



论文狗论文检测报告 www.lunwengo.net

报告编号：1034887683144D32860F860345B1A1CD

标题：lunwen

论文作者：Russ

文档字数：33155

时间：2018-05-09 14:14:28

检测范围：学术期刊、学位论文、会议论文、中国高等教育文献保障库

一、检测结果：

总相似比：29.27% [复写率与引用率之和]

检测指标：自写率 70.73% 复写率 29.27% 引用率 0.0%

相似比：中国高等教育文献保障库 0.0% 学术期刊 0.0% 学位论文 0.0%

二、全文相似详情：（红色字体为相似片段、浅蓝色字体为引用片段、深蓝色字体为可能遗漏的但被系统识别到与参考文献列表对应的引用片段、黑色字体为自写片段）

摘要

21世纪是计算机行业与金融行业蓬勃发展的黄金时代，我国的债券交易产业也在进入21世纪之后发展得如火如荼，然而随着金融产业的发展，债券交易市场越来越具有复杂性，许多不可预知的风险威胁着债券交易市场，传统的金融知识以及金融工具已经很难再保障债券交易的安全性，这就需要使用新的工具、新的方法来为债券交易进行保驾护航众所周知金融知识中运用了许多数理统计、线性代数与概率论知识，计算的过程中还需要采集大量的历史数据进行分析推断，人工计算往往会事半功半，但是计算机与大数据概念的出现，使得金融问题的解决有了更高效方案该毕业设计主要解决了以下一些问题：

使用Python爬虫获取债券交易数据，实现了用户自定义收益率数据的收集

通过PCA主成分分析方法研究债券标准期限结构，提取出了3个主成分，计算出了三个主因子对固定收益产品价值波动的贡献度，并分析了这些主因子对债券期限结构的影响

通过蒙特卡洛模拟法计算了给定债券的在险价值（风险价值），使得用户对于债券的投资风险有了直观的概念

结合插值法完成了基于Nelson-Siegel模型的收益率曲线拟合功能，使得用户能够拟合自定义的收益率曲线

通过Web前端技术结合Node Express代理和MongoDB数据库实现软件开发，为用户提供数据展示、数据查询、曲线拟合、PCA分析、VaR计算等功能

关键词：收益率曲线Nelson-SiegelPCA数据采集VaR Node ExpressMongoDB

Abstract

The 21st century is the golden age for the booming development of the computer industry and financial industry China's bond trading industry is also developing rapidly in the 21st century However, with the development of the financial industry, the bond trading market is increasingly complex and unpredictable The risk poses a threat to the bond trading market Traditional financial knowledge and financial instruments can no longer guarantee the security of bond trading This requires to use new tools and new methods to protect bond transactions It is well known that many mathematical statistics, linear algebra and probability theory are used in financial knowledge, and a large amount of historical data needs to be collected in the calculation process for analysis and inference Manual calculation using these methods often requires a lot of manpower and material resources, But with the emergence of computer and the concept of big data, a more efficient solution has been made to the solution of financial problems This graduation design mainly solves the following problems:

Using Python spiders to obtain bond transaction data, realized the collection of user-defined yield data

Using the principal component analysis(PCA) method to study the bond term structure, extracting three principal components, calculating the contribution of the three main factors to the value fluctuation of fixed income products, and analyzing the term structure of bonds with these principal factors ' impact

Calculate the value at risk (VaR) of a given bond through the Monte-Carlo simulation method, so that the user has an intuitive concept of the investment risk of the bond

The Nelson-Siegel model-based yield curve fitting function is implemented by the interpolation method, so that the user can fit a customized yield curve

Software development through Web front-end technology combined with Node Express agent and MongoDB database to provide users with data display, data query, curve fitting, PCA analysis, VaR calculation and other functions

Key words: yield curve Nelson-Siegel PCA data acquisition VaR Node Express MongoDB

第1章 绪论

贵阳高登世德金融科技公司是一家中国资产证券化行业内顶尖的金融科技公司，主要负责为金融机构设计资产证券化产品以及优化证券化产品的评级优化，而在这家公司的实习促进了这个毕业设计的诞生，公司的客户需要一款能够实现债券数据采集、分析、期限结构影响因子挖掘以及在险价值计算的软件，所以这个毕业设计的主要目的就是设计一款符合客户需求的债券分析软件

1.1 研究背景

在停止发行国债22年之后，1981年我国重新开始发行国债，从1981年以来，中国政府债券市场已经经历了近三十载的发展在这近三十年的发展过程中，政府债券规模不断扩大，政府债券品种不断完善，政府债券市场也从无到有，建立了完整而健全的政府债券市场体系但与此同时，中国国债市场还具有许多的不足，比如立法基础薄弱、市场分割、投资者结构不合理、市场流动性不足等等

我国于1984年首次发行企业债券，到现在为止，经历了萌芽阶段（1984-1986）、快速发展阶段（1987-1992）、整顿阶段（1993-1995）和规范发展阶段（1996至今）目前而言，我国上市公司债券规模还不够大，相比较于股票的发行远远落后，但是从1984年发行企业债券以来，我国企业债券发行总量有了大幅度增长，发行主体也越来

越呈现出多元化的趋势，企业债券品种越来越丰富，其作为投资工具的地位也逐年上升随着对企业债券的限制越来越少，企业债券发行节奏明显加快发行债券的品种和主体都越来越丰富

伴随着我国国债和企业债市场的飞速发展，诞生了一些优秀的债券数据分析管理企业，比较著名的比如中债登、万得、和讯等企业，这些企业积累了大量的中国债券市场交易数据，比如收益率数据、估值数据、VaR数据、成交指数等等，这些企业还拥有了比较丰富的国内债券数据处理经验和数据，比如中债登采用独特的Hermite插值法拟合中国国债收益率曲线，采用历史模拟法实现VaR的计算通过这些优秀的企业，可以获取到许多宝贵的数据以及数据处理方法，进而可以对债券进行评估定价和风险预测，达到毕业设计的目的

1.2 研究目的和意义

在购买债券时，消费者应该根据什么指标来判断一支债券的价值呢债券购买者往往通过收益率曲线来判断，研究债券期限结构具有重要的意义,债券投资者根据它来预测债券的发行投标利率债券发行人以它作为其发行债券、进行风险管理的参考，所以收益率曲线以及收益率数据的收集与研究就显得十分有价值

本设计通过从各大证券交易所收集收益率数据并且对这些数据进行曲线拟合、主因子提炼、分析主要因子的影响以及判断未来曲线的走势预测债券购买价值并给出投资策略

本设计的意义：通过对已有数据的分析挖掘，提炼出曲线以及方程，根据PCA方法分析主要影响因子，根据这些因子可以给出未来的债券投资策略，达到预测债券走势的目的

虽然该毕业设计主要虽然研究的是固定收益债券定价和投资策略分析，但是主要研究对象还是收益率曲线要了解设计的目的，首先要了解收益率曲线收益率曲线是根据某支债券的标准期限数据所绘制的图形，它反映市场当前的收益率水平

1.3 国内外研究现状

在中国，由于包括国债市场在内的债券市场发展时间非常短，流动性和市场化程度尚需进一步加强，有关利率期限结构和国债收益率曲线的研究也相对较薄弱，这增加了对国债收益率曲线进行研究的必要性和迫切性国内最早的关于收益率曲线的公开研究文献是杨大楷和杨勇姚长辉和梁跃军

由于中国债券市场的发展起步较晚，对中国债券市场的理论研究落后，收益率曲线研究始于90年代后期近年来值得一提的工作是应用国外广泛使用的模型对我国债券收益率曲线进行实证研究和比较分析研究表明，针对中国的现状短期债券发行量小，长期性差，中期债券比例大，不健康的期限结构，Nelson-Siegel模型构建出来的中国国债收益率曲线比较切合实际应当利用中国债券市场数据，利用国际上公认的几种收益率曲线拟合模型进行实证研究，寻找适合中国债券市场的拟合收益率曲线的模型

Diebold和Li（2006）、Diebold、Rudebusch、Aruoba（2006）、Diebold Piazzesi和Rudebusch（2005）已经发现，在美国的封闭经济环境中，广义的Nelson-Siegel模型精确地预估了收益率曲线的走势，预测结果较为切合实际

Diebold-Li分解单个国家的Nelson-Siegel收益率曲线是：

$$y_i() = l_i + s_i((1 - e^{-i})/i) + c_i((1 - e^{-i})/i - e^{-i}) + v_i() \quad (1.1)$$

其中 $y_{it}()$ 表示月债券的连续复利的零息票名义收益率， l_i 、 s_i 、 c_i 、 v_i 均为待定参数，并且具有标准偏差的干扰项 l_i 跟Diebold之后，通过允许参数随时间变化来动态化模型：

$$y_{it}() = l_{it} + s_{it}((1 - e^{-i_{it}})/i_{it}) + c_{it}((1 - e^{-i_{it}})/i_{it} - e^{-i_{it}}) + v_{it}() \quad (1.2)$$

