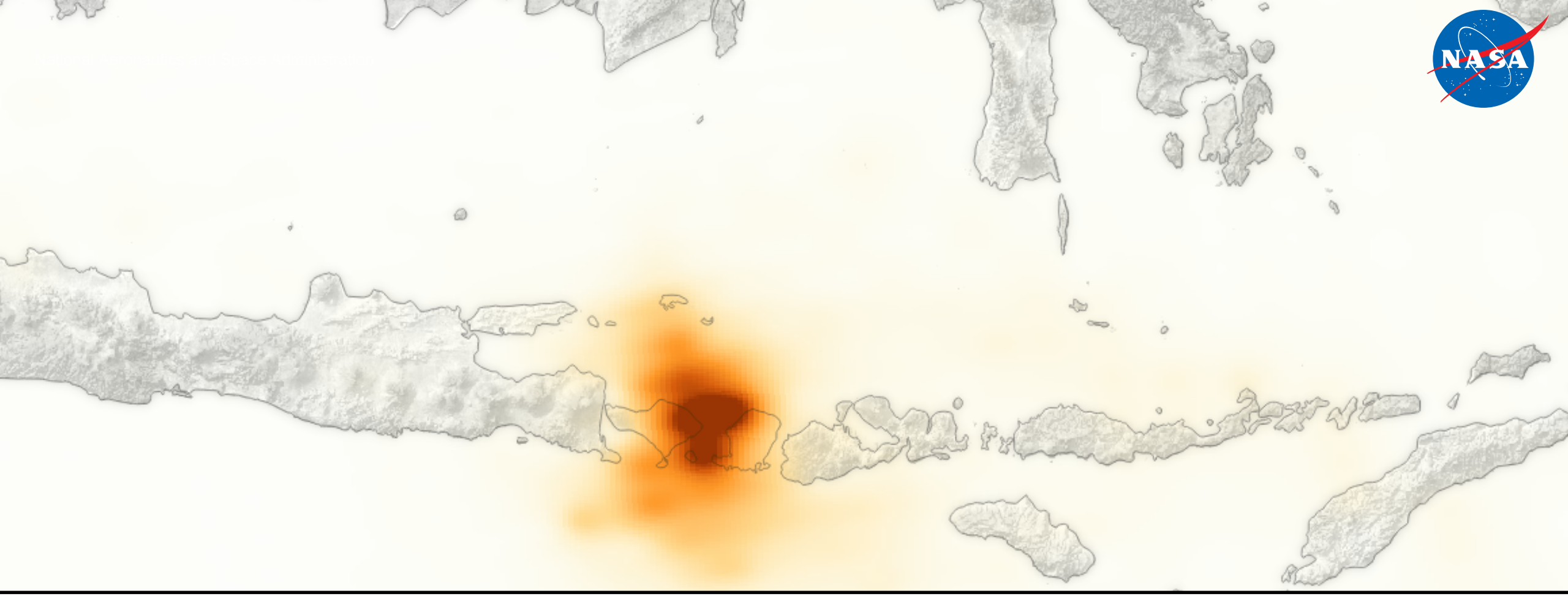
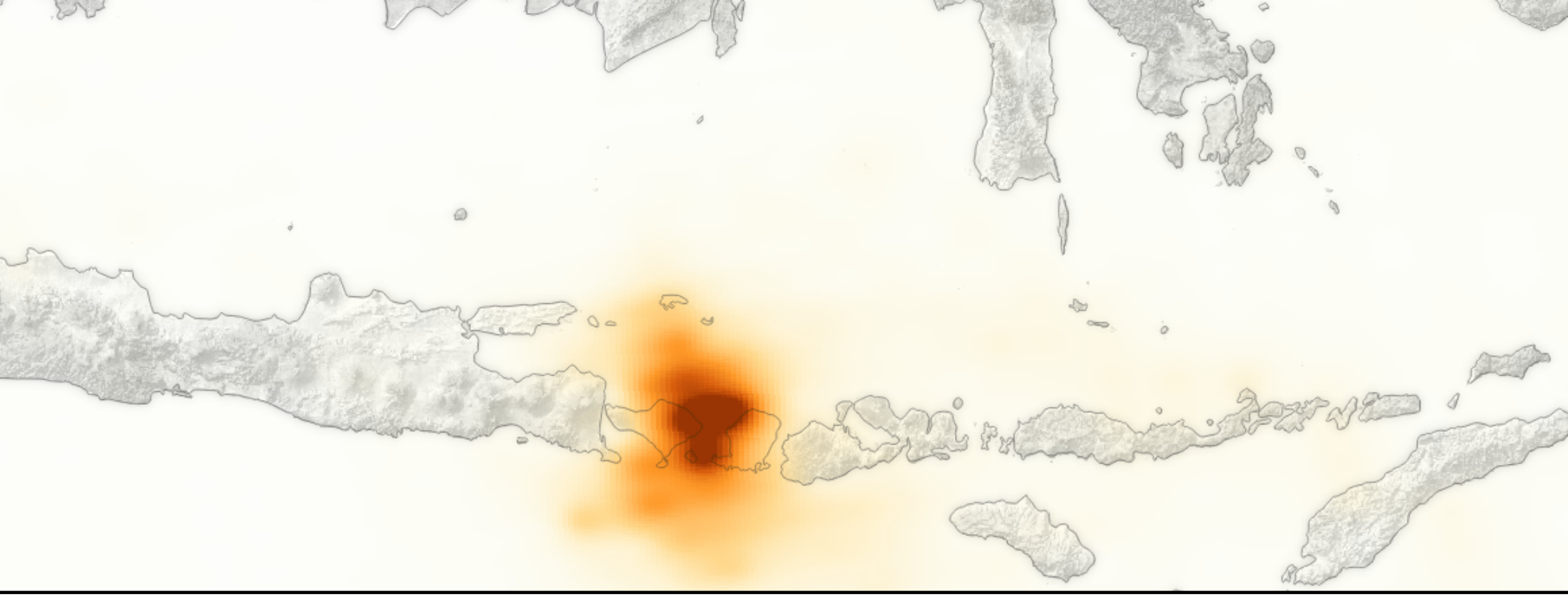


Lea, Mapee y Extraiga Datos de Aerosoles de MODIS Usando Scripts de Python

Herramientas para el Análisis de Conjuntos de Datos Satelitales de la Calidad del Aire de Alta Resolución

Pawan Gupta y Melanie Follette-Cook, 17 al 22 de enero de 2018



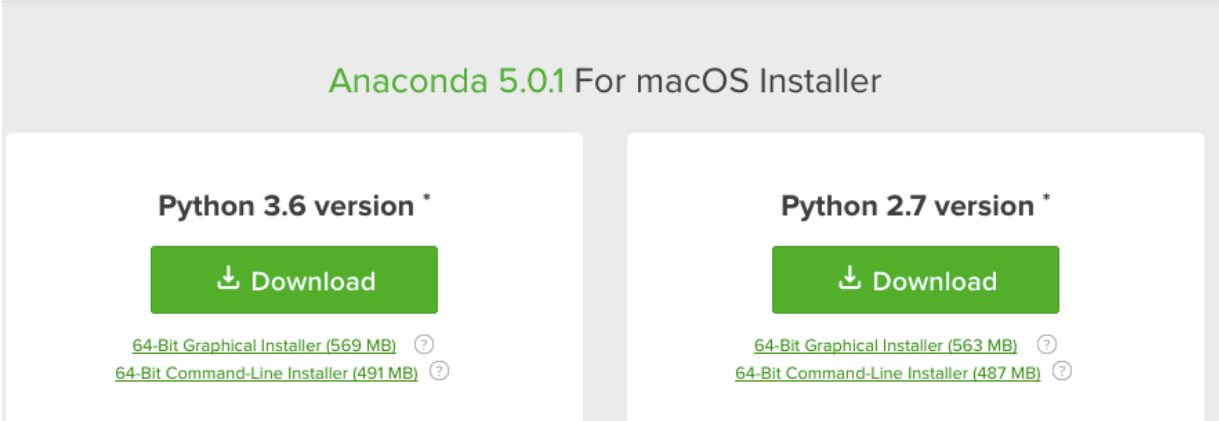
Requisitos para su Computadora y Python

Sistema Operativo

- Tanto la Sesión 2 como la Sesión 3 cubrirán scripts de Python para realizar tareas similares.
- La Sesión 2 se presentará usando un sistema operativo Mac (Apple).
- Toda la estructura de los directorios y el diseño de Python serán según el sistema operativo para Mac.
- La Sesión 3 se presentará usando una máquina Windows.
- El mostrar usando ambos sistemas operativos permitirá a los usuarios experimentar el diseño de cada uno.
- Anticipamos que los códigos se ejecuten exitosamente en los sistemas operativos de Mac, Windows y Linux.

Requisitos Computacionales

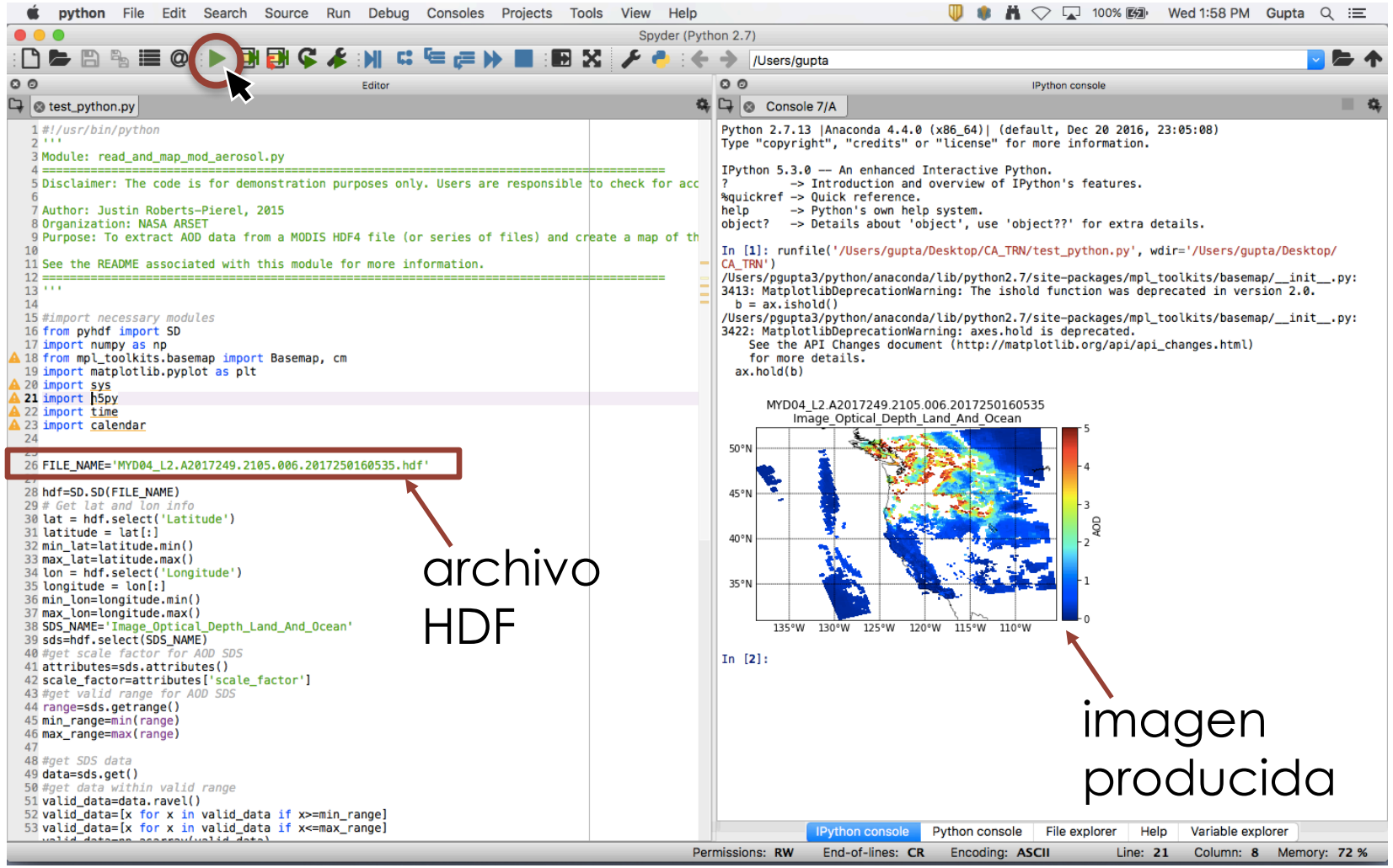
- Instalar Python 2.7 usando Anaconda
- Instalar todos los paquetes (packages) de Python requeridos
  - Lista de paquetes (derecha)
- Probar la instalación de Python y los paquetes usando el siguiente código de prueba para Python
  - test_python.py
- Descargar los datos de MODIS Data y códigos de Python de la página en línea de la capacitación
  - https://arset.gsfc.nasa.gov/airquality/webinars/2018-hiresdatasets
- Para más detalles sobre el código, visite: https://arset.gsfc.nasa.gov/airquality/python-scripts-aerosol-data-sets-merra-modis-and-omi

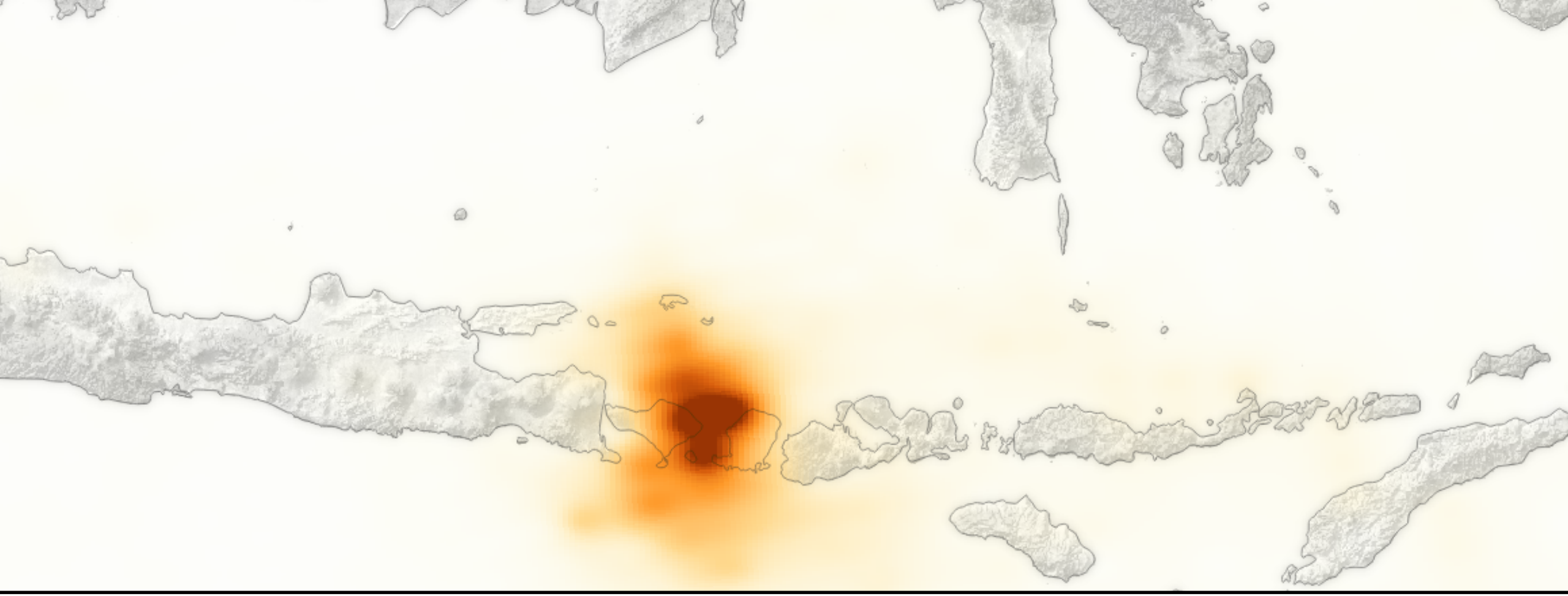


- Lista de paquetes de Python:
  - pyhdf
  - numpy
  - sys
  - mpl_toolkits.basemap
  - matplotlib
  - linearSegmentedColormap
  - h5py
  - time
  - calendar

