**从企业支付网络中计算客户的违约概率**

刘泽军 201830730338

唐浩翔 201830701109

邓宏睿 201864040069

目录

**[1.](#_Toc76080079)****[项目背景概述](#_Toc76080079)** [3](#_Toc76080079)

**[1.1 选题背景](#_Toc76080080)** [3](#_Toc76080080)

**[1.2 Networkx：图操作和分析](#_Toc76080081)** [4](#_Toc76080081)

**[1.3 图网络的特征：PageRank](#_Toc76080082)** [4](#_Toc76080082)

**[1.4 PageRank的改进：TrustRank](#_Toc76080083)** [6](#_Toc76080083)

**[2.](#_Toc76080084)****[假设与改进](#_Toc76080084)** [7](#_Toc76080084)

**[2.1 问题分析与改进](#_Toc76080085)** [7](#_Toc76080085)

**[2.2 模拟数据](#_Toc76080086)** [8](#_Toc76080086)

**[3.](#_Toc76080087)****[估计违约概率](#_Toc76080087)** [8](#_Toc76080087)

**[3.1 项目结构](#_Toc76080088)** [8](#_Toc76080088)

**[3.2 演示说明](#_Toc76080089)** [9](#_Toc76080089)

**[3.3 更多打算](#_Toc76080090)** [12](#_Toc76080090)

**[参考文献](#_Toc76080091)** [12](#_Toc76080091)

**摘要：**

本文研究支付网络中企业违约概率的估计，尝试量化一个节点或一组节点的网络属性与节点所代表的公司风险之间的依赖关系。我们利用公司风险与相应节点的网络特征之间的相关性，通过模拟不同的支付方和收款方，预测公司的风险评级和违约概率。我们首先通过PageRank算法对违约概率进行初步估计，得到一组结果；在此基础上运用TrustRank算法改进估计方法，引入公司“好与坏”以及交易金额和交易频率对违约的影响，模型结果有较大改善。

# 项目背景概述

**1.1 选题背景**

传统金融借贷业务贷前流程可能会是这样的：用户前来申请借贷，机构首先经过欺诈识别，把欺诈团伙和主观欺诈的个人拒绝掉，然后对通过的人做信用评估，最后根据额度模型，算出利润最大化时的放款金额。

对于借款人的欺诈识别和违约概率分析，有很多种图算法方法，比如知识图谱、标签传播算法、PageRank算法、社团发现算法等等。图的好处在于概念和术语通常是直观的，各个借款之间的联系可以直观的用图或网络的形式表现出来，便于网络特征的提取和网络信息的深度挖掘。

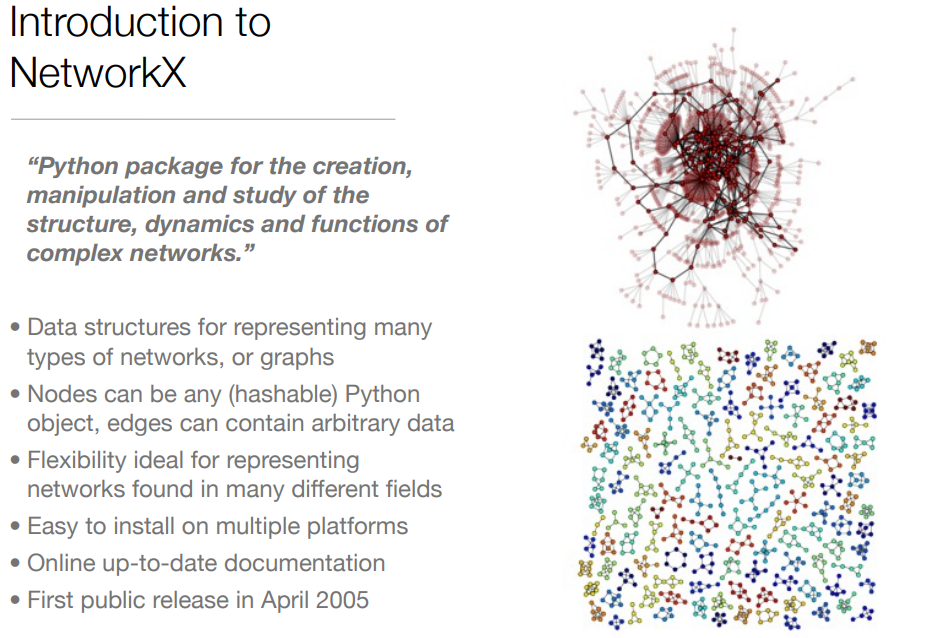
对于企业的支付网络而言，道理是类似的，但企业支付网络的违约概率分析可能会更加复杂。企业支付网络中流通的金额通常会更大，并且资金的链条通常会非常的长，这种复杂的交织网络便产生了系统性风险。总体风险和系统性风险的大小取决于两个因素：参与方主体的风险和个体之间相互作用网络的拓扑结构。对于前者比较好理解，如果交易合约的价值越大，风险敞口也就越大，违约的概率也会相应有所提高；对于后者，企业公司不是孤立存在的，于外界有着频繁的交流互动，包括供应链、支付、业务合作、金融合同和共同所有权等等而产生的交互，相互作用的结构是复杂而多样的。比如，即使一家公司的风险不是所有同行都知道，它可能会影响它与其他公司互动的能力。较差的评级（即高风险）可能会阻止其获得信贷，从而可能导致对供应商的付款减少或延迟。如果付款方有高风险，未付款或延迟付款会阻止公司自身对外的其他付款，从而增加一连串未付款和蔓延财务困境的可能性。

