

# 深度易经

编者：不避风云





Deepin Bible

# 深度易经

深度操作系统爱好者 编著

尚未出版

版权 © 2018 深度操作系统爱好者

尚未出版

<https://github.com/bubifengyun/deepin-bible>

版权采用 Creative Commons Attribution-NonCommercial 3.0 Unported License, 内容见 <http://creativecommons.org/licenses/by-nc/3.0>。共享版权, 商业用途请联系我们。

尚未出版于 2018 年 07 月 10 日

## 前言

下面介绍为什么要写这本书？起源于去武汉参观辛亥革命博物馆。深深的被孙中山的“敢为天下先”所感动。另外也要感谢深度科技公司的 **jingle** 提供的帮助和支持，内心总是感觉深度科技就是敢为天下先的代表，俺也不能自甘落后，于是下决心写本书。

另外感谢深度公司还给资助了二百来块钱，太感谢啦，写在这里<sup>1</sup>的稿费。还有很多默默支持的小伙伴，在此一并表示感谢。

另外还要说一个中国早就存在的二大爷精神。虽然 60 多年过去了，你二大爷还是你二大爷。它代表了一种个性要强不怕输敢于拼搏的精神。也用于纪念我逝去的二大爷。写这本书，也表达对他的敬意和怀念。

本书的封面，是我的好友王一凡帮助设计的，特别表示感谢。相信大家一看到这两个动物就会明白所代表的啥意思。不再过多解释。

本书结构如下。打算模仿《Linux Bible》，采取深入浅出的方式，先介绍一些简单的使用方法，然后后面进行深入。

文件名采用如下方式，xxx-name.Rmd。其中 xxx 表示从 000 到 999，第一位表示第几部分，后两位表示章节。name 是对应章节的名字。一章一个文件。

本书编译时的软件版本信息。

```
1 ## R version 3.4.1 (2017-06-30)
2 ## Platform: x86_64-pc-linux-gnu (64-bit)
3 ## Running under: Deepin 15
4 ##
5 ## Matrix products: default
6 ## BLAS: /usr/lib/libblas/libblas.so.3.7.1
7 ## LAPACK: /usr/lib/lapack/liblapack.so.3.7.1
8 ##
9 ## locale:
10 ## [1] LC_CTYPE=zh_CN.UTF-8
11 ## [2] LC_NUMERIC=C
12 ## [3] LC_TIME=en_US.UTF-8
13 ## [4] LC_COLLATE=zh_CN.UTF-8
14 ## [5] LC_MONETARY=zh_CN.UTF-8
15 ## [6] LC_MESSAGES=zh_CN.UTF-8
16 ## [7] LC_PAPER=zh_CN.UTF-8
17 ## [8] LC_NAME=C
```

<sup>1</sup><https://gitee.com/bubifengyun/viewwuhan>

```
18 ## [9] LC_ADDRESS=C
19 ## [10] LC_TELEPHONE=C
20 ## [11] LC_MEASUREMENT=zh_CN.UTF-8
21 ## [12] LC_IDENTIFICATION=C
22 ##
23 ## attached base packages:
24 ## [1] stats      graphics  grDevices  utils      datasets
25 ## [6] base
26 ##
27 ## other attached packages:
28 ## [1] knitr_1.20
29 ##
30 ## loaded via a namespace (and not attached):
31 ## [1] compiler_3.4.1  backports_1.1.2 magrittr_1.5
32 ## [4] bookdown_0.7.13 rprojroot_1.3-2 htmltools_0.3.6
33 ## [7] tools_3.4.1     rstudioapi_0.7  yaml_2.1.19
34 ## [10] Rcpp_0.12.16    stringi_1.2.2   rmarkdown_1.10
35 ## [13] methods_3.4.1  stringr_1.3.1   digest_0.6.15
36 ## [16] xfun_0.3        evaluate_0.10.1
```

本书的代码规范。

- 文件和目录用斜体表示，也就是用 \* 括起来。
-

## 目 录

前言	I
插图索引	X
表格索引	XI
致谢	XIII
作者简介	XV
<b>第一部分 Linux 入门</b>	<b>1</b>
<b>第一章 GNU/Linux 操作系统简介</b>	<b>3</b>
1.1 GNU/Linux 使用范围 . . . . .	3
1.2 Linux 是什么 . . . . .	3
1.3 Linux 发行版的出现 . . . . .	4
1.4 如何提高自己的 Linux 技术 . . . . .	4
<b>第二章 Linux 桌面系统简介</b>	<b>5</b>
<b>第三章 深度操作系统简介</b>	<b>7</b>
3.1 深度科技简介 . . . . .	7
3.2 使用人群 . . . . .	8
3.2.1 程序员 . . . . .	8
3.2.2 文员 . . . . .	8
3.2.3 游戏爱好者 . . . . .	8
3.3 深度桌面使用 . . . . .	8
3.3.1 开机 . . . . .	9
3.3.2 关机 . . . . .	9
3.3.3 控制中心-系统设置 . . . . .	10
3.3.4 开机自启动 . . . . .	10

3.3.5	安装软件	10
3.3.6	卸载软件	11
3.4	深度桌面常用的几款软件	11
3.4.1	搜狗输入法	11
3.4.2	QQ	12
3.4.3	深度截图	12
3.4.4	网易云音乐	17
3.4.5	深度系统监视器	20
3.4.6	深度无线投屏	22
<b>第四章</b>	<b>教育学习用的 Linux</b>	<b>35</b>
<b>第二部分</b>	<b>其他的知识入门</b>	<b>37</b>
<b>第五章</b>	<b>shell 用法简介</b>	<b>39</b>
5.1	引言	39
5.2	用 shell 有啥好处?	39
5.3	深度操作系统下 shell 简介	39
<b>第六章</b>	<b>深度系统安装</b>	<b>43</b>
6.1	UEFI 和 legacy BIOS 区别和联系	43
6.2	优盘安装	43
6.2.1	第一步下载并校验 deepin.iso 文件	43
6.3	安装遇到的问题	44
<b>第七章</b>	<b>软件安装</b>	<b>45</b>
7.1	npm 软件的安装	45
7.2	veil 软件安装	45
7.3	dpkg 安装	46
7.4	其他安装方式	46
7.4.1	bitnami	46
7.4.2	turnkeylinux	46
7.5	you-get 软件的安装	46
7.6	字体的安装	47



7.7 输入法的安装 . . . . .	48
<b>第八章 管理账户</b>	<b>49</b>
8.1 引言 . . . . .	49
8.2 创建账户 . . . . .	49
8.2.1 图形方式 . . . . .	49
8.2.2 命令方式 . . . . .	52
<b>第三部分 安全</b>	<b>53</b>
<b>第四部分 附录及其他</b>	<b>55</b>
<b>附录甲 愚公移山</b>	<b>57</b>
<b>附录乙 如何制作本书</b>	<b>59</b>
乙.1 准备工作 . . . . .	59
乙.1.1 Linux 下使用 . . . . .	59
乙.1.2 Windows 下使用 . . . . .	63
乙.1.3 苹果操作系统下使用 . . . . .	63
乙.2 模板选项 . . . . .	63
乙.3 编译模板 . . . . .	63
乙.4 文件布局 . . . . .	64
乙.5 主要文件介绍 . . . . .	64
乙.5.1 L <sup>A</sup> T <sub>E</sub> X 模板文件 . . . . .	64
乙.5.2 各章源文件 . . . . .	64
乙.5.3 配置文件 . . . . .	65
乙.5.4 图片文件夹 figure . . . . .	65
乙.5.5 参考文献数据库 bib . . . . .	65
乙.5.6 辅助文件 . . . . .	65
乙.6 捐赠 . . . . .	66
<b>附录丙 各章格式说明</b>	<b>67</b>
丙.1 摘要 . . . . .	67
丙.2 摘要后的部分章节 . . . . .	68

丙.3 正文第一章 . . . . .	68
丙.4 正文后续章节 . . . . .	69
丙.5 全文总结 . . . . .	69
丙.6 附录第一章 . . . . .	69
丙.7 附录后续章节 . . . . .	69
丙.8 后缀部分 . . . . .	69
<b>附录丁 兼容 L<sup>A</sup>T<sub>E</sub>X 排版</b>	<b>71</b>
丁.1 列表环境 . . . . .	71
丁.1.1 无序列表 . . . . .	71
丁.1.2 有序列表 . . . . .	71
丁.1.3 描述型列表 . . . . .	71
丁.1.4 自定义列表样式 . . . . .	72
丁.2 数学排版 . . . . .	72
丁.2.1 公式排版 . . . . .	72
丁.2.2 SI 单位 . . . . .	72
丁.2.3 定理环境 . . . . .	73
丁.3 向文档中插入图像 . . . . .	74
丁.3.1 支持的图片格式 . . . . .	74
丁.3.2 长标题的换行 . . . . .	75
丁.3.3 添加图注 . . . . .	76
丁.3.4 绘制流程图 . . . . .	76
丁.4 表格 . . . . .	78
丁.5 参考文献管理 . . . . .	78
丁.6 用 listings 插入源代码 . . . . .	80
丁.7 用 algorithm 和 algorithmicx 宏包插入算法描述 . . . . .	81
<b>附录戊 RMarkdown/Bookdown 排版示例</b>	<b>85</b>
戊.1 Markdown syntax . . . . .	85
戊.1.1 Inline formatting . . . . .	85
戊.1.2 Block-level elements . . . . .	86
戊.1.3 Math expressions . . . . .	87
戊.2 Markdown extensions by bookdown . . . . .	88
戊.2.1 Number and reference equations . . . . .	89

---

戊.2.2 Theorems and proofs . . . . .	90
戊.2.3 Special headers . . . . .	95
戊.2.4 Text references . . . . .	96
戊.3 R code . . . . .	97
戊.4 Figures . . . . .	98
戊.5 Tables . . . . .	103
戊.6 Cross-references . . . . .	107
戊.7 Custom blocks . . . . .	108
戊.8 Citations . . . . .	111
戊.9 Index . . . . .	113
戊.10 HTML widgets . . . . .	113
戊.11 Web pages and Shiny apps . . . . .	116
<b>附录己 常见问题</b>	<b>119</b>
<b>附录庚 大事记</b>	<b>121</b>
<b>索引</b>	<b>123</b>
<b>参考文献</b>	<b>125</b>



## 插图索引

3-1 深度公司图片	7
3-2 启动项	9
3-3 任务栏打开软件动画	10
3-4 文件管理器	11
3-5 文件管理器	12
3-6 文件管理器	13
3-7 关机	13
3-8 控制中心	14
3-9 开机自启动	14
3-10 深度商店	15
3-11 深度截图中文版	15
3-12 深度截图中文版	16
3-13 深度截图中文版	17
3-14 深度截图中文版	18
3-15 深度截图中文版	19
3-16 网易云音乐	19
3-17 网易云音乐	20
3-18 网易云音乐	21
3-19 网易云音乐	22
3-20 网易云音乐	23
3-21 网易云音乐	24
3-22 网易云音乐	25
3-23 网易云音乐	26
3-24 深度系统监视器	26
3-25 深度系统监视器	27
3-26 深度系统监视器	28
3-27 深度系统监视器	29
3-28 深度系统监视器	30
3-29 无线投屏	31

3-30 无线投屏 . . . . .	32
3-31 无线投屏 . . . . .	33
8-1 图形方式创建账户 . . . . .	51
8-2 图形方式配置账户 . . . . .	51
8-3 图形方式配置账户 . . . . .	52
乙-1 支付宝红包及收款的二维码 . . . . .	66
丁-1 这里将出现在插图索引中 . . . . .	74
丁-2 插入 eps 和 pdf 的例子 (使用 subcaptionbox 方式) . . . . .	74
丁-3 插入 eps 和 pdf 的例子 (使用 subfigure 方式) . . . . .	75
丁-4 这里将出现在插图索引 . . . . .	75
丁-5 出现在插图索引中 . . . . .	75
丁-6 出现在插图索引中 . . . . .	76
丁-7 绘制流程图效果 . . . . .	77
戊-1 A figure example with the specified aspect ratio, width, and alignment . . . . .	100
戊-2 A figure example with a relative width 70% . . . . .	100
戊-3 Two plots placed side by side . . . . .	101
戊-4 Three knitr logos included in the document from an external PNG image file . . . . .	102
戊-5 A table widget rendered via the DT package . . . . .	114
戊-6 A Shiny app created via the miniUI package . . . . .	117

## 表格索引

丁-1 指向一个表格的表目录索引 . . . . .	78
丁-2 出现在表目录的标题 . . . . .	78
戊-1 Theorem environments in <b>bookdown</b> . . . . .	91
戊-2 A table of the first 10 rows of the mtcars data. . . . .	103
戊-3 A Tale of Two Tables. . . . .	104
戊-4 A table generated by the longtable package. . . . .	104





## 致谢

首先感谢讯飞语音输入法，用他来输入汉字实在是太快捷了。这样我才可能在上班的时候，掏出手机，看着书，发表发表感慨，也不至于被写字给打断，就把想做的笔记给录下来了，变成汉字了。稍作调整，就可以当做 issue 记录下来。还有这等好事，对就是有这等好事。

其次感谢开源中国提供的手机客户端，我可以随时掏出手机做笔记。虽然不够完美，但对于积累临时的想法还是大有好处的。

再次感谢 bookdown 的开发者，谢益辉和他的小伙伴们，提供了这么好的一个做笔记的模板。

当然也要感谢《Linux Bible》的作者，以及为 Linux 做出贡献的那些人。

也有深度科技公司，谢谢他们给提供了这么好的一个操作系统。

也要谢谢各位支持我的好友、网友和亲人。给了我很大的动力，让我开启这个活动。能不能坚持下去，相信自己。



## 作者简介

这将是是一群深度操作系统爱好者的杰作!!!

我一直致力于寻找合作者，但是我感觉现在有点迷茫。这边进度实在是慢，常常一个月写不了多少内容，可能一个月就几百字，都写不到一页内容。经常停下来。我只是想告诉别人，这个事业还在继续中。我却找不到一点点进步的身影。

现在（2017-10-29）写作进行的很困难，主要是零碎的时间很难用于整理大篇大篇的文字。还有其他繁重的任务要做，似乎不太容易写。我觉得可以分析一下困难，研究一下对策。不要再让这本书半途而废了。现在遇到的问题，负责带教，挤占了较多的时间，一时半会，他们也帮不上忙。还有两个网站的内容在整理，不可荒废。包括驾校也要报名。结婚的事情，还有一大堆。也就是有这么几件事，婚前准备，两个网站的维护，驾校考试，带教学习，完成本书第一版草稿，基本就是这么个优先级顺序，当然时刻伴随着要上班这个无法逃脱的事情。鉴于此，我认为接下来可以这样安排。婚前准备虽然优先级高，但是实际占场时间不多，如果可以快速下决定，花不了太长时间的。两个网站的维护，基本上要浪费掉很多晚上的时间。可以每天抽出两个小时的时间，弄一弄。驾校的事情，周末可以去试试，也可以在工作间隙去练练。带教学习，也是很耗费时间的，每次可能要一到两个小时，一周也就两到三次。可以跟网站维护交叉进行。完成本书第一版草稿，更是要耗费几年时间的任务。另外带教学习的就是本书，可以充分利用这个带教的事情，毕竟有徒弟可以帮着写写画画的。制定这么一个时间表，有时间里面把婚前准备的事情给结束掉，避免多次设想，带来的巨大时间浪费。每周抽出两到三天晚上带教，可以做好任务安排，每完成一章课程，作业就是帮我整理本草稿。剩下的两到三天去编写网站代码，做好维护，且不再增加新的网站或者其他任务了。驾校的事情，暂且缓一缓，找个时间去把科目一考了。

终于领结婚证了（2017-11-03），我现在觉得写作特别耗费时间，需要“庙算”一下，好好规划一下，要写哪些内容。github的一些功能是非常值得充分利用的，比如 issue，这样可以充分利用零碎时间。半夜我坐在一个孤独的房间，又不看书，怎么可能写好这个笔记。必须边学习边做笔记，也就是写到 issue 上。不要手机整天微信、支付宝了。当前贡献者：黄煌、贺鹏飞、董春柏、邱鹏飞，还有参考网页的一些好友，抱歉没来得及写在这里，以后慢慢添加，在此表示感谢。



# 第一部分

## Linux 入门

“合抱之木，生于毫末；九层之台，起于垒土；千里之行，始于足下。”

—— 老 子

第一章 GNU/Linux 操作系统简介	3
第二章 Linux 桌面系统简介	5
第三章 深度操作系统简介	7
第四章 教育学习用的 Linux	35

这里需要写点啥的。



## 第一章 GNU/Linux 操作系统简介

通过阅读本章，你将会了解到以下几项内容。

- Linux 使用范围。
- Linux 发展简史以及 GPL 协议。
- 如何学习 Linux。

### 1.1 GNU/Linux 使用范围

有人说，Linux 作为服务器操作系统，在市场上占有绝对优势，在超级计算机前 500 名中更是占有垄断地位。好吧，这些都不是我们关心的事情。我们关心的是 Linux 能在我用着方便的领域能占多大范围。**如果你是程序员**。你可以用它方便的写代码，并很容易调试。有功能异常强大的终端，供你跟整个世界沟通。方便你写文档。当然也可以用它来听听音乐，看看电影啦。**如果你只是在家里使用**。比如趣味性，听听音乐，看看电影，这些都是可以办得到的。那办公呢，搜狗拼音输入法，赫然在列，金山的 WPS 办公套件也是能满足你基本需求的。那 QQ 呢，这个也是必须有的，能不能告诉你，在深度 Linux 已经实现视频聊天了呢。那打游戏呢？很不幸的告诉你，Linux 操作系统下能玩的游戏真心不多，你如果特别喜欢玩游戏，还是不建议你在 Linux 上玩啦。**如果你是办公室汉字作业者**。那办公呢，搜狗拼音输入法，赫然在列，金山的 WPS 办公套件也是能满足你基本需求的。那 QQ 呢，这个也是必须有的，能不能告诉你，在深度 Linux 已经实现视频聊天了呢。当然还有很多特殊的办公软件在 Linux 下没有可用的版本。比如 photo shop，autoCAD 等。不过也有对应的 Linux 开源版本。

另外 Linux 桌面操作系统，稳定性不是很高，但也没有 Windows 下全家桶系列乱弹窗的软件。如果你对桌面清洁有特殊爱好，且 Linux 能满足你日常生活需要，不妨来试试它啦。

### 1.2 Linux 是什么

什么？其实他就是一个电脑操作系统啦。那操作系统是啥，那我也不知道啦，这个你可以百度一下，简单的说就是方便你操作电脑的一个软件系统。关于 Linux 有啥发展历史，以及跟 GPL 协议有啥关系。这些网上都有铺天盖地的介绍，本书就不介绍啦。

不过还是要介绍两个人，一位是 GNU 的老大 Richard Stallman<sup>1</sup>，一位是 Linux 的老大 Linus<sup>2</sup>。

### 1.3 Linux 发行版的出现

### 1.4 如何提高自己的 Linux 技术

好好学习，天天向上。毛爷爷如是说。有人说学习 Linux 要不断的折腾。

---

<sup>1</sup><http://www.stallman.org/>

<sup>2</sup>[https://en.wikipedia.org/wiki/Linus\\_Torvalds](https://en.wikipedia.org/wiki/Linus_Torvalds)



## 第二章 Linux 桌面系统简介

随着智能手机平板的发展，移动操作系统趋于完善。电脑桌面操作系统，虽然不能说是夕阳产业，但也趋向一个完美的天花板。我认为很多在 windows 和苹果桌面操作系统下面能操作的软件和体验，linux 桌面会慢慢的赶上来的，可以使用的软件也会越来越多，软件的种类也越来越齐全。我们已经可以在 linux 桌面下做很多工作。Linux 桌面有如下几种，全特性的 GNOME 和 KDE 等，轻量级的 LXDE 和 Xfce，以及本书要讲的深度桌面 DDE(Deepin Desktop Environment)。

下面将介绍 linux 桌面技术。

几乎所有的电脑桌面操作系统都会提供图形界面供用鼠标和键盘操作。当然一些服务器只有终端接口的除外。

当前主要的 linux 桌面发行版基本都是采用 x window 系统 (http)，来提供桌面服务。以及 Wayland 。这部分先完全摘抄，当了解这部分知识后，再用自己的语言表述。

接下来的介绍建议从网上复制过来，还有王勇的一篇文章也可以拿过来放在最后面供参考。这些内容如果纯用手机来输入的话，非常不方便，效率比较低下。

参考：

- <https://baike.baidu.com/item/X%20Window/7249336?fr=aladdin>
- <https://baike.baidu.com/item/Wayland/7429696?fr=aladdin>



## 第三章 深度操作系统简介

摘自：深度商业官网<sup>1</sup>、深度社区官网<sup>2</sup>。已获得许可<sup>3</sup>。

### 3.1 深度科技简介

武汉深之度科技有限公司（简称：深度科技）成立于 2011 年，是专注基于 Linux 的国产操作系统研发与服务的商业公司。

作为国内顶尖的操作系统研发团队，深度科技以提供安全可靠、美观易用的国产操作系统与开源解决方案为目标，拥有操作系统研发、行业定制、国际化、迁移和适配、交互设计、支持服务与培训等多方面专业人才，能够满足不同用户和应用场景对操作系统产品的广泛需求。



图 3-1 深度公司图片

<sup>1</sup><https://www.deepin.com/>

<sup>2</sup><https://www.deepin.org/>

<sup>3</sup><https://bbs.deepin.org/forum.php?mod=viewthread&tid=39632&page=10#pid377474>

深度科技作为国产操作系统生态的打造者，不但与各芯片、整机、中间件、数据库等厂商结成了紧密合作关系，还与 360、金山、网易、搜狗等企业联合开发了多款符合中国用户需求的应用软件。深度科技的操作系统产品，已通过了公安部安全操作系统认证、工信部国产操作系统适配认证、入围国管局中央集中采购名录，并在国内党政军、金融、运营商、教育等客户中得到了广泛应用。

截止到 2016 年，深度操作系统下载超过 5000 万次，提供 32 种不同的语言版本，以及遍布六大洲 28 个国家 84 个镜像站点的升级服务。在全球开源操作系统排行榜上，深度操作系统已跃居前十名，也是排名最高的中国操作系统产品。未来，深度科技将继续秉承开源和创新精神，以操作系统的自主可控替代为契机，争取在三年内成为中国市场主要操作系统供应商，五年后成为具备国际影响力的主流操作系统厂商。

注：深度论坛网友 [licardo<sup>1</sup>](#) 建议，国产操作系统改为国内发行版。深以为然。但是考虑到转帖的别人的文章，就保留不变了，特此说明。

## 3.2 使用人群

### 3.2.1 程序员

作为一个写过代码的人，用深度 Linux 感觉自己的电脑就是自己的。可以尽情写代码，不受各种弹窗的干扰。四个桌面，开发网站时，左右逢源，一个放置待测试网页，一个放置 vim + 源代码，一个 vscode 做笔记，一个打开浏览器查询各种资料。

### 3.2.2 文员

我感觉对于办公的最基本的需求，搜狗输入法和 WPS 是提供了的。

### 3.2.3 游戏爱好者

深度自带 steam 游戏平台，好多游戏可以畅玩。快来玩玩吧。

## 3.3 深度桌面使用

(本节仿照《Linux Bible》第二章内容书写。)

---

<sup>1</sup><https://bbs.deepin.org/forum.php?mod=viewthread&tid=145309&page=1#pid378170>

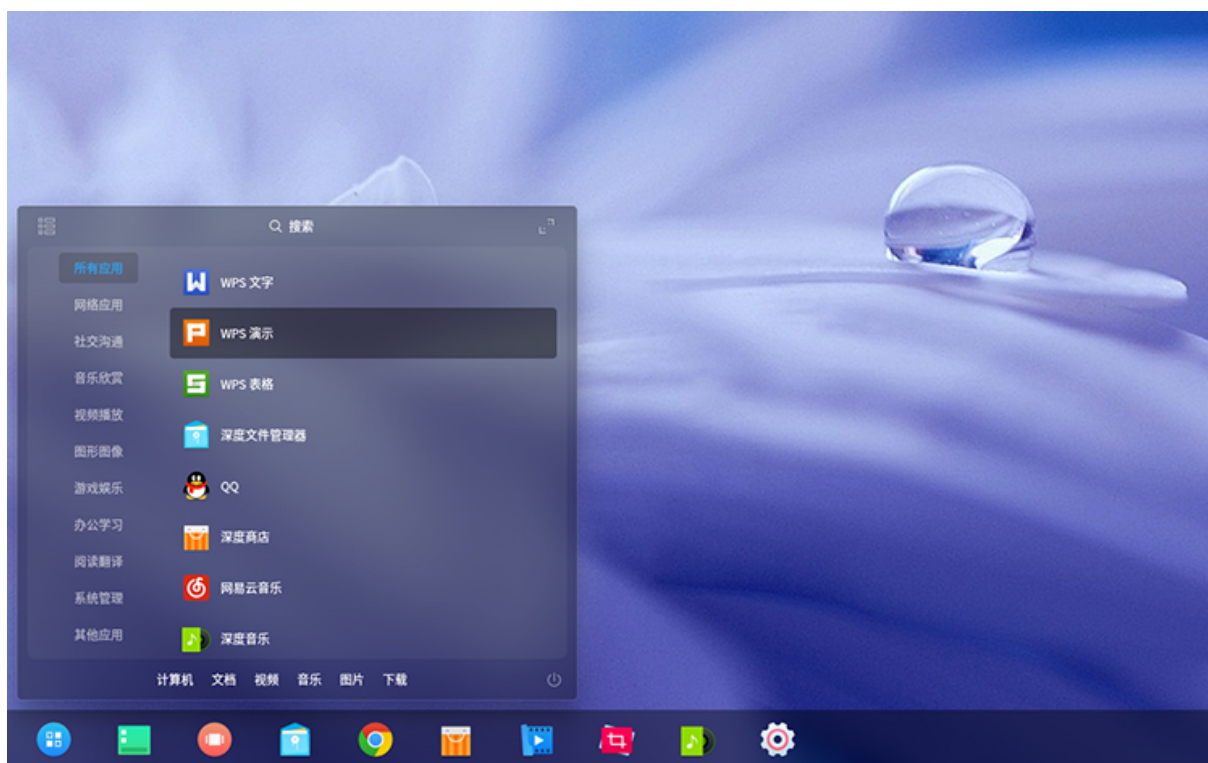


图 3-2 启动项

### 3.3.1 开机

如果你使用过 Windows 操作系统，那么使用深度桌面操作系统也不是难事。这里主要讲社区版深度桌面操作系统。开机登录后，点击 Windows 键会弹出如下界面。

