一．赛题与数据介绍

1.1 赛题及背景

半导体产业是一个信息化程度高的产业。高度的信息化给数据分析创造了可能性。基于数据的分析可以帮助半导体产业更好的利用生产信息，提高产品质量。现有的解决方案是，生产机器生产完成后，对产品质量做非全面的抽测，进行产品质量检核。这往往会出现以下状况，一是不能即时的知道质量的好坏，当发现质量不佳的产品时，要修正通常都为时以晚，二是在没有办法全面抽测的状况下，存在很大漏检的风险。

主办方希望着由机器生产参数去预测产品的质量，来达到生产结果即时性以及全面性。更进一步的，可基于预先知道的结果，去做对应的决策及应变，对客户负责，也对制造生产更加敏感。

1.2 问题痛点

1.工业数据的抽检采样数据难以获取，数据样本量非常少

2.生产工序多，每道工序都有可能会对产品的品质产生影响

3.数据记录可能会存在异常（如测点仪表的波动导致、设备工况漂移等现象），模型需要足够稳定性和鲁棒性。

1.3 数据

具体到本次比赛，主办方给出了 500 个样本在 8029 个维度的生产数据，包括反应机台的温度，气体，液体流量，功率，制成时间等数据，预测目标是某匿名的产品质量相关的连续指标。Test：A 榜数据 100 条，B 榜数据 121 条。

