

Step 1) Install dependencies

WORKFLOW OPTION A (recommended):

create clean environment"

```
python3 -m venv nlse_env
source nlse_env/bin/activate # Windows: nlse_env\Scripts\activate
pip install numpy scipy matplotlib scikit-learn tqdm
```

Step 2) Run the canonical (preload) scripts IN ORDER

From repo root:


```
python preload_scripts/2p_preload_Masterscript_analysis.py
python preload_scripts/3p_preload_1B_fig2_generator.py
python preload_scripts/6p_preload_etaV2paper1B_viz_final.py
python preload_scripts/7p_preload_G_T_derivatives.py
python preload_scripts/8p_preload_statistical_analysis.py
```

WORKFLOW OPTION B (Advanced):

Full regeneration pipeline (SLOW)

This reproduces every experiment from scratch.

Use this ONLY if you want to regenerate all .npz data.

 Important: Scripts 1 and 5 take hours, depending on your machine.

Step 1) Run the full pipeline in order

From repo root:

```
python scripts/1_Master_script_statistics.py
python scripts/2_Masterscript_analysis.py
python scripts/3_1B_fig2_generator.py
python scripts/5_etaV2paper1B_final.py
python scripts/6_etaV2paper1B_viz_final.py
python scripts/7_G_T_derivatives.py
```

python scripts/8_statistical_analysis.py

What Each Script Does (Reviewer-level clarity)

Script	Purpose	Runtime
1_Master_script_statistics.py	Full 15-point η sweep × 10 seeds (NLSE vs LSE accuracy)	Hours
2_Masterscript_analysis.py	Regression / decay-rate	Seconds
3_1B_fig2_generator.py	Generates NLSE–LSE superiority figure	Seconds
5_etaV2paper1B_final.py	T-sweep and g-sweep parameter scans	Hours
6_etaV2paper1B_viz_final.py	g-margin visualization	Seconds
7_G_T_derivatives.py	Derivative / curvature analysis	Seconds
8_statistical_analysis.py	t-tests, Cohen's d, Bonferroni	Seconds

FROM ROOT:

```
python preload_scripts/2p_preload_Masterscript_analysis.py
python preload_scripts/3p_preload_1B_fig2_generator.py
python preload_scripts/6p_preload_etaV2paper1B_viz_final.py
python preload_scripts/7p_preload_G_T_derivatives.py
python preload_scripts/8p_preload_statistical_analysis.py
```