Tencent 腾讯 | DataFun.

因果推断在游戏中的应用

房栋 | 腾讯游戏 专家数据科学家

CONTENTS

Tencent 腾讯 | DataFun.

01

游戏中的因果推断：挑战与解决方案

游戏中的因果推断：挑战与解决方案

问题

游戏场景中

- 因为用户体验及隐私个方面的因素，通常是缺乏实验数据
- 而观察数据中的干预不是随机的，带有人工运营或算法的选择偏差
  - 高活跃用户与低活跃用户被过度干预

可行性方案

利用观测数据：

- 可使用ATT（干预组平均处理效应）$ATT = E[Y_1 - Y_0|T = 1]$来评估对受到干预人群的效应
  - 例如使用倾向性得分匹配（PSM），PSM可以将干预组和对照组进行一对一匹配，并且可以提取匹配后的用户个体
- 可使用**因果推断**计算ATE（平均处理效应）$ATE = E[Y_1 - Y_0]$来评估整体效应，需要通过加权的方式使得样本分布均衡
  - 备选方案有Inverse-Probability-Treatment-Weighting(IPTW) / Double-Machine-Learning(DML) / Double-Robust-Estimator(DRE) / X-Learner
  - 例如使用双重稳健估计（DRE），由于在业务中可能无法覆盖全部的混淆因子，DRE在这种场景下更加稳健，并且DRE可以在倾向性分数预测不准的情况下，通过结果预测来调整

游戏中的因果推断：挑战与解决方案

技术挑战

○ 一般因果推断无法解决我们业务中遇到的数据量巨大的问题

- 常见的因果推断工具集例如微软econml, dowhy和uber causalml均不是分布式实现

○ 众多业务均需要精细化运营策略，多场景的数据量挑战巨大

解决方案

○ 面对游戏中的大规模推断场景，本演讲将涵盖以下三个方面：

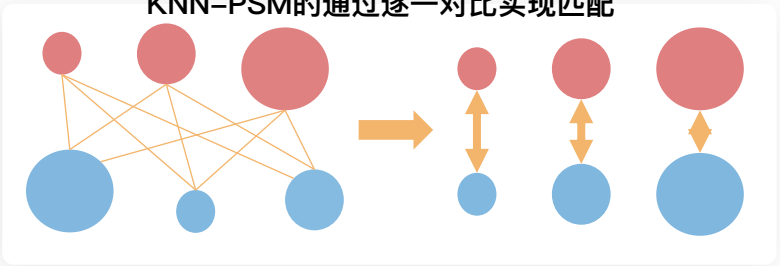
- 分布式低复杂度倾向性分数匹配
- 分布式鲁棒双重稳健估计
- 分布式面板双重差分

02

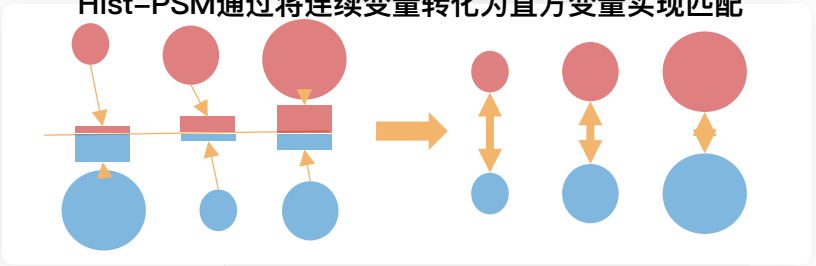
分布式低复杂度倾向性分数匹配

分布式低复杂度倾向性分数匹配（Hist-PSM）

KNN-PSM的通过逐一对比实现匹配



Hist-PSM通过将连续变量转化为直方变量实现匹配



算法
HistPSM

输入
(X, t, y, K)

