

Proyecto - Seminario de Sistemas 2

EDA Monovariable y Multivariable

Lectura de datos

```
In [2]: import pandas as pd
import geopandas as gpd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sqlalchemy import create_engine

In [67]: conn_str = 'postgresql+psycopg2://postgres:12345@localhost/covid_data'

# Crear el motor de SQLAlchemy
engine = create_engine(conn_str)

# Consulta SQL para extraer datos
sql_query_municipios = 'SELECT * FROM municipio;'
sql_query_departamentos = 'SELECT * FROM departamento;'

# Consulta para obtener los datos de muertes por municipio junto con nombres de mun
sql_query_fallecidos_municipio = 'SELECT FM.id_municipio, FM.fecha, FM.fallecidos,'

# Consulta para obtener todas las muertes a nivel nacional
sql_query_muertes_mundiales = 'SELECT * FROM casos_mundiales;'

# Consulta para obtener la población total por departamento
sql_query_poblacion_departamento = 'SELECT D.codigo_departamento AS codigoDepartame

# Ejecutar la consulta y cargar los resultados en DataFrames
df_municipality = pd.read_sql(sql_query_municipios, engine)
df_department = pd.read_sql(sql_query_departamentos, engine)
df_municipality_deaths = pd.read_sql(sql_query_fallecidos_municipio, engine)
df_country_deaths = pd.read_sql(sql_query_muertes_mundiales, engine)
df_department_population = pd.read_sql(sql_query_poblacion_departamento, engine)

# Asegurar que la columna 'fecha' sea de tipo datetime64[ns]
df_country_deaths['fecha'] = pd.to_datetime(df_country_deaths['fecha'])

# Filtrar registros directamente en el mismo DataFrame
df_country_deaths = df_country_deaths[df_country_deaths['fecha'] > '2020-03-15']

# Agrupar datos por municipio para calcular métricas necesarias
df_municipality_aggregated = df_municipality_deaths.groupby('id_municipio').agg(
    total_muertes_acumuladas=('fallecidos', 'sum'),
    promedio_nuevas_muertes=('fallecidos', 'mean')
).reset_index()
```

```
# Combinar con la información de municipios (nombre y población)
df_municipality_data = df_municipality.merge(
    df_municipality_aggregated,
    left_on='codigo_municipio',
    right_on='id_municipio'
)[['nombre_municipio', 'poblacion', 'promedio_nuevas_muertes', 'total_muertes_acumu
```

Tablas

Municipios

In [4]: `df_municipality.head(4)`
`df_municipality.tail(4)`

Out[4]:

	codigo_municipio	nombre_municipio	poblacion	id_departamento
333	908	SAN MIGUEL SIGÜILA	9607	9
334	715	SAN PABLO LA LAGUNA	7950	7
335	1605	TAMAHU	23858	16
336	607	SAN JUAN TECUACO	12599	6

Departamentos

In [5]: `df_department.head(4)`
`df_department.tail(4)`

Out[5]:

	codigo_departamento	nombre_departamento
18	14	QUICHE
19	6	SANTA ROSA
20	22	JUTIAPA
21	21	JALAPA

Muertes por Municipio

In [6]: `df_municipality_deaths.head(4)`
`df_municipality_deaths.tail(4)`

Out[6]:

	id_municipio	fecha	fallecidos	casos_mundiales	nombre_municipio	nombre_depart
18868	1901	2020-12-28	0	138	ZACAPA	
18869	1901	2020-12-29	0	132	ZACAPA	
18870	1901	2020-12-30	0	846	ZACAPA	
18871	1901	2020-12-31	0	879	ZACAPA	

Muertes a Nivel Nacional

In [7]:

```
df_country_deaths.head(4)
df_country_deaths.tail(4)
```

Out[7]:

	cod_pais	fecha	new_cases	cumulative_cases	new_death	cumulative_death
52	GT	2020-12-28	138	135309	5	4768
53	GT	2020-12-29	132	135441	5	4773
54	GT	2020-12-30	846	136287	8	4781
55	GT	2020-12-31	879	137166	22	4803

Poblacion Departamental

In [8]:

```
df_department_population.tail(4)
```

Out[8]:

	codigodedepartamento	nombredepartamento	poblaciontotal
18	19	ZACAPA	267718
19	20	CHIQUIMULA	441579
20	21	JALAPA	394234
21	22	JUTIAPA	546005

EDA Monovariable

Analisis de Nuevas Muertes

Observaciones:

1. El valor promedio de nuevas muertes, es de 19.25 aproximado 19.
2. El valor de la desviación estandar es de 12.53, se puede observar una variabilidad significativa en los datos. Indica que las muertes nuevas varian considerablemente alrededor del valor promedio
3. El número mínimo de muertes nuevas es 2, mientras que el máximo es 61. Esto refleja un rango bastante amplio, con valores muy bajos y muy altos, lo que puede indicar que algunos periodos han tenido eventos excepcionales (por ejemplo, picos de muertes).

In [9]:

```
# Estadísticas Descriptivas
describe = df_country_deaths['new_death'].describe()
print(f"Estadísticas descriptivas\n{describe}\n")

# Estadísticas de Conteo
count = df_country_deaths['new_death'].count()
print(f"Estadísticas de Conteo\n{count}\n")

# Valores únicos
unique = df_country_deaths['new_death'].unique()
nunique = df_country_deaths['new_death'].nunique()
print(f"Valores Únicos\n{unique}")
print(f"Valores Únicos (cantidad): {nunique}\n")
# Promedio
mean = df_country_deaths['new_death'].mean()
print(f"Promedio\n{mean}\n")

# Cuartiles
# Mediana (50%)
median_new_deaths = df_country_deaths['new_death'].median()

# Cuartiles (25%, 75%)
q1_new_deaths = df_country_deaths['new_death'].quantile(0.25)
q3_new_deaths = df_country_deaths['new_death'].quantile(0.75)
iqr_new_deaths = q3_new_deaths - q1_new_deaths

print("\nCuartil 25% (Q1):", q1_new_deaths)
print("Cuartil 75% (Q3):", q3_new_deaths)
print("Rango intercuartílico (IQR):", iqr_new_deaths)

# Límites para identificar outliers
lower_bound = q1_new_deaths - 1.5 * iqr_new_deaths
upper_bound = q3_new_deaths + 1.5 * iqr_new_deaths

print("\nLímite inferior para outliers:", lower_bound)
print("Límite superior para outliers:", upper_bound)
```

Estadísticas descriptivas

```

count      56.00000
mean      19.25000
std       12.52815
min       2.00000
25%       8.00000
50%       17.00000
75%       26.75000
max       61.00000
Name: new_death, dtype: float64

```

Estadísticas de Conteo

56

Valores Únicos

```
[ 3 23 20 25 37 15 11  5 21 13 14 35  7 17 22 10 40 12  6  9 61 42  2 16
 8 26 29 18 31 34 41 38 32 30]
```

Valores Únicos (cantidad): 34

Promedio

19.25

Cuartil 25% (Q1): 8.0

Cuartil 75% (Q3): 26.75

Rango intercuartílico (IQR): 18.75

Límite inferior para outliers: -20.125

Límite superior para outliers: 54.875

```
In [10]: # Identificar outliers
outliers = df_country_deaths[(df_country_deaths['new_death'] < lower_bound) |
                               (df_country_deaths['new_death'] > upper_bound)]
print("\nValores atípicos (outliers):")
print(outliers)
```

Filtrar datos sin outliers

```
filtered_data = df_country_deaths[
    (df_country_deaths['new_death'] >= lower_bound) &
    (df_country_deaths['new_death'] <= upper_bound)
]
```

Valores atípicos (outliers):

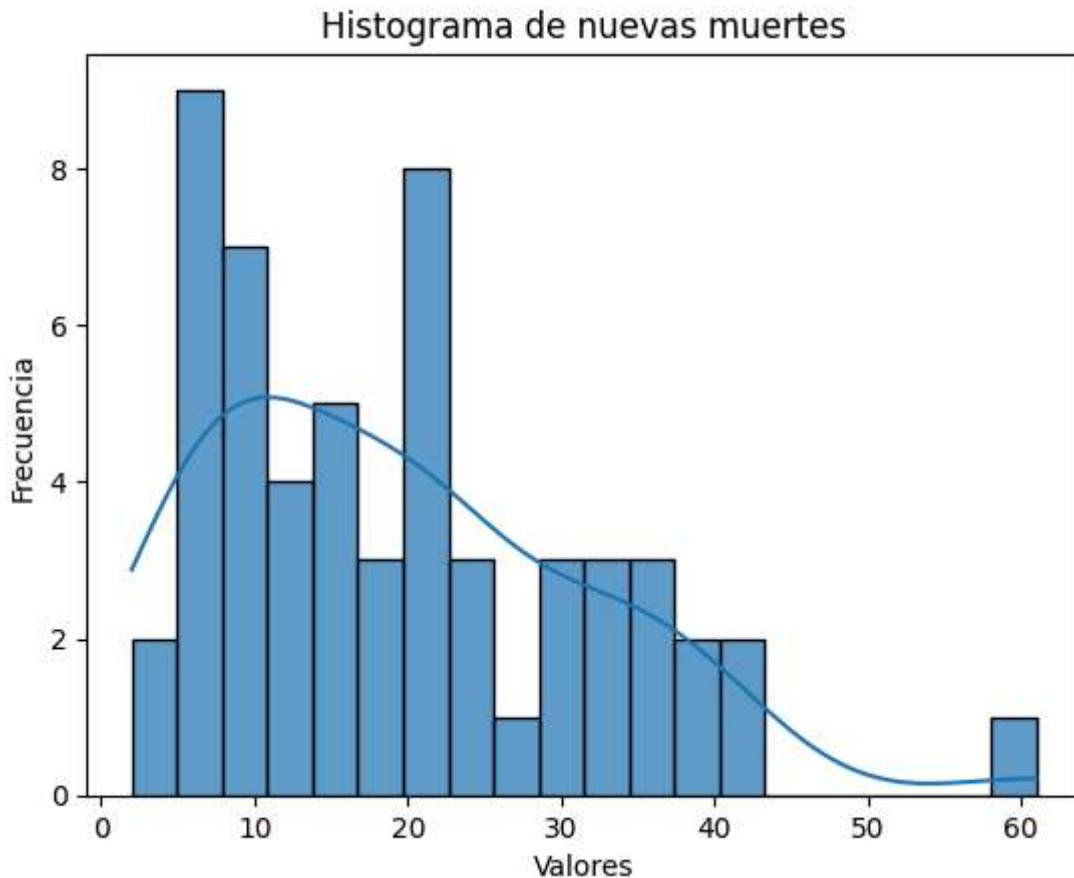
cod_pais	fecha	new_cases	cumulative_cases	new_death	\
25	GT 2020-11-19	651	116381	61	
				cumulative_death	
25				4008	

Filtramos todos los registros de new_death que sean menores o iguales a 0. Por que las muertes reportadas como 0 no tienen significado para un analisis logaritmico.

Para el boxplot de Nuevas Muertes, se decidio dejar los outliers, ya que ayudan a identificar picos significativos, es decir picos o cambios significativos en la tendencia de fallecimientos

```
In [ ]: # Filtra los valores mayores que cero
filtered_data = df_country_deaths['new_death'][df_country_deaths['new_death'] > 0]

# Crea el histograma con la transformación Logarítmica
sns.histplot(filtered_data, bins=20, kde=True, edgecolor='black', alpha=0.7)
plt.xlabel('Valores')
plt.ylabel('Frecuencia')
plt.title('Histograma de nuevas muertes')
plt.show()
```

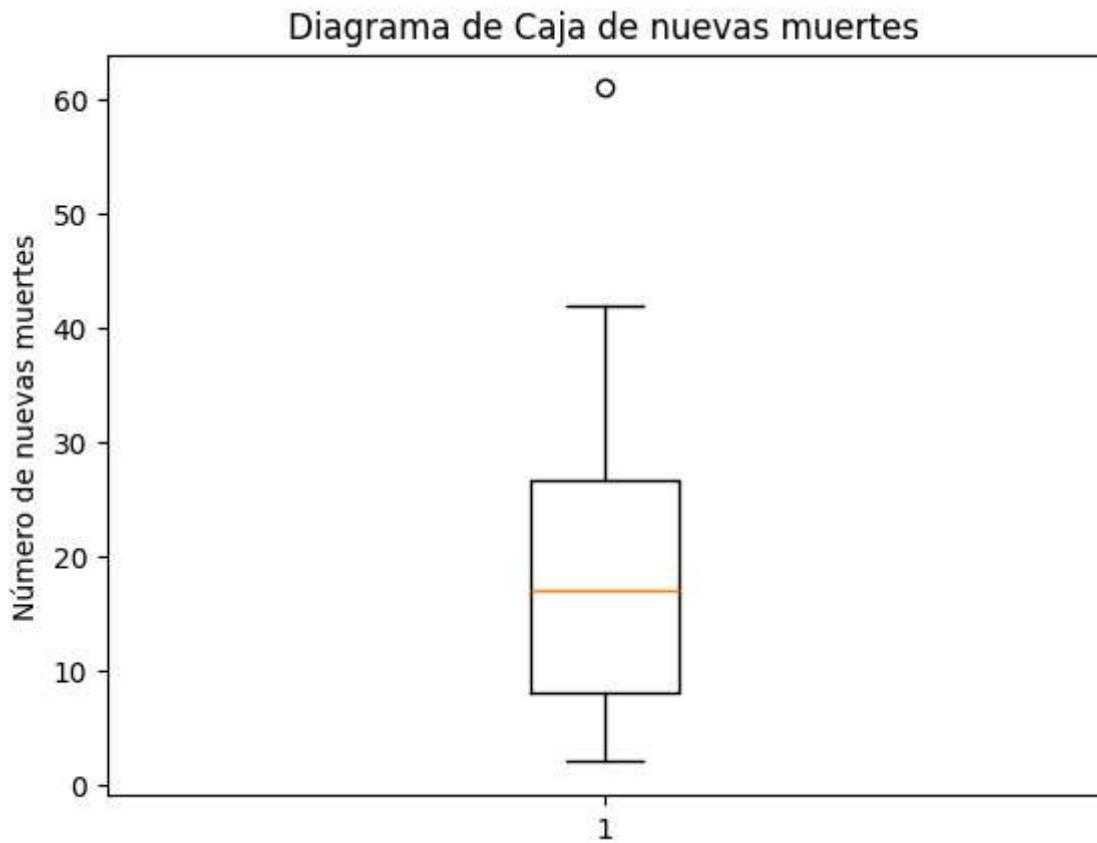


Observaciones:

- La mayoría de los días se registraron entre 8 y 26 nuevas muertes.
- Existe un día en el que se registró un número inusualmente alto de muertes (61). Esto se considera un outlier, se consideró dejar el dato ya que esto puede significar un pico de muertes, es decir el día con más muertes de COVID-19
- La distribución de las nuevas muertes presenta una ligera asimetría positiva.

```
In [12]: plt.boxplot(df_country_deaths['new_death'])

plt.title('Diagrama de Caja de nuevas muertes')
plt.ylabel('Número de nuevas muertes')
plt.show()
```



Analisis de Cantidad de Muertes Acumuladas

Observaciones

- En la variable de muertes acumuladas se puede ver una mejor distribución de los datos, ya que los cuartiles están un poco más proporcionados exceptuando los datos del primer cuartil. Esto se puede deber a que por mucho tiempo la población Guatemalteca acató medidas de prevención de tal forma que no hubieron muchos contagios al inicio.
- El promedio es de 4081 un valor que puede ser considerado muy alto, sin embargo, debido a que se trata de estadísticas acumulativas, es entendible que este valor sea alto.
- El total de registros es de 56 los cuales concuerdan con los días en que se registraron muertes por covid y aquellos datos que pasaron la limpieza de datos hecha anteriormente

```
In [78]: # Estadísticas Descriptivas
describe = df_country_deaths['cumulative_death'].describe()
print(f"Estadísticas descriptivas\n{describe}\n")

count = df_country_deaths['cumulative_death'].count()
print(f"Estadísticas de Conteo\n{count}\n")

unique = df_country_deaths['cumulative_death'].nunique()
print(f"Valores Únicos\n{unique}\n")

# Promedio
```

```

mean = df_country_deaths['cumulative_death'].mean()
print(f"Promedio\n{mean}\n")

# Mediana (50%)
median_new_deaths = df_country_deaths['cumulative_death'].median()

# Cuartiles (25%, 75%)
q1_new_deaths = df_country_deaths['cumulative_death'].quantile(0.25)
q3_new_deaths = df_country_deaths['cumulative_death'].quantile(0.75)
iqr_new_deaths = q3_new_deaths - q1_new_deaths
print("\nMediana para 'cumulative_deaths':", median_new_deaths)
print("\nCuartil 25% para 'cumulative_deaths':", q1_new_deaths)
print("\nCuartil 75% para 'cumulative_deaths':", q3_new_deaths)
print("Rango intercuartílico (IQR):", iqr_new_deaths)

lower_bound = q1_new_deaths - 1.5 * iqr_new_deaths
upper_bound = q3_new_deaths + 1.5 * iqr_new_deaths

print("\nLímite inferior para outliers:", lower_bound)
print("Límite superior para outliers:", upper_bound)

```

Estadísticas descriptivas

count	56.000000
mean	4081.553571
std	463.511495
min	3387.000000
25%	3649.250000
50%	4070.500000
75%	4520.250000
max	4803.000000
Name:	cumulative_death, dtype: float64

Estadísticas de Conteo

56

Valores Únicos

56

Promedio

4081.5535714285716

Mediana para 'cumulative_deaths': 4070.5

Cuartil 25% para 'cumulative_deaths': 3649.25

Cuartil 75% para 'cumulative_deaths': 4520.25

Rango intercuartílico (IQR): 871.0

Límite inferior para outliers: 2342.75

Límite superior para outliers: 5826.75

In [14]: # Identificar outliers

```

outliers = df_country_deaths[(df_country_deaths['cumulative_death'] < lower_bound)
                             | (df_country_deaths['cumulative_death'] > upper_bound)]
print("\nValores atípicos (outliers):")

```

```

print(outliers)

# Filtrar datos sin outliers
filtered_data = df_country_deaths[
    (df_country_deaths['cumulative_death'] >= lower_bound) &
    (df_country_deaths['cumulative_death'] <= upper_bound)
]

```

Valores atípicos (outliers):

Empty DataFrame

Columns: [cod_pais, fecha, new_cases, cumulative_cases, new_death, cumulative_death]

Index: []

Filtramos todos los registros de cumulative_death que sean menores o iguales a 0. Por que las muertes reportadas como 0 no tienen significado para un analisis logaritmico.

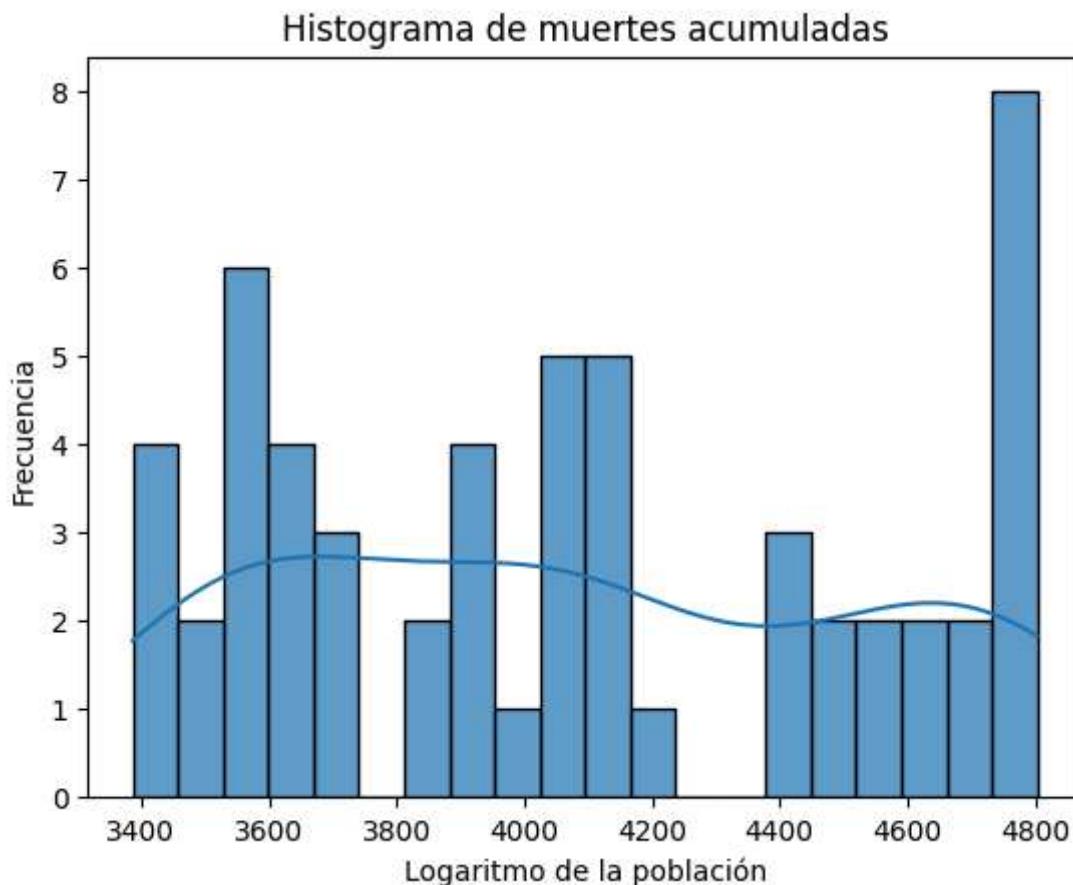
Para el boxplot de Muertes Acumuladas, no existen outliers entonces las graficas se generan con los datos existentes

```

In [ ]: # Filtra los valores mayores que cero
filtered_data = df_country_deaths['cumulative_death'][df_country_deaths['cumulative_death'] > 0]

sns.histplot(filtered_data, bins=20, kde=True, edgecolor='black', alpha=0.7)
plt.xlabel('Logaritmo de la población')
plt.ylabel('Frecuencia')
plt.title('Histograma de muertes acumuladas')
plt.show()

```

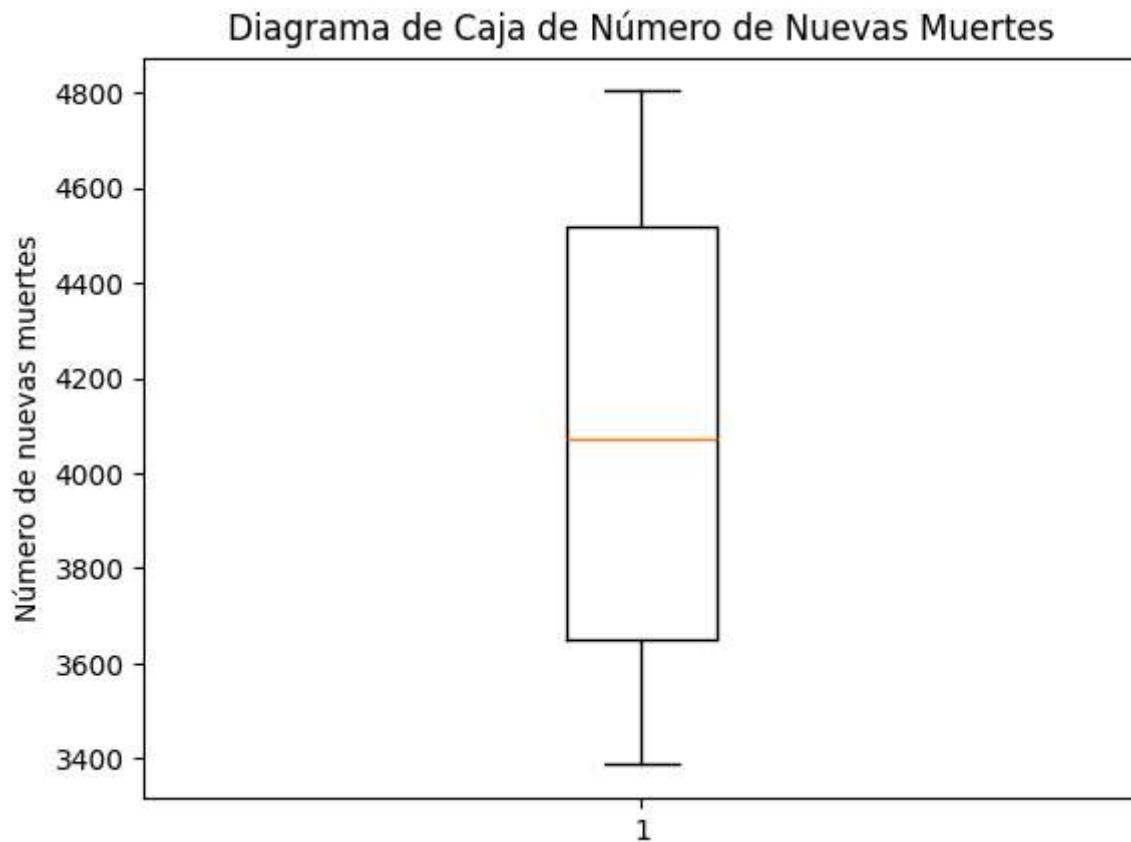


Observaciones:

- La mayoría de los municipios presentan un número de nuevas muertes relativamente similar, agrupado alrededor de la mediana de 4000.
- La ausencia de valores atípicos indica que no hay municipios con un número de nuevas muertes significativamente mayor o menor que el resto.
- Esta grafica representa el numero de nuevas muertes a nivel nacional, por ende los datos representan un nvel de agregacion alto.

```
In [16]: plt.boxplot(df_country_deaths['cumulative_death'])
```

```
plt.title('Diagrama de Caja de Número de Nuevas Muertes')
plt.ylabel('Número de nuevas muertes')
plt.show()
```



Analisis de Poblacion en los Municipios

Observaciones

- Probablemente la estadística más controversial, ya que los valores tienen una separación muy grande. Esto se debe a la población en el país de Guatemala, un número muy grande de habitantes suele concentrarse en ciudades muy específicas como lo pueden ser la Ciudad de Guatemala, Mixco, Villa Nueva, etc.

