

# Análisis avanzado y predicción de series temporales aplicados al estudio de retrasos en aeropuertos.

TRABAJO FIN DE MÁSTER

Curso 2024/2025



MÁSTER EN INGENIERÍA DE SISTEMAS DE DECISIÓN

Amaia Sainz Villa

Tutora: Ana Elizabeth García Sipols

Madrid, junio 2025

# Índice de contenidos

<b>1. Introducción</b>	<b>8</b>
1.1. Motivación . . . . .	8
1.2. Objetivos . . . . .	9
1.3. Plan de trabajo . . . . .	9
1.4. Material de trabajo . . . . .	10
<b>2. Recopilación, limpieza y estructuración de los datos</b>	<b>11</b>
2.1. Origen de datos . . . . .	11
2.2. Obtención de series temporales . . . . .	12
<b>3. Técnicas de análisis de conglomerados</b>	<b>15</b>
3.1. Cálculo de distancias . . . . .	15
3.2. Creación de grupos . . . . .	17
3.3. Elección de representantes . . . . .	21
<b>4. Técnicas de predicción</b>	<b>24</b>
4.1. Introducción a la predicción de series temporales . . . . .	24
4.1.1. Enfoque general y modelos considerados . . . . .	24
4.1.2. Construcción y estructura del conjunto de representantes . . . . .	25
4.1.3. Análisis visual del comportamiento temporal . . . . .	25
4.1.4. Generación y ejecución de modelos de predicción . . . . .	27
4.1.5. Evaluación de modelos . . . . .	27
4.2. Fundamentación teórica de los modelos . . . . .	29
4.2.1. ARIMA . . . . .	29
4.2.2. Holt-Winters . . . . .	29
4.2.3. KNN . . . . .	31
4.2.4. ARNN . . . . .	31
4.2.5. SVM . . . . .	32
4.2.6. TBATS . . . . .	33
4.2.7. LSTM . . . . .	34
4.2.8. Combinación de modelos . . . . .	35
4.3. Aplicación de modelos al conjunto de datos reales . . . . .	37

4.3.1. Comparación inicial de resultados . . . . .	37
4.3.2. Predicciones a futuro . . . . .	42
<b>5. Conclusiones</b>	<b>55</b>
<b>Bibliografía</b>	<b>57</b>

## Índice de figuras

1.	Datos iniciales de retrasos en septiembre 2017 . . . . .	12
2.	Gráfico inicial de retrasos en todas las paradas en septiembre 2017 . . . . .	12
3.	Gráfico inicial de retrasos por parada en septiembre 2017 (parte 1) . . . . .	13
4.	Gráfico inicial de retrasos por parada en septiembre 2017 (parte 2) . . . . .	14
5.	Gráfica del método del codo . . . . .	18
6.	Gráfica del método de la silueta . . . . .	19
7.	Gráfica del estadístico Gap . . . . .	21
8.	Dendograma para tres clusters . . . . .	22
9.	Gráfico de los representantes . . . . .	26
10.	Boxplots del MAE y del MSE (clusters 1 y 2) . . . . .	40
11.	Boxplots del MAE y del MSE (cluster 3) . . . . .	41
12.	Gráfico de días de datos reales frente a predicción del cluster 1 . . . . .	44
13.	Gráfico de horas de datos reales frente a predicción del cluster 1 . . . . .	45
14.	Gráfico de predicciones combinadas del cluster 1 . . . . .	45
15.	Gráfico de predicciones por el método CLS del cluster 1 . . . . .	46
16.	Gráfico de predicciones para el 1 de octubre en el cluster 1 . . . . .	47
17.	Gráfico de días de datos reales frente a predicción del cluster 2 . . . . .	48
18.	Gráfico de horas de datos reales frente a predicción del cluster 2 . . . . .	48
19.	Gráfico de predicciones combinadas del cluster 2 . . . . .	49
20.	Gráfico de predicciones por el método CLS del cluster 2 . . . . .	50
21.	Gráfico de predicciones para el 1 de octubre en el cluster 2 . . . . .	50
22.	Gráfico de días de datos reales frente a predicción del cluster 3 . . . . .	51
23.	Gráfico de horas de datos reales frente a predicción del cluster 3 . . . . .	52
24.	Gráfico de predicciones combinadas del cluster 3 . . . . .	52
25.	Gráfico de predicciones por el método TBATS del cluster 3 . . . . .	53
26.	Gráfico de predicciones para el 1 de octubre en el cluster 3 . . . . .	54

## Índice de cuadros

1.	MAE promedio de cada modelo por cluster . . . . .	38
2.	MSE promedio de cada modelo por cluster . . . . .	39
3.	Errores MAE y MSE para el histórico 5 del cluster 1 . . . . .	43
4.	Errores MAE y MSE para la predicción combinada del cluster 1 . . . . .	46
5.	Errores MAE y MSE para el histórico 5 del cluster 2 . . . . .	47
6.	Errores MAE y MSE para la predicción combinada del cluster 2 . . . . .	49
7.	Errores MAE y MSE para el histórico 5 del cluster 3 . . . . .	51
8.	Errores MAE y MSE para la predicción combinada del cluster 3 . . . . .	53

# Resumen

En un contexto aeroportuario cada vez más exigente y competitivo, contar con herramientas para anticipar el comportamiento del sistema y optimizar recursos es crucial para garantizar una gestión eficiente y resiliente. Este trabajo aborda esa necesidad mediante análisis de datos avanzados y predicción de retrasos en aeropuertos, utilizando técnicas de minería de datos y modelado de series temporales. El proyecto está fundamentado en datos reales de tráfico aéreo y retrasos medios diarios recopilados de 50 aeropuertos europeos durante el mes de septiembre de 2017.

El objetivo principal es comprender los patrones que explican la aparición de retrasos y construir modelos predictivos que permitan anticiparlos con suficiente antelación para facilitar una toma de decisiones proactiva. Para ello, se utiliza clusterización, de forma que se agrupan aeropuertos con dinámicas similares, permitiendo seleccionar casos representativos que reducen la complejidad del problema sin perder capacidad explicativa.

Sobre estas series, se implementan y comparan diferentes modelos de predicción como ARIMA, ARNN, KNN y combinaciones de estos, integrando el tráfico aéreo como variable regresora cuando es posible. Se analizan los resultados dentro y fuera de muestra, prestando especial atención a escalas relevantes para la operación aeroportuaria, como predicciones a 24 horas.

Este enfoque ofrece una metodología robusta, escalable y fácilmente replicable, con aplicaciones directas en la planificación de operaciones, asignación de recursos y mejora de la puntualidad en infraestructuras críticas. En última instancia, el proyecto contribuye a transformar datos históricos en conocimiento estratégico, potenciando una gestión más inteligente y adaptativa del sistema aeroportuario europeo.

**Palabras clave:** predicción de retrasos, tráfico aéreo, series temporales, clusterización, modelos de predicción, regresores, ARIMA, ARNN, KNN, LSTM, horizonte de predicción, eficiencia operativa.

# Abstract

In an increasingly demanding and competitive airport environment, having tools to anticipate system behaviour and optimising resources is crucial to ensure efficient and resilient management. This work addresses this need through advanced data analysis and prediction of airport delays, using data mining and time series modelling techniques. The project is based on real air traffic data and daily average delays collected from 50 European airports during the month of September 2017.

The main objective is to understand the patterns that explain the occurrence of delays and to build predictive models to anticipate them early enough to facilitate proactive decision making. To do this, clustering is used, so that airports with similar dynamics are grouped together, allowing the selection of representative cases that reduce the complexity of the problem without losing explanatory capacity.

Based on these series, different prediction models such as ARIMA, ARNN, KNN and combinations of these are implemented and compared, integrating air traffic as a regressor variable when possible. In-sample and out-of-sample results are analysed, paying special attention to scales relevant to airport operations, such as 24-hour forecasts.

This approach offers a robust, scalable and easily replicable methodology with direct applications in operations planning, resource allocation and improving punctuality in critical infrastructure. Ultimately, the project contributes to transforming historical data into strategic knowledge, enabling smarter and more adaptive management of the European airport system.

**Keywords:** delay prediction, air traffic, time series, clustering, prediction models, regressors, ARIMA, ARNNN, KNN, LSTM, prediction horizon, operational efficiency.

# 1. Introducción

## 1.1. Motivación

Debido a la saturación del espacio aéreo y la creciente complejidad en la gestión operativa, el desafío para la mayoría de los aeropuertos es reducir los **retrasos operacionales** y mantener un **alto nivel de servicio**, minimizando el riesgo de interrupciones. Esto implica no solo **costes económicos y medioambientales**, sino también una afectación directa a la **experiencia del pasajero**. En este contexto, contar con **capacidades predictivas** se ha convertido en un **factor estratégico clave** para la **competitividad** en la **planificación y toma de decisiones** dentro del sector [BJ23].

Gracias a la disponibilidad de **grandes volúmenes de datos**, hoy es posible aplicar **técnicas avanzadas de análisis predictivo**, incluso cuando los datos no están perfectamente ordenados o provienen de meses no consecutivos. El uso de **modelos estadísticos** como **ARIMA** o **TBATS**, junto con métodos de **inteligencia artificial** como las **redes neuronales autorregresivas (ARNN)** o los **modelos de memoria a largo y corto plazo (LSTM)**, han demostrado ser herramientas muy eficaces para anticiparse a **patrones complejos** en **entornos dinámicos** [BJR08, STS18].

Este proyecto se centra en **analizar y predecir retrasos** en el tráfico aéreo a partir del número de **vuelos diarios**. Para ello, se han utilizado **series temporales** de 50 aeropuertos europeos, combinando **técnicas de agrupación** con **modelos de predicción avanzados**. El objetivo es evaluar no solo la **precisión de cada modelo individualmente**, sino también el **valor añadido de su combinación**. La hipótesis principal es que, al **agrupar aeropuertos con comportamientos similares** y seleccionar casos representativos, se puede **mejorar significativamente la calidad de las predicciones**.

En primer lugar, se agruparán los datos mediante técnicas heurísticas como el **k-medias** para detectar **patrones comunes** entre aeropuertos con trayectorias similares de tráfico y retrasos [PKG11]. Luego, se seleccionarán los **representantes de cada grupo**, los cuales se utilizarán en distintos modelos predictivos, con y sin **variables regresoras**, buscando siempre **mejorar la precisión de las predicciones** [DG24].

La eficacia de cada modelo se evaluará mediante indicadores como el *Mean Absolute Error (MAE)* y el *Mean Squared Error (MSE)*, en un horizonte de predicción de **1 a 24 horas**.

El estudio está organizado en cinco capítulos. El primero expone el **enfoque general** y el **contexto**. El segundo capítulo trata sobre la **recopilación y limpieza de datos**. El tercero aborda la **agrupación y selección representativa** de series. En el cuarto se detallan la **im-**



plementación y evaluación de los modelos predictivos. Finalmente, el quinto presenta las conclusiones y posibles líneas de mejora y trabajo futuro.

## 1.2. Objetivos

Los principales objetivos de este trabajo son:

- Comprender los **fundamentos teóricos de la predicción de series temporales**, enfocándose en su aplicación al contexto del **tráfico aéreo** y los **retrasos en aeropuertos**.
- Analizar la **relación entre el volumen de tráfico aéreo y los retrasos operacionales**.
- Aplicar **técnicas de agrupación (clusterización)** para identificar aeropuertos con **patrones de comportamiento similares**.
- Implementar **modelos de predicción de series temporales**, evaluando tanto su **rendimiento individual** como su **potencial combinado**.
- Incorporar **variables regresoras** en los modelos que lo permitan, para **mejorar la calidad de las predicciones**.
- Comparar los resultados mediante **métricas de error**, tanto **dentro como fuera de muestra**.
- Evaluar la **evolución de errores** en distintos **horizontes de predicción** (de 1 a 24 horas), con especial interés en los intervalos más relevantes para la **toma de decisiones operacionales**.
- Visualizar y analizar los resultados mediante **gráficos y tablas comparativas** que faciliten la **interpretación de los modelos**.

## 1.3. Plan de trabajo

Las fases desarrolladas a lo largo del proyecto fueron:

1. **Investigación.** Durante las primeras semanas se realizó una investigación sobre conceptos clave de **series temporales**, junto a una revisión de documentación técnica, tutoriales y recursos sobre **técnicas y librerías de R**.
  - Implementación del algoritmo *K-Nearest Neighbors (KNN)* mediante el paquete `caret` [Tiw17].

- Uso de `frbs.learn` para sistemas de inferencia difusa [RDoND].
  - Predicción con `tsfknn` en R [MFC+19].
  - Modelado de alto rendimiento con *Auto ARIMA* en Python y R [HK08, Sin25].
  - Aplicación de **redes neuronales autorregresivas** [GR19] mediante librerías externas no disponibles directamente en R.
  - Uso del parámetro `xreg` en `auto.arima` y `tbats` para incorporar variables regresoras [HR11].
  - Criterios de selección de modelos para **series diarias con estacionalidades complejas** [HA18, Kou16].
2. **Desarrollo.** Una vez consolidados los conceptos, se realizó el **tratamiento de las series temporales** con datos de **retrasos medios** y **tráfico aéreo**. Se identificaron **patrones comunes** y se seleccionaron **series representativas** para aplicar modelos predictivos, evaluando su rendimiento mediante distintas **métricas de error**.
  3. **Interpretación.** Se hizo un **análisis comparativo de los resultados**, evaluando la eficacia de combinar modelos. Se generaron **gráficos y tablas resumen** para interpretar los errores a distintos **horizontes de predicción**, y se predijeron valores futuros.

## 1.4. Material de trabajo

Se empleó **Excel** para almacenar y visualizar preliminarmente los datos, y el lenguaje de programación **R** para todo el proceso de **tratamiento, predicción y visualización**. Para facilitar el uso de este lenguaje, se trabajó en el entorno libre y de código abierto **RStudio**. Además, para implementar modelos complejos basados en **redes neuronales**, se configuró un entorno **Conda** localmente, que fue integrado en R mediante la librería `reticulate`. Esto permitió utilizar `tensorflow` directamente desde R, facilitando la ejecución de modelos avanzados y mejorando significativamente la **flexibilidad** y **escalabilidad** del análisis.

## 2. Recopilación, limpieza y estructuración de los datos

El proceso de **recopilación y preparación de los datos** es **crítico** en este tipo de proyectos, ya que **impacta directamente en la calidad del análisis y la fiabilidad de los resultados**. En muchos estudios de este tipo, esta etapa tiende a consumir una gran parte del tiempo total, especialmente cuando los datos proceden de sistemas reales y presentan **problemas de formato, valores perdidos o inconsistencias**. Lejos de ser un paso único, se trata de un **proceso iterativo** que se va refinando a medida que avanza el análisis [Osb12].

### 2.1. Origen de datos

Los datos empleados en este proyecto proceden de **ficheros Excel** con información de **50 aeropuertos europeos relevantes**, con registros por hora, desde las 07:00 hasta las 23:00 UTC. Sin embargo, una **limitación importante** con la que se contó fue la **falta de datos consecutivos para tener un mayor histórico**, ya que la información solo abarcaba meses específicos (marzo, junio, septiembre y diciembre) durante cuatro años (2015-2018). Por cuestiones legales, no se pudieron obtener datos de manera continua, por lo que **solo se empleó septiembre de 2017** para el estudio principal, garantizando así la consistencia del análisis.

De esta forma, se dispuso de dos conjuntos de datos diferenciados:

- Un fichero con los **retrasos medios de los aterrizajes**.
- Un fichero con el **tráfico aéreo por aeropuerto**.

Los datos abarcan **algunos de los aeropuertos más importantes de Europa**, como London Heathrow, Paris Charles de Gaulle, Amsterdam Schiphol, Frankfurt y Madrid-Barajas, entre otros. Cada aeropuerto cuenta con **17 observaciones diarias**, correspondientes a los registros horarios de 07:00 a 23:00 UTC.

Los **retrasos** se definieron como la **media**, para todos los vuelos que han aterrizado a una hora determinada, **de la diferencia (en segundos) entre la hora real de aterrizaje y la hora programada de aterrizaje**. Es posible encontrar valores negativos, lo que indica que un avión aterrizó antes de lo previsto, aunque estos casos son poco frecuentes.

Por último, se aplicó una **corrección previa a los datos** para eliminar registros inválidos, como retrasos registrados en horas sin ningún vuelo programado, asegurando así una base de datos **más limpia y fiable** desde el inicio del análisis.

## 2.2. Obtención de series temporales

Una vez completada la fase de recopilación de datos, se procede al **análisis exploratorio** del fichero que contiene los registros de retrasos correspondientes al mes de septiembre de 2017. Esta primera visualización global resulta fundamental para obtener una comprensión preliminar del comportamiento temporal de los aeropuertos y detectar posibles patrones o anomalías.

El conjunto de datos cuenta con una estructura horaria detallada y permite representar gráficamente las series temporales asociadas a los aeropuertos europeos. Esta representación inicial, mostrada en la *Figura 1*, proporciona una visión panorámica de los retrasos registrados y constituye el punto de partida sobre el cual se construirá el posterior modelado predictivo.

