Real Information Status

Repo created for real information research files

Jupyter:

- MutualInformation.ipynb

Lit:

- DRBSD-22 Paper
- Klower Paper
- Presentations
- Peter Correspondence

pinarda update — e16840f · 16 minutes ago — 4 Commits

jupyter	update	16 minutes ago
lit	update	16 minutes ago
real_info_py	init	2 months ago
.gitignore	Initial commit	2 months ago
LICENSE	Initial commit	2 months ago
README.md	Initial commit	2 months ago
environment.yml	update	16 minutes ago

README GPL-3.0 license

real_info_py

Existing Code Functions in ldcpy and jupyter

get_dict_list: Compute current and adjacent bit probabilities

get_mutual_info: Compute mutual info for a set of probabilities

get_real_info: Uses above functions to get real info array

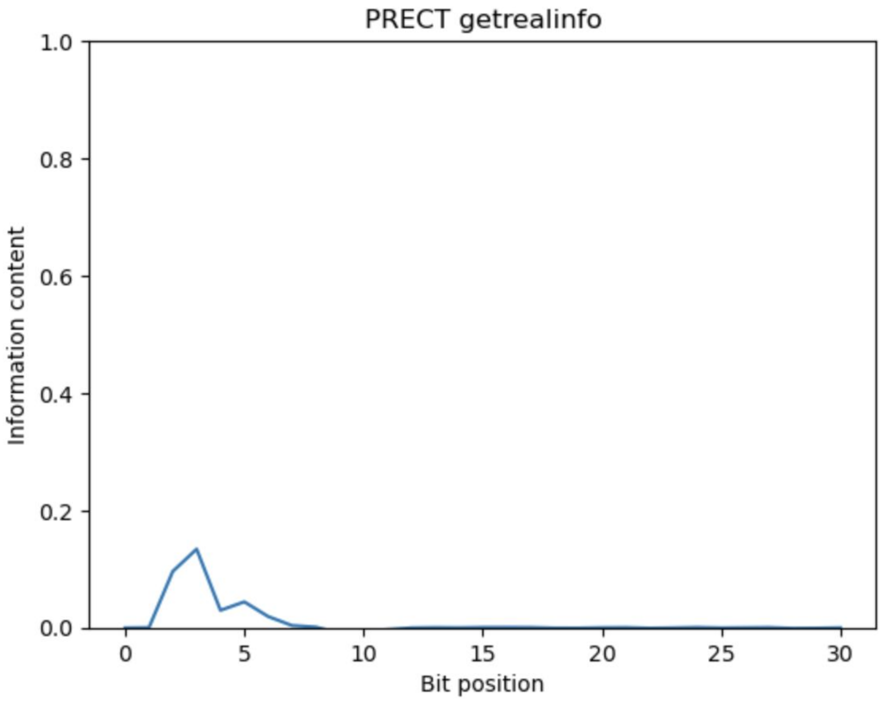
get_bit_cutoff: Get cutoff to maintain given % of real info

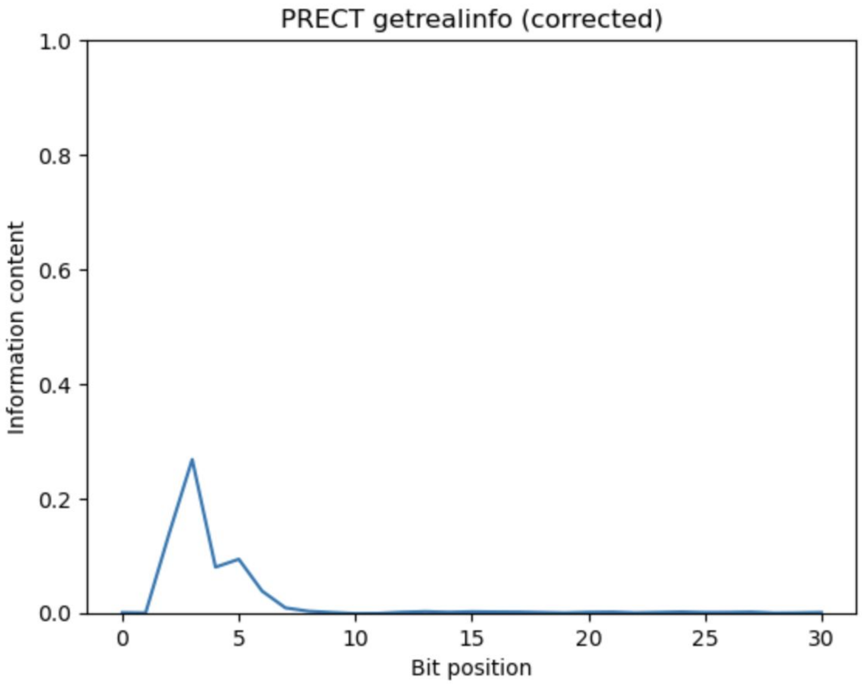
get_compression_level: Find level so that, up to cutoff bit, diff between orig and compressed data real info is small

get_real_info_all: Loop over get_real_info for multiple arrays.

