

基于Python软件的统计课程教学探索与实践

何鑫

西华大学理学院, 四川 成都

收稿日期: 2025年4月15日; 录用日期: 2025年5月16日; 发布日期: 2025年5月23日

摘要

随着人工智能技术的快速发展, 传统统计学软件(如SPSS、SAS和R)在本科统计课程教学中的局限性日益显现。本文以Python软件为基础, 探讨其在统计学课程教学中的实践方案。相比传统统计软件, Python凭借其开源生态和跨学科优势, 更适用于辅助统计学核心内容的教学。通过Python编程实践, 学生不仅能加深对统计知识的理解, 还能提升实际应用能力。此外, 本文引入大语言模型DeepSeek作为教学辅助工具, 验证其在Python代码生成与修改、数据分析支持以及个性化答疑等方面的赋能作用。该模式有望显著提升学生的编程能力和统计应用水平, 为统计教学改革提供新的思路。

关键词

Python, 统计教学, DeepSeek, 教学改革, 实践教学

Exploration and Practice on Teaching Statistics Courses Based on Python Software

Xin He

School of Science, Xihua University, Chengdu Sichuan

Received: Apr. 15th, 2025; accepted: May 16th, 2025; published: May 23rd, 2025

Abstract

With the rapid development of artificial intelligence technology, the limitations of traditional statistical software (such as SPSS, SAS, and R) in undergraduate statistics education have become increasingly evident. This paper explores a practical approach to integrating Python into statistics courses. Compared to traditional statistical software, Python's open-source ecosystem and interdisciplinary advantages make it more suitable for supporting the teaching of core statistical concepts. Through Python programming practice, students can not only deepen their understanding of statistical knowledge but also enhance their practical application skills. Furthermore, this paper intro-

duces the large language model DeepSeek as a teaching assistant tool, examining its role in Python code generation and modification, data analysis support, and personalized tutoring. This approach has the potential to significantly improve students' programming and statistical application abilities, providing new insights for statistical education reform.

Keywords

Python, Statistics Teaching, DeepSeek, Teaching Reform, Practical Teaching

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

在数据科学和人工智能技术快速发展的时代背景下，统计课程的教学模式也在不断的变化。统计软件作为统计课程教学和应用的重要辅助工具，其教学的方式直接影响着学生学习理论知识和解决实际问题的能力。然而，传统统计课程中广泛使用的 SPSS、SAS 等软件，由于操作封闭、功能受限，已经难以满足现代统计分析对跨学科性和灵活性的需求[1]-[3]。例如，学生可以在 SPSS 中通过点击菜单完成回归分析，但却不清楚背后的计算过程；而 R 语言虽然功能强大，但复杂的语法对初学者来说并不友好。相比之下，Python 以各种丰富的开源库(如 Numpy、Pandas、SciPy、StatsModels 等)和相对简单的语法，成为统计课程教学的理想选择[4] [5]。与此同时，大语言模型的出现也为统计软件课程教学提供了新的可能性。例如，DeepSeek 和 ChatGPT 这样的工具，可以帮助学生修复代码错误、生成示例数据，甚至解释统计概念，从而降低理论学习和代码编程的难度[6]-[8]。本文围绕统计课程教学，分析 Python 与传统统计软件的主要区别，并设计适用于课堂教学的 Python 实践案例。同时，我们探讨如何合理利用当下流行的大语言模型 DeepSeek，帮助学生更高效地学习编程和统计知识，最终构建一种结合 DeepSeek 辅助的“学 - 练 - 问”一体化教学模式，为统计教学改革提供新思路。

2. Python 在统计课程教学中的探索与优势分析

在统计课程教学中，软件工具的选择直接影响学生对统计理论的理解和实践能力的培养，常见的统计软件包括 SPSS、R、SAS 和 Python，它们在功能、易用性、计算能力和应用场景上各具特色，适用于不同的教学需求[9]。以下对这些软件的优劣势进行简要分析，以明确 Python 在统计课程教学中的优势。

