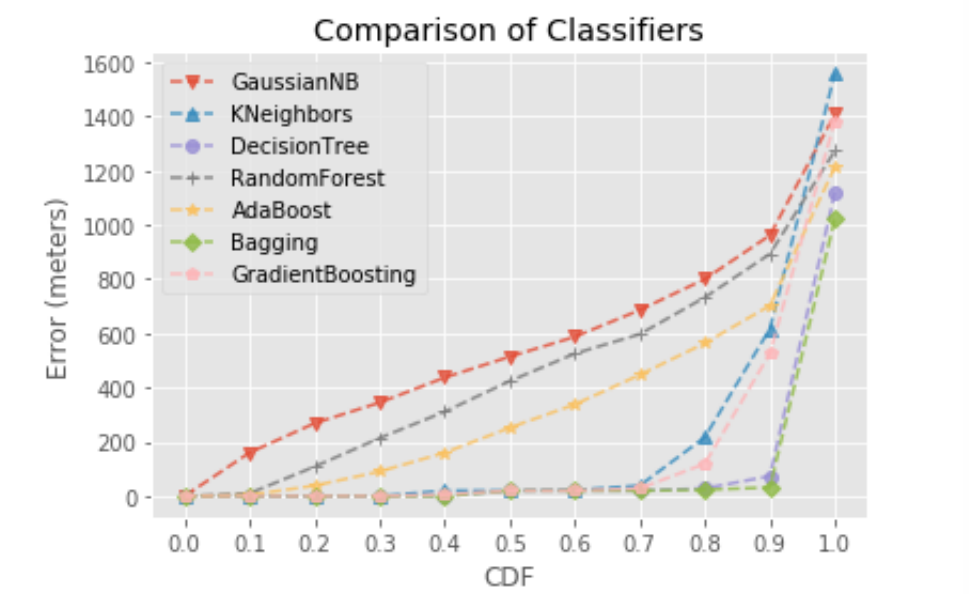
DM-HW3-Q1

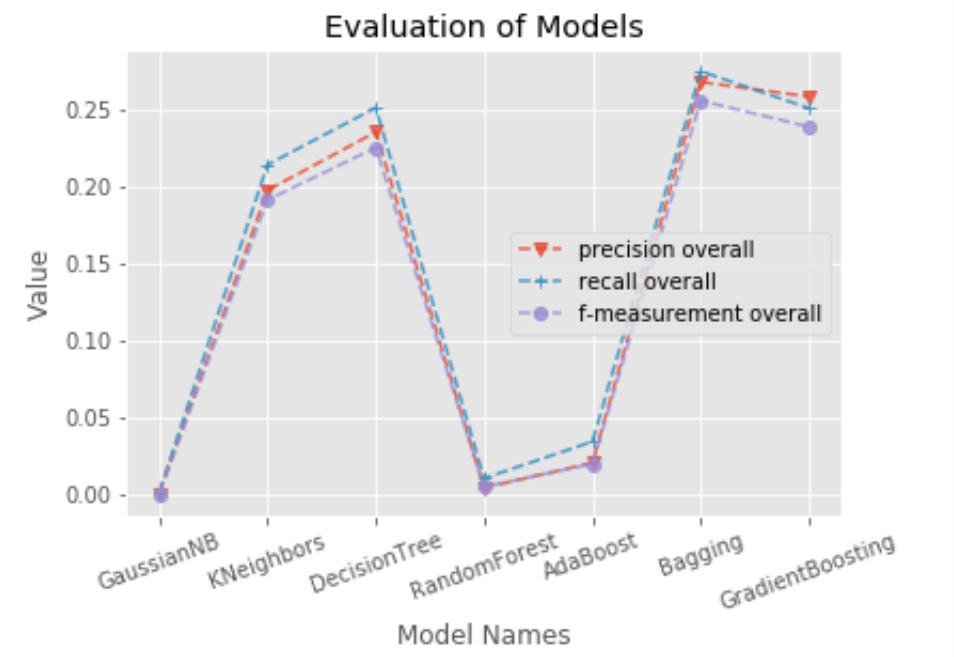
a

1. 代码运行结果拷贝

1. 各分类器下平均误差概率分布图



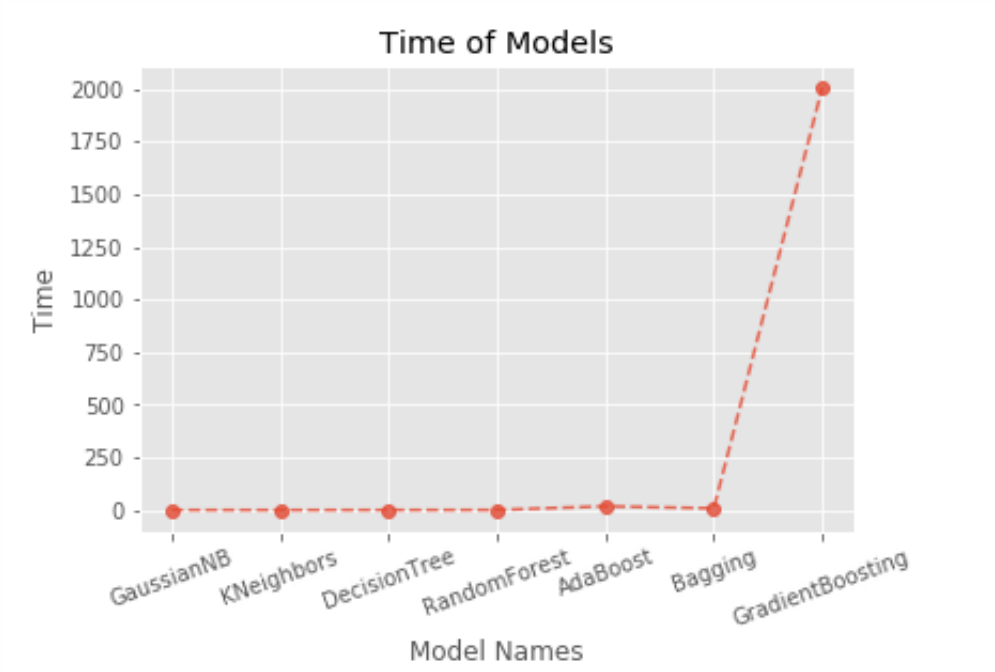
2. 各分类器下的precision/recall/f-measurement（overall）



3. 各分类器下的precision/recall/f-measurement（each grid）

```
{'AdaBoost': {'f-measurement foreach': array([ 0.        ,  0.        ,  0.        ,  0.28571429,  0.22222222,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.36363636,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.15384615,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.12698413,  0.        ,  0.        ,  0.        ,
