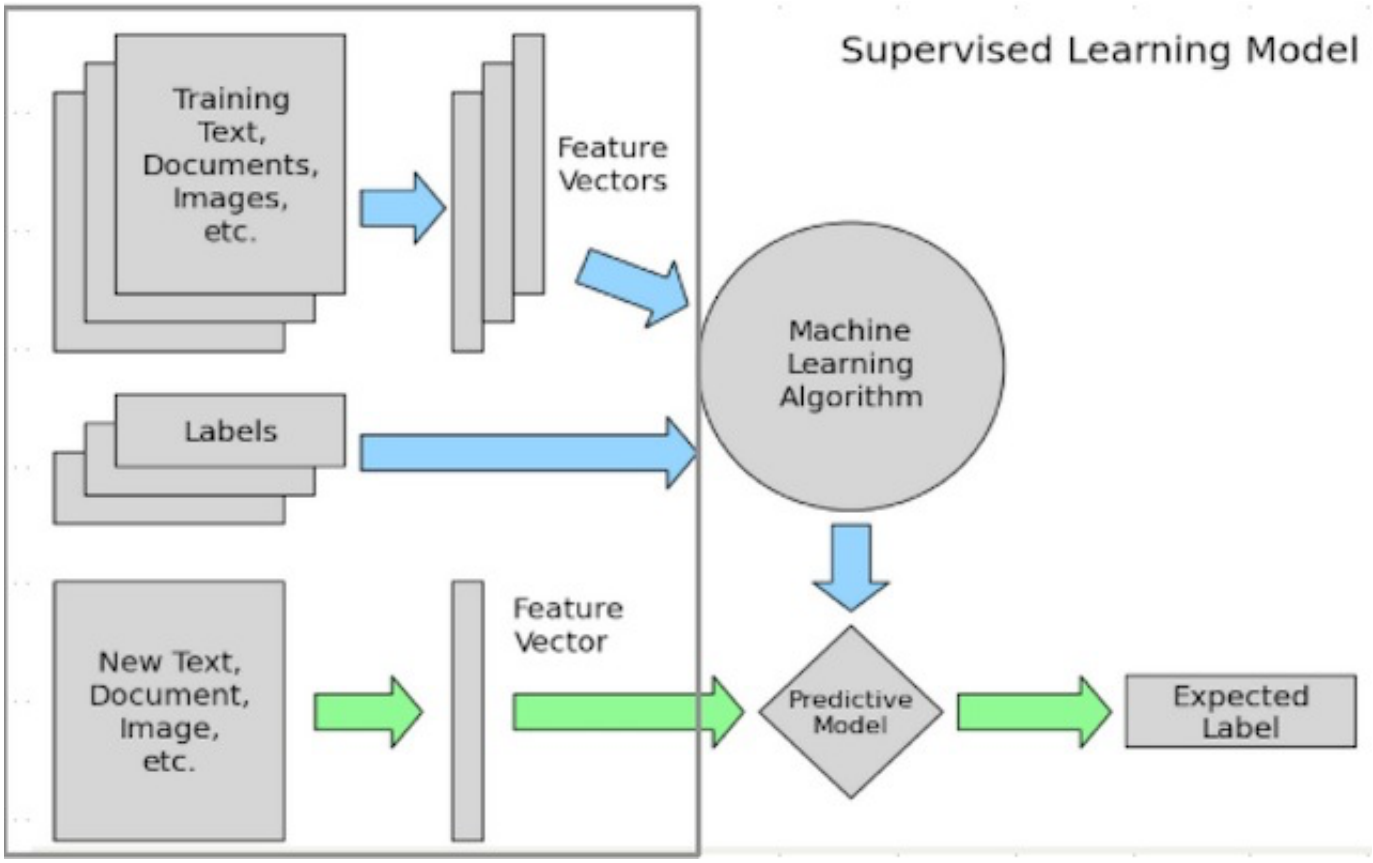
机器学习算法系列（40）：机器学习中的数据清洗与特征处理综述

一、背景

随着美团交易规模的逐步增大，积累下来的业务数据和交易数据越来越多，这些数据是美团做为一个团购平台最宝贵的财富。通过对这些数据的分析和挖掘，不仅能给美团业务发展方向提供决策支持，也为业务的迭代指明了方向。目前在美团的团购系统中大量地应用到了机器学习和数据挖掘技术，例如个性化推荐、筛选排序、搜索排序、用户建模等等，为公司创造了巨大的价值。本文主要介绍在美团的推荐与个性化团队实践中的数据清洗与特征挖掘方法。主要内容已经在内部公开课"机器学习InAction系列"讲过，本博客的内容主要是讲座内容的提炼和总结。

二、综述



如上图所示是一个经典的机器学习问题框架图。数据清洗和特征挖掘的工作是在灰色框中框出的部分，即“数据清洗=>特征，标注数据生成=>模型学习=>模型应用”中的前两个步骤。
灰色框中蓝色箭头对应的是离线处理部分。主要工作是

- 从原始数据，如文本、图像或者应用数据中清洗出特征数据和标注数据。

- 对清洗出的特征和标注数据进行处理，例如样本采样，样本调权，异常点去除，特征归一化处理，特征变化，特征组合等过程。最终生成的数据主要是供模型训练使用。

灰色框中绿色箭头对应的是在线处理的部分。所做的主要工作和离线处理的类似，主要的区别在于1.不需要清洗标注数据，只需要处理得到特征数据，在线模型使用特征数据预测出样本可能的标签。2.最终生成数据的用处，最终生成的数据主要用于模型的预测，而不是训练。
在离线的处理部分，可以进行较多的实验和迭代，尝试不同的样本采样、样本权重、特征处理方法、特征组合方法等，最终得到一个最优的方法，在离线评估得到好的结果后，最终将确定的方案在线上使用。
