Entregable 3: Solución de la Problemática

Integrantes:

Melanie Astrid Montaño Ramos	**A01643711**
Karla Rebeca Munguía Romero	**A01741255**
Gabriel Eduardo Meléndez Zavala	**A01638293**
Guillermo Villegas Morales	**A01637169**

Optimización determinista

Grupo 502

Profesores:
Dr. Abelardo Ernesto Damy Solís
MSc. Ramiro Zermeño Díaz

Fecha de entrega:
16 de junio de 2024

Planteamiento del problema

Para este reto se nos planteó el problema de que se tienen diferentes tiendas a las que la marca Come Verde tiene que brindarles servicio de promotoría, y que, por lo tanto, tienen que ser visitadas con cierta frecuencia y en ciertos horarios. Lo que se busca es generar las rutas más adecuadas para cada uno de los 11 promotores para realizar en el mes.

Como consideraciones se plantearon las siguientes:

- Se tienen 11 promotores que duran aproximadamente 40 minutos por tienda.
- Cada promotor debe empezar su ruta en una sucursal de Walmart, Chedraui o La Comer.
- Se estandarizó que el traslado de los promotores entre tienda y tienda es en tiempos de traslado por transporte público.
- Se descartaron las sucursales de Farmacias del Ahorro y Farmacias Benavides a falta de información proporcionada.
- Las tiendas se visitan por mes en la siguiente frecuencia:
    - Nutrisa: 1 vez
    - Walmart: 4-8 veces
    - Chedraui: 4-8 veces
    - La Comer: 4 veces

Formulación matemática del modelo

El algoritmo de Clarke-Wright, también conocido como el algoritmo de ahorro de Clarke-Wright, es una heurística utilizada para resolver el problema del enrutamiento de vehículos (VRP, por sus siglas en inglés). Este algoritmo busca minimizar la distancia total recorrida por un grupo de vehículos que deben visitar un conjunto de clientes, comenzando y terminando en un depósito.

El ahorro para combinar la ruta de dos clientes i y j es $S_{ij} = C_{0i} + C_{0j} - C_{ij}$, donde C_{0i} es la distancia del depósito al cliente i, C_{ij} es la distancia entre los clientes i y j, y S_{ij} es la distancia ahorrada del nodo i al nodo j (Tunnisaki & Sutarman, 2023). **Condición:** Un par de nodos i y j se pueden unir si ambos están disponibles y no violan las restricciones de capacidad del vehículo.

Procedimiento del Algoritmo:

1. Comenzar con una solución inicial donde cada cliente es servido por un vehículo individualmente, formando una ruta (0−i−0) para cada cliente iii.
2. Calcular los ahorros Sij para todos los pares (i,j).
3. Ordenar los ahorros Sij en orden descendente.
4. Iterar sobre la lista de ahorros ordenada y combinar rutas donde sea posible, aplicando los ahorros siempre que no se violen las restricciones de capacidad y los clientes no se visiten más de una vez.

Metodología de solución

1. Conectar las bases de datos. Se unieron las bases de datos que se nos proporcionaron a partir del DET de las tiendas indicadas en la base de datos de “TIENDAS CDMX TEC”, con su dirección y la frecuencia con que se visitan por mes.
2. Mapear las tiendas. Una vez relacionadas las tiendas a trabajar con su dirección, se ingresaron en un documento de Google Maps. Al hacerlo, se pudieron obtener las coordenadas geográficas de cada establecimiento, las cuales resultan más sencillas de manipular en código.
3. obtener pares de coordenadas
4. Clustering. Aquí podemos observar que el número de elementos por grupo no es consistente, esto debido a que existen muchas tiendas concentradas y grupos de menos tiendas más alejados del centro. Como necesitamos que el número de elementos por grupo sea 25, recurriremos a un método especial de K-Means. Para poder utilizar el algoritmo de agrupamiento fue necesario considerar la distancia euclidiana entre tiendas
5. Google Maps API. Para poder utilizar la API que calcula la distancia/tiempo entre tiendas tuvimos que poner como restricción que cada cluster estuviera compuesto de un máximo de 25 tiendas, el número máximo que la API era capaz de procesar.
6. Matriz de distancias. Se construyó la matriz de distancias
7. Clark-Wright. Se aplicó el algoritmo de ahorros de Clark-Wright para crear las rutas.
8. Distribución de rutas. Se distribuyeron las rutas generadas en una propuesta semanal para cada promotor.

