

# notebook

December 15, 2020

## 1 Pandemia de COVID-19 en Bolivia

### 1.1 1. Antecedentes y Objetivo

#### 1.1.1 1.1 Antecedentes

#### 1.1.2 ¿Que es el COVID-19?

La COVID-19 es la enfermedad infecciosa causada por el coronavirus que se ha descubierto más recientemente. Tanto este nuevo virus como la enfermedad que provoca eran desconocidos antes de que estallara el brote en Wuhan (China) en diciembre de 2019. Actualmente la COVID-19 es una pandemia que afecta a muchos países de todo el mundo.

Los síntomas más habituales de la COVID-19 son la fiebre, la tos seca y el cansancio. Otros síntomas menos frecuentes que afectan a algunos pacientes son los dolores y molestias, la congestión nasal, el dolor de cabeza, la conjuntivitis, el dolor de garganta, la diarrea, la pérdida del gusto o el olfato y las erupciones cutáneas o cambios de color en los dedos de las manos o los pies. Estos síntomas suelen ser leves y comienzan gradualmente. Algunas de las personas infectadas solo presentan síntomas levísimos.

#### 1.1.3 ¿Como llego a Bolivia?

Se confirmó que el virus se había extendido a Bolivia cuando el entonces Ministro de Salud Aníbal Cruz informó los primeros dos casos el 10 de marzo de 2020. Se trataba de dos mujeres de los departamentos de Oruro y Santa Cruz, que habían regresado desde Italia.

El 12 de marzo el gobierno de la presidente Jeanine Áñez adoptó las primeras medidas, declarándose Estado de emergencia sanitaria por COVID-19, cuya duración estaba prevista hasta el 30 de abril de 2020, pero que fue extendida hasta el 10 de mayo de 2020, para posteriormente aplicar la “cuarentena dinámica”. El 1 de agosto de 2020 se decidió una nueva extensión de la cuarentena “condicionada y dinámica” hasta el 31 de agosto de 2020.

El sistema sanitario boliviano se vio prácticamente colapsado a partir de junio de 2020. Los centros de salud llegaron al límite de admisión de pacientes, cientos de personas murieron sin atención hospitalaria, y los cementerios quedaron saturados.

Hasta el 2 de septiembre de 2020 se habían registrado 117 928 casos confirmados, 64 074 recuperaciones y 5203 muertes. La tasa de letalidad (fallecidos respecto a confirmados) es del 4% y la tasa de recuperación (recuperaciones respecto a confirmados) es del 54%.

### 1.1.4 1.2 Objetivo del Notebook

El objetivo de este Notebook es estudiar el brote de COVID-19 en Bolivia, para esto se utilizarán técnicas básicas de visualización. A su vez se realizarán predicciones y pronósticos para estudiar la propagación que tendrá el COVID-19 en Bolivia en los siguientes días.

## 1.2 2. Dataset y Librerías

### 1.2.1 2.1 Acerca del Dataset

El conjunto de datos covid19\_municipios, contiene información de los casos COVID-19 acumulado por cada semana epidemiológica a nivel de municipios, registrados en el Sistema de Información en Salud SNIS (<https://snis.minsalud.gob.bo/>).

El conjunto de datos covid19\_municipios, la cual es base para la elaboración del mapa de REPORTE SEMANAL DE CASOS COVID-19 publicado en la página <https://snis.minsalud.gob.bo/>. La fuente de la información es el Sistema Nacional de Información en Salud y Vigilancia Epidemiológica - SNISVE.

- Fecha de creación: 17 de Julio de 2020, 10:28 (UTC-04:00)
- Última actualización: 1 de Diciembre de 2020, 11:53 (UTC-04:00)

### 1.2.2 2.2 Especificaciones del Dataset

- CODDEPTO: Código del Departamento (Código INE).
- DEPARTAMENTO: Nombre del Departamento.
- Codigo: Código del Municipio.
- MUNICIPIO: Nombre del Municipio.
- SE: Semana epidemiológica.
- CONFIRMADOS: Número de casos confirmados de coronavirus.
- ACTIVOS: Número de casos activos de coronavirus.
- FALLECIDOS: Número de personas fallecidas por coronavirus.
- RECUPERADOS: Número de personas recuperadas de coronavirus.

