



UNIVERSIDAD DE SONORA

DIVISIÓN DE CIENCIAS EXACTAS Y NATURALES  
DEPARTAMENTO DE FÍSICA

---

VISUALIZANDO DATOS CON PANDAS Y  
MATPLOTLIB

---

ALVAREZ SANCHEZ FRANCISCO EDUARDO

PROFESOR: CARLOS LIZÁRRAGA CELAYA

28 DE FEBRERO DE 2017

# Índice

<b>1. Resumen</b>	<b>2</b>
<b>2. Introducción</b>	<b>2</b>
<b>3. Desarrollo</b>	<b>2</b>
3.1. Gráficas . . . . .	4
3.2. E Tefigrama . . . . .	8
<b>4. Conclusión</b>	<b>10</b>

## 1. Resumen

En la siguiente actividad fue para aprender un poco mas acerca de área de trabajo de python, los pandas utilizados para poder trabajar con las graficas y utilizaremos metodos para comparar los datos interpretados.

Aprenderemos también a trabajar con tefigramas en python y cual puede ser una de las utilidades de estos.

## 2. Introducción

En el siguiente trabajo se estudiara otro aspecto de los sondeos de la localizacion escogida en un principio. Usaremos los datos de presion, temperatura y temperatura de rocío, en este caso de la fecha del 15 de Febrero de 2017. Para poder trabajar con los datos fue necesario trabajar con la plataforma de Jupyter Notebook, con esto trabajamos con python para poder graficar mas sencillamente.

Para comparar las gráficas utilizaremos un paquete llamado tephi que nos servira para realizar tefigramas y compararlo con el que se realiza donde se realizan los sondeos. Para esto usaremos un fork a nuestro propio repositorio de Github y clonarlo en la computadora para poder instalarlo.

```
https://github.com/franciscoeas/tephi  
pip install --user tephi
```

## 3. Desarrollo

Para graficar el tephi necesitamos la temperatura contra la presión y la temperatura de rocío contra presión, para cada uno tuvimos que realizar dos documentos con los datos separados y ordenados.

Ya teniendo eso los leemos desde python para poder trabajar con ellos. Hacemos la siguiente tabla para poder trabajar con los datos.

