《大数据技术及应用》本科课程教学大纲

一、课程基本信息

<table>
<tr><td rowspan="2">课程名称</td><td colspan="6">（中文）大数据技术及应用</td></tr>
<tr><td colspan="6">（英文）Big Data Technology and Application</td></tr>
<tr><td>课程代码</td><td>2055043</td><td colspan="2">课程学分</td><td colspan="2">3</td></tr>
<tr><td>课程学时</td><td>48</td><td>理论学时</td><td>16</td><td>实践学时</td><td>32</td></tr>
<tr><td>开课学院</td><td>信息技术学院</td><td colspan="2">适用专业与年级</td><td colspan="2">计算机科学与技术专业、大学三年级</td></tr>
<tr><td>课程类别与性质</td><td>选修课</td><td colspan="2">考核方式</td><td colspan="2">考查</td></tr>
<tr><td>选用教材</td><td colspan="3">Python 网络爬虫技术与实践，吕云翔著，机械工业出版社，2023 年 6 月</td><td>是否为
马工程教材</td><td>否</td></tr>
<tr><td>先修课程</td><td colspan="5">计算机导论</td></tr>
<tr><td>课程简介</td><td colspan="5">大数据技术及应用是一门新兴的交叉性学科，是在信息技术领域和人工智能领域迅速兴起的计算机技术。数据挖掘技术面向应用，在很多重要的领域，数据挖掘都发挥着积极的作用。广大从事数据库应用与决策支持，以及数据分析等学科的科研工作者和工程技术人员迫切需要了解和掌握它。因此数据挖掘已经成为计算机专业及相关专业的重要课程之一。
本课程为计算机专业学生的专业实践课程。本课程主要介绍数据挖掘的基本概念，原理、方法和技术。旨在通过一学期的学习，使学生理解数据挖掘的基本流程，掌握数据挖掘的基本理论和技术，熟悉数据挖掘成果的显示；掌握数据挖掘的基本方法，能熟练地应用数据挖掘技术对现实数据进行有效的分析；结合相关软件能从大量数据中获取有价值的信息。</td></tr>
<tr><td>选课建议与学习要求</td><td colspan="5">本课程是适用于计算机类专业的专业实践课程，要求具有计算机导论的基础、基本编程能力和系统设计能力、对于数据库的基本操作能力和算法设计能力。</td></tr>
<tr><td>大纲编写人</td><td colspan="3">董辛酉（签名）</td><td>制/修订时间</td><td>2024 年 12 月</td></tr>
<tr><td>专业负责人</td><td colspan="3">戴志明（签名）</td><td>审定时间</td><td>2025 年 02 月</td></tr>
<tr><td>学院负责人</td><td colspan="3">（签名）</td><td>批准时间</td><td></td></tr>
</table>

二、课程目标与毕业要求

（一）课程目标

类型	序号	内容
知识目标	1	熟练掌握 Python 的基本语法与核心库（如 Pandas、NumPy、Matplotlib、Scikit-learn 等），能够高效处理金融数据的清理、转换、分析与可视化。
	2	掌握网络爬虫的基本原理与技术，包括静态网页采集（Requests、Beautiful Soup、lxml）与动态内容抓取（Puppeteer、Selenium），能够从多种数据源（如开放数据集、API、网页）中获取金融数据。
	3	使用 TA-Lib 计算常见技术指标（如 MA、RSI、MACD、布林带等），并基于这些指标设计交易策略（如趋势跟踪、均值回归等），掌握金融数据分析与策略优化的方法。
	4	了解金融数据的存储方式及其在大数据场景下的应用，掌握传统关系型数据库（如 MySQL）与分布式存储技术（如 HDFS、HBase、NoSQL 数据库）的使用方法，能够根据数据特性与性能需求选择合适的存储方案。
技能目标	5	数据采集与预处理能力，能够熟练运用网络爬虫技术从多种数据源中采集金融数据，并对数据进行清理、转换与预处理，确保数据质量。
	6	金融数据分析与策略设计能力，能够使用 TA-Lib 计算常见技术指标，并基于这些指标设计、优化与回测交易策略。同时，能够利用 Python 工具链（如 NumPy、Matplotlib、Scikit-learn）进行数据分析、可视化与建模。
	7	大数据存储与高效计算能力，能够根据数据特性与性能需求，选择合适的存储方案，并利用 Python 多进程技术实现高效的数据处理与并行计算，提升量化系统的性能。
素养目标（含课程思政目标）	8	学生将深刻理解数据安全与隐私保护的重要性，遵守相关法律法规，在数据采集、存储与分析过程中恪守职业道德，杜绝非法爬取、滥用数据等行为。
	9	树立科技报国与创新驱动发展的使命感，学生将认识到金融科技对国家经济发展的重要性，激发科技报国的使命感，培养创新精神与实践能力，努力将所学知识应用于金融领域的自主创新，助力我国金融科技的国际化与竞争力提升。

