

“大雾实验工具”的开发

《程序设计进阶与实践》大作业报告

| | | | |
|---------|----------------------------|----|------------|
| 姓名（组长） | 孙旭磊 | 学号 | PB21000270 |
| 姓名（组员1） | 秦沁* | 学号 | PB21111630 |
| 姓名（组员2） | 赵弈 | 学号 | PB21000033 |
| 姓名（组员3） | 鲍政廷 | 学号 | PB21111741 |
| 姓名（组员4） | 张学涵 | 学号 | PB21000079 |
| 项目名称 | 大雾实验工具——绘制图像&计算不确定度&生成计算公式 | | |

1 项目需求分析

大物实验在[评课社区](#)、[知乎](#)等网站上一直饱受争议，有多名同学指出实验报告撰写耗时长、专业作图软件难以使用、Word 中打数学公式麻烦等问题。鉴于此，曾经有学长在开发过一款[大物实验数据处理工具](#)，这是非常好的创意。但是，这款软件入门成本太高，故了解它的人很少。

本小组开发的[大雾实验工具](#)是一款网页应用，无需安装任何软件，更不需要有编程基础，没有任何学习成本。本工具的目标用户是中国科学技术大学一本本科生，着力于解决其撰写实验报告时最耗时的三件事情，即“绘制图像”“计算不确定度”“在电脑上书写公式”。

当然，一些高级软件也能出色地完成上述的本工具的功能，如专业绘图软件 Origin, 专业计算软件 Matlab 等。但我们的项目不是去取代这些强大的软件，而是将它们本地化。这些软件功能繁多，故学习成本相对较高，但我们的软件为每一个大物实验都写了专门的处理工具，封装到只需要用户上传数据表格的程度。相比动辄几个 GB 的专业软件来说，我们的工具更加友好，更加便捷，更加有针对性——更加有效。

2 项目功能设计

2.1 总体功能说明

大雾实验工具是本组成员 2022 春季学期程序设计进阶与实践的大作业项目。本工具搭建于网页平台，支持任何设备自由访问。传入实验数

据后，本工具立刻完成绘制图像、计算不确定度、生成计算公式一系列操作，并将最终结果整理成一份 Word 文档，下载后即可直接使用。本工具支持一级大物的所有实验，大大提升了学生们撰写实验报告的效率。由于本工具只是将传入的实验数据进行自动分析，故不会造成抄袭、造假等学术不端问题。

2.2 具体功能点说明

使用本工具时，用户只需输入他们做实验时测量到的原始数据，而无需任何额外的计算处理，用户所要做的只有按照规定的格式上传 Excel 文档。本工具支持 `xlsx`, `csv` 等各种格式的数据表格。具体而言，每个实验都会有一张示例数据表供用户参考，如图 1 的界面所示。用户也可以直接下载示例数据，并直接在它的基础上进行修改。因此，本工具没有任何学习成本，是一款即点即用、免安装的简单轻应用。



图 1：“拉伸法测钢丝杨氏模量”的工具界面

绘制图像

本工具根据输入的数据以及实验原理，自动生成美观的实验图像，支持平滑去噪、数据拟合、双 y 图等多种图像生成需求，如图 2 所示。

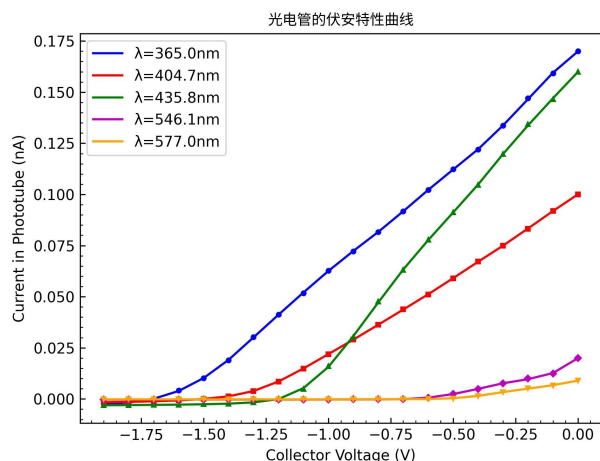


图 2: 平滑连接的光电效应伏安特性曲线

计算不确定度

本工具在生成的 Word 文档中渲染了各种公式，如图 3 所示。用户可以直观看到不确定度每一步的计算过程，并在自己的报告中直接使用这些算式与结果。

生成计算公式

在 Word 文档中除了有已经渲染好的公式外，我们还提供了它们的 L^AT_EX 源码，如图 4 所示。这极大方便了用 L^AT_EX, Markdown 等排版实验报告的用户，他们再也不需要手动敲入每一个算式了。

