

白起鹏的博客

白起鹏

2018 年 5 月 4 日

目录

第一部分 自然科学	2
第一章 地震学	3
1.1 韩国地震 CAP 机制解	3
1.2 地震时间处理	6
1.3 新西兰地震余震可视化	8
1.4 gCAPjoint 使用手册	10
1.5 如何对地震学问题进行误差分析	11
1.6 CAPtel 使用手册	11
第二章 计算机科学	12
第三章 物理学	13
第四章 数学	14
第二部分 其他	15
第五章 博客框架	16
5.1 你好	16
5.2 Hugo 安装配置	16
5.3 使用 MkDocs 撰写文档手册	19
5.4 使用 Overleaf 撰写 LaTeX 文档	20
第六章 Linux 操作系统	23

第一部分

自然科学

第一章 地震学

1.1 韩国地震 CAP 机制解

2016-09-12 在韩国庆州境内发生两次地震，本文给出 CAP 近震的机制解反演结果。

1.1.1 地震速报

此反演结果使用的数据来源于[IRIS M4.9](#) [IRIS M5.4](#)，地震基本信息来源于[USGS M4.9](#) [USGS M5.4](#)。

NEWS

【正式测定：韩国庆州 4.9 级和 5.4 级地震】USGS 正式测定：2016-09-12 在韩国庆州境内发生两次地震，M4.9 级震源深度 10.0km，M5.4 级震源深度 10.0km。

SCIENTIFIC Neic earthquake location information:

Magnitude: 5.4 Mb ± 0.0 Location: 35.769°N 129.227°E ± 4.9 km

Depth: 10.0 km ± 1.8

Origin Time: 2016-09-12 11:32:54.940 UTC

USGS moment tensors

1.1.2 CAPloc 机制解

M4.9

Crust1.0 CAPloc 反演结果:

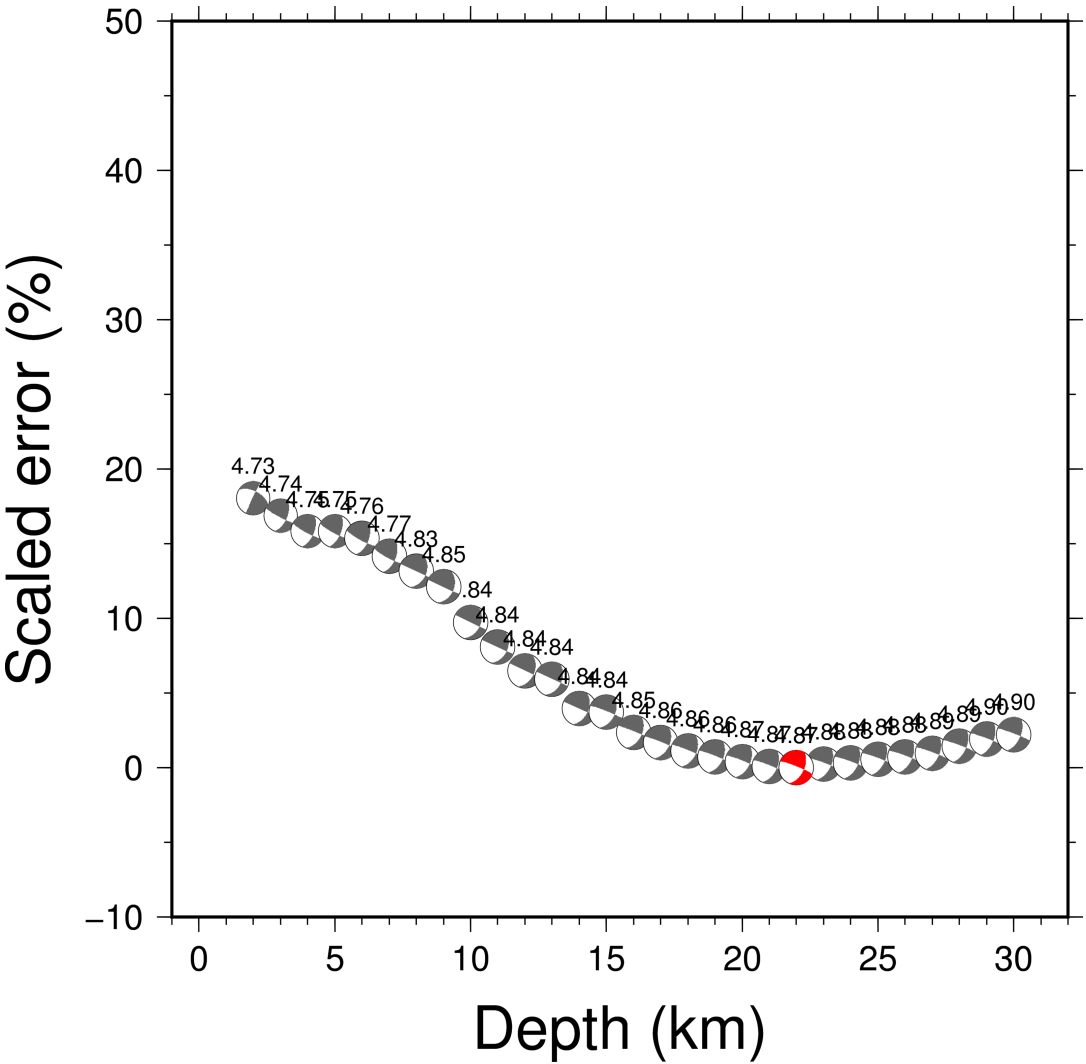
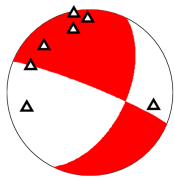
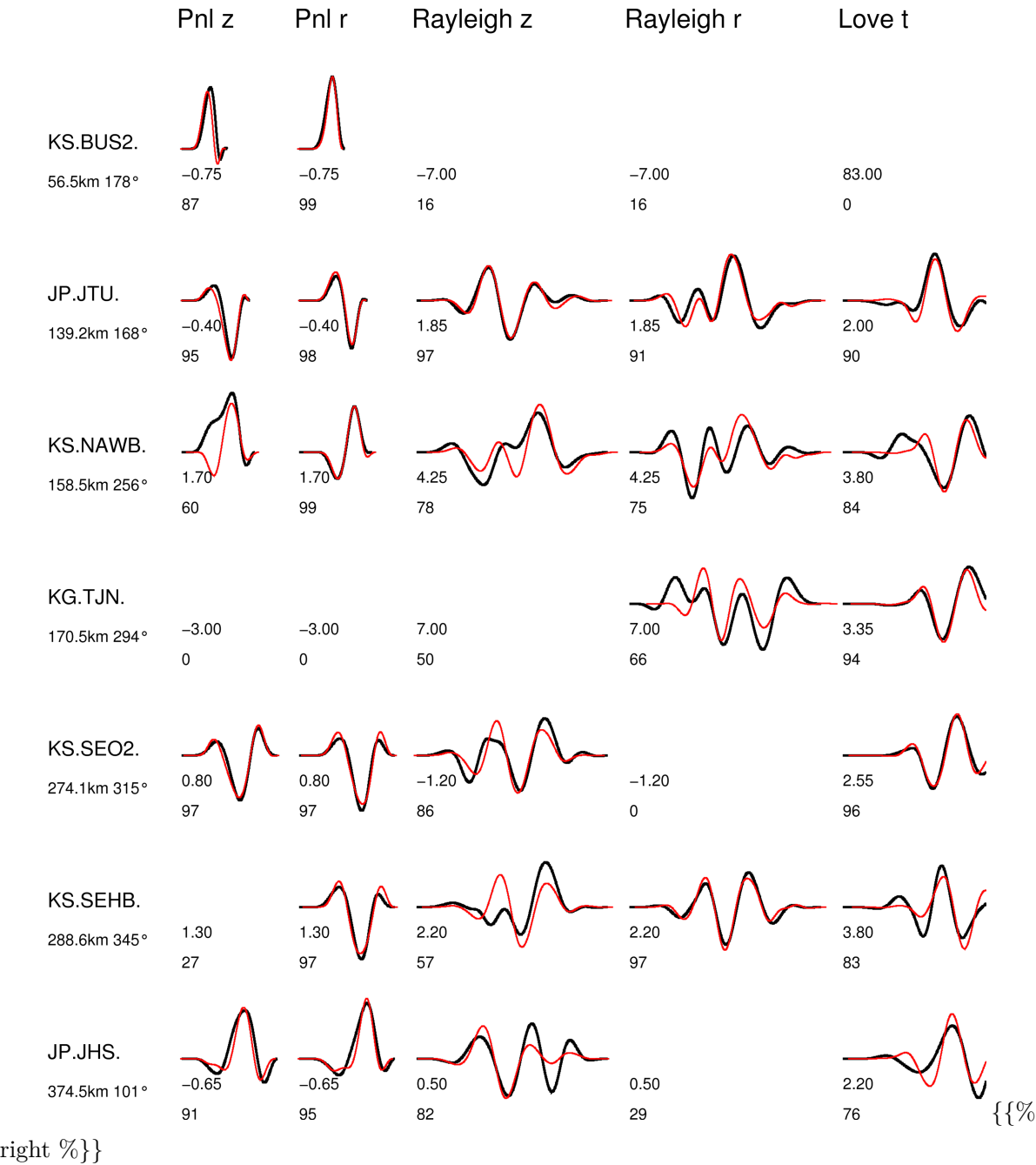


