

针对 VASP 的材料计算教程 为材料计算而生

Jiaqi Z.¹

2024 年 8 月 11 日

¹Copyright © 2024 Jiaqi Z. All rights reserved.

目录

I	Linux 基础	1
II	VASP 计算	3
1	电子性质	5
III	Python 与机器学习	7

前言

“为材料计算而生”，是抱着多大的觉悟说出这种话啊。这只是一本书，有办法背负其他人的人生吗？

真是满脑子只想着自己呢……¹

俗话说得好，好记性不如烂笔头，这句话在任何时候都显得格外贴切。尤其是在科研领域（特别是材料科学这样规范和流程化的学科），记录的重要性更是不言而喻。随着计算任务的不断增加，我们掌握的计算方法和参数也日益繁多。将这些知识、操作和方法记录下来，不仅能够帮助我们避免遗忘，还能在遇到类似问题时快速查找，而不必在海量的网络搜索结果中苦苦寻觅。

正是基于这样的考虑，我们结合自己科研团队在材料计算方面遇到的一些实际问题，整理编写了这本书。我们编写这本书的目的有两个：一是为了方便自己，在面对类似问题时能够迅速回忆起解决方案；二是为了通过集体的智慧，汇聚大家的方法和思路，以便在遇到新问题时能够迅速找到答案。

这本书的诞生，既是为了服务于材料计算，也是因材料计算而生。既然如此，我们为何不称它为“为材料计算而生”呢？

我们衷心希望这本书能够惠及更多的人，无论是我们团队的新成员，还是其他团队的老师或同学，都能从中获得帮助。

同时，我们也清楚地认识到自己的能力和知识是有限的，书中的内容难免会有疏漏或错误。我们诚挚地希望读者在使用过程中能够提出宝贵的意见和建议，或者分享你们的经验，共同促进我们的成长和进步。

最后，再次感谢您阅读并使用这本书。

Jiaqi Z.

2024 年 8 月青岛

¹ 以上内容改编自动漫《BanG Dream! It's MyGO!!!!!》中丰川祥子的台词

如何联系作者

可通过以下任意一种方式联系:

- GitHub 的 Issue, 这是最直接的方式²;
- email, 请发送邮件至 zhangjq00@sdust.edu.cn 或 zhangjq_sd@163.com

如何使用这本书

在使用时, 请按照如下方法:

1. 根据研究问题, 寻找合适的章节; 如果没有, 可以在 GitHub 上提交 Issue 或者贡献 Pull Request;
2. 在每一节开始, 会介绍本节的内容和知识点, 查看是否与你的研究问题符合; 如果不符合, 返回第 1 步重新查找新的章节;
3. 阅读这一节内容, 并试着针对自己的问题进行操作 (或简单检查自己操作是否正确)。如果报错或出现异常结果, 进行第 4 步; 如果成功, 进行第 5 步;
4. 在该节后面的“错误处理”部分, 会介绍如何处理报错或异常结果, 并给出解决方案。请查找是否有你需要的解决方案, 并尝试解决。如果已经解决, 进入第 5 步; 否则重新查找新的解决方案; 若所有解决方案都无法解决, 请提交 Issue 或者贡献 Pull Request;
5. 放下教程, 继续你的研究; 或者阅读这一章其他内容, 了解其他相关内容。

上述步骤可能 (也一定) 会重复许多次

关于本笔记的版权使用说明

- 本书可免费用于学习, 科研等非商业活动;

²GitHub 仓库地址: <https://github.com/JackyZhang00/Computational-Materials-Tutorial>

- 可以以非商业目的进行传播,但在传播过程中必须保证内容的完整性(截止到最新发布时,包括但不限于仓库内 Latex 源码, pdf 文件等.下同),需保证作者信息完整,不得进行修改;
- 本书不可用于任何商业用途(如确有需要,需联系作者);
- 除在 GitHub 仓库以 pull request 形式进行编辑修改外,不允许进行修改并公开传播私自修改版本(以 GitHub 仓库版本为标准版本);
- 本书著作权归作者(Jiaqi Z.)所有,其他进行创作的人员也可获得著作权,其他著作权所有者不得违反上述版权说明;
- 如因违反上述说明传播而造成不良影响,与作者和其他创作者无关,特此声明;
- 以上说明解释权归 Jiaqi Z. 所有,且如有后续更新,以 GitHub 仓库最新版说明为准.

