

# 大数据分析的一条入门途径

## ——以拍拍贷风控模型预测为例

范方达

Kesci “魔镜杯” 风控算法大赛 涌泉队

2016.4

# 出发点

- 大数据是更多人可以理解的
- 大数据的方法也是更多人可以理解的
- 大数据没有祖传秘方——不要把曾经初学的我们拦在外面
- 这并不是唯一一个正确答案，而是恩典在面对每一个小小的困难中的累积

# 目的

- 为数据分析初学者提供一点点数据分析的思路
- 为Python初学者提供一点点Python处理数据的技巧
- 为机器学习过程遇到的难题提供一点点解决方案

# 大纲

预备

数据读取

数据摘要与清洗

模型选择

模型训练与评估

模型组合与预测

回顾

# 大纲

## 预备

## 数据读取

## 数据摘要与清洗

## 模型选择

## 模型训练与评估

## 模型组合与预测

## 回顾

# 数据与目标

## “魔镜杯”风控算法大赛复赛数据

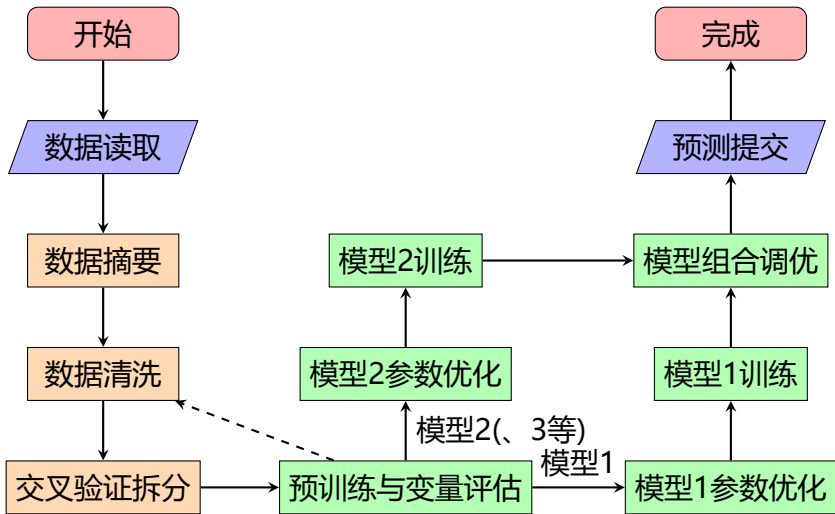
- 样本
  - 训练样本：初赛训练集+初赛预测集+复赛训练集（8万）
  - 预测样本：复赛预测集（1万）
- 自变量
  - 主表（226个）
  - 登录信息（4个，但每个index有多条）
  - 用户更新信息（3个，但每个index有多条）
- 预测变量Y: 每个index的6个月内贷款逾期情况（0-1）
- 优化目标：预测变量Y在预测样本的AUC得分

# 代码平台

## Python 3.5

- Packages :
  - 代码笔记本 : jupyter
  - 基础: numpy, scipy, pandas, matplotlib, time, re
  - 模型: sklearn, xgboost, keras (theano), hyperopt
- Windows下建议Anaconda , 包含科学计算的众多常用包

# 流程预览





# 大纲

预备

数据读取

数据摘要与清洗

模型选择

模型训练与评估

模型组合与预测

回顾

# 数据读取

- Q: 数据集有一些不同的文件，怎样合成一个数据呢？
- A: 首先我们可以根据数据类型为它们重命名来分门别类
  - 项目名(PPD)可以做前缀，区分项目时一目了然
  - 主表(da)、历史记录(dah)、辅助(daa)、初赛预测列(day)
  - 训练集(t)、预测集(v)
  - 重复的可以通过字段和数字序号添加后缀标识
- 用pandas包批量读数据
  - `pd.concat + map + pd.read_csv + 文件名的list`
  - 记得读数据时将文件中表示空值的一些符号标记为空值
  - 通过主表DataFrame的`fillna`把初赛预测列填充好