Solución del problema

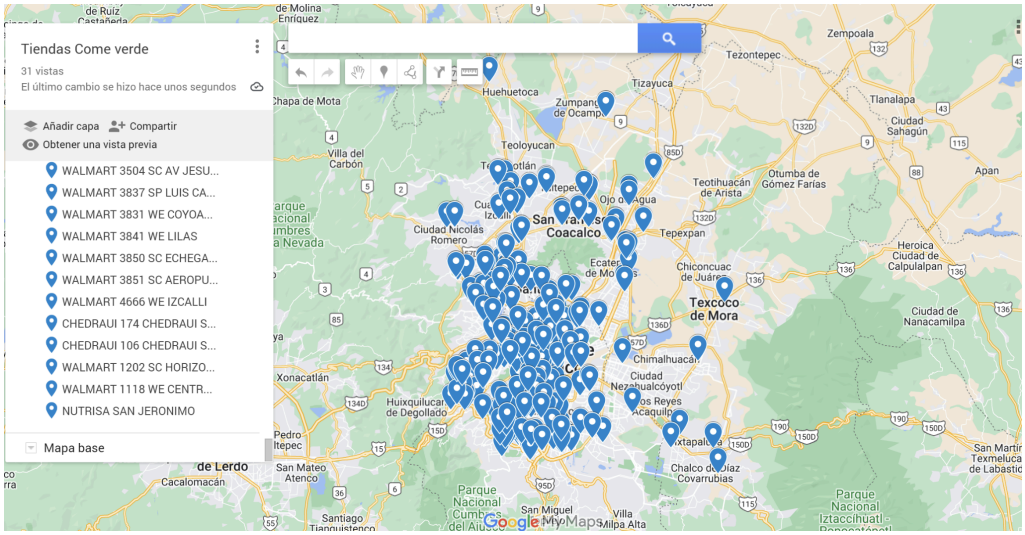


Figura 1. Tiendas a visitar

Gráficas

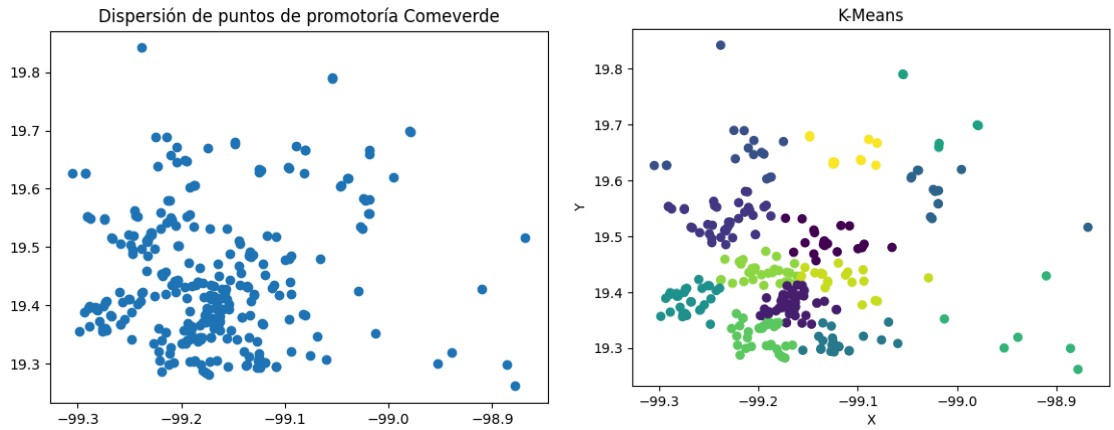


Figura 2. Resultados del K-means

Tablas

Tabla de frecuencia .XLSX

Archivo Editar Ver Insertar Formato Datos Herramientas Ayuda

Tienda	ID	Long	Lat	Frecuencia
LA COMER SUMESA CENTENARIO	1	-99.163	19.35766	3
NUTRISA PORTAL SAN ANGEL	2	-99.1898	19.36109	1
NUTRISA CHED PICACHO	3	-99.2185	19.28717	1
NUTRISA SATELITE 2	4	-99.2339	19.5104	1
NUTRISA CHED POLANCO	5	-99.2064	19.44218	1
NUTRISA PLAZA LORETO	6	-99.1945	19.33987	1
NUTRISA PLAZA LINDAVISTA	7	-99.1336	19.49154	1
NUTRISA ENCUENTRO TLANEPANTLA	8	-99.1728	19.53216	1
