Anaconda2-2019.03 (Jul. 4, 2019)

Usage

There so many detailed guides of conda/anaconda elsewhere on the internet. Please also check them.

Initialization

You can activate conda settings by loading initialization script.

bash, zsh:

```
% . /local/apl/lx/anaconda2-2019Jul/conda_init.sh
```

csh (tcsh):

```
% source /local/apl/lx/anaconda2-2019Jul/conda_init.csh
```

Upon this stage, you may be in the "base" environment. The environment name will be shown at the far left on your prompt like the following example.

```
(base) %
```

Environments

- base: base environment, can be considered as default environment.
  - python 2.7.16
  - numpy 1.16.4, scipy 1.2.1, pandas 0.24.2, pyyaml 5.1.1, pillow 6.0.0
  - tensorflow 1.14.0, keras 2.2.4, chainer 5.3.0, pytorch 1.1.0
    - GPU packages are not installed. This contains popular python packages for academic research and CPU version packages for deep learning (tensorflow, keras, pytorch, chainer).
- chainer_gpu: Chainer GPU environment.
  - python 2.7.16
  - chainer 5.3.0, cupy 6.0.0, cudatoolkit 9.0
- pytorch_gpu: PyTorch GPU environment.
  - python 2.7.16
  - pytorch(-gpu) 1.1.0, torchvision(-gpu) 0.3.0, cudatoolkit 9.0
- tensorflow_gpu: TensorFlow + Keras GPU environment.
  - python 2.7.16
  - tensorflow(-gpu) 1.13.1, keras(-gpu) 2.2.4, cudatoolkit 9.0

The installed packages in the environments above won't be updated. In other word, environments will be kept intact. New version of packages will be installed upon creating new environments or creating another anaconda installation.

Switching Environments

You can change environment by executing "conda active" command after the initialization of conda itself.

Example:

```
(base) % conda activate pytorch_gpu
(pytorch_gpu) %
```

Current environment name will be shown at the leftmost of the prompt regardless of your login shell type (csh or bash or zsh).
You can exit from the conda environment by "conda deactivate" command.

Example:

```
(pytorch_gpu) % conda deactivate
```

```
(base) % conda deactivate
%
```

Creating your own environment

Users cannot add packages in this anaconda directory, since this anaconda is installed in the system domain (/local/apl/lx).
Instead, you can build your own conda environment using this pre-installed anaconda.
The new environment files will be placed under ~/.conda directory in the default setting.

Example: new anaconda env creation ("myenv" is your new environment name):

```
(base) % conda create -n myenv anaconda
```

Example: clone "base" environment ("myenv" is your new environment name):

```
(base) % conda create -n myenv --clone base
```

(note: conda_init.sh, conda_init.csh, samples_rccs/ will also be cloned. You can remove those files after the cloning.)

You can also directly install anaconda (or miniconda) under your home directory.
If you need a very special environment, this way may be a better solution.

Run MNIST sample

MNIST sample scripts are available in /local/apl/lx/anaconda2-2019Jul/samples_rccs directory.
You can run them just by copying directory and submit by using jsub.
The script initializes conda, chooses environment if necessary, and then run MNIST sample.

Note: if your login shell is csh and try to run GPU sample, you may need to modify script (see script).

tensorflow exmaple:

```
% cp -r /local/apl/lx/anaconda2-2019Jul/samples_rccs/tensorflow/ ~/tensorflow_sample_mnist
% cd ~/tensorflow_sample_mnist
% jsub -q PN sample-tensorflow-cpu.sh
```

(MNIST data is downloaded on /local/apl/lx/anaconda2-2019Jul/samples_rccs/datasets in advance. Please note that computation nodes cannot access internet.)

Extra: Install packages without conda envs

You can also install packages under your home directory via "pip" or "pip3" command without conda.

