On information and complexity

Information, complexity, description length, data compression and their application in hydrology

Steven Weijs
Steven.Weijs@ubc.ca
a place of mind
THE UNIVERSITY OF BRITISH COLUMBIA

Storyline

- Def (just 1) .. Why complexity important
- Missing info, info content, randomness, long description, low alg prob.

Take home

- Info content data using infotheory is ill defined if no complexity control
- AIT / description length can determine info content of single objects with referring to underlying distribution
- Info content is always for a question, depends on prior knowledge

Outline

- What / Why complexity?
- Description length and information content

Complexity

No agreed definition of complexity, but in general:

- Many parts
- Many interactions, feedbacks
- Difficult
- Complicated
- Hard to describe

Definition of complexity

- Systems: Nonlin, feedbacks, high-D
- Weaver: organised vs non organised
- Networks: number of links/ nodes
- Computational : resources needed
- Algorithmic: description length

Description length

- Something that is complex needs long description.
- Need a much information transfer to learn about.
- Optimal description length is not arbitrary. Number of short words is limited.
- Of course still language dependent

Language dependence

- NL : Polder
- EN: hydrologically separated area with water levels controlled by pumps
- Description length of comprehensive history of Dutch water management in English:
- L_EN (book) = C* L_NL (book) ????

NO! Additive!

- L_EN (book) = C + L_NL (book)
- C ?
- C = dictionary NL → EN

Data compression

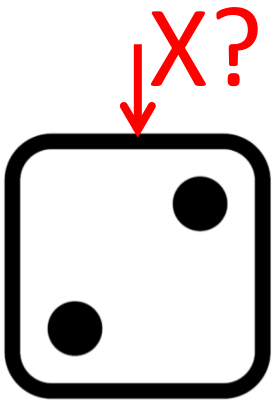
- Relates minimal description length to data compression
- Gives bounds and many analogies

Why is compression relevant

- Practical: harddisk ≠ free
- Theoretical: compression ⇔ information
  - Compression = learning / inference
  - Compression = quality of model/predictions
  - Minimal compressed file size = info content

Info content and prior knowledge

- IT: info content = -log(P_{obs})
- P_{obs} can include prior knowledge
- Info content = relative to prior



$$P(X|y = \text{⚁}) = \left(\frac{1}{5}, 0, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}\right)$$

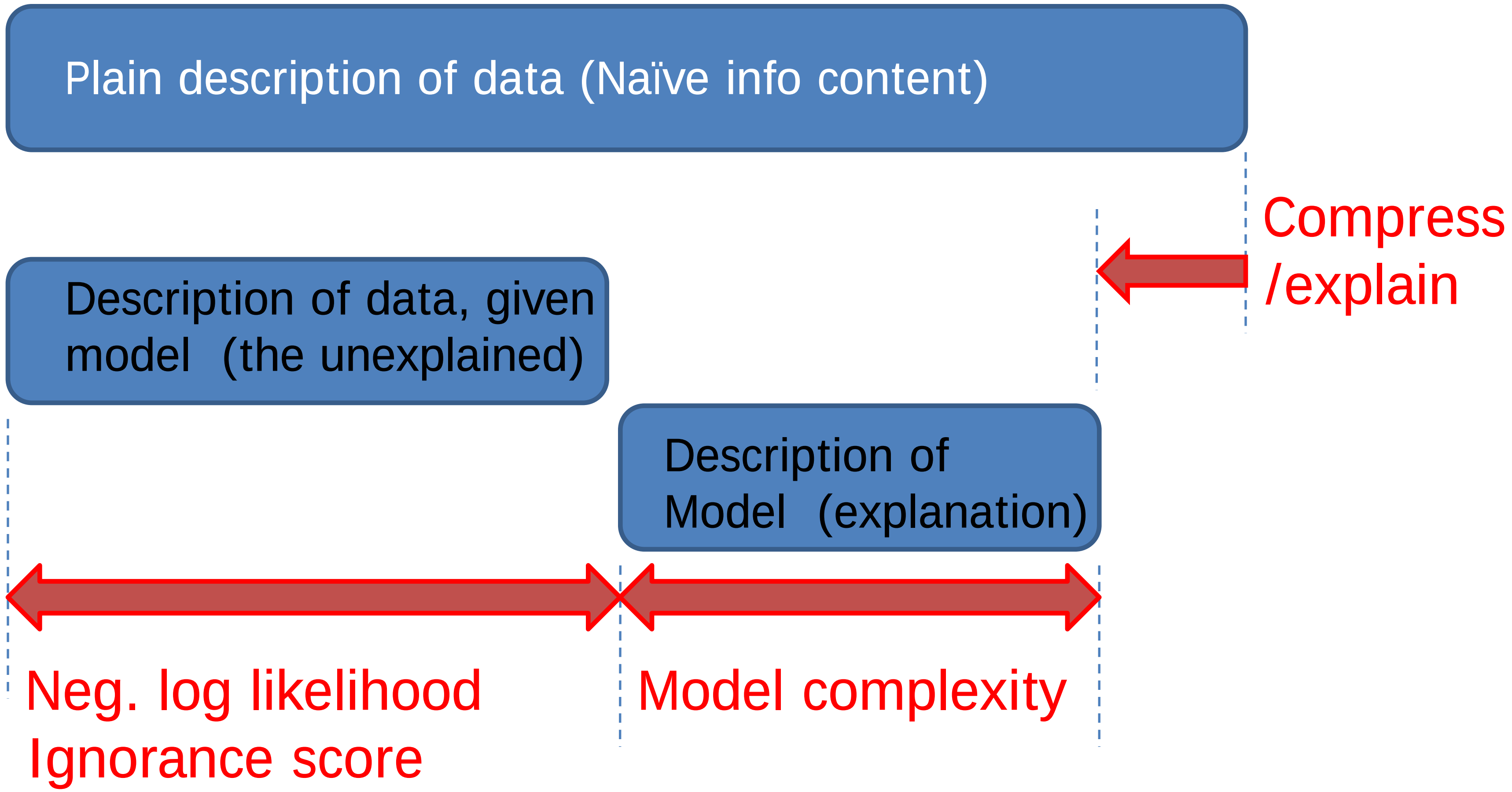
⚁ + ⚁ =7 $\quad P\left(X|y = \text{⚁}\right) = \left(\frac{1}{4}, 0, \frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}\right)$

, ⚁≠⚁ $\quad P\left(X|y = \text{⚁}\right) = \left(0, 0, \frac{1}{2}, \frac{1}{2}, 0, 0\right)$

OBJECTIVES

- Test Zipped size = info content
- Compress with prior knowledge : conditional info content
- Show parallel compression <> inference

Background: compression view



Shannon view description length

signal	
event	freq
CC	0.5
YY	0.25
GG	0.125
RR	0.125

Signal source,
Entropy H=1.75 bits

Signal to describe

YYCCCCRRCCGGYYCC

Shannon view description length

signal		code words	
event	freq	A	B
CC	0.5	00	0
YY	0.25	01	10
GG	0.125	10	110
RR	0.125	11	111

Dictionary

YYCCCCRRCCGGYYCC

0100001000110100 CODE A: 16 bits, 2/color

Naïve code:
2 bits for each color

Shannon view description length

signal		code words	
event	freq	A	B
CC	0.5	00	0
YY	0.25	01	10
GG	0.125	10	110
RR	0.125	11	111

Dictionary

YYCCCCRRCCGGYYCC
0100001000110100
10001110110100

CODE A: 16 bits, 2/color
CODE B: 14 bits, 1.75/color

Compress

Short codes for frequent events: compress to H=1.75 bps

Shannon view description length

signal		code words		
event	freq	A	B	C
CC	0.5	00	0	
YY	0.25	01	10	
GG	0.125	10	110	
RR	0.125	11	111	
YYCCCCRRCCGGYYCC	1			0

Dictionary C ??

YYCCCCRRCCGGYYCC
0100001000110100
10001110110100
0

Compress??

CODE A: 16 bits, 2/color
CODE B: 14 bits, 1.75/color
CODE C: 1 bit , .125/color

Compression, but need dictionary to unzip
Dictionary is as long as original → no compression

Shannon view description length

signal		code words		
event	freq	A	B	C
CC	0.5	00	0	
YY	0.25	01	10	
GG	0.125	10	110	
RR	0.125	11	111	
YYCCCCRRCCGGYYCC	1			0

Dictionary C ??

YYCCCCRRCCGGYYCC

0100001000110100

10001110110100

0

Compress??

```
CODE A: 16 bits, 2/color
CODE B: 14 bits, 1.75/color
CODE C:  1 bit ,  .125/color
```

Unless dictionary is prior knowledge for receiver!

