I understand you have a Conda environment named "myenv". Let's use this information to ensure we're running the script with the correct Python interpreter. Here's what we'll do:

1. First, let's verify the exact path of the Python interpreter in your "myenv" environment:
   - SSH into your server: `ssh ubuntu@165.246.21.118 -p 22`
   - Activate your environment: `conda activate myenv`
   - Find the Python path: `which python`
   - Note down the path it shows (it should be something like `/home/ubuntu/anaconda3/envs/myenv/bin/python`)
2. Now, let's update your PyCharm Run Configuration:
   - Go to Run > Edit Configurations
   - Set the following:
     - Script path: `/home/ubuntu/test.py`
     - Python interpreter: Make sure it's set to your remote "myenv" interpreter
     - Working directory: `/home/ubuntu`
3. In the same Run Configuration window, click on "Environment" and ensure:
   - "Add content roots to PYTHONPATH" is checked
   - "Add source roots to PYTHONPATH" is checked
4. For Path mappings:
   - Click the folder icon next to "Path mappings"
   - Add a mapping: Local path: [Your local project directory] Remote path: `/home/ubuntu`
5. Apply these changes and click OK
6. Now, let's create a debug script on the remote server:
   - SSH into your server
   - Create a new file: `nano /home/ubuntu/debug.py`
   - Add the following content:

</> **Debug Script**
Click to open code

- Save and exit (Ctrl+X, then Y, then Enter)

7. Update your PyCharm Run Configuration to use `/home/ubuntu/debug.py` as the script path
8. Run this debug script from PyCharm and share the complete output

This debug script will provide us with crucial information about the Python environment, working directory, and the presence of your test script. With this information, we should be able to identify why PyCharm is having trouble running your script and resolve the issue.

A

```
(myenv) ubuntu@ubuntu:~$ which python
/usr/bin/python
```

Thank you for providing that information. It appears that even though you've activated the "myenv" Conda environment, the system is still pointing to the default Python installation. This could be the root cause of our issues. Let's rectify this and set up PyCharm correctly:

1. First, let's properly locate your Conda environment:
   - In your SSH session, run these commands:

     Copy

     ```
     conda activate myenv
     conda info --envs
     ```

   - This will show all your Conda environments. Find the path for "myenv".
2. Now, let's find the correct Python executable:

   Copy

   ```
   echo $CONDA_PREFIX/bin/python
   ```

   This should give you the correct path to the Python in your "myenv" environment.
3. Verify this Python is from your Conda environment:

   Copy

   ```
   $CONDA_PREFIX/bin/python --version
   ```

4. Now, in PyCharm, update your Run Configuration:
   - Go to Run > Edit Configurations
   - Set the following:
     - Script path: `/home/ubuntu/test.py`
     - Python interpreter: Click on the dropdown, then "Show All"
     - In the "Interpreters" dialog, select your remote interpreter for "myenv"
     - If it's not listed, add a new SSH interpreter with the path you found in step 2
     - Working directory: `/home/ubuntu`
5. In the same Run Configuration window:
   - Ensure "Add content roots to PYTHONPATH" is checked
   - Ensure "Add source roots to PYTHONPATH" is checked

6. For Path mappings:
    - Click the folder icon next to "Path mappings"
    - Add a mapping: Local path: [Your local project directory] Remote path: `/home/ubuntu`
7. Apply these changes and click OK
8. Now, let's create an updated debug script on the remote server:
    - SSH into your server
    - Create a new file: `nano /home/ubuntu/debug.py`
    - Add the following content:

</>

Updated Debug Script
Click to open code

- Save and exit (Ctrl+X, then Y, then Enter)

9. Update your PyCharm Run Configuration to use `/home/ubuntu/debug.py` as the script path
10. Run this debug script from PyCharm and share the complete output

This updated debug script will give us a comprehensive view of the Python environment PyCharm is using, which will help us ensure it's correctly using your Conda environment and accessing the right files. Once we have this information, we can make any necessary adjustments to get your script running properly.

⚠ Claude does not have the ability to run the code it generates yet.

A Based on this Give the instruction step by step

```
(base) ubuntu@ubuntu:~$ conda activate myenv
(myenv) ubuntu@ubuntu:~$ which python
/usr/bin/python
(myenv) ubuntu@ubuntu:~$ conda info --envs
# conda environments:
#
                         /home/ubuntu/PROF_KIM/amber22_src/build/CMakeFiles/miniconda/install
base                     /home/ubuntu/anaconda3
FemtoDet                 /home/ubuntu/anaconda3/envs/FemtoDet
YOLOU                    /home/ubuntu/anaconda3/envs/YOLOU
YOLOX                    /home/ubuntu/anaconda3/envs/YOLOX
YOLOv6                   /home/ubuntu/anaconda3/envs/YOLOv6
ddrm                     /home/ubuntu/anaconda3/envs/ddrm
myenv                 *  /home/ubuntu/anaconda3/envs/myenv
nanodet                  /home/ubuntu/anaconda3/envs/nanodet
newenv                   /home/ubuntu/anaconda3/envs/newenv
py36                     /home/ubuntu/anaconda3/envs/py36
py37                     /home/ubuntu/anaconda3/envs/py37
py38                     /home/ubuntu/anaconda3/envs/py38
torch                    /home/ubuntu/anaconda3/envs/torch
torch_1131_cu117         /home/ubuntu/anaconda3/envs/torch_1131_cu117
torch_181_cu111          /home/ubuntu/anaconda3/envs/torch_181_cu111
yolov7                   /home/ubuntu/anaconda3/envs/yolov7
                         /opt/intel/oneapi/intelpython/latest
                         /opt/intel/oneapi/intelpython/latest/envs/2023.1.0
                         /opt/intel/oneapi/intelpython/latest/envs/tensorflow
                         /opt/intel/oneapi/modin/latest
                         /opt/intel/oneapi/tensorflow/latest
(myenv) ubuntu@ubuntu:~$ echo $CONDA_PREFIX/bin/python
/home/ubuntu/anaconda3/envs/myenv/bin/python
(myenv) ubuntu@ubuntu:~$ $CONDA_PREFIX/bin/python --version
Python 3.8.19
(myenv) ubuntu@ubuntu:~$
```

