

# Evaluación número 1

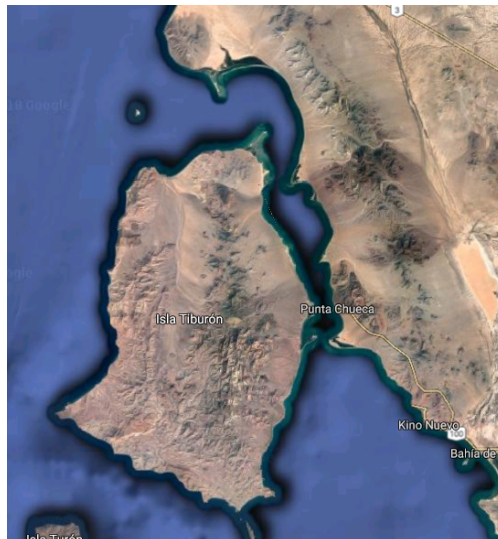
Barajas Ibarria Juan Pedro

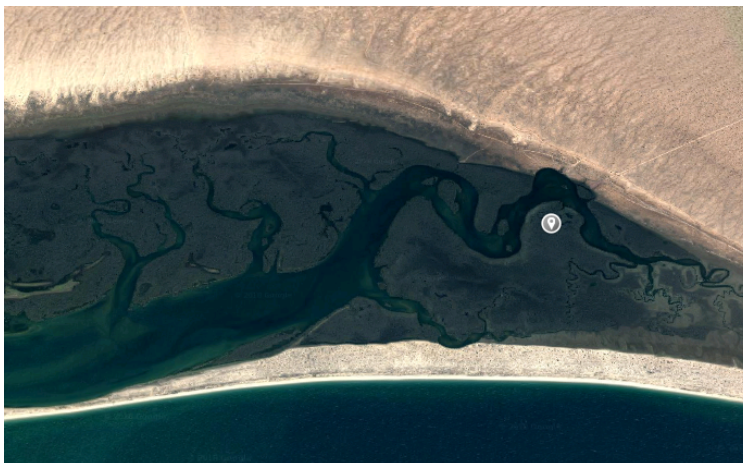
A 08 de Marzo del 2018



## Introducción

Se tiene una estación de monitoreo de variables atmosféricas, CO<sub>2</sub>, radiación solar, nivel de agua y salinidad en el Manglar El Sargento, en una bahía en la costa frente a la parte norte de la Isla Tiburón.





## Resumen

### La isla tiburón

La isla Tiburón [2] o isla del Tiburón. es una isla mexicana, la más grande del golfo de California y tiene una extensión de 1208 km cuadrados. Administrativamente, pertenece al estado de Sonora, específicamente al municipio de Hermosillo, y se ubica aproximadamente a la misma latitud que Hermosillo. Está separada del continente por un estrecho canal de sólo tres kilómetros de ancho llamado estrecho del Infiernillo.

La isla está deshabitada, a excepción de una instalación militar ubicada en la zona oriental de la isla. Está administrada como una reserva ecológica por el gobierno de los seris, en conjunto con el gobierno Federal. En siglos anteriores, la isla fue habitada por tres grupos ("bandas") de los seris: los Tahejöc comcaac, los heeno comcaac y los xiica Hast ano coii (en una parte). Esta isla es para los seris un sitio sagrado, ya que consideran la isla como la cuna de su pueblo.

### Manglar

El manglar [1] es un área biótica o bioma, formado por árboles muy tolerantes a las sales existentes en la zona intermareal cercana a la desembocadura de cursos de agua dulce en latitudes tropicales y subtropicales. Así, entre las áreas con manglares se incluyen estuarios y zonas costeras. Tienen una gran diversidad biológica con alta productividad, encontrándose muchas especies de aves como de peces, crustáceos, moluscos y otras.

Su nombre deriva de los árboles que los forman, los mangles, el vocablo mangle (de donde se deriva mangrove en alemán, francés e inglés) proviene de una voz caribe o arahuaca,1 quizá guaraní[cita requerida] y significa árbol retorcido. Normalmente se dan como barrera motivos de desarrollo, la costa ha sufrido una rápida erosión. También sirven de hábitat para numerosas especies y proporcionan una protección natural contra fuertes vientos, olas producidas por huracanes e incluso por maremotos.

## Desarrollo de la actividad

- La estructura de datos esta dada con una por una separación por columna en el caso de los datos de salinidad tenemos 6 columnas.
1. Al iniciar con la actividad se descargan dos archivos de datos de los meses de octubre y noviembre del 2017 de la estación *El Sargento* los cuales los cuales fueron tomados en intervalos de 15 minutos los cuales analizaremos.

2. Descargando los datos pasamos a ejecutar jupyter notebook donde en Pandas cargamos las librerías Numpy, Pyplot de Matplotlib y la biblioteca datetime, continuando cargamos los dos archivos de datos los cuales son:

- Sargento17.csv [3], cargado en la variable dfsar
- Sargento17sanidad [4], cargad en la variable dfsal

Donde agregamos los nombres de las columnas igual que en el archivo original y enseguida pasamos a darle la estructura de datos bajo los nombres que aparecen los siguientes segmentos de código.

```
In [1]: import pandas as pd
import numpy as np
from datetime import datetime
import matplotlib.pyplot as plt
```

```
In [2]: # Lee un archivo de texto con la función Pandas "read_csv", con elementos separados
# un espacio, brincándose 4 renglones del inicio (encabezados)
dfsar = pd.read_csv('sargento17.csv', skiprows=2, sep=',', names=['#', 'Datetime',
', 'Abspres', 'Temp', 'Waterlevel'])
dfsal = pd.read_csv('sargento17sanidad.csv', skiprows=2, sep=',', names=['#', 'Dateti
', 'CondhR', 'Temp', 'SpecCon', 'Sal'])
# "Shift + Enter"
```

```
In [4]: dfsarg=pd.DataFrame(dfsar)
dfsani=pd.DataFrame(dfsal)
```