NUTRISA UNIVERSIDAD	9	-99.1815	19.34183	1
NUTRISA ASTURIAS MEGA COM MEX	10	-99.1329	19.4076	1
LA COMER??305 CITY MARKET INTERLOMAS	11	-99.2722	19.40665	3
LA COMER??55 LA COMER VILLA COAPA	12	-99.126	19.30499	3
NUTRISA??TOWN CENTER EL ROSARIO	13	-99.2031	19.50309	1
NUTRISA ANTARA	14	-99.2047	19.4391	1
NUTRISA LAS AMERICAS ECATEPEC	15	-99.024	19.58353	1
NUTRISA PATIO UNIVERSIDAD	16	-99.1671	19.36571	1
CHEDRAUI 814 SUPER CHEDRAUI SELECTO WTC 10-22 SEARS	17	-99.173	19.39386	3
LA COMER 127 LA COMER TULTITLAN	18	-99.0804	19.66654	3

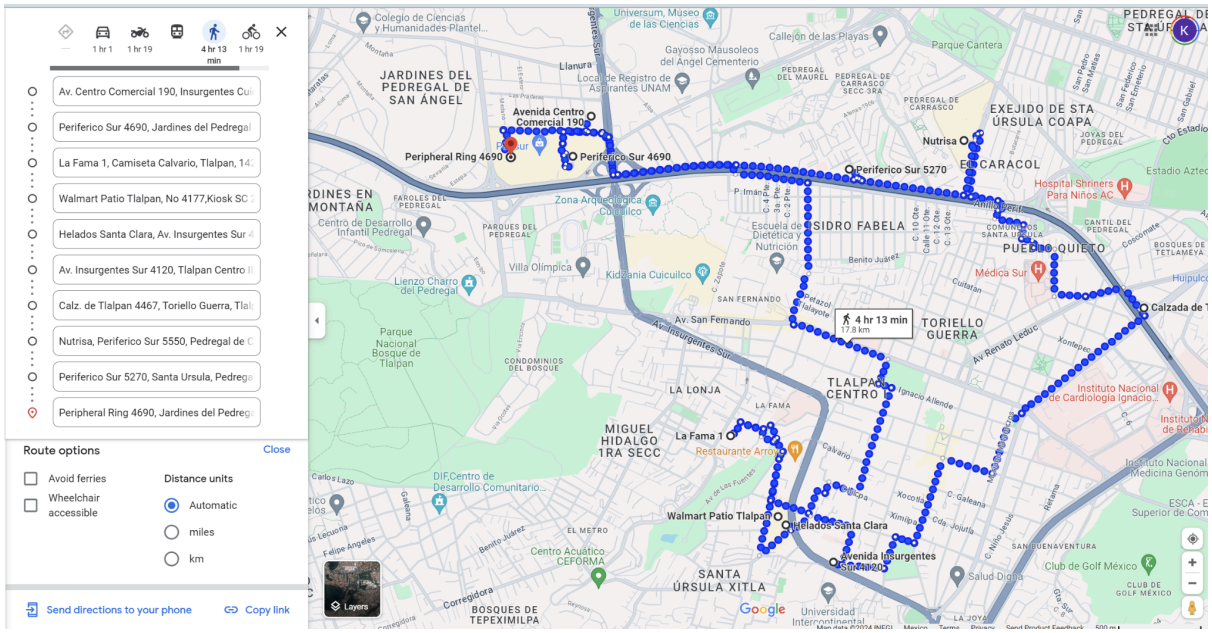
Id	*Frec. mensual*
1	Una vez
2	Dos veces
3	Cuatro veces
4	Ocho veces

