

## Introduction

MUMmer软件用于快速比对非大的DNA或者氨基酸序列。该软件使用suffix tree data structure进行有效地模式匹配。该软件包含5个最常用的包，mummer, nucmer, promer, run-mummer1, run-mummer3。

## Program descriptions

### 1. Maximal exact matching

mummer流程均有3个主要部分组成，首次在两输入之间识别明确的最大程度的完整匹配，然后将这些匹配聚集成群构建优质的比对anchors，最后在这些聚集成群的比对之间延伸比对实现最终的gapped比对。These match lists have great versatility because they contain huge amounts of information and can be passed forward to other interpretation programs for clustering, analysis, searching, etc.

- mummer，用于另序列间查询最大的唯一匹配。最适合生成点图描述一系列精确匹配信息。使用至少包含一个ref文件和一个query文件，其都应为multi-Fasta格式，比对过程中区分大小写。

```
mummer [options] <reference file> <query file1> ... [query file32]
```

计算MUMs, Maximal Unique Matcher；默认为-mumreference，三选一

-mum，计算同时满足ref和query的唯一匹配序列

-mumreference，计算ref上唯一匹配，query无需唯一匹配序列

-maxmatch，不考虑ref和query唯一性的条件下，计算匹配序列

当处理masking DNA序列时，使用-n选型可避免匹配到masking字符

-b/-r/-c，-b表示针对正向和反向互补，-r仅反向互补互补序列匹配，-c为描述query反向互补位置时，使用相对于query序列正向的位置；默认仅输出正向的匹配

-F，第一列输出reference的ID

-s，输出匹配序列字符信息

-L，输出匹配query长度

输出头为query名称后Reverse表反向互补匹配，接着为query序列长度；下一行四列分别为，ref ID，ref起点，query起点，延伸长度；再下一行表对应match序列，参数使用为 -s -F -b -L；

```
81 > NODE_38_length_34527_cov_46.438372 Len = 34527
82 > NODE_38_length_34527_cov_46.438372 Reverse Len = 34527
83 test_seq 1 4 124
84 caggcctcagcattttattatggtgatcccctgggcgaaatgcgcttggttaagcagagt
85 test_seq 1165 34402 125
86 ggcactgttgcaaatagtcggtggtgataaacttatcatccccttttgctgatggagct
```

- repeat-match

用于查询单个输入序列中的最大准确重复序列，因此输入文件经针对第一个序列查询重复序列

```
repeat-match [options] <sequence file>
```

-f, 表仅针对正向序列，默认报告正向和反向互补序列匹配情况

-n, 指定最小的匹配长度，默认为20

输出文件三列，分别为重复序列的第一，第二位置，以及长度；反向互补重复时，第二位置后加r，表相对于序列正向的位置

32874	33006	27
4649	204531	20
49982	75201r	27
285375	285400r	26

- exact-tandems

类似repeat-match，查询单条序列串联重复序列

```
exact-tandems <sequence file> <min length>
```

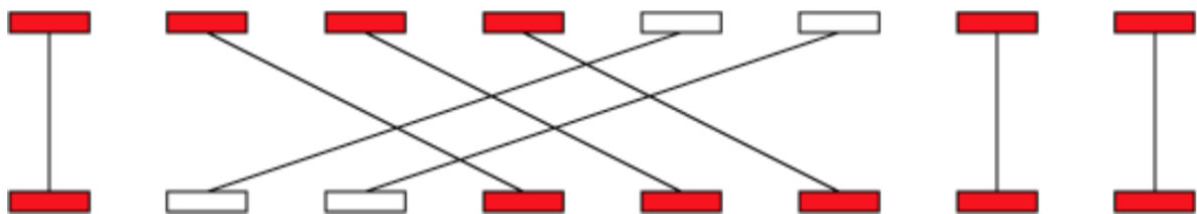
```
$exact-tandems 168_combined_contigs.fasta 10
Finding matches
Genome Length = 366384    Used 236128 internal nodes
Tandem repeats
  Start   Extent  UnitLen  Copies
  11910    14      3        4.7
  19860    20      9        2.2
  26301    13      3        4.3
```

