

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Logística y Transporte

Optimización del ruteo de vehículos para la distribución de una empresa fabricante de bebidas en la ciudad de Quito.

PROYECTO INTEGRADOR

Previo la obtención del Título de:

Ingeniero en Logística y Transporte

Presentado por:

Jorge Luis Martinez Vega

Franklin Eduardo Vega Honores

GUAYAQUIL - ECUADOR

Año: 2020

DEDICATORIA

El presente proyecto lo dedico a Dios quien ha sido mi guía, fortaleza y su mano de fidelidad y amor han estado conmigo hasta el día de hoy. A mis padres quienes con su amor, paciencia y esfuerzo me han permitido llegar a cumplir hoy un sueño más, gracias por inculcar en mí el ejemplo de esfuerzo y valentía, de no temer las adversidades porque Dios está conmigo siempre.

Jorge Martinez Vega

A Dios, pues en su infinita divinidad nos permite avanzar, cumplir nuestros anhelos y continuar soñando. A mis amados padres Franklin y Lolita, quienes jamás perdieron de vista el objetivo pues siempre han estado para darme fuerzas y ánimo, incluso cuando yo mismo dudaba de mí. A mí esposa Katherine por su paciencia y mis adorados hijos pues sus sonrisas me motivan a convertirme en mi mejor versión.

Franklin Vega Honores

AGRADECIMIENTOS

Mi más sincero agradecimiento a todas las autoridades y personal que forman parte de la Escuela Superior Politécnica del Litoral, por confiar en mí, abrirme las puertas y prepararme para ser un buen profesional.

Jorge Martinez Vega

Mis más profundos agradecimientos a Carlos Ronquillo Franco quién ha sido un pilar fundamental en mi regreso a la vida universitaria; si me permite decirlo en Ud. encontré un amigo, aprendí a comprender que sus palabras siempre van orientadas al nuestro crecimiento académico y personal; además por supuesto a Nadia Cárdenas Escobar quien con su experiencia y dedicación nos orientó durante este proceso de graduación potenciando nuestras habilidades y fortaleciendo nuestras debilidades.

Alguien me dijo alguna vez que los grandes maestros de la historia, prefieren ser recordados por sus nombres y no por sus títulos. Gracias Carlos, gracias Nadia.

Franklin Vega Honores

DECLARACIÓN EXPRESA

"Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; Jorge Martínez y Franklin Vega damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"



Jorge Martínez Vega



Franklin Vega Honores

EVALUADORES

CARLOS
ALFREDO
RONQUILLO
FRANCO

Firmado digitalmente
por CARLOS ALFREDO
RONQUILLO FRANCO
Fecha: 2021.03.08
16:44:42 -05'00'

Mgtr. Carlos Ronquillo Franco

PROFESOR DE LA MATERIA

NADIA LORENA
CARDENAS
ESCOBAR

Firmado digitalmente
por NADIA LORENA
CARDENAS ESCOBAR
Fecha: 2021.03.10
14:26:17 -05'00'

Mgtr. Nadia Cárdenas Escobar

PROFESORA TUTORA

RESUMEN

El presente trabajo ejemplifica uno de los grandes desafíos que los operadores logísticos enfrentan en sus actividades diarias; la variante del VRP con restricciones de capacidad y ventanas de tiempo CVRPTW se ajusta a las necesidades y particularidades de la operación.

Debido al problema que implica resolver operaciones de gran complejidad de forma exacta, se han desarrollado metaheurísticas como los algoritmos k-medoids y genético que ofrezcan buenas soluciones factibles. Para resolver el CVRPTW se diseñó un plan de 4 fases: levantamiento y análisis de datos; agrupamiento de clientes utilizando el algoritmo k-medoids; diseño de las rutas dentro del clúster utilizando un algoritmo genético y finalmente la visualización de los hallazgos obtenidos mediante un programa principal en Python.

La implementación del proyecto impactará en el ahorro de rubros importantes como el consumo de combustible, costos de mantenimiento e insumos de los vehículos. Además, mejorará la organización de la operación mediante la estandarización de políticas de entrega de mercadería. La reducción del porcentaje de entregas no realizadas mejorará la percepción del nivel de servicio del cliente y podría influir en nuevas oportunidades de negocio.

Palabras clave—rutas, algoritmo genético, algoritmo k-medoids, ahorro, CVRPTW.

ABSTRACT

This paper presents one of the great challenges that logistics operators face in their daily activities; the variant of the VRP with capacity restrictions and time windows (CVRPTW) adjusts to the needs and particularities of delivery operation.

Due to the problem involved in solving highly complex operations, metaheuristics such as k-medoids and genetic algorithms have been developed that offer good feasible solutions. To solve the CVRPTW, a 4-phase plan was designed: data collection and analysis; clustering using the k-medoids algorithm; design of routes within the cluster using a genetic algorithm and finally the visualization of findings obtained through a main program in Python language.

The project implementation will impact the saving of important amounts such as fuel consumption, maintenance costs and vehicle supplies. In addition, it will improve the organization through merchandise delivery policies. Reducing the percentage of missed deliveries will improve the perception of the customer's level of service and could influence new business opportunities.

Keywords—routes, genetic algorithm, k-medoids algorithm, saving, CVRPTW

ÍNDICE GENERAL

EVALUADORES.....	5
RESUMEN	I
<i>ABSTRACT</i>	II
ÍNDICE GENERAL	III
ABREVIATURAS.....	VI
SIMBOLOGÍA.....	VII
ÍNDICE DE FIGURAS	VIII
ÍNDICE DE TABLAS.....	IX
CAPÍTULO 1	1
1. Introducción	1
1.1 Descripción del problema	1
1.2 Justificación del problema	2
1.3 Objetivos	2
1.3.1 Objetivo General	2
1.3.2 Objetivos Específicos.....	3
1.4 Marco teórico.....	3
1.4.1 Estado del arte.....	3
1.4.2 Definiciones, modelos y teoremas.....	5
CAPÍTULO 2	14
2. Metodología.....	14
2.1 Técnicas de investigación.....	14
2.1.1 Levantamiento de información	14
2.1.2 Análisis de la información levantada: Situación actual.	15
2.2 Recopilación de los datos.....	15

2.3	Descripción de la solución	15
2.4	Uso de software.....	17
2.4.1	Función dataSet	17
2.4.2	Función PAM	18
2.4.3	Función makeChromosome	18
2.4.4	Función startPopulation.....	18
2.4.5	Función distance	19
2.4.6	Función duration	19
2.4.7	Función bestsolution	19
2.4.8	Función fitness	19
2.4.9	Función selection	20
2.4.10	Función generation	20
2.4.11	Función crossing.....	21
2.4.12	Función mutation.....	22
2.4.13	Función geneticAlgorithm.....	23
2.4.14	Función mapping	24
2.5	Consideraciones legales y éticas.....	24
2.6	Fases del proyecto	25
2.7	Cronograma de trabajo.....	27
CAPÍTULO 3		28
3.	Resultados y Análisis.....	28
3.1	Programa para el ruteo de vehículos.	28
3.2	Programa para graficar las rutas.	31
3.3	Análisis de los principales indicadores logísticos.	35
3.4	Análisis Costo-Beneficio	38
CAPÍTULO 4		40

4.	Conclusiones y Recomendaciones	40
4.1	Conclusiones.....	40
4.2	Recomendaciones.....	40
	BIBLIOGRAFÍA	42
	ANEXOS	45

ABREVIATURAS

AG	Algoritmo Genético
AMT	Agencia Metropolitana de Tránsito.
CSV	Comma-separated values
CVRP	Capacited Vehicle Routing Problem.
ESPOL	Escuela Superior Politécnica del Litoral.
NP	No Polynomial.
P	Polynomial.
PNG	Portable Network Graphics
PAM	Partición Alrededor de Medoids
TSP	Travelling Salesman Problem.
VRP	Vehicle Routing Problem.

SIMBOLOGÍA

Km	Kilómetros
min	Minutos
seg	Segundos
\$	Dólares

ÍNDICE DE FIGURAS

Figura 2.1 Seudocódigo de la función makeChromosome	18
Figura 2.2 Seudocódigo de la función startPopulation	19
Figura 2.3 Seudocódigo de la función selection.....	20
Figura 2.4 Seudocódigo de la función generation	21
Figura 2.5 Seudocódigo de la función crossing	22
Figura 2.6 Seudocódigo de la función mutation	23
Figura 2.7 Seudocódigo de la función geneticAlgorithm	23
Figura 2.8 Perímetro de Restricción “Hoy no Circula”. [Imagen tomada de la web de la AMT].....	25
Figura 3.1 Módulos del programa	28
Figura 3.2 Formato de los datos en el archivo de Excel	29
Figura 3.3 Puntos agrupados por el algoritmo k-medoides	30
Figura 3.4 Resultados por consola	31
Figura 3.5 Ejemplo de mapa para la ruta 1	32
Figura 3.6 Ejemplo de mapa para la ruta 2.....	33
Figura 3.7 Ejemplo de mapa para la ruta 3.....	33
Figura 3.8 Porcentaje de demanda no atendida del proyecto	36
Figura 3.9 Porcentaje de demanda no atendida real	36
Figura 3.10 Total de kilómetros recorridos.....	37
Figura 3.11 Total de gasto en combustible	38
Figura 3.12 Gasto en neumáticos	39

ÍNDICE DE TABLAS

Tabla 2.1 Principales indicadores de la operación.....	17
Tabla 2.2 Cronograma de trabajo.....	27
Tabla 3.1 Resultados del día 1.....	34
Tabla 3.2 Resultados del día 2.....	34
Tabla 3.3 Resultados del día 3.....	34
Tabla 3.4 Resultados del día 4.....	34
Tabla 3.5 Resultados del día 5.....	34
Tabla 3.6 Porcentajes de demanda no atendida.....	35
Tabla 3.7 Kilómetros recorridos y gasto en combustible.....	37
Tabla 3.8 Porcentaje de reducción de kilómetros recorridos.....	37
Tabla 3.9 Porcentaje de reducción de gasto en combustible.....	37
Tabla 3.10 Ahorro anual aproximado en neumáticos.....	38
Tabla 3.11 Beneficios de la implementación.....	39
Tabla 3.12 Costos de la implementación.....	39
Tabla 3.13 Factor entre el beneficio y el costo de la implementación.....	39

CAPÍTULO 1

1. INTRODUCCIÓN

El operador logístico con el cual se trabaja cuenta con operaciones en Perú, Bolivia, Ecuador, Colombia, El salvador, Guatemala y Honduras. Es una empresa que forma parte de un importante grupo especializado en ofrecer un servicio que integra los procesos de almacenaje, logística refrigerada, gestión documental, transporte y distribución.

La empresa necesita reducir su porcentaje de entregas diarias no realizadas mediante un diseño de rutas óptimo en las operaciones para su cliente, una empresa fabricante de bebidas localizada en la ciudad de Quito. Actualmente el proceso de ruteo es realizado por el cliente y existe cierta descoordinación lo cual lleva muchas veces a consolidar la carga de clientes que no se encuentran correctamente distribuidos en la ruta y ya que no cuenta con un plan de ruteo vehicular que utilice métodos y tecnologías modernas para optimizar la ruta de los vehículos, la empresa no logra cumplir con muchas de las entregas diarias programadas. Las entregas se realizan con una flota homogénea de camiones y en un horario laboral de 8 horas diarias. Se busca optimizar los costos de transporte para los despachos de sus clientes, así como la utilización de los camiones con la carga y ruta óptimas para que el vehículo llegue a cada punto.

Según las características de la operación esto se alinea a un problema de ruteo vehicular capacitado con flota homogénea y restricciones de tiempo. Para realizar el programa de rutas que reduzca el porcentaje de entregas diarias no realizadas se planea fraccionar el conjunto de clientes utilizando un algoritmo de agrupamiento y posteriormente diseñar una ruta óptima para cada grupo de clientes mediante el uso de una metaheurística, para el presente proyecto se utilizara un algoritmo genético.

1.1 Descripción del problema

Debido a una deficiente programación de rutas por falta de métodos y tecnologías modernas, el operador logístico responsable de la distribución de bebidas para un cliente fabricante en la ciudad de Quito mantiene un bajo nivel de cumplimiento en sus entregas diarias para los puntos de venta distribuidos en la ciudad de Quito.

Actualmente las rutas son asignadas y proporcionadas por el cliente; el operador logístico encargado de la distribución se encuentra con algunos problemas operativos adicionales tales como que ciertos pedidos no se encuentran correctamente cargados en el camión asignado en la ruta correspondiente lo cual se detecta durante la revisión de la mañana siguiente, previa salida de la ruta y el tiempo que se tarda en corregir esos errores provoca que el recorrido empiece más tarde de lo programado pudiendo impactar directamente en el tiempo disponible para completar las entregas planificadas.

1.2 Justificación del problema

El presente proyecto ayuda a un importante operador logístico a mejorar sus operaciones de distribución, lo que se verá reflejado en la disminución de puntos con desabastecimiento por consiguiente impactará positivamente en la satisfacción por parte del cliente y en el ahorro en los costos inherentes a la operación.

Al lograr optimizar las distancias recorridas por los camiones se obtendrá un ahorro en consumo de combustible, neumáticos; así como también en mantenimiento y repuestos de la flota, de esta manera el capital ahorrado puede ser invertido en otras áreas. Al brindar un mejor servicio y optimizar los tiempos de operación se podrá tener la capacidad de atender un mayor número de clientes. Todo esto ocasionará un crecimiento económico para la empresa. Al tratarse de un área con alto impacto al medio ambiente como es el transporte, se planea reducir las emisiones de CO₂ al disminuir el tiempo de operación de los camiones haciendo así a la empresa menos contaminante y más sostenible. Al mismo tiempo que se reducen los niveles estrés en los choferes que sienten la presión de cumplir con sus entregas en una ciudad con alto tráfico, se reduciría su tiempo frente al volante en algunos casos al mejorar las rutas ya sea por el acortamiento de estas o al utilizar rutas menos congestionadas.

1.3 Objetivos

1.3.1 Objetivo General

Optimizar el diseño de rutas para los vehículos encargados de la distribución de los productos de una empresa fabricante de bebidas en la ciudad de Quito.

1.3.2 Objetivos Específicos

1. Implementar un método de dos fases para obtener las rutas que reduzcan tiempo y distancia recorrida por los camiones, que realizan la distribución, en un tiempo computacional prudente.
2. Diseñar un plan de ruteo vehicular para las operaciones en Quito programando una función que plasme la solución obtenida en un mapa para tener una mejor visión de las rutas a tomar.
3. Comparar los principales indicadores de la operación, entre la situación actual utilizando los datos proporcionados por el encargado de área y la situación que se espera lograr con el nuevo plan de rutas, estimando los valores con la solución que nos entrega el método de dos fases para medir la eficacia del proyecto.
4. Analizar el costo y beneficio estimados de la implementación del proyecto, estimando cuánto dinero representa el tiempo que los empleados se tomarían en aprender a utilizar el programa, el dinero que se ahorra la empresa en repuestos y combustible.

1.4 Marco teórico

1.4.1 Estado del arte

El problema del operador logístico se encuentra encasillado en el grupo de problemas conocidos con VRP, para ser más precisos en la variante CVRPTW debido a las restricciones en la capacidad de los vehículos y los tiempos de entrega permitidos.

El problema de ruteo vehicular fue planteado hace ya más de medio siglo, Dantzig y Ramser en el año 1959 abordaron por primera vez el problema de ruteo vehicular (Dantzig y Ramser, 1959). Según Laporte, el problema del ruteo vehicular es determinar la mejor ruta de distribución o recolección para los vehículos asignados a cubrir dicha ruta, desde uno o varios depósitos hacia puntos dispersos geográficamente (Laporte et al., 1988). Cuando existe una limitación en la capacidad de los vehículos al problema se lo conoce como CVRP. Comert, Yazgan, Kir y Yener manifiestan que el CVRP en la red de distribución es uno de los más importantes problemas y también el tipo más común de VRP que encontramos. Si al mismo tiempo al CVRP se le aumentan restricciones

para los tiempos de entrega, a este nuevo problema de ruteo vehicular se lo conoce como CVRP con ventanas de tiempo (CVRPTW) (Comert et al., 2018).

