Zhang-Lab 生信小课堂 第十一期

Applied Bioinformatics Club（ABC）

和趣求真 秉实生信

（张建伟生物信息学课题组 https://zhang.hzau.edu.cn）

基于基因同源性的共线性及WGD分析

拥有物种基因组和注释文件，如何做全基因组复制分析（WGD）？

2023.4.7 二综一楼C102 15:00 欢迎大家交流学习！

主讲人：王欢

2023/04/07

目录

CONTENTS

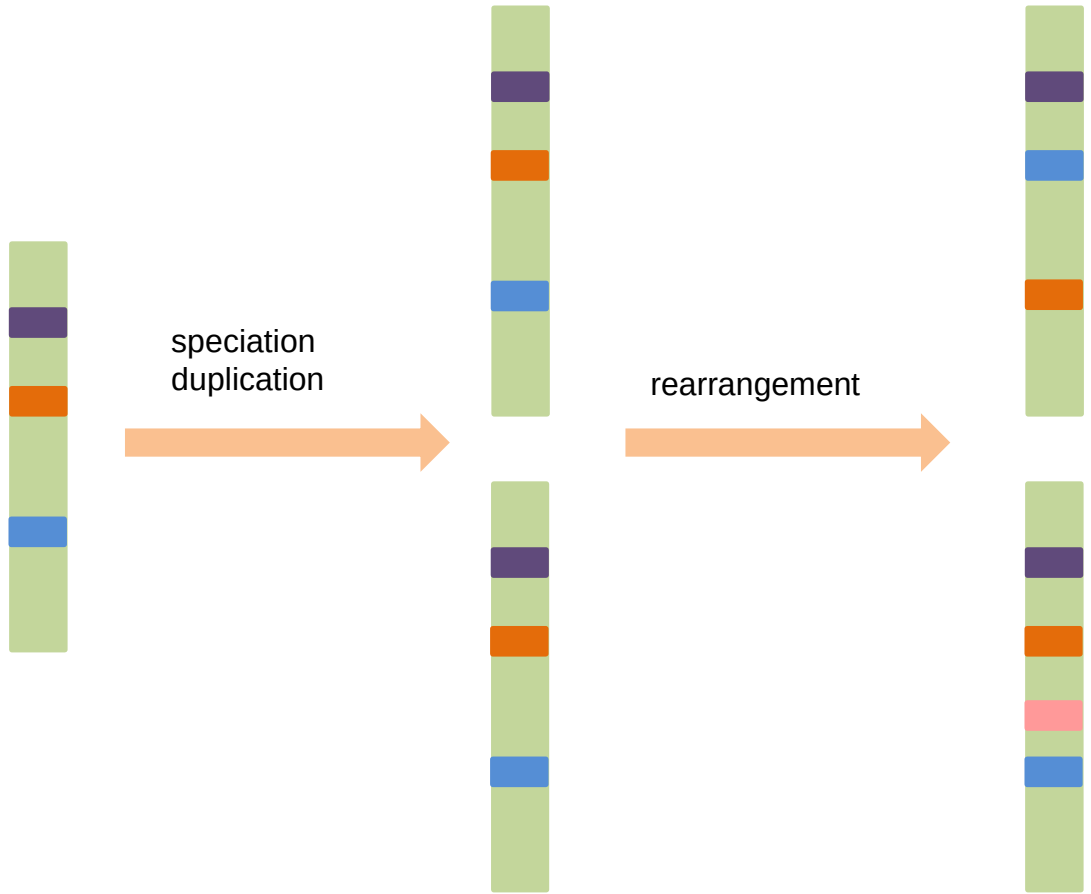
PART 01

共线性分析

ENTER YOUR SUBTITLE

1.1 概念

Synteny：多个基因/遗传位点起源与同一条染色体，可以是物种间（通过物种分化产生），或者是物种内（染色体加倍）



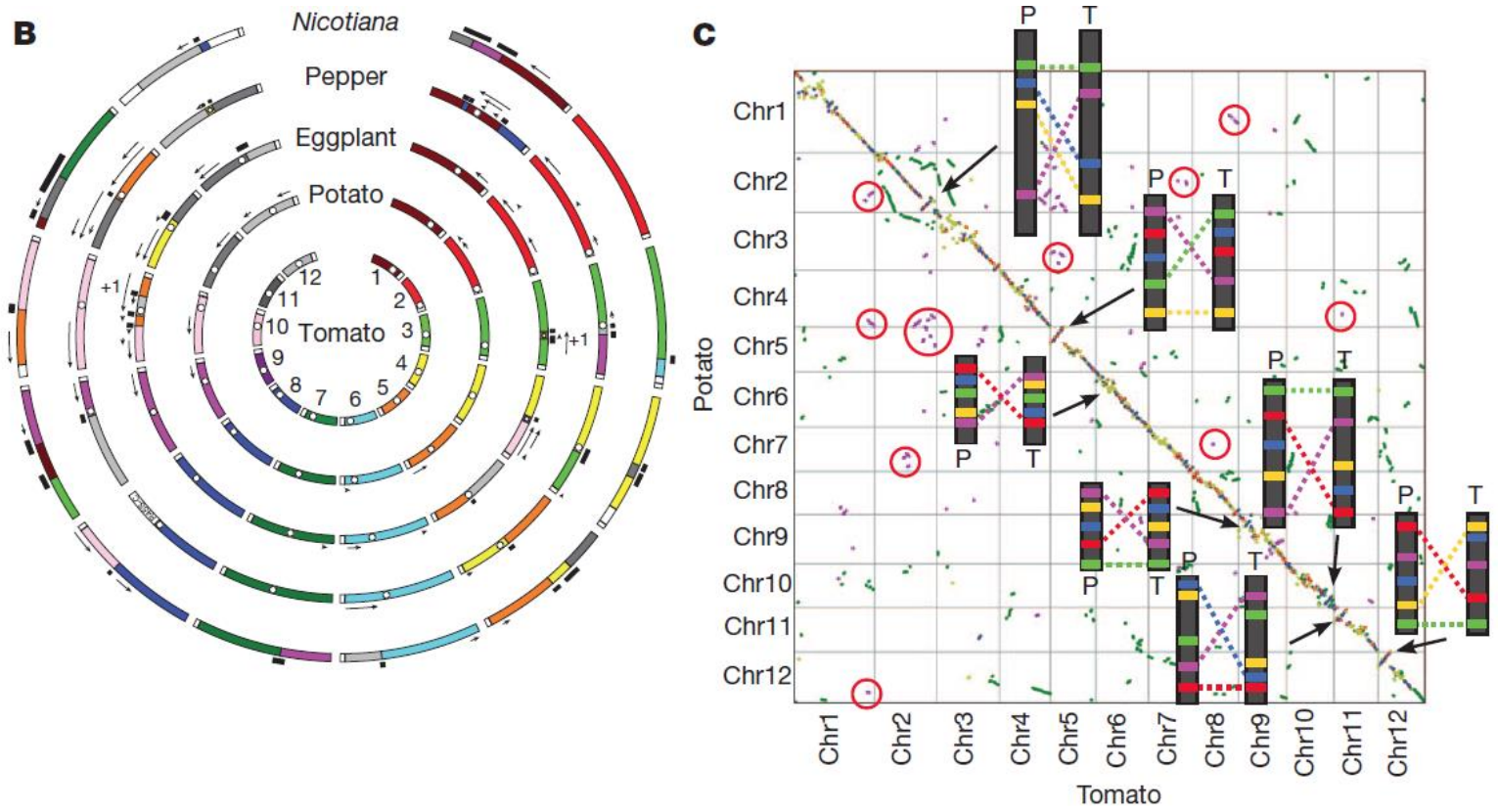
1.2 意义

（1）研究染色体结构变异：判断染色体结构的一致程度（亲缘关系，进化速率）；鉴定倒位、易位、插入、缺失、复制等事件；构建古染色体

（2）研究基因进化历史：区分直系同源，旁系同源

（3）重要功能基因的插入缺失

（4）鉴定全基因组复制事件（WGD）

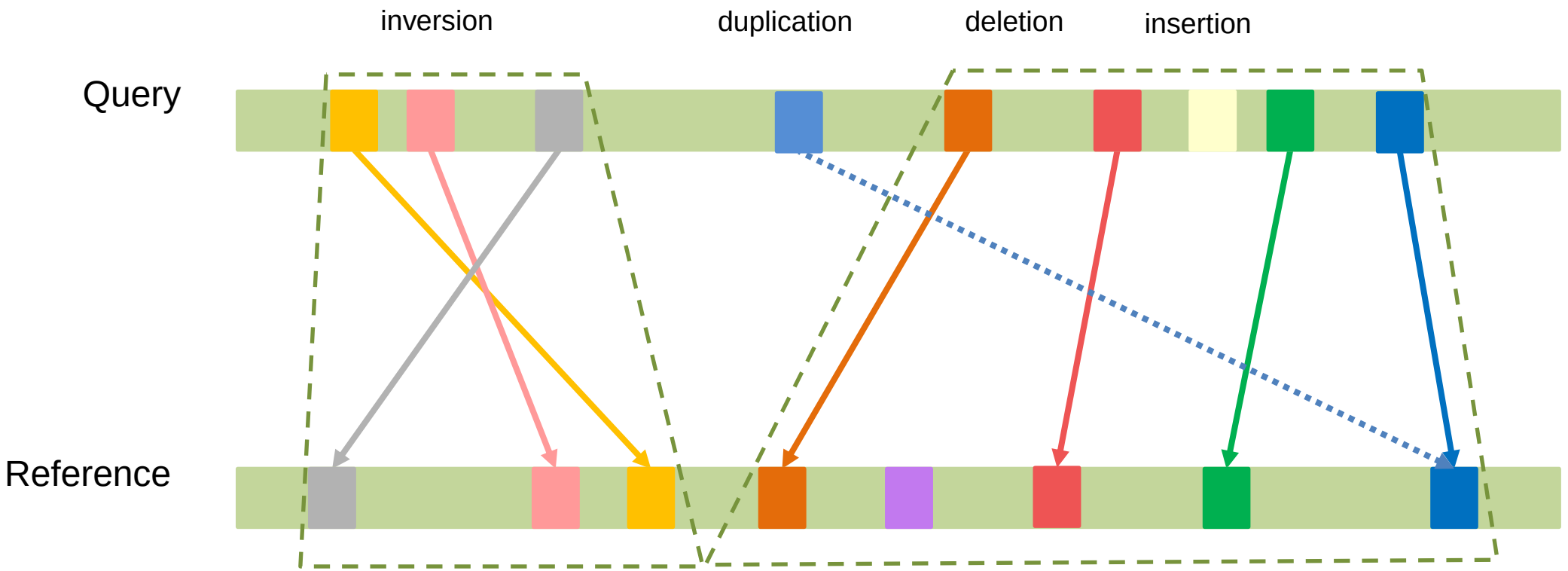


1.3 原理

（1）已知基因在染色体上的位置（gff，bed文件）以及基因序列（cds/pep）

（2）将基因的cds/pep序列进行比对，得到高序列相似性的基因对（anchoring）

（3）鉴定具有相同基因排列顺序的共线性区块（chaining）



1.4 分析方法

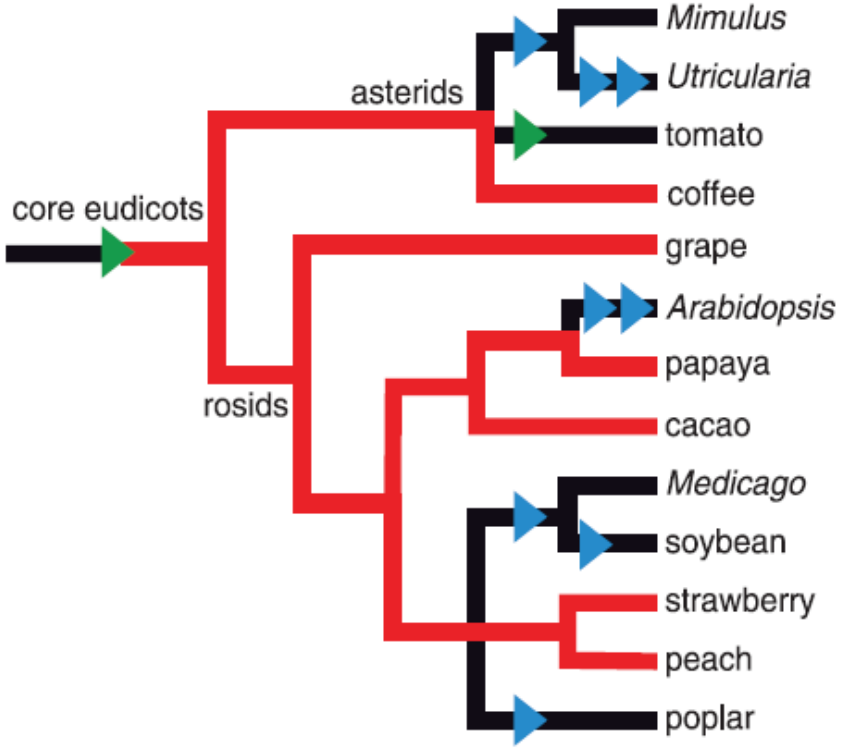
这里以大豆（Glycine max）与蒺藜苜蓿（Medicago truncatula）的作为对象，介绍mcscan，MCScanX，JCVI包中的mcscan的使用方法。

数据背景:

（1）大豆与蒺藜苜蓿同为豆科植物

（2）蒺藜苜蓿在核心真双子叶植物共有的三倍化事件（γ）后，发生过一次二倍化WGD

（3）大豆除了与蒺藜苜蓿共有的γ和二倍化WGD外，还在13 Mya前发生过一次二倍化WGD



Denoeud F, et al. Science (2014)

1.4 分析方法

1.4.1 mcscan

(1) references

- http://chibba.agtec.uga.edu/duplication/mcscan/
- **Tang, H., Bowers, J.E., Wang, X., Ming, R., Alam, M., and Paterson, A.H.** (2008) Synteny and Collinearity in Plant Genomes. *Science*, 320, 486-488.
- **Tang, H., Bowers, J.E., Wang, X., and Paterson, A.H.** (2009) Angiosperm genome comparisons reveal early polyploidy in the monocot lineage. *PNAS*.

(2) 步骤

Step1：下载大豆和蒺藜苜蓿的蛋白序列及gff文件并过滤成最长转录本，得到Gm.pep，Gm.gff，Mt.pep，Mt.gff。然后将gff转为bed格式后合并：

```
$ python -m jcvi.formats.gff bed Gm.gff --type=mRNA --key=ID > Gm.bed
$ python -m jcvi.formats.gff bed Mt.gff --type=mRNA --key=ID > Mt.bed
$ awk '{print "Gm"$1"\t"$2"\t"$3"\t"$4}' Gm.bed > Gm_vs_Mt.bed
$ awk '{print "Mt"$1"\t"$2"\t"$3"\t"$4}' Mt.bed >> Gm_vs_Mt.bed
```

- 也可以用其他方法转换格式，保证蛋白序列中的ID与bed中一致
- 为防止混淆，在染色体前加上两个字母的缩写表示物种名，并且要避免两个物种存在相同的蛋白ID
- 这里合并后的为非标准的.bed格式：sp# start end gene

1.4 分析方法

1.4.1 mcscan

(2) 步骤

Step2：将大豆的蛋白序列对蒺藜苜蓿的蛋白序列进行blastp比对：

```
$ makeblastdb -dbtype prot -in Mt.pep
$ blastp -db Mt.pep -query Gm.pep -evalue 1e-5 -num_threads 32 -outfmt 6 -out Gm_vs_Mt.blast
```

Step3：处理比对结果文件格式，保留qseqid sseqid evalue三列，使用mcscan自带的filter_blast.py进行过滤，再使用mcl聚类：

```
$ cut -f 1,2,11 Gm_vs_Mt.m8 > Gm_vs_Mt.unfiltered.blast
$ filter_blast.py Gm_vs_Mt.unfiltered.blast Gm_vs_Mt.blast
$ more Gm_vs_Mt.blast | mcl - --abc --abc-neg-log -abc-tf 'mul(0.4343), ceil(200)' -o
Gm_vs_Mt.mcl
```

- filter_blast.py将会让每对基因只保留一个evalue（去除反向比对以及HSP）

Step4：运行mcscan：

```
$ mcscan Gm_vs_Mt
```

- mcscan将读入.blast文件、.mcl文件和.bed文件，因此保证三者前缀一致（Gm_vs_Mt）

1.4 分析方法

1.4.1 mcscan

（3）结果解释

将会生成两个文件：Gm_vs_Mt.aligns 和 Gm_vs_Mt.blocks

Gm_vs_Mt.aligns文件：

```
############### Parameters ###############
# MATCH_SCORE: 40
# MATCH_SIZE: 5
# UNIT_DIST: 2
# GAP_SCORE: -2
# OVERLAP_WINDOW: 8
# EXTENSION_DIST: 40
# E_VALUE: 1e-05
# PIVOT: ALL
##########################################

## Alignment 0: score=688.0 e_value=1e-35 N=20 GmChr01&Mtchr1 plus
  0-  0:        Glyma.01G232800.1.Wm82.a2.v1    Medtr1g103160.2.JCVIMt4.0v1          0
  0-  1:        Glyma.01G236300.1.Wm82.a2.v1    Medtr1g103180.1.JCVIMt4.0v1       3e-18
  0-  2:        Glyma.01G236500.1.Wm82.a2.v1    Medtr1g103320.1.JCVIMt4.0v1       4e-22
  0-  3:        Glyma.01G236800.1.Wm82.a2.v1    Medtr1g103500.3.JCVIMt4.0v1       2e-89
  0-  4:        Glyma.01G237000.1.Wm82.a2.v1    Medtr1g103550.1.JCVIMt4.0v1       1e-48
  0-  5:        Glyma.01G237300.1.Wm82.a2.v1    Medtr1g103570.1.JCVIMt4.0v1       2e-47
  0-  6:        Glyma.01G237400.1.Wm82.a2.v1    Medtr1g103600.1.JCVIMt4.0v1      1e-146
  0-  7:        Glyma.01G238400.1.Wm82.a2.v1    Medtr1g104500.2.JCVIMt4.0v1      1e-172
  0-  8:        Glyma.01G238500.1.Wm82.a2.v1    Medtr1g104520.1.JCVIMt4.0v1       7e-35
  0-  9:        Glyma.01G238700.1.Wm82.a2.v1    Medtr1g104680.1.JCVIMt4.0v1          0
  0- 10:        Glyma.01G239200.1.Wm82.a2.v1    Medtr1g104870.1.JCVIMt4.0v1          0
  0- 11:        Glyma.01G239800.1.Wm82.a2.v1    Medtr1g105075.1.JCVIMt4.0v1       1e-34
  0- 12:        Glyma.01G239900.1.Wm82.a2.v1    Medtr1g105305.1.JCVIMt4.0v1      5e-103
  0- 13:        Glyma.01G240000.1.Wm82.a2.v1    Medtr1g105415.1.JCVIMt4.0v1       3e-93
  0- 14:        Glyma.01G240500.3.Wm82.a2.v1    Medtr1g105555.2.JCVIMt4.0v1       1e-45
```

.aligns文件包含各个共线性区块与基因对。

- ## 开头的为一个alignment。此外还包括score，evalue，基因对数量，染色体，比对方向等信息。
- 第1列为alignment编号
- 第2列为基因对编号
- 第3列、第4列分别为一对基因。
- 第5列为blast比对的evalue

1.4 分析方法

1.4.1 mcscan

（3）结果解释

将会生成两个文件：Gm_vs_Mt.aligns 和 Gm_vs_Mt.blocks

Gm_vs_Mt.blocks文件：

```
############### Parameters ###############
# MATCH_SCORE: 40
# MATCH_SIZE: 5
# UNIT_DIST: 2
# GAP_SCORE: -2
# OVERLAP_WINDOW: 8
# EXTENSION_DIST: 40
# E_VALUE: 1e-05
# PIVOT: ALL
########################################

## View 0: pivot GmChr01
0-  0:   Glyma.01G000100.1.Wm82.a2.v1                          .                          .  .  .
0-  1:   Glyma.01G000200.1.Wm82.a2.v1                          .                          .  .  .
0-  2:   Glyma.01G000400.1.Wm82.a2.v1;Glyma.01G000600.1.Wm82.a2.v1  .                      .  .  .
0-  3:   Glyma.01G000700.1.Wm82.a2.v1                          .                          .  .  .
0-  4:   Glyma.01G000800.1.Wm82.a2.v1                          .                          .  .  .
0-  5:   Glyma.01G000900.1.Wm82.a2.v1                          .                          .  .  .
0-  6:   Glyma.01G001000.1.Wm82.a2.v1                          Medtr6g093220.1.JCVIMt4.0v1  .  .  .
0-  7:   Glyma.01G001100.2.Wm82.a2.v1                          Medtr6g093210.1.JCVIMt4.0v1  .  .  .
0-  8:   .                                                     Medtr6g093180.2.JCVIMt4.0v1  .  .  .
0-  9:   .                                                     Medtr6g093170.1.JCVIMt4.0v1  .  .  .
0-  10:  Glyma.01G001200.1.Wm82.a2.v1                          Medtr6g093150.2.JCVIMt4.0v1  .  .  .
0-  11:  .                                                     Medtr6g093100.1.JCVIMt4.0v1  .  .  .
0-  12:  .                                                     Medtr6g093070.1.JCVIMt4.0v1  .  .  .
0-  13:  Glyma.01G001300.2.Wm82.a2.v1                          .                          .  .  .
0-  14:  Glyma.01G001400.1.Wm82.a2.v1                          .                          .  .  .
0-  15:  Glyma.01G001500.1.Wm82.a2.v1                          .                          .  .  .
```

开头的为一个view。后面跟着的是reference的染色体

- 第1列为view编号
- 第2列为基因编号，第3列为reference的基因。未比对上的用 . 表示

后面的几列为比对上reference的共线性区块的基因。未比对上的用 . 表示

1.4 分析方法

1.4.2 MCScanX

（1）references

- http://chibba.pgml.uga.edu/mcscan2
- **Wang Y, Tang H, DeBarry JD, Tan X, Li J, Wang X, Lee TH, Jin H, Marler B, Guo H, Kissinger JC, Paterson AH.** (2012) *MCScanX*: a toolkit for detection and evolutionary analysis of gene synteny and collinearity. *Nucleic Acids Res*, 40(7): e49.

（2）步骤

Step1：下载大豆和蒺藜苜蓿的蛋白序列及gff文件并过滤成最长转录本，得到Gm.pep，Gm.gff，Mt.pep，Mt.gff。然后将gff转为bed格式后合并：

```
$ python -m jcvi.formats.gff bed Gm.gff --type=mRNA --key=ID > Gm.bed
$ python -m jcvi.formats.gff bed Mt.gff --type=mRNA --key=ID > Mt.bed
$ awk '{print "Gm"$1"\t"$4"\t"$2"\t"$3}' Gm.bed > Gm_vs_Mt.bed
$ awk '{print "Mt"$1"\t"$4"\t"$2"\t"$3}' Mt.bed >> Gm_vs_Mt.bed
```

- 也可以用其他方法转换格式，保证蛋白序列中的ID与bed中一致
- 为防止混淆，在染色体前加上两个字母的缩写表示物种名，并且要避免两个物种存在相同的蛋白ID
- 这里合并后的为非标准的.bed格式：#chr　　gene_id　　start　　end

1.4 分析方法

1.4.2 MCScanX

（2）步骤

Step2：将大豆的蛋白序列对蒺藜苜蓿的蛋白序列进行blastp比对：

```
$ makeblastdb -dbtype prot -in Mt.pep
$ blastp -db Mt.pep -query Gm.pep -evalue 1e-5 -num_threads 32 -outfmt 6 -out Gm_vs_Mt.blast
```

Step3：运行MCScanX：

```
$ MCScanX Gm_vs_Mt
```

- MCScanX会按照前缀读入.gff文件和.blast文件。因此保证两者前缀一致。

重要参数:

- -s：要求一个共线性区域内包含的最小的基因数目，默认是5个。这个值越大，最后得到的共线性结果就越可信，但是共线性区块数量可能越小。对于亲缘关系较远的物种，可以将此值调小，以便得到两个物种之间更多共线性区块
- -w：要求一个共线性区域内相邻基因之间可以间隔的最多其他非共线性基因，默认是5个
- -a：只输出共线性结果（.collinearity file），不输出网页结果

1.4 分析方法

1.4.2 MCScanX

（3）结果解释

将会生成文件Gm_vs_Mt.collinearity和文件夹Gm_vs_Mt.html

Gm_vs_Mt.collinearity文件：

```
############### Parameters ###############
# MATCH_SCORE: 50
# MATCH_SIZE: 5
# GAP_PENALTY: -1
# OVERLAP_WINDOW: 5
# E_VALUE: 1e-05
# MAX GAPS: 25
############### Statistics ###############
# Number of collinear genes: 57579, Percentage: 54.59
# Number of all genes: 105485
##########################################
## Alignment 0: score=1017.0 e_value=1.7e-69 N=23 GmChr01&Mtchr1 plus
 0-  0:        Glyma.01G232800.1.Wm82.a2.v1    Medtr1g103160.2.JCVIMt4.0v1          0
 0-  1:        Glyma.01G234900.1.Wm82.a2.v1    Medtr1g103420.1.JCVIMt4.0v1       2e-06
 0-  2:        Glyma.01G235100.1.Wm82.a2.v1    Medtr1g103490.1.JCVIMt4.0v1       1e-06
 0-  3:        Glyma.01G236800.1.Wm82.a2.v1    Medtr1g103500.3.JCVIMt4.0v1       2e-89
 0-  4:        Glyma.01G237000.1.Wm82.a2.v1    Medtr1g103550.1.JCVIMt4.0v1       1e-48
 0-  5:        Glyma.01G237300.1.Wm82.a2.v1    Medtr1g103570.1.JCVIMt4.0v1       2e-47
 0-  6:        Glyma.01G237400.1.Wm82.a2.v1    Medtr1g103600.1.JCVIMt4.0v1      1e-146
 0-  7:        Glyma.01G237900.1.Wm82.a2.v1    Medtr1g103690.1.JCVIMt4.0v1       9e-41
 0-  8:        Glyma.01G238200.1.Wm82.a2.v1    Medtr1g103830.1.JCVIMt4.0v1          0
 0-  9:        Glyma.01G238400.1.Wm82.a2.v1    Medtr1g104500.2.JCVIMt4.0v1      1e-172
 0- 10:        Glyma.01G238500.1.Wm82.a2.v1    Medtr1g104520.1.JCVIMt4.0v1       7e-35
 0- 11:        Glyma.01G238700.1.Wm82.a2.v1    Medtr1g104680.1.JCVIMt4.0v1          0
 0- 12:        Glyma.01G238800.1.Wm82.a2.v1    Medtr1g104750.1.JCVIMt4.0v1          0
 0- 13:        Glyma.01G238900.1.Wm82.a2.v1    Medtr1g104800.1.JCVIMt4.0v1       2e-82
```

格式与mcscan的.aligns格式一致，包含各个共线性区块与基因对。

- ## 开头的为一个alignment。此外还包括score，evalue，基因对数量，染色体，比对方向等信息。
- 第1列为alignment编号
- 第2列为基因对编号
- 第3列、第4列分别为一对基因。
- 第5列为blast比对的evalue

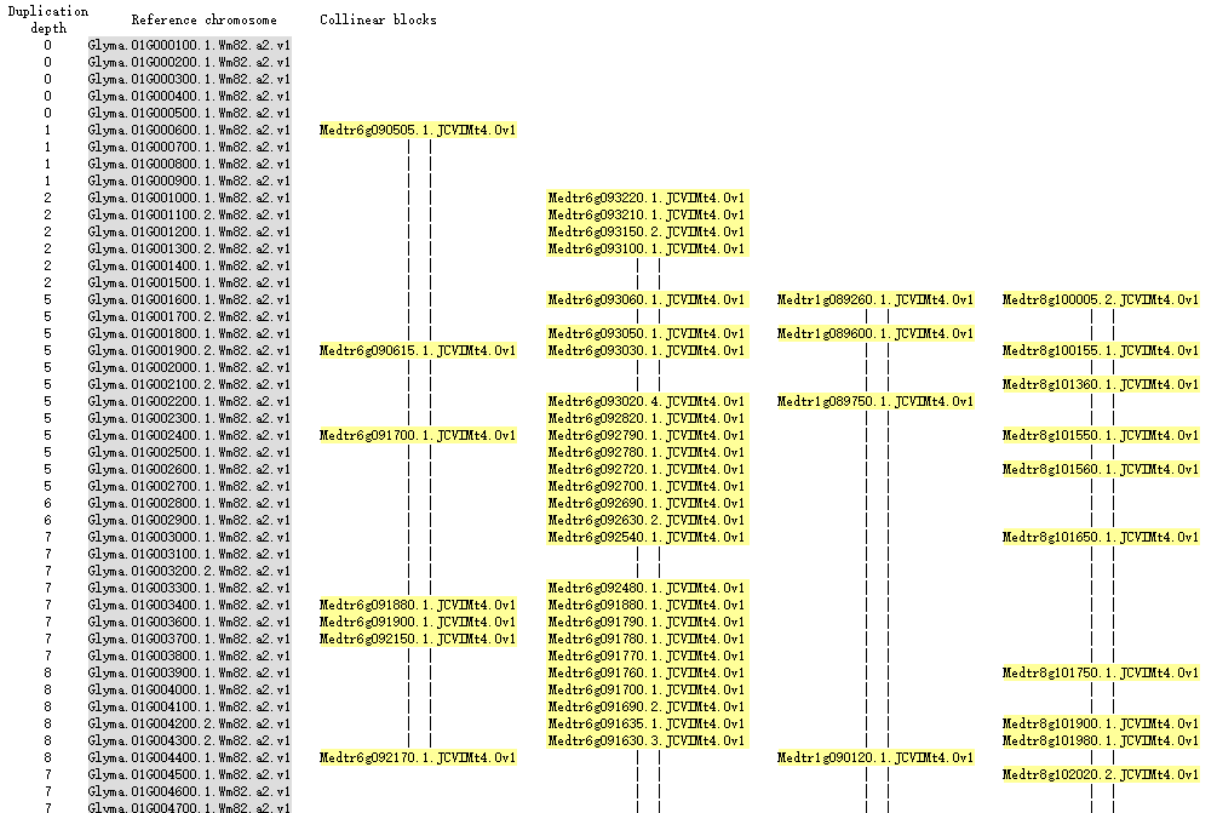
1.4 分析方法

1.4.2 MCScanX

（3）结果解释

将会生成文件Gm_vs_Mt.collinearity和文件夹Gm_vs_Mt.html

Gm_vs_Mt.html目录下的网页文件：



Duplication depth	Reference chromosome	Collinear blocks			
0	Glyma.01G000100.1.Wm82.a2.v1				
0	Glyma.01G000200.1.Wm82.a2.v1				
0	Glyma.01G000300.1.Wm82.a2.v1				
0	Glyma.01G000400.1.Wm82.a2.v1				
0	Glyma.01G000500.1.Wm82.a2.v1				
1	Glyma.01G000600.1.Wm82.a2.v1	Medtr6g090505.1.JCVIMt4.0v1			
1	Glyma.01G000700.1.Wm82.a2.v1				
1	Glyma.01G000800.1.Wm82.a2.v1				
1	Glyma.01G000900.1.Wm82.a2.v1				
2	Glyma.01G001000.1.Wm82.a2.v1		Medtr6g093220.1.JCVIMt4.0v1		
2	Glyma.01G001100.2.Wm82.a2.v1		Medtr6g093210.1.JCVIMt4.0v1		
2	Glyma.01G001200.1.Wm82.a2.v1		Medtr6g093150.2.JCVIMt4.0v1		
2	Glyma.01G001300.2.Wm82.a2.v1		Medtr6g093100.1.JCVIMt4.0v1		
2	Glyma.01G001400.1.Wm82.a2.v1				
2	Glyma.01G001500.1.Wm82.a2.v1				
5	Glyma.01G001600.1.Wm82.a2.v1		Medtr6g093060.1.JCVIMt4.0v1	Medtr1g089260.1.JCVIMt4.0v1	Medtr8g100005.2.JCVIMt4.0v1
5	Glyma.01G001700.2.Wm82.a2.v1				
5	Glyma.01G001800.1.Wm82.a2.v1		Medtr6g093050.1.JCVIMt4.0v1	Medtr1g089600.1.JCVIMt4.0v1	
5	Glyma.01G001900.2.Wm82.a2.v1	Medtr6g090615.1.JCVIMt4.0v1	Medtr6g093030.1.JCVIMt4.0v1		Medtr8g100155.1.JCVIMt4.0v1
5	Glyma.01G002000.1.Wm82.a2.v1				
5	Glyma.01G002100.2.Wm82.a2.v1				Medtr8g101360.1.JCVIMt4.0v1
5	Glyma.01G002200.1.Wm82.a2.v1		Medtr6g093020.4.JCVIMt4.0v1	Medtr1g089750.1.JCVIMt4.0v1	
5	Glyma.01G002300.1.Wm82.a2.v1		Medtr6g092820.1.JCVIMt4.0v1		
5	Glyma.01G002400.1.Wm82.a2.v1	Medtr6g091700.1.JCVIMt4.0v1	Medtr6g092790.1.JCVIMt4.0v1		Medtr8g101550.1.JCVIMt4.0v1
5	Glyma.01G002500.1.Wm82.a2.v1		Medtr6g092780.1.JCVIMt4.0v1		
5	Glyma.01G002600.1.Wm82.a2.v1		Medtr6g092720.1.JCVIMt4.0v1		Medtr8g101560.1.JCVIMt4.0v1
5	Glyma.01G002700.1.Wm82.a2.v1		Medtr6g092700.1.JCVIMt4.0v1		
6	Glyma.01G002800.1.Wm82.a2.v1		Medtr6g092690.1.JCVIMt4.0v1		
6	Glyma.01G002900.1.Wm82.a2.v1		Medtr6g092630.2.JCVIMt4.0v1		
7	Glyma.01G003000.1.Wm82.a2.v1		Medtr6g092540.1.JCVIMt4.0v1		Medtr8g101650.1.JCVIMt4.0v1
7	Glyma.01G003100.1.Wm82.a2.v1				
7	Glyma.01G003200.2.Wm82.a2.v1				
7	Glyma.01G003300.1.Wm82.a2.v1		Medtr6g092480.1.JCVIMt4.0v1		
7	Glyma.01G003400.1.Wm82.a2.v1	Medtr6g091880.1.JCVIMt4.0v1	Medtr6g091880.1.JCVIMt4.0v1		
7	Glyma.01G003600.1.Wm82.a2.v1	Medtr6g091900.1.JCVIMt4.0v1	Medtr6g091790.1.JCVIMt4.0v1		
7	Glyma.01G003700.1.Wm82.a2.v1	Medtr6g092150.1.JCVIMt4.0v1	Medtr6g091780.1.JCVIMt4.0v1		
7	Glyma.01G003800.1.Wm82.a2.v1		Medtr6g091770.1.JCVIMt4.0v1		
8	Glyma.01G003900.1.Wm82.a2.v1		Medtr6g091760.1.JCVIMt4.0v1		Medtr8g101750.1.JCVIMt4.0v1
8	Glyma.01G004000.1.Wm82.a2.v1		Medtr6g091700.1.JCVIMt4.0v1		
8	Glyma.01G004100.1.Wm82.a2.v1		Medtr6g091690.2.JCVIMt4.0v1		
8	Glyma.01G004200.2.Wm82.a2.v1		Medtr6g091635.1.JCVIMt4.0v1		Medtr8g101900.1.JCVIMt4.0v1
8	Glyma.01G004300.2.Wm82.a2.v1		Medtr6g091630.3.JCVIMt4.0v1		Medtr8g101980.1.JCVIMt4.0v1
8	Glyma.01G004400.1.Wm82.a2.v1	Medtr6g092170.1.JCVIMt4.0v1		Medtr1g090120.1.JCVIMt4.0v1	
7	Glyma.01G004500.1.Wm82.a2.v1				Medtr8g102020.2.JCVIMt4.0v1
7	Glyma.01G004600.1.Wm82.a2.v1				
7	Glyma.01G004700.1.Wm82.a2.v1				

Gm_vs_Mt.html中的网页展示了多个区块比对的结果。

- 第1列为比对深度
- 第2列为reference上的基因
- 后面几列为比对上的共线性区块。

相比于mcscan的.blocks文件，更清晰地将同一染色体上的blocks分开。此外不显示没有比对上的基因。

1.4 分析方法

1.4.2 MCScanX

（4）绘图：MCScanX内置了若干绘图程序，可用于共线性比对结果的图形展示

① 绘制共线性dotplot:

```
$ java dot_plotter -g Gm_vs_Mt.gff -s Gm_vs_Mt.collinearity -c dot.ctl -o dotplot.png
```

- 此外，也可以对mcscan的结果进行绘图，需要修改.bed格式，.aligns可以直接使用
- dot.ctl为控制文件：

```
800     //dimension (in pixels) of x axis
800     //dimension (in pixels) of y axis
GmChr01,GmChr02,GmChr03,GmChr04,GmChr05,GmChr06,GmChr07,GmChr08,GmChr09,GmChr10,GmC
hr11,GmChr12,GmChr13,GmChr14,GmChr15,GmChr16,GmChr17,GmChr18,GmChr19,GmChr20 //chro
mosomes in x axis
Mtchr1,Mtchr2,Mtchr3,Mtchr4,Mtchr5,Mtchr6,Mtchr7,Mtchr8 //chromosomes in y axis
```

- 第1行和第2行分别为图形x轴和y轴的像素
- 第3行和第4行分别为x轴和y轴展示的染色体

注意:

- //为注释，与前面的参数之间用tab隔开
- 染色体用逗号隔开，且不能有空格

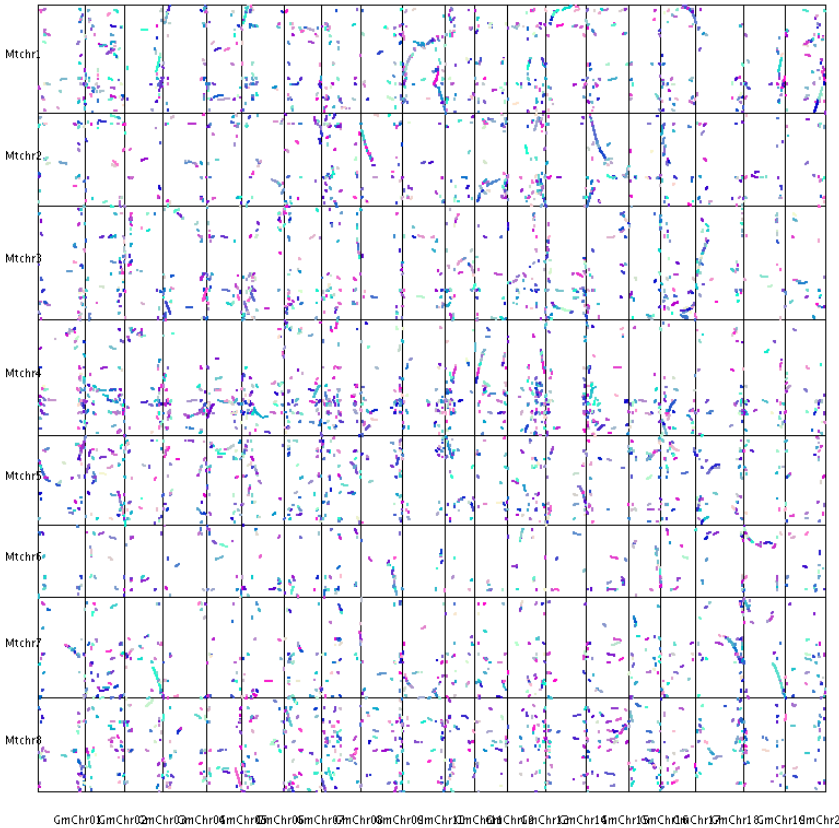
1.4 分析方法

1.4.2 MCScanX

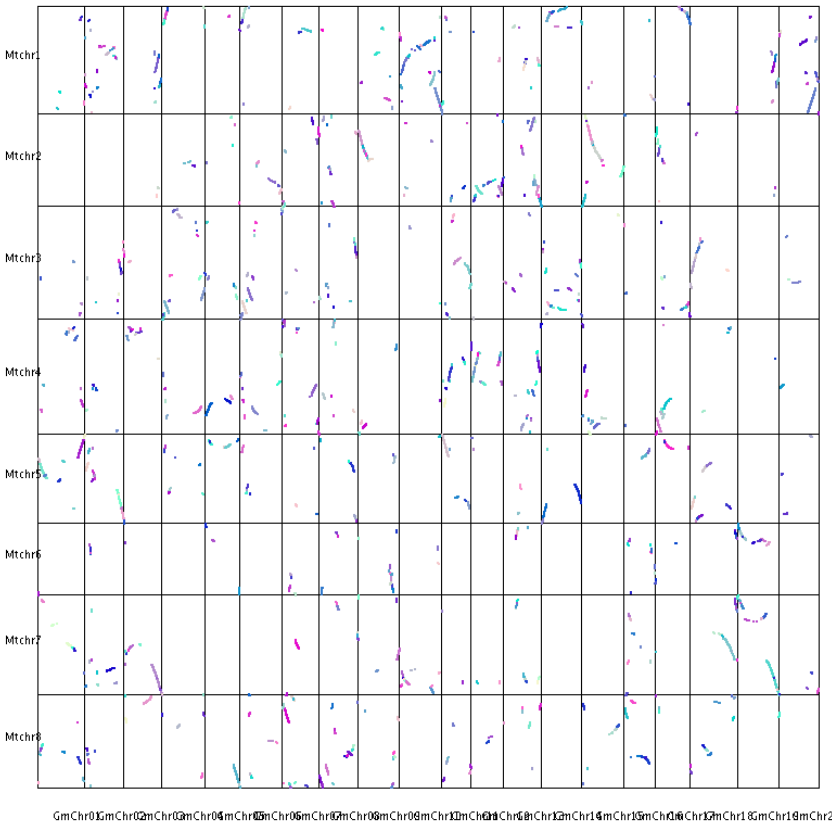
（4）绘图：MCScanX内置了若干绘图程序，可用于共线性比对结果的图形展示

① 绘制共线性dotplot：

MCScanx



mcscan



1.4 分析方法

1.4.2 MCScanX

(4) 绘图

② 绘制共线性dual synteny图:

```
$ java dual_synteny_plotter -g Gm_vs_Mt.gff -s Gm_vs_Mt.collinearity -c dual_synteny.ctl
-o dual_synteny.png
```

- 同样，也可以对mcscan的结果进行绘图，需要修改.bed格式，.aligns可以直接使用
- dual_synteny.ctl为控制文件

```
600     //plot width (in pixels)
2000    //plot height (in pixels)
GmChr01,GmChr02,GmChr03,GmChr04,GmChr05,GmChr06,GmChr07,GmChr08,GmChr09,GmChr10,GmChr11,GmChr
12,GmChr13,GmChr14,GmChr15,GmChr16,GmChr17,GmChr18,GmChr19,GmChr20 //chromosomes in the left
column
Mtchr1,Mtchr2,Mtchr3,Mtchr4,Mtchr5,Mtchr6,Mtchr7,Mtchr8 //chromosomes in the right column
```

- 第1行和第2行分别为图形宽和高的像素
- 第3行和第4行分别为左侧和右侧展示的染色体

注意:

- //为注释，与前面的参数之间用tab隔开
- 染色体用逗号隔开，且不能有空格

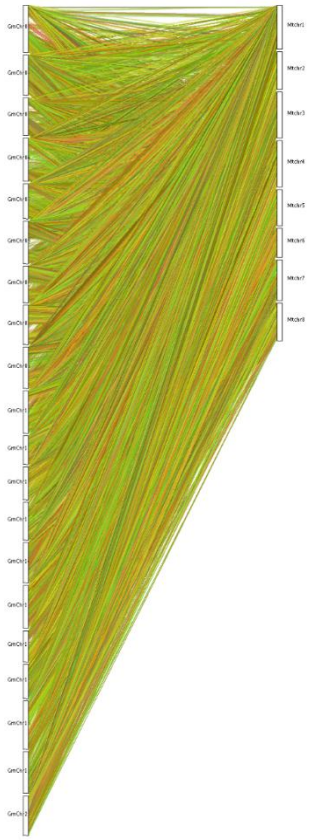
1.4 分析方法

1.4.2 MCScanX

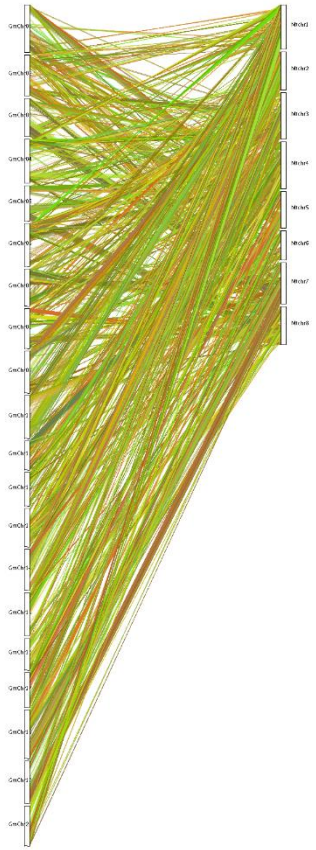
(4) 绘图

② 绘制共线性dual synteny图：

MCScanx



mcscan



1.4 分析方法

1.4.2 MCScanX

(4) 绘图

③ 绘制共线性circle图:

```
$ java circle_plotter -g Gm_vs_Mt.gff -s Gm_vs_Mt.collinearity -c circle.ctl -o circle.png
```

- 同样，也可以对mcscan的结果进行绘图，需要修改.bed格式，.aligns可以直接使用
- circle.ctl为控制文件:

```
800     //plot width and height (in pixels)
GmChr01,GmChr02,GmChr03,GmChr04,GmChr05,GmChr06,GmChr07,GmChr08,GmChr09,GmCh
r10,GmChr11,GmChr12,GmChr13,GmChr14,GmChr15,GmChr16,GmChr17,GmChr18,GmChr19,
GmChr20,Mtchr1,Mtchr2,Mtchr3,Mtchr4,Mtchr5,Mtchr6,Mtchr7,Mtchr8 //chromosome
s in the circle
```

- 第1行为图宽和高的像素
- 第2行为染色体排列顺序，从3点钟方向，逆时针排列
- 注意:
- //为注释，与前面的参数之间用tab隔开
- 染色体用逗号隔开，且不能有空格

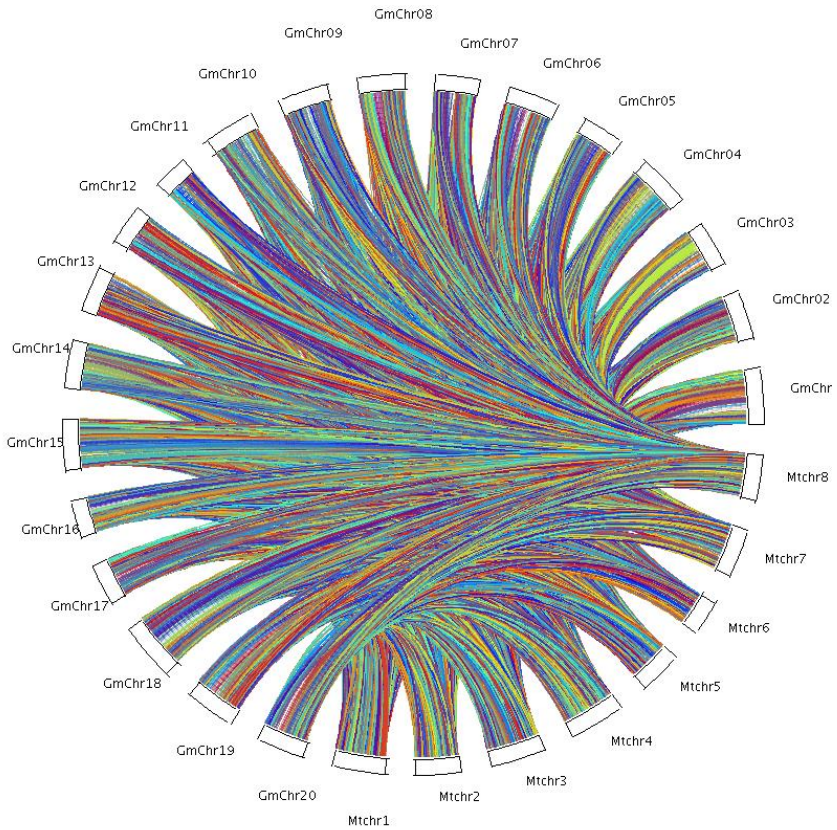
1.4 分析方法

1.4.2 MCScanX

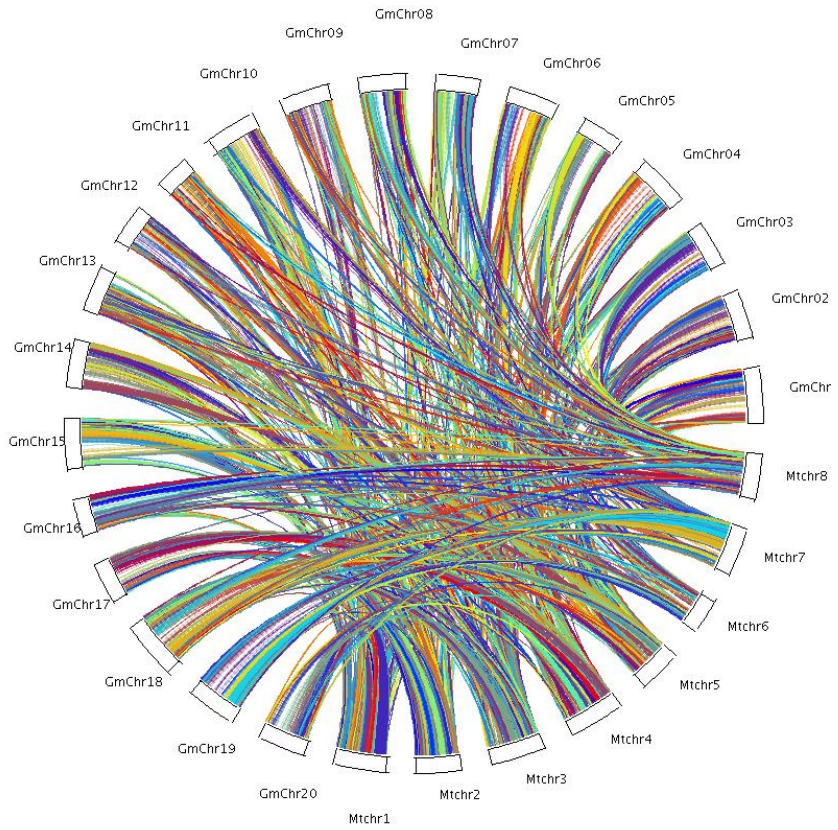
(4) 绘图

③ 绘制共线性circle图：

MCScanx



mcscan



1.4 分析方法

1.4.2 MCScanX

（4）绘图

④ 绘制染色体条形图（可用于古染色体分析）：

```
$ java bar_plotter -g Gm_vs_Mt.gff -s Gm_vs_Mt.aligns -c bar.ctl -o bar.png
```

- 同样，也可以对mcscan的结果进行绘图，需要修改.bed格式，.aligns可以直接使用
- bar.ctl为控制文件：

```
800     //dimension (in pixels) of x axis
800     //dimension (in pixels) of y axis
Mtchr1,Mtchr2,Mtchr3,Mtchr4,Mtchr5,Mtchr6,Mtchr7,Mtchr8 //reference chromosomes
GmChr01,GmChr02,GmChr03,GmChr04,GmChr05,GmChr06,GmChr07,GmChr08,GmChr09,GmChr10,GmChr11,GmChr12,
GmChr13,GmChr14,GmChr15,GmChr16,GmChr17,GmChr18,GmChr19,GmChr20 //target chromosomes
```

- 第1行和第2行分别为图形x轴和y轴的像素
- 第3行和第4行分别为reference和target基因组染色体

注意：

- //为注释，与前面的参数之间用tab隔开
- 染色体用逗号隔开，且不能有空格

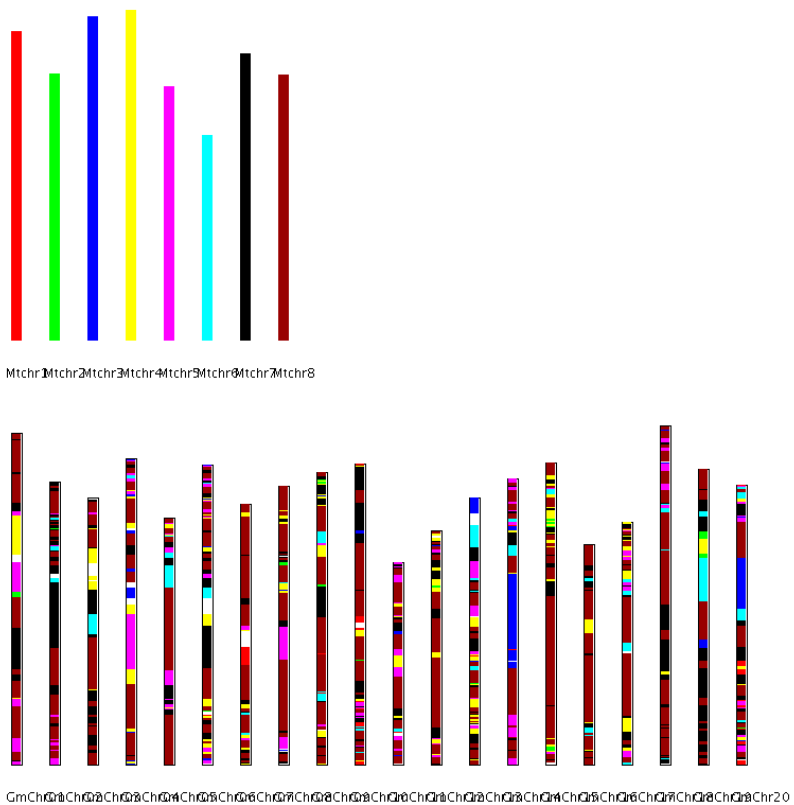
1.4 分析方法

1.4.2 MCScanX

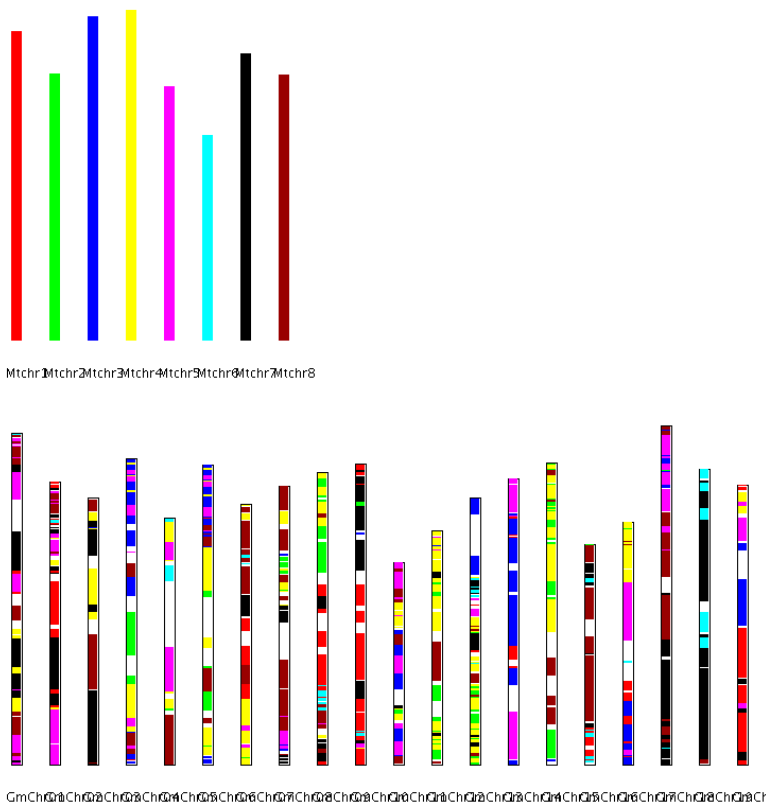
（4）绘图

④ 绘制染色体条形图（可用于古染色体分析）：

MCScanx



mcscan



1.4 分析方法

1.4.3 mcscan in JCVI package (Python version)

（1）references

- https://github.com/tanghaibao/jcvi
- **Haibao Tang et al**. (2015). jcvi: JCVI utility libraries. Zenodo. 10.5281/zenodo.31631.

（2）步骤

Step1：下载大豆和蒺藜苜蓿的蛋白序列及gff文件并过滤成最长转录本，得到Gm.pep，Gm.gff，Mt.pep，Mt.gff。然后将gff转为bed格式后合并：

```
$ python -m jcvi.formats.gff bed Gm.gff --type=mRNA --key=ID > Gm.bed
$ python -m jcvi.formats.gff bed Mt.gff --type=mRNA --key=ID > Mt.bed
```

Step2：将大豆的蛋白序列对蒺藜苜蓿的蛋白序列进行blastp比对：

```
$ makeblastdb -dbtype prot -in Mt.pep
$ blastp -db Mt.pep -query Gm.pep -evalue 1e-5 -num_threads 32 -outfmt 6 -out Gm.Mt.last
```

- 该版mcscan默认调用LAST并进行cds序列的比对，如果需要使用蛋白，必须要自己手动进行蛋白序列的比对，并将文件名命名为query.subject.last

1.4 分析方法

1.4.3 mcscan in JCVI package (Python version)

（2）步骤

Step3：将.bed文件和.last文件中所有基因ID中的”.”替换为”_”

- 该版本mcscan会将最后一个”.”当作可变剪接，在后续分析中会自动去掉，引发bug

Step4：进行共线性比对并生成共线性dotplot：

```
$ python -m jcvi.compara.catalog ortholog Gm Mt
# 使用 python –m jcvi.graphics.dotplot Gm.Mt.anchors可获得更多设置
```

- 该版mcscan引入了cscore和quota两个参数，可以按照自己的需求进行过滤
- .last.filtered：按照cscore和quota过滤后的比对文件）

  .anchors：##代表共线性区块的开始，第1、2列为基因对，第三列为bitscore

  .lifted.anchors：格式同.anchors，将共线性区块往周围延伸后的结果

  .pdf：共线性dotplot

1.4 分析方法

1.4.3 mcscan in JCVI package (Python version)

(2) 步骤

Step5：绘制karyotype图：

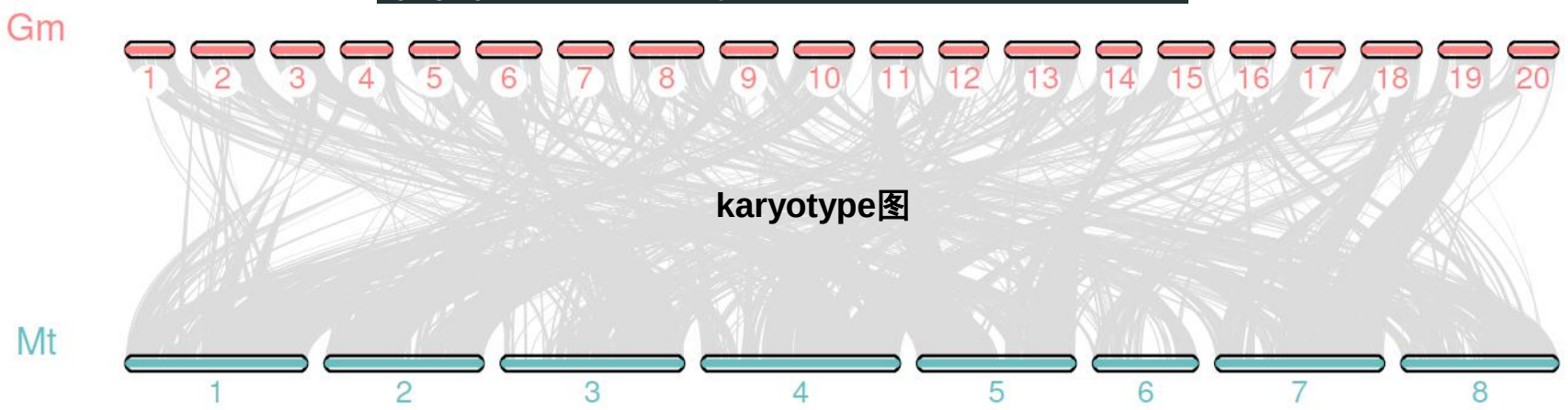
```
$ python -m jcvi.compara.synteny screen --minspan=30 --simple Gm.Mt.anchors Gm.Mt.anchors.simple
$ python -m jcvi.graphics.karyotype seqids layout
```

seqids

```
Chr01,Chr02,Chr03,Chr04,Chr05,Chr06,Chr07,Chr08,Chr09,Chr10,Chr11,Chr12,Chr13,Chr14,Chr15,Chr16,Chr17,Chr18,Chr19,Chr20
chr1,chr2,chr3,chr4,chr5,chr6,chr7,chr8
```

layout

```
# y, xstart, xend, rotation, color, label,  va,  bed
 .6,     .1,   .9,        0,  #ff8484,   Gm, bottom,  Gm.bed
 .4,     .1,   .9,        0,  #65c0c3,  Mt, bottom, Mt.bed
# edges
e, 0, 1, Gm.Mt.anchors.simple
```



karyotype图

1.4 分析方法

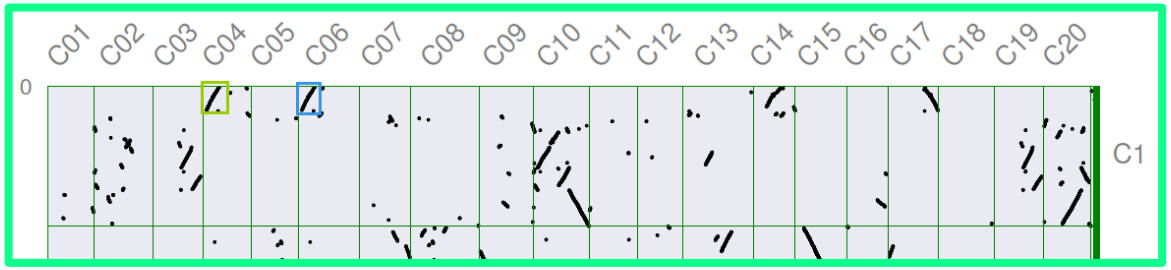
1.4.3 mcscan in JCVI package (Python version)

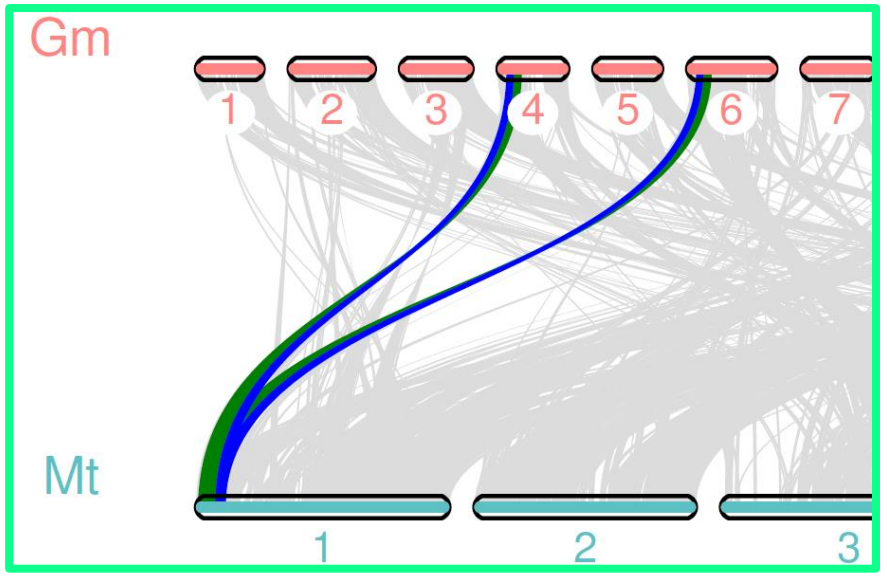
（2）步骤

Step5：绘制karyotype图：

高亮展示：编辑.simple文件，选择highlight展示的共线性blocks

Glyma_06G084300_2_Wm82_a2_v1	Glyma_06G090200_1_Wm82_a2_v1	Medtr1g004990_1_JCVIMt4_0v1	Medtr1g007380_1_JCVIMt4_0v1	54	+
Glyma_04G082900_1_Wm82_a2_v1	Glyma_04G088300_2_Wm82_a2_v1	Medtr1g004990_1_JCVIMt4_0v1	Medtr1g007420_1_JCVIMt4_0v1	55	+
Glyma_17G191300_1_Wm82_a2_v1	Glyma_17G195900_1_Wm82_a2_v1	Medtr1g004990_1_JCVIMt4_0v1	Medtr1g007580_1_JCVIMt4_0v1	53	-
Glyma_04G231700_3_Wm82_a2_v1	Glyma_04G236500_1_Wm82_a2_v1	Medtr1g007580_1_JCVIMt4_0v1	Medtr1g008230_1_JCVIMt4_0v1	45	+
Glyma_06G084300_2_Wm82_a2_v1	Glyma_06G090200_1_Wm82_a2_v1	Medtr1g004990_1_JCVIMt4_0v1	Medtr1g007380_1_JCVIMt4_0v1	54	+
Glyma_04G082900_1_Wm82_a2_v1	Glyma_04G088300_2_Wm82_a2_v1	Medtr1g004990_1_JCVIMt4_0v1	Medtr1g007420_1_JCVIMt4_0v1	55	+
Glyma_17G191300_1_Wm82_a2_v1	Glyma_17G195900_1_Wm82_a2_v1	Medtr1g004990_1_JCVIMt4_0v1	Medtr1g007580_1_JCVIMt4_0v1	53	-
Glyma_04G231700_3_Wm82_a2_v1	Glyma_04G236500_1_Wm82_a2_v1	Medtr1g007580_1_JCVIMt4_0v1	Medtr1g008230_1_JCVIMt4_0v1	45	+
Glyma_06G127800_1_Wm82_a2_v1	Glyma_06G133400_1_Wm82_a2_v1	Medtr1g007580_1_JCVIMt4_0v1	Medtr1g008230_1_JCVIMt4_0v1	48	-
Glyma_14G126200_1_Wm82_a2_v1	Glyma_14G142100_2_Wm82_a2_v1	Medtr1g008280_1_JCVIMt4_0v1	Medtr1g011800_1_JCVIMt4_0v1	165	-
g*Glyma_04G053100_1_Wm82_a2_v1	Glyma_04G080200_2_Wm82_a2_v1	Medtr1g008280_1_JCVIMt4_0v1	Medtr1g015750_1_JCVIMt4_0v1	326	-
g*Glyma_06G054000_1_Wm82_a2_v1	Glyma_06G081900_1_Wm82_a2_v1	Medtr1g008280_1_JCVIMt4_0v1	Medtr1g015750_1_JCVIMt4_0v1	331	-
Glyma_17G194400_1_Wm82_a2_v1	Glyma_17G218500_1_Wm82_a2_v1	Medtr1g009190_1_JCVIMt4_0v1	Medtr1g014240_1_JCVIMt4_0v1	229	+
Glyma_14G109000_2_Wm82_a2_v1	Glyma_14G116800_1_Wm82_a2_v1	Medtr1g012520_1_JCVIMt4_0v1	Medtr1g014260_1_JCVIMt4_0v1	67	-
Glyma_17G218700_4_Wm82_a2_v1	Glyma_17G224300_1_Wm82_a2_v1	Medtr1g014240_1_JCVIMt4_0v1	Medtr1g015110_1_JCVIMt4_0v1	68	-
Glyma_14G100600_1_Wm82_a2_v1	Glyma_14G108400_2_Wm82_a2_v1	Medtr1g014240_1_JCVIMt4_0v1	Medtr1g015110_1_JCVIMt4_0v1	79	+
Glyma_17G224700_1_Wm82_a2_v1	Glyma_17G232200_1_Wm82_a2_v1	Medtr1g015120_1_JCVIMt4_0v1	Medtr1g017020_2_JCVIMt4_0v1	99	+
Glyma_14G089200_1_Wm82_a2_v1	Glyma_14G100200_1_Wm82_a2_v1	Medtr1g015120_1_JCVIMt4_0v1	Medtr1g017400_1_JCVIMt4_0v1	129	-
b*Glyma_06G041000_2_Wm82_a2_v1	Glyma_06G051800_1_Wm82_a2_v1	Medtr1g016480_1_JCVIMt4_0v1	Medtr1g019200_2_JCVIMt4_0v1	131	-
b*Glyma_04G039800_2_Wm82_a2_v1	Glyma_04G050900_1_Wm82_a2_v1	Medtr1g016480_1_JCVIMt4_0v1	Medtr1g019200_2_JCVIMt4_0v1	132	-
Glyma_17G232300_1_Wm82_a2_v1	Glyma_17G262500_1_Wm82_a2_v1	Medtr1g017450_1_JCVIMt4_0v1	Medtr1g026560_1_JCVIMt4_0v1	399	+
Glyma_14G070000_1_Wm82_a2_v1	Glyma_14G089000_1_Wm82_a2_v1	Medtr1g018320_1_JCVIMt4_0v1	Medtr1g023170_1_JCVIMt4_0v1	244	-
Glyma_04G026200_1_Wm82_a2_v1	Glyma_04G035700_2_Wm82_a2_v1	Medtr1g021520_1_JCVIMt4_0v1	Medtr1g023690_1_JCVIMt4_0v1	125	-
Glyma_06G025800_3_Wm82_a2_v1	Glyma_06G034600_1_Wm82_a2_v1	Medtr1g021855_1_JCVIMt4_0v1	Medtr1g023760_1_JCVIMt4_0v1	112	-
Glyma_14G217400_1_Wm82_a2_v1	Glyma_14G224200_1_Wm82_a2_v1	Medtr1g023985_1_JCVIMt4_0v1	Medtr1g026560_1_JCVIMt4_0v1	90	+
Glyma_04G019500_1_Wm82_a2_v1	Glyma_04G025700_1_Wm82_a2_v1	Medtr1g024005_1_JCVIMt4_0v1	Medtr1g026410_1_JCVIMt4_0v1	83	-





- 可以看到，苜蓿1号染色体左端，比对到大豆4号和6号染色体的左端，展示了二倍化WGD事件。

1.4 分析方法

各方法特点

软件	mcscan	MCScanX	JCVI
安装	简单	简单	依赖程序较多
速度	较慢	较慢	快
anchors	26,738	77,288	40,602 / 82,786
blocks	835	5,275	1,442
anchors/block	32	14	28 / 57
绘图	无法绘图	非矢量图	矢量图，可绘制dotplot、karyotype等多种图型
优点	简单易用	简单易用	引入了quota，cscore等过滤参数；功能强大

PART 02

WGD分析

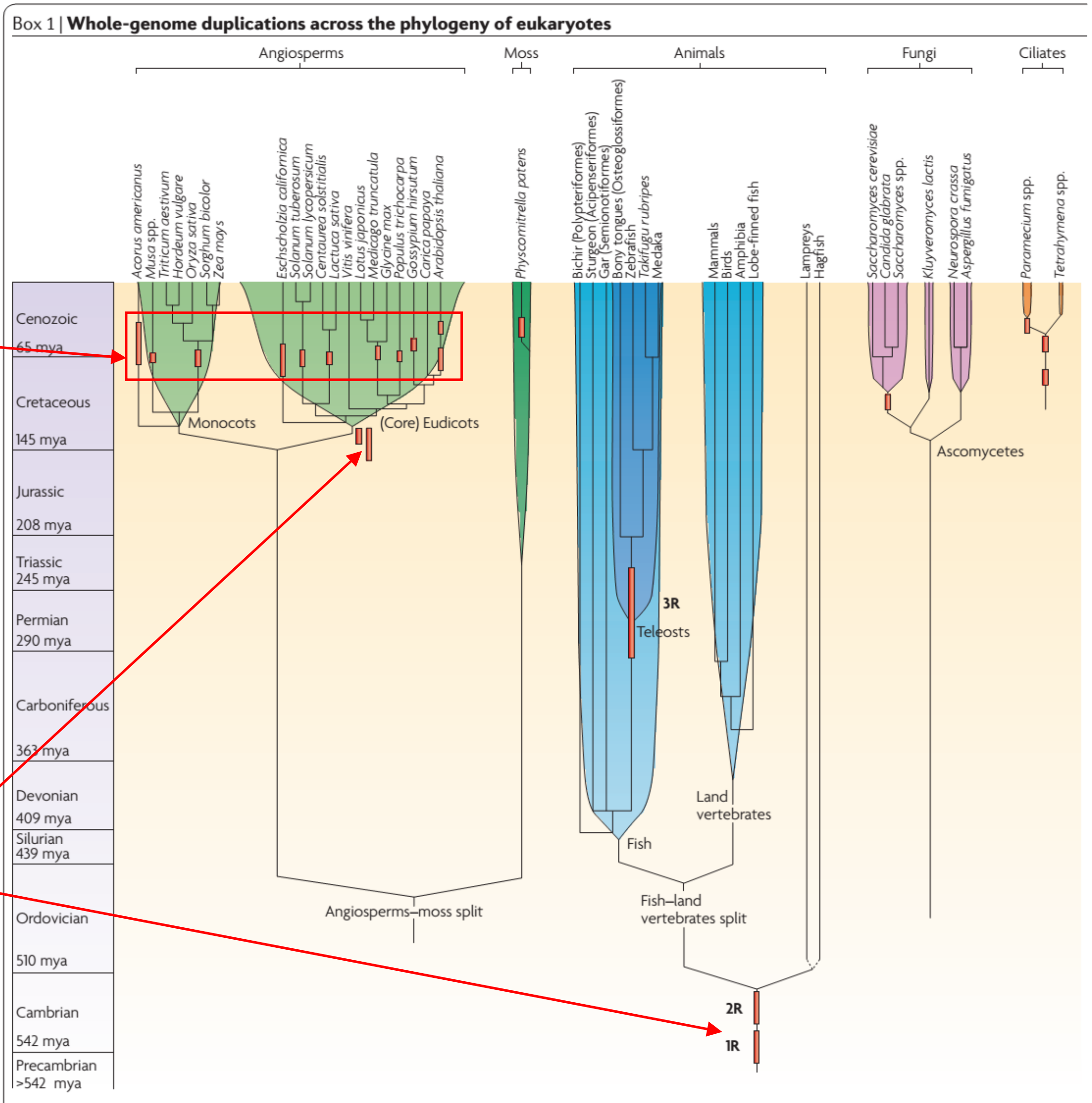
ENTER YOUR SUBTITLE

2.1 WGD背景

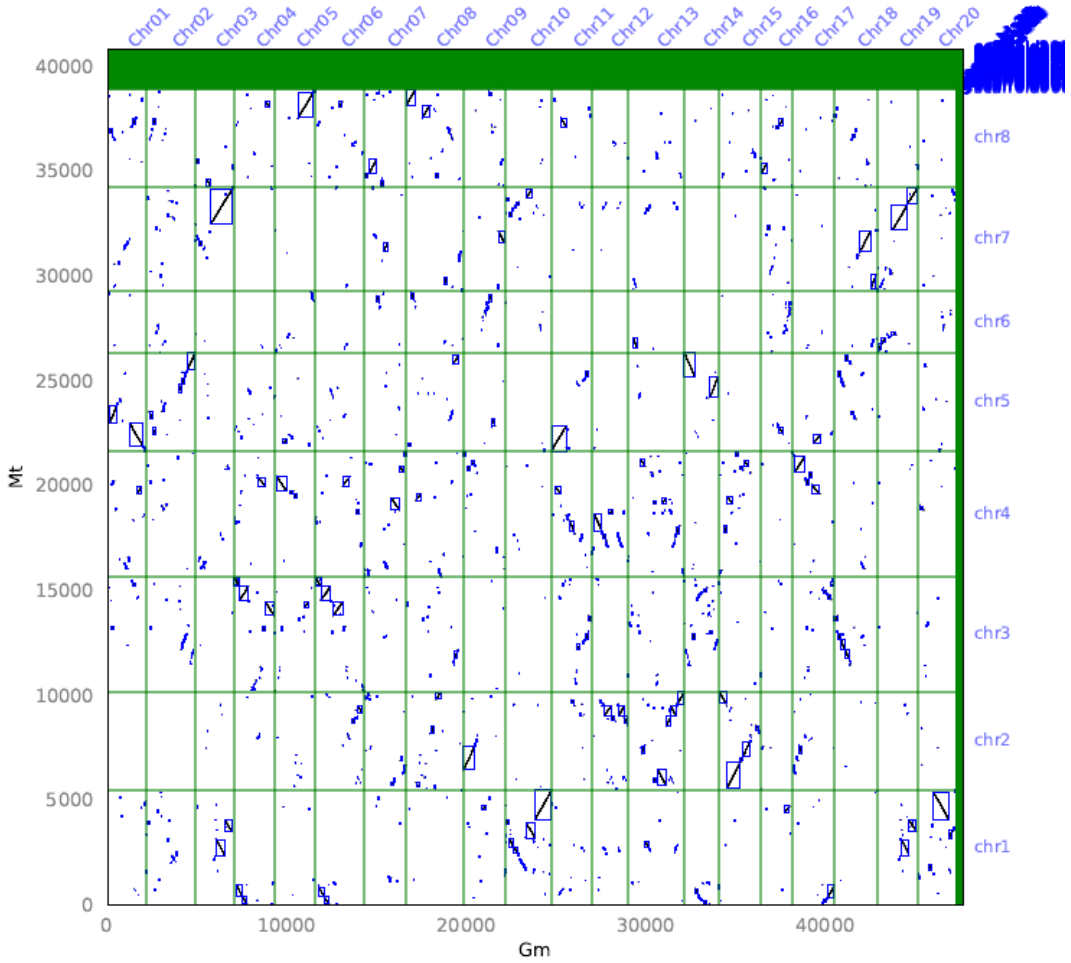
多倍体的竞争优势	增加物种多样性	创新性进化
• 减少物种灭绝的风险 • 增加生命力	• 互补基因的丢失 • 亚功能化 • 物种分化	• 基因组复制促进基因保留 • 增加物种的复杂性

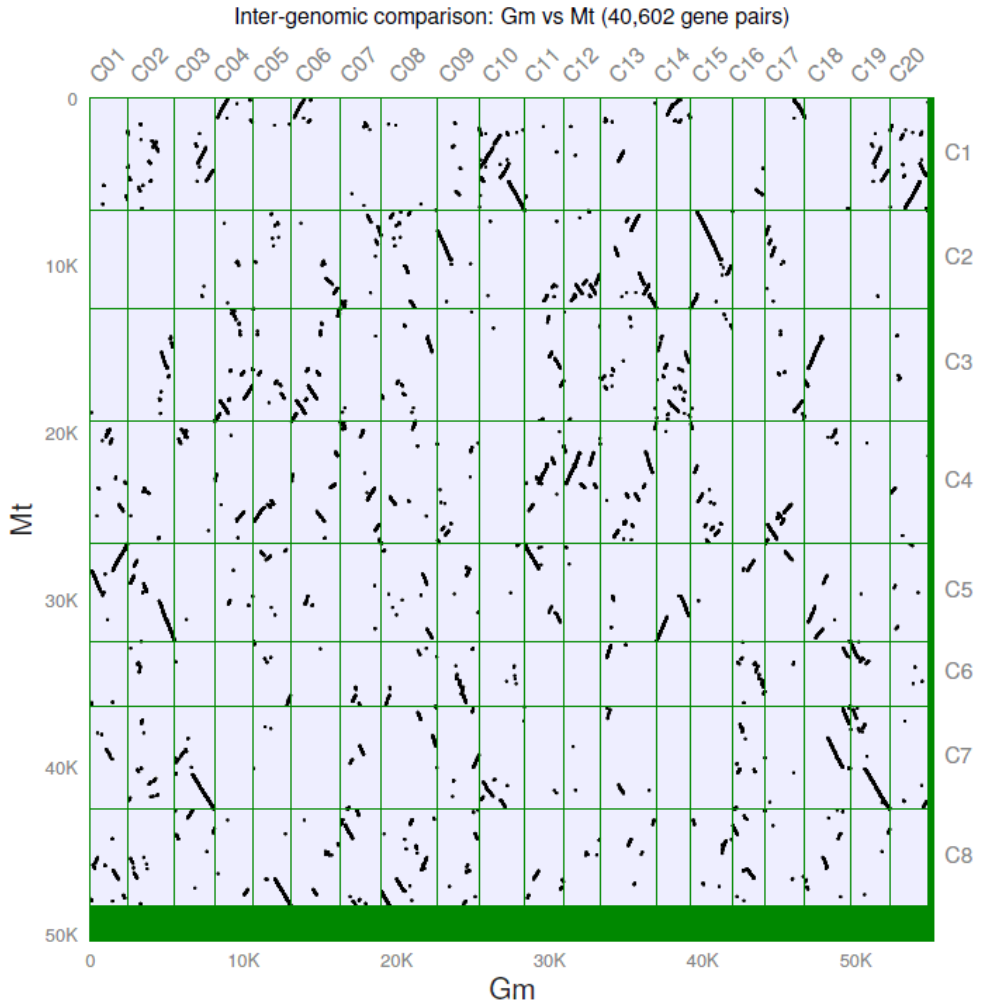
基因组复制可以减少物种灭绝的风险，最为明显的证据为：基因组复制帮助开花植物避免了物种的灭绝。例如豆科、谷类、茄科、生菜和棉花等植物在约60-70百万年前独立进行了[illegible]WGD。这一时间与"K-T灭绝界线"时间相接近，表明[illegible] [illegible]纪早期的WGD之后，出现闭合的心皮、花和双受精

右图显示的是真核生物进化过程中的WGD与WGD分布[illegible]的复杂性大幅度增加，例如脊椎动物两轮基因组复制的1R和2R之后[illegible]尽管多倍化可能[illegible]



2.2 直接通过共线性dotplot





通过观察dotplot，可以直接判断大豆vs苜蓿的WGD倍性为4：2

- 优点：可判断WGD倍性，直观
- 缺点：需要染色体结构相对保守（近缘），组装到染色体水平

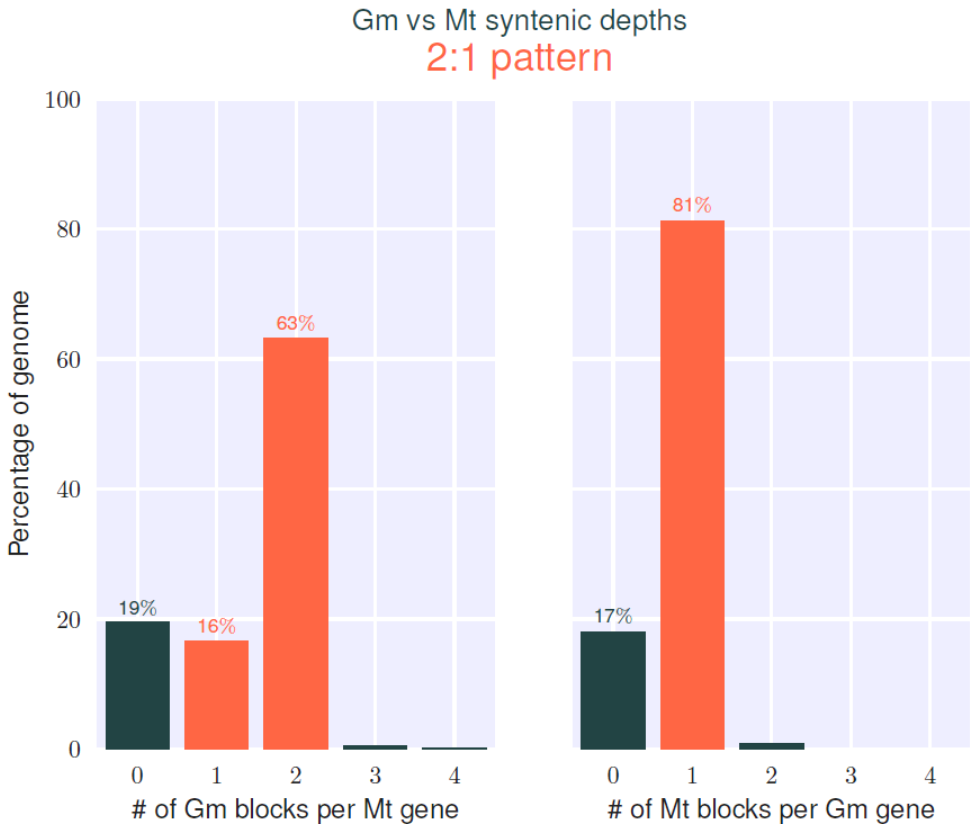
2.3 使用JCVI进行WGD分析

Step1. 从JCVI的第4步开始，通过cscore=.99过滤，得到直系同源比对结果

```
$ python -m jcvi.compara.catalog ortholog Gm Mt --cscore=.99
```

Step2. 统计每个基因，对应共线性区块的数量，以及占基因组的比值，得到WGD倍性的比值

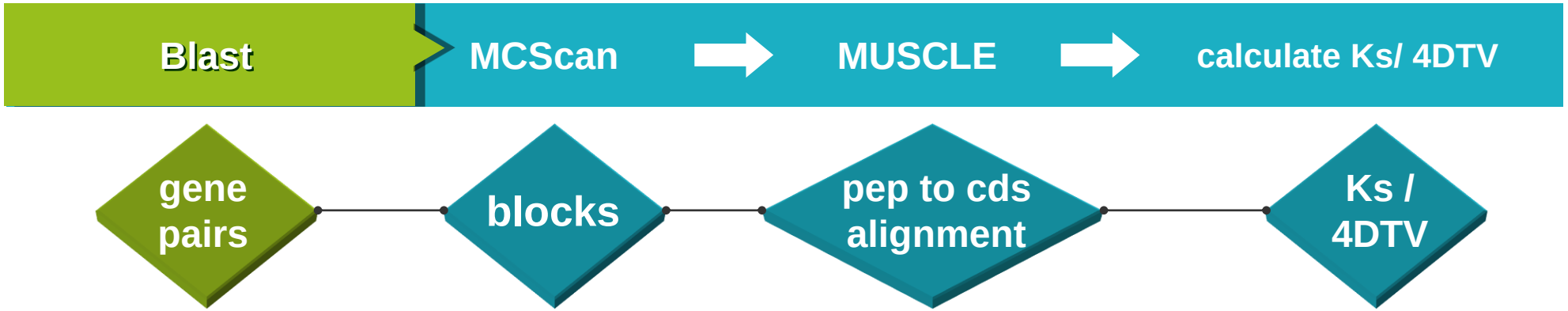
```
$ python -m jcvi.compara.synteny depth –histogram Gm.Mt.anchors
```



- 得到WGD倍性的比值为2：1
- **优点**：可直接得到WGD倍性，并生成统计的histogram。不需要组装到染色体水平，对染色体结构保守性要求较低。
- **缺点**：在高WGD倍性，以及亲缘关系过远时，分析结果可能有误。

2.4 计算anchors（共线性基因对）Ks，4dtv

(1) WGD分析思路

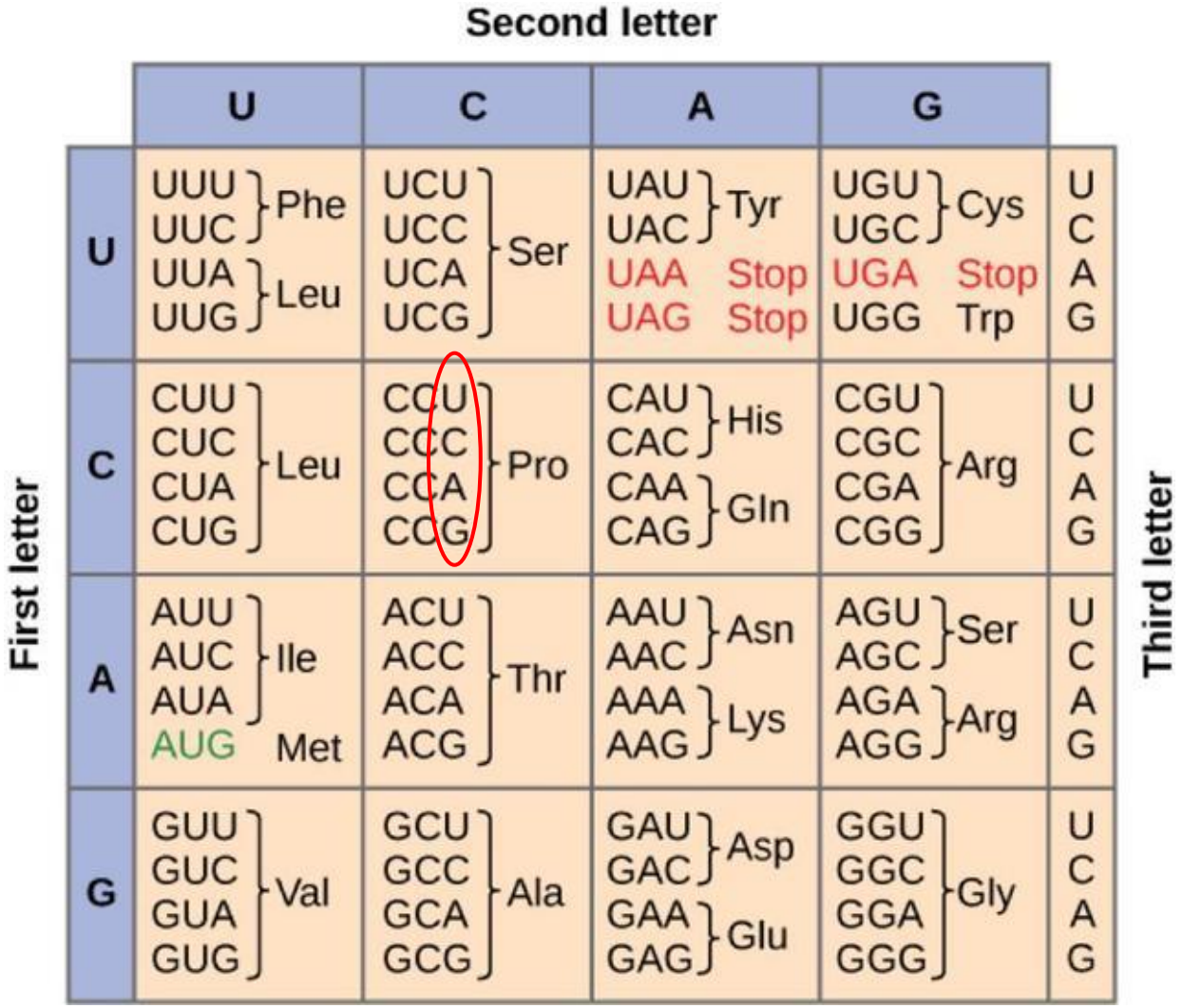


两个物种的所有蛋白序列进行Blast比对，找到同源基因，根据同源基因在染色体上的排布关系，用MCScan/ MCScanX/JCVI软件找出共线性区段，将区段内的基因进行muscle比对，把蛋白muscle矩阵转化为cds muscle矩阵，计算Ks值或者4DTV值，最后画图呈现结果

2.4 计算anchors（共线性基因对）Ks，4dtv

(2) 4dtv的概念

密码子的简并性：一个氨基酸由一个以上的三联体密码编码的现象

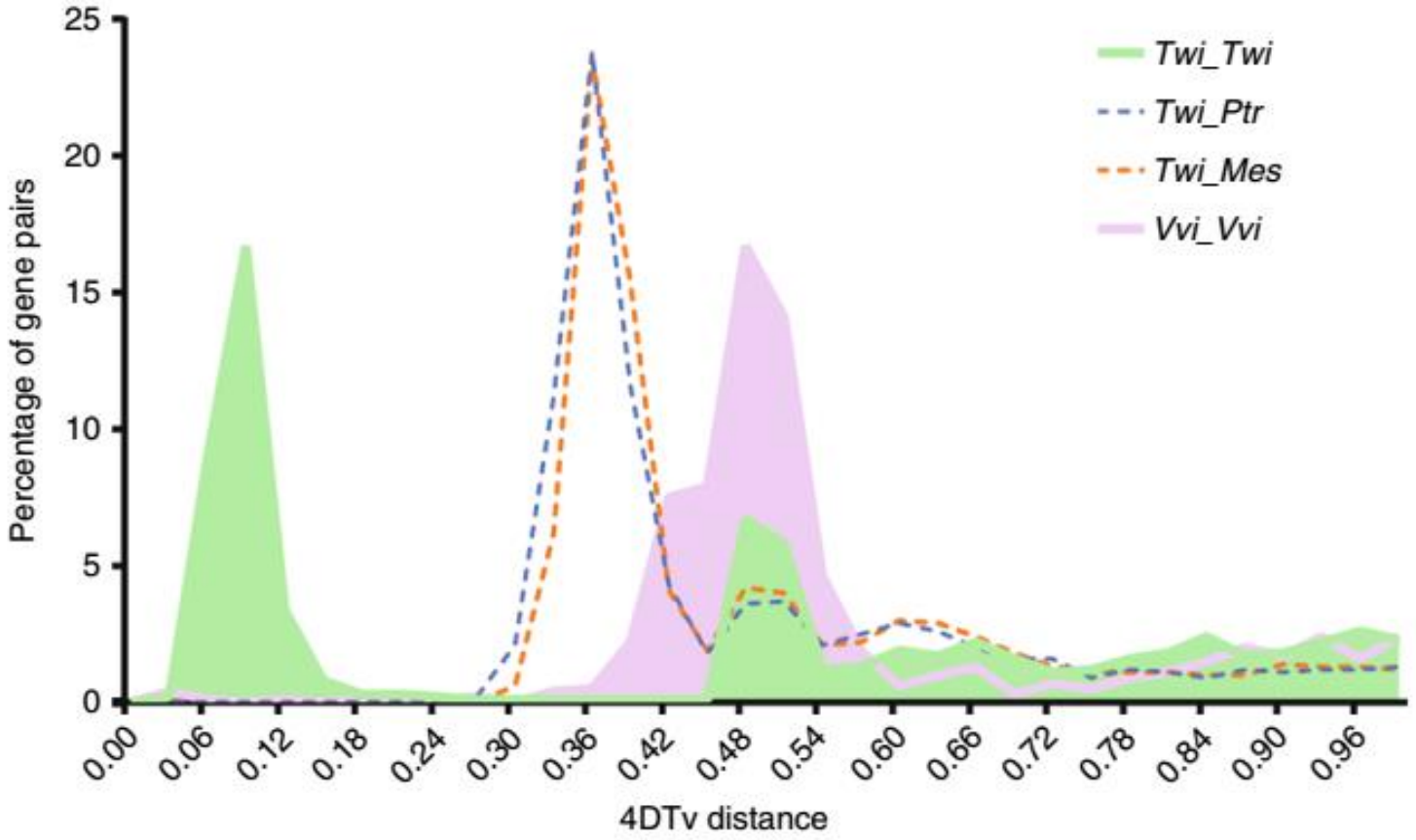


四重简并位点（Fourfold Degenerate Synonymous Site, 4DTv）：如果密码子的某个位点上任何核苷酸都编码同样的氨基酸，则这个位点为四重简并位点

2.4 计算anchors（共线性基因对）Ks，4dtv

(2) 4dtv的概念

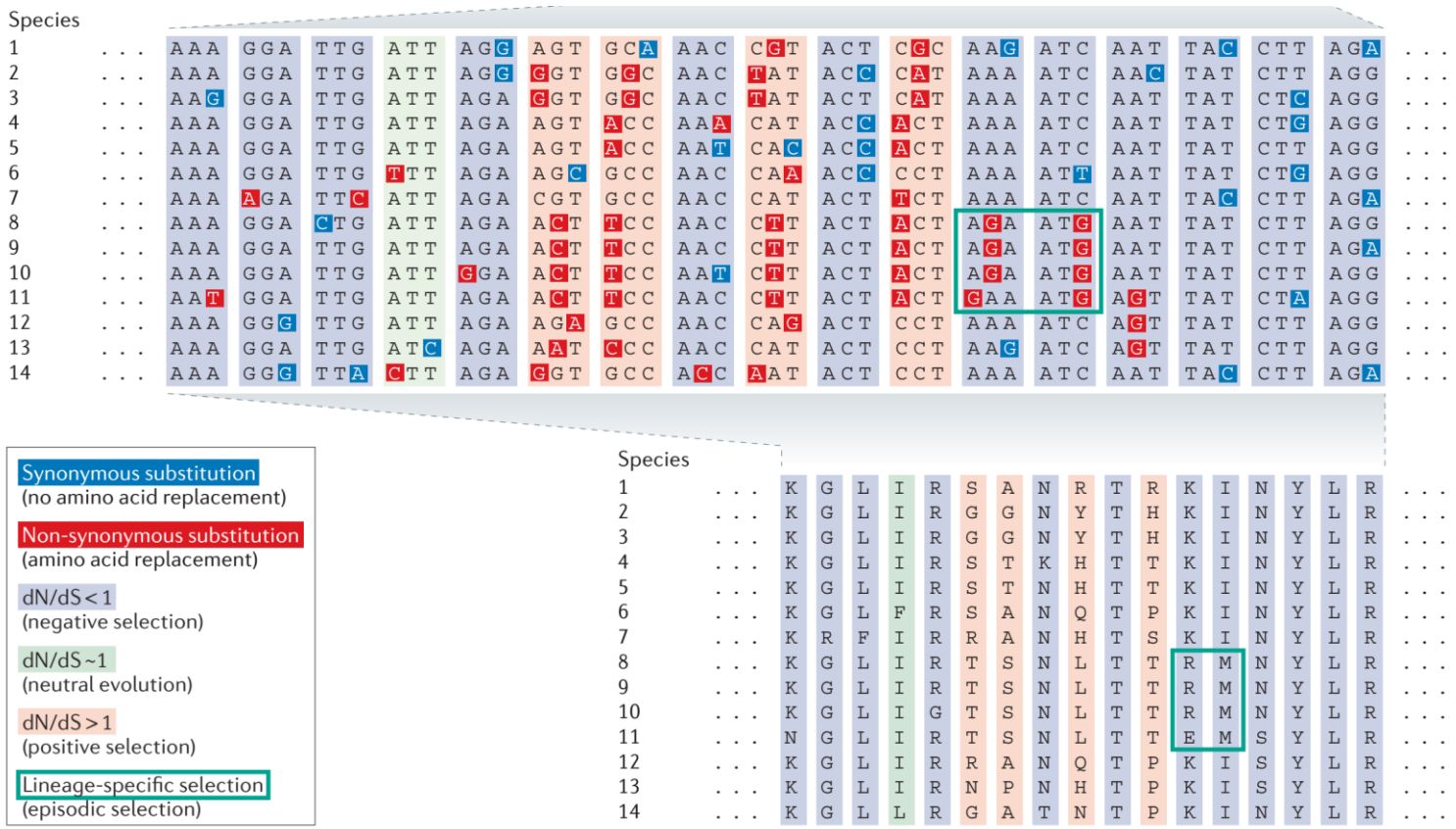
- **四重简并位点（Fourfold Degenerate Synonymous Site, 4DTv）**，在进化学上被作为评估基因组是否发生全基因组复制事件的参数；
- 右图是雷公藤基因组文献用4DTV值做的图，横轴的4DTV值，纵坐标是基因对百分比；
- 雷公藤（Twi）分别在4DTV值约0.09和0.48处出现两个峰。0.48处的峰值揭示了核心的双子叶植物γ三倍化事件，0.09处的峰值表明雷公藤从毛果杨和木薯分化后又经历了另一个全基因组复制事件。



2.4 计算anchors（共线性基因对）Ks，4dtv

(3) Ks的概念

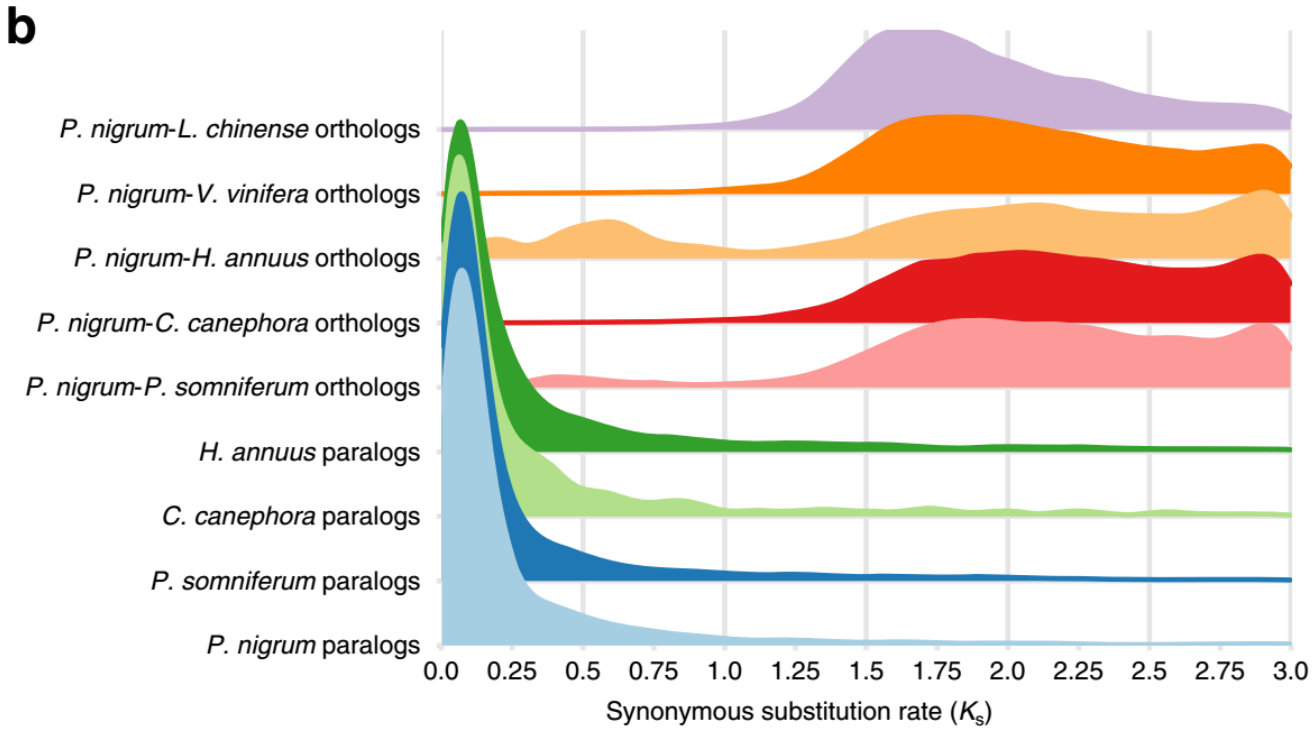
- **同义突变**(synonymous mutation)是不导致氨基酸改变的核苷酸变异，反之则称为**非同义突变**(nonsynonymous mutation)。一般认为，同义突变不受自然选择，而非同义突变则受到自然选择作用
- **同义突变频率(Ks)**=同义突变SNP数/同义位点数
- **非同义突变频率(Ka)**=非同义突变SNP数/非同义位点数
- 非同义突变率与同义突变率的比值=Ka/Ks
  - 如果Ka/Ks>1，则认为有正选择效应
  - 如果Ka/Ks=1，则认为存在中性选择
  - 如果Ka/Ks<1，则认为有纯化选择作用



2.4 计算anchors（共线性基因对）Ks，4dtv

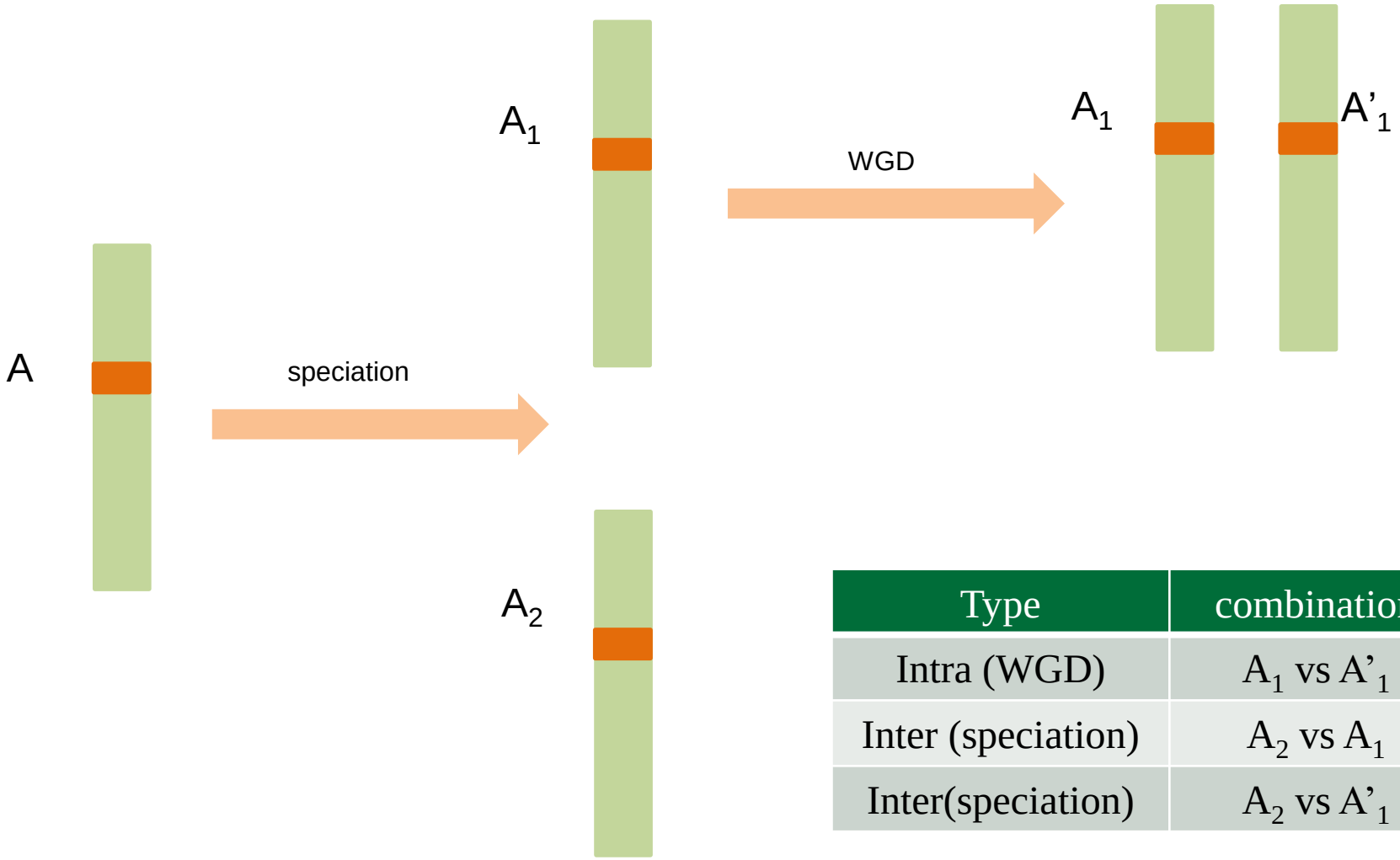
(3) Ks的概念

- 两个旁系同源序列间的Ks随着时间的推移而增加，一次基因组复制事件会产生大量的旁系同源基因对，就会在旁系同源序列对的Ks分布上有一个明显峰值
- 右图是胡椒基因组文献用Ks值做的图，横坐标是Ks值，纵坐标是基因对个数
- 胡椒（*P.nigrum*）的全基因组复制事件发生在约17.2~17.9 MYA (Ks = 0.106 ± 0.002，同义替换率为3.02E-9)



2.4 计算anchors（共线性基因对）Ks，4dtv

(4) 原理



Type	combination	Ks/4dtv
Intra (WGD)	A_1 vs A'_1	0.2
Inter (speciation)	A_2 vs A_1	0.73
Inter(speciation)	A_2 vs A'_1	0.75

2.4 计算anchors（共线性基因对）Ks，4dtv

(5) 分析步骤

Step1. 选取多个已知倍性或未知倍性的物种，两两之间（分化），以及自身与自身（WGD），利用上述软件进行共线性分析，并得到共线性基因对的信息

- mcscan：.aligns文件
- MCScanX：. collinearity文件
- JCVI：.anchors文件

Step2. 将共线性基因对进行蛋白序列的多序列比对（ muscle，mafft ），将蛋白的比对转为cds的比对（ PAL2NAL ）

```
$ muscle -in 1.pep -out 1.pep.muscle.fa
$ pal2nal.pl 1.pep.muscle.fa 1.cds -output fasta > 1.cds.muscle.fa
```

Step3. 计算Ks及4dtv

- Ks：yn00 (paml)

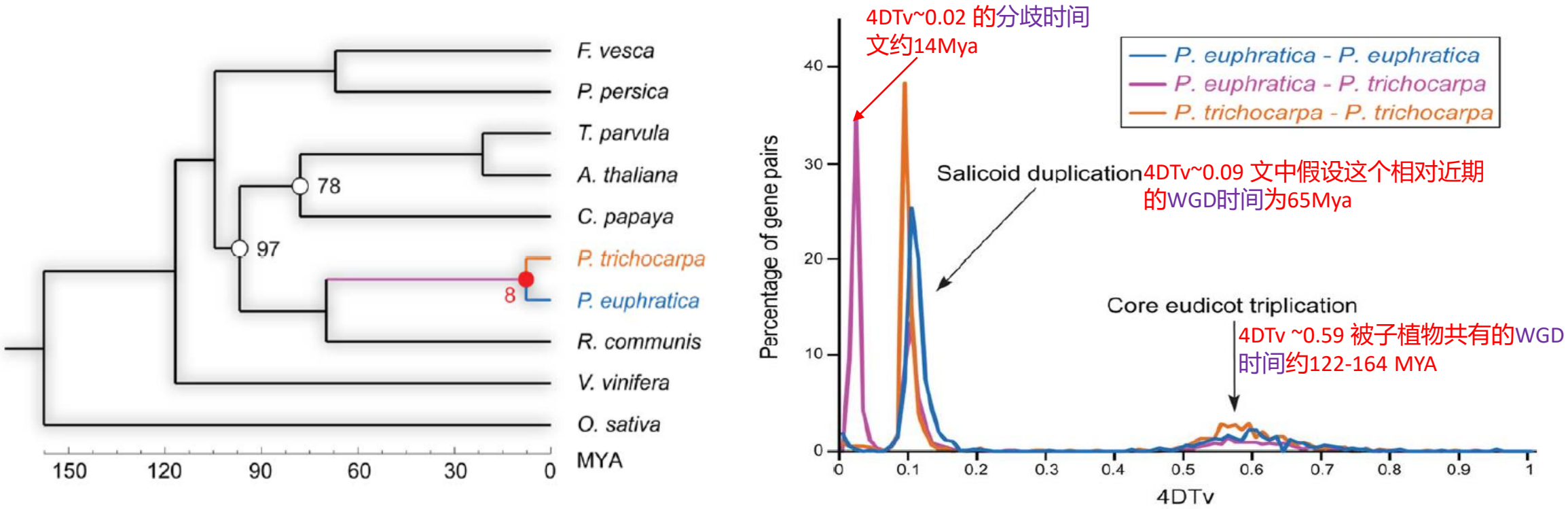
```
$ software/paml/paml4.9j_source/bin/yn00 1.ctl
```

- 4dtv： calculate_4DTV_correction.pl，custom scripts

2.4 计算anchors（共线性基因对）Ks，4dtv

(5) 分析步骤

Step4. 绘制density/直方图



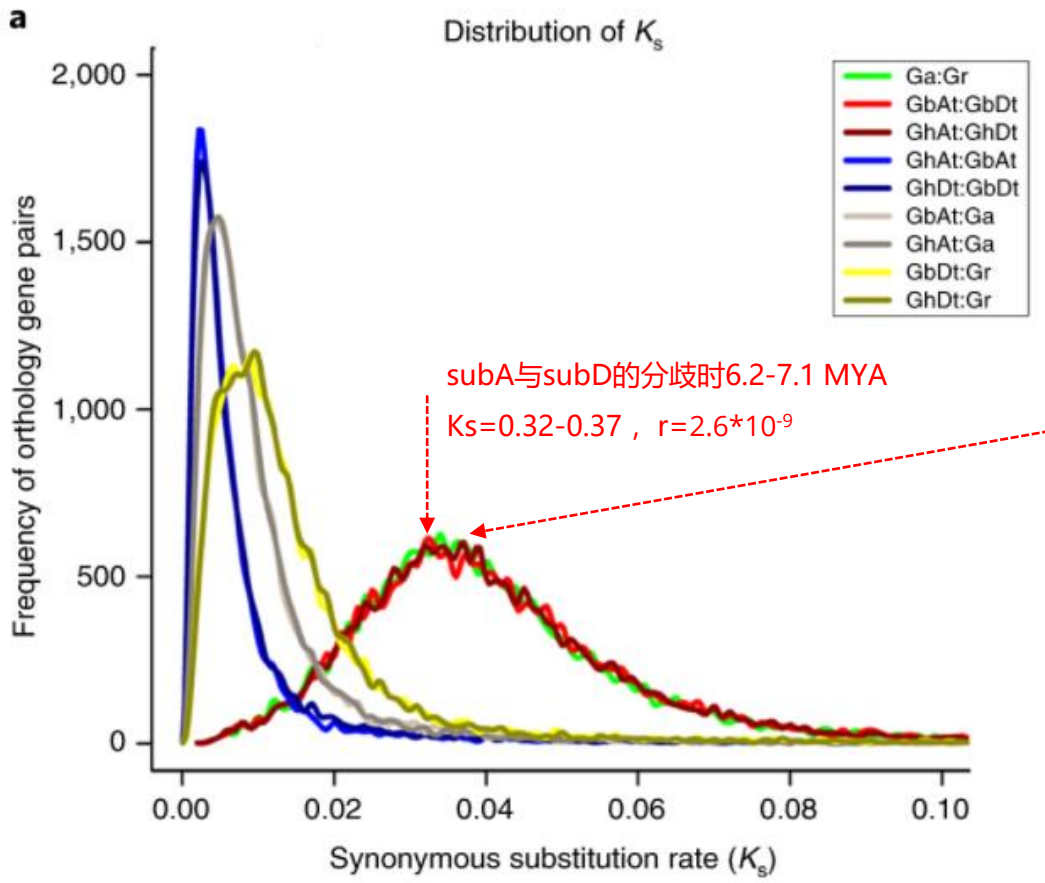
两个杨树的分歧时间：进化树分析显示为8Mya（左图），4DTV分析显示约为14Mya（右图）

Tao, Ma. , et al. *Nature Communication* (2013)

2.4 计算anchors（共线性基因对）Ks，4dtv

(5) 分析步骤

Step4. 绘制density/直方图



subA与subD的分歧时6.2-7.1 MYA

Ks=0.32-0.37， r=2.6*10^{-9}

Estimation of divergence time. On the basis of the 21,419 cotton orthologous gene sets for *G. raimondii*, *G. arboreum*, and the two subgenomes each of *G. hirsutum* and *G. barbadense*, the synonymous divergence levels (K_s) for all four cotton species were calculated. The formula $t = K_s/2r$ was used to estimate the divergence time between species, where r is the neutral substitution rate ($r = 2.6 \times 10^{-9}$).

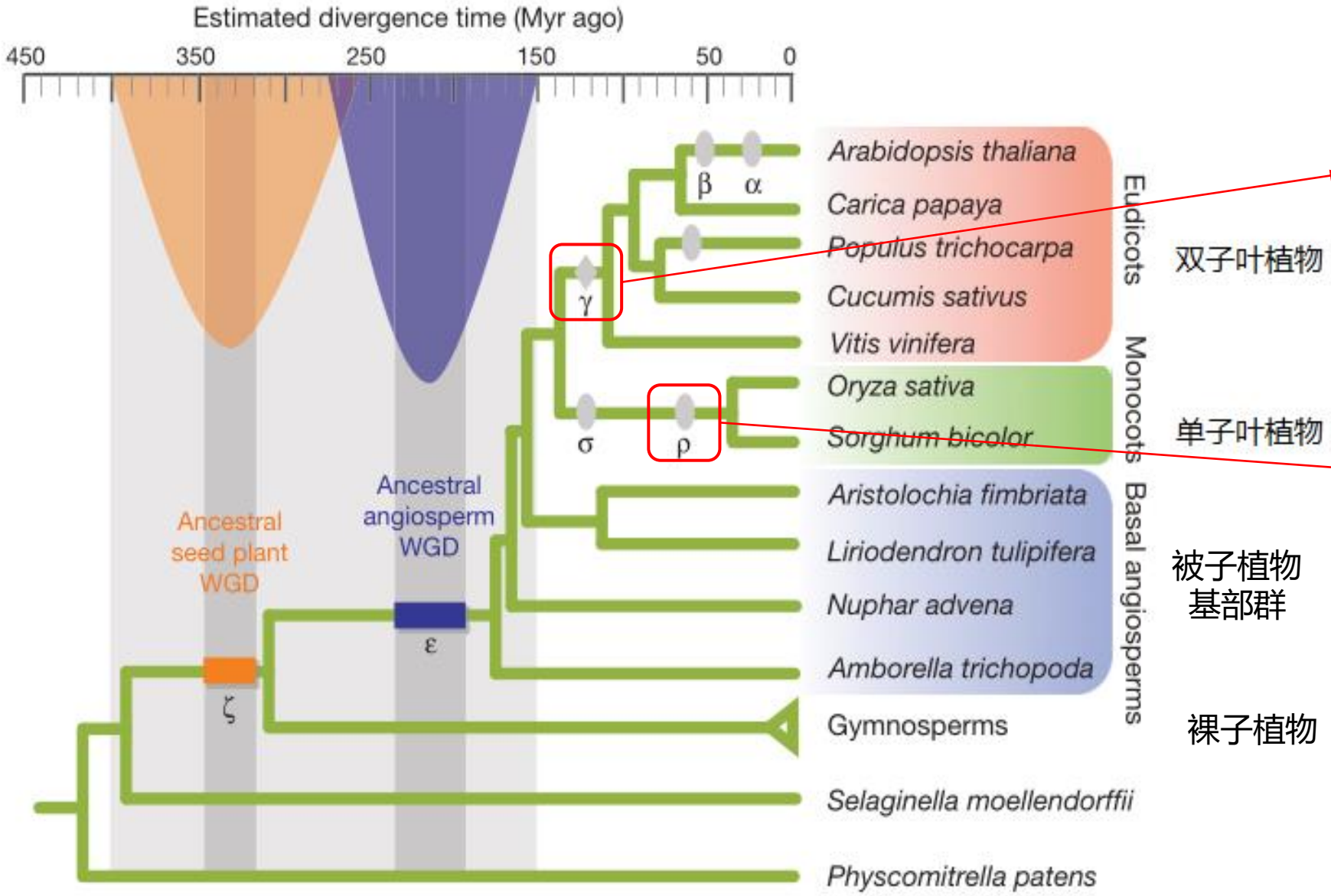
Supplementary Table 27 Peaks of each *Ks* distribution of orthologs in cotton genomes

Orthologs	*Ks* peak value	Divergence time (MYA)
G. arboreum vs *G. raimondii*	0.034	6.538
G. barbadense A_t vs *G. barbadense* D_t	0.032	6.154
G. hirsutum A_t vs *G. hirsutum* D_t	0.037	7.115
G. hirsutum A_t vs *G. barbadense* A_t	0.002	0.385
G. hirsutum D_t vs *G. barbadense* D_t	0.003	0.577
G. barbadense A_t vs *G. arboreum*	0.004	0.769
G. hirsutum A_t vs *G. arboreum*	0.005	0.962
G. barbadense D_t vs *G. raimondii*	0.009	1.731
G. hirsutum D_t vs *G. raimondii*	0.010	1.923

Note: The formula "t = Ks/2r" was used to estimate the divergence time between species, where "r" is the neutral substitution rate. A neutral substitution rate of 2.6×10^{-9} was used in the current study.

Hu, Y. , et al. *Nature Genetics* (2019)

2.5 WGD小结



- 双子叶 γ-WGD事件
- 白垩纪122-164Mya
- Ks ~ 1.5-2
- 4Dtv ~ 0.5-0.6

- 单子叶 **p**-WGD事件
- 白垩纪96-98Mya
- Ks ~ 0.6-1

勤读力耕 立己达人