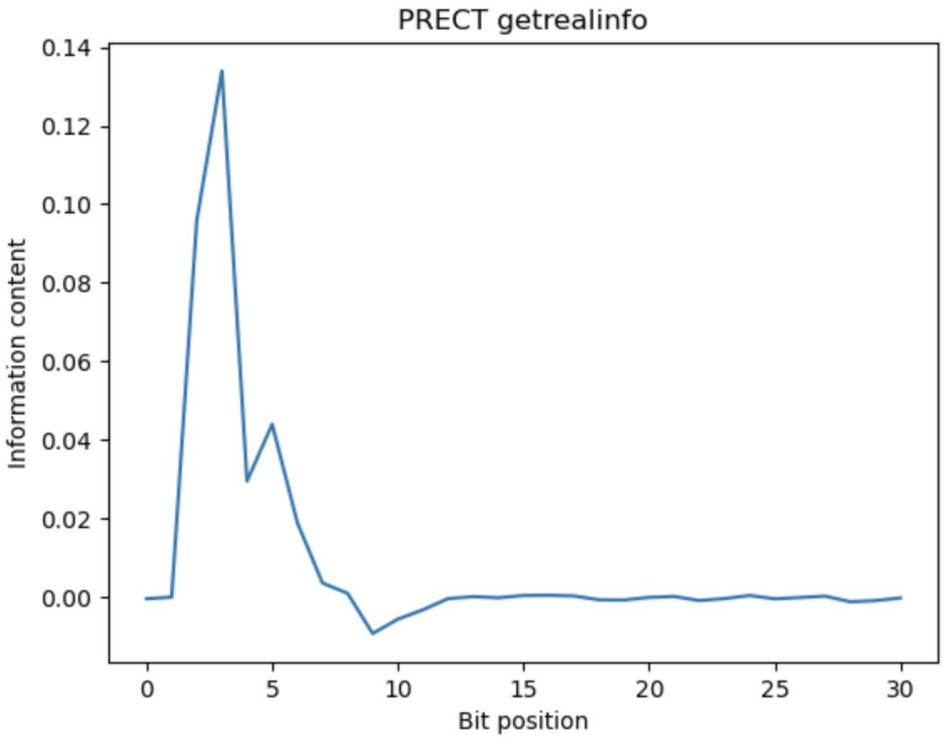
plot_real_info: Create single plot for real info over range of bits.

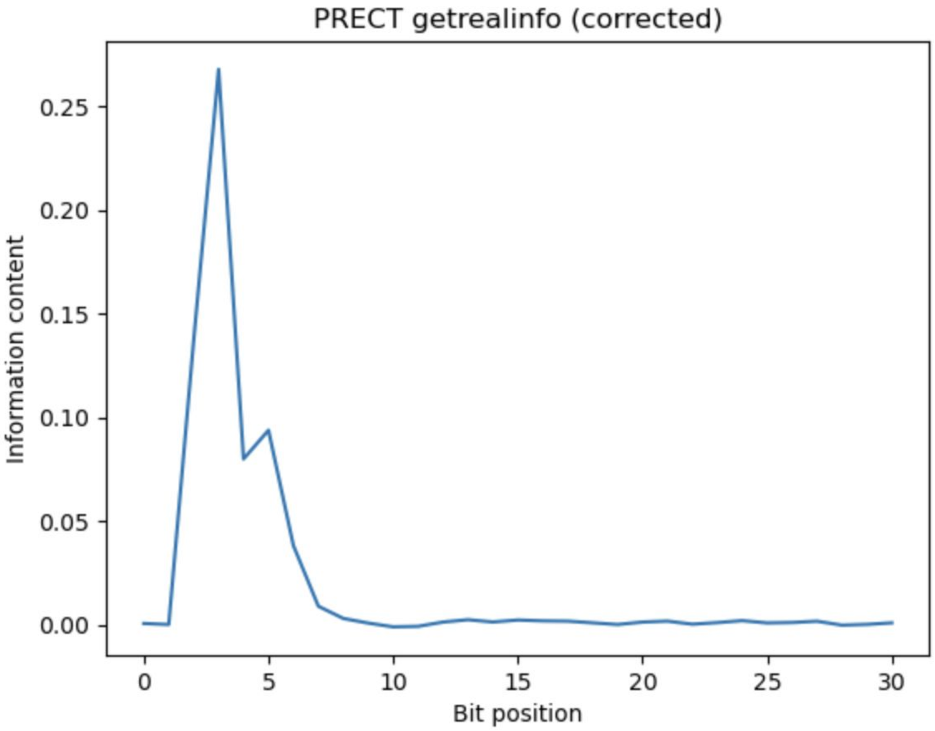
Corrected real information jupyter notebook