图 1.1: CAPloc_2016-09-12_M4.9_Sou 定: 韩国庆州 4.9 级和 5.4 级地震】USGS 正式测定: 2016-09-12 在韩国庆州境 thKorea_mecherr.png



Event: Mw4.9_SouthKoreadata Depth: 22km
FM 292 84 -36 Mw 4.87 Error 1.263e-02



(Kim et al., 1999) 文章中 model C 速度模型 CAPloc 反演结果:

Crust1.0 gCAPloc 反演结果:

1.1.3 M5.4

Crust1.0 CAPloc<img 全台站反演结果:

Crust1.0 CAPloc 挑选通道后反演结果:

Crust1.0 gCAPloc 全台站反演结果:

Crust1.0 gCAPloc 挑选通道反演结果:

© USTC 白起鹏 baiqp@mail.ustc.edu.cn

1.2 地震时间处理

通常在处理地震数据（尤其是 SAC 数据）时需要计算不同时间戳（Time Stamp），本文给出一些示例。

1.2.1 C 中的时间计算、转换

地震的时间处理，主要是对不同格式的转换，这里我先给出一个转换的 C 程序。

```
1 #include "stdio.h"
2 main()
3 {
4     int day,month,year,sum,leap;
5     scanf("%d-%d-%d",&year,&month,&day);
6     switch(month) /*先计算某月以前月份的总天数*/
7     {
8         case 1:sum=0;break;
9         case 2:sum=31;break;
10        case 3:sum=59;break;
11        case 4:sum=90;break;
12        case 5:sum=120;break;
13        case 6:sum=151;break;
14        case 7:sum=181;break;
15        case 8:sum=212;break;
16        case 9:sum=243;break;
17        case 10:sum=273;break;
18    printf("%d",sum);
19    }
20    printf("%d",sum);
21 }
```

一般来说，时间处理主要是进行以下几件工作：

- 时间格式转换
- 时间差计算

- 时间统计-程序运行

地震学中的问题主要是时间的格式转换和时差计算，

1.2.2 Shell 中的时间格式转换、时差计算

date 通常情况下可以分：年月日时分秒 一般情况下需要对 date 做四则运算。shell 中命令的空格要严格控制，等号两边无空格，有多余空格会出现错误，这中限制在日期 date 命令中就比较明显。

命令格式：

```
1 date [-u] [-d datestr] [-s datestr] [--utc] [--universal] [--date=datestr] [--set=
   datestr] [--help] [-version] [+FORMAT] [MMDDhhmm[[CC]YY][.ss]]
```

