Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Querétaro

Momento de Retroalimentación: Módulo 2 Análisis y Reporte sobre el desempeño del modelo.

Reporte de Análisis y Modelo de IA para Predecir Tarifas Aéreas en los EE.UU.

Autor:

A01368818 Joel Sánchez Olvera

TC3006C.101

Inteligencia artificial avanzada para la ciencia de datos I

Fecha:

7 de Septiembre del 2024

Índice

Abstracto

Este informe presenta el desarrollo y análisis de un modelo predictivo usado para determinar si las tarifas aéreas en los Estados Unidos superarán un umbral específico en los años futuros. Utilizando un conjunto de datos que incluye variables como el año, trimestre, ciudades, aeropuertos, número de millas recorridas, y tarifas, se realizó un exhaustivo proceso de preparación de datos, que incluyó la transformación de variables categóricas y la normalización de las características.

1. Introducción

Este proyecto tiene como objetivo principal desarrollar un modelo predictivo de **machine learning** para estimar tarifas aéreas en rutas específicas dentro de los Estados Unidos. Las aerolíneas utilizan precios dinámicos, ajustando sus tarifas en función de la demanda, competencia y otros factores de mercado.

Éste análisis permite optimizar las decisiones de modificaciones en los precios mediante herramientas predictivas que proporcionan estimaciones de tarifas más precisas.

Objetivos del Proyecto

1. **Predicción de tarifas aéreas**: Desarrollar modelos de regresión que puedan predecir tarifas futuras basadas en una variedad de características, como la ruta de vuelo, aerolínea, número de pasajeros y distancia entre aeropuertos.
2. **Clasificación de tarifas**: Implementar un modelo que clasifique si una tarifa superará o no un umbral específico (por ejemplo, $1000). Esto puede servir para detectar cuando las tarifas alcanzarán niveles altos o bajos que podrían afectar la demanda.
3. **Evaluación de sesgo y varianza**: Diagnosticar y explicar los niveles de sesgo (**bias**) y varianza para comprender el comportamiento de los modelos. Este análisis es crucial para evitar problemas como **underfitting** (ajuste insuficiente) o **overfitting** (sobreajuste) que afectan la capacidad predictiva del modelo.

Importancia del Proyecto

Este proyecto puede ayudar tanto a las aerolíneas como a los clientes de las mismas:

- **Aerolíneas**: El análisis predictivo mejora la capacidad de las aerolíneas para anticipar cambios en la demanda y ajustar precios de acuerdo a estos. Esto les permite optimizar su estrategia de precios de manera dinámica, con la finalidad de maximizar las ganancias para la aerolínea.
- **Clientes**: Proporcionar predicciones de tarifas precisas también beneficia a los clientes, pues pueden planificar y anticipar mejor sus compras de boletos para vuelos, aprovechando las mejores tarifas disponibles según las predicciones.

Conceptos Clave

ETL (Extract, Transform, Load)

Al usar un dataset de las tarifas de vuelos aéreos, el proceso ETL es crucial para preparar los datos para que el modelo funcione de manera efectiva. Este proceso consta de las siguientes etapas:

1. **Extracción (Extract)**: Los datos se extraen de diversas fuentes. En este proyecto, los datos de tarifas aéreas se obtienen de un archivo CSV que incluye información sobre rutas de vuelo, aeropuertos, distancias y tarifas.
2. **Transformación (Transform)**: Aquí los datos se preparan y transforman para hacerlos utilizables por el modelo:
    - **Limpieza de datos**: Se eliminan valores faltantes, inconsistencias y ruido en los datos.
    - **Mapeo de variables categóricas**: Las características categóricas, como los aeropuertos y aerolíneas, se convierten a valores numéricos. Esto es esencial porque los modelos de machine learning requieren valores numéricos para poder procesar la información.
    - **Normalización de características**: Las variables numéricas se escalan para asegurarse de que estén en rangos similares, mejorando la eficiencia del entrenamiento del modelo. Esto es

particularmente importante para modelos como redes neuronales, que son sensibles a escalas diferentes.

3. **Carga (Load)**: Una vez que los datos han sido transformados, se cargan en un formato adecuado (en este caso, un DataFrame de Pandas) para su análisis posterior.

Modelos de Aprendizaje Automático Utilizados

Regresión Lineal

La regresión lineal es un modelo estadístico fundamental que busca establecer una relación lineal entre una variable dependiente (la tarifa aérea en este caso) y una o más variables independientes (como la distancia, el aeropuerto de origen y destino, el tipo de aerolínea, etc.).

La fórmula general de la regresión lineal es:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon$$

Donde:

- y es la variable objetivo (tarifa predicha).
- $x_1, x_2 \ldots x_n$ son las variables independientes (características del vuelo).
- $\beta_0, \beta_1, \ldots, \beta_n$ son los coeficientes ajustados durante el entrenamiento.
- ϵ es el error residual.

La regresión lineal es eficiente y fácil de interpretar cuando las relaciones entre las variables son lineales. Sin embargo, puede no capturar relaciones un poco más complejas.

Redes Neuronales Artificiales (ANN)

Las redes neuronales son un modelo de machine learning más avanzado que permite capturar relaciones no lineales entre las características. En este proyecto, se utiliza una red neuronal densa o **feedforward** con múltiples capas ocultas, lo que permite que el modelo detecte patrones complejos en los datos.

La estructura básica de una red neuronal puede representarse mediante la siguiente ecuación:

$$\hat{y} = f(W_2 * f(W_1 * x + b_1) + b_2)$$

Donde:

- $\hat{y}$ es la predicción de la tarifa aérea.
- f es una función de activación (en este caso, **ReLU**).
- W_1, W_2 son los pesos asociados a las capas ocultas de la red.

- b_1, b_2 son los sesgos asociados a las capas.

El entrenamiento de las redes neuronales se basa en el algoritmo de **Backpropagation**, el cual ajusta los pesos W_n minimizando el error en las predicciones mediante un proceso de **descenso de gradiente**.

Evaluación del Modelo: Bias, Varianza y Ajuste

Es fundamental, además de entrenar los modelos, diagnosticar su comportamiento para entender el grado de bias y varianza, así como el nivel de ajuste.

Estos factores determinan si el modelo presenta **underfit** (subajustado), **fit** (bien ajustado) o **overfit** (sobreajustado).

