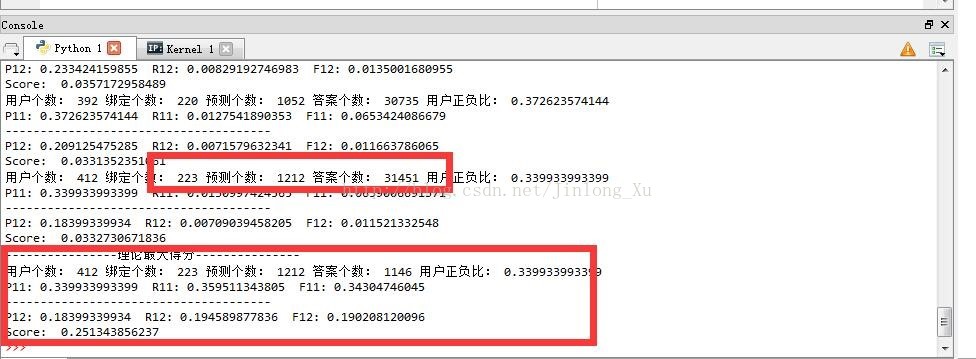
**申明**：本文由浪叫兽对京东JData算法大赛的总结，行文略微口语化，但是不得不说，真正琢磨过数据的人才能发现更多内在的东西。Mark，学习一下。

0.055规则很简单的，就是type5 大于 2，就这一条就可以了。我们搞的是kdd，主要是知识发现，所以大多是重复 eda（探索分析） etl（数据清洗），数据清洗和探索，使用现有的数据挖掘框架。重头戏还是在eda 和etl，我们只是使用框架，重复 eda etl。

你没发现老王经常说的一句话吗，为啥比你分高，说明数据里面还有规律你没找到，你坚信你不比别人笨，就探索数据，就一点能和第一拉进距离，我觉得吧老王经常重复这句话对我启发也很大，我们都是在一个挖掘框架体系下堆积代码而已，本身就是没有跳出发现阶段，既然是发现，越高的人不过是多了份运气 ，比你早发现规律，但是时间上你努力应该是要追平差距，重复性的去对数据和结果进行eda，不过比赛很有套路的是要看评分标准。

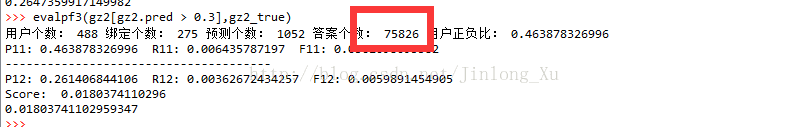
在给大家一个全局性的京东赛方案，看你们要玩规则还是模型，主要京东赛 你要设计好评分公式 ，我们都知道京东赛要滑动，带来最大的困扰是 正负样本的问题。规则走的是细挖，模型走的是多次过滤，控制阀值，设计好整体的框架在探索特征，加入第一层和第二层分别看效果，这是模型的路子。



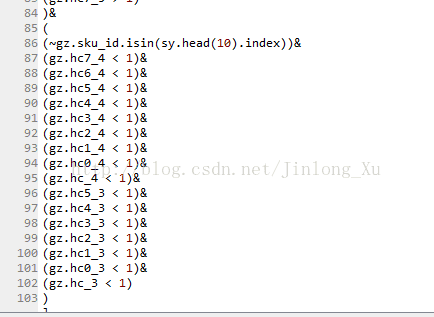
第一个是看第一层的模型输出的个数 能不能解决正负比，在看后面的最高分设定阈值，就是说我理论上都对了，能达到什么分

evalpf3(gz[(gz.user\_id.isin(gz\_true.user\_id))&(gz.pred > 0.5)],gz\_true)。

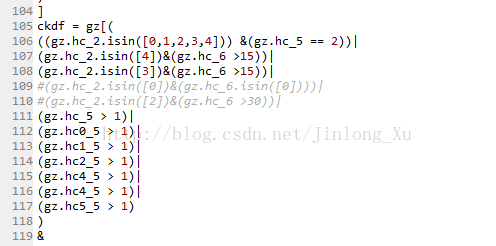
评分给上，如果都对我能多少分，以预测概率为大于 0.5的阀值为准，那么我降低要求，我只要第一层分出的，最高分越高，我就考虑降低阀值，你第二层的理论最高分就高上去了。