SPSS 主要用于社会科学和商业分析设计，具有直观的图形化界面，便于初学者快速上手。学生无需编写代码即可通过可视化界面完成概率分布、假设检验、回归分析等基本统计方法。因此，它在心理学、教育学、市场研究等领域的课程中应用广泛。然而，SPSS 也存在明显局限性。首先，它的脚本功能有限，难以进行复杂的定制化分析，同时在大规模数据处理方面能力不足。其次，作为商业软件，SPSS 价格昂贵。此外，它主要专注于传统统计分析，缺乏数据科学和机器学习能力，在现代数据分析教学中的适用性受限。R 以统计计算和数据分析见长，广泛用于科研和统计建模。其生态系统丰富，拥有如 ggplot2 (数据可视化)、lme4 (线性混合模型)、caret (机器学习)等强大的扩展包，在高级统计课程(如多元分析、时间序列分析)中优势明显。然而，R 的语法相对复杂，对初学者不够友好，学习曲线较陡。此外，与 Python 相比，R 的代码调试和管理较为困难。同时，在大规模数据处理方面，R 的计算性能略显不足。SAS 是功能强大的商业统计软件，在金融、医药和市场分析等领域应用广泛。其数据处理能力强，特别适用于

大规模数据分析，并能高效执行复杂统计计算。同时，SAS 以稳定性和安全性著称，是许多企业和政府机构的首选统计工具。然而，SAS 也存在使用门槛高的问题。首先，商业授权费用昂贵，不利于本科教学的普及。其次，SAS 的编程语言较为冗长，代码可读性较差，学习曲线陡峭。尽管 SAS 也支持界面化操作，但与 SPSS 相比仍然不够直观，难以吸引非统计专业的学生。此外，SAS 生态系统相对封闭，在统计在数据科学和机器学习等领域的扩展性较弱。

相较于上述统计软件，Python 兼具统计分析、数据处理和机器学习能力，适用于现代统计教学。Python 语法简洁，适合初学者快速掌握，同时支持 NumPy、Pandas、SciPy、StatsModels 等库，能够高效实现各种统计分析和建模任务[10]-[12]。此外，Python 提供强大的可视化工具(如 Matplotlib、Seaborn、Plotly)，可生成高质量的数据图表，提升分析的直观性[13]。Python 的另一大优势在于其开放性和跨学科适用性，不仅可用于统计学课程，还可拓展至数据科学、人工智能和自动化等领域，使学生具备更广泛的技能。同时，Python 的开源特性降低了学习成本。尽管在某些高级统计建模上可能不如 R，但其综合能力、易用性和可拓展性，使其成为本科统计教学的理想选择。

在统计课程教学中，合理的教学设计对于 Python 的有效实施至关重要。首先，在入门阶段，课程需要讲授 Python 的基础语法、数据结构和编程思维，并结合简单的数据分析案例，让学生在实践中熟悉 Python，提高编程兴趣和信心。在统计的理论教学中，应结合 Pandas、SciPy、StatsModels 等库，讲解描述统计量、假设检验、回归分析等内容，并利用 Matplotlib、Seaborn 等工具进行数据可视化，帮助学生更直观地理解统计概念，培养数据分析能力。此外，课程可引导学生完成小型数据分析项目，提升自主学习和问题解决能力。对于有更高需求的学生，可进一步拓展至机器学习和数据挖掘领域，使其掌握更高级的数据分析方法，并应用于实际问题。在教学模式上，可采用案例驱动和项目导向的方法，基于真实数据集进行分析，增强学生的实践能力。同时，利用 Jupyter Notebook 进行互动式教学，让学生即时运行代码并观察结果，提高学习效果。此外，Python 拥有丰富的开源生态，教师可结合 MOOC、GitHub 等在线资源，帮助学生拓展知识与技能。Python 凭借强大的统计分析能力、开源特性和跨学科优势，为统计课程教学提供了高效的解决方案。通过实践导向的教学方式，Python 不仅帮助学生掌握统计技能，还能培养编程思维和数据科学能力，为其未来的学术研究和职业发展奠定坚实基础。

3. Python 在统计课程教学中的应用：方法与实践

在统计课程教学中，Python 不仅是一种编程工具，更是连接理论与实际应用的重要桥梁。它不仅能用于基本的统计计算，还能帮助学生理解复杂的数据关系、进行可视化分析，甚至拓展至机器学习和人工智能领域[14]。在不同教学阶段，Python 的应用涵盖多个层面，包括基础统计量计算、数据可视化、假设检验、回归分析、时间序列分析及机器学习入门。通过精心设计的教学场景，教师可以引导学生从理论走向实践，培养统计思维和数据分析能力，并提升其解决实际问题的能力。以下介绍几个 Python 在统计课堂教学中的典型应用场景。