        0.        ,
        0.46153846,  0.        ,  0.        ,  0.        ,  0.        ,
        0.        ,  0.        ,  0.        ,  0.4       ,  0.        ,
        0.        ,  0.        ,  0.18181818,  0.        ,  0.        ,
```

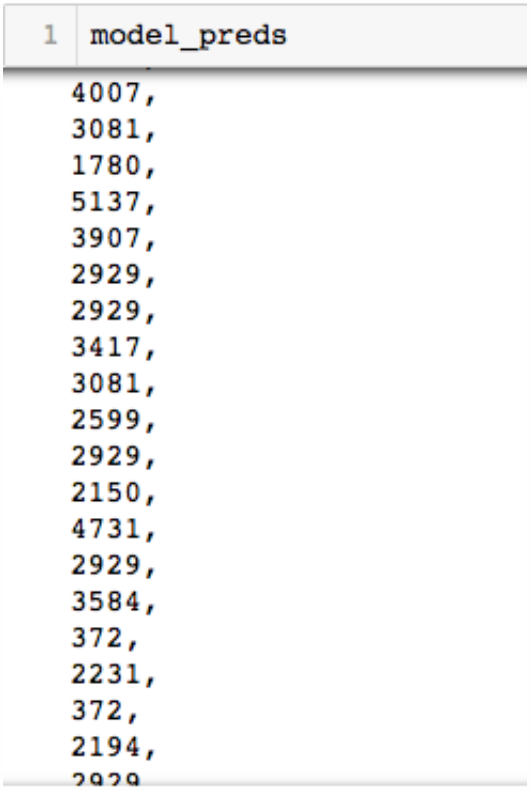
4. 各分类器下运行时间




```
1  model_times
```

```
{'AdaBoost': 17.965240399999992,
 'Bagging': 7.9605032000000051,
 'DecisionTree': 0.27994979999999992,
 'GaussianNB': 0.1829975000000002,
 'GradientBoosting': 2006.8953565999996,
 'KNeighbors': 0.072133200000000119,
 'RandomForest': 0.40618760000000015}
```

