Student Number: 219047301

BS4701 Mini Project

WormQuest

USER GUIDE

CONTENTS

ABOUT WORMQUEST

WormQuest is a highly interactive Python programme that aids in generating a shell script for HMMer search on the University of Leicester's High-Profile Computing ALICE machine and analyses the distribution of the multi-subunit enzyme Succinate Dehydrogenase (SDH) in selected nematode species.

SDH is an essential component in of both the mitochondrial respiratory chain and the Krebs cycle, and it is typically universally conserved (Benit *et al.,* 2022). Moreover, SDH inhibitors have been recommended to use to fight against plant-parasitic nematodes proliferation on produce (Benit *et al.,* 2022; Schleker *et al.,* 2022), so to interfere with respiration. However, different species of nematodes react differently to SDH inhibitors (Schleker *et al.,* 2022), which prompts to presume there may possess different SDH enzymes.

WormQuest aims to make it simpler to handling data, generating scripts and analysing relevant information to SDH and nematodes.

KEY INFORMATION – READ BEFORE USE

WormQuest requires the use of an internet connection to use. Most of the functionalities within the program will not function if there is no connection to the web.

Please make sure to have a secured internet connection when using WormQuest.

All the files should be kept in your working directory. If you are missing any of the required files, you can download them from the associated GitHub repository (refer to GIT Repository section).

To run WormQuest, you can call it on the command line:

```
python3 wormquest.py [command] [flag]
```

Note: if python3 does not work, use 'python' instead.

Not specifying any commands will default to a help message.

The `requirements.txt` file in the GitHub Repo contains all the necessary modules to run WormQuest. A warning will show to notify the user of a missing module in case it is not found within their environment. You can install the necessary files through:

```
pip install [module]
```

Please make sure to have installed all the necessary modules when running WormQuest.

WormQuest is designed to run on Linux systems. If run on any other operating system, there is a possibility the program will not work.

Pleas ensure you are executing WormQuest on a Linux operating system.

The program performs OS tasks, which may require you to have be an administrator (or have sudo rights).

Make sure to have admin privileges for your system.

You will need to have an ALICE account in order to transfer files across.

FUNCTIONALITIES

Function 1 – Downloading and unarchiving files from WormQuest and PFAM databases

In order to fetch the most up-to-date species data, WormQuest allows the user to mine the WormBase database (Sternberg *et al.*, 2024) for the available species names and their respective BioProject identifiers. The results will go in a .csv file within the working directory. You can use it with:

```
python3 wormquest.py species -m
```

The table will be mined from https://parasite.wormbase.org/ftp.html. You can also download it from the GitHub Repo, `species_data.csv`. Make sure this is saved in your working directory.

To download species-specific protein FASTA files, you can use:

```
python3 wormquest.py species -dl_fasta [speciesindex1]
[speciesindex2] [speciesindex3] …
```

This will only work if you have a `species_data.csv` within your working directory. The same folder can be found on the GitHub Repo containing three sample species. The files will be pulled from https://ftp.ebi.ac.uk/pub/databases/wormbase/parasite/releases/WBPS19/species. The .fa.gz files will be saved within a new directory called ‘wormbase’. There is no limit in the amount of species you ask for as long as they are separated by a space.

To download the necessary HMM files for the HMMsearch (http://hmmer.org/), WormQuest can also parse through datasets and download HMM annotations related to the retrieved PFAM identifiers (Mistry *et al.*, 2020). The .tsv file was provided by the instructors, however you can also download `SearchResults-succinatedehydrogenase.tsv` from the GitHub Repo. You can do this by:

```
python3 wormquest.py pfam -dl_hmm
```

The files will be pulled from https://www.ebi.ac.uk/interpro/wwwapi//entry/pfam/. All HMM files will also be unarchived for use, eliminating the compressed files. The unarchived files will be saved inside a new directory called 'Pfam_HMM'.

By the end of this, you should have two new folders in your working directory containing .fa.gz files for your selected species, and .txt files for the detected PFAM HMM files.

