Predicción del desfase de salida de vuelos con aprendizaje automático usando datos BTS TranStats (2024)

Sebastian Pelaez Acevedo, Simón Sánchez Sepulveda
Facultad de Ingeniería, Universidad de Antioquia
Medellin, Colombia
sebastian.pelaez@udea.edu.co, simon.sanchezs@udea.edu.co

I. Introducción

Las aerolíneas y aeropuertos necesitan anticipar cuántos minutos se adelantará o retrasará cada vuelo para asignar puertas, rotar aeronaves y tripulaciones, proteger conexiones y comunicar ETAs realistas. Hoy se usan reglas generales o avisos tardíos, lo que genera ineficiencias y costos. Un modelo de *Machine Learning* (ML) regresivo que prediga el desfase en minutos integra señales operativas y externas (histórico por ruta y aerolínea, rotaciones previas, congestión, franja horaria, clima, restricciones ATC) y capta relaciones no lineales que las reglas no ven. Con pronósticos más precisos y anticipados, operaciones puede actuar antes: reprogramar recursos, evitar demoras en cascada y mejorar la experiencia del pasajero.

Este artículo se organiza así: en la Sección II se describe el problema y la base de datos; en la Sección III se justifica el paradigma de aprendizaje; la Sección IV expone el estado del arte; en la Seccion V se describe el proceso de entrenamiento y evaluacion de los Modelos junto con el metodo de validacion escogido y los modelos utilizados; en la Seccion VI se explica los metodos de reduccion y extraccion de caracteristicas utilizados luego de elegir los dos mejores modelos en la seccion anterior.

II. Descripción del problema y de los datos

II-A. Fuente y construcción del conjunto de datos

La base de datos se construyó a partir de la *On-Time Performance* de *Bureau of Transportation Statistics* (BTS TranStats), un repositorio público con información de vuelos domésticos en EE. UU. actualizado mensualmente. Un autor en Kaggle consolidó todos los meses de 2024, realizó limpieza de faltantes y estandarizó nombres de características. El conjunto unificado contiene más de 7 millones de vuelos y 35 características por muestra.

Para reducir la complejidad computacional del entrenamiento se aplicó un *submuestreo* preservando la proporción de vuelos por mes, obteniendo una muestra de 10 000 registros.

II-B. Cobertura y campos

Esta base describe origen y destino, rendimiento operativo, horas programadas y reales, minutos de retraso (negativos si salió antes), cancelaciones y desvíos, entre otros. En la Tabla I se listan las variables principales con su descripción.

Cuadro I
Características del dataset y su descripción.

Nombre de la columna	Descripción
year	Año del vuelo
month	Mes del vuelo (1–12)
day_of_month	Día del mes
day_of_week	Día de la semana (1=Lunes … 7=Domingo)
fl_date	Fecha del vuelo (AAAA-MM-DD)
op_unique_carrier	Código único de la aerolínea operadora
op_carrier_fl_num	Número de vuelo de la aerolínea reportante
origin	Código del aeropuerto de origen
origin_city_name	Nombre de la ciudad de origen
origin_state_nm	Nombre del estado de origen
dest	Código del aeropuerto de destino
dest_city_name	Nombre de la ciudad de destino
dest_state_nm	Nombre del estado de destino
crs_dep_time	Hora programada de salida (local, hhmm)
dep_time	Hora real de salida (local, hhmm)
dep_delay	**Retraso en la salida en minutos (negativo si salió antes)**
taxi_out	Tiempo de rodaje de salida (min)
wheels_off	Hora en que el avión despega (local, hhmm)
wheels_on	Hora en que el avión aterriza (local, hhmm)
taxi_in	Tiempo de rodaje de llegada (min)
crs_arr_time	Hora programada de llegada (local, hhmm)
arr_time	Hora real de llegada (local, hhmm)
arr_delay	Retraso en la llegada (min; negativo si llegó antes)
cancelled	Indicador de vuelo cancelado (0=No, 1=Sí)
cancellation_code	Motivo de la cancelación (si aplica)
diverted	Indicador de vuelo desviado (0=No, 1=Sí)
crs_elapsed_time	Tiempo programado de vuelo (min)
actual_elapsed_time	Tiempo real de vuelo (min)
air_time	Tiempo en el aire (min)
distance	Distancia entre origen y destino (millas)
carrier_delay	Retraso atribuible a la aerolínea (min)
weather_delay	Retraso por condiciones meteorológicas (min)
nas_delay	Retraso por el Sistema Nacional de Aviación (NAS/ATC) (min)
security_delay	Retraso por motivos de seguridad (min)
late_aircraft_delay	Retraso por llegada tardía de la aeronave (min)

