|  |
| --- |
| **机器学习** |
| **卡特彼勒管定价案例名** |
| 主 研 人：刘一阳  参 研 人： |
| 审 核 人：  方 向：工业算法案例研究  版 本 号：A |
|  |

**版本更新**

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **更新描述** | **作者** |
| 2018/10/31 | A | 初稿 | 刘一阳 |

**选题表格**

|  |  |  |  |
| --- | --- | --- | --- |
| 时间 | 竞赛名 | 竞赛背景描述（50字以内） | 类型（分类/回归） |
| 2018/10/31 | Caterpillar Tube Pricing | 卡特彼勒公司向全球各地的公司销售采矿和建筑设备，其生产的管子（主要产品）每个都具有不同特征，竞赛的目的是通过提供管子的管材，组建，体积等数据集，来预测报价 | 回归 |

目录

[1. 背景描述 4](#_Toc25575_WPSOffice_Level1)

[1.1 竞赛赛题描述 4](#_Toc5940_WPSOffice_Level2)

[1.2 评估指标描述 5](#_Toc12357_WPSOffice_Level2)

[2. 数据来源及描述性统计分析 5](#_Toc5940_WPSOffice_Level1)

[2.1 大赛数据来源 5](#_Toc22683_WPSOffice_Level2)

[2.2 数据的描述性统计 5](#_Toc16174_WPSOffice_Level2)

[2.2.1 数据基本情况描述： 5](#_Toc5940_WPSOffice_Level3)

[2.2.2 数据字段介绍： 5](#_Toc12357_WPSOffice_Level3)

[2.2.3 数据描述性统计 6](#_Toc22683_WPSOffice_Level3)

[3. 优秀算法思路 6](#_Toc12357_WPSOffice_Level1)

[3.1 方案一 6](#_Toc32293_WPSOffice_Level2)

[3.1.1 方案一数据预处理及特征工程部分方案 6](#_Toc16174_WPSOffice_Level3)

[3.1.2 方案一模型设计、建立部分方案 6](#_Toc32293_WPSOffice_Level3)

[3.1.3 方案一结果、排名等 6](#_Toc29922_WPSOffice_Level3)

[3.1.4 方案一算法流程图 6](#_Toc4787_WPSOffice_Level3)

[3.2 方案二 7](#_Toc29922_WPSOffice_Level2)

[… 8](#_Toc4787_WPSOffice_Level2)

[4. 算法比较 8](#_Toc22683_WPSOffice_Level1)

[表4-1 算法比较 8](#_Toc7253_WPSOffice_Level2)

[5. 总结与展望 8](#_Toc16174_WPSOffice_Level1)

[5.1 总结 8](#_Toc23198_WPSOffice_Level2)

[5.2 建模思路 8](#_Toc16619_WPSOffice_Level2)

1. 背景描述

卡特彼勒公司（Caterpillar，CAT），成立于1925年，卡特彼勒公司总部位于美国伊利诺州。是世界上最大的工程机械和矿山设备生产厂家、燃气发动机和工业用燃气轮机生产厂家之一，也是世界上最大的柴油机厂家之一。

90多年来，卡特彼勒公司一直致力于全球的基础设施建设，并与全球代理商紧密合作，在各大洲积极推进持续变革。卡特彼勒2005 年销售收入达到 363.4 亿美元，是建筑机械、矿用设备、柴油和天然气发动机以及工业用燃气轮机领域的技术领导者和全球领先制造商。

目前，卡特彼勒依靠各种供应商来制造这些管组件，每个管组件都有自己独特的定价模型。该竞赛于2016年由卡特彼勒公司发布于Kaggle网站，奖金额度为＄30000。

**1.1 竞赛赛题描述**

卡特彼勒向全球各地的公司销售各种各样的超大型建筑和采矿设备。每台机器都依赖于一套复杂的管子来保持叉车升降，装载机装载和推土机工作。 由于需求的多样性，很难在卡特彼勒的各种机械目录中找到两个完全相同的管子。管子可以在多个维度上变化，包括基础材料，弯曲数量，弯曲半径，螺栓图案和端部类型。

