整体流程：

1. 自我介绍：

* 简单介绍学校；毕业；实习的时间点
* 介绍第一份工作的主要工作内容；

宜信：

主要做一些特征工程 ；数据梳理；数据清洗

数据梳理；（咨询 ； 进件 ；审核 ；贷后（车房贷 与信贷 ））主要对以上几个系统进行梳理及表之间的相关性关系；并对字段进行可用性评估（覆盖率；准确率；时间滑动及获取难度等）

以上数据为一级数据库 不会对数据进行数据删除 防止数据信息有损失

为模型做准备：

数据清洗：异常值清洗（年龄过大 或者过小 ；性别中有汉字 ；学历婚姻状况中有根据时间的推进含义不一致情况）等 ；空置清洗：根据均值（或者 -999） 填充 ；

后面主要做了一些风控模型（申请风险模型 和收益风险模型）还做过一些小模型（基于规则的贷后评分 和高精端客户转化模型）

* 介绍第二份工作 搜狗

识别pc网页库中的无线版式页面，在搜索时屏蔽。

了解html 和css 在无线 pc 页面展现上面的差异，主要通过 url ； html ；渲染；content ；版式 几个角度构建特征集合 应用树模型进行特征选择。构建准确率（100%）的响应性规则

1. 介绍在两个公司印象笔记深刻的项目：

宜信：申请风险模型 和收益风险模型

申请风险评分模型：

* 数据范围：主要根据城市信贷为基础：精英贷、新薪贷、新薪宜楼贷、精英贷（银行合作）、新薪贷（银行合作）
* 时间范围：20141001～20160331   由于之后的跟跑选择16年之后的数据 ，要大约有一年左右的表现期。
* 标签：

主要根据系统里面的是否放款：好人与拒贷（拒贷中的拒贷码进行好坏人的定义）

拒贷 并不是全部定义为坏人 ：主要根据拒贷原因进行统计，拒贷原因中主要是逾期相关及无还款能力低等作为原则上的坏人，有一些欺诈用户不在标签其中

因为 欺诈用户的信息模型是学习不出来的。

|  |  |
| --- | --- |
| 坏人 | 163728.0 |
| 不确定为坏人 | 484158.0 |
| 好人 | 647234.0 |

* 特征选择：

首先根据覆盖率 distinct值进行剔除覆盖率较低且

特征平稳——训练集和测试集的好坏趋势一致：主要画出 训练集测试集的密度图：查看相关差异性

有区分性——好坏趋势有规律，有差异 主要根据WOE进行划分

模型选择：要求 有可解释性 虽然用GBDT xgboost 跑过，但是最后展示出来的结果要用LR 解释性强

针对LR 做的特征工程：

连续址：

离散化、分箱  WOE 离散化

woe反映的是在自变量每个分组下违约用户对正常用户占比和总体中违约用户对正常用户占比之间的差异；从而可以直观的认为woe蕴含了自变量取值对于目标变量（违约概率）的影响

主要根据好人比例图进行划分bin值：横坐标 为此特征排序的占比，纵坐标为好人占比，

在分bin时候，主要保证每个小bin数据量不小于 2000  通话好人比例图，可以找到连续值的好人占比较高的特征区间即可以找到分割点

离散值：one hot encoding

运用LR 模型：L1 正则 主要调节lamda

收益模型：标签：有净收益最小期数问题

总收入(all\_in)=sum（0-i）（每期实还本金+每期实还利息+每期服务费）（求第 i期 之前所有实还总和   每一期都会发生变化）

总支出(all\_out)=本金+资金成本-前期服务费     (固定)

