MATLAB之正则表达式

Notebook: nanzt's notebook
Created: 3/31/2014 3:11 AM **Updated:** 3/31/2014 4:15 AM
Tags: computer

MATLAB之正则表达式
(giscn@msn.com)

拿到一批MTS实验数据，每个文件结构如下

MTS793|MPT|ENU|1|2|.|/|:|1|0|0|A
(第2行空行)
Operator Information Time: 3.0378418 Sec 5/12/2013 12:35:40 PM
Sample Name lanzhouhuangtu
Sample 13ZHT61
Height 125.6 mm
Test Name 13ZHT61
Diameter 62 mm
Density 1.62 g/mm^3
Water 23.97 %
Temperature -1 deg_C
Loading Rate
Operator Information End
(第14行是空行)
Data Acquisition Preload Data Time: 34.053955 Sec 5/12/2013 12:36:11 PM
Time Axial Displacement Axial Force Confining Displacement Confining Pressure
Sec mm kN mm MPa
4.0551758 5.364418e-05 -0.29039565 68.696938 0.050356787
...
(空行)
Data Acquisition Hold Data Time: 5154.0703 Sec 5/12/2013 2:01:31 PM
Time Axial Displacement Axial Force Confining Displacement Confining Pressure
Sec mm kN mm MPa
39.066162 -0.0011855364 -0.97346318 79.855125 0.31741175
...
(空行)
Data Acquisition Time: 7264.0635 Sec 5/12/2013 2:36:41 PM
Time Axial Displacement Axial Force Confining Displacement Confining Pressure
Sec mm kN mm MPa
7235.084 -0.00048279762 -0.88194877 80.901726 0.30021426
...
(空行)
(重复多个Data Acquisition头、数据及空行的组合)
Cyclic Acquisition Time: 28867.717 Sec 5/12/2013 8:36:45 PM
Stored at: 5 cycle Stored for: 1 segments

Points: 25
Time Axial Displacement Axial Force Confining Displacement Confining Pressure
Sec mm kN mm MPa
28866.082 -2.9164114 -1.8504251 79.36274 0.29931739
...
(空行)
(重复多个Cyclic Acquisition头、数据及空行的组合)
(文件末)

其中Data Acquisition部分是静荷载实验数据，Cyclic Acquisition是动荷载数据。我们要将静荷载和动荷载两部分分解开，输出到两个单独的文件。无论静、动荷载输出文件，都包括前14行的信息头。静荷载部分的Preload和Hold Data两部分（包括头信息和数据）不需要输出，其余Data Acquistion部分输出，但头信息只保留第一个即可。

即需要处理成两个文件：a_static.dat （静荷载）和a_dynamic.dat（动荷载），其中a_static.dat结构如下：

MTS793|MPT|ENU|1|2|.|/|:|1|0|0|A
(第2行空行)
Operator Information Time: 3.0378418 Sec 5/12/2013 12:35:40 PM
Sample Name lanzhouhuangtu
Sample 13ZHT61
Height 125.6 mm
Test Name 13ZHT61
Diameter 62 mm
Density 1.62 g/mm^3
Water 23.97 %
Temperature -1 deg_C
Loading Rate
Operator Information End
(第14行是空行)
Data Acquisition Time: 7264.0635 Sec 5/12/2013 2:36:41 PM
Time Axial Displacement Axial Force Confining Displacement Confining Pressure
Sec mm kN mm MPa
7235.084 -0.00048279762 -0.88194877 80.901726 0.30021426
...
(空行)
(重复多个Data Acquisition头、数据及空行的组合)
(文件末)

其中a_dynamic.dat结构如下：

MTS793|MPT|ENU|1|2|.|/|:|1|0|0|A
(第2行空行)
Operator Information Time: 3.0378418 Sec 5/12/2013 12:35:40 PM
Sample Name lanzhouhuangtu
Sample 13ZHT61
Height 125.6 mm
Test Name 13ZHT61

Diameter 62 mm
Density 1.62 g/mm^3
Water 23.97 %
Temperature -1 deg_C
Loading Rate
Operator Information End
(第14行是空行)
Cyclic Acquisition Time: 28867.717 Sec 5/12/2013 8:36:45 PM
Stored at: 5 cycle Stored for: 1 segments
Points: 25
Time Axial Displacement Axial Force Confining Displacement Confining Pressure
Sec mm kN mm MPa
28866.082 -2.9164114 -1.8504251 79.36274 0.29931739
...
(空行)
(重复多个Cyclic Acquisition头、数据及空行的组合)
(文件末)

这种文本文件很大，比如我测试的样本文件，一共1281889行，约61.7MB，其余文件有超过120M的。对于这种大文件，手工处理十分费劲，必须要写代码处理。

因此流程很简单：

1. 找到第一个Cyclic Acquisition，将此前的全部行，放到比如字符串s1；
2. 从s1中移去preload data部分（包括头信息和数据），得到s2；
3. 从s2中移去hold data 剖分（包括头信息和数据），得到s3；
4. 从s3中移去多余的Data Acquisition头信息，即只保留第一个Data Acquisition头信息，得到s4；
5. 将s4输出到a_static.dat。
6. 得到1-14行头信息；
7. 得到第一个Cyclic Acquisition及以后的全部行
8. 合并6、7得到的字符串
9. 输出到a_dynamic.dat。

所以这里的关键是如何得到我们需要的字符串，或者将某些字符串移去。一个很好的办法是使用正则表达式(regular expression），MATLAB里的函数是regexp。另外有regexprep是用于将匹配上的字符串用指定字符串替代，比如应用在步骤2、3、4。