	210X1	210X2	210X3	210X4	210X5	210X6	210X7	210X8	210X9	210X10	210X11	210X12	210X13	210X14	210X15	210X16	210X17	210X18	210X19	210X20	210X21	210X22	210X23	210X24	210X25
0	102.05	0.465	0.27	1.43	67.45	4.62	-0.54	-1.05	-0.13	26.3	27.95	0.532	0.077	0.079	0.078	0.079	750	0.4	2	14.3	14	14	14.2	2.02E+13	0.68
1	100.95	0.805	0.22	3.477	62.08	3.412	-2.12	1.02	0.08	28.2	24.27	2.653	0.072	0.065	0.073	0.067	750	0.4	2	13.9	14	14	13.9	2.02E+13	0.72
2	98.56	0.555	0.24	1.172	56.7	3.08	-2.25	0.88	0.17	26.6	24.51	0.523	0.076	0.072	0.077	0.073	750	0.398	2	14.4	14	14	14.7	2.02E+13	0.75
3	100.35	0.901	0.22	3.631	62.25	3.949	-1.98	0.82	0.08	25.2	24.38	2.582	0.074	0.068	0.074	0.067	750	0.4	2	14	14	14	13.8	2.02E+13	0.7
4	100.25	0.854	0.23	3.429	61.42	3.63	-1.89	1.02	0.08	27.3	24.36	2.535	0.073	0.067	0.074	0.067	750	0.4	2	14.2	14	14	14.1	2.02E+13	0.72
5	100.55	0.882	0.22	3.462	61.85	3.747	-2.1	0.93	0.08	25.3	24.34	2.434	0.073	0.067	0.073	0.067	750	0.4	2	14.2	14	14	14.1	2.02E+13	0.72
6	102.2	0.44	0.27	1.429	67.38	4.628	-0.2	-0.44	-0.13	26.5	27.79	0.496	0.077	0.079	0.078	0.079	750	0.4	2	14.4	14	14	14.1	2.02E+13	0.67
7	102.25	0.42	0.27	1.312	67.47	4.614	-0.99	-0.99	-0.13	27.1	27.9	0.485	0.076	0.08	0.078	0.079	750	0.4	2	14.3	14	14	14.2	2.02E+13	0.67
8	102.25	0.421	0.27	1.311	67.44	4.631	0.32	-1.19	-0.14	25.9	27.88	0.487	0.078	0.079	0.078	0.079	750	0.4	2	14.3	14	14	14.4	2.02E+13	0.67
9	100.85	0.958	0.22	3.705	62.88	4.194	-2.14	0.99	0.08	27.7	24.37	2.474	0.073	0.067	0.074	0.067	750	0.4	2	14.2	14	14	14.1	2.02E+13	0.72
10	98.95	0.525	0.25	1.132	57.14	2.992	-2.04	0.3	0.15	27.3	24.88	0.487	0.077	0.072	0.077	0.072	750	0.398	2	14.6	14	14	14.5	2.02E+13	0.75
11	101.5	0.312	0.4	0.554	71.63	3.697	0.2	-0.44	-0.16	24.3	41.75	0.349	0.079	0.085	0.079	0.085	750	0.4	2	13.9	14	14	14.1	2.02E+13	0.72
12	101.45	0.311	0.4	0.56	71.61	3.712	0	-0.61	-0.18	24.9	41.74	0.336	0.079	0.085	0.079	0.085	750	0.4	2	13.9	14	14	14.4	2.02E+13	0.72
13	101.45	0.314	0.4	0.571	71.67	3.715	0.03	-0.58	-0.16	25	41.73	0.335	0.079	0.085	0.079	0.085	750	0.4	2	13.9	14	14	14.2	2.02E+13	0.72
14	101.75	0.808	0.22	3.487	62.54	3.524	-1.96	1.08	0.08	26.4	24.24	2.475	0.071	0.066	0.072	0.066	750	0.4	2	14.3	14	14	14	2.02E+13	0.72
15	99.1	0.569	0.24	1.134	56.66	3.46	-1.96	0.7	0.14	26	24.51	0.497	0.077	0.072	0.077	0.072	750	0.398	2	14.5	14	14	14.7	2.02E+13	0.75
16	102.1	0.439	0.27	1.391	67.24	4.618	-0.08	-0.96	-0.13	26.3	27.93	0.528	0.078	0.079	0.078	0.079	750	0.4	2	14.5	14	14	14.1	2.02E+13	0.67
17	101.95	0.843	0.22	3.704	63.19	3.745	-2.06	0.93	0.08	27.7	24.24	2.489	0.071	0.065	0.071	0.066	750	0.4	2	14.3	14	14	14.1	2.02E+13	0.72
18	99.13	0.575	0.24	1.224	56.64	3.431	-1.98	0.62	0.14	27.3	24.51	0.497	0.077	0.072	0.077	0.072	750	0.398	2	14.6	14	14	14.4	2.02E+13	0.75
19	102.15	0.448	0.27	1.411	67.26	4.627	-0.68	-0.61	-0.1	26.2	27.91	0.535	0.077	0.08	0.078	0.079	750	0.4	2	14.3	14	14	14.1	2.02E+13	0.67
20	101.5	0.794	0.22	3.396	62.02	3.372	-1.96	1.19	0.08	28	24.24	2.488	0.072	0.066	0.072	0.066	750	0.4	2	14.3	14	14	14.1	2.02E+13	0.72
21	99.11	0.571	0.24	1.208	56.61	3.431	-1.96	0.65	0.14	28.4	24.51	0.499	0.077	0.071	0.077	0.072	750	0.398	2	14.5	14	14	14.7	2.02E+13	0.75
22	102.15	0.461	0.27	1.516	67.38	4.603	-0.68	-1.05	-0.12	26.2	27.92	0.529	0.077	0.08	0.078	0.079	750	0.4	2	14.1	14	14	14.4	2.02E+13	0.67

此次比赛给出的工业生产数据来自于 16 组不同工序，有 12 种不同的机器，不同机器有 2-8 种不同机器类型

一大难点是，全部的生产数据都是匿名的，计量单位与精确度全部未知，列标仅仅给出了数据所处的工序，而数据的数值差异也十分巨大，有杂乱的缺失，较为脏乱

二．我的解决方案概述

2.1 高维小样本问题整体解决方案

经过对论文，相关技术文章的查阅，我发觉高维小样本问题的处理是现在机器学习问题的一个研究方向，简单总结而言，我认为本问题有以下几种思路可以优化问题的求解：

1. 在样本方面：如果有可能的话，可以增量采样
2. 特征方面：谨慎严格的特征筛选与压缩十分重要
3. 算法选取方面：使用尽量简单的模型，避免使用神经网络这种表达能力过于强大的模型，加上工业数据本身由于示波器波动等原因，数据噪音本就偏大，再加上样本数量很小，神经网络预计会容易过拟合
4. 交叉验证方面：样本稀少，需要更加严格的模型评估手段，传统的三折五折过少，小样本的最佳交叉验证方式是留一法 LOOCV，这样可以减少样本集合划分的随机性
5. 模型评价方面：由于超参数是需要按照交叉验证来优化的，极小的样本量容易造成超参的选择对模型过拟合，使用嵌套交叉验证对模型产生无偏估计
6. 多模型融合：对于单模型来说，它们都有可能捕获到不同方面的噪音，而一个健壮的 ensemble 模型会有更好的鲁棒性

