



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

MODELO DE FLUJO DE TRÁFICO DE QUITO USANDO ALGORITMOS
DE OPTIMIZACIÓN EN REDES COMPLEJAS

AUTORES

Alexander Josué Jaramillo Fuelantala

Kevin Rafael Soto Espinosa

AÑO

2020



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

MODELO DE FLUJO DE TRÁFICO DE QUITO USANDO ALGORITMOS DE
OPTIMIZACIÓN EN REDES COMPLEJAS

Trabajo de Titulación presentado en conformidad con los requisitos establecidos
para optar por el título de Ingeniero en Sistemas de Computación e Informática.

Profesor Guía

PhD. Mario Salvador González Rodríguez

Autores

Alexander Josué Jaramillo Fuelantala

Kevin Rafael Soto Espinosa

Año

2020

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo, Modelo de flujo de tráfico de Quito usando algoritmos de optimización en redes complejas, a través de reuniones periódicas con los estudiantes Alexander Josué Jaramillo Fuelantala, Kevin Rafael Soto Espinosa, en el semestre 202020, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación”.



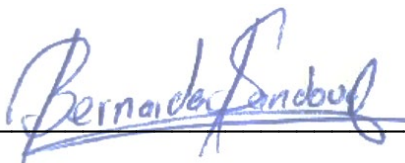
Mario Salvador González Rodríguez

Doctor en Ingeniería Informática y Telecomunicación

CI: 0958376345

DECLARACIÓN DEL PROFESOR CORRECTOR

“Declaro haber revisado este trabajo, Modelo de flujo de tráfico de Quito usando algoritmos de optimización en redes complejas, de los estudiantes Alexander Josué Jaramillo Fuelantala y Kevin Rafael Soto Espinosa, en el semestre 202020, dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación”.



Bernarda Cecibel Sandoval Romo

Mestra en Ciencia da Computacao

CI: 1709974453

DECLARACIÓN DE AUTORIA DE LOS ESTUDIANTES

“Declaramos que este trabajo es original, de nuestra autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetan las disposiciones legales que protegen los derechos de autor vigentes”.



Alexander Josué Jaramillo

Fuelantala

CI: 1726232885



Kevin Rafael Soto Espinosa

CI: 23005365

AGRADECIMIENTOS

Agradezco a mi familia por siempre brindarme el apoyo necesario para seguir adelante y superar cualquier adversidad, a mis profesores, quienes me inculcaron el conocimiento necesario para desempeñarme correctamente en las diversas áreas de la profesión, a mis amigos, con quienes he compartido momentos amenos e inolvidables durante la carrera, y mi más emotivo agradecimiento para mi madre, Soraya Fuelantala, ya que sin ella todo esto no hubiera sido posible.

Josué Jaramillo

DEDICATORIA

Dedico este trabajo a mi familia, especialmente a mi madre y a mi tío, por ayudarme siempre a continuar con mis estudios. A mi tutor de tesis Mario Gonzáles y a Jean Michel Clairand, por guiarnos durante este proceso. A mi novia Heydi Cruz, y a mis amigos, Kevin Soto, Sebastián Manzano, Steward Solano, Paúl Villacís, Estéfano Rodríguez, Juan Landines, Solange Bustos, Pedro Nogales, por su amistad durante este trayecto y cada momento compartido.

Josué Jaramillo

AGRADECIMIENTO

Agradezco a mi familia por el apoyo que se me ha brindado durante todos estos años, a mis profesores, los cuales me han impartido sus conocimientos y me han enseñado como desempeñar varias actividades correspondientes con la profesión, a mis amigos, con los cuales he pasado grandes momentos durante toda la carrera, por último, mi más grande agradecimiento a mi madre, la cual ha sido un pilar fundamental durante toda mi vida y me ha educado para ser el hombre que soy ahora.

Kevin Soto

DEDICATORIA

Dedico este trabajo a mi familia, en especial a mi madre y mi abuela, las cuales han sido ejemplo de vida. A mi tutor de tesis Mario Gonzáles y a Jean Michel Clairand, por guiarnos durante este proceso. A mis amigos, Sebastián Manzano, Steward Solano, Paúl Villacís, Estéfano Rodríguez, Juan Landines, Solange Bustos, a mi compañero de proyecto Josué Jaramillo, por brindarme su amistad incondicional todo el tiempo y por todos los momentos que hemos vivido juntos

Kevin Soto

RESUMEN

Con los desafíos ambientales globales y el crecimiento de las áreas urbanas, ha surgido la idea de ciudades sostenibles, incluido el transporte sostenible. Los principales problemas de transporte incluyen la contaminación del aire, el ruido y el tráfico, por mencionar algunos. Para abordarlos, es fundamental recopilar información importante sobre el tráfico y las condiciones de las carreteras de las ciudades. Por lo tanto, este trabajo presenta un enfoque de modelado de red que identifica los puntos extremos de la ciudad, a través de distintos métodos y algoritmos de recorte de redes complejas, conjuntamente con el mapeo del flujo de tráfico entre ellos, con la finalidad de agruparlos según su proximidad geográfica y obtener los rangos de tiempo correspondientes a los horarios con mayor afluencia de tráfico dentro de la ciudad, siendo esto la principal información de entrada para realizar e implementar planes de ciudad sostenible en la ciudad de Quito.

Palabras Clave: Contaminación, urbanas, transporte, sostenibles, Quito.

ABSTRACT

With global environmental challenges and the growth of urban areas, the idea of sustainable cities has emerged, including sustainable transport. The main transportation problems include air pollution, noise and traffic, to name a few. In order to address them, it is essential to gather important information on the traffic and road conditions of cities. Therefore, this work presents a network modeling approach that identifies the extreme points of the city, through different methods and algorithms for trimming complex networks, together with the mapping of the traffic flow between them, in order to group them according to their geographical proximity and obtain the time ranges corresponding to the schedules with the highest traffic flow within the city, this being the main input information to carry out and implement sustainable city plans in the city of Quito.

Keywords: Pollution, urban, transportation, sustainable, Quito.

ÍNDICE

1. INTRODUCCIÓN.....	1
1.1. Antecedentes.....	3
1.2. Alcance.....	5
1.3. Justificación.....	6
1.4. Objetivo General.....	7
1.5. Objetivos Específicos.....	7
1.6. Metodología.....	8
2. MARCO TEÓRICO.....	8
2.1. Conceptos Generales.....	9
2.1.1. Data Science.....	9
2.1.2. Teoría de Grafos.....	9
2.1.3. Multigrafos.....	10
2.1.4. Nodos.....	10
2.1.5. Análisis mediante clústeres.....	11
2.1.6. Sistema Geodésico Mundial de Coordenadas.....	11
2.1.7. Distancia euclidiana.....	12
2.1.8. Agrupamiento jerárquico.....	12
2.1.9. Método de Ward.....	13
2.2. Mecanismos de simplificación.....	13
2.2.1. Media Aritmética.....	13
2.2.2. Percentil.....	14
2.2.3. Algoritmo de Dijkstra.....	14

2.3. Herramientas utilizadas.....	15
2.3.1. OSMnx.....	15
2.3.2. Open Street Map	16
2.3.3. Google Maps	16
2.3.4. Directions API.....	17
2.3.5. Distance Matrix API	17
2.3.6. Leaflet.....	17
2.3.7. Folium.....	18
2.3.8. Anaconda Navigator	18
2.3.9. Jupyter Notebook.....	19
2.3.10. Inkscape	20
2.4. Lenguajes de programación y formatos.....	21
2.4.1. Lenguaje Python.....	21
2.4.2. JavaScript.....	22
2.4.3. JSON Geográfico.....	22
2.4.4. Formato SVG.....	23
3. DESARROLLO DE LA SOLUCIÓN.....	24
3.1. Flujo de procesos	24
3.2. Extracción de la red	25
3.3. Simplificación de la red	26
3.4. Recorte de red por distancia.....	30
3.5. Winner-Takes-All	31
3.6. Extracción del flujo de tráfico.....	34
3.7. Clustering y mapeo del flujo de tráfico	37

4. RESULTADOS	38
4.1. Grafo resultante mediante OSMnx.....	38
4.2. Red resultante de la simplificación aritmética y Winner Takes All	42
4.3. Información en tiempo real.....	44
4.4. Perfilado.....	46
4.4.1. Representación Gráfica del perfilado.....	48
4.5. Clústeres Geográficos.....	49
4.6. Análisis entre clústeres	51
5. CONCLUSIONES Y RECOMENDACIONES	58
5.1. Conclusiones.....	58
5.2. Recomendaciones.....	59
REFERENCIAS	61

1. INTRODUCCIÓN

Recientemente, las ciudades sostenibles han atraído mucha atención entre los estudiosos urbanos, los planificadores y los encargados de formular políticas y, sobre todo, han llamado la atención de diferentes campos de la ingeniería (Bibri & Krogstie, 2017). Esto incluye el desarrollo de nuevas estrategias para sistemas de transporte sostenibles. Hasta ahora, los habitantes de las grandes ciudades experimentan problemas como el tiempo dedicado y los efectos adversos para la salud debido a la contaminación y el ruido de las condiciones de tráfico congestionadas. Por lo tanto, el principal desafío del transporte sostenible es poner en práctica esta idea en términos de congestión y contaminación para la movilidad urbana (Beaudoin et al. 2015).

Estos desafíos pueden abordarse mediante la implementación de nuevas políticas de transporte. Muchos gobiernos han impuesto restricciones a los propietarios de automóviles en función de los números de placas, lo que conduce a la reducción de la congestión del tráfico y el aire más limpio. Sin embargo, este tipo de estrategias producen gran incomodidad a los propietarios de automóviles y los impulsan de manera pasiva a la idea de adquirir nuevos vehículos para poder evadir estas restricciones y circular todos los días. Por lo tanto, se deben proponer nuevas estrategias. La infraestructura de transporte urbano está sujeta a cambios constantes, es necesario optimizar los nuevos recursos disponibles (Valenzuela-Levi, 2018).

Sin embargo, el principal problema que representan estos cambios es la falta de un sistema de transporte basado en áreas de importancia dentro de las zonas urbanas. Este tipo de sistemas se suelen tratar de realizar manualmente, mediante estudios largos y tediosos, donde la etapa de planificación se llega a demorar demasiado y deja el proyecto inservible ante un cambio en la topología o conexiones entre calles.

Generalmente los países del tercer mundo, como Ecuador, tienen capitales y grandes ciudades que en perspectiva se encuentran listas para aplicar un plan que permita tener una ciudad sostenible, sin embargo, la mala administración de los recursos hace de esta tarea algo difícil, ya que no se desea invertir en el personal necesario para realizar un estudio a fondo que pueda establecer los puntos de interés sobre los cuales se realizará el plan urbano.

Gestión de la calidad del aire es uno de los temas menos trabajado dentro de la gestión pública en el Ecuador, al punto que ni siquiera existe información objetiva respecto a la magnitud del problema, pese a que existen indicios de que las ciudades grandes y medianas están expuestas a la contaminación del aire, vista casi como un impacto indeseable e inevitable del progreso. Sin embargo, en este panorama se destaca el esfuerzo pionero emprendido por el Municipio del Distrito Metropolitano de Quito, a través de CORPAIRE. (Pérez, 2009).

La capital de Ecuador, Quito, es una de las primeras ciudades del país en preocuparse por tratar de evitar la contaminación, sin embargo, las disposiciones establecidas para reducir el tráfico no evitan del todo la contaminación a gran escala, esto debido a que, al igual que en otros países, las personas optaron por comprar más vehículos, en lugar de acatar las restricciones vehiculares. Es por esto por lo que la solución óptima sería lograr un plan de tráfico, el cual determine los puntos estratégicos de la ciudad para a través de estos conseguir implementar una solución como la ubicación de puntos de carga para autos eléctricos u otros medios que motiven a los ciudadanos a utilizar medios de transporte que no produzcan altos niveles de contaminación, mejorando así la calidad de vida en general dentro de la zona metropolitana de la ciudad.

La ciudad ya existente con sus propias normativas previamente establecidas, frente a la construcción de una nueva ciudad capaz de reformar la manera de

transportarse, constituye un reto fundamental de las sociedades actuales. Es importante establecer marcos institucionales y poner en práctica nuevas políticas que eviten cometer los mismos errores del pasado. (Higueras, 2009).

El presente trabajo busca optimizar este proceso de investigación, mediante el tratamiento de la topología de la ciudad a manera de red compleja, permitiendo así el uso de algoritmos basados en estructuras de datos como nodos, árboles, y distintos tipos de grafos, para obtener los puntos de importancia en base a su ubicación y conexión con toda la ciudad, relacionándolos entre sí según el criterio deseado y estableciendo las zonas óptimas para la implementación de un plan de ciudad sostenible dentro del Distrito Metropolitano de Quito.

1.1. Antecedentes

Análisis de flujo de tráfico (*Traffic Flow Analysis*) es un término utilizado por investigadores para referirse a la necesidad de realizar un modelo estadístico el cual describe en una serie de tiempo un comportamiento específico de vehículos en un área determinada. (Williams & Hoel, 1999).

El tipo de análisis mencionado ha sido objetivo de varios estudios alrededor del mundo, siempre en busca de la mejora en conjunto de la sociedad mediante el establecimiento de políticas que administren la forma en la que las personas circulan por la ciudad, regulando medios de transporte y asignando zonas específicas para cada tipo de vehículo, siendo estos temas que en nuestra realidad no han pasado de una idea debido a la falta de un modelo real que sirva como punto de partida para la creación de planes de ciudad sostenible.

El resultado de este tipo de modelado son datos de comportamiento vehicular para ser usados en análisis y experimentación sobre diferentes tipos de criterio

con información real. El modelado de tráfico utiliza una estructura específica alimentada con datos reales para discutir detalladamente sobre un tema de relevancia social asociado al tráfico de una ciudad o área específica. (Hoogendoorn & Bovy, 2001).

La ciudad de Quito se ha convertido en una gran centro metropolitano que se encuentra a la altura de las demás ciudades del mundo en cuanto a una comparación a escala de la cantidad de medios de transporte que se mueven a través de esta diariamente, sin embargo, esto atrae una serie de problemas que no han sido tomados en cuenta ni solucionados de manera eficiente por los dirigentes de la ciudad, quienes están siempre en búsqueda de nuevas soluciones, pero sin una base que indique los mejores lugares para una adecuada implementación de estas.

Desde hace varios años atrás, el principal problema que sufren las grandes ciudades, en especial en sus zonas urbanas, es el aglomeramiento de vehículos en distintas horas del día en ciertas zonas específicas, produciendo varios tipos de contaminación perjudiciales para el medio ambiente y la salud. (Tapia, 1998).

En la constante búsqueda de la solución de los problemas relacionados al tráfico, se creó la teoría de tráfico vehicular, la cual busca, a través de métodos analíticos, conocer la base conceptual de este fenómeno, para determinar contramedidas, capaces de corregir el caso estudiado. (Fernández, 2011).

Al hablar de modelado y análisis de datos, la técnica de agrupación por clústeres es una herramienta que permite el análisis por agrupación de sectores en base a una similitud existente según un parámetro de entrada, sin embargo, su principal ventaja es su capacidad de normalización y visualización de datos,

representándolos de una forma amigable y entendible al usuario. (Sturn et al, 2002).

Finalmente, el propósito de este trabajo es emplear conocimientos sobre modelado y simulación, orientándolos a resolver problemas de tráfico en la ciudad de Quito, mediante la aplicación de la teoría de tráfico vehicular para determinar la mejor forma de modelar una ciudad con una topología tan compleja, esto debido a la forma alargada de la ciudad y a la estructura de sus calles, y que a la vez proporcione un resultado entendible y fácil de visualizar, el cual sirva de base para la implementación de políticas que permitan mejorar la circulación vehicular, entre otros planes relacionados a ciudades sostenibles y sus medios de transporte urbano.

1.2. Alcance

Brindar información y proponer un modelo, sustentado en datos reales de la ciudad de Quito, sobre el comportamiento del tráfico vehicular, para determinar la ubicación geográfica de puntos de interés que pueden ser de gran aporte en la implementación de cualquier plan de ciudad sostenible dentro del área metropolitana de Quito.

