信用违约预测

|  |  |
| --- | --- |
| 姓名 | 学号 |
| 余若涵 | 3180105412 |
| 范源颢 | 3180103574 |

格式：小四 1.25倍行距 10页以上 （意思是最后需要转成word格式？）

# 一、研究背景与意义

TODO

估计客户的信用评级是银行风险管控非常重要的组成部分，银行借给债务人的借款如果无法按时收回，银行的资产将面临不必要的风险，同时债务人的信用也会受到影响。为此，银行需要在借款之前根据数据估计客户的还款能力，评价信用风险等级。

# 二、国内外研究现状与存在问题

TODO

# 三、研究目标与研究内容

TODO

在本项目中，我们的目的是：根据客户的历史表现判断客户是否有能力及时还款。

# 四、数据集分析

## **4.1** 数据集基本信息

本项目中，我们使用的是由Home Credit公司提供的贷款信息数据集。数据集包含application\_train.csv

和application\_test.csv两个文件。训练集共提供307511条记录，每条记录包含121个特征和1个Target 列；测试集包含48744条记录。

数据集中的每条记录都表示了一次贷款，Target的0/1值反映了借款人是否按时还款。121个特征中，有

106个数值特征和16个标称特征，这些特征可分为以下四类(只列出部分)：

1. 借款人基本信息

CODE\_GENDER - 性别，

DAYS\_BIRTH - 客户年龄，

CNT\_CHILDREN - 子女数，

CNT\_FAM\_MEMBERS - 家人数，

OCCUPATION\_TYPE - 职业，

……

1. 借款人经济背景

FLAG\_OWN\_CAR - 是否拥有轿车，

FLAG\_OWN\_REALTY - 是否拥有不动产，

AMT\_INCOME\_TOTAL - 借款人总收入，

APARTMENT\_AVG - 住房评分，

……

1. 本次贷款相关信息

AMT\_CREDIT - 信用额度，

AMT\_ANNUITY - 贷款年金，

WEEKDAY\_APPR\_PROCESS\_START - 申请贷款的星期，

HOUR\_APPR\_PROCESS\_START - 申请贷款的小时，

……

1. 其他信息

FLAG\_MOBIL - 是否提供手机号码，

FLAG\_CONT\_MOBILE - 手机是否可打通，

REGION\_RATING\_CLIENT - 所在地区评分，

EXT\_SOURCE\_1/EXT\_SOURCE\_2/EXT\_SOURCE\_3 - 通过其他信息计算的该客户评分，

……

## **4.2** 可视化分析

以下可视化分析针对未经预处理的训练集aplication\_train.csv

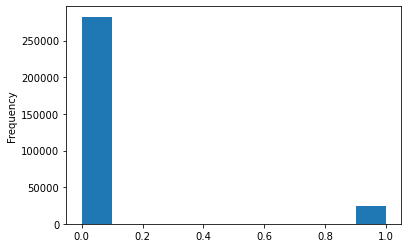
### 4.2.1 Target列的分布

app\_train**[**'TARGET'**].**value\_counts**()**

app\_train**[**'TARGET'**].**astype**(int).**plot**.**hist**()**

输出：

1. 282686
2. 24825



从上图可以看出，训练数据集中的违约记录数远大于未违约记录数，整个训练集是极度不平衡的，这种情况下，模型会花费更多的时间和“精力”去拟合未违约记录(target为0)，所以为了提供模型的效率，在之后我们将采取欠采样或过采样的方法来平衡数据集。

### 4.2.2 特征与Target的相关性

1. 计算特征与**Target**的相关性

# Calculate the correlations and sort

correlations **=** app\_train**.**corr**()[**'TARGET'**]**

correlations **=** correlations**.**drop**([**"TARGET"**]).abs().**sort\_values**()**

1. **10**个相关性最小的特征

# 10 least relevant features

correlations**.**head**(**10**)**

输出：

FLAG\_DOCUMENT\_20 0.000215

FLAG\_DOCUMENT\_5 0.000316

FLAG\_CONT\_MOBILE 0.000370

FLAG\_MOBIL 0.000534

FLAG\_DOCUMENT\_12 0.000756

AMT\_REQ\_CREDIT\_BUREAU\_WEEK 0.000788

AMT\_REQ\_CREDIT\_BUREAU\_HOUR 0.000930

FLAG\_DOCUMENT\_19 0.001358

FLAG\_DOCUMENT\_10 0.001414

FLAG\_DOCUMENT\_7 0.001520

Name: TARGET, dtype: float64

上面输出结果中的FLAG\_DOCUMENT\_X特征表示的是在贷款时是否提交某份文件，这类特征有20个，大部分相关性都很低。FLAG\_MOBIL和FLAG\_CONT\_MOBILE表示的分别是客户是否拥有移动电话、电话是否可打通，AMT\_REQ特征表示的是申请贷款前客户向信贷局进行查询的次数。