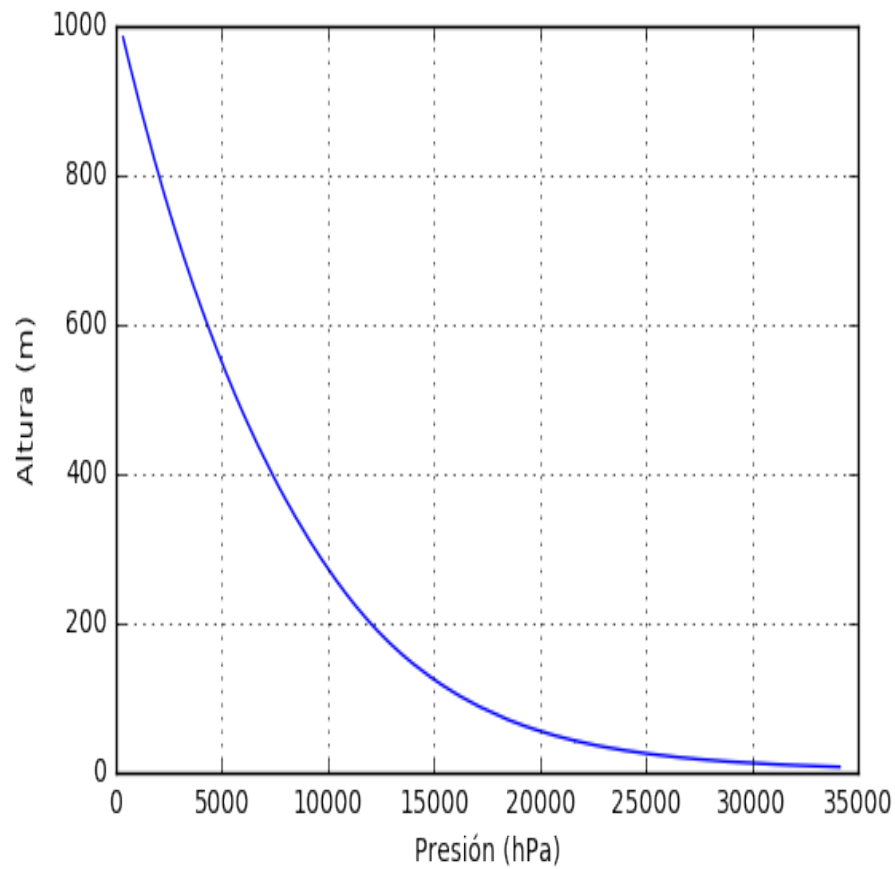
	Presión	Altura	Temperatura	DWPT	RELH	MIXR	DRCT	SKNT	THTA	THTE	THTV
0	983.0	338	7.0	1.0	66	4.20	283	8	281.5	293.4	282.2
1	978.0	380	7.4	0.4	61	4.04	287	10	282.3	293.9	283.0
2	952.0	603	6.6	-2.4	53	3.38	309	20	283.7	293.5	284.3
3	951.1	610	6.5	-2.5	53	3.37	310	20	283.7	293.5	284.3
4	925.0	839	4.8	-4.2	52	3.04	330	19	284.2	293.1	284.7
5	916.5	914	4.3	-5.0	51	2.89	340	20	284.4	292.9	284.9
6	896.0	1098	3.0	-7.0	48	2.54	358	23	284.9	292.5	285.4
7	882.6	1219	2.4	-7.6	48	2.45	10	25	285.5	292.8	285.9
8	850.0	1523	0.8	-9.2	47	2.25	10	25	287.0	293.7	287.4
9	849.0	1532	0.8	-9.2	47	2.25	10	25	287.1	293.8	287.5
10	844.0	1580	0.8	-8.2	51	2.45	10	26	287.6	294.9	288.0
11	837.0	1647	0.8	-4.2	69	3.36	10	28	288.2	298.2	288.8
12	832.0	1695	0.6	-4.4	69	3.33	10	29	288.5	298.4	289.1
13	818.1	1829	-0.3	-4.7	72	3.31	10	33	288.9	298.8	289.5
14	787.5	2134	-2.4	-5.4	80	3.27	15	37	289.8	299.6	290.4
15	781.0	2200	-2.9	-5.5	82	3.26	16	33	290.0	299.8	290.6
16	771.0	2303	0.2	-5.8	64	3.23	18	28	294.4	304.2	295.0
17	758.1	2438	-0.1	-7.0	60	2.99	20	20	295.5	304.7	296.0
18	729.7	2743	-0.9	-9.7	51	2.51	315	20	297.9	305.8	298.4

### 3.1. Gráficas

Ahora para graficar usaremos los siguientes comandos: Antes que nada declaramos las variables