但你要考虑第一层输出的答案个数，这个比你直接堆积获得的正负比是不在一个档次，就这个思想就决定了你要高别人一筹了，当然依次可以设计 第二层，第三层 ，这就是模型的建模思路了。在说规则把，你们都说规则不可以解释，但是如果大家都用的同一个测试集，一样的优化思路，和模型也是相当的威力，我给大家举一个例子。

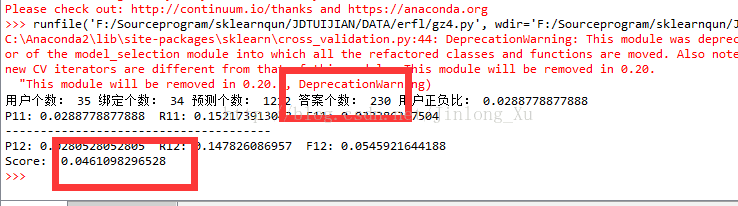


         基础规则大家都知道，加入了购物车的 不会重复购买，这些我们自然给打上。

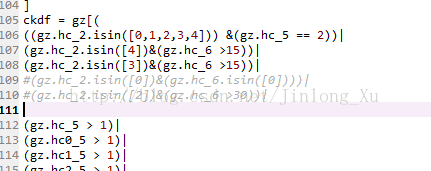


这里我们就需要控制输出个数了 ，首先我用我的规则给你们演示，看看我这个输出的答案是多少，不过规则我把

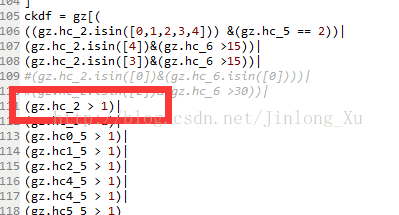
前8天的都卷进来了，有点耗内存。



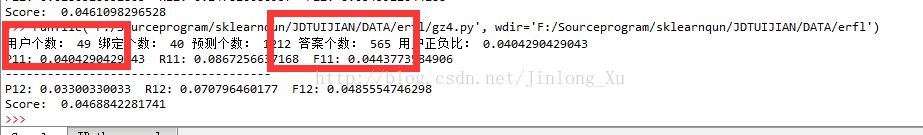
看见没有，规则 只要输出 230 个就线下 0.046 了 ，这个规则我提交过，说明这条规则就放出几百个答案。



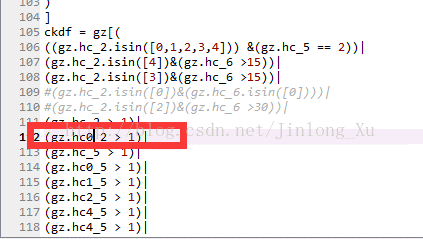
我们加上我们要分析的大头，就是  都加入过购物车的。



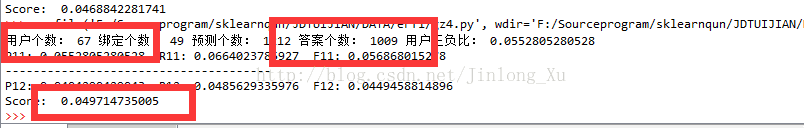
先开一个区间，最后一天加入购物车的。



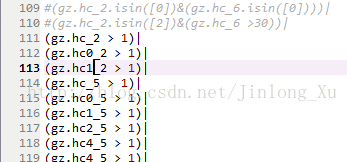
看见没有输出的个数又多了几百个，分也上去了，用户个数也多了 ，这个正负比我们还是能接受的 ，10 个里面 有一个 是对的人 。



