

# 安徽师范大学

## 本科毕业论文

题    目：                    安徽师范大学  
                    本科毕业论文模板示例文档

学生姓名：          王子韵                    学号：          21111202046

院（系）：          数学与统计学院                    专业：          统计学

入学时间：                    二〇二一年  九月

导师姓名：          王佩君                    职称/学位：          副教授

导师单位：                    数学与统计学院

完成时间：                    二〇二五年  六月

## 标题

王子颀, 数学与统计学院

**摘 要:** 本文是中国科学技术大学本科毕业论文的  $\text{\LaTeX}$  模板。本模板作者为 XPS, 本示例文档作者为 heavenseven。2011 年由 wzj 进行了部分更新。

**关键词:** 中国科学技术大学 (USTC)、毕业论文、 $\text{\LaTeX}$  模板

## title

Zijie Wang, School of Mathematics and Statistics

**Abstract:** This article is a thesis template for undergraduate students of Univ.of Science and Technology of China. The template is written by XPS, and heavenseven is the author of this article. Latest update is finished in 2024 by wzj. The abstract chapter consists of cnabstract environment and the abstract environment. Pls replace these words with your own to complete your abstract.

**Keywords:** Univ. of Science and Technology of China (USTC), Thesis,  $\text{\LaTeX}$  Template

# 1 引言：CT<sub>E</sub>X 系统的安装和使用

本模板（ustcthesis.cls）是按照教务处的本科论文要求做出的 L<sup>A</sup>T<sub>E</sub>X 模板，作者为 XPS。

本文是使用上述模板生成的示例文档，目的在于帮助使用者熟悉该模板的使用方法，并且方便使用者学位论文的撰写。

## 1.1 系统要求，CT<sub>E</sub>X 安装

本文只针对 Windows 环境下 CT<sub>E</sub>X 套装。至于 Linux 使用者，我觉得你们有能力解决自己的问题。

如果电脑上尚未安装 L<sup>A</sup>T<sub>E</sub>X 系统，那么到 CTeX.org 下载最新完整版 CT<sub>E</sub>X 套装，并安装之。

此模板只支持 UTF-8 编码。将其他编码的文件转化为 UTF-8 的方法是：用记事本打开这些文件，然后点击文件—另存为—在最下方选择 UTF-8 编码。在 WinEdt 中首次保存时，选择 save as...，弹出的窗口中保存类型选择 UTF-8 即可。

## 1.2 WinEdit 界面介绍和使用

CT<sub>E</sub>X 套装试用 WinEdt 作为默认编辑器，什么都配置好了，很方便，推荐使用。

打开 CT<sub>E</sub>X 套装内的 WinEdt 编辑器，可以看到如图 1.1 的界面。

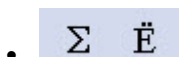
其中，中部的大块白色背景的区域就是键入文章内容的编辑区。编辑区上方的一条工具栏上有所有当前打开的文件的标签，单击即可跳转到对应文件。标签栏上方是面板，上面有一些很常用的工具按钮，鼠标悬停可以看到按钮名称。下面我们来看其中的几个。在使用本模板时，将会用到这些按钮。