3. Después de darle la estructura de datos pasamos a convertir la columna de datos datetime a datos temporales como se muestra en el segmento de código, estos acompañados con el comando `head()` el cual los muestra de tal manera que cada columna de datos tiene nombre.

```
In [5]: # Convertir la cadena de caracteres 'Date' en variable temporal 'NDate'
dfsani['Ndate'] = pd.to_datetime(dfsani['Datetime'], format='%m/%d/%Y %H:%M:%S')
dfsani['month'] = dfsani['Ndate'].dt.month
dfsani.head()
```

```
Out[5]:
```

	#		Datetime	CondhR	Temp	SpecCon	Sal	\
0	1	10/26/2017	12:45:00	54525.5	25.21	54301.2	35.9195	
1	2	10/26/2017	13:00:00	54525.5	24.91	54622.1	36.1588	
2	3	10/26/2017	13:15:00	54525.5	24.82	54719.0	36.2311	
3	4	10/26/2017	13:30:00	54525.5	24.76	54783.8	36.2794	
4	5	10/26/2017	13:45:00	54525.5	24.75	54794.6	36.2875	

		Ndate	month
0	2017-10-26	12:45:00	10
1	2017-10-26	13:00:00	10
2	2017-10-26	13:15:00	10
3	2017-10-26	13:30:00	10
4	2017-10-26	13:45:00	10

```
In [6]: # Convertir la cadena de caracteres 'Date' en variable temporal 'NDate'
dfsarg['Ndate'] = pd.to_datetime(dfsarg['Datetime'], format='%m/%d/%Y %H:%M:%S')
dfsarg['month'] = dfsarg['Ndate'].dt.month
dfsarg.head()
```

```
Out[6]:
```

	#		Datetime	Abspres	Temp	Waterlevel	Ndate	\
0	1	10/26/2017	13:00:00	105.612	24.448	-0.150	2017-10-26 13:00:00	
1	2	10/26/2017	13:15:00	105.513	24.351	-0.160	2017-10-26 13:15:00	
2	3	10/26/2017	13:30:00	105.433	24.351	-0.168	2017-10-26 13:30:00	
3	4	10/26/2017	13:45:00	105.385	24.351	-0.173	2017-10-26 13:45:00	
4	5	10/26/2017	14:00:00	105.321	24.351	-0.179	2017-10-26 14:00:00	

	month
0	10
1	10
2	10
3	10
4	10

4. Al terminar de convertir todos los datos necesarios continuamos con las gráficas, donde con ayuda de seaborn [?] se realizaron gráficas de caja para:

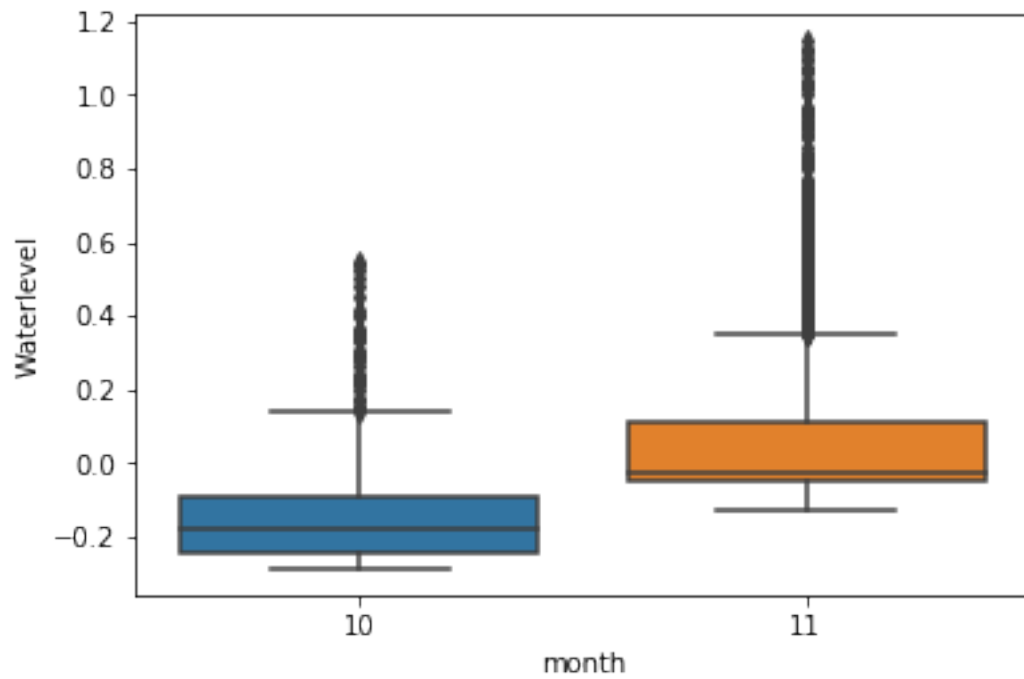
- a) a) Nivel del mar (Metros)
- b) b) Salinidad (PPM)
- c) c) Temperatura de agua (grados centigrados )

Respectivamente como se muestran a continuación con el segmento de código utilizado seguido de la imagen de la gráfica de caja.

También después de ellas presentamos con el comando `.describe()` sus datos estadísticos principales para poder interpretar los diagramas de caja.

