Casa abierta al tiempo

UNIVERSIDAD AUTÓNOMA METROPOLITANA

Unidad Iztapalapa

Posgrado en Ciencias y Tecnologías de la Información

Nombre del proyecto:

Plataforma de optimización de la movilidad del STC Metro de la Ciudad de México mediante un ensamble de búsqueda local

Presenta:

Lic. Diana Antonia Martínez Sánchez

Asesores:

Dr. Benjamín Moreno Montiel

Dr. René MacKinney Romero

Sinodales:

Presidente: Dr. Maria Elena Lárraga Ramirez

Secretario: Ing. Luis Fernando Castro Careaga

Vocal: Dr. Eduardo Rodríguez Flores

Índice general

Índice de figuras

Índice de tablas

I

Resumen

En este Proyecto de Investigación se aborda la problemática de la movilidad del Sistema de Transporte Colectivo(STC) Metro de la Ciudad de México, mediante un ensamble de búsqueda local; con el cual se buscó el poder ofrecer al usuario posibles rutas que representen una menor probabilidad de contagio ante el Covid-19 y que proporcionen menor tiempo de traslado.
Actualmente, la movilidad de la Ciudad de México no cuenta con alguna base de datos que nos permita almacenar y analizar la información obtenida del STC Metro. Por lo que el primer paso y el más importante en el desarrollo de este trabajo fue la recolección de datos, los cuales conforman nuestra Base de datos llamada Mobidas-CDMX (por su acronimo en ingles *Mobility Data Bases*), cuya finalidad es categorizar cada una de las estaciones que conforman al STC Metro de acuerdo al tipo de riesgo que puedan llegar a representar ante el Covid-19. Esto se realizó mediante el uso de los algoritmos del Aprendizaje maquinal $K - medias$ y la clasificación mediante tablas de decisión. Con cada una de las estaciones categorizadas, el siguiente paso fue el desarrollo de diferentes algoritmos de búsqueda local para la obtención las posibles rutas entre las cuales se pueda seleccionar la mejor ruta, los algoritmos implementados fuerón los siguientes:

- Algoritmos de búsqueda no informada
  - BFS
- Algoritmos de búsqueda informada
  - UCS
  - Dijkstra
  - A*
  - A*Mejorado

Estos Algoritmo nos permitieron la construcción de un ensamble de mezcla de expertos, en el cual mediante algunas reglas de decisión se establece la evaluación de cada una de las rutas mediante el uso del modelo epidemiológico de la viruela de Bernoulli y la formula del movimiento rectilíneo uniforme, dando paso al análisis y selección de las diferentes rutas obtenidas y con ello, obtener la mejor ruta correspondiente a la opción seleccionada por el usuario. El usuario tiene la posibilidad de seleccionar entre dos diferentes opciones para la generación de una ruta, estas son:

- Ruta que proporcionen al usuario una menor probabilidad de contagio.
- Ruta que proporcionen al usuario un menor tiempo de traslado.

Finalmente con el ensamble de mezcla de expertos, se desarrolló el primer prototipo de la plataforma de movilidad del STC Metro de la Ciudad de México.

Capítulo 1

Introducción

1.1. Introducción

1.1.1. Motivación

Según el diccionario de la Real Academia Española, la movilidad se define como el desplazamiento o transporte de personas y cosas a través de medios de locomoción de bajo coste social, ambiental y energético[1]. Con base en lo anterior, se entiende que la movilidad urbana es la circulación de las personas en las ciudades, independientemente del medio que utilicen para desplazarse, ya sea a pie, en transporte público, automóvil, bicicleta, etc. La movilidad es un factor determinante, tanto para la productividad económica de la ciudad, la calidad de vida de sus ciudadanos y el acceso a servicios básicos de salud y educación[2].

La Ciudad de México cuenta con diferentes medios de trasporte, estos se pueden dividir en las siguientes categorías:

- Servicio de trasporte público masivo, en el cual se incluye: Sistema de Transporte Colectivo (STC) Metro, Red de Transporte Público RTP, Sistema de Transporte Eléctrico STE, Trolebús y Tren ligero.
- Servicio de trasporte público colectivo, en el cual tenemos presente al Metrobús y Servicio público colectivo de ruta.
- Servicio de transporte de pasajeros público individual, en el cual se incluyen: taxis, bicitaxi/mototaxi, entre otros.

Los servicios de transporte público son utilizados por usuarios provenientes de la Ciudad de México y el área metropolitana, siendo los principales medios de transporte, sin embargo, la movilidad en la Ciudad de México no se encuentra distribuida de forma homogénea y su infraestructura no se encuentra en óptimas condiciones, debido a que algunos de los servicios de transporte público cuentan con varios años en funcionamiento y no han contado con el mantenimiento adecuado en algunos casos, sumado a ésto el mal uso que se le llega a dar por parte de los usuarios.

Un factor que ha impactado a los Servicios de Transporte Público es la presencia de un nuevo virus que ha azotado a la Cuidad de México y al mundo entero los últimos dos años, nos referimos por supuesto al Covid-19; donde la movilidad de las personas se ha visto gravemente afectada debido a las normas de seguridad sanitaria establecidas. De los servicios de transporte público más afectados, se encuentra principalmente el STC Metro, el cual cuenta con 53 años en servicio aproximadamente y es el medio de transporte con mayor demanda, ya que proporciona accesibilidad a diferentes puntos de la Cuidad de México y cuenta con un costo accesible.
De los principales problemas en el STC Metro tenemos:

- Los tiempos de espera de llegada de tenes y abordaje de vagones, en promedio pueden llegar a superar los 20 minutos.
- Se se excede la capacidad de usuarios en cada vagón, lo cual ocasiona atrasos en la salida del tren e incrementa la inseguridad de los usuarios.
- Horario de llegada y de salida de los trenes.
- Trenes fuera de servicio. El STC Metro cuenta con aproximadamente 101 trenes, de los cuales el 27 % se encuentran fuera de servicio.
- Falta de organización. Uno de los problemas dentro de cada vagón es que no existe una adecuada organización en las áreas restringidas, ocasionando que personas vulnerables se encuentren dentro de cualquier andén.
- Poco o nulo seguimiento de las normas de seguridad sanitarias.

Tomando en consideración lo anterior, en el 2019, el gobierno de la Ciudad de México presentó el Plan Estratégico de Movilidad de la ciudad de México 2019[3], el cual diagnostica a la movilidad como un sistema fragmentado, ineficiente e inequitativo, por lo cual establece tres ejes principales para atacar estas problemáticas:

- Eje 1 Integrar. Integrar física y operacionalmente los medios de movilidad.
- Eje 2 Mejorar. Realizar mejoras a la infraestructuras y servicios de transporte existentes en estado de abandono y deterioro.
- Eje 3 Proteger. Garantizar la protección de los ciudadanos que utilizan los medios de transporte.

Es posible encontrar trabajos similares al Plan Estratégico de Movilidad de la Ciudad de México 2019, como The Transport Strategy for the Highlands and Islands[4] la cual es otra estrategia de movilidad, es realizada por *InnovateUK* junto con la *RGU (Robert Gordon University)*. Esta estrategia se realizó en la región de las tierras altas de Escocia tomando en cuenta los medios de transporte existentes, infraestructura y vialidades, puertos y rutas fluviales tiene como objetivo mejorar la interconexión entre los sistemas de transporte y

servicios públicos, para obtener crecimiento de la región, mejora en la seguridad y una mejor gestión del impacto ambiental.
Evaluando funcionalidad, adecuación y evaluación del funcionamiento a futuro.
Áreas que forman el núcleo de la estrategia:

- Viaje activo.
- Aviación y red aérea de la región.
- Transporte comunitario y sanitario de viajeros.
- Congestión y problemas urbanos.
- Transporte de mercancías.
- Red localmente significativa y mantenimiento de carreteras de la región.
- Transporte de pasajeros convencional.
- Puertos, transbordadores y transporte fluvial.
- Costo de transporte y viaje.
- Impactos ambientales.
- La red estratégica y regional

Podemos notar que, ambas estrategias comparten algunos objetivos los cuales son principalmente: la mejora del servicio de transporte público, el incremento de la seguridad de los usuarios y mantenimiento de las infraestructuras para poder tener mejor acceso a servicios públicos.
Por lo cual, en este Proyecto de Investigación se propone el desarrollo de una herramienta que nos permita modelar los datos de la movilidad del STC Metro, mediante el uso de una Base de Datos de movilidad de la Cuidad de México, Mobidas-CDMX (por su acrónimo en ingles Mobility Data Bases), la cual nos permita abordar alguno de los siguientes puntos:

- Eficiencia energética.
- Mejoras en el servicio.
- Accesibilidad a servicios públicos.
- Mayor seguridad para los usuarios.
- Distribución de áreas (áreas restringidas).
- Asignación de asientos (asientos reservados).
- Tolerancia a fallas.

- Horarios de llegada de los trenes.
- Menores tiempos de espera.
- Evaluar rutas por tiempos de llagada a una estación destino y por probabilidad de contagio de la Covid-19 que esta ruta represente considerando principalmente distancias interestacionales y afluencia de usuarios en cada una de las estaciones.

Mediante el uso de:

- Algoritmos de Aprendizaje Predictivo.
- Algoritmos de Optimización.
- Aprendizaje Maquinal.

Los Algoritmos de Aprendizaje Predictivo son utilizados en la Minería de Datos, se encargan de la búsqueda de patrones o tendencias dentro de un conjunto de datos, para poder identificar riesgos y oportunidades. Los Algoritmos de Optimización tienen como objetivo encontrar dentro de un conjunto de datos posibles alternativas a situaciones de la vida cotidiana con base a un criterio de decisión; es decir buscaran localizar una solución óptima dentro de un espacio de soluciones.
Se pueden emplear dos tipos de estrategias, la búsqueda local y la búsqueda poblacional. Dentro de los algoritmos de búsqueda local tenemos a los algoritmos voraces, entre ellos se encuentran el Algoritmo de Dijkstra, Algoritmo de Clarke y Wright, Hill climbing, entre otros. De los algoritmos de búsqueda poblacional tenemos los algoritmos evolutivos, imitan los principios de la evolución natural éstos incluyen algoritmos genéticos, programación genética / evolutiva, computación evolutiva, estrategias de evolución y evolución diferencial.

1.1.2. Hipótesis

Las hipótesis son las siguientes:

1. Con el uso del Aprendizaje maquinal es posible el desarrollo de técnicas para la construcción de aplicaciones en las que se tenga una adaptación del conocimiento de manera automática, generando modelos sencillos de problemas recurrentes en la movilidad del STC Metro y a partir de ellos generar modelos con situaciones más complejas, y poder adaptarse a situaciones nuevas, de manera que sean estos modelos más flexibles y eficaces.

2. En investigaciones anteriores, se ha demostrado que la Minería de Datos nos permite el análisis información, utilizado para el reconocimiento de patrones o asociaciones, los cuales pueden ser representados por medio de Bases de Datos en forma de tablas o matrices de tamaño MxN.

3. Es posible que mediante el uso de algoritmos predictivos podamos realizar la búsqueda de patrones de comportamiento para poder predecir posibles situaciones. Por ejemplo, dentro del contexto del STC Metro nos interesaría poder predecir tiempos de espera o llegada de un tren a la estación.

4. Mediante el uso de algoritmos de Aprendizaje Maquinal, por ejemplo K-Medias o K-NN, nos permitirán tener una categorización de la información recopilada en una Base de Datos u obtener mejoras en la categorización de cada uno de los datos.

5. Es posible que mediante el uso del algoritmo de Mezcla de Expertos en conjunto con Algoritmos de Optimización, como por ejemplo el Algoritmo de A*, Dijkstra, Recosido Simulado o Algoritmo Genético, podamos obtener rutas con ciertos criterios como son distancias más cortas o menor probabilidad de contagio de la Covid-19.

1.1.3. Objetivos

Para el desarrollo de este Proyecto de Investigación nos enfocamos en el Sistema de Transporte Colectivo Metro; tomando como punto referencia el Plan Estratégico de Movilidad para la ciudad y los tres ejes que este se mencionan, buscando poder atacar la problemática de la movilidad para el Sistema de Transporte Colectivo Metro considerando:

1. Espacio vial

2. Recursos

3. Seguridad para los usuarios

En este tercer punto, ”seguridad para los usuarios” abordamos la evaluación mediante diferentes algoritmos del Aprendizaje Maquinal y Algoritmos de Optimización para la obtención de posibles rutas.

Objetivo general

Construir algoritmos del Aprendizaje Maquinal y búsqueda local para generar un sistema basado en mezcla de expertos para mejorar la movilidad del STC Metro.

Objetivos particulares

1. Ensamblar Algoritmos del Aprendizaje Maquinal y búsqueda local para el problema de movilidad.

2. Unificación de los datos movilidad del STC Metro para generar el primer componente de Mobidas-CDMX (Mobility Data Bases).

3. Programar un sistema de mezcla de expertos que permitan mejorar la seguridad de los usuarios del STC Metro. Obteniendo rutas con menores tiempos de traslado y baja probabilidad de contagio de la Covid-19.

4. Ensamblar el primer módulo para la Plataforma de Optimización de la movilidad de la Ciudad de México.

1.1.4. Justificación

De acuerdo al Reglamento de la ley de movilidad del Distrito Federal publicado en la Gaceta Oficial de la Ciudad de México el 15 de septiembre de 2017, Título segundo de la planeación y la política de movilidad capítulo primero de la planeación de la movilidad, Artículo 14[5] se deben de considerar el establecimiento de normas generales, políticas y estrategias institucionales para garantizar la adecuada movilidad de las personas y establecer una mejor convivencia ciudadana. Además, se deberá asegurar la calidad y seguridad en los servicios proporcionados por el Sistema de Movilidad, tanto en las unidades móviles como en la infraestructura. La movilidad en la Ciudad de México es compleja, ya que cuenta con diferentes sistemas de transporte, programas y proyectos de movilidad que se orientarán a incrementar la accesibilidad, disminuir los tiempos de traslado y garantizar viajes cómodos y seguros. El gobierno de la ciudad de México ha creado medidas de seguridad sanitaria en el transporte público para reducir el riesgo de COVID-19 las cuales se enfocan en la prevención de contagios y reducción de aglomeraciones en los sistemas de transporte público utilizados por usuarios de la Ciudad de México y Zona Metropolitana[6]. De los servicios de transporte públicos con mayor número de usuarios diarios tenemos el Tren Ligero, Metrobús, Trolebús, RTP, Cablebús y Metro.
El STC Metro debido a su gran demanda, alto rendimiento y años de servicio es considerado como el principal transporte público en la Ciudad de México y por esa misma razón es uno de los más afectados por la Covid-19, de ahí nace la importancia de generar Mobidas-CDMX la cual nos permita analizar la información disponible relacionada con tipos de infraestructura y afluencia de pasajeros, esto con el fin de poder realizar:

1. Categorización de estaciones, lo cual nos permitirá evaluar qué tipo de riesgo ante la Covid-19, puede llegar a representar para la salud de un usuario cada una de las estaciones.

2. Recomendación de rutas: estas pueden ser rutas que representen una menor probabilidad de contagio o menor tiempo de traslado.

3. Análisis de la información proporcionada por el usuario: Estos datos proporcionados pueden ser la hora, edad del usuario, estación de inicio y de destino, fecha, entre otras.

Es por ello que con la utilización de clasificadores del Aprendizaje Maquinal como K-Medias se pueden obtener clasificaciones, de riesgo permitiéndonos agrupar un conjunto de observaciones en K clusters, definidos previamente en Mobidas-CDMX los cuales nos puedan permitir más adelante el análisis de información obtenida por un usuario. Al utilizar algunos de los Algoritmos de Optimización como Dijkstra,UCS, A* en conjunto con la información obtenida podamos recomendar al usuario rutas la cuales nos permitan minimizar el riesgo de contagio del Covid-19 o tener menores tiempo de traslado.

1.1.5. Organización de la tesis

El desarrollo de este trabajo está dividido de la siguiente manera: En el Capítulo 2 revisaremos parte de la literatura relacionada con el Aprendizaje Maquinal y Algoritmos de Optimización, los cuales nos permitirán abordar conceptos, algoritmos y temas de investigación que nos ayudaran en el desarrollo de nuestro Proyecto. En el Capítulo 3: Revisaremos trabajos relacionados con nuestra problemática y que nos permitirán obtener visión y guía del planteamiento y desarrollo de nuestro Proyecto. En el Capítulo 4 se describirá la creación de Mobidas-CDMX, los clasificadores del Aprendizaje maquinal y algoritmos de Optimización utilizados para el análisis de información del STC Metro recopilados. En el Capítulo 5 revisaremos los resultados obtenidos en la implementación de cada uno de los Clasificadores del Aprendizaje maquinal, los algoritmos de Optimización y los análisis comparativos de las diferentes rutas recomendadas. Finalmente, en el Capítulo 6 revisaremos las conclusiones, aportaciones y trabajo a futuro.