通常我们拿到的时间格式会有以下几种：

- CSV: %Y-%m-%dT%H:%M:%SZ 例如：2016-10-21T05:07:23.000Z
- SAC: %Y %j %H %M %S 例如：2016 295 05 07 23.00, 或者是 sac1st kzdate kztime f sacfile 得到 %Y/%m/%d %H:%M:%S 的时间格式
- 其他: 例如 2016年 10月 21日 星期五 05: 07: 23 CST等

date 显示指定格式的时间 date -d "STRING" "+FORMAT", 尽量加上双引号, 避免空格。由此我们可以这样计算, 取出时间处理成 %Y-%m-%d %H:%M:%S 格式, 然后转换成自 UTC 时间 1970-01-01 00:00:00 以来所有时间的秒数, 然后通过秒数进行四则运算, 最后再转换为想要的格式。这里暂且定义 %Y-%m-%d %H:%M:%S 为标准格式, 其他时间都以此为对比标准。

```
1 #!/bin/bash
2 catalog=query.csv
3 outfile=LocalTime
4 localtimezone=$1
5
6 gawk -F "," 'NR>=2 {print $1}' $catalog | sed 's/Z/ /g' | sed 's/T/ /g' >
   StandardTime
7 echo -e "\033[35m\tStandard time: \033[0m"
8 cat StandardTime
9 cat StandardTime | while read line
10 do
11     UTCtimestamp=`date -d "$line" +%s`
12     localtimestamp=`expr $UTCtimestamp + $localtimezone \* 60 \* 60 | bc` #
        oneday=86400(s)
13     echo `date -d @$localtimestamp +"%Y %j %H %M %S"` >> LocalTime
14 done
15 echo -e "\033[35m\tLocal time: \033[0m"
16 cat LocalTime
17 rm -rf StandardTime

1 #!/bin/bash
2 if [ $# -ne 2 ];
3 then
```



```

4 |         echo 'Usage: SAC2standTime.sh filepath timezone(hour)'
5 | else
6 | sacpath=$1
7 | localtimezone=$2 # 指定时区校正值,
8 | sac="*.U"
9 | cd $sacpath/
10 | line=`sac1st kzdate kztime f $sac | head -1 | gawk '{split($2,aa,"/");print aa[1]"
    |         -"aa[2]"-"aa[3],$3}'`
11 | localtimestamp=`date -d "$line" +%s`
12 | UTctimestamp=`expr $localtimestamp - $localtimezone \* 60 \* 60 | bc` # oneday
    |         =86400(s) onehour=3600(s)
13 | echo `date -d @$UTctimestamp +"%Y %j %H %M %S"`
14 | fi

```

通过上述两个 shell 脚本将标准时间和 SAC 中的时间进行互相转换, 有效!

1.2.3 Reference:

1.3 新西兰地震余震可视化

地震学中的可视化工作多种多样, 这里我们首先对新西兰地震余震震中随时间演化的可视化给出一个示例。

Python 具有十分丰富的扩展库来进行可视化工作, 例如 Matplotlib, Mayavi, VTK 等等, 这里我们主要关注的是地震学的数据可视化。

1.3.1 Aftershock

The smaller aftershocks following a mainshock have a characteristic distribution in size and time. As Fig. 4.5-9, most aftershock occur on or near the mainshock's fault plane, so their location are used to distinguish between the fault and auxiliary planes and to estimate the fault area.

– Seth Stein, An Introduction to Seismology Earthquake and Earth Structure

作为例子, 我们先选取 **2016-11-20-M7.8 新西兰地震** 作为主震, 从 USGS earthquake feed 下载 csv 格式的余震目录数据, 这里使用了 Python3 的 urllib 包作为获取目录的模块, 原始的 csv 文件是后发生地震在前, 为了画出随主震后时间演化的地震, 我们将数据从后往前取值。

```

1 | import urllib
2 | import numpy as np
3 | import matplotlib
4 | matplotlib.rcParams['toolbar'] = 'None'
5 | import matplotlib.pyplot as plt
6 | from mpl_toolkits.basemap import Basemap
7 | #from matplotlib.animation import FuncAnimation
8 | import matplotlib.animation as animation
9 |

