

结对编程对《计算机科学导论》课程学生考试成绩影响的实证研究*

唐国明**

赖晶晶

李文烨

香港科技大学(广州) 信息枢纽 数据科学
与分析学域, 广州 511455香港科技大学(广州) 教育科学学院 数
理教育基柱, 广州 511455

摘要 针对《计算机科学导论》课程初学者编程实践中面临的认知负荷高、即时反馈缺失等问题,以某高校10个平行Lab班约300名学生为对象,采用准实验设计考察结对编程对考试成绩的影响。研究基于期中考试成绩进行1:1最近邻匹配以消除初始差异,并综合运用动态分层、匹配双重差分 and 协方差分析考察干预净效应。结果显示,结对编程对整体成绩净效应不显著,但呈现明显的异质性,对高基础学生具有显著的相对保护效应,能有效降低其在复杂编程任务中的成绩回撤风险;同时,AI-Human与Human-Human模式在学业增益上并无显著差异。研究表明,结对编程更适合作为一种差异化教学支持策略,且AI工具在辅助编程教学中具有替代人类同伴的潜力。

关键词 计算机导论, 结对编程, 准实验设计, 双重差分, 异质性效应, AI辅助教学

An Empirical Study on the Impact of Pair Programming on Examination Performance in an Introductory Computer Science Course *

Guoming Tang Jingjing Lai

Wenye Li

Data Science and Analytics Thrust, Information
Hub, The Hong Kong University of Science and
Technology (Guangzhou), Guangzhou, 511455,
ChinaPillar of STEM Education, College of
Education Science, The Hong Kong University
of Science and Technology (Guangzhou),
Guangzhou, 511455, China

Abstract—To address insufficient programming-practice support for novices in introductory computer science courses, this study examined the effect of pair programming on students' examination performance in a quasi-experimental classroom setting. Approximately 300 first-year students from ten parallel laboratory sections were included. Two sections received pair-programming interventions, including an AI-Human mode and a Human-Human mode, while the remaining sections followed conventional individual programming instruction. Based on midterm scores, 1:1 nearest-neighbor matching was used to construct balanced treatment and comparison samples. Dynamic stratification, matched difference-in-differences, and ANCOVA were then applied. The results indicate no statistically significant overall net effect of pair programming on final examination performance. However, a significant relative protective effect was observed among high-baseline students. No statistically significant difference was found between the AI-Human and Human-Human modes after controlling for baseline performance. These findings suggest that pair programming may be more useful as a differentiated support strategy than as a uniform intervention for all novice learners.

Keywords—introductory computer science, pair programming, quasi-experimental design, difference-in-differences, heterogeneous effects, AI-assisted teaching

1 引言

《计算机科学导论》是高校计算机基础教育的重要入门课程,承担程序设计思维、问题求解能力和计算思维启蒙等任务。对初学者而言,语法规则、控

制结构、调试方法和问题抽象往往同时构成学习负荷;若课堂实践环节缺乏及时反馈与持续支持,学生容易在编程初期形成挫败体验,并进一步影响后续课程学习。

结对编程是一种协作式程序开发方式,通常由两名学习者围绕同一任务共同完成编码。其中,Driver负责实际编码与运行,Navigator负责理解题意、检查逻辑、发现错误并提出修改建议,二者在任务推进过程中定期轮换角色。除工业界的广泛应

***基金资助**: 本文得到港科大(广州)实践研究项目(ROP)(HKUST(GZ)-ROP2025031)、广州市高校“新工科、新医科、新师范”建设项目(2025XGYS012)资助。

****通讯作者**: 唐国明 guomingtang@hkust-gz.edu.cn

用外,既有研究表明,结对编程在计算机基础教育中也往往具有良好的实践效果[1,3],结对学习方法与模式对提升编程能力具有积极意义[2]。

近年来,人工智能(AI)工具被逐步引入编程学习场景,为学生提供解释、调试、提示和反馈等支持。考虑到结对编程方法在计算机科学课堂中具有的重要价值[4,5],把两者有机结合,在《计算机科学导论》课堂中考察AI辅助结对模式的实际效果,既具有教学实践意义,也有助于理解AI工具在大班编程教学中的适用边界,强化计算思维作为编程教育的核心能力培养目标[6]。

