



"El saber de mis hijos
hará mi grandeza"

UNIVERSIDAD DE SONORA

División de Ciencias Exactas y Naturales

Licenciatura En Física

Física Computacional I

Reporte de Actividad 3

*"Medición de Propiedades de la Atmósfera con el
Apoyo de Sondeos por Globos Meteorológicos"*

César Omar Ramírez Álvarez

Profr. Carlos Lizárraga Celaya

Hermosillo, Sonora

Febrero 14 de 2018

Introducción

La práctica de la semana ha consistido en trabajar apoyándonos con los datos de sondeos atmosféricos que organiza la Universidad de Wyoming. Dichos datos se han tomado del sitio web de la Universidad con la finalidad de analizar el comportamiento que se ha tenido en "*Camborne Observations*" en Europa los días 22 de Junio y 22 de Diciembre de 2017.

Como es el caso de las prácticas anteriores, esta práctica también está encaminada a seguir practicando con el entorno de programación que ofrece Jupyter Notebook para programar con Python, también se han agregado las bibliotecas auxiliares que son Pandas, Numpy y Pyplot de Matplotlib que nos ofrecen distintas herramientas para organizar la información obtenida y realizar graficas como sea necesario y requerido.

Durante el desarrollo del presente reporte se mostrarán los elementos necesarios para la realización de la práctica, así como los fundamentos teóricos, procedimientos, conclusiones y bibliografías.

Fundamentos

Primeramente, vamos a conocer sobre lo que es un sondeo atmosférico y algunas de sus características. Un sondeo atmosférico es una medición de la distribución vertical de las propiedades físicas de la columna atmosférica, como presión, temperatura, velocidad del viento y dirección del viento, contenido de agua líquida, concentración de ozono, contaminación y otras propiedades. Tales mediciones se realizan en una variedad de formas que incluyen detección remota y observaciones en el sitio.

La herramienta mas utilizada para realizar los sondeos es el globo meteorológico, este aparato durante el día se encarga de recabar información de las propiedades físicas antes mencionadas, tomando datos por radiosondeos a diferentes alturas. Entre muchas otras funciones que cumplen los sondeos es validar los modelos para el pronóstico del tiempo.

Al mencionar que el globo toma datos a diferentes alturas de la atmósfera, debemos recordar la teoría de la primera práctica, podemos rescatar de ella el hecho de que la atmósfera se encuentra clasificada en capas según la relación que cumplen sus propiedades (temperatura, presión, densidad, etc.) y a diversas alturas nos encontramos con estas 5 principales capas: 1.- Tropósfera (0 a 12 km), 2.- Estratósfera (12 a 50 km), 3.- Mesósfera (50 a 80 km), 4.- Termósfera (80 a 700 km) y 5.- Exósfera (700 a 10,000 km). Sabiendo

esto es mas que suficiente para comprender y complementar la información generada con los datos de la práctica.

Análisis de Datos

Como dijimos en un principio, trabajamos apoyándonos con los datos de sondeos atmosféricos que organiza la Universidad de Wyoming. Dichos datos se han tomado del sitio web de la Universidad con la finalidad de analizar el comportamiento que se ha tenido en "*Camborne Observations*" en Europa los días 22 de Junio y 22 de Diciembre de 2017. Descargamos los dos archivos que contiene a los datos y observamos la estructura que tienen, verificando no tener datos basura y datos que nos puedan causar problemas al momento de leer desde Jupyter Notebook. Debemos de cargar las bibliotecas de Pandas, Numpy y Matplotlib antes de iniciar con cualquier lectura de datos desde nuestro archivo en Jupyter Notebook. A continuación, mostramos la estructura de los datos:

03808 Camborne Observations at 12Z 22 Jun 2017										
PRES hPa	HGHT m	TEMP C	DWPT C	RELH %	MIXR g/kg	DRCT deg	SKNT knot	THTA K	THTE K	THTV K
1006.0	88	16.6	13.9	84	10.01	290	10	289.3	317.6	291.0
1002.0	121	15.0	13.4	90	9.73	283	11	288.0	315.4	289.7
1000.0	137	14.8	13.1	90	9.56	280	12	287.9	314.8	289.6
996.0	171	14.6	12.6	88	9.27	280	15	288.1	314.2	289.7
993.0	197	14.4	12.2	87	9.06	280	16	288.1	313.7	289.7
03808 Camborne Observations at 12Z 22 Dec 2017										
PRES hPa	HGHT m	TEMP C	DWPT C	RELH %	MIXR g/kg	DRCT deg	SKNT knot	THTA K	THTE K	THTV K
1026.0	88	10.6	10.6	100	7.88	330	9	281.7	303.4	283.0
1012.0	202	10.0	10.0	100	7.65	325	13	282.1	303.3	283.4
1000.0	301	9.4	9.4	100	7.45	340	14	282.6	303.2	283.8
993.0	359	9.1	9.1	100	7.35	350	15	282.8	303.2	284.1
986.0	418	8.8	8.8	100	7.25	355	15	283.1	303.2	284.3

Como nos podemos percatar, las primeras filas no aportan nada (excepto la que nombra cada una de las columnas), por lo que, para evitar problemas al momento de leer, nos saltaremos las primeras 5 filas y nombraremos una por una cada columna.

Primeramente añadiremos las bibliotecas:

```
# Cargar a la memoria de trabajo las bibliotecas: Pandas (manejo de datos,
# Numpy (numerical python) y la biblioteca de gráficas Matplotlib
# Se asignan nombres cortos.
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

Posteriormente procedemos a leer y hacer los saltos necesarios:

```
# Lee un archivo de texto con la función Pandas
# Se leerá el archivo del 22 de Junio de 2017
df1 = pd.read_csv('DatosJun.txt', skiprows=5, sep='\s+')
# Se leerá el archivo del 22 de Diciembre de 2017
df2 = pd.read_csv('DatosDic.txt', skiprows=5, sep='\s+')
# Nombramos las columnas
df1.columns = ['PRES', 'HGHT', 'TEMP', 'DWPT', 'RELH', 'MIXR', 'DRCT', 'SKNT', 'THTA', 'THTE', 'THTV']
df2.columns = ['PRES', 'HGHT', 'TEMP', 'DWPT', 'RELH', 'MIXR', 'DRCT', 'SKNT', 'THTA', 'THTE', 'THTV']
```

Verificamos que al leer ya nos aparezcan los nombres de las columnas y los datos:

	PRES	HGHT	TEMP	DWPT	RELH	MIXR	DRCT	SKNT	THTA	THTE	THTV
0	1002.0	121	15.0	13.4	90	9.73	283.0	11.0	288.0	315.4	289.7
1	1000.0	137	14.8	13.1	90	9.56	280.0	12.0	287.9	314.8	289.6
2	996.0	171	14.6	12.6	88	9.27	280.0	15.0	288.1	314.2	289.7
3	993.0	197	14.4	12.2	87	9.06	280.0	16.0	288.1	313.7	289.7
4	990.0	222	14.2	12.6	90	9.35	280.0	17.0	288.1	314.5	289.8
5	988.0	239	14.0	12.9	93	9.55	282.0	17.0	288.1	315.0	289.8

	PRES	HGHT	TEMP	DWPT	RELH	MIXR	DRCT	SKNT	THTA	THTE	THTV
0	1012.0	202	10.0	10.0	100.0	7.65	325.0	13.0	282.1	303.3	283.4
1	1000.0	301	9.4	9.4	100.0	7.45	340.0	14.0	282.6	303.2	283.8
2	993.0	359	9.1	9.1	100.0	7.35	350.0	15.0	282.8	303.2	284.1
3	986.0	418	8.8	8.8	100.0	7.25	355.0	15.0	283.1	303.2	284.3
4	975.0	510	8.3	8.3	100.0	7.09	345.0	13.0	283.5	303.2	284.7
5	961.0	630	7.7	7.7	100.0	6.88	350.0	11.0	284.0	303.3	285.2

En la estructura de los datos nos dimos cuenta que al final de documento tenemos datos incompletos, por lo que evitaremos que se lean para evitar posibles errores usando lo siguiente para cada mes:

```
# Eliminamos datos incompletos
df1 = df1.drop(df1.index[len(df1)-2])
df2 = df2.drop(df2.index[len(df2)-1])
```

Comprobamos que nuestros datos son valores numéricos para poder operar:

# Tipo de datos (Junio)			# Tipo de datos (Diciembre)	
df1.dtypes			df2.dtypes	
PRES	float64		PRES	float64
HGHT	int64		HGHT	int64
TEMP	float64		TEMP	float64
DWPT	float64		DWPT	float64
RELH	int64		RELH	float64
MIXR	float64		MIXR	float64
DRCT	float64		DRCT	float64
SKNT	float64		SKNT	float64
THTA	float64		THTA	float64
THTE	float64		THTE	float64
THTV	float64		THTV	float64
dtype: object			dtype: object	

<-Junio y Diciembre->

Al tener todos nuestros datos como valores numéricos no tenemos problema para iniciar con las gráficas requeridas.

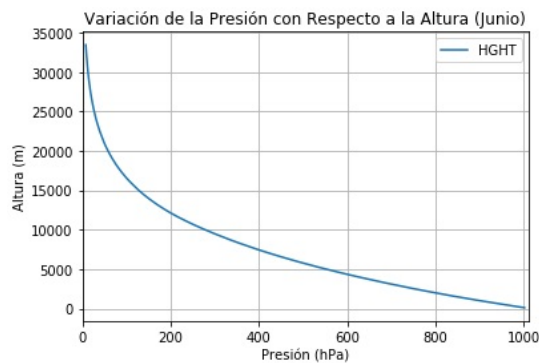
Resultados

Para la presentación de las gráficas se mostrarán tanto el código como la gráfica para ambos meses. Considerando que para una de ellas se tuvo que trabajar con limites en sus

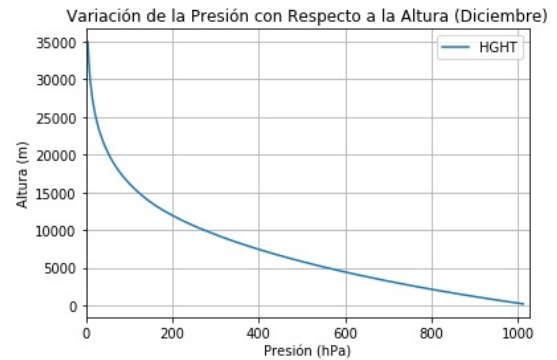
ejes para poder apreciar y comprender mejor la apreciación de lo que muestran.

La primera de ellas será una gráfica de la variación de la presión con respecto a la altura:

```
# Graficamos la variación de la presión con respecto a la altura (Junio)
df01 = df1[['HGHT', 'PRES']]
plt.figure(); df01.plot(x='PRES');plt.legend(loc='best')
plt.title('Variación de la Presión con Respecto a la Altura (Junio)')
plt.xlim([0,1010])
plt.xlabel('Presión (hPa)')
plt.ylabel('Altura (m)')
plt.grid(True)
plt.show()
```



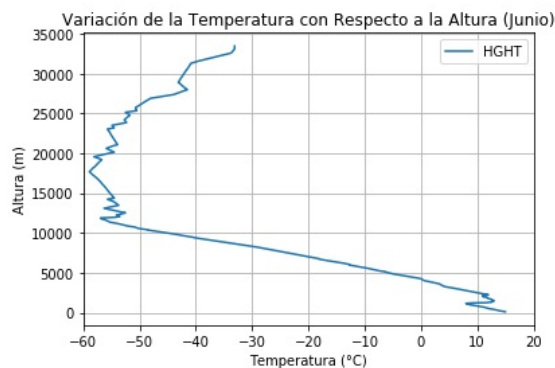
```
# Graficamos la variación de la presión con respecto a la altura (Diciembre)
df02 = df2[['HGHT', 'PRES']]
plt.figure(); df02.plot(x='PRES');plt.legend(loc='best')
plt.title('Variación de la Presión con Respecto a la Altura (Diciembre)')
plt.xlim([0,1025])
plt.xlabel('Presión (hPa)')
plt.ylabel('Altura (m)')
plt.grid(True)
plt.show()
```



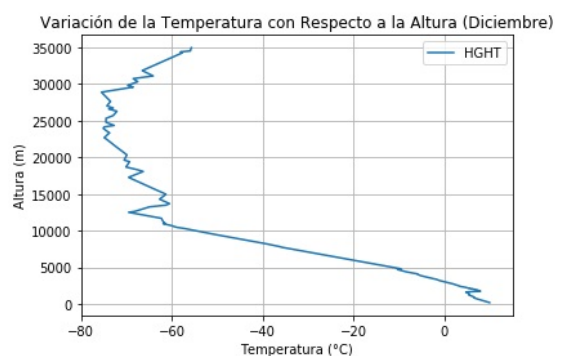
Se hace notorio el descenso de presión mientras mas alejados de la superficie de la Tierra nos encontremos.

Continuando, tenemos una gráfica de la variación de la temperatura con respecto a la altura:

```
# Graficamos la variación de la temperatura con respecto a la altura (Junio)
df01 = df1[['HGHT', 'TEMP']]
plt.figure(); df01.plot(x='TEMP');plt.legend(loc='best')
plt.title('Variación de la Temperatura con Respecto a la Altura (Junio)')
plt.xlim([-60,20])
plt.xlabel('Temperatura (°C)')
plt.ylabel('Altura (m)')
plt.grid(True)
plt.show()
```



```
# Graficamos la variación de la temperatura con respecto a la altura (Diciembre)
df02 = df2[['HGHT', 'TEMP']]
plt.figure(); df02.plot(x='TEMP');plt.legend(loc='best')
plt.title('Variación de la Temperatura con Respecto a la Altura (Diciembre)')
plt.xlim([-80,15])
plt.xlabel('Temperatura (°C)')
plt.ylabel('Altura (m)')
plt.grid(True)
plt.show()
```

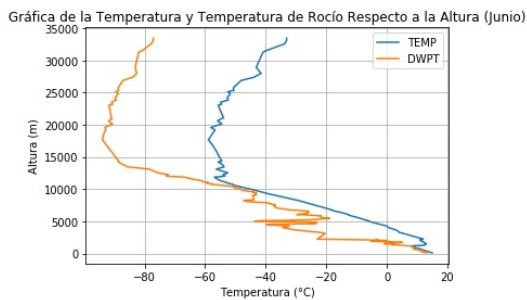


La pregunta propuesta *¿Hay cambios significativos en la tropopausa entre las dos fechas?* Podemos afirmar que si se presenta un cambio. Según conocemos la tropopausa se encuentra entre los 9 y 17 km, observamos las graficas y alrededor de esos valores encontramos

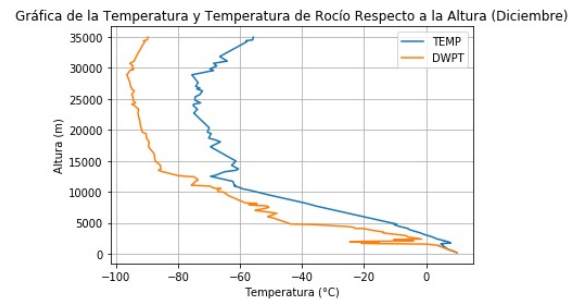
un descenso (pareciera que es el máximo) y aumento de nuevo conforme ganamos mas altura.

Siguiendo, podemos ver la gráfica de la temperatura y temperatura de rocío, como función de la altura:

```
# Gráfica de la temperatura (TEMP) y temperatura de rocío (DWP), como función de la altura (Junio)
x1=df1['TEMP']
x2=df1['DWP']
y=df1['HGHT']
plt.plot(x1, y, label='TEMP')
plt.plot(x2, y, label='DWP')
plt.legend(loc='best')
plt.title("Gráfica de la Temperatura y Temperatura de Rocío Respecto a la Altura (Junio)")
plt.xlabel("Temperatura (°C)")
plt.ylabel("Altura (m)")
plt.grid(True)
plt.show()
```



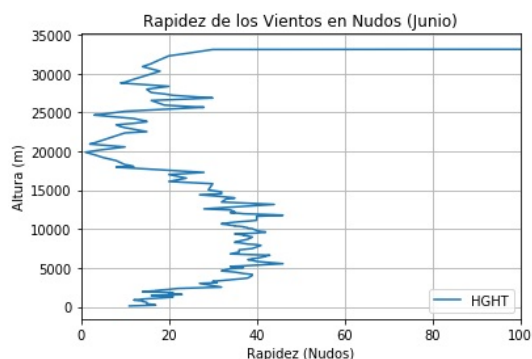
```
# Gráfica de la temperatura (TEMP) y temperatura de rocío (DWP), como función de la altura (Diciembre)
x1=df2['TEMP']
x2=df2['DWP']
y=df2['HGHT']
plt.plot(x1, y, label='TEMP')
plt.plot(x2, y, label='DWP')
plt.legend(loc='best')
plt.title("Gráfica de la Temperatura y Temperatura de Rocío Respecto a la Altura (Diciembre)")
plt.xlabel("Temperatura (°C)")
plt.ylabel("Altura (m)")
plt.grid(True)
plt.show()
```



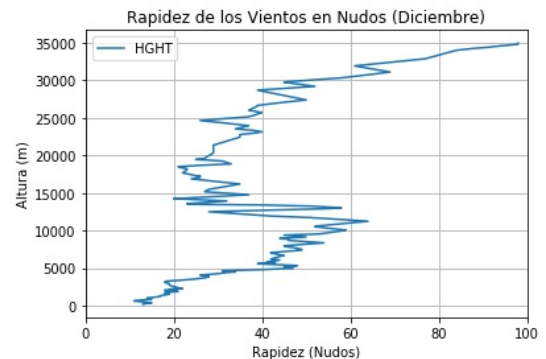
Según las graficas podemos inferir que desde cierta capa (Estratósfera) estas dos temperaturas llevan el mismo comportamiento, salvo que la temperatura de rocío permanece mas baja respecto a la temperatura.

La próxima grafica nos presenta la rapidez de los vientos en nudos usando todos los datos proporcionados:

```
# Gráfica de La rapidez de Los vientos en nudos (SKNT) (Junio)
df01 = df1[['SKNT', 'HGHT']]
plt.figure(); df01.plot(x='SKNT'); plt.legend(loc='best')
plt.title("Rapidez de los Vientos en Nudos (Junio)")
plt.xlabel("Rapidez (Nudos)")
plt.ylabel("Altura (m)")
plt.xlim([0,100])
plt.grid(True)
plt.show()
```



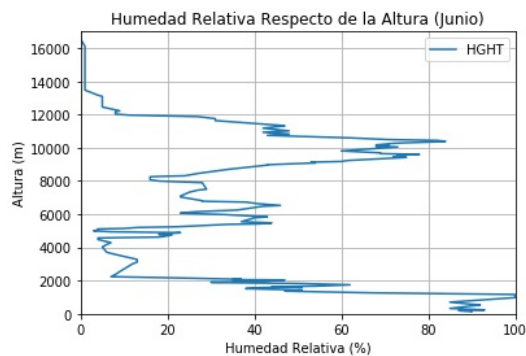
```
# Gráfica de La rapidez de Los vientos en nudos (SKNT) (Diciembre)
df02 = df2[['SKNT', 'HGHT']]
plt.figure(); df02.plot(x='SKNT'); plt.legend(loc='best')
plt.title("Rapidez de los Vientos en Nudos (Diciembre)")
plt.xlabel("Rapidez (Nudos)")
plt.ylabel("Altura (m)")
plt.xlim([0,100])
plt.grid(True)
plt.show()
```



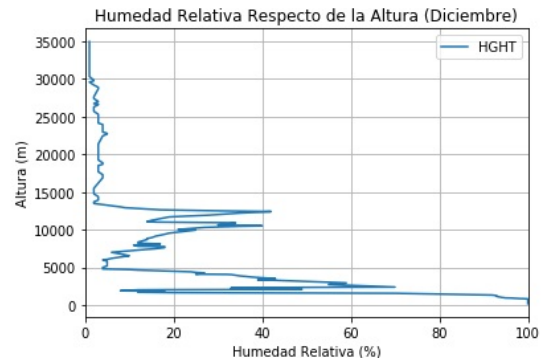
Con respecto a la rapidez de los vientos, observamos que no tenemos un "buen comportamiento" es decir, constante, aunque en Junio es mejor comportado que en Diciembre.

Y por último, tenemos la gráfica de la humedad relativa como función de la altura:

```
# Gráfica de La humedad relativa como función de La altura (Junio)
df01 = df1[['RELH','HGHT']]
plt.figure(); df01.plot(x='RELH');plt.legend(loc='best')
plt.title('Humedad Relativa Respecto de la Altura (Junio)')
plt.xlabel('Humedad Relativa (%)')
plt.ylabel('Altura (m)')
plt.xlim([0,100])
plt.ylim([0,17000])
plt.grid(True)
plt.show()
```



```
# Gráfica de La humedad relativa como función de La altura (Diciembre)
df02 = df2[['RELH','HGHT']]
plt.figure(); df02.plot(x='RELH');plt.legend(loc='best')
plt.title('Humedad Relativa Respecto de la Altura (Diciembre)')
plt.xlabel('Humedad Relativa (%)')
plt.ylabel('Altura (m)')
plt.xlim([0,100])
plt.grid(True)
plt.show()
```



En ambas graficas es notoria su variacion, aunque se estabiliza a menor altura en Diciembre.

Conclusión

Desde el moento que leí la práctica pense que todo iba a ser fácil, pero con el desarrollo me di cuenta que no era del todo fácil, pues al mostrarnos una cantidad grande de datos el analizarlos y clasificarlos para leerlos se puede complicar demasiado, segun los límites de cada quién, por lo qué se ocupó visitar frecuentemente varios sitios de complementación para poder tener listos los datos al moento de realizar sus graficas respectivas.

La parte gráfica no se me complico, pues ya se habia estado trabajando con ellas en actividades anteriores y partí de esos códigos para realizar éstas.

En cuanto a la teoría de la actividad me pareció muy interesante y muy útil para presentarse de éstam manera, la de analizar datos de este tipo y ver lo que realmente sucede a diversas alturas de la atmósfera y a su vez, complementando las prácticas anteriores.

Puedo decir que se cumplió el obojtivo, pues se llegó a realizar la práctica con éxito y sobre todo, darnos cuenta que el análisis de datos puede complicarse según los límites y herramientas de cada quién.

Bibliografía

- Atmospheric sounding. (2018). En.wikipedia.org. Recuperado el 13 de Febrero de 2018, desde https://en.wikipedia.org/wiki/Atmospheric_sounding
- Atmosphere of Earth. (2018). En.wikipedia.org. Recuperado el 13 de Febrero de 2018, desde https://en.wikipedia.org/wiki/Atmosphere_of_Earth
- Tutorial de pandas | Pybonacci. (2018). Pybonacci.org. Recuperado el 8 de Febrero de 2018, desde <http://www.pybonacci.org/tag/tutorial-de-pandas/>
- Usando El Tipo DataFrame de Python Pandas. (2018). Relguzman.blogspot.mx. Recuperado el 8 de Febrero de 2018, desde <http://relguzman.blogspot.mx/2015/06/usando-el-tipo-dataframe-de-python.html>

Apéndice

1. ¿Cuál es tu opinión general de esta actividad?

Me parecio muy interesante por que complementamos la actividad anterior, pero esta vez se trabajo más con análisis y preparación de datos para despues graficar.

2. ¿Qué fue lo que más te agradó? ¿Lo que menos te agradó?

En general me gustó todo, pero el hecho de graficar de una gran cantidad de datos me parece sorprendente, la verdad cada práctica parece gustarme más Phyton como enguaje de programación, algo que en un principio me desagradó fue cuando tuvimos que preparar los datos (evitar cosas innecesarias), pues puede perderse mucho tiempo.

3. ¿Que consideras que aprendiste en esta actividad?

En general, formas de darle formato a las gráficas (color, estilo, etiquetas, etc) y comandos para análisis y acomodo de datos.

4. ¿Qué le faltó? ¿O le sobró?

Estuvo muy completa para el nivel en el que vamos, considero que conforme avancemos seguiremos aprendiendo y complementando mas sobre la programación en Phyton.

5. ¿Que mejoras sugieres a la actividad?

Siempre mostrar bibliografía de donde apoyarnos, no solo para la parte teórica si no para comando y herramientas posibles a utilizar.