```
Timer unit: 1e-06 s

Total time: 621.589 s
File: /tmp/ipykernel_6482/1333631649.py
Function: fit_predict at line 21

Line #      Hits         Time  Per Hit   % Time  Line Contents
==============================================================
    21                                           def fit_predict(self, k_num:int = 3, max_step:int = 500, conv_threshold: float = 1e-5)
    22                                             '''
    23                                             Membuat model KMeans dengan K tertentu. Akan mengkembalikan hasil prediksi cluster.
    24                                             Poin kluster akan disimpan pada variable point
    25                                             '''
    26                                             # Setting up cluster arry for every record
    27        10       3615.0    361.5      0.0      cluster = np.zeros(len(self.training_arr))
    28
    29                                             # normalize data
    30        10      33692.0   3369.2      0.0      data = self.__normalize_data__(self.training_arr)
    31
    32                                             # Initialize centroid using KMeans++
    33        10  191813860.0 19181386.0     30.9      point = self.__initialize_centroids__(data, k_num)
    34
    35                                             # Setup convergence and counter
    36        10         31.0      3.1      0.0      convergence = False
    37        10          9.0      0.9      0.0      step = 0
    38
    39       102        123.0      1.2      0.0      while not convergence and (step < max_step):
    40        92        115.0      1.2      0.0        initial_point = point
    41        92  345325733.0 3753540.6     55.6         distance = self.__calculate_distance__(data, point)
    42        92   39661540.0 431103.7      6.4        cluster = self.__clustering__(distance)
    43        92   31295640.0 340170.0      5.0        new_point = self.__point_nomralization__(data, point, cluster)
    44        92      21643.0    235.2      0.0        convergence = self.__convergence_check__(initial_point, new_point, conv_threshold)
    45
    46        92        101.0      1.1      0.0        if convergence:
    47        10          7.0      0.7      0.0          point = new_point
    48        10       1450.0    145.0      0.0          print("It's convergence!")
    49                                               else:
    50        82         48.0      0.6      0.0          point = new_point
    51        82         61.0      0.7      0.0          step += 1
    52        82      15979.0    194.9      0.0          print("STEP:", step)
    53
    54
    55        10   13038992.0 1303899.2      2.1       self.inertia = self.__calculate_inertia__(data, cluster, point)
    56        10     376837.0  37683.7      0.1      self.point = self.__denormalize_point__(point, self.training_arr)
    57        10         22.0      2.2      0.0      return cluster
```