Example: (--user is the key option, this is required.)

```
% pip install cupy-cuda91 chainer --user
```

(use pip3 instead of pip if you are using Python3)

Installation Details

Webpage

https://www.anaconda.com/

Version

2019.03

Files

- Anaconda2-2019.03-Linux-x86_64.sh

Installation Procedure

The following commands are run by root. Most of logs are cut, since it is too long.

base environment (non-GPU env)

```
% sh Anaconda2-2019.02-Linux-x86_64.sh
...
Anaconda2 will now be installed into this location:
/root/anaconda2

  - Press ENTER to confirm the location
  - Press CTRL-C to abort the installation
  - Or specify a different location below

[/root/anaconda2] >>> /local/apl/lx/anaconda2-2019Jul
...
Do you wish the installer to initialize Anaconda2
by running conda init? [yes|no]
[no] >>> no
% /local/apl/lx/anaconda2-2019Jul/bin/conda shell.bash hook > /local/apl/lx/anaconda2-2019Jul/conda_init.sh
% /local/apl/lx/anaconda2-2019Jul/bin/conda shell.csh hook > /local/apl/lx/anaconda2-2019Jul/conda_init.csh
% . /local/apl/lx/anaconda2-2019Jul/conda_init.sh
(base) % conda update --all
...
The following packages will be DOWNGRADED:

  anaconda                          2019.03-py27_0 --> custom-py27h4a00acb_0
  contextlib2                     0.5.5-py27hbf4c468_0 --> 0.5.5-py27_0
  cycler                        0.10.0-py27hc7354d3_0 --> 0.10.0-py27_0
  et_xmlfile                      1.0.1-py27h75840f5_0 --> 1.0.1-py27_0
  locket                         0.2.0-py27h73929a2_1 --> 0.2.0-py27_1
  nbformat                        4.4.0-py27hed7f2b2_0 --> 4.4.0-py27_0
  prompt_toolkit                   1.0.15-py27h1b593e1_0 --> 1.0.15-py27_0
  singledispatch                  3.4.0.3-py27h9bcb476_0 --> 3.4.0.3-py27_0
  traitlets                      4.3.2-py27hd6ce930_0 --> 4.3.2-py27_0
  unicodecsv                      0.14.1-py27h5062da9_0 --> 0.14.1-py27_0
  wcwidth                         0.1.7-py27h9e3e1ab_0 --> 0.1.7-py27_0
...
(base) % conda install tensorflow keras chainer opencv
...
  chainer            pkgs/main/linux-64::chainer-5.3.0-py27_0
  keras              pkgs/main/linux-64::keras-2.2.4-0
  keras-applications pkgs/main/noarch::keras-applications-1.0.8-py_0
  keras-base         pkgs/main/linux-64::keras-base-2.2.4-py27_0
  keras-preprocessi~ pkgs/main/noarch::keras-preprocessing-1.1.0-py_1
  tensorboard        pkgs/main/linux-64::tensorboard-1.14.0-py27hf484d3e_0
  tensorflow         pkgs/main/linux-64::tensorflow-1.14.0-mkl_py27h957988d_0
  tensorflow-base    pkgs/main/linux-64::tensorflow-base-1.14.0-mkl_py27h7ce6ba3_0
  tensorflow-estima~ pkgs/main/noarch::tensorflow-estimator-1.14.0-py_0
...
The following packages will be REMOVED:

  pytables-3.5.2-py27h71ec239_1

The following packages will be DOWNGRADED:

  h5py                           2.9.0-py27h7918eee_0 --> 2.8.0-py27h989c5e5_3
  hdf5                             1.10.4-hb1b8bf9_0 --> 1.10.2-hba1933b_1

...
```

```
(base) % conda install -c pytorch pytorch-cpu torchvision-cpu
...
  pytorch-cpu        pytorch/linux-64::pytorch-cpu-1.1.0-py2.7_cpu_0
  torchvision-cpu    pytorch/linux-64::torchvision-cpu-0.3.0-py27_cuNone_1
...
(base) % conda install pytables
...
  pytables           pkgs/main/linux-64::pytables-3.4.4-py27ha205bf6_0
...
```