可以看出，这些特征的与Target列的相关性都在万分之一数量级，可以归为very weak级别。在之后的特征工程中我们将会试试它们是否可以衍生出足够好的新特征，或者直接删除这些特征，以减少运算量并提高预测的准确度。

**2) 20**个相关性最大的特征

# Display 20 most relevant featrues

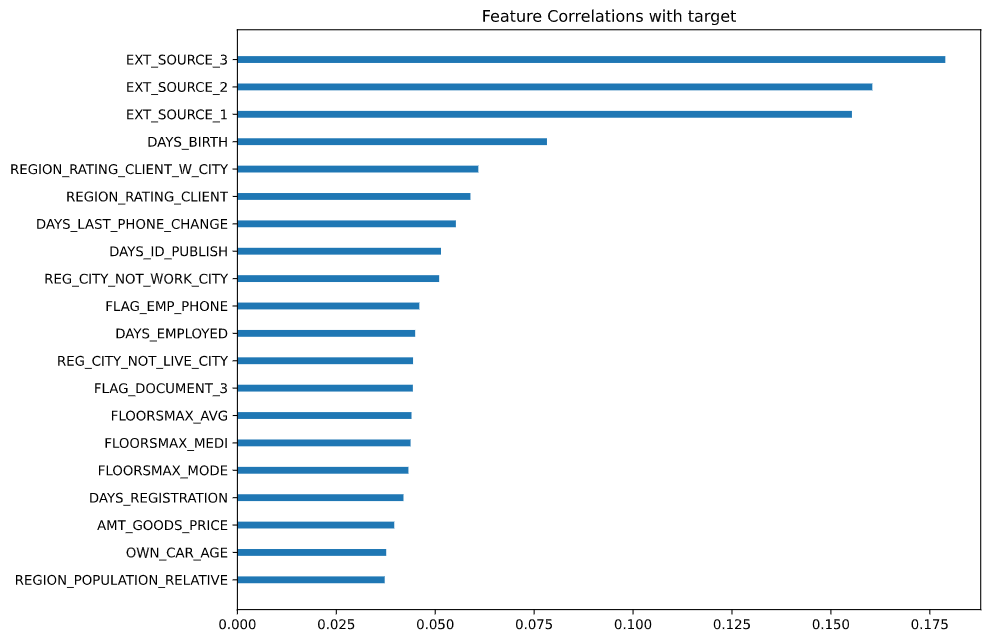
corrs **=** correlations**.**tail**(**20**)**

plt**.**figure**(**figsize **=** **(**10**,** 8**))**

plt**.**bar**(** x**=**0**,** bottom**=**corrs**.**index**.**astype**(str),**

height**=**0.25**,** width**=abs(**corrs**.**values**),** orientation**=** "horizontal"**)**

plt**.**title**(**'Feature Correlations with target'**)**



上图中（相关性已取绝对值）排在前三的EXT\_SOURCE特征是专业人士依据外部数据对该客户进行的评分，值域为[0, 1]，它们的相关性都超过了0.15；DAY\_BIRTH表示的是客户的年龄；两个

REGION\_RATING特征是对客户所在地的评分，分为三档，用{1, 2, 3}表示。接下来我们将对这些特征进行可视化分析。

**4.2.3** 客户年龄分布

plt**.**figure**(**figsize **=** **(**10**,** 8**))**

# KDE plot of loans that were repaid on time

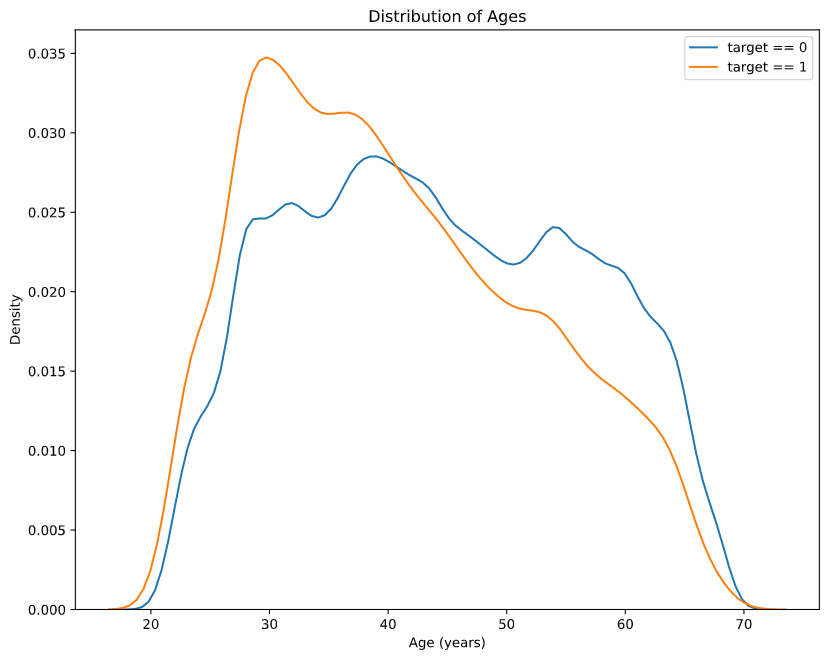
sns**.**kdeplot**(**app\_train**.**loc**[**app\_train**[**'TARGET'**]** **==** 0**,** 'DAYS\_BIRTH'**]** **/** **-**365**,** label **=** 'target == 0'**)**