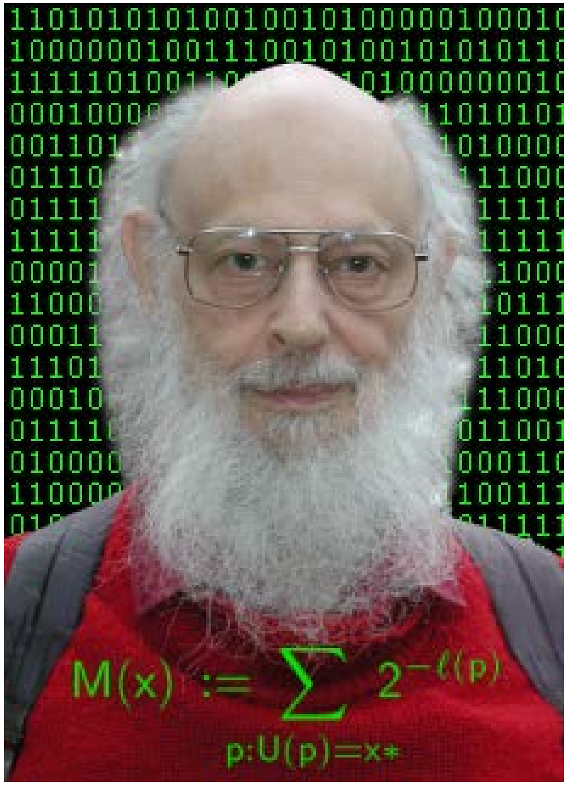
Deeper notion of description length

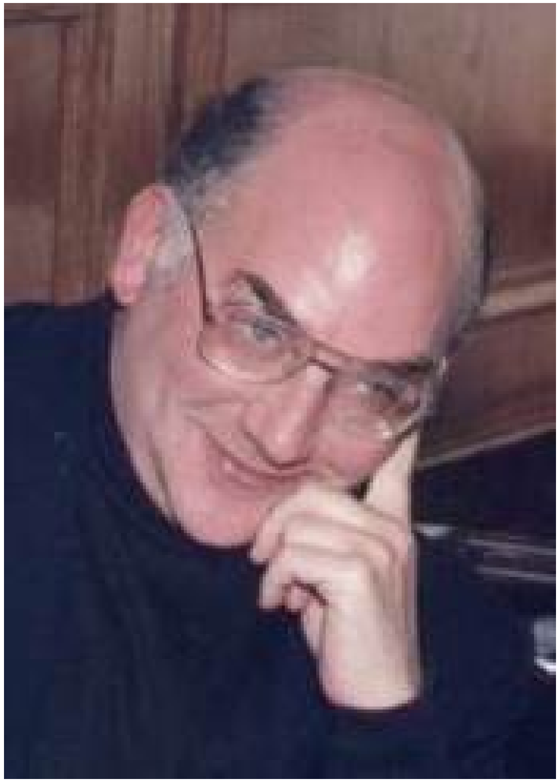
- Description length depends on language
- Not just "nouns", but add grammar
- Allows more efficient descriptions (recursive)

- Computer program=Full language of math (Church-Turing thesis)

Algorithmic Information Theory

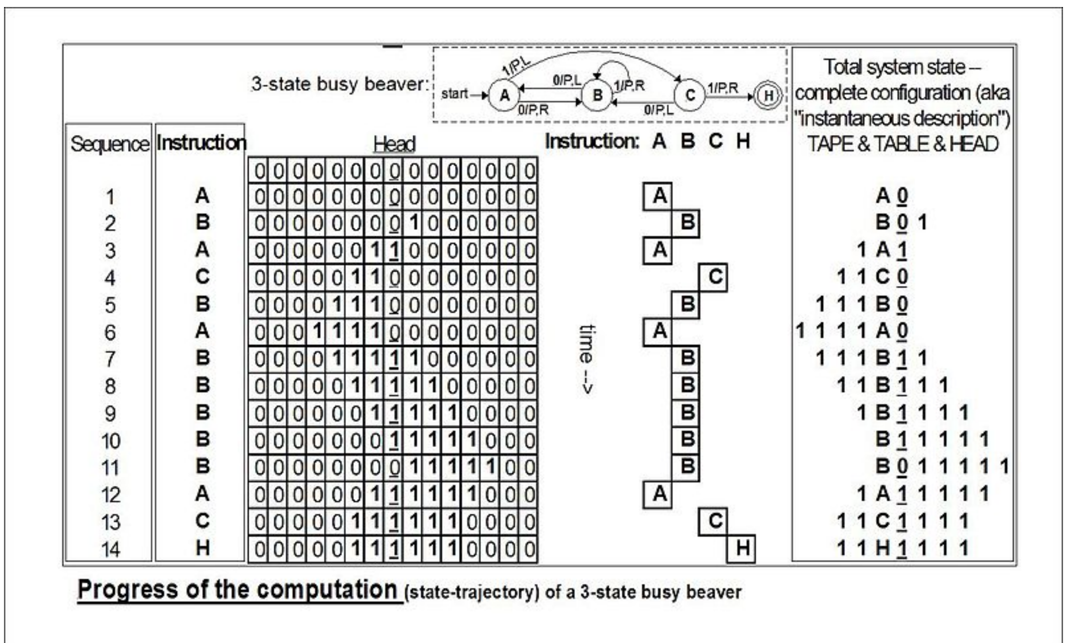
independently developed by Kolmogorov(1968), Solomonoff (1964) and Chaitin (1966)

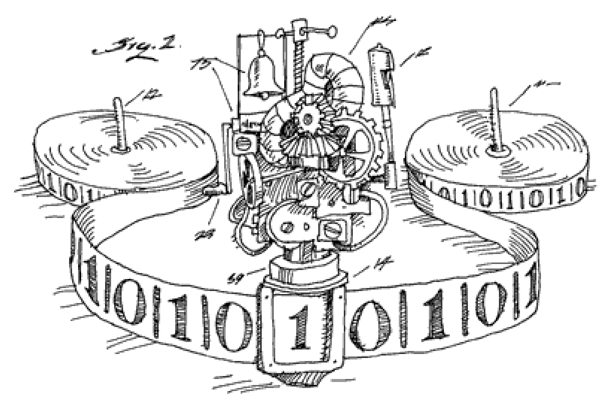




Key concepts of AIT

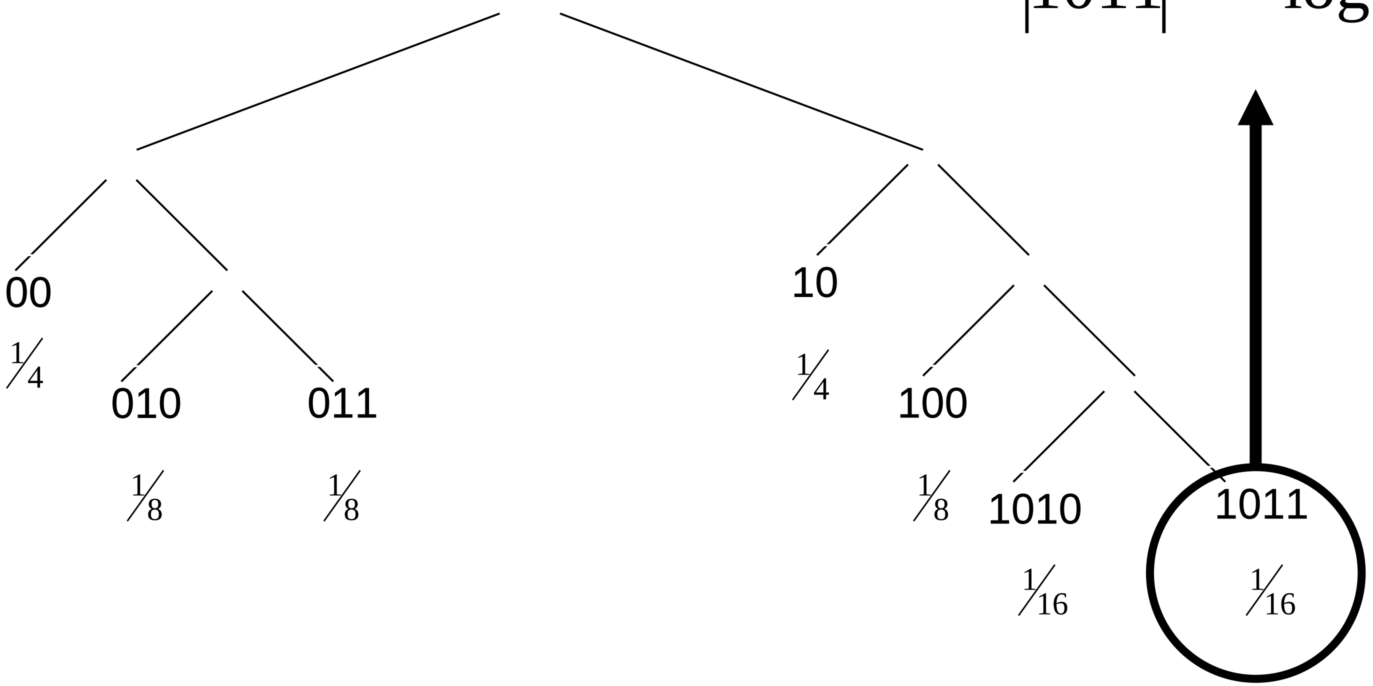
- theories are programs for universal computer
- randomness = absence of patterns
- structure enables short descriptions