此竞赛提供详细的管材，组件和年度体积数据集，要求挑战团队预测供应商为给定管组件报价的价格。

**1.2 评估指标描述**

竞赛中评估模型优劣的指标。

选手预计的报价与实际报价的结果进行对比，以均方根对数误差（RMSLE）为评价指标，结果越小越好，均方根对数误差计算公式如下：

其中:

* n: 测试集中的报价数量
* ： 选手的预测报价
* ： 实际报价（供应商提供）
* : 自然对数

1. 数据来源及描述性统计分析

**2.1 大赛数据来源**

大赛数据提供方卡特彼勒公司（Caterpillar），为大赛提供真实生产数据，提供工艺专家的专业指导，从软硬件环境诸多方面提供大赛支撑。

数据获取链接：<https://pan.baidu.com/s/1czwM__dPmBd2FoBjDgNzJA>

**2.2 数据的描述性统计**

**2.2.1 数据基本情况描述：**

数据集由大量关系表组成，这些关系表描述了管组件的物理特性。需要将每个管组件的特性与供应商定价动态相结合，以便预测每个管的报价。报价在数据中标记为成本。

**2.2.2 数据字段介绍：**

train\_set.csv和test\_set.csv：

此文件包含有关供应商报价的信息。价格可以通过两种方式报价：括号（bracket pricing）和非括号定价（non-backet pricing）。括号定价是根据最小购买数量给予的价格及假定购买该最小数量的倍数的量。非支架定价具有适用价格的最小订单数量（min\_order）。每个报价都附有年度使用量，并估计在给定年份内将购买多少个管组件。

**表2-1 train\_set.csv数据表字段介绍**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 变量含义 | 变量类型 | 缺失率 |
| tube\_assembly\_id | 名称 | 名称 | 0% |
| Supplier | 供应商 | 名称 | 0% |
| Quote\_date | 报价日期 | 名称 | 0% |
| Annual\_usage | 年度使用量 | 离散 | 0% |
| min\_order\_quantity | 最小下单数量 | 离散 | 0% |
| bracket\_pricing | 括号定价 | 是否判断 | 0% |
| Quantity | 数量 | 离散 | 0% |
| Cost | 价格（以quantity  为数量级） | 离散 | 0% |

**表2-2 test\_set.csv数据表字段介绍**

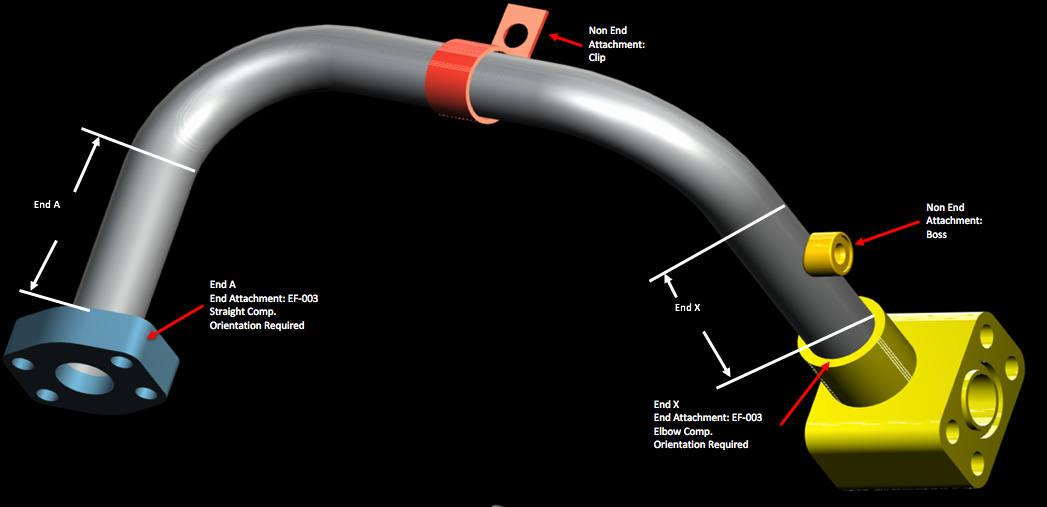
|  |  |  |  |
| --- | --- | --- | --- |
|  | 变量含义 | 变量类型 | 缺失率 |
| Id | 编号 | 连续 | 0% |
| tube\_assembly\_id | 名称 | 名称 | 0% |
| Supplier | 供应商 | 名称 | 0% |
| Quote\_date | 报价日期 | 名称 | 0% |
| Annual\_usage | 年度使用量 | 离散 | 0% |
| min\_order\_quantity | 最小下单数量 | 离散 | 0% |
| bracket\_pricing | 括号定价 | 是否判断 | 0% |
| Quantity | 数量 | 离散 | 0% |

