona de máxima degradación de fricción y reconocer su valor de "etiqueta" como valor mínimo de referencia para toda la pista. Asimismo, permite analizar la evolución del fenómeno en diferentes momentos de la vida útil del pavimento. El análisis demuestra que la extracción de datos mediante un algoritmo de aprendizaje CART (*Classification And Regression Tree*) arroja resultados especialmente coherentes y precisos. Además, el algoritmo hace que los datos sean autónomos y lo suficientemente elásticos como para evaluar las distintas zonas de degradación desde el punto de vista tanto de la extensión como de la intensidad.

Esto se resume en la Tabla 3.23.

Tabla 3.23: Resumen trabajos de “mantenimiento de campo de vuelos” (Fuente: elaboración propia)

ID	Descripción	Método
48	Predicción de la degradación de la fricción en las pistas de vuelo	CART

#### 3.1.4 Conclusiones

Como se ha comentado, con la revisión sistemática de la literatura expuesta arriba se pretende dar respuesta a las preguntas de investigación en las que se basa este trabajo:



- PI-1: ¿Qué aplicaciones de *machine learning* se están usando en la gestión operativa de los aeropuertos?
- PI-2: ¿Cómo puede el *machine learning* apoyar la toma de decisiones para mejorar la eficiencia operativa y la gestión de recursos en los aeropuertos?

Una vez llevada a cabo y analizados los resultados, se puede concluir que la SLR permite responder satisfactoriamente a estas preguntas, lo cual confirma la validez del estudio realizado.

Por una parte, con respecto a la pregunta más genérica, PI-2, se ha obtenido que el ML tiene aplicaciones en numerosos campos de la gestión aeroportuaria, permitiendo al gestor poder anticiparse a eventos que afectarán a la operatividad, con el fin de activar los medios de manera proporcionada y eficiente, con el consiguiente ahorro en términos energéticos y económicos, y la mejora de la satisfacción de los pasajeros y *stakeholders*.

De estos campos, la “predicción de demoras” es el más estudiado. Esto puede tener su justificación en que los retrasos representan un problema importante en la industria del transporte aéreo que genera cuantiosas pérdidas económicas a todas las partes implicadas (gestores, compañías aéreas, compañías de *handling*, etc.). Si a esto se une el fuerte crecimiento de este sector en los últimos años, es de esperar que se inviertan todos los esfuerzos posibles en buscar soluciones para reducir al nivel más bajo posible la ocurrencia de demoras y para mitigar los efectos de las mismas. En este contexto, el *machine learning* se abre paso como una herramienta muy útil a la hora de predecir la ocurrencia de retrasos, identificar los puntos en los que estos se generan y evaluar el efecto cascada de su acumulación en las distintas fases de los vuelos, tanto en el aire como en tierra. La mayoría de los modelos propuestos emplean como datos de partida los datos de vuelo y los datos meteorológicos, aplicando distintos algoritmos de predicción. Otros introducen mayor innovación proponiendo el uso de IoT para obtener los datos de entrada.

Muy relacionado con el anterior está el estudio de la “meteorología”. Su importancia radica tanto en el efecto negativo que puede provocar sobre la operatividad de los aeropuertos con las consecuentes demoras, como en la necesidad de asegurar en todo momento la seguridad operacional (*safety*), que representa uno de los pilares de la aviación. Las principales aplicaciones de ML en este campo ayudan en la identificación de patrones en

---

las condiciones meteorológicas que afectan a las operaciones, la evaluación del impacto de condiciones adversas que afectan a la capacidad aeroportuaria y la predicción de ocurrencia de eventos de baja visibilidad y reducción del RVR. Esto proporciona al gestor una información valiosa para anticiparse en la toma de decisiones y minimizar impactos, por ejemplo, con la activación del LVP (*Low Visibility Procedure*).

El tercer área con mayor estudio es el “rodaje de las aeronaves”. El fuerte interés en una gestión eficiente del rodaje tiene varias fuentes. Por un lado, está el requisito de establecer unos flujos de rodaje que garanticen la seguridad operacional (por ejemplo, con la gestión de los *hot spots*) y que minimicen las emisiones de contaminantes y el ruido; por otro, está la importancia de esta fase operativa desde el punto de vista del aeropuerto como sistema, ya que tiene afección sobre el resto de los componentes: secuenciación de pistas, asignación de puertas, etc.; y finalmente, en ocasiones, la fase de taxi es el origen de demoras importantes, especialmente en aeropuertos congestionados. En este contexto, el ML aporta contribuciones relevantes tanto en la predicción de los tiempos de rodaje como en la regulación eficiente de las velocidades de taxi.

Los siguientes campos por volumen de publicaciones son “edificio terminal” y “pasajeros”. Se han separado por una cuestión de enfoque, pero en la realidad estos campos están totalmente interrelacionados, ya que el edificio terminal tiene como principal propósito albergar las instalaciones adecuadas para proporcionar un servicio óptimo a los pasajeros desde su llegada al mismo hasta que toman un vuelo. Además, en el terminal se ubican las principales fuentes de ingresos no aeronáuticos (ingresos mayoritarios en gran parte de los aeropuertos comerciales del mundo), por lo que su gestión eficiente tiene un enorme impacto en las cuentas del gestor.

En estos campos, las aplicaciones de ML son muy variadas: desde la predicción de demanda y de presentación de flujos de pasajeros en los distintos procesos, o la predicción de necesidades de diferentes subsistemas tanto para mejorar la experiencia del pasajero como para fortalecer las distintas unidades de negocio del aeropuerto; hasta el análisis del comportamiento de los pasajeros a través del edificio y en las colas de los distintos procesos, o la evaluación de riesgos de seguridad mediante el análisis de emociones y

---

reconocimiento de expresiones faciales. Todas ellas mejoran la eficiencia operativa y la gestión de recursos.

Por otra parte, los departamentos de “operaciones” de los aeropuertos también pueden apoyarse en el ML para actividades como la predicción de los tiempos de ocupación de pista, la secuenciación de operaciones de aterrizaje y despegue o el desarrollo de herramientas de aprendizaje automático para el conocimiento situacional que sustituyan en parte o complementen al A-CDM. Esto incrementa el conocimiento sobre la capacidad disponible y/o requerida en cada momento, de tal manera que se pueden activar los medios correspondientes.

Otro de los campos de aplicación es el sistema de “tratamiento de equipajes”, cuya gestión queda definida por dos objetivos principales: garantizar la seguridad (*security*) y ofrecer una buena experiencia al pasajero. Para cada uno de ellos, el aprendizaje automático ofrece soluciones con buenos resultados: por una parte, mediante la predicción de los tiempos de proceso se puede ofrecer al pasajero un servicio que reduzca sus tiempos de espera, así como gestionar de manera eficiente los equipos que se activan en cada momento; por otra parte, mediante técnicas como el reconocimiento de amenazas en el tratamiento de equipajes o la detección de equipajes abandonados a partir de imágenes de video se incrementan los niveles de seguridad.

El siguiente área con mayor estudio es el “ruido”, lo cual va en línea con los esfuerzos que las distintas partes involucradas de la aviación están invirtiendo en reducir el impacto de los aeropuertos en el entorno. En esta revisión de la literatura se han encontrado dos aplicaciones de ML a este respecto: una es la predicción de niveles de ruido aeroportuario, y la otra es la clasificación de eventos sonoros para tratar de reducir el ruido en la aplicación de empuje de reversa durante el aterrizaje. Con ambas se obtiene mayor información sobre los impactos de ruido y, por tanto, permiten anticiparse en la toma de decisiones de aplicación de medidas mitigadoras.

Por último, como en muchos otros sectores, el ML proporciona soluciones muy eficaces para el mantenimiento predictivo. Una técnica innovadora es su aplicación en el “mantenimiento del campo de vuelos”, a través de la predicción de la degradación de la fricción de los pavimentos, que permite predecir las necesidades de mantenimiento/

actuación sobre los mismos en el tiempo y, por tanto, facilita la programación de estas tareas en momentos o tramos en los que el impacto en las operaciones sea el mínimo.

En cuanto a la PI-1, además de todo lo incluido como respuesta a la anterior, también se han encontrado respuestas con ejemplos de aplicación en varios aeropuertos. Por ejemplo, Fraport está aplicando ML en el Aeropuerto de Frankfurt en varios aspectos: gestión del flujo de pasajeros para reducir los tiempos de espera en los distintos procesos dentro del edificio terminal, reconocimiento de tendencias en retail para predecir ingresos comerciales, optimización del estacionamiento de las aeronaves para incrementar los ingresos comerciales, y cálculo de la hora estimada de puesta de calzos para apoyar a la gestión de las puertas de embarque y equipos de *handling*.

Para finalizar, tras un análisis previo y en base a la rápida y constante evolución de esta disciplina, en la elaboración de esta SLR se partió como criterio de inclusión que el año de la publicación fuera de 2010 en adelante, excluyendo las publicaciones anteriores. Con el fin de evaluar la validez de esta decisión a posteriori, la Figura 3.6 muestra la evolución anual del número de publicaciones encontradas por campo de estudio.

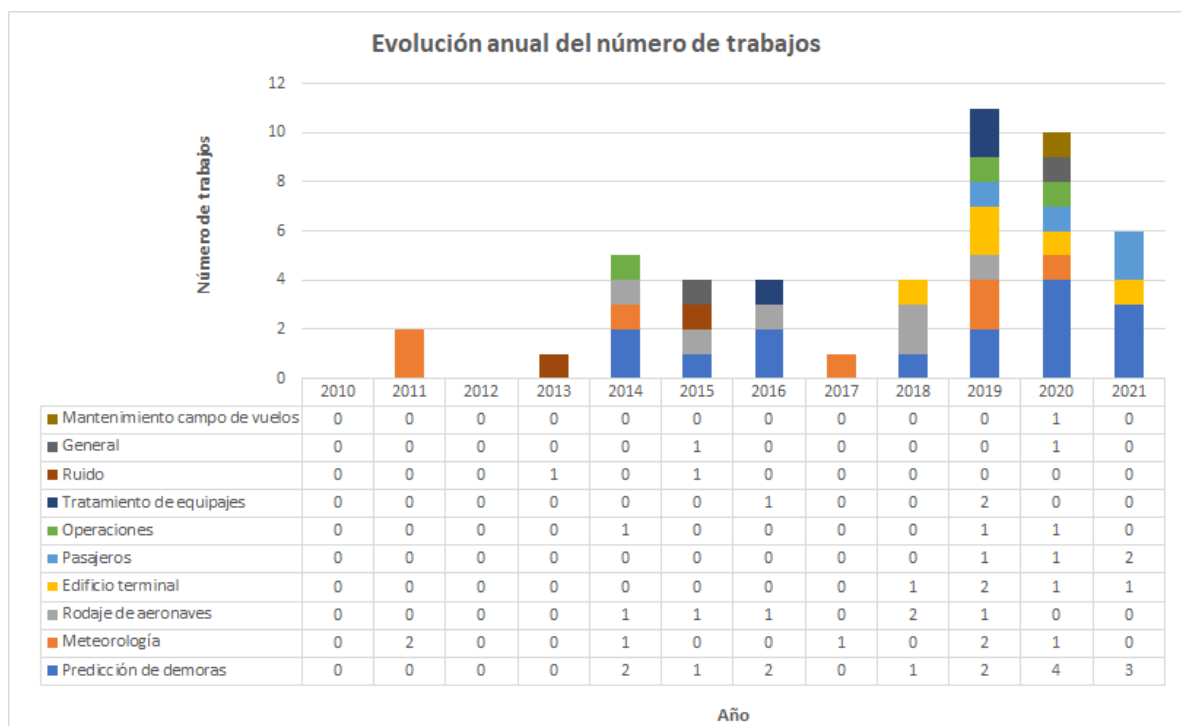


Figura 3.6: Evolución anual del número de trabajos (*Fuente: elaboración propia*)

Cabe señalar que los datos de 2021 corresponden solo hasta el mes de junio.

Como se puede apreciar, los primeros años del periodo elegido tienen muy pocas publicaciones, con ninguna en 2010 y 2012, dos en 2011 y una en 2013. Luego, en 2014 el número aumenta y se mantiene estable hasta 2018 (salvo el caso de 2017, con una única publicación). No obstante, es a partir de 2019 cuando se produce el gran salto en la tendencia, y los datos de la primera mitad de 2021 confirman la estabilización en esos niveles. Por tanto, se puede validar la asunción inicial.

Una vez hecha la revisión sistemática de la literatura, con las conclusiones expuestas, se ha decidido profundizar en este trabajo en la aplicación de *machine learning* en el edificio terminal para dar soporte a los distintos procesos que siguen los pasajeros en el mismo. Esta decisión se fundamenta en dos aspectos: el primero es que el edificio terminal y los servicios a los pasajeros son el campo donde se requiere mayor intervención en las decisiones del operador aeroportuario (por ejemplo, en ATC la responsabilidad está más dirigida al gestor del espacio aéreo); y el segundo es el gran esfuerzo invertido estos últimos años en la innovación para este campo, lo cual también se aprecia en la Figura 3.6.

### 3.2 APLICACIONES INSTALADAS ACTUALMENTE EN LOS AEROPUERTOS DEL MUNDO

Los resultados de la SLR responden a las preguntas de investigación desde una perspectiva más académica. Sin embargo, son muchas las empresas que desarrollan sus propias aplicaciones y servicios, y que pueden no aparecer en las búsquedas en los motores más científicos. Así, con el fin de identificar esas aplicaciones y completar los resultados, se incluye en este apartado una compilación de las aplicaciones instaladas actualmente en los aeropuertos del mundo, bien como proyectos piloto o ya en pleno funcionamiento.

La más visible desde el punto de vista del pasajero es el uso de la biométrica, que no es una aplicación exclusivamente de *machine learning*, sino que entran otras muchas disciplinas en juego, pero merece especial mención por el enorme cambio que está introduciendo en la forma de procesar a los pasajeros en los edificios terminales. Con registrar sus datos biométricos una única vez, el pasajero puede realizar el resto de los procesos sin presentar ningún tipo de documentación ni tener que tocar ningún equipo. En la actualidad, son

muchos los aeropuertos que tienen instalados sistemas biométricos y ya han demostrado su eficacia. A continuación, la Tabla 3.24 incluye algunos ejemplos.

Tabla 3.24: Ejemplos de biométrica en aeropuertos (*Fuente: elaboración propia*)

Aeropuerto	Sistema	Estado
Abu Dhabi International Airport	Smart Travel system	Piloto
Tokyo's Narita International Airport	'One ID' check in to boarding process	En uso
Beijing Capital International Airport	SITA Smart Path	En uso
Singapore Changi Airport	IDEMIA Fast and Seamless Travel (FAST)	En uso
Aruba International Airport	Aruba Happy Flow	En uso
Avalon Airport	Elenium Automation	En uso
Toronto Pearson International Airport	Known Traveller Digital Identity (KTDI)	Piloto
Vancouver International Airport	Innovative Travel Solutions BORDERXPRESS technology	En uso
Fort Lauderdale-Hollywood International Airport	Amadeus biometrics	En uso
McCarran International Airport y JFK International Airport	Collins Aerospace biometric technology	En uso
Lyon-Saint Exupery Airport	MONA - travel assistant by Vinci Airports	Piloto
Munich International Airport y Frankfurt Airport	Star Alliance Biometrics platform	En uso
London Heathrow Airport	dormakaba Self Boarding Gates and PaxCheck Self Boarding Solution	Piloto
Aeropuerto A.S. Madrid Barajas y Aeropuerto de Menorca	Reconocimiento facial Iberia, Inetum y Thales	Piloto

Actualmente, uno de los principales limitantes encontrados para esta tecnología es la reticencia de los pasajeros a compartir sus datos biométricos. Esto hace que por el

momento la biométrica esté instalada como complemento, dando la posibilidad de hacer uso de sistemas tradicionales si se desea. No obstante, se espera que con uso cada vez más común y la evolución paralela de la normativa de protección de datos esta limitación se vaya diluyendo y el futuro de las terminales pase por estar basado completamente en biométrica.