Capítulo 2

Antecedentes

En este capítulo se presentan algunos de los conceptos que ayudaran al lector a comprender de mejor manera este trabajo.

2.1. Antecedentes

El Aprendizaje Maquinal es una rama de la Inteligencia Artificial, se encarga del desarrollo de técnicas para la construcción de aplicaciones en las que se tenga una adaptación al conocimiento de manera automática, identificación de patrones y toma de decisiones con mínima intervención, se apoya en la Minería de Datos y del uso de Clasificadores de Datos para la extracción del conocimiento, utilizando alguno de los 4 tipos de aprendizaje: inductivo, analítico o deductivo, genético Y conexionista.

2.1.1. Clasificación de datos

Existen empresas y negocios que han almacenado información por años y no le habían dado el valor que realmente tiene, hoy en día esa información es un recurso valioso, y como dijo Sócrates *Sólo hay un bien: el conocimiento, sólo hay un mal: la ignorancia.* Esto se debe a que los datos pueden ser manipulados de manera que permitan obtener el conocimiento de ciertas reglas y poder predecir ciertos comportamientos. Por tal razón, hoy en día, los datos que están siendo almacenados en gigantescos repositorios, por tal razón se han desarrollado técnicas de extracción de datos y búsqueda patrones.
El reconocimiento de patrones es una parte integral de la mayoría de los sistemas de inteligencia artificial creados para la toma de decisiones, cuyo objetivo es la clasificación de objetos en una serie de categorías o clases[7], se han desarrollado herramientas para el reconocimiento de patrones, por ejemplo en fábricas donde se puede identificar si un objeto fabricado es defectuoso o no, reconocimiento de voz, diagnóstico asistido por computadora, el cual sirve de apoyo a los médicos a tomar decisiones con respecto a un diagnóstico, reconocimiento de caracteres (letras o números), entre otros.

Para aprovechar la gran cantidad de datos almacenados, existen los modelos comunicacionales, los cuales se pueden dividir en dos categorías:

1. Modelos Descriptivos (aprendizaje supervisado), los cuales identifican los patrones que permiten predecir o clasificar los datos (asociación y agrupamiento), utilizan modelos estadísticos como la regresión lineal y lógica, se le conoce como análisis cuantitativo.

2. Modelos Predictivos (aprendizaje no supervisado), éstos se encargan de estimar valores de las variables de interés y reconocer la conexión existente con otras variables dentro del conjunto de entrenamiento, todas las variables serán tratadas en un mismo nivel y no existieran hipótesis de causalidad, se le conoce como análisis cualitativo.

La organización de los datos se realiza por medio de las Bases de Datos.

2.1.2. Base de Datos(BD)

Las Bases de Datos son un conjunto de información de un mismo contexto, organizada de manera sistemática que permite recuperar, analizar o difundir dicha información, un ejemplo de esto pueden ser los directorios telefónicos (actualmente en desuso). Éstos contenían cientos de números telefónicos a los que cualquier persona podía acceder y se encontraban ordenados por orden alfabético, o bien por secciones que permitía encontrar, por ejemplo, un plomero; de esta misma manera las Bases de Datos son clasificadas de la siguiente manera:

- Bases de Datos Relacionales: En este tipo de BD, se almacenan y crea un acceso a puntos de datos relacionados entre sí; se basan en el modelo relacional, el cual es una forma intuitiva y directa de representar datos en tablas donde cada fila de la tabla es un registro con un Identificador (ID) único llamado llave. Las columnas de la tabla contienen atributos de los datos, y cada registro generalmente tiene un valor para cada atributo, esto nos permite establecer fácilmente relaciones entre datos[8].

- Bases de Datos Espaciales: En este tipo de BD. se almacena información geográfica como: datos cartográficos, redes de transporte o tráfico, entre otras, son utilizados por ejemplo en aplicaciones que utilizan información geográfica como GPS o mapas.

- Bases de Datos Multimedia: En este tipo de BD se almacenan imágenes, audio o vídeo, texto; son utilizados en plataformas de streaming, plataformas de compra-venta o procesamiento de imágenes y texto, entre otros.

- Bases de Datos Documentales: En este tipo de BD, se tiene un conjunto de repositorios como documentos de texto, de tamaños variables. Estas BD pueden contener desde palabras hasta grandes resúmenes, se usan principalmente para la difusión de información.

- World Wide Web (WWW): Este es uno de los repositorios de información más grande, actualmente contiene todo tipo de datos, los cuales pueden ser clasificados dentro de las BD anteriormente mencionados.

Las Bases de Datos se pueden representar en forma de tabla o matriz de tamaño MxN donde:
N son cada una de las filas y corresponde al número de registros en la BD, cada registro contiene un identificador (ID)
M son cada una de las columnas y corresponden al número de atributos que componen a la BD. Estos atributos pueden ser discretos (tienen un número finito de valores posibles) y continuos (tienen un número infinito de valores)

2.1.3. Extracción de conocimiento de las Bases de Datos

El proceso de extracción de conocimiento en Bases de Datos (KDD-knowledge discovery in databases), tiene los siguientes objetivos:

- Procesar de manera automática grandes cantidades de datos.
- Identificar los patrones más significativos y relevantes.
- Interpretar los patrones más significativos y representarlos como conocimiento útil.
- Llevar a cabo el proceso KDD, como podemos observar en la Figura 2.1, las fases del proceso KDD son las siguientes:

    1. Selección, almacenamiento de datos.
    2. Procesamiento: Consiste en la selección de datos para preprocesamiento y limpieza de datos: elimina información errónea e inconsistente e irrelevantes (criba).
    3. Transformación de los datos. Estos tres primeros pasos incorporan diferentes técnicas como son análisis de decisión, regresión lineal, redes neuronales, técnicas bayesianas, entre otras, e incorpora diferentes áreas entre las observadas en la Figura 2.2.
    4. Minería de Datos: En esta fase, se realiza a partir de modelos predictivos y descriptivos.
    5. Interpretación/evaluación de los datos.
    6. Uso o difusión de los datos (conocimiento obtenido)

Con la finalidad de transformar los datos en información, surge la necesidad del análisis de los datos; la Minería de Datos tiene como objetivo la identificación de patrones y tendencias a través del análisis de datos utilizando tecnologías de reconocimiento de patrones, redes neuronales, lógica difusa, algoritmos genéticos, entre otros [9].

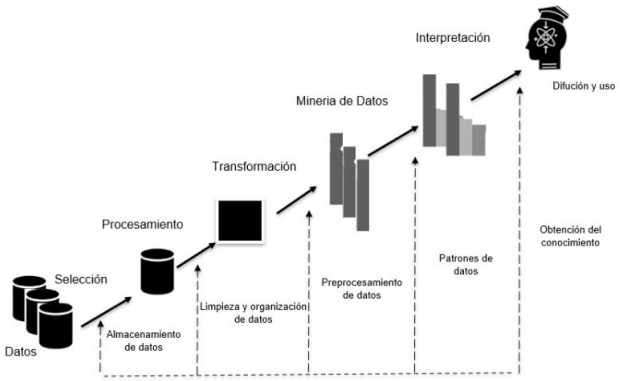


Figura 2.1: Obtención del conocimiento

2.1.4. Minería de Datos

La minería de datos es un campo de la computación, tiene como objetivo descubrir patrones en grandes volúmenes de conjuntos de datos, mediante el uso del aprendizaje maquinal, bases de datos, inteligencia artificial y estadística Las tareas que lleva a cabo la Minería de Datos son:

- Predicción: Descubrir el comportamiento a futuro de algunos atributos.
- Identificación: Identificar la existencia de objetos, eventos y actividades dentro de los datos.
- Agrupamiento: Minimiza las diferencias entre los objetos, mediante algún criterio de agrupamiento
- Asociación: Las reglas de la asociación intentan descubrir cuáles son las conexiones que se pueden tener entre los objetos identificados
- Clasificación: Separa los datos de acuerdo con las clases o etiquetas que sean asignadas a cada ejemplo en los datos. Esto se realiza mediante el uso de diversos clasificadores del aprendizaje maquinal, los cuales pueden ser clasificadores de aprendizaje supervisado (árboles de decisión, Bayes Ingenuo, perceptrón multicapa con retropropagación) y aprendizaje no supervisado (análisis de clústeres).

La Minería de Datos está relacionada con otras áreas como se observa en la Figura 2.2, las cuales son parte fundamental para el correcto manejo de la información. En la Minería de Datos se utilizan diferentes técnicas para la clasificación y agrupamiento de información, una de ellas es el de K-Medias el cual es un algoritmo de agrupamiento de aprendizaje no supervisado.

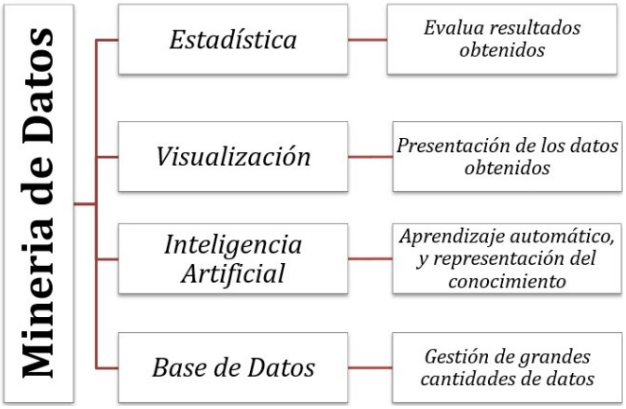


Figura 2.2: Áreas relacionadas con la Minería de Datos

2.1.5. K-Medias

La técnica de análisis de clúster consiste en agrupar un conjunto de datos en un número establecido de clústeres o grupos. Este agrupamiento se realiza mediante la idea de distancia o similitud entre cada una de las observaciones.
Esto se obtiene mediante una medida de similitud, cualquier distancia que utilicemos para medir la similitud entre objetos debe cumplir las siguientes 4 propiedades.
Dados dos vectores x_1 y $x_2 \in \mathbb{R}^+$ se dirá que se ha establecido una distancia entre entre ellos si se define una función d con las siguientes propiedades:

1. $d : \mathbb{R}^k \times \mathbb{R}^k \to \mathbb{R}^+$ por lo que $d(x_i, x_j) \geq 0$
2. $d(x_i, x_j) = 0 \ \forall i$, la distancia entre un elemento y a sí mismo es 0
3. $d(x_i, x_j) = d(x_j, x_i)$, la distancia es simétrica
4. $d(x_i, x_j) \leq d(x_i, x_p) \geq 0 + d(x_p, x_j)$, la distancia cumple la propiedad triangular.

Las distancias que son utilizadas para medir la similitud entre objetos son:

1. Distancia Minkowski. Para los objetos l_1 y l_2 medido según las variables x_1 y x_2, que denotan dos vectores de tamaño 2, la distancia Minkowski entre ambas es:

$$d(l_1, l_2) = (|x_{11} - x_{21}|^m + |x_{12} - x_{22}|^m)^{\frac{1}{m}}$$

Donde $m \in N$
Se puede generalizar con más dimensiones (variables) de la siguiente forma:

$$d(l_1, l_2) = \sqrt[m]{\sum_{k=1}^{p}(|x_{1k} - x_{2k}|^m)}$$

2. Distancia Euclidiana: si $m = 2$
   Se puede formular esta distancia considerando a dos objetos l_1 y l_2 medido según dos variables x_1 y x_2, que denotan dos vectores de tamaño 2, y $m = 2$, la distancia euclidiana entre ambos objetos es:

$$d(l_1, l_2) = \sqrt{(x_{11} - x_{21})^2 + (x_{12} - x_{22})^2}$$

   Y se puede generalizar con más dimensiones de la siguiente forma:

$$d(l_1, l_2) = \sqrt{\sum_{k=1}^{p} (x_{1k} - x_{2k})^2}$$

3. Distancia de Manhattan: Sí $m = 1$
   La distancia de Manhattan hace una similitud con la estructura de las calles de la isla de Manhattan y su estructura se asemeja a una cuadrícula, como los puntos que estarán representados en un plano de una dimensión.
   De la misma forma, si se tienen dos objetos l_1 y l_2 medido según dos variables x_1 y x_2, que denotan dos vectores de tamaño 2, y tomando el coeficiente $m = 1$, la distancia de Manhattan entre ambos es:

$$d(l_1, l_2) = (|x_{11} - x_{21}| + |x_{12} - x_{22}|)$$

   Generalizando:

$$d(l_1, l_2) = \sum_{k=1}^{p} |x_{1k} - x_{2k}|$$

El método de K-Medias, permite agrupar un conjunto de observaciones en K clústeress, definidos previamente por el usuario (distancia existente entre cada observación al centro y (media) al clúster más cercano), también es muy utilizada sobre grandes volúmenes de datos (clasificación de observaciones). Se itera en busca de los centroides de cada clúster para ver si hay una nueva asignación de una observación a un nuevo clúster. Este será el criterio de paro, si no existe ninguna nueva asignación en una interacción el clasificador finalizará su ejecución.
Método de las K-Medias:

1. Se toma de manera aleatoria K centroides iniciales.

2. Para cada una de las observaciones, se calcula la distancia que hay a los centroides y se reasignan a los que estén más próximos. Una vez que se concluye esta reasignación, se vuelven a calcular los centroides de cada uno de los clústeres.

3. Se repiten los dos primeros pasos hasta que no se presente una nueva reasignación de alguna observación a uno de los k centroides.

Ejemplo:
Supongamos que tenemos dos variable x_1 y x_2 y 4 elementos A, B, C, D; a cada elemento le corresponden los siguientes valores:

$$A = (6, -5)$$
$$B = (2, 3)$$
$$C = (6, -2)$$
$$D = (-2, 4)$$

Paso 1:
A estos 4 elementos los dividimos en dos grupos por lo cual $K = 2$ los cuales corresponden a los clústeres AB y CD Calculamos los centroides de cada clúster x_1, x_2
clústeres (AB): $x_1 = \frac{6+2}{2} = 4$ y $x_2 = \frac{-5+3}{2} = -1$
clústeres (CD): $x_1 = \frac{6+(-2)}{2} = 2$ y $x_2 = \frac{-2+4}{2} = 1$
Paso 2:
Calculamos la distancia euclidiana de cada uno de las observaciones al centroide de los clústeres

$$d^2(A, (AB)) = (6-4)^2 + (-5-(-1))^2 = 26$$
$$d^2(A, (CD)) = (6-2)^2 + (-5+1)^2 = 32$$

Observamos que la distancia menor es la que corresponde al clúster AB por lo cual no se reasigna A

$$d^2(B, (AB)) = (-2-4)^2 + (3-(-1))^2 = 20$$
$$d^2(B, (CD)) = (2-2)^2 + (3-1)^2 = 4$$

Observamos que la distancia menor es la que corresponde al clúster CD por lo cual se reasigna B al clúster CD
Paso 3: repetir Paso 1 y paso 2
Paso 1:
Los nuevos clústeres formados son A BCD y se calculan sus centroides.
clústeres (A): $x_1 = 6$ y $x_2 = -5$
clústeres (BCD): $x_1 = \frac{2+2}{2} = 2$ y $x_2 = \frac{3+1}{2} = 2$
Paso 2:
Se calculan distancias euclidianas.

$$d^2(A, (A)) = (6-6)^2 + (-5-(-5))^2 = 0$$
$$d^2(A, (BCD)) = (6-2)^2 + (-5-2)^2 = 65$$

No se reasigna A

$$d^2(B, (A)) = (2-6)^2 + (3-(-5))^2 = 80$$
$$d^2(B, (BCD)) = (2-2)^2 + (3-2)^2 = 1$$

No se reasigna B

$$d^2(C, (A)) = (6-6)^2 + (-2-(-5))^2 = 9$$

$$d^2(C,(BCD)) = (6-2)^2 + (-2-2)^2 = 32$$

Se reasigna C al clúster A
Paso 3: repetir Paso 1 y paso 2
Paso 1:
Los nuevos clústeres formados son AC BD y se calculan sus centroides
clústeres (AC): $x_1 = \frac{6+6}{2} = 6$ y $x_2 = \frac{-5+(-2)}{2} = -3{,}5$
clústeres (BD): $x_1 = \frac{2+(-2)}{2} = 0$ y $x_2 = \frac{3+4}{2} = 3{,}5$
Paso 2:
Se calculan distancias euclidianas.

$$d^2(A,(AC)) = (6-6)^2 + (-5-(-3{,}5))^2 = 2{,}25$$
$$d^2(A,(BD)) = (6-0)^2 + (-5-3{,}5)^2 = 108{,}25$$

No se reasigna A

$$d^2(B,(AC)) = (2-6)^2 + (3-(-3{,}5))^2 = 58{,}25$$
$$d^2(B,(BD)) = (2-0)^2 + (3-3{,}5)^2 = 4{,}25$$

No se reasigna B

$$d^2(C,(AC)) = (6-6)^2 + (-2-(-3{,}5))^2 = 2{,}25$$
$$d^2(C,(BD)) = (6-0)^2 + (-2-3{,}5)^2 = 66{,}25$$

No se reasigna C

$$d^2(D,(AC)) = (-2-6)^2 + (4-(-3{,}5))^2 = 120$$
$$d^2(D,(BD)) = (-2-0)^2 + (4-3{,}5)^2 = 4{,}25$$

No se reasigna D.
Como no se observan cambios tenemos como solución para $K = 2$, que los clústeres son AC BD.

