Towards Safe and Fearless Lossy Compression of Weather and Climate Model Data

Juniper Tyree[1,5], Sara Faghih-Naini[2,5], Milan Klöwer[3], Tim Reichelt[3,6], Peter Dueben[2,5], Karsten Peters-von Gehlen[4,5], and Heikki J. Järvinen[1,5]

[1] INAR, University of Helsinki, [2] ECMWF, [3] University of Oxford, [4] DKRZ, [5] ESiWACE3, [6] Embed2Scale

CONTINENTS Webinar, 28.05.2025

Funded by the European Union. This work has received funding from the European High Performance Computing Joint Undertaking (JU) under grant agreement No 101093054.

Co-funded by the European Union

Feel free to take pictures of most slides …
… unless the camera is red

Towards exascale weather and climate simulations

Supporting the community of weather and climate modelling in Europe

03

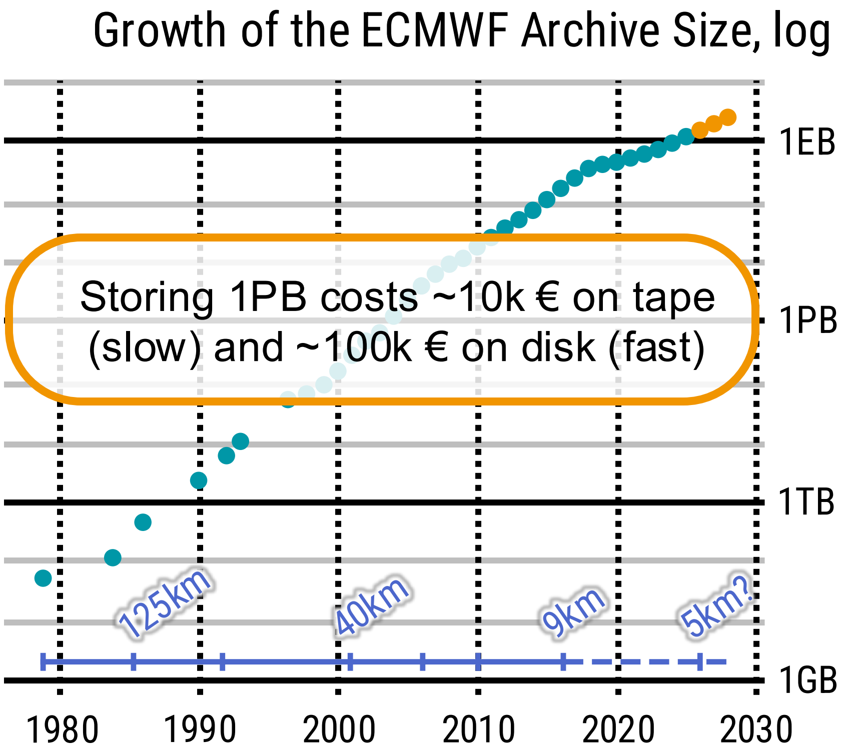
Develop tools to tackle the data challenge of high-resolution models

3.2

Domain-specific data compression

What is Lossy Compression, Why do we need it, and Why is it not used?

Problem: Growth in Weather & Climate Model Data Volumes



European Centre for Medium-Range Weather Forecasts archive size
Dataset provided by Mike Hawkins (ECMWF), Figure adapted from Milan Klöwer (Oxford)

We need **lossy compression**

But losing data accuracy is ***scary***

Problem: Fear of silently losing problem-specific information

Computer scientist: "I can give you great savings by using lossy compression."

Domain scientist: "Nice! … Actually, did we remember to keep the …

- **logarithmic error** norms small for **specific humidity**
- L_2 **error** small for **temperature**
- **budgets** correct when **integrating** over long climate trajectories
- L_∞ **norm** small for **extreme precipitation events**
- **integral in the vertical** correct even though it sums across **several orders of magnitude**
- **delta** at high precision for CO_2

…

And how does lossy data compression change the **enthalpy budget** when cold rain is entering a warm ocean and changing the vertical layering of the vertical column in an ocean model???"

Computer scientist: "Just check the compressed outputs to see if anything broke"

Problem: We need lossy compression but are afraid of using it

Lossy Compression is crucial to handle Data Growth

Unclear safety requirements and error tolerances for lossy compression in weather and climate science

Lack of Trust in Lossy Compression
to uphold the safety requirements
for everyday domain-scientist users

How do we **bridge the gap** between lossy compression and user needs to **build trust**?

Our roadmap towards Safe, Trusted, and Fearless Lossy Compression

The Roadmap to *Fearless* Lossy Compression

Open Science Online Laboratory to Try it out and Convince Yourself

We don't even know what error tolerance we have!
Specify clear safety requirements

What compressor should I use?
Benchmark compressors for safety and performance

But my safety requirements are far stricter than yours!
Create safeguards to use any compressor without fear

Compression should be *Easy* & *Fearless*

“We don’t want to disrupt the science, but disrupt the data challenge”

Heikki Järvinen, ESiWACE3 kick-off, BOG5 session, 01.02.2023

What are our safety requirements? What errors can we tolerate?

Juniper Tyree[1,4], Milan Klöwer[2], Sara Faghih-Naini[3,4], and Peter Dueben[3,4]
[1] INAR, University of Helsinki, [2] University of Oxford, [3] ECMWF, [4] ESiWACE3

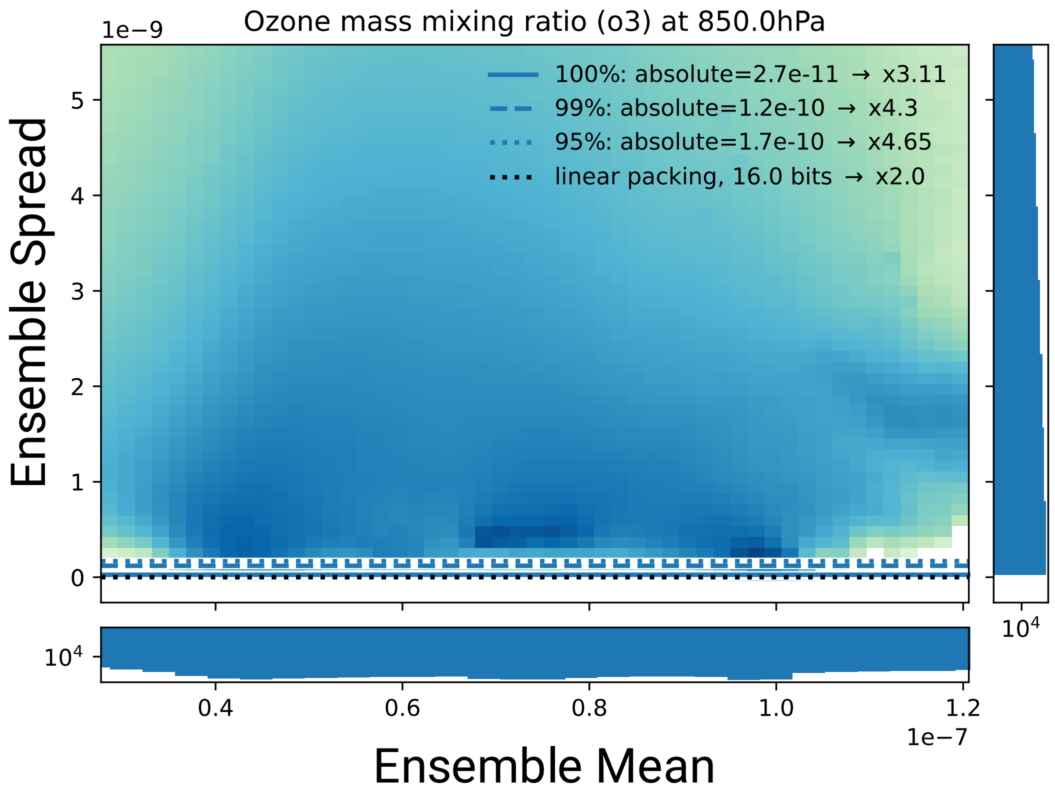
Problem: Unclear which compression errors can be tolerated

Model outputs are stored with excessive precision
Trailing mantissa bits contain false information / noise