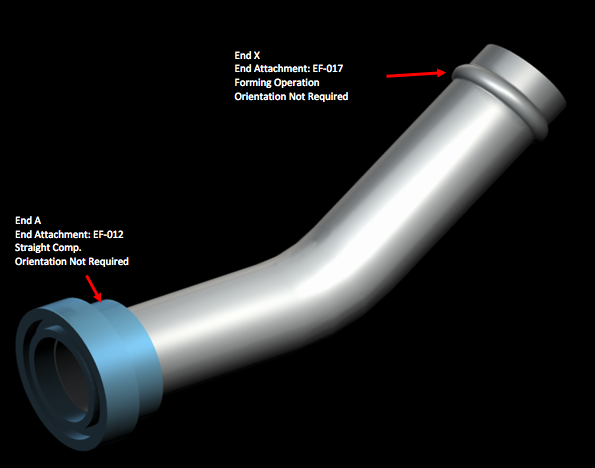
tube.csv：

该文件包含管组件的信息，这是比赛的主要焦点。管组件由多个部件组成。主要部件是具有特定直径，壁厚，长度，弯曲数量和弯曲半径的管子。管的任一端（End A或End X）通常具有某种形式的端部连接，允许管组件连接到其他部件。特殊的部件是指短长直末端（end\_a\_1x，end\_a\_2x，如果端部长度小于管直径的1倍或2倍则被需求）。其他部件可以永久地连接到管子上，例如凸台，支架或其他市场定制特征。

**表2-3 tube.csv数据表字段介绍**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 变量含义 | 变量类型 | 缺失率 |
| tube\_assembly\_id | 名称 | 名称 | 0% |
| material\_id | 材料编号 | 名称 | 0% |
| Diameter | 半径 | 离散 | 0% |
| Wall |  | 离散 | 0% |
| Length | 长度 | 离散 | 0% |
| num\_bends | 拐角数量 | 离散 | 0% |
| bend\_radius | 拐角半径 | 离散 | 0% |
| end\_a\_1x | 特征 | N/Y判断 | 0% |
| end\_a\_2x | 特征 | N/Y判断 | 0% |
| end\_x\_1x | 特征 | N/Y判断 | 0% |
| end\_x\_2x | 特征 | N/Y判断 | 0% |
| end\_a | 型号 | 名称 | 0% |
| end\_x | 型号 | 离散 | 0% |
| num\_boss | 特征 | 离散 | 0% |
| num\_bracket | 特征 | 离散 | 0% |
| Other | 特征 | 离散 | 0% |

**图2-1 tube 结构介绍**



**bill\_of\_materials.csv:**

该文件包含每个管组件上使用的组件列表及其数量

**表2-4 bill\_of\_material.csv数据表字段介绍**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 变量含义 | 变量类型 | 缺失率 |
| tube\_assembly\_id | 名称 | 名称 | 0% |
| component\_id\_i(1~8) | 名称 | 名称 | 0% |
| quantity\_i(1~8) | 数量 | 离散 | 0% |

**specs.csv：**

该文件包含管组件的唯一规格列表。这些可以指材料，工艺，防锈等。

**表2-5 specs.csv数据表字段介绍**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 变量含义 | 变量类型 | 缺失率 |
| tube\_assembly\_id | 名称 | 名称 | 0% |
| component\_id\_i(1~8) | 名称 | 名称 | 0% |
| quantity\_i(1~8) | 数量 | 离散 | 0% |

**tube\_end\_form.csv**

该文件主要包括了一些管的端部特征的list。

**表2-6 tube\_end\_form.csv数据表字段介绍**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 变量含义 | 变量类型 | 缺失率 |
| end\_form\_id | 名称 | 名称 | 0% |
| forming | 是否存在判断 | Yes/No | 0% |