（二）课程支撑的毕业要求

L02 问题分析： 能够应用数学、自然科学和工程科学的基本原理，识别、表达、并通过文献研究分析复杂工程问题，以获得有效结论。 ①具备对系统设计、软硬件开发等涉及到的复杂工程问题进行识别与判断，并结合专业知识进行有效分解的能力。 ④在充分理解专业知识的基础上，能够运用所学知识开展文献检索和资料查询。
L06 工程与社会： 能够基于工程相关背景知识进行合理分析，评价专业工程实践和复杂工程问题解决方案对社会、健康、安全、法律以及文化的影响，并理解应承担的责任。 ②熟悉计算机专业领域相关的技术标准、知识产权、产业政策和法律法规。 ③能客观评价计算机应用项目的实施对社会、健康、安全、 法律以及文化的影响。
L07 环境和可持续发展： 能够理解和评价针对复杂工程问题的专业工程实践对环境、社会可持续发展的影响。 ①了解与本专业相关的职业和行业的生产、设计、研究与开发、环境保护和可持续发展等方面的方针、政策和法律、法规。 ②能正确认识并评价计算机科学在现实社会中应用的影响。

（三）毕业要求与课程目标的关系

毕业要求	指标点	支撑度	课程目标	对指标点的贡献度
L02	①	L	掌握网络爬虫的基本原理与技术，包括静态网页采集（Requests、Beautiful Soup、lxml）与动态内容抓取（Puppeteer、Selenium），能够从多种数据源（如开放数据集、API、网页）中获取金融数据。	10%
			使用 TA-Lib 计算常见技术指标（如 MA、RSI、MACD、布林带等），并基于这些指标设计交易策略（如趋势跟踪、均值回归等），掌握金融数据分析与策略优化的方法。	30%
			了解金融数据的存储方式及其在大数据场景下的应用，掌握传统关系型数据库（如 MySQL）与分布式存储技术（如 HDFS、HBase、NoSQL 数据库）的使用方法，能够根据数据特性与性能需求选择合适的存储方案。	30%
			大数据存储与高效计算能力，能够根据数据特性与性能需求，选择合适的存储方案，并利用 Python 多进程技术实现高效的数据处理与并行计算，提升量化系统的性能。	30%
	④	M	熟练掌握 Python 的基本语法与核心库（如 Pandas、NumPy、Matplotlib、Scikit-learn	20%