创作者名单

感谢以下人员参与贡献了内容:

Jiaqi Z.

0.1 关于如何编写模板（教程）

本节作者：Jiaqi Z.

在本节，你将要学到：

- 一些基本的 L^AT_EX 语法
- 如何输入公式
- 如何插入代码
- 如何插入代码块

0.1.1 文章结构

请在编写正文内容时，以“节”（section）为单位创建 tex 文档，同时为方便引用，请在每个小节的后面按照 `\label{sec:节标题}` 的格式创建标签。

若需要添加小节，使用 `\subsection{小节标题}` 命令，同时类似于上方关于节标题标签的创建规范，以 `\label{subsec:节标题-小节标题}` 的格式创建标签，方便他人引用。

对于更小一级的小节（`\subsubsection{}`），对标签不作规范。事实上，我们不建议在引用时涉及到这一层级。通常涉及到小节即可。

***** 以下内容为注意 *****

注意：若你需要修改某一节（或小节）的标题，编译后需要确认是否与他人的标签产生冲突（这通常出现在他人提前按照原有格式引用后发生了修改，从而导致无法指向正确标签）。因此，你需要检查编译后文件是正确的，至少要求通过编译，在必要时需要修改他人代码当中的引用标签与新标签一致。

***** 以上内容为注意 *****

对于正文内容，请使用正常的 L^AT_EX 语法。例如，当你希望对某一段文字进行强调时，请使用 `\emph{}` 语句。例如，这是一句强调的话在代码中体现为 `\emph{这是一句强调的话}`。你并不需要关注具体的强调格式—L^AT_EX 会按照统一的格式进行编排。

当你希望分段时，使用空行即可。

对于条目,请在必要的时候使用`itemize`环境(没有顺序列表)或`enumerate`环境(有顺序列表)。在环境内使用`\item`进行编号。环境之间可以嵌套。例如:

- 列表 1
- 列表 2
 - 1. 列表 2-1
 - 2. 列表 2-2
- 列表 3

在代码中体现为:

```
\begin{itemize}
  \item 列表1
  \item 列表2
  \begin{enumerate}
    \item 列表2-1
    \item 列表2-2
  \end{enumerate}
  \item 列表3
\end{itemize}
```

0.1.2 一些特殊的环境

为统一教程格式,当你希望添加一段让读者注意的文字时,请使用环境`attention`例如,下面的语句

```
\begin{attention}
  当你写注意语句时,不需要在前面加任何符号。
\end{attention}
```

在编译后的结果为:

***** 以下内容为注意 *****

注意: 当你写注意语句时,不需要在前面加任何符号。

***** 以上内容为注意 *****

0.1.3 数学公式

当你希望添加数学公式时,请使用`equation`环境。同时,在使用`\label`语句进行标签注明时,请如同代码所示那样,添加“节标题”避免冲突且方便引用。

```
\begin{equation}
  \label{eqn:数学公式-1}
  a^2+b^2=c^2
\end{equation}
```

$$a^2 + b^2 = c^2 \quad (1)$$

在引用公式时,请使用如`\ref{eqn:数学公式-1}`方式进行交叉引用。请勿直接在正文内写编号以免出现引用错误。

***** 以下内容为注意 *****

注意: 在引用公式时, 所引用的公式尽量保持在本节内。同时, 为避免他人引用, 请在编写完成后尽量不要修改相关标签。

为避免公式删除导致的错误, 如确实需要引用其他章节的公式, 一个较合理的做法是将其他章节的公式在使用时拷贝至当前章节, 同时另起标签名。之后的引用限制在当前章节内。

