现代 LaTeX 入门讲座

Modern LaTeX in a Nutshell

曾祥东

2019 年 4 月 18 日

复旦大学　物理系

"Your paper makes no goddamn sense,
but it's the most beautiful thing
I have ever laid eyes on."

From r/ProgrammerHumor

介绍

历史回眸

高德纳（Donald E. Knuth）
TEX

Leslie Lamport
LATEX

图片来源：

LaTeX 是什么?

LaTeX 是什么?

- 发音:
  - /ˈlɑːtɛx/ or /ˈleɪtɛx/ or anything you like

LaTeX 是什么?

- 发音:
  - /ˈlɑːtɛx/ or /ˈleɪtɛx/ or anything you like
- 打公式方便?
  - 「复杂公式输入哪家强，当然首选 LaTeX 帮忙」

LaTeX 是什么?

- 发音:
  - /ˈlɑːtɛx/ or /ˈleɪtɛx/ or anything you like
- 打公式方便?
  - 「复杂公式输入哪家强，当然首选 LaTeX 帮忙」
- 写论文神器?
  - 「想要轻松给论文排版，当然少不了 LaTeX 啦」

LaTeX 是什么?

- 发音:
  - /ˈlɑːtɛx/ or /ˈleɪtɛx/ or anything you like
- 打公式方便?
  - 「复杂公式输入哪家强，当然首选 LaTeX 帮忙」
- 写论文神器?
  - 「想要轻松给论文排版，当然少不了 LaTeX 啦」
- 不想做宏编程的标记语言不是好的排版引擎?
  - LaTeX is a high-quality typesetting system; it includes features designed for the production of technical and scientific documentation. LaTeX is the *de facto* standard for the communication and publication of scientific documents. LaTeX is available as free software. 🔗

LaTeX 是什么？——What you *think* is what you get!

```latex
\documentclass{article}
\usepackage{amsmath,graphicx}
\title{Normal distribution}
\author{Wikipedia, the free encyclopedia}

\begin{document}
\maketitle
\section{Introduction}
% 省略一些内容……
The probability density of the normal
distribution is
\begin{equation}
  f(x|\mu, \sigma)
  = \frac{1}{\sqrt{2\pi\sigma^2}}
    e^{-\frac{(x-\mu)^2}{2\sigma^2}}
\end{equation}
where
\begin{itemize}
  \item $\mu$ is the mean of the distribution
  \item $\sigma$ is the standard deviation
\end{itemize}
\end{document}
```

来源：Wikipedia 🔗

LaTeX 是什么？ ——What you *think* is what you get!

```latex
\documentclass{article}
\usepackage{amsmath,graphicx}
\title{Normal distribution}
\author{Wikipedia, the free encyclopedia}

\begin{document}
\maketitle
\section{Introduction}
% 省略一些内容……
The probability density of the normal
distribution is
\begin{equation}
  f(x|\mu, \sigma)
  = \frac{1}{\sqrt{2\pi\sigma^2}}
    e^{-\frac{(x-\mu)^2}{2\sigma^2}}
\end{equation}
where
\begin{itemize}
  \item $\mu$ is the mean of the distribution
  \item $\sigma$ is the standard deviation
\end{itemize}
\end{document}
```

来源：Wikipedia

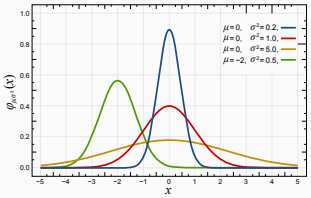
Normal distribution

Wikipedia, the free encyclopedia

April 18, 2019

1 Introduction

In probability theory, the normal distribution is a very common continuous probability distribution. Normal distributions are important in statistics and are often used in the natural and social sciences to represent real-valued random variables whose distributions are not known. A random variable with a Gaussian distribution is said to be normally distributed and is called a normal deviate.



The probability density of the normal distribution is

$$f(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{1}$$

where

- μ is the mean of the distribution
- σ is the standard deviation

1

基本原则

- 排版 vs 文字处理
  - 《别把 LaTeX 当 Word 用》

基本原则

- 排版 vs 文字处理
  - 《别把 LaTeX 当 Word 用》
- 遵循业（xué）界（xiào）规范
  - 《管教务处 *or* 研究生院 *or* 物理系叫爸爸》

基本原则

- 排版 vs 文字处理
  - 《别把 LaTeX 当 Word 用》
- 遵循业（xué）界（xiào）规范
  - 《管教务处 *or* 研究生院 *or* 物理系叫爸爸》
- 追求良好的阅读体验（readability）

基本原则

- 排版 vs 文字处理
  - 《别把 LaTeX 当 Word 用》
- 遵循业（xué）界（xiào）规范
  - 《管教务处 *or* 研究生院 *or* 物理系叫爸爸》
- 追求良好的阅读体验（readability）
- 内容与格式分离

基本原则

- 排版 vs 文字处理
  - 《别把 LaTeX 当 Word 用》
- 遵循业（xué）界（xiào）规范
  - 《管教务处 *or* 研究生院 *or* 物理系叫爸爸》
- 追求良好的阅读体验（readability）
- 内容与格式分离
- **内容永远比格式重要！**

安裝

选择发行版

- TEX 发行版（distribution）
  - 引擎、宏包、字体、文档的综合体
  - 类比 Visual Studio
  - TEX Live、MacTEX、W32TEX、MiKTEX 等

选择发行版

- TEX 发行版（distribution）
  - 引擎、宏包、字体、文档的综合体
  - 类比 Visual Studio
  - TEX Live、MacTEX、W32TEX、MiKTEX 等
- TEX Live
  - 官方维护，首选，跨平台
  - MacTEX ≈ macOS 下的 TEX Live
  - 缺点：体积大（3GB+）、每年需要重装（2019 版将于月底发布）
- MiKTEX
  - 由 Christian Schenk 维护（是个狠人）
  - 宏包随用随装
  - 缺点：曾经只有 Windows 版本、网络问题

选择发行版

- TEX 发行版（distribution）
  - 引擎、宏包、字体、文档的综合体
  - 类比 Visual Studio
  - TEX Live、MacTEX、W32TEX、MiKTEX 等
- TEX Live
  - 官方维护，首选，跨平台
  - MacTEX ≈ macOS 下的 TEX Live
  - 缺点：体积大（3GB+）、每年需要重装（2019 版将于月底发布）
- MiKTEX
  - 由 Christian Schenk 维护（是个狠人）
  - 宏包随用随装
  - 缺点：曾经只有 Windows 版本、网络问题
- **不要安装 CTEX 套装！**
  - **存在严重 bug，并且完全过时**

下载

- 选择国内 CTAN 镜像
  - 清华大学开源软件镜像站
  - 上海交通大学软件源镜像服务
  - 中国科学技术大学开源软件镜像

下载

- 选择国内 CTAN 镜像
  - 清华大学开源软件镜像站
  - 上海交通大学软件源镜像服务
  - 中国科学技术大学开源软件镜像
  - 复旦大学……

下载

- 选择国内 CTAN 镜像
  - 清华大学开源软件镜像站
  - 上海交通大学软件源镜像服务
  - 中国科学技术大学开源软件镜像
  - 复旦大学……
- 建议使用 ISO 镜像离线安装
- 在线安装要求网络稳定

安装流程

- 新手建议安装完整版 TEX Live 或 MacTEX
  - 完整版可避免后续维护烦恼
  - 一路点击「下一步」
  - 保持耐心，做好重装的打算

安装流程

- 新手建议安装完整版 TEX Live 或 MacTEX
  - 完整版可避免后续维护烦恼
  - 一路点击「下一步」
  - 保持耐心，做好重装的打算
- Linux specials
  - 软件源更新较慢，可以考虑 Vanilla TEX Live
  - GUI 安装界面需要 `perl-tk` 等
  - 环境变量、`fontconfig`、dummy package 配置
- macOS specials
  - 推荐 Homebrew

神圣的战争——选择编辑器

- 专用型
  - TeXworks：TEX Live 自带
  - TeXstudio：功能丰富，对新手友好
  - TeXShop：MacTEX 自带
  - WinEdt：功能丰富，收费