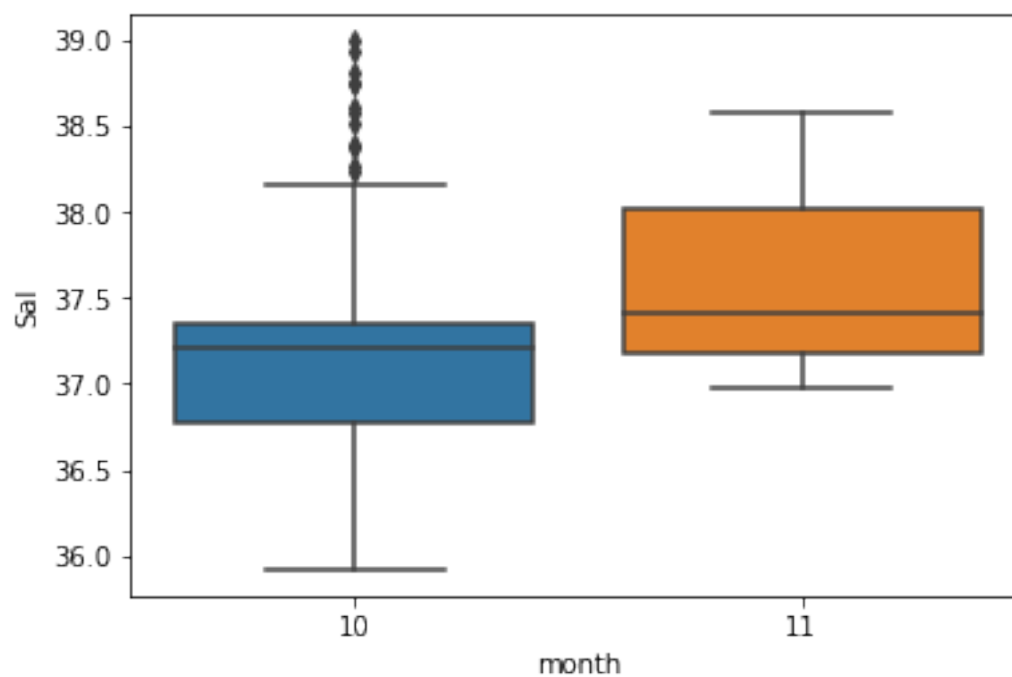
a)

```
In [7]: # graficar Boxplots por mes
# Biblioteca Seaborn
import seaborn as sns
import matplotlib.pyplot as plt
ax = sns.boxplot(x="month", y="Waterlevel", data=dfsarg)
plt.show()
```



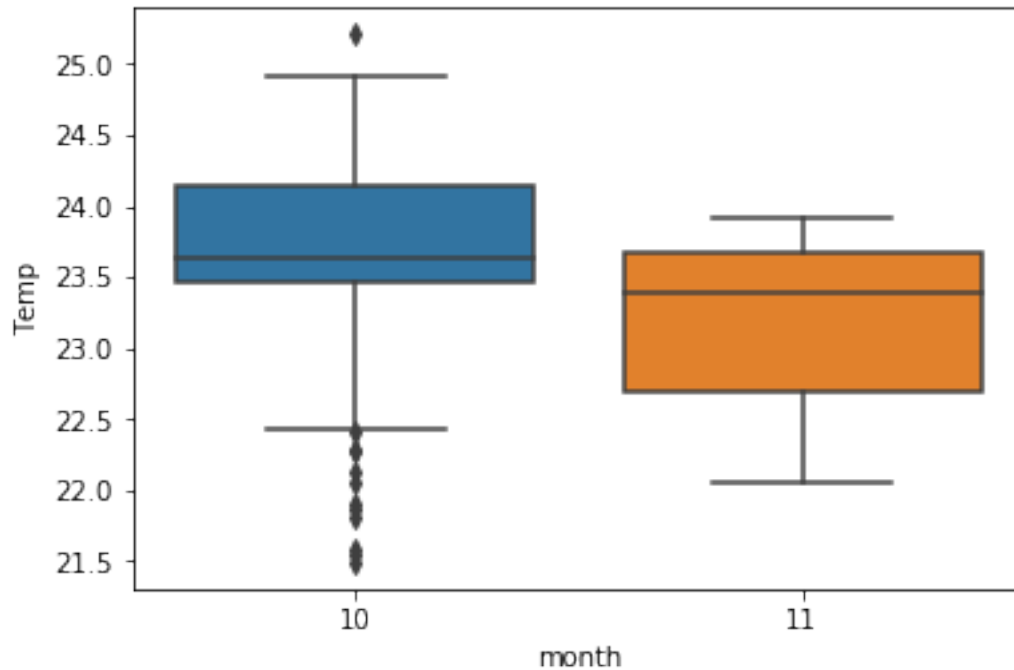
b)

```
In [9]: # graficar Boxplots por mes
# Biblioteca Seaborn
import seaborn as sns
import matplotlib.pyplot as plt
ax = sns.boxplot(x="month", y="Sal", data=dfsani)
plt.show()
```



c)

```
In [11]: # graficar Boxplots por mes
# Biblioteca Seaborn
import seaborn as sns
import matplotlib.pyplot as plt
ax = sns.boxplot(x="month", y="Temp", data=dfsani)
plt.show()
```



```
In [12]: dfsarg.describe()
```

```
Out[12]:
```

	#	Abspres	Temp	Waterlevel	month
count	2395.00000	2395.000000	2395.000000	2395.000000	2395.000000
mean	1198.00000	107.429820	23.120315	0.030845	10.781211
std	691.52127	2.371366	0.564123	0.235926	0.413512
min	1.00000	104.229000	21.760000	-0.288000	10.000000
25%	599.50000	106.407000	22.525000	-0.071000	11.000000
50%	1198.00000	106.764000	23.388000	-0.035000	11.000000
75%	1796.50000	107.303000	23.484000	0.018500	11.000000
max	2395.00000	118.641000	24.448000	1.146000	11.000000

```
In [13]: dfsani.describe()
```

```
Out[13]:
```

	#	Condhr	Temp	SpecCon	Sal \
count	2395.00000	2395.000000	2395.000000	2395.000000	2395.000000
mean	1198.00000	54524.973027	23.317436	56385.960835	37.479086
std	691.52127	11.874193	0.548286	620.837037	0.465969
min	1.00000	54105.700000	21.490000	54301.200000	35.919500
25%	599.50000	54525.500000	22.730000	55949.700000	37.151400
50%	1198.00000	54525.500000	23.490000	56185.600000	37.328300
75%	1796.50000	54525.500000	23.700000	57053.700000	37.980300
max	2395.00000	54525.500000	25.210000	58398.700000	38.994200

	month
count	2395.000000
mean	10.780793
std	0.413795
min	10.000000
25%	11.000000
50%	11.000000
75%	11.000000
max	11.000000

Podemos apreciar en las gráficas que, en a:

- Los niveles de agua crecieron muy poco en noviembre, se puede notar que los valores máximo y mínimo son 1.146m y 0.288 m respectivamente.
- En octubre la media se mantuvo muy centrada en el primer y segundo cuarti y en noviembre de igual forma se concentran en los primeros cuartiles pero la media es menor aquí.