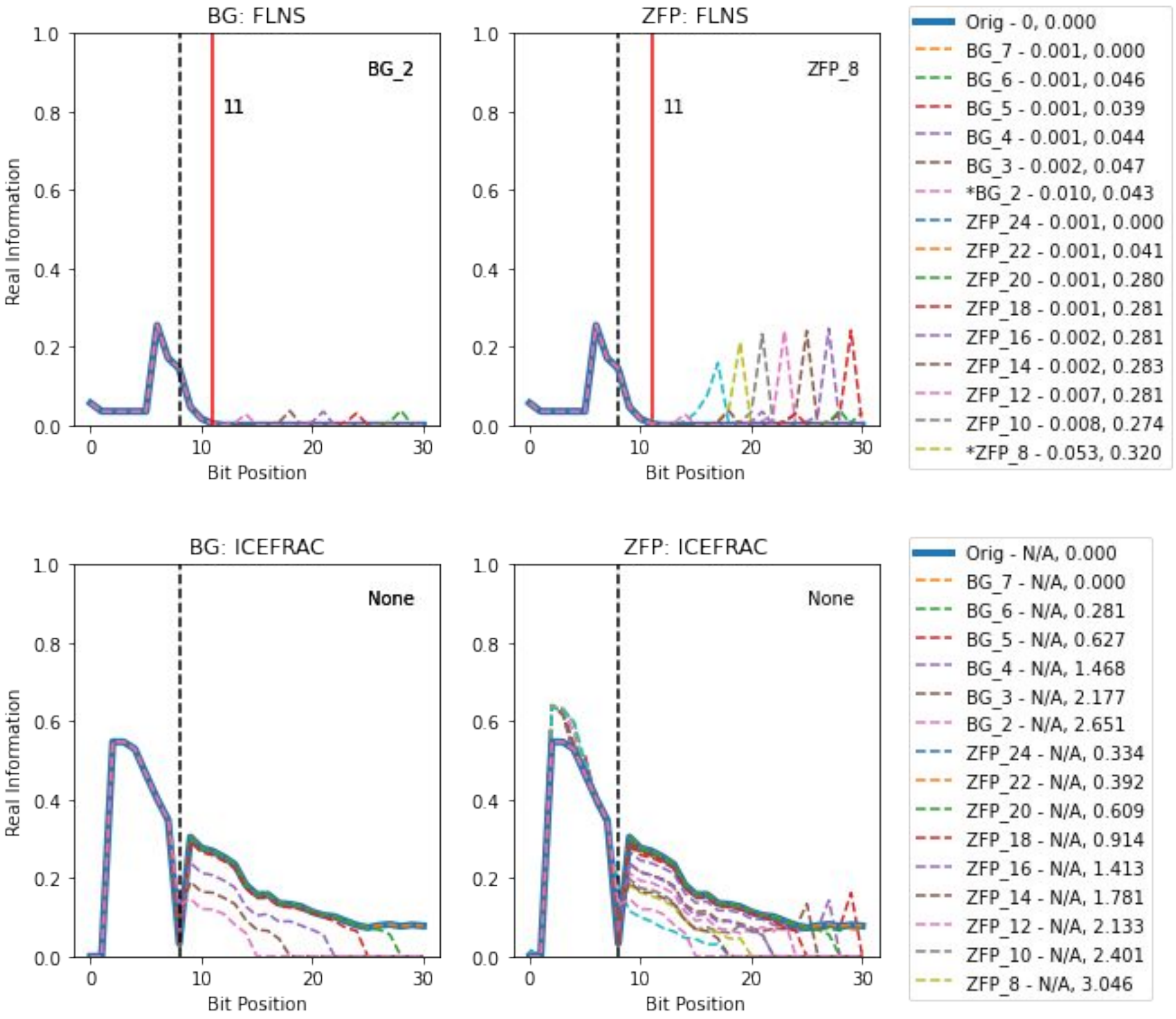


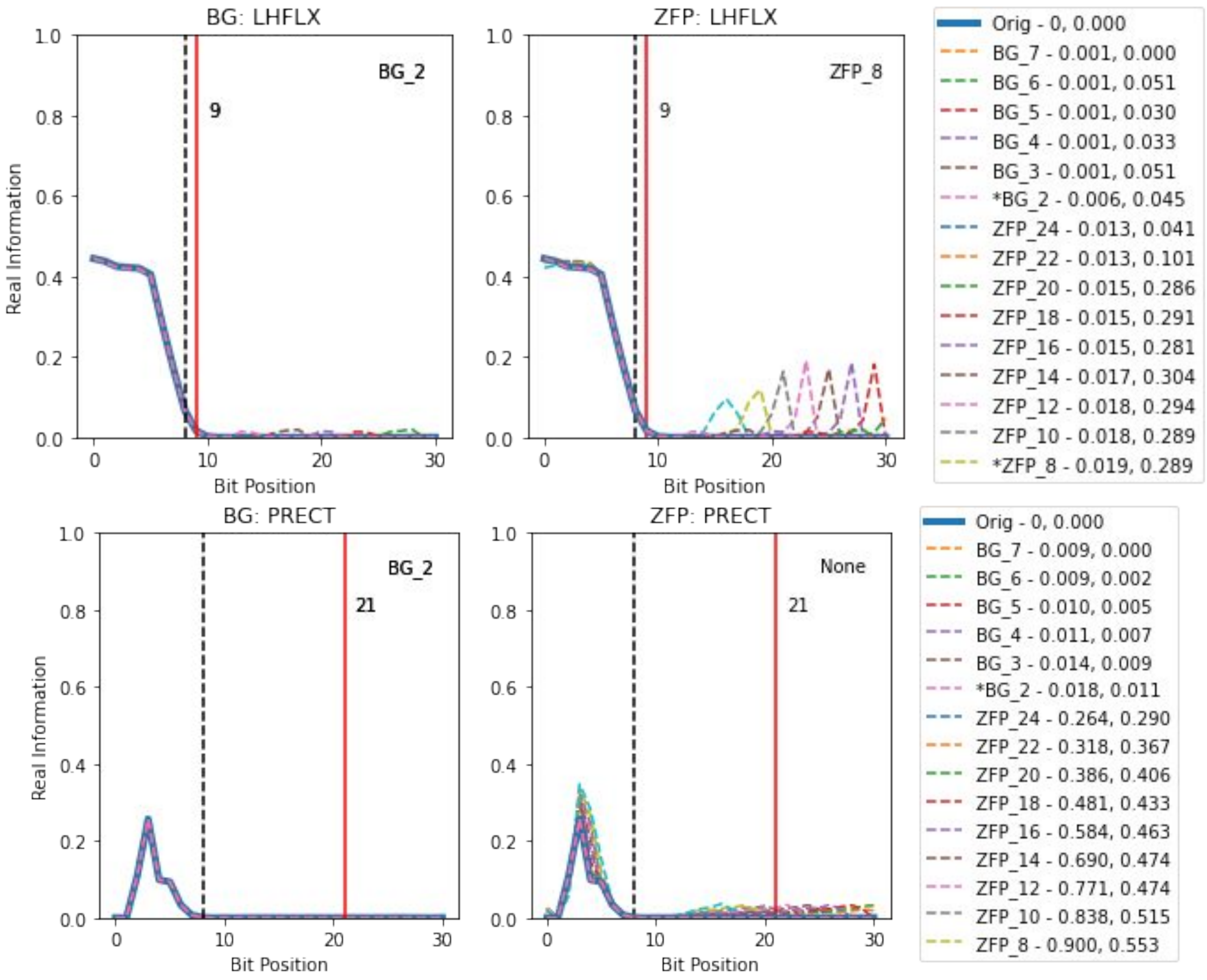