Para el presente caso de estudio, el operador logístico presenta ciertas características de interés tales como que los vehículos son de idénticas características (capacidad) y la distribución se realiza desde un único depósito central. Además, se consideran restricciones de capacidad para los vehículos y restricciones de tiempo para las entregas de los productos, ya las mismas solo pueden realizarse dentro del horario laboral. “El CVRP tiene como objetivo minimizar el costo total necesario para atender a todos los clientes” (Sbai et al., 2020). Los algoritmos que se utilizan para resolver los diferentes tipos de VRP se dividen en dos principales categorías, las cuales son: los algoritmos exactos y los algoritmos aproximados que se conocen como heurísticas (Sbai et al., 2020). Los métodos exactos que se conocen como los más exitosos son el algoritmo Branch-and-cut (Augerat et al., 1998; Letchford et al., 2002 and Lysgaard et al., 2004), Branch and Bound (Christofides and Eilon, 1969) y el Branch-Cut-and-Price combinado con el Branch-and-Cut y Relajación Lagrangeana/Generación de Columnas (Fukasawa et al., 2006; Baldacci et al., 2008; Pecin et al., 2014, 2017). Pero como en el trabajo de Ortega y compañía, Optimización del enrutamiento de vehículos en la distribución productos de una empresa en la ciudad de Sincelejo – Colombia, el cual presenta características muy parecidas al problema del operador logístico en mención de nuestro proyecto con lo que podemos notar que los métodos exactos no son muy efectivos debido al tiempo de ejecución y costo computacional, por su trabajo podemos concluir que “...se evidencia la eficiencia de los métodos exactos en la optimización de modelos de programación entera mixta, lo que motiva la necesidad de diseñar algoritmos heurísticos, metaheurísticos y matemáticos que logren reducir este tipo de factores”(Ortega et al., 2018).

Según Uchoa, El tiempo para que los algoritmos exactos nos otorguen una solución óptima es demasiado alto ya que es considerado un problema NP-Hard, por esta razón al momento de resolver instancias con más de 20 clientes, estos métodos se vuelven inconsistentes “(Uchoa et al. 2017). Para resolver los problemas en cuyas instancias de clientes sean mayores se han propuesto diversos enfoques heurísticos, metaheurísticos e híbridos para obtener buenas soluciones (Adewumi y Adeleke, 2018). Como ejemplo

tenemos a Ines Sbai, Saoussen Krichen y Olfa Limam, quienes usaron un algoritmo genético en su trabajo del 2020 (Sbai et al., 2020).

Con el objetivo de reducir los tiempos de ejecución para la obtención de rutas óptimas se utiliza un método jerárquico en el cual se agrupará los clientes primero y después se procederá a hacer el ruteo de cada grupo. Los grupos tienen como criterio de agrupación que los datos sean de características similares (Comert et al., 2018). Cuando los conjuntos de datos son demasiado grandes, las técnicas de agrupación clásicas no sirven. Por esta razón, la escalabilidad ha sido punto clave para el desarrollo de algunos algoritmos (Venkates et al., 1999). “El método de agrupación varía según el tipo de datos y el propósito del estudio” (Comert et al., 2018). Los resultados del caso de estudio de Comert, Yazgan, Kir y Yener nos indican que el mejor algoritmo de agrupamiento es el K-medoids.

Para conseguir un plan de ruteo vehicular óptimo en un tiempo de ejecución aceptable es necesario dividir el problema en dos fases, la primera es agrupar mediante el algoritmo K-medoids para de esta manera en la segunda fase rutear cada uno de los grupos que se forman en la primera fase mediante un algoritmo genético. De esta forma se espera reducir el porcentaje de pedidos diarios no atendidos.

1.4.2 Definiciones, modelos y teoremas

1.4.2.1 Modelo matemático

Los modelos matemáticos tienen como objetivo explicar de manera teórica un fenómeno que existe fuera del campo de la Matemática, como por ejemplo los pronósticos del clima o los pronósticos económico. La eficacia de un modelo matemático está estrechamente ligada a la precisión con la que se construya la representación numérica.

Los modelos matemáticos son muy útiles cuando se quiere estudiar el comportamiento de las estructuras complejas frente situaciones que no son fáciles de ver en el ámbito real. Se pueden clasificar en dos tipos de acuerdo con la procedencia de la información en que se basa el modelo. El primero es el modelo heurístico, este se basa en las definiciones de las causas naturales que originan el fenómeno en cuestión, y modelo empírico, que se basa en el estudio de los resultados de la experimentación.

Los modelos matemáticos también se pueden clasificar con respecto al tipo de resultado que se pretende obtener. Unos son los modelos cualitativos que se apoyan en gráficos

y no pretenden un resultado de tipo exacto, sino que buscan detectar incrementos o disminuciones de determinado valor. Los otros son los modelos cuantitativos, que basan en fórmulas matemáticas de variada complejidad para encontrar un número preciso. Otra forma de clasificar los modelos matemáticos es por la aleatoriedad de la situación inicial. Según esta clasificación tenemos a los modelos deterministas en los cuales no existe incertidumbre ya que los datos y los resultados se conocen y los modelos estocásticos que nos entregan la probabilidad de que cierto resultado sea obtenido. Por último, tenemos la clasificación de acuerdo con el objetivo del modelo, esta divide a los modelos en tres grupos: modelos de simulación, modelos de optimización y modelos de control.

1.4.2.2 Optimización combinatoria

La optimización combinatoria es una rama de la optimización en matemáticas aplicadas y en ciencias de la computación, que está fuertemente relacionada con la investigación operativa, la teoría algorítmica y la teoría de la complejidad computacional, entre otros campos. En los problemas de optimización combinatoria el espacio de soluciones posibles es discreto, debido a esto, el óptimo se podría obtener mediante la exploración del espacio de soluciones, pero esta posibilidad solo es efectiva en problemas cuyo tamaño sea muy pequeño. Los algoritmos de optimización combinatoria nos ayudan a encontrar una buena solución logrando reducir el tamaño efectivo del espacio de búsqueda, y explorando este espacio de soluciones eficientemente.

1.4.2.3 Complejidad computacional

La teoría de la complejidad computacional es una rama de la teoría de computación, la cual se encarga del estudio de los recursos computacionales necesarios para resolver un problema dado. Estos recursos usualmente se miden en términos de parámetros abstractos como son el tiempo y el espacio. El tiempo hace referencia al número de pasos requeridos para resolver el problema y el espacio hace referencia a la cantidad de memoria que se utiliza para la resolución del problema. También se puede considerar como recurso a el número de procesadores en paralelo que se requieren para resolver un problema en paralelo. Los problemas según el tiempo de complejidad se clasifican en dos grupos: polinomiales y no polinomiales.

La clase de complejidad P es el conjunto de problemas de decisión que pueden ser resueltos por una máquina determinista en tiempo polinomial. Esta clase corresponden a la idea intuitiva de los problemas que se puede resolver de modo efectivo en los peores casos.

La clase de complejidad NP es el conjunto de problemas de decisión que pueden resolverse por una máquina no determinista en tiempo polinomial. La principal propiedad de los problemas de esta clase es que sus soluciones pueden ser revisadas de modo eficiente en tiempo polinomial.

1.4.2.4 Red o grafo

Una red se compone de un conjunto de nodos unidos por arcos (o ramas). La notación para describir una red es (N, A) , donde N es el conjunto de nodos, y A es el conjunto de arcos. Se dice que un arco está dirigido u orientado si permite el flujo positivo sólo en una dirección. Una red dirigida tiene todos los arcos dirigidos. Una ruta es un conjunto de arcos que unen dos nodos distintos, y que pasan a través de otros nodos en la red. Una ruta forma un ciclo o un bucle si conecta un nodo de vuelta a sí mismo a través de otros nodos. Se dice que una red está conectada si cada dos nodos distintos están conectados en al menos una ruta. Un árbol es una red conectada libre de ciclos compuesta de un subconjunto de todos los nodos, y un árbol de expansión es un árbol que une todos los nodos de la red.

1.4.2.5 Problema del Agente Viajero

Es un problema de optimización y combinatoria que pertenece a la clase de los problemas difíciles (NP-Hard). El objetivo que se busca al resolver el TSP es encontrar el recorrido con la menor distancia o el costo mínimo en una situación donde existen n ciudades, todas las ciudades deben ser visitadas y cada ciudad solo puede ser visitada una vez antes de regresar al punto de partida. A esto en teoría de grafos se lo conoce como tour o ciclo hamiltoniano.

Sea un grafo $G = (V, A, C)$ donde V es el conjunto de vértices, A es el de aristas y $C = (c_{ij})$ es la matriz de costos. Esto es, c_{ij} es el costo o distancia de la arista (i, j) .

Defina:

$$x_{ij} = \begin{cases} 1, & \text{si se llega a la ciudad } j \text{ desde la ciudad } i \\ 0, & \text{de lo contrario} \end{cases}$$

El modelo TSP se da como:

$$\text{Minimizar } z = \sum_{i=1}^n \sum_{j=1}^n d_{ij} x_{ij}, d_{ij} = \infty \text{ para todas las } i = j \quad (2.1)$$

Sujeto a:

$$\sum_{j=1}^n x_{ij} = 1, i = 1, 2, \dots, n \quad (2.2)$$

$$\sum_{i=1}^n x_{ij} = 1, j = 1, 2, \dots, n \quad (2.3)$$

$$x = (0, 1) \quad (2.4)$$

$$\sum_{(i,j) \in \partial(S)} x_{ij} \geq 2 \quad \forall S \subseteq \{1, 2, \dots, n\}, 3 \leq |S| \leq [n/2] \quad (2.5)$$

Donde $\partial(S)$ representa el conjunto de aristas incidentes con exactamente un vértice de S .

La ecuación (2.1) es la función objetivo a minimizar.

Las restricciones (2.2), (2.3) y (2.4) definen un modelo de asignación regular donde x_{ij} es igual a 1 si el nodo (ciudad) i está conectado a un nodo (ciudad) j , y cero en caso contrario, esto se ve reflejado en la variable binaria x_{ij} . Las restricciones que aparecen en (2.6) reciben el nombre de restricciones de eliminación de subtours y garantizan que la solución no contenga subciclos. El problema es que al haber una por cada subconjunto del conjunto de vértices, aparecen en una cantidad del orden de 2^n , lo cual hace inmanejable tal formulación.

Al asignar una penalización muy alta a los elementos diagonales de la matriz de distancias prohíbe que una ciudad esté conectada a sí misma. Si $d_{ij} = d_{ji}$ para todas las i y j en un modelo TSP, entonces es simétrico. Caso contrario, es asimétrico.

1.4.2.6 Heurística

La heurística es una técnica de búsqueda directa que mediante el uso de técnicas favorables y prácticas logra determinar buenas soluciones con rapidez, aunque suele desconocerse la calidad de esta solución con respecto a la óptima. Estos procedimientos suelen diseñarse de modo específico para cada problema. En Inteligencia Artificial se suele denominar función heurística a aquella que dirige la búsqueda (o construcción) de una solución, utilizando algún mecanismo con una sencillez prudente. Para que un algoritmo heurístico sea considerado bueno, eficiente y robusto su resultado debe estar

muy cerca del óptimo y la probabilidad de obtener una mala solución debe ser muy pequeña, además esta solución debe requerir un esfuerzo computacional realista.

Entre los principales tipos de heurísticas tenemos las heurísticas constructivas, las heurísticas de búsqueda local, las heurísticas voraces y las heurísticas con BackTracking(Taha, 2012).

Las heurísticas constructivas son aquellas en las que en cada paso se va añadiendo un elemento más a una solución que aún no ha sido completamente construida, este elemento es añadido de acuerdo con ciertos criterios de decisión, estos son diferentes para cada tipo de problema.

Las heurísticas de búsqueda de entornos o de mejora local son aquellas que parten de una solución cualquiera, esta solución puede ser aleatoria o no, y van buscando en el vecindario más próximo una solución que sea mejor a la que ya se dispone, produciendo mejoras hasta que ningún elemento en el vecindario es mejor al que ya se tiene.

Las heurísticas de tipo voraz son aquellas que siempre las soluciones mejores de modo local para generar el resultado. Algunas heurísticas voraces pueden ser óptimas, pero no siempre es así. Los algoritmos de tipo voraz suelen ser muy rápidos ya que no consideran alternativas.

Por último, tenemos a las Heurísticas con BackTracking. Estas son heurísticas que, si alcanzan un punto en el cual no pueden seguir construyendo una buena solución, estas retroceden en el proceso de reconstrucción, para analizar caminos abandonados que pueden contener mejores soluciones.

1.4.2.7 Metaheurísticas

Es una nueva generación de métodos heurísticos que mejora la calidad de las soluciones heurísticas ya permite el escape de óptimos locales, según Taha en su libro "Investigación de Operaciones, Novena Edición" esto se logra al permitir la búsqueda de una trampa de escape en óptimos locales. Esta ventaja se logra a costa de incrementar los cálculos.

Los algoritmos metaheurísticos están especializados en resolver problemas de optimización con un marco referencial de nivel superior que se combina con otras heurísticas, esta combinación es una estrategia para la búsqueda eficiente de mejores soluciones factibles.

Las metaheurísticas deben ser generales y sencillas. Cuando se centran en un problema en específico van perdiendo esa generalidad y exigen más conocimiento del problema, Además de los parámetros que definen a cada algoritmo, las metaheurísticas necesitan algún tipo de representación que es propia de cada problema.

Entre las metaheurísticas más conocidas se pueden destacar las siguientes:

- Algoritmos de Hormigas
- Algoritmos Genéticos
- Algoritmos Miméticos
- Redes Neuronales
- Scatter Search
- Recocido Simulado
- Búsqueda Tabú
- Target Analysis
- GRASP.

1.4.2.8 Algoritmo genético

El algoritmo genético establece una analogía entre el conjunto individuos en una población y el de soluciones en un problema, imitando el proceso de evolución biológica conocido como “la supervivencia del más apto”. Esta analogía se logra codificando cada solución factible del problema en un vector numérico o binario que representa el conjunto de genes presente en el cromosoma. El conjunto de N soluciones factibles hace referencia a una población con N cromosomas. El fundamento del AG es seleccionar dos individuos dentro de una población que representaran a los padres, estos deben ser los más aptos. Los individuos más aptos se seleccionan a través de una función objetivo apropiada para cada tipo de problema, la función objetivo entregara un mejor valor a los cromosomas más aptos. Los genes de los dos padres se cruzan para producir dos hijos. Además, como sucede en la naturaleza, existe la posibilidad de que al momento del cruce estos genes muten. La descendencia reemplazará a los dos cromosomas cuyo valor de la función objetivo sea menor. Así el proceso de seleccionar nuevos padres se repite durante varias generaciones, mejorando cada vez la población con individuos más aptos. Existen varias reglas para seleccionar padres, crear hijos, mutar genes y substituir cromosomas dentro de una población.

A continuación, se muestra la implementación de un algoritmo genético propuesta por Michalewicz (1996), en donde se combinan los elementos genéticos con la búsqueda local.

1. Generar soluciones — Construir un conjunto de soluciones P con tamaño $PopSize$ mediante generación aleatoria.

2. Mejorar soluciones — Aplicar un método de búsqueda local a cada solución del conjunto P .

Mientras (número de evaluaciones $<$ $MaxEval$)

3. Evaluación — Evaluar las soluciones en P y actualizar, si es necesario, la mejor solución almacenada.

4. Supervivencia — Calcular la probabilidad de supervivencia basada en la calidad de las soluciones. Según dichas probabilidades seleccionar aleatoriamente $PopSize$ soluciones (con reemplazamiento) de P . Sea P el nuevo conjunto formado por las soluciones seleccionadas (algunas pueden aparecer repetidas).

5. Combinación — Seleccionar una fracción pc de soluciones de P para ser combinadas. La selección es aleatoria y equi-probable para todos los elementos de P . Los elementos seleccionados se emparejan al azar y, por cada pareja, se generan dos descendientes que reemplazarán a los padres en P .

6. Mutación — Una fracción pm de las soluciones de P se selecciona para aplicar el operador de mutación. La solución resultante reemplaza a la original en P .

Los valores que se usan habitualmente para los parámetros en los algoritmos genéticos son los siguientes:

- Población: entre 50 y 100 cromosomas.
- Longitud del cromosoma: depende del problema.
- Generaciones: entre 20 y 1000.
- Probabilidad (o porcentaje) de cruce: de 20% a 60% por cromosoma.
- Probabilidad (o porcentaje) de mutación: de 0.1% a 5% por gen.

1.4.2.9 Algoritmo K-medoids capacitado

K-medoids es un algoritmo de agrupamiento que divide el conjunto de puntos en grupos, minimizando la distancia entre los puntos que se añadirían a un grupo y otro punto designado como el centro de ese grupo. La diferencia entre los algoritmos K-means y K-medoids es que el centro de los conjuntos se establece como el objeto más cercano al centro de los conjuntos, no los valores medios de los objetos del conjunto. Por esta razón el algoritmo es más sensible a los datos ruidosos que el método K-means.

Pasos del algoritmo K-medoids

1. Determinar el número k conjuntos.
2. Elegir k objetos como medoides iniciales.
3. Asignar el resto de los conjuntos a los medoides x .
4. Calcular la función objetiva.
5. Determinar la coincidencia del punto y .
6. Si cambiar x e y minimizará la función objetiva, cambie x e y .
7. Si no hay cambios, siga repitiendo el Paso 3–6.

1.4.2.10 Notaciones y Modelo Propuesto

Las notaciones utilizadas para el problema CVRPTW y el modelo matemático se detallan a continuación:

Conjuntos

- | | |
|-----------|---|
| N_d | Nodo que representa la empresa (depósito/almacén) |
| N_c | Conjunto de los clientes. |
| N | Conjunto de los clientes más la empresa |
| κ | Representa el vehículo |
| Q_k | Capacidad del vehículo |
| Cv_k | Costo variable por uso del vehículo. |
| C_{ij} | Distancia en kilómetros (Km). |
| D_p | Demanda de los clientes. |
| T_{ijk} | Tiempo de viaje del vehículo k . |

T_{sjk} Tiempo servicio del vehículo a cada uno del cliente representado en minutos

T_k Tiempo máximo de ruta del vehículo representado en minutos.