式中 y_{it} 、 l_{it} 、 c_{it} 解释为为潜在因素，特别是，如 l_i 跟Diebold所发现，它们分别是水平参数，斜率参数和曲率参数，因为这些因子参数是常数，所以这是一个关于的递减函数和凹函数由于收益率因子随时间而变化，这种Nelson-Siegel模型能够生成各种随时间变化的收益率曲线形状

1.4 研究内容

该论文主要描述一款基于Web前端页面、Node Express后台接口、Python数据采集分析脚本、PCA算法、Monte Carlo算法以及Mongo DB数据库的债券数据采集分析软件的开发过程这款软件实现了用户自定义债券收益率数据的查询、自定义债券收益率曲线的拟合、自定义债券期限结构数据的PCA分析以及自定义债券的在险价值计算

这款软件的核心在于后台Python脚本程序，通过Python脚本程序，实现了债券数据采集和过滤、债券数据的PCA分析、债券收益率曲线的拟合以及债券在险价值的计算，这一部分算法以及数据都是整个软件的核心通过Python程序采集到数据后并进行过滤整理后，将所有数据分类保存在MongoDB数据库中，以便之后Node脚本程序和Python程序对数据进行查询调用在前端部分，采用了目前最为流行的MVC框架——React进行开发，负责呈现软件的用户交互界面，包括数据展示、收益率曲线展示、用户输入界面等等后台部分，使用当前比较流行的Nodejs以及Nodejs构建的Express框架作为后台代理，实现了RESTful风格的后台接口开发以及服务器搭建另外通过npm模块实现了Node脚本程序与Python脚本程序之间的交互，使得Node服务器可以调用Python程序进行数据处理

1.5 论文基本结构

这篇论文的正文整体上分为6章，按照总-分-总的架构讲述整个软件设计过程中的思路、原理、算法、心得等方面

第一章为绪论，主要讲述了这个毕业设计的需求来源、设计研究的背景、目的意义、当前国内外现状、设计的研究内容以及论文的大致结构

第二章主要讲述整个软件的底层算法，因为这些算法是整个软件的灵魂，所有的逻辑业务都基于这些算法展开，所以把算法的讲述放在第二章，主要包括PCA主成分分析法、Nelson-Siegel模型、基于Monte Carlo模拟法的VaR计算法

第三章主要讲述整个软件的总体设计方案，包括总体开发流程、总体架构设计、总体解决方案以及总体功能设计

第四章主要讲述服务端（后台）软件设计，包括Python脚本程序设计、Node脚本程序设计、Node Express服务器的搭建、API接口设计和后台数据库结构设计

第五章主要讲述前端Web软件设计，包括界面设计、模块的选用、路由设计方案和页面效果的展示

第六章主要讲述整个软件开发过程中得到的结论，以及对于未来工作的展望

第2章 软件算法研究

2.1 PCA算法

算法介绍

PCA是一种常用的高维数据降维的方法，能够产生一个因子载荷矩阵，当数据在载荷矩阵上投影时能够有效的识别数据之间的相似性和主要的差异此外，PCA的一个最大的优点是贡献度越大的成分保留的原数据信息越多，根据这个特征，可以在信息损失量最小的情况下对高维数据进行降维处理，减少数据量从数学的角度来解释PCA的原理，主成分分析就是根据协方差矩阵所反映的各维度相关性，对相关性较高的维度进行合并提取，使得在信息损失较小的情况下，提取出来的数据能够尽可能的代表原数据，即使其方差最大根据以上结论，对某一组数据进行主成分分析，就是求它归一化之后矩阵的协方差矩阵的各特征值及相应的正交单位化特征向量将特征值按降序排列，选出对应的特征向量，组合成一个新的矩阵，矩阵的原变量的线性组合即为该数据的第一、第二、主成分，各主成分的方差等于相应的特征值

假设有一组原始数据如下：

$$X=(X_{ij})_{(n \times p)} \quad i=1,2,\dots,n; j=1,2,\dots,p \quad (2.1)$$

其中n为数据条数，p为样本维度数目对它进行主成分分析的具体步骤如下：

第一步：元数据归一化，公式如下：

$$Z_{ij}=(x_{ij}-x_{.j})/S_{.j} \quad i=1,2,n; j=1,2,\dots,p \quad (2.2)$$

计算元数据矩阵的每一列的平均值，然后对于元数据矩阵的每一个元素都减去列平均值，这样做的目的是避免数据量纲的影响

第二步：计算数据的协方差矩阵：

公式如下：

$$R=(r_{ij})_{p \times p} \quad (2.3)$$

其中：

$$r_{ij}= 1/(n-1) \sum_{k=1}^n Z_{ki} Z_{kj} \quad i=1,2,n; j=1,2,\dots,p \quad (2.4)$$

第三步：计算协方差矩阵的特征根、特征向量,并将特征根按降序排列：

R的特征值 λ_i ($i=1,2,p$)，使其按大到小的顺序排列，即为主成分的方差序列，特征值的大小反映了其对于原始样本的贡献度，或者说是权重每个特征值对应的特征向量为 L_1, L_2, \dots, L_p 通过特征向量将标准化的指标转化为主成分：

$$F_g=Z \times L_g \quad g=1,2,\dots,p \quad (2.5)$$

F为主成分，共有p个。

第四步：选择贡献度靠前的几个主成分作为新的基

通常选择主成分个数的标准是累计贡献度达到85%之上，由于债券期限结构数据的影响因子已经基本有了定论，所以该设计中选择提取3个主要的因子

第五步：把原数据在新的基下进行投影

将得到的因子载荷矩阵作为权重，对归一化后数据的每一列加权求和即可得到主成分综合得分

PCA研究中国国债

通过Python爬虫，从中债网上爬取了2002年到2017年全年度每日债券收益率详细数据，存储在16个Excel表格当中，共有数据大概75000余条，但是这些数据年度跨度较大，利率期限结构发生了变化，从最初的19个标准期限（2002年～2009年），变化到了21个标准期限（2010年～2015年），然后变化到了16个标准期限（2016年～2017年）为了统一维度，我选择了16年当中共有的14个标准期限，这些期限包括0年, 0.08年, 0.17年, 0.25年, 0.5年, 0.75年, 1年, 3年, 5年, 7年, 10年, 15年, 20年, 30年通过Python程序对这16年的数据进行筛选，共筛选出3996条标准期限数据，这些数据包含前文所提的14个维度

通过SPSS软件和Python程序的两次分析，确保得到了正确的分析结果，结果包括提取出的所有主因子、各维度相关系数矩阵、总方差解释（贡献度）、碎石图、成分

矩阵以及各指标对应系数

因为主要目的是主成分分析，所以最关心的应该是贡献度详细（总方差解释），下面即为详细：

图2.1 中国国债数据各维度累计贡献度数据

查看初始特征值的累计项，可以看到第一个主因子解释了期限结构77.875%的变化，前两个主因子解释了期限结构91.803%的变化，根据数据挖掘PCA的基本原则，超过85%的贡献率即可提取主因子，但是为了尽可能的逼近事实，解释足够多的期限结构变化，我选择前三个主因子，前三个主因子的解释度达到了96.071%，即可以认为前三个主因子可以解释期限结构96.071%的变化

前三个主因子初始特征值及贡献度：

表2.1 PCA分析因子累计贡献度

初始特征值贡献度（%） 累计贡献度（%）

5.07477.87577.875

0.90713.92891.803

0.2784.26896.071

不难发现，第一主因子对于利率期限结构影响最大，第二主因子较小，第三主因子影响最小

经过SPSS软件的分析，得到成分矩阵：

图2.2 PCA分析中国国债成分矩阵

根据主成分分析理论，我们可以得出各指标对应系数等于成分矩阵对应项除以初始特征值的平方根，即：

指标对应系数= 成分矩阵对应项/初始特征值(2.6)

但是通过Python程序，可以很快的得到各项指标对应系数：

图2.3 PCA分析中国国债因子载荷矩阵

由于数据量比较大，可能这样的表还不太直观，下方展示两张Excel绘制的图标，能够更加直观的展示三大主因子对于利率期限结构的影响，柱状图：

图2.4 三大主因子对利率期限结构的影响

折线图：

图2.5 三大主因子对利率期限结构的影响

图中各点代表的是各指标对应系数，所以当系数大于0时，对指标是正影响当系数小于0时，对指标是负影响通过上方的图表，不难发现第一主因子对于14期标准期限的收益率而言都是正值，并且全部为了0.3左右，相当稳定，所以第一主因子对于所有期限收益具有相同方向的影响，可以理解为影响收益率曲线的平移第二主因子对于早期期限收益而言，是负影响，而对于长期收益而言，是正影响，所以第二主因子影响了期限结构的斜率第三主因子对于早期和长期收益而言，均为负影响，对于中期

