2024 年 3 月整理, 完整版

【单个物种基因组鉴定 retrogene】

1.所需文件:

（全基因组基因的）

- ①蛋白序列 fasta
- ②cds 序列 fasta
- ③基因的核酸序列 fasta
- ④全基因组 gff 文件

2.所需的主要软件

- blast++
- genewise
- easyfig
- 还有其他文件处理实用小脚本, 文中直接复制黏贴就好

3.主要原则:

- 相似基因对(母基因 parental gene-反转座基因 retrogene), 存在至少一次内含子丢失事件;
- 考虑发生了选择性剪接的和结构变异(内含子化&嵌合结构)retrogene
- 未考虑 retrocopy 的情况

4.主要内容:

(一) (1)基因组基因, 单条序列 fasta 文件制备

(二) (2)genewise, 定位每个基因的外显子 内含子 loci, (不以染色体整条序列位置为基准)

(三) (3)基因组基因, blastn 同源序列比对

(四) (4)剔除 DNA-based duplication

(五) (5)retrogene, 发生内含子丢失位置的, 连续性判断

(六) (6)parental gene, 发生内含子丢失位置的分析, 再筛选

(七) (7)候选 parental-retrogene 基因对, retrogene, 1 对多个 parent 的, 只取 1 个

(八) (8)可视化候选 parental-retrogene 基因对, 人工再检查

获得 final candidate parental gene-retrogene pairs

5.注意 前 3 大步, 可同时进行, 也相较于后面的步骤费时间

6.操作位置: arvense 服务器

hyt_GeneWise 容器与 hyt 容器

START* * *

1.单条序列获取

全基因组基因拆分为单个序列的 fasta 文件

基因 id 统一有。1，就统一删除

```
sed '/>/ s/\.1//' cds.shortid.fa -i
sed '/>/ s/\.1//' pro.shortid.fa -i
```

```shell
perl /blast/utility/software/FGENESHPIPE-x86_64-linux/SCRIPTS/split_multi_fasta/split_multi_fasta.pl ../cds.shortid.fa -by_name > seq.list
rm seq.list
蛋白序列同理
```

2.genewise

单个基因外显子、内含子位置进行标记

首先确保所有基因的蛋白序列和核酸序列（包含内含子，而非 cds 序列）的 id 和个数都对应，并都分别拆分为单条序列。

2.1 基因的 genomic 序列提取，并全部处理为正向的(正义链)。

一般从 gff 文件自行提取基因的核酸序列

gff 处理为 5 列：染色体 id、起始位置、终止位置、基因 id、基因方向

```
awk '$3=="gene"{print $0}' Crucihimalaya.v2.gff > ../gff
awk '{print $1,$4,$5,$7,$9}' gff > bed
sed 's/;/ /' bed -i
sed 's/ID=//' -i bed
sed 's/Name=//' -i bed
sed 's/XMJ//' -i bed
sed 's/Chr00*/Chr/' bed -i
sed 's/Scaf00*/Scaffold/' bed -i
sed 's/g/ g/' -i bed
```

```shell
awk '{print $1"\t"$2-1"\t"$3"\t"$4}'
awk '{print $1"\t"$2-1"\t"$3"\t"$4}'
bedtools getfasta -fi ../genome.shortid.fa -bed bed.z-1 -name -fo z.fa
bedtools getfasta -fi ../genome.shortid.fa -bed bed.f-1 -name -fo f.fa
seqkit seq -p -r f.fa > f.fxhb.fa
cat z.fa f.fxhb.fa > zong.z.fa
```

```
seqkit split ../zong.z.gene.fa -i
rename "s/zong\.z\.gene\.part_XMJ/XMJ/" ./*
```

★**注意!!! **

* ·bed 文件前三列，必须出现，必须为 tab 空开的文件
* ·染色体第一个碱基的位置是 0 ★ 开头数字减掉 1★
* 比如 atcg 序列，提取 1-3：就只有 tc，没有 a，所以开头要减掉 1

seqkit seq 参数

-p, --complement complement sequence, flag '-v' is recommended to switch on

--dna2rna DNA to RNA

-r, --reverse reverse sequence

--rna2dna RNA to DNA

-v, --validate-seq validate bases according to the alphabet

2.2 genewise

只能单个基因进行 genewise

速度很慢,12h 不到 1w 个, 因此拆分 再批量

保险起见, 最好不要拆分, 慢慢来, 不会出错

ctrl+p ctrl+q 可以脱离容器并保持运行

再次进入该容器,则看不到之前跑的页面,约等于用了 screen

```bash
for i in `cat 27391.short.id`;do genewise single-pro/$i.fa single-gene/$i.fa -both -gff > genewise/$i.gw;done
```

2.3 genewise 结果处理

```bash
1.检查空文件
find blastn-jg1to1/ -name "*" -type f -size 0|wc -l
2.删除空文件
find blastn-jg1to1/ -name "*" -type f -size 0c | xargs -n 1 rm -f
3.有的 genewise 有两个预测结果， ∴只取第一个结果
mkdir genewise-only1
for i in `ls ./genewise/`;do sed 's/\/\//@@/' ./genewise/$i |sed '/@@/,$d' > ./genewise-only1/$i;done
4. 计数每个基因的外显子个数
for i in `ls ./genewise-only1/`;do exon=$(cat ./genewise-only1/$i|grep cds|wc -l); echo $i" "$exon|sed 's/\.gw//' >> genewis.only1.exons;done
```

结果:

```
ls genewise2/|wc -l
27391
find genewise2/ -name "*" -type f -size 0|wc -l
0
find genewise-only1/ -name "*" -type f -size 0|wc -l
ls genewise-only1/|wc -l
27391
wc -l genewis.only1.exons
27391
```

3.同源基因对搜索

3.1 blastp 全基因组 基因 蛋白序列互相两两比对

```bash
①蛋白比对
time blastp -query Crucihimalaya.v2.pep -db ./Crucihimalaya.v2.pep -evalue 1e-20 -outfmt 6 -qcov_hsp_perc 66 -num_threads 10 -outjg-e20-66cov
或者(暂不要)
time blastp -query Crucihimalaya.v2.pep -db ./Crucihimalaya.v2.pep -evalue 1e-20 -outfmt 6  -num_threads 30 -out jg-e20
②去除自身比对的基因对
awk '$1!=$2{print $0}'  jg-e20-66cov > jg-e20-66cov-noself
```

参数:

HSP 是 high scoring pair 的首字母缩写。局部比对的特点就是一条 query 和一条 subject 会产生多个 HSP。

每个高分比对序列中,query 覆盖率的百分比 ,(Percent query coverage per hsp)

NCBI 官方解释:

HSP: A High-scoring Segment Pair (HSP) is a local alignment with no gaps that achieves one of the highest alignment scores in a given search.