En b:

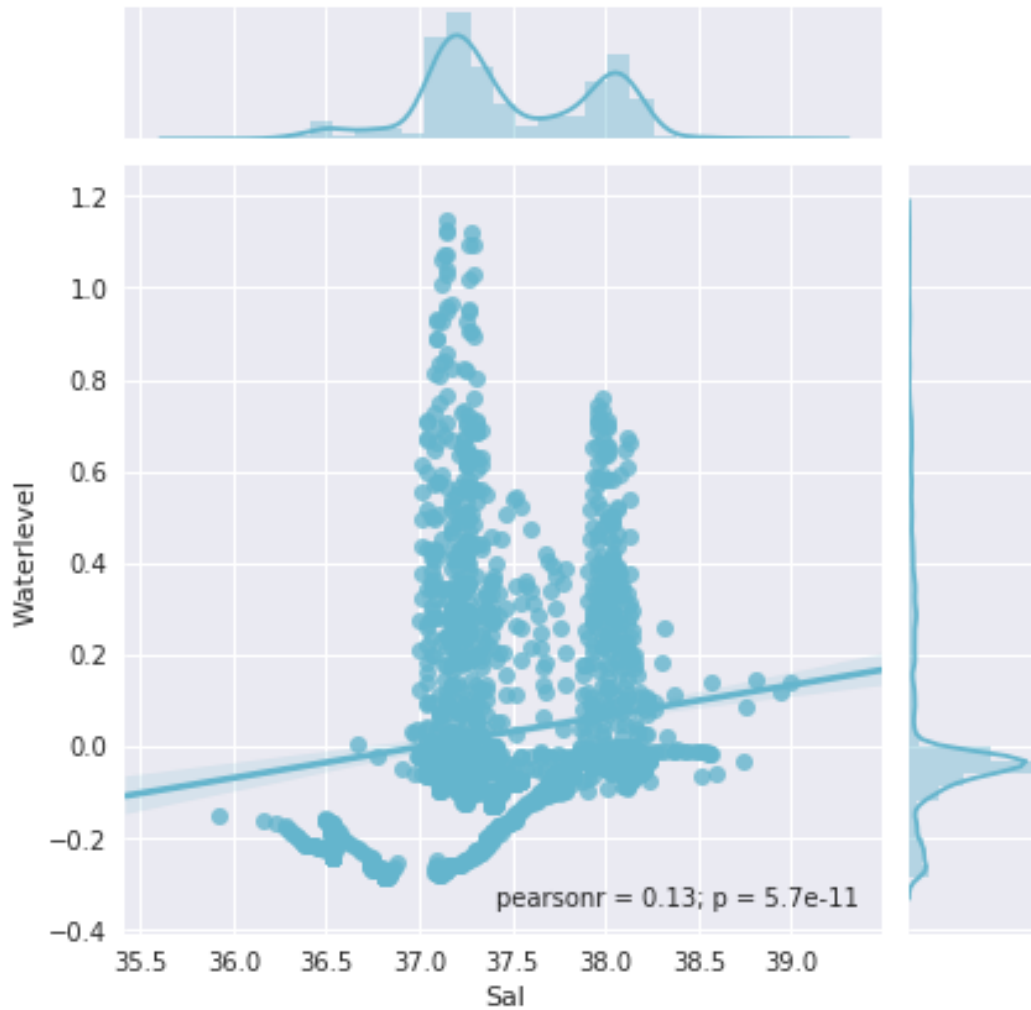
- La salinidad media de estos dos meses es 37.4790 ppm donde la distribución encuentra la media alejada del centro de la concentración de la caja, lo cual nos dice que no vario mucho fuera de ese foco.
- También vemos que la máxima y mínima son 38.994ppm y 35.919 ppm respectivamente.

En c:

- La La temperatura media fue 23.12.. grados centigrados donde la distribución de los dos cuartiles centrales están por arriba de 23 grados y en noviembre tiene una disminución con la distribución mas amplia.
  - Las temperaturas máxima y mínima son 24.44 grados centigrados y 21.76 grados centigrados respectivamente.
5. Después de sacar los datos estadísticos principales pasamos a generar gráficas de de regresión lineal con distribuciones marginales, donde cada gráfica contiene el nombre de sus ejes para interpretar su relación lineal.

```
In [27]: # Gráfica de Nivel de mar-Salinidad
sns.set(style="darkgrid", color_codes=True)
dfjuntos=pd.concat([dfsarg, dfsani], axis=1, join_axes=[dfsani.index])
g = sns.jointplot("Sal", "Waterlevel", data=dfjuntos,kind="reg", color="c")
plt.show(g)
```