- Si comparamos la población de los municipios anteriormente mencionados con algunos otros municipios del área rural, hay una diferencia de población muy grande. Y es importante tomar en cuenta que el área rural comprende la mayor parte del territorio de la República, por lo cual es entendible que tanto la gráfica de caja como la gráfica inicial del histograma posean dicha forma.
- Como se mencionó, la gráfica inicial que tenía el histograma tenía una forma extraña, a tal punto que la misma estaba sesgada. Por lo cual se tuvo que aplicar una **transformación logarítmica** para componer la visualización de la misma.
- De igual manera por lo que se menciono anteriormente, los outliers como vemos, corresponden a municipios con poblaciones significativamente mayores en comparacion con los demás municipios. Por lo cual, para nuestro analisis, no se eliminaron los outliers, por que son parte de nuestro analisis pero en la transformación logarítmica aplicada sobre los datos los valores extremadamente grandes se "comprimen" cuando se toma el logaritmo. Sin embargo, esto elimina nuestros outliers, sino que solo reduce su impacto.

In [17]:

```
# Estadísticas Descriptivas
describe = df_municipality['poblacion'].describe()
print(f"Estadísticas descriptivas\n{describe}\n")

# Estadísticas de Conteo
count = df_municipality['poblacion'].count()
print(f"Estadísticas de Conteo\n{count}\n")

# Valores únicos
unique = df_municipality['poblacion'].nunique()
print(f"Valores Únicos\n{unique}\n")

# Promedio
mean = df_municipality['poblacion'].mean()
print(f"Promedio\n{mean}\n")

# Cuartiles
# Mediana (50%)
median_new_deaths = df_municipality['poblacion'].median()

# Cuartiles (25%, 75%)
q1_new_deaths = df_municipality['poblacion'].quantile(0.25)
q3_new_deaths = df_municipality['poblacion'].quantile(0.75)
iqr_new_deaths = q3_new_deaths - q1_new_deaths

print("\nMediana para 'Poblacion':", median_new_deaths)
print("\nCuartil 25% para 'Poblacion':", q1_new_deaths)
print("\nCuartil 75% para 'Poblacion':", q3_new_deaths)
print("Rango intercuartílico (IQR):", iqr_new_deaths)

# Límites para identificar outliers
lower_bound = q1_new_deaths - 1.5 * iqr_new_deaths
```

```
upper_bound = q3_new_deaths + 1.5 * iqr_new_deaths

print("\nLímite inferior para outliers:", lower_bound)
print("Límite superior para outliers:", upper_bound)
```

Estadísticas descriptivas

count	3.370000e+02
mean	4.997144e+04
std	8.158246e+04
min	2.563000e+03
25%	1.746500e+04
50%	3.091200e+04
75%	5.852600e+04
max	1.205668e+06
Name:	poblacion, dtype: float64

Estadísticas de Conteo

337

Valores Únicos

336

Promedio

49971.44213649852

Mediana para 'Poblacion': 30912.0

Cuartil 25% para 'Poblacion': 17465.0

Cuartil 75% para 'Poblacion': 58526.0

Rango intercuartílico (IQR): 41061.0

Límite inferior para outliers: -44126.5

Límite superior para outliers: 120117.5

```
In [18]: # Identificar outliers
outliers = df_municipality[(df_municipality['poblacion'] < lower_bound) |
                           (df_municipality['poblacion'] > upper_bound)]
print("\nValores atípicos (outliers):")
print(outliers)

# Filtrar datos sin outliers
filtered_data = df_municipality[
    (df_municipality['poblacion'] >= lower_bound) &
    (df_municipality['poblacion'] <= upper_bound)
]
```

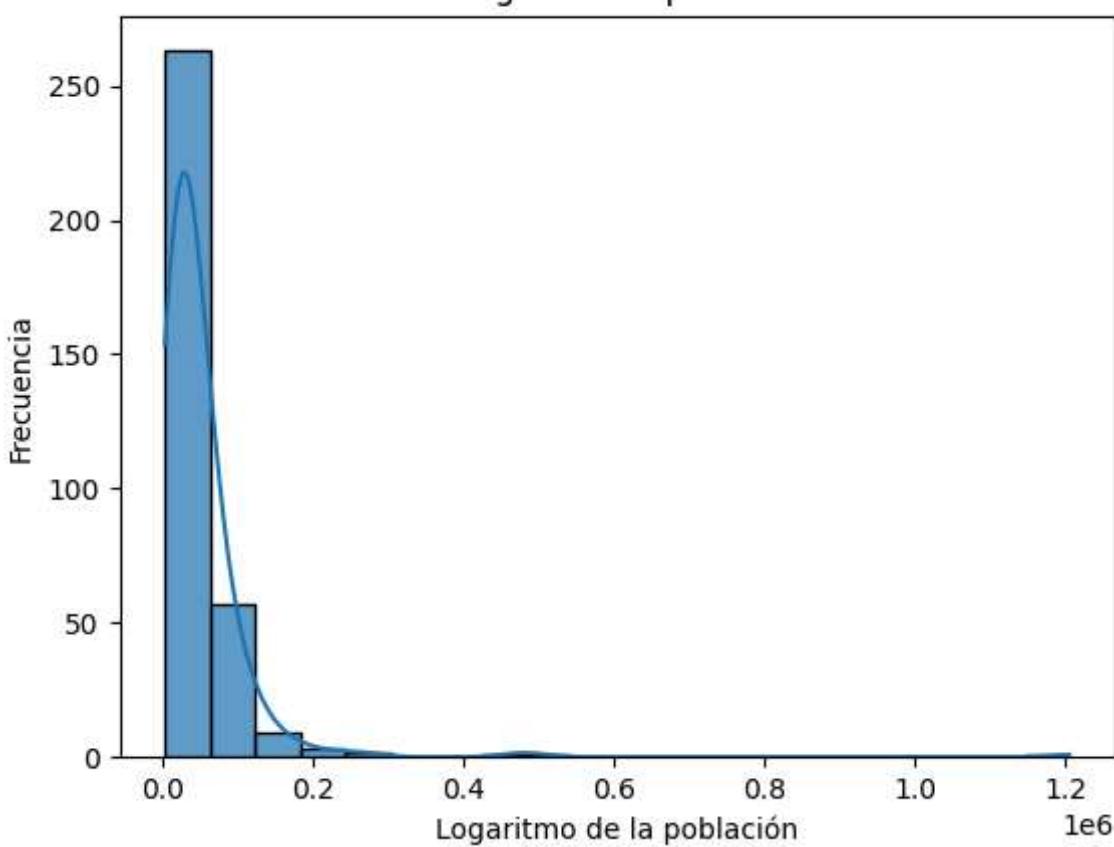
Valores atípicos (outliers):

	codigo_municipio	nombre_municipio	poblacion	id_departamento
2	101	GUATEMALA	1205668	1
8	501	ESCUINTLA	166078	5
14	115	VILLA NUEVA	464528	1
15	108	MIXCO	494561	1
17	502	SANTA LUCIA COTZUMALGUAPA	123979	5
18	1609	SAN PEDRO CARCHA	248880	16
19	116	VILLA CANALES	165026	1
23	901	QUETZALTENANGO	196867	9
26	117	SAN MIGUEL PETAPA	145417	1
29	114	AMATITLAN	147604	1
30	110	SAN JUAN SACATEPEQUEZ	276836	1
34	106	CHINAUTLA	121721	1
51	1406	CHICHICASTENANGO	148943	14
68	2201	JUTIAPA	163665	22
118	1601	COBAN	224109	16
143	2101	JALAPA	188522	21
161	1301	HUEHUETENANGO	129232	13
193	805	MOMOSTENANGO	136909	8
252	1326	BARILLAS	122466	13

```
In [19]: filtered_data = df_municipality['poblacion'][df_municipality['poblacion'] > 0]

# Crea el histograma con la transformación logarítmica
sns.histplot(filtered_data, bins=20, kde=True, edgecolor='black', alpha=0.7)
plt.xlabel('Logaritmo de la población')
plt.ylabel('Frecuencia')
plt.title('Histograma de población')
plt.show()
```

Histograma de población



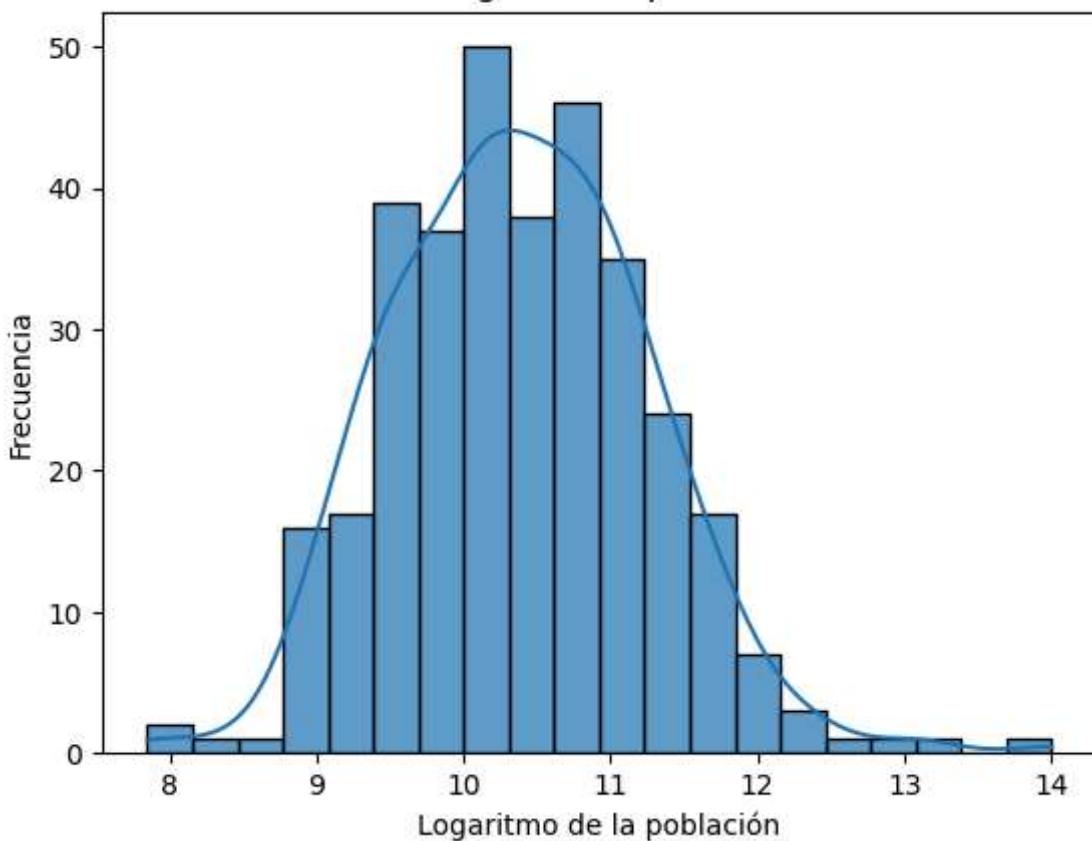
- El diagrama posee un sesgo, por lo cual, se hara una transformacion logaritmica

In [20]:

```
# Filtra los valores mayores que cero
filtered_data = df_municipality['poblacion'][df_municipality['poblacion'] > 0]

# Crea el histograma con la transformación Logarítmica
sns.histplot(np.log(filtered_data), bins=20, kde=True, edgecolor='black', alpha=0.7
plt.xlabel('Logaritmo de la población')
plt.ylabel('Frecuencia')
plt.title('Histograma de población')
plt.show()
```

Histograma de población

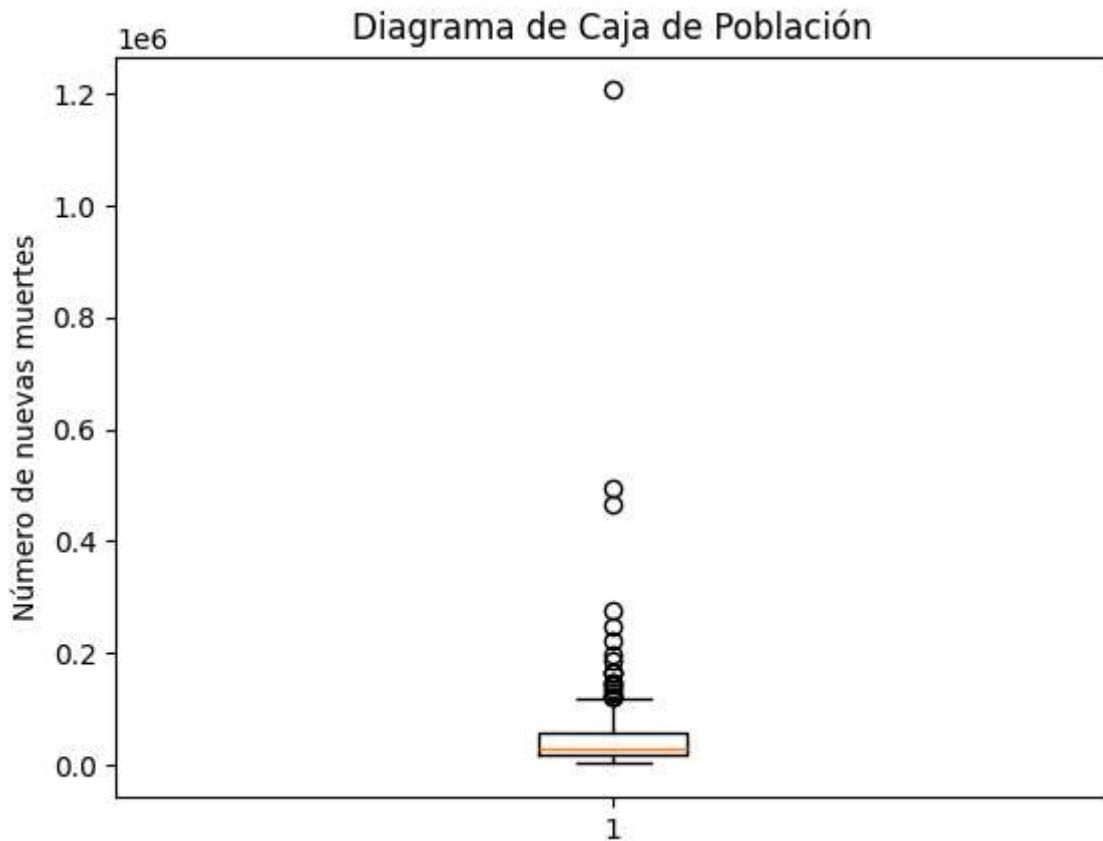


Observaciones:

- La presencia de outliers y el amplio rango de la caja indican una gran variabilidad en el tamaño de la población entre los municipios analizados. Hay municipios muy pequeños y otros mucho más grandes.
- La mayoría de los municipios tienen una población relativamente pequeña, como lo indica la mediana y el primer cuartil.
- Los outliers representan municipios con una población significativamente mayor que el promedio. Estos municipios segun la informacion del resto de graficas y calculos, son Guatemala, Mixco o Villa Nueva, que son municipios que generalmente tienen mayor concentracion de personas.

```
In [21]: plt.boxplot(df_municipality['poblacion'])

plt.title('Diagrama de Caja de Población')
plt.ylabel('Número de nuevas muertes')
plt.show()
```

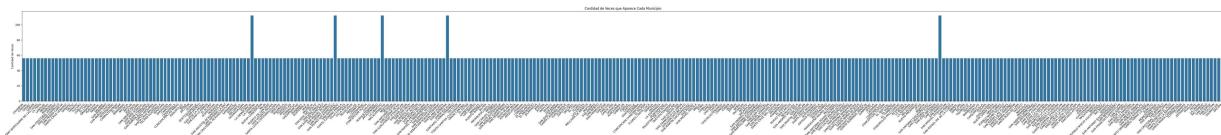


Diagramas de Barras para Variables cualitativas

Observaciones

- Importante tomar en cuenta que hay municipios como San Lorenzo, Santa Barbara, San Pedro Sacatepequez, La Libertad y la Democracia que aparecen más veces en los registros, esto puede influir en los resultados finales.
- Los departamentos que mas aparecen son Huehuetenango, San Marcos y Quetzaltenango, esto no debería representar una alteración en los datos ya que esto depende de la cantidad de municipios que cada departamento tenga.

```
In [22]: plt.figure(figsize=(80, 6))
sns.countplot(x='nombre_municipio', data=df_municipality_deaths)
plt.xticks(rotation=45, ha='right')
plt.title('Cantidad de Veces que Aparece Cada Municipio')
plt.xlabel('Municipio')
plt.ylabel('Cantidad de Veces')
plt.show()
```

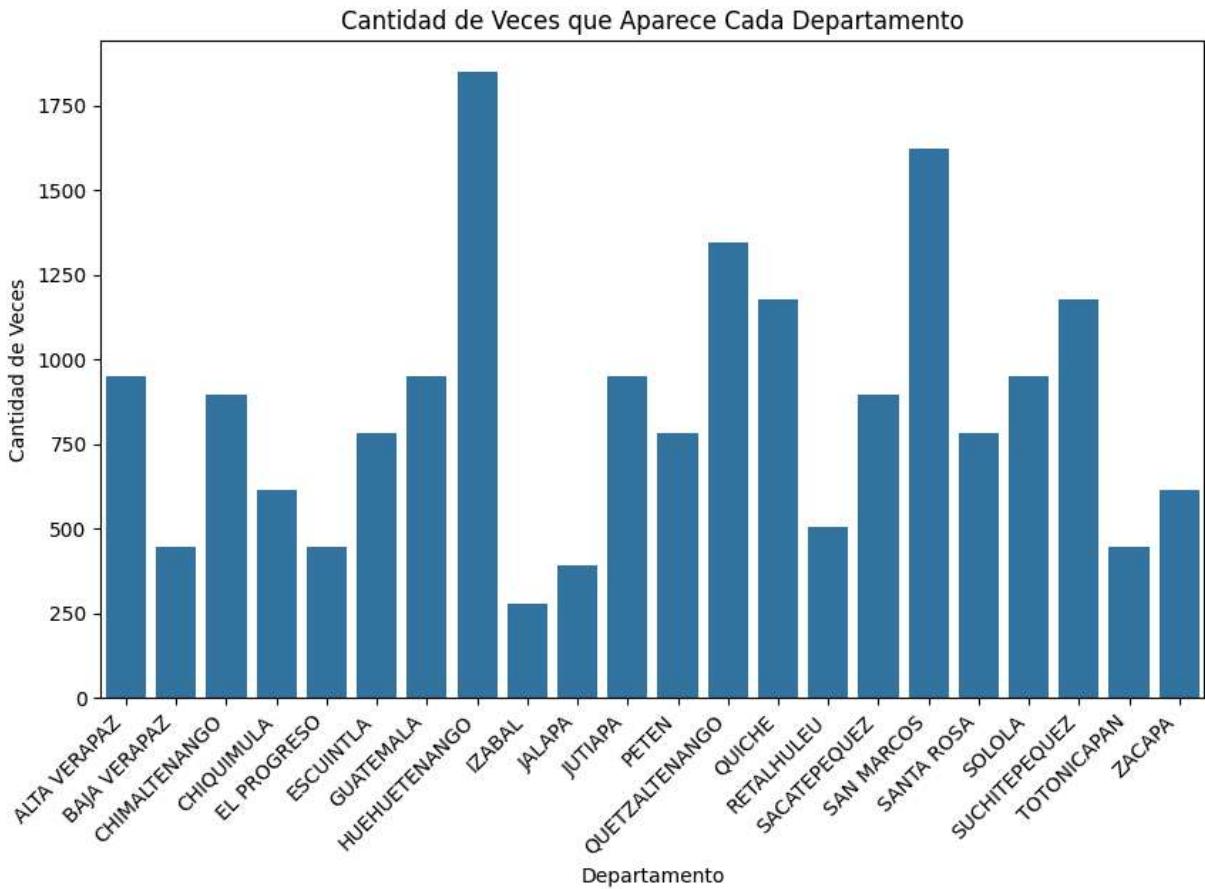


```
In [23]: # Gráfica de barras para departamento
plt.figure(figsize=(10, 6))
```

```

sns.countplot(x='nombre_departamento', data=df_municipality_deaths)
plt.xticks(rotation=45, ha='right')
plt.title('Cantidad de Veces que Aparece Cada Departamento')
plt.xlabel('Departamento')
plt.ylabel('Cantidad de Veces')
plt.show()

```



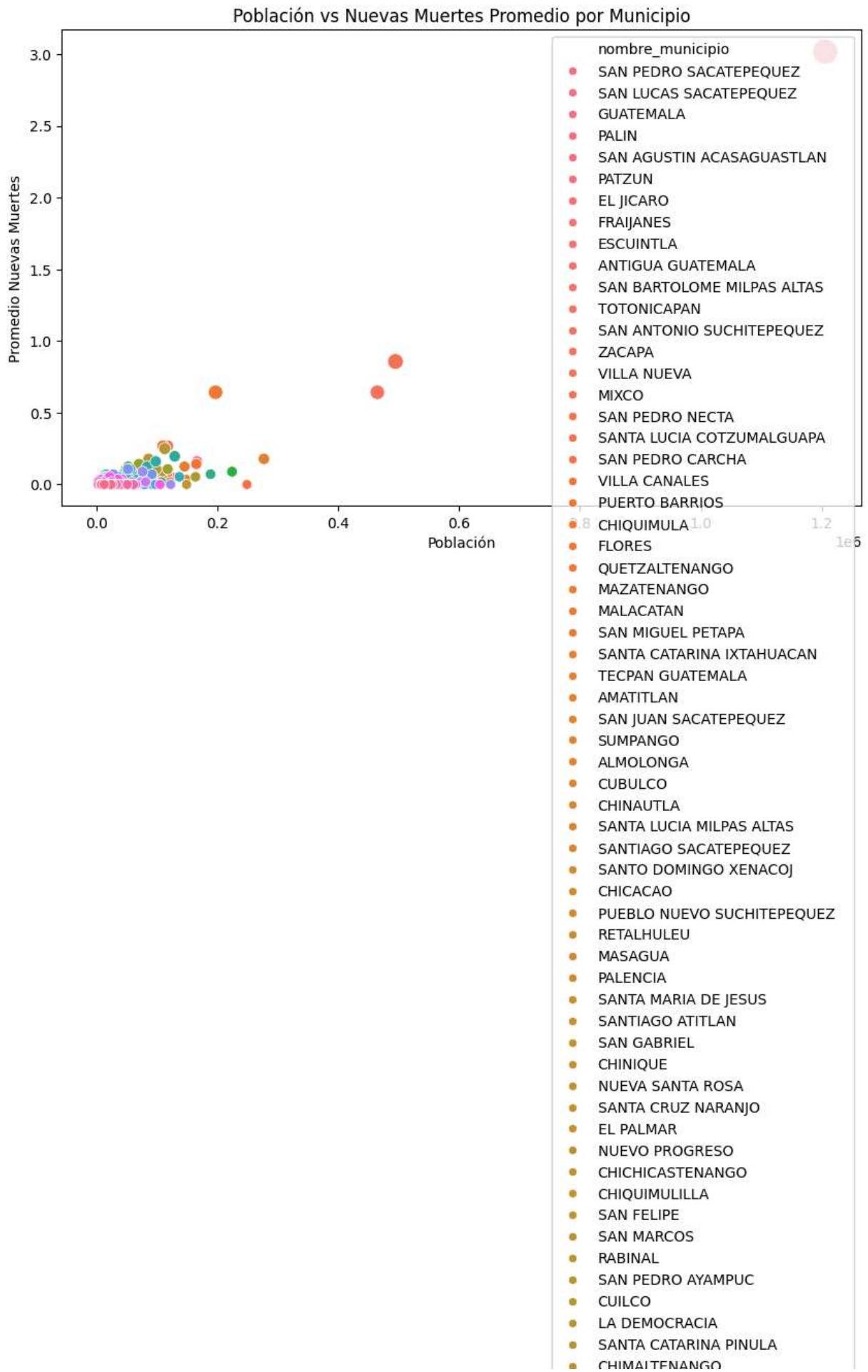
EDA Multivariable

**Graficas de Dispersion (Poblaciones vs Muertes
Prometio por Municipio y Poblacion vs Muertes
Acumuladas por Municipio)**

