



Universidad del
Rosario

Escuela de Ingeniería,
Ciencia y Tecnología

ANÁLISIS DEL TRÁFICO AÉREO MEDIANTE GRAFOS

Presentado para obtener el título de

MAGÍSTER EN MATEMÁTICAS APLICADAS Y CIENCIAS DE LA COMPUTACIÓN

Diego Alberto Rodríguez Cruz

Dirección:

Juan Felipe Romero Ramírez

Universidad del Rosario

Escuela de Ingeniería, Ciencia y Tecnología

Maestría en Matemáticas Aplicadas y Ciencias de la Computación

DEDICATORIA

Madre, por todo aquello que nos alcanzaste a enseñar antes de tu partida.

Hermanos, por todo el tiempo compartido y apoyo que me brindan.

Padre, por el esfuerzo y esmero para salir adelante día a día.

¡Gracias por todas sus enseñanzas y consejos!

AGRADECIMIENTOS

Agradecer al director de la tesis, Juan Felipe Romero, con su experiencia y conocimientos me ha ayudado adentrarme en un tema en el cual he querido aprender desde hace tiempo y el cual me ha brindado las herramientas necesarias para explorar más en el día a día.

Así mismo, agradecimientos a los compañeros de maestría quienes con su experiencia en diferentes áreas apoyaron el desarrollo de las clases y aportaron su granito de arena en el desarrollo de estas, y un especial agradecimiento a los compañeros de clase Héctor y Jorge.

ABSTRACT

Español

La planificación de vuelos es crucial para las aerolíneas, ya que afecta directamente su oferta y, por lo tanto, la demanda de pasajeros, lo que influye en sus ganancias. Además, la representación efectiva de las rutas y el flujo de pasajeros es esencial para tomar decisiones informadas en una aerolínea. Este estudio utiliza herramientas de análisis de redes, como grafos, para examinar el tráfico aéreo y representar la información de manera clara.

Es importante destacar que la planificación de la demanda también involucra aspectos relacionados con la fijación de precios, especialmente en el contexto de las aerolíneas que operan en un entorno de que se considera oligopolista. Este enfoque en la planificación de la demanda y la estrategia de precios es esencial para optimizar el desempeño de las aerolíneas en mercados con limitada competencia.

El objetivo principal de este trabajo es utilizar grafos para visualizar y analizar la demanda de pasajeros entre ciudades, utilizando la información de la Aerocivil. Estas visualizaciones y análisis ayudarán a las aerolíneas a tomar decisiones más efectivas en la planificación de sus rutas y la gestión de la demanda. Este enfoque tiene un alto valor estratégico y puede contribuir significativamente a la eficiencia y rentabilidad de las aerolíneas en dicho contexto.

English

The flight planning is crucial for airlines, as it directly affects their supply and, therefore, passenger demand, which influences their profits. In addition, an effective representation of routes and passenger flow is essential to make informed decisions in an airline. This study uses network analysis tools, such as graphs, to examine air traffic and represent the information clearly.

It is important to note that demand planning also involves pricing issues, especially in the context of airlines operating in what is considered an oligopolistic environment. This focus on demand planning and pricing strategy is essential to optimize airline performance in markets with limited competition.

The main objective of this work is to use graphs to visualize and analyze inter-city passenger demand using Aerocivil information. These visualizations and analyses will help airlines to make more effective decisions in route planning and demand management. This approach has a high strategic value and can contribute significantly to airline efficiency and profitability in this context.

TABLA DE CONTENIDO

INTRODUCCIÓN	1
JUSTIFICACIÓN Y DESCRIPCIÓN DEL PROBLEMA.....	2
OBJETIVOS	3
Objetivo General.....	3
Objetivos Específicos:	3
MARCO TEÓRICO.....	4
Grafo	4
Tipos de grafos.....	4
Grafo dirigido.....	4
Grafo no dirigido.....	5
Multígrafo	6
Grafo ponderado	6
Representación de grafos	7
Matriz de adyacencia	7
Lista de aristas.....	7
Métricas de un grafo	8
Longitud de ruta.....	8
Eficiencia global	8
Coeficiente de agrupación.....	8
Aprendizaje Automático	9
Aprendizaje Automático de Grafos (GML).....	9
Representation Learning (network embedding).....	10
Embeddings.....	11
Node2Vec	12
Edge2Vec	12
Algoritmo de similitud del Coseno.....	13
Regresión Lineal	13
Ridge.....	14
Lasso	15
Elastic Net.....	15

ESTADO DEL ARTE.....	17
Aplicación de la teoría de grafos en el diseño de rutas de transporte desde las zonas de producción agrícola hasta la planta de procesamiento.....	17
Aproximación al problema de ruta más corta con trasbordos.....	19
Representación de grafos para la cuantificación de la percepción urbana espacial.....	20
METODOLOGÍA.....	21
Adquisición de Datos.....	21
EDA - Análisis exploratorio de datos.....	22
Preprocesamiento.....	24
Procesamiento de información.....	24
Construcción de Grafo.....	25
RESULTADOS Y DISCUSIÓN.....	27
Grafo.....	27
Embedding.....	28
Similitud.....	30
Modelo.....	31
Representación geográfica.....	33
Ruta no establecida.....	34
CONCLUSIONES.....	40
REFERENCIAS.....	42
APÉNDICE.....	44
Análisis de datos.....	44
Base de Datos Origen – Destino.....	44
Maestra aeródromo.....	45
Aeropuertos del mundo.....	46
Referencias de Aeropuertos de Colombia.....	48
Rutas de prueba.....	53

LISTA DE TABLAS

Tabla 1. Adquisición de datos.....	22
Tabla 2. Conjunto de datos origen destino.....	22
Tabla 3. Maestra aeródromo	23
Tabla 4. Aeropuertos del mundo.....	23
Tabla 5. Resultado de métricas.	27
Tabla 6. Similitud, rutas con mayor demanda de pasajeros.....	30
Tabla 7. Métricas obtenidas del modelo.	31
Tabla 8. Validación cruzada	32
Tabla 9. Valores obtenidos en la evaluación de la nueva ruta.	36
Tabla 10. Evaluación de rutas.	38
Tabla 11. Tabla de aeropuertos de Colombia.	48
Tabla 12. Rutas para prueba de modelos.	53

LISTA DE FIGURAS

Figura 1. Ejemplo de grafo. [2].....	4
Figura 2. Grafo dirigido [4]	5
Figura 3. Grafo no dirigido [4]	5
Figura 4. Multígrafo [4]	6
Figura 5. Grafo ponderado. [4]	6
Figura 6. Ejemplo matriz de adyacencia.....	7
Figura 7. Ejemplo de representación de un grafo y su lista de aristas. [5]	7
Figura 8. Ejemplo de Network Embedding. [5].....	10
Figura 9. Embedding. [10]	11
Figura 10. Aplicación de Node2Vec a un grafo (izquierda), representación de embeddings del grafo [10]	12
Figura 11. Matriz de distancias. Problema de rutas con vehículos capacitados (CVRP). [14]	19
Figura 12. Relación de fuentes. Elaboración propia.....	25
Figura 13. Construcción de grafo. Elaboración propia.....	26
Figura 14. Representación visual del grafo.....	26
Figura 15. (Izquierda) Grafo General. (Derecha) Grafo con zoom a los nodos.	27
Figura 16. Representación de resultado Node2Vec del grafo.....	29
Figura 17. Representación geográfica, (izquierda) ubicación de nodos, (derecha) Conexión entre nodos con tamaño de acuerdo con la cantidad de pasajeros.	34
Figura 18. Ruta nueva.....	35
Figura 19. Grafo de ruta nueva.....	35
Figura 20. Embedding de ruta nueva.....	35
Figura 21. Representación geográfica de la ruta nueva.....	36
Figura 22. Grafo de rutas no establecidas.....	37
Figura 23. Representación geográfica del grafo de prueba.....	38
Figura 24. Embedding de grafo de prueba.....	38
Figura 25. Descripción estadística de Origen – Destino.....	44
Figura 26. Tipos de dato (Origen – Destino)	44
Figura 27. Visualización de nulos (Origen – Destino)	45
Figura 28. Descripción estadística maestra aeródromo	45
Figura 29. Tipos de datos (Maestra Aeródromo).....	46
Figura 30. Visualización de nulos (Maestra Aeródromo).....	46
Figura 31. Descripción estadística de Aeropuertos del mundo.....	46
Figura 32. Tipos de datos (Aeropuertos del mundo)	47
Figura 33. Visualización de nulos (Aeropuertos del mundo)	47
Figura 34. Visualización de nulos a nivel país (Colombia).....	47

ABREVIATURAS

ACID: Atomicity Consistency Isolation Durability. Atomicidad Consistencia Aislamiento Durabilidad.

BD: Base de Datos.

EDA: Exploratory Data Analysis. Análisis Exploratorio de Datos

GML: Graph Machine Learning. Aprendizaje Automático de Grafos

IATA: International Air Transport Association. Asociación Internacional de Transporte Aéreo.

ICAO: International Civil Aviation Organization. Organización Internacional de Aviación Civil.

JSON: JavaScript Object Notation.

MAE: Mean Absolute Error. Error Absoluto Medio.

MCO: Mínimos Cuadrados Ordinarios, en inglés *OLS* (Ordinary Least Squares).

MSE: Mean Square Error. Error Cuadrático Medio.

NoSQL: Not Only SQL. No solo SQL.

INTRODUCCIÓN

Hoy en día, la eficiencia en la planificación de rutas aéreas desempeña un papel fundamental en el éxito de la operación de las aerolíneas; la demanda de pasajeros en rutas existentes como en aquellas aún no exploradas, constituye un elemento crucial que impacta directamente en la toma de decisiones estratégicas y en la identificación de oportunidades de mercado.

La presente tesis tiene como objetivo principal generar un modelo innovador basado en grafos para la planificación de la demanda de pasajeros en rutas aéreas dentro de Colombia. En la cual se busca identificar oportunidades de mercado, proporcionando a las aerolíneas herramientas precisas y avanzadas para anticipar y satisfacer las necesidades de los pasajeros en un entorno aéreo dinámico.

Se recopila información investigando páginas del estado con información de rutas aéreas, con el fin de generar el punto de partida para la generación del grafo.

Con la construcción del grafo, se exploran herramientas de aprendizaje automático que darán las bases para generar un modelo predictivo para la planificación de la demanda de pasajeros.

Por último, se visualizará el grafo generado en un mapa donde se evidenciará la afluencia de las rutas con mayor afluencia de pasajeros.

JUSTIFICACIÓN Y DESCRIPCIÓN DEL PROBLEMA

Hoy en día en la industria aérea de transporte de pasajeros, se evidencia una necesidad de generar más rentabilidad y ofrecer las mejores opciones en rutas aéreas a los clientes; al tener en muchos casos una mala planificación de las rutas y gestión de la demanda de pasajeros conlleva a lo que se conoce como “reventa” de asientos y, por ende, en generar una crisis en el transporte aéreo. En base a lo anterior, a pesar de que existen herramientas para seguimiento de vuelos no se ve aplicada un modelo que genere una proyección de demanda de pasajeros y así una mejor planificación de rutas aéreas.

Es con ello, que mediante el uso de los grafos se pueden descubrir aquellos patrones en el comportamiento de viajes de los pasajeros y así proyectar las rutas de más demanda de estos.

¿Cómo a través del análisis de las rutas aéreas se puede ayudar en la planificación de la demanda de viajes?

Analizar la relación entre las rutas aéreas y demanda de viajes, utilizando herramientas como grafos, ayudan en el entendimiento de relaciones y/o conexiones entre los datos; dado que un grafo representa la relación entre sus nodos, en este caso se pueden evidenciar patrones entre los elementos de este.

Se puede representar las ciudades de origen/destino y la demanda de pasajeros de una ciudad a otra como relaciones entre los nodos.

Se han de recopilar la información presente en la [Aerocivil](#) referente al tráfico aéreo, con el fin de identificar los patrones y tendencias del tráfico aéreo, así como evaluar la frecuencia de viajes entre ciudades.

Se espera contribuir en el desarrollo o planificación de viajes aéreos con base a la información recopiladas.

OBJETIVOS

Objetivo General

Generar un modelo basado en grafos para la planificación de la demanda de pasajeros en rutas aéreas dentro de Colombia, que no solo analice el tráfico existente, sino que también tenga la capacidad de estimar la demanda en rutas no existentes, identificando así oportunidades de mercado.

Objetivos Específicos:

1. Recopilar información sobre el origen, destino y la cantidad de pasajeros en rutas existentes y potenciales.
2. Utilizar “Machine Learning para grafos” con el fin de visualizar y analizar las relaciones entre los nodos de las rutas aéreas, permitiendo una comprensión de la estructura del tráfico aéreo en Colombia.
3. Desarrollar un modelo predictivo que utilice la información extraída de la Aerocivil para estimar la demanda de pasajeros en rutas futuras.