你可以对所需要的软件进行搜索。直接点击软件图标，就可以打开软件了。分类放置的软件。在任务栏驻留的软件打开时，还会有动画显示。

#### 深度文件管理器

双击桌面上的计算机图标，会弹出深度文件管理器，用于打开各类文件。

可以使用 Super + 左右箭头或者 Super + S 切换桌面，Alt + tab, Super + W 切换窗口。

### 3.3.2 关机

同时按 Ctrl + Alt + Delete，或者点击右下角关机按钮，就会弹出关机界面，如下图所示。

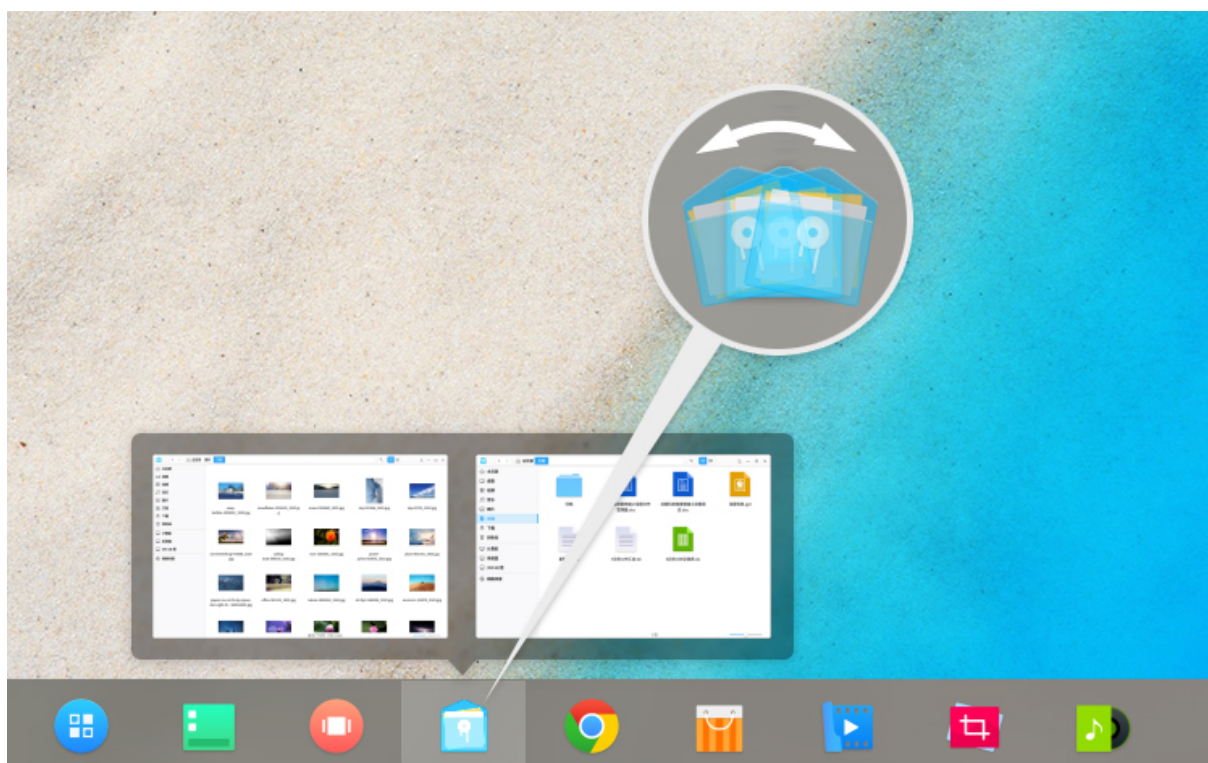


图 3-3 任务栏打开软件动画

### 3.3.3 控制中心-系统设置

点击任务栏，或者鼠标滑向桌面右下角几次，会在右侧出现控制中心面板。如下图所示。

首页展示快速入口、常用快捷设置等，让操作更加方便和快捷；天气详情、通知中心通过插件展示，今后会开放插件接口，让您可以更多的参与定制。

### 3.3.4 开机自启动

一些软件需要开机自启动，比如 QQ 还有一些保护眼睛的软件，比如定时休息的 workrave，减少蓝光的 red-shift。点击 Super（Windows）键，弹出所有的软件图标，看中要开机自启动的，右击选择开机启动。

### 3.3.5 安装软件

使用电脑，不可避免的要用到很多软件。深度商店为您解忧愁，方便您查找并安装相关的软件。当然除了深度商店，您还可以使用命令行安装软件，关于命令行安装软件

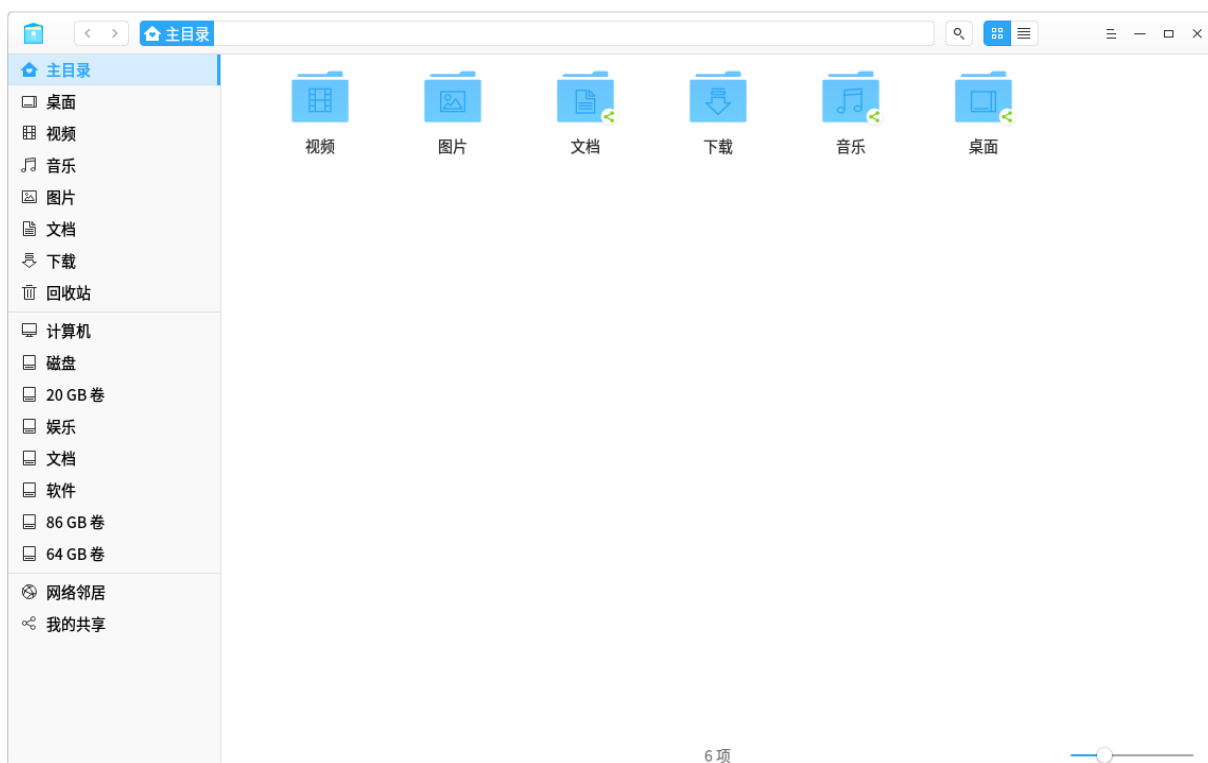


图 3-4 文件管理器

的介绍详见后面章节。

### 3.3.6 卸载软件

按 windows 键，弹出的软件，右击选择卸载。或者命令行，这里就不介绍了，在后面详细说明。

## 3.4 深度桌面常用的几款软件

下面这些软件，非常适合普通人使用了。特摘抄下来。

### 3.4.1 搜狗输入法

这个是系统默认自带的了，贴它是为了怀念曾经被输入法折磨的岁月。官网有详细介绍：<http://pinyin.sogou.com/linux/?r=pinyin>

不过我最希望有讯飞语音输入法，这个如果可以在 Linux 桌面下生存，是最好不过的。不过当前还没有 Linux 版本。

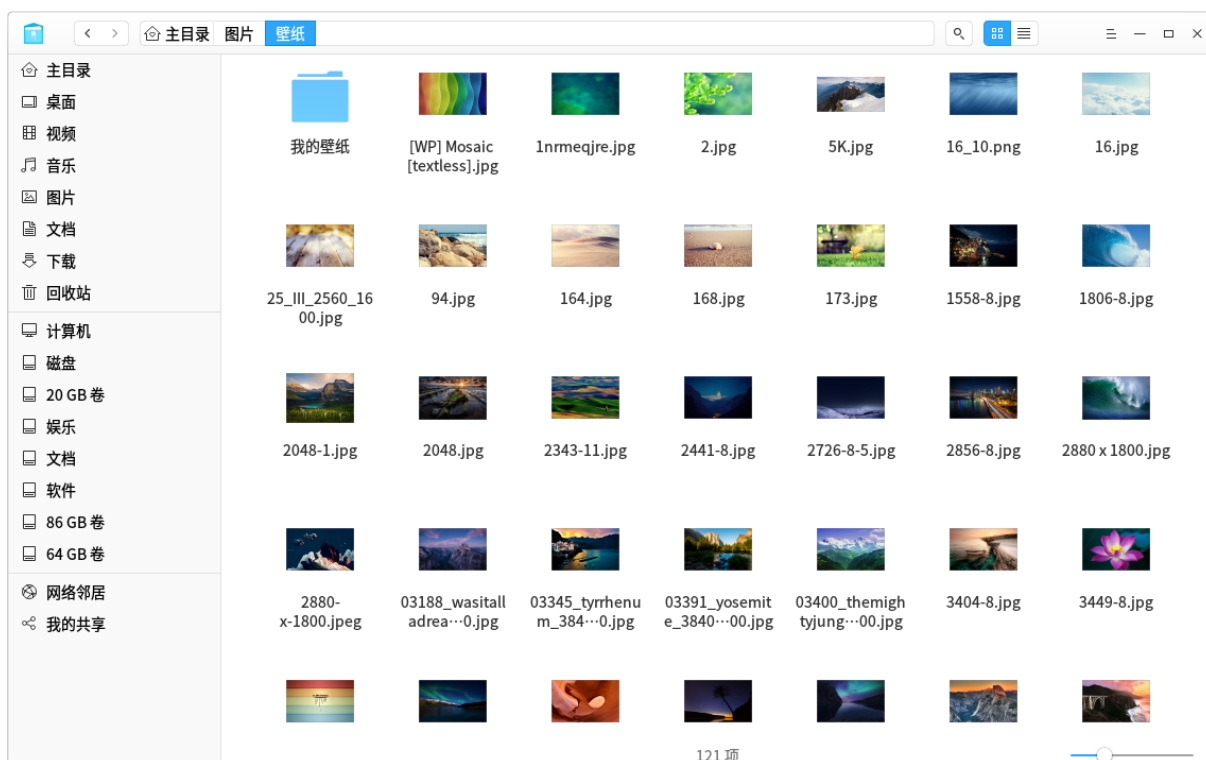


图 3-5 文件管理器

### 3.4.2 QQ

国人挚爱。可以视频。就不废话了。命令行安装。

```
1 $ sudo apt-get update
2 $ sudo apt-get install deepin.com.qq.im
```

### 3.4.3 深度截图

深度截图是深度科技团队开发的一款精巧截图应用，它具有智能窗口识别、快捷键支持、图片编辑、延迟截图、智能保存、调节图像分辨率等功能。深度截图 V4.0 作为系统预装的截图工具，继承沿用了之前版本的大部分功能，对代码进行了重构和优化，截图打开速度真正做到秒开。全新设计了截图工具栏和选取框，新增直接截取右键菜单界面等功能。

#### 重构优化，速度秒开

采用 C++ 重构代码，提升打开截图启动速度，优化性能和交互体验，从此告别打





图 3-6 文件管理器

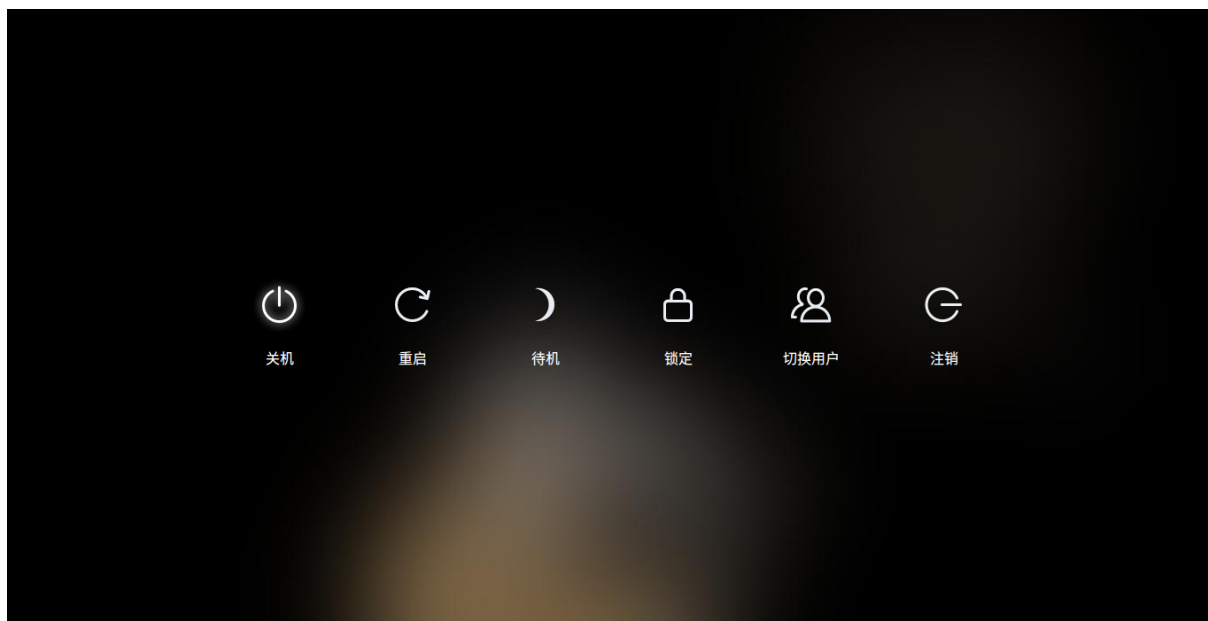


图 3-7 关机

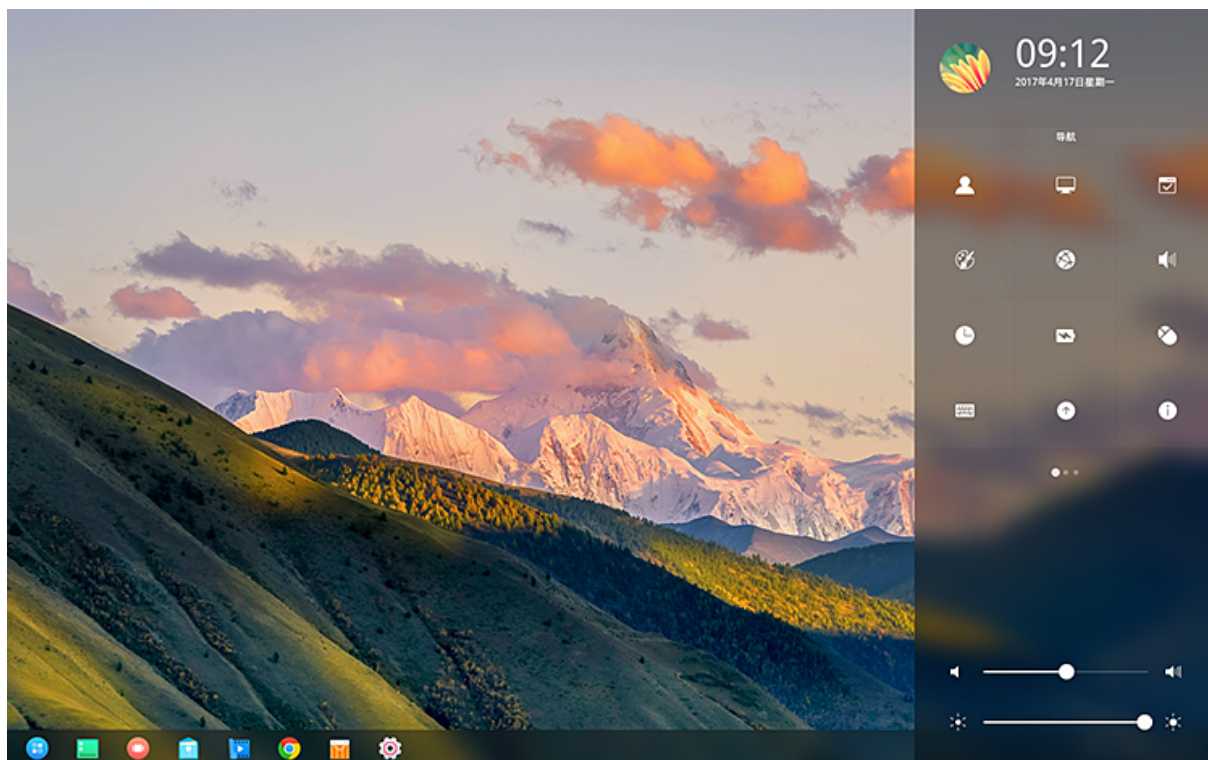


图 3-8 控制中心



图 3-9 开机自启动

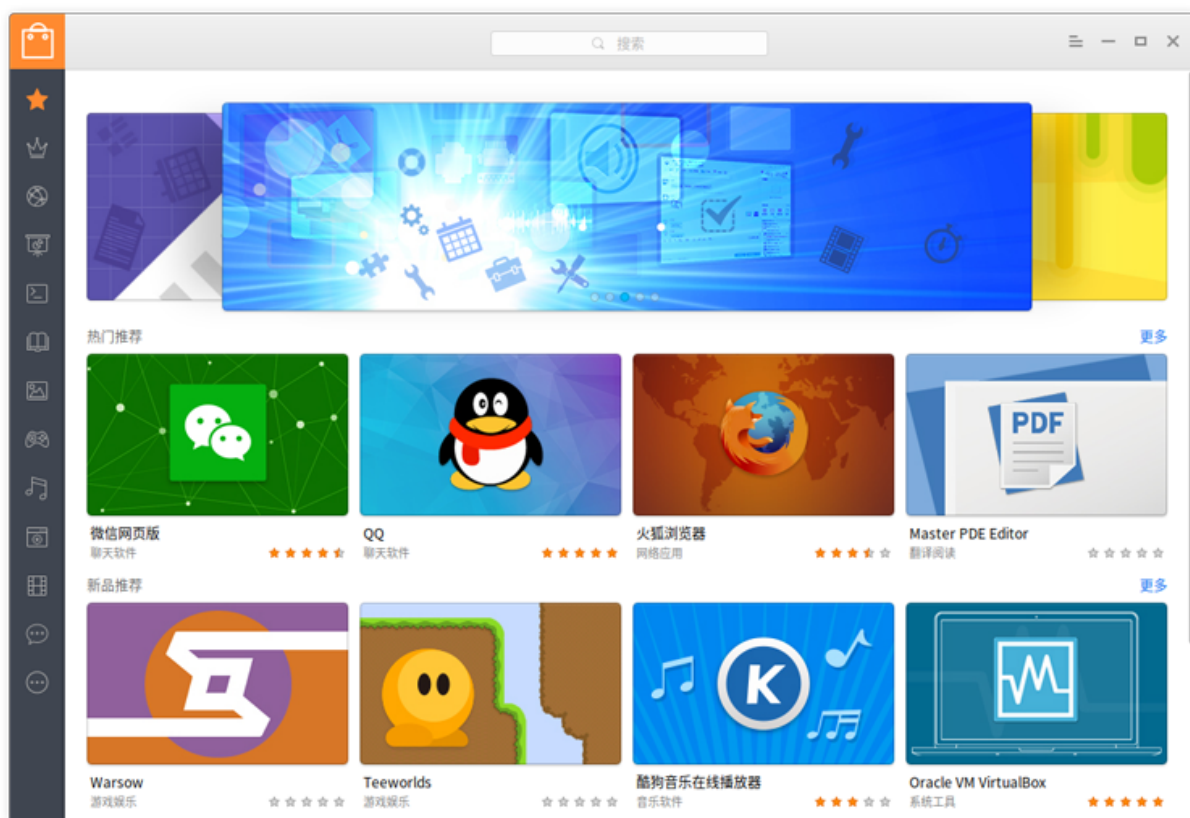


图 3-10 深度商店



图 3-11 深度截图中文版

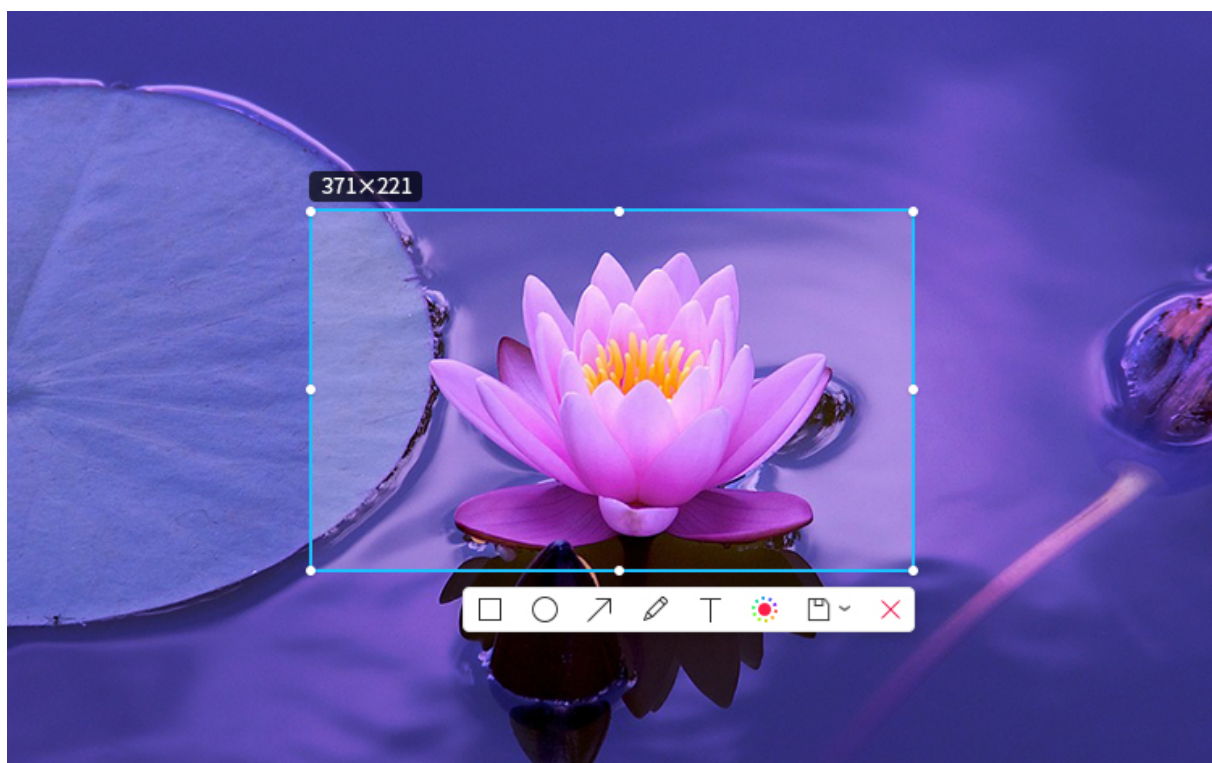


图 3-12 深度截图中文版

开慢、截图卡死的现象，真正做到秒开。

#### 统一选框，白色主题

优化选取框控件，统一系统界面交互和显示；默认将工具栏变换成白色主题，操作更加清晰明了。

#### 右键截取，一键即达

除了全屏截图、窗口截图、延时截图之外，本次新增右键菜单界面直接截图。无需设置延时后操作等待截取，只需一键便可截取右键菜单界面。

#### 化繁为简，效率为王

打开深度截图后，按下 F1 键直接调用帮助手册，简化旧版本调用帮助的交互。专注截图操作，取消分享功能，提高效率和避免出现问题。

#### 轻松下载，简单更新

大家可以通过更新系统以获取深度截图 V4.0，或者直接在深度商店搜索下载。也可以采用命令行安装。

```
1 $ sudo apt-get install deepin-screenshot
```

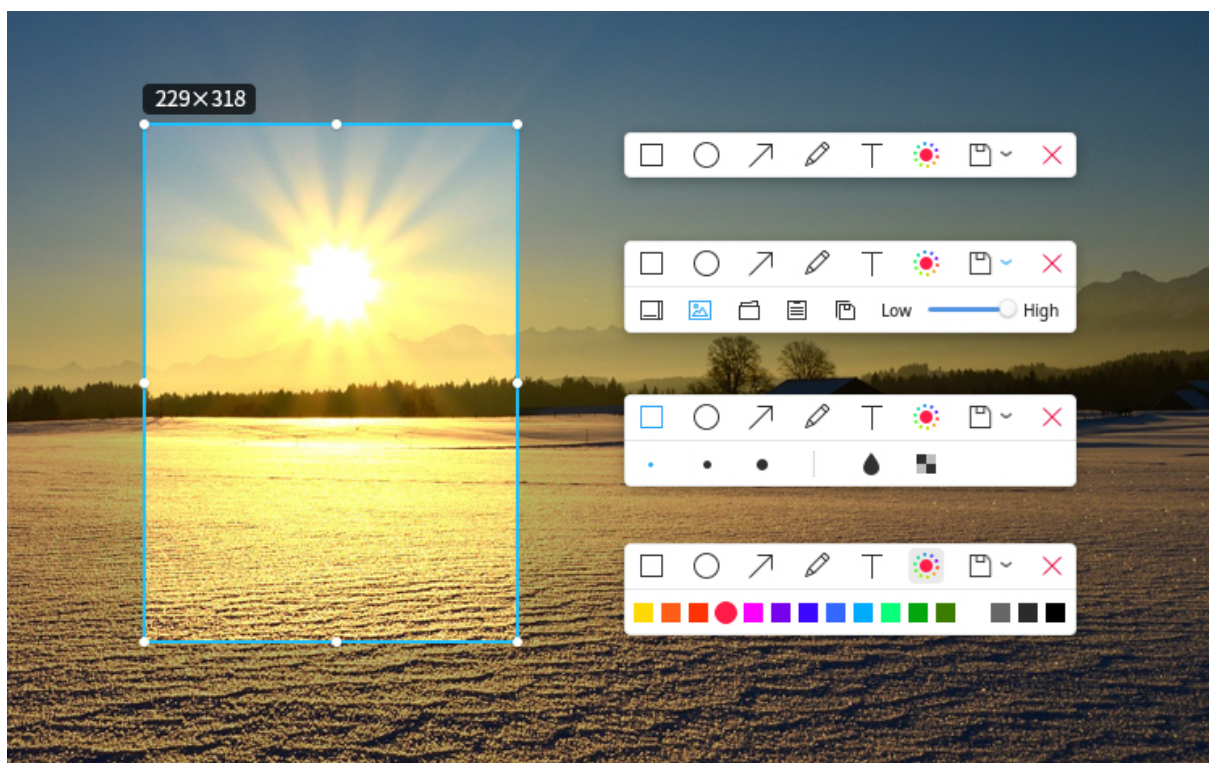


图 3-13 深度截图中文版

#### 3.4.4 网易云音乐

为了带来更好的音乐体验，实现对音乐高品质的追求，经过网易云音乐与深度科技团队长达半年多的联合开发，大家期待已久的网易云音乐正式登陆 Linux 平台！

网易云音乐是一款专注于发现与分享的音乐产品，依托专业音乐人、DJ、好友推荐及社交功能，为用户打造全新的音乐生活。

##### 时尚界面传承最新设计

完美继承了 Windows 版本的最新界面风格和功能，不同的平台，同样的用户体验。

##### 千万曲库首首超高品质

网易云音乐拥有广袤的曲库，囊括千万首 320Kbps 超品质音乐，让你随时沉醉在美妙的音乐层次中。

##### 明星入驻亲自推荐音乐

过千名大牌明星、专业音乐人、DJ 进驻，专业的私房“歌单”和音乐推荐，格调顿现。

##### 轻松社交引发内心共鸣



图 3-14 深度截图中文版

你可以关注明星、DJ 和好友，浏览他们的动态、收藏和分享；还可以通过其他人的评论体会歌曲中的故事和感受。

### 平台互通歌单实时同步

用户只需一个帐号，无论在哪个版本上收藏歌曲或创建歌单，只要连上网络，便可立即同步自己的数据信息，随时随地畅享好音乐。

### 个性推荐打造私人定制

网易云音乐采用独一无二的算法，轻易邂逅那些你未听过又很喜欢的歌。【个性化推荐】和【私人 FM】，比你更了解你。同时也是首个以“歌单”为核心架构的音乐产品，一步轻松创建“歌单”，音乐分享又容易又有趣。

