In [1]:
```
import requests
import json
import pandas as pd
import matplotlib.pyplot as plot
```

In [2]:
```
za la llamada al API para la obtencion de datos

tps://datos.cdmx.gob.mx/api/3/action/datastore_search?resource_id=48fcb848-2
= requests.get(url)
response.status_code
 == requests.codes.ok: #Si la llamada al servicio regresa un 200
("ok")
(response)
nse_decoded = response.content.decode("utf-8") #Se decodifica el contendido
nse_json = json.loads(response_decoded) # El contenido se trata de convertir
t = response_json["result"]
ds  = result["records"]
pd.json_normalize(records, max_level=0)
```

```
ok
<Response [200]>
```

In [3]:
```
df
```

Out[3]:

	_id	ao_hechos	mes_hechos	fecha_hechos	ao_inicio	mes_inicio	fecha_inicio	deli
0	1	2016	Enero	2016-01-31 22:16:00	2016	Febrero	2016-02-01T00:25:44	DAÑO E PROPIEDA AJEN INTENCIONA
1	2	2016	Enero	2016-01-31 20:50:00	2016	Febrero	2016-02-01T00:52:37	ROBO I VEHICULO I SERVIC PARTICULAR CO V
2	3	2016	Febrero	2016-02-01 00:30:00	2016	Febrero	2016-02-01T01:33:26	NARCOMENUDE POSESIO SIMP
3	4	2016	Enero	2016-01-31 22:00:00	2016	Febrero	2016-02-01T02:09:11	ROBO TRANSEUNTE E VIA PUBLICA CO VIOLENC
4	5	2015	Diciembre	2015-12-25 12:00:00	2016	Febrero	2016-02-01T02:16:49	DENUNCIA I HECHO
...	...	...	...	...	...	...	...	
95	95	2016	Febrero	2016-02-01 15:30:00	2016	Febrero	2016-02-01T19:10:52	ROBO TRANSEUNTE E VIA PUBLICA CO VIOLENC
96	96	2016	Enero	2016-01-30 10:50:00	2016	Febrero	2016-02-01T19:12:51	ROBO TRANSEUNTE E VIA PUBLICA CO VIOLENC
97	97	2014	Octubre	2014-10-24 12:30:00	2016	Febrero	2016-02-01T19:13:34	FRAUI
98	108	2016	Febrero	2016-02-01 18:15:00	2016	Febrero	2016-02-01T21:08:18	DAÑO E PROPIEDA AJEN INTENCIONA
99	98	2016	Febrero	2016-02-01 19:00:00	2016	Febrero	2016-02-01T19:21:30	VIOLENC FAMILIA

100 rows × 20 columns

1. **¿Qué pruebas identificarías para asegurar la calidad de estos datos? No es necesario hacerlas, sólo describe la prueba y lo que te dice cada una.**

In [4]:
```
df.count() # nos dice por cada columna cuantos valores tenemos sin contar N
```

Out[4]:
```
_id                    100
ao_hechos              100
mes_hechos             100
fecha_hechos           100
ao_inicio              100
mes_inicio             100
fecha_inicio           100
delito                 100
fiscalia               100
agencia                100
unidad_investigacion   100
categoria_delito       100
calle_hechos           100
calle_hechos2          100
colonia_hechos         100
alcaldia_hechos        100
competencia            100
longitud               100
latitud                100
tempo                  100
dtype: int64
```

Se puede ver que la API esta devolviendo 100 filas, comparado con el archivo csv en la página este tiene mas de 1M, por lo que se decide mejor cargar el archivo que tiene mas información.

In [5]:
```
df = pd.read_csv('carpetas_completa_febrero_2022.csv')
df.count()
```

```
/Users/aletapia/opt/anaconda3/lib/python3.9/site-packages/IPython/core/in
teractiveshell.py:3444: DtypeWarning: Columns (15) have mixed types.Speci
fy dtype option on import or set low_memory=False.
  exec(code_obj, self.user_global_ns, self.user_ns)
```