Figura 3. Tabla de relación de posición y frecuencia de cada tienda

cluster_12_duration.csv

A	B	C	D	E	F	G
0	1	2	3	4	5	6
0	3943	4562	2967	2312	2755	2427
3119	0	5607	1798	1874	3800	2702
6067	6953	0	7258	5696	3404	7398
3158	1512	5646	0	2547	3839	2741
2686	2069	4766	3353	0	3136	3065
3390	4368	2992	4581	3111	0	3844
3024	4314	5727	3245	3264	3920	0

Figura 4. Matriz de duración de traslado entre tiendas



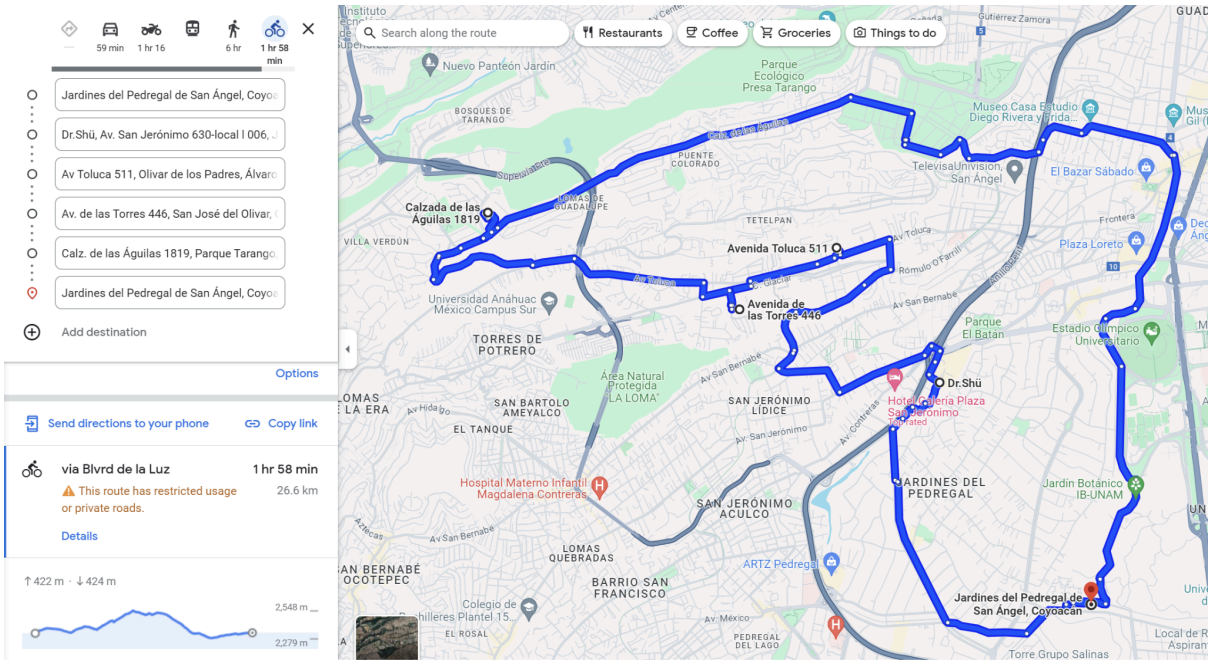


Figura 6. Ejemplos de rutas generadas por el algoritmo

Interpretación de la solución

La implementación del código arrojó 70 rutas cuyas consideraciones de tiempo de jornada laboral fueron pre consideradas en la implementación del algoritmo, sin embargo, para los casos en los que las rutas generadas fueron muy pequeñas por el esfuerzo del algoritmo por incluir cada una de las tiendas, aunque al final resultaran inconvenientes en el proceso de resolución, se decidió juntarlas con otras dentro de su mismo cluster para que pudieran valer la pena para cada promotor.

