# 模型环境

所用代码均在python3.6下运行.

python 3.6

joblib 0.12.5

matplotlib 2.2.2

numpy 1.15.2

pandas 0.23.4

scikit-learn 0.20.0

tensorflow 1.8.0

# 合并源文件

将源文件的2016年至2017年数据合并为一份csv格式数据集，并保存至DataSet.csv文件中，供后续分析使用。

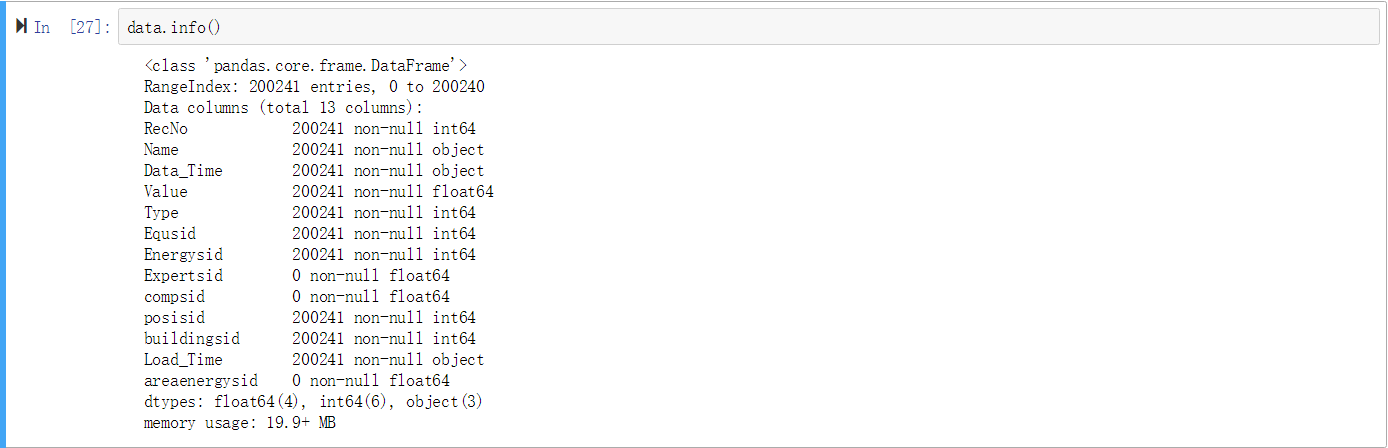
# 数据探索和清洗

大部分操作均在jupyter notebook中执行。最初我们尝试了区分电机、区分消纳类型这两种方式，分别对不同的电机或不同的消纳类型以半小时或一天为刻度建模，但是效果都很不理想，并且时间损耗极大。后来将6台消纳的负荷合并，每半小时只有一行数据或每天只有一行数据，重新进行数据分析和建模过程。

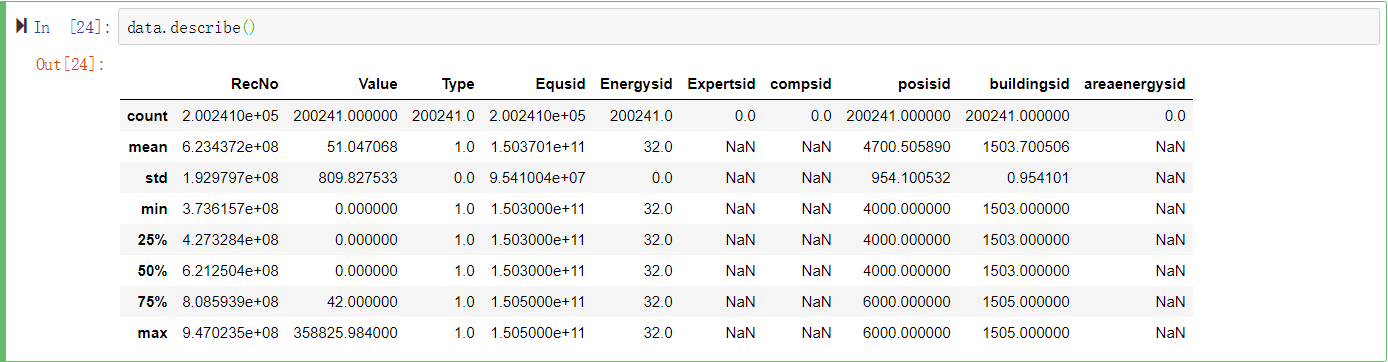
|  |  |  |  |
| --- | --- | --- | --- |
|  | 区分6台电机 | 区分3类消纳 | 合并所有消纳 |
| 半小时为刻度 | 不理想 | 不理想 | 不理想 |
| 一天为刻度 | 不理想 | 不理想 |  |

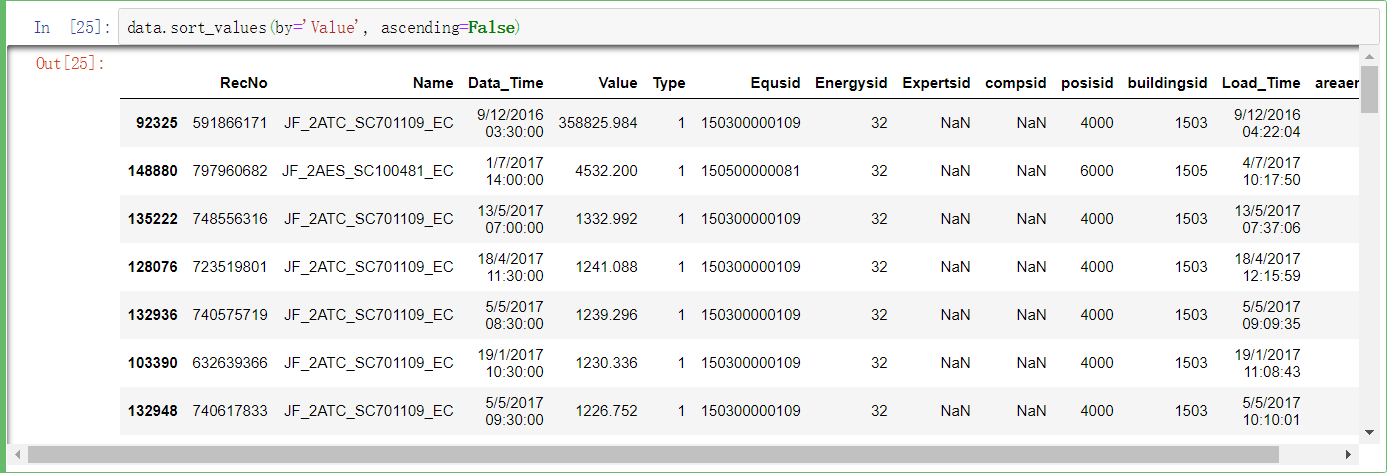
#### 查看数据信息

所有行数据格式统一。注意到‘compsid’, ‘Expertsid’, ‘areaenergysid’列为空，理应删除。



#### 查看数据描述

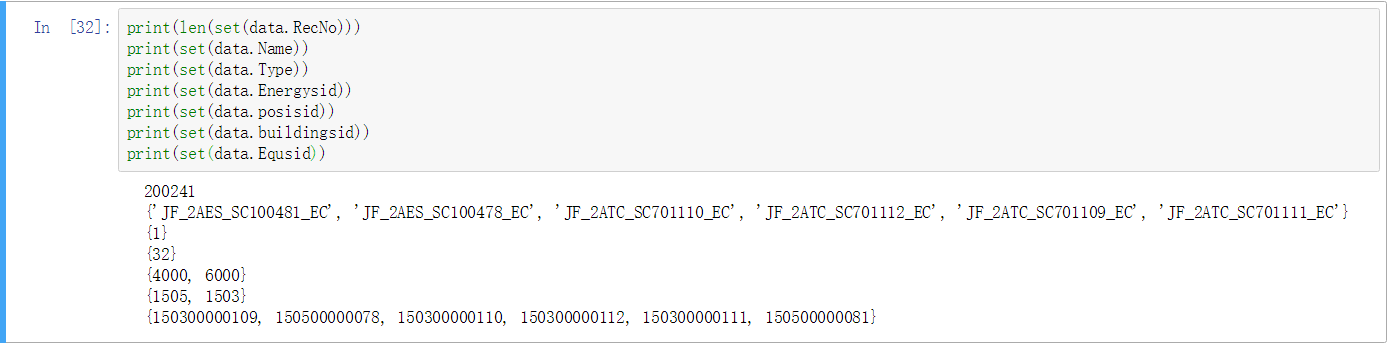
明显注意到Value的最大值居然高达35W，疑似异常数据；同时数据的总行数为200241行，简单计算可知若每半个小时收集6行数据，理论上应有210528行数据，因此数据集应该存在缺失行数据的现象。‘Type’和‘Energysid’列的标准差为0，说明无二值，也应删除。

