

Ανάλυση προϊόντων NWC-CT/CMA και ανάπτυξη προγράμματος προβολής τους.

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών

Τμήμα θετικών επιστημών
Τμήμα Αεροδιαστημικής Επιστήμης και Τεχνολογίας
Επιβλέπων καθηγητής : Σταύρος



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικόν και Καποδιστριακόν
Πανεπιστήμιον Αθηνών
— ΙΔΡΥΘΕΝ ΤΟ 1837 —

I. ΕΙΣΑΓΩΓΗ

Τα σύννεφα έχουν μεγάλη επιρροή στις ζωές των ανθρώπων αλλά και στα οικοσυστήματα της γης. Μπορούν να είναι πηγή ενός δροσερού και γόνιμου περιβάλλοντος αλλά και καταστροφών για οικισμούς και οικοσυστήματα. Για αυτό είναι πολύς σημαντική η μελέτη και η κατανόηση αυτού του φαινομένου . Μαζί με την ανάπτυξη των δορυφόρων τα τελευταία χρόνια έχουν αναπτυχθεί διάφορα προϊόντα για την ανάλυσή τους. Ένα από αυτά τα προϊόντα είναι τα προϊόντα nwc-saf από την Eumetsat τα οποία είναι προϊόντα που προέρχονται από δορυφόρο σε γεωστατική τροχιά .

Παρακάτω θα ακολουθήσουν τα εξής :

- 1) Τα προϊόντα NWC-SAF-CT/CMA
- 2) Δεδομένα
- 3) Επεξεργασία
- 4) Αποτελέσματα
- 5) Συμπεράσματα

II. ΤΑ ΠΡΟΪΟΝΤΑ NWC-SAF-CT/CMA

Όπως προαναφέρθηκε στο πλαίσιο αυτής της εργασίας θα χρησιμοποιηθούν δεδομένα του προϊόντος nwc-saf της Eumetsat ποιο συγκεκριμένα θα αναλυθούν τα προϊόντα cma (cloud mask.), το οποίο δείχνει την θέση των νεφών μορφή , μαζί και με άλλα δεδομένα όπως ηφαιστειακή σκόνη και ct(cloud type), το οποίο κάνει κατανομή των νεφών σε υποκατηγορίες βάση των χαρακτηριστικών τους.

A. Επισκόπηση των δεδομένων cloud type (CT)

Τα δεδομένα ct(cloud type) όπως προαναφέρθηκε προσφέρουν πληροφορία για τον τύπο και χαρακτηριστικά των νεφών όπως και το που βρίσκονται αυτά . Ποιο συγκεκριμένα έχουν 'δείκτες' για κάθε χαρακτηριστικό οι οποίοι έχουν μία συγκεκριμένη έννοια ανάλογα το χαρακτηριστικό. Επίσης τα δεδομένα συμπεριλαμβάνουν

επίσης παλέτες για κάθε χαρακτηριστικό για έγχρωμη απεικόνιση των δεδομένων. Και τέλος παρέχονται δεδομένα γεωγραφικής αναφοράς σε μέτρα. Παρακάτω φαίνεται πίνακας των δεικτών για κάθε χαρακτηριστικό των νεφών .