**components.csv**

该文件包含所有使用组件的列表。Component\_type\_id指的是每个组件所属的类别。

**表2-7 componens.csv数据表字段介绍**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 变量含义 | 变量类型 | 缺失率 |
| component\_id | 名称 | 名称 | 0% |
| name | 名称 | 名称 | 0% |
| component\_type\_id | 名称 | 名称 | 0% |

**comp\_ [type].csv**

这些文件包含每个组件的信息。

**表2-8 comp\_ [type].csv数据表字段介绍**

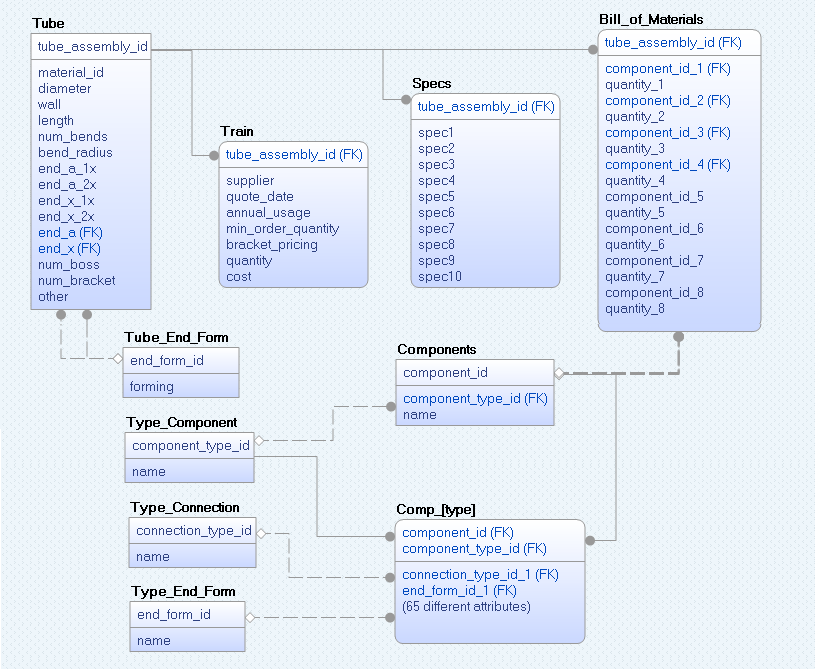
|  |  |  |  |
| --- | --- | --- | --- |
|  | 变量含义 | 变量类型 | 缺失率 |
| component\_id | 名称 | 名称 | 0% |
| component\_type\_id | 名称 | 名称 | 0% |
| 特性 | 规格，有无该性质等信息 | 名称 | 0% |

**type\_ [type].csv**

这些文件包含每个功能的名称。

**表2-9 type\_ [type].csv数据表字段介绍**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 变量含义 | 变量类型 | 缺失率 |
| [type]\_type\_id | 名称 | 名称 | 0% |
| name | 名称 | 名称 | 0% |

