Flights Delay Prediction

Integrantes:
Davier Sánchez Bello C-412
Brian Ameht Inclan Quesada C-412
Marcos Antonio Ochil Trujillo C-412
Maykol Luis Martínez Rodríguez C-412

July 8, 2024

1 Enlaces a nuestro proyecto en Kaggle

Enlaces:

- Flight Delays Prediction Part I (cleaning)

https://www.kaggle.com/code/daviersnchez/flight-delays-prediction-part-i-cleaning

- Flight Delays Prediction Part II (feature engineering)

https://www.kaggle.com/code/daviersnchez/flight-delay-prediction-p-ii-feature-engineering

- Flight Delays Prediction Part II (using models)

https://www.kaggle.com/code/daviersnchez/flight-delays-prediction-pt-iii-using-models

- Airline Encoding Analysis

https://www.kaggle.com/code/daviersnchez/airlines-encoding-analysis

- Airport Saturation Data

https://www.kaggle.com/code/daviersnchez/airport-saturation-data

- Airport Encoding Analysis

https://www.kaggle.com/code/daviersnchez/airport-encoding-analysis

- US Airport Timezones

https://www.kaggle.com/datasets/daviersnchez/us-airports-timezones

2 Introducción

La aviación ha experimentado un crecimiento exponencial en las últimas décadas, convirtiéndose en uno de los medios más eficientes y rápidos de transporte global. Sin embargo, este avance no está exento de desafíos, siendo uno de los más significativos el problema de los retrasos de vuelo. Los retrasos pueden tener consecuencias directas e indirectas, afectando tanto a los pasajeros como a las aerolíneas y, por ende, al sistema económico global.

2.1 Motivaciones

A pesar de los numerosos estudios y modelos desarrollados para predecir el retraso de los vuelos, la mayoría de estos se basan en datos complejos y detallados, los cuales son difíciles de obtener en la práctica. La literatura existente sugiere que la predicción precisa de los retrasos de vuelo requiere modelos altamente sofisticados y datos exhaustivos, lo cual rara vez se logra en situaciones reales. Esta brecha entre la teoría y la aplicación plantea una oportunidad para mejorar la precisión y accesibilidad de las predicciones de retraso de vuelo.

Además, hemos decidido continuar el trabajo presentado en "An Intelligent Approach for Flight Delays Prediction" de Mahmout y Ezzas, adoptando su enfoque basado en Time Flight Delay (TFD) y Previous Flight Delay (PFD), TFD es la demora entre las operaciones y PFD es la demora del vuelo anterior. Este enfoque nos permite explorar la posibilidad de utilizar la menor cantidad de datos posible, enfocándonos en características medias que representen el comportamiento típico de aeropuertos y aerolíneas.

2.2 Problematica

Las principales problemáticas identificadas en el ámbito de la predicción de retrasos de vuelo incluyen:

- Complejidad de los Modelos: Muchos modelos existentes requieren grandes cantidades de datos detallados, lo cual puede ser impracticable en entornos operacionales.

- Acceso a Datos: La disponibilidad de datos de alta calidad y relevancia para entrenar modelos de machine learning es limitada.

- Saturación de Aeropuertos: El fenómeno de la saturación de aeropuertos, caracterizado por el aumento de tráfico y la falta de capacidad, no se

considera adecuadamente en la mayoría de los modelos actuales.

2.3 Objetivos generales y específicos o hipótesis o preguntas científicas

La mayor parte de los papers que aparecieron mientras realizábamos el survey, así como una parte ostensible de los que nos leímos, consisten en enfoques al problema de predecir el retraso de los vuelos usando modelos extremadamente complejos o data que no es posible o es muy poco probable con la que podamos contar en tiempo real. En cambio escogimos seguir la pauta marcada por el paper A Novel Inteligent Approach for Flight Delays Predictions, que a su vez se basa en el Flight Delay Prediction Using Gradient Boosting Machine Learning Classifiers de Mingdao Lu, de abstraer data temporal secuencial dentro de features de cada vuelo, además de seguir a los aviones individualmente.

Nuestro objetivo fue tomar este enfoque y radicalizarlo, realizando un pre procesamiento de la data basado en el Conocimiento del negocio que nos permitiera embeber en cada datapoint las features necesarias para que se pueda volver agnóstico con respecto al tiempo en que ocurrió, de manera que la data puede ser analizada por modelos de mucha menor complejidad que los usados normalmente para datos que constituyen secuencias temporales.

Queríamos aprovechar lo vago en los papers mencionados acerca del encoding de las variables categóricas, así como nuestras discrepancias con la que realizan para introducir nuevas modalidades de encoding, como la saturación de los aeropuertos en los momentos de la operación, o datos particulares del clima relacionados con el mes y el aeropuerto en cuestión, ademas de particularizar los delays en cuanto a su responsable, aerolínea aeropuerto o clima.

Quisimos también hacer un pequeño estudio de cuales modelos clásicos de ML se comportarían mejor sobre la data generada por el tipo de preprocesamiento que realizamos.

En definitiva, nos propusimos ahondar en la intención de crear un modelo que requiera solo data que sea posible obtener en el momento de realizar la predicción, a través de un encoding especial de las variables categóricas, que es el punto que consideramos débil de los papers predecesores en este campo y que a su vez no fuera necesario un modelo especialmente complejo.

3 Estado del arte/Preliminares

El problema de predecir retrasos de vuelos es un problema de Machine Learning ampliamente abordado. Predomina abrumadoramente el tratamiento del problema como un problema de series temporales, y su enfrentamiento usando modelos cada vez mas complejos. A continuación un breve resumen de algunos papers leídos

Uno de los primeros papers que leímos y que entra en la categoría de los papers que emplean modelos very fancys es Social ski driver conditional autoregressive-based deep learning classifier for flight delay prediction. En este paper los autores presentan el uso de una técnica denominada Social Ski Driver Conditional Autoregressive-based deep learning classifier for flight delay prediction.