从排序结果可以肯定，35W为异常数据（2016.12月），并和公司人员确认，同时得知数据中无冲击负荷。

#### 删除无效列

从结果来看，RecNo列为标记类型数据，Name为电机标记，posisid与buildingsid为建筑标记，均为无效数据，只需要保留’Equsid’列即可。’Load\_Time’为生成数据的时间，对预测Value没有帮助，也可以删除。’Equsid’列数值过大，将其重新标记为123456，方便后续处理。

代码文件：createDataSet.py



#### 拆分时间数据

‘Data\_Time’为时间序列，间隔为半小时，将其拆解为’Year’, ‘Month’, ‘Day’, ‘Hour’, ‘Half’共5列，其中’Half’标记是否为半小时，整点为0，半点为1。

拆分时间数据之后，数据文件更新为DataSet1.0.csv。

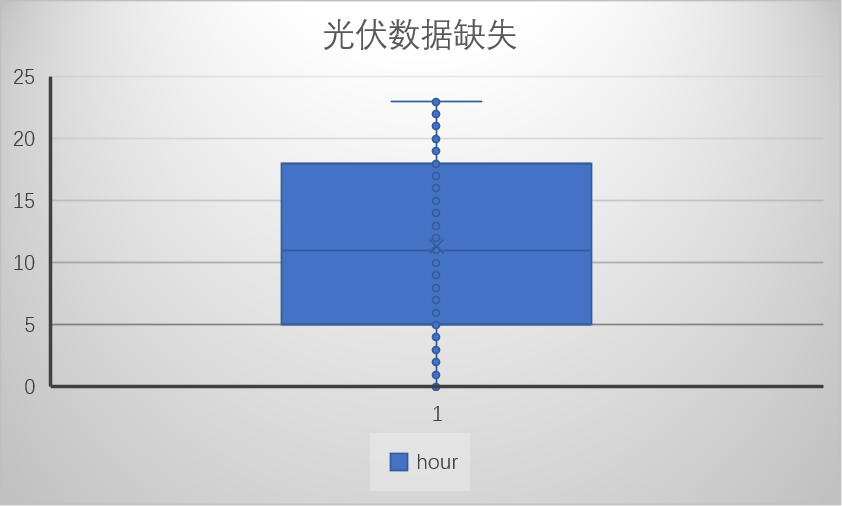
代码文件：createDataSet.py

#### 填补行数据缺失

利用dataAnalysis.py文件中的detect\_missing\_value()函数收集缺失的数据，保存在lose\_data.csv文件中。结果显示共有125天缺失数据，有7个小时是完全没有数据的（2016.5.1 20:30~23:30，2017.5.14 14:00~15:00， 2017.10.23 00:00~03:00），有3385个半点总共10203行数据缺失。

将数据扩充为标准的210528行，每半个小时均有6行数据，所有补齐的’Value’列暂时空缺。这个过程中注意到，数据集存在连续几个小时都缺失数据的现象，疑似停电或检修。缺失数据中大部分为某半点时刻光伏数据缺失，通常情况为3台机组同时缺失数据，缺失的时间点几乎全天各个时段都有。

填补缺失数据之后数据文件更新为DataSet2.0.csv。



代码文件：fillRowData.py

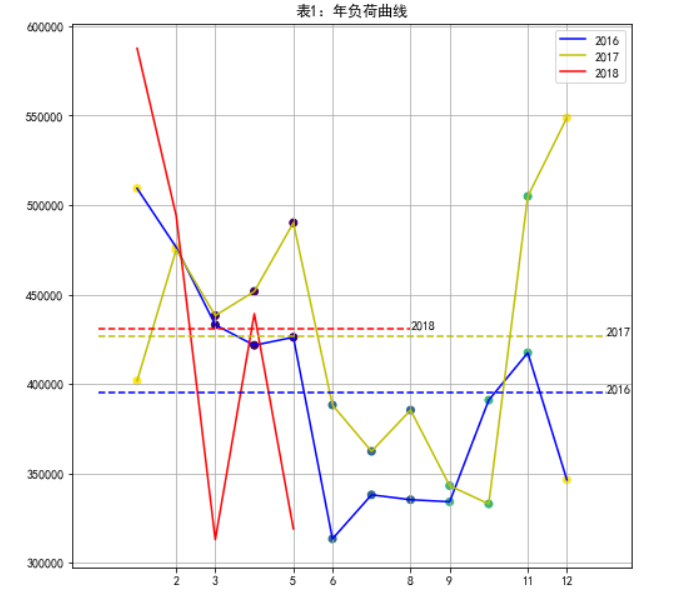
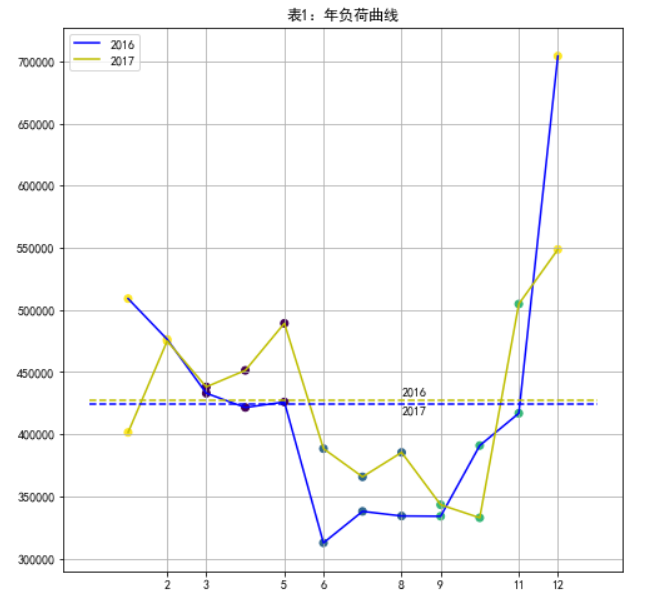
#### 分析负荷曲线

对DataSet2.0.csv分析负荷曲线。负荷统计为6台消纳的总数据。

代码文件：dataAnalysis.py

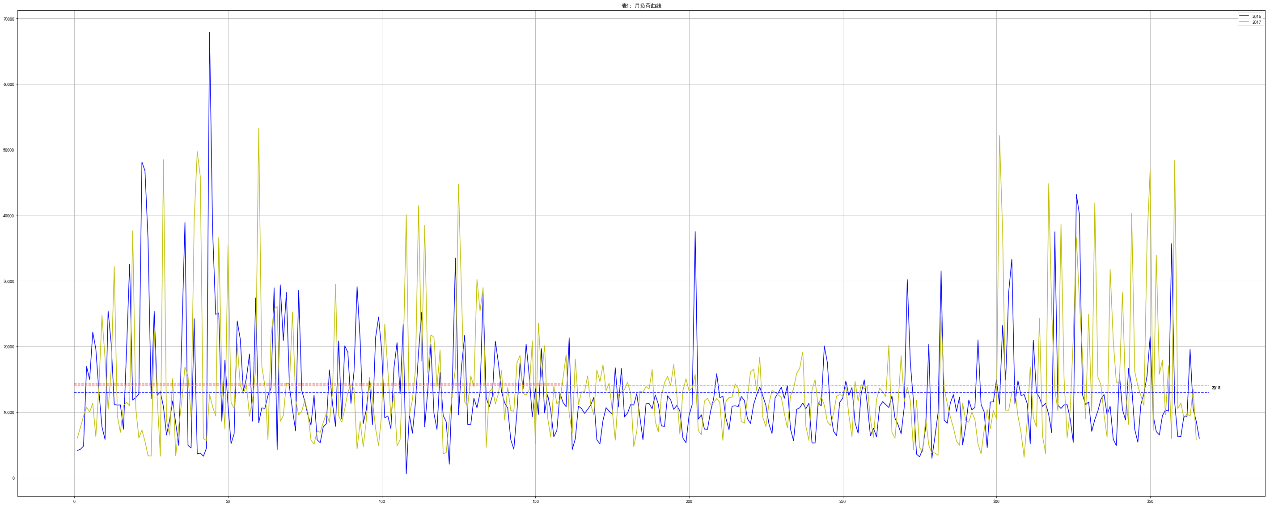
##### 1）年负荷曲线

暂不考虑12月的异常数据，负荷曲线没有明显的年周期性。第一张表未处理任何异常值，第二张表处理了异常值并加入了18年数据。夏季负荷为全年最低，冬季负荷为全年最高。2年中6~9月的负荷都明显低于全年均值。



