

作业：

一、留存

1.留存的含义：

留存：指基准日到APP的用户在之后的n日当天返回APP的人数；

留存率 = 基准日之后的n天当日返回的用户数 / 基准日的用户数 * 100%

或者指基准日产生某个行为的用户在之后的第一天，第二天，第三天.....第n天的当天再次产生该行为的人数。

2.留存的意义

留存代表一个用户愿意再次使用你的产品；而一个产品能够被用户再次使用，意味着这个产品是能够满足用户长期需求的，能够让用户产生一定粘性的产品。

如今互联网产品大多为免费产品，依靠持续的广告转化、用户持续的购买转化、用户持续的会员付费来维持收益。

那么，怎样才能达到持续的广告转化？持续的购买转化？持续的会员付费？

广告转化、购买、会员付费，我们都需要用户来完成。那么持续的广告转化、购买、会员付费，我们就需要每天都有一定量的用户来完成以上过程，也就是说我们需要我们的产品每天都有活跃用户的。



那么从活跃用户的构成，我们就知道怎么维持我们的活跃用户规模：一方面持续有新用户流入，另一方面持续有留存用户留存；

那么为什么留存用户重要？部分有过从业经验的同学应该知道，产品的自然新增用户是比较少的，如果我们仅靠产品自然用户增长，那么必然我们的活跃用户量级会非常小，虽然能够获得收益，但是这部分收益很可能养不起我们的公司；但是如果我们想要用购买新用户的方式获得持续的较大规模的用户池，又会产生比较大的成本，这样做容易亏本。所以我们需要新增留存两手抓；想方设法留住我们新增的用户，让其长期地在我们的产品内产生价值，这样做，比起单做新增，在成本和效果上都是要好的。

3.留存的实际应用

留存的主要应用场景有两个：

一个是产品整体视角的持续留存情况；也就是我们需要定期分析用户在产品的留存是否维持在一个正常的范围，有没有突然地涨跌情况，如果有的话问题出在哪里，为什么会出现这样的问题，如何解决留存异常的问题。遇到这类问题首先需要确定的是整体用户的留存异常还是个别群体用户的留存异常，如果是整体用户的留存异常，我们需要分析我们的产品是否发生异常，可以通过用户的行为路径确定异常产品位置；如果是个别用户的留存异常，需要通过用户拆解的方法来定位有异常的用户群体，然后通过这类用户的特征来分析问题原因；

一个是新增用户视角的每批新增用户留存情况。一般出现在计算投放ROI的场景，也就是我们投放一批用户，能够给我们带来多少收益。我们会用这些用户的“整个生命周期能够带来的收入”（LTV）去除以“投放时的支出”来计算我们的ROI。“整个生命周期能够带来的收入”计算的是引入一批新用户，当用户完全流失时，这部分用户所带来的所有收入。但是有时我们需要提前预估新用户在整个生命周期能够带来的收入，我们就需要首先用留存来估计用户的生命周期，然后用估计出的生命周期再去计算用户的整个生命周期能够带来的收入。

4.什么企业不关心留存

需要用户产生持续活跃，持续转化，持续消费的企业都关心用户的留存。那么，什么样的企业不关心用户留存？

部分线下企业存在数据难以获取的问题，所以对留存关注比较少，比如线下的商场，除非用户产生购买行为，否则很难探知用户在什么情况下有回流；

部分存在一锤子买卖的企业也不太关心留存，比如说用户在一次买断商品后再也不会产生后续购买或价值转化行为，这类企业也不怎么关注留存。比如PC单机游戏；比如一次付费的工具类产品（线上线下类工具都存在这个情况）；

问题：

1.计算某APP的每日留存数据。用SQL分解留存计算，每步请用视图建立（90分）

如何计算留存：

首先我们的目的是建立如下的留存表：“用户每日留存率统计表”