Correspondence code-length <> probability

$$|1011| = -\log(1/16)$$

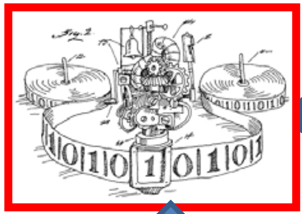


Consequences

- short description ~ higher probability
- shorter program → a priori more likely theory
- Sum over all programs M_i of $2^{-|M_i|}$ is 1
- Universal prior over computable functions
- Use as natural complexity penalization

AIT view

Output: data; (X)



Program for universal computer

Compress
explain
learn

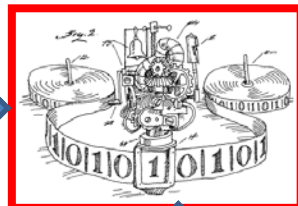
Remove redundancy
Find patterns

Program length; K(X)
K(X) kolmogorov complexity is analog to H(X)
Algorithmic entropy

AIT view

Output: data; (X)

Prior info (Y)



K(Y)

Compress /explain

Program for universal computer

$K(XY) = K(X|Y) + K(Y)$

$K(XY) \leq K(X) + K(Y)$

Program length; $K(X|Y)$

Info content prior knowledge: $K(X)-K(X|Y)$

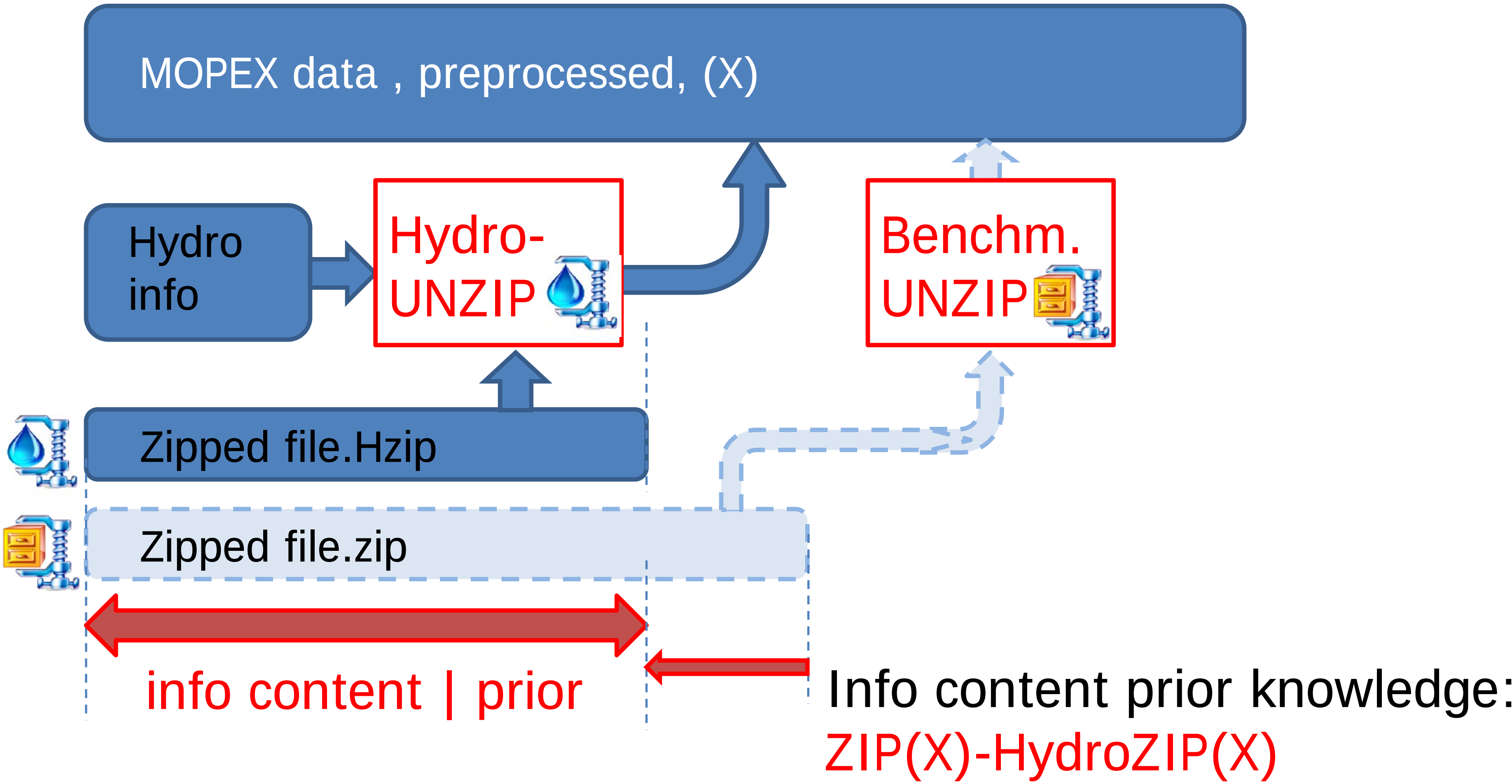
Example application in Hydrology

HYDROZIP!

HydroZIP

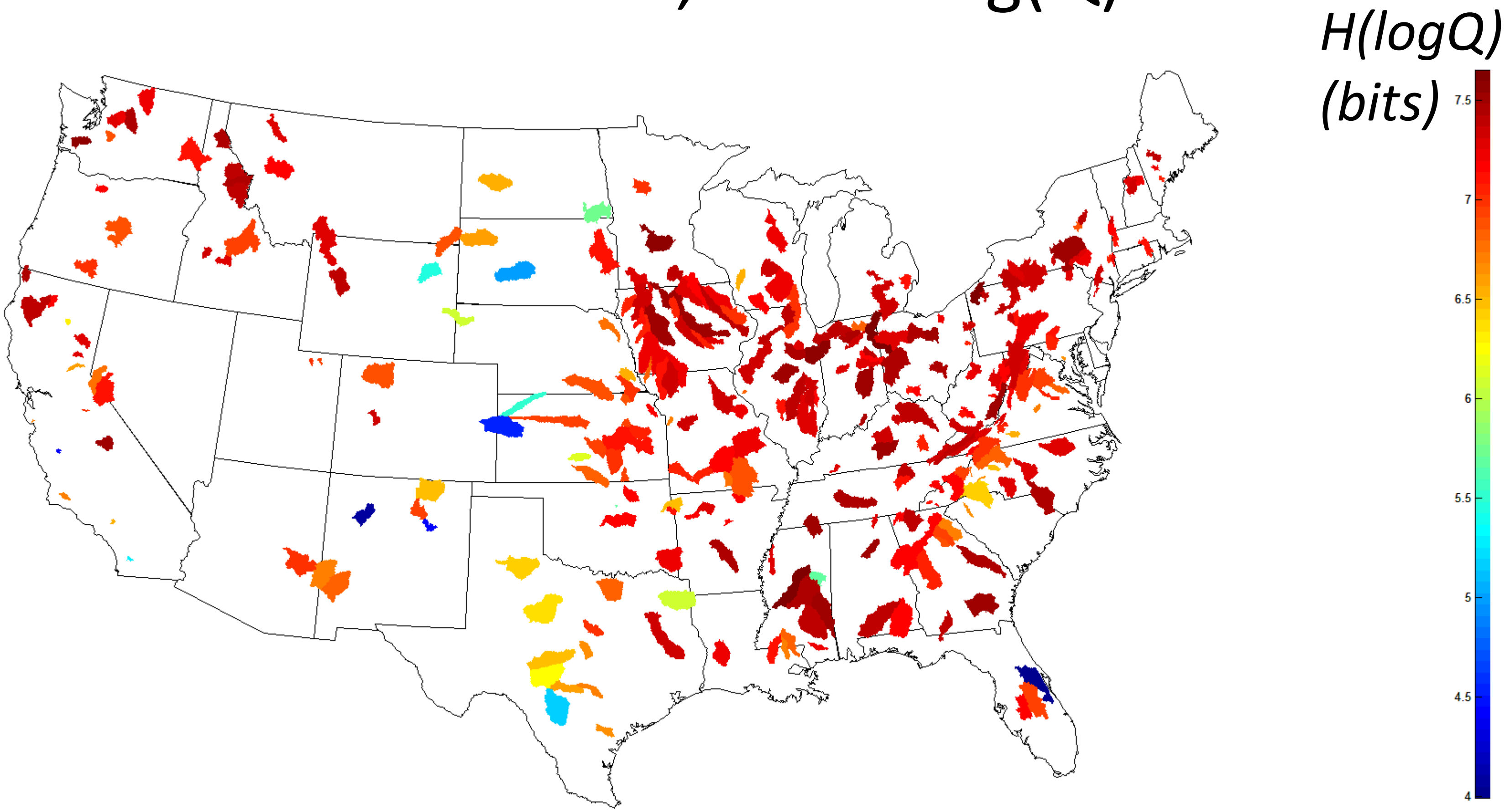
- Coding based on probability distribution:
  - Arithmetic coding
  - Huffman coding
- Use parametric distributions, not histogram
- Use temporal dependencies:
  - RLE on zero to exploit dry spells
  - Take differences to exploit autocorrelation