### 轻松获取开启音乐之旅

深度操作系统用户可运行深度商店，搜索“网易云音乐”进行安装体验，非深度操作系统用户可前往网易云音乐官网下载 Linux 客户端安装体验。

深度操作系统是一个致力于为全球用户提供美观易用、安全可靠的 Linux 发行版。建立完善的生态系统，打造更多的原生应用，给用户更好的体验，一直都是我们前进的方向。此次和网易云音乐合作，给 Linux 平台下的用户也带来高品质的在线音乐享受，

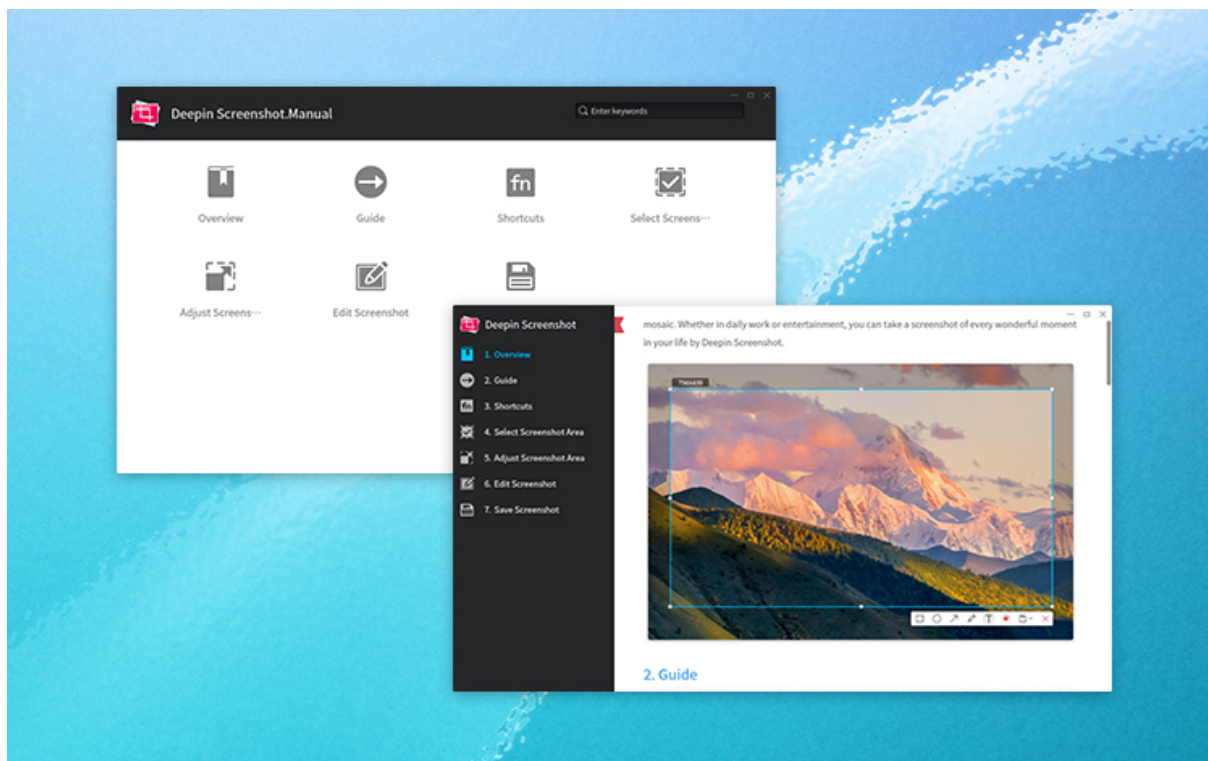


图 3-15 深度截图中文版



图 3-16 网易云音乐



图 3-17 网易云音乐

而后续我们将不断努力开拓更多更好的应用。可以采用命令行安装。

```
1 $ sudo apt-get install netease-cloud-music
```

### 3.4.5 深度系统监视器

深度家族又添加了一个新成员，**深度系统监视器 V1.0** 正式发布啦！

深度系统监视器是深度科技团队打造一款直观易用的系统监视器应用，它可以实时监控处理器状态、内存占用率、网络上传下载速度；还可以管理您的系统进程和应用进程，支持搜索进程和强制结束进程。

#### 黑白主题，简洁直观

集成黑白两种主题，用户可以自由切换；统一设计界面，布局更加简洁直观，实时动态显示处理器、内存、网络占用率以及进程列表等信息。

#### 标签显示，快速定位



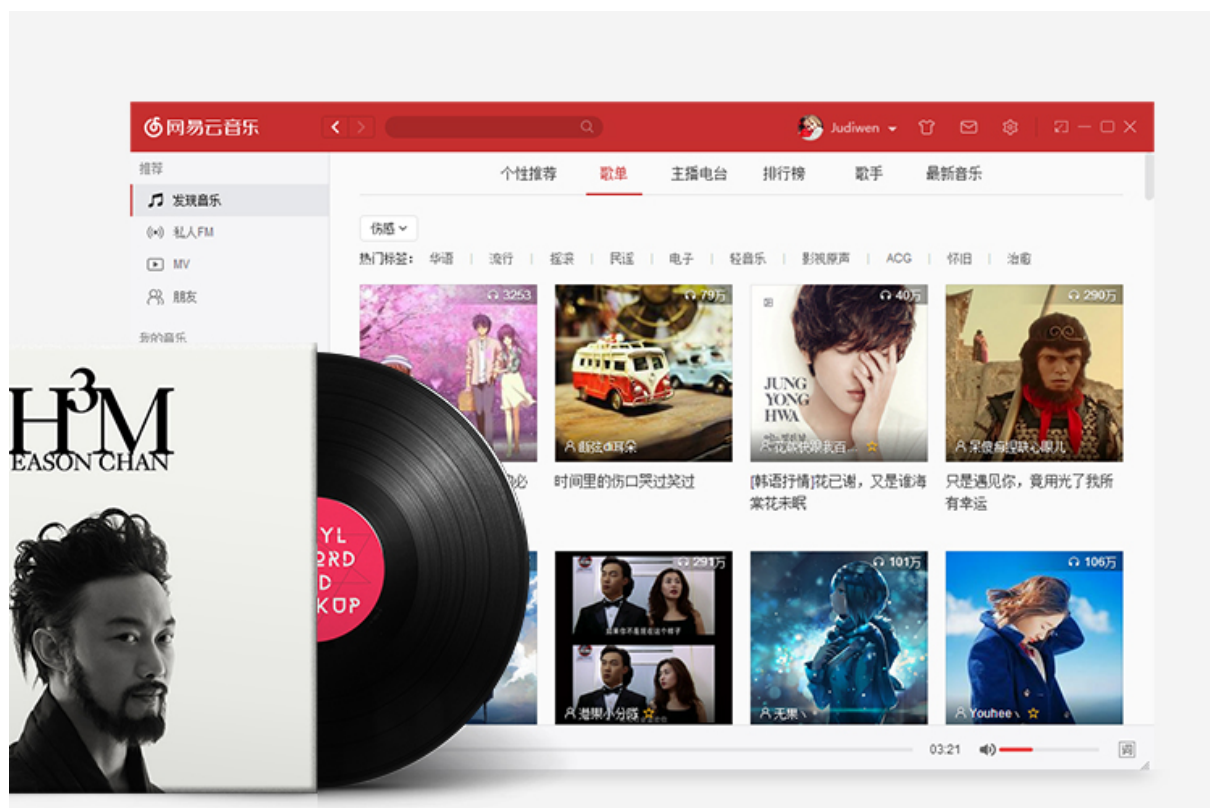


图 3-18 网易云音乐

通过标签分类的方式显示：应用程序进程、我的进程和所有进程，快速切换自己想要的进程显示；同时应用程序进程名称国际化处理，进程名称一目了然。当需要快速定位到某个进程时，可以快速搜索定位。

### 列表展示，高效右键

系统的进程采用列表方式展示，可以自定义处理器、内存、磁盘写入、磁盘读取、下载、上传、进程号是否显示，还可以根据列表排序显示；同时对进程还可以右键菜单快速操作。

### 捕捉窗口，即点即“杀”

当在使用系统的过程，不知道进程的 ID 或者想直接结束某个应用进程，只需点击菜单中强制结束进程选项，自动采用红色透明遮框捕捉窗口，点击即可结束进程。

大家可以通过更新系统以获取深度系统监视器 V1.0，或者直接在深度商店搜索下载。也可以采用命令行安装。

```
1 $ sudo apt-get install deepin-system-monitor -y
```

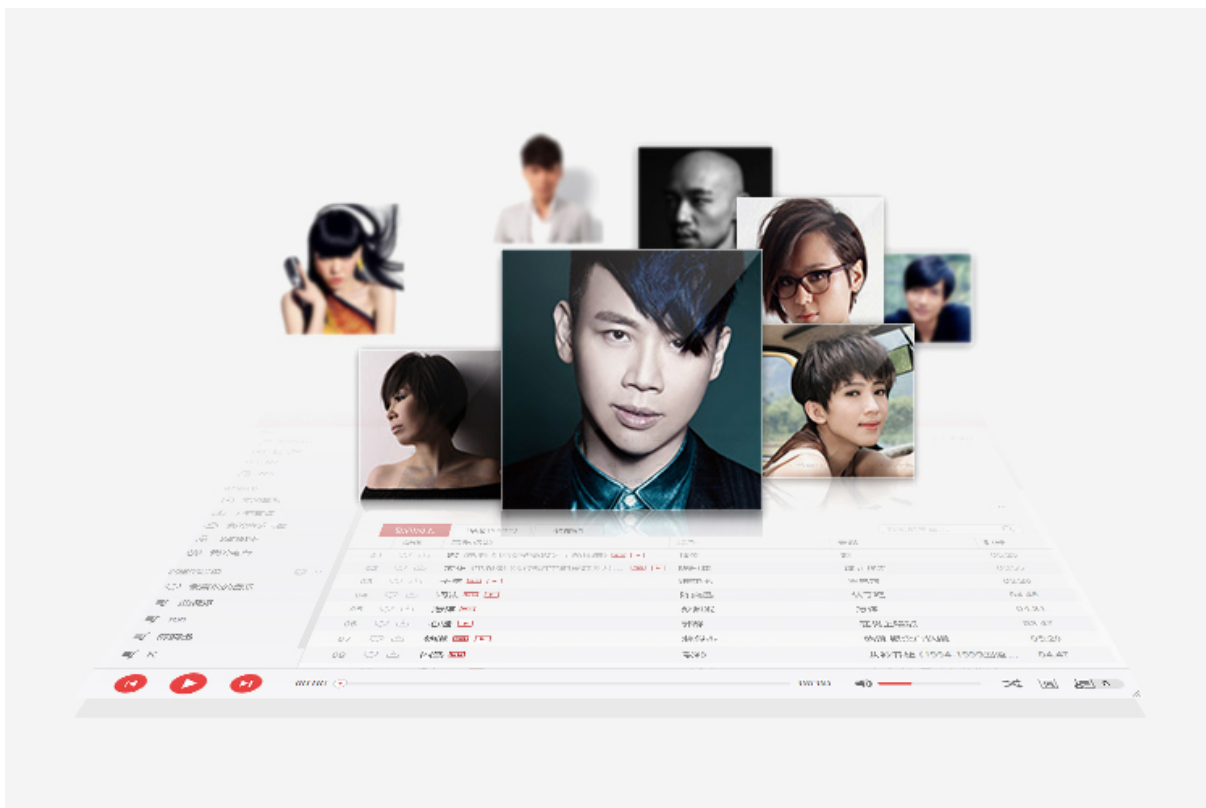


图 3-19 网易云音乐

### 3.4.6 深度无线投屏

深度家族又添加了一个新成员，深度演示助手 V1.0 正式发布啦！

深度演示助手是一款用于手机内容分享的演示工具。它支持照片分享功能，分享时可以对照片进行旋转、标记、聚焦等操作。甚至还可以直接将手机屏幕投射到电脑上，视频、游戏都轻松分享。另外还支持幻灯片远程控制，让您在演示幻灯片时无拘无束。

#### 无线投射，大屏享受

完美将小屏幕投射到大屏幕上，在投影仪等设备上看电影、玩游戏时感受大屏上面带来的视觉体验。

#### PPT 演示，快人一步

手机即是 PPT 遥控器，自动检测识别电脑端是否存在打开 PPT 文件。不仅是翻页工具，更能掌控全场，从此演示快人一步。

#### 照片分享，尽情涂鸦

让真实的生活完美展现，快速分享手机上的照片到大屏幕，还可以对照片进行涂

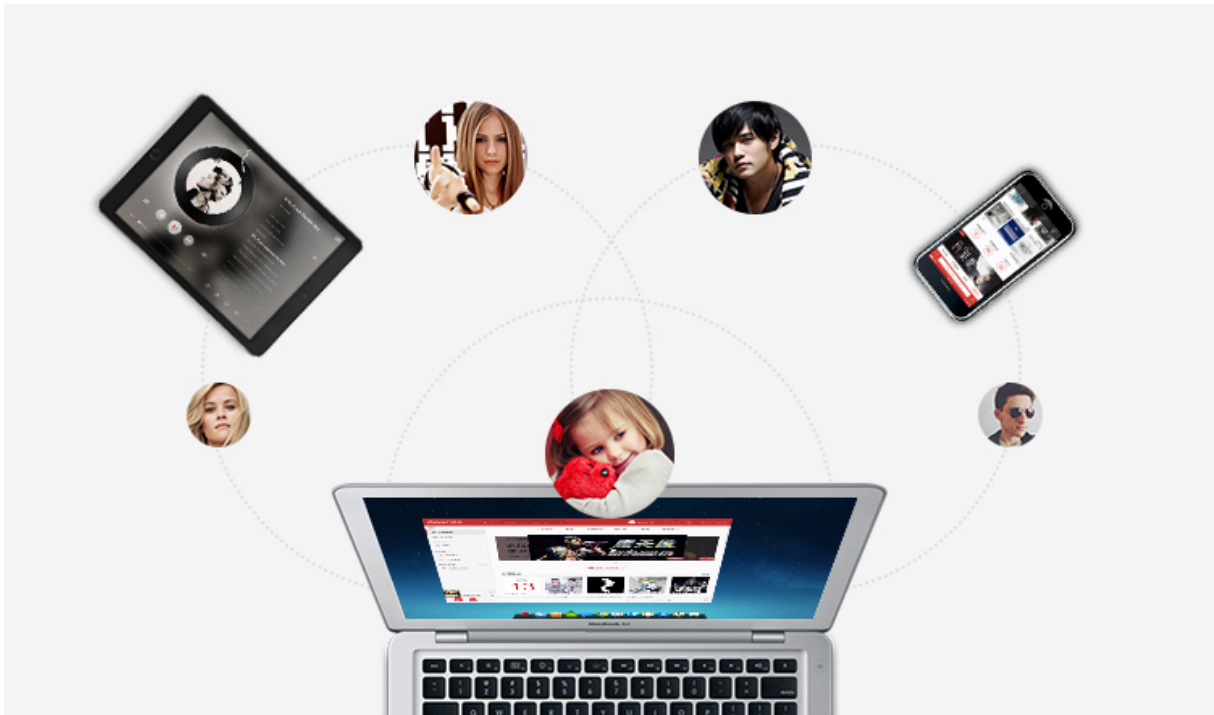


图 3-20 网易云音乐

鸭、编辑等处理。

大家可以通过深度商店搜索下载获取深度演示助手 V1.0。或者采用命令行安装。

```
1 $ sudo apt-get install deepin-presentation-assistant -y
```



