88 tane takımyıldızının içerdiği parlak yıldız sayıları, bu yıldızların bulunduğu enlem ve boylam dereceleri excel tablosuna eklenmişti. Excel tablosundaki bu veriler kullanılarak yıldızların tolerans değişkenine göre genel enerjilerini hesaplamak için kullanılan kod:

!pip install hypernetx

import pandas as pd

import hypernetx as hnx

import matplotlib.pyplot as plt

from collections import defaultdict

def hyper\_matrix(H):

    n = H.shape[0]

    matrix = np.zeros((n, n))

    data = H.dataframe

    grouped = defaultdict(set)

    for node, edge in zip(data["nodes"], data["edges"]):

        grouped[node].add(edge)

    nodes = list(grouped.keys())

    for i in range(n):

        for j in range(i + 1, n):

            common\_edges = len(grouped[nodes[i]] & grouped[nodes[j]])

            matrix[i][j] = common\_edges

            matrix[j][i] = common\_edges

    return matrixd

def hyper\_energy(H):

    from numpy import linalg as LA

    m = hyper\_matrix(H)

    e = np.abs(LA.eigvalsh(m))

    return np.sum(e)

Cluster9060 = []

Cluster6030 = []

Cluster3000 = []

Cluster0030 = []

Cluster3060 = []

Cluster6090 = []

for name, enlem in zip(data["NAME"], data["ENLEM"]):

    value = int(eval(enlem.split(':')[0]))

    if  60 <= value <= 90:

        Cluster6090.append([name, value])

    if  30 <= value < 60:

        Cluster3060.append([name, value])

    if  0 <= value < 30:

        Cluster0030.append([name, value])

    if  -30 <= value < 0:

        Cluster3000.append([name, value])

    if  -60 <= value < -30:

        Cluster6030.append([name, value])

    if  -90 <= value < -60:

        Cluster9060.append([name, value])

EdgeString = ["Cluster9060", "Cluster6030", "Cluster3000", "Cluster0030", "Cluster3060", "Cluster6090"]

Clusters = []

Names = []

for cluster in EdgeString:

    for node in eval(cluster):

        Names.append((node[0],node[1]))

        Clusters.append(cluster)

Tolerans = 10

e1 = []

for u in Cluster9060:

    if  u[1] in range(-60-Tolerans, -60):

        e1.append((u[0], u[1]))

for v in Cluster6030:

    if  v[1] in range(-59, -59+Tolerans):

        e1.append((v[0], v[1]))

e2 = []

for u in Cluster6030:

    if  u[1] in range(-30-Tolerans, -30):

        e2.append((u[0], u[1]))

for v in Cluster3000:

    if  v[1] in range(-29, -29+Tolerans):

        e2.append((v[0], v[1]))

e3 = []

for u in Cluster3000:

    if  u[1] in range(-Tolerans, 0):

        e3.append((u[0], u[1]))

for v in Cluster0030:

    if  v[1] in range(1, 1+Tolerans):

        e3.append((v[0], v[1]))

e4 = []

for u in Cluster0030:

    if  u[1] in range(30-Tolerans, 30):

        e4.append((u[0], u[1]))

for v in Cluster3060:

    if  v[1] in range(31, 31+Tolerans):

        e4.append((v[0], v[1]))

e5 = []

for u in Cluster3060:

    if  u[1] in range(60-Tolerans, 60):

        e5.append((u[0], u[1]))

for v in Cluster6090:

    if v[1] in range(61, 61+Tolerans):

        e5.append((v[0], v[1]))

n = 1

for newedge in [e1, e2, e3, e4, e5]:

    for name in newedge:

        Names.append(name)

        Clusters.append(f"e{n}")

    n+=1

newdata = pd.DataFrame({"Edges":Clusters, "Nodes":Names})

H = hnx.Hypergraph(newdata, edge\_col = "Edges", node\_col="Nodes")

NewEdgeString = ["Cluster9060","e1", "Cluster6030", "e2", "Cluster3000", "e3", "Cluster0030","e4", "Cluster3060", "e5", "Cluster6090"]

for cluster in NewEdgeString:

    print(f"{cluster} içerisinde ki yıldızlar:\n")

    for star in eval(cluster):

        print(star)

    print("\n")

for cluster in NewEdgeString:

    print(f"{cluster} içerisinde ki yıldız sayısı = {len(eval(cluster))}:\n")