La semana epidemiológica está definida en el calendario epidemiológico publicado por el SNISVE como se muestra en el cuadro abajo.

### 1.2.3 2.3 Importación de librerías y Dataset

```
[1]: #Importando librerías
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots
import numpy as np
import pandas as pd
from datetime import date, datetime, timedelta
from sklearn import linear_model
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
```

```

from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.model_selection import train_test_split
from math import sqrt
import sys
if not sys.warnoptions:
    import warnings
    warnings.simplefilter("ignore")

#Importando el Dataset
data = pd.read_csv("dataset/consolidado_covid_19-mun_se.csv")

```

## 1.3 3. Preprocesamiento y Análisis Exploratorio

### 1.3.1 3.1 Preprocesamiento

**Identificar valores perdidos** Identificamos si el Dataset contiene valores faltantes con ayuda de librerías y hacemos la suma de estos valores, así como se muestra a continuación.

```
[2]: data.isna().sum()
```

```

[2]: CODDEPTO          0
    DEPARTAMENTO        0
    Codigo              0
    MUNICIPIO           0
    SE                  0
    CONFIRMADOS         0
    ACTIVOS             0
    FALLECIDOS          0
    RECUPERADOS(*)      0
    dtype: int64

```

Con la información obtenida anteriormente, muestras que el Dataset no contiene valores perdidos.

**Agregando una nueva columna** Se agregará una nueva columna fecha con la información Semana epidemiológica (SE) del Dataset. Con el objetivo de una mejor comprensión cuando se realice el análisis exploratorio.

```

[3]: #Agregando una columna al Dataset fecha esto en base a la semana epidemiologica
    list_se = []
    list_se.append(datetime(2019, 12, 22))
    #cincuenta y tres semanas epidemiologicas del año 2020
    for k in range(1, 53):
        list_se.append(list_se[k - 1] + timedelta(days = 7))
    data_se = data['SE']
    semana = []
    for j in data_se.index:
        semana.append(list_se[data_se[j]])
    #Agregando nueva columna

```

```
data['fecha'] = semana
#Guardando el nuevo Dataset copia de seguridad
data.to_csv("dataset/consolidado_covid_19.csv")
```

### 1.3.2 3.2 Análisis Exploratorio

Para entender mejor el Dataset se realizara un pequeño análisis exploratorio, utilizando algunas herramientas estadísticas junto con visualizaciones.

```
[4]: #Mostrando numero de registros, valores nulos y tipos de datos del Dataset
print("Numero de registros del Dataset: ", data['CODDEPTO'].count())
print("Verificando si existen valores nulos: ", data.isnull().any().any())
print("Verificando Tipo de Dato de cada columna:")
data.dtypes
```

```
Numero de registros del Dataset: 6144
Verificando si existen valores nulos: False
Verificando Tipo de Dato de cada columna:
```

```
[4]: CODDEPTO          int64
DEPARTAMENTO         object
Codigo              int64
MUNICIPIO           object
SE                  int64
CONFIRMADOS         int64
ACTIVOS             int64
FALLECIDOS          int64
RECUPERADOS(*)      int64
fecha               datetime64[ns]
dtype: object
```

```
[5]: #Analisis estadistico del Dataset
data.describe(include = "all", datetime_is_numeric = True)
```

```
[5]:
```