Para entrenar el modelo usan solo los datos de los vuelos retrasados y de estos usan solo unos cuantos features de interés. Aplican el método de Yeo-Johnson para estabilizar la data, dado que produce data más organizada y sencilla de usar. Luego realizan una Feature Fusion, agrupando los features que más correlación presentan entre sí, pero usando una fórmula basada en un valor beta que esoptimizado a través del uso de una DeepRNN cuya input son los features fusionados hasta ahora (no me pregunten mucho al respecto)

Una vez realizada la feature selection se emplea una DeepLSTM como modelo para realizar la predicción, basándose en la data modelada tras la feature fusion. En ambos casos de entrenamientos de redes, tanto el de la DeepRNN como el de la DeepLSTM, se usa la técnica SSDCA para la optimización de los resultados, que no es otra cosa sino la mezcla de los algoritmos de optimización SSD y CaViar

Obtienen unos resultados brutales que oscilan entre 92 y 93 porciento de precisión.

Otro de los papers que nos estudiamos pero que en nada se acercaban a nuestro próposito fue Flight Delay Prediction Model Based on Lightweight Network ECA-MobileNetV3. Este paper estudia el algoritmo de red neuronal liviana MobileNetV3 y el algoritmo mejorado ECA-MobileNetV3. Estos son arquitecturas de red neuronal convolucional diseñada específicamente para aplicaciones móviles. Se propone un modelo de predicción de retrasos en vuelos basado en el algoritmo de red liviana ECA-MobileNetV3. El algoritmo primero preprocesa los datos con información real de vuelos e información meteorológica. Luego, para aumentar la precisión del modelo sin aumentar demasiado la complejidad computacional, la extracción de carac-

terísticas se realiza utilizando el algoritmo liviano ECA-MobileNetV3 con la adición del mecanismo Efficient Channel Attention. Finalmente, el nivel de predicción de la clasificación del retraso del vuelo se genera a través de un clasificador Softmax. En los experimentos de conjuntos de datos de aeropuertos individuales y grupos de aeropuertos, la precisión óptima del algoritmo ECA-MobileNetV3 es 98,97 porciento y 96,81 porciento.

El paper Predicting Flight Delay with Spatio-Temporal Trajectory Convolutional Network and Airport Situational Awareness Map de los autores Shao y Prabowo, lo descartamos rápidamente, porque apunta al uso de data espacio temporal compleja obtenida en tiempo real sobre los vuelos. Reconocemos que en el futuro, quizás sea mucho más sencillo tener toda esta data a mano, tanto en tiempo real como de manera histórica, terminando por desbancar nuestros modelos minimalistas para siempre.

En el paper Flight Delay Regression Prediction Model Based on Att-Conv-LSTM, su autor chino Weikan Xie rechaza centrar su atención en series temporales, para centrarse en información espacio temporal. Para enfrentar esto, lo que se propone es un método de predicción de retrasos de vuelos basado en Att-Conv-LSTM. Primero, para extraer completamente la información temporal y espacial contenida en el conjunto de datos, se utiliza la red de memoria a corto plazo para obtener características temporales y se adopta una red neuronal convolucional para obtener características espaciales. Luego, se agrega el módulo del mecanismo de atención para mejorar la eficiencia de iteración de la red. El autor afirma obtener una precisión de dos dígitos por encima de los modelos LSTM y CNN convencionales.

El paper Flight Delay Prediction Using Gradient Boosting Machine Learning Classifiers de Mingdao Lu, en retrospectiva realmente nos agradó. Cronológicamente es el primero en presentar la base del concepto de PFD y TFD (al menos entre aquellos papers que nos hemos leído), aunque nos lo leímos luego de conocer estos conceptos. Introducen ademas, features que tienen como objetivo expresar el estado de los aeropuertos de salida y llegada involucrados en cada vuelo, al igual que hacemos nosotros, aunque lo hacen de una manera mucho más sencilla, teniendo en cuenta solo la ruta en cuestión, y el estado del nivel de retrasos en los aeropuertos en dos horas alrededor de las operaciones. Esto para nosotros tiene dos problemas. Primero que todo no creemos que la ruta en particular (o sea el aeropuerto de salida y aeropuerto de Destino) sea lo suficientemente relevante y ademas, consideramos que los datos que usan respecto a los delays en el aeropuerto en las dos horas alrededor de las operaciones en cuestión, no solo es mas difícil de calcular que

nuestra variable saturación, sino que además creemos que es más propenso a propagarse destructivamente en caso de querer predecir largas cadenas de vuelos futuros.

El paper Predicting Flight Delay Using KNN (2023, Saravanakumar) es bastante cercano a lo que finalmente terminamos por hacer, basándose en data histórica de vuelos para realizar predicciones. Alegan haber obtenido una precisión del 80 porciento. No brindan ningún tipo de información sobre sus métricas o sobre los features que usaron, y usan como encoder para las columnas categóricas el provisto por sklearn directamente. No obstante su objetivo era predecir que modelos sencillos también pueden alcanzar buenos resultados prediciendo retrasos de vuelos basados en data historia, lo cual fue inspirador para nuestro propósito, al igual que lo fue el hecho de que los modelos seleccionados por ellos como los de mejor resultados fueran también los que nos dieron mejor resultado.

El paper Predicting flight delay based on multiple linear regression de Yi Ding, es otro de los que prefiere atenerse a un modelo sencillo, probando Multiple Linear Regression y Naive Bayes. Logra un 80 porciento de accuracy en predecir si un vuelo se retrasara o no usando Multiple Linear Regression. Si bien los features que usa se parecen un tanto a aquellos que tomamos, se diluye luego cuando añade entre sus features datos relacionados con el viento, que obviamente no estarán disponibles para predicciones a largo termino, así como realmente no creemos que funcionen para vuelos que cruzan largas distancias.

Algunos papers leídos no nos parece que hayan estado a la altura del resto, presentando modelos de escasa eficacia, tomando decisiones que nos parecen erradas o centrándose mas bien en cuestiones de ingeniería, obviamente centrados en la obtención de títulos de grado como es el caso de: "Modelo para identificar los vuelos afectados por retrasos o cancelaciones en el aeropuerto El Dorado de Bogotá, Colombia".