Date	EGLL	LFPG	EHAM	EDDF	LEMD	LEBL	EDDM	EGKK	LIRF	LFPD	EIDW	LSZH	EKCH
1 2017-09-01 07:00:00	808	194	1053	83	-154	299	170	353	-129	154	-7	240	5
2 2017-09-01 08:00:00	811	426	767	239	138	484	300	519	183	87	90	221	-71
3 2017-09-01 09:00:00	775	323	784	170	483	681	389	515	107	-5	234	417	110
4 2017-09-01 10:00:00	909	356	521	287	351	583	206	510	283	-43	563	795	-65
5 2017-09-01 11:00:00	1109	631	995	183	197	919	-74	796	409	616	147	479	282
6 2017-09-01 12:00:00	1038	653	523	580	448	367	400	1277	2021	447	248	563	-1
7 2017-09-01 13:00:00	1268	644	736	471	596	375	473	1440	2173	360	880	124	-28
8 2017-09-01 14:00:00	1036	586	970	295	571	87	414	1918	934	170	378	379	531
9 2017-09-01 15:00:00	1720	417	1083	58	345	299	64	1777	1410	381	303	571	303
10 2017-09-01 16:00:00	1151	649	921	213	494	454	96	1012	1072	260	608	-105	222
11 2017-09-01 17:00:00	1018	604	550	495	381	410	256	799	734	291	339	538	161
12 2017-09-01 18:00:00	893	1258	921	161	295	494	31	1022	535	751	219	221	255
13 2017-09-01 19:00:00	1401	2130	985	156	349	1107	365	1344	620	1556	-199	463	248
14 2017-09-01 20:00:00	1176	371	732	-74	409	612	247	870	-164	784	338	380	274
15 2017-09-01 21:00:00	753	348	591	45	718	358	200	364	343	645	-37	620	297

Figura 1: Datos iniciales de retrasos en septiembre 2017

Viendo todas las paradas primero, los retrasos tendrían la siguiente forma:

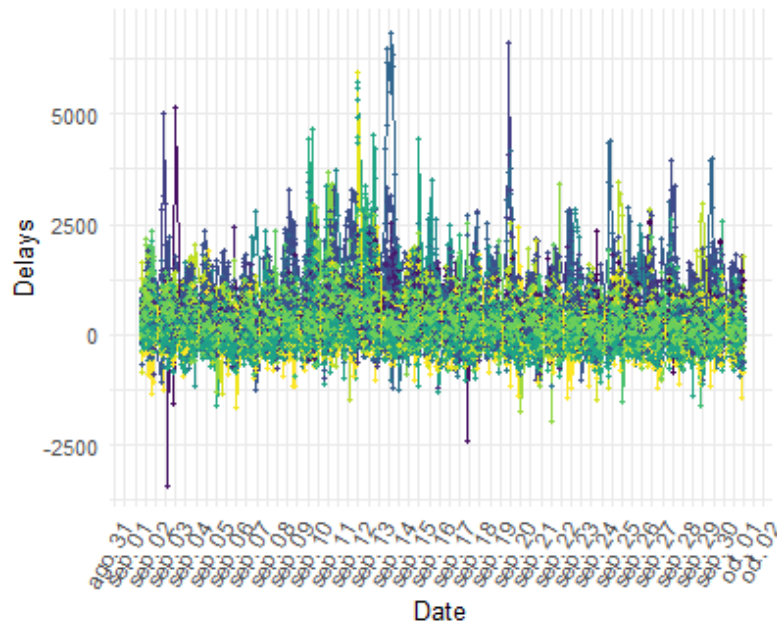


Figura 2: Gráfico inicial de retrasos en todas las paradas en septiembre 2017

Como en la *Figura 2* no se aprecia bien qué sucede con cada aeropuerto, para ver el detalle y poder entender mejor los retrasos en los vuelos, se dibujaron gráficos de una parada para cada aeropuerto mediante un bucle.

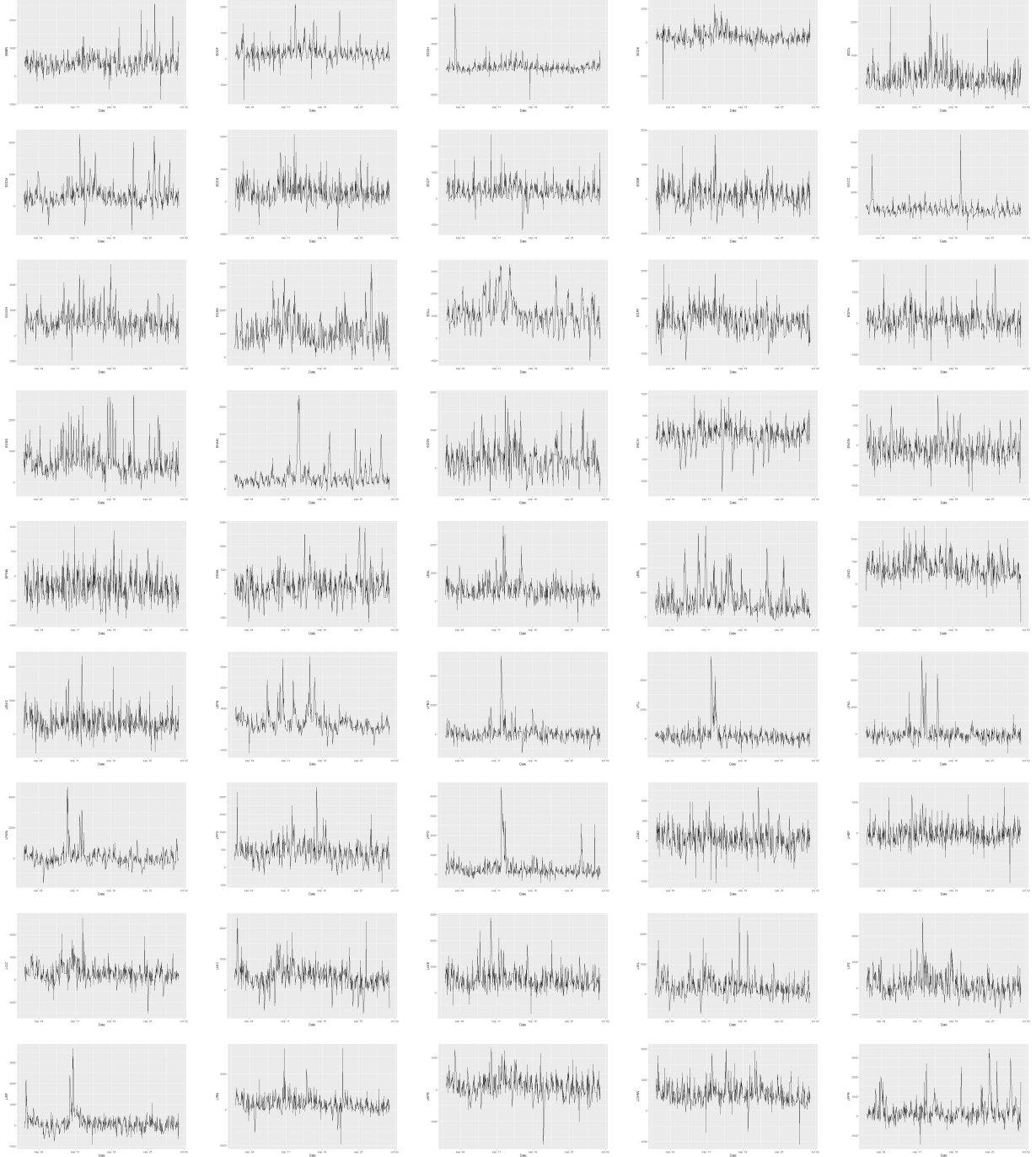


Figura 3: Gráfico inicial de retrasos por parada en septiembre 2017 (parte 1)

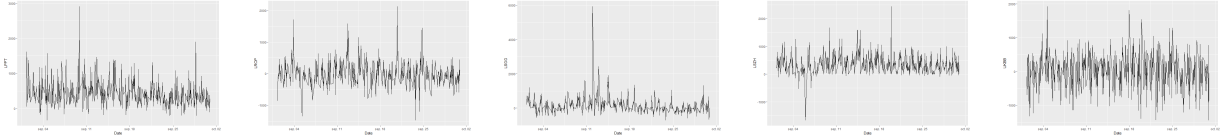


Figura 4: Gráfico inicial de retrasos por parada en septiembre 2017 (parte 2)

En términos generales, analizando tanto *Figura 3* como *Figura 4*, el comportamiento de las series temporales refleja una **evolución relativamente estable, sin tendencias de crecimiento o decrecimiento claras**. No obstante, se observan **picos aislados en numerosos aeropuertos**, probablemente asociados a **eventos excepcionales o disrupciones puntuales** en las operaciones. Aunque no se detecta una estacionalidad evidente a simple vista, **no se descarta la existencia de ciertos patrones recurrentes** que podrían emerger en fases posteriores del análisis detallado.

Además, es importante señalar que tanto la **intensidad** como la **frecuencia de estos picos varían significativamente entre aeropuertos**, lo cual sugiere posibles diferencias en la **sensibilidad operativa**, la **capacidad de respuesta ante incidencias** o en el propio **volumen de tráfico aéreo** de cada uno.

Antes de proceder con el modelado, se comprobó que efectivamente no existen series constantes entre los datos. Para ello, se calculó la **desviación estándar** de cada serie temporal, considerando que una serie será válida si su desviación estándar es **distinta de cero**. En este caso, **todas las series presentan variabilidad suficiente**, por lo que **no es necesario eliminar ninguna**.

A continuación, se convirtió la matriz de datos al **formato de serie temporal**, requisito esencial para los modelos que se aplicarán en fases posteriores. Dado que las observaciones cubren el horario de 7:00 a 23:00 (17 horas al día) durante el mes de septiembre (30 días), **cada serie consta de 510 observaciones**. El objeto final contiene 50 columnas, cada una representando a un aeropuerto europeo incluido en el estudio.

### 3. Técnicas de análisis de conglomerados

En la actualidad, el análisis de datos se enfrenta con frecuencia a conjuntos masivos de información, caracterizados por un **volumen elevado de observaciones** y una **gran variedad de variables**. Este aumento en la disponibilidad de datos permite obtener una **visión más detallada** de los fenómenos estudiados, pero también introduce **desafíos significativos**, especialmente en lo que respecta a la **identificación de patrones relevantes**, la **reducción de ruido** y la **interpretación adecuada de la información**.

Por este motivo, resulta fundamental el uso de **técnicas analíticas robustas** que permitan **reducir la complejidad** del conjunto **sin perder los patrones esenciales** del sistema. Para abordar este desafío, se recurre a metodologías consolidadas en el ámbito del análisis de series temporales y el aprendizaje automático no supervisado.

Entre los enfoques más utilizados destacan, por un lado, los **métodos de análisis de series temporales**, que permiten capturar y modelar la **evolución dinámica** de los datos a lo largo del tiempo, incorporando aspectos como **tendencias, ciclos, estacionalidades o eventos atípicos**. Por otro lado, las **técnicas de agrupamiento** o *clustering* son útiles para **segmentar el conjunto de observaciones en grupos homogéneos**, identificando **comportamientos recurrentes o similares** sin necesidad de una variable objetivo, lo cual resulta especialmente útil.

La aplicación conjunta de estos enfoques permite no solo entender la estructura temporal y la heterogeneidad del sistema analizado, sino también preparar los datos para etapas posteriores como la predicción, la detección de anomalías o la toma de decisiones estratégicas basadas en evidencias. En definitiva, se busca alcanzar un **equilibrio entre reducción de la dimensionalidad, conservación de la información crítica y facilitación del análisis**, sentando así las bases para un estudio **sólido, riguroso y orientado a resultados**.

#### 3.1. Cálculo de distancias

En el análisis de series temporales, la **selección de una métrica adecuada** para evaluar la similitud entre observaciones es **fundamental**, ya que su elección no solo condiciona la **calidad del clustering** resultante, sino que también determina en gran medida la **fidelidad** con la que se capturan las estructuras dinámicas subyacentes [ASW15]. Una métrica mal escogida puede inducir agrupaciones menos relevantes, comprometiendo así la interpretación de los resultados y su aplicabilidad posterior.

En este proyecto, las distancias se calcularon considerando **todas las observaciones dispo-**

nibles, con las series **previamente estandarizadas**. Este proceso de normalización, consistente en centrar las series restando su media y escalarlas dividiendo por la desviación estándar de cada serie, tiene como objetivo **homogeneizar las escalas para evitar que series con valores absolutos mayores dominen la medida de distancia**. Posteriormente, se **traspuso la matriz** para que cada fila represente un aeropuerto y cada columna una observación horaria, adaptando así la estructura a las funciones de cálculo de distancias multivariantes.

Se exploraron diversas métricas incluidas en el paquete `TSclust` de R [MV14], seleccionando finalmente la **distancia basada en la función de autocorrelación (ACF)** como la más adecuada para este contexto. Esta elección se fundamenta en su capacidad para capturar la **dependencia temporal** de las series, así como su **robustez** frente a desplazamientos y variaciones de escala. En concreto, esta métrica mide la distancia entre dos series  $X$  e  $Y$  como la norma euclídea del vector diferencia entre sus autocorrelaciones truncadas hasta un orden  $k$ :

$$d_{ACF}(X, Y) = \left\| \rho_X^{(k)} - \rho_Y^{(k)} \right\|_2$$

donde  $\rho_X^{(k)}$  y  $\rho_Y^{(k)}$  son los vectores de **autocorrelaciones** de las series hasta el rezago  $k$ , representando así la **estructura interna** y el **comportamiento dinámico** de cada aeropuerto.

Desde un punto de vista teórico, la función de autocorrelación permite capturar información relevante sobre la **persistencia temporal**, la **estacionalidad** y las **dinámicas recurrentes** en las series, lo cual resulta fundamental en entornos operativos como el de la gestión aeroportuaria. La **matriz de distancias** obtenida con esta métrica fue simétrica y reflejó la disimilitud relativa entre pares de aeropuertos. Por ejemplo, la distancia entre París (LFPG) y Frankfurt (EDDF) fue de 0,479, lo que indica una alta similitud estructural en sus patrones horarios de retraso, mientras que entre París (LFPG) y Dublín (EIDW), el valor ascendió a 0,715, señalando una mayor discrepancia en el comportamiento temporal.

Comparando la **métrica ACF** frente a otras métricas tradicionales, las ventajas clave son las siguientes:

- **Captura eficaz de la dependencia temporal**, lo que permite detectar patrones recurrentes o persistentes.
- **Invarianza a traslaciones verticales y horizontales**, garantizando que las diferencias de nivel o fase no distorsionen el análisis.
- **Comparabilidad entre series con distinta magnitud**, siempre que conserven una estructura temporal semejante.



- **Facilidad de interpretación**, especialmente en aplicaciones orientadas a entender comportamientos operativos o estructurales.

Para el resto de métricas se puede destacar lo siguiente [DM09]:

- **Distancia basada en periodogramas (PER)**: compara el contenido espectral de las series, es decir, su comportamiento en el dominio de la frecuencia. Aunque es útil para detectar periodicidades comunes, esta métrica fue descartada al no aportar mejoras relevantes en este conjunto de datos.
- **Distancia euclídea**: mide diferencias punto a punto, pero ignora la estructura temporal, por lo que puede ser inapropiada para datos con autocorrelación significativa.

En síntesis, la utilización de la **distancia ACF** permitió construir una **matriz de similitudes sólida y estadísticamente justificada**, sirviendo como base para el posterior **agrupamiento jerárquico aglomerativo**. Esta estrategia ha demostrado ser especialmente eficaz para extraer **similitudes dinámicas en datos temporales complejos**, proporcionando una **caracterización robusta y significativa** del sistema aeroportuario analizado.

## 3.2. Creación de grupos

Una vez definida una **métrica robusta de similitud** entre series temporales, se procede a aplicar **técnicas de agrupamiento no supervisado** con el objetivo de **identificar patrones comunes y estructuras latentes** en el comportamiento de los aeropuertos analizados. Esta fase resulta clave para **simplificar la complejidad del sistema** y favorecer una **interpretación estratégica** de los datos.

En este trabajo, se ha optado por el enfoque de *clustering* jerárquico aglomerativo, una metodología ampliamente reconocida por su capacidad para **construir representaciones visuales jerárquicas** (dendrogramas) y por **no requerir la especificación previa del número de clusters** [K+24]. Esta característica resulta especialmente ventajosa cuando se trabaja con conjuntos de datos heterogéneos y de alta dimensionalidad, como es el caso del tráfico aéreo.

Formalmente, dado un conjunto de series temporales  $\{X_1, X_2, \dots, X_n\}$  y su correspondiente matriz de distancias simétrica  $D = [d_{ij}]$ , donde  $d_{ij} = d(X_i, X_j)$ , el algoritmo comienza considerando cada serie como un cluster individual. En cada iteración, se identifican los dos clusters más parecidos, según un **criterio de enlace** específico y se fusionan, generando así una **jerarquía progresiva de agrupamientos** [Fra05].

Para la **selección del número óptimo de clusters**, se emplearon tanto **métodos visuales** como el **corte del dendrograma**, como **métricas cuantitativas de validación interna** que

evalúan la **cohesión y separación** de los grupos formados. Ahora, se presentan los fundamentos teóricos de los tres principales métodos utilizados para este propósito:

- **Método del codo (*Elbow Method*)**: este enfoque evalúa la **suma total de cuadrados intra-cluster** (*Within-Cluster Sum of Squares*, WCSS) en función del número de clusters  $k$ . El principio fundamental consiste en identificar el punto a partir del cual **incrementar el número de clusters aporta una mejora marginal** en la compactación de los grupos. Dicho punto, conocido como el codo, representa un **compromiso óptimo entre simplicidad del modelo y precisión** [SEK21].

En la práctica, tal como se observa en la *Figura 5*, se produce una **disminución pronunciada de la WCSS** a medida que aumenta  $k$ , siendo especialmente notable hasta  $k = 3$ . A partir de ese valor, las ganancias adicionales en homogeneidad se reducen considerablemente, formando un **claro punto de inflexión**. Por tanto, este método sugiere de forma razonada que el **número óptimo de grupos es 3**.

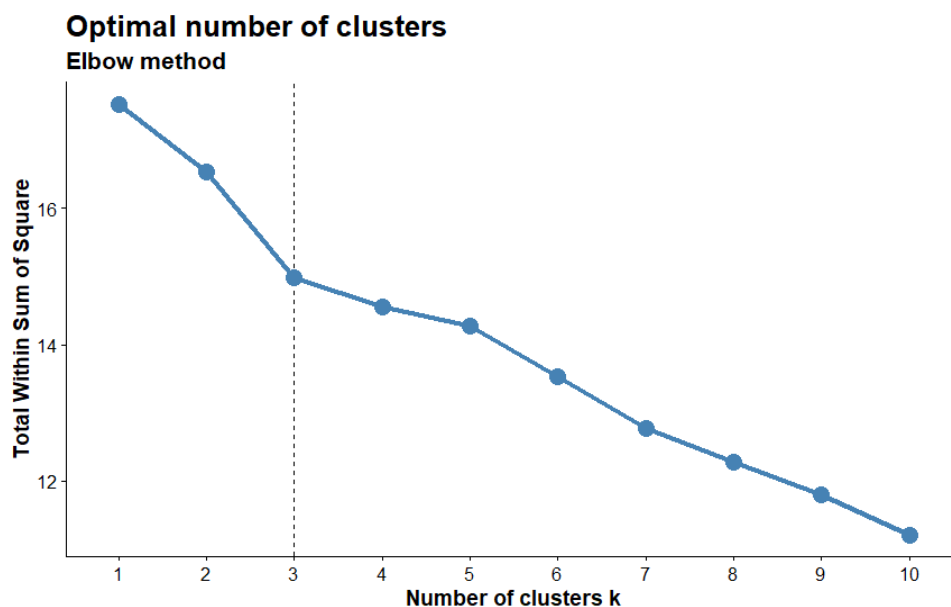


Figura 5: Gráfica del método del codo

- **Coefficiente de silueta (*Silhouette Method*)**: este método permite evaluar la **calidad del agrupamiento** mediante una medición conjunta de la **cohesión interna** y la **separación entre clusters** [Mod23]. Mide qué tan bien se encuentra una observación dentro de su propio grupo en comparación con otros grupos cercanos. Para cada observación  $i$ , se calcula un **coeficiente de silueta**  $s_i$  que toma valores en el intervalo  $[-1, 1]$ .

