Virtual Microbes quick start manual

VirtualMicrobes can be installed on a LINUX machine (tested on Ubuntu 16.04). We highly recomment installing it using a virtual python environment, which ensures all dependencies (and their version) do not conflict and break your current python projects ;)

Before you begin, make sure you have 1) python2.7, 2) virtualenv, and 3) PyQt4 installed:

Installing PIP / virtualenv: https://pip.pypa.io/en/stable/installing/
Installing PyQt4: sudo apt-get install python-qt4

Installation

Clone the virtualmicrobes code:
```
user@computer: git clone https://thocu@bitbucket.org/thocu/virtualmicrobes.git
```
Make a virtualenv (e.g. in your home dir), and include system site packages
```
user@computer: virtualenv ~/mypy --system-site-packages
```
Activate your virtual environment
```
user@computer: source ~/mypy/bin/activate
(mypy) user@computer:
```
Use pip install to get the Virtual Microbe package. (last argument is where you cloned the code, default to 'virtualmicrobes')
```
(mypy) user@computer: pip install -e virtualmicrobes
```
That's it! Now if you want to run a simulation, use the command "vermicelli.py":
```
(mypy) user@computer: vermicelli.py evo --name MyFirstMicrobes
```

All output will be directed to the folder MyFirstMicrobes. CSV files with information on genomes, cell sizes, resource concentrations, etc. can be found in MyFirstMicrobes/data/. Images of gene regulatory networks, metabolomes, and the grid are stored in MyFirstMicrobes/plots/.

For easy exploration of your microbes, use the (offline) Firefox webapplication. (the app also works online for all other browsers, but has security problems when used offline)
`file://.../MyFirstMicrobes/00_webapplication.html`

Pandora's box

Using the default options might be boring. There's a lot you can change.

Display all general options
```
user@computer: vermicelli.py --help
```

Displays all options specific to evolutionary simulations
```
user@computer: vermicelli.py evo --help
```

Displays all options specific to analysing lines of descent
```
user@computer: vermicelli.py ancestry --help
```

Here's some examples to get you started

Running a short simulation with more data
```
user@computer: source ~/mypy/bin/activate
(mypy) user@computer: vermicelli.py evo --name MyFirstMicrobes_MoreData --plot-time 10 --store-data-time 5 --save-time 50 --duration 100
```
Continue the previous simulation a bit longer in the same directory
```
(mypy) user@computer: vermicelli.py evo --load-file MyFirstMicrobes_MoreData/MyFirstMicrobes_MoreData_100.sav --duration 150
```
Continue simulation with a change in the rules, in a new directory (by giving it a new name)
```
(mypy) user@computer: vermicelli.py --universal-mut-rate-scaling 0.0 evo --load-file MyFirstMicrobes_MoreData/MyFirstMicrobes_MoreData_500.sav --name
MyFirstMicrobes_MoreData_nomutations
```

Running a short simulation with MUCH more metabolites and reactions
```
(mypy) user@computer: vermicelli.py evo --nr-resource-classes 20 --nr-energy-classes 3 --nr-building-blocks 3  --name MySecondMicrobes --duration 150
```

Trouble shooting:

When I type vermicelli.py it says 'file not found'
Did pip install VirtualMicrobe run without errors? Does 'which vermicelli.py' give a path to your virtual environment? Did you activate your local environment (mypy in the above examples)
I run vermicelli.py but the code gives an error about 'ete3' about 'TreeStyles' or 'TreeNodes'
You probably did not have PyQt4 installed correctly before you installed VirtualMicrobes. Install PyQt and reinstall the software.
I tried to use an option, but it says it doesn't recognize it
General options (vermicelli.py --help) go BEFORE evo, while other go AFTER evo. It's a mess, but trial and error gets you there. If you use keyword options (such as mutation-rates) makes sure to include a dummy argument (-) before evo:
vermicelli.py --mutation-rates point_mutation=0.8 tandem_dup=0.8 stretch_del=0.8 stretch_invert=0.8 stretch_translocate=0.8 - evo --name Rediculous_Mutation_Rates