El paper que más influencia ejerció sobre nosotros y en el cual nos inspiraríamos para el presente trabajo fue el paper A Novel intelligent Approach for Flight Delays Prediction, publicado por los investigadores egipcios Mahmouts y Ezzat. La premisa de este paper es plantear un enfoque de procesamiento de la data que lleve a modelos sencillos a partir de data fácilmente disponible. Para predecir el retraso de un vuelo se centra en el itinerario del avión en cuestión en particular y mas específicamente, en el retraso que tuvo la última operación (salida o llegada) del avión.

En esencia:

Para cada vuelo crean dos datapoints, uno referido a su partida y otro referido a su llegada.

Ordenan los datapoints por id del avión, y luego por fecha y hora. De esta manera todos los datapoints protagonizados por un mismo avión quedan agrupados.

Independientemente de que los datapoints se refieran a dos momentos diferentes, dígase la salida o la llegada, todos tienen el mismo conjunto de features, de los cuales los mencionados en el paper y de relevancia son:

(Hora planificada, Orientación(salida o llegada), origen, destino, FTD, PFD, demora)

FTD y PFD son los nuevos features que consisten en:

-FTD (flight time duration): el tiempo planificado que durara el avión en una determinada labor. Para los datapoints de llegada corresponde al tiempo planificado que dure el vuelo (Tiempo estimado de llegada - Tiempo estimado de salida). Para los datapoints de salida corresponde al tiempo estimado que se mantendrá el avión en el suelo haciendo labores de mantenimiento o lo que sea, es decir (Tiempo estimado de salida - Tiempo estimado para la llegada anterior)

-PFD (previous flight delay): corresponde a la demora que tuvo la última operación realizada por el avión (en el caso de la primera operación de cada avión el PFD es 0)

O sea, de esta manera para cada avión tenemos dos datapoints por cada vuelo, uno correspondiente a los datos de su salida y uno correspondiente a los datos de su llegada. A su vez estos datapoints se encuentran separados para cada avión, de manera que seguimos la trayectoria del avión en particular, y vamos a través del feature PFD acumulando los delays previos.

La predicción:

Sea cual sea el modelo de regresión que se use, el objetivo es siempre predecir la demora de la siguiente partida o llegada de un avión en particular.

El modelo es entrenado con los datos históricos, tratando de predecir las demoras y luego comparando el resultado contra el PFD ya calculado.

Por supuesto, para los vuelos futuros que queremos predecir el PFD no se ha calculado, sino solo para el primero de ellos.

Luego, para realizar un volumen grande de predicciones en cuanto a la demora de los vuelos, basta ir pasando uno a uno los datapoints de los futuros vuelos al modelo, y el modelo ira prediciendo la demora de cada datapoint (partida o llegada), que luego será usada como PFD del siguiente datapoint en el itinerario planificado para el avión. O sea, para predecir el retraso de

grandes volúmenes de vuelos se limita a predecir el retraso de la siguiente operación de cada avión.

En fin, que el modelo tomara el último datapoint relativo a un avión, consistente en Hora planificada, Orientación, origen, destino, ftd y pfd y sera capaz de predecir el delay. Luego ese delay sera empleado como PFD en el calculo del delay del próximo datapoint (n-veces) hasta predecir todos los delays en vuelos requeridos.

Este paper de Mahmout y Ezzat deja de explotar data que podría ser accesible respecto a los vuelos futuros, como por ejemplo el nivel de saturación que tendrán los aeropuertos que intervengan en el vuelo. Además deja abierta una brecha enorme acerca de como realizar el encoding a sus variables categóricas. Además, Mahomut y Ezzat a pesar de, no plantearse como objetivo seleccionar un modelo en lo absoluto, y predicar desde el inicio del paper el uso de modelos sencillos, prueban su forma de preprocesar la data con modelos bien complejos de Ensemble Learning, RNN + LSTM, además de Gradient Boosting y Random Forests. Por último señalar, que en todos los papers que leímos se trata de predecir los retrasos de vuelos como series temporales, este fue el más cercano sin duda a deshacerse de la temporalidad, al agregarle a cada datapoint columnas que abstraen información acerca del instante de tiempo en que ocurrieron, pero luego deliberadamente o no, abandonan el camino de deshacerse de la temporalidad al incluir dentro de los features a pasar a su modelo la hora planificada de las operaciones, así como usar modelos especializados en series temporales como los RNN+LSTM.

4 Propuestas de solucion

Nuestra propuesta de solución, sigue la tónica del paper 'A novel intelligent approach for flight delay prediction'. En ese paper se proponen predecir tardanzas con un uso de datos minimal, emulando la realidad de los planificadores de vuelos, que conocen no mucho más allá respecto a los vuelos futuros que sus fechas de partida y de llegada. Su enfoque, al igual que el nuestro, va hacia un preprocesado especial de la data que permite enfatizar ciertas verdades importantes del Conocimiento del Negocio y parte, al igual que nosotros de la asunción de que los retrasos de vuelos pueden predecirse mejor si seguimos la ruta de cada avión de manera individual. Pero los autores del paper dejan sin cubrir campos importantes, respecto a la manera en que la data categórica debe ser codificada. Además, creemos que los autores no explotaron todas las posibles fuentes de datos que podrían estar disponibles para planear los vuelos con la antelación suficiente con mayor precisión.

En resumen, nuestra solución es la siguiente:

Procesado de data:

Partimos de los datos históricos de vuelos de aviones. Solo necesitamos los siguientes features:

'TAIL NUMBER' : número de cola del avión

'ORIGIN AIRPORT' : aeropuerto de origen

'DESTINATION AIRPORT' : aeropuerto de llegada

'AIRLINE' : aerolínea

'SCHEDULED DEPARTURE' : hora de salida planificada

'DEPARTURE TIME' : hora de salida acontecida

'DEPARTURE DELAY' : retraso en la salida

'SCHEDULED ARRIVAL' : hora de llegada planificada

'ARRIVAL TIME' : hora de llegada acontecida

'ARRIVAL DELAY' : retraso en la llegada

'AIRLINE DELAY' : cuál parte del retraso fue provocada por la aerolínea

'WEATHER DELAY' : cuál parte del retraso fue provocada por el tiempo

'AIR SYSTEM DELAY' : cuál parte del retraso fue provocada por el sistema aéreo

'SECURITY DELAY' : cuál parte del retraso fue provocada por motivos de seguridad

'LATE AIRCRAFT DELAY' : cuál parte del retraso fue provocada por retraso del avión en llegar.