Prueba de Python

- Abra el editor “spyder” dentro de Anaconda
- Abra **test_python.py**
- Asegúrese que el directorio tenga el código Python y el archivo HDF
- Abra la consola **ipython** en el spider
- Ejecute el código usando la **flecha verde** en la parte superior
- Debe producirse una imagen como la que se muestra aquí

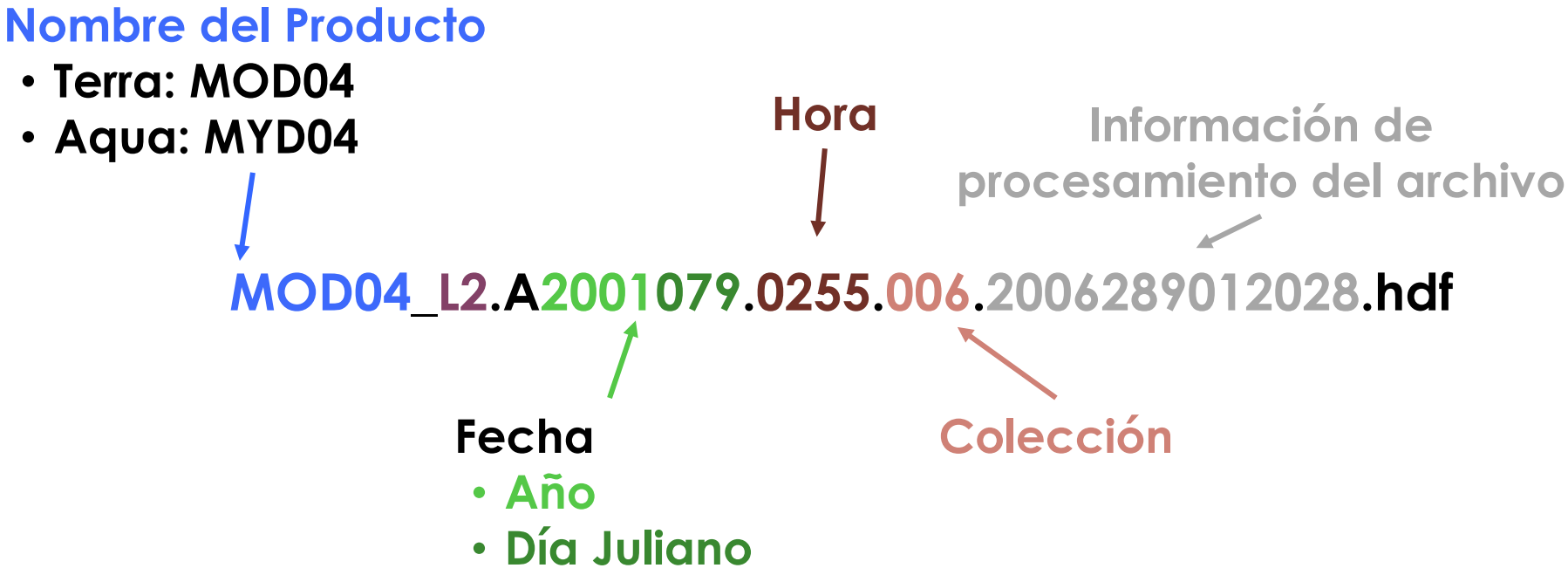




Conozca Sus Datos

Cómo Entender el Nombre de un Archivo de MODIS

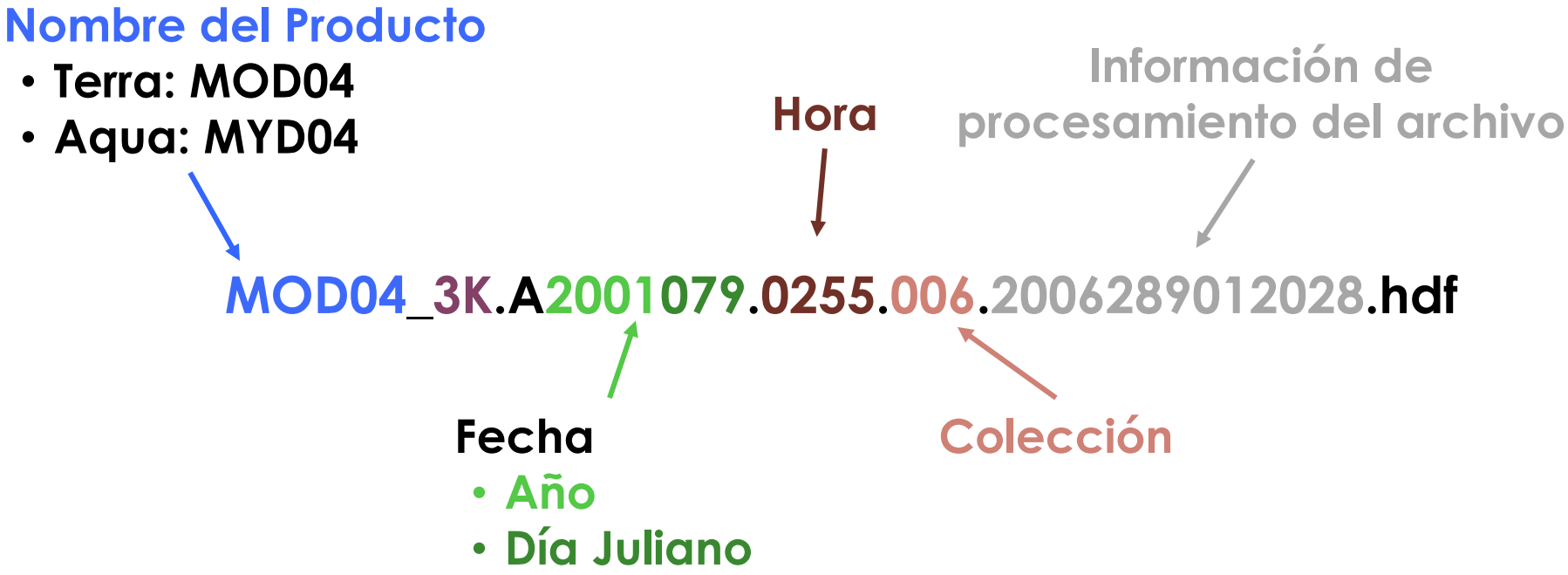
Producto de Aerosoles, Nivel 2, 10 km



Se puede usar HDFLook, Panoply, IDL, Python, Fortran, MatLab y otros para leer los datos

Cómo Entender el Nombre de un Archivo de MODIS

Producto de Aerosoles, Nivel 2, 3 km



Se puede usar HDFLook, Panoply, IDL, Python, Fortran, MatLab y otros para leer los datos

Parámetros de Aerosoles de MODIS (SDS)

- Optical_Depth_Land_and_Ocean
  - Se recupera usando el algoritmo de objetivo oscuro(Dark Target Algorithm)
  - Sólo datos de alta calidad
    - QA (garantía de calidad) = 3
    - QA = 1, 2, 3
  - 10 km y 3km
- Dark_Target_Deep_Blue_Optical_Depth_550_Combined
  - Producto combinado de los algoritmos Deep Blue (azul profundo) y Dark Target (objetivo oscuro)
  - Sólo de 10 km
- Quality_Assurance_Land
  - Indicación de calidad asociada con producto DD

La Garantía de Calidad (Quality Assurance) es Sumamente Importante

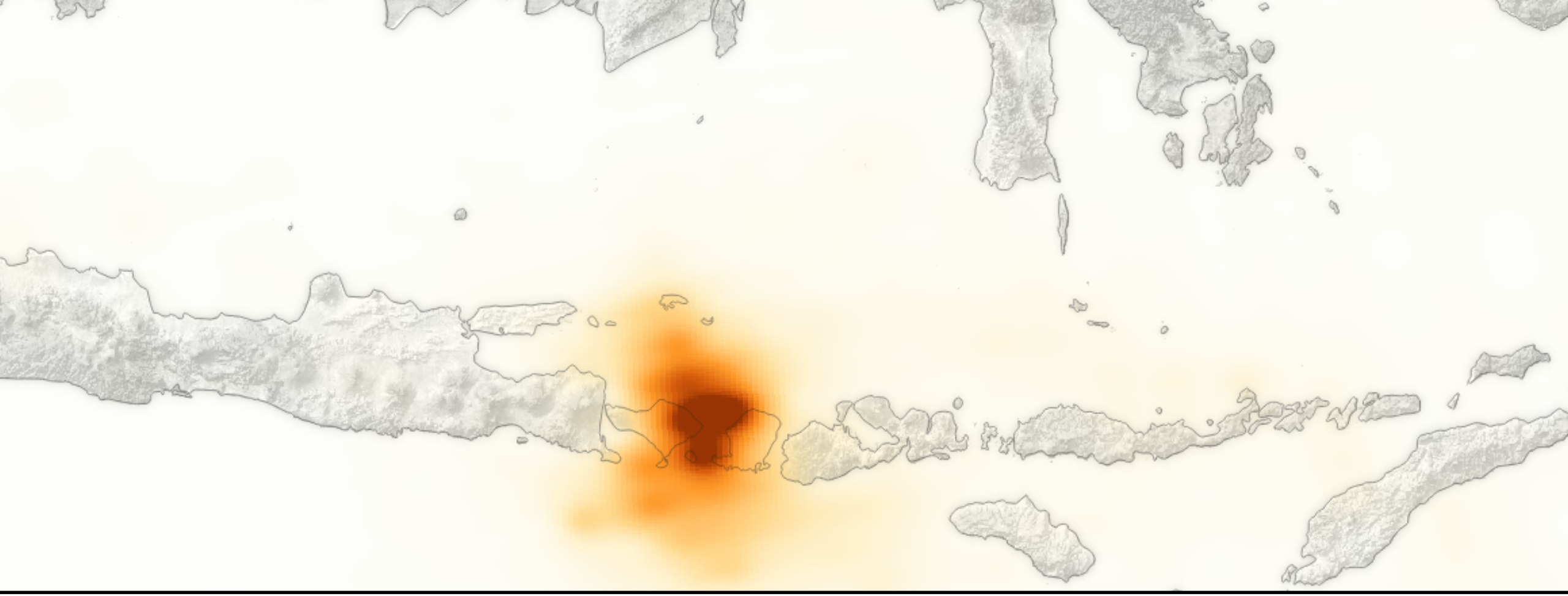
La Garantía de Calidad (QA) indica la confianza en la calidad de la imagen recuperada

Quality_Assurance_Ocean

- Escala de 0 a 3
- Se Recomienda QA por encima de 1,2,3 para Océano
- Factores:
  - Número de pixeles
  - Adecuación de errores
  - **Proximidad al destello**

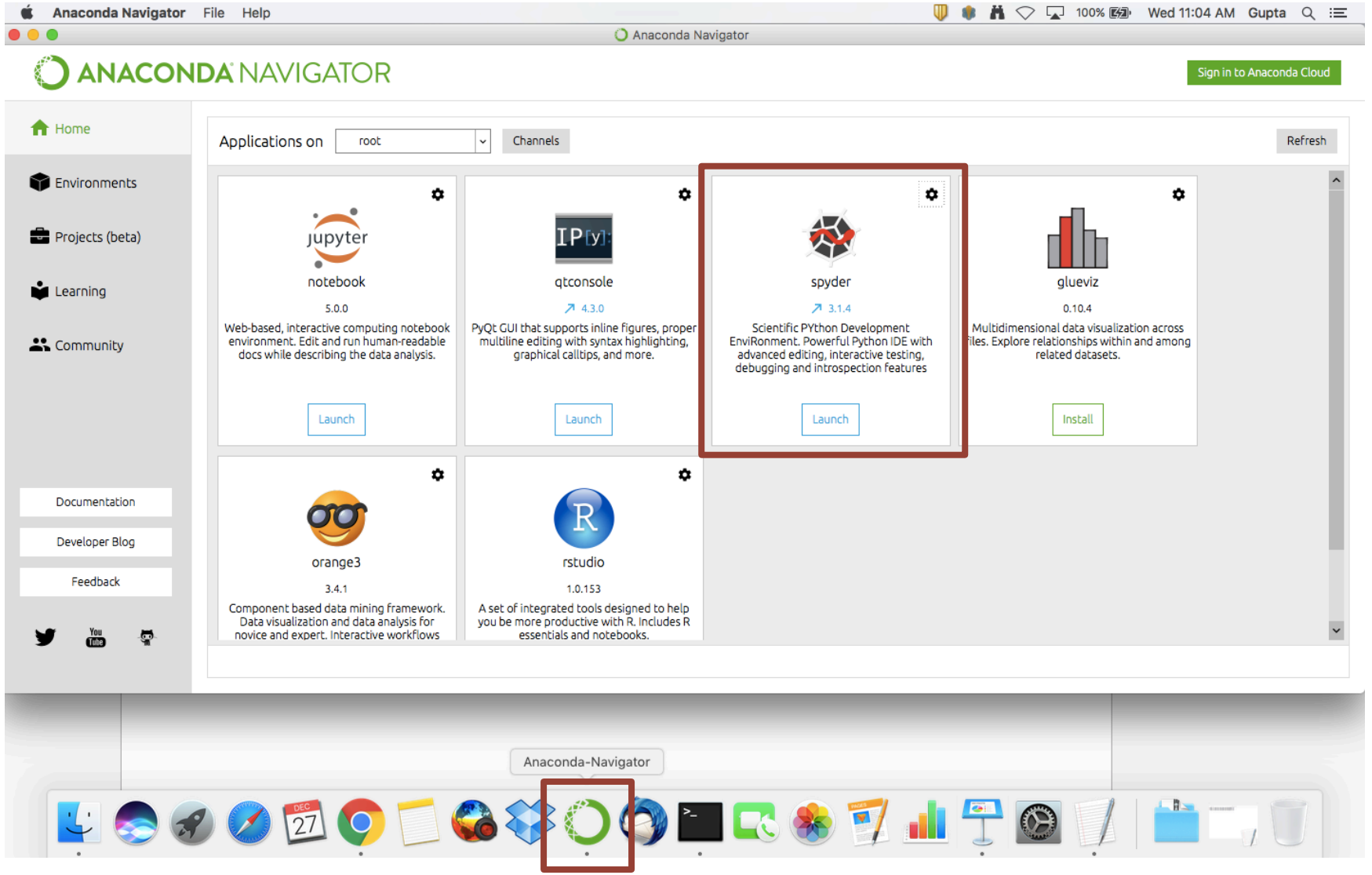
Quality_Assurance_Land

- Escala de 0 a 3
- Se Recomienda QA de 3 para Tierra
- Factores:
  - Número de pixeles
  - Adecuación de errores
  - **Reflectancia superficial**

