

# Actividad 5: Preparando datos con ayuda de Emacs

Melissa Matrecitos Avila

6 de Marzo de 2018

## 1 Introducción

El siguiente reporte describe la actividad 5 realizada en el curso de Física Computacional 1, la cual trato sobre la limpieza y análisis de los datos obtenidos en la actividad pasada, enfocados a las variables de CAPE y PW.

A continuación se describirá el significado físico de las variables mencionadas (CAPE y PW), también La preparación del documento para su limpieza en EMACS y su análisis en Python. Por último se presenta una conclusión sobre la actividad.

## 2 Conceptos importantes

Dos conceptos muy importantes en la actividad fueron el CAPE y PW, de los cuales se hizo un análisis de los datos obtenidos en la estación de Lindenberg, Alemania. Sin embargo antes de pasar a el trabajo con los datos, se debe hablar sobre las variables que se estudiaron.

Las siglas CAPE (Convective Available Potential Energy) hacen referencia a la energia potencial disponible para la convección en un momento dado. CAPE es efectivamente la flotabilidad positiva de un paquete de aire y es un indicador de la inestabilidad atmosférica , lo que lo hace muy valioso para predecir el mal tiempo. Se trata de un parámetro que nos indica cuanta energía está disponible para la convección en caso de que esta se inicie. Por lo tanto, la consulta de este parámetro se tiene que complementar siempre con la lectura de otros campos del modelo que nos permitan determinar la probabilidad de que la convección se inicie. Es una forma de inestabilidad del fluido que se encuentra en atmósferas térmicamente estratificadas en las que un fluido más frío se superpone a uno más cálido. Como se explica a continuación, cuando una masa de aire es inestable, el elemento de la masa de aire que se desplaza hacia arriba se acelera por la diferencia de presión entre el aire desplazado y el aire ambiente a la altitud (más alta) a la que se desplazó. Esto generalmente crea nubes desarrolladas verticalmente por convección, debido al movimiento ascendente, que eventualmente puede conducir a tormentas eléctricas.

El agua precipitable, o PW por sus siglas en inglés, es la profundidad del agua en una columna de la atmósfera, si toda el agua en esa columna se precipitó como lluvia. Como profundidad, el agua precipitable se mide en milímetros o pulgadas. A menudo abreviado como "TPW" = total precipitable water. Hay diferentes técnicas de medición:

- Un tipo de medición se basa en la medición de la irradiancia solar en dos longitudes de onda, una en una banda de absorción de agua, y la otra no. La columna de agua precipitable se determina usando las irradiancias en estas bandas y la ley de Beer-Lambert .
- El agua precipitable también se puede calcular mediante la integración de datos de radiosonda ( humedad relativa , presión y temperatura ) en toda la atmósfera .
- Los datos se pueden ver en un índice Lifted-K. Los números representan pulgadas de agua como se mencionó anteriormente para una ubicación geográfica.
- Recientemente, se han desarrollado métodos que usan el Sistema de Posicionamiento Global .
- Se han realizado algunos trabajos para crear relaciones empíricas entre la humedad específica de la superficie y el agua precipitable en base a mediciones localizadas (generalmente un polinomio de segundo a quinto orden). Sin embargo, este método no ha recibido un uso generalizado en parte porque la humedad es una medida local y el agua precipitable es una medición de columna total.

### 3 Proceso de limpieza y preparación de los datos

En esta sección se explica el procedimiento utilizado para preparar los datos para su análisis.

Lo primero que se hizo fue crear un script que seleccionaría los renglones con información de la variable CAPE y PW del archivo con los datos de todo el año creado en la actividad 4, y los enviara a un nuevo archivo llamado "df2017CAPE\_PW.csv":

```
#!/bin/bash
egrep '10393|CAPE|Precip' df2017-2.csv > df2017CAPE_PW.csv
```

Transformando el archivo que se muestra a la izquierda en el de la derecha, donde solo esta la información que nos interesa, el cual se copió a la carpeta de Actividad 5:



Por último se separaron los datos en dos archivos, correspondientes al lanzamiento, es decir los 00Z y 12z utilizando el siguiente script:

```
#!/bin/bash
egrep '00Z' df2017CAPE_PW.csv > df2017CAPE_PW_00Z.csv
egrep '12Z' df2017CAPE_PW.csv > df2017CAPE_PW_12Z.csv
```

El cual selecciona los renglones que tienen 00Z y los envía a un archivo con nombre df2017CAPE\_PW\_00Z.csv, y a los renglones con 12Z los envía al archivo con nombre:df2017CAPE\_PW\_12Z.csv.