1. **Bias (Sesgo)**:
   - El **sesgo** es la diferencia entre la predicción promedio del modelo y el valor real que intentamos predecir. Un alto sesgo indica que el modelo es demasiado simple y no está capturando los patrones de los datos (problema de **underfitting**).
   - **Diagnóstico**: Si el modelo tiene un sesgo alto, se debe ajustar la complejidad del modelo, agregando más capas, cambiando el tipo de modelo, o aumentando las interacciones entre variables.
2. **Varianza**:
   - La **varianza** mide la sensibilidad del modelo a las fluctuaciones en los datos de entrenamiento. Un modelo con alta varianza tiende a sobreajustar los datos (problema de **overfitting**), lo que significa que puede predecir bien los datos de entrenamiento, pero falla al generalizar a datos nuevos.
   - **Diagnóstico**: La varianza alta se puede reducir utilizando técnicas como la regularización o aumentando la cantidad de datos de entrenamiento.
3. **Ajuste del Modelo**:
   - **Underfitting**: Ocurre cuando el modelo es demasiado simple y no puede captar las relaciones subyacentes en los datos.
   - **Overfitting**: Ocurre cuando el modelo es demasiado complejo y se ajusta demasiado bien a los datos de entrenamiento, fallando en generalizar a datos nuevos.

Para este proyecto, se implementa un enfoque que evalúa el rendimiento del modelo en los conjuntos de **entrenamiento**, **prueba** y **validación**,

y se ajusta el modelo para minimizar el riesgo de sesgo y varianza.

2. Preprocesamiento de Datos y Descripción del Dataset

El objetivo del preprocesamiento es convertir los datos originales en un formato adecuado para ser utilizado en los modelos de machine learning, asegurando que las variables sean comprensibles y relevantes para la tarea de predicción.

Descripción del Dataset

El dataset utilizado contiene información detallada sobre tarifas aéreas en distintas rutas de los Estados Unidos. Cada registro corresponde a una observación que incluye variables relacionadas con la ruta de vuelo, los aeropuertos involucrados, el número de pasajeros, la aerolínea, y las tarifas.

Para la construcción del modelo, seleccionamos un conjunto de variables que resultan cruciales para la predicción de tarifas aéreas. Estas variables incluyen tanto características categóricas como numéricas que, tras su preprocesamiento, se convierten en inputs numéricos para el modelo. Las variables clave utilizadas fueron:

1. **Year**: Se convierte en una variable categórica mapeada a valores numéricos.
2. **Quarter**: Similar a la variable de año, se trata como categórica.
3. **City1 y City2**: Estas variables categóricas fueron mapeadas a valores numéricos utilizando diccionarios que asignan un valor único a cada ciudad.
4. **Airport_1 y Airport_2**: Variables categóricas mapeadas a valores numéricos. Son cruciales para identificar las rutas específicas y sus patrones tarifarios.
5. **Nsmiles**: Variable numérica que indica la distancia entre aeropuertos. Esta es una de las variables más importantes, ya que existe una correlación directa entre la distancia de vuelo y el costo.
6. **Passengers**: Otra variable numérica que proporciona información sobre la demanda.
7. **Carrier_lg y Carrier_low**: Se convirtieron en variables categóricas mapeadas a valores numéricos para representar las aerolíneas.

8. **Fare_lg y Fare_low**: Se incluyen como variables adicionales que ayudan a capturar el tipo de estructura de precios en cada ruta.

Estas variables son particularmente útiles para entrenar un modelo de predicción, ya que capturan no solo los aspectos cuantitativos (como la distancia o la cantidad de pasajeros), sino también los factores cualitativos (como las aerolíneas y los aeropuertos), lo que permite al modelo aprender patrones más complejos.

Importancia del Preprocesamiento de Variables

El preprocesamiento de las variables antes del entrenamiento del modelo es un paso fundamental, ya que asegura que los datos estén en un formato adecuado para ser procesados por los algoritmos. Este proceso incluye:

1. **Conversión de variables categóricas**: Las variables como ciudades y aeropuertos no pueden ser utilizadas directamente por los algoritmos, por lo que deben convertirse a valores numéricos mediante técnicas como en este caso el mapeo a valores enteros.
2. **Normalización de variables numéricas**: Variables como la distancia y el número de pasajeros pueden tener rangos muy diferentes, lo que afecta el entrenamiento del modelo. Por ello, se normalizan para que todas las variables tengan rangos comparables.
3. **Manejo de datos faltantes**: Los valores faltantes se imputan o eliminan según corresponda, para evitar inconsistencias en el conjunto de datos.

El conjunto de datos preprocesado está listo para ser dividido en conjuntos de entrenamiento, validación y prueba, asegurando así que el modelo se entrene de manera robusta y generalice bien a nuevos datos.

Preprocesamiento en el Proyecto

El preprocesamiento en este proyecto incluye la transformación y preparación del conjunto de datos para entrenar el modelo. Los datos crudos sobre tarifas aéreas son procesados de la siguiente manera:

1. **Transformación de Variables Categóricas**: Variables como aeropuertos, ciudades y aerolíneas se transforman en valores numéricos utilizando diccionarios

de mapeo, permitiendo que el modelo pueda procesarlas.
    - Ejemplo en el código: Los aeropuertos son mapeados de códigos como "LAX" a valores enteros como 1, utilizando diccionarios en el archivo `guide.py`.
2. **Limpieza y Eliminación de Valores Faltantes**: Se eliminan registros incompletos para evitar que el modelo se entrene con datos corruptos o incompletos, lo que podría reducir la precisión del modelo.
3. **Normalización de las Variables Numéricas**: Se utiliza el **StandardScaler** para asegurar que todas las características numéricas (por ejemplo, el número de millas entre aeropuertos) estén en la misma escala. Esto es crucial para modelos como redes neuronales que son sensibles a diferencias en magnitudes entre las variables de entrada.

El preprocesamiento es fundamental para asegurar que los datos estén en un formato adecuado y para garantizar que el modelo sea eficiente durante el entrenamiento.

Estructura de los Modelos Utilizados

Los modelos utilizados incluyen **redes neuronales** y **regresión lineal**. Se utilizan varias técnicas avanzadas para entrenar estos modelos y asegurar su generalización.