Observaciones

- En la grafica de dispersión, se puede observar que hay municipios con poca población que parecen tener un promedio de nuevas muertes bajo o cercano a cero. Esto puede ser esperado, ya que municipios más pequeños generalmente tienen menos casos y, por lo tanto, menos muertes en promedio. Hay algunos municipios con poblaciones más grandes que tienen un promedio de nuevas muertes relativamente alto. Esto podría ser indicativo de una mayor densidad poblacional o de otros factores de riesgo en esos lugares.

```
In [70]: # Gráfica 1: Población vs Nuevas Muertes Promedio
plt.figure(figsize=(10, 6))
sns.scatterplot(
    data=df_municipality_data,
    x='poblacion',
    y='promedio_nuevas_muertes',
    hue='nombre_municipio',
    size='promedio_nuevas_muertes',
    sizes=(50, 300)
)
plt.title('Población vs Nuevas Muertes Promedio por Municipio')
plt.xlabel('Población')
plt.ylabel('Promedio Nuevas Muertes')
plt.show()
```

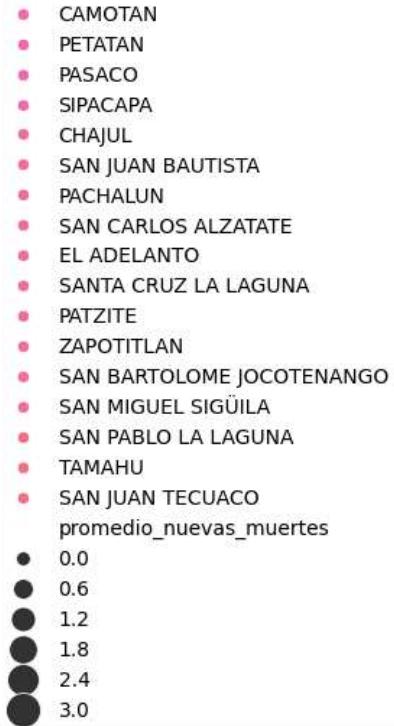


- NUEVA CONCEPCION
- SIPACATE
- SANTA MARIA IXHUATAN
- CAHABON
- SOLOLA
- CONCEPCION CHIQUIRICHAPA
- SAN JUAN ATTITAN
- JUTIAPA
- CHUARRANCHO
- SAN JOSE PINULA
- MAGDALENA MILPAS ALTAS
- OCOS
- YUPILTEPEQUE
- COATEPEQUE
- DOLORES
- SANSARE
- FLORES COSTA CUCA
- SAN PEDRO SOLOMA
- SAN ANTONIO LA PAZ
- SAN PABLO JOCOPILAS
- SALAMA
- CATARINA
- LA ESPERANZA
- AYUTLA
- PAJAPITA
- SAN LORENZO
- SAN ANDRES ITZAPA
- CUILAPA
- SAN RAFAEL PIE DE LA CUESTA
- MOYUTA
- SANTA CATARINA MITA
- PUEBLO NUEVO VIÑAS
- EL ASINTAL
- LIVINGSTON
- SANARATE
- BARBERENA
- SAN PABLO
- GUASTATOYA
- SAN MIGUEL DUEÑAS
- PUERTO DE SAN JOSE
- TAJUMULCO
- CHIANTLA
- SAN JUAN OSTUNCALCO
- SAN ANDRES VILLA SECA
- SAN BENITO
- PASTORES
- LA BLANCA
- SAN ANTONIO AGUAS CALIENTES
- RIO HONDO
- MATAQUESCUINTLA
- SAN FRANCISCO EL ALTO
- JOCOTENANGO
- ACATENANGO
- PARRAMOS
- COMITANCILLO
- CHICAMAN
- COBAN
- ALOTENANGO
- SAN JUAN LA LAGUNA
- TEJUTLA
- LAS CRUCES
- SAN VICENTE PACAYA
- ZUNIL

- SAN FRANCISCO ZAPOTITLAN
- SANTO DOMINGO SUCHITEPEQUEZ
- SAN JUAN CHAMELCO
- ZARAGOZA
- LA GOMERA
- SANTO TOMAS LA UNION
- SIQUINALA
- OLINTEPEQUE
- RIO BRAVO
- SANTA CRUZ VERAPAZ
- TACTIC
- MORALES
- CANTEL
- SAMAYAC
- SANTA BARBARA
- SANTA ROSA DE LIMA
- SAN CARLOS SIJA
- PATULUL
- JALAPA
- CHAMPERICO
- TECTITAN
- IXCAN
- SAN CRISTOBAL TOTONICAPAN
- COLOMBA
- CHISEC
- CIUDAD VIEJA
- EL TEJAR
- EL TUMBADOR
- QUEZADA
- TIQUISATE
- TAXISCO
- SAN BERNARDINO
- AGUACATAN
- QUETZALTEPEQUE
- PANAJACHEL
- CUYOTENANGO
- HUEHUETENANGO
- SAN CRISTOBAL VERAPAZ
- LA LIBERTAD
- SAN DIEGO
- SALCAJA
- EL CHAL
- HUITE
- TECULUTAN
- MORAZAN
- SAN MARTIN ZAPOTITLAN
- FRAY BARTOLOME DE LAS CASAS
- LOS AMATES
- SANTA CRUZ MULUA
- SAN RAYMUNDO
- NENTON
- SANTA CRUZ DEL QUICHE
- SAYAXCHE
- GUALAN
- YEPOCAPA
- NUEVO SAN CARLOS
- COLOTENANGO
- USUMATLAN
- GENOVA
- ORATORIO
- PATZICIA
- SAN ANDRES SEMETABAJ
- CABRICAN
- ESOUIPULAS

- GUANAGAZAPA
- GUAZACAPAN
- SACAPULAS
- MOMOSTENANGO
- EL RODEO
- JOCOTAN
- SAN PEDRO LA LAGUNA
- SAN ANDRES XECUL
- SAN MARTIN SACATEPEQUEZ
- SAN MATEO
- SAN RAFAEL PETZAL
- SAN PEDRO PINULA
- CONCEPCION TUTUAPA
- SAN LUCAS TOLIMAN
- EL QUETZAL
- SAN MARTIN JILOTEPEQUE
- SAN FRANCISCO LA UNION
- SAN ILDEFONSO IXTAHUACAN
- SAN PEDRO JOCOPILAS
- SAN JOSE EL IDOLO
- SAN ANTONIO SACATEPEQUEZ
- POPTUN
- RAXRUHA
- MELCHOR DE MENCOS
- SANTA CATARINA BARAHONA
- EL ESTOR
- ZUNILITO
- ESQUIPULAS PALO GORDO
- ESTANZUELA
- POCHUTA
- SAN SEBASTIAN
- SAN MIGUEL CHICAJ
- SAN JUAN COMALAPA
- SANTA ANA
- ATESCATEMPA
- IPALA
- COMAPA
- LA UNION
- SANTA LUCIA LA REFORMA
- CANILLA
- SAN JOSE CHACAYA
- SENAHU
- SAN LUIS
- PANZOS
- SAN JORGE
- SAN ANTONIO PALOPO
- NAHUALA
- SANTA ANA HUISTA
- SAN RAFAEL LA INDEPENDENCIA
- ASUNCION MITA
- SANTA CLARA LA LAGUNA
- LANQUIN
- TUCURU
- SANTIAGO CHIMALTENANGO
- SAN JERONIMO
- CHICHE
- NEBAJ
- SIBILIA
- SAN MIGUEL ACATAN
- EL PROGRESO
- CAJOLA
- BARILLAS
- TODOS SANTOS CUCHUMATAN

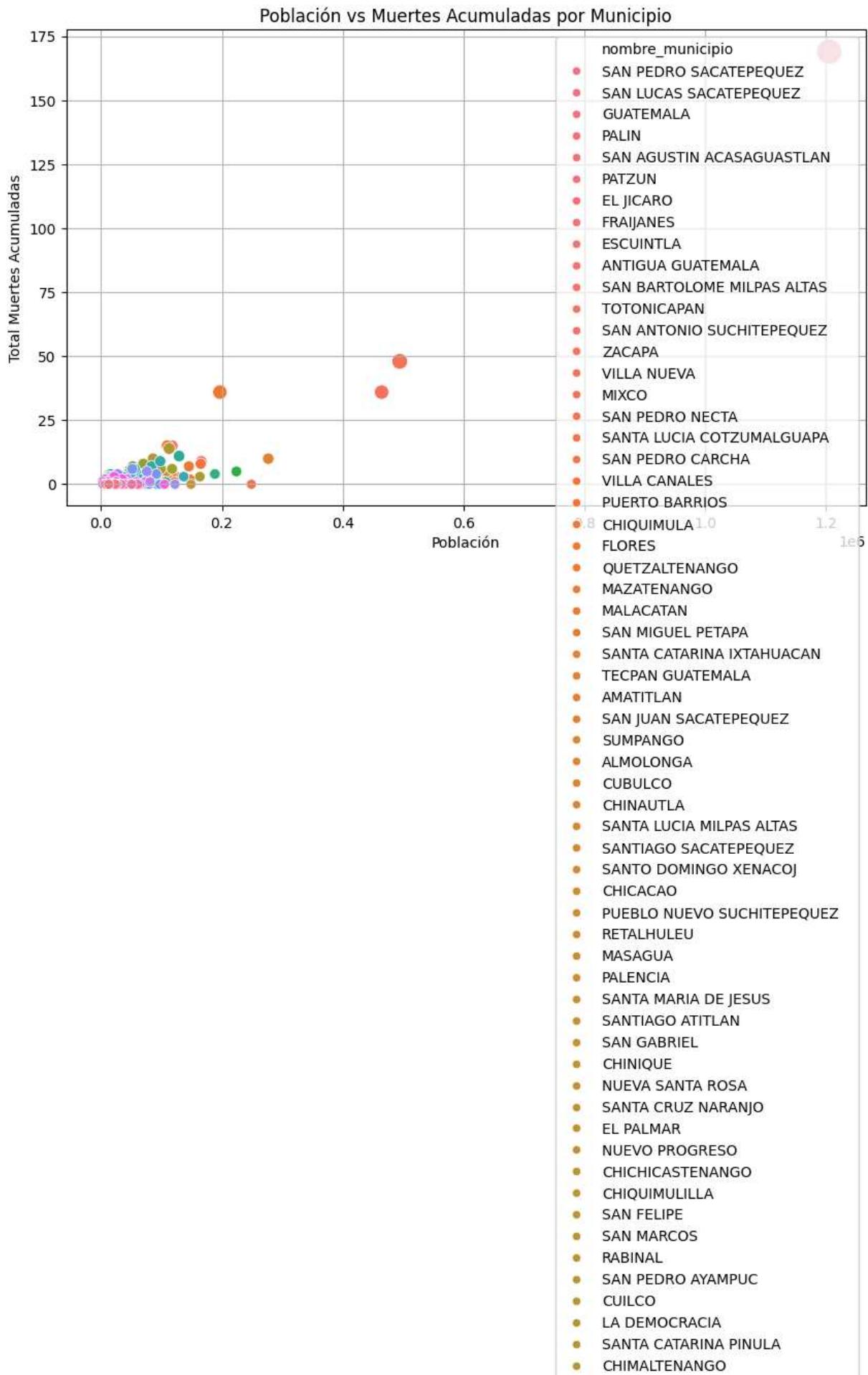
- CHAHAL
- SAN ANTONIO ILOTENANGO
- GRANADOS
- LA REFORMA
- SANTA EULALIA
- LA TINTA
- SAN LUIS JILOTEPEQUE
- SANTA MARIA CHIQUIMULA
- CUNEN
- SAN JUAN COTZAL
- SAN CRISTOBAL CUCHO
- SAN MIGUEL USPANTAN
- TACANA
- JALPATAGUA
- SAN JOSE
- EL CHOL
- SAN MIGUEL IXTAHUACAN
- UNION CANTINIL
- SAN JUAN IXCOY
- CABANAS
- SAN JOSE OJETENAM
- CONCEPCION HUISTA
- SAN ANDRES SAJCABAJA
- JEREZ
- SAN MATEO IXTATAN
- AGUA BLANCA
- HUITAN
- PALESTINA DE LOS ALTOS
- JACALTENANGO
- PURULHA
- IZTAPA
- SIBINAL
- SAN SEBASTIAN HUEHUETENANGO
- SAN ANDRES
- SANTA LUCIA UTATLAN
- IXCHIGUAN
- SAN GASPAR IXCHIL
- SAN JOSE ACATEMPA
- OLOPA
- SANTA APOLONIA
- JOYABAJ
- SANTA CRUZ BALANYA
- SANTA MARIA VISITACION
- SAN SEBASTIAN COATAN
- SAN ANTONIO HUISTA
- MALACATANCITO
- SAN BARTOLO
- SAN JACINTO
- CASILLAS
- SAN MARCOS LA LAGUNA
- MONJAS
- CONCEPCION LAS MINAS
- SAN JOSE LA ARADA
- CONGUACO
- SAN JUAN LA ERMITA
- SAN CRISTOBAL ACASAGUASTLAN
- SAN JOSE DEL GOLFO
- SAN FRANCISCO
- SAN MIGUEL PANAN
- SAN MANUEL CHAPARRON
- SAN JOSE LA MAQUINA
- ZACUALPA
- SAN JOSE POAQUIL
- SAN RAFAEL LAS FLORES



Observaciones

- Como se esperaba, municipios más grandes parecen tener un número más alto de muertes acumuladas en general, aunque no de manera uniforme. Por ejemplo, algunos municipios grandes, como Guatemala, pueden tener valores más altos en muertes acumuladas.
- Algunos puntos cercanos a la parte inferior del gráfico, con poblaciones bajas, tienen pocos casos de muertes acumuladas, lo cual es coherente con lo esperado en áreas menos densamente pobladas.

```
In [71]: # Gráfica 2: Población vs Muertes Acumuladas
plt.figure(figsize=(10, 6))
sns.scatterplot(
    data=df_municipality_data,
    x='poblacion',
    y='total_muertes_acumuladas',
    hue='nombre_municipio',
    size='total_muertes_acumuladas',
    sizes=(50, 300)
)
plt.title('Población vs Muertes Acumuladas por Municipio')
plt.xlabel('Población')
plt.ylabel('Total Muertes Acumuladas')
plt.grid(True)
plt.show()
```

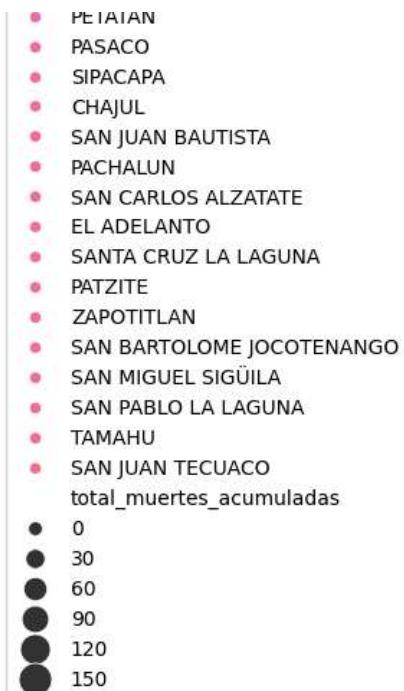


- NUEVA CONCEPCION
- SIPACATE
- SANTA MARIA IXHUATAN
- CAHABON
- SOLOLA
- CONCEPCION CHIQUIRICHA
- SAN JUAN ATTAN
- JUTIAPA
- CHUARRANCHO
- SAN JOSE PINULA
- MAGDALENA MILPAS ALTAS
- OCOS
- YUPILTEPEQUE
- COATEPEQUE
- DOLORES
- SANSARE
- FLORES COSTA CUCA
- SAN PEDRO SOLOMA
- SAN ANTONIO LA PAZ
- SAN PABLO JOCOPILAS
- SALAMA
- CATARINA
- LA ESPERANZA
- AYUTLA
- PAJAPITA
- SAN LORENZO
- SAN ANDRES ITZAPA
- CUILAPA
- SAN RAFAEL PIE DE LA CUESTA
- MOYUTA
- SANTA CATARINA MITA
- PUEBLO NUEVO VIÑAS
- EL ASINTAL
- LIVINGSTON
- SANARATE
- BARBERENA
- SAN PABLO
- GUASTATOYA
- SAN MIGUEL DUEÑAS
- PUERTO DE SAN JOSE
- TAJUMULCO
- CHIANTLA
- SAN JUAN OSTUNCALCO
- SAN ANDRES VILLA SECA
- SAN BENITO
- PASTORES
- LA BLANCA
- SAN ANTONIO AGUAS CALIENTES
- RIO HONDO
- MATAQUESCUINTLA
- SAN FRANCISCO EL ALTO
- JOCOTENANGO
- ACATENANGO
- PARRAMOS
- COMITANCILLO
- CHICAMAN
- COBAN
- ALOTENANGO
- SAN JUAN LA LAGUNA
- TEJUTLA
- LAS CRUCES
- SAN VICENTE PACAYA
- ZUNIL
- SAN FRANCISCO ZAPOTITLAN

- SANTO DOMINGO SUCHITEPEQUEZ
- SAN JUAN CHAMELCO
- ZARAGOZA
- LA GOMERA
- SANTO TOMAS LA UNION
- SIQUINALA
- OLINTEPEQUE
- RIO BRAVO
- SANTA CRUZ VERAPAZ
- TACTIC
- MORALES
- CANTEL
- SAMAYAC
- SANTA BARBARA
- SANTA ROSA DE LIMA
- SAN CARLOS SIJA
- PATULUL
- JALAPA
- CHAMPERICO
- TECTITAN
- IXCAN
- SAN CRISTOBAL TOTONICAPAN
- COLOMBA
- CHISEC
- CIUDAD VIEJA
- EL TEJAR
- EL TUMBADOR
- QUEZADA
- TIQUISATE
- TAXISCO
- SAN BERNARDINO
- AGUACATAN
- QUETZALTEPEQUE
- PANAJACHEL
- CUYOTENANGO
- HUEHUETENANGO
- SAN CRISTOBAL VERAPAZ
- LA LIBERTAD
- SAN DIEGO
- SALCAJA
- EL CHAL
- HUISTE
- TECULUTAN
- MORAZAN
- SAN MARTIN ZAPOTITLAN
- FRAY BARTOLOME DE LAS CASAS
- LOS AMATES
- SANTA CRUZ MULUA
- SAN RAYMUNDO
- NENTON
- SANTA CRUZ DEL QUICHE
- SAYAXCHE
- GUALAN
- YEPOCAPA
- NUEVO SAN CARLOS
- COLOTENANGO
- USUMATLAN
- GENOVA
- ORATORIO
- PATZICIA
- SAN ANDRES SEMETABAJ
- CABRICAN
- ESQUIPULAS
- GUANAGAZAPA

- GUAZACAPAN
- SACAPULAS
- MOMOSTENANGO
- EL RODEO
- JOCOTAN
- SAN PEDRO LA LAGUNA
- SAN ANDRES XECUL
- SAN MARTIN SACATEPEQUEZ
- SAN MATEO
- SAN RAFAEL PETZAL
- SAN PEDRO PINULA
- CONCEPCION TUTUAPA
- SAN LUCAS TOLIMAN
- EL QUETZAL
- SAN MARTIN JILOTEPEQUE
- SAN FRANCISCO LA UNION
- SAN ILDEFONSO IXTAHUACAN
- SAN PEDRO JOCOPILAS
- SAN JOSE EL IDOLO
- SAN ANTONIO SACATEPEQUEZ
- POPTUN
- RAXRUHA
- MELCHOR DE MENCOS
- SANTA CATARINA BARAHONA
- EL ESTOR
- ZUNILITO
- ESQUIPULAS PALO GORDO
- ESTANZUELA
- POCHUTA
- SAN SEBASTIAN
- SAN MIGUEL CHICAJ
- SAN JUAN COMALAPA
- SANTA ANA
- ATESCATEMPA
- IPALA
- COMAPA
- LA UNION
- SANTA LUCIA LA REFORMA
- CANILLA
- SAN JOSE CHACAYA
- SENAHU
- SAN LUIS
- PANZOS
- SAN JORGE
- SAN ANTONIO PALOPO
- NAHUALA
- SANTA ANA HUISTA
- SAN RAFAEL LA INDEPENDENCIA
- ASUNCION MITA
- SANTA CLARA LA LAGUNA
- LANQUIN
- TUCURU
- SANTIAGO CHIMALTENANGO
- SAN JERONIMO
- CHICHE
- NEBAJ
- SIBILIA
- SAN MIGUEL ACATAN
- EL PROGRESO
- CAJOLA
- BARILLAS
- TODOS SANTOS CUCHUMATAN
- CHAHAL
- SAN ANTONIO ILotenango

- SAN ANTONIO HUEHUETENANGO
- GRANADOS
- LA REFORMA
- SANTA EULALIA
- LA TINTA
- SAN LUIS JILOTEPEQUE
- SANTA MARIA CHIQUIMULA
- CUNEN
- SAN JUAN COTZAL
- SAN CRISTOBAL CUCHO
- SAN MIGUEL USPANTAN
- TACANA
- JALPATAGUA
- SAN JOSE
- EL CHOL
- SAN MIGUEL IXTAHUACAN
- UNION CANTINIL
- SAN JUAN IXCOY
- CABANAS
- SAN JOSE OJETENAM
- CONCEPCION HUISTA
- SAN ANDRES SAJCABAJA
- JEREZ
- SAN MATEO IXTATAN
- AGUA BLANCA
- HUITAN
- PALESTINA DE LOS ALTOS
- JACALTENANGO
- PURULHA
- IZTAPA
- SIBINAL
- SAN SEBASTIAN HUEHUETENANGO
- SAN ANDRES
- SANTA LUCIA UTATLAN
- IXCHIGUAN
- SAN GASPAR IXCHIL
- SAN JOSE ACATEMPA
- OLOPA
- SANTA APOLONIA
- JOYABAJ
- SANTA CRUZ BALANYA
- SANTA MARIA VISITACION
- SAN SEBASTIAN COATAN
- SAN ANTONIO HUISTA
- MALACATANCITO
- SAN BARTOLO
- SAN JACINTO
- CASILLAS
- SAN MARCOS LA LAGUNA
- MONJAS
- CONCEPCION LAS MINAS
- SAN JOSE LA ARADA
- CONGUACO
- SAN JUAN LA ERMITA
- SAN CRISTOBAL ACASAGUASTLAN
- SAN JOSE DEL GOLFO
- SAN FRANCISCO
- SAN MIGUEL PANAN
- SAN MANUEL CHAPARRON
- SAN JOSE LA MAQUINA
- ZACUALPA
- SAN JOSE POAQUIL
- SAN RAFAEL LAS FLORES
- CAMOTAN