新建和打开按钮。



单击左侧的按钮可设置主文件，单击右侧的取消主文件。



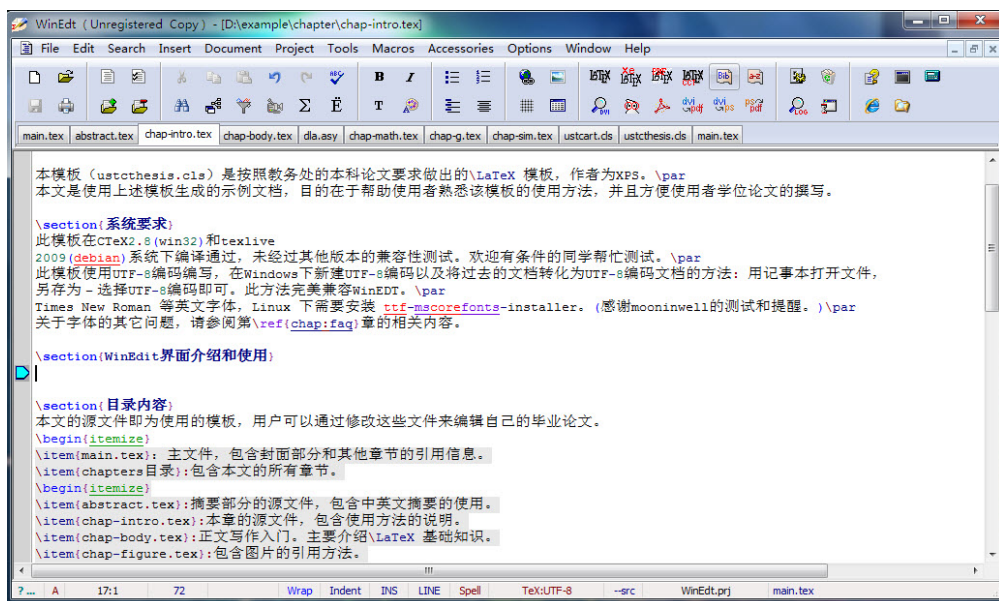


图 1.1 WinEdit 界面

单击可以分别打开 TeX SYMBOLS GUI(用于输入数学公式)和 ASCII 字符表。



编译通过之后，可以点工具栏上的 **Adobe** 按钮，打开预览。在 CTeX2.8 套装的 WinEdit 中，会弹出 SumatraPDF，在这个窗口中，看到不合意的地方可以双击回到源文件的相应位置，逆向搜索到原文中的对应位置进行修改，还是比较方便的。

### 例 1.3.1. *Hello World*

将以下内容复制到新文件中，保存为  $T_E X$  文件，编译，然后按 *Adobe* 按钮预览结果。

```
\documentclass{article}
\begin{document}
    Hello world!
\end{document}
```

为了得到正确的目录、交叉引用、参考文献等信息，需要编译两到三遍，正确的流程如下：

1.  $X_{\text{L}}\text{A}T_{\text{E}}X$  编译一次。
2. BibTeX 编译一次。
3.  $X_{\text{L}}\text{A}T_{\text{E}}X$  编译两到三次。

在本示例文档中，提供了 `make.bat`，双击之即可完成以上工作。

## 1.4 目录内容

本文的源文件即为使用的模板，用户可以通过修改这些文件来编辑自己的毕业论文。

- `main.tex`: 主文件，包含封面部分和其他章节的引用信息。
- `chapters` 目录: 包含本文的所有章节。
  - `abstract.tex`: 摘要部分的源文件，包含中英文摘要的使用。
  - `chap-intro.tex`: 本章的源文件，包含使用方法的说明。
  - `chap-body.tex`: 正文写作入门。主要介绍  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  基础知识。
  - `chap-figure.tex`: 包含图片的引用方法。
  - `chap-table.tex`: 包含表格的示例。

- chap-math.tex: 包含数学公式排版的基础知识。
- chap-app.tex: 附录。
- thanks.tex 包含致谢部分。
- figures 目录: 存放文章内所用的图像文件。
- bib/tex.bib: 参考文献信息。

需要特别说明的是, 这些文件名并不是固定的, 你可以新建一个 tex 文件, 例如 zhangsan.tex, 放在 chapters 目录下, 并且在 main.tex 中使用

```
\include{chapters/zhangsan.tex}
```

来引用之。当然你也可以重命名这些文件, 只要 include 中的文件名是存在的, L<sup>A</sup>T<sub>E</sub>X 总能找到这些文件的。

在你写作某一章节的时候, 你可能需要随时预览排版效果并 debug, 这时你可以在其他章节的\include命令前加上一个%, 这代表注释掉本行, 例如:

```
%%%%%%%%%%%%%
%% 正文部分
%%%%%%%%%%%%%
\mainmatter
  \include{chapter/chap-intro}
% \include{chapter/chap-figure}
% \include{chapter/chap-table}
% \include{chapter/chap-math}
% \include{chapter/chap-faq}
```

那么, 编译的时候就只编译未加% 的一章, 在这个例子中, 即本章。

理论上, 并不一定要把每章放在不同的文件中, 但是这种分章节写作、编译的方法有利于提高效率, 大大减少 debug 过程中的编译时间, 同时减小风险。

## 2 正文写作

本章中介绍了一些 L<sup>A</sup>T<sub>E</sub>X 不得不说的基础知识，包括 L<sup>A</sup>T<sub>E</sub>X 语句的基本概念，L<sup>A</sup>T<sub>E</sub>X 中的特殊字符，章节命令，空格、分行和段落等等。

### 2.1 L<sup>A</sup>T<sub>E</sub>X 语句

L<sup>A</sup>T<sub>E</sub>X 语句主要有命令、数据和注释组成。命令有控制词和控制符两种，其中控制词以右斜杠\开始，以非字母字符结束（如\textbackslash）；控制符以\开始，紧跟着一个非字母、数字的字符（如\;）。注释以% 开始，每行自% 之后的内容都不会被编译。