La parte de la data que describe a que se debió cada porción del retraso, no se usa directamente para caracterizar cada datapoint y entrenar al modelo, sino para caracterizar aerolíneas, vuelos y fechas temporales, para realizar un encoding de ellas.

La limpieza de la data la abordaremos en las particularidades de nuestra implementación, ya que al tratar de replicar lo que hicimos pueden presentarse retos distintos.

Una vez que se tienen todos los vuelos limpios con todos los campos relevantes, dividimos cada vuelo en dos datapoints, la departure y el arrival. Procedemos luego a ordenarlos por número de cola del avión y como segundo criterio de ordenamiento por la fecha planificada de ocurrencia. De esta manera, podremos para cada avión modificar una a una las filas incluyendo data relativa a sus vuelos anteriores que creamos necesaria.

Ahora calcularemos, al igual que en el paper de Mahmout y Ezzat, el retraso de la operación anterior realizada por el avión y el tiempo previsto a transcurrir entre la operación anterior y la actual, llamados PFD y TFD, parámetros que ya fueron explicados anteriormente. Llega el momento de codificar la data categórica.

Ahora mismo los datos categóricos a codificar son los siguientes:
'TAIL NUMBER'
'ORIGIN AIRPORT'
'DESTINATION AIRPORT'
'AIRLINE'

Queremos además codificarlo de manera que en cada datapoint quede incluida la mayor cantidad de información relevante a la fecha de manera que podemos luego eliminarla, y cada datapoint sea independiente temporalmente, o sea, nuestro modelo no necesite tener los datapoints en orden para realizar una predicción (manteniendo el modelo simple).

Los encodings usados en los papers que pudimos leer fueron One-Hot-Encoding y Label Encoding. No nos gustó ninguno de los dos para nuestro modelo.

One Hot Encoding eleva demasiado la dimensionalidad, y la cantidad de columnas que tenemos con información relevante para cada datapoints no nos parecen la suficiente como para defenderse semejante gula dimensional (Hay 23 aerolíneas en el dataset que usamos pero en la vida real pudiera ser mucho más, por no hablar de la cantidad de aeropuertos). Por otro lado eso conllevaría complejizar en demasía el modelo.

Label Encoding no nos gusta, pues introduce un sesgo de ordinalidad ajeno al conocimiento que tenemos del negocio.

En cambio escogimos usar Target Encoding, técnica que sin elevar la dimensionalidad de la matriz pretende reflejar la influencia de cada categoría sobre la data. Target Encoding es la sustitución de cada categoría por el promedio que alcanza la variable prevista en los datapoints que toman su valor.

Nosotros, sin embargos no decidimos usar el promedio. Nos parece que el promedio de los delays no es una variable lo suficientemente expresiva, ya que da tanto peso a las categorías que producen pocos retrasos pero de larga duración como a las categorías que retrasan mucho pero cortos períodos. Además, dado que la mayoría de los vuelos no se retrasan o retrasan muy poco o cantidad negativa se distorisionaría la data y además esta estadística no nos permite trazar una línea clara entre retrasos reales (aquellos mayores a 15 minutos) y retrasos por ejemplo de 14 o 13 minutos que no cuentan realmente como retrasos acorde al sistema estadounidense de aviación.

En cambio, nuestro target encoding atendiendo a las limitaciones del target encoding promediado, sustituye cada columna categórica a la que se lo aplicaremos por dos columnas: la proporción de los vuelos con esa categoría que retrasan, y el promedio de los retrasos, cuando sea que estos ocurren, de manera que se incluyen tanto los datos de la cantidad de retrasos y promedio de estos cuando son, sin distorsionarse ni afectarse mutuamente. De hecho, comprobamos la covarianza entre ambas variables en las ocasiones en que la usamos, siempre encontrando un valor negativo medio (-0.6, -0.5), de manera que no introducimos colinearidad en la data al usarlas. Vayamos caso a caso con el encoding

Aerolíneas:

Sustituimos el nombre de cada aerolínea por la proporción de los vuelos de esa aerolínea que se retrasan por causas inducidas por la propia aerolínea, así como por el promedio de duración de tales retrasos.

Aeropuertos:

Para codificar los aeropuertos, empleamos target encoding, pero no usamos solo dos columnas, sino 5, debido a que los aeropuertos agrupan en sí varios factores que influyen directamente en el retraso de los aviones. Esta codificación nos permite además, deshacernos de las columnas temporales for good. Sin más adentrémonos en ellas.

La calidad operacional de los aeropuertos varía, por diversos motivos, como el diseño del aeropuerto o el interés socioeconómico en él, que permita

o no modernizarlo. La calidad operacional de un aeropuerto es independiente de la etapa del año, y viene expresada por los vuelos que son explícitamente categorizados como retrasados por el Sistema de Control del aeropuerto. Dada la atemporalidad de esta data, podemos describirla de la misma manera que describimos los retrasos inducidos por las aerolíneas, como la proporción de los vuelos retrasados por el sistema de control del aeropuerto y cuanto duraron en promedio esta clase de retrasos.

La localización del aeropuerto, influye a través del factor climático. El clima varía de mes a mes de formas diferentes en aeropuertos diferentes, razón por la cual esta data no es atemporal. Luego, para cada mes y cada aeropuerto computamos la proporción de retrasos taggeados como relacionados con el clima y el promedio de la duración de estos.

Un último parámetro relacionado con los aeropuertos y que no queremos dejar fuera, y cuya inclusión consideramos novedosa de nuestro enfoque es la saturación que presenta el aeropuerto en el momento de realizar la operación en cuestión. Definámosla.

Separando los eventos del dataset en intervalos de 5 minutos podemos saber para cada aeropuerto cuantos aviones hay realizando operaciones de despegue o aterrizaje en este preciso momento. La saturación de un momento se define como cuán a tope está de aviones el aeropuerto y es el cociente de la cantidad de aviones que realizan operaciones en el aeropuerto en el momento en cuestión dividida por el máximo valor que ha tenido tal estadística. Pero no quisimos dejarlo así, y calculamos la saturación para una operación X como una ponderación de cuan saturado estaba el aeropuerto en el intervalo de 5 minutos el momento de la operación en cuestión y la saturación en algunos otros intervalos alrededor del intervalo, sobre todo los anteriores y el inmediatamente posterior (este último en el caso de que el vuelo nos quedara casi en el límite de su intervalo).