## 历史记录处理

- Q: 历史记录的两个表LogInfo和UserUpdate怎样使用呢？
- A: 通常地说，历史记录与主表建立联系的难点在于：
  - 每个index对应多条记录
  - 每条记录分属不同的事件类别
  - 各条记录时间有先后顺序
- 风控中，登录/信息上传的起始时间和频率对衡量借款人的行为或许较重要。进一步，我们也可统计每类事件的频率
- 我们对此有解法：对历史记录按index来分组汇总（比如总频率和起始时间），使每个index记录唯一，就可以与主表接合了
  - 针对时间性问题：同时按index和时间分组计数/去重/...等
  - 针对类别性问题：同时按index和类别分组计数/去重/...等

## 数据批处理实现

这两个批量整理数据的组合非常实用，也是我们后续进行数据摘要与清洗的主力

- `pd.concat + map + (function) + (list)`
  - 逐行/逐列批量应用函数
  - `axis = 0`，按行：批量整理样本等
  - `axis = 1`，按列：批量整理变量等
- `pd.DataFrame: groupby + count/first/aggregate/...`
  - 将行/列分组，逐组汇总数据
  - `axis = 0`，按行：整理分组样本等（如历史记录）
  - `axis = 1`，按列：整理系列变量等（如不同类别/时期的第三方信息变量）

# 大纲

预备

数据读取

**数据摘要与清洗**

模型选择

模型训练与评估

模型组合与预测

回顾

# 数据摘要

- Q: 当我整理好数据集之后，我要做什么呢？
- A: 首先我们要从大局出发，简化并理解数据特征。  
具体地，可以通过循环/`pd.concat+map`对各变量处理汇总成一个表格。各行是变量名（原数据的每一列），而各列的内容有：
  - 变量是什么类型
  - 变量的空值/非空值数量
  - 变量出现频数前5大的值与数量，和其他值的数量（尾巴）
  - 数值变量的统计量：均值、方差、四分位数、最值



# 数据摘要展示：摘要后

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	
1	Columns	n	type	mean	std	min	0	25%	50%	75%	max	value 1	value 2	value 3	value 4	value 5	freq 1	freq 2	freq 3	freq 4	freq 5	freq	freq NA	
2	Education_Info1	89999	int64	0.06	0.24							0	1	0	1		84408	5591				0	0	
3	Education_Info2	89999	object									E	A	AM	AQ	AN	84408	2886	2044	300	233	128	0	
4	Education_Info3	89999	object									E	毕业	结业			84408	5416	175			0	0	
5	Education_Info4	89999	object									E	T	F	AR	V	84408	4281	791	209	182	128	0	
6	Education_Info5	89999	int64	0.03	0.18	0	0	0	0	0	1	0	1				87110	2889				0	0	
7	Education_Info6	89999	object									E	A	AM	AQ	U	87110	1640	1015	145	83	6	0	
8	Education_Info7	89999	object									E	不详				87110	2889				0	0	
9	Education_Info8	89999	object									E	T	不详	V		87110	2173	334	210	82	90	0	
10	ListingInfo	89999	datetime64[ns]									#####	#####	#####	#####	#####	1320	1093	1049	966	914	84657	0	
11	SocialNetwork_1	89999	int64	0.00	0.04	0	0	0	0	0	2	0	1	2			89891	101	7			0	0	
12	SocialNetwork_10	22293	float64	298.74	#####		0	25	89	261	75009	0	1	2	4	5	687	329	301	269	264	20443	67706	
13	SocialNetwork_11	40	float64	10.88	58.41	0	0	0	0.25	368	0	1	2	3	68	30	4	1	1	1	1	3	89959	
14	SocialNetwork_12	22702	float64	0.01	0.10	0	0	0	0	0	1	0	1				22492	210				0	67297	
15	SocialNetwork_13	89999	int64	0.22	0.42	0	0	0	0	0	4	0	1	2	3	4	70454	19326	210	8	1	0	0	
16	SocialNetwork_14	89999	int64	0.06	0.24	0	0	0	0	0	3	0	1	2	3		84426	5542	29	2		0	0	
17	SocialNetwork_15	89999	int64	0.03	0.16	0	0	0	0	0	2	0	1	2			87506	2491	2			0	0	
18	SocialNetwork_16	89999	int64	0.02	0.13	0	0	0	0	0	1	0	1				88465	1534				0	0	
19	SocialNetwork_17	89999	int64	0.25	0.44	0	0	0	0	1	3	0	1	2	3		67297	22620	80	2		0	0	
20	SocialNetwork_2	89999	int64	0.03	0.20	0	0	0	0	0	2	0	1	2			87558	2061	380			0	0	
21	SocialNetwork_3	2441	float64	#####	#####	0	9	47	221	1E+06	1	0	3	5	4	200	66	61	59	57	1998	87558		
22	SocialNetwork_4	2441	float64	217.29	378.52	0	30	86	222	3000	0	30	1	2	28	107	31	30	29	23	2221	87558		
23	SocialNetwork_5	2439	float64	418.56	#####	0	8	64	306	37924	0	1	2	3	4	215	98	81	56	40	1949	87560		
24	SocialNetwork_6	2438	float64	293.31	195.94	0	0	0	3	4870	0	1	2	3	4	1392	239	131	71	66	539	87561		
25	SocialNetwork_7	2493	float64	0.07	0.26	0	0	0	0	0	1	0	1				2313	180				0	87506	
26	SocialNetwork_8	22211	float64	139.76	#####	0	36	74	124	129496	0	1	2	70	50	611	229	198	178	172	20823	67788		
27	SocialNetwork_9	22244	float64	142.69	230.23	0	29	74	164	4304	0	2	3	4	6	445	358	293	269	262	20617	67755		
28	ThirdParty_Info_Pe	89480	float64	19.47	30.72	0	1	7	24	790	0	1	2	3	4	20445	6129	4777	3781	3202	51146	519		
29	ThirdParty_Info_Pe	89480	float64	0.97	1.63	0	0	0	0	1	28	0	1	2	3	4	50411	18647	8863	5007	2678	3874	519	