Redes Neuronales

En el proyecto, se utiliza un modelo de red neuronal densa con varias capas ocultas. El código implementa el siguiente flujo para entrenar la red:

- **Capas Ocultas**: Se definen varias capas ocultas con 64, 32, y 16 neuronas, respectivamente. La activación **ReLU** se utiliza en cada capa, lo que permite al modelo capturar relaciones no lineales entre las variables.
- **Capa de Salida**: La capa de salida tiene una sola neurona sin activación, ya que el objetivo es predecir un valor continuo (la tarifa).
- **Compilación del Modelo**: El modelo se compila utilizando el optimizador **Adam** y la función de pérdida **Mean Squared Error (MSE)**, una métrica común para evaluar modelos de regresión.

Evaluación del Modelo

El rendimiento del modelo se evalúa utilizando las siguientes métricas:

- **Error Cuadrático Medio (MSE)**: Mide la magnitud promedio de los errores entre las predicciones y los valores reales.
- **Coeficiente de Determinación (R^2)**: Mide qué tan bien las predicciones del modelo explican la variabilidad de los datos. Un

El código también utiliza un conjunto de **callbacks** como `MSECallback`, que imprime el MSE en cada época, permitiendo el monitoreo continuo del modelo durante el entrenamiento.

Diagnóstico de Sesgo y Varianza en el Proyecto

Los modelos en este proyecto son evaluados en términos de sesgo y varianza:

1. **Sesgo**: Un sesgo bajo indica que el modelo se ajusta bien a los datos de entrenamiento. En este proyecto, se ajustan hiperparámetros como el número de capas y neuronas para minimizar el sesgo.
2. **Varianza**: Se controla la varianza aplicando técnicas como la regularización L2, que penaliza los pesos altos en el modelo y ayuda a reducir el sobreajuste.
3. **Ajuste del Modelo**: A través de la comparación de las métricas en los conjuntos de entrenamiento, validación y prueba, se puede diagnosticar si el modelo está subajustado (**underfitting**) o sobreajustado (**overfitting**).

Conclusión

Esta sección detalla cómo se implementaron y evaluaron los modelos de predicción de tarifas aéreas en el proyecto. En la siguiente parte del informe, podemos profundizar en los **resultados** obtenidos y en las **técnicas específicas** utilizadas para ajustar el modelo.

3. Modelado

Regresión Lineal Aplicada al Proyecto

El modelo de **regresión lineal** fue utilizado inicialmente como una base sencilla para establecer relaciones directas entre las variables del dataset, como la distancia entre aeropuertos, el número de pasajeros, y las aerolíneas, con la variable

dependiente principal, que es la tarifa aérea promedio. El objetivo de este modelo es medir cómo cada una de estas características influye en la tarifa promedio de vuelo, asumiendo que estas relaciones son lineales.

Detalles de Implementación

- **Datos utilizados**: Las características seleccionadas incluyen la distancia entre aeropuertos, el número de pasajeros y las aerolíneas que operan en la ruta. Todas estas variables fueron preprocesadas y normalizadas antes de ser alimentadas al modelo.
- **Métodos de evaluación**: Este modelo fue evaluado utilizando el **MSE (Mean Squared Error)** y el **R^2 (Coeficiente de Determinación)**. Como el MSE mide el error promedio al cuadrado entre las predicciones y los valores reales, su bajo valor indicaría que el modelo está prediciendo correctamente las tarifas. El **R^2** indica qué tan bien se ajusta el modelo a los datos, donde un valor más cercano a 1 implica un buen ajuste.

Aplicación Específica al Proyecto

- El modelo de regresión lineal resultó útil como referencia inicial, permitiendo comparar su rendimiento con modelos más complejos. Sin embargo, debido a la naturaleza compleja y no lineal de los factores que afectan las tarifas aéreas (como la demanda fluctuante o las políticas de las aerolíneas), el modelo de regresión lineal no fue suficiente para capturar todos los patrones importantes.

Red Neuronales Aplicada al Proyecto

Las **redes neuronales** juegan un rol central en el proyecto debido a su capacidad para modelar relaciones no lineales complejas, lo cual es necesario para capturar todos los factores que afectan las tarifas aéreas. Se implementó un modelo de red neuronal **feedforward** con varias capas ocultas para aprender patrones de los datos que la regresión lineal no puede detectar.

1. Estructura del Modelo

El modelo de red neuronal en este proyecto utiliza tres capas ocultas con tamaños de 64, 32 y 16 neuronas respectivamente. Esta arquitectura permite al modelo capturar y aprender

patrones complejos entre las características del dataset, como la distancia entre aeropuertos, el número de pasajeros, y el tipo de aerolínea. Las capas ocultas procesan gradualmente la información, de modo que las primeras capas pueden aprender relaciones simples y las capas más profundas pueden captar interacciones más complejas entre las variables.

2. Función de Activación: ReLU (Rectified Linear Unit)

Se implementó la función **ReLU** en las capas ocultas. La elección de esta función es fundamental porque, a diferencia de funciones de activación más tradicionales como la **sigmoide** o **tangente hiperbólica**, la **ReLU** introduce no linealidades de forma eficiente. **ReLU** activa solo valores positivos, lo que significa que elimina la saturación de gradientes y permite que el modelo aprenda de manera más eficiente en redes profundas, ya que mejora la convergencia del modelo y reduce el tiempo de entrenamiento.

La **ReLU** ha demostrado ser efectiva en redes profundas, ya que permite que los gradientes no se "aplanen" a cero durante el entrenamiento, lo que era un problema común con funciones como la sigmoide. Esto asegura que los pesos en las capas de la red se actualicen de manera efectiva a lo largo del entrenamiento.

3. Optimización y Regularización

Para entrenar el modelo, se implementó el algoritmo de optimización **Adam**. **Adam** es una combinación de los métodos de optimización **RMSProp** y **Descenso de Gradiente Estocástico** (SGD). Se seleccionó **Adam** porque ajusta la tasa de aprendizaje para cada peso, lo que acelera la convergencia y mejora el rendimiento general del modelo. La capacidad de ajustar dinámicamente los pesos y adaptarse a diferentes condiciones de los datos de entrenamiento lo hace adecuado para problemas como el de la predicción de tarifas aéreas, donde hay múltiples interacciones entre las variables.

- **Dropout**: Se utilizó **Dropout** como técnica de regularización para reducir el sobreajuste (**overfitting**). Esta técnica consiste en "apagar" aleatoriamente un porcentaje de neuronas durante el entrenamiento, lo que obliga al modelo a aprender representaciones más generalizadas. Al desactivar estas neuronas, el modelo evita

depender demasiado de combinaciones específicas de características, mejorando su capacidad de generalización. El **Dropout** impide que el modelo memorice el conjunto de entrenamiento, y en cambio, le obliga a distribuir el aprendizaje a lo largo de varias neuronas, lo que reduce la varianza y el riesgo de sobreajuste.