D) Python 作为计算与实验工具。在统计学基础教学中，均值、中位数、四分位数、方差、标准差等基本统计量是学生首先要掌握的概念。传统教学通常依赖公式推导和手工计算，但随着数据量的增加，手工计算效率低且容易出错。引入 Python 可以使计算过程更高效，并帮助学生快速比较不同数据集的特征。在实际教学中，教师可以让学生计算不同样本的数据，并观察数据分布的变化。例如，当数据存在极端值时，均值和中位数如何变化？方差和标准差如何影响数据的分布？通过 Python 的计算实验，学生可以更直观地理解统计量的意义，而不再局限于公式推导。例如，假设有一个包含 60 个样本的数据集 df，统计了不同性别和地区的大学生月消费支出。教师可以编写如下 my_summary 函数，计算各组的统计量：

```
def my_summary(df, col = ['性别']):
```

```

df_res = pd.DataFrame()
df_res['n'] = df.groupby(col)['月生活费支出'].count()
df_res['平均数'] = df.groupby(col)['月生活费支出'].mean().round(3)
df_res['中位数'] = df.groupby(col)['月生活费支出'].median()
df_res['标准差'] = df.groupby(col)['月生活费支出'].std().round(4)
df_res['全距'] = df.groupby(col)['月生活费支出'].apply(lambda x: x.max()-x.min())
df_res['变异系数'] = df.groupby(col)['月生活费支出'].apply(lambda x: x.std()/x.mean())
df_res['偏度系数'] = df.groupby(col)['月生活费支出'].skew()
return df_res

```

利用此函数，可以计算不同分组的消费统计量，例如：`df1 = my_summary(df)`计算按性别分组的月消费数据；同样可以利用 `df2 = my_summary(df, ['地区'])`计算地区分组的月消费数据；也可以 `df3 = my_summary(df, ['性别', '家庭所在地'])`同时按地区和性别进行分组。通过 Python 的统计计算函数或自编函数，学生不仅能更好地理解数学公式，还能在编程实践中增强学习兴趣，提高编程能力和问题解决能力。

II) Python 数据可视化。数据可视化是统计分析的重要部分，能够帮助学生直观理解数据分布、趋势及潜在关系。Python 提供了 Matplotlib、Seaborn、ggplot 等可视化库，可用于绘制直方图、箱线图、散点图等，使数据特征更加直观可感知。在课堂教学中，教师可以引导学生绘制不同数据集的直方图，并让他们思考：数据是否呈正态分布？是否存在偏态或异常值？此外，还可以结合箱线图进行异常值检测，让学生观察去除异常值后的数据变化。Python 常用的数据可视化函数包括：

```

import matplotlib.pyplot as plt
import seaborn as sns
plt.barh() # 水平或垂直条形图
plt.pie() # 饼图
sns.histplot() # 直方图
sns.kdeplot() # 核密度图
sns.boxplot() # 箱型图
sns.violinplot() # 小提琴图

```

这些工具可用于展示统计数据、概率分布、线性回归等内容，使学生能更直观地理解统计理论。

III) Python 实现假设检验。在传统统计教学中，学生通常需要学习 z 检验、t 检验、卡方检验和方差分析等方法。然而，这些方法涉及较复杂的数学推导，容易让学生感到困惑。Python 提供了 `scipy.stats` 和 `statsmodels` 等库，能够帮助学生快速执行假设检验，使他们专注于结果的解读和应用。例如，以下代码演示了两个独立小样本(x1 和 x2)均值差异的 t 检验：

使用 Python 内置函数：

```

from statsmodels.stats.weightstats import ttest_ind
xbar1 = x1.mean()
xbar2 = x2.mean()

```

总体方差相等：

```
t, p_value, df = ttest_ind(x1 = x1, x2 = x2, alternative = 'two-sided', usevar = 'pooled')
```

总体方差不相等

```
t, p_value, df = ttest_ind(x1 = x1, x2 = x2, alternative = 'two-sided', usevar = 'unequal')
```

结合假设检验公式 t 检验函数：

```

from scipy.stats import t
def ttest(x1, x2, flag = True):
    n1, n2 = len(x1), len(x2)
    mean1, mean2 = np.mean(x1), np.mean(x2)
    std1, std2 = np.std(x1, ddof = 1), np.std(x2, ddof = 1)
    if flag:
        df = n1 + n2 - 2
        sp2 = ((n1 - 1)*std1**2 + (n2 - 1)*std2**2)/df
        t_value = (mean1 - mean2)/np.sqrt(sp2*(1/n1 + 1/n2))
    else:
        up = (std1**2/n1 + std2**2/n2)**2
        down = (std1**2/n1)**2/(n1 - 1) + (std2**2/n2)**2/(n2 - 1)
        df = up/down
        t_value = (mean1 - mean2)/np.sqrt(std1**2/n1 + std2**2/n2)
    p_value = 2 * (1 - t.cdf(abs(t_value),df))
    return t_value, p_value, df
t2, p_value2, df2 = ttest(x1, x2, flag = True)

