



Exploratory Data Science

Course 1

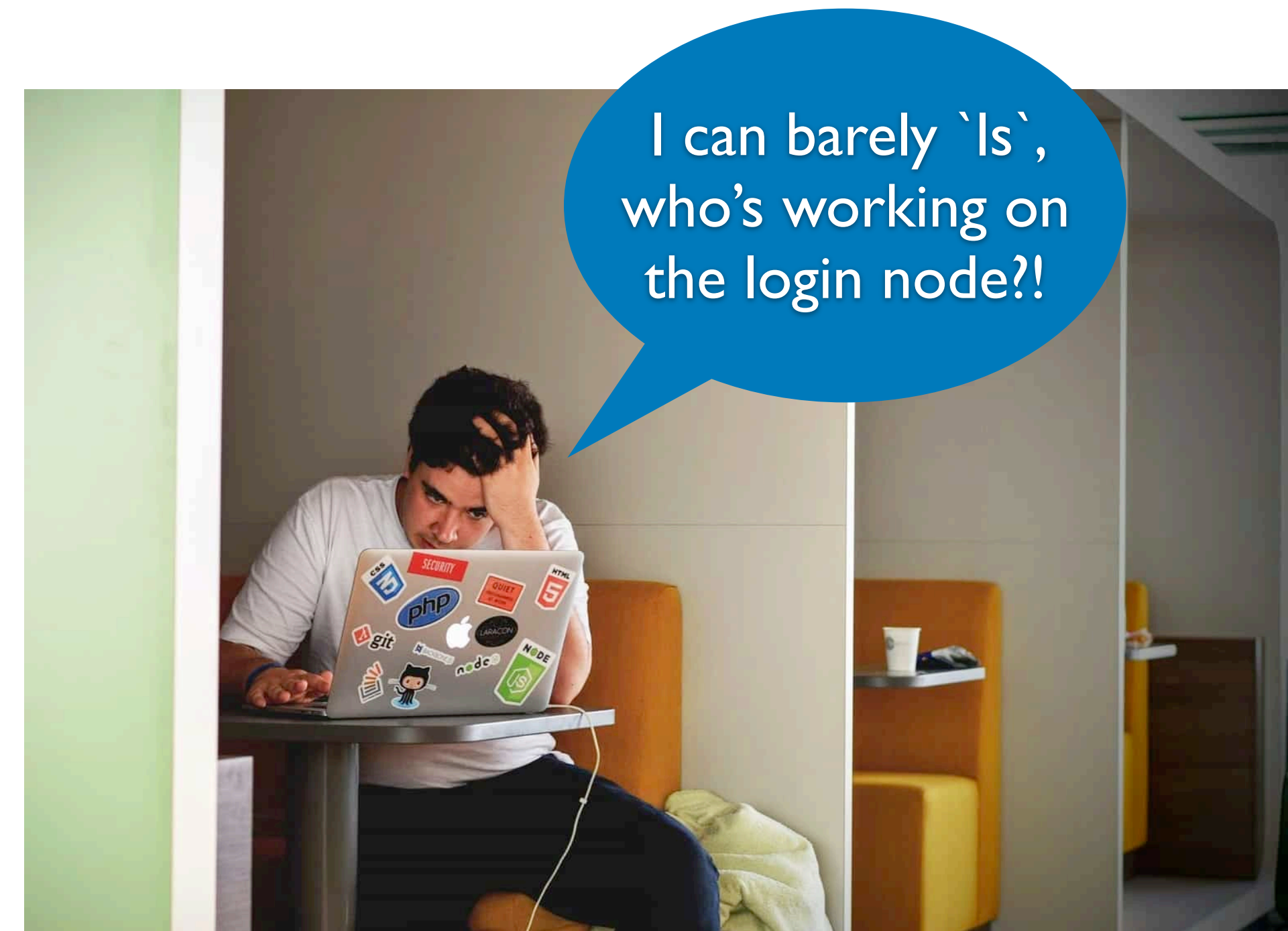
Brian Pollack

9/24/19

Working Nodes at PSC



- ★ **When using a computer cluster (like PSC), you must work on *working nodes* instead of the *login node*.**
 - Working nodes have access to cpu/gpu/ram resources that the login does not. Running intensive jobs on the login node is **bad**, hurts your coworkers, and will get you angry emails from IT.
- ★ **Working nodes come in two flavors: Interactive and Batch.**
 - Batch is for queuing up many jobs and running them to completion.
 - Interactive is for active development, testing, exploration.



Student with deadline

The Interact Command



```
user:$ interact -p DBMI --egress -t 02:00:00 -A XXXXXX --mem=120GB
```

The Interact Command



```
user:$ interact -p DBMI --egress -t 02:00:00 -A XXXXXX --mem=120GB
```

Command to request
interactive node

The Interact Command



```
user:$ interact -p DBMI --egress -t 02:00:00 -A XXXXXX --mem=120GB
```

Command to request
interactive node

Designate partition
(DBMI, RM, GPU, GPU-AI ...)
Different partitions have
different properties

The Interact Command



```
user:$ interact -p DBMI --egress -t 02:00:00 -A XXXXXX --mem=120GB
```

Command to request
interactive node

Allow access to the
outside world.
(downloads from the web)

Designate partition
(DBMI, RM, GPU, GPU-AI ...)
Different partitions have
different properties

The Interact Command



```
user:$ interact -p DBMI --egress -t 02:00:00 -A XXXXXX --mem=120GB
```

Command to request
interactive node

Allow access to the
outside world.
(downloads from the web)

Designate partition
(DBMI, RM, GPU, GPU-AI ...)
Different partitions have
different properties

