

推荐系统实用分析技巧

纳米酱 搜索与推荐Wiki 2020-09-21



点击标题下「[搜索与推荐Wiki](#)」可快速关注

▼ 精彩推荐 ▼

- 1、计算广告与推荐系统有哪些区别？使用的主流模型有哪些？
- 2、独孤九剑：算法模型训练的一般流程
- 3、特征工程 | 文本特征处理的四大类主流方法
- 4、从DSSM语义匹配到Google的双塔深度模型召回和广告场景中的双塔模型思考
- 5、最差的算法工程师也不过如此了
- 6、算法工程师的数学基础 | 如何理解概率分布函数和概率密度函数

作者：知乎 / 纳米酱（文末「阅读原文」查看原作内容）

这篇文章主要讨论推荐系统的分析技巧，杜绝无脑调参，探讨如何靠简单无脑而且 low 到爆的办法，快速搞一把，做到指标增长。

本来我想取一个高大上的题目：推荐系统0-1高速增长打法，这种互联网style强烈的题目，让我感觉我的格局大，ego也很大，所以算了，写一份实用分析手册，让我格局很小，ego也很小，比较符合我当前这种水平（**实打实不玩虚的，作者**）。

下面以问答形式解答一线常见问题和技巧手段。

问：指标上不去，rank加特征能够提升吗？

答：这是一个常见的疑惑，大家指标卡在瓶颈的时候，很容易迷信大力出奇迹的方案。其实，rank并不是提升指标的工具，你加多少特征，本质只是方便 rank 更好地还原系统分布的工具，策略才是提供增长，带动系统往良性（黄赌毒方向）发展的利器。要分析系统的指标瓶颈是不是卡在rank缺特征上，就需要从各种角度去分析，常见的分析方案是考察带条件的copc。

分析办法：把rank分数分成若干区间，每个区间统计真实的ctr，更进一步，可以拆分成多个桶，比如按照某个特征拆分成 A，B两组，单独统计每组的真实ctr。

分数区间	A桶真实 ctr	A桶pv	B桶真实 ctr	B桶pv
0.0-0.1	0.001	113516	0.002	2623626
0.1-0.2	0.002	15156	0.001	262365
0.2-0.3	0.005	31737	0.004	426256
0.3-0.4	0.015	11111	0.02	34612
0.4-0.5	0.035	14688	0.034	4545
0.5-0.6	0.025	1824684	0.02	23525223
0.6-0.7	0.055	8364	0.06	525
0.7-0.8	0.055	194795	0.065	453245
0.8-0.9	0.045	13975	0.07	4545
0.9-1.0	0.065	3552	0.08	35335

知乎·@纳米酱

无非出现如下几种情况：

rank分数单调递增，ctr没有单调递增：这个原因多半是你线上线下分布不一致导致的，道理很简单，如果你的rank真的拟合好了分布，没理由高分数区间段的ctr会低于低分数区间。这个不一致，有可能是你特征没做好线上线下的统一，也有可能是你模型没拟合好线上的分布。总之这种情况，你先别急着加特征，先把线上线下特征梳理一下，看看分布是否一致，或者模型训练是不是有问题。

rank 分数单调递增，ctr 单调递增，但是增长非常慢：比如说0.9-1.0区间的ctr 仅仅比0.3-0.4区间的 ctr 高一丁点。这个原因才是你模型缺特征，尤其是缺乏活跃用户的特征，通常活跃用户，系统 rank 为了指标，会使用很重的行为画像作为特征，很容易放大历史点击记录，更加倾向于把他们排上去，如果你高估了该用户的点击倾向，就会导致分数给的很高，但是现实用户不怎么点的现象

这个也会触发新的问题，也就是常说的离线AUC很高，但是线上没效果，道理都一样，你模型仅仅是把正负样本的间隔拉开了，并没有真正改善用户看到的内容和布局，才导致高分段ctr不见增长。

rank分数单调递增，ctr也单调递增，但是 A，B两组的ctr比值差异过大：比如 A，B表示上午和下午，如果这两个时间段，同一个分数区间的ctr差异过大，说明模型对时间这个维度的建模不足，需要进一步改善。

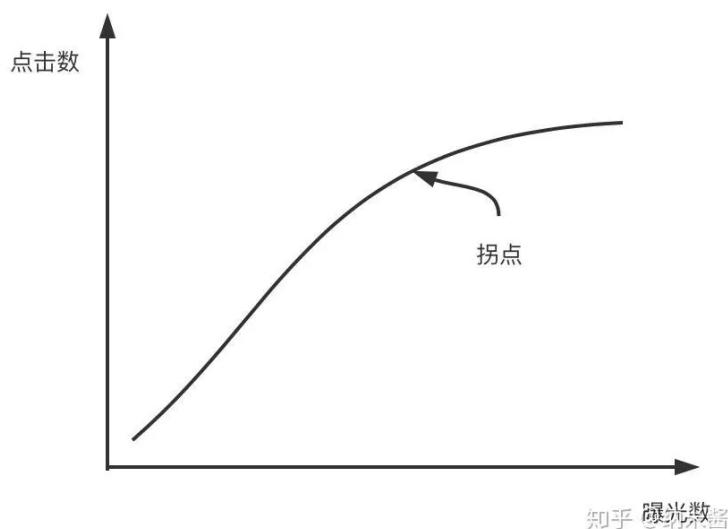
数学很好证明：假设模型分布是 $q(y|A)$, $q(y|B)$ ，真实的线上分布是 $p(y|A)$, $p(y|B)$ ，A，B处于同一个分数段，数学上等价于： $q(y|A) = q(y|B)$ ，由于你假设你模型正确拟合了真实分布，也就是 $q(y|A) = p(y|A)$, $q(y|B) = p(y|B)$ ，但是现实上 A，B的ctr并不相等， $P(y|A) \neq p(y|B)$ ，故此，你模型正确拟合A，B两个条件分布的假设不成立。

