

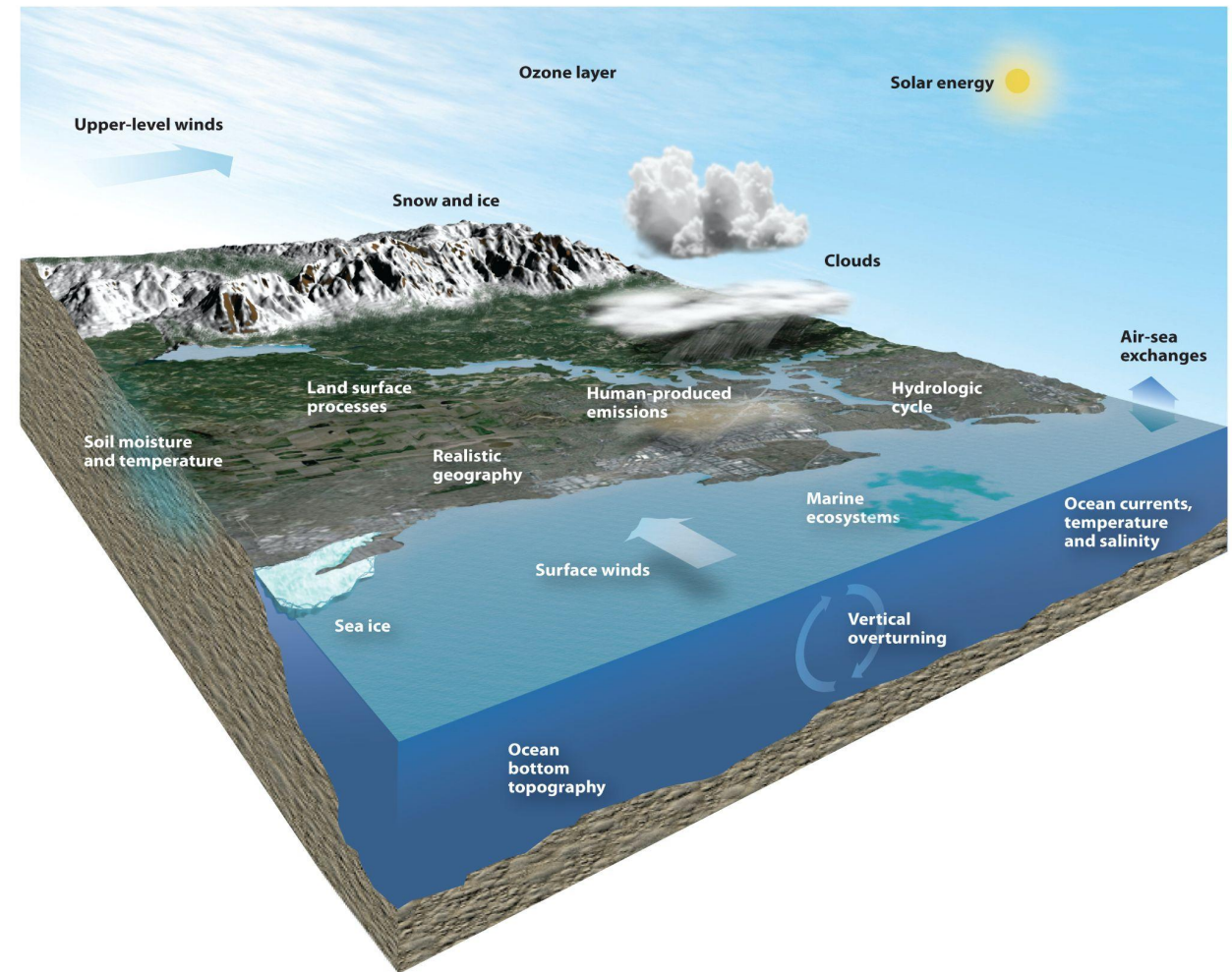
Training Statistical Models to Identify Optimal Lossy Compression Parameters