Se pudiera pensar que este dato no está fácilmente disponible para entrenar el modelo, o que mucho menos para predecir un vuelo que está lejano en el futuro. Pero por el contrario. La saturación para cada aeropuerto en cada intervalo de 5 minutos es trivial de calcular para vuelos ya ocurridos si se dispone de la data histórica. Por otro lado, para vuelos en el futuro, se pueden adoptar dos enfoques para el cálculo de la saturación. El primero de ellos y más natural parte del hecho de que la predicción de los vuelos futuros se realizara uno a uno, en orden de su fecha de salida planificada. A medida que se vayan prediciendo sus retrasos se puede mantener un estimado de cuanto esperamos que se retrase y por tanto de cuando esperamos que llegue

a su aeropuerto de destino, por lo que bastaría mantener los datos para cada aeropuerto de cuántos aviones tenemos previsto que lleguen o salgan en cada intervalo de tiempo.

Si por otro lado, se quisiera predecir los futuros retrasos para un solo avión, sin tener en cuenta ningún otro vuelo, no sería posible tener calculado el dato de la saturación del aeropuerto, pero dado que la cantidad de aviones entrando y saliendo del aeropuerto en cada intervalo de tiempo es una serie de tiempo clásica, para aeropuertos muy grandes pudiera ser predicha muy eficientemente, por lo cual este dato también estaría accesible si alimentáramos el modelo de predicción de series de tiempo con la data del aeropuerto en cuestión. De hecho, lo intentamos con prophet, pero no obtuvimos resultados satisfactorios, dado que los intervalos de tiempo de 5 minutos parecen ser un intervalo muy corto.

Este parámetro saturación, realmente nos ahorra tener que dar al modelo datos como el mes, el día y la hora, ya que, en lugar el modelo tener que buscar la relación relacionada entre las fechas y los retrasos directamente le estamos dando la saturación, causa directa de los retrasos y directamente relacionada con la hora, así como anteriormente le habíamos dado la data climática que abstrae la relación de los meses y el clima.

Este parámetro de la saturación es especialmente importante para los retrasos de vuelos, y no leímos ningún paper que hiciera uso de él o mencionara su uso en alguna referencia al historia del arte.

Falta por mencionar respecto a la codificación de los aeropuertos, que hay aeropuertos muy grandes y aeropuertos muy pequeños. Para los aeropuertos pequeños calcular los valores de proporción de retrasos y promedio de retrasos es un riesgo, porque se tiene poca data de ellos y el resultado muy probablemente esté distorsionada. Por tanto decidimos establecer un threshold tal que, para los aeropuertos que tuvieran un número de vuelos mayor que el threshold fueran calculadas estas estadísticas de manera individual, mientras que para todos los que tuvieran un número de vuelos menor que éste, calcular su valor como conjunto y asignarle el resultado a todo. Para seleccionar el threshold procedimos a la inspección visual de gráficos de dispersión realizados sobre la data y el número de vuelos disponibles para cada aeropuerto, y mediante la inspección visual pudimos distinguir claramente un límite a partir del cual los valores se desequilibraban. Este límite fue de 25000 vuelos para el cálculo de las variables promediadas anuales, como es el caso de la proporción y duración promedio de los retrasos para cada aeropuerto provocados por el propio aeropuerto, así como 2500 para los

datos mensuales de proporción y promedio de vuelos retrasados por el clima.

Tiempo

Creemos que la mayor parte si no toda la influencia del tiempo queda correctamente descrita, con las sustituciones que hicimos relativas al clima y la saturación de los aeropuertos, puesto que, intuitivamente que otra razón podría haber, la industria aeroportuaria y más a la escala donde estamos trabajando (solo con las aerolíneas que transportan más del 1 porciento del tráfico aéreo de EEUU) está amortizada contra todo tipo de fecha especial. Quizás solo la variable de retrasos por seguridad aumente, por ejemplo, en fechas cercanas a fechas históricas como el 4 de Julio o el 11 de Septiembre. Pero realmente el volumen de vuelos retrasados por seguridad nos pareció lo suficientemente pequeño como para ignorarlo, además de que todo lo hemos hecho contrarreloj. De lo contrario, hubiésemos expresado las cancelaciones relacionadas con motivos de seguridad, añadiendo una columna con el riesgo en seguridad de la semana en cuestión.

La variable tail number no la necesitamos durante el entrenamiento luego de la codificación de la data, ya que no se la pasaremos al modelo. Habrá otro campo llamado ORIENTATION que contiene el tipo de operación a ralizar DEPARTURE o ARRIVAL, que se usará para separar la data en caso de que se quiera entrenar un modelo para las departure y otro para los arrival (lo cual fue en efecto lo que hicimos).

Al concluir la parte de limpieza de la data y Feature Engineering quedarán para entrenar al modelo y para describir a los futuros datapoints a describir los siguientes features:

TFD

PFD

SATURATION : saturación del aeropuerto donde se produce la operación en cuestión.

AIRLINE DELAY PROB : proporción de retrasos inducidos por la aerolínea en cuestión

AIRLINE AVG DELAY : promedio de duración de los retrasos inducidos por esta aerolínea

ORIGIN AIRPORT DELAY PROB : proporción de retrasos inducidos por el aeropuerto de partida

ORIGIN AIRPORT AVG DELAY : promedio de duración de los retrasos inducidos por el aeropuerto de partida

ORIGIN WEATHER DELAY PROB : proporción de retrasos inducidos por el clima en el aeropuerto de salida en el mes en cuestión.