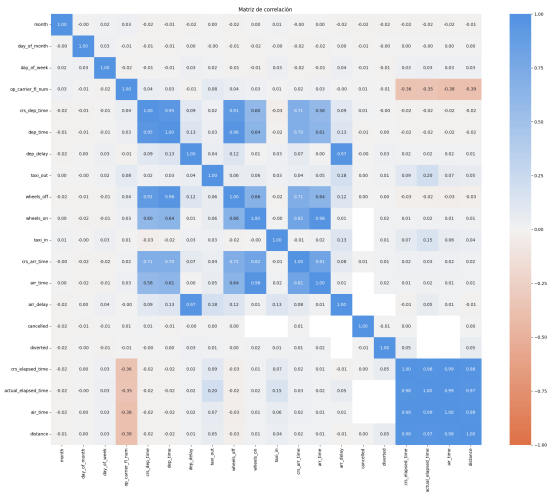

Figura 1. Matriz de correlación entre variables numéricas del conjunto BTS-2024.

II-C. Análisis de correlaciones

La matriz de correlación entre variables numéricas revela tres bloques claros. (i) **Reloj operativo**: las variables de tiempos programados y reales (`crs_dep_time`, `dep_time`, `wheels_off`, `wheels_on`, `crs_arr_time`, `arr_time`) están fuertemente correlacionadas entre sí (aprox. 0.60–0.98), pues describen la misma secuencia temporal del vuelo. (ii) **Duraciones y distancia**: `distance`, `air_time`, `actual_elapsed_time` y `crs_elapsed_time` presentan correlaciones muy altas (0.97–0.99), evidenciando redundancia: vuelos más largos implican mayores tiempos de vuelo, programados y reales. (iii) **Retrasos**: `dep_delay` y `arr_delay` están fuertemente ligados (0.97); sin embargo, `dep_delay` muestra correlaciones bajas con el resto, siendo `taxi_out` una de las mayores (0.18). Esto sugiere que el desfase de salida depende más de interacciones y relaciones no lineales (aeropuerto, aerolínea, franja, congestión) que de un único predictor numérico.

Las variables de calendario (`month`, `day_of_month`, `day_of_week`) aportan poca señal directa sobre `dep_delay`; `cancelled` y `diverted` son poco frecuentes. El `op_carrier_fl_num` presenta correlaciones negativas moderadas con `distance` y duraciones (0.36 a 0.39), por lo que conviene tratarlo como *categoría* y no como valor continuo.

Implicaciones para el modelado: para predicción antes del despegue debe evitarse la fuga de información (no usar variables observadas a posteriori como `dep_time` real, `arr_time`, `actual_elapsed_time` o `arr_delay`). Dada la baja correlación lineal de `dep_delay`, se recomiendan modelos no lineales (ensambles de árboles, SVR) e ingeniería de atributos (franjas horarias, indicadores por aeropuerto y aerolínea, variables de congestión). También es razonable reducir o agrupar el bloque de *duraciones y distancia* por su alta redundancia.

II-D. Faltantes, codificación e imputación

Las variables temporales (`dep_time`, `arr_time`, `wheels_off`, `wheels_on`) se tratan como enteros `hhmm` con extracción de atributos (hora, minuto, franja). Las categóricas (`origin`, `dest`, `op_unique_carrier`) se codifican vía *one-hot* para los modelos. Una gran cantidad de variables son eliminadas antes de empezar con el entrenamiento, esto a causa de dos grandes razones; primeramente las variaiones de delay en la base de datos, ya que el autor del dataset relleno muchos datos faltantes con ceros, asi afectando el comportamiento original de la variables; por segundo se eliminan variables que no se pueden conseguir antes del despegue, debido a que el proyecto se abordo con una contexto de negocio en mente, este contexto requeria que la informacion que debia ser utilizada en desarrollo del modelo de ML, fuera adquirible antes de el despegue, para asi poderle dar un valor real a la prediccion.La variable objetivo es `dep_delay` (min).

