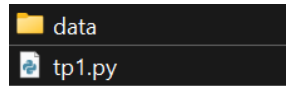


TP1 – Analyse des radiosondes

Télécharger puis extraire dans un dossier dédié le **78970_2016_10.zip**. Ce dernier contient les données de radiosondage (par ballon météo) pour le mois d'octobre 2016. Chaque jour est stocké dans un fichier.

Créer un fichier **tp1.py** et placer à l'extérieur de ce dossier :



Ouvrir un terminal, (chercher cmd sous Windows) puis installer les modules suivants avec

`pip install nom_module :`

```
pip install numpy
pip install matplotlib
pip install python-csv
pip install windrose openpyxl
```

Ouvrir Visuel Studio Code, puis faite **Fichier>Ouvrir le dossier...** (en anglais File>Open folder...)

Choisir le dossier où se trouve votre TP1 et cliquer sur **Accepter de faire confiance aux éléments du dossier**.

C'est bon, vous êtes prêts à travailler.

Dans votre code python, utilisez la librairie `os` pour lister les fichiers de radiosondage présents dans le dossier `data` :

```
import os
for file in os.listdir("./data/"):
    if ('.wyo' in file) and file.endswith(".csv"):
        print(file)
#liste les fichiers
#teste les extensions
#affiche le nom du fichier
```

Dans votre code python, utilisez la librairie `csv` pour lire le contenu d'un fichier, puis `numpy` pour convertir une liste en tableau :

```
import csv
import numpy as np
def getDataFile(fname):
    with open(fname, newline='') as csvfile:
        rd = csv.reader(csvfile, delimiter=',')
        h = next(rd)
        data=[]
        for line in rd:
            l=[]
            for v in line:
                if v=="":
                    l.append(float('nan'))
                else:
                    l.append(float(v.replace(",",".")))
            data.append(l)
    return np.asarray(data)
#extraite l'entête
#initialise une liste vide : data
#parcours les lignes du fichier
#initialise une liste vide : l
#parcours les colonnes de la ligne
#remplace les champs vide par NaN, rajoute à la liste l
#remplace les "," par des ".", rajoute à la liste l
#rajoute à la liste data
#renvoie la liste data sous forme de tableau
```

Exercice 1 : Fichiers, listes et tableaux

- a) Ecrire une fonction python qui calcul et renvoie la moyenne des MIXR de surface (<100m) pour un fichier donné.
- b) Ecrire une fonction qui renvoie le niveau de pression en hPa de passage à des températures négatives (à voir dans Temp (C)) pour un fichier donné.
- c) Ecrire une fonction qui renvoie la vitesse maximale en nœud (voir dans SKNT (knot)) pour un fichier donné.

--

Dans votre rapport, présenter les fonctions demandées puis à partir de tous les radiosondages déterminer automatiquement :

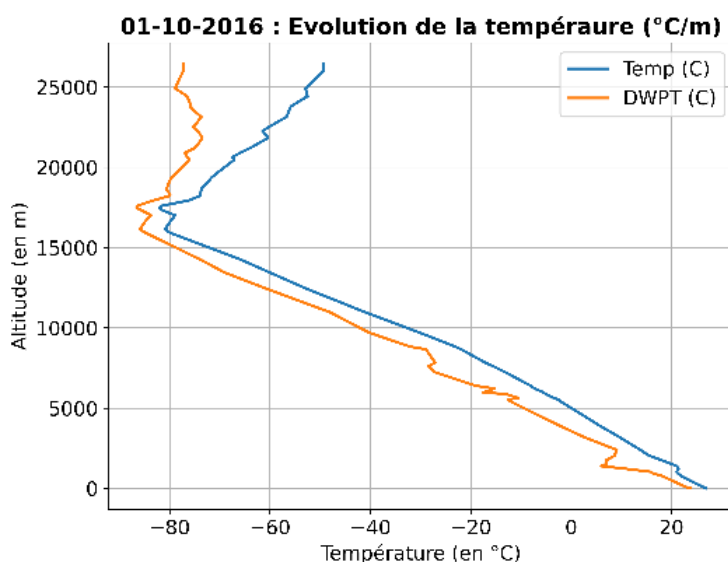
- le nombre et les dates des jours manquants dans le mois d'octobre 2016 ;
- le jour où la moyenne des MIXR de surface est la plus faible ;
- le plus haut niveau de pression (hPa) où le changement de température en négative se produit ;
- la moyenne des vitesses maximales ;

Exercice 2 : Graphiques

Pour tracer des graphiques, vous devez utiliser la librairie matplotlib, les codes suivants permettent de créer différentes courbes, compléter les pour répondre aux questions, puis intégrer les dans votre rapport.