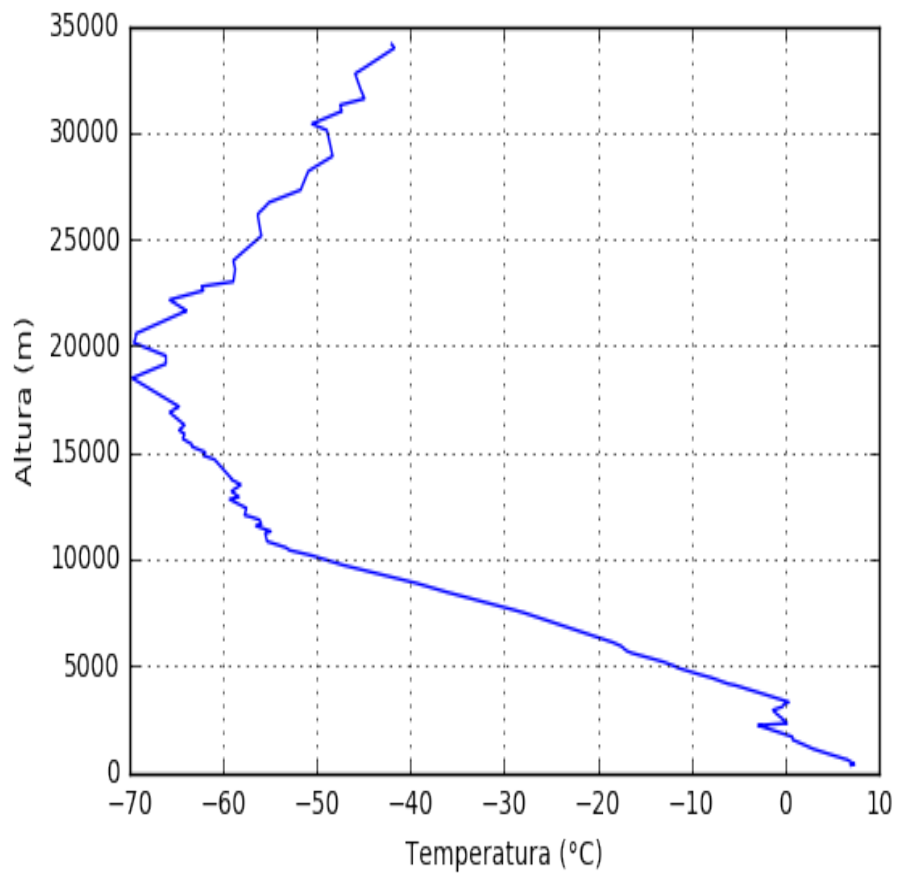
Graficaremos lo siguiente: Para graficar la presión contra altura

```
x=df[u'HGHT']  
y=df[u'PRES']  
Para graficar:  
plt.plot(x,y)  
plt.grid(True)  
plt.xlabel('Altura (m)')  
plt.ylabel('Presión (hPa)')
```



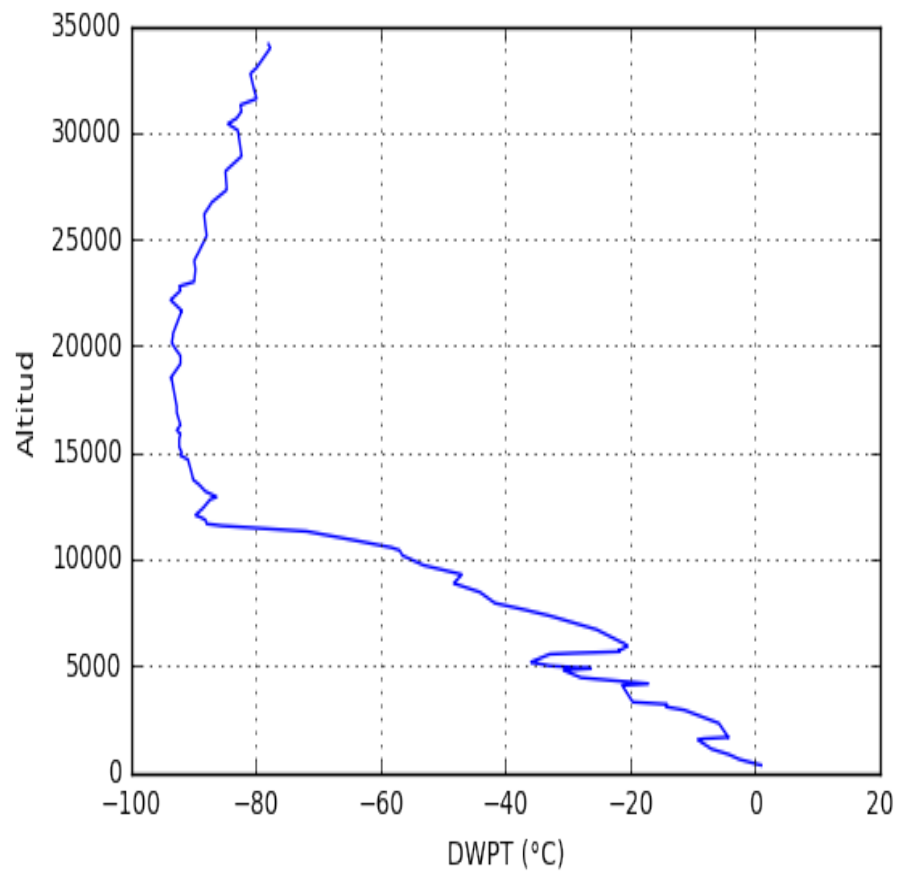
Para graficar temperatura contra altura:

```
x=df[u'Temperatura']  
y=df[u'Altura']  
mplt.plot(x,y)  
mplt.grid(True)  
plt.xlabel('Temperatura (°C)')  
plt.ylabel('Altura (m)')
```