这样，我们可以通过表格的信息来对变量特性一目了然，并能帮助我们进行后续清洗工作。



# 数据清洗：目的

- Q: 为什么我们要进行数据清洗？
- A: 模型向往的是分布良好的数值，数据却有着骨感的现实
  - 空缺、类别（字符串）.....——模型陷进了Bug中
  - 稀疏性、共线性、极端值.....——模型迷失在数学中
  - 时间、地理名称.....——模型在人类知识面前踌躇不进
- 我们要为模型铺平数据的道路，使模型能在其上飞驰
- 整个数据分析流程的重中之重

## 数据清洗：思路

- Q: 这么多变量，真的需要我一个一个看来清洗吗？
- A: 不必的，我们要搭建通用的5步法先后处理掉清洗工作：
  1. 数值变量保留，**非数值变量全部转为数值变量**：
    - **有额外信息的非数值变量**可以转化为根据先验知识得到的数值中（比如时间转为年、月、日、星期、以及相对天数等，地名转为经纬度和城市等级，定序变量保留序数等）
    - **其余非数值变量全部用OneHotEncoder转为0-1哑变量**
  2. 对**一系列相似变量**可取求和、中位数、方差、最值、空值数等统计量取代原变量（比如几个省份、城市、不同时期的第三方信息变量等）。选取统计量重精不重多，尽量互相独立
  3. 删掉空值/同一值占绝大比例(比如99.9%)的**稀疏变量**
  4. 以相关矩阵的下三角阵中包含接近 $\pm 1$ 来筛选删除**共线变量**
  5. 用中位数（或平均数）填充**空值**，再进行标准化

## 数据清洗：实践

- Q: 什么时候该选用中位数而不是平均数填充空值呢？
- A: 数据分布不对称时，中位数比平均数更能保持排序关系
- Q: 我该怎样批量清洗这些变量呢？
- A: 我们可继续让批量转换`pd.concat+map`组合和分组汇总`groupby+aggregate`组合大显身手，而我们只需做几个对变量执行不同清洗功能的函数，然后把各变量按类别分类扔给它们
- 当我们完成了清洗的工作后，即将踏入建模阶段

# 大纲

预备

数据读取

数据摘要与清洗

模型选择

模型训练与评估

模型组合与预测

回顾

# Logistic Regression

- Q: 如果我刚入门机器学习，应该从什么模型开始？
- A: Logistic Regression：最简洁、快速、稳健的做法，可解释性强，适于工业界
- 但由于比赛以精度为标准，由于Logistic Regression对变量关系的线性限制，难以达到精度最优
- 但是我们在建模时可以充分发挥它的特性：
  - 通过增加L2罚函数减少过拟合
  - 作为基准，对数据清洗效果和模型表现进行快速评估
  - 与结构不同的模型加权组合预测，补充原模型精度和稳健性