在这样的背景下，我们想要了解公司在网络中的角色以及在何种度量中可以提供其风险的相关信息以揭示违约概率，基于节点的网络属性和风险之间存在的相关性或许可以帮助我们在一定程度上解决这个问题，可以初步展示或改进风险评估，即违约概率。

**1.2 Networkx：图操作和分析**

Networkx是目前最流行的用于操作和分析图形的Python包之一，内置常用的图与复杂网络分析算法，可以方便的进行复杂网络数据分析、仿真建模等工作。

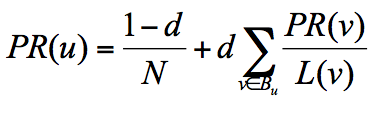
利用Networkx，我们可以描述现实生活中的许多事物，当然也包括企业支付网络中的各个企业。接下来，我们将展示如何使用节点的网络属性来预测企业支付网络中的违约概率和相应风险。



**1.3 图网络的特征：PageRank**

PageRank通常被用来度量社交影响力，只要是有网络的地方就存在出链和入链，这样便可以计算出一个用来度量某个人或某个网站影响力的PR值。但PageRank算法本身存在一定的缺陷：比如一些存在时间较久的网页通常会有较高的评分；一些网站可以手动创建很多网页以提高排名；同时可能出现等级泄露（Rank Leak）和等级沉没（Rank Sink）的现象，即一个节点没有出链或者没有入链从而导致PR值为0。

拉里·佩奇提出了 PageRank 的随机浏览模型，一定程度上解决了PageRank算法存在的一些问题。他定义了阻尼因子d，这个因子代表了用户按照跳转链接来上网的概率，通常可以取一个固定值0.85，而 1-d=0.15 则代表了用户不是通过跳转链接的方式来访问网页的，代表一个随机的过程。

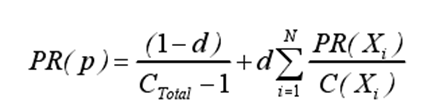


其中 N 为网页总数，加入阻尼因子 d，一定程度上解决了等级泄露和等级沉没的问题。通过数学定理也可以证明，最终 PageRank 随机浏览模型是可以收敛的，也就是可以得到一个稳定正常的 PR 值。对于企业支付网络中的企业节点，同样具有类似的特性。因此我们决定采用PageRank作为我们进行企业支付网络中违约概率估计的初步方法。