Approximate K(X) and K(X|Y) by ZIP

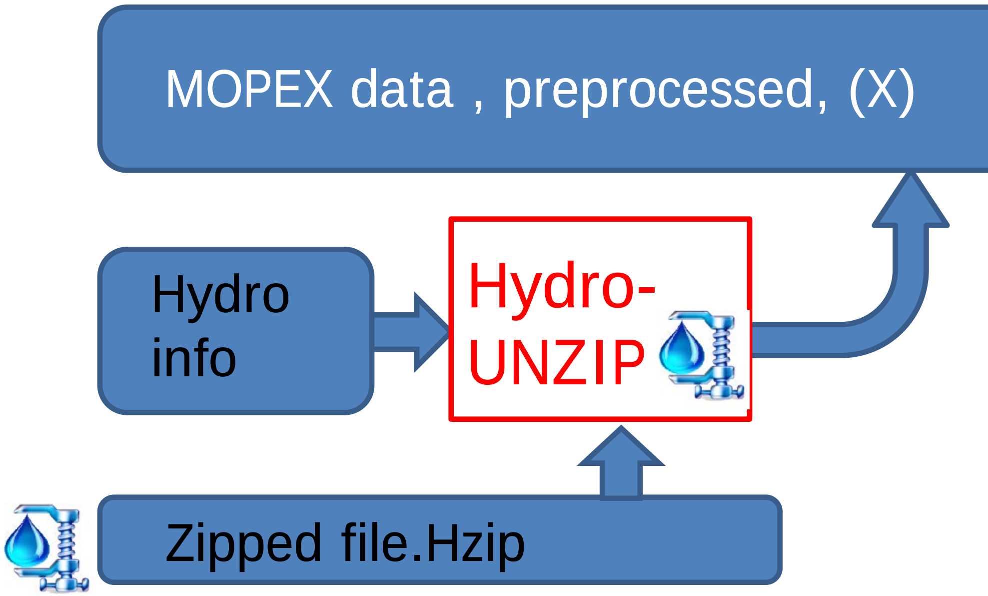


Compression experiment: Data

- MOPEX 431 basins; P and log(Q)