Un valor cercano a 1 indica que la observación está **bien ubicada** dentro de su cluster y se encuentra **claramente separada** de los demás. Por el contrario, un valor cercano

a  $-1$  sugiere una **asignación errónea**, y valores cercanos a 0 reflejan ambigüedad en la asignación.

Formalmente, para un punto  $i$  en el cluster  $C_i$ , el coeficiente de silueta  $s_i$  se define como:

$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)}$$

donde:

- $a_i$  es la **distancia promedio** del punto  $i$  a los demás puntos dentro de su mismo cluster  $C_i$  (**cohesión**).
- $b_i$  es la **distancia promedio** del punto  $i$  al cluster vecino más cercano (el que no contiene a  $i$  y cuya distancia promedio es mínima) (**separación**).

El valor promedio de  $s_i$  para todos los puntos del conjunto de datos se calcula para distintos valores de  $k$ , y el número de clusters óptimo es aquel que **maximiza dicho promedio**, indicando la mejor estructura de agrupamiento, con grupos bien definidos y diferenciados.

En el caso de estudio, como se observa en la *Figura 6*, el **ancho de silueta promedio alcanza su valor máximo alrededor de  $k = 2$** . A partir de ese punto, el coeficiente tiende a disminuir de forma más atenuada, indicando que configuraciones más complejas no mejoran sustancialmente la calidad del agrupamiento.

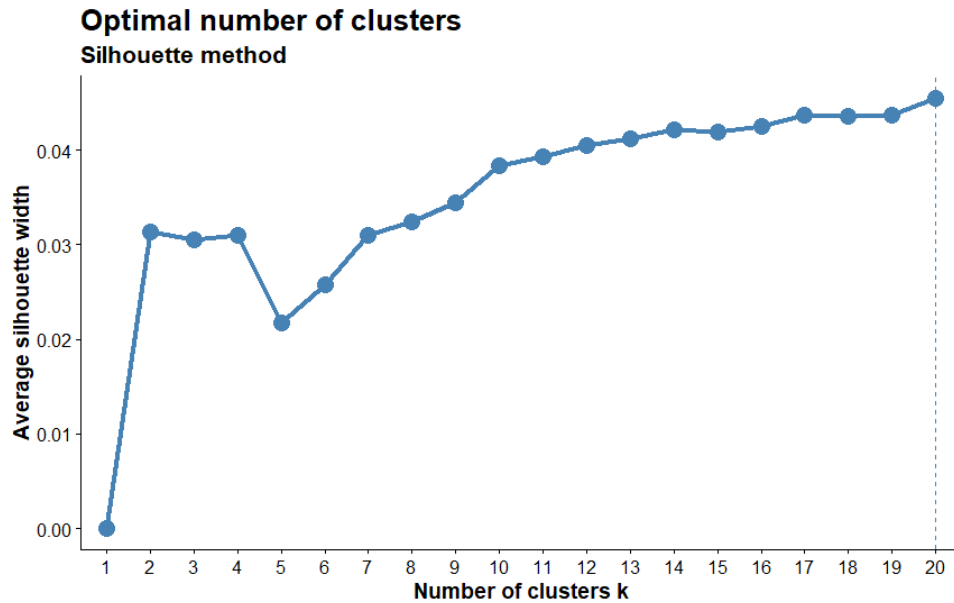


Figura 6: Gráfica del método de la silueta

- **Estadístico Gap (*Gap Statistic*)**: este estadístico evalúa la **significancia estadística de una estructura de clustering** comparando la **dispersión intra-cluster observada** con la que se esperaría bajo una **distribución nula aleatoria** [C+04]. La idea central es detectar si los agrupamientos generados por el algoritmo son más compactos que los que surgirían por azar, lo que permitiría validar su relevancia estadística. El estadístico *Gap* para un número de clusters  $k$  se define como:

$$Gap(k) = E_n^*\{\log(W_k^*)\} - \log(W_k)$$

donde:

- $W_k$  es una medida de la **dispersión intra-cluster** para  $k$  clusters en los datos observados.
- $E_n^*\{\log(W_k^*)\}$  es el valor esperado de  $\log(W_k)$  para  $k$  clusters en datos generados bajo una **distribución nula** de referencia (estimado mediante *bootstrapping*, método de remuestreo estadístico que permite aproximar la distribución).

La elección del número óptimo de clusters, denotado como  $\hat{k}$ , se realiza seleccionando el valor más pequeño de  $k$  tal que:

$$Gap(k) \geq Gap(k+1) - s_{k+1}$$

donde  $s_{k+1}$  es la **desviación estándar** del Gap statistic para  $k+1$  clusters. Esta formulación introduce una penalización por complejidad, buscando evitar particiones excesivas que no representen mejoras sustanciales.

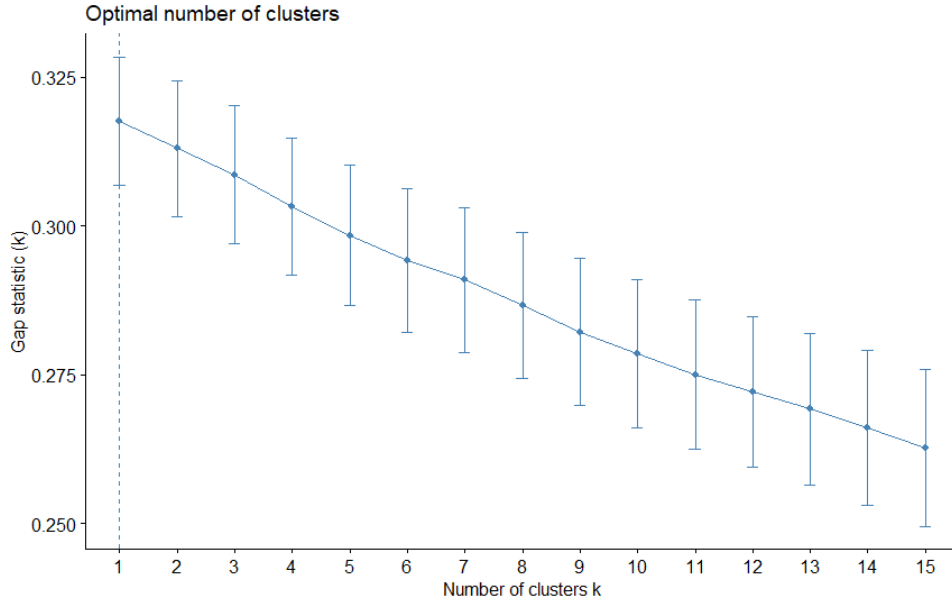


Figura 7: Gráfica del estadístico Gap

En la práctica, al inspeccionar la *Figura 7* y aplicar el criterio de selección, se observa que para  $k = 1$ , el valor del estadístico Gap es relativamente elevado, y además se cumple la condición  $Gap(1) \geq Gap(2) - s_2$ . Esto sugiere que  $k = 1$  podría ser considerado como el número óptimo de clusters, al no encontrarse evidencia estadística suficiente que justifique una partición más compleja. Esta conclusión respalda la hipótesis de que el conjunto presenta un **comportamiento homogéneo** o con baja diferenciación estructural entre series.

### 3.3. Elección de representantes

Ante la **disparidad en los resultados** proporcionados por los distintos métodos de validación, se optó por seleccionar  $k = 3$  como número óptimo de clusters. Esta elección busca alcanzar un **equilibrio adecuado entre la granularidad y la interpretabilidad del agrupamiento**, permitiendo capturar **variaciones estructurales relevantes** presentes en las series temporales sin caer en una fragmentación excesiva que complique su análisis o interpretación posterior.

Definido el número de clusters, se representó la **estructura jerárquica del agrupamiento** mediante un **dendrograma**, construido utilizando la función `fviz_dend()`. Esta representación gráfica ofrece una visión clara de cómo las series individuales se agrupan progresivamente, destacando:

- Las **hojas** del dendrograma, que corresponden a cada una de las series temporales analizadas.

- Las **ramas** que conectan grupos de series en función de su **similitud estructural**.
- La **altura** de cada unión, que refleja la **distancia** entre los clusters que se fusionan en cada etapa del proceso.

Esta visualización aporta claridad en la interpretación de las relaciones entre los grupos formados y constituye una herramienta clave para identificar **niveles jerárquicos de similitud**, ofreciendo un recurso valioso para decisiones analíticas o estratégicas basadas en dichos patrones.

Adicionalmente, para facilitar una **evaluación visual directa de la separación y cohesión entre los clusters**, se realizó una proyección bidimensional mediante la función `fviz_cluster()`, la cual permitió observar la **disposición espacial de los grupos** en un plano reducido, conservando la estructura relativa entre ellos y facilitando la identificación de posibles solapamientos o agrupaciones bien definidas.

En conjunto, ambas representaciones proporcionan una base sólida para interpretar y comunicar los resultados del proceso de *clustering*, reforzando la **validez técnica** del enfoque adoptado y destacando su utilidad práctica en el contexto de análisis de datos temporales complejos.

En la práctica, el dendrograma obtenido para el caso de estudio fue el siguiente:

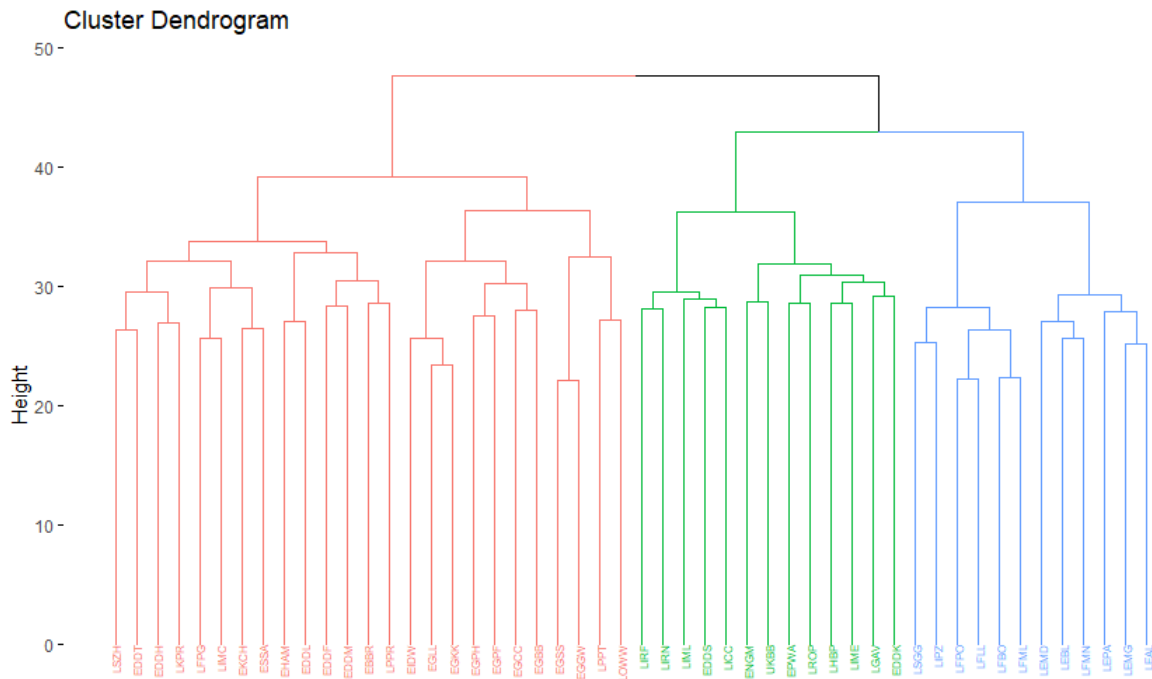


Figura 8: Dendrograma para tres clusters

La *Figura 8* permite apreciar los **tres grupos distintos de aeropuertos** con patrones de comportamiento temporal similares. El primer grupo, compuesto por **25 aeropuertos**, incluye

grandes aeropuertos europeos como London Heathrow y Paris Charles de Gaulle; el segundo, con **12 aeropuertos**, contiene entre otros a Madrid-Barajas y Barcelona-El Prat; y el tercero, con **13**, incluye casos como Roma Fiumicino y Atenas Eleftherios Venizelos. Cada grupo se encuentra **diferenciado cromáticamente**, y el **punto de corte jerárquico** para su segmentación se sitúa en una **altura coherente y comparable**, lo que refuerza la validez de la agrupación obtenida.

Con el objetivo de profundizar en el análisis, se procede a la **selección de una serie representativa por cada cluster**, que servirá como ejemplar canónico de la dinámica interna de su grupo. Para ello, se emplea como criterio principal la **varianza intracluster**, identificando dentro de cada grupo la serie que presenta la **mayor variabilidad temporal**, debido a que una serie con mayor varianza tiende a reflejar **una estructura más rica y matizada** del comportamiento del cluster.

En base a este criterio, según la visualización de las varianzas, las series seleccionadas como representantes fueron: **EHAM** (grupo 1), **LFML** (grupo 2) y **UKBB** (grupo 3). Estas fueron posteriormente **exportadas en archivos separados** utilizando el paquete **openxlsx**, con el fin de **preservar y facilitar su análisis detallado en futuras etapas** del proyecto.

Adicionalmente, se exploraron otros posibles criterios de selección, como la **media temporal más elevada** o la **presencia de valores extremos**. Sin embargo, fueron finalmente descartados al no ofrecer **mejoras sustanciales en términos de representatividad** frente al enfoque basado en la varianza. Esta elección metodológica refuerza el compromiso con una **selección técnica robusta** que permita extraer conclusiones fiables a partir de las series elegidas.

## 4. Técnicas de predicción

Tras haber obtenido, de cada uno de los grupos, las **series temporales representantes**, en este capítulo se aborda la **predicción de las mismas**. Para ello, se han recopilado **varias técnicas**: algunas son propias de las series temporales (ARIMA y TBATS), otras suelen enseñarse con datos de sección cruzada (redes neuronales, K vecinos más próximos, máquinas de vectores soporte) y las demás surgen de la combinación de las predicciones obtenidas en otros modelos con el fin de mejorar los resultados.

Cabe destacar que, el problema se basó en la predicción de una **variable continua**, ya que no era en un problema de clasificación, al no buscar obtener una variable categórica como respuesta.

Los distintos apartados de esta sección fundamentaran todos los aspectos comunes a los modelos de predicción, explicaran la base teórica de los distintos modelos utilizados, además de la combinación de estos, y mostrarán y analizarán los resultados obtenidos para cada cluster, respectivamente.

### 4.1. Introducción a la predicción de series temporales

#### 4.1.1. Enfoque general y modelos considerados

Tras haber identificado los **representantes de cada grupo**, el análisis avanza hacia una de sus fases más estratégicas: la **predicción del comportamiento futuro de las mismas**. Este paso es fundamental, ya que permite transformar el conocimiento estructural extraído en **capacidades anticipativas** que apoyen la toma de decisiones.

Para ello, se ha diseñado un **marco comparativo robusto** que integra una amplia gama de modelos predictivos. Se incluyen tanto **métodos clásicos de series temporales**, como **ARIMA**, **HOLT** y **TBATS**, conocidos por su solidez en entornos estacionarios y con patrones estacionales definidos; como también **algoritmos modernos de aprendizaje automático**, tales como **KNN**, **ARNN**, **SVM** y **LSTM**. Estos últimos ofrecen una mayor **flexibilidad no lineal** y una mejor adaptación a la **complejidad estructural** del tráfico aeroportuario.

Aunque algunos de estos modelos, especialmente los de tipo **KNN**, **SVM** o incluso **LSTM**, han sido concebidos originalmente para datos **transversales o secuenciales de distinta índole**, aquí han sido **adaptados cuidadosamente al dominio de las series temporales**, formulándolos como **problemas de regresión**. El foco se ha puesto en predecir **valores continuos futuros**, lo que permite una estimación directa de la evolución del tráfico sin recurrir a discretizaciones que comprometan la precisión.

Además del uso aislado de cada modelo, se ha explorado la posibilidad de **combinar múlti-**



ples técnicas con el objetivo de **incrementar la robustez y precisión de las predicciones**. Estas combinaciones se evalúan de forma sistemática sobre los datos reales, utilizando **métricas comparativas de rendimiento** que permiten valorar su eficacia relativa.

Este planteamiento metodológico no sólo se orienta a obtener **resultados precisos**, sino también a comprender **en profundidad el comportamiento diferencial de cada técnica** frente a las características propias del tráfico aeroportuario. El análisis permite, así, detectar **tanto fortalezas como limitaciones** de cada enfoque, generando una base sólida para futuras implementaciones en contextos operativos o estratégicos.

En general, este enfoque de predicción, riguroso y diversificado, refuerza el valor del estudio al combinar **potencia predictiva, interpretabilidad y aplicabilidad práctica**, posicionando al modelo seleccionado como una herramienta eficaz para el **apoyo a la planificación y gestión del transporte aéreo**.

#### 4.1.2. Construcción y estructura del conjunto de representantes

Con el fin de realizar un análisis visual detallado de los representantes seleccionados en cada grupo, se procedió inicialmente a la construcción de un ***dataframe* estructurado** que resume la evolución temporal de las series correspondientes. Esta estructura tabular fue diseñada siguiendo el **formato largo**, ampliamente recomendado en análisis de series temporales por su **flexibilidad para la visualización y modelado**. El *dataframe* generado se compone de tres columnas fundamentales:

- **Entries**: almacena los **valores numéricos observados** en las series temporales seleccionadas.
- **Rep**: identifica el **aeropuerto o entidad representativa** a la que pertenece cada serie, permitiendo su clasificación y agrupación.
- **Date**: contiene las **marcas temporales** asociadas a cada entrada, lo que preserva la dimensión cronológica de los datos.

A partir de esta estructura, se generaron las visualizaciones de los datos.

