Route Optimization Project by Stella Wong

Background:

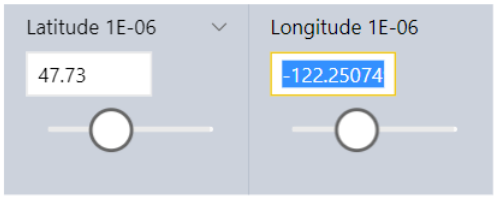
1. As the economic recovers from the COVID-19 pandemic, a lot of the customers call in to resume their waste collection services and some want to start new business with us. However, our customer service team were not able to confirm the service day immediately on the phone, they need to wait until dispatch put those customers into a route and call back.

2. We learned from the financial reviews that our transportation expenses are increasingly high, in addition to the storage of driver, we need to find a way to reduce trucking cost. We are not changing the route schedule of the existing customers, but we can optimize routing for new or returning customers.

Objective:
In order to simplify the operation procedure and improve customer service quality, this tool was designed to help our customer service representatives route a potential location. More importantly, it would minimize the transportation cost by choosing the most efficient route we have.

Modeling:
For each of our existing routes, the model compares the distances between the potential location and each existing stops and picks the distance to the closest stop. Then, by comparing the distance to the closest stop of all existing routes, the route with the shortest distance will be optimal as it should take the least of time to go from one stop to another.

Route Optimization Tool:



Get the coordinates of a place
On your computer, open Google Maps.
Right-click the place or area on the map.
To copy the coordinates automatically, select the latitude and longitude.

Distance to the closest stop by "Route" - "DayOfWeek"

MaterialCat	GeoMiles
RECYCLE	**2.1**
***F**0 - WED	2.1
***F**0 - FRI	2.1
***F**0 - MON	2.2
***F**2 - MON	2.2
***F**0 - THU	2.4
***F**0 - TUE	2.4
***F**2 - SAT	3.9
***R**1 - FRI	3.9
***R**1 - MON	3.9
***R**1 - THU	3.9
Distance to the closest stop	**2.1**

MaterialCat
- ☐ ORGANICWST
- ■ RECYCLE

DayOfWeek
- ☐ (1) MON
- ☐ (2) TUE
- ☐ (3) WED
- ☐ (4) THU
- ☐ (5) FRI
- ☐ (6) SAT

Instruction:

1. Obtain the geocode of the potential location from Google Maps, then enter the Latitude and Longitude boxes separately.

2. Select a Material Category of waste the customer requested, i.e., Organic Waste or Recycle.

3. If the customer had specified their preferred Day of Week, check the Day of Week filter too. If there is no available route, nothing will return on the Distance Table.

4. The Distance Table will show distance from the geocode you entered to the closest stop for each existing route in Geo Miles, with the "Route" – "DayOfWeek" combination sorted from the best (on the top) to worst (at the bottom).

5. If you expand the "Route" – "DayOfWeek" combination using the +/- button on the table, all existing stops in the route will be sorted by stops with the shortest distance on the top.

Example:

For a *recycle* customer with geocode *47.729271, -122.250749*, it would be optimal to schedule into Route ***F**0 on Wed (Top 1) or Route ***F**0 on Fri (Top 2) because it's only 2.1 Geo Miles away from the closest existing stop in these two routes.

Note: I've replaced part of the route number with asterisks (*) for privacy reason.

Dax code:

1. Use SQL query to generate existing route information – Route, Day of Week, Latitude of stops, Longitude of stops, Material Category, etc. and load the table into Power BI, name it as `'ExistingStops'`

2. Create measures to capture the geocode of an existing stop

```
stop_x = IFERROR(VALUES('ExistingStops'[Latitude]), BLANK())
stop_y = IFERROR(VALUES('ExistingStops'[Longitude]), BLANK())
```

3. Create the parameter tables for potential latitudes and longitudes on Power BI using `GENERATESERIES`

```
Latitude = GENERATESERIES(47, 49, 1E-06)
Longitude = GENERATESERIES(-123, -121, 1E-06)
```

4. Create measures to capture the geocode entered on the tool

```
Latitude Value = SELECTEDVALUE('Latitude'[Latitude])
Longitude Value = SELECTEDVALUE('Longitude'[Longitude])
```

5. Apply **Haversine formula** to find distance between the geocode entered on the tool and stops in an existing route in Geo Miles:

```
GeoMiles =
VAR r = 3958.7558657441
VAR dx = [Latitude Value]-[stop_x]
VAR dy = [Longitude Value]-[stop_y]
```

```
VAR a = POWER(SIN((dx*PI()/180)/2), 2) + (COS([Latitude
Value]*PI()/180)*COS([stop_x]*PI()/180)*POWER(SIN((dy*PI()/180)/2), 2))

VAR c = 2*ATAN(SQRT(a)/SQRT(1-a))
VAR d = r*c

RETURN d
```

6. Then use `MINX` to compute the closest distance:

```
MinGeoMiles = MINX('ExistingStops', [GeoMiles])
```

Note: The measure `MinGeoMiles` has been renamed as "GeoMiles" on the tool (see screenshot)

Reference:

1. Project 3, Part B: Descriptive Spatial Statistics | GEOG 586: Geographic Information Analysis (psu.edu)
2. SPATIAL STATISTICS (arcgis.com)