La creación de la red de tráfico será realizada en Python, lenguaje de programación el cual actualmente es considerado uno de los mejores para el análisis de datos e información. De igual manera, para alimentar la red de tráfico se usarán dos APIs de Google, las cuales serán *Directions API* y *Distance Matrix API*, esta última proveerá la información en tiempo real del tráfico. También se usará la librería de Python *Open Street Maps NetworkX* (OSMNX), la cual brinda información de los nodos de la ciudad y sus conexiones.

Esta información será tratada mediante la aplicación de distintos tipos de algoritmos que permitan el podado de la red al tratarla como una estructura de datos compleja, la cual permita obtener un modelo base que sirva de guía para la realización de un plan sobre el área metropolitana de Quito. Incluyendo entre estos algoritmos modificaciones adecuadas a algoritmos establecidos para estructuras de datos, obteniendo así variaciones propias que permitan alcanzar los objetivos del proyecto y determinar de manera autónoma los sectores de mayor importancia metropolitana.

Finalmente, como resultado entregable, se tendrán los notebooks de *Jupyter* con comentarios que detallen los procedimientos que se siguieron para la extracción de datos, la optimización de los nodos y sus conexiones, y la creación de la nueva red de tráfico detallada utilizando las APIs de Google con información en tiempo real. Cabe recalcar, que como parte del proyecto no se pretende realizar la implementación de un software, ni algún plan de ciudad sostenible, solo se establecen las bases necesarias para su futuro desarrollo, mediante la entrega de un modelo correctamente estructurado y fundamentado.

1.3. Justificación

Muchos de los gases producidos por los vehículos con motor de combustión interna son dañinos para los seres vivos y para el medio ambiente. Entre los gases más peligrosos que producen estos vehículos tenemos dióxido de carbono, el cual es uno de los principales causantes del cambio climático en el planeta. Monóxido de carbono, el cual es venenoso para el cuerpo humano y reduce la capacidad de la sangre para absorber oxígeno. Hidrocarburos, los cuales son considerados cancerígenos. Óxidos de nitrógeno, estos gases pueden causar daño en la membrana pituitaria y son responsables también de la contaminación atmosférica. (Ortmann, 2005).

Por otro lado, la contaminación auditiva también afecta de gran manera a la sociedad actual y a la salud de las personas, provocando malestares tanto físicos como psicológicos, por ejemplo, incremento de la presión arterial, cambios hormonales, alteraciones en el sistema circulatorio, nerviosismo, ansiedad, etc. (Alfie y Salinas, 2017).

Es por esto por lo que es importante tener en cuenta el tráfico vehicular en las grandes ciudades, ya que los vehículos son los principales agentes relacionados a los problemas de contaminación mencionados. En la ciudad de Quito, el 52% de emisiones de CO₂ es producida por los vehículos. (El Telégrafo, 2017). De igual manera, los vehículos automotores son causantes del 80% del ruido total en las ciudades. (Alfie y Salinas, 2017).

El presente proyecto pretende brindar información real del tráfico vehicular de la ciudad de Quito, la cual sirva como base para futuras investigaciones, que permitan solucionar las problemáticas mencionadas anteriormente y mejorar la calidad de vida en la ciudad, mediante la implementación de políticas de ciudad sostenible.

1.4. Objetivo General

Crear un modelo de flujo de tráfico de Quito usando algoritmos de optimización en redes complejas, para la construcción de una red de puntos de interés geográfico, agrupados mediante clústeres.

1.5. Objetivos Específicos

- Realizar un adecuado proceso de extracción y simplificación de la red de Quito.

- Crear algoritmos de redes complejas capaces de determinar los puntos de interés relevantes, en conjunto con su respectiva información de tráfico vehicular.
- Obtener un modelo de flujo de tráfico, mediante algoritmos de agrupación de nodos en base a su ubicación.

1.6. Metodología

En el presente proyecto se hará uso de dos métodos de investigación, los cuales se acoplan al modelado y simulación para fines relacionados a *Data Science*. El método inductivo basado en el modelado permitirá la realización de un diseño de red que abarque de manera general los temas tratados en la investigación y permita obtener resultados aproximados de la red de tráfico en Quito. Por otra parte, el método analítico basado en simulación brindará la posibilidad de partir de estos datos generales a enfoques más específicos para el estudio a detalle de los diferentes factores, los cuales intervendrán en cuanto a la necesidad de encontrar las ubicaciones óptimas, generando así, teorías capaces de comprender el comportamiento previsto en el tráfico de la ciudad como efecto de la implementación de planes de sostenibilidad.

2. MARCO TEÓRICO

Durante el desarrollo de este trabajo se utilizó una serie de herramientas y conceptos necesarios para la adecuada implementación de un modelo de flujo de tráfico basado en el área metropolitana de Quito. Los puntos detallados a continuación son la base estructural del presente trabajo y son el punto de partida para la implementación de la solución.

2.1. Conceptos Generales

Considerando la gran cantidad de conceptos que abarca el presente proyecto, se decidió agruparlos según su uso a lo largo del desarrollo de este, sin embargo, los siguientes temas presentados en esta sección, hacen referencia a conceptos generales e introductorios para el entendimiento del proceso descrito a lo largo del documento.

2.1.1. Data Science

Data Science es un término que ha tenido gran acogida en los últimos años, y es el mismo que se utiliza para referirse al área de conocimiento que está dirigido este trabajo. De acuerdo con Van Der Aalst “*Data Science* es el estudio de la extracción de conocimiento generalizable de los datos” (Van Der Aalst, 2016). Actualmente, datos referentes a distintos aspectos de la vida están siendo constantemente recopilados, sin embargo, lo que se hace con estos es muy poco, sobre todo en los países del tercer mundo. Por lo cual la inclusión de proyectos que se basen en análisis de datos según lo propone la rama de la ciencia conocida como *Data Science*, es de vital importancia para el beneficio de la sociedad, demostrando que la automatización de procesos de estudio permite tomar medidas correctivas rápidamente.

2.1.2. Teoría de Grafos

La librería utilizada en este trabajo, OSMnx, permite la consulta de información geográfica devolviendo una cantidad determinada de datos en la estructura de datos conocida como grafo. Como lo menciona Barrero, “Los grafos son una estructura de datos no lineal, la cual se puede usar para modelar distintas aplicaciones” (Barrero et al. 2010). Siendo la principal fuente de información sobre la cual se trabaja para obtener los datos de los diferentes nodos de la red

de la ciudad, representando estas las ubicaciones geográficas y otorgando, además de coordenadas, información relevante como distancia entre conexiones y tipo de calle.

2.1.3. Multigrafos

La información que se trata en este trabajo no es un grafo simple, por el contrario, se manejan tipos de datos complejos con una estructura más compleja que permita abarcar el comportamiento y topología de una ciudad, en este caso, multigrafos. Según Pakhrou, “Un multigrafo es un grafo $G = (V, A)$, donde V se refiere a un conjunto de vértices o nodos, mientras que A es un conjunto de pares de nodos ordenados, también llamados aristas, con la peculiaridad de que A contiene dos o más aristas con el mismo par de extremos. Las aristas distintas que unen el mismo par de vértices se llaman aristas múltiples” (Pakhrou, 2013). Esto se refiere a que, en la red obtenida, más de una arista puede conectar dos nodos, e incluso pueden existir conexiones consigo mismo.

2.1.4. Nodos

El principal componente del trabajo realizado son los nodos, a través de estos y sus conexiones se encuentra toda la información necesaria y se realiza los pasos que permiten obtener el modelo final. De acuerdo con Bianchini “Los nodos funcionan como colecciones de datos no estructurados los cuales son agrupados de forma tal de crear una entidad lógica” (Bianchini, 1999).

Dicha entidad lógica, en este caso se representa mediante un punto geográfico en el área metropolitana de Quito, no necesariamente un punto importante, tan solo un fragmento de una calle, sin embargo, su aporte para el estudio realizado

es de vital importancia, ya que se obtuvo la información relevante mediante la agrupación de estos.

2.1.5. Análisis mediante clústeres

Para la etapa final del modelo realizado se decidió expresar la información mediante una agrupación geográfica a manera de clústeres, es decir, según su ubicación en el mapa se determinó la cantidad necesaria de agrupaciones y se agrupó cada una en su respectiva zona de interés.

El análisis de conglomerados comprende una variedad de métodos para clasificar datos multivariados en subgrupos. Al organizar datos multivariados en dichos subgrupos, la agrupación puede ayudar a revelar las características de cualquier estructura o patrón presente. Estas técnicas han demostrado ser útiles en distintos campos de la ciencia. (Everitt et al. 2011).

2.1.6. Sistema Geodésico Mundial de Coordenadas.

Los valores de las coordenadas obtenidas en este trabajo se transformaron y mantuvieron durante todo el proceso en base al Sistema Geodésico Mundial de Coordenadas (WGS84), el cual es un estándar en la actualidad para todos los documentos científicos que tienen relación con información geográfica.

El WGS84 y el Datum norteamericano de 1983 (NAD83) son actualmente los sistemas de referencia espacial más ampliamente utilizados en América del Norte. El WGS84 es el sistema "nativo" predeterminado utilizado por el Sistema de Posicionamiento Global (GPS) y receptores GPS comerciales. La realización física de estos sistemas de referencia ha sufrido varias actualizaciones desde

que se introdujeron por primera vez hace más de dos décadas. WGS84, no tiene una red terrestre de acceso público. Solo es accesible a través de órbitas de transmisión que proporcionan posiciones con una precisión de aproximadamente un metro en el mejor de los casos (con correcciones aumentadas). (Craymer, 2006).

2.1.7. Distancia euclidiana

Durante el análisis realizado, se decidió utilizar la distancia euclidiana como medio para saber la distancia resultante entre los puntos del mapa, independientemente de las restricciones de movilidad existentes por la topología de la ciudad.

Lo más importante que se requiere saber en este punto del análisis es la separación entre los clústeres, de modo que, como lo menciona Pérez, “La distancia euclidiana es un número positivo que indica la separación que tienen dos puntos en un espacio donde se cumplen los axiomas y teoremas de la geometría de Euclides”. (Pérez, 2020).

2.1.8. Agrupamiento jerárquico

Existen diversos tipos de agrupamientos, sin embargo, durante la realización de este trabajo se determinó que el agrupamiento mediante jerarquía, o también conocido como jerárquico, es aquel que sirve para satisfacer las necesidades que se plantearon durante la investigación.

Los algoritmos jerárquicos son aquellos en los que se va particionando el conjunto de datos por niveles, de modo tal que en cada nivel

generalmente, se unen o se dividen dos grupos del nivel anterior, según si es un algoritmo aglomerativo o divisivo. (Pascual et al. 2007).

2.1.9. Método de Ward

Para completar el análisis de clúster mediante una agrupación jerárquica, se decidió utilizar el método de Ward, el cual menciona Everitt, “Se trata de un procedimiento general donde el criterio para la elección del par de clústeres a mezclar en cada paso se basa en el valor óptimo de una función objetivo”. (Everitt et al. 2001). Donde en este caso la función objetivo se comprende como las necesidades de la investigación presentada.

2.2. Mecanismos de simplificación

Una parte fundamental de este trabajo es la utilización de diferentes tipos de simplificaciones, las cuales permiten obtener la red deseada. Para esto se utiliza conceptos aritméticos y estadísticos como son el uso de media y el percentil, e incluso conceptos de algoritmos de redes complejas como es el caso del algoritmo de Dijkstra.

2.2.1. Media Aritmética

En el presente trabajo se realizó varios procesos de simplificación, entre los cuales uno se basó en la media, la cual Paz define como “Un promedio simple que muestra el valor central de los datos constituyendo ser la medida de ubicación que más se utiliza. En general, es calculada sumando los valores de interés y dividiendo entre el número de valores sumados”. (Paz, 2007).

2.2.2. Percentil

Una de las medidas estadísticas que se utilizaron como discriminante al momento de realizar el proceso de simplificación es el percentil. Definido por Galton como “Una medida de posición usada en estadística que indica, una vez ordenados los datos de menor a mayor, el valor de la variable por debajo del cual se encuentra un porcentaje dado de observaciones en un grupo” (Galton, 1885).

2.2.3. Algoritmo de Dijkstra

El algoritmo de Dijkstra es un algoritmo eficiente (de complejidad $O(n^2)$ donde n es el número de vértices) que sirve para encontrar el camino de coste mínimo desde un nodo origen a todos los demás nodos del grafo. Fue diseñado por el holandés Edsger Wybe Dijkstra en 1959. El fundamento sobre el que se asienta este algoritmo es el principio de optimalidad: si el camino más corto entre los vértices u y v pasa por el vértice w , entonces la parte del camino que va de w a v debe ser el camino más corto entre todos los caminos que van de w a v . De esta manera, se van construyendo sucesivamente los caminos de coste mínimo desde un vértice inicial hasta cada uno de los vértices del grafo, y se utilizan los caminos conseguidos como parte de los nuevos caminos. (Torrubia & Terrazas, 2001).

Este algoritmo se toma como base para diferentes tipos de estudios, incluyendo el trabajo realizado, sin embargo, este es una base, mas no se realizó su implementación directa ya que se realizó un algoritmo propio que fuera capaz de resolver la problemática de la simplificación dentro de redes complejas, como lo son los multigrafos utilizados.

2.3. Herramientas utilizadas

Este trabajo fue realizado mediante el uso de varios tipos de herramientas y librerías que permitieron recolectar, modificar y presentar los datos obtenidos de manera que generen información valiosa para el proyecto de investigación. Estas herramientas se detallan a continuación.

2.3.1. OSMnx

Los académicos urbanos han estudiado las redes de calles de varias maneras, sin embargo, en muchas ocasiones no existe una alta disponibilidad de datos, lo cual genera limitaciones de consistencia en cuanto a planificación urbana y análisis de redes de calles. Para abordar estos desafíos, se presenta OSMnx, una nueva herramienta para hacer que la recopilación de datos, la creación y análisis de redes de calles sean simples, consistentes, automatizables y sólidas desde las perspectivas de la teoría de grafos, el transporte y el diseño urbano. (Boeing, 2017).

Debido al gran uso que tiene esta herramienta dentro del análisis de redes complejas basadas en la topología de las ciudades, es el medio principal para la implementación de cualquier tipo de problemática relacionada al análisis computacional del tráfico. Como se mencionó anteriormente, esta herramienta permite trabajar con una de las estructuras de datos más utilizadas para la resolución de problemas mediante *Data Science*. El uso de grafos permite relacionar información de todo tipo, de tal manera que siempre existan conexiones entre estas que permitan observar su relación y evaluarla en base a las necesidades, determinando así los nodos con información relevante para cada tipo de análisis.

2.3.2. Open Street Map

Open Street Map es un servicio de datos espaciales, proporcionados de manera gratuita por lo cual actualmente son altamente consumidos por desarrolladores de todo el mundo para su implementación en distintos proyectos.

La cantidad de datos espaciales voluntarios del proyecto OpenStreetMap ha aumentado rápidamente en los últimos cinco años. En muchas áreas, la calidad de los datos ya coincide con la de los datos de mapas comerciales, si no es que la supera por completo. (Luxen & Vetter, 2011).

Los usuarios han recopilado y cargado los datos de OSM como ya se mencionó anteriormente. Cualquier persona que se haya registrado en el sitio web de OSM puede subir datos y contribuir al proyecto OSM. Las capas (tipos de información) disponibles en el sitio web de OSM son las siguientes: autopista, barrera, ciclovía, tipo de vía, vía fluvial, ferrocarril, vía aérea, poder, hecho por el hombre, ocio, servicios, tienda, turismo, histórico, *landuse*, militar, natural, ruta, límite, deporte, colindantes, accesorios, propiedades, restricciones, nombre, referencias, lugares, direcciones y anotaciones. El dato utilizado en OSM OpenStreetMap es el dato WGS-84 lat / long, todas las pistas y las ediciones deben estar en la misma proyección, ya que este es el dato predeterminado para GPS receptores. (Kounadi, 2009).