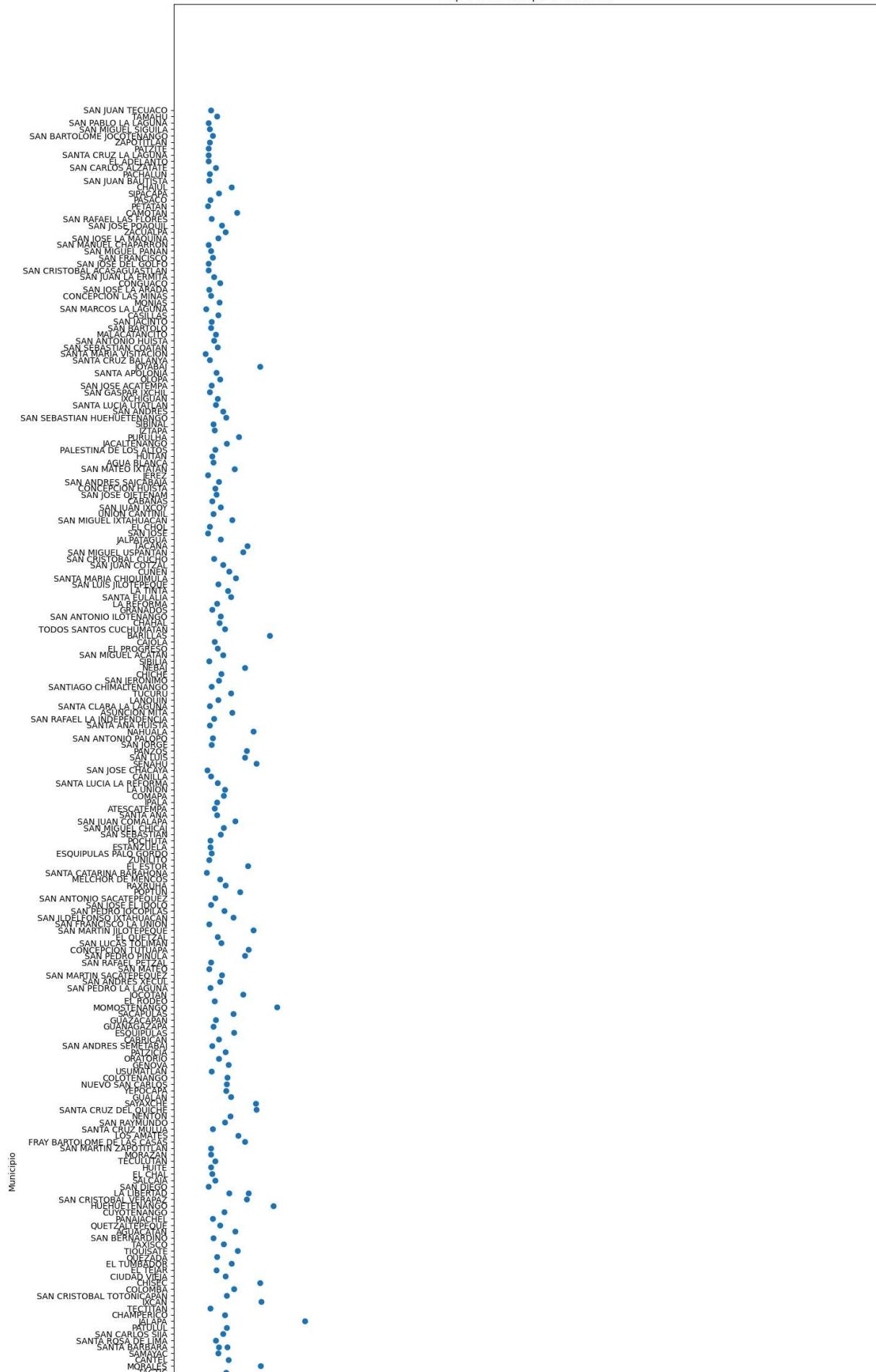


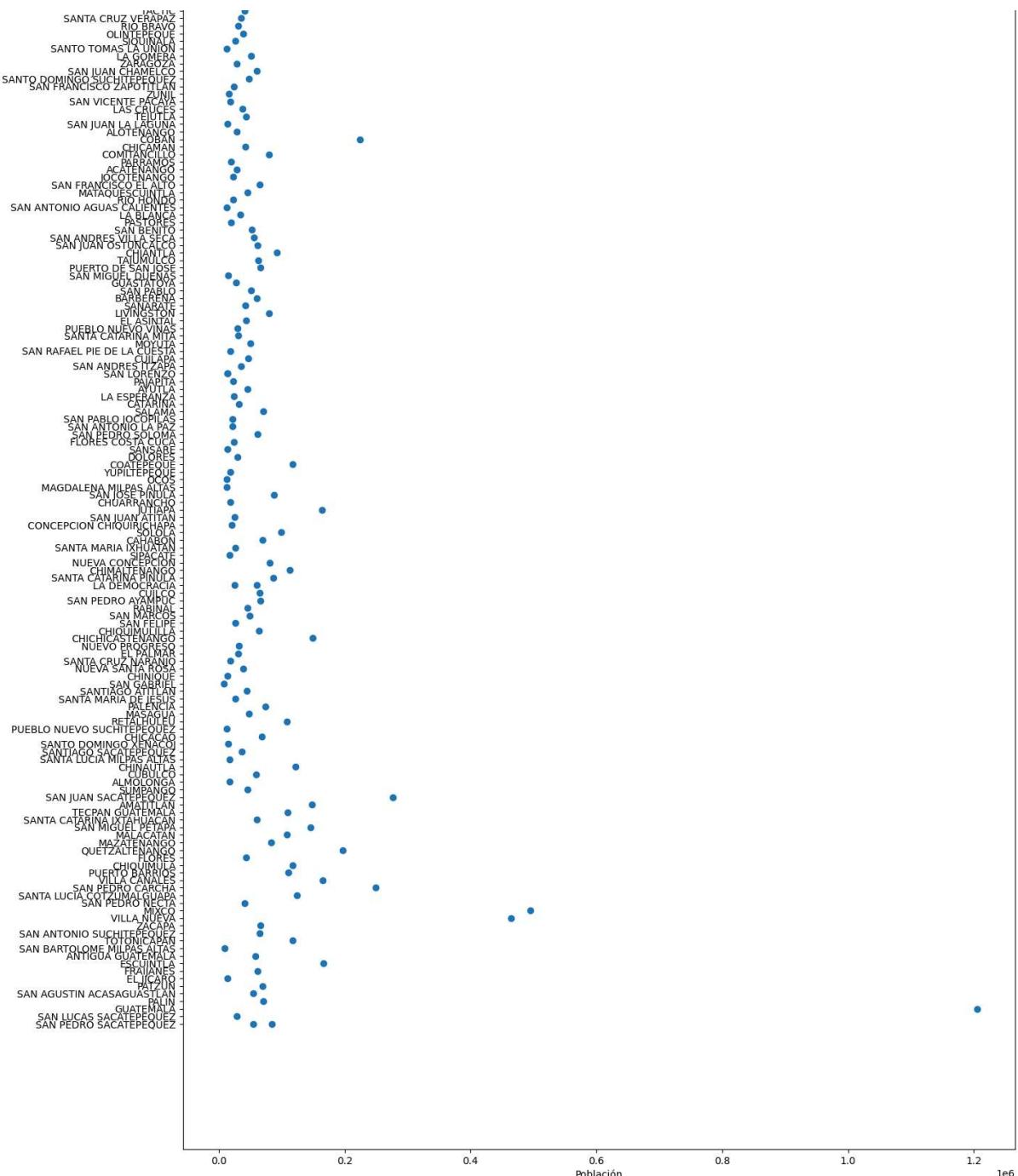
Observaciones

- Esta gráfica de dispersión nos confirma precisamente lo que se mencionaba en puntos anteriores, a pesar de que la ciudad de Guatemala es la ciudad con más habitantes en el país (con una diferencia abismal), el territorio Guatemalteco se compone en su mayoría por municipios que no poseen una población tan grande como la ciudad de Guatemala, lo cual provoca que los datos se encuentren más agrupados en la mayoría del país, mientras que la ciudad de Guatemala se encuentra demasiado dispersa.

```
In [72]: plt.figure(figsize=(15, 50))
plt.scatter(df_municipality['poblacion'], df_municipality['nombre_municipio'])
plt.xlabel('Población')
plt.ylabel('Municipio')
plt.title('Dispersión: Municipio vs Población')
plt.show()
```

Dispersión: Municipio vs Población

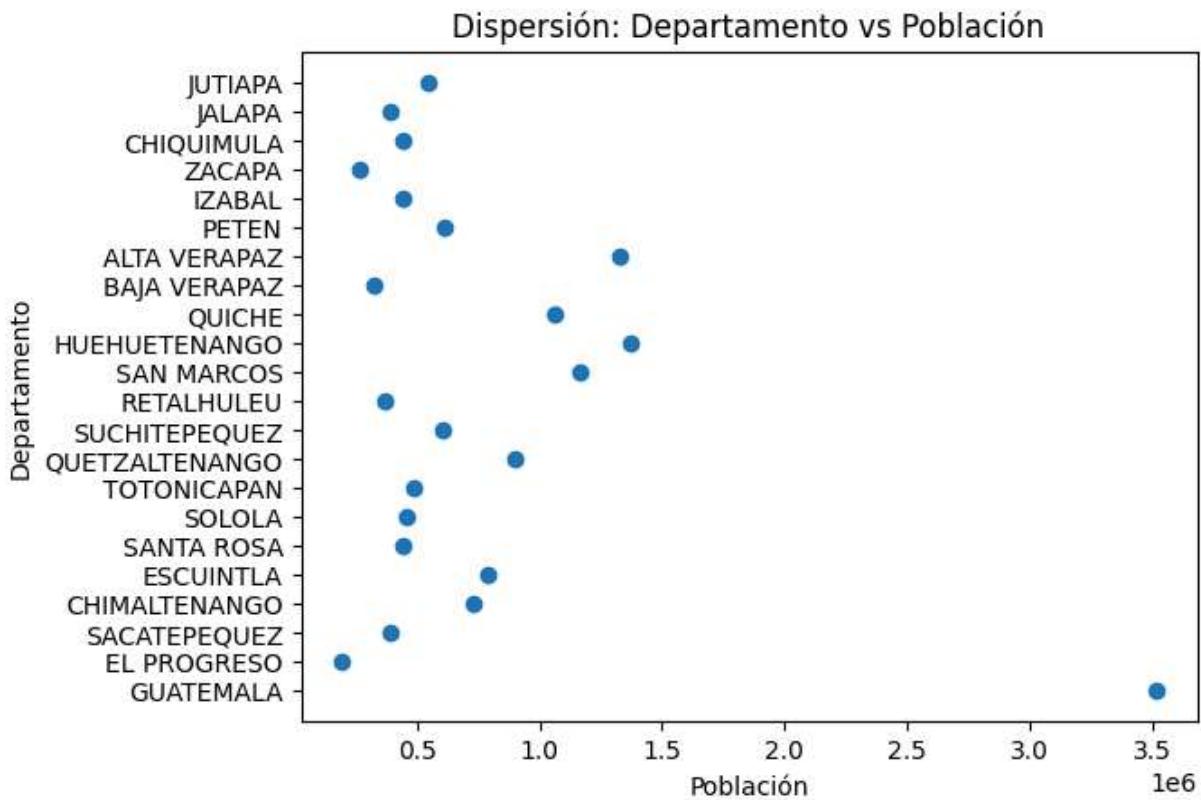




Observaciones

- Una gráfica muy similar a la anterior, sin embargo, ahora que se analiza la población por departamento, podemos observar que los datos se encuentran más dispersos, esto se debe a que al ser porciones de territorio más grandes la diferencia entre poblaciones tiende a ser mayor.

```
In [27]: plt.scatter(df_department_population['poblaciontotal'], df_department_population['n'])
plt.xlabel('Población')
plt.ylabel('Departamento')
plt.title('Dispersión: Departamento vs Población')
plt.show()
```



Graficas

Observacion

- Estas gráficas muestran la cantidad de días en los que cada municipio ha registrado al menos una muerte. En ese caso ya podemos observar algunas variaciones en comparación a las estadísticas de población, ya que, a pesar de que la ciudad de Guatemala sigue siendo el número 1, ya la diferencia se reduce bastante (principalmente porque el contexto de la gráfica es otro).
- La gráfica también nos muestra otras variaciones, como por ejemplo la ciudad de Quetzaltenango, ya que, a pesar de que es el número 7 en la gráfica de población, en esta gráfica se encuentra en el puesto 4. Esto se puede deber al comercio que se da en esta ciudad que a pesar de no tener tantos habitantes como otras ciudades, el comercio si es muy relevante.

```
In [28]: # Filtrar las filas donde total_deaths no es igual a cero
df_filtered = df_municipality_deaths[df_municipality_deaths['fallecidos'] != 0]

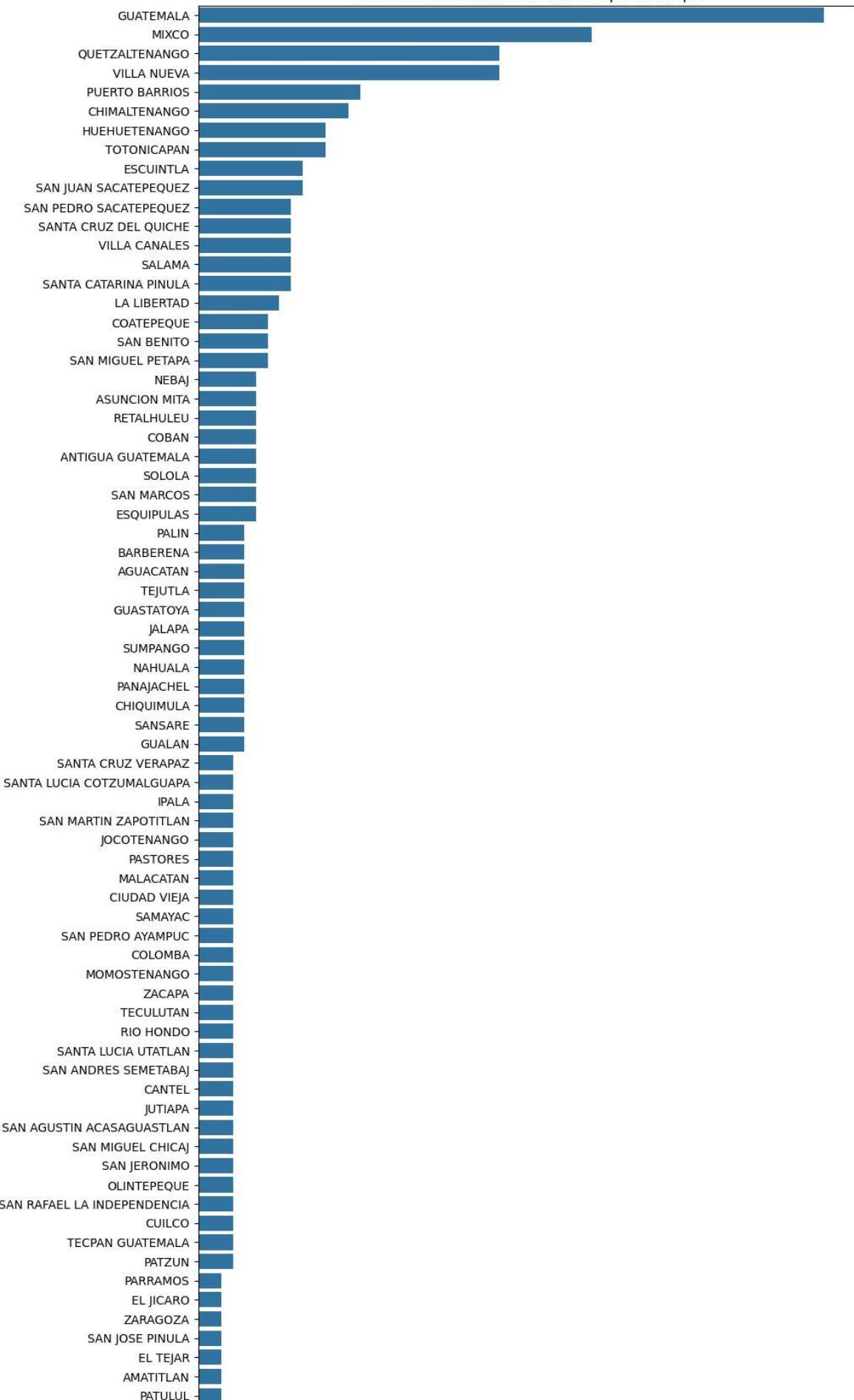
# Contar las ocurrencias de cada "name" en el DataFrame filtrado
df_municipality_count = df_filtered['nombre_municipio'].value_counts().reset_index()
df_municipality_count.columns = ['nombre_municipio', 'count']

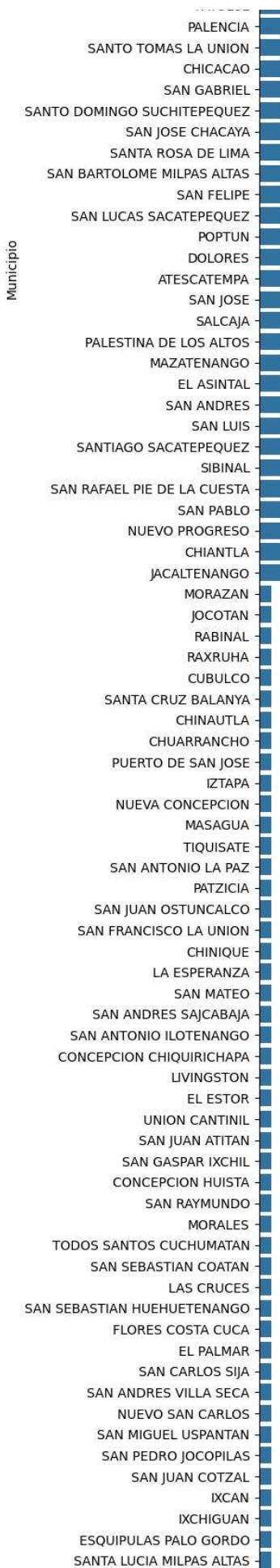
# Crear una gráfica de barras con seaborn
plt.figure(figsize=(10, 50))
sns.barplot(x='count', y='nombre_municipio', data=df_municipality_count)
```

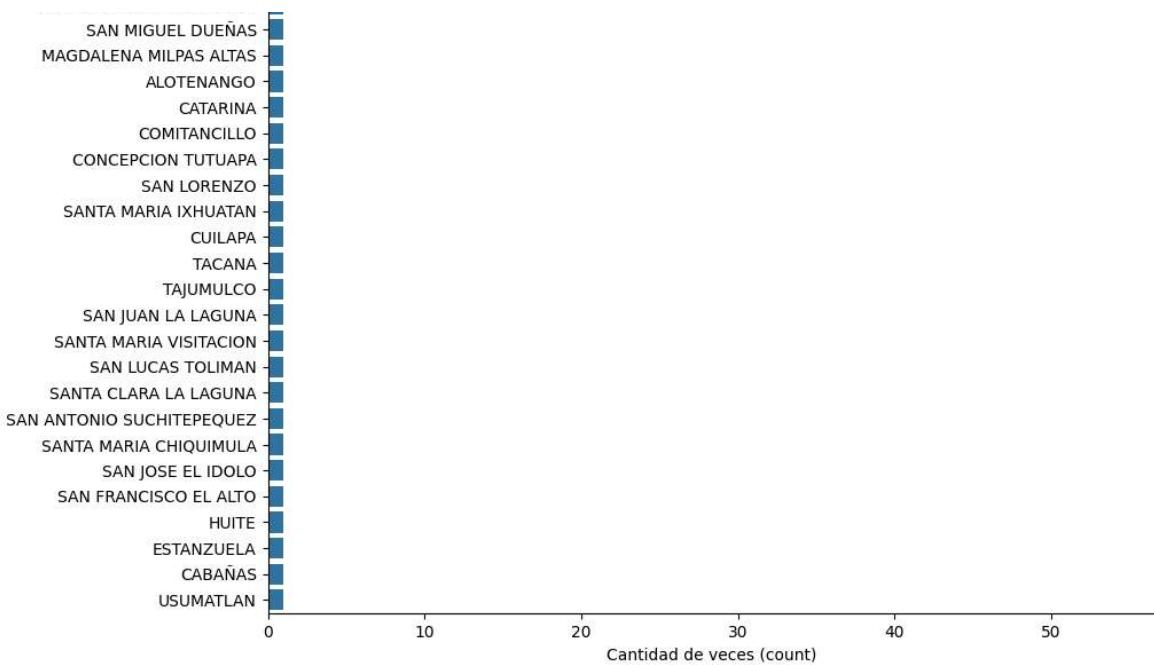
```
# Añadir etiquetas y título
plt.xlabel('Cantidad de veces (count)')
plt.ylabel('Municipio')
plt.title('Conteo de Días con Nuevas Muertes por Municipio')

# Mostrar el gráfico
plt.show()
```

Conteo de Días con Nuevas Muertes por Municipio







```
In [29]: df_filtered = df_municipality_deaths[df_municipality_deaths['fallecidos'] != 0]

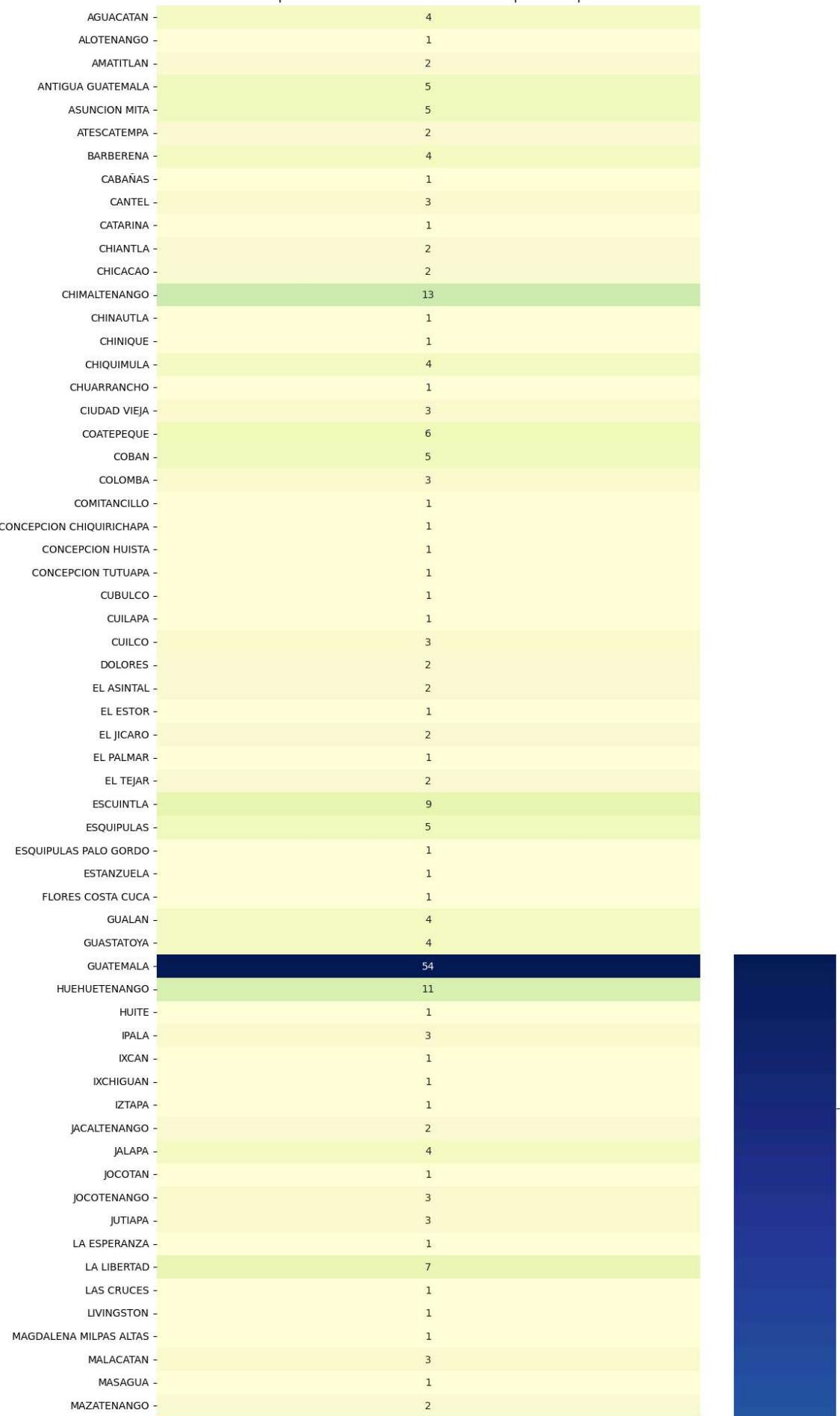
# Contar las ocurrencias de cada "name" en el DataFrame filtrado
df_municipality_count = df_filtered['nombre_municipio'].value_counts().reset_index()
df_municipality_count.columns = ['nombre_municipio', 'count']

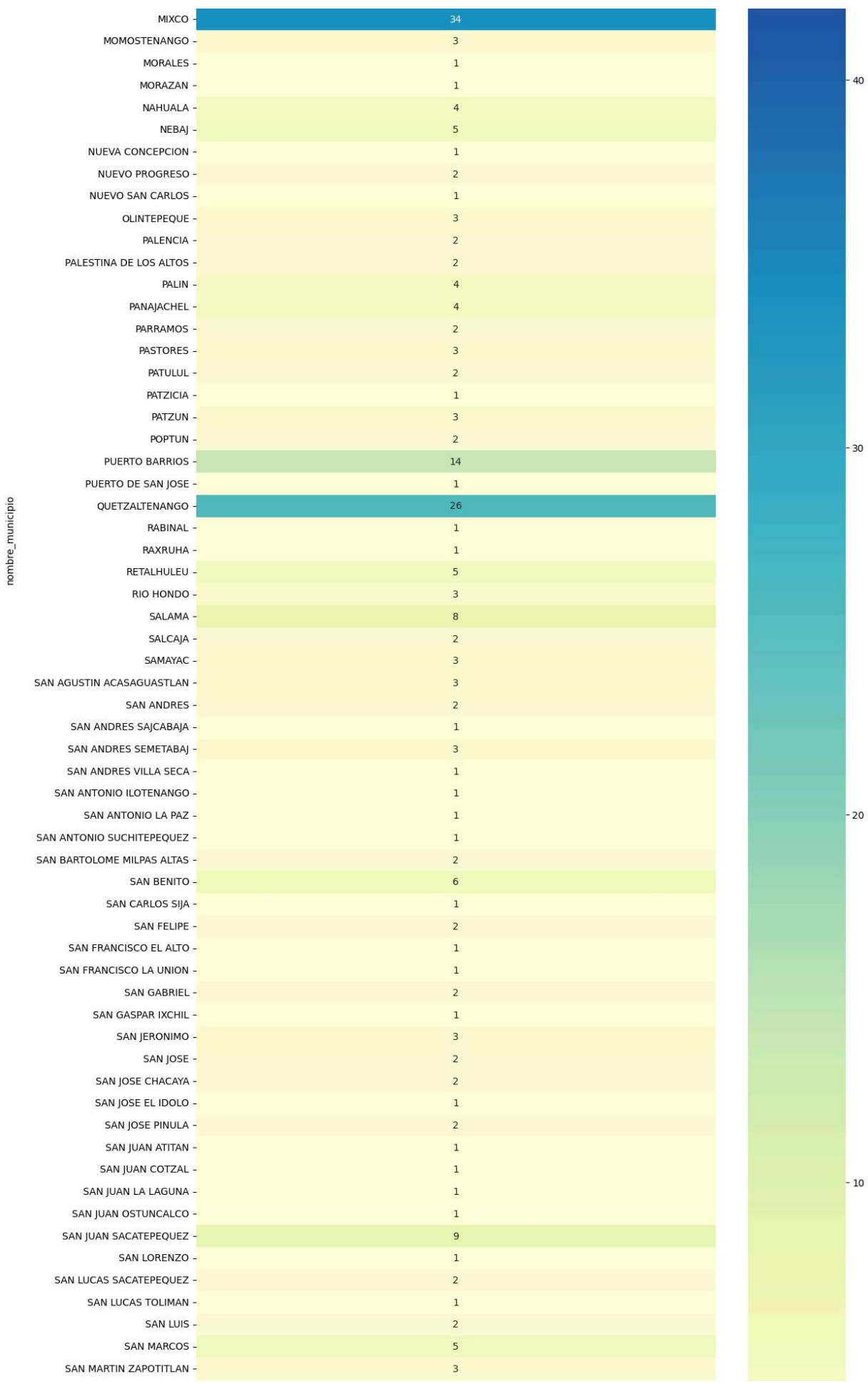
# Crear un mapa de calor con seaborn
plt.figure(figsize=(12, 70))
sns.heatmap(df_municipality_count.pivot_table(index='nombre_municipio', values='cou

# Añadir título
plt.title('Mapa de Calor: Días con Nuevas Muertes por Municipio')

# Mostrar el gráfico
plt.show()
```

