SSH Guide for Projects at CGM

Welcome to CGM!

To build and run your project you'll need a GPU. Soon you'll be given access to a Linux machine with a GPU for that purpose. You will not work directly on that machine, but on your personal computer, connected to a remote server (the one at the lab).

Follow this guide for a smooth beginning of your project.

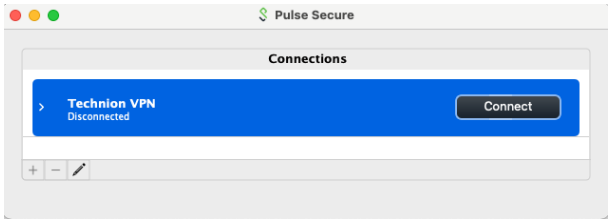
Step 1 – Pre-requirements:

1. Generate a token (link) in order to connect the Technion network from home. Note: this stage needs to be done from the Technion.
2. Install PyCharm Professional (link) on your laptop. Use your @campus account for free access.

Step 2 – Set up your remote project

1. Make sure you received the following:
    a. IP address of your machine, i.e. 132.68.54.39
    b. Username for the machine
    c. Password for the machine

2. Connect the Technion VPN via PulseSecure:



3. Open a terminal in your laptop. Make sure you have access to your machine by:

    ping <ip>

    You should expect a similar output:

```
(base) eladhirsch@Elads-MacBook-Pro ~ % ping 132.68.54.39
PING 132.68.54.39 (132.68.54.39): 56 data bytes
64 bytes from 132.68.54.39: icmp_seq=0 ttl=59 time=22.811 ms
64 bytes from 132.68.54.39: icmp_seq=1 ttl=59 time=23.698 ms
64 bytes from 132.68.54.39: icmp_seq=2 ttl=59 time=17.283 ms
64 bytes from 132.68.54.39: icmp_seq=3 ttl=59 time=22.958 ms
64 bytes from 132.68.54.39: icmp_seq=4 ttl=59 time=17.291 ms
```

Then press Ctrl+C

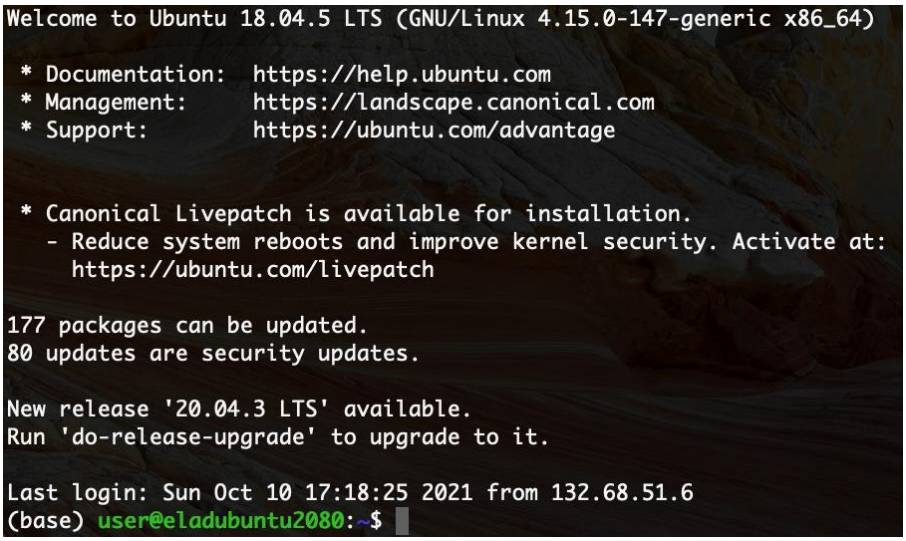
4. In the terminal, initiate a ssh session:

   ssh <username>@<ip>

   enter your password.

   For example:

   After successfully entering your password, you can run commands on your machine:



5. Make sure you have anaconda installed

   conda info

   if not installed, you can install it (link).

   wget https://repo.anaconda.com/archive/Anaconda3-2021.05-Linux-x86_64.sh
   bash ./Anaconda3-2021.05-Linux-x86_64.sh

   Reset the session

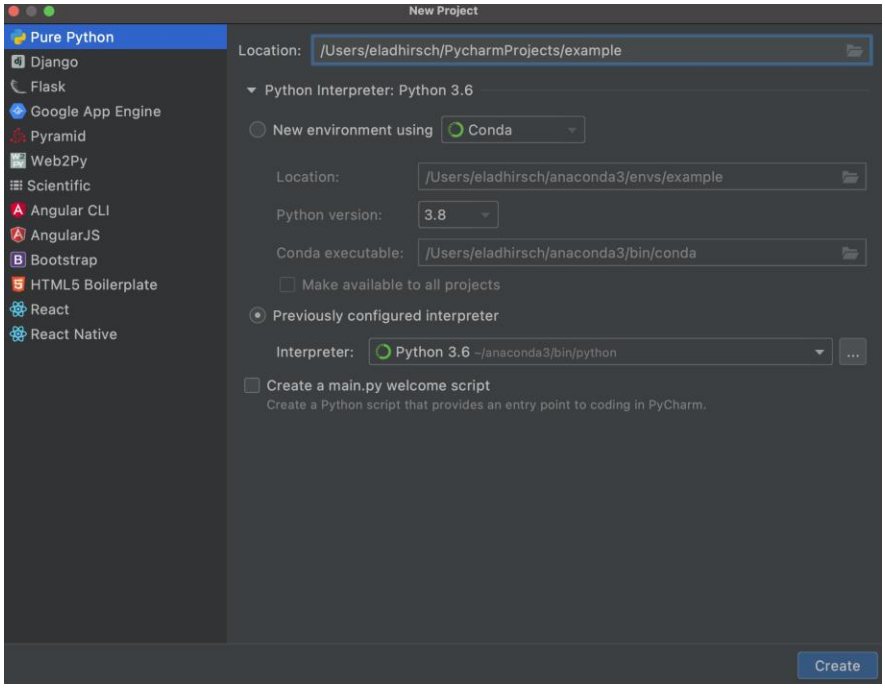
6. Create a new conda environment for your project