Thank you for providing this detailed information. Based on this, I'll give you step-by-step instructions to correctly set up PyCharm to use your Conda environment:

1. Confirm the correct Python interpreter path: The correct path for your "myenv" Python interpreter is: `/home/ubuntu/anaconda3/envs/myenv/bin/python`
2. Open PyCharm on your local machine.
3. Set up the remote interpreter:

  - Go to File > Settings (on Windows/Linux) or PyCharm > Preferences (on macOS)
  - Navigate to Project: [Your Project Name] > Python Interpreter
  - Click the gear icon next to the interpreter dropdown and select "Add"
  - Choose "SSH Interpreter" from the left menu

4. Configure the SSH connection:
   - Host: 165.246.21.118
   - Port: 22
   - Username: ubuntu
   - Authentication type: Password (or use Key-based authentication if you have it set up)
   - Click "Next"
5. Set up the remote interpreter path:
   - In the "Interpreter" field, enter: `/home/ubuntu/anaconda3/envs/myenv/bin/python`
   - Click "Finish"
6. Configure deployment (to sync your local files with the remote server):
   - Go to Tools > Deployment > Configuration
   - Click the "+" button to add a new server
   - Choose SFTP as the connection type
   - Fill in the same SSH details as before (host, port, username)
   - In the "Root path" field, specify: `/home/ubuntu`
   - Go to the "Mappings" tab and set up the mapping: Local path: [Your local project directory] Deployment path: /
7. Enable automatic upload:
   - Go to Tools > Deployment > Options
   - Check "Upload changed files automatically to the default server"
   - Choose when you want files to be uploaded (e.g., "On explicit save action")
8. Create a test Python script in your project named `test.py` with a simple content:

   python Copy

   ```python
   print("Hello from myenv!")
   ```

9. Set up a Run Configuration:
   - Go to Run > Edit Configurations
   - Click the "+" and choose "Python"

   - Set the following:
     - Name: Test myenv
     - Script path: `/home/ubuntu/test.py`
     - Python interpreter: Select the remote interpreter you just configured
     - Working directory: `/home/ubuntu`

10. Apply the changes and click OK
11. Make sure your `test.py` file is uploaded to the server:
    - Right-click on `test.py` in the Project view
    - Select Deployment > Upload to [Your Server Name]
12. Run your script by clicking the green "Run" button or using Shift+F10

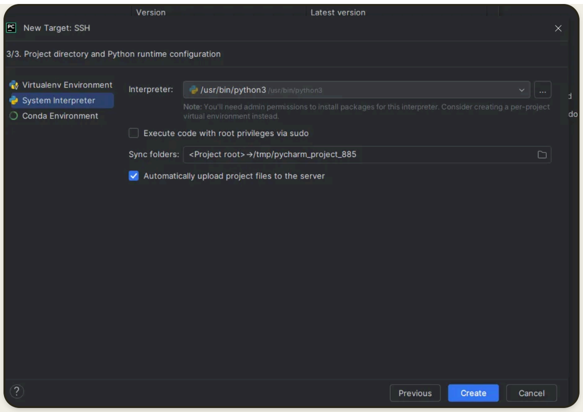
If you encounter any issues, create a debug script:

Debug Script for myenv
Click to open code

Save this as `debug.py`, upload it to the server, and run it using the same Run Configuration (just change the script path to `debug.py`).

If you still encounter issues, please run this debug script and share the output. This will help identify any remaining problems with the setup.

⚠ Claude does not have the ability to run the code it generates yet.



New Target: SSH

3/3. Project directory and Python runtime configuration

Interpreter: /usr/bin/python3 /usr/bin/python3

Note: You'll need admin permissions to install packages for this interpreter. Consider creating a per-project virtual environment instead.

☐ Execute code with root privileges via sudo

Sync folders: <Project root>→/tmp/pycharm_project_885

☑ Automatically upload project files to the server

A From this i should select conda environment or system interpretor?

Based on the information you provided earlier about your Conda environment "myenv", you should select "Conda Environment" instead of "System Interpreter" or "Virtualenv Environment".

Here's what you should do:

1. In this window, click on "Conda Environment" in the left panel.
2. After selecting "Conda Environment", you should see new options:

3. For the "Conda executable" field, enter: `/home/ubuntu/anaconda3/bin/conda`
4. For "Use existing environment", select "myenv" from the dropdown if it's available. If not, you may need to enter the path manually: `/home/ubuntu/anaconda3/envs/myenv`
5. Make sure "Automatically upload project files to the server" is checked.
6. For "Sync folders", you can leave it as is for now (`<Project root>->/tmp/pycharm_project_885`), though we may need to change this later to a more permanent location.
7. Do not check "Execute code with root privileges via sudo" unless you specifically need this for your project.
8. Click "Create" to finish setting up the interpreter.

This setup will ensure that PyCharm uses your Conda environment "myenv" when running Python scripts on the remote server. It will use the correct Python interpreter (`/home/ubuntu/anaconda3/envs/myenv/bin/python`) and have access to all the packages installed in that environment.

After setting this up, you should be able to run your scripts using the correct Conda environment. If you encounter any issues after this setup, please let me know, and we can troubleshoot further.