钢丝直径 d 的平均值:

$$\bar{d} = \frac{1}{n} \sum_{i=1}^n d_i = \frac{0.291 + 0.292 + 0.293 + 0.294 + 0.295}{5} \text{ mm} = 0.293 \text{ mm}$$

钢丝直径 d 的标准差:

$$\sigma_d = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (d_i - \bar{d})^2} = \sqrt{\frac{(0.291 - 0.293)^2 + (0.292 - 0.293)^2 + (0.293 - 0.293)^2 + (0.294 - 0.293)^2 + (0.295 - 0.293)^2}{5-1}} \text{ mm} = 0.0015811 \text{ mm}$$

钢丝直径 d 的 B 类不确定度:

$$\Delta_{B,d} = \sqrt{\Delta_{\text{仪}}^2 + \Delta_{\text{估}}^2} = \sqrt{0.004^2 + 0.005^2} \text{ mm} = 0.0064031 \text{ mm}$$

钢丝直径 d 的展伸不确定度:

$$U_{d,P} = \sqrt{\left(t_P \frac{\sigma_d}{\sqrt{n}}\right)^2 + \left(k_P \frac{\Delta_{B,d}}{C}\right)^2} = \sqrt{\left(2.78 \times \frac{0.0015811}{\sqrt{5}}\right)^2 + \left(1.96 \times \frac{0.0064031}{3}\right)^2} \text{ mm} = 4.6222 \times 10^{-3} \text{ mm}, P = 0.95$$

图 3: 不确定度计算的详细过程

```
0.293)^2+(0.295-0.293)^2}{5-1}}\mathrm{mm})\\
&=0.0015811\mathrm{mm}\\
\end{aligned}
$$$
```

钢丝直径 d 的 B 类不确定度:

```
$$$
\Delta_{B,d}=\sqrt{\Delta_{\text{仪}}^2+\Delta_{\text{估}}
)^2}=\sqrt{(0.004^2+0.005^2)}\mathrm{mm}=0.0064031\mathrm{mm}
$$$
```

钢丝直径 d 的展伸不确定度:

```
$$$
\begin{aligned}
U_{d,P}&=\sqrt{\left(t_P\frac{\sigma_d}{\sqrt{n}}\right)^2+\left(k_P\frac{\Delta_{B,d}}{C}\right)^2}\\
&=\sqrt{\left(2.78\times\frac{0.0015811}{\sqrt{5}}\right)^2+\left(1.96\times\frac{0.0064031}{3}\right)^2}\mathrm{mm}\\
&=4.6222\times 10^{-3}\mathrm{mm},P=0.95
\end{aligned}
```

图 4: 不确定度算式的 L^AT_EX 源码

2.3 功能点设计细节

本工具后端使用 Python 编写，使用的包与模块如表 1 所示。前端由 HTML 编写，并使用了 Flask Web 应用框架。

表 1: 本工具使用的全部 Python 包与模块

| Python 包或模块 | 用途 |
|--------------------|---|
| chardet | 检测用户上传的数据表格的编码 |
| collections | 通过 <code>namedtuple</code> 使代码更清晰 |
| Flask | Web 应用框架 |
| latex2mathml | L ^A T _E X 代码转换为 MathML 代码 |
| lxml | MathML 代码转 Office MathML |
| math, numpy | 不确定度数字运算 |
| matplotlib | 绘制物理图像 |
| os, random, shutil | 后台文件操作与管理 |
| pandas | 数据表格处理 |
| python-docx | 生成 Word 文档 |
| scipy | 数据拟合 |
| sympy | 不确定度符号运算 |
| time, threading | 定时删除生成的 Word 文档 |
| traceback | 打印运行错误以便调试 |

2.3.1 图像绘制

图像由 matplotlib 绘制。我们的规范如下：

- 面向绘图对象作图：`fig, ax = matplotlib.pyplot.subplots()`
- 设置副刻度为主刻度的一半，主刻度为默认：`ax.xaxis.set_minor_locator(matplotlib.ticker.AutoMinorLocator(2))`
- 刻度朝内：`matplotlib.rcParams["xtick.direction"] = matplotlib.rcParams["ytick.direction"] = "in"`

- 若一张图有且只有一组点线，则点使用红色 (`color="r"`)，线使用蓝色 (`color="b"`)，且线覆盖在点的上面；若一张图有多组点线，则同一组点线的颜色应当相同，并依次使用蓝(`b`)、红(`r`)、绿(`g`)、紫(`m`)、橙(`orange`)、青(`c`)。

- 点的类型使用实心圆 ("`o`")，若一张图有多组点线，则依次使用实心圆(`o`)、正方形(`s`)、上三角(`^`)、菱形(`D`)、下三角(`v`)、星号(`*`)。