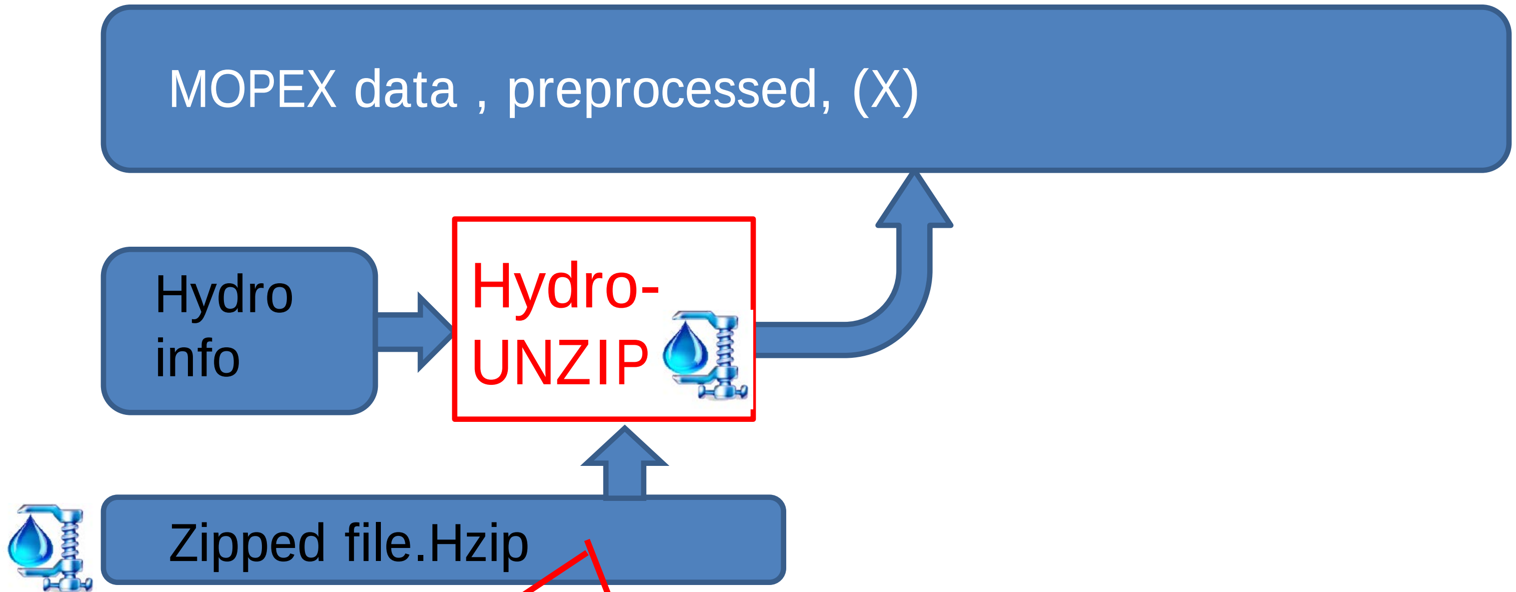


Methods: 1) make Hydro(UN)ZIP



Prior hydro knowledge:
- Parametric distr.
- Autocorrelation
- Dry spells

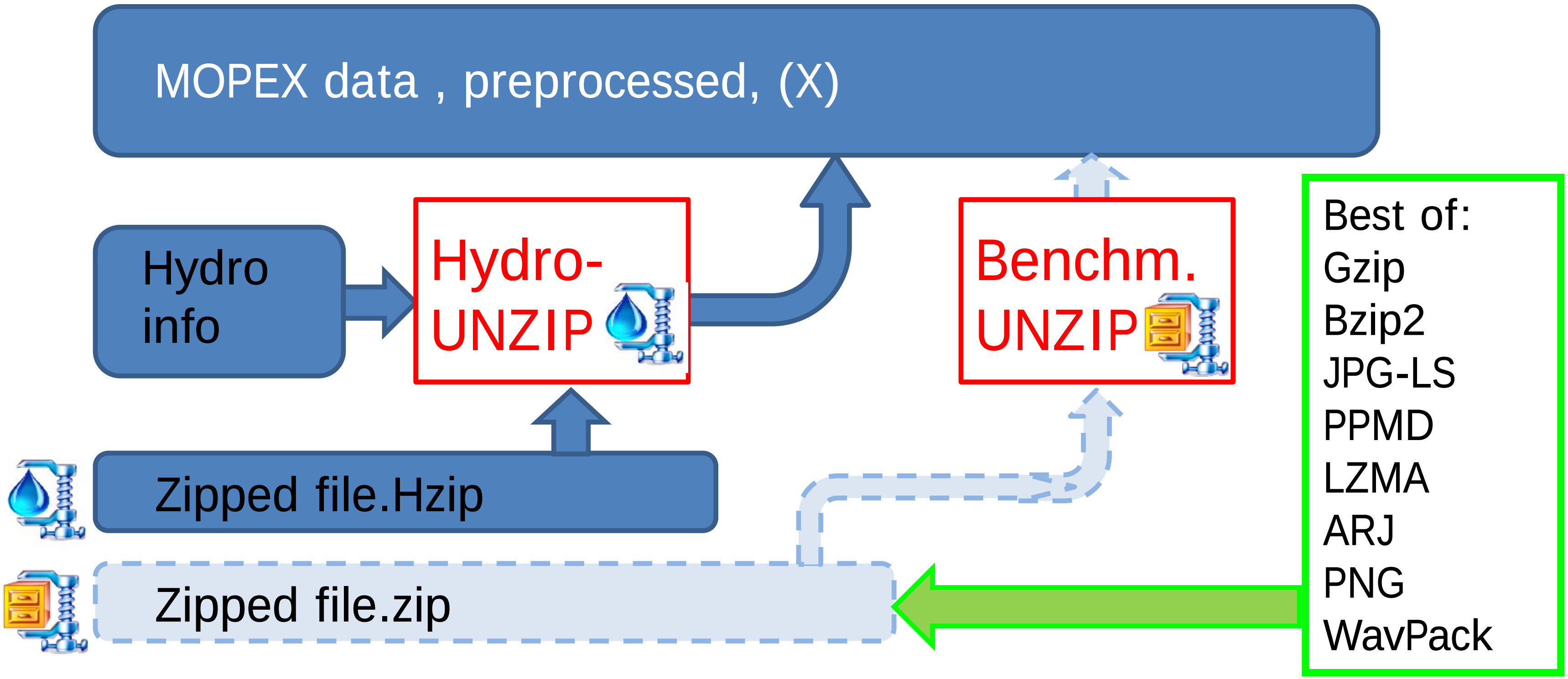
Methods: 2) look at patterns



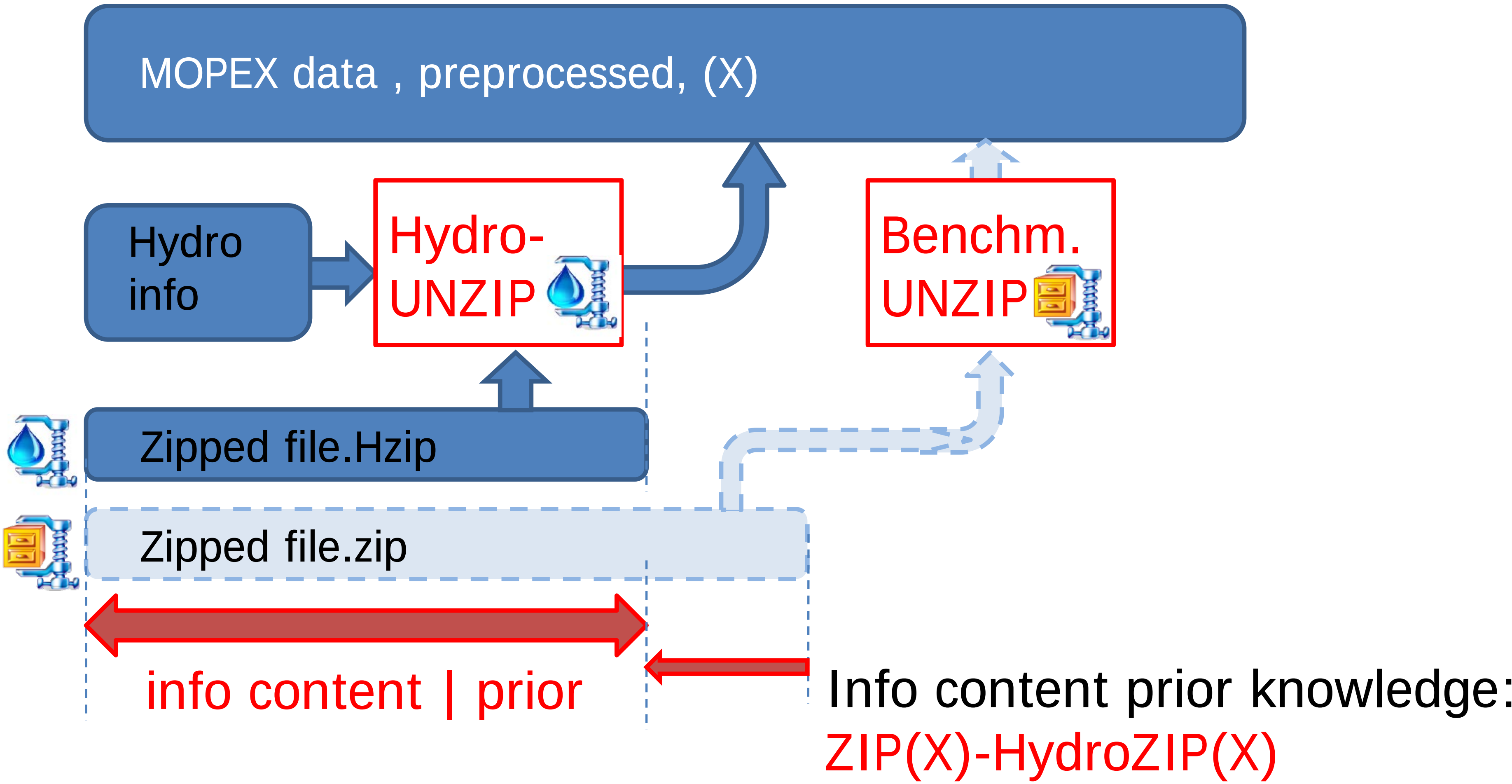
Dependencies
- RLE on 0 works?
- Differencing?

Distributions
- (Log-)normal
- Skew-Laplace
- Exponential + P(0)
- 2 par. Gamma + P(0)

Methods: 3) do benchmark



Methods: 4) compare



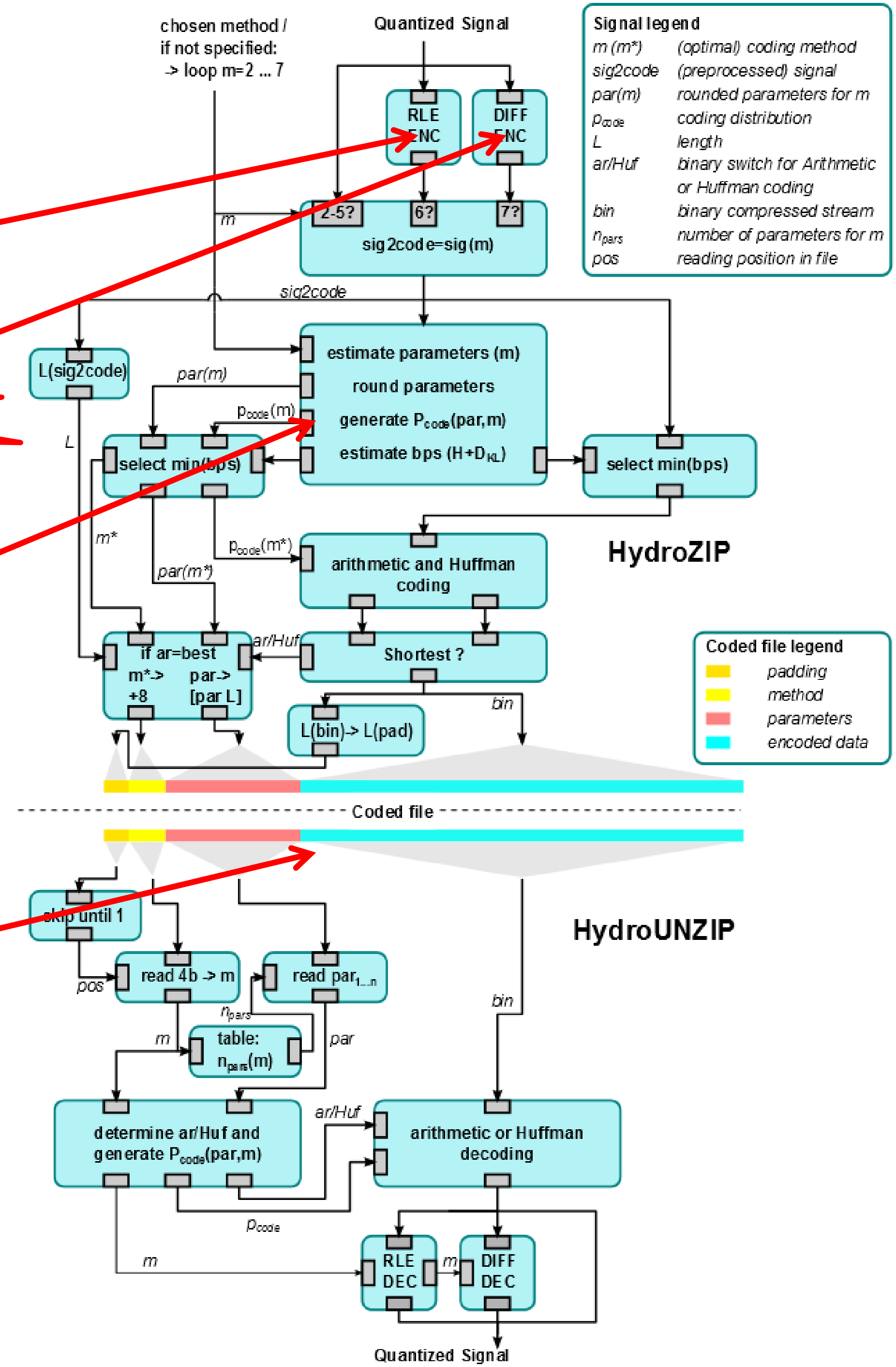
Hydro(UN)ZIP:

RLE on dry spells

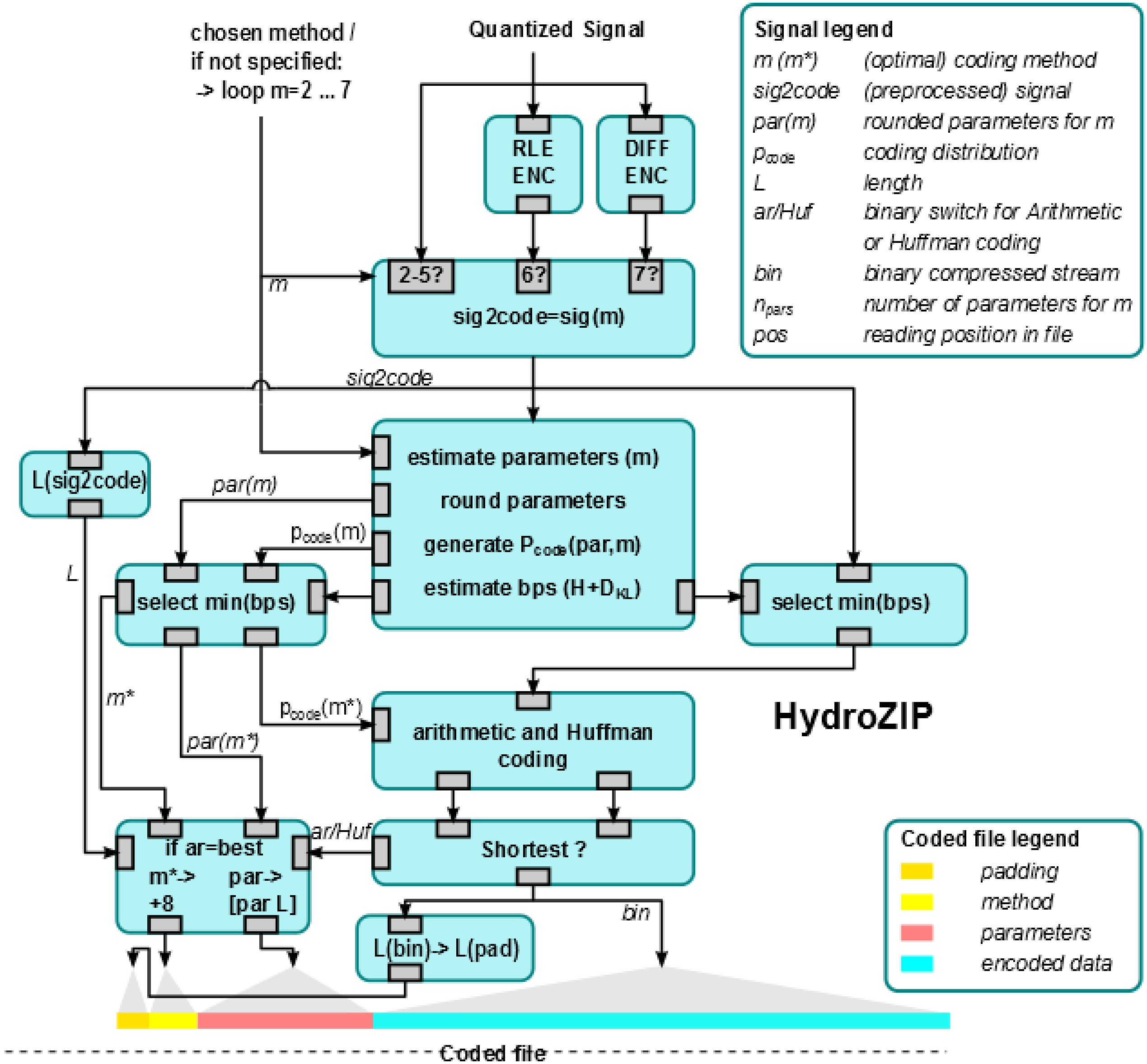
Take differences

Try parametric distributions

File is complete description

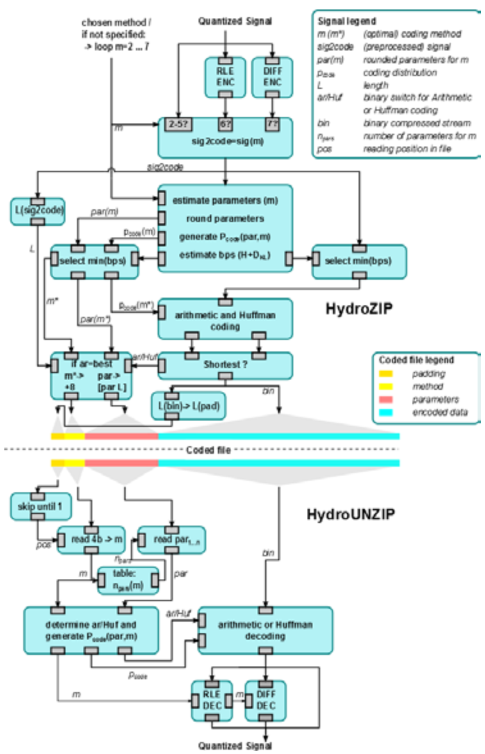


coding

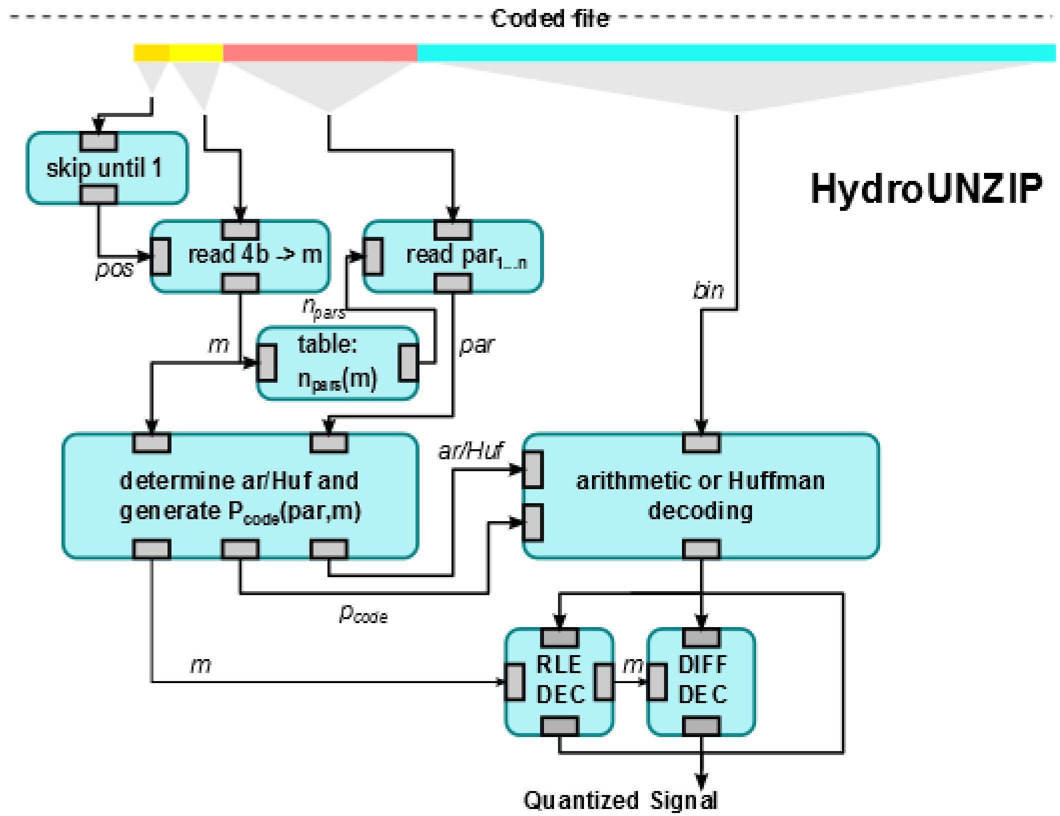


chosen method / if not specified: -> loop m=2 ... 7
Quantized Signal
RLE ENC
DIFF ENC
2-5?
6?
7?
m
sig2code=sig(m)
sig2code
L(sig2code)
L
estimate parameters (m)
round parameters
generate Pcode(par,m)
estimate bps (H+DKL)
par(m)
Pcode(m)
select min(bps)
select min(bps)
m*
Pcode(m*)
par(m*)
arithmetic and Huffman coding
HydroZIP
if ar=best
m*-> +8
par-> [par L]
ar/Huf
Shortest ?
L(bin)-> L(pad)
bin
Coded file
Signal legend
m (m*) (optimal) coding method
sig2code (preprocessed) signal
par(m) rounded parameters for m
pcode coding distribution
L length
ar/Huf binary switch for Arithmetic or Huffman coding
bin binary compressed stream
npars number of parameters for m
pos reading position in file
Coded file legend
padding
method
parameters
encoded data