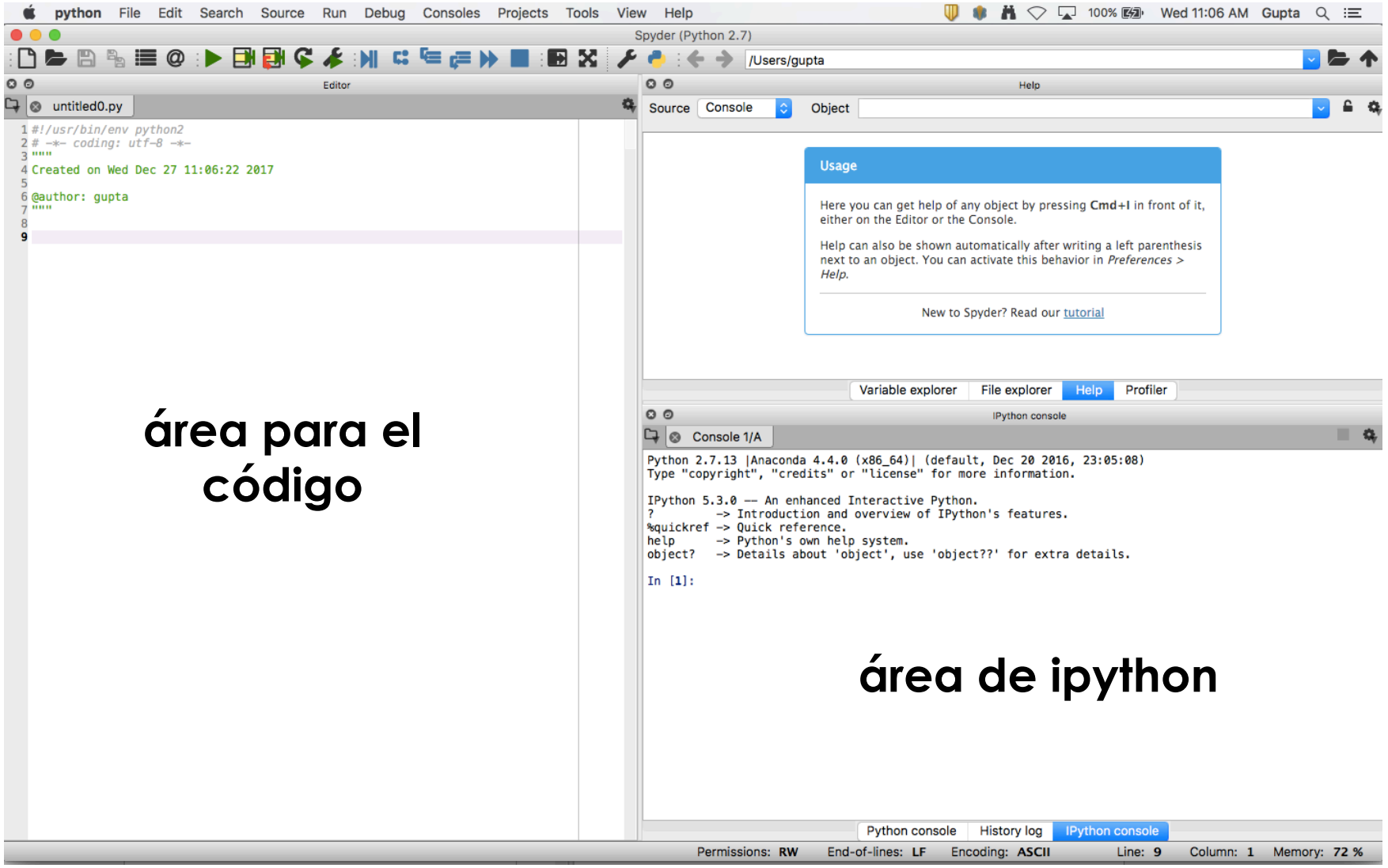


Alistándose con Python

Anaconda y el Editor Spyder



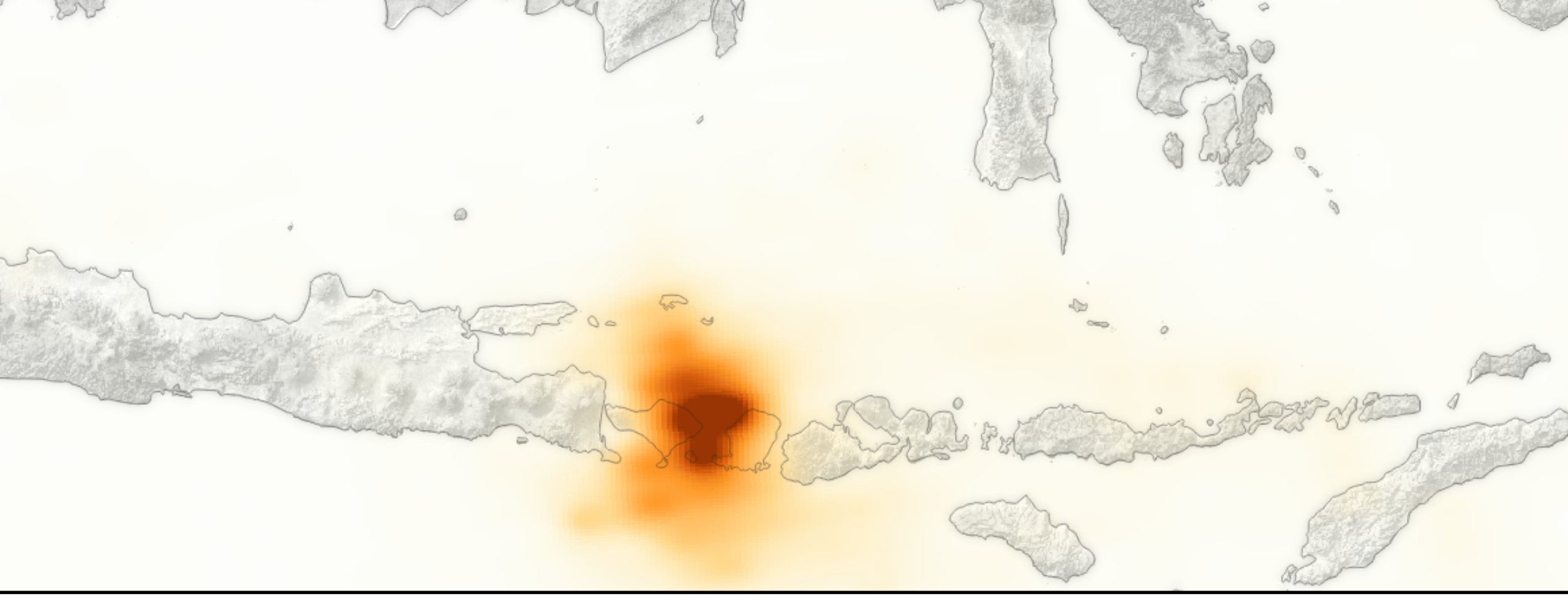
Vista de Spyder



Vista del Directorio Actual y Lista de Archivos

- Cree una lista de archivos HDF 'fileList.txt'
- El directorio debe tener
  - Todos los códigos Python
  - Todos los archivos HDF
  - Una lista de los archivos HDF titulado 'fileList.txt'

```
gs614-guptaml:CA_TRN gupta$ vi fileList.txt
gs614-guptaml:CA_TRN gupta$ ls
MYD04_3K.A2017232.2200.006.2017233154505.png
MYD04_3K.A2017232.2200.006.2017233154505.txt
MYD04_L2.A2017232.1520.006.2017233154749.png
MYD04_L2.A2017232.2200.006.2017233154546.png
MYD04_L2.A2017249.1925.006.2017250160408.hdf
MYD04_L2.A2017249.1925.006.2017250160408.txt
MYD04_L2.A2017249.1930.006.2017250160703.hdf
MYD04_L2.A2017249.2105.006.2017250160535.hdf
fileList.txt
py1
read_and_map_mod_aerosol.py
read_aod_and_calculate_pm25.py
read_mod_aerosol_and_dump_ascii.py
read_mod_aerosol_and_list_sds.py
read_mod_aerosol_at_a_location.py
readme
gs614-guptaml:CA_TRN gupta$ ls *.hdf
MYD04_L2.A2017249.1925.006.2017250160408.hdf
MYD04_L2.A2017249.1930.006.2017250160703.hdf
MYD04_L2.A2017249.2105.006.2017250160535.hdf
gs614-guptaml:CA_TRN gupta$ ls *.hdf >fileList.txt
gs614-guptaml:CA_TRN gupta$ more fileList.txt
MYD04_L2.A2017249.1925.006.2017250160408.hdf
MYD04_L2.A2017249.1930.006.2017250160703.hdf
MYD04_L2.A2017249.2105.006.2017250160535.hdf
gs614-guptaml:CA_TRN gupta$
```

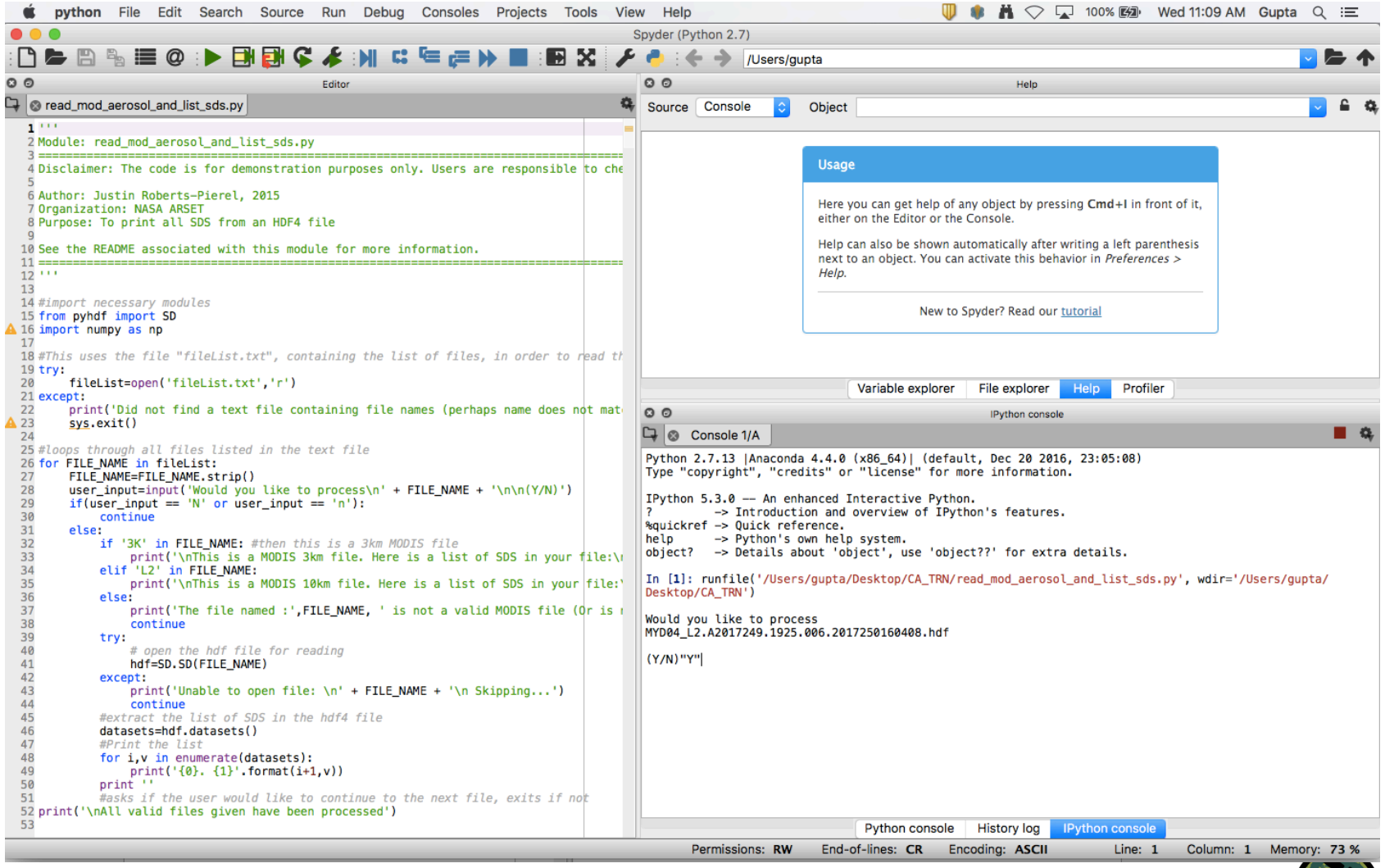



Lea un Archivo de Aerosoles de MODIS (HDF) e Imprima la Lista de SDS

Imprima Conjuntos de Datos Científicos (Scientific Data Sets o SDSs)

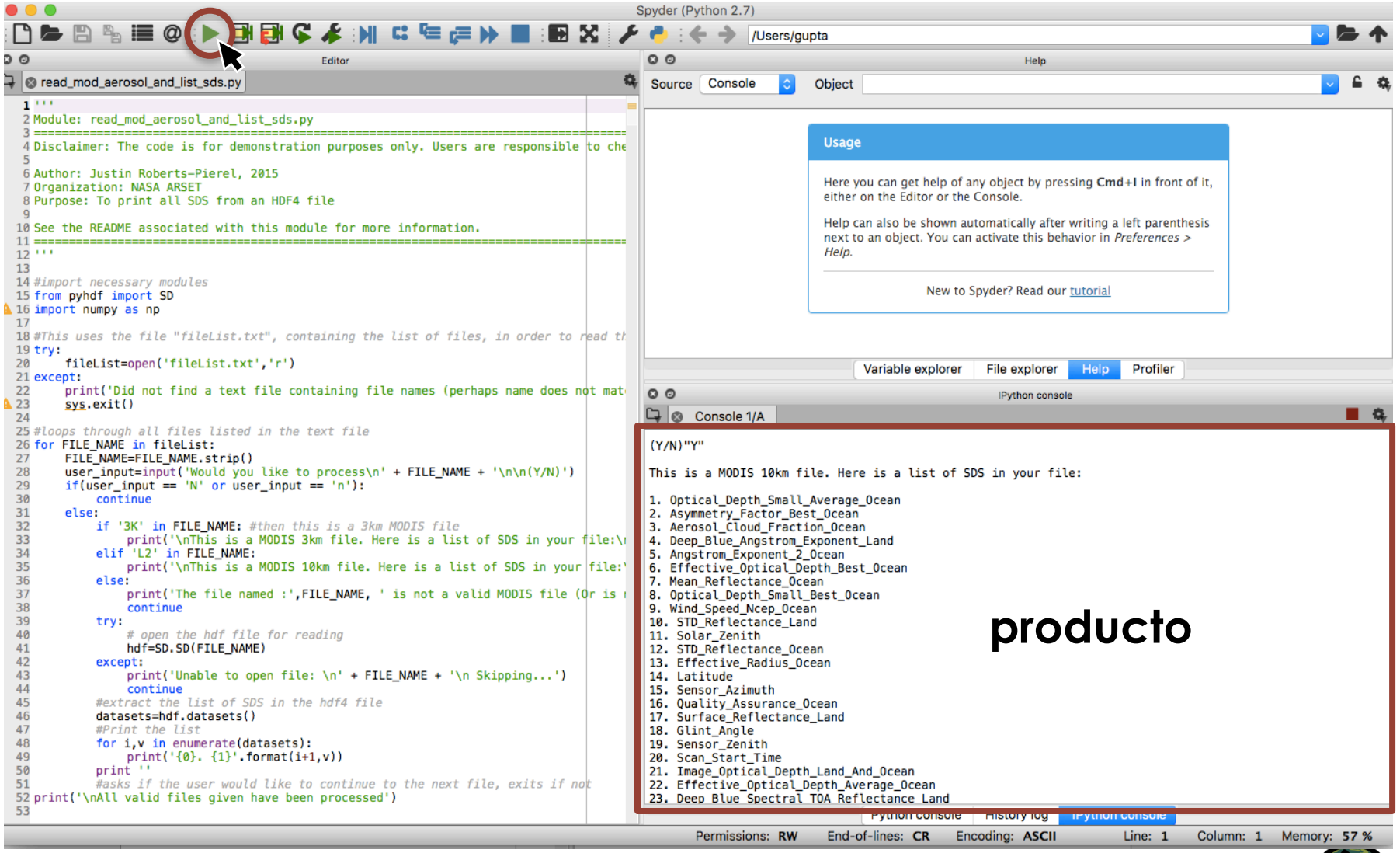
read_mod_aerosol_and_list_sds.py

- **Propósito**: leer un archivo de aerosoles de MODIS de nivel 2 en formato HDF e imprimir todos los conjuntos de datos científicos (**Scientific Data Sets** o SDS)
- El código sirve para productos de 10 km y de 3 km