MARCO TEÓRICO

Grafo

Un grafo G consiste en un conjunto de vértices $V(G)$ y un conjunto de aristas $E(G)$, donde cada arista es un par $\{u, v\} \in V(G)$. [1]



Figura 1. Ejemplo de grafo. [2]

Los grafos son uno de los temas unificadores de la informática: una representación abstracta que describe la organización de los sistemas de transporte, las interacciones humanas y las redes de telecomunicaciones. El hecho de que tantas estructuras diferentes puedan modelarse utilizando un único formalismo es una fuente de gran poder para un programador. [3]

Tipos de grafos

Grafo dirigido

Sea V un conjunto finito no vacío, y sea la relación binaria $E \subseteq V \times V$. El par ordenado (V, E) es un grafo dirigido sobre V , o dígrafo, donde V es el conjunto de vértices o nodos y E es su conjunto de aristas, $G = (V, E)$.

A continuación, se representa un grafo dirigido o dígrafo, con vértices $V = \{A, B, C\}$ y aristas $E = \{(B, A), (A, C), (C, A), (C, B)\}$. [4]

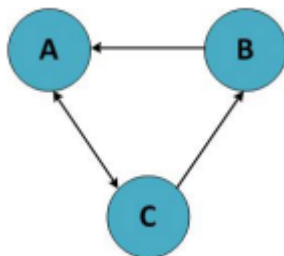


Figura 2. Grafo dirigido [4]

La dirección de una arista se indica al colocar una flecha dirigida sobre ella como se muestra en Figura 2. Grafo dirigido para cualquier arista; por ejemplo (B, A) , el vértice B es origen o fuente, mientras que el vértice A es el termino o vértice terminal. En el caso de tener una flecha en los dos sentidos, se dice que el vértice A es origen de vértice C y al mismo tiempo el vértice C es origen de A. [4]

Grafo no dirigido

Cuando no importa la dirección de las aristas, la estructura $G = (V, E)$, donde E es ahora un conjunto de pares no ordenados sobre V , es decir el conjunto de aristas representa una relación simétrica binaria, donde si V_j y V_k son vértices cualesquiera del conjunto de vértices V de un grafo, $(V_j, V_k) \in E \rightarrow (V_k, V_j) \in E$, corresponde a un grafo no dirigido.

En Figura 3. Grafo no dirigido se puede ver cómo se representan los grafos no dirigidos, con vértices $V = \{A, B, C, D\}$ y aristas $E = \{(A, B), (B, C), (C, D), (D, A)\}$

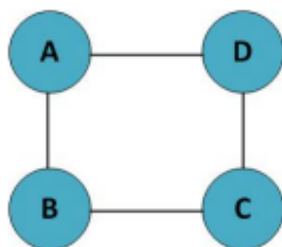


Figura 3. Grafo no dirigido [4]

Multígrafo

Un multígrafo G se define como $G = (V, E)$, donde V es un conjunto de nodos y E es un multiconjunto (un conjunto que permite múltiples instancias para cada uno de sus elementos) de aristas.

Un multígrafo se denomina multígrafo dirigido si E es un multiconjunto de pares ordenados; en caso contrario, si E es un multiconjunto de dos conjuntos, se denomina multígrafo no dirigido. [5]

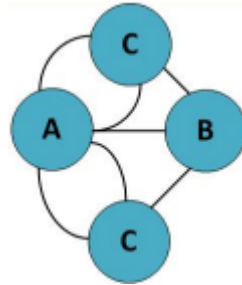


Figura 4. Multígrafo [4]

Grafo ponderado

Un grafo ponderado G se define como $G = (V, E, w)$ donde V es un conjunto de nodos, E es un conjunto de aristas, y $w: E \rightarrow \mathbb{R}$ es la función ponderada que asigna a cada arista $e \in E$ un peso expresado como un número real.

Un grafo G ponderado por nodos se define como $G = (V, E, w)$, donde V es un conjunto de nodos, E es un conjunto de aristas, y $w: V \rightarrow \mathbb{R}$ es la función ponderada que asigna a cada nodo $v \in V$ un peso expresado como un número real. [5]

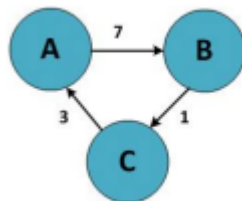


Figura 5. Grafo ponderado. [4]

Representación de grafos

Matriz de adyacencia

Es una matriz cuadrada de dimensión $n \times n$, donde n representa el número de nodos del grafo. [6]

Dicha matriz se compone de 1 y 0, sus elementos a_{ij} de la matriz (G) está dados por,

$$a_{ij} = \begin{cases} 1 & \text{si } (i,j) \in (G) \\ 0 & \text{de otra forma} \end{cases}$$

	0	1	2	3	4	5	6	7	8	9
0	0	1	0	0	0	0	1	0	1	0
1	1	0	0	0	1	0	1	0	0	1
2	0	0	0	0	1	0	1	0	0	0
3	0	0	0	0	1	1	0	0	1	0
4	0	1	1	1	0	1	0	0	0	1
5	0	0	0	1	1	0	0	0	0	0
6	1	1	1	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	1	1
8	1	0	0	1	0	0	0	1	0	0
9	0	1	0	0	1	0	0	1	0	0

Figura 6. Ejemplo matriz de adyacencia.

Lista de aristas

La lista de aristas L de un grafo $G = (V, E)$ es una lista de tamaño $|E|$ de tipo matriz tal que su elemento L_i es un par que representa la cola y el nodo final de la arista i . [5]

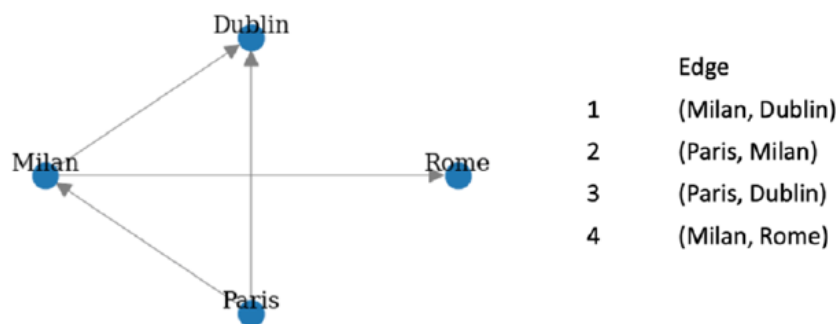


Figura 7. Ejemplo de representación de un grafo y su lista de aristas. [5]

Métricas de un grafo

Longitud de ruta

La longitud de ruta característica se define como la media de todas las longitudes de camino más cortas entre todos los pares de nodos posibles. Si l_i es la longitud media del camino entre el nodo y todos los demás nodos, la longitud de camino característica se calcula de la siguiente manera:

$$\frac{1}{q(q-1)} \sum_{i \in V} l_i$$

Donde, V es el conjunto de nodos en el grafo y $q = |V|$ representa el orden. Esta es una de las métricas más utilizadas para medir la eficacia con la que se difunde la información a través de una red. Las redes con longitudes de ruta características más cortas favorecen la transferencia rápida de información y disminuyen los costos. Sin embargo, esta métrica no siempre se puede definir ya que en algunos casos no es posible calcular la ruta entre todos los nodos en un grafo desconectado. Por esta razón, se suele usar la eficiencia de la red como métrica. [5]

Eficiencia global

La eficiencia global es la media de la longitud inversa del camino más corto para todos los pares de nodos. Esta métrica puede considerarse una medida de la eficacia con la que se intercambia información a través de una red. Consideremos que l es el camino más corto entre un nodo i y un nodo j . La eficiencia de la red se define del siguiente modo:

$$\frac{1}{q(q-1)} \sum_{i \in V} \frac{1}{l_{ij}}$$

La eficiencia es máxima cuando un grafo está totalmente conectado. [5]

Coefficiente de agrupación

Corresponde al grado de agrupación de los nodos, es decir, corresponde a la cantidad de vecinos del nodo que se relacionan entre sí.

Aprendizaje Automático

El aprendizaje automático (Machine Learning en inglés), es un conjunto de técnicas que hacen parte de la inteligencia artificial, que basadas en algoritmos buscan el aprendizaje dentro de grandes conjuntos de datos. Una característica muy importante de estos algoritmos es la predicción de nuevos casos basándose en la experiencia aprendida del conjunto de datos utilizados para su entrenamiento, a esto se le conoce en la literatura como generalización.

El aprendizaje automático se puede dividir en tres categorías

- **Aprendizaje supervisado:** estos algoritmos cuentan con un aprendizaje previo basado en un sistema de etiquetas asociadas a unos datos que les permiten tomar decisiones o hacer predicciones. Un ejemplo es un detector de spam que etiqueta un e-mail como spam o no dependiendo de los patrones que ha aprendido del histórico de correos (remitente, relación texto/imágenes, palabras clave en el asunto, etc.).
- **Aprendizaje no supervisado:** estos algoritmos no cuentan con un conocimiento previo. Se enfrentan al caos de datos con el objetivo de encontrar patrones que permitan organizarlos de alguna manera. Por ejemplo, en el campo del marketing se utilizan para extraer patrones de datos masivos provenientes de las redes sociales y crear campañas de publicidad altamente segmentadas.
- **Aprendizaje por refuerzo:** su objetivo es que un algoritmo aprenda a partir de la propia experiencia. Esto es, que sea capaz de tomar la mejor decisión ante diferentes situaciones de acuerdo con un proceso de prueba y error en el que se recompensan las decisiones correctas. En la actualidad se está utilizando para posibilitar el reconocimiento facial, hacer diagnósticos médicos o clasificar secuencias de ADN. [7]

Aprendizaje Automático de Grafos (GML)

Es la aplicación del aprendizaje automático (Machine Learning) a grafos específicamente para tareas predictivas y prescriptivas. GML tiene una variedad de casos de uso en la cadena de suministro, detección de fraudes, recomendaciones, Customer 360, descubrimiento de medicamentos y más. [8]

Representation Learning (network embedding)

Es la tarea que tiene como objetivo aprender una función de mapeo función, $f: G \rightarrow \mathbb{R}^n$, de un grafo discreto a un dominio continuo. La función será capaz de realizar una representación vectorial de baja dimensión tal que las propiedades (locales y globales) del grafo G sean preservadas.

Una vez aprendido el mapeo f , podría aplicarse al grafo y el mapeo resultante podría utilizarse como conjunto de características para un algoritmo de aprendizaje automático.

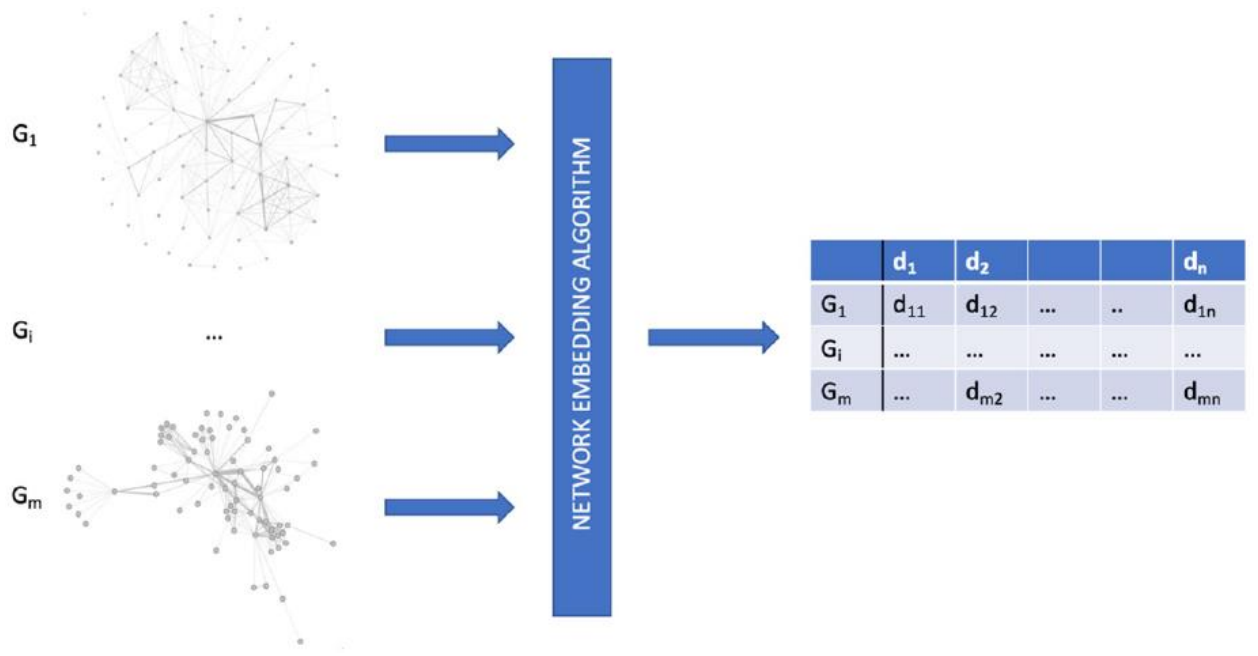


Figura 8. Ejemplo de Network Embedding. [5]

La función de mapeo f también puede aplicarse para aprender la representación vectorial de nodos $f: V \rightarrow \mathbb{R}^n$ conocida como *node embedding* o aristas $f: E \rightarrow \mathbb{R}^n$ conocida como *edge embedding*. Esas funciones de mapeo tratan de construir un espacio vectorial tal que las relaciones geométricas en el nuevo espacio reflejen la estructura del grafo original, nodos o aristas originales.