## XGBoost ( 梯度强化树 )

- Q: 如果我对机器学习已经有所了解，打算以精度为目标，用什么模型效果好？
- A: 考虑这是一个非线性的分类问题，变量成分较多元，样本和变量间无固定模式关联（图像、语音、时间序列等）。如果以精度为目标，综合考虑稳健性、速度、通用性等因素可以首选XGBoost
- Q: XGBoost的原理是什么？有哪些重要参数？
- A: XGBoost一种梯度强化树(Gradient Boosting Trees)。好比用大石头雕刻人像，每棵决策树都凿掉一些石头（残差），然后对剩下的石头继续雕刻，直到雕出人形
  - 步长(eta)雕刀：大斧子 vs. 小凿子
  - 变量抽样(colsample\_bylevel)匠师：项羽 vs. 刘邦
  - 深度(depth)刀法：平推 vs. 直钻

## Keras (神经网络)

- Q: 如果我对XGBoost的精度仍不满足，想达到更好的预测效果，该如何做？
- A: 可以尝试神经网络Keras，并把XGBoost与多种模型组合
- XGBoost的出发点是各变量完全独立，而从决策树的二分关联叠加向真实关联趋近；而神经网络的出发点是各变量充满复杂的非线性关联，而不断去优化网络权重向真实关联趋近。两种模型结构具有较高的互补性
- 由于神经网络内部结构复杂，寻找最优解困难，除了合理搭建网络结构、优化参数之外，对数据建议把X各变量**标准化为正态分布**。
  - 目的：去除影响神经网络训练的有偏分布和极端值
  - 实现方法：可以通过rank和正态分布的百分位函数复合
  - 在本数据集能明显提高神经网络和LR的效果

## 模型对比

- 精度以相同的10-folds交叉验证为准
- 训练样本8万，变量经清洗后共389个，正态分布标准化
- 计算平台：Intel Core i5 4300U 双核 2.5 GHz, 8 GB 内存

模型	LR	XGBoost	Keras
类型	逻辑回归	梯度强化树	神经网络
平均精度(AUC)	0.775	<b>0.787</b>	0.771
最差精度(AUC)	0.759	<b>0.768</b>	0.753
单模型时间(s)	<b>8</b>	350	400
调参个数	<b>2</b>	8	10+
可解释性	<b>好</b>	中	水平不行
支持分布式	<b>是</b>	<b>是</b>	否



# 大纲

预备

数据读取

数据摘要与清洗

模型选择

模型训练与评估

模型组合与预测

回顾

## 交叉验证与模型训练

- Q: 为什么要用交叉验证？怎样用？
- A: 较比用单训练预测集建模，交叉验证的优势主要有：
  - 更准确地估计模型的预测精度
  - 可以预估模型预测效果的区间范围
  - 减少模型优化过程中对单验证集的过拟合情况
- 以10-folds为例做交叉验证：
  1. 把数据按列分成X和Y（预测目标列为Y，其他的列为X）
  2. 把样本行的index随机拆成10份保存起来
  3. 每次取1份index作验证集，另外9份index粘起来作训练集，以取的X和Y的训练和验证集训练模型，把模型保存起来
  4. 依次取10组不同的index，得到一组10个模型
  5. 预测时用10个模型预测结果取平均

# 模型评估

- Q: 模型训练的过程中，我们如何评价模型的效果？
- A: 我们可以从验证集分数和训练时间两个角度予以评价。其中交叉验证的验证集分数包含着模型效果的更多信息：
  - 均值：反映模型精度
  - 标准差/置信区间：反映模型稳健性，预估模型预测效果在不同数据集上的波动范围。在独立正态假设下，用模型均值预测不同数据集（比如排行榜和非公开的验证集）的标准差大约是验证分数标准差的 $\frac{1}{\sqrt{K}}$ （K-folds交叉验证）
  - 箱线图(box-plot)：可视化地展现验证分数的分布规律并发现异常情况

