

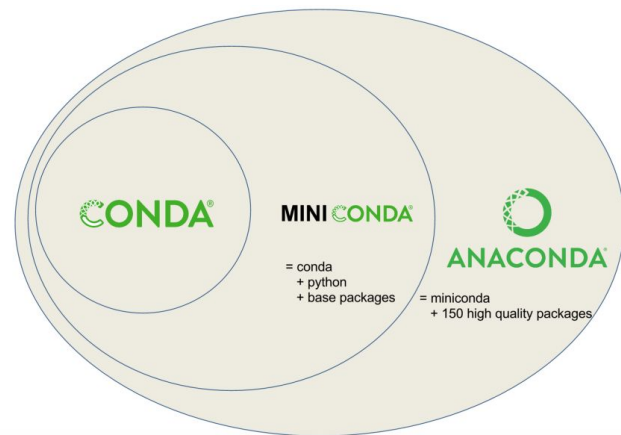
Setting up



Summer of #UQAI: 2021: Getting started in AI : Gayan Kulatilleke : g.kulatilleke@uq.edu.au

MINI CONDA[®] Introduction

venv	pip	conda
Environment Mgr		Environment Mgr
	Package Mgr	Package Mgr
		different python versions
		Compiling and installing non-python packages (gcc, CUDA, scipy/numpy)



written in C and just wrapped as a python library

need to be compiled locally for maximum performance + proper linking with libraries like glibc

Pip: Compiles everything from source. Installs binary wheels if `pip install --use-wheel`.

Why do I need this?

- Working on multiple projects at once (python 2.7 and python 3.8)
- Separate projects from each other
- Provide different environments for each project

Download and install

<https://docs.conda.io/en/latest/miniconda.html>

```
wget https://repo.anaconda.com/miniconda/Miniconda-1.6.0-Linux-x86_64.sh
```

```
bash Miniconda3-latest-Linux-x86_64.sh
```

```
Do you accept the license terms? [yes|no]  
[no] >>> yes
```

```
Miniconda3 will now be installed into this location:
```

```
/clusterdata/uqgkulat/
```

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

```
[/clusterdata/uqgkulat/miniconda3] >>> ./miniconda
```

```
Do you wish the installer to initialize Miniconda3  
by running conda init? [yes|no] >>> yes
```

Adds conda to the shell

```
(base) % ← restart shell or source .bashrc
```

Creating Environments

```
conda create --name conda-env python
```

```
conda create -n conda-env python=3.7
```

```
conda create -n conda-env numpy requests
```

```
conda create -n conda-env python=3.7 numpy=1.16.1
```

```
Conda env list
```

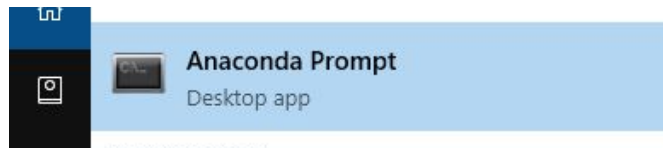
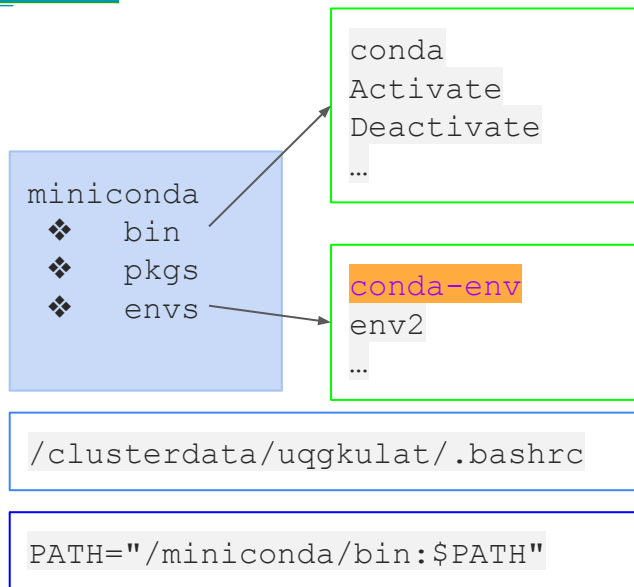
Activate / Deactivate Environment

```
> conda activate conda-env
```

```
(conda-env) %
```

```
conda deactivate
```

```
>
```