Alex Pinard¹, Allison Baker², Dorit Hammerling¹

¹Colorado School of Mines, ²National Center for Atmospheric Research

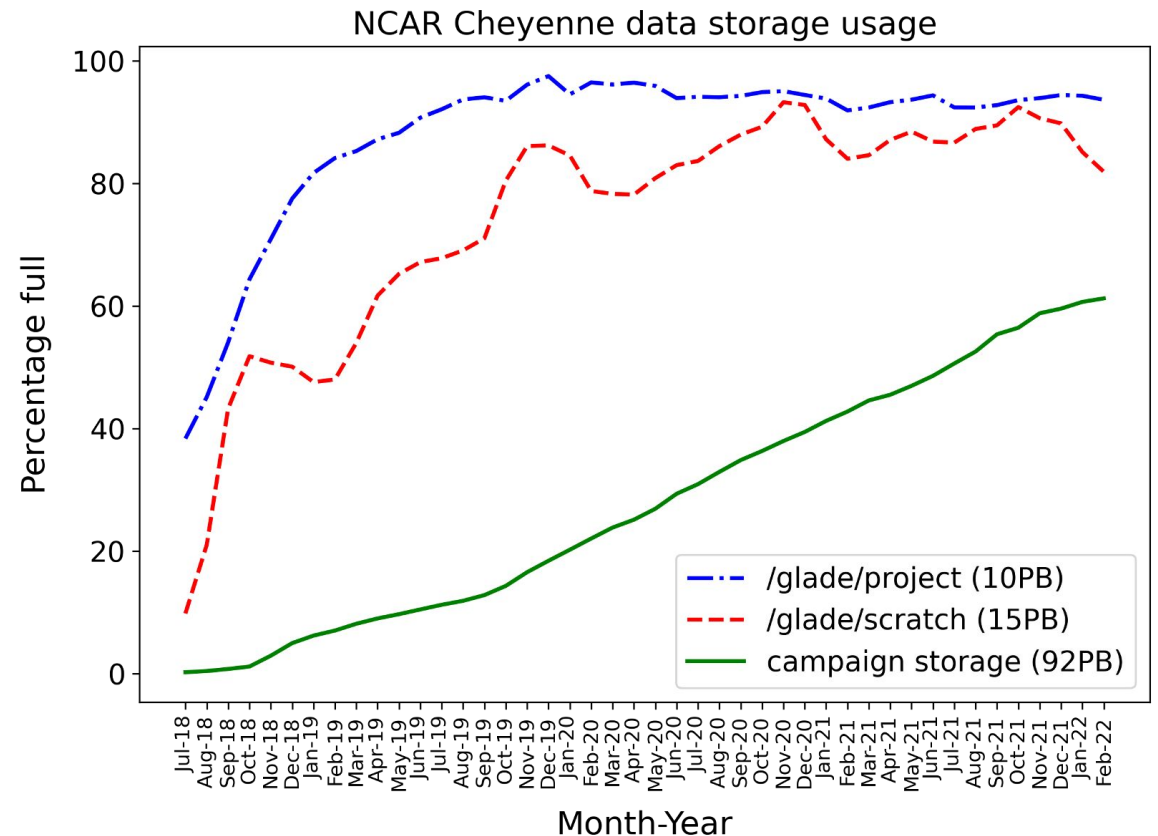
Data Storage for ESMs

- Climate simulations such as the Community Earth System Model (or CESM) have been used in large-scale projects such as the Coupled Model Intercomparison Project Phase 6.
- The total size of the output for an ensemble is massive (multiple petabytes).
- Goal: reducing the volume of these datasets without systematically altering them in any way that could affect scientific conclusions.
- We do not know in advance what kind of analysis the climate scientists will be performing on the data - or what the societal implications may be.



Reducing Data Size

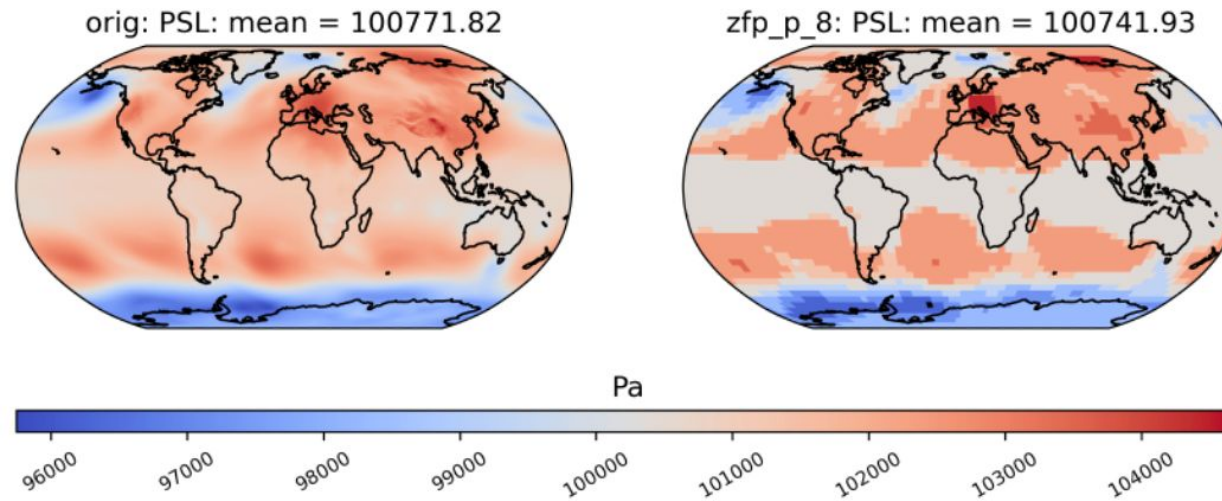
- Lossless compression algorithms do not effectively reduce data volume of floating-point data.
- As a result, scientists are forced to constrain the size of their models.
- Using lossy compressors can greatly reduce the data size, but this comes at a cost of data quality – so a tradeoff must be made.