2.3.3. Google Maps

El principal medio de obtención de datos en tiempo real es *Google Maps*, también conocido como *Maps*, servicio que provee una cantidad amplia de APIs que los desarrolladores pueden consultar ateniéndose a un costo establecido

dependiendo del servicio prestado. Siendo utilizadas en este proyecto *Distance Matrix API* y *Directions API* para la obtención de información de calidad.

Ofrece imágenes de satélite, mapa de calles, vista panorámica (Street View), estado del tráfico en tiempo real y planificación de rutas para viajar a pie, en coche, en bicicleta o en transporte público. A esto se suman millones de referencias geográficas y comerciales que ofrecen imágenes, datos de contacto y llamadas a la acción; llamar por teléfono, acceder a la Web o generar rutas para llegar. (Maps, 2015).

2.3.4. Directions API

Directions API es un servicio que calcula indicaciones entre ubicaciones. Puede buscar direcciones para varios modos de transporte, incluidos el tránsito, la conducción, el caminar o el ciclismo. (Google, 2020).

2.3.5. Distance Matrix API

Es un servicio que proporciona distancia y tiempo de viaje para una matriz de orígenes y destinos, según la ruta recomendada entre los puntos de inicio y finalización. (Google, 2020).

2.3.6. Leaflet

Leaflet es la biblioteca JavaScript de código abierto líder en mapas interactivos para dispositivos móviles. Tiene todas las características de mapeo que la mayoría de los desarrolladores necesitan. Leaflet está diseñado teniendo siempre en cuenta la simplicidad y rendimiento como su objetivo principal.

Funciona de manera eficiente con todas las principales computadoras de escritorio y plataformas móviles. Tiene una API bien documentada y un código fuente simple. (Leaflet, 2019).

Sobre esta herramienta se desarrolló la mayoría de las acciones referentes a la visualización del mapa, añadiendo elementos gráficos y distintivos con información relevante sobre cada uno de los puntos geográficos obtenidos.

2.3.7. Folium

La implementación de la solución se realizó en su mayoría en el lenguaje de programación Python, lo cual obligó a buscar alternativas que permitan utilizar librerías de JavaScript dentro de Python, obteniendo como forma de visualización de datos la librería Folium, a través de la cual se realizó los gráficos respectivos para el entendimiento de la información obtenida.

Folium facilita la visualización de datos que han sido manipulados en Python en un mapa de Leaflet interactivo. Permite tanto la vinculación de datos a un mapa para visualizaciones de mapas coropléticos, como el paso de visualizaciones de vectores / ráster / HTML como marcadores en el mapa. La biblioteca tiene varios mosaicos integrados de OpenStreetMap, Mapbox y Stamen, y admite mosaicos personalizados con claves API Mapbox o Cloudmade. Folium admite superposiciones de imagen, video, GeoJSON y TopoJSON. (Folium, 2013).

2.3.8. Anaconda Navigator

Para la obtención del entorno de desarrollo ideal, se decidió utilizar *Anaconda Navigator*, debido a que durante su proceso de instalación esta herramienta

provee la instalación de distintas versiones de Python, además de una serie de programas indispensables para codificar en el mismo, siendo *Jupyter Notebook* el más utilizado en este trabajo.

Anaconda Navigator es una interfaz gráfica de usuario (GUI) de escritorio incluida en la distribución de Anaconda® que le permite iniciar aplicaciones y administrar fácilmente paquetes, entornos y canales de Conda sin usar comandos de línea de comandos. *Navigator* puede buscar paquetes en Anaconda Cloud o en un repositorio local de Anaconda. Está disponible para Windows, macOS y Linux. (Anaconda, 2020).

2.3.9. Jupyter Notebook

Jupyter Notebook es una aplicación web de código abierto que le permite crear y compartir documentos que contienen código, ecuaciones, visualizaciones y texto narrativo. Los usos incluyen: limpieza y transformación de datos, simulación numérica, modelado estadístico, visualización de datos, aprendizaje automático y mucho más. (Jupyter, 2020).

Esta herramienta se utiliza generalmente para investigaciones científicas que tienen que ver con análisis de datos, esto se debe a que su interfaz gráfica proporciona una visualización previa de lo que se está realizando, permitiendo así no solo compartir el código, también los resultados obtenidos, asegurando transparencia en la validación de procesos analíticos exitosos y permitiendo que los resultados sean visibles para otras partes interesadas sin la necesidad de tener el ambiente de desarrollo en su computador, apoyándose en su capacidad de exportación a distintos formatos de archivo.

Jupyter es una herramienta web interactiva gratuita de código abierto conocida como un cuaderno computacional, que los investigadores pueden usar para combinar código de software, salida computacional, texto explicativo y recursos multimedia en un solo documento. Los cuadernos computacionales han existido durante décadas, pero *Jupyter* en particular ha crecido en popularidad en los últimos años. Esta rápida incorporación ha sido ayudada por una comunidad entusiasta de desarrolladores, usuarios y una arquitectura rediseñada que permite que el cuaderno hable docenas de lenguajes de programación, un hecho reflejado en su nombre, que fue inspirado, según el cofundador Fernando Pérez, por los lenguajes de programación Julia (Ju), Python (Py) y R. (Perkel, 2018).

2.3.10. Inkscape

Debido a la naturaleza propia de este trabajo al ser utilizado como documento científico, se decidió mejorar los gráficos exportados con otras herramientas para procurar que la calidad de las imágenes obtenidas sea la adecuada para una correcta visualización de la información. Esto se logró mediante el uso de la herramienta Inkscape y su propia línea de comandos.

Inkscape es un editor de gráficos vectoriales de código abierto, similar a programas como Adobe Illustrator, Corel Draw, Freehand o Xara X. Lo que lo hace único es que usa como formato nativo el *Scalable Vector Graphics* (SVG), un estándar abierto de W3C basado en XML. Tiene herramientas de dibujo vectorial sofisticadas con capacidades comparables con los estándares de la industria. Puede exportar e importar varios formatos de archivo, incluyendo SVG, AI, EPS, PDF, PS y PNG. Tiene funcionalidades muy fáciles de comprender, una interfaz sencilla, soporte multi-idiomias y está diseñado para ser extensible, los usuarios pueden personalizar las funcionalidades de Inkscape con sus múltiples extensiones. (Inkscape, 2020).

2.4. Lenguajes de programación y formatos

La gran mayoría de scripts realizados en el proyecto se desarrollaron con el lenguaje de programación Python, sin embargo, también se utilizó JavaScript para la visualización en mapas. Además, se utilizó la Notación de Objetos de JavaScript (JSON) para el manejo de datos geográficos a través de las diferentes librerías, en conjunto con Gráficos Vectoriales Escalables (SVG) para la visualización gráfica de los resultados.

2.4.1. Lenguaje Python

Python es un lenguaje de programación interpretado, de alto nivel y de propósito general. Creado por Guido van Rossum y lanzado por primera vez en 1991, la filosofía de diseño de Python enfatiza la legibilidad del código con su uso notable de espacios en blanco significativos. Sus construcciones de lenguaje y su enfoque orientado a objetos tienen como objetivo ayudar a los programadores a escribir código claro y lógico para proyectos de pequeña y gran escala. (Van Rossum, 2007).

Python es un excelente lenguaje de "dirección" para códigos científicos escritos en otros idiomas. Sin embargo, con herramientas básicas adicionales, Python se transforma en un lenguaje de alto nivel adecuado para el código científico y de ingeniería que a menudo es lo suficientemente rápido como para ser inmediatamente útil, pero también lo suficientemente flexible como para ser acelerado con extensiones adicionales. (Oliphant, 2007).

Al ser el lenguaje de programación por excelencia para tareas relacionadas con *Data Science*, se escogió Python para la completa realización del modelo a ser implementado, sin embargo, debido a ciertas limitaciones de las librerías se tuvo

que hacer uso de componentes externos para obtener una correcta visualización de los datos.

2.4.2. JavaScript

JavaScript (JS) es un lenguaje ligero e interpretado, orientado a objetos con funciones de primera clase, más conocido como el lenguaje de script para páginas web, pero también usado en muchos entornos sin navegador, tales como node.js, Apache CouchDB y Adobe Acrobat. Es un lenguaje script multiparadigma, basado en prototipos, dinámico, soporta estilos de programación funcional, orientada a objetos e imperativa. (Mozilla, 2020).

Este es el lenguaje en el que se desarrolló Leaflet, sin embargo, a pesar de utilizar las versiones relacionadas con Python, estas cuentan con ciertas limitaciones. Es por esto por lo que se decidió utilizar JavaScript para crear varios mapas durante la fase de visualización de resultados, los cuales gracias a este lenguaje se exportaron en formatos óptimos para su correcta interpretación gráfica.

2.4.3. JSON Geográfico

La información manejada en este proyecto se recibe en un formato JSON específico, el cual está creado para contener datos geográficos que pueden ser consumibles por los diferentes programas o librerías que utilizan como entrada este tipo de datos, facilitando la manipulación de estos mediante un estándar claramente definido.

GeoJSON es un formato de intercambio de datos geoespaciales basado en JSON. Define varios tipos de objetos JSON y la manera en que se combinan para representar datos sobre características geográficas, sus propiedades y sus extensiones espaciales. GeoJSON utiliza un sistema de referencia de coordenadas geográficas, WGS84, y unidades de grados decimales. (Butler et al. 2016).

2.4.4. Formato SVG

SVG es un lenguaje para describir gráficos bidimensionales en XML [XML10]. Permite tres tipos de objetos gráficos: formas gráficas vectoriales (por ejemplo, trazados que consisten en líneas rectas y curvas), imágenes y texto. Los objetos gráficos se pueden agrupar, diseñar, transformar y componer en objetos renderizados. El conjunto de características incluye transformaciones anidadas, trazados de recorte, máscaras alfa, efectos de filtro y objetos de plantilla. Los dibujos SVG pueden ser interactivos y dinámicos. Las animaciones se pueden definir y activar declarativamente (es decir, mediante la incorporación de elementos de animación SVG en el contenido SVG) o mediante secuencias de comandos. Las aplicaciones sofisticadas de SVG son posibles mediante el uso de un lenguaje de script complementario que accede a *SVG Document Object Model* (DOM), que proporciona acceso completo a elementos, atributos y propiedades. (Ferraiolo et al. 2000).

Debido a la necesidad de exportar los gráficos resultantes del modelo en imágenes de alta calidad que le otorguen información al usuario, se decidió hacer uso del formato SVG como estándar para la presentación de todos los datos, por lo cual en el código realizado siempre se procuró la exportación a SVG para la manipulación de estas imágenes en otras herramientas y utilización de las mismas en documentos científicos.

3. DESARROLLO DE LA SOLUCIÓN

Esta sección describe el proceso utilizado para mapear el flujo de tráfico entre puntos de interés dentro del área metropolitana de la ciudad de Quito. Todos los procesos, desde extracción de la red de la ciudad hasta la selección de los nodos, se automatizaron mediante el uso de distintos tipos de estructuras de datos y sus respectivos algoritmos, además de la creación de algoritmos propios los cuales se desarrollaron con la finalidad de tener un código completamente funcional e independiente de datos fijos, si la ciudad sufre algún cambio, al correr el código arroja nuevos resultados en base a la realidad actual, por lo cual se obtuvo un modelo listo para ser partícipe de la implementación de planes de ciudad sostenible, gracias a su capacidad de adaptarse a las restricciones geográficas.

3.1. Flujo de procesos

En primer lugar, se extrajo la red (grafo) de la ciudad. Luego se realizó un proceso de simplificación para reducir el número de nodos en el grafo de la ciudad. La red también se recortó en base a la distancia de las calles (conexiones) conectando entre sí los nodos en el grafo de la ciudad. Una vez se tienen los nodos finales, se realiza una red totalmente distinta, la cual se vuelve a recortar de acuerdo con la distancia máxima entre puntos, obteniendo un mapa transversal de puntos en la ciudad. Un análisis de agrupamiento jerárquico basado en distancia euclidiana se realiza inmediatamente para identificar las agrupaciones de los puntos de acuerdo con su localización geográfica. Por último, el flujo de tráfico entre clústeres se calcula para ilustrar los tiempos que toman atravesar la ciudad. Los procesos realizados se ilustran en la Figura 1.

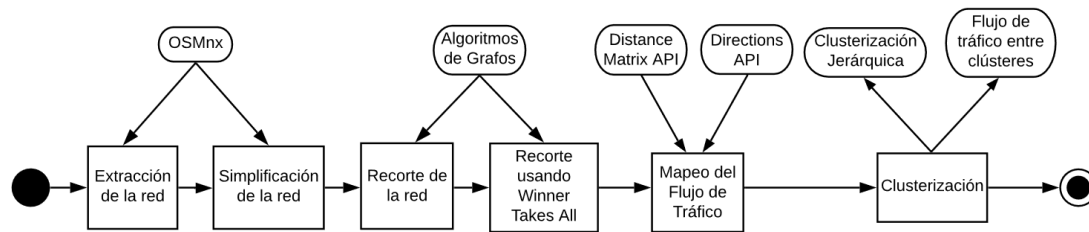


Figura 1. Representación esquemática de los procesos.

3.2. Extracción de la red

En el análisis de mapas para diferentes fines, se tiene varios métodos para obtener la información, sin embargo, muchos mapas no cuentan con la capacidad de brindar una red de nodos, pero como menciona Boeing, “La extracción de la red de la ciudad a manera de estructura de datos puede realizarse utilizando OSMnx”. (Boeing, 2017). Foulds describe OSMnx como “Una herramienta para la creación y recolección de datos de calles, que permite analizarlas desde la perspectiva de la teoría de grafos”. (Foulds, 2012). Siendo esto lo que se espera obtener para comenzar con el análisis de la información geográfica.

Estas redes son útiles para estudios de transporte y diseños urbanos. La red puede ser definida como un grafo el cual es una tupla $G = (V, E)$, donde $v_i \in V; i = 1, \dots, n$, son el conjunto de nodos pertenecientes a la red, y $E \subseteq \{\{v_i, v_j\} | (v_i, v_j) \in V^2 \wedge i \neq j\}$, son las conexiones entre los nodos i y j . Una conexión es una tupla (v_i, v_j) , la dirección es implícita en esta tupla, donde la conexión va de v_i a v_j . Los nodos en la red son puntos de interés (en su mayoría esquinas) de la ciudad y las conexiones son las calles. La red es un grafo dirigido y con peso, donde los pesos de las conexiones son la distancia entre los nodos. OSMnx devuelve información de grandes avenidas, carreteras, las cuales están

conformadas por una serie de nodos internos con datos relevantes sobre la topografía de la zona requerida.

OSMnx, al ser una librería de *Open Streets Maps*, permite la extracción mediante varios métodos, estos pueden ser según un punto determinado consultando coordenadas geográficas, un lugar específico, dirección, etc. Para propósitos de este proyecto se utilizó el método de coordenadas geográficas, debido a que en la búsqueda por lugar no se encuentra actualmente el nombre de la ciudad. Sin embargo, se asignó las coordenadas (-0.181100, -78.478611), las cuales corresponden a la ciudad de Quito. Partiendo de este punto se aplicó ciertas características o parámetros a la red general, los cuales permitieron extraer la red deseada. Los parámetros que se aplicaron son el tipo de red, solicitando una red vehicular, y una distancia hacia cada dirección de 6 Kilómetros. Gracias a esto se logró extraer una red completa del Distrito Metropolitano de Quito, abarcando por completo sus zonas de interés y sus salidas por los extremos.

El resultado obtenido de la extracción de la red es un multigrafo compuesto por nodos que representan partes de la ciudad distribuidos según su ubicación geográfica. Este multigrafo es un tipo de grafo con la peculiaridad que en su composición múltiples conexiones son permitidas o incluso necesarias entre nodos, permitiendo también conexiones con el mismo nodo. OSMnx además brinda información acerca de la red urbana extraída, entre la cual destacan datos como total de distancia de cada calle, el promedio de las distancias de todas las calles, número de intersecciones, tipo de calle, entre otros datos relevantes para cualquier tipo de investigación basada en datos geográficos.