收益却是正影响，它的变化使得早期和长期收益朝同方向发展，而中期收益朝相反方向发展，所以第三主因子影响了收益率曲线的曲度

以上研究与国外的研讨结论较为统一，根据国外研究，以上三个主因子分别可称为水平因子、斜率因子和曲率因子

根据贡献度的数据，我们可以认为水平因子（及第一主因子）对债券收益率曲线走势的影响最大，对收益率曲线的走势起主导性作用，斜率因子对收益率曲线走势的影响次之，曲率因子对于收益率曲线走势的影响最小但是对于不同的期限，三个因子的影响也有所不一，短期债券只对于水平因子敏感，长期债券对斜率因子、曲率因子都敏感

PCA综合研究与结论

除了使用PCA算法研究中国国债期限结构之外，还使用PCA算法研究了美国国债（2006~2018年）、SHIBOR（2006~2018年）、LIBOR（2001~2013年）的期限结构，综合以上所有的研究结果进行横向对比，得到了一下结论：

中国国债和美国国债都可以通过三个主成分解释，并且可以解释收益率曲线绝大部分的变化三个主成分可以解释中国国债96.07%的变化，解释美国国债99.78%的变化这三个主成分分别可以归纳为水平因子、斜率因子和曲率因子其中，水平因子影响了收益率曲线的平移，斜率因子影响了收益率曲线倾斜度的变化，曲率因子主要影响收益率曲线的曲度

图2.6 中国国债PCA碎石图

图2.7 美国国债PCA碎石图

在对美国国债的研究中，第一主成分（水平因子）可以解释92.75%的收益率曲线变化，但是中国国债的第一主成分（水平因子）只能解释77.87%的收益率曲线变化通过后续对SHIBOR以及LIBOR的研究，可以发现这可能是由于中国的基准利率规范不够市场化，国债购买者通过国债收益率曲线获取到的信息不足，所以导致资产定价不够稳定

中国国债的第二、第三主成分对收益率曲线走势的影响远大于美国国债，导致收益率曲线的变化更加复杂，造成这一情况的原因可能是由于中国债券市场还不够成熟，可供购买者选择的产品不多，所以市场上存在许多炒作和投机现象，使得收益率曲线的变化更具有复杂性

中国国债的第四个主成分还有着2.12%的贡献率，美国国债的第四个主成分贡献度却只有0.12%，这说明相对于美国债券市场，中国国债未被分解出的成分更多，这使得之后的预测过程相对于美国国债而言会出现更多的未知因素，中国债券市场的波动性也会更大

图2.8 美国国债各期限收益率曲线图

图2.9 中国国债各期限收益率曲线图

上面的两张图是中国国债以及美国国债的不同到期期限债券的日收益率，美国国债有3061个数据点（3061天的日收益率数据），中国国债包含有3900余条日收益率数据从上面的两张图我们可以发现美国国债多年（2006~2018年）以来收益率一直维持在一个小的范围内，变化相比于中国国债而言比较简单而且少，中国国债在这段时间内（2002~2017年）收益率数据变化次数比较多，并且变化比较复杂此外我们可以发现美国国债收益率曲线中的多条曲线（即多个维度）的走势具有高度相似性，可以理解为形状大致相同，所以多个维度之间的相关系数会比较大，提取出的主成分中第一主成分贡献度会比较高，这两个结论符合我们之前所做的推断

SHIBOR与中美国债类似，也可以由主成分分析法（PCA）提取出3个主成分，这三个主成分的解释度可以达到97.89%，第一主成分贡献度就达到了84.71%，可以说SHIBOR的主成分构成有点类似美国国债了，因为影响的因素比较少，所以SHIBOR的曲线相对而言会比较平稳，变化不会十分复杂

图2.10 SHIBOR收益率曲线

对LIBOR进行主成分分析可以发现其第一主成分贡献度达到了惊人的98.85%，第二主成分就只有0.871%的贡献度了，即通过一个主成分就可以比较完善地解释LIBOR收益率曲线的变化了

图2.11 LIBOR收益率曲线

通过图2.11，我们可以发现LIBOR收益率数据的多个维度之间具有高度的相关性，走势几乎是完全一致的，只不过维度之间具有垂直方向的差异，即不同期限债券收益不同，这几乎完全是由水平因子导致的，所以LIBOR的第一主成分（水平因子）贡献度会达到惊人的98.85%

图2.12 LIBOR收益率数据相关系数矩阵

图2.12是LIBOR收益率数据的相关系数矩阵，其中的每一项都代表了LIBOR收益率数据两个维度之间的相关系数，可以看到在图2.12中，几乎每一项都十分接近于1，所以LIBOR收益率数据的各个维度具有高度一致性

2.2 Nelson-Siegel模型

曲线拟合的概念及目的

通常，债券购买者得到的期限结构数据是二维空间中一系列离散的点，但是为了更加直观的观察债券收益走势，更加合理的利用期限结构获取更多有用的信息，通常会利用这些离散的点构造一条平滑的曲线，即为收益率曲线，利用离散的点构造这条收益率曲线即为收益率曲线拟合

比如，我们从中债网上可以获取到2018年4月13号当日的债券收益日信息，填写在表格中，即：

表2.2 拟合国债数据表

Maturity(Year)	Yield(%)
0.02	2.316
0.082	5.617
0.172	5.693
0.252	6.842
0.53	0.242
0.753	1.843
1.03	2.013
3.03	3.682

5.03.5205

7.03.6916

10.03.7062

15.03.9689

20.04.0079

30.04.1207

40.04.162

50.04.1808

使用Excel描绘散点图：

图2.13 期限结构散点图

由此可以得到一个期限结构数据描绘的散点图，但是这张散点图不够直观，如果债券购买者想要获得到期期限为35个月的债券的收益率数据时，将无从下手，所以需要根据这些二维空间中的散点，构造一条光滑曲线，这条曲线并不只是简单的将散点用光滑曲线连接，还需要对未给出的节点做出价格推测这就是曲线拟合以及曲线拟合的目的和意义

下图是中债网所给出的收益率曲线：

图2.14 中债网收益率曲线图

曲线拟合方法及模型的选择

为了得到一条这样的拟合曲线，最重要的是要选择合适的拟合方法和模型常见的收益率曲线拟合方法或模型有多项式插值法、Hermite模型插值法、Nelson-Siegel模型、Hull-White等等接下来会使用同一种数据TestData.csv使用其中几种模型或方法进行曲线拟合，然后对拟合精度进行评估，使用到的评估标准是均方根误差（下文使用RMSE表示）

TestData.csv数据：

表2.3 收益率曲线拟合所用数据表

02.2316
0.082.5617
0.172.5693
0.252.6842
0.53.0242
0.753.1843
13.2013
33.3682
53.5205
73.6916
103.7062
153.9689
204.0079
304.1207
404.162

多项式插值法

构造多项式拟合已有收益率数据，然后通过插值构造平滑收益率曲线的方法即为多项式插值法

多项式中拟合收益率曲线效果最好的是三次多项式，即为立方多项式可以通过Python程序进行拟合，并查看拟合效果

首先在Python中通过numpy工具进行三次多项式拟合，拟合的三次多项式系数：

表2.4 拟合三次多项式系数

三次项系数二次项系数一次项系数常数

5.77597e-05-0.005400.155392.74509

然后计算出所有的估计值，然后计算出均方根误差进行输出，下面是程序运行的结果：

图2.15 三次多项式拟合的均方根误差

说明对于TestData.csv这一组数据而言，利用三次多项式进行拟合，计算出的均方根误差是0.045

下面是拟合数据与原始数据的对比：

图2.16 三次多项式拟合效果图

从图中可以看出，三次多项式的拟合效果较差，误差较大，甚至出现了折现的情况，不符合我们需要平滑曲线的要求

Hermite插值法

Hermite插值法拟合收益率曲线在国外使用的不是很多，但是这里之所以要研究Hermite模型，是因为它比较适合于拟合中国债券市场的收益率曲线，比较有代表性的使用者就是中债网，中债网使用Hermite多段插值法进行收益率曲线拟合，具有较好的拟合效果

中债网使用的是多段插值法，具体做法是在每两个节点之间，使用三次Hermite进行插值，所以根本上而言，这还是使用的三次多项式模型，但是这种方法不仅要求节点处的值要相同，而且在节点处的倒数也要相同，所以拟合这与一般的三次多项式拟合存在不同

接下来我们仍然使用Python程序进行拟合，并计算RMSE进行效果比较，这里Python程序没有实现Hermite算法，使用的是已有的工具包——numpy通过Python程序拟合，我们得到了拟合的三阶Hermite多项式