Compressing Data Safely

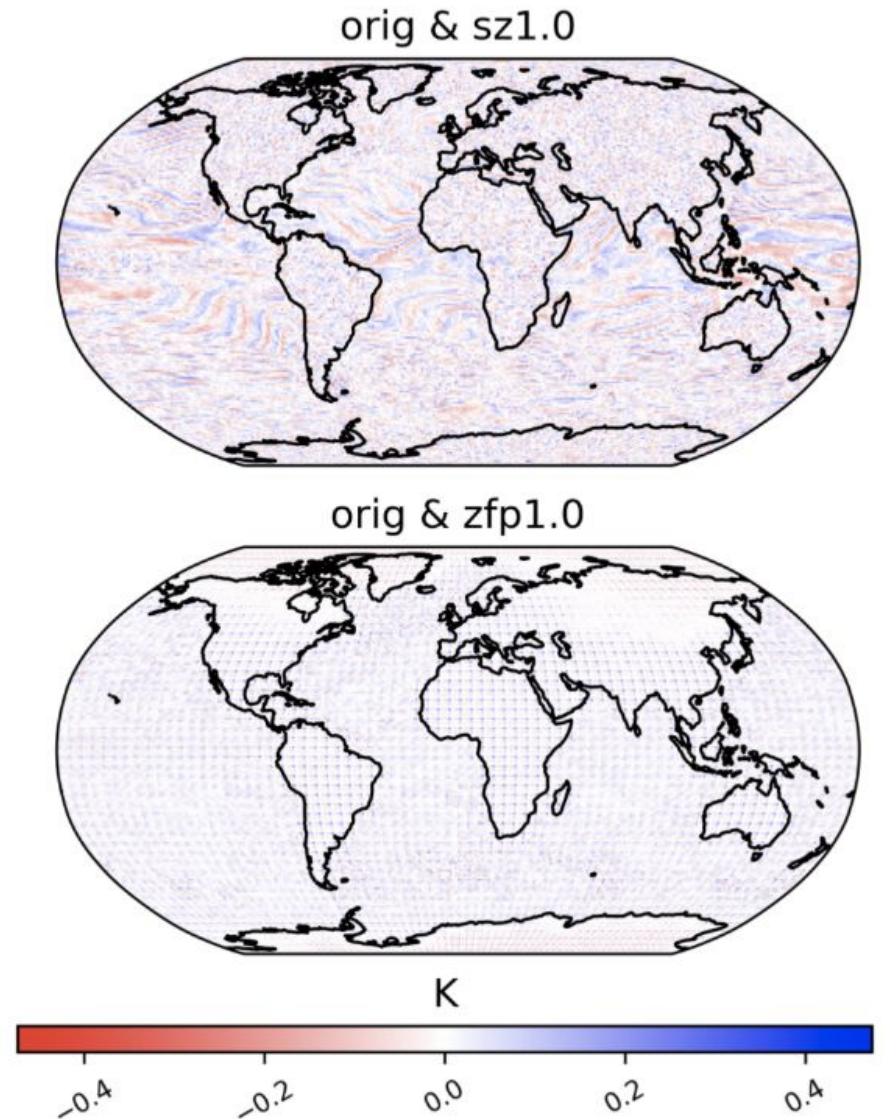
Scientists are understandably concerned about compression affecting the results of their analysis. We take the following steps to reduce the potential biases introduced in the data:

- Collaboration with compression algorithm creators to reduce artifacts in the data.
- Treating each climate variable individually to preserve **spatiotemporal** properties in a computationally efficient way.
- Working closely with application scientists, and providing tools so they can see the effects of compression on their analyses.



Evaluating Compression Quality

- Common compression metrics including RMSE, PSNR, and maximum error are not sufficient as they do not capture spatial or temporal dependencies that may exist in the errors. These may vary greatly between climate variables.
- Ensuring that compression does not adversely affect user analysis requires more **specialized metrics** that can be quickly computed on a dataset.



Example: Structural Similarity Index Measure

- Often scientific decisions are made based on visual inspection of data. The SSIM provides an estimate of similarity between two images by taking corresponding subsets of the images and evaluating luminance, contrast, and brightness. Then, these local SSIM values are averaged to reach the final mean SSIM value for comparing the images.