# KDE plot of loans which were not repaid on time

sns**.**kdeplot**(**app\_train**.**loc**[**app\_train**[**'TARGET'**]** **==** 1**,** 'DAYS\_BIRTH'**]** **/** **-**365**,** label **=** 'target == 1'**)**

# Labeling of plot

plt**.**xlabel**(**'Age (years)'**);** plt**.**ylabel**(**'Density'**);** plt**.**title**(**'Distribution of Ages'**);**



上图中，橘线展示的是违约用户的年龄分布，蓝线为未违约用户的年龄分布。从总可以看出，年轻用户

([25, 40]年龄区间内)更容易违约，而年纪较大的用户([50, 65]年轻区间内)则更可能按时付清欠款。

为了更好地分析客户年龄与违约的关系，我们以5年为一个区间宽度进行分箱处理，并用柱状图展示各个区间内的违约率。如下所示。

# Age information into a separate dataframe

age\_data **=** app\_train**[[**'TARGET'**,** 'DAYS\_BIRTH'**]]**

age\_data**[**'YEARS\_BIRTH'**]** **=** age\_data**[**'DAYS\_BIRTH'**]** **/** **-**365

# Bin the age data

age\_data**[**'YEARS\_BINNED'**]** **=** pd**.**cut**(**age\_data**[**'YEARS\_BIRTH'**],** bins **=** np**.**linspace**(**20**,** 70**,** num **=** 11**))**

age\_data**.**head**(**10**)**

# Group by the bin and calculate averages

age\_groups **=** age\_data**.**groupby**(**'YEARS\_BINNED'**).**mean**()**

plt**.**figure**(**figsize **=** **(**8**,** 8**))**

# Graph the age bins and the average of the target as a bar plot

plt**.**bar**(**age\_groups**.**index**.**astype**(str),** 100 **\*** age\_groups**[**'TARGET'**])**

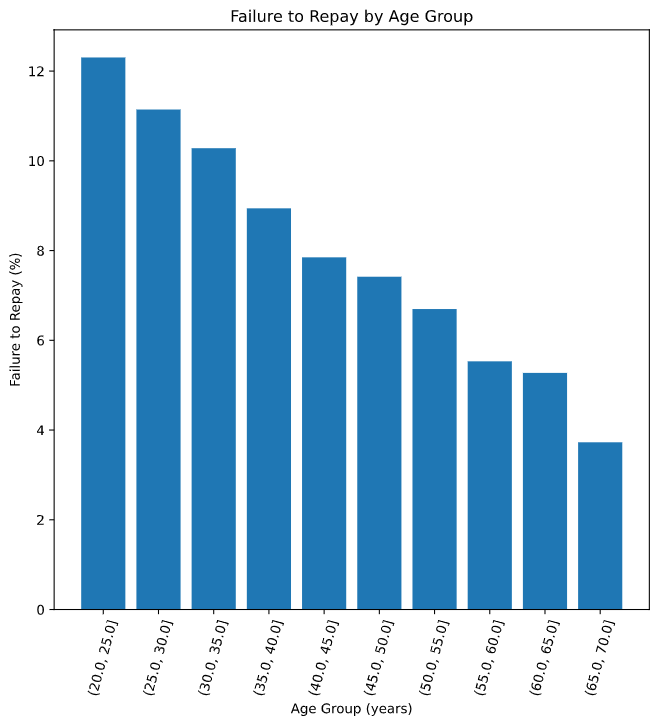
# Plot labeling

plt**.**xticks**(**rotation **=** 75**);**

plt**.**xlabel**(**'Age Group (years)'**);**

plt**.**ylabel**(**'Failure to Repay (%)'**)**

plt**.**title**(**'Failure to Repay by Age Group'**);**



从图中可以看出，随着年龄的增高，违约率一直在下降。年纪最小的三个群体有超过10%的违约率，而年纪最大群体的违约率则低于5%. 这提示我们，年龄将会是一个非常重要的特征，在之后的特征工程中，我们也许可以基于年龄衍生出新的重要特征。