3.3. Simplificación de la red

Con los resultados obtenidos de la extracción de la red, se procedió con el proceso de simplificación, el cual inicialmente se logró mediante el mismo

OSMnx. Esta librería permite realizar una simplificación de la red obtenida, esto dado a que la red original puede ser muy grande, lo cual puede hacerla computacionalmente compleja. Es por esto por lo que se requiere realizar un proceso de simplificación luego de extraer los datos. OSMnx permite juntar todos los nodos que son parte de un camino de intersecciones, removiéndolos de la red, pero añadiendo el peso a una sola conexión la cual es usada como punto final para mantener las distancias correctas entre los nodos y acumular su información.

Esto se puede realizar al establecer en verdadero el atributo *simplify* durante la extracción de la red, sin embargo, este proceso de simplificación temprana genera gran pérdida de información. Debido a la necesidad de tener la información completa, se utilizó una manera distinta al copiar el resultado en una nueva variable y llamar al método correspondiente que proporciona OSMnx. El método *simplify* al trabajar sobre la red completa permite minimizar la pérdida de información relevante. Para poder comparar los resultados obtenidos y su variación se realizó un cambio de color dependiendo de si son nodos por eliminarse o nodos a mantenerse. La Figura 2 marca los nodos de la red, donde los nodos de color rojo se removieron y los nodos de color azul se mantuvieron con los datos acumulados de los nodos removidos que se encontraban entre estos.

Finalmente, gracias a este proceso de comparación se pudo determinar que los nodos resultantes eran los deseados para la continuidad del trabajo, no solo por su ubicación geográfica, también porque se minimizó drásticamente la cantidad de datos obtenidos al prescindir de miles de nodos que solo aportaban información de distancia que ya se encuentra acumulada en las nuevas conexiones.

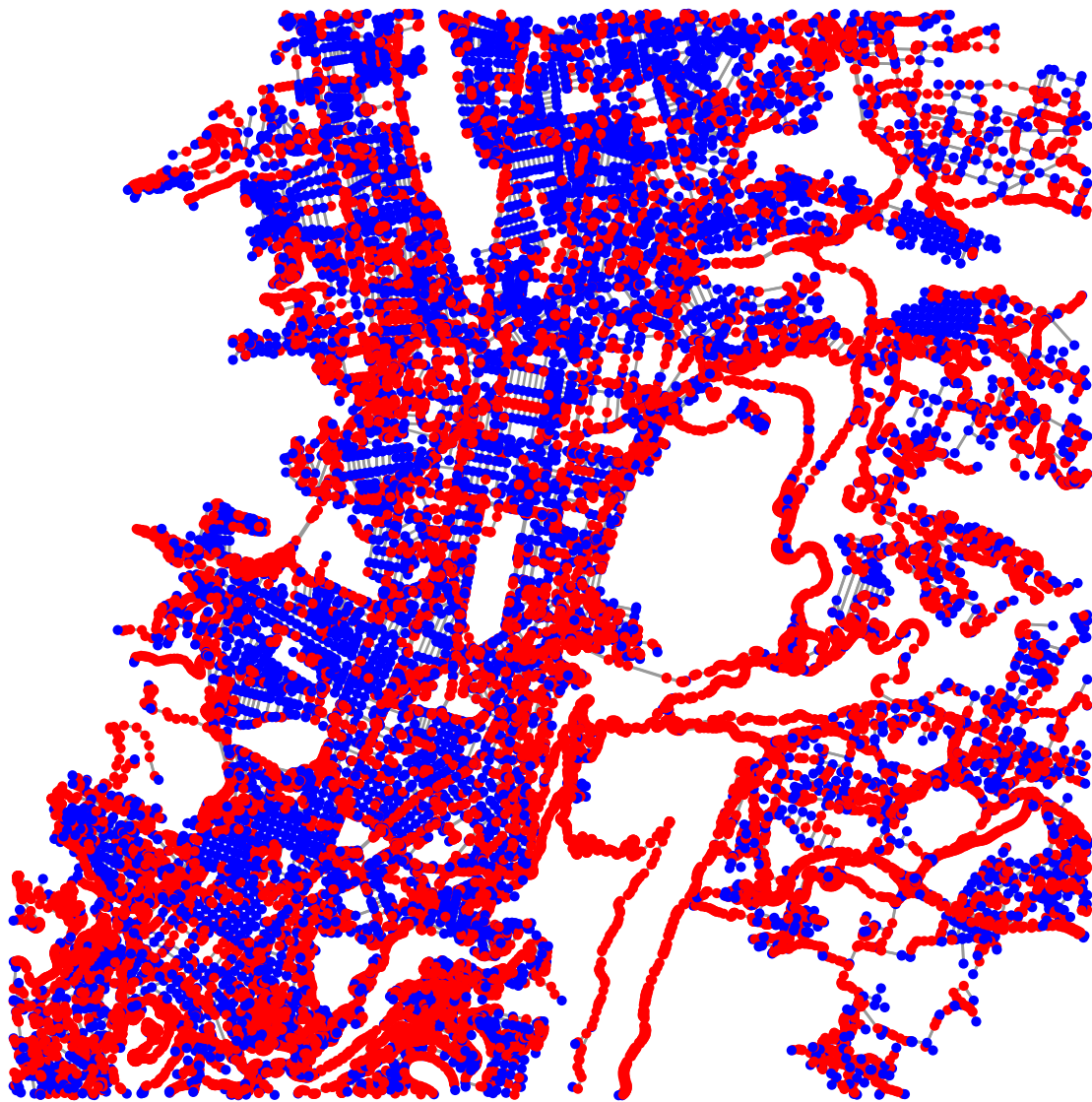


Figura 2. Nodos resultantes y eliminados del proceso de simplificación.

Como se puede observar en la figura previa, al realizar este proceso se obtiene una red con conexiones únicas, las cuales son creadas con las distancias acumuladas de todos los nodos correspondientes al camino simplificado. Estas conexiones finales están compuestas de un OSM ID, la información de los nodos “origen” y “destino” y la distancia entre ellos. La red simplificada ($G = (V, E)$), puede ser representada como la matriz de adyacencia J y puede ser expresada como el elemento del producto $J = C \circ D = J_{ij} = C_{ij} \cdot D_{ij}$ con la matriz de conectividad C donde, existe una conexión entre el nodo i y el nodo j , y $C_{ij} = 0$

en el caso contrario; Y la matriz de distancia de conducción D , con $D_{ij} = d_{ij}$ donde d_{ij} es la distancia del nodo i hacia el nodo j . Cabe recalcar que la matriz de distancia no es simétrica, es decir $D_{ij} \neq D_{ji}$, esto se debe a que la ruta de conducción del nodo i hacia el nodo j , no siempre es la misma que la ruta de conducción del nodo j hacia el nodo i . De igual manera las conexiones sobre un mismo nodo no están permitidas, $C_{ij} = 0$ para $i = j$. Un ejemplo más comprensible sobre lo que realiza el proceso se puede observar en la Figura 3 y la Figura 4, mostradas a continuación. Aquí se visualiza un acercamiento a la zona del parque La Carolina que permite observar de mejor manera lo que se realizó.

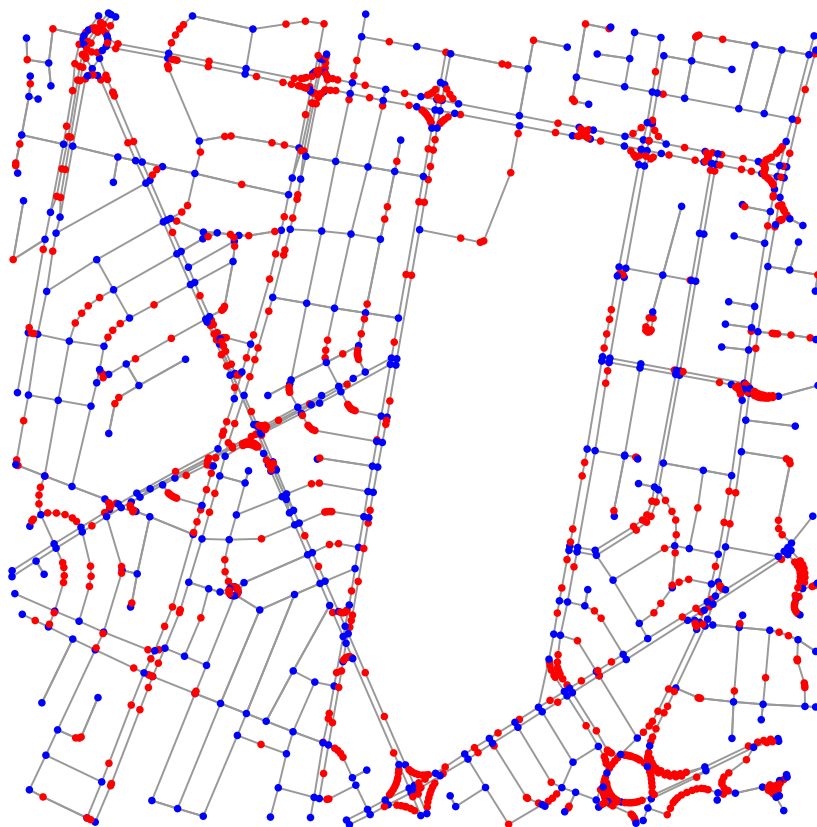


Figura 3. Ejemplo de simplificación sobre la zona del parque La Carolina.

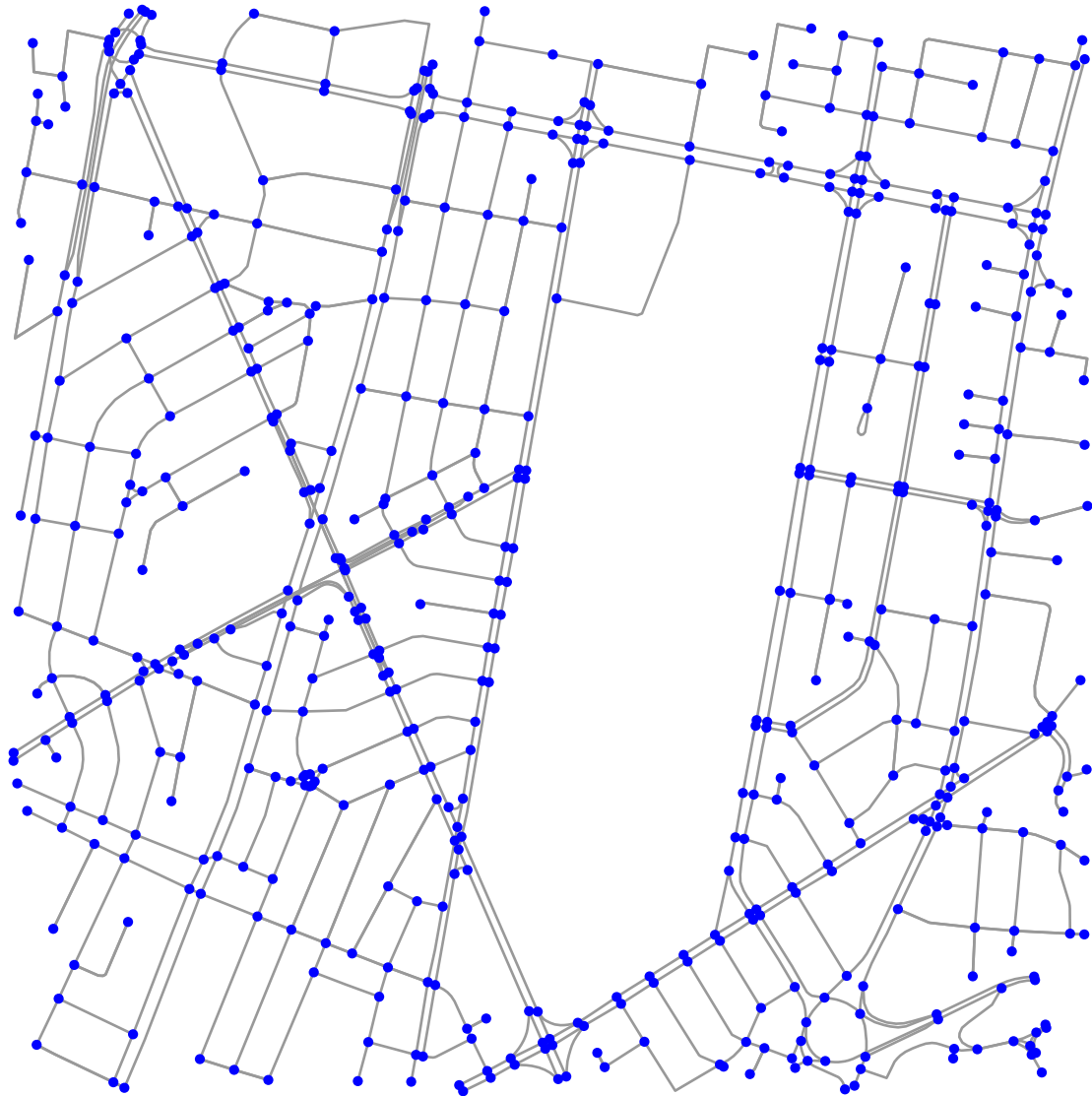


Figura 4. Resultado de la simplificación sobre la zona del parque La Carolina.

3.4. Recorte de red por distancia

Una vez se obtiene la red simplificada J , la red es recortada dependiendo de la distancia entre los nodos D_{ij} . El propósito de este proceso es extraer los nodos de interés de la red, en este caso los extremos de la ciudad, así como también, las distancias entre ellos. Los nodos son removidos de acuerdo con el criterio $C_{ij} = 0$ para $D_{ij} < \theta$. Este proceso remueve las conexiones entre los nodos i y

los nodos j , en cada valor de D_{ij} que es menor al valor de θ . El valor de θ corresponde a la distancia límite que se usa para realizar el recorte.

En este caso de estudio se toma como medidas discriminantes el uso de la media aritmética y el percentil 97, por lo cual estas se establecen como el valor de θ , y se realiza un proceso de recorte por cada una de estas medidas, por lo cual se comparan todas las conexiones de la red y se eliminan las conexiones menores al valor de θ . Los nodos sin conexiones son removidos de la red, manteniendo en su mayoría las conexiones pertenecientes a las calles y avenidas principales que interconectan la ciudad, siendo estos nodos puntos en los extremos del área metropolitana de Quito.

3.5. Winner-Takes-All

Una vez se obtienen los nodos de la red recortada, se construye una red totalmente conectada utilizando *Directions API*, la cual nos brinda información real entre dos puntos geográficos. Para esto se cogen los nodos sobrantes como puntos de origen y destino, se establece el modo de viaje en conducción y del resultado de la consulta se toma el parámetro de distancia, de esta manera se crea una red totalmente conectada, representada de la siguiente forma: $C_{ij} = 1, \forall ij$. De igual manera, se excluyen las conexiones sobre un mismo nodo y la matriz de distancia se vuelve asimétrica. La red totalmente conectada que se obtiene se aprecia en la Figura 5.

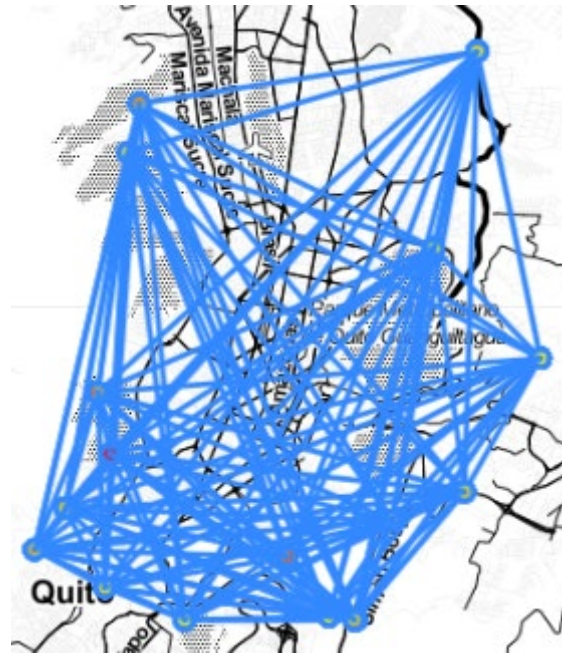


Figura 5. Red del área metropolitana de la ciudad de Quito totalmente conectada.

