Getting started with SINS

Slides to guide you through the steps given below: SINS - Getting started

1. **Installation :**
   Download the latest version of SINS from (https://github.com/PopConGen/SINS/releases), unzip it and **rename** it as ‘SINS’.

2. **Check Java Version:**
   Check your Java version with this command in the terminal -

```
$ java -version
```

   (Ensure you have Java version 8 (or later) installed on your machine.)

3. **Folder Structure:**
   Inside the SINS file, the ‘input’ and ‘output’ folders are crucial for the user.

   - **Input:** Place your project (for instance, ‘`Your_Simulation_Project`’) inside this folder. All simulation scenarios you wish to run MUST be inside the “input” folder inside “SINS” folder.
   - **Output:** After running SINS, your output files will be automatically saved in the ‘output’ folder by the same name (for instance, ‘`Your_Simulation_Project`’).

NOTE: If you do not see the ‘output’ folder initially, it will be created automatically after running your first simulation.

4. **Test Run:**

- To ensure SINS is functioning correctly, run the premade file ‘Premade_SINS_Project’ located in the ‘input’ folder. This will help verify that SINS is running smoothly on your system.

Steps to run Premade SINS Project

1. Inside the **‘input’** folder, you will find **‘Premade_SINS_Project’**. This file is a toy example of the format of input files for SINS.

   ```
   >SINS/input/Premade_SINS_Project
   ```

   ```
   >SINS/input/My_New_Simulation_Project_Must_Be_Here
   ```

2. To run SINS using ‘`Premade_SINS_Project`’ as input, open the terminal inside the SINS folder and run the following command:

```
$ java -jar SINS2.jar -projectName Premade_SINS_Project -outDir
output -numberOfSimulations 19 -compress noComp -parallel false
-parallelCores 0 -verbose false -outputFormat adegenet
-makeDemographicImages false
```

3. You can modify the above command as you need. You can change the output directory by changing the argument of -outDir, change the number of simulations by changing the argument of -numberOfSimulations. For eg, To run simulations faster, run in parallel and provide a number for parallel cores. Following code is an example-

```
$ java -jar SINS2.jar -projectName Premade_SINS_Project -outDir
output -numberOfSimulations 19 -compress noComp -parallel true
-parallelCores 3 -verbose false -outputFormat adegenet
-makeDemographicImages false
```

4. After running the above command, an ‘output’ folder will be created inside the “SINS” folder containing ‘Premade_SINS_Project’. Note that all the output files will be saved here if you don't change the argument of -outDir.

5. This initial example was only meant to ensure SINS run on your system, and we will not analyze the outputs. Now, lets move on to a simpler simulation scenario.

6. For the first scenario, we will simulate a panmictic population
    - Of population size K = 1000 diploid individuals
    - Mutation rate u = 0.001 per generation
    - Simulation time: 1000 generations.
    - Each individual has 10 loci and each loci has 2 alleles at the start of the simulation.

7. **Steps:**
    - Download the input file and unzip it inside ‘input’ folder of SINS

   1deme_K1000_u001_1000gen.zip

    - Run the simulation by following command:

```
$ java -jar SINS2.jar -projectName 1deme_K1000_u001_1000gen
-outDir output -numberOfSimulations 19 -compress noComp -parallel
true -parallelCores 3 -verbose false -outputFormat adegenet
```

```
-makeDemographicImages false
```

8. **Output:**
   The output folder will contain a directory named '1deme_K1000_u001_1000gen' with 19 independent simulations of the same scenario. To improve speed, these simulations were run using 3 parallel cores.

9. **Converting outputs:**
   To make the simulation outputs usable in R, we will convert them into a compatible format using **SINS_Sampler**.

Importing SINS output files into RStudio for analysis

Download the 'SINS_Sampler' from the link below and unzip in the same folder where SINS is downloaded -
SINS_Sampler.zip

Modifications:
Changes to be made in SINS_Sampler before using it.

- Go to SINS_Sampler>bin
- In 'run_sampler.sh' change `path_script according to your path.`
- In 'sampler.sh', change `pathToDir` and `directory_path`
- In 'run_GetData.sh', change `path_to_script`
- In 'Get_Data.sh', change `path`
- In 'Get_Data.R', change `path_sins`.

Install Required R packages:
You will need to install several R packages:
adegenet, ade4, ggplot2, reshape2, gridExtra, hierfstat, pegas, Hmisc, plyr, data.table, StAMPP

This can be done by copying this in R:

```
packages <- c("adegenet", "ade4", "hierfstat", "pegas", "ggplot2",
"reshape2", "gridExtra", "Hmisc", "plyr", "data.table", "StAMPP")

for (pkg in packages) {
  if (!require(pkg, quietly = TRUE)) {
    install.packages(pkg, dependencies = TRUE)
  }
}
```

Run SINS_Sampler

After the above changes have been made, Open the terminal inside the `SINS_Sampler>bin directory.`

And run the following commands one after another -

```
$ ./run_sampler.sh 1deme_K1000_u001_1000gen 10 19
```

Format : ./run_sampler.sh -filename   -no_of_loci   -no_of_simulations

```
$./run_GetData.sh 1deme_K1000_u001_1000gen 0 1000 10 19 100 50
```

Format : ./run_GetData.sh   -filename   -time_start   -time_end
-no_of_loci   -no_of_simulations   -interval_size
-no_of_individuals_sampled

After running above commands, new R files are created in `SINS>output>1deme_K1000_u001_1000gen folder. You can import these files into` R for further analysis of the simulated data.

Analyze the Output

SINS-guide-data-analysis.pdf

The above report is an example of analyzing SINS data for the mean number of alleles and expected Heterozygosity.