特征工程

|  |  |  |  |
| --- | --- | --- | --- |
| 特征名 | Key | 描述 | 维度 |
| Week | Date | 样本日期所对应周几的one-hot编码，以及最后一维是否是假日 | 1-8 |
| TID | Time\_ID | 时间片ID | 9 |
| FPOI | District | 区域的POI特征 | 10-185 |
| FineArriveSelf | District, Date,  Time\_ID | 样本时间片前半个小时中，按照5分钟间隔滑动大小为10分钟的窗口，统计从本地到本地有司机接单订单数 | 186-190 |
| FineArrive | District, Date,  Time\_ID | 样本时间片前半个小时中，按照5分钟间隔滑动大小为10分钟的窗口，统计到本地有司机接单的订单数 | 191-195 |
| FineGap | District, Date,  Time\_ID | 样本时间片前半个小时中，按照5分钟间隔滑动大小为10分钟的窗口，统计从本地出发没有司机接单的订单数 | 196-200 |
| FineDemand | District, Date,  Time\_ID | 样本时间片前半个小时中，按照5分钟间隔滑动大小为10分钟的窗口，统计从本地的总订单数 | 201-205 |
| FWeatherOHE | Date | 对天气进行OHE映射，温度，PM2.5 | 206-214 |
| FDTGap | District,  Time\_ID | 前21天，每天的gap值，求出的均值，中位数，标准差，最小值，最大值 | 215-219 |
| FDTDemand | District,  Time\_ID | 前21天，每天的Demand值，求出的均值，中位数，标准差，最小值，最大值 | 220-224 |
| FDTGapByHoliday | District,  Time\_ID | 前21天，每天的Gap值按照是否是假日分开求出的均值，中位数，标准差，最小值，最大值 | 225-234 |
| FDTSupply | District,  Time\_ID | 前21天，每天的供应值按照是否是假日分开求出的均值，中位数，标准差，最小值，最大值 | 235-239 |
| FDateDistrict | District | 前21天， 每天每个区域总gap数，求均值，中位数，标准差，最小值，最大值 | 240-244 |
| FDateTimeGap | Time\_ID | 前21天，每天每时间片所有区域总gap数，求均值，中位数，标准差，最小值，最大值 | 245-249 |
| FHolidayTimeGap | Time\_ID | 前21天，每天每个时间片所有区域gap数，按照是否是节假日分别求均值，中位数，标准差，最小值，最大值 | 250-259 |
| FWeekTimeGap | WeekDay,  Time\_ID | 前21天，每天每个时间片总gap数，按照是周几分别求均值，中位数，标准差，最小值，最大值 | 260-264 |
| FHolidayDistrictGap | Time\_ID | 前21天，每天每个区域总gap数，按照是否是节假日分别求均值，中位数，标准差，最小值，最大值 | 265-274 |
| FWeekDistrictGap | WeekDay,  Time\_ID | 前21天，每天每个区域总gap数，按照是周几分别求均值，中位数，标准差，最小值，最大值 | 275-279 |
| FPrefixGap | District,  Date,  Time\_ID | 当前时间片往前2,4,6,8,10,…,30分钟累计的gap数 | 280-294 |
| FineTimeGap | Date,  Time\_ID | 样本时间片前半个小时中，按照5分钟间隔滑动大小为10分钟的窗口，统计出所有地区gap总数 | 295-299 |
| FineGapStat | Date,  Time\_ID | 对FineTimeGap的每个时间片所对应的值求均值，中位数，标准差，最小值，最大值 | 300-304 |
| FTrafficTotal |  | 每个区域路况信息中四个level数的总数 | 307 |

模型选取

使用算法工具：xgboost

自定义损失函数：

参数选择：

{

"objective": "reg:linear",

"nthread" : 15,

"eta" : 0.05,

"max\_depth":5,

"min\_child\_weigth": 10,

"gamma" : 0,

"subsample" : 0.8,

"colsample\_bytree" : 0.9,

"n\_round" :3000

}

模型融合：

在上述参数的基础之上，随机化max\_depth, eta, subsample, colsample\_bytree, seed等参数。

params['max\_depth'] = params['max\_depth'] + random.randint(-1, 1)

params['eta'] = params['eta'] + (random.random() - 0.5) / 50

params['subsample'] = params['subsample'] + (random.random() - 0.5) / 5

params['colsample\_bytree'] = params['colsample\_bytree'] + (random.random() - 0.5) / 5

params['seed'] = int(time.time())

训练出25个xgboost模型，最后预测结果取这24个模型输出结果的平均值。