命令可能会有一些参数，如

```
\includegraphics[width=10cm]{1.jpg}
```

其中中括号括起来的是可选参数，当省略时，L<sup>A</sup>T<sub>E</sub>X 会根据默认值来选择；大括号内的是不可省略的参数。这个命令的含义见第4章。

### 2.2 L<sup>A</sup>T<sub>E</sub>X 中的特殊字符

#### 2.2.1 空格与回车

在 L<sup>A</sup>T<sub>E</sub>X 中，连续的空格被认为是一个空格。行首的空格通常被忽略。按下回车产生的断行也被认为是空格。

而当为了美观需要，不得不强行插入空格时，有很多选择，这里给出其中的一部分：

- ~，产生不可断行的半个空格；
- \ ，即\之后再加一个空格，产生 1/3 个字母 m 宽度空格；
- \hspace{1cm}，其中 1cm 可以随便替换，需要加长度单位，如 em，pt 等；
- \quad，一个字母 m 的宽度；
- \qquad，两个字母 m 的宽度；

一个回车会得到一个空格，而两个回车会得到一个空行，空行代表开始一个新段落。连续的空行被当作是一个空行。

### 2.2.2 引号

左右引号分别用不同的字符```（数字 1 左边的键）和`'`（通常所用的单引号）表示。双引号则是连续两个左引号 + 连续两个右引号。

我听见韩麦尔先生对我说：“唉，总要把学习拖到明天，这正是阿尔萨斯人最大的不幸。现在那些家伙就有理由对我们说了：‘怎么？你们还自己说是法国人呢，你们连自己的语言都不会说，不会写！……’不过，可怜的小弗郎士，也并不是你一个人的过错。”

我听见韩麦尔先生对我说：``唉，总要把学习拖到明天

图 2.1 引号

### 2.2.3 控制字符

以下这些是  $\text{\LaTeX}$  中的控制字符：

`# $ % ^ & _ { } ~ \`

它们被用作控制符，因而不能直接输入。要输入这些字符，分别可以用以下命令：

`\# \$ \% \^{} \& \_ \{ \} \sim{} \textbackslash`

可以看到，一般来说，在这些字符前加上右斜杠`\`，就可以正常输出。`\`除外，这是一个换行命令（见下一节）。

### 2.2.4 横线

一个短横线`-`代表连字符，两个短横线`--`会产生一个用于数字范围的短破折号，三个短横线`---`产生普通的破折号。而代表负数的负号则应该是`$-`。

## 2.3 控制字体样式

大部分时候， $\text{\LaTeX}$  会自动控制字样。当为了强调某些文字，需要改变字体时，以下是一些例子：



|                           |                   |                              |                  |
|---------------------------|-------------------|------------------------------|------------------|
| <code>\textrm{...}</code> | roman             | <code>\textbf{...}</code>    | <b>bold face</b> |
| <code>\textsf{...}</code> | sans serif        | <code>\textit{...}</code>    | <i>italic</i>    |
| <code>\texttt{...}</code> | typewriter        | <code>\textsl{...}</code>    | <i>slanted</i>   |
| <code>\emph{...}</code>   | <i>emphasized</i> | <code>\underline{...}</code> | <u>underline</u> |

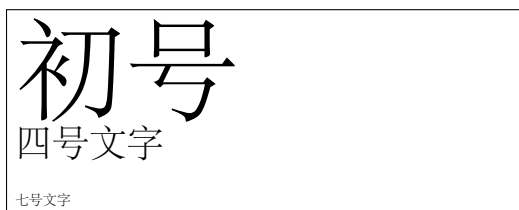
表中列出了大部分常用的文字样式选择命令，使用时将需要改变字样的文字放在命令后的大括号中作为参数即可。

## 2.4 改变字体大小

ustcthesi 模板中提供了字号选择命令：

```
\chuhao,\xiaochuhao,\yihao,\erhao,\xiaoerhao,...\qihao
```