A cada uno de los archivos seleccionados, se le eliminó el inicio del renglón, ya que todos los datos correspondían al mismo lanzamiento (00 o 12)

Los archivos quedarón de la siguiente manera, donde el de la izquierda corresponde al 00 y el de la derecha al 12:

01 01 2017, 0.00, 8.84	01 01 2017, 0.00, 7.22
02 01 2017, 0.00, 7.56	02 01 2017, 0.00, 6.41
03 01 2017, 0.00, 6.79	03 01 2017, 0.00, 11.44
04 01 2017, 0.00, 12.29	04 01 2017, 41.96, 7.89
05 01 2017, 0.00, 6.30	05 01 2017, 4.79, 4.89
06 01 2017, 0.00, 3.14	06 01 2017, 0.00, 3.80
07 01 2017, 0.00, 4.53	07 01 2017, 0.00, 7.60
08 01 2017, 0.00, 8.42	08 01 2017, 0.00, 6.66
09 01 2017, 0.00, 6.58	09 01 2017, 0.00, 7.39
10 01 2017, 0.00, 6.92	10 01 2017, 0.00, 4.93
11 01 2017, 0.00, 5.40	11 01 2017, 0.00, 6.42
12 01 2017, 0.00, 10.23	12 01 2017, 0.16, 9.90
13 01 2017, 0.00, 10.85	13 01 2017, 0.00, 10.45
14 01 2017, 4.12, 6.34	14 01 2017, 0.00, 7.45
15 01 2017, 0.07, 6.59	15 01 2017, 0.00, 7.81
16 01 2017, 0.00, 7.38	16 01 2017, 0.00, 4.14
17 01 2017, 0.00, 5.23	17 01 2017, 1.27, 4.40
18 01 2017, 0.00, 4.94	18 01 2017, 0.00, 6.74
19 01 2017, 0.00, 9.58	19 01 2017, 0.00, 9.74
20 01 2017, 0.00, 10.37	20 01 2017, 0.00, 10.69
21 01 2017, 0.00, 9.44	21 01 2017, 0.00, 9.34
22 01 2017, 0.00, 7.52	22 01 2017, 0.00, 6.17
23 01 2017, 0.00, 4.71	23 01 2017, 0.00, 5.12
24 01 2017, 0.00, 6.46	24 01 2017, 0.00, 6.43
25 01 2017, 0.00, 9.35	25 01 2017, 0.00, 7.85
26 01 2017, 0.00, 10.83	26 01 2017, 0.00, 7.92
27 01 2017, 0.00, 6.62	27 01 2017, 0.00, 4.73
28 01 2017, 0.00, 4.61	28 01 2017, 0.00, 3.80
29 01 2017, 0.00, 6.09	29 01 2017, 0.00, 10.86
30 01 2017, 0.00, 10.79	30 01 2017, 0.00, 12.91
31 01 2017, 0.00, 15.53	31 01 2017, 0.00, 10.22
01 02 2017, 0.00, 12.92	01 02 2017, 0.00, 8.27
02 02 2017, 0.00, 9.14	02 02 2017, 0.00, 6.90
--:-- df2017CAPE_PW_00Z.csv	--:-- df2017CAPE_PW_12Z.csv
For information about GN	For information about GN

## 4 Análisis de datos

A continuación se muestra solo el análisis de los datos correspondientes al archivo de los lanzamientos en 00Z, ya que para el archivo de 12Z se hizo lo mismo: Primeramente se importaron las librerías a utilizar, en cuales había una nueva "datetime" que sirve para dar formato de fecha:

```
#Importar bibliotecas
import pandas as pd
import numpy as np
from datetime import datetime
```

Posteriormente se colocaron los datos en un dataframe, asignando nombres a la columnas y trasformando la variable CAPE a numérica:

```
# Leer archivo de datos
# Convertir la columna CAPE de objeto a número
#Mostrar cabeza de la tabla
df00=pd.read_csv("df2017CAPE_Pw_00Z.csv", header=None, names=['Date', 'CAPE', 'PW'])
df00.CAPE=pd.to_numeric(df00.CAPE, errors='coerce')
df00.head()
```

Quedándonos la siguiente tabla:

	Date	CAPE	PW
0	01 01 2017	0.0	8.84
1	02 01 2017	0.0	7.56
2	03 01 2017	0.0	6.79
3	04 01 2017	0.0	12.29
4	05 01 2017	0.0	6.30