[5]

Embeddings

En un grafo, un *embedding* determina una representación vectorial de longitud fija para cada entidad (generalmente nodos) en el grafo. Estos *embeddings* son una representación de menor dimensión del grafo y conservan la topología del grafo. [9]

El objetivo es codificar los nodos como vectores de baja dimensión que resumen su posición en el grafo y la estructura de su vecindario local. Estas incrustaciones de baja dimensión pueden verse como la codificación, o proyección, de nodos en un espacio latente, donde las relaciones geométricas en este espacio latente corresponden a interacciones (por ejemplo, aristas) en el grafo original. [10]

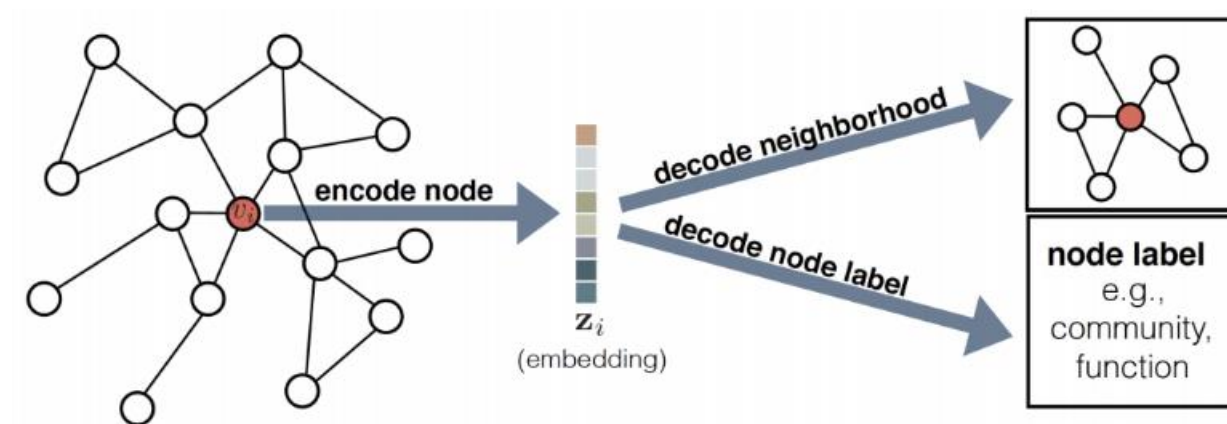


Figura 9. Embedding. [10]

En la Figura 9. Embedding. Se representa una visión general del enfoque codificador-decodificador correspondiente a los *embedding*. En primer lugar, el codificador (ENC) asigna el nodo, v_i , a un vector de baja dimensión z_i , basado en la posición del nodo en el grafo, su estructura de vecindad local y/o sus atributos. A continuación, el decodificador (DEC) extrae la información especificada por el usuario del *embedding*, que puede ser información sobre el vecindario local de v_i (por ejemplo, la identidad de sus vecinos) o una etiqueta de clasificación asociada a v_i (por ejemplo, una etiqueta de comunidad). Mediante la optimización conjunta del codificador y el

decodificador, el sistema aprende a comprimir la información sobre la estructura del grafo en el espacio del *embedding*. [10]

Node2Vec

Es un algoritmo de *embedding* de nodos que calcula una representación vectorial de un nodo basado en recorridos aleatorios en el grafo. Los nodos vecinos se muestrean a través de caminatas aleatorias. Utilizando una serie de muestras de vecindad aleatorias, el algoritmo entrena una única red neuronal de capa oculta. La red neuronal se entrena para predecir la probabilidad de que un nodo ocurra en un paseo en función de la aparición de otro nodo. [9]

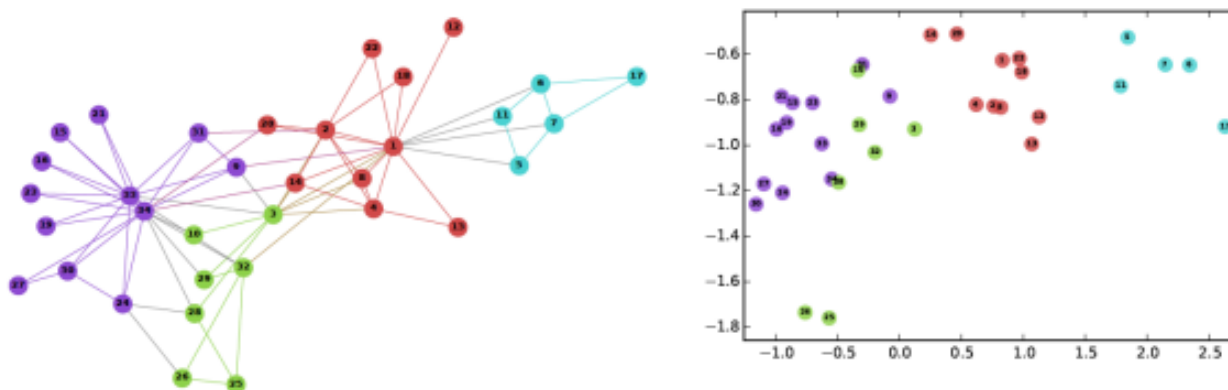


Figura 10. Aplicación de Node2Vec a un grafo (izquierda), representación de embeddings del grafo [10]

Edge2Vec

El algoritmo Edge to Vector (Edge2Vec) genera el espacio de incrustación en las aristas, en lugar de en los nodos. Este algoritmo es un simple efecto secundario de la incrustación generada al utilizar Node2Vec. La idea principal es utilizar el *embedding* de dos nodos adyacentes para realizar algunas operaciones matemáticas básicas con el fin de extraer la incrustación de los nodos adyacentes. para extraer la incrustación de la arista que los conecta. [5]

Algoritmo de similitud del Coseno

Se encarga de conocer el ángulo entre dos vectores n-dimensionales en un espacio n-dimensional. Esto significa que el resultado que obtenemos es el producto escalar de los dos vectores dividido por el producto de las longitudes o magnitudes de los dos vectores.

Se calcula entre dos vectores distintos de cero dentro del espacio interno del producto que mide el coseno del ángulo entre ellos. El coseno de 0° es 1, y es menor que uno para cualquier ángulo que se encuentre en el intervalo $(0, \pi]$ radianes. Así que este caso se trata de un cálculo que da origen a un juicio de orientación y no de magnitud.

Dentro de la similitud de coseno dos vectores con la misma orientación tienen una similitud de coseno de 1, dos vectores orientados a 90° entre sí tienen una similitud de 0, y dos vectores diametralmente opuestos tienen una similitud de -1, independientemente de su magnitud. [11]

$$\vec{u} \cdot \vec{v} = |u||v| \cos \theta$$

$$\frac{\vec{u} \cdot \vec{v}}{|u||v|} = \text{similitud}$$

Regresión Lineal

La regresión lineal es una técnica de modelado estadístico que se emplea para describir una variable de respuesta continua como una función de una o varias variables predictoras. Puede ayudar a comprender y predecir el comportamiento de sistemas complejos.

Las técnicas de regresión lineal permiten crear un modelo lineal. Este modelo describe la relación entre una variable dependiente y (también conocida como la respuesta) como una función de una o varias variables independientes X_i (denominadas predictores). La ecuación general correspondiente a un modelo de regresión lineal es:

$$Y = \beta + \sum \beta_k X_k + \epsilon_i$$

Donde β representa las estimaciones de parámetros lineales que se deben calcular y ϵ representa los términos del error.

La regresión lineal cuenta con ciertas características ideales para las siguientes aplicaciones:

- Predicción o pronóstico: utilice un modelo de regresión para crear un modelo de pronóstico para un conjunto de datos específico. A partir de la moda, puede usar la regresión para predecir valores de respuesta donde solo se conocen los predictores.
- Fuerza de la regresión: utilice un modelo de regresión para determinar si existe una relación entre una variable y un predictor, y cuán estrecha es esta relación. [12]

Ridge

La regularización Ridge penaliza la suma de los coeficientes elevados al cuadrado ($\|\beta\|_2^2 = \sum_{j=1}^p \beta_j^2$). A esta penalización se le conoce como l_2 y tiene el efecto de reducir de forma proporcional el valor de todos los coeficientes del modelo, pero sin que estos lleguen a cero. El grado de penalización está controlado por el hiper parámetro λ . Cuando $\lambda = 0$, la penalización es nula y el resultado es equivalente al de un modelo lineal por mínimos cuadrados ordinarios (OLS). A medida que λ aumenta, mayor es la penalización y menor el valor de los predictores.

$$\sum_{i=1}^n (y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij})^2 + \lambda \sum_{j=1}^p \beta_j^2 = \text{suma residuos cuadrados} + \lambda \sum_{j=1}^p \beta_j^2$$

La principal ventaja de aplicar Ridge frente al ajuste por mínimos cuadrados ordinarios (OLS) es la reducción de varianza. Por lo general, en situaciones en las que la relación entre la variable respuesta y los predictores es aproximadamente lineal, las estimaciones por mínimos cuadrados tienen poco *bias*, pero aún pueden sufrir alta varianza (pequeños cambios en los datos de entrenamiento tienen mucho impacto en el modelo resultante). Este problema se acentúa conforme el número de predictores introducido en el modelo se aproxima al número de observaciones de entrenamiento, llegando al punto en que, si $p > n$, no es posible ajustar el modelo por mínimos cuadrados ordinarios. Empleando un valor adecuado de λ , el método de Ridge es capaz de reducir varianza sin apenas aumentar el *bias*, consiguiendo así un menor error total.

La desventaja del método Ridge es que, el modelo final, incluye todos los predictores. Esto es así porque, si bien la penalización fuerza a que los coeficientes tiendan a cero, nunca llegan a ser exactamente cero (solo si $\lambda = \infty$). Este método consigue minimizar la influencia sobre el modelo de los predictores menos relacionados con la variable respuesta, pero, en el modelo final, van a

seguir apareciendo. Aunque esto no supone un problema para la precisión del modelo, sí lo es para su interpretación. [13]

Lasso

La regularización Lasso penaliza la suma del valor absolutos de los coeficientes de regresión ($\|\beta\|_1 = \sum_{j=1}^p |\beta_j|$). A esta penalización se le conoce como *l1* y tiene el efecto de forzar a que los coeficientes de los predictores tiendan a cero. Dado que un predictor con coeficiente de regresión cero no influye en el modelo, Lasso consigue excluir los predictores menos relevantes. Al igual que en Ridge, el grado de penalización está controlado por el hiper parámetro λ . Cuando $\lambda = 0$, el resultado es equivalente al de un modelo lineal por mínimos cuadrados ordinarios. A medida que λ aumenta, mayor es la penalización y más predictores quedan excluidos.

$$\sum_{i=1}^n (y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij})^2 + \lambda \sum_{j=1}^p |\beta_j| = \text{suma residuos cuadrados} + \lambda \sum_{j=1}^p |\beta_j|$$

La principal diferencia práctica entre Lasso y Ridge es que el primero consigue que algunos coeficientes sean exactamente cero, por lo que realiza selección de predictores, mientras que el segundo no llega a excluir ninguno. Esto supone una ventaja notable de Lasso en escenarios donde no todos los predictores son importantes para el modelo y se desea que los menos influyentes queden excluidos. [13]

Elastic Net

Elastic net incluye una regularización que combina la penalización *l1* y *l2* ($\alpha\lambda\|\beta\|_1 + \frac{1}{2}(1 - \alpha)\|\beta\|_2^2$). El grado en que influye cada una de las penalizaciones está controlada por el hiper parámetro α . Su valor está comprendido en el intervalo $[0,1]$. Cuando $\alpha = 0$, se aplica Ridge y cuando $\alpha = 1$ se aplica Lasso. La combinación de ambas penalizaciones suele dar lugar a buenos resultados. Una estrategia frecuentemente utilizada es asignarle casi todo el peso a la penalización

l1 (α muy próximo a 1) para conseguir seleccionar predictores y un poco a la l2 para dar cierta estabilidad en el caso de que algunos predictores estén correlacionados. [13]

$$\frac{\sum_{i=1}^n (y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij})^2}{2n} + \lambda \left(\alpha \sum_{j=1}^p |\beta_j| + \frac{1-\alpha}{2} + \sum_{j=1}^p \beta_j^2 \right)$$

ESTADO DEL ARTE

A continuación, se mencionan trabajos (tesis, informes) relacionados con la utilización de grafos en el desarrollo para optimización de rutas de transporte y/o en problemas cotidianos.