Out[5]:
```
ao_hechos               1400873
mes_hechos              1400873
fecha_hechos            1400873
ao_inicio               1401331
mes_inicio              1401331
fecha_inicio            1401328
delito                  1401331
fiscalia                1401329
agencia                 1401331
unidad_investigacion    1401104
categoria_delito        1401331
calle_hechos            1397390
calle_hechos2            539997
colonia_hechos          1340993
alcaldia_hechos         1397166
competencia              337252
longitud                1341941
latitud                 1341941
tempo                         0
dtype: int64
```

In [6]:
```
df.head()
```

Out[6]:

	ao_hechos	mes_hechos	fecha_hechos	ao_inicio	mes_inicio	fecha_inicio	delito	
0	2016.0	Enero	2016-01-31 22:16:00	2016	Febrero	2016-02-01 00:25:44	DAÑO EN PROPIEDAD AJENA INTENCIONAL	INV
1	2016.0	Enero	2016-01-31 20:50:00	2016	Febrero	2016-02-01 00:52:37	ROBO DE VEHICULO DE SERVICIO PARTICULAR CON VI...	INV A D
2	2016.0	Febrero	2016-02-01 00:30:00	2016	Febrero	2016-02-01 01:33:26	NARCOMENUDEO POSESION SIMPLE	INV NII
3	2016.0	Enero	2016-01-31 22:00:00	2016	Febrero	2016-02-01 02:09:11	ROBO A TRANSEUNTE EN VIA PUBLICA CON VIOLENCIA	INV EN
4	2015.0	Diciembre	2015-12-25 12:00:00	2016	Febrero	2016-02-01 02:16:49	DENUNCIA DE HECHOS	INV

In [7]:
```
len(df)
```

Out[7]:
```
1401331
```

Se aplica la function count() o info() para si todas las columnas tienen un valor y poder darnos una idea que porcentaje respecto al total de filas tiene un valor valido por ejemplo las columnas calle_hechos2,competencia,tempo son carateristicas que tienen la mayoria valores invalidos.Y tomar la decision de descartarlas o ver como podemos llenar los datos faltantes.

Tambien existen los diagramas de bigotes para saber la distribucion de valores númericos y asi saber si tenemos valores atopicos tenemos y tomar la decision de descartarlos.

In [8]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1401331 entries, 0 to 1401330
Data columns (total 19 columns):
 #   Column                Non-Null Count    Dtype  
---  ------                --------------    -----  
 0   ao_hechos             1400873 non-null  float64
 1   mes_hechos            1400873 non-null  object 
 2   fecha_hechos          1400873 non-null  object 
 3   ao_inicio             1401331 non-null  int64  
 4   mes_inicio            1401331 non-null  object 
 5   fecha_inicio          1401328 non-null  object 
 6   delito                1401331 non-null  object 
 7   fiscalia              1401329 non-null  object 
 8   agencia               1401331 non-null  object 
 9   unidad_investigacion  1401104 non-null  object 
 10  categoria_delito      1401331 non-null  object 
 11  calle_hechos          1397390 non-null  object 
 12  calle_hechos2         539997 non-null   object 
 13  colonia_hechos        1340993 non-null  object 
 14  alcaldia_hechos       1397166 non-null  object 
 15  competencia           337252 non-null   object 
 16  longitud              1341941 non-null  float64
 17  latitud               1341941 non-null  float64
 18  tempo                 0 non-null        float64
dtypes: float64(4), int64(1), object(14)
memory usage: 203.1+ MB
```

Describe() Tambien nos ayuda para saber la mediana , el conteo de filas, los cuartiles, desviacion standar (nos ayuda saber que tan dispensos estan nuestros datos) y min y maximo valor de una columna.

In [9]:
```
df.describe()
```

Out[9]:

	ao_hechos	ao_inicio	longitud	latitud	tempo
count	1.400873e+06	1.401331e+06	1.341941e+06	1.341941e+06	0.0
mean	2.018462e+03	2.018617e+03	-9.913714e+01	1.938701e+01	NaN
std	2.022233e+00	1.728106e+00	6.015875e-02	7.029166e-02	NaN
min	1.906000e+03	2.016000e+03	-1.002319e+02	1.909535e+01	NaN
25%	2.017000e+03	2.017000e+03	-9.917560e+01	1.933889e+01	NaN
50%	2.018000e+03	2.019000e+03	-9.914198e+01	1.938953e+01	NaN
75%	2.020000e+03	2.020000e+03	-9.909932e+01	1.943780e+01	NaN
max	2.022000e+03	2.022000e+03	-9.894686e+01	1.958333e+01	NaN

2. Identifica los delitos que van a la alza y a la baja en la CDMX (ten cuidado con los delitos con pocas ocurrencias)

Se me ocurre hacer una regresion lineal para predecir el siguiente dia el número de incidencias por delito y asi saber conforme a la linea de tiempo yo tomaría la variable fecha_hechos como base para saber el comportamiento (pero antes tranformaria esta variable a que solo sea yyyy-mm-dd) y obtener el conteo de incidencias por cada dia de esta forma veremos el comportamiento y de cierta el valor que se predijo se puede comparar con el valor anterior y ver el procentaje de cambio por ejemplo si el dia 9 de abril se tuvo 10 incidencias y el valor que se predijo para el 10 abril es 5 incidencias hubo un cambio del -50% entonces indica que va a la baja. O otra solución sin regresion lineal es calcular el procentaje de cambio por cada dia como lo hicimos anteriormente y comparar un dia anterior con el dia siguiente y ver si este porcentaje es positivo o negativo, sabremos si va a la alza o a la baja.

3.¿Cuál es la alcaldía que más delitos tiene y cuál es la que menos? ¿Por qué crees que sea esto?.

Primero obtengo el numero de filas por alcaldia, en este caso como veo que la columna ´categoria_delito´ tiene todos sus filas con valor la tomo como referencia para hacer un conteo de ocurrencias por alcaldia

In [10]:
```
dfDelito = df.groupby("alcaldia_hechos", as_index=False)[["categoria_delito
dfDelito = dfDelito.rename(columns={'categoria_delito': 'incidencias'})