图 3-21 网易云音乐



图 3-22 网易云音乐

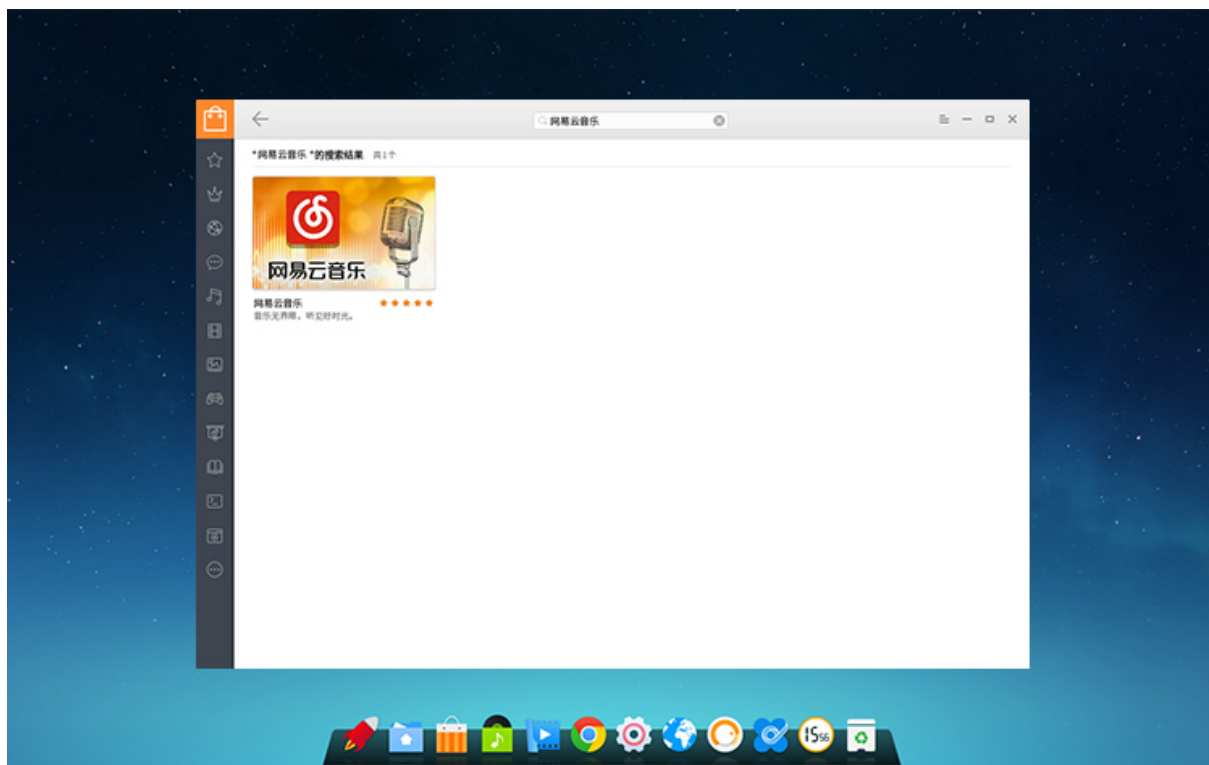


图 3-23 网易云音乐



图 3-24 深度系统监视器

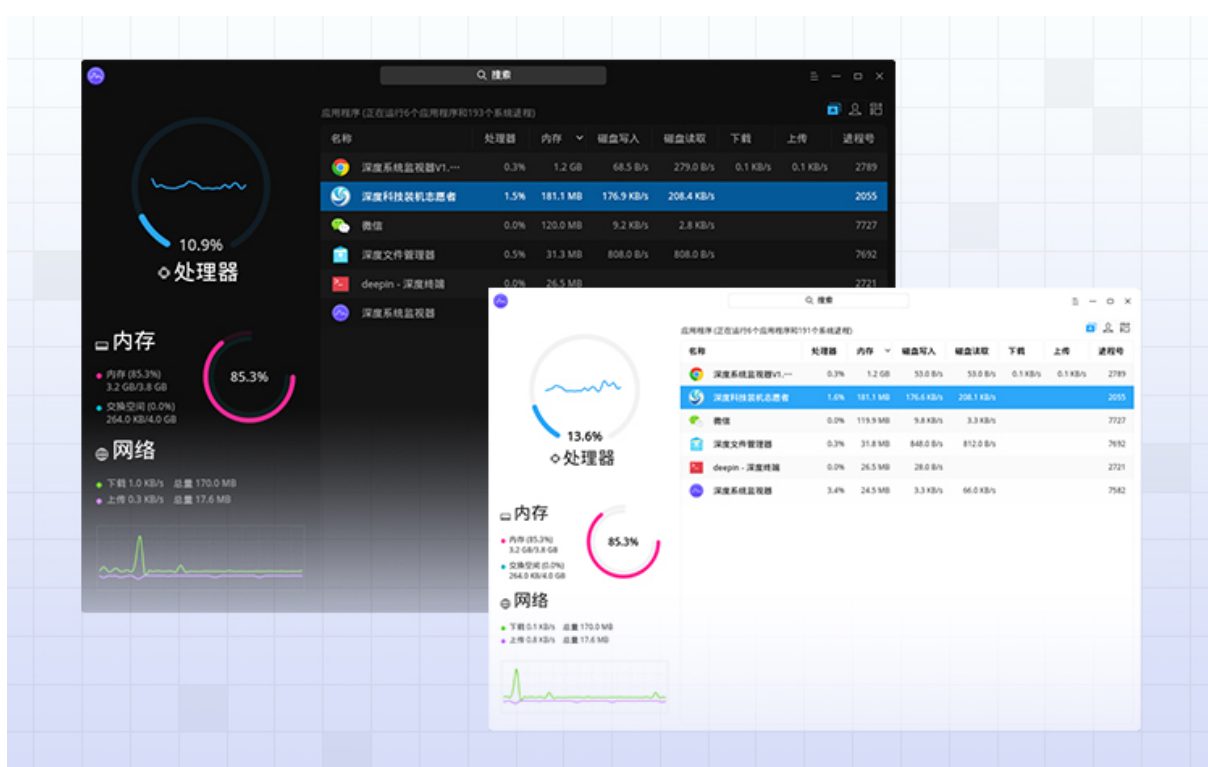


图 3-25 深度系统监视器

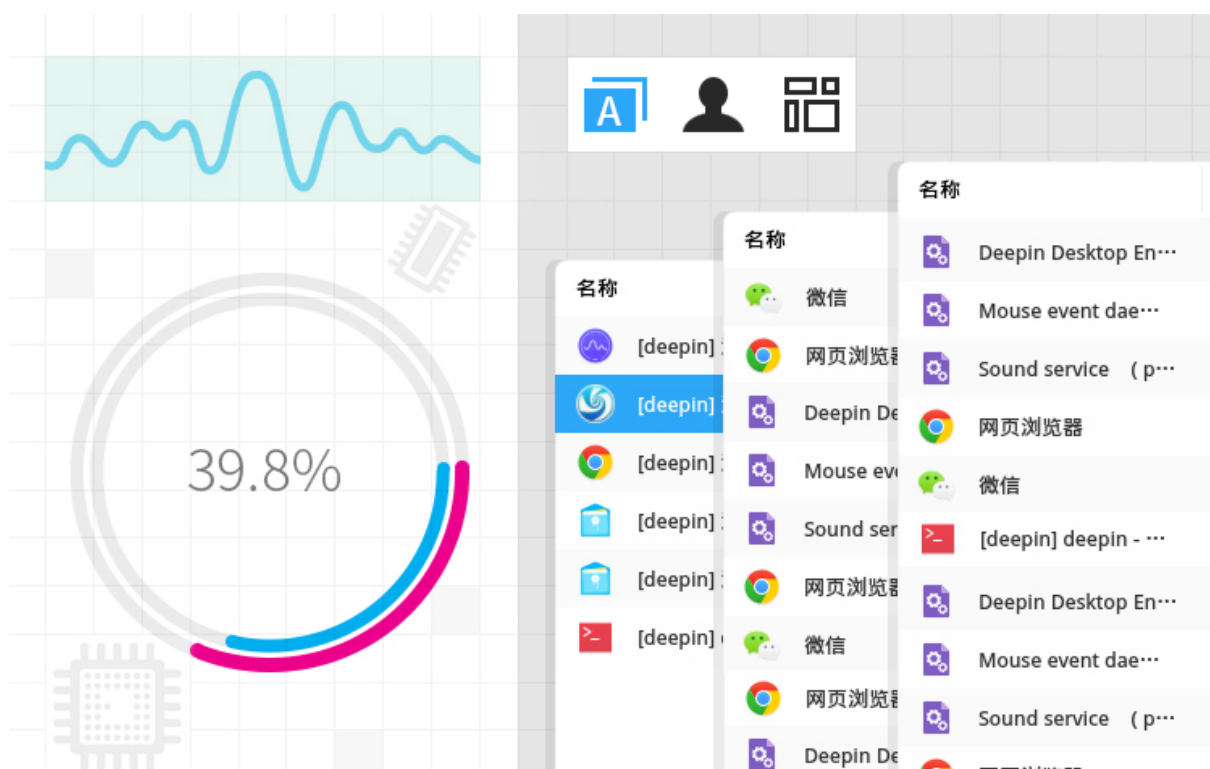


图 3-26 深度系统监视器





图 3-27 深度系统监视器



图 3-28 深度系统监视器



图 3-29 无线投屏

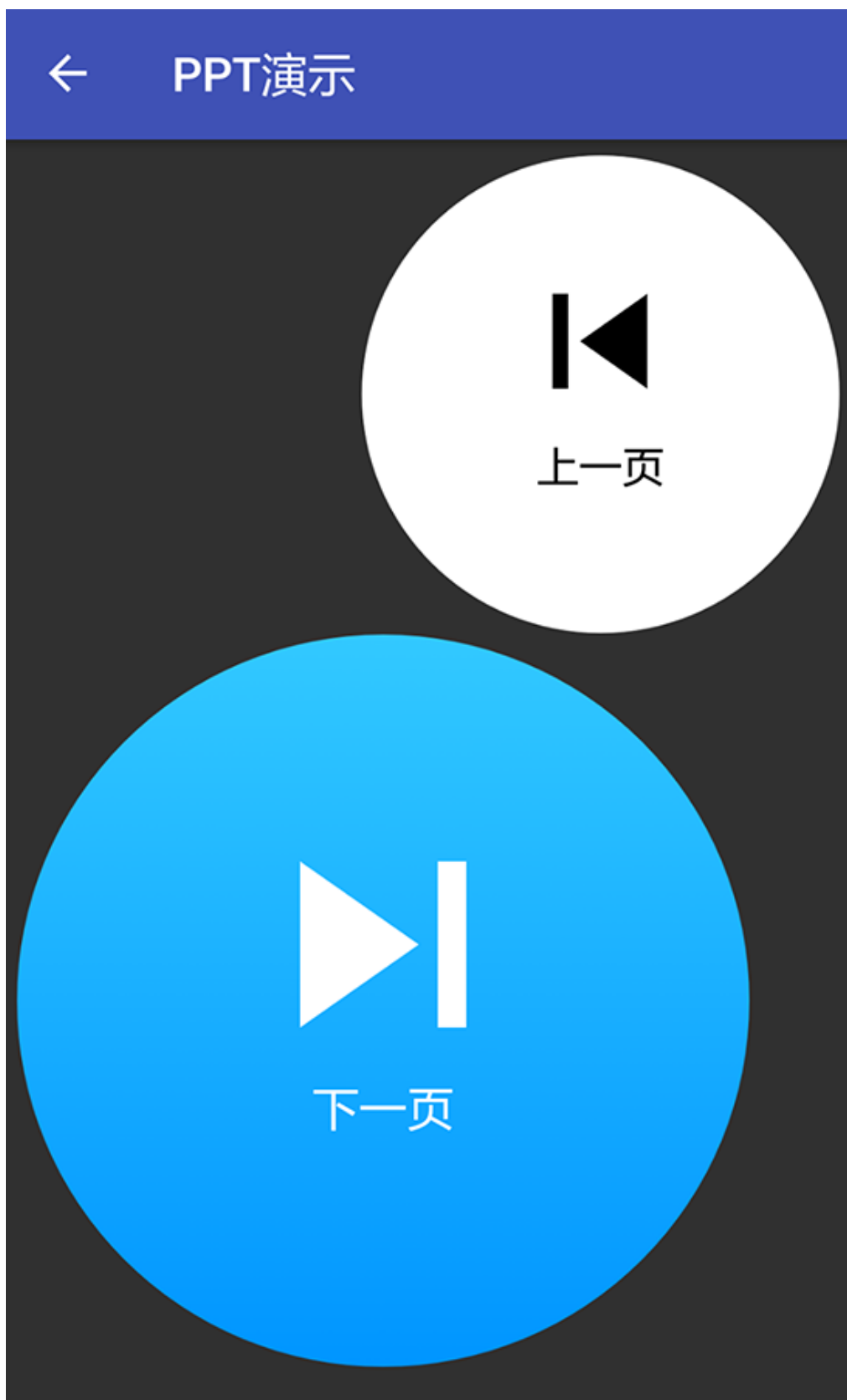


图 3-30 无线投屏

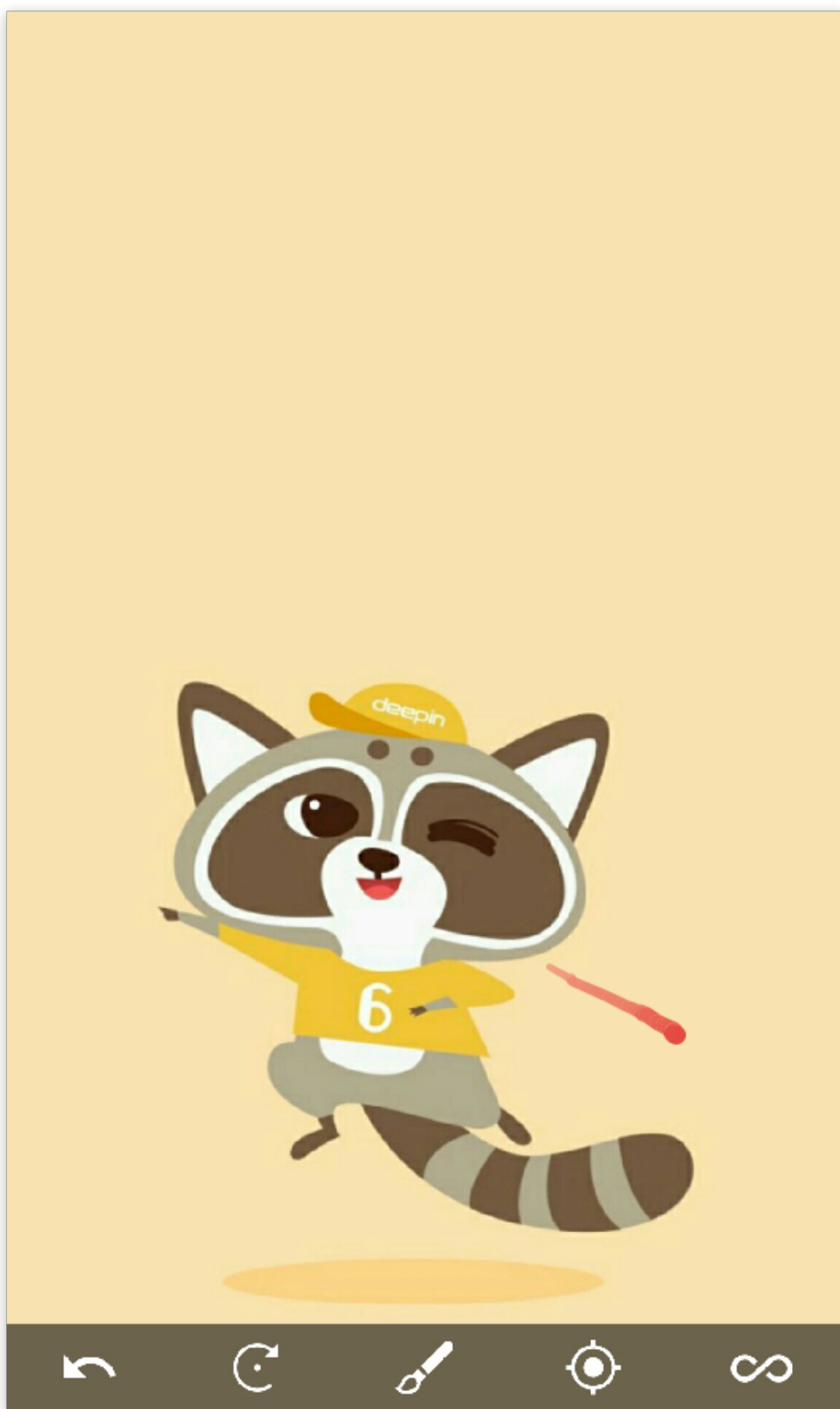


图 3-31 无线投屏



## 第四章 教育学习用的 Linux

摘自：深度商业官网<sup>1</sup>、深度社区官网<sup>2</sup>。

---

<sup>1</sup><https://www.deepin.com/>

<sup>2</sup><https://www.deepin.org/>





# 第二部分

## 其他知识入门

“春江水暖鸭先知”

—— 苏轼

这里需要写点啥的。

第五章 shell 用法简介	39
第六章 深度系统安装	43
第七章 软件安装	45
第八章 管理账户	49



## 第五章 shell 用法简介

### 5.1 引言

作为普通人，使用电脑除了窗口图形界面外，还有个叫终端的字符界面。Linux 中通过在终端敲击字符跟电脑交互，这玩意就叫做 shell，并且相比图形界面 (Graphic User Interfaces, GUI)，通常 shell 功能更加强大。

深度系统默认采用的 shell 是 bash (Bourne Again Shell)，继承兼容于 UNIX 早期的 Bourne shell (作者: Stephen Bourne, 命令名为 sh)。曾经有一段时间深度默认 shell 是 zsh，经群众投票又改回 bash 了，其实那次投票是我发起的。

除此之外，类 UNIX 系统还有其他的一些 shell，比如 ksh, csh, tcsh，有兴趣的可以查相关资料，这里不说了。

### 5.2 用 shell 有啥好处？

为了装吗？不见得。

刚接触 Linux 桌面发行版的时候，按着教材说的，就在学习终端哦。遇到糟糕的 Linux 图形界面卡住了、死机啦，不得不网上搜索解决方案，能不用终端 shell 吗？作为小白，遇到各种各样的问题，查询询问的结果，大神的建议，也多半是在 shell 下解决啊，这叫不得不用。

后来慢慢脱离低级小白，开始喜欢上终端远程连接其他 Linux 电脑，享受这种快捷方便的操作。再后来终端下用 vim 写代码，键盘翻飞，似乎成为了一种享受。

再后来，开始接管一些服务器，开始学习计算机安全知识，似乎更离不开了 shell 了。似乎儿时那种黑客的感觉，只有通过 shell 才可以实现哦。

其实 Windows 下也有 DOS 和 CMD 这类字符操作界面，不过功能弱爆了，powershell 可能功能强大一些，有兴趣的可以查相关资料，这里不说了。

### 5.3 深度操作系统下 shell 简介

按照《Linux Bible》的说法，有三种方式打开 shell 操作界面，分别叫 shell 提示符 (shell prompt)，终端窗口 (Terminal window)，和虚拟控制台 (virtual console)。我感觉深度操作系统就后面两种，第一种说是登录的时候出现的那种字符界面，不太容易遇到，暂且跳过。

深度操作系统登录后，进入桌面，按下 `Ctrl + Alt + T` 或者 `F4` 会弹出深度终端窗口，这个软件在第三章已经作了介绍。可用于 `shell` 练习。对于虚拟控制台，可以通过按下 `Ctrl + Alt + Fi` 其中 `Fi` 表示 `F1, F2, ..., F6`。分别弹出六个虚拟控制台，详情见后面叙述。

下面以深度终端窗口为例来说明。

### 打开深度终端的方式

- 快捷键的方式：按下 `Ctrl + Alt + T` 或者 `F4`，弹出深度终端。
- 图形方式：按下 `super` 键，或者点击左下角的图标，会弹出已安装软件图标，找到深度终端的图标，点击即可。或者在最上面搜索框中搜索终端。
- 鼠标右键：在桌面上，右击选择在终端中打开，也可以进入深度终端。

### 深度终端的显示说明

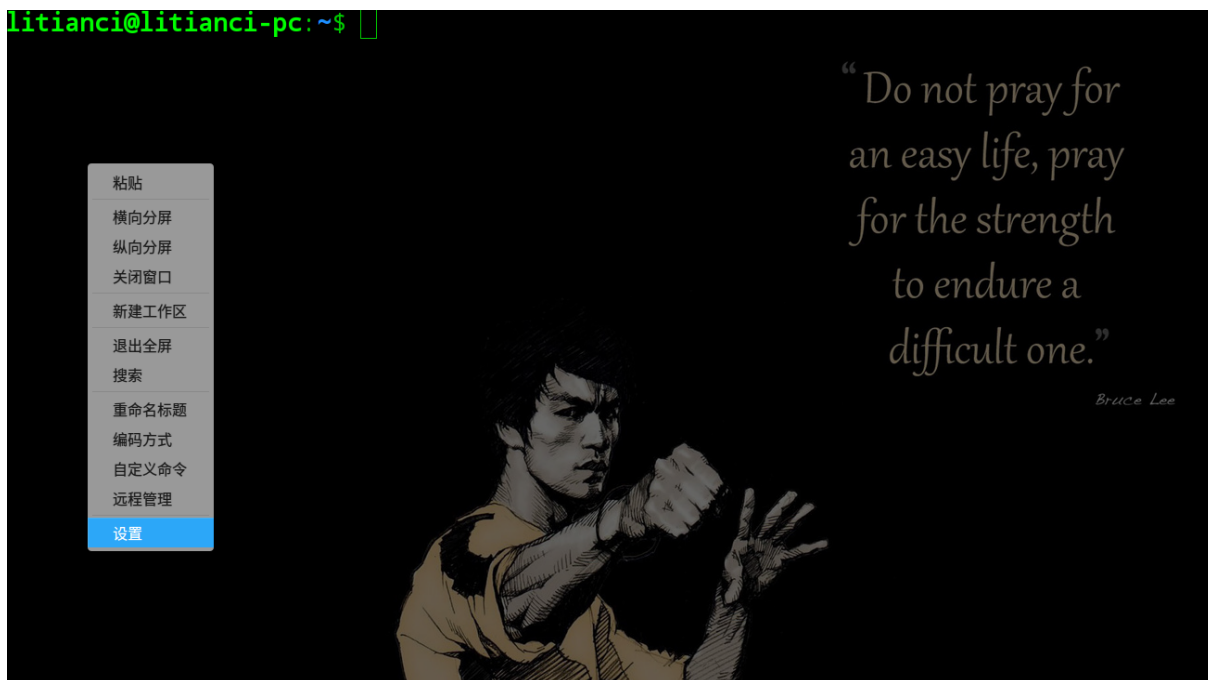
在每行命令行

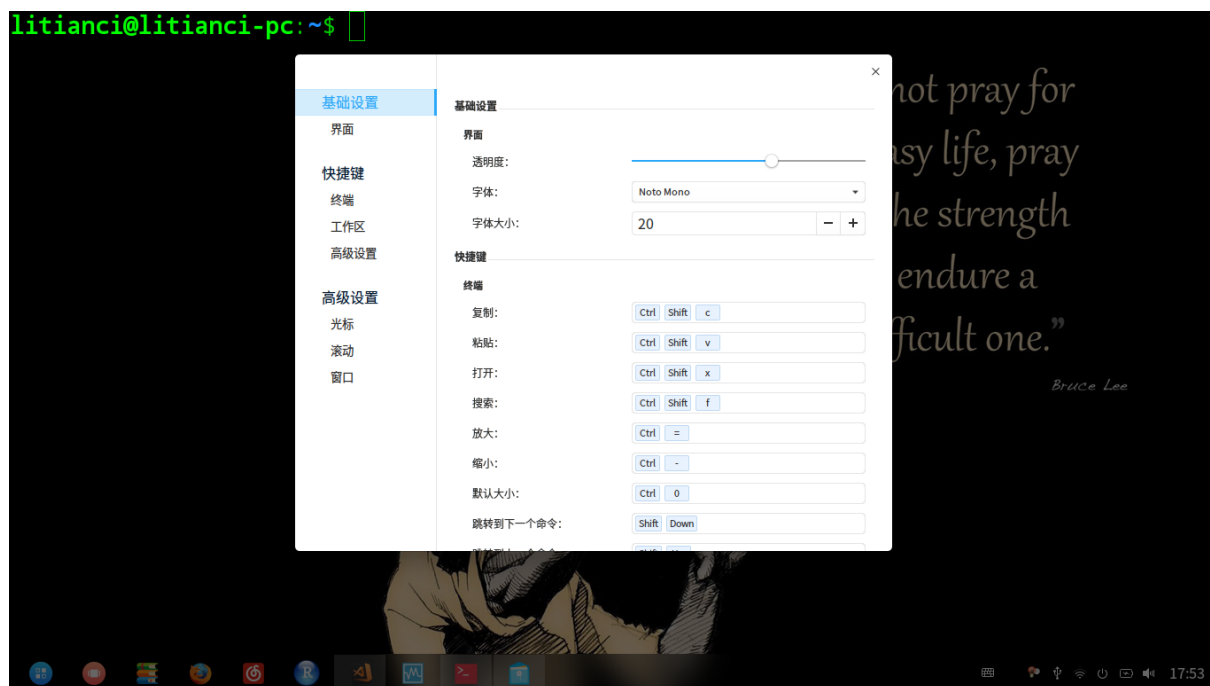
```
l | litianci@litianci-pc:~$
```

`$` 一般表示普通用户，`##` 一般表示 `root` 用户。

### 深度终端设置

按下 `Ctrl + Alt + T` 打开深度终端。如下图所示。







## 第六章 深度系统安装

本章讲介绍如何从优盘、光盘和硬盘安装深度操作系统，以及 LiveCD 模式简介。

参考网页：

- <https://bbs.deepin.org/forum.php?mod=viewthread&tid=135051>
- <https://wiki.deepin.org/index.php?title=%E5%8E%9F%E7%94%9F%E5%AE%89%E8%A3%85>

对以上作者表示感谢。

### 6.1 UEFI 和 legacy BIOS 区别和联系

参考网页：

- <https://baike.baidu.com/item/UEFI/3556240?fr=aladdin>

有些电脑过于老旧，不支持 UEFI 模式，或者没能设置启动（boot）为 UEFI 模式，导致安装失败。

### 6.2 优盘安装

一般电脑都是可以安装的，当然也存在安装失败的情况，对于遇到困难的用户，可以百度或者谷歌怎么解决。这里只叙述需要注意的问题。现在的电脑一般都带 USB 接口，支持优盘安装的。USB 接口，现有 1.0,2.0,3.0 等各种版本，更多了解可以参考百度百科<sup>1</sup>。在采用优盘安装的时候，需要事先检查你电脑的 USB 接口是否工作正常，之前帮一同事安装系统，折腾了一上午才发现 USB 接口是坏的。更有甚者，部分电脑的 USB 3.0 的接口，可能不支持优盘安装，这个都是需要特别注意的。

接下来是制作优盘。你可以采用 windows 或者已经安装的深度操作系统制作优盘，当然对优盘大小等还是有限制的，一般大于 8G 为宜。

#### 6.2.1 第一步下载并校验 deepin.iso 文件

本节转载自深度维基百科<sup>2</sup>，有改动。对前人的辛勤工作，表示感谢。

#### 官方镜像

<sup>1</sup><https://baike.baidu.com/item/USB%E6%8E%A5%E5%8F%A3/493294?fr=aladdin>

<sup>2</sup><https://wiki.deepin.org/index.php?title=%E5%8E%9F%E7%94%9F%E5%AE%89%E8%A3%85>

访问 deepin 社区下载页面<sup>1</sup>，下载深度操作系统系统最新版本的镜像文件（以便您能够体验到最新特性）。（据我本人的经验，如果你没有百度 VIP 账号，不建议从百度网盘下载。国内从官网或者 sourceforge<sup>2</sup>相对快一点，也不是绝对的，可以根据实际情况选用不同的下载点）注意：为了更加专注系统的发展，deepin 15.4 及后续版本将不再提供 32 位官方 iso 镜像，如需获取和技术支持，[deepin@deepin.org](mailto:deepin@deepin.org)<sup>3</sup>。

### MD5 校验

下载深度操作系统镜像完成后，需要对其进行校验，非官方或不完整的镜像将不能用于深度操作系统的安装：

- Windows 系统：下载 Hash 软件，校验您下载的镜像的 MD5 值与下载页面提供的 MD5 值是否一致。（MD5 值在立即下载按钮下方）
- Linux 系统：在对应的镜像文件下，打开深度终端，执行 `md5sum deepin-xxx.iso` 命令，请确认下载的镜像的 MD5 值与下载页面提供的 MD5 值是否一致。（MD5 值在立即下载按钮下方）。说明：deepin-xxx.iso 即为下载的系统镜像文件名，可使用 Tab 键自动补全文件名。

## 6.3 安装遇到的问题

参考网页：

- <https://bbs.deepin.org/forum.php?mod=viewthread&tid=146222>
- <https://bbs.deepin.org/forum.php?mod=viewthread&tid=146224>

基本全文转载，有改动。已获许可<sup>4</sup>。

---

<sup>1</sup><https://www.deepin.org/download/>

<sup>2</sup><https://sourceforge.net/projects/deepin/>

<sup>3</sup><mailto:deepin@deepin.org>

<sup>4</sup><https://bbs.deepin.org/forum.php?mod=viewthread&tid=146224&page=1#pid384937>



## 第七章 软件安装

这里会介绍如下几种安装方式，

- 直接双击安装
- apt 安装
- dpkg 安装
- .run 类型的文件安装
- 其他软件安装，比如 you-get 的安装
- 字体的安装
- 源码安装
- 输入法的安装

### 7.1 npm 软件的安装

这里以 you-get 软件为例，介绍 npm 软件的安装方式。

### 7.2 veil 软件安装

其他发行版可以直接用 apt-get 直接安装的，比如 veil，

```
1 $ sudo apt-get install veil
```

在 Kali Linux 就可以直接运行，但是在深度操作系统下就无法下载，这个时候不妨网上搜索一下，加入私有源。当然最好的是搜到这个软件的官网，比如 veil 的官网：<https://github.com/Veil-Framework/Veil>。

```
1 $ sudo apt-get -y install git
2 $ git clone https://github.com/Veil-Framework/Veil.git
3 $ cd Veil/
4 $ cd setup
5 $ sudo ./setup.sh -c
```

这样一路点击确认下来，慢慢等待，就可以成功安装 veil 了，当然还需要对 `setup.sh` 文件检测操作系统的代码改一改，让他支持 deepin。详情见 [pull request<sup>1</sup>](https://github.com/Veil-Framework/Veil/pull/130)。用方法有所变

<sup>1</sup><https://github.com/Veil-Framework/Veil/pull/130>

样。

也可以到 [debian 软件包库<sup>1</sup>](#) 去搜索，相关的软件包，选择合适的版本安装。不过 `veil` 没有对应的 `deb` 包。这是一个比较庞大的软件，估计也不会有一个小的包。

## 7.3 dpkg 安装

## 7.4 其他安装方式

很多软件，由于依赖或者其他原因，安装起来特别费劲，对于初学者，不啻于一场灾难。有好事者，就提供了各种一键安装包。下面介绍两个，限于本人水平，也就只知道这两，有同仁知道更多的谢谢发表评论。

### 7.4.1 bitnami

官网：<https://bitnami.com/stacks>

提供一键安装软件包，或者虚拟机级别文件等。有需要的可以去查询，肯定有你所需要的。比如 `gitlab`，`OwnCloud` 等等

### 7.4.2 turnkeylinux

官网：<https://www.turnkeylinux.org/>

对于常见的 `gitlab`, `lampp` 等提供虚拟机安装包，可以尽情享受快捷。

另外，我常用的 `xampp` 一键安装包，却是从其他网站上下载的。如果你安装一个软件特别费劲，（不包含无法破解软件的哦）不妨试试搜索一下“xxx 一键安装包”，看看有没有其他人已经帮你解决这个问题了。

## 7.5 you-get 软件的安装

参考网页：

- <https://github.com/soimort/you-get>
- 

根据 `you-get` 源码网站<sup>2</sup>的提示，在 `deepin` 下，用如下命令安装，较为合适。

```
1 $ sudo pip3 install you-get
```

<sup>1</sup><http://www.debian.org/distrib/packages>

<sup>2</sup><https://github.com/soimort/you-get>

但是，你可能会发现，自己的电脑没有安装 pip3 软件。试着 `sudo apt-get install pip3` 也不存在这个软件包啊。接着百度 how to install pip3，找到一大堆页码。参考这个<sup>1</sup>，如下命令安装。

```
1 $ sudo apt-get install python3-pip python3-dev build-essential
```

这样就安装好了 pip3，我的理解，pip3 就是 python3-pip 的意思，不知此解对不对。那么接下来安装 you-get 就容易多啦。

```
1 $ sudo pip3 install you-get
```

如果需要更新 you-get，可以如下。

```
1 $ sudo pip3 install --upgrade you-get
```

对于 you-get 如何使用，这里就不赘述啦。详见官网说明<sup>2</sup>。

## 7.6 字体的安装

有些书籍的制作需要特殊的字体，在 Windows、Mac 下安装字体比较简单，在 Deepin 下安装字体同样很简单。参考这里<sup>3</sup>，

下载需要的字体，下面以安装交大论文需要的字体为例，网上下好需要的字体<sup>4</sup>，解压缩后，双击安装即可，也可以仿照上面网页<sup>5</sup>提供的方式安装。

```
1 $ mkdir ~/.fonts
2 $ cp *.ttf ~/.fonts # 当前用户可用新字体
3 $ sudo cp *.ttf /usr/share/fonts/local/ # 所有用户可以使用新字体
4 $ sudo fc-cache -f
```

**解释** - `cp *.ttf ~/.fonts` 的 `*.ttf` 表示你下载的那些字体，需要你修改为对应字体的路径。

<sup>1</sup><http://ask.xmodulo.com/install-pip-linux.html>

<sup>2</sup><https://github.com/soimort/you-get>

<sup>3</sup>[https://github.com/weijianwen/SJTUThesis/blob/master/tex/app\\_setup.tex](https://github.com/weijianwen/SJTUThesis/blob/master/tex/app_setup.tex)

<sup>4</sup><http://download.csdn.net/download/bubifengyun/9695994>

<sup>5</sup>[https://github.com/weijianwen/SJTUThesis/blob/master/tex/app\\_setup.tex](https://github.com/weijianwen/SJTUThesis/blob/master/tex/app_setup.tex)

## 7.7 输入法的安装

这里以搜狗输入法为例，讲解怎么在没有安装搜狗输入法的 Linux 桌面发行版上安装搜狗输入法，其实深度操作系统自带搜狗输入法的。当然第一步百度搜索了，搜索到搜狗输入法的 Linux 专用版网站<sup>1</sup>，上面网址如果不对，请继续搜索，毕竟某搜索网址广告比较厉害，不一定就第一时间提供有用的答案。进入官网，下载对应 \*.deb 文件，使用命令

```
1 $ sudo dpkg -i /path/to/***.deb
```

就可以安装了。本来这个输入法安装应该放在上面的，但是考虑到很多人找不到怎么安装，特来说一下。深度操作系统下，可以直接双击下载的 \*.deb 安装包的，一会就会自动安装完。不过汉语简体版的深度操作系统已经自带了该输入法。

---

<sup>1</sup><http://pinyin.sogou.com/linux/>

## 第八章 管理账户

### 8.1 引言

本章会涉及到

- 账户 (user account) 管理
- 账户组管理
- 其他方式的账户管理

深度操作系统安装过程中，会引导你设置一个普通账户，非根账户，也就是没有最高权限的普通账户。当有家人、朋友或者同事要使用我们的电脑的时候，可能要涉及到账户的添加和权限管理。这就是本章要讨论的内容。为了更方便的管理多个具有相同权限的账户，还会涉及到账户组的设置等。接下来会讲到怎么使用 `useradd`, `usermod` 等命令，配置自家目录、默认的 `shell`、分组、以及账户和组编号等。

### 8.2 创建账户

为了更好的分配资源权限，保护信息安全，为某些使用者创建新的账户还是很有必要的。下面介绍，在深度操作系统下，两种创建账户的方式：图像方式 (GUI)，命令方式 (Shell)。