神圣的战争——选择编辑器

- 专用型
  - TeXworks：TEX Live 自带
  - TeXstudio：功能丰富，对新手友好
  - TeXShop：MacTEX 自带
  - WinEdt：功能丰富，收费
- 通用型
  - Visual Studio Code：利益相关（逃
  - Atom：听说很卡?
  - Sublime Text：收费
  - Vim：`q`、`q!`、`wq`、`wq!`

神圣的战争——选择编辑器

- 专用型
  - TeXworks：TEX Live 自带
  - TeXstudio：功能丰富，对新手友好
  - TeXShop：MacTEX 自带
  - WinEdt：功能丰富，收费
- 通用型
  - Visual Studio Code：利益相关（逃
  - Atom：听说很卡?
  - Sublime Text：收费
  - Vim：`q`、`q!`、`wq`、`wq!`
- 在线服务
  - ShareLaTeX 和 Overleaf（现已合并）

神圣的战争——选择编辑器

- 专用型
  - TeXworks：TEX Live 自带
  - TeXstudio：功能丰富，对新手友好
  - TeXShop：MacTEX 自带
  - WinEdt：功能丰富，收费
- 通用型
  - Visual Studio Code：利益相关（逃
  - Atom：听说很卡?
  - Sublime Text：收费
  - Vim：`q`、`q!`、`wq`、`wq!`
- 在线服务
  - ShareLaTeX 和 Overleaf（现已合并）
- 编辑器对比：

开始之前……

命令行基础

- 打开终端
  - 🪟：右键开始菜单、空白处 ⇧ + 右键、 ⊞ + R & `cmd`
  - 🐧： Ctrl + Alt + T
  - 🍎： ⌘ + ␣ 搜索 Terminal、可在 Finder 中添加服务
- 基本命令：
  - `cd`、`ls/dir`、`rm/del`、`clear/cls`
  - 选项：`-h`、`--help`、`/?`
- 其他：
  - 复制粘贴： Ctrl/⇧ + Ins、 Ctrl/⌘ + C/V、
  - 路径连接符：斜线（`/`）或反斜线（`\`）
  - 换行符：LF（`\n`）或 CRLF（`\r\n`）
  - 结束进程： Ctrl + C

命令行基础

- 打开终端
  - 🪟：右键开始菜单、空白处 ⇧ + 右键、 ⊞ + R & `cmd`
  - 🐧： Ctrl + Alt + T
  - 🍎： ⌘ + ␣ 搜索 Terminal、可在 Finder 中添加服务
- 基本命令：
  - `cd`、`ls/dir`、`rm/del`、`clear/cls`
  - 选项：`-h`、`--help`、`/?`
- 其他：
  - 复制粘贴： Ctrl/⇧ + Ins、 Ctrl/⌘ + C/V、
  - 路径连接符：斜线（`/`）或反斜线（`\`）
  - 换行符：LF（`\n`）或 CRLF（`\r\n`）
  - 结束进程： Ctrl + C
- **尽量不要用中文；避免空格、特殊符号**

编码

关于 LaTeX 源文件的编码，我们给出如下结论：

关于 LaTeX 源文件的编码，我们给出如下结论：

编码定理

一般地，在任何场合使用（不带 BOM）的 **UTF-8** 编码均是最优选择.

关于 LaTeX 源文件的编码，我们给出如下结论：

编码定理

一般地，在任何场合使用（不带 BOM）的 **UTF-8** 编码均是最优选择.

此定理的证明留做习题.

填写内容

Hello world!

```latex
% 用 pdfLaTeX、XeLaTeX 或 LuaLaTeX 编译
\documentclass{article}
\begin{document}
Hello world!
\end{document}
```

Hello world!

```latex
% 用 pdfLaTeX、XeLaTeX 或 LuaLaTeX 编译
\documentclass{article}
\begin{document}
Hello world!
\end{document}

% 用 XeLaTeX 或 LuaLaTeX 编译
\documentclass{ctexart}
\begin{document}
你好，世界！
\end{document}
```

引擎与格式

- **引擎**：TEX 的实现
  - pdfTEX：直接生成 PDF，支持 micro-typography
  - XƎTEX：支持 Unicode、OpenType 与复杂文字编排（CTL）
  - LuaTEX：支持 Unicode，内联 Lua，支持 OpenType
  - (u)pTEX：日本方面推动，生成 `.dvi`，（支持 Unicode）
  - ApTEX：底层 CJK 支持，内联 Ruby，Color Emoji（手动斜眼笑）

引擎与格式

- **引擎**：TEX 的实现
  - pdfTEX：直接生成 PDF，支持 micro-typography
  - XƎTEX：支持 Unicode、OpenType 与复杂文字编排（CTL）
  - LuaTEX：支持 Unicode，内联 Lua，支持 OpenType
  - (u)pTEX：日本方面推动，生成 `.dvi`，（支持 Unicode）
  - ApTEX：底层 CJK 支持，内联 Ruby，Color Emoji（手动斜眼笑）
- **格式**：TEX 的语言扩展（命令封装）
  - plain TEX：Knuth 同志专用
  - LATEX：排版科技类文章的事实（*de facto*）标准
  - ConTEXt：基于 LuaTEX 实现，优雅、易用（吗?）

引擎与格式

- **引擎**：TEX 的实现
  - pdfTEX：直接生成 PDF，支持 micro-typography
  - XƎTEX：支持 Unicode、OpenType 与复杂文字编排（CTL）
  - LuaTEX：支持 Unicode，内联 Lua，支持 OpenType
  - (u)pTEX：日本方面推动，生成 `.dvi`，（支持 Unicode）
  - ApTEX：底层 CJK 支持，内联 Ruby，Color Emoji（手动斜眼笑）
- **格式**：TEX 的语言扩展（命令封装）
  - plain TEX：Knuth 同志专用
  - LATEX：排版科技类文章的事实（*de facto*）标准
  - ConTEXt：基于 LuaTEX 实现，优雅、易用（吗?）
- **程序**：引擎 + dump 之后的格式代码
  - **英文文章：pdfLATEX、XƎLATEX 或 LuaLATEX**
  - **中文文章：XƎLATEX 或 LuaLATEX**

编译

- 现代 TEX 引擎均可直接生成 PDF

编译

- 现代 TEX 引擎均可直接生成 PDF
- 命令行
  - `pdflatex`/`xelatex`/`lualatex` + `<文件名>[.tex]`
  - 多次编译：读取并排版中间文件

编译

- 现代 TEX 引擎均可直接生成 PDF
- 命令行
  - `pdflatex`/`xelatex`/`lualatex` + `<文件名>[.tex]`
  - 多次编译：读取并排版中间文件
  - 推荐 `latexmk`：`latexmk [<选项>] <文件名>`

编译

- 现代 TEX 引擎均可直接生成 PDF
- 命令行
  - pdflatex/xelatex/lualatex + <文件名>[.tex]
  - 多次编译：读取并排版中间文件
  - 推荐 latexmk：latexmk [<选项>] <文件名>
- 编辑器
  - 按钮的背后仍然是命令
  - PATH 环境变量：确定可执行文件的位置
  - VS Code：配置 tools 和 recipes

语法

- 注释以 `%` 开头，忽略其后所有内容
- 命令以 `\` 开头，区分大小写
  - `\foo{arg}`：必选参数放在 `{...}` 中
  - `\foo[bar]{arg}`：可选参数放在 `[...]`
- 环境

```
\begin{env}
  ...
\end{env}
```

- 特殊符号需要转义：`\%`、`\$`、`\&`、`\textbackslash` 等
- 连续多个空格 = 单个空格 = 单个换行符

语法

- 注释以 `%` 开头，忽略其后所有内容
- 命令以 `\` 开头，区分大小写
  - `\foo{arg}`：必选参数放在 `{...}` 中
  - `\foo[bar]{arg}`：可选参数放在 `[...]`
- 环境

```
\begin{env}
  ...
\end{env}
```

- 特殊符号需要转义：`\%`、`\$`、`\&`、`\textbackslash` 等
- 连续多个空格 = 单个空格 = 单个换行符
- TEX/LATEX 的语法可以修改

文件结构

```latex
% 用 UTF-8 编码，命名为 xxx.tex
\documentclass{article}                    % 指明文档类型：文章
% 导言区：设置文档样式
\usepackage{amsmath}                       % 调用宏包，实现各种功能
\newcommand\keyword[1]{\textbf{#1}}        % 自定义命令

\begin{document}
% 正文：套用格式
In quantum mechanics, the \keyword{Schr\"odinger equation} is a
mathematical equation that describes the changes over time of a
physical system in which quantum effects, such as \keyword{wave--%
particle duality}, are significant.

% 上面的空行表示分段
In classical mechanics, Newton's second law
($\boldsymbol{F}=m\boldsymbol{a}$) is used to make a\ldots{}

Time-dependent Schrödinger equation can be written as  % ö 也能直接用
\[ i\hbar \frac{d}{dt} |\Psi(t)\rangle = \hat{H} |\Psi(t)\rangle. \]
\end{document}
```

Schrödinger equation

In quantum mechanics, the **Schrödinger equation** is a mathematical equation that describes the changes over time of a physical system in which quantum effects, such as **wave–particle duality**, are significant.