一个简单的例子：



```
{\chuhao 初号}\\
{\sihao 四号文字}\\
{\qihao 七号文字}
```

图 2.2 字号选择

如例子中的那样，对于一段需要改变字号的文字，应当用大括号将其包围起来，将字号选择命令置于大括号开头。这样做可以防止字号选择命令影响之后的文字。

正文中默认使用的字号是小四号。

## 2.5 控制空格，分行，分段，分页，章节

**分行** 这里我们说的是段内分行。在  $\text{\LaTeX}$  中，普通的回车会被视为空格，而分行的命令则是 `\\`。分行与分段的区别在于下一行的段首不会缩进，并且这两行之间的间距还是普通行间距，比段间距要小。

**分段** 一个空行意味着一段的结束和下一段的开始。也可以使用分段符：`\par`。

**分页** 使用 `\newpage` 命令强制分页。

**章节 L<sup>A</sup>T<sub>E</sub>X** 使用章节类命令控制文章结构。章节类命令可以自动生成对应的格式，并且会自动插入目录中。

对于本科毕业论文，从大到小的章节依次为：

```
\chapter, \section, \subsection, \subsubsection
```

章节命令的参数是该章节的名字。

## 2.6 环境

环境命令 (environment) 是一类特殊的 L<sup>A</sup>T<sub>E</sub>X 命令。通常，一个环境命令定义了环境内部的字体、对齐和缩进方式、自动编号等信息，用于方便用户输入公式、注释、引文等。一个环境命令如下：

```

    上文
    \begin{环境名}
        ...
        环境内的文字
        ...
    \end{环境名}
    下文

```

几个常用的环境有：

- center：居中
- quote：引文
- verbatim：原文照排，即不经过处理，直接按原样输出。
- enumerate：自动编号的有序列表
- itemize：枚举，带有项目符号的无序列表

### 例 2.6.1. quote 环境

quote 环境会自动设置缩进量和上下行距，如下图。

在 itemize 和 enumerate 环境中用 item 命令表示列举项。两者可以互相嵌套，如图2.3。  
关于其他数学环境以及公式的输入，请参阅第3章的相关内容。

唐伯虎曾经写道：

别人笑我太疯癫，  
我笑他人看不穿。

引自《桃花庵》。

唐伯虎曾经写道：

```
\begin{quote}
```

别人笑我太疯癫，\\我笑他人看不穿。

```
\end{quote}
```

引自《桃花庵》。

- fruit
  - apple
  - strawberry
- drinks
  - 1. coffee
  - 2. tea
- chocolate

```
\begin{itemize}
  \item fruit
  \begin{itemize}
    \item apple
    \item strawberry
  \end{itemize}
  \item drinks
  \begin{enumerate}
    \item coffee
    \item tea
  \end{enumerate}
  \item chocolate
\end{itemize}
```

图 2.3 itemize 和 enumerate

### 3 数学公式

数学公式需在数学环境中排版。数学环境一般分正文公式和显示公式。正文公式用 $\dots$ 表示，而显示公式则用 `equation,align` 等数学环境。简单的几个例子：

这是一个显示公式：

$$x + y = z \quad (3-1)$$

谢谢。

这是一个显示公式：

```
\begin{equation}
```

```
x+y=z
```

```
\end{equation}
```

谢谢。

图 3.1 显示公式

大家好， $e^{\pi i} + 1 = 0$  是我最喜欢的方程。

大家好， $e^{\pi i} + 1 = 0$  是我最喜欢的方程。

图 3.2 正文公式

常用的显示公式环境有：

- `equation`：最常用的公式环境。
- `displaymath`：与 `equation` 的差别在于不会被自动编号。  
`\begin{displaymath}\dots\end{displaymath}` 可以用 `\[...\]` 来代替。
- `align`：适用于排版多个需要对齐的公式。

#### 3.1 数学公式排版基础

上下标、希腊字母、加减乘除等常用符号：

|      | 命令                        | 例子                         | 代码  |
|------|---------------------------|----------------------------|---|
| 上标   | $\wedge$                  | $x^2$                      | <code>\$x^2\$</code>                      |
| 下标   | $_$                       | $x_2$                      | <code>\$x_2\$</code>                      |
| 希腊字母 | 对应的英文                     | $\pi, \rho, \sigma, \dots$ | <code>\$\pi, \rho, \sigma, \dots\$</code> |
| 点乘   | <code>\cdot</code>        | $x \cdot y$                | <code>\$x \cdot y\$</code>                |
| 叉乘   | <code>\times</code>       | $x \times y$               | <code>\$x \times y\$</code>               |
| 分数   | <code>\frac{}{}</code>    | $\frac{abc}{def}$          | <code>\$\frac{abc}{def}\$</code>          |
| 根号   | <code>\sqrt[n]{}{}</code> | $\sqrt{x}, \sqrt[3]{y}$    | <code>\$\sqrt{x}, \sqrt[3]{y}\$</code>    |
| 积分   | <code>\int</code>         | $\int, \iint, \iiint$      | <code>\$\int, \iint, \iiint\$</code>      |