Όνομα (unit8)	Δείκτες
ct	1:Στεριά χωρίς σύννεφα 2:Θάλασσα χωρίς σύννεφα 3:χιόνι πάνω σε στεριά 4:Πάγος στην θάλασσα 5:Πολύ χαμηλά σύννεφα 6:Χμηλά σύννεφα 7: Μεσαίου ύψους σύννεφα 8: Υψηλά αδιαφανή σύννεφα 9: Πολύ υψηλά αδιαφανή σύννεφα 10: Κλασματικά σύννεφα 11:Υψηλά ημιδιαφανή λεπτά σύννεφα 12: Υψηλά ημιδιαφανή μέτρια σύννεφα 13: Υψηλά ημιδιαφανή παχιά σύννεφα 14:Υψηλά ημιδιαφανή πάνω από χαμηλά ή μεσαία σύννεφα 15: Υψηλό ημιδιαφανές πάνω από χιόνι πάγο
ct_cumuliform	1: Στρατομορφική κατάσταση 2: Κατάσταση cumuliform 3: Μικτή κατάσταση 4: Χωρίς σύννεφα 5: Απροσδιόριστα προβλήματα διαχωρισμού
ct_status_flag	1: Θερμική αναστροφή χαμηλού επιπέδου στο πεδίο NWP 2: Η θερμοκρασία Τροπόπαυση είναι διαθέσιμη από το NWP 4:138um χρησιμοποιείται για την αναγνώριση cirrus 8: Χρησιμοποιούνται δορυφορικά δεδομένα υψηλής ανάλυσης 16: Καμία μέθοδος για στρωματοειδή σωρευτικό διαχωρισμό 18: Καμία μέθοδος για πολλαπλές στρώσεις

B. Επισκόπηση των δεδομένων cloud mask (CMA)

Αντίστοιχα το προϊόν cma(cloud mask) προσφέρει δεδομένα για την μορφή των νεφών και από τί αποτελούνται όπως σκόνη , καπνό και χιόνι. Όπως και προηγουμένως κάθε ένα από τα χαρακτηριστικά συνοδεύετε από δίκες για την ένδειξη ύπαρξης ή όχι του χαρακτηριστικού αλλά και συμπλήρωση πληροφορίας για το χαρακτηριστικό .Παρακάτω φαίνεται πίνακας των δεικτών για κάθε χαρακτηριστικό των νεφών .

Όνομα (unit8)	Δείκτες
cma_cloudsnow	0: χωρίς σύννεφα 1: σύννεφα(εκτός πάγο πάνω σε χιόνι) 2: λεπτός σύννεφα πάγου πάνω από χιόνι/πάγο 3: χιόνι/πάγος
cma	0: χωρίς σύννεφα 1: σύννεφα
cma_dust	0: χωρίς σκόνη 1: σκόνη 2: απροσδιόριστο
cma_smoke	0: χωρίς καπνό 1: καπνός 2: απροσδιόριστο

C. Τρόπος αποθήκευσης και επεξεργασίας

Ο τρόπος αποθήκευσης αυτών των δεδομένων είναι τα μέσω αρχείων netCDF.Το οποίο είναι ένα σύνολο από βιβλιοθήκες και αυτό-περιγραφόμενες μορφές δεδομένων και προσφέρει πρόσβαση κοινή χρήση και δημιουργία επιστημονικών δεδομένων σε μορφή πινάκων.

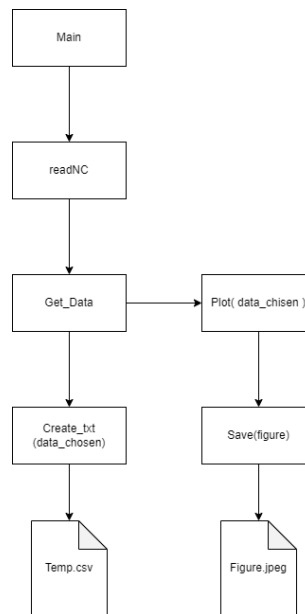
III. ΕΠΕΞΕΡΓΑΣΕΙΑ

Στην εποχή της πληροφορίας η επεξεργασία δεδομένων και η ανάκτηση τους είναι ένα βασικό βήμα και έχουν αναπτυχθεί πολλές γλώσσες προγραμματισμού για την επίτευξη αυτών. Μία γλώσσα προγραμματισμού από αυτές είναι η Python η οποία χρησιμοποιείτε σε αυτή την εργασία.

Τα βήματα για την επίτευξη αυτής της εργασίας είναι δημιουργία γραφικού περιβάλλοντος ,εξόρυξη των δεδομένων από τα αρχεία netCDF, δημιουργία αρχείου csv και δημιουργία γραφήματος για την προβολή των εικόνων . Για αυτό χρησιμοποιούνται οι παρακάτω βιβλιοθήκες .

