NLR-parser additional scripts: exNLRs.sh, plot-nlr.r, exSeqList.py

Description

The shell script exNLRs.sh was written around the java application NLR-parser.jar to automate the extraction of NLRs after prediction of proteins. The script works as follows:

Article for NLR-parser:

> **NLR-parser: Rapid annotation of plant NLR complements**
> Steuernagel, B., Jupe, F., Witek, K., Jones, J. D.G., Wulff, B. B.H.
> Bioinformatics 2015
> doi: 10.1093/bioinformatics/btv005

Github page of NLR-parser:

https://github.com/steuernb/NLR-parser

What does the script do

1. make directory NLR in the same directory where the multi fasta file is located
2. call mast with the meme.xml file from NLR-parser
3. parse mast results with NLR-parser.jar
4. generate some stats about the amount of different NLRs identified with mast
5. plot the stats with plot-nlr.r
6. extract the **complete** NLR sequence names and subsequently the corresponding sequence with exSeqList.py (exSeqList.py can extract any fasta sequence from any multifasta file: exSeqList.py seqNames multifasta.fa)
7. everythings saved in the directory NLR
8. all errors are printed to file exNLR.stderr

What does it depend on

Dependencies of this script:

- A running version of NLR-parser with mast v4.9.1
- Rscript
    - library: ggplot2
- python 2.7 or 3.6
    - python script exSeqList.py

How to get it running

Get the script running:

- Install and download the dependencies
- Put all the scripts in the NLR-parser directory
- Put your NLR-parser directory in your path (edit .bash_profile):
    - export PATH="/path/to/NLR-parser:$PATH"
- edit line 6 in exNLRs.sh to fit the path to your NLR-parser.jar directory
- Make the file executable:
    - $ chmod a+x /usr/local/bin/exNLRs.sh
- Run the script:
    - $ exNLRs.sh /path/to/fasta/file.fa

Tested on MacOS and Ubuntu Linux.