```

```

10 |
11 | # Open the earthquake data
12 | # -----
13 | # -> http://earthquake.usgs.gov/earthquakes/feed/v1.0/csv.php
14 | feed = "http://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/"
15 |
16 | # Significant earthquakes in the past 30 days
17 | # url = urllib.request.urlopen(feed + "significant_month.csv")
18 |
19 | # Earthquakes of magnitude > 4.5 in the past 30 days
20 | # url = urllib.request.urlopen(feed + "4.5_month.csv")
21 | url = urllib.request.urlopen(feed + "4.5_week.csv")
22 |
23 | # Earthquakes of magnitude > 2.5 in the past 30 days
24 | # url = urllib.request.urlopen(feed + "2.5_month.csv")
25 | # url = urllib.request.urlopen(feed + "2.5_week.csv")
26 |
27 | # Earthquakes of magnitude > 1.0 in the past 30 days
28 | # url = urllib.request.urlopen(feed + "1.0_month.csv")
29 |
30 | # Set earthquake data
31 | data = url.read()
32 | data = data.split(b'\n')[1:-1]
33 | E = np.zeros(len(data), dtype=[('position', float, 2),
34 |                                ('magnitude', float, 1)])
35 | for i in range(len(data)):
36 |     row = data[i].split(b',')
37 |     E['position'][i] = float(row[2]), float(row[1])
38 |     E['magnitude'][i] = float(row[4])
39 |
40 |
41 | fig = plt.figure()
42 | ax = plt.subplot(1,1,1)
43 | P = np.zeros(50, dtype=[('position', float, 2),
44 |                          ('size', float, 1),
45 |                          ('growth', float, 1),
46 |                          ('color', float, 4)])
47 |
48 | # Basemap projection
49 | map = Basemap(projection='mill')
50 | map.drawcoastlines(color='0.40', linewidth=0.25)
51 | map.fillcontinents(color='0.60')
52 | scat = ax.scatter(P['position'][:,0], P['position'][:,1], P['size'], lw=0.5,
53 |                  edgecolors = P['color'], facecolors='None', zorder=10)
54 |
55 |
56 | def update(frame):
57 |     current = frame % len(E)

```

```

58 | i = frame % len(P)
59 |
60 | P['color'][:,3] = np.maximum(0, P['color'][:,3] - 1.0/len(P))
61 | P['size'] += P['growth']
62 |
63 | magnitude = E['magnitude'][current]
64 | P['position'][i] = map(*E['position'][current])
65 | P['size'][i] = 5
66 | P['growth'][i] = np.exp(magnitude) * 0.1
67 |
68 | if magnitude < 6:
69 |     P['color'][i] = 0,0,1,1
70 | else:
71 |     P['color'][i] = 1,0,0,1
72 | scat.set_edgecolors(P['color'])
73 | scat.set_facecolors(P['color']*(1,1,1,0.25))
74 | scat.set_sizes(P['size'])
75 | scat.set_offsets(P['position'])
76 |
77 | plt.title("Earthquakes > 4.5 in the last 30 days")
78 | anim = animation.FuncAnimation(fig, update, interval=100)
79 | anim.save('line.gif', dpi=100, writer='imagemagick')

```

演示效果 (pdf 版无法查看 gif 图片):

1.4 gCAPjoint 使用手册

如果每个文档字数超过 3000 字, 则单独出来作为独立内容。优化文章结构。

1.4.1 CAP 方法原理

1.4.2 CAP 方法软件安装汇总

1.4.3 依赖软件

1.4.4 CAP 衍生及改进方法介绍

1.4.5 gCAPjoint 方法原理及使用

1.4.6 地震算例简报汇总

以下为例 ## 2018-01-23T11:22:00 M6.5 帕米尔地震

地震速报简图 [USGS], 或中国地震局目录 确定关注参数, 写地震报告。报告流程: Markdown 工作网站框架计划 (数量最多); Word 文档详细记录总结 (数量可以少一些, 准确性系统性要多一些); Latex 格式化正式发布 (精品)。所有文档案例都要有一个 Markdown 版本作为网站的索引筛选。