其各项系数如下：

表2.5 三次Hermite拟合各项系数

三次项系数二次项系数一次项系数常数

7.21997e-06-1.34920e-037.77377e-022.74239

拟合后计算出的均方根误差为：

图2.17 三次Hermite拟合的均方根误差

这说明相对于TestData.csv这组数据而言，Hermite插值法拟合曲线得到的数据均方根误差为0.094，这比三次多项式拟合的效果还要差其拟合数据与原数据对比如下：

图2.18 三次Hermite拟合效果图

Nelson-Siegel插值法

Nelson-Siegel模型是金融届（尤其是债券行业）大名鼎鼎的一个模型，它是1987年Nelson C和Siegel A提出来的一个参数曲线拟合模型，它需要确定的参数少（只有四个），但是拟合的效果却很好，而且每一个参数都具有其很强的独特的金融含义

其具体公式如下：

$$y() = _L + _S \left(\frac{1 - e^{(-)} }{ } \right) + _C \left(\frac{1 - e^{(-)} }{ } \right) e^{(-)} \quad (2.7)$$

其中，_L、_S、_C是三个具有金融含义的参数是与时间有关的常数，可称为时间常数是到期期限，y是到期收益率

同样的，通过Python程序进行Nelson-Siegel模型拟合，由于有四个参数需要确定，所以会存在多组解，我们通过优化方法，使得拟合出的效果最好（以各项数据方差之和为参考标准），最终效果最好的一组参数即为我们拟合的Nelson-Siegel模型的四个参数

最终拟合的Nelson-Siegel模型的四个参数如下：

表2.6 Nelson-Siegel模型拟合参数

_L _S _C

4.34119 -1.286334.455712e-030.19455

最终计算出的RMSE为：

图2.19 Nelson-Siegel模型拟合的均方根误差

拟合效果：

图2.20 Nelson-Siegel模型拟合效果图

结合RMSE来看，三次多项式的拟合效果要好于Nelson-Siegel模型拟合的效果，但是通过曲线观察，可以发现三次多项式的拟合效果差于Nelson-Siegel模型的拟合效果，这是因为在使用Nelson-Siegel模型拟合债券期限结构的时候，由于其他因素限制，只使用了整数到期期限的数据进行拟合，所以导致其RMSE比三次多项式更低，由于这个原因，可以认为Nelson-Siegel模型的拟合效果好于三次多项式，所以该毕设还是采用Nelson-Siegel模型结合插值法进行收益率曲线的拟合

Nelson-Siegel模型的优点

采用Nelson-Siegel模型进行拟合的理由还有以下几点：

Nelson-Siegel在业界久负盛名，非常适合与拟合收益率曲线根据《中国动态Nelson-Siegel利率期限结构模型研究》一书中的统计，比利时、芬兰、法国、意大利、西班牙、等多个国家的中央银行采用Nelson-Siegel模型作为利率期限结构的估计方法还有许多其他国家采用的是基于Nelson-Siegel模型的Nelson-Siegel-Svensson模型，所以Nelson-Siegel模型的名气可见一斑

Nelson-Siegel模型构造简单，只需要计算四个参数即可完成构建

Nelson-Siegel模型的四个参数具有强烈的金融含义，所以容易该模型更理解之前说过参数是只与时间有关的常数，不同日期的收益率数据可能不相同如果我们确定了，以_L、_S、_C和为参数，我们可以做出当 = 0.0664时，_L、_S、_C三个参数的系数随发生变化的图形：

图2.21 N-S模型参数变化规律图

图2.21实际上就是影响债券收益率走势的三个主成分的因子载荷，可以看出_L对收益率曲线的影响是恒定的（水平位移），_S对收益率曲线的影响随着期限的延长而慢慢变小（曲线的斜率），_C对于收益率曲线的影响随着期限的延长先变大然后慢慢变小（曲线的曲度）根据之前对中国国债、美国国债的研究，_L代表着第一主成分，即水平因子_S代表着第二主成分，即斜率因子_C代表着第三主成分，即曲度因子所以说，Nelson-Siegel模型的参数有着强烈的金融含义

Nelson-Siegel模型的实际拟合效果

在最终的成品软件中，实现了自定义收益率曲线拟合的功能整个拟合逻辑使用Python程序实现，通过将用户上传债券的收益率数据（至少5个到期期限的到期收益率数据）传递给Python程序，得到经过Python计算之后的各期限数据，即为拟合的收益率曲线数据，通过这些数据，可以在前端页面展示出一条较为平滑且逼近真实的收益率曲线为了说明Nelson-Siegel模型的实际债券期限结构的拟合效果，接下来会使用最终的软件拟合一期实际的中国国债收益率曲线

真实的收益率曲线数据如下：

表2.7 实际收益率数据

标准期限(年)	收益率(%)
0.2	0.8352
0.082	0.9012
0.172	0.7707
0.252	0.6882
0.52	0.81

0.752.9413

12.977

33.241

53.2505

73.6601

103.6467

153.9351

203.9741

304.1113

404.1499

504.1661

该数据来源于中债网2018年5月3日的日到期收益率数据，官网所拟合的收益率曲线如下：

图2.22 中债网实际收益率曲线图

使用毕业设计所做的软件，将标准期限数据上传之后，得到了拟合的收益率曲线数据，软件所拟合的收益率曲线如下：

图2.23 软件拟合中债收益率曲线图

可以看到收益率曲线的走势是大致吻合的，下面是两者关于几个标准期限之外的到期期限对应的到期收益率的对比：

表2.8 收益率数据对比

Maturity81113182128334246

软件拟合3.613.723.783.893.954.044.094.154.17

中债拟合3.653.683.823.973.984.084.134.164.16

通过上表中的数据对比，可以发现相比于中债的拟合值而言，误差基本被限制在0.05之内，达到了较好的拟合效果，符合预期的设计要求

2.3 Monte Carlo模拟法计算VaR

VaR的产生

最初提出VaR这个概念，是为了应对金融风险中的市场风险金融风险曾给投资者带来过极大的损失，1995年2月，英国历史最悠久的巴林银行宣告破产，原因就是交易员Leeson在交易中损失了13亿美金一度被认为保守的巴林银行的破产给许多银行机构敲响了警钟，然而类似的例子还有很多，所以风险衡量就应运而生了

最初的风险衡量标准是波动性，但是波动具有双向性，可能是向上波动，也有可能是向下波动，如果我们使用波动性来衡量风险，那么向上波动也会是风险的一种，当然投资者不会认为收益是一种风险，他们只会关注赔钱的可能性，所以需要更加合理的风险衡量标准VaR风险价值就是这个更加合理的风险衡量标准

VaR的概念

VaR，是Value at Risk的缩写，即在险价值通常来说，VaR由三个部分组成：时间段、置信水平和损失金额我们可以用VaR的三个部分来描述某个投资组合的风险，比如说我可以以95%的置信水平确定下个月投资者最多会损失100万美金或者说给定置信度为95%，某投资组合24小时内的VaR为100万美金，那么认为未来24小时内，该投资组合亏损100万美金的概率为5%

使用数学表达式表述VaR：

$$P(PtVaR)=a(2.8)$$

其中：

P代表资产价值损失小于最大可能损失的概率，原意为Probability

P代表某一投资组合在一定持有期t的价值损失额

VaR代表给定置信水平a下的在险价值，即可能的最大损失

a代表给定的置信水平。

VaR的计算方法

VaR的计算方法主要有三种：德尔塔-正态法、历史模拟法和Monte-Carlo模拟法下面简单介绍一下这三种方法

德尔塔-正态法

德尔塔-正态法假定投资组合的回报（return）服从正态分布，所以可以利用正态分布置信度与分位数对应的良好特性来简化计算
由回报服从正态分布我们就可以得到VaR计算公式如下：

$$\text{VaR} = \text{Pt}(2.9)$$

其中：

VaR代表风险价值。

P代表给定置信度对应的分位数

代表回报所服从的正态分布的标准差

t代表持有期。

历史模拟法

历史模拟法的原理是根据历史回报数据的变化来推测未来回报数据的走势，其公式如下：

$$R_{(p,k)} = \sum_{i=1}^N w_{(i,t)} R_{(i,k)} \quad (k=1,2,3,t) \quad (2.10)$$

其中：

$R_{(p,k)}$ 代表投资组合在时间k的收益率

$w_{(i,t)}$ 代表时间t的投资权重

$R_{(i,k)}$ 代表组合中第i只证券在时间k的收益

大致计算步骤如下：

建立一个多年所有市场变量的每日变动数据库

第一次模拟假设每个市场变量的波动率与数据库覆盖时间的第二天相应变量的波动率相同

第二次模拟假设各市场变量的波动率与数据库覆盖时段的第二天相应变量的波动率相同

以此类推，每次模拟就可以计算出一个投资组合的样本值

Monte Carlo模拟法

Monte-Carlo模拟法又称为计算机随机模拟方法，是一种基于随机数的计算方法它起源于第二次世界大战当中的“曼哈顿计划”，为了解决原子弹研制中的中子随机扩散过程，美国科学家提出了这一模拟方法