***** 以上内容为注意 *****

如需要添加多行公式, 请使用`gather`环境或`align`环境。例如,

$$a^2 + b^2 = c^2 \quad (2)$$

$$\int_a^b f(x) dx = F(b) - F(a) \quad (3)$$

0.1.4 图片

当添加图片前, 请首先在相关文件夹内创建名为`fig`的文件夹, 在插入图片时如正常 L^AT_EX 代码插入即可。即使用下面的代码方式



图 1: 图片标题

```

\begin{figure}
  \centering
  \includegraphics[width=1\linewidth]{图片路径}
  \caption{图片标题}
  \label{fig:图片-标签}
\end{figure}

```

***** 以下内容为注意 *****

注意：在引用图片时，务必使用`\ref{fig:图片-标签}`进行引用。例如，图 1。不要使用“如上图”、“如下图”的表述形式，以免图片位置发生移动造成指代不明。

***** 以上内容为注意 *****

0.1.5 代码

代码使用`lstlisting`环境。

```
SYSTEM = 0
ISMEAR = 0
SIGMA = 0.05
NSW = 1
```

如果希望在代码左侧添加行号以示说明,请在引用环境的右侧添加`numbers=left`设置,即采用下面的代码:

```
\begin{lstlisting}[numbers=left]
..
```

得到效果如下所示

```
1 SYSTEM = 0
2 ISMEAR = 0
3 SIGMA = 0.05
4 NSW = 1
```

***** 以下内容为注意 *****

注意:若你希望在代码块中添加代码名称(例如文件名),可以使用`caption`选项进行说明,例如,下面的代码:

```
\begin{lstlisting}[caption=简单的 INCAR 文件]
..
```

实际编译后结果为

Listing 1: 简单的 INCAR 文件

```
SYSTEM = 0
ISMEAR = 0
SIGMA = 0.05
```

```
NSW = 1
```

***** 以上内容为注意 *****

0.1.6 添加索引

建议在编写过程中，为文中的程序关键字创建索引。为方便使用，可以使用命令`\keyword{}`。例如，在 VASP 讲解 SIGMA 函数时，可以使用下面的代码

```
\keyword{SIGMA}
```

从而在表述关键字的同时创建索引。同时，也可以使用`\keyword[]{}{}`的形式创建带有所属关系的索引，例如

```
\keyword[INCAR]{SIGMA}
```

便是创建了属于 INCAR 条目下的 SIGMA 索引。

***** 以下内容为注意 *****

注意：在编写时，鼓励使用索引方便他人根据关键字直接检索。在创建索引时，请提前确认现有的索引表是否有现成的索引。例如，

`\keyword{SIGMA}`和`\keyword[INCAR]{SIGMA}`会得到不同的结果。

同时，在编写过程中已经将输出和索引集成在一个命令中，你不需要特地再编写一个输出的命令如 `SIGMA\keyword{SIGMA}`，只需要使用`\keyword{SIGMA}`便可在输出 SIGMA 的同时创建索引。

***** 以上内容为注意 *****

对于一些特殊的内容，可能不希望给出索引（通常是命令或者文件名等），它们既不属于关键字，也不属于简单的英文单词。为了将它们区分，使用`\code{}`命令进行编写。例如，`\code{cd \OLDPWD}`的输出结果为 `cd $OLDPWD`

***** 以下内容为注意 *****

注意：在一些特殊的情况下（例如上面的例子），可能会包含 \LaTeX 本身的特殊符号（如 $\$$ 本身作为公式符号）。在输入时，应当使用反斜杠 `\` 作为转义字符。

***** 以上内容为注意 *****

Part I

Linux 基础

Part II

VASP 计算

Chapter 1

电子性质

Contents

1.1 能带	5
------------------	---

1.1 能带

Part III

Python 与机器学习