|        | CODDEPTO    | DEPARTAMENTO | Codigo       | MUNICIPIO | SE \        |
|--------|-------------|--------------|--------------|-----------|-------------|
| count  | 6144.000000 | 6144         | 6144.000000  | 6144      | 6144.000000 |
| unique | NaN         | 9            | NaN          | 322       | NaN         |
| top    | NaN         | LA PAZ       | NaN          | MINEROS   | NaN         |
| freq   | NaN         | 1347         | NaN          | 25        | NaN         |
| mean   | 4.378255    | NaN          | 44498.364095 | NaN       | 35.847005   |
| min    | 1.000000    | NaN          | 10101.000000 | NaN       | 20.000000   |
| 25%    | 2.000000    | NaN          | 21203.000000 | NaN       | 30.000000   |
| 50%    | 4.000000    | NaN          | 40505.000000 | NaN       | 36.000000   |
| 75%    | 7.000000    | NaN          | 70501.000000 | NaN       | 44.000000   |
| max    | 9.000000    | NaN          | 90503.000000 | NaN       | 48.000000   |
| std    | 2.407762    | NaN          | 23989.887463 | NaN       | 7.736337    |

|  | CONFIRMADOS | ACTIVOS | FALLECIDOS | RECUPERADOS(*) \ |
|--|-------------|---------|------------|------------------|
|--|-------------|---------|------------|------------------|

|        |              |              |             |              |
|--------|--------------|--------------|-------------|--------------|
| count  | 6144.000000  | 6144.000000  | 6144.000000 | 6144.000000  |
| unique | NaN          | NaN          | NaN         | NaN          |
| top    | NaN          | NaN          | NaN         | NaN          |
| freq   | NaN          | NaN          | NaN         | NaN          |
| mean   | 349.277018   | 125.211426   | 15.833008   | 208.232585   |
| min    | 0.000000     | 0.000000     | 0.000000    | 0.000000     |
| 25%    | 5.000000     | 1.000000     | 0.000000    | 0.000000     |
| 50%    | 23.000000    | 7.000000     | 1.000000    | 7.000000     |
| 75%    | 96.000000    | 33.000000    | 4.000000    | 36.000000    |
| max    | 29486.000000 | 16677.000000 | 2040.000000 | 25806.000000 |
| std    | 1958.170217  | 650.373054   | 98.757952   | 1524.478155  |

|        |                               |
|--------|-------------------------------|
|        | fecha                         |
| count  | 6144                          |
| unique | NaN                           |
| top    | NaN                           |
| freq   | NaN                           |
| mean   | 2020-08-28 22:17:48.750050048 |
| min    | 2020-05-10 00:00:00           |
| 25%    | 2020-07-19 00:00:00           |
| 50%    | 2020-08-30 00:00:00           |
| 75%    | 2020-10-25 00:00:00           |
| max    | 2020-11-22 00:00:00           |
| std    | NaN                           |

```
[6]: #Mostrando cabecera del dataset
data.head()
```

```
[6]: CODDEPTO DEPARTAMENTO  Codigo  MUNICIPIO  SE  CONFIRMADOS  ACTIVOS  \
0          1    CHUQUISACA   10101      SUCRE    20           14        11
1          1    CHUQUISACA   10501  MONTEAGUDO    20            1         1
2          1    CHUQUISACA   10902    CULPINA    20            1         1
3          2        LA PAZ   20101    LA PAZ    20          152        59
4          2        LA PAZ   20105    EL ALTO    20          126        70
```

|   |            |                |            |
|---|------------|----------------|------------|
|   | FALLECIDOS | RECUPERADOS(*) | fecha      |
| 0 | 2          | 1              | 2020-05-10 |
| 1 | 0          | 0              | 2020-05-10 |
| 2 | 0          | 0              | 2020-05-10 |
| 3 | 8          | 85             | 2020-05-10 |
| 4 | 8          | 48             | 2020-05-10 |

```
[7]: #Mostrando tipos de casos por semana epidemiologica en Bolivia
casos_semana = data.groupby(["fecha"]).agg({"CONFIRMADOS": 'sum', "ACTIVOS":
→ 'sum', "FALLECIDOS": 'sum', "RECUPERADOS(*)": 'sum'})
print(casos_semana)
```

|             |         |            |                |
|-------------|---------|------------|----------------|
| CONFIRMADOS | ACTIVOS | FALLECIDOS | RECUPERADOS(*) |
|-------------|---------|------------|----------------|