III. Paradigma de aprendizaje

El paradigma seleccionado es **regresión supervisada** para predecir el desfase continuo (minutos) respecto a la salida programada. Inicialmente se consideró una clasificación binaria (retraso vs. no retraso), pero al tratarse de una variable objetivo numérica con signo (positivos = retraso; negativos = adelanto), se puede observar el comportamiento de la variable objetivo en la figura 2, la regresión ofrece mayor valor operativo: permite estimar magnitudes y umbrales adaptativos, priorizar recursos y evaluar impactos de *what-if*. La evaluación se plantea con MAE y RMSE; si el coste de subestimar es mayor, puede emplearse una función de pérdida asimétrica.

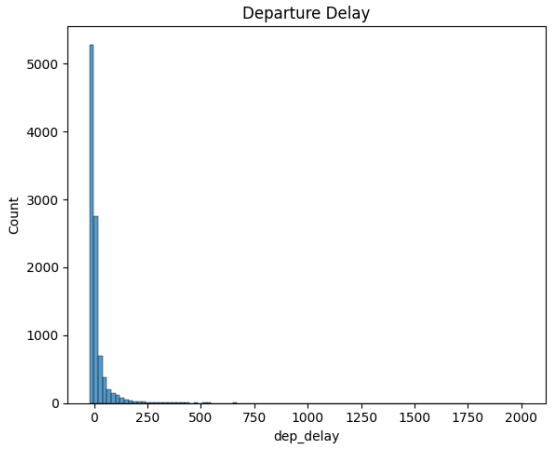

Figura 2. Histograma de distribucion dep-delay.

IV. Estado del arte

Haciendo una basta investigacion del estado del arte no se logro encontrar un informe, estudio o articulo que utilizara exactamente la misma base de datos utilizada por nosotros en el proyecto, sin embargo los trabajos elegidos para utilizar de referencia, estan hechos sobre la base de datos BTS, pero en

distintos años y con algunas variaciones de las features, tambien algunos de los trabajos utilizarion informacion meteorologica extra, cosa que los beneficio mucho en los resultados de su modelos predictivos.

La literatura sobre puntualidad y retrasos de vuelo se estructura en tres formulaciones: (i) **clasificación binaria** (retraso vs. no retraso) para decisiones rápidas; (ii) **regresión** del desfase en minutos, más útil para asignación final de recursos; y (iii) **esquemas de dos etapas** que primero detectan si habrá retraso y luego estiman su magnitud. Los modelos más eficaces reportados combinan **árboles potenciados** (Gradient Boosting, XGBoost/LightGBM), **Random Forest** y **MLP/DNN**; varias propuestas incorporan **segmentación previa** (p. ej., K-Means) para entrenar clasificadores/regresores. La validación típica separa entrenamiento/validación/prueba y, cuando procede, se añade K-Fold o particiones temporales; se trata el desbalance con *class weights* u *oversampling*. En clasificación se priorizan precisión, *recall* y F1 (macro/ponderado), además de *accuracy*; en regresión se emplean MAE y RMSE/MSE, y a veces R^2 o regresión cuantil para intervalos.

Thiagarajan et al. presentan un *pipeline* supervisado en dos etapas con datos históricos BTS y meteorología: (1) **clasificación** (retraso sí/no) y (2) **regresión** (minutos). Comparan Random Forest, Extra-Trees, AdaBoost, **Gradient Boosting** y MLP/DNN; usan partición *train/test* (3:1) y validaciones K-Fold (K=5) y *Leave-One-Year-Out*, con *grid search* para hiperparámetros. En clasificación reportan *accuracy*, precisión y *recall*; en regresión, MSE y R^2. Sus mejores resultados son: **94,35 % de *accuracy*** en llegada con Gradient Boosting; en regresión, **Extra-Trees** obtiene **MSE=68,31** (arribo) y, tras *entrenamiento selectivo* por ruta (p. ej., ATL–CLT), **MSE=26,36** [4].

Ortega (Trabajo Final Master) estudia BTS (aeropuerto Internacional Hartsfield–Jackson de Atlanta) y llegadas a Madrid-Barajas (MAD) enriquecidas con METAR (informe meteorológico rutinario de aeródromo usado en aviación). Formula principalmente **clasificación binaria** y explora un enfoque híbrido **K-Means y Random Forest** (segmentación y clasificador especializado), además de **XGBoost** y MLP. La validación separa *train/valid/test*, emplea *early stopping* y búsquedas de hiperparámetros (grid/aleatoria); trata el desbalance con *class weights*/SMOTE. Para el **aeropuerto Internacional Hartsfield–Jackson de Atlanta (BTS)**, reporta: Naive Bayes (Acc. 85, W-F1 0,85), Reg. Logística (84, 0,84), Random Forest (86, 0,84), **XGBoost (87, 0,87)**, **MLP (87, 0,87)** y **K-Means y Random Forest (90, W-F1 0,90)**; las variables más influyentes en XGBoost son *matrícula*, *día de la semana* y *número de vuelo*. Para **MAD**, la inclusión directa de meteorología reduce el tamaño útil del set y no mejora de forma consistente: **XGBoost** (Acc. 72, W-F1 0,72) y **MLP** (Acc. 78, W-F1 0,70); destacan *callsign*, *día de la semana* y *aeronave* como atributos relevantes [3]. En conjunto, confirma que los **árboles potenciados** y la **segmentación previa** ofrecen mejor equilibrio rendimiento–interpretabilidad.