#### 8.2.1 图形方式

如下图所示，创建账户是非常简单的，打开设置侧栏，找到账户相关的那一部分，点击创建账户，进入到下图，填上用户名和密码等需要的数据，就可以了。



如果需要更改头像、设置全名、更改密码、设置是否自动登录免密码登录，可以选中需要修改的账户，点击进去就可以啦。

图形方式默认创建的账户是普通账户，比如是 `phptester` 账户，在 `/etc/passwd/etc/group`



图 8-1 图形方式创建账户

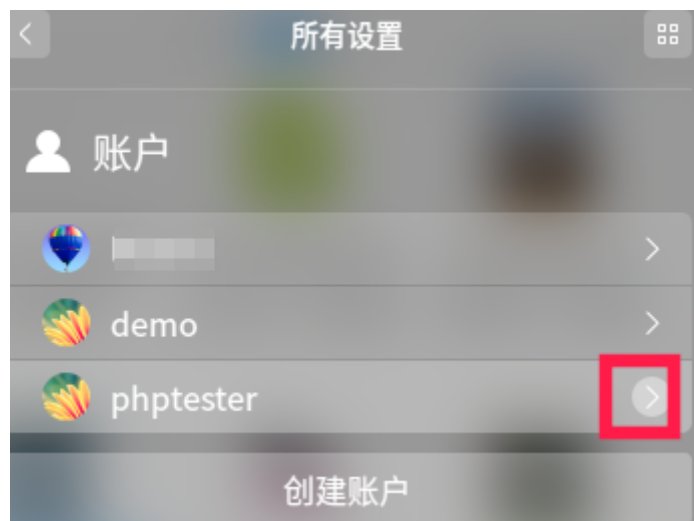


图 8-2 图形方式配置账户



图 8-3 图形方式配置账户

会出现 phptester 相关的信息。有兴趣的读者，可以先去看看上述俩文件，后面还会对这两文件作进一步的讲解。

### 8.2.2 命令方式



# 第三部分

## 安全

“这只是万里长征走完了第一步，以后的路程更长，工作更伟大，更艰苦。务必使同志们继续地保持谦虚、谨慎、不骄不躁的作风，务必使同志们继续地保持艰苦奋斗的作风。”

—— 毛泽东

这里需要写点啥的。



# 第四部分

## 附录及其他

“这只是万里长征走完了第一步，以后的路程更长，工作更伟大，更艰苦。务必使同志们继续地保持谦虚、谨慎、不骄不躁的作风，务必使同志们继续地保持艰苦奋斗的作风。”

—— 毛泽东

附录甲 愚公移山	57
附录乙 如何制作本书	59
附录丙 各章格式说明	67
附录丁 兼容 L <sup>A</sup> T <sub>E</sub> X 排版	71
附录戊 RMarkdown/Bookdown 排版 示例	85
附录己 常见问题	119
附录庚 大事记	121
索引	123
参考文献	125

这里需要写点啥的。



## 附录甲 愚公移山

——先秦 列御寇

太行、王屋二山，方七百里，高万仞，本在冀州之南，河阳之北。

北山愚公者，年且九十，面山而居。惩山北之塞，出入之迂也，聚室而谋曰：“吾与汝毕力平险，指通豫南，达于汉阴，可乎？”杂然相许。其妻献疑曰：“以君之力，曾不能损魁父之丘，如太行、王屋何？且焉置土石？”杂曰：“投诸渤海之尾，隐土之北。”遂率子孙荷担者三夫，叩石垦壤，箕畚运于渤海之尾。邻人京城氏之孀妻有遗男，始龀，跳往助之。寒暑易节，始一反焉。

河曲智叟笑而止之曰：“甚矣，汝之不惠！以残年余力，曾不能毁山之一毛，其如土石何？”北山愚公长息曰：“汝心之固，固不可彻，曾不若孀妻弱子。虽我之死，有子存焉；子又生孙，孙又生子；子又有子，子又有孙；子子孙孙无穷匮也，而山不加增，何苦而不平？”河曲智叟亡以应。

操蛇之神闻之，惧其不已也，告之于帝。帝感其诚，命夸娥氏二子负二山，一厓朔东，一厓雍南。自此，冀之南，汉之阴，无陇断焉。



## 附录乙 如何制作本书

### 乙.1 准备工作

本模板使用 bookdown 实现了由 R Markdown (Bookdown) -> L<sup>A</sup>T<sub>E</sub>X-> PDF 的一系列转换，其中用到的依赖有：

- bookdown
- pandoc
- Tex
- zhmakeindex (尚未包含到 texlive)

需要安装这些软件，才可以使用。

#### 乙.1.1 Linux 下使用

下面以深度操作系统 15.5 版本为例说明。安装过程中，请确保当前 rstudio 的版本高于 1.0.0, texlive 为 2015 年及之后的版本，pandoc 的版本高于 2.0.0。

```
$ sudo apt-get install r-base r-base-dev \  
rstudio texlive-full pandoc make
```

如果不能满足，请参考相关软件官网，下载最新版本。

下面安装 zhmakeindex 用于产生中文索引，如果不需要中文索引，跳过下面这些操作，并注释掉 *.latex/template.tex* 中 zhmakeindex 相关的语句

```
1 \usepackage[texindy]{imakeidx}  
2 \makeindex[title=索引,columns=2,intoc=true,columnseprule=true]  
3 \indexsetup{  
4   othercode={%  
5     \thispagestyle{main}%  
6   }  
7 }  
8 \printindex
```

以及删除 *.index.Rmd* 中 zhmakeindex 相关的语句

```
1 , tinytex.makeindex = 'zhmakeindex', tinytex.makeindex.args = c('-s',  
  'imakeidx.ist')
```

下面介绍怎么安装 `zhmakeindex`，到这里<sup>1</sup>或者直接点击下载<sup>2</sup>，下载可执行文件。解压缩后，有如下文件夹。根据需要放到系统能够找到的地方。比如 `$ sudo cp ./bin/linux_x64/zhmakeindex /usr/local/bin/`



为了方便大家简单操作，本文自带了 `zhmakeindex` 软件。其中索引有个坑，见这里<sup>3</sup>，也就是对于文章标题索引时不可以放在标题末尾。

#### 乙.1.1.1 texlive 2018 的安装

下载 `texlive2018.iso` 文件<sup>4</sup>，到某个文件夹 `~/Downloads/texlive2018.iso`，创建临时文件夹 `texlive`，用于挂载下载的文件。

```
$ mkdir ~/texlive
$ sudo mount ~/Downloads/texlive2018.iso ~/texlive
```

进入 `texlive` 文件夹，开始安装，选项如下。

```
$ cd ~/texlive
$ sudo ./install-tl
```

会弹出如下信息，



<sup>1</sup><https://ctan.org/pkg/zhmakeindex>

<sup>2</sup><http://mirrors.ctan.org/indexing/zhmakeindex.zip>

<sup>3</sup><https://github.com/jgm/pandoc/issues/4762>

<sup>4</sup><https://mirrors.sjtug.sjtu.edu.cn/ctan/systems/texlive/Images/>



```

<O> options:
  [ ] use letter size instead of A4 by default
  [X] allow execution of restricted list of programs via \write18
  [X] create all format files
  [X] install macro/font doc tree
  [X] install macro/font source tree
  [ ] create symlinks to standard directories
  [X] after install, use tlnet on CTAN for package updates

<V> set up for portable installation

Actions:
<I> start installation to hard disk
<P> save installation profile to 'texlive.profile' and exit
<H> help
<Q> quit

Enter command: O

```

出现如下情况

```

=====

Options customization:

<P> use letter size instead of A4 by default: [ ]
<E> execution of restricted list of programs: [X]
<F> create all format files: [X]
<D> install font/macro doc tree: [X]
<S> install font/macro source tree: [X]
<L> create symlinks in standard directories: [ ]
      binaries to:
      manpages to:
      info to:
<Y> after installation, get package updates from CTAN: [X]

Actions: (disk space required: 5381 MB)
<R> return to main menu
<Q> quit

Enter command: L
New value for binary directory [/usr/local/bin]: 敲Enter键
New value for man directory   [/usr/local/man]: 敲Enter键
New value for info directory  [/usr/local/info]: 敲Enter键

```

类似如此结果，

```
上面略去一部分文件
<L> create symlinks in standard directories: [X]
      binaries to: /usr/local/bin
      manpages to: /usr/local/man
      info to: /usr/local/info
<Y> after installation, get package updates from CTAN: [X]

Actions: (disk space required: 5381 MB)
<R> return to main menu
<Q> quit

Enter command: R
```

接着回到主界面

```
Actions:
<I> start installation to hard disk
<P> save installation profile to 'texlive.profile' and exit
<H> help
<Q> quit

Enter command: I
```

输入 I 开始安装。成功后，弹出类似下文的界面

```
Welcome to TeX Live!

See /usr/local/texlive/2018/index.html for links to documentation.
The TeX Live web site (http://tug.org/texlive/) contains any updates
and
corrections. TeX Live is a joint project of the TeX user groups around
the
world; please consider supporting it by joining the group best for you
. The
list of groups is available on the web at http://tug.org/usergroups.html.
```

```
Add /usr/local/texlive/2018/texmf-dist/doc/man to MANPATH.  
Add /usr/local/texlive/2018/texmf-dist/doc/info to INFOPATH.  
Most importantly, add /usr/local/texlive/2018/bin/x86_64-linux  
to your PATH for current and future sessions.  
  
Logfile: /usr/local/texlive/2018/install-tl.log
```

### 乙.1.2 Windows 下使用

略

### 乙.1.3 苹果操作系统下使用

略

## 乙.2 模板选项

sjtuthesis-rmd 提供了一些常用选项，可以组合使用，在 *.index.Rmd* 文件中配置。具体配置方法，见后文。这些选项包括：

- 学位类型：bachelor(学位)、master(硕士)、doctor(博士)，是必选项。
- 中文字体：fandol(Fandol 开源字体)、windows(Windows 系统下的中文字体)、mac(macOS 系统下的华文字体)、ubuntu(Ubuntu 系统下的文泉驿和文鼎字体)、adobe(Adobe 公司的中文字体)、founder(方正公司的中文字体)，默认根据操作系统自动配置。
- 英文模版：使用 english 选项启用英文模版。
- 盲审选项：使用 review 选项后，论文作者、学号、导师姓名、致谢、发表论文和参与项目将被隐去。

## 乙.3 编译模板

模板默认使用 GNU Make 构建，后续如无特殊说明，默认执行命令的文件夹以及当前文件夹均为本模板的根目录。

```
$ make
```

在已经安装 RStudio 的前提下，也可手动编译。打开 RStudio 软件，在弹出的界面，左上角点击 File->Open Project，选中 *.SJTUThesis-Rmd.Rproj* 文件打开，点击右上角靠

下的一栏，有个 Build 格，会出现 Build Book 等按钮。点击 Build Book 就会生成 PDF 书籍了，位于 `./_book/sjtu-thesis-rmd.pdf`。

如果需要查看字数，先生成文件，然后执行，

```
$ make wordcount
```

## 乙.4 文件布局

使用 `tree` 命令所得。

代码 乙-1 模板文件布局

```
dd
```

## 乙.5 主要文件介绍

### 乙.5.1 L<sup>A</sup>T<sub>E</sub>X 模板文件

格式控制文件控制着论文的表现形式，包括 `./sjtuthesis.cfg` 和 `./sjtuthesis.cls`。其中，“cls”控制论文主体格式，“cfg”为配置文件。上述文件全部来自上海交通大学学位论文 L<sup>A</sup>T<sub>E</sub>X 模板<sup>1</sup>

### 乙.5.2 各章源文件

主要位于 `./rmd/` 文件夹。由于现阶段部分功能实现不全，暂且把部分内容以 L<sup>A</sup>T<sub>E</sub>X 形式存放在 `./tex/` 文件夹下。

- 中英文摘要 (`./rmd/000-abstract.Rmd`)，符号表 (`./rmd/001-symbol.Rmd`) 等。前言 (frontmatter) 的其他部分，中英文封面、原创性声明、授权信息在 `sjtuthesis.cls` 中定义，不单独分离为 Rmd 文件。
- 目录文件 (`./tex/content.tex`)，设置目录内容。如果不需要图片、算法等的目录，可以修改该文件。
- 模板文件 (`./tex/template.tex`)，统筹最终生成的 `_book/sjtu-thesis-rmd.tex` 内容。
- 正文 (mainmatter)，是文章的主体，位于 `./rmd/` 文件夹下，基本用 Rmarkdown 实现，部分不容易实现的，照搬 L<sup>A</sup>T<sub>E</sub>X 原文，均在 `xxx.Rmd` 文件内。

<sup>1</sup><https://github.com/sjtug/SJTUThesis>

- 附录 (*./rmd/xxx-appendix-name.Rmd*), 类似正文, 使用 Rmarkdown 实现, 部分内容用  $\LaTeX$  实现, 均在 *xxx.Rmd* 文件内。
- 致谢 (*./tex/ack.tex*)、攻读学位论文期间发表的学术论文目录 (*./tex/pub.tex*)、个人简历 (*./tex/resume.tex*) 组成附录后的部分 (backmatter)。这部分能力所限, 未能实现从  $\LaTeX$  到 Rmarkdown 的转换, 仍旧保留为  $\LaTeX$  格式。参考文献列表由 bibtex 插入, 不作为一个单独的文件。

### 乙.5.3 配置文件

主要为 *./index.Rmd* 文件, 另外两个配置文件 (*./\_bookdown.yml*、*./\_output.yml*) 没有特殊需求不需要更改。

对于 *./index.Rmd* 文件, 论文题目、作者等信息直接更改为自己的即可。对第乙.2节提到的论文选项, 比如学位类型、英文模板、盲审选项、最终提交等可以放在 `classoption` 的 `[]` 内。比如 `classoption: [doctor, openright, twoside, submit]`, 就表示博士论文的提交版本。

### 乙.5.4 图片文件夹 figure

`figure` 文件夹放置了需要插入文档中的图片文件 (支持 PNG/JPG/PDF/EPS 格式的图片), 可以在按照章节划分子目录。模板文件中使用 `\graphicspath` 命令定义了图片存储的顶层目录, 在插入图片时, 顶层目录名 “figure” 可省略。

### 乙.5.5 参考文献数据库 bib

目前参考文件数据库目录只存放一个参考文件数据库 *./bib/thesis.bib*。关于参考文献引用, 可参考第丁章中的例子。

### 乙.5.6 辅助文件

- *./SJTUThesis-Rmd.Rproj* 项目文件, 方便使用 `rstudio` 打开, 如果仅使用 `make`, 可以不要。
- *./\_render.R* 渲染文件, 复制自谢益辉 Bookdown 中文模板<sup>1</sup>, 致谢。
- *./Makefile* `make` 的文件, 综合上海交通大学学位论文  $\LaTeX$  模板<sup>2</sup>和谢益辉 Bookdown 中文模板<sup>3</sup>。

<sup>1</sup><https://github.com/yihui/bookdown-chinese>

<sup>2</sup><https://github.com/sjtug/SJTUThesis>

<sup>3</sup><https://github.com/yihui/bookdown-chinese>

- `./handed_pdf/` 复制自上海交通大学学位论文 L<sup>A</sup>T<sub>E</sub>X 模板<sup>1</sup>手写授权。
- `./_book/` 最终生成文件所在文件夹。
- `./_bookdown_files/` 过程中生成的文件所在文件夹。

## 乙.6 捐赠

如果您也想感谢这份模板的话，可以捐款支持本人工作。如果钱少，就刷个支付宝红包二维码在消费的时候支持一下；如果您想直接捐款，可以直接扫支付宝里的付款，多少随意。谢谢。



图 乙-1 支付宝红包及收款的二维码

Figure 乙-1 Zhi Fu Bao Pay

<sup>1</sup><https://github.com/sjtug/SJTUThesis>

## 附录丙 各章格式说明

Bookdown 编译各 Rmd 文件时，是按照文件名的字母排序拼接在一起的，故而我们不需要专门写一个文件，类似 L<sup>A</sup>T<sub>E</sub>X 模板的 `thesis.tex` 文件，但是也要求文件名需要特殊处理才可以做到有序。本模板的文件名采用如下命名方式，`xxx-name.Rmd`。其中 `xxx` 表示从 000 到 999，用于排序。`name` 是对应章节的名字。一章一个文件。

由于自身水平所限，部分格式实现起来费劲，就采用直接书写 L<sup>A</sup>T<sub>E</sub>X 代码的形式实现了。

### 丙.1 摘要

摘要文件需要放在除了扉页的最开头位置。由于扉页等已经在模板中处理生成，故摘要需要放在最开头，命名为 `./rmd/000-abstract.Rmd`。

代码 丙-1 摘要代码框架示例

```

1 \frontmatter
2 \pagestyle{main}
3
4 \abstract
5
6 上海交通大学是我国历史最悠久的高等学府之一。
7
8 \keywords{\large 上海交大 \quad 饮水思源 \quad 爱国荣校}
9
10 \englishabstract
11
12 An imperial edict issued in 1896 by Emperor Guangxu, established
    Nanyang Public School in Shanghai.
13
14 \englishkeywords{\large SJTU, Thesis, Rmarkdown/Bookdown template}
15
16 \include{tex/content}

```

#### 解释

1. 第 1、2 行，表示下面属于论文前言部分，采用**罗马数字对前言编号**，页面格式属于正文格式。
2. 第 4、10 行，表示下面是中英文的摘要部分。

3. 第 16 行，表示接下来为目录部分。
4. 摘要全文属于  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  原文复制，局部改动的方法实现。为了减少文件个数，把文件格式以及目录也放在该文件了，但也造成了文件不专用的问题。请根据爱好，自行斟酌。

## 丙.2 摘要后的部分章节

摘要后的部分章节，属于标准的 Markdown 文件，代码框架示例如代码丙-2所示。

代码 丙-2 前言后文代码框架示例

```

1 # 主要符号对照表
2
3 \begin{longtable}{rl}
4 $\epsilon$      & 介电常数 \\
5 $\mu$         & 磁导率 \\
6 \end{longtable}

```

### 解释

1. 第 1 行，标准 Markdown 语句，表示这是一章。
2. 第 3-6 行，该章的具体内容，本章直接复制的  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  源码，也可用使用 Markdown 书写。

## 丙.3 正文第一章

代码 丙-3 正文第一章代码框架示例

```

1 \mainmatter
2 \pagestyle{main}
3
4 # 这是什么 {#chap:intro}
5
6 这是上海交通大学(非官方)学位论文 Bookdown 模板。

```

### 解释

1. 第 1、2 行，表示下面属于论文正文部分，采用阿拉伯数字对正文编号，页面格式属于正文格式。
2. 第 4 行及以后，属于标准的 Markdown 内容。



## 丙.4 正文后续章节

框架格式类似摘要后的文件，如代码丙-2所示，略。

## 丙.5 全文总结

全文总结比较特殊，没有编入章节序号，采用如下方式。

代码 丙-4 全文总结代码框架示例

```
1 \summary
2
3 这里是全文总结内容。
```

### 解释

1. 第 1 行，表示下面为全文总结部分。
2. 第 3 行及以后，书写全文总结内容，可以使用  $\LaTeX$  或 Markdown 书写。

## 丙.6 附录第一章

代码 丙-5 附录代码框架示例

```
1 \appendix
2
3 # bookdown 入门简介 {#bookdown-intro}
```

### 解释

1. 第 1 行，表示下面属于论文附录部分，采用**英文字母对附录编号**。
2. 第 3 行及以后，属于标准的 Markdown 内容。

## 丙.7 附录后续章节

框架格式类似摘要后的文件，如代码丙-2所示，略。

## 丙.8 后缀部分

相关代码为  $\LaTeX$  形式，位于 `.tex/` 文件夹下，请参考上海交通大学学位论文  $\LaTeX$  模板<sup>1</sup>。

---

<sup>1</sup><https://github.com/sjtug/SJTUThesis>



## 附录丁 兼容 L<sup>A</sup>T<sub>E</sub>X 排版

本章完全照抄交大论文模板的 L<sup>A</sup>T<sub>E</sub>X 排版例子，除了动了标题和多了这行话，运行良好，但是部分地方会多出括号来，请对照原文删除。

### 丁.1 列表环境

#### 丁.1.1 无序列表

以下是一个无序列表的例子，列表的每个条目单独分段。

- 这是一个无序列表。
- 这是一个无序列表。
- 这是一个无序列表。

使用 `itemize*` 环境可以创建行内无序列表。

- 这是一个无序列表。• 这是一个无序列表。• 这是一个无序列表。

行内无序列表条目不单独分段，所有内容直接插入在原文的段落中。

#### 丁.1.2 有序列表

使用环境 `enumerate` 和 `enumerate*` 创建有序列表，使用方法无序列表类似。

1. 这是一个有序列表。
2. 这是一个有序列表。
3. 这是一个有序列表。

使用 `enumerate*` 环境可以创建行内有序列表。

1. 这是一个默认有序列表。2. 这是一个默认有序列表。3. 这是一个默认有序列表。

行内有序列表条目不单独分段，所有内容直接插入在原文的段落中。

#### 丁.1.3 描述型列表

使用环境 `description` 可创建带有主题词的列表，条目语法是 `\item[主题]` 内容。

**主题一** 详细内容

**主题二** 详细内容

**主题三** 详细内容 ...

### 丁.1.4 自定义列表样式

可以使用 `label` 参数控制列表的样式，详细可以参考 WikiBooks<sup>1</sup>。比如一个自定义样式的行内有序列表

a) 这是一个自定义样式有序列表。b) 这是一个自定义样式有序列表。c) 这是一个自定义样式有序列表。

## 丁.2 数学排版

### 丁.2.1 公式排版

这里有举一个长公式排版的例子，来自《Math mode》<sup>2</sup>：

$$\frac{1}{2}\Delta(f_{ij}f^{ij}) = 2 \left( \sum_{i<j} \chi_{ij}(\sigma_i - \sigma_j)^2 + f^{ij}\nabla_j\nabla_i(\Delta f) + \nabla_k f_{ij}\nabla^k f^{ij} + f^{ij}f^k [2\nabla_i R_{jk} - \nabla_k R_{ij}] \right) \quad (\text{丁-1})$$

### 丁.2.2 SI 单位

使用 `siunitx` 宏包可以方便地输入 SI 单位制单位，例如 `\SI{5}{\um}` 可以得到  $5\mu\text{m}$ 。

#### 丁.2.2.1 一个四级标题

这是全文唯一的一个四级标题。在这部分中将演示了 `mathtools` 宏包中可伸长符号（箭头、等号的例子）的例子。

$$A \xleftarrow[n=0]{} B \xrightarrow[LongLongLongLong]{n>0} C$$

$$f(x) \xleftrightarrow{A=B} B \quad (\text{丁-2})$$

$$\underbrace{B}_{\text{below}}$$

$$\overleftrightarrow{\underbrace{B}_{\text{below}}} \quad (\text{丁-3})$$

<sup>1</sup>[https://en.wikibooks.org/wiki/LaTeX/List\\_Structures#Customizing\\_lists](https://en.wikibooks.org/wiki/LaTeX/List_Structures#Customizing_lists)

<sup>2</sup><http://www.tex.ac.uk/tex-archive/info/math/voss/mathmode/Mathmode.pdf>

又如：

$$\begin{aligned} & I(X_3; X_4) - I(X_3; X_4 | X_1) - I(X_3; X_4 | X_2) \\ &= [I(X_3; X_4) - I(X_3; X_4 | X_1)] - I(X_3; X_4 | \tilde{X}_2) \end{aligned} \quad (\text{丁-4})$$

$$= I(X_1; X_3; X_4) - I(X_3; X_4 | \tilde{X}_2) \quad (\text{丁-5})$$

### 丁.2.3 定理环境

模板中定义了丰富的定理环境 `algo`(算法), `thm`(定理), `lem`(引理), `prop`(命题), `cor`(推论), `defn`(定义), `conj`(猜想), `exmp`(例), `rem`(注), `case`(情形), `bthm`(断言定理), `blem`(断言引理), `bprop`(断言命题), `bcor`(断言推论)。amsmath 还提供了一个 `proof`(证明) 的环境。这里举一个“定理”和“证明”的例子。

**定理 丁.1** (留数定理). 假设  $U$  是复平面上的一个单连通开子集,  $a_1, \dots, a_n$  是复平面上有限个点,  $f$  是定义在  $U \setminus \{a_1, \dots, a_n\}$  上的全纯函数, 如果  $\gamma$  是一条把  $a_1, \dots, a_n$  包围起来的可求长曲线, 但不经过任何一个  $a_k$ , 并且其起点与终点重合, 那么:

$$\oint_{\gamma} f(z) dz = 2\pi i \sum_{k=1}^n I(\gamma, a_k) \text{Res}(f, a_k) \quad (\text{丁-6})$$

如果  $\gamma$  是若尔当曲线, 那么  $I(\gamma, a_k) = 1$ , 因此:

$$\oint_{\gamma} f(z) dz = 2\pi i \sum_{k=1}^n \text{Res}(f, a_k) \quad (\text{丁-7})$$

在这里,  $\text{Res}(f, a_k)$  表示  $f$  在点  $a_k$  的留数,  $I(\gamma, a_k)$  表示  $\gamma$  关于点  $a_k$  的卷绕数。卷绕数是一个整数, 它描述了曲线  $\gamma$  绕过点  $a_k$  的次数。如果  $\gamma$  依逆时针方向绕着  $a_k$  移动, 卷绕数就是一个正数, 如果  $\gamma$  根本不绕过  $a_k$ , 卷绕数就是零。

定理丁.1的证明。

**证明.** 首先, 由……

其次, ……

所以…… □

上面的公式例子中, 有一些细节希望大家注意。微分号  $d$  应该使用“直立体”也就是用 `mathrm` 包围起来。并且, 微分号和被积函数之间应该有一段小间隔, 可以插入 `\,` 得到。斜体的  $d$  通常只作为一般变量。i,j 作为虚数单位时, 也应该使用“直立体”为了

明显，还加上了粗体，例如 $\mathbf{i}$ 。斜体  $i, j$  通常用作表示“序号”。其他字母在表示常量时，也推荐使用“直立体”譬如，圆周率  $\pi$  (需要 `upgreek` 宏包)，自然对数的底  $e$ 。不过，我个人觉得斜体的  $e$  和  $\pi$  很潇洒，在不至于引起混淆的情况下，我也用这两个字母的斜体表示对应的常量。

## 丁.3 向文档中插入图像

### 丁.3.1 支持的图片格式

X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X 可以很方便地插入 PDF、PNG、JPG 格式的图片。

插入 PNG/JPG 的例子如丁-1所示。这两个水平并列放置的图共享一个“图标题”(table caption)，没有各自的小标题。

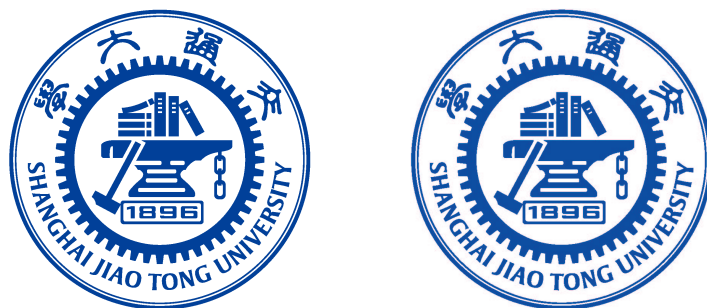


图 丁-1 中文题图

Figure 丁-1 English caption

这里还有插入 EPS 图像和 PDF 图像的例子，如图丁-2a和图丁-2b。这里将 EPS 和 PDF 图片作为子图插入，每个子图有自己的小标题。子图标题使用 `subcaption` 宏包添加。



(a) EPS 图像

上海交通大学

SHANGHAI JIAO TONG UNIVERSITY

(b) PDF 图像，注意这个图略矮些。如果标题很长的话，它会自动换行

图 丁-2 插入 eps 和 pdf 的例子 (使用 `subcaptionbox` 方式)

Figure 丁-2 An EPS and PDF demo with `subcaptionbox`

更多关于 L<sup>A</sup>T<sub>E</sub>X 插图例子可以参考《L<sup>A</sup>T<sub>E</sub>X 插图指南》<sup>1</sup>。

<sup>1</sup><http://www.cs.duke.edu/junhu/Graphics3.pdf>



(a) EPS 图像

上海交通大学

SHANGHAI JIAO TONG UNIVERSITY

(b) PDF 图像，注意这个图略矮些。subfigure 中同一行的子图在顶端对齐。

图 丁-3 插入 eps 和 pdf 的例子 (使用 subfigure 方式)

Figure 丁-3 An EPS and PDF demo with subfigure

### 丁.3.2 长标题的换行

图丁-4和图丁-5都有比较长图标题，通过对比发现，图丁-5的换行效果更好一些。其中使用了 minipage 环境来限制整个浮动体的宽度。



图 丁-4 上海交通大学是我国历史最悠久的高等学府之一，是教育部直属、教育部与上海市共建的全国重点大学。

Figure 丁-4 Where there is a will, there is a way.