Para poder llevar a cabo la tarea de clasificación de datos en aprendizaje supervisado, se tienen dos tipos de conjuntos de datos, conjunto de entrenamiento (nos permite construir diferentes clasificadores) y conjunto de prueba (nos permite corroborar el correcto funcionamiento del clasificador utilizado). Para poder poner a prueba los clasificadores utilizando estos dos conjuntos de datos se tienen dos métodos, método tradicional (utiliza el 100 % de los datos be la BD para prueba y entrenamiento) y la validación cruzada (utiliza un subconjunto de los datos de la BD para entrenamiento y el resto para prueba).

2.1.6. K-Vecinos más cercanos

K-NN (K-Nearest Neighbours) es uno de los clasificadores de datos más utilizados en Aprendizaje Maquinal, es de tipo aprendizaje supervisado. Dado un conjunto de datos de entrenamiento, y un conjunto de datos de prueba, realiza una comparación por cada nuevo dato y busca similitudes para decidir a qué clase de datos pertenece, es decir busca en las

observaciones más cercanas a la que se está tratando de predecir y la clasifica basado en la mayoría de datos que le rodean utilizando la distancia euclidiana para calcular la distancia entre datos[10]. Método:

1. Dado un elemento a clasificar se calcula la distancia entre el y el conjunto de datos de entrenamiento.

2. Selecciona los elementos más cercanos.

3. Se realiza conteo y/o votación entre los elementos seleccionados para decidir la clase a la que pertenecerá.

2.1.7. Bayes Ingenuo

Al igual que K-NN este clasificador es de aprendizaje supervisado, está basado en el teorema de Bayes, asume que existe independencia de los ejemplos entre cada clase. Utiliza la probabilidad *a priori* esto significa que, dado un conjunto de entrenamiento, se cuenta el número de elementos de muestra y este se divide entre el tamaño del conjunto de entrenamiento. Se divide en tres fases:

1. Representación de los datos, los datos del conjunto de entrenamiento son representados en una tabla o matriz.

2. Cálculo de las probabilidades a priori.

3. Cálculo de las probabilidades condicionales.

Supongamos que tenemos W_1 :clase donde las muestras son verdaderas y W_2 :clase donde las muestras son falsas se tiene la siguiente regla de clasificación con probabilidad a priori:
si $P(W_1) > P(W_2)$ **entonces**
X es un verdadero
otro
X es un falso
La probabilidad es expresada de la siguiente manera:

$$P(x|W_i)\forall i = 1, 2$$

Y si queremos calcular si un ejemplo (X) pertenece a una de las dos clases se utiliza la probabilidad condicional la cual es expresada de la siguiente manera:

$$P(W_i|X)\forall i = 1, 2$$

Utilizando el Teorema de Bayes se puede calcular la probabilidad condicional de la siguiente forma:

$$P(x|W_i) = \frac{P(x|W_i)P(W)}{P(x)}\forall i = 1, 2$$

donde $P(x)$ se calcula de la siguiente forma:

$$P(x) = P(x|W_1)P(W_1) + P(x|W_2)P(W_2)$$

Por lo tanto se toma la clase W_1 si:

$$\frac{P(x|W_1)P(W)}{P(x)} > \frac{P(x|W_2)P(W)}{P(x)}$$

Y eliminando $P(x)$ la regla de clasificación con probabilidad condicional es la siguiente:
si $P(W_i|X) = P(x|W_1)P(W_1) > P(W_i|X) = P(x|W_2)P(W_2)$ **entonces**
X pertenece a W_1
otro
X pertenece a W_2
Suponiendo que se tienen c clases denotadas por $W_1, W_2, ..., W_c$ y que se conocen las probabilidades a priori $P(W_i)$ y las probabilidades condicionales de las clases $P(W_i|X) \forall i = 1, ..., c$. El costo de asignar a la observación de la clase W_i cuando debería ser de la clase W_j se expresa de la siguiente manera:

$$\lambda(W_i|W_j)$$

Entonces la regla de clasificación con probabilidad y riesgo condicionales seria la siguiente:
si $|\lambda_{11} - \lambda_{21}|P(W_1|x) > |\lambda_{22} - \lambda_{12}|P(W_2|x)$ **entonces**
tomar las clase W_1
otro
tomar la clase W_2

La distribución gaussiana para el Teorema de Bayes está definida de la siguiente forma:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$$

la cual tiene una distribución $N(\mu, \sigma^2)$ donde μ es la media y σ^2 es la varianza. La regla de clasificación con probabilidad condicional de distribución gaussiana es la siguiente: **si** $R_1 = P(W_1|x) > R_2P(W_2|x)$ **entonces**
los puntos son asignados a la clase W_1
otro
los puntos son asignados a la clase W_2
En el caso de que $R_1 = R_2$ se le conoce como la frontera que divide a las dos funciones gaussianas.

2.1.8. Clasificación mediante Tablas de Decisión

La Clasificación Mediante Tablas de Decisión, es un algoritmo de Aprendizaje Maquinal, que hace uso de un algoritmo de conteo y agrupamiento para la generación de reglas de decisión[11].
Método:

- Generación de tabla inicial: A partir de un conjunto de datos (Base de Datos) se genera una tabla con cada una de las posibles entradas que en esta se tienen.

- Generación de tabla candidata: Contiene todas las posibles entradas del punto anterior y además se le agregan los valores de $g-count$: número de tuplas equivalentes en cada renglón , $c-count$: número de tuplas del mismo tipo en D donde $D = n\acute{u}merodeestradas$ y estos valores nos permitirán almacenar la información relacionada con el número de tuplas en el conjunto de entrenamiento, sup el cual es el soporte y está definido como: $\frac{g-count}{D}$ y $conf$ el cual corresponde a la confianza y está definido como $\frac{g-count}{c-count}$, A partir de esta tabla candidata se forma una tabla de decisión.

- Generación de tabla de decisión:

  - De la tabla candidata, se leen las tuplas que pertenezcan al mismo grupo (total de tuplas).
  - Se calcula el soporte sup y la confianza $conf$
  - Se definen los umbrales $minsup$ y $minconf$, nosotros los definimos $minsup > 0$ y $minconf > 0$
  - Se descartan tuplas redundantes
  - Continuar estos pasos hasta procesar todas las tuplas en la tabla candidata

- Clasificación de una nueva instancia:

  - Se busca en la tabla de decisión el reglón que sea equivalente a la nueva instancia u, esto es que se busca el reglón que contenga los valores en sus atributos como ANY o sea equivalente de u

  En el caso que exista más de un renglón equivalente se siguen los siguientes pasos:

  - Asignar la clase del renglón donde $conf$ sea mayor, si hay empates asignar la clase donde sup sea mayor.
  - Asignar la clase donde $conf * sup$ sea mayor.
  - Si no se encuentra ningún reglón equivalente se puede utilizar el teorema de Bayes para calcular la probabilidad de cada clase.

2.1.9. Mezcla de Expertos

En el Aprendizaje Maquinal también existen los clasificadores basados en ensambles, donde se plantea como hipótesis inicial, el uso de un conjunto de expertos para aportar beneficios en los principales índices de rendimiento, con respecto a los esquemas tradicionales de clasificación, haciendo una hipótesis inicial sobre la clasificación de una instancia en los datos, utilizando una de las categorías predefinidas, las cuales representan diferentes decisiones.

La decisión se obtiene con base a la fase de entrenamiento del clasificador utilizando un conjunto de datos de entrenamiento, este algoritmo tiene diferente comportamiento para cada base de datos, puede ser utilizado en los siguientes casos: grandes y pequeñas cantidades de datos y razones estadísticas: Problema del sobreajuste y divide y vencerás: límites de decisión complejo dado el caso que un problema son difícil de resolver para un clasificador tradicional. Un experto tiene como objetivo tomar una decisión a partir de la elección de una opción de un conjunto previamente definido de opciones.
El algoritmo de Mezcla de Expertos combina las clasificaciones individuales de diferentes clasificadores, mediante un criterio de votación ponderada, en la mezcla de expertos se tienen dos componentes:

- Un conjunto de clasificadores de base
- Método para la asignación de los pesos

Además, cuenta con un componente que utiliza un criterio de votación ponderada para encontrar la clasificación final del conjunto de prueba.
Como ya se ha mencionado, la Minería de Datos y el uso de clasificadores nos permite la extracción del conocimiento, y fusionado con los Algoritmos de Optimización (búsqueda local o búsqueda poblacional) es posible optimizar las predicciones obtenidas.

2.1.10. Árboles de Decisión

Un Árbol de Decisión es la representación gráfica y analítica de los clasificadores, cada nodo representará un atributo y las ramificaciones serán los posibles caminos que se pueden tomar para determinar la clase de un nuevo dato, el cual podrá ser alguno de los nodos terminales del Árbol de Decisión, utilizan técnicas de búsqueda heurística.

Clasificadores de tipo Árboles de Decisión- Algoritmo ID3

El algoritmo ID3 fue desarrollado por Quinlan en 1983 (considerado un algoritmo seminal), se basa en la teoría de la información (1948) desarrollada por Claude Elwood Shanoon, el cual estudia los mecanismos de codificación de los mensajes y el costo asociado a su transmisión. La cantidad de información se define de la siguiente manera:
Sea $M = m_1, m_2, ..., m_c$ un conjunto de mensajes que tiene cada uno la probabilidad $P(m_i)$; la cantidad de información I en un mensaje

$$I(M) = \sum_{i=1}^{n} -(P(m_i)log_2(P(m_i)))$$

En el algoritmo ID3 se parte de un árbol vacío y se va construyendo de manera recursiva, donde cada nodo será el que tiene mayor grado de información para que sea menos la cantidad que falte cubrir, en la elección de atributos en el conjunto de prueba se buscara que sea mayormente de una clase haciendo uso de la Entropía (E) y la cantidad de información.

la cantidad de información está definida de la siguiente manera:
Dado un conjunto de ejemplos X clasificados en las clases $C = c_1, c_2, ..., c_n$ siendo $|c_i|$ la cardinalidad de la clase c_i y $|X|$ el número total de ejemplos.

$$I(X, C) = -\sum c_i \in c \frac{|c_i|}{|X|} log_2(\frac{|c_i|}{|X|})$$

Entropía:
Para cada uno de los atributos A_i y $v_{1i}, ..., v_{in}$ el conjunto de posibles valores para A_i; $|[A_i(C) = v_{i,j}]|$ el número de ejemplos que tienen el valor v_{ij} en el atributo A_i, la función de entropía está definida como:

$$E(X, C, A_i) = -\sum v_{ij} \in A_i \frac{|[A_i(c) = v_{ij}]|}{|X|} I([A_i(C) = v_{ij}], C)$$

La ganancia se obtendrá del resultado de la cantidad de información y la entropía y se define como:

$$G(|X|, C, A_i) = I(X, C) - E(X, C, A_i)$$

A continuación, se muestra el Algoritmo ID3

Funcion ID3$(X, C, A) \equiv (X : Ejemplos, C : Clases, A : Atributos)$ **inicio**
- **si** *Todos los ejemplos son de la misma clase* **entonces**
  - **Regresar** Hoja con la clase
- **en otro caso**
  - Calcular la función de cantidad de información de los Ejemplos (I)
- **fin**
- **para** *Cada atributo de A* **hacer**
  - **inicio**
    - Calcular la función de entropía (E)
    - Calcular la ganancia de información (G)
    - Escoger el atributo que maximiza (G)
    - Suponiendo que es a
    - Eliminar a de la tabla de atributos (A)
  - **fin**
  - **para** *Cada Partición generada por los valores v_i del atributo a* **hacer**
    - **inicio**
      - $árbol_i \leftarrow$ ID3($Y \leftarrow$ejemplos de X con $a \leftarrow v_i$, clase de Y , Atributos restantes)
      - Generar árbol con $a \leftarrow v_i$ y $árbol$
    - **fin**
    - $ID3 \leftarrow$ la unión de todos los árboles
  - **fin**
- **fin**

fin

Algoritmo 1: Algoritmo ID3

Búsqueda en Amplitud (BFS)

Es un algoritmo de búsqueda no informada utilizado para recorrer todos los nodos de un árbol de manera ordenada, partiendo de un nodo raíz y donde a partir de este se exploran todos los vecinos de este nodo, el recorrido puede realizase iniciando con los nodos de la derecha o por los nodos a la izquierda Como se muestra en la Figura 2.3 se van explorando los nodos de un árbol por niveles partiendo de la raíz, y es un recorrido por la derecha.

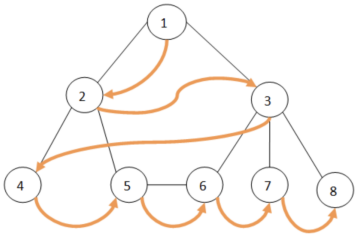


Figura 2.3: Búsqueda en amplitud (BFS)

Búsqueda de Costo Uniforme (UCS)

Es un algoritmo de búsqueda no informada donde se recorren los nodos un árbol buscando el camino con coste mínimo entre un nodo raíz y un nodo destino. Como se muestra en Figura 2.4 este va explorando los nodos del árbol guiándose por el nodo donde se obtenga el menor coste.

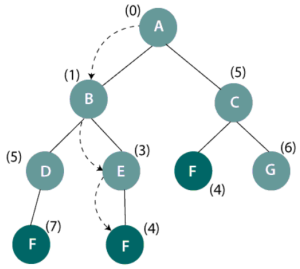


Figura 2.4: Búsqueda de Costo Uniforme (UCS)

Algoritmo de búsqueda A*

Es un algoritmo de búsqueda informada, fue desarrollado en 1964 por Peter E. Hart, Nils J. Nilsson y Bertram Raphael[12], busca el camino de menor coste entre un nodo origen y uno objetivo guiado por una función heurística. El algoritmo de A* usa la función de evaluación

$f(n) = g(n) + h'(n)$ donde:
$f(n)$: coste total
$g(n)$: coste de ir de un nodo a otro
$h'(n)$: función heurística

2.1.11. Búsqueda Local

Un vecindario es una función $N : S \to 2^2$ para cada solución $s \in S$, define un conjunto de soluciones $N(s)$.
Una solución $s \in N(s)$ es una solución vecina de s.
Una solución $i \in S$ es localmente mínima con respecto a N si $\forall j \in N(i), f(i) \leq f(j)$.
Una solución $i \in S$ es óptima (mínimo global) si $\forall j \in S, f(i) \leq f(j)$.
Los algoritmos de búsqueda local nos permiten resolver problemas de optimización discretos, en donde el objetivo es encontrar la mejor solución según una función objetivo, como podemos observar en la Figura 2.5, explorando un espacio de soluciones.
Algunos de los algoritmos de búsqueda local son los algoritmos voraces (greedy) entre ellos

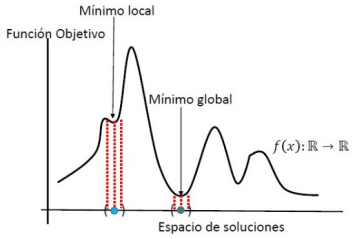


Figura 2.5: Búsqueda Local

se encuentran:

- Algoritmo de Dijkstra: Se le conoce también como algoritmo de caminos mínimos, nos permite determinar el camino más corto, a partir de un punto de partida.
- Algoritmo de Clarke y Wright: Consiste en el principio de combinar una solución de dos rutas distintas y obtener una ruta nueva para obtener un menor costo.
- Hill climbing: Busca el camino más corto y si ya eligió uno y encuentra otro mejor elegirá este segundo, encuentra óptimos locales, pero no garantiza encontrar un óptimo global en el espacio de soluciones.