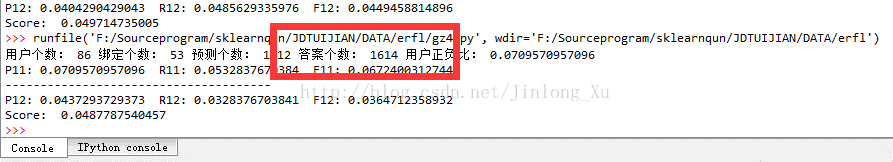
我们开一个区间，就是前面两天加入了购物车的都提交，看看线下得分：



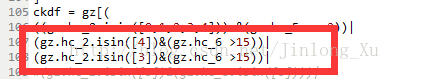
       答案个数为 1000 个，对的user\_id，67 个，评分又多了 ，所以你看看京东赛的聊天记录  有人会说加了两天的购物车 的规则会更好，并不是线下测试不出来的，这个正负样本比我们还能接受 ，继续在放一个区间：



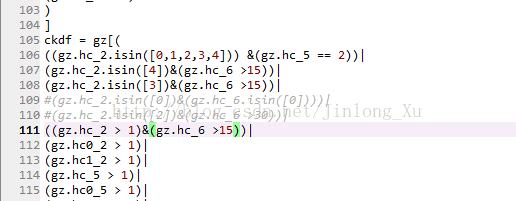
其实我们这里总体来说是一条规则 ，就是加入了购物车的 ，这个是整体的基础规则：



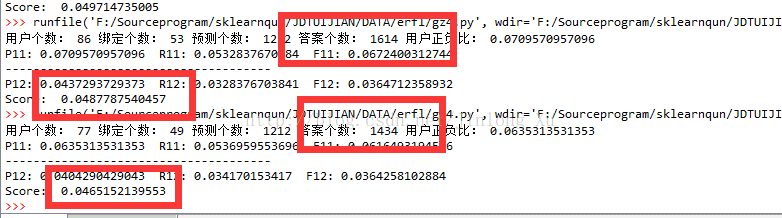
答案个数 1600 多个，对了80 多个用户，这里要是你想夺 0.1，就考虑对这个1600 个用户继续加细分规则：