另外，由于在线和离线环境不同，存储数据、获取数据的方法存在较大的差异。例如离线数据获取可以将数据存储在Hadoop，批量地进行分析处理等操作，并且容忍一定的失败。而在线服务获取数据需要稳定、延时小等，可以将数据建入索引、存入KV存储系统等。后面在相应的部分会详细地介绍。

本文以点击下单率预测为例，结合实例来介绍如何进行数据清洗和特征处理。首先介绍下点击下单率预测任务，其业务目标是提高团购用户的用户体验，帮助用户更快更好地找到自己想买的单子。这个概念或者说目标看起来比较虚，我们需要将其转换成一个技术目标，便于度量和实现。最终确定的技术目标是点击下单率预估，去预测用户点击或者购买团购单的概率。我们将预测出来点击或者下单率高的单子排在前面，预测的越准确，用户在排序靠前的单子点击、下单的就越多，省去了用户反复翻页的开销，很快就能找到自己想要的单子。离线我们用常用的衡量排序结果的AUC指标，在线的我们通过ABTest来测试算法对下单率、用户转化率等指标的影响。

三、特征使用方案

在确定了目标之后，下一步，我们需要确定使用哪些数据来达到目标。需要事先梳理哪些特征数据可能与用户是否点击下单相关。我们可以借鉴一些业务经验，另外可以采用一些特征选择、特征分析等方法来辅助我们选择。具体的特征选择，特征分析等方法我们后面会详细介绍。从业务经验来判断，可能影响用户是否点击下单的因素有：

- 距离，很显然这是一个很重要的特征。如果购买一个离用户距离较远的单子，用户去消费这个单子需要付出很多的代价。 当然，也并不是没有买很远单子的用户，但是这个比例会比较小。
- 用户历史行为，对于老用户，之前可能在美团有过购买、点击等行为。 用户实时兴趣。
- 单子质量，上面的特征都是比较好衡量的，单子质量可能是更复杂的一个特征。
- 是否热门，用户评价人数，购买数等等。