信息	结果 1	剖析	状态			
dates_a	device_v	day_1	day_2	day_3	day_7	
2019-11-18	226	65.04%	64.60%	60.18%	60.18%	
2019-11-19	222	69.37%	63.51%	61.71%	57.21%	
2019-11-20	231	63.20%	59.31%	61.90%	57.14%	
2019-11-21	232	62.07%	62.93%	65.09%	62.50%	
2019-11-22	226	69.47%	64.16%	65.49%	63.27%	
2019-11-23	241	66.80%	66.39%	58.51%	63.49%	
2019-11-24	236	69.07%	63.98%	66.53%	65.25%	
2019-11-25	243	64.61%	63.37%	66.26%	61.73%	
2019-11-26	221	67.87%	67.42%	65.16%	56.56%	
2019-11-27	232	70.69%	66.38%	65.52%	62.07%	
2019-11-28	240	67.92%	63.33%	64.17%	57.50%	
2019-11-29	243	69.55%	67.49%	63.79%	67.08%	
2019-11-30	244	63.52%	68.03%	64.75%	64.75%	
2019-12-01	245	66.12%	60.41%	66.12%	63.27%	
2019-12-02	241	67.22%	66.80%	67.22%	64.73%	
2019-12-03	235	69.79%	67.23%	67.23%	66.38%	
2019-12-04	247	68.02%	68.42%	65.99%	66.80%	
2019-12-05	242	68.18%	64.05%	65.70%	80.99%	
▶ 2019-12-06	250	65.20%	64.40%	64.40%	61.60%	
2019-12-07	241	69.29%	68.46%	69.71%	62.24%	
2019-12-08	247	69.23%	70.04%	73.28%	61.94%	
2019-12-09	251	68.92%	70.52%	80.48%	59.36%	

大家可以看到这个留存表由以下几个字段构成：

dates_a 计算留存的基准日日期。这个基准日可以为任意一天，如果基准日就是今天，也就是没有“之后的第一天”、“之后的第二天”……等等天的数据，那那几天的留存可以以0、或者空值代替；

device_v 基准日当日的活跃用户数。不同产品的活跃用户定义不同，在这里，我们将活跃用户定义为在当天有过打开APP的用户；

day_1 次日留存率，也就是“基准日之后的第一天”的用户的回访比例。

day_2 2日留存率，也就是“基准日之后的第二天”的用户的回访比例。

day_3 3日留存率，也就是“基准日之后的第三天”的用户的回访比例。

day_7 7日留存率，也就是“基准日之后的第七天”的用户的回访比例。

也就是说：这个表格，表达的是：每天对应多少活跃用户，每天的活跃用户对应的次留率、2日留率、3日留率、7日留率是多少。我们能从这个表格中可以看出我们产品的用户活跃程度，以及产品对当日活跃用户的短期、长期粘性情况。

而我们的原数据，长这样：

信息	结果 1	剖析	状态
user_id	dates		
98047837	2019-12-06		
97726136	2019-12-09		
98607707	2019-12-18		
98662432	2019-12-06		
98145908	2019-12-16		
93784494	2019-12-03		
94832743	2019-12-13		
95290487	2019-11-27		
96610296	2019-12-11		
100684618	2019-12-05		
100509623	2019-12-08		
101781721	2019-12-13		
103891828	2019-12-01		
101260672	2019-12-12		

是一个每日app活跃用户表。其字段分为两列：

user_id 即当日活跃的用户id

dates 即用户活跃日期

那我们如何能够根据这个“每日活跃表”，做出上面那个“用户每日留存率统计表”呢？

关键的两个步骤：

1. 我们需要将每日活跃的用户数据与之后几日该用户的活跃数据连接起来；这样，我们就能知道这个用户在之后的哪一天有回访行为；
2. 通过SQL的数据列联功能，也就是“聚合函数(case when 条件)”的方式，将满足“基准日之后的第一天”、“基准日之后的第二天”.....的用户数据统计出来，最后再通过基准日的用户数，计算用户留存率；