Siguiendo con la actividad, se transformó la columna con nombre Date de tipo objeto a fecha, esto con la nueva biblioteca:

```
# Convertir la cadena de caracteres 'Date' en variable temporal 'NDate'
df00['NDate'] = pd.to_datetime(df00['Date'], format='%d %m %Y')
df00['month'] = df00['NDate'].dt.month
df00.head()
```

Dejando los datos listos para hacer las gráficas pertinentes:

	Date	CAPE	PW	Ndate	month
0	01 01 2017	0.0	8.84	2017-01-01	1
1	02 01 2017	0.0	7.56	2017-01-02	1
2	03 01 2017	0.0	6.79	2017-01-03	1
3	04 01 2017	0.0	12.29	2017-01-04	1
4	05 01 2017	0.0	6.30	2017-01-05	1

Por último se hicieron las gráficas solicitadas con las siguientes entradas:

```
# graficar Boxplots por mes
# Biblioteca Seaborn
import seaborn as sns
import matplotlib.pyplot as plt
ax = sns.boxplot(x="month", y="CAPE", data=df00)
plt.show()

# Biblioteca Seaborn
# Gráfica de la relación entre ambas variables
import seaborn as sns
sns.set(style="darkgrid", color_codes=True)
g = sns.jointplot("CAPE", "PW", data=df00, kind="reg",
                  color="r", size=7)
plt.show(g)
```

```
# graficar Boxplots por mes
# Biblioteca Seaborn
import seaborn as sns
import matplotlib.pyplot as plt
ax = sns.boxplot(x="month", y="PW", data=df00)
plt.show()

# Gráfica de Las dos muestras
g = sns.lmplot(x="CAPE", y="PW", hue="month",
               truncate=True, size=5, data=df00)
plt.show(g)
```

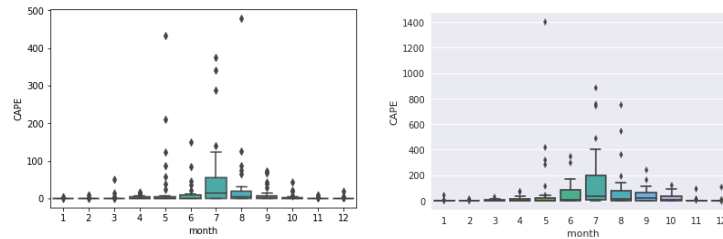
Las cuales serán explicadas en la sección de "Resultados del análisis".

## 5 Resultados del análisis

A continuación se presentan las gráficas producidas en el análisis de los datos. En todas las gráficas mostradas, las de la izquierda corresponden a los lanzamientos en 00Z y las de la derecha a los lanzamientos en 12Z.

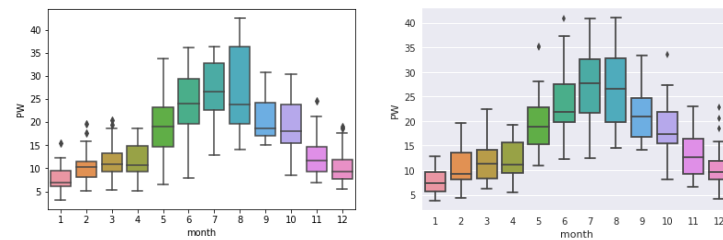
### CAPE:

Al analizar esta variable se puede decir que a diferencia de la variable CAPE, los valores de PW están alejados del cero, esto se intensifica en los meses de mayo a septiembre, llegando a tener valores desde 20 hasta 35, que corresponde al verano en Alemania. Mientras que en invierno, tiene valores cercanos al 10. Otra diferencia notable entre las variables CAPE y PW es que esta última no tiene tantos valores fuera del rango como la primera.



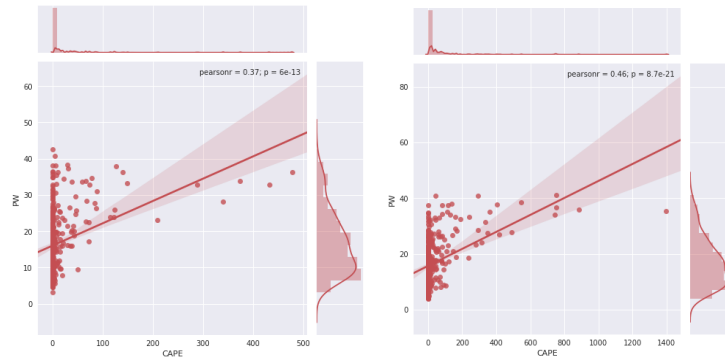
### PW:

Aquí se muestran los diagramas de caja para los dos lanzamientos con respecto a la variable CAPE. Se puede observar como mantiene valores muy cercanos al cero, excepto por los meses de mayo a agosto, dónde se da un aumento considerable, es decir hay valores fuera del rango.