2.1.12. Búsqueda Poblacional

Los algoritmos de búsqueda poblacional encuentran buenas soluciones seleccionando y combinando soluciones existentes de un conjunto (población)[13].

Operan sobre un conjunto o población de soluciones y utilizan dos mecanismos para buscar buenas soluciones: la selección de soluciones predominantemente de alta calidad de la población y el recombinado de esas soluciones en otras nuevas, utilizando operadores especializados que combinan los atributos de dos o más soluciones. Después de la recombinación, se reinserta en la población, posiblemente requiriendo que satisfagan condiciones como la factibilidad o las exigencias de calidad mínima, para reemplazar otras soluciones (generalmente de baja calidad).
Dentro de la búsqueda poblacional se encuentra la optimización Bioinspirada o algoritmos bioinspirados los cuales que imitan el comportamiento de la naturaleza, buscan una solución óptima (óptima global) a algún problema dentro de un conjunto de posibles soluciones óptimas (óptima local); estos algoritmos están clasificados de la siguiente manera:

- Evolutivos, que incluye: algoritmos genéticos, programación genética, estrategia evolutiva y evolución diferencial.
- Enjambre, que incluye los algoritmos: Enjambre de partículas, colonia de hormigas, colonia artificial de abejas, banco de peces y sistema inmune artificial.
- Ecología, que incluye los algoritmos de biogeografía, colonia de hiervas y simbiosis.
- Mixtos incluye los algoritmos del campo de arroz, sistema del río natural y salto de rana.

Existen herramientas que nos permiten crear simulaciones de movilidad y que podrían llegar a ser de utilidad para el desarrollo de este Proyecto de Investigación, una de ellas es SUMO.

2.1.13. Sumo

El simulador SUMO[14] (Simulation of Urban Mobility) es de software libre, desarrollado en C++ en el 2001 por el *Institute of Transportation Systems*, se utiliza para la simulación de vías de tránsito, patrones de movilidad, etc.

Como podemos observar en la Figura 2.6, SUMO permite modelar diferentes escenarios como vías con varios carriles, límites de velocidad, intersecciones con semáforos, generar mapas de rutas mediante una aplicación denominada Netgen o importarlos desde otras herramientas disponibles como TIGER y OSM, Con el uso de Sumo se puede simular comportamientos generales del entorno o comportamientos particulares de cada uno de los vehículos dentro de la simulación, utiliza el modelo car-following para implementar el comportamiento de un conductor. Sumo cuenta con herramientas que permiten la simulación de tráfico microscópica. Utiliza MOVE (Mobility Model Generator for Vehicular Networks) la cual es una herramienta gráfica utilizada por SUMO para la generación de movilidad y Netconvert que permite importar redes desde otros simuladores como VISUM, VISSIM, etc.

- SUMO-GUI: Es la interfaz gráfica de SUMO que nos permite visualizar de forma más amigable los escenarios generados.

Figura 2.6: Simulación de SUMO

- NETCONVERT: Esta aplicación de línea de comandos, utiliza redes viales generadas por otras herramientas en diferentes formatos para convertirlas al formato que emplea SUMO, es capaz de importar mapas con los formatos "SUMO native"XML descriptions (*.edg.xml, *.nod.xml, *.con.xml), OpenStreetMap (*.osm.xml), VISSUM (*.net), VISSIM (*.net), openDRIVE (*.xodr), MATsim (*.xml), SUMO (*.net.xml), Shapefiles (.shp, .shx, .dbf)

- NETGENERATE: Se encarga de generar redes de carreteras abstractas que pueden ser utilizadas por otras herramientas de SUMO. Permite generar 3 tipos de redes: redes con forma de rejilla, redes con forma de tela de araña y redes aleatorias.

- OD2TRIPS Esta aplicación importa matrices O-D y luego las divide en viajes individuales de vehículos.

- DUAROUTER: Otra aplicación para generar rutas de vehículos definiendo algunas características como origen, destino, tipo de vehículo, etc.

- JTRROUTER: Es otro generador de rutas del paquete de SUMO. En este caso calcula las rutas utilizando porcentajes de giro en las intersecciones, para generar los vehículos define un parámetro con valor entre 0 y 1 que significa que incluirá un número de vehículos por unidad de tiempo.

- DFROUTER: Esta aplicación permite definir y realizar cálculos de las rutas que seguirán los vehículos mediante flujos. De este modo, se pueden definir varios grupos de vehículos con un origen y un destino común, con iguales velocidades y tiempos de salidas entre otras características POLYCONVERT: Se utiliza para importar formas geométricas (polígonos) desde diversas fuentes de datos como OSM, VISSUM, XML y los convierte a una representación que puede ser visualizada por SUMO-GUI.

- ACTIVITYGEN: lee la definición de una población que coincide en una red dada y calcula los posibles movimientos de esta.

Capítulo 3

Estado del Arte

En este capítulo se presentan los artículos consultados durante el desarrollo de este trabajo de investigación, los cuales abordan problemáticas de movilidad semejantes a la problemática de movilidad del STC Metro de la Ciudad de México y problemáticas del Aprendizaje Maquinal.

3.1. Estado del Arte

La problemática de movilidad es un problema al que se le ha buscado solución en diferentes ciudades mundo, ejemplo de ello es el trabajo realizado por Carballo,et al., [15], en las ciudades de Málaga (España) y París (Francia), en el cual se aborda el problema del aumento en la cantidad de semáforos en las grandes urbes, el nivel de contaminación, emisiones y consumo de combustible, realizando una simulación del tráfico vial, utilizan dos algoritmos metaheurísticos, el algoritmo genético celular (cGA) y el algoritmo de Recocido Simulado (en inglés, Simulated Annealing, SA) para optimizar la planificación de los programas de ciclos de los semáforos. Otros trabajos que atacan de igual manera el problema de la movilidad es el trabajo realizado por Neal, et al., [16], el cual se realizó teniendo como primer escenario la red de transporte público de Londres, Reino Unido. Transport for London (TfL) tiene una variedad de billetes con opciones que varían en precio, modalidad de transporte, validez temporal y límites geográficos. la selección del billete(óptimo) dependerá de qué descuentos son elegibles y tres factores que afectan directamente el costo: adónde viajan, hacia donde viajan y desde donde viajan (sus requisitos geográficos), hora en que se realiza el viaje (por ejemplo, las horas pico o días tiempo) y la frecuencia con que se mueve un usuario de un lugar a otro, a lo largo de períodos de tiempo que van desde días a todo un año.
Y el trabajo realizado por S. Foell, et al., [18], para el transporte público de la ciudad de Lisboa, Portugal, en el cual se presenta un estudio predictivo de los patrones de movilidad de los usuarios del transporte público para poner las bases del sistema de información del transporte con capacidades proactivas. Haciendo uso de los datos de tarjetas de viaje de los usuarios de autobuses, haciendo una comparación de diferentes algoritmos de predicción que pueden incorporar varios factores que influyen en la movilidad de las redes de transporte

público, por ejemplo, distancia de viaje o puntos importantes de viaje. Demostrando que al combinar patrones de movilidad personal y poblacional podemos mejorar la precisión de la predicción, incluso con poco conocimiento del comportamiento pasado de los usuarios del transporte público.

Aportaciones

Artículo [15]:

- Recocido Simulado (Simulated Annealing-SA). Es una generalización de un método de Monte Carlo para la evaluación de las ecuaciones de estados y estados congelados de sistemas de n-cuerpos. SA se basa en una analogía de la termodinámica que se ocupa de la manera en que los metales se enfrían. Si se enfría un metal líquido lentamente, sus átomos forman un cristal puro correspondiente al estado de energía mínima para el metal. El metal llega a un estado de menor energía si se enfría rápidamente, es un algoritmo de búsqueda local.

- Algoritmo Genético Celular (cGA). Codifica las variables de decisión de un problema de búsqueda en cadenas de variables de longitud finita de algún alfabeto de cierta cardinalidad. Las cadenas son soluciones candidatas y se llaman cromosomas. De la misma forma que en un GA a cada una de las variables que forman el cromosoma se las denomina gen y alelo a los distintos valores que pueden tomar los genes. Codificado el problema que resolver a través de uno o varios cromosomas (también llamados individuos) y teniendo definida la función de aptitud, se evolucionan las soluciones al problema, es decir la población de soluciones teniendo en cuenta los siguientes pasos: Inicialización, evaluación, selección, recombinación, mutación y reemplazo.

- Simulación, mediante SUMO. El cual permite la simulación microscópica de los diferentes elementos involucrados en el tráfico: vehículos, peatones, transporte público, entre otros, simulación de tráfico multimodal, por ejemplo, vehículos, transporte público y peatones, planificación de programas de control de semáforos, los mismos que pueden ser importados o generados automáticamente por SUMO, entre otras.

Artículo [16]: Técnicas de evaluación de datos:

- Línea de Base: Este clasificador devuelve la mayor frecuencia en un conjunto de entrenamiento, corrige los casos de usuarios que compraron tarjetas de viaje sin necesidad de hacerlo.

- Bayes Ingenuo: Este clasificador está basado en el teorema de Bayes, asume que cada característica de los perfiles de los usuarios es independiente de los demás. Calcula la probabilidad para la selección de un billete con la mejor tarifa.

- k-Vecinos más cercanos: Esta técnica opera encontrando, para cada perfil de prueba, k perfiles similares; la clase predicha es la clase más frecuente que aparece en el conjunto vecino. Primero se define similitud como la diferencia absoluta entre dos perfiles (por lo tanto, valores más pequeños indican mayor similitud).
- Árboles de decisión: El algoritmo C4.5 [17] es un clasificador estadístico que genera un árbol de decisión que se puede utilizar para clasificar instancias de prueba.

Artículo [18]: Algoritmos de predicción.

- Movilidad Personal el cual es el historial de viajes de usuario, las cuales serán las paradas relevantes en el pasado (historial de transporte del usuario), para determinar el número de usuarios en diferentes paradas y paradas visitadas con mayor frecuencia.
- Movilidad Global, los patrones de movilidad global se concentran en las paradas de autobús más populares, por lo que se define la Movilidad Global como la parada con mayor popularidad entre los usuarios.
- Movilidad Geográfica, utiliza la distancia geográfica que calcula distancias de viaje personalizadas en función de las paradas visitadas previamente por, también se incorpora el factor de peso para calcular la distancia ajustada la cual se basa en la relación inversa entre la popularidad de una parada y la popularidad de la parada más alta, el factor de peso puede ser considerado como una fuerza de tracción o empuje en la distancia. Si la parada es impopular, la distancia se aleja más.
- Movilidad de la Red: Es una métrica de distancia más significativa para identificar las rutas de viaje preferidas de los pasajeros que se revelan mediante rutas que están bien conectadas en términos del diseño de la red de transporte.
- Filtrado colaborativo: Se refiere la recomendación basada en elementos sobre un enfoque basado en el usuario cuando el número de elementos supera al número de usuarios.

Un característica recurrente de los artículos anteriores es la agrupación o clasificación de información, el trabajo realizado por Mac Kinney y Montiel[19] presenta un esquema paralelo de las tablas de decisión (ParalTabs), hace uso de memoria compartida, es decir, mediante el uso de hilos que se comunican entre sí leyendo y escribiendo en el mismo espacio físico de direcciones, sigue la estrategia de divide y vencerás(D & C), los datos se entregan a diferentes subprocesos para trabajar y los resultados se recopilan para obtener la tabla de decisión final. Toma en cuenta las limitaciones del esquema secuencial del clasificador mediante Tablas de Decisión dada por el número de combinaciones posibles en ejecución, ya que el número de estas crece exponencialmente a medida que aumenta el tamaño de los datos, atributos y clases, teniendo como consecuencia que los índices de las medidas de desempeño no sean óptimos y el tiempo de ejecución aumente, afectando principalmente en el rendimiento y tiempos de ejecución. Consideran como una posible solución a estos dos problemas aplicar computación paralela para reducir el tiempo de ejecución aumentando el número de combinaciones permitidas, dividiendo grandes conjuntos de datos en conjuntos más pequeños, procesarlos y luego

combinar las soluciones encontradas para obtener un clasificador único.
De igual manera Mac Kinney y Montiel[20] propone un Sistema de Clasificación Paralelo basado en los Ensambles de Mezcla de Expertos (PCEM), para la clasificación de grandes cantidades de datos, considera un conjunto de aprendices débiles(WeLe) que en combinación con un criterio de votación se convierten en aprendices fuertes. El PCEM utiliza un esquema paralelo de votación ponderado (sPGWC), en el que se asigna un peso a cada WeLe mediante un algoritmo genético el cual busca la mejor combinación de pesos; implementa esquemas paralelos para cada WeLe (sPC) y para un algoritmo genético (sPGA).
Otro trabajo realizado por Mac Kinney y Montiel[21], propone realizar una clasificación de datos utilizando un sistema basado en conjuntos. El objetivo de los clasificadores basados en conjuntos es usar varios tipos de clasificadores para mejorar la precisión, mediante un Clasificador Híbrido con Ponderación Genética (HCGW), el cual utiliza un sistema basado en conjuntos de tipo Mezcla de Expertos y un criterio de votación de mayoría ponderada para combinar las clasificaciones individuales de cada clasificador, es decir, cada clasificador tiene un peso diferente según los resultados de un algoritmo genético.

Aportaciones

Artículo [19]: Construcción de Tablas de Decisión.
Artículo [20]: Clasificadores

- Construcción Sistema de Clasificación Paralelo basado en los Ensambles de Mezcla de Expertos.
- Clasificación mediante el uso de Bayes Ingenuo.
- Clasificación mediante Tablas de Decisión.
- Algoritmo C4.5.
- k-vecinos más cercanos
- K-Medias.

Artículo [21]: Clasificadores

- Clasificador Híbrido con Ponderación Genética (HCGW), basado en conjuntos de tipo Mezcla de Expertos.
- Clasificación mediante el uso de Bayes Ingenuo
- Clasificación mediante ADTree
- Clasificación mediante Tablas de Decisión
- Algoritmo C4.5.
- k-vecinos más cercanos
- K-Medias.

Capítulo 4

Desarrollo

En este capítulo se presenta el trabajo realizado para el desarrollo de la Plataforma de Movilidad del STC Metro de la Ciudad de México.

4.1. Construcción Mobidas-CDMX

Como objetivo general, se estableció la construcción de una plataforma de optimización de la movilidad del STC Metro de la ciudad de México, como podemos observar en la Figura 4.1, la plataforma de movilidad tendrá como motor de funcionamiento una Base de Datos la cual lleva por nombre Mobidas-CDMX.

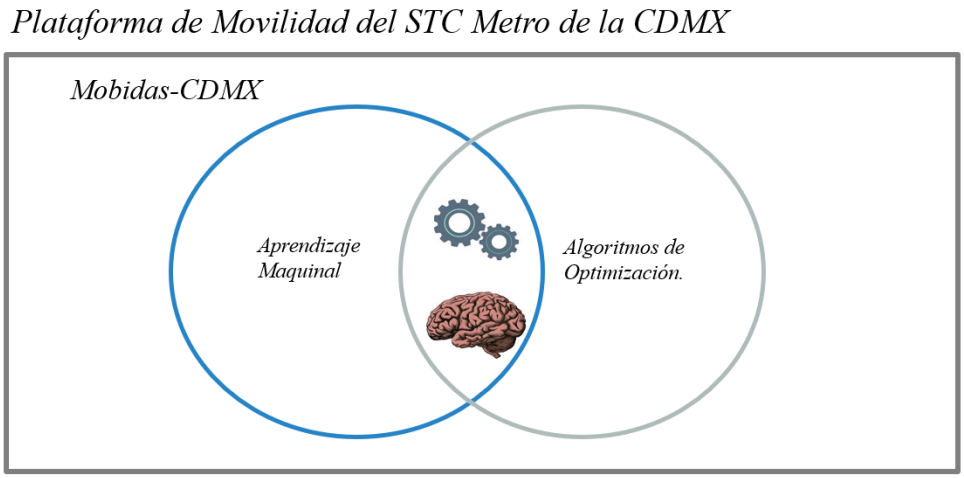


Figura 4.1: Diagrama de Venn para la plataforma de Movilidad de la CDMX

Mobidas-CDMX es la parte crucial para el desarrollo de este Proyecto de investigación, por lo cual se hizo recopilación de datos relevantes de la movilidad del STC Metro los cuales nos permitieran su construcción, como parte de los datos que se recopilaron tenemos los siguientes:

- Estaciones que existen en el Metro de la Ciudad de México[22].
- Tipo de Infraestructura en cada una de las estaciones[23].
- Tipo de estación (con correspondencia y sin correspondencia) [23].
- Longitud interestacional[24].
- Longitudes recorridas por línea[24].
- Horarios de las horas pico[25].
- Consumo energético[25].
- Afluencia de pasajeros por tipo de acceso (acceso normal o de acceso gratuito) [25].
- Afluencia trimestral de cada una de las estacione[26].
- Afluencia anual de cada una de las estaciones[27].
- Longitudes interestacionales[28].
- Conformación de los carros[29].