dfDelito
```

Out[10]:

	alcaldia_hechos	incidencias
0	ABALA	1
1	ACAMBARO	3
2	ACAMBAY	5
3	ACAPULCO DE JUAREZ	73
4	ACATLAN	3
...	...	...
569	ZINAPECUARO	2
570	ZIRACUARETIRO	1
571	ZITACUARO	5
572	ZITLALTEPEC DE TRINIDAD SANCHEZ SANTOS	1
573	ZUMPANGO	62

574 rows × 2 columns

Se verifica que el dataframe tenga 2 columnas para hacer una gráfica , por la cantidad de clases de

la variable alcaldia la grafica no ayuda mucho en ver quien tiene mayores o menores ocurrencias

Por lo que se obtiene el maximo y el minimo de incidencias

In [12]:
```
maximo = dfDelito["incidencias"].max()
print(maximo)
```

```
218016
```

In [13]:
```
minimo = dfDelito["incidencias"].min()
print(minimo)
```

```
1
```

Se hace un filtrado para saber las alcaldias con menores y mayores incidencias

In [14]:
```
dfDelito[dfDelito["incidencias"] == minimo]
```

Out[14]:

	alcaldia_hechos	incidencias
0	ABALA	1
5	ACATZINGO	1
9	ACONCHI	1
11	ACUAMANALA DE MIGUEL HIDALGO	1
12	ACULCO	1
...	...	...
562	ZAPOTLAN DE JUAREZ	1
563	ZAPOTLAN EL GRANDE	1
567	ZIMATLAN DE ALVAREZ	1
570	ZIRACUARETIRO	1
572	ZITLALTEPEC DE TRINIDAD SANCHEZ SANTOS	1

242 rows × 2 columns

In [15]:
```
dfDelito[dfDelito["incidencias"] == maximo]
```

Out[15]:

	alcaldia_hechos	incidencias
145	CUAUHTEMOC	218016

4. ¿Existe alguna tendencia estacional en la ocurrencia de delitos (mes, semana, día de la semana, quincenas) en la CDMX? ¿A qué crees que se deba?

Por cada delito como es una variable catagorica la tranformaria a variables numerica y esta por lo cual ya tendría valores númericos, despues haría la transformacion de la variable fecha_hechos y dividirla en varias columnas para saber el mes (donde extraeria la parte de mes 1-12) , semana (1-4 donde calcularia el núm de semana del mes 1-4) , quincenas (1-2) y despues calcularia el coeficiente de correlacion respecto a la variable cantidad de incidencias vs mes, semana y asi por cada columna para saber si tienen relación.

5.¿Cuáles son los delitos que más caracterizan a cada alcaldía? Es decir, delitos que suceden con mayor frecuencia en una alcaldía y con menor frecuencia en las demás.

Se obtiene la cantidad de incidencias por alcaldia y delito

In [16]:
```
dfDelitoAlcadia = df.groupby(["alcaldia_hechos", "delito"], as_index=False)
dfDelitoAlcadia = dfDelitoAlcadia.rename(columns={'categoria_delito': 'inci

