Instituto Politécnico Nacional

Escuela Superior de Cómputo

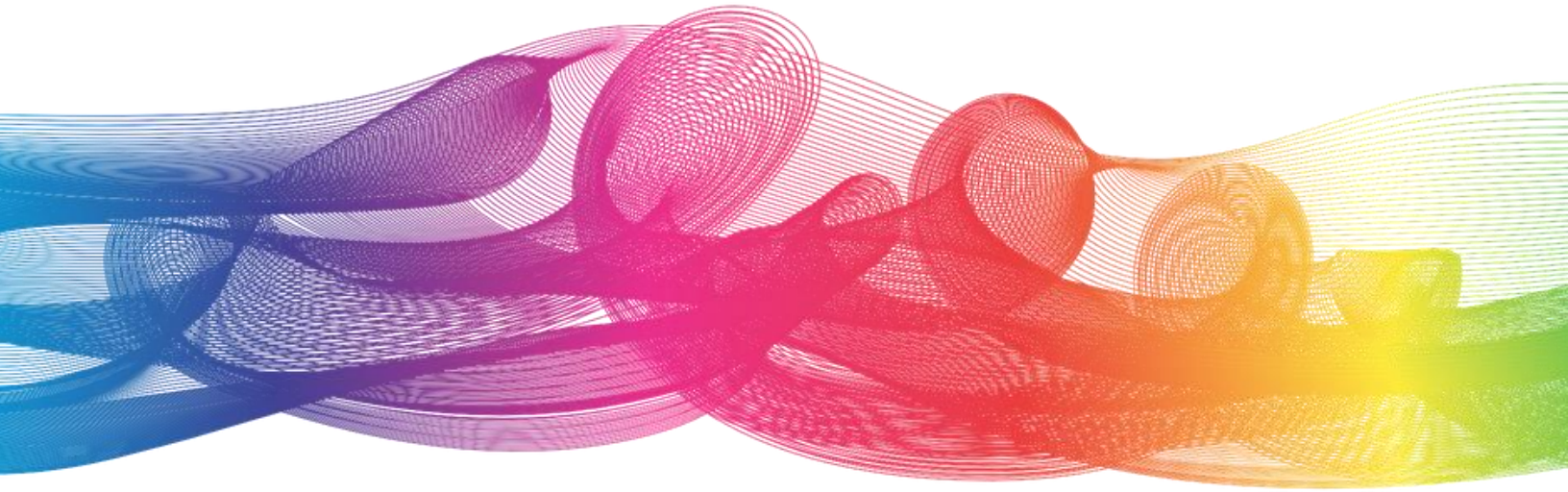
Grupo 3CV19 Data Mining

Profesor Zagal Flores Roberto Eswart

3er Parcial Practica #7: Lattice de cubo de datos y exploración del datawarehouse del proyecto semestral.

Alumno García González Aarón Antonio

Lunes 31 de Mayo de 2021



Índice

Objetivo

Comprender el concepto de lattice de cubos de datos, y utilizar la arquitectura de datos para minería.

Introducción

Un viajero de negocios o un turista, puede elegir la opción que más se ajuste a su presupuesto y a sus necesidades, paga por medio de Internet, coordina con el dueño por medio de la misma aplicación y listo.

AirBnb se gana una pequeña comisión por eso. El propietario recibe el dinero cuando el huésped deja el hospedaje.

Ambos dejan una calificación uno del otro y así cada uno aprovecha esta forma de negocio moderna de compartir recursos que antes estaban ociosos.

Como en todo negocio, algunos tienen mucho éxito y otros fracasan, o ganan poco dinero.

El enfoque de este proyecto es sobre conocimiento inicial de negocio AirBnB en CDMX, datos que se esperan encontrar serían de utilidad para personas o emprendedores que comienzan en esta área de oportunidad, con el objetivo de reconocer en que delegaciones son las más competitivas, la influencia que tiene el número de hoteles, estaciones de metro y metrobus, que atributos generales como el numero de noches mínimas de estadía, el numero de reviews, precios etc, pueden influir y en que importancia lo hacen.

Desarrollo

Procedimiento: Objetivo: Desarrollar la automatización de la lattice de cubos de datos para construir el data warehouse del proyecto semestral, y hacer la exploración de este.