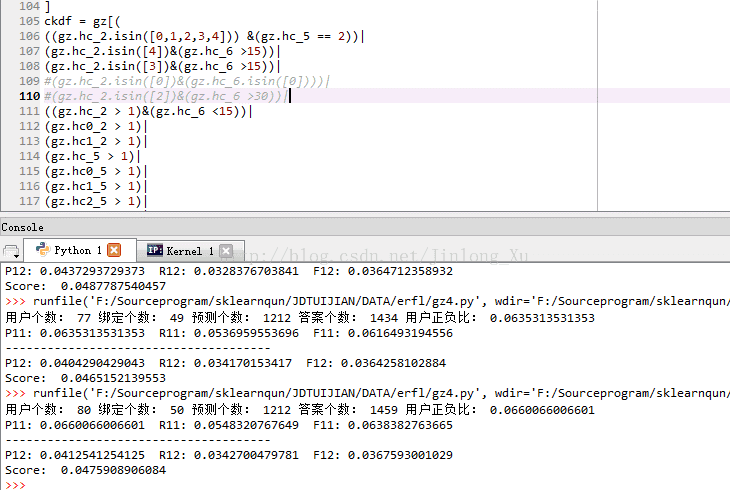
类似这种



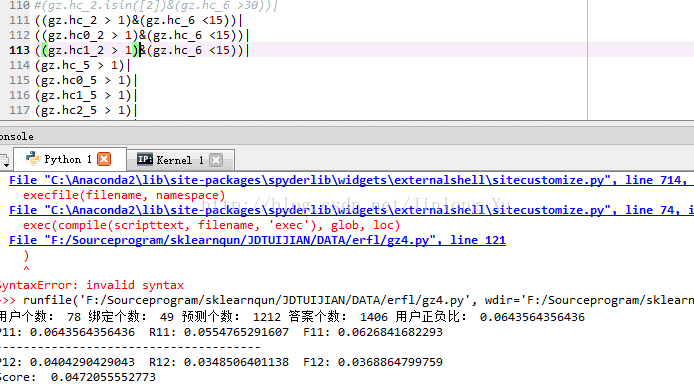
我给第一天的加上限制规则，运行：

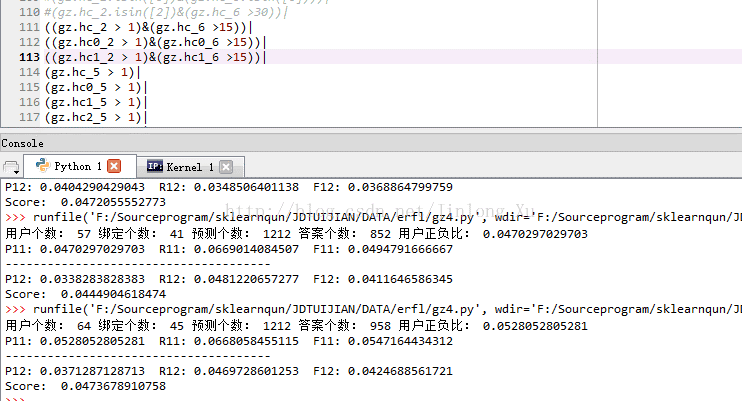


     发现答案少了200 多个，线下分也低了，那么很简单的思，我把大于 改为小于，这逻辑大家都知道吧。

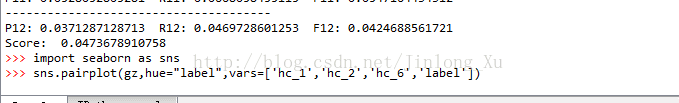


      不过有点打脸的是这个区间好像在前面的区间包含了，所以应该 三天的都加上：

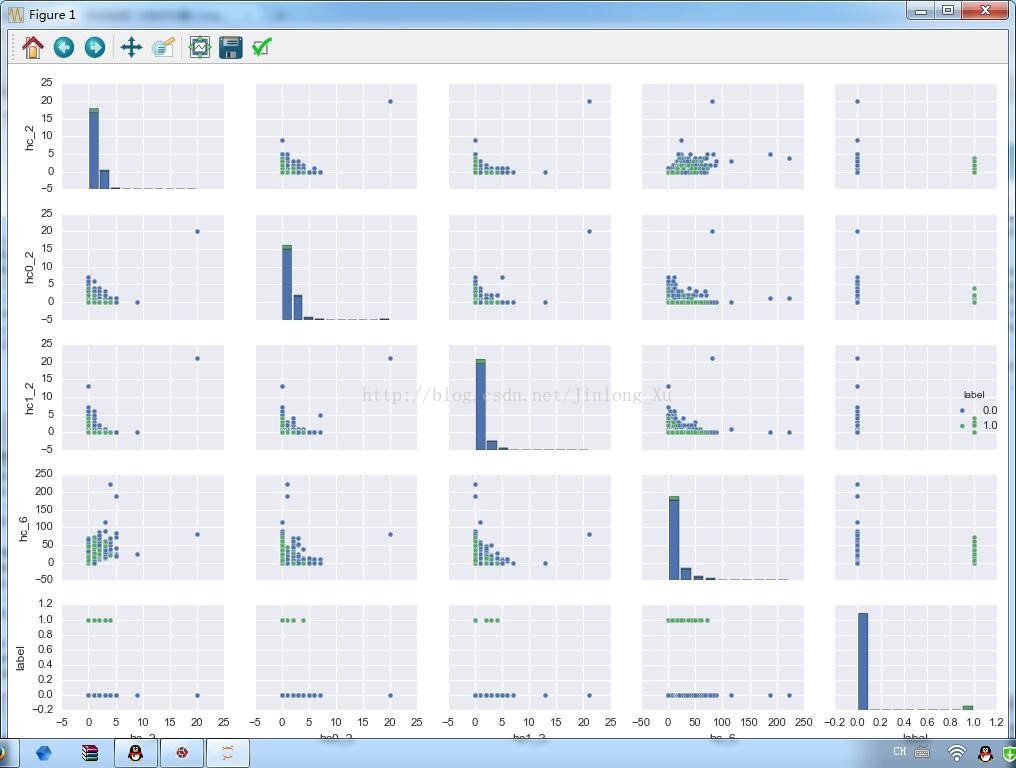




      反复实验只能证明这个区间可以提高比例：



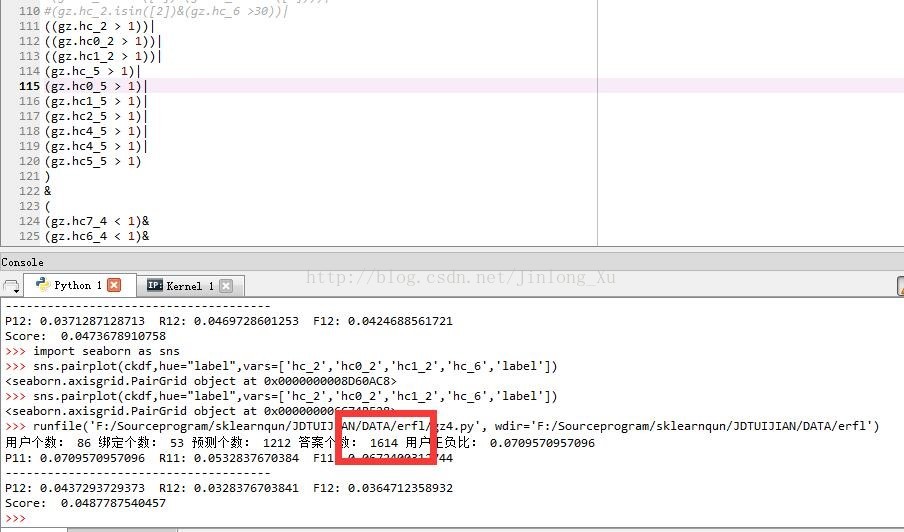
所以上肉眼看看：