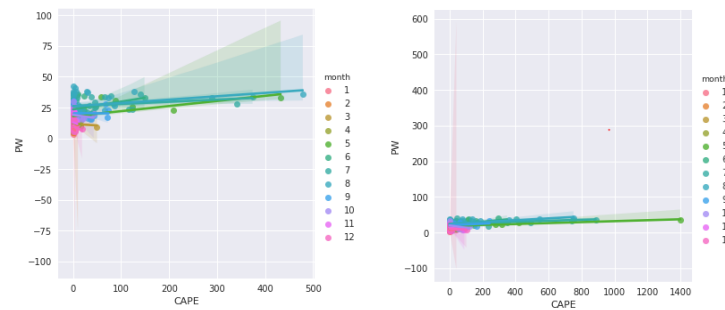
### RELACION PW y CAPE:

Las siguientes gráficas muestran la distribución de cada variable (arriba la de CAPE y al lado la de PW) y la relación que hay entre ambas. En cuanto a las distribuciones de cada variable no parece haber mucha diferencia entre los lanzamientos, ni la recta que mejor se ajusta. Las gráficas también muestran el coeficiente de correlación de Pearson, el cuál indica que tienen una correlación positiva y al tener valores mayores que cero, se puede decir que pueden estar de algún modo relacionadas linealmente.



### PAREADOS:

Las siguientes gráficas muestran los datos pareados, es decir uno en función del otro pero de cada mes, creando una recta de regresión lineal para cada uno. Esto nos puede ayudar a visualizar los cambios conforme pasa el año. Se puede observar también, como la variable CAPE toma valores mayores a los de la variable PW.



## 6 Conclusión

Como se ha mencionado en las actividades pasadas, el proceso de limpieza y preparación de datos es lo más laborioso, incluso más que su propio análisis. Es por esto que el aprender a utilizar EMACS para el trabajo de datos es de suma importancia ya que facilita el trabajo, haciendo que este sea menos pesado. Otro

aspecto importante es la automatización de los procesos, ya que permite mayor rapidez en la limpieza de los archivos.

Respecto al análisis de los datos en Python, las nuevas bibliotecas utilizadas fueron de gran ayuda para visualizar la relación que había entre los datos, facilitando de esta manera, la interpretación de la información.

## 7 Bibliografía

- CAPE y algunas nociones básicas sobre convección atmosférica. (2018). Meteoillesbalears.com. Recuperado el 5 de Marzo de 2018, de <http://www.meteoillesbalears.com/?p=623>
- Precipitable water. (2018). En.wikipedia.org. Recuperado el 5 de Marzo de 2018, de [https://en.wikipedia.org/wiki/Precipitable\\_water](https://en.wikipedia.org/wiki/Precipitable_water)
- Convective available potential energy. (2018). En.wikipedia.org. Recuperado el 5 de Marzo de 2018, de [https://en.wikipedia.org/wiki/Convective\\_available\\_potential\\_energy](https://en.wikipedia.org/wiki/Convective_available_potential_energy)

## 8 Apendice

1. ¿Cómo se te hizo esta actividad? ¿Compleja, Difícil, Sencilla? Una vez que entendí que era lo que estaba haciendo y para que lo hacía, me pareció muy sencilla.
2. ¿Qué te llamó más la atención? Las funciones de EMACS enfocadas ala limpieza de datos, ya que antes solo lo había usado como editor de texto, sin saber todo lo que podía hacer en este.
3. ¿Qué parte fue la que menos te interesó hacer? Todo me pareció interesante, ya que fue como un resumen de las actividades pasadas pero una manera más eficiente.
4. ¿Cómo mejorarías esta actividad? ¿Qué le faltó? ¿Qué sobró? Me pareció una practica muy completa, que me sirvió para reforzar lo visto en actividades pasadas.
5. ¿Hasta este punto, que te parece el uso de Jupyter para programar en Python? Me parece un ambiente muy amigable y muy util para el análisis de los datos, sobre todo por las bibliotecas con las que se trabajó.