输出
匹配人群MatchingDf

步骤1：计算Propensity Score（PS）

步骤2： PS分桶: 将每一个实验组与对照组的个体的连续PS映射到K个PS分桶

步骤3：计算实验组与对照组在每个PS分桶的个体数量

步骤4：计算每个PS分桶的阈值：取每个PS分桶中实验组与对照组中的最小个体数量

步骤5：基于PS分桶阈值过滤实验组数据D1：在实验组的每个PS分桶中，随机提取阈值数量的个体

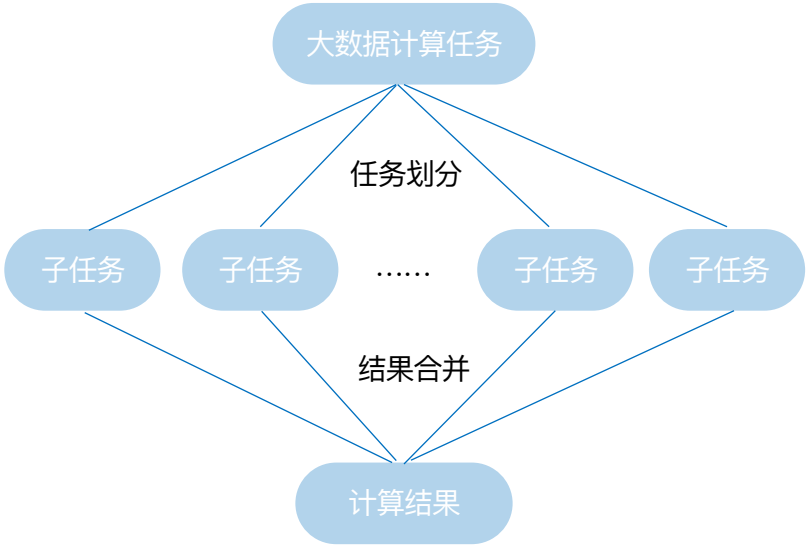
步骤6：基于PS分桶阈值过滤对照组数据D0：在对照组的每个PS分桶中，随机提取阈值数量的个体

步骤7：合并数据D0与D1，输出MatchingDf

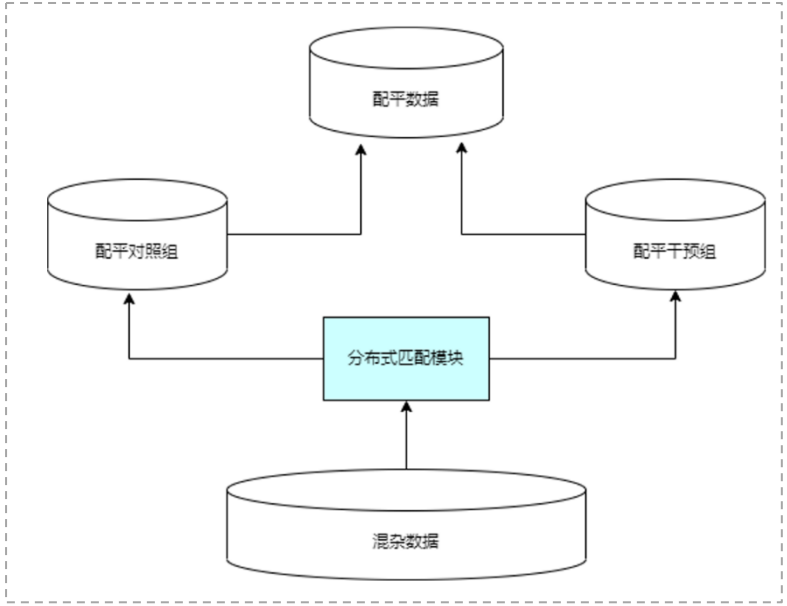
分布式低复杂度倾向性分数匹配（Hist-PSM）

如何将将因果推断做成大数据计算任务?