1. Revise las clases relacionadas a la Lattice de cubos de datos, y la clase de exploración de tableau y mapas de calor. ☑
2. Seleccione entre 4 y 6 dimensiones (n) de la tabla de hechos diseñada en la clase anterior. Recuerde que la cantidad máxima de cubo de datos es 2n. ☑

→ n = # dimensiones
(1) fecha (año)
(2) fecha (mes)
(3) # noches mínimas
(4) Tipo de air bnb
(5) Alcaldía

→ Combinaciones

Grupos de n=2

$$C_m^n = \frac{5!}{2!(5-2)!} = \frac{5!}{2!3!} = \frac{4 \times 5}{2} = \frac{20}{2} = 10$$

Grupos de n=3

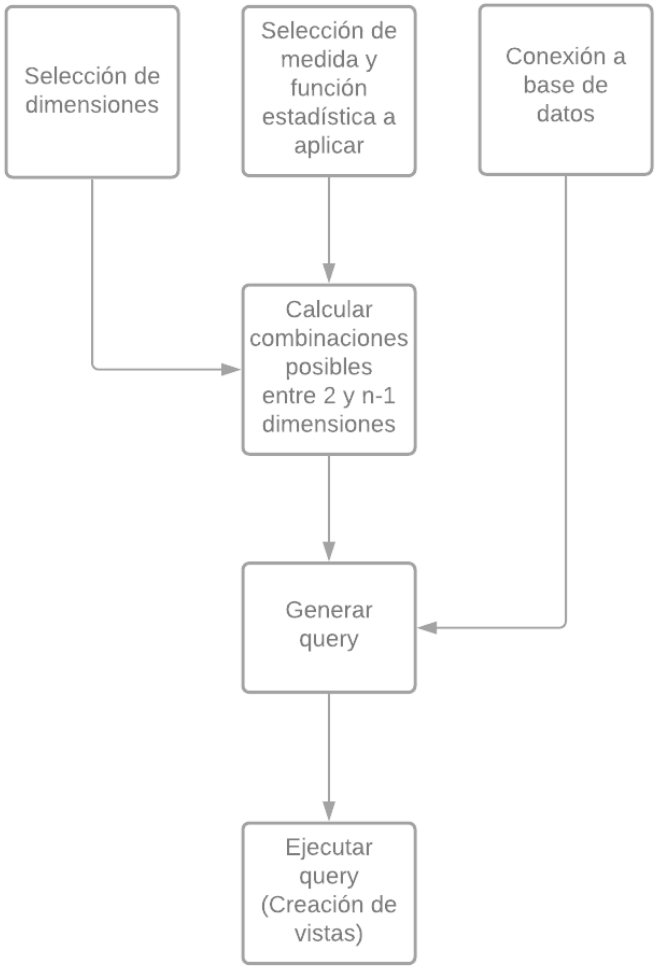
$$C_m^n = \frac{5!}{3!(5-3)!} = \frac{5!}{3!2!} = \frac{4 \times 5}{2} = \frac{20}{2} = 10$$

Grupos de n=4

$$C_m^n = \frac{5!}{4!(5-4)!} = 5$$

Total de combinaciones diferentes 25

3. Construya un programa, en el lenguaje de programación de preferencia, donde construya la lattice de cubos de su proyecto. Tienes dos posibilidades sobre la materialización de los cubos
   a. Crear vistas en la base de datos, esto ahorraría espacio en disco, pero requerirá tiempo de procesamiento al conectar al cubo a otro software como tableau. También podrá controlar el código fuente con el que la vista o cubo fue realizado. ☑
   b. Crear tablas en la base de datos. Esto quitará espacio en disco duro, pero ahorraría tiempo de ejecución y procesamiento al exportar el cubo.
   c. Desarrolle un diagrama de flujo o de procesos, explique el diagrama en el reporte y agregue fragmentos de código en las tareas más importantes.
   d. Recuerde explicar las medidas estadísticas usadas: count, avg, max, min, etc.



Decidí seleccionar las dimensiones de fecha (año), fecha (mes), delegación, tipo de renta y numero de noches mínimos, considero que son de los datos más importantes, y como medida estoy utilizando el promedio en precio.

Se trabaja en lenguaje Python usando librerías de conexión a base de datos y pandas, el motor con el cual se conecta es con sql server.