Mapa de Calor: Días con Nuevas Muertes por Municipio







Departamento vs Nuevas Muertes

Observaciones

- En esta grafica la tendencia es nuevamente al departamento de Guatemala, ya que este posee una gran cantidad de municipios que poseían un alto indice de fallecimientos por covid.

- Podemos observar a simple vista que los departamentos con mayor presencia de municipios urbanos son aquellos que han sido más afectados. Sin embargo hay casos como el de San Marcos, donde se nota a simple vista que ha sido muy afectado, pero éste es de los pocos casos donde hay gran presencia de área rural y el covid ha afectado fuertemente.
- El caso de San Marcos puede darse debido a ser aledaño al departamento de Quetzaltenango

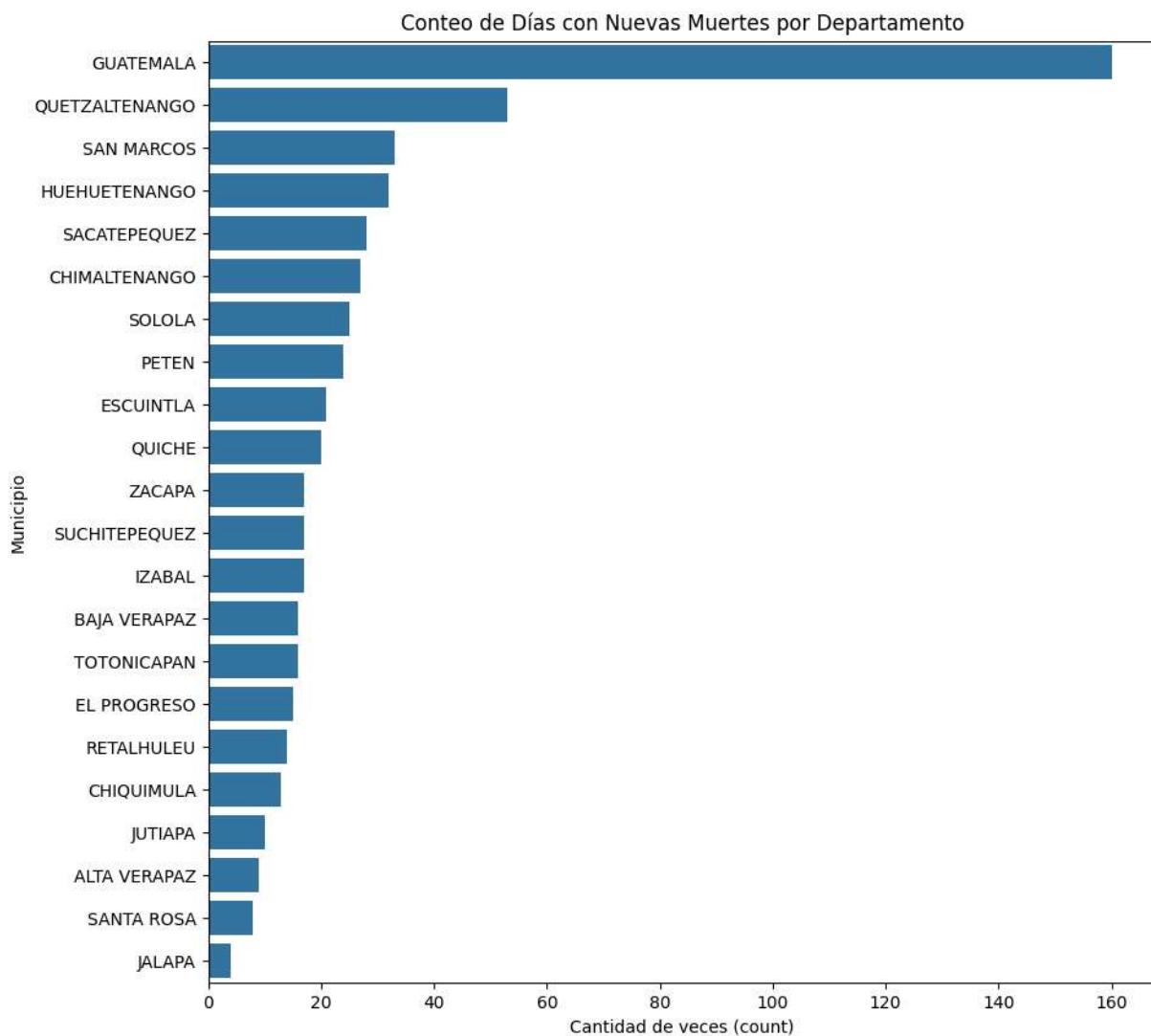
```
In [30]: df_filtered = df_municipality_deaths[df_municipality_deaths['fallecidos'] != 0]

# Contar las ocurrencias de cada "name" en el DataFrame filtrado
df_municipality_count = df_filtered['nombre_departamento'].value_counts().reset_index()
df_municipality_count.columns = ['nombre_departamento', 'count']

# Crear una gráfica de barras con seaborn
plt.figure(figsize=(10, 10))
sns.barplot(x='count', y='nombre_departamento', data=df_municipality_count)

# Añadir etiquetas y título
plt.xlabel('Cantidad de veces (count)')
plt.ylabel('Municipio')
plt.title('Conteo de Días con Nuevas Muertes por Departamento')

# Mostrar el gráfico
plt.show()
```

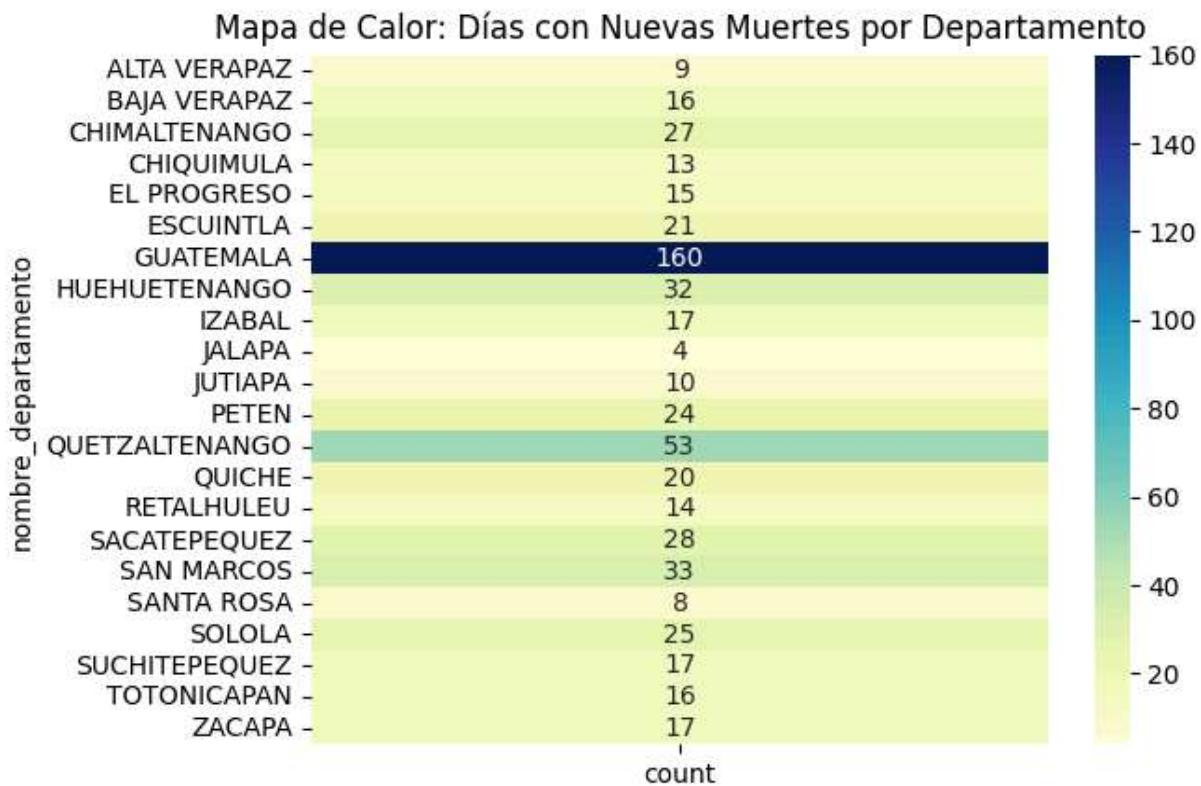


```
In [31]: df_municipality_count = df_filtered['nombre_departamento'].value_counts().reset_index()
df_municipality_count.columns = ['nombre_departamento', 'count']

sns.heatmap(df_municipality_count.pivot_table(index='nombre_departamento', values='count', aggfunc='sum'))

# Añadir título
plt.title('Mapa de Calor: Días con Nuevas Muertes por Departamento')

# Mostrar el gráfico
plt.show()
```



Municipios vs Poblacion

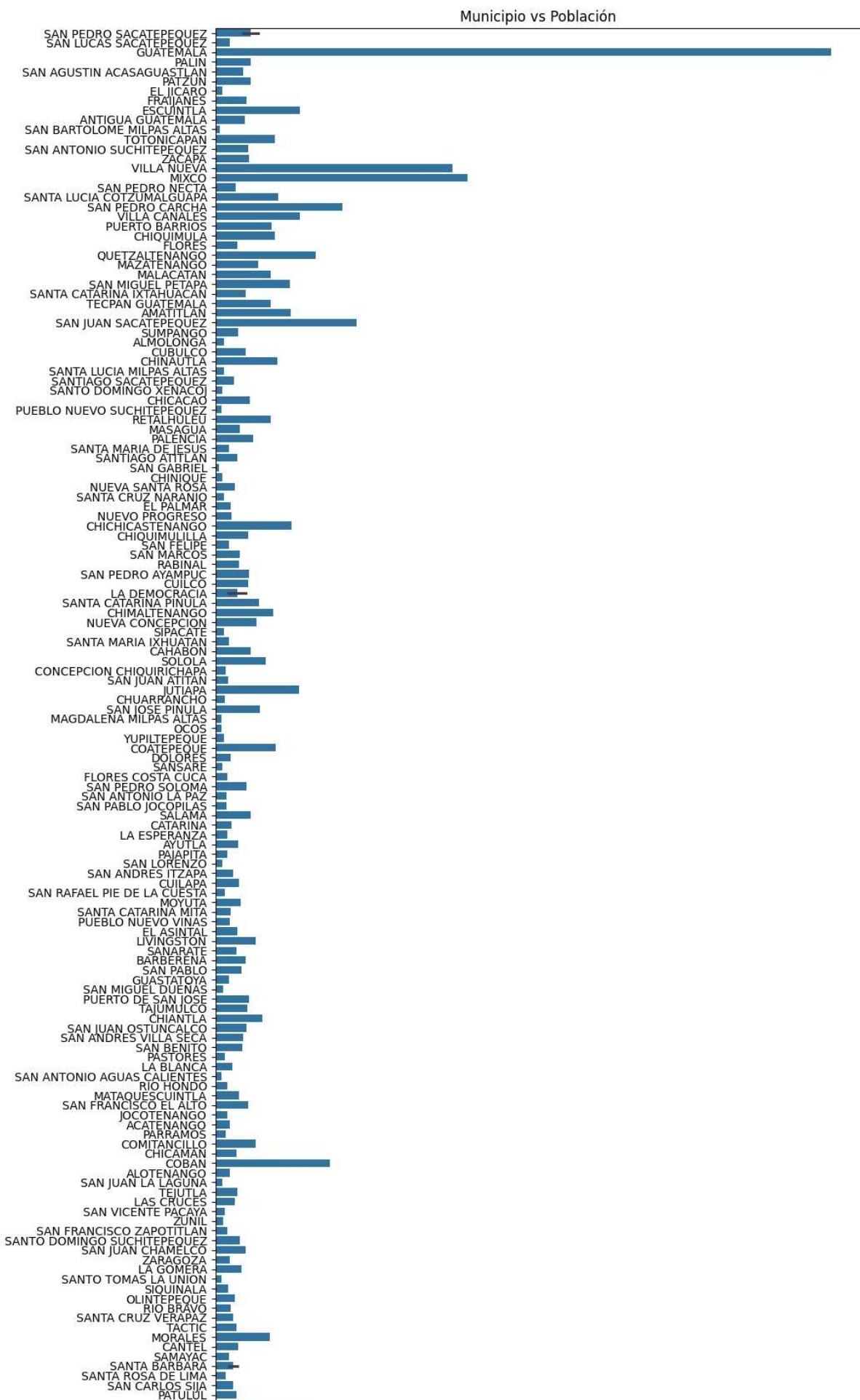
Observaciones

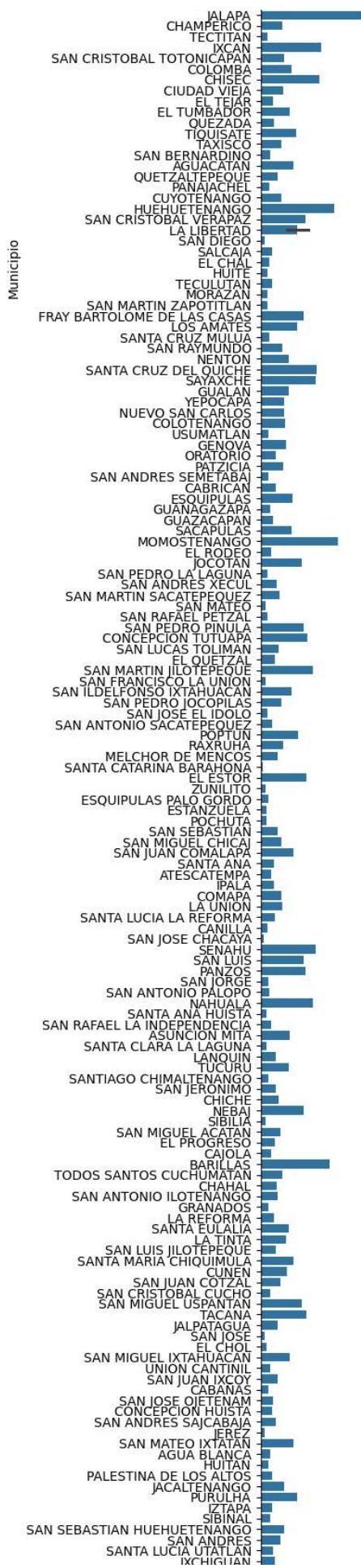
- La Grafica que denota la poblacion de cada municipio del pais, posee un comportamiento similar al de la gráfica de dispersión anteriormente mostrada.
- Podemos observar y notar a simple vista que los municipios pertenecientes al área urbana poseen la mayor concentración de pobladores, esto se da porque el comercio se da de una forma más abundante en estas áreas.

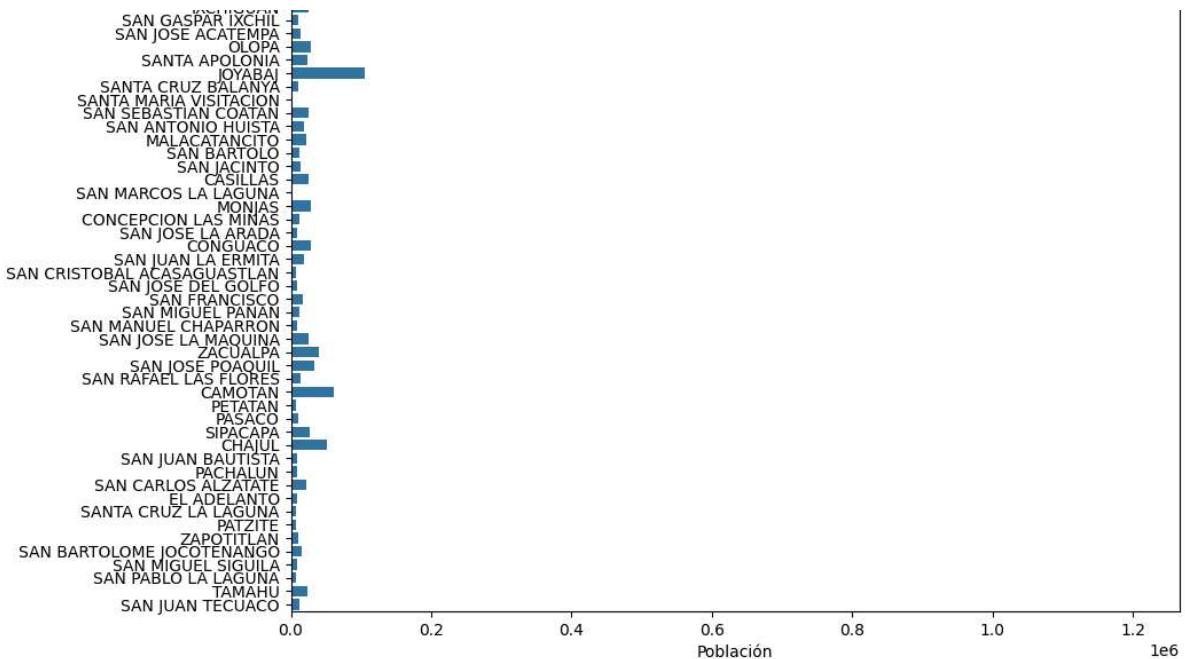
```
In [32]: # Crear una gráfica de barras con seaborn
plt.figure(figsize=(10, 50))
sns.barplot(x='poblacion', y='nombre_municipio', data=df_municipality)

# Añadir etiquetas y título
plt.xlabel('Población')
plt.ylabel('Municipio')
plt.title('Municipio vs Población')

# Mostrar el gráfico
plt.show()
```