Para graficar la temperatura de rocío contra la altitud:

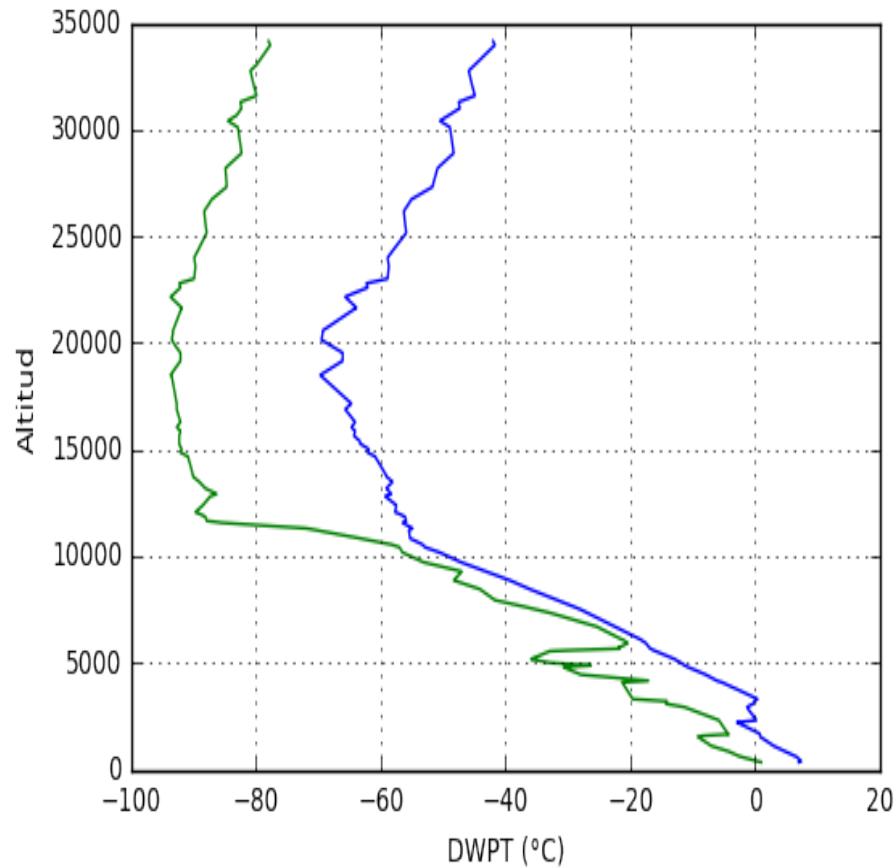
```
x=df[u'DWPT']  
y=df[u'Altura']  
mplt.plot(x,y)  
mplt.grid(True)  
plt.xlabel('DWPT (°C)')  
plt.ylabel('Altitud')
```



Para graficar la comparativa entre las temperaturas:

```
Para la primera gráfica
x=df[u'Temperatura']
y=df[u'Altura']mplt.plot(x,y)
mplt.grid(True)
plt.xlabel('Temperatura (°C)')
plt.ylabel('Altura (m)')
Para la segunda gráfica
x=df[u'DWPT']
y=df[u'Altura']
mplt.plot(x,y)
mplt.grid(True)
plt.xlabel('DWPT (°C)')
plt.ylabel('Altitud')
```





### 3.2. E Tefigrama

El tefigrama es un diagrama termodinámico que se usa para trazar perfiles verticales de temperatura, humedad y viento atmosférico. Este se utiliza para evaluar una amplia gama de condiciones meteorológicas, como la estabilidad atmosférica.