Se consideró como principales necesidades obtener rutas que permitan al usuario prevenir y disminuir los contagios del Covid-19, en contraste con obtener rutas que nos ofrezcan menores tiempos de traslado (menor distancia interestacional recorrida), se establecieron las siguientes características para conformar Mobidas-CDMX las cuales se describen en la Tabla 4.1:

Mobidas-CDMX se representó en forma de tabla de tamaño MxN donde:

- M: Está compuesto por 16 registros los cuales son: estación, tipo, horario, trimestre (afluencia por trimestre), afluencia (anual por trimestre), Clase de los atributos y coordenadas (longitud, latitud).
- N: Está compuesto por 195 atributos los cuales corresponden a las 195 estaciones que componen el STC-Metro en sus 12 líneas.

Campo	Tipo de dato	Criterio de selección
Estación	Discreto	Este campo nos permitirá definir el número de registros que contendrá nuestra base de datos, esto es: Cada una de las estaciones corresponderá a un registro.
Tipo	Discreto	Este campo corresponde al tipo de infraestructura, subterráneo, elevado y superficial, estos pueden llegar a afectar el tiempo que tomara trasladarse de una estación a otra, puede existir otros factores que modifiquen el tiempo de llegada con respecto al tipo de infraestructura, como el caso del del clima o tipo de estación (con o sin correspondencias), pero estos aun no son incluidos.
Horario	Discreto	En este campo se consideraron tres valores que representará los horarios en los que el número de la afluencia de usuarios de la siguiente manera: 6:00 - 9:00 horas: 2 hora pico 10:00 - 14:00 horas: 0 hora normal 17:00 - 22:00 horas: 1 hora pico
Afluencia trimestral	Discreto	Este campo corresponde a la afluencia por trimestre de cada una de las estaciones
Afluencia	Discreto	Este campo corresponde a la afluencia anual entre la afluencia por trimestre de cada una de las estaciones
Clase de los atributos	Discreto	Este campo corresponde a la asignación de la clase de cada uno de los registros, esto fue asignado utilizando K-medias en conjunto con clasificación arbitraria en consideración de los campos Estación y Tipo
Coordenadas	Discreto	Este campo corresponde a la a las coordenadas geodésicas de cada una de las estaciones

Tabla 4.1: Atributos de Mobidas-CDMX

En la Tabla 4.2, podemos observar un fragmento de Mobidas-CDMX, el cual corresponde a las primeras 6 estaciones de la Línea 1 que va de Pantitlán a Observatorio, para la asignación de los clústeres iniciales se consideró un análisis estadístico para la agrupación de datos en categorías o clases [30] utilizando principalmente la afluencia y el tipo de estación.

A partir de la generación inicial de Mobidas-CDMX se realizó el siguiente trabajo:

- Una primera categorización de la clase de los atributos mediante el uso de K-Medias.
- Implementación de algoritmo de BFS, UCS, Dijkstra, A^*.
- Una segunda categorización utilizando la clasificación mediante Tablas de Decisión, la cual utiliza información en tiempo real ingresada por el usuario lo que dio paso a $A^*Mejorado$.
- Implementación $A^*Mejorado$, el cual es la fusión de A^* y la clasificación mediante Tablas de Decisión.
- Implementación de un ensamble baso en mezcla de expertos.
- Y finalmente con el prototipo de la plataforma de movilidad del STC Metro de la CDMX.

Lo cual se describe a continuación.

Estación	tipo	Horario	Promedio Afluencia Trimestral	Promedio Afluencia Trimestral / Anual	Clase de los atributos	Coordenadas
Pantitlan	subterráneo	2-0-1	4121204.50 3718295.67 3704706.83 4730863.50	1.37 1.34 1.29 1.57	medio	19.415359 -99.072132
Zaragoza	subterráneo	2-0-1	4499464.50 4155385.83 4286403.33 4301785	1.50 1.50 1.49 1.42	medio	19.412344 -99.08241
Gómez_Farias	subterráneo	2-0-1	3056072.83 2918453.67 2930699.50 2994471	1.02 1.05 1.02 0.99	bajo	19.416472 -99.09035
Blvd_Puerto_Aereo	subterráneo	2-0-1	2223748.83 2122113.17 2112119.83 2169223.83	0.74 0.77 0.73 0.72	bajo	19.41967 -99.09595
Balbuena	subterráneo	2-0-1	1161058.67 1070116.83 1115030.17 1156651.67	0.39 0.39 0.39 0.38	bajo	19.423231 -99.102302
Moctezuma	subterráneo	2-0-1	2006286.67 1843166.00 1972425.50 1992314.50	0.67 0.67 0.68 0.66	bajo	19.427218 -99.110305

Tabla 4.2: Mobidas-CDMX

4.2. Trabajo Realizado

Para el análisis de los datos que conforman Mobidas-CDMX, donde se buscó hacer uso del Aprendizaje Maquinal, Minería de Datos, Clasificadores de Datos para la extracción del conocimiento de Mobidas-CDMX, en combinación con Algoritmos de optimización.
Se realizó una primera categorización de las estaciones utilizando el algoritmo de K-Medias para obtener la primera versión de Mobidas-CDMX.

4.2.1. K-Medias

La técnica de análisis de clústeres llamada K-Medias, consiste en agrupar un conjunto de datos en un número establecido de clústeres o grupos, nuestro caso se agrupo en tres diferentes clústeres,estos se definieron de acuerdo al tipo de riesgo de contagio que pudiera representar una estación ante el Covid-19, este agrupamiento se realiza mediante la idea de distancia o similitud entre cada una de las observaciones utilizando lo que es la distancia euclidiana o la distancia de Manhattan.
Para la asignación de los clústeres iniciales se consideró un análisis estadístico para el tratamiento de datos y distribución de frecuencias para la agrupación de datos en categorías o clases[29].
Para el uso del método de K-Medias se agrupo el conjunto de observaciones en 3 clústeres como se observa en la Figura4.2, a partir de estas se fue buscando la distancia existente entre cada observación al centro (media) del clúster más cercano, K-medias itera en busca de los centroides de cada clúster para ver si hay una nueva asignación de una observación a un nuevo clúster, y el criterio de paro se dará cuando no exista ninguna nueva asignación, lo que

nos permitirá obtener la categorización de cada estación en alguno de los 3 clústeres .

En la implementación de K-Medias, se asignó para cada clúster un valor de 0 para bajo, 1

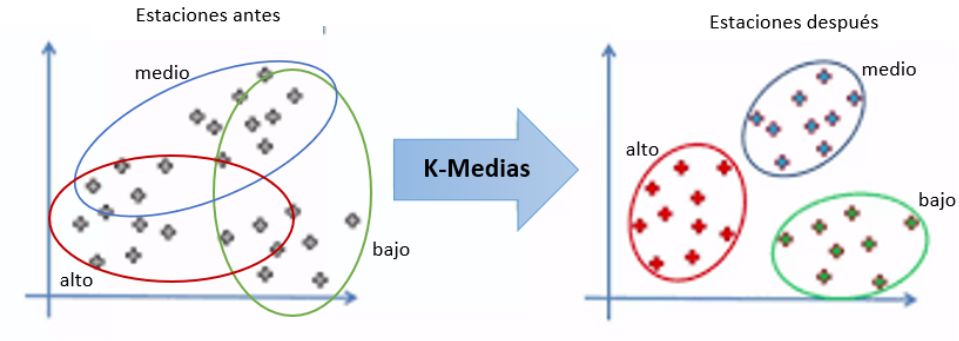


Figura 4.2: Categorización mediante k-Medias

para medio y 2 para alto, el método de K-Medias es el siguiente:

1. Se toma de manera aleatoria K centroides iniciales para tomar como referencia inicial.
2. Para cada una de las observaciones, se calcula la distancia de la observación a los centroides y se reasignan a los que estén más próximos. Una vez que se concluye esta reasignación, se vuelven a calcular los centroides de cada uno de los clústeres.
3. Se repiten los dos primeros pasos hasta que no se presente una nueva reasignación de alguna observación a uno de los K centroides.

Así obtenemos la primera versión de Mobidas-CDMX como se muestra en la Tabla4.3, la categorización correspondiente para la línea 1.

Estación	**Categorización 0:bajo, 1: medio, 2:alto**
Pantitlan	2
Zaragoza	1
Gómez_Farias	1
Blvd_Puerto_Aereo	0
Balbuena	1
Moctezuma	1
San_Lazaro	1
Candelaria	2
Merced	0
Pino_Suarez	1
Isabel_la_Catolica	1
Salto_del_Agua	2
Balderas	1
cuauhtemoc	2
Insurgentes	1
Sevilla	2
Chapultepec	0
Juanacatlan	0
Tacubaya	2
Observatorio	1

Tabla 4.3: Mobidas-CDMX con K-Medias, línea 1

Como podemos observar en la Tabla 4.3, se representó de forma gráfica cada una de las líneas del metro por medio de un árbol de decisión en el cual cada una de las estaciones representa cada uno de los nodos y las vías representan cada una de las ramificaciones.

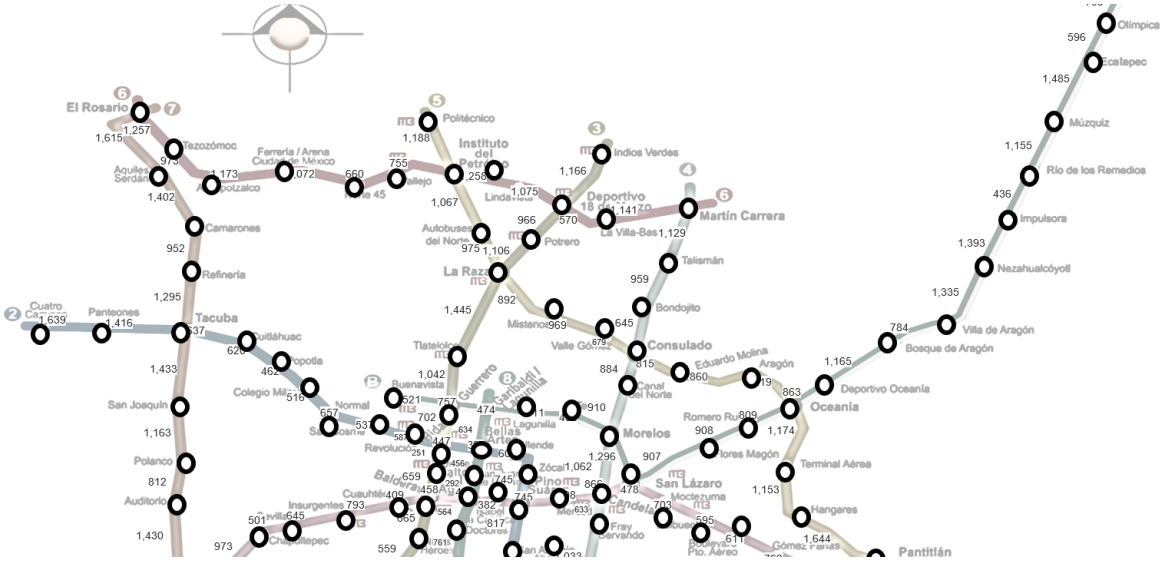


Figura 4.3: Representación gráfica del STC Metro

4.2.2. Algoritmos de Búsqueda Local

Para poder tener puntos de comparación se implementaron los algoritmos de búsqueda local de tipo:

- No informados como son:

  - Búsqueda en Anchura (en inglés BFS - Breadth First Search): Generamos un grafo a partir de una estación inicial el cual será nuestro nodo raíz y las estaciones con las que conecta las cuales serán los nodos vecinos y consecutivamente va visitando a cada uno de los nodos vecinos de donde este posicionado, hasta terminar de recorrer todo el grafo. Este algoritmo nos permite tener una ruta para llegar de una estación a otra. La característica de este algoritmo se encuentra en que no hay algún tipo de guía que nos permita decidir si una estación es mejor que otra.

    Para su implementación se utilizó una estructura de datos mediante la función hash para identificar cada una de las estaciones con las que conecta una estación teniendo como llave el nombre de cada una de las estaciones como se puede observar en la Figura 4.4.

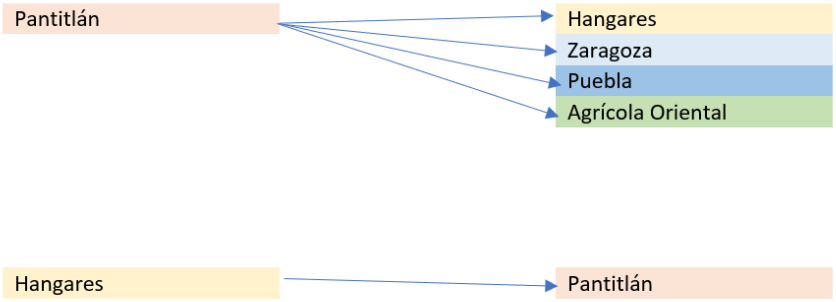


Figura 4.4: Representación de la tabla hash de las estaciones del STC Metro

- Búsqueda de Costo Uniforme (en inglés UCS - Search Cost Uniform): Sigue casi la misma dinámica que BFS pero la diferencia es que este algoritmo recorre el grafo con el camino de menor costo entre la estación inicial o nodo raíz y la estación final o nodo destino, donde $costo = distanciainterestacional$ entre dos estaciones. Inicia la búsqueda a partir de la estación inicial, visitando la siguiente estación con la cual conecta y que tenga un menor costo total desde la raíz. Este algoritmo nos permite tener una ruta para llegar de una estación a otra, eligiendo aquella estación que tenga menor distancia interestacional.

  Para su implementación se utilizaron dos estructuras de datos mediante la función hash para identificar cada una de las estaciones con las que conecta una estación teniendo como llave el nombre de cada una de las estaciones y otra para identificar las distancias interestacionales a cada estación teniendo como llave el nombre de cada una de las estaciones como se puede observar en la Figura 4.5.

- Informados
  - Dijkstra: Nos permitirá conocer el camino más corto a partir de una estación

Figura 4.5: Representación de la tabla hash de las estaciones del STC Metro y sus distancias interestacionales correspondientes

inicial a un destino, a diferencia de UCS y A^* donde cada nodo guarda la información de un coste, en este caso cada estación tendrá la siguiente estructura: $(distancia\_acumulada, padre)$ donde:
$distancia\_acumulada$: distancia mínima desde la estación inicial a cada una de las estaciones. $padre$: estación predecesora con el camino mínimo de la estación inicial al nodo en el que se esté posicionado Los pasos son los siguiente:

1. Visitar estación que no haya sido visitada y que tenga el menor valor de distancia acumulada
2. Sumar la distancia de las estaciones adyacentes con la distancia acumulada y guardar información con la estructura $(distancia\_acumulada, padre)$ en cada estación, si está ya contiene una etiqueta, se comparan las distancias acumuladas y se conserva la que tenga la distancia acumulada menor.
3. Se marca como estación visitada.
4. Se regresa a paso 1.

- A^*o: A^* es un algoritmo de búsqueda en grafos de tipo heurístico o informado, consiste al igual que BCU en buscar el camino con el menor costo, pero adicionando una función de evaluación $f(n) = h(n) + g(n)$ donde $h(n)$ corresponde a una función heurística la cual nos permite conocer que tan cerca estamos de la solución, usando como guía la información correspondiente a la afluencia de usuarios de cada una de las estaciones correspondiente al horario (horas pico y horas normales) en que el usuario lo este consultando, $g(n)$ corresponde al costo en cada estación desde el nodo raíz, en el cual utilizamos nuevamente la distancia interestacional de cada una de las estaciones, esto condicionara el costo de cada uno de los nodos.

4.2.3. A* Mejorado

$A^*Mejorado$ es un algoritmo en el cual se realizó una fusión entre el algoritmo de A^* y la Clasificación mediante Tablas de Decisión, como podemos observar en la Figura 4.6, la fusión entre A^* y la Clasificación mediante Tablas de Decisión, permite proporcionar mayor información a la función de evaluación $f(n) = h(n) + g(n)$.
Donde $g(n)$ corresponde al coste, el cual es la distancia interestacional entre dos estaciones y $h(n)$ corresponde a la función heurística, la cual hace uso de la información obtenida como: fecha y hora, edad, estaciones de inicio y final, los cuales se relacionan con los atributos de

Mobidas-CDMX.

