前回のドキュメント

与えられていた貪欲法の上で

①

(step5\_greedy\_two\_opts.pyの11から15行まで)

総距離を計算する関数total\_distanceを定義して、

ツアー(eg.[0, 3, 2, 1])の巡回順序通りに前後二つのcityの距離を計算して足し、最後は総距離を返す。

②

(step5\_greedy\_two\_opts.pyの38行から51行)

2-opt法を使って、貪欲法の結果を改善する2\_opt関数を定義します。

二重for文のループを使った部分的に巡回順序を逆にして、例えば[0,3,2,1]の１番目から３番目を逆にして[0,1,2,3]にする。

そして総距離を計算する関数を使って総距離が縮んだら新しい順序を際よする。

結果を見ると

Challenge 0 : Total distance = 3418.1015991327126

Challenge 1 : Total distance = 3832.2900939051992

Challenge 2 : Total distance = 5232.964923917401

Challenge 3 : Total distance = 9261.012772746048

Challenge 4 : Total distance = 11591.844674406604

Challenge 5 : Total distance = 22314.04512090243

Challenge6:ずっと実行していたが結果が出ない。

貪欲法のサンプルの０から５までの結果よりは少し良くなっています。

0: 3,418.10

1: 3,832.29

2: 5,449.44

3: 10,519.16

4: 12,684.06

5: 25,331.84