dfDelitoAlcadia
```

Out[16]:

	alcaldia_hechos	delito	incidencias
0	ABALA	DENUNCIA DE HECHOS	1
1	ACAMBARO	PRIVACION DE LA LIBERTAD PERSONAL	1
2	ACAMBARO	VIOLACION	1
3	ACAMBARO	VIOLENCIA FAMILIAR	1
4	ACAMBAY	ABUSO SEXUAL	1
...	...	...	...
7913	ZUMPANGO	SUSTRACCION DE MENORES	2
7914	ZUMPANGO	USURPACIÓN DE IDENTIDAD	2
7915	ZUMPANGO	VIOLACION	1
7916	ZUMPANGO	VIOLACION EQUIPARADA	1
7917	ZUMPANGO	VIOLENCIA FAMILIAR	14

7918 rows × 3 columns

Se obtiene el maximo y minimo de incidencias por alcaldia

In [17]:
```
dfDelitoAlcadiaMax = dfDelitoAlcadia.groupby(["alcaldia_hechos"], as_index=
```

In [18]:
```
dfDelitoAlcadiaMax
```

Out[18]:

	alcaldia_hechos	incidencias
0	ABALA	1
1	ACAMBARO	1
2	ACAMBAY	1
3	ACAPULCO DE JUAREZ	8
4	ACATLAN	1
...	...	...
569	ZINAPECUARO	1
570	ZIRACUARETIRO	1
571	ZITACUARO	1
572	ZITLALTEPEC DE TRINIDAD SANCHEZ SANTOS	1
573	ZUMPANGO	14

574 rows × 2 columns

In [19]:
```
aMin = dfDelitoAlcadia.groupby(["alcaldia_hechos"], as_index=False)[["incide
```

Despues se arma un conjunto con máximos y minimos

In [20]:
```
dfDelitofilter = pd.concat([dfDelitoAlcadiaMax,dfDelitoAlcadiaMin])
```

Se toma como ejemplo la alcaldia ZUMPANGO para verificar su maximo y minimo para despues ver que delitos fueron con mayor o menor frecuencia

In [21]:
```
dfDelitofilter[dfDelitofilter["alcaldia_hechos"] == "ZUMPANGO"]
```

Out[21]:

	alcaldia_hechos	incidencias
573	ZUMPANGO	14
573	ZUMPANGO	1

Despues se hace un filtrado al conjunto dfDelitoAlcadia para obtener los delitos con mayor y menos frecuencia en un solo dataframe

In [22]:
```
esMenores = pd.merge(dfDelitoAlcadia, dfDelitofilter, how="inner", on=["alca
esMenores[dfAlcDelitosMayoresMenores["alcaldia_hechos"] == "ZUMPANGO"]
```

Out[22]:

	alcaldia_hechos	delito	incidencias
3642	ZUMPANGO	DENUNCIA DE HECHOS	1
3643	ZUMPANGO	DESPOJO	1
3644	ZUMPANGO	HOMICIDIO CULPOSO FUERA DEL D.F (ATROPELLADO)	1
3645	ZUMPANGO	HOMICIDIO CULPOSO POR ARMA DE FUEGO	1
3646	ZUMPANGO	HOMICIDIO CULPOSO POR TRÁNSITO VEHICULAR (COLI...	1
3647	ZUMPANGO	INSOLVENCIA ALIMENTARIA	1
3648	ZUMPANGO	LESIONES CULPOSAS	1
3649	ZUMPANGO	LESIONES CULPOSAS POR CAIDA	1
3650	ZUMPANGO	LESIONES CULPOSAS POR TRANSITO VEHICULAR	1
3651	ZUMPANGO	LESIONES CULPOSAS POR TRANSITO VEHICULAR EN CO...	1
3652	ZUMPANGO	LESIONES INTENCIONALES	1
3653	ZUMPANGO	LESIONES INTENCIONALES POR ARMA DE FUEGO	1
3654	ZUMPANGO	PERDIDA DE LA VIDA POR QUEMADURA	1
3655	ZUMPANGO	PERSONAS EXTRAVIADAS Y AUSENTES	1
3656	ZUMPANGO	RETENCIÓN O SUSTRACCIÓN DE MENORES INCAPACES	1
3657	ZUMPANGO	ROBO A CASA HABITACION SIN VIOLENCIA	1
3658	ZUMPANGO	ROBO A REPARTIDOR Y VEHICULO CON VIOLENCIA	1
3659	ZUMPANGO	ROBO A TRANSPORTISTA Y VEHICULO PESADO SIN VIO...	1
3660	ZUMPANGO	ROBO DE OBJETOS	1
3661	ZUMPANGO	ROBO DE VEHICULO DE SERVICIO PARTICULAR SIN VI...	1
3662	ZUMPANGO	VIOLACION	1
3663	ZUMPANGO	VIOLACION EQUIPARADA	1
3664	ZUMPANGO	VIOLENCIA FAMILIAR	14

6. Diseña un indicador que mida el nivel de “inseguridad”. Genéralo al nivel de desagregación que te parezca más adecuado (ej. manzana, calle, AGEB, etc.). Analiza los resultados ¿Encontraste algún patrón interesante? ¿Qué decisiones se podrían tomar con el indicador?

Se puede hacer un indicador por colonia, calle de ahi agrupar por categoria de delito y asi saber mas a detalle que colonias, calle son las mas peligrosas y en que categoria estan las incidencias como de esta forma

In [24]:
```
dfDelitoCol = df.groupby(["colonia_hechos"], as_index=False)[["categoria_de
dfDelitoCol = dfDelitoCol.rename(columns={'categoria_delito': 'incidencias'