在确定好要使用哪些数据之后，我们需要对使用数据的可用性进行评估，包括数据的获取难度，数据的规模，数据的准确率，数据的覆盖率等，

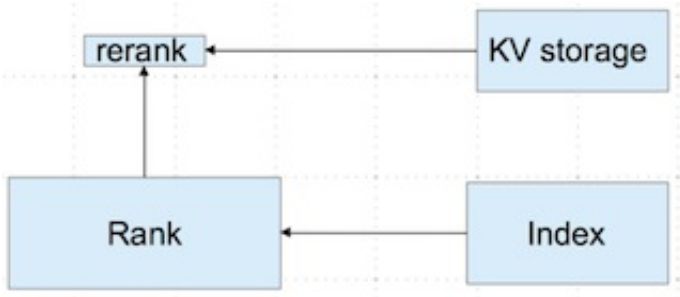
- 数据获取难度：例如获取用户id不难，但是获取用户年龄和性别较困难，因为用户注册或者

购买时，这些并不是必填项。即使填了也不完全准确。这些特征可能是通过额外的预测模型预测的，那就存在着模型精度的问题。

- 数据覆盖率：数据覆盖率也是一个重要的考量因素，例如距离特征，并不是所有用户的距离我们都能获取到。PC端的就没有距离，还有很多用户禁止使用它们的地理位置信息等。
- 用户历史行为，只有老用户才会有行为。
- 用户实时行为，如果用户刚打开app，还没有任何行为，同样面临着一个冷启动的问题。 数据的准确率
- 单子质量，用户性别等，都会有准确率的问题。

四、特征获取方案

Ok，在选定好要用的特征之后，我们需要考虑一个问题。就是这些数据从哪可以获取？只有获取了这些数据我们才能用上。否则，提一个不可能获取到的特征，获取不到，提了也是白提。下面就介绍下特征获取方案。

- 离线特征获取方案：离线可以使用海量的数据，借助于分布式文件存储平台，例如HDFS等，使用例如MapReduce，Spark等处理工具来处理海量的数据等。
- 在线特征获取方案：在线特征比较注重获取数据的延时，由于是在线服务，需要在非常短的时间内获取到相应的数据，对查找性能要求非常高，可以将数据存储在索引、kv存储等。而查找性能与数据的数据量会有矛盾，需要折衷处理，我们使用了特征分层获取方案，如下图所示。 

出于性能考虑。在粗排阶段，使用更基础的特征，数据直接建入索引。精排阶段，再使用一些个性化特征等。

五、特征与标注数据清洗

在了解特征数据放在哪儿、怎样获取之后。下一步就是考虑如何处理特征和标注数据了。下面3节都是主要讲的特征和标注处理方法

5.1 标注数据清洗

首先介绍下如何清洗特征数据，清洗特征数据方法可以分为离线清洗和在线清洗两种方法。

- 离线清洗数据：离线清洗优点是方便评估新特征效果，缺点是实时性差，与线上实时环境有

一定误差。对于实时特征难以训练得到恰当的权重。
- 在线清洗数据：在线清洗优点是实时性强，完全记录的线上实际数据，缺点是新特征加入需要一段时间做数据积累。

5.2 样本采样与样本过滤

特征数据只有在和标注数据合并之后，才能用来做为模型的训练。下面介绍下如何清洗标注数据。主要是数据采样和样本过滤。

数据采样，例如对于分类问题：选取正例，负例。对于回归问题，需要采集数据。对于采样得到的样本，根据需要，需要设定样本权重。当模型不能使用全部的数据来训练时，需要对数据进行采样，设定一定的采样率。采样的方法包括随机采样，固定比例采样等方法。

