K-means, 일부 나무 데이터

import pandas as pd

from sklearn.cluster import KMeans

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.pipeline import make\_pipeline

from sklearn.preprocessing import StandardScaler

import numpy as np

from mpl\_toolkits.mplot3d import Axes3D

import pcl

pc = pcl.load("/content/result2\_.pcd") # "pc.from\_file" Deprecated

pc\_array = pc.to\_array() # pc to Numpy

#cloud = pcl.load\_XYZRGBA("tabletop.pcd")

# Read Tree Sample data

data = pd.DataFrame(pc\_array)

data.columns=['X','Y','Z']

print(data)

# we can find appropriate the number of clusters with Inertia

ks = range(20, 40) # range denpend on k's number

inertias = []

for k in ks:

model = KMeans(n\_clusters=k)

model.fit(data)

inertias.append(model.inertia\_)

print(inertias)

# Plot ks vs inertias

plt.plot(ks, inertias, '-o')

plt.xlabel('number of clusters, k')

plt.ylabel('inertia')

plt.xticks(ks)

plt.show()

# create model and prediction

k\_Num = int(input("input k: "))

model = KMeans(n\_clusters=k\_Num, algorithm='auto')

model.fit(data)

predict = pd.DataFrame(model.predict(data))

predict.columns=['predict']

# concatenate labels to df as a new column

r = pd.concat([data,predict],axis=1)

max = -1

for index in r.predict:

if max < index:

max = index

print("Num of K: ", max)

print(r)

# Clustering data visualization

#2D plot

plt.scatter(r['X'], r['Y'], r['Z'], c=r['predict'], alpha=0.5)

# 3d plot

fig = plt.figure(figsize=(10, 10))

ax = fig.add\_subplot(111, projection='3d') # Axe3D object

ax.scatter(r['X'], r['Y'], r['Z'], c=r['predict'], alpha=0.5)

centers = pd.DataFrame(model.cluster\_centers\_,columns=['X','Y','Z'])

center\_x = centers['X']

center\_y = centers['Y']

center\_z = centers['Z']

# if you want to see 2D centroid plot, delete below '#'

# plt.scatter(center\_x,center\_y,center\_z, marker='D',c='b')

# if you want to see 3D centroid plot

ax.scatter(center\_x,center\_y,center\_z,s=50, marker='D',c='r')

plt.show()

X Y Z

0 5.73202 -0.74213 2.00003

1 -5.85224 -3.51000 2.00003

2 -1.55024 -6.14023 2.00004

3 -0.15085 -2.35509 2.00004

4 5.57676 5.75816 2.00004

... ... ... ...

46770 -6.55157 0.85491 2.49994

46771 -0.05673 -2.31840 2.49994

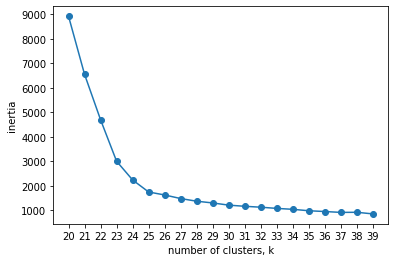
46772 1.28519 4.79566 2.49997

46773 1.36988 2.13100 2.49999

46774 1.40314 2.09217 2.50002

[46775 rows x 3 columns]

[8918.818638159231, 6546.632383378622, 4693.453648785334, 3002.0173563139797, 2239.235594644138, 1748.1479831015743, 1626.6173074734586, 1480.4413028615622, 1372.078249685934, 1301.4495006146878, 1212.142640566387, 1164.9938732442874, 1127.6946357940067, 1079.205219673388, 1041.4154036739021, 982.8324826731722, 951.5050924656799, 915.9172447472489, 920.4560019976028, 853.9163089894693]



input k: 26

Num of K: 25

X Y Z predict

0 5.73202 -0.74213 2.00003 12

1 -5.85224 -3.51000 2.00003 2

2 -1.55024 -6.14023 2.00004 6

3 -0.15085 -2.35509 2.00004 3

4 5.57676 5.75816 2.00004 8

... ... ... ... ...

46770 -6.55157 0.85491 2.49994 7

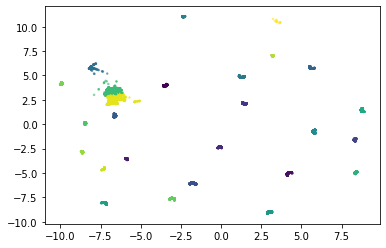
46771 -0.05673 -2.31840 2.49994 3

46772 1.28519 4.79566 2.49997 10

46773 1.36988 2.13100 2.49999 4

46774 1.40314 2.09217 2.50002 4

[46775 rows x 4 columns]



문구, 연필, 담장이(가) 표시된 사진

자동 생성된 설명