Kalliguddi y Leboulluec (2017) formulan el problema como regresión para predecir los minutos de retraso de salida. Comparan Regresión Lineal Múltiple, Árboles de Decisión y Random Forest sobre datos de Bureau of Transportation Statistics de 2016, con limpieza e imputación, y división en entrenamiento y prueba (con poda del árbol). Evalúan con RMSE (raíz del error cuadrático medio) y R2 (coeficiente de determinación). Resultados: Regresión Lineal Múltiple con R2 aproximado 0.84 y RMSE 21.2 minutos; Árbol de Decisión con RMSE 26.5; Random Forest con R2 aproximado 0.94 y RMSE 12.5 (mejor desempeño) [5].

Moreno Maderuelo (2022) aborda una clasificación binaria en dos etapas con datos de Estados Unidos 2019: primero salida con retraso y luego llegada con retraso. Prueba Regresión Logística, Árboles de Clasificación, k Vecinos más Cercanos, Máquinas de Vectores de Soporte (lineal y polinómica), Random Forest y Gradient Boosting, usando particiones de entrenamiento, validación y prueba, y priorizando la sensibilidad en la segunda etapa. Emplea Exactitud (tasa de acierto), Sensibilidad (recall), Curva ROC y AUC (área bajo la curva). Resultados: en la etapa de salida, Máquinas de Vectores de Soporte polinómica logra Exactitud 59.7 y AUC aproximadamente 0.649 en el conjunto de prueba; en el resumen global del flujo, Gradient Boosting alcanza Exactitud 60.1 y Sensibilidad 60.2 [6].

V. Entrenamiento y Evaluación de los Modelos

El objetivo fue predecir el desfase continuo en minutos de `dep_delay`. Se trabajó con un subconjunto balanceado por mes del BTS 2024: tras depurar 116 registros sin variable objetivo, quedaron 9 884 muestras con 15 variables. Las variables categóricas se representaron con codificación de una sola vez, manteniendo la capacidad de generalizar ante categorías no vistas; las numéricas se escalaron para evitar efectos de escala en modelos sensibles. La separación de datos se hizo en entrenamiento (80 %) y prueba (20 %) con semilla fija para reproducibilidad.

V-A. Configuración experimental y validación

Para ajustar y comparar modelos se empleó validación cruzada con cinco particiones y barajado en el conjunto de entrenamiento. El criterio de selección fue el error absoluto medio sobre las particiones de validación. Una vez elegido el mejor conjunto de hiperparámetros para cada modelo, se reentrenó con todo el bloque de entrenamiento y se evaluó una única vez sobre el conjunto de prueba. Al tratarse de un problema de regresión, no se aplicaron técnicas de submuestreo o sobremuestreo; la robustez provino de la validación cruzada y del preprocesamiento consistente en todos los pliegues.

V-B. Modelos comparados y mallas de hiperparámetros

Se evaluaron cinco familias, como exige la rúbrica: (i) un modelo paramétrico de regresión, (ii) un modelo no paramétrico, (iii) un ensamble de árboles, (iv) una red neuronal y (v) una máquina de vectores de soporte. La Tabla II resume los hiperparámetros explorados.

Cuadro II
HIPERPARÁMETROS ANALIZADOS POR MODELO Y MALLA DE VALORES.

Modelo	Malla de hiperparámetros
Regresión Lineal (Ridge/LS)	ajuste de intercepto {sí, no}; restricción de coeficientes positivos {sí, no}
K Vecinos más Cercanos (KNN)	número de vecinos {3, 5, 11}; función de ponderación {uniforme, por distancia}; métrica Manhattan o Euclídea
Support Vector Regression (RBF)	penalización {1, 10}; margen epsilon {0.1, 1.0}; parámetro del núcleo {automático, escala}
Random Forest Regressor	número de árboles {100, 300}; profundidad máxima {sin límite, 10, 20}; división mínima de muestras {2, 5}
Red Neuronal (MLPRegressor)	capas ocultas {(50), (100)}; activación {ReLU, tanh}; regularización L2 {1e-4, 1e-3}

Cuadro III
DESEMPEÑO EN PRUEBA.