			等），能够高效处理金融数据的清理、转换、分析与可视化。	
			掌握网络爬虫的基本原理与技术，包括静态网页采集（Requests、Beautiful Soup、lxml）与动态内容抓取（Puppeteer、Selenium），能够从多种数据源（如开放数据集、API、网页）中获取金融数据。	20%
			使用 TA-Lib 计算常见技术指标（如 MA、RSI、MACD、布林带等），并基于这些指标设计交易策略（如趋势跟踪、均值回归等），掌握金融数据分析与策略优化的方法。	30%
			了解金融数据的存储方式及其在大数据场景下的应用，掌握传统关系型数据库（如 MySQL）与分布式存储技术（如 HDFS、HBase、NoSQL 数据库）的使用方法，能够根据数据特性与性能需求选择合适的存储方案。	30%
L06	②	M	数据采集与预处理能力，能够熟练运用网络爬虫技术从多种数据源中采集金融数据，并对数据进行清理、转换与预处理，确保数据质量。	50%
			学生将深刻理解数据安全与隐私保护的重要性，遵守相关法律法规，在数据采集、存储与分析过程中恪守职业道德，杜绝非法爬取、滥用数据等行为。	50%
	③	M	金融数据分析与策略设计能力，能够使用 TA-Lib 计算常见技术指标，并基于这些指标设计、优化与回测交易策略。同时，能够利用 Python 工具链（如 NumPy、Matplotlib、Scikit-learn）进行数据分析、可视化与建模。	50%
			树立科技报国与创新驱动发展的使命感，学生将认识到金融科技对国家经济发展的重要性，激发科技报国的使命感，培养创新精神与实践能力，努力将所学知识应用于金融领域的自主创新，助力我国金融科技的国际化与竞争力提升。	50%
L07	①	M	学生将深刻理解数据安全与隐私保护的重要性，遵守相关法律法规，在数据采集、存储与分析过程中恪守职业道德，杜绝非法爬取、滥用数据等行为。	50%

			树立科技报国与创新驱动发展的使命感，学生将认识到金融科技对国家经济发展的重要性，激发科技报国的使命感，培养创新精神与实践能力，努力将所学知识应用于金融领域的自主创新，助力我国金融科技的国际化与竞争力提升。	50%
	②	M	学生将深刻理解数据安全与隐私保护的重要性，遵守相关法律法规，在数据采集、存储与分析过程中恪守职业道德，杜绝非法爬取、滥用数据等行为。	50%
			树立科技报国与创新驱动发展的使命感，学生将认识到金融科技对国家经济发展的重要性，激发科技报国的使命感，培养创新精神与实践能力，努力将所学知识应用于金融领域的自主创新，助力我国金融科技的国际化与竞争力提升。	50%

三、课程内容与教学设计

（一）各教学单元预期学习成果与教学内容

第一单元 Python 基础及网络爬虫

本单元聚焦大数据技术及其应用，帮助学生掌握大数据的基本概念、核心特征及其时代背景，并培养运用工具解决实际问题的能力。课程将简明介绍 Python 基本语法，包括变量、数据类型、控制结构、函数与模块等，并展示其在 Web 开发、数据分析及机器学习等领域的应用。结合课程重点，将深入讲解网络爬虫技术，涵盖 HTTP 协议、网页结构解析、数据提取方法及反爬虫策略，通过实践帮助学生掌握爬虫程序设计与实现，为大数据分析提供数据支持。课程要求学生理解大数据概念与技术框架，掌握 Python 编程并完成数据分析与爬虫任务，通过团队合作完成综合性项目，同时积极参与课堂讨论，培养解决复杂问题的能力。

理论课时数：2；实验课时数：4

第二单元 金融数据采集与预处理

金融数据的采集是构建高质量数据集的关键步骤。数据来源多样，包括开放数据集、API、爬虫技术以及传感器数据。通过多种数据源的整合，学生将掌握如何构建一个支持分析与决策的金融数据集，理解数据采集的流程与技术要点，包括数据请求、解析、存储与管理。

数据预处理是确保数据质量的核心环节，其目标包括提高数据一致性、消除噪声、增强数据可用性。课程将系统讲解多种预处理方法，例如缺失值处理、异常值检测与处理、

数据转换、数据合并与重塑、特征工程、数据分割以及时间序列处理。通过理论与实践结合，学生将掌握从原始数据到高质量数据集的完整流程。

理论课时数：2；实验课时数：4

第三单元 静态网页采集

金融数据的爬取是量化分析与投资决策的重要基础。常见的静态网页采集技术主要依赖 HTTP 请求库和 HTML 解析库。首先，通过 HTTP 请求库向目标网页发送请求，获取网页内容。常用的工具包括 Requests 库，它是 Python 中最流行的 HTTP 请求库，支持 GET、POST 等多种请求方式，并能高效获取服务器响应。其次，利用 HTML 解析库对网页内容进行解析，提取所需数据。Beautiful Soup 是一个功能强大的 Python 库，能够将 HTML 或 XML 文档转化为易于操作的数据结构，支持通过标签、类名、ID 等选择器提取数据。此外，lxml 是另一个高效的解析库，尤其适用于处理大型文档，其解析速度通常优于 Beautiful Soup。