Ejecución y Producto

- Haga clic en la flecha verde para ejecutar el código
- El código procesará todos los archivos en la lista **fileList.txt** uno por uno
- Siga las instrucciones en la terminal de **ipython** (es decir, ingrese 'Y' o 'N' cuando éste se lo pida y teclee Enter)



Editando el Código

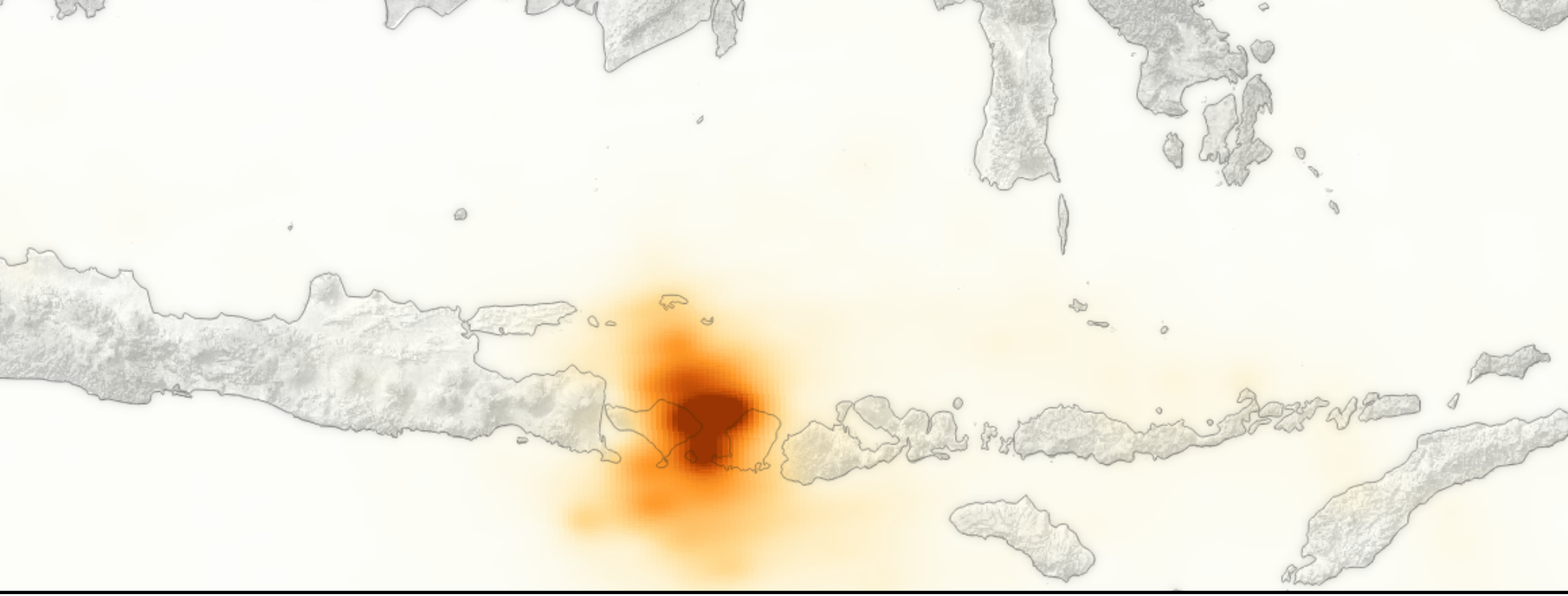
```
Module: read_mod_aerosol_and_list_sds.py
================================================================================
Disclaimer: The code is for demonstration purposes only. Users are responsible to check for accuracy and revise to fit their obj

Author: Justin Roberts-Pierel, 2015
Organization: NASA ARSET
Purpose: To print all SDS from an HDF4 file

See the README associated with this module for more information.
================================================================================
'''

#import necessary modules
from pyhdf import SD
import numpy as np

#This uses the file "fileList.txt", containing the list of files, in order to read the files
try:
    fileList=open('fileList.txt','r')
except:
    print('Did not find a text file containing file names (perhaps name does not match)')
    sys.exit()

#loops through all files listed in the text file
for FILE_NAME in fileList:
    FILE_NAME=FILE_NAME.strip()
    user_input=input('Would you like to process\n' + FILE_NAME + '\n\n(Y/N)')
    if(user_input == 'N' or user_input == 'n'):
        continue
    else:
        if '3K' in FILE_NAME: #then this is a 3km MODIS file
            print('\nThis is a MODIS 3km file. Here is a list of SDS in your file:\n')
        elif 'L2' in FILE_NAME:
            print('\nThis is a MODIS 10km file. Here is a list of SDS in your file:\n')
        else:
            print('The file named :',FILE_NAME, ' is not a valid MODIS file (Or is named incorrectly). \n')
            continue
        try:
            # open the hdf file for reading
            hdf=SD.SD(FILE_NAME)
        except:
            print('Unable to open file: \n' + FILE_NAME + '\n Skipping...')
            continue
        #extract the list of SDS in the hdf4 file
        datasets=hdf.datasets()
        #Print the list
        for i,v in enumerate(datasets):
            print('{0}. {1}'.format(i+1,v))
        print ''
        #asks if the user would like to continue to the next file, exits if not
print('\nAll valid files given have been processed')
```

cambie el nombre de fileList.txt por cualquiera que le guste

Este código ha sido probado para archivos de aerosoles de MODIS de Nivel 2 de 3 km y 10 km

Aplicaciones

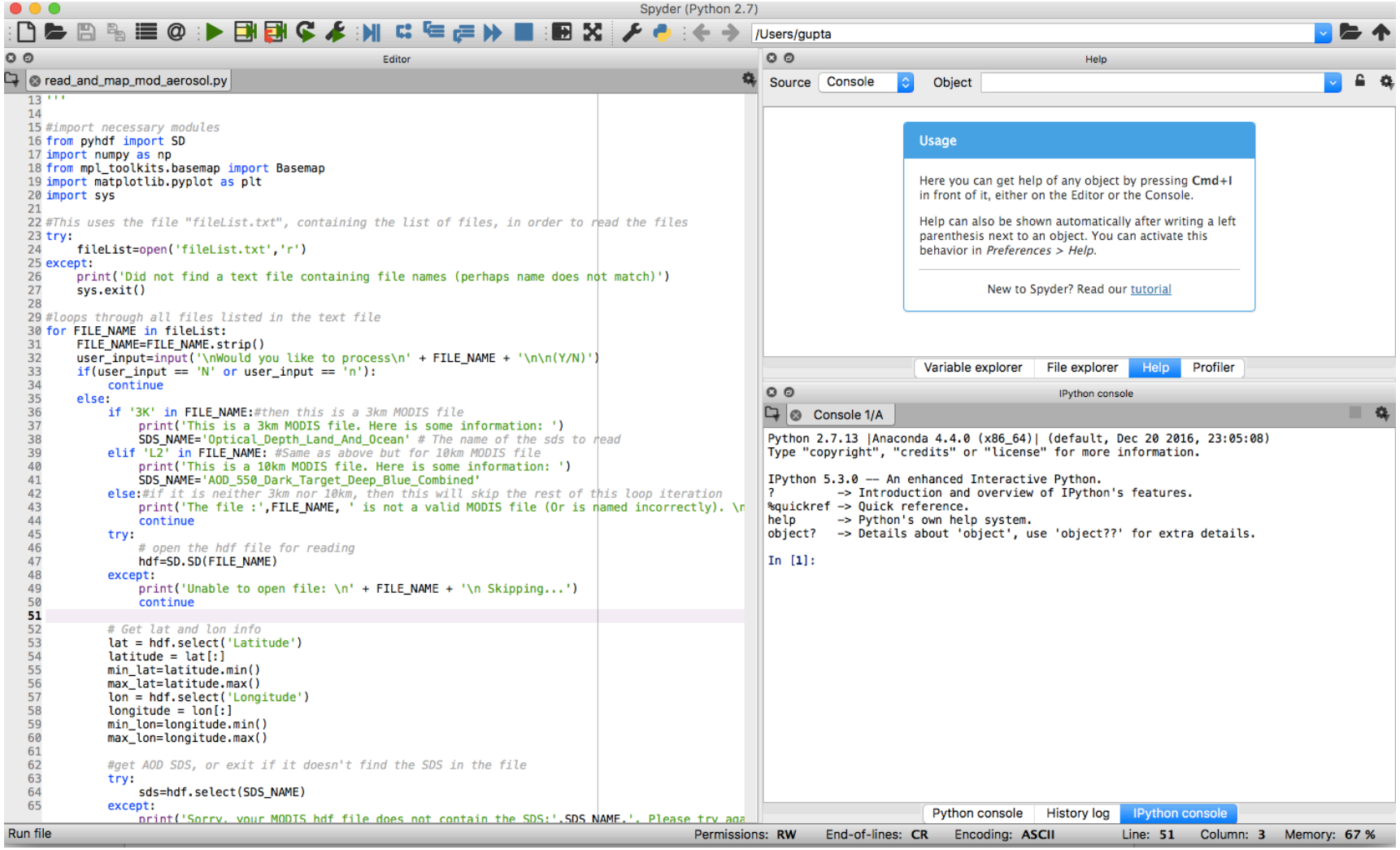
- Los datos de aerosoles de MODIS de Nivel 2 están guardados en archivos HDF
- Cada archivo HDF contiene varios parámetros geofísicos
- Se requieren códigos y herramientas especiales para abrir archivos HDF
- Este código ayuda a los usuarios a ver los nombres de los SDSs dentro de un archivo HDF para análisis adicional



Mapee el Espesor Óptico de Aerosoles

Trace y Guarde un Mapa del AOD de MODIS

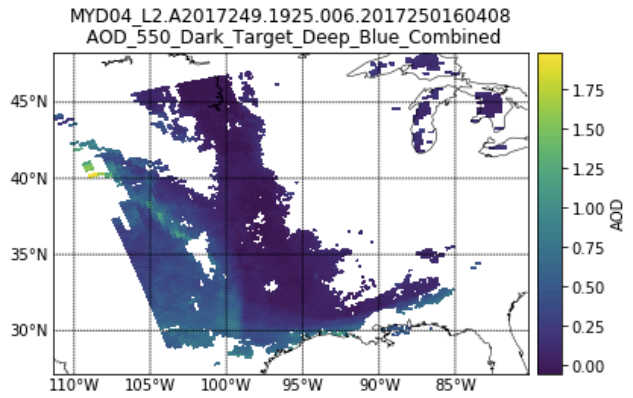
read_and_map_mod_aerosol.py



Ejecución y Producto Output

Estadísticas del AOD

```
In [1]: runfile('/Users/gupta/Desktop/CA_TRN/read_and_map_mod_aerosol.py', wdir='/Users/gupta/Desktop/
CA_TRN')


Would you like to process
MYD04_L2.A2017249.1925.006.2017250160408.hdf

(Y/N)"Y"
This is a 10km MODIS file. Here is some information:
('\nThe valid range of values is: ', -0.1, ' to ', 5.0, '\nThe average is: ', 0.178, '\nThe standard
deviation is: ', 0.23)
('The range of latitude in this file is: ', 27.187273, ' to ', 48.299458, 'degrees \nThe range of
longitude in this file is: ', -111.39777, ' to ', -80.255447, ' degrees')

Would you like to create a map of this data? Please enter Y or N
"Y"
/Users/pgupta3/python/anaconda/lib/python2.7/site-packages/mpl_toolkits/basemap/__init__.py:3413:
MatplotlibDeprecationWarning: The ishold function was deprecated in version 2.0.
  b = ax.ishold()
/Users/pgupta3/python/anaconda/lib/python2.7/site-packages/mpl_toolkits/basemap/__init__.py:3422:
MatplotlibDeprecationWarning: axes.hold is deprecated.
    See the API Changes document (http://matplotlib.org/api/api_changes.html)
    for more details.
  ax.hold(b)
```



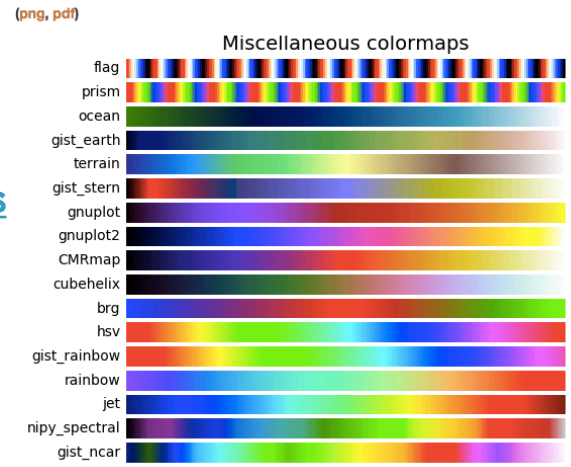
Mapa del AOD producido

```
Would you like to save this map? Please enter Y or N
```

Editando el Código

Cambie de Escala Cromática

```
data=data.astype(float)
data[data == fv] = np.nan
#create the map
data = np.ma.masked_array(data, np.isnan(data))
m = Basemap(projection='cyl', resolution='l', llcrnrlat=min_lat, urcrnrlat = max_lat, llcrnrlon=min_lon,
m.drawcoastlines(linewidth=0.5)
m.drawparallels(np.arange(-90., 120., 5.), labels=[1, 0, 0, 0])
m.drawmeridians(np.arange(-180., 181., 5.), labels=[0, 0, 0, 1])
x, y = m(longitude, latitude)
m.pcolormesh(x, y, data*scale_factor, cmap=plt.cm.jet)
plt.autoscale()
#create colorbar
cb = m.colorbar()
#label colorboar
cb.set_label('AOD')
```



https://matplotlib.org/examples/color/colormaps_reference.html

Cambie de SDS

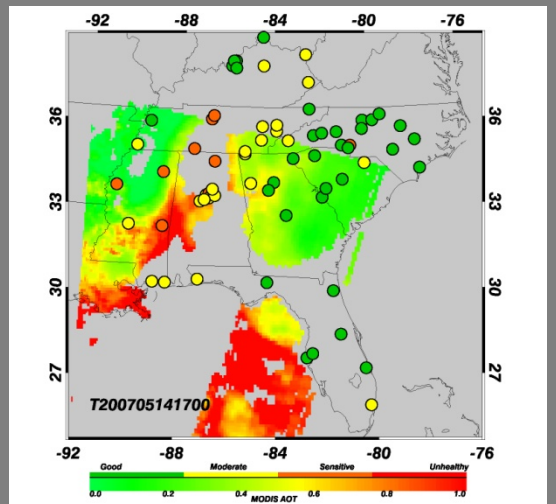
```
#loops through all files listed in the text file
for FILE_NAME in fileList:
    FILE_NAME=FILE_NAME.strip()
    user_input=input('\nWould you like to process\n' + FILE_NAME + '\n\n(Y/N)')
    if(user_input == 'N' or user_input == 'n'):
        continue
    else:
        if '3K' in FILE_NAME:#then this is a 3km MODIS file
            print('This is a 3km MODIS file. Here is some information: ')
            SDS_NAME='Optical_Depth_Land_And_Ocean' # The name of the sds to read
        elif 'L2' in FILE_NAME: #Same as above but for 10km MODIS file
            print('This is a 10km MODIS file. Here is some information: ')
            SDS_NAME='AOD_550_Dark_Target_Deep_Blue_Combined'
        else:#if it is neither 3km nor 10km, then this will skip the rest of this loop iteration
            print('The file :',FILE_NAME, ' is not a valid MODIS file (Or is named incorrectly). \n')
            continue
        try:
            # open the hdf file for reading
            hdf=SD.SD(FILE_NAME)
        except:
            print('Unable to open file: \n' + FILE_NAME + '\n Skipping...')
            continue
```

Aplicaciones

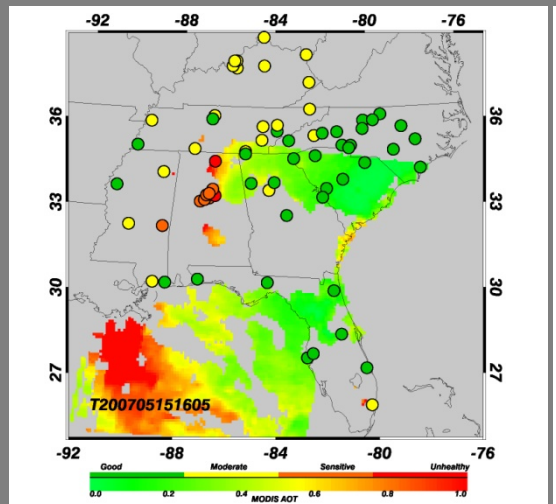
- Éste es un ejemplar de un código para leer y mapear los datos de aerosoles de MODIS de Nivel 2
- Se puede modificar el código para diferentes necesidades cartográficas
- Los usuarios pueden crear mapas diarios del AOD sobre ciertas regiones y comenzar a analizar cambios a través del tiempo
- Los mapas del AOD también pueden ayudar a identificar regiones con niveles de contaminación elevados

Ejemplo:
Valores elevados del AOD por causa del humo concuerdan estrechamente con monitores en la superficie (círculos).