##### 2）月负荷曲线

红线为后续添加进去的2018.1~5月用电量均值。这里将12月的异常值35W数据归0，月负荷曲线并未显示出明显的周期性。可能存在滞后性的规律。



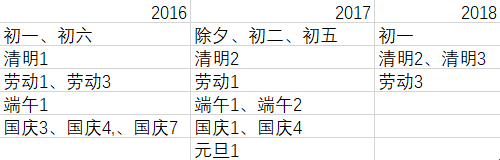
##### 3）日负荷曲线

日负荷曲线较为复杂，但是也没有表现出明显的周期性。数据中并未给出检修/大功率作业的时间标记。

##### 4）节假日负荷曲线

节假日负荷曲线普遍偏低，但是也有一些异常点如下。下图中的几个点的负荷值是明显高于周围几个点的。

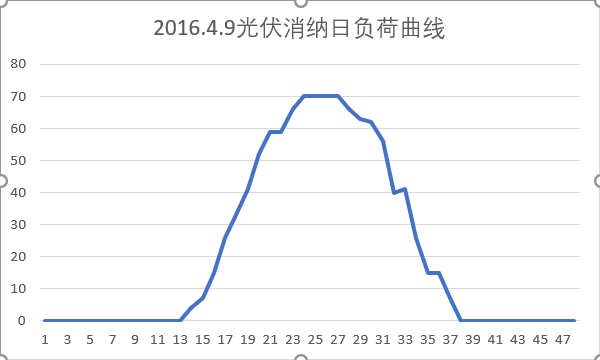




##### 5）结论

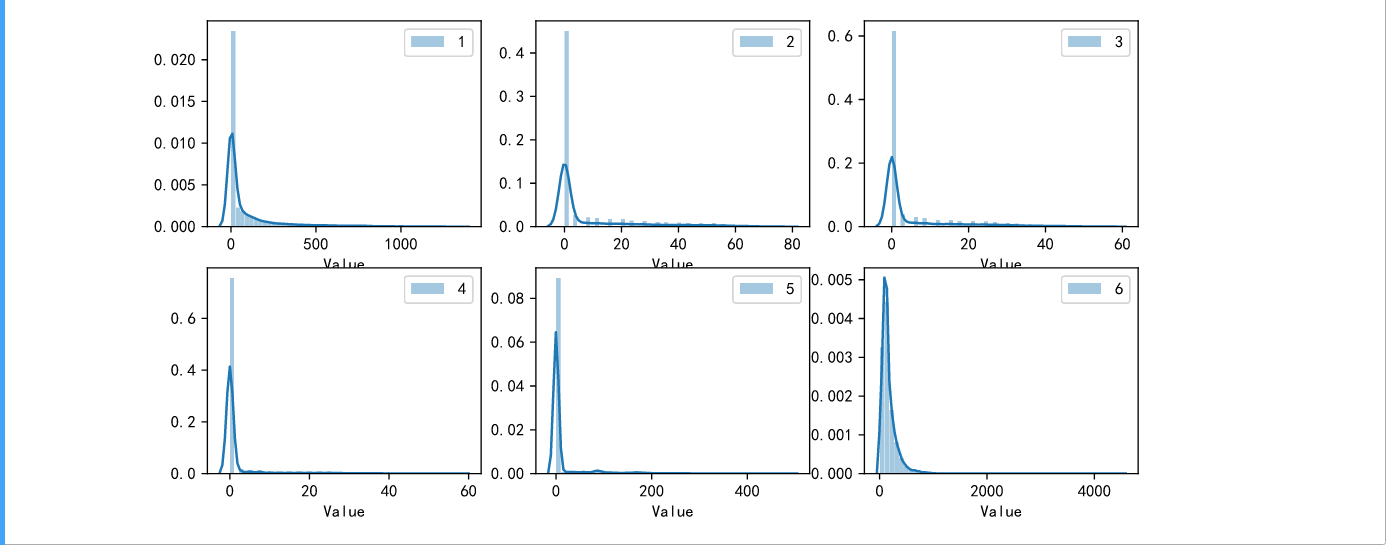
原始数据集中，负荷没有表现出明显的时间上的规律。时间序列法无法使用。考虑根据不同的消纳类型绘制负荷曲线，观察是否有周期性。

通过3种不同的消纳类型的负荷曲线可知，光伏消纳有明显的日周期性，全年每天的0~5点，19~24点光伏消纳皆为0，日负荷曲线呈现明显的二次函数形状。但是单独预测光伏消纳意义不大，其用电量只占总体的极小比例。

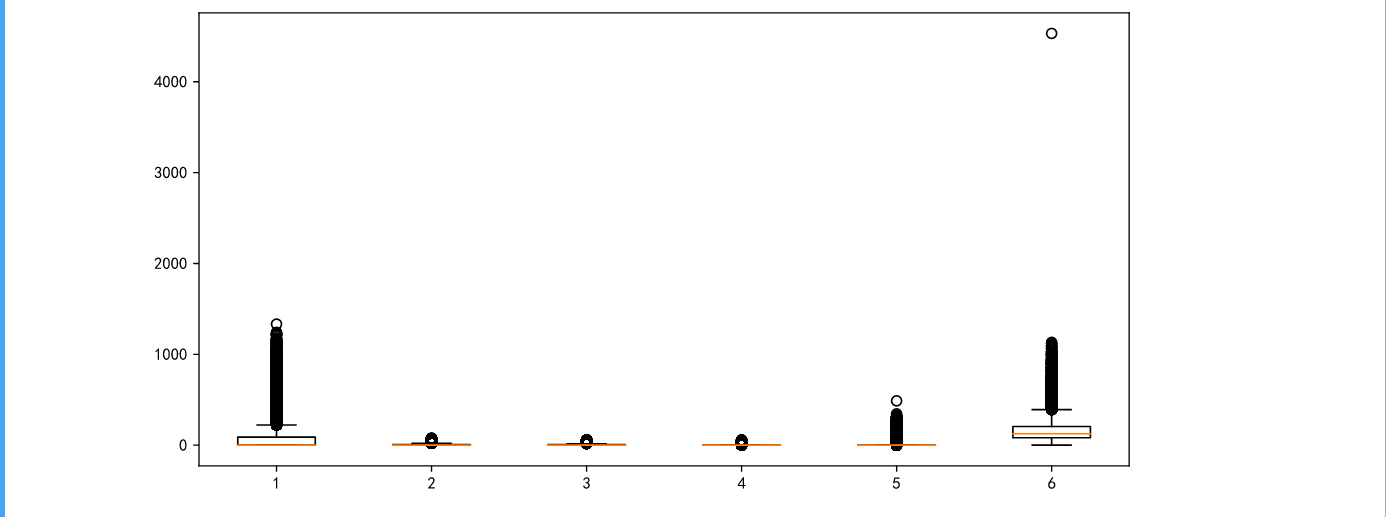


#### 查看数据分布

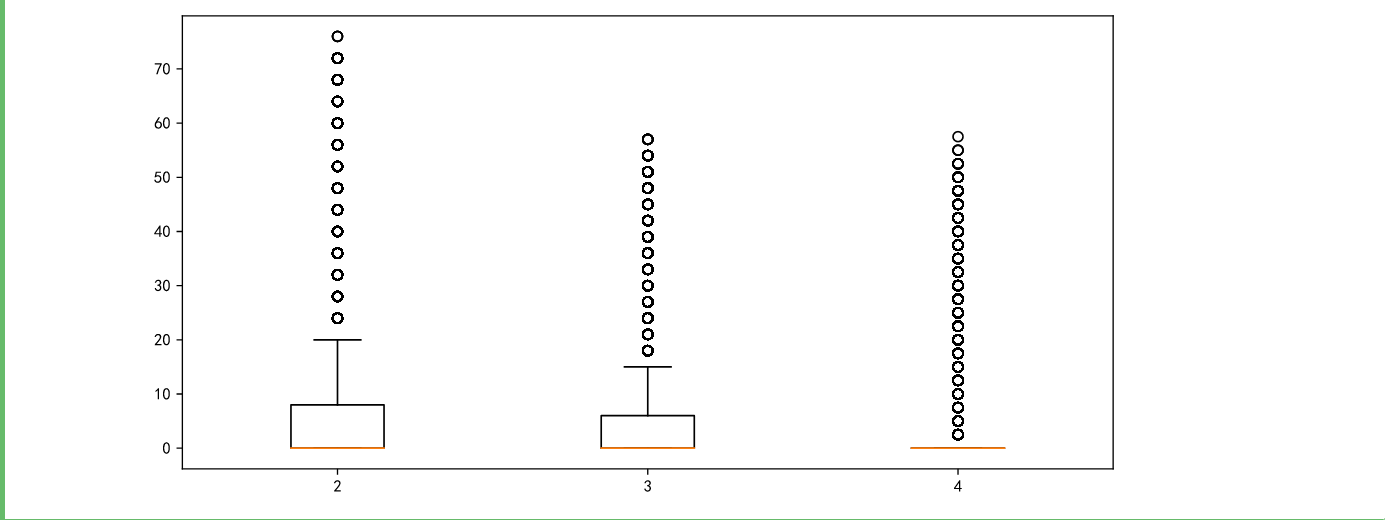
6台消纳的数据分布密度曲线如下所示。统计了2年的半小时消纳数据。可以看出6台消纳的值基本集中在某个范围之内。