Aplicación de la teoría de grafos en el diseño de rutas de transporte desde las zonas de producción agrícola hasta la planta de procesamiento

El presente trabajo presenta una aplicación de la teoría de grafos en la optimización del sistema de transporte y la reducción de costos en la operación logística de acopio de jalapeños en una empresa agroindustrial. AIMS A opera en la sierra y selva central del Perú, en el 2013 y 2014 ha logrado un crecimiento de 20% de crecimiento en ventas de los cuales el jalapeño representa el 50%. Uno de los problemas que se ha identificado es que el acopio de los jalapeños para trasladarse hasta la planta de procesamiento afronta tres problemas principales: el primero que las llegadas no son en horarios regulares, existiendo retrasos de hasta dos días en el cumplimiento del programa semanal, el segundo es que se usaba unidades de transporte sin medir la capacidad contrastado con la cantidad de jalapeño cosechado por el operador productivo (agricultor) y tercero que el recorrido por cada operador productivo no estaba establecido. Partiendo del programa semanal de cosecha se ha agrupado en “clúster” esto ha permitido asignar la unidad de transporte con capacidades de acuerdo a los Kg, de cosechas, el procedimiento se ajusta “agrupar primero y rutear después”, luego hicimos el ruteo con la ayuda del software denominado *Grafos v 1.2.3*¹ teniendo como restricciones la capacidad del vehículo y la oferta de materia prima de cada operador productivo, cabe resaltar que esto cambia de semana en semana de acuerdo a los “clúster” formados, finalmente se realizó el programa de la flota de vehículos. Los resultados obtenidos fueron: 144 rutas con un recorrido total de 5,838 Km. se han acopiado 1,365,379 Kg de Jalapeño utilizando una capacidad de flota de 1,480,595 Kg, con un costo unitario de transporte de S/. 0.159 cada Kg. Finalmente, para validar el método de dos fases realizamos la programación lineal entera y lo corrimos en AMPL² encontrando oportunidades aun por mejorar tanto en recorrido y en costos. [14]

¹ Grafos es un software para la construcción, edición y análisis de grafos. Este software es utilizado para la docencia y el aprendizaje de la teoría de grafos ([\[Grafos - software para la construcción, edición y análisis de grafos.\]](#)).

² A Mathematical Programming Language. Lenguaje de modelado algebraico.

Dentro de la tesis se evidencia la utilización la segmentación de los nodos del grafo en 4 clúster, en la tesis se modela la red de transporte como, $X = (G, s, t, c)$ es un “dígrafo” del tipo $G = (V, A)$ cuyos vértices identificados son, s y t , y la función c de capacidad que da a cada arco $a = (u, v) \in A(G)$ un número absoluto entero $c(a) = c(u, v)$ el cuál es la capacidad del arco a .

Si tenemos una red de transporte X , la función $\phi: A \rightarrow Z$ que satisface las siguientes condiciones:

$$0 \leq \Phi(a) = \Phi(u, v) \leq c(a) \text{ para cada } a = (u, v) \in A(G)$$

$$\sum_{v \in \Gamma^+(u)} \phi(u, v) = \sum_{v \in \Gamma^-(u)} \phi(v, u) \text{ para cada } u \in V(G) - \{s, t\}$$

se denomina un flujo en X , también podemos decir que $\Phi(a)$ es el flujo que pasa por el arco a . [14] Como input de información, se utiliza la información correspondiente a la flota de vehículos con sus capacidades y costos, y los clientes que en este caso son los agricultores con la oferta de materia prima. Como Nodo origen “0” al centro de acopio que se localiza en Pichanaky y los Nodos del “1” al “16” que correspondían a los operadores productivos. Las aristas representan a las distancias que existe entre cada nodo, estas distancias tienen concordancia al acceso factible que existe realmente entre los nodos, por otro lado, existen nodos que no pueden ser conectados entre sí a las que denominaremos distancias no factibles debido a que no existe vía de acceso entre ellos, por ejemplo el Nodo “11” y el Nodo “14”, para que ambos se conecten tienen que pasar por el Nodo “0” (origen), otro ejemplo que podríamos mencionar son que los Nodos “7” y “11” tampoco pueden ser conectados directamente sin pasar por el nodo “10” por lo tanto a estas aristas lo estamos representado por una distancia muy grande “M”.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	###	M	M	M	M	M	M	15.3	M	15.4	14.5	14.8	M	M	M	M	22.0
1	M	##	1.4	1.1	1.0	0.6	1.4	1.6	M	M	M	M	M	M	M	M	M
2	M	1.4	###	0.7	M	1.2	2.0	2.2	1.5	M	M	M	M	M	M	M	M
3	M	1.1	0.7	###	1.3	0.9	1.7	1.9	M	M	M	M	M	M	M	M	M
4	M	1.0	1.6	1.3	###	0.8	1.6	1.8	M	M	M	M	M	M	M	M	M
5	M	0.6	1.2	0.9	0.8	###	1.2	1.4	2.7	M	M	M	M	M	M	M	M
6	M	1.4	2.0	1.7	1.6	1.2	###	1.8	M	3.9	M	M	M	M	M	M	M
7	15.3	1.6	2.2	1.9	1.8	1.4	1.8	###	M	M	1.2	M	M	M	M	M	M
8	M	M	1.5	2.2	M	2.7	M	M	###	M	M	M	M	M	M	M	M
9	15.4	M	M	M	M	M	M	2.1	M	###	2.1	1.4	M	M	M	M	M
10	14.5	M	M	M	M	M	M	1.2	M	2.1	###	0.7	M	M	M	M	M
11	14.8	M	M	M	M	M	M	M	M	1.4	0.7	###	M	M	M	M	M
12	M	M	M	M	M	M	M	M	M	M	M	M	###	0.2	M	0.5	M
13	M	M	M	M	M	M	M	M	M	M	M	M	0.2	###	3.0	0.7	M
14	M	M	M	M	M	M	M	M	M	M	M	M	M	3.0	###	M	3.5
15	M	M	M	M	M	M	M	M	M	M	M	M	0.5	0.7	M	###	M
16	22.0	M	M	M	M	M	M	M	M	M	M	M	6.7	6.5	3.5	7.2	###

Figura 11. Matriz de distancias. Problema de rutas con vehículos capacitados (CVRP). [14]

De los resultados obtenidos, usando CVRP y el software grafos, se concluye que un modelo matemático con alto requerimiento computacional reduce el número de rutas hasta 130 y los Km de recorrido hasta 5654 Km; el mismo modelo matemático también reduce los costos globales hasta 203,900, con un costo unitario de transporte de S/. 0.149 por Kg. [14]

Aproximación al problema de ruta más corta con trasbordos

Trabajo de grado, con el fin de diseñar e implementar una herramienta para la composición de rutas en medios de transporte masivo, que tengan como criterios de optimización el tiempo y número de trasbordos. El problema es analizado dentro del contexto del sistema Transmilenio, principal proyecto de transporte masivo y recuperación vial de la ciudad de Bogotá, para el cual se definió un grafo fiel a las características del sistema, teniendo en cuenta características complejas como el comportamiento dinámico de la red debido a la salida y entrada en funcionamiento de los servicios troncales de acuerdo con el tiempo.

El estudio de este problema pretende hallar en una red de nodos y arcos el trayecto o ruta más “corta” (el atributo depende del valor total del costo de la ruta, calculado como una medida de la distancia recorrida o tiempo empleado, entre otros) para trasladarse de un nodo de partida a otro de llegada.

El grafo G se compone de un conjunto de nodos N y un conjunto de arcos A , ($G = (N, A)$). El conjunto de nodos N está formado por dos subconjuntos: Nodos de salida y llegada ($N1$) y Nodos de tránsito ($N2$). [15]

Dentro de la tesis, se hace uso del algoritmo *Brumbaugh-Smith and Shier's* el cual se base en el principio de SPP (Shortest Path Problem).

Representación de grafos para la cuantificación de la percepción urbana espacial

Mediante el uso de redes neuronales con grafos y usando *embeddings* con Node2Vec, este trabajo el enfoque es descubrir el efecto que tienen las estructuras de grafo en la tarea de predicción de percepción urbana. Los experimentos realizados muestran el nivel de información que entrega cada tipo de dato y el desempeño resultante de combinar ambas fuentes. [16]

Como input para el algoritmo *Node2Vec*, se construye un grafo de entrada y se estructura la ciudad usando las ubicaciones como nodos; además, usa la proximidad para definir las aristas.

En la tesis se entrenaron tres modelos MLP³ sólo con imágenes y grafos codificados de Node2Vec como input. Los resultados mostraron que el input de imagen tiene mejores resultados que los inputs de Node2Vec, sugiriendo que las características visuales son una más fértil fuente de información que sólo la estructura de grafo. El tercer experimento mezcló datos de ambos tipos, por lo que uso GNNs⁴ para la predicción. Se observó que los modelos GNN obtienen mejores resultados que los modelos MLP de Node2Vec, pero no alcanzan el mismo desempeño que las MLP de imagen como input. Esta información implica que los inputs visuales capturan cualquier información subyacente entregada por una estructura de grafo. [16]

³ Perceptrón multicapa

⁴ Graph Neural Networks

METODOLOGÍA

Para desarrollo del presente trabajo se ha de usar Python, el cual cuenta con un amplio de desarrollo de librerías para el manejo y visualización de grafos, así como una comunidad que constantemente le aportando para su crecimiento y desarrollo.

En el presente trabajo, se ha de usar la librería de *Networkx* con el fin de crear estructuras de grafos y la librería *Pyvis* para visualización e interacción con los mismos.

Networkx: es un paquete de Python para la creación, manipulación y estudio de la estructura, dinámica y funciones de redes complejas. Proporciona: herramientas para el estudio de la estructura y dinámica de redes sociales, biológicas y de infraestructuras; una interfaz de programación estándar y una implementación de grafos adecuada para muchas aplicaciones; un entorno de desarrollo rápido para proyectos multidisciplinarios en colaboración; una interfaz para algoritmos numéricos existentes y código escrito en C, C++ y FORTRAN; y la capacidad de trabajar sin problemas con grandes conjuntos de datos no estándar.

Con NetworkX puede cargar y almacenar redes en formatos de datos estándar y no estándar, generar muchos tipos de redes aleatorias y clásicas, analizar la estructura de las redes, construir modelos de redes, diseñar nuevos algoritmos de redes, dibujar redes y mucho más. [17]

Proyecto Pyvis: El objetivo de este proyecto es construir un enfoque basado en Python para construir y visualizar grafos de red en el mismo espacio. Un grafo Pyvis puede personalizarse por nodo o por arista. Los nodos pueden tener colores, tamaños, etiquetas y otros metadatos. Se puede interactuar con cada grafo, permitiendo arrastrar, pasar el ratón y seleccionar nodos y aristas. El algoritmo de diseño de cada gráfico se puede ajustar también para permitir la experimentación con la representación de gráficos más grandes. Pyvis se basa en la biblioteca VisJS. [18]

Adquisición de Datos

Los datos para generación de grafos consisten en la descripción de rutas aéreas (origen-destino), cantidad de pasajeros transportados y ubicación del aeropuerto con sus coordenadas geográficas

para aquellos casos en los cuales exista la información, así como la información de su código IATA y/o ICAO. Esta información fue obtenida de la página de la [Aerocivil](#) e [Infraestructura Visible](#).

Tabla 1. Adquisición de datos

Fuente	Formato	Registros	Columnas
Base de Datos Origen - Destino	Xlsx	6529	17
Maestra Aeródromo	Xlsx	1004	4
Aeropuertos del mundo	CSV	9300	8

EDA - Análisis exploratorio de datos

El conjunto de datos denominado “Base de Datos Origen – Destino” contiene la información correspondiente a las rutas aéreas entre una ciudad de origen hacia una ciudad destino, para un determinado mes, este conjunto de datos no contiene datos complementarios como georreferenciación ni información de IATA y/ICAO.

Así mismo, la información presentada contiene registros de rutas recurrentes por una misma empresa, evidenciando el cambio en el número de pasajeros transportados.

La “Base de Datos Origen - Destino”, consta de 6529 registros y 17 columnas

Tabla 2. Conjunto de datos origen destino

#	Campo	Descripción
1	Sigla Empresa	Sigla correspondiente a la empresa transportadora
2	Nombre	Nombre empresa
3	Fecha	Fecha
4	Año	Año
5	Número de Mes	Mes
6	Origen	Sigla de aeropuerto origen
7	Nombre.1	Nombre origen
8	Ciudad Origen	Ciudad Origen
9	País Origen	País Origen
10	Destino	Destino origen
11	Nombre.2	Nombre destino
12	Ciudad Destino	Ciudad Destino
13	País Destino	País Destino

14	Tráfico (N/I)	N → Nacional I → Internacional
15	Tipo Vuelo	Tipo Vuelo R → Regular
16	Pasajeros	Cantidad pasajeros
17	Carga + Correo (Kg)	Valor en kilogramos de la carga transportada.

El conjunto de datos denominado “Maestra aeródromo”, contiene la información asociada a la sigla designada por la Aerocivil para identificar la ciudad y departamento del aeródromo, cabe señalar que la información únicamente corresponde a los aeródromos de Colombia; consta de 1004 registros y 4 columnas.