Chainer GPU env

```
% . /local/apl/lx/anaconda2-2019Jul/conda_init.sh
(base) % conda create --name chainer_gpu --clone base
...
(base) % conda activate chainer_gpu
(chainer_gpu) % conda remove chainer tensorflow keras pytorch-cpu torchvision-cpu
...
(chainer_gpu) % conda install cudatoolkit=9.0 cupy chainer
...
The following NEW packages will be INSTALLED:

  chainer            pkgs/main/linux-64::chainer-5.3.0-py27_0
  cudatoolkit        pkgs/main/linux-64::cudatoolkit-9.0-h13b8566_0
  cudnn              pkgs/main/linux-64::cudnn-7.6.0-cuda9.0_0
  cupy               pkgs/main/linux-64::cupy-6.0.0-py27h686fdb1_0
  fastrlock          pkgs/main/linux-64::fastrlock-0.4-py27he6710b0_0
  nccl               pkgs/main/linux-64::nccl-1.3.5-cuda9.0_0

...
```

PyTorch GPU env

```
% . /local/apl/lx/anaconda2-2019Jul/conda_init.sh
(base) % conda create --name pytorch_gpu --clone base
...
(base) % conda activate pytorch_gpu
(pytorch_gpu) % conda remove chainer tensorflow keras pytorch-cpu torchvision-cpu
...
(pytorch_gpu) % conda install -c pytorch cudatoolkit=9.0 pytorch torchvision cudnn
...
The following NEW packages will be INSTALLED:

  cudatoolkit        pkgs/main/linux-64::cudatoolkit-9.0-h13b8566_0
  cudnn              pkgs/main/linux-64::cudnn-7.6.0-cuda9.0_0
  pytorch            pytorch/linux-64::pytorch-1.1.0-py2.7_cuda9.0.176_cudnn7.5.1_0
  torchvision        pytorch/linux-64::torchvision-0.3.0-py27_cu9.0.176_1
...
```

(cudnn may be unnecessary for pytorch-gpu?)

TensorFlow + Keras GPU env

```
% . /local/apl/lx/anaconda2-2019Jul/conda_init.sh
(base) % conda create --name tensorflow_gpu anaconda
..
(base) % conda activate tensorflow_gpu
(tensorflow_gpu) % conda install anaconda=custom=py27h4a00acb_0
...
(tensorflow_gpu) % conda install cudatoolkit=9.0 tensorflow-gpu keras-gpu
...
  keras-applications pkgs/main/noarch::keras-applications-1.0.8-py_0
  keras-base         pkgs/main/linux-64::keras-base-2.2.4-py37_0
  keras-gpu          pkgs/main/linux-64::keras-gpu-2.2.4-0
  keras-preprocessi~ pkgs/main/noarch::keras-preprocessing-1.1.0-py_1
...
```

```
  tensorboard        pkgs/main/linux-64::tensorboard-1.13.1-py37hf484d3e_0
  tensorflow         pkgs/main/linux-64::tensorflow-1.13.1-gpu_py37hd37c573_0
  tensorflow-base    pkgs/main/linux-64::tensorflow-base-1.13.1-gpu_py37h8f37b9b_0
  tensorflow-estima~ pkgs/main/noarch::tensorflow-estimator-1.13.0-py_0
  tensorflow-gpu     pkgs/main/linux-64::tensorflow-gpu-1.13.1-h0d30ee6_0
...
(tensorflow_gpu) % conda install opencv
...
The following packages will be REMOVED:

  pytables-3.5.1-py27h71ec239_0

The following packages will be DOWNGRADED:

  h5py                        2.9.0-py27h7918eee_0 --> 2.8.0-py27h989c5e5_3
  hdf5                          1.10.4-hb1b8bf9_0 --> 1.10.2-hba1933b_1

...
(tensorflow_gpu) % conda install pytables
...
The following NEW packages will be INSTALLED:

  pytables           pkgs/main/linux-64::pytables-3.4.4-py27ha205bf6_0
...
```

(Cloned environment of "base" encountered dependency problem. So we employed the procedure above unlike the other two.)