另外，在图像识别等方面，小样本可以使用迁移学习来解决

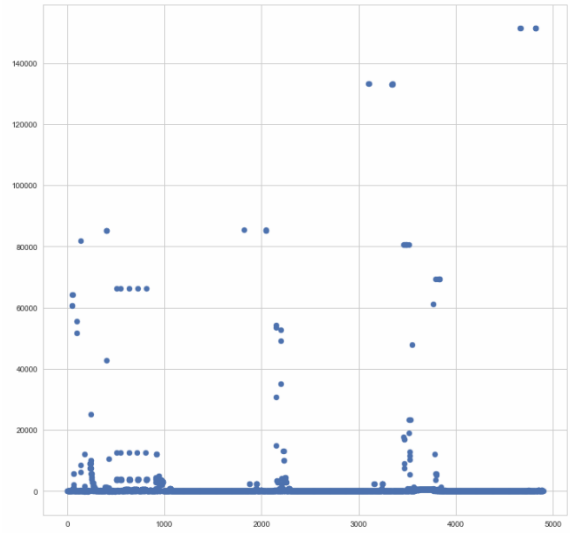
2.2 我的解决方案

三．数据探索

3.1 特征的取值范围

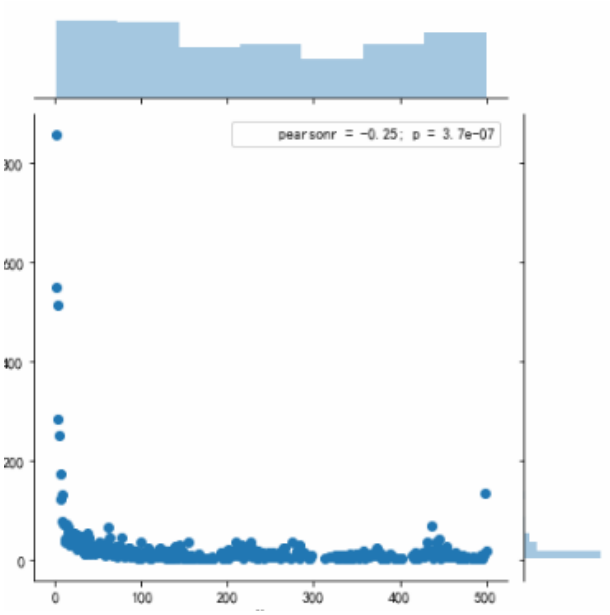
```
train_numerical_feature_median7=train_numerical_feature_median6[train_numerical_feature_median6<1.0E+7]
plt.scatter(np.arange(len(train_numerical_feature_median7.values)),train_numerical_feature_median7.values)
```

```
<matplotlib.collections.PathCollection at 0x1e51f320>
```



