



**Prototipo para predecir operaciones aéreas con el fin de optimizar turnos laborales
para los controladores en el aeropuerto Camilo Daza**

Elaborado por:

Harold Felipe Zuluaga Grisales

Ingeniería de sistemas

César Alejandro Bustos Moreno

Ingeniería de sistemas

Sandra Liliana Martín Fúquene

Ingeniería de sistemas – Ingeniería de producción

Universidad EAN

Proyecto de integración

Bogotá, 2022



Resumen Ejecutivo

El objetivo es generar una solución para programar turnos de los controladores aéreos del aeropuerto Camilo Daza de la ciudad de Cúcuta, por medio de la predicción de vuelos, creando un modelo analítico que permita identificar patrones, y tomar decisiones de una manera eficiente por medio del machine learning.

La importancia repercute en decidir la configuración más óptima sin poner en riesgo la seguridad aérea, de igual forma controlar la cantidad de horas controlador con el fin de mejorar los tiempos de descanso y mejorar los turnos laborales.

Con la recopilación de datos se pretende crear un modelo el cual nos genere predicciones, por medio de la cual permita conocer los tiempos con mayor demanda con el fin de determinar el capital humano acorde a este postulado; la importancia de la implementación de este proyecto radica en la utilización de herramientas 4.0 para la toma de decisiones, permitiendo la optimización del recurso humano, aumentando la productividad de los controladores aéreos, su desconexión laboral y la seguridad de las personas.

Introducción

Desde hace varios años se han creado procedimientos para mantener la aviación como un medio de transporte seguro. Según lo menciona la aeronáutica del Ecuador (2008), “La seguridad de la aviación es el objetivo más importante de la Organización de Aviación Civil Internacional”.

Entre las herramientas más exitosas empleadas de cara a la reducción de accidentes está el análisis profundo el comportamiento de las máquinas, como los aviones y el capital humano. Gracias a esto, las aeronaves han evolucionado por medio de la tecnología a tal punto que su incidencia en accidentes es mucho inferior a la del ser humano.

Los pilotos son agentes que tienen alta implicación en el desarrollo de las operaciones aéreas. Sin embargo, no son los únicos. Entre otros grupos que interfieren están los ingenieros de vuelo, mecánicos de aviones, personal en tierra de cada aerolínea y controladores de tránsito aéreo.

La principal función de este último es prevenir colisiones entre aeronaves. Para ello tiene que llevar a cabo una serie de procedimientos que permitan garantizar un mínimo de separación entre cada tránsito como, por ejemplo: La planeación con anterioridad, toma de decisiones generalmente bajo presión y manejo de las emociones para afrontar situaciones imprevistas. Lo que la hace una labor que genera fatiga y estrés debido a la gran cantidad de personas que están bajo su responsabilidad. Adicionando la obligación legal que asumen cada vez que toman una decisión. Esto puede deteriorar la salud física y mental de los trabajadores a la vez que reduce la seguridad operacional.

A lo largo del tiempo se han realizado estudios de factores humanos en los controladores. Los hallazgos encontrados fue la afectación que sufre el cuerpo para obtener descanso y recuperarse cuando se trabaja por turnos rotativos. El cuerpo humano ha evolucionado para trabajar de manera óptima durante el día y bajar sus niveles de energía en las noches para dormir.

Entre las recomendaciones para mitigar esta afectación está en que no se puede trabajar más de cierto número de turnos de noche por semana de acuerdo con las necesidades de cada



aeropuerto igual que después de un turno nocturno no debería tener menos de 24 horas para su descanso antes de volver a la operación. Adicionalmente, menciona que la cantidad de trabajadores por turno debe ser proporcional a la exigencia. Por lo que en las horas con mayor densidad debería haber más fuerza laboral, (skybrary, s.f.). Llevar a cabo esas recomendaciones permiten mantener un desempeño óptimo en sus actividades, así como un beneficio para el cuidado de su salud.

Su trabajo por turnos permite distribuir la carga laboral equitativamente. Una mala planificación del capital humano puede aumentar eventos no deseados como incidentes o accidentes. Más allá una mala planificación regularmente ocurre por variables que están fuera de control. Entre ellas: Aumento de operaciones a causa de aeropuertos cercanos que cierran debido a condiciones meteorológicas; Incremento de operaciones de aviación no regular, como lo son los vuelos ambulancia, por temas de salubridad en áreas cercanas; O la reducción del personal para cumplir actividades administrativas en un momento donde hay un imprevisto aumento de operaciones.

Si bien se han realizado estudios para mejorar la asignación de controladores que trabajan por turnos haciendo uso de la metaheurística, estos no han hecho mejoras sustanciales. (Suarez, 2017)

En el presente trabajo se tiene como propósito diseñar una aplicación que permita predecir la cantidad de aterrizajes y despegues mensuales en el aeropuerto de Cúcuta para que con base a este pronóstico se pueda diseñar una planilla de turnos en la que la carga laboral sea distribuida equitativamente y siempre la densidad de tránsito sea acorde con la cantidad de personal.

El lenguaje de programación que se va a utilizar para el desarrollo del trabajo es Python. Se pretende hacer uso de sus librerías y características para visualización de la información y manipulación de datos. Esto permitirá generar predicciones por medio de aprendizaje autónomo o Machine learning en inglés.



Por último, como propósito final se desea ser referente para que más aeropuertos a nivel nacional e internacional hagan uso de esta aplicación y puedan distribuir su recurso humano de manera eficaz.

El documento se encuentra estructura de la siguiente manera: El capítulo de introducción, seguido de los la sección de objetivos, la descripción del problema, la respectiva justificación, el capítulo del marco teórico, continuando con el análisis de restricciones, seguido de la metodología para la selección y desarrollo de la solución, luego el análisis de costos, la sección descriptiva de la solución de ingeniería, seguido del modelo de machine learning, continuando con la evidencia del aplicativo web, finalizando con el capítulo de las conclusiones y la sección de referencias.

Objetivos

Objetivo general

Crear una solución basada en un modelo de predicción de Machine Learning, que permita pronosticar la densidad de operaciones y poder así optimizar la programación de turnos de los controladores aéreos en el aeropuerto Camilo Daza de la ciudad de Cúcuta.

Objetivos específicos

- Realizar un proceso de ETL (extracción, transformación y carga) en la base de datos utilizada en el modelo de Machine Learning.
- Generar un análisis exploratorio de los datos que permitan conocer las variables a utilizar.
- Diseñar el modelo de ML (machine learning) a través de Python que permita establecer una alta precisión en la predicción de la densidad del tráfico aéreo.
- Presentar un pronóstico de manera gráfica de hasta 3 meses que indique si la densidad de tránsito aumenta o disminuye respecto a un mes anterior.

Descripción Del Problema

De acuerdo con la revista de negocios American economy (2016), el control de tránsito aéreo está entre los trabajos más estresantes del mundo. Una de las causas para que esto ocurra es el constante análisis para tomar decisiones con un alto grado de responsabilidad. De acuerdo con la organización internacional de la Aviación Civil (2007), el objetivo de cada decisión tomada por un controlador es prevenir colisiones entre aeronaves y acelerar y mantener ordenadamente el movimiento del tránsito. Por tanto, es el que asume todo compromiso moral y legal de sus decisiones, no la empresa. Es por esto que el control de tránsito aéreo es catalogado como un trabajo exigente y de alto riesgo a nivel mundial.

En cuanto a la toma de decisiones, un componente importante que apoye a esta tarea es la planeación. Esta actividad no solo incluye las acciones que debe tomar cada controlador para prever qué instrucciones dar a cada tránsito en particular, sino también la cantidad de personal disponible en un momento dado.

Actualmente, en la ciudad de Cúcuta la programación de turnos es realizada cada 15 días y modificadas constantemente para suplir las necesidades del servicio. Esta labor se lleva a cabo de forma manual lo que la hace un poco dispendiosa dando paso a errores humanos, mayor carga laboral y consumo de más tiempo. Tampoco existen pronósticos para tener una idea de la cantidad y tipo de aeronaves con anterioridad. Además, la densidad del tránsito es muy variable a lo largo del año por razones como condiciones meteorológicas o cierres en aeropuertos cercanos.

Es importante tener en cuenta que un aumento imprevisto de aeronaves exige que el controlador aplique un mayor número de procedimientos, ocupe la frecuencia para dar más información a todos los vuelos a cargo, y realice más coordinaciones con otros colegas interesados. Consecuentemente, la carga laboral se incrementa requiriendo más atención y concentración para mantener niveles de seguridad aérea aceptables.

Por ahora no existe un software que pueda dar solución al inconveniente, ni los recursos para un proyecto local como éste. Esto a menudo ocasiona que la lista de turnos sea alterada a última hora para suplir la demanda del servicio o bien que haya programado una cantidad de personal elevado para momentos con una afluencia de tránsito baja.

Como base a la planteado se propone la pregunta investigativa:

¿Qué aporte puede hacer el aprendizaje automático en la programación de turnos para los controladores de tránsito aéreo en el aeropuerto Camilo Daza de Cúcuta?

Justificación

La realización de la lista de turnos es una actividad importante ya que permite distribuir el recurso humano para prestar un servicio acorde con la cantidad de aeronaves. Debido a que las herramientas que se disponen para realizar dicha actividad no toman en cuenta la densidad de tránsito, el resultado es una planilla de turnos que no siempre satisface con el personal para cumplir con la demanda de tránsito en un momento dado.

Por lo anterior, se plantea diseñar un prototipo de software para predecir cantidad y tipo de aeronaves en intervalos de tiempo específicos a través de un aplicativo intuitivo que permita manipular información e interactuar con el sistema. Esto permitirá que: 1) Exista una distribución de controladores de acuerdo con las necesidades del servicio y menos alteraciones en la lista de turnos; 2) facilite mantener niveles de seguridad aérea altos ya que el incremento de la carga laboral esperado es contrarrestado con el aumento del personal; 3) prever un incremento en las llegadas. Esto permitirá racionalizar la plataforma de parqueo de una mejor manera para usar la pista solo para aterrizajes y despegues, qué es lo ideal. Considerando que este espacio es pequeño y se puede presentar ocasiones en las que se debe usar la segunda pista para ubicar aviones; 4) determinar las temporadas con baja afluencia para llevar a cabo actividades relacionadas con el trabajo sin que afecte las operaciones. Estas actividades pueden ser:



reentrenamientos en la escuela que deben tomar los controladores periódicamente, cursos y diplomados ofrecidos por la misma empresa para fortalecer el conocimiento en diferentes áreas de la aviación y que esto se vea retribuido en el servicio o para programar las vacaciones sin quedar corto de personal. Es por esto por lo que se considera valioso diseñar e implementar este sistema de predicción de densidad de tránsito.

El impacto de una solución al problema en cuestión va más allá de los anteriormente mencionados. Por un lado, las personas a cargo de controlar el tránsito contarán con un horario más estable lo que les permite planear y llevar a cabo sus actividades familiares sin que éstas sean interrumpidas por obligaciones laborales. Adicionalmente, se espera ser un referente para que demás aeropuertos nacionales e internacionales puedan hacer uso de esta herramienta y puedan así distribuir la carga laboral por medio de la asignación de turnos de una manera acorde a las necesidades del servicio. Resultando en mejores índices de seguridad y una mejora en la salud de sus trabajadores.

Marco Teórico

1. Control de tránsito aéreo

Cómo lo explica la entidad reguladora de la Aviación Civil en Colombia (s.f.):

"El Controlador de Tránsito aéreo, es un profesional encargado de proveer los servicios de tránsito aéreo (Información de vuelo, alerta y Control), de acuerdo con las características propias del espacio en donde se desarrolle cada operación (movimiento aéreo). Su capacitación, entrenamiento y desarrollo de competencias, asegura un eficiente y adecuado desempeño en situaciones de alta complejidad operacional, debiendo tomar decisiones en fracciones muy cortas de tiempo. Debe manejar adecuadamente los procesos de comunicación de forma asertiva en el idioma español e inglés, validados de acuerdo con las normas internacionales, realizando su labor con altos niveles de atención y concentración para optimizar la eficiencia de las operaciones, con un altísimo nivel de seguridad operacional"

Adicionalmente, según indica Piña y Antonio (2012, p.41):

“los controladores de tránsito aéreo o CTA realizan labores que implican conocimiento, alta experiencia y habilidades blandas, entre las que destaca la comunicación. Además, menciona que, los CTA se forman como sujetos que deben tomar decisiones en ambientes laborales de alto riesgo y contingentes”

Por lo anterior, ser controlador es una tarea que permite mantener unos estándares altos de seguridad aérea gracias a un personal que es capacitado constantemente para atender las necesidades del servicio. Adicionalmente, no existe una habilidad que será clave para desarrollar esta actividad. En cambio, se requiere un grupo de cualidades entre las que se encuentran toma

de decisiones bajo presión, comunicación efectiva, manejo del estrés y capacidad de mantener altos niveles de atención.

1.1. Objetivos del control de tránsito aéreo

Entre los principales objetivos de los controladores de tránsito aéreo Cabrera y Beltrán (2020, p.25-26) mencionan:

“

- a) Prevenir colisiones entre aeronaves.
- b) Prevenir colisiones entre aeronaves en el área de maniobras y entre esas y los obstáculos que haya en dicha área.
- c) Acelerar y mantener ordenadamente el movimiento del tránsito aéreo.
- d) Asesorar y proporcionar información útil para la marcha segura y eficaz de los vuelos.
- e) Notificar a los organismos pertinentes respecto a las aeronaves que necesitan ayuda de búsqueda y salvamento, prestando la mayor colaboración posible a dichos organismos según sea necesario.”

Su labor es compleja a causa del denso tránsito de aeronaves, el posible deterioro de la meteorología y otros imprevistos. Los controladores de tránsito aéreo se seleccionan entre personas con gran percepción y ubicación espacial, recibiendo un intensivo entrenamiento con ayuda de profesores altamente calificados con mucha experiencia laboral en el área y simuladores especializados para controladores de tránsito aéreo.