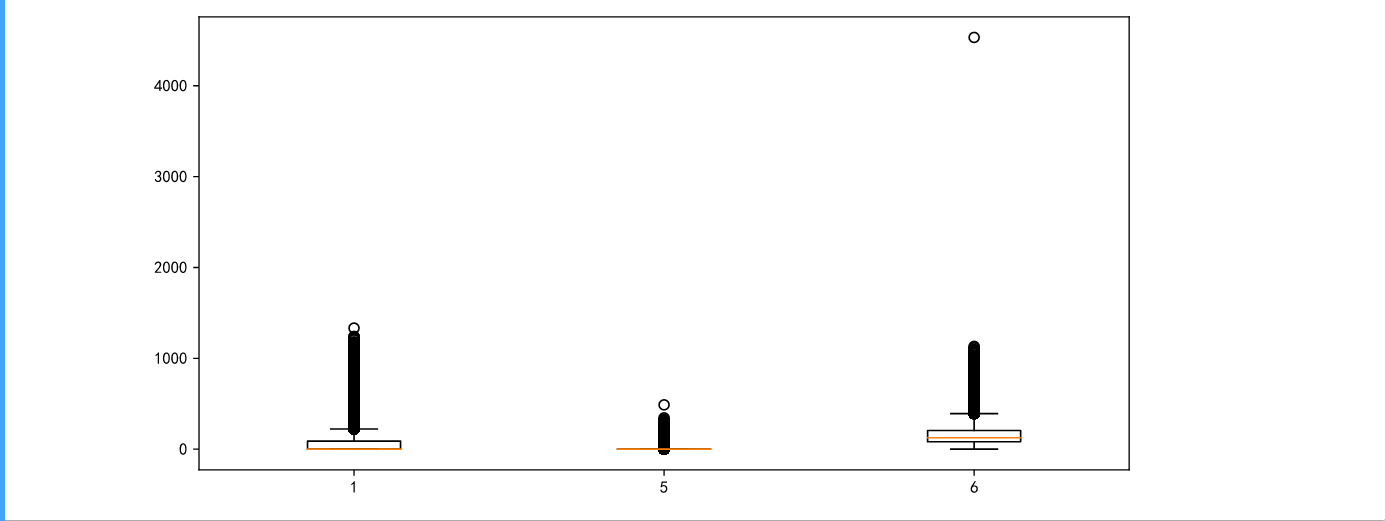
箱型图显示5号和6号大电网消纳有明显的2个离群点。



3台光伏消纳也有一些离群点。



风力消纳和大电网消纳的离群点较多。



结论：数据中存在较少的离群点，考虑到无冲击负荷且负荷的变化应较为平稳，因此可以采用平滑处理等方式校正离群点。

#### 增加特征

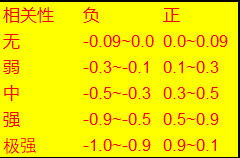
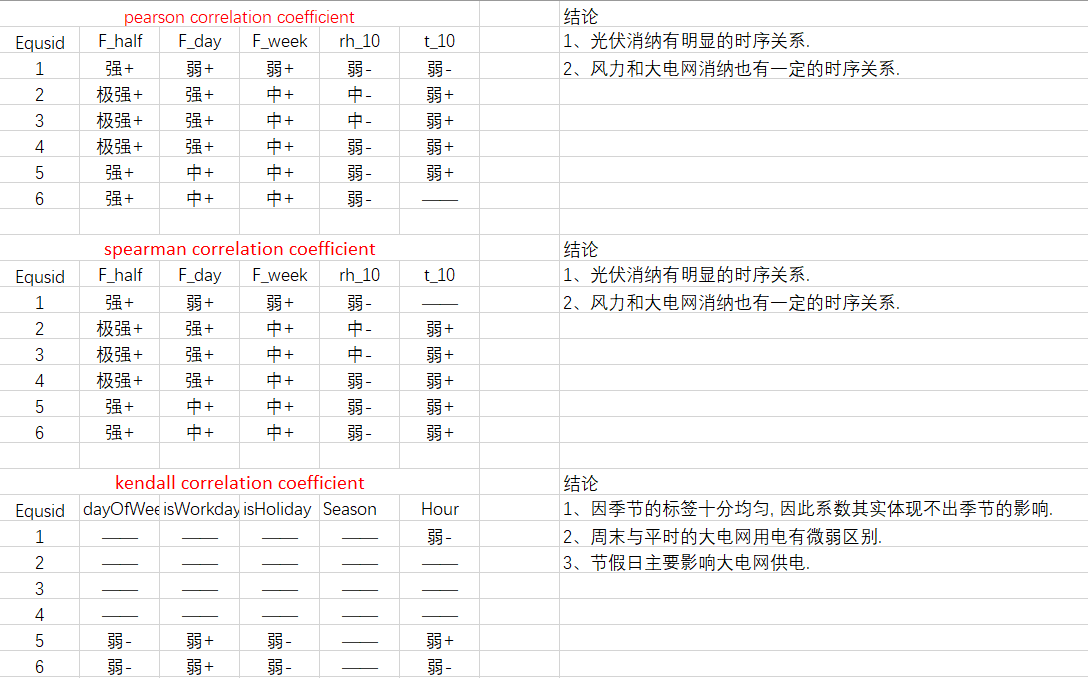
将气候特征（降水量、湿度）、日期特征加入分析，日期特征为星期几，是否为工作日，是否为节假日，季节。注意到气候信息中有相当大一部分半点时刻为空值，用前后时间点的平局值填充。尝试添加降水量数据，但是很遗憾，没有收集到。在DataSet3.0.csv中添加了气候特征，并在DataSet4.0.csv中填补了气候数据的缺失值，加入了日期特征。

查阅相关文献可知，负荷曲线与其之前时间的负荷可能存在某些关系，因此在按半小时预测的数据集中可以加入同一台消纳半小时之前的负荷、同一台消纳同一时间前一天的负荷、同一台消纳同一时间前一周的负荷这3列特征；在按天预测的数据集中可以加入前一天、前两天、前三天、前一周的负荷这4列特征。

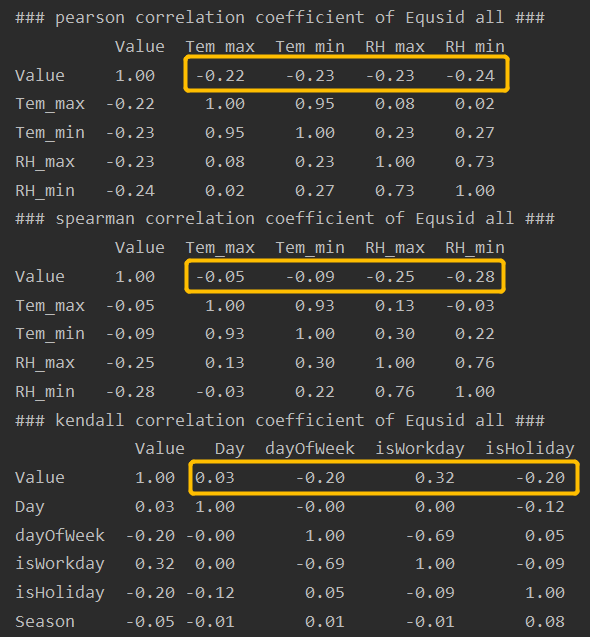
代码文件：DataAnalysis.py

##### 相关性分析

这里运用了pearson相关系数、spearman秩相关、Kendall Tau相关系数。下表中F\_half、F\_day、F\_week分别表示同一台消纳半小时之前的负荷、同一台消纳同一时间前一天的负荷、同一台消纳同一时间前一周的负荷。