Una variante a las soluciones completamente biométricas son los controles ABC (*Automatic Border Control*) que sí están instalados en la mayor parte de aeropuertos internacionales del mundo y no solo como complemento, sino como único sistema para el control de pasaportes. En este caso, no se registran los datos biométricos, únicamente se pasa el pasaporte por un lector, que va acompañado de un sistema de reconocimiento facial o lector de huellas dactilares. El sistema compara ambas lecturas y hace *match* o *no match* para permitir el paso del pasajero.

Tras la pandemia del COVID-19, ambas tecnologías han incrementado su interés y su uso se ha acelerado notablemente, ya que permiten los procesos sin contacto. No obstante, ya antes de esta crisis apuntaban a ser el futuro del procesamiento de los pasajeros en las terminales de transporte.

Otra de las aplicaciones más emergentes es *computer vision*, cuyo uso más extendido en los aeropuertos consiste en combinar cámaras y algoritmos de *machine learning* para monitorizar las actividades de *handling*, detectar problemas de seguridad y emitir alertas. Uno de los primeros aeropuertos en instalar esta tecnología ha sido London Gatwick, y entre los beneficios más destacados se tiene la mejor gestión de los tiempos de rotación de las aeronaves y la mejora de la seguridad operacional.

En lo que respecta al sistema de tratamiento de equipajes y el *handling* en tierra, como en muchos otros procesos industriales, el ML permite la adopción del mantenimiento predictivo, cuya principal ventaja en este caso es poder ajustar las actuaciones sobre los equipos a las franjas con menor tránsito, de tal manera que el impacto en la operatividad es el mínimo posible.

Otra solución que ayuda a los aeropuertos en la gestión de equipajes mejorando los índices de lectura de códigos de barras tradicionales, es el uso de *deep learning* para el reconocimiento de caracteres que permite leer con precisión el texto de las etiquetas.

Asimismo, una de las aplicaciones más disruptivas en este campo es el uso de vehículos autónomos para el transporte de los equipajes. Por ejemplo, Vanderlande ofrece la solución FLEET, ya instalada en varios aeropuertos de USA y Europa, basada en vehículos autónomos y software inteligente. Estos vehículos transportan de forma autónoma los equipajes dentro de la sala y hasta los puestos de las aeronaves en plataforma, permitiendo una gestión flexible, eficiente y sostenible.

Por otro lado, uno de los campos con mayor expansión y que aporta más valor es el uso de *data analytics* para incrementar el conocimiento del comportamiento de los pasajeros. En este contexto, son numerosas las aplicaciones que proporcionan estos servicios a los gestores aeroportuarios, aunque destacan las siguientes:

- Crowdvision, [77]: ofrece una solución de análisis de datos de pasajeros en tiempo real. El sistema procesa las entradas de video en directo de sensores LIDAR y cámaras ópticas distribuidas por el terminal. Con ello, realiza recuento de pasajeros, análisis de colas y analiza el comportamiento de los flujos mediante sofisticadas técnicas de ML e IA. Esto proporciona una información muy valiosa en dos aspectos: primero, para una gestión eficiente de las operaciones en tiempo real (por ejemplo, si se detecta acumulación en colas en un proceso, se pueden activar más medios en tiempo real para solucionarlo); y por otra parte, permitiendo la creación de bases de datos históricas para su uso en planificación y en la toma de decisiones de inversión (por ejemplo, al almacenar datos del comportamiento de los flujos a lo largo del edificio, se conocen las áreas en las que se tiene mayor circulación y se puede aprovechar para incrementar los ingresos comerciales).
- Smart Flows, [78]: ofrece un sistema que obtiene datos de las infraestructuras de red existentes en el aeropuerto (Wi-Fi, Bluetooth, cámaras, etc.) para recoger las señales telefónicas de los pasajeros en tiempo real. Con estos datos y haciendo uso de tecnologías de ML e IA, el sistema genera mapas de calor, datos de tiempo de permanencia en cada área, densidades y flujos de pasajeros en tiempo real que permiten un monitoreo de la situación en vivo. Además, incluye una potente herramienta de análisis de datos que almacena toda la información recogida y permite analizar comportamientos pasados calculando las métricas clave que



describen el comportamiento del flujo (cuadros de tráfico por días, semanas y meses, perfiles de presentación, flujos por áreas e instalaciones, etc.), y realizar predicciones a futuro (tráfico previsto en todas las zonas, simulaciones de tiempos de permanencia, etc.).

Todo ello posibilita ajustar la activación de medios y los horarios del personal a las demandas esperadas en cada área e instalación. Y más aún, adaptar los datos de flujos de pasajeros a otros datos de las distintas unidades de negocio. En resumen, se convierten datos brutos que no aportan mayor información en datos procesables que otorgan conocimiento a los gestores para tomar las mejores decisiones.

- *SITA Passenger Flow Management*, [79]: proporciona supervisión automatizada de los pasajeros en tiempo real y emite alertas cuando se alcanza la capacidad de pasajeros de una zona definida, anticipando los cuellos de botella y la saturación. Esta información permite a los aeropuertos gestionar las áreas y equipos de forma proactiva, controlando el ratio de pasajeros, desviando flujos y desplegando recursos y personal adicionales en caso de ser necesario, así como alinear los perfiles de los pasajeros con las compañías aéreas, crear una experiencia de viaje que se adapte a sus necesidades e identificar las oportunidades de negocio en cada caso para aumentar los ingresos.

Con un impacto diferente y a menor escala, existen otras muchas aplicaciones que hacen uso de ML en el día a día de los aeropuertos. Algunos de los ejemplos más comunes son *chatbots* para la comunicación con los pasajeros, personalización de productos, plataformas de *e-commerce*, o etiquetas inteligentes para los equipajes.

Todas estas aplicaciones introducen importantes beneficios tanto para los aeropuertos como para el resto de *stakeholders*:

- Se mejora la seguridad y se reduce el error humano.
- Se reducen los tiempos de proceso.
- Se ofrece una mejor experiencia al pasajero.
- Se mejora la gestión de mayores volúmenes de pasajeros.
- Se reduce el OPEX.
- Se minimizan los impactos de eventos disruptivos.

- Se reducen demoras.

Con todo lo expuesto, se puede concluir que ya el presente, pero sobre todo el futuro próximo de las infraestructuras aeroportuarias está marcado por la aplicación cada vez más creciente de procesos *data-driven* y nuevas tecnologías de *machine learning* e inteligencia artificial para incrementar su eficiencia, competitividad y sostenibilidad, ofreciendo la mejor experiencia al pasajero.

### 3.3 NORMATIVA

La normativa que aplica en el uso de *machine learning* e inteligencia artificial abarca toda la regulación aplicable en términos de protección de datos, incluyendo a nivel europeo el Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo (Reglamento General de Protección de Datos, RGPD), y a nivel nacional la Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales (LO 3/2018).

También es de aplicación el Reglamento (UE) 2018/1807, de 14 de noviembre de 2018, relativo a un marco para la libre circulación de datos no personales en la Unión Europea, entendiéndose por datos de carácter no personal aquéllos que no están incluidos en el RGPD. Así como la Directiva 2002/58/CE, que fue transpuesta en España mediante Real Decreto-Ley 13/2012 de 30 de marzo, modificando la Ley 34/2002, de 11 de julio, de servicios de la sociedad de la información y de comercio electrónico (LSSI).

Esto viene acompañado por normativa todavía no vinculante como el “Libro Blanco sobre la inteligencia artificial – un enfoque europeo orientado a la excelencia y la confianza” de la Comisión Europea, la cual está trabajando para adoptar nuevas normas y medidas para favorecer la excelencia y la confianza en la inteligencia artificial.

Por otra parte, todo sistema que se instale en los aeropuertos españoles tiene que cumplir con:

- Normas y recomendaciones del Convenio sobre Aviación Civil Internacional y sus 19 Anexos, así como otros documentos de OACI.
- Normativa EASA a nivel europeo.
- Normativa AESA a nivel nacional.

Por último, se encuentra el ADRM de IATA que, si bien no representa una norma, se trata del documento de mayor relevancia y prestigio mundial para la planificación y el diseño de aeropuertos. En él se habla de cómo estas nuevas tecnologías favorecen el incremento de la eficiencia y una mejor experiencia de los pasajeros, e insta a los aeropuertos a explorar las nuevas soluciones que pueden mejorar su gestión.

Además, dentro de sus programas IATA ha incorporado el *Fast Travel*, *Smart Security*, *Automated Border Control*, o el OneID. Todos ellos hacen uso del aprendizaje automático y la inteligencia artificial en sus soluciones.

La última propuesta, el OneID, plantea la solución biométrica para hacer el flujo de los pasajeros más fluido y seguro a través del edificio terminal. Entre sus ventajas destaca:

- Reducción del 10% del tiempo de *check in*.
- Reducción del 40% del tiempo de proceso de embarque.
- Reducción del 4% del tiempo empleado en el control de seguridad.
- Reducción del 80% en tiempo invertido en el control de pasaportes para salidas y llegadas.

Esta propuesta se muestra en la Figura 3.7.



Figura 3.7: Mejora en los procesos IATA OneID (Fuente: [80])

## 4. Concepto *smart airports*

El concepto de *smart airport* o aeropuerto inteligente no tiene una definición única y varía entre los diferentes estudios dependiendo del ámbito en el que se esté tratando. Por ejemplo, Aena se refiere en [81] a plataformas inteligentes capaces de integrar múltiples fuentes de datos, de analizar de forma integral dicha información y de actuar en tiempo real sobre los procesos.

Por su parte, [82] se refiere al aeropuerto inteligente como un sistema de cooperación hombre-máquina que combina software y hardware, en el que tanto el gestor como los pasajeros, las aerolíneas y los proveedores de servicio participan para lograr la prestación de servicios, la difusión de información y la interacción entre las partes interesadas, lo que hace que el proceso de servicio sea más refinado, personalizado y eficiente.

Asimismo, la Agencia Europea para la Ciberseguridad, ENISA, lo define en [83] como aquel aeropuerto que hace uso de capacidades de respuesta en red y basadas en datos que, por un lado, proporcionan a los pasajeros una experiencia de viaje mejor y más fluida y, por otro, pretenden garantizar mayores niveles de seguridad para los pasajeros y los operadores. Estos aeropuertos están implementando nuevos componentes inteligentes para ofrecer a los pasajeros una cartera de servicios que abarca desde la auto facturación o la facturación automática y la comprobación de equipajes y documentos, hasta el control de fronteras y los controles de seguridad automatizados.

El concepto de *smart airport* viene a ser un sinónimo del Aeropuerto 4.0, ya que va de la mano de la transformación digital. Se trata básicamente de hacer uso de las tecnologías más avanzadas para, a partir de los grandes volúmenes de datos que se manejan, desarrollar soluciones estratégicas y de toma de decisiones en tiempo real, incrementar la eficiencia de las operaciones, y ofrecer la mejor experiencia al pasajero; todo ello basándose en la sostenibilidad, tanto en plano económico como en el social y medioambiental.

En este contexto, el *machine learning* y el campo completo de la inteligencia artificial juegan un papel clave, tal y como se aprecia en la Figura 4.1 que resume las principales características del concepto de *smart airport*.



Figura 4.1: Concepto de *smart airport* (Fuente: [84])

Este concepto engloba y debe garantizar tres áreas principales que, por el propio carácter del aeropuerto como sistema, están totalmente interrelacionadas: seguridad, experiencia del pasajero y sostenibilidad y resiliencia de la infraestructura.

## 4.1 SEGURIDAD

Entendida en sus dos definiciones de seguridad física (*security*) y seguridad operacional (*safety*). En el aeropuerto inteligente la garantía de ambas se incrementa con respecto a uno que no lo es.

Por una parte, existen estudios que han comprobado que la biométrica, apoyada también por otros algoritmos de ML, incrementa los niveles de seguridad en los aeropuertos, por ejemplo, introduciendo las siguientes mejoras:

- Gestión sólida de la identidad de cada persona y clasificación de los pasajeros y sus pertenencias en función del nivel de riesgo que representan, de tal manera que los protocolos y servicios de seguridad pueden concentrar sus medios y esfuerzos en el seguimiento y control de aquellos con mayor nivel de riesgo.
- Limitación de los errores humanos.
- Reducción de la posibilidad de que individuos crucen las fronteras con una identidad falsa, contribuyendo además a la lucha contra el tráfico de personas y otras actividades delictivas transfronterizas.
- Reducción de las acumulaciones de gente en las distintas áreas, haciéndolas menos vulnerables a actos de terrorismo.

Asimismo, la aplicación de técnicas de reconocimiento y clasificación de amenazas en el tratamiento de equipajes o la detección de equipajes u objetos abandonados a partir de imágenes de video también incrementan los niveles de seguridad ante actos de interferencia ilícita.

Por otra parte, en cuanto a la seguridad operacional, el *smart airport* hace uso de algoritmos para gestionar el movimiento de las aeronaves en tierra de una manera eficiente y segura, garantizando separaciones mínimas, detectando la intrusión de animales u objetos que puedan suponer un peligro, y calculando las rutas de rodaje óptimas en cada momento. También permite perfeccionar los procedimientos de seguridad en plataforma, mediante el monitoreo de las actividades de *handling*, la detección durante el mismo de problemas de seguridad y la emisión de alertas.

## 4.2 EXPERIENCIA DEL PASAJERO

Los avances tecnológicos adoptados por un aeropuerto inteligente tienen uno de sus mayores focos en ofrecer la mejor experiencia posible a los pasajeros. Y consiguen este objetivo por medio de los siguientes beneficios:

- Se reducen los tiempos de espera y los tiempos de proceso.
- Se bajan los niveles de estrés percibidos.
- Se evitan las acumulaciones de pasajeros.
- Se ofrece información en tiempo real de los distintos procesos a los pasajeros para que gestionen su tiempo.
- Se reducen los puntos de contacto y se simplifican los procesos.
- Se proporciona mayor conectividad.
- Se ofrece un servicio más personalizado basado en preferencias personales.
- Se proporciona una experiencia más *seamless* por su término en inglés, que viene a expresar un paso por el terminal fluido y sin fricciones.
- Se mejora la oferta comercial y de entretenimiento.

## 4.3 SOSTENIBILIDAD Y RESILIENCIA

El uso de técnicas basadas en datos proporciona mayor resiliencia a las infraestructuras aeroportuarias, mejorando su capacidad de adaptación a los cambios en tiempo real y de amortiguar el impacto de variaciones bruscas en la demanda de mayor duración.

Asimismo, estas soluciones ofrecen mejoras a la sostenibilidad, entendiéndose esta tanto desde el punto de vista medioambiental como económico.

Por una parte, con respecto al control de los impactos sobre el medio ambiente, se tienen los siguientes beneficios:

- Reducción del consumo de agua y energía.
- Control de los residuos generados.
- Mitigación del impacto del ruido aeroportuario.
- Control de las emisiones generadas durante los movimientos en tierra mediante una gestión eficiente del rodaje.

Por otra parte, en cuanto a la sostenibilidad económica de la infraestructura, se introducen los siguientes avances:

- Incremento de la certidumbre en la toma de decisiones, tanto operativas como de inversión, al aumentar el conocimiento de la operativa y las necesidades de activos.
- Potenciación de la transferencia de datos entre todos los involucrados que permite a todas las partes adaptarse y responder de manera más eficiente a los cambios.
- Reducción de los costes operativos mediante el uso flexible y proporcionado de los medios y el personal.
- Aumento de la eficiencia del personal y aumento de la capacidad al reducir el tiempo dedicado a los controles manuales de identificación.
- Mejora de la eficiencia del uso del espacio y oportunidades para aliviar la inversión adicional en infraestructura.
- Incremento de los ingresos comerciales a través de la creación de nuevas oportunidades de negocio y potenciando las existentes.