2. Factores humanos en la aviación

La organización de la Aviación Civil (ICAO, 2007, p.A5) afirma que: “Los factores humanos se definen como un campo multidisciplinario, dedicado a la optimización del rendimiento humano para que los errores no se repitan y la reducción de este en las operaciones aéreas.”

También menciona que:

“Su objetivo es la eficacia y eficiencia del sistema para la seguridad aérea y el bienestar del individuo.

Eficiencia: para obtener los resultados requeridos sin dificultad o esfuerzo excesivo por parte del operador.

Seguridad: Mínimo de error en el procesamiento de la información aeronáutica.

Motivación: Adaptación del hombre a su medio ambiente (Entorno Laboral).”

El psicólogo Edwards en 1972, seguido por Hawkins en 1975 amplían el conocimiento en el área creando el modelo SHELL.

“La clave de este modelo es la de colocar al ser humano como referencia central, reconociendo a las limitaciones propias que inciden en el desempeño de sus tareas, e imponiendo un ajuste al resto de componentes del modelo a fin de estudiar su eficiencia en su ambiente de trabajo. El ambiente y la máquina son los que han de acomodarse al hombre y nunca a la inversa.” (Acuña, 2013, p.18)

Por lo anterior, el estudio de factores humanos busca sistemáticamente mejorar el rendimiento del capital humano para que las fallas no sean recurrentes. Para ello, se implementan mejoras al sistema para que el manejo y el procesamiento de datos sea fácil y ágil.

2.1. Evolución de los Factores humanos

A medida que las técnicas estudiadas en factores humanos fueron probadas con el tiempo, a finales de la década de los 80, se decidió trabajar este tema a más profundidad. Esto sucedió debido al aumento de los accidentes aéreos. Es allí donde se creó el concepto de gestión de recursos en cabina, CRM por sus siglas en inglés.

Daniel Muñoz-Marrón (2018, p. 192) menciona que “El CRM puede definirse como la óptima utilización, por parte de una tripulación, de todos los recursos disponibles (información, equipos materiales y recursos humanos) para la consecución de operaciones de vuelo seguras y eficientes (Lauber, 1984). Diseñado para reducir el error e incrementar la efectividad de las tripulaciones aéreas (Wiener, Kanki y Helmreich, 1993)”

La primera generación del CRM fue la introducción del concepto como tal, posteriormente va evolucionando, mejorando sus prácticas y abarcando diferentes áreas de la aviación.

Respecto a las siguientes generaciones: Daniel Muñoz-Marrón menciona:

“Los programas de la segunda generación incorporan la aplicación práctica de diferentes conceptos a la dinámica de trabajo que se produce entre los componentes de la tripulación, tales como dinámicas de grupo en cabina, orientadas fundamentalmente hacia el trabajo de la tripulación como equipo, y al entrenamiento en estrategias de toma de decisiones o cómo romper la cadena del error”. (Muñoz, 2018)

En cuanto a la tercera generación:

“En la tercera instancia evolutiva, se abrieron necesariamente los conceptos de entrenamiento al resto de los actores del sistema aeronáutico: despacho operacional, control del tránsito aéreo y aeronavegabilidad. Por ejemplo, en el área técnica comenzaron a desarrollarse los programas MRM (Maintenance Resource Management).” (investigar para prevenir, 2018)



La cuarta generación, según Daniel Muñoz-Marrón (2018): “en el año 1990, la F.A.A. da un paso más y convierte el CRM en requisito reglamentario para todas las líneas aéreas. El A.Q.P. permite crear entrenamientos CRM a medida, posibilitando que cada sector haga frente a sus necesidades específicas. Además, como parte de la integración del CRM en las operaciones, muchas compañías comienzan a protocolizar procedimientos”

Por último, respecto a la quinta generación dice que es en la cual se trabaja actualmente y se centra en el ser humano como parte fundamental para evitar que los eventos no deseados ocurran. Su estrategia se basa en crear barreras o mecanismos de defensa que permitan romper la cadena del error. Se parte considerando que la posibilidad de errar o de que ocurra una falla siempre está latente por tanto se deben crear protocolos para que se puedan identificar con facilidad y remediarlas de la mejor manera. (Muñoz, 2018)

Debido a la importancia que se le da a la actuación humana en el estudio de factores humanos, los controladores diseñan su propia planilla de turnos con el fin de tener en cuenta las preferencias de cada uno en la medida de lo posible. Más allá, estas son acciones que crean barreras para reducir el error proveniente del estrés laboral generadas por condiciones propias del trabajo. Para ilustrarlo, un controlador que tenga familia preferirá trabajar turnos entre semana, pero descansar fines de semana. Este trabajador a fin de lograr lo que quiere puede coordinar con otro compañero que no tenga la familia en el lugar donde trabaja para que cubra sus turnos algunos fines de semana. Su compañero a cambio de trabajar los fines de semana podría lograr coordinar algunos días de descanso contiguos para visitar a su familia que se encuentre en otra ciudad. Este episodio puede ser común ya que muchos controladores son enviados a aeropuertos, diferentes a su lugar de origen, por necesidades del servicio. Estos hechos en los que el controlador es involucrado y tenido en cuenta en el momento de realizar la planilla de turnos pueden mejorar el rendimiento de los trabajadores, reducir la fatiga laboral y aumentar la seguridad en las operaciones tal cómo concluye la publicación de manejo de turnos de trabajo en la gestión del tránsito aéreo de la organización europea para la seguridad y la navegación aérea

(2006). Esta publicación además menciona que los horarios de los turnos deberían distribuir la carga laboral equitativamente.

3. Fatiga en los controladores.

Según Bruno Pedraz-Petrozzi (2018):

“En el ámbito biomédico, la fatiga es para algunos investigadores concebida como algo que ocurre a lo largo de diversas actividades realizadas por el ser humano. El estrés prolongado suele ser el efecto detonante y genera así la fatiga.”

Los controladores de tránsito aéreo son susceptibles a la fatiga, ya que se requiere mantener un desempeño alto durante un período de turno. Como normalmente trabajan en turnos rotativos, habrá momentos dentro del turno, por ejemplo, en las noches, en las que se requiera una alerta alta cuando el horario nocturno los haga sentir somnolientos. (skybrary, 2015)

En el artículo la fatiga laboral en los controladores de tránsito aéreo (Escalona, Urosa, et al., 1996) se determinaron las posibles causas generadoras de fatiga y las consecuencias que éstas traen sobre la salud física y mental de los cta.

Una de ellas es la afectación que el ruido tiene sobre las áreas de trabajo debido a la cercanía con los hangares de los aviones. Esto puede dificultar la comunicación entre controladores de una misma sala aumentando así el esfuerzo y desgaste para realizar correctamente sus tareas. Según (AEMPPI Ecuador, 2018): “La exposición al ruido puede aumentar el riesgo de padecer HTA, angina de pecho o un infarto agudo de miocardio. Esto se debe a una activación de hormonas nerviosas, que va a provocar el aumento de la tensión arterial o la vasoconstricción, entre otras.”

Igualmente se menciona que se debe identificar las horas en que ocurre mayor ruido y rotar a los trabajadores para que tomen su tiempo de descanso.



En el mismo estudio resaltan que los controladores con más antigüedad en la empresa presentan problemas de salud como gastritis, insomnio, hipertensión arterial y afectaciones a nivel nervioso.

En 2012 Marco Antonio Leyva Piña (Leyva, 2012) publica un ensayo en el que muestra el deterioro de la Aviación en México mostrando la perspectiva de los controladores. Allí menciona que los CTA presentan inconformismo relacionados con cambios de horario especialmente a aquellos que generan estrés laboral, lo que impacta negativamente la seguridad operacional y el bienestar laboral.

“Por ejemplo, el incremento de la carga de trabajo, contribuye a la exigencia de emplear más procedimientos que llegan a consumir más tiempo y simultáneamente puede producirse una reducción progresiva de información de vuelo, ambos procesos impactan negativamente en la calidad de la seguridad y en las expectativas de eficiencia, pues se incrementa el número de decisiones que deben ser tomadas en rangos de tiempo reducidos, lo que puede determinar sobrecargas de trabajo, no entendidas a partir de excedentes de horarios sino de intensidad del trabajo que llega a generar altos riesgos laborales como el denominado pérdida de la conciencia situacional.” (Leyva, 2012, p.42)

Por último, se han encontrado trabajos que buscan repartir la carga laboral entre los controladores uniformemente por medio de la planilla de turnos. En este caso, se usó de la metaheurística para solucionar un problema de timetabling. Aunque los resultados que obtuvieron al optimizar la función objeto son mejores que los que se obtienen de la manera regular, esta mejora fue leve. Además, proponen comparar esos resultados con los de otras técnicas. (Suarez, 2017).

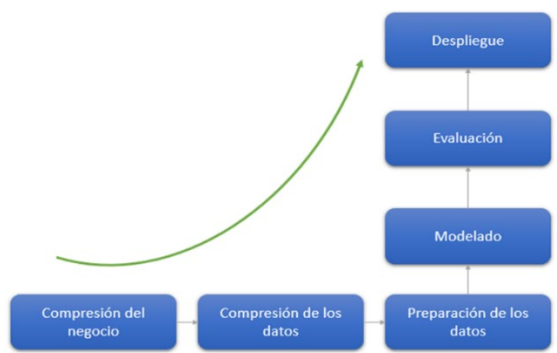
4. Metodología CRISP-DM

El desarrollo de este proyecto se realizará mediante la metodología CRISP-DM

Minería de datos

Es el proceso mediante el cual recolectamos, limpiamos y hallamos anomalías, patrones y correlaciones de los datos para comenzar a trabajar la data. Para este proceso se realizará la aplicación de la metodología CRISP-DM

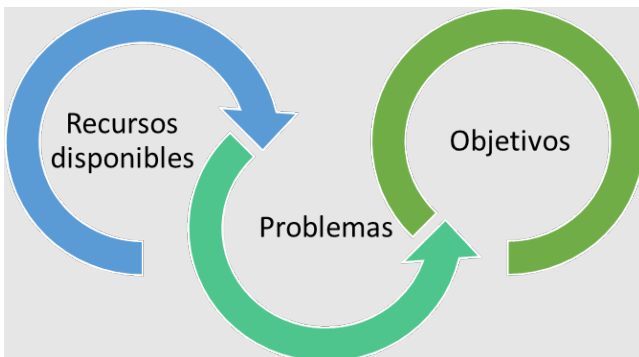
Ilustración 1. Metodología CRISP-DM



Fuente: Elaboración propia a partir Minguillón, J. Casas, J. y Minguillón, J. (2017)

Compresión del negocio y compresión de los datos

Ilustración 2. Comprensión del negocio



Fuente: elaboración propia



Para realizar el proceso se comenzará con la comprensión del negocio en esta tarea se evalúan los objetivos, componentes, requerimientos y limitaciones y que estos contribuyan a la ejecución de los objetivos trazados; La comprensión de datos se refiere a trabajar la materia prima los datos, su estructura la forma en que está presentada sus variables, para esta etapa debemos realizar una tarea exploratoria de los datos, descripción del juego de datos y generar un informe de la calidad de los datos, con el fin de identificar problemas y proporcionar soluciones tempranas.

Para esto también se definimos si se usan los datos existentes, si necesitamos adquirir datos externos demográficos o específicos o si por el contrario se requieren datos adicionales como seguimientos o encuestas. De igual forma se realiza la descripción de los datos es decir la cantidad de datos, tipos de valores y los esquemas de codificación.

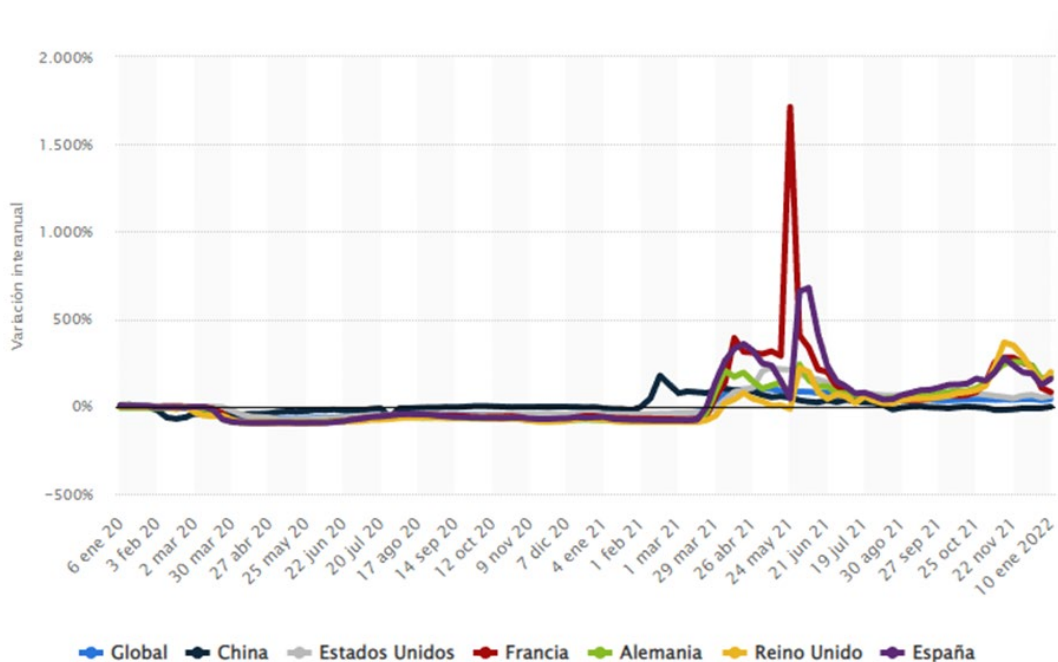
Tratamiento de datos

Para el tratamiento de datos y lograr nuestro objetivo se requiere construir una base de datos, con la cual nos aporte la misma cantidad de información que el controlador use para tomar la decisión; de igual forma se tendrán datos históricos desde el 2.017 hasta el 2.021 ya que el covid afecto de manera drástica la navegabilidad aérea en el 2.020.

Primer problema

Según un estadístico por la página statista, muestra la afectación, y presenta la variación interanual de los vuelos programados por las aerolíneas en el mundo del 6 de enero de 2020 al 10 de enero de 2022, por país de origen. El impacto del coronavirus (SARS-CoV-2) se pudo ver en todos los países más afectados, así como a nivel mundial.