相关文章分析简报

1.5 如何对地震学问题进行误差分析

1.5.1 地震学震源参数反演时参数误差怎么处理、控制？

地震学领域的参数误差控制有多重多样的方式，例如试错法、重采样法等等，这里将参数误差处理控制方法分类如下：

1. 试错法参数误差控制；
2. 重采样法参数误差控制；

参考书目整理如下：

1. 《误差分析导论·物理测量中的不确定度》英文原名《An Introduction to Error Analysis: The Study of Uncertainties in Physical Measurements》第二版；

1.6 CAPtel 使用手册

1.6.1 数据下载

在 [IRIS](#) 官网下载远震数据，震中距 30° – 90° ，以 seed 格式放置在个人目录

1.6.2 反演简单操作步骤

将计算断层辅助面的脚本放置在用户 bin 执行目录，根据实际情况执行如下命令 (XXX 为实际目录位置，注意修改)：

```
1 | $ mv cap_source/etc_commands/to_conjugate.awk ~/soft/bin/
```

进入个人目录，根据具体情况执行如下命令

```
1 | $ mv 2017-7-17_6.2_Russia.562133.seed tel.seed
2 | $ sh LeadData.sh tel.seed
3 | $ sh LeadCAP.cmd
4 | $ cd data
5 | $ cp ../etc_commands/mecherr.cmd .
6 | $ sh mecherr.cmd
7 | $ gs mecherr.ps
8 | $ gs vmodel_20.ps
```

1.6.3 结果分析

结果分析请参考相应的论文和文章。

第二章 计算机科学

第三章 物理学

第四章 数学

第二部分

其他

第五章 博客框架

5.1 你好

欢迎来到 **白起鹏** 的个人博客。这时一个测试博客哦:smile:

本博客始于 2016 年，2018 年测试发布，主要关注地震学、计算机科学（深度学习）、心理学，其他学科等的知识分享。

5.2 Hugo 安装配置

Hugo 是由 Go 语言实现的静态网站生成器。简单易用、高效、易扩展、快速部署。

5.2.1 Hugo 安装配置

具体可参考官方[中文参考文档](#)

直接安装步骤如下：

- 从[Hugo Releases](#)下载对应操作系统的Hugo二进制文件，放在/bin/目录下运行
- 安装依赖工具：Git, Mercurial, Go(1.3+)
- 安装 [Visual Studio Code](#)，方便修改代码

注意 Go 的高版本编译需要 Go 1.4 作为初始编译器，因此要先安装后再安装高版本，可参考 [Go 高版本安装](#)。

让 go get 显示进度

在使用 Go 下载 github 的包比较大时，需要让屏幕显示进度，可以通过修改 Go 源码来实现。打开 /usr/local/go/src/cmd/go/vcs.go 文件，如下修改：

```
1 // vcsGit describes how to use Git.
2 var vcsGit = &vcsCmd{
3     name: "Git",
4     cmd: "git",
5
6     createCmd: "clone {repo} {dir}", // 此处修改为 clone --progress {repo} {dir}
        update --init --recursive"},
```

```

1 | var buf bytes.Buffer
2 |     cmd.Stdout = &buf
3 |     cmd.Stderr = &buf
4 |     cmd.Stdout = os.Stdout // 重定向标准输出
5 |     cmd.Stderr = os.Stderr // 重定向标准输出
6 |     err = cmd.Run()

```

然后运行 `src/all.bash` 重新编译 Go，等编译完成后使用 `go get` 可以看到进度条。

安装 pandoc

使用二进制包安装，具体流程如下：

```

1 | $ wget -c https://github.com/jgm/pandoc/releases/download/2.1.3/pandoc-2.1.3-linux
   |     .tar.gz
2 | $ tar -zxvf pandoc-2.1.3-linux.tar.gz
3 | $ su
4 | # mv pandoc-2.1.3 /usr/local/
5 | $ vim ~/.bashrc
6 | At the end of the file, add "export PATH=/usr/local/pandoc-2.1.3/bin:${PATH}"
7 | $ source ~/.bashrc

```

5.2.2 主题安装

参考 [even](#)

- 修改 `/themes/even/src/css/` 目录下的版本文件，调整了网页的样式。
- 修改 `/themes/even/src/fonts/` 目录下的字体文件及 `/themes/even/src/css/_custom/` 的 `_custom.scss` 文件个性化定制页面。