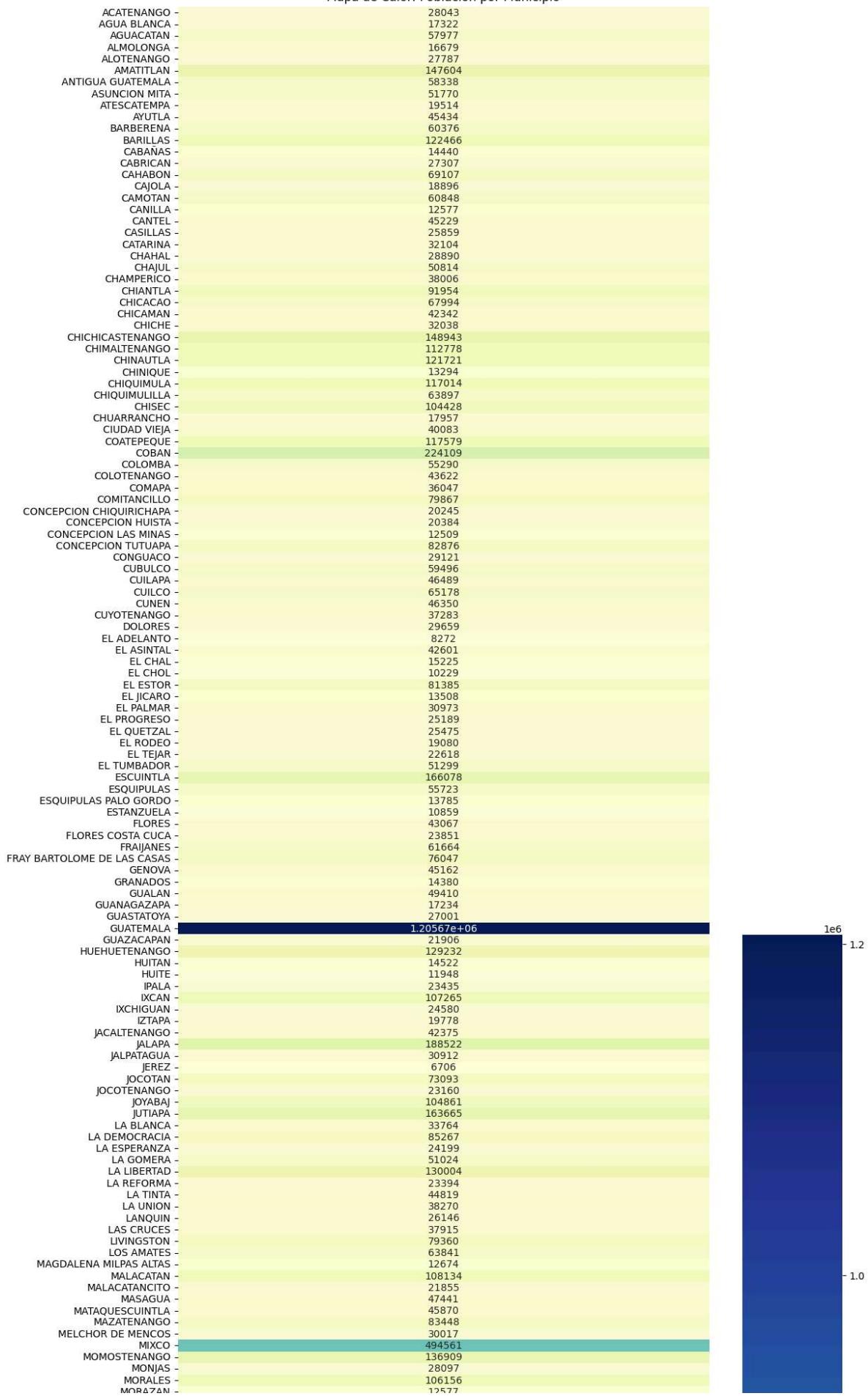


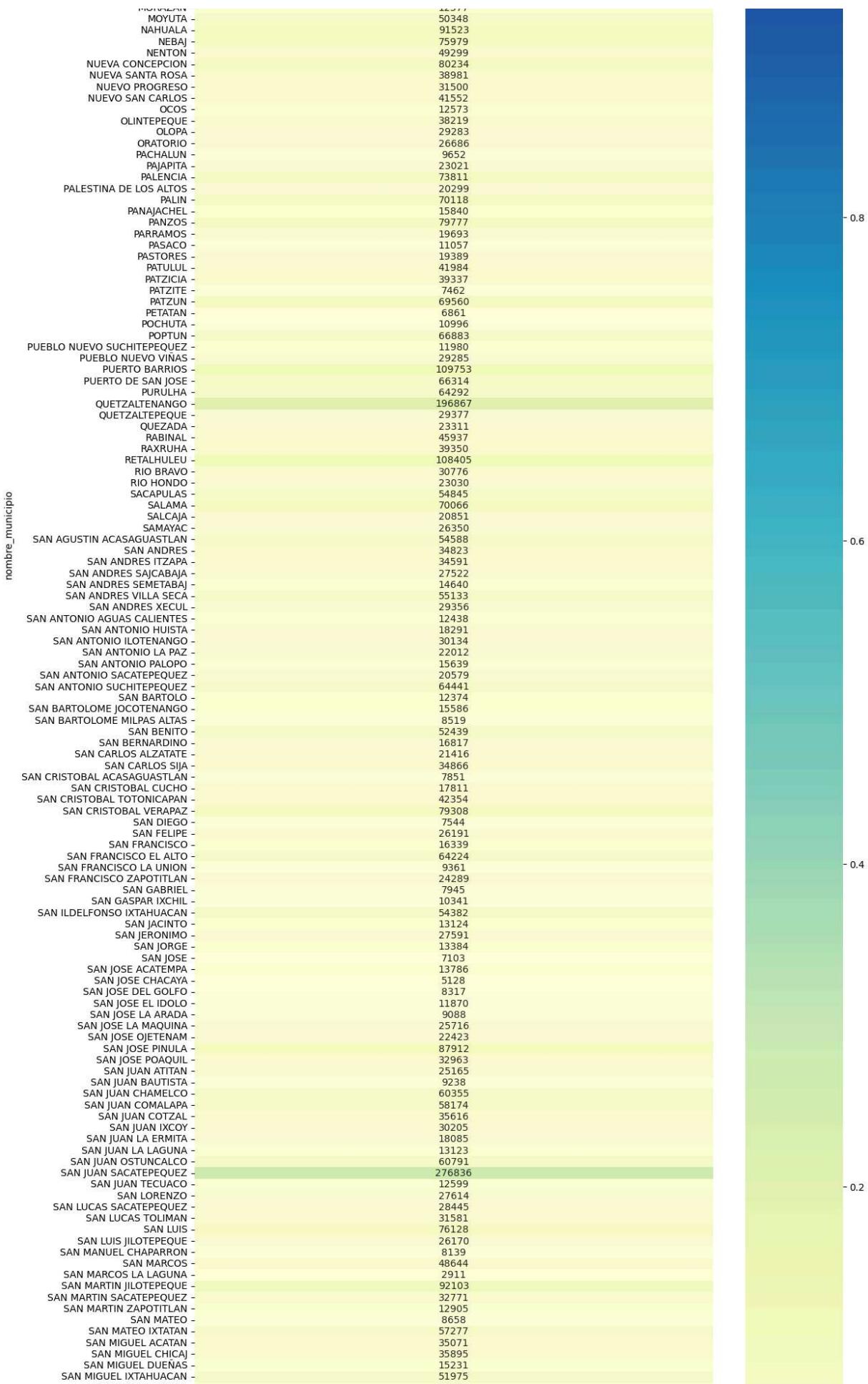
```
In [33]: # Crear un mapa de calor con seaborn
plt.figure(figsize=(12, 70))
sns.heatmap(df_municipality.pivot_table(index='nombre_municipio', values='poblacion'

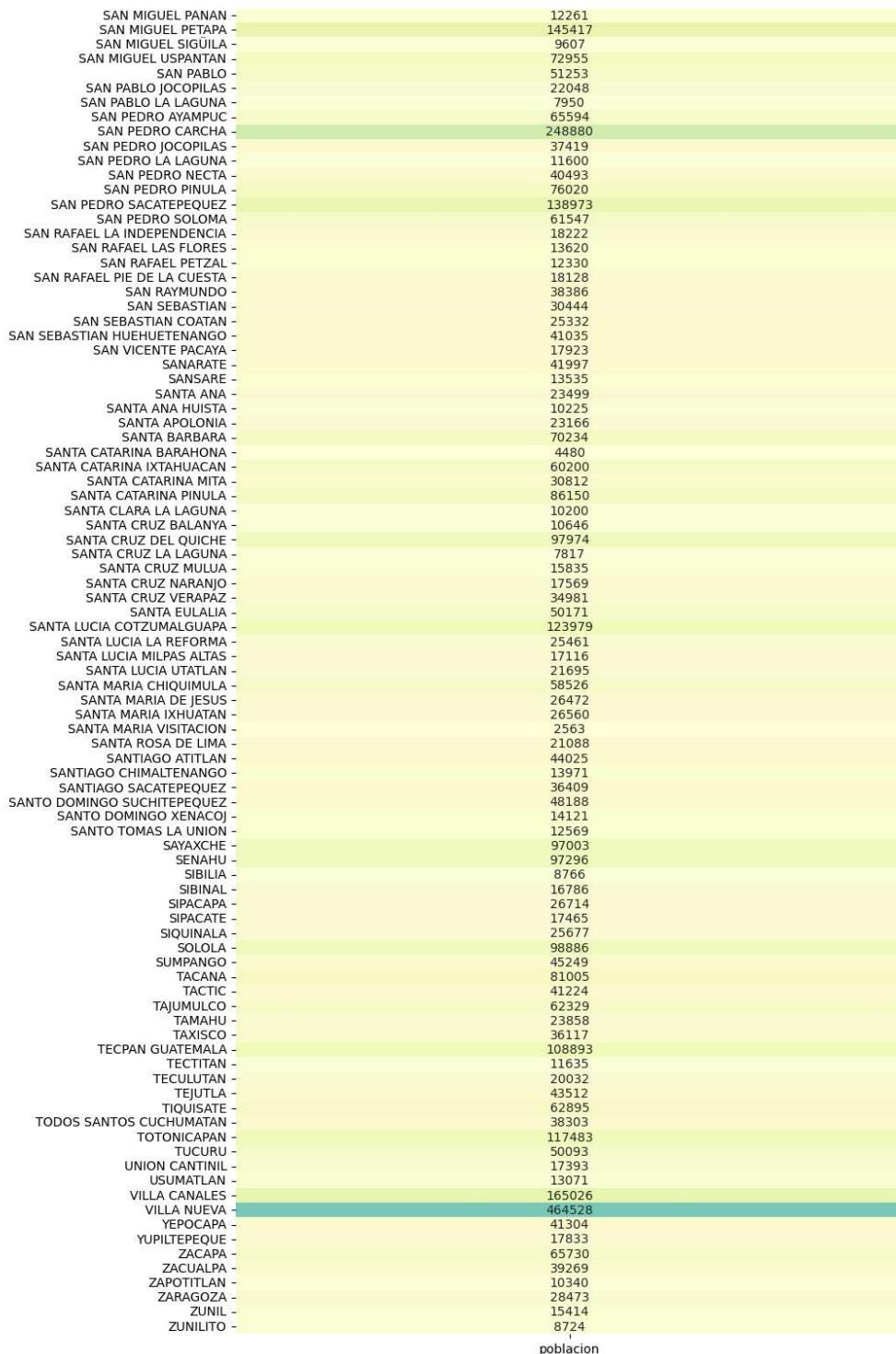
# Añadir título
plt.title('Mapa de Calor: Población por Municipio')

# Mostrar el gráfico
plt.show()
```

Mapa de Calor: Población por Municipio







Departamento vs Poblacion

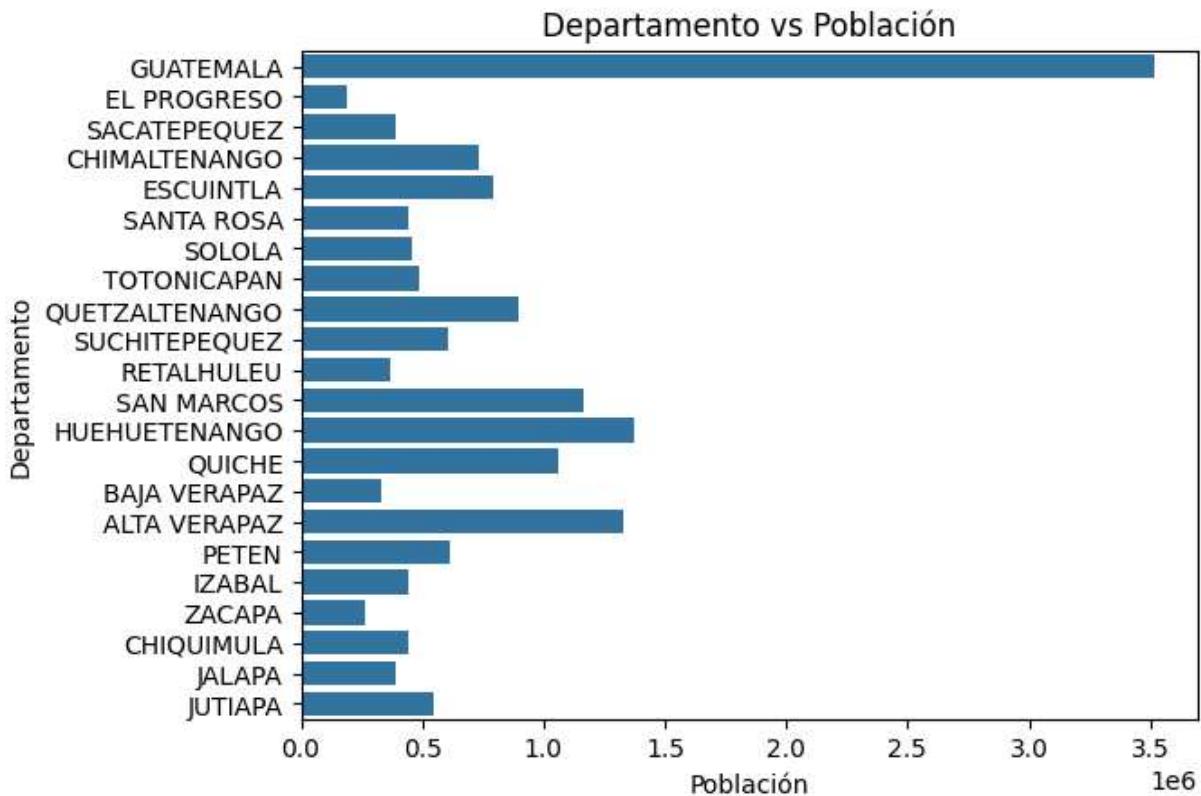
Observaciones

- Se puede observar a simple vista que la diferencia entre poblaciones es más grande cuando englobamos los departamentos, esto debido a que hay departamentos que contienen varios municipios del área urbana.
- Se destaca el municipio de Alta Verapaz, en el cual la influencia del municipio de Cobán juega un rol muy importante para obtener una población aceptable.

```
In [34]: sns.barplot(x='poblaciontotal', y='nombredepartamento', data=df_department_population)

# Añadir etiquetas y título
plt.xlabel('Población')
plt.ylabel('Departamento')
plt.title('Departamento vs Población')

# Mostrar el gráfico
plt.show()
```



```
In [35]: # Crear un mapa de calor con seaborn
sns.heatmap(df_department_population.pivot_table(index='nombredepartamento', values='poblaciontotal'))

# Añadir título
plt.title('Mapa de Calor: Población por Departamento')

# Mostrar el gráfico
plt.show()
```



Municipio vs Muertes Acumuladas

Observaciones

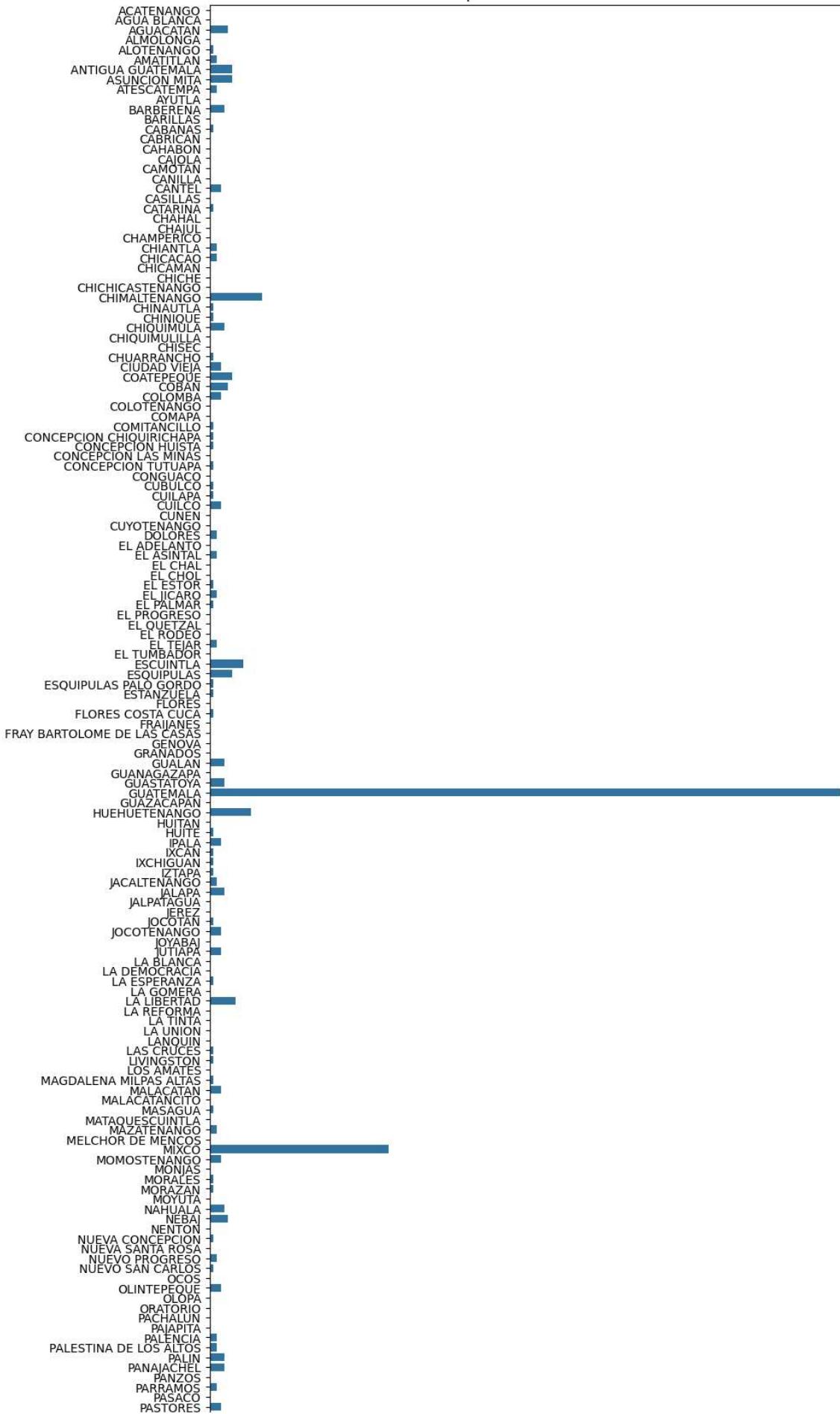
- La situación de los municipios de Guatemala, Mixco y Villa Nueva es sumamente preocupante, ya que la cantidad de personas que han fallecido es muy grande, si bien, es cierto que esto se debe a su población es considerablemente mayor, estamos hablando de vidas humanas, por lo cual la recomendación primordial sería implementar medidas para evitar que siguieran multiplicándose.
- Si obviamos estos 3 municipios, también debemos centrarnos en el municipio de Quetzaltenango, ya que a pesar de que la cantidad de fallecidos es por mucho, menor a Guatemala, Mixco y Villa Nueva, Quetzaltenango

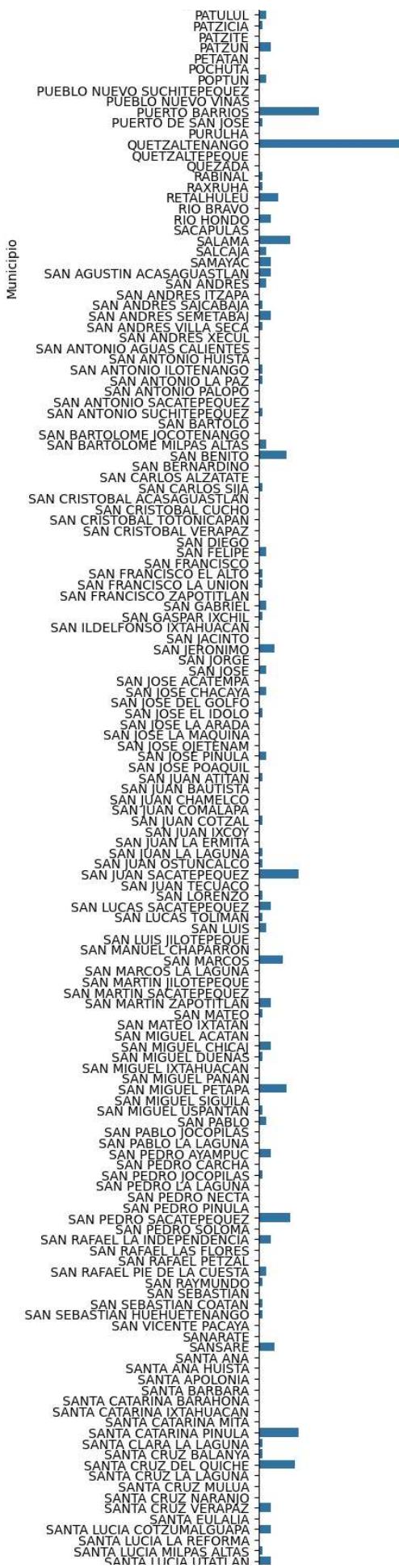
```
In [36]: sum_deaths_by_municipality = df_municipality_deaths.groupby('nombre_municipio')['fa
plt.figure(figsize=(10, 50))
sns.barplot(x='fallecidos', y='nombre_municipio', data=sum_deaths_by_municipality)

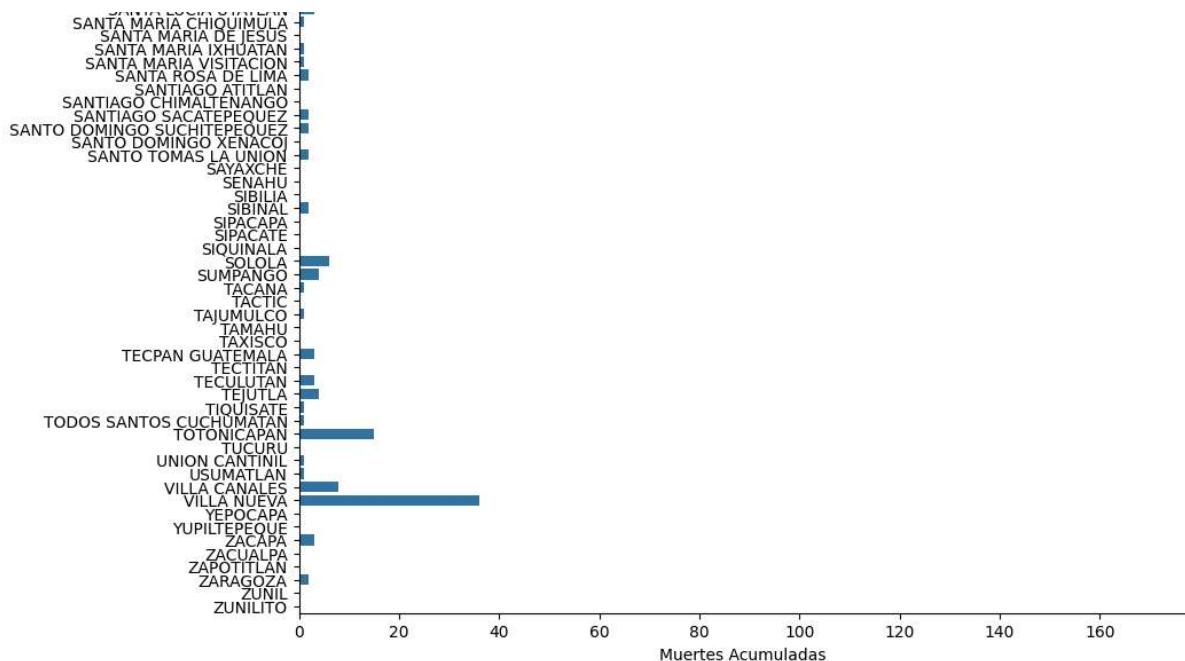
# Añadir etiquetas y título
plt.xlabel('Muertes Acumuladas')
plt.ylabel('Municipio')
plt.title('Municipio vs Muertes Acumuladas')

# Mostrar el gráfico
plt.show()
```