Tabla 3. Maestra aeródromo

#	Campo	Descripción
1	Sigla Aerocivil	Sigla
2	Nombre	Descripción aeródromo
3	Ciudad Origen	Ciudad Origen
4	Departamento Origen	Departamento Origen

El conjunto de datos denominado “Aeropuertos del Mundo” corresponde a la información a nivel internacional de los aeropuertos con descripciones de georreferenciación e indicaciones de códigos ICAO y/o IATA. Consta de 9300 registros y 8 columnas.

Tabla 4. Aeropuertos del mundo

#	Campo	Descripción
1	ICAO	Código ICAO (International Civil Aviation Organization)
2	IATA	Código IATA (International Air Transport Association)
3	nombre	Nombre del aeropuerto
4	ciudad	Ciudad en la cual se localiza el aeropuerto
5	país	País en el cual se localiza el aeropuerto
6	altitud	Altitud medida en metros sobre el nivel del mar a la que se encuentra el aeropuerto
7	latitud	Latitud medida en grados a la que se encuentra el aeropuerto
8	longitud	Longitud medida en grados a la que se encuentra el aeropuerto

Preprocesamiento

En la información de los conjuntos de datos (en inglés, “data set”), se realiza el análisis estadístico evidenciando:

- En la “Base de Datos Origen Destino”, se evidencian 187 registros con información en blanco/nula en el campo “Carga + Correo (Kg)”; dado que no es información relevante para la generación del grafo, se puede diligenciar con 0 en los casos en los cuales se encuentre sin diligenciar dicho campo. Así mismo, se evidencian 5 registros con información faltante para el campo “Pasajeros”.
- En “Maestra Aeródromo”, se evidencian que para las columnas IATA, nombre, altitud se tienen 5738, 5108 y 5135 registros sin información respectivamente; al filtrar la información de Colombia, se evidencia que para IATA, nombre y altitud se encuentran 11, 1 y 1 registros sin información respectivamente. Lo anterior denota una disminución en los valores faltantes en la información que es de interés, en este caso, la información de Colombia. Por otro lado, las columnas latitud y longitud, tienen la información de tipo texto, pero al ser coordenadas de georreferenciación se procede a realizar su conversión por datos en donde se reemplaza la coma (,) por un punto(.) para ser convertido a un dato de tipo decimal. La presente tabla es un puente o cruce de información para los datos de la origen y destino de rutas aéreas.

Para ver la información estadística de las fuentes de información e imágenes asociadas, se puede consultar el Apéndice: Análisis de datos.

Procesamiento de información

Con la información obtenida, se procede a realizar una tabla intermedia de relación entre las fuentes, con el fin de tener una entidad de referencia con los datos de los aeródromos y su respectiva ubicación geográfica.

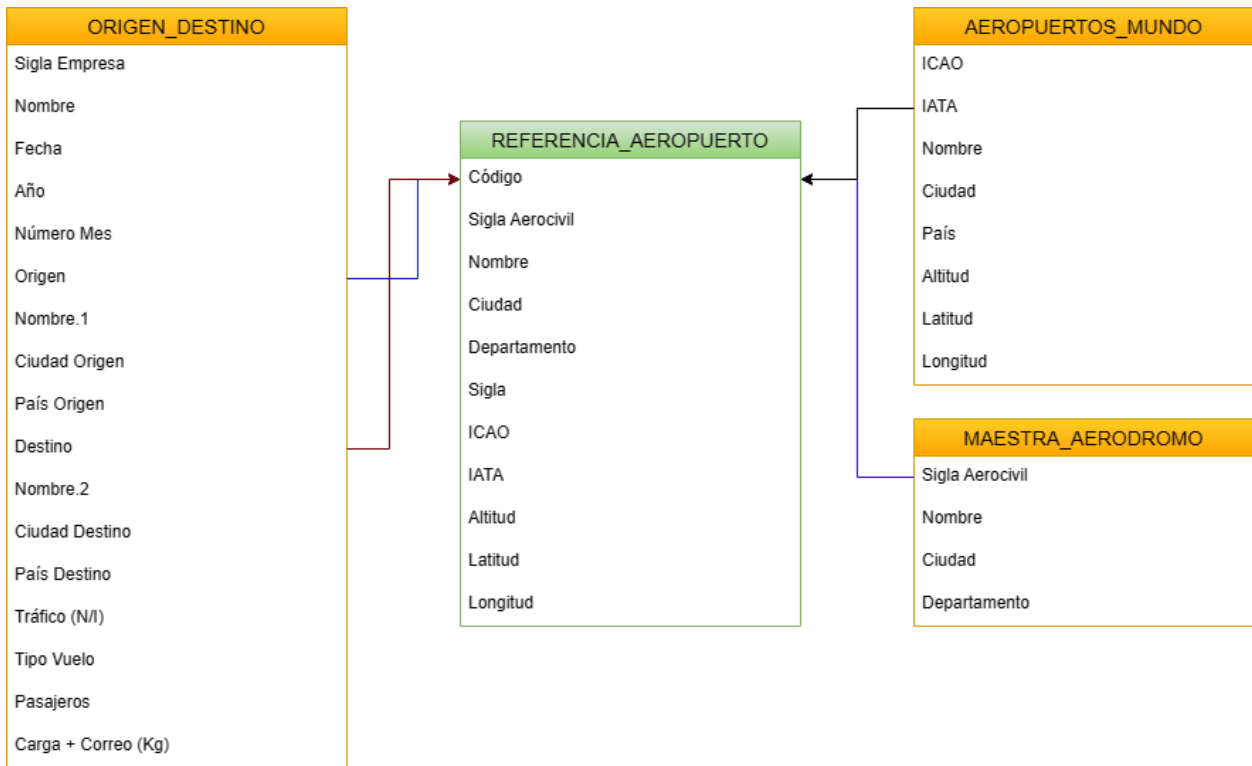


Figura 12. Relación de fuentes. Elaboración propia.

Construcción de Grafo

Con el uso de la librería de Networkx, se procede a realizar la carga de información tomado como

$$G = (V, E)$$

$$V_o = \{ Origen \}$$

$$V_d = \{ Destino \}$$

$$E = \{ v_o, v_d, p \}$$

Donde V_o , representa el nodo origen, V_d corresponde al nodo destino; E representa la relación entre V_o y V_d con p asociada a la cantidad de pasajeros entre la relación (V_o, V_d) .

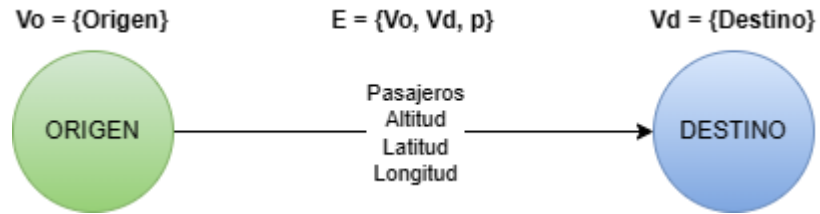


Figura 13. Construcción de grafo. Elaboración propia.

Se procede a realizar la construcción de un grafo dirigido, dado que se requiere representar el sentido de las rutas desde un origen hacia un destino.

Dado que en el conjunto de datos se presentan las mismas rutas (origen, destino) solo variando el número de pasajeros transportados, se procede a realizar una agrupación de rutas sumando la cantidad de pasajero con el fin de unificar las mismas.

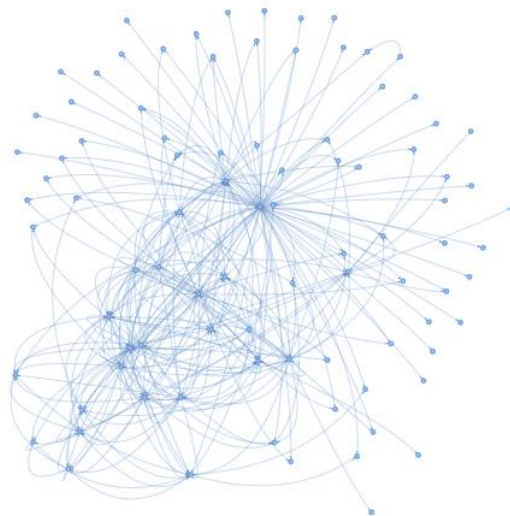


Figura 14. Representación visual del grafo

RESULTADOS Y DISCUSIÓN

Grafo

El grafo generado $G = (V, E)$:

$$V = 90$$

$$E = 321$$

Donde, V corresponde a la cantidad de nodos origen y destino generados, E corresponde a la cantidad de aristas/relaciones entre cada nodo.

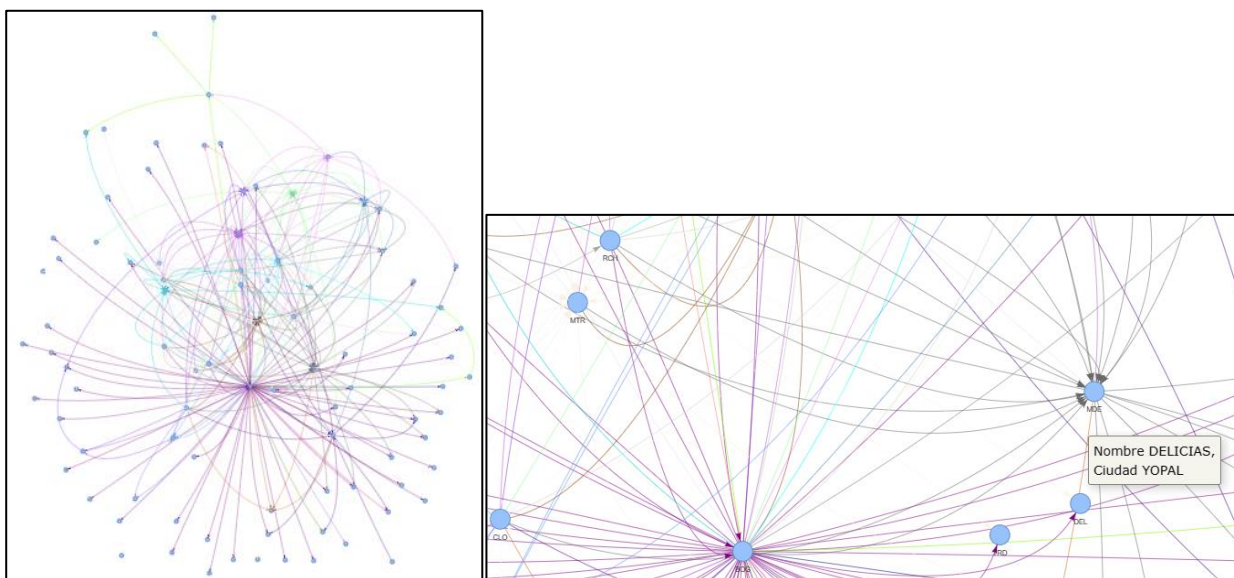


Figura 15. (Izquierda) Grafo General. (Derecha) Grafo con zoom a los nodos.

Tabla 5. Resultado de métricas.

Métrica	Valor
Longitud de ruta	2.0719101123595505
Eficiencia global	0.515043695380785
Eficiencia local	0.5924115601763409
Coefficiente de agrupación	0.5535312122533126

Interpretación de resultados de las métricas obtenidas:

- Longitud de ruta: corresponde a la longitud promedio de nodos que se deben atravesar o recorrer para conectar un nodo con cualquiera de los demás. Esto significa que la mayoría de los viajes implican dos vuelos entre un origen y un destino; así mismo, representa que la mayoría de los viajes requiere dos conexiones intermedias entre el origen y el destino.
- Eficiencia global: el valor obtenido, corresponde a una transmisión de información entre los nodos de forma media, lo que corresponde a que el grafo no es completo o está totalmente conectado. Este valor sugiere que los aeropuertos están conectados de forma moderada, es decir, si el valor es 1, quiere decir que todos los aeropuertos comparten una buena conexión entre sí; el valor 0.51 significa que no todos los aeropuertos están conectados de forma general y las rutas que pasan por ellos.
- Eficiencia local: Un valor de 0.59 en la eficiencia local sugiere que, en promedio, los nodos y sus vecinos directos en el grafo de rutas aéreas están relativamente bien conectados, lo que significa que es eficiente moverse desde un aeropuerto a sus destinos cercanos.
- Coeficiente de agrupación: Corresponde al grado de agrupación de los nodos, es decir, corresponde a la cantidad de vecinos del nodo que se relacionan entre sí. El valor obtenido es relativamente alto, teniendo en cuenta que el valor se da entre 0 y 1; lo cual indica que existen comunidades bien definidas y muy bien conectadas entre sí. El valor obtenido, sugiere que existen agrupaciones o grupos de nodos (aeropuertos) que tienden a estar más densamente conectados entre sí. Estos grupos pueden representar regiones geográficas o rutas aéreas específicas que tienen una alta cantidad de pasajeros que viajan entre ellos.