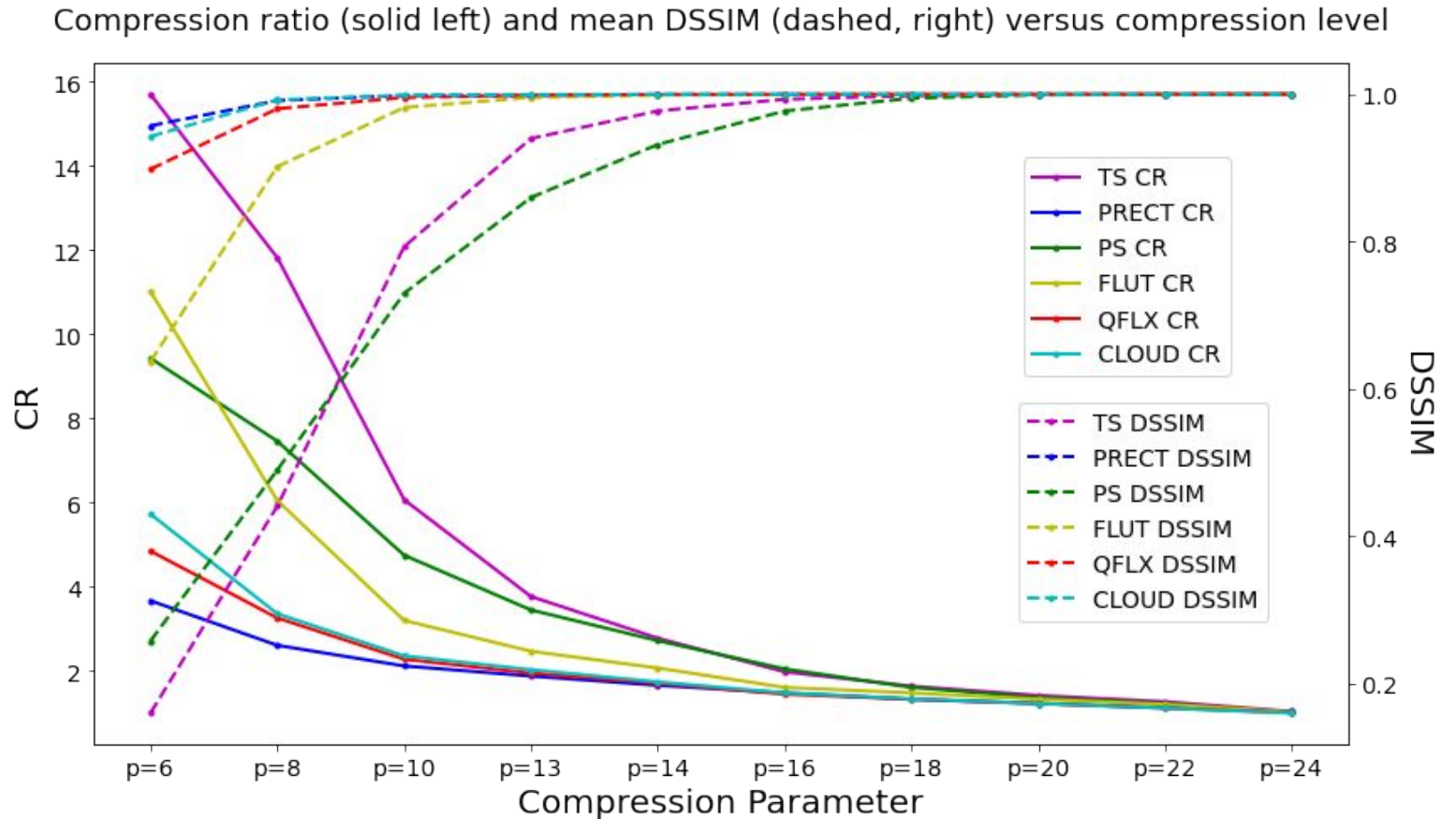
Definition (SSIM)

$$SSIM(\mathbf{x}_i, \mathbf{y}_i) = S_1(\mathbf{x}_i, \mathbf{y}_i)S_2(\mathbf{x}_i, \mathbf{y}_i),$$
$$S_1(\mathbf{x}_i, \mathbf{y}_j) = \frac{(2\mu_{\mathbf{x}_i}\mu_{\mathbf{y}_i} + C_1)}{(\mu_{\mathbf{x}_i}^2 + \mu_{\mathbf{y}_i}^2 + C_1)}, \quad S_2(\mathbf{x}_i, \mathbf{y}_j) = \frac{(2\sigma_{\mathbf{x}_i\mathbf{y}_i} + C_2)}{(\sigma_{\mathbf{x}_i}^2 + \sigma_{\mathbf{y}_i}^2 + C_2)},$$
$$SSIM(\mathbf{X}, \mathbf{Y}) = \frac{1}{M} \sum_{i=1}^M SSIM(\mathbf{x}_i, \mathbf{y}_i)$$



SSIM for Data (DSSIM)

- The DSSIM calculation is similar to the SSIM, but operates directly on datasets.
- DSSIM works similarly to the SSIM, where 1 indicates that two datasets are identical.
- We can use a threshold for the DSSIM to determine likely visual indistinguishability.



Other Metrics

Overall goal: automate the compression process while preserving scientific integrity.

Additional compression metrics:

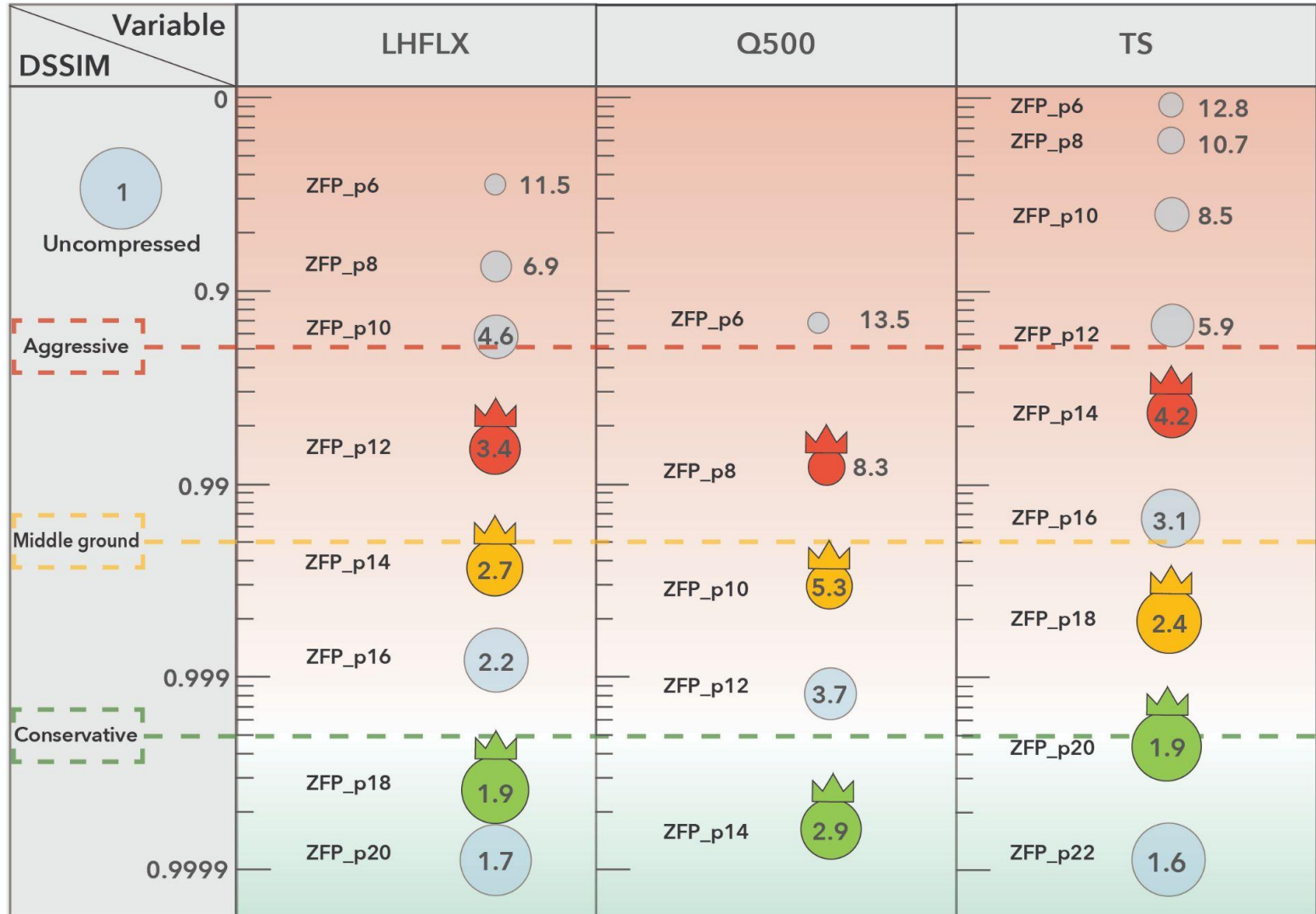
- Pearson Correlation Coefficient
- Spatial Relative Error
- Kolmogorov-Smirnov p-value
- Real Information Content

The choice of proper metrics may be highly dependant on the scientific application - these are a few examples.

pearson correlation coefficient	1
ks p-value	1
spatial relative error(% > 0.0001)	5.20833
spatial relative error (% > 0.001)	0
spatial relative error (% > 0.01)	0
max spatial relative error	0
data SSIM	0.999514

Determining “Optimal” Compression

- We determine “optimal” compression as the highest level of compression that passes the suite of metrics.
- The colored circles in the right figure indicate the dataset considered “optimal” under different metric threshold values, in this case for the DSSIM.
- This process is repeated for the other metrics and the lowest compression level over all metrics is taken as optimal.



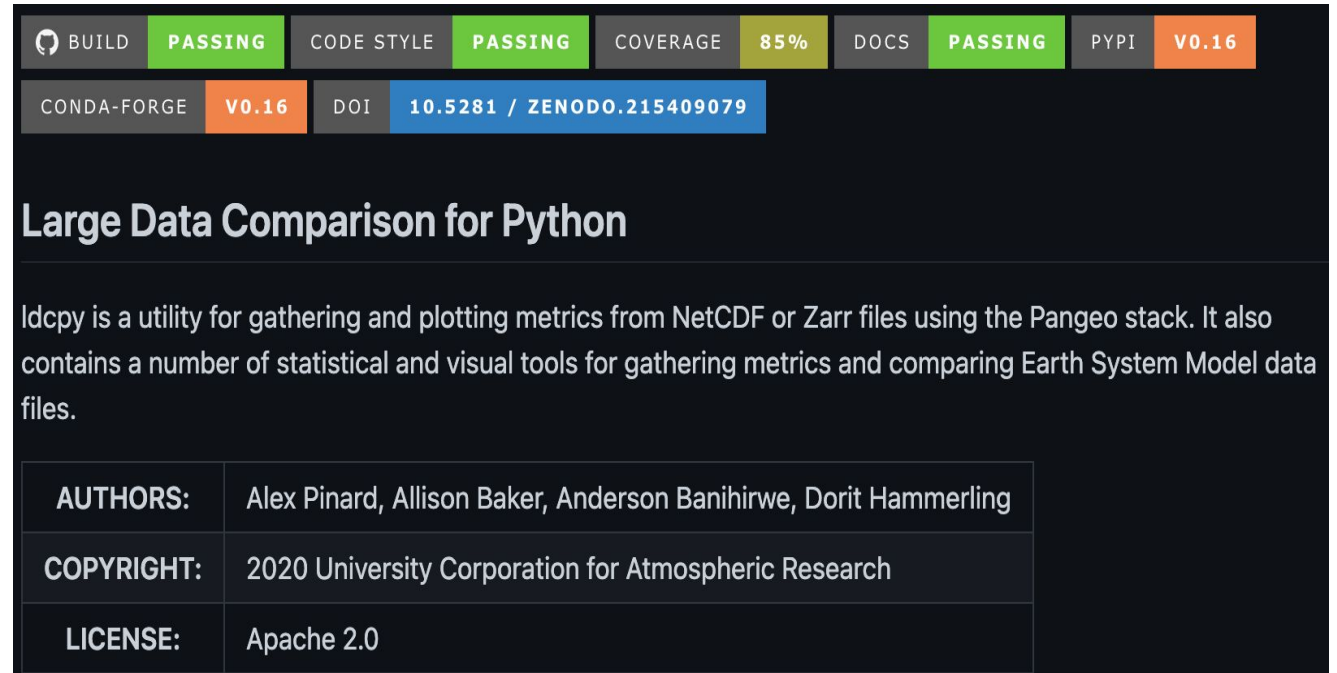
CR = Compression Ratio    Optimal compression level  Non-optimal compression level