| fecha      |        |       |      |        |
|------------|--------|-------|------|--------|
| 2020-05-10 | 3826   | 3247  | 165  | 414    |
| 2020-05-17 | 5916   | 5128  | 239  | 549    |
| 2020-05-24 | 9592   | 8469  | 310  | 813    |
| 2020-05-31 | 13358  | 11002 | 454  | 1902   |
| 2020-06-07 | 17842  | 14489 | 585  | 2768   |
| 2020-06-14 | 23512  | 17686 | 740  | 5086   |
| 2020-06-28 | 38071  | 25421 | 1378 | 11272  |
| 2020-07-05 | 47200  | 31113 | 1754 | 14333  |
| 2020-07-12 | 58138  | 37832 | 2106 | 18200  |
| 2020-07-19 | 68281  | 44795 | 2535 | 20951  |
| 2020-07-26 | 78793  | 51769 | 3056 | 23968  |
| 2020-08-02 | 89055  | 56564 | 3587 | 28904  |
| 2020-08-09 | 99146  | 59505 | 4003 | 35638  |
| 2020-08-16 | 108427 | 63401 | 4442 | 40584  |
| 2020-08-23 | 115354 | 54176 | 4938 | 56240  |
| 2020-08-30 | 120241 | 45756 | 5398 | 69087  |
| 2020-09-06 | 125982 | 37938 | 5727 | 82317  |
| 2020-09-13 | 132313 | 39410 | 6018 | 86885  |
| 2020-09-27 | 136569 | 33044 | 6503 | 97022  |
| 2020-10-04 | 138463 | 31039 | 6721 | 100703 |
| 2020-10-25 | 141757 | 23264 | 7207 | 111286 |
| 2020-11-01 | 142427 | 21187 | 7272 | 113968 |
| 2020-11-08 | 143181 | 19275 | 7323 | 116583 |
| 2020-11-15 | 143922 | 17631 | 7386 | 118905 |
| 2020-11-22 | 144592 | 16158 | 7431 | 121003 |

```
[8]: #Grafica de crecimiento de tipo de casos por semana epidemiologica
numero_casos_semana=[]
confirmados_semana=[]
recuperados_semana=[]
fallecidos_semana=[]

for i in list(casos_semana.index):
    confirmados_semana.append(casos_semana[casos_semana.index==i]["CONFIRMADOS"].
    →iloc[-1])
    recuperados_semana.append(casos_semana[casos_semana.
    →index==i]["RECUPERADOS(*)"].iloc[-1])
    fallecidos_semana.append(casos_semana[casos_semana.index==i]["FALLECIDOS"].
    →iloc[-1])
    numero_casos_semana.append(i)

fig=go.Figure()
fig.add_trace(go.Scatter(x=numero_casos_semana, y=confirmados_semana,
    →mode='lines+markers', name='Crecimiento semanal de casos confirmados'))
fig.add_trace(go.Scatter(x=numero_casos_semana, y=recuperados_semana,
    →mode='lines+markers', name='Crecimiento semanal de casos recuperados'))
```

```
fig.add_trace(go.Scatter(x=numero_casos_semana, y=fallecidos_semana,
    →mode='lines+markers', name='Crecimiento semanal de casos fallecidos'))
fig.update_layout(title="Crecimiento semanal de diferentes tipos de casos en
    →Bolivia", xaxis_title="Numero de semana",yaxis_title="Numero de
    →casos",legend=dict(x=0,y=1,traceorder="normal"))
fig.show()
```

```
[9]: # Seleccionando datos por departament
casos_depto = data.groupby(["DEPARTAMENTO"]).agg({"CONFIRMADOS":'sum',"ACTIVOS":
    →'sum',"FALLECIDOS":'sum',"RECUPERADOS(*)":'sum'})
# Grafica en barras
# Casos confirmados por departamento
fig=px.bar(x=casos_depto.index, y=casos_depto["CONFIRMADOS"])
fig.update_layout(title="Casos confirmados por departamento",
    →xaxis_title="Departamento",yaxis_title="Numero de casos confirmados SE 46")
fig.show()
```

```
[10]: # Casos activos por departamento
# Grafica en barras
fig=px.bar(x=casos_depto.index, y=casos_depto["ACTIVOS"])
fig.update_layout(title="Casos activos por departamento",
    →xaxis_title="Departamento",yaxis_title="Numero de casos activos SE 46")
fig.show()
```

```
[11]: # Casos activos por departamento
# Grafica en barras
fig=px.bar(x=casos_depto.index, y=casos_depto["FALLECIDOS"])
fig.update_layout(title="Casos fallecidos por departamento",
    →xaxis_title="Departamento",yaxis_title="Numero de casos fallecidos SE 46")
fig.show()
```

```
[12]: # Casos activos por departamento
# Grafica en barras
fig=px.bar(x=casos_depto.index, y=casos_depto["RECUPERADOS(*)"])
fig.update_layout(title="Casos recuperados por departamento",
    →xaxis_title="Departamento",yaxis_title="Numero de casos recuperados SE 46")
fig.show()
```

```
[13]: #Grafica en barras distribucion del numero de casos activos en Bolivia por
    →semana epidemiologica
fig=px.bar(x=casos_semana.index,
    →y=casos_semana["CONFIRMADOS"]-casos_semana["RECUPERADOS(*)"]-casos_semana["FALLECIDOS"])
fig.update_layout(title="Distribución del número de casos activos Bolivia",
    →xaxis_title="Semana Epidemiologica",yaxis_title="Numero de casos Bolivia")
fig.show()
```

