GPU

Jump to bottom

Joseph Lizier edited this page 5 minutes ago · 1 revision

How to use GPU estimators

Demos > GPU

JIDT now ships with GPU implementations for the KSG estimators (for mutual information and conditional mutual information, and measures using these such as transfer entropy and active information storage).

In order to use these, there are three steps to follow:

1. Install CUDA toolkit;
2. Make the JIDT GPU estimator library;
3. Call the JIDT GPU estimators from your code.

These are outlined in the following:

Install CUDA toolkit

Before making or using the JIDT GPU estimators, you need to have the NVIDIA CUDA toolkit installed. This may be packaged for your platform, e.g. the `nvidia-cuda-toolkit` package on ubuntu, or else obtained from NVIDIA

Make JIDT GPU library

On each platform you install JIDT and wish to run the GPU estimators, you should build the JIDT GPU library, by running `ant gpu`. You can test this with `ant gputest`

Call the JIDT GPU estimators

In order to use the JIDT GPU estimators, you simply set the property `"USE_GPU"` to `true` on the KSG estimator that you want to run in GPU mode rather than in CPU. That's all!

There are several ways to see the code for this in more detail:

1. Simple Java demos: see the `Example10GPUBenchmark.java` in the `demos/java/infodynamics/demos` folder; you can run this with `demos/java/example10GPUBenchmark.sh` and then `plotExample10BenchmarkResults.sh`
2. Unit tests: see the three JUnit tests `GPUCondMutualInfoTester.java`, `GPUMutualInfoTester.sh` and `GPUPerformanceTester.sh` in the folder `java/unittests/infodynamics/measures/continuous/kraskov`; you can run these with `ant gputest` from the top level folder
3. AutoAnalyser: For any of the relevant KSG estimators, set the property `"USE_GPU"` to `true` using the drop-down property menus; generate the code in your language of choice (Java, Matlab or Python) and then run the generated code in that language.

▸ **Pages** 36

- Home
- Getting started
  - Downloads
  - Installation

    - Documentation
    - Tutorial
    - Demos
- ImplementedMeasures
- Demos
    - Auto analyser demo
    - Simple Java demos
    - Non-Java environments
        - Matlab/Octave demos
        - Python demos
        - R demos
        - Julia demos
        - Clojure demos
    - GPU
    - Cellular Automata
    - Schreiber Transfer entropy demos
    - Detecting interaction lags
    - Null distributions
    - Interregional transfer
- Course (long)
- Tutorial (short)
- Non-Java environments
    - Matlab/Octave
        - Array conversion to/from Octave
    - Python
    - R
    - Julia
    - Clojure
- FAQs
- Miscellaneous
    - Related toolkits
    - Road map for new features
    - Extra features
- For serious developers!
    - Unit tests
    - Ant scripts
    - Making a new release
- Publications resulting

Clone this wiki locally

```
https://github.com/jlizier/jidt.wiki.git
```