```
Timer unit: 1e-06 s

Total time: 56.0192 s
File: /tmp/ipykernel_24079/2577465532.py
Function: fit_predict at line 18

Line #      Hits         Time  Per Hit   % Time  Line Contents
==============================================================
    18                                           def fit_predict(self, k_num:int = 3, max_step:int = 500, conv_threshold: float = 1e-5)
    19                                               '''
    20                                               Membuat model KMeans dengan K tertentu. Akan mengembalikan hasil prediksi cluster.
    21                                               Poin kluster akan disimpan pada variable point
    22                                               '''
    23                                               # Setting up cluster arry for every record
    24         5       1602.0    320.4      0.0     cluster = np.zeros(len(self.training_arr))
    25
    26                                               # normalize data
    27         5      14979.0   2995.8      0.0     data = self.__normalize_data__(self.training_arr)
    28
    29                                               # Initialize centroid using KMeans++
    30         5   12884278.0 2576855.6    23.0      point = self.__initialize_centroids__(data, k_num)
    31
    32                                               # Setup convergence and counter
    33         5         14.0      2.8      0.0     convergence = False
    34         5          5.0      1.0      0.0     step = 0
    35
    36        23         29.0      1.3      0.0     while not convergence and (step < max_step):
    37        18         20.0      1.1      0.0       initial_point = point
    38        18   25677209.0 1426511.6    45.8        distance = self.__calculate_distance__(data, point)
    39        18    6562904.0 364605.8    11.7       cluster = self.__clustering__(distance)
    40        18    5293422.0 294079.0     9.4       new_point = self.__point_nomralization__(data, point, cluster)
    41        18       2507.0    139.3      0.0       convergence = self.__convergence_check__(initial_point, new_point, conv_threshold)
    42
    43        18         35.0      1.9      0.0       if convergence:
    44         1          0.0      0.0      0.0         point = new_point
    45         1        114.0    114.0      0.0         print("It's convergence!")
    46                                                 else:
    47        17         10.0      0.6      0.0         point = new_point
    48        17         13.0      0.8      0.0         step += 1
    49        17       4013.0    236.1      0.0         print("STEP:", step)
    50
    51
    52         5    5416397.0 1083279.4     9.7      self.inertia = self.__calculate_inertia__(data, cluster, point)
    53         5     161626.0  32325.2      0.3     self.point = self.__denormalize_point__(point, self.training_arr)
    54         5         11.0      2.2      0.0     return cluster
```