在实际操作中，首先使用 Requests 库发送 HTTP 请求，获取目标网页的 HTML 内容；随后，通过 Beautiful Soup 或 lxml 解析 HTML 文档，提取金融数据（如股票价格、交易量等）。例如，可以通过标签或类名定位特定数据，并将其存储为结构化数据。这些工具的组合为金融数据的自动化采集提供了高效、灵活的解决方案，是量化分析与策略研究的重要支撑。

理论课时数：2；实验课时数：4

第四单元 数据存储

本单元重点介绍金融数据的存储方式及其在大数据场景下的应用。金融数据具有高频率、大规模和多样化的特点，传统的关系型数据库（如 MySQL）在处理海量数据时面临性能瓶颈。为此，课程引入分布式存储技术，如 Hadoop Distributed File System (HDFS)，它通过分布式架构实现数据的高效存储与并行处理，适合存储大规模非结构化数据。此外，课程还将探讨 Apache HBase，这是一种基于 Hadoop 的分布式列式存储数据库，支持快速读写和随机访问，特别适用于金融领域的高并发场景。

课程还将介绍 NoSQL 数据库（如 Apache Cassandra 和 MongoDB）的基本使用方法，这些数据库以其灵活的数据模型和高扩展性，广泛应用于金融数据的实时处理与分析。同时，课程涵盖 Tornado，一个分布式框架，用于高吞吐量的实时数据传输与处理，在金融日志和事件流处理中发挥重要作用。这些大数据技术共同构建了高性能、高可用的数据存储与处理系统，能够满足现代金融数据分析的需求。在实际应用中，存储方式的选择需综合考虑数据特性、工作负载、性能需求及云服务支持等因素。

理论课时数：2；实验课时数：4

第五单元 JavaScript 与动态内容

在金融数据的动态采集中，目标网站常使用 JavaScript 渲染动态内容，传统的静态爬虫只能获取页面的初始 HTML，无法捕获经 JavaScript 处理后的 DOM 结构和异步

加载的数据。为解决这一问题，可采用无头浏览器（Headless Browser）技术。无头浏览器是一种无图形界面的浏览器，能够在后台运行并执行 JavaScript，模拟用户行为，从而获取动态生成的内容。常用的无头浏览器工具包括 Puppeteer 和 Selenium，它们可以加载页面、执行脚本并提取完整的 DOM 数据，确保爬虫能够捕获到动态渲染的金融数据。

处理 JavaScript 渲染页面的核心在于模拟浏览器环境。通过无头浏览器，爬虫可以执行页面中的 JavaScript 代码，获取异步请求的数据，并提取最终的 DOM 结构。例如，使用 Puppeteer 可以控制 Chrome 浏览器加载页面、点击按钮、滚动页面等操作，从而获取完整的动态内容。此外，结合 XPath 或 CSS 选择器，可以精准定位并提取所需数据。这种方法不仅适用于金融数据的采集，还可扩展至其他需要动态内容抓取的场景，为数据分析提供高质量的数据源。

理论课时数：2；实验课时数：4

第六单元 Python 多进程技术

在量化系统中，Python 多进程技术是提升数据处理效率的关键工具。多进程通过并行计算充分利用多核 CPU 资源，显著加速数据采集、指标分析、策略和回测等任务。在多进程数据采集中，可通过 multiprocessing 模块实现并行爬取金融数据（如股票行情、新闻等），避免单进程的 I/O 阻塞问题，大幅缩短数据获取时间。例如，将目标网站划分为多个子任务，由不同进程同时执行，最后合并结果，适用于高频数据抓取场景。