#### 4.1.3. Análisis visual del comportamiento temporal

La **visualización de series temporales** constituye una herramienta esencial durante la fase exploratoria del análisis, ya que permite identificar de forma directa e intuitiva **patrones, tendencias, estacionalidades y anomalías**. En este estudio, se recurrió a la librería **ggplot2**

del ecosistema `tidyverse` [Val10] para construir un **gráfico de líneas** en el que cada serie representativa de cluster se visualizara con un **color distintivo**. Este enfoque gráfico facilita la comparación visual de la **dinámica temporal** entre series, lo que permite no solo analizar sus trayectorias individuales, sino también realizar una **validación visual del agrupamiento** obtenido mediante el algoritmo jerárquico. Se observan diferencias claras en el comportamiento estructural de las series, particularmente en términos de **variabilidad, intensidad de eventos y regularidad temporal**.

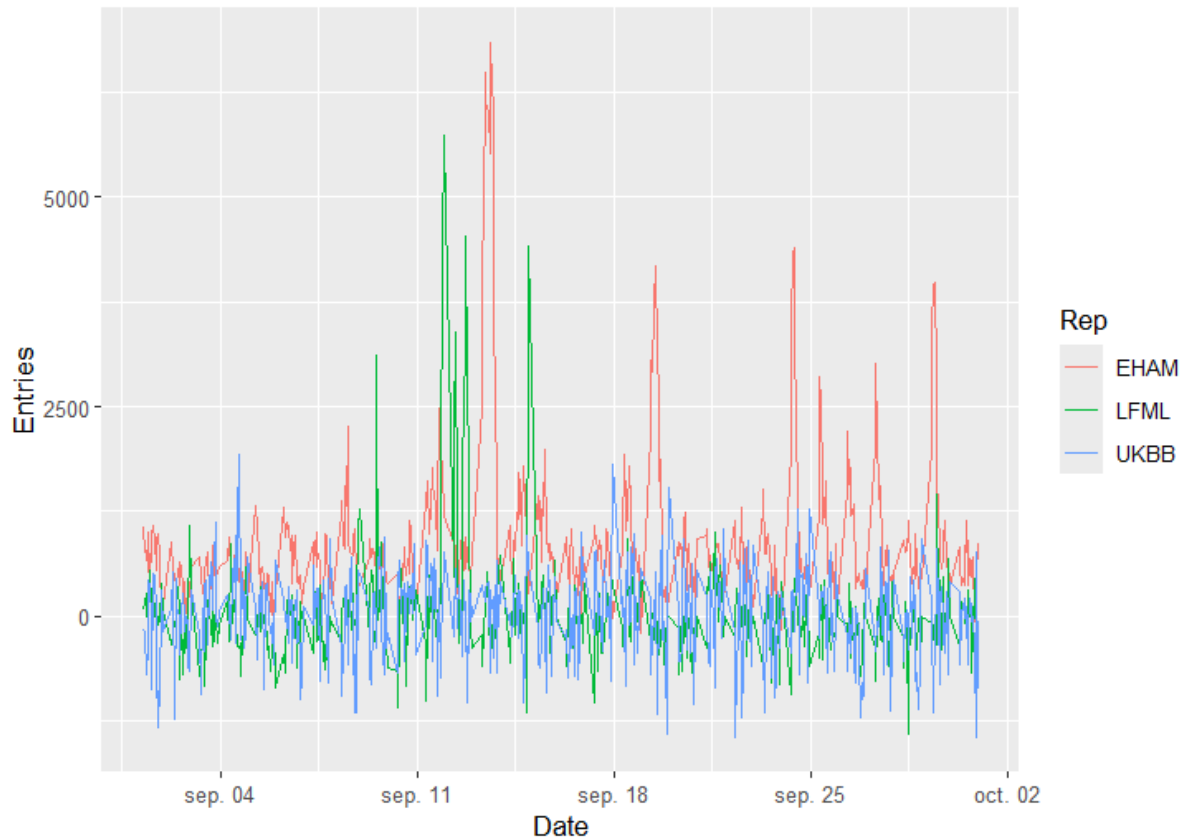


Figura 9: Gráfico de los representantes

Al analizar los resultados obtenidos tras la representación de las series seleccionadas en la *Figura 9*, se identifican diferencias notables en sus comportamientos temporales. En primer lugar, el representante **EHAM** evidencia una **alta variabilidad en la cantidad de entradas**, caracterizada por la presencia de **múltiples picos a lo largo del período analizado**. El **evento más destacado** se localiza en torno al **11 de septiembre**, donde la serie supera las **5500 entradas**, lo cual sugiere una posible **respuesta a un evento externo de gran impacto**. Además, se identifican **otros picos relevantes** en fechas como el **18, 25 y 30 de septiembre**, aunque de menor intensidad, lo que refuerza la idea de una **sensibilidad estructural** ante variaciones o perturbaciones externas.

En segundo lugar, la serie correspondiente al representante **LFML** presenta un **comportamiento análogo** al de EHAM. También muestra un **pico de gran magnitud** en fechas cercanas al **11 de septiembre**, lo cual refuerza la hipótesis de un **evento común** que haya afectado simultáneamente a ambos aeropuertos. No obstante, tras ese suceso, se observa una **tendencia a la estabilización**, aunque con **presencia de picos residuales esporádicos**, lo que podría estar indicando una **variabilidad local** o una **recuperación heterogénea** en el comportamiento del tráfico.

Por último, el representante **UKBB** exhibe un patrón claramente **más estable** en comparación con EHAM y LFML. Aunque se observan **ciertas fluctuaciones**, la **amplitud de oscilación** es sustancialmente más reducida y **ningún valor supera las 2000 entradas**. Esta dinámica sugiere una **dispersión mucho menor**, posiblemente asociada a un **menor volumen de operaciones**, un **entorno más controlado** o una **menor exposición a factores disruptivos**. Desde una perspectiva comparativa, la serie de UKBB puede considerarse como un **modelo de estabilidad estructural**, sirviendo como referencia dentro del conjunto de representantes seleccionados.

#### 4.1.4. Generación y ejecución de modelos de predicción

La ejecución sistemática de un conjunto de **modelos de predicción a corto plazo** permite anticipar el comportamiento futuro de las series temporales. Para cada serie representativa se empleó un enfoque basado en **ventanas móviles**, en el que cada modelo recibe como entrada una secuencia de datos históricos de longitud  $h$  y genera una predicción para el siguiente instante temporal.

Para garantizar la comparabilidad entre modelos, se definió un horizonte de predicción homogéneo de **un día**, y se mantuvo constante la longitud de la ventana histórica en  $h = 7$  días (una semana).

Cada modelo fue ejecutado de forma independiente para cada representante de cluster, y los resultados fueron almacenados en estructuras tabulares que incluyen tanto las predicciones como los valores reales correspondientes. Este diseño permite realizar posteriormente una **evaluación cuantitativa del desempeño** y una comparación cruzada entre algoritmos.

#### 4.1.5. Evaluación de modelos

Para comparar y evaluar la **precisión de los modelos de predicción**, se recurrió a dos métricas de error ampliamente aceptadas en el ámbito del análisis cuantitativo de series temporales: el **Error Cuadrático Medio** (MSE) y el **Error Absoluto Medio** (MAE). Ambas permiten

cuantificar el **desempeño predictivo** midiendo la discrepancia entre los valores observados  $y_i$  y los valores estimados  $\hat{y}_i$  a lo largo de  $n$  observaciones.

- **MSE (Mean Squared Error):**

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2$$

El **MSE** introduce una penalización cuadrática sobre los errores, lo que implica que las desviaciones más grandes influyen desproporcionadamente en la métrica. Esta propiedad lo convierte en una herramienta particularmente útil cuando se desea que el modelo sea **sensible a errores extremos**, y es adecuado en contextos donde se asume que los errores siguen una **distribución normal**. Matemáticamente, representa la **varianza del error residual**, y se asocia con estimaciones optimizadas por mínimos cuadrados [HK06].

- **MAE (Mean Absolute Error):**

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{y}_i - y_i|$$

A diferencia del MSE, el **MAE** aplica una penalización lineal, lo que lo convierte en una métrica **más robusta frente a valores extremos**. Ofrece una medida más realista del error promedio esperado, siendo más interpretable directamente como la desviación promedio respecto al valor reallo que la hace especialmente adecuada en entornos donde se prioriza la **interpretabilidad y la estabilidad frente a ruido o anomalías** [WM05].

En conjunto, estas métricas ofrecen **perspectivas complementarias** sobre la calidad de las predicciones:

- El **MSE**, al amplificar el impacto de errores grandes, favorece modelos que minimizan las desviaciones severas, incluso si esto implica asumir una mayor frecuencia de errores pequeños.
- El **MAE**, al ponderar todos los errores por igual, proporciona una visión más equilibrada y resistente al efecto de valores extremos.
- Desde un enfoque estadístico, el **MAE** se vincula con la **mediana** de los errores absolutos, mientras que el **MSE** se asocia con la **media cuadrática**, lo que tiene implicaciones prácticas en la elección de técnicas de optimización.

- La elección entre ambas métricas dependerá del objetivo predictivo: si se desea evitar errores significativos a toda costa, el **MSE** es más adecuado; si se busca una medida **promedio realista del rendimiento global**, el **MAE** resulta preferible.

En el presente estudio, se emplean **ambas métricas (MSE y MAE)** para evaluar el rendimiento de los modelos de predicción aplicados a las series temporales segmentadas por clusters.

## 4.2. Fundamentación teórica de los modelos

### 4.2.1. ARIMA

El modelo ARIMA (*AutoRegressive Integrated Moving Average*), denotado por  $(p, d, q)$ , constituye una de las metodologías clásicas más empleadas en el análisis y predicción de series temporales univariadas. Su formulación combina tres componentes fundamentales: un término **autorregresivo** (AR) de orden  $p$ , una parte de **integración** (I) de orden  $d$ , y un término de **media móvil** (MA) de orden  $q$ . Se define como:

$$\phi(B)(1 - B)^d y_t = \theta(B)\varepsilon_t$$

En esta expresión,  $B$  denota el **operador de rezago**, tal que  $B^k y_t = y_{t-k}$ ;  $\phi(B) = 1 - \phi_1 B - \dots - \phi_p B^p$  es el **polinomio autorregresivo** de orden  $p$ ;  $\theta(B) = 1 + \theta_1 B + \dots + \theta_q B^q$  es el **polinomio de medias móviles** de orden  $q$ ;  $d$  representa el **número de diferenciaciones necesarias** para alcanzar la estacionariedad en la serie original  $y_t$ ; y  $\varepsilon_t$  es un **proceso de ruido blanco** con media cero y varianza constante  $\sigma^2$ .

El modelo ARIMA se fundamenta en la premisa de que la serie temporal puede transformarse en un proceso **estacionario en media y varianza** mediante diferenciación sucesiva. A partir de dicha transformación, se modelan las dependencias lineales temporales mediante la estructura ARMA subyacente. Este modelo es especialmente adecuado para series que exhiben una dinámica temporal lineal y una estructura de correlación temporal significativa, habiéndose convertido en una herramienta estándar en contextos como la econometría, la demografía o la ingeniería del control [BJR08].

### 4.2.2. Holt-Winters

Este modelo, también llamado *Triple Exponential Smoothing* es una extensión del suavizado exponencial clásico que incorpora explícitamente tres componentes fundamentales: **nivel** (o componente base), **tendencia** y **estacionalidad**. Está especialmente diseñado para series

temporales que presentan una estructura evolutiva compleja, caracterizada por una **tendencia sistemática a lo largo del tiempo** y **patrones estacionales recurrentes** de forma aditiva o multiplicativa.

En su versión aditiva, el modelo se define mediante el siguiente sistema recursivo de ecuaciones:

$$\begin{aligned}\ell_t &= \alpha y_t + (1 - \alpha)(\ell_{t-1} + b_{t-1}) \\ b_t &= \beta(\ell_t - \ell_{t-1}) + (1 - \beta)b_{t-1} \\ s_t &= \gamma(y_t - \ell_t) + (1 - \gamma)s_{t-s} \\ \hat{y}_{t+h} &= \ell_t + hb_t + s_{t+h-s(k+1)}\end{aligned}$$

Donde:

- $\ell_t$  representa el **nivel suavizado** de la serie en el instante  $t$ .
- $b_t$  es el componente de **tendencia local**, que mide la tasa de crecimiento o decrecimiento en el tiempo.
- $s_t$  es el **componente estacional** aditivo con periodicidad  $s$  (por ejemplo,  $s = 12$  para datos mensuales con estacionalidad anual).
- $\alpha, \beta, \gamma \in (0, 1)$  son los **parámetros de suavizado** para el nivel, la tendencia y la estacionalidad, respectivamente.
- $\hat{y}_{t+h}$  es la **predicción** del valor futuro  $h$  pasos adelante.

Este modelo se construye bajo la suposición de que los efectos estacionales son aproximadamente constantes a lo largo del tiempo, aunque también admite una versión multiplicativa para tratar variaciones estacionales proporcionales al nivel de la serie.

Desde una perspectiva estadística, puede interpretarse como un **modelo de espacio de estados no lineal** en el que las actualizaciones de los estados (nivel, tendencia y estacionalidad) se realizan por medio de **filtros exponenciales**. La optimización de los parámetros se lleva a cabo típicamente mediante minimización de una función de pérdida (como el MSE) a lo largo del histórico observado. Su simplicidad estructural y capacidad para modelar componentes sistemáticos lo hacen particularmente útil en entornos operacionales como la logística, la planificación de demanda y el pronóstico financiero [FS16].

### 4.2.3. KNN

Este algoritmo, conocido también como el de los  $K$  vecinos más próximos, es un **método no paramétrico** de aprendizaje supervisado que se adapta al contexto de series temporales bajo el principio de **analogía histórica**. Su fundamento consiste en identificar, dentro del historial de la serie, **ventanas temporales pasadas** que presenten **configuraciones similares** al estado actual, y emplear sus trayectorias subsiguientes como base para generar predicciones.

Sea  $\{y_t\}_{t=1}^T$  una serie temporal y  $\mathbf{x}_t = [y_{t-L+1}, \dots, y_t]$  un vector embebido de longitud  $L$  (ventana de entrada). El procedimiento se estructura en los siguientes pasos:

1. Para el instante actual  $t$ , se define un vector objetivo  $\mathbf{x}_t$  y se calcula la **distancia** (normalmente euclídea o DTW) entre  $\mathbf{x}_t$  y todas las subseries históricas  $\{\mathbf{x}_j\}_{j=1}^{T-L-H+1}$ , donde  $H$  es el horizonte de predicción.
2. Se identifican los  $K$  vectores más próximos, es decir, los índices  $\{j_1, \dots, j_K\}$  tales que:

$$d(\mathbf{x}_t, \mathbf{x}_{j_k}) \leq d(\mathbf{x}_t, \mathbf{x}_j), \quad \forall j \notin \{j_1, \dots, j_K\}$$

3. Se obtienen las trayectorias futuras asociadas a cada vecino:  $\{\mathbf{y}^{(k)} = [y_{j_k+1}, \dots, y_{j_k+H}]\}_{k=1}^K$ .
4. Finalmente, la predicción  $\hat{\mathbf{y}}_{t+1:t+H}$  se construye mediante **promedio** o **ponderación** de dichas trayectorias:

$$\hat{\mathbf{y}}_{t+1:t+H} = \frac{1}{K} \sum_{k=1}^K \mathbf{y}^{(k)}$$

En este estudio, se implementa bajo el paradigma **MIMO** (*Multi-Input Multi-Output*), el cual permite **predecir múltiples pasos hacia adelante de forma simultánea** sin necesidad de recurrir a esquemas recursivos, reduciendo así la acumulación de errores de pronóstico.

Este método presenta ventajas notables en contextos donde la serie posee **dinámicas no lineales**, **regímenes cambiantes** o **estructuras ruidosas difíciles de capturar por modelos paramétricos tradicionales**. Además, al no requerir suposiciones sobre la distribución de los datos, el enfoque KNN resulta altamente flexible y adaptable a diversos dominios [BLH+15].

### 4.2.4. ARNN

ARNN, cuyas siglas provienen de *AutoRegressive Neural Network*, constituye una extensión no lineal del paradigma autorregresivo clásico, basada en una arquitectura de **red neuronal artificial** tipo *perceptrón multicapa* (MLP). Su objetivo es modelar la dependencia dinámica entre los valores actuales y pasados de una serie temporal mediante funciones de activación

no lineales, capturando así **relaciones complejas y patrones no lineales** que escapan a los modelos tradicionales.

Dado un proceso temporal  $\{y_t\}_{t=1}^T$ , el modelo ARNN toma como entrada un conjunto de  $p$  retardos de la serie:

$$\hat{y}_t = f(y_{t-1}, y_{t-2}, \dots, y_{t-p})$$

donde  $f : R^p \rightarrow R$  representa una función de predicción no lineal aproximada por una red neuronal, entrenada mediante algoritmos de retropropagación. La función  $f$  se compone típicamente de una combinación lineal seguida de una función de activación no lineal en cada capa oculta.

La topología del modelo ARNN se caracteriza por:

- **Número de retardos (*lags*):** determina la dimensionalidad del vector de entrada. En este trabajo, se emplea una ventana de tamaño  $p = 17$ , seleccionada empíricamente para capturar la memoria relevante del proceso.
- **Número de neuronas ocultas (*H*):** controla la capacidad expresiva del modelo. En nuestro caso, se utiliza una única capa oculta con  $H = 3$  unidades, lo que permite mantener una buena capacidad de generalización sin incurrir en sobreajuste.

Este enfoque resulta especialmente adecuado para modelar series que presentan **comportamientos no lineales, discontinuidades** o efectos de **interacción compleja entre observaciones pasadas**. Su naturaleza flexible lo convierte en una herramienta eficaz en contextos donde la linealidad asumida por modelos como ARIMA no se sostiene empíricamente [ZPH98].

#### 4.2.5. SVM

Para la predicción de series temporales, se adopta una variante del algoritmo de **Máquinas de Vectores de Soporte** (SVM) diseñada específicamente para tareas de regresión. Esta formulación, conocida como **regresión  $\varepsilon$ -insensible** (SVR), busca encontrar una función de predicción lo más plana posible, tolerando errores dentro de una banda de tamaño  $\varepsilon$  alrededor de los datos observados [SS04].

El modelo se formula como el siguiente problema de optimización convexa:

$$\min_{w, b, \xi_i, \xi_i^*} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*)$$

sueto a:



$$\begin{aligned}
y_i - (w^T x_i + b) &\leq \varepsilon + \xi_i, \\
(w^T x_i + b) - y_i &\leq \varepsilon + \xi_i^*, \\
\xi_i, \xi_i^* &\geq 0, \quad i = 1, \dots, n.
\end{aligned}$$

Donde:

- $w \in R^d$  es el vector de pesos que define el hiperplano de predicción.
- $b \in R$  es el sesgo (intercepto).
- $x_i \in R^d$  representa el vector de características del instante  $i$ .
- $\varepsilon$  define la **zona de tolerancia** dentro de la cual los errores no son penalizados.
- $\xi_i, \xi_i^* \geq 0$  son variables de holgura que permiten errores fuera de la zona  $\varepsilon$ .
- $C > 0$  es un parámetro de regularización que controla el equilibrio entre la **complejidad del modelo** y la **tolerancia al error**.