表中，\$ 代表数学环境。在 equation 等环境中排版时，不再需要使用 \$ 符号。

更多的符号，请点击 WinEdt 面板上的  $\Sigma$  符号 (TeX Symbols GUI)，在弹出的面板中查找。

当控制符作用于多个字符时，需要用大括号将这些字符都括起来，如

| 例子                  | 代码                         |
|---------------------|----------------------------|
| $e^{\pi i} + 1 = 0$ | <code>e^{\pi i}+1=0</code> |
| $x_{12}^3$          | <code>x_{12}^3</code>      |
| $x_{12}^3$          | <code>{x_{12}}^3</code>    |

### 3.2 控制字体

本科学位论文对于公式内字体的使用有严格的规定。一般变量用白斜体，常量和特殊函数等用白正体，矢量用黑斜体，张量用黑正体。改变字体的命令如下表：

|      | 字体  | 例子           | 代码                          |
|------|-----|--------------|-----------------------------|
| 一般变量 | 白斜体 | $x$          | <code>\$x\$</code>          |
| 常量   | 白正体 | $c$          | <code>\$\mathrm{c}\$</code> |
| 矢量   | 黑斜体 | $\mathbf{B}$ | <code>\$\bm{B}\$</code>     |
| 张量   | 黑正体 | $\mathbf{H}$ | <code>\$\mathbf{H}\$</code> |

表中，\$ 代表数学环境。在 equation 等环境中排版时，不再需要使用 \$ 符号。

### 3.3 括号

中括号、小括号和竖线可以用键盘上对应的键来输入。大括号是  $\text{\LaTeX}$  中的控制符，必须用  $\text{\{...\}}$  来输入。态函数所用的尖括号是  $\text{\langle}$  和  $\text{\rangle}$ 。一个句点代表一个不显示的括号。

使用  $\text{\left}$  和  $\text{\right}$  可以自动控制括号的大小。使用时,将所需要用的括号紧跟在  $\text{\left}$  或  $\text{\right}$  后面就好。例如:

$$\left[ \frac{\left( \frac{abc}{def} + ghi \right) \cdot jkl}{mno} + pqr \right] \times st = 0$$

```
\[
\left[\frac{\left(\frac{abc}{def}+ghi\right)\cdot jkl}{mno}+pqr\right]
\times st=0
\]
```

图 3.3 括号。 $\text{\[}$ 和 $\text{\]}$  代表  $\text{displaymath}$  环境,下同。

使用括号时,不一定要用相同的括号来匹配。甚至可以用  $\text{\left.}$  这样不被显示括号来匹配正常括号,以达到排版目的。如

$$|\psi\rangle$$

```
\left|psi\right\rangle
```

$$\begin{Bmatrix} a \\ b \\ c \end{Bmatrix}$$

```
\left\{\begin{array}{c}a\\b\\c\end{array}\right.
```

图 3.4 不同括号的匹配

### 3.4 矩阵

输入矩阵一般使用  $\text{array}$  环境。

矩阵中使用  $\text{\&}$  来分位,  $\text{\&\&}$  来分行。 $\text{array}$  环境的参数指定了列的数量和每列的对齐方式。 $\text{c}$  代表一个居中对齐的列,  $\text{l}$  是靠左对齐,  $\text{r}$  则是靠右对齐。图3.5是一个简单的例子。

$\text{array}$  环境可以自嵌套, 达到输入复杂矩阵的目的。

$$\begin{bmatrix} C_{Xx} & C_{Yx} & C_{Zx} \\ C_{Xy} & C_{Yy} & C_{Zy} \\ C_{Xz} & C_{Yz} & C_{Zz} \end{bmatrix}$$

```
\left[\begin{array}{c c c}
C_{Xx} & C_{Yx} & C_{Zx}\\
C_{Xy} & C_{Yy} & C_{Zy}\\
C_{Xz} & C_{Yz} & C_{Zz}
\end{array}\right]
```

图 3.5 矩阵

### 3.5 对齐的多行公式

`align` 环境提供了书写多行公式，并指定其在某一位置对齐的功能，使用 `&` 来代表对齐点。如图3.6。其中，`\notag` 表示不对这一行公式进行编号。

$$\begin{aligned} a_0 &= (A_1 + A_2 + A_3)/3g_e\beta_e & (3-2) \\ b_0 &= [A_1 - (A_2 + A_3)/2]/3g_e\beta_e \\ c_0 &= (|A_2| - |A_3|)/2g_e\beta_e & (3-3) \end{aligned}$$

```
\begin{align}
a_0&=(A_1+A_2+A_3)/3\text{term}{g}_e\beta_e\\
b_0&=[A_1-(A_2+A_3)/2]/3\text{term}{g}_e\beta_e\notag\\
c_0&=(|A_2|-|A_3|)/2\text{term}{g}_e\beta_e
\end{align}
```