**Casos activos = Número de casos confirmados - Número de casos recuperados - Número de**

## casos de fallecidos

La disminucion en el numero de casos activos es probablemente una indicacion de aumento en el numero de casos de personas fallecidas y el numero de casos de personas recuperadas.

```
[14]: #grafica en barrar para la distribucion del numero de casos cerrados por semana
      ↳epidemiologica
fig=px.bar(x=casos_semana.
      ↳index,y=casos_semana["RECUPERADOS(*)"]+casos_semana["FALLECIDOS"])
fig.update_layout(title="Distribución del número de casos cerrados_
      ↳Bolivia",xaxis_title="Semana",yaxis_title="Numero de casos Bolivia")
fig.show()
```

## Casos cerrados = Número de casos recuperados + Número de casos de muerte

El aumento en el número de casos cerrados indica que más pacientes se están recuperando de la enfermedad o más personas están muriendo a causa de COVID-19.

```
[15]: #Calculando el porcentaje de casos fallecidos y porcentaje de casos recuperados
casos_semana["tasa de mortalidad"]=(casos_semana["FALLECIDOS"] /
      ↳casos_semana["CONFIRMADOS"])*100
casos_semana["indice de recuperacion"]=(casos_semana["RECUPERADOS(*)"] /
      ↳casos_semana["CONFIRMADOS"])*100
casos_semana["casos_
      ↳activos"]=casos_semana["CONFIRMADOS"]-casos_semana["RECUPERADOS(*)"]-casos_semana["FALLECIDOS"]
casos_semana["casos_
      ↳cerrados"]=casos_semana["RECUPERADOS(*)"]+casos_semana["FALLECIDOS"]

print("Media de fallecidos",casos_semana["tasa de mortalidad"].mean())
print("Mediana de fallecidos",casos_semana["tasa de mortalidad"].median())
print("Media de recuperados",casos_semana["indice de recuperacion"].mean())
print("Mediana de recuperados",casos_semana["indice de recuperacion"].median())

#Plotting tasa de falleciminetos e indice de recuperacion
fig = make_subplots(rows=2, cols=1,
      subplot_titles=("Indice de Recuperacion Bolivia", "Tasa de_
      ↳Fallecimientos Bolivia"))
fig.add_trace(
    go.Scatter(x=casos_semana.index, y=(casos_semana["RECUPERADOS(*)"] /
      ↳casos_semana["CONFIRMADOS"])*100,name="Indice de recuperacion Bolivia"),
    row=1, col=1
)
fig.add_trace(
    go.Scatter(x=casos_semana.index, y=(casos_semana["FALLECIDOS"] /
      ↳casos_semana["CONFIRMADOS"])*100,name="Tasa de mortalidad Bolivia"),
    row=2, col=1
)
fig.update_layout(height=1000,legend=dict(x=-0.1,y=1.2,traceorder="normal"))
```



```
fig.update_xaxes(title_text="Fecha", row=1, col=1)
fig.update_yaxes(title_text="Indice de recuperacion Bolivia", row=1, col=1)
fig.update_xaxes(title_text="Fecha", row=1, col=2)
fig.update_yaxes(title_text="Tasa de mortalidad Bolivia", row=1, col=2)
fig.show()
```