蒙特卡洛模拟法利用随机数产生收益率或者资产价格的一个分布我们可以通过随机数产生一个服从正太分布的数据集来模拟未来数据的变动，从而求得未来可能的损

失，得到VaR在险价值

Monte-Carlo模拟法计算VaR值常常使用以下步骤：

生成随机模拟收益率或者收益。

重复历史模拟法中的步骤计算VaR

其原理如下：

图2.24 Monte Carlo模拟法计算VaR原理图

在该毕业设计中，使用的就是蒙特卡洛模拟法计算VaR在险价值
实证分析

在这个毕业设计中，按照要求使用蒙特卡洛模拟法来计算VaR在险价值假设现有的回报数据是服从正态分布的，那么由于回报数据服从正态分布，所以可以使用德尔塔-正态法简化计算德尔塔-正态法需要3个参数，置信度对应分位数P、正态分布标准差以及持有期t通过对历史数据进行计算，可以求得标准差和平均值，然后通过这两个参数可以使用随机数模拟出一个新的正太分布，然后可以作为计算在险价值的新数据

具体可以通过如下步骤进行计算：

求历史数据的标准差以及平均值x

使用随机数模拟至少10000个数据，服从标准差为以及平均值为x的正态分布

求出指定置信度对应的分位数，以及新数据的标准差

根据德尔塔-正态法的公式计算在险价值

查询中债网VaR数据，可以得到2018年4月20日代码为1282337的债券的最新在险价值数据

具体如下：

图2.25 中债网在险价值数据图

通过在Wind财经软件上查询，可以得到3月1日到4月20日1282337号债券最新的财经数据
数据截图如下：

图2.26 债券数据截图

其中包含中债网估计的收益率数据，可以利用该数据进行在险价值预测通过Python程序计算，这组收益率数据的平均值为 $\mu = 5.28$ ，标准差为0.2利用Python的随机数生成模块，可以得到一万组模拟收益率数据，数据如下：

图2.27 蒙特卡洛模拟数据截图

通过Python程序的matplotlib模块，可以描绘出这一万组数据的分布情况，如下图：

图2.28 蒙特卡洛模拟数据分布图

所以通过这一万组数据，以及它的正态分布模型，可以进一步利用德尔塔-正态法计算在险价值VaR

公式如下：

$$\text{VaR} = \text{Pt}(2.11)$$

所以通过Python程序的计算，可以得到置信水平为95%，持有期为1时，在险价值VaR = 0.3342，截图如下：

图2.29 VaR计算结果截图

通过Python程序的计算，可以得到置信水平为99%，持有期为1时，在险价值VaR = 0.4727，截图如下：

图2.30 VaR计算结果截图

通过对比，可以发现与中债网所登记的数据基本吻合，由于数据量有限，以及随机模拟存在误差，所以对比时还是有一点点小误差

第3章 总体设计方案

3.1 总体开发流程

整个软件的设计开发流程应当遵循以下步骤：

系统功能划分，对应着软件开发中的客户需求分析，在开发之前首先要了解客户需要什么功能，然后才能着手设计开发

系统架构设计，在了解了客户的真实需求之后，需要对软件设计的整体思路有一个规划，在之后的开发过程中需要遵循这个思路来进行

软件开发，这一部分按照系统架构的不同，可以分为3个主要部分：前端开发、后台服务器开发和数据库设计

软件测试，在开发完成之后，需要对代码的可靠性和各种可能出现的异常情况进行测试，做好正向测试与反向测试，保证开发的软件在复杂的环境下具有很高的可靠性

软件的整体设计流程图如下：

图3.1 软件设计流程图

总体架构设计

软件架构指的就是软件的基本结构，合适的架构是一款软件成功的重要因素之一软件架构可以按照结构的不同分为分层架构、事件驱动架构、微核架构、微服务架构以及云架构其中最基本且最常见的一种软件架构，所以该毕业设计就采用分层架构设计软件

分层架构将软件分成若干个水平层，每一层都有清晰的角色和分工，不需要知道其他层的细节，层与层之间通过接口通信常见的分层架构将软件设计分为4层：表现层、业务层、持久层和数据库层由于毕业设计的软件采用NoSQL模式的MongoDB数据库，所以没有太多用于查询数据的SQL脚本，所以用于提示数据的持久层就被取消了，采用3层分层架构设计——表现层、业务层和数据库层，由于设计的实际需求，这三层架构对应成为了用户视图层、Node代理层和应用层

整体架构设计如下：

图3.2 软件架构设计图

其中，用户视图层负责向用户展示视图（数据、图表等），与用户进行交互Node代理层负责搭建后台服务器，实现后台接口，实现请求派发以及路由控制应用层负责处理派发的任务，执行指定的代码，返回系统所需的数据整体工作流程大致如下：

图3.3 软件工作流程示意图

其中，每个步骤所代表的含义如下：

用户输入网站网址，浏览器向服务器发起HTTP请求

Node Express收到请求，返回HTML页面给用户浏览器

用户使用软件的某项功能，浏览器向服务器发起HTTP请求

Node Express收到请求以及附带的参数，并把参数传递给任务需要执行的Python脚本程序

Python程序开始执行，在MongoDB数据库中查询所需要的数据

MongoDB向Python程序返回其所需要的数据

Python执行完所有的任务，并把结果返回给Node程序

Node Express返回数据给用户浏览器，给用户呈现执行结果

以上基本是一个完整的交互流程，但是有时候不一定需要执行Python脚本程序，Node程序可以直接查询MongoDB数据库中的数据并返回给用户的浏览器

总体解决方案

后台程序主要负责业务逻辑的处理，主要包括数据获取、数据存储、数据分析、数据处理和数据查询等任务后台开发又分为Python程序开发、Node程序开发和MongoDB数据库开发，其中数据获取、数据分析、数据处理由Python程序实现，服务器搭建、开发后台接口、调用Python脚本程序和数据查询由Node程序实现，数据存储由MongoDB数据库实现整个程序设计可以放在同一个文件夹中构成一个完整的解决方案，解决方案的目录结构如下：

图3.4 解决方案结构图

其中bonds_system文件夹代表前端开发的项目文件夹，Data2Mongo代表用于处理数据并存储到MongoDB数据库的Python程序项目文件夹，NodeProject表示用于搭建服务器、查询数据的Node脚本程序的项目文件夹，PythonProject表示用于数据处理、数据分析、数据获取的Python程序项目文件夹

总体功能设计

根据毕业设计的任务书，整个系统应该具有以下功能：

收益率曲线数据

中国国债历年信息

美国国债历年信息

SHIBOR历年信息

LIBOR历年信息

最新市场债券信息

PCA分析

VaR在险价值

(1) 中债VaR数据展示

(2) VaR计算器

中债估值数据展示

数据查询

自定义收益率曲线拟合

第4章 后端程序设计

4.1 Python程序设计

Python语法简洁清晰，使用Python编程简明清晰，容易理解其次Python具有丰富和强大的库它常被昵称为胶水语言，能够把用其他语言制作的各种模块（尤其是C/C++）很轻松地联结在一起这就使得在使用Python编写一些难度较大的软件时，能够找到很多已经开发好的模块或者包，简化编程工作，也使得Python这门语言更加强大使用C/C++作为底层的好处就是Python程序运行速度很快，扩展空间很大，兼容性很好，功能更完善这也是为什么Python这门语言在当今编程语言百花齐放，开源环境纷繁复杂的情况下，仍然能够占据一席之地，甚至在金融大数据、人工智能、深度学习等方面甚至越来越重要的原因

第3章3.3节中已经说明数据获取、数据分析、数理由Python程序实现，所以本节就按照Python所需执行的任务依次阐述Python程序的实现过程

爬取数据

Python爬虫

Python爬虫，是一种专门用于抓取网络上指定数据的程序通过Python爬虫，可以从互联网上抓取大批量的数据，节省大量的人力物力由于在这次设计中需要从网络上抓取大量的债券数据，所以需要使用到Python爬虫技术

爬取数据可以分为网页数据爬取和接口数据爬取，根据爬取对象的不同，Python爬虫也大致分为两种，一种直接请求指定接口的数据，另一种通过请求HTML网页文件，然后解析HTML获取数据**本次毕业设计两种方式都有使用到**

接下来以爬取中国国债日收益率历史数据为例，阐述Python程序的开发，由于数据爬取过程相对而言比较简单，所以没有使用已经封装好的Python爬虫框架，直接使用Python自带的模块实现

爬取中国国债数据

本次数据爬取的对象确定为中债网，中债网的中国国债日收益率以Excel文件的形式提供给用户下载：

图4.1 下载页面截图

通过查看其浏览器的控制台可以看到其HTML代码如下：

图4.2 下载页面代码截图

可以发现是一个ul标签，内嵌了一系列的li标签，形成了一个链接列表，其中的a标签指明了下载Excel数据文件的url和标题，所以只需要解析到这个a标签即可，然后根据对应的url下载服务器资源