- Customtkinter (για το γραφικό περιβάλλον)
- netCDF4 (για την ανάκτηση των δεδομένων από αρχεία netCDF)
- numpy & pandas (για την δημιουργία αρχείου)
- matplotlib (για την σχεδίαση γραφημάτων)

Ποιο αναλυτικά στην παρακάτω εικόνα (Εικόνα 1) φαίνεται το σχεδιάγραμμα που δείχνει την σχεδίαση του κώδικα (βλέπε παράρτημα) βάση των συναρτήσεων.



Εικόνα 1 Διάγραμμα προγράμματος

Όπως φαίνεται στην εικόνα το πρόγραμμα αποτελείτε από έξι συναρτήσεις οι οποίες θα αναφερθούν αναλυτικά παρακάτω.

1) Main()

Αυτή η συνάρτηση είναι υπεύθυνη για την δημιουργία του γραφικού περιβάλλοντος συγκεκριμένα δύο πλαίσιων , τον ορισμό πέντε κουμπιών , από τα οποία εμφανίζει ένα και ενός Entry για την απόκτηση του μονοπατιού του αρχείου .

2) readNC()

Αυτή η συνάρτηση εκτελείτε όταν επιλέγετε ένα κουμπί και δημιουργεί Checkboxes με τα bands των εικόνων και εμφανίζει το δεύτερο κουμπί .

3) Get_Data()

Όταν γίνεται επιλογή του δεύτερου κουμπιού η συνάρτηση παίρνει την έξοδο των checkboxes και βάση αυτής καλούνται τα δεδομένα από το αρχείο .Και γίνεται εμφάνιση δύο κουμπιών

4) Create_txt()

Όταν επιλέγετε το τρίτο κουμπί εκτελείτε αυτή η συνάρτηση η οποία δημιουργεί ένα αρχείο temp.csv με τα δεδομένα που επιλέχθηκαν προηγουμένως και αντικαθιστά τα '--' με την τιμή 5.

5) Plot()

Με την επιλογή του τέταρτου κουμπιού γίνεται η εκτέλεση αυτής της συνάρτησης . Η οποία δημιουργεί έναν καμβά στον οποίο παρουσιάζετε το διάγραμμα που δημιουργείτε βάση

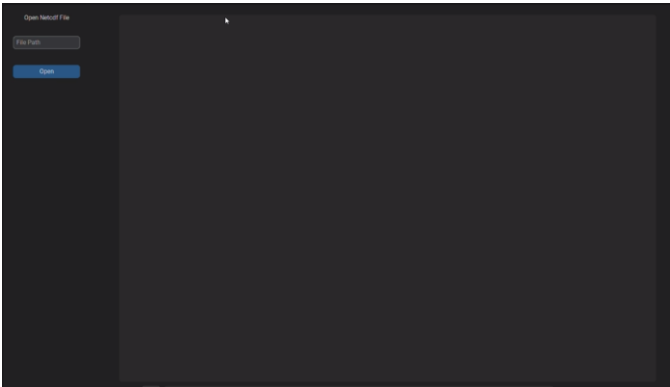
των επιλεγμένων δεδομένων και εμφανίζεται το τελευταίο κουμπί .

6) Save()

Με την επιλογή του κουμπιού που προαναφέρθηκε εκτελείτε αυτή η συνάρτηση και αποθηκεύει το διάγραμμα σε εικόνα Figure.jpeg.