现有相关研究多基于小规模课程案例、实验室情境或线上学习环境展开,针对真实大班课堂中多个平行班的准实验研究仍相对不足。尤其在非随机分班条件下,若仅比较实验组与对照组期末均分,容易受到初始能力差异、教师变量和均值回归等因素干扰。为提高结果解释的稳健性,本研究引入1:1最近邻匹配、动态分层、匹配双重差分法和协方差分析,系统考察结对编程在《计算机科学导论》课程中的成绩效应与异质性特征。

基于上述背景,本文重点回答三个研究问题:第一,与传统独立编程教学相比,结对编程是否能够显著提升学生在程序设计考试中的整体表现;第二,结对编程的干预效果是否因学生先验基础不同而呈现异质性;第三,引入AI工具辅助的人机(AI-Human)结对模式与传统的人人(Human-Human)结对模式在成绩表现上是否存在显著差异。

2 研究设计与教学干预

2.1 研究对象与分组

本研究对象为某高校《计算机科学导论》课程的一年级新生,覆盖10个平行Lab班,总样本规模约300人。研究采用准实验设计,其中2个Lab班设为实验组:L08班采用AI-Human结对模式,L09班采用Human-Human结对模式;其余8个Lab班作为对照组,采用传统独立编程教学方式。

实验组在课堂实验环节实施结对编程干预。每次实验课开始后,教师首先进行任务讲解并明确角色分工;随后学生以两人为一组完成指定编程任务,每完成一道题后进行Driver与Navigator角色轮换;课程结束前完成代码提交与反思反馈。

2.2 教学任务与实施流程

本研究的教学干预涵盖Python编程的多个核心主题,包括但不限于基本语法、控制流结构、函数定义与使用、递归算法、API调用、模块导入以及常用第三方库(如NumPy、Pandas等)的应用。实验任务

设计遵循由浅入深的原则,从基础概念逐步过渡到综合应用,确保学生能够系统掌握Python编程的核心技能。

在Human-Human模式中,全班学生两人一组,每道题后交换Driver与Navigator角色,要求双方围绕解题思路、程序逻辑和调试过程进行持续沟通,配合完成编程任务。在AI-Human模式中,学生将AI工具设定为结对编程中的Navigator,要求AI通过提问、解释思路或给出提示进行引导,而不是直接输出完整代码;学生与AI进行若干轮互动,并保留典型对话片段用于课后反思。

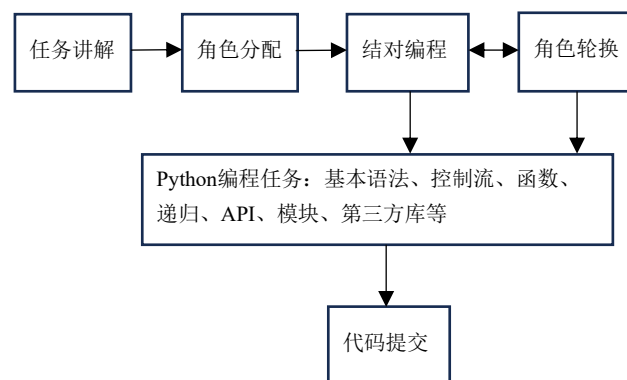


图1 结对编程教学流程图

图1展示了实验组课堂干预的基本流程。分为三个阶段:第一阶段(0-10分钟)为任务讲解与角色分配,明确Driver负责编码、Navigator负责检查逻辑的分工。第二阶段(10-50分钟)为结对编程实施,要求学生围绕Python编程任务完成编码、运行和调试,任务类型涵盖基本语法、控制流结构、函数定义、递归算法、API调用、模块导入及第三方库应用等,每完成一题后轮换角色。第三阶段为代码提交与反思反馈,学生完成所有题目后提交Python源程序文件。

2.3 数据来源

本研究使用课程形成性与终结性评价中已经产生的成绩数据。期中考试成绩作为干预前的基线数据,期末考试成绩作为干预后的结果数据。为保证不同考试权重下成绩具有可比性,数据处理阶段将期中与期末成绩统一换算为百分制,并剔除期中、期末或实验成绩缺失以及明显异常的记录。