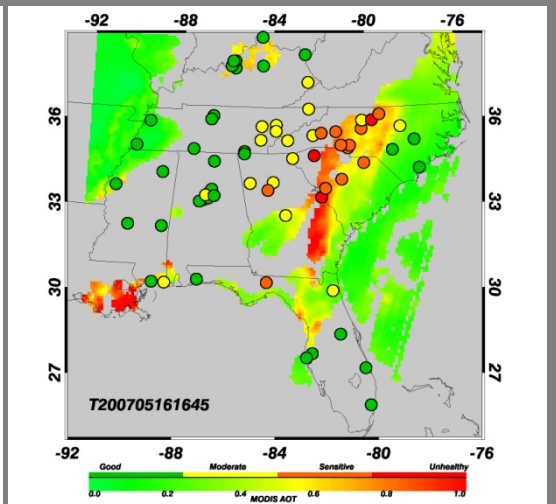
14 de mayo de 2007

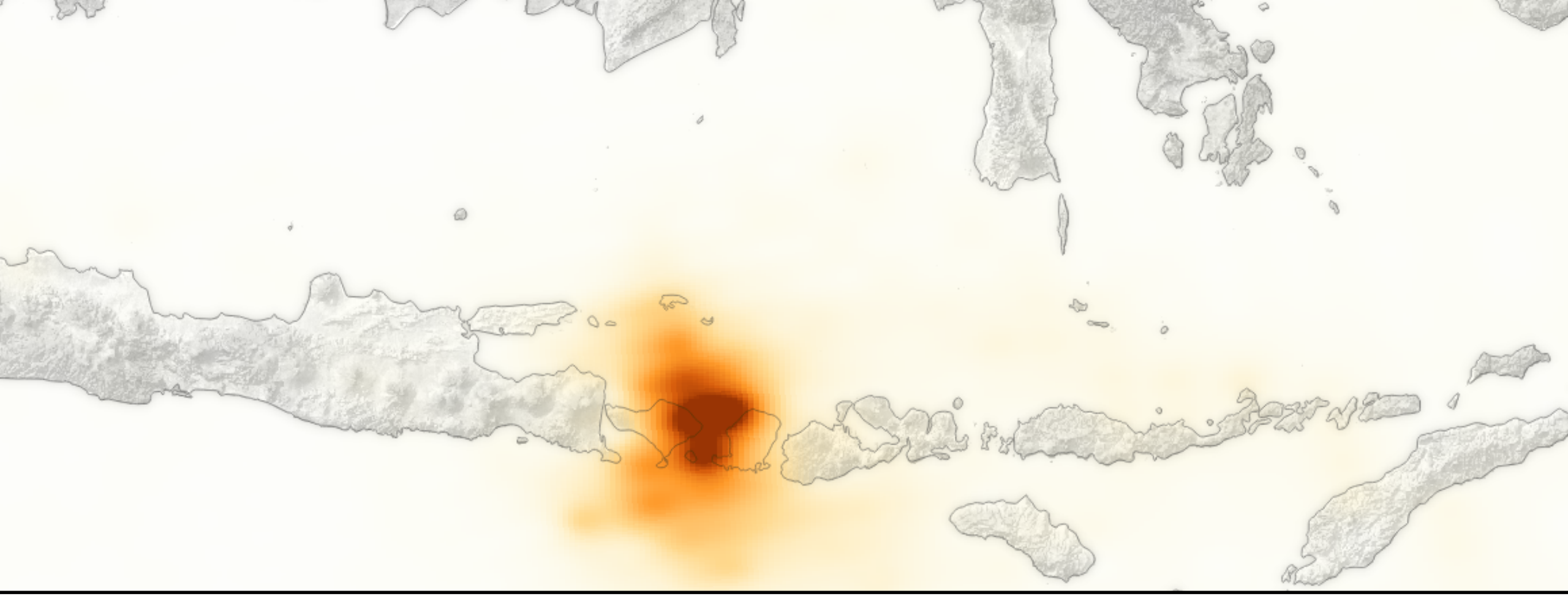


15 de mayo de 2007



16 de mayo de 2007



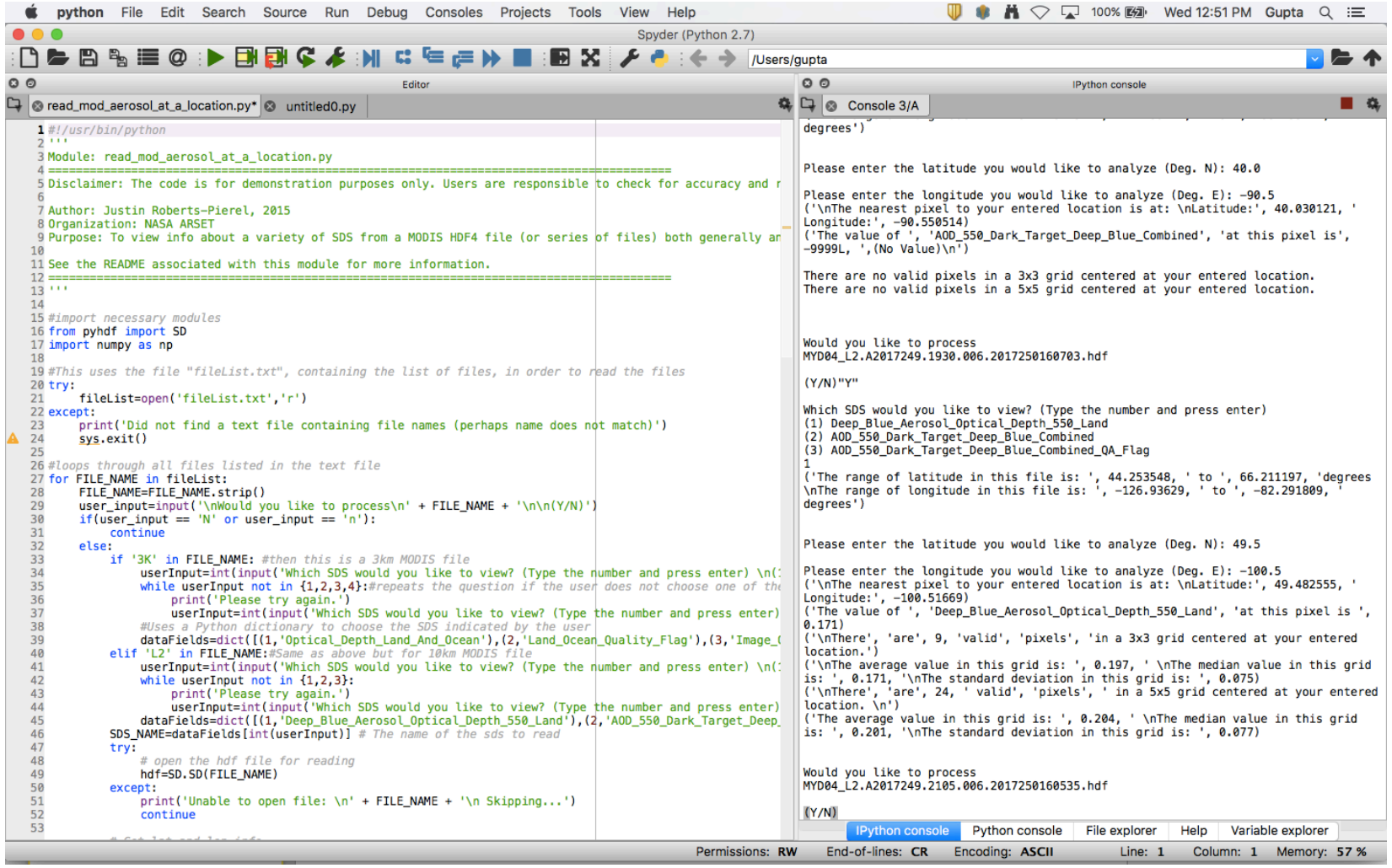


Extraiga el AOD de una Estación Superficial

Extraiga los Valores del AOD en un Lugar Indicado

read_mod_aerosol_at_a_location.py

- **Propósito:** leer un archivo de datos de MODIS de nivel 2 en formato HDF y extraer los valores del AOD en una ubicación en la superficie indicada
- El código sirve para productos de 10 km y 3 km



Ejecución y Producto

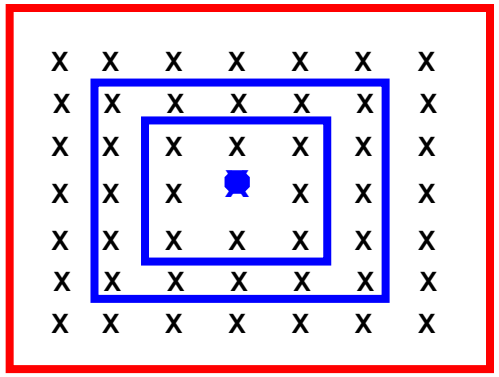
Teclee "Y" para procesar el archivo, "N" para saltar

Seleccione SDS

Lat. y Lon. de la estación

```
Would you like to process
MYD04_L2.A2017249.1930.006.2017250160703.hdf

(Y/N)"Y"

Which SDS would you like to view? (Type the number and press enter)
(1) Deep_Blue_Aerosol_Optical_Depth_550_Land
(2) AOD_550_Dark_Target_Deep_Blue_Combined
(3) AOD_550_Dark_Target_Deep_Blue_Combined_QA_Flag
1
('The range of latitude in this file is: ', 44.253548, ' to ', 66.211197, 'degrees
\nThe range of longitude in this file is: ', -126.93629, ' to ', -82.291809, '
degrees')


Please enter the latitude you would like to analyze (Deg. N): 49.5

Please enter the longitude you would like to analyze (Deg. E): -100.5
('\nThe nearest pixel to your entered location is at: \nLatitude:', 49.482555, '
Longitude:', -100.51669)
('The value of ', 'Deep_Blue_Aerosol_Optical_Depth_550_Land', 'at this pixel is ',
0.171)
('\nThere', 'are', 9, 'valid', 'pixels', 'in a 3x3 grid centered at your entered
location.')
('\nThe average value in this grid is: ', 0.197, ' \nThe median value in this grid
is: ', 0.171, '\nThe standard deviation in this grid is: ', 0.075)
('\nThere', 'are', 24, ' valid', 'pixels', ' in a 5x5 grid centered at your entered
location. \n')
('The average value in this grid is: ', 0.204, ' \nThe median value in this grid
is: ', 0.201, '\nThe standard deviation in this grid is: ', 0.077)


Would you like to process
MYD04_L2.A2017249.2105.006.2017250160535.hdf

(Y/N)
```

Productos



Editando el Código – Cambie de SDS

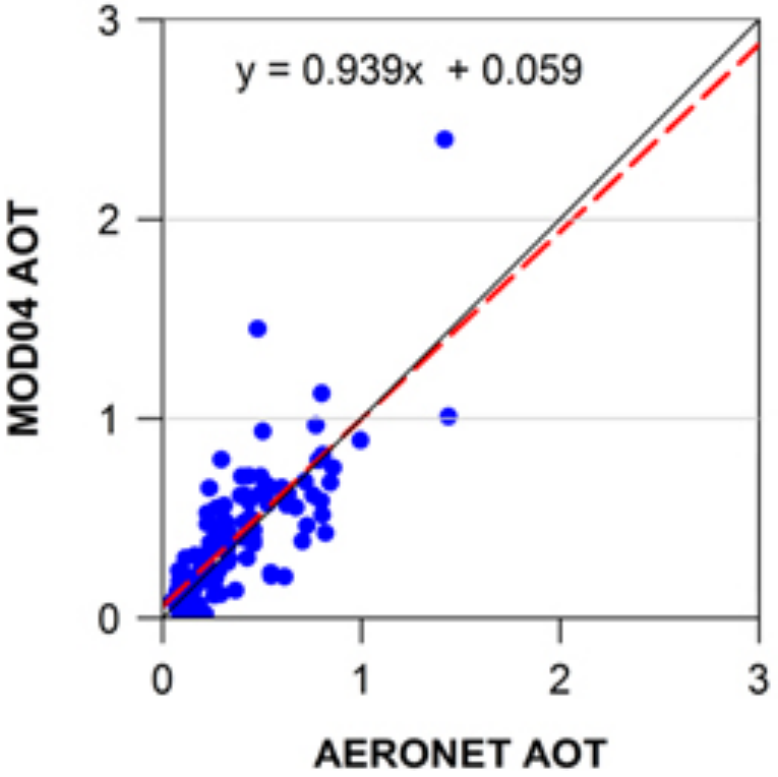
```
s through all files listed in the text file
ILE_NAME in fileList:
FILE_NAME=FILE_NAME.strip()
user_input=input('\nWould you like to process\n' + FILE_NAME + '\n\n(Y/N)')
if(user_input == 'N' or user_input == 'n'):
    continue
else:
    if '3K' in FILE_NAME: #then this is a 3km MODIS file
        userInput=int(input('Which SDS would you like to view? (Type the number and press enter) \n(1) Optical_Depth_Land_And_Ocean \n(2
        while userInput not in {1,2,3,4}:#repeats the question if the user does not choose one of the options
            print('Please try again.')
            userInput=int(input('Which SDS would you like to view? (Type the number and press enter) \n(1) Optical_Depth_Land_And_Ocean
        #Uses a Python dictionary to choose the SDS indicated by the user
        dataFields=dict([(1,'Optical_Depth_Land_And_Ocean'),(2,'Land_Ocean_Quality_Flag'),(3,'Image_Optical_Depth_Land_And_Ocean'),(4,'La
    elif 'L2' in FILE_NAME:#Same as above but for 10km MODIS file
        userInput=int(input('Which SDS would you like to view? (Type the number and press enter) \n(1) Deep_Blue_Aerosol_Optical_Depth_55
        while userInput not in {1,2,3}:
            print('Please try again.')
            userInput=int(input('Which SDS would you like to view? (Type the number and press enter) \n(1) Deep_Blue_Aerosol_Optical_Dep
        dataFields=dict([(1,'Deep_Blue_Aerosol_Optical_Depth_550_Land'),(2,'AOD_550_Dark_Target_Deep_Blue_Combined'),(3,'AOD_550_Dark_Tar
    SDS_NAME=dataFields[int(userInput)] # The name of the sds to read
    try:
        # open the hdf file for reading
        hdf=SD.SD(FILE_NAME)
    except:
        print('Unable to open file: \n' + FILE_NAME + '\n Skipping...')
        continue
```