Luego la red es recortada de acuerdo con la distancia máxima conectando dos nodos: $C_{ij} = 1$ para $\max[\max(D_{ij}, D_{ji}) \text{ para } j \in k(i)]$. La única conexión que se mantiene en cada nodo i es la conexión con distancia máxima del nodo en su vecindario $k(i)$. El resto de las conexiones son removidas en un aproximamiento al método winner-takes-all. Siguiendo el criterio de recorte mencionado, las conexiones de la red son consideradas no direccionales. La red resultante conecta los puntos extremos de la ciudad y puede ser utilizada para identificar el tiempo que toma atravesar la ciudad.

En primer lugar, es importante mencionar que el nombre *Winner Takes All* es un término utilizado en redes neuronales, según Dlugosz et al. (2010), es la forma de referirse a la necesidad de eliminar neuronas muertas que no participan en el proceso de aprendizaje, siendo una carga que pueda generar errores que perjudiquen el rendimiento de la red. Para este proyecto se adjudicó dicho nombre debido a la similitud de la red de tráfico resultante con las redes neuronales y la necesidad de eliminar los nodos que no proporcionan valor al

caso de estudio. Sin embargo, cabe recalcar que el algoritmo de este proyecto, a pesar de llevar el mismo nombre, es creación propia de los autores y está dirigido a redes de tráfico vehicular.

De una forma más general, el algoritmo *Winner Takes All* creado en este trabajo, se realizó con la finalidad de ser un opuesto al conocido algoritmo de Dijkstra, pero manteniendo sus bases fundamentales en cuanto a la discriminación de conexiones. Su aplicación se realiza mediante la eliminación de todos los caminos posibles de un nodo, manteniendo solo el camino que tenga el peso más alto, que en este caso es la distancia más larga, siendo contradictorio a Dijkstra, y asegurando que de esta forma cada nodo participante en la red sea punto de partida, a la vez que puede ser punto de destino o no.

De esta manera se obtiene una interconexión entre todos los puntos del mapa, abarcando toda la ciudad, sin embargo, sin ser simétrica ya que algunos puntos proporcionan mayor cantidad de conexiones que otros, lo cual geográficamente hablando se puede observar que está representando correctamente la topología de la ciudad y sus respectivas avenidas que la interconectan.

El resultado esperado de este algoritmo se puede observar en la Figura 6, donde los nodos amarillos son puntos de partida, los nodos rojos de destino y los naranjas tanto partida como destino.

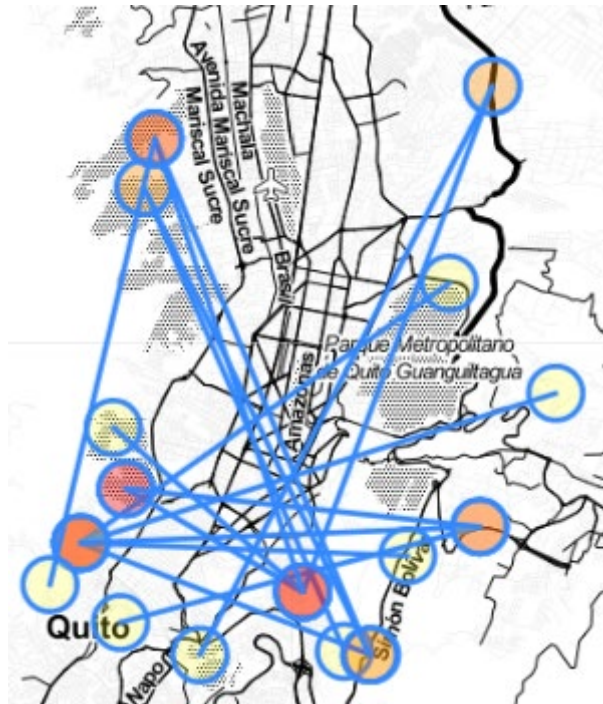


Figura 6. Red del área metropolitana de Quito luego de aplicar el algoritmo Winner Takes All.

3.6. Extracción del flujo de tráfico

Una vez se completa el proceso de extracción de la red, es necesario obtener la información del flujo de tráfico. Para esto se utiliza *Distance Matrix API*, la cual es un servicio proporcionado por Google que devuelve información de distancia y tiempo de viaje basado en la ruta recomendada entre dos puntos. El resultado obtenido se encuentra en formato JSON. Las consultas se realizaron tomando en cuenta que, el punto de inicio y de llegada corresponden a los nodos de origen y destino de cada conexión. Estas consultas se ejecutaron cada cierto tiempo de manera autónoma durante varios días, de esta forma se obtiene una mayor cantidad de datos y luego, al procesar estos datos se puede tener estimados más reales de tiempo. Cada registro, para el flujo de tráfico, guarda la información con la estructura mostrada en la Tabla 1, la cual tiene los siguientes atributos, ID el cual es un identificador de cada consulta realizada, Date que contiene la fecha y hora en la que se realizó la consulta, ORG aquí están las

coordenadas del punto de origen de la consulta, DEST es la coordenada de destino de la consulta, DIST es la distancia en metros entre los puntos de origen y destino de la consulta, esta distancia puede variar según la ruta más óptima que la API de Google nos proporciona, y por último TIME que es el tiempo que se demora en viajar del punto de origen al punto de destino.

Tabla 1.

Estructura de recolección de datos.

ID	DATE	ORG	DEST	DIST	TIME
1	2019-12-21 19:36:26	-0.235261, -78.4715368	-0.1370769, -78.5124341	22599	2194

Nota: Date = Fecha y hora de recolección de datos; ORG = Punto de origen en la consulta realizada; DEST = Punto de destino en la consulta realizada; DIST = Distancia en metros de la mejor ruta obtenida en la consulta; TIME = Tiempo en segundos que demora el viaje por la ruta escogida.

Sobre esta información se realizó un proceso mediante el cual se resumió los datos recopilados todos los días, para sacar un promedio de la distancia y tiempo que toma el recorrido de un punto a otro, lo cual se estableció como un punto de inicio para un plan de ciudad sostenible ya que por su propia naturaleza ya describe lugares de importancia y los valores relacionados a las restricciones existentes como el tiempo y distancia. Sin embargo, para una mejor comprensión de los datos, además de otorgar información más significativa, se realiza un análisis de clústeres, debido a que este es el punto óptimo de partida para la implementación de planes de ciudad sostenible sobre el área metropolitana de Quito. Previo a este análisis de clústeres se tiene una tabla de especificación de los datos y procesos realizados hasta este punto.

En la Tabla 2 se explican varios atributos de los datos y como se obtuvieron. En la columna de la izquierda están los títulos y en la de la derecha una breve explicación de lo que se hizo.

Tabla 2.

Especificación de datos.

Adquisición de datos	Proceso de simplificación a través de una red extraída de OSMNX y llamadas continuas a la API de matriz de distancia proporcionada por Google, con intervalos de treinta minutos.
Formato de Datos	Datos en bruto
Parámetros para la recolección de datos	Los autores realizaron una serie de procesos automáticos y manuales para simplificar una red de la ciudad de Quito extraída con la biblioteca OSMNX. Los datos obtenidos se utilizaron para obtener coordenadas de los puntos más importantes para el caso de estudio, y estas coordenadas fueron los puntos de inicio y fin utilizados en las consultas enviadas a la API, cuyos datos resultantes se registraron automáticamente en un archivo csv.
Descripción de la recolección de los datos	<p>Para el proceso de simplificación de la red de la ciudad, los datos se recopilaron de la siguiente manera.</p> <p>La primera fase implicó una simplificación automática proporcionada por la biblioteca OSMnx, en la que se eliminaron todos los nodos individuales que componen la calle y sus pesos se unieron como una conexión desde el punto inicial y final de las calles.</p>

En la segunda fase, los datos resultantes se simplificaron utilizando la media del peso como discriminante para mantener las conexiones cuyo peso es mayor que el valor obtenido.

En la fase tres, se realizó un proceso similar, pero utilizando el percentil noventa y siete como valor discriminante. Para la cuarta fase, los datos resultantes tenían muchas conexiones con el mismo punto de inicio, para resolver esto, se realizó un proceso llamado *winner takes all* que solo mantiene la conexión con mayor peso de cada punto de partida, y agrega un identificador único a cada punto resultante.

Finalmente, en la última fase, las coordenadas de los puntos inicial y final de las conexiones restantes se obtuvieron y se enviaron a la API de *Google Distance Matrix* para obtener valores de tráfico en tiempo real cada treinta minutos, y se guardaron automáticamente en un archivo csv.

Fuente de los datos Open Street Maps y Google Maps

Nota: Esta tabla describe a breves rasgos el proceso realizado hasta el momento con los datos obtenidos.

3.7. Clustering y mapeo del flujo de tráfico

Los sitios se agrupan mediante su proximidad geográfica, reduciendo las dimensiones de la red de flujo de tráfico y resumiendo el promedio de los tiempos de viaje entre direcciones cardinales de una ciudad. La información de las coordenadas es proyectada utilizando el WGS84. Las coordenadas proyectadas

son luego estandarizadas, debido a que las técnicas de agrupamiento son sensibles a las escalas de las variables.

Una vez las distancias son calculadas para cada par de puntos, se realiza un agrupamiento jerárquico utilizando la afinidad de distancia euclidiana y el método de Ward. Luego de que se identifican los clústeres mediante los métodos mencionados, la información del flujo de tráfico se puede resumir (promediar) sobre la conectividad de los clústeres. Esto se realiza al comparar el clúster de origen y destino de las conexiones, y promediando el tiempo de las conexiones de los nodos pertenecientes a los clústeres a comparar. Los clústeres se pueden observar en la Figura 7.

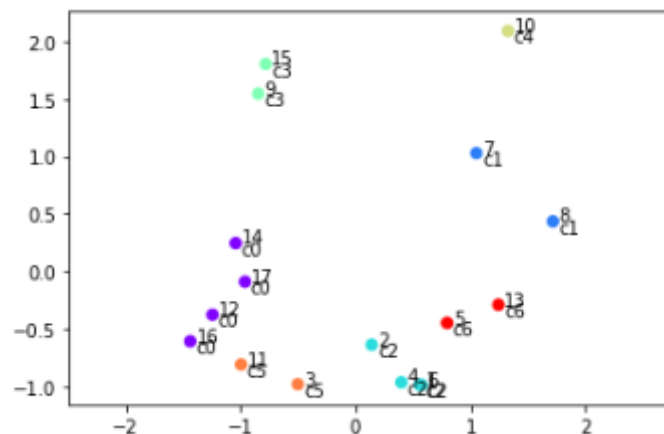


Figura 7. Representación visual de los clústeres.

4. RESULTADOS

4.1. Grafo resultante mediante OSMnx

El proceso de extracción de la red arroja un grafo completo, pero sumamente grande que representa la red del área metropolitana de Quito, llegando a un

máximo detalle de crear nodos y conexiones diminutas que conforman parte de una sola calle grande o avenida. Con el proceso de simplificación realizado utilizando métodos propios de OSMnx para manejar estas pequeñas conexiones, se redujo considerablemente la cantidad de nodos y conexiones existentes, sin embargo, el grafo obtenido todavía tiene demasiada información que se requiere filtrar según el caso de estudio. El mapa resultante de este primer proceso se observa en la Figura 8.



Figura 8. Resultado de simplificación con OSMnx sobre el área metropolitana de Quito.

Como se puede visualizar en la figura mostrada anteriormente, a pesar de la simplificación realizada, se mantiene una gran cantidad de información, obteniendo un total de 9.469 nodos y 21.133 conexiones. La cantidad obtenida está conformada a su vez por grandes cantidades de datos otorgados por OSMnx, los cuales representan información de utilidad en muchos casos e información que podría no ser necesaria en el caso de estudio que se aborda, es decir, se puede prescindir de ciertos datos que incluso vienen vacíos al momento de retornar la consulta.

La estructura del grafo retorna información tanto en los nodos como en las conexiones, permitiendo distinguirlos principalmente por un identificador único conocido como osmid. Este identificador permite conocer con cuales nodos se formaron las conexiones resultantes.

En la siguiente tabla se muestra como está estructurada la información de los nodos que nos brinda OSMnx, en la cual OSMID es un identificador proporcionado por la librería, Y corresponde a la latitud en la que se encuentra el nodo, X corresponde a la longitud en la que se encuentra el nodo, HIGHWAY nos dice el tipo de avenida en la que está ubicado el nodo y REF nos brinda información relevante del nodo, en caso de que exista.

Tabla 3.

Estructura de nodos resultantes del proceso de simplificación.

OSMID	Y	X	HIGHWAY	REF
4247650314	-0.1851197	-78.4886854	-	-
3061841953	-0.2322382	-78.5236394	turning_circle	-

Nota: OSMID = Identificador único para cada nodo proporcionado por OSMnx; Y = Latitud; X = Longitud; HIGHWAY = Tipo de avenida; REF = Información relevante sobre el punto en caso de existir.

Con una estructura similar se recopiló los datos más importantes para el estudio en base a la basta información que OSMnx devuelve dentro de las conexiones simplificadas. La información mostrada en la Tabla 4 prescinde de varia información adicional como nombre, tipo de avenida, referencia, velocidad máxima, túnel, líneas y geometría. Esta información no se consideró parte del estudio y se la excluyó ya que son datos generales y promediados, siendo contrario al objetivo del estudio el cual es la búsqueda de datos en tiempo real para realizar un correcto análisis de tráfico que permitan desarrollar un modelo adecuado para el área metropolitana de Quito. La información que se presenta en la Tabla 4 es la usada para los procesos realizados luego del proceso de simplificación.

En la Tabla 4, se muestra la estructura de las conexiones que presentan los nodos, en donde ID es el identificador que se le brinda a la conexión, SOURCE es el id que proporciona OSMnx al nodo de origen, TARGET es el id con el cual OSMnx identifica al nodo de destino, OSMID recopila los id de los nodos que forman la conexión y LENGTH es la distancia entre el nodo de origen y el nodo de destino.

Tabla 4.

Estructura de conexiones resultantes del proceso de simplificación, información principal.

ID	SOURCE	TARGET	OSMID	LENGTH
15	269221967	6973456802	[492187418, 41097125, 533146198]	1720.817

98	5165154564	6326303575	[532924995, 532228590]	224.9299
----	------------	------------	---------------------------	----------

Nota: ID = Identificador de cada conexión; SOURCE = OSMID del nodo inicial; TARGET = OSMID del nodo de destino; OSMID = Identificadores de los nodos que ahora forman parte de esa conexión; LENGTH = Distancia de la conexión expresada en metros.

4.2. Red resultante de la simplificación aritmética y Winner Takes All

Con la gran cantidad de datos resultantes y su estructura con información relevante y no relevante, se procedió a realizar los procesos de simplificación mediante la utilización de la media y el percentil 97 como discriminantes, de donde se obtuvo una red más pequeña y manejable, pero con varias conexiones innecesarias.

La resolución de este problema se realizó mediante la creación e implementación del algoritmo *Winner Takes All*, el cual permitió reducir drásticamente las conexiones, dando como resultado final 17 nodos, los cuales, actuando como punto de partida, destino o ambos en algunos casos, permitió generar 16 conexiones que son capaces de conectar el área metropolitana de Quito de extremo a extremo. La red como tal se muestra de forma gráfica mediante NetworkX en la Figura 9.

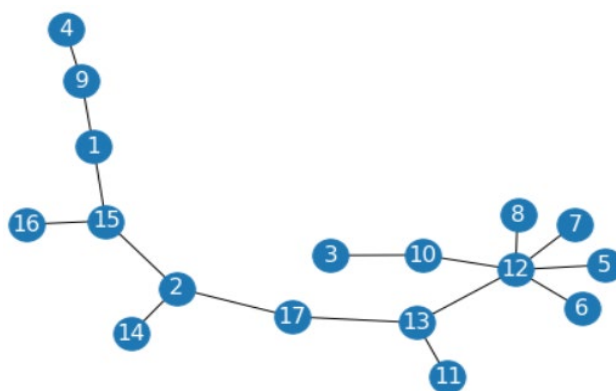


Figura 9. Gráfico de la red resultante del área metropolitana de Quito realizado en NetworkX.