这种分组copc的技巧，是一种前期快速判断rank的不足之处，精准化打击系统，对于不见兔子不撒鹰的主，有理由推动对方更快地推进业务迭代

rank分数单调递增，ctr单调递增，各种维度分组下的ctr比值也接近平稳：恭喜你，到达这一步，表示你 rank 几乎没事情可做了，你剩下要做就是优化召回，在策略上引导你rank往新的产品思路上走，在更加高的层面带动系统往良性地方发展（黄赌毒方向）。

问：PV增大，但是CTR跌的很厉害

答：数学上，PV增加CTR就会跌，但一般都会是常数，如果PV增加CTR跌太厉害，你就需要警惕你的投放人群了。



投入过多曝光，会导致点击增长变缓

数学上来说，假设你有 A，B 两个人群，A 人群活跃喜欢点击，ctr 高，B 人群不太活跃，不怎么点击，ctr 比较低。如果是随机投放，CTR 一般来说都是比较平稳的。但是现实 rank 如果做的不好，都喜欢向 A 这种活跃人群投放大量同类内容，甚至高热内容，这些用户短时间内接受到大量同质的内容冲击后，A 人群兴趣饱和，贡献的点击就会下降。然后，系统随后的收益都是 B 人群零零星星提供的，导致 PV 一直增加，但是点击数不见增长。

另外一种可能的情况是，你投放的用户注意力发生转移，比如同一个时间内，旁边还有活动位置啥的干扰用户，也会导致 PV 上去，点击不见增长。

如果老板要你不择手段保 CTR，可以参考广告 pacing 思路，在短时间内，某个 item 触达的优质用户达到一定峰值，就需要退场冷却一下，防止一直被 rank 推到高位，消费大量注意力和曝光坑位。但是这个办法指标不治本，毕竟为了粘性，人们喜欢点的还是要多推。

问：如何寻找快速提升指标的策略

答：有一些野路子可以提供，说其是野路子，着实是没法有通用的解法，每个人都有自己的一套特殊的秘方。其次，生产线大部分策略都是被动产生，比如修各种 bad case，或者紧急针对某个单一业务指标做提升，或者产品拍了脑袋找你改进样式啥的。主动寻找增量策略本身是一个比较老中医的方案。

一种做法是复制上下文环境：这种技巧需要花钱买教训收集一定的反馈。打个比方，你要提升用户点击视频的概率，前期你并不知道那些用户爱看视频，所以在不同时间点，人群和位置上，随机试投了一阵，收益当然不佳。然后你认为这些随机当中，还是有一些样本误打误撞到最佳的策略上的，于是你把用户分成A，B两组，A组用户特爱点视频（你认为他们爱点，是因为恰好蒙对了策略环境），B组用户不怎么点。

复制上下文环境就是说，给B人群营造A人群的环境。数学上来说，就是统计一下A组的环境s都是啥（比如说视频都在啥位置，一次性投放视频的数目，视频热度等），然后想办法把这些上下文迁移到B组人群身上。

迁移需要一定技巧去离线预估效益，以决定要不要上线该策略，比如说A用户上下文环境分布为 $p(s|A)$ ，B用户上下文环境分布为 $p(s|B)$ ，强行迁移A的环境分布到B上，得到的B组的预期收益为：

$$\sum_s R(s) \times \min \left(\frac{p(s|B)}{p(s|A)}, 1 \right) \times p(s|A)$$

其中min里面是带截断的IPS分数，如果离线评测凑合。上线以后就靠 $\arg\max p(s|A)$ 生成你的策略环境。

这种办法缺点在于，A，B两组很可能天生就是两批用户，以至于你给B人群营造A人群的环境，对方一样不鸟你。

第二种是增强关联：这种技巧抽象来说，就是溯源，找到影响某个指标，最有可能的特征A，然后强化该指标和该特征的关系，需要和rank配合着打。

问：如何统计曝光次数少的 item 的热量

答：一般来说，很多item的曝光次数可能只有数十次，高热度的item曝光可能是上万，甚至百万次，曝光过低的item，只要产生少数几次点击，其ctr就有可能非常高，甚至吊打高热item的点击率，统计学上针对这种问题，一般是采取 wilson ctr纠正，但是现实来说，wilson ctr非常不靠谱，曝光低的item，大概率是你精准投放人群导致的，并不满足wilson ctr随机投放的基本假设。确切来说，我们要分人群去统计相对的ctr，消除投放人群的bias。

方法：假设item A被投放给N个人，曝光200次，产生10次点击，同时，这N个人当中，高热item B给他们曝光了100000次，产生 900 次点击。所以，A和B 在同一批人群当中的 ctr 分别是： $(10 / 200, 900 / 100000)$ ，一般我们认为高热的 item 都是无关个性化的，比如热点新闻，促销商品，黄色暴力内容，大家都爱点，高热item的点击率和投放人群的关系不是很大，几乎人人都会点，可以作为CTR本底。扣除这种ctr表示，相比大众货，用户更喜欢点那些item，用这种相对的ctr作为item热度的衡量。

— The end —



真正的努力，都不喧嚣！



搜索与推荐Wiki

All In CTR、DL、ML、RL、NLP

 分享，点赞，在看，安排一下？

阅读原文

喜欢此内容的人还喜欢

转化率预估中的贝叶斯平滑

搜索与推荐Wiki