Embedding

En la exploración de las características de origen-destino y su potencial influencia en la demanda de pasajeros, se empleó el algoritmo *Node2Vec*. Este algoritmo, especializado en la generación de representaciones vectoriales para los nodos en un grafo, permite capturar de manera efectiva propiedades inherentes a la estructura del grafo, tales como las comunidades de nodos. La utilización de *Node2Vec* en este análisis facilita la comprensión detallada de las relaciones y patrones subyacentes en el sistema de origen y destino, enriqueciendo así la investigación sobre los determinantes de la demanda de pasajeros en este contexto.

Similitud

Los *embedding* corresponden a una representación vectorial de los nodos, cuando se aplica la similitud del coseno a estos vectores, se está midiendo cuán similares son en términos de estas características.

Con los resultados obtenidos, y usando la similitud de coseno, se evidencia que los nodos con mayor demanda de pasajeros (basados en sus características vectoriales) corresponden a BOG → Bogotá y MDE → Medellín. Si Bogotá y Medellín resultan tener una alta similitud del coseno, esto sugiere que son muy similares en términos de las características consideradas, lo que podría incluir una alta demanda de pasajeros.

Nodos con mayor demanda de pasajeros: ('BOG', 'MDE')

A continuación, se lista un conjunto de 5 nodos con similitud de rutas con mayor demanda:

Tabla 6. Similitud, rutas con mayor demanda de pasajeros.

Nodo Origen	Ciudad Origen	Nodo Destino	Ciudad Destino	Valor de Similitud
BOG	Bogotá	MDE	Medellín	0.5257835984230042
BOG	Bogotá	CTG	Cartagena	0.3679354367322772
CTG	Cartagena	MDE	Medellín	0.3397145878943951
CLO	Cali	MDE	Medellín	0.28650994086670684
BOG	Bogotá	CLO	Cali	0.2280888929986054

Para consultar la información relacionada con las siglas de la Aerocivil de los aeropuertos de Colombia, dirigirse al Apéndice: Referencias de Aeropuertos de Colombia.

Modelo

Para los modelos, los nodos fueron caracterizados mediante la utilización de *embeddings* y otras características “features” relevantes para llevar a cabo la predicción mediante un modelo de regresión. Se optó por no realizar predicciones directas de *edges* o aristas mediante una regresión simple sobre el grafo. Esto se debió a la falta de suficientes características tanto desde la perspectiva de negocio como en términos de dimensionalidad. En general, las regresiones sobre grafos se llevan a cabo únicamente cuando la volumetría de datos lo permite, ya que, de lo contrario, los resultados podrían carecer de validez y precisión.

Empleando los *embedding* y el algoritmo *Node2Vec*, se generó una serie de modelos con la variable a predecir, es decir, la cantidad de pasajeros (y). Para este propósito, los datos resultantes de *Node2Vec* se dividieron en dos conjuntos: un 80% destinado al entrenamiento de los modelos y el 20% restante reservado para realizar pruebas.

A continuación, se presentan los valores obtenidos por los distintos modelos: Regresión lineal (MCO), Ridge, Lasso, Elastic Net. Para evaluar los modelos, se emplearon las siguientes métricas:

Tabla 7. Métricas obtenidas del modelo.

Modelo	MSE		MAE	
	Entrenamiento	Prueba	Entrenamiento	Prueba
MCO	0.007361	0.014767	0.049632	0.051472
Ridge	0.007903	0.015253	0.038067	0.052832
Lasso	0.008996	0.017146	0.041202	0.051803
Elastic Net	0.008280	0.015777	0.38989	0.053167

De acuerdo con los datos obtenidos del modelo, se puede interpretar:

- En los datos obtenidos, se observa que el MSE en los datos de entrenamiento es bastante bajo para los diferentes modelos evaluados. Un MSE más bajo indica un mejor ajuste del

modelo a los datos. Esto quiere decir, que las predicciones de la demanda de pasajeros en las rutas aéreas se acercan significativamente a los valores reales durante el entrenamiento.

- El MSE de 0.007361 para MCO en los datos de entrenamiento, se interpreta como una medida cuadrática del promedio de los errores entre las predicciones y los valores reales de la demanda de pasajeros. En este caso, un MSE tan bajo sugiere una capacidad significativa del modelo para capturar e identificar patrones en los datos de entrenamiento.
- Se evidencia el MAE relativamente bajo tanto en los datos de entrenamiento como en los de prueba. Un MAE de 0.03 en los datos de entrenamiento, indica que en promedio las predicciones difieren en aproximadamente en 0.03 unidades de la demanda real de pasajeros en las rutas aéreas. Dicho a modo del negocio, si por 100 pasajeros que se transportan desde un origen a un destino, pueden ser que 3 pasajeros no toman la ruta o existen 3 pasajeros adicionales hacia el destino.
- En los datos de prueba, se observa un MAE de 0.05 lo cual sugiere que en promedio se difiere de 0.05 unidades de la demanda real.
- Con un bajo MSE y un MAE reducido obtenido en los modelos, en los datos de entrenamiento y prueba indica una capacidad razonable de predicción de la demanda de pasajeros en las rutas aéreas, con errores de predicción en el rango de 0.03 a 0.05 unidades, lo que podría equivaler a la variación de algunos pasajeros en la demanda entre rutas aéreas.

Tabla 8. Validación cruzada

Modelo	MSE	MAE
MCO	0.009088	0.039998
Ridge	0.009066	0.040766
Lasso	0.010002	0.044150
Elastic Net	0.009440	0.042023

En la Tabla 8. Validación cruzada se evidencian las métricas de evaluación para diferentes modelos de regresión, conforme el problema planteado en la planificación de la demanda de pasajeros entre las rutas de un origen a un destino dentro de Colombia.

Con un valor de 0.09088, el modelo MCO muestra cómo se comporta en términos de predicción de la demanda de pasajeros en las rutas aéreas en promedio a través de diferentes particiones de los datos. Este valor proporciona una medida cuadrática del rendimiento del modelo, con un MSE relativamente bajo indicando un ajuste razonable de los datos. El MAE de 0.039998 indica que las predicciones, en promedio, difieren en 0.039998 unidades de la demanda real, considerando todas las particiones de datos.

En Ridge, los valores de 0.009066 (MSE) y 0.040766 (MAE) sugieren que el modelo tiende a realizar predicciones con errores cuadráticos y absolutos medios inferiores en comparación con el modelo MCO. Esto indica una mejora en la capacidad predictiva del modelo Ridge en múltiples conjuntos de datos.

El modelo Lasso, tiene un MSE de 0.010002 y un MAE de 0.044150, los valores indican que en promedio las predicciones del modelo tienen errores cuadráticos y absolutos medios más altos en comparación con MCO y Ridge; se podría mencionar que la penalización en Lasso está afectando la capacidad del modelo para ajustarse a los datos de manera óptima.

Los valores obtenidos en Elastic Net, se evidencia como un rendimiento intermedio entre Ridge y Lasso, lo que indica que la regularización L1 y L2 no tiene un impacto significativo en la precisión del modelo.

Representación geográfica

A continuación, se visualiza la representación geográfica los nodos.

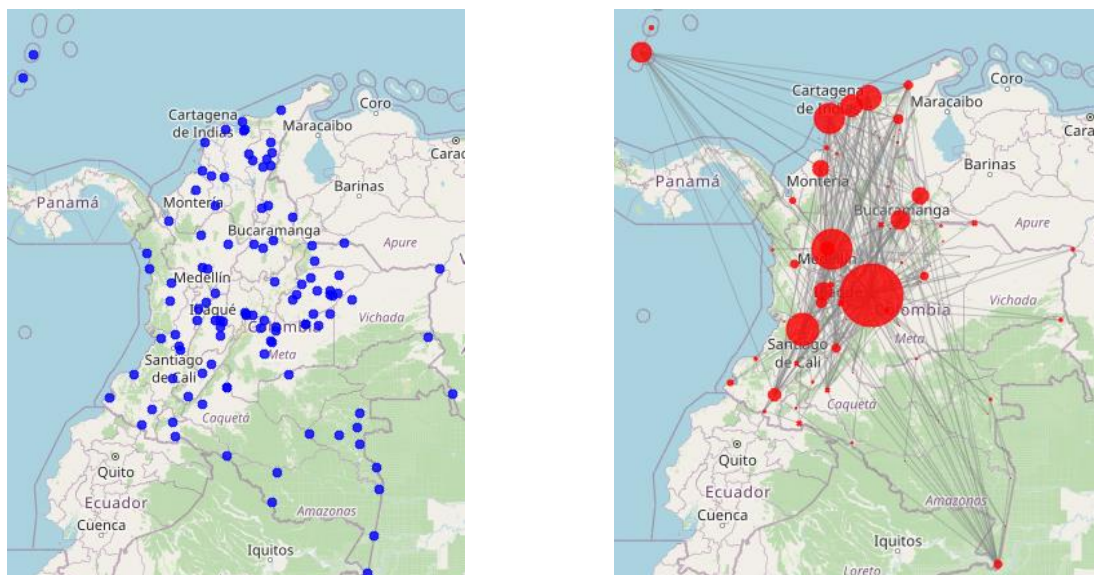


Figura 17. Representación geográfica, (izquierda) ubicación de nodos, (derecha) Conexión entre nodos con tamaño de acuerdo con la cantidad de pasajeros.

En la Figura 17, se puede evidenciar una representación de la posición geográfica de los nodos (en color azul). En dicha representación se evidencian los puntos donde se encuentran ubicados los aeropuertos de Colombia representados como los nodos del grafo, y se evidencian las aristas de conexión entre los diferentes nodos, como las rutas entre un origen y un destino.

Se puede visualizar que existen nodos (en color rojo) de diferente tamaño, lo que significa una alta demanda de pasajeros en ese nodo/ruta, conforme el nodo tiene mayor tamaño mayor es su afluencia de pasajeros que transitan hacia dicho punto, un menor tamaño del nodo significa baja afluencia y/o rutas de poco tránsito de pasajeros hacia ese destino.

Se evidencia, por ejemplo, que el punto central de mayor tránsito de pasajeros corresponde al aeropuerto ubicado en Bogotá, visualmente se identifica otros destinos de mayor afluencia como lo son Medellín y Cali.

Ruta no establecida

A continuación, se ingresa al modelo una ruta que no conoce, la cual el origen es Leticia (LET) y el destino es Arauca (AUC).

Se crea el registro con los datos de la ruta a generar

Nombre.1	Nombre.2	CiudadOrigen	CiudadDestino	Origen	Destino	Codigo_x	altitud_x	latitud_x	longitud_x	color	Codigo_y	altitud_y	latitud_y	
0	LETICIA- ALFREDO VASQUEZ COBO	ARAUCA- SANTIAGO PEREZ QUIROZ	LETICIA	ARAUCA- MUNICIPIO	LET	AUC	LET	85.0	-4.193	-60.943	darkturquoise	LET	85.0	-4.193

Figura 18. Ruta nueva.

Representación del grafo, de la ruta



Figura 19. Grafo de ruta nueva.

Se procede a realizar los *embedding* del grafo

```

{'LET': array([ 0.03902659,  0.06394354,  0.10058941,  0.11736231,  0.01900493,
                -0.02247083,  0.03314611, -0.00207798, -0.00522451,  0.05826742,
                -0.060528 ,  0.10244523,  0.02166174,  0.00056264,  0.0963959 ,
                0.08712672, -0.0336675 ,  0.07295706, -0.06948906,  0.01209706,
                -0.00412511,  0.08076721,  0.0297804 ,  0.12821378,  0.05984847,
                -0.086622 ,  0.04840361,  0.12281844, -0.00620894,  0.04793999,
                -0.08597195,  0.06972005, -0.02982543, -0.02979383,  0.03193457,
                -0.0071806 , -0.03173108,  0.07858301,  0.10003977,  0.0140268 ,
                -0.066932 , -0.04943797, -0.02411066, -0.11353161, -0.02473474,
                -0.00709931,  0.04159763, -0.02346383,  0.06508009,  0.00862312,
                -0.09587651,  0.07171834,  0.00462731, -0.03434258, -0.01247621,
                -0.06525937, -0.1088116 ,  0.03648119,  0.07709054, -0.06563181,
                0.01883404,  0.09768336,  0.0022674 ,  0.04892963], dtype=float32),
'AUC': array([ 0.03489141,  0.05208468,  0.08727442,  0.10827448,  0.01064955,
                -0.00669353,  0.02194634,  0.0048006 , -0.00341908,  0.05307875,
                -0.05515658,  0.08948928,  0.02745192, -0.00995955,  0.08239721,
                0.07744109, -0.01355277,  0.06601495, -0.06055588, -0.00077721,
                -0.01679387,  0.07482232,  0.03611576,  0.11332836,  0.04261554,
                -0.07775211,  0.03436875,  0.11352605,  0.00121589,  0.03584333,
                -0.07360995,  0.06734089, -0.02015408, -0.03817985,  0.02199285,
                -0.01738407, -0.01695975,  0.06941251,  0.08362642,  0.00491821,
                -0.05681449, -0.02969288, -0.02764992, -0.09744286, -0.0138732 ,
                -0.0173215 ,  0.04361002, -0.02988416,  0.05945758, -0.00160303,
                -0.08915344,  0.05711643,  0.00571399, -0.0242264 , -0.01584913,
                -0.0615627 , -0.09278298,  0.01593648,  0.06553567, -0.04676741,
                0.00703966,  0.08957789, -0.00043716,  0.04798403], dtype=float32)}
  
```

Figura 20. Embedding de ruta nueva.