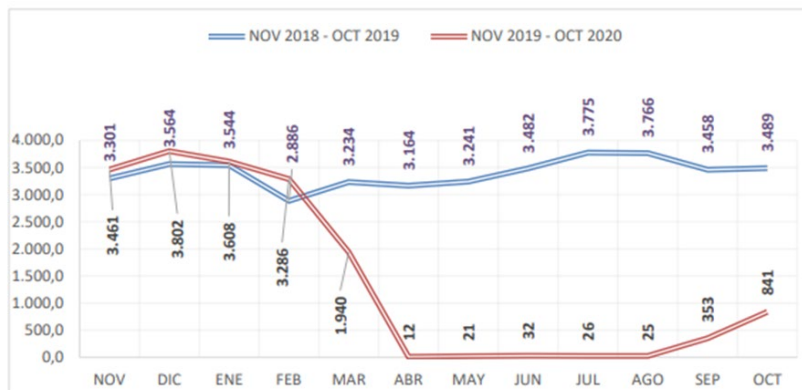
Ilustración 3. Variación interanual del número de vuelos programados a la semana del 13 de enero de 2020 al 10 de enero de 2022, por país de origen



Fuente: Gráfico lineal de vuelos programados extraído de, STATISTA, (enero 2022)

Colombia no ha sido la excepción en afectaciones por restricciones aéreas, donde se presenta el siguiente estudio, reflejan el fuerte impacto generado en Colombia por la Emergencia Sanitaria Internacional originada entre otros, por el COVID-19, debido a la Suspensión del desembarque con fines de ingreso o conexión en territorio colombiano, de pasajeros procedentes del exterior, por vía aérea a partir del 23 de marzo 2020, (Decreto 439 del 20 de marzo de 2020), restricción que fue levantada mediante Resolución No 1769 del 15 de septiembre de 2020.

Ilustración 4. Pasajeros movilizados noviembre 2019-octubre 2020 Vs noviembre



Fuente: Gráfico lineal extraído de documento AERONÁUTICA CIVIL, Roberto Held, Claudia Esguerra B, Jorge Quintana C, Luz Melba Castañeda L, (s.f.)

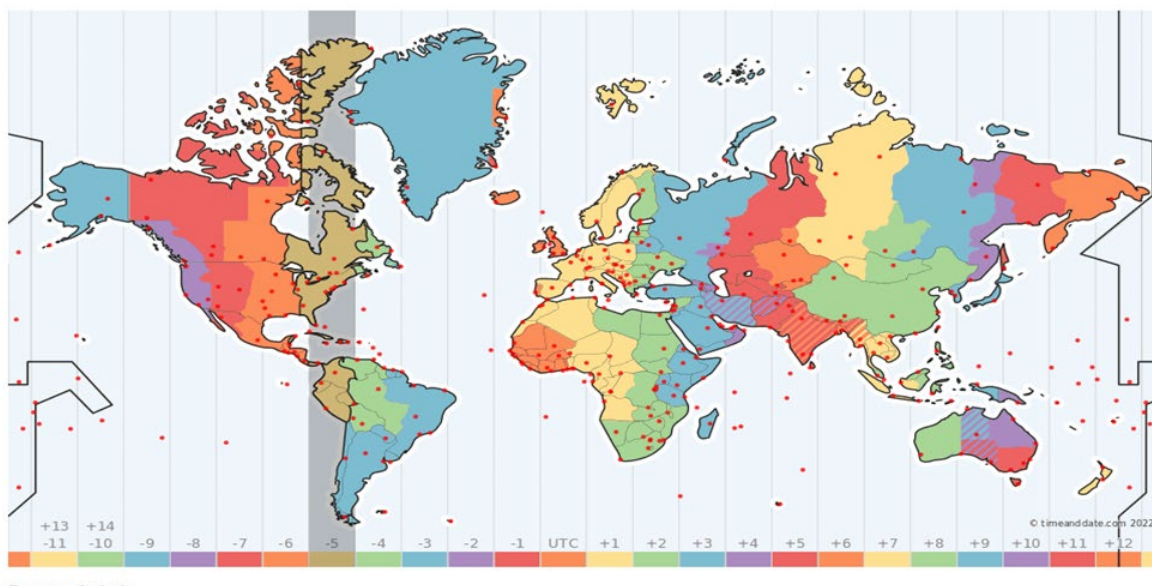
Es importante anotar que con anterioridad el Gobierno Nacional a través del Ministerio de Transporte estableció que a partir del 16 de marzo de 2020 ningún viajero extranjero, proveniente de cualquier lugar del mundo podría ingresar al territorio colombiano. Esta medida se adoptó primero, en marzo 13, para los viajeros que hubieran estado en aquellos países donde se presentaban altos índices de contagio y con Decreto 457 del 22 de marzo, se suspendió el transporte doméstico por vía aérea a partir del día 25 de marzo de 2020, permitiendo el transporte doméstico por vía aérea, solamente en los siguientes casos: 1. Emergencia humanitaria. 2. El transporte de carga y mercancía. 3. Caso fortuito o fuerza mayor. Roberto, H., Esguerra B, C., Quintana C, J., & Castañeda L, L. M. (2019)

Y que solamente hasta el 21 de julio de 2020 se reinició la operación de transporte aéreo a través de rutas piloto para vuelos domésticos en los aeropuertos de Cúcuta y Bucaramanga. Esto sin duda es una limitante y puede que nos genere sesgos dentro de nuestros resultados pues habría que evaluar si se tienen en cuenta los datos históricos de 2.020.

Segundo problema

Otro tratamiento que se debe aplicar es que se tiene que modificar la acorde a la UTC (Universal Time Coordinated), tiempo universal coordinado, que es el estándar de tiempo por el cual se regula los relojes del mundo. Distribuye al planeta en bandas longitudinales de 15 grados de ancho y divide al mundo en 24 zonas horarias. Para Colombia se debe restar 5 horas en las registradas en la data. Esto debe hacerse con sumo cuidado pues puede guiarnos a resultados inesperados por el tratamiento de la información.

Ilustración 5. Mapa de zona horaria



Fuente: Mapa extraído de TimeAndDate (2022)

Proceso ETL

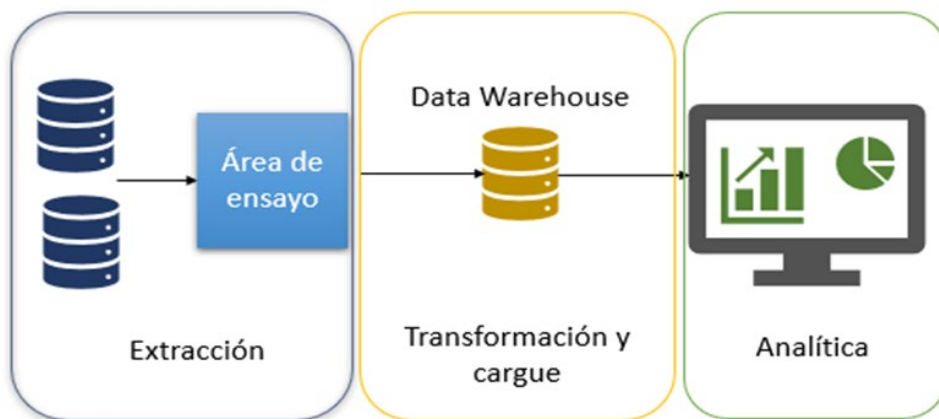
Herramientas de minería de datos

Las herramientas de minería de datos permiten extraer relaciones, patrones, tendencias y regularidades para describir y comprender mejor los datos y para predecir comportamientos futuros. Para lograr el objetivo se realizará el proceso de ETL a la información recolectada.

ETL

Existe un sistema especializado para realizar la carga y mantenimiento de un almacén de datos, denominado sistema o proceso ETL (Extract, Transform, Load), extraer, transformar y cargar. Este sistema se encarga de la lectura de datos transaccionales, de la incorporación de datos externos, creación de claves, integración de datos, agregaciones, limpieza y mantenimiento, entre otros.

Ilustración 6. Proceso ETL



Fuente Elaboración propia

Extracción: Proceso de lectura de datos de diversas fuentes tales como:

- Bases de datos de aplicaciones
- CRM
- Archivos CSV
- Dataset públicos

Transformación: Proceso de limpieza de datos para su posterior inserción en un data warehouse

- Limpieza
- Estructura
- Enriquecimiento
-

Data Warehouse: Es un conjunto de datos históricos que se presentan de una o más fuentes, su estructura ordenada permite resumir, describir y analizar los datos.

Load o cargue: Es el proceso de inserción de datos a la data warehouse

Selección De Técnicas De Modelado

En este proceso se revisan las técnicas de predicción acorde a las características de los datos, las necesidades de los objetivos planteados, que cumpla con los requisitos del problema y que la técnica a usar sea la correcta para que nos muestre los resultados correctos.

Series de tiempo

Acorde con Nava (2013). las series de tiempo es un conjunto ordenado de valores en los cuales se realiza predicciones de valores futuros, utilizando la información contenida en los valores pasados de la serie temporal, que mide la evolución de las variables de objeto de estudio. En el cual tenemos T predicciones $(x_1, x_2 \dots x_t)$

En el cual predécimos valores futuros $(x_{t+1}, x_{t+2} \dots x_{t+h})$

Comportamiento estadístico X fin de la distribución conjunta de datos

$$D_X = (X_1, X_2, \dots, X_T, X_{T+1}, X_{T+2}, \dots, X_{T+H})$$

Distribución de valores futuros condicional al pasado

$$D X_2 | X_1 | (X_{T+1}, X_{T+2}, \dots, X_{T+H} | X_1, X_2, \dots, X_T |)$$

Igualando las dos últimas formulas se tiene que

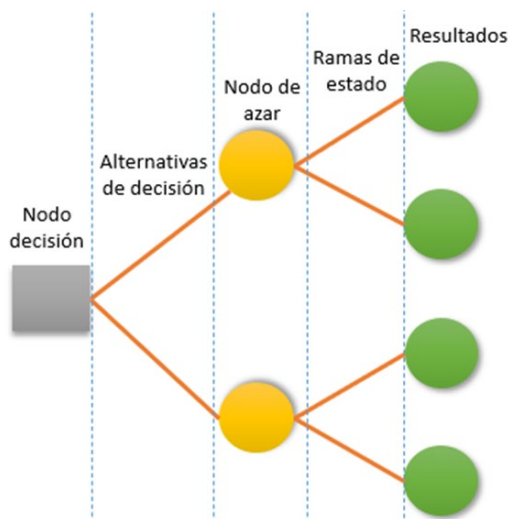
X_1 y D_1

conocido podemos calcular $D X_2 | X_1 |$ y la probabilidad asociada a valores futuros de X e intervalos de predicción

Árboles de decisión

Es un modelo usado en la minería de datos que subdivide los espacios de entrada y generar regiones a elementos que pertenezcan a la misma clase, su estructura consta de nodos y terminales, cada nodo designa una selección de opciones de resultados posibles o inciertos, este debe ser realizado en orden cronológico de izquierda a derecha y en los resultados se realiza una calificación cuantitativa que permite tomar la decisión con el puntaje más alto y podar las ramas en las cuales se tenga una menor denotación; Pero como sabemos que resultados es el más favorable para nuestro ejercicio para esto realizamos el análisis mediante la teoría de Bayes.

Ilustración 7. Árbol de decisión



Fuente: elaboración propia

Teorema de Bayes

Es una técnica que expresa la probabilidad condicional de un evento, donde se agrega otro suceso. Esto permite tener la probabilidad condicional de un evento determinado como A dado B. Para esto debemos saber:

Sea A_1, A_2, \dots, A_n

Un conjunto de datos excluyentes, tales que la probabilidad de ellos sea diferente de cero.

$$P[A_i] \neq 0 \text{ para } i = 0, 1 \dots n$$

Si B es un suceso cualquiera donde se conocen las probabilidades condicionales, se tiene:

$$P\left(\frac{A_i}{B}\right) = \frac{P\left(\frac{B}{A_i} \cdot P(A_i)\right)}{P(B)}$$

Donde:

son las probabilidades a priori

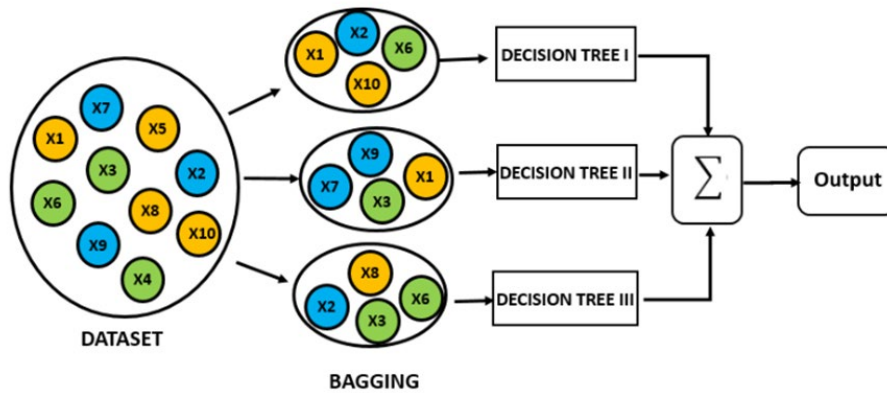
es la probabilidad de B en la hipótesis de

son las probabilidades a posteriori.

Random Forest

De acuerdo Espinosa-Zúñiga, J. J. (2020) El algoritmo Random Forest es una técnica de aprendizaje supervisado que genera múltiples árboles de decisión sobre un conjunto de datos de entrenamiento: los resultados obtenidos se combinan a fin de obtener un modelo único más robusto en comparación con los resultados de cada árbol por separado.

Ilustración 8. Random forest



Fuente: Imagen del proceso extraída de la documentación de Orellana J, (2018) Arboles de decisión y random forest

Se crean múltiples subconjuntos de datos, con múltiples modelos. Donde se hace robusta los datos a analizar con el fin tener un mayor número de posibilidades, y se abarcan todas las posibles salidas que toma la predicción y que tan relevantes son para el estudio.