Editando el Código – Cambie los Cálculos del AOD

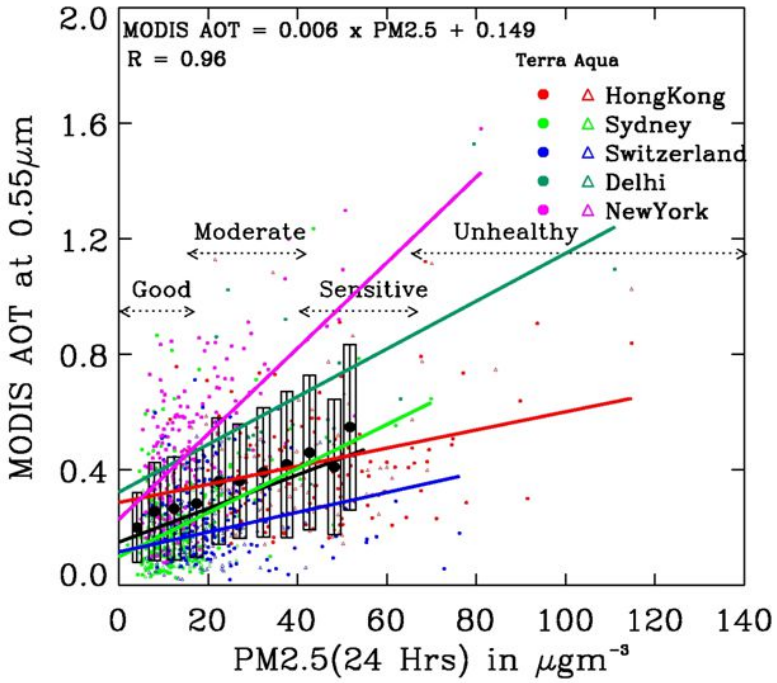
```
#calculates mean, median, stdev in a 3x3 grid around nearest point to entered location
if x < 1:
     x+=1
if x > data.shape[0]-2:
     x-=2
if y < 1:
     y+=1
if y > data.shape[1]-2:
     y-=2
three_by_three=data[x-1:x+2,y-1:y+2]
three_by_three=three_by_three.astype(float)
three_by_three[three_by_three==float(fillvalue)]=np.nan
nnan=np.count_nonzero(~np.isnan(three_by_three))
if nnan == 0:
     print ('\nThere are no valid pixels in a 3x3 grid centered at your entered location.')
else:
     three_by_three=three_by_three*scale_factor
     three_by_three_average=np.nanmean(three_by_three)
     three_by_three_std=np.nanstd(three_by_three)
     three_by_three_median=np.nanmedian(three_by_three)
     if nnan == 1:
          npixels='is'
          mpixels='pixel'
     else:
          npixels='are'
          mpixels='pixels'
     print('\nThere',npixels,nnan,'valid',mpixels,'in a 3x3 grid centered at your entered location.')
     print('\nThe average value in this grid is: ',round(three_by_three_average,3),' \nThe median value in this grid is: ',round(three
```


Aplicaciones

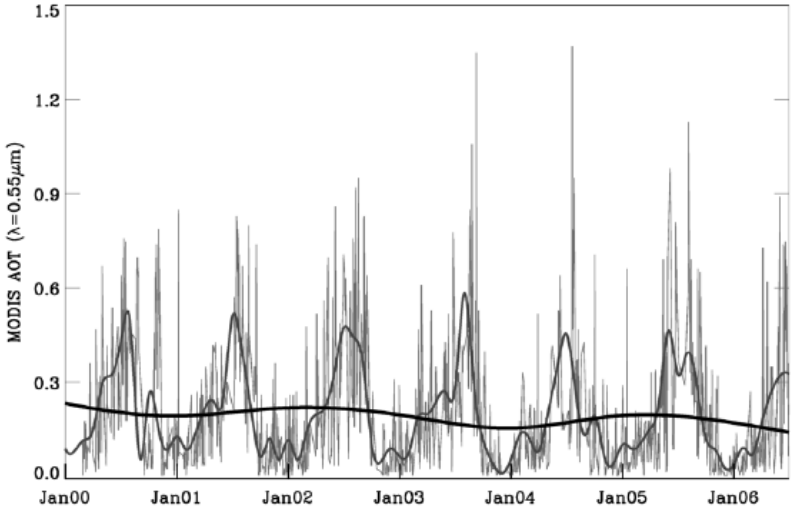
Validación del AOD por Satélite

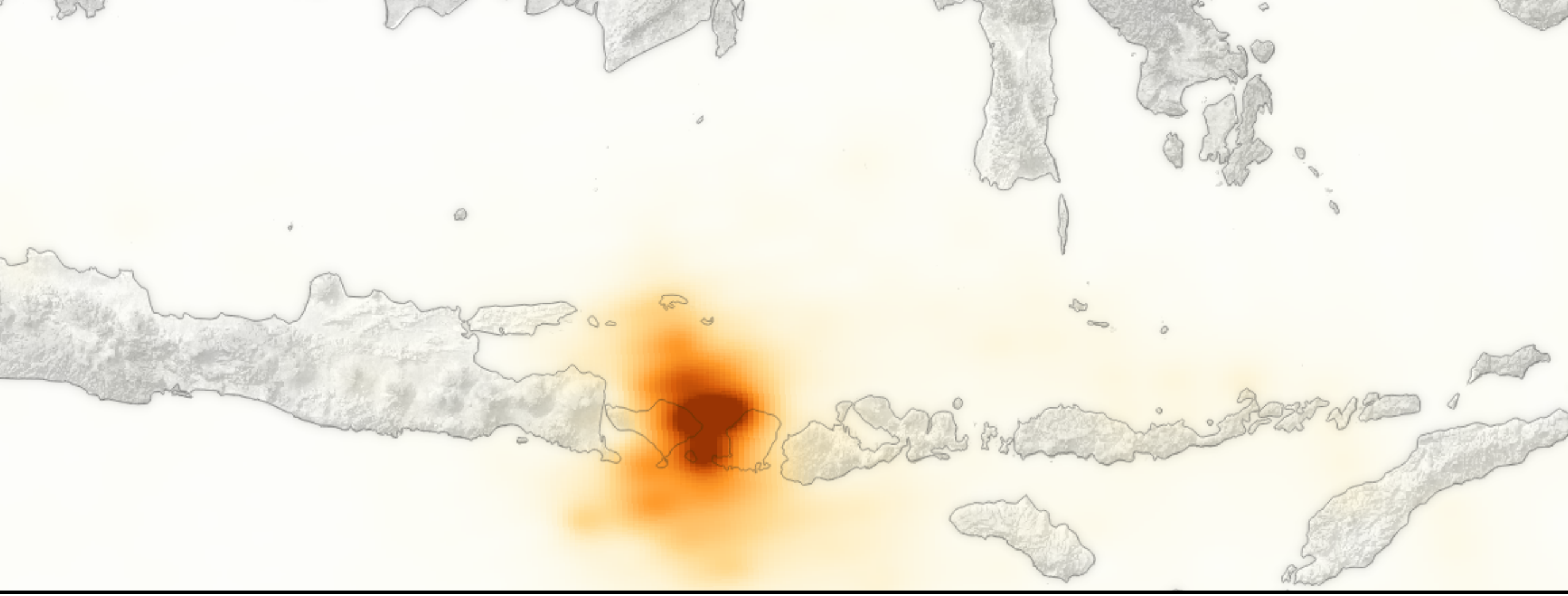


Relación AOD-$PM_{2.5}$



Análisis de Series Temporales



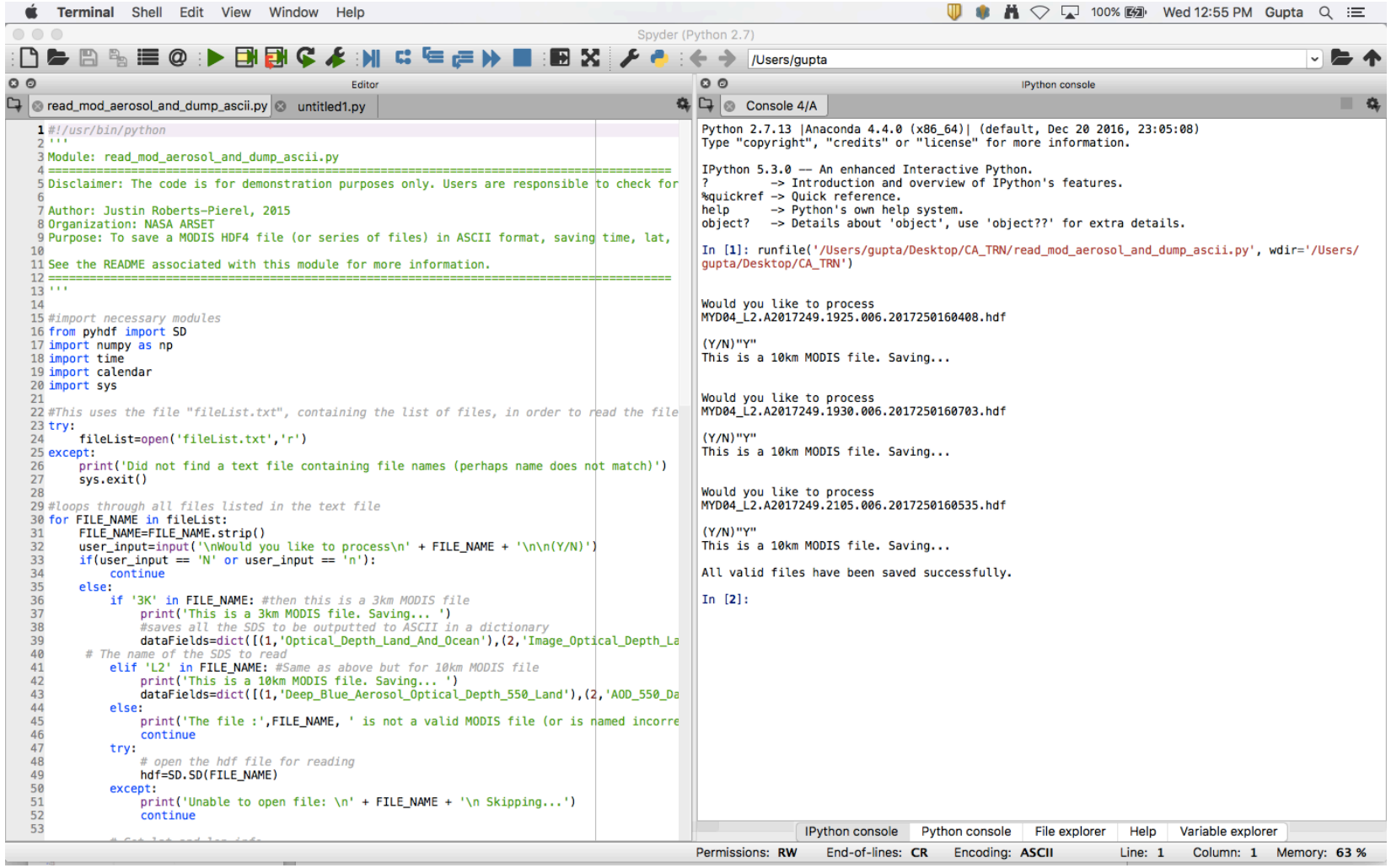


Convierta las Variables del HDF en CSV

Convierta variables de un HDF Nivel 2 de MODIS en un Archivo CSV

read_mod_aerosol_and_dump_ascii.py

- **Propósito**: leer un archivo de datos de MODIS de Nivel 2 en formato HDF y convertir ciertos SDSs en un archivo csv (texto)
- El código funciona para productos de 10 km y 3 km



Producto

MYD04_L2.A2017249.1925.006.2017250160408.txt

```
Year,Month,Day,Hour,Minute,Second,Latitude,Longitude,Deep_Blue_Aerosol_Optical_Depth_550_Land,AOD_550_Dark_Target_Deep_Blue_Combined,AOD_550_Dark_Target_Deep_Blue_Combined_QA_Flag
2017.0,9.0,6.0,19.0,25.0,9.0,30.4542312622,-80.2554473877,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.4285984039,-80.7235641479,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.4032402039,-81.1592407227,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.3782196045,-81.5666427612,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.3535690308,-81.9493026733,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.329334259,-82.3096008301,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.3055496216,-82.6497421265,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.2822036743,-82.9720306396,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.2592868805,-83.2782287598,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.2367897034,-83.5699691772,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.2147369385,-83.8481216431,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.1930789948,-84.1143035889,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.1718177795,-84.3694152832,0.322000015294,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.1509361267,-84.6143875122,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.1304397583,-84.8497695923,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.1103076935,-85.0764007568,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.0905189514,-85.2949752808,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.0710792542,-85.5059127808,0.496000023559,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.0519447327,-85.7100372314,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.0331192017,-85.9076538086,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,30.0145874023,-86.0991973877,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.9963378906,-86.2850646973,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.9783554077,-86.4656219482,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.9606361389,-86.6411895752,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.9431610107,-86.8120880127,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.9259204865,-86.9785919189,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.9089050293,-87.1409683228,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.8921031952,-87.2994613647,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.87550354,-87.4542999268,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.8590984344,-87.6056747437,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.8428764343,-87.7537918091,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.8268318176,-87.8988342285,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.8109512329,-88.040977478,-9999.0,0.483000022941,1.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.7952270508,-88.1803741455,-9999.0,0.537000025506,3.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.7796516418,-88.3171691895,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.7642173767,-88.4515228271,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.7489128113,-88.5835418701,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.7337341309,-88.7133789062,-9999.0,0.523000024841,1.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.7186717987,-88.8411331177,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.7037162781,-88.9669265747,-9999.0,0.400000018999,1.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.688867569,-89.0908660889,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.6741104126,-89.2130584717,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.6594429016,-89.3335876465,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.6448535919,-89.452545166,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.6303424835,-89.5700378418,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.6158943176,-89.6861343384,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.6015090942,-89.8009262085,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.587179184,-89.9144897461,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.5728931427,-90.0269012451,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.5586509705,-90.1382369995,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.5444412231,-90.248550415,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.5302619934,-90.3579177856,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.5161094666,-90.4663772583,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.5019721985,-90.5740280151,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.4878482819,-90.6808853149,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.4737319946,-90.7870483398,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.4596118927,-90.8925476074,-9999.0,-9999.0,0.0
2017.0,9.0,6.0,19.0,25.0,9.0,29.4454841614,-90.9974594116,-9999.0,-9999.0,0.0
```

Este código crea un archivo .csv, el cual se puede abrir con Excel, un editor de texto u otros códigos o software