下图中Value为日负荷。



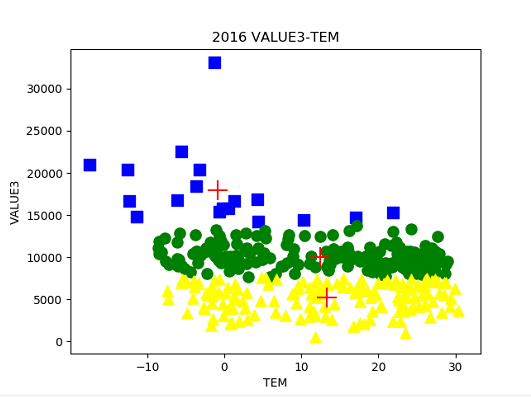
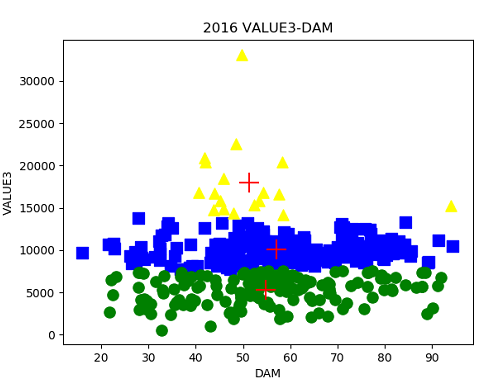
相关性分析的结果显示，光伏消纳有明显的时序关系，用电量与气候弱相关，日用电量在周末与节假日时较低。

我们也对特征之间的相关性进行了分析，结果显示除了3列负荷特征之外，负荷的各个特征之间都是几乎没有相关性的。

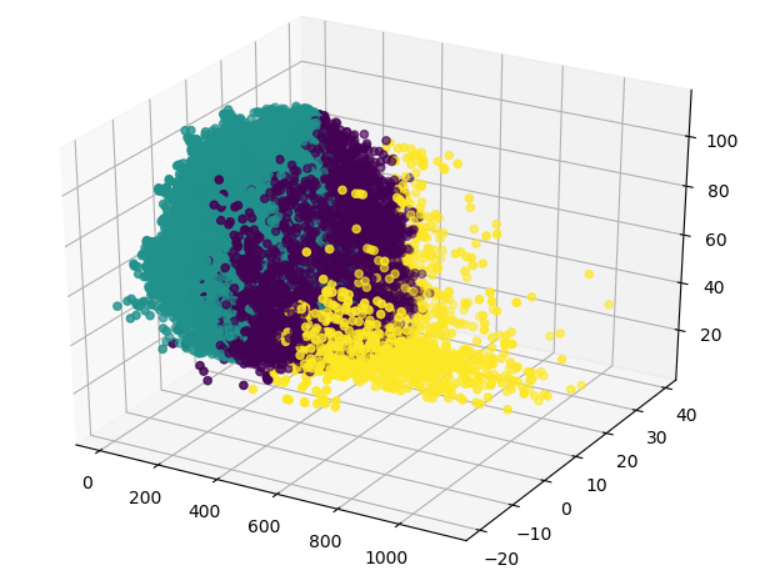
单纯的时间序列用不到特征，所以不予考虑，特征列中可以尝试增加上述3列特征，但是这样在预测的时候只能一个样本一个样本预测，无法并行计算，时间损耗极大。

##### 聚类分析

考虑到负荷与气候特征可能存在某种聚类关系，聚类分析的结果如下所示。TEM表示温度，DAM表示湿度。

聚类结果显示，负荷Value与气候特征之间几乎没有聚类关系。如果考虑3维空间，聚类结果如下所示。结果表明同样负荷Value与气候特征之间几乎没有聚类关系。



#### 填补缺失值

DataSet2.0版本有非常多的缺失值需要填充，观察lose\_data.csv文件缺失数据分布可知，大电网为主要负荷, 辅以风力消纳与光伏消纳. 消纳顺序大致为6 5 1 2 3 4. 这也和公司人员描述的3种消纳类型不区分负荷类型供电，大电网为主要供电方式契合。且当大电网或风电数据缺失时，通常为连续数小数的缺失，疑似为停电检修。因此我们检测半小时数据，发现一条基本规律，当6的Value为0时, 12345必为0. 当5为0时, 1234必为0. 当1为0时, 234必为0。DataSet5.0.csv完成了填补缺失值的工作。

因此填补策略为:

---第一步, 156全部填0, 234每天的0~6点、19~24点全部填0.

---第二步, 234剩下时段采用平滑处理.

测试数据集（2018年）所有缺失值均填0.

代码文件：fillLoseData.py

#### 处理离群点

离群点的2种类型——波峰异常、冲击负荷。DataSet6.0.csv完成了处理离群点的操作。

代码文件：handleOutliers.py

##### 1）波峰异常

波峰异常的判定依据如下, 超过阈值则改为阈值。这里其实总共只修改了不到5行数据的值。

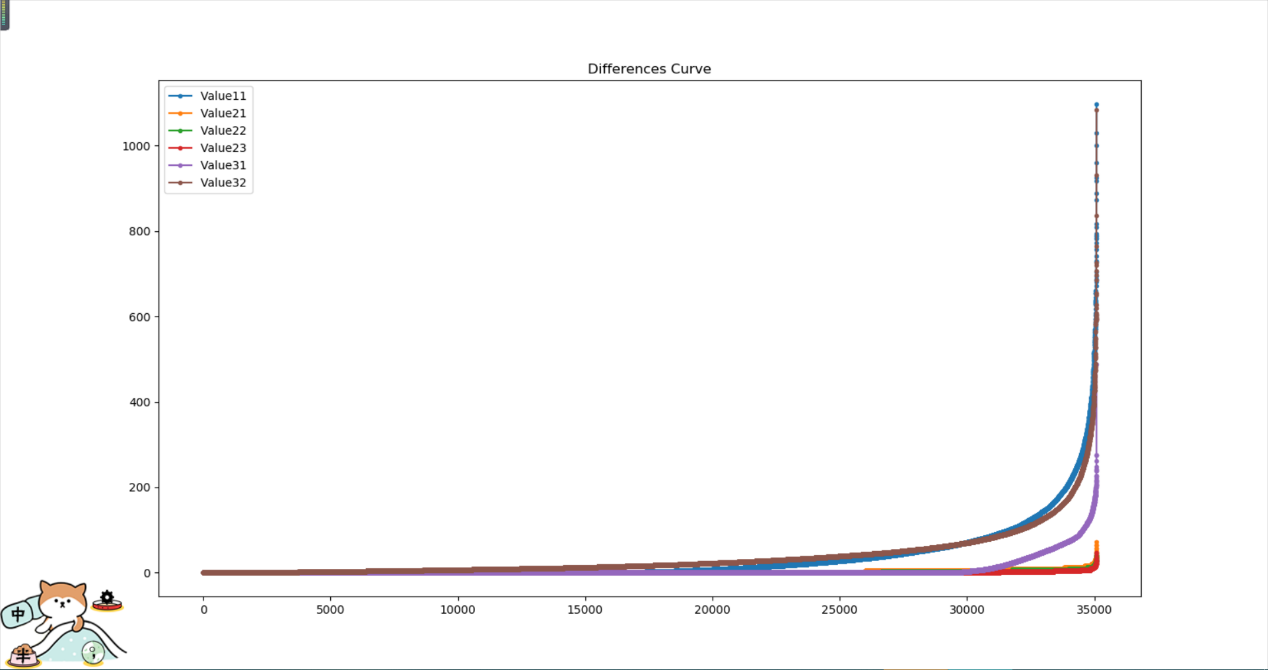
---消纳1 > 1333

---消纳2 > 78 消纳3 > 78 消纳4 > 78

---消纳5 > 444 消纳6 > 1133

##### 2）冲击负荷

相同消纳前后半小时负荷值之差应该处于一个合理的范围之内。6台电机的前后时刻误差曲线如下所示。



我们取了0.05%的差值结点，以此为阈值校正负荷值。

冲击负荷的判定依据如下, 超过阈值则调整为相邻值加/减阈值.

