

# Reporte de Evaluación 1

Diego Iván Moreno Campa

8 de Marzo, 2018

## 1 Introducción

Esta actividad es la primera evaluación de la materia de Física computacional I en la licenciatura de Física.

El propósito de esta actividad es realizar el análisis de unos datos tomados para el nivel del mar y la salinidad en un manglar conocido como El sargento.

## 2 Procedimiento

Comenzamos descargando dos archivos *sargento-270218.csv* para las lecturas del nivel del mar y *sargento-salinidad-270218.csv* para obtener las lecturas de salinidad.

El archivo *sargento-270218.csv* contiene una tabla con los datos del tiempo, presión absoluta, temperatura en grados centígrados y el nivel del agua en metros. Las mediciones fueron tomadas cada 15 minutos durante el tramo del mes de febrero de 2018.

El archivo *sargento-salinidad-270218.csv* contiene de manera similar, una tabla con datos del tiempo, "Cond high RNG", temperatura, conductividad específica y por supuesto, salinidad en partes por mil (ppt)

Se limpiaron los archivos utilizando el comando `egrep` para seleccionar todo renglón que no tuviera la primera y última fecha, después con `emacs` se eliminaron los caracteres que se repetían al utilizar `egrep`.

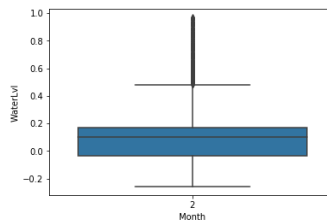
Con esto, continuamos a utilizar Python para analizar los datos, utilizando mayormente lo que ya conocíamos por las prácticas pasadas.

### 2.1 boxplot:

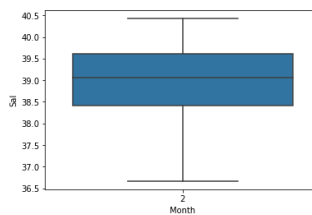
Después de acomodar los tipos de fecha de cada dataframe, se nos pide realizar gráficas de caja para lo siguiente. De la misma manera que la actividad 5, se utilizó la gráfica de cajas proporcionada por la librería Seaborn con el siguiente segmento de código, ajustándolo a nuestras necesidades:

```
ax = sns.boxplot(x="month", y="CAPE", data=df00)
plt.show()
```

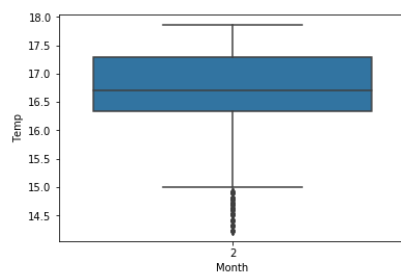
- Nivel de mar (metros):



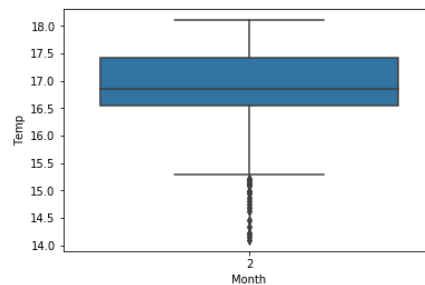
- Salinidad (ppt):



- Temperatura del agua(°C)



(a) Primer Sensor



(b) Segundo Sensor

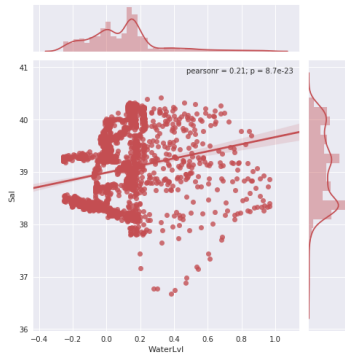
## 2.2 correlación de Pearson (Regresión líneal):

Luego, se pide observar las relaciones entre las siguientes variables. Al igual que en la actividad 5, se utilizó el siguiente código para realizar la regresión lineal, reemplazando "CAPE" y "PW" por lo que deseamos comparar:

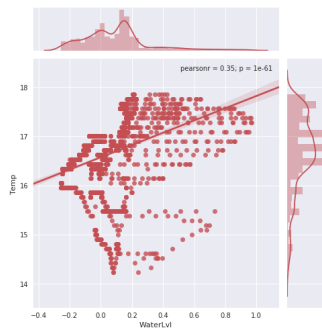
```
sns.set(style="darkgrid", color_codes=True)

g = sns.jointplot("CAPE", "PW", data=df00, kind="reg",
                  color="r", size=7)
plt.show(g)
```

- Nivel de mar-Salinidad: Para esta gráfica se necesitó juntar las columnas requeridas de cada dataframe, para esto se utilizó *pd.concat*



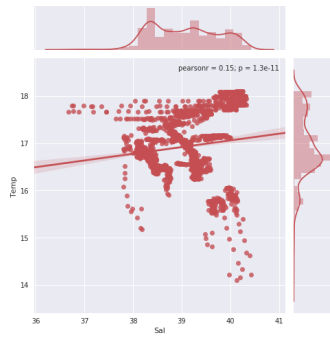
- Nivel de mar-Temperatura del agua:



- Salinidad-Temperatura del agua:

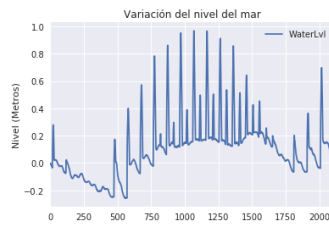
## 2.3 Gráficas independientes:

Después se procedió a realizar gráficas contra el tiempo utilizando matplotlib de los siguientes datos. Para realizar estas gráficas se utilizó el segmento de código que utilizamos en la Actividad 2 para graficar los datos individualmente, con su cantidad de mediciones:



```
plt.figure(); df.VELS.plot(); plt.legend(loc='best')
plt.title("Variación de la Rapidez de los Vientos")
plt.ylabel("Rapidez (m/s)")
plt.grid(True)
plt.show()
```

- Nivel del mar:



- Salinidad:



- Temperatura:



## 2.4 Gráficas superpuestas con doble eje vertical

Luego se nos pidió realizar gráficas superpuestas con doble eje vertical utilizando las siguientes variables. Para este caso se utilizó el siguiente segmento de código para poner dos variables en el eje-y, reemplazando la x por nuestro valor ordenado y reemplazando y1 y y2 con sus respectivas variables que deseamos comparar:

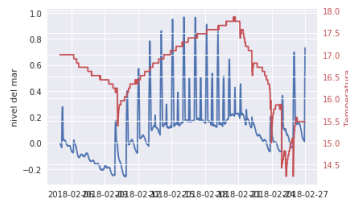
```
fig = plt.figure()
ax1 = fig.add_subplot(111)
ax1.plot(x, y1)
ax1.set_ylabel('y1')

ax2 = ax1.twinx()
ax2.plot(x, y2, 'r-')
ax2.set_ylabel('y2', color='r')
for tl in ax2.get_yticklabels():
    tl.set_color('r')
plt.show()
```

- Nivel de mar y Salinidad:



- Nivel de mar y Temperatura:

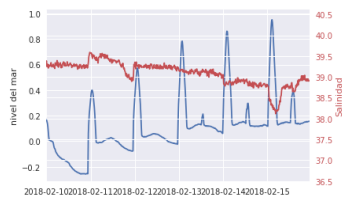


## 2.5 Doble eje-y en un intervalo:

Luego se nos pide revisar si hay alguna relación entre las variables anteriores, para esto veremos un segmento de 5 días en las gráficas anteriores: Para esto utilizamos el segmento de código anterior para realizar las gráficas, únicamente se agrego la siguiente instrucción para delimitar a la fecha entre los días 10 de febrero y 15 de febrero

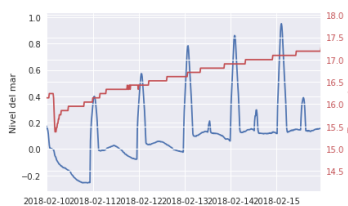
```
plt.xlim("2018-02-10 00:00:00","2018-02-15 23:00:00")
```

- Salinidad y Nivel de mar del 10 de febrero al 15 de febrero: Considero



que no se nota tan claramente una dependencia en este caso, debido a que se mantiene relativamente constante.

- Nivel de Mar y Temperatura del agua: En este caso parece ser que la



temperatura crece junto con el nivel del mar, lo cual da una señal de que haya una dependencia.