图 3.6 多行公式

### 3.6 定理，证明，和其他环境

`ustcthesi` 中提供了定理，证明等一些其他环境。环境和名称如下表：

| 环境名        | 中文名 | 环境名         | 中文名 |
|------------|-----|-------------|-----|
| theorem    | 定理  | lemma       | 引理  |
| example    | 例   | algorithm   | 算法  |
| definition | 定义  | axiom       | 公理  |
| property   | 性质  | proposition | 命题  |
| corollary  | 推论  | remark      | 注解  |
| condition  | 条件  | conclusion  | 结论  |
| assumption | 假设  | prove       | 证明  |



## 4 图形

### 4.1 浮动图形

$\text{\LaTeX}$  支持 `jpg` 和 `eps` 格式的图片。我们所用的编译方法支持 `jpg` 和 `eps` 等格式的图片。如果你已经有一篇 `word` 文档，想把其中的图全部导出，那么可以将它另存为 `html` 文件，这时会在同目录下找到一个文件夹，里面是所有用到的图片。

将插图放在 `figures` 文件夹中，想要引用的地方使用类似这样的命令插入图形文件：

```
\begin{figure}[!ht]
\centering
\includegraphics[width=0.2\textwidth]{ustc_logo_new.eps}
\caption{中国科学技术大学校徽（在页面中间）}
\label{fig:ustc1}
\end{figure}
```

结果如下：



图 4.1 中国科学技术大学校徽（在页面中间）

在上面这段命令中，可选参数 `[!ht]` 代表插图的位置。其中 `!` 让  $\text{\LaTeX}$  忽略审美标准，试图用最严格的标准来放置浮动图形；`h(ere)` 代表有限放在此处；`t(op)` 代表如果此处放不下，那么放在下一页页首。

`width=0.2\textwidth` 代表图形的宽度是文字宽度的 0.2 倍，你也可以使用其他长度单位，如 `12cm`，`4.5in` 等等。

`caption` 命令的参数代表图的名称，或者说注解。`label` 的参数用于交叉引用，见下节。

## 4.2 交叉引用

通过使用交叉引用功能，我们可以定位那些  $\text{\LaTeX}$  自动编号的内容，比如浮动图形、表格、章节等等。其中，`label` 和 `ref` 的参数是引用的名字，可以随意，但须保持一致。

引用图4.1  
引用第1章

引用图`\ref{fig:ustc1}``\`  
引用第`\ref{chap:introduction}`章

图 4.2 交叉引用示例

## 4.3 并列的子图

使用 `subfigure` 命令，一个例子：



(a) sf 1



(b) sf 2



(c) sf 3



(d) sf 4

图 4.3 `subfigure` 使用示例。

```
\begin{figure}[!hbt]
\centering
\subfigure[sf 1]{
\includegraphics[width=0.2\textwidth]{ustc_logo_new.eps}\label{f:1}}
\subfigure[sf 2]{
\includegraphics[width=0.2\textwidth]{ustc_logo_new.eps}\label{f:2}}
\subfigure[sf 3]{
\includegraphics[width=0.2\textwidth]{ustc_logo_new.eps}\label{f:3}}
\subfigure[sf 4]{
\includegraphics[width=0.2\textwidth]{ustc_logo_new.eps}\label{f:4}}
\caption{\label{f:s}subfigure使用示例。}
\end{figure}
```

分两行放：

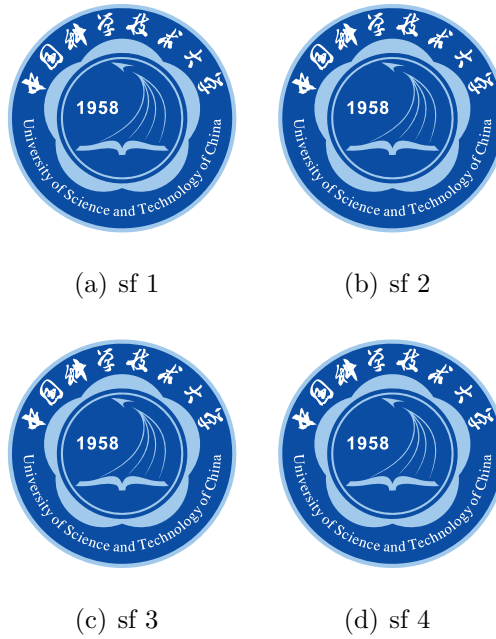


图 4.4 subfigure 使用示例。

```
\begin{figure}[!hbt]
\centering
\subfigure[f 1]{
\includegraphics[width=0.2\textwidth]{ustc_logo_new.eps}\label{f:1}}
\subfigure[f 2]{
\includegraphics[width=0.2\textwidth]{ustc_logo_new.eps}\label{f:2}}\\
\subfigure[f 3]{
\includegraphics[width=0.2\textwidth]{ustc_logo_new.eps}\label{f:3}}
\subfigure[f 4]{
\includegraphics[width=0.2\textwidth]{ustc_logo_new.eps}\label{f:4}}
\caption{\label{f:1}分行的字图使用示例。}
\end{figure}
```