4. 部分预测结果

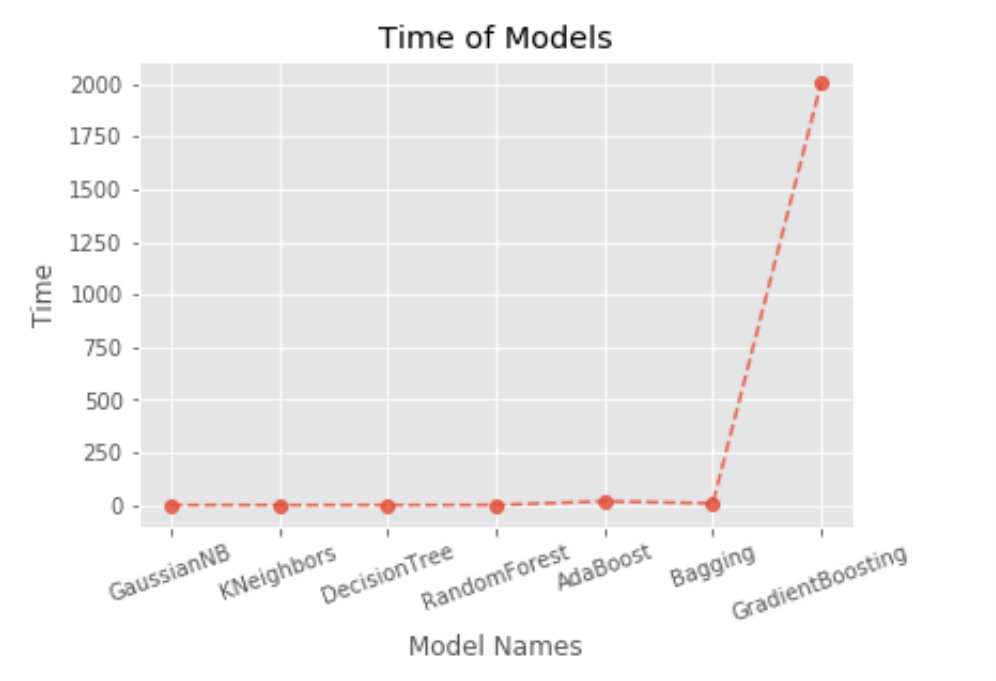


2. 讨论分析部分

1. 以平均误差概率作为标准，KNeighbors/RandomForest/Bagging/GradientBoosting 四种分类器表现都比较好
2. 以 precision/recall/f-measurement 作为标准，也是 yeshiKNeighbors/RandomForest/Bagging/GradientBoosting 四种分类器表现较好
3. GradientBoosting的时间远大于其他分类器