Modelo	MAE	RMSE	R2
Regresión Lineal	24.7057	49.72	-0.0262
SVR (RBF)	**18.2139**	50.74	-0.0887
KNN	24.7975	50.37	-0.0532
Random Forest	24.7516	52.43	-0.1410
MLP	34.1796	56.96	-0.3465

V-C. Métricas de desempeño

Se reportan tres medidas complementarias: *MAE* (error absoluto medio, interpreta "minutos de error promedio"), *RMSE* (raíz del error cuadrático medio, enfatiza errores grandes) y *R2* (proporción de varianza explicada). Para comparar configuraciones durante la validación se usó MAE; en la sección de resultados se informan MAE, RMSE y R2 sobre validación y prueba. Los intervalos de confianza se estiman sobre las particiones de la validación cruzada (media $\pm$ error estándar multiplicado por 1.96).

V-D. Resultados del entrenamiento y evaluación

En validación cruzada, la máquina de vectores de soporte con núcleo RBF fue la que logró el menor error absoluto medio entre los modelos analizados. En el conjunto de prueba, esta misma familia mantuvo la ventaja en MAE, mientras que la regresión lineal obtuvo el menor RMSE, lo que indica que penaliza menos los picos de error. Todos los R2 quedaron cercanos a cero y ligeramente negativos, lo que es coherente con la baja correlación lineal entre las variables disponibles y el desfase, y sugiere que para superar el nivel base se requieren variables operativas adicionales (por ejemplo, rotaciones de aeronave, congestión de puerta o clima enriquecido).

En cuanto al efecto de los hiperparámetros: en Random Forest, aumentar el número de árboles y permitir mayor profundidad mejoró el error en validación hasta estabilizarse; en KNN, valores de vecinos más altos redujeron el ruido pero perdieron detalle local; en SVR, la combinación de penalización moderada y margen más amplio ofreció el mejor equilibrio sesgo–varianza; en MLP, arquitecturas compactas con regularización intermedia funcionaron mejor que redes más grandes en este volumen de datos.

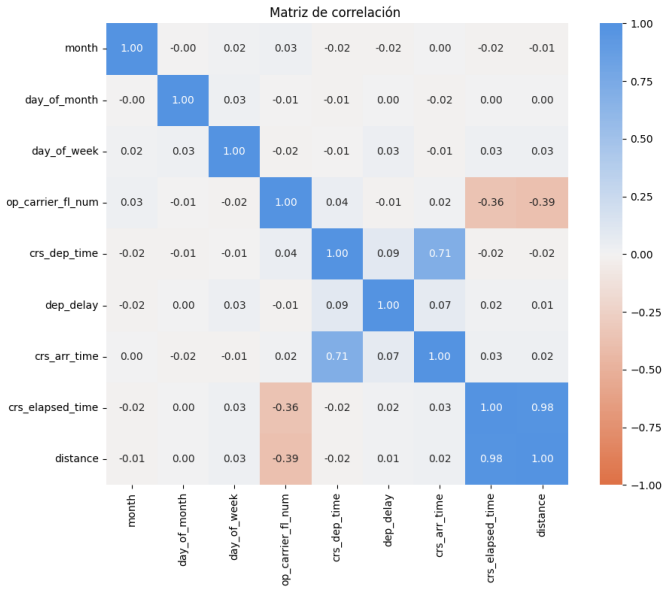

Figura 3. Matriz de correlación con las features elegidas.

VI. REDUCCIÓN DE DIMENSIÓN

VI-A. Análisis individual de variables

Las correlaciones lineales con `dep_delay` son bajas; la hora programada de llegada muestra relación leve y la distancia se asocia sobre todo con los tiempos de vuelo. Esto explica el buen desempeño relativo de métodos no lineales y la dificultad de modelos puramente lineales para capturar la variabilidad total del retraso sin variables adicionales.

VI-B. Extracción lineal: PCA

Se aplicó reducción lineal preservando el 95 por ciento de la varianza acumulada. Tras la codificación, el espacio original contenía del orden de cientos de variables; la proyección redujo la dimensionalidad a 113 componentes (reducción aproximada del 83.7 por ciento) sin pérdida material de precisión. Reentrenado sobre esta representación, el mejor modelo mantuvo errores muy próximos a los del espacio completo, lo que sugiere redundancia informativa en las variables originales.