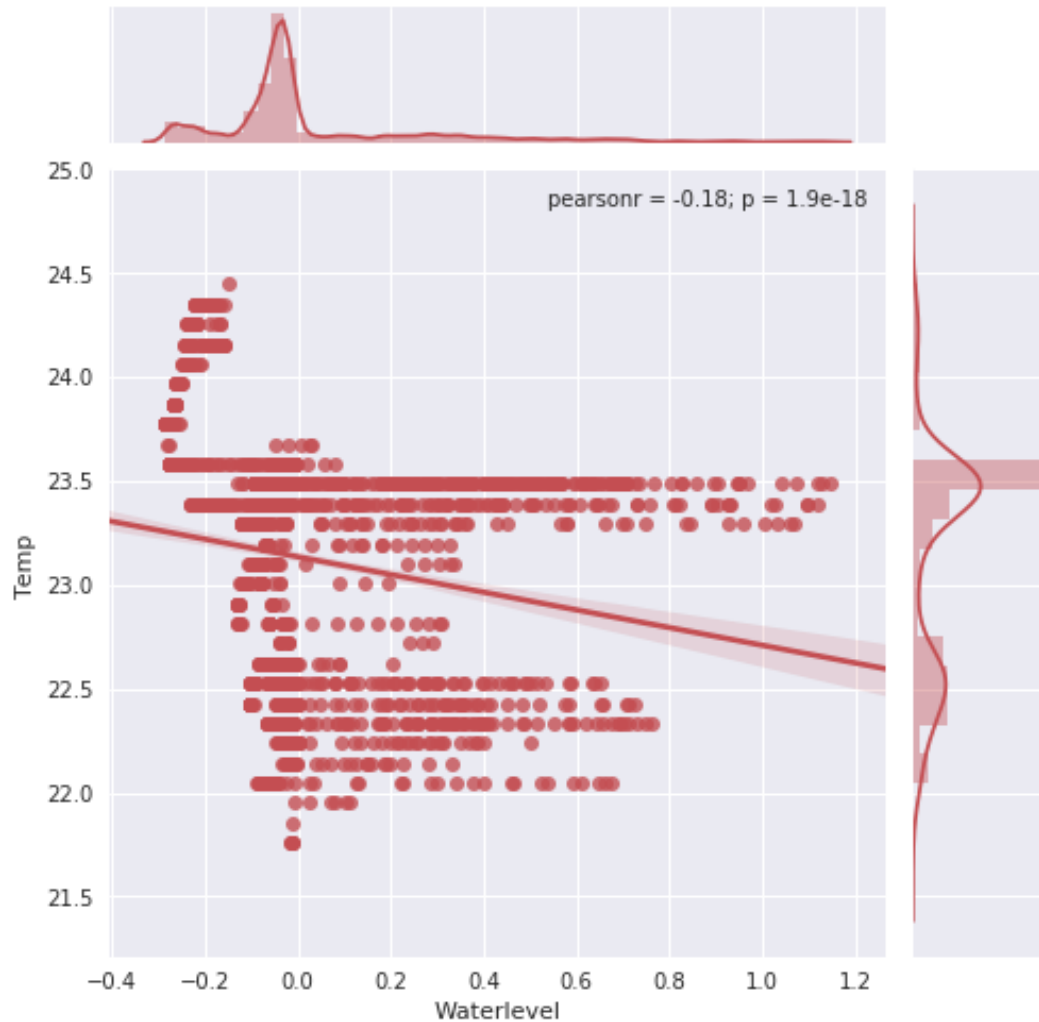




En la primer gráfica de nivel del agua contra salinidad vemos que hay una relación lineal mínima o casi nula donde  $\text{pearsonr}=0.13$  lo cual nos confirma que los datos tienen muy poca relación lineal.

```
In [31]: #Nivel del agua-temperatura
import seaborn as sns
sns.set(style="darkgrid", color_codes=True)

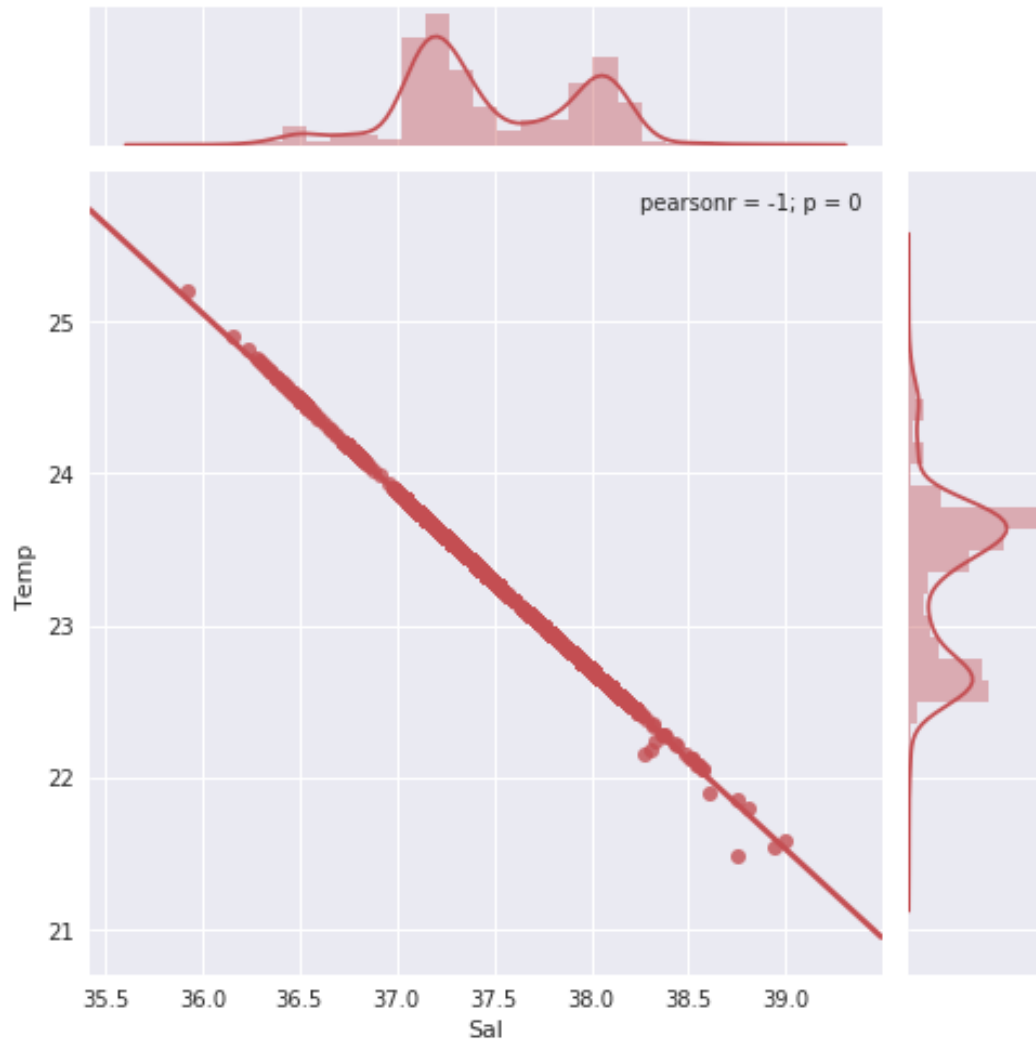
g = sns.jointplot("Waterlevel", "Temp", data=dfsarg, kind="reg",
                  color="r", size=7)
plt.show(g)
```



En la segunda gráfica de temperatura contra nivel del agua vemos que la relación lineal de nuevo tiene muy poca relación, esto nos dice que los datos no tienen mucho que ver con los otros, para confirmar esto vemos que  $\text{pearsonr} = -0.18$ .

```
In [32]: #Salinidad-Temperatura
import seaborn as sns
sns.set(style="darkgrid", color_codes=True)

g = sns.jointplot("Sal", "Temp", data=dfsani, kind="reg",
                  color="r", size=7)
plt.show(g)
```



