



LICENCIATURA EN FÍSICA
DEPARTAMENTO DE FÍSICA

FÍSICA COMPUTACIONAL 1

Evaluación 1

Alumna:
Brambilla Zamorano Fátima Fernanda

Fecha:
08/03/18

1 Descarga y Depuración de Datos

En primer lugar descargamos un par de archivos como se nos fue indicado, el primero dada los datos del nivel del mar en el manglar *El Sargento*, y el segundo los de la salinidad en el mismo. Y se nos pido que por medio de los comandos de Linux y Emacs revisáramos que ambos archivos (independientes uno del otro) tuvieran la misma cantidad de líneas, y columnas de datos, así como se nos fue pedido que los editáramos de no ser este el caso. De modo que al archivo de los datos de salinidad le borre las últimas dos líneas de datos para que tuviera la misma cantidad de filas que el archivo de los datos del nivel de agua.

2 Procedimiento de lectura de Datos

Lo siguiente fue leer los archivos con Python, por medio del entorno de trabajo de Jupyter Notebook, donde debíamos cargar las bibliotecas que serían requeridas, como "date time", "pandas", entre otras. Después de importar las bibliotecas, le indicamos al programa que lea los datos de ambos archivos, y muestra las primeras líneas de la tabla para comprobar que los esta leyendo como es debido. A continuación, se hizo una transformación de la columna "Date" en los archivos originales, para poder trabajar más comodamente con ella durante el programa que desarrollaría unas gráficas.

```
In [4]: gh['Ndate'] = pd.to_datetime(gh['Date'], format='%m/%d/%Y %H:%M:%S')
gh['month'] = gh['Ndate'].dt.month
gh.head()
```

Out[4]:

	N	Date	AbsP	Temp	WL	Ndate	month
0	1	02/04/2018 09:45:00	108.068	17.094	0.060	2018-02-04 09:45:00	2
1	2	02/04/2018 10:00:00	107.815	16.903	0.035	2018-02-04 10:00:00	2
2	3	02/04/2018 10:15:00	107.791	16.903	0.032	2018-02-04 10:15:00	2
3	4	02/04/2018 10:30:00	107.791	16.903	0.032	2018-02-04 10:30:00	2
4	5	02/04/2018 10:45:00	107.791	16.903	0.032	2018-02-04 10:45:00	2

```
In [5]: df['Ndate'] = pd.to_datetime(df['Date'], format='%m/%d/%Y %H:%M:%S')
df['month'] = df['Ndate'].dt.month
df.head()
```

Out[5]:

	N	Date	CHigh	Temp	SC	Sal	Ndate	month
0	1	02/04/2018 09:30:00	50782.6	17.53	59295.4	39.6728	2018-02-04 09:45:00	2.0
1	2	02/04/2018 09:45:00	50617.2	17.33	59362.4	39.7235	2018-02-04 10:00:00	2.0
2	3	02/04/2018 10:00:00	50646.3	17.23	59526.8	39.8482	2018-02-04 10:15:00	2.0
3	4	02/04/2018 10:15:00	50772.9	17.17	59753.7	40.0203	2018-02-04 10:30:00	2.0
4	5	02/04/2018 10:30:00	51007.8	17.14	60068.6	40.2595	2018-02-04 10:45:00	2.0

Una vez editada la columna *Date*, comenzamos a hacer las gráficas correspondientes. En primer lugar, hicimos tres gráficos de caja que mostraban los siguientes datos:

1. El nivel del mar
2. La salinidad en partes por millón
3. La temperatura del agua

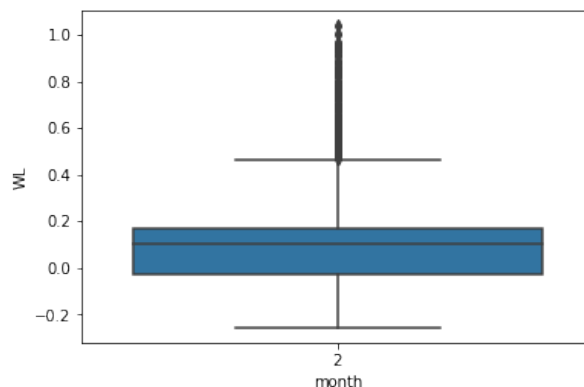
Después de hacer las gráficas de caja, utilizamos las función *describe* para calcular los maximos y minimos de los cuartiles. A continuación, se hicieron gráficas

