Population genomics: popular software

Jinliang Yang

03-03-2022

Path Normalization

Population genomics

- NGS and diversity measurement
  - θ_π: pairwise necleotide diversity
  - θ_W: watterson's θ, using total number of segregating sites
  - $\epsilon_1 = S_1$: the number of derived singletons in a sample.
  - η_1: based on all singletons in a sample.

–

- Population differentiation
  - F_{ST}

–

- Scan for direct selection
  - d_N/d_S or π_N/π_S

–

- Scan for linked selection
  - Tajima's D
  - Fu and Li's D, F, D*, F*

Population genomics

- NGS and diversity measurement
  - θ_π: pairwise necleotide diversity
  - θ_W: watterson's θ, using total number of segregating sites
  - $\epsilon_1 = S_1$: the number of derived singletons in a sample.
  - η_1: based on all singletons in a sample.

ANGSD

ANGSD is a software for conducting population genomics analysis using next generation sequencing data.

One advantages of this software is that it can handle mapped reads to imputed genotype probabilities.

Installation, however, is painful

Installation

```
cd $HOME/bin
```

Install from github

```
git clone --recursive https://github.com/samtools/htslib.git
git clone https://github.com/ANGSD/angsd.git
cd htslib;make;cd ../angsd ;make HTSSRC=../htslib
```

Install by directly downloading

```
wget http://popgen.dk/software/download/angsd/angsd0.936.tar.gz
tar xf angsd0.936.tar.gz
cd htslib;make;cd ..
cd angsd
make HTSSRC=../htslib
cd ..
```

Instead, use `Conda`

`Conda` is a package and environment manager! - by far the **easiest way to handle installing** most of the tools we want to use in bioinformatics.

It has been installed on the HCC.

To learn more about conda, read this introduction.

Making a new environment

The simplest way we can create a new conda environment is like so:

```
module load anaconda
conda create -n mypopgen
```

```
conda activate mypopgen
```

Entering an environment

Installing packages The first thing I usually do is just search in a web-browser for `conda install` plus whatever program I am looking for.

```
conda install -c bioconda angsd
```

```
conda deactivate
```

Exiting an environment

NGS and diversity measurement

```
cd largedata;
mkdir lab7
cp /common/jyanglab/jyang21/courses/2022-agro932-lab/largedata/lab5/bamlist.txt lab7
cp /common/jyanglab/jyang21/courses/2022-agro932-lab/largedata/lab5/sorted_l* lab7
cp /common/jyanglab/jyang21/courses/2022-agro932-lab/largedata/lab5/Zea_mays.B73_RefGen_v4.dna.chromoso
```

Use our simulated data from `lab5`

```
conda activate mypopgen
angsd
```

Activate my conda environment Or

```
module load angsd
angsd
```

ANGSD for diversity meansurement

```
cd lab7
module load samtools
samtools faidx Zea_mays.B73_RefGen_v4.dna.chromosome.Mt.fa

# run ANGSD to calculated folded SFS
angsd -bam bamlist.txt -out output -doMajorMinor 1 -doMaf 1 -doSaf 2 -uniqueOnly 0 -anc Zea_mays.B73_Re

# use realSFS to calculate sfs
realSFS output.saf.idx -fold 1 > output.sfs

# try this version
/common/jyanglab/gxu6/software/angsd/misc/realSFS output.saf.idx -fold 1 > output.sfs
```

```
## cp sfs to the cache/ folder
cp output.sfs ../../cache/
```

Copy the result to cache/ folder

Calculate the thetas

```
/common/jyanglab/gxu6/software/angsd/misc/realSFS saf2theta output.saf.idx -sfs output.sfs -outname outp
```

For each site The output from the above command are two files `output.thetas.gz` and `output.thetas.idx`.
- A formal description of these files can be found in the `doc/formats.pdf` in the angsd package.

```
/common/jyanglab/gxu6/software/angsd/misc/thetaStat do_stat output.thetas.idx -win 5000 -step 1000  -ou
# Copy the result to `cache/` folder
cp output.theta.5k.gz.pestPG ../../cache/
```

```
git add --all
git commit -m "theta values"
git push
```

For stepping window

Visualize the results

In the local computer, using `R`:

```
s <- scan('cache/output.sfs')
s <- s[-c(1,length(s))]
s <- s/sum(s)
barplot(s,names=1:length(s), main='SFS')
```

Barplot for SFS

```
theta <- read.table("cache/output.theta.5k.gz.pestPG", header=TRUE)
hist(theta$tW, xlab="theta_w", main="disverity")
```

Histgram distribution of the theta values

```
plot(theta$WinCenter, theta$Tajima, xlab="Physical position", ylab="Tajima's D", col="#5f9ea0", pch=16)
```

Scatter plot of the Tajima's D values

Fst using `vcftools`

input data:

- variant call format (or the VCF/BCF file)
- need to determine the populations

```
module load bcftools
# you must be in your lab7/ folder
cp /common/jyanglab/jyang21/courses/2022-agro932-lab/largedata/lab5/snps.bcf .
bcftools view snps.bcf | head -n 40
```

sorted_l10.bam sorted_l11.bam sorted_l12.bam sorted_l13.bam sorted_l14.bam sorted_l15.bam sorted_l16.bam sorted_l17.bam sorted_l18.bam sorted_l19.bam sorted_l1.bam sorted_l20.bam sorted_l2.bam sorted_l3.bam sorted_l4.bam sorted_l5.bam sorted_l6.bam sorted_l7.bam sorted_l8.bam sorted_l9.bam

```
for ((i=1;i<=10;i++)) ; do echo "sorted_l$i.bam" >> pop1.txt;  done
for ((i=11;i<=20;i++)) ; do echo "sorted_l$i.bam" >> pop2.txt;  done
```

Fst using `vcftools`

Window based Fst

```
module load vcftools
vcftools --bcf snps.bcf --weir-fst-pop pop1.txt --weir-fst-pop pop2.txt --fst-window-size 10000 --fst-w
```

Store the Weir Fst results

```
## cp Fst to the cache/ folder
cp win_1k.windowed.weir.fst ../../cache/
```

XP-CLR approach for selection scan

input data:

- variant call format (VCF file only)
- need to determine the populations

```
module load xpclr/1.1
module load bcftools
bcftools convert snps.bcf -O v -o snp.vcf

xpclr --input snp.vcf --out ./xpclr_res.txt  --format vcf --samplesA pop1.txt --samplesB pop2.txt --chr
```

XP-CLR approach for selection scan

using slurm script

```
cd ../../
pwd
```

```
/common/jyanglab/jyang21/courses/2022-agro932-lab
```

```
#!/bin/bash -l
#SBATCH -D /common/jyanglab/jyang21/courses/2022-agro932-lab
#SBATCH -o /common/jyanglab/jyang21/courses/2022-agro932-lab/slurm-log/xpclr-stdout-%j.txt
#SBATCH -e /common/jyanglab/jyang21/courses/2022-agro932-lab/slurm-log/xpclr-stderr-%j.txt
#SBATCH -J xpclr
#SBATCH -t 10:00:00
#SBATCH --mail-user=your_email_address@gmail.com
#SBATCH --mail-type=END #email if ends
#SBATCH --mail-type=FAIL #email if fails
set -e
set -u

### your script here:
module load xpclr/1.1
xpclr --input snp.vcf --out ./xpclr_res.txt  --format vcf --samplesA pop1.txt --samplesB pop2.txt --chr
```

```
vi slurm-script/xpclr.sh
i
# copy and paste the above code
```

XP-CLR approach for selection scan

using slurm script

```
sbatch --licenses=common --ntasks=2 --mem=10G slurm-script/my_theta.sh
## check your job status
squeue | grep "YOUR USER ID"
```

Store the XP-CLR results

```
cd largedata/lab7
cp xpclr_res.txt ../../cache/
```

Type git command to version control your results

```
git add --all
git commit -m "Fst and XP-CLR results"
git push
```