
DETERMINACIÓN DE VOLÚMENES DE CAUDALES Y SU UBICACIÓN EN PYTHON

Campos Rodríguez José Luis (1), Gutiérrez Camacho Reyes Omar (2), Vega Solorio Karla Leticia (3)

(1) Centro de cómputo de la Facultad de Ingeniería Civil, Campus Coquimatlán, Km. 9 Carretera, Colima - Coquimatlán, Jardines del Llano, 28400 Coquimatlán, Col., jcampos6@ucol.mx, rgutierrez34@ucol.mx, kvega0@ucol.mx

Resumen

Se generará un programa en Python el cual tiene como propósito leer un archivo de Excel, en el que se encuentran datos de un levantamiento de secciones sobre un canal que se encuentra en el campus Coquimatlán en la Facultad de Ingeniería Civil, para obtener como resultado que el programa genere de manera automática el volumen obtenido de cierto canal proporcionándole ciertos datos clave, también para un mayor entendimiento visual graficar los resultados.

Palabras clave: 1. Programa. 2. Python. 3. Canales

Abstract

A Python program will be generated whose purpose is to read an Excel file, in which data from a survey of sections on a canal located on the Coquimatlán campus in the Faculty of Civil Engineering are found, to obtain as a result that the program automatically generates the volume obtained from a certain channel, providing you with certain key data, also for a better visual understanding, graph the results.

Keywords: 1. Program 2. Python 3. Channels.

1. Introducción

El presente informe se hablara del cálculo de volúmenes apoyándose de la área de geolocalización por medio del uso de Python y Google Colab, así como se pretende la elaboración de un código en lenguaje Python que nos sea de ayuda para realizar cálculos sobre volúmenes, velocidad del agua por segundo en los canales así como la graficacion de una sección en particular del canal, con el uso de distintas bases de la programación lograr que nuestro código Python pueda mostrarnos la localización del canal mediante el uso de distintas librerías, además poder identificar si la pendiente es positiva o negativa, como elemento un poco más enfocado a lo geoespacial se tiene planeado realizar un

apartado en el código que nos permita que mediante las coordenadas del lugar nos arroje la ubicación del canal en donde se realizó el trabajo.

Dicho todo esto, daremos inicio al informe comenzando con el desarrollo de este proyecto.

2. Desarrollo

Primeramente, antes de comenzar la realización de los programas y mostrar el resultado final, se realizan diversas investigaciones en las cuales se tiene como objetivo tener una idea más clara de cuál será el proceso que se deberá seguir para que nuestros códigos de Python sean lo suficientemente satisfactorio.

Hay que destacar que este proyecto tiene gran importancia, ya que se va a evaluar la calidad del producto de una serie de procedimientos que se realizaron para poder así llevarlo a cabo y los resultados que se obtendrán sean favorables.

Los datos sobre la sección en específico se toman de una practica topográfica la cual consistía en el levantamiento de secciones transversales justamente de un caudal de interés dentro de las instalaciones de la facultad, con esto obtuvimos los datos mas relevantes y de importancia para la realización de este proyecto final.

Algunas de las herramientas / materiales que utilizamos y fueron fundamentales necesitan para poder llevar a cabo este proyecto final son:

- Nivel de mano
- Estadales
- Longimetro
- Python
- Google Colab.
- Excel

Posterior a esto se procede a investigar a que tipo de canal pertenece el canal en el que realizamos el estudio, esto para saber como calcular el volumen y hacer que nuestros datos sean los necesarios para realizar esos cálculos correspondientes en Python.

Se comienza la investigación sobre librerías de Python que se puedan utilizar para poder darnos la ubicación del área de estudio, esto va enfocado mas a lo geoespacial, entre estas librerías se encuentran Basemap, Folium, entre otras, las cuales nos pueden marcar mediante un punto donde se encuentra el lugar de trabajo.

1. Exportamos los datos obtenidos de las secciones a una hoja de cálculo en Excel, para tener un mejor manejo de

	A	B
1	Cadenamiento	Cotas
2	0+000	353.285
3	0+020	351.197
4	0+040	350.000
5	0+053	349.465

los datos.

Imagen 1. Datos del canal.