**4.2.4** 附加特征**(EXT\_SOURCE)**的分布上文4.2.2的分析显示，三个EXT\_SOURCE列与Target列有很大的相关性。下面探究它们的分布。

plt**.**figure**(**figsize **=** **(**10**,** 12**))**

# iterate through the sources

**for** i**,** source **in** **enumerate([**'EXT\_SOURCE\_1'**,** 'EXT\_SOURCE\_2'**,** 'EXT\_SOURCE\_3'**]):**

# create a new subplot for each source

plt**.**subplot**(**3**,** 1**,** i **+** 1**)**

# plot repaid loans

sns**.**kdeplot**(**app\_train**.**loc**[**app\_train**[**'TARGET'**]** **==** 0**,** source**],** label **=** 'target == 0'**)**

# plot loans that were not repaid

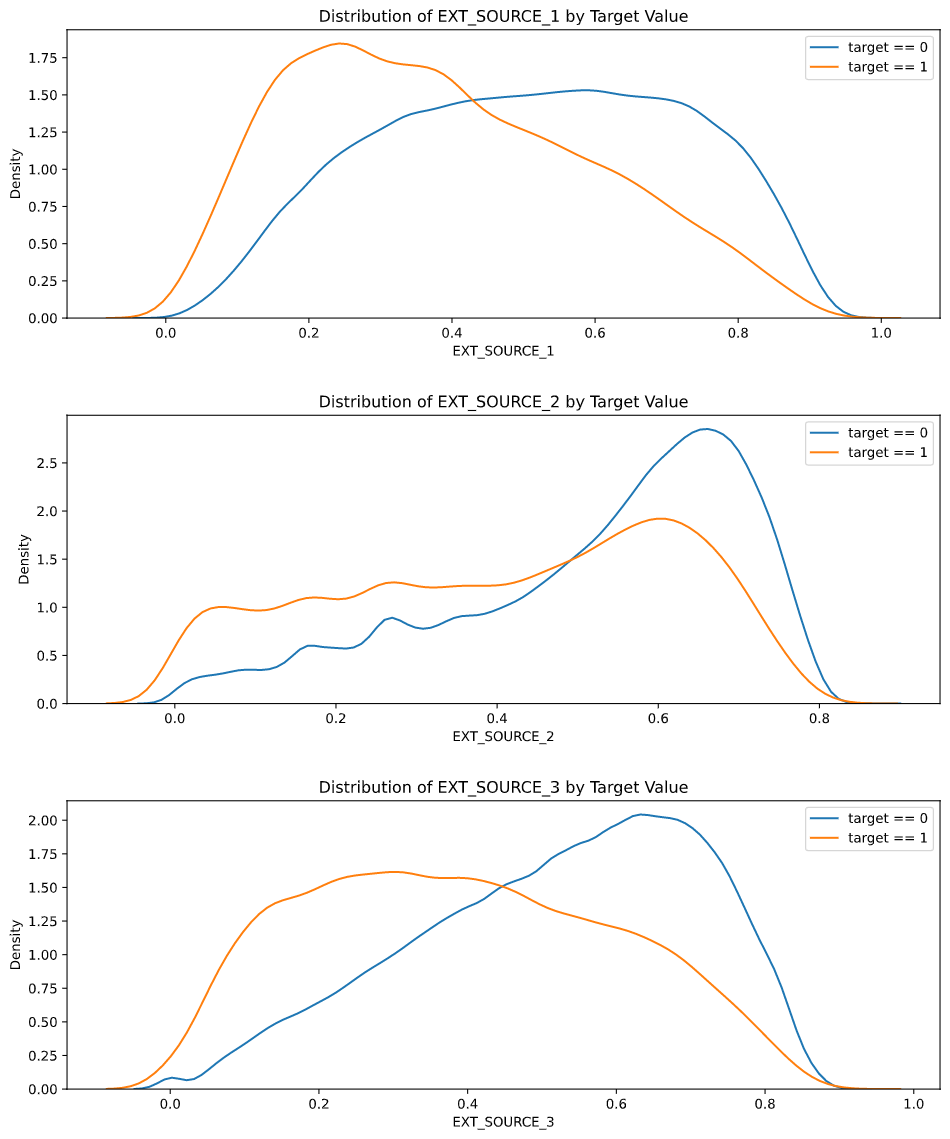
sns**.**kdeplot**(**app\_train**.**loc**[**app\_train**[**'TARGET'**]** **==** 1**,** source**],** label **=** 'target == 1'**)**