数据分析阶段以编码后的学生标识进行样本关联与统计建模,研究结果仅以群体统计形式报告,不呈现可识别个体身份的信息。教学干预嵌入课程正常实验环节,不改变课程考核规则;AI-Human模式要求学生将AI作为提示型Navigator使用,避免直接替代学生完成完整代码。

3 数据处理与分析模型

3.1 样本匹配与动态分层

由于实验组与对照组来自不同平行班，学生初始基础可能存在差异。为降低非随机分组带来的选择偏差，本研究以期中成绩为协变量，在对照组样本池中为实验组学生寻找基础水平最接近的对照样本，采用 1:1 最近邻匹配构建均衡样本。匹配后形成实验组 59 人、匹配对照组 59 人的分析样本。基线均衡性检验显示，两组期中成绩差异极小 ($p=0.9924$, $SMD=-0.0018$)，说明匹配后两组在干预前成绩起点上具有较高可比性。

为考察不同先验基础学生的异质性效应，本文以全体学生期中成绩分布为依据计算分位数，并将匹配样本划分为高基础组 (Top 25%)、中高组 (25%–50%)、中低组 (50%–75%) 和低基础组 (Bottom 25%)。相比固定分数线，动态分层更能反映同一课程群体内部的相对基础差异。

3.2 统计检验与模型设定

本文首先采用配对样本 t 检验比较实验组学生期中与期末成绩变化，用于描述干预前后的组内纵向变化；其次采用独立样本 t 检验检验 L08 班与 L09 班在期中成绩上的基线一致性；最后采用匹配双重差分模型作为主检验，并使用协方差分析进行稳健性检验。显著性水平设定为 $\alpha=0.05$ 。

匹配双重差分模型设定为：

$$Score_{it} = \beta_0 + \beta_1 \cdot Time_t + \beta_2 \cdot Treat_i + \beta_3 \cdot (Time_t \times Treat_i) + \varepsilon_{it} \quad (1)$$

其中， $Time_t$ 表示考试阶段，期中为 0、期末为 1； $Treat_i$ 表示是否属于实验组；交互项系数 β_3 表示结对编程相对于对照组时间趋势的净干预效应； ε_{it} 表示误差项（随机扰动）。协方差分析模型设定为：

$$Final_i = \alpha_0 + \alpha_1 \cdot Treat_i + \alpha_2 \cdot Midterm_i + \varepsilon_i \quad (2)$$

用于在控制期中成绩后检验实验组与对照组期末表现差异。

在 AI-Human 与 Human-Human 模式比较中，本文进一步构建：

$$Final_i = \gamma_0 + \gamma_1 \cdot Mode_i + \gamma_2 \cdot Midterm_i + \varepsilon_i \quad (3)$$

其中 $Mode_i$ 表示结对模式。该模型用于检验在控制基线成绩后，两种结对模式对期末成绩的影响是否存在显著差异。需要说明的是，未发现显著差异并不等同于严格统计等效，本文仅在本研究样本和测量条件下作谨慎解释。

4 结果分析

4.1 实验组纵向成绩变化

配对样本 t 检验结果表明，实验组整体在干预前后呈现统计学显著的绝对成绩增长，期末较期中均分提升 4.00 分 ($p=0.0095$)。分层结果显示，低基础组纵向提升幅度最大，均分提升 8.40 分 ($p=0.0166$)；中低组提升 5.80 分但未达到 0.05 显著性水平；高基础组和中高组的纵向变化均不显著。具体结果见表 1。