---

## 5. Aplicación práctica

### 5.1 DESCRIPCIÓN DEL CASO DE ESTUDIO

Como caso de estudio se plantea la construcción de un modelo para predecir el número de controles de seguridad que un aeropuerto debe tener abiertos en cada intervalo horario, de tal manera que permita al gestor aeroportuario realizar la asignación de medios de manera flexible y eficiente, ajustando los medios activos en cada momento a la demanda esperada.

Esto permite, entre otras cosas, reducir el OPEX mediante la reducción del consumo de energía, los gastos de personal, y las necesidades de mantenimiento de los equipos. Mientras, el CAPEX se incrementa ligeramente por la necesidad de inversión para la implementación de este tipo de tecnologías, pero la reducción del OPEX es más notable por lo que el periodo de recuperación de la inversión es corto.

Por otra parte, el control de seguridad es el proceso que mayor incomodidad y estrés genera a los pasajeros. La solución que se propone en este caso de estudio permite reducir el tiempo invertido por los pasajeros en este proceso y hacerlo más *seamless*, ya que se reducen las esperas y se evita la acumulación de personas. Esto hará que su experiencia sea mejor y que ese tiempo ahorrado se convierta en tiempo de permanencia (*dwelling time*) en la zona comercial del aeropuerto, favoreciendo el consumo y, por tanto, el aumento de los ingresos.

El resultado total es un pasajero más satisfecho y un aeropuerto más eficiente operativa y económicamente.

### 5.2 ANÁLISIS Y PROCESAMIENTO DE LOS DATOS

El primer paso antes de crear el modelo es analizar y procesar los datos disponibles de partida, de manera que no incluyan anomalías o datos perdidos que puedan afectar negativamente a la predicción obtenida. Además, como se trata de datos reales de operación, es necesario ajustarlos al formato de entrada en los algoritmos.



---

Como datos de partida se tiene un registro de operaciones y pasajeros en un aeropuerto europeo mediano entre enero de 2016 y septiembre de 2018. Los datos que se incluyen en el registro son:

- Año Real: incluye el año de la operación y, por tanto, presenta uno de los tres valores 2016, 2017 o 2018.
- Mes Real: incluye el mes en que tiene lugar la operación, en letra.
- Mes: incluye la misma información que las columnas anteriores, pero con el formato “mes abreviado” - “año”.
- Día del Mes: incluye el día del mes en que se produce la operación y, por tanto, toma valores del 1 al 31.
- Día de la Semana Real: incluye el día de la semana en letra, es decir, varía entre los días de lunes a domingo.
- Temporada Real: incluye la temporada en la que se produce la operación.

En el ámbito de las operaciones de aeronaves, se distinguen 2 temporadas: verano e invierno (*S – Summer* y *W – Winter*). La temporada de verano abarca desde el último domingo de marzo al sábado anterior al último domingo de octubre de un mismo año; mientras que la de invierno va desde el último domingo de octubre al sábado anterior al último domingo de marzo del año siguiente. Por tanto, en un año se tienen tres temporadas distintas.

Esta información de la temporada se utiliza, sobre todo, en la gestión de los slots aeroportuarios. El formato que sigue esta columna es “Temporada” “año”.

- Temporada: incluye la misma información que la columna anterior, pero en formato resumido, es decir, en lugar de incluir el nombre completo de la temporada, incluye “S” o “W”, y en vez de incluir el año completo, incluye 16, 17 o 18.
- Hora Operación Real: incluye la hora de operación en UTC, en formato con únicamente la hora, independientemente del minuto.

En el campo de las operaciones de aeronaves, tanto los datos de capacidad como de demanda se suelen referir a datos horarios y de ahí que en muchos casos se agrupen los datos de esta manera. Es decir, por ejemplo, para las 12 h se incluirán todos los datos registrados entre las 12:00:00 h y las 12:59:59 h.

---

Cabe señalar que con hora de operación se habla de hora de despegue o de aterrizaje de la aeronave.

- Movimiento: incluye la información de si la operación es salida o llegada, en letra.
- Pasajeros Total: incluye la suma de todos los pasajeros operados bajo las condiciones incluidas en las columnas anteriores.

Una vez analizado el conjunto de datos y comprendido el significado de las distintas variables, es necesario realizar una serie de simplificaciones o hipótesis de partida:

1. Dado que no se tiene información del tipo de pasajero (negocios, turismo, etc.) se considera que todos los pasajeros pasan por los controles de seguridad estándar, es decir, no se consideran los controles *fast track*.
2. La diferencia entre la hora de la operación (que se refiere a la hora en la que el avión aterriza o despegue) y el momento en que los pasajeros se presentan en el control de seguridad depende tanto del tipo de pasajero como del destino (nacional/internacional o corto/largo alcance).

Como no se dispone de esta información, se propone aplicar un intervalo de 2 horas entre el paso por el control de seguridad y la hora de despegue para todos los pasajeros.

Con la descripción del problema a resolver, y en base a las definiciones de las distintas técnicas de ML expuestas a lo largo del documento, se plantea la aproximación para la resolución del caso mediante la aplicación de técnicas de aprendizaje automático supervisado.

El lenguaje de programación elegido para el desarrollo de este trabajo es Python.

Una vez cargado el conjunto de datos inicial, se buscan datos faltantes y registros duplicados. El resultado de este proceso se recoge en la Figura 5.1.

Los datos nulos en cada columna son:

Año Real	0
Mes Real	0
Mes	0
Día del Mes Real	0
Día de la Semana Real	0
Temporada Real	0
Temporada	0
Hora Operacion Real	0
Movimiento	0
Pasajeros Total	0

Los registros duplicados son: 0

Figura 5.1: Resultados de datos faltantes y duplicados (*Fuente: elaboración propia*)

A simple vista se puede comprobar que ninguna columna presenta datos faltantes y que no se encuentran registros duplicados.

Tras esta comprobación, se analiza la distribución de los pasajeros según distintas combinaciones de variables, de tal manera que se puede realizar un estudio de cómo se comportan estos.

Primero, se analiza el comportamiento mensual. Nótese que para el año 2018 no se tiene datos de los tres últimos meses.

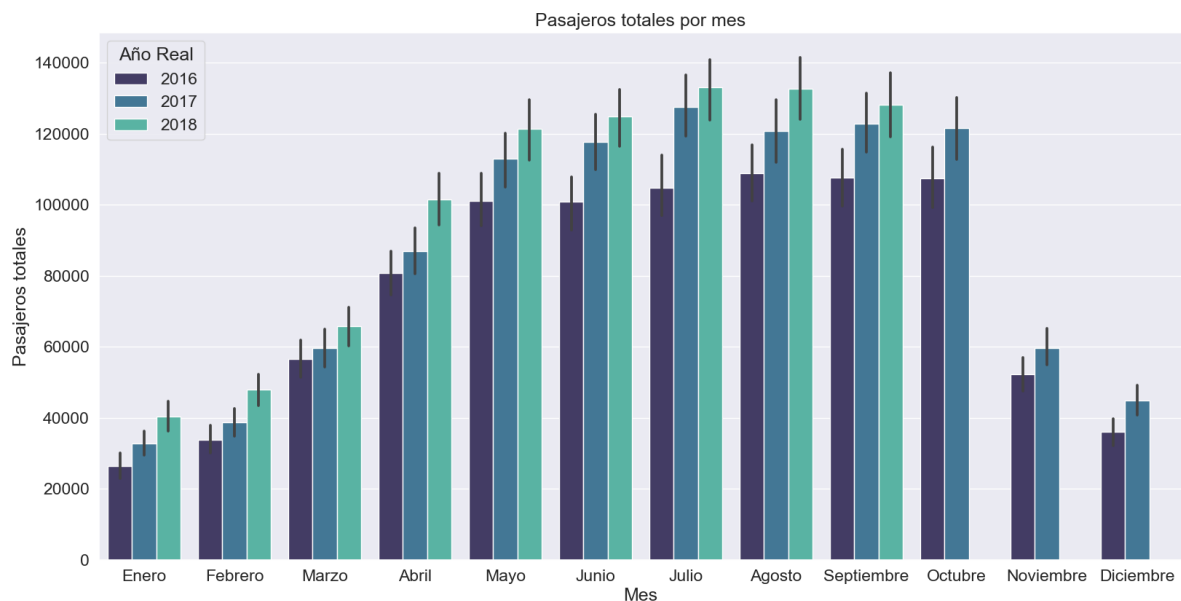


Figura 5.2: Pasajeros totales por mes (*Fuente: elaboración propia*)

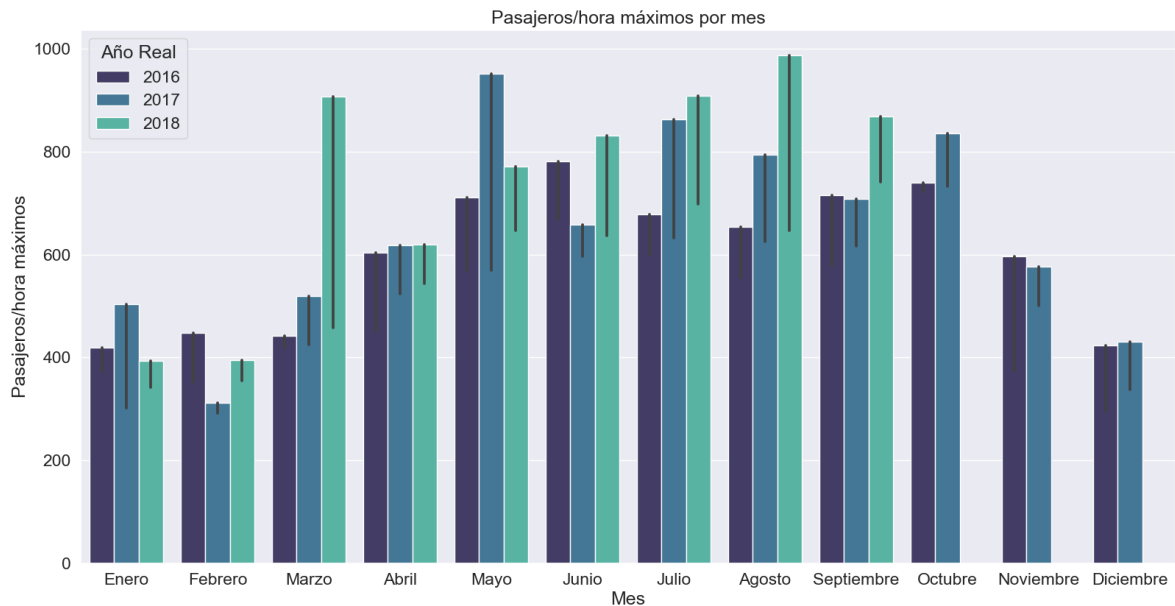


Figura 5.3: Pasajeros/hora máximos por mes (Fuente: elaboración propia)

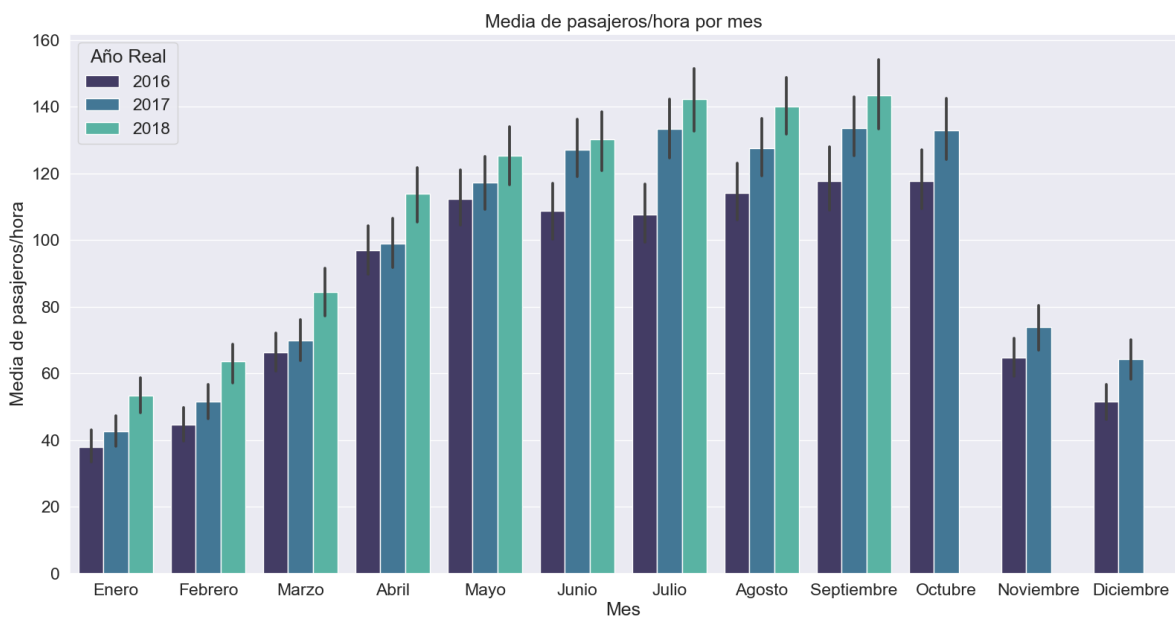


Figura 5.4: Media de pasajeros/hora por mes (Fuente: elaboración propia)

De las Figura 5.2, Figura 5.3 y Figura 5.4 se pueden extraer las siguientes conclusiones:

1. Los meses de la temporada de verano son los que presentan mayor número de pasajeros, mientras que en la temporada de invierno se reducen notablemente.

2. Los valores de pasajeros totales presentan un crecimiento anual progresivo, mientras que este comportamiento no se da en los pasajeros/hora. Esta es una característica muy común de la demanda de pasajeros en los aeropuertos. Que se produzca un incremento en el número de pasajeros totales no conlleva necesariamente a que también aumenten los pasajeros/hora, ya que este último valor depende de la casuística de operación establecida en el aeropuerto.
3. Existe una diferencia importante entre la media de pasajeros/hora y el valor máximo de pasajeros/hora. Esto indica que en el aeropuerto se dan horas pico y horas valle.

Para seguir con este análisis y caracterización de la demanda, se representan en las Figura 5.5, Figura 5.6 y Figura 5.7 los datos de pasajeros en función del día de la semana.

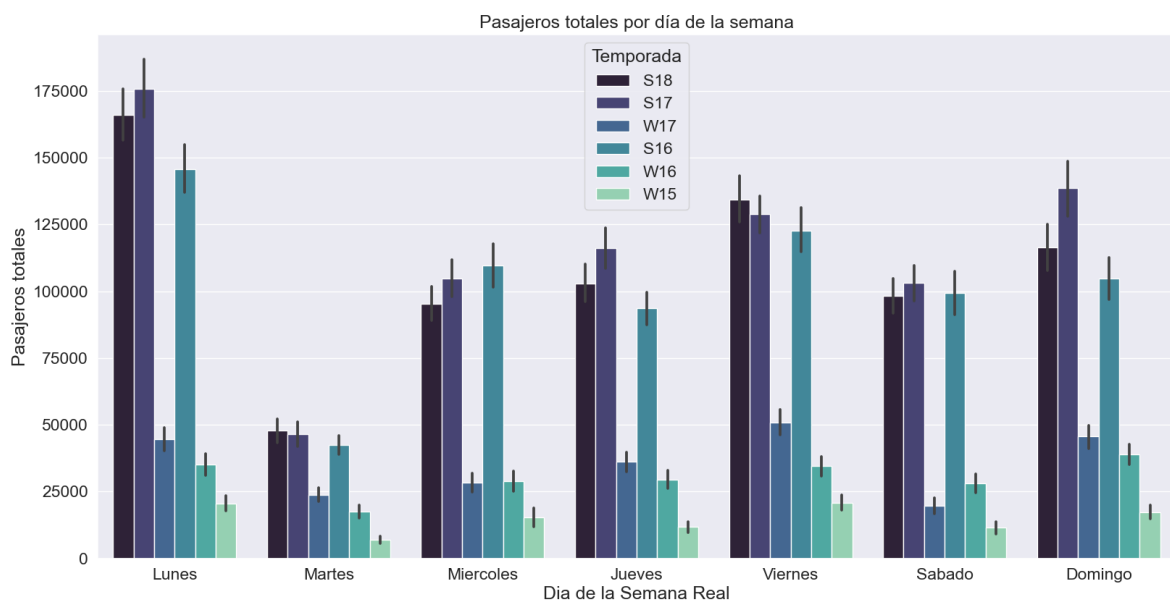


Figura 5.5: Pasajeros totales por día de la semana (*Fuente: elaboración propia*)

En la Figura 5.5 se puede apreciar que los lunes son los días de la semana con mayor número de pasajeros totales y los martes son los días de menor afluencia.