Municipio vs Muertes Acumuladas







```
In [37]: # Filtrar las filas donde "municipality_name" no sea igual a "GUATEMALA" "MIXCO" o
df_filtered = df_municipality_deaths.query('nombre_municipio not in ["GUATEMALA", "']

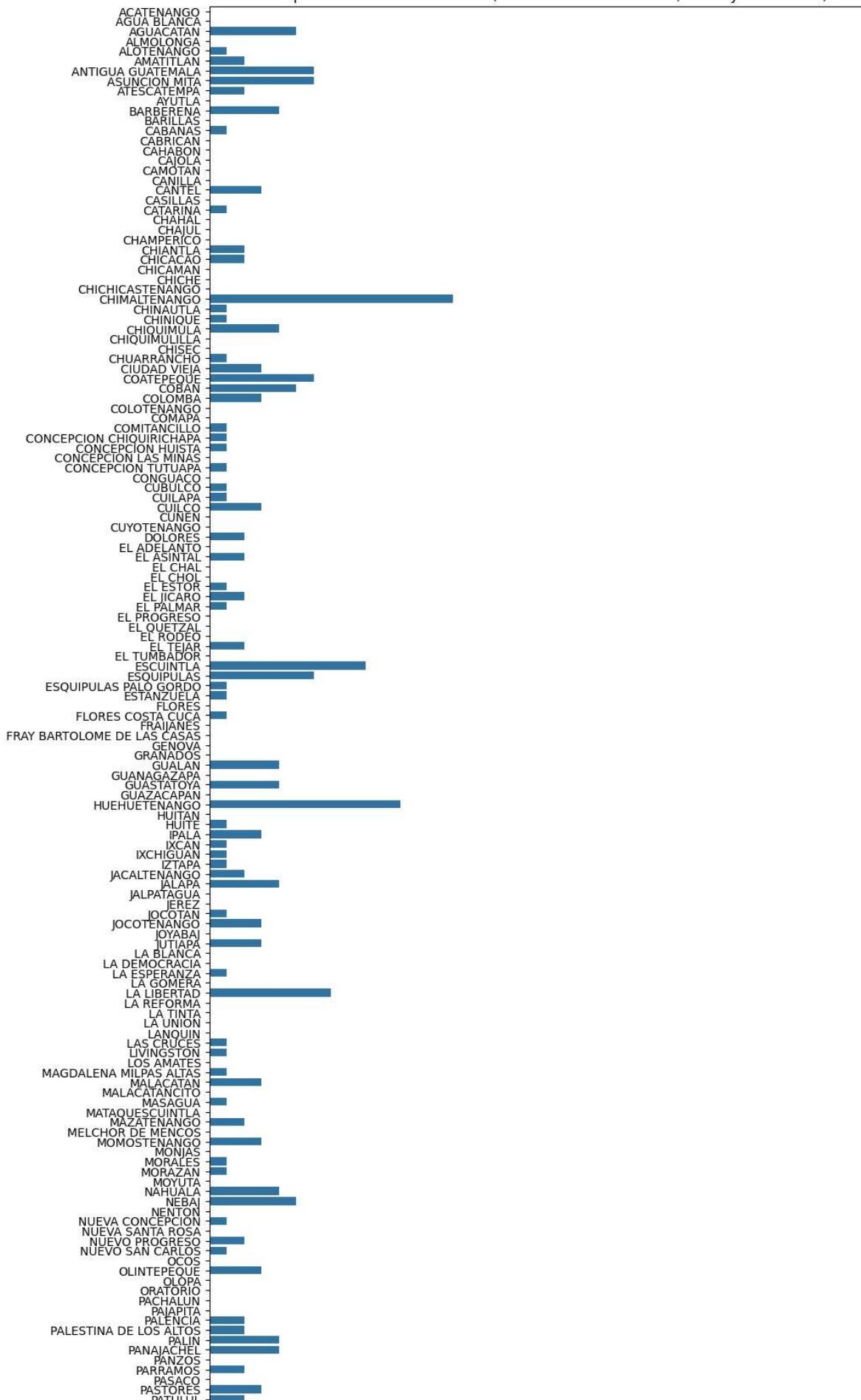
# Agrupar por "municipality_name" y sumar la cantidad total de muertes
sum_deaths_by_municipality = df_filtered.groupby('nombre_municipio')['fallecidos'].

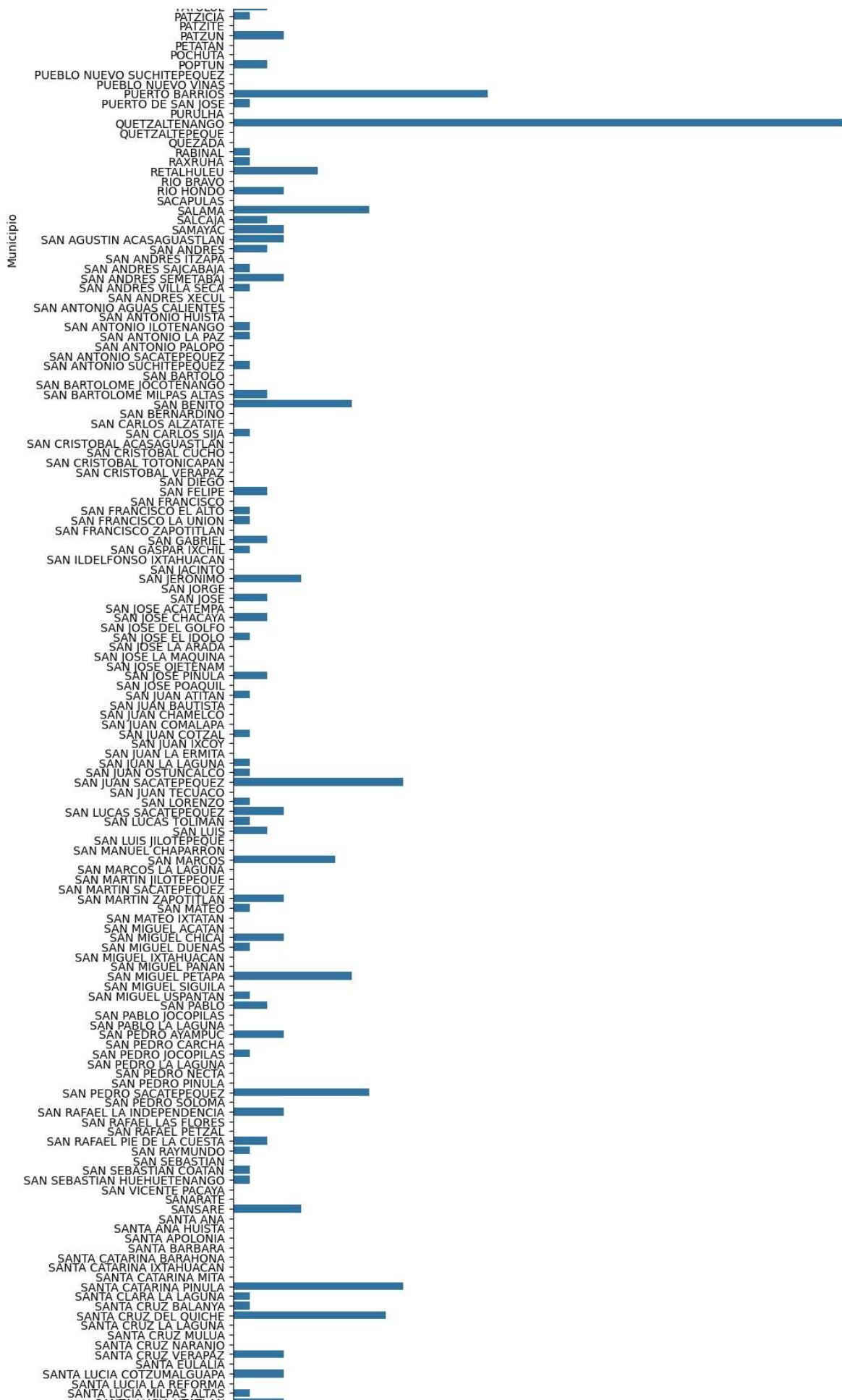
plt.figure(figsize=(10, 50))
sns.barplot(x='fallecidos', y='nombre_municipio', data=sum_deaths_by_municipality)

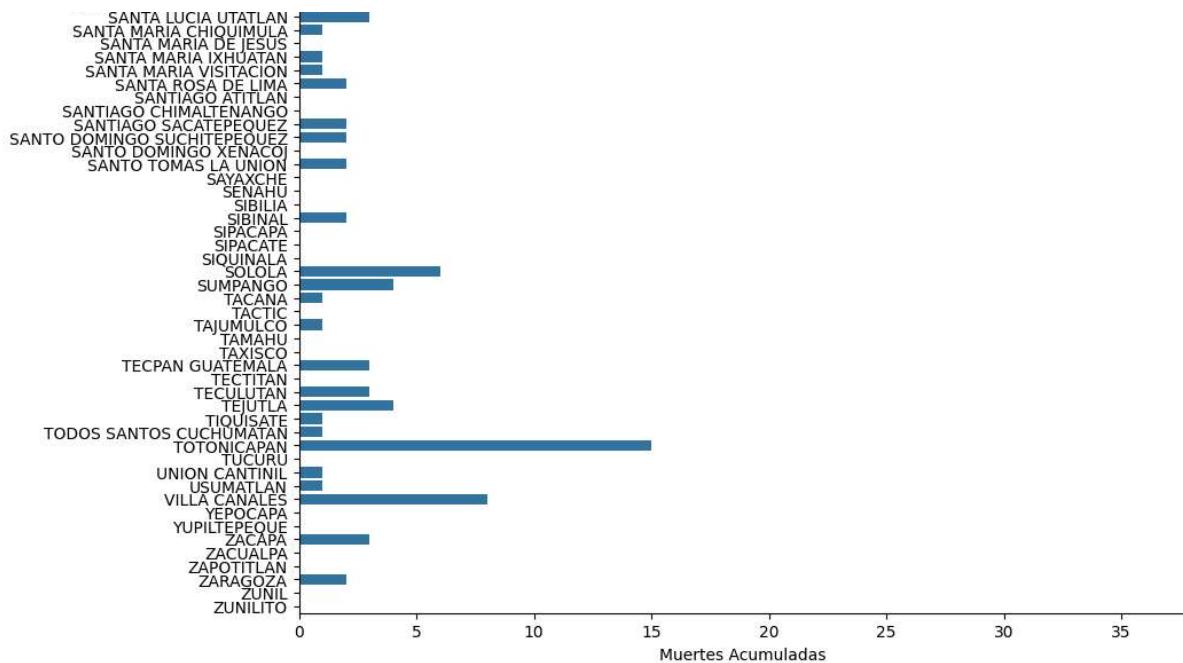
# Añadir etiquetas y título
plt.xlabel('Muertes Acumuladas')
plt.ylabel('Municipio')
plt.title('Municipio vs Muertes Acumuladas (Sin Ciudad de Guatemala, Mixco y Villa

# Mostrar el gráfico
plt.show()
```

Municipio vs Muertes Acumuladas (Sin Ciudad de Guatemala, Mixco y Villa Nueva)

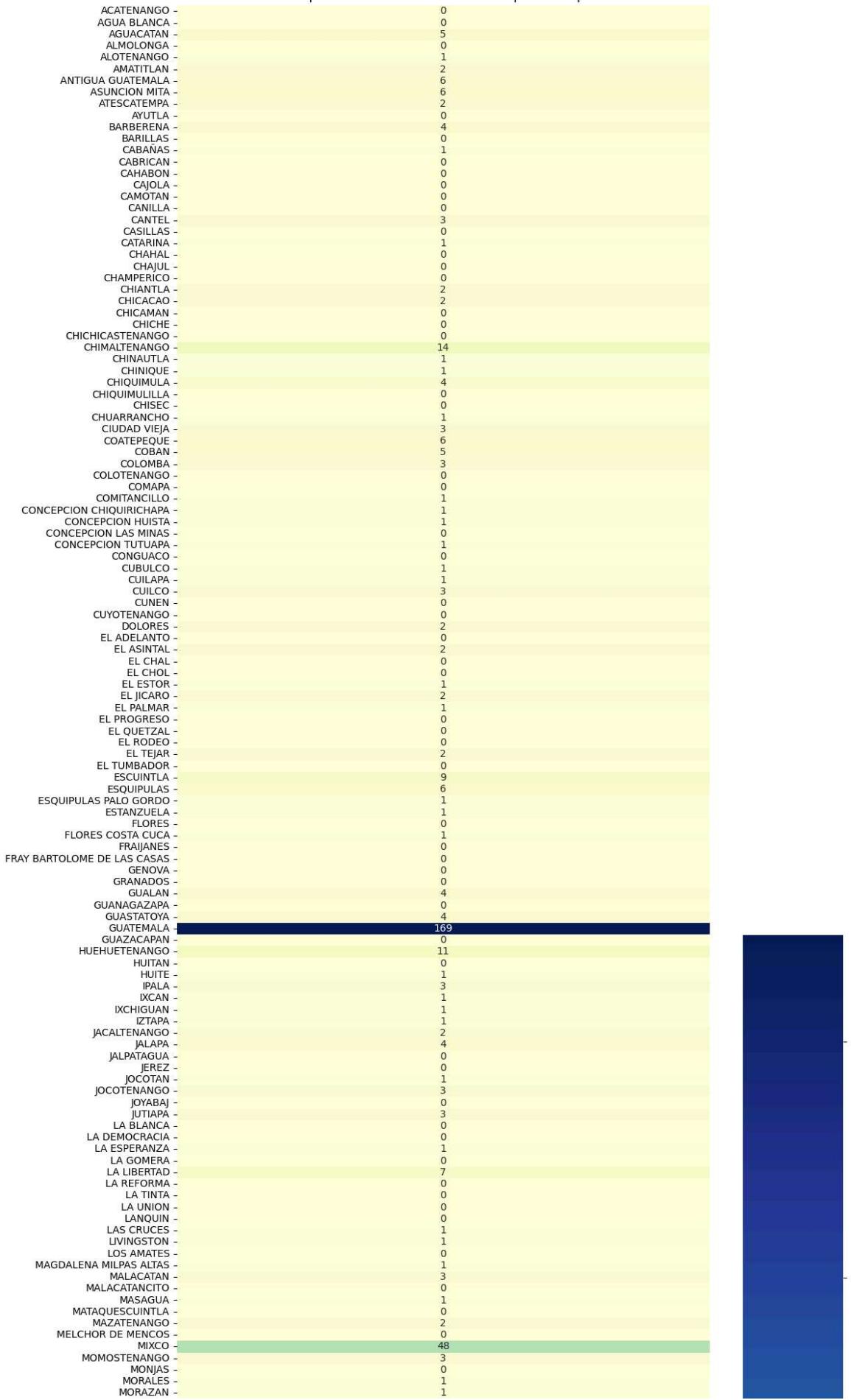


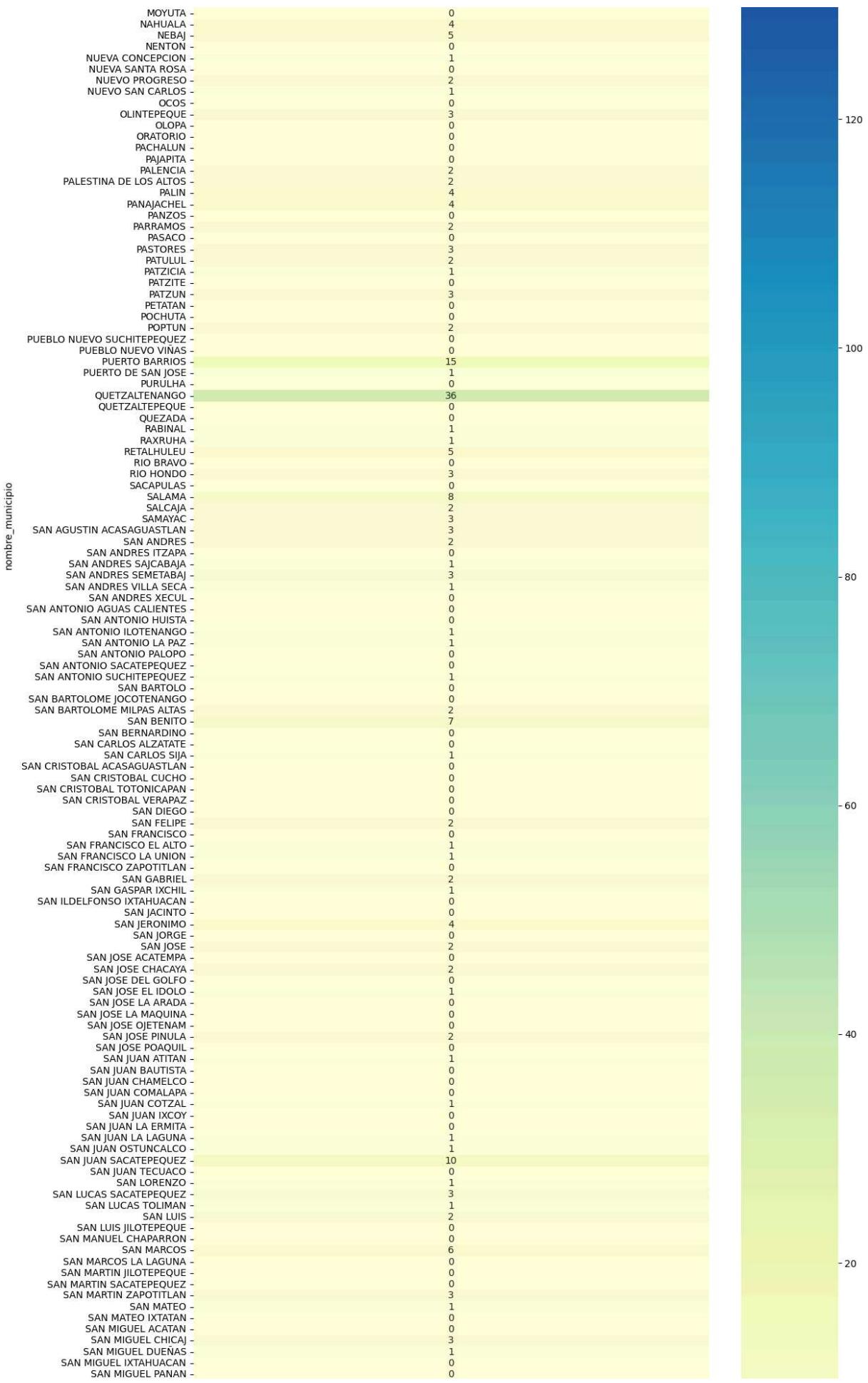




```
In [38]: sum_deaths_by_municipality = df_municipality_deaths.groupby('nombre_municipio')['fa  
  
plt.figure(figsize=(12, 70))  
# Crear un mapa de calor con seaborn  
sns.heatmap(sum_deaths_by_municipality.pivot_table(index='nombre_municipio', values=  
  
# Añadir título  
plt.title('Mapa de Calor: Muertes Acumuladas por municipio')  
  
# Mostrar el gráfico  
plt.show()
```

Mapa de Calor: Muertes Acumuladas por municipio







SAN MIGUEL PETAPA -	7
SAN MIGUEL SIGÜLA -	0
SAN MIGUEL USPANTAN -	1
SAN PABLO -	2
SAN PABLO JOCOPILAS -	0
SAN PABLO LA LAGUNA -	0
SAN PEDRO AYAMPUC -	3
SAN PEDRO CARCHA -	0
SAN PEDRO JOCOPILAS -	1
SAN PEDRO LA LAGUNA -	0
SAN PEDRO NECTA -	0
SAN PEDRO PINULA -	0
SAN PEDRO SACATEPEQUEZ -	8
SAN PEDRO SOLOMA -	0
SAN RAFAEL LA INDEPENDENCIA -	3
SAN RAFAEL LAS FLORES -	0
SAN RAFAEL PETZAL -	0
SAN RAFAEL PIE DE LA CUESTA -	2
SAN RAYMUNDO -	1
SAN SEBASTIAN -	0
SAN SEBASTIAN COATAN -	1
SAN SEBASTIAN HUEHUETENANGO -	1
SAN VICENTE PACAYA -	0
SANARATE -	0
SANSARE -	4
SANTA ANA -	0
SANTA ANA HUISTA -	0
SANTA APOLONIA -	0
SANTA BARBARA -	0
SANTA CATARINA BARAHONA -	0
SANTA CATARINA IXTAHUACAN -	0
SANTA CATARINA MITA -	0
SANTA CATARINA PINULA -	10
SANTA CLARA LA LAGUNA -	1
SANTA CRUZ BALANYA -	1
SANTA CRUZ DEL QUICHE -	9
SANTA CRUZ LA LAGUNA -	0
SANTA CRUZ MULUA -	0
SANTA CRUZ NARANJO -	0
SANTA CRUZ VERAPAZ -	3
SANTA EULALIA -	0
SANTA LUCIA COTZUMALGUAPA -	3
SANTA LUCIA LA REFORMA -	0
SANTA LUCIA MILPAS ALTAS -	1
SANTA LUCIA UTATLAN -	3
SANTA MARIA CHIQUIMULA -	1
SANTA MARIA DE JESUS -	0
SANTA MARIA IXHUATAN -	1
SANTA MARIA VISITACION -	1
SANTA ROSA DE LIMA -	2
SANTIAGO ATTITLAN -	0
SANTIAGO CHIMALTENANGO -	0
SANTIAGO SACATEPEQUEZ -	2
SANTO DOMINGO SUCHITEPEQUEZ -	2
SANTO DOMINGO XENACOJ -	0
SANTO TOMAS LA UNION -	2
SAYAXCHE -	0
SENAHU -	0
SIBILIA -	0
SIBINAL -	2
SIPACAPA -	0
SIPACATE -	0
SIQUINALA -	0
SOLOLA -	6
SUMPANGO -	4
TACANA -	1
TACTIC -	0
TAJUMULCO -	1
TAMAHU -	0
TAXISCO -	0
TECPAN GUATEMALA -	3
TECTITAN -	0
TECULUTAN -	3
TEJUTLA -	4
TIQUISTATE -	1
TODOS SANTOS CUCHUMATAN -	1
TOTONICAPAN -	15
TUCURU -	0
UNION CANTINIL -	1
USUMATLAN -	1
VILLA CANALES -	8
VILLA NUEVA -	36
YEPOCAPA -	0
YUPILTEPEQUE -	0
ZACAPA -	3
ZACUALPA -	0
ZAPOTITLAN -	0
ZARAGOZA -	2
ZUNIL -	0
ZUNILITO -	0

fallecidos

Departamento vs Muertes Acumuladas

Observaciones

- Además de la casi obvia situación del departamento de Guatemala, la gráfica también nos denota ver la situación del departamento de Quetzaltenango, ya que en anteriores gráficas pudimos ver que solo en el municipio de Quetzaltenango habían aproximadamente 36 fallecimientos, mientras que aquí podemos ver que el

departamento de quetzaltenango completo tiene aproximadamente 63 muertes, lo cual nos indica que la mayor parte proviene de la cabecera municipal.

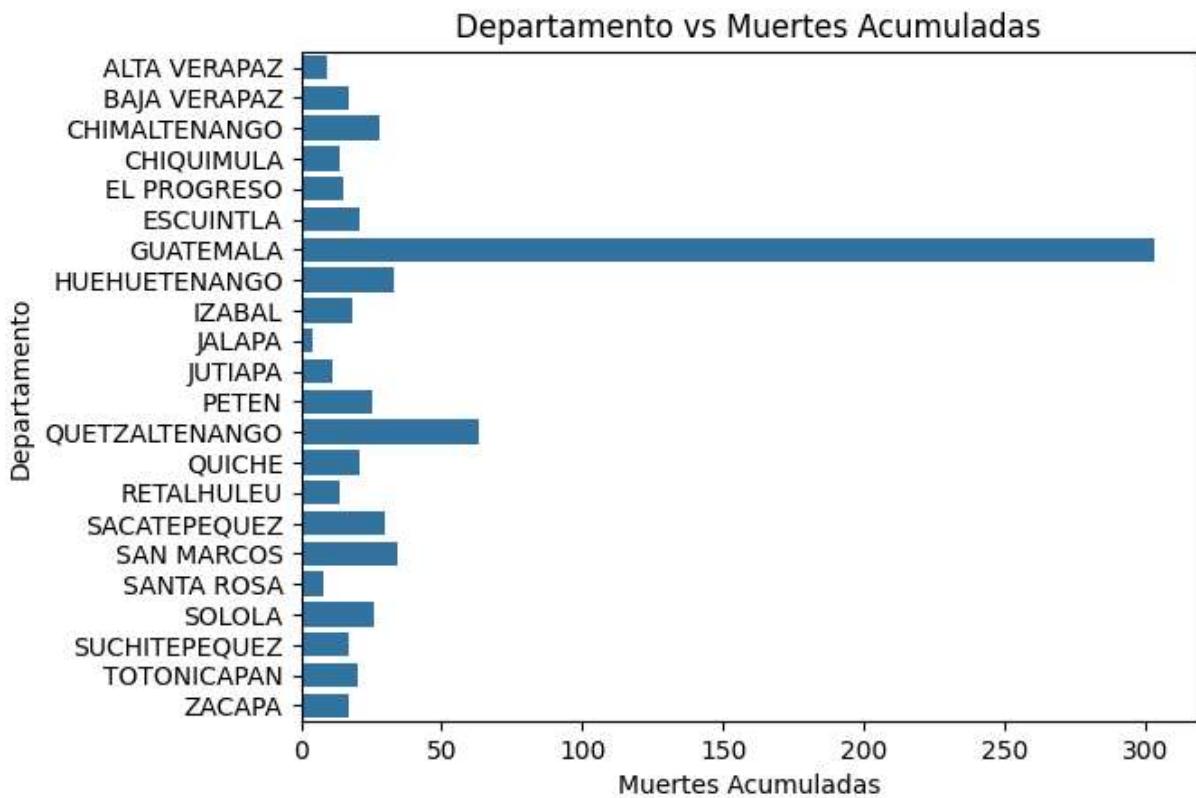
- Importante destacar el manejo de la pandemia en departamentos como Jalapa o Santa Rosa, los cuales no reportan grandes cantidades de fallecimiento.

```
In [39]: sum_deaths_by_municipality = df_municipality_deaths.groupby('nombre_departamento')[

sns.barplot(x='fallecidos', y='nombre_departamento', data=sum_deaths_by_municipalit

# Añadir etiquetas y título
plt.xlabel('Muertes Acumuladas')
plt.ylabel('Departamento')
plt.title('Departamento vs Muertes Acumuladas')

# Mostrar el gráfico
plt.show()
```