3. 性能比较图表

各分类器下运行时间

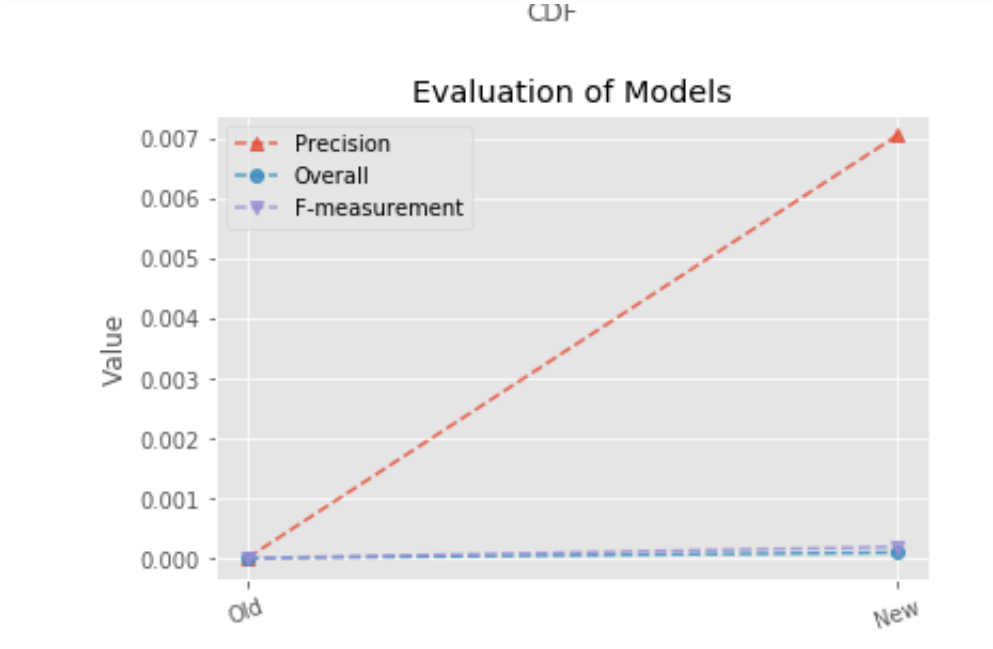


b

1.代码运行结果拷贝

1. 新旧方案对比（平均误差概率分布图）

2. 新旧方案对比（precision/recall/f-measurement）



2. 讨论分析部分

1. 修正算法简述：
    1. 设定最大速度不超过2m/s
    2. 针对a题的计算结果，找到预测结果点时间序列上前一个点，若速度超过2m/s，则使用其前后亮点的中点来替代
2. 结果分析：

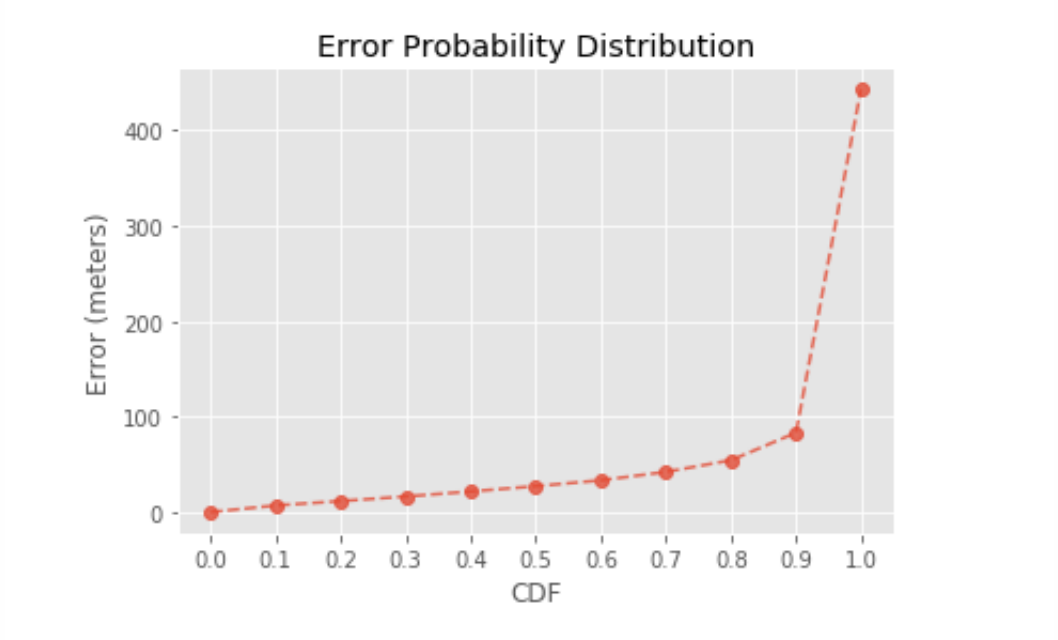
    从上述对比图来看，该修正算法对于修正误差较小的点的作用不大，但可以修正部分误差较大的点