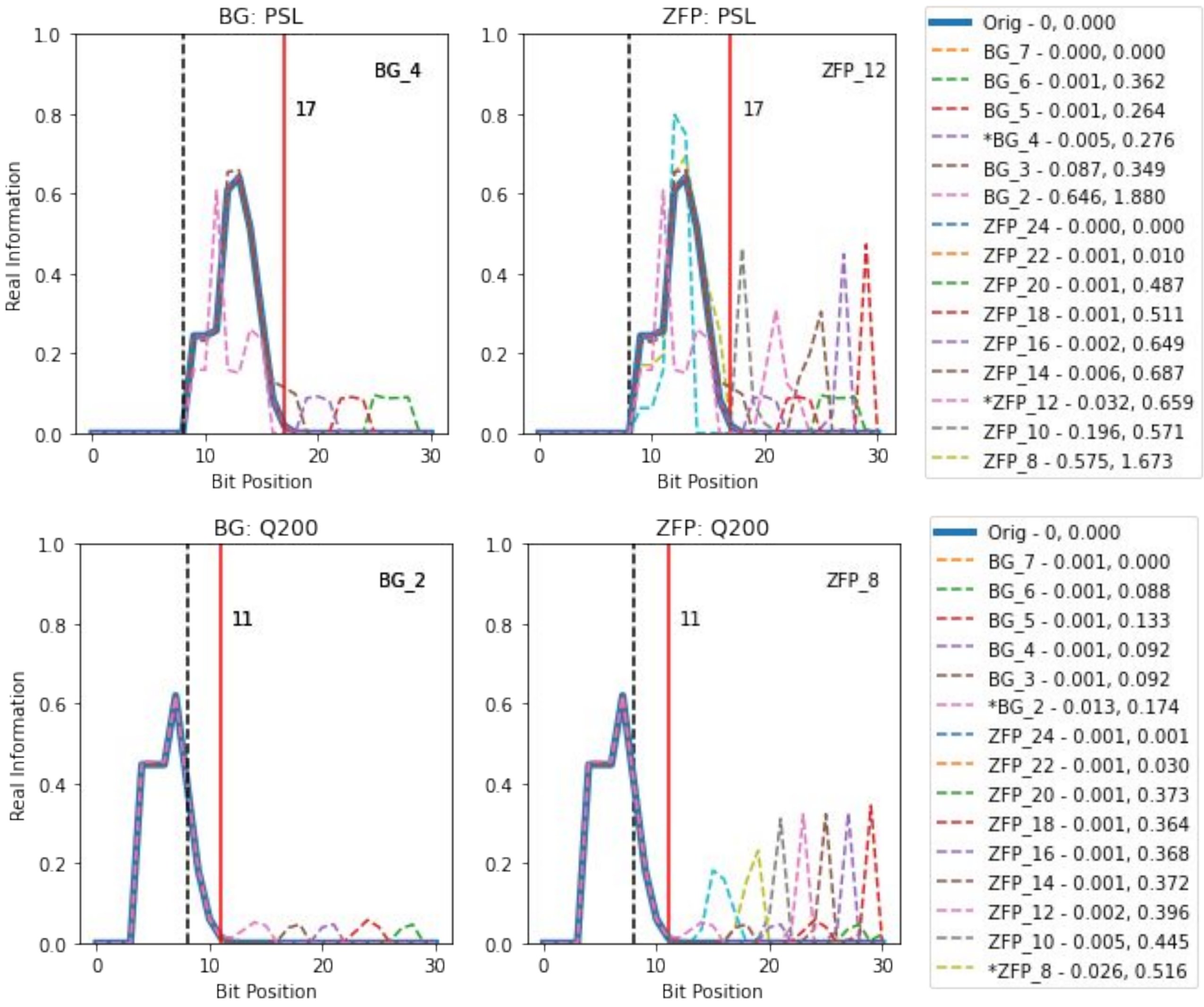
Corrected real information jupyter notebook