In classical mechanics, Newton's second law ($\boldsymbol{F} = m\boldsymbol{a}$) is used to make a…

Time-dependent Schrödinger equation can be written as

$$i\hbar \frac{d}{dt} |\Psi(t)\rangle = \hat{H} |\Psi(t)\rangle.$$

谋篇布局

- 文档部件
  - 标题：`\title`、`\author`、`\date` → `\maketitle`
  - 摘要：`abstract` 环境
  - 目录：`\tableofcontents`
  - 章节：`\chapter`、`\section`、`\subsection` 等
  - 文献：`\bibliography`
- 文档划分
  - 凤头猪肚豹尾：`\frontmatter`、`\mainmatter`、`\backmatter`
  - 分文件编译：`\include`、`\input`

文本标记

- 加粗：{\bfseries ...} 或 \textbf{...}
- 倾斜：{\itshape ...} 或 \textit{...}
- 字号：\tiny、\small、\large、\Large 等
- 换行：\\
- 缩进：\indent
- 居中：\centering 或 center 环境

请忘记上一页

文本标记

- 为什么要有不同的标记?

文本标记

- 为什么要有不同的标记？——表达不同的**语义**

文本标记

- 为什么要有不同的标记？——表达不同的**语义**
- `\textbf` 这样的命令是否表达语义？

文本标记

- 为什么要有不同的标记？——表达不同的**语义**
- `\textbf` 这样的命令是否表达语义？
- 再提一遍基本原则：**内容与格式分离**

文本标记

- 为什么要有不同的标记？——表达不同的**语义**
- `\textbf` 这样的命令是否表达语义？
- 再提一遍基本原则：**内容与格式分离**
- 正确（或曰：合理）的做法
  - 强调文字（意大利体）：`\emph{...}`
  - 摘要（居中，小字号，带有标题）：`abstract` 环境
  - 引用（左右边距较大）：`quote` 或 `quotation` 环境
  - 自定义新的命令、环境

文本标记

- 为什么要有不同的标记？——表达不同的**语义**
- `\textbf` 这样的命令是否表达语义？
- 再提一遍基本原则：**内容与格式分离**
- 正确（或曰：合理）的做法
  - 强调文字（意大利体）：`\emph{...}`
  - 摘要（居中，小字号，带有标题）：`abstract` 环境
  - 引用（左右边距较大）：`quote` 或 `quotation` 环境
  - 自定义新的命令、环境
- 报告，我想偷懒！

浮动体与交叉引用

- 浮动体机制
  - `figure` 和 `table` 环境
  - 文本为主，图、表为辅
  - 希望浮动体不要乱跑：「这是病，得治」🔗
  - 避免「见上图」、「见下表」

浮动体与交叉引用

- 浮动体机制
  - `figure` 和 `table` 环境
  - 文本为主，图、表为辅
  - 希望浮动体不要乱跑：「这是病，得治」🔗
  - 避免「见上图」、「见下表」
- 以标签控制交叉引用
  - 被引处：`\label`
  - 引用处：`\ref`、`\eqref` 等
  - 用有意义的标签：`\label{eq:euler-lagrange-eq}`
  - 需多次编译——推荐 `latexmk`

如何在论文中画出漂亮的插图?

- 外部插入
  - Mathematica、MATLAB
  - PowerPoint、Visio、Adobe Illustrator、Inkscape
  - Python Matplotlib、`Plots.jl`、R、Plotly 等

如何在论文中画出漂亮的插图?

- 外部插入
  - Mathematica、MATLAB
  - PowerPoint、Visio、Adobe Illustrator、Inkscape
  - Python Matplotlib、`Plots.jl`、R、Plotly 等
- TEX 内联
  - Asymptote
  - **`pgf/TikZ`、`pgfplots`**

如何在论文中画出漂亮的插图?

- 外部插入
  - Mathematica、MATLAB
  - PowerPoint、Visio、Adobe Illustrator、Inkscape
  - Python Matplotlib、`Plots.jl`、R、Plotly 等
- TEX 内联
  - Asymptote
  - **pgf/TikZ、pgfplots**
- 插图格式
  - 矢量图：`.pdf`
  - 位图：`.jpg` 或 `.png`
  - **不再推荐 `.eps`**
  - 不（完全）支持 `.svg`、`.bmp`

如何在论文中画出漂亮的插图?

- 外部插入
  - Mathematica、MATLAB
  - PowerPoint、Visio、Adobe Illustrator、Inkscape
  - Python Matplotlib、`Plots.jl`、R、Plotly 等
- TEX 内联
  - Asymptote
  - **`pgf/TikZ`、`pgfplots`**
- 插图格式
  - 矢量图：`.pdf`
  - 位图：`.jpg` 或 `.png`
  - **不再推荐 `.eps`**
  - 不（完全）支持 `.svg`、`.bmp`
- 参考：🔗 🔗 🔗

文献

- 建议自动生成

文献

- 建议自动生成（你只有三篇参考文献?）

文献

- 建议自动生成（你只有三篇参考文献?）
- `.bib` 数据库
  - Google Scholar 可直接复制
  - 用 Zotero、Jabref 等生成

文献

- 建议自动生成（你只有三篇参考文献?）
- `.bib` 数据库
  - Google Scholar 可直接复制
  - 用 Zotero、Jabref 等生成
- 传统方法：BibTeX 后端
  - 控制文献、引用样式：`natbib` 宏包
  - 国家标准 GB/T 7714–2015 🔗 🔗：**`gbt7714` 宏包**

文献

- 建议自动生成（你只有三篇参考文献?）
- `.bib` 数据库
  - Google Scholar 可直接复制
  - 用 Zotero、Jabref 等生成
- 传统方法：BibTeX 后端
  - 控制文献、引用样式：`natbib` 宏包
  - 国家标准 GB/T 7714–2015 🔗 🔗：**`gbt7714` 宏包**
- 现代方法：`biber` 后端 + `biblatex` 宏包
  - 国家标准：`biblatex-gb7714-2015` 宏包

文献

- 建议自动生成（你只有三篇参考文献?）
- `.bib` 数据库
  - Google Scholar 可直接复制
  - 用 Zotero、Jabref 等生成
- 传统方法：BibTeX 后端
  - 控制文献、引用样式：`natbib` 宏包
  - 国家标准 GB/T 7714–2015 🔗 🔗：**`gbt7714` 宏包**
- 现代方法：`biber` 后端 + `biblatex` 宏包
  - 国家标准：`biblatex-gb7714-2015` 宏包
- 需多次编译——再次推荐 `latexmk`

邐

U+30EDD (TBD)

中文支持

- 中文有什么特殊?