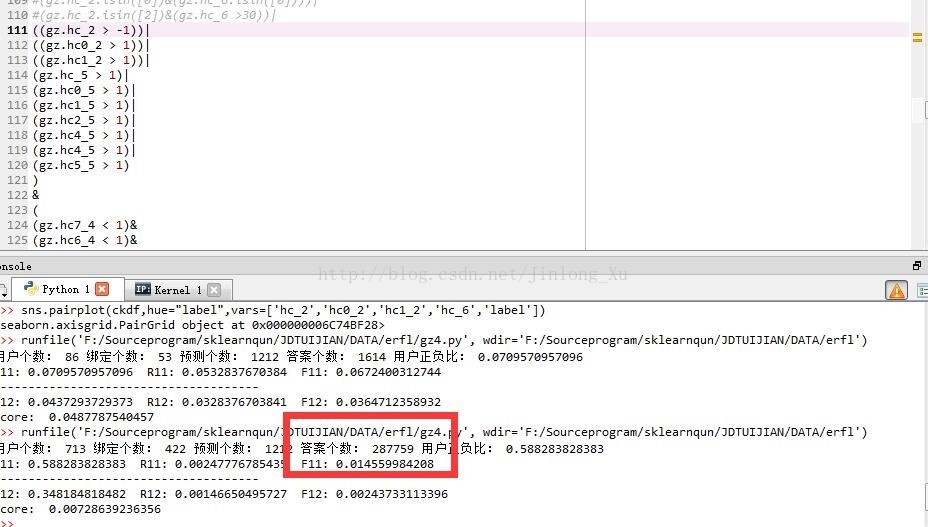
        这样就看看取什么值合理了，反正整体的思路就是开大区间之后做细分 ，控制输出的个数的准确率，如果有什么好规则能删除一些 user\_id 和 sku\_id ，细分下去的困难就越小，模型是我输出10000 个都行，我在用第二层优化，规则是  我宁愿只输出 200个，但是我要保证正确率。特征决定模型的上线，对规则和模型算法来说是一样的。

        规则对分布和概率，要有深刻的认识，真心能玩到老王那样的程度了 ，这种探索是家常便饭，有时候我都搞不清到底是规则让人弱智还是模型让人弱智 ，还是都让人玩久了变弱智，概率分布召回，准确率，其实大家都是在玩这几个玩意而已，并没什么高尚的，耐心细心 加上很深刻的知道我们不过是在一个框架里面发现东西而已，人家说 cv就是cv ，stacking 就是 stacking，他就是一个框架里面的工具。