Para el manejo adecuado de estos nodos durante el resto del proyecto se les asigno un identificador único en base a sus coordenadas, identificando así el nodo sobre el que se está manejando en todo momento, además de su ubicación geográfica e interacción a través de las conexiones. En la Tabla 5 se observa los identificadores asignados a cada nodo. En esta tabla está estructurada de tal manera que en la columna TODO se encuentren las coordenadas geográficas de los nodos, y en la columna UNQID un identificador único para cada nodo.

Tabla 5.

Asignación de identificadores por coordenadas únicas.

TODO	UNQID
-0.235261,-78.4715368	1
-0.2229887,-78.4842786	2
-0.2350708,-78.5038906	3
-0.2344857,-78.4763959	4
-0.2163172,-78.464282	5
-0.2352449,-78.4710614	6
-0.1643507,-78.4565223	7
-0.1852329,-78.4362088	8
-0.1461535,-78.5144057	9
-0.1269491,-78.4481262	10
-0.2290423,-78.5189432	11
-0.2138075,-78.5265402	12
-0.2107426,-78.4506721	13
-0.1919044,-78.5203769	14
-0.1370769,-78.5124341	15

-0.2219491,-78.5324428	16
-0.2037,-78.5179089	17

Nota: TODO = Coordenadas de cada nodo; UNQID = Identificador único para cada nodo;

4.3. Información en tiempo real

La información referente a los nodos y conexiones, así como su respectiva asignación de identificadores únicos se utilizó para consultar a *Google Maps* por datos de tráfico en tiempo real. Para esto se generó un script que se ejecute de manera continua, consultando cada 20 minutos, durante 6 días, el estado del tráfico en las rutas que permiten movilizarse de extremo a extremo del área de interés establecida en el estudio. Se utilizaron estos valores con el fin de tener la mayor cantidad de datos posibles con el presupuesto con el que se contaba.

Constantemente se almacenó en un archivo csv los datos de las 16 conexiones existentes entre los 17 nodos que componen la red. En la Tabla 6 se muestra la estructura final en la que se presentan los datos obtenidos durante la fase de recopilación.

Los datos de esta tabla están estructurados de la siguiente manera: ID es un identificador único para cada consulta, DATE es la fecha y hora en la que se realizó la consulta, ORG es la coordenada del punto de origen de la consulta, DEST es la coordenada del punto de destino de la consulta, DIST es la distancia en metros entre los puntos de origen y destino, TIME es el tiempo en el que se tarda en llegar un vehículo desde el punto de origen al punto de destino, al momento de la consulta, UNQ_ID_ORG es el identificador único del nodo de origen, UNQ_ID_DEST es el identificador único del nodo de destino.

Tabla 6.

Formato de presentación de datos recopilados durante la fase de recolección.

ID	DATE	ORG	DEST	DIST	TIME	UNQ ID_ ORG	UNQ ID_ DES T
0	2019- 12-21 19:36: 26	-0.235261,- 78.4715368	-0.1370769,- 78.5124341	22599	2194	1	15
1	2019- 12-21 19:36: 26	-0.2229887,- 78.4842786	-0.2037,- 78.5179089	10124	1848	2	17
2	2019- 12-21 19:36: 26	-0.2350708,- 78.5038906	-0.1269491,- 78.4481262	26507	1768	3	10
3	2019- 12-21 19:36: 26	-0.2344857,- 78.4763959	-0.1461535,- 78.5144057	23250	2432	4	9
4	2019- 12-21 19:36: 26	-0.2163172,- 78.464282	-0.2138075,- 78.5265402	20917	2203	5	12
5	2019- 12-21 19:36: 26	"-0.2352449,- 78.4710614	-0.2138075,- 78.5265402	25759	2342	6	12
6	2019- 12-21 19:36: 26	-0.1643507,- 78.4565223	-0.2138075,- 78.5265402	25609	2320	7	12
7	2019- 12-21 19:36: 26	-0.1852329,- 78.4362088	-0.2138075,- 78.5265402	25942	2623	8	12
8	2019- 12-21 19:36: 26	-0.1461535,- 78.5144057	-0.235261,- 78.4715368	21545	2085	9	1
9	2019- 12-21 19:36: 26	-0.1269491,- 78.4481262	-0.2138075,- 78.5265402	32056	2857	10	12

10	2019- 12-21 19:36: 26	-0.2290423,- 78.5189432	-0.2107426,- 78.4506721	18765	2081	11	13
11	2019- 12-21 19:36: 26	-0.2138075,- 78.5265402	-0.2107426,- 78.4506721	21201	2573	12	13
12	2019- 12-21 19:36: 26	-0.2107426,- 78.4506721	-0.2037,- 78.5179089	20307	2492	13	17
13	2019- 12-21 19:36: 26	-0.1919044,- 78.5203769"	-0.2229887,- 78.4842786	13881	2219	14	2
14	2019- 12-21 19:36: 26	-0.1370769,- 78.5124341	-0.2229887,- 78.4842786	20990	2686	15	2
15	2019- 12-21 19:36: 26	-0.2219491,- 78.5324428	"-0.1370769,- 78.5124341	18451	2096	16	15

Nota: ID = Identificador de cada consulta, referente a cada conexión; DATE = Fecha y hora de recolección de datos; ORG = Coordenadas del punto de origen; DEST = Coordenadas del punto de destino; DIST = Distancia de viaje expresada en metros; TIME = Tiempo de viaje expresado en segundos; UNQ_ID_ORG = Identificador único del nodo de origen; UNQ_ID_DEST = Identificador único del nodo de destino.

4.4. Perfilado

Con los resultados en tiempo real obtenidos se realizó un análisis del comportamiento del tráfico en la ciudad según la hora del día, para lo cual se decidió crear un promedio en base a esta información fidedigna, el cual indique la distancia y tiempo aproximado que se demora generalmente un vehículo en viajar por cada una de las 16 conexiones resultantes, obteniendo los resultados mostrados en la Tabla 7, la tabla está estructurada de la siguiente manera: la columna FROM cuenta con el id de cada nodo de origen, la columna TO cuenta

con el id del nodo de destino, la columna TIME_WEIGHT (min) cuenta con el tiempo promedio en minutos entre el nodo de origen y de destino y la columna DIST_WEIGHT (Km) cuenta con la distancia en kilómetros entre el nodo de origen y el nodo de destino.

Tabla 7.

Resultados de tiempo y distancia de viaje en base a información real.

FROM	TO	TIME_WEIGHT (min)	DIST_WEIGHT (Km)
4	9	38.75916666666667	24.1022125
9	1	32.925208333333334	21.631679166666665
15	2	36.361111111111111	21.349108333333334
8	12	42.789236111111111	23.6545375
7	12	38.014444444444444	24.430349999999997
12	13	40.463541666666664	21.308529166666666
5	12	36.560069444444444	20.8906
3	10	26.662013888888889	23.69145
10	12	44.346180555555556	26.263620833333334
1	15	34.169513888888886	23.361229166666668
13	17	40.145138888888894	16.7770875
16	15	32.323194444444445	18.537070833333335
6	12	38.955416666666665	24.662141666666667
14	2	28.380555555555553	14.119508333333334
2	17	30.535	10.355579166666667
11	13	31.537152777777778	18.737416666666668

Nota: FROM = Identificador único del nodo de partida; TO = Identificador único del nodo de destino; TIME_WEIGHT = Tiempo aproximado expresado en minutos que toma realizar el viaje; DIST_WEIGHT = Distancia aproximada que se recorre en el viaje.

Las rutas sobre las que se desenvuelven las conexiones mostradas anteriormente representan una interconexión total entre el área metropolitana de Quito, a continuación, se muestra la Figura 10, donde los puntos verdes hacen referencia a los puntos de partida, los rojos puntos de llegada y los amarillos puntos que actúan tanto como inicio y destino.

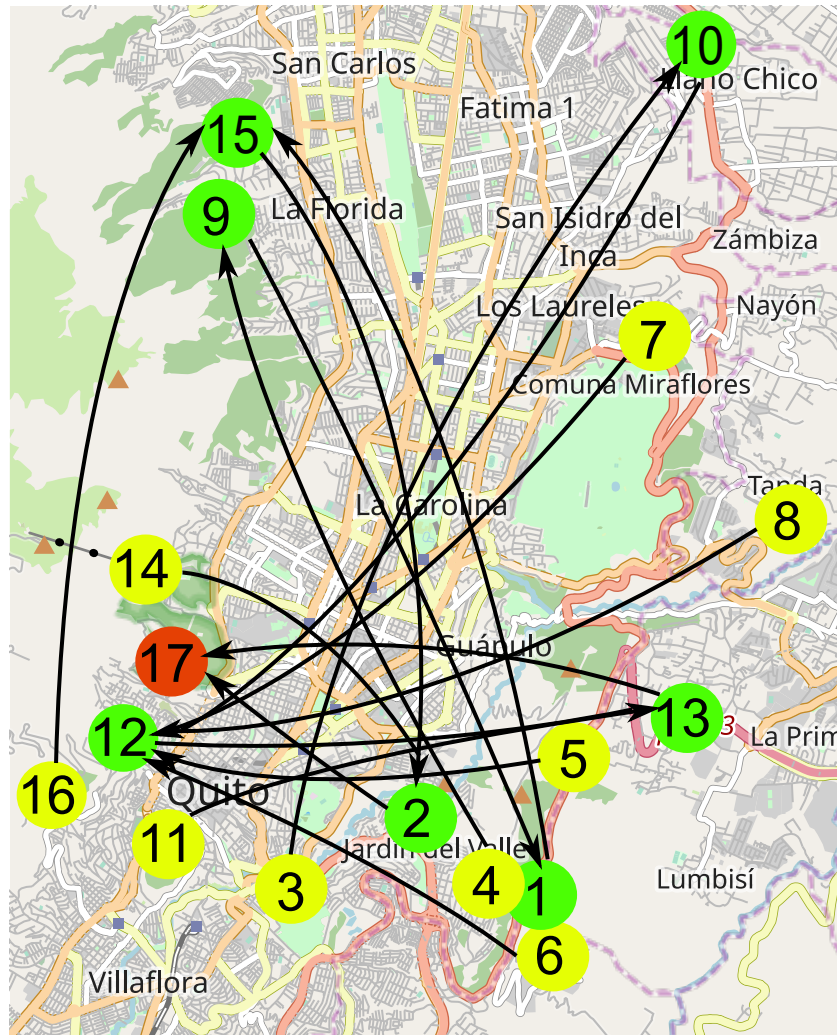


Figura 10. Representación geográfica de los puntos resultantes para el caso de estudio.

4.4.1. Representación Gráfica del perfilado

Mediante el análisis realizado se desarrolló un gráfico que representa correctamente las horas con más tráfico durante el transcurso del día, como siendo este uno de los primeros entregables valiosos del proyecto de investigación realizado, ya que este ya es un buen punto de inicio para un plan de ciudad sostenible al saber los horarios en los que existe mayor afluencia de vehículos en los puntos seleccionados. Con esta información y perfilado se puede determinar que los horarios más transitados son alrededor de las siete de

la mañana y las ocho de la noche, dónde las personas viajan de sus casas a sus trabajos y viceversa. Esto se visualiza en la Figura 11.

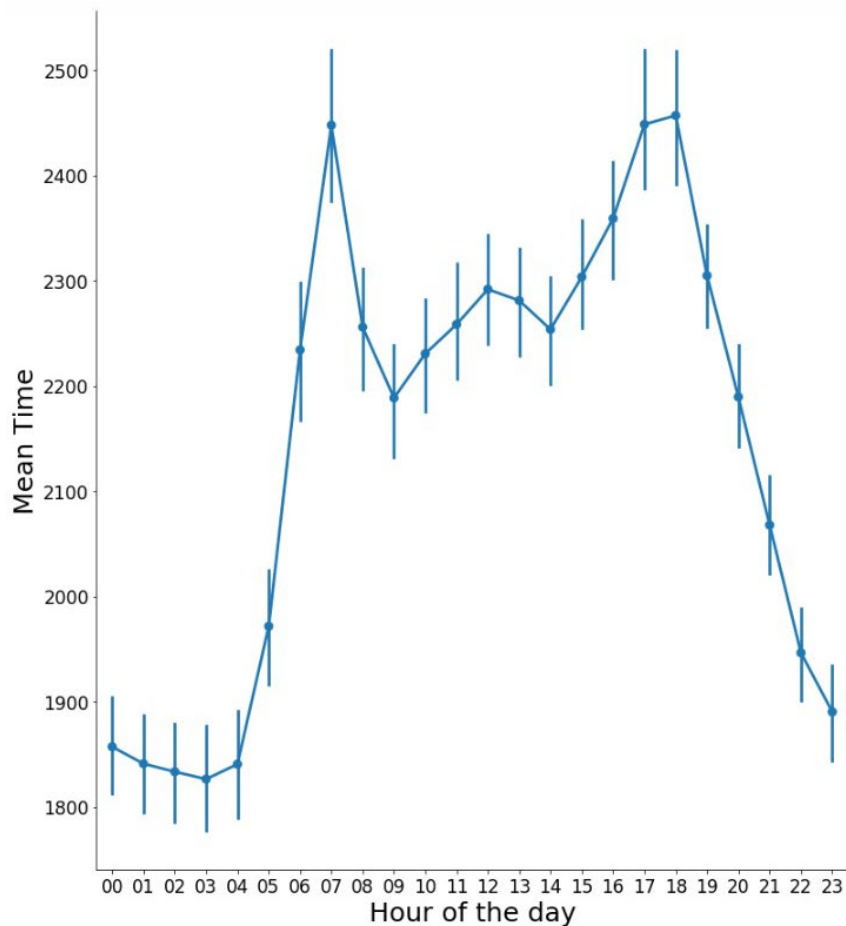


Figura 11. Perfil de flujo de tráfico vehicular en el área metropolitana de Quito en base a horas y tiempo que toma llegar al destino.

4.5. Clústeres Geográficos

El proceso de agrupación por clústeres se realizó en base a su posicionamiento geográfico, dónde se prueba distintos tipos de agrupaciones hasta encontrar una que separe correctamente las zonas más importantes de la ciudad. Como recomendación generalmente se menciona que un agrupamiento de clústeres puede generar buenos resultados al realizarse en 4 grupos, sin embargo, como

se muestra en la Figura 12, este tipo de agrupamiento no sectoriza correctamente la ciudad, dado que agrupa ubicaciones muy lejanas, por ejemplo, el nodo 10 con los nodos 7 u 8, el nodo 13 con los nodos 2, 4, 1, 6, el nodo 3 con los nodos 14 y 17.