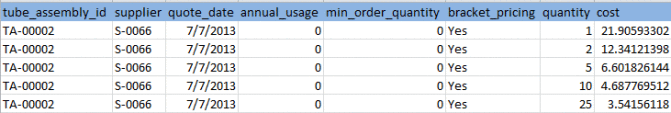
**2.2.3 数据描述性统计**

3. 优秀算法思路

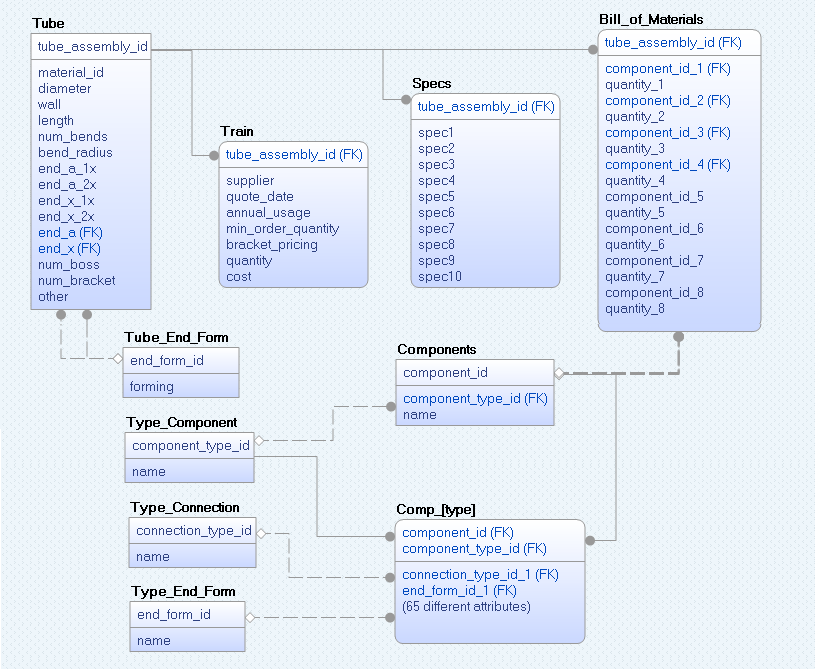
**3.1 方案一**

**3.1.1** 方案一数据预处理及特征工程部分方案

1) 对train\_set.csv的前几行分析：

[](https://vanishingcodes.files.wordpress.com/2015/09/train_data_view.png)

这是主要训练文件，其中包含用于培训的所有行。但是，此文件中没有足够的功能来训练您的模型并获得良好的预测。如果您能猜到，大多数有价值的信息可能隐藏在您需要从竞争中提供的所有其他数据中挖掘的tube\_assembly\_id col后面。

查看ER图（由EaswerC在竞赛[论坛中](https://www.kaggle.com/c/caterpillar-tube-pricing/forums/t/14963/entity-relationship-diagram-of-the-given-data)创建），了解所提供数据集中每个文件之间的复杂关系：

可以得出：

1. 每个tube\_assembly\_id代表一个管，其参数可以在文件tube.csv中找到（如直径，长度，墙壁等）。
2. 每个管可能包含0-4个不同的组件，这些组件可以位于文件bill\_of\_materials.csv中。
3. 最后，所有组件的参数都可以放在所有Comp\_ [type] .csv文件中，这是特征工程中最具挑战性的部分，因为所有comp文件的格式都不同，因此难以为所有组件创建一致的功能管子。

因而可以提出两类特征：管特征、组件特征

1. 管功能
2. 基于tube.csv，按每个tube\_assembly\_id分组，添加所有管参数，如'直径'，'墙'，'长度'等。
3. 基于tube\_end\_form.csv为'end\_x'和'end\_a'cols创建'形成'特征（逻辑T / F）
4. 对“供应商”功能进行分类，并将这些罕见级别转换为一个级别
5. 组件功能
6. 基于bill\_of\_materials.csv（bill），计算每个tube\_assembly中包含的所有组件的总权重
7. 根据账单，计算每个tube\_assembly中包含的所有组件的最大重量
8. 根据账单，计算 每个tube\_assembly中包含的所有组件的平均重量
9. 在components.csv文件中基于component\_type\_id对组件进行分类，并创建30个 component\_types cols，并为每个tube\_assembly，one-hot编码该tube\_assembly中包含的component\_type / component\_types，如果有的话
10. 根据所有comp\_ [type] .csv文件，创建'unique\_feature'功能（逻辑T / F col），指示每个tube\_assembly是否包含具有'unique\_feature'的组件。
11. 根据所有comp\_ [type] .csv文件，创建'orientation'功能（逻辑T / F col），指示每个tube\_assembly是否包含需要'orientation'的组件。
12. 从components.csv文件中，确定重要/重要的前10个特殊处理（例如，在CONNECTOR-SEAL中，SEAL被视为特殊处理）。为所有tube\_assemblies创建10个'特殊处理列'，为每个tube\_assembly创建一个热门编码，对其组件进行特殊处理。
    * 1. 方案一模型设计、建立部分方案
13. ***基于上述数据集的模型（1级）***

我在任何培训之前所做的预处理步骤是使用log（1 + cost）来转换成本。随后，来自论坛的人提到成本的根变换实际上比日志变换更好，所以我尝试了根变换（N = 15）和日志变换。我使用的第一个模型是xgboost。对于eta = 0.1和max\_depth = 8的1000轮，我一次运行得到0.227（CV和LB）。套袋 xgb实际上将结果提升到0.219！ （设置子样本率= 0.8）