净收益(net\_revenue)=总收入-总支出。

结果评价：

P R F1

Roc ks FindbestThreshold

识别pc网页库中的无线版式页面，在搜索时屏蔽：

特征构建：

主要从三个方面进行特征抽取

URL ； HTML ；渲染差异

主要了解HTML PC端和无线端差异：

主要是html 的head标签； body的a标签 ；页面板式（渲染之后的元素宽度高度等信息）；通栏信息

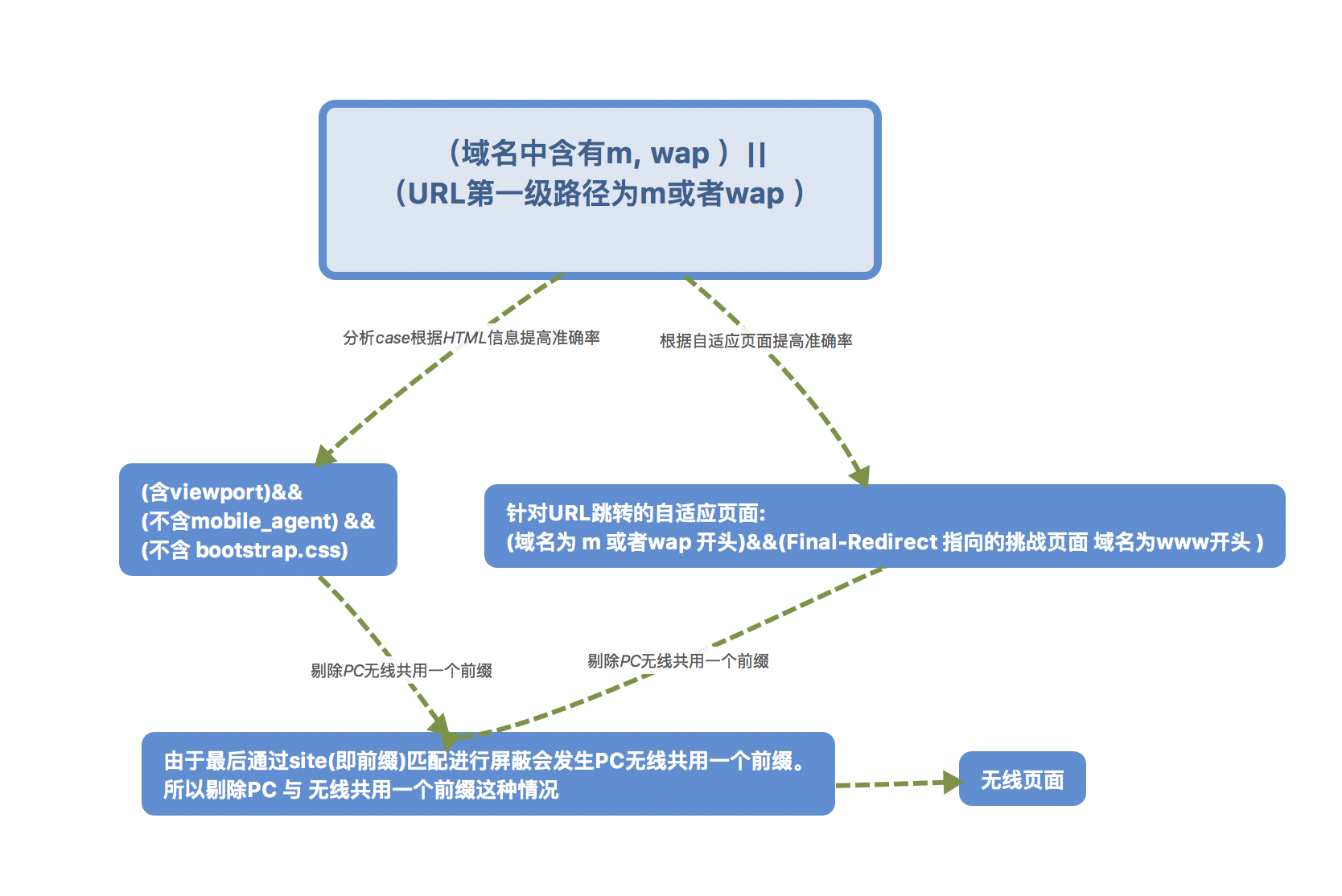
|  |  |
| --- | --- |
| title\_feature | Title 中含有3g版；手机版；触屏版；行動版 等 |
| maxWidth | 最大宽度 |
| getbannercnt | 通栏个数 |
| getPercent | 通栏占比 |
| isBanner | 是否是通栏页面 |
| viewportBaseCount | 包含viewport 的个数 |
| apple\_touch | 包含apple-touch-fullscreen |
| apple\_mobile | 在iOS中有两个meta值，apple-mobile-web-app-capable和apple-mobile-web-app-status-bar-style，这两个会让网页内容以应用程序风格显示，并使状态栏透明。包含 apple-mobile-web-app 标签. value=‘yes’ |
| mobileagentcount | 包含 mobile-agent的个数 |
| userscalablevalue | 放缩大小 |
| handheldfriendly | handheldfriendly. value=true |
| mobileoptimized | MobileOptimized |
| href\_skip | 1 ！only pc and notstartpc 2: only\_wap 4:：都不为null ，only start with pc  3:：都不为null ，only start with wap 5:：都不为null ，都匹配 wap长度长 other 6 |
| isadaption | .\*(width\\s\*=\\s\*\\d{3,4}).\* |
| maximum\_scale |  |
| URL\_has\_mwap | 域名中包含 m，wap 标签 |
| URL\_has\_3g,4g,mobi | 域名中包含 3g 4g mobil 关键字 |
| First\_path\_mwao | 第一级目录中包含map |
| Has\_**viewport** | **含<meta name="****viewport" value="...">标签的页** |
| Has\_ apple-mobile-web-app | 含 name="apple-mobile-web-app" value="...">标签 |
| Not\_has\_ mobile-agent | 不含<meta http-equiv="mobile-agent" value="... |
|  |  |