La solución se puede extender al caso no lineal mediante el uso de **funciones núcleo**  $K(x_i, x_j)$  que permiten proyectar los datos a espacios de alta dimensión sin necesidad de una transformación explícita. Esto es especialmente útil en contextos con relaciones temporales complejas y no lineales.

#### 4.2.6. TBATS

El modelo TBATS constituye una **extensión moderna de los modelos de suavizado exponencial** diseñado específicamente para el tratamiento de series temporales con **estacionalidades múltiples, no enteras y potencialmente no lineales**. Recibe su nombre de *Trigonometric, Box-Cox, ARMA, Trend, Seasonal* e integra de forma coherente varios componentes esenciales para capturar la estructura dinámica de datos complejos [Per22].

La forma general del modelo puede expresarse como:

$$y_t^{(\omega)} = \mu_t + \sum_{k=1}^K \gamma_{k,t} + \varepsilon_t$$

donde:

- $y_t^{(\omega)}$  es la transformación de Box-Cox del dato observado  $y_t$ , aplicada con parámetro  $\omega$  para estabilizar la varianza:

$$y_t^{(\omega)} = \begin{cases} \frac{y_t^\omega - 1}{\omega}, & \omega \neq 0 \\ \log(y_t), & \omega = 0 \end{cases}$$

- $\mu_t$  representa el componente de tendencia, que puede modelarse como una tendencia lineal local o amortiguada:

$$\mu_t = \mu_{t-1} + \phi b_{t-1} + \alpha \varepsilon_t \quad b_t = b_{t-1} + \beta \varepsilon_t$$

con  $\phi$  como parámetro de amortiguamiento de la pendiente.

- $\gamma_{k,t}$  modela la estacionalidad a través de componentes trigonométricos del tipo:

$$\gamma_{k,t} = \sum_{j=1}^{J_k} \left( a_{kj} \cos \left( \frac{2\pi j t}{m_k} \right) + b_{kj} \sin \left( \frac{2\pi j t}{m_k} \right) \right)$$

donde  $m_k$  es el período estacional correspondiente al componente  $k$ , y  $J_k$  el número de pares seno-coseno utilizados.

- $\varepsilon_t$  sigue una estructura ARMA( $p, q$ ), es decir:

$$\varepsilon_t = \sum_{i=1}^p \phi_i \varepsilon_{t-i} + \sum_{j=1}^q \theta_j \eta_{t-j} + \eta_t$$

lo cual permite capturar autocorrelaciones residuales no explicadas por los componentes estructurales.

La principal fortaleza del modelo TBATS reside en su **capacidad para manejar múltiples ciclos estacionales**, incluso cuando estos no son múltiplos enteros del horizonte temporal. Además, la inclusión de la transformación de Box-Cox y el componente ARMA contribuyen a su **robustez frente a heterocedasticidad y dependencia estructural residual**, respectivamente.

Este enfoque resulta particularmente útil en contextos donde las series presentan **variabilidad estacional compleja, efectos de calendario o estructuras periódicas irregulares**, características comunes en aplicaciones como el tráfico aéreo, demanda energética o flujos de red.

#### 4.2.7. LSTM

Su nombre hace referencia a *Long Short-Term Memory* y se trata de una **red neuronal recurrente** diseñada para **capturar dependencias de largo plazo** en secuencias temporales. La clave de su arquitectura reside en la incorporación de **puertas de entrada, olvido y salida**, las cuales regulan dinámicamente el flujo de información para **preservar o descartar datos relevantes** según el contexto temporal. Estas puertas se definen mediante las siguientes expresiones matemáticas:

$$\begin{aligned}
f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\
i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\
\tilde{C}_t &= \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\
C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\
o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\
h_t &= o_t * \tanh(C_t)
\end{aligned}$$

Aquí,  $\sigma$  representa la función que controla la activación de las puertas, mientras que  $\tanh$  aporta no linealidad al modelo. Los vectores  $h_t$  y  $C_t$  son el estado oculto y la memoria celular en el tiempo  $t$ , respectivamente. La combinación de estas componentes permite a la LSTM **manejar dependencias temporales no lineales y retardadas** de manera eficiente, superando las limitaciones de las redes recurrentes tradicionales.

Gracias a esta capacidad, las LSTM se posicionan como uno de los modelos más potentes y flexibles para tareas de **predicción y modelado de series temporales complejas** [HS97].

En cuanto a su implementación práctica, en el entorno R se emplea el paquete `tensorflow` mediante la interfaz `reticulate`, la cual facilita la integración y gestión de entornos de Python dentro del ecosistema R, ampliando así las capacidades computacionales disponibles para el análisis estadístico avanzado.

#### 4.2.8. Combinación de modelos

Para cada cluster, una vez elegidos los tres mejores métodos, se pueden **combinar los pronósticos generados** con el fin de **mejorar la precisión predictiva** con diversas técnicas, empleando el paquete `ForecastComb` [LK23].

Utilizar la combinación de predicciones no es solo una opción, sino que constituye una **metodología avanzada que potencia significativamente la capacidad predictiva del proyecto**. Esta estrategia permite superar las limitaciones inherentes a los modelos individuales, ya que al integrar la información proveniente de diferentes perspectivas de modelado, se obtiene una **visión más completa y robusta** de la compleja dinámica de los retrasos aeroportuarios.

Como consecuencia directa, esta integración se traduce en **predicciones más fiables y precisas**, lo cual es **fundamental para la toma de decisiones operacionales** en un entorno altamente dinámico y sensible como el aeroportuario. Así, la combinación de modelos no solo mejora la exactitud, sino que también aporta mayor confianza y estabilidad en las estimaciones, elementos cruciales para optimizar recursos y planificar estrategias en tiempo real.

La **motivación teórica fundamental** que sustenta esta estrategia se basa en que **cada modelo captura diferentes aspectos** del proceso generador de datos subyacente. En consecuencia, su combinación puede **reducir la varianza del error total** y **compensar posibles sesgos individuales** de los modelos, mejorando así las predicciones finales.

Matemáticamente, partiendo de  $M$  modelos para el tiempo  $t$ , la combinación de pronósticos se define generalmente como un promedio ponderado:

$$\hat{y}_t = \sum_{i=1}^M w_i \hat{y}_{i,t}$$

donde el vector de pesos  $\mathbf{w} = (w_1, w_2, \dots, w_M)$  satisface

$$w_i \geq 0, \quad \sum_{i=1}^M w_i = 1.$$

Los métodos que se emplean en el presente análisis son:

- **Media Aritmética Simple (SA)**: asigna **pesos iguales** a cada modelo, es decir,  $w_i = \frac{1}{M}$ .

Matemáticamente:

$$\hat{y}_t^{SA} = \frac{1}{M} \sum_{i=1}^M \hat{y}_{i,t}.$$

Este enfoque es **bueno en errores extremos** de modelos individuales, ya que no depende de la calidad relativa de cada uno. Sin embargo, puede ser menos aconsejable si algunos modelos son consistentemente mejores que otros.

- **Método Bates y Granger (BG)** [BG69]: optimiza los pesos asignándolos en función de la **matriz de varianza-covarianza** de los errores históricos de los modelos. La idea es otorgar **mayor peso a modelos con menor varianza** y menor correlación con otros errores, buscando minimizar la varianza del error combinado. Formalmente, los pesos se calculan como:

$$\mathbf{w}^{BG} = \frac{\Sigma^{-1} \mathbf{1}}{\mathbf{1}^\top \Sigma^{-1} \mathbf{1}},$$

donde  $\Sigma$  es la matriz de covarianza de los errores de los modelos y  $\mathbf{1}$  es un vector columna de unos. Este método es **óptimo en sentido de mínima varianza** bajo la suposición de que los errores son normales y no sesgados, y permite aprovechar la información conjunta sobre precisión y correlación entre modelos.

- **Mínimos Cuadrados con Restricciones (CLS)**: consiste en encontrar el vector de pesos que minimiza el **error cuadrático medio** entre la combinación y las observaciones

reales, sujeto a las restricciones de no negatividad y suma unitaria:

$$\min_w \sum_{t=1}^T \left( y_t - \sum_{i=1}^M w_i \hat{y}_{i,t} \right)^2 \quad \text{s.a.} \quad w_i \geq 0, \quad \sum_{i=1}^M w_i = 1,$$

donde  $y_t$  es el valor observado en el tiempo  $t$  [GPP16]. Se suele utilizar para **ajustar los pesos en función de la calidad predictiva real** de cada modelo, asignando mayor peso a aquellos con menor error histórico, y es computacionalmente eficiente.

En general, la combinación de pronósticos se justifica teóricamente en que **ningún modelo individual es capaz de capturar completamente la complejidad del proceso generador de datos**, especialmente en contextos multifacéticos y no lineales como la dinámica de los retrasos aeroportuarios.

Cada modelo tiene un **enfoque particular**, por ejemplo, la **SVM** modela relaciones no lineales mediante márgenes máximos, el **ARNN** incorpora memoria para patrones secuenciales y no lineales, y el **ARIMA** modela la estacionalidad y autocorrelación lineal. Al combinar sus predicciones, se integran estas diferentes perspectivas o inteligencias modelísticas.

Esto permite:

- **Reducir la varianza del error total:** la combinación actúa como un suavizado, ya que es poco probable que todos los modelos cometan errores grandes y en la misma dirección simultáneamente.
- **Compensar sesgos individuales:** los sesgos de un modelo pueden ser contrarrestados por otro modelo que enfoque la dinámica desde una óptica distinta.

## 4.3. Aplicación de modelos al conjunto de datos reales

### 4.3.1. Comparación inicial de resultados

La implementación práctica de los modelos propuestos requirió una ejecución total aproximada de una hora. Durante este proceso, se aplicó una estrategia basada en la extracción de **segmentos consecutivos de 7 días**, a partir de los cuales se realizaron predicciones sobre las **17 observaciones correspondientes al día siguiente**. Este enfoque permitió evaluar empíricamente la **capacidad predictiva de cada modelo** en contextos temporales acotados, así como analizar su rendimiento frente a series con distintas estructuras y patrones internos. De este modo, se facilitó una **elección informada y fundamentada del modelo más adecuado** para cada grupo de series, en función de sus características específicas y del comportamiento observado en las fases de validación.

Al calcular las dos métricas de error ya introducidas previamente, se obtienen los siguientes resultados para el MAE:

Modelo	Cluster 1	Cluster 2	Cluster 3
ARIMA	758.14	570.58	403.20
HOLT	867.11	884.62	405.48
KNN	973.60	595.91	459.28
ARNN	1096.08	478.59	416.02
SVM	868.03	<b>361.46</b>	<b>364.89</b>
TBATS	<b>622.68</b>	566.59	373.43
LSTM	743.48	562.35	412.86

Cuadro 1: MAE promedio de cada modelo por cluster

Como se observa en la *Tabla 1*, los valores más bajos de MAE por cluster y, por tanto, los mejores modelos para cada uno, corresponden a:

- **Cluster 1:** el modelo **TBATS** presenta el menor error absoluto medio, con un valor de 622.68, seguido por los modelos LSTM y ARIMA. Este resultado sugiere que TBATS se adapta especialmente bien a las características estructurales de las series agrupadas en este clúster.
- **Cluster 2:** en este grupo, el mejor desempeño lo alcanza el modelo **SVM**, con un MAE de 361.46. Este rendimiento superior apunta a que las series temporales contenidas en este clúster presentan **relaciones no lineales** entre sus observaciones, las cuales son mejor capturadas por un enfoque basado en máquinas de soporte vectorial.
- **Cluster 3:** nuevamente, el modelo **SVM** muestra el mejor desempeño, alcanzando un MAE de 364.89, superando al resto de los modelos evaluados. Este resultado refuerza la **capacidad de generalización de la SVM** y su efectividad en contextos donde los datos presentan cierta regularidad, pero sin patrones estacionales o de tendencia marcados.

A continuación, se analizan los valores del **error cuadrático medio (MSE)**, que complementan el análisis anterior al penalizar en mayor medida los errores grandes, proporcionando así una medida más sensible al comportamiento extremo de los modelos.

Modelo	Cluster 1	Cluster 2	Cluster 3
ARIMA	1569217	1150100	266547
HOLT	1700984	5449426	271969
KNN	2608918	916,926	335861
ARNN	2721310	606,506	285287
SVM	11072929	<b>381232</b>	<b>227009</b>
TBATS	<b>1218484</b>	1123774	241919
LSTM	1482774	1225850	273796

Cuadro 2: MSE promedio de cada modelo por cluster

Según la *Tabla 2*:

- **Cluster 1:** el modelo **TBATS** también presenta el menor MSE con 1218484, seguido por ARIMA. Cabe destacar que el SVM obtiene un MSE considerablemente alto, lo que indica la presencia de algunos errores extremos que afectan su desempeño en este grupo.
- **Cluster 2:** el mejor modelo según esta métrica es nuevamente el **SVM** con un MSE de 381232. Este resultado es coherente con el MAE y confirma que el modelo es consistente en su capacidad predictiva para este cluster.
- **Cluster 3:** **SVM** logra el menor MSE con 227009, seguido por TBATS. Esto demuestra una vez más la eficacia de SVM en los grupos donde las relaciones no lineales son predominantes y donde se logra evitar errores extremos.

Del análisis comparativo realizado, se puede concluir que el modelo **SVM** resulta ser **particularmente eficaz** en los **clusters 2 y 3**, donde las series temporales presentan **patrones predominantemente no lineales** y una menor presencia de ruido estructural. Su capacidad para capturar relaciones complejas mediante funciones **kernel** le otorga una ventaja significativa frente a métodos lineales tradicionales.

Por otro lado, el modelo **TBATS** destaca claramente en el **cluster 1**, lo que sugiere que las series agrupadas en este conjunto presentan **estacionalidades múltiples o complejas**, así como una estructura temporal más sofisticada. La arquitectura del modelo TBATS, al incorporar transformaciones Box-Cox, componentes armónicos y tendencias dinámicas, le permite adaptarse con mayor precisión a este tipo de configuraciones.

En contraste, los modelos tradicionales como **ARIMA** y **HOLT** muestran un **rendimiento aceptable**, pero son consistentemente superados por **enfoques más flexibles y modernos**

como SVM y TBATS, los cuales logran capturar con mayor fidelidad las distintas dinámicas subyacentes en los datos. Esto evidencia la relevancia de emplear metodologías avanzadas que se ajusten a la complejidad del proceso generador de los retrasos aeroportuarios, especialmente cuando se trabaja con datos heterogéneos y segmentados.

Finalmente, se construyeron representaciones gráficas mediante **diagramas de cajas (*box-plots*)** con el objetivo de visualizar la **distribución de los errores absolutos medios** correspondientes a cada modelo dentro de cada clúster  $j$ , con  $j \in \{1, 2, 3\}$ . Para favorecer una presentación más limpia y evitar la distorsión visual causada por valores extremos, se utilizó el argumento `outline = FALSE`, el cual **suprime la visualización de los outliers** en los gráficos.

De forma análoga, se generaron también diagramas de caja que representan la **distribución del error cuadrático medio** para los distintos modelos en cada cluster, proporcionando así una visión complementaria que **penaliza de forma más severa los errores grandes** y permite contrastar el rendimiento desde una perspectiva más conservadora.

Estas visualizaciones permiten realizar una **evaluación más robusta y comparativa** del comportamiento predictivo de cada modelo en función de la estructura de los datos agrupados, facilitando la interpretación de su rendimiento global y su adecuación a las características particulares de cada subconjunto de series.

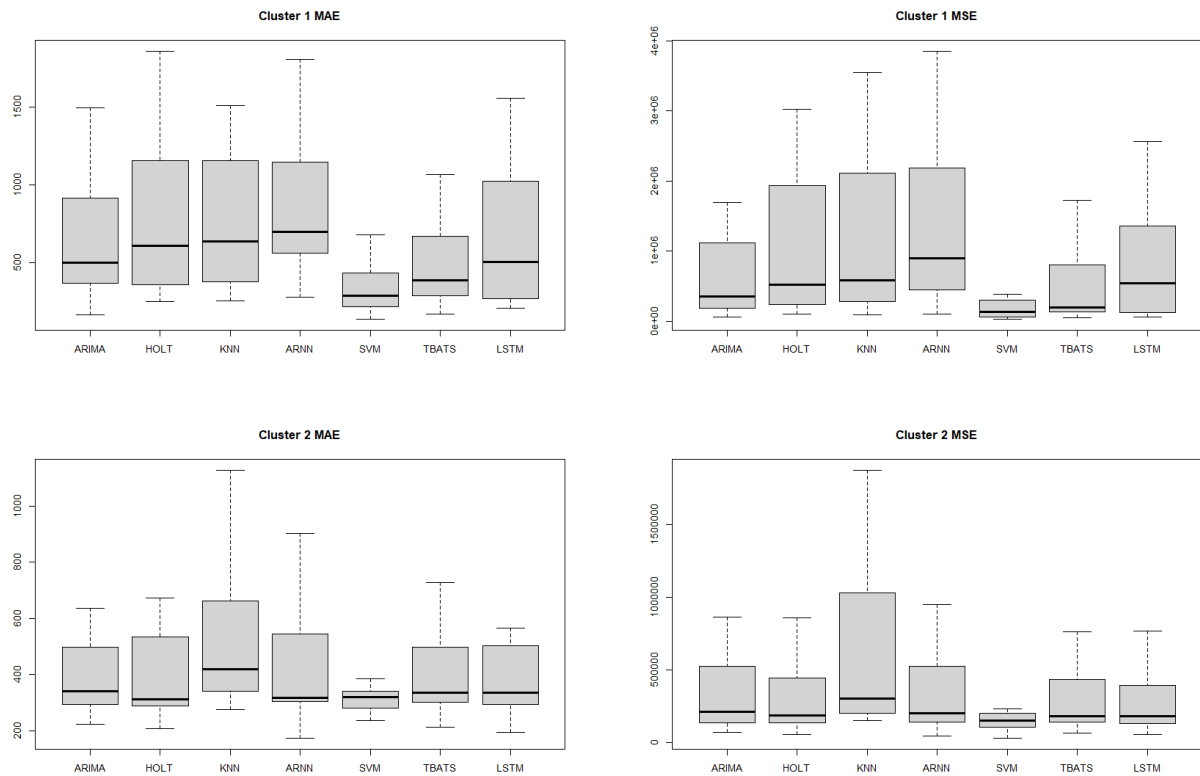


Figura 10: Boxplots del MAE y del MSE (clusters 1 y 2)



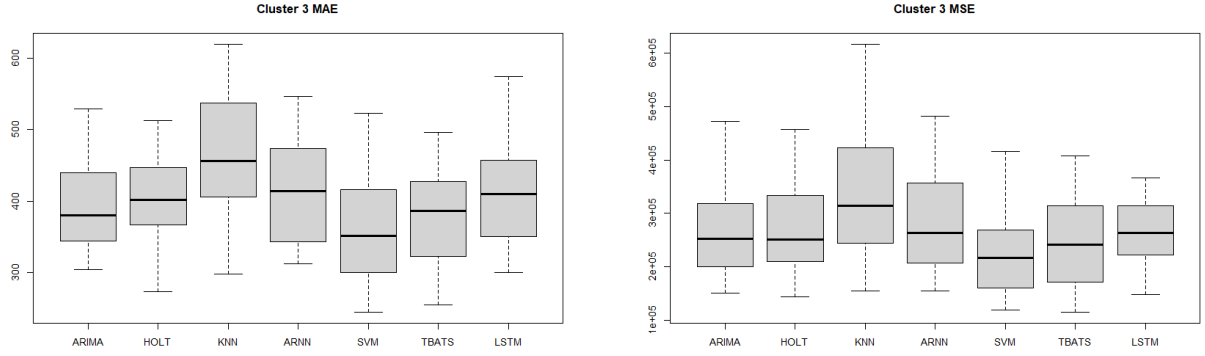


Figura 11: Boxplots del MAE y del MSE (cluster 3)

Al analizar las representaciones gráficas presentadas en la *Figura 10* y la *Figura 11*, se pueden extraer conclusiones relevantes sobre el rendimiento comparativo de los modelos en cada clúster.