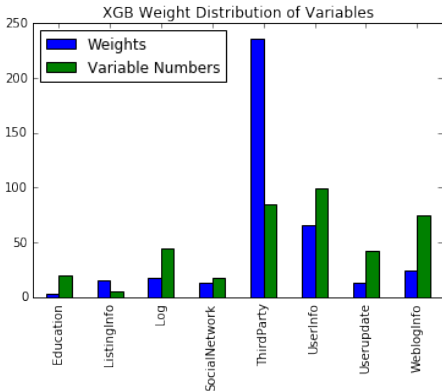
## 变量评估

- Q：我们该如何了解Y受哪些变量影响呢？
- A：可以用XGBoost判断变量重要性，再用LR看影响方向
  - 线性影响：Logistic Regression的系数来评估
    - 正负号表示对Y值的正影响或负影响
    - （变量X标准化后）系数绝对值越高影响越大
    - 受共线性影响大，可能使变量与Y虚假相关
  - 非线性影响：XGBoost各变量的相对频率来评估
    - 可用  $\frac{\text{fscore}}{\text{Mean}(\text{fscore})}$  计算相对频率，fscore为XGB模型 `get_fscore`得到的树分支挑选各变量的频数
    - 相对频率大于1的为在模型中明显有效的变量，远小于1则为较不重要的变量
    - 受共线性影响小，稳健性高

## 变量评估展示

如图为XGB变量相对频率按组汇总，可据此改进我们的

- 数据收集：增加对重要变量的收集
- 变量处理：针对重要变量在模型清洗阶段进一步转换组合



## 参数优化

- Q: 如何进行参数优化？怎么选取初始值？
- A: 模型调参是非常考验耐心和时间的过程
  1. 在调参前，首先要理解模型和参数的含义，这步非常关键
  2. 先用单数据集，从默认值开始，手工逐个调参熟悉模型，小范围用等差数列，大范围用等比数列，确定合理参数范围
  3. 确定大致范围后，可以用交叉验证+自动搜索来得到最优参数，如Python的HyperOpt包
- 如果模型训练是以10-folds验证的话，我们可以用5-folds交叉验证来自动搜索寻找最优参数
  - 节约调参时间
  - 数据集不同，减少对交叉验证结果的过拟合
- 在找到最优参数后，我们重新在原交叉验证集上用最优参数训练模型，至此模型训练阶段结束

# 大纲

预备

数据读取

数据摘要与清洗

模型选择

模型训练与评估

模型组合与预测

回顾

## 模型组合

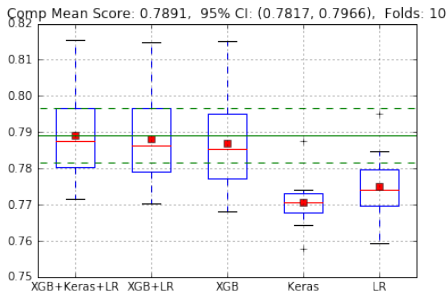
当训练优化好各组交叉验证模型后，就可将各组模型**加权平均**预测了，比如我们这里使用XGBoost，Keras和LR三组模型加权平均，并使用HyperOpt取得最优加权重

- Q: 为什么要使用模型加权平均而不是最优模型预测？
- A: 数学上，如果一个无偏模型的预测方差为 $V_1$ ，当我们加入另一个无偏而完全独立的模型，该模型预测方差为 $V_2$ ，当我们对两模型的预测结果加权平均取最优解时，预测方差会变成原 $V_1$ 的 $\frac{V_2}{V_1+V_2}$ 倍
- 当然因为实际数据集和模型结构所限，真实的模型往往是有偏的，而只有一小部分相互独立，因此改进效果并没有理论上那样明显，但至少是一种比较稳健的方法
- 预测提交时，我们先对三组模型的交叉验证预测Y分别算术平均，再把这三个Y照权重加权平均，就可以提交了



## 效果展示

- 不同模型组合在同一10-folds交叉验证集上的得分分布
- 训练样本8万，变量经清洗后共389个，正态分布标准化
- 最优权重：XGB+LR=90:10，XGB+Keras+LR=75:20:5
  - Keras虽然预测精度较低，但结构互补进一步改善模型效果