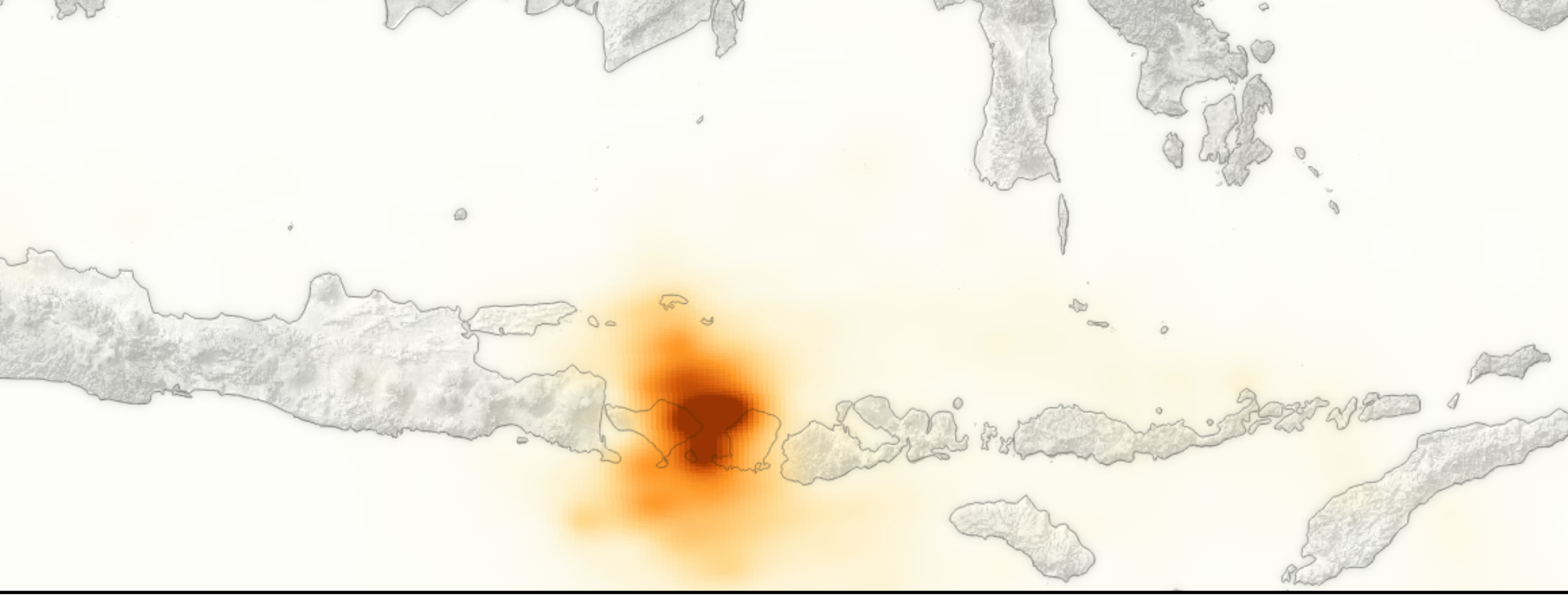
Editando el Código

Cambie el SDS de la lista a que sea escrito como un producto

```
21
22 #This uses the file "fileList.txt", containing the list of files, in order to read the file
23 try:
24     fileList=open('fileList.txt','r')
25 except:
26     print('Did not find a text file containing file names (perhaps name does not match)')
27     sys.exit()
28
29 #loops through all files listed in the text file
30 for FILE_NAME in fileList:
31     FILE_NAME=FILE_NAME.strip()
32     user_input=input('\nWould you like to process\n' + FILE_NAME + '\n\n(Y/N)')
33     if(user_input == 'N' or user_input == 'n'):
34         continue
35     else:
36         if '3K' in FILE_NAME: #then this is a 3km MODIS file
37             print('This is a 3km MODIS file. Saving... ')
38             #saves all the SDS to be outputted to ASCII in a dictionary
39             dataFields=dict([(1,'Optical_Depth_Land_And_Ocean'),(2,'Image_Optical_Depth_La
40          # The name of the SDS to read
41         elif 'L2' in FILE_NAME: #Same as above but for 10km MODIS file
42             print('This is a 10km MODIS file. Saving... ')
43             dataFields=dict([(1,'Deep_Blue_Aerosol_Optical_Depth_550_Land'),(2,'AOD_550_Da
44         else:
45             print('The file :',FILE_NAME, ' is not a valid MODIS file (or is named incorre
46             continue
47         try:
48             # open the hdf file for reading
49             hdf=SD.SD(FILE_NAME)
50         except:
51             print('Unable to open file: \n' + FILE_NAME + '\n Skipping...')
52             continue
53
```


Aplicaciones

- Éste es un ejemplar de un código para leer y extraer los datos de aerosoles de MODIS de Nivel 2
- Se puede modificar el código para extraer múltiples SDSs en un solo archivo .csv
- Se puede modificar el código fácilmente para extraer datos sobre una región en particular
- El archivo producido se puede abrir en Excel o cualquier otra herramienta para analizar datos

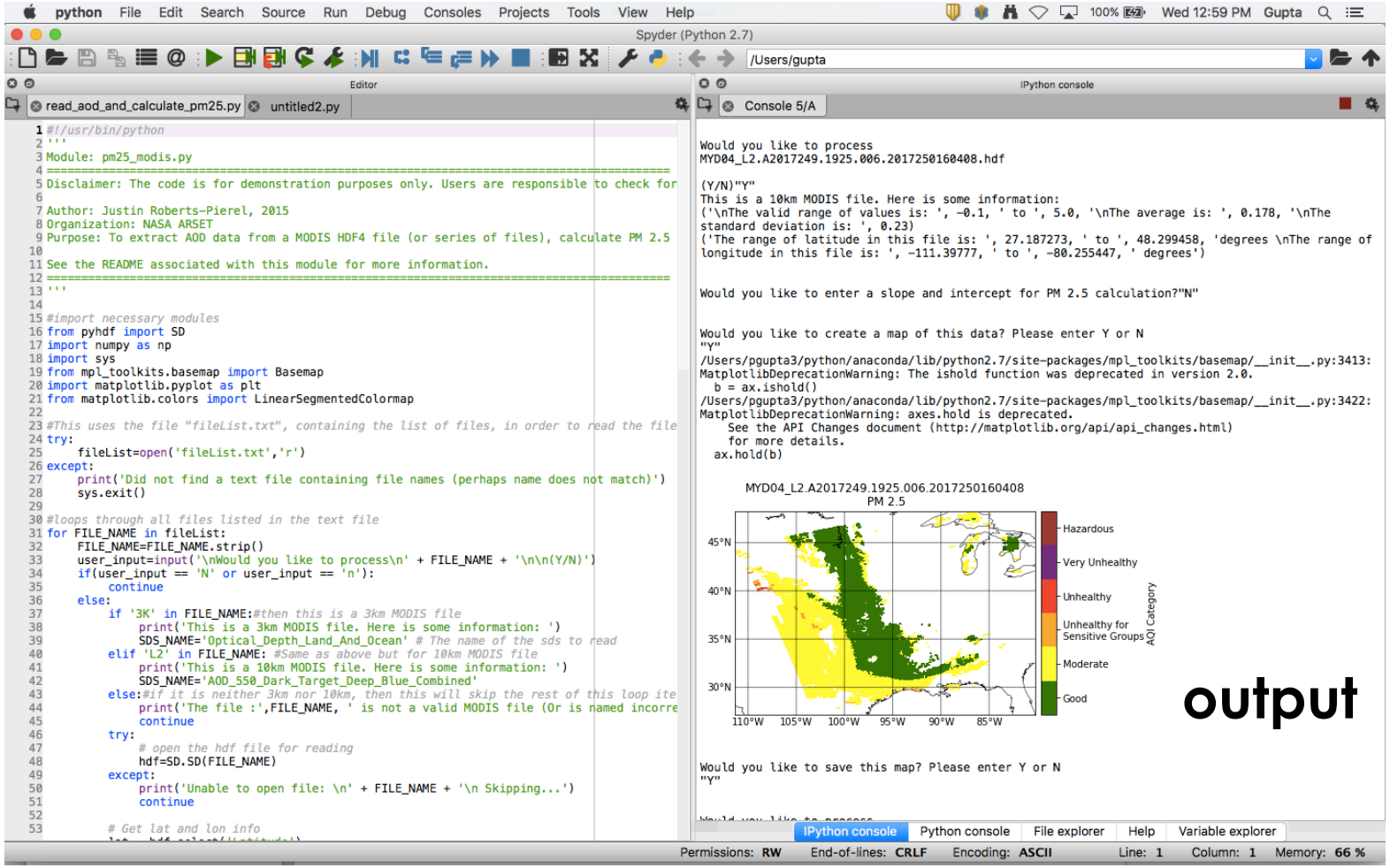


Cree Mapas de la Calidad del Aire

Cree un Mapa de la Calidad del Aire

read_aod_and_calculate_pm25.py

- **Propósito**: leer un archivo de datos de aerosoles de MODIS de nivel 2 en formato HDF y crear un mapa de categorías de calidad del aire usando la relación entre el AOD y PM2.5
- El código sirve para productos de 10 km y 3 km



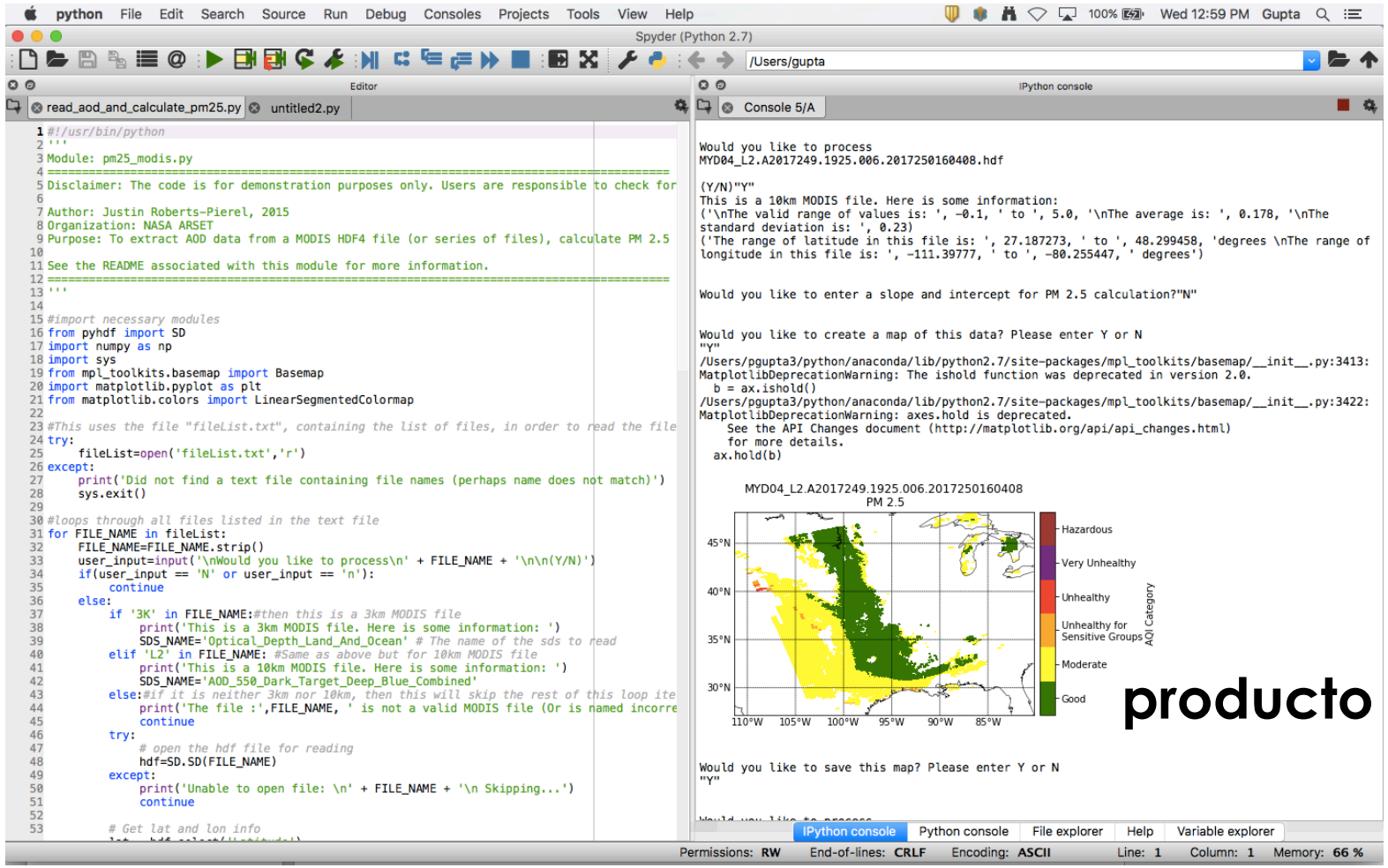
output

Cree un Mapa de la Calidad del Aire

read_aod_and_calculate_pm25.py

Advertencia: Éste es sólo un ejemplar de un código. La relación AOD-PM2.5 por defecto que se usa aquí es la relación supuesta sobre EEUU.

Los usuarios de este código tienen la responsabilidad de verificar la validez de esta relación y se les recomienda utilizar relaciones locales para visualizar la Calidad del Aire en diferentes partes del mundo.



producto

Editando el Código – Cambie el SDS

El usuario puede cambiar el SDS del AOD a ser usado en el cálculo del PM2.5

```
30 #loops through all files listed in the text file
31 for FILE_NAME in fileList:
32     FILE_NAME=FILE_NAME.strip()
33     user_input=input('\nWould you like to process\n' + FILE_NAME + '\n\n(Y/N)')
34     if(user_input == 'N' or user_input == 'n'):
35         continue
36     else:
37         if '3K' in FILE_NAME:#then this is a 3km MODIS file
38             print('This is a 3km MODIS file. Here is some information: ')
39             SDS_NAME='Optical_Depth_Land_And_Ocean' # The name of the sds to read
40         elif 'L2' in FILE_NAME: #Same as above but for 10km MODIS file
41             print('This is a 10km MODIS file. Here is some information: ')
42             SDS_NAME='AOD_550_Dark_Target_Deep_Blue_Combined'
43         else:#if it is neither 3km nor 10km, then this will skip the rest of this loop iteration
44             print('The file :',FILE_NAME, ' is not a valid MODIS file (Or is named incorrectly). \n')
45             continue
46         try:
47             # open the hdf file for reading
48             hdf=SD.SD(FILE_NAME)
49         except:
50             print('Unable to open file: \n' + FILE_NAME + '\n Skipping...')
51             continue
52
```


Editando el Código: Cambie la Relación AOD-$PM_{2.5}$ y el AQI

El código usa
$PM_{2.5}$ = Slope * AOD + Intercept
como la ecuación de regresión lineal para calcular $PM_{2.5}$ a partir del AOD

El código utiliza la definición de la Administración de Protección Ambiental de EEUU de categorías de calidad del aire a base de $PM_{2.5}$