En el **Cluster 1**, el modelo **TBATS** destaca como el más preciso, presentando los **valores más bajos de error** tanto en términos de **MAE** como de **MSE**. Le sigue en rendimiento el modelo **SVM**, mientras que otros enfoques como **KNN** y **ARNN** muestran **errores más elevados y una mayor dispersión**, lo que indica menor estabilidad y capacidad de ajuste a las series contenidas en este grupo.

En el **Cluster 2**, el comportamiento de los modelos varía ligeramente. El modelo **SVM** presenta nuevamente el **mejor desempeño**, con los **menores valores de MAE y MSE**, lo que reafirma su eficacia para capturar relaciones no lineales en entornos de baja varianza. El modelo **TBATS** también ofrece **resultados competitivos**, aunque con una dispersión ligeramente superior en el error cuadrático medio. Los modelos tradicionales como **ARIMA** y las redes neuronales como **ARNN** muestran un **rendimiento intermedio**, razonable pero menos sobresaliente.

Para el **Cluster 3**, se mantiene la tendencia observada previamente: los modelos **SVM** y **TBATS** continúan posicionándose como los más efectivos, mostrando los **errores más bajos y menor dispersión**. En este grupo, **ARIMA**, **HOLT** y **ARNN** ocupan posiciones intermedias, mientras que el modelo **KNN** nuevamente exhibe el **peor desempeño**, confirmando su limitada capacidad de adaptación en contextos más suaves o no lineales.

En conjunto, los modelos **SVM** y **TBATS** se consolidan como los **más consistentes** a lo largo de los diferentes clusters, lo cual sugiere una **mayor capacidad de generalización y adaptación** a distintas configuraciones estructurales de series temporales. Esta evidencia empírica respalda su elección preferente en contextos de alta heterogeneidad y complejidad temporal.

Para facilitar el análisis cuantitativo, los errores absolutos medios se organizaron en un **marco de datos** (*dataframe*) denominado `df.mae`, cuya estructura es la siguiente:

- **MAE:** valor del error absoluto medio correspondiente a una predicción individual sobre una serie temporal.
- **Method:** modelo de predicción utilizado (ARIMA, HOLT, KNN, ARNN, SVM, TBATS, LSTM).
- **Cluster:** cluster al que pertenece la serie temporal evaluada.

Este *dataframe* se construye **concatenando los resultados de los tres clusters**, asignando a cada valor de MAE su correspondiente etiqueta de método y cluster. Estos resultados respaldan la necesidad de aplicar enfoques híbridos y flexibles, adaptados a las características específicas de cada grupo de series temporales.

#### 4.3.2. Predicciones a futuro

Con el objetivo de evaluar la **capacidad predictiva** de los modelos presentados anteriormente, se procede a un **análisis individualizado por clúster**, tomando como referencia las agrupaciones obtenidas en la etapa de segmentación. Esta aproximación permite examinar el rendimiento de cada modelo en contextos diferenciados, caracterizados por patrones estructurales específicos.

Este **análisis detallado a nivel de cluster** no solo facilita la identificación de los **modelos con mayor precisión** en distintos escenarios, sino que también permite detectar **patrones comunes** en el comportamiento de las series pertenecientes a cada agrupación. En consecuencia, se obtiene una visión más profunda de la **dinámica temporal del tráfico aeroportuario**, lo que contribuye a establecer una **base metodológica sólida** para la toma de decisiones fundamentadas en predicciones más **fiables y adaptadas** a las características particulares de cada segmento de datos.

##### ▪ Análisis del cluster 1

Se selecciona el cluster  $i = 1$  y el histórico  $j = 5$ , correspondiente a **datos comprendidos entre viernes y jueves**, con **predicción del siguiente viernes**. Así, los errores de predicción para los distintos modelos se resumen en la siguiente tabla:

Modelo	MAE	MSE
ARIMA	1100.18	1265292.8
HOLT	1537.41	2497725.5
KNN	1220.53	1678895.4
ARNN	994.82	1210553.6
SVM	<b>185.73</b>	<b>57101.2</b>
TBATS	1261.65	1721901.2
LSTM	258.65	97410.88

Cuadro 3: Errores MAE y MSE para el histórico 5 del cluster 1

Según la *Tabla 3*, el modelo **SVM** evidencia un **desempeño superior** en el análisis del histórico, registrando los **errores más bajos** tanto en el **error absoluto medio** como en el **error cuadrático medio**. A continuación, se posiciona el modelo **ARNN**, que también presenta resultados competitivos. En contraste, el modelo **Holt-Winters** exhibe el **rendimiento más deficiente** bajo ambas métricas evaluadas.

Estos resultados indican que los **modelos no lineales** y aquellos basados en técnicas avanzadas de **aprendizaje automático** son considerablemente más efectivos para **capturar la compleja estructura temporal** que caracteriza a esta serie de datos. La superioridad del modelo **SVM** y, en segundo lugar, del **ARNN**, sugiere que la flexibilidad y capacidad de modelado no lineal de estos métodos les permite adaptarse mejor a patrones dinámicos y variaciones que los modelos tradicionales, como **Holt-Winters**, no logran representar con igual precisión. Esto resalta la importancia de seleccionar modelos adecuados a la naturaleza específica de los datos para maximizar la capacidad predictiva.

A continuación, se presentan las visualizaciones comparativas entre las **predicciones generadas** para el histórico 5 y sus correspondientes **valores observados reales**. Estas gráficas permiten una evaluación visual complementaria al análisis cuantitativo, facilitan la identificación de desviaciones, tendencias y posibles sesgos en los pronósticos de cada modelo. Este tipo de representación es fundamental para validar la adecuación del modelo en contextos aplicados, donde la interpretación intuitiva de los resultados es clave para la toma de decisiones operativas y estratégicas.

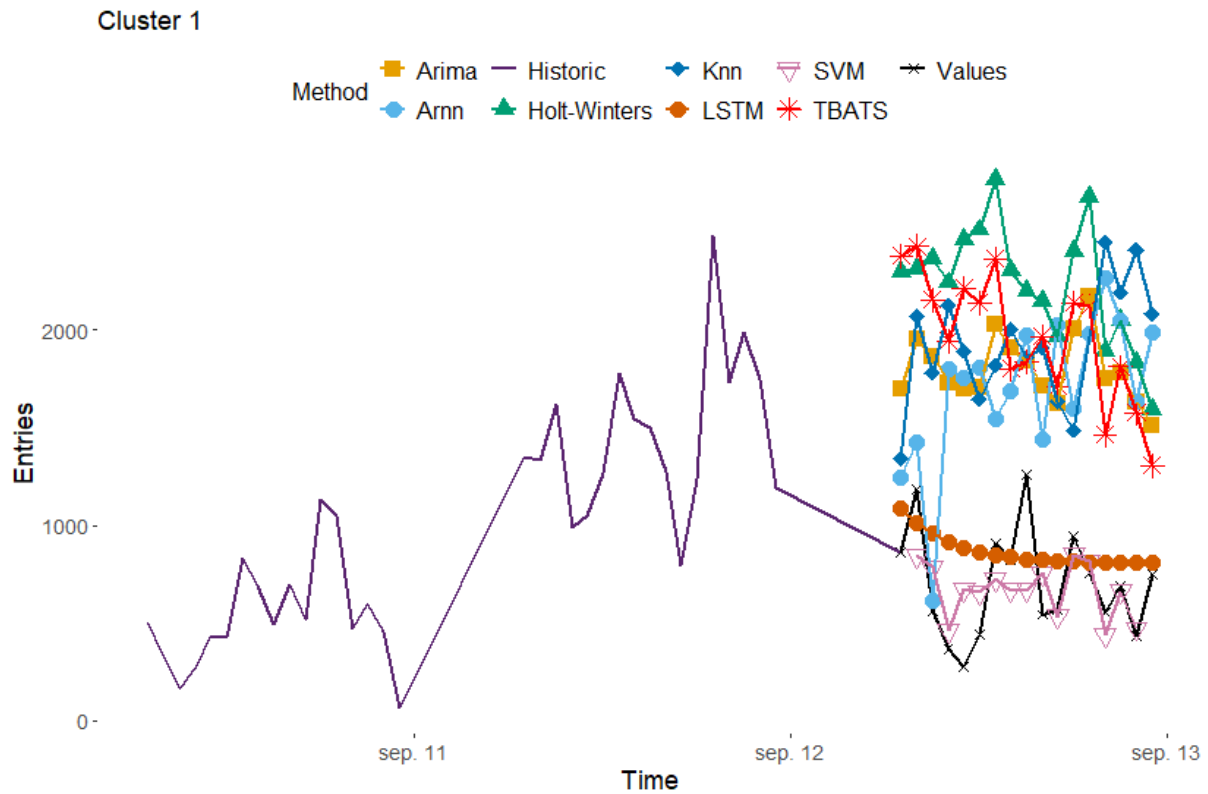


Figura 12: Gráfico de días de datos reales frente a predicción del cluster 1

La *Figura 12* confirma todo lo comentado antes tras el análisis de las evidencias numéricas. Se observa que los modelos basados en **redes neuronales (LSTM y ARNN)** exhiben consistentemente los errores más bajos en ambas métricas, lo que indica un ajuste superior a los datos y una mayor capacidad para capturar patrones complejos. En contraste, modelos como **ARIMA**, **KNN** y **SVM** muestran errores promedio significativamente más altos, sugiriendo un rendimiento inferior para la predicción de esta serie temporal. Los modelos **TBATS** y **Holt-Winters** se sitúan en un rango intermedio, demostrando un rendimiento sólido para series con estacionalidad, aunque no alcanzan la precisión de los modelos neuronales. Este análisis cuantitativo valida la superioridad de los enfoques de *Deep Learning* para esta tarea.

Detallando ahora las predicciones horarias para ese día en cuestión, se obtiene la *Figura 13*, donde se puede apreciar lo analizado más detallado:

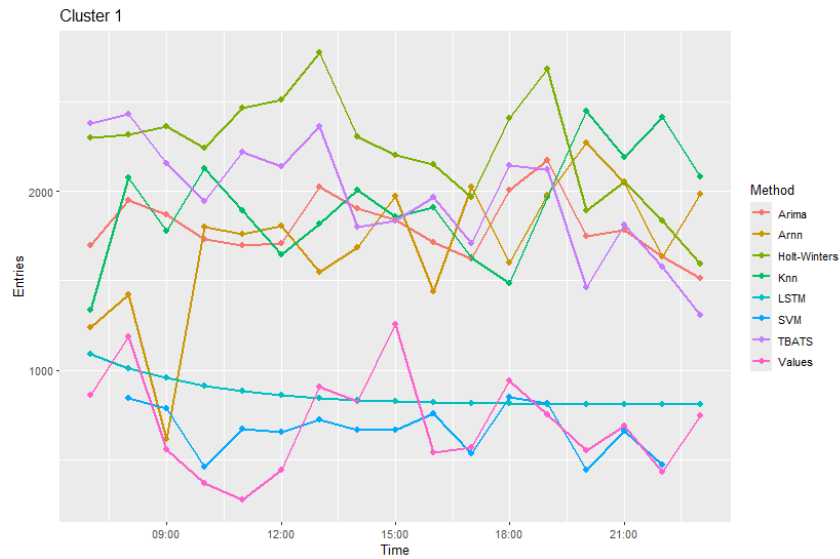


Figura 13: Gráfico de horas de datos reales frente a predicción del cluster 1

Tras comprobar que encajan los valores numéricos con lo visto en los gráficos, se construyó una matriz de predicciones con los métodos que proporcionan un valor más aproximado al real, siendo estos **SVM**, **LSTM** y **ARNN**:

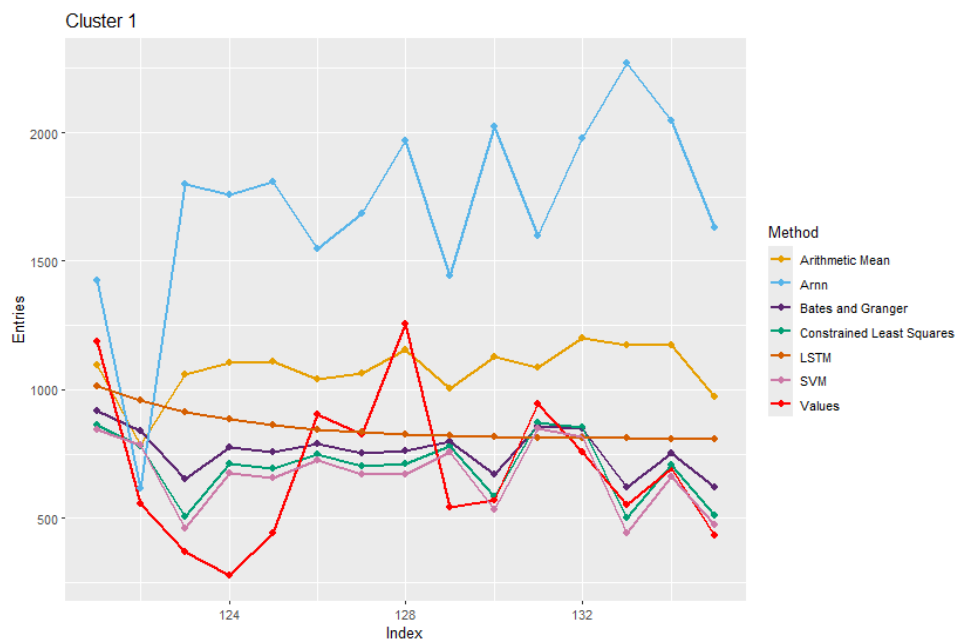


Figura 14: Gráfico de predicciones combinadas del cluster 1

En la *Figura 14* se ve que **CLS**, **BG** y **SVM** van bastante alineados, pero se sacan los errores para verificar cuál funciona mejor.

Modelo	MAE	MSE
Arithmetic Mean (AM)	415.20	226881.70
Bates and Granger (BG)	212.13	65358.27
Constrained Least Squares (CLS)	<b>183.27</b>	<b>55924.33</b>

Cuadro 4: Errores MAE y MSE para la predicción combinada del cluster 1

Como la *Tabla 4* muestra que los menores errores se dan en el caso de la regresión por mínimos cuadrados, se toma este método junto con el SVM como posibles opciones, aunque valorando los errores de ambos se decide sacar las predicciones con la **combinación por el método de mínimos cuadrados**.

Se compara el día que se está prediciendo con los valores reales.

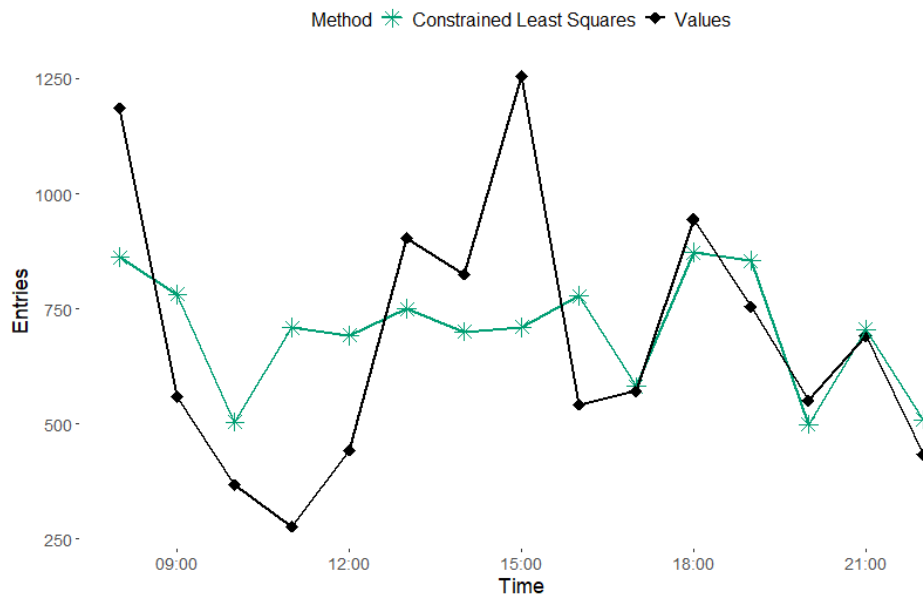


Figura 15: Gráfico de predicciones por el método CLS del cluster 1

La *Figura 15* muestra que hay margen de mejora en las partes de picos más pronunciados, principalmente, pero se ajusta bastante bien en la parte final del gráfico, **siguiendo ambas un patrón similar**.