## 5 表格

标准三线表：

表 5.1 三线表格示例

| 一列 | 两列 | 三列 | 四列 |
|----|----|----|----|
| 1  | 2  | 3  | 4  |
| 2  | 3  | 4  | 5  |
| 3  | 4  | 5  | 6  |

```
\begin{table}[!ht]
\label{tab:table3}
\caption{三线表格示例}
\centering
\begin{tabular}{c c c c}
\whline 一列 & 两列 & 三列 & 四列 \\
1 & 2 & 3 & 4 \\
2 & 3 & 4 & 5 \\
3 & 4 & 5 & 6 \\
\whline
\end{tabular}
\end{table}
```

与 figure 一样，table 环境也是浮动体，常用 !ht 来表明位置。caption、label 的意义都与 figure 环境相同。

而画表格的 tabular 环境则与 array 环境几乎相同，只是在每行之间可以用 \hline 和 \hline 这些命令来画横线。c 代表一个居中对齐的列，l 代表靠左对齐，r 代表靠右对齐。有竖线的例子：

表 5.2 表格示例 2

| 一列 | 两列 | 三列 | 四列 |
|----|----|----|----|
| 1  | 2  | 3  | 4  |
| 2  | 3  | 4  | 5  |
| 3  | 4  | 5  | 6  |

```
\begin{table}[!ht]
\label{tab:table2}
\caption{表格示例2}
\centering
\begin{tabular}{Ic|c|c|cI}
\whline  一列 & 两列 & 三列 & 四列  \\\hline
1 & 2 & 3 & 4  \\\hline
2 & 3 & 4 & 5  \\\hline
3 & 4 & 5 & 6  \\\hline
\end{tabular}
\end{table}
```

即竖线是|，粗竖线是大写字母I，横线是\hline，粗横线是\whline。

## 6 参考文献的使用

本模板使用 bibtex 来管理参考文献。在 BibTeX 中，典型的参考文献信息如下：

```
@book{deng:01,
  author =      {{邓建松,~彭冉冉,~陈长松}},
  title =      {\LaTeXe{}}~科技排版指南},
  publisher =   {科学出版社,~书号:~7-03-009239-2/TP.1516},
  year =       {2001},
  address =     {北京},
}
```

其中，第一行的 **book** 指明了文献的类别；**deng:01** 是引用时的标号，可以随意。在大括号之内则指明了参考文献信息，大家可以望文生义。在正文中参考文献的方法如下：

引用示例 [? ]

引用示例 \cite{deng:01}

引用时还有许多选择，如下：

|                               |                  |
|-------------------------------|------------------|
| \citet{deng:01}               | [? ]             |
| \citep[p.~99]{deng:01}        | [? , p. 99]      |
| \citep[e.g.] []{deng:01}      | [e.g. ? ]        |
| \citep[e.g.] [p.~99]{deng:01} | [e.g. ? , p. 99] |
| \citeauthor{deng:01}          | ?                |
| \citeyear{deng:01}            | ?                |

## 7 FAQ

这里是我们已经收到的一些常见问题以及一些一般的 Debug 技巧. 对于更多的信息或者关于本模板和文档的任何疑问, 请移步瀚海星云  $\text{T}_{\text{E}}\text{X}$  讨论区。

**Question:** 从哪里下载  $\text{C}_{\text{T}}\text{E}_{\text{X}}$  套装? 应该下载哪一个  $\text{C}_{\text{T}}\text{E}_{\text{X}}$  套装?

**Answer:** 在 $\text{C}_{\text{T}}\text{E}_{\text{X}}$  官方网站可以下载, 进入网站后点击网页右边的“项目”栏目中“最新稳定版”, 即可进入下载链接. 对于有良好网络链接的同学, 我们强烈建议下载较大的那个完整版, 实在是网络速度不佳的同学可以下载精简版的, 但是编译的时候会经常实时下载某些宏包, 很麻烦, 因此不推荐.