![85a5228b64185ee765343c25779ee320.png](en-resource://database/2840:1)

结果计量

```
270041
awk '$1!=$2{print $0}'  jg-e20-66cov |wc -l
242650
```

3.2 blastn 相似基因对 基因的核酸序列比对

需要以 1 条序列为文件

注意: 线程数设置无效, 只有 1~2 个 cpu, 因此很慢

```bash
①给每个核酸序列建库
for i in `ls single-gene/`;do makeblastdb -in single-gene/$i -dbtype nucl;done
②批量进行 blastn
awk '{print $1}' ../jg-e20-66cov-noselfshortid > query.id
awk '{print $2}' ../jg-e20-66cov-noselfshortid > db.id
cd blast
#/usr/bin/bash
mkdir blastn-jg1to1
lines=$(cat db.id |wc -l)
for line in $( seq 1 $lines )
do
hangshu="$line"
p="p"
sureline=$hangshu$p
query=$(sed -n $sureline query.id)
db=$(sed -n $sureline db.id)
blastn -task dc-megablast  -query ../../single-gene/$query.fa  -db ../../single-gene/$db.fa -out blastn-jg1to1/$query"-"$db -outfmt 6 -evalue 1e-6 -num_threads 30
done
①检查空文件
find blastn-jg1to1/ -name "*" -type f -size 0|wc -l
删除空文件【40w 的要 1h】
find blastn-jg1to1/ -name "*" -type f -size 0c | xargs -n 1 rm -f
```

结果:

```
ls blastn-jg1to1|wc -l
219113
242650 query.id
$ find blastn-jg1to1/ -name "*"-type f -size 0|wc -l
162335
ls blastn-jg1to1|wc -l
56778
```

4.剔除 DNA-based duplication

```bash
```

```
单行比对结果的不要了
for i in `ls blastn-jg1to1/`;do n=$(cat jg-gene2gene/$i |wc -l);echo $i" "$n >> gene1to1.num;done
awk '$2>1{print $3}' gene1to1.num|wc -l
观察不同行数的比对数量
awk '$2>1{print $2}' gene1to1.num |sort |uniq -c |sort -n -r > gene1to1.hits.lines
```

结果

```
awk '$2>1{print $3}' gene1to1.num|wc -l
23121
```

5.连续性判断

query 当做 retrogene, subject 当做母基因

比对上的相似序列的位置:

7、q. start：比对区域在查询序列(Query id)上的起始位点

8、q. end：比对区域在查询序列(Query id)上的终止位点

9、s. start：比对区域在目标序列(Subject id)上的起始位点

10、s. end：比对区域在目标序列(Subject id)上的终止位点

5.1 针对 retrogene 的分析

算法逻辑:

①对于 retrogene 来说,与母基因比对上的相邻的两段, 即两组第 7 与 8 列的位置信息, 应该:

前面的 8 应该与后面的 7,位置应该 首位相连,(允许 19bp 的重叠或远离)

(1)序列位置信息准备:

```bash
1）依据该 id 获取对应的 blast 结果中，比对位置
2）从小到大, 安装顺序捋一捋
3）把文件依据行数分组，并变成一行
mkdir lianxuxing
cd lianxuxing
awk '$2>1{print $1}' ../gene1to1.num > filename
mkdir jg-1
for i in `cat filename`;do awk '{print $7" "$8}' ../blastn-jg1to1/$i > jg-1/$i;done
mkdir jg-2
for i in `ls  jg-1/`;do sort -n  jg-1/$i >  jg-2/$i;done
mkdir jg-3
cd jg-3
将不同行数的基因比对, 分开
```

```
mkdir 2 3 4 5 6  后半截的考虑下 7 8 9 10 11 12
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 2 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 2/$i;fi;done
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 3 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 3/$i;fi;done
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 4 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 4/$i;fi;done
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 5 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 5/$i;fi;done
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 6 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 6/$i;fi;done
mkdir 7 8 9 10 11 12
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 7 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 7/$i;fi;done
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 8 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 8/$i;fi;done
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 9 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 9/$i;fi;done
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 10 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 10/$i;fi;done
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 11 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 11/$i;fi;done
for i in `ls ../jg-2/`;do n=$(cat ../jg-2/$i|wc -l); if [ "$n" -eq 12 ] ; then sed ':t;N;s/\n/ /;b t'  ../jg-2/$i > 12/$i;fi;done
```