在多进程金融指标分析、策略和回测中，多进程技术可并行计算大量金融指标（如移动平均线、RSI、MACD 等），并同时对多个策略进行回测。通过将历史数据分段处理，每个进程独立计算指标或回测策略，最后汇总结果，显著提升分析效率。例如，使用 Pool.map 方法将数据分块分配给多个进程，适用于大规模数据集的复杂计算。结合 Python 的高效生态（如 Pandas、NumPy），多进程技术为量化系统提供了强大的性能支持，是高频交易和大规模数据分析的必备工具。

理论课时数：2；实验课时数：4

第七单元 金融数据的分析与处理

本单元重点介绍如何使用 TA-Lib 分析金融数据，涵盖常见技术指标与交易策略。TA-Lib 提供了超过 150 种技术指标，如移动平均线（MA）、相对强弱指数（RSI）、布林带（Bollinger Bands）和 MACD 等，适用于股票、外汇和加密货币等市场。通过 Python 编程，学生将学习如何计算这些指标，并基于指标设计交易策略，例如利用 RSI 判断超买超卖、结合 MACD 识别趋势反转等。课程将通过实际案例，帮助学生掌握如何运用 TA-Lib 进行金融数据分析与策略优化。

在数据爬取与分析过程中，学生将学习并使用多种 Python 工具。Pandas 用于数据清理与预处理，处理缺失值、异常值和数据转换；NumPy 提供高效的数值计算支持；Matplotlib 和 Seaborn 用于数据可视化，创建折线图、柱状图、热力图等多种图表。特征工程阶段，Scikit-learn 提供丰富的工具用于特征选择与提取；建模与预测阶段，Scikit-learn 的机器学习算法（如回归、分类、聚类）将被广泛应用。最后，通过 Jupyter

Notebooks，学生将进行交互式数据分析并生成可视化报告，全面掌握数据处理与分析的完整流程。 理论课时数：4；实验课时数：8

（二）教学单元对课程目标的支撑关系

课程目标 / 教学单元	1	2	3	4	5	6	7	8	9
第一单元 Python 基础及网络爬虫	√	√	√	√	√	√	√	√	√
第二单元 金融数据采集与预处理	√	√	√	√	√		√	√	
第三单元 静态网页采集	√	√	√	√			√	√	
第四单元 数据存储	√			√	√		√	√	
第五单元 JavaScript 与动态内容			√		√		√	√	
第六单元 Python 多进程技术	√				√		√		
第七单元 金融数据的分析与处理			√			√		√	√

（三）课程教学方法与学时分配

教学单元	教与学方式	考核方式	学时分配		
			理论	实践	小计

第一单元 Python 基础及网络爬虫	讲授法	大作业、平时作业（含实验报告）	2	4	6
第二单元 金融数据采集与预处理	讲授法、直观演示法、讨论法、理实一体化	大作业、平时作业（含实验报告）	2	4	6
第三单元 静态网页采集	讲授法、直观演示法、讨论法、理实一体化	大作业、平时作业（含实验报告）	2	4	6
第四单元 数据存储	讲授法、直观演示法、讨论法、理实一体化	大作业、平时作业（含实验报告）	2	4	6
第五单元 JavaScript 与动态内容	讲授法、直观演示法、讨论法、理实一体化	大作业、平时作业（含实验报告）	2	4	6
第六单元 Python 多进程技术	讲授法、直观演示法、讨论法、理实一体化	大作业、平时作业（含实验报告）	2	4	6
第七单元 金融数据的分析与处理	讲授法、直观演示法、讨论法、理实一体化	大作业、平时作业（含实验报告）	4	8	12
合计			16	32	48