输出为tandem起点，重复延伸长度，重复单元长度，重复次数

## 2. Clustering

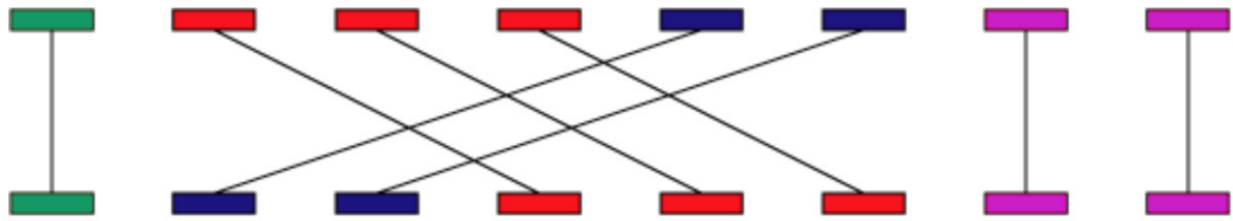
MUMmer的clustering算法用于将小的个体匹配聚集成大的匹配。可以绘制对应点图，来查看匹配情况。

- gaps, 为run-mummer1主要的clustering算法，实际上更类似一个排序的过程。gaps使用LIS(longest increasing subset)算法来从两序列的匹配中提取最大的连续匹配集合，从而生成单个clustering "straight-line", 不包含重排和颠倒匹配情况。因此run-mummer1适用不包含大量变异的非常类似的两序列文件，仅适用与1对1的序列比对。如下图仅红色框包含在LIS中，空白框会被丢弃，因此gaps最适合用于比较几乎相同的序列来查询SNPs和indels。



- mgaps用于更好的处理包含大规模重排和重复的序列比对，不同于gaps，mgaps使用full clustering算法，能够输出多重一致性顺序的匹配集合。mgaps主要的优势在于能够识别保守的"islnds"序列，因此可帮助识别大规模重排，重复以及基因家族等，适用与1对多的序列比对。下图描述的每一独立的clustering使用独立的颜色表示，因此mgaps更使用于整体比对而不是局

部位置突变检测。



### 3. Alignment generators

实现以上两个步骤maximal exact matching和clustering后，就可以进行alignment过程了。以下程序均独立完成matching和clustering步骤，然后针对每个cluster生成pair-wise比对。每个cluster中的每一个match被用作alignment过程中的anchors，且仅match中的错配才使用Smith-Waterman动态矩阵处理。

- NUCmer

用于比对两个多重近源核酸序列(multiple closely related nucleotide sequences)。**nucmer**常用于决定一套序列contigs相对于ref genome的位置和方向。该程序首先使用mummer识别Maxima exact matching，然后使用mgaps对个体matches进行clustering，最后自clustering向外延伸增加整体比对覆盖度。

```
nucmer [options] <reference file> <query file>
```

--mum,--mumreference,--maxmatch, 说明同mummer，默认为--mumreference

--mincluster, --minmatch, 降低该值将会增加比对敏感性，从而产生一些可信度降低的比对

--maxgap, 显著性增加该值(1000)，应用于差异较大的基因组比对

--noextend, 会阻止cluster的延伸，从而增加运行速度

--nodelta, 将进一步提速，阻止cluster间的比对

--nosimplify, 用于非准确性重复序列的检出

nucmer和promer输出同样格式文件，后缀delta

```
1 /Data_analysis/mummer_test/GCF_000240185.1_ASM24018v2_genomic.fna /Data_analysis/mummer_test/168_combined_contigs.fasta
2 NUCMER
3 >NC_016845.1 NODE_1_length_366384_cov_31.240309 5333942 366384
4 1358934 1360505 327341 325770 199 199 0
5 -648
6 5
7 -309
8 4
9 0
10 3578503 3578632 366255 366384 0 0 0
```

第1行为输入比对文件名

第2行为使用的比对程序nucmer/promer

第3行为参与比对的ref和query序列名称，及各自长度

第4行为ref起点终点，query起点终点，如果终点大于起点表示比对到了反向链，接着3列表示，错配数目(indels+snps)，类似错配数目(non-positive match scores，该值针对promer有特殊意义)，终止密码子数目(nucmer时均为0；针对promer输出，对应为起始密码子和终止密码子位置，满足(end-start+1)%3=0。

接下来的几行描述了插入缺失的之间的距离，正值表示ref出现插入；负值表示ref出现缺失(即query出现插入)。例如promer: 1, -3, 4, 0, 表示相互之间的距离

```
A = ABCDACBDCAC$
B = BCCDACDCAC$
Delta = (1, -3, 4, 0)
A = ABC.DACBDCAC$
B = .BCCDAC.DCAC$
```

表示ref在1位置出现插入，A；query在位置3出现插入，C；ref在位置7出现了插入，B。

- PROmer

用于比对距离偏远的核酸序列(multiple somewhat divergetn nucleotide sequences)。运行类似nucmer，不同的是首先将输入核酸序列按照6种方式转录成氨基酸序列，敏感度高于nucmer。为保证比对精确性，可能需要mask输入序列避免比对到不感兴趣的序列位置，或者改变uniqueness constraints减少重复带来的比对情况。

```
promer [options] <reference file> <query file>
```

参数选型类似nucmer，输出格式为delta

--masklen, 将位于终止密码子之间的指定数目的氨基酸mask，例如设为4，纳米将mask等于或大于4的长度

```
...AAA*AAAA*AAA... 将mask为 ...AAA*XXXX*AAA...
```

--matrix, 设置BLOSUM矩阵用于错配打分，1假设量序列差异很大，3假设差异很小

- run-mummer1

同nucmer/promer, matching/clustering/extension, 只是run-mummer1采用gaps用于clustering步骤，不允许重排。因此，**更适用于检出SNP和indel**。

```
run-mummer1 <reference file> <query file> <prefix> [-r]
```

-r, 仅比对query反向互补序列，默认为仅比对query的正向序列，ref均为正向序列

**ref和query文件需为fasta格式同时仅含一条序列**，run-mummer1使用简化的打分函数无法识别masking 字符，不推荐对输入文件masking。输出4个文件，分别为out, gaps, errorsgaps, align文件

out文件格式同mumer输出文件，只是不含头文件：

1	549775	46864	50
2	1507590	46862	42
3	3578503	366255	130
4	3579153	1	50
5	3579204	52	251
6	3579456	304	202
7	3579659	507	38

gaps文件：

```

1 > test_ref.fasta Consistent matches
2 3579153      1      50      none      -      -
3 3579204      52     251     none      1      1
4 3579456     304     202     none      1      1
5 3579659     507      38     none      1      1
6 4098791     2762   23310   none  519094   2217
7 4122102     26073   1195     none      1      1
8 4123298     27269   6267     none      1      1
9 4129565     33763   8834     -79       0     227

```

第一行为参考文件名

第二行前3列通mummer输出

第二行后3列为与上一个match之间的重合信息；当前ref的match起点和上一个match终点之间的gap距离；当前match的match起点和上一个match终点之间的gap距离

当gap大小为1时，表两序列之间出现了snp；an overlap like seen in the last line of the Consistent matches indicates the existence of a tandem repeat，这里个人理解为存在"-79"的串联重复序列； "-"表示gap大小无法计算

```

36 > test_ref.fasta Other matches
37 1507590     46862     42      none      -      -
38 549775      46864     50      none      -      -
39 3578503     366255    130     none  3028678  319341

```

Other matches显示了不包含在LIS中的匹配(like the white boxes in the above image)

errors gaps文件：

该输出文件为gaps格式的注释版本，可能时run-mummer1用于识别SNPs的最有用文件，最后一列的1表示snp， "-"表示距离太大无法计算

```

1 > test_ref.fasta Consistent matches
2 3579153      1      50      none      -      -
3 3579204      52     251     none      1      1      1
4 3579456     304     202     none      1      1      1
5 3579659     507      38     none      1      1      1
6 4098791     2762   23310   none  519094   2217      -
7 4122102     26073   1195     none      1      1      1
8 4123298     27269   6267     none      1      1      1
9 4129565     33763   8834     -79       0     227     227

```

align文件：

包含比对信息，可通过"^"字符识别错配，如果比对太大无法显示，则表示为 "\*\*\* Too long \*\*\*"，每个错配前后均显示10bp序列。

```

3 3579659      507      38      none      1      1
4      Errors = 1
5 T:  attttaatcaacgagtcagct
6 S:  attttaatcagcgagtcagct
7           ^
8 4098791      2762    23310      none 519094      2217
9
10 *** Too long ***

```

- run-mummer3

run-mummer3的matching和clustering过程同nucmer和promer，只是采用了不同的extension处理。run-mummer3仅能处理单个ref序列，但是query文件可包含多重序列，同时能够检出大大重排。输出格式同run-mummer1，只是align文件中使用"="表示MUM部分。

## Utilities

- delta-filter

用于出列nucmer和promer输出的delta编码的比对文件。

```
delta-filter [options] <delta file> < filtered delat file>
```

```

-q          Query alignment using length*identity weighted LIS. For each query, leave only the
           alignments which form the longest consistent set for the query

-r          Reference alignment using length*identity weighted LIS. For each reference, leave only
           the alignments which form the longest consistent set for the reference.

```

-g选项输出最长的一致性比对(determine the longest mutually cosistent set of matches); -r和-q仅要求匹配和ref或query一致即可，主要不同为，-g不允许inversions和translocations，然而-r和-q允许，但是该3个选项均不允许多重重复拷贝。-g参数保证整体一致性；-r保证ref到query的最佳比对(1对多)；-q保证query到ref的最佳比对(多对1)，-r和-q同时使用保证了ref到query1对1的比对。

- show-aligns

show-aligns解析nucmer和promer的delta输出，展示成对比对情况，适合识别准确错配位置，查找两序列间的snps。

```
show-aligns [options] <delta file> <IdR> <IdQ>
```

IdR为目的ref序列的头标签，IdQ为目的query序列的的头标签

```
show-aligns ref_168.delta NC_016845.1 NODE_1_length_366384_cov_31.240309
```

```

1 /Data_analysis/mummer_test/GCF_000240185.1_ASM24018v2_genomic.fna /Data_analysis/mummer_test/168_combined_contigs.fasta
2
3 =====
4 -- Alignments between NC_016845.1 and NODE_1_length_366384_cov_31.240309
5
6 -- BEGIN alignment [ +1 1358934 - 1360505 | -1 327341 - 325770 ]
7
8
9 1358934      cctggttcaatttatcgaagctagtttgaaagcaaggttatgaggaatc
10 327341      cctggaccaatttatcgaagcaagtttgaaagcagggttatgaggaacc
11           ^^          ^          ^          ^
12
13 1358983      catatgtctgaacaacaagcacaggtgccgatgaggccattgacctta
14 327292      aacatgtctgaacaacaagcacagggcgctgacgcggcaattgacctta
15           ^ ^          ^ ^ ^ ^ ^ ^

```



```

136
137 1360500    gtaatc
138 325775    ataatc
139          ^
140
141
142 -- END alignment [ +1 1358934 - 1360505 | -1 327341 - 325770 ]
143 -- BEGIN alignment [ +1 3578503 - 3578632 | +1 366255 - 366384 ]
144

```

promer输出类似nucmer输出，空白表错配，"+"表相似(positive alignment scores)，"\*"表示插入删除。

- show-coords

常用于分析delta输出，用于展示位置，一致性等信息。

```
show-coords [options] <delta file>
```

-c选项显示覆盖度，-l选项显示线序列长度，在比对两组contigs时可帮助查看是否比对跨越整个contig

-b选项可帮助识别另基因组间的syntenic区域，biologists usually refer to synteny as the conservation of blocks of order within two sets of chromosomes that are being compared with each other

-g选项仅显示包含在longest ascending subset中的比对信息，推荐和-r或-q一起使用，当在完整的序列上比对代表性reads时有帮助。(g option comes in handy when comparing sequences that share a linear alignment relationship, that is there are no rearrangements)

```

1 /Data_analysis/mummer_test/GCF_000240185.1_ASM24018v2_genomic.fna /Data_analysis/mummer_test/168_combined_contigs.fasta
2 NUCMER
3
4 [S1] [E1] | [S2] [E2] | [LEN 1] [LEN 2] | [TAGS]
5 =====
6 17991 18120 | 100500 100629 | 130 130 | NC_016838.1 NODE_21_length_100629_cov_30.374948
7 26489 26616 | 126 253 | 128 128 | NC_016838.1 NODE_223_length_253_cov_39.904762
8 26490 26616 | 8934 9060 | 127 127 | NC_016838.1 NODE_55_length_9060_cov_38.296429
9 26490 28276 | 1 1787 | 1787 1787 | NC_016838.1 NODE_84_length_1787_cov_88.657229
0 26585 27909 | 1776 3100 | 1325 1325 | NC_016838.1 NODE_74_length_3283_cov_49.370089
1 28150 29222 | 127 1199 | 1073 1073 | NC_016838.1 NODE_104_length_1199_cov_50.346082

```

- show-snps

用于展示promer/nucmer输出delta文件中包含的snp信息，它将delta文件中的所有snp和indel进行分类。

```
show-snps [options] <delta file>
```

-c选项将限制输出信息，避免输出重复区域的snps信息，该区域展示的snp是有问题的，可能由于simple repeats, tandem repeats导致，大多数情况应使用-c

-H,-T选项适用于进一步分析

-x用于打印指定ref和query位置位置两侧的序列信息数目，"."表示删除，"-"表示序列末端

```

1 /Data_analysis/mummer_test/GCF_000240185.1_ASM24018v2_genomic.fna /Data_analysis/mummer_test/168_combined_contigs.fasta
2 NUCMER
3
4 [P1] [SUB] [P2] | [BUFF] [DIST] | [R] [Q] | [CTX R] [CTX Q] | [FRM] [TAGS]
5 =====
6 26592 C G 3093 | 8 191 | 3 0 | TGCCGCCAGACGTAATCGCCG CTGCCGCCAGAGGTAATCGCCG | 1 -1 NC_016838.1 NODE_74_length_3283_cov_49.370089
7 26605 T G 3080 | 13 204 | 3 0 | AATGCCCGTTAGGTTGATGT AATGCCCGGTGAGGTTGATGT | 1 -1 NC_016838.1 NODE_74_length_3283_cov_49.370089
8 26623 G A 3062 | 3 222 | 1 0 | TGTGCTCCAGCCAGCGGCG TGTGCTCCCAACCGAGCGGCG | 1 -1 NC_016838.1 NODE_74_length_3283_cov_49.370089
9 26626 C G 3059 | 3 225 | 1 0 | GCTCCAGCCAGCGGCGACA GCTCCCAACCGAGCGGCGACA | 1 -1 NC_016838.1 NODE_74_length_3283_cov_49.370089
0 26639 A T 3046 | 2 238 | 1 0 | CGGCGACAGGAATTGCAGCAG CGGCGACAGGTACTGCAACAG | 1 -1 NC_016838.1 NODE_74_length_3283_cov_49.370089
1 26641 T C 3044 | 2 240 | 1 0 | GCGACAGGAATTGCAGCAGCT GCGACAGGTACTGCAACAGCG | 1 -1 NC_016838.1 NODE_74_length_3283_cov_49.370089

```

[p1]为ref上snp位置，若为indel，位置描述为indel前的一个位置，例如，开始位置描述为0；若处在反向互补链，位置描述为反向互补链上indel前一个位置在正向链的位置，例如，反向互补链末端的indel，将描述为1；[P2]对应为query位置

[SUB]对应描述ref和query该位置信息

[BUFF]描述同一比对上该位置到最近错配距离

[DIST]描述该snp到最近序列末端距离

[R]多少个重复比对包含该ref位置；[Q]多少重复比对包含该query位置

[LEN R]ref序列长度；[LEN Q]query序列长度

[FRM]序列或者reading frame方向

以上所有位置均为相对于DNA输入序列正链的位置。

- show-tiling

构建query contigs比对到ref序列的tiling路径，将决定每个query contig的最佳比对位置。由于每个contig将被tiling一次，因此重复区域将给该分析过程带来困难。

```
show-tiling [options] <delta file>
```

-c假设参考序列为环状，同时允许tiling contigs跨越参考序列起点

-i指定最小一致性，nucmer默认90%，promer默认55%

-l指定最小contig长度，默认1

-v设定最小的contig比对覆盖度，nucmer默认90%，promer默认55%；-V设定判断contig一个map优于另一个map的差异值，nucmer默认10%，promer默认30%；为包含最可能的contigs，设置-V为0，同时降低-i和-v值

-u指定输出文件包含为使用的contigs比对信息，格式略！

```
1 >NC_016838.1 122799 bases
2 26490 28276 94523 1787 100.00 100.00 - NODE_84_length_1787_cov_88.657229
3 >NC_016839.1 105974 bases
4 28627 28754 14573 128 100.00 100.00 - NODE_261_length_128_cov_186.000000
5 43328 43455 -127 128 99.22 100.00 - NODE_263_length_128_cov_77.000000
6 43329 43456 62518 128 100.00 100.00 + NODE_259_length_128_cov_262.000000
7 >NC_016845.1 5333942 bases
8 21078 120603 4677 99526 100.00 99.99 + NODE_22_length_99526_cov_31.218543
9 125281 125513 -124 233 100.00 99.57 - NODE_231_length_233_cov_34.660377
0 125390 125542 41 153 100.00 98.69 - NODE_244_length_153_cov_66.730769
1 125584 212475 5020 86892 100.00 100.00 - NODE_25_length_86892_cov_31.338708
2 217496 222197 282 4702 100.00 100.00 - NODE_70_length_4702_cov_30.554973
3 222480 257471 5088 34992 100.00 100.00 + NODE_36_length_34992_cov_31.226502
```

```
1 >NC_016838.1 122799 bases
2 26490 28276 121012 1787 100.00 100.00 - NODE_84_length_1787_cov_88.657229
3 >NC_016839.1 105974 bases
4 28627 28754 14573 128 100.00 100.00 - NODE_261_length_128_cov_186.000000
5 43328 43455 -127 128 99.22 100.00 - NODE_263_length_128_cov_77.000000
6 43329 43456 91144 128 100.00 100.00 + NODE_259_length_128_cov_262.000000
7 >NC_016845.1 5333942 bases
8 -47034 15975 5102 63010 99.80 99.98 - NODE_29_length_63010_cov_31.420320
9 21078 120603 4677 99526 100.00 99.99 + NODE_22_length_99526_cov_31.218543
0 125281 125513 -124 233 100.00 99.57 - NODE_231_length_233_cov_34.660377
1 125390 125542 41 153 100.00 98.69 - NODE_244_length_153_cov_66.730769
```



第1/2列为ref起始终止位置

第3列, 该contig和下一contig之间第gap长度

第4/5/6列为该conitg的长度, 覆盖度及一致性百分比

第7/8列为contig方向(-表示反向互补序列)和ID

起点位置为负值表示该contig包围了该ref序列起点位置

## Usage

- 比对两连续序列, 然后使用mummerplot查看整体比对情况

```
mummer -num -b -c ref.fasta qry.fasta > ref_qry.num
```

```
mummerplot --postscript --prefix=ref_qry ref_qry.mums
```

```
gnuplot ref_qry.gp
```

- 比对高度相似无重排序列, 用于检测snp和小的indel, 同时ref和query仅包含1条序列

```
run-mummer1 ref.fasta qry.fasta ref_qry
```

```
run-mummer1 ref.fasta qry.fasta ref_qry -r
```

同样可以使用nucmer进行snp检出

- 含重排的高度相似序列比对, 两序列可能含有大段序列重排, 颠倒或插入等, ref仅含1条序列, query可含多条序列

```
run-mummer3 ref.fasta qry.fasta ref_qry
```

- 相当相似的序列比较, run-mummer\*着重于差异, 而nucmer着重于相似; 同时重排, 倒置, 重复都会被nucmer识别, ref和query可包含多重序列

```
nucmer --maxgap=500 --mincluster=100 --prefix=ref_qry ref.fasta qry.fasta
```

```
show-coords -r ref_qry.delta > ref_qry.coords
```

```
show-aligns ref_qry.delta refname qryname > ref_qry.aligns
```

refname和qryname为fasta序列IDs, 同时可使用mummerplot查看, delta-filter过滤输出, 例如选择1对1的比对

```
delta-filter -q -r ref_qry.delta > ref_qry.filter
```

```
mummerplot ref_qry.filter -R ref.fasta -Q qry.fasta
```

- 相当不相似的序列, promer将转录DNA, 然后用于比对, ref和query可包含多重序列

```
promer --prefix=ref_qry ref.fasta qry.fasta
```

```
show-coords -r ref_qry.delta > ref_qry.coords
```

```
show-aligns -r ref_qry.delta refname qryname > ref_qry.aligns
```

以上使用-k选项将输出多重比对的最佳情形; 同样可使用mummerplot查看, deleta-filter过滤输出, 例如选择1对1的比对

```
delta-filter -q -r ref_gry.delta > ref_gry.filter
```

```
mummerplot ref_gry.filter -R ref.fasta -Q gry.fasta
```

- 比对两基因组组装序列，可使用nucmer或promer，分别针对相似或不相似的两fasta文件

```
nucmer --prefix=ref_gry ref.fasta gry.fasta
```

```
show-coords -rcl ref_gry.delta > ref_gry.coords
```

```
show-aligns ref_gry.delta refname qryname > ref_gry.aligns
```

由于多重fasta文件比对输出信息很多，使用delta-filter过滤输出，-r比对位置以ref为准，-q比对位置以query序列为准，-q -r指定输出1对1的比对，同时可使用mummerplot查看比对情况

```
mummerplot ref_gry.delta -R ref.fasta -Q gry.fasta --filter -layout
```

- 比对组装的scaffolds文件到参考基因组，可用于根据ref基因组判断query contig的位置和方向，帮助完成draft序列组装

```
nucmer --prefix=ref_gry ref.fasta gry.fasta
```

```
show-coords -rcl ref_gry.delta > ref_gry.coords
```

```
show-aligns ref_gry.delta refname qryname > ref_gry.aligns
```

```
show-tiling -c ref_gry.delta > ref_gry.tiling
```

当ref和query包含多重序列时，show-aligns步骤可重复运行展示比对情况。假如不想比对draft序列到重复的位置，使用delta-filter选择每个draft序列在ref上的最佳位置

```
delta-filter -q ref_gry.delta > ref_gry.filter
```

- 检出SNPs，比对序列，选择1对1的比对情况，检出SNP位置；nucmer可在存在多个重排的两基因组之间检出SNPs

```
nucmer --prefix=ref_gry ref.fasta gry.fasta
```

```
show-snp -Clr ref_gry.delta > ref_gry.snps
```

-C选项指定唯一比对的序列用于检出SNPs，排除了重复序列内的SNPs；或者首先排除重复序列位置；-1，表示1对1允许重排比对，适用于SNP查询

```
nucmer --prefix=ref_gry ref.fasta gry.fasta
```

```
delta-filter -1 -r -q ref_gry.delta > ref_gry.filter
```

```
show-snp -Clr ref_gry.filter > ref_gry.snps
```

- 非精确识别重复序列，尽管mummer并不是设计用于检测重复序列，但是可通过一些参数选择检出重复序列和串联重复序列；可以将序列与自身进行比较，同时设置参数--maxmatch/--nosimplify并且排除两输入序列相同位置的hits，进而得到重复序列输出

```
nucmer --maxmatch --nosimplify --prefix=seq_seq seq.fasta seq.fasta
```

```
show-coords -r -b seq_seq.delta > seq_seq.coords
```

查询准确重复长度不低于50的序列

```
repeat-match -n 50 seq.fasta > seq.repeats
```

查询准确串联重复长度不低于50的序列

```
exact-tandems seq.fasta 50 > seq.tandems
```