Se ejecuta el respectivo modelo, con los *embedding* generados obteniendo los siguientes resultados:

Tabla 9. Valores obtenidos en la evaluación de la nueva ruta.

Modelo	Valor Obtenido
MCO	0.02977463
Ridge	-0.19246736
Lasso	0.02779188
Elastic Net	0.02779188

Con el MCO, se estima que la cantidad de pasajeros asociada con la ruta planteada es aproximadamente 0.02977463, lo cual sugiere que la cantidad de pasajeros sea relativamente baja. Lo mismo sucede al evaluar la información con Lasso y Elastic Net.

Con respecto a Ridge, el valor obtenido en la evaluación de la nueva ruta puede sugerir un impacto negativo en la demanda de pasajeros; en términos de negocio, se podría interpretar como una ruta de muy baja afluencia o nula de pasajeros desde el origen al destino seleccionado.

A continuación, se realiza una representación geográfica de la ruta a establecer.



Figura 21. Representación geográfica de la ruta nueva.

Se procede a ampliar la cantidad de rutas a 60 registros que no existen actualmente en el modelo, las cuales se puede consultar en el Apéndice: Rutas de prueba.

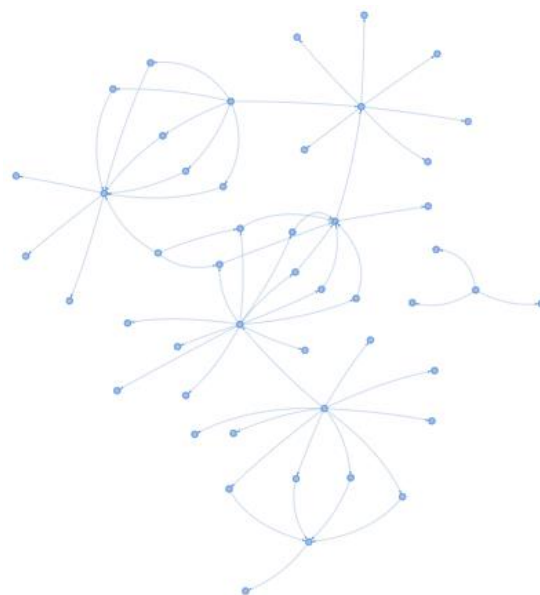
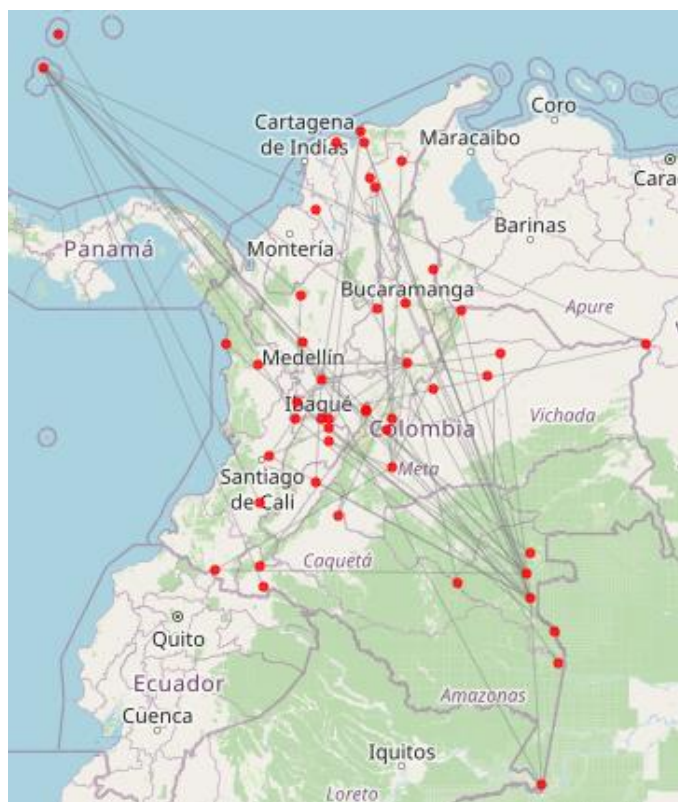


Figura 22. Grafo de rutas no establecidas.



El valor negativo en Elastic Net muestra una tendencia a la baja en la demanda, similar al modelo MCO. Elastic Net combina regularización L1 y L2, por lo que puede ser beneficioso en situaciones donde hay características correlacionadas.

CONCLUSIONES

En el presente trabajo, se logró desarrollar un modelo basado en grafos para la planificación de la demanda de pasajeros en rutas aéreas dentro de Colombia, aplicando técnicas de *machine learning* para grafos.

El grafo generado presenta una complejidad representativa, con 90 nodos (aeropuertos) y 321 aristas (rutas), lo que proporciona una estructura de interés para analizar las conexiones y patrones de tráfico aéreo en el país, esto a su vez fue complementado con métricas de longitud de ruta, eficiencia global, eficiencia local y coeficiente de agrupación que revelan patrones específicos, como la eficiencia en la conexión entre los aeropuertos y la existencia de comunidades bien definidas en el grafo lo que representa cercanía entre destinos.

Se realizó una representación visual de los *embeddings* generados con Node2Vec los cuales revela similitudes entre rutas que comparten características y las cuales se aproximan entre sí.

De los datos obtenidos en los modelos se evidencian similitud en los valores, lo que no representa diferencias significativas entre ellos y de los cuales se puede hacer uso en la planificación de la demanda de pasajeros.

De los resultados obtenidos para la demanda de pasajeros en Colombia, se puede complementar podría complementar con datos en diferentes épocas del año como, por ejemplo, temporada de vacaciones o festividades navideñas, con lo cual se puede generar una herramienta para la toma de decisiones estratégicas en la industria aérea colombiana.

Trabajos futuros:

El presente estudio, se puede complementar a futuro con:

- Integración de datos con información de épocas especiales durante el año (navidad, semana santa, temporada de vacaciones en colegios/universidades)
- Información financiera a nivel de las aerolíneas que operan en una ruta y su proyección de demanda y ganancias conforme a los datos fiscales que se puedan recopilar.

- Ampliar la planificación de la demanda de pasajeros a nivel internacional añadiendo países al grafo, constituyendo así comunidades específicas de operación de las entre origen y destino.
- Despliegue del modelo en un entorno Cloud y conexión a las bases de datos de la demanda de pasajeros con el fin de alimentar el modelo en base a datos actualizados.

REFERENCIAS

- [1] W. Kocay y D. Kreher, *Graphs, Algorithms, and Optimization (Discrete Mathematics and Its Applications)*, Segunda ed., New York: CRC Press, 2023.
- [2] M. Newman, *Networks: An Introduction*, New York: Oxford University Press, 2010.
- [3] S. Skiena, *The Algorithm Design Manual*, Segunda ed., New York: Springer, 2008, p. 145.
- [4] INAOEP, «Matemáticas Discretas - Grafos,» [En línea]. Available: <https://shorturl.at/cimHZ>. [Último acceso: Octubre 2023].
- [5] C. Stamile, A. Marzullo y E. Deusebio, *Graph Machine Learning*, Birmingham: Packt Publishing Ltd, 2021.
- [6] C. F. Montañez Molina, «Formación de vehículos aéreos con aplicaciones a búsqueda y vigilancia,» Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California, México, 2020.
- [7] Iberdrola, «Qué es el 'machine learning',» 2022. [En línea]. Available: [https://www.iberdrola.com/innovacion/machine-learning-aprendizaje-automatizado#:~:text=El%20Machine%20Learning%20es%20una,elaborar%20predicciones%20\(an%C3%A1lisis%20predictivo\)..](https://www.iberdrola.com/innovacion/machine-learning-aprendizaje-automatizado#:~:text=El%20Machine%20Learning%20es%20una,elaborar%20predicciones%20(an%C3%A1lisis%20predictivo)..) [Último acceso: Octubre 2023].
- [8] Z. Blumenfeld, «Graph Machine Learning: An Overview,» 4 Abril 2023. [En línea]. Available: <https://towardsdatascience.com/graph-machine-learning-an-overview-c996e53fab90>. [Último acceso: Octubre 2023].
- [9] Neo4j, Inc, «Graph Embeddings,» 2023. [En línea]. Available: <https://neo4j.com/developer/graph-data-science/graph-embeddings/#:~:text=A%20graph%20embedding%20determines%20a,and%20preserve%20the%20graph%27s%20topology..> [Último acceso: Octubre 2023].
- [10] W. Hamilton, R. Ying y J. Leskovec, «Representation Learning on Graphs: Methods and Applications,» California, 2018.
- [11] Graph Everywhere, «Algoritmo de similitud de coseno,» [En línea]. Available: <https://www.grapheverywhere.com/algoritmo-de-similitud-de-coseno/>. [Último acceso: Octubre 2023].
- [12] The MathWorks, Inc., «¿Qué es la regresión lineal?,» [En línea]. Available: <https://la.mathworks.com/discovery/linear-regression.html>. [Último acceso: Noviembre 2023].

- [13] J. Amat Rodrigo, «Selección de predictores, regularización ridge, lasso, elastic net y reducción de dimensionalidad,» Noviembre 2020. [En línea]. Available: https://cienciadedatos.net/documentos/31_seleccion_de_predictores_subset_selection_ridge_lasso_dimension_reduction. [Último acceso: Noviembre 2023].
- [14] F. Arias Rafael, «Aplicación de la teoría de grafos en el diseño de rutas de transporte desde las zonas de producción agrícola hasta la planta de procesamiento [Tesis de Maestría],» San Miguel, 2017.
- [15] J. P. Suárez Sánchez, «Aproximación al problema de ruta más corta con trasbordos [Tesis de Maestría],» Bogotá, 2005.
- [16] F. Lucchini Wortzma, «Representación de grafos para la cuantificación de la percepción urbana espacial [Tesis de Maestría],» Santiafo de Chile, 2022.
- [17] A. A. Hagberg, D. A. Schult y P. J. Swart, *Exploring Network Structure, Dynamics, and Function using NetworkX*, Pasadena, CA USA: Gäel Varoquaux, Travis Vaught, and Jarrod Millman, 2008, pp. 11-15.
- [18] West Health Institute, «Interactive network visualizations,» 2018. [En línea]. Available: <https://pyvis.readthedocs.io/en/stable/introduction.html>. [Último acceso: 14 Octubre 2023].
- [19] A. Grover y J. Leskovec, «ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,» 2016.

APÉNDICE

Análisis de datos

Base de Datos Origen – Destino

	Fecha	Año	Número de Mes	Pasajeros	Carga + Correo (Kg)
count	6529	6529.0	6529.0	6524.000000	6.342000e+03
mean	2023-03-01 00:00:00	2023.0	3.0	575.101165	1.140166e+04
min	2023-03-01 00:00:00	2023.0	3.0	0.000000	0.000000e+00
25%	2023-03-01 00:00:00	2023.0	3.0	3.000000	0.000000e+00
50%	2023-03-01 00:00:00	2023.0	3.0	18.000000	0.000000e+00
75%	2023-03-01 00:00:00	2023.0	3.0	108.250000	0.000000e+00
max	2023-03-01 00:00:00	2023.0	3.0	56010.000000	7.808290e+06
std	NaN	0.0	0.0	2803.696504	1.443139e+05

Figura 25. Descripción estadística de Origen – Destino

```
Data columns (total 17 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Sigla Empresa          6529 non-null   object
1   Nombre                 6529 non-null   object
2   Fecha                 6529 non-null   datetime64[ns]
3   Año                   6529 non-null   int64
4   Número de Mes         6529 non-null   int64
5   Origen                 6529 non-null   object
6   Nombre.1              6529 non-null   object
7   Ciudad Origen         6529 non-null   object
8   Pais Origen           6529 non-null   object
9   Destino               6529 non-null   object
10  Nombre.2              6529 non-null   object
11  Ciudad Destino        6529 non-null   object
12  Pais Destino          6529 non-null   object
13  Tráfico (N/I)        6529 non-null   object
14  Tipo Vuelo            6529 non-null   object
15  Pasajeros             6524 non-null   float64
16  Carga + Correo (Kg)   6342 non-null   float64
dtypes: datetime64[ns](1), float64(2), int64(2), object(12)
memory usage: 867.3+ KB
```

Figura 26. Tipos de dato (Origen – Destino)

	Vacío/Nulo
Sigla Empresa	0
Nombre	0
Fecha	0
Año	0
Número de Mes	0
Origen	0
Nombre.1	0
Ciudad Origen	0
Pais Origen	0
Destino	0
Nombre.2	0
Ciudad Destino	0
Pais Destino	0
Tráfico (N/I)	0
Tipo Vuelo	0
Pasajeros	5
Carga + Correo (Kg)	187