IV. ΑΠΟΤΕΛΕΣΜΑΤΑ

Όταν το πρόγραμμα που προαναφέρθηκε εκτελείτε δημιουργείτε το παρακάτω γραφικό περιβάλλον (Εικόνα 2)



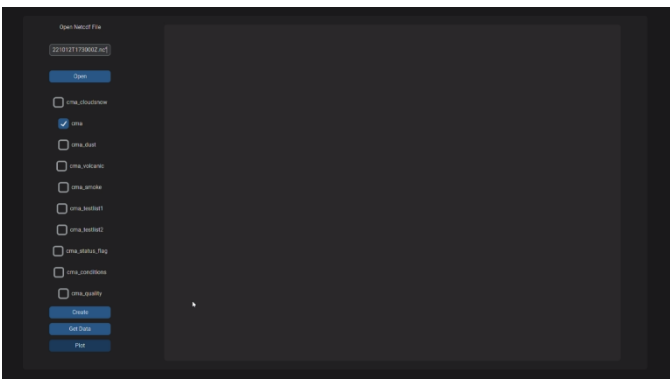
Εικόνα 2 Αρχικό γραφικό περιβάλλον

Έπειτα όταν το κουμπί ‘Open ’ εκτελείτε η συνάρτηση readNC() και το γραφικό περιβάλλον γίνεται όπως φαίνεται στην παρακάτω εικόνα(Εικόνα 3).



Εικόνα 3 Δημιουργία Checklist

Όταν γίνεται η επιλογή των band και επιλέγετε το κουμπί ‘Create’ εκτελείτε η συνάρτηση Get_Data() και το γραφικό περιβάλλον ανανεώνετε στο παρακάτω(Εικόνα 4).



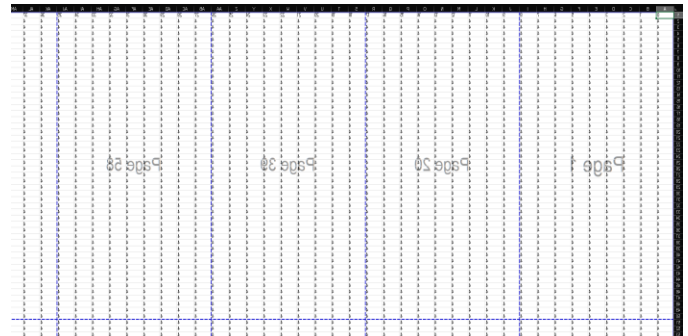
Εικόνα 4 Μετά την εκτέλεση του Get_Data()

Και τέλος όταν επιλέγετε το κουμπί Plot εμφανίζετε το διάγραμμα που επιλέχθηκε και το γραφικό περιβάλλον ανανεώνετε όπως φαίνεται στην παρακάτω εικόνα(Εικόνα5)

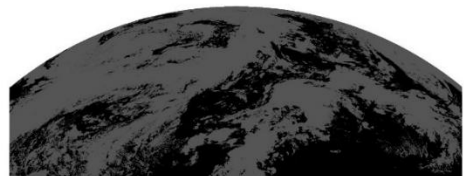


Εικόνα 5 Εμφάνιση γραφικής παράστασης

Τα αποτελέσματα αυτού του προγράμματος είναι το Temp.csv που δημιουργείτε από την συνάρτηση Create_txt() που προαναφέρθηκε . Και ένα αρχείο Figure.jpeg το οποίο δημιουργείτε από την προαναφερόμενη συνάρτηση Save(). Παράδειγμα αυτών φαίνεται στις παρακάτω εικόνες (Εικόνα 6,Εικόνα 7)



Εικόνα 6 Temp.csv αρχείο με τα επιλεγόμενα δεδομένα.



Εικόνα 7 Figure.jpeg παράδειγμα

ΠΗΓΕΣ

- [1] Algorithm Theoretical Basis Document for the Cloud Product Processor of the NWC/GEO
https://www.nwcsaf.org/Downloads/GEO/2018/Documents/Scientific_Docs/NWC-CDOP2-GEO-MFL-SCI-ATBD-Cloud_v2.1.pdf
- [2] NetCDF Documentation
https://docs.unidata.ucar.edu/netcdf/current/index.html#what_is_netcdf