中文支持

- 中文有什么特殊?
  - 汉字太多（87,887+）

中文支持

- 中文有什么特殊?
  - 汉字太多（87,887+）
  - 横排 + 直排、标点禁则、行间注 🔗

中文支持

- 中文有什么特殊?
  - 汉字太多（87,887+）
  - 横排 + 直排、标点禁则、行间注 🔗
  - にほんご、देवनागरी、العَرَبِيَّة、𓅓𓏏𓅱𓀀𓁹

中文支持

- 中文有什么特殊?
  - 汉字太多（87,887+）
  - 横排 + 直排、标点禁则、行间注 🔗
  - にほんご、देवनागरी、العَرَبِيَّة、
- 已淘汰:
  - CCT 系统、CJK 宏包（裸用）
  - CTEX 套装

中文支持

- 中文有什么特殊?
  - 汉字太多（87,887+）
  - 横排 + 直排、标点禁则、行间注 🔗
  - にほんご、देवनागरी、العَرَبِيَّة、
- 已淘汰:
  - CCT 系统、CJK 宏包（裸用）
  - CTEX 套装
- 目前推荐手段:
  - **ctex 宏集**（此 ctex 非彼 CTEX）
  - XELATEX 编译

中文支持

- 中文有什么特殊?
  - 汉字太多（87,887+）
  - 横排 + 直排、标点禁则、行间注 🔗
  - にほんご、देवनागरी、العَرَبِيَّة、
- 已淘汰:
  - CCT 系统、CJK 宏包（裸用）
  - CTEX 套装
- 目前推荐手段:
  - **ctex 宏集**（此 ctex 非彼 CTEX）
  - XELATEX 编译
- 可以用，不推荐:
  - xeCJK 宏包（裸用）
  - ctex 宏集 + 其他引擎编译

中文支持

- 中文有什么特殊?
  - 汉字太多（87,887+）
  - 横排 + 直排、标点禁则、行间注 🔗
  - にほんご、देवनागरी、العَرَبِيَّة、
- 已淘汰:
  - CCT 系统、CJK 宏包（裸用）
  - CTEX 套装
- 目前推荐手段:
  - **ctex 宏集**（此 ctex 非彼 CTEX）
  - XELATEX 编译
- 可以用，不推荐:
  - xeCJK 宏包（裸用）
  - ctex 宏集 + 其他引擎编译

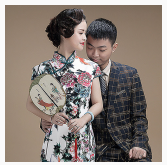
刘海洋

李清

吴凌云

江疆

马起园


黄晨成

鲁尚文

李泽平

图片来源：GitHub，知乎

模板

- 是什么?
  - 设计好的格式框架
  - 专注于内容：**不要追求与期刊排版一致**
  - Word 中的样式：「学好 LaTeX 可以更科学地使用 Word」

模板

- 是什么?
  - 设计好的格式框架
  - 专注于内容：**不要追求与期刊排版一致**
  - Word 中的样式：「学好 LaTeX 可以更科学地使用 Word」
- 有哪些?
  - 期刊：`revtex`、`elsarticle`、`IEEEtran`……
  - 学位论文：`thuthesis`、`ustcthesis`、**`fduthesis`**……

模板

- 是什么?
  - 设计好的格式框架
  - 专注于内容：**不要追求与期刊排版一致**
  - Word 中的样式：「学好 LaTeX 可以更科学地使用 Word」
- 有哪些?
  - 期刊：`revtex`、`elsarticle`、`IEEEtran`……
  - 学位论文：`thuthesis`、`ustcthesis`、**`fduthesis`**……
- 怎么用?
  - `\documentclass{...}`，配置参数，照常编写
  - **看文档，看文档，看文档**

模板

- 是什么?
  - 设计好的格式框架
  - 专注于内容：**不要追求与期刊排版一致**
  - Word 中的样式：「学好 LaTeX 可以更科学地使用 Word」
- 有哪些?
  - 期刊：`revtex`、`elsarticle`、`IEEEtran`……
  - 学位论文：`thuthesis`、`ustcthesis`、**`fduthesis`**……
- 怎么用?
  - `\documentclass{...}`，配置参数，照常编写
  - **看文档，看文档，看文档**
- 去哪里找?
  - CTAN 或 GitHub
  - 期刊官网
  - 「湿兄用 U 盘拷给你的模板一定是过时的」

公式

数学模式

- 一切数学公式都要在数学模式下输入
  - 不受外界字体命令控制
  - 数学模式中空格不起作用，尽管用
  - **不建议用 MathType 生成 LaTeX 公式**
  - 建议始终调用 `amsmath` 宏包

数学模式

- 一切数学公式都要在数学模式下输入
  - 不受外界字体命令控制
  - 数学模式中空格不起作用，尽管用
  - **不建议用 MathType 生成 LaTeX 公式**
  - 建议始终调用 `amsmath` 宏包
- 行内（inline）公式
  - 用一对美元符号（公式值千金）：`$...$`
  - 示例：理想气体状态方程可以写为 $PV = nRT$, 其中 P、V 和 T 分别是压强、体积和绝对温度

数学模式

- 一切数学公式都要在数学模式下输入
  - 不受外界字体命令控制
  - 数学模式中空格不起作用，尽管用
  - **不建议用 MathType 生成 LaTeX 公式**
  - 建议始终调用 `amsmath` 宏包
- 行内（inline）公式
  - 用一对美元符号（公式值千金）：`$...$`
  - 示例：理想气体状态方程可以写为 $PV = nRT$，其中 P、V 和 T 分别是压强、体积和绝对温度
- 独显（display）公式
  - 无编号：`\[...\]` 或 `equation*` 环境
  - 编号：`equation` 环境
  - **不要用 `$$...$$`**

结构

- 上下标
  - ^ 和 _：f^ab 和 f^{ab}，e^x^2、{e^x}^2 和 e^{x^2}
  - 张量：R^a{}_b{}^{cd} 或使用 tensor 宏包
  - 配合积分、求和、极限使用：\int、\sum、\lim；\(no)limits
- 分式
  - \frac{⟨分子⟩}{⟨分母⟩}
  - 行内分式、小分式不好看：改用 a/b，或改用独显公式
  - **不推荐 \frac**
- 根式
  - \sqrt[⟨次数⟩]{⟨内容⟩}
  - 复杂情况改用分数指数：{...}^{1/n}
- 矩阵与行列式
  - matrix、pmatrix、vmatrix 等环境
  - 语法类似表格：& 分列，\\ 换行
  - 推荐 physics 宏包

括号与定界符

- 基本括号
  - `(...)`、`[...]`、`\{...\}`、
  - 绝对值、范数：`|...|` 或 `\vert...\vert`、`\Vert...\Vert`
  - Dirac 符号：`\angle...\rangle`、`|...\rangle`

括号与定界符

- 基本括号
  - `(...)`、`[...]`、`\{...\}`、
  - 绝对值、范数：`|...|` 或 `\vert...\vert`、`\Vert...\Vert`
  - Dirac 符号：`\angle...\rangle`、`|...\rangle`
- 自动调节
  - `\left(...\right)` 等
  - 大型括号是拼出来的

括号与定界符

- 基本括号
  - `(...)`、`[...]`、`\{...\}`、
  - 绝对值、范数：`|...|` 或 `\vert...\vert`、`\Vert...\Vert`
  - Dirac 符号：`\angle...\rangle`、`|...\rangle`
- 自动调节
  - `\left(...\right)` 等
  - 大型括号是拼出来的
- 手动调节
  - 只有 4 + 1 档：`\big`、`\Big`、`\bigg`、`\Bigg`
  - 声明左中右：`\bigl`、`\bigm`、`\bigr` 等

符号与字体

- 符号不是按钮点出来的，也不是天上掉下来的

符号与字体

- 符号不是按钮点出来的，也不是天上掉下来的
  - （几乎）所有的符号都由字体提供

符号与字体

- 符号不是按钮点出来的，也不是天上掉下来的
  - （几乎）所有的符号都由字体提供
  - 分清「它是什么」和「它长什么样」（术语：character 和 glyph）

符号与字体

- 符号不是按钮点出来的，也不是天上掉下来的
  - （几乎）所有的符号都由字体提供
  - 分清「它是什么」和「它长什么样」（术语：character 和 glyph）
- 寻找符号
  - S. Pakin. *The Comprehensive LaTeX Symbol List* 🔗
  - 手写识别（不全）：Detexify 🔗

符号与字体

- 符号不是按钮点出来的，也不是天上掉下来的
  - （几乎）所有的符号都由字体提供
  - 分清「它是什么」和「它长什么样」（术语：character 和 glyph）
- 寻找符号
  - S. Pakin. *The Comprehensive LaTeX Symbol List* 🔗
  - 手写识别（不全）：Detexify 🔗
- 数学字体
  - 你们要的「Times New Roman」：`newtxmath` 宏包
  - **不要用 `times` 和 `mathptmx` 宏包**
  - 加粗：使用 `bm` 宏包