2. Subimos nuestro archivo Excel a nuestro drive y procedemos a crear el código que nos permitirá graficar la sección de dicho canal ya mencionado, al código tenemos que ponerle el nombre del archivo que subimos a drive y ejecutara los datos de las secciones.

```

+ Código + Texto
[6] import folium
mapa = folium.Map(location=[19.21120633201805, -98.80469475848845])
mapa

from google.colab import files
from google.colab import drive
import matplotlib.pyplot as plt
import pandas as pd
import xrd

drive.mount('/gdrive')
df = pd.read_excel('/gdrive/MyDrive/Colab Notebooks/SECCIONES/SECCIONES.xlsx', sheet_name='Hoja1')

VALORES = df[['Cadenamientos', 'Cotas']]
ax = VALORES.plot(x='Cadenamientos', y='Cotas', rot=0)
plt.xlabel('Cadenamientos')
plt.ylabel('Cotas')
plt.show

```

Imagen 2. Código grafico de secciones con el uso de la librería matplotlib.

3. Ejecutamos y nos dará el resultado de la sección referida de dicho canal graficada con el uso de las librerías.

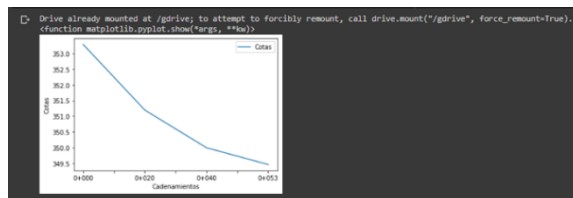


Imagen 3. Resultado grafico obtenido del código grafico de secciones.

4. Además, este programa nos permitirá conocer dicha ubicación del canal ubicado en la Facultad de Ingeniería Civil.

```
# Código geoespacial para obtener la ubicación.

%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.basemap import Basemap

fig = plt.figure(figsize=(8, 8))
m = Basemap(projection='ortho', resolution=None,
            width=8E6, height=8E6,
            lat_0=45, lon_0=-100,)
m.etopo(scale=1.0, alpha=1.0)

# Map (long, lat) to (x, y) for plotting
x, y = m(-103.80450468775071, 19.21095061186164)
plt.plot(x, y, 'ok', markersize=5)
plt.text(x, y, 'Caudal_FIC', fontsize=22);
```

Imagen 4. Código Python geoespacial.

5. Ya como último paso que realizamos fue crear el programa en Python para poder calcular el volumen (caudal) de dicho canal con los datos obtenidos, para realizar el código tomamos en cuenta las características de los canales y ciertos aspectos que nos parecieron interesantes para poder conocer el volumen (caudal), a partir de ciertos datos a ingresar de forma manual al programa. otra parte interesante que tiene el código de programación es que se pueden integrar los datos de manera manual, para poder así tener el manejo de los datos.

```
[ ] #Código para calculo de volumen canal

# Datos de entrada
base = float(input("Base(m) : "))
z = float(float(input("Z : ")))
S = float(input("pendiente de fondo S: "))
n = float(input("rugorosidad : "))
y = float(input("tirante normal (m) : "))

# Calculos
A = base*y+z*y**2
P = base+2*y*(1+z**2)**.5
R = round(A/P,2)
print('El radio hidraulico es (m) : ', R)

#Velocidad
V = round(1/(n)*R**(2/3)*S**(.5,2)
print('La velocidad del agua es de (m/s) : ', V)
Q = round(V*A,2)
print('El volumen es de (m3):',Q)
```

Imagen 6. Código Python Volumen canal

La ejecución del código mostro lo siguiente:



Imagen 5. Código ejecutado resultado con el uso de las librerías matplotlib y basemap.

Para el cálculo del volumen del canal en Python realizamos un código que a partir de las características del canal nos permitiera conocer cuál sería su volumen.

Resultado obtenido después de la ejecución:

```
Base(m) : 1.3
Z : 2
pendiente de fondo S: 3
rugorosidad : 1
tirante normal (m) : 12
El radio hidraulico es (m) : 5.52
La velocidad del agua es de (m/s) : 5.41
El volumen es de (m3/s): 1642.48
```

Imagen 7. Resultado código ejecutado cálculo de volúmenes

6. Se realizó también dos códigos en colab los cuales también van de entrada en nuestra parte geoespacial del proyecto realizado, el primer código realizado nos permite obtener la ubicación del lugar de estudio en un mapa interactivo implementando la librería folium.

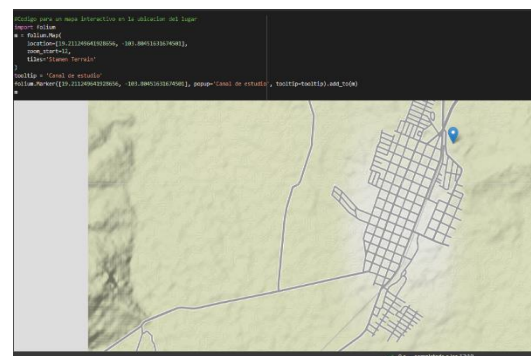


Imagen 8. Resultado mapa interactivo con folium.