```

在实际教学中，教师可以让学生使用 Python 内置函数与手写函数进行对比，以加深对假设检验理论的理解。

4. 大语言模型为 Python 统计教学赋能

随着人工智能技术的快速发展，大语言模型，如 ChatGPT、DeepSeek 等，在教育领域的应用日益广泛，尤其在 Python 统计教学中发挥了重要作用。其不仅能够提供即时的代码示例和解释，还能辅助数据分析、优化学习路径，并提升学生的自主学习能力[15] [16]。通过引入这些智能工具，Python 统计教学变得更加高效、互动性更强，并能满足不同层次学生的学习需求。本文以 DeepSeek 为例，通过具体应用场景说明大语言模型对 Python 统计课程教学的帮助。

I) 提供代码示例与解释。在 Python 统计教学中，学生需要理解各种统计概念及其实现方法。DeepSeek 可以帮助学生快速生成代码，并提供详细解释。例如，在讲解线性回归时，教师可以引导学生使用 DeepSeek 生成 Python 代码，以帮助他们理解回归分析的实现过程。

教师可以向 DeepSeek 提出需求：“请用 Python 的 statsmodels 库实现一个简单的线性回归，并解释代码含义。” DeepSeek 给出的示例代码如下：

```

# 导入必要的库
import numpy as np
import statsmodels.api as sm
# 1. 生成模拟数据
np.random.seed(0) # 确保结果可重复
X = np.random.rand(100, 1) # 生成 100 个随机特征值(0~1 均匀分布)
y = 3 + 2 * X.flatten() + np.random.randn(100) # 生成带噪声的线性关系
# 2. 添加常数项(截距)
X = sm.add_constant(X) # 在特征矩阵前添加一列常数 1

```

```

# 3. 创建并拟合线性回归模型
model = sm.OLS(y, X) # 初始化 OLS 模型(普通最小二乘法)
results = model.fit() # 执行参数估计计算
# 4. 输出回归结果
print(results.summary())
# 5. 提取关键参数
print("\n 回归系数: ")
print(f"截距: {results.params[0]:.4f}")
print(f"斜率: {results.params[1]:.4f}")

```

代码解释: numpy 用于生成模拟数据: X: 100 个在 0~1 均匀分布的随机值作为特征; y: 基于 $3 + 2X$ 生成的线性关系, 添加了正态分布噪声(np.random.randn); sm.add_constant(): 为特征矩阵添加常数项列, 用于估计截距参数。模型构建与拟合: sm.OLS(y, X): 创建普通最小二乘回归模型, 第一个参数: 因变量 y; 第二个参数: 自变量(特征矩阵)X(需包含常数项); fit()方法: 执行参数估计计算, 返回包含所有结果的对象。结果输出: summary(): 输出完整的回归结果报告, 包含: 模型基本信息(R-squared, F-statistic 等)系数估计值及其统计检验结果其他诊断信息(Durbin-Watson, Omnibus 等)。关键参数提取: results.params: 数组形式返回参数估计值第一个元素: 截距项(对应添加的常数项)第二个元素: 自变量的系数。

上述代码展示了最基本的线性回归流程, 实际教学过程中教师可以基于此, 引导学生如何修改代码以适用于更广泛的场景。

II) 辅助数据分析与可视化。Python 统计教学的重要部分包括数据分析和可视化, 学生需要掌握如何使用 Pandas 和 Matplotlib 等工具处理数据。DeepSeek 可以帮助学生快速完成数据预处理, 并生成可视化代码。

示例: 使用 DeepSeek 生成箱线图(Boxplot)代码教师可以向 DeepSeek 提问: “如何使用 Python 绘制箱线图来展示数据的分布?” DeepSeek 给出的示例代码如下:

```

import matplotlib.pyplot as plt
import numpy as np
# 生成示例数据
np.random.seed(42)
data = [np.random.normal(0, std, 100) for std in [1, 2, 3]]
# 创建箱线图
plt.figure(figsize = (8, 6))
plt.boxplot(data,vert = True, # 垂直方向
            patch_artist = True, # 填充颜色
            labels = ['Group 1', 'Group 2', 'Group 3']) # 分组标签
# 添加样式
plt.title("Boxplot Example with Matplotlib")
plt.xlabel("Groups")
plt.ylabel("Values")
plt.grid(axis = 'y', linestyle = '--', alpha = 0.7)
plt.show()

```

该代码展示了如何对给定数据绘制箱线图, 课堂上教师可以进一步讲解如何通过对 DeepSeek 进行合

理提问来获取更详细的函数参数解释。

III) 修正学生代码并指出问题。在统计课堂上，学生经常会写出错误或低效的代码，导致计算结果不准确或程序无法运行。DeepSeek 可以帮助检测这些问题，并提供优化建议。