- 线条粗细使用 `linewidth=1.5`，点的大小使用 `markersize=3`，可视数据量、数据组数适当调整，但应保持统一性。

- 绘制双 y 轴图使用 `matplotlib.axes.Axes` 对象的 `twinx()` 方法。

- 只有一组点线的图，一般不显示图例。

- 图像字体：SourceHanSansSC-Regular.otf

- 轴标签和标题中的物理量名称与单位应使用 L^AT_EX。

2.3.2 数据处理

无论使绘制图像时的线性拟合，还是计算不确定度的大小，都绕不开数据处理。我们利用 pandas, scipy, sympy 等包自主编写了 `calc.py` 应用程序接口，它提供以下函数：

科学计数法输出 `numlatex: (num: float,`

`prec: int = 5) -> str`

返回一个数的科学计数法形式的 L^AT_EX 代码

`num`: 要转成科学计数法的数字

`prec`: 有效数字位数（默认值：5）

不确定度计算 analyse: (data: pandas.DataFrame, symbol_Y: str = "Y", unit_m: str = "", delta_b1: float = 0., delta_b2: float = 0., unit_b: str = "") -> collections
symbol: str = "x", unit: str = "",
confidence_C: float = 3., confidence_P: float = 0.95) -> collections.namedtuple
("AnalyseData", ["average", "averagex", "averagex2", "sigma", "sigmax", "sigmax2", "delta_b", "delta_bx", "delta_bx2", "unc", "uncx", "uncx2"])

计算一组数据的平均值、标准差、不确定度

data: 要处理的一组实验数据

delta_b1: 仪器最大允差 $\Delta_{\text{仪}}$ (默认值: 0)

delta_b2: 估读最大允差 $\Delta_{\text{估}}$ (默认值: 0)

symbol: 数据的物理符号 (默认值: "x")

unit: 数据的单位 (默认值: "")

confidence_C: 置信系数 C (默认值: 3)

confidence_P: 置信概率 P (默认值: 0.95)

AnalyseData.average/averagex/averagex2: 数据的平均值/平均值计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

AnalyseData.sigma/sigmax/sigmax2: 数据的标准差/标准差计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

AnalyseData.delta_b/delta_bx/delta_bx2: 数据的 B 类不确定度/B 类不确定度计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

AnalyseData.unc/uncx/uncx2: 数据的展伸不确定度/展伸不确定度计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

最小二乘法线性回归 analyse_lsm: (data_X: pandas.DataFrame, data_Y: pandas.DataFrame, symbol_X: str = "X",

.namedtuple("AnalyseLsmData", ["m", "mx", "mx2", "b", "bx", "bx2", "r", "rx", "rx2", "s_m", "s_mx", "s_mx2", "s_b", "s_bx", "s_bx2"])

将一组数据用最小二乘法拟合成一条直线

data_X: x轴数据 (自变量数据)

data_Y: y轴数据 (因变量数据)

symbol_X: 自变量物理符号 (默认值: "X")

symbol_Y: 因变量物理符号 (默认值: "Y")

unit_m: 斜率的单位 (默认值: "")

unit_b: 截距的单位 (默认值: "")

AnalyseData.m/mx/mx2: 拟合直线的斜率/斜率计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

AnalyseData.b/bx/bx2: 拟合直线的截距/截距计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

AnalyseData.r/rx/rx2: 线性拟合的相关系数/相关系数计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

AnalyseData.s_m/s_mx/s_mx2: 拟合直线的斜率标准差/斜率标准差计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

AnalyseData.s_b/s_bx/s_bx2: 拟合直线的截距标准差/截距标准差计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

不确定度合成 analyse_com: (exp: str, varr: tuple = (), constt: tuple = (), unit: str = "", confidence_P: float = 0.95) -> collections.namedtuple("AnalyseComData", ["ans", "unc", "ansx", "uncx", "finalx",

"ansx2", "uncx2", "finalx2"]])

根据表达式计算物理量的值和其对应的不确定度

exp: 物理量计算表达式（字符串），为一个物理量=一些物理量（或常量）之积与之商的形式，如 $E=4\pi^2 l/T^2$ 代表 $E = \frac{4\pi^2 l}{T^2}$

varr: 物理量（元组），元组的每个元素均为元组，该子元组的第 1 个元素为物理量名，第 2 个元素为物理量值，第 3 个元素为其不确定度（默认值：()）

constt: 常量（元组），元组的每个元素均为元组，该子元组的第 1 个元素为常量名，第 2 个元素为常量值（默认值：()）

unit: 要计算的物理量的单位（默认值：""）

confidence_P: 置信概率 P （默认值：0.95）

AnalyseData.ans/ansx/ansx2: 表达式的计算值/表达式计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

AnalyseData.unc/uncx/uncx2: 合成不确定度的计算值/合成不确定度计算过程的 \LaTeX 代码/计算过程的 MathML- \LaTeX 代码

