

Brewing Data Cup 2020 - Reto México

Equipo OrozCuspi Data_Mx: Felipe Orozco & Victor Cuspinera

29/11/2020

Problema

El reto para los participantes en México es de logística, específicamente para **optimizar ruta**, con el objetivo de dividir un centro de distribución en seis territorios de reparto a través de un modelo de optimización que busque minimizar la distancia recorrida, con diversas restricciones: balancear los territorios por volumen sin exceder la capacidad de reparto por día, los territorios deben estar en función de la frecuencia de visita al cliente, y territorios balanceados con exactamente el mismo número de clientes.

Análisis

Inicialmente cargamos librerías y base de datos.

##	Id_Cliente	id_Agencia	Frecuencia	Vol_Entrega	lat	lon
## 1	1	A1	1	2	20.50605	-98.21238
## 2	2	A1	1	17	20.45813	-98.22125
## 3	3	A1	1	2	20.53043	-98.23686
## 4	4	A1	3	15	20.48866	-98.20294
## 5	5	A1	1	2	20.53325	-98.21967
## 6	6	A1	1	19	20.53106	-98.18264

Repetimos los renglones n veces la 'Frecuencia' correspondiente.

```
data2 = data.frame()
for (i in 1:3){
  for (j in 1:i){
    aux <- data %>% filter(Frecuencia == i)
    aux$repetition <- j
    data2 <- rbind(data2, aux)
    # data_freq2 <- data %>% filter(Frecuencia == 2)
    # data_freq3 <- data %>% filter(Frecuencia == 3)
    # data2 <- rbind(data_freq1, data_freq2, data_freq2, data_freq3, data_freq3, data_freq3)
    # data2
  }
}

data2 %>% filter(Id_Cliente==171)
```

##	Id_Cliente	id_Agencia	Frecuencia	Vol_Entrega	lat	lon	repetition
## 1	171	A1	3	15.33333	20.46837	-98.17653	1
## 2	171	A1	3	15.33333	20.46837	-98.17653	2
## 3	171	A1	3	15.33333	20.46837	-98.17653	3

Calculamos el centroide, suponiendo que esa es la ubicación del Centro de Distribución. El centroide se calcula a partir de Longitud/ Latitud

```
# data[,c("lon", "lat")]
```

```
ubicaciones.coord<-data2[,c("lon", "lat")]
centroide <- centroid(ubicaciones.coord)
centroide
```

```
##          lon      lat
## [1,] -98.20556 20.49266
```

```
data2 <- rbind(data2, c(9999, "CEDI", 1, 0, centroide[2], centroide[1]))
data2[nrow(data2),]
```

```
##      Id_Cliente id_Agencia Frecuencia Vol_Entrega      lat
## 3978      9999      CEDI          1          0 20.4926616230121
##                                lon repetition
## 3978 -98.2055581242414          9999
```

Calculamos distancias entre clientes

```
latitud = data2[, 5]
longitud = data2[, 6]

matriz_dist <- geo_distance( data.frame(longitud, latitud) )
```

```
# añadir penalización
matriz_dist[matriz_dist == 0 ] <- 20* max(matriz_dist)
```

```
## fragmento de matriz de distancias
# matriz_dist[(nrow(data2)-30):nrow(data2), (nrow(data2)-30):nrow(data2)]
```

Referencias

Arga Adyatama. (2020). “Capacitated Vehicle Routing Problem (CVRP) with Nearest Neighbour and Genetic Algorithm”. RPubS. Consultado el 2020-11-27. Link: <https://rpubs.com/Argaadya/cvrp>

Arga Adyatama. (2020). “Machine Learning Implementation in Manufacturing Industry”. GitHub repository. Consultado el 2020-11-27. Link: <https://github.com/Argaadya/manufacturing>