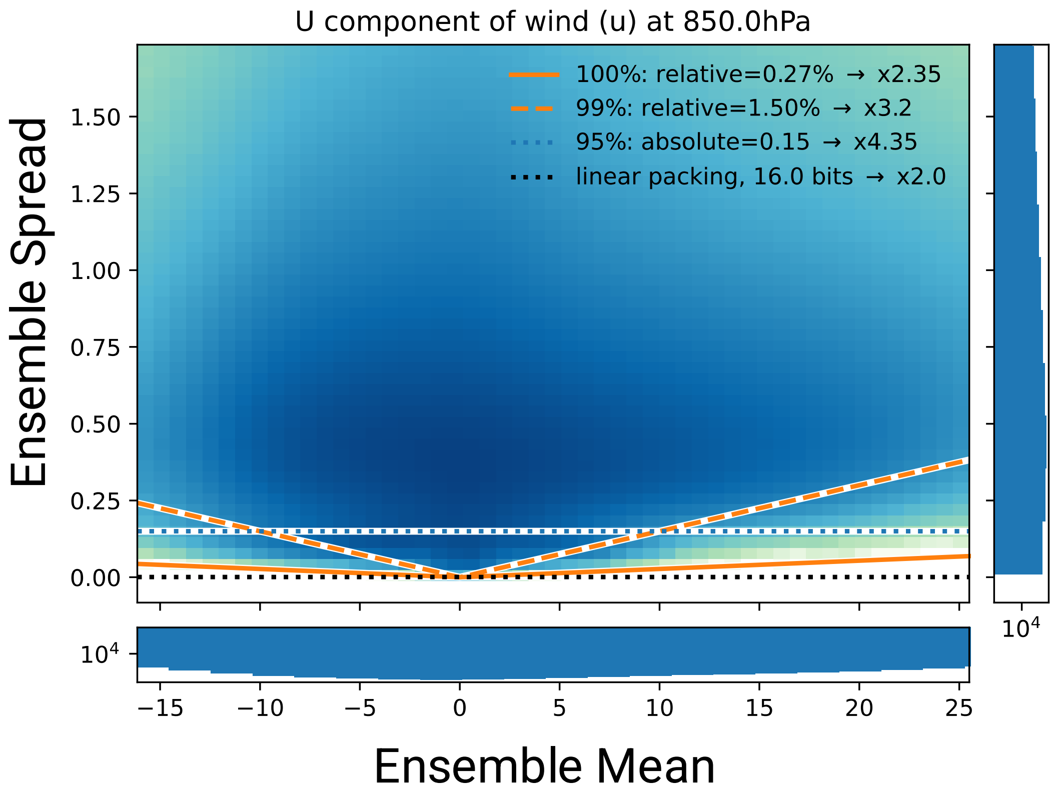
Tolerance varies by variable, resolution, use case
Model output archival must use conservative tolerances

Community needs recommendations and standards
GRIB already uses lossy linear packing, improve on that

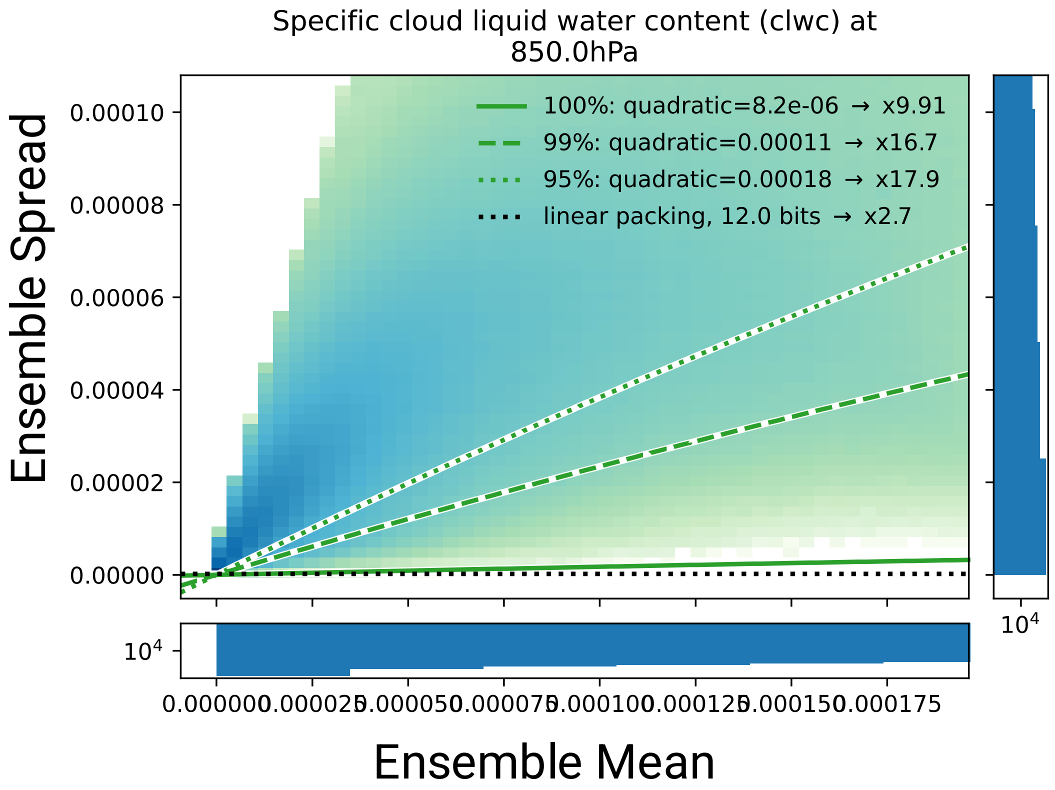
Results: Variables with absolute error bounds, e.g. O3, temp



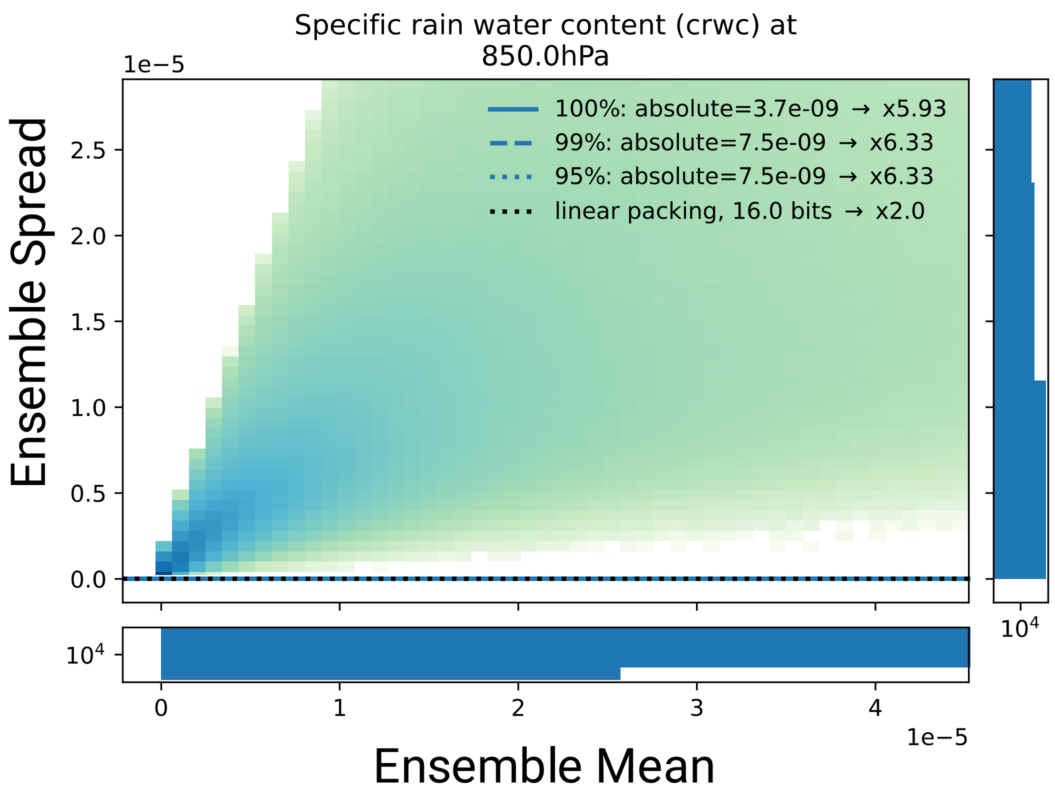
Results: Variables with relative error bounds, e.g. u/v wind