除了采样外，经常对样本还需要进行过滤，包括

- 1.结合业务情况进行数据的过滤，例如去除crawler抓取，spam，作弊等数据。
- 2.异常点检测，采用异常点检测算法对样本进行分析，常用的异常点检测算法包括 偏差检测，例如聚类，最近邻等。
  - 基于统计的异常点检测算法
  - 例如极差，四分位数间距，均差，标准差等，这种方法适合于挖掘单变量的数值型数据。全距(Range)，又称极差，是用来表示统计资料中的变异量数(measures of variation)，其最大值与最小值之间的差距；四分位距通常是用来构建箱形图，以及对概率分布的简要图表概述。
  - 基于距离的异常点检测算法，主要通过距离方法来检测异常点，将数据集中与大多数点之间距离大于某个阈值的点视为异常点，主要使用的距离度量方法有绝对距离（曼哈顿距离）、欧氏距离和马氏距离等方法。
  - 基于密度的异常点检测算法，考察当前点周围密度，可以发现局部异常点，例如LOF算法

六、特征分类

在分析完特征和标注的清洗方法之后，下面来具体介绍下特征的处理方法，先对特征进行分类，对于不同的特征应该有不同的处理方法。

根据不同的分类方法，可以将特征分为(1)Low level特征和High level特征。(2)稳定特征与动态特征。(3)二值特征、连续特征、枚举特征。

Low level特征是较低级别的特征，主要是原始特征，不需要或者需要非常少的人工处理和干预，例如文本特征中的词向量特征，图像特征中的像素点，用户id，商品id等。Low level特征一般维度比较高，不能用过于复杂的模型。High level特征是经过较复杂的处理，结合部分业务逻辑或

者规则、模型得到的特征，例如人工打分，模型打分等特征，可以用于较复杂的非线性模型。Low level 比较针对性，覆盖面小。长尾样本的预测值主要受high level特征影响。 高频样本的预测值主要受low level特征影响。

稳定特征是变化频率(更新频率)较少的特征，例如评价平均分，团购单价格等，在较长的时间段内都不会发生变化。动态特征是更新变化比较频繁的特征，有些甚至是实时计算得到的特征，例如距离特征，2小时销量等特征。或者叫做实时特征和非实时特征。针对两类特征的不同可以针对性地设计特征存储和更新方式，例如对于稳定特征，可以建入索引，较长时间更新一次，如果做缓存的话，缓存的时间可以较长。对于动态特征，需要实时计算或者准实时地更新数据，如果做缓存的话，缓存过期时间需要设置的较短。

二值特征主要是0/1特征，即特征只取两种值：0或者1，例如用户id特征：目前的id是否是某个特定的id，词向量特征：某个特定的词是否在文章中出现等等。连续值特征是取值为有理数的特征，特征取值个数不定，例如距离特征，特征取值为是0~正无穷。枚举值特征主要是特征有固定个数个可能值，例如今天周几，只有7个可能值：周1，周2，...，周日。在实际的使用中，我们可能对不同类型的特征进行转换，例如将枚举特征或者连续特征处理为二值特征。枚举特征处理为二值特征技巧：将枚举特征映射为多个特征，每个特征对应一个特定枚举值，例如今天周几，可以把它转换成7个二元特征：今天是否是周一，今天是否是周二，...，今天是否是周日。连续值处理为二值特征方法：先将连续值离散化（后面会介绍如何离散化)，再将离散化后的特征切分为N个二元特征，每个特征代表是否在这个区间内。

6.1 特征归一化，离散化，缺省值处理

主要用于单个特征的处理。

- 归一化
  不同的特征有不同的取值范围，在有些算法中，例如线性模型或者距离相关的模型像聚类模型、knn模型等，特征的取值范围会对最终的结果产生较大影响，例如二元特征的取值范围为[0，1]，而距离特征取值可能是[0，正无穷)，在实际使用中会对距离进行截断，例如[0，3000000]，但是这两个特征由于取值范围不一致导致了模型可能会更偏向于取值范围较大的特征，为了平衡取值范围不一致的特征，需要对特征进行归一化处理，将特征取值归一化到［0，1］区间。常用的归一化方法包括1.函数归一化，通过映射函数将特征取值映射到［0，1］区间，例如最大最小值归一化方法，是一种线性的映射。还有通过非线性函数的映射，例如log函数等。2.分维度归一化，可以使用最大最小归一化方法，但是最大最小值选取的是所属类别的最大最小值，即使用的是局部最大最小值，不是全局的最大最小值。3.排序归一化，不管原来的特征取值是什么样的，将特征按大小排序，根据特征所对应的序给予一个新的值。