注意修改 `/themes/even/src/` 内任一文件时，要再次编译，编译命令如下：

```

1 | $ cd src/
2 | $ npm install
3 | $ npm run build

```

5.2.3 字体修改

字体安装

字体转换工具可使用 [Everthing Fonts](#)

Abode Source Code Pro

```

1 | $ sudo yum install adobe-source-code-pro-fonts
2 | $ cd /usr/share/fonts/adobe-source-code-pro

```

字体修改

字体修改可参考 [CSS3@font-face](#)，修改 `/themes/even/src/css/_custom/_custom.scss` 的字体变量列表，具体不再列出。

执行 `npm install`，`npm run build` 会报错 `Unexpected character '' (1:0)`，可通过如下办法解决：

```
1 | $ vim /themes/even_m/src/webpack.config.js
2 | // 添加如下语句
3 | {
4 |   test: /OpenSans-BoldItalic\.woff|woff2|eot|ttf|otf|svg)/,
5 |   use: 'file-loader?name=[path][name].[ext]'
```

5.2.4 网站图标修改

修改网站的图表，可修改 `themes/even/static/` 文件夹下的几个图片。也可在根目录 `static/` 中放置此类文件进行覆盖，优先级高于主题模版文件。

- `android-chrome-192x192.png`
- `android-chrome-512x512.png`
- `apple-touch-icon.png`
- `browserconfig.xml`
- `favicon.ico`
- `favicon-16x16.png`
- `favicon-32x32.png`
- `manifest.json`
- `mstile-150x150.png`
- `safari-pinned-tab.svg`

可通过 [favicon generator](#) 生成图片

5.2.5 Hugo 站点说明

[参考博客](#)

- `archetypes`：存放 `default.md`，头文件格式
- `content`：`content` 目录存放博客文章（`.markdown/.md` 文件）
- `data`：存放自定义模版，导入的 `toml` 文件（或 `json`，`yaml`）
- `layouts`：`layouts` 目录存放的是网站的模板文件
- `static`：`static` 目录存放 `js/css/img` 等静态资源
- `config.toml`：`config.toml` 是网站的配置文件

5.2.6 建站基本技能

需要掌握了解多种语言及程序，包括如下：

- Hugo
- Markdown
- TOML
- Git
- Html
- Shell
- Pandoc
- Python
- Latex
- CSS
- JavaScripts
- YAML

依赖的账户等：

- GitHub
- Google Analysts
- GoDaddy

5.3 使用 MkDocs 撰写文档手册

申请到顶级域名后，可以将二级域名和 GitHub Pages 的项目仓库进行绑定，做到分仓库管理网站。这样可以在不修改主站的情况下对不同工作进行分类汇总。

5.3.1 DNS 二级域名设置

在 GoDaddy 域名管理中修改 DNS 如下：

类型	名称	值	TTL
A	@	192.30.252.153	1 小时
CNAME	www	@	1 小时
CNAME	blog	@	1 小时
...			

5.3.2 绑定 GitHub Pages

在二级域名设置好后，在 GitHub 页面上传建成的网站，这里可以使用三种方式放置页面，`master`、`gh-pages`、`master/docs`，推荐使用第二种方式。

项目仓库根目录下放置一 `CNAME` 文件，里面写二级域名，例如 `blog.baiqp.info`，然后在仓库设置页面调整好 GitHub Pages 的设置。等待一定时间提醒配置好后进入页面。

这里列出常用的几个二级链接：

1. [blog](#)

- 2. [music](#)
- 3. [wiki](#)

项目撰写注意事项

因为这里使用的 GitHub Pages 进行发布，需要符合标准，请仔细阅读[错误帮助手册](#)，完整的 GitHub 帮助手册请参考 [Help github](#)。格式有错误会导致发布报错。

- Markdown 语法参考[\[1\]](#)
- 图片引用的名称和 `images/` 下的命名保持一致；
- 表格的格式要注意，具体格式如下：使用 Markdown 画的表格，如下表（冒号表示对齐）

col. 1	col. 2	col. 3
col 3 is	right-aligned	\$1600
col 2 is	centered	\$12
zebra stripes	are neat	\$1