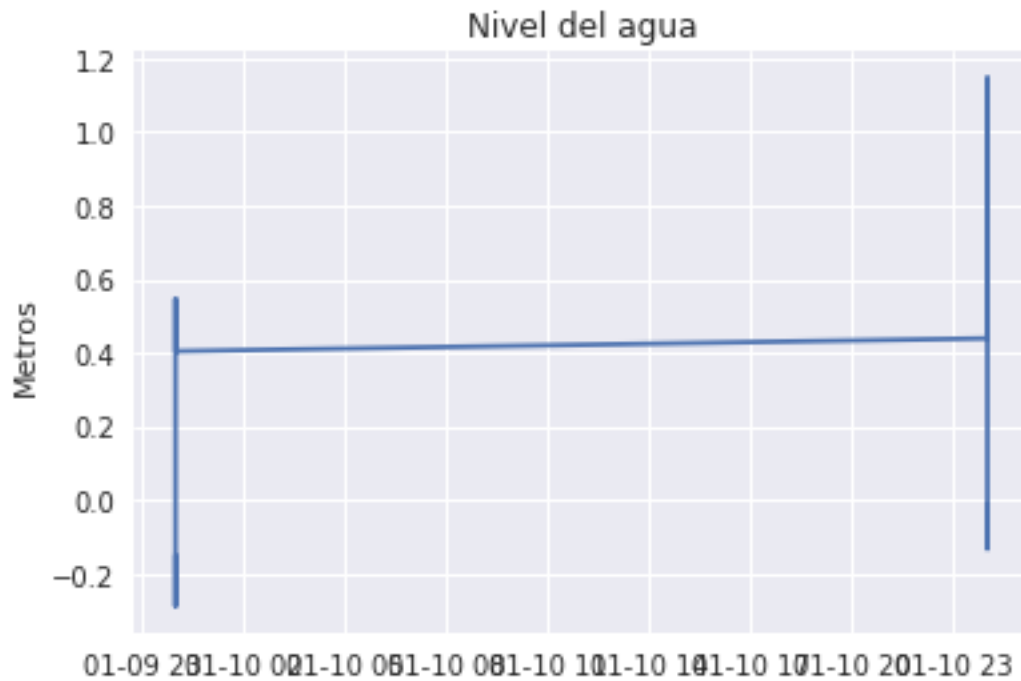
En la tercer gráfica de temperatura contra salinidad del agua vemos una recta muy bien ajustada a los datos, lo cual intuitivamente nos muestra la relación directa de estos dos datos, que nos dice que mientras la temperatura descienda la concentración de sal será mayor, esto lo confirma el dato  $\text{pearsonr} = -1$ , que efectivamente como se predice, la gráfica muestra una asociación lineal negativa perfecta.

6. Continuando con el análisis realizamos tres gráficas independientes de las variables: d) Nivel del mar, e) Salinidad y f) Temperatura del Agua, para ver su variabilidad como función del tiempo, las cuales se encuentran en orden respectivo junto con su segmento de código con el que se generaron.

d)

```
In [42]: plt.plot_date(x=dfsani.month, y=dfsarg.Waterlevel, fmt="b-")
         plt.title("Nivel del agua")
         plt.ylabel("Metros")
```

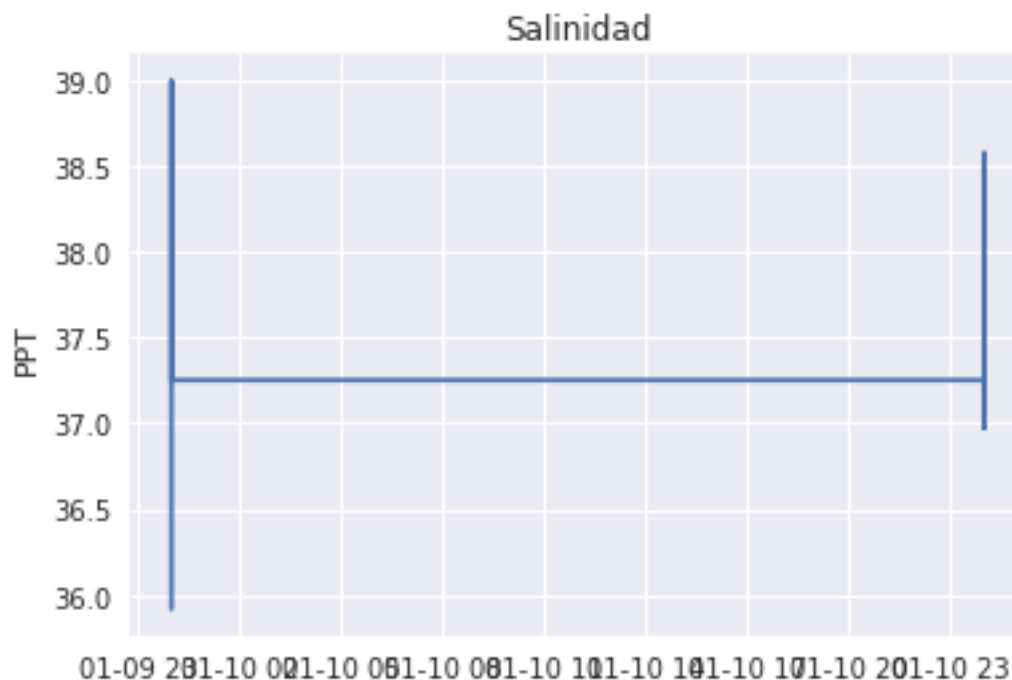
```
plt.grid(True)
plt.show()
```



En esta gráfica vemos, como se dijo en las gráficas de caja pero de manera mas clara, que en noviembre encontramos un ligero aumento en el nivel del agua conforme avanzaba el tiempo habiendo unos valores un poco extremos pero que igual se encuentra en su mayoría al rededor de la media con una desviación estándar de 0.2358.

e)