我尝试过的另一个模型得到了相对较好的结果是Extra Trees（ R中的extraTrees包），它在LB和CV上都给出了大约0.23。所以第一个集合模型是0.8 \* xgb\_bagging + 0.2 \* extra\_trees，结果大约为0.2165。 - 1）第二个模型 - metaBagging - 代码可以在我的github中找到 - 给出大约0.2156。 - 2）。

（MetaBagging基本上是xgb的包装。然而，OOB用于训练一些基本模型（在我的情况下我使用extraTrees）而不是丢弃每个装袋轮次的OOB数据，然后将其预测堆叠为新的BAG和测试数据的元特征，然后我们使用BAG来训练元模型并预测测试数据。我们做了很多轮类似于套袋。meta Bagging的细节可以在这个帖子中找到，Kaggle Master Mike Kim为Otto Group Competition推出了这个着名的元装袋方法。

然而，我没有使用替换采样，而是使用了无需更换的采样，并使用了80％的采样率（以前100％用于替换采样）。剩下的20％是我的OOB数据，80％是BAG数据。使采样率成为可调参数有时会非常强大，特别是对于这种情况，并且由于使用的OOB数据和OOB数据较少，因此可以更快。）

1）和2）的整体给出大约0.215LB和 0.212CV， 我用作我的1级整体模型。

1. ***基于2级数据的模型***

现在我有了我的1级整体模型，接下来我做的是创建五重交叉预测作为元特征，并且仅使用这些预测特征来训练我的2级模型。

我用来创建这些五重预测的算法是xgboost（对数和根变换），神经网络的包装（h2o），额外的树，随机森林，svm，KNN（k范围从2到1024），线性模型和xgboost训练了前140个cols的PCA（90％）。

这些预测功能组合在一起成为我的2级数据集。然后我使用线性回归（主要模型），与其他模型如NN，xgb，rf和ET组合。第2级的最终预测是我的第二个整体模型，它给出 了0.216的LB和0.2125的CV。

1. ***最后的合奏***

我的1级和2级整体模型（50/50）的整体给出了我的LB为0.214和CV 0.210。

加入团队MDT（Nabil＆Adil）

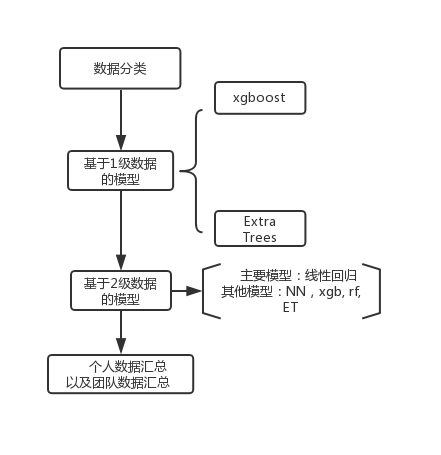
在这一刻，我可以说我的合奏和堆叠已经挤出了我的数据集中的每一滴潜能，并且很难提高得分。由于我的所有模型都是基于相同的数据集构建的，因此只有在我可以与另一个可以引入不同数据集的团队合并时才能实现实质性的改进。

我加入了MDT团队，通过使用我们最好的模型的简单集合，我们在LB上将分数提高到0.209！

**3.1.3** 方案一结果、排名等

方案结果： 0.209

方案排名： 7/1323

**3.1.4** 方案一算法流程图

**图3-1**

**3.2 方案二**

**3.2.1** 方案二数据预处理及特征工程部分方案

我们花了大约70％的时间在这上面限制了我们在模型上可以做的事情学习部分。这部分主要包括正确编码不同的分类特征，添加新功能，尝试一些功能选择，可视化功能的重要性及以上所有人都试图理解我们所做的事情。

* 1. 关于目标：
* 日志转换：大多数回归算法优化RMSE或MSE。从而，我们必须对目标应用日志变换（x - > log（x + 1））。
* 我们尝试了其他转换，如电源转换（如x - > x ^ 1/16，如在论坛中建议的那样）和混合模型，但它没有用。
  1. 数据库提供的数据。
* 订单功能：