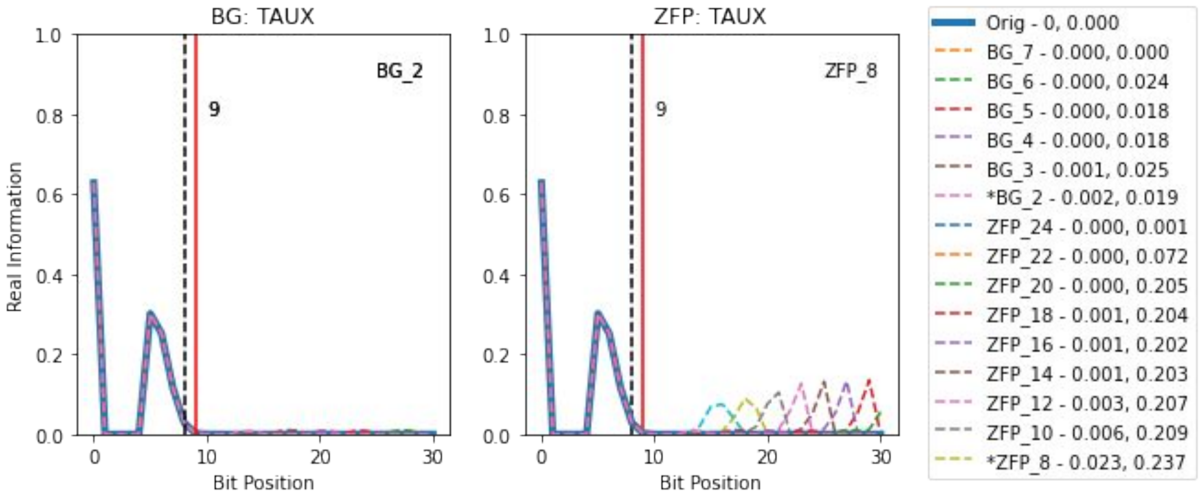


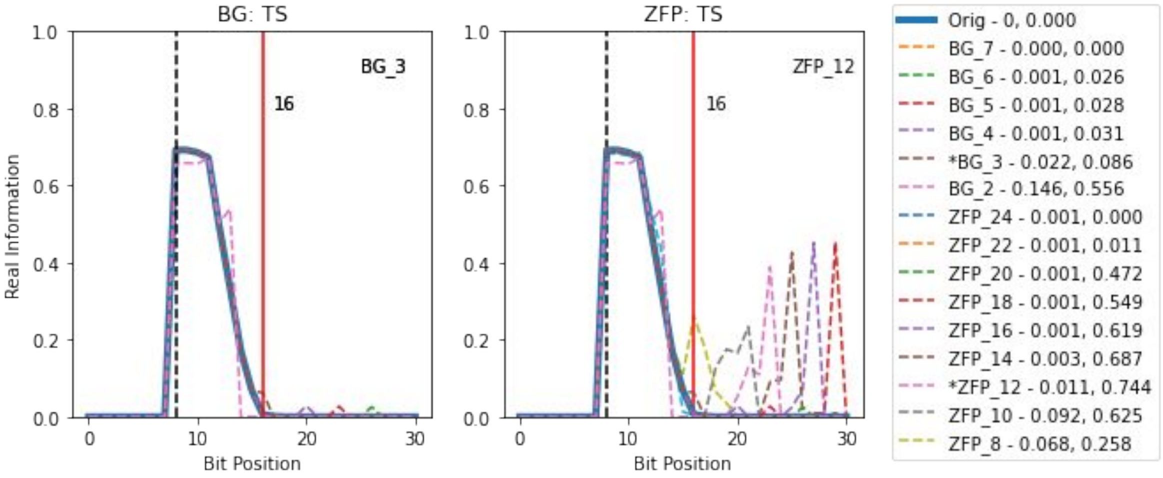


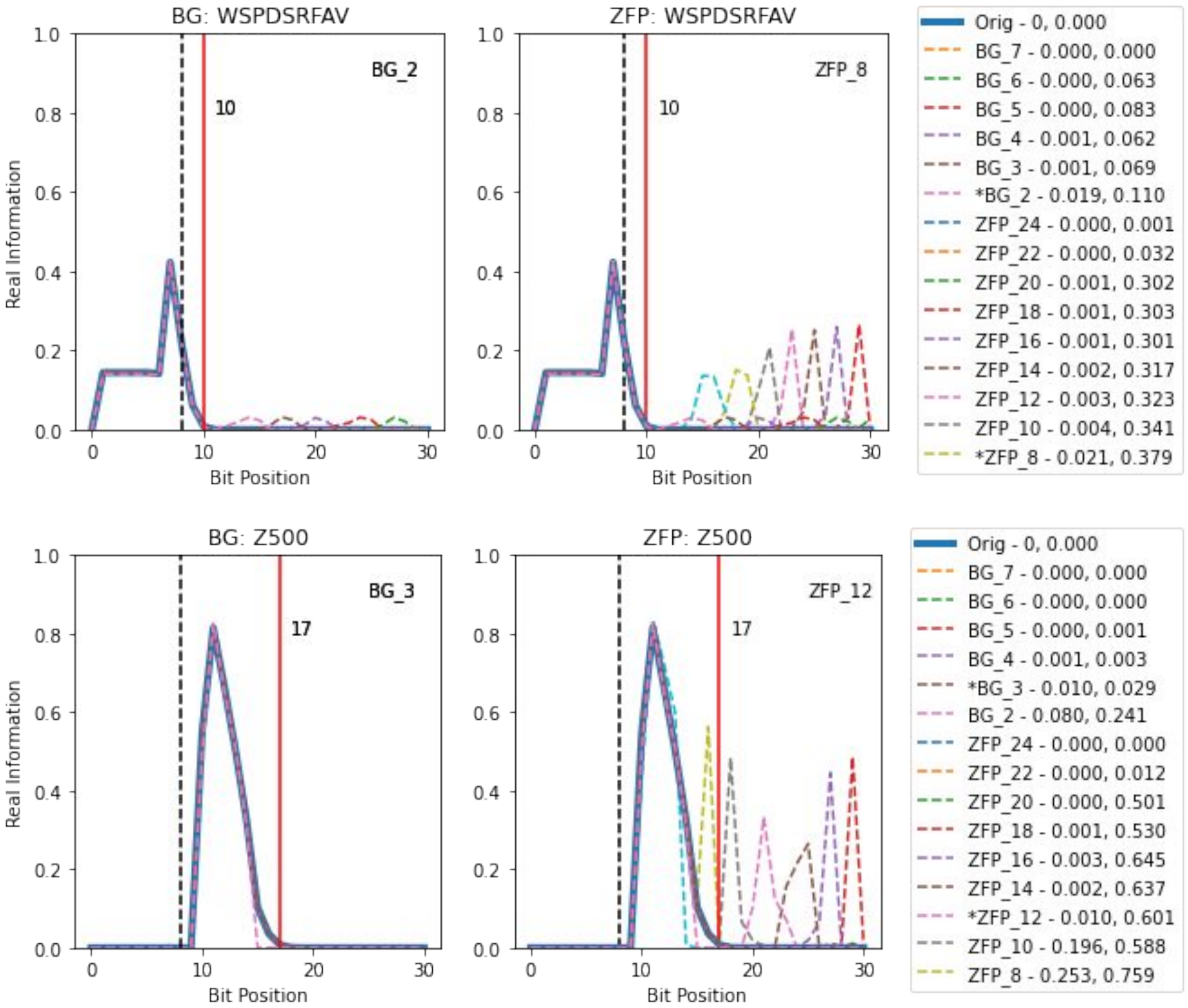

BG: FLNS
ZFP: FLNS
BG_2
ZFP_8
11
Real Information
Bit Position
Orig - 0, 0.000
BG_7 - 0.001, 0.000
BG_6 - 0.001, 0.046
BG_5 - 0.001, 0.039
BG_4 - 0.001, 0.044
BG_3 - 0.002, 0.047
*BG_2 - 0.010, 0.043
ZFP_24 - 0.001, 0.000
ZFP_22 - 0.001, 0.041
ZFP_20 - 0.001, 0.280
ZFP_18 - 0.001, 0.281
ZFP_16 - 0.002, 0.281
ZFP_14 - 0.002, 0.283
ZFP_12 - 0.007, 0.281
ZFP_10 - 0.008, 0.274
*ZFP_8 - 0.053, 0.320
BG: ICEFRAC
ZFP: ICEFRAC
None
Orig - N/A, 0.000
BG_7 - N/A, 0.000
BG_6 - N/A, 0.281
BG_5 - N/A, 0.627
BG_4 - N/A, 1.468
BG_3 - N/A, 2.177
BG_2 - N/A, 2.651
ZFP_24 - N/A, 0.334
ZFP_22 - N/A, 0.392
ZFP_20 - N/A, 0.609
ZFP_18 - N/A, 0.914
ZFP_16 - N/A, 1.413
ZFP_14 - N/A, 1.781
ZFP_12 - N/A, 2.133