Amount of time before
sessions ends.
Max: 8 Hours

The Interact Command



```
user:$ interact -p DBMI --egress -t 02:00:00 -A XXXXXX --mem=120GB
```

Command to request
interactive node

Allow access to the
outside world.
(downloads from the web)

Project ID
(given to you by your adviser/PI)

Designate partition
(DBMI, RM, GPU, GPU-AI ...)
Different partitions have
different properties

Amount of time before
sessions ends.
Max: 8 Hours

The Interact Command



```
user:$ interact -p DBMI --egress -t 02:00:00 -A XXXXXX --mem=120GB
```

Command to request
interactive node

Allow access to the
outside world.
(downloads from the web)

Project ID
(given to you by your adviser/PI)

Designate partition
(DBMI, RM, GPU, GPU-AI ...)
Different partitions have
different properties

Amount of time before
sessions ends.
Max: 8 Hours

Memory Allocation.
We typically use a lot of memory.
More memory, less problems.

Jupyter on PSC



- ★ When running Jupyter Notebook on PSC, we can access the notebook via tunneling or OnDemand.
- For tunneling, we use the `startupjupyter` script located in `helper_files`. This script allows you to create an ssh tunnel to the jupyter notebook and display it in your local browser.

```
(data_course) bpollack@dr008:/pghbio/dbmi/batmanlab/bpollack/data_course$ startupjupyter
Your Jupyter Notebook is ready for use.
-----
Step 1: action.
Mac/Linux users: launch another terminal and paste the following command:
ssh -L 8888:dr008.opa.bridges.psc.edu:8888 bridges.psc.edu -l bpollack
Windows users: run cmd then cd to your PuTTY directory then paste the following command:
plink -L 8888:dr008.opa.bridges.psc.edu:8888 bridges.psc.edu -l bpollack
-----
Step 2: Open a browser on your computer to http://localhost:8888
Step 3: Enjoy Jupyter Notebook!
ssh -L 8888:dr008.opa.bridges.psc.edu:8888 bridges.psc.edu -l bpollack[I 10:40:09.918 NotebookApp]
[jupyter_nbextensions_configurator] enabled 0.4.1
[I 10:40:09.919 NotebookApp] Serving notebooks from local directory: /pghbio/dbmi/batmanlab/bpollack/data_course
[I 10:40:09.919 NotebookApp] The Jupyter Notebook is running at:
[I 10:40:09.919 NotebookApp] http://dr008.pvt.bridges.psc.edu:8888/?token=af8501510917f5ce0012e88e6ab4881c53ba4137b046da37
[I 10:40:09.919 NotebookApp] or http://127.0.0.1:8888/?token=af8501510917f5ce0012e88e6ab4881c53ba4137b046da37
[I 10:40:09.919 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 10:40:10.002 NotebookApp]

To access the notebook, open this file in a browser:
file:///home/bpollack/.jupyter/nbserver-10330-open.html
Or copy and paste one of these URLs:
http://dr008.pvt.bridges.psc.edu:8888/?token=af8501510917f5ce0012e88e6ab4881c53ba4137b046da37
or http://127.0.0.1:8888/?token=af8501510917f5ce0012e88e6ab4881c53ba4137b046da37
```

ssh tunneling command

web address

login token (needed for first login)

Painful Surprises



The following scary stories are based on TRUE events.

The names and projects have been changed to protect the victims...

Painful Surprises



★ **Accidental Deletion:**

- Billy wants to remove some files from a git repo he downloaded.
- Git repo files can be protected and require a “Y” for each file you delete. How annoying! I don’t have all day! We need to delete faster!



Painful Surprises

★ Accidental Deletion:

- Billy wants to remove some files from a git repo he downloaded.
- Git repo files can be protected and require a “Y” for each file you delete. How annoying! I don’t have all day! We need to delete faster!

```
$ rm -r some_repo/junk*  
rm: remove write-protected regular file [x] ?
```


Painful Surprises



★ Accidental Deletion:

- Billy wants to remove some files from a git repo he downloaded.
- Git repo files can be protected and require a “Y” for each file you delete. How annoying! I don’t have all day! We need to delete faster!

```
$ rm -r some_repo/junk*  
rm: remove write-protected regular file [x] ?  
$ rm -rf some_repo/junk *  
  
...  
  
...  
  
...
```




Painful Surprises

★ Accidental Deletion:

- Billy wants to remove some files from a git repo he downloaded.
- Git repo files can be protected and require a “Y” for each file you delete. How annoying! I don’t have all day! We need to delete faster!

```
$ rm -r some_repo/junk*  
rm: remove write-protected regular file [x] ?  
$ rm -rf some_repo/junk *  
...  
...  
...
```



Painful Surprises

★ Accidental Deletion:

- Billy wants to remove some files from a git repo he downloaded.
- Git repo files can be protected and require a “Y” for each file you delete. How annoying! I don’t have all day! We need to delete faster!

```
$ rm -r some_repo/junk*  
rm: remove write-protected regular file [x] ?  
$ rm -rf some_repo/junk *  
...  
...  
...
```





Painful Surprises

★ Accidental Deletion:

- Billy wants to remove some files from a git repo he downloaded.
- Git repo files can be protected and require a “Y” for each file you delete. How annoying! I don’t have all day! We need to delete faster!

```
$ rm -r some_repo/junk*  
rm: remove write-protected regular file [x] ?  
$ rm -rf some_repo/junk *  
...  
...  
...
```



Congrats! You accidentally deleted your entire working directory!

Painful Surprises



★ **Cross-project Contamination**

- Amy doesn't bother with virtual environments, who has time for that?
- Amy installs the newest version of an ML package for her new project, then goes back to keep working on her thesis project...

