translate_markers

Noëlle Schenk

July 20, 2018

Re-formatting data

The downloaded data from Guo 2017 needs to be manually reformatted. The table "41598_2017_3528_MOESM4_ESM-1.xls" contains rows for bin names and their location on the map, as well as the chromosome name. The chromosome number was inserted in the third column (named "chromosome") and all rows which did not contain information on bin name, location and chromosome were deleted. The file "41598_2017_3528_MOESM5_ESM.xls" was just converted to .csv.

Reading in data and loading required modules.

```
require(data.table)
```

```
## Loading required package: data.table
```

```
scguo <- read.csv("41598_2017_3528_MOESM5_ESM.csv")
names(scguo)[1] <- "bin"
head(scguo)
```

```
##           bin           PhyChr  PhyPos
## 1   AE_bin1_2 Peaxi162Scf00261  527310
## 2   AE_bin1_2 Peaxi162Scf00295  982218
## 3   AE_bin2_2 Peaxi162Scf00353  887063
## 4   AE_bin2_2 Peaxi162Scf00164  188121
## 5 AE_bin3_202 Peaxi162Scf00124  356183
## 6 AE_bin3_202 Peaxi162Scf00044 1234085
```

This table contains 330 unique bins.

```
binguo <- read.csv("41598_2017_3528_MOESM4_ESM.csv")
head(binguo)
```

```
##          bin location chromosome
## 1 AE_bin64_2    0.000          1
## 2 AE_bin66_3    1.220          1
## 3 AE_bin65_2    1.911          1
## 4 AE_bin70_1    4.993          1
## 5 AE_bin67_3    5.274          1
## 6 AE_bin72_5    5.736          1
```

This table contains 368 unique bins. There are more bins in the bin map.

```
# combine the two tables by bin name
guo <- merge(binguo, scguo, by="bin")
head(guo)
```

```
##           bin location chromosome           PhyChr  PhyPos
## 1 AE_bin100_3    1.146          3 Peaxi162Scf00038 2016598
## 2 AE_bin100_3    1.146          3 Peaxi162Scf00038 2016597
## 3 AE_bin100_3    1.146          3 Peaxi162Scf00038 2016578
## 4 AE_bin101_1    5.884          3 Peaxi162Scf00038 1819777
## 5 AE_bin102_6    2.914          3 Peaxi162Scf00038 1848611
```

```
## 6 AE_bin102_6    2.914          3 Peaxi162Scf00038 1848645
```

The map from Guo 2017 contained 6291 SNP markers, unlike reported in the paper (6582).

Bins from genetic map which do not occurr in the SNP genetic map : 0

The combined data has 3396 SNP markers (all belonging to a bin).

Generate the locations which will be read out from *P.axillaris* reference genome.

The table with scaffold names and according bins contained 2895 bins which were not found in the bin genetic map. ## Question why are ther some bins which were not included in the published map?

```
rm(scguo); rm(binguo); gc()
```

```
##          used (Mb) gc trigger (Mb) max used (Mb)
## Ncells 518601 27.7     940480 50.3   940480 50.3
## Vcells 996770  7.7    1650153 12.6  1312497 10.1
```

```
halflength <- 100
guo[,"snppos"] <- halflength + 1
guo[,"locstart"] <- guo[,"PhyPos"] - halflength
guo[,"locend"] <- guo[,"PhyPos"] + halflength + 1
# starting position can never be smaller than 0. Convert all negative values to 0 - there are no negati
# which(guo[, "locstart"] < 1)
rm(halflength); gc()
```

```
##           used (Mb) gc trigger (Mb) max used (Mb)
## Ncells  518691 27.8     940480 50.3   940480 50.3
## Vcells 1007035  7.7    2060183 15.8  1312497 10.1
```

Generate the bash file for reading out positions.

```
bashinput <- paste("samtools faidx /home/exserta/Documents/master_project_noelle/data/axillaris_genome_
guo[,"inp"] <- paste(paste(guo$PhyChr, guo$locstart, sep = ":"), guo$locend, sep="-")
write(bashinput, "find_marker_sequences.sh")
```

Run bash script which creates a new fasta file containing sequences for all markers.

```
chmod u+x find_marker_sequences.sh
./find_marker_sequences.sh
cat deleteme.fasta | awk '/^>/ {printf("\n%s\n",$0);next; } { printf("%s",$0);}  END {printf("\n");}' >
rm deletme.fasta
```

```
seqs <- read.table("marker_seqs.fasta")
seqs <- as.vector(seqs[,1])
inds <- grep ("Peaxi", seqs)
inds2 <- seq(1:length(seqs))[!seq(1:length(seqs)) %in% inds]
names <- seqs[inds] # the contig names
names <- gsub(">","", names)
sequ <- seqs[inds2] # the marker sequences
mseq <- cbind(names, sequ)
colnames(mseq) <- c("inp", "sequence")
guo[,"inp"] <- paste(guo$PhyChr, ":", guo$locstart, "-", guo$locend, sep="")
masterguo <- merge(guo, mseq, by="inp")
write.csv(masterguo, file="mastertable_guomap.csv")
rm(seqs); rm(inds); rm(inds2); rm(names); rm(sequ); rm(mseq); gc()
```

```
##           used (Mb) gc trigger (Mb) max used (Mb)
## Ncells  546676 29.2     940480 50.3   940480 50.3
```

```
## Vcells 1170376  9.0    2060183 15.8  2056923 15.7
```

Blast marker sequences in *P.exserta* genome

To find the corresponding *P. exserta* marker names and positions, the sequences of the *P. axillaris* markers are blasted against the *P. exserta* genome.

```
blastn -db /home/exserta/Documents/master_project_noelle/data/exserta_genome_NGS/P.EXSERTA.contigs.v1.1
cat tresults.out | grep -v '#' | tr '\t' ',' > trestable.csv
```

To get the position of the SNPs, generate a .fasta file with the given marker sequences from *P. exserta* and compare to the *P. axillaris* sequence.

Question

find the position of the SNP as a control?

Read and further process in R

```
blast <- read.csv("restable.csv", header=F)
names(blast) <- c("query_id", "subject_id", "%_identity", "alignment_length", "mismatches", "gap_opens"
```

Some sample output how the alignment should ideally look like (SNP at position 99, expected at position 100). ## Question Is that position good enough? Why is it slightly shifted? As I understand they used the reference genome as reference, even if the parental reads were different - or was it reverse?

From Guo 2017 (p.9): "Because the *P.axillaris* accession used for RIL population was not the same genotype as the reference genome, the genotyping data from D2B were further corrected based on the consistency of the parental genotypes and the progeny. Briefly, for loci where he genotype of the parental line *p.axillaris* was different than the reference *P.axillaris* genome, the enotypes of the entire population was switched to the other genotpye."

That means switched to the *P.exserta* genotype, right?

```
Query= Peaxi162Scf00038:1848512-1848713

Length=202
                                                                      Score     E
Sequences producing significant alignments:                          (Bits)  Value

Peex113Ctg18165                                                       368     3e-100



Query_10         1       GCTGGCCCTGAGGCTTTGTATAAACTCTGACAAGACCCATCGCATCCCACATTTTTGCAA  60
Peex113Ctg18165  930110  ............................................................  930051

Query_10         61      GTAGTCTGTCCAGATTCAGCTGTATCAGAAAAATAAACGATACATTAAATATATAGACAA  120
Peex113Ctg18165  930050  .....................................A......................  929991

Query_10         121     GGTAATGGTAAGACTTGTAGAGCATGTCGATATGTCCAGATCATTTGCCCTAATTCCAAA  180
Peex113Ctg18165  929990  ............................................................  929931

Query_10         181     ATCAGAAGCAGTGGGTCCAGTG  202
Peex113Ctg18165  929930  ......................  929909
```

```
Lambda      K        H
    1.33    0.621     1.12

Gapped
Lambda      K        H
    1.28    0.460    0.850

Effective search space used: 270281205425
```

There were 1536 unique Peaxi markers found in the P. exserta genome and 1851 queries found more than once. (from 3387)

Question

If there are several matches, the markers are filtered out according to treshold values for the given statistics. Good idea?

```
# set tresholds used to delete markers with > 1 blast result
tresholds <- c(99, 200, 4, 0.01, 214)

names(tresholds) <- c("identity", "alignm_len", "n_mismatches", "evalue", "bit_score")
dups <- as.vector(unique(blast[duplicated(blast$query_id),"query_id"]))
keep <- list()
keep_long <- data.frame(blast[1,])
names(keep_long)[3] <- "%_identity"
for(i in dups){
  # every duplicated value is searched in the blast results
  # only the best match is taken.
  a <- blast[which(blast$query_id == i), ]

  one <- which(a$`%_identity` >= tresholds["identity"])
  two <- which(a$alignment_length > tresholds["alignm_len"])
  three <- which(a$mismatches < tresholds["n_mismatches"])
  four <- which(a$evalue < tresholds["evalue"])
  five <- which(a$bit_score > tresholds["bit_score"])

  if(length(one) == 0 | length(two) == 0 | length(three) == 0 | length(four) == 0 | length(five) == 0){
  else{
    keep[[i]] <- Reduce(intersect, list(one,two,three, four, five))
    keep_long <- rbind(keep_long, a[keep[[i]][1],])
  }
}
# delete the first row, as it was added to create the data.frame quickly
keep_long <- keep_long[-1,]

# TODO : if more than 1 match is good, which one should be chosen? or should they be excluded both? nou
```

1034 of the duplicate markers fit the treshold and the best blast result will be accepted.

Question

more filtering? The markers that were found only once in the p.exserta genome sometimes don't have any mismatch or they have a lot!

Find the snp position! I can't just take the middle of the sequence as the marker position... often, the sequence is a bit shorter...

```
# merge filtered duplicates with unique matches (the ones not included in the vector dups)
a <- blast[which(!blast$query_id %in% dups),]
translated_map <- rbind(a, keep_long)
```

```
# find the genetic information from the Guo2017 map and add to translated_map table
names(guo)[9] <- "query_id"
a <- guo[which(guo$query_id %in% translated_map$query_id),]
translated_map <- merge(a, translated_map, by="query_id")
```

```
translated_map <- translated_map[,!(names(translated_map) %in% c("PhyChr", "PhyPos", "snppos", "locstar
length(unique(translated_map$query_id)) == nrow(translated_map) # only unique markers in the final map
```

```
## [1] TRUE
```

```
finalmap <- cbind(translated_map[c("subject_id", "s.start", "s.end")], "bla" = paste(translated_map$chr
write.table(finalmap, "guo_geneticmap.bed", sep="\t", col.names = F, quote=F, row.names = F)
```

The final map consists of 2431 markers of the 6,291 originally reported markers. The remaining ones could not be mapped well enough to the P.exserta sequence.

Wie genau schreibe ich das ins bed file? mit den bins? Die einfach weglassen, ist das wohl ok?