Practical Compression Challenges

- We have thousands of variables and large volumes of data.
- Each climate variable has different characteristics, and characteristics may vary between time slices. Additionally, new variables may be output at any time.
- To ensure the highest level of compression possible, we would need to try many different compression algorithms and parameter combinations separately for each individual time slice of output data.
- We are working on creating a statistical model that takes in data (or derived quantities thereof) of new or preexisting climate variables and predicts the ideal compression settings. This will be used as a baseline against which the compression can be further tweaked by application.

Idcpy

- To compute metrics on massive spatial datasets, we developed a Python software package called Idcpy.
- This package also allows us to calculate other derived quantities of the data, and provides visualization tools.
- Idcpy design goals:
 - Interoperability with the Pangeo software ecosystem
 - Easy interaction through Jupyter Notebooks
 - Suitability for a wide range of data volumes (single time slice to many years)
 - Supports datasets in NetCDF and object store data formats
 - Extensible analysis and plotting capabilities



CONDA-FORGE **V0.16** DOI **10.5281 / ZENODO.215409079**

Large Data Comparison for Python

Idcpy is a utility for gathering and plotting metrics from NetCDF or Zarr files using the Pangeo stack. It also contains a number of statistical and visual tools for gathering metrics and comparing Earth System Model data files.

AUTHORS:	Alex Pinard, Allison Baker, Anderson Banihirwe, Dorit Hammerling
COPYRIGHT:	2020 University Corporation for Atmospheric Research
LICENSE:	Apache 2.0

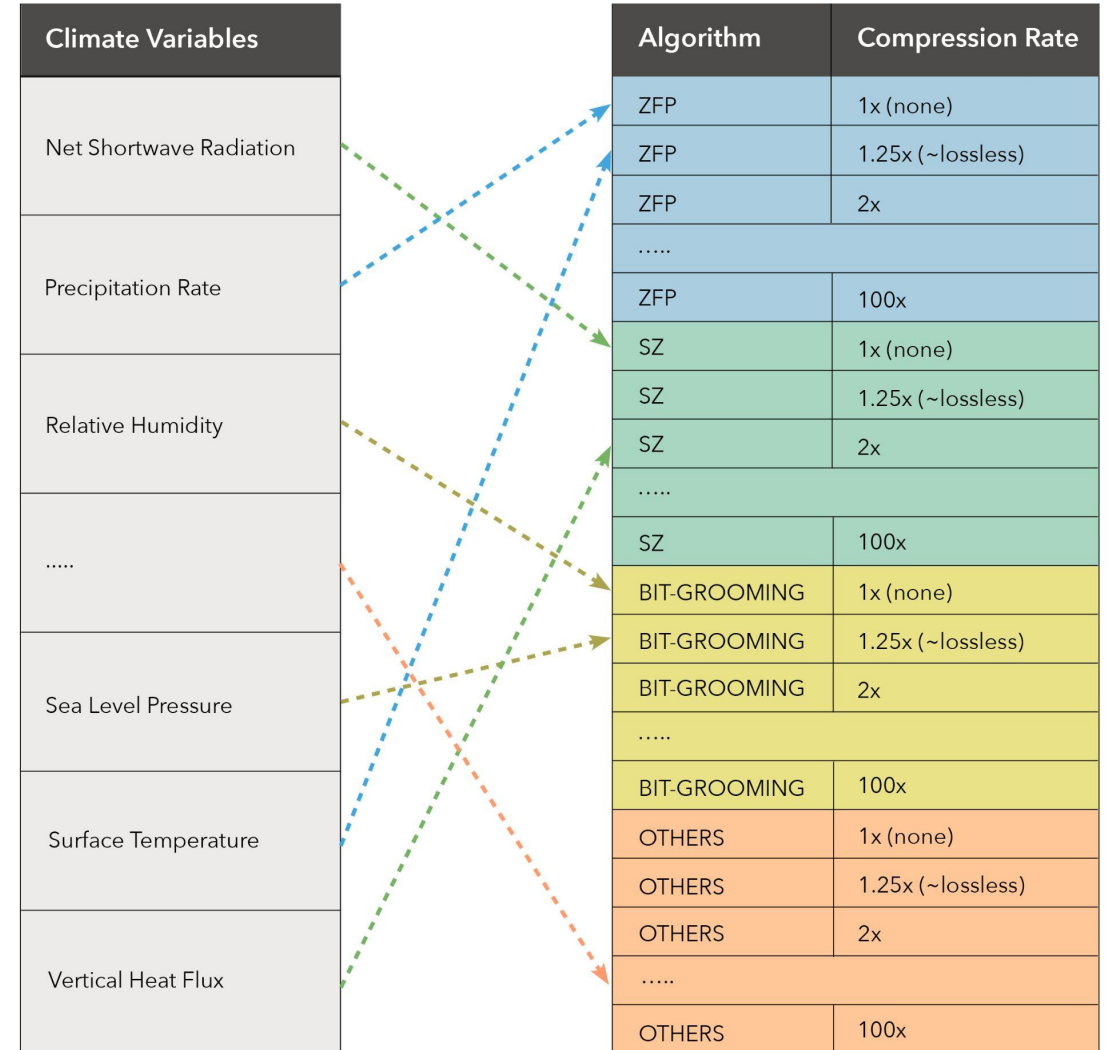
Label spread by variable

- For each climate variable, we look at single time slices of the spatial field and classify them according to their optimal compression level.
- The result are optimal compression levels for every climate variable, with varying level distributions for each climate variable.