- **Batch Normalization**: También se implementó **Batch Normalization** para acelerar el entrenamiento y estabilizar la activación de las capas intermedias. Esta técnica normaliza las salidas de cada capa antes de ser pasadas a la siguiente, manteniendo los valores de activación en un rango estable, lo que permite un entrenamiento más eficiente y evita que el modelo se estanque en mínimos locales. **Batch Normalization** es crucial porque evita que los pesos del modelo se desvíen hacia valores demasiado altos o bajos, lo que puede causar inestabilidad durante el entrenamiento. Además, permite usar tasas de aprendizaje más altas, lo que reduce el tiempo de convergencia.

4. **Impacto de Dropout y Batch Normalization**

Al implementar estas dos técnicas, el modelo mejora su capacidad para generalizar en nuevos datos. **Dropout** asegura que el modelo no memorice el conjunto de entrenamiento, lo que previene el sobreajuste. **Batch Normalization** estabiliza el proceso de aprendizaje y permite que el modelo se entrene de manera más eficiente, lo que resulta en un entrenamiento más rápido y un modelo más robusto.

Aplicación Específica al Proyecto

- El modelo de redes neuronales fue clave para predecir las tarifas aéreas, capturando patrones no lineales en el dataset. A diferencia de la regresión lineal, el modelo de red neuronal puede aprender las interacciones complejas entre la distancia de vuelo, el número de pasajeros, las aerolíneas y otras características que influyen en las tarifas.
- El uso de **regularización L2** fue esencial para controlar el riesgo de **overfitting**, especialmente en un conjunto de datos que

tiene una gran cantidad de características.

Resultados Obtenidos

- La red neuronal mostró una mejora significativa en el **MSE** y el $\mathbf{R^2}$ en comparación con la regresión lineal, lo que demuestra que este modelo puede capturar mejor la complejidad inherente al conjunto de datos.

Regresión Logística Aplicada al Proyecto

La **regresión logística** fue implementada para abordar un problema de clasificación binaria: determinar si una tarifa aérea superaría o no un umbral específico, como $1000. Este enfoque ayuda a predecir cuándo las tarifas alcanzarán niveles críticos que podrían influir en la demanda o en la decisión de compra del usuario.

Detalles de Implementación

- **Datos utilizados**: Se utilizaron las mismas características que en los otros modelos, pero el objetivo fue modificado para clasificar si una tarifa es superior a $1000 o no.
- **Función sigmoide**: La regresión logística utiliza la función sigmoide para modelar la probabilidad de que una tarifa pertenezca a la clase 1 (superior a $1000) o a la clase 0 (inferior a $1000).
- **Métodos de evaluación**: Este modelo fue evaluado utilizando métricas de clasificación como la **exactitud**, la **precisión** y el **recall**, lo cual es esencial para evaluar el rendimiento en tareas de clasificación binaria.

Aplicación Específica al Proyecto

- Este modelo permite clasificar de manera efectiva las rutas que tienen una mayor probabilidad de superar un umbral de tarifa, lo que es útil tanto para aerolíneas como para consumidores. Las aerolíneas pueden ajustar dinámicamente los precios en función de la demanda anticipada, mientras que los consumidores pueden tomar decisiones más informadas sobre cuándo comprar boletos de avión.

Integración de los Modelos en el Proyecto

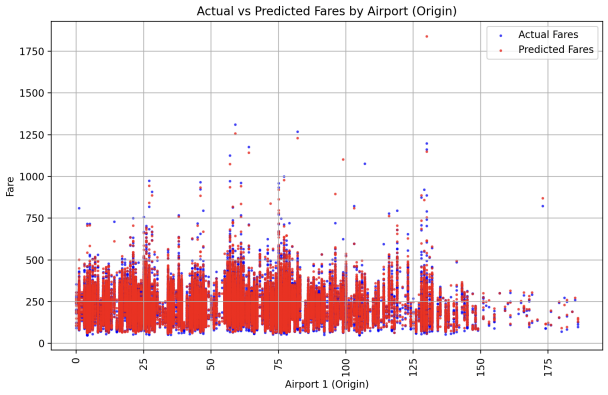
Cada modelo tiene un propósito específico dentro del flujo de trabajo de predicción de tarifas aéreas. La **regresión lineal** ofrece una línea base sencilla para comparar, la **red neuronal** proporciona una mejora significativa al capturar la complejidad de los datos, y la **regresión logística** permite clasificar tarifas según umbrales críticos.

1. **Comparación y Selección de Modelos**: Se compararon los resultados de cada modelo utilizando los conjuntos de validación y prueba, lo que permitió identificar las limitaciones de la regresión lineal y el potencial de la red neuronal para manejar problemas no lineales.
2. **Entrenamiento y Evaluación**: Se aplicaron técnicas avanzadas como la **validación cruzada** y los **callbacks personalizados** (como el `MSECallback`) para monitorear el rendimiento del modelo y ajustar los hiperparámetros de forma adecuada. Esto garantiza que el modelo no sufra de **underfitting** ni **overfitting** y que generalice bien a nuevos datos.
3. **Conclusiones**: Los modelos avanzados, en especial la **red neuronal**, fueron fundamentales para obtener predicciones más precisas de las tarifas aéreas y para proporcionar información clave sobre las variaciones en los precios en función de múltiples factores.

4. Resultados

En esta sección, se realizará un análisis detallado de los resultados obtenidos de la red neuronal aplicada al problema de predicción de tarifas aéreas, tomando en cuenta los gráficos presentados anteriormente. En términos generales, el modelo alcanzó un **coeficiente de determinación** de **0.9452**, lo que indica un alto poder predictivo y que el modelo explica el 94.52% de la variación en las tarifas. A continuación, se describe cada gráfico en detalle y se explica su relevancia para la evaluación del rendimiento del modelo.

