


正则表达式简介

杨宇昌 

中国科学院植物研究所
系统与进化植物学国家重点实验室

2019 年 11 月 8 日 ver 1.1

使用许可协议

本文档依 CC BY-NC-SA 4.0 协议¹授权



只要遵守以下条件，即可自由地共享和演绎本作品：

- 署名 (BY)
- 非商业性使用 (NC)
- 相同方式共享 (SA)

本文档的源代码见

https://github.com/Mikumikunisiteageru/RegEx_Lecture

¹<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.zh>

目录

绪论

正则表达式 中文编码问题 不可见字符

语法

转义
集合、量词与分组
位置与或运算

例题

日期格式化
动物的分类法
学名中的字体形状

目录

绪论

正则表达式 中文编码问题 不可见字符

转义
集合、量词与分组
位置与或运算

日期格式化
动物的分类法
学名中的字体形状

进阶查找与替换：正则表达式

正则表达式 / Regular Expression：按照规则匹配模式

推荐软件：Notepad++¹，使用的正则表达式流派为 PCRE²

本幻灯片记号约定

- 这样的文字 或 `such text` 表示实际的文本
- 这样的文字 或 `such_text` 表示正则表达式
- 查找目标Source ⇒ 替换为Target 表示一次替换操作

¹<https://notepad-plus-plus.org/>

²http://docs.notepad-plus-plus.org/index.php/Regular_Expressions 5/36

中文编码：ANSI 与 UTF-8

在“格式”菜单中，可以检查当前文本文件的编码

- ANSI 是坏编码，不能抵抗字节丢失，容错性低
TGCATGCCTGCA... → GCATGCCTGCA...

C7 D2 B3 D6 C3 CE B1 CA CA E9 C6 E6 BE B0

且持梦笔书奇景 → 页置伪适楮婢?

D2 B3 D6 C3 CE B1 CA CA E9 C6 E6 BE B0

- UTF-8 是好编码，有分组机制，局部错误不会扩散
TGC ATG CCT GCA... → GC ATG CCT GCA...

E6 97 A5 E7 A0 B4 E4 BA 91 E6 B3 A2 E4 B8 87 E9 87 8C E7 BA A2

日破云波万里红 → ??破云波万里红

97 A5 E7 A0 B4 E4 BA 91 E6 B3 A2 E4 B8 87 E9 87 8C E7 BA A2

执行查找或替换操作前，必须确认编码方式为 UTF-8

观察不可见字符

按第七组第二个按钮 ¶¹ 可以看到平时不可见的字符

- 空格，按 Spacebar 键输入，记作 `␣`，显示为 `·`
`space␣between␣words` ⇔ `space·between·words`
- 全角空格，在中文输入法全角状态下按 Spacebar 键输入
- 制表符，按 Tab 键输入，记作 `\t`，显示为 `→`
`→` 随位置不同可长可短，但都同样是一个制表符
- 换行符，按 Enter 键输入，记作 `\r\n`，显示为 `CR LF`
Windows 风格的换行是 `\r\n`，其他操作系统的是 `\n`

用正则表达式替换时，最好打开 ¶ 开关，以便观察效果

¹或“视图”——“显示符号”——“显示所有字符”

制表符的作用

制表符常用于与电子表格软件¹联合作业

电子表格复制到纯文本环境时，列用 `\t` 分隔，即 TSV 格式

中文名	学名
南蝠	<i>Ia io</i> Thomas
大蟾蜍	<i>Bufo bufo</i> L.
喜鹊	<i>Pica pica</i> L.
翻车鱼	<i>Mola mola</i> L.
玉蜀黍	<i>Zea mays</i> L.
早熟禾	<i>Poa annua</i> L.

⇒

- 1 中文名 → 学名
- 2 南蝠 → Ia_io_Thomas
- 3 大蟾蜍 → Bufo_bufo_L.
- 4 喜鹊 → Pica_pica_L.
- 5 翻车鱼 → Mola_mola_L.
- 6 玉蜀黍 → Zea_mays_L.
- 7 早熟禾 → Poa_annua_L.

¹如 Microsoft Excel

制表符的作用

制表符常用于与电子表格软件¹联合作业

TSV 复制到电子表格时, `\t` 分隔列, 不带字体风格

中文名	学名
南蝠	Ia io Thomas
大蟾蜍	Bufo bufo L.
喜鹊	Pica pica L.
翻车鱼	Mola mola L.
玉蜀黍	Zea mays L.
早熟禾	Poa annua L.