ZFP_10 - N/A, 2.401
ZFP_8 - N/A, 3.046


BG: LHFLX
ZFP: LHFLX
BG: PRECT
ZFP: PRECT
BG_2
ZFP_8
BG_2
None
9
9
21
21
Real Information
Bit Position
Orig - 0, 0.000
BG_7 - 0.001, 0.000
BG_6 - 0.001, 0.051
BG_5 - 0.001, 0.030
BG_4 - 0.001, 0.033
BG_3 - 0.001, 0.051
*BG_2 - 0.006, 0.045
ZFP_24 - 0.013, 0.041
ZFP_22 - 0.013, 0.101
ZFP_20 - 0.015, 0.286
ZFP_18 - 0.015, 0.291
ZFP_16 - 0.015, 0.281
ZFP_14 - 0.017, 0.304
ZFP_12 - 0.018, 0.294
ZFP_10 - 0.018, 0.289
*ZFP_8 - 0.019, 0.289
Orig - 0, 0.000
BG_7 - 0.009, 0.000
BG_6 - 0.009, 0.002
BG_5 - 0.010, 0.005
BG_4 - 0.011, 0.007
BG_3 - 0.014, 0.009
*BG_2 - 0.018, 0.011
ZFP_24 - 0.264, 0.290
ZFP_22 - 0.318, 0.367
ZFP_20 - 0.386, 0.406
ZFP_18 - 0.481, 0.433
ZFP_16 - 0.584, 0.463
ZFP_14 - 0.690, 0.474
ZFP_12 - 0.771, 0.474
ZFP_10 - 0.838, 0.515
ZFP_8 - 0.900, 0.553


BG: PSL