3. 性能比较图表

a题（RandomForest）	b题（RandomForest）
0.4s	1.7s

1. 代码运行结果拷贝

1. RandomForest下的平均误差概率分布



2. 预测结果

```
Out[519]: {(5198,
            16058,
            121.22063899999999,
            31.281872999999997):              IMSI        MRTime   Longitude   Latitude  Num_connected  \
           0    4.600163e+14  1.510221e+12  121.209226  31.284592              5
           1    4.600163e+14  1.510221e+12  121.209226  31.284592              5
           2    4.600163e+14  1.510221e+12  121.209207  31.284641              5
           3    4.600163e+14  1.510221e+12  121.209198  31.284655              5
           4    4.600163e+14  1.510221e+12  121.209197  31.284652              6
           5    4.600163e+14  1.510221e+12  121.209195  31.284671              6
           6    4.600163e+14  1.510221e+12  121.209189  31.284683              6
           7    4.600163e+14  1.510221e+12  121.209187  31.284682              6
           8    4.600163e+14  1.510221e+12  121.209187  31.284682              6
           9    4.600163e+14  1.510221e+12  121.209189  31.284710              6
           10   4.600163e+14  1.510221e+12  121.209189  31.284710              7
           11   4.600163e+14  1.512005e+12  121.213481  31.288796              2
           12   4.600163e+14  1.512005e+12  121.213534  31.288770              2
           13   4.600163e+14  1.512005e+12  121.213581  31.288735              4
           14   4.600163e+14  1.512005e+12  121.213682  31.288658              7
```

3. 运行时间

```
7.920025699997856
```

2. 讨论分析部分

1. 由于经纬度是连续的，所以选择回归模型而不是分类模型
2. 从RandomForest下的平均误差概率分布来看，结果好于a，原因可能是将地图栅格化后标记后不能准确的确定经纬度，并且对于地理位置这种连续的数据使用回归更好