Variables de Decisión

X_{ijk} Variable binaria que representa la decisión de; 1 si el vehículo viaja de un cliente a otro y 0 en caso contrario.

Función objetivo

$$\sum_{k \in K} \sum_{i \in A} \sum_{j \in A} x_{ijk} (Cv_k C_{ij}) \quad (2.6)$$

Restricciones

$$\sum_{k \in K} \sum_{j \in N_c} X_{ijk} = 1; \forall i \in N_c \quad (2.7)$$

$$\sum_{j \in N_c} X_{iN_dk} = \sum_{j \in N_c} X_{N_djk}; \forall k \in K, N_d \in N \quad (2.8)$$

$$\sum_{k \in K} \sum_{j \in N_c} X_{ijk} D_j \leq Q_k; \forall k \in K \quad (2.9)$$

$$\sum_{i \in N_c} \sum_{j \in N_c} X_{ijk} \leq D_j; \forall k \in K \quad (2.10)$$

$$\sum_{i \in N_c} T_{sijk} \sum_{j \in N_c} X_{ijk} + \sum_{i \in N_c} \sum_{j \in N_c} X_{ijk} T_{ijk} \leq T_k; \forall k \in K \quad (2.11)$$

$$\sum_{i \in N_c} \sum_{j \in N_c} X_{ijk} \leq D_j; \forall k \in K \quad (2.12)$$

$$X_{ij} \in \{0,1\}; \forall i, j \in A; k \in K \quad (2.13)$$

La ecuación (2.6) representa el costo total de la ruta del CVRPTW. La ecuación (2.7) garantiza que cada uno de los clientes sean visitados solo una vez por el vehículo asignado. Por su parte, (2.8) asegura que el vehículo inicie en la bodega de la empresa y finalice en el mismo lugar. De la misma manera, (2.9) garantiza que no se exceda la capacidad del vehículo por cada ruta. Así mismo, (2.10) establece que la cantidad de productos que se cargue al vehículo no exceda la demanda o requerimientos por parte de los clientes. Igualmente (2.11) permite establecer que la suma del tiempo requerido para el servicio de los clientes y el tiempo requerido para viajar entre ellos no exceda el tiempo máximo del cual se puede emplear el vehículo y (2.12) funciona como restricción que elimina los sub – tours. Por último, (2.13) define el dominio de las variables.

CAPÍTULO 2

2. METODOLOGÍA

El presente capítulo del proyecto tiene como fin detallar las directrices a seguir para el cumplimiento de los objetivos trazados.

Primero se detalla cómo se obtuvieron los datos de interés para el desarrollo del proyecto y cuales son dichos datos de interés. Esto fue primordial para conocer la situación en la cual se encuentra la empresa y saber con qué recursos se contó para realizar el ruteo. En segundo lugar, se procede con la descripción del método que se utilizó para realizar el ruteo vehicular, así como también se detallan cada uno de los algoritmos utilizados y las funciones contenidas en dichos algoritmos. Posteriormente se muestra cómo se desarrolló el reporte final que contiene el mapa de rutas.

Para finalizar, se indica cómo se realizó la estimación de los principales indicadores logísticos de la operación, así como también la estimación de los costos asociados a la operación.

Adicionalmente se muestran las consideraciones legales que se tomó y las fases del proyecto.

2.1 Técnicas de investigación

A continuación, se detalla cómo se obtuvieron los datos y el estado en que se encuentra la compañía con respecto a la operación en la que se centra el desarrollo del proyecto.

2.1.1 Levantamiento de información

Se realizaron entrevistas con el encargado del área en la ciudad de Quito quien nos explicó todos los detalles sobre la operación. Para dichas entrevistas se utilizó el programa de videollamadas y reuniones virtuales "ZOOM". Además, nos proporcionó algunos informes con los pedidos diarios, la localización de los clientes, las capacidades de los vehículos, así como algunas particularidades de la operación de carga y distribución.

2.1.2 Análisis de la información levantada: Situación actual.

Al tratarse de un operador logístico, debemos identificar que es una empresa que se encarga de realizar almacenamiento, transporte y/o distribución de otras empresas según sea el caso. El papel del operador logístico radica en garantizar los camiones y ponerlos a disposición de su cliente, pero el plan de rutas para esta operación es realizado por el cliente. Este plan de rutas es desarrollado de forma empírica sin el uso de métodos y tecnologías modernas motivo por el cual al finalizar la jornada laboral quedan varios pedidos sin atenderse. La empresa tiene asignada para esta operación una flota homogénea de tres vehículos con capacidad de 200 unidades de carga en cada vehículo y necesita satisfacer una demanda promedio de 100 pedidos diarios. Las entregas se realizan en horario laboral y los pedidos que no llegan a concretarse son aplazados al día siguiente.

2.2 Recopilación de los datos

Se utilizaron 5 días de operación el presente proyecto, los datos fueron recopilados en los días comprendidos desde el 12 de enero del 2021 hasta el 16 de enero 2021; en cada fila de la tabla se detalla la demanda y las coordenadas del cliente que ha hecho un pedido en dicho día, las coordenadas corresponden a la latitud y longitud del cliente. El centro de distribución se encuentra ubicado en las coordenadas:

-0.3522888411859445, -78.53859531606719.

En el apartado Anexos se encontrarán los pedidos del día 12 de enero del 2021 al 16 de enero 2021.

2.3 Descripción de la solución

Según las características de la operación, el problema se alinea con la variante del VRP en la que se incluyen restricciones de tiempo y capacidad. Esta variante es conocida como CVRPTW y según la literatura revisada en el capítulo anterior se llega a la conclusión de que resolver este tipo de problema con métodos exactos para instancias de más de 20 clientes conlleva a un costo computacional muy elevado al tratarse de un

problema NP-HARD. Por este motivo, se optó por la implementación de un algoritmo que consistió en un enfoque de fases.

La primera fase redujo la complejidad del problema al partitionarlo utilizando un algoritmo de agrupamiento conocido como k-medoids. Para realizar esta primera parte se utilizó el PAM que es la aplicación práctica más común, aunque puede no entregar la solución óptima al realizar una búsqueda golosa, el PAM es más rápido que la búsqueda exhaustiva y nos entrega una buena solución. El valor de k utilizado en el proyecto fue el número de vehículos disponibles (3 vehículos) de esta manera los pedidos quedaron separados en 3 grupos y cada uno de los vehículos se encargó de la distribución de un grupo de pedidos.

La segunda fase consistió en encontrar una buena ruta para cumplir la demanda de cada grupo de pedidos. Para esto se programó un algoritmo genético que se implementó a cada grupo de pedidos. El programa principal finalizó entregando un mapa con la ruta a seguir de cada vehículo y una lista con los pedidos que no se alcanzaron a realizar por cuestiones de tiempo de darse el caso.

Para medir la eficacia del programa se realizaron dos estimaciones, la primera fue la de los costos involucrados en la operación y la segunda fueron de dos principales indicadores logísticos para la operación. Estas estimaciones se compararon con el costo y los indicadores para la operación con planificación empírica que se venía usando. A continuación, se detalla la tabla con los indicadores. (Véase Tabla 2.1)

Tabla 2.1 Principales indicadores de la operación.

INDICADOR	FÓRMULA
Porcentaje de pedidos no realizados	$\frac{\text{Numero de pedidos no realizados}}{\text{Total de pedidos}}$
Kilómetros totales recorridos	$\sum_{i=1}^K \text{Kilometros recorridos por el vehiculo } i$
Gasto total de combustible	$\text{Total de Km recorridos} * \frac{\text{Galones de diesel}}{\text{Kilometro}} * \text{precio del diesel}$
Gasto de neumáticos	$\sum_{i=1}^3 \left(\frac{\text{Costo neumático [\$]}}{\text{Rendimiento esperado [Km]}} * 6 * \text{vehículo } i \right) * \text{Total kilómetros recorridos [Km]}$

2.4 Uso de software

El proyecto se realizó con el lenguaje de programación Python, se utilizó la versión 8.3 de Python. Las librerías utilizadas fueron “random” que nos ayuda a generar números pseudoaleatorios, “math” que contiene varias funciones matemáticas y “NumPy” que nos permite crear vectores y matrices grandes multidimensionales, junto con una gran colección de funciones matemáticas de alto nivel para operar con ellas. Para obtener las distancias y los tiempos de recorrido se plantearon dos alternativas, en la primera se utilizó la fórmula del semiverso para encontrar la distancia entre dos puntos sobre una esfera, y la segunda fue utilizar la API de Google “Distance Matrix API”, para el análisis de resultados se utilizó solo la primera opción por pedido del cliente ya que la segunda opción genera un costo extra de 5 dólares cada 1000 consultas. También se utilizó Microsoft Excel para el ingreso de los datos.

2.4.1 Función dataSet

Esta función recibe un archivo de Microsoft Excel en el cual están contenida la información de los pedidos diarios y la procesa para entregar cuatro arreglos de NumPy (librería de Python). El primer arreglo contiene las coordenadas de los pedidos, el segundo arreglo contiene las demandas de dichos pedidos, el tercer arreglo contiene las

distancias entre puntos y el cuarto arreglo contiene el tiempo que se demora en ir de un punto a otro. El tercer y cuarto arreglo incluyen al depósito.

2.4.2 Función PAM

Esta función recibe un arreglo que contiene las coordenados de los pedidos y un parámetro K el cual indica el número de grupos en los que se van a dividir los pedidos, este valor K corresponde al número de vehículos disponible para la operación. La función entrega un diccionario cuyo clave será el número de conjunto y su valor una lista con los índices de los pedidos contenidos en el respectivo conjunto.

2.4.3 Función makeChromosome

Esta función recibe el número total de pedidos y entrega una lista que contiene una sucesión aleatoria de números que van desde el 1 hasta el número total de pedidos. Estos números representan los genes y la lista representa un cromosoma.

```
X = [Numeros desde 1 hasta total de pedidos]  
S = []  
Mientras(Longitud(Sucesion) < Total de pedidos)  
    x = Elemento aleatorio de X  
    Se agrega x a S  
    Se elimina x de X
```

Figura 2.1 Pseudocódigo de la función makeChromosome

2.4.4 Función startPopulation

Esta función recibe un parámetro N el cual indica el número de individuos que estarán contenidos en la población, cuando hablamos de individuo nos estamos refiriendo a un cromosoma. La función entrega un arreglo de numpy cuyas filas representan a cada individuo presente en la población inicial.

```
Poblacion = [ ]  
  
Para i = 1 hasta N hacer  
  
    Indiviudo = makeChromosome  
  
    Agregar Indiviudo a Poblacion
```

Figura 2.2 Pseudocódigo de la función startPopulation

2.4.5 Función distance

Esta función recibe un arreglo que contiene la distancia entre puntos y una secuencia de puntos que representan al individuo. La función entrega la duración total del individuo.

2.4.6 Función duration

Esta función recibe un arreglo que contiene los tiempos de demora en ir de un punto a otro y una secuencia de puntos que representan al individuo. La función entrega el tiempo total del individuo.

2.4.7 Función bestsolution

Esta función recibe un arreglo de numpy que representa a la población y nos entrega una fila de este arreglo la cual contiene al individuo de la población cuya función objetivo sea la mejor de todas.

2.4.8 Función fitness

Esta función recibe un arreglo de numpy que representa a la población y nos entrega una fila aleatoria de este arreglo que representa a un individuo. La función calcula la duración de cada individuo en la población de tal manera que el individuo que tenga la menor duración tendrá mayor probabilidad de ser escogido.

2.4.9 Función selection

Esta función recibe un arreglo de numpy con la población que va a pasar por el proceso de selección. Se alterará la población inicial creando una nueva población con el mismo número de individuos, los individuos de la nueva población son escogidos a través de la función fitness.

```
nuevaPoblacion = []  
  
Para i = 1 hasta N hacer  
  
    Indiviudo = fitness(Poblacion)  
  
    Agregar Indiviudo a nuevaPoblacion  
  
Poblacion = nuevaPoblacion
```

Figura 2.3 Pseudocódigo de la función selection

2.4.10 Función generation

Esta función recibe un arreglo que representa una población y parámetro pc el cual indica el porcentaje de individuos que se cruzaran. La función nos entrega una población de nuevos individuos. Para crear los nuevos individuos se seleccionan dos individuos de la población primaria que representan a un padre y una madre, estos se combinarán formando dos nuevos individuos que representan a los hijos, estos hijos reemplazarán a los padres creando una nueva población. Este procedimiento se repetirá hasta completar el porcentaje de cruce.

```

x = N * porcentajeCruce

Si x es impar

    
$$n = \frac{x + 1}{2}$$


    
$$n = \frac{x}{2}$$


Para i = 1 hasta n

    Padre = fitness(Poblacion)

    Madre = fitness(Poblacion)

    Hijo1, Hijo2 = corssing(Padre, Madre)

    Eliminar Padre y Madre de la Poblacion

    Agregar Hijo1 e Hijo2 a la Poblacion

```

Figura 2.4 Pseudocódigo de la función generation

2.4.11 Función crossing

Esta función recibe dos individuos y nos entrega dos hijos. Primero se crea un punto de corte aleatorio. El padre y la madre se dividirán según este punto de corte. Un hijo estará formado por el primer segmento del padre y se van agregando los elementos de la madre que no estén en este primer segmento del padre hasta completar todos los genes. Siguiendo el mismo procedimiento, el segundo hijo estará formado por el primer segmento de la madre y los elementos del padre que no estén incluidos en este primer segmento.

```

u = aleatorio(1, n)

hijo1 = padre(1, u)

hijo2 = madre(1, u)

m = u

Para i = 1 hasta n
    Si madre(i) ∈ hijo1
        u = u + 1
        hijo1(u) = madre(i)
    Si padre(i) ∈ hijo2
        m = m + 1
        hijo2(m) = padre(i)

```

Figura 2.5 Pseudocódigo de la función crossing

2.4.12 Función mutation

Esta función recibe un individuo y un parámetro p_m el cual indica la probabilidad de mutación. La función nos entrega un individuo mutado. Para cada gen en el individuo, se calcula una probabilidad para saber si este gen muta o no. Si la probabilidad es menor al p_m , el gen es intercambiado por otro gen escogido aleatoriamente en el individuo.

```

Para gen en individuo

    u = aleatorio(0,1)

    Si u < porcentajeMutacion

        x = aleatorio(1, n - 1)

        auxliar = gen

        gen = individuo(x)

        individuo(x) = auxliar

```

Figura 2.6 Pseudocódigo de la función mutation

2.4.13 Función geneticAlgorithm

Función que simula el procedimiento genético-biológico. Recibe todos los parámetros involucrados y los arreglos del dataSet. La función entrega una secuencia que representa el mejor individuo de todas las generaciones.

```

Poblacion = startPopulation

MejorSolucion = bestSolution(Poblacion)

Para i = 1 hasta numeroGeneraciones

    Poblacion = selection(Poblacion)

    Poblacion = generation(Poblacion)

    MejorSolucionGeneracion = bestSolution(Poblacion)

    Si MejorSolucionGeneracion < MejorSolucion

        MejorSolucion = MejorSolucionGeneracion

```

Figura 2.7 Pseudocódigo de la función geneticAlgorithm

2.4.14 Función mapping

Esta función recibe al mejor individuo y genera un mapa de la ciudad de Quito donde se muestra la secuencia a seguir por cada camión.

2.5 Consideraciones legales y éticas

Para las programaciones se validó que las jornadas de horas de trabajo ininterrumpidas de los choferes no excedan los máximos según las disposiciones de las autoridades en el acuerdo ministerial no. MDT -2015-, este dicta que conforme a lo establecido en el artículo 325 del código de trabajo, en atención a la naturaleza especial del trabajo de transporte por carreteras, su duración podrá exceder las 8 horas diarias, siempre que se establezcan turnos en la forma que acostumbraren hacerlo los empleadores, de acuerdo con las necesidades del servicio, incluyéndose como jornada de trabajo los sábados, domingos y días de descanso obligatorio. Además, se validaron todas las demás consideraciones legales estipuladas en dicho acuerdo como son las pausas, los tiempos de descanso obligatorio, trabajo suplementario y extraordinario, y el trabajo nocturno. Adicional se tomaron en consideración los permisos de las autoridades respecto a la circulación de los camiones dentro de la ciudad de Quito en horarios determinados. Estas regulaciones son puestas por la Agencia Metropolitana de Tránsito (AMT) del Distrito Metropolitano de Quito y la Federación Nacional de Transporte Pesado del Ecuador donde se puede destacar el pico y placa también conocido como "hoy no circula" que dicta:

"Los propietarios de los vehículos que se encontraren circulando contraviniendo los planes de restricción y de regulación serán sancionados a través del órgano competente del Distrito Metropolitano, por la comisión de la infracción administrativa, con una multa equivalente al 15% de una remuneración básica unificada; en caso de reincidencia, por segunda ocasión, con una multa equivalente al 25% de una remuneración básica unificada y en caso de reincidencia, por tercera ocasión o más, con una multa equivalente al 50% de una remuneración básica unificada." (Ordenanza reformativa 001 de la Ordenanza Metropolitana No. 305 Art. I. 473 (4) Implementación de hoy no circula)



Figura 2.8 Perímetro de Restricción “Hoy no Circula”. [Imagen tomada de la web de la AMT]

La operación que realiza la compañía es catalogada como una actividad comercial, debido a esto se le otorgan permisos para circular por esta zona sin restricción alguna. Estos permisos deben ser renovados cada 180 días.