zfp_level variable	6	8	10	12	14	16	18	20	22	24	26	never passed
ABSORB	0	0	0	0	0	0	51	9	0	0	60	240
ANRAIN	360	0	0	0	0	0	0	0	0	0	0	0
ANSNOW	360	0	0	0	0	0	0	0	0	0	0	0
AODABS	0	0	0	0	5	47	7	1	0	0	60	240
AODDUST1	0	0	0	0	30	29	1	0	0	0	60	240
AODDUST2	60	0	0	0	0	0	0	0	0	0	60	240
AODDUST3	0	0	0	2	48	10	0	0	0	0	60	240
AODVIS	0	0	0	0	0	41	19	0	0	0	60	240
AQRAIN	360	0	0	0	0	0	0	0	0	0	0	0
AQSNOW	360	0	0	0	0	0	0	0	0	0	0	0
AREI	360	0	0	0	0	0	0	0	0	0	0	0
AREL	360	0	0	0	0	0	0	0	0	0	0	0
AWNC	360	0	0	0	0	0	0	0	0	0	0	0
AWNI	360	0	0	0	0	0	0	0	0	0	0	0
BURDENBC	0	0	0	0	3	239	117	1	0	0	0	0
BURDENDUST	0	0	0	26	288	44	2	0	0	0	0	0
BURDENPOM	0	0	0	0	14	268	78	0	0	0	0	0
BURDENSEASALT	0	0	0	0	0	318	41	1	0	0	0	0
CCN3	0	0	0	0	0	3	287	70	0	0	0	0
CDNUMC	0	0	0	0	38	318	4	0	0	0	0	0
CLDTOT	0	0	0	0	188	172	0	0	0	0	0	0
CO2_LND	0	0	0	0	0	0	0	0	0	0	0	360
CO2_OCN	0	0	0	0	0	0	0	0	0	0	0	360
DCQ	121	5	56	136	40	2	0	0	0	0	0	0
DTCOND	202	4	27	86	41	0	0	0	0	0	0	0
DTV	359	0	1	0	0	0	0	0	0	0	0	0
EXTINCT	0	0	0	0	0	0	51	9	0	0	60	240
FICE	334	1	2	11	12	0	0	0	0	0	0	0
FLDS	0	0	0	0	0	0	328	32	0	0	0	0
FLNS	0	0	0	0	0	360	0	0	0	0	0	0
FLNSC	0	0	0	0	0	0	360	0	0	0	0	0
FLNTC	0	0	0	0	0	0	7	353	0	0	0	0
FLUTC	0	0	0	0	0	0	3	353	4	0	0	0

Classifying Datasets

- The approach described previously only works for small datasets.
- We can treat this as a supervised learning problem and try to model the optimal level based on dataset features.
- The model is intended to be metric-agnostic.



Generating features using Idcpy

- We use explicit feature models, such as random forest models, to predict optimal compression levels and indicate which features are relevant to making predictions.
- We also use implicit feature models, in this case CNNs, as they are designed to capture regularities in image data.

climate variable	mean importance	standard deviation
ns_con_var	0.096	0.050
ew_con_var	0.11	0.065
w_e_first_differences	0	0
w_e_first_differences_max	0.11	0.043
n_s_first_differences	0.055	0.042
n_s_first_differences_max	0.053	0.041
FFT_max_horizontal	0.049	0.028
FFT_horizontal_ratio	0.056	0.045
FFT_max_vertical	0.049	0.022
FFT_vertical_ratio	0.051	0.042
magnitude_range	0.041	0.026
magnitude_range_ew	0.066	0.044
magnitude_range_ns	0.016	0.014
entropy	0.11	0.048
real_information	0.14	0.082

Early Results

- Results using basic statistical learning models are mixed.
- Higher accuracy for predicting a new timeslice of a preexisting climate variable versus predicting a timeslice for a previously unforeseen climate variable.
- Major issues: difficult to discern additional features, require more data to fully explore the feature space than the 183 variables we have.