ORIGIN AVG WEATHER DELAY : promedio de duración de los retrasos inducidos por el clima en el aeropuerto de partida en el mes en cuestión

DESTINATION AIRPORT DELAY PROB : proporción de retrasos inducidos por el aeropuerto de llegada

DESTINATION AIRPORT AVG DELAY : promedio de duración de los retrasos inducidos por el aeropuerto de llegada

DESTINATION WEATHER DELAY PROB : proporción de retrasos inducidos por el clima en el aeropuerto de salida en el mes en cuestión

DESTINATION AVG WEATHER DELAY : promedio de duración de los retrasos inducidos por el clima en el aeropuerto de llegada

Predicción de resultados de vuelos futuros:

El modelo que se use (los modelos que probamos dan muy buenos resultados), estará diseñado para dadas las categorías mencionadas, predecir el delay del vuelo, independientemente del modelo en que se haya producido el vuelo o de la secuencia de vuelos precedente. De hecho, de eso se trata el procesamiento de la data, de embeber en cada datapoint toda la información relativa a los eventos temporales de manera que se vuelva independiente. Por tal motivo pudimos probar el modelo con datapoints aleatorios separados de la data justo al iniciar.

Pero el hecho de que hayamos abstraído la temporalidad, no implica que nuestro modelo no sea capaz de predecir series de vuelos, por el contrario, está diseñado para ello. Cómo predeciríamos, por ejemplo, el retraso de todos los vuelos del mes siguiente?.

Primero que todo, haríamos un preprocesaddo sobre lo que sabemos de esos vuelos, sus fechas de salida y llegada y sus aeropuertos de salida y llegada y la aerolínea. Al igual que en el preprocesado original, los dividiríamos en dos operaciones y luego los ordenaríamos por número de cola del avión y por fecha. Dado que nuestro modelo fue entrenado con la data de todo un año, tenemos los datos climáticos necesarios y las demás datas categóricas ya sabemos como codificarlas. Solo dejaríamos el feature Saturación y el feature PFD sin calcular (el TFD puede ser calculado usando las horas previstas solamente dato que tendríamos desde el principio).

Nos ampararíamos en la data que ya tenemos sobre el último vuelo realizado por el avión en cuestión para calcular el TFD y el PFD para la primera operación que aparezca para cada avión en la data a predecir. Si el avión no lo teníamos en nuestro dataset anterior, sencillamente asignamos cero a ambos campos. Por otro lado para calcular la saturación, mantendremos siempre la información de cuantos aviones están realizando una operación en

cada instante de tiempo en cada aeropuerto, partiendo de que tal data que ya fue computada para el dataset histórico y la iremos incrementando vuelo a vuelo a medida que realicemos las predicciones en orden cronológico. O sea tanto el PFD como la saturación de cada data point a predecir se calculan en el momento de su predicción, a partir de los datos históricos precomputados incrementados con las nuevas predicciones que hemos realizado.

De esta manera se puede predecir hasta N vuelos después de la última fecha en los datos históricos, con la correspondiente pérdida de precisión a medida que avanza el tiempo por los errores en cálculos de delay acumulados, que sin embargo estará amortiguada, por el hecho de que la programación de los vuelos, está diseñada de tal manera que muy rápido un avión alcanzará una pausa larga donde los delays se resetean por decirlo de alguna manera, por lo que los errores en predicciones de delay quedarán acotados por las pausas 'largas' que se toman los aviones. No era nuestro objetivo primario probar nuestro modelo bajo estas situaciones, por motivos de tiempo principalmente, sino presentar la introducción de los nuevos features y el como era posible obtener muy buenos resultados de precisión usando estos features. No obstante, remarcamos, la eficacia de esta forma de predecir vuelos estará dada mayormente por el modelo para predecir datapoints individuales independientes, y este sí pudimos probarlo y obtuvo resultados muy buenos.

Detalles particulares de la limpieza de datos que realizamos

Usamos el dataset provisto en el siguiente enlace https://www.kaggle.com/datasets/usdot/flight-delays, porque fue el único que encontramos en un principio que contaba con TAIL NUMBER para cada vuelo.

De primera y pata este dataset tiene un problema insalvable con el mes de octubre, en todos los vuelos del mes de Octubre los aeropuertos aparecen codificados usando números de cinco dígitos, a diferencia del resto del dataset donde se usa código IATA. Esto fue un problema señalado por otros usuarios de Kaggle numerosas veces, para el que no encontramos solución, y la investigación no arrojó de donde provenía este código de 5 dígitos. Luego, nos deshicimos de todos los vuelos de Octubre.

Luego nos deshicimos de la data que contenía datos null en las columnas esenciales. Esto no representa un problema, puesto que representaban alrededor de 100mil datos en un dataset con 5 millones 300 mil, y el enfoque adoptado no necesita de largas secuencias de vuelos encadenados, sino de que los datos que estén sean precisos.

El siguiente problema vino con la codificación temporal. Las fechas venían expresadas de una manera muy poco conveniente. En columnas diferentes

aparecían mes, día, y las horas planificadas de salida y de llegada en formato militar. Esta abominación provocaba que por ejemplo, si un vuelo salia hoy y llegaba mañana debíamos adivinarlo a partir del hecho de que la hora de llegada iba a ser menor que la hora de partida. Decidimos, codificar todas las fechas con un dato único y radical, la cantidad minutos pasados desde que comenzó el año. Para esto debimos hacer un trabajo excepcionalmente cuidadoso identificando posibles deformaciones resultantes del hecho comentado de que no teníamos otra forma de conocer el día exacto en que un vuelo salio y llego mas allá de que la diferencia entre las fechas fuera irracional.

Dado que el día asignado a cada vuelo corresponde al día en que estaba previsto que saliera, identificamos cuatro fuentes de conflictos:

- Vuelos que se planean que lleguen mañana
- Vuelos cuya salida ocurrió al día siguiente
- Vuelos cuya salida ocurrió el día anterior
- Vuelos cuya llegada ocurrió al día siguiente (aunque estaba prevista para hoy)

que provocarían las siguientes distorsiones respectivamente una vez transformadas las fechas a minutos pasados desde el 1ero de enero:

- SCHEDULED DEPARTURE >SCHEDULED ARRIVAL
- DEPARTURE TIME >ARRIVAL TIME, DELAY $\approx$ 1440
- DELAY $\approx$ -1440
- DELAY $\approx$ -1440

que por tanto debemos arreglar aplicando respectivamente los siguientes ajustes:

- SCHEDULED ARRIVAL += 1440
- DEPARTURE TIME -= 1440
- DEPARTURE TIME += 1440
- ARRIVAL TIME += 1440

Aun después de esto debemos eliminar los vuelos que atraviesan las horas en que ocurrió los cambios de fecha del 2005, para evitar distorsiones, son pocos, 294.