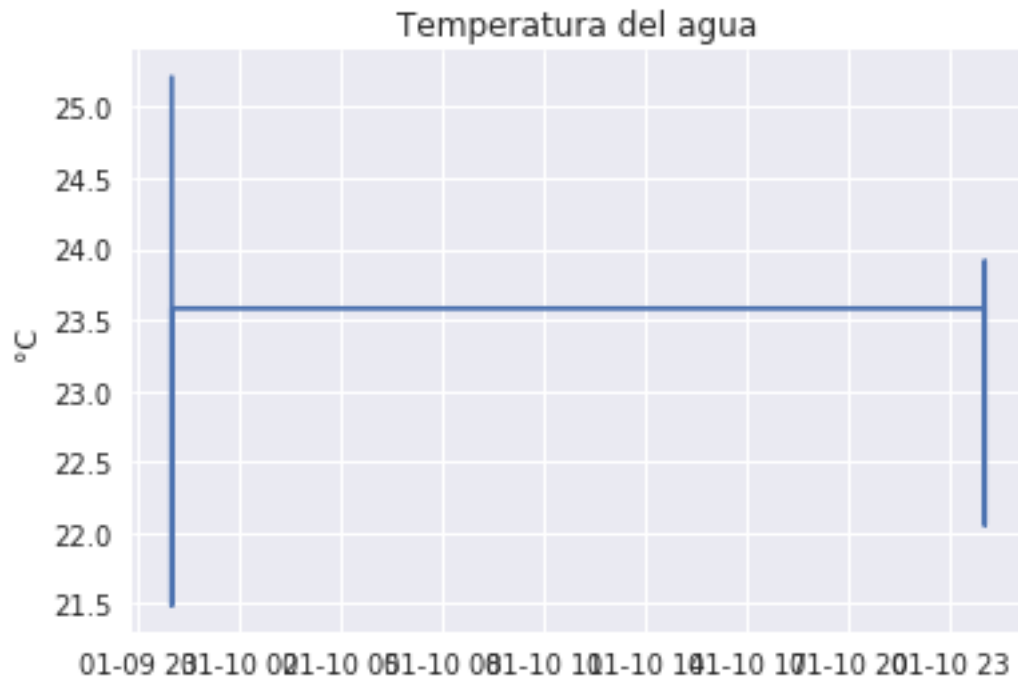
```
In [44]: plt.plot_date(x=dfsani.month, y=dfsani.Sal, fmt="b-")
plt.title("Salinidad")
plt.ylabel("PPT")
plt.grid(True)
plt.show()
```



En esta gráfica encontramos también la relación directa con los diagramas e caja donde su comportamiento se muy constante salvo unos valores extremos al principio y al final, aquí con las descripciones anteriores tenemos que la desviación estándar es 0.4659 lo cual fue causada por esos valores extremos mencionados.

f)

```
In [47]: plt.plot_date(x=dfsani.month, y=dfsani.Temp, fmt="b-")
         plt.title("Temperatura del agua")
         plt.ylabel("°C")
         plt.grid(True)
         plt.show()
```



En la gráfica final de este tipo que es de temperatura vemos que también hubo cierto valor constante con máximos y mínimos muy extremos los cuales desestabilizan los datos estadísticos, de aquí se sabe que la desviación estándar es 0.5482, la cual es muy grande.

7. Por ultimo producimos gráficas superpuestas con doble eje vertical (izquierda, derecha): g) Nivel de mar y Salinidad; h) Nivel de mar y Temperatura.

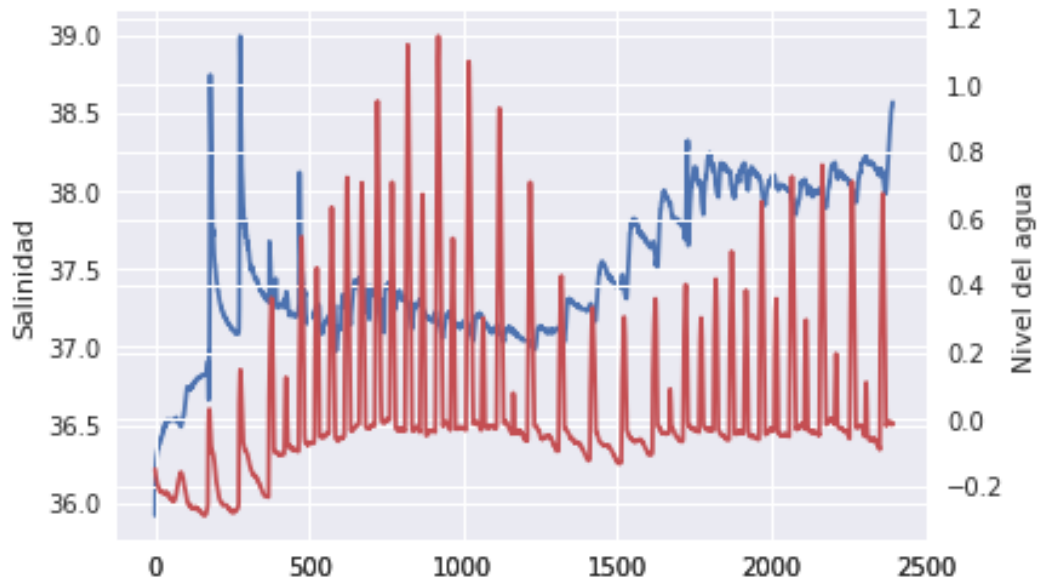
g)

```
In [50]: from pylab import figure, show, legend, ylabel
fig1 = figure()

ax1 = fig1.add_subplot(111)
line1 = ax1.plot(dfsani['Sal'], 'b-')
ylabel("Salinidad")

ax2 = fig1.add_subplot(111, sharex=ax1, frameon=False)
line2 = ax2.plot(dfsarg['Waterlevel'], 'xr-')
ax2.yaxis.tick_right()
ax2.yaxis.set_label_position("right")
ylabel("Nivel del agua")

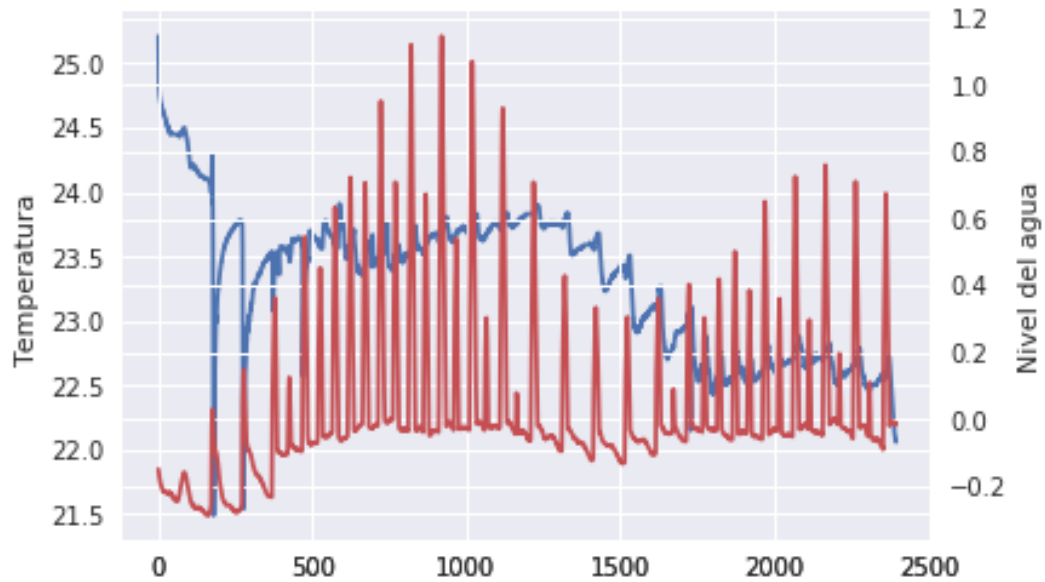
show()
```