3. Manejo de datos

Las librerías utilizadas para realizar los códigos correspondientes utilizando el lenguaje Python de este proyecto son las siguientes:

- **Numpy:** La librería NumPy nos permitió una generación y manejo de datos extremadamente rápido. NumPy tiene su propia estructura de datos incorporada llamado arreglo que es similar a la lista normal de Python, pero puede almacenar y operar con datos de manera mucho más eficiente.
- **Matplotlib:** Esta librería nos ayudo a realizar el grafico de las secciones del canal. Matplotlib es una librería de Python especializada en la creación de gráficos en dos dimensiones. Permite crear y personalizar los tipos de gráficos más comunes, entre ellos: Diagramas de barras.

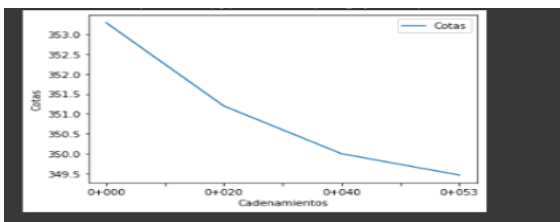


Imagen 9. Código ejecutado resultado con el uso de las librerías matplotlib.

- **Basemap:** Basemap no es exactamente una librería como tal. De hecho, es un conjunto de herramientas, una extensión, de la propia librería Matplotlib de Python.

Esta librería nos fue de gran ayuda ya que nos permitió por medio del código obtener como resultado el mapa con la ubicación de dicho canal.



Imagen 10. Resultado con el uso de las librerías basemap.

- **Folium:** Librería que se utiliza para visualizar los mapas y el manejo mediante el uso de Python.
- **Geopandas:** GeoPandas es una de las librerías geoespaciales más completas de Python. Podríamos decir que es sencilla a la vez que compleja.

Esta librería nos ayudo a obtener la ubicación geoespacial del canal, ubicado en la Facultad de Ingeniería Civil.



Imagen ilustrativa 11. Resultado con el uso de las librerías Geopandas.

- **GeoPy:** GeoPy facilita a desarrolladores de Python localizar las coordenadas de direcciones, ciudades, países y puntos de referencia en todo el mundo mediante geocodificadores de terceros y otras fuentes de datos.

La librería nos ayudó a localizar la ubicación del canal.

```
[ ] from geopy.geocoders import Nominatin
geolocalizador = Nominatin()
ubicacion = geolocalizador.reverse("19.211186523699826, -103.80443942468253")
print(ubicacion.address)
print(ubicacion.latitude, ubicacion.longitude)
print(ubicacion.raw)

/usr/local/lib/python3.7/dist-packages/geopy/geocoders/osm.py:143: UserWarning: Using N
UserWarning
Campus Coquimatlán - U de C., Calle Agustín George, Coquimatlán, Colima, 28620, México
19.211698849999998 -103.8035841576899
{'place_id': 225894598, 'licence': 'Data © OpenStreetMap contributors, ODbL 1.0. https://
```

Imagen 12. Resultado con el uso de las librerías GeoPy.

El uso de estas librerías fue esencial para el manejo de los datos en Python.

Como recomendación, para la graficación de la sección del canal sugerimos que al utilizar el programa realizado por nosotros se debe de tener antes muy bien ordenada la información de nuestro dataframe (Excel) pues solo se podrá ejecutar de la manera correcta si se cuenta con la información ordenada en columnas y cada una de ellas con los nombres correspondientes (Cadenamientos) y (Cotas).

4. Resultados

Respecto a los resultados, podemos decir que, si se lograron cumplir con los objetivos que nos planteamos para la elaboración de nuestro programa de Python, es decir, se pudo programar y obtener el volumen de dicho canal, se logro graficar las secciones del canal, además de poder geolocalizar el canal por medio de librerías geoespaciales en donde se especifica a detalle la ubicación y las coordenadas de este mismo.

Conforme al procesos del desarrollo de los programas pudimos comprobar que los datos obtenidos eran correctos tanto para el cálculo del volumen del canal y el grafico de las secciones del canal, además el uso de estas librerías nos sirvió de mucha ayuda de acuerdo con el objetivo del proyecto.