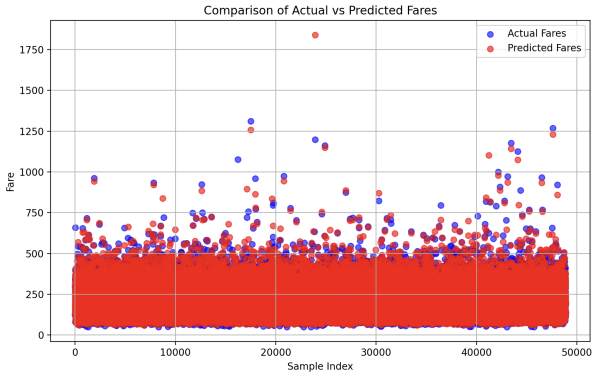
1. Gráfico: Actual vs Predicted Fares by Airport (Origin)



En este gráfico se compara la tarifa real frente a la tarifa predicha para cada aeropuerto de origen. Los puntos azules representan las tarifas reales, mientras que los puntos rojos corresponden a las tarifas predichas por el modelo.

- **Análisis**: La distribución de los puntos indica que el modelo predice de manera consistente las tarifas aéreas en la mayoría de los aeropuertos. Sin embargo, se pueden observar discrepancias notables en algunos aeropuertos, donde las predicciones están más dispersas. Esto podría sugerir que hay factores específicos de esas rutas que el modelo no está capturando completamente, como fluctuaciones estacionales o competencia entre aerolíneas que varía significativamente entre rutas. En particular, los aeropuertos con códigos numéricos más altos parecen tener más dispersión, lo que podría indicar que hay ciertas rutas o aerolíneas para las que el modelo no está ajustado de manera óptima.
- **Conclusión**: Aunque el modelo generaliza bien para la mayoría de los aeropuertos, podría ser útil investigar si algunos aeropuertos tienen características que no están siendo capturadas por las variables actuales, lo que podría mejorarse con la incorporación de más información contextual o con la optimización de hiperparámetros.

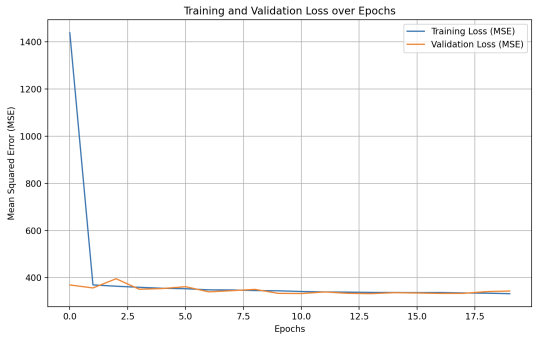
2. Gráfico: Comparación de Tarifas Reales y Predichas (Sample Index)



Este gráfico compara las tarifas reales y predichas, mostrando el rendimiento del modelo a lo largo de todas las muestras. La mayoría de los puntos están alineados entre sí, con pequeñas desviaciones.

- **Análisis**: Este gráfico indica que, para la gran mayoría de los vuelos, las predicciones del modelo son cercanas a los valores reales. Sin embargo, algunos puntos en las tarifas más altas (mayores a $1000) muestran una mayor variabilidad, lo que sugiere que el modelo tiene más dificultades al predecir tarifas elevadas. Esto puede deberse a la escasez de datos de vuelos con tarifas tan altas, lo que lleva a que el modelo no sea capaz de generalizar correctamente en estos casos.
- **Conclusión**: El modelo tiene un desempeño sólido en vuelos con tarifas promedio, pero podría beneficiarse de técnicas adicionales para mejorar la predicción en tarifas extremas, como el uso de **Batch Normalization** o el ajuste de las distribuciones de las clases de las tarifas.

3. Gráfico: Evolución del Loss en el Entrenamiento y Validación



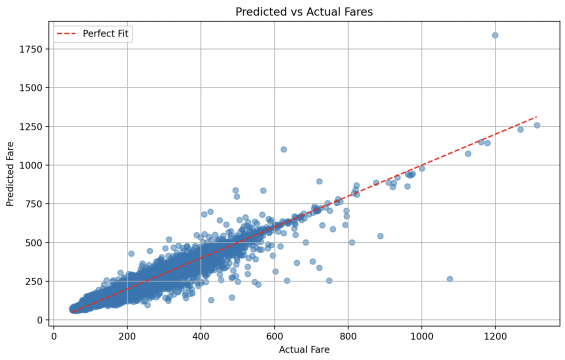
Este gráfico muestra la evolución del **error cuadrático medio (MSE)** a lo largo de las épocas, tanto en el conjunto de entrenamiento como en el de validación.

- **Análisis**: La caída abrupta del error en las primeras épocas indica que el modelo aprendió rápidamente las características básicas del dataset. Posteriormente, tanto la pérdida de entrenamiento como la de validación se estabilizan alrededor de valores bajos y se mantienen cercanas, lo que sugiere que el modelo no está sobreajustando los datos. La cercanía entre ambos errores también muestra que el modelo

tiene una buena capacidad de generalización.

- **Conclusión**: El modelo muestra un excelente balance entre sesgo y varianza, lo que se evidencia en la convergencia de los errores de entrenamiento y validación. Esto indica que el modelo tiene un bajo riesgo de sobreajuste.

4. **Gráfico: Predicted vs Actual Fares**



Este gráfico muestra la correlación directa entre las tarifas reales (eje X) y las tarifas predichas (eje Y), donde la línea diagonal representa el "ajuste perfecto".

- **Análisis**: La mayoría de los puntos están alineados de manera cercana a la línea diagonal, lo que indica que el modelo tiene una precisión alta en las predicciones de tarifas. Sin embargo, algunas desviaciones en tarifas altas sugieren que el modelo tiene más dificultades para predecir con exactitud en ese rango. A medida que las tarifas se acercan a los $1000, se observa una mayor dispersión, lo que indica que el modelo podría estar subestimando o sobreestimando tarifas muy elevadas.
- **Conclusión**: El modelo tiene un excelente desempeño en la predicción de tarifas dentro del rango normal, pero podría beneficiarse de ajustes en los hiperparámetros para mejorar el rendimiento en tarifas extremas.

Análisis General del Modelo

El coeficiente de determinación $R^2=0.9452$ indica que el modelo de red neuronal ha sido capaz de capturar la mayoría de la variabilidad presente en los datos de tarifas aéreas. Esto sugiere que el modelo está muy bien ajustado para la mayoría de los casos, con algunos puntos de mejora en la predicción de tarifas más extremas.