A continuación, muestro código donde se obtienen las combinaciones necesarias y a tomar en cuenta, dado que son 5 dimensiones, se exploran niveles 2D, 3D y 4D.

```
def potencia(c):
    if len(c) == 0:
        return [[]]
    r = potencia(c[:-1])
    return r + [s + [c[-1]] for s in r]


def combinaciones(c, n):
    return [s for s in potencia(c) if len(s) == n]


def combinaciones_intermedias(c):
    result = []
    for i in range(2, len(c)):
        result.extend(combinaciones(c,i))
```

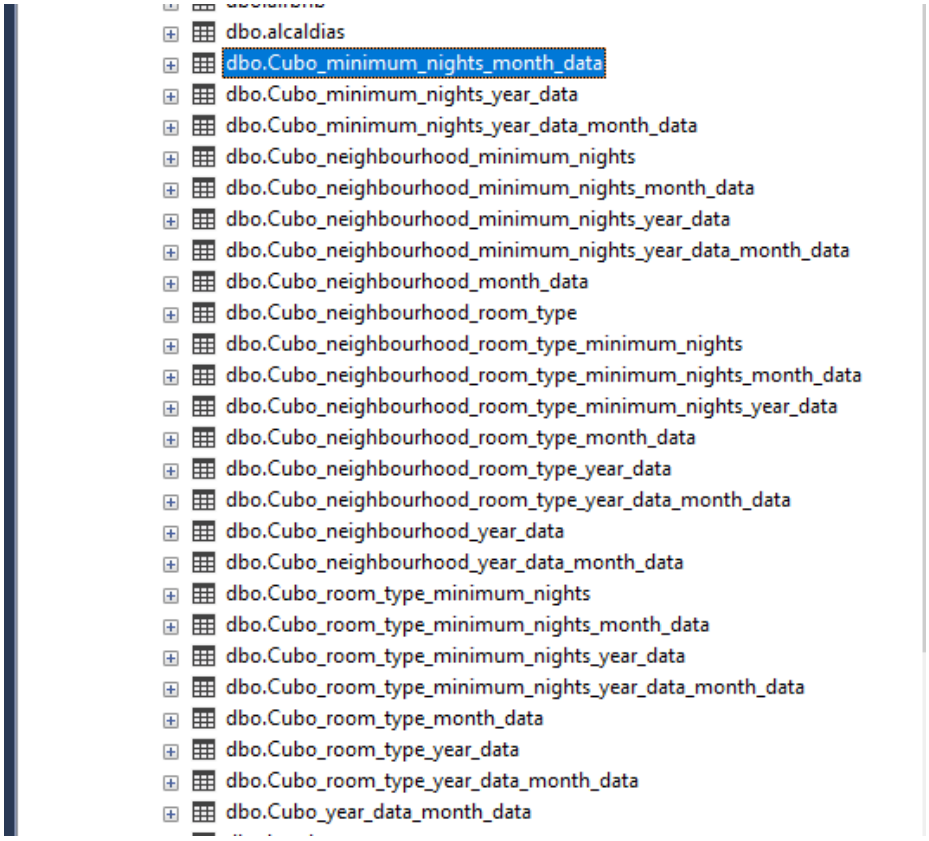
Posteriormente para generar el query de manera dinámica, donde primero incluimos las dimensiones a consultar acorde a la combinación que recibe como argumento, posterior a ello agregamos la medida en promedio de manera estática, concatenamos los nombres de las dimensiones separadas por guion bajo que serán los nombres de las tablas destino, y me manera dinámica incluimos el nombre de las dimensiones involucradas en el group by.

```
def generate_query_1(dimensions):
    result = "SELECT "

    for index, dimension in enumerate(dimensions):
        result += "[airbnb].[dbo].[airbnb].[" + dimension + "], "


    result += "AVG([airbnb].[dbo].[airbnb].[price]) AS 'Precio promedio' "
    result += "INTO [airbnb].[dbo].[" + generar_nombre_cubo(dimensions) + "] "
    result += "FROM [airbnb].[dbo].[airbnb] "


    result += "GROUP BY "

    for index, dimension in enumerate(dimensions):
        if index != len(dimensions) -1:
            result += "[airbnb].[dbo].[airbnb].[" + dimension +"], "
        else:
            result += "[airbnb].[dbo].[airbnb].[" + dimension +"]"

    result += ";"

    return result
```