(2)retrogene 比对上的序列 位置连续性分析:

```bash
for i in `ls 2/`;do r=$(awk '$3-$2<20&&$3-$2>-20{print $0}' 2/$i); if [ -n "$r" ] ; then echo $i >> 2p.id;fi;done
for i in `ls 3/`;do r=$(awk '($3-$2<20&&$3-$2>-20);($5-$4<20&&$5-$4>-20){print $0}' 3/$i); if [ -n "$r" ] ; then echo $i >> 3p.id;fi;done
for i in `ls 4/`;do r=$(awk '($3-$2<20&&$3-$2>-20);($5-$4<20&&$5-$4>-20);($7-$6<20&&$7-$6>-20) {print $0}' 4/$i); if [ -n "$r" ] ; then echo $i >> 4p.id;fi;done
for i in `ls 5/`;do r=$(awk '($3-$2<20&&$3-$2>-20);($5-$4<20&&$5-$4>-20);($7-$6<20&&$7-$6>-20);($9-$8<20&&$9-$8>-20) {print $0}' 5/$i); if [ -n "$r" ] ; then echo $i >> 5p.id;fi;done
for i in `ls 6/`;do r=$(awk '($3-$2<20&&$3-$2>-20);($5-$4<20&&$5-$4>-20);($7-$6<20&&$7-$6>-20);($9-$8<20&&$9-$8>-20);($11-$10<20&&$11-$10>-20) {print $0}' 6/$i); if [ -n "$r" ] ; then echo $i >> 6p.id;fi;done
for i in `ls 7/`;do r=$(awk '($3-$2<20&&$3-$2>-20);($5-$4<20&&$5-$4>-20);($7-$6<20&&$7-$6>-20);($9-$8<20&&$9-$8>-20);($11-$10<20&&$11-$10>-20);($13-$12<20&&$13-$12>-20) {print $0}' 7/$i); if [ -n "$r" ] ; then echo $i >> 7p.id;fi;done
for i in `ls 8/`;do r=$(awk '($3-$2<20&&$3-$2>-20);($5-$4<20&&$5-$4>-20);($7-$6<20&&$7-$6>-20);($9-$8<20&&$9-$8>-20);($11-$10<20&&$11-$10>-20);($13-$12<20&&$13-$12>-20);($15-$14<20&&$15-$14>-20) {print $0}' 8/$i); if [ -n "$r" ] ; then echo $i >> 8p.id;fi;done
for i in `ls 9/`;do r=$(awk '($3-$2<20&&$3-$2>-20);($5-$4<20&&$5-$4>-20);($7-$6<20&&$7-$6>-20);($9-
```

```
$8<20&&$9-$8>-20);($11-$10<20&&$11-$10>-20);($13-$12<20&&$13-$12>-20);($15-$14<20&&$15-$14>-20);($17-$16<20&&$17-$16>-20)   {print $0}' 9/$i); if [ -n "$r" ] ; then echo $i >> 9p.id;fi;done
for i in `ls 10/`;do r=$(awk '($3-$2<20&&$3-$2>-20);($5-$4<20&&$5-$4>-20);($7-$6<20&&$7-$6>-20);($9-$8<20&&$9-$8>-20);($11-$10<20&&$11-$10>-20);($13-$12<20&&$13-$12>-20);($15-$14<20&&$15-$14>-20);($17-$16<20&&$17-$16>-20);($19-$18<20&&$19-$18>-20)   {print $0}' 10/$i); if [ -n "$r" ] ; then echo $i >> 10p.id;fi;done
for i in `ls 11/`;do r=$(awk '($3-$2<20&&$3-$2>-20);($5-$4<20&&$5-$4>-20);($7-$6<20&&$7-$6>-20);($9-$8<20&&$9-$8>-20);($11-$10<20&&$11-$10>-20);($13-$12<20&&$13-$12>-20);($15-$14<20&&$15-$14>-20);($17-$16<20&&$17-$16>-20);($19-$18<20&&$19-$18>-20);($21-$20<20&&$21-$20>-20)   {print $0}' 11/$i); if [ -n "$r" ] ; then echo $i >> 11p.id;fi;done
for i in `ls 12/`;do r=$(awk '($3-$2<20&&$3-$2>-20);($5-$4<20&&$5-$4>-20);($7-$6<20&&$7-$6>-20);($9-$8<20&&$9-$8>-20);($11-$10<20&&$11-$10>-20);($13-$12<20&&$13-$12>-20);($15-$14<20&&$15-$14>-20);($17-$16<20&&$17-$16>-20);($19-$18<20&&$19-$18>-20);($21-$20<20&&$21-$20>-20);($23-$22<20&&$23-$22>-20)   {print $0}' 12/$i); if [ -n "$r" ] ; then echo $i >> 12p.id;fi;done
```

---

6 连续性再确认---针对 parental gene 的分析

对于母基因来说, 对应的两段, 即两组的第 9 和 10 列

①: 任何一段不完全在一个 intron 里. 各自的 9 和 10 不小于等于某个内含子

②: 对应的链接位置不在同一个 exon: 前面的 10 和后面的 9 不在同一个 exon

但是

只有 有两个 hits 能做, 因为其他的分不清是哪一对是有效的有连续性的

因此这一步, 只能对两个 hits 的起作用

多余两个 hits, 要再分别确认, 具体哪些位置是连续

6.1 两行

```bash
(1)提取对应的的亲本比对位置
mkdir 2parent
cd  2parent
mkdir jg-1
for i in `cat ../2p.id`; do awk '{print $9" "$10}' ../../../blastn-jg1to1/$i |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do  sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
(2) 分析
#/usr/bin/bash
```

```
for i in `ls parent1hang/`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJChr.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ] ; then echo $i >> 2p.parent.plus1;fi;
done
```

6.2 三行

```
mkdir 3parent
```

(1) 分别分析每个有连续性的位点

第一个位置有连续性的基因对 ID:

第二个位置有连续性的基因对 ID:

```
cd 3parent
for i in `ls ../3/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../3/$i); if [ -n "$r" ] ; then echo $i >>3-1p.id;fi;done
for i in `ls ../3/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../3/$i); if [ -n "$r" ] ; then echo $i >>3-2p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

比如三行的，第一个连续性位置就是：第一、二行的，9、10 列

第二个连续性位置就是：第二、三行的，9、10 列

(由于 gene 序列已全被处理为正向的，所以不考虑反向比对上的情况，因此取出来的 9、10 列直接 sort，不用考虑是否对应 retrogene，因为都是正向的 retrogene 与 parental gene 比对上的两段，都是前段 hits 上前段，后段 hits 上后段)

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```
mkdir 3-1parent 3-2parent
```

```
cd  3-1parent
mkdir jg-1
for i in `cat ../3-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../3-2parent
mkdir jg-1
for i in `cat ../3-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'2,3p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

```
cd  ../3-1parent
vi do-parent.sh
#/usr/bin/bash
for i in `cat ../3-2p.id`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >>3-2p.parent.plus;fi;
done
```

(2.3) 综合

```
cat 3-1parent/3-1p.parent.plus 3-2parent/3-2p.parent.plus |sort|uniq > 3.final.pairs
```

计量结果

```
200 3.final.pairs
```

6.2 三行

(1) 分别分析每个有连续性的位点

第一个位置有连续性的基因对 ID：

第二个位置有连续性的基因对 ID：

```
mkdir 3parent
cd 3parent
for i in `ls ../3/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../3/$i); if [ -n "$r" ] ; then echo $i >>3-1p.id;fi;done
for i in `ls ../3/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../3/$i); if [ -n "$r" ] ; then echo $i >>3-2p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

比如三行的，第一个连续性位置就是：第一、二行的，9、10 列

第二个连续性位置就是：第二、三行的，9、10 列

(由于 gene 序列已全被处理为正向的，所以不考虑反向比对上的情况，因此取出来的 9、10 列直接 sort，不用考虑是否对应 retrogene，因为都是正向的 retrogene 与 parental gene 比对上的两段，都是前段 hits 上前段，后段 hits 上后段)

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```
mkdir 3-1parent 3-2parent
cd  3-1parent
mkdir jg-1
for i in `cat ../3-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../3-2parent
mkdir jg-1
for i in `cat ../3-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'2,3p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

```
cd  ../3-1parent
vi do-parent.sh
```

```
#/usr/bin/bash
```

```
for i in `cat ../3-2p.id`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >>3-2p.parent.plus;fi;
done
```

(2.3) 综合

```
cat 3-1parent/3-1p.parent.plus 3-2parent/3-2p.parent.plus |sort|uniq > 3.final.pairs
```