⇐

- 1 中文名 → 学名
- 2 南蝠 → Ia_io_Thomas
- 3 大蟾蜍 → Bufo_bufo_L.
- 4 喜鹊 → Pica_pica_L.
- 5 翻车鱼 → Mola_mola_L.
- 6 玉蜀黍 → Zea_mays_L.
- 7 早熟禾 → Poa_annua_L.

¹如 Microsoft Excel

目录

正则表达式
中文编码问题
不可见字符

语法

转义
集合、量词与分组
位置与或运算

日期格式化
动物的分类法
学名中的字体形状

特殊字符的转义

正则表达式中，以下 14 个字符是特殊的，不能直接表示

* . ? + \$ ^ [] () { } | \

要表示这些字符，需要在前面加上反斜线 \

例：* 表示 *，\. 表示 .，\\ 表示 \

特殊字符在正则表达式中承担语法功能，如 . 匹配任何字符

例：3.14 可匹配 3.14、3014、3d14 或 3点14

3\.14 只可匹配 3.14

集合与通配符

- `[...]` 正集合，匹配其中任何字符
- `[^...]` 负集合，匹配其外任何字符，慎用
- `.` 匹配任意字符（除了 `\r` 与 `\n`¹）
- `\d` 匹配阿拉伯数字，等价于 `[0-9]`
- `\l` 匹配小写罗马字母，等价于 `[a-z]`
- `\u` 匹配大写罗马字母，等价于 `[A-Z]`
- `[\x{3400}-\x{9FFF}]` 匹配中日韩汉字

例：`[一二三]` 球悬铃木 可匹配 一球悬铃木 或 三球悬铃木
`201\d` 可匹配 2010 或 2019

¹关闭 “`.` 匹配新行” 的选项即可如此

量词

量词表示其前面的模式重复的次数，不可单独使用

- $\{a, b\}$ 至少 a 次，至多 b 次¹
- $\{a, \}$ 至少 a 次，无上限
- $\{a\}$ 恰好 a 次，等价于 $\{a, a\}$
- $+$ 至少 1 次，无上限，等价于 $\{1, \}$
- $*$ 至少 0 次，无上限，等价于 $\{0, \}$
- $?$ 至少 0 次，至多 1 次，等价于 $\{0, 1\}$

例: nat?ive 可匹配 naive 或 native
 em+ 可匹配 em 或 emmmmmmm ，但不可匹配 e
 $\backslash\text{d}\{4\}$ 可匹配 2019 ，也可匹配 6666

¹ a 与 b 均为非负整数，且 $a \leq b$ ，下同

量词的贪婪性

量词默认是贪婪的，即匹配尽可能长的字符串

例：字符串 20190831 中，
`\d{4,8}` 倾向于匹配 20190831 而不是 2019

要抑制贪婪，需要在量词后添加 `?`

例：字符串 20190831 中，
`\d{4,8}?` 倾向于匹配 2019 而不是 20190831

注意：`*?` 和 `??` 都是合法的，但没得必要

分组与捕捉

圆括号 (…) 可以把模式分组，形成更长的模式

例：`(\d+.){2}` 可以匹配 1年365天 或 10下10下

分组按照开始的顺序，分别自动命名为 `\1`、`\2`、`\3` 等

例：`(.)(.)\1\2` 可匹配 开心开心，不可匹配 开开心心

`(.)\1(.)\2` 可匹配 开开心心，不可匹配 开心开心

分组在替换中仍然有效，可以 `\1`、`\2` 等方式引用

例：要把碱基序列 TGCATGCCTGCA 每三个字母用空格分隔，
作替换 `(\u{3})` \Rightarrow `\1` 即可得到 TGC ATG CCT GCA

锚点

锚点匹配位置，没有宽度

- `^` 行开始处 (`\r\n` 之后或文档开始处)
- `$` 行结束处 (`\r\n` 之前或文档结束处)
- `\<` 单词¹开始处
- `\>` 单词结束处
- `\b` 单词边界处

例：The car is scarlet 中，
`car` 有两处匹配，`\<car\>` 或 `\bcar\b` 只匹配前一处

¹ 中文无词式书写习惯，不适用于 Scintilla 准则，故无单词一说

例：去除空行

Source

1	Non-empty
2	
3	Non-empty
4	
5	Non-empty
6	
7	
8	Non-empty

$\sim\backslash r\backslash n \Rightarrow \emptyset^1$
或
 $(\backslash r\backslash n)\{2,\} \Rightarrow \backslash 1$