VI-C. Extracción no lineal: UMAP

Se evaluó una proyección no lineal con un número de componentes igual al usado en PCA para facilitar la comparación. El desempeño resultó prácticamente equivalente al de la proyección lineal: ligeras diferencias en MAE y RMSE, con valores cercanos a los de la representación completa. Esto indica que, con el conjunto actual de variables, las estructuras relevantes para la predicción ya quedan capturadas por una reducción lineal moderada.

La gran ventaja de UMAP es que tambien es un gran metodo para visualizar formas de la distribucion de los datos en una dimension mucho mas baja, conservando las vecindades locales de dimensiones muchisimo mas altas, entonces se decidio hacer

Cuadro IV
COMPARACIÓN DE LOS DOS MEJORES MODELOS TRAS REDUCCIÓN DE DIMENSIÓN (MINUTOS).

Método	Componentes	MAE	RMSE	R2
PCA (95 por ciento var.)	113	18.2056	50.7333	-0.0682
PCA (variante)	113	18.2097	50.7613	-0.0696
UMAP	113	18.4662	50.9581	-0.0777

una reduccion a dos dimensiones para ver la data en un plano cartesiano.

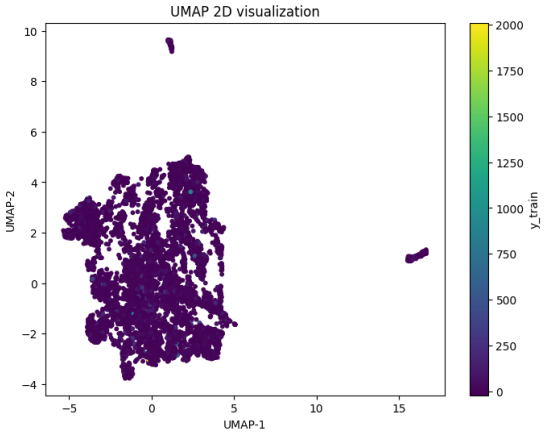

Figura 4. Visualizacion de los datos en 2D.

VI-D. Discusión y conclusiones

Los experimentos muestran que, con el conjunto de variables disponible, los métodos más competitivos en términos de minutos de error son las máquinas de vectores de soporte y, en menor medida, la regresión lineal por su RMSE. La reducción de dimensión permite compactar el espacio de entrada en más del ochenta por ciento manteniendo el rendimiento, lo que simplifica el modelo sin coste relevante en precisión. Para incrementar la capacidad explicativa (R2) y reducir los picos de error, será clave incorporar variables operativas y de entorno con mayor poder predictivo (rotación de aeronaves, congestión, condiciones meteorológicas codificadas con más detalle y rezagos), así como explorar pérdidas asimétricas o regresión cuantil cuando el coste de subestimar un retraso sea mayor que el de sobreestimarlo.

REFERENCIAS

[1] Bureau of Transportation Statistics (BTS), "On-Time Performance (TranStats)." [En línea]. Disponible en: https://www.transtats.bts.gov/ontime/. Accedido: oct. 2025.

[2] Autor del dataset en Kaggle, "US Flights On-Time Performance 2024 (compilación BTS)." 2024. [En línea].

[3] Universitat Politècnica de València, "Predicción de retrasos de vuelos: estudio comparativo y modelos híbridos." 2020. Disponible en: https://riunet.upv.es/server/api/core/bitstreams/9fbb9fff-b9d8-45bd-bdb0-c4225b988984/content. Accedido: oct. 2025.

[4] B. Thiagarajan, L. Srinivasan, A. V. Sharma, D. Sreekanthan y V. Vijayaraghavan, "A Machine Learning Approach for Prediction of On-Time Performance of Flights," en *Proc. IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, 2017, pp. 1–10. doi: 10.1109/DASC.2017.8102138.

[5] A. M. Kalliguddi y A. K. Leboulluec, "Predictive Modeling of Aircraft Flight Delay," *Universal Journal of Management*, vol. 5, n.º 10, pp. 485–491, 2017. doi: 10.13189/ujm.2017.051003..

[6] A. M. Moreno Maderuelo, "Análisis y clasificación del retraso de los vuelos comerciales de Estados Unidos en 2019," Trabajo de Fin de Máster, Facultad de Estudios Estadísticos, 2022. .