Programa principal

```
def main():
    dimenciones = ["neighbourhood", "room_type", "minimum_nights", "year_data",
"month_data"]

    server = 'localhost'
    database = 'airbnb'
    username = 'Garcia'
    password = '123456'
    cnxn = pyodbc.connect('DRIVER={ODBC Driver 17 for SQL
Server};SERVER='+server+';DATABASE='+database+';UID='+username+';PWD='+ password)
    cursor = cnxn.cursor()


    for combinacion in combinaciones_intermedias(dimenciones):
        query = generate_query_1(combinacion)
        print("Generando cubo: " + str(generar_info_cubo(combinacion)) + " ...")
        print(query)
        print("")

        cursor.execute(query)
        cnxn.commit()


if __name__ == "__main__":
    main()
```

4. Realice los ajustes que considere necesarios, elimine dimensiones, renombre el nombre de las dimensiones. Trate de utilizar valores numéricos en medida de lo posible en su tabla de hechos que usará para crear la lattice.
   a. Recuerde que después de crear el cubo de datos puede vincularlo con sus respectivas dimensiones o catálogos.
   b. En caso de que el punto anterior no sea posible (e.j un cálculo por delegación debe considerar en el agrupamiento dicho campo de la dimensión)



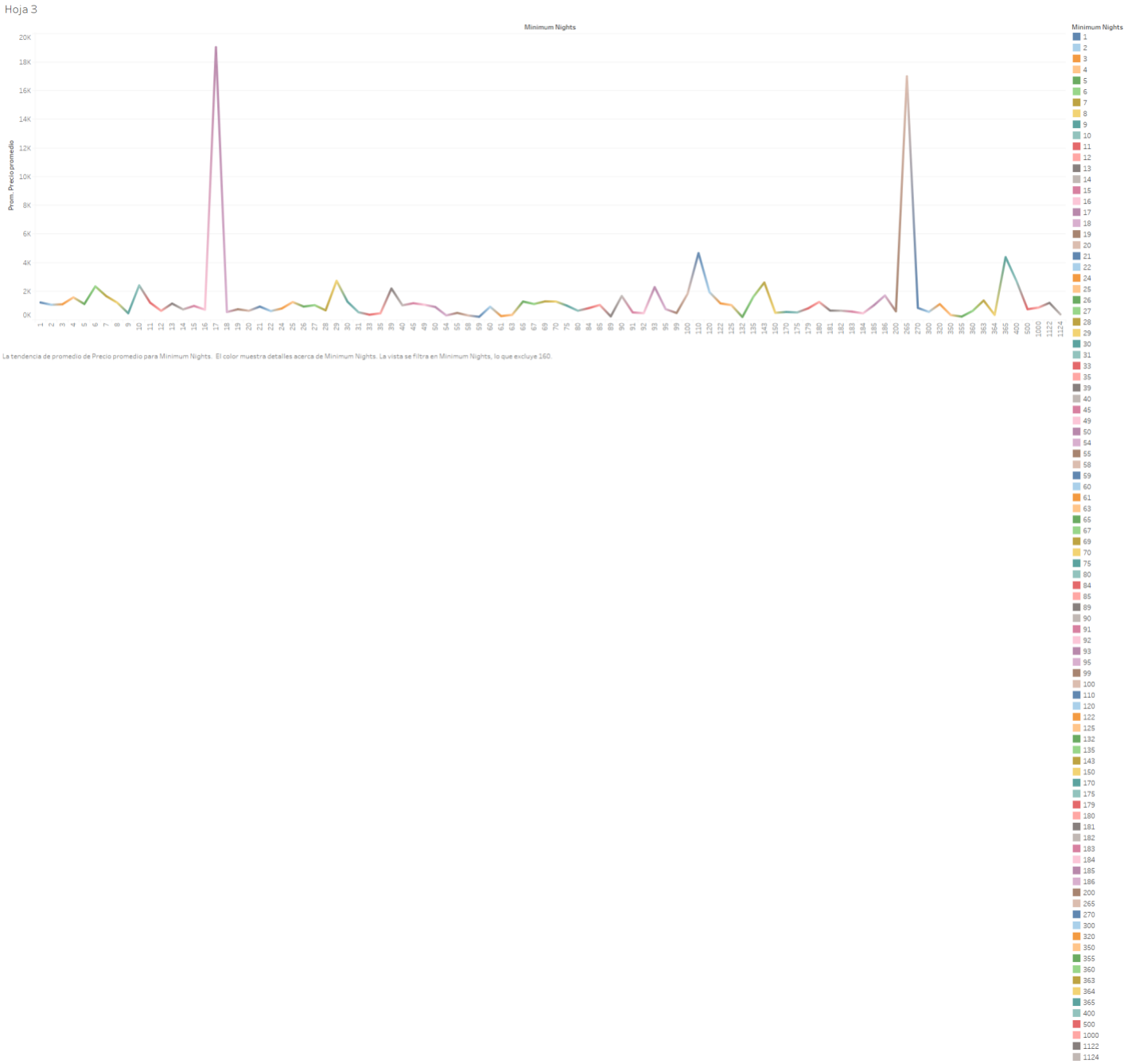
No se renombro ningún campo, esto se realizo desde el procedimiento ETL, comprobamos que obtenemos los 25 cubos intermedios, es decir 2 o más dimisiones, pero menor al número total de dimensiones.