Regresión lineal

De un modo general se dice que existe regresión de los valores de una variable con respecto a los de la otra cuando hay alguna línea, denominada línea de regresión, que se ajusta más o menos claramente a los valores observados. La regresión se usa para la identificación de relaciones potencialmente causales o bien, cuando no existen dudas sobre su relación causal, para predecir una variable a partir de la otra. Cuando dos variables tienen una relación de tipo determinista, el valor de una define exactamente el valor de la otra; un ejemplo puede ser la relación entre la presión y el volumen de un gas a temperatura constante. En los fenómenos biológicos, sin embargo, la relación entre las variables es de tipo aleatorio ya que existen múltiples factores, muchos de ellos desconocidos, que influyen sobre la relación que existe entre las dos variables. En el primer caso, al hacer las mediciones en una relación determinista, los resultados caerán muy cerca de la línea de regresión; la cercanía o coincidencia de los puntos observados con esa

línea estará definida por el error de la medición. En el segundo caso, a esta variabilidad se debe agregar aquella introducida por esos múltiples factores adicionales.

El procedimiento para seguir puede dividirse en cuatro etapas:

- La primera aproximación es a través de dibujar los puntos en un gráfico cartesiano que muestre la relación entre las dos variables.
- Luego se determina la ecuación de la línea que mejor describa dichos puntos.
- A continuación, se calcula la variabilidad de la muestra en torno a la línea de regresión calculada.
- Finalmente se pueden hacer inferencias.

Universidad Católica de Chile (2014). Regresión lineal

5. Lenguajes de programación para tratamiento de predicción de datos

Entre los cientos de lenguajes de programación con los que podemos aplicar la solución, debe tomar en cuenta los siguientes factores; para aumentar o mejorar la optimización del proceso, se prioriza un lenguaje interpretado, el cual, consume menos tiempo y recursos en su ejecución en tiempo real; con respecto al paradigma del lenguaje, con este nos permite ayudar a organizar los modelos (en caso de que lo permita) para segmentar los métodos y/o agrupaciones sería útil el paradigma Orientado a Objetos y/o funcional, además de un tipado fuertemente para disminuir los errores de los tipos de las variables al momento del desarrollo y finalmente no es necesario que su objetivo sea enfocado a la IA, si es general será mejor ya que tendrá más soporte y comunidad de la cual se usarán recursos de uso libre.

5.1.Lenguajes elegibles

Tabla 1 Características de lenguajes de programación

Lenguaje	Interpretado	Compilado	Paradigmas	Tipado	Objetivo	Plataformas
Python	Sí	No	<ul style="list-style-type: none"> • Orientado a objetos • Orientado a procedimientos • Funcional 	Fuertemente	General	Windows Mac Linux
Ruby	Sí	No	Orientado a objetos	Fuertemente	General	
R	Sí	No	Funcional y Orientado a objetos	Dinámico	Análisis de datos	
C++	No	Sí	Orientado a objetos	Débilmente	Scripts	
C#	No	Sí	Orientado a objetos	Fuertemente	Aplicaciones	
Prolog	Semi-interpretado		Lógico declarativo	No	Inteligencia artificial	
Java	Sí	Sí	Orientado a objetos	Fuertemente	General	
JS (Node JS)	Sí	No	Orientado a objetos	Débilmente	Scripts	

Fuente Elaboración propia

5.2. Lenguaje elegido

Python

Este lenguaje es uno de los más conocidos por su versatilidad para realizar múltiples tareas; páginas web, bots, web-scraping y muchos más propósitos entre ellos el *machine learning*, aunque entre los lenguajes elegibles está uno especial para el desarrollo de inteligencia artificial, se opta por la optimización de tiempo, ¿es más rápido desarrollar una inteligencia artificial en Python que en Prolog?, la respuesta es relativa y nos lleva a otra pregunta ¿qué tanto sabes de cada lenguaje?, la ventaja de Python para cumplir el objetivo planteado, y como se menciona al inicio de este párrafo, es uno de los más conocidos, esto ocasiona que el soporte y los recursos (librerías y/o documentación) se extienden cada vez más, además de su “similitud” a los demás lenguajes tipados con unas variaciones de sintaxis, así que, su aprendizaje es más rápido, mayor documentación y foros de otros usuarios que posiblemente han intentado y/o implementado lo mismo que tú, y la realización de métodos o ya existen o existen recursos que ya lo hacen, con esto se reduce el esfuerzo y optimiza el tiempo. Prolog sería la mejor opción si tienes el tiempo necesario para dedicarle a su aprendizaje y el tiempo experimental para cada vez mejorar y entender las mejores maneras de trabajar con este.



Marco institucional

Unidad Administrativa Especial de Aeronáutica Civil

Es un organismo gubernamental (rama ejecutiva del poder público) vinculado al Ministerio de Transporte. Su sede central se encuentra localizada cerca al Aeropuerto Internacional el Dorado en el edificio NEAA (Nuevo Edificio Administrativo de Aerocivil).

La Aerocivil es la autoridad aeronáutica en la República de Colombia, ocupándose de la reglamentación, vigilancia y control de la aviación civil, además de la prestación de los servicios de navegación aérea y administración del espacio aéreo bajo la jurisdicción del Estado Colombiano; también opera el Centro de Estudios de Ciencias Aeronáuticas, CEA, entidad de educación superior, técnica e intermedia con programas de formación y entrenamiento en especialidades del campo aeronáutico.

La Unidad Administrativa Especial de Aeronáutica Civil rige todas las actividades de la aviación civil, las cuales quedan sometidas a la inspección, vigilancia y reglamentación del Gobierno. El Gobierno por medio de la Aeronáutica Civil podrá prohibir, condicionar o restringir, por razones de interés público la utilización de los espacios, la navegación aérea sobre determinadas regiones, el uso de ciertas aeronaves o el transporte de determinadas cosas.

De igual forma se regula lo concerniente a la navegación aérea, aeronaves, personal aeronáutico, infraestructura aeronáutica, daños o terceros en la superficie, abordaje, búsqueda, rescate, asistencia y salvamento, investigación de accidentes de aviación, explotador de aeronaves, transporte privado, escuelas de aviación, aeronaves dedicadas al turismo y mantenimiento de aeronaves, transporte aéreo, de pasajeros, de carga y equipajes y demás normas aplicables.



Misión

Trabajamos por el crecimiento ordenado de la aviación civil, la utilización segura del espacio aéreo colombiano, la infraestructura ambientalmente sostenible, la conexión de las regiones entre sí y con el mundo, impulsando la competitividad y la industria aérea y la formación de un talento humano de excelencia para el sector.

Visión

Al 2030, movilizar 100 millones de pasajeros y duplicar el transporte de carga partiendo del 2018, en un entorno institucional claro, competitivo, conectado, seguro y sostenible, soportado en una infraestructura renovada, una industria robustecida y un talento humano de excelencia

ANÁLISIS DE REQUERIMIENTOS

Se decide utilizar una metodología de desarrollo ágil, en este caso SCRUM.

En donde se realiza la construcción del backlog con las historias de usuario y su respectiva puntuación con la estimación de secuencia Fibonacci.

Tabla 2 Historia de usuario #1

Nombre	Selección de lenguaje de programación	Número	Puntuación
		1	2
Descripción	Investigar casos de uso de los lenguajes de programación más usados en la IA y sus derivaciones, para tener una idea general del lenguaje más popular para un machine learning; con base en esa información compararla con nuestra finalidad y curva de aprendizaje de cada lenguaje para elegir uno óptimo pero que además sea "fácil" de implementar posibles recursos externos que sean necesarios para todo el desarrollo.		
Criterios de aceptación	<ul style="list-style-type: none"> * El lenguaje debería ser interpretado (mejora su poco su velocidad de ejecución) * Lenguaje orientado a Objetos y/o funcional * Lenguaje tipado 		

Fuente Elaboración propia

Tabla 3 Historia de usuario #2

Nombre	Obtener información real del aeropuerto de Cúcuta	Número	Puntuación
		2	4
Descripción	Solicitar información al aeropuerto internacional de Cúcuta, de los vuelos de salida y llegada.		

Criterios de aceptación	<ul style="list-style-type: none"> * Mínimo 2 años de información * Información de años previamente consecutivos * Debe contener fecha y hora de llegada, como fecha y hora de salida
-------------------------	--

Fuente Elaboración propia

Tabla 4 Historia de usuario #3

Nombre	Acercamiento al lenguaje de programación elegido	Número	Puntuación
		3	4
Descripción	Hacer pruebas de uso al lenguaje y/o entorno elegido, como también leer su documentación y en caso de no conocerlo, dedicarle tiempo para la curva de aprendizaje necesario para la aplicación que se le va a dar.		
Criterios de aceptación	<ul style="list-style-type: none"> * Entender la manera de ejecutar funciones básicas dentro del lenguaje * Conocer el método de instalación y adición de recursos openSource al lenguaje 		

Fuente Elaboración propia

Tabla 5 Historia de usuario #4

Nombre	Minería de datos	Número	Puntuación
		4	6
Descripción	Hallar posibles anomalías, patrones y correlaciones de datos; basándose en una metodología de minería de datos.		

Fuente Elaboración propia

Tabla 6 Historia de usuario #5

Nombre	Tratamiento de datos	Número	Puntuación
		5	5
Descripción	Se debe crear o integrar una base de datos con la que se pueda interactuar desde el aplicativo principal, para insertar información y mantenerla, aunque la aplicación no esté en ejecución.		
Criterios de aceptación	<ul style="list-style-type: none"> * La base de datos debe ser relacional * Debe ser accesible únicamente por el aplicativo * Puede ser base local o en línea * Debe mantener los datos aun cuando el aplicativo principal no esté en ejecución 		

Fuente Elaboración propia

Tabla 7 Historia de usuario #6

Nombre	ETL	Número	Puntuación
		6	8
Descripción	Nativamente no todos los lenguajes de programación tienen un método para leer archivos locales; para esto se debe integrar una librería open Source para la lectura de archivos y el tratamiento de estos para su inserción a la base de datos.		
Criterios de aceptación	<ul style="list-style-type: none"> * Debe permitir leer archivo xls, xlsx, csv y txt * Debe ser posible que permita programación reactiva, para ir limpiando/formateando los datos mientras se extraen * Si el archivo a leer no se puede abrir o genera error, debe pausar el proceso y solicitarle al usuario intentar de nuevo o elegir otro archivo. 		

Fuente Elaboración propia

Tabla 8 Historia de usuario #7

Nombre	Árbol de decisiones	Número	Puntuación
		7	8
Descripción	Evaluar y establecer una serie de condiciones que se deben desencadenar en un árbol de decisiones para posteriormente realizar un análisis de probabilidad condicional y determinar a través de algún método conocido los valores favorables.		

Fuente Elaboración propia

Tabla 9 Historia de usuario #8

Nombre	Visualización de resultados	Número	Puntuación
		8	6
Descripción	El usuario debe tener una manera gráfica para visualizar el resultado de las predicciones, nativamente los lenguajes de programación permiten generar una consola donde el usuario interactúa, pero las tablas de información compleja pueden distorsionarse para su entendimiento; por esto integrar alguna librería para graficar el resultado.		
Criterios de aceptación	* Debe permitir distinguir puntos máximos y mínimos preferiblemente en un gráfico de línea. * En caso contrario permitir exportar la información en Excel para usar su graficador.		

Fuente Elaboración propia



Análisis De Restricciones

Restricción de recursos:

Información: Para el análisis de la data que debe realizar el machine learning y pueda generar los datos de predicciones, y ésta se requiere con un alto grado de precisión, así mismo se necesita una cantidad considerable de datos de años pasados de llegadas y salidas al aeropuerto Camilo Daza. La restricción está en que aún es incierto la cantidad de datos de años pasados que nos pueda proporcionar para el proyecto. Actualmente tenemos un par de años del pasado, 2018 y 2019 para el análisis de la cantidad de vuelos, y para lograr una precisión de predicción razonable de 4 años, necesitaríamos 4 años de datos.

Restricción de tiempo:

El retraso en la entrega de información nos retrasa en su análisis, planeación y ejecución para la construcción de machine-learning, además las pruebas de calidad de las predicciones afectando la precisión de estas.

Restricción de contratiempos con librerías:

Para desarrollar el machine-learning en el lenguaje de Python, no es suficiente con su base, debemos añadir una o varias librerías open source diseñadas para Python especialmente con enfoque a IA o sus variaciones. Debemos basarnos en la documentación y soporte de estas, así como sus restricciones de lógica a la cual debemos enfrentarnos debido a la escasa experiencia con estas.

Metodología Para La Selección Y Desarrollo De La Solución

Dentro de la metodología CRISP-DM se realizan los siguientes pasos:

Comprensión del negocio

En esta tarea se recopila información de la empresa situación actual, así como registrar los objetivos, el alcance del proyecto y las herramientas disponibles para el desarrollo de modelo

- Se recopila la información para el diseño del modelo, en este caso el registro histórico de vuelos en el aeropuerto Camilo Daza, desde el año 2.017 a 2.019.
- Comprensión del negocio y de las variables a ser utilizadas, no se tiene una herramienta eficiente para el diseño de los horarios de los controladores aéreos
- Con base a los objetivos se diseña un modelo que permita tener un grado de exactitud y precisión óptimos

Comprensión de los datos

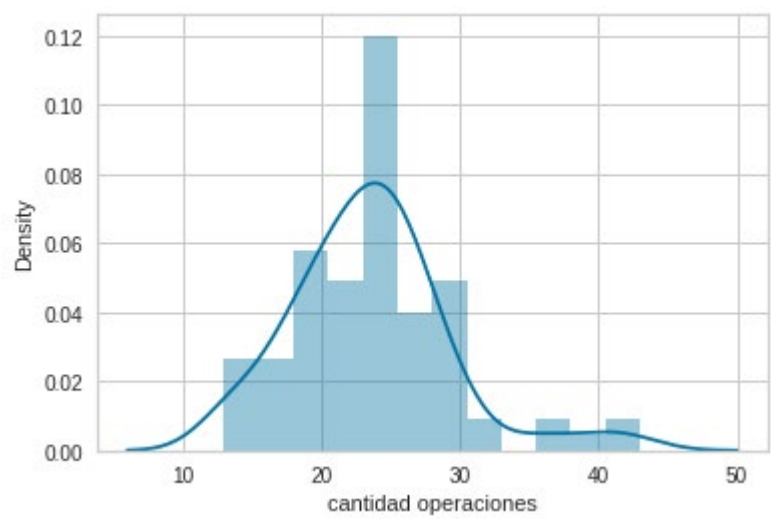
En esta tarea se realiza la inmersión de los datos seleccionando las variables relevantes para el estudio, así como validar si se requieren datos adicionales. Para nuestro estudio se tomó el registro histórico de 2.017 a 2.019, en el cual contiene registros de tipo numérico. Debido a las afectaciones sufridas por la pandemia no se tendrá en cuenta el año 2.020 ya que puede sesgar nuestro objeto de estudio.