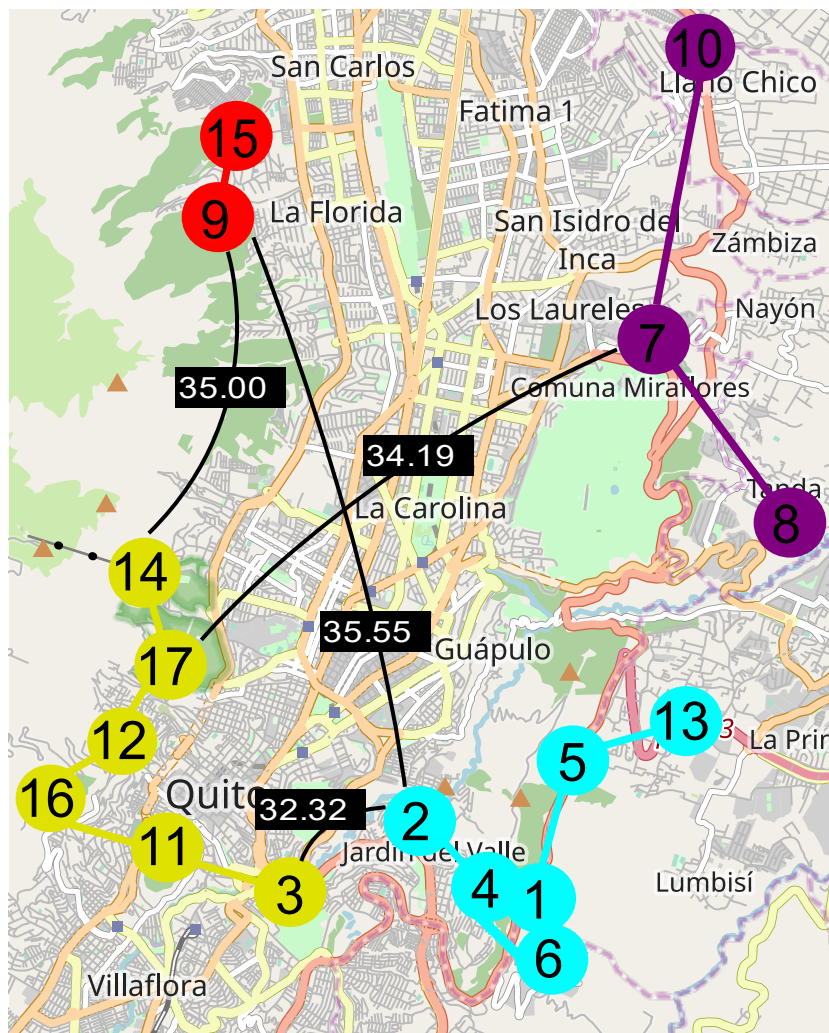


Figura 12. Agrupación de 4 clústeres y promedio de tiempo de viaje entre estos.

Siguiendo el análisis realizado, se determinó que el mejor modelo posible es el que involucra 7 clústeres, ya que sus resultados permiten la mejor sectorización posible en base a la ubicación geográfica y las vías de acceso en el área metropolitana de Quito. Para la generación final del gráfico se utiliza *Inkscape* como herramienta de vectorización mediante su línea de comandos que permite

generar un mejor contenido SVG para la presentación en documentos científicos. Este resultado se puede observar en la Figura 13.

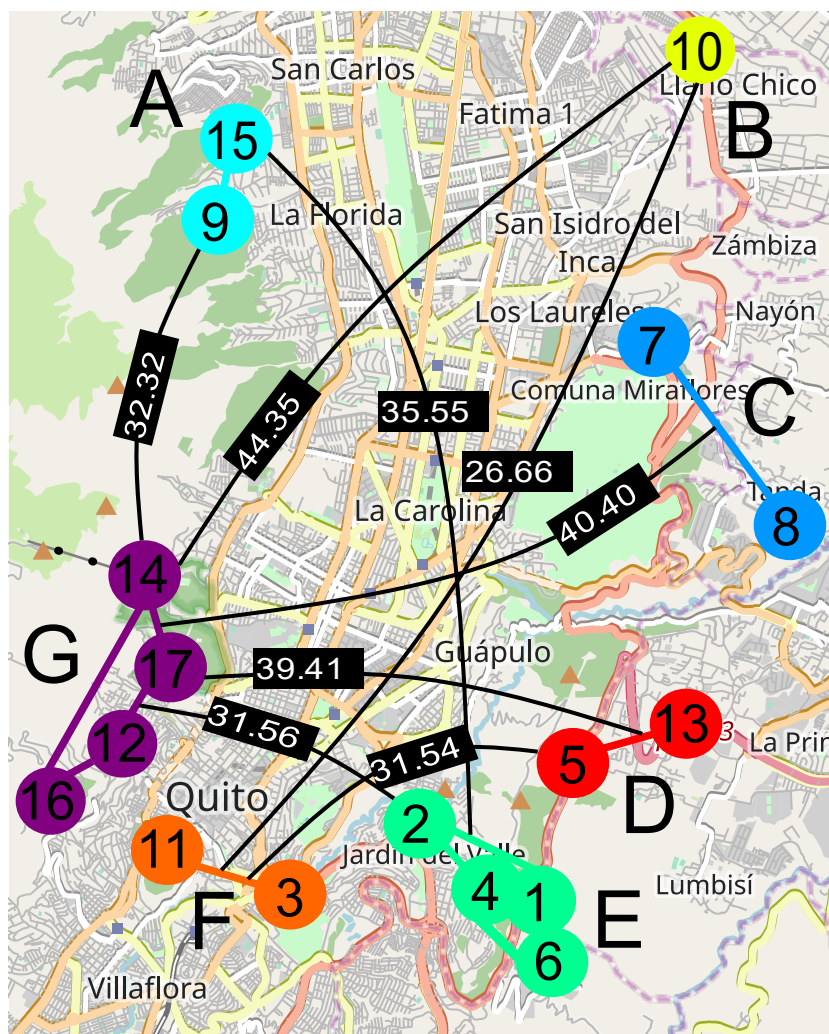


Figura 13. Resultado final con 7 clústeres y sus respectivos promedios de tiempo de viaje entre clúster.

4.6. Análisis entre clústeres

Sobre los resultados obtenidos se realizó un análisis donde se presenta una tabla con toda la información necesaria referente a las conexiones, permitiendo observar de manera detallada los datos presentados de forma gráfica en la Figura 12. A continuación se presenta la Tabla 8 con la información necesaria

para comprender el resultado del proceso de agrupación por clústeres. Esta tabla está estructurada de tal manera que la columna TODO tiene las coordenadas geográficas de cada nodo, la columna UNQ_ID tiene el identificador único que corresponde a cada nodo, la columna CLUSTER tiene el número del clúster al que pertenece cada nodo y la columna LETTERS_ID tiene una letra que identifica a cada clúster.

Tabla 8.

Detalle de agrupación por clústeres.

TODO	UNQ_ID	CLUSTER	LETTERS_ID
-0.235261,-78.4715368	1	2	E
-0.2229887,-78.4842786	2	2	E
-0.2350708,-78.5038906	3	5	F
-0.2344857,-78.4763959	4	2	E
-0.2163172,-78.464282	5	6	D
-0.2352449,-78.4710614	6	2	E
-0.1643507,-78.4565223	7	1	C
-0.1852329,-78.4362088	8	1	C
-0.1461535,-78.5144057	9	3	A
-0.1269491,-78.4481262	10	4	B
-0.2290423,-78.5189432	11	5	F
-0.2138075,-78.5265402	12	0	G
-0.2107426,-78.4506721	13	6	D
-0.1919044,-78.5203769	14	0	G
-0.1370769,-78.5124341	15	3	A
-0.2219491,-78.5324428	16	0	G
-0.2037,-78.5179089	17	0	G

Nota: TODO = Coordenadas geográficas del nodo; UNQ_ID = Identificador único del nodo; CLUSTER = Número de clúster; LETTERS_ID = Letra con la que se representa en el mapa el clúster.

En la Tabla 9 se muestra el flujo entre los clústeres obtenidos, donde en la columna FROM están los clústeres de origen, en la columna TO se encuentran los clústeres de destino, la columna LETTER_FROM contiene las letras que representan al clúster de origen, la columna LETTER_TO cuenta con las letras

que representan al clúster de destino y la columna WEIGHT contiene los tiempos en minutos que toma llegar entre el tiempo de origen y de destino.

Estos datos se obtuvieron sacando el promedio de los tiempos de cada conexión existente entre los nodos que pertenecen al clúster de origen hacia los nodos que pertenecen a un mismo clúster de destino.

Tabla 9.

Detalle de los flujos entre los clústeres.

FROM	TO	LETTER_FROM	LETTER_TO	WEIGHT
3	2	A	E	35.55375
6	0	D	G	39.40807292
2	0	E	G	31.56288194
5	4	F	B	26.66201389
5	6	F	D	31.53715278
0	3	G	A	32.32319444
1	0	C	G	40.40184028
4	0	B	G	44.34618056

Nota: FROM = Clúster de origen; TO = Clúster de destino; LETTER_FROM = Letra con la que se representa en el mapa al clúster de origen; LETTER_TO = Letra con la que se representa en el mapa al clúster de destino; WEIGHT=El tiempo de viaje entre la conexión de origen y destino.

En base a estos datos y las conexiones entre clústeres se realizó un perfilado de cada una de las rutas, dónde se puede apreciar las horas pico de tráfico al movilizarse de un sector a otro en el área metropolitana de Quito a lo largo del día, como se muestra desde la Figura 14 a la Figura 21.

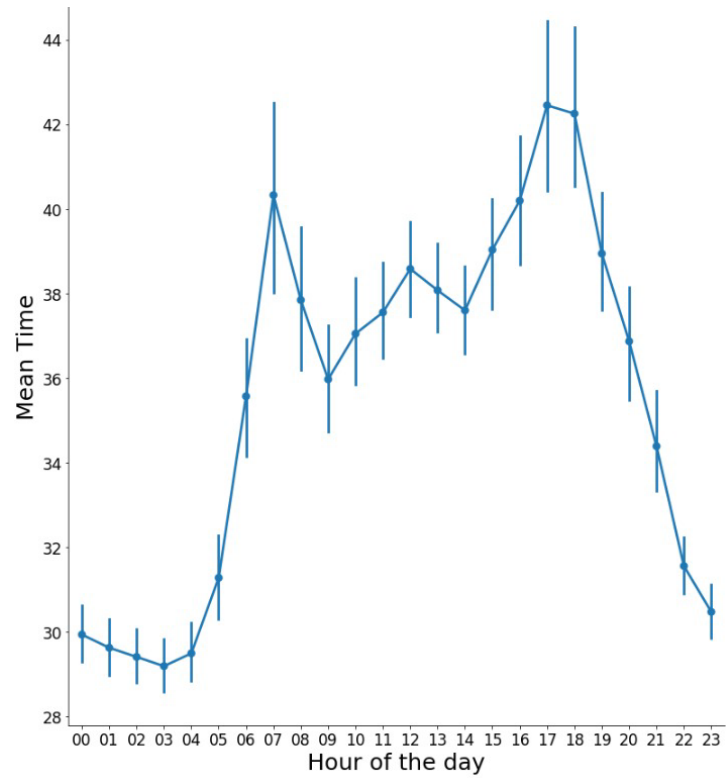


Figura 14. Perfilado de tráfico entre los clústeres A y E.

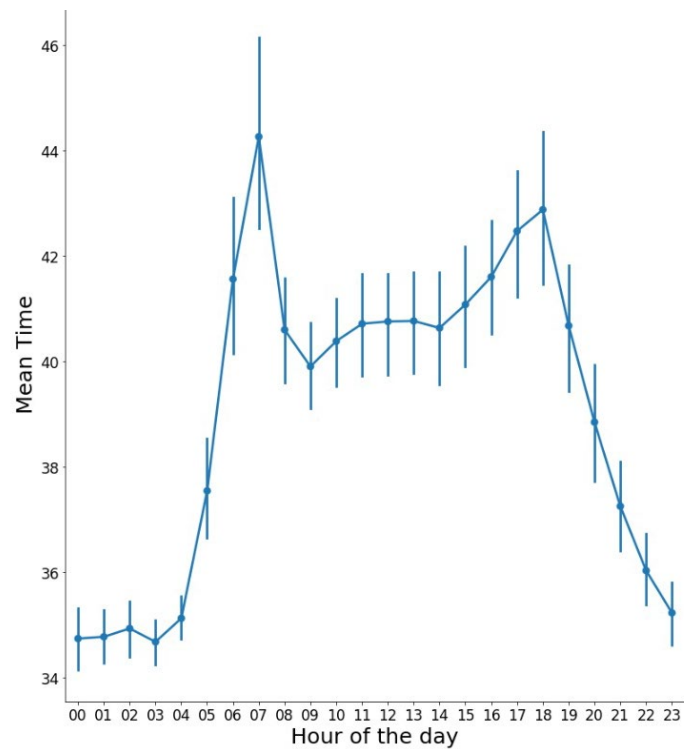


Figura 15. Perfilado de tráfico entre los clústeres D y G.

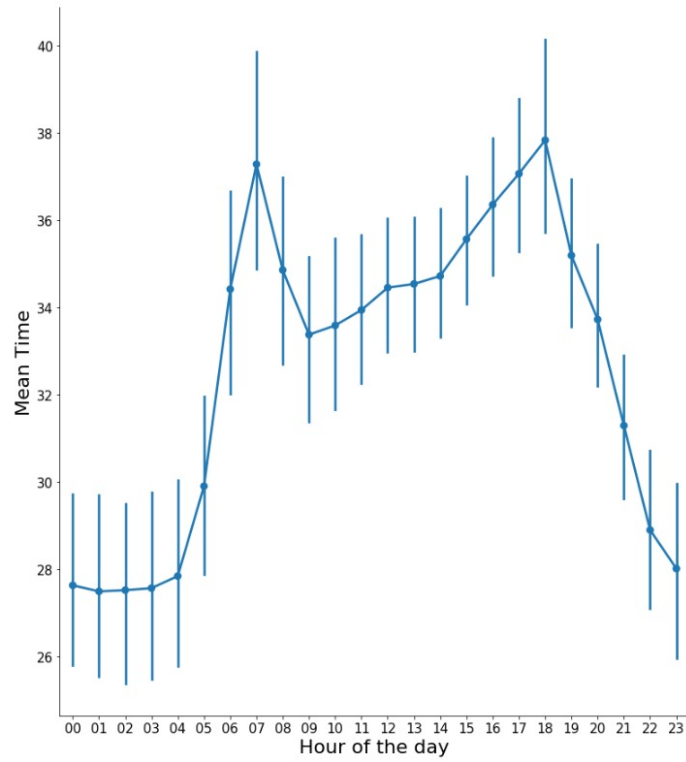


Figura 16. Perfilado de tráfico entre los clústeres E y G.

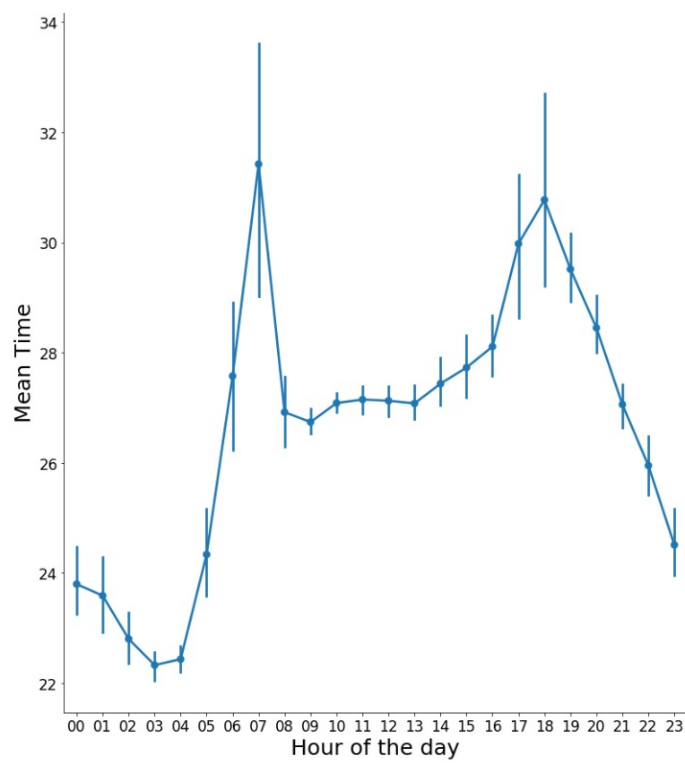


Figura 17. Perfilado de tráfico entre los clústeres F y B.