Function 2 – Generating HMMsearch for Alice's SLURP

In order to work on ALICE, the generated directories need to be transferred over to the remote directory. There is an integrated function within WormQuest to do so:

```
python3 wormquest.py hpc -tr
```

It will ask for your ALICE username (ex. ab123) to perform the request. In order to complete the transfer, it will require you to input your password, and a Microsoft verification code if you have two-factor authentication. It will place 'wormbase' and 'Pfam_HMM' within your ALICE home directory (ex. ab123@alice.le.ac.uk:~).

Otherwise, you can transfer the files manually as long as they are within the working directory where you will run the shell script.

To generate the shell script to perform HMMsearch within ALICE, you can use:

```
python3 wormquest.py hpc -hmms
```

It will ask for the user's username and email address. The username is required for path creation, while the email is used to provide notifications on the the SLURP job status. Please ensure the username inputted matches the user environment. To generate the script, you will need to have followed through 'Function 1' fully, or have downloaded the data folders from the GitHub Repo.

The program gives the option to transfer the script directly from within the code. The shell script assumes that 'wormbase' and 'Pfam_HMM' will be in the working directory of ALICE; if they are not there, the script will not work. If you do not transfer through WormQuest, please ensure the folders and the shell script are both present in your ALICE working directory.

By the end of this, you should have generated a customised shell script optionally transferred to ALICE.

Once you have your script in your ALICE working directory, you can access ALICE through either NoMachine (https://www.nomachine.com/) or by using:

```
ssh (username)@alice.le.ac.uk
```

Then send the job to SLURP through:

```
sbatch hmmsearch_(yourusername).sh
```

For example:

```
ssh ab123@alice.le.ac.uk
sbatch hmmsearch_ab123.sh
```

You should receive an e-mail when the job begins, ends or fails. In case of a fail, please refer to the slurp output file within your working directory. If the job fails due to not finding HMMsearch within ALICE, please ensure that the `'hmmsearch'` variable in the shell script matches the most up to date path. It could also be that the modules could have been updated. You can check both by:

```
module load spider hmmsearch
[load the required modules with module load {module_name}]
which hmmsearch
```

Then edit the script (path and modules) accordingly if needed.

Function 3 – Demo HMMsearch Analysis

WormQuest has a demo for its analysis function.

The command provides a guided demo of how WormQuest can analyse HMMer output files. You can try it using:

```
python3 wormquest.py analysis -d
```

The command will give you a run through all the possible analysis. It uses pre-generated data available on the GitHub Repo ('demo' directory), and you are not required to download anything independently. The program will generate tables, graphs and summaries of the HMMsearch output data:

- .csv tables with parsed HMMsearch data.
- Distribution of E-values and bit scores as histograms.
- Comparison of the number of queries for three different species for the same PFAM identifier.

The last two will pop up on the screen, and the code will pause until they are closed. You can interact with the figures within the pop-up menu.

This command will generate a new directory called 'WormQuest_demo' within your current working directory containing tables and graphs. If you only want

to see what type of data can be generated, you can find all the material within the GitHub Repo under the same folder name.

By the end of this, you should have had a demonstration within the command line, and have analysed data in a new directory within your current one.

GIT REPOSITORY

The source code and project files can be accessed at: https://github.com/momiskiv/WormQuest. Last updated on the 16th December 2024.

AUTHOR AND VERSION INFORMATION

The author of WormQuest is Student 219047301. The user guide was last updated on the 16th December 2024. This is based on version 1.0.0 of WormQuest.

REFERENCES

Benit, P. *et al.* (2022) 'Succinate dehydrogenase, succinate, and superoxides: A genetic, epigenetic, metabolic, environmental explosive crossroad', *Biomedicines,* 10(8), pp. 1788. doi: 10.3390/biomedicines10081788. Available at: https://doi.org/10.3390/biomedicines10081788

Mistry, J. *et al.* (2020) 'Pfam: The protein families database in 2021', *Nucleic Acids Research,* 49(D1), pp. D412–D419. Available at: https://doi.org/10.1093/nar/gkaa913

Schleker, A.S. *et al.* (2022) 'Mode of action of fluopyram in plant-parasitic nematodes', *Scientific Reports,* 12(1), pp. 11954. Available at: https://doi.org/10.1038/s41598-022-15782-7

Sternberg, P.W. *et al.* (2024) 'WormBase 2024: Status and transitioning to alliance infrastructure', *Genetics,* 227(1), pp. iyae050. Available at: https://doi.org/10.1093/genetics/iyae050