decoding



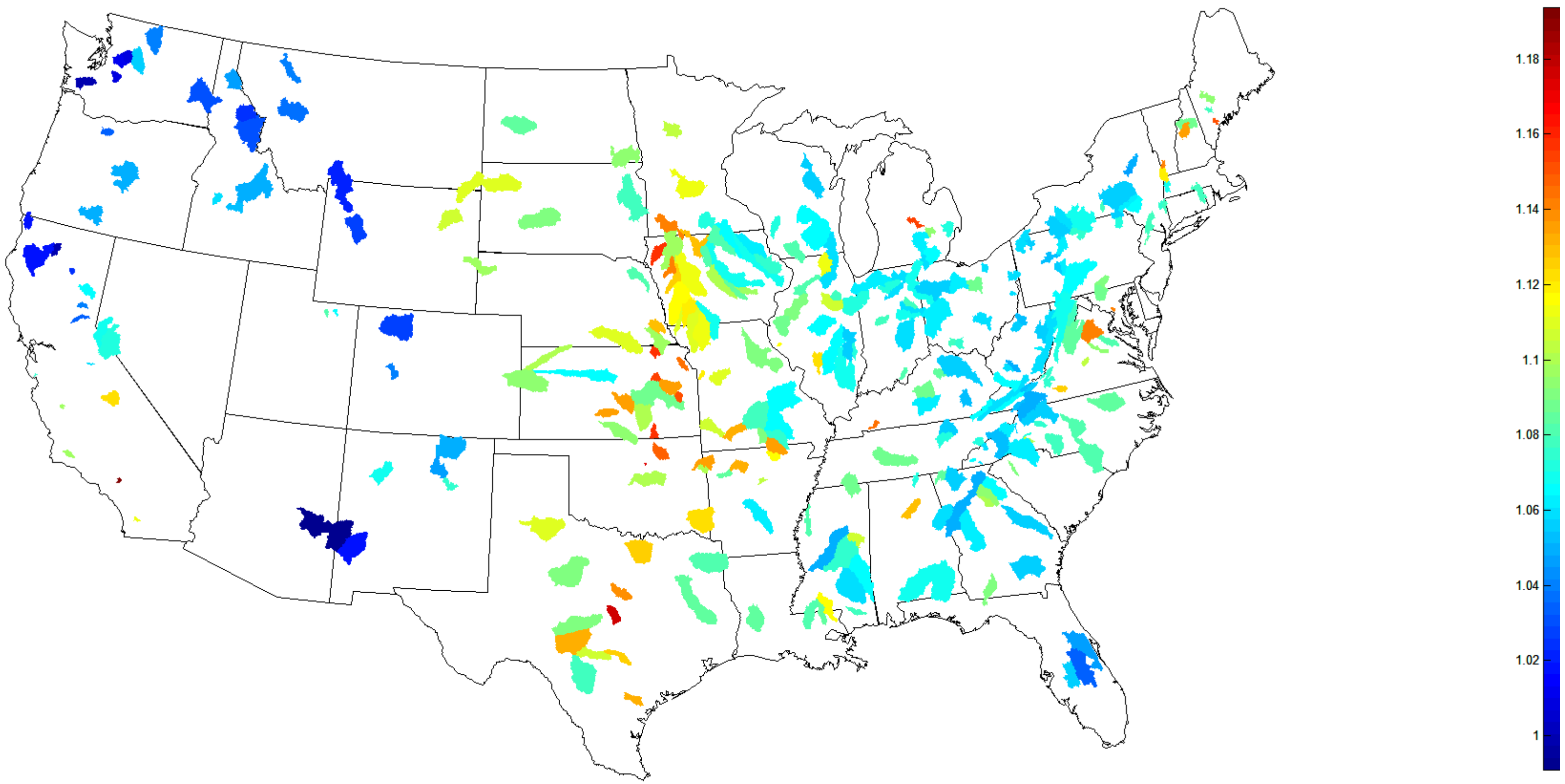
Quantized Signal
Signal legend
HydroZIP
Coded file legend
padding
method
parameters
encoded data
Coded file
HydroUNZIP
Quantized Signal



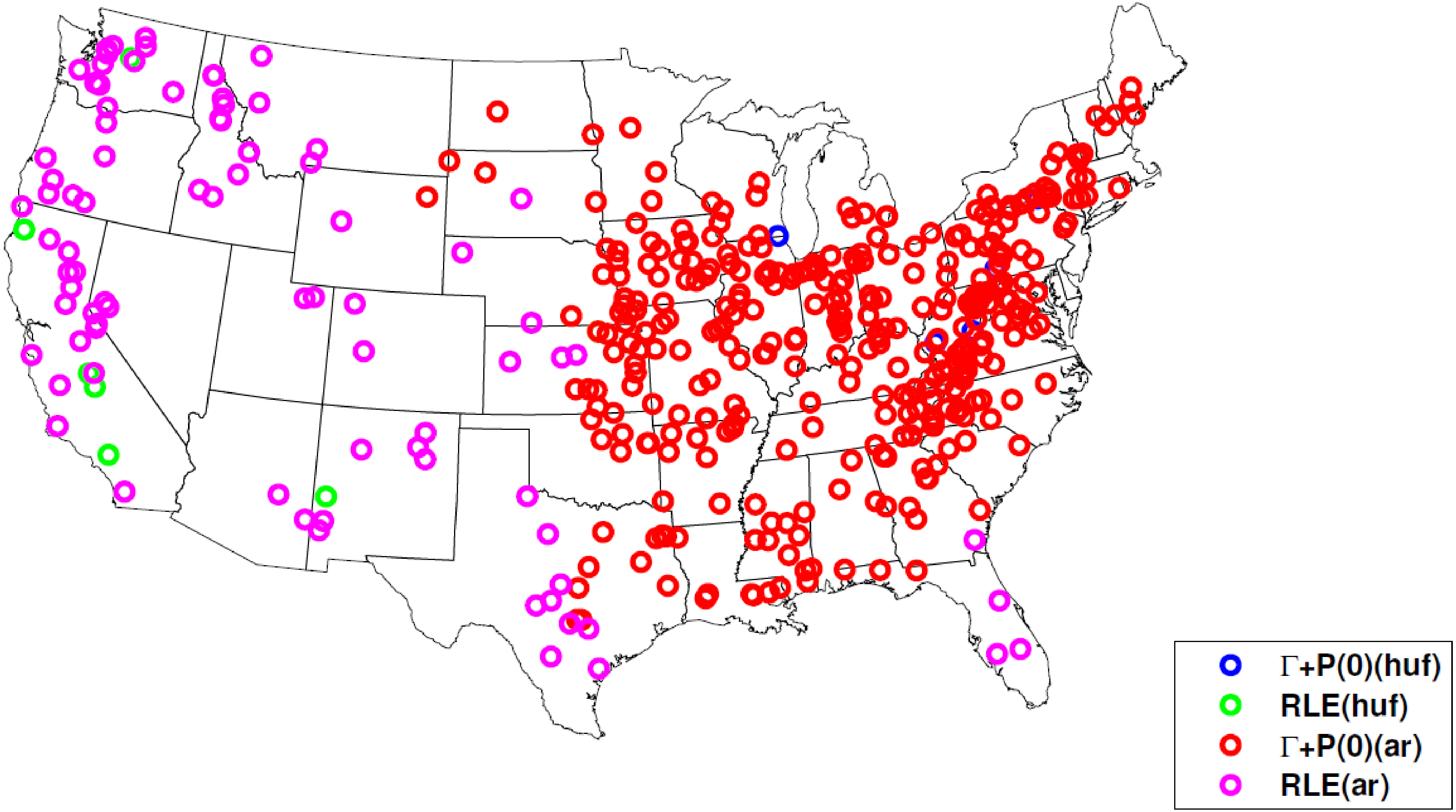
Coded file
skip until 1
pos
read 4b -> m
read par$_{1...n}$
HydroUNZIP
bin
n_{pars}
table: n_{pars}(m)
m
par
determine ar/Huf and generate P_{code}(par,m)
ar/Huf
arithmetic or Huffman decoding
p_{code}
m
RLE DEC
m
DIFF DEC
Quantized Signal