---消纳1前后差值 > 875.5

---消纳2前后差值 > 32 消纳3前后差值 > 32 消纳4 后差值 > 32

---消纳5前后差值 > 269.5 消纳6前后差值 >1059.5

#### 可用的数据集

上述所有操作执行完毕后，重新添加F\_half、F\_day、F\_week三列特征，生成了可用于训练和测试的DataSet7.0.csv文件。

对2018年的测试数据集做同样的处理，不同的是F\_half、F\_day、F\_week三列设为空值，填补缺失值的方式为全部填充0值，且不处理任何离群点。

代码文件：addFeatures.py

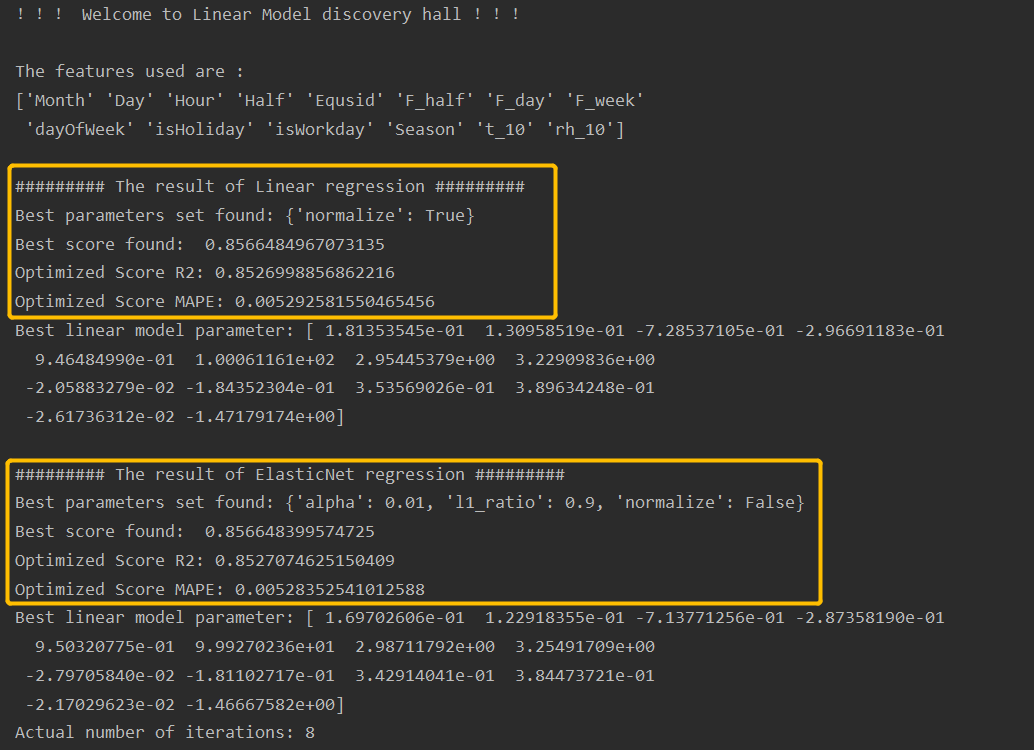
# 模型选择

以最终得到的DataSet7.0.csv为数据集，随机划分2年数据，取30%的数据集为测试数据集，通过暴力寻参测（6折交叉验证）试多种模型的效果。评价指标为R²与测试集上总的MAPE。

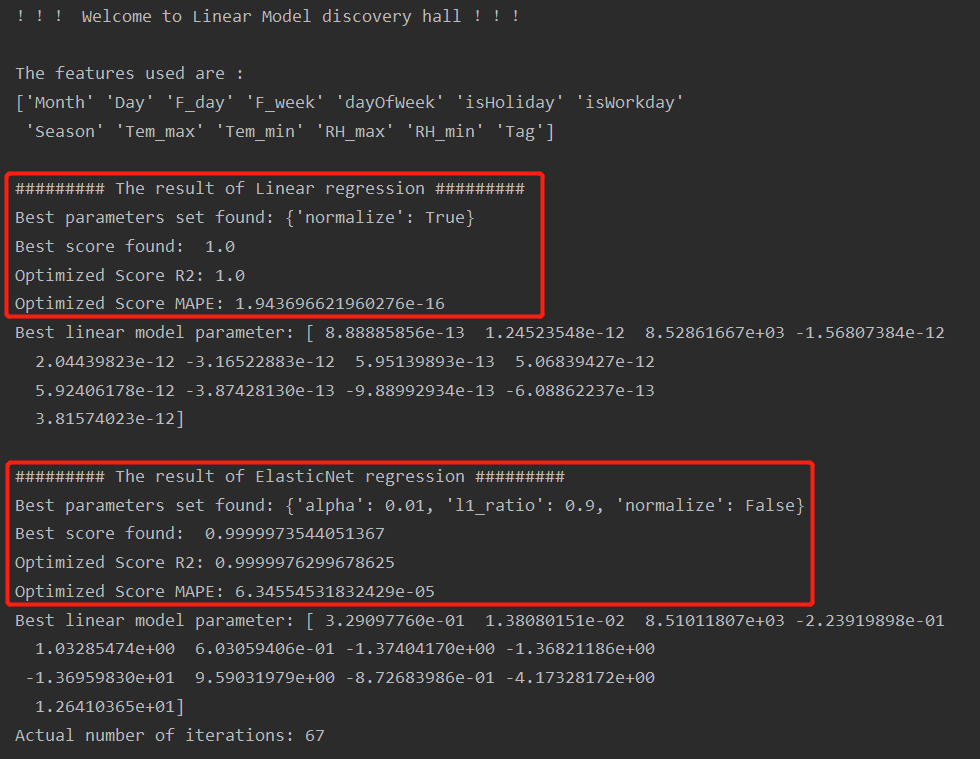
##### 线性模型

线性模型主要包括了线性回归、Lasso回归、岭回归和ElasticNet回归。结果显示在以半小时为刻度区分电机的数据集上，4种回归的效果几乎一样。R²接近0.86，MAPE更是惊人的只有0.5%。而以天为刻度合并负荷的数据集上效果更是几乎能够百分之百的预测准确。模型的最佳参数、最佳系数如下图所示。结果显示Hour、Equsid、F\_half、F\_day、F\_week、rh\_10为主要特征。我们也依照电机序号将数据集划分为6堆，单独做模型测试，结果几乎一致。