使用到的Python包为urllibrequest和pyquery，其中urllibrequest用于发起HTTP请求，而pyquery是一个类似jQuery的HTML DOM操作库，但是它是在服务器端对DOM进行渲染和操作，而非浏览器，可以用于对返回的HTML文件进行解析由于pyquery是第三方模块，所以需要使用pip（Python包管理器）安装整个程序只有两个函数，一个用于获取Excel下载的url，一个用于根据提供的url下载对应资源，并保存到本机指定位置

获取url的函数代码如下：

图4.3 代码截图

根据url下载对应Excel数据文件的函数代码如下：

图4.4 代码截图

依次执行以上两个函数，所有Excel文件会被下载到指定的路径下：

图4.5 下载文件截图

以上就是爬取中国国债日收益率历史数据的方法，其他数据的爬取方式与此大同小异，不再赘述
处理数据

4.1.1节中，已经将所有Excel数据文件爬取到了指定目录下，但是程序运行过程中不直接使用Excel文件，最好是将数据存储到数据库中，这样数据结构统一，程序运行时查询数据的速度也远远快于直接访问Excel文件所以于下载好的Excel数据文件，还需要对其中的数据进行整理，如果Python进行数据分析，还有可能需要使用CSV格式的文件接下来仍然以中国国债日收益率历史数据为例进行阐述

整合成CSV文件

经过对比发现，中国国债由于还在飞速发展的过程中，所以期限结构仍然还不是特别稳定，从2002年1月4号开始到2009年11月27号共有19期标准期限从2009年11月30号到2015年12月31号共有21期标准期限从2016年1月4号至今共有16期标准期限为了便于研究，取所有数据的共有标准期限，然后整合成一个CSV文件，这14期标准期限为：0, 0.08, 0.17, 0.25, 0.5, 0.75, 1, 3, 5, 7, 10, 15, 20, 30，单位是年

由于需要处理所有的Excel文件，所以需要遍历指定的文件夹，使用Python读取Excel文件需要使用到xlrd模块，xlrd模块是第三方模块，需要使用pip进行安装

此处进行数据处理的函数也分为两个：getFileNames函数用于获取指定目录下所有文件的文件路径，readXLSX函数用于根据传入的Excel文件路径集合读取所有指定的Excel，如果是dimensions中的共有标准期限，那么就将数据存储到集合中，最后将所有数据写入一个指定的CSV文件中，达到数据过滤整理的目的

getFileNames函数代码如下：

图4.6 代码截图

readXLSX函数代码如下：

图4.7 代码截图

依次运行两个函数，可以得到一个汇总数据的CSV文件，总共有4077条数据，14个维度

存储数据到MongoDB

由于需要遍历目录、读取Excel以及插入数据到MongoDB，所以需要使用的模块有os、xlrd和pymongo，其中pymongo是一个第三方的用于MongoDB操作的Python驱动，需要使用pip进行安装该部分代码分为两个函数，getFileNames函数与前一部分相同，用于获取指定目录下所有的文件的路径readXLSX函数区别于前一部分，因为需要向MongoDB插入数据，所以需要将每一条数据构建为一个字典，然后将所有数据汇总到一个list结构中，最后使用pymongo的MongoClient构造方法构造的对象中的insert_many方法批量插入到MongoDB中，以便提高插入速度

readXLSX函数代码如下：

图4.8 代码截图

插入MongoDB数据库的代码如下：

图4.9 代码截图

分析数据

Python程序要实现的数据分析任务包括PCA分析、收益率曲线拟合以及VaR在险价值计算由于Python所要实现的数据分析程序要被Node程序调用，所以在本节之后的所有Python程序都需要引入sys模块，以接受外部程序传入的参数[接下来按照不同的任务依次阐述设计过程](#)

PCA分析

在第二章第一节中已经研究了PCA算法的理论，并使用SPSS统计学软件进行了计算分析，接下来使用Python程序进行PCA分析Node程序调用Python进行PCA分析时，需要将CSV数据的地址以及所要提取的主成分个数传入，Python使用传入的数据进行PCA分析，提取出指定个数的主成分之后，将各个维度的贡献度、累计贡献度以及因子载荷矩阵数据存储到指定路径下的CSV文件中，Node程序通过读取CSV文件中的数据向用户返回分析结果数据

该程序主要使用第三方模块numpy进行主成分分析，需要先使用pip进行安装，安装之后在文件开头通过import引入为了防止生成的CSV文件重名，此处使用uuid模块生成全球独一无二的标志，最终作为文件名保存

这个程序分为四个主要函数：readCSV函数负责读取指定路径的CSV文件中的数据，返回一个numpy matrix矩阵结构变量calculateContribRate函数负责计算个维度累计贡献度D2Matrix2CSV函数负责将numpy matrix变量写入CSV文件最重要的PCA方法实现PCA分析，传入多维数据，返回特征向量和因子载荷矩阵

readCSV函数代码如下：

图4.10 代码截图

此处使用的是numpy模块的genfromtxt方法读取CSV文件中的数据

calculateContribRate函数代码如下：

图4.11 代码截图

D2Matrix2CSV函数代码如下：

图4.12 代码截图

这里主要是使用matrix矩阵转化为list变量，然后通过for循环遍历其中的元素，构造一个符合CSV格式的字符串，最终保存到指定路径的文件中即可

PCA函数代码如下：

图4.13 代码截图

所有注释基本都写在了代码中，基本步骤完全遵循2.1.1节

收益率曲线拟合

基于2.2节的研究，选择Nelson-Siegel模型作为收益率曲线拟合模型使用第三方数学分析模块scipy的optimizefmin函数优化Nelson-Siegel参数，选择最合适的四个参数确定Nelson-Siegel模型，最后返回拟合的数据

此程序除了依赖scipy模块之外，还依赖pandas、numpy、pymongo等第三方模块，这些模块都需要使用pip进行安装另外，还需要引入内置模块json，以便于Node与Python进行数据交互

程序代码如下：

图4.14 代码截图

基于Monte Carlo模拟法计算VaR

基于2.3节的结论，很容易得到VaR的计算方法，其大致步骤如下：

接受一组一维参数，代表某支债券的历史到期收益率数据

计算这组一维历史数据的平均值和标准差

根据求出的平均值和标准差，使用随机函数生成一组符合要求的且服从正态分布的随机数

根据这组随机数，计算VaR。

当然在实际开发中还要考虑到其他参数，比如置信区间、债券的持有期以及Monte Carlo模拟的次数

这个程序的开发依赖一下模块：numpy、scipystats、xlrd、os、re，其中numpy、scipystats和xlrd需要使用pip命令安装

整个程序的核心是一个函数：calculateVaR，它负责计算VaR其具体代码如下：

图4.15 代码截图

实证分析

在实现所有数据分析功能之后，仍然使用中国国债数据作为样本，验证程序分析的正确性基于软件的数据分析如下：

PCA分析

通过软件将中国国债数据上传之后，点击“PCA分析”，得到贡献度与因子载荷矩阵如下：

图4.16 PCA分析结果截图

与图2.1所代表的SPSS软件的PCA分析结果图对比，相差不超过0.05%

图4.17 因子载荷矩阵图

上图是软件分析出的中国国债数据的因子载荷矩阵，下图是SPSS软件分析出的因子载荷矩阵：

图4.18 SPSS分析因子载荷矩阵截图

对比也可以发现结果非常的接近，可以证明程序的分析效果很好

曲线拟合

关于软件进行收益率曲线拟合的实证分析，请阅读本文2.2.4节——Nelson Siegel模型的实际拟合效果，本节不再赘述

VaR计算

关于软件进行VaR计算的实证分析，请阅读本文2.3.4——实证分析，本节不在赘述

4.2 Node程序设计

Nodejs是一个基于Chrome V8 引擎的 JavaScript 运行环境，它允许我们在服务器上运行JavaScript，使得JavaScript不再局限于浏览器环境Nodejs是一门基于事件驱动的、非阻塞式I/O的异步编程语言，单线程异步编程是其最大的优点也是其最大的缺点缺点在于JavaScript单线程运行，不方便充分利用多核CPU的性能，并且异步编程思想与常见的C++、Java和Python等同步编程语言有很大的差异，不太符合程序员的编程思维，有时候容易写出“callback hell”代码，难以维护但是单线程异步编程也是Nodejs的一大优点，因为区别于传统的多线程语言，JavaScript是基于事件驱动的，对于高并发的处理具有优秀的表现，并且对于I/O频繁的程序，JavaScript能够充分利用其优势，无需等待I/O完成，使得单线程代码具有多线程的效果

