机器学习算法系列（20）：项目模型优化四要素

title: 机器学习算法系列（20）：项目模型优化四要素
date: 2017-06-16 23:14:45
categories: 机器学习
tags:
- 业务
- 特征
- 数据
- 模型

mathjax2: true

本文转载自美团点评技术团队博客，该文以业界视角介绍了机器学习如何发挥其实际价值。作者胡渼，目前是美团算法工程师，毕业于哥伦比亚大学。先后在携程、支付宝、美团从事算法开发工作。了解风控、基因、旅游、即时物流相关问题的行业领先算法方案与流程。

一、机器学习工程师的知识图谱

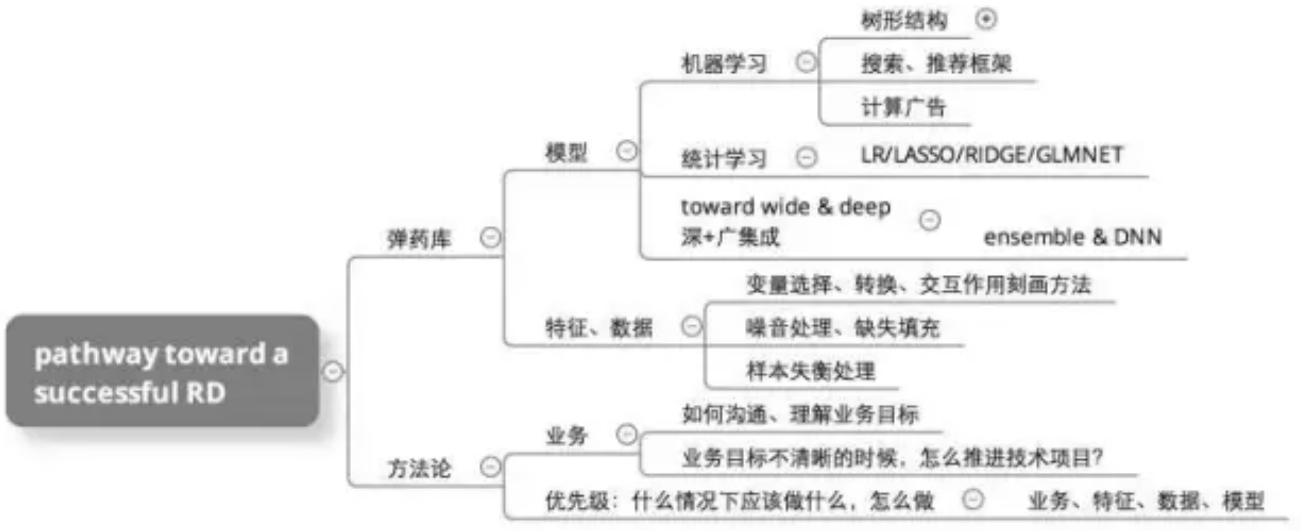

图1 机器学习工程师的知识图谱

上图列出了我认为一个成功的机器学习工程师需要关注和积累的点。机器学习实践中，我们平时

都在积累自己的“弹药库”：分类、回归、无监督模型、Kaggle上特征变换的黑魔法、样本失衡的处理办法、缺失值填充.....这些大概可以归类成模型和特征两个点。我们需要参考成熟的做法、论文，并自己实现，此外还需要多反思自己方法上是否还可以改进。如果模型和特征这俩个点都已经做的很好了，你就拥有了一张绿卡，能跨过在数据相关行业发挥模型技术价值的准入门槛。