ΠΑΡΑΡΤΗΜΑ :

```
import customtkinter as ctk
import netCDF4 as nc

import numpy as np
import pandas as pd

from matplotlib.pyplot import *
from matplotlib.backends.backend_tkagg import (
    FigureCanvasTkAgg, NavigationToolbar2Tk)

def readNC():

    global keys,data, check_new

    path=entry1.get()
    path=path.replace("'",' ')
    data = nc.Dataset(path)
    keys=data.variables.keys()
    print("opened: ", path)

    if check_new == False:
        global checkBoxes
        checkBoxes = []
        for key in keys:
            if key == 'ny':
                break
            checkBox=ctk.CTkCheckBox(master=frame,text=key)
            checkBox.pack(pady=12,padx=10)
            checkBoxes.append(checkBox)

        button2=ctk.CTkButton(master=frame,text="Create")
        button2.bind("<Button-1>", Get_Data)
        button2.pack(pady=5,padx=50)

    else:
        indx = 0
        for key in keys:
            if key == 'ny':
                break
            checkBoxes[indx].configure(text=key)
            indx+=1

    check_new = True
```

```

def create_txt(plot_data):
#fd = open(f'temp.txt', "w")
for key in plot_data:
values = data.variables[key][:]

df = pd.DataFrame(values)
df=df.fillna(5)
print(df)
df.to_csv('temp.csv', index=False)

print('Done')

def Plot(*args, **kwargs):

data_to_plot=np.genfromtxt('temp.csv', delimiter=',')

canvas.draw()
canvas.get_tk_widget().pack(side=ctk.TOP, fill=ctk.BOTH, expand=1)
button5.pack(pady=5,padx=60)
ax.spines[['right', 'top','left','bottom']].set_visible(False)
ax.axes.get_xaxis().set_visible(False)
ax.axes.get_yaxis().set_visible(False)
ax.imshow(data_to_plot, cmap='gray', vmin=0, vmax=3)

def Get_Data(*args,**kwargs):
answer=[]
global Buttonia
for list in checkBoxes:
answer.append(list.get())
j=0
for key in keys:
if key == 'ny':
break
if answer[j]==1:
plot_data.append(key)
j=j+1

if Buttonia==False:
button3.pack(pady=5,padx=60)
button4.pack(pady=5,padx=60)
Buttonia=True

def Save(fig):
fig.savefig('Figure.jpeg')

if __name__ == '__main__':

ctk.set_appearance_mode("dark")
ctk.set_default_color_theme("dark-blue")

root=ctk.CTk()
root.geometry("1024x800")

check_new = False
Buttonia=False

```

```

plot_data = []

frame=ctk.CTkFrame(master=root)
frame.pack(pady=20,padx=60,fill="both",expand=True)

frame2=ctk.CTkFrame(master=frame)
frame2.pack(side=ctk.RIGHT, pady=20,padx=60,fill="both",expand=True)

fig = matplotlib.pyplot.Figure(figsize=(5, 4), dpi=100)

ax = fig.add_subplot(1,1,1)

canvas = FigureCanvasTkAgg(fig, master=frame2) # A tk.DrawingArea.

label=ctk.CTkLabel(master=frame,text="Open Netcdf File")
label.pack(pady=12,padx=5)

#entry path
entry1=ctk.CTkEntry(master=frame,placeholder_text = "File Path ")
entry1.pack(pady=12,padx=10)

button=ctk.CTkButton(master=frame,text="Open",command=readNC)
button.pack(pady=20,padx=20)

button3=ctk.CTkButton(master=frame,text="Get Data")
button3.bind("<Button-1>",lambda event ,p=plot_data: create_txt(p)) #downloading csv

button4=ctk.CTkButton(master=frame,text="Plot")
button4.bind("<Button-1>", lambda event, ax=ax, canvas=canvas:Plot(ax, canvas))

button5=ctk.CTkButton(master=frame,text="Download Jpeg")
button5.bind("<Button-1>", lambda event, fig=fig :Save(fig))
root.mainloop()

```