计量结果

```
200 3.final.pairs
```

6.3 四行

(1) 分别分析每个有连续性的位点

```
mkdir 4parent
cd 4parent
for i in `ls ../4/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../4/$i); if [ -n "$r" ] ; then echo $i >>4-1p.id;fi;done
for i in `ls ../4/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../4/$i); if [ -n "$r" ] ; then echo $i >>4-2p.id;fi;done
for i in `ls ../4/`;do r=$(awk '($7-$6<20&&$7-$6>-20) {print $0}' ../4/$i); if [ -n "$r" ] ; then echo $i >>4-3p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```
mkdir 4-1parent 4-2parent 4-3parent
cd  4-1parent
mkdir jg-1
for i in `cat ../4-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../4-2parent
mkdir jg-1
for i in `cat ../4-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'2,3p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../4-3parent
mkdir jg-1
for i in `cat ../4-3p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'3,4p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

cd ../4-2parent

vi do-parent.sh

```
#/usr/bin/bash
for i in `cat ../4-3p.id`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
```

```
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );

lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );

if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >>4-3p.parent.plus;fi;

done
```

(2.3) 综合

```
cat 4-1parent/4-1p.parent.plus 4-2parent/4-2p.parent.plus 4-3parent/4-3p.parent.plus

|sort|uniq > ../4.final.pairs
```

计量结果

200 3.final.pairs

6.3 五行

(1) 分别分析每个有连续性的位点

```

mkdir 5parent

cd 5parent

for i in `ls ../5/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../5/$i); if [ -n "$r" ] ; then echo $i >>5-1p.id;fi;done

for i in `ls ../5/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../5/$i); if [ -n "$r" ] ; then echo $i >>5-2p.id;fi;done

for i in `ls ../5/`;do r=$(awk '($7-$6<20&&$7-$6>-20) {print $0}' ../5/$i); if [ -n "$r" ] ; then echo $i >>5-3p.id;fi;done

for i in `ls ../5/`;do r=$(awk '($9-$8<20&&$9-$8>-20) {print $0}' ../5/$i); if [ -n "$r" ] ; then echo $i >>5-4p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```

mkdir 5-1parent 5-2parent 5-3parent 5-4parent

cd  5-1parent

mkdir jg-1

for i in `cat ../5-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done

mkdir parent1hang
```

```
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../5-2parent
mkdir jg-1
for i in `cat ../5-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'2,3p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../5-3parent
mkdir jg-1
for i in `cat ../5-3p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'3,4p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../5-4parent
mkdir jg-1
for i in `cat ../5-4p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'4,5p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

cd ../5-1parent

vi do-parent.sh

```
#/usr/bin/bash
for i in `cat ../5-4p.id`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >> 5-4p.parent.plus;fi;
```

```
done
```

(2.3) 综合

```
cat    |sort|uniq > ../5.final.pairs
```

计量结果

```
200 3.final.pairs
```

六行

(1) 分别分析每个有连续性的位点

```
mkdir 6parent
cd 6parent
for i in `ls ../6/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../6/$i); if [ -n "$r" ] ; then echo $i >> 6-1p.id;fi;done
for i in `ls ../6/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../6/$i); if [ -n "$r" ] ; then echo $i >> 6-2p.id;fi;done
for i in `ls ../6/`;do r=$(awk '($7-$6<20&&$7-$6>-20) {print $0}' ../6/$i); if [ -n "$r" ] ; then echo $i >> 6-3p.id;fi;done
for i in `ls ../6/`;do r=$(awk '($9-$8<20&&$9-$8>-20) {print $0}' ../6/$i); if [ -n "$r" ] ; then echo $i >> 6-4p.id;fi;done
for i in `ls ../6/`;do r=$(awk '($11-$10<20&&$11-$10>-20) {print $0}' ../6/$i); if [ -n "$r" ] ; then echo $i >> 6-5p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```
mkdir 6-1parent 6-2parent 6-3parent 6-4parent 6-5parent
cd  6-1parent
mkdir jg-1
for i in `cat ../6-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../6-2parent
mkdir jg-1
for i in `cat ../6-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'2,3p'  |sort -n > jg-1/$i;done
```

```
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../6-3parent
mkdir jg-1
for i in `cat ../6-3p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'3,4p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../6-4parent
mkdir jg-1
for i in `cat ../6-4p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'4,5p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../6-5parent
mkdir jg-1
for i in `cat ../6-5p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'5,6p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

cd ../6-1parent

vi do-parent.sh

```
#/usr/bin/bash
for i in `cat ../6-5p.id`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
```

```
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >> 6-5p.parent.plus;fi;
done
```

(2.3) 综合

```
cat 6-1parent/6-1p.parent.plus 6-2parent/6-2p.parent.plus 6-3parent/6-3p.parent.plus 6-4parent/6-4p.parent.plus 6-5parent/6-5p.parent.plus |sort|uniq > ../6.final.pairs
```

计量结果

```
  62     6.final.pairs
130     6p.id
```

七行

(1) 分别分析每个有连续性的位点

```
mkdir 7parent
cd 7parent
for i in `ls ../7/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../7/$i); if [ -n "$r" ] ; then echo $i >> 7-1p.id;fi;done
for i in `ls ../7/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../7/$i); if [ -n "$r" ] ; then echo $i >> 7-2p.id;fi;done
for i in `ls ../7/`;do r=$(awk '($7-$6<20&&$7-$6>-20) {print $0}' ../7/$i); if [ -n "$r" ] ; then echo $i >> 7-3p.id;fi;done
for i in `ls ../7/`;do r=$(awk '($9-$8<20&&$9-$8>-20) {print $0}' ../7/$i); if [ -n "$r" ] ; then echo $i >> 7-4p.id;fi;done
for i in `ls ../7/`;do r=$(awk '($11-$10<20&&$11-$10>-20) {print $0}' ../7/$i); if [ -n "$r" ] ; then echo $i >> 7-5p.id;fi;done
for i in `ls ../7/`;do r=$(awk '($13-$12<20&&$13-$12>-20) {print $0}' ../7/$i); if [ -n "$r" ] ; then echo $i >> 7-6p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```
mkdir 7-1parent 7-2parent 7-3parent 7-4parent 7-5parent 7-6parent
cd  7-1parent
```

```
mkdir jg-1
for i in `cat ../7-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../7-2parent
mkdir jg-1
for i in `cat ../7-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'2,3p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../7-3parent
mkdir jg-1
for i in `cat ../7-3p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'3,4p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../7-4parent
mkdir jg-1
for i in `cat ../7-4p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'4,5p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../7-5parent
mkdir jg-1
for i in `cat ../7-5p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'5,6p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../7-6parent
mkdir jg-1
for i in `cat ../7-6p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'6,7p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

cd ../7-1parent

vi do-parent.sh

```
#/usr/bin/bash
for i in `cat ../7-6p.id`;
```

```
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >> 7-6p.parent.plus;fi;
done
```

(2.3) 综合

```
cat 7-1parent/7-1p.parent.plus 7-2parent/7-2p.parent.plus 7-3parent/7-3p.parent.plus 7-4parent/7-4p.parent.plus 7-5parent/7-5p.parent.plus 7-6parent/7-6p.parent.plus |sort|uniq > ../7.final.pairs
```

计量结果

```
  34    7.final.pairs
  65    7p.id
```

八行

(1) 分别分析每个有连续性的位点

```
mkdir 8parent
cd 8parent
for i in `ls ../8/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../8/$i); if [ -n "$r" ] ; then echo $i >> 8-1p.id;fi;done
for i in `ls ../8/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../8/$i); if [ -n "$r" ] ; then echo $i >> 8-2p.id;fi;done
for i in `ls ../8/`;do r=$(awk '($7-$6<20&&$7-$6>-20) {print $0}' ../8/$i); if [ -n "$r" ] ; then echo $i >> 8-3p.id;fi;done
for i in `ls ../8/`;do r=$(awk '($9-$8<20&&$9-$8>-20) {print $0}' ../8/$i); if [ -n "$r" ] ; then echo $i >> 8-4p.id;fi;done
```

```
for i in `ls ../8/`;do r=$(awk '($11-$10<20&&$11-$10>-20) {print $0}' ../8/$i); if [ -n "$r" ] ; then echo $i >> 8-5p.id;fi;done
for i in `ls ../8/`;do r=$(awk '($13-$12<20&&$13-$12>-20) {print $0}' ../8/$i); if [ -n "$r" ] ; then echo $i >> 8-6p.id;fi;done
for i in `ls ../8/`;do r=$(awk '($15-$14<20&&$15-$14>-20) {print $0}' ../8/$i); if [ -n "$r" ] ; then echo $i >> 8-7p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```
mkdir 8-1parent 8-2parent 8-3parent 8-4parent 8-5parent 8-6parent 8-7parent
cd  8-1parent
mkdir jg-1
for i in `cat ../8-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../8-2parent
mkdir jg-1
for i in `cat ../8-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'2,3p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../8-3parent
mkdir jg-1
for i in `cat ../8-3p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'3,4p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../8-4parent
mkdir jg-1
for i in `cat ../8-4p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'4,5p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../8-5parent
mkdir jg-1
for i in `cat ../8-5p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'5,6p'  |sort -n > jg-1/$i;done
mkdir parent1hang
```

```
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../8-6parent
mkdir jg-1
for i in `cat ../8-6p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'6,7p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../8-7parent
mkdir jg-1
for i in `cat ../8-7p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'7,8p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

cd ../8-1parent

vi do-parent.sh

```
#/usr/bin/bash
for i in `cat ../8-7p.id`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >> 8-7p.parent.plus;fi;
done
```

(2.3) 综合

```
cat 8-1parent/8-1p.parent.plus 8-2parent/8-2p.parent.plus 8-3parent/8-3p.parent.plus 8-4parent/8-4p.parent.plus 8-5parent/8-5p.parent.plus 8-6parent/8-6p.parent.plus 8-7parent/8-7p.parent.plus
```

```
|sort|uniq > ../8.final.pairs
```

计量结果

```
 21      8.final.pairs
 39      8p.id
```

九行

(1) 分别分析每个有连续性的位点

```
mkdir 9parent

cd 9parent

for i in `ls ../9/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../9/$i); if [ -n "$r" ] ; then echo $i >> 9-1p.id;fi;done

for i in `ls ../9/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../9/$i); if [ -n "$r" ] ; then echo $i >> 9-2p.id;fi;done

for i in `ls ../9/`;do r=$(awk '($7-$6<20&&$7-$6>-20) {print $0}' ../9/$i); if [ -n "$r" ] ; then echo $i >> 9-3p.id;fi;done

for i in `ls ../9/`;do r=$(awk '($9-$8<20&&$9-$8>-20) {print $0}' ../9/$i); if [ -n "$r" ] ; then echo $i >> 9-4p.id;fi;done

for i in `ls ../9/`;do r=$(awk '($11-$10<20&&$11-$10>-20) {print $0}' ../9/$i); if [ -n "$r" ] ; then echo $i >> 9-5p.id;fi;done

for i in `ls ../9/`;do r=$(awk '($13-$12<20&&$13-$12>-20) {print $0}' ../9/$i); if [ -n "$r" ] ; then echo $i >> 9-6p.id;fi;done

for i in `ls ../9/`;do r=$(awk '($15-$14<20&&$15-$14>-20) {print $0}' ../9/$i); if [ -n "$r" ] ; then echo $i >> 9-7p.id;fi;done

for i in `ls ../9/`;do r=$(awk '($17-$16<20&&$17-$16>-20) {print $0}' ../9/$i); if [ -n "$r" ] ; then echo $i >> 9-8p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```
mkdir 9-1parent 9-2parent 9-3parent 9-4parent 9-5parent 9-6parent 9-7parent 9-8parent

cd  9-1parent

mkdir jg-1
```

```
for i in `cat ../9-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../9-2parent
mkdir jg-1
for i in `cat ../9-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'2,3p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../9-3parent
mkdir jg-1
for i in `cat ../9-3p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'3,4p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../9-4parent
mkdir jg-1
for i in `cat ../9-4p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'4,5p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../9-5parent
mkdir jg-1
for i in `cat ../9-5p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'5,6p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../9-6parent
mkdir jg-1
for i in `cat ../9-6p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'6,7p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../9-7parent
mkdir jg-1
for i in `cat ../9-7p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'7,8p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../9-8parent
mkdir jg-1
for i in `cat ../9-8p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'8,9p'  |sort -n > jg-1/$i;done
```

```
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

cd ../9-1parent

vi do-parent.sh

```
#/usr/bin/bash
for i in `cat ../9-8p.id`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >> 9-8p.parent.plus;fi;
done
```

(2.3) 综合

```
cat 9-1parent/9-1p.parent.plus 9-2parent/9-2p.parent.plus 9-3parent/9-3p.parent.plus 9-4parent/9-4p.parent.plus 9-5parent/9-5p.parent.plus 9-6parent/9-6p.parent.plus 9-7parent/9-7p.parent.plus 9-8parent/9-8p.parent.plus |sort|uniq > ../9.final.pairs
```

计量结果

```
  8     9.final.pairs
17      9p.id
```

十行

(1) 分别分析每个有连续性的位点

```
mkdir 10parent
cd 10parent
for i in `ls ../10/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../10/$i); if [ -n "$r" ] ; then echo $i >> 10-1p.id;fi;done
for i in `ls ../10/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../10/$i); if [ -n "$r" ] ; then echo $i >> 10-2p.id;fi;done
for i in `ls ../10/`;do r=$(awk '($7-$6<20&&$7-$6>-20) {print $0}' ../10/$i); if [ -n "$r" ] ; then echo $i >> 10-3p.id;fi;done
for i in `ls ../10/`;do r=$(awk '($9-$8<20&&$9-$8>-20) {print $0}' ../10/$i); if [ -n "$r" ] ; then echo $i >> 10-4p.id;fi;done
for i in `ls ../10/`;do r=$(awk '($11-$10<20&&$11-$10>-20) {print $0}' ../10/$i); if [ -n "$r" ] ; then echo $i >> 10-5p.id;fi;done
for i in `ls ../10/`;do r=$(awk '($13-$12<20&&$13-$12>-20) {print $0}' ../10/$i); if [ -n "$r" ] ; then echo $i >> 10-6p.id;fi;done
for i in `ls ../10/`;do r=$(awk '($15-$14<20&&$15-$14>-20) {print $0}' ../10/$i); if [ -n "$r" ] ; then echo $i >> 10-7p.id;fi;done
for i in `ls ../10/`;do r=$(awk '($17-$16<20&&$17-$16>-20) {print $0}' ../10/$i); if [ -n "$r" ] ; then echo $i >> 10-8p.id;fi;done
for i in `ls ../10/`;do r=$(awk '($19-$18<20&&$19-$18>-20) {print $0}' ../10/$i); if [ -n "$r" ] ; then echo $i >> 10-9p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```
mkdir 10-1parent 10-2parent 10-3parent 10-4parent 10-5parent 10-6parent 10-7parent 10-8parent 10-9parent
cd  10-1parent
mkdir jg-1
for i in `cat ../10-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../10-2parent
mkdir jg-1
for i in `cat ../10-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'2,3p'  |sort -n > jg-1/$i;done
```

```
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../10-3parent
mkdir jg-1
for i in `cat ../10-3p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'3,4p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../10-4parent
mkdir jg-1
for i in `cat ../10-4p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'4,5p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../10-5parent
mkdir jg-1
for i in `cat ../10-5p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'5,6p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../10-6parent
mkdir jg-1
for i in `cat ../10-6p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'6,7p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../10-7parent
mkdir jg-1
for i in `cat ../10-7p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'7,8p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../10-8parent
mkdir jg-1
for i in `cat ../10-8p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'8,9p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../10-9parent
mkdir jg-1
for i in `cat ../10-9p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'9,10p'  |sort -n > jg-1/$i;done
mkdir parent1hang
```

```
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

cd ../10-2parent

vi do.sh

```
#/usr/bin/bash
for i in `cat ../10-2p.id`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >> 10-2p.parent.plus;fi;
done
```

(2.3) 综合

```
cat 10-1parent/10-1p.parent.plus 10-2parent/10-2p.parent.plus 10-3parent/10-3p.parent.plus 10-4parent/10-4p.parent.plus 10-5parent/10-5p.parent.plus 10-6parent/10-6p.parent.plus 10-7parent/10-7p.parent.plus 10-8parent/10-8p.parent.plus 10-9parent/10-9p.parent.plus |sort|uniq > ../10.final.pairs
```

计量结果

```
  6    10.final.pairs
15     10p.id
```

十一行

(1) 分别分析每个有连续性的位点

```
```

```
mkdir 11parent
cd 11parent
for i in `ls ../11/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../11/$i); if [ -n "$r" ] ; then echo $i >> 11-1p.id;fi;done
for i in `ls ../11/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../11/$i); if [ -n "$r" ] ; then echo $i >> 11-2p.id;fi;done
for i in `ls ../11/`;do r=$(awk '($7-$6<20&&$7-$6>-20) {print $0}' ../11/$i); if [ -n "$r" ] ; then echo $i >> 11-3p.id;fi;done
for i in `ls ../11/`;do r=$(awk '($9-$8<20&&$9-$8>-20) {print $0}' ../11/$i); if [ -n "$r" ] ; then echo $i >> 11-4p.id;fi;done
for i in `ls ../11/`;do r=$(awk '($11-$10<20&&$11-$10>-20) {print $0}' ../11/$i); if [ -n "$r" ] ; then echo $i >> 11-5p.id;fi;done
for i in `ls ../11/`;do r=$(awk '($13-$12<20&&$13-$12>-20) {print $0}' ../11/$i); if [ -n "$r" ] ; then echo $i >> 11-6p.id;fi;done
for i in `ls ../11/`;do r=$(awk '($15-$14<20&&$15-$14>-20) {print $0}' ../11/$i); if [ -n "$r" ] ; then echo $i >> 11-7p.id;fi;done
for i in `ls ../11/`;do r=$(awk '($17-$16<20&&$17-$16>-20) {print $0}' ../11/$i); if [ -n "$r" ] ; then echo $i >> 11-8p.id;fi;done
for i in `ls ../11/`;do r=$(awk '($19-$18<20&&$19-$18>-20) {print $0}' ../11/$i); if [ -n "$r" ] ; then echo $i >> 11-9p.id;fi;done
for i in `ls ../11/`;do r=$(awk '($21-$20<20&&$21-$20>-20)  {print $0}' ../11/$i); if [ -n "$r" ] ; then echo $i >> 11-10p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```
mkdir 11-1parent 11-2parent 11-3parent 11-4parent 11-5parent 11-6parent 11-7parent 11-8parent 11-9parent 11-10parent
cd  11-1parent
mkdir jg-1
for i in `cat ../11-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../11-2parent
mkdir jg-1
```

```
for i in `cat ../11-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'2,3p'  |sort -n > jg-1/$i;done

mkdir parent1hang

for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done

cd  ../11-3parent

mkdir jg-1

for i in `cat ../11-3p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'3,4p'  |sort -n > jg-1/$i;done

mkdir parent1hang

for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done

cd  ../11-4parent

mkdir jg-1

for i in `cat ../11-4p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'4,5p'  |sort -n > jg-1/$i;done

mkdir parent1hang

for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done

cd  ../11-5parent

mkdir jg-1

for i in `cat ../11-5p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'5,6p'  |sort -n > jg-1/$i;done

mkdir parent1hang

for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done

cd  ../11-6parent

mkdir jg-1

for i in `cat ../11-6p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'6,7p'  |sort -n > jg-1/$i;done

mkdir parent1hang

for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done

cd  ../11-7parent

mkdir jg-1

for i in `cat ../11-7p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'7,8p'  |sort -n > jg-1/$i;done

mkdir parent1hang

for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done

cd  ../11-8parent

mkdir jg-1

for i in `cat ../11-8p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'8,9p'  |sort -n > jg-1/$i;done

mkdir parent1hang

for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done

cd  ../11-9parent

mkdir jg-1

for i in `cat ../11-9p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'9,10p'  |sort -n > jg-1/$i;done
```

```
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../11-10parent
mkdir jg-1
for i in `cat ../11-10p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n'10,11p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

```
cd  ../11-1parent
vi do.sh
```

```
#/usr/bin/bash
for i in `cat ../11-1p.id`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >> 11-1p.parent.plus;fi;
done
```

(2.3) 综合

```
cat 11-1parent/11-1p.parent.plus 11-2parent/11-2p.parent.plus 11-3parent/11-3p.parent.plus 11-4parent/11-4p.parent.plus 11-5parent/11-5p.parent.plus 11-6parent/11-6p.parent.plus 11-7parent/11-7p.parent.plus 11-8parent/11-8p.parent.plus 11-9parent/11-9p.parent.plus 11-10parent/11-10p.parent.plus |sort|uniq > ../11.final.pairs
```

计量结果

```
14	11p.id
 7	11.final.pairs
```

十二行

(注意: 有的对应行数, 并没有连续的信号)

(1) 分别分析每个有连续性的位点

```
mkdir 12parent
cd 12parent
for i in `ls ../12/`;do r=$(awk '($3-$2<20&&$3-$2>-20) {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-1p.id;fi;done
for i in `ls ../12/`;do r=$(awk '($5-$4<20&&$5-$4>-20) {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-2p.id;fi;done
for i in `ls ../12/`;do r=$(awk '($7-$6<20&&$7-$6>-20) {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-3p.id;fi;done
for i in `ls ../12/`;do r=$(awk '($9-$8<20&&$9-$8>-20) {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-4p.id;fi;done
for i in `ls ../12/`;do r=$(awk '($11-$10<20&&$11-$10>-20) {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-5p.id;fi;done
for i in `ls ../12/`;do r=$(awk '($13-$12<20&&$13-$12>-20) {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-6p.id;fi;done
for i in `ls ../12/`;do r=$(awk '($15-$14<20&&$15-$14>-20) {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-7p.id;fi;done
for i in `ls ../12/`;do r=$(awk '($17-$16<20&&$17-$16>-20) {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-8p.id;fi;done
for i in `ls ../12/`;do r=$(awk '($19-$18<20&&$19-$18>-20) {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-9p.id;fi;done
for i in `ls ../12/`;do r=$(awk '($21-$20<20&&$21-$20>-20)  {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-10p.id;fi;done
for i in `ls ../12/`;do r=$(awk '($23-$22<20&&$23-$22>-20)  {print $0}' ../12/$i); if [ -n "$r" ] ; then echo $i >> 12-11p.id;fi;done
```

(2)分别

分析相应亲本基因 hits，是否，不在任何一个 intron，以及对应的链接位置不在同一个 exon

(2.1)分别 提取对应的的亲本比对位置，也理顺为横排

```
```

```
mkdir 12-1parent 12-2parent 12-3parent 12-4parent 12-5parent 12-6parent 12-7parent 12-8parent 12-9parent 12-10parent 12-11parent
cd  12-1parent
mkdir jg-1
for i in `cat ../12-1p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '1,2p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../12-2parent
mkdir jg-1
for i in `cat ../12-2p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '2,3p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../12-3parent
mkdir jg-1
for i in `cat ../12-3p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '3,4p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../12-4parent
mkdir jg-1
for i in `cat ../12-4p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '4,5p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../12-5parent
mkdir jg-1
for i in `cat ../12-5p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '5,6p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../12-6parent
mkdir jg-1
for i in `cat ../12-6p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '6,7p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../12-7parent
mkdir jg-1
for i in `cat ../12-7p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '7,8p'  |sort -n > jg-1/$i;done
mkdir parent1hang
```

```
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../12-8parent
mkdir jg-1
for i in `cat ../12-8p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '8,9p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../12-9parent
mkdir jg-1
for i in `cat ../12-9p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '9,10p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../12-10parent
mkdir jg-1
for i in `cat ../12-10p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '10,11p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
cd  ../12-11parent
mkdir jg-1
for i in `cat ../12-11p.id`; do awk '{print $9" "$10}' ../../../../blastn-jg1to1/$i |sed -n '11,12p'  |sort -n > jg-1/$i;done
mkdir parent1hang
for i in `ls jg-1/`;do sed ':t;N;s/\n/ /;b t'  jg-1/$i > parent1hang/$i;done
```

(2.2) 分析：相应的亲本基因，比对的 hits，不在单独一个内含子中，以及对应的链接位置不在同一个外显子

cd ../12-1parent

vi do.sh

```
#/usr/bin/bash
for i in `cat ../12-1p.id`;
do
q1=$(awk '{print $1}' parent1hang/$i);
z1=$(awk '{print $2}' parent1hang/$i);
q2=$(awk '{print $3}' parent1hang/$i);
z2=$(awk '{print $4}' parent1hang/$i);
```

```
parent=$(echo $i|sed 's/XMJ.*-//');
intron1=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q1"  -v z="$z1"  'q>=$4 && z<=$5 {print $0}' );
intron2=$(cat ~/ch/genewise-only1/$parent.gw|grep intron|awk -v q="$q2"  -v z="$z2"  'q>=$4 && z<=$5 {print $0}' );
lian1=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$z1"   'x>=$4 && x<=$5 {print $0}' );
lian2=$(cat ~/ch/genewise-only1/$parent.gw|grep cds|awk -v x="$q2"   'x>=$4 && x<=$5 {print $0}' );
if [  -z "$intron1" ] && [ -z "$intron2" ] && [  "$lian1" != "$lian2" ]   ; then echo $i >> 12-1p.parent.plus;fi;
done
```

(2.3) 综合

```
cat 12-1parent/12-1p.parent.plus 12-2parent/12-2p.parent.plus 12-3parent/12-3p.parent.plus 12-4parent/12-4p.parent.plus 12-5parent/12-5p.parent.plus 12-6parent/12-6p.parent.plus 12-7parent/12-7p.parent.plus 12-8parent/12-8p.parent.plus 12-9parent/12-9p.parent.plus 12-10parent/12-10p.parent.plus 12-11parent/12-11p.parent.plus |sort|uniq > ../12.final.pairs
```

计量结果

```
  7     12p.id
  2      12.final.pairs
```

所有从 2 行 hits 到 12 行的，结果合并

---

7 retrogene, 1 对多个 parent 的, 只取 1 个

计量结果

```
904 all.final.pairs
```

```
awk -F "-" '{print $1}'   all.final.pairs |sort|uniq  > all.r.id
```

```
619
```

```
awk -F "-" '{print $2}'   all.final.pairs |sort|uniq  > all.p.id
```

```
692
```

```
comm -12 all.r.id all.p.id |wc -l
```

```
92
```