CLUSTER 1						
1.1	303	260	241	245		
1.2	88	179	112	191	83	
1.3	185	262	293	236	113	225
1.4	281	72	290	2	237	
1.5	267	71				
1.6	286	5				

CLUSTER 2					
2.1	164	70			
2.2	250	138	17		
2.3	91	261	98	59	
2.4	4	246	204	60	
2.5	198	97			
2.6	155	109	186	89	
2.7	118	154	56	190	
2.8	187	140			

CLUSTER 3						
3.1	244	162	127	273		
3.2	21	199	201	7		
3.3	289	253	82	26	168	
3.4	93	80	271	243	153	43
3.5	135	99	78	214	137	
3.6	35					

CLUSTER 4					
4.1	66	81	4	142	
4.2	238	39	13	58	
4.3	218	242	141		
4.4	19	287	64	295	75
4.5	172	278	224	22	291
4.6	47	28	284	223	

CLUSTER 5						
5.1	203	160	41	9	116	68
5.2	276	221	23	106	188	215
5.3	132	128	232	69	110	96
5.4	34	38	27	45	84	107
5.5	100					

CLUSTER 6					
6.1	202	167	144	104	
6.2	282	150	55	270	30
6.3	192	222	239	49	279
6.4	212	251	174	157	126
6.5	177	213	44	11	114

CLUSTER 7

CLUSTER 9				
9.1	211	297	266	169

Figura 7. Rutas generadas por cluster

	lunes	martes	miercoles	jueves	viernes
Promotor 1	1.1	1.2	1.3	1.4	1.5,1.6
Promotor 2	2.2	2.3	2.4	2.6	2.7
Promotor 3	3.1	3.2	3.3	3.4	3.5
Promotor 4	4.1	4.2	4.3	4.4	4.5
Promotor 5	5.1	5.2	5.3	5.4	12.3
Promotor 6	6.1	6.2	6.3	6.4	6.5
Promotor 7	7.1	7.2	7.3	7.4	12.2
Promotor 8	8.1	8.2	8.3	8.5	8.6
Promotor 9	9.1	9.2	9.4	9.4	9.5
Promotor 10	10.1	1.2	10.3	10.4	10.5
Promotor 11	11.1	11.2	11.3	11.4	12.1

Figura 8. Propuesta de ruteo por semana

Referencias

Daza, Julio Mario, Montoya, Jairo R, & Narducci, Francesco. (2009). RESOLUCIÓN DEL PROBLEMA DE ENRUTAMIENTO DE VEHÍCULOS CON LIMITACIONES DE CAPACIDAD UTILIZANDO UN PROCEDIMIENTO METAHEURÍSTICO DE DOS FASES. Revista EIA, (12), 23-38. Retrieved June 09, 2024, from http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1794-12372009000200003&lng=en&tlng=es.

Hernán Calderón, J., Gómez, A., & Góngora, A. (2016). Metodología para el diseño de rutas en un sistema de logística inversa. Caso: banco de alimentos. INVENTUM, 11(20), 15–23. https://doi.org/10.26620/uniminuto.inventum.11.20.2016.15-23

Tunnisaki, F., & Sutarman, N. (2023). Clarke and Wright Savings Algorithm as Solutions Vehicle Routing Problem with Simultaneous Pickup Delivery (VRPSPD). Journal of Physics. Conference Series, 2421(1), 012045. https://doi.org/10.1088/1742-6596/2421/1/012045

Bong, S. (2022). A new solution to vehicle routing problem - Sethatevy Bong - medium. Medium. https://sethatevy-bong.medium.com/a-new-solution-to-vehicle-routing-problem-fee8f2c1115f