5. Conecte el datawarehouse a tableau y Desarrolle la exploración de los cubos de datos. Desarrolle el análisis sobre los cubos de datos de nivel 3 (dimensiones de espacio y tiempo, más una dimensión diferente a las diferentes (seleccione una dimensión interesante a su criterio)). Posteriormente, agregue otros cubos de distintos niveles que a su criterio considere más importantes.
   a. Recuerde que su proyecto debe incluir alguna referencia geográfica, por lo tanto, desarrolle un mapa de calor en Tableau.
   b. Explique los filtros usados y las razones por las que se usan las dimensiones seleccionadas
6. Al agregar estas gráficas, mapas y resultados a su reporte, intente dar una interpretación de los resultados obtenidos en los dos puntos anteriores.
   a. Recuerde que la interpretación depende de las medidas estadísticas usadas: count, avg, max, min, etc.

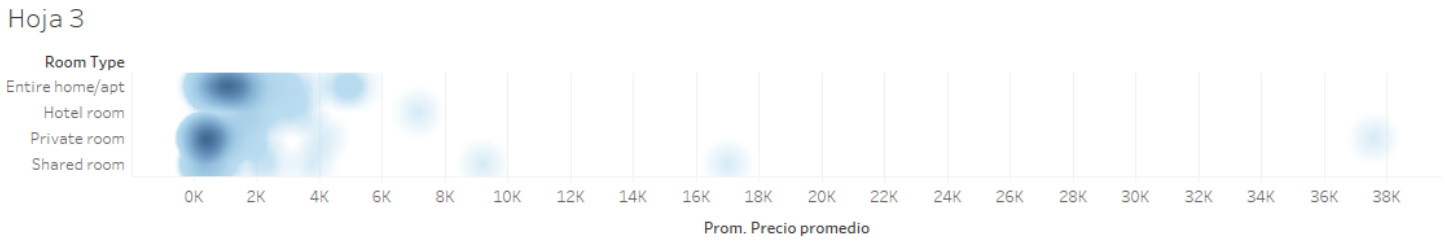
Hoja 2

Suma de Precio promedio para cada Alcaldía1 desglosado por Room Type. El color muestra detalles acerca de Neighbourhood.

Claramente como lo vimos en la practica anterior, la delegación Miguel Hidalgo siempre destaca, en el caso de renta de departamento entero y cuarto de hotel esta ultima delegación tiene los precios mas altos, debo admitir que me genera duda para el caso de cuarto privado que la delegación Iztapalapa tiene los mayores precios seguido de Xochimilco, para el caso de renta de habitación compartida la delegación Benito Juárez es la que precios más altos tiene.