independientes para los mismos datos mencionados anteriormente, esta vez en función del tiempo, de modo que se pudiera ver mejor su comportamiento. En estas últimas gráficas pudimos notar como es que estos datos tienen un comportamiento oscilatorio, mostrando varios picos en las gráficas. Después de analizar el comportamiento de estas tres variantes, se nos fue pedido que hicieramos gráficas dobles, ver el comportamiento de estas variables en pareja y para ello fue necesario unir los datos de ambos archivos, usando de pormedio una de las funciones de las bibliotecas pandas, concat, para unir las columnas de datos deseados. De este modo fue como si trabajáramos con un tercer archivo de datos, el cual combinaba únicamente las partes deseadas de los archivos leídos previamente. Una vez que hicimos este arreglo, fue posible hacer las últimas gráficas requeridas, las cuales tuvieron dos versiones, una con todos los datos del archivo, y la segunda limitando los datos a cinco días de medición, para poder observar mejor el comportamiento que tenían el nivel del agua, la salinidad en esta y su temperatura.

3 Gráficas obtenidas

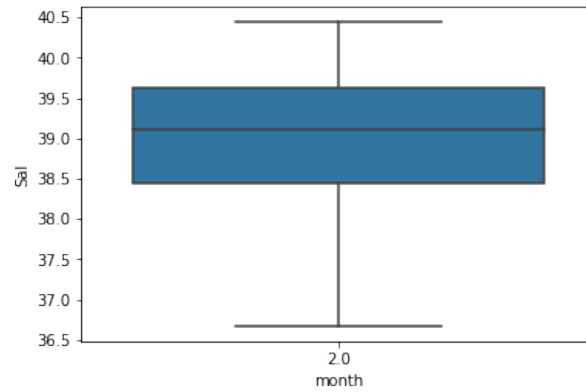
En esta sección se presentaran las gráficas obtenidas a lo largo de la actividad que consistió en la evaluación parcial, así como una breve explicación de cada una de ellas.

La primera gráfica corresponde al gráfico de caja de los datos del nivel del agua durante el mes de febrero. A simple vista no es fácil decir cual es la dispersión

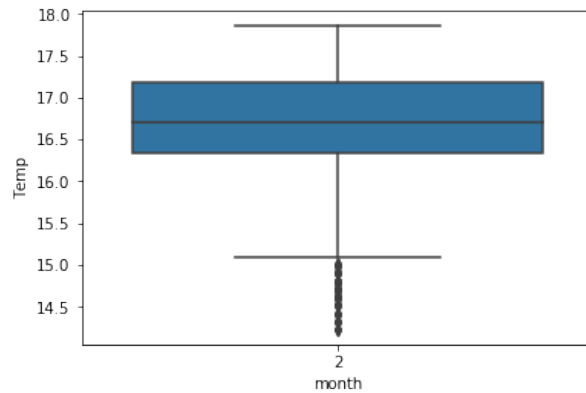


de datos, sin embargo, es sencillo ver donde se encuentran los cuartiles así como la media de los datos. Para esto, mostrare más adelante una tabla con estos datos mencionados.

El siguiente gráfico corresponde al diagrama de caja para los datos de la salinidad del agua durante el mes:



Al contrario que el caso anterior, la dispersión de datos en la salinidad del agua parece estar menos dispersa, lo cual se puede notar por los puntos que aparecen en la recta en la primera gráfica, fuera de la caja, y que en la segunda gráfica no aparecen. Por último, presente el gráfico de caja para los datos de la temperatura. A pesar de que se hizo un gráfico para la temperatura marcada en cada uno de los archivos de datos, ya que estos datos son bastante similares, solo se expondrá una de las gráficas Como en el caso del gráfico del nivel del



agua, se puede observar que hay una dispersión de datos algo grande, ya que algunos de estos salen de la caja y se ven en forma de puntos.