2.6 Fases del proyecto

Tarea 1. Preparación de los datos

Tarea 1.1. Recopilar y Analizar los datos.

Tarea 1.2. Diseñar de la función que procesa los datos.

Tarea 1.3. Programar la función procesadora de datos.

Tarea 2. Programar el método de dos fases para el ruteo.

Tarea 2.1. Diseñar la primera fase: agrupamiento.

Tarea 2.2. Programar el algoritmo k-medoids.

Tarea 2.3. Diseñar la segunda fase: ruteo.

Tarea 2.4. Programar el algoritmo genético.

Tarea 3. Realizar el programa principal.

Tarea 3.1. Diseñar la función que crea el mapa con las rutas a seguir por los vehículos.

Tarea 3.2. Programar la función que crea el mapa de rutas.

Tarea 3.3. Crear el programa principal, integrando las funciones de procesamiento de datos y mapeo de solución junto con los algoritmos que conforman el método de dos fases.

Tarea 4. Análisis de Resultados

Tarea 4.1. Implementar el programa principal para la obtención de rutas.

Tarea 4.2. Estimar los principales indicadores logísticos y compararlos con los de la situación actual de la empresa

Tarea 4.3. Estimar el costo y beneficio de la implementación de proyecto.

Tarea 4.4. Redactar conclusiones y recomendaciones del proyecto.

Para la tarea 1 primero se recogió la información pertinente para el programa y se le dio un formato predeterminado en un archivo de excel para su posterior lectura en Python, luego se diseñaron las funciones que procesan la información de acuerdo con el formato de los datos de entrada. Al finalizar, se programaron las funciones para obtener la información que se necesitó en el método de dos fases para realizar el ruteo. Todas estas funciones quedaron contenidas en un módulo llamado “dataSet.py”.

Para la tarea 2 primero se diseñaron las funciones que necesito el algoritmo de agrupamiento k-medoids, utilizando los datos obtenidos en la Tarea 1. Luego se programaron dichas funciones y se guardaron en un módulo llamado “Kmedoids.py”. Posteriormente se diseñaron las funciones que necesitó el algoritmo genético para luego ser programadas y probadas con los datos obtenidos del algoritmo k-medoids.

Para la tarea 3 se realizó el programa principal, donde se utilizó todos los módulos de funciones anteriormente programados para obtener las rutas que los vehículos deberán seguir. Adicionalmente se diseñó un módulo de funciones para realizar el mapa de dichas rutas, luego se programaron estas funciones y se almacenaron en un módulo llamado “Mapping.py”. El programa principal es un script con el nombre “Main”.

Para la tarea 4 se calcularon los indicadores del proyecto, luego se compararon los de los indicadores estimados de empresa para obtener las tablas y los gráficos de información. Finalmente se estimó el costo y beneficio de la implementación del proyecto.

2.7 Cronograma de trabajo

A continuación, se detallan las actividades a realizarse y los plazos programados para cumplirlos:

Tabla 2.2 Cronograma de trabajo

Actividades	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6
	16nov-22nov	23nov-29nov	30nov-06dic	07dic-13dic	14dic-20dic	21dic-27dic
Tarea 1.1	X					
Tarea 1.2		x				
Tarea 1.3			X	x		
Tarea 2.1		x				
Tarea 2.2			X	x		
Tarea 2.3		x				
Tarea 2.4			X	x		
Tarea 3.1					x	
Tarea 3.2					x	
Tarea 3.3					x	
Tarea 4.1						X
Tarea 4.2						X
Tarea 4.3						X
Tarea 4.4						X

CAPÍTULO 3

3. RESULTADOS Y ANÁLISIS.

3.1 Programa para el ruteo de vehículos.

Como resultado del proyecto tenemos primero cuatro scripts de Python donde se encuentran el programa principal y tres módulos que contienen las funciones que se utilizarán en este.

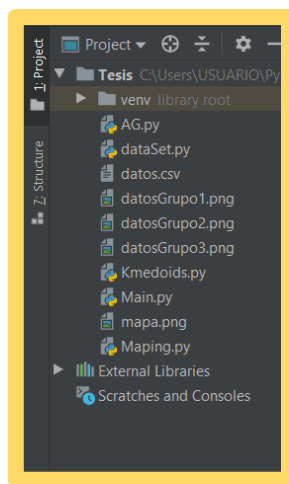


Figura 3.1 Módulos del programa

El primer módulo contiene las funciones para el procesamiento de los datos, los datos son ingresados en un documento de excel y guardado en formato CSV, cada fila debe contener la demanda, la latitud y la longitud de un punto de entrega respectivamente.

	A	B	C	D
1	CANT	LAT	LONG	
2	3	-0.20085844	-78.4311586	
3	10	-0.2018027	-78.3996531	
4	1	-0.2018027	-78.3996531	
5	4	-0.1971608	-78.4533312	
6	3	-0.1800619	-78.388281	
7	1	-0.1704982	-78.4774841	
8	1	-0.1704982	-78.4774841	
9	2	-0.2170888	-78.4015897	
10	10	-0.2935197	-78.5481135	
11	3	-0.2969074	-78.5699332	
12	1	-0.1727721	-78.481059	
13	2	-0.1727721	-78.481059	
14	3	-0.2237133	-78.3442194	
15	3	-0.2533211	-78.5532932	
16	4	-0.245207	-78.5408881	
17	1	-0.245207	-78.5408881	

Figura 3.2 Formato de los datos en el archivo de Excel

Los otros dos módulos corresponden a las funciones del algoritmo k-medoids y el módulo del algoritmo genético.

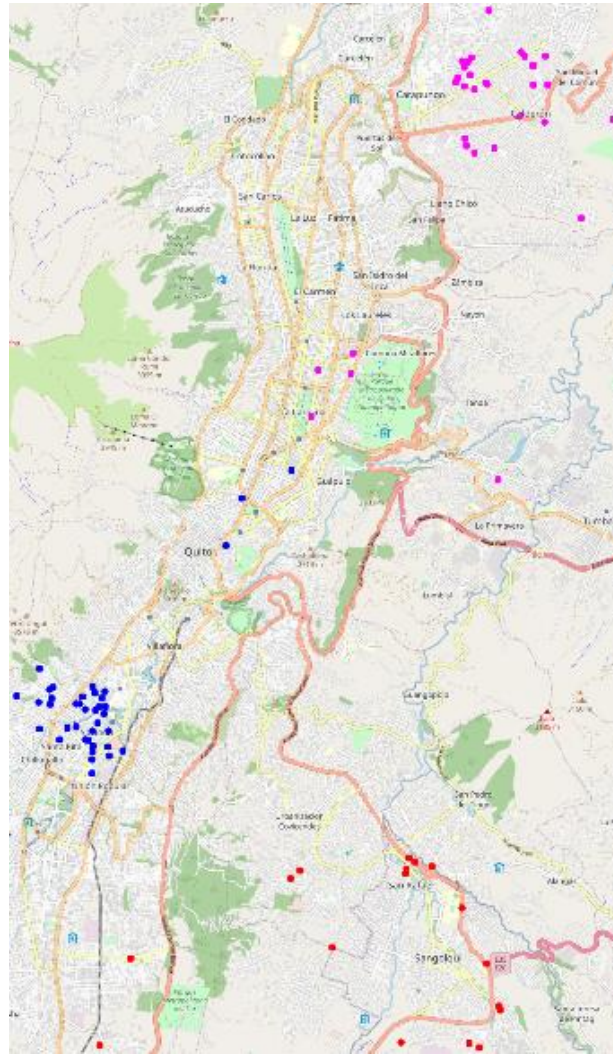
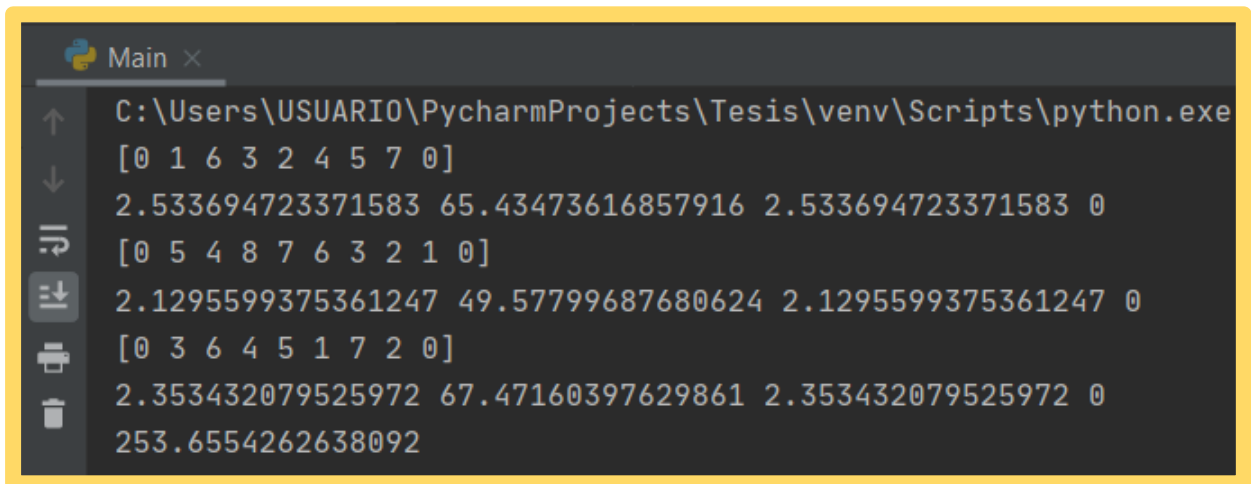


Figura 3.3 Puntos agrupados por el algoritmo k-medoids

El programa nos muestra por consola la secuencia de visita de cada punto junto con el tiempo de recorrido, la distancia recorrida y la cantidad de demanda no atendida de darse el caso. Al final nos entrega el tiempo de compilación en segundos.



```
Main x
C:\Users\USUARIO\PycharmProjects\Tesis\venv\Scripts\python.exe
[0 1 6 3 2 4 5 7 0]
2.533694723371583 65.43473616857916 2.533694723371583 0
[0 5 4 8 7 6 3 2 1 0]
2.1295599375361247 49.57799687680624 2.1295599375361247 0
[0 3 6 4 5 1 7 2 0]
2.353432079525972 67.47160397629861 2.353432079525972 0
253.6554262638092
```

Figura 3.4 Resultados por consola

3.2 Programa para graficar las rutas.

Para una mejor apreciación del orden de visita, las funciones del módulo “mapping” nos crea 3 imágenes en formato PNG con la ruta que debe seguir cada camión y estas imágenes se guardan automáticamente en la carpeta del proyecto. El mapa para cada camión se muestra de color diferente como se puede apreciar en las siguientes figuras:

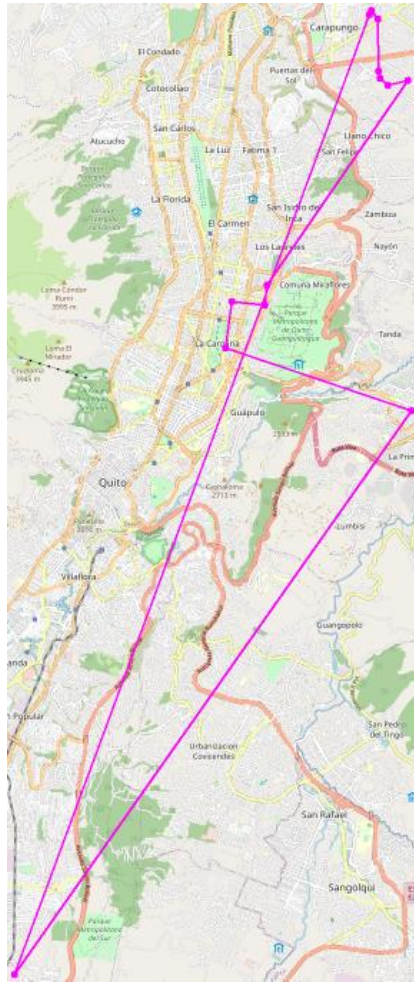


Figura 3.5 Ejemplo de mapa para la ruta 1



Figura 3.6 Ejemplo de mapa para la ruta 2

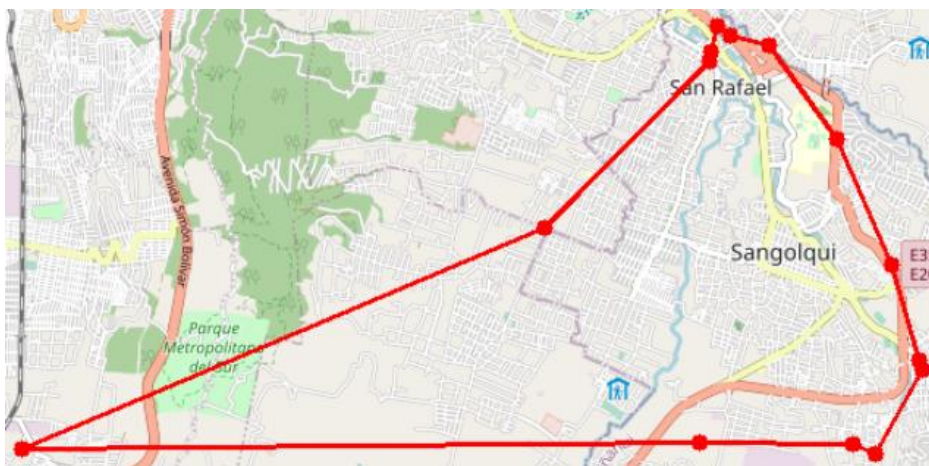


Figura 3.7 Ejemplo de mapa para la ruta 3

Para el proyecto se utilizaron como parámetros: 100 generaciones, el tamaño de la población fue de 80 individuos, el porcentaje de cruce fue de 50%, el porcentaje de mutación de 5%, el tiempo máximo de 8 horas, la capacidad de los vehículos fue de 200 unidades de carga, el tiempo de carga fue de media hora y el tiempo de descarga fue de 0.017 horas por unidad de carga. Implementado el programa durante 5 días de operación se obtuvieron los siguientes resultados:

Tabla 3.1 Resultados del día 1

DIA 1			
	Distancia (Km)	Tiempo (H)	No Atendido
Camión 1	65.98	5.29	0
Camión 2	4.71	7.56	106
Camión 3	49.71	2.86	92

Tabla 3.2 Resultados del día 2

DIA 2			
	Distancia (Km)	Tiempo (H)	No Atendido
Camión 1	36.15	6.72	78
Camión 2	12.16	6.01	103
Camión 3	15.21	7.38	91

Tabla 3.3 Resultados del día 3

DIA 3			
	Distancia (KM)	Tiempo (H)	No Atendido
Camión 1	44.29	3.21	0
Camión 2	134.58	6.72	93
Camión 3	62.61	3.03	5

Tabla 3.4 Resultados del día 4

DIA 4			
	Distancia (Km)	Tiempo (H)	No Atendido
Camión 1	43.24	1.26	0
Camión 2	43.36	7.03	90
Camión 3	130.70	7.54	14

Tabla 3.5 Resultados del día 5

DIA 5			
	Distancia (Km)	Tiempo (H)	No Atendido
Camión 1	76.21	6.98	0
Camión 2	244.35	7.08	229
Camión 3	35.00	5.55	122

3.3 Análisis de los principales indicadores logísticos.

Para medir la eficacia del método se compararon los principales indicadores logísticos de la operación obtenidos del proyecto vs los estimados de la empresa. De esta manera se puede apreciar que el porcentaje de puntos de entrega no atendidos se redujo en un 7%. El número total de kilómetros recorridos se redujo aproximadamente en 37,24%, el gasto de combustible se redujo en un 42,77% y el gasto anual de neumáticos mejoró en \$ 1046,83 para el período analizado. Para estimar el gasto de combustible del proyecto se utilizó el precio del combustible diésel en esas fechas el cual fue \$1.359 dólares por galón.