**Question:** 我自己新建了一个文件并且写了一段文字, 为什么编译出来的中文是乱码呢? 但是用模板里面的文件编译正常.

**Answer:** 请仔细确认你新建的文件是不是 UTF-8 编码! 如果不是, 请按文章前述方法转换成 UTF-8 编码.

**Question:** 我编译总是报错, 按 e 定位总是定位到主文件?

**Answer:** 这种问题比较复杂, 可能是你的某个正文中的命令与模板冲突, 或者有时与宏包冲突也会出现这种情况. 这时我们只能手动定位错误, 具体方式如下: 首先确定问题在哪一章? 方法是一章一章的编译, 然后确定问题在哪一节, 方法是只编译问题章节, 然后一节一节的编译. 最后确定问题在哪一行. 一般对于新手而言, 容易在数学公式中出问题, 所以, 请优先检查数学公式. 这样通过逐级定位, 我们就可以精确的获知错误的位置, 然后仔细检查, 就可以得到错误的原因, 即使你无法理解错误, 也可以对症下药, 在  $\text{T}_{\text{E}}\text{X}$  版求问.

**Question:** 编译错误时, DOS 窗口里报了一大堆英文, 如何看懂啊?

**Answer:** 前面的都是编译信息不用理睬, 后面的是错误信息要仔细看, 比如这一段

```
! Undefined control sequence.
1.16 \adga
      agasgarga
?
```

这一段有 4 行, 最后一行是问你如何处理, 我们在这里输入我们的处理方案. 现在看前三行. 第一行由一个感叹号开头, 这里是错误原因, 说明我用了一个没有定义的命令. 第二行

是1.16开头,说明第16行出问题了,然后这里有一个断行,这个断行表示错误的精确位置,这里,由于没有`\adga`这个命令,自然就会报错.后面的第三行是辅助文字,表示错误后面的一些代码,这样可以方便我们精确定位,这种显示方式在大段的数学公式出错时尤其方便.至于第4行,前述办法是:一般按E,回车,方便的在代码中回溯到我们需要的行.在这里我们再介绍一个命令:`h---`回车.`h`的意思是`help`,这样,`TeX`的编译器会使用尽可能你熟悉的语言告诉你具体为什么错了.有的时候这是很有帮助的,而有的时候你却完全不知道他在说什么.这是很正常的,因为电脑的长处在于执行命令,而不是主动读懂你的想法.(Knuth in TeXbook)

Question: 一个诡异的错误:

Answer: 也许你已经学习了一些公式的编写,那么,请看下面一段代码:

```
\begin{eqnarray}
[X_1, X_2] &=& 0\\
[X_3, X_4] &=& Q_{\pm}
\end{eqnarray}
```

你肯定觉得这是个很简单的括号问题,但是实际上这是无法编译的!解决方案是在第一行公式后面加上`\relax`,原因很巧妙,换行符`\\`后面的中括号代表该命令的可选参数,表示空多少尺寸,在上述公式中,第二行的公式恰好是以`[`开头,自然无法编译过去.这在`TeXbook`中叫做“weird mistakes”.其实,我们有很多可能遇到各种诡异的问题,有时这种问题的思路是超越我们的想法的,因此,希望大家可以多多发问,善于在网络上与同学们讨论问题,这样才能有助于问题的解决.



## 附录 A 附录

附录是作为说明书（论文）的补充部分，并不是必需的。

### 1. 下列内容可以作为附录编于说明书（论文）之后：

- (a) 为了说明书（论文）的完整，但编入正文又损于正文的处理和逻辑性，这一类材料包括比正文更为详细的信息研究方法和技术的途述，对于了解正文内容具有重要的补充意义；
- (b) 由于篇幅过大或取材于复制品而不便编入正文的材料；
- (c) 某些重要的原始数据、数学推导、计算程序、注释、框图、统计表、打印机输出样片、结构图等。

### 2. 附录中的有关格式

说明书（论文）的附录依次为“附录 A”、“附录 B”、“附录 C”等编号。如果只有一个附录，也应编为“附录 A”。

附录中的图、表、公式的命名方法也采用上面提到的图、表、公式命名方法，只不过将章的序号换成附录的序号。

## 致 谢

### **From XPS:**

参考了以下内容: ctex(Package), book.cls, cctbook.cls, article.cls, latex 源代码, titlesec(doc) , natbib(doc), caption(doc), fancyhdr(doc) 字号--pt 对照表 (From Web), ustcthesis(博士论文测试版), bbs.ctex.org, www.wikipedia.org 感谢以上文档以及他们的维护者.

### **From heaven7:**

感谢 XPS 同学辛苦制作的模板。

参考内容见参考文献。

### **From ywg:**

对 XPS 同学的辛苦工作表示由衷的感谢!

感谢大家对本模板更新工作的支持!