Painful Surprises



★ Cross-project Contamination

- Amy doesn't bother with virtual environments, who has time for that?
- Amy installs the newest version of an ML package for her new project, then goes back to keep working on her thesis project...

```
$ conda install xtremetorch-deepnetz  
The following packages will be UPGRADED:  
pytorch 1.0.1 -> pytorch 2.5.2  
...
```




Painful Surprises

★ Cross-project Contamination

- Amy doesn't bother with virtual environments, who has time for that?
- Amy installs the newest version of an ML package for her new project, then goes back to keep working on her thesis project...

```
$ conda install xtremetorch-deepnetz
The following packages will be UPGRADED:
pytorch 1.0.1 -> pytorch 2.5.2
...
$ python do_thesis.py
RuntimeError: cuda runtime error (30) :
unknown error at /amy2019/thesis/codes/src/
codes2/do_thesis.py:188
```




Painful Surprises

★ Cross-project Contamination

- Amy doesn't bother with virtual environments, who has time for that?
- Amy installs the newest version of an ML package for her new project, then goes back to keep working on her thesis project...

```
$ conda install xtremetorch-deepnetz
The following packages will be UPGRADED:
pytorch 1.0.1 -> pytorch 2.5.2
...
$ python do_thesis.py
RuntimeError: cuda runtime error (30) :
unknown error at /amy2019/thesis/codes/src/
codes2/do_thesis.py:188
```



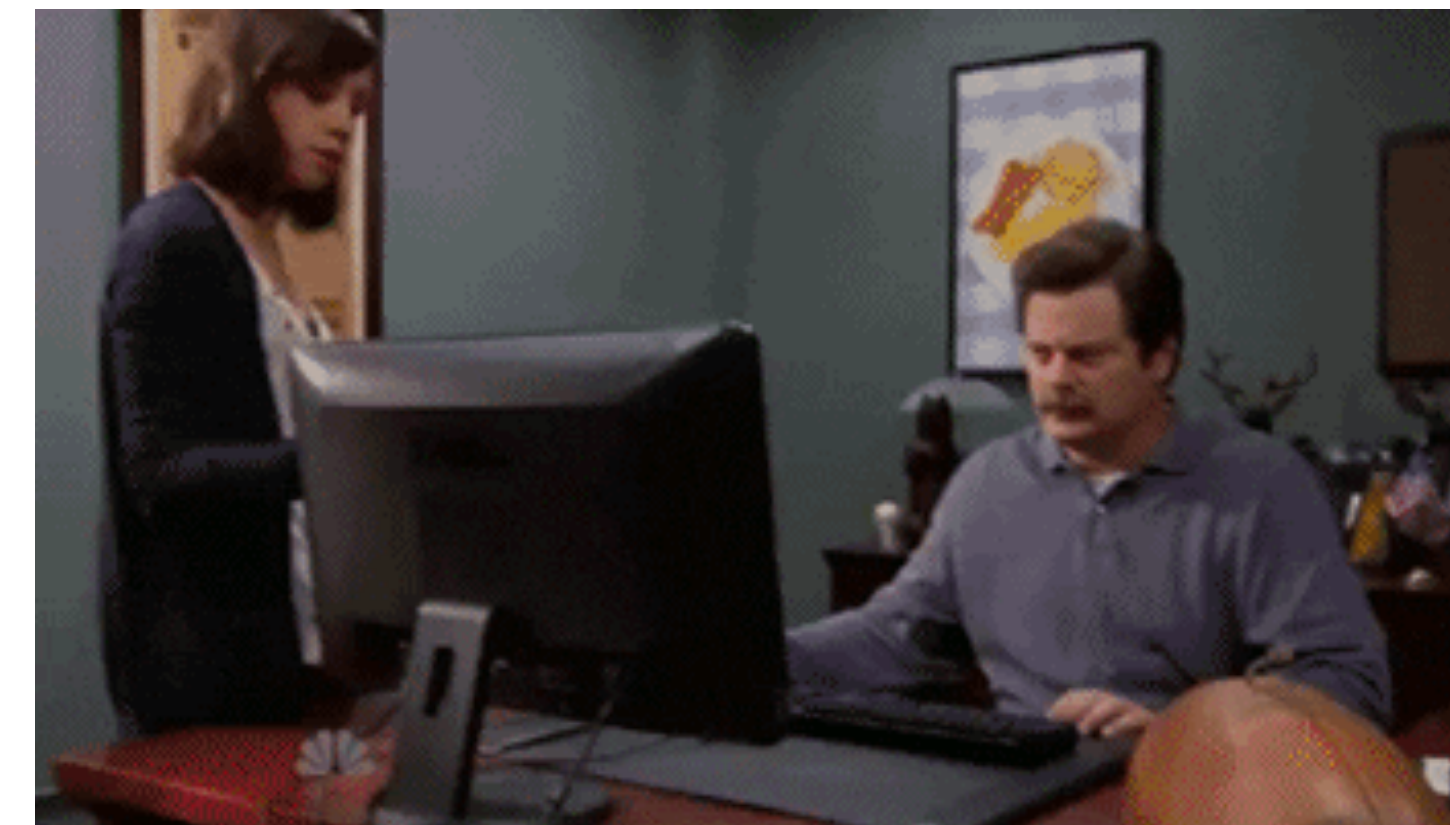


Painful Surprises

★ Cross-project Contamination

- Amy doesn't bother with virtual environments, who has time for that?
- Amy installs the newest version of an ML package for her new project, then goes back to keep working on her thesis project...

```
$ conda install xtremetorch-deepnetz
The following packages will be UPGRADED:
pytorch 1.0.1 -> pytorch 2.5.2
...
$ python do_thesis.py
RuntimeError: cuda runtime error (30) :
unknown error at /amy2019/thesis/codes/src/
codes2/do_thesis.py:188
```



Great Job! You broke your thesis project!