Tabla 3.6 Porcentajes de demanda no atendida

	Demanda	Demanda No Atendida Proyecto	Porcentaje	Demanda No Atendida Real	Porcentaje
Dia 1	761	198	26%	260	34%
Dia 2	1149	272	24%	362	32%
Dia 3	427	98	23%	178	42%
Dia 4	688	104	15%	211	31%
Dia 5	1193	351	29%	473	40%

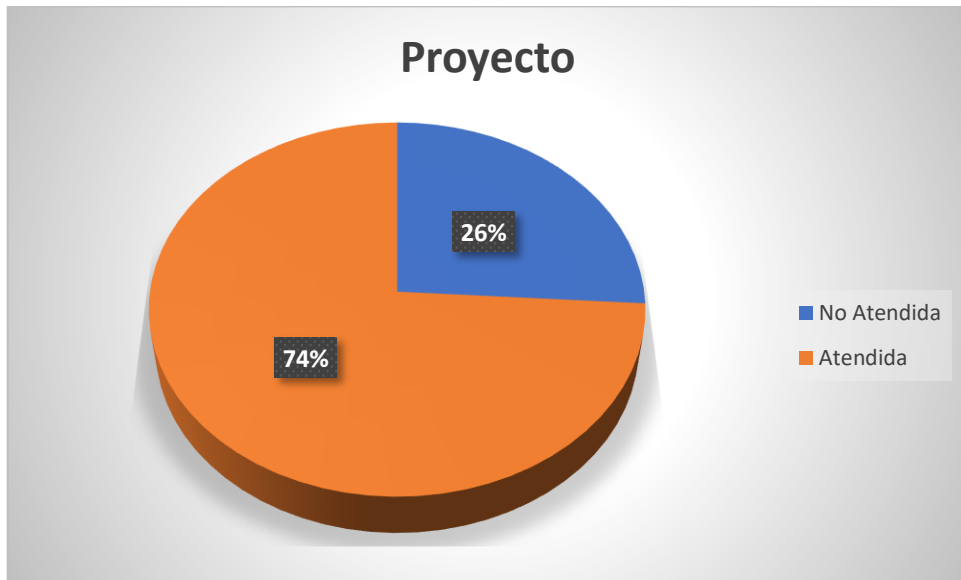


Figura 3.8 Porcentaje de demanda no atendida del proyecto

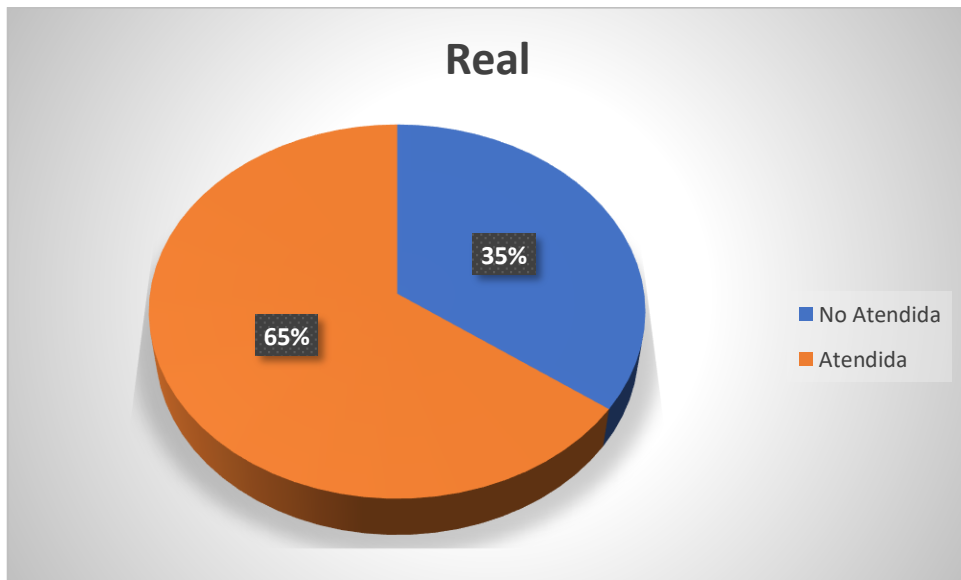


Figura 3.9 Porcentaje de demanda no atendida real

Tabla 3.7 Kilómetros recorridos y gasto en combustible

Período	Kilómetros recorridos (Km)	Gasto de combustible (\$)
Día 1	120,40	10,35
Día 2	63,52	5,46
Día 3	241,48	20,77
Día 4	217,30	18,69
Día 5	355,56	30,58

Tabla 3.8 Porcentaje de reducción de kilómetros recorridos

Kilómetros recorridos (Km)		
Real Aproximado	Proyecto	Porcentaje
1590,60	998,26	37,24%

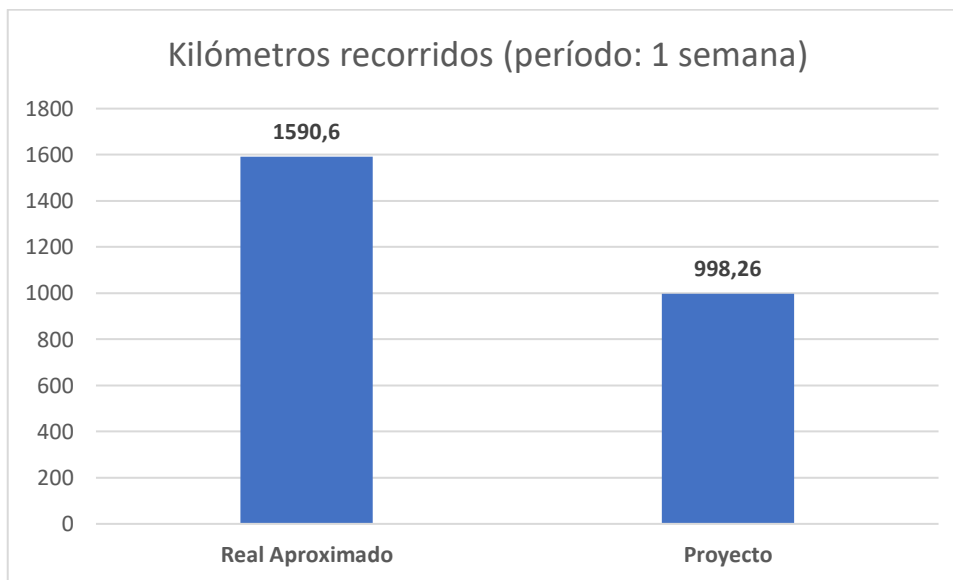


Figura 3.10 Total de kilómetros recorridos

Tabla 3.9 Porcentaje de reducción de gasto en combustible

Gasto en combustible		
Real	Proyecto	Ahorro
\$ 150	\$ 85,85	\$ 64,15

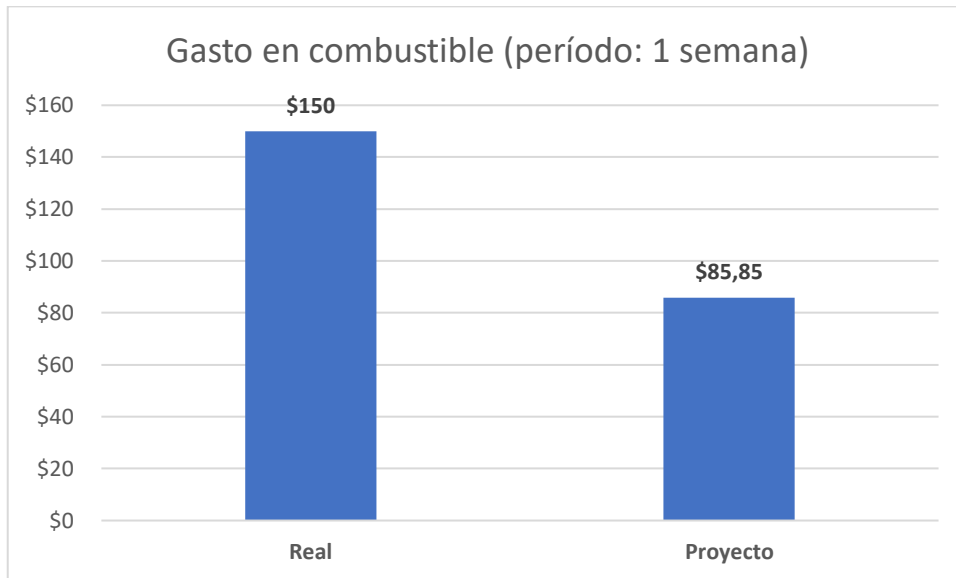


Figura 3.11 Total de gasto en combustible

3.4 Análisis Costo-Beneficio

El costo de implementación del proyecto que incluye desarrollo, adquisición de activos, capacitaciones, entre otros; resultó en aproximadamente \$ 3116,67 dólares, mientras que el beneficio se ve reflejado en el ahorro de combustible y consumo de neumáticos. Se ha proyectado el uso de la herramienta por un período aproximado de 1 año, lo que nos refleja un beneficio aproximado de \$ 4126,01 dólares. El factor entre el beneficio y el costo es igual a 1.32.

Tabla 3.10 Ahorro anual aproximado en neumáticos

Ahorro en neumáticos		
Real	Proyecto	Ahorro
\$ 2811,02	\$ 1,764.20	\$ 1046,83

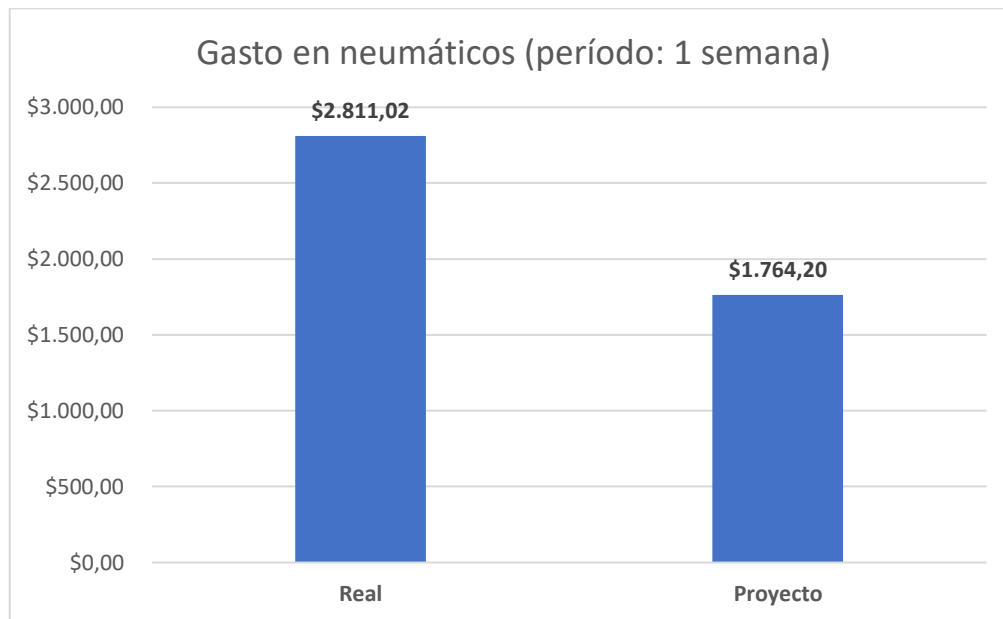


Figura 3.12 Gasto en neumáticos

Tabla 3.11 Beneficios de la implementación

Ahorro semanal (Combustible)	Ahorro anual (Neumáticos)	Meses de implementación	Beneficios de implementación
\$ 64,15	\$ 1046,83	12	\$ 4126.01

Tabla 3.12 Costos de la implementación

Costo de Desarrollo (incluye capacitación a conductores)	\$ 1600
Computadora con procesador CORE i7 8th generación (Requisitos estándar)	\$ 1200
Costo de aprendizaje/acompañamiento del encargado del software (período: 1 semana)	\$ 116,67
Costo acompañamiento a los 03 conductores medio recorrido (período: 1 semana)	\$ 200
Costo de implementación	\$ 3116,67

Tabla 3.13 Factor entre el beneficio y el costo de la implementación

B/C	\$ 4126.01 / \$ 3116,67	1.32
------------	-------------------------	-------------

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones.

- El programa de ruteo implementando el método de dos fases realiza un agrupamiento satisfactorio al reducir los tiempos de transporte entre los puntos de cada grupo. Adicional, el algoritmo genético dentro del programa nos entrega soluciones aceptables en un tiempo computacional prudente utilizando los parámetros establecidos en el proyecto.
- El programa nos entrega un mapa claro con la secuencia de visita para cada camión con un color diferente para facilitar la visualización. Además, los mapas solo muestran el área donde se encuentran los puntos de entrega del camión asignado para una mejor apreciación de la secuencia de visitas.
- La implementación del proyecto significa un ahorro importante para la empresa en los costos inherentes a la operación tales como neumáticos, combustible, mantenimiento, etc. Además, la reducción en el porcentaje de los pedidos no realizados al final del día influye directamente con la satisfacción de los clientes que son atendidos.
- Podemos concluir que es pertinente realizar el proyecto debido que el análisis de costo y el beneficio mostró un valor para el factor B/C superior a uno, esto nos indica que el proyecto es rentable.

4.2 Recomendaciones.

- Se sugiere seguir mejorando el programa, si bien las soluciones han dado un ahorro positivo a la empresa, el algoritmo genético puede ser mejorado combinándolo con heurísticas que reduzcan el tiempo de compilado, de la misma manera podría entregar soluciones que mejoren el recorrido de los camiones.

- Se recomienda buscar alternativas menos costosas que la API de Google para encontrar tiempos y distancias entre los puntos de entrega, pues es determinante para el correcto funcionamiento del programa y más fiel a la realidad que la obtenida por la fórmula del semiverso.
- Se aconseja concientizar a los choferes y despachadores para cumplir con los tiempos de carga/descarga establecidos, puesto que su parte del proceso determina la efectividad del programa para obtener los resultados esperados. El aplicativo desarrollado brinda la facilidad de modificar los tiempos de carga/descarga.

BIBLIOGRAFÍA

Adewumi AO, Adeleke OJ (2018) A survey of recent advances in vehicle routing problems. *Int J Syst Assur Eng Manag* 9(1):155–172

Augerat P, Belenguer JM, Benavent E, Corberan A, Naddef D, Rinaldi G (1998) Computational results with a branch-and-cut code for the capacitated vehicle routing problem. Technical report 949-M, Universit Joseph Fourier, Grenoble, France

Baldacci R, Christofdes N, Mingozzi A (2008) An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Math Program Ser A* 115(2):351–385

Comert, S. E., Yazgan, H. R., Kir, S., & Yener, F. (2018). A cluster first-route second approach for a capacitated vehicle routing problem: A case study. *International Journal of Procurement Management*, 11(4), 399–419. <https://doi.org/10.1504/IJPM.2018.092766>

Christofdes N, Eilon S (1969) An algorithm for the vehicle-dispatching problem. *J Oper Res Soc* 20(3):309–318

Dantzig GB, Ramser JH (1959) The truck dispatching problem. *Manag Sci* 6(1):80–91

Fukasawa R, Longo H, Lysgaard J, de Arago MP, Reis M, Uchoa E, Werneck RF (2006) Robust branchand-cut-and-price for the capacitated vehicle routing problem. *Math Program* 106(3):491–511

Dean, W. (2016) Computational Complexity Theory. The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University.

Laporte, G., Nobert, Y. and Taillefer, S. (1988) ‘Solving a family of multi-depot vehicle routing and location-routing problems’, *Transportation Science*, Vol. 22, No. 3, pp.161–172.

Letchford AN, Eglese RW, Lysgaard J (2002) Multistars, partial multistars and the capacitated vehicle routing problem. *Math Program* 94(1):21–40

Lysgaard J, Letchford AN, Eglese RW (2004) A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Math Program* 100(2):423–445

Michalewicz, Z. (1996), *Genetic Algorithms + Data Structures = Evolution Programs*, tercera edición, Springer Verlag.

Park, H.S., Jun, C.H. (2009) A simple and fast algorithm for K-medoids clustering, *Expert Systems with Applications*, 36, (2), 3336–3341.

Pecin D, Pessoa A, Poggi M, Uchoa E (2014) Improved branch-cut-and-price for capacitated vehicle routing. *Integer Program Comb Optim* 8494:393–403

Pecin D, Pessoa A, Poggi M, Uchoa E (2017) Improved branch-cut-and-price for capacitated vehicle routing. *Math Program Comput* 9(1):61–100

Ortega, G. P. M., Rodríguez, C. J. V., Rodríguez, M. O., & Atencia, J. G. (2018). Optimization of vehicle routing in the distribution of products of a company in the city of Sincelejo - Colombia | Optimización del ruteo de Vehículos en la distribución de productos de una empresa en la Ciudad de Sincelejo - Colombia. *Proceedings of the LACCEI International Multi-Conference for Engineering, Education and Technology, 2018-July*.
<https://doi.org/10.18687/LACCEI2018.1.1.19>

Sbai, I., Krichen, S., & Limam, O. (2020). Two meta-heuristics for solving the capacitated vehicle routing problem: the case of the Tunisian Post Office. *Operational Research*.
<https://doi.org/10.1007/s12351-019-00543-8>

Taha, H. A. (2012). *Investigación de Operaciones*. Novena edición, Naucalpan de Juárez, Mexico: Pearson Educación.