Luego nos deshicimos de esos datos que tuvieran problemas en su calculo del delay ya fuera a la partida o a la llegada, 220 mil vuelos cumplieron esta propiedad, de manera que los descartamos y nos íbamos quedando por el momento con 5 millones de datos

Luego de esto, para cada avión calculamos la cantidad de tiempo que paso en tierra en cada aeropuerto siempre que pudimos (o sea que aparecían vuelos concatenados para este avión, donde terminaba en el aeropuerto A y

comenzaba en A precisamente). Por supuesto, estancias en tierra de duración negativa no tienen sentido, pero aparecieron cerca de 1 millón de datos con duración en tierra negativa.

Pensamos un poco en el asunto y se nos ocurrió que muy probablemente se debiera a los diferentes husos horarios en los aeropuertos de EEUU, por lo cual realizamos un experimento para demostrarlo. De ser así, sucedería que todos los vuelos con estas características ocurrirían cargando consigo una modificación horaria del mismo signo, por lo que nos lanzamos a calcular la modificación horaria de cada vuelo. Esto represento un reto de por si, ya que no existía dataset en Kaggle que asociara aeropuerto con uso horario, por lo cual creamos uno que esperemos sirva a la comunidad en el futuro. Sin embargo, al estudiar la modificación horaria de los vuelos con duración en suelo negativa, resulto que la mayor parte ni siquiera se movía de zona horaria, por lo que quedo descartada la hipótesis de las zonas horarias.

Para verificar que no fuera un error nuestro en la conversión de tiempo, asociamos estos vuelos a sus datos originales y pudimos apreciar que estos datos originales tenían errores claros de hora de llegada y de partida, muchas veces anotando su llegada horas antes de su partida. Sin más nos deshicimos de estos datos.

En resumen, nos deshicimos de poco mas de 300 mil de filas de 5 millones 300 mil inicialmente, lo cual no nos parece mal.

Inicialmente pensamos usar un k-fold con k=10 para realizar cross validation sobre nuestro modelo, pero luego nos dimos cuenta que por la cantidad enorme de datos no sería necesario, y en efecto, cuando lo probamos para la sustitución categórica de las aerolíneas, los valores de fold a fold tenían una varianza del orden de las millonésimas, de modo que decidimos no usar cross-validation en lo absoluto.

Nos propusimos realizar un preprocesamiento de la data que nos permitiera mantener el modelo simple, y que fuera lo más expresivo relativo a los datos disponibles sobre cada vuelo con la anterioridad que sea.

Nuestro modelo, sabiendo apenas unos features que están disponibles previamente al vuelo, puede predecir el retraso del vuelo en cuestión, y a la hora de predecir un largo volumen de vuelos futuros, cada vez que predizcamos el retraso de uno de los vuelos, este se tomará en cuenta para calcular el retraso de los próximos. De esta manera, se pueden predecir hasta n vuelos futuros, pudiendo ser n tan grande como se quiera, solo sabiendo su origen y su destino y sus horarios previstos de salida y de llegada.

Kaggle (https://www.kaggle.com/datasets/usdot/flight-delays)

Para demostrar el efecto de nuestro preprocesamiento de la data, separamos la data en 80 porciento entrenamiento y 20 porciento test de manera aleatoria, y testeamos cuan bien el modelo predice los casos de test, con todos sus features calculados excepto el delay por supuesto. Los features e encontraran calculados y esto no influye en la calidad de la prueba, porque el modelo en la vida real cada vez que vaya a predecir la duración de una operación aeroportuaria tendrá los features calculados, ya sea a partir de la data histórica, o basados en los delays previstos para los vuelos a predecir anteriores. Los valores de los features promediados por los que se sustituyen los datos categóricos se calculan usando solo los datos del conjunto de pruebas, no obstante que si lo hubiéramos hecho usando todos los datos en general no se produciría data leakeage porque los datos han demostrado ser homogéneos cuando hemos tratado de separarlos por folds por ejemplo.

5 Experimentos y resultados

Se realizaron experimentos para evaluar la factibilidad de la introducción de los datos ya mencionados, para esto se entrenaron diversos modelos con el 80 porciento de los datos con los que se contaba y se realizó test con el 20 porciento restante.

Los modelos utilizados fueron Linear Regression (LR), Random Forest (RF), Gradient Boosting (GB), KNN y Ridge Regression (RR), las métricas utilizadas fueron el Error Absoluto Medio (MAE), el Error Cuadrático Medio (MSE) y la precisión con respecto a tipos de retraso, definiendo como resultado positivo si la medición real y la predicción son menores o mayores al mismo tiempo que un valor k (Accuracy <k).

Métrica	LR	RF	GB	KNN	RR
MAE	13.36	11.15	17.44	12.6	13.36
MSE	670.43	528.13	1075.13	688.44	670.43
Accuracy (< 15)	88.96	87.38	83.85	89.28	88.96
Accuracy (< 60)	96.48	97.05	95.03	96.95	96.48
Accuracy (< 90)	97.83	98.41	97.26	98.27	97.83

Como se puede observar los modelos presentan un error de poco más de 10 minutos de predicción, sin embargo la identificación del tipo de retraso es muy efectivo y permite estar preparado en una gran cantidad de casos.

Pero queríamos identificar si la separación de la data en distintas clases nos permitia obtener mejores resultados, por lo que separamos los datos en "Arrivals" y "Departures" y analizamos los resultados en todos los modelos probados.