Media de fallecidos 4.207038578820469  
 Mediana de fallecidos 4.096765565772363  
 Media de recuperados 45.01652706040434  
 Mediana de recuperados 35.94497004417727

**Tasa de mortalidad = (Número de casos de muerte / Número de casos confirmados) x 100**

**Tasa de recuperación = (Número de casos recuperados / Número de casos confirmados) x 100**

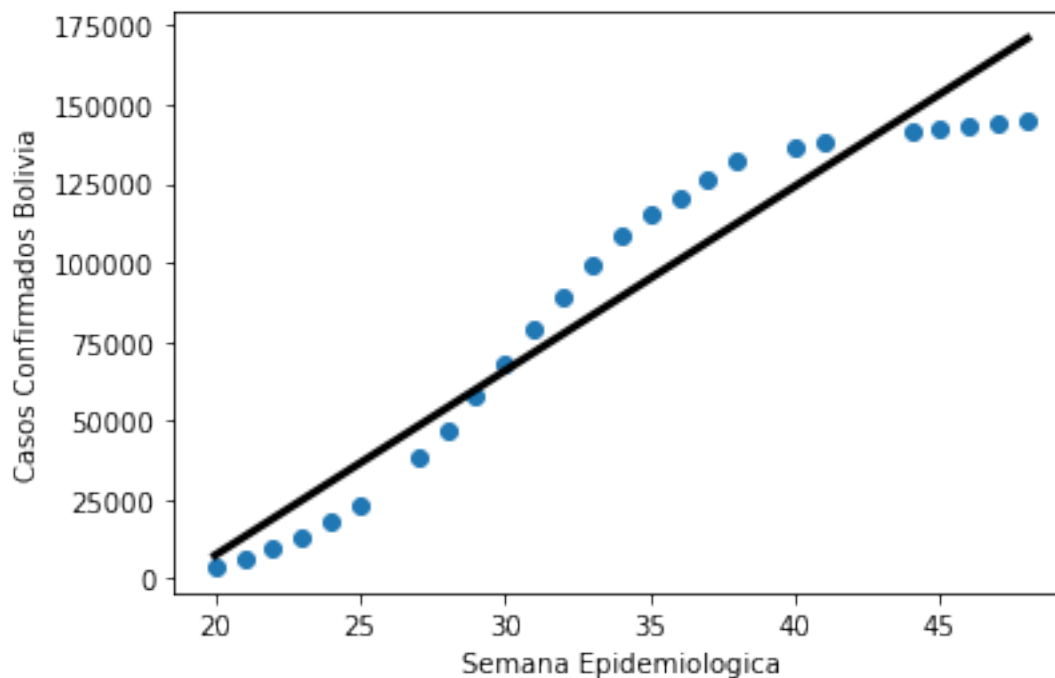
La tasa de mortalidad está mostrando un nivel considerable durante bastante tiempo, lo cual es un signo positivo. La tasa de recuperación ha comenzado a recuperarse, lo que es una buena señal, otra razón de apoyo por la que el número de casos cerrados está aumentando.