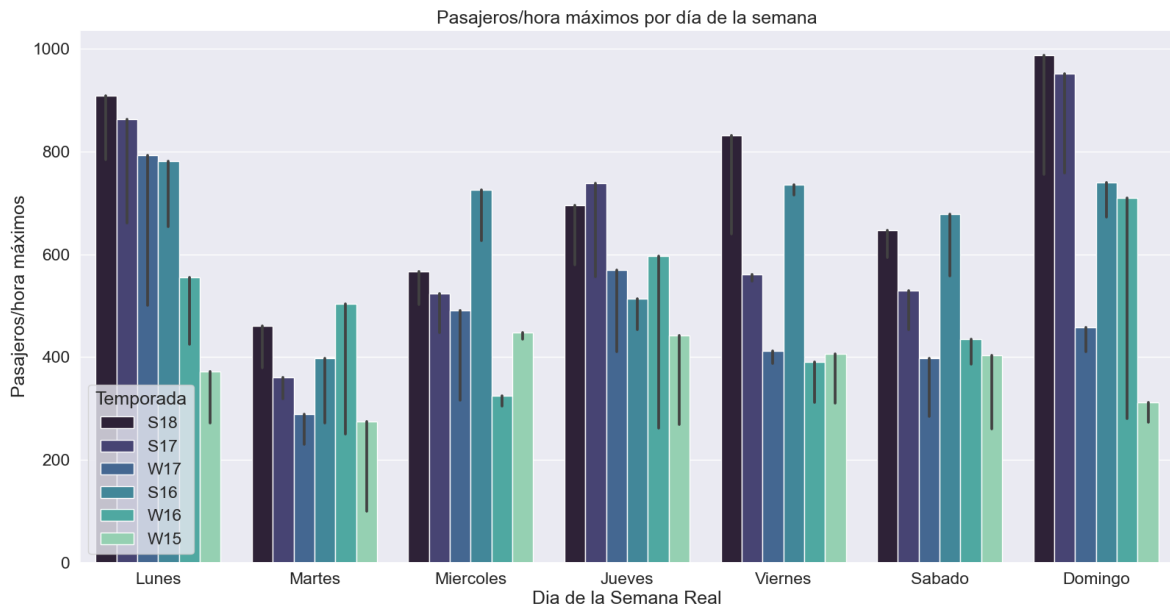


Figura 5.6: Pasajeros/hora máximos por día de la semana (*Fuente: elaboración propia*)

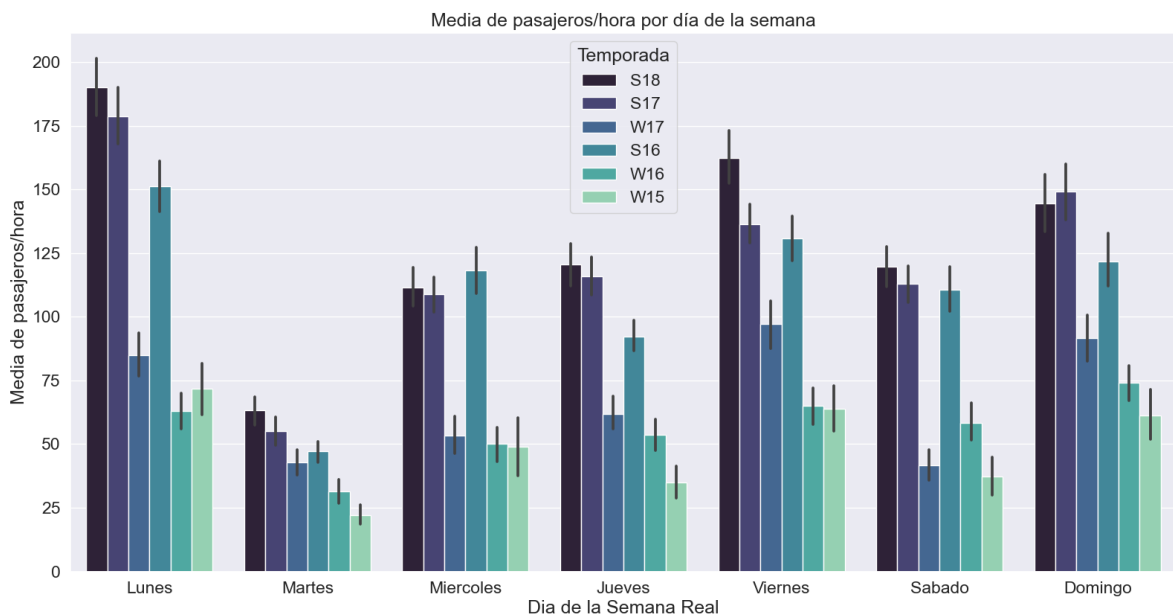


Figura 5.7: Media de pasajeros/hora por día de la semana (*Fuente: elaboración propia*)

En cuanto a valores de pasajeros/hora de las Figura 5.6 y Figura 5.7, se concluye que los lunes presentan el mayor valor medio, pero la punta máxima se da los domingos.

Finalmente, se representan los pasajeros/hora máximos y medios en cada hora del día.

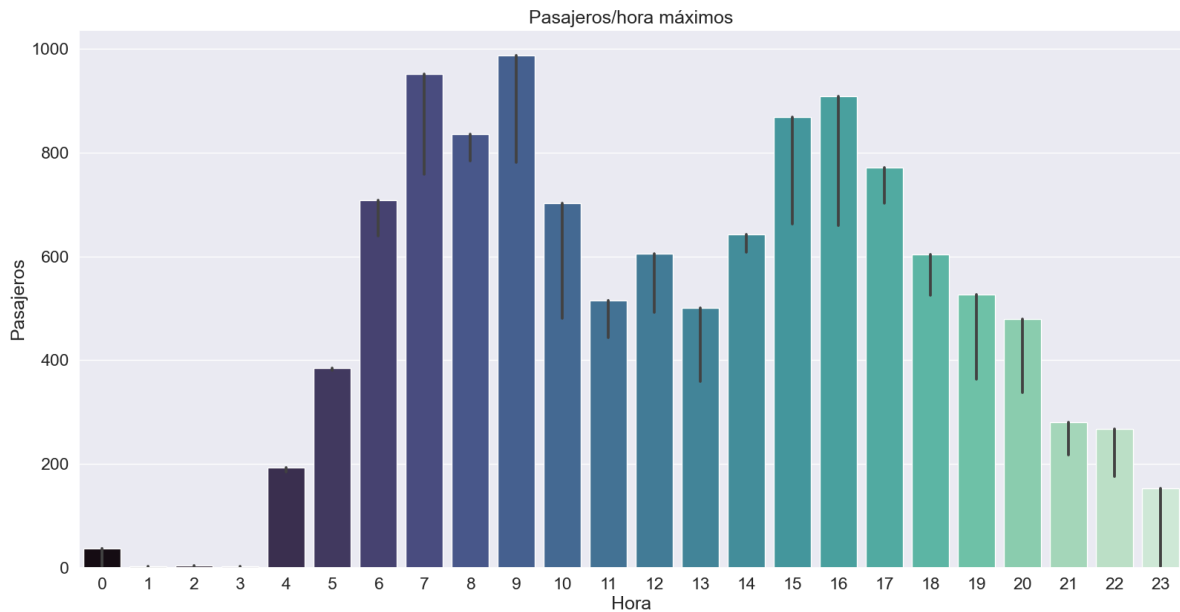


Figura 5.8: Pasajeros/hora máximos (Fuente: elaboración propia)

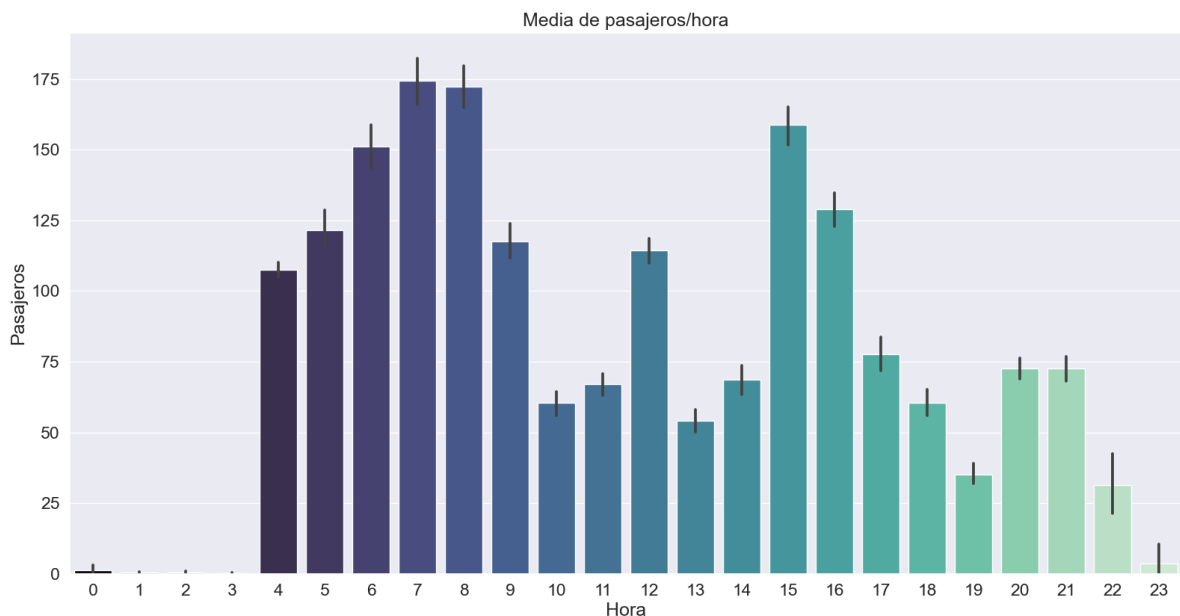


Figura 5.9: Media de pasajeros/hora (Fuente: elaboración propia)

En las Figura 5.8 y Figura 5.9 se puede apreciar una punta a primera hora de la mañana (entre las 7 h y las 9 h) y a primera hora de tarde (entre las 15 h y las 17 h). Además, entre las 23 h y las 3 h apenas se tiene presencia de pasajeros. Esto confirma la presencia de horas punta y horas valle.

Una vez caracterizada la demanda, se pasa a realizar el procesamiento de los datos. En las Figura 5.10 y Figura 5.11 se puede observar el tipo de dato leído en Python.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 28516 entries, 0 to 28515
Data columns (total 10 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Año Real              28516 non-null  int64
1   Mes Real              28516 non-null  object
2   Mes                  28516 non-null  datetime64[ns]
3   Día del Mes Real     28516 non-null  int64
4   Día de la Semana Real 28516 non-null  object
5   Temporada Real       28516 non-null  object
6   Temporada            28516 non-null  object
7   Hora Operacion Real  28516 non-null  int64
8   Movimiento           28516 non-null  object
9   Pasajeros Total      28516 non-null  int64
dtypes: datetime64[ns](1), int64(4), object(5)
memory usage: 2.2+ MB
```

Figura 5.10: Datos originales leídos en Python (*Fuente: elaboración propia*)

	Año Real	Mes Real	Mes	Día del Mes Real	Día de la Semana Real	\
0	2018	Agosto	2018-08-01	26	Domingo	
1	2018	Marzo	2018-03-01	26	Lunes	
2	2017	Mayo	2017-05-01	28	Domingo	
3	2018	Septiembre	2018-09-01	10	Lunes	
4	2018	Julio	2018-07-01	23	Lunes	

	Temporada Real	Temporada	Hora Operacion Real	Movimiento	Pasajeros Total
0	Verano 2018	S18	9	SALIDA	987
1	Verano 2018	S18	16	LLEGADA	907
2	Verano 2017	S17	7	SALIDA	951
3	Verano 2018	S18	15	LLEGADA	868
4	Verano 2018	S18	16	SALIDA	908

Figura 5.11: Muestra datos originales (*Fuente: elaboración propia*)

Lo primero que se puede concluir es que la columna “Mes” no aporta información nueva, ya que simplemente agrupa información contenida en las otras columnas y, por tanto, puede eliminarse. Lo mismo ocurre con la columna “Temporada Real” que únicamente incluye la información contenida en “Temporada”, pero con otro formato, por lo que también se puede eliminar.



Asimismo, vista la distribución mensual de la demanda se decide realizar el estudio por temporada en lugar de por mes, por lo que la columna “Mes Real” también se puede eliminar.

Por otra parte, como se quiere aplicar al control de seguridad, únicamente interesan los datos de salida, es decir, las filas cuyas columnas “Movimiento” indiquen “LLEGADA” no interesan en este caso de aplicación. Por tanto, primero se eliminan estas filas y, posteriormente, se elimina esa columna ya que en todo caso tendrá el valor “SALIDA” y, por tanto, no aportará información.

Tras eliminar la información redundante, analizando el tipo de variable que incluye cada columna, se puede observar que existen columnas tipificadas como *object*, las cuales se refieren a variables categóricas. Para que el modelo de *machine learning* a implementar pueda leerlas, es necesario convertir estas variables categóricas en variables numéricas. Para ello, existen dos métodos: codificación ordinal y codificación *one-hot*.

La codificación ordinal consiste en reemplazar cada valor de las variables categóricas por un número ordinal distinto; mientras que el *one-hot* crea una variable binaria (conocida como variable *dummy*) para cada valor de las variables categóricas. Para este caso, se elige la aplicación del segundo método, ya que la clasificación ordinal introduce una cierta jerarquía entre las variables que puede provocar que el modelo malinterprete los datos.

Las variables que hay que convertir son “Día de la Semana Real” y “Temporada”. Para convertirlas se utiliza *get.dummies* de Pandas.

Por último, como se ha comentado anteriormente, la columna “Hora Operación Real” incluye el dato de la hora de despegue de la aeronave, pero los pasajeros se presentan antes en el control de seguridad (se ha establecido como hipótesis 2 h de diferencia). Por tanto, se crea una nueva columna que incluye la hora de paso por el control llamada “hora\_control”, que únicamente es una resta de -2 h a la “Hora Operación Real”.

Tras hacer los pasos anteriores, los datos se convierten a los mostrados en la Figura 5.12.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14213 entries, 0 to 14212
Data columns (total 16 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Año Real              14213 non-null  int64
1   Día del Mes Real     14213 non-null  int64
2   Hora Operacion Real  14213 non-null  int64
3   Pasajeros Total      14213 non-null  int64
4   Jueves                14213 non-null  uint8
5   Lunes                14213 non-null  uint8
6   Martes               14213 non-null  uint8
7   Miercoles            14213 non-null  uint8
8   Sabado               14213 non-null  uint8
9   Viernes              14213 non-null  uint8
10  S17                  14213 non-null  uint8
11  S18                  14213 non-null  uint8
12  W15                  14213 non-null  uint8
13  W16                  14213 non-null  uint8
14  W17                  14213 non-null  uint8
15  hora_control         14213 non-null  int64
dtypes: int64(5), uint8(11)
memory usage: 708.0 KB
```

Figura 5.12: Datos leídos tras el procesamiento (*Fuente: elaboración propia*)

Se puede observar que:

- Las columnas con información redundante se han eliminado.
- El número de registros se ha reducido considerablemente al eliminar los datos correspondientes a las llegadas.
- Tras la conversión de las variables categóricas mediante *one-hot*, se han creado tantas columnas como valores distintos de las variables había menos 1. Esto es porque el método está añadiendo una variable extra redundante que es combinación lineal de las otras y, por tanto, se elimina ya que si el resto de las categorías contienen un 0, se entiende que la categoría correspondiente es la de la variable eliminada.
- Aparece una nueva columna con la hora de paso de los pasajeros por el control de seguridad.