分布式计算架构



分布式匹配思想

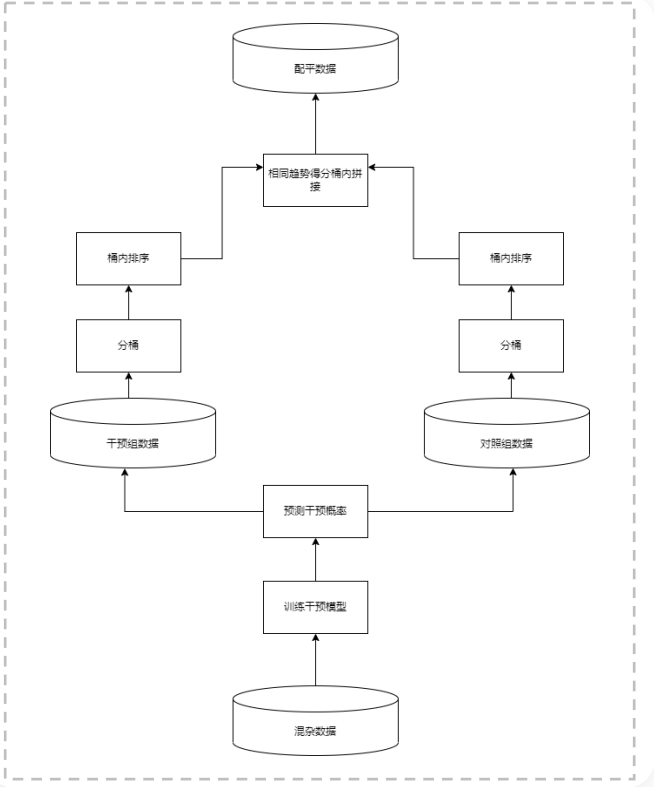


分布式计算+ matching

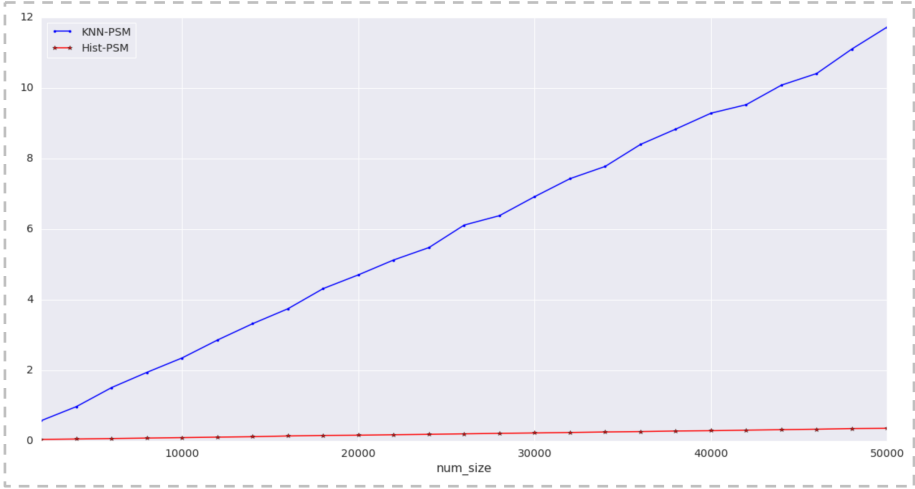
分布式低复杂度倾向性分数匹配（Hist-PSM）

如何将将因果推断做成大数据计算任务?

HistPSM的工程实现



HistPSM的工程实现



○ **内存占用更小：**KnnPSM 需要用 32 位的浮点数去存储特征值，并用 32 位的整形去存储索引，而 HistPSM 只需要用 8 位去存储直方图，相当于减少了 1/8