Figura 4.6: Fusión de A* y Tablas de decisión

Se evalúa la información obtenida por la tabla de decisión y se obtiene como resultado la clasificación de una estación de acuerdo al tipo de riesgo que represente como: bajo, medio o alto, esta clasificación se utiliza por $A^*Mejorado$ para seleccionar la función heurística correspondiente al tipo de riesgo ante el Covid-19.

En la Tabla 4.4, se observa un ejemplo de los datos que se obtienen por un usuario, los cuales son utilizados por de $A^*Mejorado$ y las tablas de decisión, generando con esto una categoría que asigna para la estación Plantillan.

Estación	Horario	Trimestre	Afluencia	Categoría a la que pertenece
Pantitlán	33	3704706.83	1.37	alto

Tabla 4.4: Asignación de categoría en $A^*Mejorado$

Clasificación mediante Tablas de Decisión

La Clasificación Mediante Tablas de Decisión, es un algoritmo de Aprendizaje Maquinal, que hace uso de un algoritmo de conteo y agrupamiento para la generación de reglas de decisión[18].
Su implementación fue de la siguiente manera:

- A partir de nuestra base de datos generamos una tabla con cada uno de las posibles entradas que en esta se tienen y a partir de esta se genera una tabla candidata.
- Generación de tabla candidata: contiene todas las posibles entradas del punto anterior y además se le agregan los valores de $g-count$: número de tuplas equivalentes en cada renglón $c-count$: Número de tuplas del mismo tipo en D donde $D = n\acute{u}merodeestaciones$ y estos valores nos permitirán almacenar la información relacionada con el número de tuplas en el conjunto de entrenamiento, sup el cual es el soporte y está definido como: $\frac{g-count}{D}$ y $conf$ el cual corresponde a la confianza y está definido como $\frac{g-count}{c-count}$ y a partir de esta tabla candidata finalmente se forma una tabla de decisión,como podemos observan en la Figura4.7.

Estación	Tipo	Horario			Trimestre				Afluencia				Clase	g-count	c-count	sup	conf
Pantitlan	subterraneo	37	30	33	4121204.50	3718295.67	3704706.83	4730863.50	1.37	1.34	1.29	1.57	medio	1	1	0.005128	1.00000
Zaragoza	subterraneo	37	30	33	4499464.50	4155385.83	4286403.33	4301785.00	1.50	1.50	1.49	1.42	medio	1	1	0.005128	1.00000
Gomez_Farias	subterraneo	37	30	33	3056072.83	2918453.67	2930699.50	2994471.00	1.02	1.05	1.02	0.99	bajo	1	1	0.005128	1.00000
Boulevard_Pto_Aereo	subterraneo	37	30	33	2223748.83	2122113.17	2112119.83	2169223.83	0.74	0.77	0.73	0.72	bajo	1	1	0.005128	1.00000
Balbuena	subterraneo	37	30	33	1161058.67	1070116.83	1115030.17	1156651.67	0.39	0.39	0.39	0.38	bajo	1	1	0.005128	1.00000
Moctezuma	subterraneo	37	30	33	2006286.67	1843166.00	1972425.50	1992314.50	0.67	0.67	0.68	0.66	bajo	1	1	0.005128	1.00000
San_Lazaro	subterraneo	37	30	33	2807253.33	2566768.00	2697189.33	2756418.67	0.94	0.93	0.94	0.91	medio	1	1	0.005128	1.00000
Candelaria	subterraneo	37	30	33	2100574.33	1923334.83	2231077.00	2336646.83	0.70	0.70	0.77	0.77	medio	1	1	0.005128	1.00000
Merced	subterraneo	37	30	33	4329212.67	4016910.83	3902630.17	4841583.17	1.44	1.45	1.35	1.60	bajo	1	1	0.005128	1.00000
Pino_Suarez	subterraneo	37	30	33	2721212.67	2441129.83	2985545.50	3131031.83	0.91	0.88	1.04	1.04	medio	1	1	0.005128	1.00000
Isabel_la_Catolica	subterraneo	37	30	33	2053043.67	1832137.50	1951223.33	2012342.00	0.68	0.66	0.68	0.67	bajo	1	1	0.005128	1.00000
Salto_del_Agua	subterraneo	37	30	33	1978072.50	1788755.00	1865247.83	1872443.83	0.66	0.65	0.65	0.62	medio	1	1	0.005128	1.00000
Balderas	subterraneo	37	30	33	1952046.17	1669871.83	1815063.33	1863673.17	0.65	0.60	0.63	0.62	medio	1	1	0.005128	1.00000
Cuauhtemoc	subterraneo	37	30	33	1952438.17	1850650.17	1880711.33	1919010.33	0.65	0.67	0.65	0.63	bajo	1	1	0.005128	1.00000
Insurgentes	subterraneo	37	30	33	5061578.50	4679668.17	4782904.00	4904948.67	1.69	1.69	1.66	1.62	bajo	1	1	0.005128	1.00000
Sevilla	subterraneo	37	30	33	2746306.33	2585321.67	2702539.33	2663497.50	0.92	0.93	0.94	0.88	bajo	1	1	0.005128	1.00000
Chapultepec	subterraneo	37	30	33	4832659.83	4448110.83	4562187.83	4519933.17	1.61	1.61	1.58	1.50	bajo	1	1	0.005128	1.00000
Juanacatlan	subterraneo	37	30	33	1037772.83	926699.83	968455.83	983185.67	0.35	0.33	0.34	0.33	bajo	1	1	0.005128	1.00000
Tacubaya	subterraneo	37	30	33	3043233.50	2766628.83	2933329.17	3025691.17	1.01	1.00	1.02	1.00	bajo	1	1	0.005128	1.00000
Observatorio	superficial	37	30	33	6331476.83	6009034.00	6211822.00	6265972.67	2.11	2.17	2.16	2.07	medio	1	1	0.005128	1.00000

Figura 4.7: Tabla candidata inicial

- Generación de tabla de decisión:
  - De la tabla candidata se leen las tuplas que pertenezcan al mismo grupo (total de tuplas)
  - Se calcula el soporte sup y la confianza $conf$
  - Se definen los umbrales $minsup$ y $minconf$, nosotros los definimos $minsup > 0$ y $minconf > 0$

- Se descartan tuplas redundantes
- Continuar estos pasos hasta procesar todas las tuplas en la tabla candidata

Con esto generamos una tabla candidata final como la que se muestra en la Figura 4.8 y a partir de esta podemos clasificar una nueva entrada.

Estación	Tipo	Horario			Trimestre				Afluencia				Clase	g-count	c-count	sup	conf
Pantitlan	subterraneo	37	30	33	4121204.50	3718295.67	3704706.83	4730863.50	1.37	1.34	1.29	1.57	medio	1	1	0.005128	1.00000
Zaragoza	subterraneo	37	30	33	4499464.50	4155385.83	4286403.33	4301785.00	1.50	1.50	1.49	1.42	medio	1	1	0.005128	1.00000
Gomez_Farias	subterraneo	37	30	33	3056072.83	2918453.67	2930699.50	2994471.00	1.02	1.05	1.02	0.99	bajo	1	1	0.005128	1.00000
Boulevard_Pto_Aereo	subterraneo	37	30	33	2223748.83	2122113.17	2112119.83	2169223.83	0.74	0.77	0.73	0.72	bajo	1	1	0.005128	1.00000
Balbuena	subterraneo	37	30	33	1161058.67	1070116.83	1115030.17	1156651.67	0.39	0.39	0.39	0.38	bajo	1	1	0.005128	1.00000
Moctezuma	subterraneo	37	30	33	2006286.67	1843166.00	1972425.50	1992314.50	0.67	0.67	0.68	0.66	bajo	1	1	0.005128	1.00000
San_Lazaro	subterraneo	37	30	33	2807253.33	2566768.00	2697189.33	2756418.67	0.94	0.93	0.94	0.91	medio	1	1	0.005128	1.00000
Candelaria	subterraneo	37	30	33	2100574.33	1923334.83	2231077.00	2336646.83	0.70	0.70	0.77	0.77	medio	1	1	0.005128	1.00000
Merced	subterraneo	37	30	33	4329212.67	4016910.83	3902630.17	4841583.17	1.44	1.45	1.35	1.60	bajo	1	1	0.005128	1.00000
Pino_Suarez	subterraneo	37	30	33	2721212.67	2441129.83	2985545.50	3131031.83	0.91	0.88	1.04	1.04	medio	1	1	0.005128	1.00000
--																	
Pantitlan	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	medio	1	4	0.005128	0.25000
Zaragoza	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	medio	1	1	0.005128	1.00000
Gomez_Farias	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	bajo	1	1	0.005128	1.00000
Boulevard_Pto_Aereo	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	bajo	1	1	0.005128	1.00000
Balbuena	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	bajo	1	1	0.005128	1.00000
Moctezuma	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	bajo	1	1	0.005128	1.00000
San_Lazaro	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	medio	1	2	0.005128	0.50000
Candelaria	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	medio	2	2	0.010256	1.00000
Merced	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	bajo	1	1	0.005128	1.00000
Pino_Suarez	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	ANY	medio	2	2	0.010256	1.00000

Figura 4.8: Tabla candidata final

- Clasificación de una nueva instancia:
  - Se busca en la tabla de decisión el reglón que sea equivalente a la nueva instancia u, esto es que se busca el reglón que contenga los valores en sus atributos como ANY o sea equivalente de u

  En el caso que exista más de un renglón equivalente se siguen los siguientes pasos

  - asignar la clase del renglón donde $conf$ sea mayor, si hay empates asignar la clase donde sup sea mayor
  - Asignar la clase donde $conf * sup$ sea mayor
  - Si no se encuentra ningún reglón equivalente se puede utilizar el teorema de Bayes para calcular la probabilidad de cada clase.

En la Tabla 4.5 se muestra un ejemplo de clasificación de una nueva instancia mediante la tabla de decisión generada, esta corresponde a la estación Candelaria.

Con cada uno de los algoritmos descritos anteriormente, podemos obtener rutas con las siguientes características:

- Menor probabilidad de contagio, las cuales nos generara valores como alto, medio o bajo, y de acuerdo

Estación	tipo	Trimestre	Horario	Categoría a la que pertenece
Candelaria	subterráneo	1923334,83	30	alto
Clasificación obtenida				
Candelaria				medio

Tabla 4.5: Clasificación de una nueva instancia

- Menor tiempo de traslado, se obtiene mediante el cálculo las distancias internacionales o bien costo de una ruta generada.

4.2.4. Ensamble basado en Mezcla de Expertos

Con el ensamble basado en Mezcla de Expertos podemos combinar las diferentes características de cada uno de los algoritmos individualmente nos va proporcionando, como podemos observar en la Figura 4.9 para su construcción consideramos lo siguiente:

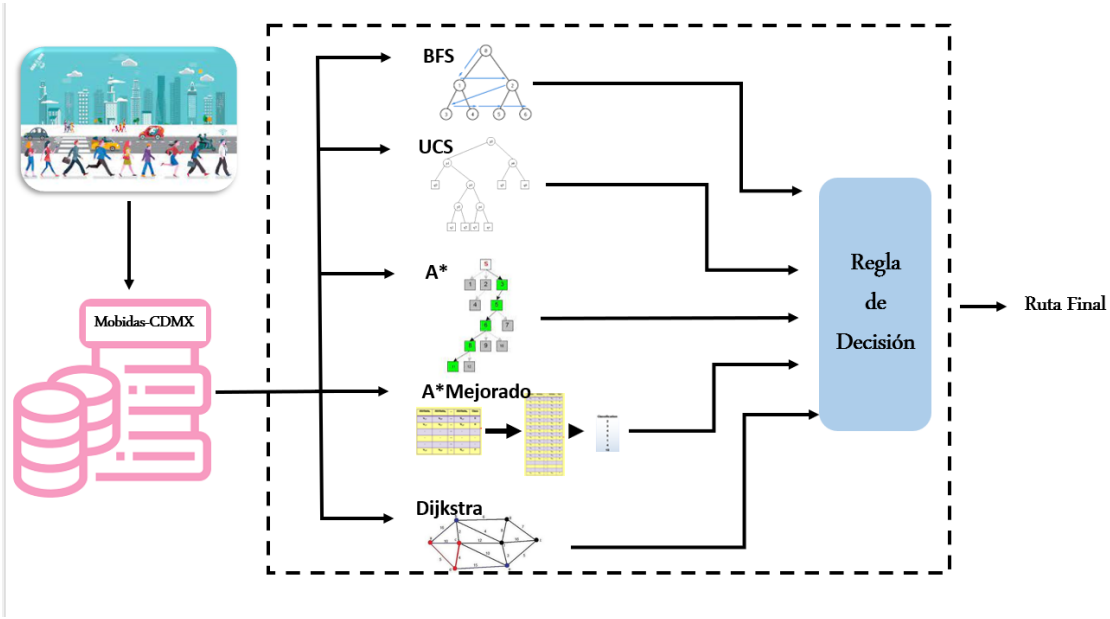


Figura 4.9: Ensamble basado en mezcla de expertos

- Conjunto de entrada: Corresponden a los datos de movilidad del STC Metro los cuales conforman Mobidas-CDMX

- Conjunto de algoritmos base los cuales son BFS, UCS, A^*, $A^*Mejorado$ y Dijkstra.

- Características a considerar para la elección del Algoritmo que sea el mejor candidato: Se consideran las rutas que al ser evaluadas de acuerdo con la regla de decisión proporcionen una menor probabilidad de contagio o menores tiempos de traslado.

- Regla de decisión: Se establecen las reglas de decisión bajo dos criterios:

    1. Distancia interestacional y distancia geodésica: Se evalúan cada una de las rutas generadas por cada algoritmo, y se comparan para seleccionar la que nos proporcione la menor distancia, lo cual está relacionado directamente con un menor tiempo de traslado el cual se calcula con el modelo de la velocidad $v = \frac{d}{t}$, en donde v equivale a una velocidad promedio de $24{,}9m/s$, considerando que se tienen las mejores condiciones en cuestión de afluencia de usuarios y velocidad de avance del convoy y d corresponde a la distancia interestacional o distancia geodésica total de una ruta generada.
    2. Probabilidades de contagio: Se evalúan cada una de las rutas generadas por cada algoritmo, y se comparan para seleccionar la que nos proporcione la menor probabilidad de contagio, esta es generada mediante el modelo de Daniel Bernoulli, en la cual propuso construir una fórmula que, bajo supuestos razonables, proporcione el número S de personas que no han tenido la viruela, de una edad x, en función de dicha edad y del número P de supervivientes, argumentando los siguiente:
    Los supervivientes, S, que no han tenido viruela decrecen por:
    (i) aquellos que cogen viruela (muriendo o no de ello) y,
    (ii) aquellos que mueren de otras causas sin haber tenido viruela alguna vez[31].

    Obteniendo el modelo epidemiológico de la viruela, que relaciona el número de personas con edad x susceptibles de ser infectadas $S(x)$ con el número de personas vivas con esa edad $P(x)$, la expresión a la que llegó fue: $\frac{S(x)}{P(x)} = \frac{1}{((1-p)exp(qx)+p)}$, para fines del desarrollo de este Proyecto de Investigación modificamos este modelo con los datos disponibles del STC Metro, para que nos permitiera calcular un valor representativo para la probabilidad de contagio (representado en porcentaje), esto se realizó de la siguiente manera:

    - Número de personas susceptibles de ser infectadas $S(x)\rightarrow$ afluencia total de una ruta
    - Número de personas vivas con esa edad $P(x) \rightarrow$ Promedio del total de la afluencia anual (histórico 5 años)
    - x edad $\rightarrow$ edad del usuario
    - p tasa de mortalidad$\rightarrow$ número de muertes[32]
    - q tasa de contagio$\rightarrow$número de personas contagiada confirmadas[32]

Observamos en la Figura 4.10 que una vez que se obtienen los valores para la probabilidad de contagio y se evalúa de acuerdo a la regla de decisión en las diferentes rutas, se

seleccionará aquella con el valor mínimo, esto es, se seleccionará la ruta que nos proporcione menor probabilidad de contagio.