Una vez analizados y preparados los datos, se comienza con la creación del modelo.

### 5.3 CONSTRUCCIÓN DEL MODELO

La construcción del modelo consta de dos fases con dos funciones principales:

1. Calcular el número de controles de seguridad necesarios en función de los pasajeros/hora.
2. Realizar las predicciones.

Para lo primero, se aplica la metodología definida por IATA en su ADRM, 11 edición. Este manual es la referencia más empleada y de mayor reputación a nivel mundial para la planificación aeroportuaria y la determinación de necesidades operativas y requerimientos de sistemas y áreas en los aeropuertos.

En concreto, para el control de seguridad plantea la ecuación 5.1.

$$SS = \frac{Demand \cdot PT / 60}{\Delta t + MQT} \quad (5.1)$$

Donde:

- SS: número de controles de seguridad requeridos
- *Demand*: número de pasajeros en el intervalo de tiempo considerado. Generalmente, se utilizan las demandas en los intervalos de 15 min, 30 min, 45 min y 60 min, se hacen los cálculos para cada uno de ellos, y el intervalo con mayor requerimiento es el que se aplica. En este caso, los datos disponibles son por cada hora, por lo que el intervalo aplicado es el de 60 min, y la demanda es la correspondiente a los pasajeros por hora.
- PT: tiempo de proceso, en segundos. Se refiere al tiempo que cada pasajero tarda en pasar el control de seguridad (sin incluir el tiempo de espera en cola).
- $\Delta t$ : intervalo de tiempo considerado en relación a la demanda, en minutos. Como se ha explicado este valor es de 60 minutos.
- MQT: tiempo máximo de espera en fila, en minutos.

Por tanto, *demand* es en este caso la columna de "Pasajeros Total" del conjunto de datos, y  $\Delta t$  es 60 min, como se ha explicado.

En cuanto al tiempo de proceso, generalmente, en los aeropuertos se llevan a cabo estudios de mediciones de tiempos de procesos que permiten al gestor tener un conocimiento del tiempo empleado por los pasajeros en la realización de los distintos

procesos que llevan a cabo (*check in*, control de seguridad, control de pasaportes, etc.). Los datos obtenidos en esos estudios son lo que suelen emplearse para este tipo de cálculos, pero en el momento de redacción de este trabajo no se ha tenido acceso a los mismos. No obstante, existen valores de tiempos de proceso basados en la experiencia internacional comúnmente utilizados en planificación. Para un aeropuerto mediano como el que se está analizando, se considera representativo utilizar un PT de 15 segundos. Por último, el tiempo máximo de espera en fila se define en función del nivel de servicio (LoS – *Level of Service*) que se proporciona en el aeropuerto, según la Figura 5.13.

LoS Guidelines	SPACE GUIDELINES [sqm/PAX]			MAXIMUM WAITING TIME GUIDELINES Economy Class [minutes]			MAXIMUM WAITING TIME GUIDELINES Business Class / First Class / Fast Track [minutes]			OTHER GUIDELINES & REMARKS			
	LoS Parameter:	Over-Design	Optimum	Sub-Optimum	Over-Design	Optimum	Sub-Optimum	Over-Design	Optimum	Sub-Optimum	Over-Design	Optimum	Sub-Optimum
Public Departure Hall		> 2.3	2.0 - 2.3	< 2.0		n/a			n/a				Optimum proportion of seated occupants: 15 - 20%*
Check-In	Self-Service Kiosk (Boarding Pass / Bag Tagging)	> 1.8	1.3 - 1.8	< 1.3	< 1	1 - 2	> 2	< 1	1 - 2	> 2			
	Bag Drop Desk (queue width 1.4 - 1.6m)	> 1.8	1.3 - 1.8	< 1.3	< 1	1 - 5	> 5	< 1	1 - 3	> 3			
	Check-in Desk (queue width: 1.4 - 1.6m)	> 1.8	1.3 - 1.8	< 1.3	< 10	10 - 20	> 20	< 3	Business Class 3 - 5	> 5			
Security Control (queue width: 1.2m)		> 1.2	1.0 - 1.2	< 1.0	< 5	5 - 10	> 10	< 1	Fast Track 1 - 3	> 3			
	Emigration Control (Outbound Passport Control) (queue width: 1.2m)	Staffed Emigration Desk	> 1.2	1.0 - 1.2	< 1.0	< 5	5 - 10	> 10	< 1	Fast Track 1 - 3	> 3		
	Automatic Border Control	> 1.2	1.0 - 1.2	< 1.0	< 1	1 - 5	> 5		n/a				
Gate Holdrooms ***	Seating	> 2.2	1.8 - 2.2	< 1.8		n/a			n/a				Optimum proportion of seated occupants: 50 - 70%*
	Standing	> 1.5	1.2 - 1.5	< 1.2		n/a			n/a				
Immigration Control (Inbound Passport Control) (queue width: 1.2m)	Staffed Immigration Desk	> 1.2	1.0 - 1.2	< 1.0	< 5	5 - 10	> 10	< 1	Fast Track 1 - 5	> 5			
	Automatic Border Control	> 1.2	1.0 - 1.2	< 1.0	< 1	1 - 5	> 5		n/a				
Baggage Reclaim	Narrow Body Aircraft	> 1.7	1.5 - 1.7	< 1.5	< 0	0 / 15	> 15	< 0	0 / 15	> 15			The first waiting time value relates to "first passenger to first bag". The second waiting time value relates to "last bag on belt" (counting from the first bag delivery).**
	Wide Body Aircraft	> 1.7	1.5 - 1.7	< 1.5	< 0	0 / 25	> 25						
Customs Control		> 1.8	1.3 - 1.8	< 1.3	< 1	1 - 5	> 5	< 1	1 - 5	> 5			Waiting times refer to a procedure when 100% of the passengers are being checked by Customs
Public Arrival Hall		> 2.3	2.0 - 2.3	< 2.0		n/a			n/a				Optimum proportion of seated occupants: 15 - 20%*

\* Lower limit to be considered only if extensive F+B seating is provided (within concession zones)

\*\* The time between the first passenger arriving at the reclaim belt and the first baggage arriving on the reclaim belt should be zero minutes, in order to maximize the efficiency of checking a hold bag for the passenger. Bags delivered to the reclaim prior to passengers arriving at the reclaim belt (negative waiting times) can be considered over-design. The time to deliver all bags from a flight should be no more than first-bag delivery +15 minutes for narrow body aircraft flights and +25 minutes for a wide body aircraft flights.

\*\*\* The space requirements for Gate Holdrooms have been updated incorporating the Maximum Occupancy factor in the space requirements

NB with regards to chapter 3.4.5.2 - LoS Category UNDER-PROVIDED: For processing facilities, the LoS UNDER-PROVIDED only results when both space and waiting time parameters are sub-optimum. For the boarding gate lounge and holdrooms, the LoS UNDER-PROVIDED only results when space parameter and seating requirements is sub-optimum. For the public departure and arrival halls, the LoS UNDER-PROVIDED only results when the space per occupant is 80% or less than the targeted optimum LoS parameter.

Figura 5.13: Niveles de servicio definidos por IATA (Fuente: [85])

Es decir, IATA clasifica este LoS en tres niveles: *over-design* (sobredimensionado), *optimum* (óptimo) y *sub-optimum* (infra dimensionado); y, para cada uno de ellos, establece un tiempo de espera máximo. En este caso, se selecciona el nivel de servicio óptimo, definido por IATA como aquel en el que se dispone de suficiente espacio para acomodar satisfactoriamente a la demanda, con tiempos aceptables de procesos y espera en colas,

por ser el valor con el que se planifican y diseñan las infraestructuras aeroportuarias. Por tanto, el MQT tiene que estar entre los 5 min y los 10 min. Se aplica el valor de 10 min, típicamente empleado en planificación.

Con todo esto, se añade una columna al conjunto de datos que guarda el número de controles de seguridad necesarios correspondiente a cada valor de pasajeros/hora. La Figura 5.14 muestra que esta columna queda incluida en el conjunto.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14213 entries, 0 to 14212
Data columns (total 17 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Año Real              14213 non-null  int64
1   Día del Mes Real     14213 non-null  int64
2   Hora Operacion Real  14213 non-null  int64
3   Pasajeros Total      14213 non-null  int64
4   Jueves               14213 non-null  uint8
5   Lunes               14213 non-null  uint8
6   Martes              14213 non-null  uint8
7   Miercoles           14213 non-null  uint8
8   Sabado              14213 non-null  uint8
9   Viernes             14213 non-null  uint8
10  S17                 14213 non-null  uint8
11  S18                 14213 non-null  uint8
12  W15                 14213 non-null  uint8
13  W16                 14213 non-null  uint8
14  W17                 14213 non-null  uint8
15  hora_control        14213 non-null  int64
16  Controles           14213 non-null  float64
dtypes: float64(1), int64(5), uint8(11)
memory usage: 819.0 KB
```

Figura 5.14: Datos leídos tras añadir columna de “Controles” (*Fuente: elaboración propia*)

El siguiente paso es, por tanto, la generación del modelo de predicción. Como ya se ha comentado, se trata de un problema de aprendizaje supervisado. Asimismo, cabe señalar que el objetivo del modelo es ayudar en la gestión de los medios disponibles actualmente en el aeropuerto, no determinar las necesidades futuras de equipos, por lo que su aplicación presupone que el aeropuerto cuenta con los controles necesarios para hacer frente a la demanda esperada. Se sabe que el aeropuerto actualmente cuenta con 4 controles de seguridad instalados. Con todo ello, el caso se convierte en un problema de clasificación, en el que se pretende predecir qué número de controles de esos 4 disponibles se necesitan activos (en operación).

Para la implementación, se aplican tres algoritmos distintos: Random Forest, K-NN y SVM; se comparan los resultados obtenidos con cada uno de ellos y se elige el que mejor calidad en la predicción ofrezca. Para ello se hace uso de la librería *Scikit-Learn* de Python.

Se comienza con la definición de las variables independientes (características o *features*) y de la variable dependiente (variable a predecir). Según las variables incluidas en la Figura 5.14, se definen como sigue:

- Variables independientes, x:
  - hora\_control
  - Jueves
  - Lunes
  - Martes
  - Miercoles
  - Sabado
  - Viernes
  - S17
  - S18
  - W15
  - W16
  - W17

Es decir, se eligen como variables independientes la hora de paso por el control, el día de la semana y la temporada.

- Variable dependiente, y:
  - Controles

Una vez establecidas las variables, se implementan los predictores.

### 5.3.1 Implementación Random Forest

El Random Forest es un método de *ensemble* de  $n$  árboles de decisión que hace uso del *bootstrap aggregation* en el entrenamiento. Esto quiere decir que si se tienen  $n$  árboles, cada uno de ellos entrena con una muestra aleatoria diferente de los datos y ninguno “ve”

el conjunto total de datos de entrenamiento. En problemas de clasificación, cada árbol da un resultado de clasificación, y la predicción final se elige como la opción más votada (la clase más frecuente).

La librería *Scikit-Learn* incluye el *RandomForestClassifier* con hiperparámetros que vienen definidos por defecto. El modelo funciona bien con ellos, pero con el fin de obtener los mejores resultados posibles, se realiza una optimización de hiperparámetros basada en la validación cruzada utilizando el método de *GridSearch*.

Los hiperparámetros que vienen por defecto son los definidos en la Figura 5.15.

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=100, *, criterion='gini', max_depth=None, min_samples_split=2,
min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0,
bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, class_weight=None,
ccp_alpha=0.0, max_samples=None)
```

Figura 5.15: Hiperparámetros por defecto Random Forest (*Fuente: [86]*)

Según [87] y el resto de bibliografía consultada, los más útiles son:

- `n_estimators`: número de árboles. En general, aumentar el número de árboles nunca empeora la predicción, y cuantos más árboles se utilicen mejor. No obstante, sí existe un número a partir del cual los resultados no mejoran (se mantienen) y lo único que se consigue es que el modelo sea más lento.
- `n_jobs`: número de cores de la CPU que se pueden usar en el entrenamiento. Como cada árbol es independiente del resto, el entrenamiento es paralelizable.
- `max_features`: número máximo de características para cada árbol.
- `max_depth`: profundidad máxima del árbol. Si se deja en “None”, los nodos se expanden hasta que todas las hojas sean puras o hasta que todas las hojas contengan menos muestras que `min_samples_split`.
- `min_samples_split`: número mínimo de muestras necesarias antes de dividir el nodo.
- `min_samples_leaf`: número mínimo de muestras que debe haber en un nodo final (hoja).
- `max_leaf_nodes`: número máximo de nodos finales.

De estos, los que más influyen en los resultados de la predicción son el número de árboles y la profundidad máxima. Por tanto, se aplica *GridSearch* para buscar los valores óptimos de ambos. Los resultados muestran que los valores óptimos son:

- `n_estimators = 1000`
- `max_depth = 70`

Por tanto, se crea el modelo definiendo estos valores y dejando en el resto los definidos por defecto.

Asimismo, el *RandomForestClassifier* permite hacer un análisis de la importancia de las características con el fin de identificar si alguna de las elegidas tiene muy poca importancia en la predicción con respecto al resto y si está introduciendo ruido, de tal manera que se pueda eliminar. Los resultados del modelo construido muestran lo incluido en la Figura 5.16.



Figura 5.16: Importancia de las características (*Fuente: elaboración propia*)

Es decir, claramente se observa que la característica más influyente es la hora de paso por el control de seguridad, mientras que el resto tienen valores aproximados de importancia. Esto es lo esperado de acuerdo a las figuras incluidas en el apartado 5.2 sobre el comportamiento de la demanda.



Con estos resultados, se decide no eliminar ninguna característica.  
Queda, por tanto, definido el primer modelo de Random Forest.

### 5.3.2 Implementación K-NN

Como se ha comentado a lo largo del documento, K-NN realiza la clasificación calculando la distancia del elemento a clasificar con el resto de los elementos, seleccionando los K más próximos y eligiendo como respuesta la clase dominante entre estos (votación de mayoría). A diferencia del Random Forest, el entrenamiento se realiza con todo el conjunto definido. La librería *Scikit-Learn* incluye el *KNeighborsClassifier* con los valores por defecto mostrados en la Figura 5.17.

```
class sklearn.neighbors.KNeighborsClassifier(n_neighbors=5, *, weights='uniform', algorithm='auto', leaf_size=30, p=2,
metric='minkowski', metric_params=None, n_jobs=None)
```

Figura 5.17: Hiperparámetros por defecto K-NN (Fuente: [88])

En este caso, el valor de K vecinos utilizado es el que tiene una mayor influencia en los resultados de la predicción. Aumentar K no implica necesariamente mejorar los resultados y sí que supone que modelo sea más lento. Por tanto, para elegir el valor óptimo para este caso de aplicación se opta por representar la precisión que ofrece cada valor K, variando este entre 1 y 20 (se considera un rango adecuado que ofrece un equilibrio entre la calidad de los resultados y el tiempo de cálculo requerido). Esta representación se muestra en la Figura 5.18.