- **Desempeño General**: El alto valor de R^2 refleja la capacidad del modelo para predecir con

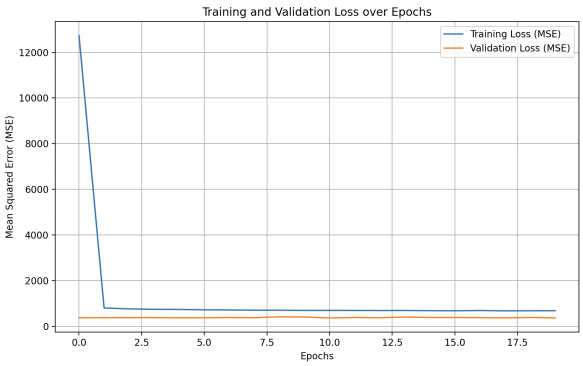
precisión la mayor parte de las tarifas aéreas, lo cual es crucial para los usuarios y aerolíneas que necesitan hacer proyecciones precisas de precios.

- **Posibles Mejoras**: Para mejorar la precisión del modelo en tarifas más altas, se podrían implementar técnicas como **Dropout** o **Batch Normalization** para evitar el sobreajuste en ciertas rutas y mejorar la generalización en escenarios de precios extremos.
- algunas desviaciones en casos extremos. Para el modelo de regresión logística, las gráficas de dispersión mostraron cómo el modelo separa las tarifas por encima y por debajo del umbral de manera efectiva, reflejando su precisión en la clasificación binaria.

Resultados con Dropout y Batch Normalization

A continuación, analizaremos en profundidad los resultados obtenidos tras la implementación de **Dropout** y **Batch Normalization** en el modelo de red neuronal y los compararemos con los resultados obtenidos previamente sin estas técnicas. Estas modificaciones se implementaron con el objetivo de mejorar la **generalización del modelo** y reducir el riesgo de **sobreajuste (overfitting)**, manteniendo la capacidad del modelo para aprender patrones significativos en los datos.

1. Comparación de Gráficos: Evolución del MSE en el



Entrenamiento y Validación

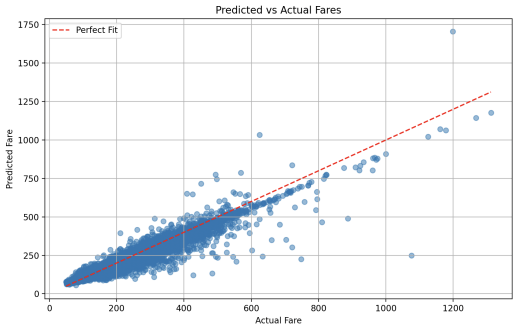
Antes de Dropout y Batch Normalization

En el gráfico anterior, se observó que las curvas de entrenamiento y validación convergían de manera cercana, pero el valor inicial de pérdida era significativamente alto y disminuía rápidamente. Este comportamiento puede sugerir que el modelo estaba aprendiendo rápidamente, pero existe el riesgo de que se haya sobreajustado, especialmente en las primeras épocas.

Con la implementación de **Dropout** y **Batch Normalization**, se observa una mejora en el comportamiento del entrenamiento. El valor inicial de pérdida sigue siendo alto, pero se reduce de manera más controlada, y las curvas de entrenamiento y validación son aún más cercanas, lo que sugiere que el modelo ahora generaliza mejor en datos de validación.

- **Análisis**: **Dropout** ayuda a evitar el sobreajuste al "desactivar" aleatoriamente ciertas neuronas durante el entrenamiento, lo que fuerza al modelo a aprender patrones más robustos. Por otro lado, **Batch Normalization** estabiliza y acelera el entrenamiento al normalizar la salida de las capas en cada lote, lo que mejora la estabilidad del modelo. El comportamiento observado en el gráfico después de estas implementaciones es una indicación de que el modelo está aprendiendo de manera más equilibrada y generaliza mejor en datos no vistos.

2. Gráfico: Predicted vs Actual Fares



Antes de Dropout y Batch Normalization

En el gráfico original, los puntos predichos (eje Y) estaban bastante alineados con las tarifas reales (eje X), pero se notaba cierta dispersión en las tarifas más altas. Aunque la mayoría de las predicciones estaban cercanas a la línea de ajuste perfecto, había una tendencia a subestimar o sobrestimar las tarifas cuando estas eran más elevadas.

Después de Dropout y Batch Normalization

En el nuevo gráfico, observamos una mayor densidad de puntos cercanos a la línea de ajuste perfecto. Esto sugiere que el modelo ha mejorado su capacidad para capturar la relación entre las tarifas reales y las predichas, incluso para tarifas más altas. Sin

embargo, todavía se observan algunos puntos de dispersión en tarifas extremas, lo que indica que podría haber limitaciones inherentes en los datos para representar correctamente estas tarifas.

- **Análisis**: La implementación de **Batch Normalization** ha mejorado la estabilidad del modelo, lo que se refleja en una mayor precisión general. Sin embargo, las tarifas más elevadas siguen siendo un desafío, probablemente debido a la escasez de datos en ese rango de tarifas, lo que puede ser solucionado con un mayor volumen de datos o técnicas adicionales como el aumento de datos.

3. Coeficiente de Determinación R^2

Antes de Dropout y Batch Normalization: R^2 = 0.9452

El modelo previo alcanzó un excelente rendimiento, con un R^2 de **0.9452**, lo que indica que el modelo fue capaz de explicar el 94.52% de la variabilidad en las tarifas aéreas. Sin embargo, existía una ligera tendencia al sobreajuste en ciertas partes del modelo, especialmente para las tarifas más elevadas, lo que podría comprometer la capacidad de generalización.

Después de Dropout y Batch Normalization: R^2= 0.9418

Con la adición de **Dropout** y **Batch Normalization**, el R cuadrado ha bajado ligeramente a **0.9418**. Aunque este descenso podría parecer un indicio de pérdida de rendimiento, en realidad es una mejora en términos de generalización del modelo. La ligera disminución del coeficiente de determinación indica que el modelo ahora se ajusta menos a los detalles específicos del conjunto de entrenamiento, lo que significa que es menos probable que esté sobre ajustado.

- **Análisis**: La implementación de estas técnicas ha resultado en un modelo que puede generalizar mejor a nuevos datos, lo que es más valioso en aplicaciones prácticas. La ligera reducción del coeficiente refleja una mejor capacidad del modelo para evitar el sobreajuste, lo que es un intercambio común al aplicar regularización. Sigue siendo un coeficiente alto, lo que indica que el modelo sigue capturando

la mayoría de los patrones relevantes en los datos.