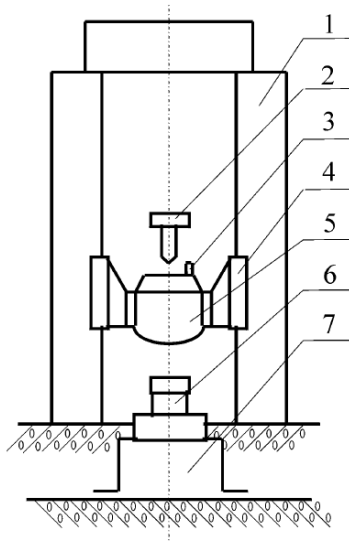


图 丁-5 上海交通大学是我国历史最悠久的高等学府之一，是教育部直属、教育部与上海市共建的全国重点大学。

Figure 丁-5 Where there is a will, there is a way.

### 丁.3.3 添加图注

当插图中组成部件由数字或字母等编号表示时，可在插图下方添加图注进行说明，如图丁-6所示。



1. 立柱 2. 提升释放机构 3. 标准冲击加速度计
4. 导轨 5. 重锤 6. 被校力传感器 7. 底座

图丁-6 示例图片来源于 [1]

Figure 丁-6 Stay hungry, stay foolish.

### 丁.3.4 绘制流程图

图丁-7是一张流程图示意。使用 tikz 环境，搭配四种预定义节点 (`startstop`、`process`、`decision` 和 `io`)，可以容易地绘制出流程图。



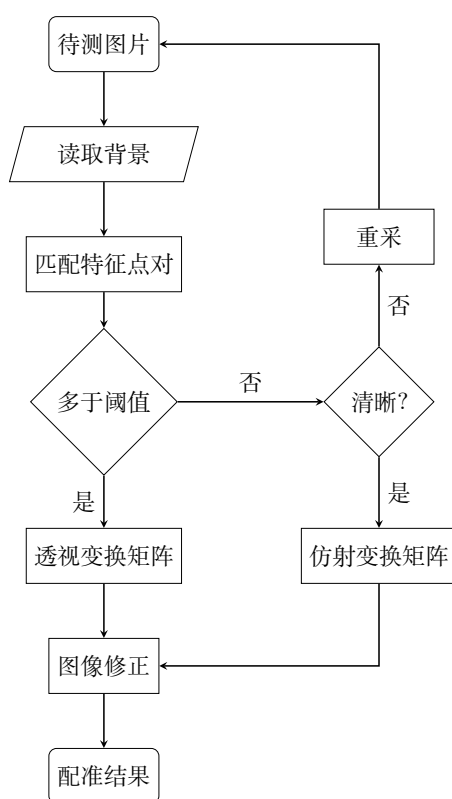


图 丁-7 绘制流程图效果

Figure 丁-7 Flow chart

## 丁.4 表格

这一节给出的是一些表格的例子，如表丁-1所示。

表 丁-1 一个颇为标准的三线表格<sup>1</sup>

Table 丁-1 A Table

Item		
Animal	Description	Price (\$)
Gnat	per gram	13.65
	each	0.01
Gnu	stuffed	92.50
Emu	stuffed	33.33
Armadillo	frozen	8.99

下面一个是一个更复杂的表格，用 `threeparttable` 实现带有脚注的表格，如表丁-2。

表 丁-2 一个带有脚注的表格的例子

Table 丁-2 A Table with footnotes

total	20 <sup>1</sup>		40		60	
	www	k	www	k	www	k
4.22 (2.12)	120.0140 <sup>2</sup>	333.15	0.0411	444.99	0.1387	
168.6123	10.86	255.37	0.0353	376.14	0.1058	
6.761	0.007	235.37	0.0267	348.66	0.1010	

<sup>1</sup> the first note.

<sup>2</sup> the second note.

## 丁.5 参考文献管理

L<sup>A</sup>T<sub>E</sub>X 具有将参考文献内容和表现形式分开管理的能力，涉及三个要素：参考文献数据库、参考文献引用格式、在正文中引用参考文献。这样的流程需要多次编译：

<sup>1</sup>这个例子来自《Publication quality tables in L<sup>A</sup>T<sub>E</sub>X》<sup>1</sup>(booktabs 宏包的文档)。这也是一个在表格中使用脚注的例子，请注意与 `threeparttable` 实现的效果有何不同。

1. 用户将论文中需要引用的参考文献条目，录入纯文本数据库文件 (bib 文件)。
2. 调用 xelatex 对论文模板做第一次编译，扫描文中引用的参考文献，生成参考文献入口文件 (aux) 文件。
3. 调用 bibtex，以参考文献格式和入口文件为输入，生成格式化以后的参考文献条目文件 (bib)。
4. 再次调用 xelatex 编译模板，将格式化以后的参考文献条目插入正文。

参考文献数据库 (thesis.bib) 的条目，可以从 Google Scholar 搜索引擎<sup>1</sup>、CiteSeerX 搜索引擎<sup>2</sup>中查找，文献管理软件 Papers<sup>3</sup>、Mendeley<sup>4</sup>、JabRef<sup>5</sup>也能够输出条目信息。

下面是在 Google Scholar 上搜索到的一条文献信息，格式是纯文本：

代码 丁-1 从 Google Scholar 找到的参考文献条目

```
@phdthesis{白 2008 信用风险传染模型和信用衍生品的定价，
  title={信用风险传染模型和信用衍生品的定价}，
  author={白云芬}，
  year={2008}，
  school={上海交通大学}
}
```

推荐修改后在 bib 文件中的内容为：

代码 丁-2 修改后的参考文献条目

```
@phdthesis{bai2008,
  title={信用风险传染模型和信用衍生品的定价}，
  author={白云芬}，
  date={2008}，
  address={上海}，
  school={上海交通大学}
}
```

按照教务处的要求，参考文献外观应符合国标 GB/T 7714 的要求<sup>6</sup>。在模板中，表现形式的控制逻辑通过 biblatex-gb7714-2015 包实现<sup>7</sup>，基于 {BibL<sup>A</sup>T<sub>E</sub>X} 管理文献。在目前的多数 TeX 发行版中，可能都没有默认包含 biblatex-gb7714-2015，需要手动安装。

<sup>1</sup><https://scholar.google.com>

<sup>2</sup><http://citeseerx.ist.psu.edu>

<sup>3</sup><http://papersapp.com>

<sup>4</sup><http://www.mendeley.com>

<sup>5</sup><http://jabref.sourceforge.net>

<sup>6</sup>[http://www.cces.net.cn/guild/sites/tmxb/Files/19798\\_2.pdf](http://www.cces.net.cn/guild/sites/tmxb/Files/19798_2.pdf)

<sup>7</sup><https://www.ctan.org/pkg/biblatex-gb7714-2015>

正文中引用参考文献时，用 `\cite{key1,key2,key3...}` 可以产生“上标引用的参考文献”，如<sup>[2-4]</sup>。使用 `\parencite{key1,key2,key3...}` 则可以产生水平引用的参考文献，例如 [5-7]。请看下面的例子，将会穿插使用水平的和上标的参考文献：关于书的 [2, 5, 7]，关于期刊的<sup>[3, 8]</sup>，会议论文 [4, 9, 10]，硕士学位论文 [6, 11]，博士学位论文<sup>[12-14]</sup>，标准文件 [7]，技术报告<sup>[15]</sup>，电子文献 [16, 17]，用户手册 [18]。

总结一些注意事项：

- 参考文献只有在正文中被引用了，才会在最后的参考文献列表中出现；
- 参考文献“数据库文件” bib 是纯文本文件，请使用 UTF-8 编码，不要使用 GBK 编码；
- 参考文献条目中默认通过 date 域输入时间。兼容使用 year 域时会产生编译 warning，可忽略。

## 丁.6 用 listings 插入源代码

原先 ctexbook 文档类和 listings 宏包配合使用时，代码在换页时会出现莫名其妙的错误，后来经高人指点，顺利解决了。感兴趣的话，可以看看这里<sup>1</sup>。这里给使用 listings 宏包插入源代码的例子，这里是一段 C 代码。另外，listings 宏包真可谓博大精深，可以实现各种复杂、漂亮的效果，想要进一步学习的同学，可以参考 listings 宏包手册<sup>2</sup>。

代码 丁-3 一段 C 源代码

```
1 #include <stdio.h>
2 #include <unistd.h>
3 #include <sys/types.h>
4 #include <sys/wait.h>
5
6 int main() {
7     pid_t pid;
8
9     switch ((pid = fork())) {
10    case -1:
11        printf("fork failed\n");
12        break;
13    case 0:
14        /* child calls exec */
15        execl("/bin/ls", "ls", "-l", (char*)0);
16        printf("execl failed\n");
```

<sup>1</sup><http://bbs.ctex.org/viewthread.php?tid=53451>

<sup>2</sup><http://mirror.ctan.org/macros/latex/contrib/listings/listings.pdf>

```

17     break;
18 default:
19     /* parent uses wait to suspend execution until child finishes */
20     wait((int*)0);
21     printf("is completed\n");
22     break;
23 }
24
25 return 0;
26 }

```

## 丁.7 用 `algorithm` 和 `algorithmicx` 宏包插入算法描述

`algorithmicx` 比 `algorithmic` 增加了一些命令。示例如算法丁-1和算法丁-2，后者的代码来自 xhSong 的博客<sup>1</sup>。`algorithmicx` 的详细使用方法见官方 README<sup>2</sup>。使用算法宏包时，算法出现的位置很多时候不按照 `tex` 文件里的书写顺序，需要强制定位时可以使用 `\begin{algorithm}[H]`<sup>3</sup>

这是写在算法丁-1前面的一段话，在生成的文件里它会出现在算法丁-1前面。

---

### 算法 丁-1 求 100 以内的整数和

---

输出: 100 以内的整数和

```

1:  $sum \leftarrow 0$ 
2: for  $i = 0 \rightarrow 100$  do
3:    $sum \leftarrow sum + i$ 
4: end for

```

---

这是写在两个算法中间的一段话，当算法丁-1不使用 `\begin{algorithm}[H]` 时它也会出现在算法丁-1前面。

对于很长的算法，单一的算法块 `\begin{algorithm}...\end{algorithm}` 是不能自动跨页的<sup>4</sup>，会出现的情况有：

- 该页放不下当前的算法，留下大片空白，算法在下一页显示
- 单一页面放不下当前的算法，显示时超过页码的位置直到超出整个页面范围

<sup>1</sup><http://hustsxh.is-programmer.com/posts/38801.html>

<sup>2</sup><http://mirror.hust.edu.cn/CTAN/macros/latex/contrib/algorithmicx/algorithmicx.pdf>

<sup>3</sup><http://tex.stackexchange.com/questions/165021/fixing-the-location-of-the-appearance-in-algorithmicx-environment>

<sup>4</sup><http://tex.stackexchange.com/questions/70733/latex-algorithm-not-display-under-correct-section>

解决方法有：

- (推荐) 使用 `algstore{algnam}` 和 `algrestore{algnam}` 来讲算法分为两个部分<sup>1</sup>，如算法丁-2。
- 人工拆分算法为多个小的部分。

---

### 算法 丁-2 用归并排序求逆序数

---

输入: *Array* 数组, *n* 数组大小

输出: 逆序数

```

1: function MERGESORT(Array, left, right)
2:   result  $\leftarrow$  0
3:   if left < right then
4:     middle  $\leftarrow$  (left + right)/2
5:     result  $\leftarrow$  result + MERGESORT(Array, left, middle)
6:     result  $\leftarrow$  result + MERGESORT(Array, middle, right)
7:     result  $\leftarrow$  result + MERGER(Array, left, middle, right)
8:   end if
9:   return result
10: end function
11:
12: function MERGER(Array, left, middle, right)
13:   i  $\leftarrow$  left
14:   j  $\leftarrow$  middle
15:   k  $\leftarrow$  0
16:   result  $\leftarrow$  0
17:   while i < middle and j < right do
18:     if Array[i] < Array[j] then
19:       B[k ++]  $\leftarrow$  Array[i ++]
20:     else
21:       B[k ++]  $\leftarrow$  Array[j ++]
22:       result  $\leftarrow$  result + (middle - i)
23:     end if
24:   end while

```

---

<sup>1</sup><http://tex.stackexchange.com/questions/29816/algorithm-over-2-pages>

---

```

25:  while  $i < middle$  do
26:       $B[k ++] \leftarrow Array[i ++]$ 
27:  end while
28:  while  $j < right$  do
29:       $B[k ++] \leftarrow Array[j ++]$ 
30:  end while
31:  for  $i = 0 \rightarrow k - 1$  do
32:       $Array[left + i] \leftarrow B[i]$ 
33:  end for
34:  return  $result$ 
35: end function

```

---

这是写在算法丁-2后面的一段话,但是当算法丁-2不使用`\begin{algorithm}[H]`时它会出现现在算法丁-2 甚至算法丁-1前面。

对于算法的索引要注意`\caption`和`\label`的位置,必须是先`\caption`再`\label`<sup>1</sup>,否则会出现`\ref{algo:sum_100}`生成的编号跟对应算法上显示不一致的问题。

根据 Werner 的回答<sup>2</sup> 增加了Switch 和Case 的支持, 见算法丁-3。

---

### 算法 丁-3 Switch 示例

---

```

1: switch ( $s$ )
2:   case  $a$ :
3:      $assert(0)$ 
4:   case  $b$ :
5:      $assert(1)$ 
6:   default :
7:      $assert(2)$ 
8: end switch

```

---

<sup>1</sup><http://tex.stackexchange.com/questions/65993/algorithm-numbering>

<sup>2</sup><http://tex.stackexchange.com/questions/53357/switch-cases-in-algorithmic>





## 附录戊 RMarkdown/Bookdown 排版示例

本文无意全文翻译 RMarkdown/Bookdown 的使用方法，只是简单的复制，详细内容请参考页面：

- <https://rmarkdown.rstudio.com/lesson-1.html>
- <https://bookdown.org/yihui/bookdown/>
- <https://bookdown.org/yihui/bookdown/components.html>
- <http://pandoc.org/>

以下为全文复制 <https://bookdown.org/yihui/bookdown/components.html>，有改动。

This chapter demonstrates the syntax of common components of a book written in **bookdown**, including code chunks, figures, tables, citations, math theorems, and equations. The approach is based on Pandoc, so we start with the syntax of Pandoc's flavor of Markdown.

### 戊.1 Markdown syntax

In this section, we give a very brief introduction to Pandoc's Markdown. Readers who are familiar with Markdown can skip this section. The comprehensive syntax of Pandoc's Markdown can be found on the Pandoc website <http://pandoc.org>.

#### 戊.1.1 Inline formatting

You can make text *italic* by surrounding it with underscores or asterisks, e.g., `_text_` or `*text*`. For **bold** text, use two underscores (`__text__`) or asterisks (`**text**`). Text surrounded by `~` will be converted to a subscript (e.g., `H~2~S0~4~` renders  $H_2SO_4$ ), and similarly, two carets (`^`) produce a superscript (e.g., `Fe^2+^` renders  $Fe^{2+}$ ). To mark text as inline code, use a pair of backticks, e.g., ``code``.<sup>1</sup> Small caps can be produced by the HTML tag `span`, e.g., `<span style="font-variant:small-caps;">Small Caps</span>` renders SMALL CAPS. Links are created using `[text](link)`, e.g., `[RStudio](https://www.rstudio.com)`, and the syntax for images is similar: just add an exclamation mark, e.g., `![alt text or image title](path/to/image)`. Footnotes

---

<sup>1</sup>To include literal backticks, use more backticks outside, e.g., you can use two backticks to preserve one backtick inside: ```code```.

are put inside the square brackets after a caret `^[]`, e.g., `^[This is a footnote.]`. We will talk about citations in Section 戊.8.

## 戊.1.2 Block-level elements

Section headers can be written after a number of pound signs, e.g.,

```
1 # First-level header
2
3 ## Second-level header
4
5 ### Third-level header
```

If you do not want a certain heading to be numbered, you can add `{-}` after the heading, e.g.,

```
1 # Preface {-}
```

Unordered list items start with `*`, `-`, or `+`, and you can nest one list within another list by indenting the sub-list by four spaces, e.g.,

```
1 - one item
2 - one item
3 - one item
4   - one item
5   - one item
```

The output is:

- one item
- one item
- one item
  - one item
  - one item

Ordered list items start with numbers (the rule for nested lists is the same as above), e.g.,

```
1 1. the first item
2 2. the second item
3 3. the third item
```

The output does not look too much different with the Markdown source:

1. the first item
2. the second item
3. the third item

Blockquotes are written after `>`, e.g.,

```
1 > "I thoroughly disapprove of duels. If a man should challenge me,
2   I would take him kindly and forgivingly by the hand and lead him
3   to a quiet place and kill him."
4 >
5 > --- Mark Twain
```

The actual output (we customized the style for blockquotes in this book):

“I thoroughly disapprove of duels. If a man should challenge me, I would take him kindly and forgivingly by the hand and lead him to a quiet place and kill him.”

— Mark Twain

Plain code blocks can be written after three or more backticks, and you can also indent the blocks by four spaces, e.g.,

```
1 ```
2 This text is displayed verbatim / preformatted
3 ```
4
5 Or indent by four spaces:
6
7     This text is displayed verbatim / preformatted
```

### 戊.1.3 Math expressions

Inline LaTeX equations can be written in a pair of dollar signs using the LaTeX syntax, e.g., `$f(k) = \binom{n}{k} p^k (1-p)^{n-k}$` (actual output:  $f(k) = \binom{n}{k} p^k (1-p)^{n-k}$ ); math expressions of the display style can be written in a pair of double dollar signs, e.g., `$$f(k) = \binom{n}{k} p^k (1-p)^{n-k}$$`, and the output looks like this:

$$f(k) = \binom{n}{k} p^k (1-p)^{n-k}$$

You can also use math environments inside `$ $` or `$$ $$`, e.g.,

```

1 \begin{array}{ccc}
2 x_{11} & x_{12} & x_{13} \\
3 x_{21} & x_{22} & x_{23} \\
4 \end{array}

```

$$\begin{array}{ccc} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \end{array}$$

```

1 X = \begin{bmatrix} 1 & x_{1} \\
2 1 & x_{2} \\
3 1 & x_{3} \\
4 \end{bmatrix}

```

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \end{bmatrix}$$

```

1 \Theta = \begin{pmatrix} \alpha & \beta \\
2 \gamma & \delta \\
3 \end{pmatrix}

```

$$\Theta = \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}$$

```

1 \begin{vmatrix} a & b \\
2 c & d \\
3 \end{vmatrix} = ad - bc

```

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

## 戊.2 Markdown extensions by bookdown

Although Pandoc's Markdown is much richer than the original Markdown syntax, it still lacks a number of things that we may need for academic writing. For example, it supports

math equations, but you cannot number and reference equations in multi-page HTML or EPUB output. We have provided a few Markdown extensions in **bookdown** to fill the gaps.

### 戊.2.1 Number and reference equations

To number and refer to equations, put them in the equation environments and assign labels to them using the syntax (`\##eq:label`), e.g.,

```
1 \begin{equation}
2   f\left(k\right) = \binom{n}{k} p^k \left(1-p\right)^{n-k}
3   \label{eq:binom}
4 \end{equation}
```

It renders the equation below:

$$f(k) = \binom{n}{k} p^k (1-p)^{n-k} \quad (\text{戊-1})$$

You may refer to it using `\eqref{eq:binom}`, e.g., see Equation (戊-1).



Equation labels must start with the prefix `eq:` in **bookdown**. All labels in **bookdown** must only contain alphanumeric characters, `:`, `-`, and/or `/`. Equation references work best for LaTeX/PDF output, and they are not well supported in Word output or e-books. For HTML output, **bookdown** can only number the equations with labels. Please make sure equations without labels are not numbered by either using the `equation*` environment or adding `\nonumber` or `\notag` to your equations. The same rules apply to other math environments, such as `eqnarray`, `gather`, `align`, and so on (e.g., you can use the `align*` environment).

We demonstrate a few more math equation environments below. Here is an unnumbered equation using the `equation*` environment:

```
1 \begin{equation*}
2   \frac{d}{dx} \left( \int_a^x f(u) du \right) = f(x)
3 \end{equation*}
```

$$\frac{d}{dx} \left( \int_a^x f(u) du \right) = f(x)$$

Below is an `align` environment (戊-2):

```

1 \begin{align}
2 g(X_{\{n\}}) &= g(\theta) + g'(\tilde{\theta})(X_{\{n\}} - \theta) \notag \\
3 \sqrt{n}[g(X_{\{n\}}) - g(\theta)] &= g'(\tilde{\theta})\sqrt{n}[X_{\{n\}} - \theta] \\
4 \sqrt{n}[X_{\{n\}} - \theta] &\label{eq:align} \\
5 \end{align}

```

$$\begin{aligned}
 g(X_n) &= g(\theta) + g'(\tilde{\theta})(X_n - \theta) \\
 \sqrt{n}[g(X_n) - g(\theta)] &= g'(\tilde{\theta})\sqrt{n}[X_n - \theta]
 \end{aligned}
 \tag{戊-2}$$

You can use the `split` environment inside `equation` so that all lines share the same number (戊-3). By default, each line in the `align` environment will be assigned an equation number. We suppressed the number of the first line in the previous example using `\notag`. In this example, the whole `split` environment was assigned a single number.

```

1 \begin{equation}
2 \begin{split}
3 \mathrm{Var}(\hat{\beta}) &= \mathrm{Var}((X'X)^{-1}X'y) \\
4 &= (X'X)^{-1}X'\mathrm{Var}(y)((X'X)^{-1}X')' \\
5 &= (X'X)^{-1}X'\mathrm{Var}(y)X(X'X)^{-1} \\
6 &= (X'X)^{-1}X'\sigma^2IX(X'X)^{-1} \\
7 &= (X'X)^{-1}\sigma^2 \\
8 \end{split} \\
9 \label{eq:var-beta} \\
10 \end{equation}

```

$$\begin{aligned}
 \mathrm{Var}(\hat{\beta}) &= \mathrm{Var}((X'X)^{-1}X'y) \\
 &= (X'X)^{-1}X'\mathrm{Var}(y)((X'X)^{-1}X')' \\
 &= (X'X)^{-1}X'\mathrm{Var}(y)X(X'X)^{-1} \\
 &= (X'X)^{-1}X'\sigma^2IX(X'X)^{-1} \\
 &= (X'X)^{-1}\sigma^2
 \end{aligned}
 \tag{戊-3}$$

## 戊.2.2 Theorems and proofs

Theorems and proofs are commonly used in articles and books in mathematics. However, please do not be misled by the names: a “theorem” is just a numbered/labeled environment,

表 戊-1 Theorem environments in **bookdown**.

Environment	Printed Name	Label Prefix
theorem	Theorem	thm
lemma	Lemma	lem
corollary	Corollary	cor
proposition	Proposition	prp
conjecture	Conjecture	cnj
definition	Definition	def
example	Example	exm
exercise	Exercise	exr

and it does not have to be a mathematical theorem (e.g., it can be an example irrelevant to mathematics). Similarly, a “proof” is an unnumbered environment. In this section, we always use the *general* meanings of a “theorem” and “proof” unless explicitly stated.

In **bookdown**, the types of theorem environments supported are in Table 戊-1. To write a theorem, you can use the syntax below:

```
1 ```{theorem}
2 Here is my theorem.
3 ```
```

To write other theorem environments, replace ````{theorem}` with other environment names in Table 戊-1, e.g., ````{lemma}`.

A theorem can have a name option so its name will be printed, e.g.,

```
1 ```{theorem, name="Pythagorean theorem"}
2 For a right triangle, if  $c$  denotes the length of the hypotenuse
3 and  $a$  and  $b$  denote the lengths of the other two sides, we have
4  $a^2 + b^2 = c^2$ 
5 ```
```

If you want to refer to a theorem, you should label it. The label can be written after ````{theorem}`, e.g.,

```
1 ```{theorem, label="foo"}
2 A labeled theorem here.
```

```
3 ```
```

The `label` option can be implicit, e.g., the following theorem has the label `bar`:

```
1 ```{theorem, bar}
2 A labeled theorem here.
3 ```
```

After you label a theorem, you can refer to it using the syntax `\ref {prefix:label}`. See the column `Label Prefix` in Table 戊-1 for the value of `prefix` for each environment. For example, we have a labeled and named theorem below, and `\ref {thm:pyth}` gives us its theorem number 戊.1:

```
1 ```{theorem, pyth, name="Pythagorean theorem"}
2 For a right triangle, if  $c$  denotes the length of the hypotenuse
3 and  $a$  and  $b$  denote the lengths of the other two sides, we have
4
5  $a^2 + b^2 = c^2$ 
6 ```
```

**定理 戊.1** (Pythagorean theorem). *For a right triangle, if  $c$  denotes the length of the hypotenuse and  $a$  and  $b$  denote the lengths of the other two sides, we have*

$$a^2 + b^2 = c^2$$

The proof environments currently supported are `proof`, `remark`, and `solution`. The syntax is similar to theorem environments, and proof environments can also be named. The only difference is that since they are unnumbered, you cannot reference them.

We have tried to make all these theorem and proof environments work out of the box, no matter if your output is PDF, HTML, or EPUB. If you are a LaTeX or HTML expert, you may want to customize the style of these environments anyway (see Chapter ??). Customization in HTML is easy with CSS, and each environment is enclosed in `<div></div>` with the CSS class being the environment name, e.g., `<div class="lemma"></div>`. For LaTeX output, we have predefined the style to be `definition` for environments `definition`, `example`, and `exercise`, and `remark` for environments `proof` and `remark`. All other environments use the `plain` style. The style definition is done through the `\theoremstyle{}` command of the **amsthm** package.

Theorems are numbered by chapters by default. If there are no chapters in your document, they are numbered by sections instead. If the whole document is unnumbered (the output format



option `number_sections = FALSE`), all theorems are numbered sequentially from 1, 2, ..., N. LaTeX supports numbering one theorem environment after another, e.g., let theorems and lemmas share the same counter. This is not supported for HTML/EPUB output in **bookdown**. You can change the numbering scheme in the LaTeX preamble by defining your own theorem environments, e.g.,

```
1 \newtheorem{theorem}{Theorem}
2 \newtheorem{lemma}[theorem]{Lemma}
```

When **bookdown** detects `\newtheorem{theorem}` in your LaTeX preamble, it will not write out its default theorem definitions, which means you have to define all theorem environments by yourself. For the sake of simplicity and consistency, we do not recommend that you do this. It can be confusing when your Theorem 18 in PDF becomes Theorem 2.4 in HTML.