En la siguiente imagen se pueden observar los datos que componen las tres gráficas anteriores, siendo la primera tabla correspondiente al archivo de los

```
In [10]: gh.describe()
Out[10]:
```

	N	AbsP	Temp	WL	month
count	2216.00000	2216.000000	2216.000000	2216.000000	2216.0
mean	1108.50000	108.472155	16.676223	0.100208	2.0
std	639.84842	2.170574	0.756794	0.215948	0.0
min	1.00000	104.878000	14.230000	-0.257000	2.0
25%	554.75000	107.199750	16.332000	-0.026000	2.0
50%	1108.50000	108.488000	16.713000	0.102000	2.0
75%	1662.25000	109.178000	17.189000	0.170000	2.0
max	2216.00000	117.899000	17.855000	1.038000	2.0

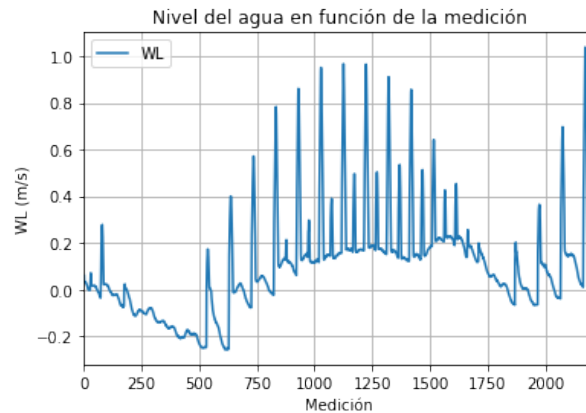
```

In [11]: df.describe()
Out[11]:
```

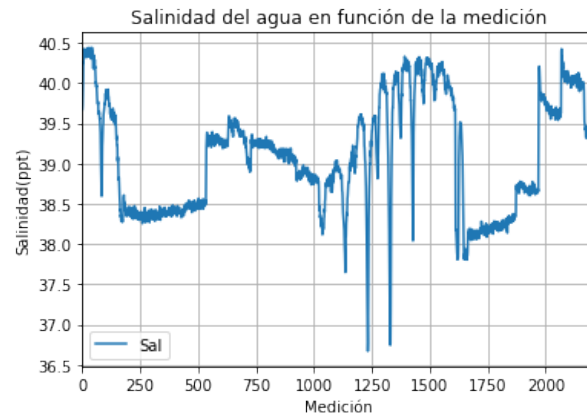
	N	CHigh	Temp	SC	Sal	month
count	2218.00000	2218.000000	2218.000000	2218.000000	2218.000000	2216.0
mean	1109.50000	49435.553336	16.902182	58526.761677	39.092089	2.0
std	640.42577	1147.713269	0.670626	922.110180	0.697061	0.0
min	1.00000	46569.900000	14.100000	55305.400000	36.669100	2.0
25%	555.25000	48410.500000	16.530000	57655.400000	38.433300	2.0
50%	1109.50000	49294.000000	16.850000	58559.300000	39.115650	2.0
75%	1663.75000	50068.200000	17.390000	59244.200000	39.634000	2.0
max	2218.00000	52095.300000	18.100000	60306.400000	40.440200	2.0

datos del nivel del agua, y la segunda de los datos de la salinidad de la misma.

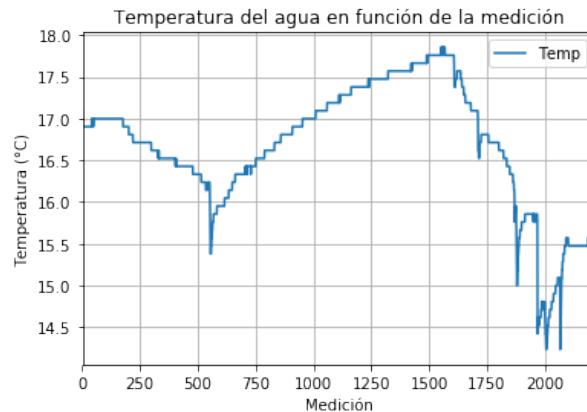
Las siguientes gráficas son las gráficas independientes para el nivel del agua, la salinidad en el agua y la temperatura del agua, teniendo las tres variables en función del tiempo. En primer lugar mostraré la del nivel del agua.