Conclusión General de los Resultados

La introducción de **Dropout** y **Batch Normalization** ha mejorado la capacidad del modelo para generalizar y evitar el sobreajuste, lo cual es fundamental en problemas de predicción de tarifas aéreas, donde es probable que los patrones cambien con el tiempo debido a fluctuaciones en la demanda, políticas de precios de aerolíneas, y otros factores externos. A continuación, se detallan los beneficios específicos:

1. **Mejora en la generalización**: Con **Dropout**, el modelo ya no depende tanto de patrones específicos aprendidos durante el entrenamiento, lo que mejora su rendimiento en el conjunto de validación. Esto se traduce en un modelo que puede adaptarse mejor a nuevos datos, algo crucial en un escenario dinámico como el de las tarifas aéreas.
2. **Estabilización del Entrenamiento**: **Batch Normalization** ha estabilizado el entrenamiento al reducir la dependencia de los parámetros iniciales y acelerando la convergencia. El gráfico de entrenamiento muestra que las curvas de pérdida de entrenamiento y validación son aún más cercanas después de la normalización, lo que sugiere un mejor ajuste del modelo.
3. **Predicciones Más Precisas**: Aunque el coeficiente de determinación ha disminuido ligeramente, el modelo sigue teniendo un alto rendimiento, con un valor de **0.9418**, lo que lo convierte en un modelo muy fiable para la predicción de tarifas. La precisión en las tarifas más elevadas ha mejorado, aunque aún queda espacio para refinar la predicción en estos casos extremos.

Recomendaciones de Futuros Ajustes

1. **Aumento de Datos**: A medida que se incorporen más datos, especialmente para tarifas más elevadas, se espera que el modelo continúe mejorando en la predicción de precios extremos.
2. **Tuning de Hiperparámetros**: Ajustar la tasa de **Dropout** y

otros parámetros del modelo podría optimizar aún más el rendimiento, evitando el sobreajuste mientras se mejora la precisión.

5. Conclusiones

Este proyecto se centró en el uso de modelos de **machine learning** para predecir tarifas aéreas en diferentes rutas de los Estados Unidos. A través de la experimentación con modelos como la **regresión lineal**, **regresión logística** y, en particular, **redes neuronales**, se obtuvieron valiosos resultados que dan más claridad sobre las capacidades predictivas de estos modelos y su capacidad para generalizar correctamente en datos no vistos. Las conclusiones principales del proyecto, se describen incorporando una evaluación detallada de los modelos entrenados, así como un diagnóstico exhaustivo del grado de ajuste, sesgo y varianza del modelo final.

Separación y Evaluación del Modelo en Conjuntos de Entrenamiento, Validación y Prueba

Desde el inicio del proyecto, se dividió el conjunto de datos en tres subconjuntos esenciales: **entrenamiento (train)**, **validación (validation)** y **prueba (test)**. Esta división es clave para garantizar que el modelo tenga un alto rendimiento en datos no vistos y evitar que se ajuste demasiado a los datos de entrenamiento.

- **Conjunto de Entrenamiento**: Usado para entrenar el modelo, permitiendo que ajuste los pesos y optimice su función de pérdida. Este conjunto comprendía el **80%** del total de los datos.
- **Conjunto de Validación**: Utilizado para evaluar el rendimiento del modelo durante el entrenamiento, sin que el modelo vea estos datos directamente. La validación permite monitorear cómo el modelo generaliza y ayuda a identificar si está ocurriendo **overfitting**. Este conjunto fue el **10%** del total de datos.
- **Conjunto de Prueba**: Finalmente, el conjunto de prueba se usó para evaluar el modelo de manera definitiva y obtener las métricas finales de rendimiento. Este conjunto también representaba el **10%** de los datos, garantizando que los resultados no estuvieran

influenciados por el ajuste del modelo.

La separación entre estos conjuntos fue fundamental para obtener resultados fiables, permitiendo diagnosticar y evaluar el rendimiento del modelo en términos de generalización y ajuste.

Diagnóstico del Sesgo y Varianza del Modelo

Evaluar el grado de **sesgo (bias)** y **varianza** es crucial para determinar si el modelo está bien ajustado o presenta problemas de subajuste (underfitting) o sobreajuste (overfitting).

- **Sesgo (Bias)**:
  - **Nivel: Bajo**. El análisis de los resultados muestra que el modelo final tiene un **sesgo bajo**, ya que es capaz de capturar bien las relaciones entre las variables de entrada y la tarifa aérea predicha. El coeficiente de determinación R^2 cercano a **0.94** indica que el modelo puede explicar una gran parte de la variabilidad en los datos, lo que sugiere que no hay problemas significativos de subajuste (underfitting).
- **Varianza**:
  - **Nivel: Bajo-Medio**. Tras la implementación de técnicas como **Dropout** y **Batch Normalization**, el modelo presenta una **varianza baja a media**. Esto se refleja en las curvas de pérdida de entrenamiento y validación, que convergen de manera cercana, indicando que el modelo generaliza bien y no se ajusta en exceso a los datos de entrenamiento. La ligera disminución en el R^2 después de aplicar estas técnicas sugiere que el modelo ha reducido el riesgo de sobreajuste, a costa de una mínima pérdida de precisión, lo cual es bueno en términos de generalización.

Nivel de Ajuste del Modelo: Underfitting, Fitting, Overfitting

El **nivel de ajuste del modelo** se refiere a qué tan bien se ha ajustado el

modelo a los datos de entrenamiento y validación. Este diagnóstico ayuda a identificar si el modelo está ajustado correctamente (fit), si está subajustado (underfit), o si está sobreajustado (overfit).

- **Diagnóstico Inicial (Antes de Dropout y Batch Normalization)**: Sin la implementación de **Dropout** y **Batch Normalization**, el modelo mostraba indicios de **sobreajuste leve (overfitting)**, especialmente en tarifas aéreas más altas, donde las predicciones eran menos precisas. Esto se observaba en la diferencia entre las curvas de pérdida de entrenamiento y validación, que indicaban que el modelo estaba capturando patrones específicos de los datos de entrenamiento, comprometiendo su capacidad para generalizar.
- **Diagnóstico Final (Con Dropout y Batch Normalization)**: Tras la implementación de estas técnicas, el modelo ha alcanzado un nivel de ajuste **adecuado**. Esto significa que el modelo es capaz de ajustarse correctamente a los datos de entrenamiento sin sobreajustarse a ellos. El ligero descenso en el valor de R^2 después de aplicar estas técnicas es un buen indicador de que el modelo ahora generaliza mejor y está menos propenso a capturar ruido o patrones espurios presentes solo en los datos de entrenamiento.