Uchoa E, Pecin D, Pessoa A, Poggi M, Vidal T, Subramanian A (2017) New benchmark instances for the capacitated vehicle routing problem. *Eur J Oper Res* 257(3):845–858

Venkates, G., Gehrke, J. and Ramakrishnan, R. (1999) ‘Mining very large databases’, *IEEE Computer*, Vol. 32, No. 9, pp.38–45.

Cook, W., Cunningham, W., Pulleyblank, W., Schrijver, A. (1997) Combinatorial Optimization. Primera edición, New Jersey, John Wiley & Sons.

ANEXOS

Datos

FECHA	FACT	CANT	LONG	LAT	PROVINC
02/01/2021	000015073	3	-78.431158605963	-0.200858437456191	PICHINCHA
02/01/2021	000015069	10	-78.3996531	-0.2018027	PICHINCHA
02/01/2021	000015069	1	-78.3996531	-0.2018027	PICHINCHA
02/01/2021	000015074	4	-78.4533312	-0.1971608	PICHINCHA
02/01/2021	000015068	3	-78.388281	-0.1800619	PICHINCHA
02/01/2021	000015075	1	-78.4774841	-0.1704982	PICHINCHA
02/01/2021	000015075	1	-78.4774841	-0.1704982	PICHINCHA
02/01/2021	000015071	2	-78.4015897	-0.2170888	PICHINCHA
02/01/2021	000015082	10	-78.5481135	-0.2935197	PICHINCHA
02/01/2021	000015083	3	-78.5699332	-0.2969074	PICHINCHA
02/01/2021	000015078	1	-78.481059	-0.1727721	PICHINCHA
02/01/2021	000015078	2	-78.481059	-0.1727721	PICHINCHA
02/01/2021	000015067	3	-78.3442194	-0.2237133	PICHINCHA
02/01/2021	000015081	3	-78.5532932	-0.2533211	PICHINCHA
02/01/2021	000015079	4	-78.5408881	-0.245207	PICHINCHA
02/01/2021	000015079	1	-78.5408881	-0.245207	PICHINCHA
02/01/2021	000015076	2	-78.4865749	-0.1307724	PICHINCHA
02/01/2021	000015072	2	-78.4265451	-0.1964994	PICHINCHA
02/01/2021	000015077	4	-78.483599	-0.167235	PICHINCHA
02/01/2021	000015080	12	-78.5433578	-0.25081179	PICHINCHA
02/01/2021	000015080	3	-78.5433578	-0.25081179	PICHINCHA
02/01/2021	000015070	1	-78.4042798	-0.2146015	PICHINCHA
31/12/2020	000015005	1	-78.5339162	-0.2581323	PICHINCHA
31/12/2020	000014997	7	-78.5434739291667	-0.243281344883144	PICHINCHA
31/12/2020	000014996	1	-78.52549274	-0.249794293	PICHINCHA
31/12/2020	000014996	1	-78.52549274	-0.249794293	PICHINCHA
31/12/2020	000014996	1	-78.52549274	-0.249794293	PICHINCHA
31/12/2020	000015058	4	-78.4526204	-0.2183992	PICHINCHA
31/12/2020	000015052	1	-78.52014868	-0.245616585	PICHINCHA
31/12/2020	000015034	10	-78.5027185	-0.0930568	PICHINCHA
31/12/2020	000015035	10	-78.5027185	-0.0930568	PICHINCHA
31/12/2020	000015040	8	-78.543457	-0.256642	PICHINCHA
31/12/2020	000015040	4	-78.543457	-0.256642	PICHINCHA
31/12/2020	000015040	6	-78.543457	-0.256642	PICHINCHA
31/12/2020	000015029	1	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015029	2	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015029	1	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015029	1	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015029	1	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015029	3	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015029	1	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015029	2	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015029	2	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015029	1	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015029	2	-78.476217	-0.170734	PICHINCHA
31/12/2020	000015015	1	-78.491568	-0.203165	PICHINCHA
31/12/2020	000015002	1	-78.5326317	-0.2610139	PICHINCHA
31/12/2020	000014993	2	-78.5250552	-0.2502255	PICHINCHA
31/12/2020	000015009	0	-78.5303053	-0.26757629	PICHINCHA
31/12/2020	000015009	1	-78.5303053	-0.26757629	PICHINCHA
31/12/2020	000015022	2	-78.4866795	-0.1725536	PICHINCHA
31/12/2020	000015039	2	-78.4726365376264	0.0892318971455097	PICHINCHA
31/12/2020	000015047	2	-78.53301142	-0.24423461	PICHINCHA

31/12/2020	000015050	3	-78.5224406	-0.245534	PICHINCHA
31/12/2020	000015001	2	-78.533945	-0.261294	PICHINCHA
31/12/2020	000014999	4	-78.5408857	-0.2630201	PICHINCHA
31/12/2020	000015056	1	-78.520773	-0.241233	PICHINCHA
31/12/2020	000015006	1	-78.5330817	-0.2569089	PICHINCHA
31/12/2020	000015049	1	-78.52916835	-0.23831968	PICHINCHA
31/12/2020	000015003	1	-78.53422299	-0.259243338	PICHINCHA
31/12/2020	000015062	1	-78.537030000000	-0.253493000000	PICHINCHA
31/12/2020	000015064	57	-78.503784	-0.2540165	PICHINCHA
31/12/2020	000015023	2	-78.4868277	-0.1724336	PICHINCHA
31/12/2020	000015024	2	-78.4868277	-0.1724336	PICHINCHA
31/12/2020	000015024	2	-78.4868277	-0.1724336	PICHINCHA
31/12/2020	000015018	4	-78.4872172772884	-0.177485458552837	PICHINCHA
31/12/2020	000015018	4	-78.4872172772884	-0.177485458552837	PICHINCHA
31/12/2020	000015044	2	-78.529457	-0.246226	PICHINCHA
31/12/2020	000015044	2	-78.529457	-0.246226	PICHINCHA
31/12/2020	000015044	4	-78.529457	-0.246226	PICHINCHA
31/12/2020	000015008	1	-78.5291699	-0.2563374	PICHINCHA
31/12/2020	000015063	3	-78.48389821	-0.166800417	PICHINCHA
31/12/2020	000015019	7	-78.487041	-0.1767296	PICHINCHA
31/12/2020	000015054	1	-78.52110254	-0.241848877	PICHINCHA
31/12/2020	000015004	2	-78.53414428	-0.258840965	PICHINCHA
31/12/2020	000015041	2	-78.4717466309667	-0.122788636945188	PICHINCHA
31/12/2020	000015007	6	-78.53022	-0.2591152	PICHINCHA
31/12/2020	000015045	1	-78.53231265	-0.24742958	PICHINCHA
31/12/2020	000015045	1	-78.53231265	-0.24742958	PICHINCHA
31/12/2020	000015013	0	-78.54668833	-0.29979667	PICHINCHA
31/12/2020	000015038	1	-78.47236529	-0.088757690	PICHINCHA
31/12/2020	000015011	1	-78.529151	-0.263475	PICHINCHA
31/12/2020	000014995	5	-78.5302924	-0.2501219	PICHINCHA
31/12/2020	000014994	5	-78.5252031	-0.2500427	PICHINCHA
31/12/2020	000014994	1	-78.5252031	-0.2500427	PICHINCHA
31/12/2020	000015021	2	-78.4862149693072	-0.172614776529372	PICHINCHA
31/12/2020	000014998	20	-78.5413937	-0.2500363	PICHINCHA
31/12/2020	000014990	5	-78.5336419	-0.2450364	PICHINCHA
31/12/2020	000015033	8	-78.502191	-0.098568	PICHINCHA
31/12/2020	000015033	2	-78.502191	-0.098568	PICHINCHA
31/12/2020	000015061	1	-78.4422576	-0.2298758	PICHINCHA
31/12/2020	000015061	1	-78.4422576	-0.2298758	PICHINCHA
31/12/2020	000015061	1	-78.4422576	-0.2298758	PICHINCHA
31/12/2020	000015010	12	-78.520824	-0.254692	PICHINCHA
31/12/2020	000015010	3	-78.520824	-0.254692	PICHINCHA
31/12/2020	000015010	1	-78.520824	-0.254692	PICHINCHA
31/12/2020	000015025	1	-78.486667	-0.172241	PICHINCHA
31/12/2020	000015025	1	-78.486667	-0.172241	PICHINCHA
31/12/2020	000015026	2	-78.4867001	-0.1720066	PICHINCHA
31/12/2020	000015026	1	-78.4867001	-0.1720066	PICHINCHA
31/12/2020	000015036	1	-78.4846509	-0.1556482	PICHINCHA
31/12/2020	000015036	1	-78.4846509	-0.1556482	PICHINCHA
31/12/2020	000015032	1	-78.477571	-0.1547	PICHINCHA
31/12/2020	000015032	1	-78.477571	-0.1547	PICHINCHA
31/12/2020	000015012	16	-78.5480679	-0.2795984	PICHINCHA
31/12/2020	000015012	4	-78.5480679	-0.2795984	PICHINCHA
31/12/2020	000015060	1	-78.4095506	-0.2167041	PICHINCHA
31/12/2020	000015060	1	-78.4095506	-0.2167041	PICHINCHA
31/12/2020	000015030	1	-78.4795011	-0.1664999	PICHINCHA
31/12/2020	000015030	1	-78.4795011	-0.1664999	PICHINCHA

31/12/2020	000015055	1	-78.520665	-0.241791	PICHINCHA
31/12/2020	000015031	1	-78.476647	-0.157586	PICHINCHA
31/12/2020	000015017	5	-78.481723	-0.195849	PICHINCHA
31/12/2020	000015037	2	-78.468076	-0.085302	PICHINCHA
31/12/2020	000015037	2	-78.468076	-0.085302	PICHINCHA
31/12/2020	000015037	1	-78.468076	-0.085302	PICHINCHA
31/12/2020	000015014	1			PICHINCHA
31/12/2020	000015000	2	-78.5349213	-0.2605712	PICHINCHA
31/12/2020	000015027	3	-78.48654465	-0.171536023	PICHINCHA
31/12/2020	000015057	1	-78.50996743	-0.213442062	PICHINCHA
31/12/2020	000015028	1	-78.4774618	-0.1705215	PICHINCHA
31/12/2020	000015059	10	-78.3666596	-0.1830998	PICHINCHA
31/12/2020	000015059	1	-78.3666596	-0.1830998	PICHINCHA
31/12/2020	000015020	6	-78.4860811	-0.1727978	PICHINCHA
31/12/2020	000015016	8	-78.4924674	-0.1899829	PICHINCHA
31/12/2020	000015016	2	-78.4924674	-0.1899829	PICHINCHA
31/12/2020	000015053	8	-78.51792404	-0.244660209	PICHINCHA
31/12/2020	000015053	2	-78.51792404	-0.244660209	PICHINCHA
31/12/2020	000015048	1	-78.5265124	-0.246556	PICHINCHA
31/12/2020	000015048	1	-78.5265124	-0.246556	PICHINCHA
31/12/2020	000015051	1	-78.5226026	-0.244834	PICHINCHA
31/12/2020	000015046	10	-78.533784	-0.2449618	PICHINCHA
31/12/2020	000015046	2	-78.533784	-0.2449618	PICHINCHA
31/12/2020	000014991	1	-78.5229577	-0.2609349	PICHINCHA
31/12/2020	000014991	0	-78.5229577	-0.2609349	PICHINCHA
31/12/2020	000014992	2	-78.5253044	-0.2499737	PICHINCHA
31/12/2020	000015042	8	-78.4398051	-0.1581621	PICHINCHA
31/12/2020	000015042	2	-78.4398051	-0.1581621	PICHINCHA
31/12/2020	000015043	7	-78.4398051	-0.1581621	PICHINCHA
30/12/2020	000014922	3	-78.4809768	-0.1850295	PICHINCHA
30/12/2020	000014943	1	-78.4298628	-0.0672668	PICHINCHA
30/12/2020	000014983	1	-78.4506067167968	-0.325580444186926	PICHINCHA
30/12/2020	000014985	3	-78.449608	-0.327537	PICHINCHA
30/12/2020	000014985	1	-78.449608	-0.327537	PICHINCHA
30/12/2020	000014985	1	-78.449608	-0.327537	PICHINCHA
30/12/2020	000014985	1	-78.449608	-0.327537	PICHINCHA
30/12/2020	000014966	4	-78.441949	-0.315587	PICHINCHA
30/12/2020	000014966	3	-78.441949	-0.315587	PICHINCHA
30/12/2020	000014915	10	-78.5487063	-0.3671759	PICHINCHA
30/12/2020	000014915	1	-78.5487063	-0.3671759	PICHINCHA
30/12/2020	000014915	1	-78.5487063	-0.3671759	PICHINCHA
30/12/2020	000014904	3	-78.54201	-0.33838	PICHINCHA
30/12/2020	000014904	1	-78.54201	-0.33838	PICHINCHA
30/12/2020	000014982	3	-78.4530512150377	-0.325722098350525	PICHINCHA
30/12/2020	000014982	1	-78.4530512150377	-0.325722098350525	PICHINCHA
30/12/2020	000014982	1	-78.4530512150377	-0.325722098350525	PICHINCHA
30/12/2020	000014890	5	-78.558175	-0.302605	PICHINCHA
30/12/2020	000014939	3	-78.4271513	-0.0885956	PICHINCHA
30/12/2020	000014970	1	-78.4878032	-0.3077995	PICHINCHA
30/12/2020	000014927	2	-78.4877	-0.174942	PICHINCHA
30/12/2020	000014927	2	-78.4877	-0.174942	PICHINCHA
30/12/2020	000014918	2	-78.4772469	-0.1819162	PICHINCHA
30/12/2020	000014892	1	-78.5480527	-0.3046679	PICHINCHA
30/12/2020	000014892	1	-78.5480527	-0.3046679	PICHINCHA
30/12/2020	000014923	2	-78.47904	-0.185223	PICHINCHA
30/12/2020	000014984	5	-78.4500110149384	-0.325863542966545	PICHINCHA

30/12/2020	000014917	10	-78.4582889	-0.1547137	PICHINCHA
30/12/2020	000014950	1	-78.438778	-0.093825	PICHINCHA
30/12/2020	000014948	3	-78.4419817	-0.092026	PICHINCHA
30/12/2020	000014934	2	-78.4272685	-0.1065422	PICHINCHA
30/12/2020	000014934	1	-78.4272685	-0.1065422	PICHINCHA
30/12/2020	000014894	3	-78.5473384	-0.309689822	PICHINCHA
30/12/2020	000014893	7	-78.547533	-0.308061	PICHINCHA
30/12/2020	000014925	3	-78.450505	-0.1030716	PICHINCHA
30/12/2020	000014911	3	-78.5486327	-0.3587156	PICHINCHA
30/12/2020	000014929	8	-78.440849	-0.112198	PICHINCHA
30/12/2020	000014929	10	-78.440849	-0.112198	PICHINCHA
30/12/2020	000014929	2	-78.440849	-0.112198	PICHINCHA
30/12/2020	000014916	1	-78.5707985	-0.5042653	PICHINCHA
30/12/2020	000014910	30	-78.5518836	-0.3519408	PICHINCHA
30/12/2020	000014886	30	-78.5594742	-0.2771201	PICHINCHA
30/12/2020	000014953	7	-78.4741212241352	-0.0882095564156771	PICHINCHA
30/12/2020	000014944	2	-78.435014	-0.075923	PICHINCHA
30/12/2020	000014961	50	-78.4582726	-0.3515073	PICHINCHA
30/12/2020	000014888	4	-78.566455	-0.30736	PICHINCHA
30/12/2020	000014903	10	-78.537513	-0.3369468	PICHINCHA
30/12/2020	000014988	1	-78.4456588793546	-0.338020529597998	PICHINCHA
30/12/2020	000014988	1	-78.4456588793546	-0.338020529597998	PICHINCHA
30/12/2020	000014920	1	-78.4798275	-0.1790557	PICHINCHA
30/12/2020	000014919	4	-78.4800861	-0.1802057	PICHINCHA
30/12/2020	000014919	2	-78.4800861	-0.1802057	PICHINCHA
30/12/2020	000014905	1	-78.545045	-0.339918	PICHINCHA
30/12/2020	000014930	1	-78.4413984417915	-0.106950150802732	PICHINCHA
30/12/2020	000014930	1	-78.4413984417915	-0.106950150802732	PICHINCHA
30/12/2020	000014959	10	-78.5233759	-0.3870364	PICHINCHA
30/12/2020	000014989	60	-78.4616534	-0.0828271	PICHINCHA
30/12/2020	000014989	15	-78.4616534	-0.0828271	PICHINCHA
30/12/2020	000014989	130	-78.4616534	-0.0828271	PICHINCHA
30/12/2020	000014989	40	-78.4616534	-0.0828271	PICHINCHA
30/12/2020	000014989	90	-78.4616534	-0.0828271	PICHINCHA
30/12/2020	000014989	20	-78.4616534	-0.0828271	PICHINCHA
30/12/2020	000014989	90	-78.4616534	-0.0828271	PICHINCHA
30/12/2020	000014989	20	-78.4616534	-0.0828271	PICHINCHA
30/12/2020	000014981	2	-78.4494995	-0.3247484	PICHINCHA
30/12/2020	000014981	2	-78.4494995	-0.3247484	PICHINCHA
30/12/2020	000014962	1	-78.437444	-0.352872	PICHINCHA
30/12/2020	000014896	2	-78.5524728	-0.3182525	PICHINCHA
30/12/2020	000014908	3	-78.55164109	-0.350119849	PICHINCHA
30/12/2020	000014932	10	-78.4847082	-0.1941034	PICHINCHA
30/12/2020	000014954	10	-78.47324464	-0.078091238	PICHINCHA
30/12/2020	000014895	2	-78.55406212	-0.317828273	PICHINCHA
30/12/2020	000014895	2	-78.55406212	-0.317828273	PICHINCHA
30/12/2020	000014895	3	-78.55406212	-0.317828273	PICHINCHA
30/12/2020	000014931	1	-78.4327391	-0.1099366	PICHINCHA
30/12/2020	000014889	10	-78.560584	-0.308455	PICHINCHA
30/12/2020	000014889	1	-78.560584	-0.308455	PICHINCHA
30/12/2020	000014975	2	-78.4562907367945	-0.305976220406592	PICHINCHA
30/12/2020	000014952	25	-78.4350732	-0.0957957	PICHINCHA
30/12/2020	000014952	1	-78.4350732	-0.0957957	PICHINCHA
30/12/2020	000014952	1	-78.4350732	-0.0957957	PICHINCHA
30/12/2020	000014882	8	-78.5415811	-0.2707999	PICHINCHA
30/12/2020	000014882	1	-78.5415811	-0.2707999	PICHINCHA