Por último, se obtiene la **predicción fuera de muestra** para el primer día de octubre de ese mismo año con el método elegido, con el objetivo de **evaluar la capacidad de generalización de los modelos**. A diferencia de la predicción dentro de muestra, donde el modelo estima sobre datos que ya ha visto, la predicción fuera de muestra permite comprobar su comportamiento sobre datos nuevos, no utilizados durante el entrenamiento. Esta

evaluación es esencial para **validar que el modelo no ha sobreajustado** (*overfitting*) y que puede ofrecer **pronósticos fiables** en escenarios reales.

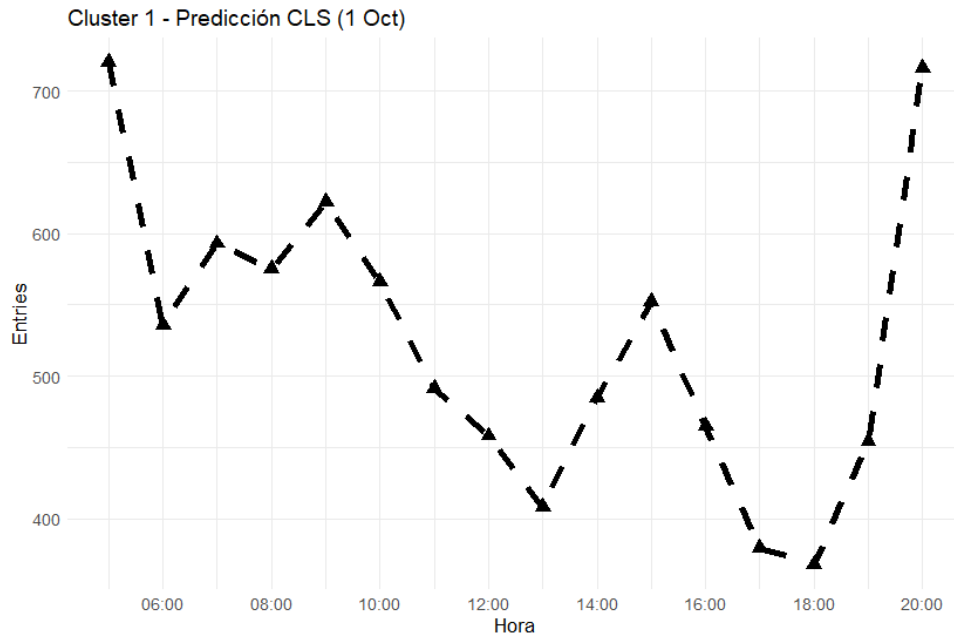


Figura 16: Gráfico de predicciones para el 1 de octubre en el cluster 1

Los valores que muestra la *Figura 16*, presentan **varios picos**. Comienzan con un número de entradas mayor a 700, van decreciendo en general y después tienen un pico de crecimiento y otro de decrecimiento más hasta volver a subir a la posición inicial más o menos.

#### ■ **Análisis del cluster 2:**

Se sigue el **procedimiento análogo para este representante**, seleccionando el cluster  $i = 2$  y el histórico  $j = 5$ . Los errores que se obtienen son:

Modelo	MAE	MSE
ARIMA	3758.35	16140704
HOLT	9856.00	111865881
KNN	1407.82	3131977
ARNN	1745.71	4348640
SVM	<b>998.60</b>	<b>1608639</b>
TBATS	3750.35	16080634
LSTM	4078.12	18629458

Cuadro 5: Errores MAE y MSE para el histórico 5 del cluster 2

Según la *Tabla 5*, el modelo **SVM** presenta los menores errores MAE y MSE, siendo **la opción más acertada para este cluster**. **KNN** y **ARNN** también ofrecen un **buen desempeño**, mientras que **Holt-Winters** obtiene los **peores resultados**.

Tras este análisis numérico, se obtienen las **gráficas de la predicción**. Primero, una con la predicción más general del día y sus valores históricos previos y, después, una con el detalle por horas:

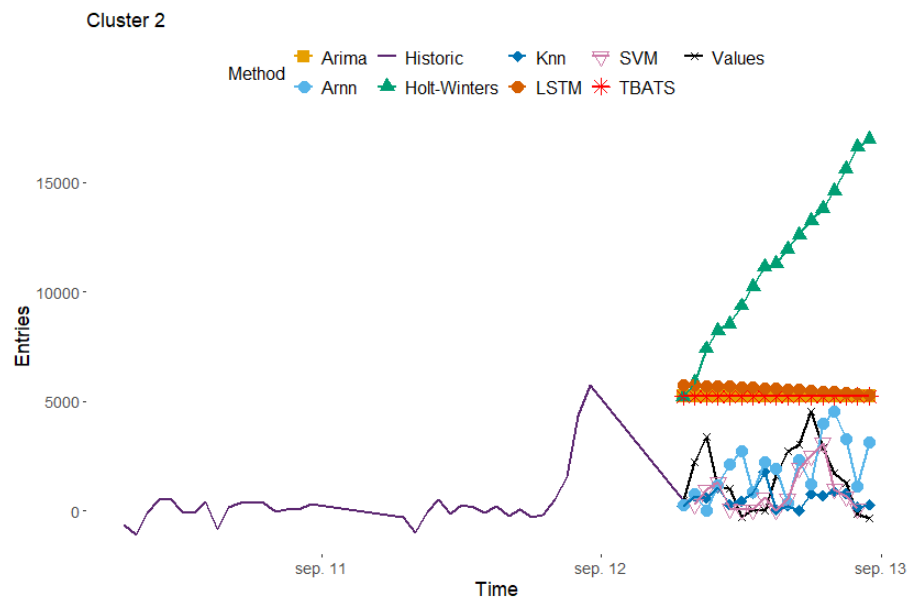


Figura 17: Gráfico de días de datos reales frente a predicción del cluster 2

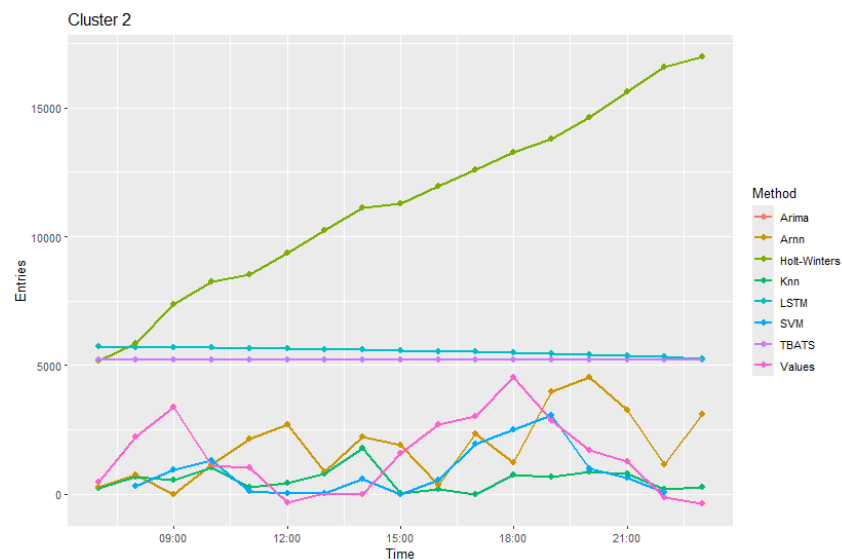


Figura 18: Gráfico de horas de datos reales frente a predicción del cluster 2

En base a la Figura 17 y a la Figura 18, tras un análisis detallado, **se descartan ARIMA, Holt-Winters, LSTM y TBATS**, ya que no funcionan bien para este grupo. Esto encaja



con lo analizado numéricamente en base a los errores, por lo que se probará con los modelos SVM, KNN y ARNN y sus combinaciones por si mejorasen las predicciones. Visualizando las combinaciones de estas, se obtiene la Figura 19.

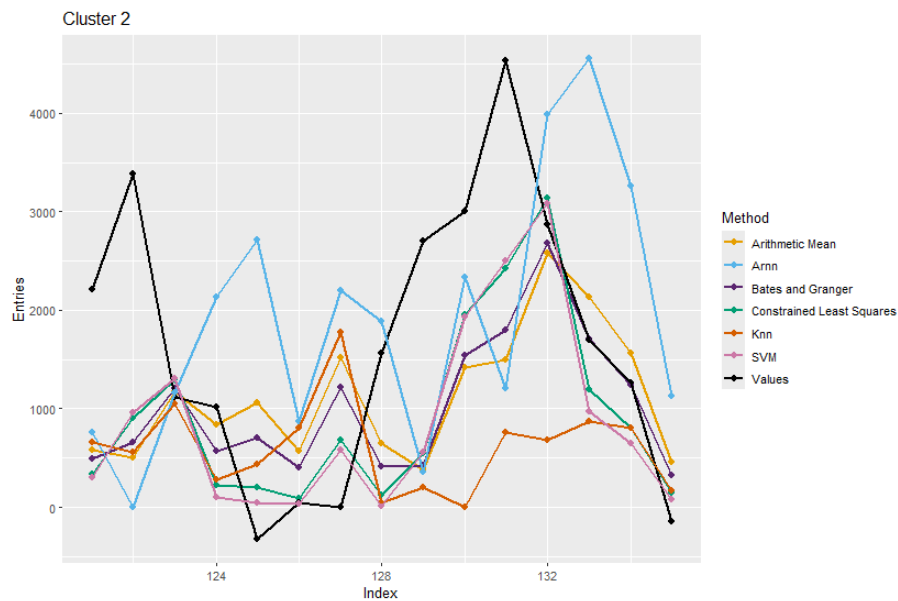


Figura 19: Gráfico de predicciones combinadas del cluster 2

Visualmente, se ve que el **SVM** y **CLS** son los mejores modelos. Parece que los resultados son mejores al combinar varias muestras, lo que se comprueba calculando el MAE y el MSE.

Modelo	MAE	MSE
Arithmetic Mean (AM)	1178.27	2282163
Bates and Granger (BG)	1059.80	1974388
Constrained Least Squares (CLS)	<b>990.53</b>	<b>1598080</b>

Cuadro 6: Errores MAE y MSE para la predicción combinada del cluster 2

Así, analizando la *Tabla 6*, se aprecia que, aunque el mejor modelo simple era SVM, empleando la **regresión por el método de los mínimos cuadrados** y metiendo el resto de variables, se consigue mejorar un poco. Se visualiza cómo quedaría este mejor modelo dentro de muestra.

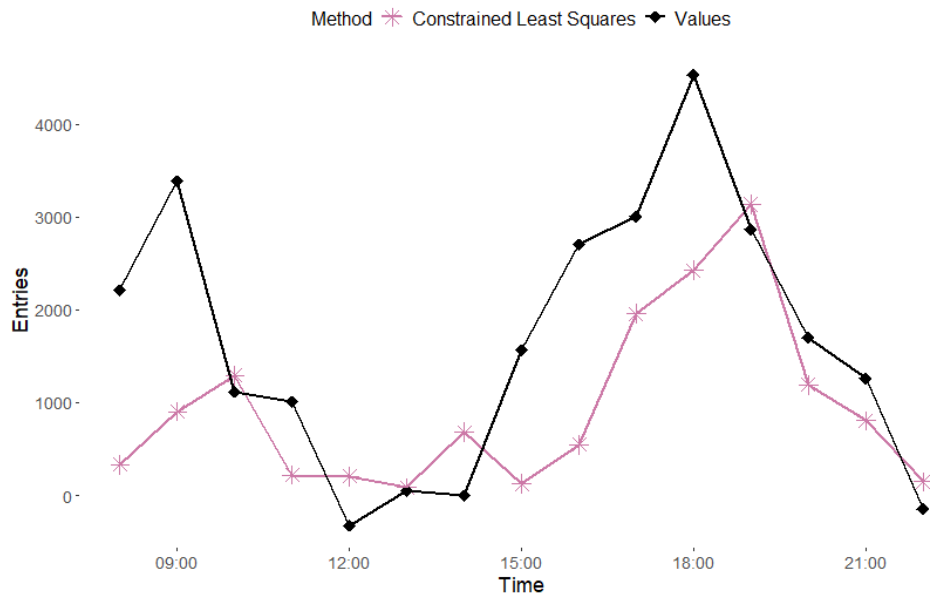


Figura 20: Gráfico de predicciones por el método CLS del cluster 2

Al estudiar la *Figura 20*, los resultados son buenos, aunque no detecta todos los picos bien y en algunos casos predice los casos con un **desplazamiento entre horas**. Para el siguiente día fuera de muestra la predicción quedaría así:

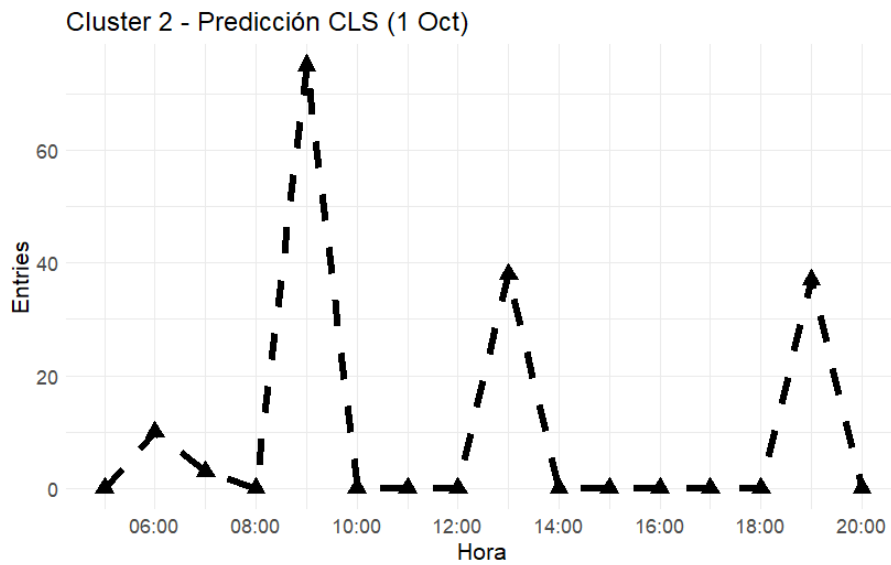


Figura 21: Gráfico de predicciones para el 1 de octubre en el cluster 2

Según la predicción para el día 1 de octubre de la *Figura 21*, no se darán picos bruscos de descenso, sino que **partirá de 0 e irá subiendo y bajando a 0 de nuevo según distintos picos de intensidad a lo largo del día**. Se observa también un **periodo plano**, sin entradas.

■ **Análisis del cluster 3:**

Como en el cluster 1 y en el 2, se sigue el mismo análisis, cuyos errores resultantes son:

Modelo	MAE	MSE
ARIMA	304.06	150807.2
HOLT	441.41	281934.0
KNN	492.94	324825.2
ARNN	325.65	192073.3
SVM	298.53	155185.5
TBATS	<b>270.18</b>	<b>133387.1</b>
LSTM	307.65	153588.9

Cuadro 7: Errores MAE y MSE para el histórico 5 del cluster 3

En ste caso, haciendo referencia a la *Tabla 7*, el modelo **TBATS** presenta los **menores errores MAE y MSE**, posicionándose como la mejor opción para este cluster. Los modelos **SVM** y **ARNN** también muestran un **desempeño competitivo**, mientras que **KNN** y **Holt-Winters** presentan **errores mayores**. Estos resultados sugieren que modelos que capturan estacionalidades complejas, como TBATS, se ajustan mejor a esta serie temporal. Ahora, los valores reales frente a los de la predicción, se verían dibujados en la *Figura 22*.



Figura 22: Gráfico de días de datos reales frente a predicción del cluster 3

Con más detalle, se ven las horas y cómo se predicen.

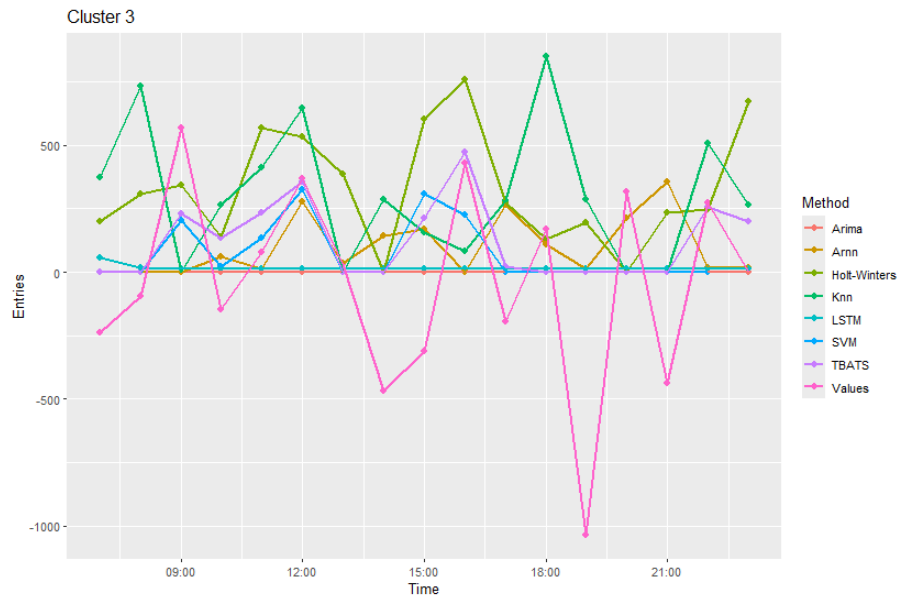


Figura 23: Gráfico de horas de datos reales frente a predicción del cluster 3

Así, a partir de la *Figura 23*, se **corroborar de forma visual lo apreciado mediante los errores anteriores**. Es importante destacar que **ninguno de estos modelos funciona bien para los vuelos que salen antes de tiempo**, es decir, los valores de retrasos negativos. En principio, esto no sucede en tantas ocasiones.

En este caso, se tomarán los modelos **TBATS**, **SVM** y **ARIMA**, y se combinarán como en los otros clusters.