符号与字体

- 符号不是按钮点出来的，也不是天上掉下来的
  - （几乎）所有的符号都由字体提供
  - 分清「它是什么」和「它长什么样」（术语：character 和 glyph）
- 寻找符号
  - S. Pakin. *The Comprehensive LaTeX Symbol List* 🔗
  - 手写识别（不全）：Detexify 🔗
- 数学字体
  - 你们要的「Times New Roman」：`newtxmath` 宏包
  - **不要用 `times` 和 `mathptmx` 宏包**
  - 加粗：使用 `bm` 宏包
- 新方案：`unicode-math`
  - 符号、字体、样式精调的一揽子解决方案
  - 彻底修改底层，不可与传统方案混用

多行公式

- 以下均要求 amsmath 宏包
- 独立数学环境
  - 多行居中 gather、多行手动对齐 align、跨行 multiline
  - 手动对齐：关系符前加 &
- 内联数学环境
  - 条件 cases、多行对齐 split、...ed

多行公式

- 以下均要求 amsmath 宏包
- 独立数学环境
  - 多行居中 gather、多行手动对齐 align、跨行 multiline
  - 手动对齐：关系符前加 &
- 内联数学环境
  - 条件 cases、多行对齐 split、...ed
- 精细调整
  - mathtools、empheq 等
  - 自动换行：breqn
  - **避免使用 eqnarray 环境**

专业功能（一）

- 更高更妙的物理：physics 宏包
  - 括号：`\qty(...)`、`\qty\big{...}`
  - 矩阵：`\mqty(...)`、`\mqty|...|`、`\dmat{a,b,c,...}`
  - Dirac 符号：`\ket`、`\bra`、`\ev`
  - 向量、导数、微分、更多函数名……

专业功能（一）

- 更高更妙的物理：physics 宏包
  - 括号：`\qty(...)`、`\qty\big{...}`
  - 矩阵：`\mqty(...)`、`\mqty|...|`、`\dmat{a,b,c,...}`
  - Dirac 符号：`\ket`、`\bra`、`\ev`
  - 向量、导数、微分、更多函数名……
- 国际单位：siunitx 宏包
  - `$4.18 \times 10^3 J mol^{-1} K^{-1}$`
    - $4.18 \times 10^3 Jmol^{-1}K^{-1}$—No!
  - `\SI{4.18e3}{J.mol^{-1}.K^{-1}}`
    - $4.18 \times 10^3\,\mathrm{J\,mol^{-1}\,K^{-1}}$—Yes!

专业功能（一）

- 更高更妙的物理：physics 宏包
  - 括号：`\qty(...)`、`\qty\big{...}`
  - 矩阵：`\mqty(...)`、`\mqty|...|`、`\dmat{a,b,c,...}`
  - Dirac 符号：`\ket`、`\bra`、`\ev`
  - 向量、导数、微分、更多函数名……
- 国际单位：siunitx 宏包
  - `$4.18 \times 10^3 J mol^{-1} K^{-1}$`
    - $4.18 \times 10^3 Jmol^{-1}K^{-1}$—No!
  - `\SI{4.18e3}{J.mol^{-1}.K^{-1}}`
    - $4.18 \times 10^3\,\mathrm{J\,mol^{-1}\,K^{-1}}$—Yes!
  - 注 1：此宏包代码比 LaTeX 内核还长

专业功能（一）

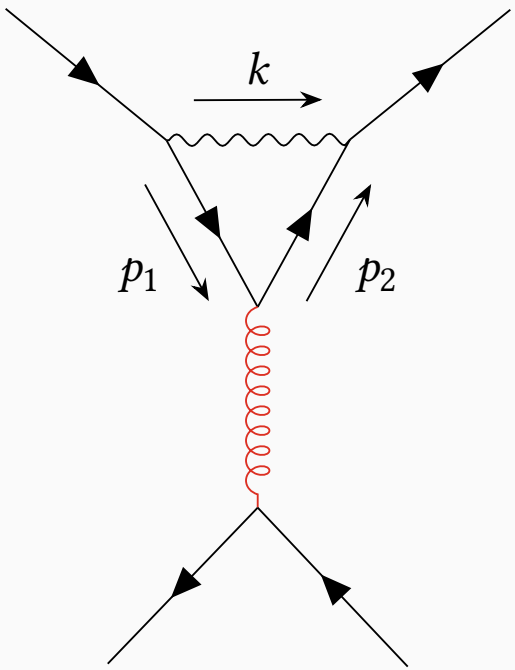
- 更高更妙的物理：physics 宏包
  - 括号：`\qty(...)`、`\qty\big{...}`
  - 矩阵：`\mqty(...)`、`\mqty|...|`、`\dmat{a,b,c,...}`
  - Dirac 符号：`\ket`、`\bra`、`\ev`
  - 向量、导数、微分、更多函数名……
- 国际单位：siunitx 宏包
  - `$4.18 \times 10^3 J mol^{-1} K^{-1}$`
    - $4.18 \times 10^3 Jmol^{-1}K^{-1}$—No!
  - `\SI{4.18e3}{J.mol^{-1}.K^{-1}}`
    - $4.18 \times 10^3\,\mathrm{J\,mol^{-1}\,K^{-1}}$—Yes!
  - 注 1：此宏包代码比 LaTeX 内核还长
  - 注 2：新定义并不会影响使用 🔗

专业功能（二）

- 花式图表
  - Feynman 图：`tikz-feynman` 宏包 (arXiv: 1601.05437 🔗)
  - Feynman 斜线：`slashed` 宏包
  - Wick 定理：`simplewick` 宏包、`simpler-wick` 宏包
  - Young 表、Young 图：`ytableau` 宏包
  - 电路图：`circuitikz` 宏包
  - 拓扑量子场论：`tqft` 宏包
  - ……

专业功能（二）

- 花式图表
  - Feynman 图：`tikz-feynman` 宏包 (arXiv: 1601.05437 🔗)
  - Feynman 斜线：`slashed` 宏包
  - Wick 定理：`simplewick` 宏包、`simpler-wick` 宏包
  - Young 表、Young 图：`ytableau` 宏包
  - 电路图：`circuitikz` 宏包
  - 拓扑量子场论：`tqft` 宏包
  - ……



专业功能（三）

- 抄录：忽略所有特殊类别码（catcode），原样显示
  - `\verb⟨char⟩...⟨char⟩`、`verbatim` 环境
  - `verbatim`、`fancyvrb` 宏包

专业功能（三）

- 抄录：忽略所有特殊类别码（catcode），原样显示
  - `\verb⟨char⟩...⟨char⟩`、`verbatim` 环境
  - `verbatim`、`fancyvrb` 宏包
- 语法高亮
  - `listings` 宏包
  - `minted` 宏包
    - 需要 Python，且开启 `--shell-escape`

专业功能（三）

- 抄录：忽略所有特殊类别码（catcode），原样显示
  - \verb⟨char⟩...⟨char⟩、verbatim 环境
  - verbatim、fancyvrb 宏包
- 语法高亮
  - listings 宏包
  - minted 宏包
    - 需要 Python，且开启 --shell-escape

```c
/* A standard Hello World program in C. */
#include <stdio.h>

int main(int argc, char** argv) {
    printf("Hello, world!\n");
    return 0;
}
```

字体排印

先看一个例子

Typography is the art and technique of arranging type to make written language legible, readable, and appealing when displayed. The arrangement of type involves selecting typefaces, point sizes, line lengths, line-spacing(*leading*), and letter-spacing (*tracking*), and adjusting the space between pairs of letters(*kerning*). The term typography is also applied to the style, arrangement, and appearance of the letters, numbers, and symbols created by the process. **Type design** is a closely related craft, sometimes considered part of typography;most typographers do not design typefaces, and some type designers do not consider themselves typographers. Typography also may be used as a decorative device, unrelated to communication of information.