30/12/2020	000014882	2	-78.5415811	-0.2707999	PICHINCHA
30/12/2020	000014933	4	-78.431815	-0.1130306	PICHINCHA
30/12/2020	000014901	30	-78.534578	-0.3315066	PICHINCHA
30/12/2020	000014955	21	-78.5025803	-0.0934284	PICHINCHA
30/12/2020	000014951	1	-78.4379715844989	0.0942655233666301	PICHINCHA
30/12/2020	000014978	1	-78.4563242	-0.3213904	PICHINCHA
30/12/2020	000014928	1	-78.4840478	-0.1750637	PICHINCHA
30/12/2020	000014928	1	-78.4840478	-0.1750637	PICHINCHA
30/12/2020	000014900	1	-78.53012063	-0.329056964	PICHINCHA
30/12/2020	000014900	1	-78.53012063	-0.329056964	PICHINCHA
30/12/2020	000014937	1	-78.4197022	-0.0967632	PICHINCHA
30/12/2020	000014937	1	-78.4197022	-0.0967632	PICHINCHA
30/12/2020	000014885	1	-78.511186	-0.218397	PICHINCHA
30/12/2020	000014885	1	-78.511186	-0.218397	PICHINCHA
30/12/2020	000014941	7	-78.4233830496669	0.0906046433374286	PICHINCHA
30/12/2020	000014941	2	-78.4233830496669	0.0906046433374286	PICHINCHA
30/12/2020	000014941	1	-78.4233830496669	0.0906046433374286	PICHINCHA
30/12/2020	000014941	1	-78.4233830496669	0.0906046433374286	PICHINCHA
30/12/2020	000014960	20	-78.5221716	-0.3853403	PICHINCHA
30/12/2020	000014891	4	-78.553368	-0.306087	PICHINCHA
30/12/2020	000014891	1	-78.553368	-0.306087	PICHINCHA
30/12/2020	000014926	10	-78.481664	-0.2041153	PICHINCHA
30/12/2020	000014972	1	-78.4251242	-0.1954076	PICHINCHA
30/12/2020	000014979	3	-78.4497153	-0.3226658	PICHINCHA
30/12/2020	000014979	2	-78.4497153	-0.3226658	PICHINCHA
30/12/2020	000014938	1	-78.4267382	-0.090192	PICHINCHA
30/12/2020	000014938	1	-78.4267382	-0.090192	PICHINCHA
30/12/2020	000014897	1	-78.5560051	-0.3225924	PICHINCHA
30/12/2020	000014969	7	-78.4576105	-0.304888	PICHINCHA
30/12/2020	000014921	1	-78.481381	-0.179878	PICHINCHA
30/12/2020	000014936	7	-78.425112	-0.1013362	PICHINCHA
30/12/2020	000014936	1	-78.425112	-0.1013362	PICHINCHA
30/12/2020	000014936	1	-78.425112	-0.1013362	PICHINCHA
30/12/2020	000014909	7	-78.5523963	-0.3496251	PICHINCHA
30/12/2020	000014971	8	-78.478542	-0.294665	PICHINCHA
30/12/2020	000014971	2	-78.478542	-0.294665	PICHINCHA
30/12/2020	000014884	1	-78.537162	-0.261855	PICHINCHA
30/12/2020	000014902	1	-78.53354112	-0.340115544	PICHINCHA
30/12/2020	000014913	2	-78.5482132	-0.3629643	PICHINCHA
30/12/2020	000014986	5	-78.4443176	-0.3283781	PICHINCHA
30/12/2020	000014986	2	-78.4443176	-0.3283781	PICHINCHA
30/12/2020	000014965	3	-78.43400057	-0.336507470	PICHINCHA
30/12/2020	000014965	3	-78.43400057	-0.336507470	PICHINCHA
30/12/2020	000014977	1	-78.4524325467646	-0.309064616449177	PICHINCHA
30/12/2020	000014958	5	-78.5466708	-0.4022138	PICHINCHA
30/12/2020	000014906	1	-78.54863726	-0.343684474	PICHINCHA
30/12/2020	000014924	4	-78.4781608	-0.1864121	PICHINCHA
30/12/2020	000014964	1	-78.432327	-0.341713	PICHINCHA
30/12/2020	000014964	1	-78.432327	-0.341713	PICHINCHA
30/12/2020	000014946	1	-78	-0	PICHINCHA
30/12/2020	000014946	1	-78	-0	PICHINCHA
30/12/2020	000014974	3	-78.45714652	-0.3062949	PICHINCHA
30/12/2020	000014887	1	-78.540177	-0.337082	PICHINCHA
30/12/2020	000014883	5	-78.5414828	-0.2611603	PICHINCHA
30/12/2020	000014945	2	-78.4351792	-0.076412	PICHINCHA

30/12/2020	000014973	10	-78.4659302607179	-0.303165642544627	PICHINCHA
30/12/2020	000014963	3	-78.4326160512865	-0.346970474347472	PICHINCHA
30/12/2020	000014967	6	-78.449593	-0.303276	PICHINCHA
30/12/2020	000014967	1	-78.449593	-0.303276	PICHINCHA
30/12/2020	000014942	1	-78.419853	-0.088291	PICHINCHA
30/12/2020	000014935	10	-78.420065	-0.105815	PICHINCHA
30/12/2020	000014935	1	-78.420065	-0.105815	PICHINCHA
30/12/2020	000014987	1	-78.4466664	-0.3286307	PICHINCHA
30/12/2020	000014968	1	-78.4576037	-0.3043668	PICHINCHA
30/12/2020	000014968	1	-78.4576037	-0.3043668	PICHINCHA
30/12/2020	000014899	7	-78.536354	-0.317232	PICHINCHA
30/12/2020	000014914	1	-78.54904	-0.36362	PICHINCHA
30/12/2020	000014914	2	-78.54904	-0.36362	PICHINCHA
30/12/2020	000014949	30	-78.439909	-0.093139	PICHINCHA
30/12/2020	000014949	1	-78.439909	-0.093139	PICHINCHA
30/12/2020	000014898	1	-78.547571	-0.3323171	PICHINCHA
30/12/2020	000014907	10	-78.5516237	-0.3502578	PICHINCHA
30/12/2020	000014907	1	-78.5516237	-0.3502578	PICHINCHA
30/12/2020	000014907	2	-78.5516237	-0.3502578	PICHINCHA
30/12/2020	000014957	3	-78.5093729	-0.1414608	PICHINCHA
30/12/2020	000014912	1	-78.548318	-0.362102	PICHINCHA
30/12/2020	000014912	1	-78.548318	-0.362102	PICHINCHA
30/12/2020	000014956	7	-78.4862128	-0.1355464	PICHINCHA
30/12/2020	000014976	1	-78.4531035181135	-0.308105139993131	PICHINCHA
30/12/2020	000014940	1	-78.4242104	-0.0918987	PICHINCHA
30/12/2020	000014940	1	-78.4242104	-0.0918987	PICHINCHA
30/12/2020	000014980	1	-78.4492084477097	-0.324285440146923	PICHINCHA
30/12/2020	000014947	1	-78.4413652	-0.0911175	PICHINCHA
29/12/2020	000014852	1	-78.4845708590001	-0.195763916708529	PICHINCHA
29/12/2020	000014852	1	-78.4845708590001	-0.195763916708529	PICHINCHA
29/12/2020	000014852	2	-78.4845708590001	-0.195763916708529	PICHINCHA
29/12/2020	000014861	3	-78.4809768	-0.1850295	PICHINCHA
29/12/2020	000014803	16	-78.5477779	-0.2803628	PICHINCHA
29/12/2020	000014803	4	-78.5477779	-0.2803628	PICHINCHA
29/12/2020	000014828	2	-78.54436378	-0.26482873	PICHINCHA
29/12/2020	000014843	2	-78.486438	-0.175802	PICHINCHA
29/12/2020	000014850	1	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014850	1	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014850	2	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014850	1	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014850	1	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014850	3	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014850	1	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014850	2	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014850	2	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014850	1	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014850	2	-78.476217	-0.170734	PICHINCHA
29/12/2020	000014859	2	-78.4772469	-0.1819162	PICHINCHA
29/12/2020	000014810	3	-78.5517120826989	-0.261301137506962	PICHINCHA
29/12/2020	000014814	2	-78.54508105	-0.26113294	PICHINCHA
29/12/2020	000014848	5	-78.4582889	-0.1547137	PICHINCHA
29/12/2020	000014868	100	-78.566591	-0.51121	PICHINCHA
29/12/2020	000014868	25	-78.566591	-0.51121	PICHINCHA

29/12/2020	000014797	1	-78.566591	-0.51121	PICHINCHA
29/12/2020	000014818	1	-78.54237455	-0.26253398	PICHINCHA
29/12/2020	000014820	2	-78.54393129	-0.26101511	PICHINCHA
29/12/2020	000014820	1	-78.54393129	-0.26101511	PICHINCHA
29/12/2020	000014801	2	-78.5462053	-0.295636	PICHINCHA
29/12/2020	000014804	40	-78.5131812	-0.2274107	PICHINCHA
29/12/2020	000014804	10	-78.5131812	-0.2274107	PICHINCHA
29/12/2020	000014804	1	-78.5131812	-0.2274107	PICHINCHA
29/12/2020	000014804	3	-78.5131812	-0.2274107	PICHINCHA
29/12/2020	000014804	1	-78.5131812	-0.2274107	PICHINCHA
29/12/2020	000014844	7	-78.4741212241352	0.0882095564156771	PICHINCHA
29/12/2020	000014822	1	-78.53817	-0.258749	PICHINCHA
29/12/2020	000014813	1	-78.5458759	-0.2639015	PICHINCHA
29/12/2020	000014813	2	-78.5458759	-0.2639015	PICHINCHA
29/12/2020	000014881	50	-78.4582726	-0.3515073	PICHINCHA
29/12/2020	000014821	1	-78.5409355	-0.256122	PICHINCHA
29/12/2020	000014872	4	-78.51776048	-0.2419838	PICHINCHA
29/12/2020	000014872	1	-78.51776048	-0.2419838	PICHINCHA
29/12/2020	000014827	5	-78.5408857	-0.2630201	PICHINCHA
29/12/2020	000014855	1	-78.4798275	-0.1790557	PICHINCHA
29/12/2020	000014857	4	-78.4800861	-0.1802057	PICHINCHA
29/12/2020	000014857	2	-78.4800861	-0.1802057	PICHINCHA
29/12/2020	000014798	10	-78.5609553	-0.3131914	PICHINCHA
29/12/2020	000014869	50	-78.503784	-0.2540165	PICHINCHA
29/12/2020	000014876	1	-78.4827993	-0.203026	PICHINCHA
29/12/2020	000014864	1	-78.4782319	-0.1883326	PICHINCHA
29/12/2020	000014880	48	-78.2676483	0.2289627	PICHINCHA
29/12/2020	000014880	12	-78.2676483	0.2289627	PICHINCHA
29/12/2020	000014880	620	-78.2676483	0.2289627	PICHINCHA
29/12/2020	000014880	20	-78.2676483	0.2289627	PICHINCHA
29/12/2020	000014880	20	-78.2676483	0.2289627	PICHINCHA
29/12/2020	000014817	2	-78.54339489	-0.2618099	PICHINCHA
29/12/2020	000014816	2	-78.544522	-0.261452	PICHINCHA
29/12/2020	000014811	2	-78.54619024	-0.26351838	PICHINCHA
29/12/2020	000014811	2	-78.54619024	-0.26351838	PICHINCHA
29/12/2020	000014812	2	-78.5463318	-0.2641682	PICHINCHA
29/12/2020	000014854	1	-78.4879059	-0.1990129	PICHINCHA
29/12/2020	000014856	8	-78.4783874	-0.1786201	PICHINCHA
29/12/2020	000014856	2	-78.4783874	-0.1786201	PICHINCHA
29/12/2020	000014870	3	-78.4806861821562	-0.170048447325826	PICHINCHA
29/12/2020	000014870	4	-78.4806861821562	-0.170048447325826	PICHINCHA
29/12/2020	000014875	1	-78.5026804	-0.1956012	PICHINCHA
29/12/2020	000014873	2	-78.51324397	-0.21556114	PICHINCHA
29/12/2020	000014879	1	-78.5115177	-0.1766723	PICHINCHA
29/12/2020	000014800	3	-78.548893	-0.298033	PICHINCHA
29/12/2020	000014802	1	-78.5440804436803	-0.295034064911306	PICHINCHA
29/12/2020	000014802	1	-78.5440804436803	-0.295034064911306	PICHINCHA
29/12/2020	000014833	4	-78.54747368	-0.27246838	PICHINCHA
29/12/2020	000014833	1	-78.54747368	-0.27246838	PICHINCHA
29/12/2020	000014874	19	-78.52335166	-0.211311666	PICHINCHA
29/12/2020	000014874	1	-78.52335166	-0.211311666	PICHINCHA
29/12/2020	000014815	2	-78.544608	-0.261366	PICHINCHA
29/12/2020	000014853	1	-78.4840478	-0.1750637	PICHINCHA
29/12/2020	000014853	1	-78.4840478	-0.1750637	PICHINCHA
29/12/2020	000014819	2	-78.5423571	-0.2624342	PICHINCHA
29/12/2020	000014834	2	-78.54271	-0.272299	PICHINCHA