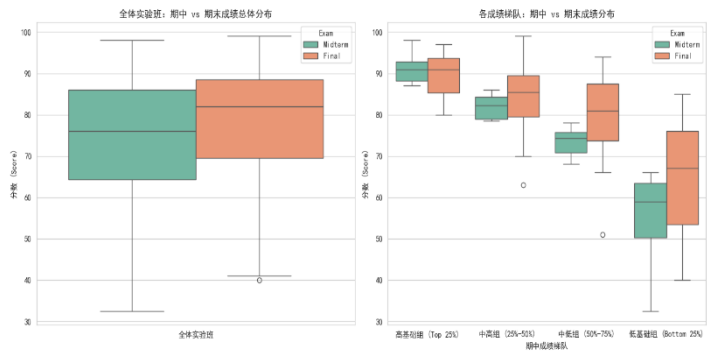


图2 实验组期中与期末成绩箱线图分布

表1 实验组各分层纵向成绩变化检验表

群体分层	样本量 N	期中前测均 分	期末后测均 分	绝对增 量	p值	显著性
总体实验组	59	73.90	77.90	4.00	0.0095	**
高基础组 (Top 25%)	14	90.90	89.40	-1.50	0.3775	ns
中高组 (25%—50%)	12	82.00	83.30	1.30	0.6640	ns
中低组 (50%—75%)	14	73.40	79.20	5.80	0.0707	ns
低基础组 (Bottom 25%)	19	56.70	65.10	8.40	0.0166	*

注：**表示 $p<0.01$ ，*表示 $p<0.05$ ，ns表示不显著。

4.2 匹配双重差分与稳健性检验

匹配双重差分结果显示, 结对编程在整体样本层面对期末成绩未产生统计学显著的净效应(DiD交互项 $p=0.6841$), 协方差分析结果同样不显著($p=0.4078$)。这一结果表明, 若仅从全样本均值角度评价, 结对编程并未表现出稳定的额外成绩增益。

进一步的分层检验显示, 仅高基础组出现显著净效应, DiD估计值为 5.68 分($p=0.0273$), 且 ANCOVA 处理效应检验同样达到显著水平($p=0.0191$)。中高组、中低组和低基础组均未显示显著净干预效应。具体结果见表 2。

表 2 结对编程干预净效应与稳健性检验表

群体分层	DiD净效应(分)	DiD交互项p值	显著性	ANCOVA处理效应p值	显著性
总体检验	1.48	0.6841	ns	0.4078	ns
高基础组	5.68	0.0273	*	0.0191	*
中高组	-1.75	0.6681	ns	0.6523	ns
中低组	3.21	0.4562	ns	0.3998	ns
低基础组	-0.84	0.8685	ns	0.8239	ns