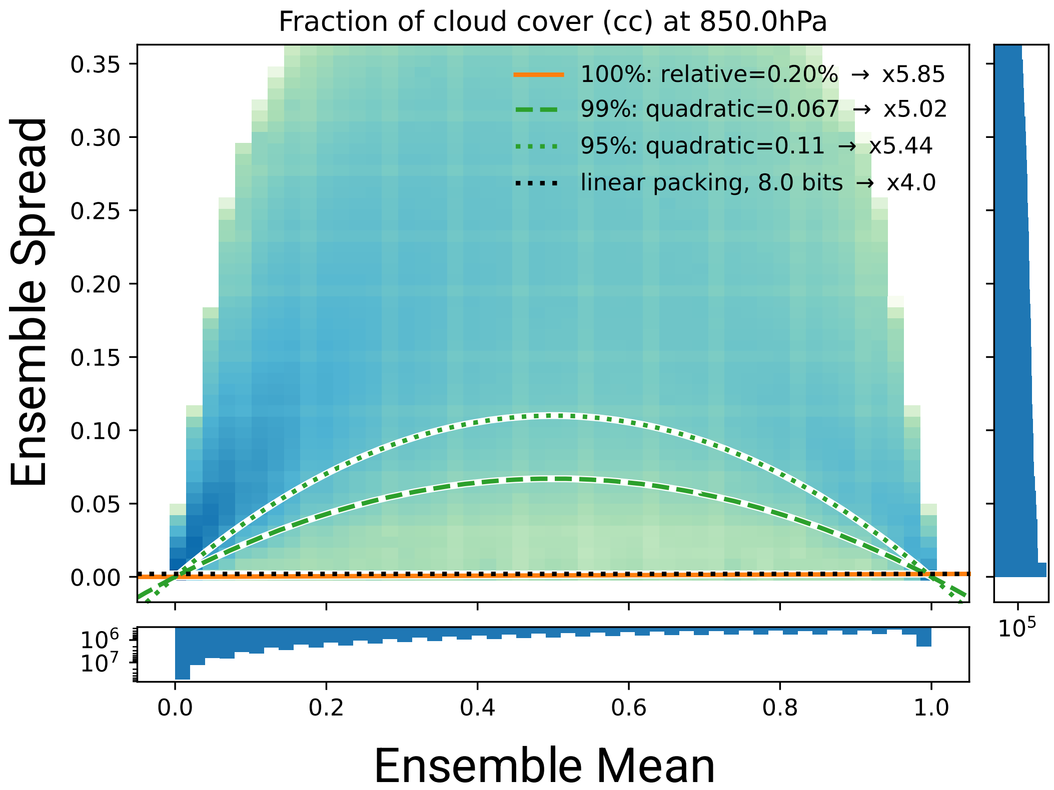
Results: Variables with relative error bounds, e.g. cloud water



Results: Variables with unclear error bounds, e.g. rainwater



Results: Variables with quadratic error bounds, e.g. cloud cover



Outlook: Next steps, publication, and future work

-Q2

Comparing expert and automatic bounds

Experience vs data analysis

Q2-Q3

Initial publication and table of error bounds

Three suggested levels of error bounds per variable

...

Outreach, Feedback, Refinement, Standards

Further feedback and analyses, replace linear packing

How do existing compressors perform? Are they safe to use?

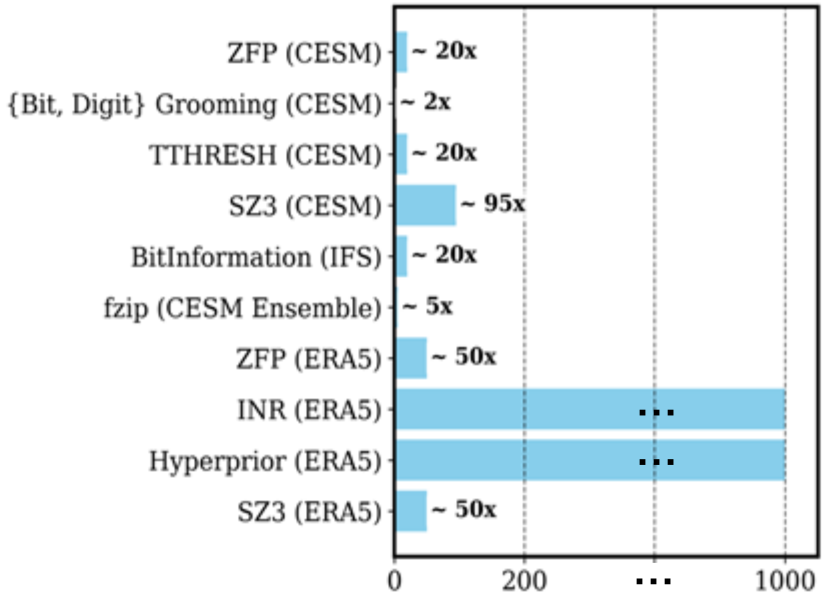
ClimateBenchPress: A Benchmark for Compression of Climate Data

Tim Reichelt[1,7], Juniper Tyree[2,8], Milan Klöwer[1], Peter Dueben[3,8], Bryan Lawrence[4], Dorit Hammerling[5], Allison Baker[6], Sara Faghih-Naini[3,8], and Philip Stier[1,7]

[1] University of Oxford, [2] INAR, University of Helsinki, [3] ECMWF, [4] NCAS, University of Reading,
[5] Colorado School of Mines, [6] NCAR, [7], Embed2Scale, [8] ESiWACE3

Problem: Why do we need a benchmark?

Compression ratios reported in the literature …



are **difficult to compare** because they use different

- Datasets and **resolutions**
- **Variables** and dimensions
- Compression **error bounds**
- Measurement techniques

Aims: What do we want from a good benchmark?

Fair and Comparable Results

Same data, same error bounds (3 levels), same tests

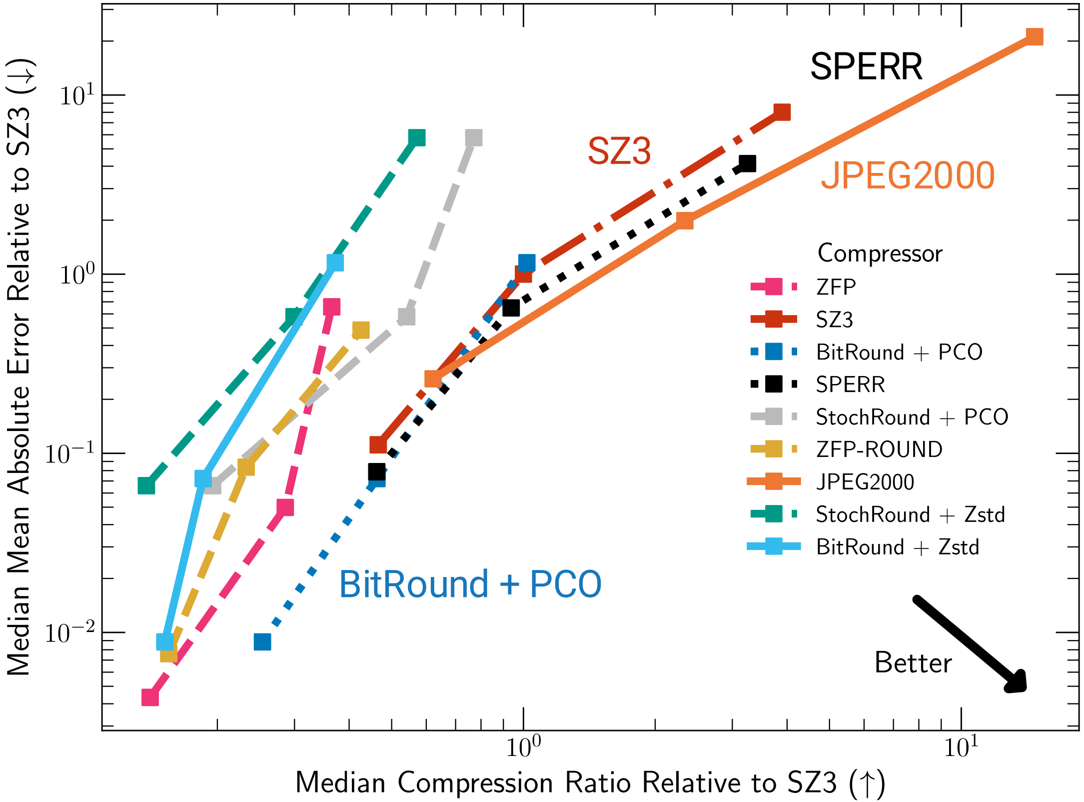
Interpretable Results from Representative Sources

2D & 3D from CMIP6, ERA5, NextGEMS, CAMS, Satellite

Ease of use and expansion

Modular and extensible, several dataset sizes

Results: Preliminary Insights, averaged across several variables



- **SZ3**, **SPERR**, and JPEG2000: highest compression
  - JPEG2000 only has global error bounds (not pointwise)
- **Bit Rounding + PCO** is competitive (but is quite conservative)