- 离散化
  在上面介绍过连续值的取值空间可能是无穷的，为了便于表示和在模型中处理，需要对连续

值特征进行离散化处理。常用的离散化方法包括等值划分和等量划分。等值划分是将特征按照值域进行均分，每一段内的取值等同处理。例如某个特征的取值范围为[0，10]，我们可以将其划分为10段，[0，1)，[1，2)，...，[9，10)。等量划分是根据样本总数进行均分，每段等量个样本划分为1段。例如距离特征，取值范围［0，3000000］，现在需要切分成10段，如果按照等比例划分的话，会发现绝大部分样本都在第1段中。使用等量划分就会避免这种问题，最终可能的切分是[0，100)，[100，300)，[300，500)，..，[10000，3000000]，前面的区间划分比较密，后面的比较稀疏。

- 缺省值处理
  有些特征可能因为无法采样或者没有观测值而缺失，例如距离特征，用户可能禁止获取地理位置或者获取地理位置失败，此时需要对这些特征做特殊的处理，赋予一个缺省值。缺省值如何赋予，也有很多种方法。例如单独表示，众数，平均值等。

6.2 特征降维

在介绍特征降维之前，先介绍下特征升维。在机器学习中，有一个VC维理论。根据VC维理论，VC维越高，打散能力越强，可容许的模型复杂度越高。在低维不可分的数据，映射到高维是可分。可以想想，给你一堆物品，人脑是如何对这些物品进行分类，依然是找出这些物品的一些特征，例如：颜色，形状，大小，触感等等，然后根据这些特征对物品做以归类，这其实就是一个先升维，后划分的过程。比如我们人脑识别香蕉。可能首先我们发现香蕉是黄色的。这是在颜色这个维度的一个切分。但是很多东西都是黄色的啊，例如哈密瓜。那么怎么区分香蕉和哈密瓜呢？我们发现香蕉形状是弯曲的。而哈密瓜是圆形的，那么我们就可以用形状来把香蕉和哈密瓜划分开了，即引入一个新维度：形状，来区分。这就是一个从“颜色”一维特征升维到二维特征的例子。

那问题来了，既然升维后模型能力能变强，那么是不是特征维度越高越好呢？为什么要进行特征降维&特征选择？主要是出于如下考虑：1. 特征维数越高，模型越容易过拟合，此时更复杂的模型就不好用。2. 相互独立的特征维数越高，在模型不变的情况下，在测试集上达到相同的效果表现所需要的训练样本的数目就越大。 3. 特征数量增加带来的训练、测试以及存储的开销都会增大。4.在某些模型中，例如基于距离计算的模型KMeans，KNN等模型，在进行距离计算时，维度过高会影响精度和性能。5.可视化分析的需要。在低维的情况下，例如二维，三维，我们可以把数据绘制出来，可视化地看到数据。当维度增高时，就难以绘制出来了。在机器学习中，有一个非常经典的维度灾难的概念。用来描述当空间维度增加时，分析和组织高维空间，因体积指数增加而遇到各种问题场景。例如，100个平均分布的点能把一个单位区间以每个点距离不超过0.01采样；而当维度增加到10后，如果以相邻点距离不超过0.01小方格采样单位超一单位超正方体，则需要10^{20} 个采样点。

正是由于高维特征有如上描述的各种各样的问题，所以我们需要进行特征降维和特征选择等工作。特征降维常用的算法有PCA，LDA等。特征降维的目标是将高维空间中的数据集映射到低维空间数据，同时尽可能少地丢失信息，或者降维后的数据点尽可能地容易被区分