注: *表示 $p<0.05$, ns表示不显著。

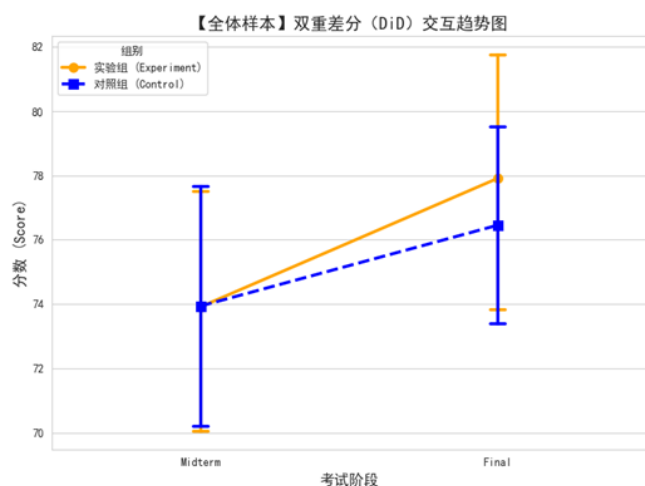


图 3 全体样本双重差分 (DiD) 交互趋势图

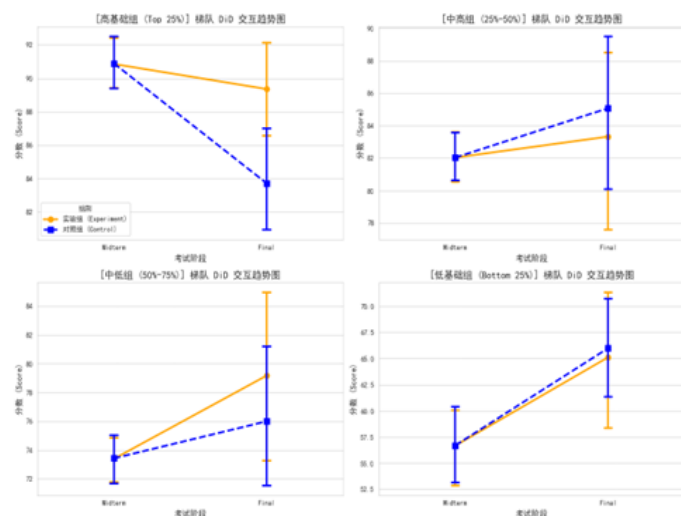


图 4 各成绩分层双重差分 (DiD) 交互趋势图

4.3 AI-Human 与 Human-Human 模式比较

为检验实验组内部两种结对模式的差异, 本文比较 L08 班 (AI-Human) 与 L09 班 (Human-Human) 的期中基线与期末表现。独立样本检验显示, 两班期中成绩无显著差异 ($p=0.9213$), 说明两种模式比较具有较好的基线一致性。在绝对成绩增量上, AI-Human 组均值增量为 5.60 分, Human-Human 组均值增量为 2.45 分; 但在控制期中成绩后, ANCOVA 结果显示两组期末表现差异未达到统计学显著水平 ($p=0.2799$)。具体结果见表 3。

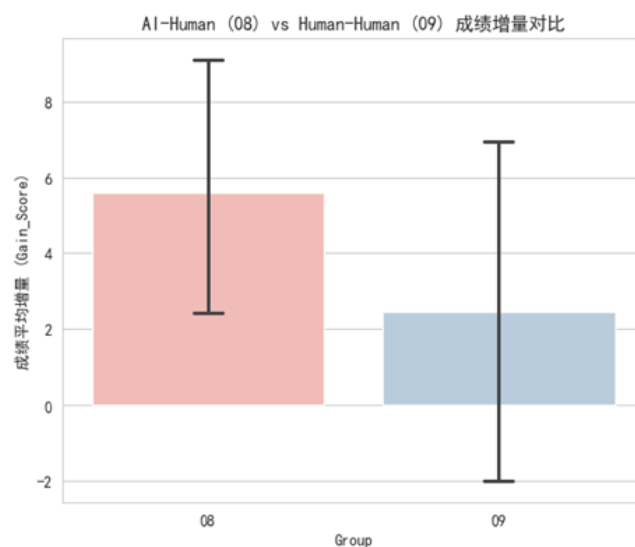


图 5 AI-Human 与 Human-Human 结对模式成绩增量对比直方图

表 3 不同结对模式成绩效能对比表

结对模式	班级	样本量	期中前测均分	前测p值	绝对成绩增量均值	ANCOVA p值	显著性
AI-Human	L08	29	73.72	0.9213	5.60	0.2799	ns
Human-Human	L09	30	74.10	—	2.45	—	—

注： *表示 $p<0.05$ ，ns表示不显著。

5 讨论

5.1 结果讨论

第一，整体净效应不显著提示结对编程并非在所有学生群体中均能产生稳定成绩增益。实验组纵向成绩显著提升，但匹配双重差分与 ANCOVA 结果均表明，全样本层面的净干预效应不显著。这说明学生随课程推进产生的自然适应、复习强化和考试结构变化等因素可能解释了部分成绩增长，仅凭实验组自身前后测提升不能直接推断教学干预有效。

第二，高基础组的显著净效应更适合解释为相对保护效应而非简单提分效应。高基础组实验学生期中均分已达较高水平，纵向成绩没有显著上升，存在明显天花板效应。但与匹配对照组相比，实验组在期末阶段表现出更稳定的成绩保持，DiD 结果显示其相对优势达到 5.68 分。该结果表明，在更复杂的期末任务中，结对编程可能通过同伴审查、逻辑复核和调试支持，帮助高基础学生降低成绩回撤风险。

第三，中低基础群体的纵向提升不能简单归因于结对编程。低基础组在实验组内提升幅度最大，但 DiD 结果不显著，说明该群体的成绩增长可能更多来自课程推进、练习积累或均值回归，而非结对编程带来的额外净收益。中等基础组结果波动较大，提示单一结对规则可能难以满足不同学生的学习需求，后续应考虑任务难度、配对策略等差异化设计。