Preparación de los datos

Para el proceso de preparación de datos debemos realizar criterios de selección, con el fin de tener los resultados esperados acorde a los objetivos trazados, realizar tareas de exploración de datos, secuencias y posibles sesgos en la recolección de la información; para la preparación de datos se debe contar con el modelo en el cual se van a aplicar los datos, la calidad de los datos y las limitaciones. Para esto realizamos los siguientes pasos:

- Al tener el alcance de las necesidades y definición de nuestro problema, se establecen los objetivos de la minería de datos.
- Se determinan las fuentes de los datos y las variables a usar y que sean relevantes para el cálculo.
- Se realiza la limpieza de los datos, para este se valida errores humanos espacios en blanco, errores al leer. Y que tenga la estructura adecuada acorde a nuestro estudio.
- Se realiza el tratamiento de datos acorde a los conocimientos adquiridos del negocio, objetivos, características etc.
- Se realiza la corrección de las horas acorde al modelo de horas UTC, se realiza la separación de fecha y hora para establecer los mapas de calor.
- Se realizan filtros por mes, con el fin de comprender y proyectar el comportamiento de cada mes.

Ilustración 14. Predicción de operaciones mes 6



Fuente Elaboración propia

Ilustración 15. Predicción de operaciones aéreas por random forest junio 2022

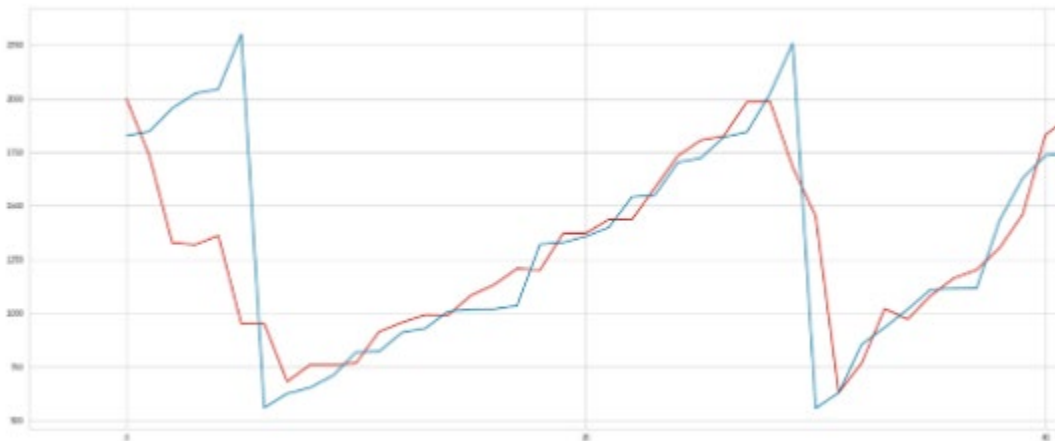
Fuente Elaboración propia

Ilustración 16. Grafica de la predicción



Fuente Elaboración propia

Ilustración 17. Predicción de operaciones por semana



Fuente Elaboración propia

Selección de alternativas

Para la selección de la mejor alternativa decidimos realizar una tabla que permitió comparar y analizar las opciones que se tenían llegando a tres secciones principales; la primera es el lenguaje de programación, la segunda el ambiente de desarrollo que se va a utilizar y la tercera es el algoritmo que se pretende implementar. Cabe señalar que se hizo uso de colores para mayor comprensión, donde las secciones están representadas de color amarillo y las opciones se

representan de color verde o rojo, teniendo en cuenta que el color verde significa “alternativa seleccionada” y el color rojo se usó para reconocer la “alternativa que menos se tomará en cuenta”.

Tabla 10 Selección de alternativas

Lenguaje de programación	Python	Muy conocido por Machine learning, su curva de aprendizaje es rápida, tiene una variedad de librerías y buen soporte
	Prolog	Tiene un enfoque específico hacia la inteligencia artificial, pero se requiere de más tiempo para aprenderlo, su comunidad no es tan grande por lo que cuenta con menos soporte para solucionar problemas
Ambiente de desarrollo	Google Colab	Permite un trabajo colaborativo ya que sus archivos son almacenados en el Drive de Google. El código es ejecutado en los servidores de Google Por lo que no es necesario tener un equipo personal potente.
	Pycharm	Puede presentarse a limitaciones si el proyecto necesitas un GPU y TPU exigente. Tiene muchas funcionalidades que facilitan la tarea al programador.
Algoritmo por utilizar	Naive Bayes	Es fácil de implementar sin embargo solo se puede usar para trabajos de clasificación. Por tanto, no se ve muy útil ya que el resultado es no binario
	Regresión lineal	Estos modelos de regresión lineal no demandan altas especificaciones de hardware para generar predicciones. Adicionalmente, pueden ser implementados de una manera práctica y sencilla por medio del lenguaje de programación de Python.

Fuente Elaboración propia



En conclusión, se seleccionó Python como lenguaje de programación haciendo uso de la herramienta Google colab puesto que la misma permite un trabajo más colaborativo y con mayor armonía sin tener limitaciones de hardware. Por último, el primer algoritmo de prueba será Regresión Lineal.



ANÁLISIS DE COSTOS

Adquisición y mantenimiento de hardware y/o software

Debido a la naturaleza del producto, el cual es un software, y al uso de OpenSource, no hubo costo de adquisición de algún servicio u otro software complementario de pago.

\$0

Gastos de comunicaciones

Los gastos en la comunicación del equipo para realizar los avances y alineamientos de realización se gestionaban por videollamadas con un servicio inherente al inicio del desarrollo de este y algún otro proyecto, el servicio de internet. Por lo cual, no es una adquisición exclusiva para el proyecto y no genera un gasto adicional.

\$0

Gastos de instalación

En este apartado se toman en cuenta el acondicionamiento de algún lugar, cableado, materiales, viajes en caso de ser necesarios para poder crear un producto; sin embargo, no fueron necesarios para el procedimiento de “fabricación” de este software. Costos diferentes a los que deberá asumir el cliente si desea poner en ejecución este software, gastos que serán asumidos por cada cliente.

\$0



Costo de desarrollo

Es la similitud de la mano de obra; para el desarrollo de este software se dedicó horas de trabajo con conocimiento basado. Generando un aproximado de 2 meses de trabajo en días hábiles, podemos generando un estimado de un salario mensual para un desarrollador middle Python de 5 millones de pesos colombianos.

\$ 10.000.000

Gastos de consultoría

Para el proceso de desarrollo, aunque se consideró que se tenía el conocimiento requerido, aun así, para generar un producto de calidad se consultó con un experto en el desarrollo y realización de algoritmos enfocados en machine learning. Durante la etapa de inicio del desarrollo se realizó la consulta para trazar un roadmap. La consulta que fue en varias sesiones durante 1 semana

\$ 1.000.000

Gastos de material

Los materiales de documentación utilizados durante la profundización de información en machine learning y Python fueron recursos públicos en la web.

\$0



SOLUCIÓN DE INGENIERÍA

El proyecto de investigación tiene como fin el diseño de un prototipo que permita predecir las operaciones aéreas del aeropuerto Camilo Daza con el fin de optimizar los turnos de los controladores aéreos del mismo, tomando como base los datos contenidos en el Anexo 1.

El análisis fue realizado mediante Machine Learning en lenguaje Python.

A continuación, se van a presentar partes importantes del código con su respectiva explicación de lo que se quiere lograr en cada sección y una breve descripción de cómo se hace, el código se adjuntará y podrá ser abierto en cualquier computador con Google colab.

Código fuente

Anexo el enlace para la visualización del código:

<https://colab.research.google.com/drive/1-bNeP5LcFO66LCHQK5jIw1hW6L8ReRzl?usp=sharing>

El siguiente link permite descargar los archivos de Excel necesarios para correr el código:

<https://drive.google.com/drive/folders/16vOAR7RYqnYGCnhbP7KIK9bAA2RMn86C?usp=sharing>

Librerías que se importan

Las librerías que se importan para el análisis en Python como lo son statsmodels la cual incluye funciones de modelado estadístico, pandas la cual se especializa en el manejo y análisis de datos estructurado.

Ilustración 18 Importación librerías

```
import re
import warnings
warnings.filterwarnings("ignore")

import numpy as np           # vectors and matrices
import pandas as pd         # tables and data manipulations
import matplotlib.pyplot as plt # plots
import seaborn as sns       # more plots

from dateutil.relativedelta import relativedelta # working with dates with style
from scipy.optimize import minimize             # for function minimization

import statsmodels.formula.api as smf          # statistics and econometrics
import statsmodels.tsa.api as smt
import statsmodels.api as sm
import scipy.stats as scs

from itertools import product                 # some useful functions
from tqdm import tqdm_notebook

from sklearn.metrics import r2_score, median_absolute_error, mean_absolute_error
from sklearn.metrics import median_absolute_error, mean_squared_error, mean_squared_log_error

def mean_absolute_percentage_error(y_true, y_pred):
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

%matplotlib inline
```

Fuente Elaboración propia

Importación de Datos

En esta parte cargamos los archivos XLSX, los cuales fueron descargados de una base de datos y entregados por parte de la Aeronáutica Civil. Para cargar los archivos utilizamos la librería de pandas.

Ilustración 19 Ejecución manual: Carga de archivos

```
[0] from google.colab import files
    uploaded = files.upload()

Elegir archivos 3 archivos
• registrovuelos2017.xlsx(application/vnd.openxmlformats-officedocument.spreadsheetml.sheet) - 1082303 bytes, last modified: 4/5/2022 - 100% done
• registrovuelos2018.xlsx(application/vnd.openxmlformats-officedocument.spreadsheetml.sheet) - 1096110 bytes, last modified: 4/5/2022 - 100% done
• registrovuelos2019.xlsx(application/vnd.openxmlformats-officedocument.spreadsheetml.sheet) - 1163661 bytes, last modified: 4/5/2022 - 100% done
Saving registrovuelos2017.xlsx to registrovuelos2017.xlsx
Saving registrovuelos2018.xlsx to registrovuelos2018.xlsx
Saving registrovuelos2019.xlsx to registrovuelos2019.xlsx

import pandas as pd
import io

df = pd.read_excel(io.BytesIO(uploaded['registrovuelos2017.xlsx']))
print(df)

df = pd.read_excel(io.BytesIO(uploaded['registrovuelos2018.xlsx']))
print(df)

df = pd.read_excel(io.BytesIO(uploaded['registrovuelos2019.xlsx']))
print(df)
```

	Fecha	ID	SSR	Callsign	Matrícula	Tip	Aer	Empresa	# Vuelo	\
0	10117	84613	1305.0	AVA9451	N590EL	A318	AVA	9451.0		
1	10117	85285	1156.0	ARE3189	CCCPF	A319	ARE	3189.0		
2	10117	84389	1220.0	AVA9505	N821AV	A320	AVA	9505.0		
3	10117	84412	1227.0	AVA9459	N567AV	A320	AVA	9459.0		
4	10117	85350	1242.0	AVA9447	N691AV	A319	AVA	9447.0		

Fuente Elaboración propia

En la anterior grafica se muestran el documento xlsx una vez es exportado. Lo que se pretende hacer en el siguiente código es eliminar todas las columnas excepto por la de “Fecha”. Una vez realizado este procedimiento se escriben algunas líneas de código para determinar cuántas operaciones hubo por mes.

Ilustración 20 Lectura de archivos y formatear dato fecha

```
[9] df9 = pd.read_excel('registrovuelos2019.xlsx')
df8 = pd.read_excel('registrovuelos2018.xlsx')
df7 = pd.read_excel('registrovuelos2017.xlsx')

[20] []
dfc1 = pd.concat([df9,df8,df7], ignore_index=True)
dfc2 = pd.concat([df9,df8,df7], ignore_index=True)
dfc = pd.concat([dfc1, dfc2], ignore_index=True)

[22] columns = ['Fecha','ID',]
dfs = dfc[columns]
dfs['Fecha'] = dfs['Fecha'].astype(str)

dfs['date'] = dfs['Fecha'].str.split('')
# []
for i in range(len(dfs)):
    #dfs['test'] = dfs.loc[1,'date']
    year = '20'+ dfs.loc[i,'date'][-3] + dfs.loc[i,'date'][-2]
    month = dfs.loc[i,'date'][-5] + dfs.loc[i,'date'][-4]
    dfs.loc[i,'Fecha'] = dfs.loc[i,'date'][-7] + dfs.loc[i,'date'][-6] + ' ' + month + ' ' + year

dfs = dfs[columns]
dfs['Fecha'] = pd.to_datetime(dfs.Fecha)

[23] dfs.set_index('Fecha', inplace=True)
```

Fuente Elaboración propia

Al importar los archivos XLSL, obtendremos nuestra primera gráfica, como se muestra en la siguiente imagen, la cual presenta las operaciones aéreas que ocurrieron mensualmente a partir del año 2017 hasta diciembre de 2019.

Se puede observar que los primeros cinco meses del año 2017 el número de operaciones aéreas en el aeropuerto camilo daza estuvo entre 1300 y 1500 por mes, mientras que en 2018 en estos meses el número de operaciones aéreas fue de 1300 a 1500 por mes, ya en 2019 en los primeros cinco meses el número de operaciones aéreas se mantuvo entre 1450 y 1550 operaciones por mes. Para esta primera parte podemos notar que el número de operaciones dentro de los primeros cinco meses en los años 2017 a 2019 se encuentra en un rango de 1300 y 1600 por mes.