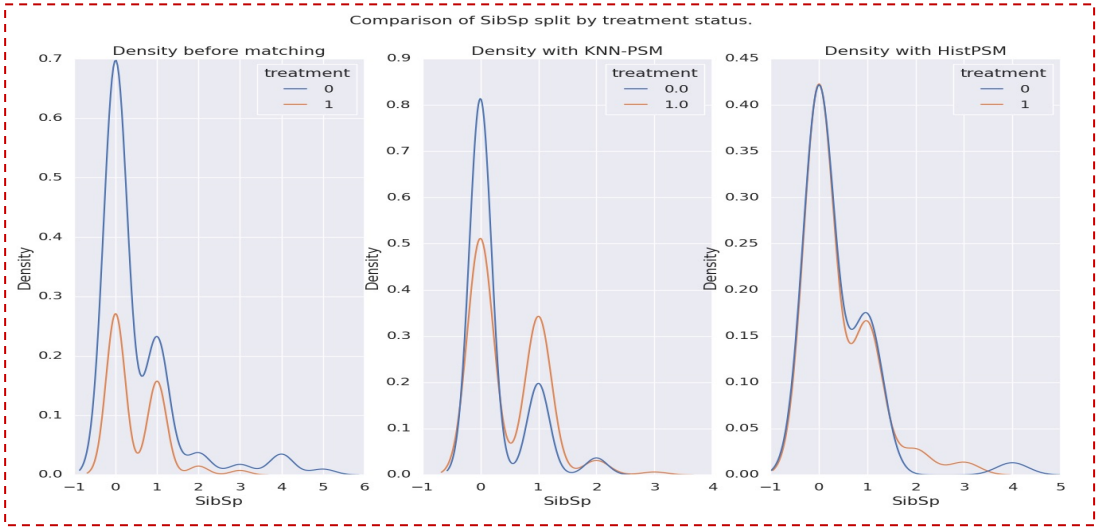
○ **计算代价更小：**计算特征分裂增益时，KnnPSM 需要遍历一次数据找到最佳分裂点，而 HistPSM 只需要遍历一次 k 次

分布式HistPSM for 大规模推断

如何将将因果推断做成大数据计算任务?

Attribute	t值 (匹配前)	p值 (匹配前)	t值 (KNN-PSM)	p值 (KNN-PSM)	t值 (Hist-PSM)	p值 (Hist-PSM)
Age	-6.263	0.000	0.153	0.879	0.441	0.660
SibSp	1.029	0.304	-3.273	0.001	-0.299	0.766
Parch	-0.850	0.396	-0.251	0.802	-0.263	0.793
Fare	-10.263	0.000	0.401	0.689	-0.635	0.527
propensity_score	-30.675	0.000	0.010	0.992	-0.150	0.881

SibSp属性分布



检验结论

- 通过平衡性检验，判断匹配后的样本pair是否相似。
- Hist-PSM可以通过平衡性检验，挑选出的“对照组”在各类混淆变量的分布与干预组近似。

03

分布式鲁棒双重稳健估计

分布式鲁棒双重稳健估计

如何将将因果推断做成大数据计算任务?

技术挑战：由于双重稳健估计的最初设计是针对连续结果问题（例如学生的分数、工人的收入等），使用倾向值得分的倒数进行加权，权重很大的时候方差也大。对于二元结果场景会存在以下问题：

- 双重稳健估计没有均一化（Uniformization）的过程，对于倾向值得分的倒数较大的场景，**会导致大量小于-1或者大于1的ATE出现.**

传统双重稳健估计

$$\widehat{Z_1} = \frac{\sum\left[\frac{T_i\left(Y_i - \hat{\mu}_1(X_i)\right)}{\hat{p}(X_i)} + \hat{\mu}_1(X_i)\right]}{N}$$

$$\widehat{Z_0}$$

$$= \frac{\sum\left[\frac{(1-T_i)\left(Y_i - \hat{\mu}_0(Xi)\right)}{1-\widehat{p(X_i)}} + \hat{\mu}_0(X_i)\right]}{N}$$