Painful Surprises



★ Unexpected Results

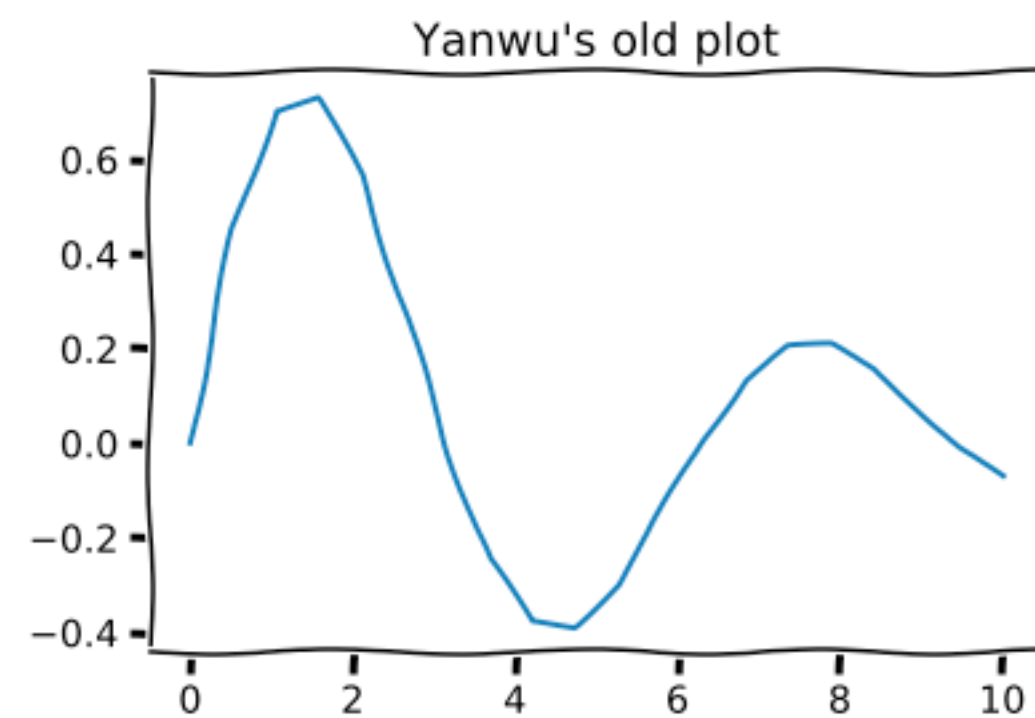
- Yanwu has been working on his project for over a year. He's made many changes since the initial results.
- His boss asks him to redo an early plot for a new conference submission...

Painful Surprises



★ Unexpected Results

- Yanwu has been working on his project for over a year. He's made many changes since the initial results.
- His boss asks him to redo an early plot for a new conference submission...

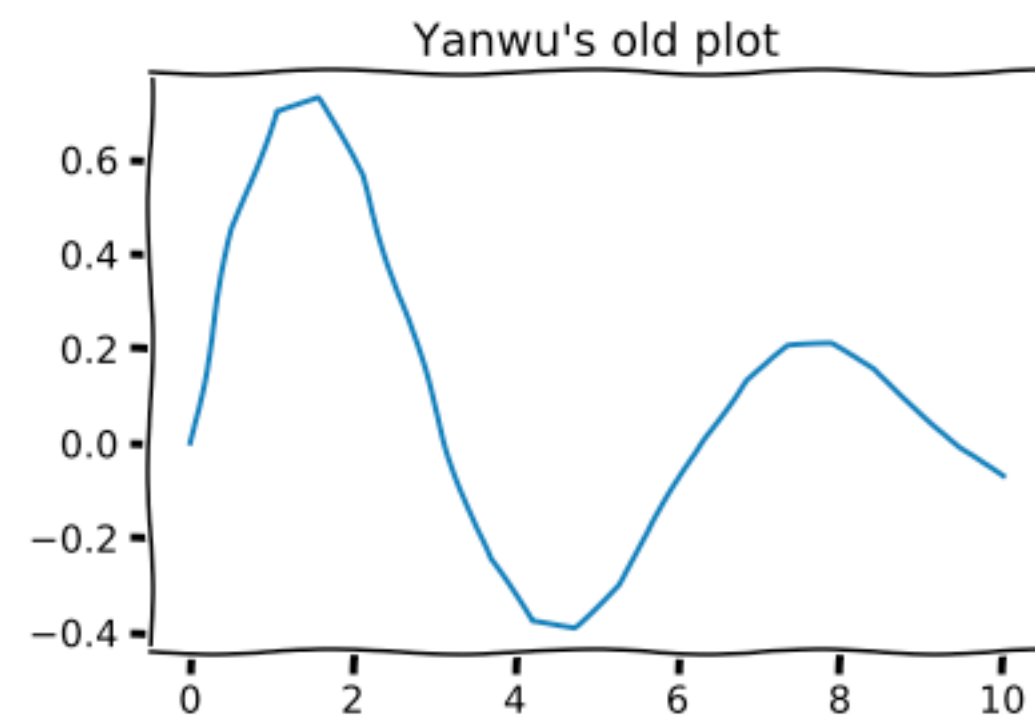


Painful Surprises

★ Unexpected Results

- Yanwu has been working on his project for over a year. He's made many changes since the initial results.
- His boss asks him to redo an early plot for a new conference submission...

```
$ python make_cool_plot_v1.py
```