1. 供应商：dummified（比简单的单热编码器好得多）

2. tube\_assembly\_id（保留相关号码：“TA-1624” - > 1624）

3. quote\_date：保留年，日（每年）和day\_of\_week

4. annual\_usage, min\_order\_quantity, bracket\_pricing, quantity, diff\_quantity =quantity-min\_order\_quantity

● 管特点：

1. material\_id：not dummified（保留相关号码）

2. 直径，墙，长度，num\_bends，bend\_radius

3. merged end\_a, end\_x with tube\_end\_form + added diff\_end (end\_a == end\_x)

4. 规格数量

●组件功能：

1. bill\_of\_materials merged with components: component\_id, component\_type\_id, weights of components, quantities (每个组件4个，因此4 \* 8个特征）

2. 我们添加：总重量，组件总数和不同数量组件。这些功能非常有价值。

* 1. 什么没有奏效：

1. dummifying day\_of\_week，月份

2. 体积，密度等......

3. 删除tube\_assembly\_id：id的顺序很重要!!

4. 保留所有规格

5. 方向特征，俯仰特征，形状特征（以及其他特定组件特征）

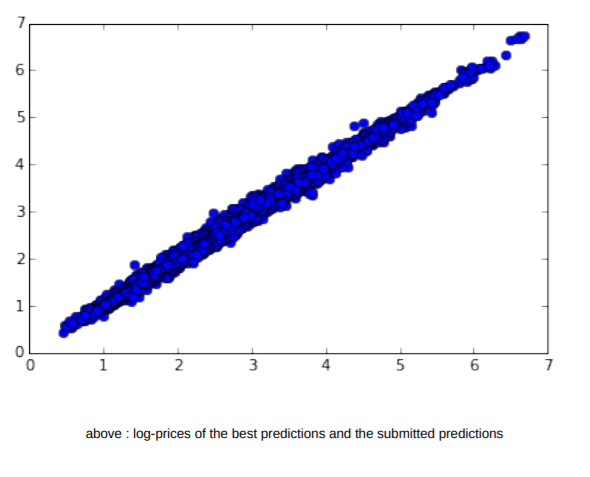
每当我们想要测试一个新功能时，我们就会在正确的CV中信任我们的分数（通常使用在tube\_assembly\_id上制作2个折叠）+在LB上得分

**3.2.2** 方案二模型设计、建立部分方案

* 1. 基于原始数据的模型(First stacked model):
* 15% of 15 AdaBoosted XGBoost
* 10% of 15 BaggingRegressor with base estimator = XGBoost
* 75% of different tuned XGBoosts : 3 different max\_depth.
  1. 基于1级数据的模型(Second stacked model):
* layer 1: RF + Adaboost (loss=linear) + Adaboost (loss=square) + XGBoost + ET）
* layer 2: linear regression

注意: 在重新学习所有数据的第一层之前使用3倍学习完成学习“可推广的”线性回归

* 1. 提交文件的“盲目”加权平均值：每次提交时，我们都会绘制预测我们最佳提交的价格（在Y轴上）和提交的预测（在X轴）。 我们还计算了皮尔森相关系数。 如果2个csv文件是“足够”去相关并给出了“好”的结果，我们混合并获得了最后结果



**图3-2-1**

**3.2.3** 方案二结果、排名等

方案结果：0.208742

方案排名 < 2%

图片包含 屏幕截图

自动生成的说明**3.2.4** 方案二算法流程图

**图3-2-2**

1. 算法比较

**表4-1 算法比较**

|  | **评估指标** | **特征工程** | **基础算法** | **基本库** |
| --- | --- | --- | --- | --- |
| **算法1** | **预测结果值大小** |  | **线性回归** | **XGboost，**  **Xgb** |
| **算法2** | **预测结果大小** |  | **线性回归** | XGboost，hyperopt package |

5. 总结与展望

**5.1 总结**

因为网上能找到的前几名的资料比较少，因而挑选了数据比较全的两位，但不管是阅读前几位的采访还是这次分享的两个方案，都使用了XGboost 可见其实用价值。

前期对数据的分析整理很重要。

**5.2 建模思路**

因为没有学习过XGboost，因而仅就模型建立提一哈想法：

1. 对数据分类（因为对应的特征属性过多）
2. 筛选出重要的属性
3. 根据筛选出的属性设置适当的权重
4. 利用线性回归计算