Outlook: Next steps, publication, and future work

Q1-Q2

Preparing the small version of the benchmark

Initial subset of data, compressors, and metrics

Q3-Q4

Initial publication and recommendations

Per-variable compressor scoreboard with trade-offs

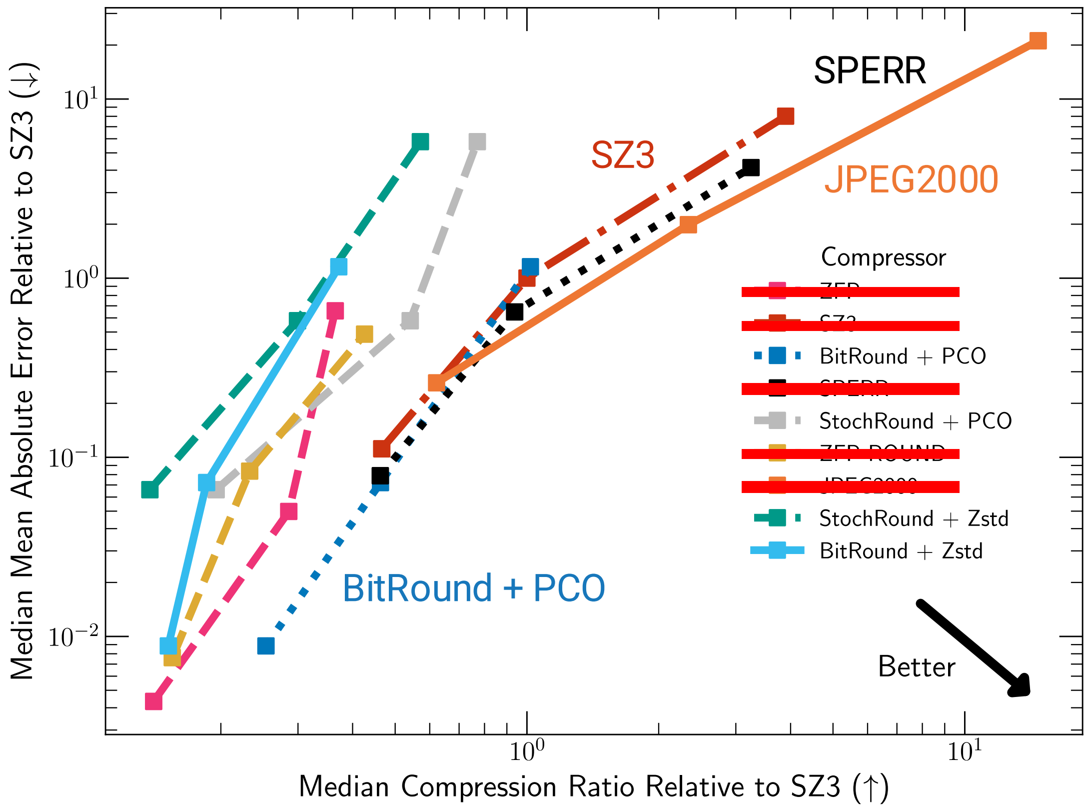
2026-

Expanded benchmark, renew recommendations

Larger datasets, more compressors (ML), more metrics

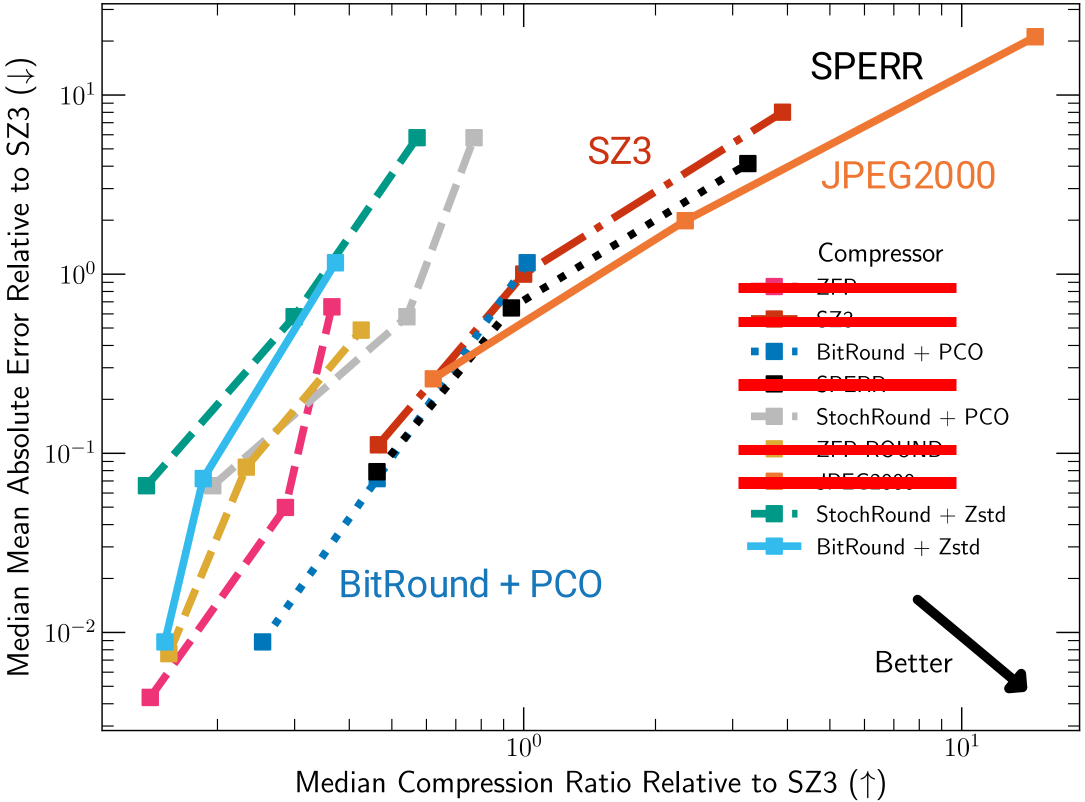
Spooky stories of what can go wrong with lossy compression

Problem: Many popular compressors don't handle edge cases



- **ZFP:** error bias, no NaNs, no Inf's
- **ZFP-ROUND:** no NaNs, no Inf's
- **SZ3:** no NaNs (v3.2)
- **SPERR:** no NaNs or big values, loose bounds
- **JPEG2000:** no NaNs, no Inf's, no pointwise error bounds

Problem: Many popular compressors don't handle edge cases

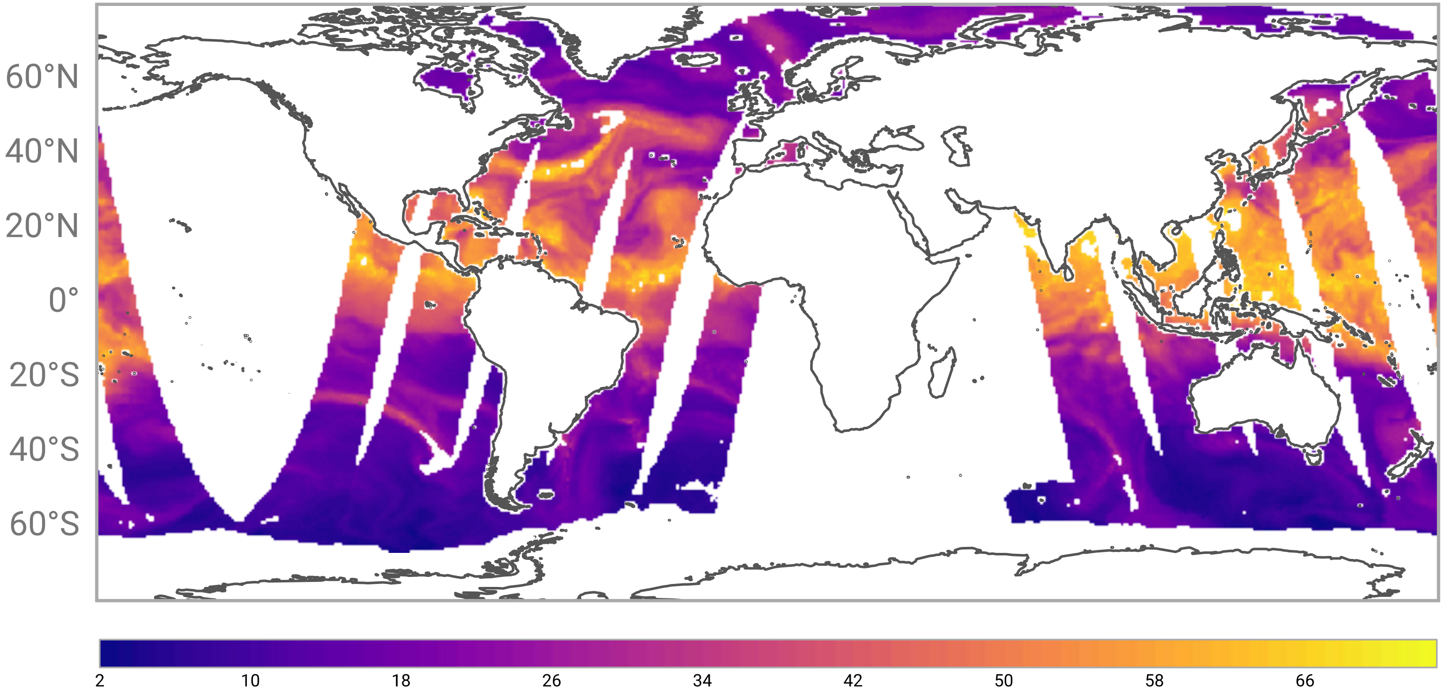


- **ZFP:** error bias, no NaNs, no Inf's
- **ZFP-ROUND:** no NaNs, no Inf's
- **SZ3:** no NaNs (v3.2)
- **SPERR:** no NaNs or big values, loose bounds
- **JPEG2000:** no NaNs, no Inf's, no pointwise error bounds

