

### Step 0) **Unzip the file**

You will now have a folder named:

`solitonic_reservoir-main/`

Step 1) Open a terminal inside that folder:

Mac:

`cd path/to/solitonic_reservoir-main`

Windows (PowerShell):

- Shift + Right-click → *Open PowerShell window here*
- Or navigate manually into the folder.

WORKFLOW OPTION A (fast, recommended):

Step 2A) create clean environment and install dependencies:

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

Step 3 — Run Canonical (“Preload”) Scripts in Order

From the repository root (e.g., `solitonic_reservoir-main/`):

```
python preload_scripts/2_p_preload_Masterscript_analysis.py
python preload_scripts/3_p_preload_1B_fig2_generator.py
python preload_scripts/6_p_preload_etaV2paper1B_viz_final.py
python preload_scripts/7_p_preload_G_T_derivatives.py
python preload_scripts/8_p_preload_statistical_analysis.py
```

## What This Produces

- Robustness sweep plots
- NLSE vs LSE superiority figures
- g-sweep visualization
- Derivative/curvature analysis
- All statistics reported in the paper

All plots save into:

preload\_figures/


#####

## 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 0) Unzip download

Step 1) Change directory

Step 2B) Install dependencies

Step 3) 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

This regenerates:

- Full  $\eta$  sweep (15 points  $\times$  10 seeds)
- Complete T-sweep
- Complete g-sweep
- All figures and statistics

Outputs save into:

data/  
figures/

Script	Purpose	Runtime
<b>1_Master_script_statistics.py</b>	Full 15-point jitter sweep $\times$ 10 seeds (NLSE vs LSE)	<b>Hours</b>
<b>2_Masterscript_analysis.py</b>	Linear regression / decay-rate analysis	Seconds
<b>3_1B_fig2_generator.py</b>	Generates NLSE–LSE superiority margin figure	Seconds
<b>5_etaV2paper1B_final.py</b>	Time-sweep and g-sweep simulations	<b>Hours</b>
<b>6_etaV2paper1B_viz_final.py</b>	g-margin visualization	Seconds
<b>7_G_T_derivatives.py</b>	Derivative and curvature analysis	Seconds
<b>8_statistical_analysis.py</b>	t-tests, Bonferroni correction, Cohen's d	Seconds

## **Quick Command Summary**

From the repo root:

### **Canonical (Fast) Workflow**

```
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
```

### **Full Regeneration (Slow) Workflow**

```
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
```