预估ATE: $\widehat{\Delta_{DR}} = \widehat{Z_1} - \widehat{Z_0}$

Binary双重稳健估计:将二元结果问题转化为连续回归问题，使用线性回归模型的预测值逼近分类任务真实标记的对数几率

$$\widehat{Z_{1B}} = \frac{\sum\left[T_i\left(\log\left(1-{}^{1}/_{Y_i}\right) - \log\left(1-{}^{1}/_{\hat{\mu}_1(Xi)}\right)\right)/\hat{p}(X_i) + \log\left(1-{}^{1}/_{\hat{\mu}_1(Xi)}\right)\right]}{N}$$

$$\widehat{Z_{0B}}$$

$$= \frac{\sum\left[(1-T_i)\left(\log\left(1-{}^{1}/_{Y_i}\right) - \log\left(1-{}^{1}/_{\hat{\mu}_0(Xi)}\right)\right)/(1-\widehat{p(X_i)}) + \log\left(1-{}^{1}/_{\hat{\mu}_0(Xi)}\right)\right]}{N}$$

预估ATE: $\widehat{\Delta_{DR,\,Binary}} = \frac{1}{1+e^{\widehat{Z_{1B}}}} - \frac{1}{1+e^{\widehat{Z_{0B}}}}$

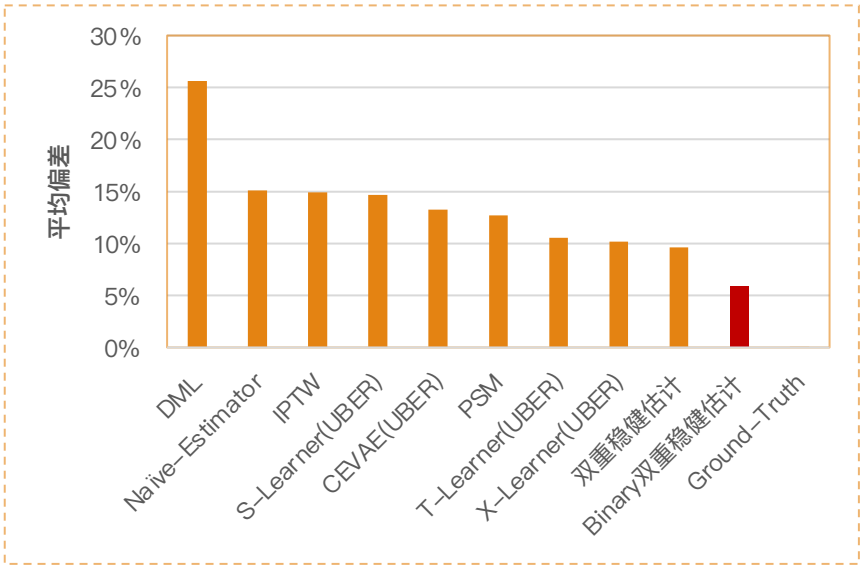
$\hat{\mu}_1(X_i)$是 $E[Y|X,T=1]$ 的使用逻辑回归估计, 而 $\hat{\mu}_0(X_i)$ 是 $E[Y|X,T=0]$的使用逻辑回归估计.

分布式鲁棒双重稳健估计

如何将将因果推断做成大数据计算任务?

在具有Hidden-Confounder的二元结果的环境下，我们进行了1万次ATE拟合仿真检验

- **相比UBER表现最好的算法UBER-X-Learner，Binary双重稳健估计将平均偏差降低了42.16%**
- **相比传统双重稳健估计，Binary双重稳健估计将平均偏差降低了38.54%**