Explicit failure to compress

Problem: Hallucination in non-ML lossy compressors

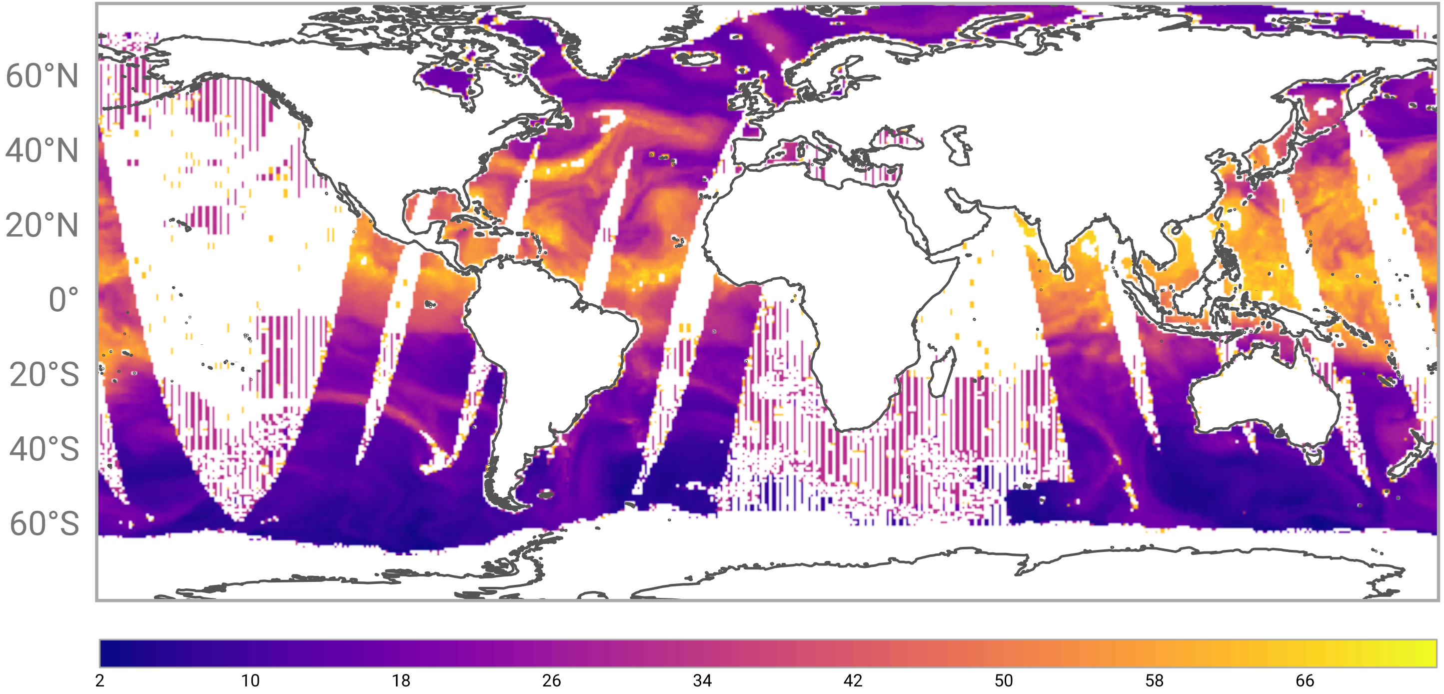
Original data, NaN values on land and from missing coverage



Water Vapor Path [kg · m^{-2}], Satellite data at 06:00 on 02.08.2020

Problem: Hallucination in non-ML lossy compressors

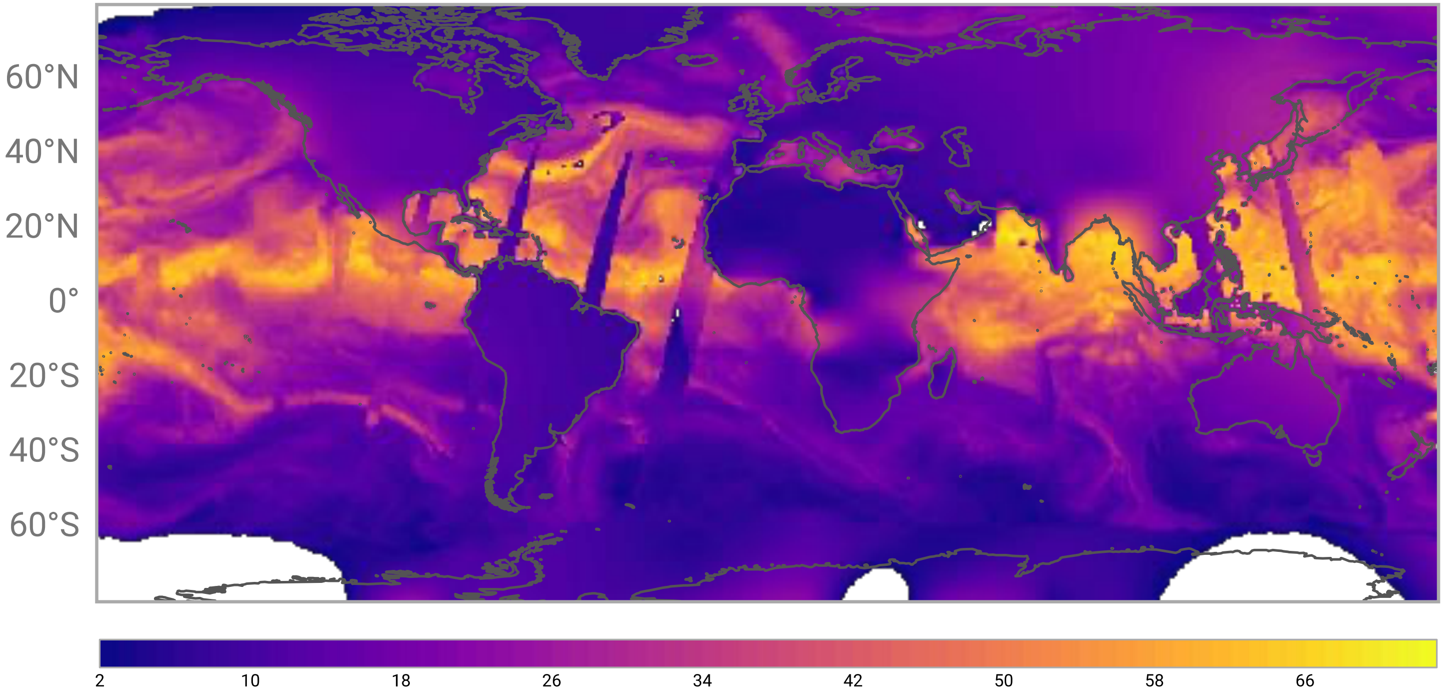
ZFP, absolute error bound $\epsilon_{\text{abs}} = 0.1\ \text{kg} \cdot \text{m}^{-2}$



Water Vapor Path [kg · m^{-2}], Satellite data at 06:00 on 02.08.2020

Problem: Hallucination in non-ML lossy compressors

SZ3, absolute error bound $\epsilon_{\text{abs}} = 0.1\ \text{kg} \cdot \text{m}^{-2}$



Water Vapor Path [kg · m^{-2}], Satellite data at 06:00 on 02.08.2020

Compression Safeguards: Fearless Lossy Compression with declarative Safety

Juniper Tyree[1,4], Milan Klöwer[2], Clément Bouvier[1,4], Daniel Köhler [1], Robert Underwood[3], Heikki J. Järvinen[1,4], et al.*
[1] INAR, University of Helsinki, [2] University of Oxford, [3] Argonne National Laboratory, [4] ESiWACE3, * pending authors

How can the compression safeguards help **You**?

Declare your safety requirements, the safeguards guarantee them

Preserve **special values** (NaN, Inf, 0, missing value, etc.)

Regional pointwise error bounds (abs, rel, noa, ratio)

Preserve properties, e.g. **signs** & **monotonic sequences**

Error bounds over **quantities of interest** (x^2, ∂X, ...)