Ilustración 21 Gráfica de data de archivos cargadas

```
[24] #dfs.set_index('Fecha', inplace=True)
      t1 = dfs['ID'].resample('M').count()
      plt.figure(figsize=(10, 1))
      plt.plot(t1)
      plt.title('...')
      plt.grid(True)
      plt.show()
```



Fuente Elaboración propia

Media Móvil

Para el análisis de los registros de operaciones aéreas del aeropuerto Camilo Daza utilizaremos en nuestro código la media móvil. Como primera medida, esto facilita la visualización de la cantidad de operaciones a lo largo de los meses en los años graficados. Por otro lado, permite hacer un análisis exploratorio a mayor profundidad. Esto se ve ilustrado cuando por medio de la gráfica se pueden identificar tendencias, ya sean alcistas o bajistas o inclusive periodos de tiempo con un comportamiento estable con poca variabilidad.

Ilustración 22 Media Movil

```
def plotMovingAverage(series, window, plot_intervals=False, scale=1.96, plot_anomalies=False):  
  
    """  
    series - dataframe with timeseries  
    window - rolling window size  
    plot_intervals - show confidence intervals  
    plot_anomalies - show anomalies  
    """  
  
    rolling_mean = series.rolling(window=window).mean()  
  
    plt.figure(figsize=(15,2))  
    plt.title("Moving average\n window size = {}".format(window))  
    plt.plot(rolling_mean,marker='+', label="Rolling mean trend")  
  
    # Plot confidence intervals for smoothed values  
    if plot_intervals:  
        mae = mean_absolute_error(series[window:], rolling_mean[window:])  
        deviation = np.std(series[window:] - rolling_mean[window:])  
        lower_bond = rolling_mean - (mae + scale * deviation)  
        upper_bond = rolling_mean + (mae + scale * deviation)  
        plt.plot(upper_bond, "r--", label="Upper Bond / Lower Bond")  
        plt.plot(lower_bond, "r--")  
  
    plt.plot(series[window:], label="Actual values")  
    plt.legend(loc="upper left")  
    plt.grid(True)
```

Fuente Elaboración propia

En la imagen anterior se puede observar que en el código que se trabajó con el promedio móvil simple (SMA), el cual nos permite obtener las tendencias a lo largo de cierto tiempo. Con la librería pandas podemos utilizar la función `rolling().mean()`, la cual nos permitirá hallar el SMA.

Promedio Móvil = suma de datos a lo largo del tiempo/ período de tiempo

El siguiente grafico podemos observar los intervalos de confianza, representados por los extremos inferiores y superiores de las operaciones aéreas (rojo). Adicionalmente, se describe la tendencia de la media móvil (verde). Como resultado, con la media móvil se pueden sacar conclusiones como las tendencias y con los intervalos de confianza definir entre que rangos es probable que estén los valores reales.

Ilustración 23 Intervalos de confianza



Fuente Elaboración propia

Tabla Machine Learning

Para la generación de la siguiente tabla lo que se hace es utilizar la función DataFrame de la librería Pandas, la cual permite graficar los datos en una tabla par luego realizar predicciones, la tabla presenta unos ítems con la etiqueta de lag_3, lag_6, lag_9, y lag_12, estos ítems corresponden a la operaciones aéreas en los mese s y años correspondientes, obteniendo una relación con los meses de acuerdo a la secuencia, esto quiere decir 3 meses antes, 6 meses antes, 9 meses antes, y 12 meses antes de la fecha proporcionada, esto va generando un patrón que nos lleva a una predicción.

Ilustración 24 Preparación tabla con patrón

```
data = pd.DataFrame(t1.copy())
data.columns = ["y"]

# Adding the lag of the target variable from 6 steps back up to 24
for i in [3, 6, 9, 12]:
    data["lag_{}".format(i)] = data.y.shift(i)
# take a look at the new dataframe
data.tail(7)
```

	y	lag_3	lag_6	lag_9	lag_12
Fecha					
2019-06-30	1550	1526.0	1484.0	1480.0	1504.0
2019-07-31	1464	1464.0	1498.0	1438.0	1492.0
2019-08-31	1530	1482.0	1560.0	1486.0	1526.0
2019-09-30	1486	1550.0	1526.0	1484.0	1480.0
2019-10-31	1596	1464.0	1464.0	1498.0	1438.0
2019-11-30	1586	1530.0	1482.0	1560.0	1486.0
2019-12-31	1676	1486.0	1550.0	1526.0	1484.0

```
[ ] from sklearn.linear_model import LinearRegression
     from sklearn.model_selection import cross_val_score, TimeSeriesSplit

# for time-series cross-validation set 5 folds
tscv = TimeSeriesSplit(n_splits=5)
```

Fuente Elaboración propia

Predicción

El siguiente código nos permite separar los datos en dos de tal manera que con la primera parte se puede entrenar el modelo y la segunda parte probarlo y evaluar que tan bien predice.

Ilustración 25 Aprendizaje y evaluación de precisión.

```
def timeseries_train_test_split(X, y, test_size):  
    """  
        Perform train-test split with respect to time series structure  
    """  
  
    # get the index after which test set starts  
    test_index = int(len(X)*(1-test_size))  
  
    X_train = X.iloc[:test_index]  
    y_train = y.iloc[:test_index]  
    X_test = X.iloc[test_index:]  
    y_test = y.iloc[test_index:]  
  
    return X_train, X_test, y_train, y_test  
y = data.dropna().y  
X = data.dropna().drop(['y'], axis=1)  
  
# reserve 30% of data for testing  
X_train, X_test, y_train, y_test = timeseries_train_test_split(X, y, test_size=0.5)  
# machine learning in two lines  
lr = LinearRegression()  
lr.fit(X, y)  
#lr.fit(X_train, y_train)
```

Fuente Elaboración propia

El código a continuación crea una tabla para poder ingresar al modelo y esta pueda predecir las operaciones de los meses de enero febrero y marzo de 2020.

Ilustración 26 Ingreso al modelo

LinearRegression()

```
[ ] predictSize = 3
dataPredict = data.copy().tail(predictSize)
dataPredict.drop(['lag_12'], axis=1, inplace=True)
dataPredict.columns = ['lag_3', 'lag_6', 'lag_9', 'lag_12']

dataPredict.reset_index(inplace=True)
#dataPredict.index[0] = dataPredict.index[2]
for i in range(1,predictSize+1):
    dataPredict.Fecha.iloc[i-1] = dataPredict.Fecha.iloc[-1] + relativedelta(months=i)

dataPredict.set_index('Fecha',inplace=True)
dataPredict
```

	lag_3	lag_6	lag_9	lag_12
Fecha				
2020-01-31	1596	1464.0	1464.0	1498.0
2020-02-29	1586	1530.0	1482.0	1560.0
2020-03-31	1676	1486.0	1550.0	1526.0

```
[ ] kk = x_test.copy()
kk = kk.append(dataPredict)
#dataPredict.append(kk)
```

Fuente Elaboración propia

MODELO MACHINE LEARNING

El siguiente código permite generar la gráfica que describe el comportamiento del modelo de Machine Learning con respecto a los valores reales.

Ilustración 27 Código predicción

```
def plotModelResults(model, X_train=X_train, X_test=X_test, plot_intervals=False, plot_anomalies=False):  
    """  
        Plots modelled vs fact values, prediction intervals and anomalies  
    """  
  
    prediction = model.predict(X_test)  
  
    trainedPrediction = model.predict(X_train)  
  
    futurePrediction = model.predict(kk)  
    lab = X_test.index#[6,7,8,9,10,11,12,13,14,15]  
    plt.figure(figsize=(10, 7))  
  
    plt.plot(lab,prediction, '--',color='blue', marker='+',label="Predicción", linewidth=2.0)  
    #plt.plot(lab, y_test.values ,color='blue', label="Real", linewidth=2.0)  
  
    plt.plot(X_train.index, trainedPrediction,'--', marker='+',color='green',label="Predicción Entrenada", linewidth=2.0)  
    #plt.plot(X_train.index, y_train.values ,color='green',label="Real", linewidth=2.0)  
  
    plt.plot(kk.index[-(predictSize+1):], futurePrediction[-(predictSize+1):],'--',marker='+',color='c', label="Predicción Futura", linewidth=2.0)  
  
    plt.plot(t1.tail(24),label="Valores Reales")  
    if plot_intervals:  
        cv = cross_val_score(model, X_train, y_train,  
                             cv=tscv,  
                             scoring="neg_mean_absolute_error")  
  
        mae = cv.mean() * (-1)  
        deviation = cv.std()  
  
        scale = 1.96  
        lower = prediction - (mae + scale * deviation)  
        upper = prediction + (mae + scale * deviation)  
  
        plt.plot(lab,lower, "r--", label="upper bond / lower bond", alpha=0.5)  
        plt.plot(lab,upper, "r--", alpha=0.5)  
  
    error = mean_absolute_percentage_error(prediction, y_test)  
    plt.title("Mean absolute percentage error {0:.2f}%".format(error))  
    plt.legend(loc="best")  
    plt.tight_layout()  
    plt.grid(True);
```

Fuente Elaboración propia

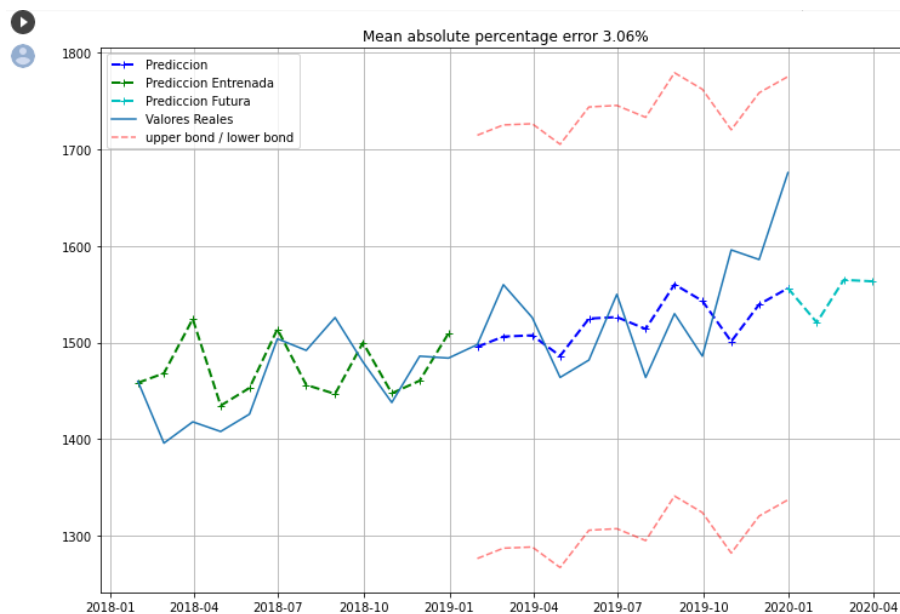
La siguiente gráfica describe el comportamiento del modelo de Machine Learning con respecto a los valores reales. Primeramente, la línea azul continua detalla la cantidad de operaciones que hubo a lo largo de los años 2018, 2019 y 2020. Por otro lado, la línea verde

intermitente es la predicción que hace conociendo los datos reales. Ese tipo de predicción entrenada tiene una particularidad y es que el modelo de Machine learning conoce cuáles son los valores reales y con base a ellos busca los mejores coeficientes. La línea azul intermitente refleja qué tan eficiente es el modelo para predecir las operaciones en meses que no sabe cuál es el valor real. Esto nos permite conocer visualmente qué tan preciso es el modelo para predecir. El azul primario cian indica las predicciones para el mes de enero, febrero y marzo de 2020.

Lo que la predicción de esos 3 meses del año 2020 nos indica es que para el mes de enero las operaciones van a tener una leve caída con respecto al mes de diciembre sin embargo para el mes de febrero y marzo la cantidad de operaciones pueden aumentar a niveles similares al del mes de diciembre.

Cabe aclarar que los datos de las operaciones del año 2020 hasta la fecha no se tomaron en cuenta ya que en estas fechas las operaciones tuvieron un comportamiento anormal donde las operaciones cayeron drásticamente por causa de la pandemia. Por consiguiente, utilizar esa información de la pandemia podría sesgar los datos y generar patrones anormales o dispersos.

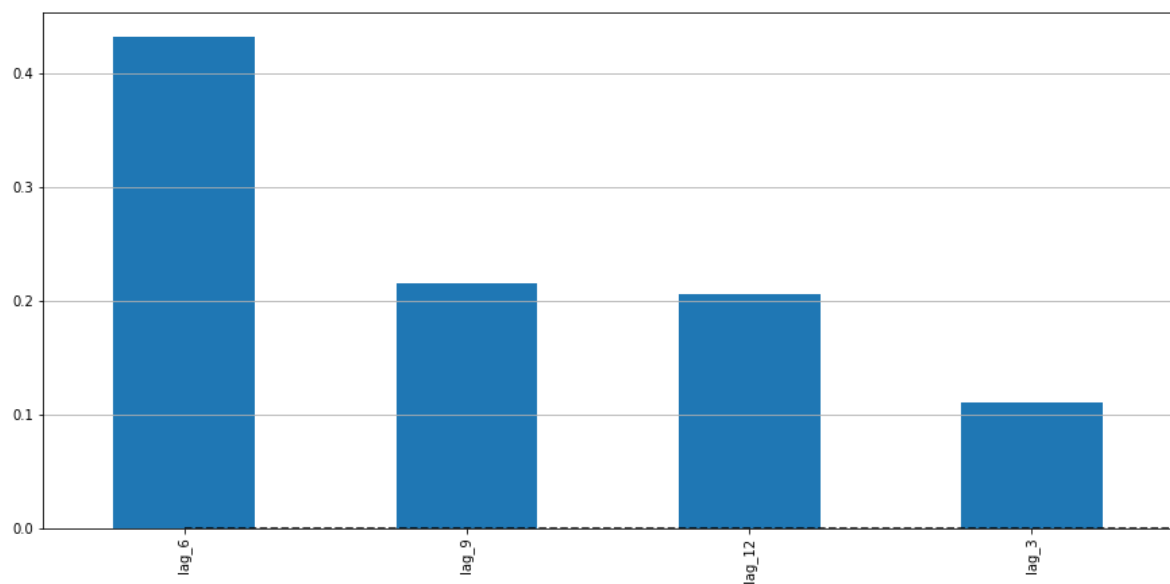
Ilustración 28 Gráfica predicción



Fuente Elaboración propia

La siguiente tabla muestra los coeficientes para cada variable que se usó en el modelo. La mejor variable fue el lag-6. Esto quiere decir que estos datos son los que mejor ayudan a la predicción. Por el contrario, el que menos aportes entrega es el lag 3. En caso de tener que eliminar una variable esta última sería la seleccionada.