在这个时候，比较关键的一步，就是搞笑的技术变现能力。

所谓高效，就是解决业务核心问题的专业能力。本文将描述这些专业能力，也就是模型优化的四个要素：模型、数据、特征、业务，还有更重要的，就是他们在模型项目中的优先级。

二、模型项目推进的四要素

项目推进过程中，四个要素相互之间的优先级大致是：业务>特征>数据>模型。

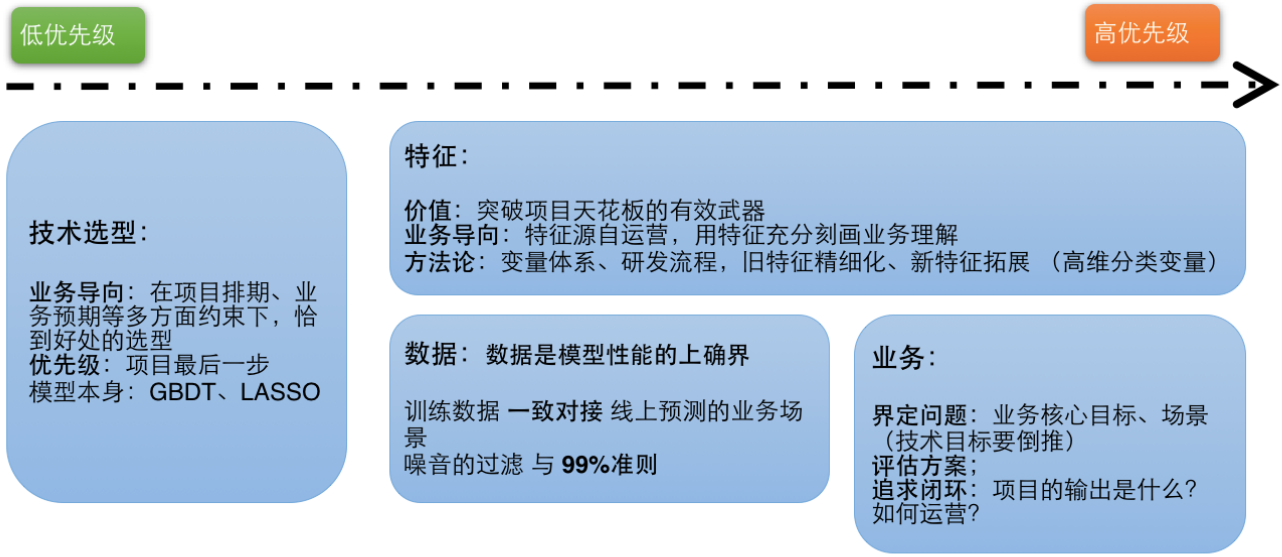

图2 四要素解决问题细分+优先级

2.1 业务

一个模型项目有好的技术选型、完备的特征体系、高质量的数据一定是很加分的，不过真正决定项目好与坏还有一个大前提，就是在这个项目的技术目标是否在解决当下核心业务问题。

业务问题包含两个方面：业务KPI和Deadline。举个例子，业务问题在两周之内降低目前手机丢失带来的支付宝销赃风险。这时如果你的方案是研发手机丢失的核心特征，比如改密是否合理，基本上就死的很惨，因为两周根本完不成，改密合理性也未必是模型优化好的切入点；反之，如果你的方案是和运营同学看bad case，梳理现阶段的作案通用手段，并通过分析上线一个简单模型或者业务规则的补丁，就明智很多。如果上线之后，案件量真掉下来了，就算你的方案准确率很糟糕、方法很low，但你解决了业务问题，这才是最重要的。

虽然业务目标很关键，不过一般讲，业务运营同学真的不太懂得如何和技术有效的沟通业务目

标，比如：

- 我们想做一个线下门店风险评级的项目，希望运营通过反作弊模型角度帮我们给门店打个分，这个分数包含的问题有：风险是怎么定义的、为什么要做风险评级、更大的业务目标是什么、怎么排期的、这个风险和我们反作弊模型之间的腋窝你是怎么看的?
- 做一个区域未来10min的配送时间预估模型。我们想通过运营的模型衡量在恶劣天气的时候每个区域的运力是否被击穿（业务现状和排期？运力被击穿可以扫下盲吗？运力击穿和配送时间之间是个什么业务逻辑、时间预估是刻画运力紧张度的最有效手段么？业务的关键场景是恶劣天气的话，我们仅仅训练恶劣天气场景的时间预估模型是否就好了？）