29/12/2020	000014805	3	-78.5608605109155	-0.258864560164511	PICHINCHA
29/12/2020	000014809	3	-78.55145915	-0.25957169	PICHINCHA
29/12/2020	000014824	1	-78.5300088	-0.2588182	PICHINCHA
29/12/2020	000014862	2	-78.481412	-0.1853394	PICHINCHA
29/12/2020	000014877	1	-78.4804393	-0.2019052	PICHINCHA
29/12/2020	000014877	1	-78.4804393	-0.2019052	PICHINCHA
29/12/2020	000014836	20	-78.5407078	-0.2796493	PICHINCHA
29/12/2020	000014846	7	-78.487074	-0.127133	PICHINCHA
29/12/2020	000014846	1	-78.487074	-0.127133	PICHINCHA
29/12/2020	000014838	1	-78.5415855	-0.2837442	PICHINCHA
29/12/2020	000014838	1	-78.5415855	-0.2837442	PICHINCHA
29/12/2020	000014806	3	-78.5532932	-0.2533211	PICHINCHA
29/12/2020	000014858	1	-78.481381	-0.179878	PICHINCHA
29/12/2020	000014867	1	-78.47145334	-0.190265807	PICHINCHA
29/12/2020	000014867	1	-78.47145334	-0.190265807	PICHINCHA
29/12/2020	000014847	1	-78.4865749	-0.1307724	PICHINCHA
29/12/2020	000014847	20	-78.4865749	-0.1307724	PICHINCHA
29/12/2020	000014823	3	-78.537977	-0.2594768	PICHINCHA
29/12/2020	000014837	1	-78.5408333	-0.2803984	PICHINCHA
29/12/2020	000014841	6	-78.4831974	-0.1914544	PICHINCHA
29/12/2020	000014799	20	-78.51454295	-0.2489255	PICHINCHA
29/12/2020	000014799	3	-78.51454295	-0.2489255	PICHINCHA
29/12/2020	000014799	3	-78.51454295	-0.2489255	PICHINCHA
29/12/2020	000014835	1	-78.53835912	-0.272703878	PICHINCHA
29/12/2020	000014829	1	-78.544063	-0.26969	PICHINCHA
29/12/2020	000014878	1	-78.5087192	-0.1782077	PICHINCHA
29/12/2020	000014878	1	-78.5087192	-0.1782077	PICHINCHA
29/12/2020	000014860	4	-78.4781608	-0.1864121	PICHINCHA
29/12/2020	000014866	8	-78.471766	-0.189614	PICHINCHA
29/12/2020	000014866	2	-78.471766	-0.189614	PICHINCHA
29/12/2020	000014845	10	-78.460609	-0.098017	PICHINCHA
29/12/2020	000014845	1	-78.460609	-0.098017	PICHINCHA
29/12/2020	000014807	4	-78.5520471	-0.257144	PICHINCHA
29/12/2020	000014849	3	-78.4852985817699	-0.192728713599302	PICHINCHA
29/12/2020	000014871	4	-78.51908537	-0.24585354	PICHINCHA
29/12/2020	000014871	1	-78.51908537	-0.24585354	PICHINCHA
29/12/2020	000014839	1	-78.5525428	-0.2586231	PICHINCHA
29/12/2020	000014840	1	-78.5525428	-0.2586231	PICHINCHA
29/12/2020	000014842	7	-78.4871411	-0.1800731	PICHINCHA
29/12/2020	000014842	1	-78.4871411	-0.1800731	PICHINCHA
29/12/2020	000014842	1	-78.4871411	-0.1800731	PICHINCHA
29/12/2020	000014830	1	-78.54686667	-0.26869465	PICHINCHA
29/12/2020	000014830	2	-78.54686667	-0.26869465	PICHINCHA
29/12/2020	000014830	3	-78.54686667	-0.26869465	PICHINCHA
29/12/2020	000014830	1	-78.54686667	-0.26869465	PICHINCHA
29/12/2020	000014830	4	-78.54686667	-0.26869465	PICHINCHA
29/12/2020	000014830	3	-78.54686667	-0.26869465	PICHINCHA
29/12/2020	000014831	3	-78.5492788	-0.2705434	PICHINCHA
29/12/2020	000014832	3	-78.5492788	-0.2705434	PICHINCHA
29/12/2020	000014851	1	-78.4800523	-0.1756575	PICHINCHA
29/12/2020	000014851	1	-78.4800523	-0.1756575	PICHINCHA
29/12/2020	000014865	1	-78.4720277	-0.1897131	PICHINCHA
29/12/2020	000014865	1	-78.4720277	-0.1897131	PICHINCHA
29/12/2020	000014865	1	-78.4720277	-0.1897131	PICHINCHA
29/12/2020	000014865	1	-78.4720277	-0.1897131	PICHINCHA
29/12/2020	000014826	7	-78.5404904	-0.2629443	PICHINCHA
29/12/2020	000014863	1	-78.4783095	-0.188676	PICHINCHA
29/12/2020	000014808	7	-78.552056	-0.259621	PICHINCHA
29/12/2020	000014808	1	-78.552056	-0.259621	PICHINCHA

29/12/2020	000014825	4	-78.5374002	-0.2621112	PICHINCHA
28/12/2020	000014775	3	-78.4778584	-0.1864965	PICHINCHA
28/12/2020	000014775	1	-78.4778584	-0.1864965	PICHINCHA
28/12/2020	000014775	3	-78.4778584	-0.1864965	PICHINCHA
28/12/2020	000014775	1	-78.4778584	-0.1864965	PICHINCHA
28/12/2020	000014747	1	-78.5152448	-0.2113007	PICHINCHA
28/12/2020	000014790	1	-78.490469	-0.148718	PICHINCHA
28/12/2020	000014790	5	-78.490469	-0.148718	PICHINCHA
28/12/2020	000014790	1	-78.490469	-0.148718	PICHINCHA
28/12/2020	000014790	1	-78.490469	-0.148718	PICHINCHA
28/12/2020	000014790	1	-78.490469	-0.148718	PICHINCHA
28/12/2020	000014742	20	-78.5477779	-0.2803628	PICHINCHA
28/12/2020	000014778	3	-78.4792791	-0.2795883	PICHINCHA
28/12/2020	000014778	2	-78.4792791	-0.2795883	PICHINCHA
28/12/2020	000014776	1	-78.4828027	-0.187981	PICHINCHA
28/12/2020	000014776	1	-78.4828027	-0.187981	PICHINCHA
28/12/2020	000014749	1	-78.5078255	-0.2269655	PICHINCHA
28/12/2020	000014782	1	-78.49510816	-0.2162663	PICHINCHA
28/12/2020	000014782	1	-78.49510816	-0.2162663	PICHINCHA
28/12/2020	000014768	10	-78.4527289308608	-0.119471079669893	PICHINCHA
28/12/2020	000014751	1	-78.5093574	-0.2426596	PICHINCHA
28/12/2020	000014757	12	-78.523683	-0.23962604	PICHINCHA
28/12/2020	000014757	3	-78.523683	-0.23962604	PICHINCHA
28/12/2020	000014784	0,50	-78.49577142	-0.213122586	PICHINCHA
28/12/2020	000014779	0,25	-78.48548574	-0.235093524	PICHINCHA
28/12/2020	000014779	0,25	-78.48548574	-0.235093524	PICHINCHA
28/12/2020	000014779	1	-78.48548574	-0.235093524	PICHINCHA
28/12/2020	000014756	1	-78.5213502	-0.2424882	PICHINCHA
28/12/2020	000014770	7	-78.459884	-0.144861	PICHINCHA
28/12/2020	000014764	3	-78.48389821	-0.166800417	PICHINCHA
28/12/2020	000014785	1	-78.50546325	-0.209479099	PICHINCHA
28/12/2020	000014786	1	-78.50546325	-0.209479099	PICHINCHA
28/12/2020	000014767	3	-78.4345158	-0.1050698	PICHINCHA
28/12/2020	000014767	0,25	-78.4345158	-0.1050698	PICHINCHA
28/12/2020	000014759	1	0	0	PICHINCHA
28/12/2020	000014744	0,25	-78.52341427	-0.211109672	PICHINCHA
28/12/2020	000014744	1	-78.52341427	-0.211109672	PICHINCHA
28/12/2020	000014744	3	-78.52341427	-0.211109672	PICHINCHA
28/12/2020	000014745	10	-78.52341427	-0.211109672	PICHINCHA
28/12/2020	000014758	5	-78.52096483	-0.246123019	PICHINCHA
28/12/2020	000014780	0,25	-78.4831110667437	-0.231818631291389	PICHINCHA
28/12/2020	000014780	0,25	-78.4831110667437	-0.231818631291389	PICHINCHA
28/12/2020	000014780	1	-78.4831110667437	-0.231818631291389	PICHINCHA
28/12/2020	000014781	0,25	-78.4917777	-0.2266927	PICHINCHA
28/12/2020	000014781	0,25	-78.4917777	-0.2266927	PICHINCHA
28/12/2020	000014781	1	-78.4917777	-0.2266927	PICHINCHA
28/12/2020	000014771	7	-78.4659527242184	-0.165277845226228	PICHINCHA
28/12/2020	000014771	1	-78.4659527242184	-0.165277845226228	PICHINCHA
28/12/2020	000014793	4	-78.4794442355633	-0.33371755387634	PICHINCHA
28/12/2020	000014748	1	-78.511186	-0.218397	PICHINCHA
28/12/2020	000014753	16	-78.517744	-0.243558	PICHINCHA
28/12/2020	000014753	4	-78.517744	-0.243558	PICHINCHA
28/12/2020	000014788	1	-78.500796	-0.206237	PICHINCHA
28/12/2020	000014789	1	-78.500796	-0.206237	PICHINCHA
28/12/2020	000014789	2	-78.500796	-0.206237	PICHINCHA

28/12/2020	000014763	3	-78.4855408128351	-0.208297963254154	PICHINCHA
28/12/2020	000014763	1	-78.4855408128351	-0.208297963254154	PICHINCHA
28/12/2020	000014763	1	-78.4855408128351	-0.208297963254154	PICHINCHA
28/12/2020	000014777	10	-78.4840255	-0.1957677	PICHINCHA
28/12/2020	000014777	2	-78.4840255	-0.1957677	PICHINCHA
28/12/2020	000014777	2	-78.4840255	-0.1957677	PICHINCHA
28/12/2020	000014773	2	-78.4801055	-0.1810592	PICHINCHA
28/12/2020	000014773	1	-78.4801055	-0.1810592	PICHINCHA
28/12/2020	000014773	1	-78.4801055	-0.1810592	PICHINCHA
28/12/2020	000014752	3	-78.511713	-0.24616	PICHINCHA
28/12/2020	000014752	1	-78.511713	-0.24616	PICHINCHA
28/12/2020	000014752	1	-78.511713	-0.24616	PICHINCHA
28/12/2020	000014772	10	-78.48232622	-0.174343524	PICHINCHA
28/12/2020	000014791	3	-78.4535085	-0.3073028	PICHINCHA
28/12/2020	000014765	2	-78.4849619	-0.3377483	PICHINCHA
28/12/2020	000014765	1	-78.4849619	-0.3377483	PICHINCHA
28/12/2020	000014765	1	-78.4849619	-0.3377483	PICHINCHA
28/12/2020	000014774	2	-78.4801059	-0.1813025	PICHINCHA
28/12/2020	000014769	3	-78.46373529	-0.1414976	PICHINCHA
28/12/2020	000014769	3	-78.46373529	-0.1414976	PICHINCHA
28/12/2020	000014766	10	-78.4166197758168	0.0744342757388949	PICHINCHA
28/12/2020	000014766	1	-78.4166197758168	0.0744342757388949	PICHINCHA
28/12/2020	000014766	3	-78.416619758168	-0.074434257388949	PICHINCHA
28/12/2020	000014755	4	-78.51908537	-0.24585354	PICHINCHA
28/12/2020	000014755	1	-78.51908537	-0.24585354	PICHINCHA
28/12/2020	000014754	12	-78.51792404	-0.244660209	PICHINCHA
28/12/2020	000014754	3	-78.51792404	-0.244660209	PICHINCHA
28/12/2020	000014754	0,25	-78.51792404	-0.244660209	PICHINCHA
28/12/2020	000014743	1	-78.549780083892	-0.280283424071968	PICHINCHA
28/12/2020	000014743	1	-78.549780083892	-0.280283424071968	PICHINCHA
28/12/2020	000014740	5	-78.5498133	-0.2979417	PICHINCHA
28/12/2020	000014762	10	-78.420065	-0.105815	PICHINCHA
28/12/2020	000014762	1	-78.420065	-0.105815	PICHINCHA
28/12/2020	000014762	1	-78.420065	-0.105815	PICHINCHA
28/12/2020	000014783	1	-78.496449598129	-0.213091112673283	PICHINCHA
28/12/2020	000014746	7	-78.5221899	-0.2099823	PICHINCHA
28/12/2020	000014741	10	-78.5523201	-0.3496635	PICHINCHA
28/12/2020	000014750	3	-78.5081943	-0.2374153	PICHINCHA
28/12/2020	000014792	10	-78.44991026	-0.327137969	PICHINCHA
28/12/2020	000014792	1	-78.44991026	-0.327137969	PICHINCHA
28/12/2020	000014787	20	-78.507995	-0.207467	PICHINCHA
28/12/2020	000014787	2	-78.507995	-0.207467	PICHINCHA
26/12/2020	000014735	10	-78.4870917	-0.1317742	PICHINCHA
26/12/2020	000014715	2	-78.547533	-0.308061	PICHINCHA
26/12/2020	000014723	3	-78.5300581	-0.2485776	PICHINCHA
26/12/2020	000014728	20	-78.5501395	-0.4048824	PICHINCHA
26/12/2020	000014721	1	-78.5213502	-0.2424882	PICHINCHA
26/12/2020	000014738	140	-78.4821564	-0.1842761	PICHINCHA
26/12/2020	000014738	114	-78.4821564	-0.1842761	PICHINCHA
26/12/2020	000014738	200	-78.4821564	-0.1842761	PICHINCHA
26/12/2020	000014738	50	-78.4821564	-0.1842761	PICHINCHA
26/12/2020	000014738	100	-78.4821564	-0.1842761	PICHINCHA
26/12/2020	000014738	50	-78.4821564	-0.1842761	PICHINCHA
26/12/2020	000014738	50	-78.4821564	-0.1842761	PICHINCHA
26/12/2020	000014738	50	-78.4821564	-0.1842761	PICHINCHA
26/12/2020	000014716	8	-78.54489667	-0.246378709	PICHINCHA

26/12/2020	000014716	2	-78.54489667	-0.246378709	PICHINCHA
26/12/2020	000014737	8	-78.4837407	-0.1695983	PICHINCHA
26/12/2020	000014737	2	-78.4837407	-0.1695983	PICHINCHA
26/12/2020	000014731	4	-78.4847082	-0.1941034	PICHINCHA
26/12/2020	000014714	7	-78.5492471	-0.3283748	PICHINCHA
26/12/2020	000014725	10	-78.509913	-0.218129	PICHINCHA
26/12/2020	000014725	32	-78.509913	-0.218129	PICHINCHA
26/12/2020	000014725	8	-78.509913	-0.218129	PICHINCHA
26/12/2020	000014724	2	-78.452553832035	-0.31056405510734	PICHINCHA
26/12/2020	000014724	3	-78.452553832935	-0.310564055107534	PICHINCHA
26/12/2020	000014722	5	-78.5252031	-0.2500427	PICHINCHA
26/12/2020	000014722	2	-78.5252031	-0.2500427	PICHINCHA
26/12/2020	000014712	250	-78.52335166	-0.211311666	PICHINCHA
26/12/2020	000014733	11	-78.4767928	-0.1280004	PICHINCHA
26/12/2020	000014733	20	-78.4767928	-0.1280004	PICHINCHA
26/12/2020	000014717	3	-78.5461543	-0.239889	PICHINCHA
26/12/2020	000014727	20	-78.5221716	-0.3853403	PICHINCHA
26/12/2020	000014732	1	-78.442875776552	-0.089319016641128	PICHINCHA
26/12/2020	000014718	1	-78.542165	-0.2401421	PICHINCHA
26/12/2020	000014713	3	-78.5500743	-0.3292694	PICHINCHA
26/12/2020	000014713	1	-78.5500743	-0.3292694	PICHINCHA
26/12/2020	000014720	1	-78.520665	-0.241791	PICHINCHA
26/12/2020	000014736	15	-78.483896	-0.1671485	PICHINCHA
26/12/2020	000014726	2	-78.471381012303	-0.328410384245217	PICHINCHA
26/12/2020	000014729	3	-78.4142717	-0.2112724	PICHINCHA
26/12/2020	000014734	7	-78.4791173	-0.1281202	PICHINCHA
26/12/2020	000014719	3	-78.5414356	-0.2398397	PICHINCHA

Programa

```
1 import numpy as np
2 import math as m
3 import random as rn
4
5 import dataSet as ds
6 import Kmedoids as km
7 import AG as ag
8 import Mapping as mp
9 import cv2 as cv
10
11 import time
12
13 '''Programa Principal'''
14 starttime.time()
15 archivos=['dat1','dat2','dat3','dat4','dat5']
16 #archivos=['dat1','dat2','dat3','dat4','dat5']
17 Ngen=10
18 N=10
19 pc=0.5
20 pm=0.2
21 Tmax=8
22 CAP=200
23 tC=0.5
24 tD=0.017
25 color=[(255,0,0),(255,0,255),(0,0,255)]
26
27
28 #doc = open('resultados.csv', 'w')
29 #for arch in range(5):
30 coordenadas, demandas = ds.read(archivo)
31
32 T_D=ds.N(coordenadas)
```

```
1 import ...
2
3 '''Procesamiento de datos'''
4
5 def read(name):...
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

```
1 import random as rn
2 import math as m
3 import numpy as np
4
5 '''Algoritmo Genetico'''
6
7 def randSolution(D):...
8
9 def VHC(D):...
10
11 def IniPob(N, D):...
12
13 def fitness(P, D, T, Tmax, demd, CAP, tC, tD):...
14
15 def elitism(P, D, T, Tmax, demd, CAP, tC, tD):...
16
17 def bestSolution(P, D, T, Tmax, demd, CAP, tC, tD):...
18
19 def selection(poblacion, D, pc, pm, T, Tmax, demd, CAP, tC, tD):...
20
21 def crossover(padre, madre, pm):...
22
23 def mutation(hijo, pm):...
24
25 def cost(S, D, T, Tmax, demd, CAP, tC, tD):...
26
27 def algoritmoGenetico(Ngen, N, pc, pm, D, T, Tmax, demd, CAP, tC, tD):...
28
29 def presentacion(S, demd, CAP):...
30
31 algoritmoGenetico()
```

```
1 import numpy as np
2 import math as m
3 import random as rn
4
5 '''Algoritmo k-medoides'''
6
7 def kmedoids(X, demandas, cap):...
8
9 def Kmediods(coordenadas, demandas, T):...
10
11 def Kmediods2(coordenadas, demandas, T):...
12
13 def Kmediods3(coordenadas, demandas, T):...
```