Figura 27. Visualización de nulos (Origen – Destino)

Maestra aeródromo

	Sigla Aerocivil	Nombre	Ciudad Origen	Departamento Origen
count	1004	1004	1004	1004
unique	1004	932	343	33
top	0RR	SAN FELIPE	MITU	CASANARE
freq	1	4	40	178

Figura 28. Descripción estadística maestra aeródromo

```

Data columns (total 4 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Sigla Aerocivil        1004 non-null   object
1   Nombre                 1004 non-null   object
2   Ciudad Origen          1004 non-null   object
3   Departamento Origen    1004 non-null   object
dtypes: object(4)
memory usage: 31.5+ KB

```

Figura 29. Tipos de datos (Maestra Aeródromo)

	Vacío/Nulo
Sigla Aerocivil	0
Nombre	0
Ciudad Origen	0
Departamento Origen	0

Figura 30. Visualización de nulos (Maestra Aeródromo)

Aeropuertos del mundo

	altitud	latitud	longitud
count	4165.000000	9300.000000	9300.000000
mean	300.471309	10.932368	-0.618812
std	476.997088	22.306613	50.993560
min	-385.000000	-90.000000	-177.381000
25%	21.000000	0.000000	0.000000
50%	96.000000	0.000000	0.000000
75%	364.000000	26.773250	2.113000
max	4060.000000	82.518000	179.340000

Figura 31. Descripción estadística de Aeropuertos del mundo

```

Data columns (total 8 columns):
#   Column      Non-Null Count  Dtype
---  -
0   icao          9300 non-null    object
1   iata         3562 non-null    object
2   nombre       4195 non-null    object
3   ciudad       9300 non-null    object
4   pais         9300 non-null    object
5   altitud      4165 non-null    float64
6   latitud      9300 non-null    float64
7   longitud     9300 non-null    float64
dtypes: float64(3), object(5)
memory usage: 581.4+ KB

```

Figura 32. Tipos de datos (Aeropuertos del mundo)

	Vacío/Nulo
icao	0
iata	5738
nombre	5105
ciudad	0
pais	0
altitud	5135
latitud	0
longitud	0

Figura 33. Visualización de nulos (Aeropuertos del mundo)

	Vacío/Nulo
icao	0
iata	11
nombre	1
ciudad	0
pais	0
altitud	1
latitud	0
longitud	0

Figura 34. Visualización de nulos a nivel país (Colombia)

Referencias de Aeropuertos de Colombia

A continuación, se presenta la tabla de referencias creada con las bases de la Aerocivil:

Tabla 11. Tabla de aeropuertos de Colombia.

Código	Nombre	Ciudad	Departamento
7FU	LA ESCONDIDA	PUERTO Gaitán	META
7GL	LA JAGUA	LA JAGUA IBIRICO	CESAR
7GS	EL ENCANTO	CUMARIBO	VICHADA
7GX	LA ATARRAYA - CA	AYAPEL	CÓRDOBA
7NS	JAGUAR	PUERTO GAITÁN	META
7NT	OCELOTE	PUERTO GAITÁN	META
9AK	CANANARI	MITÚ	VAUPÉS
9AS	SIRENAS	TRINIDAD	CASANARE
9BY	SAN EDUARDO	OROCUE	CASANARE
9CU	COROCORA	TRINIDAD	CASANARE
9DE	LUNA PARK	LA PRIMAVERA	VICHADA
9DI	SAN FELIPE	SAN FELIPE	GUAINÍA
ADZ	SAN ANDRÉS-GUSTAVO ROJAS PINILLA	SAN ANDRÉS - ISLA	SAN ANDRÉS Y PROVIDENCIA
AGH	AGUACHICA HACARITAMA	AGUACHICA	CESAR
AGU	MARÍA ANGELICA	AGUAZUL	CASANARE
AMA	LA AMALIA	CIÉNAGA	MAGDALENA
APO	ANTONIO ROLDAN BETANCOURT	CAREPA	ANTIOQUIA
AUC	ARAUCA - SANTIAGO PÉREZ QUIROZ	ARAUCA - MUNICIPIO	ARAUCA
AXM	ARMENIA - EL EDÉN	ARMENIA	QUINDÍO
BAQ	BARRANQUILLA-E. CORTISSOZ	BARRANQUILLA	ATLÁNTICO
BER	BECERRIL	BECERRIL	CESAR

BGA	BUCARAMANGA - PALONEGRO	BUCARAMANGA	SANTANDER
BOG	BOGOTÁ - ELDORADO	BOGOTÁ	DISTRITO CAPITAL
BSC	BAHÍA SOLANO - JOSÉ C. MUTIS	BAHÍA SOLANO	CHOCO
BUN	BUENAVENTURA	BUENAVENTURA	VALLE
CAI	CAJA DE VIVIENDA MILITAR - BOGOTÁ	BOGOTÁ	DISTRITO CAPITAL
CAN	CANARIAS	NUNCHIA	CASANARE
CEN	CENTRO INTERNACIONAL TEQUENDAMA	BOGOTÁ	DISTRITO CAPITAL
CHA	CHICORAL ESTRA	EL ESPINAL	TOLIMA
CHO	CHICAMOCHA	VILLAVICENCIO	META
CLO	CALI - ALFONSO BONILLA ARAGÓN	CALI	VALLE
COG	CONDOTO MANDINGA	CONDOTO	CHOCO
CPC	CUPICA	BAHÍA SOLANO	CHOCO
CPE	CAPELLANÍA	ARIGUANÍ	MAGDALENA
CTG	CARTAGENA - RAFAEL NÚÑEZ	CARTAGENA	BOLÍVAR
CUC	CÚCUTA - CAMILO DAZA	CÚCUTA	NORTE DE SANTANDER
CZU	COROZAL - LAS BRUJAS	COROZAL	SUCRE
DAR	DARIÉN CALIMA	DARIÉN - VALLE	VALLE
DEL	DELICIAS	YOPAL	CASANARE
EJA	BARRANCABERMEJA- YARIGUIES	BARRANCABERME JA	SANTANDER
ELP	EL PAILÓN	CANDELARIA	VALLE
EOH	MEDELLÍN - OLAYA HERRERA	MEDELLÍN	ANTIOQUIA

EYP	EL YOPAL	EL YOPAL	CASANARE
FLA	GUSTAVO ARTUNDUAGA PAREDES	FLORENCIA	CAQUETÁ
FLR	LAS FLORES	CODAZZI	CESAR
FTE	CAÑO BLANCO	FUENTE DE ORO	META
GPI	GUAPI - JUAN CASIANO	GUAPI	CAUCA
HEL	HELITAXI	BOGOTÁ	DISTRITO CAPITAL
IBE	IBAGUÉ - PERALES	IBAGUÉ	TOLIMA
IDA	PUERTO INÍRIDA - CESAR GAVIRIA TRUJILLO	PUERTO INÍRIDA	GUAINÍA
IPI	IPIALES - SAN LUIS	ALDANA	NARIÑO
LAD	LA DILIA	CODAZZI	CESAR
LAP	LA CARPETA	PAZ DE ARIPORO	CASANARE
LCH	LA CHORRERA - VIRGILIO BARCO VARGAS	LA CHORRERA	AMAZONAS
LET	LETICIA-ALFREDO VÁSQUEZ COBO	LETICIA	AMAZONAS
LFT	LA FLORESTA	YAGUARÁ	HUILA
LOS	LOS MANGOS	SALDAÑA	TOLIMA
LPA	LA PALMA	BOSCONIA	CESAR
LPD	LA PEDRERA	LA PEDRERA	AMAZONAS
LQM	PUERTO LEGUIZAMO	PUERTO LEGUIZAMO	PUTUMAYO
MAA	MATA DE AGUA	PAZ DE ARIPORO	CASANARE
MDE	RIONEGRO - JOSÉ M. CÓRDOBA	RIONEGRO - ANTIOQUIA	ANTIOQUIA
MGN	MAGANGUÉ - BARACOA	MAGANGUÉ	BOLÍVAR
MIT	WACARICUARA	MITÚ	VAUPÉS
MSN	SAN PABLO - VAUPÉS	MITÚ	VAUPÉS

MTR	MONTERÍA - LOS GARZONES	MONTERÍA	CÓRDOBA
MTT	CACHIPORRO	MITÚ	VAUPÉS
MVP	MITÚ	MITÚ	VAUPÉS
MZL	MANIZALES - LA NUBIA	MANIZALES	CALDAS
NUE	NUEVO MUNDO	TRINIDAD	CASANARE
NVA	NEIVA - BENITO SALAS	NEIVA	HUILA
OCV	OCAÑA - AGUAS CLARAS	OCANA	NORTE DE SANTANDER
ORH	ORIHUECA	CIÉNAGA	MAGDALENA
OTU	REMEDIOS OTU	REMEDIOS	ANTIOQUIA
PCR	CARREÑO-GERMAN OLANO	PUERTO CARREÑO	VICHADA
PEI	PEREIRA - MATECAÑAS	PEREIRA	RISARALDA
PEN	PENSILVANIA	PENSILVANIA	CALDAS
PIP	PAIPA JUAN JOSÉ RONDÓN	PAIPA	BOYACÁ
PLS	TOBASIA	PUERTO LÓPEZ	META
PPN	POPAYÁN – GUILLERMO LEÓN VALENCIA	POPAYÁN	CAUCA
PRA	LA OPIA	PIEDRAS	TOLIMA
PSO	PASTO - ANTONIO NARIÑO	PASTO	NARIÑO
PTL	PITALITO -CONTADOR	PITALITO	HUILA
PUU	PUERTO ASIS - 3 DE MAYO	PUERTO ASIS	PUTUMAYO
PVA	PROVIDENCIA- EL EMBRUJO	PROVIDENCIA	SAN ANDRÉS Y PROVIDENCIA
RCH	RIOHACHA-ALMIRANTE PADILLA	RIOHACHA	GUAJIRA
RVE	SARAVENA- COLONIZADORES	ARAUCA - MUNICIPIO	ARAUCA
SCY	SAN JERÓNIMO	FLANDES	TOLIMA

SIN	LA CEIBA - META	SAN MARTIN	META
SJE	SAN JOSÉ DEL GUAVIARE	SAN JOSÉ DEL GUAVIARE	GUAVIARE
SMR	SANTA MARTA - SIMÓN BOLÍVAR	SANTA MARTA	MAGDALENA
SNU	SAN MIGUEL - META	SAN CARLOS DE GUAROA	META
STR	SANTA RITA DE ITUANGO	SANTA RITA DE ITUANGO	ANTIOQUIA
SVI	SAN VICENTE DEL CAGUÁN	SAN VICENTE DEL CAGUÁN	CAQUETÁ
TAP	TARAPACÁ	LETICIA	AMAZONAS
TAR	TARAIRA	TARAIRA	VAUPÉS
TCO	TUMACO - LA FLORIDA	TUMACO	NARIÑO
TDA	TRINIDAD	TRINIDAD	CASANARE
TLU	TOLÚ	TOLÚ	SUCRE
TME	TAME	TAME	ARAUCA
TRD	PALMARITO 2	TRINIDAD	CASANARE
TRI	LOS TOROS	TRINIDAD	CASANARE
UIB	QUIBDÓ - EL CARAÑO	QUIBDÓ	CHOCO
VAR	VARSOVIA	SAN VICENTE DEL CAGUÁN	CAQUETÁ
VGP	VILLA GARZÓN	VILLA GARZÓN	PUTUMAYO
VUP	VALLEDUPAR-ALFONSO LÓPEZ P.	VALLEDUPAR	CESAR
VVC	VANGUARDIA	VILLAVICENCIO	META

Rutas de prueba

Tabla 12. Rutas para prueba de modelos.

Origen	Destino
ADZ	EOH
ADZ	EJA
ADZ	PCR
ADZ	STR
ADZ	TAR
ADZ	PRA
ADZ	MIT
ADZ	PUU
ADZ	PPN
ADZ	CAI
AXM	PVA
AXM	PCR
AXM	STR
AXM	TAR
AXM	PRA
CHA	CLO
CHA	SMR
CHO	LPD
CHO	CPE
CHO	VUP
FTE	VVC
FTE	LOS
FTE	LPA
FTE	IPI
FTE	RVE
MIT	CUC

MIT	BGA
MIT	BSC
MIT	CZU
MIT	AMA
MIT	PEI
MIT	LAP
MIT	NVA
MIT	LET
MIT	CLO
MIT	SMR
MSN	VVC
MSN	LOS
MSN	LPA
MSN	IPI
MSN	RVE
MSN	CEN
MSN	CHA
MSN	BAQ
MSN	AGU
PEN	PEI
PEN	LAP
PEN	MVP
PEN	NVA
PEN	LET
PEN	CLO
PEN	SMR
PIP	IBE
PIP	FTE
PIP	SVI
PIP	VGP

PIP	UIB
PIP	PEN
PIP	MTT
PIP	TRI