## 预测反馈

- Q: 为什么排行榜上的结果要比交叉验证的结果要好/差？
- A: 通常来说，预测的结果稍可能比交叉验证略好，原因是  
在不同数据集的交叉验证模型取平均形成部分互补减小误差
- 当然，因为预测集数据分布有随机性，预测效果的区间大致  
可以通过交叉验证的均值  $\pm \frac{2}{\sqrt{K}}$  标准差来估算（K-folds）
- 我们也要在全程中注意避免过拟合，包括：
  - 避免将Y的真值/预测信息在数据清洗或建模时引入到X中
  - 模型优化时采用另外划分的交叉验证集
  - 尽量能说清所做每一步处理的必要性和通用性
  - 注意：反复尝试变量组合提高验证集分数时，可能造成过拟合

# 大纲

预备

数据读取

数据摘要与清洗

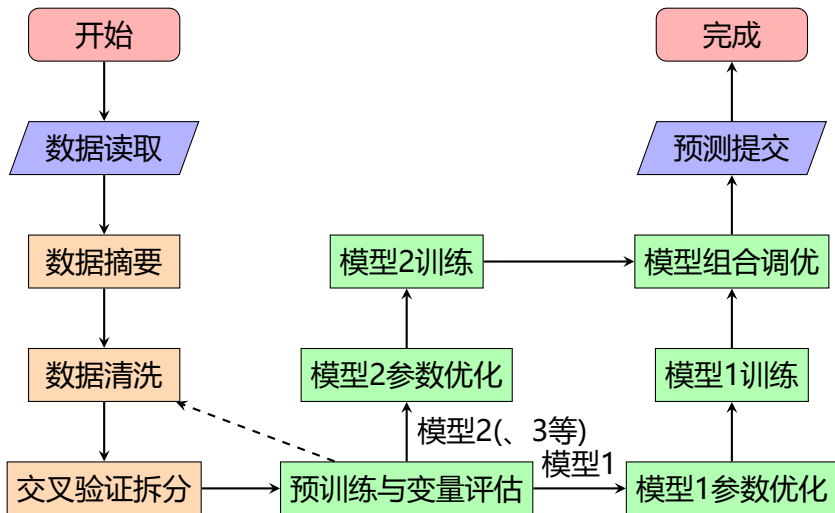
模型选择

模型训练与评估

模型组合与预测

回顾

# 流程回顾



## 流程思想要点

- 为数据建模逐步搭建通用的函数（清洗、拆分、训练、优化等），将整个流程尽量自动化、可重复、可移植
- 注意对数据建模的整个过程进行评估（时间、复杂性、过拟合等），减少不必要的中间环节
- 清洗数据时，构造的人工变量要少而精，在相互独立和完备覆盖之间取得平衡，从而为模型增加有效信息帮助预测

## 改进潜力

- 当我们完成所有数据建模的必要工作时，在有需要而且有资源足够的前提下，可以在当前预测精度上进一步改进
  - 数据清洗：对预测Y较重要的变量之间，可尝试不同种类的组合变换（四则运算、各种分布变换、系列变量的各类统计量等），增加模型可以发掘的有效信息
  - 模型选择：可引入更多种类的模型，如随机森林，不同结构（层数、激活函数等）的神经网络等，改善模型互补性
  - 参数优化：减小梯度模型步长，在更多参数维度中搜索最优
- 但是
  - 会继续指数级增加所需时间、精力、计算量
  - 精度提升和算法的通用性改进会明显减少
  - 可能陷入为改进而改进的循环中
  - 直到机器学习界的AlphaGo取代人工劳作的数据分析师

# 局限与反思

- 同时，我们目前所做的数据模型是很有限的：
  - 数据的预测局限：当试图穷尽数据处理、模型、调参等方法时，投入时间、复杂度与计算量会呈现指数级增长，然而往往仅能取得1%，甚至0.1%的提升，与真理相去仍然甚远
  - 模型的功能局限：模型只是指引决策的参考，却不能为它的决策本身进行价值判断和承担责任
  - 模型的反馈局限：模型在欠拟合的经济/数据体系中发挥正面作用，当经济/数据体系已经过拟合，模型和体系的系统性风险会加倍扩大（金融危机、评级垄断、高频交易、...）
- 我们到目前所学习与创作的，只是浩如烟海的世界中一块小小的砖瓦，我们生命的盼望却不在这里
- 愿恩惠平安从主基督耶稣临到所见的人

神爱世人，甚至将他的独生子赐给他们，叫一切信他的，不至灭亡，反得永生。——约翰福音3章16节