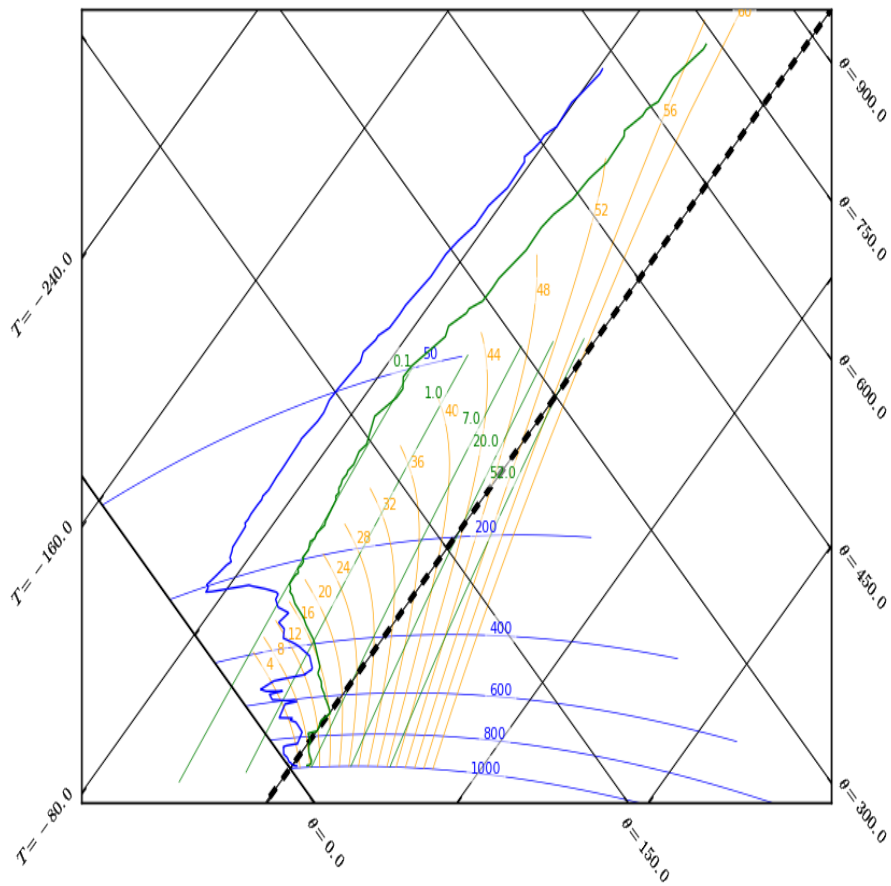
Ahora para hacer el tefigrama pondremos los siguientes comandos:

```
import os.path
import teph as tph
dew_point = pd.read_csv("/home/franciscoeas/Actividad 4/DATOS/
presionvsrocio.csv", names=["Presión", "DWPT"])
```

```

dry_bulb = pd.read_csv("/home/franciscoeas/Actividad 4/DATOS/
presionvstemperatura.csv", names=["Presión", "TEMP"])
tpg = tph.Tephigram()
tpg.plot(dew_point)
tpg.plot(dry_bulb)
plt.title('Presión vs. Temperatura y Temperatura de Rocío')
plt.show()

```



## 4. Conclusión

En el trabajo anterior aprendimos a utilizar algunos de los comando de python y practicamos algunos que se utilizaron en las prácticas anteriores. Aprendimos a utilizar algunos paquetes como tephi para gráficar tafigramas y poder comparar las gráficas comunes en un esquema diferente.

## Referencias

- [1] University of Wyoming, sitio web: <http://weather.uwyo.edu/upperair/sounding.html>  
Última fecha de consulta: 12 de Febrero 2017.
- [2] The Tephi user guide, sitio web: <http://tephi.readthedocs.io/en/latest/index.html>  
Última fecha de consulta: 27 de Febrero 2017.