Los códigos realizados fueron los siguientes:

```
#Codigo para cálculo de volumen
Canal.

# Datos de entrada
base = float(input("Base(m) : "))
z = float(float (input("Z : ")))
S = float(input("pendiente de fondo S: "))
n = float(input("rugorosidad : "))
y = float(input("tirante normal (m) : "))

# Cálculos
A = base*y+z*y**2
P = base+2*y*(1+z**2)**.5
```

```
R = round(A/P,2)
print('El radio hidráulico es (m) : ', R)

#Velocidad
V = round(1/(n)*R**(2/3)*S**.5,2)
print('La velocidad del agua es de (m/s) :', V)
Q = round(V*A,2)
print('El volumen es de (m3):',Q)
```

Código para graficar las secciones.

```
from google.colab import files
from google.colab import drive
import matplotlib.pyplot as plt
import pandas as pd
import xlrd

drive.mount('/gdrive')
df= pd.read_excel('/gdrive/MyDrive/Colab Notebooks/SECCIONES/SECCION
ES.xlsx',sheet_name= 'Hoja1')

VALORES = df[['Cadenamientos', 'Cotas']]
ax= VALORES.plot(x='Cadenamientos', y= 'Cotas',rot=0)
plt.xlabel('Cadenamientos')
plt.ylabel('Cotas')
plt.show
```

```
# Código para obtener la ubicación
mediante las coordenadas.
from geopy.geocoders import Nominatim
geolocalizador = Nominatim()
ubicacion = geolocalizador.reverse(
("19.211186523699826, -
103.80443942468253"))
print(ubicacion.address)
print(ubicacion.latitude, ubicacion.longitude)
print(ubicacion.raw)
```



```
# Código Geoespacial para
obtener la ubicación.

%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.basemap import
Basemap

fig = plt.figure(figsize=(8, 8))
m = Basemap(projection='ortho', re
solution=None,
            width=8E6, height=8E6,

            lat_0=45, lon_0=-100,)
m.etopo(scale=1.0, alpha=1.0)

# Map (long, lat) to (x, y) for pl
otting
x, y = m(-
103.80450468775071,19.210950611861
64)
plt.plot(x, y, 'ok', markersize=5)
plt.text(x, y, ' Caudal_FIC', font
size=22);

#Codigo para un mapa interactivo
en la ubicacion del lugar
import folium
m = folium.Map(
    location=[19.211249641928656,
-103.80451631674501],
    zoom_start=12,
    tiles='Stamen Terrain'
)
tooltip = 'Canal de estudio'
folium.Marker([19.211249641928656,
-103.80451631674501], popup='Canal
de estudio',
tooltip=tooltip).add_to(m)
```

5. Conclusiones

A modo de cierre de este proyecto es importante mencionar que la intención de realizarlo era poder hacer un programa en Python, el cual nos calculara el volumen del canal, graficara las secciones y pudiera

geolocalizar el lugar donde se ubica dicho canal. El cual logramos cumplir nuestros objetivos puestos para este proyecto.

Al ser un éxito este programa, podemos usarlo como una gran herramienta de suma importancia para nosotros estudiantes, futuros ingenieros, y nos servirá de mucho al realizar los cálculos (volúmenes, secciones, ubicación) de un canal de una manera más rápida y sencilla.

Por ello, no cabe duda alguna que la programación juega un papel muy útil en la vida humana, y en las ingenierías no es la excepción, pues con ella podemos realizar varios procesos u las cuales manualmente no tomaría bastante tiempo realizarlas, por lo que con el uso de programas informáticos podemos implementar y ejecutar de manera más rápida, precisa y eficaz, tal y como se pudo demostrar en este proyecto.

Para finalizar, el realizar este código nos deja una gran enseñanza, pues ejemplifica cómo es que podemos emplear la programación o los lenguajes de programación para resolver cuestiones de la vida diaria de cualquier tipo, principalmente enfocados en el área académica.

Referencias

- Cordova, P. L. A. (2018, 13 marzo). AREAS Y VOLUMENES DE FIGURAS EN PYTHON.
- 6 formas de calcular el volumen. (2019, 13 marzo).
- https://www.fao.org/fishery/docs/CDrom/FAO_Training/FAO_Training/General/x6708s/x6708s08.htm
- Estructuras en Canales. (2003, 31 agosto). oocities.
- El paquete Numpy — Fundamentos de Programación en Python. (s. f.).
- Gonzalez, L. (2022, 23 septiembre). Introducción a la Librería Matplotlib de Python. Aprende IA.

- Estévez, R. (2019, 9 agosto). Creando mapas con Python, Matplotlib y Basemap. geomapik.
- Rédac, T. (2022, 2 agosto). Folium: descubra la biblioteca de Python Open Source. Formation Data Science | DataScientest.com.
- Morales, A. (2021, 25 noviembre). GeoPandas: Análisis de datos geográficos en Python. MappingGIS.
- Morales, A. (2021, 25 noviembre). GeoPandas: Análisis de datos geográficos en Python. MappingGIS