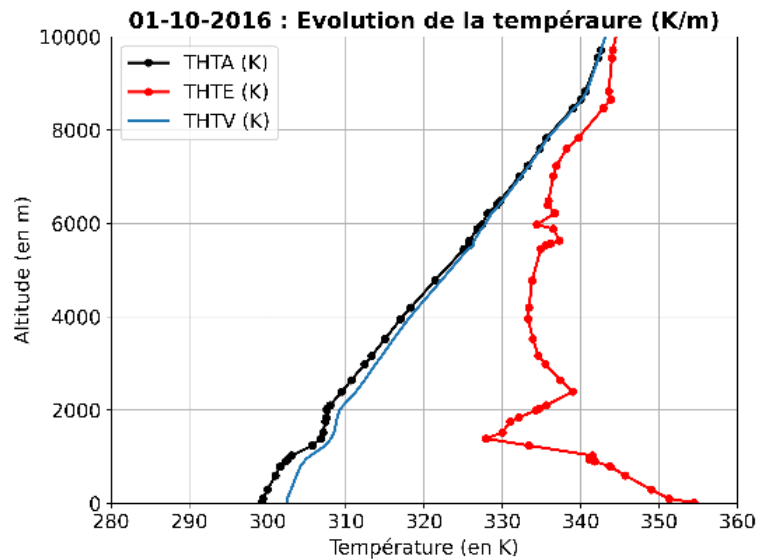
```
##Graphique 1
import matplotlib.pyplot as plt
fs=14                                     #taille du texte
lw=2                                     #épaisseur courbes
plt.rcParams.update({'font.size': fs})   #taille de la police
fig=plt.figure(figsize=(8,6))           #proportion de la figure
plt.plot(x2, y1, label='nom_courbe1', linewidth=lw) #courbe1
plt.plot(x2, y2, label='nom_courbe2', linewidth=lw) #courbe2
plt.xlabel('nom_axe1', fontsize=fs)      #nom de l'axe des abscisses
plt.ylabel('nom_axe2', fontsize=fs)      #nom de l'axe des ordonnées
plt.title('titre_graphique', fontsize=fs+2, fontweight="bold")
plt.grid(True)                          #afficher grille
plt.legend()                             #afficher legende
fig.savefig('nom_img.png', dpi=600)     #enregistrer fichier
plt.show()                              #afficher graphique
```

- a) Ecrire un programme python qui permet de reproduire ce graphique, à partir du code donné en exemple mais pour le fichier en date du 21-10-2016.



```
##Graphique 2
import matplotlib.pyplot as plt
fs=14                                     #taille du texte
lw=2                                     #épaisseur courbes
ms=10                                    #taille puce
fig=plt.figure(figsize=(8,6))           #proportion de la figure
plt.plot(x1, y1, '.-', label='nom_courbe1', color='black', linewidth=lw, markersize=ms)
plt.plot(x2, y2, '.-', label='nom_courbe2', color='red', linewidth=lw, markersize=ms)
plt.plot(x3, y3, label='nom_courbe3', linewidth=lw)
plt.xlabel('nom_axe1', fontsize=fs)      #nom de l'axe des abscisses
plt.ylabel('nom_axe2', fontsize=fs)      #nom de l'axe des ordonnées
plt.title('titre', fontsize=fs+2, fontweight="bold")
plt.grid(True)                          #afficher grille
plt.legend()                             #afficher legende
plt.xlim((limit_x1,limit_x2))           #limitation de l'axe des abscisses
plt.ylim((limit_y1,limit_y2))           #limitation de l'axe des ordonnées
fig.savefig('nom_fichier.png', dpi=600) #enregistrer fichier
plt.show()                              #afficher graphique
```

- b) Ecrire un programme python qui permet de reproduire ce graphique, à partir du code donné en exemple mais pour le fichier en date du 21-10-2016.



```
##Graphique 3
import matplotlib.pyplot as plt
from windrose import WindroseAxes as wra
fs=14 #taille du texte
ax = wra.from_ax() #création figure rose des vents
fig = plt.gcf() #récupération de l'objet figure
ax.bar(x1, x2, bins=[1, 3, 6, 10, 16, 21, 27, 33, 40, 47, 55, 63], normed=True)
plt.grid(True) #affichage d'une grille
plt.title("titre", fontsize=fs+6, fontweight="bold") #définition titre
plt.legend() #définition légende
fig.savefig('nom_fichier.png', dpi=600) #enregistrer fichier
plt.show() #afficher graphique
```

- c) Ecrire un programme python qui permet de reproduire ce graphique, à partir du code donné en exemple mais pour le fichier en date du 21-10-2016.