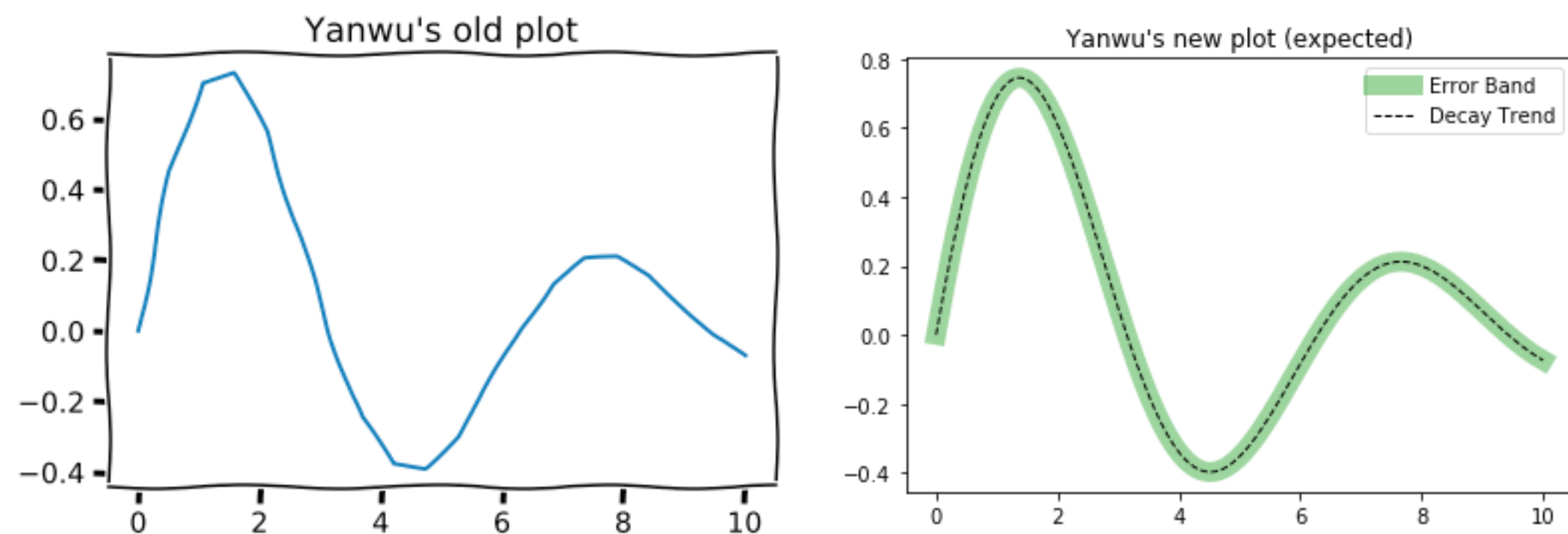


Painful Surprises

★ Unexpected Results

- Yanwu has been working on his project for over a year. He's made many changes since the initial results.
- His boss asks him to redo an early plot for a new conference submission...

```
$ python make_cool_plot_v1.py
```