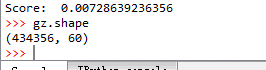
通过看评分，看你放了多少个样本进来：



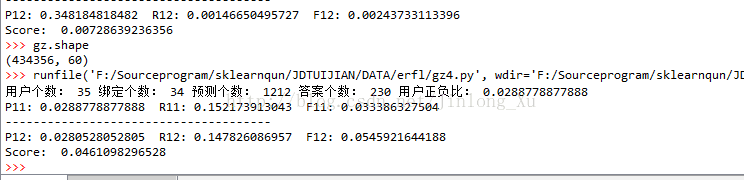
就这个规则来说，你就放了  1600 多个，对了  86 个，你自己算算，把这个规则过滤后放入模型训练：



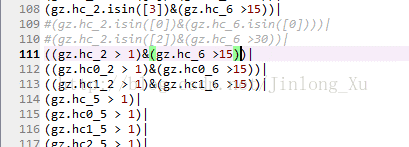
和你放出这么多个，对了 713 个，怎么样能让你走好下一步，你自己权衡：



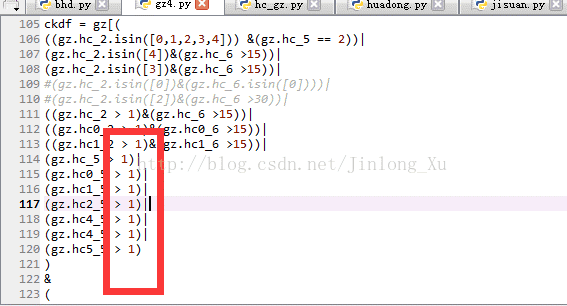
总比你放入全集   样本严重失去平衡来的好把，这些是没人会告诉你们的 ，反过来：



       我就用规则的这几百个，加入到 我最后的答案里面 ，本来 最后一天大家 一个是  0.210 ,一个是 0.209 ，我前期故意保留这个规则得分，最后一天绝杀0.211 ，这不就是规则党喜欢干的么，计算规则的得分具体能增加多少，以达到绝杀的目的，或是 你 0.20 我 0.21 ，我看你还努力，我直接 加到 0.22 了，你觉得提升 0.001 都难的情况下，我干到了 一次提高  0.22 ，士气上是不是压倒趋势，还有前期我用规则把样本搞到了50:1 ，而你 一直是100:1 的模型，我随便训练就秒杀你啊，这些东西你们在外面是听不到的  ，而且这个也是我自己一直在思考规则和模型的区别做的一个小总结，至于大家玩比赛喜欢怎么玩，喜欢什么战略，很难摸清楚的。就规则来说：



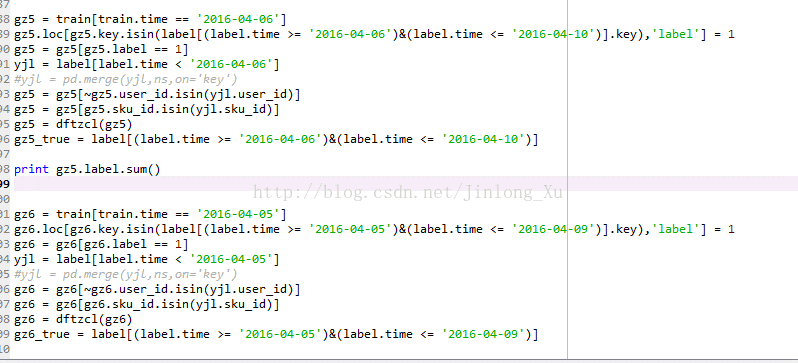
        我加了这个细分，我线下是降低了，但是我完全可以提交一下，说不好 线上就是涨的呢，那我就保留这个区间的值呗，或者你完全可以继续线下干到 0.2 ，这里就存在先验试探了，你本身不知道这个规则是会让线上上升的，但是你通过测试，保留了这个区间的取值，而你线下是降低的，这不就是耍流氓吗，而且随着你的规则到了0.2，你这个试探性区间就毫无影响了，规则和模型都可以干这种事情。  
        以信心知道  你不比第一名差，拉近距离到前十，剩下的很多就看看算了，除非是换数据后的第一次得分，其实换数据也不一定知道有没有用了技巧，最终可能获胜在一个小知识点上，全局上说不好前 20 都可能是一个解决方案，是大局上的，只是细致的内容不一样而已，我们做的工作是发现，本身就不是什么高大上的玩意。套路上的东西 一点就会了，实在不会看看博客也行了 。  
        这么多规则一点一点堆起来效果是有了，线上会不会有点乱？不要细分过厉害啊，硬是扣那一个值是 11  还是 12，实在有疑惑，你就提交呗，线上会给你反馈的，这很舒服的事情啊，反正不要细分太严重了 ，规则一样有过拟合啊，而且值 一样可以用算法调优的。



