Creating lightweight environment in the python environment selected below reduced ArcGIS Pro package that originally had it all, but when working with chatgbt on bathymetry notebook environment and disk space issues with EC2 you stripped a lot of this ArcGIS Pro environment away unintentionally, when trying to complete other notebook work. Still connected to ArcGIS Pro just not fully integrated. Thankfully you are in a development testing work environment, but if in real production you would need to reinstall this again to have fully ArcGIS Pro capabilities.

Moral of the story although things chatgbt can help you do are great, if you don’t fully understand the implications of these actions and what you are asking it to do, you can create unintended consequences like above.

A screenshot of a computer

AI-generated content may be incorrect.

you said

when we did the lightweigth environment in the bathymetry notebook was that just affecting the notebook or would it have affected the whole anaconda navigator system in terms of the environment we were working in?

**ChatGPT said:**

Excellent question — here’s the **clear distinction**:

**🧠 The “lightweight environment” setup you did affected only this:**

🟩 **Environment:** (arcgispro-py3-clone1-clean-gis-env)  
📁 **Impact scope:** *Just this specific environment — not Anaconda Navigator itself, nor the base environment.*

**✅ So to answer directly:**

**Did it affect the whole Anaconda Navigator system?**  
**👉 No.** It only modified the *specific* conda environment you were working in:  
arcgispro-py3-clone1-clean-gis-env

Anaconda Navigator just acts as a GUI **manager** for multiple environments. Each environment is **fully isolated**, so:

* You could have base, arcgispro-py3-clone, and arcgispro-py3-clone1-clean-gis-env all with **different libraries**, versions, and configurations.
* Working in a lightweight environment only affects that one — you haven’t broken, removed, or downgraded anything globally.

**🧾 What You Did in That Lightweight Setup:**

* Installed just arcgis, pytorch, torchvision, and other essentials (but not the full deep-learning-essentials).
* Ran the notebook inside that environment.
* Encountered limited library access (like fastai, MaskRCNN) — but that was expected due to the lean install.

**🧰 If Needed:**

You can still:

* Switch to another environment in Navigator (like base or arcgispro-py3-clone)
* Create a new environment with full tools later
* Delete or archive this one anytime

Let me know if you want to make a **duplicate environment**, or revert to a fuller one in future.

**You said:**

in the previous screenshot and here it is again, the arcgis pro packages are missing, this is what I was concerned about regarding my previous message you just commented on

**ChatGPT said:**

Ah — thank you for clarifying. You're absolutely right to be cautious.

**⚠️ You are observing that ArcGIS Pro-specific packages like arcgispro, arcgispro-py, or extensions (like deeplearning, arcgis.learn) are not visible in the Navigator UI.**

Let's address this directly:

**✅ What’s Actually Happening:**

You're inside this environment:

CopyEdit

arcgispro-py3-clone1-clean-gis-env

And yes — **this is the custom lightweight environment** we created during the notebook work.

This environment **does not inherit** the full set of packages that are included in the default arcgispro-py3 environment used by ArcGIS Pro itself.

**🔍 Why ArcGIS Pro Packages Are Missing:**

1. **ArcGIS Pro uses a separate pre-packaged environment** (arcgispro-py3) installed alongside ArcGIS Pro.
2. Your lightweight clone (-clean-gis-env) only includes what you added manually via conda install, such as:
   * arcgis
   * torch
   * torchvision
   * Maybe arcpy if copied from arcgispro-py3

👉 So, **you are not seeing ArcGIS Pro's full environment because this clone doesn't include all of those packages**.

**🧠 So your previous question is now clear:**

Did our lightweight setup affect the whole Anaconda Navigator?

**No**, it did not affect ArcGIS Pro’s environment or the base environment.

But:

Did we switch to a stripped-down environment that lacks ArcGIS Pro’s full package set?