示例：修正学生编写的 t 检验代码。学生提交的代码如下：

```
import scipy.stats as stats
import numpy as np
# 生成数据
group1 = np.random.randn(100)
group2 = np.random.randn(100) + 1 # 均值不同的数据
# 计算 t 检验
t_stat, p_value = stats.ttest_ind(group1, group2, var_equal = False)
print("t 值:", t_stat)
print("p 值:", p_value)
```

老师可以向 DeepSeek 提问：“请检查我的 t 检验代码是否有错误？”以下是 DeepSeek 指出的问题：1) 方差齐性假设冲突：您生成的数据中，两组数据的方差实际相等(group1 和 group2 都是 np.random.randn 生成，标准差为 1) 但代码使用了 var_equal = False (即 Welch's t 检验)，该检验适用于方差不相等的情况；2) 统计学效率损失：当方差相等时，Student's t 检验(var_equal = True)的统计效能更高 Welch's t 检验在方差相等时仍可用，但会略微降低检验功效；3) 没有设置随机种子，np.random.randn()每次生成的随机数不同，运行结果会发生变化，影响实验的可复现性。DeepSeek 在修正代码的同时，还能培养学生良好的编程习惯，并提高他们对统计推断的理解能力。

5. 结论与展望

Python 在统计课程教学中不仅是编程工具，更是理论与实践的桥梁。通过 Python 软件，学生可高效计算统计量、可视化数据、执行假设检验，培养统计思维与数据分析能力。随着 ChatGPT、DeepSeek 等大语言模型的发展，统计教学变得更加智能化。大语言模型可协助生成代码示例、辅助数据分析、优化学习路径，并帮助学生修正代码，提高学习效率。未来，Python 与大语言模型的结合将推动统计教学向智能化、个性化发展，提升学生的数据分析与创新能力。

基金项目

西华大学人才引进项目(RX2400002043)。

参考文献

- [1] 郑珊珊, 杨林. “SPSS 统计软件应用”课程混合式教学改革研究[J]. 科技风, 2022(28): 125-127.
- [2] 张志成. R 软件在概率论与数理统计课程教学中的应用[J]. 河南工学院学报, 2020, 28(5): 78-80.
- [3] 刘平武, 周开兵. SAS 生物统计分析应用[M]. 北京: 化学工业出版社, 2024.
- [4] 茆诗松, 吕晓玲. 数理统计学[M]. 第 2 版. 北京: 中国人民大学出版社, 2016.
- [5] 贾俊平, 吴艳琳. 统计学——基于 Python [M]. 北京: 中国人民大学出版社, 2024.
- [6] 李全兴, 吴国林. 论人工智能的理解——以 DeepSeek-R1 等新兴人工智能为例[J]. 华南理工大学学报(社会科学版), 2025, 27(2): 9-18.
- [7] 唐琳, 张佳鑫, 徐照光. 大语言模型在计算机编程实践课程教学中的应用[J]. 计算机教育, 2025(2): 97-101, 106.
- [8] 鞠小林, 张艳梅, 王皓晨, 等. 基于大语言模型辅助教学的 Python 编程课程教学探索[J]. 计算机教育, 2024(9):

33-37.

- [9] 王尚九, 宋杰, 李少勇, 等. 面向能力培养的统计软件课程教学改革与探索[J]. 科教文汇, 2022(18): 79-82.
- [10] 丁毅涛, 任水利. 基于 Python 语言的数理统计课程应用改革探索——以应用统计学专业为例[J]. 科技风, 2021(28): 25-27.
- [11] 何小年, 段凤华. Python 在统计学课程实训中的应用探索[J]. 软件, 2021, 42(9): 32-34.
- [12] 郭鹏. Python 在应用统计学课程教学中的应用[J]. 现代信息科技, 2024, 8(14): 183-189.
- [13] 何鑫. Python 数据可视化在统计学课程教学中的应用[J]. 教育进展, 2025, 15(2): 650-657.
- [14] 石也牧. 例谈 Python 在概率论中的应用[J]. 科学技术创新, 2025(2): 117-121.
- [15] 宋敏毅. 基于大语言模型的 Python 编程教学实践探索——以循环结构为例[J]. 中国信息技术教育, 2025(6): 58-60.
- [16] 林志玮, 周尚泰, 周金兰. 大语言模型与课堂教学: 理论与实践探究[J]. 教育教学论坛, 2025(11): 149-152.