Logical **combinations** over all of the above

Outlook: Next steps, publication, and future work

-Q3

Compression Safeguards Software

Initial integration with numcodecs, possibly Zarr

Q4-

Initial publication and evaluation

Examples of preserving complex safety requirements

2026-

Expansion and Operationalization

Further safeguards, CDF integration, perf optimizations

Convince Yourself through Outreach and an Open Science Laboratory

Trying something out for yourself is especially important if you have **doubts**

To **convince yourself** that lossy compression is safe, you need to **try it out yourself**

Trying something out should be easy to **engage earlier** and **delay setup costs** after **convincing** users

Solution for the entire community

i.e. not just for our project (lossy compression) but also for your project, documentation, demos, …

The Online Laboratory for Climate Science and Meteorology

What's easier than opening a URL in your browser?

Serverless In-Browser Interactive Computing

jupyterlite

PYODIDE

WA

Code and data stay *local*

Internet icon by Freepik at https://www.flaticon.com/free-icon/internet_4861920

No setup, **no installation**, <1min to start

Reproducible and version-locked
Ensure your examples keep working

Supports many compiled scientific + especially Earth Science Python packages

Extra support for **accessing large datasets**

Ease of use: same code, same results

Example: Software documentation is non-interactive

earthkit-plots.readthedocs.io

earthkit-plots

earthkit plots

stable

Search docs

EXAMPLES

- Examples
- Gallery

DOCUMENTATION

- User guide
- API Reference
- Development

INSTALLATION

- Installation
- License

PROJECTS

- earthkit

Welcome to the earthkit-plots documentation

ESEE Foundation | Maturity Incubating | License Apache 2.0

earthkit-plots leverages the power of the **earthkit** ecosystem to make producing publication-quality scientific graphics as simple and convenient as possible.

It is built on top of popular data science and visualisation tools like **numpy**, **xarray**, **matplotlib** and **cartopy**, but provides a very high-level API enriched with domain-specific knowledge, making it exceptionally easy to use.

Key features include:

⚡ **Concise, high-level API** Generate high-quality visualisations with minimal code.

🧠 **Intelligent formatting** Titles and labels automatically adapt based on common metadata standards.

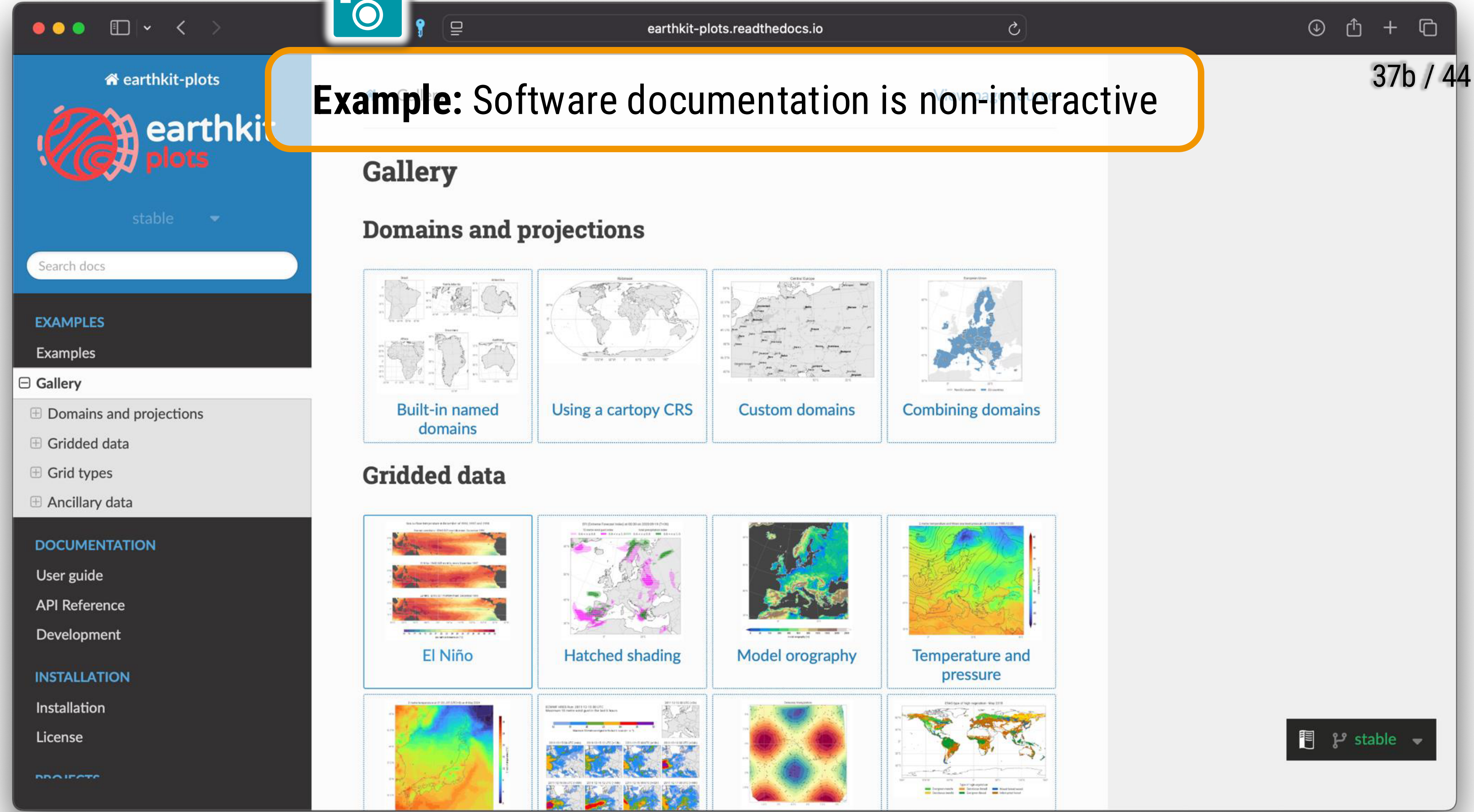
🎨 **Customisable style libraries** Easily swap styles to match your organisation, project, or personal preferences.

🔍 **Automatic data styling** Detects metadata like variables and units to optionally apply appropriate formatting and styling.

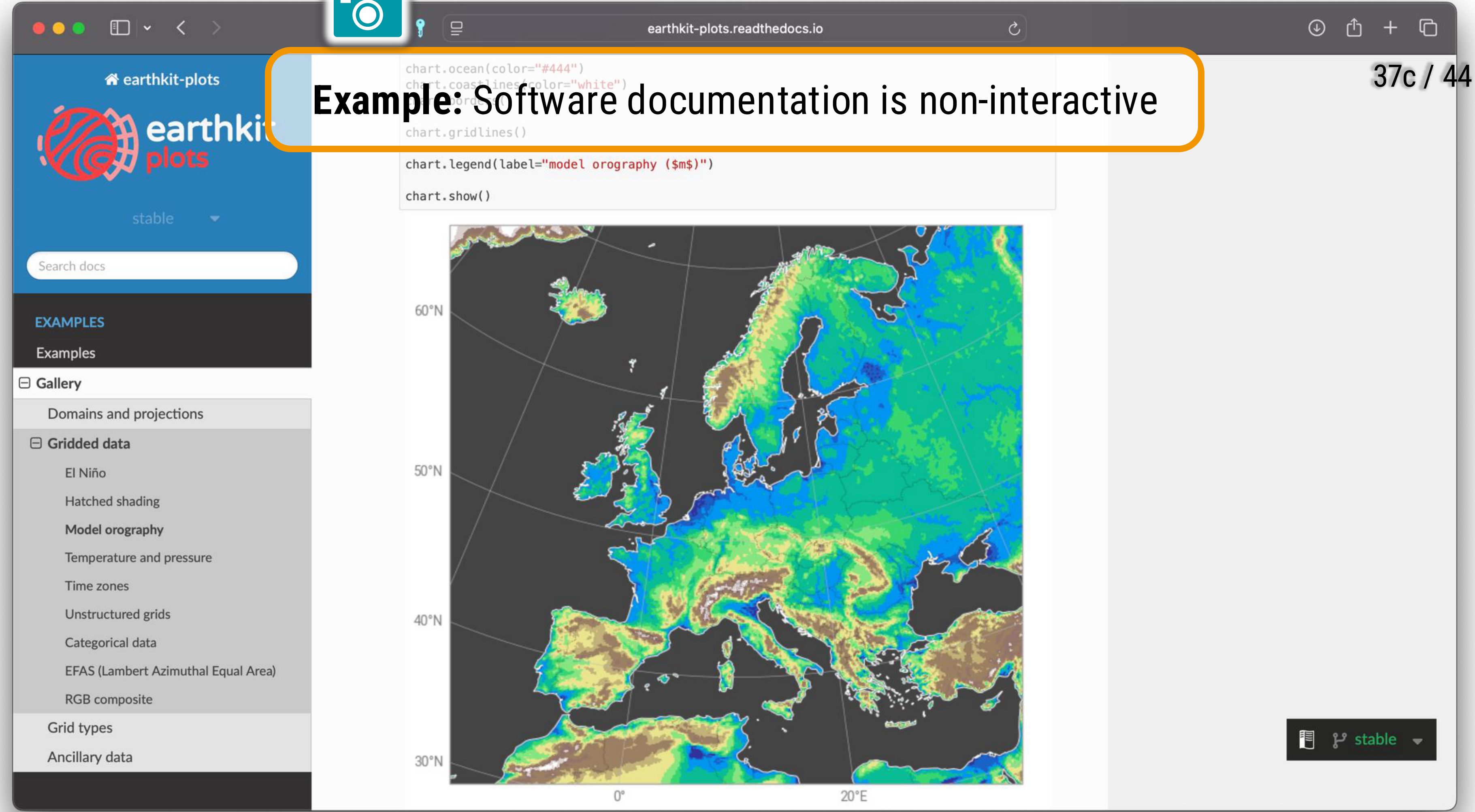
🌍 **Complex grids supported out-of-the-box** Visualise grids like HEALPix and reduced gaussian without any extra legwork.