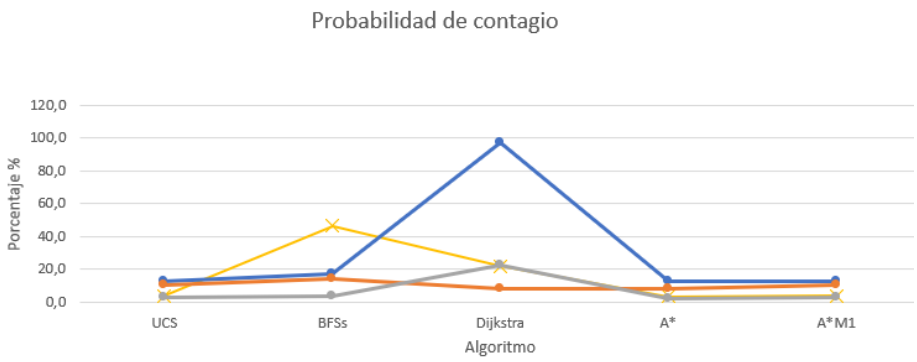


Figura 4.10: Probabilidad de Contagio en diferentes rutas

De forma similar como podemos observar en la Figura 4.11, se obtienen las distancias interestacionales y con ello el cálculo de los menores tiempos de traslado en cada una de las diferentes rutas, posteriormente se evalúa de acuerdo con la regla de decisión, y se selecciona la mejor ruta.

4.2.5. Prototipo de la plataforma de movilidad del STC Metro

Para el prototipo de la plataforma de movilidad del STC Metro se trabajó con dos versiones las cuales se describen a continuación:

- versión 1: Esta versión se realizó con cada uno de los algoritmos implementados (BFS, UCS, Dijkstra, A* y A*Mejorado)
  - Vistas: Se compone de dos ventanas diferentes:
    - Principal: En esta vista se pueden ingresar los datos de entrada y consultar los datos de salida, también los botones para borrar datos y consultar el mapa, como se observa en la Figura 4.12 .
    - Mapa: Como se puede observar en la Figura 4.13 en esta vista se puede consular el mapa del STC Metro.

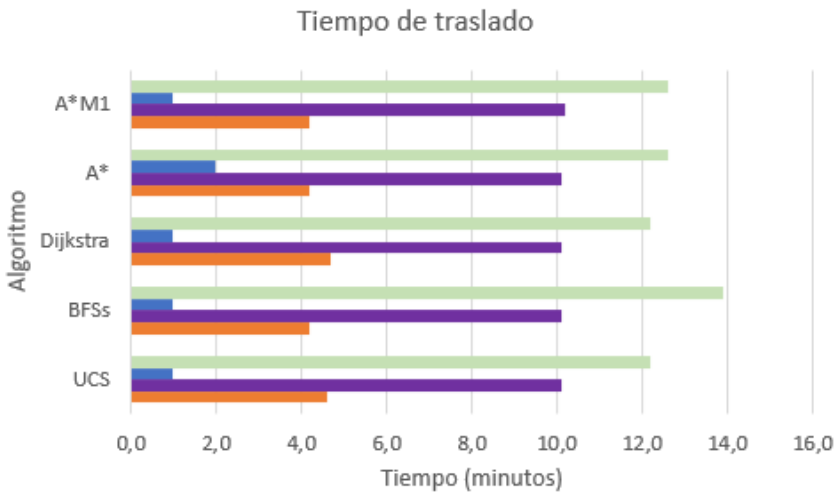


Figura 4.11: Tiempos de Traslado en diferentes rutas

- Datos de entrada: El usuario puede seleccionar una estación de inicio y una de destino, puede ingresar su edad y seleccionar que algoritmo sea el que le proporcione una ruta.
- Datos de salida: Se obtendrá una ruta con los datos de entrada proporcionados, además de un tiempo de traslado y una probabilidad de contagio correspondientes a la ruta obtenida.

▪ versión 2: Esta versión se realizó con el ensamble de mezcla de expertos.

- Vistas: Se compone de dos ventanas diferentes:
  - Principal: En esta vista se pueden ingresar los datos de entrada y consultar los datos de salida, también los botones para borrar datos y consultar el mapa como se observa en la Figura 4.14.
  - Mapa: Como se puede observar en la Figura 4.13 en esta vista se puede consular el mapa del STC Metro.
- Datos de entrada: El usuario puede seleccionar una estación de inicio y una de destino, puede ingresar su edad y seleccionar si desea una ruta que le proporcione menos tiempo de traslado o menor probabilidad de contagio.
- Datos de salida: Se obtendrá una ruta con los datos de entrada proporcionados, además de un tiempo de traslado y una probabilidad de contagio correspondientes a la ruta obtenida, distancia recorrida y tiempo de respuesta para generar la mejor ruta.

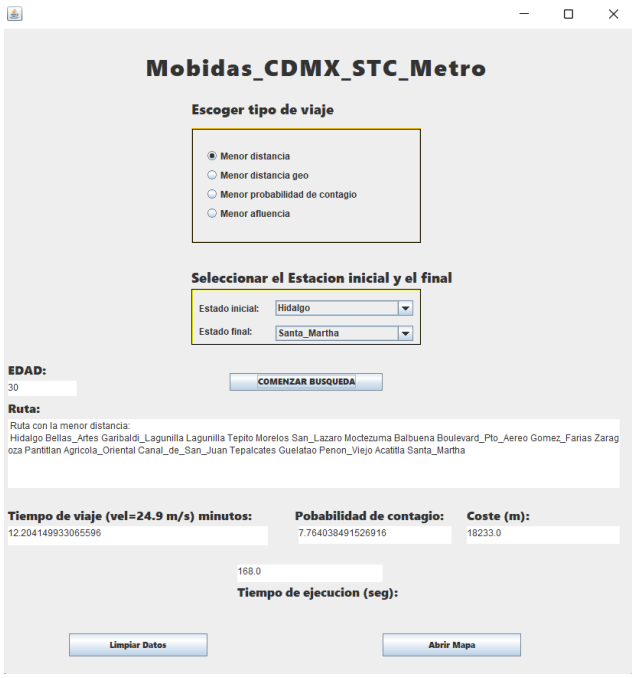


Figura 4.12: Vista del prototipo plataforma de movilidad v1

En la siguiente sección de experimentos y resultados podremos ver el resultado obtenido al evaluar diferentes rutas y las vistas del prototipo de la plataforma de movilidad del STC Metro.

Figura 4.13: Vista del mapa del STC Metro

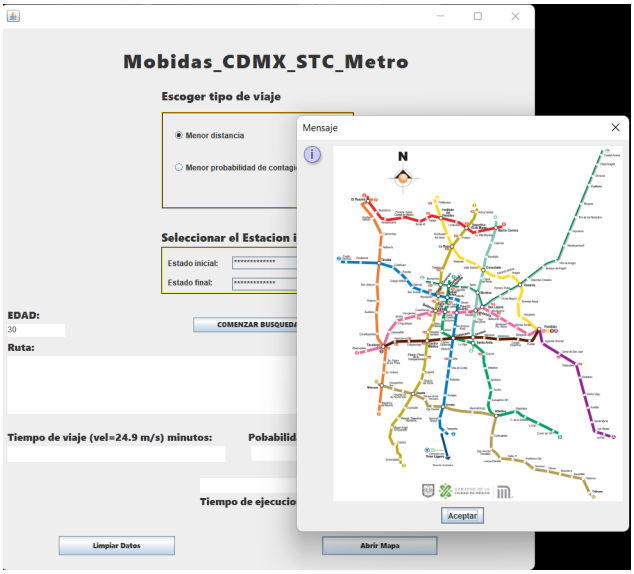


Figura 4.14: Vista del prototipo plataforma de movilidad v2

Capítulo 5

Experimentos y resultados

En este capítulo se presentan los datos obtenidos de la implementación de diferentes algoritmos, utilizados para la construcción del ensamble de mezcla de expertos para la Plataforma de Movilidad del STC Metro de la Ciudad de México.

5.1. Experimentos y resultados

En ésta sección mostraremos los experimentos y resultados del desarrollo del primer prototipo de la plataforma de optimización del STC Metro de la Ciudad de México, se trabajó con dos diferentes versiones v1 y v2, las cuales se describen más adelante, lo que nos permite dividir este capítulo en las siguientes secciones:

1. v1. Resultados obtenidos por cada uno de los algoritmos.

2. v2. Resultados obtenidos por el ensamble de mezcla de expertos.

3. Descripción del primer Prototipo de la plataforma de optimización del STC Metro de la Ciudad de México.

Se debe destacar que se consideró una velocidad promedio de 24,9 m/s y la fórmula que Daniel Bernoulli en 1760 formuló para el modelo epidemiológico de la viruela . Además de considerar tres diferentes edades 7, 25 y 30 años, esto con el propósito de poder representar la susceptibilidad por rango de edad que en su momento se observó en el Covid-19, los valores obtenidos son representados en porcentajes (%) y son valores representativos para poder visualizar con mayor claridad los resultados obtenidos de las probabilidades de contagio, en cuestión de los tiempos de traslado se consideran distancias interestacionales representadas en metros (m) y el tiempo representado en minutos (min), todos los valores resultantes se obtuvieron con los datos que componen Mobidas-CDMX.

5.1.1. v1. Resultados obtenidos por cada uno de los algoritmos

Para la primera versión de la plataforma de optimización se consideran como valores de entrada edad, la cual irá variando conforme un usuario lo ingrese, y las estaciones de inicio y destino, a partir de estos datos se genera una posible ruta de acuerdo al algoritmo seleccionado, en la Tabla 5.1, podemos observar los resultados obtenidos con las siguientes rutas, Hidalgo - Santa Martha, La Raza - Tláhuac, Pino Suárez - Chabacano, El Rosario - Zapata y San Juan de Letrán - San Pedro de los Pinos.

Probabilidad de contagio					
30 años					
Ruta	Hidalgo - Santa Martha	La Raza - Tláhuac	Pino Suárez - Chabacano	El Rosario - Zapata	San Juan de Letrán - San Pedro de los Pinos
Algoritmo	Probabilidad (%)				
BFS	10,8	19,8	15,5	8,6	6
UCS	7,8	16	15,5	8,6	7,3
Dijkstra	8,1	16	15,5	8,6	7,3
A^*	6,2	15	15,2	8,6	6
$A^* Mejorado$	12,2	15	10,2	7,7	6
25 años					
Ruta	Hidalgo - Santa Martha	La Raza - Tláhuac	Pino Suárez - Chabacano	El Rosario - Zapata	San Juan de Letrán - San Pedro de los Pinos
Algoritmo	Probabilidad (%)				
BFS	9	16,3	13	7,2	6,2
UCS	6,5	13,5	13	7,2	5
Dijkstra	6,5	13,5	13	7,2	6,2
A^*	5,3	12,5	12,7	7,2	5
$A^* Mejorado$	6,8	12,5	8,6	6	5
7 años					
Ruta	Hidalgo - Santa Martha	La Raza - Tláhuac	Pino Suárez - Chabacano	El Rosario - Zapata	San Juan de Letrán - San Pedro de los Pinos
Algoritmo	Probabilidad (%)				
BFS	2,5	4,6	3,7	2	1,7
UCS	2	3,8	3,7	2	1,4
Dijkstra	2	2,1	3,7	2	1,7
A^*	1,5	2,1	3,5	2	1,4
$A^* Mejorado$	1,5	3,5	3,2	1,7	1,3

Tabla 5.1: Comparación de probabilidades de contagio entre los diferentes algoritmos implementados con diferentes edades

De la Tabla anterior podemos notar lo siguiente:

1. Observamos que las probabilidades de contagio pueden variar sus valores con respecto a la edad del usuario, una característica que en su momento se vio en el Covid-19, esto se puede visualizar mejor en la Figura 5.1 donde se observan las probabilidades de contagio de la ruta Hidalgo - Santa Martha y en la Figura 5.2 podemos observar las probabilidades de contagio de la ruta La Raza Tláhuac, con las tres diferentes edades de prueba, 30, 25 y 7 años.

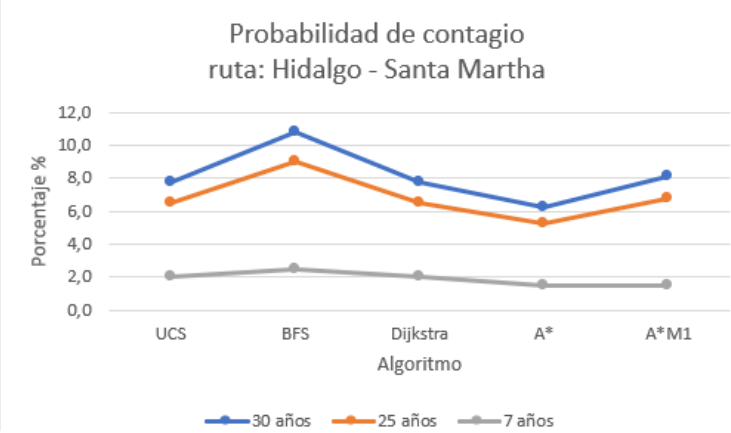


Figura 5.1: Probabilidad de contagio ruta: Hidalgo - Santa Martha

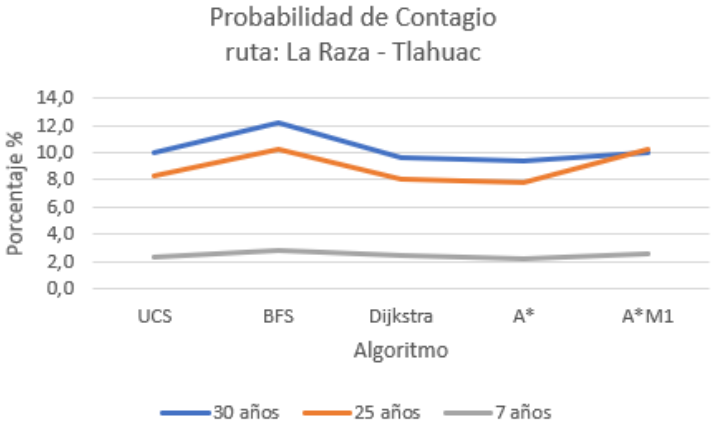


Figura 5.2: Probabilidad de contagio ruta: La Raza - Tláhuac

2. Las rutas obtenidas con el algoritmo de BFS el cual no tiene ningún tipo de guía para realizar el recorrido en la mayoría de las pruebas nos representara una mayor probabilidad de contagio.

3. Las rutas obtenidas por los algoritmos de UCS y Dijkstra los cuales son guiados por el costo (distancia interestacional) pueden llegar a tener valores similares

4. Las rutas obtenidas por los algoritmos de A^* y $A^*Mejorado$, los cuales son guiados por diferentes funciones de evaluación de acuerdo al tipo de riesgo que representa una ruta y los costos de esta, podemos notar que $A^*Mejorado$ nos proporciona en algunos casos menores o iguales probabilidades de contagio.

En el caso de los tiempos de traslado de acuerdo a los valores obtenidos durante las pruebas los cuales podemos observar en la Tabla 5.2, podemos resumir que el tiempo de traslado depende del algoritmo seleccionado, si bien entre los algoritmos como UCS o A^*, por ejemplo, notamos tiempos de traslado muy similares.

Menor tiempo de traslado					
Algoritmo	Hidalgo - Santa Martha	La Raza - Tláhuac	Pino Suárez - Chabacano	El Rosario - Zapata	San Juan de Letrán - San Pedro de los Pinos
Algoritmo	Tiempo (min)				
BFS	13,9	18	1	10,1	4,6
UCS	12,2	18	1	10,1	4,2
A^*	12,6	18	2	10,1	4,2
$A^*Mejorado$	12,2	18,5	1	10,1	7,9
Dijkstra	12,2	18	1	10,1	4,6

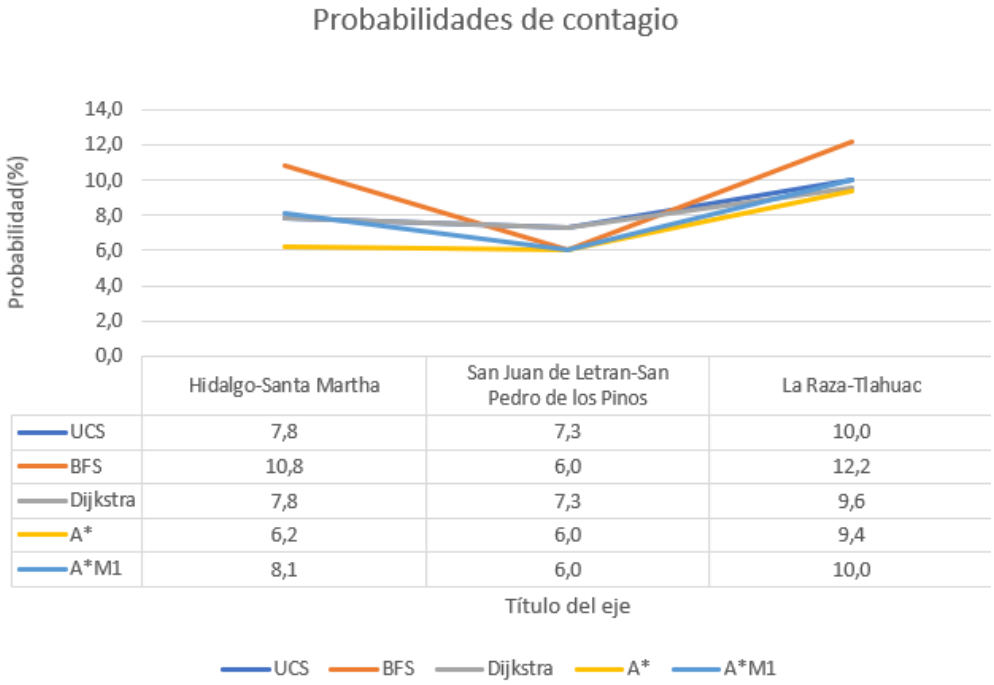
Tabla 5.2: Comparación de los tiempos de traslado entre los diferentes algoritmos implementados

Considerando los resultados obtenidos para la generación de rutas, podemos resumir que de los algoritmos implementados, no hubo alguno con mejores resultados en todas o en la mayoría de las pruebas. Lo que nos permitió dar paso a generar un ensamble de mezcla de expertos de búsqueda local para poder seleccionar la mejor ruta.