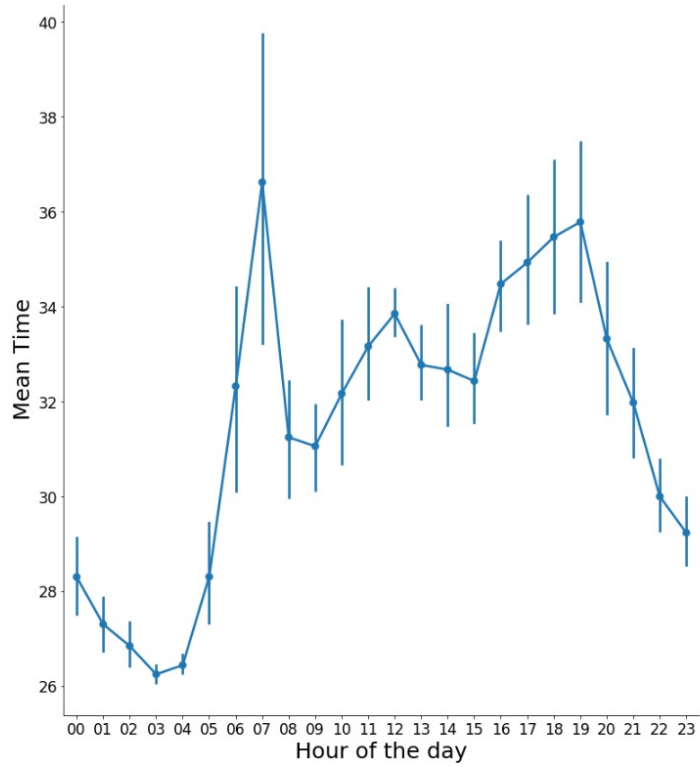


Figura 18. Perfilado de tráfico entre los clústeres F y D.

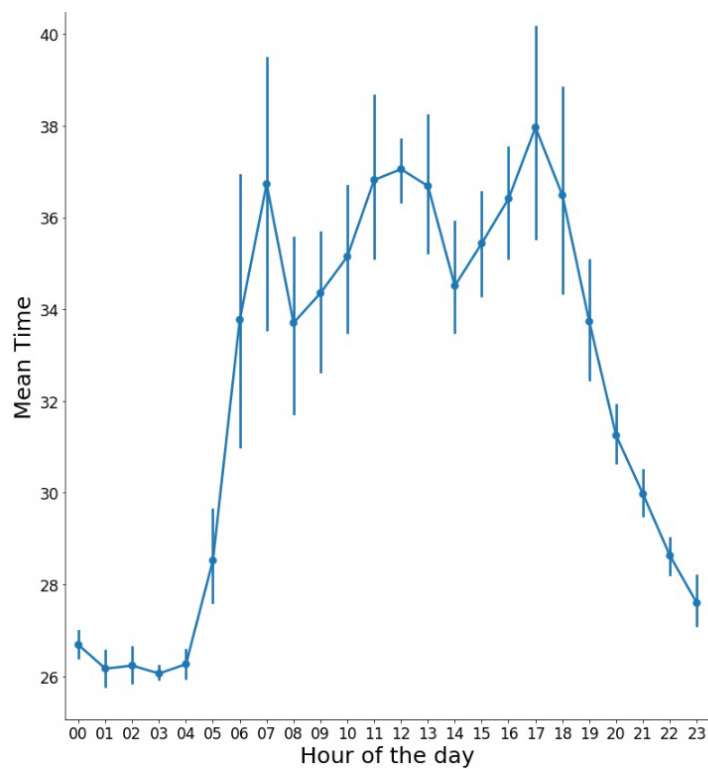


Figura 19. Perfilado de tráfico entre los clústeres G y A.

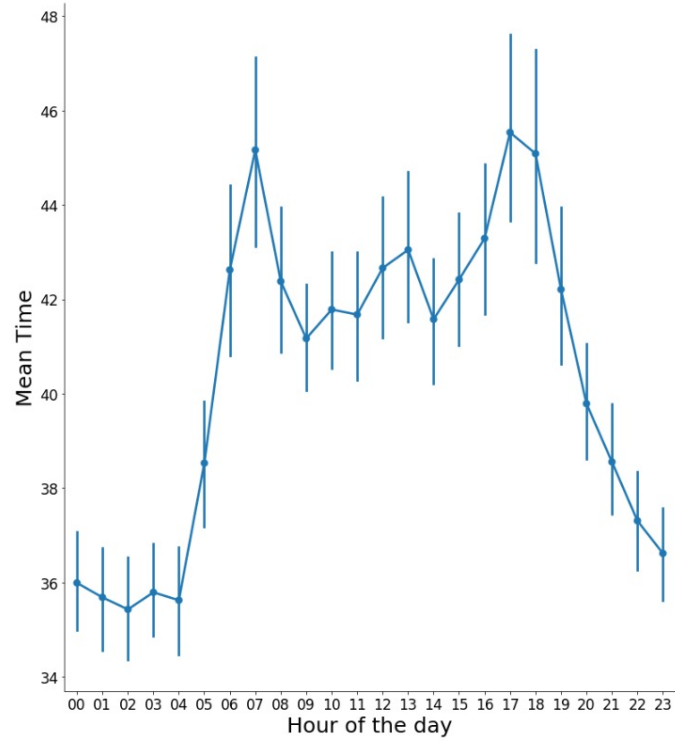


Figura 20. Perfilado de tráfico entre los clústeres C y G.

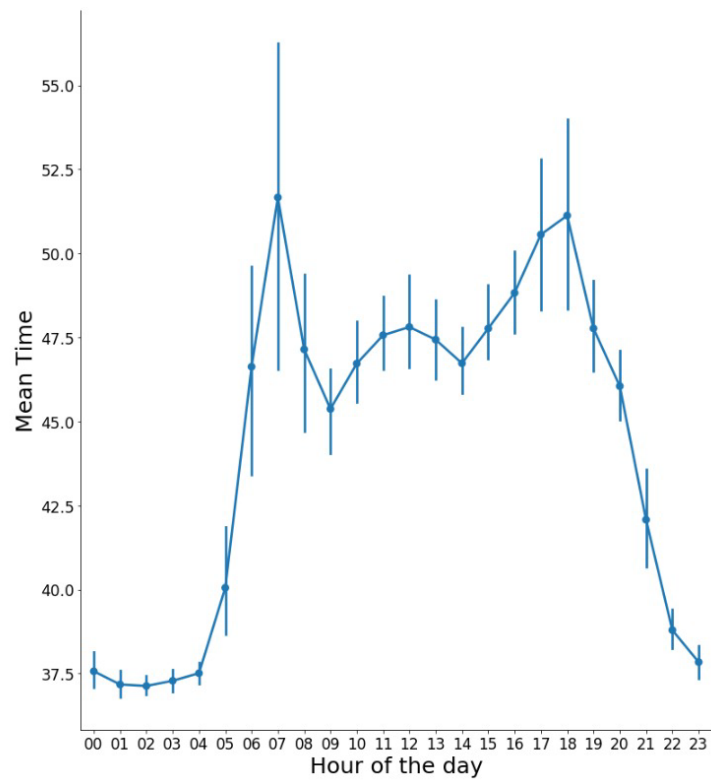


Figura 21. Perfilado de tráfico entre los clústeres B y G.

5. CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

Python fue el lenguaje de programación adecuado para realizar el proyecto de investigación, ya que este no solo es fácil e intuitivo de utilizar, sino que también cuenta con una gran variedad de librerías que soportan y facilitan la extracción y manipulación de datos, las cuales son optimizadas y actualizadas constantemente. De igual manera permite utilizar librerías que, en un principio, están en otro lenguaje de programación, como es el caso de Leaflet, la cual es utilizada principalmente en JavaScript para la creación de varios tipos de mapas.

Con respecto a los procesos de extracción y simplificación de la red extraída mediante OSMnx, se notó que los nodos principales, obtenidos gracias al recorte de nodos y conexiones, se encuentran en los extremos de la ciudad, esto se debe a que las vías principales que conectan el norte y el sur de Quito se encuentran en los bordes. Con esto se pudo realizar un análisis de tráfico de la ciudad, en el cual se determinó que la hora con más tráfico en el área metropolitana de Quito, durante la mañana, es las 7:00 y durante la tarde entre las 18:00 y las 19:00, esto se debe a que en estas franjas de tiempo los ciudadanos se dirigen a sus trabajos o están saliendo de estos, y para movilizarse entre sus domicilios y lugar de trabajo se utilizan estas vías.

Los clústeres obtenidos representan sectores de la ciudad que pueden servir como puntos de interés para otras investigaciones. Para determinar la cantidad final de clústeres se realizaron algunas pruebas, tomando la decisión final de agrupar los puntos en 7 clústeres, debido a que con una menor se agrupaban nodos lejanos en un mismo clúster y con una mayor cantidad se descartaban puntos cercanos de un clúster.

Finalmente, para que la información presentada en este trabajo sea comprensible y fácil de interpretar, se decidió mostrar los puntos y clústeres obtenidos dentro de un mapa de la ciudad de Quito y así estos se puedan observar de una manera más gráfica y se muestre más información como la ubicación y los flujos en una sola imagen. De igual manera se mostró información gráfica del flujo entre los clústeres en gráficos que muestren los datos necesarios.

5.2. Recomendaciones

Se sugiere guardar los datos obtenidos en cada proceso de preferencia en archivos de formato csv, esto porque las librerías con las que se extrae la información pueden actualizarse y añadir información, haciendo que los datos que se extraen en un principio sean diferentes, alimentando los demás procesos con distintos datos, provocando que, si se ejecuta el código de nuevo, los datos cambien. Al guardar los datos obtenidos, tanto de extracción como otros procesos, se puede mantener una información consistente, o se puede comparar la nueva información con la previa.

Al momento de trabajar con software geográfico, se recomienda utilizar siempre un sistema de coordenadas correctamente definido y transformar los datos obtenidos por distintos medios a este, para siempre mantener un estándar y tener un alto grado de consistencia y coherencia de datos durante todo el desarrollo del proyecto. Además de esto existen otros datos importantes como distancias y tiempo que deben ser siempre transformados a un sistema estandarizado, ya que, muchas herramientas suelen devolver los datos en diferentes medidas para diferenciarse de otros proveedores, dificultando esto su uso fuera de su propia plataforma de servicios.

Finalmente, se recomienda utilizar los resultados de este trabajo para la implementación de planes de ciudad sostenible que necesiten conocer ubicaciones importantes, como el caso de implementación de estaciones de carga para autos eléctricos o redirección del tráfico en determinados rangos de tiempo durante el día. Haciendo uso de este trabajo se tiene un punto de partida importante para la implementación de diferentes soluciones que promuevan un estilo de vida mejor dentro del área metropolitana de la ciudad de Quito.

REFERENCIAS

- Alfie Cohen, M., & Salinas Castillo, O. (2017). Ruido en la ciudad. Contaminación auditiva y ciudad caminable. *Estudios demográficos y urbanos*, 32(1), 65-96.
- El Telégrafo. (2017). *Los vehículos son los que más contaminan el aire*. Recuperado de: <https://www.eltelegrafo.com.ec/noticias/quito/1/los-vehiculosson-los-que-mas-contaminan-el-aire>.
- Anaconda (2020). Anaconda Navigator. *Anaconda Documentation*.
- Barrero, A. C., de García, G. W., & Parra, R. M. M. (2010). *Introducción a la Teoría de Grafos*. ELIZCOM SAS.
- Beaudoin, J., Farzin, Y. H., & Lawell, C. Y. C. L. (2015). Public Transit Investment and Sustainable Transportation.
- Bianchini, A. (1999). Conceptos y definiciones de hipertexto. *Caracas: Universidad Simón Bolívar*.
- Bibri, S. E., & Krogstie, J. (2017). Smart sustainable cities of the future: An extensive interdisciplinary literature review. *Sustainable cities and society*, 31, 183-212.
- Boeing, G. (2017). OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems*, 65, 126-139.
- Butler, H., Daly, M., Doyle, A., Gillies, S., Hagen, S., & Schaub, T. (2016). The geojson format. *Internet Engineering Task Force (IETF)*.
- Craymer, M. (2006). Making sense of evolving datums: WGS84 and NAD83. *Retrieved June 2, 2011*.
- Dlugosz, R., Talaska, T., Pedrycz, W., & Wojtyna, R. (2010). Realization of the conscience mechanism in CMOS implementation of winner-takes-all self-organizing neural networks. *IEEE Transactions on Neural Networks*, 21(6), 961-971.

- Everitt, B. S., Landau, S. and Leese, M. (2001), *Cluster Analysis, 4th Edition*, Oxford University Press, Inc., New York; Arnold, London.
- Everitt, B. S., Landau, S., Leese, M., & Stahl, D. (2011). *Cluster analysis*. John Wiley & Sons.
- Fernández, R. (2011). *Elementos de la teoría del tráfico vehicular*. Pontificia Universidad Católica del Perú.
- Ferraiolo, J., Jun, F., & Jackson, D. (2000). *Scalable vector graphics (SVG) 1.0 specification*. Bloomington: iuniverse.
- Folium (2013). Python-visualization. *Folium*.
- Galton, F. (1885). Some results of the Anthropometric Laboratory. *J. Anthropol. Inst.*, 16, 275-287.
- Google (2020). Directions API. *Google Developers*.
- Google (2020). Distance Matrix API. *Google Developers*.
- Higueras, E. (2009). El reto de la ciudad sostenible. *Madrid: Editorial DAPP*.
- Hoogendoorn, S. P., & Bovy, P. H. (2001). *State-of-the-art of vehicular traffic flow modelling. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 215(4), 283-303.
- Jupyter (2020). Jupyter Notebook. *Project Jupyter*.
- Kounadi, O. (2009). Assessing the quality of OpenStreetMap data. *Msc geographical information science, University College of London Department of Civil, Environmental And Geomatic Engineering*.
- Leaflet (2020). LeafletJS. *OpenStreetMap*.
- Luxen, D., & Vetter, C. (2011, November). Real-time routing with OpenStreetMap data. In *Proceedings of the 19th ACM SIGSPATIAL international conference on advances in geographic information systems* (pp. 513-516).
- Maps, G. (2015). Google maps. *Dipetik Desember, 14, 2015*.

- Mozilla (2020). JavaScript. *MDN web docs*.
- Oliphant, T. E. (2007). Python for scientific computing. *Computing in Science & Engineering*, 9(3), 10-20.
- Ortmann, R. (2005). *Técnica de los gases de escape para motores de gasolina*. Reverte.
- Pakhrou, T. (2013). Matemática Discreta. *Matemática Discreta*.
- Pascual, D., Pla, F., & Sánchez, S. (2007). Algoritmos de agrupamiento. *Método Informáticos Avanzados*, 164-174.
- Paz, I. K. (2007). Media aritmética simple. *Boletín electrónico*, 7, 1-13.
- Pérez, C. P. (2009). Gestión de la contaminación atmosférica urbana: El caso de Quito.
- Pérez, R. (2020). Distancia euclidiana: concepto, fórmula, cálculo, ejemplo. *Lifeder*.
- Perkel, J. M. (2018). Why Jupyter is data scientists' computational notebook of choice. *Nature*, 563(7732), 145-147.
- Sanchís Milla, F. J. (2016). *Definición e implantación de un proceso QA para desarrollo de software* (Doctoral dissertation).
- Sturn, A., Quackenbush, J., & Trajanoski, Z. (2002). Genesis: cluster analysis of microarray data. *Bioinformatics*, 18(1), 207-208.
- Tapia Granados, J. A. (1998). La reducción del tráfico de automóviles: una política urgente de promoción de la salud. *Revista Panamericana de Salud Pública*, 3, 137-151.
- Torrubia, G., & Terrazas, V. (2012). Algoritmo de Dijkstra. Un tutorial interactivo. *VII Jornadas de Enseñanza Universitaria de la Informática (JENUI 2001)*.

- Valenzuela-Levi, N. (2018). Why do more unequal countries spend more on private vehicles? Evidence and implications for the future of cities. *Sustainable Cities and Society*, 43, 384-394.
- Van Der Aalst, W. (2016). Data science in action. In *Process mining* (pp. 3-23). Springer, Berlin, Heidelberg.
- Van Rossum, G. (2007, June). Python Programming Language. In *USENIX annual technical conference* (Vol. 41, p. 36).
- Williams, B. M., & Hoel, L. A. (1999). *Modeling and forecasting vehicular traffic flow as a seasonal stochastic time series process* (No. LTVA/29242/CE99/103).