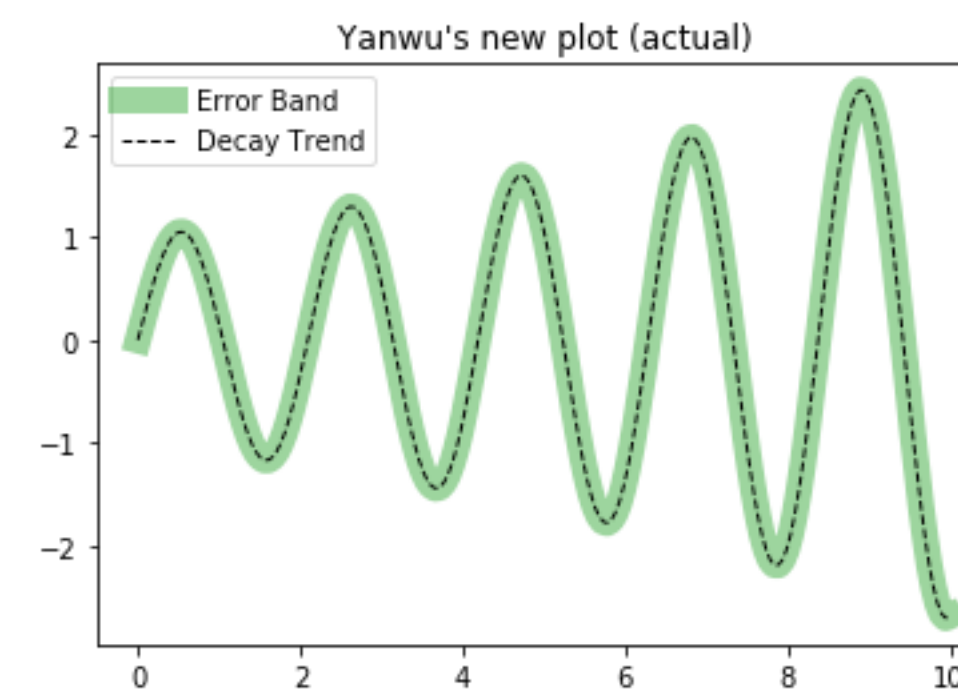
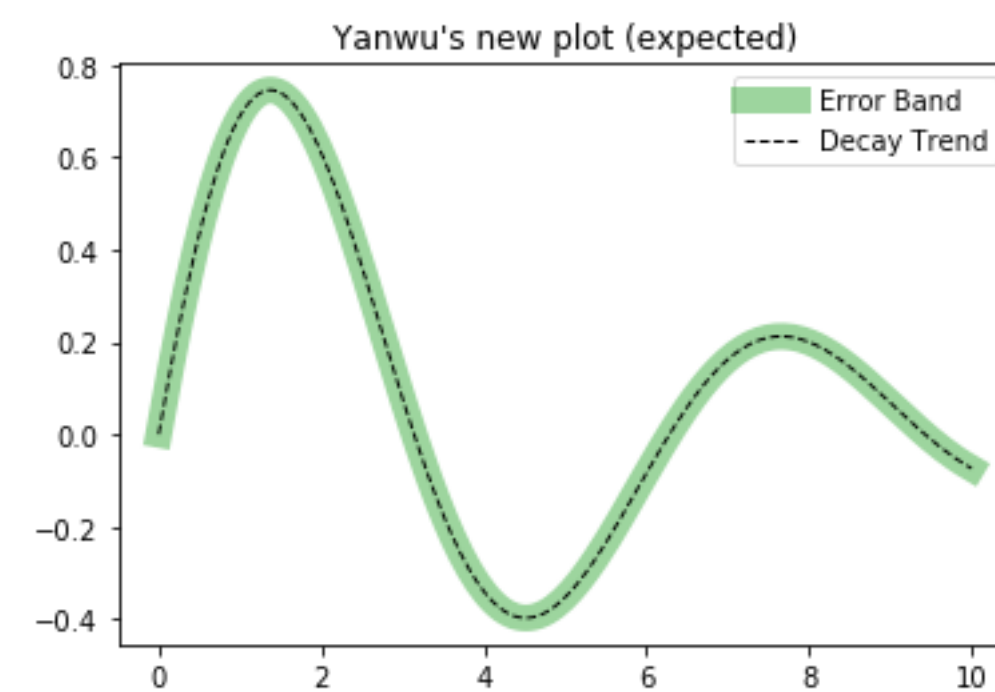
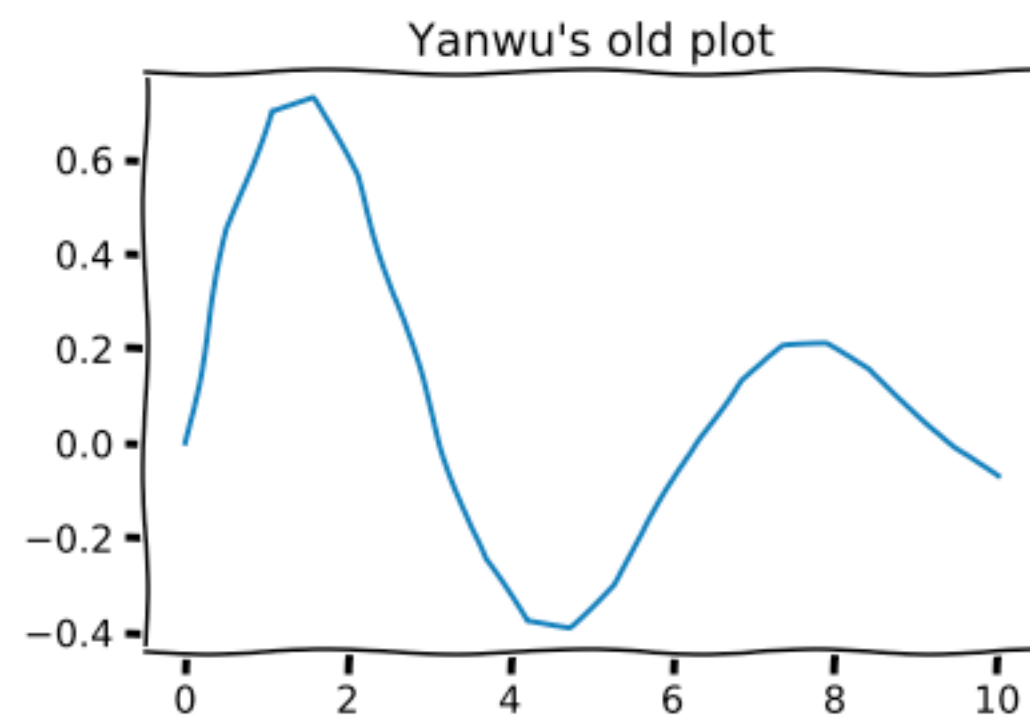


Painful Surprises

★ Unexpected Results

- Yanwu has been working on his project for over a year. He's made many changes since the initial results.
- His boss asks him to redo an early plot for a new conference submission...

```
$ python make_cool_plot_v1.py
```