5.1.2. v2. Resultados obtenidos por el ensamble de mezcla de expertos

Para la segunda versión de la plataforma de optimización se consideran como valores de entrada edad, la cual irá variando conforme un usuario lo ingrese, y las estaciones de inicio y destino. Considerando las estaciones Hidalgo - Santa Martha, La Raza - Tláhuac, El Rosario - Zapata, Revolución - Tlaltenco, y San Juan de Letrán - San Pedro de los Pinos.

Las probabilidades de contagio obtenidas por el ensamble de mezcla de expertos de búsqueda local, al evaluar cada una de las rutas, las podemos observar en la Figura 5.3, notamos que para la ruta Hidalgo - Santa Martha, de los resultado obtenidos, se seleccionará el algoritmo con menor probabilidad de contagio, el cual es A^*, en la ruta La Raza - Tláhuac se obtiene la ruta generada por A^*, y para la ruta de San Juan de Letrán - San Pedro de los Pinos podemos destacar que se tienen tres rutas con una probabilidad de contagio similar por lo que se selecciona la ruta generada por BFS, como se muestra en la Tabla 5.3.



	Hidalgo-Santa Martha	San Juan de Letran-San Pedro de los Pinos	La Raza-Tlahuac
UCS	7,8	7,3	10,0
BFS	10,8	6,0	12,2
Dijkstra	7,8	7,3	9,6
A*	6,2	6,0	9,4
A*M1	8,1	6,0	10,0

Figura 5.3: Comparación de la probabilidad de contagio obtenidos en el ensamble de mezcla de expertos

Menor Probabilidad de Contagio					
Ruta	Hidalgo - Santa Martha	La Raza - Tláhuac	El Rosario-Zapata	Revolución-Tlaltenco	San Juan de Letrán - San Pedro de los Pinos
Algoritmo	A^*	A^*	$A^*Mejorado$	Dijkstra	BFS
Probabilidad (%)	6,2	9,4	7,7	8,5	6

Tabla 5.3: Comparación diferentes Algoritmos obtenidos por el ensamble de mezcla de expertos

De forma similar se evalúa cada una de la rutas y de acuerdo a la regla de decisión para la selección de rutas con menores tiempo de traslado podemos observar en la Tabla 5.4 los resultados obtenidos, que si bien comparamos con los resultados de la Tabla 5.3, podemos notar que los algoritmos seleccionados no son los mismos.

Tiempo de Traslado					
Ruta	Hidalgo - Santa Martha	La Raza - Tláhuac	El Rosario-Zapata	Revolución-Tlaltenco	San Juan de Letrán - San Pedro de los Pinos
Algoritmo	UCS	BFS	Dijkstra	UCS	Dijkstra
Tiempo(min)	12,2	18	10,1	15,9	42

Tabla 5.4: Comparación entre los diferentes Algoritmos

Otro dato obtenido de igual manera importante son los tiempos de ejecución generado por el ensamble de mezcla de expertos, en la Tabla 5.5, podemos destacar que el ensamble selecciona la ruta optima en tiempos menores de 3.5 minutos, lo cual puede considerarse elevado en cuestiones de espera para el usuario.

Tiempo (minutos)					
Ruta	Hidalgo - Santa Martha	La Raza - Tláhuac	El Rosario-Zapata	Revolución-Tlaltenco	San Juan de Letrán - San Pedro de los Pinos
Menor probabilidad de contagio	2.8	0.9	1.1	2.55	3.2
Menor tiempo de traslado	2.8	0.9	1.1	2.55	3.2

Tabla 5.5: Tiempos de Ejecución del ensamble de mezcla de expertos

5.1.3. Descripción del primer Prototipo de la plataforma de optimización del STC Metro de la Ciudad de México

Se realizó la primera versión la cual podemos observar en la Figura 5.4 la cual nos permita seleccionar que algoritmos generan la ruta las cuales son:

- UCS
- BFS
- Dijkstra
- A^*
- A^*M

El usuario puede ingresar su edad y seleccionar la estación de inicio y destino, con esa información se genera ruta, además de mostrar tiempo de traslado y la probabilidad de contagio que representara la ruta generada.

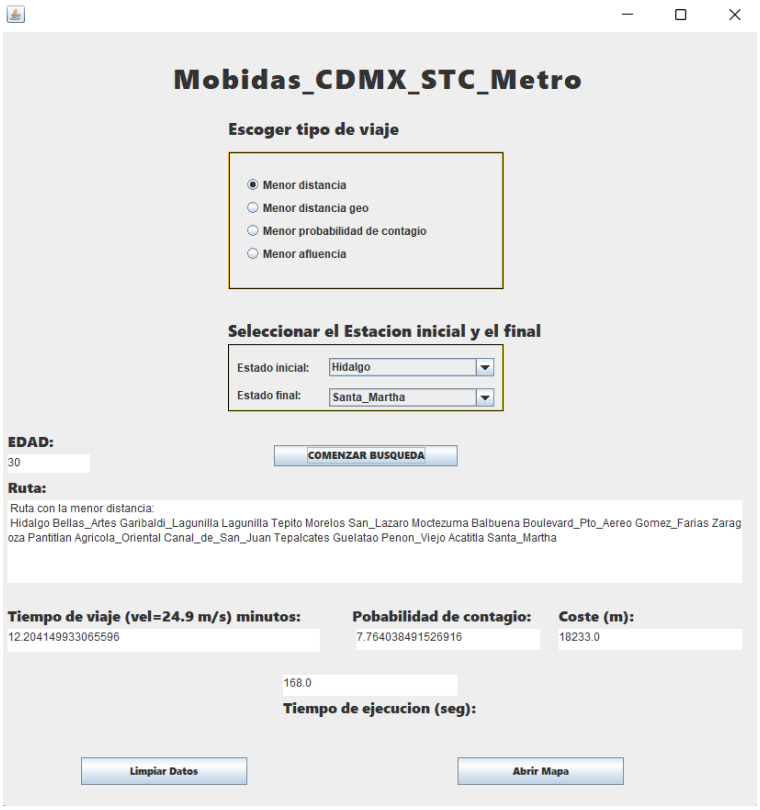


Figura 5.4: Prototipo plataforma de movilidad v1

La segunda versión la cual podemos observar en la Figura 5.5 se adaptó para el ensamble de mezcla de expertos en el cual permite al usuario seleccionar entre las opciones:

- Menor distancia
- Menor probabilidad de contagio.

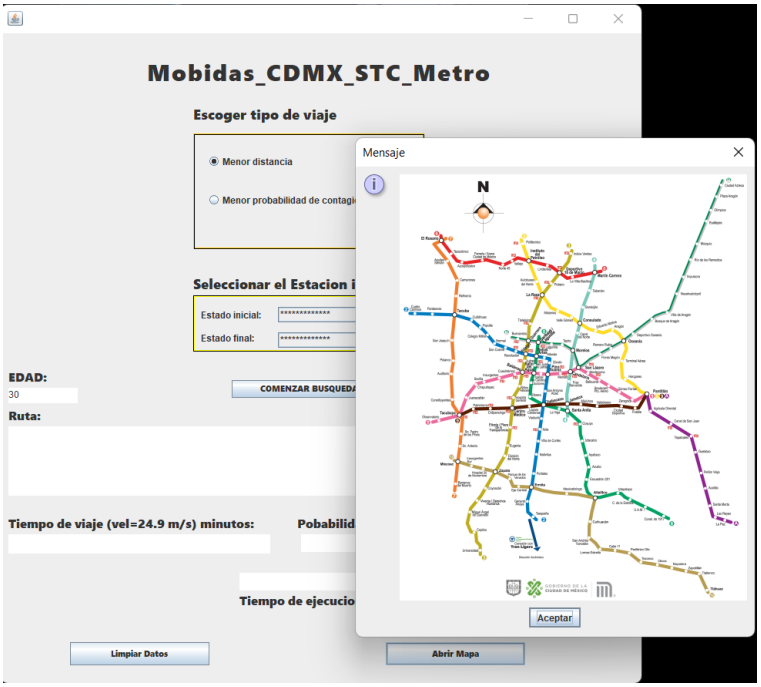


Figura 5.5: Prototipo plataforma de movilidad v2

El usuario puede ingresar su edad y seleccionar la estación de inicio y destino, con esa información se genera la mejor ruta y se muestra el tiempo de traslado, distancia recorrida y la probabilidad de contagio que representara la ruta generada.

Capítulo 6

Conclusiones y trabajo a futuro

6.1. Conclusiones

Pudimos observar durante las pruebas realizadas que los criterios de optimización en los diferentes algoritmos implementados, no fueron compatibles de manera simultánea para la obtención de las mejores rutas con menor probabilidad de contagio y las rutas con menores tiempos de traslado. Con respecto al ensamble de mezcla de expertos se observo que con las ponderaciones establecidas se permite seleccionar la mejor ruta de entre en conjunto de los diferentes algoritmos.
En cuestión de los Algoritmos de Optimización Bioinspirada, de acuerdo a los atributos que componen Mobidas-CDMX no fueron suficiente para poder proporcionar mayores criterios de selección entre las estaciones para la generación rutas con mejores resultados a los que se obtuvieron con los algoritmos de búsqueda local, por lo que se propone en la siguiente sección utilizarlos en trabajos a futuro.

6.1.1. Aportaciones

En el desarrollo de este Trabajo de Investigación se obtuvieron las siguientes aportaciones:

- Mobidas-CDMX.
- Algoritmo $A^*Mejorado$.
- Primer prototipo de la plataforma de Optimizacion del STC Metro dela CDMX.

6.2. Trabajo a futuro

Como trabajo a futuro se propone, agregar más atributos a Mobidas-CDMX que puedan ser analizados mediante algoritmos del Aprendizaje Maquinal y Algoritmos de Optimización, un ejemplo de posibles atributos a agregar son los siguientes:

- Clima.
- Consumo energético.
- Conformación de los carros.
- Tipo de infraestructura.
- distancia de trasbordos.
- Tarjetas de pago a acceso al STC Metro (y otros medios de transporte publico).

Con los atributos como el consumo energético o tarjetas de pago se propone implementar Algoritmos de Optimización Bioinspirada buscando un menor consumo energético u optimizar los viajes de un usuario con respecto a minimizar el gasto realizado con las tarjetas de pago.

También se considera establecer nuevos modelos para el análisis de las probabilidades de contagio e incluir más medios de transporte público a Mobidas-CDMX ya que en este Proyecto de Investigación se acoto la base de datos a atributos exclusivamente del STC Metro y finalmente generar la plataforma de Optimización de la CDMX la cual incluya la mayoría de los sistemas de transporte públicos.

Bibliografía

[1] R. A. Española. movilidad, 2022.

[2] CONUEE. Movilidad urbana sostenible, 2018.

[3] SEMOVI. Plan estrategico de movilidad de la ciudad de méxico 2019, 2019.

[4] HITRANS. The transport strategy for the highlands and islands, 2011.

[5] G. de la Ciudad de México. Ley de movilidad del distrito federal, 2017.

[6] SEMOVI. Segundo informe anual, 2020.

[7] Oracle. Base de datos relacionales, 2022.

[8] S. T. Konstantinos. Pattern recognition, 2019.

[9] S. G. D. PEREZ LOPEZ, CESAR. Minería de datos. técnicas y herramientas: técnicas y herramientas, 2018.

[10] Aprendemachinelearning. K-nn, 2018.

[11] ieee. Paraltabs: A parallel scheme of decision tables construction, 2013.

[12] A. M. Savina, B. J. Gabriela Lucia, G. F. Melina, R. M. Cristobal Julian. Inteligencia Artificial(Búsqueda A Estrella), 2019

[13] A. R. K. S. Marc Sevaux, Philippe Coussy, 2010.

[14] A. B. Reyes. Anexo a: Descripción de simuladores de tráfico, 2021.

[15] L. Carballo, A. Villagra, M. Errecalde. Movilidad inteligente: Reducción de emisión de gases, 2019.

[16] N. Lathia. Mining mobility data to minimise travellers'spending on public transport,2011.

[17]B. L. TAKEYAS. Algoritmo c4.5, 2015.

[18] S. Foell. Catch me if you can: predicting mobility patterns of public transport users,2014.

[19] R. M.-K. R. Benjamin Moreno Montiel. Paraltabs: A parallel scheme of decision tables construction, 2013.

[20] R. M.-K. R. Benjamin Moreno Montiel. Parallel classification system based on an ensemble of mixture of experts,2014.

[21] R. M.-K. R. Benjamin Moreno Montiel. A hybrid classifier with genetic weighting, 2011.
[22] M. CDMX. Cronologia del metro cdmx, 2022.

[23] M. CDMX. Estaciones por uso y tipo, 2022.

[24] M. CDMX. Longitud lineas, 2022.

[25] M. CDMX. Indicadores de operacion, 2022.

[26] Metro. Estaciones con mayor afluencia, 2022.

[27] M. CDMX. Afluencia por estacion, 2022.

[28] M. CDMX. Longitud lineas, 2022.

[29] M. CDMX. Conformacion de los carros, 2022.

[30] F. . M.Guitart. Estadistica descriptiva y analisis de datos, 2022.

[31] F. J. O. I. José Antonio Camúñez Ruiz, Jesús Basulto Santos. La memoria de daniel bernoulli sobre lainoculación contra la viruela (1760):un problema de decision bajo incertidumbr, 2002. [32] CONACYT COVID-19 Tablero México, 2022.

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA

ACTA DE EXAMEN DE GRADO

No. 00098
Matrícula: 2192802612

Plataforma de optimización de la movilidad del STC Metro de la Ciudad de México mediante un ensamble de búsqueda local

En la Ciudad de México, se presentaron a las 15:00 horas del dia 5 del mes de agosto del año 2022 en la Unidad Iztapalapa de la Universidad Autónoma Metropolitana, los suscritos miembros del jurado:

DRA. MARIA ELENA LARRAGA RAMIREZ
DR. EDUARDO RODRIGUEZ FLORES
ING. LUIS FERNANDO CASTRO CAREAGA

Bajo la Presidencia de la primera y con carácter de Secretario el último, se reunieron para proceder al Examen de Grado cuya denominación aparece al margen, para la obtención del grado de:

MAESTRA EN CIENCIAS (CIENCIAS Y TECNOLOGIAS DE LA INFORMACION)

DE: DIANA ANTONIA MARTINEZ SANCHEZ

y de acuerdo con el articulo 78 fracción III del Reglamento de Estudios Superiores de la Universidad Autónoma Metropolitana, los miembros del jurado resolvieron:

Aprobar

Acto continuo, la presidenta del jurado comunicó a la interesada el resultado de la evaluación y, en caso aprobatorio, le fue tomada la protesta.

DIANA ANTONIA MARTINEZ SANCHEZ
ALUMNA

REVISÓ

MTRA. ROSALIA SERRANO DE LA PAZ
DIRECTORA DE SISTEMAS ESCOLARES

DIRECTOR DE LA DIVISIÓN DE CBI

Román Linares Romero
DR. ROMAN LINARES ROMERO

PRESIDENTA

DRA. MARIA ELENA LARRAGA RAMIREZ

VOCAL

DR. EDUARDO RODRIGUEZ FLORES

SECRETARIO

ING. LUIS FERNANDO CASTRO CAREAGA