下图为以半小时为刻度区分电机的模型结果图。

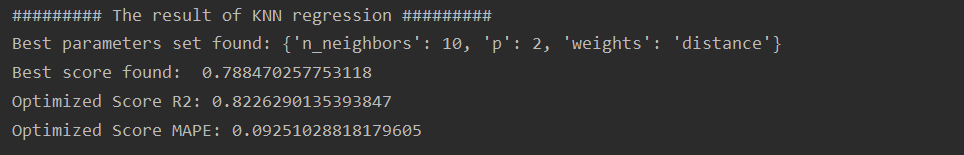


下图为以天为刻度合并负荷的模型结果。



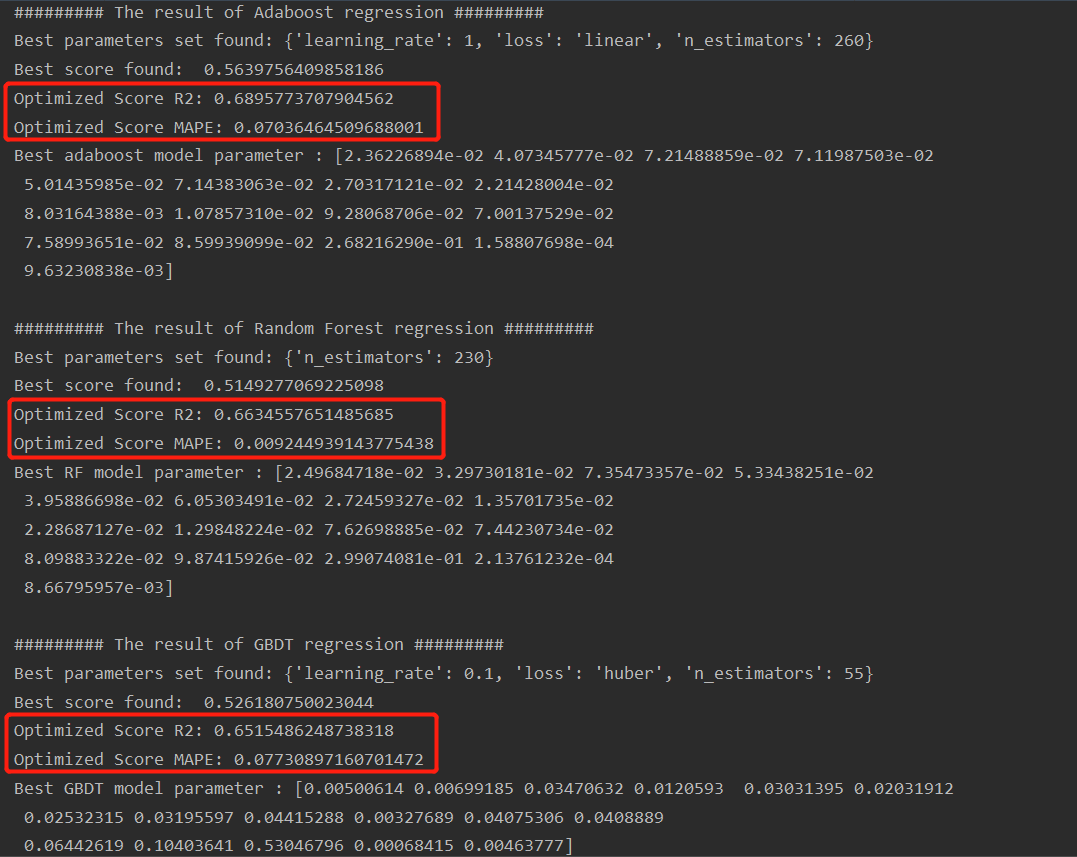
##### KNN

KNN的结果确实不理想，原因是KNN本身无法很好的利用分类特征。



##### 集成学习

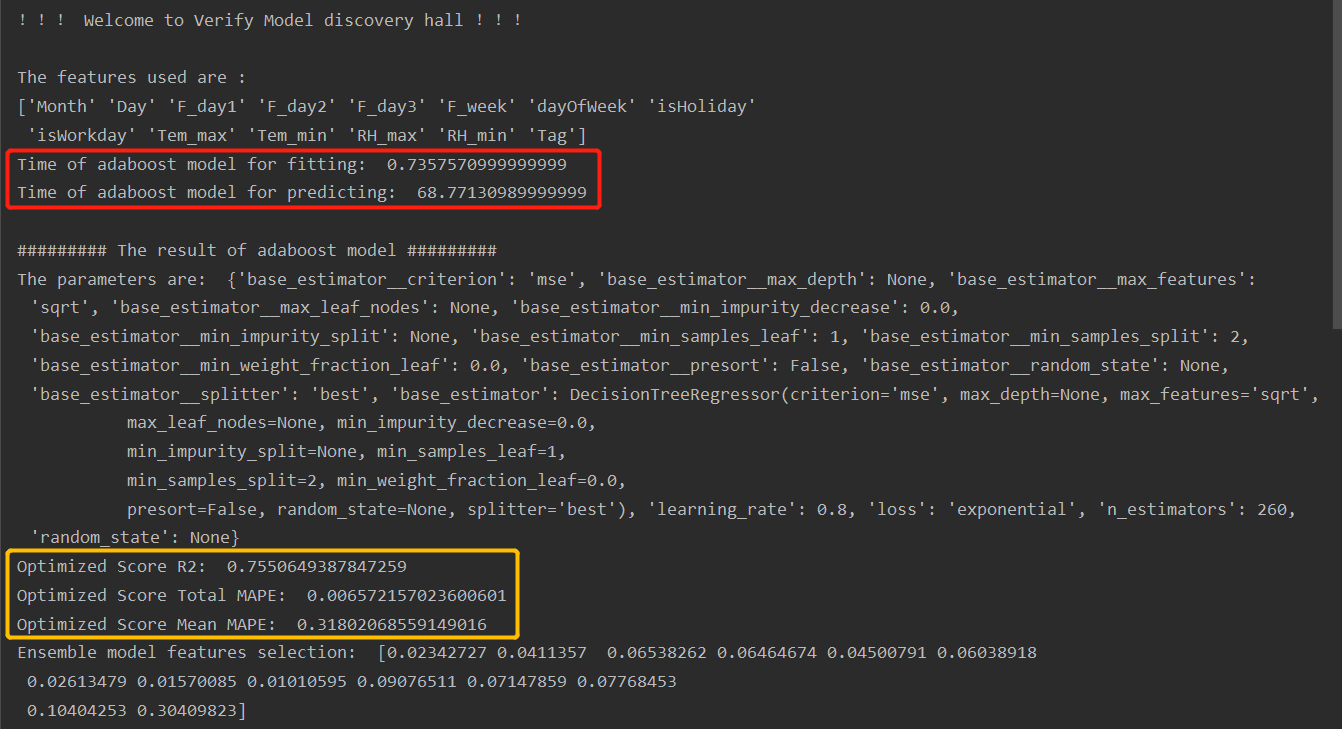
可以看出集成学习的效果在过去2年里面也挺不错的。

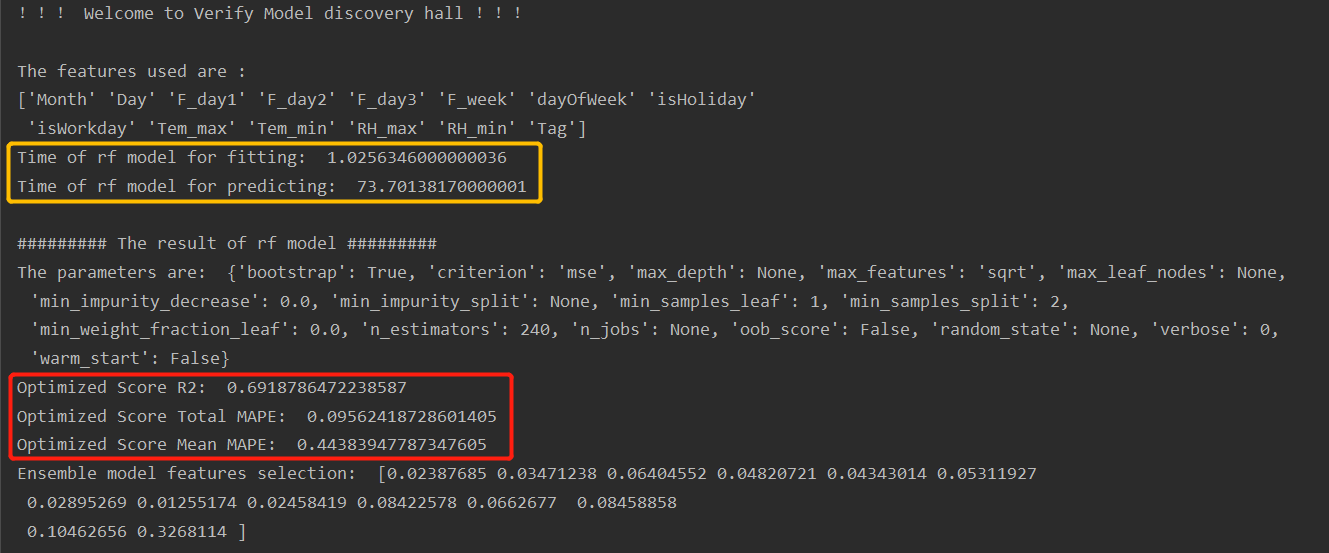


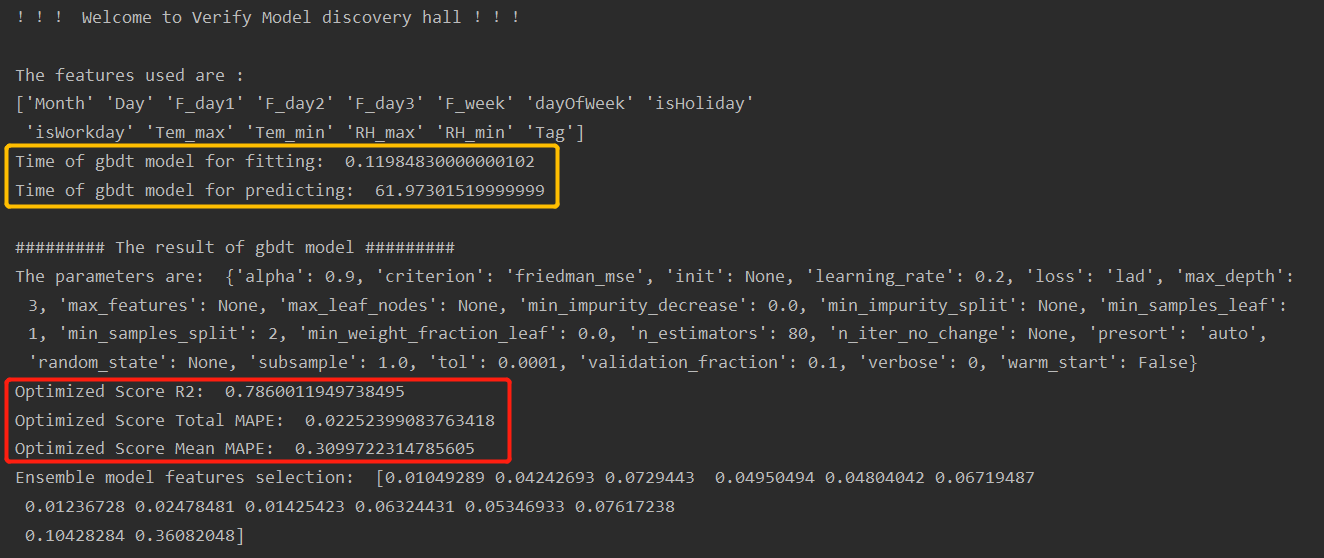
# 验证模型

从理论上分析，线性模型和KNN都很难利用到分类特征的信息，而基于树的模型则恰恰相反，因此使用Adaboost、Random Forest、GBDT的最佳参数重新训练2016~2017年的数据，并在2018年的数据集上做验证。