# Label the plots

plt**.**title**(**'Distribution of %s by Target Value' **%** source**)**

plt**.**xlabel**(**'%s' **%** source**);** plt**.**ylabel**(**'Density'**);**

plt**.**tight\_layout**(**h\_pad **=** 2.5**)**



可以看出，在以上三张图中都能很好地区分违约与未违约这两条曲线。其中EXT\_SOURCE\_3的效果最好，当其值较小时，有更大概率发生违约。而EXT\_SOURCE\_2虽然也能很好区分两条曲线，但是两曲线的整体趋势非常相似，所以也许它在模型的效果会不如另两个特征。

### 4.2.5 地区评分(REGION\_RATING)的分布

**for** i**,** source **in** **enumerate([**'REGION\_RATING\_CLIENT'**,** 'REGION\_RATING\_CLIENT\_W\_CITY'**]):**

plt**.**figure**(**figsize **=** **(**5**,** 3**))**

tot **=** app\_train**[**app\_train**[**'TARGET'**]!=-**1**].**groupby**(**source**).**count**()[**'TARGET'**]**

default **=** app\_train**[**app\_train**[**'TARGET'**]==**1**].**groupby**(**source**).**count**()[**'TARGET'**]**

plt**.**bar**(**tot**.**index**,** default **/** tot**,** width**=**0.35**,** color**=**'lightskyblue'**)**

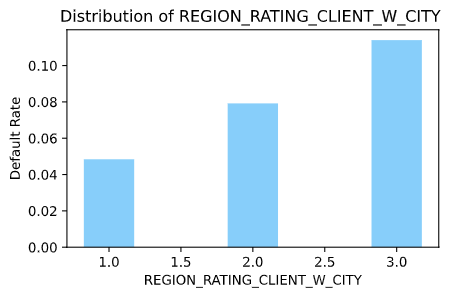
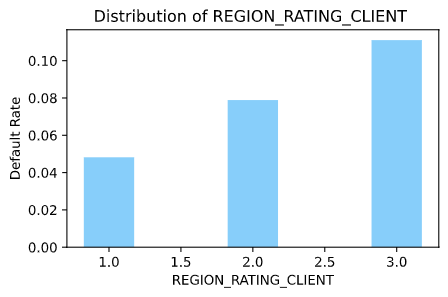
# Label the plots

plt**.**title**(**'Distribution of %s ' **%** source**)**

plt**.**xlabel**(**'%s' **%** source**);**

plt**.**ylabel**(**'Default Rate'**);**

plt**.**show**()**



图中柱状图表示的是违约率即违约客户占总客户的比例。两个特征都呈现出一样的趋势，即评分越高则违约率越大。可以看出这两张柱状图的趋势和数值都非常相似，所以我们推测这两个特征的相关性应该非常高，接下来进行具体分析。

**4.2.6** 特征间的相关度

由于本数据集特征过多，不适合对所有特征进行相关度可视化，所以这一部分我们只选取了上面讨论过的六个特征进行分析。

# Extract the EXT\_SOURCE variables and show correlations

ext\_data **=** app\_train**[[**'TARGET'**,** 'EXT\_SOURCE\_1'**,** 'EXT\_SOURCE\_2'**,** 'EXT\_SOURCE\_3'**,** 'DAYS\_BIRTH'**,** 'R

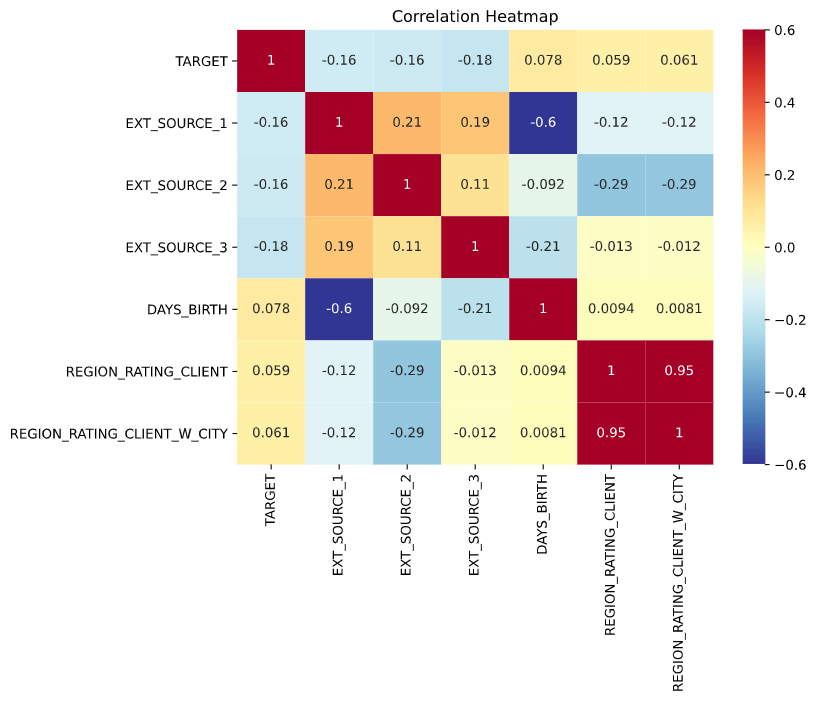
ext\_data\_corrs **=** ext\_data**.**corr**()**