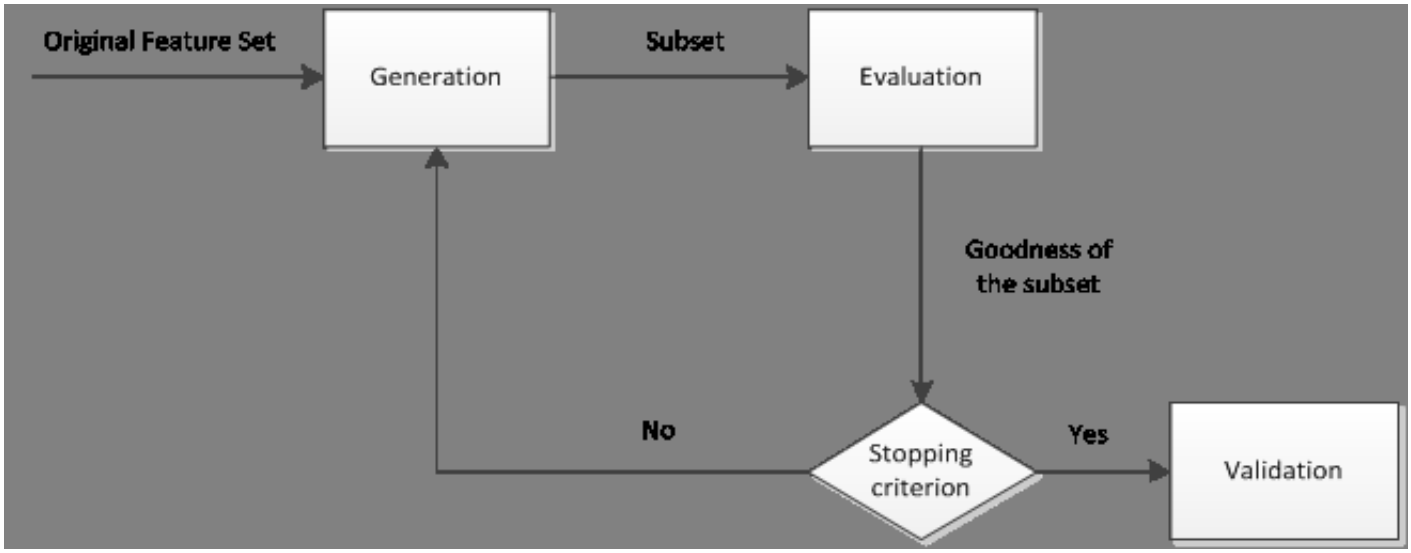
- PCA算法
  通过协方差矩阵的特征值分解能够得到数据的主成分，以二维特征为例，两个特征之间可能存在线性关系（例如运动的时速和秒速度），这样就造成了第二维信息是冗余的。PCA的目标是发现这种特征之间的线性关系，并去除。

- LDA算法
  考虑label，降维后的数据点尽可能地容易被区分

6.3 特征选择

特征选择的目标是寻找最优特征子集。特征选择能剔除不相关(irrelevant)或冗余(redundant)的特征，从而达到减少特征个数，提高模型精确度，减少运行时间的目的。另一方面，选取出真正相关的特征简化模型，协助理解数据产生的过程。

特征选择的一般过程如下图所示：



主要分为产生过程，评估过程，停止条件和验证过程。

6.3.1 特征选择-产生过程和生成特征子集方法

> 完全搜索(Complete)

- 广度优先搜索(Breadth First Search)：广度优先遍历特征子空间。枚举所有组合，穷举搜索，实用性不高。
- 分支限界搜索(Branch and Bound)：穷举基础上加入分支限界。例如：剪掉某些不可能搜索出比当前最优解更优的分支。
- 其他，如定向搜索 (Beam Search)，最优优先搜索 (Best First Search)等

> 启发式搜索(Heuristic)

- 序列前向选择(SFS ， Sequential Forward Selection)：从空集开始，每次加入一个选最

优。
- 序列后向选择(SBS ， Sequential Backward Selection)：从全集开始，每次减少一个选最优。
- 增L去R选择算法 (LRS ， Plus-L Minus-R Selection)：从空集开始，每次加入L个，减去R个，选最优（L>R)或者从全集开始，每次减去R个，增加L个，选最优(L<R)。