Theorem and proof environments will be hidden if the chunk option `echo` is set to `FALSE`. To make sure they are always shown, you may add the chunk option `echo=TRUE`, e.g.,

```
1 ```{theorem, echo=TRUE}
2 Here is my theorem.
3 ```
```

Below we show more examples<sup>1</sup> of the theorem and proof environments, so you can see the default styles in **bookdown**.

**定义 戊.1.** The characteristic function of a random variable  $X$  is defined by

$$\varphi_X(t) = E[e^{itX}], \quad t \in \mathcal{R}$$

**例 戊.1.** We derive the characteristic function of  $X \sim U(0, 1)$  with the probability density function  $f(x) = \mathbf{1}_{x \in [0, 1]}$ .

---

<sup>1</sup>Some examples are adapted from the Wikipedia page [https://en.wikipedia.org/wiki/Characteristic\\_function\\_\(probability\\_theory\)](https://en.wikipedia.org/wiki/Characteristic_function_(probability_theory))

$$\begin{aligned}
\varphi_X(t) &= \mathbb{E} [e^{itX}] \\
&= \int e^{itx} f(x) dx \\
&= \int_0^1 e^{itx} dx \\
&= \int_0^1 (\cos(tx) + i \sin(tx)) dx \\
&= \left( \frac{\sin(tx)}{t} - i \frac{\cos(tx)}{t} \right) \Big|_0^1 \\
&= \frac{\sin(t)}{t} - i \left( \frac{\cos(t) - 1}{t} \right) \\
&= \frac{i \sin(t)}{it} + \frac{\cos(t) - 1}{it} \\
&= \frac{e^{it} - 1}{it}
\end{aligned}$$

Note that we used the fact  $e^{ix} = \cos(x) + i \sin(x)$  twice.

**引理 戊.2.** *For any two random variables  $X_1, X_2$ , they both have the same probability distribution if and only if*

$$\varphi_{X_1}(t) = \varphi_{X_2}(t)$$

**定理 戊.3.** *If  $X_1, \dots, X_n$  are independent random variables, and  $a_1, \dots, a_n$  are some constants, then the characteristic function of the linear combination  $S_n = \sum_{i=1}^n a_i X_i$  is*

$$\varphi_{S_n}(t) = \prod_{i=1}^n \varphi_{X_i}(a_i t) = \varphi_{X_1}(a_1 t) \cdots \varphi_{X_n}(a_n t)$$

**命题 戊.4.** *The distribution of the sum of independent Poisson random variables  $X_i \sim \text{Pois}(\lambda_i)$ ,  $i = 1, 2, \dots, n$  is  $\text{Pois}(\sum_{i=1}^n \lambda_i)$ .*

**证明.** The characteristic function of  $X \sim \text{Pois}(\lambda)$  is  $\varphi_X(t) = e^{\lambda(e^{it}-1)}$ . Let  $P_n = \sum_{i=1}^n X_i$ . We know from Theorem 戊.3 that

$$\begin{aligned}\varphi_{P_n}(t) &= \prod_{i=1}^n \varphi_{X_i}(t) \\ &= \prod_{i=1}^n e^{\lambda_i(e^{it}-1)} \\ &= e^{\sum_{i=1}^n \lambda_i(e^{it}-1)}\end{aligned}$$

This is the characteristic function of a Poisson random variable with the parameter  $\lambda = \sum_{i=1}^n \lambda_i$ . From Lemma 戊.2, we know the distribution of  $P_n$  is  $\text{Pois}(\sum_{i=1}^n \lambda_i)$ .  $\square$

注 1. In some cases, it is very convenient and easy to figure out the distribution of the sum of independent random variables using characteristic functions.

**推论 戊.5.** *The characteristic function of the sum of two independent random variables  $X_1$  and  $X_2$  is the product of characteristic functions of  $X_1$  and  $X_2$ , i.e.,*

$$\varphi_{X_1+X_2}(t) = \varphi_{X_1}(t)\varphi_{X_2}(t)$$

**练习 戊.1** (Characteristic Function of the Sample Mean). Let  $\bar{X} = \sum_{i=1}^n \frac{1}{n} X_i$  be the sample mean of  $n$  independent and identically distributed random variables, each with characteristic function  $\varphi_X$ . Compute the characteristic function of  $\bar{X}$ .

**解答.** Applying Theorem 戊.3, we have

$$\varphi_{\bar{X}}(t) = \prod_{i=1}^n \varphi_{X_i}\left(\frac{t}{n}\right) = \left[\varphi_X\left(\frac{t}{n}\right)\right]^n.$$

### 戊.2.3 Special headers

There are a few special types of first-level headers that will be processed differently in **bookdown**. The first type is an unnumbered header that starts with the token `(PART)`. This kind of headers are translated to part titles. If you are familiar with LaTeX, this basically means `\part{}`. When your book has a large number of chapters, you may want to organize them into parts, e.g.,

```
1 # (PART) Part I {-}
2
3 # Chapter One
```

```
4 |
5 | # Chapter Two
6 |
7 | # (PART) Part II {-}
8 |
9 | # Chapter Three
```

A part title should be written right before the first chapter title in this part. You can use `(PART\*)` (the backslash before `*` is required) instead of `(PART)` if a part title should not be numbered.

The second type is an unnumbered header that starts with `(APPENDIX)`, indicating that all chapters after this header are appendices, e.g.,

```
1 | # Chapter One
2 |
3 | # Chapter Two
4 |
5 | # (APPENDIX) Appendix {-}
6 |
7 | # Appendix A
8 |
9 | # Appendix B
```

The numbering style of appendices will be automatically changed in LaTeX/PDF and HTML output (usually in the form A, A.1, A.2, B, B.1, ...). This feature is not available to e-books or Word output.

## 戊.2.4 Text references

You can assign some text to a label and reference the text using the label elsewhere in your document. This can be particularly useful for long figure/table captions (Section 戊.4 and 戊.5), in which case you normally will have to write the whole character string in the chunk header (e.g., `fig.cap = "A long long figure caption."`) or your R code (e.g., `kable(caption = "A long long table caption.")`). It is also useful when these captions contain special HTML or LaTeX characters, e.g., if the figure caption contains an underscore, it works in the HTML output but may not work in LaTeX output because the underscore must be escaped in LaTeX.

The syntax for a text reference is `(ref:label) text`, where `label` is a unique label<sup>1</sup> through-

---

<sup>1</sup>You may consider using the code chunk labels.

out the document for `text`. It must be in a separate paragraph with empty lines above and below it. The paragraph must not be wrapped into multiple lines, and should not end with a white space. For example,

```
1 (ref:foo) Define a text reference here.
```

Then you can use `(ref:foo)` in your figure/table captions. The text can contain anything that Markdown supports, as long as it is one single paragraph. Here is a complete example:

```
1 A normal paragraph.
2
3 (ref:foo) A scatterplot of the data `cars` using base R graphics.
4
5 ```{r foo, fig.cap='(ref:foo)'}
6 plot(cars) # a scatterplot
7 ```
```

Text references can be used anywhere in the document (not limited to figure captions). It can also be useful if you want to reuse a fragment of text in multiple places.

### 戊.3 R code

There are two types of R code in R Markdown/**knitr** documents: R code chunks, and inline R code. The syntax for the latter is ``r R_CODE``, and it can be embedded inline with other document elements. R code chunks look like plain code blocks, but have `{r}` after the three backticks and (optionally) chunk options inside `{}`, e.g.,

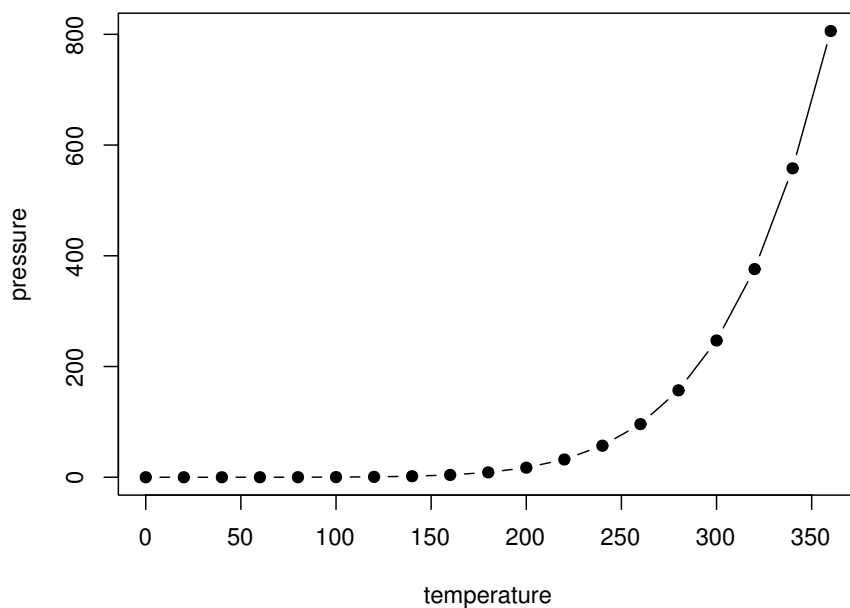
```
1 ```{r chunk-label, echo = FALSE, fig.cap = 'A figure caption.'}
2 1 + 1
3 rnorm(10) # 10 random numbers
4 plot(dist ~ speed, cars) # a scatterplot
5 ```
```

To learn more about **knitr** chunk options, see Xie [19] or the web page <http://yihui.name/knitr/options>. For books, additional R code can be executed before/after each chapter; see `before_chapter_script` and `after_chapter_script` in Section ??.

## 戊.4 Figures

By default, figures have no captions in the output generated by **knitr**, which means they will be placed wherever they were generated in the R code. Below is such an example.

```
1 par(mar = c(4, 4, .1, .1))
2 plot(pressure, pch = 19, type = 'b')
```



The disadvantage of typesetting figures in this way is that when there is not enough space on the current page to place a figure, it may either reach the bottom of the page (hence exceeds the page margin), or be pushed to the next page, leaving a large white margin at the bottom of the current page. That is basically why there are “floating environments” in LaTeX: elements that cannot be split over multiple pages (like figures) are put in floating environments, so they can float to a page that has enough space to hold them. There is also a disadvantage of floating things forward or backward, though. That is, readers may have to jump to a different page to find the figure mentioned on the current page. This is simply a natural consequence of having to typeset things on multiple pages of fixed sizes. This issue does not exist in HTML, however, since everything can be placed continuously on one single page (presumably with infinite height), and there is no need to split anything across multiple pages of the same page size.

If we assign a figure caption to a code chunk via the chunk option `fig.cap`, R plots will be put into figure environments, which will be automatically labeled and numbered, and can also be cross-referenced. The label of a figure environment is generated from the label of the code chunk, e.g., if the chunk label is `foo`, the figure label will be `fig:foo` (the prefix `fig:` is added

before `foo`). To reference a figure, use the syntax `\ref {label}`,<sup>1</sup> where `label` is the figure label, e.g., `fig:foo`.

To take advantage of Markdown formatting *within* the figure caption, you will need to use text references (see Section 戊.2.4). For example, a figure caption that contains `_italic text_` will not work when the output format is LaTeX/PDF, since the underscore is a special character in LaTeX, but if you use text references, `_italic text_` will be translated to LaTeX code when the output is LaTeX.



If you want to cross-reference figures or tables generated from a code chunk, please make sure the chunk label only contains *alphanumeric* characters (a-z, A-Z, 0-9), slashes (/), or dashes (-).

The chunk option `fig.asp` can be used to set the aspect ratio of plots, i.e., the ratio of figure height/width. If the figure width is 6 inches (`fig.width = 6`) and `fig.asp = 0.7`, the figure height will be automatically calculated from `fig.width * fig.asp = 6 * 0.7 = 4.2`. Figure 戊-1 is an example using the chunk options `fig.asp = 0.7`, `fig.width = 6`, and `fig.align = 'center'`, generated from the code below:

```
1 par(mar = c(4, 4, .1, .1))
2 plot(pressure, pch = 19, type = 'b')
```

The actual size of a plot is determined by the chunk options `fig.width` and `fig.height` (the size of the plot generated from a graphical device), and we can specify the output size of plots via the chunk options `out.width` and `out.height`. The possible value of these two options depends on the output format of the document. For example, `out.width = '30%'` is a valid value for HTML output, but not for LaTeX/PDF output. However, **knitr** will automatically convert a percentage value for `out.width` of the form `x%` to `(x / 100) \linewidth`, e.g., `out.width = '70%'` will be treated as `.7\linewidth` when the output format is LaTeX. This makes it possible to specify a relative width of a plot in a consistent manner. Figure 戊-2 is an example of `out.width = 70%`.

```
1 par(mar = c(4, 4, .1, .1))
2 plot(cars, pch = 19)
```

<sup>1</sup>Do not forget the leading backslash! And also note the parentheses () after `ref`; they are not curly braces {}.

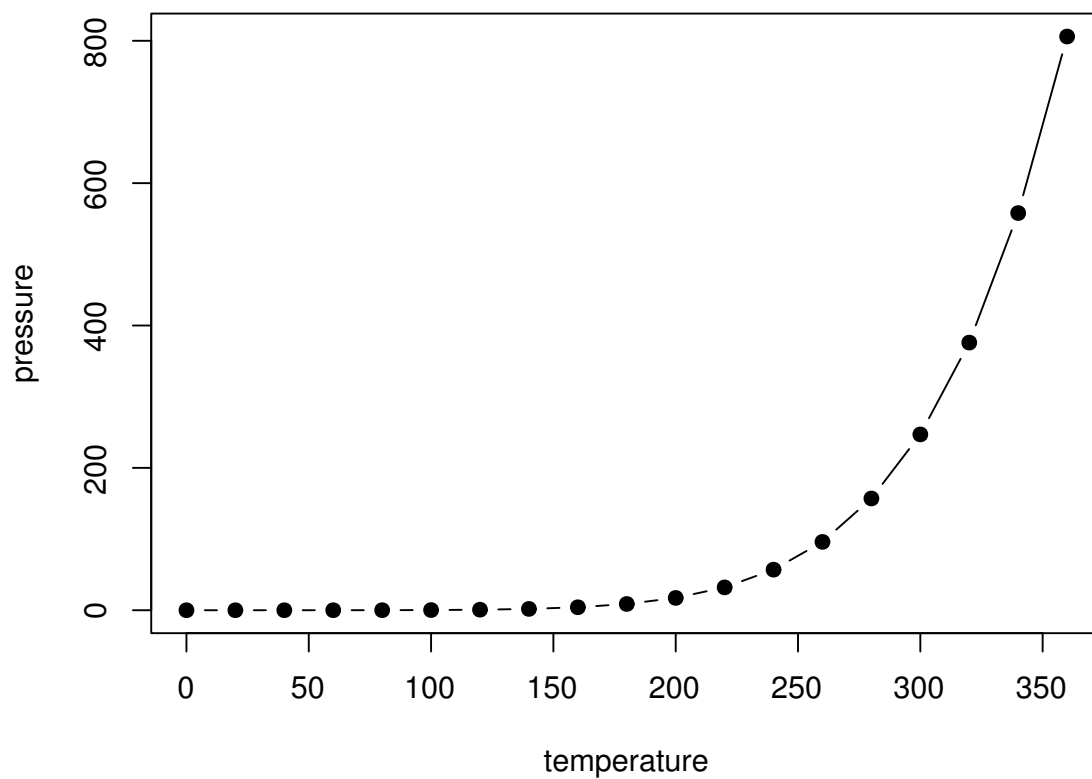


图 戊-1 A figure example with the specified aspect ratio, width, and alignment.

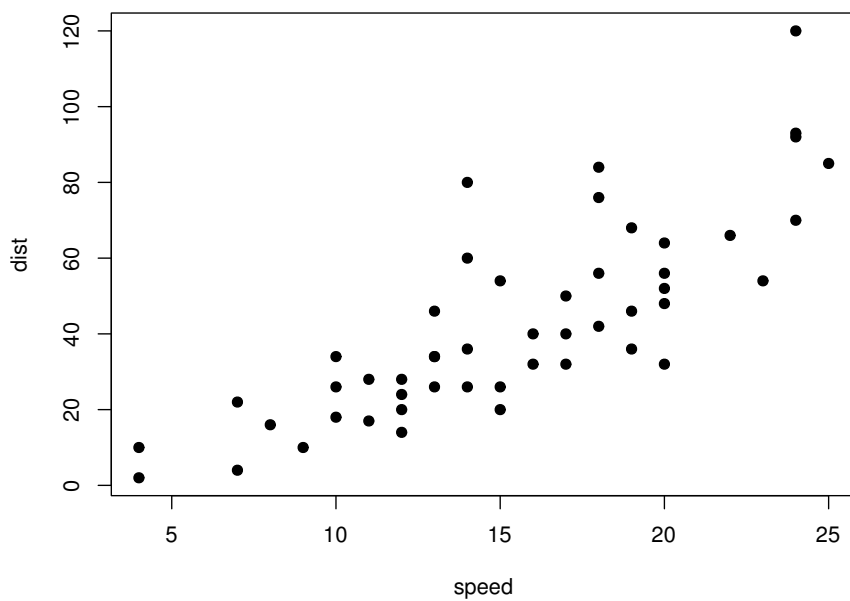


图 戊-2 A figure example with a relative width 70%.



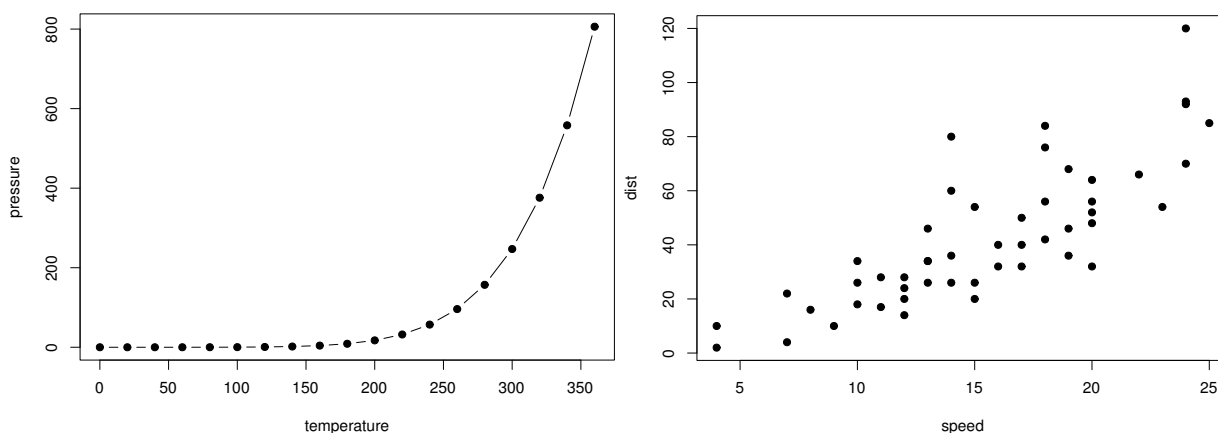


图 戊-3 Two plots placed side by side.

If you want to put multiple plots in one figure environment, you must use the chunk option `fig.show = 'hold'` to hold multiple plots from a code chunk and include them in one environment. You can also place plots side by side if the sum of the width of all plots is smaller than or equal to the current line width. For example, if two plots have the same width 50%, they will be placed side by side. Similarly, you can specify `out.width = '33%'` to arrange three plots on one line. Figure 戊-3 is an example of two plots, each with a width of 50%.

```
1 par(mar = c(4, 4, .1, .1))
2 plot(pressure, pch = 19, type = 'b')
3 plot(cars, pch = 19)
```

Sometimes you may have certain images that are not generated from R code, and you can include them in R Markdown via the function `knitr::include_graphics()`. Figure 戊-4 is an example of three **knitr** logos included in a figure environment. You may pass one or multiple image paths to the `include_graphics()` function, and all chunk options that apply to normal R plots also apply to these images, e.g., you can use `out.width = '33%'` to set the widths of these images in the output document.

```
1 knitr::include_graphics(rep('images/bookdown/knit-logo.png', 3))
```

There are a few advantages of using `include_graphics()`:

1. You do not need to worry about the document output format, e.g., when the output format is LaTeX, you may have to use the LaTeX command `\includegraphics{}` to include an image, and when the output format is Markdown, you have to use `![]()`. The

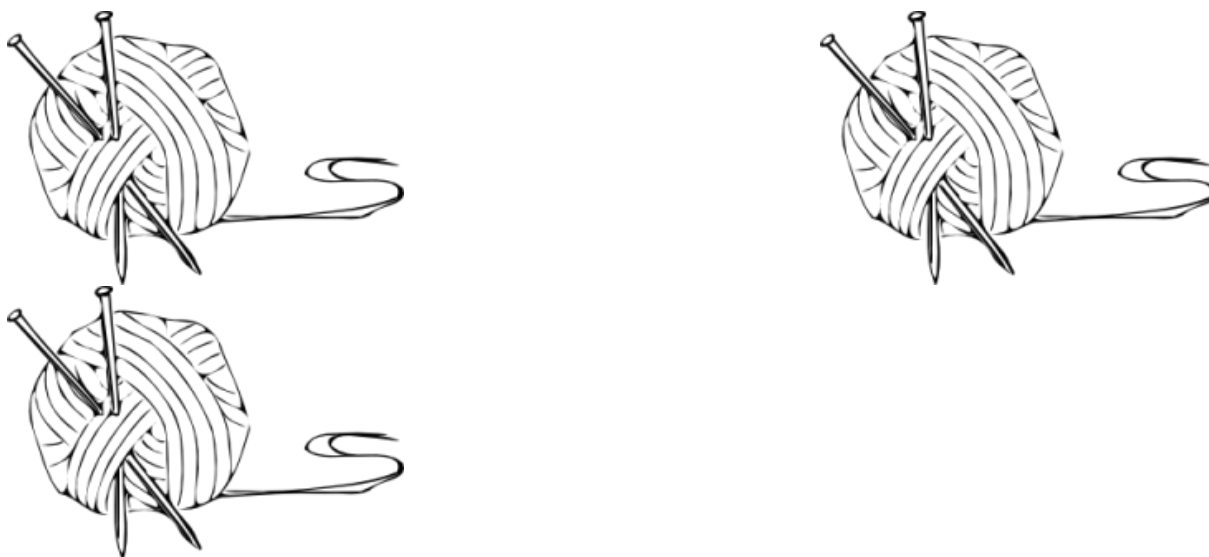


图 戊-4 Three knitr logos included in the document from an external PNG image file.

function `include_graphics()` in **knitr** takes care of these details automatically.

2. The syntax for controlling the image attributes is the same as when images are generated from R code, e.g., chunk options `fig.cap`, `out.width`, and `fig.show` still have the same meanings.
3. `include_graphics()` can be smart enough to use PDF graphics automatically when the output format is LaTeX and the PDF graphics files exist, e.g., an image path `foo/bar.png` can be automatically replaced with `foo/bar.pdf` if the latter exists. PDF images often have better qualities than raster images in LaTeX/PDF output. To make use of this feature, set the argument `auto_pdf = TRUE`, or set the global option `options(knitr.graphics.auto_pdf = TRUE)` to enable this feature globally in an R session.
4. You can easily scale these images proportionally using the same ratio. This can be done via the `dpi` argument (dots per inch), which takes the value from the chunk option `dpi` by default. If it is a numeric value and the chunk option `out.width` is not set, the output width of an image will be its actual width (in pixels) divided by `dpi`, and the unit will be inches. For example, for an image with the size 672 x 480, its output width will be 7 inches (7in) when `dpi = 96`. This feature requires the package **png** and/or **jpeg** to be installed. You can always override the automatic calculation of width in inches by providing a non-NULL value to the chunk option `out.width`, or use `include_graphics(dpi = NA)`.

表 戊-2 A table of the first 10 rows of the mtcars data.

	mpg	cyl	disp	hp	drat	wt	qsec	vs
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1

## 戊.5 Tables

For now, the most convenient way to generate a table is the function `knitr::kable()`, because there are some internal tricks in **knitr** to make it work with **bookdown** and users do not have to know anything about these implementation details. We will explain how to use other packages and functions later in this section.

Like figures, tables with captions will also be numbered and can be referenced. The `kable()` function will automatically generate a label for a table environment, which is the prefix `tab:` plus the chunk label. For example, the table label for a code chunk with the label `foo` will be `tab:foo`, and we can still use the syntax `\ref {label}` to reference the table. Table 戊-2 is a simple example.

```

1 knitr::kable(
2   head(mtcars[, 1:8], 10), booktabs = TRUE,
3   caption = 'A table of the first 10 rows of the mtcars data.'
4 )

```

If you want to put multiple tables in a single table environment, wrap the data objects (usually data frames in R) into a list. See Table 戊-3 for an example. Please note that this feature is only available in HTML and PDF output.

```

1 knitr::kable(

```

表 戊-3 A Tale of Two Tables.

Sepal.Length	Sepal.Width		mpg	cyl	disp
5.1	3.5	Mazda RX4	21.0	6	160
4.9	3.0	Mazda RX4 Wag	21.0	6	160
4.7	3.2	Datsun 710	22.8	4	108
		Hornet 4 Drive	21.4	6	258
		Hornet Sportabout	18.7	8	360

```

2 list(
3   head(iris[, 1:2], 3),
4   head(mtcars[, 1:3], 5)
5 ),
6 caption = 'A Tale of Two Tables.', booktabs = TRUE
7 )

```

When you do not want a table to float in PDF, you may use the LaTeX package **longtable**,<sup>1</sup> which can break a table across multiple pages. To use **longtable**, pass `longtable = TRUE` to `kable()`, and make sure to include `\usepackage{longtable}` in the LaTeX preamble (see Section ?? for how to customize the LaTeX preamble). Of course, this is irrelevant to HTML output, since tables in HTML do not need to float.

```

1 knitr::kable(
2   iris[1:55, ], longtable = TRUE, booktabs = TRUE,
3   caption = 'A table generated by the longtable package.'
4 )

```

表 戊-4 A table generated by the longtable package.