通过上面的介绍，不难发现Nodejs的优势在于高并发、多I/O的应用场景，而弱势在于计算复杂且频繁的场景由于该毕业设计所需设计的软件涉及的计算不是很多，而面向用户时可能会存在高并发的场景，所以选择Nodejs开发服务器，而Python来完成数据计算，充分地利用了两者的优势

根据第3章的总体设计方案，Node程序负责搭建服务器、开发后台接口、查询数据以及运行Python脚本，充当的是一个代理的角色所以下面分为4节阐述Node程序的开发

4.2.1 搭建服务器

Node自身具有的http模块就可以很方便的搭建HTTP服务器了，但是涉及到cookie、session等操作时原生方法比较麻烦，所以选择用一个已经封装好各种方法的Node框架来搭建HTTP服务器，这里使用Express

Express是一个基于 Nodejs 平台的极简、灵活的 Web 应用开发框架，它提供一系列强大的特性，帮助开发者创建各种 Web 和移动设备应用Express丰富的中间件可以帮助开发者快速的调用各种API，以及针对HTTP数据信息做处理

由于实际的服务器搭建脚本程序代码很多，所以不将所有代码展示，只阐述一下大致步骤：

使用npm包管理器安装Express框架

在Node脚本中引入express模块

使用express模块的构造方法构造app应用

使用应用的static方法，指定静态文件目录，将网站文件放入该指定的文件夹

使用app的get方法，为路由“/”指定处理函数，在回调函数中将网站的入口文件indexhtml使用res对象的sendFile方法发送给浏览器

使用app的listen方法指定网站的端口，本设计中指定为8000

以上步骤基本就完成了了一个简单的服务器的搭建，使用Node指定这个脚本，然后打开浏览器输入指定的网址，就可以访问到静态目录下文件生成的网站

app的get方法代表的就是为指定路由的GET方法指定处理函数，后续开发后台API时仍然需要使用这个方法，其他类似方法包括app.post等

4.2.2 开发后台接口

作为一个完整的软件，前端页面是软件的GUI，后台程序是软件的逻辑业务代码，数据库是软件的数据中心前端页面必须与后台程序进行通信才能实现软件的完整功能，而API接口就是前后台通信的关键API接口的作用就是在收到客户端的HTTP请求时，解析参数并执行指定任务后返回结果给客户端在Node服务器代码中，通过使用app对象的一系列方法（get、post、delete、put等）可以实现后台服务器接口开发

根据3.4节的总体功能设计及实际开发需要，设计了以下11个公开的接口，部分其余内部接口不公开

接口文档如下：

GET “/News” 获取新闻公告数据，无需传递参数

POST “/BondsData” 返回指定年份指定类型的所有收益率数据及数据库中该类型数据的所有年份参数为数据类型type（int型参数，1代表中债、2代表美债、3代表SHIBOR、4代表LIBOR），数据年份year（比如2018，如果为空则返回数据库中最新年份的数据）

GET “/Download” 返回指定年份指定类型的Excel文件数据，参数为数据类型type（字符串类型数据，‘1’代表中债、‘2’代表美债、‘3’代表SHIBOR、‘SZ’代表深交所债券数据、‘SH’代表上交所债券数据、‘GZ’代表国家发行的债券数据）和year（数据年份）

GET “/DetailData” 返回指定类型的所有最新债券数据，参数为数据类型type（字符串类型，比如‘SZ’、‘SH’和‘GZ’）

GET “/YieldCurve” 返回指定日期与类型的债券的拟合收益率曲线数据参数为date（数据日期，比如‘2018/05/07’）和type（数据类型，比如‘SZ’）

GET “/ZZValuation” 返回所有中债估值数据，按时间降序排列

GET “/ZZVaR” 返回所有中债VaR数据，按时间降序排列

POST “/CalculateVaR” 返回指定置信区间、持有期和模拟次数的数据的在险价值参数为数据上传方式type（1代表手写字符串数据、2为Excel或CSV文件数据上传），手写字符串数据data（使用文件上传方式时data可为空），上传文件的文件名file（比如‘datacsv’）、置信区间confidenceLevel（比如0.95），持有期holdingPeriod（比如1），模拟次数simCount（比如10000）

POST “/Admin” 验证账号密码登录，返回值为登录码（-1代表账户不存在、0代表密码错误、1代表登录成功）参数为账户名userName、密码password和是否记住密码的标志量remember（比如true）

POST “/CalculatePCA” 计算PCA主成分的接口，返回各维度的累计贡献度以及因子载荷矩阵参数为数据上传方式type（1代表手写字符串数据、2为CSV文件数据上传），手写字符串数据data（使用文件上传方式时data可为空），上传CSV文件的文件名file（比如‘datacsv’）、所要提取主成分的个数componentCount

POST “/Search” 根据搜索条件搜索数据参数为债券代码code（比如127326）、债券简称name（比如‘15国网05’）、债券类型type（0代表全部、1代表国债、2代表沪企债、3代表深企债、4代表中债VaR、5代表中债估值）

4.2.3 查询数据

由于用户获取的数据不是每次都要经过计算才能得到，有时候只需要直接到数据库查询数据就好，所以没有必要使用Python脚本，直接使用Node进行数据库查询就可以

实现，这样免去了中间的多余环节，加快了数据查询速度

Node连接MongoDB数据库需要使用到mongodb模块，使用npm包管理器进行安装在Node脚本中引入mongodb模块之后，使用其中的MongoClient对象的connect连接数据库，在回调中可以得到一个数据库连接对象，通过这个对象可以指定数据库和集合进行增删查改以获取债券详细信息的函数为例，阐述具体开发时如何使用Node查询MongoDB

getDetailData函数是“/DetailData”接口的处理函数，它接收三个参数：url代表数据库连接的url，type代表债券类型，res代表Node的HTTP响应对象具体代码如下：

图4.19 代码截图

获取到dbo数据库连接对象之后，可以通过该对象的db方法获取数据库对象，通过数据库对象的collection方法，可以获取到指定的集合对象使用集合对象的find方法，可以对数据库中的数据库指定条件进行查询

4.2.4 运行Python脚本

Node作为中间代理层，负责搭建服务器、响应请求、查询数据，但是用户还需要数据分析，这一部分不是Node的长处，需要使用Python实现，所以Node必须能够运行Python脚本并且能够与Python程序进行通信

本次设计中选择python-shell这个Node模块作为驱动，运行Python程序，通过npm安装之后再脚本内引入即可使用通过python-shell模块的run方法可以运行指定路径的Python脚本

以运行更新所有数据的Python脚本为例说明Node如何运行Python脚本程序，具体代码如下：

图4.20 代码截图

可以看到run方法接受2个参数，第一个参数代表所要运行的Python脚本的路径，第二个参数为获取到运行结果后的回调函数，result即为运行后的结果

其实run方法还可以接受第三个参数，当接受三个参数时，第一个参数为Python脚本程序的路径，第二个参数为要传入的参数数组，第三个参数是回调函数

4.3 数据库设计

由于所要存储的数据结构多变，并且同类型数据可能结构也不一样，所以如果使用传统的关系型数据库的话就会比较麻烦，所以这个毕业设计选用流行的非关系型数据库MongoDB作为后台数据库

除了以上原因之外，MongoDB还有许多优点是选择它的原因MongoDB是基于分布式文件存储的、使用C++编写的、基于类似json格式的bson格式存储的高性能数据库它没有数据表和数据行的概念，对应的是集合和文档，一个文档就是一条数据记录，一个集合就是一个数据表MongoDB的文档即为一个JavaScript对象，可以很方便的转化为json字符串数据，并且在前后端之间的查询、删除、存储十分方便Python中的字典、JavaScript中的对象可以直接使用insert方法插入数据库Python中的list、JavaScript中的数组对象，可以传递给insert批量插入数据库，大大提高插入效率根据测试结果，MongoDB的增删查改操作比MySQL数据库要快，这也是其优势所在

为了便于管理，只设计了一个数据库BondsData，这个数据库下包含9个集合：ChinaBonds、AmericaBonds、SHIBOR、LIBOR、News、DetailData、ZZVaR、ZZValuation和Accounts其中ChinaBonds存储所有的中国国债日收益率历史数据，AmericaBonds存储美国国债日收益率历史数据，SHIBOR存储上海同业银行拆借利率日收益率历史数据，LIBOR存储伦敦同业银行拆借利率日收益率历史数据，News存储所有的新闻公告数据，DetailData存储所有的债券交易数据（包括国债、上交所、深交所数据），ZZVaR存储中债VaR数据，ZZValuation存储中债估值数据，Accounts存储所有的管理员账户信息