构建40维度左右的特征集合

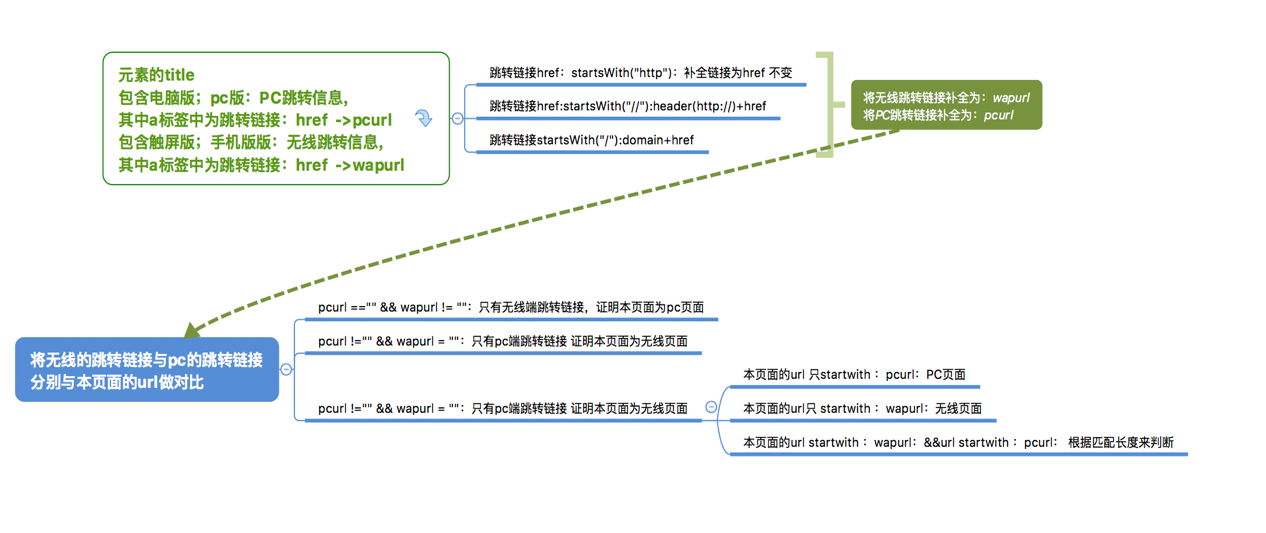
构建出一批纯度较高的wap pc url 各抽取2万数据集合 将数据根据以上model 跑出base line 准确率 88% 但是上线要求 ：准确率 100% 由于做的是基础数据库的删选，所有不能黑名单 不能加载 所以根据 gbdt 二叉树结构进行特征构建。

