

experiments__variable.py

Last modified: 2013-07-10 hannoell

What it does:

This files run simulations while allowing the user to choose the variable that will be varied in the simulations. At the end it also runs various datachecks on the data to make sure they make sense. It also gives the user the option to choose the minimum and maximum values and the increment that the variable will iterate over.

How to run:

To run a simulation type

```
python experiments_variable.py [experiment name] [num samples] [test var] [set iteration range]
```

Example:

```
python experiments_variable.py test 10 latency no hannoell@umich.edu
```

There are only 10 acceptable test variables that this file will accept. They are latency, sim_length, arrival_rate, tick_size, reentry_rate, mean_value, kappa, shock, private_value. You must enter it as a parameter exactly as it is written here or it won't work.

Set iteration range takes a yes or a no. If you select yes, you will be asked to select the minimum and maximum value to check for your variable and the increment you want to iterate with. If you select no, the default parameters are latency = 0, sim_length = 15000, arrival_rate = 0.075, tick_size = 1, reentry_rate = 0.005, mean_value = 100000, kappa = 0.05, shock = 150000000, private_value = 100000000.

Where output is saved:

The output of this file works the same as it did in experiments_latency. The observation files and simulation spec file are saved in folder saved as simulations/expname_testvar_testvarvalue

For example, if we were called the experiment test and were testing latency from 10 to 1000 with an increment of 100 a folder could be

simulations/test_latency_0. Inside that fold is a log folder that holds the log files.

When the simulation finishes, it will send you an email that contains information from running the simulation and information about the datachecks run. A sample email might look like this:

```
start time: Wed Jul  3 13:48:02 2013
Samples for simulations/trial_latency_0
Parsing
Samples for simulations/trial_latency_100
Parsing
Samples for simulations/trial_latency_200
Parsing
Samples for simulations/trial_latency_300
Parsing
Samples for simulations/trial_latency_400
Parsing
Samples for simulations/trial_latency_500
Parsing
Samples for simulations/trial_latency_600
Parsing
Samples for simulations/trial_latency_700
Parsing
Samples for simulations/trial_latency_800
Parsing
Samples for simulations/trial_latency_900
Parsing
Samples for simulations/trial_latency_1000
Parsing
end time: Wed Jul  3 13:48:32 2013
```

```
Test number of column headers equals equal max columns in data: FAILED
    Number of column headers do not match number of columns in data
Testing range of latency values: PASSED
Testing total undiscounted surplus at latency 0: PASSED
Testing transaction num at latency 0: PASSED
Testing total discounted surplus at latency 0 except central call: PASSED
Testing execution speed at latency 0: PASSED
Testing spreads of med nbbo at latency 0: PASSED
Testing avg vol (std dev of log of midquote prices) at latency 0: PASSED
Testing avg vol (log of std dev of midquote prices) at latency 0: PASSED
Testing avg vol (std dev of log returns) at latency 0: PASSED
Testing vol in markets (log of std dev of midquote prices) at latency 0: PASSED
Testing vol in markets (std dev of log returns) at latency 0: PASSED
ml
```

check-for-changes:

compile-src:

build-jar:

BUILD SUCCESSFUL

Total time: 0 seconds

>> Building... done

>> Running simulation 1... done

>> Running simulation 2... done

>> Building...

Buildfile: /home/hannoell/noelle/hft/build.xml

The build successful part will appear as many times as you chose to run the simulation