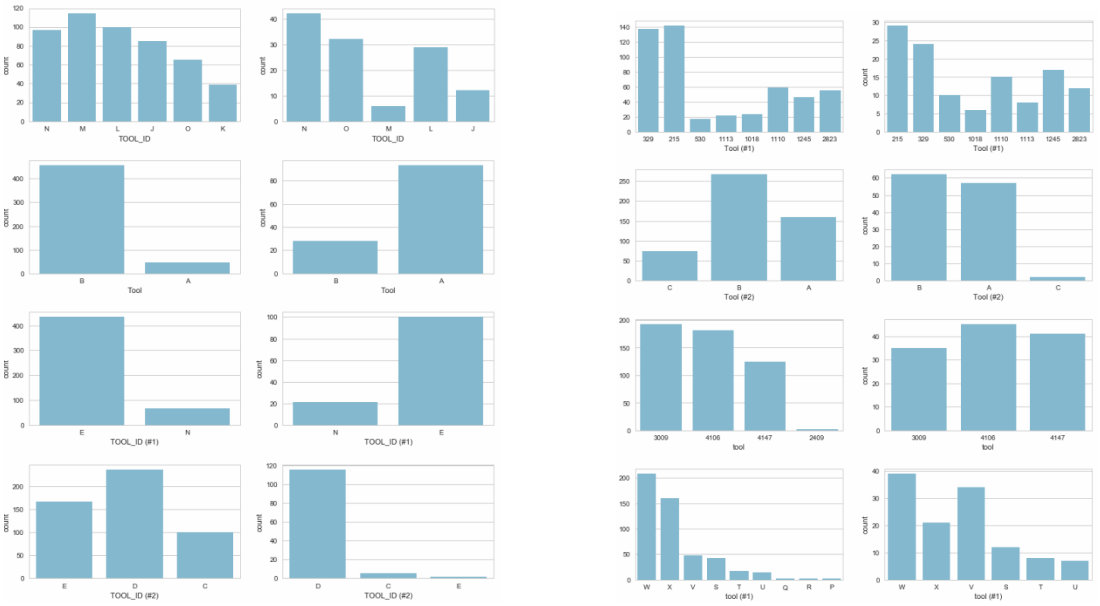
数值在-100 到 100 之间分布的特征较多，有少部分大数值特征

3.2 特征数值的取值种类

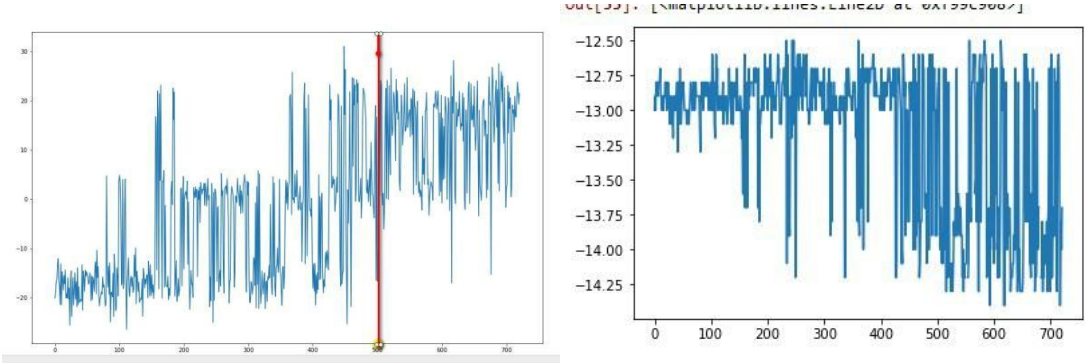


大部分特征在 500 个样本上有 50 种一下的取值类型

3.3 样本的不同机器上的分布



两张图的左侧是训练集样本在 8 种机器上的分布，右边是预测集样本在相同机器上分布的对应，可以看出训练集和预测集的样本构成并不是相似的，甚至是有很大差异的，这一点在某些数值特征上也可以看出：