其他如双向搜索(BDS ， Bidirectional Search)，序列浮动选择(Sequential Floating Selection)等

> 随机搜索(Random)

- 随机产生序列选择算法(RGSS， Random Generation plus Sequential Selection)
- 随机产生一个特征子集，然后在该子集上执行SFS与SBS算法。
- 模拟退火算法(SA， Simulated Annealing)：以一定的概率来接受一个比当前解要差的解，而且这个概率随着时间推移逐渐降低
- 遗传算法(GA， Genetic Algorithms)：通过交叉、突变等操作繁殖出下一代特征子集，并且评分越高的特征子集被选中参加繁殖的概率越高。

随机算法共同缺点:依赖随机因素，有实验结果难重现。

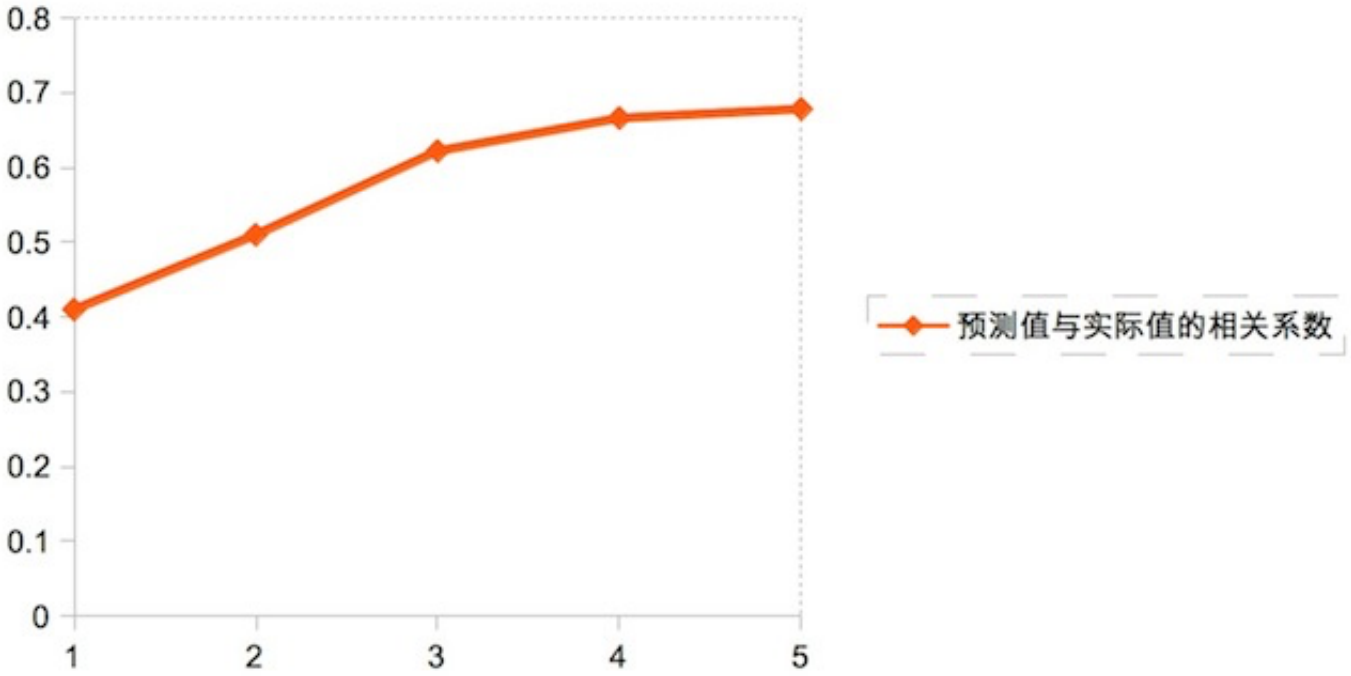
6.3.2 特征选择－有效性分析

对特征的有效性进行分析，得到各个特征的特征权重，根据是否与模型有关可以分为1.与模型相关特征权重，使用所有的特征数据训练出来模型，看在模型中各个特征的权重，由于需要训练出模型，模型相关的权重与此次学习所用的模型比较相关。不同的模型有不同的模型权重衡量方法。例如线性模型中，特征的权重系数等。2.与模型无关特征权重。主要分析特征与label的相关性，这样的分析是与这次学习所使用的模型无关的。与模型无关特征权重分析方法包括(1)交叉熵，(2)Information Gain，(3)Odds ratio，(4)互信息，(5)KL散度等