# Heatmap of correlations

plt**.**figure**(**figsize **=** **(**6**,** 6**))**

sns**.**heatmap**(**ext\_data\_corrs**,** cmap **=** plt**.**cm**.**RdYlBu\_r**,** vmin **=** **-**0.6**,** annot **=** **True,** vmax **=** 0.6**)**

plt**.**title**(**'Correlation Heatmap'**);**



上面的热力图中，颜色越深表示二者之间相关度越高。可以看出两个REGION\_RATING类特征有高达 0.95的相关性，这两个特征都是对客户所在地区的评分，二者的差异仅在于：客户所在城市是否在考虑范围内，于是我们可以认为，它们包含的信息是基本一致的，在特征工程中我们会考虑删除这两个特征中的一个，以提高效率。此外，EXT\_SOURCE\_1和年龄的相关度也有0.6，这说明年龄可能是影响 EXT\_SOURCE\_1得分的一个重要因素。

## **4.3** 数据预处理

TODO

编码

缺失

异常

# 五、特征工程

本项目中的特征工程，主要分为新特征衍生和特征降维两部分。在新特征衍生部分，我们将尝试使用多项式特征构造法、专业知识特征构造法以及特征自动生成工具来衍生出新特征，并保留其中效果较好的参与建模。在特征降维中，我们会去除一些贡献度较小的特征：如缺失值过多的特征、与Target相关度过小的特征、共线特征等，以提高模型的效率。

## **5.1** 新特征衍生

**5.1.1** 多项式特征

多项式特征是一种非常简单的特征构造方法，其衍生的新特征为原有特征的n次多项式，形如f1 \* f2 ^

2，f3 ^ 3。

实操中，我们参考了[GitHub上的一篇文章。](https://jakevdp.github.io/PythonDataScienceHandbook/05.04-feature-engineering.html)

Dcikit-Learn直接为我们提供了一个PolynomialFeatures的类来构造多项式特征，创建该类的实例时需要传入一个参数degree即为多项式的最高幂次。为了避免模型出现过拟合，degree不应取得过高。项目中我们将degree设为3，使用上文分析过的年龄、EXT\_SOURCE作为原始特征，尝试构造多项式特征。具体操作如下。

**from** sklearn**.**preprocessing **import** PolynomialFeatures

# Select features

poly\_features **=** app\_train**[[**'EXT\_SOURCE\_1'**,** 'EXT\_SOURCE\_2'**,** 'EXT\_SOURCE\_3'**,** 'DAYS\_BIRTH'**,** 'TARGET

# Create the polynomial object with specified degree

poly\_transformer **=** PolynomialFeatures**(**degree **=** 3**)**

# Train the polynomial features

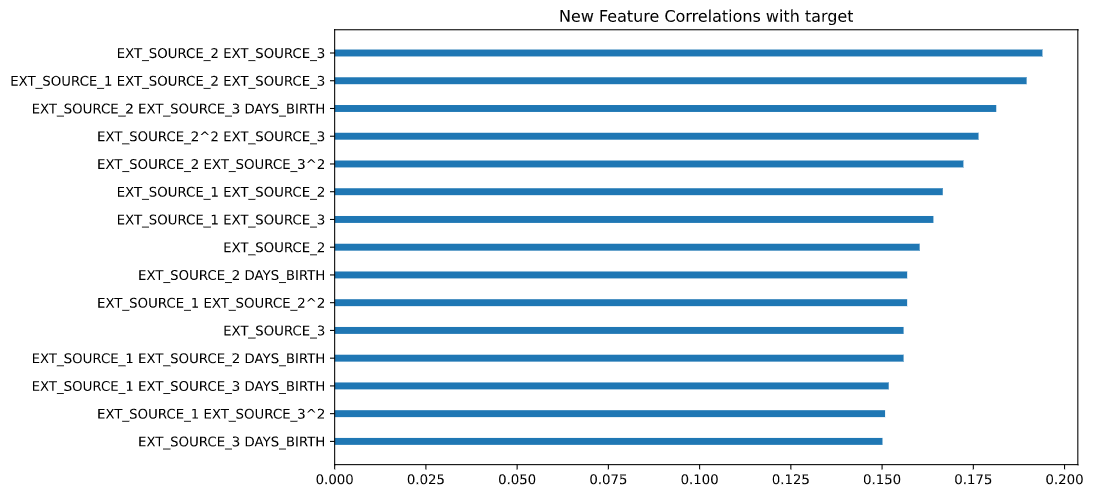
poly\_transformer**.**fit**(**poly\_features**)**