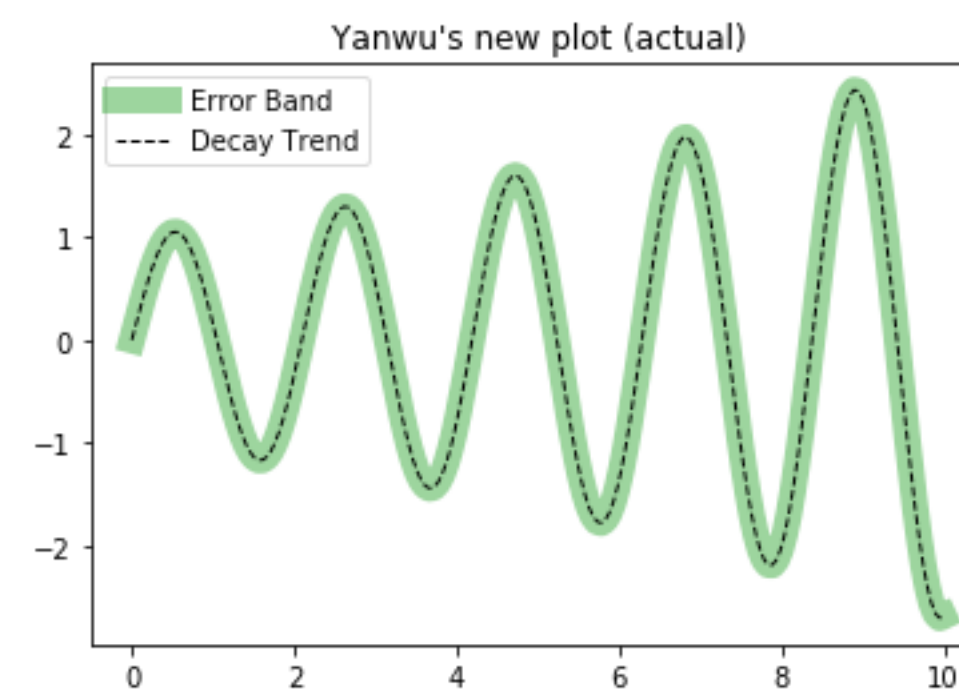
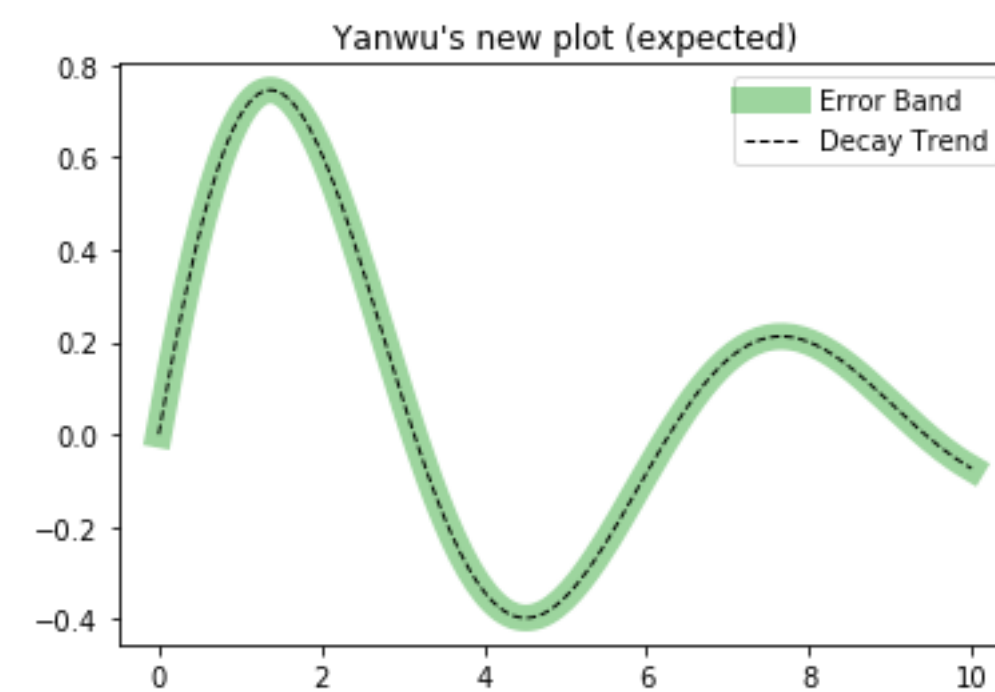
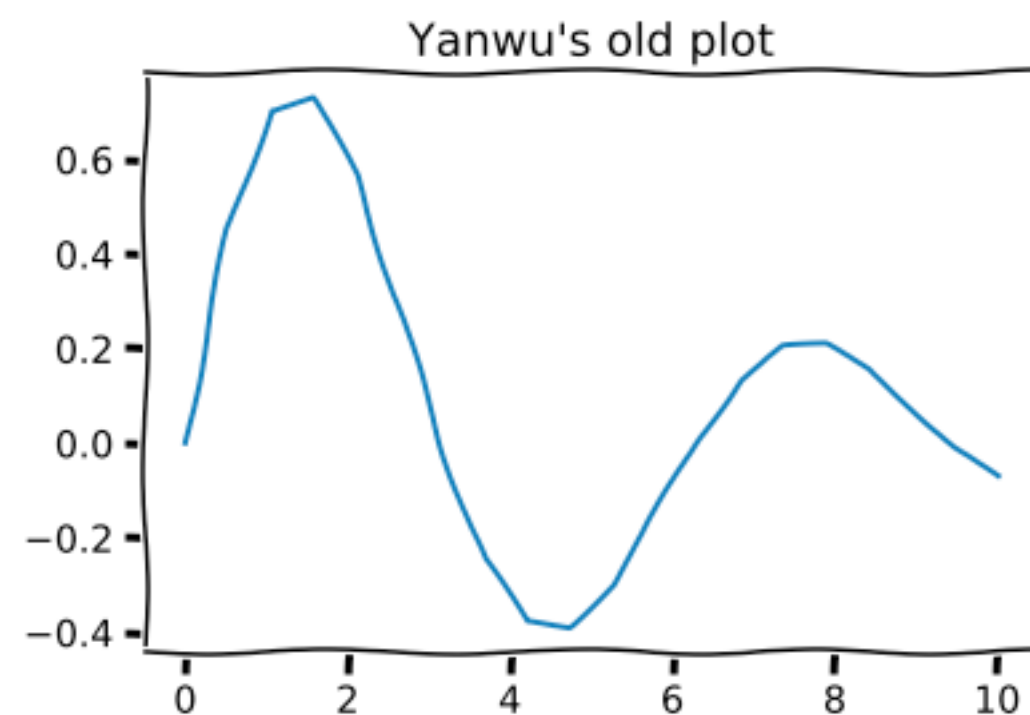


Painful Surprises

★ Unexpected Results

- Yanwu has been working on his project for over a year. He's made many changes since the initial results.
- His boss asks him to redo an early plot for a new conference submission...

```
$ python make_cool_plot_v1.py
```



Hooray! You can't reproduce your results!