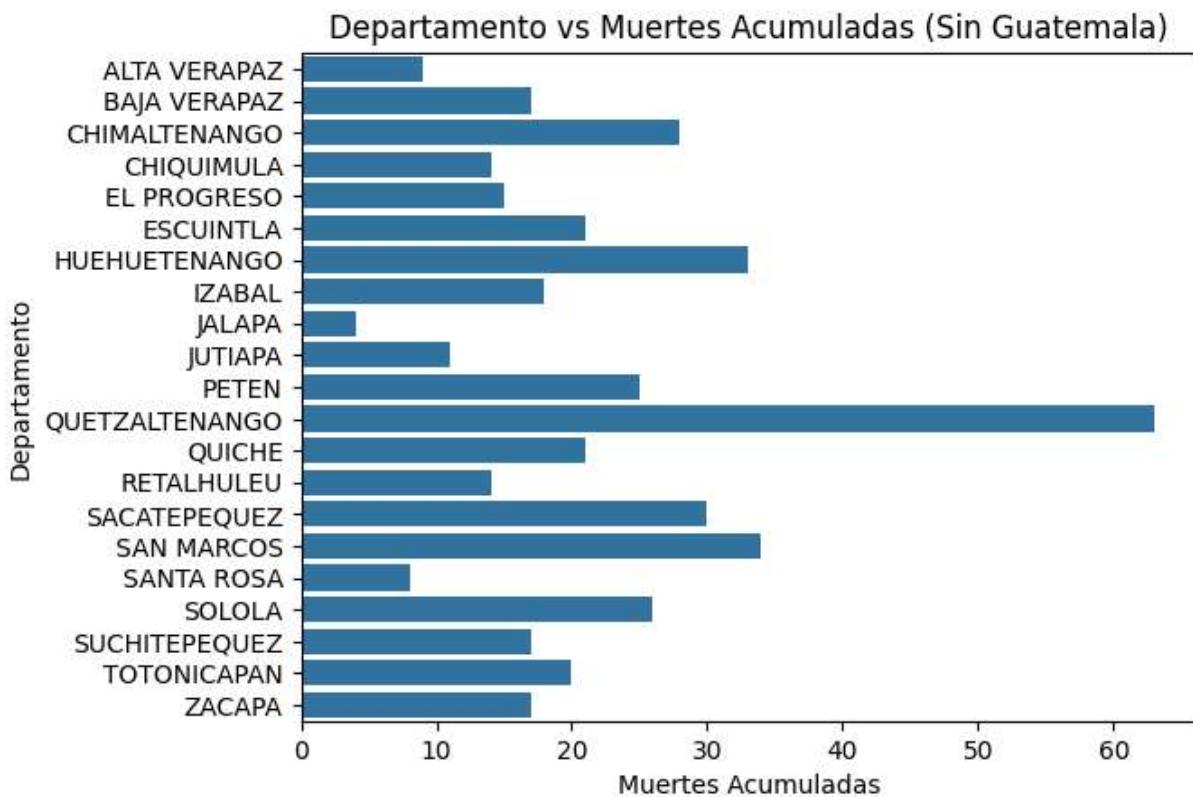
```
In [40]: # Filtrar las filas donde "municipality_name" no sea igual a "GUATEMALA" Transforma
df_filtered = df_municipality_deaths.query('nombre_departamento not in ["GUATEMALA"]')

# Agrupar por "municipality_name" y sumar la cantidad total de muertes
sum_deaths_by_municipality = df_filtered.groupby('nombre_departamento')['fallecidos']

sns.barplot(x='fallecidos', y='nombre_departamento', data=sum_deaths_by_municipalit

# Añadir etiquetas y título
plt.xlabel('Muertes Acumuladas')
plt.ylabel('Departamento')
plt.title('Departamento vs Muertes Acumuladas (Sin Guatemala)')
```

```
# Mostrar el gráfico
plt.show()
```

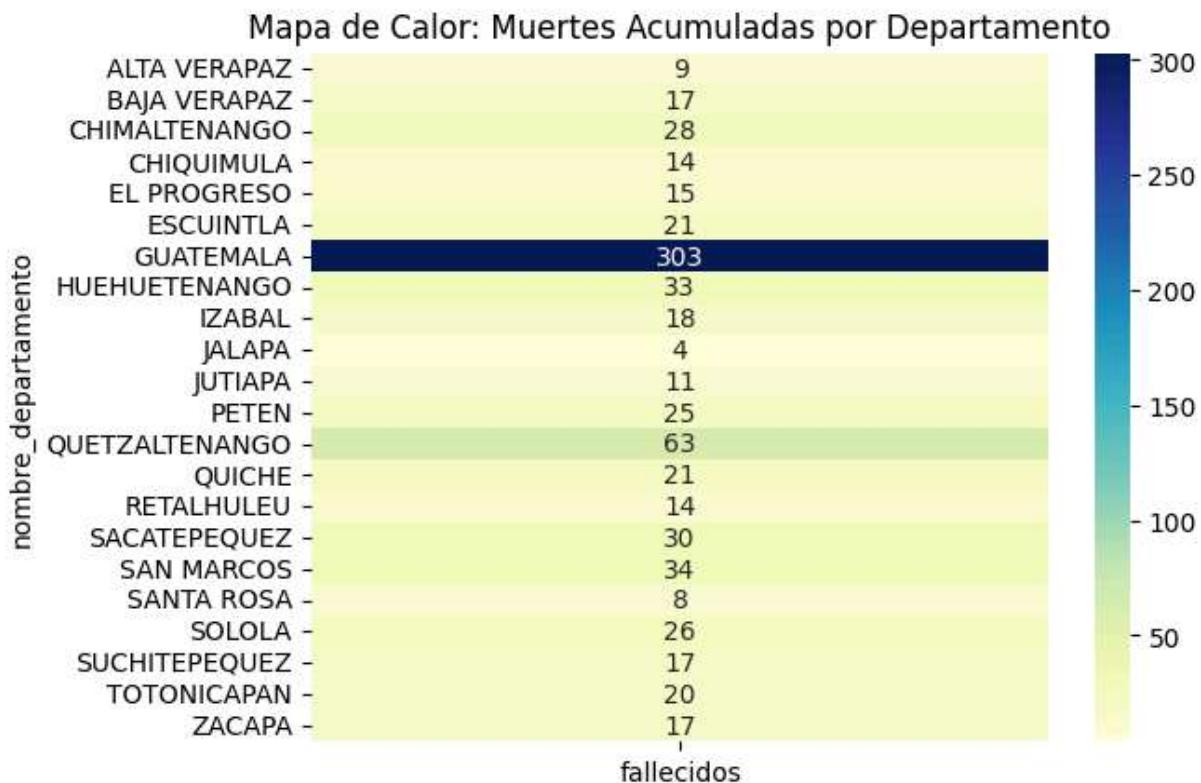


```
In [73]: sum_deaths_by_municipality = df_municipality_deaths.groupby('nombre_departamento')[

# Crear un mapa de calor con seaborn
sns.heatmap(sum_deaths_by_municipality.pivot_table(index='nombre_departamento', val

# Añadir título
plt.title('Mapa de Calor: Muertes Acumuladas por Departamento')

# Mostrar el gráfico
plt.show()
```



Matriz de Correlacion de Pearson

Observaciones (Matriz de Correlacion de Pearson):

El mapa utiliza el coeficiente de correlación de Pearson, que mide la fuerza y dirección de una relación lineal entre dos variables. Un valor de 1 indica una correlación positiva perfecta (cuando una variable aumenta, la otra también), -1 indica una correlación negativa perfecta (cuando una variable aumenta, la otra disminuye), y 0 indica que no hay correlación lineal.

Los valores de correlación en el mapa son muy cercanos a 1, lo que indica una correlación positiva muy fuerte entre todas las variables. Esto significa que:

- A medida que aumenta la población, también aumenta el promedio de nuevas muertes y el total de muertes acumuladas.
- A medida que aumenta el promedio de nuevas muertes, también aumenta el total de muertes acumuladas.

Esta alta correlación positiva es esperable en muchos contextos, especialmente en el análisis de datos epidemiológicos. Una población más grande generalmente tendrá un mayor número de casos y, por lo tanto, más muertes.

```
In [79]: import seaborn as sns
import matplotlib.pyplot as plt
```

```
# Selección de variables para análisis de correlación
variables_numericas = df_municipality_data[['poblacion', 'promedio_nuevas_muertes', 'total_muertes_acumuladas']]

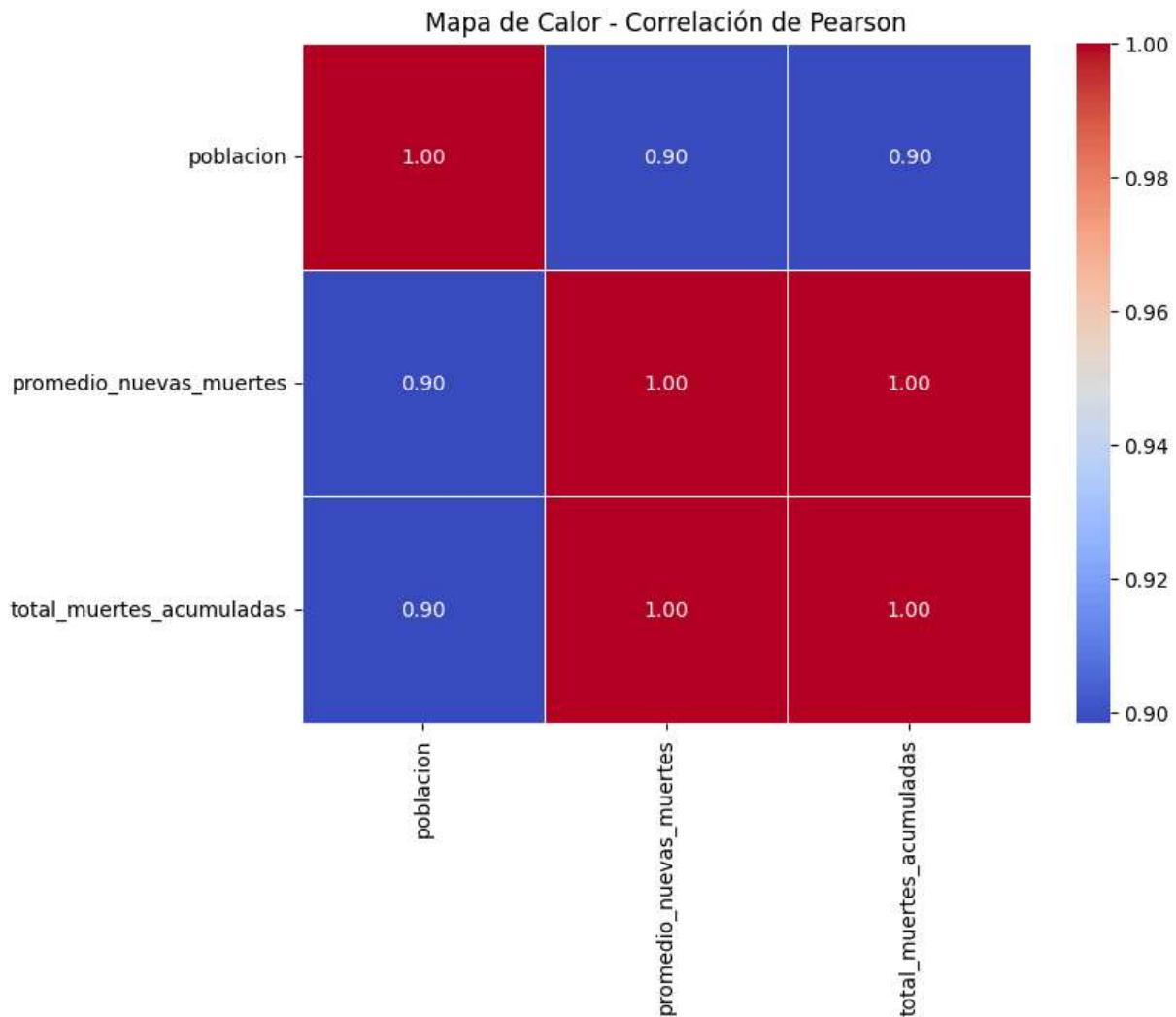
# Matriz de correlación de Pearson
pearson_corr = variables_numericas.corr(method='pearson')
print("Matriz de correlación de Pearson:\n", pearson_corr)

# Gráfico de calor de la correlación
plt.figure(figsize=(8, 6))
sns.heatmap(pearson_corr, annot=True, cmap='coolwarm', fmt='.2f', linewidths=0.5)
plt.title('Mapa de Calor - Correlación de Pearson')
plt.show()
```

Matriz de correlación de Pearson:

	poblacion	promedio_nuevas_muertes	\
poblacion	1.000000	0.898315	
promedio_nuevas_muertes	0.898315	1.000000	
total_muertes_acumuladas	0.898315	1.000000	

	total_muertes_acumuladas		
poblacion		0.898315	
promedio_nuevas_muertes		1.000000	
total_muertes_acumuladas		1.000000	



Observaciones (Correlacion de Kendall):

La correlación entre la población y el promedio de nuevas muertes, así como el total de muertes acumuladas, es de 0.28. Este valor indica una correlación positiva débil. Esto sugiere que a medida que aumenta la población, hay una ligera tendencia a que aumenten también las muertes, pero esta relación no es muy fuerte.

La correlación entre estas dos variables es de 1.00. Esto indica una correlación positiva perfecta. Es lógico, ya que el total de muertes acumuladas es la suma de las nuevas muertes a lo largo del tiempo. Si aumentan las nuevas muertes, necesariamente aumentará también el total acumulado.

Comparación con la matriz de Pearson:

Si comparamos esta matriz con la anterior basada en el coeficiente de Pearson, podemos notar algunas diferencias:

- La correlación de Kendall entre la población y las muertes es más débil que la correlación de Pearson. Esto podría indicar que la relación entre estas variables no es estrictamente lineal.
- Tanto Kendall como Pearson muestran una correlación perfecta entre estas dos variables, lo que confirma la relación directa entre ellas.

Aunque existe una ligera tendencia positiva, la relación entre la población y el número de muertes no es muy fuerte. Esto podría deberse a otros factores que influyen en el número de muertes, como la edad promedio de la población, la tasa de vacunación, etc.

```
In [ ]: # Calcular La matriz de correlación de Kendall
spearman_corr = df_municipality_data[['poblacion', 'promedio_nuevas_muertes', 'total']]
print("Matriz de correlación de Spearman:")
print(spearman_corr)

import seaborn as sns
import matplotlib.pyplot as plt

# Calcular La matriz de correlación de Kendall
kendall_corr = df_municipality_data[['poblacion', 'promedio_nuevas_muertes', 'total']]

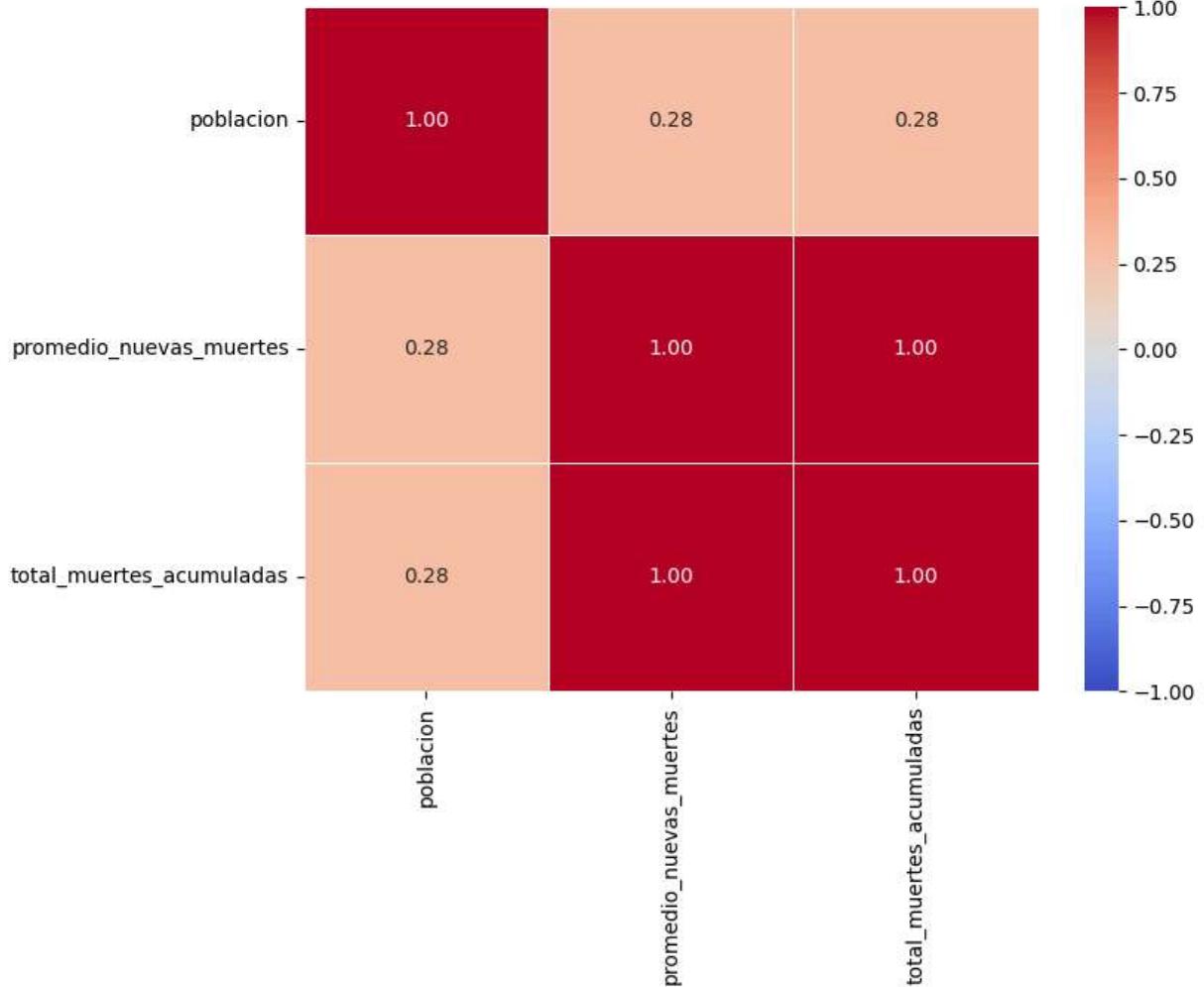
# Graficar La matriz de correlación de Kendall como un mapa de calor
plt.figure(figsize=(8, 6))
sns.heatmap(kendall_corr, annot=True, cmap='coolwarm', linewidths=0.5, fmt='.2f', v
plt.title('Matriz de Correlación de Kendall')
plt.show()
```

Matriz de correlación de Spearman:

	poblacion	promedio_nuevas_muertes	\
poblacion	1.000000	0.365046	
promedio_nuevas_muertes	0.365046	1.000000	
total_muertes_acumuladas	0.365046	1.000000	

	total_muertes_acumuladas
poblacion	0.365046
promedio_nuevas_muertes	1.000000
total_muertes_acumuladas	1.000000

Matriz de Correlación de Kendall



Comparación con la matriz de Pearson y Kendall:

- Los resultados de Spearman son bastante similares a los obtenidos con Kendall. Ambas medidas indican una correlación positiva débil entre la población y las muertes, y una correlación positiva perfecta entre las nuevas muertes y el total acumulado.
- Aunque la tendencia general es similar, los valores específicos de correlación pueden variar ligeramente entre Pearson, Spearman y Kendall. Esto se debe a que cada coeficiente mide diferentes aspectos de la relación entre las variables.

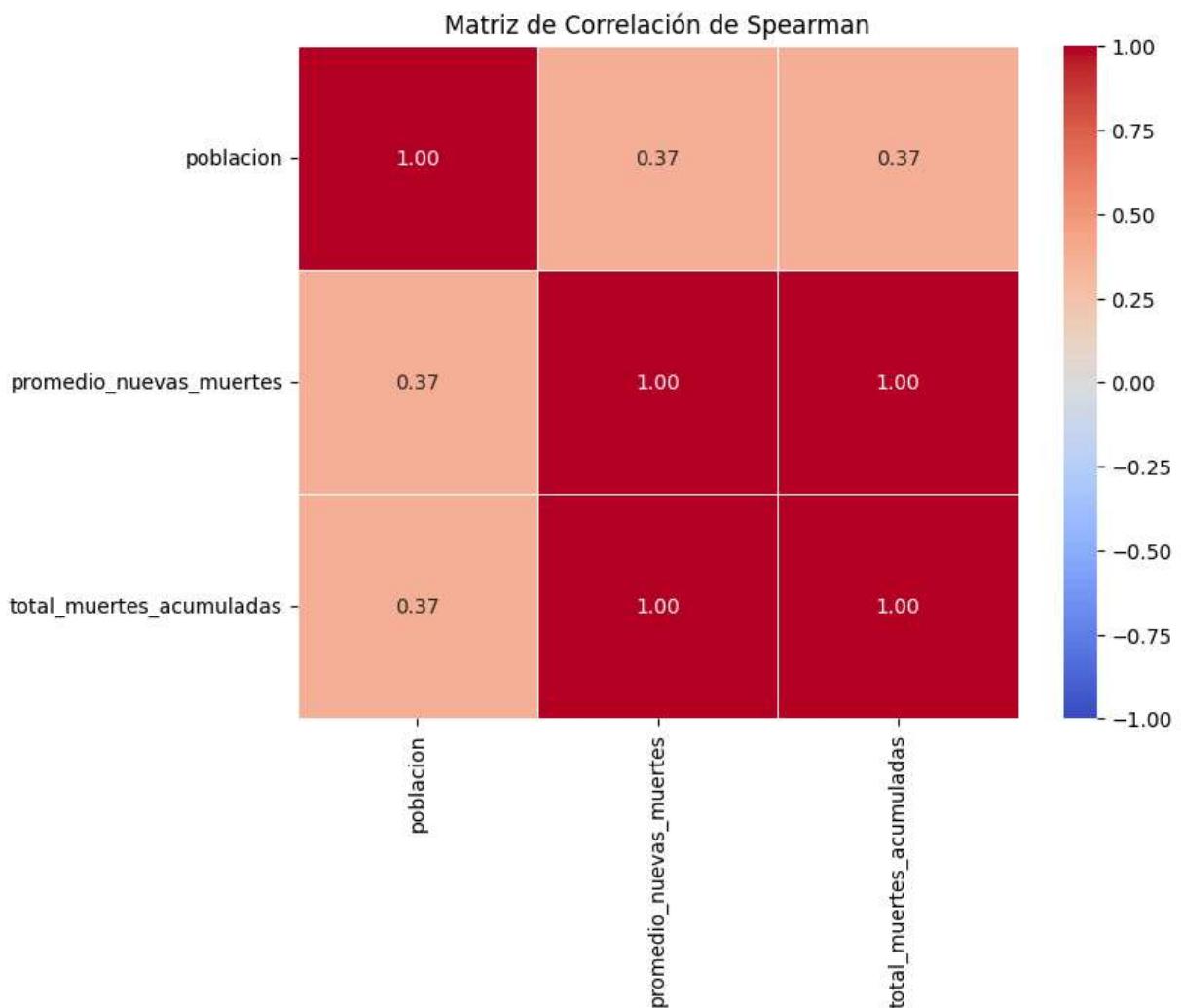
Al igual que en los análisis anteriores, la relación entre la población y el número de muertes es débil. Esto sugiere que otros factores, además del tamaño de la población, influyen en el

número de muertes.

La correlación perfecta entre estas dos variables confirma la relación directa y esperable entre ellas.

```
In [84]: # Calcular La matriz de correlación de Spearman
spearman_corr = df_municipality_data[['poblacion', 'promedio_nuevas_muertes', 'total_muertes_acumuladas']]

# Graficar La matriz de correlación de Spearman como un mapa de calor
plt.figure(figsize=(8, 6))
sns.heatmap(spearman_corr, annot=True, cmap='coolwarm', linewidths=0.5, fmt='.2f',
            plt.title('Matriz de Correlación de Spearman')
            plt.show()
```



Conclusiones:

En base a los datos estadísticos analizados anteriormente se puede concluir lo siguiente:

1. Los municipios que fueron más afectados durante la pandemia COVID-19, fueron aquellos pertenecientes al área urbana y esto puede ser consecuencia del estilo de vida que las personas tienen en estas áreas, además de la población tan grande que habita

estos lugares. Recordemos que en su mayoría el área rural, cuenta con espacios abiertos, que ayudan a que el aire circule de mejor manera que en el área urbana, donde en su mayoría, los espacios donde existe aglomeración de personas, son espacios cerrados, donde el aire no circula con regularidad.

2. Aunque la reactivación económica era crucial, la pandemia dejó claro que el balance entre comercio y salud pública es esencial. Los comerciantes y empresas que implementaron medidas de prevención, como reuniones virtuales, comercio al aire libre y protocolos de bioseguridad, lograron minimizar el impacto de la pandemia en sus comunidades.
3. Algunos municipios rurales, como Chinautla, Coatepeque, Huehuetenango, Puerto Barrios, Santa Catarina Pinula, Santa Lucía Cotzumalguapa, Totonicapán y Zacapa, tuvieron un incremento notable de casos y muertes. A pesar de contar con menor densidad poblacional, estos municipios vieron un aumento en las muertes debido a factores como la falta de infraestructura de salud adecuada, el acceso limitado a servicios médicos y la falta de educación sobre medidas preventivas.
4. En 2020, algunos municipios aún no habían reportado muertes por COVID-19, lo que generó dudas sobre la veracidad de los datos. En algunos casos, podría haberse tratado la monitorización o reporte de los datos o falta de acceso a servicios médicos adecuados. En otros, las prácticas preventivas en la comunidad pudieron haber sido efectivas.
5. A lo largo de la pandemia, varios municipios y departamentos demostraron ser efectivos en la contención del virus, lo que resultó en una menor tasa de mortalidad. Para evitar futuros brotes o picos de muertes, es crucial continuar promoviendo el cumplimiento de las medidas preventivas. Además, se debe fomentar un estilo de vida saludable y preventivo en la población, lo que no solo ayudará a controlar enfermedades infecciosas como el COVID-19, sino que también contribuirá a una mejor calidad de vida a largo plazo.