之后的作业题，就是引导学员们如何将以上两个过程实现。

1.0.建立用户活跃日期表（5分）

表名：temp_user_act

字段：

字段名 字段类型 字段说明

user_id int 用户id

dates date 活跃日期

1.1.用用户活跃日期表做自连接，连接方式使用左连接，连接字段使用“用户id”字段，保留两表的用户id与两表的日期（20分）

1.2.筛选出右表日期大于或等于左表日期的内容（20分）

1.3.计算以左表日期为基准日的当日用户数，第二日回访用户数，第三日回访用户数，第四日回访用户数，第八日回访用户数；（20分）

提示

提示：

- datediff(B,A)=1 表示日期B为日期A之后的一天。
- count(distinct case when datediff(B,A)=1 then uid else null end) 或者 count(distinct if(datediff(B,A)=1,uid,null)) 就是求出“日期B为基准日A之后的第一天”的用户数计数

1.4.利用上述数据计算出每日的当日用户数以及次日留存率，二日留存率，三日留存率，7日留存率(率需要使用百分比表示结果)；（20分）（在下一部分的案例中，我们会给大家介绍一个一次性完成的留存计算方法，给大家加深留存计算的印象。）

1.5（拔高题）*.求出每日的次留与次留的周环比；（5分）

注：

周环比：周环比适用于以周为周期变动的数据，用于观察每天的数据较上周同一天的数据变化。

现有互联网数据多以周为周期变动，一周内每一天的数据都有其特点：

如工作类APP，周一会是每一周的高峰，然后向周五逐渐递减，周五到周六又会有一个锐减的过程，周日与周六趋于平缓；

如娱乐类APP，周一会较周末锐减，然后向周四逐渐递减，周五会有一个跃升的过程，周六会再次跃升，周日会较周六下降；

一般遇到这样的变动趋势，如果单纯的分析每一天较上一天的变动情况，就容易分析出错误结论。所以，在我们分析带有周期类数据时，一定要先将周期提炼出来，使用周期的环比数据做分析，会得出更有价值，更加准确的结论。

1.计算方式：

假如今日为周一，那么周环比=(本周一数据 - 上周一数据) / 上周一数据 * 100%

2.没有上周对应日期的数据的部分可以为空；

二（拔高题）*.作者活跃行为统计（10分）

如今的大多数内容APP都是以平台模式运营的。

什么叫做平台模式？

比如我们现在能接触到的大多数内容产品“抖音”、“Bilibili”、“微博”、“今日头条”等等。我们使用到的这个APP，是由APP的制作公司提供的，也就是这个APP会有几个主要的页面，每个页面怎么引导用户进入内容页面，播放内容的方式有哪些、内容间切换会用什么方式等等，都是我们的APP制作公司决定的；而我们在这个APP里看到的大多数内容，都是由内容作者提供的。

为了能够让我们的平台能够有持续的内容产出，我们就需要分析作者的活跃行为。主要为我们的作者运营提供有力帮助。一方面让我们找出活跃作者，增加政策扶持；一方面让我们找到快要流失的作者，及时召回这部分作者、挽回损失。

某短视频公司有作者发布视频统计表如下：

表名：temp_author_act

字段:

字段名 字段类型 字段说明

dates date 发布日期

author_id varchar(5) 作者id

2.1.请求用SQL出作者的最近三个月内的最大断更天数、平均断更天数和最大持续更新天数; (5分)

名词解释:

持续更新天数: 如果一个作者在某几天中每一天都有更新, 那么这段时间的天数称为这个作者的持续更新天数。

断更天数: 如果一个作者两次更新中间隔了几天没有更新, 那么这几天的天数称为这个作者的断更天数。

2.2.运营人员需要对作者做电话访问, 需要你用SQL求出每位作者在最大断更天数时对应的日期范围。用于访问该日期内的断更原因。 (5分)