## 1.4 4. Predicciones usando modelos

### Modelo de regresión lineal para la predicción de casos confirmados en Bolivia

```
[16]: from sklearn.linear_model import LinearRegression
x = list(set(data['SE']))
x = np.array(x).reshape(-1, 1)
y = data.groupby(["SE"]).agg({"CONFIRMADOS" : 'sum'})
y = np.array(y).reshape(-1, 1)
reg = LinearRegression()
predict_space = np.linspace(min(x), max(x)).reshape(-1, 1)
reg.fit(x,y)
predicted = reg.predict(predict_space)
print('R^2 score: ',reg.score(x, y))
plt.plot(predict_space, predicted, color = 'black', linewidth = 3)
plt.scatter(x = x,y = y)
plt.xlabel('Semana Epidemiologica')
plt.ylabel('Casos Confirmados Bolivia')
plt.show()
```

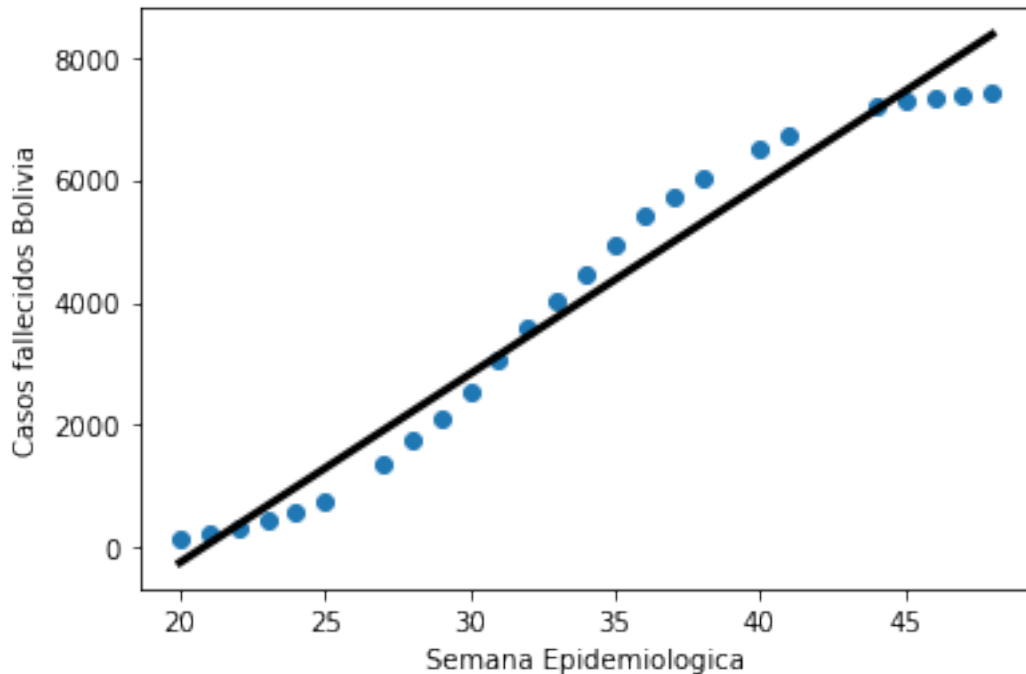
R^2 score: 0.925212258738751



### Modelo de regresión lineal para la predicción de casos fallecidos en Bolivia

```
[17]: from sklearn.linear_model import LinearRegression
x = list(set(data['SE']))
x = np.array(x).reshape(-1, 1)
y = data.groupby(["SE"]).agg({"FALLECIDOS" : 'sum'})
y = np.array(y).reshape(-1, 1)
reg = LinearRegression()
predict_space = np.linspace(min(x), max(x)).reshape(-1, 1)
reg.fit(x,y)
predicted = reg.predict(predict_space)
print('R^2 score: ',reg.score(x, y))
plt.plot(predict_space, predicted, color = 'black', linewidth = 3)
plt.scatter(x = x,y = y)
plt.xlabel('Semana Epidemiologica')
plt.ylabel('Casos fallecidos Bolivia')
plt.show()
```