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa

<sup>1</sup><https://www.ctan.org/pkg/longtable>

---

5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa
5.8	4.0	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa
5.4	3.4	1.7	0.2	setosa
5.1	3.7	1.5	0.4	setosa
4.6	3.6	1.0	0.2	setosa
5.1	3.3	1.7	0.5	setosa
4.8	3.4	1.9	0.2	setosa
5.0	3.0	1.6	0.2	setosa
5.0	3.4	1.6	0.4	setosa
5.2	3.5	1.5	0.2	setosa
5.2	3.4	1.4	0.2	setosa
4.7	3.2	1.6	0.2	setosa
4.8	3.1	1.6	0.2	setosa
5.4	3.4	1.5	0.4	setosa
5.2	4.1	1.5	0.1	setosa
5.5	4.2	1.4	0.2	setosa
4.9	3.1	1.5	0.2	setosa
5.0	3.2	1.2	0.2	setosa

---

5.5	3.5	1.3	0.2	setosa
4.9	3.6	1.4	0.1	setosa
4.4	3.0	1.3	0.2	setosa
5.1	3.4	1.5	0.2	setosa
5.0	3.5	1.3	0.3	setosa
4.5	2.3	1.3	0.3	setosa
4.4	3.2	1.3	0.2	setosa
5.0	3.5	1.6	0.6	setosa
5.1	3.8	1.9	0.4	setosa
4.8	3.0	1.4	0.3	setosa
5.1	3.8	1.6	0.2	setosa
4.6	3.2	1.4	0.2	setosa
5.3	3.7	1.5	0.2	setosa
5.0	3.3	1.4	0.2	setosa
7.0	3.2	4.7	1.4	versicolor
6.4	3.2	4.5	1.5	versicolor
6.9	3.1	4.9	1.5	versicolor
5.5	2.3	4.0	1.3	versicolor
6.5	2.8	4.6	1.5	versicolor

---

Pandoc supports several types of Markdown tables,<sup>1</sup> such as simple tables, multiline tables, grid tables, and pipe tables. What `knitr::kable()` generates is a simple table like this:

```

1 Table: A simple table in Markdown.
2
3   Sepal.Length  Sepal.Width  Petal.Length  Petal.Width
4   -----
5           5.1           3.5           1.4           0.2
6           4.9           3.0           1.4           0.2
7           4.7           3.2           1.3           0.2
8           4.6           3.1           1.5           0.2
9           5.0           3.6           1.4           0.2
10          5.4           3.9           1.7           0.4

```

<sup>1</sup><http://pandoc.org/MANUAL.html#tables>

You can use any types of Markdown tables in your document. To be able to cross-reference a Markdown table, it must have a labeled caption of the form `Table: (\##label) Caption here`, where `label` must have the prefix `tab:`, e.g., `tab:simple-table`.

If you decide to use other R packages to generate tables, you have to make sure the label for the table environment appears in the beginning of the table caption in the form `(\##label)` (again, `label` must have the prefix `tab:`). You have to be very careful about the *portability* of the table generating function: it should work for both HTML and LaTeX output automatically, so it must consider the output format internally (check `knitr::opts_knit$get('rmarkdown.pandoc.to')`). When writing out an HTML table, the caption must be written in the `<caption></caption>` tag. For simple tables, `kable()` should suffice. If you have to create complicated tables (e.g., with certain cells spanning across multiple columns/rows), you will have to take the aforementioned issues into consideration.

## 戊.6 Cross-references

We have explained how cross-references work for equations (Section 戊.2.1), theorems (Section 戊.2.2), figures (Section 戊.4), and tables (Section 戊.5). In fact, you can also reference sections using the same syntax `\ref {label}`, where `label` is the section ID. By default, Pandoc will generate an ID for all section headers, e.g., a section `## Hello World` will have an ID `hello-world`. We recommend you to manually assign an ID to a section header to make sure you do not forget to update the reference label after you change the section header. To assign an ID to a section header, simply add `{##id}` to the end of the section header. Further attributes of section headers can be set using standard Pandoc syntax<sup>1</sup>.

When a referenced label cannot be found, you will see two question marks like `??`, as well as a warning message in the R console when rendering the book.

You can also create text-based links using explicit or automatic section IDs or even the actual section header text.

- If you are happy with the section header as the link text, use it inside a single set of square brackets:
  - `[Section header text]`: example “[A single document]” via `[A single document]`
- There are two ways to specify custom link text:
  - `[link text][Section header text]`, e.g., “[non-English books][Internationalization]” via `[non-English books][Internationalization]`

---

<sup>1</sup><http://pandoc.org/MANUAL.html#header-identifiers>

- `[link text](##ID)`, e.g., “Table stuff” via `[Table stuff](##tables)`

The Pandoc documentation provides more details on automatic section IDs<sup>1</sup> and implicit header references.<sup>2</sup>

Cross-references still work even when we refer to an item that is not on the current page of the PDF or HTML output. For example, see Equation (戊-1) and Figure 戊-4.

## 戊.7 Custom blocks

You can generate custom blocks using the `block` engine in **knitr**, i.e., the chunk option `engine = 'block'`, or the more compact syntax ```{block}`. This engine should be used in conjunction with the chunk option `type`, which takes a character string. When the `block` engine is used, it generates a `<div>` to wrap the chunk content if the output format is HTML, and a LaTeX environment if the output is LaTeX. The `type` option specifies the class of the `<div>` and the name of the LaTeX environment. For example, the HTML output of this chunk

```
1 ```{block, type='F00'}
2 Some text for this block.
3 ```
```

will be this:

```
1 <div class="F00">
2 Some text for this block.
3 </div>
```

and the LaTeX output will be this:

```
1 \begin{F00}
2 Some text for this block.
3 \end{F00}
```

It is up to the book author how to define the style of the block. You can define the style of the `<div>` in CSS and include it in the output via the `includes` option in the YAML metadata. Similarly, you may define the LaTeX environment via `\newenvironment` and include the definition in the LaTeX output via the `includes` option. For example, we may save the following style in a CSS file, say, `style.css`:

<sup>1</sup>[http://pandoc.org/MANUAL.html#extension-auto\\_identifiers](http://pandoc.org/MANUAL.html#extension-auto_identifiers)

<sup>2</sup>[http://pandoc.org/MANUAL.html#extension-implicit\\_header\\_references](http://pandoc.org/MANUAL.html#extension-implicit_header_references)



```
1 div.FOO {
2   font-weight: bold;
3   color: red;
4 }
```

And the YAML metadata of the R Markdown document can be:

```
1 ---
2 output:
3   bookdown::html_book:
4     includes:
5       in_header: style.css
6 ---
```

We have defined a few types of blocks for this book to show notes, tips, and warnings, etc. Below are some examples:



R is free software and comes with ABSOLUTELY NO WARRANTY. You are welcome to redistribute it under the terms of the GNU General Public License versions 2 or 3. For more information about these matters see <http://www.gnu.org/licenses/>.



R is free software and comes with ABSOLUTELY NO WARRANTY. You are welcome to redistribute it under the terms of the GNU General Public License versions 2 or 3. For more information about these matters see <http://www.gnu.org/licenses/>.



R is free software and comes with ABSOLUTELY NO WARRANTY. You are welcome to redistribute it under the terms of the GNU General Public License versions 2 or 3. For more information about these matters see <http://www.gnu.org/licenses/>.



R is free software and comes with ABSOLUTELY NO WARRANTY. You are welcome to redistribute it under the terms of the GNU General Public License versions 2 or 3. For more information about these matters see <http://www.gnu.org/licenses/>.



R is free software and comes with ABSOLUTELY NO WARRANTY. You are welcome to redistribute it under the terms of the GNU General Public License versions 2 or 3. For more information about these matters see <http://www.gnu.org/licenses/>.

The `knitr` block engine was designed to display simple content (typically a paragraph of plain text). You can use simple formatting syntax such as making certain words bold or italic, but more advanced syntax such as citations and cross-references will not work. However, there is an alternative engine named `block2` that supports arbitrary Markdown syntax, e.g.,

```

1 ```{block2, type='F00'}
2 Some text for this block [@citation-key].
3
4 - a list item
5 - another item
6
7 More text.
8 ```
```

The `block2` engine should also be faster than the `block` engine if you have a lot of custom blocks in the document, but its implementation was based on a hack,<sup>1</sup> so we are not 100% sure if it is always going to work in the future. We have not seen problems with Pandoc v1.17.2 yet.

One more caveat for the `block2` engine: if the last element in the block is not an ordinary paragraph, you must leave a blank line at the end, e.g.,

```

1 ```{block2, type='F00'}
2 Some text for this block [@citation-key].
3
4 - a list item
5 - another item
```

<sup>1</sup><https://github.com/jgm/pandoc/issues/2453>

```

6 - end the list with a blank line
7
8 ```

```

The theorem and proof environments in Section 戊.2.2 are actually implemented through the `block2` engine.

For all custom blocks based on the `block` or `block2` engine, there is one chunk option `echo` that you can use to show (`echo = TRUE`) or hide (`echo = FALSE`) the blocks.

## 戊.8 Citations

Although Pandoc supports multiple ways of writing citations, we recommend you to use BibTeX databases because they work best with LaTeX/PDF output. Pandoc can process other types of bibliography databases with the utility `pandoc-citeproc` (<https://github.com/jgm/pandoc-citeproc>), but it may not render certain bibliography items correctly (especially in case of multiple authors per item), and BibTeX can do a better job when the output format is LaTeX. With BibTeX databases, you will be able to define the bibliography style if it is required by a certain publisher or journal.

A BibTeX database is a plain-text file (with the conventional filename extension `.bib`) that consists of bibliography entries like this:

```

1 @Manual{R-base,
2   title = {R: A Language and Environment for Statistical
3     Computing},
4   author = {{R Core Team}},
5   organization = {R Foundation for Statistical Computing},
6   address = {Vienna, Austria},
7   year = {2016},
8   url = {https://www.R-project.org/},
9 }

```

A bibliography entry starts with `@type{`, where `type` may be `article`, `book`, `manual`, and so on.<sup>1</sup> Then there is a citation key, like `R-base` in the above example. To cite an entry, use `@key` or `[@key]` (the latter puts the citation in braces), e.g., `@R-base` is rendered as R Core Team [20], and `[@R-base]` generates “20”. If you are familiar with the `natbib` package in LaTeX, `@key` is basically `\citet{key}`, and `[@key]` is equivalent to `\citep{key}`.

<sup>1</sup>The type name is case-insensitive, so it does not matter if it is `manuaL`, `ManuaL`, or `MANUAL`.

There are a number of fields in a bibliography entry, such as `title`, `author`, and `year`, etc. You may see <https://en.wikipedia.org/wiki/BibTeX> for possible types of entries and fields in BibTeX.

There is a helper function `write_bib()` in **knitr** to generate BibTeX entries automatically for R packages. Note that it only generates one BibTeX entry for the package itself at the moment, whereas a package may contain multiple entries in the `CITATION` file, and some entries are about the publications related to the package. These entries are ignored by `write_bib()`.

```
1 # the second argument can be a .bib file
2 knitr::write_bib(c('knitr', 'stringr'), '', width = 60)

1 @Manual{R-knitr,
2   title = {knitr: A General-Purpose Package for Dynamic Report
3     Generation in R},
4   author = {Yihui Xie},
5   year = {2018},
6   note = {R package version 1.20},
7   url = {https://CRAN.R-project.org/package=knitr},
8 }
9 @Manual{R-stringr,
10  title = {stringr: Simple, Consistent Wrappers for Common
11    String Operations},
12  author = {Hadley Wickham},
13  year = {2018},
14  note = {R package version 1.3.1},
15  url = {https://CRAN.R-project.org/package=stringr},
16 }
```

Once you have one or multiple `.bib` files, you may use the field `bibliography` in the YAML metadata of your first R Markdown document (which is typically `index.Rmd`), and you can also specify the bibliography style via `biblio-style` (this only applies to PDF output), e.g.,

```
1 ---
2 bibliography: ["one.bib", "another.bib", "yet-another.bib"]
3 biblio-style: "apalike"
4 link-citations: true
5 ---
```

The field `link-citations` can be used to add internal links from the citation text of the author-year style to the bibliography entry in the HTML output.

When the output format is LaTeX, citations will be automatically put in a chapter or section. For non-LaTeX output, you can add an empty chapter as the last chapter of your book. For example, if your last chapter is the Rmd file `06-references.Rmd`, its content can be an inline R expression:

```
1 `r if (knitr::is_html_output()) '# References {-}'`
```

## 戊.9 Index

Currently the index is only supported for LaTeX/PDF output. To print an index after the book, you can use the LaTeX package **makeidx** in the preamble (see Section ??):

```
1 \usepackage{makeidx}
2 \makeindex
```

Then insert `\printindex` at the end of your book through the YAML option `includes -> after_body`. An index entry can be created via the `\index{}` command in the book body, e.g., `\index{GIT}`.

## 戊.10 HTML widgets

Although one of R's greatest strengths is data visualization, there are a large number of JavaScript libraries for much richer data visualization. These libraries can be used to build interactive applications that can easily render in web browsers, so users do not need to install any additional software packages to view the visualizations. One way to bring these JavaScript libraries into R is through the **htmlwidgets**<sup>1</sup> package<sup>R-htmlwidgets</sup>.

HTML widgets can be rendered as a standalone web page (like an R plot), or embedded in R Markdown documents and Shiny applications. They were originally designed for HTML output only, and they require the availability of JavaScript, so they will not work in non-HTML output formats, such as LaTeX/PDF. Before **knitr** v1.13, you will get an error when you render HTML widgets to an output format that is not HTML. Since **knitr** v1.13, HTML widgets will be rendered automatically as screenshots taken via the **webshot** package.<sup>R-webshot</sup> Of course, you need to install the **webshot** package. Additionally, you have to install PhantomJS

---

<sup>1</sup><http://htmlwidgets.org>

Show  entries Search:

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa

Showing 1 to 10 of 150 entries Previous  2 3 4 5 ... 15 Next

图 戊-5 A table widget rendered via the DT package.

(<http://phantomjs.org>), since it is what **webshot** uses to capture screenshots. Both **webshot** and PhantomJS can be installed automatically from R:

```
1 install.packages('webshot')
2 webshot::install_phantomjs()
```

The function `install_phantomjs()` works for Windows, OS X, and Linux. You may also choose to download and install PhantomJS by yourself, if you are familiar with modifying the system environment variable `PATH`.

When **knitr** detects an HTML widget object in a code chunk, it either renders the widget normally when the current output format is HTML, or saves the widget as an HTML page and calls **webshot** to capture the screen of the HTML page when the output format is not HTML. Here is an example of a table created from the **DT** package:<sup>R-DT</sup>

```
1 DT::datatable(iris)
```

If you are reading this book as web pages now, you should see an interactive table generated from the above code chunk, e.g., you may sort the columns and search in the table. If you are

reading a non-HTML version of this book, you should see a screenshot of the table. The screenshot may look a little different with the actual widget rendered in the web browser, due to the difference between a real web browser and PhantomJS's virtual browser.

There are a number of **knitr** chunk options related to screen-capturing. First, if you are not satisfied with the quality of the automatic screenshots, or want a screenshot of the widget of a particular state (e.g., after you click and sort a certain column of a table), you may capture the screen manually, and provide your own screenshot via the chunk option `screenshot.alt` (alternative screenshots). This option takes the paths of images. If you have multiple widgets in a chunk, you can provide a vector of image paths. When this option is present, **knitr** will no longer call **webshot** to take automatic screenshots.

Second, sometimes you may want to force **knitr** to use static screenshots instead of rendering the actual widgets even on HTML pages. In this case, you can set the chunk option `screenshot.force = TRUE`, and widgets will always be rendered as static images. Note that you can still choose to use automatic or custom screenshots.

Third, **webshot** has some options to control the automatic screenshots, and you may specify these options via the chunk option `screenshot.opts`, which takes a list like `list(delay = 2, cliprect = 'viewport')`. See the help page `?webshot::webshot` for the full list of possible options, and the package vignette<sup>1</sup> `vignette('intro', package = 'webshot')` illustrates the effect of these options. Here the `delay` option can be important for widgets that take long time to render: `delay` specifies the number of seconds to wait before PhantomJS takes the screenshot. If you see an incomplete screenshot, you may want to specify a longer delay (the default is 0.2 seconds).

Fourth, if you feel it is slow to capture the screenshots, or do not want to do it every time the code chunk is executed, you may use the chunk option `cache = TRUE` to cache the chunk. Caching works for both HTML and non-HTML output formats.

Screenshots behave like normal R plots in the sense that many chunk options related to figures also apply to screenshots, including `fig.width`, `fig.height`, `out.width`, `fig.cap`, and so on. So you can specify the size of screenshots in the output document, and assign figure captions to them as well. The image format of the automatic screenshots can be specified via the chunk option `dev`, and possible values are `pdf`, `png`, and `jpeg`. The default for PDF output is `pdf`, and it is `png` for other types of output. Note that `pdf` may not work as faithfully as `png`: sometimes there are certain elements on an HTML page that fail to render to the PDF screenshot, so you may

---

<sup>1</sup><https://cran.rstudio.com/web/packages/webshot/vignettes/intro.html>

want to use `dev = 'png'` even for PDF output. It depends on specific cases of HTML widgets, and you can try both `pdf` and `png` (or `jpeg`) before deciding which format is more desirable.

## 戊.11 Web pages and Shiny apps

Similar to HTML widgets, arbitrary web pages can be embedded in the book. You can use the function `knitr::include_url()` to include a web page through its URL. When the output format is HTML, an `iframe` is used;<sup>1</sup> in other cases, **knitr** tries to take a screenshot of the web page (or use the custom screenshot you provided). All chunk options are the same as those for HTML widgets. One option that may require your special attention is the `delay` option: HTML widgets are rendered locally, so usually they are fast to load for PhantomJS to take screenshots, but an arbitrary URL may take longer to load, so you may want to use a larger `delay` value, e.g., use the chunk option `screenshot.opts = list(delay = 5)`.

A related function is `knitr::include_app()`, which is very similar to `include_url()`, and it was designed for embedding Shiny apps via their URLs in the output. Its only difference with `include_url()` is that it automatically adds a query parameter `?showcase=0` to the URL, if no other query parameters are present in the URL, to disable the Shiny showcase mode, which is unlikely to be useful for screenshots or iframes. If you do want the showcase mode, use `include_url()` instead of `include_app()`. Below is a Shiny app example (Figure 戊-6):

```
1 knitr::include_app('https://yihui.shinyapps.io/miniUI/', height = '600  
   px')
```

Again, you will see a live app if you are reading an HTML version of this book, and a static screenshot if you are reading other types of formats. The above Shiny app was created using the **miniUI** package,<sup>R-miniUI</sup> which provides layout functions that are particularly nice for Shiny apps on small screens. If you use normal Shiny layout functions, you are likely to see vertical and/or horizontal scrollbars in the iframes because the page size is too big to fit an iframe. When the default width of the iframe is too small, you may use the chunk option `out.width` to change it. For the height of the iframe, use the `height` argument of `include_url()/include_app()`.

Shiny apps may take even longer to load than usual URLs. You may want to use a conservative value for the `delay` option, e.g., 10. Needless to say, `include_url()` and `include_app()` require a working Internet connection, unless you have previously cached the chunk (but web pages inside iframes still will not work without an Internet connection).

<sup>1</sup>An `iframe` is basically a box on one web page to embed another web page.



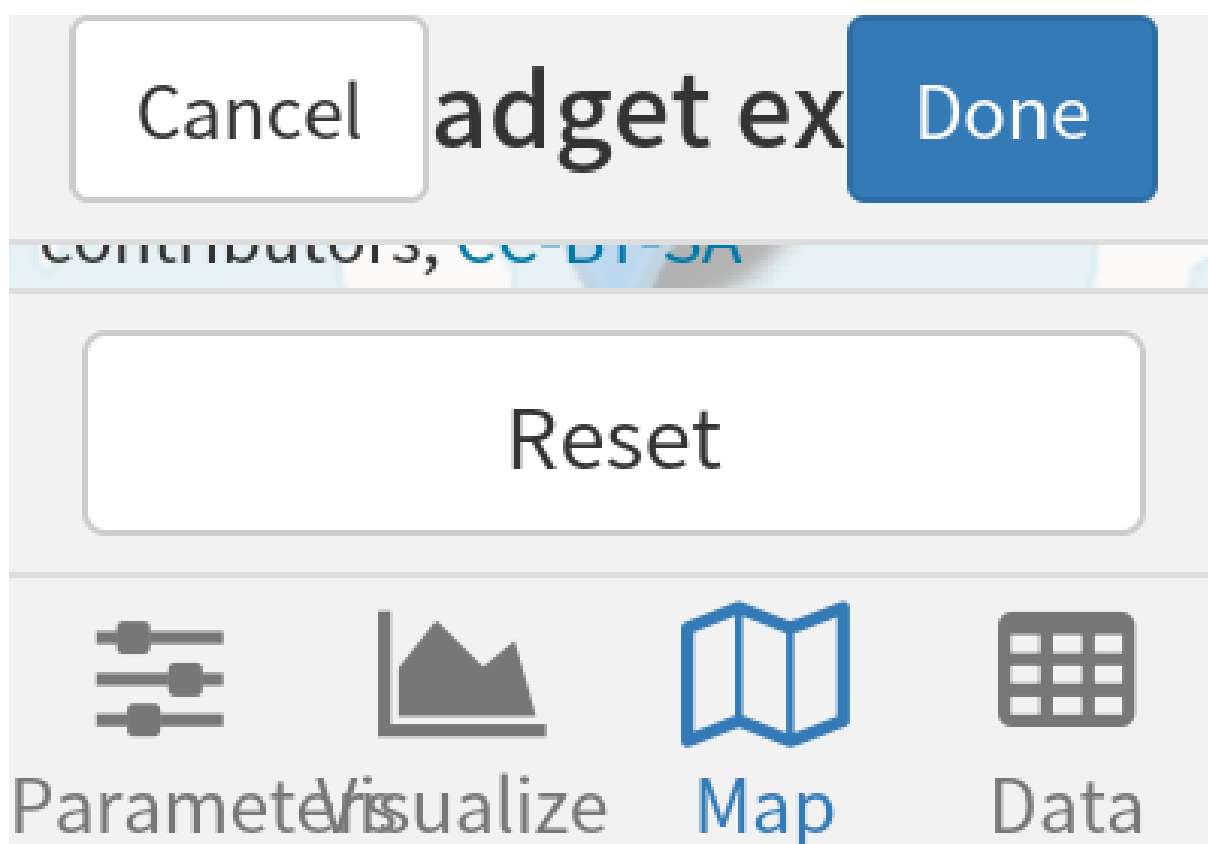


图 戊-6 A Shiny app created via the miniUI package; you can see a live version at <https://yihui.shinyapps.io/miniUI/>.



## 附录己 常见问题

### 己.1 如何修改rm命令，让删除的文件去回收站？

答. 文件被删且难以恢复，想必很多人都曾经遇到过。参考[这里](#)。具体做法如下。

```
$ sudo apt install trash-cli  
$ gedit ~/.bashrc
```

打开文件，在末尾加上，`alias rm='trash-put'`，保存退出。运行如下命令。

```
$ source .bashrc
```

重启终端即可。想用原生的 `rm`，则为 `\rm`。当然这样也有弊端，详细说明见[这里](#)。■

### 己.2 如何修改rm命令，让删除的文件去回收站？

答. 你为啥长得这么帅 ■



## 附录庚 大事记

- 2017-08-26 开工找模板。
- 2017-09-28 买到《Linux Bible》实体书中文版，可以带徒弟一起写本文了。
- 2017-10-26 第一次开课，给俩兄弟讲 Linux 下 C 语言基础。
- 2017-10-29 R 语言更新，貌似旧的安装方法已经无法在 deepin 下安装，需要调整，暂时没有调整。
- 2017-10-29 注册了优酷土豆账号，用于上传本书的相关视频。
- 2018-05-29 模板转移 SJTU 模板中
- 2018-07-02 第一版草稿成功出炉
- 2018-07-10 成功转移到新模板



## 索引

### A

appendix, 96

### B

BibTeX, 111

### C

citation, 111

code chunk, 97

cross-reference, 89, 92, 99, 103, 107

custom block, 108

### E

equation, 89

### F

figure, 98

floating environment, 98

### H

HTML widget, 113

### I

index, 113

inline R code, 97

### K

knitr::include\_graphics(), 101

### L

LaTeX math expression, 87

Linux, 3

longtable, 104

### M

Markdown, 85

### P

Pandoc, 85

part, 95

### S

shell, 39, 40

Shiny application, 116

深度操作系统, 7

深度科技, 7

### T

table, 103

theorem, 90

### Y

愚公移山, 57





## 参考文献

- [1] 何闻. 标准动态力发生装置国内外研究现状[J]. 机电工程, 1999(2): 47–49.
- [2] 崔万照, 马伟, 邱乐德, 等. 电磁超介质及其应用[M]. 北京: 国防工业出版社, 2008.
- [3] CHEN H, CHAN C T. Acoustic cloaking in three dimensions using acoustic metamaterials[J]. Applied Physics Letters, 2007, 91: 183518.
- [4] KIM S, WOO N, YEOM H Y, et al. Design and Implementation of Dynamic Process Management for Grid-enabled MPICH[C]// The 10th European PVM/MPI Users' Group Conference. Venice, Italy: [s.n.], 2003.
- [5] JOANNOPOULOS J D, JOHNSON S G, WINN J N. Photonic Crystals: Molding the Flow of Light[M]. [S.l.]: Princeton University Press, 2008.
- [6] 猪八戒. 论流体食物的持久保存[D]. 北京: 广寒宫大学, 2005.
- [7] 1363-2000 I S. IEEE Standard Specifications for Public-Key Cryptography[M]. New York: IEEE, 2000.
- [8] CHEN H, WU B I, ZHANG B, et al. Electromagnetic Wave Interactions with a Metamaterial Cloak[J]. Physical Review Letters, 2007, 99(6): 63903.
- [9] KOCHER C, JAFFE J, JUN B. Differential Power Analysis[C]// WIENER M. Advances in Cryptology (CRYPTO '99). Ed. by WIENER M. Vol. 1666. Lecture Notes in Computer Science. [S.l.]: Springer-Verlag, 1999: 388–397.
- [10] 王重阳, 黄药师, 欧阳峰, 等. 武林高手论文集[C]// 第  $N$  次华山论剑. 西安, 中国: 中国古籍出版社, 2006.
- [11] JEYAKUMAR A R. Metamori: A library for Incremental File Checkpointing[D]. Blacksburg: Virginia Tech, 2004.
- [12] 沙和尚. 论流沙河的综合治理[D]. 北京: 清华大学, 2005.
- [13] ZADOK E. FiST: A System for Stackable File System Code Generation[D]. USA: Computer Science Department, Columbia University, 2001.
- [14] 白云芬. 信用风险传染模型和信用衍生品的定价[D]. 上海: 上海交通大学, 2008.

- [15] WOO A, BAILEY D, YARROW M, et al. The NAS Parallel Benchmarks 2.0[R/OL]. The Pennsylvania State University CiteSeer Archives, 1995. <http://www.nasa.org/>.
- [16] 萧钰. 出版业信息化迈人快车道[J/OL]. 2001. <http://www.creader.com/news/20011219/200112190019.html>.
- [17] CHRISTINE M. Plant physiology: plant biology in the Genome Era[J/OL]. Science, 1998, 281: 331–332. <http://www.sciencemag.org/cgi/collection/anatmorp>.
- [18] R Core Team. R: A Language and Environment for Statistical Computing[M/OL]. Vienna, Austria: [s.n.], 2012. <http://www.R-project.org/>.
- [19] XIE Y. Dynamic Documents with R and knitr[M/OL]. 2nd. Boca Raton, Florida: Chapman, Hall/CRC, 2015. <http://yihui.name/knitr/>.
- [20] R Core Team. R: A Language and Environment for Statistical Computing[A/OL]. Vienna, Austria: R Foundation for Statistical Computing, 2017. <https://www.R-project.org/>.