   conda create --name <env-name> python=3.7

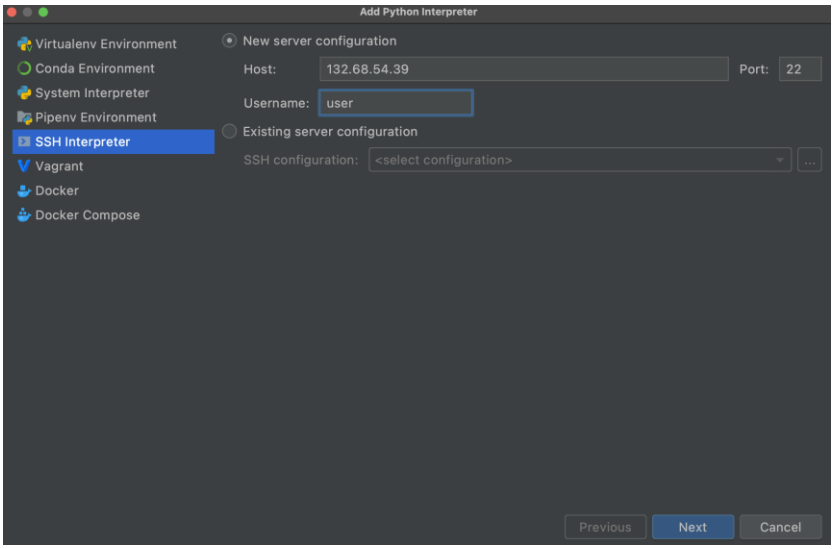
   If you already have the project requirements, you can install them (but it can also wait with that):
   conda activate <env-name>
   conda install …
7. Terminate the ssh session with Ctrl+D

8. Open PyCharm Professional

9. Create a new project (or choose a cloned project)
   a. Choose the project location (empty or existing folder)
   b. Under “Python Interpreter”, choose “Previously configured interpreter”
   c. Press the “…” on the right



   Then in the next window choose “SSH interpreter” and fill in the server’s address and username:



Continue with “Next” and enter your password.

In the next window, set the following:

1. Interpreter:
   Open the directory browser (click on the diretory icon)
   choose: /home/<user>/anaconda/envs/<env_name>/bin/python
   (might not be the exact names but should be very similar)

2. Mapping:
   In Sync folders: “Local Path” - choose your project path in your local computer, “Remote Path” - choose the project path in the server (you may create a new folder). Make sure “automatically upload project files to the server” is ticked.

Note: The IP may change if the remote host is shut down for a while. In that case, you shall have to re-configure it.

Step 3 – Edit your code

The files you create, or edit are automatically uploaded to the server.
Files from the server are not uploaded to your computer, though.

It is recommended to work on the source files locally – these are uploaded to the server. Datasets, models or other massive files – download directly to the remote machine – by visiting the lab or shell commands through ssh (see next section).

Now you may need to install new requirements to your conda environment. Open a ssh session and perform:

conda activate <env-name>
conda install …

Note that for installing PyTorch, you'll need to find the cuda version installed on the server.
In the terminal check the cuda version by nvcc --version.
Then, go to https://pytorch.org/get-started/locally/
And choose Stable, Linux, Conda, Python and your cuda version. Run the command.

Step 4 – Run your code

You can run the code either from PyCharm or the terminal.
Debugging or short runs are usually easier through PyCharm, but longer runs require tmux sessions from the terminal.

Pycharm:

1. Run -> Edit Configurations
2. Create the desired configuration (then apply)
3. Run / Debug

Terminal:

1. Tools -> Start SSH Session
2. Select your remote host
3. In the terminal window you can run commands.
4. For long runs, run 'tmux'
    a. First, if tmux not installed – run sudo apt install tmux
    b. In the new session window, remember to activate your conda environment.
    c. Run your code
    d. Press Ctrl+B, then 'd' to leave the session open in the backgroung, or Ctrl+D to terminate the tmux session.
    e. When you want to return to the tmux session, run 'tmux attach -t 0' (0 is the id of the tmux session, it can also be another id according to your active sessions and it can also be a name, link)

Note: If your code does not run due to missing packages, you should install them with "conda install …" from the terminal.