第四，AI-Human 与 Human-Human 模式未见显著差异，应谨慎理解为未观察到显著差别，而不能直接等同于严格统计等效。该结果至少说明，在本研究样本与任务条件下，将 AI 作为提示型 Navigator 并未显著弱于传统同伴 Navigator，该结果与近期人机协同编程教学的相关综述发现相一致[7]，也未显著优于传统结对模式。未来研究需要进一步引入 AI 对话质量、提示策略、代码修改轨迹和学生自我效能等过程性数据，以解释 AI 辅助结对为何对不同学生产生差异化影响。

5.2 研究贡献

本文的实践贡献在于，将结对编程与 AI 辅助编程支持置于真实大班计算机导论课堂中进行比较，

较好贴近高校基础编程教学的实际场景。研究结果表明，评价结对编程教学效果时不宜只看总体均分，而应关注学生基础差异带来的异质性效应。对于高基础学生，结对编程可能主要发挥稳定成绩、降低失误和促进反思的作用；对于中低基础学生，则需要结合更明确的任务脚手架和过程指导。

本文的方法贡献在于，在非随机分班的教育实证研究中结合最近邻匹配、动态分层、匹配双重差分 和 ANCOVA 稳健性检验，为课堂干预效果评估提供了较为透明的分析路径。与单纯前后测或期末均分比较相比，该路径能够更好地区分自然成长趋势与干预净效应。

5.3 研究局限

本研究仍存在以下局限。首先，本文主要使用考试成绩作为结果变量，缺乏代码提交日志、编译错误记录、角色轮换时长、同伴讨论质量和 AI 对话质量等过程性数据。因此，研究能够识别成绩层面的差异，但对结对协作内部机制的解释仍不充分。

其次，经过匹配与分层后，各基础梯队样本量有限，部分分层样本仅为 12—19 人，统计功效受到限制。未来研究应扩大样本规模，延长干预周期，并在更多课程和不同教师条件下重复验证，以提升结论的外部效度。

6 结束语

本文围绕《计算机科学导论》课程中的编程实践支持问题，采用准实验设计考察结对编程对学生考试成绩的影响。研究结果显示，结对编程在整体样本层面未产生统计学显著的额外成绩增益，但其效果存在明显异质性：高基础学生在匹配双重差分与 ANCOVA 检验中均表现出显著相对保护效应；中低基础学生虽有纵向成绩提升，但未显示显著净干预效应。AI-Human 模式与 Human-Human 模式在控制基线成绩后未见显著差异，说明 AI 作为提示型 Navigator 具有进一步探索价值，但其教学效果仍需结合过程数据和学习机制进行更细致检验。

后续研究可从三个方向推进：一是引入代码提交、错误修复、讨论记录和 AI 对话片段等过程性数据，打开结对编程干预的黑盒；二是依据学生先验基

础设计差异化结对策略, 为高基础学生提供更具挑战性的开放任务, 为中低基础学生提供更强的结构化脚手架; 三是结合问卷与访谈, 考察结对编程对编

程自我效能、认知负荷、协作意愿和 AI 素养的影响, 从而形成更完整的大班编程教学干预评价框架。

参考文献

- [1] 闫晟, 徐彬. 程序设计基础课程中结对编程实践与效果评估[J]. 计算机教育, 2020(4): 17-20
- [2] 张玥, 杨叶博, 杨洋, 等. 基于结对编程的结对学习方法与模式研究[J]. 电脑知识与技术, 2017, 13(14): 111-113
- [3] 黄家玮, 钟萍, 季桂树. 计算机程序设计课程配对学习模式教学研究[J]. 计算机教育, 2014(24): 66-69
- [4] 刘峰. 结对编程软件方法的研究与应用[J]. 电子设计工程, 2010, 18(7): 78-80
- [5] Braught G, Wahls T, Eby L M. The Case for Pair Programming in the Computer Science Classroom[J]. ACM Transactions on Computing Education, 2011, 11(1): 1-16
- [6] Wing J M. Computational Thinking[J]. Communications of the ACM, 2006, 49(3): 33-35
- [7] 姚善良, 徐燕萍, 郭乃琳, 等. 人机协同下大语言模型驱动的程序设计教学创新综述[J]. 计算机技术与教育学报, 2026, 14(1): 83-88