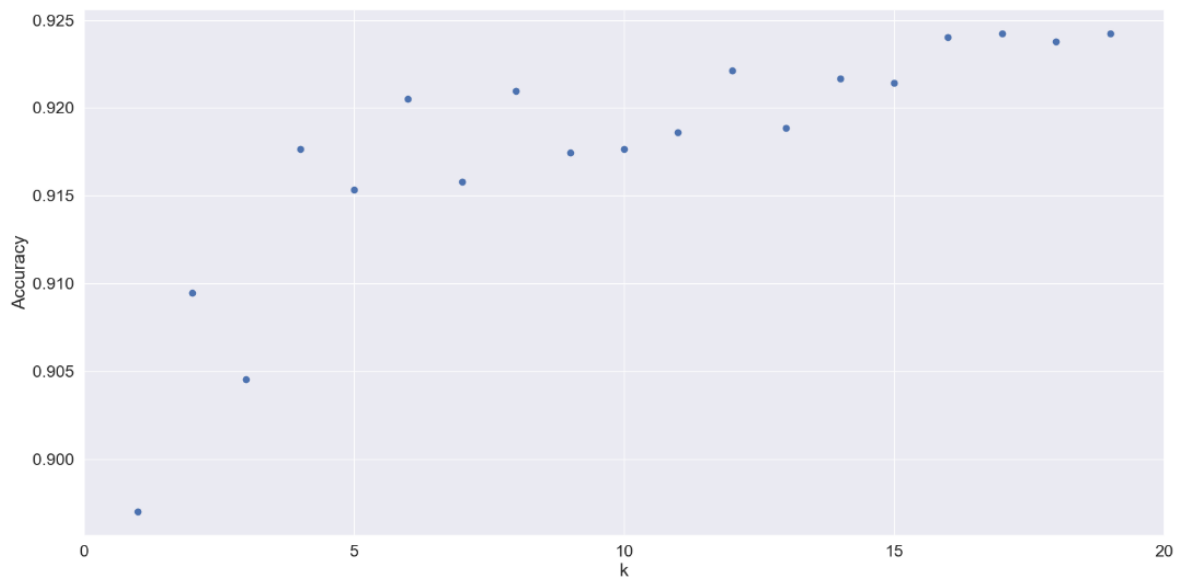


Figura 5.18: Accuracy VS valor de K (Fuente: elaboración propia)

Se elige un valor de  $K=17$ , ya que se puede apreciar que un valor superior no ofrece mejoras en los resultados.

Se define así el modelo de K-NN, dejando el resto de los valores por defecto.

### 5.3.3 Implementación SVM

La clasificación por máquinas de vectores de soporte se realiza mediante la generación del hiperplano óptimo que mejor separa las distintas clases, que será aquel que maximice el margen de separación entre las mismas. En la práctica, cuando los datos no son linealmente separables, esto se realiza utilizando un kernel que proyecta estos datos no separables linealmente en un espacio de menores dimensiones a un espacio con mayores dimensiones, de forma que los puntos de datos pertenecientes a diferentes clases se asignan a diferentes dimensiones, convirtiendo el problema en datos linealmente separables.

Para la implementación de este algoritmo, *Scikit-Learn* incorpora el *SVC* con los datos por defecto que se muestran en la Figura 5.19.

```
class sklearn.svm.SVC(*, C=1.0, kernel='rbf', degree=3, gamma='scale', coef0=0.0, shrinking=True, probability=False, tol=0.001,
cache_size=200, class_weight=None, verbose=False, max_iter=-1, decision_function_shape='ovr', break_ties=False,
random_state=None)
```

Figura 5.19: Hiperparámetros por defecto SVM (Fuente: [89])

Los tres valores que más influyen en el resultado de las predicciones son:

- **kernel:** especifica el tipo de kernel a usar, que puede ser lineal, polinómico, kernel Gaussiano de función de base radial (RBF – *Radial Based Function*) o Sigmoid. El lineal es el más básico y se utiliza cuando los datos son linealmente separables; el polinómico es una representación más generalizada del lineal; el rbf (que viene por defecto) es uno de los más preferidos y utilizados, y se suele elegir para datos no lineales o cuando no hay conocimiento previo de los mismos; por último, el Sigmoid es el preferido para redes neuronales.
- **C:** representa la función de coste. Controla el equilibrio entre un límite de decisión suave y la clasificación correcta de los puntos de entrenamiento. Aumentar su valor puede llevar a un sobreajuste de los datos de entrenamiento.
- **gamma:** es un parámetro para los hiperplanos no lineales. Cuanto más alto sea su valor, más se intentará ajustar exactamente el conjunto de datos de entrenamiento, por lo que aumentar su valor también puede conducir a un sobreajuste.

Tras probar distintas combinaciones, se construye el modelo de SVM con los valores definidos por defecto.

En el Anexo 1 se incluye el código desarrollado para la implementación descrita en este apartado.

## 5.4 ENTRENAMIENTO Y EVALUACIÓN DEL MODELO

Antes de realizar predicciones, es necesario entrenar los modelos. Para ello, tomando como base la bibliografía analizada como [32], [42] o [60], entre otros, se elige un 70% de los datos disponibles como datos de entrenamiento y el 30% restante se utiliza para el test.

Una vez entrenados, se realizan las predicciones sobre los datos de test para poder evaluar la performance de cada uno de ellos. Los resultados que devuelve el programa son los presentados en la Figura 5.20.

```
Exactitud (accuracy) Random Forest: 0.9275328330206379
Exactitud (accuracy) K-NN: 0.924249530956848
Exactitud (accuracy) SVM: 0.9036116322701688
Precisión Random Forest: 0.9188997320755495
Precisión K-NN: 0.9147774811932393
Precisión SVM: 0.8165139819739589
Valor F1 Random Forest: 0.9229774626829265
Valor F1 K-NN: 0.9191767371253071
Valor F1 SVM: 0.8578577354039573
Valor Recall Random Forest: 0.9275328330206379
Valor Recall K-NN: 0.924249530956848
Valor Recall SVM: 0.9036116322701688
```

Figura 5.20: Resultados evaluación de los modelos (Fuente: elaboración propia)

Las métricas empleadas para esta evaluación son las comúnmente empleadas en problemas de clasificación:

- Exactitud (*accuracy*): porcentaje de valores clasificados correctamente, según la ecuación 5.2.

$$Exactitud = \frac{TP + TN}{TP + TN + FP + FN} \quad (5.2)$$

Donde: TP es verdadero positivo, TN verdadero negativo, FP falso positivo y FN falso negativo.

- Precisión (*precision*): número de verdaderos positivos que son realmente positivos en comparación con el número total de valores positivos predichos (tanto verdaderos como falsos). Mide la precisión del clasificador a la hora de predecir casos positivos. Su cálculo sigue la ecuación 5.3.

$$Precision = \frac{TP}{TP + FP} \quad (5.3)$$

- *Recall*: cantidad de verdaderos positivos que el modelo ha clasificado en función del número total de valores positivos. Se calcula con la ecuación 5.4.

$$Recall = \frac{TP}{TP + FN} \quad (5.4)$$

- F1: combinación de las métricas de precisión y recall, según la ecuación 5.5.

$$F1 = 2 * \frac{precision * recall}{precision + recall} \quad (5.5)$$

La Tabla 5.1 recoge estos resultados de manera más ordenada y visual.

Tabla 5.1: Resultados evaluación de los modelos (*Fuente: elaboración propia*)

Algoritmo	Exactitud ( <i>accuracy</i> )	Precisión ( <i>precision</i> )	Recall	F1 score
Random Forest	92,75%	91,89%	92,75%	92,30%
K-NN	92,43%	91,48%	92,43%	91,92%
SVM	90,36%	81,65%	90,36%	85,79%

En vista a los resultados obtenidos, se pueden validar los modelos construidos. En el siguiente capítulo se analizan en mayor profundidad estos resultados.

En el Anexo 1 se incluye el código desarrollado para todo el caso práctico.

## 6. Resultados

El objetivo fijado para este trabajo fue analizar y comprobar las soluciones que puede aportar el aprendizaje automático en la gestión de los aeropuertos para mejorar su eficiencia y competitividad.

Por una parte, mediante la revisión sistemática de la literatura y el ejercicio de investigación de las técnicas que se están empleando en los aeropuertos del mundo y de las propuestas de los organismos internacionales más relevantes del sector, se ha encontrado que:

- Los esfuerzos de investigación y publicación de trabajos han tenido un salto importante en 2019 y se mantienen en esos altos niveles desde entonces.
- Los resultados obtenidos mediante la aplicación de técnicas de ML superan en la mayoría de los casos a los obtenidos con las técnicas tradicionales actuales.
- La implantación de sistemas basados en ML e inteligencia artificial son ya una realidad en la mayor parte de los aeropuertos del mundo, y su crecimiento va en aumento. De hecho, todos los actores involucrados confirman que el futuro de los aeropuertos pasa por la aplicación de estas nuevas tecnologías.
- El aprendizaje automático introduce ventajas en todos los subcampos de la gestión aeroportuaria: control de demoras y predicción de eventos disruptivos (como los meteorológicos), mejora de los procesos a los pasajeros en el edificio terminal, mejora del sistema de tratamiento de equipajes, aumento de la eficiencia en las operaciones, ayuda en mejorar la sostenibilidad, aumento de la seguridad y aumento de la eficiencia en la gestión de activos.

Por otra parte, con el caso de estudio se ha comprobado cómo varias técnicas pueden ayudar en la predicción de necesidades de puestos de control de seguridad, a partir de registros históricos de pasajeros por hora, día de la semana y temporada de operación. Estas técnicas evaluadas son Random Forest, K-NN y SVM. Los resultados de las métricas más relevantes para el problema (exactitud, precisión, *recall* y F1) concluyen que los tres métodos proporcionan unos resultados muy aceptables, con una exactitud (*accuracy*) superior al 90% en todo caso.

Entre ellos, el Random Forest y el K-NN resultan ser los mejores modelos, con unos resultados casi idénticos, ofreciendo una exactitud superior al 92% y el resto de las métricas por encima del 91%; mientras que el SVM está ligeramente por debajo en exactitud, con un 90,36%, y unos resultados algo más bajos en términos de precisión, pero manteniéndose por encima del 81%, lo que hace que el F1-score también baje, aunque está por encima del 85%.

Actualmente, una gran parte de los aeropuertos utilizan el sistema XOVIS para la gestión de medios en el control de seguridad. Este sistema está basado en sensores colocados en el área de colas de los controles que detectan cada persona que entra en ese área definida y cronometran su tiempo de permanencia en la misma, de tal manera que si se supera un determinado umbral preestablecido de tiempo de espera, el sistema emite alertas.

Un modelo como el propuesto en este TFM podría apoyar a este sistema, permitiendo una mayor anticipación en la predicción de necesidades de activación de puestos, es decir, se podría aplicar desde el punto de vista predictivo más que reactivo como es el sistema actual.

Por último, entre las publicaciones analizadas en este documento, se han encontrado varias que proponen la aplicación de ML en el control de seguridad. Por ejemplo, [59] aplica técnicas de aprendizaje automático basadas en redes neuronales LSTM para predecir los flujos de pasajeros en los controles de seguridad del Aeropuerto Charles De Gaulle, obteniendo predicciones con un  $R^2$  de hasta el 89%; y [61] utiliza Random Forest para predecir el comportamiento de las colas en el control de seguridad, alcanzando una exactitud de hasta el 74%. Ambos definen como variable a predecir los pasajeros y utilizan técnicas de regresión.

La principal diferencia del caso práctico implementado en este TFM es que la variable a predecir es directamente el número de controles necesarios, y se emplean técnicas de clasificación.

No obstante, el objetivo común tanto de los modelos propuestos en los estudios anteriores como en este trabajo es proporcionar una herramienta que haga más eficiente la asignación de medios en el control de seguridad, permitiendo reducir los costes de operación y proporcionar una mejor experiencia a los pasajeros.

---

## 7. Conclusiones y recomendaciones

En 2018, los pronósticos de IATA indicaban que el número de pasajeros aéreos se duplicaría para 2037. Con el impacto de la pandemia del COVID-19 y la caída sin precedentes que supuso para el transporte aéreo, la situación ahora es más incierta. No obstante, los principales organismos y compañías del sector pronostican una recuperación a los niveles prepandemia entre 2023 y 2025, para seguir a partir de ahí con un crecimiento CAGR del 3,9% hasta 2040.

Este fuerte crecimiento unido a los nuevos retos introducidos por la pandemia, la mayor preocupación y responsabilidad medioambiental, los cambios en las regulaciones y en los comportamientos y necesidades de los pasajeros, y el auge de la digitalización y la importancia del dato, hacen que los aeropuertos deban ser capaces de adaptarse para mantenerse eficientes y sostenibles a largo plazo.

En este contexto, tras el estudio llevado a cabo en este trabajo, la principal conclusión es que la aplicación de *machine learning* introduce importantes mejoras en este sentido para la gestión de la operación aeroportuaria en términos de:

- Aumento de la seguridad.
- Aumento de la eficiencia operativa.
- Aumento de la capacidad.
- Reducción de la huella medioambiental.
- Reducción de costes operativos.
- Mejora de la experiencia del pasajero.

Esto queda reafirmado con el aumento del número de publicaciones especializadas en este campo en los últimos años, y con los informes de todos los organismos del sector, que incluyen propuestas de soluciones aeroportuarias basadas en aprendizaje automático e inteligencia artificial. Todos ellos concluyen que el futuro próximo de los aeropuertos, como el de casi todos los sectores, pasa por la aplicación de estas tecnologías. Y, de hecho, las autoridades regulatorias están avanzando en un marco de aplicación para las mismas. Uno de los aspectos que todas las partes interesadas dentro del sistema aeroportuario deben mejorar para el éxito de esta implantación, es la coordinación y colaboración entre



ellas en la transferencia de datos para ofrecer soluciones aplicables horizontalmente a todo el proceso, ya que aunque cada una tenga sus propios objetivos e intereses, la mejora global implicará la mejora de todas ellas y los beneficios serán mutuos.

Como caso de estudio para confirmar lo anterior, se ha incluido un modelo de predicción de necesidades de activación de puestos de control de seguridad mediante técnicas de Random Forest, K-NN y SVM, obteniendo los mejores resultados con las dos primeras, con una exactitud por encima del 92% y una precisión superior al 91%.

Esto confirma el potencial de aplicación de estos métodos en la práctica, ya que proporcionan buenos resultados incluso con los datos empleados, que no son los más completos de los que puede disponer un gestor aeroportuario.

Por ejemplo, con unos datos más completos que incluyan el tipo de vuelo (nacional, internacional) o el tipo billete del pasajero (turista, negocios, familiar, etc.) se puede ajustar el cálculo de la hora por la que pasan por el control, haciéndolo más representativo y ajustado al comportamiento real. Además, con los datos de los informes de medición de los tiempos de proceso que se realizan en los aeropuertos, se podría introducir un valor concreto propio de la realidad del aeropuerto, lo que también ajustaría más los resultados a la operativa existente.

Otra mejora sería disponer además de datos de la hora más completos, es decir, incluyendo también los minutos. Con esto se puede aplicar la metodología IATA completa: realizar los cálculos por intervalos de 15, 30, 45, o 60 min y elegir el más demandante.

Por otra parte, se podrían implementar otros algoritmos más complejos para ver si se obtiene una mejora sustancial en los resultados de la predicción o si, por el contrario, aumentar la complejidad del modelo no conlleva a unas predicciones muy superiores que lo compensen.

Aun con todas estas posibles mejoras que se podrían introducir como evolución de este caso de estudio en un futuro, se puede concluir de todo el análisis realizado en el presente trabajo que la aplicación de *machine learning* a la gestión aeroportuaria es una realidad que introduce enormes beneficios para la eficiencia y sostenibilidad de las infraestructuras aeroportuarias.