stable

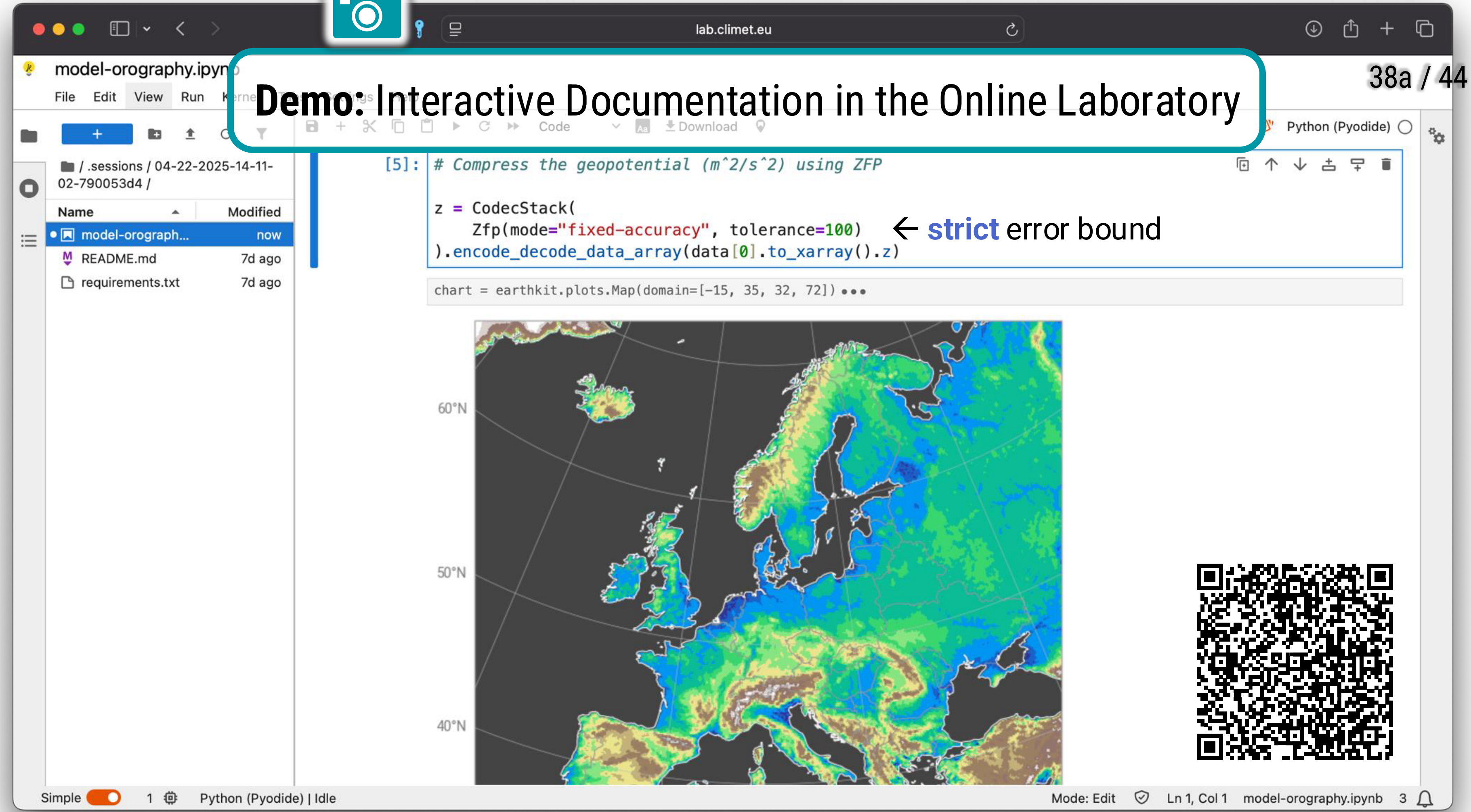
Example: Software documentation is non-interactive



Example: Software documentation is non-interactive



Demo: Interactive Documentation in the Online Laboratory



← **strict** error bound

https://lab.climet.eu/v0.3.0/raw/github-tag/ecmwf/earthkit-plots/0.3.1/docs/examples/gallery/gridded-data/model-orography.ipynb

Demo: Interactive Documentation in the Online Laboratory

lab.climet.eu

model-orography.ipynb

File Edit View Run Kernel Tabs Settings Help

/ .sessions / 04-22-2025-14-11-02-790053d4 /

Name	Modified
model-orograph...	now
README.md	7d ago
requirements.txt	7d ago

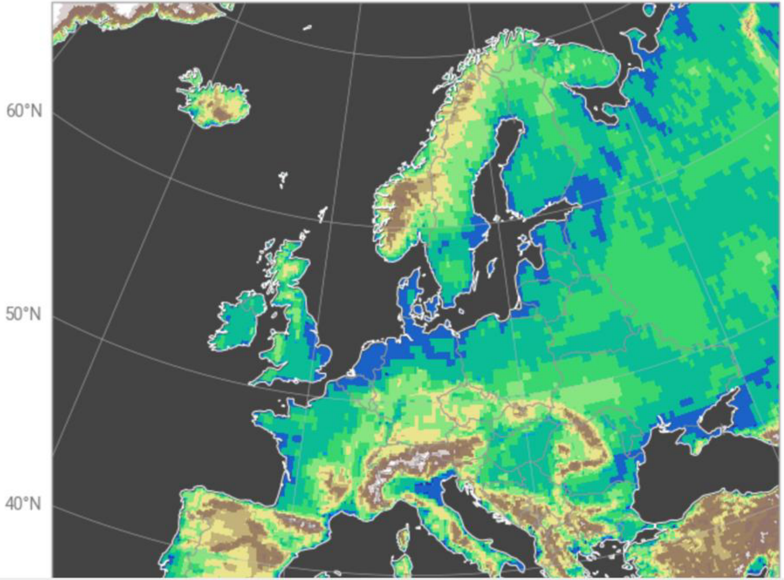
Code Download Python (Pyodide)

```
[7]: # Compress the geopotential (m^2/s^2) using ZFP

z = CodecStack(
    Zfp(mode="fixed-accuracy", tolerance=10000)
).encode_decode_data_array(data[0].to_xarray().z)
```

← **loose** error bound

```
chart = earthkit.plots.Map(domain=[-15, 35, 32, 72]) •••
```



Simple 1 Python (Pyodide) | Idle Mode: Edit Ln 1, Col 1 model-orography.ipynb 4

https://lab.climet.eu/v0.3.0/raw/github-tag/ecmwf/earthkit-plots/0.3.1/docs/examples/gallery/gridded-data/model-orography.ipynb

Demo: ClimateBenchPress

lab.climet.eu

climatebenchpress.ipynb

File Edit View Run Kernel Tabs Settings Help

Markdown Download Python (Pyodide)

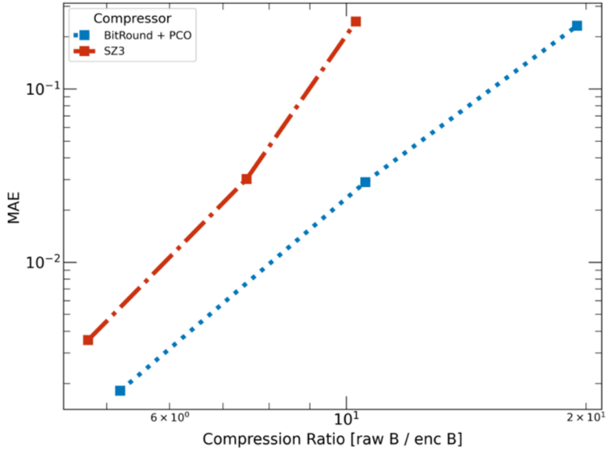
/ .sessions / 05-27-2025-14-05-03-6d83f7d4 /

Name	Modified
climatebenchpre...	41s ago
README.md	last mo.
requirements.txt	last mo.