Target

1	Non-empty
2	Non-empty
3	Non-empty
4	Non-empty

¹替换为空白，相当于删除匹配到的模式，后同

例：去除前导零

Source			Target	
1	000076962	$\sim 0^+ \Rightarrow \emptyset$	1	76962
2	00165087		2	165087
3	00247904		3	247904
4	00386584		4	386584
5	000844259		5	844259
6	00034441		6	34441
7	00034449		7	34449
8	1217872		8	1217872

例：添加前导零

	Source
1	76962
2	165087
3	247904
4	386584
5	844259
6	34441
7	34449
8	1217872

1. $\text{^\d{1,7}}\$ \Rightarrow 0\backslash 1$
2. 重复 1 至匹配不到

	Target
1	00076962
2	00165087
3	00247904
4	00386584
5	00844259
6	00034441
7	00034449
8	01217872

零宽断言

零宽断言匹配位置，也没有长度

- $(?= \dots)$ 后面会有该模式
- $(?! \dots)$ 后面不会有该模式
- $(?<= \dots)$ 前面有了该模式
- $(?<! \dots)$ 前面没有该模式

例: $(?<=\backslash d)km\backslash >$ \Rightarrow $\sqcup km$ 可以把 $3km$ 更正为 $3\ km$

注意：用负断言 $(?! \dots)$ 或 $(?<! \dots)$ 时要小心

例：用下划线分隔馆代号与标本号

	Source
1	BM000076962
2	G00165087
3	NY00247904
4	E00386584
5	K000844259
6	PE00034441
7	PE00034449
8	KUN1217872

$(?<=\backslash u)(?=\backslash d) \Rightarrow _$
或
 $(\backslash u)(\backslash d) \Rightarrow \backslash 1_2$

	Target
1	BM_000076962
2	G_00165087
3	NY_00247904
4	E_00386584
5	K_000844259
6	PE_00034441
7	PE_00034449
8	KUN_1217872

大小写转换

Notepad++ 提供了“匹配大小写”选项，平时建议开启

例：关闭该选项时，`\<mL\>` \Rightarrow `mL` 可将 `m1`、`ML` 等统一为 `mL`

强制罗马字母大小写

- `\u` 其后一个字母强制大写
- `\U... \E` 其间全部字母强制大写
- `\l` 其后一个字母强制小写
- `\L... \E` 其间全部字母强制小写

例：`\<(\l)` \Rightarrow `\u\l` 可将所有单词首字母大写，用于书名

或运算

...|... 可匹配前面的或后面的模式，其优先级低于连接运算

例：a|(truck|lorry) 可匹配 a truck 或 a lorry

a|truck|lorry 可匹配 a truck 或 lorry

正则表达式中的运算优先级

转义 > 分组、零宽断言与集合 > 量词 > 锚点与连接 > 或

目录

绪论

正则表达式
中文编码问题
不可见字符

语法

转义
集合、量词与分组
位置与或运算

例题

日期格式化
动物的分类法
学名中的字体形状

日期格式化: yyyy.m.d \rightarrow yyyy.mm.dd

Source

1	1559.2.21
2	1592.11.28
3	1638.3.15
4	1654.5.4
5	1678.12.13
6	1711.11.25
7	1760.11.13
8	1782.9.16
9	1831.7.17
10	1856.4.27
11	1871.8.14
12	1906.2.7

1. $\backslash. \Rightarrow \backslash.0$

2. $\backslash.0?(\backslash d \backslash d) \Rightarrow \backslash.\backslash 1$

或

$(?<=\backslash.)(\backslash d)(?=[\backslash.\backslash r]) \Rightarrow 0\backslash 1$

Target

1	1559.02.21
2	1592.11.28
3	1638.03.15
4	1654.05.04
5	1678.12.13
6	1711.11.25
7	1760.11.13
8	1782.09.16
9	1831.07.17
10	1856.04.27
11	1871.08.14
12	1906.02.07

日期格式化: yyyy.m.d \rightarrow yyyymmdd

	Source
1	1559.2.21
2	1592.11.28
3	1638.3.15
4	1654.5.4
5	1678.12.13
6	1711.11.25
7	1760.11.13
8	1782.9.16
9	1831.7.17
10	1856.4.27
11	1871.8.14
12	1906.2.7

1. $\backslash. \Rightarrow \backslash.0$
2. $\backslash.0?(\backslash d \backslash d) \Rightarrow \backslash 1$

	Target
1	15590221
2	15921128
3	16380315
4	16540504
5	16781213
6	17111125
7	17601113
8	17820916
9	18310717
10	18560427
11	18710814
12	19060207