Ilustración 29 Coeficientes



Fuente Elaboración propia

APLICATIVO WEB

Durante el proceso de desarrollo, en código principal del producto como bien se conoce y describe en este documento está en el lenguaje de programación de Python; sin embargo, debemos entender que, aunque cumple con su cometido, para la interacción del usuario es un poco “brusco” y puede generar frustración, dudas e inclusive errores de uso y ejecución.

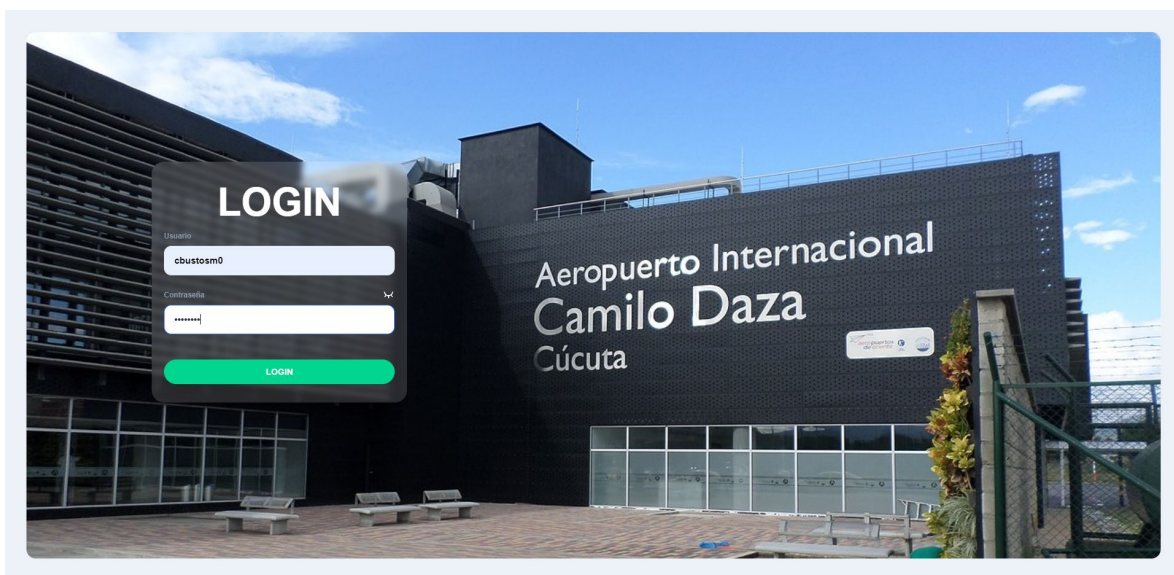
Por esto, se decidió integrar una interfaz amigable al usuario final, para controlar el proceso y controlando posibles errores de usuario; además de agregar una capa de protección de acceso con usuario y contraseña.

La interfaz fue desarrollada como página web con el framework Angular.

La interacción del usuario es la siguiente:

1. Al ingresar a la URL designada para la página, el usuario deberá ingresar usuario y contraseña pre-designados. Esto permite que solo usuarios autorizados ingresen a la plataforma y realizar el proceso.

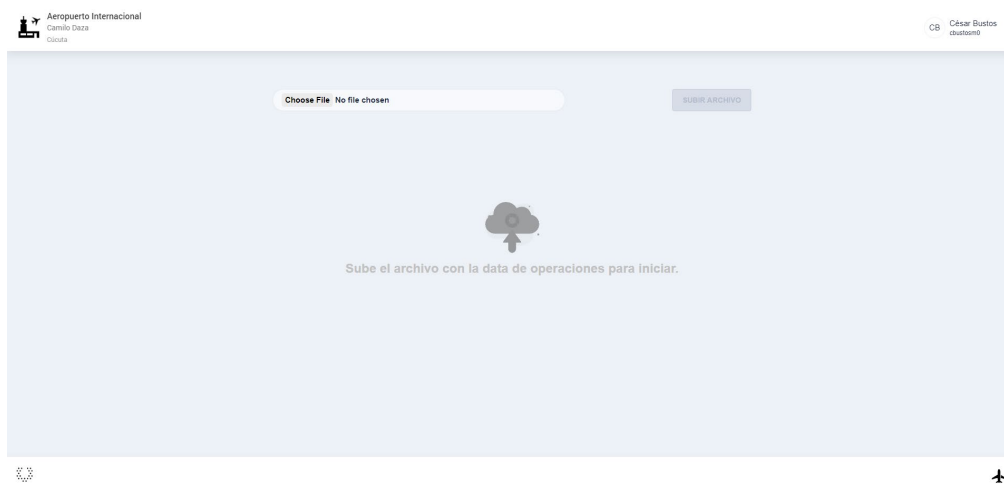
Ilustración 29 Login plataforma



Fuente Elaboración propia

2. En la página principal le solicita que seleccione de su computador el archivo con los datos de las operaciones aéreas con las que se desean realizar la predicción.

Ilustración 30 Seleccionar archivo

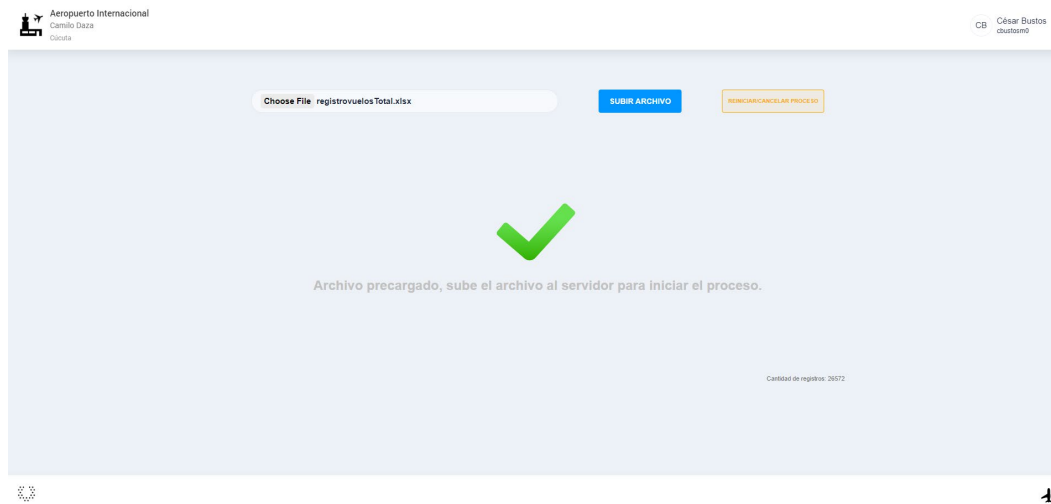


Fuente Elaboración propia

3. Una vez seleccionado el archivo, se habilita el botón de “subir archivo”, el cual subirá el archivo al servidor y dejarlo listo para su uso.

4.

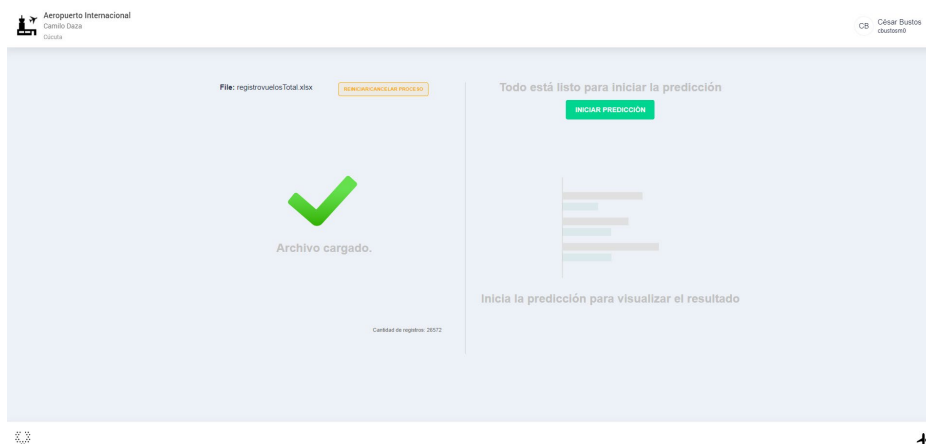
Ilustración 31 Archivo listo para subir al server



Fuente Elaboración propia

- Al subir el archivo la página habilitará una sección para que el usuario elija el momento en el que quiere iniciar el proceso de la predicción. Debido a que el proceso puede variar su duración de ejecución, se le permite al usuario el momento de iniciarlo y en caso de que decida o tenga que cancelarlo evitar el uso de recursos innecesarios.

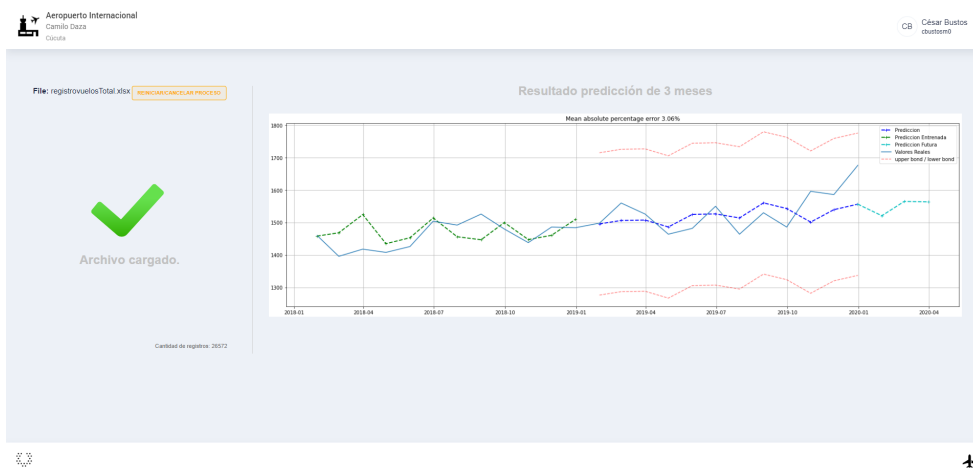
Ilustración 32 Archivo cargado listo para iniciar predicción.



Fuente Elaboración propia

- Al presionar el botón “Iniciar proceso”, y luego de esperar algunos segundos mostrará el resultado de la predicción según los datos de la data del archivo subido.

Ilustración 33 Resultado de predicción



Fuente Elaboración propia

CONCLUSIONES

En este trabajo se creó una solución basada en un modelo de predicción de Machine Learning, que permite pronosticar la densidad de operaciones y poder así optimizar la programación de turnos de los controladores aéreos en el aeropuerto Camilo Daza de la ciudad de Cúcuta. Lo más importante de esta solución es que permite mostrar mediante un modelo gráfico la densidad de operaciones para un mes futuro. Por ejemplo, permite indicar si la cantidad de operaciones va a ser mayor o menor respecto a un mes anterior. Esto permite mejorar el criterio para determinar la cantidad de controladores necesarios para atender la operación de la mejor manera posible.

El resultado es una planilla de turnos más eficaz ya que tiene en cuenta la cantidad de operaciones para distribuir así su capital humano. Es decir, si en un mes se necesitó un determinado número de personal y luego de analizar la aplicación se tiene que las operaciones aéreas aumentarán entonces la mejor decisión sería aumentar el personal para atender el pronóstico. Si, por el contrario, se espera una disminución en la densidad de trabajo entonces la cantidad de personal programado para la operación podría ser menor y el restante podría aprovechar para realizar los cursos obligatorios o demás ofertas que se tienen para el crecimiento profesional de la planta.

Por otro lado, el lenguaje principal que se utilizó para el presente trabajo fue Python por razones como: la práctico debido a su curva rápida de aprendizaje y a la documentación que existe sobre temas de aprendizaje de máquina como los que se pretendía realizar en esta ocasión.

Adicionalmente, fue muy conveniente para realizar procesos como el de extracción y transformación de los datos. Esto se puede evidenciar cuando se cargan los archivos anuales de las operaciones diarias en un formato Excel y por medio de código se transforma en una sola tabla de 2 columnas: La cantidad de operaciones y el mes al cual corresponden.



Más allá, este lenguaje fue muy valioso para realizar el análisis exploratorio de los datos ya que cuenta con librerías especializadas, como pandas y matplotlib por nombrar algunos, que se especializan en el tratamiento de datos y permiten una visualización de forma sencilla. Un ejemplo es el uso de gráficas que presentan la cantidad de operaciones en el eje Y, y el mes al cual corresponde en el eje X, junto con una media móvil. Esto permite definir su tendencia, alcista o bajista, logrando así una predicción inicial y básica para una programación de turnos más metodológica e informada.

El logaritmo empleado para la aplicación de Machine learning fue Linear Regression o regresión lineal en español. Por medio de él se pudo entrenar a un modelo que acepta el compendio de las operaciones del aeropuerto Camilo Daza, que en este caso correspondió a un período de 3 años, 2017 a 2019. Lo ideal es que anualmente se le agregue las operaciones de los 12 meses. Esto permite que su funcionamiento mejore cada vez y produzca resultados más cercanos a la realidad.

Aunque el algoritmo funciona mejor si se tiene información de años actuales, eso no se pudo lograr ya que, a causa de la pandemia que empezó en el 2020 hasta el presente año, la cantidad de vuelos disminuyó de una manera altamente impredecible. Por consiguiente, no aportarían valor para la obtención de los coeficientes matemáticos que es lo que hace el algoritmo. A pesar de lo anterior, la aplicación logra predecir de manera gráfica hasta 3 meses futuros indicando si la densidad de tráfico aumenta o disminuye respecto al mes anterior.