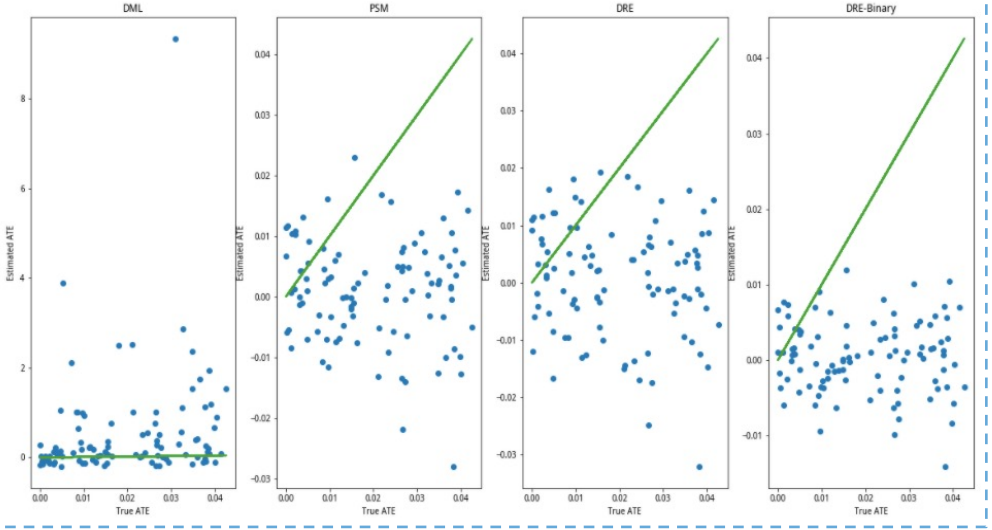
算法	ATE	平均偏差	95% C.I.
DML	0.913	25.7%	(0.907, 0.917)
Naïve-Estimator	0.836	15.1%	(0.824, 0.847)
IPTW	0.835	14.9%	(0, 1.852)
S-Learner(UBER)	0.833	14.7%	(0.821, 0.843)
CEVAE(UBER)	0.823	13.3%	(0.819, 0.831)
PSM	0.819	12.7%	(0.817, 0.819)
T-Learner(UBER)	0.803	10.5%	(0.793, 0.812)
X-Learner(UBER)	0.800	10.2%	(0.791, 0.809)
双重稳健估计	0.796	9.6%	(0.794, 0.797)
Binary双重稳健估计	0.769	5.9%	(0.767, 0.771)
Ground-Truth	0.726	0.0%	(0.724, 0.727)

仿真数据来源 Louizos et al. (2018) “Causal Effect Inference with Deep Latent-Variable Models，仅做参考

分布式鲁棒双重稳健估计

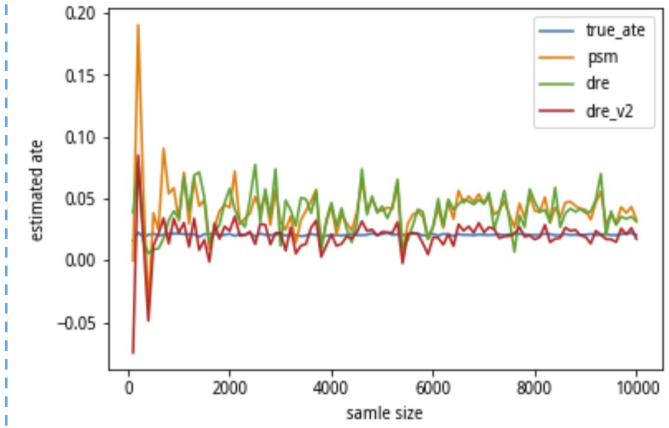
如何将将因果推断做成大数据计算任务?

安慰剂(Placebo)检验



安慰剂检验：在对输入干预随机化后，Binary双重稳健估计比PSM和DRE更加密集地分布在0附近（DML会存在大量ATE>1的点）

缩减样本量(Subset-Data)的仿真验证



缩减样本量仿真检验

- 趋势得分和结果估计都变得更加不准确
- 尤其是DML会严重偏离真实值
- Binary双重稳健依然表现出良好的ATE估计

仿真数据来源 Louizos et al. (2018) "Causal Effect Inference with Deep Latent-Variable Models，仅做参考