中易黑体，行距 0.8 倍，关闭断词；来源：🔗

没有对比就没有伤害

Typography is the art and technique of arranging type to make written language legible, readable, and appealing when displayed. The arrangement of type involves selecting typefaces, point sizes, line lengths, line-spacing (*leading*), and letter-spacing (*tracking*), and adjusting the space between pairs of letters (*kerning*). The term typography is also applied to the style, arrangement, and appearance of the letters, numbers, and symbols created by the process. **Type design** is a closely related craft, sometimes considered part of typography; most typographers do not design typefaces, and some type designers do not consider themselves typographers. Typography also may be used as a decorative device, unrelated to communication of information.

EB Garamond，默认设置

术语

- 语言（language）
- 文字（script）
- 书写系统（writting system）
- 符号（symbol）
- 字符（character）
- 字符形（glyph）
- 字符集（character set）
- 编码（encoding）
- 码位（code point）
- 字体（font）
- 字型（typeface）
- 易认性（legibility）
- 可读性（readability）
- 字偶间距（kerning）
- 字距（tracking）
- 栅格化（rasterization）
- 渲染提示（hinting）
- ……

LaTeX will do (almost) all the things for you.

Punctuations: hyphen/dash

- Hyphen (`U+002D`): `-`
  - Four-dimensional momentum
  - Hyphenation

Punctuations: hyphen/dash

- Hyphen (`U+002D`): `-`
  - Four-dimensional momentum
  - Hyphenation

A hyphen-ation algo-rithm is a set of rules, especially one codified for imple-mentation in a computer program, that decides at which points a word can be broken over two lines with a hyphen.

Punctuations: hyphen/dash

- Hyphen (`U+002D`): `-`
  - Four-dimensional momentum
  - Hyphenation
- En dash (`U+2013`): `--`
  - Ryu–Takayanagi formula (*cf.* Levi-Civita symbol)
  - pp. 187–189

A hyphen-ation algo-rithm is a set of rules, especially one codified for imple-mentation in a computer program, that decides at which points a word can be broken over two lines with a hyphen.

Punctuations: hyphen/dash

- Hyphen (`U+002D`): `-`
  - Four-dimensional momentum
  - Hyphenation
- En dash (`U+2013`): `--`
  - Ryu–Takayanagi formula (*cf.* Levi-Civita symbol)
  - pp. 187–189
- Em dash (`U+2014`): `---`
  - Red, white, and blue—these are the colors of the flag
  - Like colon, parentheses, *etc.*

A hyphen-ation algo-rithm is a set of rules, especially one codified for imple-mentation in a computer program, that decides at which points a word can be broken over two lines with a hyphen.

Punctuations: hyphen/dash

- Hyphen (`U+002D`): `-`
  - Four-dimensional momentum
  - Hyphenation
- En dash (`U+2013`): `--`
  - Ryu–Takayanagi formula (*cf.* Levi-Civita symbol)
  - pp. 187–189
- Em dash (`U+2014`): `---`
  - Red, white, and blue—these are the colors of the flag
  - Like colon, parentheses, *etc.*
- Minus (`U+2212`): `$-$`
  - $a - b$, $-a$

A hyphen-ation algo-rithm is a set of rules, especially one codified for imple-mentation in a computer program, that decides at which points a word can be broken over two lines with a hyphen.

Punctuations: quotation mark

- Left/right, single/double:
  - ‘…’ (U+2018, U+2019): `` `...' ``
  - “…” (U+201C, U+201D): `` ``...'' ``

Punctuations: quotation mark

- Left/right, single/double:
  - ‘…’ (`U+2018`, `U+2019`): `` `...' ``
  - “…” (`U+201C`, `U+201D`): `` ``...'' ``
- Different languages:
  - ‘British “English” style’ and “American ‘English’ style”
  - „German”, ”Finnish”, «French», »Danish«, *etc.*

Punctuations: quotation mark

- Left/right, single/double:
  - ‘…’ (`U+2018`, `U+2019`): `` `...' ``
  - “…” (`U+201C`, `U+201D`): `` ``...'' ``
- Different languages:
  - ‘British “English” style’ and “American ‘English’ style”
  - „German”, ”Finnish”, «French», »Danish«, *etc.*
  - Use `csquotes` package

Punctuations: quotation mark

- Left/right, single/double:
  - ‘...’ (U+2018, U+2019): `` `...' ``
  - “...” (U+201C, U+201D): `` ``...'' ``
- Different languages:
  - ‘British “English” style’ and “American ‘English’ style”
  - „German”, ”Finnish”, «French», »Danish«, *etc.*
  - Use `csquotes` package
- Programming:
  - `char* my_name = "Xiangdong Zeng";`

Punctuations: quotation mark

- Left/right, single/double:
  - ‘...’ (U+2018, U+2019): `` `...' ``
  - “...” (U+201C, U+201D): `` ``...'' ``
- Different languages:
  - ‘British “English” style’ and “American ‘English’ style”
  - „German”, ”Finnish”, «French», »Danish«, *etc.*
  - Use `csquotes` package
- Programming:
  - `char* my_name = "Xiangdong Zeng";`
- Mathematics:
  - `f' = f^{\prime}`: $f' = f^{\prime}$

中文标点符号

- 句号
  - 正常文本。科技文本.

中文标点符号

- 句号
  - 正常文本。科技文本.
- 引号
  - 『传统风格』,「某乎风格」,“标准风格”, ’奇葩风格”

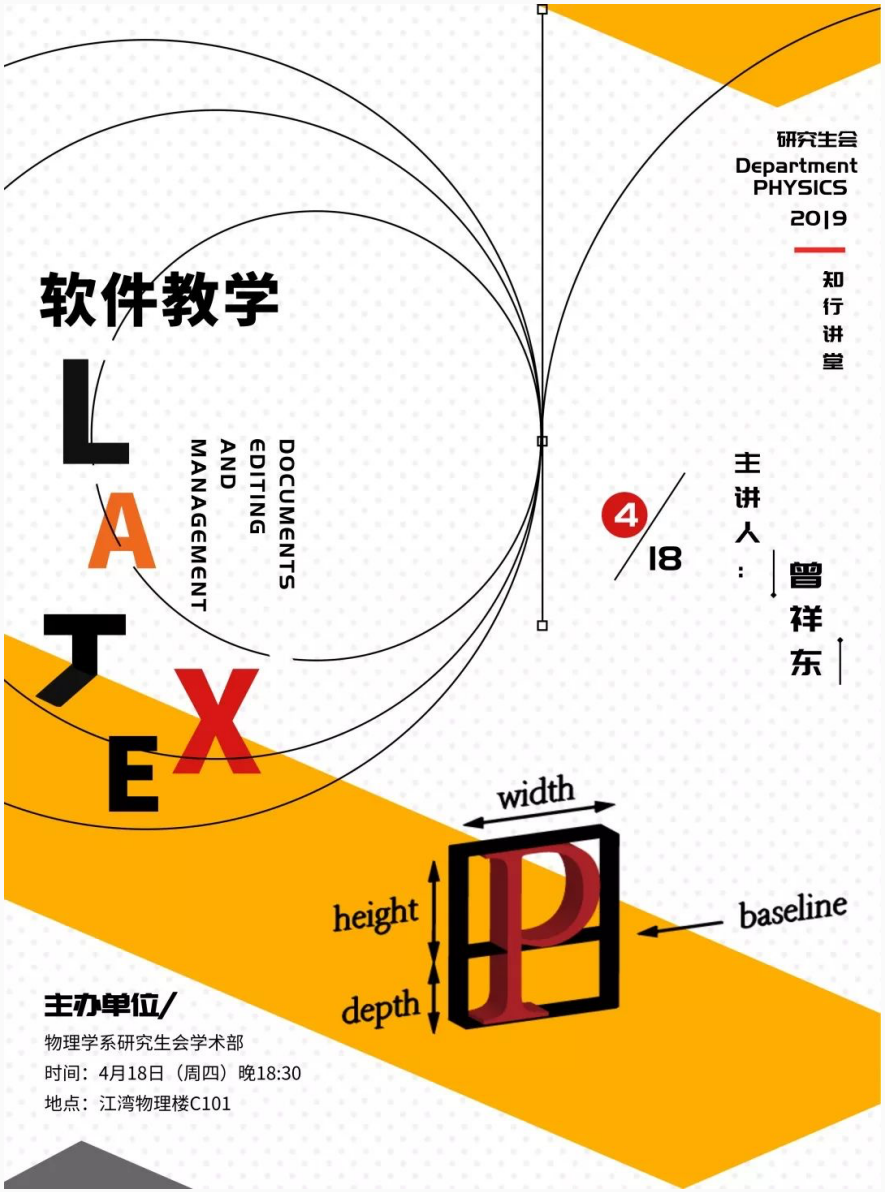
中文标点符号

- 句号
  - 正常文本。科技文本.
- 引号
  - 『传统风格』，「某乎风格」，“标准风格”，’奇葩风格”
- 破折号
  - 断开——是不好的，不断开——是好的