由于数据类型及关系复杂，所有字段数量并不统一，所以每个集合都没有统一的数据结构，可能第一条数据有21个字段，第二条数据就只有19个字段所以在论文中不

阐述每个集合的表结构

使用MongoDB的可视化工具可以直观的查看数据库中的数据：

图4.21 MongoDB数据截图

第5章 前端程序设计

5.1 界面设计

在开始写代码之前，先设计好网页的外观界面，整体设计大致如下：

图5.1 界面设计图

其中顶部是网页头部，左边是侧边栏，右侧是内容展示栏设计的网站具有多项功能，选项卡就放在左边的侧边栏中

经过细化之后，设计得到具体页面如下：

图5.2 界面设计详细图

左边的一系列选项卡用于选择具体的功能，右边的空白处负责向用户展示界面

5.2 模块选用

伴随着前后端的分离，前端开发也越来越注重工程化、系统化，传统的开发模式已经不再适用，最明显的体现就在于最近几年百花齐放的MVC、MVVM前端开发框架——比如Vue、React、Angular等，这些MVC或MVVM框架的出现使得前端必备的jQuery工具不再重要，而这些框架都有一些重要的共同点，比如注重视图层、模型层与控制层的分离，它们往往只负责视图层比如都使用组件化的思想开发比如都是数据驱动视图

为了简化开发过程，优化代码结构，方便后续的代码维护工作，需要选择一个合适的前端开发框架，该毕业设计中选择React，因为React的JSX语法简洁灵活，并且生态环境丰富，多种多样的插件层出不穷，能够快速解决开发中遇到的问题为了简化React的开发环境配置，选择使用React开发脚手架——create-react-app，使用npm安装该模块之后，即可使用

通过create-react-app搭建开发环境后，就可以开始进行编程了，在项目文件夹下使用Node命令“ npm run start ”即可建立一个本地开发环境服务器，便于开发过程中进行调试在开发过程中还需要使用其它模块，都需要通过npm包管理器进行安装

具体用到的模块及其版本号如下：

图5.3 依赖模块

其中：

antd模块是蚂蚁金服开发的一个React组件库，用于快速开发网页界面

axios模块是一个AJAX封装包，使用这个包可以快速简捷的使用AJAX访问后台接口

highcharts是一个使用纯JavaScript写的前端图表绘制库，使用这个模块可以很方便的绘制很复杂的数据图表

react、react-dom、react-scripts是create-react-app自动安装的模块，是使用React进行开发的必备模块

react-router-dom是进行前端路由管理的库，使用这个库可以很方便的生成前端路由

5.3 路由设计

在Web开发中，路由就是把某个url的请求分配到对应的程序进行处理因为所要设计的软件具有许多功能，所以应该分多个页面进行展示，最好是使用路由进行管理
React-Router是一个基于 React 的强大路由库，它可以让用户向应用中快速地添加视图和数据流，同时保持页面与 URL 间的同步

在开发之前，先根据系统所要实现的功能将网站对应的划分为如下的路由：

“ / ”：主页面，暂定为新闻公告页面

“ /BondsData/ ”：债券数据页面

“ /BondsData/ChinaBonds ”：中国国债数据页面

“ /BondsData/AmericaBonds ”：美国国债数据页面

“ /BondsData/LIBOR ”：LIBOR数据页面。

“ /BondsData/SHIBOR ”：SHIBOR数据页面。

“ /DetailData ”：深企债、沪企债、国债当前数据页面

“ /PCA/ ”：主成分分析页面

“ /VaR/ ”：VaR在险价值计算页面

“ /VaR/Data/ ”：中债网数据展示界面

“ /VaR/Calculator/ ”：根据上传数据计算在险价值页面

“ /ValuationData/ ”：中债估值数据界面

“ /Search/ ”：债券数据查询页面

“ /CurveFit/ ”：收益率曲线拟合页面

“ /Admin/ ”：管理员界面，需要登录，用于处理数据更新等问题

5.4 页面效果

根据5.3节的路由规划，开发出完整软件之后，整个软件的设计开发工作基本就完成了，本节展示软件界面效果
日收益率历史数据展示：

图5.4 中国国债日收益率历史数据

拟合国债收益率曲线效果展示：

图5.5 中国国债收益率曲线拟合图

PCA分析界面：

图5.6 PCA分析界面

VaR计算器界面：

图5.7 VaR计算器界面

中债VaR数据展示界面;

图5.8 中债VaR数据展示界面

数据搜索界面：

图5.9 数据搜索界面

第6章 总结与展望

6.1 总结

该毕业设计所做的工作是设计一款软件，能够获取自定义债券收益率数据能够利用PCA算法对债券数据进行分析，提取出3个主因子，计算累计贡献度并分析3个主因子对债券期限结构的影响能够利用Nelson-Siegel模型拟合收益率曲线，得到标准期限之外的各个期限的到期收益率数据能够利用Monte Carlo模拟法计算在险价值，使得用户对于债券的风险性及最大可能损失有直观的了解能够基于债券代码、债券简称和债券类型进行数据查询，使得用户随时可以查询某支债券的数据能够进行自定义收益率曲线拟合，使得用户可以对于任意债券可以进行收益率曲线拟合，得到某些非标准期限的到期收益率针对以上工作，本设计完成了以下工作：

编写Python爬虫程序，完成债券数据收集工作：

使用Python爬虫获取了中国国债、美国国债、SHIBOR、LIBOR日收益率历史数据，并以Excel格式存放在了指定目录下

编写Python程序，完成债券数据整理、存储、分析工作

将所有的爬取到的Excel文件中的数据分类整理到一个CSV文件中针对不同类型的数据使用Python程序完成了PCA主成分分析，提取出了三个主因子，计算出累计贡献度和因子载荷矩阵，结合Excel图表分析了各个主因子对于债券期限结构的影响使用Python程序完成了基于Nelson-Siegel模型的收益率曲线拟合工作使用Python程序完成了基于Monte Carlo模拟法的在险价值计算将所有获取到的数据分类存储到了MongoDB中

编写Node程序，完成后台服务器的搭建、后台接口的搭建

使用Express框架搭建了一个HTTP服务器，使得网页能够通过HTTP协议访问开发了一系列后台接口，满足前端开发的需求

编写前端界面，完成数据展示、图表展示、用户交互工作

基于React和React-Router完成了前端界面的开发，基于之前搭建的HTTP服务器，搭建了一个具有完整功能的网站，该网站整合了所有功能，包括数据收集、数据分析、数据下载、收益率曲线拟合、VaR计算

构建了MongoDB环境，存储所有数据

6.2 展望

整个设计已经完成，回顾整个过程，仍有许多不足值得我去改进，主要包括以下几点：

由于对于金融知识掌握不够，学习能力也比较有限，所以在设计开发过程中遇到了许多的困难，尤其是如何利用Python实现算法，由于对于金融知识不够了解，导致开发Python程序时不知从何下手

有部分开发没有做好规划，比如在开发数据存储程序时，由于没考虑到2009年中国国债有两种期限结构，导致前端展示出现bug，最后重构代码才解决问题

前端开发有部分代码没有遵守代码规范，比如因为使用了ant-design这个UI组件库，所以不需要写多少css代码，为了减少文件个数，使用了内嵌式的style样式，这样虽然文件个数少了，但是在今后维护的时候会比较困难，并且代码比较丑陋，所幸不是很多

针对以上缺陷，对今后的工作做以下展望：

磨刀不误砍柴工，在开发软件之前做好基础知识的储备工作，比如在这次开发之前，先学习如何做PCA处理，学习曲线拟合以及蒙特卡洛算法等等专业知识语言只是工具，只有学好了底层的算法，才能发挥编程的最大威力

在开发之前做好规范和计划，学习scrum敏捷开发思想，做好反向测试，比如当某个数据缺少一个字段时，代码会出现怎样的异常

新增css文件用于开发专门的css代码，因为反正会在编译时将所有的css代码整合到一个css文件中，并不会影响网站性能，遵循代码开发规范（无论是前端还是后台）

致 谢

本设计在牟新刚老师以及企业指导老师夏宇的指导下完成，在设计的过程中两位老师悉心指导，尽心尽力，有问必答，给了我极大的帮助在论文的完成过程中，牟老师为我付出了很多，从选题到开题报告，从阶段性报告到最后论文的完成，牟老师为我提出了很多意见，指出了我的论文中存在的许多问题，严格把关并且不断催促提醒我们提交应有的材料，完成阶段性任务，在此我表示由衷的感谢夏宇老师在企业实习期间给了我很多的帮助，包括金融知识的讲解以及业务问题的回答，给了我很大的帮助，使得我对陌生的金融知识有了更多的了解，能够更快的熟悉业务知识，在此也深表谢意感谢两位老师的耐心指导，谆谆教诲

至此，论文已经基本完成，这也代表着我的四年大学生活画上了句号在学校的这四年时间，是我人生中最重要的一年，我学习到了很多知识，交到了许多朋友，学会了如何做一个对社会有用的人，也留下了许多美好的记忆在此感谢我的父母以及其他关心帮助过我的朋友，他们的关心和帮助让我度过了这美好的一年，我自身的成长也离不开他们的关心和爱护

检测报告由论文狗论文检测系统生成