04

分布式面板双重差分

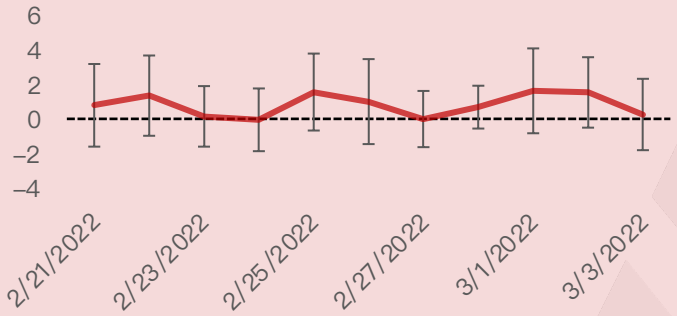
分布式面板双重差分

活动概况

痛点

- 玩家参与不同活动的次数、程度不同，受多种活动同时进行，难以对不同活动的效果进行区分
- 到活动的影响也不同

解决方案

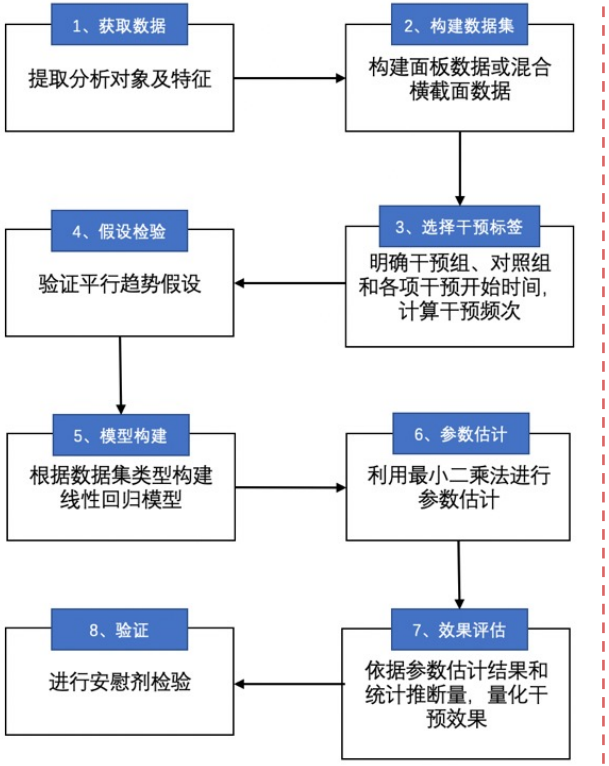


- 实验组和对照组玩家在活动开始前的人均在线时长符合平行趋势假设
- 可通过构造面板二重差分（Panel DID）模型对各个活动的效果进行归因

*示例数据（dummy data）仅做参考

分布式面板双重差分

整体分析流程



*示例数据（dummy data）仅做参考

1. 构造面板数据

玩家	时间	指标	活动1	活动2	活动3	用户特征
玩家1	2021-11-04	880	1	0	0	……
玩家1	2021-11-05	1220	1	1	0	……
玩家1	2021-11-06	0	0	0	0	……
		……				
玩家2	2021-11-04	2010	0	0	0	……
玩家2	2021-11-05	2400	1	1	1	……
玩家2	2021-11-06	1540	1	0	1	……
		……				
玩家3	2021-11-04	0	0	0	0	……
玩家3	2021-11-05	540	0	0	0	……
玩家3	2021-11-06	460	0	0	0	……

2. 构造面板二重差分模型

$$\log(Y_{it}) = \sum_{k=1}^{K} \theta_{1k} * Post_{itk} * Treatment_{ik} + \sum_{k=1}^{K} \theta_{2k} * Post_{it} * Treatment_i * Times_{it} + \sum_{j=1}^{J} X_{ijt} + \gamma_t + \epsilon_{i,t}$$

3. 参数估计，得出结果

活动名称	活动效果（ATE）
活动1	XX %
活动2	XX %
活动3	XX %

Tencent 腾讯 | DataFun.

总结与展望

总结

后续展望和规划

- 探索更多潜在的业务场景
- 优化在未知ground-truth环境下的检验
- 标准化分析方法论

Tencent 腾讯 | DataFun.

非常感谢您的观看