dfDelitoCol
```

Out[24]:

	colonia_hechos	incidencias
0	10 DE ABRIL	147
1	10 DE MAYO	467
2	12 DE DICIEMBRE	197
3	15 DE AGOSTO	544
4	16 DE SEPTIEMBRE	332
...	...	...
1667	ZONA URBANA EJIDAL LOS REYES CULHUACAN	494
1668	ZONA URBANA EJIDAL SANTA MARIA AZTAHUACAN	2265
1669	ZONA URBANA EJIDAL SANTA MARIA TOMATLAN	268
1670	ÁLAMOS	5076
1671	ÁLVARO OBREGÓN	1505

1672 rows × 2 columns

SECCION B

Yo penso que es un problema de optimización ya que necesitamos el mayor beneficio usando el minimo de recursos.

Aunque tambien se puede utilizar algoritmos de regresion lineal para predecir el número de paletas que se pueden consumir al dia siguiente y asi saber cuantas paletas necesito como minimo por maquina.

Se identifican las variables a reducir que son 100 pesos por maquina y el costo total por maquina para mantener la paletas en refrigeración

Nuestro KPI seria revisar que la indisponibilidad de paletas sea menor al 2% al mes de esta forma sabemos si hubo respuesta positiva o negativa

Por lo que empezaria a desarrollar esta formula por lo menos satisfacer la demanda por cada maquina

```
X: Cantidad de paletas disponibles
Y: Cantidad de paletas por maquina
W: Cantidad minima de paletas por maquina
Z: Cantidad de paletas por surtir
```

Entonces tendriamos Y 1-4000 , Z 1-4000 ,W 1-4000 (por que tenemos un total de 4000 máquinas)

```
Costo =  Y1 + Y2 +Y3 ...Y4000 + ((W1 - X1) > 0 ? $100 : $0 ))+
 ((W2 - X2) > 0 ? $100 : $0 ))...  ((W4000 - X4000) > 0 ? $100
: $0 ))

Y = Z + X1+ X2 + X3 ...X4000

Z = (W1 - X1 ) + (W2 - X2 )
```

Despues buscaria como obtener el minimo de la funcion Costo agregando una variable en cada cantidad de paletas por máquina por ejemplo seria (W1 - X1) + I donde I: es el incremento extra de paletas y asi saber que tanto puedo enviar de más o menos en cada maquina

In [ ]: