README

This README applies to software developed by:
#
GI Sistemas Naturales e Historia Forestal (formerly known as GI Genetica, Fisiologia e Historia Forestal)
Dpto. Sistemas y Recursos Naturales
ETSI Montes, Forestal y del Medio Natural
Universidad Politecnica de Madrid
https://github.com/ggfhf/
#
Licence: GNU General Public Licence Version 3.

This package contains the workflow Bash and R scripts used in the manuscript to analyze demultiplexed SE reads from a ddRADseq experiment in *Quercus suber*, *Quercus ilex*, and their hybrids. Please, see the details in:
"**ddRADseq analysis of *Quercus ilex* and *Q. suber* hybrids to identify species genomic boundaries and permeability**" (submitted).

1. DEPENDENCIES:

Before running bash scripts in section 2., be sure you have the following applications installed in your system:

(a) Python 3 - should be native to your Linux system

(b) Bioconda - is necessary to install the bionformatic applications to be used in all scripts (i.e. bcftools,
cutadapt, fastqc, gmap/gsnap, samtools and vcftools).
To install Bioconda Conda/Bioconda, we have to install Miniconda3 first by typing the following instructions in a terminal:

```
$ cd /PATH_TO_YOUR_APPS_DIRECTORY (where you will install Miniconda3)
$ wget https://repo.continuum.io/miniconda/Miniconda3-latest-Linux-x86_64.sh
$ chmod u+x Miniconda3-latest-Linux-x86_64.sh
$ ./Miniconda3-latest-Linux-x86_64.sh -b -p /PATH_TO_YOUR_APPS_DIRECTORY/Miniconda3
$ rm Miniconda3-latest-Linux-x86_64.sh

$ cd /PATH_TO_YOUR_APPS_DIRECTORY/Miniconda3/bin
$ ./conda config --add channels defaults
$ ./conda config --add channels conda-forge
$ ./conda config --add channels r
$ ./conda config --add channels bioconda

$ cd /PATH_TO_YOUR_APPS_DIRECTORY/Miniconda3/bin
$ ./pip install gffutils
$ ./conda install --yes joblib
$ ./conda install --yes biopython
$ ./conda install --yes matplotlib
```

```
$ cd /PATH_TO_YOUR_APPS_DIRECTORY/Miniconda3/bin
$ ./conda create --yes --name py27 python=2.7

$ cd /PATH_TO_YOUR_APPS_DIRECTORY/Miniconda3/bin
$ source activate py27
$ ./pip install gffutils
$ ./conda install --yes joblib
$ ./conda install --yes biopython
$ ./conda install --yes matplotlib
$ source deactivate py27
```

Now we can create environments within Miniconda3 to install all the required packages, by typing the following instruction in a terminal:

```
$ cd /PATH_TO_YOUR_APPS_DIRECTORY/Miniconda3/bin
$ ./conda create --yes --name bcftools bcftools
$ ./conda create --yes --name cutadapt cutadapt
$ ./conda create --yes --name fastqc fastqc
$ ./conda create --yes --name gmap gmap
$ ./conda create --yes --name samtools samtools
$ ./conda create --yes --name vcftools vcftools
$ ./conda create --yes --name tabix tabix
```

(c) SQLite3:
SQLite3 allows building SQlite relational databases with all the information generated from variant calling files, genomic gff files and functional annotation. First SQLite3 is installed by typing the following instruction in a terminal:

```
$ sudo apt-get install --yes libjpeg62
$ sudo apt-get install --yes sqlite3
$ sudo apt-get install --yes sqlite3-doc
$ sudo apt-get install --yes libsqlite3-dev
```

DB Browser for SQLite:

```
$ sudo apt-get install --yes sqlitebrowser
```

(d) NGShelper (Mora-Márquez et al. unpublished)
Download the zip version of NGShelper to yor Apps directory from:
https://github.com/GGFHF/NGShelper.
Decompress the file and set permisions to python and bash files of the NGShelper package:

```
$ cd /PATH_TO_YOUR_APPS_DIRECTORY
$unzip NGShelper-master.zip
$cd NGShelper-master
$chmod u+x *py *sh
```

(e) TOA (Mora-Márquez et al. under review)
Download the zip version of TOA to yor Apps directory from:
https://github.com/GGFHF/TOA
Follow the manual instructions to automatically upload functional annotation databases and run TOA executions with multi-fasta files.

(f) SimHyb (Soto et al. 2018)

Download the zip version of SimHyb to your Apps directory from:
https://github.com/GGFHF/SimHyb.
Decompress the file:

```
$ cd /PATH_TO_YOUR_APPS_DIRECTORY
$unzip SimHyb-master.zip
```

SimHyb was programmed in Java 8, and runs in any computer with an OS that allows for Java (https://www.java.com/), or OpenJDK (http://openjdk.java.net/). In Linux, you can install OpenJDK 8 to execute the program by typing the following instructions in a terminal:

```
$ sudo apt-get install --yes openjdk-8-jre
$ sudo apt-get install --yes openjdk-8-jdk
$ sudo apt-get install --yes openjdk-8-dbg
$ sudo apt-get install --yes openjdk-8-source
$ sudo apt-get install --yes openjdk-8-demo
$ sudo apt-get install --yes openjdk-8-doc
```

The program will simply run typing:

```
$ java -jar /PATH_TO_YOUR_APPS_DIRECTORYSimHyb-master/SimHyb/simhyb.jar
```

(g) Structure
Download Structure for Linux without graphical front end to your Apps directory from:
https://web.stanford.edu/group/pritchardlab/structure_software/release_versions/v2.3.4/html/structure.html
Structure 2.3.4 was programmed for a 32 bits environment. In order to execute it in a 64 bits environment, you need to type the following instructions in a terminal:

```
$ sudo dpkg --add-architecture i386
$ sudo apt-get update
$ sudo apt-get install libc6:i386 libncurses5:i386 libstdc++6:i386
```

Then, structure is simply executed as:

```
$ cd /PATH_TO_YOUR_APPS_DIRECTORY
$ ./structure
```

(h) NGScloud2 (Mora-Márquez et al. unpublished yet)
NGScloud2 allows to build the computational infrastructure needed to run 20 Structure processess in the AWS EC2 cloud service.
Download the zip version of NGShelper to yor Apps directory from:
https://github.com/GGFHF/NGShelper
Follow the manual instructions to automatically run NGScloud2.

(i) Introgress
Introgress is an R package, therefore, R needs to be installed in your system. We advice to use rstudio
to handle the scripts. If you want to install the last version of R and rstudio from a terminal, type:
R:

```
$ sudo apt-get install --yes r-base
$ sudo apt-get install --yes r-base-dev
$ sudo apt-get install --yes r-doc-info
$ sudo apt-get install --yes r-doc-pdf
$ sudo apt-get install --yes r-mathlib
$ #---sudo apt-get install --yes r-cran-rodbc
```

To install and update:

```
$ sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys E084DAB9
```

Add the following line in /etc/apt/sources.list:

deb https://cloud.r-project.org/bin/linux/ubuntu bionic-cran35/

Again in a terminal:

```
$ sudo apt-get remove -y 'r-cran-*'
$ sudo apt-get update
$ sudo apt-get install r-base r-base-dev
$ sudo apt-get upgrade
```

In Rstudio or R console:

```
> update.packages(ask=FALSE)
```

RStudio:

```
$ sudo apt-get install --yes libjpeg62
$ wget https://download1.rstudio.org/rstudio-xenial-1.1.463-amd64.deb
$ sudo dpkg -i rstudio-xenial-1.1.463-amd64.deb
$ rm rstudio-xenial-1.1.463-amd64.deb
```

To install introgress type in Rstudio or in R console:

```
> install.packages("introgress",dependencies=TRUE)
> library(introgress)
```

(j) parallel:

```
$ parallel -sudo apt install parallel
```

2. BASH SCRIPTS

The parameters that need to be modified in each script are indicated at the top of the script. Comments with capital letters indicate the command lines that need to be modified to include path to specific applications or to the user's working directory.

Scripts content:

A - Pre-processing: filtering / trimming and read quality assessment

01-CUTADAPT.sh
02-FASTQC.sh

B - Reference genome indexing and alignment with GSNAP

03-GSNAP-GENOME-INDEX-ALIGNMENT.sh

C - Pseudogenome assembly with adult original filtered reads, indexing and alignment of non-mapped reads with GSNAP

04-PSEUDOGENOME-ASSEMBLY.sh
05-GSNAP-PSEUDOGENOME-INDEX-ALIGNMENT.sh

D-Alignment post-processing and variant calling with samtools/bcftools

06-GENOME-VARIANT-CALLING.sh
07-PSEUDOGENOME-VARIANT-CALLING.sh

E-Merging and concatenating VCF files

08-GENOME-MERGE-INDIVIDUAL-VCF.sh
09-PSEUDOGENOME-MERGE-INDIVIDUAL-VCF.sh
10-CONCATENATE-GENOME-PSEUDOGENOME-VCF.sh

F-Two-steps imputation and filtering with NGShelper to build Scenario A and Scenario B (NGShelper)

11-ScnI-IMPUTE-ADULTS.sh
12-ScnIV-IMPUTE-ADULTS.sh
13-ScnI-IMPUTE-PROGENIES.sh
14-ScnIV-IMPUTE-PROGENIES.sh

G-Building Scenarios C and D from Scenario B

15-ScnII-BUILD.sh
16-ScnIII-BUILD.sh

H-Functional annotation of variants that did not map to the reference Q. suber genome assembly (TOA)

Sequences corresponding to pseudogenome SNPs were extracted from the pseudogenome fasta file ouput by SOAPDENOVO2.

17-EXTRACT-FASTA-SEQS-PSEUDOGENOME.sh

This multifasta file was submitted to TOA's standalone application through its graphical front-end. These sequences were sequentially annotated to Dicots PLAZA 4.0 (Van Bel et al. 2018), Monocots PLAZA 4.0 (Van Bel et al. 2018), Gymno PLAZA 1.0, (Proost *et al*., 2009) NCBI RefSeq Plant (O'Leary *et al*., 2016) and NCBI Nucleotide Database (NT). Follow TOA manual instructions to generate plant-annotation.csv report file for further use.

I-SQLite3 database construction (NGShelper)

The SQLite3 databases for each scenario will include information parsed from the vcf files, genomic gff annotation file for *Q. suber*, All_Plants.gene_info with gene descriptions for genes in plants (collected from NCBI database), TOA's plant-annotation.csv report file.

18-ScnI-LOAD-GENE-INFO.sh
19-ScnI-LOAD-GENOMIC-FEATURES.sh
20-ScnI-LOAD-ANNOTATIONS.sh
21-ScnI-LOAD-VCF-DATA.sh
22-ScnII-LOAD-GENE-INFO.sh
23-ScnII-LOAD-GENOMIC-FEATURES.sh
24-ScnII-LOAD-ANNOTATIONS.sh
25-ScnII-LOAD-VCF-DATA.sh
26-ScnIII-LOAD-GENE-INFO.sh
27-ScnIII-LOAD-GENOMIC-FEATURES.sh
28-ScnIII-LOAD-ANNOTATIONS.sh
29-ScnIII-LOAD-VCF-DATA.sh
30-ScnIV-LOAD-GENE-INFO.sh
31-ScnIV-LOAD-GENOMIC-FEATURES.sh
32-ScnIV-LOAD-ANNOTATIONS.sh
33-ScnIV-LOAD-VCF-DATA.sh

J-Select loci to keep for ScnD / select loci based on the mean allele frequency relative frequencies (NGShelper)

34-ScnII-QUERY-DATA.sh

Queries are further refined with the Automatic Filter commands in Excel/LibreOffice

K-Convert vcf files to Structure 2-lines files (NGShelper)

35-ScnI-VCF2STRU.sh

36-ScnII-VCF2STRU.sh
37-ScnIII-VCF2STRU.sh
38-ScnIV-VCF2STRU.sh

L-Build SimHyb required allele frequency files from VCF files (NGShelper)

39-ScnI-BUILD-ALLELE-FREQUENCIES.sh
40-ScnII-BUILD-ALLELE-FREQUENCIES.sh
41-ScnIII-BUILD-ALLELE-FREQUENCIES.sh
42-ScnIV-BUILD-ALLELE-FREQUENCIES.sh

M-Running SimHyb to generate virtual individuals

This sofware has to be "manually" run in the interface for each Scenario, and the output requires manual fit by pasting (paste command) and concatenating SimHyb output (concatenate command)

N-Converting SimHyb output (Structure 1 Line) to Structure 2 Lines format for virtual individuals

43-ScnI-SIMHYB2STRUCTURE.sh
44-ScnII-SIMHYB2STRUCTURE.sh
45-ScnIII-SIMHYB2STRUCTURE.sh
46-ScnIV-SIMHYB2STRUCTURE.sh

O-Building Structure input files for all cases and scenarios

This was perform manually copy/pasting the selected virtual and real cases in five case studies per scenario:
CASE1: Only virtual individuals (1% of hybrids)
CASE2: Only virtual individuals (5% of hybrids)
CASE3: Only virtual individuals (10% of hybrids, including all specific classes)
PREVALENCE1: includes virtual parental individuals and adult real individuals (1% of hybrids)
PREVALENCE10 includes virtual parental individuals, adult real individuals and progenies (10% of hybrids)

P-Running Structure in NGScloud2 (Mora-Márquez et al. 2018)

Structure was run in the AWS cloud in a m9xlarge instance and an EBS disk of 40 GiB using NGScloud2 (Mora-Márquez et al. 2018). We uploaded structure executables, input files and config files for each Scenario and case in separate directories hanging from /ngscloud/references/reference
Overall, 20 independent processes (5 cases x 4 scenarios).
Structure parameters are in the config files:

47-ScnI-CASO1-MAINPARAMS
48-ScnI-CASO2-MAINPARAMS
49-ScnI-CASO3-MAINPARAMS
50-ScnI-PREVALENCIA1-MAINPARAMS
51-ScnI-PREVALENCIA10-MAINPARAMS
52-ScnII-CASO1-MAINPARAMS
53-ScnII-CASO2-MAINPARAMS
54-ScnII-CASO3-MAINPARAMS
55-ScnII-PREVALENCIA1-MAINPARAMS
56-ScnII-PREVALENCIA10-MAINPARAMS
57-ScnIII-CASO1-MAINPARAMS
58-ScnIII-CASO2-MAINPARAMS
59-ScnIII-CASO3-MAINPARAMS
60-ScnIII-PREVALENCIA1-MAINPARAMS
61-ScnIII-PREVALENCIA10-MAINPARAMS
62-ScnIV-CASO1-MAINPARAMS

63-ScnIV-CASO2-MAINPARAMS
64-ScnIV-CASO3-MAINPARAMS
65-ScnIV-PREVALENCIA1-MAINPARAMS
66-ScnIV-PREVALENCIA10-MAINPARAMS

Q-Build and run INTROGRESS (Gompert & Buerkle 2010)

The input of Introgress is very similar to the input for Structure
Introgress is not able to run for ScnC, because it has heterozygotes with missing data, and introgress methodology is not able to process those data ("Missing data should be entered as 'NA/NA'").
Introgress runs through the following R scripts:

67-ScnI-CASO1-INTROGRESS.r
68-ScnI-CASO2-INTROGRESS.r
69-ScnI-CASO3-INTROGRESS.r
70-ScnI-PREVALENCIA1-INTROGRESS.r
71-ScnI-PREVALENCIA10-INTROGRESS.r
72-ScnII-CASO1-INTROGRESS.r
73-ScnII-CASO2-INTROGRESS.r
74-ScnII-CASO3-INTROGRESS.r
75-ScnII-PREVALENCIA1-INTROGRESS.r
76-ScnII-PREVALENCIA10-INTROGRESS.r
77-ScnIV-CASO1-INTROGRESS.r
78-ScnIV-CASO2-INTROGRESS.r
79-ScnIV-CASO3-INTROGRESS.r
80-ScnIV-PREVALENCIA1-INTROGRESS.r
81-ScnIV-PREVALENCIA10-INTROGRESS.r