假如我们通过,

```
cont = fileread(original_data_file);
```

得到数据文件里的全部字串。

步骤1通过以下语句实现：

```
% get the static part
static_part_pat = '(?<static>MTS.+?)Cyclic\sAcquisition';
static_part_text = regexp(cont, static_part_pat, 'names','dotall');
static_part_text=static_part_text(1).static;
```

static_part_pat给定匹配模式，意思是匹配MTS开始的任意多字符，直到第一个Cyclic Acquisition。其中.是代表任意字符，+代表一个或多个.，?表达最少匹配多个任意字符，即到达第1个Cyclic Acquisition即停止，\s表达空字符。因为Cyclic Acquisition我们不需要返回，所以将

之前的部分，我们用?<static>来将MTS开始的到Cyclic Acquisition的匹配命名为static。然后通过regexp的names选项，将static部分返回来。

因为这里使用.表示任意字符，包括换行回车，所以要设定另一个参数dotall，即强制.表示包括换行回车在内的任意字符。

regexp带names参数返回的是个结构体，因为有可能是多个匹配，所以是个cell，每个cell是结构体，每个结构体包括各个命名组，比如这里的static。

步骤2使用regexprep。

```
%remove preload data section
preload_section_pat = 'Data\sAcquisition\s+Preload\sData.+?Data';
text1= regexprep(static_part_text,preload_section_pat,'Data','once','dotall');
```

注意匹配模式里包括我们需要的新行的Data，所以我们将用Data来替代匹配字串，从而实现只移除Preload Data部分的内容。

步骤3类似。

```
%remove hold data section
holdload_section_pat = 'Data\sAcquisition\s+Hold\sData.+?Data';
text2= regexprep(text1,holdload_section_pat,'Data','once','dotall');
```

步骤4需要判断是否是第一条头信息，如果非第一条头信息则移除。注意到非第一条头信息向上追溯是数字，而第一条头信息向上追溯是字母(End)。

```
%remove header between data sections
pat = '(\d)\s+Data\sAcquisition.+?MPa';
text3= regexprep(text2, pat,'$1','dotall');
```

匹配模式里包括之前数字，然后替换的时候，用此数字将整个匹配字串替换掉，即实现此目的。注意到我们将此数字括号起来，表达是一个组，替换时，用$1指代此组的内容。这条命令会将全部处于中间的Data Acquisition头信息都去掉。

步骤6：

```
% get the Operator Information header
header_pat = 'MTS.+End';
header_text = regexp(cont, header_pat,'match','once','dotall');
```

注意，给定了once关键词，只匹配一次就好。

步骤7：

```
% get the dynmaic part
dynmaic_part_text = regexprep(cont, static_part_pat, 'Cyclic Acquisition','dotall');
```

这里使用的仍然是步骤1的匹配模式，使用regexprep将静荷载部分去掉。

完整的代码如下：

```
%Author: Zhuotong Nan (giscn@msn.com)
%Date: Mar 31, 2013 @ Pittsburgh, PA, USA

original_data_file = 'N1_20%_0.06kN(13zht61).dat';

[p name ext] = fileparts(original_data_file);
static_filen = [name, '_static',ext];
dynamic_filen = [name,'_dynamic',ext];

disp(['Processing ',name,'...']);
disp('..Reading in');
% get the file content
```

```
cont = fileread(original_data_file);


% get the static part
static_part_pat = '(?<static>MTS.+?)Cyclic\sAcquisition';
static_part_text = regexp(cont, static_part_pat, 'names','dotall');
static_part_text=static_part_text(1).static;

%remove preload data section
preload_section_pat = 'Data\sAcquisition\s+Preload\sData.+?Data';
text1= regexprep(static_part_text,preload_section_pat,'Data','once','dotall');
%remove hold data section
holdload_section_pat = 'Data\sAcquisition\s+Hold\sData.+?Data';
text2= regexprep(text1,holdload_section_pat,'Data','once','dotall');
%remove header between data sections
pat = '(\d)\s+Data\sAcquisition.+?MPa';
text3= regexprep(text2, pat,'$1','dotall');

% output
disp('..Output static loading part');
fid = fopen(static_filen, 'w');
fprintf(fid,'%s',text3);
fclose(fid);


% start to process dynamic part
% get the Operator Information header
header_pat = 'MTS.+End';
header_text = regexp(cont, header_pat,'match','once','dotall');
% get the dynmaic part
dynmaic_part_text = regexprep(cont, static_part_pat, 'Cyclic Acquisition','dotall');
% output
disp('..Output dynamic loading part');
fid = fopen(dynamic_filen, 'w');
fprintf(fid,'%s\n\n%s',header_text,dynmaic_part_text);
fclose(fid);

disp('..DONE..');
```

最后我们注意到有很多个这样的文件需要处理，所以写一个简单的批处理即可，比如这些数据文件是以 *.dat加以区别。

```
for f = dir('*.dat');
splitfile(f.name);
end
```

其中splitfile是前述脚本的函数形式，即在
%Author: Zhuotong Nan ~~(giscn@msn.com)~~ 之前添加
function splitfile (original_data_file)
然后，注释掉原来的 //original_data_file = 'N1_20%_0.06kN(13zht61).dat';
即可。

总结

本例演示了在MATLAB里使用正则表达式从文本中提取或替换子串。演示了regexp、regexprep的使用。本例也演示了对多个文件批处理的实现。