中文标点符号

- 句号
  - 正常文本。科技文本.
- 引号
  - 『传统风格』，「某乎风格」，“标准风格”，’奇葩风格”
- 破折号
  - 断开——是不好的，不断开——是好的
- 波浪号：
  - ~ ≠ \textasciitilde ≠ \texttildelow ≠ $\sim$ ≠ 你要的那一个
  - 那就是 青 ~ 藏 ˜ 高 ~ 原～～～～
    - U+007E: Tilde
    - U+02F7: Modifier letter low tilde
    - U+223C: Tilde operator
    - U+FF5E: Fullwidth tilde
    - …

大家来找茬 😎


研究生会
Department
PHYSICS
2019
知行讲堂
软件教学
LaTeX
DOCUMENTS
EDITING
AND
MANAGEMENT
4/18
主讲人：
曾祥东
width
height
depth
baseline
主办单位/
物理学系研究生会学术部
时间：4月18日（周四）晚18:30
地点：江湾物理楼C101

叁拾玖

使用字体：fontspec 宏包

- 字体家族
  - 衬线体：EB Garamond, Times New Roman, Latin Modern, *etc.*
  - 无衬线体：Helvetica Neue, Avenir Next, Optima, *etc.*
  - 等宽字体：`Courier`, `Menlo`, `Iosevka`, *etc.*
  - 中文字体：宋体、黑体、仿宋、楷书、（娃娃体）……

使用字体：fontspec 宏包

- 字体家族
  - 衬线体：EB Garamond, Times New Roman, Latin Modern, *etc.*
  - 无衬线体：Helvetica Neue, Avenir Next, Optima, *etc.*
  - 等宽字体：`Courier`, `Menlo`, `Iosevka`, *etc.*
  - 中文字体：宋体、黑体、仿宋、楷书、（娃娃体）……
- 样式
  - 粗体、意大利体：**Bold** vs **Faked bold**, *Italic* vs *Slant*

使用字体：fontspec 宏包

- 字体家族
  - 衬线体：EB Garamond, Times New Roman, Latin Modern, *etc.*
  - 无衬线体：Helvetica Neue, Avenir Next, Optima, *etc.*
  - 等宽字体：`Courier`, `Menlo`, `Iosevka`, *etc.*
  - 中文字体：宋体、黑体、仿宋、楷书、（娃娃体）……
- 样式
  - 粗体、意大利体：**Bold** vs **Faked bold**, *Italic* vs *Slant*
  - **汉字一般不使用斜体**

使用字体：fontspec 宏包

- 字体家族
  - 衬线体：EB Garamond, Times New Roman, Latin Modern, *etc.*
  - 无衬线体：Helvetica Neue, Avenir Next, Optima, *etc.*
  - 等宽字体：`Courier`, `Menlo`, `Iosevka`, *etc.*
  - 中文字体：宋体、黑体、仿宋、楷书、（娃娃体）……
- 样式
  - 粗体、意大利体：**Bold** vs **Faked bold**, *Italic* vs *Slant*
  - **汉字一般不使用斜体**
  - 视觉字号（optical size）：

    Tiny, Script, Footnote, Caption, Small, Normal, Large, Huge

使用字体：fontspec 宏包

- 字体家族
  - 衬线体：EB Garamond, Times New Roman, Latin Modern, *etc.*
  - 无衬线体：Helvetica Neue, Avenir Next, Optima, *etc.*
  - 等宽字体：`Courier`, `Menlo`, `Iosevka`, *etc.*
  - 中文字体：宋体、黑体、仿宋、楷书、（娃娃体）……
- 样式
  - 粗体、意大利体：**Bold** vs **Faked bold**, *Italic* vs *Slant*
  - **汉字一般不使用斜体**
  - 视觉字号（optical size）：

    Tiny, Script, Footnote, Caption, Small, Normal, Large, Huge
- OpenType 特性
  - 连字（ligature）：ff → ff, fi → fi, fl → fl
  - 老式数字（old-style number）：0123456789 → 0123456789
  - 字偶间距（kerning）：Ty → Ty, WA → WA

使用字体：fontspec 宏包

- 字体家族
  - 衬线体：EB Garamond, Times New Roman, Latin Modern, *etc.*
  - 无衬线体：Helvetica Neue, Avenir Next, Optima, *etc.*
  - 等宽字体：`Courier`, `Menlo`, `Iosevka`, *etc.*
  - 中文字体：宋体、黑体、仿宋、楷书、（娃娃体）……
- 样式
  - 粗体、意大利体：**Bold** vs **Faked bold**, *Italic* vs *Slant*
  - **汉字一般不使用斜体**
  - 视觉字号（optical size）：
    Tiny, Script, Footnote, Caption, Small, Normal, Large, Huge
- OpenType 特性
  - 连字（ligature）：ff → ff, fi → fi, fl → fl
  - 老式数字（old-style number）：0123456789 → 0123456789
  - 字偶间距（kerning）：Ty → Ty, WA → WA
- **请避免滥用过多字体**（此页除外）

进阶扩展

TEX 宏编程

- TEX 层面
  - 守序善良——定义命令：`\def`、`\gdef`、`\let`
  - 绝对中立——展开控制：`\edef`、`\expandafter`、`\aftergroup`
  - 混乱邪恶——类别码：`\catcode`

TEX 宏编程

- TEX 层面
  - 守序善良——定义命令：`\def`、`\gdef`、`\let`
  - 绝对中立——展开控制：`\edef`、`\expandafter`、`\aftergroup`
  - 混乱邪恶——类别码：`\catcode`
- LATEX 层面
  - 定义新命令：`\newcommand`、`\renewcommand`、`\newenvironment`
  - 内部命令：`\makeatletter`、`\makeatother`

TEX 宏编程

- TEX 层面
  - 守序善良——定义命令：\def、\gdef、\let
  - 绝对中立——展开控制：\edef、\expandafter、\aftergroup
  - 混乱邪恶——类别码：\catcode
- LATEX 层面
  - 定义新命令：\newcommand、\renewcommand、\newenvironment
  - 内部命令：\makeatletter、\makeatother
- LATEX3——遥遥无期的未来
  - 命令举例：\cs_new:cpx、\seq_sort:Nn、\regex_match:nnTF
  - 创建用户层命令：xparse 宏包
  - 面向对象编程：xtemplate 宏包
  - fontspec、siunitx、ctex、fduthesis 等均使用 LATEX3 实现

TEX 宏编程

- TEX 层面
  - 守序善良——定义命令：\def、\gdef、\let
  - 绝对中立——展开控制：\edef、\expandafter、\aftergroup
  - 混乱邪恶——类别码：\catcode
- LATEX 层面
  - 定义新命令：\newcommand、\renewcommand、\newenvironment
  - 内部命令：\makeatletter、\makeatother
- LATEX3——遥遥无期的未来
  - 命令举例：\cs_new:cpx、\seq_sort:Nn、\regex_match:nnTF
  - 创建用户层命令：xparse 宏包
  - 面向对象编程：xtemplate 宏包
  - fontspec、siunitx、ctex、fduthesis 等均使用 LATEX3 实现
- 外部语言调用
  - \write18 与 \directlua

深入字体

- NFSS 与字体的坐标
  - 字体族、形状、系列、编码、字号
  - TEX 仅需要 metric 信息：`.tfm`

深入字体

- NFSS 与字体的坐标
  - 字体族、形状、系列、编码、字号
  - TEX 仅需要 metric 信息：`.tfm`
- 现代方案：OpenType
  - 编码层面：支持 Unicode
  - 东亚文字：超大字符集、地区变体、竖排
  - 中东、南亚文字：Bi-di 文本、上下文连字、字符序调整