为了保证整个技术项目没有做偏，项目一开始一定要和业务聊清楚三件事情：

1. 业务核心问题、关键场景是什么。

2. 如何评估该项目的成功，指标是什么。

3. 通过项目输出什么关键信息给到业务，业务如何运营这个信息从而达到业务目标。

项目过程中，也要时刻回到业务，检查项目的健康度。

2.2 数据与特征

要说正确的业务理解和切入，在为技术项目保驾护航，数据、特征便是一个模型项目性能方面的天花板。garbage in， garbage out 就在说这个问题。

这两天有位听众微信问我一个很难回答的问题，大概意思是，数据是特征拼起来构成的集合嘛，所以这不是两个要素。从逻辑上面讲，数据的确是一列一列的特征，不过数据与特征在概念层面是不同的：数据是已经采集的信息，特征是以兼容模型、最优化为目标对数据进行加工。就比如通过word2vec将非结构化数据结构化，就是将数据转化为特征的过程。

所以，我更认为特征工程是基于数据的一个非常精细、刻意的加工过程。从传统的特征转换、交互，到embedding、word2vec、高维分类变量数值化，最终目的都是更好的去利用现有的数据。之前有聊到的将推荐算法引入有监督学习模型优化中的做法，就是在把两个本不可用的高维ID类变量变成可用的数值变量。

观察到自己和童鞋们在特征工程中会遇到一些普遍问题，比如，特征设计不全面，没有耐心把现有特征做得细致……也整理出来一套方法论，仅供参考：

图3 变量体系、研发流程

在特征设计的时候，有两个点可以帮助我们把特征想的更全面：

1. 现有的基础数据

2. 业务“二维图”

这两个方面的整合，就是一个变量的体系。变量（特征），从技术层面是加工数据，而从业务层面实际在反应RD的业务理解和数据刻画业务能力。“二维图”，实际上未必是二维的，更重要的是我们需要把业务整个流程抽象成几个核心的维度，举几个例子：

这两个方面的整合，就是一个变量的体系。变量（特征），从技术层面是加工数据，而从业务层面实际在反应RD的业务理解和数据刻画业务能力。“二维图”，实际上未必是二维的，更重要的是我们需要把业务整个流程抽象成几个核心的维度，举几个例子：

外卖配送时间业务（维度甲：配送的环节，骑手到点、商家出餐、骑手配送、交付用户；维度乙：颗粒度，订单粒度、商家粒度、区域城市粒度；维度丙：配送类型，众包、自营……）。

反作弊变量体系（维度甲：作弊环节，登录、注册、实名、转账、交易、参与营销活动、改密……；维度乙：作弊介质，账户、设备、IP、WiFi、银行卡……）。

通过这些维度，你就可以展开一个“二维图”，把现有你可以想到的特征填上去，你一定会发现很多空白，比如下图，那么哪里还是特征设计的盲点就一目了然：

图4 账户维度在转账、红包方面的特征很少；没有考虑WiFi这个媒介；客满与事件数据没考虑

数据和特征决定了模型性能的天花板。deep learning当下在图像、语音、机器翻译、自动驾驶等领域非常火，但是 deep learning在生物信息、基因学这个领域就不是热词：这背后是因为在前者，我们已经知道数据从哪里来，怎么采集，这些数据带来的信息基本满足了模型做非常准确的识别；而后者，即便有了上亿个人体碱基构成的基因编码，技术选型还是不能长驱直入——超高的数据采集成本，人后天的行为数据的获取壁垒等一系列的问题，注定当下这个阶段在生物信息领域，人工智能能发出的声音很微弱，更大的舞台留给了生物学、临床医学、统计学。