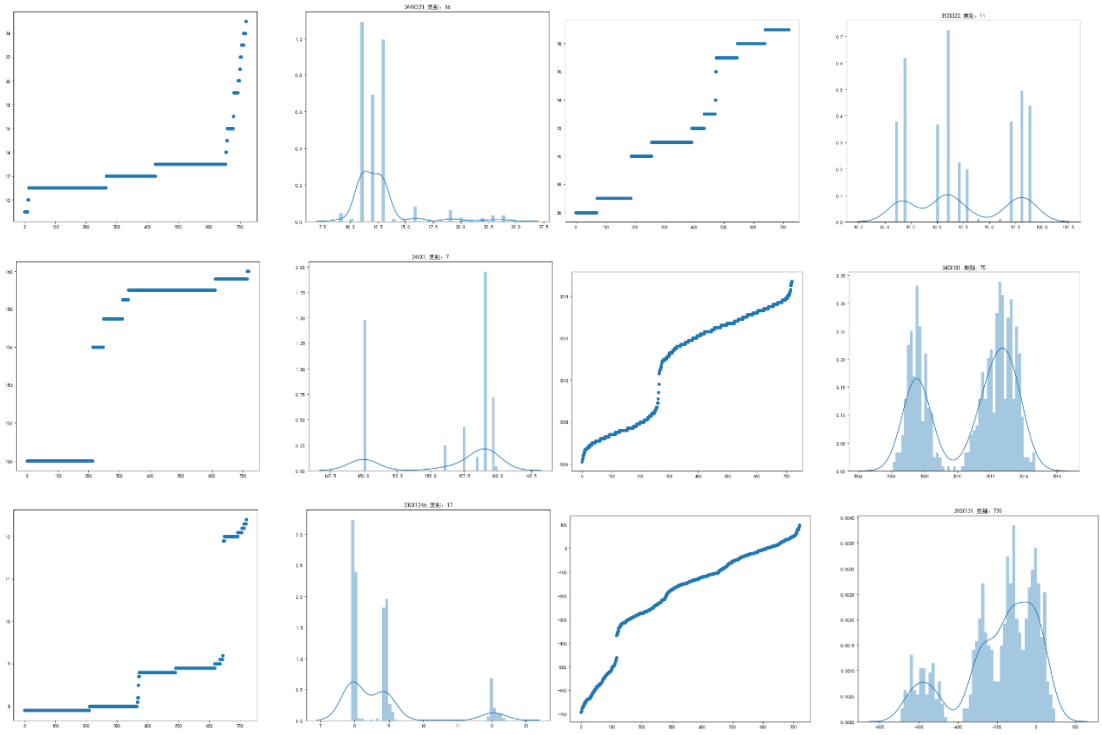


这是两组数值数据在标签 1-600（横坐标）上的分布，500-600 是预测集，可以看出分布差异的巨大。

也正是基于这个原因，建模时每个人都遇到了极其夸张的过拟合现象，本地 0.03 以下，线上 0.045 左右，这使得比赛过程中我一直难以找到一些可依赖的依据去深入下去

3.4 许多数据与机器的同分布问题

在比赛将近后期，我发现了这个数据特征，起初，我发觉有很多的特征的形态不仅不是遵循高斯分布，还是断层的



最初，我认为这种分段的现象可能意味着对特征进行聚类会提取出一些有效信息，后来经过对聚类结果的观察发现其中大部分数据的分层和机器的类别有较强的共线性，我意识到这里的机器有可能是测量机器或者统计单位不同的生产机器。

所以“数值如何去除机器类别的共线性”成为了一个重要命题，但是邻近后期，我一时没有想到合适的方法，遂放弃。

但是在赛后跟选手们交流后我学到了一种好的解决办法：

将每个特征在不同机器类型下的数值进行标准化，然后拼合在一起。

3.5 特征之间的规律

也是比赛后期，在某一次手动观察源数据集的时候我发现向右快速浏览的时候有一种直觉性的规律性存在，经过对某一区段的直接观察，我发现数据表的局部会有一些规律性的东西存在，比如经过 10 个分布波动在-1-1 之间的特征（他们之间相似度很高）之后，会规律性地出现两列分布波动在 1000-2000 的特征，这个周期会出现 5.6 段后消失。

这忽然让我想起之前看过的一些工业生产数据建模的论文，其中有一篇重点提到，对于工业数据而言，往往直接的数据指标根本不意味着什么，而关乎生产质量的指标是同一部件之间的温度差，压力差这样的“二阶特征”，也正是因为这个发现，我停止了对这个特征匿名比赛的继续探索，随着我的深入，它越发像是一个深不可测的解谜游戏，特征的构建找不到真实支撑。

三．数据清洗

3.1 空缺值

首先，空缺信息或许隐藏着一些信息，因为我观察到某些样本有成片的缺失现象，所以我构造了每个样本在不同工序内和整体上的空缺值统计数值，作为备选的特征

然后使用更免疫离群点的中位数作为空缺填补手段

3.2 0 值处理

在部分列，0 值基本可以确定代表着空缺，但在某些列，0 值却是正常的数据，但是代表空缺的 0 值的发掘过程我遇到了困难，起初我认为应当从特征的取值种类去考虑这一问题，如果是种类大于 10 种的特征，基本可以被认为是数值型特征，而在某些情况中真正有意义的 0 会以 0.00 的形式表示，那么 0 就一定代表着空缺，但后来发现，并不是所有的特征的 0 都是 float，于是最终，我将 0 值处理归并到了异常值处理

3.3 异常值处理

取了均值外 2 个标准差作为边界，若 0 值被作为异常值，则处理，否则不作处理

四．特征选择—集成特征筛选器

4.1 特征选择总体方法

整体而言，特征选择共有三种方式：

1. Filter 方法

这是一种单变量的特征选择方法，按照发散性或者相关性对各个特征进行评分，设

定阈值或者待选择阈值的个数，选择特征。

（1）方差选择

（2）相关性选择

1.卡方检验

卡方检验就是统计样本的实际观测值与理论推断值之间的偏离程度，实际观测值与理论推断值之间的偏离程度就决定卡方值的大小，卡方值越大，越不符合

2.互信息

原来我对 X 有些不确定(不确定性为 H(X))，告诉我 Y 后我对 X 不确定性变为 H(X|Y), 这个不确定性的减少量就是 X,Y 之间的互信息 I(X;Y)=H(X)-H(X|Y)