深入字体

- NFSS 与字体的坐标
  - 字体族、形状、系列、编码、字号
  - TEX 仅需要 metric 信息：`.tfm`
- 现代方案：OpenType
  - 编码层面：支持 Unicode
  - 东亚文字：超大字符集、地区变体、竖排
  - 中东、南亚文字：Bi-di 文本、上下文连字、字符序调整
- OpenType 中的数学支持（`MATH` 表）
  - Unicode Math：字母、符号支持
  - 「数学常数」：整体度量信息——上下标位置、分数线粗细等
  - `MathVariants`：大小替换（积分号、根号、括号等）
  - `MathGlyphConstruction`：字符形装配（更大的根号、括号等）

深入字体

- NFSS 与字体的坐标
  - 字体族、形状、系列、编码、字号
  - TEX 仅需要 metric 信息：`.tfm`
- 现代方案：OpenType
  - 编码层面：支持 Unicode
  - 东亚文字：超大字符集、地区变体、竖排
  - 中东、南亚文字：Bi-di 文本、上下文连字、字符序调整
- OpenType 中的数学支持（`MATH` 表）
  - Unicode Math：字母、符号支持
  - 「数学常数」：整体度量信息——上下标位置、分数线粗细等
  - `MathVariants`：大小替换（积分号、根号、括号等）
  - `MathGlyphConstruction`：字符形装配（更大的根号、括号等）
- 可变字体、絵文字（emoji, 😂）……

编写宏包

- 文学编程
  - 代码、注释与文档合为一体（.dtx 文件）
  - 使用 doc 与 docstrip 宏包

编写宏包

- 文学编程
  - 代码、注释与文档合为一体（`.dtx` 文件）
  - 使用 `doc` 与 `docstrip` 宏包
- 发布
  - 上传 CTAN 实际上并无门槛
  - 但仍有必要了解：
    - TEX 目录结构（TDS）
    - 测试系统：`l3build` 宏包
    - 版本控制、持续集成
    - 许可证选择：LATEX 内核使用 LPPL 1.3c

编写宏包

- 文学编程
  - 代码、注释与文档合为一体（`.dtx` 文件）
  - 使用 `doc` 与 `docstrip` 宏包
- 发布
  - 上传 CTAN 实际上并无门槛
  - 但仍有必要了解：
    - TEX 目录结构（TDS）
    - 测试系统：`l3build` 宏包
    - 版本控制、持续集成
    - 许可证选择：LATEX 内核使用 LPPL 1.3c
- 参考
  - LATEX 内核代码：`source2e.pdf`、`classes.pdf`、`source3.pdf`
  - 书籍：*The TEXbook*、*TEX by Topic*、*The LATEX Companion* 等

宏包推荐

- 必备
  - amsmath
  - graphicx
  - hyperref
- 样式
  - caption
  - enumitem
  - fancyhdr
  - footmisc
  - geometry
  - ntheorem
  - titlesec
- 数学
  - bm
  - mathtools
  - physics
  - unicode-math
- 表格
  - array
  - booktabs
  - longtable
  - tabularx
- 插图、绘图
  - float
  - pdfpages
  - standalone
  - subfig
  - pgf/tikz
  - pgfplots
- 字体
  - newtx
  - newpx
  - pifont
  - fontspec
- 各种功能
  - algorithm2e
  - beamer
  - biblatex
  - fancyhdr
  - listings
  - mhchem
  - microtype
  - minted
  - natbib
  - siunitx
  - xcolor
- 多语言
  - babel
  - polyglossia
  - ctex
  - xeCJK
  - luatexja

请务必先读文档!

请务必先读文档!

命令行执行 texdoc *package*

Markdown

```
# Markdown syntax

This is **bold text**.
This text is *italicized*.
Use `git status` to list all
new or modified files.

Block code:

```
git status
git add
git commit
```

Quotation:

> Markdown uses email-style `>`
> characters for blockquoting.

List

Bullet list

  * apples
  * oranges
  * pears

Numbered list

  1. wash
  2. rinse
  3. repeat

Link: from [Wikipedia]
(https://en.wikipedia.org/wiki/
Markdown)
```

Git

- 版本管理的必要性
  - 远离「初稿，第二稿，第三稿……终稿，终稿（打死也不改了）」
  - 有底气做大范围修改、重构
  - 方便与他人协同合作

Git

- 版本管理的必要性
  - 远离「初稿，第二稿，第三稿……终稿，终稿（打死也不改了）」
  - 有底气做大范围修改、重构
  - 方便与他人协同合作
- 基本用法
  - 把大象放进冰箱：`git init`、`git add`、`git commit`
  - 时空穿梭：`git reset`、`git revert`
  - 平行宇宙：`git branch`、`git checkout`、`git rebase`
  - 推荐用 VS Code 等进行可视化操作
  - 参考链接：🔗 🔗

Git

- 版本管理的必要性
  - 远离「初稿，第二稿，第三稿……终稿，终稿（打死也不改了）」
  - 有底气做大范围修改、重构
  - 方便与他人协同合作
- 基本用法
  - 把大象放进冰箱：`git init`、`git add`、`git commit`
  - 时空穿梭：`git reset`、`git revert`
  - 平行宇宙：`git branch`、`git checkout`、`git rebase`
  - 推荐用 VS Code 等进行可视化操作
  - 参考链接：🔗 🔗
- GitHub
  - 远程 Git 仓库
  - Clone & fork
  - Issues & pull requests

Git

- 版本管理的必要性
  - 远离「初稿，第二稿，第三稿……终稿，终稿（打死也不改了）」
  - 有底气做大范围修改、重构
  - 方便与他人协同合作
- 基本用法
  - 把大象放进冰箱：`git init`、`git add`、`git commit`
  - 时空穿梭：`git reset`、`git revert`
  - 平行宇宙：`git branch`、`git checkout`、`git rebase`
  - 推荐用 VS Code 等进行可视化操作
  - 参考链接：🔗 🔗
- GitHub
  - 远程 Git 仓库
  - Clone & fork
  - Issues & pull requests
  - **提醒：绑定 .edu 邮箱可以免费无限量使用私有仓库**

参考文献

[1] Knuth D E.
The TEXbook: Computers & Typesetting, volume C [M]. 1984.
Boston: Addison–Wesley Publishing Company

[2] 刘海洋.
LATEX 入门 [M]. 2013.
北京：电子工业出版社

[3] 高冈昌生.
翻译：刘庆，监修：陈嵘.
西文排版：排版的基础和规范 [M]. 2016.
北京：中信出版集团

[4] 小林章.
翻译：刘庆，监修：陈嵘.
西文字体：字体的背景知识和使用方法 [M]. 2014.
北京：中信出版集团

[5] Oetiker T, Partl H, Hyna I and Schlegl E.
翻译：CTEX 开发小组.
一份（不太）简短的 LATEX 2_ε 介绍：或 106 分钟了解 LATEX 2_ε [EB/OL]. 2018.
https://ctan.org/pkg/lshort-zh-cn

[6] 汪彧之，陈晟祺.
如何使用 LATEX 排版论文 [EB/OL]. 2018.
PDF:

[7] 刘海洋.
LATEX 入门 [EB/OL]. 2013.
PDF:

[8] 林莲枝.
漫谈 LATEX 排版常见概念误区：别把 LATEX 当 Word 用！ [EB/OL]. 2018.
PDF:

[9] Wikibooks.
LATEX—Wikibooks, The Free Textbook Project [EB/OL].
https://en.wikibooks.org/wiki/LaTeX

[10] Overleaf.
Overleaf Documentation [EB/OL].
https://www.overleaf.com/learn

[11] 刘庆.
孔雀计划：中文字体排印的思路 [EB/OL].
https://thetype.com/kongque

关于

本幻灯片：https://github.com/stone-zeng/latex-talk

许可证：Creative Commons Attribution-ShareAlike 4.0 International

Beamer 主题：萧山

正文字体：思源宋体 + Libertinus Serif

等宽字体：等距更纱黑体 + Iosevka

\bye