主要构建以下规则：

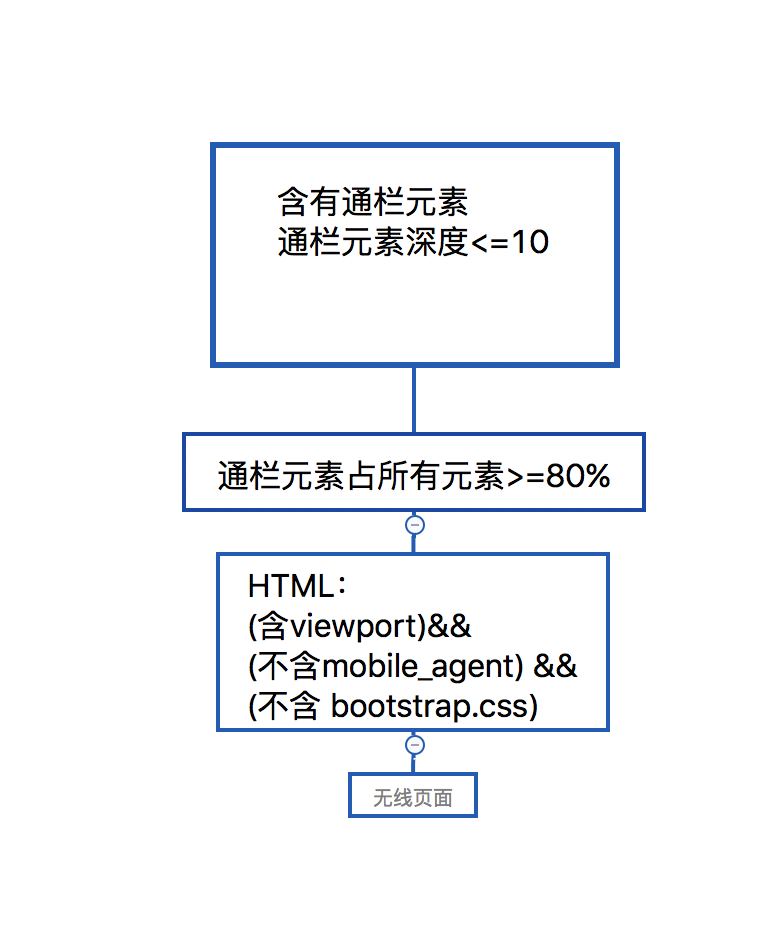
1:



2:



3:



分析case 优化规则

sbs 实验：

|  |  |
| --- | --- |
| 标注数据来源 | baidu、sogou wap日志7天数据（2018.03.06 – 2018.03.12） |
| 抽样方法 | 随机抽取 |
| 数据量 | 共60000条数据（其中baidu 30000、sogou 30000）diff查询967条，影响面1.61%，百度覆盖面：443/30000 =1.47%， 搜狗覆盖面：524/30000 =1.74% |



1. 算法推导：

* LR：本身是回归问题；加入映射来转化为分类问题 将线性不可分的问题转化为线性可分问题
* 这个映射为sigmoid 函数：sigmoid 的由来（贝叶斯 ；伯努利 角度可以推导 ok）
* 预测函数已有；此时求解参数 目的 最小化经验风险 防止过拟合 加入正则，转化为结构风险，（最大对数似然 求导 ok 求得迭代结构 sum（h（xℹ） -yi）／M ok）
* L1 L2 正则 降低模型复杂度 原理：ok

<https://liam0205.me/2017/03/30/L1-and-L2-regularizer/>

L1正则化不可导，怎么求解？坐标轴下降法（按照每个坐标轴一个个使其收敛）

* 牛顿法 梯度下降法 坐标轴下降法（半ok 状态）

https://blog.csdn.net/u014688145/article/details/53688585

<https://blog.csdn.net/u014688145/article/details/53688585>

（。。。。。。。最大熵模型 还没太看懂）

* ROC KS +F1 理论及含义 ok
* Xgboost +gbdt：差异 + 推导 基本ok

1. 数据结构 算法推导

快排

堆排 ；桶排序

编辑距离

随机采样