结果如下所示。







Adaboost模型的R²达到了0.74，总MAPE为0.66%，而日MAPE绝对值的均值为32%，这表明模型对整体的拟合较好，但是存在一些局部的过拟合或欠拟合。

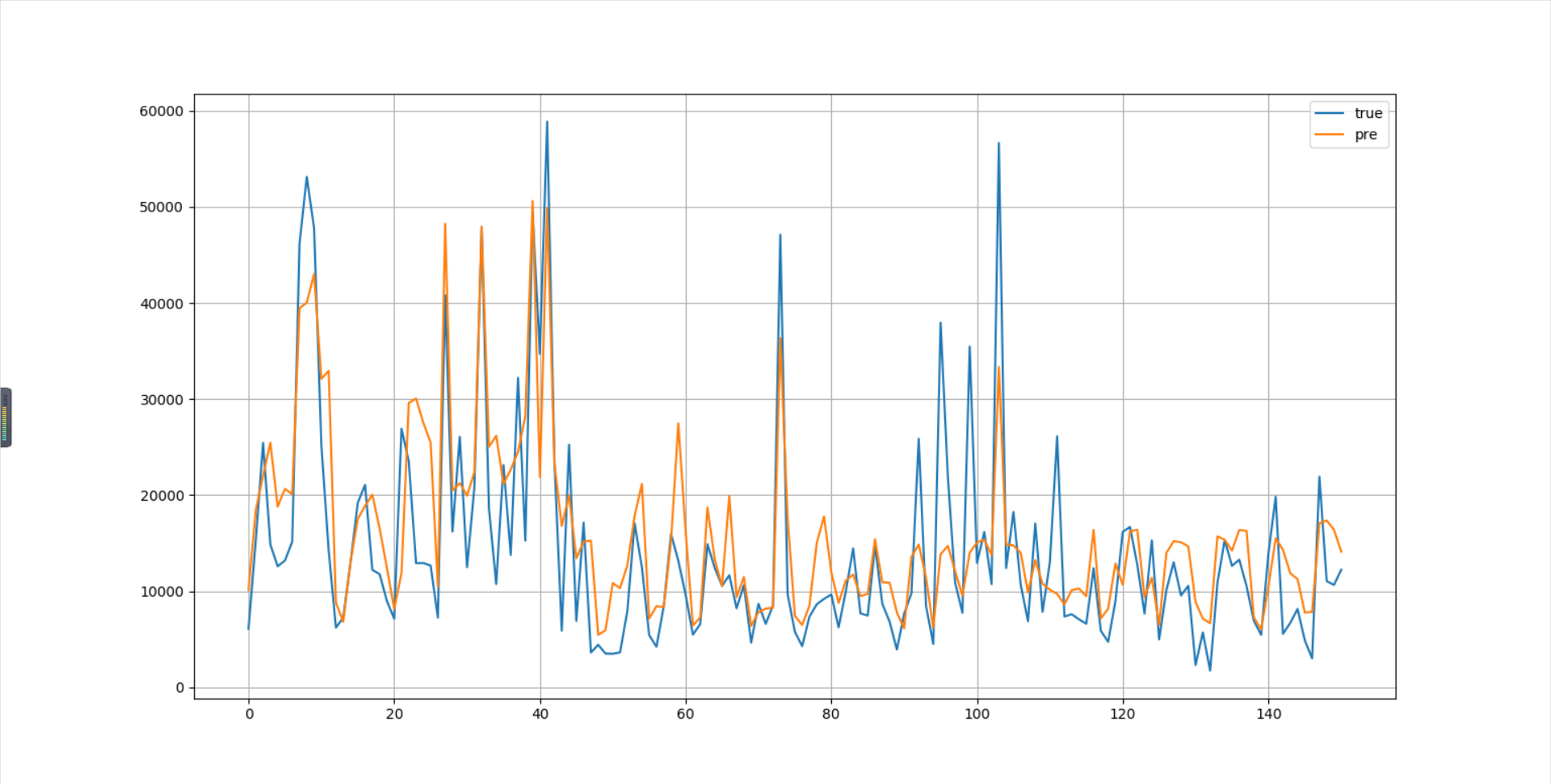
同样的RF与GBDT模型得出的结果与Adaboost类似。

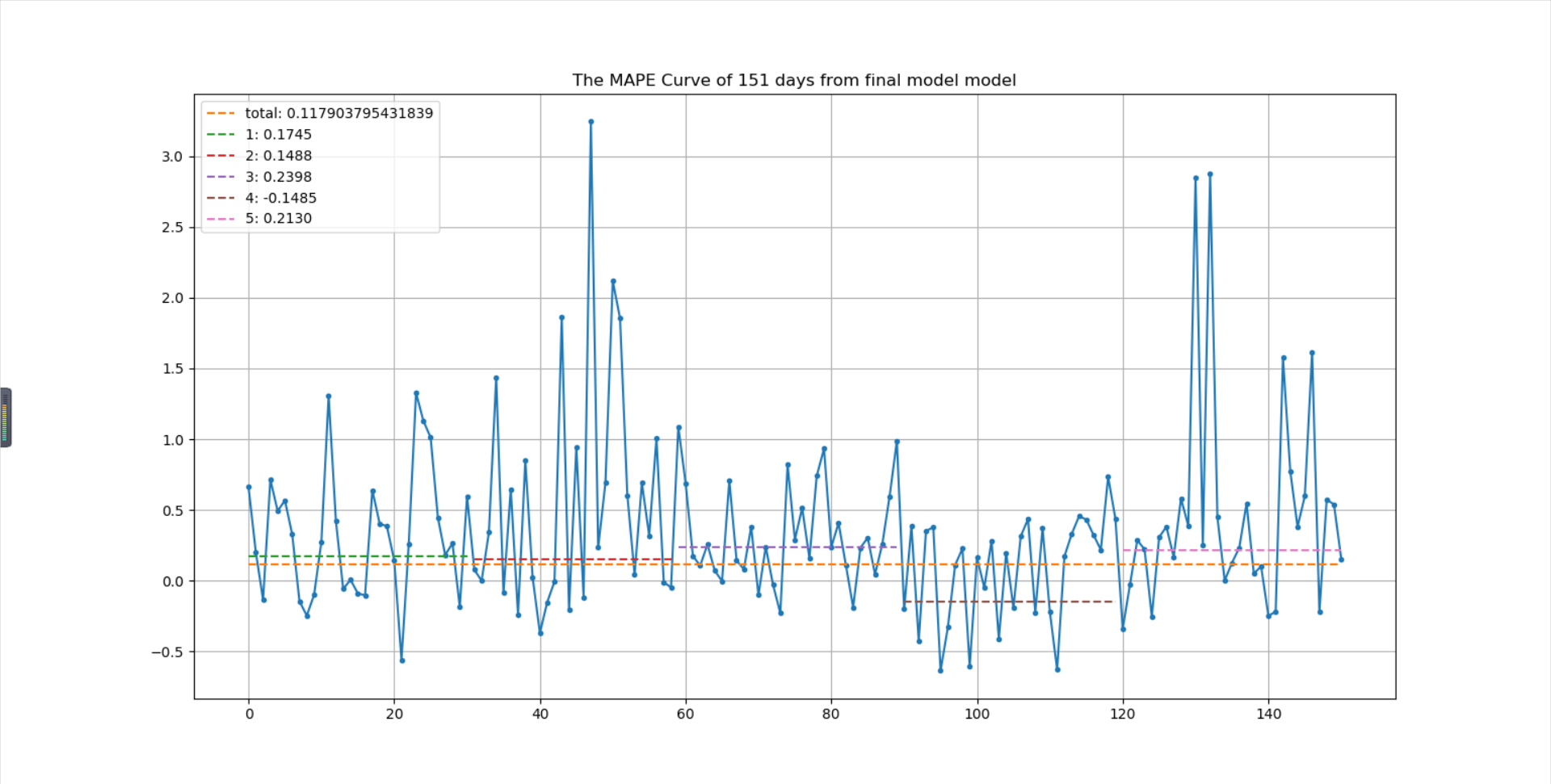
# 模型融合

考虑用线性回归或均值拟合2018年的负荷曲线。

#### 1、均值拟合

均值拟合的效果似乎变得更加不好了。





#### 2、线性回归

使用线性回归拟合负荷曲线的效果较好，5个月总的MAPE几乎为0。3/4月份的MAPE也得到了一定的修正。