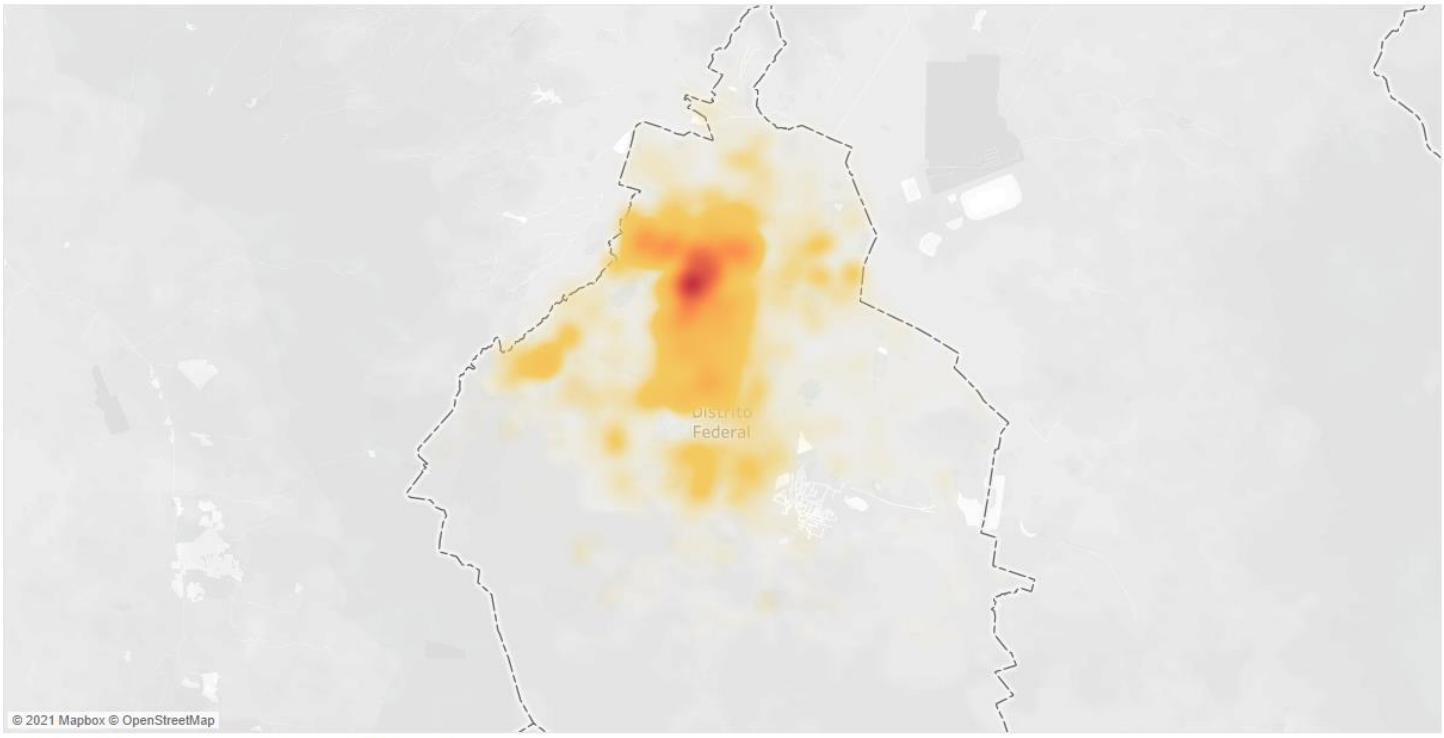


Esta gráfica muestra los precios por días mínimos, deberé revisar nuevamente sobre esta dimensión, ya que yo esperaría que conforme el numero de días aumenta, el precio también lo debería de hacer, pero pareciera que es constante el precio, quiero pensar que se muestra el precio por día y no por todo el periodo que comprende los días mininos.



Promedio de Precio promedio para cada Room Type. Se muestran detalles para Minimum Nights. La vista se filtra en Minimum Nights, lo que excluye 160.

Obviamente la renta de departamentos completos es el precio más alto, pero para habitación de hotel, cuarto privado o cuerpo compartido el precio no hay mucha variación.



Mapa basado en Longitude y Latitude. Se muestran detalles para Alcaldía.

Como lo vimos en las practicas anteriores, la mayor concentración de oferta y demanda se ubica en la región centro norte de la ciudad, es decir, delegaciones Cuauhtémoc, Benito Juárez y Miguel Hidalgo.

Conclusiones

Fue una practica sencilla, lo que más me costo fue realizar la generación de queries dinámicos para la creación de vistas o tablas, al utilizar Python con sql server que fue la primera vez que realicé esta combinación, me encontré con el problema que no se guardaban las vistas, mismo porque es necesario realizar un commit que no se había realizado.

Para generar lattice hay 2 a la n posibilidades, algunas de ellas no son necesarias o no tienen sentido, prácticamente se construye por capas o número de combinaciones.