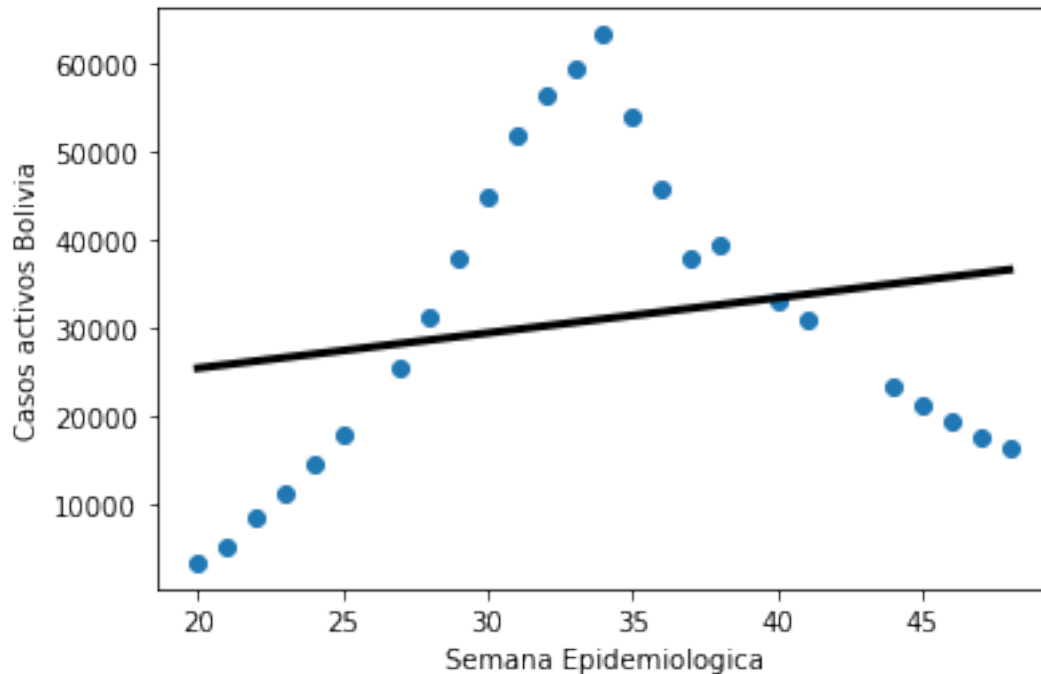
R<sup>2</sup> score: 0.9668421186171398



### Modelo de regresión lineal para la predicción de casos activos en Bolivia

```
[18]: from sklearn.linear_model import LinearRegression
x = list(set(data['SE']))
x = np.array(x).reshape(-1, 1)
y = data.groupby(["SE"]).agg({"ACTIVOS" : 'sum'})
y = np.array(y).reshape(-1, 1)
reg = LinearRegression()
predict_space = np.linspace(min(x), max(x)).reshape(-1, 1)
reg.fit(x,y)
predicted = reg.predict(predict_space)
print('R^2 score: ',reg.score(x, y))
plt.plot(predict_space, predicted, color = 'black', linewidth = 3)
plt.scatter(x = x,y = y)
plt.xlabel('Semana Epidemiologica')
plt.ylabel('Casos activos Bolivia')
plt.show()
```

R<sup>2</sup> score: 0.03783108072722641



### Modelo de regresión lineal para la predicción de casos confirmados en La Paz

```
[19]: from sklearn.linear_model import LinearRegression
df = data.loc[data.loc[:, 'DEPARTAMENTO'] == 'LA PAZ']
lp = df.groupby(["DEPARTAMENTO"]).agg({"CONFIRMADOS": 'sum', "ACTIVOS":
    → 'sum', "FALLECIDOS": 'sum', "RECUPERADOS(*)": 'sum'})
x = list(set(df['SE']))
x = np.array(x).reshape(-1, 1)
y = df.groupby(["SE"]).agg({"CONFIRMADOS" : 'sum'})
y = np.array(y).reshape(-1, 1)
reg = LinearRegression()
predict_space = np.linspace(min(x), max(x)).reshape(-1, 1)
reg.fit(x,y)
predicted = reg.predict(predict_space)
print('R^2 score: ', reg.score(x, y))
plt.plot(predict_space, predicted, color = 'black', linewidth = 3)
plt.scatter(x = x,y = y)
plt.xlabel('Semana Epidemiologica')
plt.ylabel('Casos Confirmados')
plt.show()
```

R^2 score: 0.8817917202841081