（四）课内实验项目与基本要求

序号	实验项目名称	目标要求与主要内容	实验时数	实验类型
1	数据库设计与实现	本实验将结合课程相关内容，设计一个数据库，确保量化系统的正常运行。	9	③
2	并发系统设计与实现	在现代互联网系统中，用户数量和请求量的急剧增加对系统的性能和稳定性提出了更高的要求。高并发架构设计可以有效地解决这些问题，提供良好的用户体验并支持高可用性的运行。本实验在理解高并发架构设计要素的同时，运用常用技术栈，完成一个简易的并发系统。	12	③
3	爬取金融数据	本实验将利用爬虫技术，实现对互联网中的公开金融信息进行收集、清洗和存储。	18	③
4	TA-Lib 库的运用	本实验将使用 TA-Lib 模块计算相关的指标。例如 MA、SMA、WMA、MACD、ATR 等，可应用于组装量化策略，以求达到收益最大化	9	③

实验类型：①演示型　②验证型　③设计型　④综合型

四、课程思政教学设计

在课程教学中，将通过理论讲解、案例分析与实践操作相结合的方式，帮助学生深刻理解数据安全与隐私保护的重要性。首先，在理论部分，课程将系统介绍《网络安全法》《数据安全法》《个人信息保护法》等相关法律法规，明确数据采集、存储与分析过程中的法律边界与道德要求。例如，讲解非法爬取数据的法律后果及滥用数据对个人隐私与社会安全的危害。其次，通过案例分析，展示因数据泄露、非法爬取或滥用数据引发的重大事件（如某金融机构因数据泄露导致用户信息被恶意利用），引导学生思考数据安全的重要性及其对社会的影响。最后，在实践环节中，课程将设置数据采集与处理的模拟场景，要求学生严格遵守数据使用规范，设计合法、合规的爬虫程序，并在数据存储与分析过程中采取加密、匿名化等安全措施。通过理论与实践的结合，学生将树立正确的数据使用观念，增强法律意识与职业道德，成为负责任的数据从业者。

同时，通过小组讨论与辩论，引导学生探讨数据使用中的伦理问题（如数据所有权、数据共享与隐私保护的平衡），培养其批判性思维与社会责任感。通过这些教学设计，学生不仅能够掌握数据安全的技术手段，还能够在未来的工作中自觉遵守法律法规，维护数据安全与社会公共利益。

课程将通过融入国家战略、行业前沿与创新实践，帮助学生树立科技报国与创新驱动发展的使命感。首先，在课程导论部分，将结合我国金融科技的发展现状与国际化竞争态势，阐述金融科技对国家经济发展的重要性。例如，介绍我国在移动支付、区块链、人工智能等领域的领先地位，以及金融科技在服务实体经济、推动普惠金融中的作用。通过具体案例（如蚂蚁集团、腾讯金融科技等企业的创新实践），激发学生的民族自豪感与使命感。其次，在课程内容中，将融入创新驱动发展的理念，鼓励学生探索新技术、新方法。例如，在金融数据分析与策略设计环节，引导学生结合我国金融市场特点，设计具有创新性的量化策略；在数据存储与处理环节，鼓励学生研究分布式存储与计算技术，提升数据处理效率。

此外，课程将设置创新实践项目，要求学生以团队形式完成一个金融科技领域的创新课题。通过项目实践，学生不仅能够将所学知识应用于实际问题，还能够培养团队协作能力与创新精神。同时，课程将组织学生参加金融科技领域的创新创业大赛或行业论坛，拓宽视野，了解行业前沿动态，激发其投身金融科技创新的热情。通过这些教学设计，学生将深刻认识到金融科技对国家战略的意义，树立科技报国的远大理想，努力成为推动我国金融科技国际化与竞争力提升的中坚力量。

五、课程考核

总评构成	占比	考核方式	课程目标									合计
			1	2	3	4	5	6	7	8	9	
X1	40%	大作业评审	10	10	10	10	10	10	20	10	10	100
X2	30%	实验报告			10	30	20	30	10			100
X3	30%	课堂表现考勤	20	20	10					20	30	100

评价标准细则（选填）

考核项目	课程目标	考核要求	评价标准			
			优 100-90	良 89-75	中 74-60	不及格 59-0
1						
X1						
X2						
X3						
X4						
X5						

六、其他需要说明的问题