2.3 模型

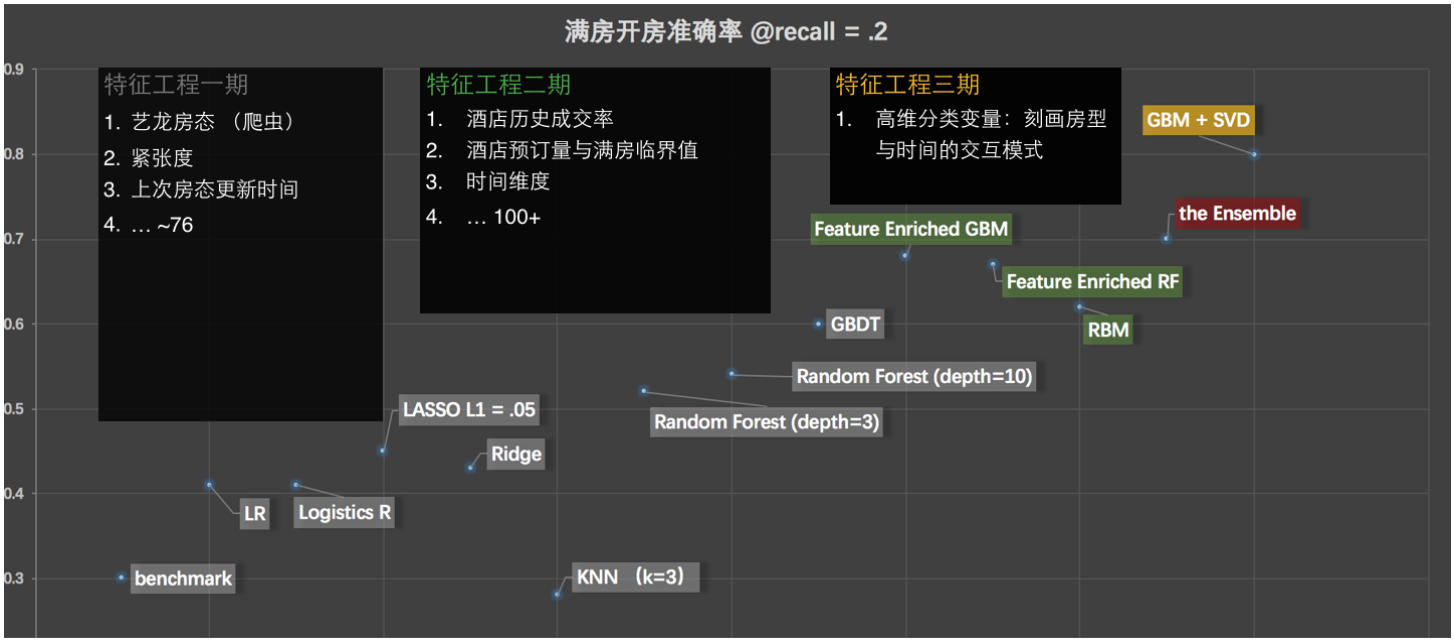

图5 满房开房的技术选型、特征工程roadmap

模型这件事儿，许多时候追求的不仅仅是准确率，通常还有业务这一层更大的约束。如果你在做一些需要强业务可解释的模型，比如定价和反作弊，那实在没必要上一个黑箱模型来为难业务。这时候，统计学习模型就很有用。

这种情况下，比拼性能的话，我觉得下面这个不等式通常成立：`Glmnet>LASSO>=Ridge>LR/Logistic`。相比最基本的LR/Logistic，ridge通过正则化约束缓解了LR在过拟合方面的问题，lasso更是通过L1约束做类似变量选择的工作。

不过两个算法的痛点是很难决定最优的约束强度，Glmnet是Stanford给出的一套非常高效的解决方案。所以目前，我认为线性结构的模型，Glmnet的痛点是最少的，而且在R、Python、Spark上面都开源了。

如果我们开发复杂模型，通常成立第二个不等式`RF（Random Forest，随机森林）<= GBDT <= XGBoost`。拿数据说话，29个Kaggle公开的winner solution里面，17个使用了类似GBDT这样的Boosting框架，其次是 DNN（Deep Neural Network，深度神经网络），RF的做法在Kaggle里

面非常少见。

RF和GBDT两个算法的雏形是CART（Classification And Regression Trees），由L Breiman和J Friedman两位作者在1984年合作推出。但是在90年代在发展模型集成思想the ensemble的时候，两位作者代表着两个至今也很主流的派系：stacking/ Bagging & Boosting。

一种是把相互独立的CART（randomized variables，bootstrap samples）水平铺开，一种是深耕的Boosting，在拟合完整体后更有在局部长尾精细刻画的能力。同时，GBDT模型相比RF更加简单，内存占用小，这都是业界喜欢的性质。XGBoost在模型的轻量化和快速训练上又做了进一步的工作，也是目前我们比较喜欢尝试的模型。