Como se mencionó hubo un obstáculo que se tuvo para la realización del presente trabajo fue la obtención de los datos. La Aerocivil, que es la entidad por la cual solicitamos la información en Excel, nos proporcionó los datos solo a partir del 2017 debido a limitaciones técnicas por parte de ellos. Aparte de lo anterior, la información a partir del 2020, que es la época cuando inició la pandemia, no se tuvo en cuenta porque no es útil tal como se mencionó anteriormente. Esto se tradujo a que solo se utilizaron 3 años para la realización de este modelo de Machine learning. Por consiguiente, para un futuro se recomienda ingresar nuevos datos; es decir la cantidad de



operaciones en un formato de Excel actualizado anualmente con el propósito de obtener unos coeficientes de predicción más precisos. Dado que entre más datos se abarquen, mejor es su rendimiento.

De igual manera para ahondar en el presente trabajo se recomienda crear nuevas variables que el modelo pueda tener en cuenta a fin de agregarle valor. Algunas variables pueden ser binarias, como por ejemplo determinar si el mes es lluvioso o no. Ese indicador podría ayudar a predecir si a causa del mal tiempo, aumentan las operaciones debido a cierres de aeropuertos cercanos que obligan al tránsito que inicialmente llegaba allí a que tengan que desviarse a Cúcuta. Otra variable puede ser continua como por ejemplo determinar qué porcentaje de las operaciones de dicho mes fueron de aerolíneas comerciales o de aviación militar. O incluso hacer uso de la media móvil para asignarle a cada mes el promedio de operaciones de un periodo determinado. Todo esto puede dar la luz un modelo para la predicción de la densidad de operaciones mucho más robusto y confiable que al final de todo brindé gran ayuda a incrementar los estándares de seguridad en la aviación y a mejorar la distribución del capital humano por medio de la hechura de turnos.

Como resultado adicional y no menos importante también se ha desarrollado una interfaz web para permitir una interacción más amigable con el usuario y que sea fácil para replicarlo en demás aeropuertos del país, de la región y del mundo. Entre las principales bondades de este logro está que a las personas autorizadas se les asigna un usuario y una contraseña para que solo personas confiables puedan manipular el sistema de forma segura. Igualmente, permite que su navegación sea intuitiva para que permita cargar la información en Excel y producir predicciones a 1, 2 o 3 meses a futuro.

Para ello también se ha creado un servidor para que la aplicación pueda ser accedida desde diferentes puntos geográficos, ya sea desde diferentes casas en una misma ciudad o en ciudades diferentes; este servidor el cual es quien proporciona el acceso a la aplicación, podrá ser expuesto



a internet y según el alcance necesario las operaciones del aeropuerto tener una exposición global, sin embargo, lo ideal es generar un despliegue local en la zona donde se requiere el uso exclusivo de este programa, como lo es solo dentro de la red del aeropuerto Camilo Daza; y ya en un futuro según su eficiencia y resultados, plantearse su expansión a otros centros de control de operaciones.

Tener en cuenta que, como todo producto de software, y en caso de querer en algún futuro continuar o mejorando el desarrollo es necesario mantenerlo actualizada sus librerías/dependencias y consultar la documentación de estas, debido a que los métodos o funcionamientos pueden cambiar e inclusive pueden de dejar de existir estos recursos siendo necesario el remplazo.

Referencias

1. ICAO. (2007). Gestión del tránsito aéreo. *Document PDF*. Recuperado de <https://www.icao.int/SAM/Documents/2010/ASTERIX/07%20%20DOC4444.pdf>
2. Marrón, D, (2018) Factores humanos en aviación: *CRM (Crew resource management – gestión de recursos de la tripulación)*. *Document PDF*. Recuperado de <http://papelesdelpsicologo.es/pdf/2870.pdf>
3. AEROCIVIL (2021) Controladores de tránsito aéreo. *Recuperado de* [https://www.aerocivil.gov.co/servicios-a-la-navegacion/Gestion-de-los-Servicios-de-Transito-Aereo-ATS/controladores-de-tr%C3%A1nsito-a%C3%A9reo-\(atco\)](https://www.aerocivil.gov.co/servicios-a-la-navegacion/Gestion-de-los-Servicios-de-Transito-Aereo-ATS/controladores-de-tr%C3%A1nsito-a%C3%A9reo-(atco))
4. Antonio, M., Piña, L., (2012) Los controladores de tráfico aéreo y la seguridad aérea. *Document PDF*. Recuperado de <https://www.redalyc.org/pdf/325/32523131005.pdf>
5. EDUWEB (s.f.) Controlador de tráfico aéreo. *Recuperado de* <https://www.educaweb.com/profesion/controlador-trafico-aereo-9/>
6. Duque, O., Edison, J., Mercado, M., (2011). Medición de la calidad percibida del servicio de control de tránsito aéreo. *Recuperado de* http://www.scielo.org.co/scielo.php?pid=S0121-50512011000300013&script=sci_abstract&tlng=es
7. ICAO (2016). Servicios de tránsito aéreo. *Document PDF*. Recuperado de <https://www.udi.edu.co/images/biblioteca/aeronautica/anexo11.pdf>
8. Beltrán, G., Cabrera, L., (2020). Diagnóstico de la función del controlador de tránsito aéreo como servidor público en la región Norte de Santander dentro de la Unidad Especial de Aeronáutica Civil. *Document PDF*. Recuperado de <https://repository.unad.edu.co/bitstream/handle/10596/38931/gabeltranlo.pdf?sequence=3&isAllowed=y>
9. Barragán, A., (2017). Establecimiento de un método casual para la predicción y medición de presentaciones de seguridad operacional de un servicio de gestión de tráfico aéreo. *Document PDF*. Recuperado de <https://repository.unad.edu.co/bitstream/handle/10596/38931/gabeltranlo.pdf?sequence=3&isAllowed=y>
10. ICAO (agosto 2007). Establecimiento de los principios de Factores Humanos dentro del AIM y elaboración de guías que faciliten su aplicación en los Estados CAR/SAM y un Plan para su implantación. *Document PDF*. Recuperado de <https://www.icao.int/SAM/Documents/GREPECAS/2007/QMTF03/QMTF03NE12.pdf>
11. Acuña, C., (2013). El entrenamiento CRM (*Crew Resource Management*) aplicado al ámbito de la fuerza de submarinos de la armada argentina (Operaciones de submarinos y de buzos tácticos). *Document PDF*. Recuperado de http://nulan.mdpu.edu.ar/1928/1/acuna_ch_2013.pdf

12. Marrón, D., (2018). Factores Humanos En Aviación: CRM (*Crew Resource Management - Gestión De Recursos De La Tripulación*) Human Factors In Aviation: Crm (*Crew Resource Management*). Recuperado de <https://www.redalyc.org/journal/778/77857281007/html/>
13. Augusto, J., (2018). Evolución y conceptos del CRM. Recuperado de <https://a21.com.mx/investigar-para-prevenir/2018/06/29/evolucion-y-conceptos-del-crm-primera-parte>
14. EUROCONTROL (2006). European air traffic management. *Document PDF*. Recuperado de <https://skybrary.aero/sites/default/files/bookshelf/690.pdf>
15. Escalona, E., Urosa, E., González, R., Romero, E., Lamarca, R., Jirón, C., Bello, C., (1996). Fatiga laboral en controladores de tránsito aéreo. Recuperado de <https://dialnet.unirioja.es/servlet/articulo?codigo=6477181>
16. Suárez, A., (2017). Asignación de controladores a sectores aéreos mediante metaheurísticas trayectoriales: Búsqueda Tabú. *Document PDF*. Recuperado de https://oa.upm.es/47930/1/TFM_ADAN_SUAREZ_CUESTA.pdf
17. Raschka, S., (2015). Python machine learning. Recuperado de https://books.google.com.co/books?hl=es&lr=&id=GOVOCwAAOBAJ&oi=fnd&pg=PP1&dq=python+machine+learning&ots=NdbAJiUUVD&sig=skGClw0S6BXYt5l0nd-O3voPPr4&redir_esc=y#v=onepage&q=python%20machine%20learning&f=false
18. Grandell, L., Peltomaki, M., Back, R., Salakoski, T., (s.f). Why Complicate Things? Introducing Programming in High School Using Python. *Document PDF*. Recuperado de https://d1wqtxts1xzle7.cloudfront.net/39124805/0912f507d5a05149d8000000-with-cover-page-v2.pdf?Expires=1645312991&Signature=b8OWho4Y~FWRS96gtSBBNrGb3yII44U5aAC01z1xCWKsvDJS3jqvxy8SUU2cUtX8iZgNDb7IoXEVJrq4FB84vpi47yJH7iARierrXgwAhwBfpEVmZKTDbthgVvarJ~do4z9GmwT7i-nDfCW-LLBrJ1s-ivGUDqWZekWKPg92cuOHS4qhVONpVil0aYG2lGuCb33ZFvuYwjYR21FB1PNU3owITzFeF8jgh9zjsPSiD2Fh4rJGbn~a5uXR3oYx0WhiAaDopHHPLWxzeYP~wFjP6ymw7m1vISIA1DezEht7VEkRK3DHI2azlBStJbqh98Q-0fGLGJWlm8rF62zisinJUXQ__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA
19. ICAO (2008) Factores humanos en el servicio de información aeronáutica. *Document PDF*. Recuperado de <https://www.icao.int/SAM/Documents/GREPECAS/2009/AIMSG12/ManualFactoresHumanos.pdf>
20. Estos serían los trabajos más estresantes, según experta en selección (29 de junio del 2016). MBA. Recuperado de <https://mba.americaeconomia.com/articulos/notas/estos-serian-los-trabajos-mas-estresantes-segun-experta-en-seleccion>
21. Petrozzi, B., (2018). Fatiga: historia, neuroanatomía y características psicopatológicas. Una revisión de la literatura. Recuperado de http://www.scielo.org.pe/scielo.php?script=sci_arttext&pid=S0034-85972018000300005

22. Fatigue Management: Guidance for air traffic controllers and air traffic engineers (s.f.) Skybrary. Recuperado de <https://skybrary.aero/articles/fatigue-management-guidance-air-traffic-controllers-and-air-traffic-engineers>
23. AEMPPI (25 de marzo del 2018). Peligros del ruido y sus efectos en nuestra salud. Recuperado de <https://www.elsevier.com/es-es/connect/actualidad-sanitaria/efectos-negativos-del-ruido-y-su-repercusion-en-nuestra-salud>
24. Malanche, S., (2010). Administración de proyectos de software aplicando restricciones: Un enfoque multidisciplinario caso de estudio. *Document pdf*. Recuperado de <https://repositorio.tec.mx/bitstream/handle/11285/628632/CEM336398.pdf>
25. Minguillón, J. Casas, J. y Minguillón, J. (2017). Minería de datos: modelos y algoritmos. Barcelona, Spain: Editorial UOC. Recuperado de <https://elibro-net.bdbiblioteca.universidadean.edu.co/es/ereader/bibliotecaean/58656?page=01>.
26. Roberto, H., Esguerra B, C., Quintana C, J., & Castañeda L, L. M. (2019). Foro IMPACTO DEL COVID-19 EN EL TRANSPORTE AÉREO: Nuevos Retos y oportunidades en la visión del Plan Estratégico Aeronáutico 2030. Aeronáutica Civil. <https://www.aerocivil.gov.co/aerocivil/IMPACTO%20DEL%20COVID19%20EN%20EL%20TRANSPORTE%20AREO/Nota%20de%20Estudio%20Competitividad.pdf>
27. Statista. (2022, 19 enero). COVID-19: variación interanual del número de vuelos desde países seleccionados 2022. <https://es.statista.com/estadisticas/1105535/covid-19-impacto-en-la-frecuencia-de-vuelos-de-las-aerolineas-mundiales/>
28. Bernal Torres, C. A. (2016). Metodología de la investigación: administración, economía, humanidades y ciencias sociales. Pearson Educación. Disponible en <https://bit.ly/3p11tXU>
29. López, J., (2017). Estimación de historias de usuario en Scrum a través de un Modelo basado en descomposición de la complejidad. *Document PDF*. Recuperado de <https://repositorioinstitucional.uabc.mx/bitstream/20.500.12930/5458/1/TIJ129490.pdf>
30. Rodríguez, C., (2015). ¿Po qué implementar Scrum? *Document PDF*. Recuperado de <https://journal.universidadean.edu.co/index.php/Revistao/article/view/1253/1218>
31. w. *Aplicación de algoritmos Random Forest y XGBoost en una base de solicitudes de tarjetas de crédito*. [https://elibro-net.bdbiblioteca.universidadean.edu.co/es/ereader/bibliotecaean/37737?page=45](http://www.scielo.org.mx/scielo.php?pid=S1405-77432020000300002&script=sci_arttext#:~:text=El%20algoritmo%20Random%20Forest%20(Breiman,de%20cada%20%C3%A1rboles%20por%20separado%20(,http://www.scielo.org.mx/scielo.php?pid=S1405-77432020000300002&script=sci_arttext#:~:text=El%20algoritmo%20Random%20Forest%20(Breiman,de%20cada%20%C3%A1rboles%20por%20separado%20(32. Nava, A. (2013). Procesamiento de series de tiempo (2a. ed.). FCE - Fondo de Cultura Económica. <a href=)
33. TimeAndDate (2022) Time zone map. Recuperado de <https://www.timeanddate.com/time/map/>
34. Orellana, J, (noviembre 2018). Árboles de decisión y random forest. Recuperado de <https://bookdown.org/content/2031/>

35. Salas, M., Alor, G., Valencia, R., Rodríguez, M., Rodríguez, A., López, J., (2015). Analyzing best practices on Web development frameworks: The lift approach. *Recuperado de* <https://www.sciencedirect.com/science/article/pii/S0167642314005735>
36. Universidad Católica de Chile (2014). Regresión lineal. *Recuperado de* <https://revistachilenadeanestesia.cl/regresion-lineal/>