BG_4
17
ZFP: PSL
ZFP_12
17
Orig - 0, 0.000
BG_7 - 0.000, 0.000
BG_6 - 0.001, 0.362
BG_5 - 0.001, 0.264
*BG_4 - 0.005, 0.276
BG_3 - 0.087, 0.349
BG_2 - 0.646, 1.880
ZFP_24 - 0.000, 0.000
ZFP_22 - 0.001, 0.010
ZFP_20 - 0.001, 0.487
ZFP_18 - 0.001, 0.511
ZFP_16 - 0.002, 0.649
ZFP_14 - 0.006, 0.687
*ZFP_12 - 0.032, 0.659
ZFP_10 - 0.196, 0.571
ZFP_8 - 0.575, 1.673
BG: Q200
BG_2
11
ZFP: Q200
ZFP_8
11
Orig - 0, 0.000
BG_7 - 0.001, 0.000
BG_6 - 0.001, 0.088
BG_5 - 0.001, 0.133
BG_4 - 0.001, 0.092
BG_3 - 0.001, 0.092
*BG_2 - 0.013, 0.174
ZFP_24 - 0.001, 0.001
ZFP_22 - 0.001, 0.030
ZFP_20 - 0.001, 0.373
ZFP_18 - 0.001, 0.364
ZFP_16 - 0.001, 0.368
ZFP_14 - 0.001, 0.372
ZFP_12 - 0.002, 0.396
ZFP_10 - 0.005, 0.445
*ZFP_8 - 0.026, 0.516
Real Information
Bit Position
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30