3.相关性系数

例如皮尔森相关性系数

（3）模型打分

根据模型的 feature_importance 选择

2. Wrapper 方法

根据目标函数（通常是预测效果评分），每次选择若干特征，或者排除若干特征，最有代表性的是递归特征消除

递归消除特征法使用一个基模型来进行多轮训练，每轮训练后，消除若干权值系数的特征，再基于新的特征集进行下一轮训练。

3. Embedded 方法

这种方法是将特征的选择完全与算法的运行过程相结合，最有代表性的技术就是正则化，正则化在损失函数中增加一个用于描述网络复杂度的项，常用的正则化有 L1 正则化与 L2 正则化，其中 L1 正则化迫使那些弱的特征所对应的系数变成 0。因此 L1 正则化往往会使学到的模型很稀疏，因而达到特征选择的效果

4.2 集成特征筛选

面对如此高的维度，根据 baseline 返回的特征评分，特征之间的差异性非常小，因此想要获得更加准确的“高信息量”特征选择，就需要更加具有鲁棒性的特征选择手段，所以在经过不断试验和选择后，我制定了这样的特征选择过程：

1. 去掉所有的时间标签特征

时间标签特征本应该有相当大的信息量，但是经过探索，将近 100 列时间标签的时间范围非常大，而且找不到任何规律，基于主办方并没有给出数据的任何

解释，这部分信息处理难度过大，因此舍去

2. 去掉所有方差大于 0.98 的特征

将维度降低到 6800 维左右

3. 设计集成特征打分器

想构造“集成特征打分器”的初衷是，由于并不确定最佳的特征数量是在什么范围内的，所以希望以某种标准为所有特征的信息价值进行排序，以备后续灵活地选择任意数量的特征。

借鉴“集成学习”的思路，尽量选择多种差异性较大的一些模型，获取他们对特征重要度评价的集成作为最终的特征排序

一共分为 3 部分

1.初步：对特征排序取平均

这些模型分别是：

Lasso 回归模型，Ridge 回归模型，随机森林，XGB,SVR

五个模型调参后分别使用 4 个不同的随机种子得到一组排序，然后相同模型内部取排序平均，然后将 5 个模型的排序进行平均

同时还将 pearson 相关系数，相互信息两个指标对特征的排序也分别计算并进行平均

最终将模型对特征的排序与相关性指标对特征的排序以 0.8:0.2 的权重加权平均，获得最终的特征排序

2.使用上述的 5 种模型分别取其重要度前 N 的特征，获得 5 个特征子集，取 5 个集合的交集，使留下的特征大约为 400 个

3.再用步骤 1 得到的排序为待选的特征集合排序，形成新的待用特征集合

4.3 降维

除了特征选择，还尝试了 PCA,LDA 这两个降维算法，查阅相关资料发现这两种算法对于非信号数据的表现并不是很好，尝试过后发现果然，信息损失太严重，故放弃

五．模型设计

2.1 单模型

将待用特征集合的特征数量作为一个超参数，分别尝试了 xgb,lgb,RF,，lassoCV,ridgeCV,SVR 这六个模型，在 50-200 之间进行特征数量的寻优，得到单模型结果，其中 xgb 单模型效果最佳

5.2 模型集成 stacking

首先对 lassoCV 和 ridgeCV

在第二层使用 LR 模型进行 stacking

5.3

六. 引申学习与感想总结

在博客中大致梳理过一些赛后感想：https://zhuanlan.zhihu.com/p/33811393

印象深刻的如下：

1.降噪自编码器

传统的特征选择方法，易于过拟合噪声，对噪声敏感，而采用降噪自编码器对输入特征进行降维/升维能够自动提取有效，低噪的特征，并用于后续模型的训练，显著提升了模型的预测精度.

2.分工序建模+模型集成

有一个队伍将每个样本的不同工序分开建模了，从一定角度来看，这也是一种增量采样的方式，或者说也是 bagging 思想的一种延伸