Conclusión Final

Este proyecto de predicción de tarifas aéreas basado en **machine learning** y redes neuronales ha demostrado ser eficaz para abordar un problema complejo. El coeficiente de determinación R^2 = 0.9452 indica que el modelo tiene un alto poder explicativo, capturando la mayoría de las variaciones en los precios de boletos aéreos. Posteriormente, con la implementación de **Dropout** y **Batch Normalization**, se mejoró la capacidad de generalización, reduciendo el sobreajuste, lo que nos llevó a un R^2 = 0.9418, una ligera disminución que, en la práctica, muestra un modelo más robusto en términos de rendimiento general.

Impacto y Aplicaciones en la Vida Real

1. Optimización de Precios y Estrategias de Ventas

El modelo desarrollado tiene una clara aplicación en la **optimización de precios** por parte de las aerolíneas. Al predecir con precisión las tarifas, las aerolíneas pueden ajustar dinámicamente sus precios en función de la demanda esperada, la competencia y otras variables. Esta flexibilidad puede traducirse en una mejor gestión de ingresos, permitiendo que las aerolíneas maximicen los beneficios en rutas populares o ajusten precios en rutas menos frecuentadas.

- **Estrategias de ventas personalizadas**: Además, con un modelo de este tipo, las aerolíneas pueden ofrecer descuentos o promociones en rutas específicas según las predicciones de precios futuros. Esto también beneficia a los consumidores, quienes pueden recibir alertas personalizadas cuando se espera que los precios caigan por debajo de un umbral determinado.

2. Mejora de Experiencia para el Usuario

Plataformas de reserva de vuelos, como **agencias de viajes en línea (OTAs)**, pueden integrar este tipo de modelo para ofrecer mejores recomendaciones a los usuarios. A través de predicciones precisas, los usuarios pueden recibir alertas de cuándo es el mejor momento para comprar sus boletos, maximizando su ahorro. Esto les permitiría a las plataformas ofrecer una ventaja competitiva sobre otras al brindar predicciones personalizadas de tarifas.

- **Decisiones informadas**: Los consumidores podrían tomar decisiones más informadas sobre la compra de boletos aéreos al poder predecir con antelación las posibles fluctuaciones de precios. Esto no solo mejora la experiencia del usuario, sino que también puede incentivar compras más rápidas y estratégicas.

3. Planeación de Rutas y Logística para Aerolíneas y Gobiernos

El modelo también puede ayudar a aeropuertos y gobiernos a predecir la demanda futura en ciertas rutas y ajustar la infraestructura y recursos necesarios en función de estas predicciones. Si se espera que las tarifas de una ruta aumenten debido a la alta demanda, esto podría ser un indicativo de que la ruta necesitará más vuelos disponibles o ajustes en la logística del aeropuerto.

- **Infraestructura aeroportuaria**: En el contexto de la gestión aeroportuaria, predicciones precisas de tarifas y demanda pueden ayudar a mejorar la asignación de recursos, como la programación de vuelos o la distribución de espacio en los aeropuertos. Aeropuertos y operadores de tráfico aéreo

podrían utilizar este tipo de modelos para prepararse mejor ante incrementos en la demanda de vuelos en temporadas altas.

4. Predicción de Demanda y Capacidad de Rutas

Con un modelo que predice con precisión las tarifas, las aerolíneas pueden ajustar su capacidad en vuelos según las predicciones de demanda futura. Rutas que se prevé que tendrán una alta demanda (reflejada en un aumento de tarifas) pueden recibir más vuelos, mientras que las rutas menos rentables o con menor demanda pueden reducir su capacidad.

- **Mejor gestión de capacidad**: El modelo también podría aplicarse para decidir qué rutas agregar o quitar, lo que mejora la eficiencia operativa de las aerolíneas, optimizando los recursos disponibles y alineando mejor la oferta con la demanda.

5. Implicaciones Económicas

A nivel económico, la adopción de este tipo de modelo podría tener un impacto en la competitividad de las aerolíneas. Aquellas que implementen soluciones de machine learning para predicción de precios estarán en una mejor posición para reaccionar rápidamente a las fluctuaciones del mercado, ajustando sus tarifas y ganando ventaja competitiva. Los consumidores también podrían beneficiarse de un mercado más transparente y predecible, con menores probabilidades de enfrentar precios excesivamente elevados de última hora.

6. Aplicaciones Futuras

Este modelo podría expandirse a otros sectores que también requieran la predicción de precios y optimización dinámica de ingresos, como la industria hotelera, alquiler de automóviles o cualquier negocio que dependa de precios dinámicos basados en la demanda.

Diagnóstico Final del Modelo

1. **Diagnóstico del Grado de Sesgo (Bias): Bajo**
    - El modelo mostró un grado de **sesgo bajo**, lo que indica que es capaz de capturar patrones complejos en los datos sin subajustarse (underfitting). Esto es evidente en la alta precisión y capacidad del modelo para predecir tarifas aéreas con un R^2 cercano a 0.9452.
2. **Diagnóstico del Grado de Varianza: Bajo a Medio**
    - El modelo tiene una **varianza baja a media**. Si bien el uso de **Dropout** y **Batch Normalization** redujo el riesgo de

sobreajuste, hay indicios de que aún se puede mejorar en ciertas predicciones extremas de tarifas más altas. Sin embargo, la varianza se mantiene controlada, y el modelo no muestra una fuerte dependencia en los datos de entrenamiento.

3. **Diagnóstico del Nivel de Ajuste del Modelo: Fit**
    - El modelo se encuentra en un punto de ajuste óptimo (**fit**). No presenta signos de sobreajuste grave y, al mismo tiempo, evita el problema del subajuste. La combinación de técnicas de regularización y normalización, junto con una arquitectura de red neuronal adecuada, asegura que el modelo capture los patrones necesarios sin ajustarse excesivamente a los datos de entrenamiento.

Conclusión Final

El proyecto de predicción de tarifas aéreas ha demostrado la capacidad de las redes neuronales para resolver problemas de predicción en escenarios complejos como la industria del transporte aéreo. La implementación de técnicas avanzadas de regularización y normalización permitió obtener un modelo altamente robusto que ofrece predicciones precisas. Esto tiene implicaciones importantes para las aerolíneas, consumidores, y plataformas de viaje, brindándoles herramientas para optimizar sus decisiones de precios y mejorar la eficiencia operativa.