BG: TAUX
ZFP: TAUX
BG_2
ZFP_8
9
Real Information
Bit Position
Orig - 0, 0.000
BG_7 - 0.000, 0.000
BG_6 - 0.000, 0.024
BG_5 - 0.000, 0.018
BG_4 - 0.000, 0.018
BG_3 - 0.001, 0.025
*BG_2 - 0.002, 0.019
ZFP_24 - 0.000, 0.001
ZFP_22 - 0.000, 0.072
ZFP_20 - 0.000, 0.205
ZFP_18 - 0.001, 0.204
ZFP_16 - 0.001, 0.202
ZFP_14 - 0.001, 0.203
ZFP_12 - 0.003, 0.207
ZFP_10 - 0.006, 0.209
*ZFP_8 - 0.023, 0.237


BG: TS
ZFP: TS
BG_3
ZFP_12
16
Real Information
Bit Position
Orig - 0, 0.000
BG_7 - 0.000, 0.000
BG_6 - 0.001, 0.026
BG_5 - 0.001, 0.028
BG_4 - 0.001, 0.031
*BG_3 - 0.022, 0.086
BG_2 - 0.146, 0.556
ZFP_24 - 0.001, 0.000
ZFP_22 - 0.001, 0.011
ZFP_20 - 0.001, 0.472
ZFP_18 - 0.001, 0.549
ZFP_16 - 0.001, 0.619
ZFP_14 - 0.003, 0.687
*ZFP_12 - 0.011, 0.744
ZFP_10 - 0.092, 0.625
ZFP_8 - 0.068, 0.258


BG: WSPDSRFAV
ZFP: WSPDSRFAV
BG_2
ZFP_8
10
10
Real Information
Bit Position
Orig - 0, 0.000
BG_7 - 0.000, 0.000
BG_6 - 0.000, 0.063
BG_5 - 0.000, 0.083
BG_4 - 0.001, 0.062
BG_3 - 0.001, 0.069
*BG_2 - 0.019, 0.110
ZFP_24 - 0.000, 0.001
ZFP_22 - 0.000, 0.032
ZFP_20 - 0.001, 0.302
ZFP_18 - 0.001, 0.303
ZFP_16 - 0.001, 0.301
ZFP_14 - 0.002, 0.317
ZFP_12 - 0.003, 0.323
ZFP_10 - 0.004, 0.341
*ZFP_8 - 0.021, 0.379
BG: Z500
ZFP: Z500
BG_3
ZFP_12
17
17
Real Information
Bit Position
Orig - 0, 0.000
BG_7 - 0.000, 0.000
BG_6 - 0.000, 0.000
BG_5 - 0.000, 0.001
BG_4 - 0.001, 0.003
*BG_3 - 0.010, 0.029
BG_2 - 0.080, 0.241
ZFP_24 - 0.000, 0.000
ZFP_22 - 0.000, 0.012
ZFP_20 - 0.000, 0.501
ZFP_18 - 0.001, 0.530
ZFP_16 - 0.003, 0.645
ZFP_14 - 0.002, 0.637
*ZFP_12 - 0.010, 0.601
ZFP_10 - 0.196, 0.588
ZFP_8 - 0.253, 0.759

Future directions

How can the cutoff bit recommended by the RIC be translated into a parameter setting such as relative error for a given compression algorithm?

We would like to further consider the question of how we can translate the cutoff bit recommended by the RIC to a parameter setting for a given compression algorithm. We want to more thoroughly explore the connection between the RIC and other quantities of interest such as the DSSIM as described in Chapter \ref{chapTwo}. Previous work in \cite{klower} performs a brute-force approach to matching the RIC bit cutoff with the median error obtained by the ZFP compression algorithm. This is related to the approach shown in \ref{FLNS}, where multiple compression algorithms and settings are applied, and the parameters that lead to median errors near the location of the bit cutoff line are a ``match" for that RIC cutoff. We will explore the parameter settings that align with the RIC and the corresponding levels of compression that can be obtained.

Future directions

What kinds of artifacts are produced by different compressors?

Another topic we want to investigate is to understand what kind of artifacts are produced by different compressors. So far we have looked at the artifacts introduced by ZFP and BitGrooming, but we will also compress data using SPERR \cite{sperr} and SZ \cite{sheng2016}, two other often used transform-based lossy compressors for climate data. We would answer the question of what is causing these artifacts, do they behave in similar ways across compressors and error modes including the precision and absolute error mode in ZFP (of which we only example the precision mode so far), and can adjustments to the compressors be made to remove these spatial artifacts?

Future directions

What are the shortcomings of the RIC, when do they apply, and how can the RIC be improved?

Finally, there are some situations in which the RIC is imperfect. The same numerical values stored in different ways, such as with exponent values that differ from neighboring points, causes alterations in the computed RIC. This will affect data that spans a large range of exponent values in a small spatial area, such as PRECT (precipitation rate) data. Adjustments to the process of computing RIC can be made to account for these shortcomings. Additionally, the RIC as defined in \cite{klower} operates 1-dimensionally, where the dataset is flattened and neighboring bits are considered to be only the bits adjacent in the resulting bit stream. The measure could be extended into two dimensions to more accurately capture the notion of spatial structure.