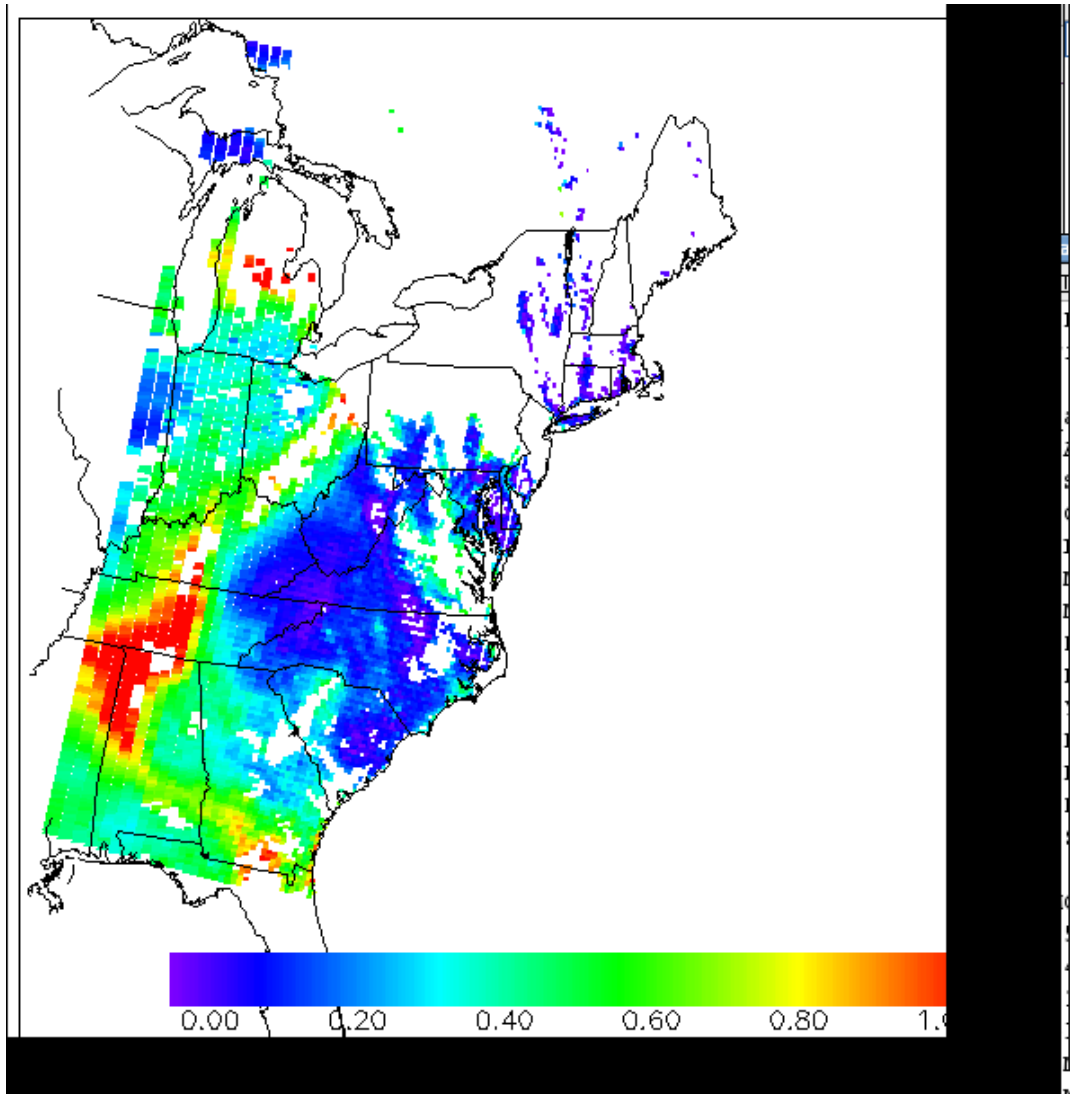
Calculador del AQI: https://airnow.gov/index.cfm?action=airnow.calculator

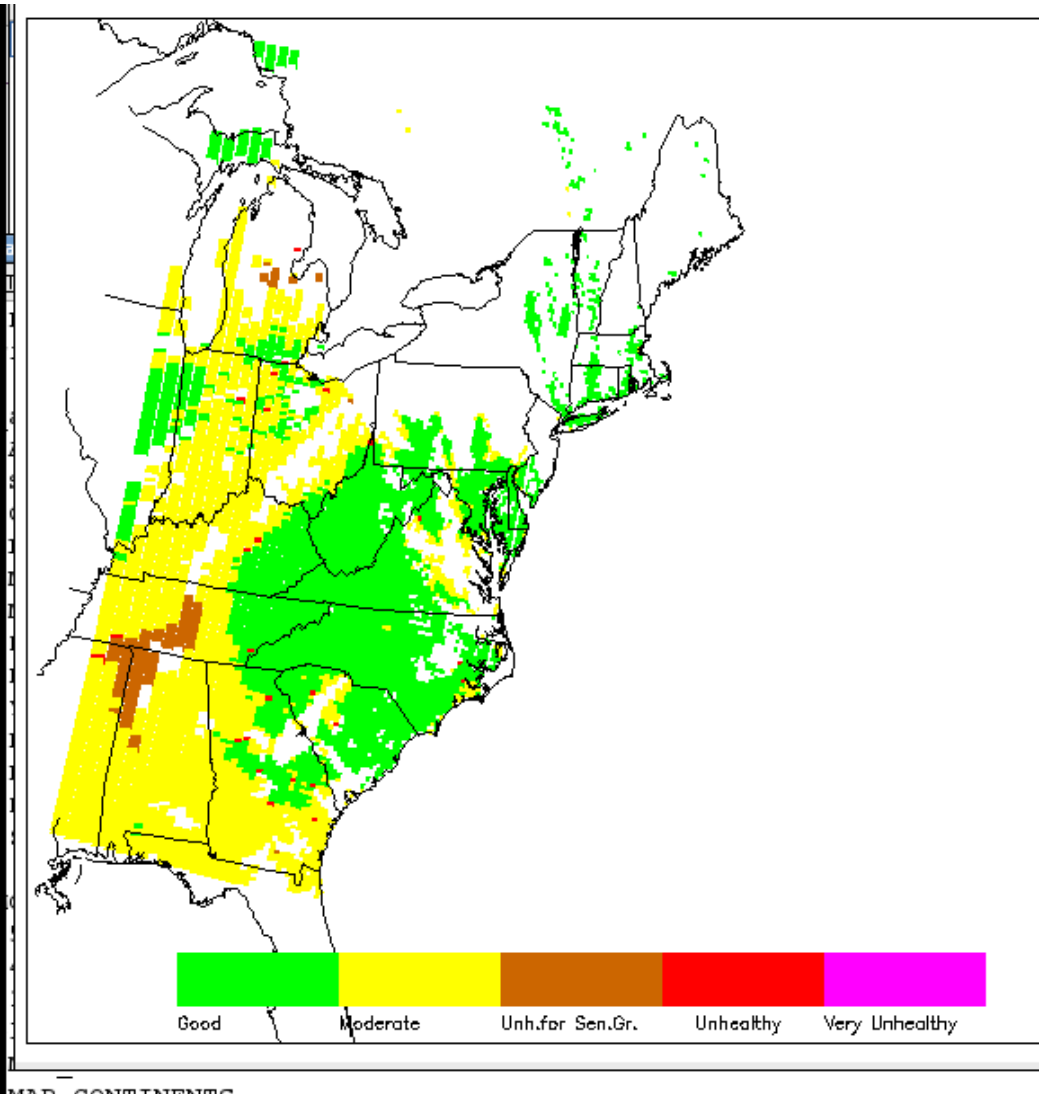
```
#asks user if they want to set PM2.5 calculation parameters
user_input=input('\nWould you like to enter a slope and intercept for PM 2.5 calculation?')
if user_input == 'Y' or user_input == 'y':
    slope=input('Please enter a slope: ')
    intercept=input('Please enter an intercept: ')
else:
    #if not, choose the following:
    slope=29.4
    intercept=8.8
valid_data=data*scale_factor
pm25=float(slope)*valid_data+float(intercept)


#Asks user if they would like to see a map
is_map=input('\nWould you like to create a map of this d
#if user would like a map, view it
if is_map == 'Y' or is_map == 'y':
    #turn fillvalues to NaN
    data=pm25.astype(float)
    data[np.logical_and(data>=0,data <= 12)]=0
    data[np.logical_and(data>12,data <= 35.4)]=1
    data[np.logical_and(data>35.4,data <= 55.4)]=2
    data[np.logical_and(data>55.4,data <= 150.4)]=3
    data[np.logical_and(data>150.4,data <= 250.4)]=4
    data[data>250.4]=5
    data[data < 0] = np.nan
    #create the map
    data = np.ma.masked_array(data, np.isnan(data))
    m = Basemap(projection='cyl', resolution='l', llcrnrlat=min_lat, urcrnrlat = max_lat, llcrnrlon=min_lon,
    m.drawcoastlines(linewidth=0.5)
    m.drawparallels(np.arange(-90., 120., 5.), labels=[1, 0, 0, 0])
    m.drawmeridians(np.arange(-180., 181., 5.), labels=[0, 0, 0, 1])
    x, y = m(longitude, latitude)
    my_cmap=LinearSegmentedColormap.from_list('mycmap', ['green','yellow','orange','red','purple','brown'],6
    m.pcolormesh(x, y, data,cmap=my_cmap)
    plt.clim(0,6)
    #create colorbar
    cb = m.colorbar()
    cb.set_label('AQI Category')
    cb.set_ticks([.5, 1.5,2.5,3.5,4.5,5.5])  # force there to be only 7 ticks
    cb.set_ticklabels(['Good', 'Moderate', 'Unhealthy for \nSensitive Groups','Unhealthy','Very Unhealthy','
```

Cambie el "slope" (pendiente) y el "intercept" preprogramados

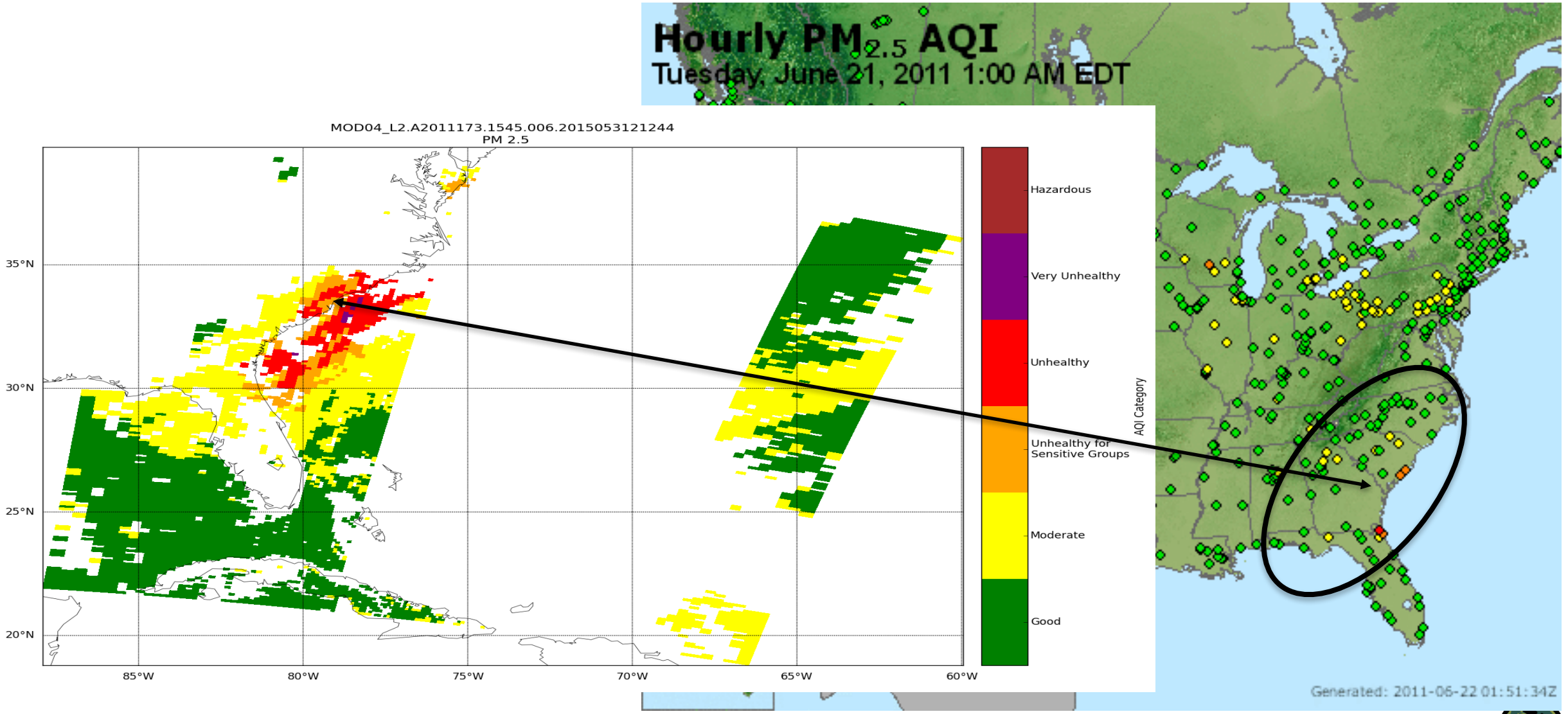
Cambie las categorías de calidad del aire

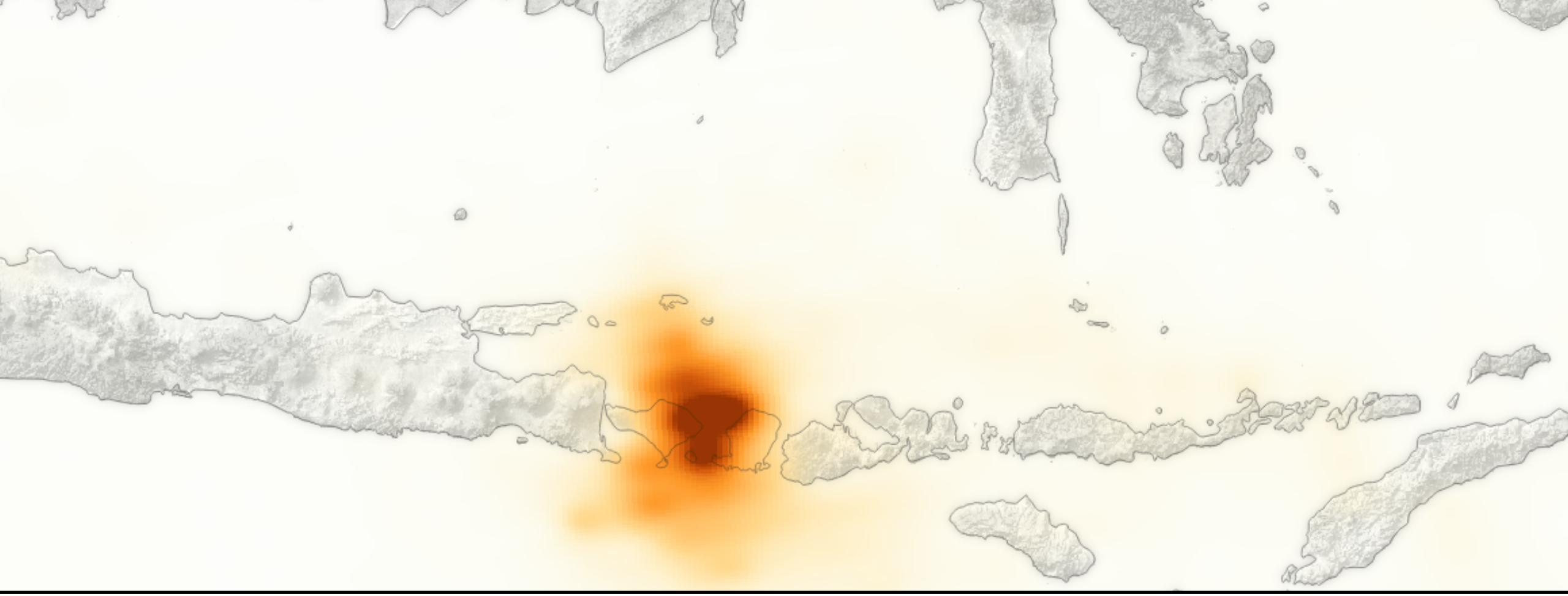
Aplicación: Convierta el AOD en $PM_{2.5}$ y Mapas de la Calidad del Aire





Aplicación – Compare Mapas Satelitales con Mapas Superficiales





Preguntas