Target (True) Class	zfp_p_14	1217	207
	zfp_p_20	471	751
		zfp_p_14	zfp_p_20

Output (Predicted) Class

Mean test classification matrix after fitting a random forest to all the training data except a single climate variable, for two (zfp) parameter sets: mean accuracy: 74.5%

Output Class	zfp_p_14	174 62%	23 8%	73 73%	0 0%	0 0%	0 0%	0 0%
	zfp_p_16	98 35%	24 8%	27 27%	0 0%	0 0%	0 0%	0 0%
	zfp_p_18	1 0%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%
	zfp_p_20	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%
	zfp_p_22	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%
	zfp_p_24	0 0%	198 66%	0 0%	0 0%	0 0%	0 0%	0 0%
	zfp_p_26	7 2%	53 18%	0 0%	0 0%	0 0%	0 0%	0 0%
		zfp_p_14	zfp_p_16	zfp_p_18	zfp_p_20	zfp_p_22	zfp_p_24	zfp_p_26

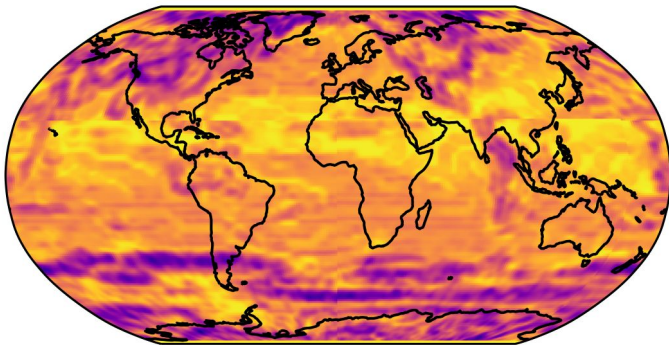
Target Class

Accuracy can drop significantly when training on more than two classes.

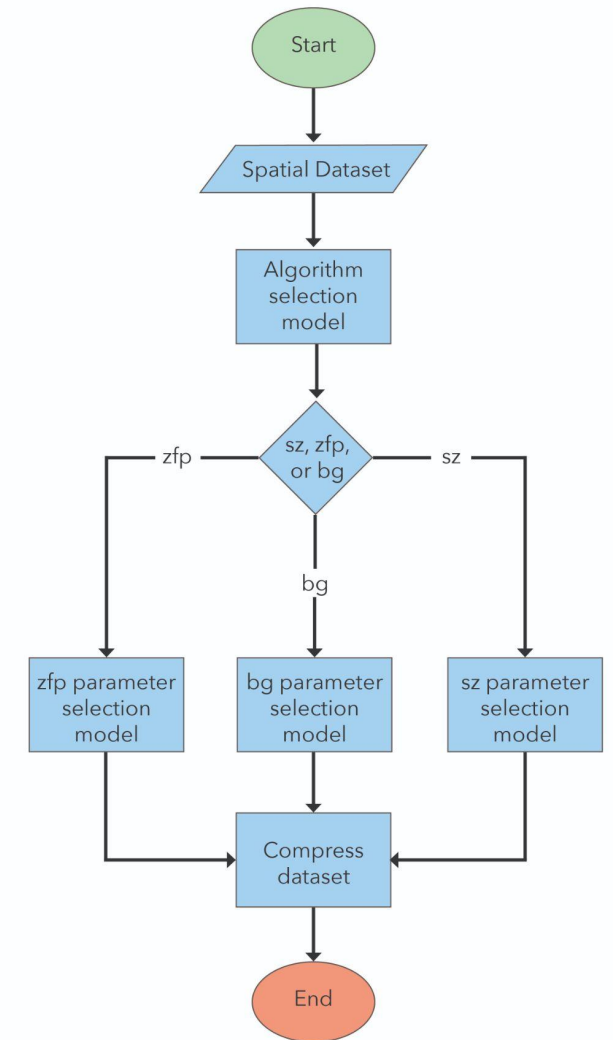
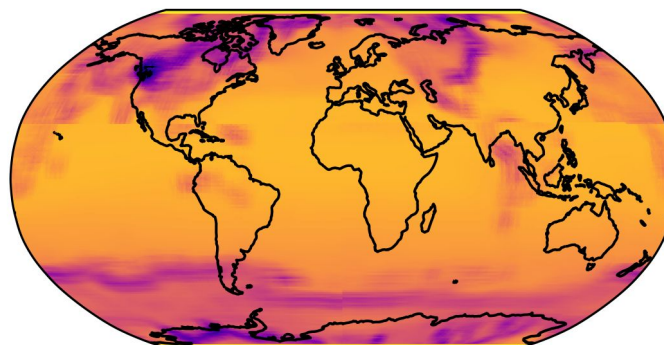
Continuing Work

- We are looking at using local spatial features to determine optimal compression levels, increasing the amount and variety of input training data.
- Training time / number of models to train is a major drawback. Improving code parallelism is a big focus.
- Development of a multi-stage model that first predicts the optimal compression algorithm, and then selects the appropriate compression parameter for the algorithm.

Actual DSSIMs: dssims: mean = -1.67



Model Predictions: dssims: mean = -1.60



Thank You!

Further Reading:

- A. H. Baker, H. Xu, D. M. Hammerling, S. Li, and J. P. Clyne, “Toward a multi-method approach: Lossy data compression for climate simulation data,” in International Conference on High Performance Computing. Springer, 2017, pp. 30–42.
- A. Pinard, A. H. Baker, and D. M. Hammerling, “A statistical approach to obtaining a data structural similarity index cutoff threshold,” National Center for Atmospheric Research, Tech. Rep. NCAR/TN-568+STR, 2021.
- “Examining variations in the optimal compression level of spatiotemporal datasets determined using the data structural similarity index measure (dssim),” National Center for Atmospheric Research, Tech. Rep. NCAR/TN-570+STR, 2021.
- A. Pinard, D. M. Hammerling, and A. H. Baker, “Assessing differences in large spatio-temporal climate datasets with a new python package,” in 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 2699–2707.