---

# Bibliografía

- [1] Airbus, “Airbus Global Market Forecast 2021-2040”.
- [2] IATA Economics, “COVID-19 State of the aviation industry”, 9<sup>th</sup> November 2021.
- [3] Nigel Halpern, Pere Suau-Sanchez, Deodat Mwesiumo, Thomas Budd, and Svein Brathen, “Ready for digital transformation? The effect of organisational readiness, innovation, airport size and ownership on digital change at airports”, *Journal of Air Transport Management* 90 (2021) 101949, January 2021.
- [4] Sean O’Neill and Brian Summers, Skift (November 20<sup>th</sup>, 2019), “The Rise of Smart Airports: A Skift Deep Dive” [Online]. Disponible en: <https://skift.com/2019/11/20/>
- [5] Marisa García (February 23, 2021), “SITA Report Charts Increased Airline Investment in Touchless Tech and Remote Work” [Online]. Disponible en: <https://apex.aero/articles/2021-02-23-sita-report-charts-touchless-tech-remote-work-investments/>
- [6] Judith Hurwitz and Daniel Kirsch, “Machine Learning For Dummies”, IBM Limited Edition, published by John Wiley & Sons, Inc., ISBN: 978-1-119-45495-3 (pbk); ISBN: 978-1-119-45494-6 (ebk), 2018.
- [7] Blog, Machine Learning o Aprendizaje Automático, “Regresión lineal logística - Teoría” [Online]. Disponible en: <https://aprendeia.com/regresion-logistica-multiple-machine-learning-teoria/>
- [8] Oswaldo Simeone, Fellow, IEEE, “A Very Brief Introduction to Machine Learning With Applications to Communication Systems”, in *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 4, pp. 648-664, Dec. 2018.
- [9] “Decision Tree Classification Algorithm” [Online]. Disponible en: <https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm>
- [10] Víctor Roman (27 de febrero de 2019), “Machine Learning Supervisado: Fundamentos de la Regresión Lineal” [Online]. Disponible en: <https://medium.com/datos-y-ciencia>

- 
- [11] Daniel Rodríguez (20 de mayo de 2019), “Random Forest” [Online]. Disponible en: <https://www.analyticslane.com/2019/05/20/random-forest/>
- [12] MathWorks “Support Vector Machine (SVM) – Hiperplanos óptimos como límites de decisión” [Online]. Disponible en: <https://es.mathworks.com/discovery.html>
- [13] Fernando Sancho Caparrimi (26 de diciembre de 2018), “Métodos combinados de aprendizaje” [Online]. Disponible en: <http://www.cs.us.es/~fsancho/?e=106>
- [14] Miguel Cárdenas-Montes, “Minería de datos - Bagging” [Online]. Disponible en: <http://wwwae.ciemat.es/~cardenas/docs/lessons/Bagging.pdf>
- [15] Miguel Cárdenas-Montes, “Minería de datos – Boosting” [Online]. Disponible en: <http://wwwae.ciemat.es/~cardenas/docs/lessons/Boosting.pdf>
- [16] Pranshu Sharma (April 26, 2021), “K Means Clustering Simplified in Python” [Online]. Disponible en: <https://www.analyticsvidhya.com/blog/2021/04/k-means-clustering-simplified-in-python/>
- [17] Ligdi Gonzalez (8 de septiembre de 2020), “Algoritmo Agrupamiento Jerárquico – Teoría” [Online]. Disponible en: <https://aprendeia.com/algoritmo-agrupamiento-jerarquico-teoria/>
- [18] MathWorks Help Center Documentation, “Fuzzy C-Means Clustering–Plot Clustering Results” [Online]. Disponible en: <https://es.mathworks.com/help/fuzzy/fuzzy-c-means-clustering.html>
- [19] Santiago de la Fuente Fernández, “Análisis de Componentes Principales”, apuntes de Instrumentos Estadísticos Avanzados, Departamento de Economía Aplicada, Facultad de Ciencias Económicas y Empresariales, Universidad Autónoma de Madrid [Online]. Disponible en: <https://www.estadistica.net/>
- [20] Luis Serrano (26 de febrero de 2019), “Análisis de componentes principales (PCA)” [Online]. Disponible en: <https://www.youtube.com/watch?v=AniiwysJ-2Y>
- [21] Chieh-Feng Cheng, inwinSTACK (30 de noviembre de 2020), “Reinforcement Learning” [Online]. Disponible en: <https://www.inwinstack.com/blog-en/blog-ai-en/6262/>

- 
- [22] ATRIA Innovation (22 de octubre de 2019), Industria 4.0, Inteligencia Artificial, “Qué son las redes neuronales y sus funciones” [Online]. Disponible en: <https://www.atriainnovation.com/que-son-las-redes-neuronales-y-sus-funciones/>
- [23] Diego Calvo (13 de julio de 2017), Aprendizaje automático, Redes neuronales, “Clasificación de redes neuronales artificiales” [Online]. Disponible en: <https://www.diegocalvo.es/clasificacion-de-redes-neuronales-artificiales/>
- [24] Luis Octavio González Salcedo, Aydee Patricia Guerrero Zúñiga, Silvio Delvasto Arjona, and Adrián Luis Ernesto Will, “Artificial Neural Model based on radial basis function networks used for prediction of compressive strength of fiber-reinforced concrete mixes”, *Ciencia e Ingeniería Neogranadina*, Vol. 29 (2), pp. 37-53, julio-diciembre 2019.
- [25] Francisco Alonso, “Neural Networks and Deep Learning. Chapter 1: Prelude” [Online]. Disponible en: <https://www.futurespace.es/en/redes-neuronales-y-deep-learning-capitulo-1-preludio/>
- [26] Alicia García Hidalgo (5 de noviembre de 2018), Grupo de Investigación GRIAL, Instituto Universitario de Ciencias de la Educación, Universidad de Salamanca, “Técnicas para llevar a cabo mapeos y revisiones sistemáticas de la literatura” [Online]. Disponible en: <https://www.slideshare.net/grialusal/tcnicas-para-llevar-a-cabo-mapeos-y-revisiones-sistematicas-de-la-literatura>
- [27] Miguel Ángel Conde González y Francisco Jesús Rodríguez Sedano, Grupo de robótica, Universidad de León, “Elaboración de una revisión sistemática de la literatura”.
- [28] Francisco J. Rodríguez Sedano, “Parsifal. Uso de herramienta on-line para la elaboración de una revisión sistemática de la literatura (SLR). Manual de usuario”, Marzo, 2019.
- [29] W. Zeng, J. Li, Z. Quan, and X. Lu, “A Deep Graph-Embedded LSTM Neural Network Approach for Airport Delay Prediction”, *Journal of Advanced Transportation*, no. 15, March 2021.

- [30] B. Ye, B. Liu, Y. Tian, and L. Wan, "A Methodology for Predicting Aggregate Flight Departure Delays in Airports Based on Supervised Learning", *Sustainability*, no. 12, April 2020.
- [31] Sina Khanmohammadi, Salih Tunun, and Yunus Kucuk, "A New Multilevel Input Layer Artificial Neural Network for Predicting Flight Delays at JFK Airport", *Procedia Computer Science* 95 (2016) 237-244.
- [32] Abdulwahab Aljubairy, Wei Emma Zhang, Ali Shemshadi, Adnan Mahmood, and Quan Z. Sheng, "A system for effectively predicting flight delays based on IoT data", *Computing* (2020) 102: 2025-2048.
- [33] Abdulwahab Aljubairy, Ali Shemshadi, and Quan Z. Sheng, "Real-Time Investigation of Flight Delays Based on the Internet of Things Data", *ADMA 2016, LNAI 10086*, pp 788-800, 2016.
- [34] Juan Jose Rebollo and Hamsa Balakrishnan, Massachusetts Institute of Technology, "Characterization and prediction of air traffic delays", *Transportation Research Part C* 44 (2014) 231-241.
- [35] Noriko Etani, "Development of a predictive model for on-time arrival flight of airliner by discovering correlation between flight and weather data", *Journal of Big Data* (2019) 6:85.
- [36] Maryam Farshchian Yazdi, Seyed Reza Kamel, Seyyed Javad Mahdavi Chabok, and Maryam Kheirabadi, "Flight delay prediction based on deep learning and Levenberg-Marquart algorithm", *Journal of Big Data* (2020) 7:106.
- [37] Wei Shao, Siyu Tan, Xinhong Hei, Arian Prabowo, Piort Koniusz, Bradley Feest, Sichen Zhao, Jeffrey Chan, and Flora D. Salim, "Flight Delay Prediction using Airport Situational Awareness Map", in *27<sup>th</sup> ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL '19)*, November 5-8, 2019, Chicago, IL, USA.
- [38] Jingyi Qu, Ting Zhao, Meng Ye, Jiayi Li, and Chao Liu, "Flight Delay Prediction Using Deep Convolutional Neural Network Based on Fusion of Meteorological Data", *Neural Processing Letters* (2020) 52: 1461-1484.

- 
- [39] Michael Bloem and Nicholas Bambos, “Ground Delay Program Analytics with Behavioral Cloning and Inverse Reinforcement Learning”, *Journal of Aerospace Information Systems*, Volume 12, Number 3, March 2015.
- [40] Avijit Mukherjee, Shon Grabbe, and Banavar Sridhar, “Predicting Ground Delay Program At An Airport Based on Meteorological Conditions”, in 14<sup>th</sup> AIAA Aviation Technology, Integration, and Operations Conference, 16-20 June 2014, Atlanta, GA, USA.
- [41] Chunzheng Wang, Minghua Hu, Lei Yang, and Zheng Zhao, “Prediction of air traffic delays: An agent-based model introducing refined parameter estimation methods”, *PLoS ONE* 16(4): e0249754, April 7, 2021.
- [42] Roberto Henriques and Inês Feiteira, Universidade Nova de Lisboa, “Predictive Modelling: Flight Delays and Associated Factors, Hartsfield – Jackson Atlanta International Airport”, *Procedia Computer Science* 138 (2018) 638-645.
- [43] Pavel Stefanovic, Rokas Strimaitis, and Olga Kurasova, “Prediction of Flight Time Deviation for Lithuanian Airports Using Supervised Machine Learning Model”, *Computational Intelligence and Neuroscience*, Volume 2020, Article ID 8878681, 10 pages, October 2020.
- [44] Grabbe S., Srdhar B., and Mukherjee A., “Clustering days with similar airport weather conditions”, *AIAA AVIATION 2014 – 14<sup>th</sup> AIAA Aviation Technology, Integration, and Operations Conference*, 16 June – 20 June 2014, Atlanta, GA, USA.
- [45] Yao Wang and Deepak Kulkarni, NASA Ames Research Center, “Modeling Weather Impact on Ground Delay Programs”, *SAE International Journal of Aerospace* 4 (2): 1207-1215, June 2011.
- [46] Yao Wang, NASA Ames Research Center, “Prediction of weather impacted airport capacity using ensemble learning”, 2011 *IEEE/AIAA 30th Digital Avionics Systems Conference*, 2011, pp. 2D6-1-2D6-11.
- [47] Ristiana Dewi, Prawito, and Hastuadi Harsa, “Fog prediction using artificial intelligence: A case study in Wamena Airport”, *Journal of Physics: Conference Series*, Volume 1528, 4<sup>th</sup> International Seminar on Sensors, Instrumentation, Measurement and Metrology, 14<sup>th</sup> November 2019, Padang, Indonesia.

- 
- [48] L. Cornejo-Bueno, C. Casanova-Mateo, J. Sanz-Justo, E. Cerro-Prada, and S. Salcedo-Sanz, "Efficient Prediction of Low-Visibility Events at Airports Using Machine-Learning Regression", *Boundary-Layer Meteorol* (2017) 165: 349-370.
- [49] Sara Cornejo-Bueno, David Casillas-Pérez, Laura Cornejo-Bueno, Mihaela I. Chidean, Antonio J. Caamaño, Julia Sanz-Justo, Carlos Casanova-Mateo, and Sancho Salcedo-Sanz, "Persistence Analysis and Prediction of Low-Visibility Events at Valladolid Airport, Spain", *Symmetry* 2020, 12, 1045.
- [50] Tuo Deng, Aijie Cheng, Wei Han, and Hai-Xiang Lin, "Visibility Forecast for airport operations by LSTM Neural Networks", in 11<sup>th</sup> International Conference on Agents and Artificial Intelligence (ICAART 2019), Prague, Czech Republic, Volume 2, pp. 466-473.
- [51] Jianan Yin, Yuxin Hu, Yuanyuan Ma, Yan Xu, Ke Han, and Dan Chen, "Machine Learning Techniques for Taxi-out Time Prediction with a Macroscopic Network Topology", *AIAA/ IEEE 37<sup>th</sup> Digital Avionics Systems Conference*, London, September 2018.
- [52] Guan Lian, Yaping Zhang, Jitamitra Desai, Zhiwei Xing, and Xiao Luo, "Predicting Taxi-Out Time at Congested Airports with Optimization-Based Support Vector Regression Methods", *Hindawi, Mathematical Problems in Engineering*, Volume 2018, Article ID 7509508, 11 pages.
- [53] Hanbong Lee, Waqar Malik, Bo Zhang, Balaji Nagarajan, and Yoon C. Jung, "Taxi Time Prediction at Charlotte Airport Using Fast-Time Simulation and Machine Learning Techniques", 15<sup>th</sup> AIAA Aviation Technology, Integration, and Operations Conference, 22-26 June 2015, Dallas, TX, USA.
- [54] Hanbong Lee and Waqar Malik, University of California, "Taxi-Out Time Prediction for Departures at Charlotte Airport Using Machine Learning Techniques", *AIAA Aviation*, 16<sup>th</sup> AIAA Aviation Technology, Integration, and Operations Conference, 13-17 June 2016, Washington D.C., USA.
- [55] Yunfan Zhou, Weilong Liu, Yandong Li, and Bo Jiang, Civil Aviation Flight University of China, "Taxiing Speed Intelligent Management of Aircraft Based on DQN for A-SMGCS", *Journal of Physics: Conference Series*, CISAT 2019, 1345 (2019) 042015.

- 
- [56] Stefan Ravizza, Jun Chen, Jason A.D. Atkin, Paul Stewart, and Edmund K. Burke, "Aircraft taxi time prediction: Comparisons and insights", *Applied Soft Computing* 14 (2014) 397-406. 2014.
- [57] Kilian Ros, Elena Mocanu, and Christin Seifert, "Airport Restroom Cleanliness Prediction Using Real Time User Feedback Data", *IEEE 5<sup>th</sup> International Conference on Collaboration and Internet Computing (CIC)*, 2019.
- [58] Rolf Felkel, Dieter Steinmann, and Frank Follert, "Hub Airport 4.0 – How Frankfurt Airport Uses Predictive Analytics to Enhance Customer Experience and Drive Operational Excellence", in: Linnhoff-Popien C., Schneider R., Zaddach M. (eds) *Digital Marketplaces Unleashed*. Springer, Berlin, Heidelberg. 2018.
- [59] Philippe Monmousseau, Gabriel Jarry, Florian Bertosio, Daniel Delahaye, and Marc Houalla, ENAC Université de Toulouse, "Predicting Passenger Flow at Charles De Gaulle Airport Security Checkpoints", *2020 International Conference on Artificial Intelligence and Data Analytics for Air Transportation (AIDA-AT)*, 2020, pp. 1-9.
- [60] Federico Orsini, Massimiliano Gastaldi, Luca Mantecchini, and Riccardo Rossi, "Neural Networks trained with WiFi traces to predict airport passenger behavior", *2019 6th International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, 2019, pp. 1-7.
- [61] Álvaro Rodríguez-Sanz, Alberto Fernández de Marcos, Javier A. Pérez-Castán, Fernando Gómez Comendador, Rosa Arnaldo Valdés, and Ángel Paris Loreiro, Universidad Politécnica de Madrid, "Queue behavioural patterns for passengers at airport terminals: A machine learning approach", *Journal of Air Transport Management* 90 (2021) 101940.
- [62] Weishi Chen, Yifeng Huang, Hao Yang, Jing Li, and Xianfeng Lu, "A passenger risk assessment method based on 5G-IoT", *EURASIP Journal on Wireless Communications and Networking* (2021) 2021:5.
- [63] Cheong Kim, Francis Joseph Costello, and Kun Chang Lee, "Integrating Qualitative Comparative Analysis and Support Vector Machine Methods to Reduce Passengers' Resistance to Biometric E-Gates for Sustainable Airport Operations", *Sustainability* 2019, 11, 5349.