七、特征监控

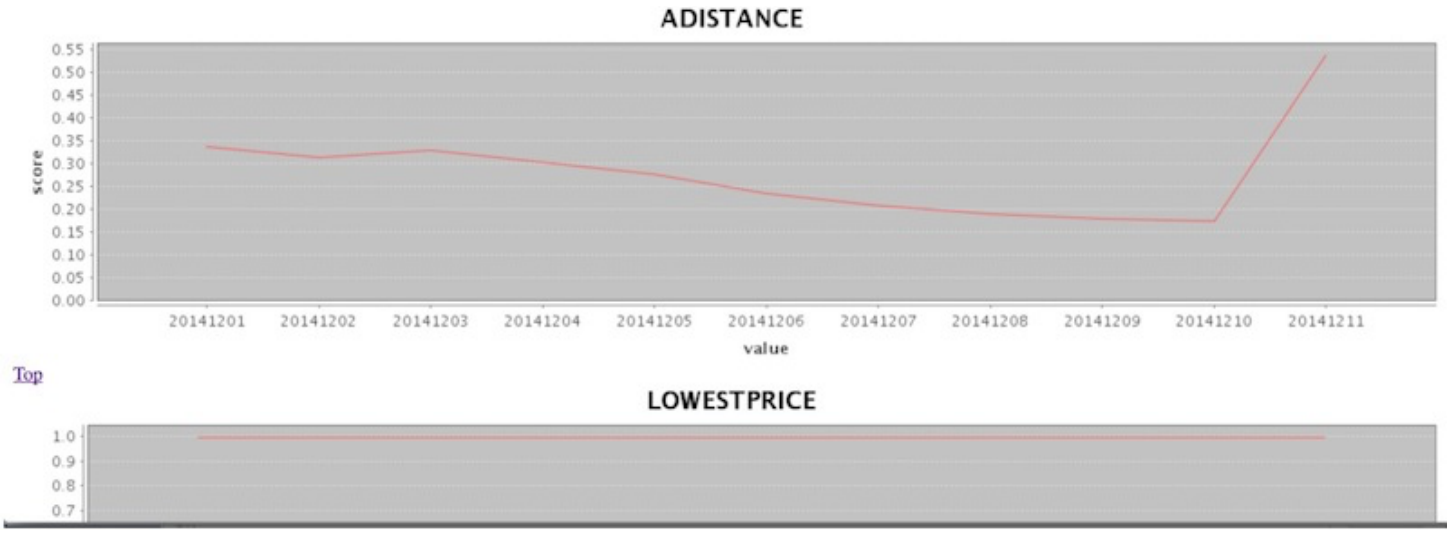
在机器学习任务中，特征非常重要。

- 个人经验，80%的效果由特征带来。下图是随着特征数的增加，最终模型预测值与实际值的相关系数变化。



- 对于重要的特征进行监控与有效性分析，了解模型所用的特征是否存在问题，当某个特别重要的特征出问题时，需要做好备案，防止灾难性结果。需要建立特征有效性的长效监控机制

我们对关键特征进行了监控，下面特征监控界面的一个截图。通过监控我们发现有一个特征的覆盖率每天都在下降，与特征数据提供方联系之后，发现特征数据提供方的数据源存在着问题，在修复问题之后，该特征恢复正常并且覆盖率有了较大提升。



在发现特征出现异常时，我们会及时采取措施，对服务进行降级处理，并联系特征数据的提供方尽快修复。对于特征数据生成过程中缺乏监控的情况也会督促做好监控，在源头解决问题。