根据PageRank算法的原理，对应到企业支付场景下， PageRank算法的两个基本假设为：

1. 数量假设：如果一个公司A与另一个公司B有交易，如果A违约了，也就是说B此时面临收不回钱的局面，A贡献了“PR值”。相应地，B如果收不到钱，对与B有交易协议的其他企业而言B的违约的可能性也就可能增加。因此，如果这种“A公司”较多，可以看作B也是一个很有可能违约的企业，所以B的PageRank值会比较高，相应的违约概率也会比较高。
2. 质量假设：紧接着上面的例子，一家高PageRank值的公司B与公司C有交易，那么相应地，C的PageRank值也会比较高，可以看作是风险聚集的一种表现，即风险不会在网络上均匀传播，而是集中在特定的地区。

对应于企业支付网络，算法的公式形式不变，每个变量对应的含义有所变化：



其中p代表某个待评估违约概率的公司，d是阻尼系数（常数）；是企业支付网络中的企业总数；N表示有N家企业与p有交易；表示第i个与该企业进行交易的企业；表示这家企业总的交易企业数量。

**1.4 PageRank的改进：TrustRank**

尽管PageRank给我们提供了一种衡量企业违约概率的方法，但PageRank算法存在一些缺点：未考虑交易金额大小以及交易频率对违约概率的影响，比如有的公司可能违约概率很小，但如果交易金额达到超出偿付范围，这笔交易也是很有可能违约的。相比大金额的交易，小金额的多笔交易显得不那么重要；同时，PageRank算法仅考虑的数量层面的影响，未考虑企业主体本身的“质量”，如果一个企业评级较低，那么它违约的可能性也会更高，因此应该将这种企业好坏考虑进来，赋予更高的违约概率。

TrustRank作为PageRank的改进或者说“替代品”，已经成为比PageRank更具有说服力的打分工具。TrustRank基于假设“好的网站很少会链接到坏的网站，反之则不成立”，在PageRank的基础上引入了好与坏的概念。当网站为更可靠、更值得信赖的信息来源时，网站的排名才会更高。

企业支付网络同样满足这个假设，在满足一定的背景调查前提下，信用好的企业通常不会和信用差的企业做交易，信用好的企业更多地会主动去寻找信用同样好或比其更好的企业去做交易；反之，信用差的企业会想要去和信用好、可靠新高、权威性高的企业去做交易，试图提高自身的信用。

因此，基于这一假设，如果能够挑选出百分之百信任的公司或者说百分之百不信任的公司，我们便能评估与这些公司有交易的公司的信用，从而得到违约概率，因为与百分百信任的公司进行交易的公司的违约概率也会很小。在本文中，我们采用TrustRank值来表示企业可能违约的概率，也就是说，如果一个企业被评估为“坏公司”，违约的概率很高，那么这个企业的TrustRank也就越高，与之进行交易的企业的TrustRank相应也会越高。

TrustRank先用人工去识别高质量的页面(即“种子”页面)，由“种子”页面指向的页面也可能是高质量页面，即其TrustRank也高，与“种子”页面的链接越远，页面的TrustRank越低。于是我们先选出有违约行为，区分成“坏企业”和”好企业“的企业集，与坏企业交易的也有很大可能是坏企业，那么其TrustRank也高。与坏企业的链接越远，企业的TrustRank越低。

根据TrustRank算法，我们能够发现风险的同质性，即企业倾向于与具有相似风险的企业相互作用。这是一个两个节点间相关联的属性，但是在更大的聚集尺度上也可以更清楚地观察到类似的行为。通过使用不同的方法检测到的企业群体，往往显示出具有统计学意义的大量特定风险类别的企业，这表明具有相似评级的企业倾向于通过支付联系在一起。这意味着，对单个公司的特殊冲击可以或多或少地传播，这取决于当地的网络结构和节点所属的社区。

# 假设与改进

**2.1 问题分析与改进**

① 传统的PageRank算法由于迭代过程复杂、时效性不强、且执行速度慢，从而导致存在以下三个弊端：（1）旧页面将有较高的排名，而新页面将有较低的排名；（2）手动创建大量提升页面以提高目标页面的页面排名；（3）排名是基于链接的数量和质量，而质量已经过验证。因此我们在PageRank的基础上，根据拉里·佩奇提出了的PageRank 的随机浏览模型对算法进行了优化，通过增加阻尼因子避免了PageRank值为0的情况。