或

col. 1	col. 2	col. 3
col 3 is	right-aligned	\$1600
col 2 is	centered	\$12
zebra stripes	are neat	\$1

- 在上传到 GitHub Pages 项目 `gh-pages` 时，可以先清空远程分支，再上传（效果还要验证）；

5.3.3 参考资料

- 1. [Even Markdown 帮助](#)

5.4 使用 Overleaf 撰写 LaTeX 文档

[Overleaf](#) 是一个网页版 LaTeX 编辑工具，里面有大量的模版可以使用，极大的方便了科研工作。

这里通过学习使用相关的内容，有助于我们的科研工作。

5.4.1 Overleaf 使用介绍

在官网上注册账号后，可以在主界面新建一个工程，选择不同的模板，包括：基本 (Basics)、学术杂志 (Academic Journal)、文献目录 (Bibliography)、书籍 (Book)、日历 (Calendar)、正式书信 (Formal Letter)、作业任务 (Homework Assignment)、时事通讯 (Newsletter)、海

报 (Poster)、展示 (Presentation)、工程/实验室报告 (Project/Lab Report)、履历/简历 (Resume/CV)、论文 (Thesis)。直接在新建的工程中修改 Tex 代码或者文本模式增加内容即可在立即查看输出的 PDF 文档。越来越多的人开始使用 Overleaf 协助写作，十分推荐 ~

5.4.2 LaTeX 基本介绍

LaTeX 是一个文档准备系统 (Document Preparing System)，它非常适用于生成高印刷质量的科技类和教学类文档。也能够生成所有其他种类的文档，小到简单的信件，大到完整的书籍。

– 《lshort-zh-cn》2017-03

使用 LaTeX 撰写文档，需要熟悉其语法，这里可以给出如下参考书目：

- 《LaTeX 入门》刘海洋
- 《一份不太简短的 LaTeX2 介绍》或称为《102 分钟了解 LaTeX2e》，英文名《lshort-zh-cn》

一篇短文的框架：

- 导言区 (preamble)
- 正文：论文标题、目录、章节、参考文献

编写文章的开始首先一部是熟悉模板文档的结构，然后针对每一部分分块修改。

5.4.3 模板

LaTeX 语法内容丰富，实际中可以选择一个模板，了解基本的框架后进行修改即可。这里我们以几个常用的模板例子作为介绍说明。其中包括 USTC 研究生模板、Overleaf 模板、个人工作报告模板等三个方面来整理。

USTC 模板

使用中科大 LaTeX 模板撰写硕士、博士论文，可以参考如下链接 [ustctug](#)，可以下载一份作为参考

Overleaf 模板

Overleaf 的模板分类多样，可以选择符合任务需求的开源模板进行必要修改。并将源码和输出文档一同放置。

个人工作报告模板

整理一份简洁的 LaTeX 模板用于日常写作，（暂时推迟）

书籍翻译模板

为了翻译地震学数据，这里需要确定一个简单可靠的模板进行日常分章节翻译，然后再整理在一起。

5.4.4 Q&A

5.4.5 文献调研报告

文献调研报告 draft: false 作为综述性报告，需要较为大量的阅读（多篇核心文献 + 辅助文献）构成，难度要大一些，可以适当选择主题多次编写修改。

5.4.6 文章阅读报告

在阅读一篇较为重要的文章后，可以通过写文章阅读报告来做整理，便于对文章的研究背景及意义、逻辑框架、优点（亮点）缺点、重要结论做出总结。训练学术水平，锻炼写作能力。

文章阅读报告以 Word 版为整理初稿，然后进行格式化。第一阶段分为两种类型：经典文章阅读报告和新文章阅读报告，经典文章阅读报告通过字句段的不断精读做到准确掌握，新文章阅读通过了解框架基本内容整理简介。

具体格式化规定如下：

- 文章大体分类（通过分类，标签区分）；
- 框架：研究意义、研究背景及现状、关注问题、数据、研究方法、亮点结果、结论、讨论及不足等；

第六章 Linux 操作系统