**✅ Yes, exactly.** That’s the behavior you’re seeing.

**🩺 How to Confirm or Recover ArcGIS Pro Environment**

If you want to:

* **Get back ArcGIS Pro’s full environment** → switch back to arcgispro-py3 in Navigator (not the -clone1-clean version).
* **Re-clone the full ArcGIS Pro environment** → do this from the command line:

bash

CopyEdit

conda create --name arcgispro-py3-full --clone arcgispro-py3

That way you can:

* Use full capabilities of ArcGIS Pro
* Still have your lightweight clone around

**✅ TL;DR Action Checklist**

| **Goal** | **Action** |
| --- | --- |
| Use full ArcGIS Pro packages again | Switch to arcgispro-py3 environment in Navigator |
| Keep lightweight notebook work | Use arcgispro-py3-clone1-clean-gis-env as-is |
| Want a fresh copy of full env | conda create --name arcgispro-py3-full --clone arcgispro-py3 |
| Worried about space? | Keep both, but only activate one at a time |

Would you like me to script the creation of a **dual-environment setup**, or show how to sync just specific packages like arcpy, arcgispro, arcgis.learn from one environment to another?

**You said:**

So we are still connected to arcgis pro within the stripped down environment we created for arcgispro-py3-clone1-clean-gis-env , we just do not have full access to everything arcgis pro offers? (in this particular environment anymore, we did before we created the lightweight environment but the lightweight environment thereafter which was created stripped some of these arcgis capabilities away?

**ChatGPT said:**

✅ **Yes — exactly.** You've got the core idea completely correct.

Let me break it down and confirm clearly:

**🧭 You *are still connected* to ArcGIS Pro in your lightweight environment...**

You’re using the **arcgispro-py3-clone1-clean-gis-env** environment.

In that environment, you still **have access to core tools like:**

* arcgis (ArcGIS Python API)
* Possibly arcpy (if it was copied in manually or inherited)
* Anything else *you specifically installed*

**⚠️ ...But you don’t have access to all of ArcGIS Pro’s native features anymore**

This lightweight environment is **not a full clone of** arcgispro-py3, the environment that ArcGIS Pro builds and relies on.

That means:

| **ArcGIS Capability** | **Available in your current env?** |
| --- | --- |
| ✅ arcgis API | Yes (you installed it manually) |
| ✅ torch, torchvision | Yes (you manually added it) |
| ⚠️ MaskRCNN, prepare\_data, deep learning extensions | ❌ Not included (from deep-learning-essentials) |
| ⚠️ ArcGIS Pro UI integration (e.g., Geoprocessing Tools calling Python) | ❌ Not guaranteed unless you're using the full arcgispro-py3 |
| ⚠️ ESRI-maintained integrations (e.g., full arcgis.learn) | ❌ Not fully present in the stripped-down env |

**🧠 Summary of What Happened:**

1. Originally, your system had:

sql

CopyEdit

arcgispro-py3 ← full ArcGIS Pro environment (all Esri deep learning tools)

1. You created:

vbnet

CopyEdit

arcgispro-py3-clone1-clean-gis-env ← custom, stripped-down version

* + This was needed due to **low disk space**
  + You installed only what you needed for the bathymetry notebook (like arcgis, torch, torchvision)
  + It **did not include** deep learning libraries, FastAI, or full ArcGIS Pro suite

**✅ You’re still connected, just not fully integrated**

You can still do:

* Notebook work using arcgis
* Some arcpy scripting
* GIS automation and API calls

But you **won’t have access to**:

* Tools like MaskRCNN, prepare\_data, arcgis.learn advanced models
* Esri deep learning model training workflows
* Full geoprocessing integration with ArcGIS Pro

**💡 What You Can Do Moving Forward:**

If you want **full capability**, either:

1. **Switch back to arcgispro-py3 environment**  
   From ArcGIS Pro or Navigator.
2. **Re-clone the full environment (if space allows)**:

bash

CopyEdit

conda create --name arcgispro-py3-full --clone arcgispro-py3

1. **Install missing packages** into your current environment *if space allows*:

bash

CopyEdit

conda install -c esri deep-learning-essentials

Let me know and I’ll help automate whichever route you prefer next.

Top of Form

Tools

Bottom of Form