h)

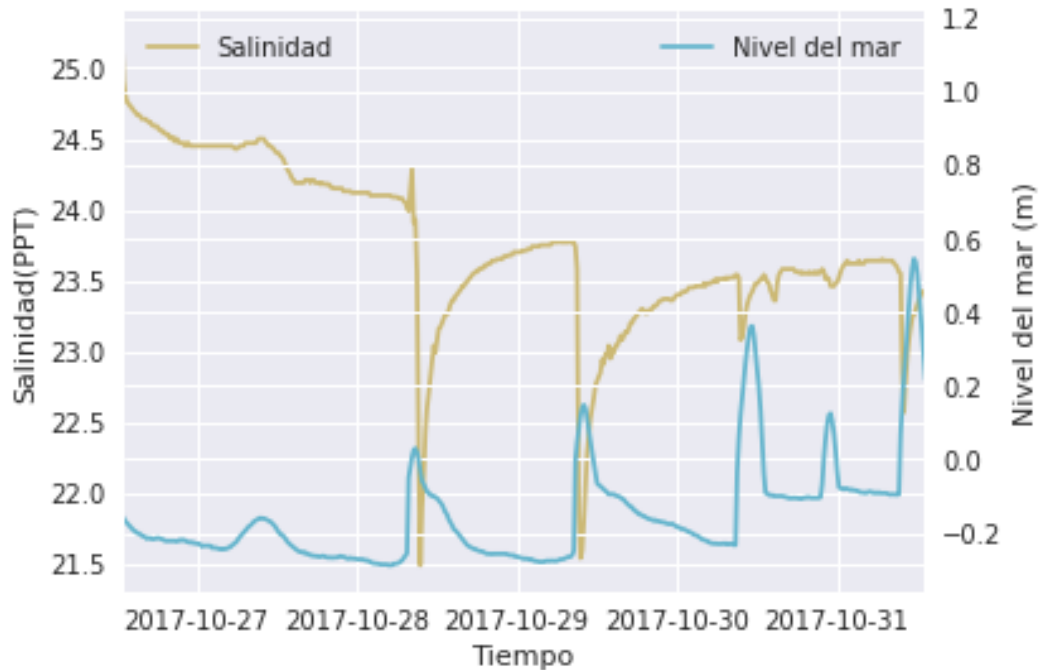
```
In [51]: from pylab import figure, show, legend, ylabel
fig1 = figure()

ax1 = fig1.add_subplot(111)
line1 = ax1.plot(dfsani['Temp'], 'b-')
ylabel("Temperatura")
ax2 = fig1.add_subplot(111, sharex=ax1, frameon=False)
line2 = ax2.plot(dfsarg['Waterlevel'], 'xr-')
ax2.yaxis.tick_right()
ax2.yaxis.set_label_position("right")
ylabel("Nivel del agua")
xlim=
show()
```



```
In [77]: fig, ax1 = plt.subplots()
         date=dfsani['Ndate']
         temp=dfsani.Temp
         WL=dfsarg.Waterlevel
         ax1.plot(date,temp,'y-', label='Salinidad'); plt.legend(loc='upper left')
         ax1.set_xlabel('Tiempo')
         ax1.set_ylabel('Salinidad(PPT)')
         ax2 = ax1.twinx()
         ax2.plot(date, WL , 'c-', label='Nivel del mar'); plt.legend(loc='upper right')
         ax2.set_ylabel('Nivel del mar (m)')
         fig.tight_layout()
         plt.xlim(("10/26/2017 13:00:00"," 10/31/2017 13:00:00"))
         plt.show()
```





Donde estas gráficas enfocándonos bien en un lapso de 5 días del 27 al 31 de octubre se ve claro el comportamiento de estos y como su relación en los dos casos si es notoria, en primera, la salinidad del agua bajara si el nivel del agua aumenta, esto por la concentración de la sal, ya que la sal se distribuirá en un mayor volumen de agua.

También podemos ver que la temperatura de igual manera descenderá si el volumen aumenta ya que mientras mas volumen de agua haya, mas energía se necesitara para calentarla, entonces como los cambios de temperatura no son lineales se da lugar a este fenómeno en la temperatura.

## Referencias

- [1] Anon. Manglar. <https://es.wikipedia.org/wiki/Manglar>, Noviembre 2017.
- [2] Anon. Isla tiburón. [https://es.wikipedia.org/wiki/Isla\\_Tibur%C3%B3n](https://es.wikipedia.org/wiki/Isla_Tibur%C3%B3n), 2018.
- [3] Carlos Lizárraga et al. Datos sargento. [http://fisicacomputacional.pbworks.com/w/file/124398099/sargento\\_201117.csv](http://fisicacomputacional.pbworks.com/w/file/124398099/sargento_201117.csv), Octubre-Noviembre 2017.
- [4] Carlos Lizárraga et al. Datos sargento-salinidad. <http://fisicacomputacional.pbworks.com/w/file/124398095/sargento-salinidad-201117.csv>, Octubre-Noviembre 2017.