② 相比于PageRank，TrustRank更适合我们的分析场景，因为TrustRank引入了好网站和坏网站之分，如同公司交易网络中存在违约和不违约的对象，违约对象流入的交易的违约风险更大，即TrustRank越高，因此使用变量default作为类似区分好坏网站的标准，更高的TrustRank意味着更高的违约概率。传统的PageRank的算法是没有考虑到一个连接导向的不同性问题（也就是每一笔交易金额不同，重要性不同），因此引入交易金额是提升PageRank违约概率算法的关键。引入交易金额后，数据集中，不论是内部数据究竟是不是很全不重要，因为只要我们拿到了最核心的数据即可，相比费劲心事去考虑很多外部的难以统计的小额的流入，了解并控制大额的核心的交易违约概率已经足够了

③ 对于未知的汇入与汇出，我们认为并不需要去考虑流入流出对象缺失问题，如果无法知道交易流入对象，我们只考虑交易金额和他交易对手是谁即可，若交易中交易双方都没有信息，那该笔交易违约与否没有意义，因为即使违约概率很大，我们也无法进行对应的风控，我们不知道风险来源在哪。

④ 在违约概率的计算中，我们考虑了付款金额的占比，因为我们认为，违约的概率和影响程度会受到交易金额占比不同的影响。

⑤ 对于集团客户，即集团内不同公司的交易，我们认为需要考虑法律主体是否一致。若法律主体一致，则不需要考虑集团间的交易，违约发生对于集团内的各个部分的影响都是一样的。但如果法律主体不一致，则需要分开考虑，将其视为一个独立的公司节点。

⑥ 对于应该使用多长的时间周期的数据来建立网络，我们认为需要根据不同企业的信用分级来判断；一般来说一年以上三年以下的数据最佳。

⑦ 我们选择使用有向图，因为无向图不符合真实的交易情景。对于资金流出方来说，合约价值为负，不具备风险；对于资金流入方来说，合约价值为正，才会有交易对手违约带来的风险。

**2.2 模拟数据**

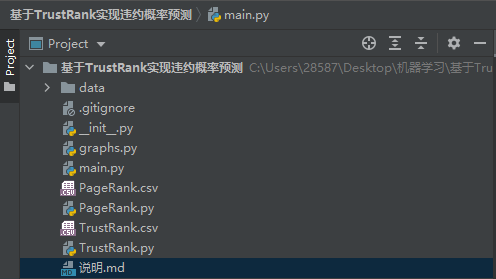
由于缺乏真实的交易数据，因此我们使用近似的方法模拟未知的付款方和收款方，并在评估违约概率时考虑支付额占比或支付频率占比。我们根据已有的节点数据，对每一条节点交易记录都增加了随机生成的交易金额以及是否违约的标记。