# Transform the features

poly\_features **=** poly\_transformer**.**transform**(**poly\_features**)**

poly\_features\_test **=** poly\_transformer**.**transform**(**poly\_features\_test**)**

计算新特征与Target的相关性并进行可视化，结果如下。（相关性大小已取绝对值）



可以看出，许多特征的相关性超过了原有特征。排在前几的特征均为EXT\_SOURCE特征的乘积，新特

征EXT\_SOUCE\_2 \* EXT\_SOURCE\_3 的相关性达到了0.19，相比原特征EXT\_SOUCE\_2的0.16略有提升。但也需注意，EXT\_SOURCE\_2 在于 DAYS\_BIRTH相乘后，新特征相关性反而略有下降。

在建模前，我们会选择一部分新特征加入到数据集中，并尝试不同的特征组合，根据效果决定最终选择。

**5.1.2** 专业知识特征

专业知识特征是利用金融知识和经验，人为地构造一些我们主观上认为可能会影响客户违约情况的新特征。

由于我们的金融知识有限，所以这一部分我们参考了[Kaggle上的一篇文章](https://www.kaggle.com/willkoehrsen/start-here-a-gentle-introduction)，构造出以下四个特征。

* CREDIT\_INCOME\_PERCENT ：信用额度(credit)占客户收入(income)的百分比
* ANNUITY\_INCOME\_PERCENT ：贷款年金(annuity)占客户收入(income)的百分比
* ANNUITY\_CREDIT\_PERCENT ：贷款年金(annuity)占信用额度(credit)的百分比
* DAYS\_EMPLOYED\_PERCENT ：客户工作天数(days\_employed)占年龄(days\_birth)的百分比

app\_train\_domain **=** app\_train**.**copy**()**

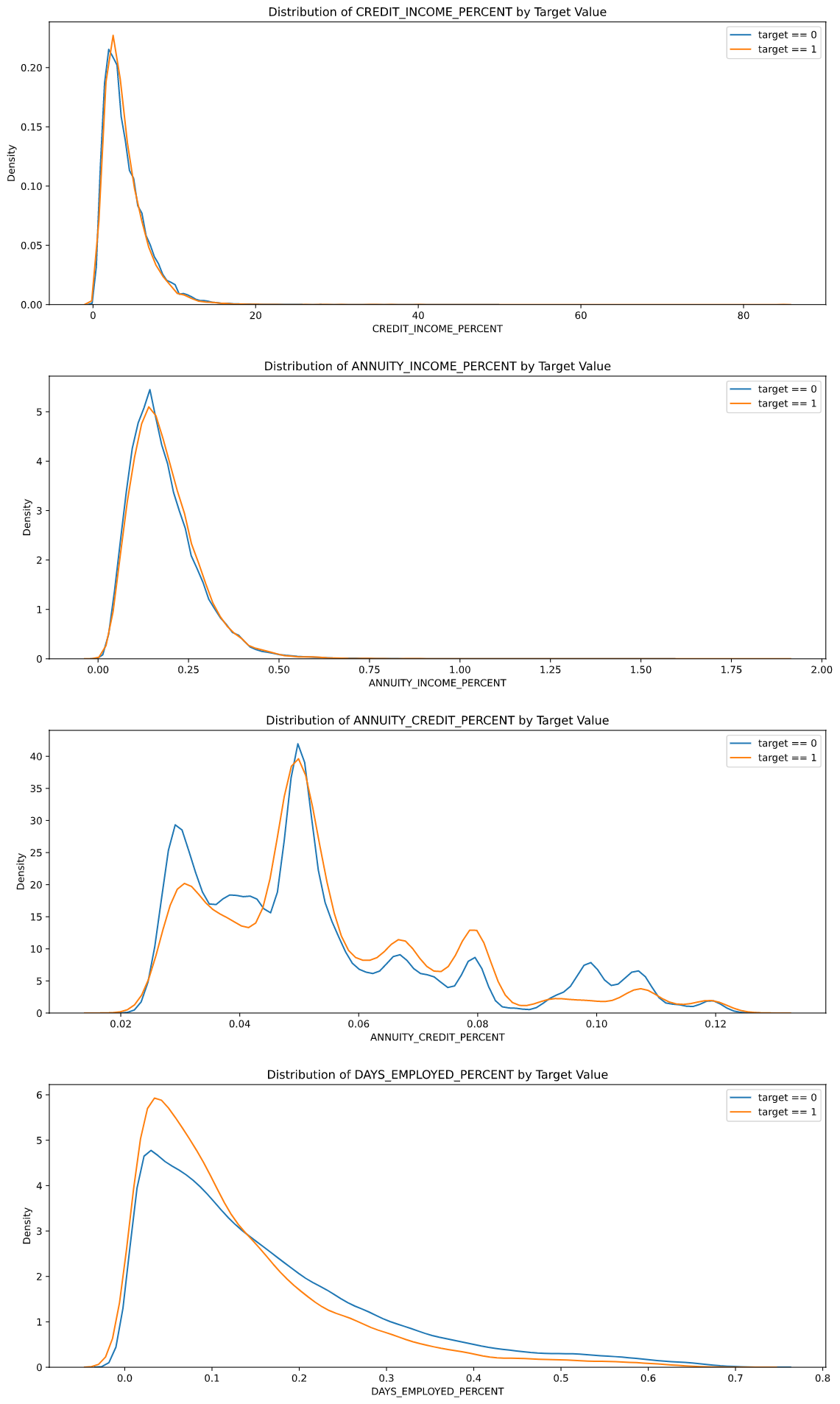
app\_train\_domain**[**'CREDIT\_INCOME\_PERCENT'**]** **=** app\_train\_domain**[**'AMT\_CREDIT'**]** **/** app\_train\_domain**[**'AMT\_INCOME\_TOTAL'**]**