AnalyseData/finalx/finalx2: 最终结果（表达式计算值 \pm 合成不确定度计算值）的 \LaTeX 代码/最终结果的 MathML- \LaTeX 代码

2.3.3 文档生成

Word 文档由 python-docx 生成。我们的规范如下：

- 字体使用微软雅黑：`document.styles['Normal'].font.name = "微软雅黑"`
- 文档第一行是实验名称：
`document.add_paragraph(name())`
随后注明：

“【 \LaTeX 代码在下面，请向下翻阅】”

- 内容跨度较大的段落之间应当用一个空行。
- 文档中插入的数据一般保留 4 或 5 位有效数字：`"%.5g"%x`，线性拟合的相关系数 r 保留 8 位有效数字。
- 若某张图片正好在第 2 页开头，而第 1 页尾部有很多空白区域，为避免误解，应在第 1 页的最后一个段落之后注明“【本文档不只有一页，请向下翻阅】”。
- 插入表格使用 `docx.document.Document` 对象的 `add_table()` 方法。

鉴于不确定度的计算方法是固定的、算法化的，我们利用 `lxml` 等包自主编写了 `insert.py` 应用程序接口，这样只需调用几个函数，就可以在 Word 文档中完成数学算式的渲染与添加。具体函数如下：

\LaTeX 代码转 Word 对象

`latex_to_word: (latex_code: str) -> Any`

MathML- \LaTeX 转为可渲染的 Office MathML

`latex_code: MathML- \LaTeX 代码`

注：使用 `docx.xml.text.paragraph.CT_P` 对象的 `append()` 方法即可在 Word 中插入渲染好的 Office MathML 算式，通常的用法是
`document.add_paragraph()._element.append(latex_to_word(latex_code))`

不确定度计算插入 `insert_data: (docu: Any, name: str, data: AnalyseData, option: str) -> None`

在 Word 文档中插入不确定度计算的全部过程

`docu: 文档对象 docx.document.Document`

name: 该物理量的名称
data: 上述 `analyse()` 函数的返回值
option: 插入选项, "word" 表示插入渲染好的公式, "latex" 表示插入 L^AT_EX 源码

最小二乘法线性回归插入 `insert_data_lsm`:

(docu: Any, data: AnalyseData, option: str) -> None

在 Word 文档中插入线性拟合的全部过程

docu, option: 与 `insert_data()` 函数一致。

data: 上述 `analyse_lsm()` 函数的返回值

3 测试、运行情况

4 设计、开发过程中的难点

第一, 编程技术上的困难。无论是编程基础较薄弱、对实现的接口不够熟悉, 还是本地调试不充分, 都会造成程序错误, 影响本项目的质量。

第二, 对物理实验本身的了解不够。有的物理实验原理复杂、参考资料有限, 且实验课程标准在不断变化, 这会使程序生成的报告存在瑕疵, 影响本项目的声誉。

5 小组分工

孙旭磊: 组织策划、前端与前后端衔接、数据处理 API、公式插入 API、数据处理程序示例、制作海报、宣传片并宣传推广、修订实验报告, 5 个实验处理模块的制作。

秦沁: 宣传推广、修订实验报告, 4 个实验处理模块的制作。

赵奕: 技术与安全支持、修订实验报告, 4 个实验处理模块的制作。

鲍政廷: 制作海报、增订实验报告, 6 个实验处理模块的制作。

张学涵: 实验报告、增订数据处理 API, 5 个实验处理模块的制作。

6 总结与收获

本次实践不仅能帮助其他同学更轻松地完成大物实验报告, 我们自己也受益良多。

分工与合作 我们分工明确, 每个人的任务都有截止时间, 这使我们小组的进度有序推进。以往的经历中, 代码与相关工作往往都是独立完成, 代码规范与项目进程完全由自己安排。但是在这种大工程中, 相关代码需要符合规范, 需要与队友交接, 工作进度也要与队友进度相符。在这种分工体系下, 每个人都要完成自己的任务, 并顾及与他人的交互。

代码规范性 本次大作业中, 我们建立了统一的码风, 并制定了自主编写的 API 的使用说明。这样做一方面可以使得产品最终具有一致性——不同人写的代码能够基本一致; 另一方面也使得最终的检验与调整能够更加方便——规范的代码提高了代码的可读性, 降低了代码的审核成本。

软件开发技巧 在本次实践中, 我们使用 git 进行协作, 代码注释清楚, 帮助文档详细。这大大提高了我们的开发效率。

7 参考资料

A Appendix

此页不打印

未完成：斜率计算过程的 \LaTeX 与 MathML 代码，截距与 Pearson's r 的计算公式错误，最终结果有效位数：不确定度应只有 1 位。