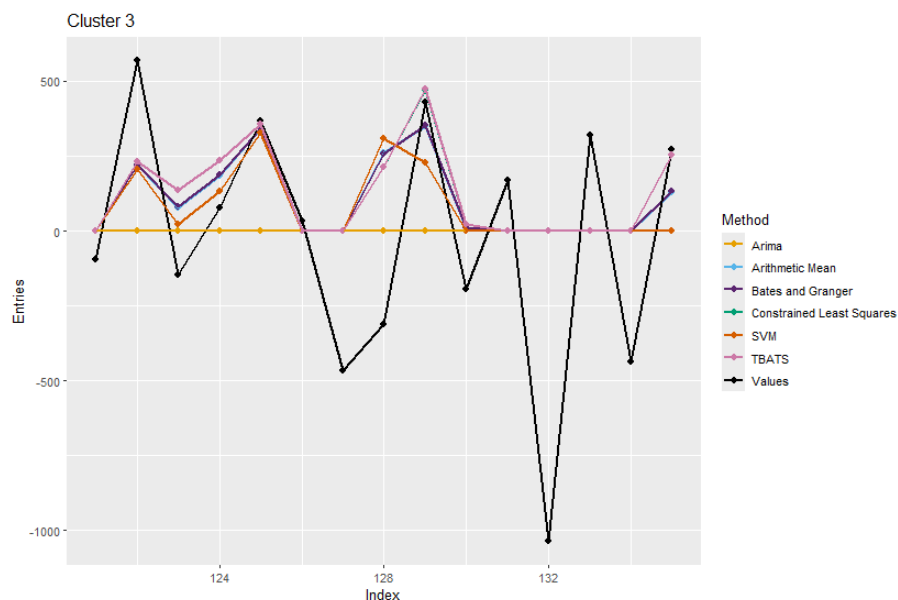


Figura 24: Gráfico de predicciones combinadas del cluster 3

Visualmente, se comprueba lo dicho y se ve que varias de las líneas se solapan en la *Figura 24*,

aunque se calculan los errores para tener una base más precisa.

Modelo	MAE	MSE
Arithmetic Mean (AM)	284.80	147251.5
Bates and Granger (BG)	284.33	147066.9
Constrained Least Squares (CLS)	<b>276.67</b>	<b>144576.8</b>

Cuadro 8: Errores MAE y MSE para la predicción combinada del cluster 3

En este caso, teniendo en cuenta la parte numérica de la *Tabla 8*, es mejor **no realizar una combinación por este método**, ya que la de TBATS era mejor viendo los errores. Así, la predicción del día con detalle en las horas quedaría representada de la siguiente forma:

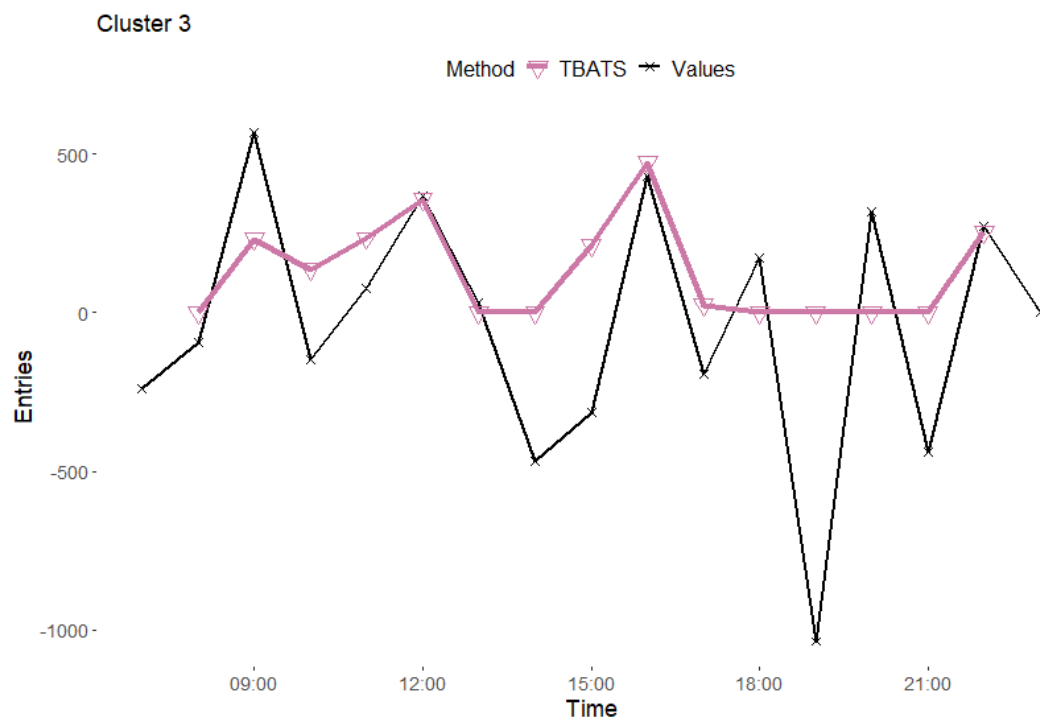


Figura 25: Gráfico de predicciones por el método TBATS del cluster 3

En la *Figura 25* se aprecia con más detalle lo analizado anteriormente. Ahora, para ver más allá, al igual que en los casos anteriores se visualiza cómo se vería la predicción para el siguiente día fuera de muestra, ya que es bastante útil en este tipo de estudios.

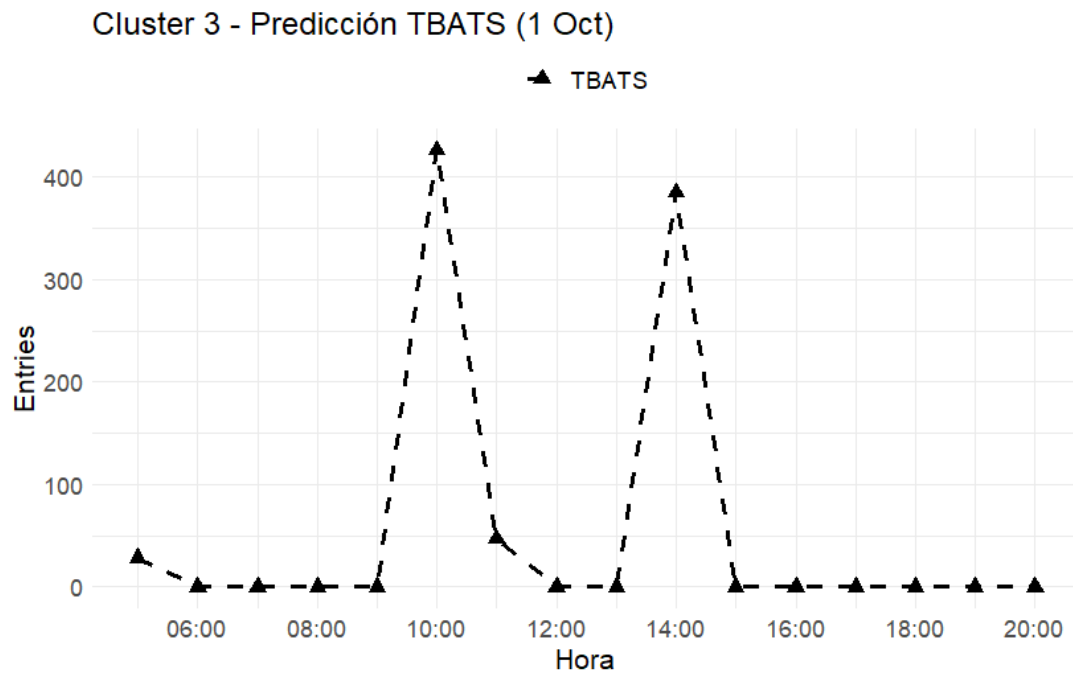


Figura 26: Gráfico de predicciones para el 1 de octubre en el cluster 3

En este último caso, la predicción de nuevos valores, como indica la *Figura 26*, solo consigue **dos picos, uno sobre las 9:00 y otro sobre las 13:00**, que tardan dos horas en estabilizarse primero.

## 5. Conclusiones

Este trabajo se ha centrado en un problema de **gran relevancia para el sector aeroportuario**: la **predicción fiable de los retrasos en los vuelos**. Esta capacidad resulta **clave para mejorar la eficiencia operativa, reducir costes, mitigar el impacto ambiental y ofrecer una mejor experiencia al pasajero**. A partir de datos horarios de tráfico aéreo y retrasos en **50 aeropuertos europeos durante septiembre de 2017**, se ha desarrollado una **metodología que combina análisis de series temporales, técnicas de *clustering* y modelos de predicción**, con el objetivo de anticipar el comportamiento del sistema aeroportuario de forma más **precisa y segmentada**.

Uno de los aspectos fundamentales ha sido la **correcta limpieza, estructuración y conversión de los datos en series temporales**. A pesar de tratarse de un conjunto de datos amplio y detallado, la **falta de observaciones consecutivas a lo largo de varios meses** limita la capacidad de ciertos modelos para capturar dinámicas de largo plazo o estacionalidades, afectando especialmente a técnicas como **ARIMA o LSTM**. Esta discontinuidad representa **una de las principales limitaciones del estudio**, aunque se ha conseguido realizar **predicciones fuera de muestra útiles**, como anticipar el comportamiento del primer día de octubre usando únicamente datos de septiembre.

Para abordar la **diversidad de comportamientos entre aeropuertos**, se aplicó un análisis de **conglomerados basado en la función de autocorrelación (ACF)**, que permitió **identificar grupos con dinámicas similares**. Esta métrica demostró ser **robusta frente a desplazamientos y escalas**, y el ***clustering* jerárquico** reveló tres grandes grupos **diferenciados**, facilitando un enfoque predictivo más adaptado a la realidad del sistema.

A partir de estos clusters, se eligieron **representantes característicos en función de la varianza**, lo que permitió **reducir la carga computacional sin comprometer la calidad del análisis**. Se aplicaron distintos modelos predictivos: **ARIMA, Holt-Winters, TBATS, SVM, KNN, redes neuronales (ARNN) y LTSM**, con especial potencial para ser **una línea prioritaria para estudios futuros**.

Los resultados indican que modelos como **SVM y TBATS ofrecen un rendimiento sólido**, mostrando capacidad para adaptarse a **dinámicas no lineales y estacionalidades irregulares**. La evaluación mediante **MAE y MSE** confirma que estos modelos **superan, en general, a alternativas más tradicionales**. También se exploró la **combinación de modelos** (media aritmética, Bates & Granger, CLS), que, aunque no siempre mejora al mejor modelo individual, **aumenta la robustez especialmente en clusters más variables**.

Desde una perspectiva práctica, esta metodología proporciona **una herramienta realista**

y escalable para anticipar cuellos de botella, optimizar recursos y mejorar la toma de decisiones operativas. La segmentación por grupos permite **ajustar las estrategias predictivas a las particularidades de cada aeropuerto**, lo que **incrementa su aplicabilidad en contextos reales**.

Las principales **líneas futuras de mejora** incluyen:

1. La **obtención de series temporales más largas y continuas**, para integrar modelos como LSTM y detectar estacionalidades reales.
2. La **exploración de diferentes configuraciones de *clustering***, dado que los métodos cuantitativos actuales no ofrecen una respuesta definitiva.
3. La **incorporación de técnicas de imputación** para tratar datos perdidos o discontinuos.
4. El **análisis del impacto de variables exógenas** como meteorología, eventos especiales o incidencias técnicas.
5. La **ampliación del horizonte de predicción** más allá de las 24 horas, junto con **implementaciones prácticas en tiempo real**.

En resumen, este trabajo establece **una base metodológica sólida** para el uso combinado de **técnicas de segmentación y predicción** en el análisis de retrasos aeroportuarios. La **validación con datos reales, la selección cuidadosa de modelos y el enfoque adaptado por clusters** demuestran que, incluso con ciertas limitaciones, es posible **ofrecer soluciones predictivas útiles para la gestión aeroportuaria**. Con la integración de datos más ricos, esta metodología tiene el potencial de **escalar a un sistema inteligente, eficiente y sostenible**, alineado con los desafíos actuales del sector.



## Bibliografía

- [ASW15] Aghabozorgi, S, Shirkhorshidi, A S y Wah, T Y. Time-series clustering – A decade review. *Information Systems* **53**, 16-38 (2015).
- [BLH+15] Bagnall, A, Lines, J, Hills, J y Bostrom, A. Time-Series Classification with COTE: The Collective of Transformation-Based Ensembles. *IEEE Transactions on Knowledge and Data Engineering* **27**, 2522-2535 (2015).
- [BG69] Bates, J M y Granger, C W J. The Combination of Forecasts. *Operational Research Quarterly* **20**. Print, 451-468 (1969).
- [BJ23] Birolini, S y Jacquillat, A. Day-ahead aircraft routing with data-driven primary delay predictions. *European Journal of Operational Research* **310**, 379-396 (2023).
- [BJR08] Box, G E P, Jenkins, G M y Reinsel, G C. *Time Series Analysis: Forecasting and Control* 4.<sup>a</sup> ed. (John Wiley & Sons, Hoboken, N.J., 2008).
- [C+04] Chang, J.-W *et al.* *An Efficient Clustering Method for High-Dimensional Data Mining* en *Advances in Artificial Intelligence – SBIA 2004* (2004), 276-285.
- [DG24] Di Fonzo, T y Girolimetto, D. Forecast Combination-Based Forecast Reconciliation: Insights and Extensions. *International Journal of Forecasting* **40**, 490-514 (2024).
- [DM09] D’Urso, P y Maharaj, E A. Autocorrelation-Based Fuzzy Clustering of Time Series. *Fuzzy Sets and Systems* **160**, 3565-3589 (2009).
- [FS16] Ferbar Tratar, L y Strmčnik, E. The Comparison of Holt–Winters Method and Multiple Regression Method: A Case Study. *Energy* **109**, 266-276 (2016).
- [Fra05] Franklin, J. The Elements of Statistical Learning: Data Mining, Inference and Prediction. *The Mathematical Intelligencer* **27**, 83-85 (2005).
- [GPP16] Gaviria-Peña, C A, Pérez-Agamez, R A y Puerta-Yepes, M E. Regresión por mínimos cuadrados parciales PLS con datos de intervalo. *Revista de la Facultad de Ciencias* **5**. Print, 148-159 (2016).

- [GR19] González, J A y Ramírez, M. ARNN: un paquete para la predicción de series de tiempo usando redes neuronales autoregresivas. *Redalyc. Red de Revistas Científicas de América Latina, el Caribe, España y Portugal* **14**, 1-12 (2019).
- [HS97] Hochreiter, S y Schmidhuber, J. Long Short-Term Memory. *Neural Computation* **9**, 1735-1780 (1997).
- [HA18] Hyndman, R J y Athanasopoulos, G. *Forecasting: Principles and Practice* 2.<sup>a</sup> ed., 1-20 (OTexts, 2018).
- [HK08] Hyndman, R J y Khandakar, Y. Automatic time series forecasting: The forecast package for R. *Journal of Statistical Software* **27**, 1-22 (2008).
- [HK06] Hyndman, R J y Koehler, A B. Another Look at Measures of Forecast Accuracy. *International Journal of Forecasting* **22**, 679-688 (2006).
- [HR11] Hyndman, R J y Razbash, S. *tbats: TBATS model (Exponential smoothing state space model with Box-Cox transformation, ARMA errors, Trend and Seasonal components)* Accedido: 20 de mayo de 2025. 2011.
- [K+24] Khadka, S *et al.* Agglomerative Hierarchical Clustering Methodology to Restore Power System Considering Reactive Power Balance and Stability Factor Analysis. *International Transactions on Electrical Energy Systems*, 1-16 (2024).
- [Kou16] Kourentzes, N. *How to choose a forecast for your time series* Accedido: 20 de mayo de 2025. 2016.
- [LK23] Lichtenberger, L y Kastner, G. *ForecastComb: Forecast Combination Methods* R package version 1.10. Último acceso: 21 de mayo de 2025 (2023).
- [MFC+19] Martínez, F, Frías, M P, Charte, F y Rivera, A J. Time Series Forecasting with KNN in R: the tsfknn Package. *The R Journal* **11**, 229-242 (2019).
- [Mod23] Modak, S. Finding Groups in Data: An Introduction to Cluster Analysis: Authored by Leonard Kaufman and Peter J. Rousseeuw, John Wiley and Sons, 2005, ISBN: 0-47-1-73578-7. *Journal of Applied Statistics* **51**, 1618-1620 (2023).
- [MV14] Montero, P y Vilar, J A. TSclust: An R package for time series clustering. *Journal of Statistical Software* **62**, 1-43 (2014).

- [Os12] Osborne, J W. *Best Practices in Data Cleaning: A Complete Guide to Everything You Need to Do Before and After Collecting Your Data* (SAGE Publications, Thousand Oaks, CA, 2012).
- [Per22] Perone, G. Comparison of ARIMA, ETS, NNAR, TBATS and hybrid models to forecast the second wave of COVID-19 hospitalizations in Italy. *The European Journal of Health Economics* **23**, 917-940 (2022).
- [PKG11] Petitjean, F, Ketterlin, A y Gançarski, P. A Global Averaging Method for Dynamic Time Warping, with Applications to Clustering. *Pattern Recognition* **44**, 678-693 (2011).
- [RDoND] RDocumentation. *frbs.learn function – Fuzzy Rule-Based Systems* <https://www.rdocumentation.org/packages/frbs/versions/3.1-0/topics/frbs.learn>. Accedido: 20 de mayo de 2025. n.d.
- [SEK21] Sammouda, R, El-Zaart, A y Khalil, A M. An Optimized Approach for Prostate Image Segmentation Using K-Means Clustering Algorithm with Elbow Method. *Computational Intelligence and Neuroscience* **2021**, 4553832 (2021).
- [STS18] Siami-Namini, S, Tavakoli, N y Siami Namin, A. *A Comparison of ARIMA and LSTM in Forecasting Time Series* en 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA) (IEEE, 2018), 1394-1401.
- [Sin25] Singh, A. *Build High Performance Time Series Models using Auto ARIMA in Python and R* Accedido: 20 de mayo de 2025. <https://www.analyticsvidhya.com/blog/2018/08/auto-arima-time-series-modeling-python-r/>.
- [SS04] Smola, A J y Schölkopf, B. A Tutorial on Support Vector Regression. *Statistics and Computing* **14**, 199-222 (2004).
- [Tiw17] Tiwari, F. *KNN implementation in R using caret package* <http://dataaspirant.com/2017/01/09/knn-implementation-r-using-caret-package/>. Accedido: 20 de mayo de 2025. 2017.
- [Val10] Valero-Mora, P M. ggplot2: Elegant Graphics for Data Analysis. *Journal of Statistical Software* **35**. Book Review (2010).

- [WM05] Willmott, C J y Matsuura, K. Advantages of the Mean Absolute Error (MAE) over the Root Mean Square Error (RMSE) in Assessing Average Model Performance. *Climate Research* **30**, 79-82 (2005).
- [ZPH98] Zhang, G, Patuwo, B E y Hu, M Y. Forecasting with Artificial Neural Networks: The State of the Art. *International Journal of Forecasting* **14**, 35-62 (1998).