Installing Packages

- From inside an active environment

```
(conda-env) % conda install pandas=0.24.1
```

```
conda create -n tf tensorflow
conda create -n tf-gpu tensorflow-gpu
conda create -n tf-gpu-cudaX tensorflow-gpu cudatoolkit=9.0 #non-default CUDA version
```

```
(conda-env) % conda install --channel conda-forge opencv # Or -c
(conda-env) % conda update pandas
```

```
(conda-env) % conda list #list of packages
```

- ~~From your default shell~~

```
% conda install -n conda-env pandas=0.24.1 # Or -p /path/to/env
```

Managing Environments

```
(conda-env) % conda env export --file environment.yml
conda env create -n conda-env -f /path/to/environment.yml
```

package Repositories

- [Anaconda Repository](#) - default
- [Anaconda Cloud](#) - hosts Conda packages provided by third party repositories like [Conda-Forge](#).

```
name: null # Our env was made with --prefix
channels:
  - conda-forge # We added a third party channel
  - defaults
dependencies:
  - numpy=1.16.3=py37h926163e_0
  - pip=19.1.1=py37_0
  - pip: # Packages installed from PyPI
    - requests==2.21.0
prefix: /Users/user-name/data-science/project-name/conda-env
```

Example : different python version and gcc compiler.

- Creating a conda env

```
(base) conda create --name conda-env python
conda activate conda-env
(conda-env) python --version
Python 3.10.0
```

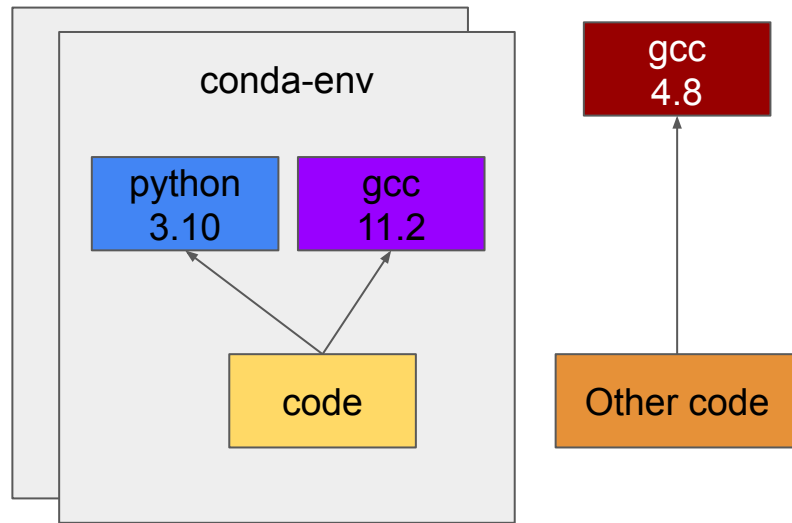
```
(conda-env) gcc --version
gcc (GCC) 4.8.5 20150623 (Red Hat 4.8.5-39)
```

```
conda install -c conda-forge gcc
(conda-env) gcc --version
gcc (GCC) 11.2.0
```

```
(conda-env) conda deactivate
```

- Current versions on wiener

```
(base) python --version
Python 3.9.5
(base) gcc --version
gcc (GCC) 4.8.5 20150623 (Red Hat 4.8.5-39)
```



Remember : one liners to set up TF

```
conda create -n tf tensorflow
conda create -n tf-gpu tensorflow-gpu
conda create -n tf-gpu-cudaX tensorflow-gpu cudatoolkit=9.0 #non-default CUDA version
```

Notes

CVL@weiner gives a linux VM

Interactive SLURM

```
alias sint='salloc -N 1 -J EAI -n 1 -c 4 --mem=32GB --partition=gpu --gres=gpu:1  
--time=02:00:00 srun --pty /bin/bash -l'
```

Filezilla - set # connectons = 1 to use with MFA

weights and Biases