对于交易金额，我们使用numpy中的random.randint方法随机生成了在1-10000之间的数据，表示企业间的交易金额在1万元至10000万元之间；

对于是否违约，我们根据2019年非金融企业债务融资工具总样本静态池首年的边际违约概率为0.62%，随机选取了样本中0.62%的交易记录并标记为违约。

# 估计违约概率

**3.1 项目结构**



**main.py：**包含运行函数，执行排名功能

**graphs.py：**有两个类: `getGraph` 和 `plotGraph`

**getGraph:** 从graph file获取输入, graph file 包含图的边edges

**plotGraph:** 执行可视化,绘制网络图并显示它如何随着算法的进行而变化

**PageRank.py：**PageRank算法类

**TrustRank.py：**TrustRank算法类

**data：**

**traderandom.txt：**

通过模拟得到的交易数据

第一列：节点1

第二列：节点2

第三列：交易金额

第四列：是否违约

**consumer\_tran\_new.csv：**在traderandom的基础上加入了时间维度

**test：**用于测试

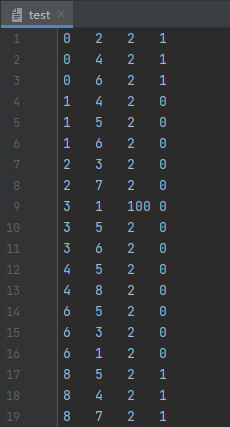
**result：**

**PageRank.csv：**运行模拟数据结果

**TrustRank.csv：**运行模拟数据结果

PageRank.csv 和 TrustRank.csv分别为两个算法的运行结果，包含500多万条评分结果

**3.2 演示说明**



由于模拟出来的数据集过大（500万条），这里我们采用一个test来说明我们的算法过程和演示结果，其中

第一列：节点1，

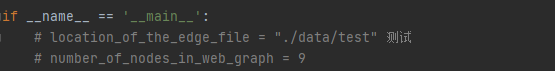
第二列：节点2，

第三列：交易金额，

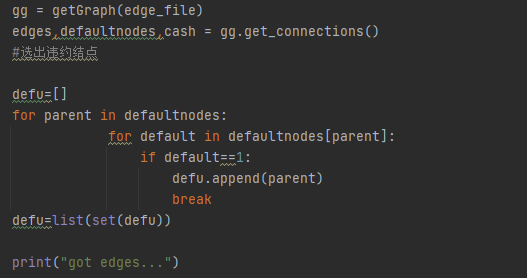
第四列：是否违约。

这里0节点和8节点是违约过的。这里违约的含义是指，如果一家企业有一笔违约，那么该企业所有需要进行支付的交易都会违约。

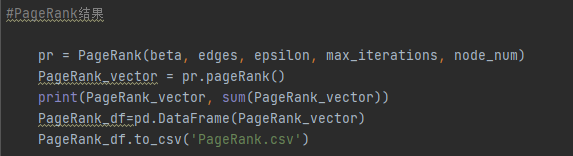




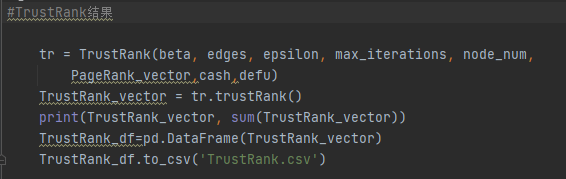
通过main主程序入口运行，通过defu选出违约节点：



调用PageRank算法计算PR值保存在PageRank\_vector中，另外计算sum(PageRank)，判断是否为1，因为一开始权重是通过等权的方式分配的（也就是1/node个数）:



类似地，调用TrustRank算法：



test得到的结果展示如下，分别对应每个节点rank的分数，其中分数越高，对应的企业的违约概率也越高：

**PageRank：**

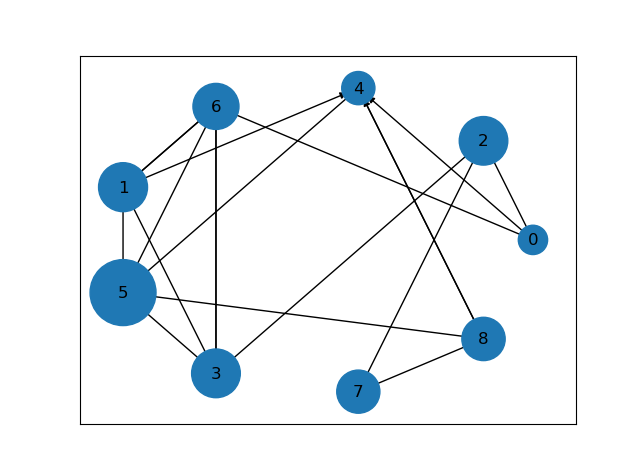


**TrustRank：**

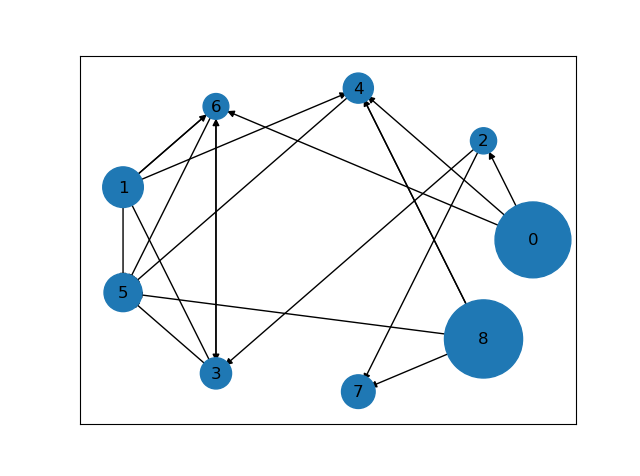


通过得分可以判断相对违约概率，可以看到最后的sum=1.0，这也是为什么算出的违约概率是相对的。

**PageRank的初始图：**



**TrustRank的结果图：**

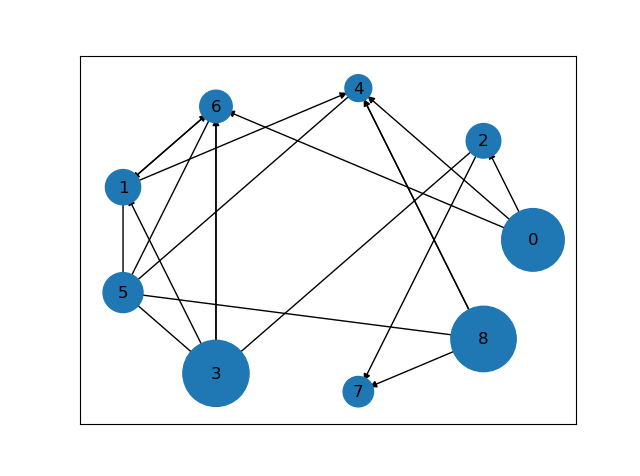


**可以看到，TrustRank结果成功显示出了企业的相应的违约概率，由于节点0和节点1均有过违约行为，因此节点更大，对应的TrustRank的值也更大。**而PageRank的模拟结果并不能正确反映出设想的结果，表示反而是5节点的违约概率最大，而这跟与5进行交易的公司数量最多有着较大的关系。

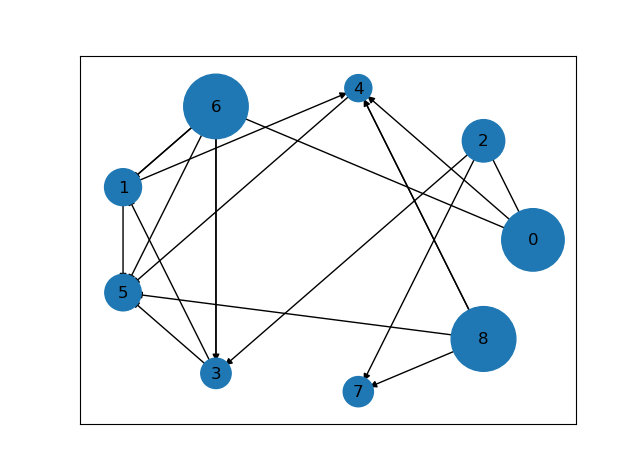
由于500万数据的图没办法画出来，但是能够得到最后的运行结果，保存在了PageRank.csv和TrustRank.csv。

接着，我们可以分别给3节点和6节点分配一个违约状态，得到如下TrustRank结果：

给3一个违约状态，可以看到3节点估计的违约概率明显增大了。



而给6一个违约状态时，可以看到6的违约概率明显增大了。



**3.3 更多打算**

在测试过程中我们发现由于模拟出的数据集过大，读取和运行的速度都比较慢，打算尝试使用mapreduce进行性能提升。Map每个节点计算流入的PageRank，通过reduce汇总，达到提升效率的目的。

对于企业子网的研究，我们希望获取更多真实可靠数据以便进行进一步的模型训练和训练结果反馈

同时进行更多的测试以及对算法进行改进，希望得出更准确地、更直观地描述方式。

P.S.详细的代码和说明在附件中。

# 参考文献

[1] Chong C , C Klüppelberg. Contagion in financial systems: A Bayesian network approach[J]. Papers, 2017.

[2] Letizia, E., Lillo, F. Corporate payments networks and credit risk rating. EPJ Data Sci. 8, 21 (2019). <https://doi.org/10.1140/epjds/s13688-019-0197-5>

[3] Detering N , Meyer-Brandis T , Panagiotou K , et al. Managing Default Contagion in Inhomogeneous Financial Networks[J]. Papers, 2016.