日期格式化: `yyyymmdd` \rightarrow `yyyy/m/d`

	Source
1	15590221
2	15921128
3	16380315
4	16540504
5	16781213
6	17111125
7	17601113
8	17820916
9	18310717
10	18560427
11	18710814
12	19060207

1. $(\backslash d\{4\})(\backslash d\backslash d)(\backslash d\backslash d) \Rightarrow \backslash 1/\backslash 2/\backslash 3$
2. $/0 \Rightarrow /$

	Target
1	1559/2/21
2	1592/11/28
3	1638/3/15
4	1654/5/4
5	1678/12/13
6	1711/11/25
7	1760/11/13
8	1782/9/16
9	1831/7/17
10	1856/4/27
11	1871/8/14
12	1906/2/7

把动物按脚的数目分类

Source

- 1 人有两只脚
- 2 海星有五只脚
- 3 狗有四只脚
- 4 猫有四只脚
- 5 章鱼有八只脚
- 6 蚊子有六只脚
- 7 蛤蟆有四只脚
- 8 蜗牛有一只脚
- 9 蝴蝶有六只脚
- 10 蟹有八只脚
- 11 鸡有两只脚

Target

- 1 一只脚的有蜗牛
- 2 两只脚的有人、鸡
- 3 五只脚的有海星
- 4 八只脚的有章鱼、蟹
- 5 六只脚的有蚊子、蝴蝶
- 6 四只脚的有狗、猫、蛤蟆

1. $\text{~}(\text{.}+)\text{有}(\text{.})\text{只脚}\$ \Rightarrow \backslash 2_ \backslash 1$
2. 将行按升序排列¹
3. $\text{~}(\text{.})_(\text{.}+)\backslash \text{r}\backslash \text{n}\backslash 1_ \Rightarrow \backslash 1_ \backslash 2、$
4. 重复 3 至匹配不到
5. $\text{~}(\text{.})_(\text{.}+)\$ \Rightarrow \backslash 1\text{只脚的有}\backslash 2$

¹Notepad++ 中可 “编辑” —— “行操作” —— “升序排列文本行”

练习：把动物按运动方式分类

Source

- 1 猪会游、走
- 2 猫会走
- 3 草鱼会游
- 4 飞鱼会游、飞
- 5 鸡会飞、走
- 6 鸭会飞、走、游

?

Target

- 1 会游的有猪、草鱼、飞鱼、鸭
- 2 会走的有猪、猫、鸡、鸭
- 3 会飞的有飞鱼、鸡、鸭

标记语言：另一种排版方式

电子文档软件¹用鼠标或快捷键选择字体形状

标记语言利用纯文本标记指示字体形状，更容易控制

例：Some *italicized* words

- HTML: Some `<i>italicized</i>` words
- Markdown: Some `*italicized*` words
- T_EX: Some `{\itshape italicized\}` words
- L^AT_EX: Some `\textit{italicized}` words

学名 *Allium hookeri* var. *muliense* Airy Shaw 用 Markdown 表示

`*Allium hookeri* var. *muliense* Airy Shaw`

字体形状由位置决定，可以由确定的规则描述，故可以自动化

¹如 Microsoft Word

利用正则表达式控制字体形状

```
1 Allium fistulosum L.  
2 Allium caput-medusae Airy Shaw  
3 Allium hookeri var. muliense Airy Shaw  
4 Allium senescens subsp. glaucum (Regel) Dostál  
5 Allium ×proliferum (Moench) Schrad. ex Willd.  
6 Allium giganteum 'Ambassador'  
7 Allium 'Gladiator'
```

利用正则表达式控制字体形状

```
1 *Allium fistulosum* L.  
2 *Allium caput-medusae* Airy Shaw  
3 *Allium hookeri* var. muliense Airy Shaw  
4 *Allium senescens* subsp. glaucum (Regel) Dostál  
5 Allium ×proliferum (Moench) Schrad. ex Willd.  
6 *Allium giganteum* 'Ambassador'  
7 Allium 'Gladiator'
```

1. $\sim(\backslash u\backslash l+)_\square([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash l_\square\backslash 2\backslash*_\square$

利用正则表达式控制字体形状

```
1 *Allium fistulosum* L.  
2 *Allium caput-medusae* Airy Shaw  
3 *Allium hookeri* var. *muliense* Airy Shaw  
4 *Allium senescens* subsp. glaucum (Regel) Dostál  
5 Allium ×proliferum (Moench) Schrad. ex Willd.  
6 *Allium giganteum* 'Ambassador'  
7 Allium 'Gladiator'
```

1. $\sim(\backslash u\backslash l+)_\square([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1_\square\backslash 2\backslash*_\square$