	LR		RF		GB		KNN		RR	
Métrica	Arrival	Departure	Arrival	Departure	Arrival	Departure	Arrival	Departure	Arrival	Departure
MAE	9.00	14.20	9.30	12.38	18.52	16.39	11.87	10.95	9.00	14.20
MSE	163.88	915.58	176.02	840.55	1098.09	1041.61	283.81	870.36	163.88	915.58
Accuracy (< 15)	93.20	84.67	92.36	82.86	84.21	84.39	92.05	88.57	93.20	84.68
Accuracy (< 60)	98.68	95.06	98.55	95.86	95.06	95.08	98.37	96.53	98.68	95.06
Accuracy (< 90)	99.33	97.21	99.27	97.76	97.23	97.28	99.14	97.98	99.33	97.21

Luego de separar en clases decidimos profundizar en los hiperparametros de los modelos utilizados.

En el caso de *Random Forest* se aumentaron la cantidad de estimadores:

Métrica	10 Estimadores	20 Estimadores
MAE	11.15	10.82
MSE	528.13	503.42
Accuracy (< 15)	87.38	87.88
Accuracy (< 60)	97.05	97.2
Accuracy (< 90)	98.41	98.49

Por otro lado en *Gradient Boosting* se aumentaron distintos hiperparámetros por separado:

Métrica	Normal	Iteraciones	Profundidad	Aprendizaje
MAE	17.44	16.65	17.4	14.65
MSE	1075.1	1000.38	1091.29	790.86
Accuracy (< 15)	83.85	86.38	84.05	87.97
Accuracy (< 60)	95.03	95.06	95.03	95.65
Accuracy (< 90)	97.26	97.27	97.26	97.35

Finalmente en KNN se aumentaron la cantidad de vecinos observados:

Métrica	5 Vecinos	10 Vecinos
MAE	12.6	12.02
MSE	688.44	687.93
Accuracy (< 15)	89.28	89.26
Accuracy (< 60)	96.95	96.96
Accuracy (< 90)	98.27	98.29

6 Discusión de los resultados

El trabajo con distintos tipos de métricas (métricas de error o precisión) nos resultó necesario dado el estudio de trabajos similares donde normalmente se presentaba uno solo de estos, imposibilitando totalmente una comparación de resultados entre trabajos (si bien es cierto que muchos se realizan con distintas bases de datos, la diferenciacion de métricas separaba más esta comparación).

Al abrir los resultados en las clases Arrivals y Departures se pudo identificar que es más complicado para nuestro modelo determinar el retraso en las entradas a un aeropuerto, por ende estos son los datos que hacen que las métricas con toda la base de datos (Primera Imagen) aumenten los errores y disminuyan la precisión con respecto a las salidas.

Por otro lado los cambios de hiperparámetros representaron un mayor costo computacional pero no reflejaron un gran cambio en las métricas utilizadas (excepto en el caso de Gradient Boosting).

7 Conclusiones y trabajo futuro

Creemos que el hecho de que la generalidad de los modelos, a pesar de su sencillez, hayan obtenido buenos resultados, demuestran que los encodings y los features usados fueron correctos. Creemos que nuestros resultados prueban que se puede montar un sistema de predicción de retrasos de vuelos partiendo de una data acuciosa y un procesamiento inteligente de ella. Confirmamos que la saturación es un feature poderoso y a la vez es sencilla de obtener para vuelos futuros en nuestra propuesta de solución.

Trabajo Futuro

Dado el poco tiempo, no pudimos llegar a montar el sistema de predicción y lanzarnos a predecir la data, por ejemplo de Enero del 2016, quisiéramos hacerlo, aunque no era uno de los propósitos de este paper

Quisiéramos explorar introducir nuevos features temporales relacionados, por ejemplo, con los retrasos relacionados con motivos de seguridad que quedan sin consideraren este trabajo.

8 Bibliografía

Papers consultados:

- Flight Delay Regression Prediction Model Based on Att-Conv-LSTM, https://www.semanticscholar.org/paper/Flight-Delay-Regression-Prediction-Model-Based-on-Qu-Xiao/7f90be96f0dec2c7a3d3cd8b949dbf405221641c

- Flight Delay Prediction Model Based on Lightweight Network ECA-MobileNetV3, https://www.semanticscholar.org/paper/Flight-Delay-Prediction-Model-Based-on-Lightweight-Qu-Chen/97d41eb9d82629036003fc344e33a9a0bbecf09a

- Social ski driver conditional autoregressive-based deep learning classifier for flight delay prediction, https://www.semanticscholar.org/paper/Social-ski-driver-conditional-autoregressive-based-Bisandu-Moulitsas/7ee4a99634ddb65be1fc4d92cad335da156c5202

- Predicting flight delay based on multiple linear regression, https://www.semanticscholar.org/paper/Predicting-flight-delay-based-on-multiple-linear-Ding/52c72862fe875a3e1f0c0b707135988f6fd5a356

- Flight Delay Classification Prediction Based on Stacking Algorithm, https://www.semanticscholar.org/paper/Flight-Delay-Classification-Prediction-Based-on-Yi-Zhang/199bf570d6e9a19722bc85274a4e588acdddfdd2

- Predicting Flight Delay Using KNN, https://www.semanticscholar.org/paper/Predicting-Flight-Delay-Using-KNN-Mr.M.Saravanakumar-Prasath.S/eff9640acf2d79682a7c657db63c1651fbb88602

- Alexa, Predict My Flight Delay, https://paperswithcode.com/paper/alexa-predict-my-flight-delay

- Predicting Flight Delay with Spatio-Temporal Trajectory Convolutional Network and Airport Situational Awareness Map, https://paperswithcode.com/paper/predicting-flight-delay-with-spatio-temporal

- Flight Delay Prediction Using Gradient Boosting Machine Learning Classifiers, https://www.semanticscholar.org/paper/Flight-Delay-Prediction-Using-Gradient-Boosting-Lu-Wei/ebb0818bad54617454927bd6c459889163afd445

- A novel intelligent approach for flight delay prediction, https://www.semanticscholar.org/paper/A-novel-intelligent-approach-for-flight-delay-Mamdouh-Ezzat/a8bb623dbefa42f8efc680bec5884e3ceb76e65d

- Modelo para identificar los vuelos afectados por retrasos o cancelaciones en el aeropuerto El Dorado de Bogotá, Colombia,

https://www.semanticscholar.org/paper/Modelo-para-identificar-los-vuelos-afectados-por-o-Quiroga-Cely/ee441baf9106d231fe18efce21858c046d9e8cd7