- 
- [64] Quang Hung Do, Shih-Kuei Lo, Jeng-Fung Chen, Chi-Luan Le, and Luong Hoang Anh, “Forecasting Air Passenger Demand: A Comparison of LSTM and SARIMA”, *Journal of Computer Science* 2020, 16 (7): 1063.1084.
- [65] Xinfang Wu, Yong Xiang, Gang Mao, Mingqian Du, Xiuqing Yang, and Xinzhi Zhou, “Forecasting air passenger traffic flow based on the two-phase learning model”, *The Journal of Supercomputing* (2021) 77: 4221-4243.
- [66] G. Stempf, I. De Visscher, M. Ellejmi, V. Brossard, A. Bonnefoy, and V. Treve, “Applying Machine Learning Modeling to Enhance Runway Throughput at A Big European Airport”, *IOP Conference Series: Materials Science and Engineering*, Volume 1024, 10<sup>th</sup> EASN International Conference on Innovation in Aviation and Space to the Satisfaction of the European Citizens (10<sup>th</sup> EASN 2020), 2<sup>nd</sup> – 5<sup>th</sup> September 2020.
- [67] Jiangjun Tang and Hussein A. Abbass, School of Engineering and Information Technology – Canberra, Australia, “Behavioral Learning of Aircraft Landing Sequencing Using a Society of Probabilistic Finite State Machines”, *IEEE Congress on Evolutionary Computation (CEC)*, July 6-11, 2014, Beijing, China.
- [68] Qian Luo, Yanru Chen, Lunyue Chen, Xiao Luo, Huan Xia, Yuanyuan Zhang, and Liangyin Chen, “Research on Situational Awareness of Airport Operation Based on Petri Nets”, in *IEEE Access*, Vol. 7, pp. 25438-25451, 2019.
- [69] Björn Rafn Gunnarsson, Seppe K. L. M. vander Broucke, and Jochen De Weerd, Research Centre for Information Systems Engineering (LIRIS), Belgium, “Predictive Process Monitoring in Operational Logistics: A Case Study in Aviation”, in: Di Francescomarino C., Dijkman R., Zdun U. (eds) *Business Process Management Workshops. BPM 2019. Lecture Notes in Business Information Processing*, vol 362. Springer, Cham.
- [70] Qian Wang, Khalid N. Ismail, and Toby P. Breckon, “An Approach for Adaptive Threat Recognition Within 3D Computed Tomography Images for Baggage Security Screening”, *Computer Science, Medicine, Journal of X-ray science and technology*, March 2019.

- 
- [71] Grzegorz Szwoch, “Extraction of stable foreground image regions for unattended luggage detection”, *Multimed Tools Appl* (2016) 75: 761 – 786.
- [72] Chen H., Deng J., Sun B., and Wang J., “An Airport Noise Prediction Model Based on Selective Ensemble of LOF-FSVR”, in: Huang Z., Sun X., Luo J., Wang J. (eds) *Cloud Computing and Security. ICCCS 2015. Lecture Notes in Computer Science*, vol 9483.
- [73] C. Asensio, M. Ruíz, M. Recuero, G. Moschioni, and M. Tarabini, “Application of intelligent instruments for the monitoring of thrust reverse noise at airports”, 19<sup>th</sup> IMEKO TC 4 Symposium and 17<sup>th</sup> IWADC Workshop, *Advances in Instrumentation and Sensors Interoperability*, July 18-19, 2013, Barcelona, Spain.
- [74] Huang Hailong, and Zhu Jinfu, Nanjing University of Aeronautics and Astronautics, “A Short Review of the Application of Machine Learning Methods in Smart Airports”, *Journal of Physics: Conference Series*, 1769 (2021) 012010, ICCSIE 2020.
- [75] Mark Azzam, German Aerospace Center, “The evolution of airports from a network perspective – an analytical concept”, *Chinese Journal of Aeronautics* 28 (2015).
- [76] Salvatore Antonio Biancardo, Francesco Abbodanti, Francesca Russo, Rosa Varopalumbo, and Gianluca Dell’ Acqua, “A Broad-Based Decision-Making Procedure for Runway Friction Decay Analysis in Maintenance Operations”, *Sustainability* 2020, 12, 3516.
- [77] CROWDVISION, “Solutions for airports – What we offer” [Online]. Disponible en: <https://www.crowdvision.com/solutions-airports/>
- [78] SMART FLOWS, “Solution” [Online]. Disponible en: <https://www.smart-flows.com/solution>
- [79] SITA, “SITA Passenger Flow Management” [Online]. Disponible en: <https://www.sita.aero/solutions/sita-at-airports/sita-passenger-processing/sita-passenger-flow-management/>
- [80] IATA, Programs, “OneID – It’s taking off!” infographics [Online]. Disponible en: <https://www.iata.org/en/programs/passenger/one-id/#tab-5>
- [81] Aena, Notas de prensa (23 de enero de 2019), “Aena apuesta por la experiencia del cliente y el Smart Airport en Fitur 2019” [Online]. Disponible en:

<https://portal.aena.es/es/corporativa/aena-apuesta-experiencia-cliente-y-smart-airport-en-fitur-2019.html?p=1237548067436>

- [82] Qi Qi, and Zhu Pan, “Internet of Things, Internet, Big Data and Airport Services Make Smart Airport Based on O2O and Humanism”, Advances in Engineering Research, volume 149, 2<sup>nd</sup> International Conference on Mechanical, Electronic, Control and Automation Engineering (MECAE 2018), 2018.
- [83] ENISA, “Securing Smart Airports”, December 2016.
- [84] Vijay Narula, OST, “Smart Airports” [Online]. Disponible en: <https://www.icao.int/safety/iStars/Documents/IUG%20Meeting%201/Presentations/Smart%20Airports%20-%20Vijay%20Narula.pdf>
- [85] IATA, “Airport Development Reference Manual”, Edition 11, 2019.
- [86] Scikit Learn, “sklearn.ensemble.RandomForestClassifier” [Online]. Disponible en: <https://scikitlearn.org>
- [87] Jose Martinez Heras (18 de septiembre de 2019), “Random Forest (Bosque Aleatorio): combinando árboles” [Online]. Disponible en: <https://www.iartificial.net/random-forest-bosque-aleatorio/>
- [88] Scikit Learn, “sklearn.neighbors.KNeighborsClassifier” [Online]. Disponible en: <https://scikitlearn.org>
- [89] Scikit Learn, “sklearn.svm.SVC” [Online]. Disponible en: <https://scikitlearn.org>

# Anexo 1: Código desarrollado

```
"""
Caso práctico: Predicción número de controles de seguridad
"""

#### Se importan las librerías necesarias

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import accuracy_score, f1_score, recall_score,
precision_score

#### Se importa el conjunto de datos

datos=pd.read_excel(r'C:\Users\Sandra\Documents\MIA\02 Segundo\TFM\10
Caso practico_clasificacion\Datos_modelo.xlsx')

#### Se comprueban anomalías y datos faltantes

print('Los datos nulos en cada columna
son:',datos.isnull().sum(),sep='\n')
print('Los registros duplicados son: ',datos.duplicated().sum())

#### Se ajustan los datos de entrada

datos.info()
pd.options.display.max_columns=None
print(datos.head(5))

# Se observan relaciones de variables
sns.set(font_scale=1.5)
fig, Fig1=plt.subplots()
Fig1=sns.barplot(x=datos['Mes Real'], y=datos['Pasajeros Total'],
hue=datos['Año Real'], estimator=sum,
order=['Enero', 'Febrero', 'Marzo', 'Abril', 'Mayo', 'Junio', 'Julio', 'Agosto',
'Septiembre', 'Octubre', 'Noviembre', 'Diciembre'], palette='mako')
plt.xlabel('Mes')
plt.ylabel('Pasajeros totales')
plt.title('Pasajeros totales por mes')

fig, Fig2=plt.subplots()
Fig2=sns.barplot(x=datos['Mes Real'], y=datos['Pasajeros Total'],
hue=datos['Año Real'],
order=['Enero', 'Febrero', 'Marzo', 'Abril', 'Mayo', 'Junio', 'Julio', 'Agosto',
'Septiembre', 'Octubre', 'Noviembre', 'Diciembre'], palette='mako')
plt.xlabel('Mes')
plt.ylabel('Media de pasajeros/hora')
```

```
plt.title('Media de pasajeros/hora por mes')

fig, Fig3=plt.subplots()
Fig3=sns.barplot(x=datos['Mes Real'], y=datos['Pasajeros Total'],
hue=datos['Año Real'], estimator=max,
order=['Enero', 'Febrero', 'Marzo', 'Abril', 'Mayo', 'Junio', 'Julio', 'Agosto',
'Septiembre', 'Octubre', 'Noviembre', 'Diciembre'], palette='mako')
plt.xlabel('Mes')
plt.ylabel('Pasajeros/hora máximos')
plt.title('Pasajeros/hora máximos por mes')

fig, Fig4=plt.subplots()
Fig4=sns.barplot(x=datos['Dia de la Semana Real'], y=datos['Pasajeros
Total'], hue=datos['Temporada'], estimator=sum,
order=['Lunes', 'Martes', 'Miercoles', 'Jueves', 'Viernes', 'Sabado', 'Domingo'
], palette='mako')
plt.xlabel('Dia de la semana')
plt.ylabel('Pasajeros totales')
plt.title('Pasajeros totales por día de la semana')

fig, Fig5=plt.subplots()
Fig5=sns.barplot(x=datos['Dia de la Semana Real'], y=datos['Pasajeros
Total'], hue=datos['Temporada'],
order=['Lunes', 'Martes', 'Miercoles', 'Jueves', 'Viernes', 'Sabado', 'Domingo'
], palette='mako')
plt.xlabel('Dia de la semana')
plt.ylabel('Media de pasajeros/hora')
plt.title('Media de pasajeros/hora por día de la semana')

fig, Fig6=plt.subplots()
Fig6=sns.barplot(x=datos['Dia de la Semana Real'], y=datos['Pasajeros
Total'], hue=datos['Temporada'], estimator=max,
order=['Lunes', 'Martes', 'Miercoles', 'Jueves', 'Viernes', 'Sabado', 'Domingo'
], palette='mako')
plt.xlabel('Dia de la semana')
plt.ylabel('Pasajeros/hora máximos')
plt.title('Pasajeros/hora máximos por día de la semana')

fig, Fig7=plt.subplots()
Fig7=sns.barplot(x=datos['Hora Operacion Real'], y=datos['Pasajeros
Total'], palette='mako')
plt.xlabel('Hora')
plt.ylabel('Pasajeros')
plt.title('Media de pasajeros/hora')

fig, Fig8=plt.subplots()
Fig8=sns.barplot(x=datos['Hora Operacion Real'], y=datos['Pasajeros
Total'], estimator=max, palette='mako')
plt.xlabel('Hora')
plt.ylabel('Pasajeros')
plt.title('Pasajeros/hora máximos')

# Se eliminan las columnas con información redundante
datos=datos.drop(['Mes', 'Temporada Real'], axis=1)

# Se eliminan los datos de mes real, porque se va a trabajar por
temporada
datos=datos.drop(['Mes Real'], axis=1)
```

```
# Se eliminan los datos correspondientes a "llegadas"
datos=datos[datos.Movimiento !="LLEGADA"]
datos=datos.reset_index()
datos=datos.drop(['index'], axis=1)

# Hecho esto, se elimina la columna de Movimiento
datos=datos.drop(['Movimiento'], axis=1)

# Se convierten las variables categoricas a numericas
dummies=pd.get_dummies(datos['Dia de la Semana Real'], drop_first=True)
dummies2=pd.get_dummies(datos['Temporada'], drop_first=True)
datos=pd.concat([datos, dummies, dummies2], axis=1)
datos=datos.drop(['Dia de la Semana Real','Temporada'], axis=1)

# Se convierte la hora de operacion en hora de paso por el control de
seguridad
hora_control=datos['Hora Operacion Real']-2
datos=pd.concat([datos, hora_control.rename('hora_control')], axis=1)
print(datos.head(10))
datos.info()

#### Se genera el modelo

# Primero se convierten los pasajeros en número de controles
demand=datos['Pasajeros Total']
PT=15 #sec
delta_t=60 #min
MQT=10 #min
SS=(demand*PT/60)/(delta_t+MQT)
SS=SS.replace(0,1)
controles=SS.apply(np.ceil)
datos=pd.concat([datos, controles.rename('Controles')],axis=1)
print(datos.head(10))
datos.info()

# Se definen las variables dependientes e independientes
x=datos[['hora_control','Jueves','Lunes','Martes','Miercoles','Sabado','Viernes','S17','S18','W15','W16','W17']]
y=datos[['Controles']]

# Se diferencian datos de entrenamiento y datos de test
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.3)

# Se implementa el algoritmo de Random Forest

#Con el gridsearch se seleccionan mejores hiperparametros
RF=RandomForestClassifier()
parameters={'max_depth':[70, 80, 90, 100], 'n_estimators':[100, 200, 300, 900, 1000, 1100]}
CV_RF=GridSearchCV(RF, parameters, cv=3)
CV_RF.fit(x_train, y_train.values.ravel())
print(CV_RF.best_params_)

RandomForest=RandomForestClassifier(n_estimators=1000, max_depth=70)
RandomForest.fit(x_train, y_train.values.ravel())
y_pred_RF=RandomForest.predict(x_test)
```

```
#Se analiza la importancia de las características
feature_imp=pd.Series(RandomForest.feature_importances_,
index=x.columns).sort_values(ascending=False)
print(feature_imp)
sns.barplot(x=feature_imp, y=feature_imp.index, palette='mako')
plt.xlabel('Valor de la importancia de las características')
plt.ylabel('Características')
plt.title('Importancia de las características')
plt.legend()
plt.show()

# Se implementa el algoritmo de K-NN
#Se busca el mejor valor de K
k_range=range(1, 20)
scores=[]
for k in k_range:
    knn=KNeighborsClassifier(n_neighbors=k)
    knn.fit(x_train, y_train.values.ravel())
    scores.append(knn.score(x_test, y_test))
plt.figure()
plt.xlabel('k')
plt.ylabel('Accuracy')
plt.scatter(k_range, scores)
plt.xticks([0,5,10,15,20])

k_vecinos=17
knn=KNeighborsClassifier(k_vecinos)
knn.fit(x_train, y_train.values.ravel())
y_pred_KNN=knn.predict(x_test)

# Se implementa el algoritmo de SVM
SVmodel=SVC()
SVmodel.fit(x_train, y_train.values.ravel())
y_pred_SVM=SVmodel.predict(x_test)

#### Se evalúan los resultados

print("Exactitud (accuracy) Random Forest: ", accuracy_score(y_test,
y_pred_RF))
print("Exactitud (accuracy) K-NN: ", accuracy_score(y_test, y_pred_KNN))
print("Exactitud (accuracy) SVM: ", accuracy_score(y_test, y_pred_SVM))
print("Precisión Random Forest: ", precision_score(y_test, y_pred_RF,
average='weighted'))
print("Precisión K-NN: ", precision_score(y_test, y_pred_KNN,
average='weighted'))
print("Precisión SVM: ", precision_score(y_test, y_pred_SVM,
average='weighted'))
print("Valor F1 Random Forest: ", f1_score(y_test, y_pred_RF,
average='weighted'))
print("Valor F1 K-NN: ", f1_score(y_test, y_pred_KNN,
average='weighted'))
print("Valor F1 SVM: ", f1_score(y_test, y_pred_SVM, average='weighted'))
print("Valor Recall Random Forest: ", recall_score(y_test, y_pred_RF,
average='weighted'))
print("Valor Recall K-NN: ", recall_score(y_test, y_pred_KNN,
average='weighted'))
print("Valor Recall SVM: ", recall_score(y_test, y_pred_SVM,
average='weighted'))
```