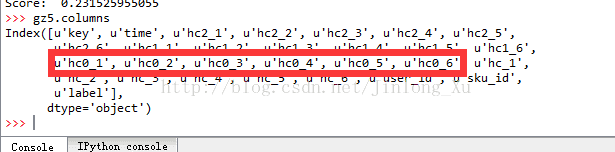
       比方说我决定了，我就用这些特征了，那我写一个网格，for，我给 一个 步长，是  1 2 3 4 5 6 7 8 9 10 ，还是  2 4 6 7 8 10，我线下暴力一下，不就可以了，细分的 1 2 3 4 5 6 7 8 9 10 ，我发现这个长度过低，线下好，线上差了，我试试 下一个步长呗。我只用四月份数据，规则和机器学习一样，都可以优化寻参，没你们想的那么无脑了，不同的比赛 要设计不同的规则 ，还有优化方向和有效特征，特征个数过多，也是头疼的，说不好规则认为预测5 天的 label置信度不高，我就预测三天的，那就用最后三天建规则啊，这也是个活的东西，我发现五天线下 0.1，线上 0.6，三天线下0.7，线上 0.55 ，我明显会去用三天建规则啊。谁傻用五天啊，所以这个也是个拟合成分，看看参赛者是什么心态了，我们玩数据讲究的就是一个理，怎么分高怎么干，至于道德，很多人只认钱的，就我玩比赛一年了，这些东西东西从来没人总结过，都是我自己琢磨出来的，你们听过类似的比赛技巧 ？？数据比赛圈很多不道德的行为的，且行且看。  
        有时候你就是到了 20 名 也不要觉的 第一名的方案比你好，要知道自己是学习的就可以了，其实前三可能就有一个菜鸟，运气好，干到前面去了，比方说我，走一次运就上去了哈，本身也没比你们厉害多少。   
        这种合理对于挖掘来说就是正当的思考过程，反正以后找到什么漏子，不要喷我就行了，我个人学识尚浅，有的东西不一定说的对，其实也不要误解规则党了，规则党和模型党本来就是一起了，都是探索而已：

滑窗

       滑窗的话，你看看隐马尔可夫的那个，公式怎么简化的，就一下明白了，这样滑窗的方法还是很多的。



      首先你假设 后面一天的数据只和窗口有关系，用五天来说就是你可以把 5天的数据堆在一起，1 - 5 天的数据变成 5 列，后面的一天的结果拼接给堆积好的那天就行了。一般你只滑一次会有问题，那么你就要叠加在往前面的一天的，这样样本就是卷起来一样，按一天天的叠加上去了，不过这个都是我的理解，至于是不是这样解释我也不清楚。



       比方说一天有 6 个特征，我滑五天就有 5 \* 6 列，卷积越多，内存足够大，效果越好。这个是我理解的 一个卷的行为，至于是不是  理论上的卷积，我没看过理论书。滑窗之后由之前的 六列特征变成30，在一天天的堆积上去，这训练样本就很明显的有一个滑 一个卷的过程。

http://img.blog.csdn.net/20170403180951347?watermark/2/text/aHR0cDovL2Jsb2cuY3Nkbi5uZXQvSmlubG9uZ19YdQ==/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70/gravity/Center