Project Structure



```
my_project/  
  setup.py  
  foo_bar/  
    __init__.py  
    foo/  
      __init__.py  
      foo_module_1.py  
      foo_module_2.py  
      foo_module_3.py  
    bar/  
      __init__.py  
      bar_module.py  
      ...
```




Project Structure

Top Level Folder (name of git repo, for instance).

my_project/

setup.py

foo_bar/

__init__.py

foo/

__init__.py

foo_module_1.py

foo_module_2.py

foo_module_3.py

bar/

__init__.py

bar_module.py

...



Project Structure

Top Level Folder (name of git repo, for instance).

Needed for installing package.

my_project/

setup.py

foo_bar/

__init__.py

foo/

__init__.py

foo_module_1.py

foo_module_2.py

foo_module_3.py

bar/

__init__.py

bar_module.py

...



Project Structure

Top Level Folder (name of git repo, for instance).

Needed for installing package.

Folder that contains actual modules
(not docs, not tests, not data).

This is your top-level package name.

```
my_project/  
  setup.py  
  foo_bar/  
    __init__.py  
    foo/  
      __init__.py  
      foo_module_1.py  
      foo_module_2.py  
      foo_module_3.py  
    bar/  
      __init__.py  
      bar_module.py  
      ...
```




Project Structure

Top Level Folder (name of git repo, for instance).

Needed for installing package.

Folder that contains actual modules
(not docs, not tests, not data).

This is your top-level package name.

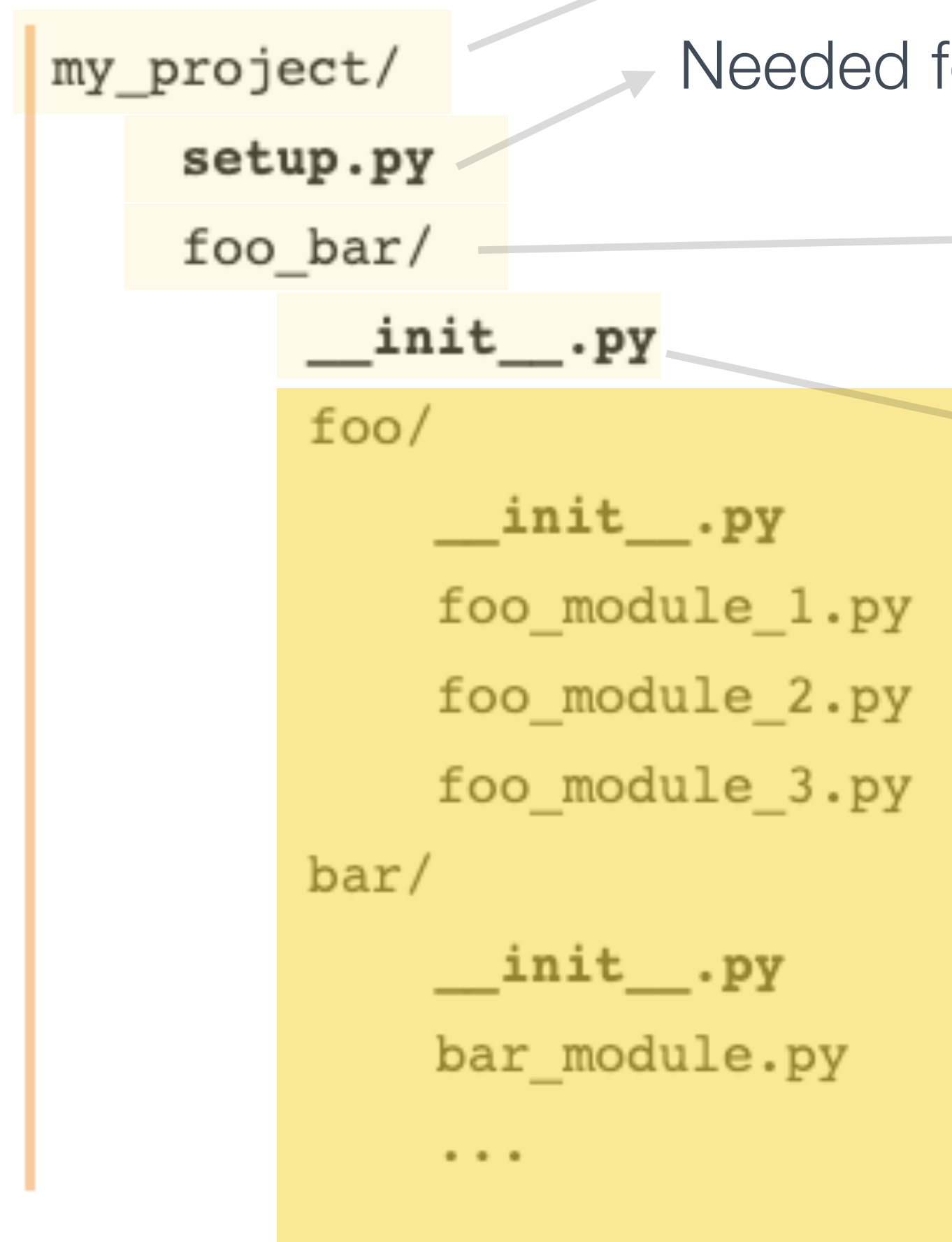
Needed for locating modules.

```
my_project/  
  setup.py  
  foo_bar/  
    __init__.py  
    foo/  
      __init__.py  
      foo_module_1.py  
      foo_module_2.py  
      foo_module_3.py  
    bar/  
      __init__.py  
      bar_module.py  
      ...
```



Project Structure

Top Level Folder (name of git repo, for instance).



Needed for installing package.

Folder that contains actual modules
(not docs, not tests, not data).

This is your top-level package name.

Needed for locating modules.

Modules and subdirectories.

Each subdir needs its own `__init__.py`