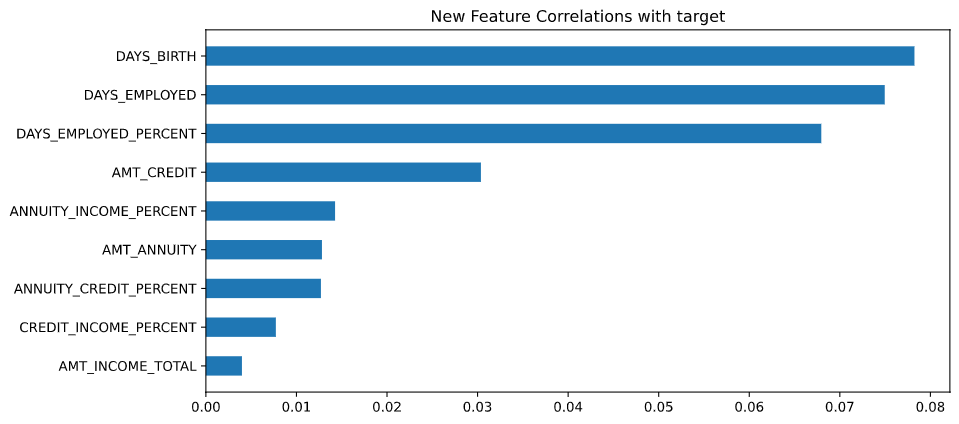
app\_train\_domain**[**'ANNUITY\_INCOME\_PERCENT'**]** **=** app\_train\_domain**[**'AMT\_ANNUITY'**]** **/** app\_train\_domain**[**'AMT\_INCOME\_TOTAL'**]**

app\_train\_domain**[**'ANNUITY\_CREDIT\_PERCENT'**]** **=** app\_train\_domain**[**'AMT\_ANNUITY'**]** **/** app\_train\_domain**[**'AMT\_CREDIT'**]**

app\_train\_domain**[**'DAYS\_EMPLOYED\_PERCENT'**]** **=** app\_train\_domain**[**'DAYS\_EMPLOYED'**]** **/** app\_train\_domain**[**'DAYS\_BIRTH'**]**

对上面四个特征进行可视化分析，得到分布图和相关性柱状图如下：





从分布图来看，前两个特征(CREDIT\_INCOME, ANNUITY\_INCOME)的违约未违约曲线拟合度非常高，所以效果应该不是很好。后两个特征(ANNUITY\_CREDIT, DAYS\_EMPLOYED)的曲线有较明显的差异，但是不同客户的曲线趋势仍是一致的。从相关性来看，各个新特征的相关性都较高，或基于原特征有部分提升。至于这些特征对我们的模型具体有多大的贡献，还需要在建模时进行实验。

**5.1.3** 特征生成工具 **Feature Tool**

## **5.2** 特征降维

**5.2.1** 共线特征去除

**5.2.2** 缺失值过多特征去除

**5.3.3** 重要性较低特征去除

# 六、研究方法与模型思路

# 这一章偏重于理论介绍

# 七、实验与分析

这一章偏重于的对(五)的实现，以及对比不同实现方法的效果好坏

七、结论与展望

TODO

# 参考文献