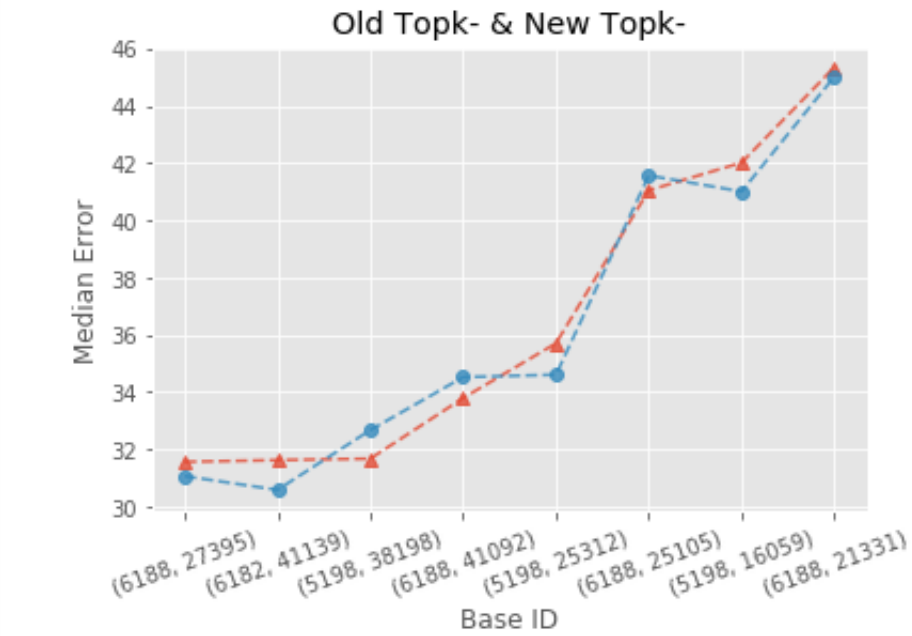
3. 性能比较图图表

a题（RandomForest）	c题（RandomForest）
0.4s	7.9s

d

1. 代码运行结果拷贝

1. c题与d题的 Top k- 中位误差比较



2. Top k- 的预测结果（部分）

```
{(5198,
  16059,
  121.209767,
  31.284987): [[(-0.0051915534288347583, 0.0028435057623187763),
   (-0.0023088244624694678, 0.0064231884502479629),
   (-0.0052692871149855759, 0.0021030453113748508),
   (-0.005142799396057789, 0.0032712558174612417),
   (-0.0055232216178096568, 0.0051962902121427236),
   (-0.0051600821894442726, 0.00534693193400984),
   (-0.0019134583224657 69, 0.0062856627060498861),
   (-0.00094621232589303748, 0.0064231884502479629),
   (-0.0044051095140431769, 0.0040117364653969659),
   (-0.005315087534152255, 0.0024648513066543136),
   (-0.0037976602281783122, 0.006677696034161457),
   (-0.0045364718823025559, 0.0057669664429085355),
   (-0.0019919059514286967, 0.0062904778191573071),
   (-0.0048503093708087792, 0.0037726600707765428),
   (0.00037208113332940651, 0.0064231884502479629),
   (-0.0036673890799490 96, 0.0031296407934460306),
   (-0.0039954213669885271, 0.0075571561884868223),
   (-0.0035758443029577 07, 0.0064655772580121638),
   (-0.0053250479043701615, 0.0021034956763747487),
   (-0.0051224994417401705, 0.0035535094723384 23),
   (-0.0049848865874803352, 0.0018783050118886904),
   (-0.0040591034567175367, 0.0076024890207331752),
   (-0.00092476276378898939, 0.0062904778191573071),
   (-0.0047045709723884934, 0.0037010618649593379),
   (-0.0052415234839220193, 0.0021218735659580936),
   (-0.0043773522452035833, 0.0046819349124282792),
```

3. 运行时间

```
16.757375000001048
```

2. 讨论分析部分

1. 从c题与d题的 Top k- 中位误差比较来看，该处理对提高精确度几乎没有起到帮助，甚至有些基站的中位置反而加大了
2. 可能是不同基站之间有不可忽视的差别，不能简单将结果好的基站数据融入结果差的中去

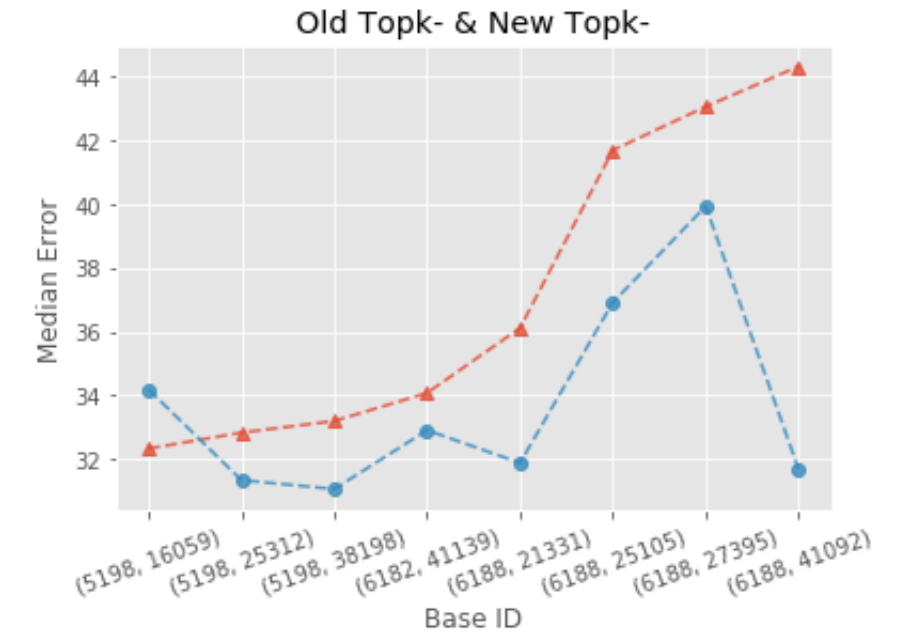
3. 性能比较图图表

c题（RandomForest）	d题（RandomForest）
7.9s	16.8s

e

运行结果拷贝

1. c题与e题的 Top k- 中位误差比较



2. 运行时间

```
86.11301800000001
```

2. 讨论分析部分

1. 从c题与d题的 Top k- 中位误差比较来看，该处理对提高精确度起到一定帮助（大部分基站的中位误差都变小了）
2. 使用与基站相似且结果较好的数据进行融合，在一定程度上能提高模型的精确度

3. 性能比较图图表

c题（RandomForest）	**d题（RandomForest）**
7.9s	86.1s