To summarize the performance of the two compressors across all the error bounds we can make a rate-distortion plot. Interestingly, for this dataset BitRounding seems to perform better than SZ3 because BitRounding produces higher compression ratios for a similar error level.

[15]:
```
plot_variable_rd_curve(
    results,
    distortion_metric="MAE",
    outfile=plotspath / "rd_curve.png"
)
Image((plotspath / "rd_curve.png").read_bytes(), width=512)
```

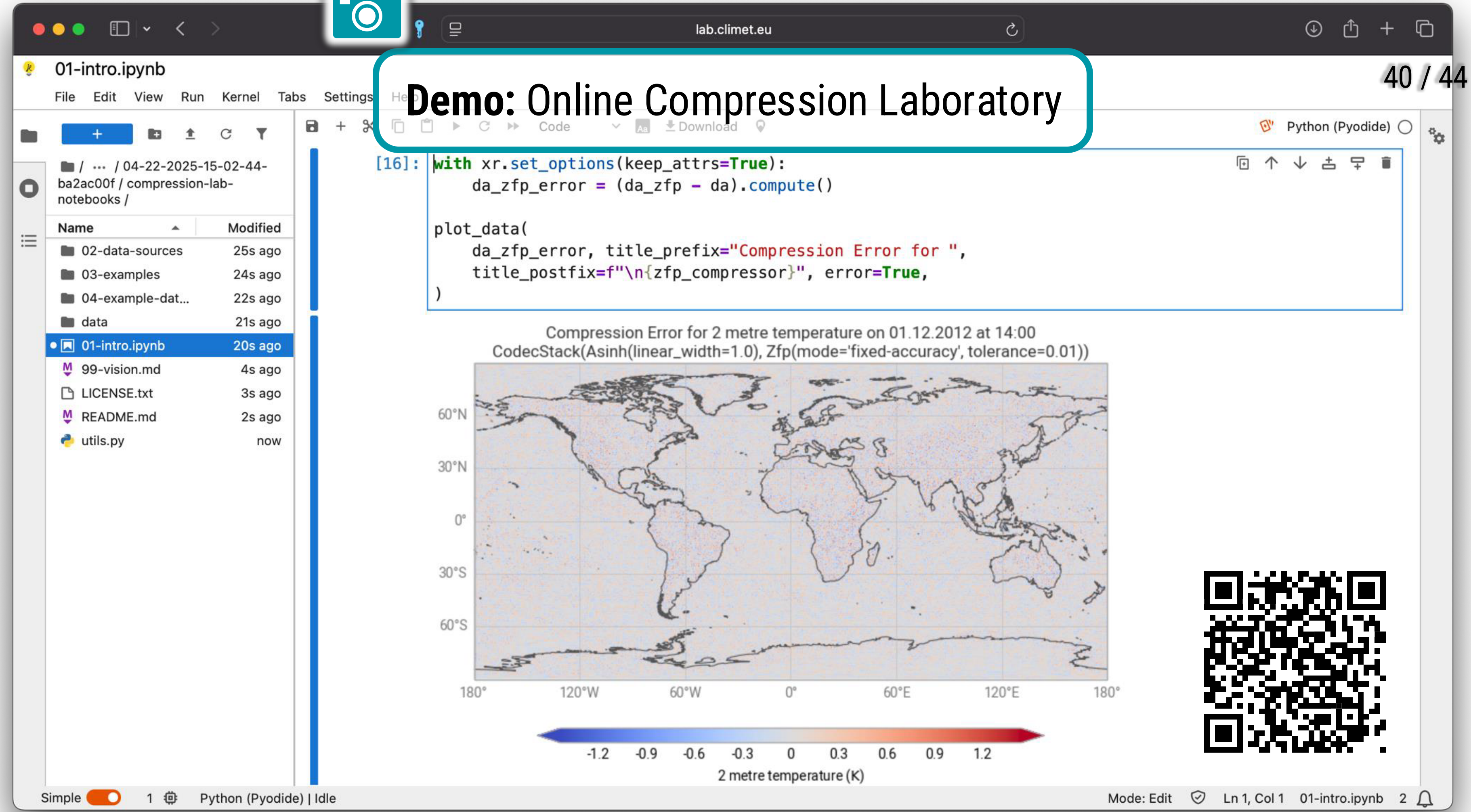
[15]:



Simple 1 Python (Pyodide) | Idle Mode: Command Ln 1, Col 1 climatebenchpress.ipynb 2

https://lab.climet.eu/v0.3.0/gist/github/juntyr/ea85adab5df19141a84939f2d322697f/climatebenchpress.ipynb?pyodideKernelLockFileURL=%22https://gist.githubusercontent.com/juntyr/ea85adab5df19141a84939f2d322697f/raw/eaa098adec0a34191302e7e61044e8553691157b/py

Demo: Online Compression Laboratory



https://compression.lab.climet.eu/v0.3.0

How can **You** benefit from the Online Laboratory today?

docs.climet.eu

Interactive and Reproducible Open Science Laboratory

Share your existing **notebooks** from repos / gists / URLs

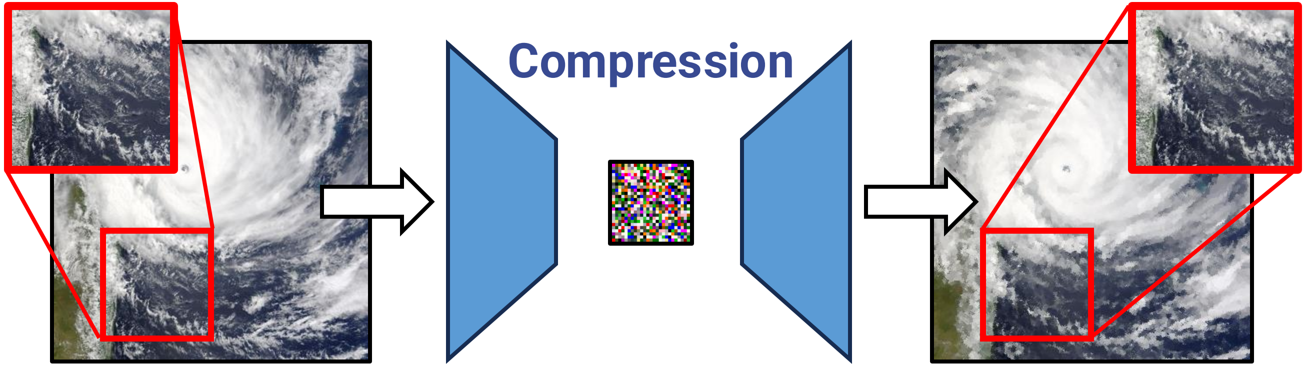
Customize the provided **packages** and **versions**

Ongoing development and support and new packages

Explore lossy data compression on climate model data

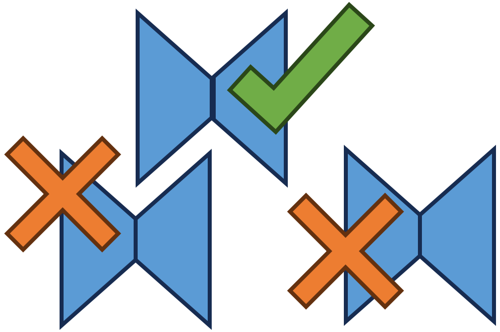
In summary ...

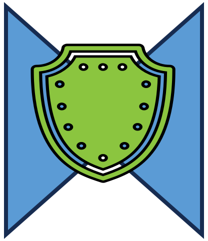
Weather & Climate Prediction will soon produce too much data



Paralyzing fear of compression artifacts or loss of details

Safe and *Fearless* Lossy Data Compression in Climate Science





Juniper Tyree
juniper.tyree@helsinki.fi

Towards Safe and Fearless Lossy Compression of Weather and Climate Model Data

Thank You for Your attention!

Juniper Tyree

0000-0002-7923-9609

juniper.tyree@helsinki.fi

juntyr

Disclaimer: *Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European High Performance Computing Joint Undertaking (JU). Neither the European Union nor the granting authority can be held responsible for them.*

CONTINENTS Webinar, 28.05.2025

Funded by the European Union. This work has received funding from the European High Performance Computing Joint Undertaking (JU) under grant agreement No 101093054.

Co-funded by the European Union