2. $(?<=_\square var\backslash._\square)([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1\backslash*_\square$

利用正则表达式控制字体形状

```
1 *Allium fistulosum* L.  
2 *Allium caput-medusae* Airy Shaw  
3 *Allium hookeri* var. *muliense* Airy Shaw  
4 *Allium senescens* subsp. *glaucum* (Regel) Dostál  
5 Allium ×proliferum (Moench) Schrad. ex Willd.  
6 *Allium giganteum* 'Ambassador'  
7 Allium 'Gladiator'
```

1. $\sim(\backslash u\backslash l+)_\square([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1_\square\backslash 2\backslash*_\square$

2. $(?<=_\square var\backslash._\square)([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1\backslash*_\square$

3. $(?<=_\square subsp\backslash._\square)([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1\backslash*_\square$

利用正则表达式控制字体形状

```
1 *Allium fistulosum* L.  
2 *Allium caput-medusae* Airy Shaw  
3 *Allium hookeri* var. *muliense* Airy Shaw  
4 *Allium senescens* subsp. *glaucum* (Regel) Dostál  
5 *Allium* ×proliferum (Moench) Schrad. ex Willd.  
6 *Allium giganteum* 'Ambassador'  
7 *Allium* 'Gladiator'
```

1. $\sim(\backslash u\backslash l+)_\square([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1_\square\backslash 2\backslash*_\square$
2. $(?<=_\square\text{var}\backslash._\square)([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1\backslash*_\square$
3. $(?<=_\square\text{subsp}\backslash._\square)([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1\backslash*_\square$
4. $\sim(\backslash u\backslash l+)_\square \Rightarrow _\square\backslash*\backslash 1\backslash*_\square$

利用正则表达式控制字体形状

```
1 *Allium fistulosum* L.  
2 *Allium caput-medusae* Airy Shaw  
3 *Allium hookeri* var. *muliense* Airy Shaw  
4 *Allium senescens* subsp. *glaucum* (Regel) Dostál  
5 *Allium* ×*proliferum* (Moench) Schrad. ex Willd.  
6 *Allium giganteum* 'Ambassador'  
7 *Allium* 'Gladiator'
```

1. $\sim(\backslash u\backslash l+)_\square([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1_\square\backslash 2\backslash*_\square$
2. $(?<=_\square\text{var}\backslash._\square)([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1\backslash*_\square$
3. $(?<=_\square\text{subsp}\backslash._\square)([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1\backslash*_\square$
4. $\sim(\backslash u\backslash l+)_\square \Rightarrow _\square\backslash*\backslash 1\backslash*_\square$
5. $(?<= \times)([\backslash l-]+)_\square \Rightarrow _\square\backslash*\backslash 1\backslash*_\square$

如何应用于电子文档?

1. 用 Markdown 解释器处理成带字体风格的文本
2. 将带字体风格的文本粘贴到电子文档中

在线 Markdown 解释器: <https://dillinger.io/>

The screenshot shows the Dillinger online Markdown editor interface. At the top, there's a dark header with the 'DILLINGER' logo, 'SAVE TO' and 'IMPORT FROM' buttons, and a settings gear icon. Below the header, the document name is 'Untitled Document.md', with 'WORDS: 30' and 'CHARACTERS: 233' on the right. The main area is split into two panes: 'MARKDOWN' on the left and 'PREVIEW' on the right. The 'MARKDOWN' pane contains a list of botanical names with asterisks for italics. The 'PREVIEW' pane shows the rendered result where the names are in italics. A small icon of a document with an arrow is visible between the two panes.

MARKDOWN	PREVIEW
1 <i>*Allium fistulosum* L.</i>	<i>Allium fistulosum L.</i>
2 <i>*Allium caput-medusae* Airy Shaw</i>	<i>Allium caput-medusae Airy Shaw</i>
3 <i>*Allium hookeri* var. <i>*muliense*</i> Airy Shaw</i>	<i>Allium hookeri var. muliense Airy Shaw</i>
4 <i>*Allium senescens* subsp. <i>*glaucum*</i> (Regel) Dostál</i>	<i>Allium senescens subsp. glaucum (Regel) Dostál</i>
5 <i>*Allium* x<i>*proliferum*</i> (Moench) Schrad. ex Willd.</i>	<i>Allium xproliferum (Moench) Schrad. ex Willd.</i>
6 <i>*Allium giganteum* 'Ambassador'</i>	<i>Allium giganteum 'Ambassador'</i>
7 <i>*Allium* 'Gladiator'</i>	<i>Allium 'Gladiator'</i>
8	<i>Allium 'Gladiator'</i>