En esta se puede observar que el comportamiento del nivel del agua es oscilatorio, teniendo altos y bajos a medida que la medición cambia. La siguiente gráfica es de la salinidad en el agua en función de la medición de datos:



Esta gráfica tiene un comportamiento más extraño si se compará con la gráfica del nivel del agua, no obstante, podría decirse que por tiempos tiene un comportamiento oscilatorio también, solo que sus minimos son más grandes en magnitud que los minimos de los datos en gráfica del nivel del mar. Por último, la gráfica de la temperatura del agua en función de las mediciones hechas



Esta gráfica tiene un comportamiento bastante diferente a las dos anteriores, por ún lado pareciera que es más estable en cuanto a sus cambios, cosa que hace que sea más sencillo apreciar los cambios en la temperatura a lo largo de cada medición realizada.

Para continuar la actividad, fue necesario hacer una combinación de los *DataFrames* que teníamos por cada archivo, de manera que tuvieramos un tercer *DataFrame* que nos combinará únicamente las columnas deseadas, para lo cual se útilizo el siguiente código:

```

In [15]: gh1 = pd.DataFrame(gh)
         df1 = pd.DataFrame(df)

In [16]: u = pd.concat([df1['Date'], df1['Sal'], gh1['WL'], gh1['Temp'], gh1['W']], axis=1, keys=['Date', 'Sal', 'W
         L', 'Temp', 'W'])

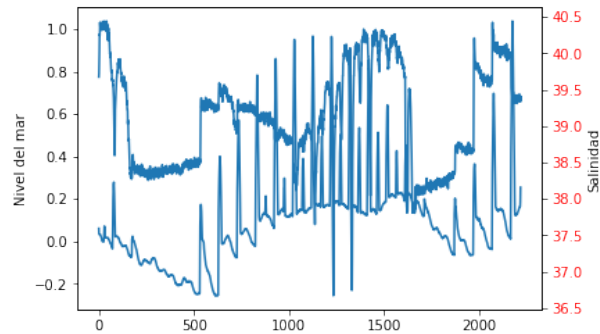
In [17]: u.head()
Out[17]:

```

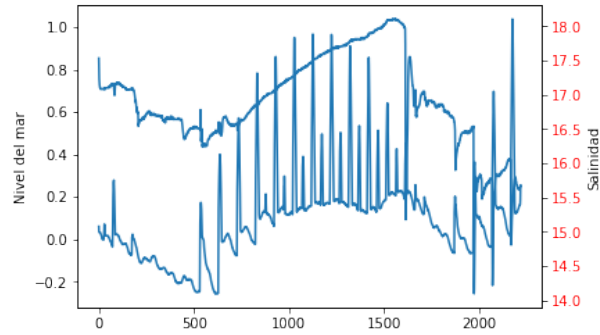
	Date	Sal	WL	Temp	N
0	02/04/2018 09:30:00	39.6728	0.060	17.094	1.0
1	02/04/2018 09:45:00	39.7235	0.035	16.903	2.0
2	02/04/2018 10:00:00	39.6462	0.032	16.903	3.0
3	02/04/2018 10:15:00	40.0203	0.032	16.903	4.0
4	02/04/2018 10:30:00	40.2595	0.032	16.903	5.0

Como se puede apreciar en la imagen, se utilizarón únicamente las columnas que eran compartidas por ambos archivos, como lo era la columna N, la cual era el número de la medición realizada, y la columna Date, que contenía día y hora de la medición, además de unir las dos columnas deseadas, WL que corresponde al nivel del agua, Temp, que corresponde a la temperatura del agua y Sal que corresponde a la salinidad. Se hizo esto para poder hacer dos nuevas gráficas, que nos mostrarían en una sola imagen:

- Nivel del agua y Salinidad



- Nivel del agua y Temperatura



Estando ambos casos en función del número de mediciones realizado. Estas gráficas, por falta de tiempo tienen ambas variables del mismo color, por lo que no es sencillo ver cual línea corresponde a cual variable, más adelante tanto este comentario como las gráficas serán editadas.

Las últimas gráficas son de nueva cuenta las gráficas independientes del nivel del agua, la salinidad en el agua y la temperatura del agua como funciones del tiempo, esta vez siendo limitadas por las mediciones hechas en cinco días, lo que se cumple aproximadamente en la medición 285. Sin embargo, debido a un error de lectura, no son estas gráficas las que hice, sino que hice dos pares de gráficas que no iban en la actividad, las cuales a pesar que no las mostraré en este documento, pueden verse en *GitHub*