给所有的 pair 加入一开始的 blastp 的 score 值, 然后选出最高分的一对

```
awk '{print $1"-"$2"\t"$3"\t"$4"\t"$5"\t"$6"\t"$7"\t"$8"\t"$9"\t"$10"\t"$11"\t"$12}'  jg-e20-66cov-noselfshortid > jg-e20-66cov-noselfshortid-id
```

(1)以 retrogene 即一对中的左 1, 保证只有 1 个为依据

为 retrogene 一对多的 pairs, 加上最初 blastp 的 score 值, 只要最高分的一个 pair

```
(1)先把 1 对 1  和 1 对多的, 分开
for i in `cat all.r.id`; do lines=$(awk -F "-" '$1=="'$i'"{print $0}' all.final.pairs|wc -l  ); if [ $lines -eq 1 ]; then  awk -F "-" '$1=="'$i'"{print $0}' all.final.pairs >> all.final.pairs.uniq; fi; done
446
mkdir lines
for i in `cat all.r.id`; do lines=$(awk -F "-" '$1=="'$i'"{print $0}' all.final.pairs|wc -l  ); if [ $lines -gt 1 ]; then awk -F "-" '$1=="'$i'"{print $0}' all.final.pairs > lines/$i ; fi; done
173+446=619 =all.r.id
(2)分析 1 对多
获取 score 值
for i in `ls lines`; do for j  in `cat lines/$i`; do awk '$1=="'$j'"{print $0}'  ~/ch/blast/jg-e20-66cov-noselfshortid-id >> lines-score/$i;done;done
按 score 值最高, 选取 1 对, 将 1 对多, 变为 1 对 1
for i in `ls lines-score`; do awk '{print $11" "$0}'  lines-score/$i|sort -n -r |head -1|awk '{print $2}' >> all.final.pairs.uniq ;done
```

最后 1 个 retrogene 对应 1 个 parent 基因的基因对, 有 619 对, 不重复的 619 个 retrogene

---

8.candidate 结构可视化

0、文件格式处理

```
cp -r genewise-only1/ genewise-only1-for-easyfig
for i in `ls genewise-only1-for-easyfig/`;do sed 's/cds/CDS/' -i genewise-only1-for-easyfig/$i;done
for i in `ls genewise-only1-for-easyfig/`;do sed 's/match/gene/' -i genewise-only1-for-easyfig/$i;done
for i in `ls genewise-only1-for-easyfig/`;do sed 's/-genewise.*$/;Parent=/' -i genewise-only1-for-easyfig/$i;done
mkdir genewise-only1-for-easyfig-final
for i in `ls genewise-only1-for-easyfig/`;do awk '{print $1"\t"$2"\t"$3"\t"$4"\t"$5"\t"$6"\t"$7"\t"$8"\tID="$9$1}'   genewise-only1-for-easyfig/$i > genewise-only1-for-easyfig-final/$i;done
```

文件名后缀保持 gw 也没影响.

1、gff3 转 gb

全基因组的基因约 3w, 耗时 4.5h 吧

```
rename "s/\.gw//" ./*
for i in `ls genewise-only1-for-easyfig-final/`;do python /home/utility/hyt/gff_to_genbank.py genewise-only1-for-easyfig-final/$i single-gene/$i.fa;done
产生的 gb 文件直接在相应的 gff 的文件夹下
ls ~/ch/genewise-only1-for-easyfig-final/|grep gb|wc -l
27391
mkdir genewise-only1-gb
mv genewise-only1-for-easyfig-final/*.gb genewise-only1-gb
```

2、批量 ——跑 easyfig

注意得使用 python2

测试下

```
python ~/hyt/Easyfig.py  -o R3.svg -svg  -f1 T -f2 100  -f CDS 255 0 0 rect-width 2500 -ann_height 100 -blast_height 1000 -glt 15  -g_height 500 -blast_files R.blast.out Rhdel01G0032900.gb Rhdel01G0032300.gb
结果 svg 图文件,在当前文件夹下
```

★注意

1\为了画图好看

虽然可以用最开始的 blastn 的比对结果

但最后用 tblastx 的结果, 更好.

2 如果额 value 值不控制, 则画出来的图, 繁杂, 不清晰

3 就是 tblastx 时 将母基因作为 query, easyfig 画出的图 上下基因的位置才理想

(1)批量 tblastx

```
#/usr/bin/bash
for line in $( seq 1 619)
  do
    hangshu=$line
    p="p"
    sureline=$hangshu$p
```

~~~
    retroid=$(sed -n $sureline/home/ch/blast/blastn-gene/lianxuxing/final/all.final.pairs.uniq |awk -F "-" '{print $1}')
    parentid=$(sed -n $sureline/home/ch/blast/blastn-gene/lianxuxing/final/all.final.pairs.uniq |awk -F "-" '{print $2}')
tblastx -query /home/ch/single-gene/$parentid.fa -db  /home/ch/single-gene/$retroid.fa  -outfmt 6  -evalue 1e-5 > $parentid-$retroid.out
done
```
~~~

检查没有空文件

~~~
$ find ./ -name "*" -type f -size 0|wc -l
0
~~~

再过滤下 identity 参数，画出来的图杂线不多，有效相似块不会太冗余

~~~
for i in `ls tblastx619/`;do awk '$12>=60{print $0}'  tblastx619/$i > tblastx619-identity60/$i;done
~~~

(2)批量跑 easy fig

~~~
```
~~~

(1)批量脚本

~~~
#/usr/bin/bash
for line in $( seq 1 619)
  do
    hangshu=$line
    p="p"
    sureline=$hangshu$p

    retroid=$(sed -n $sureline/home/ch/blast/blastn-gene/lianxuxing/final/all.final.pairs.uniq |awk -F "-" '{print $1}')
    parentid=$(sed -n $sureline/home/ch/blast/blastn-gene/lianxuxing/final/all.final.pairs.uniq |awk -F "-" '{print $2}')
    echo "python  /home/utility/hyt/Easyfig.py  -o ./${retroid}-${parentid}.svg  -svg-f1 T -f2 100  -f CDS 255 0 0 rect-width 2500 -ann_height 100 -blast_height 1000 -glt 15   -g_height 500  -blast_files /home/ch/blast/blastn-gene/tblastx619-identity60/${parentid}-${retroid}.out  /home/ch/genewise-only1-gb/${parentid}.gb    /home/ch/genewise-only1-gb/${retroid}.gb " >> 1.sh
done
~~~

(2)批量运行

~~~
cat 1.sh|parallel -j 10
```
~~~

END

END