Some results

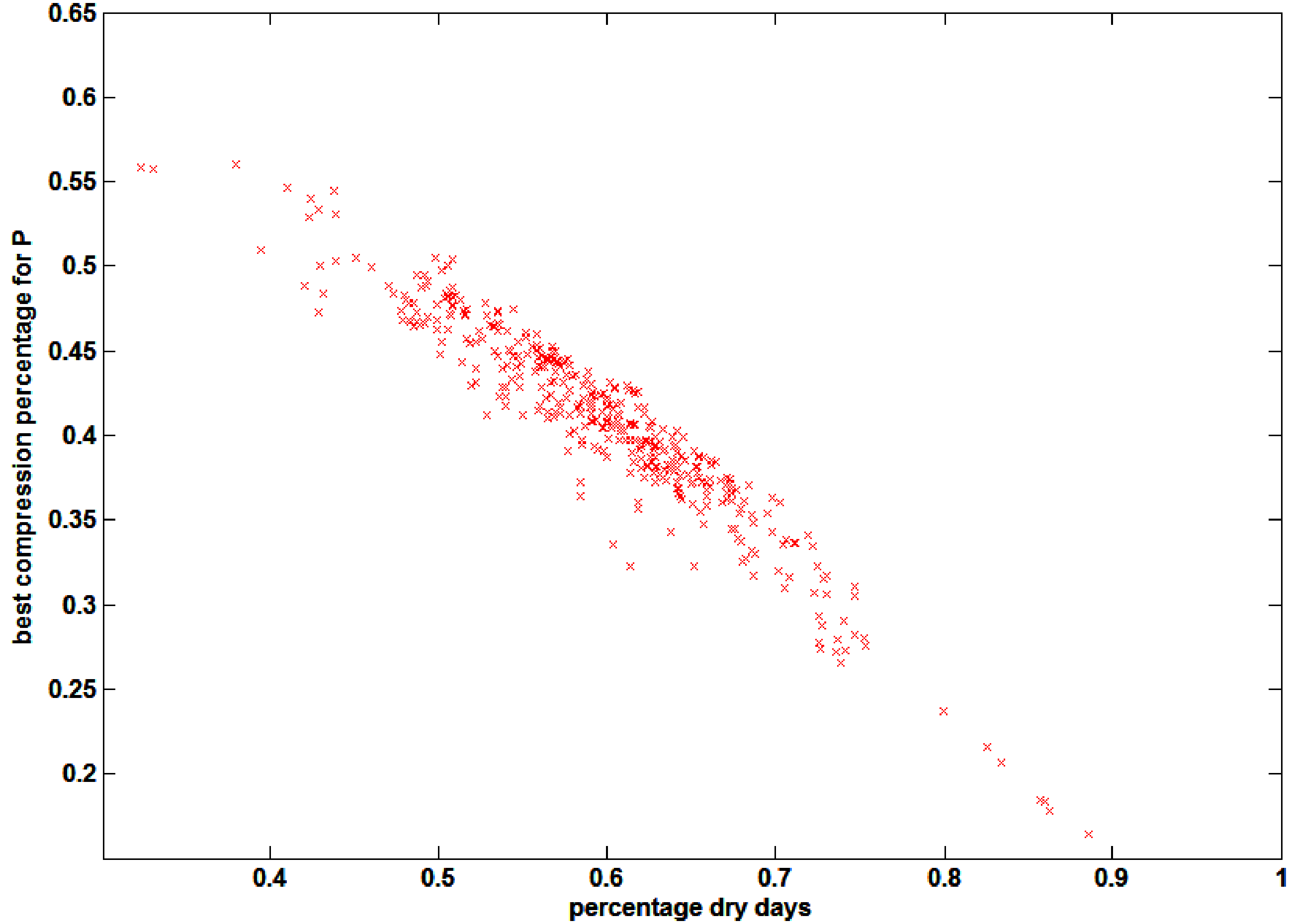
Compression/entropy for P



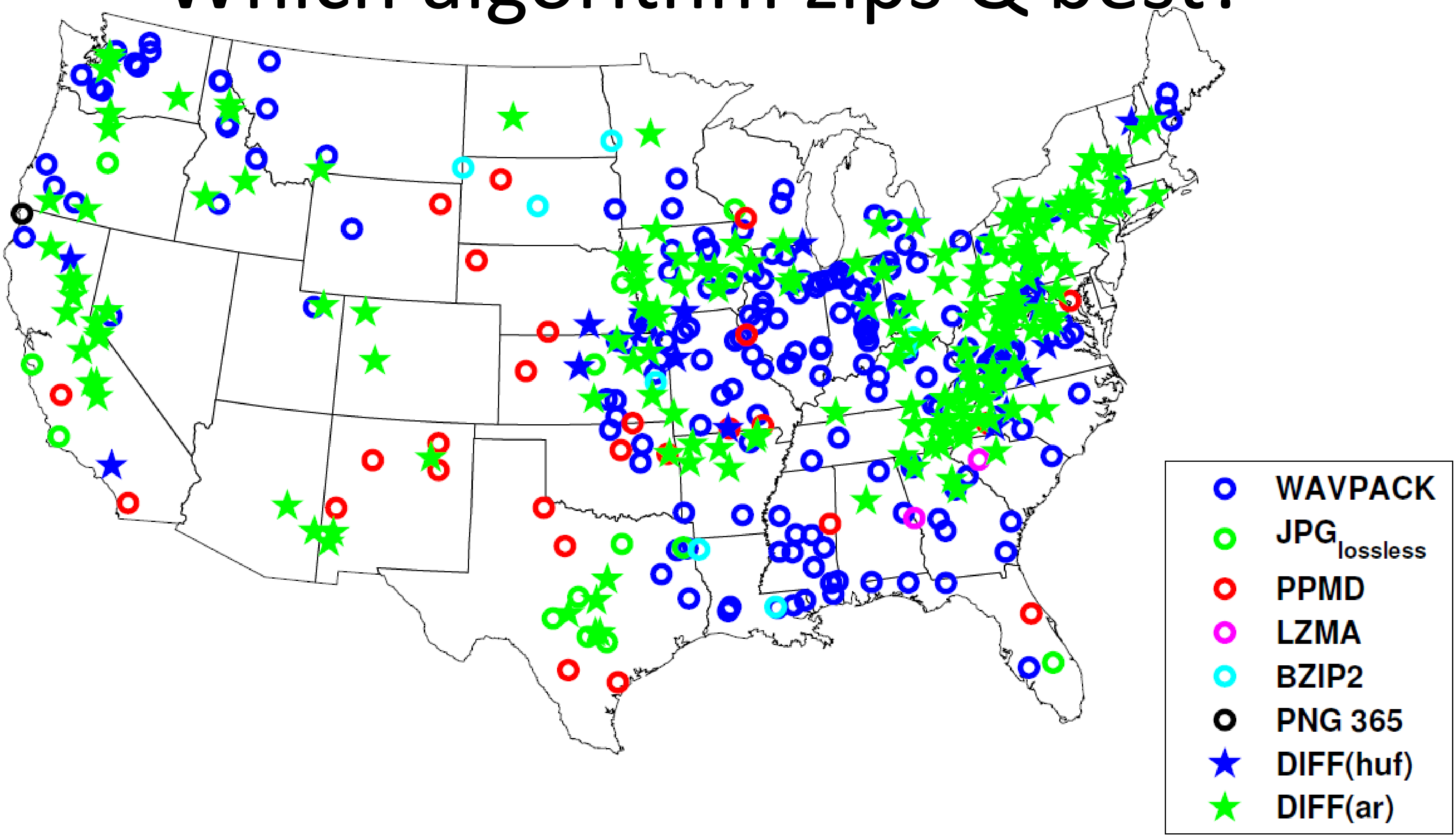
Which algorithm zips P best?

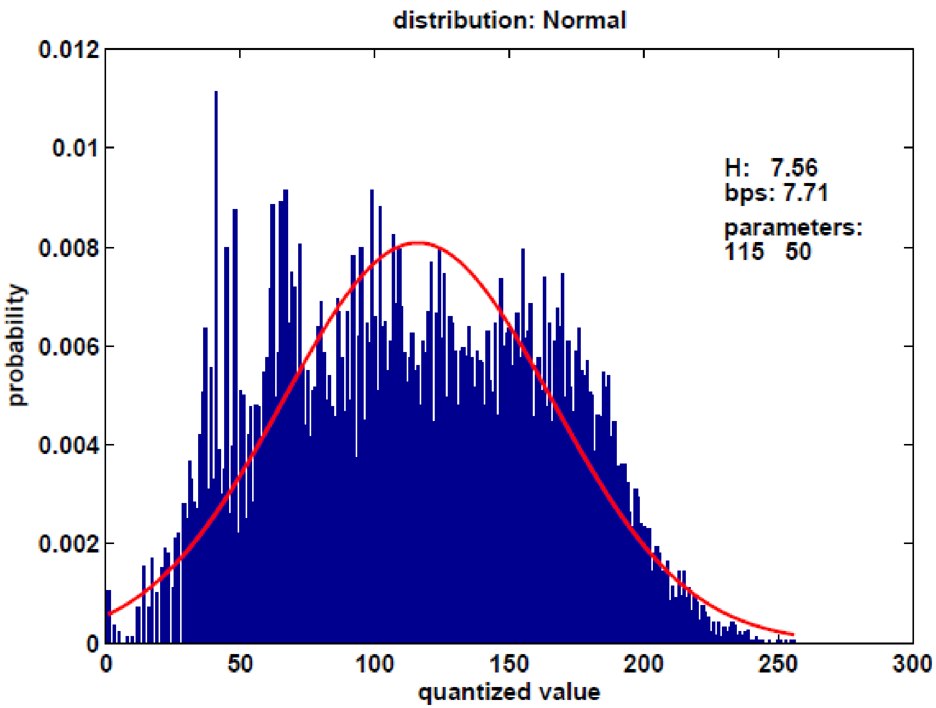


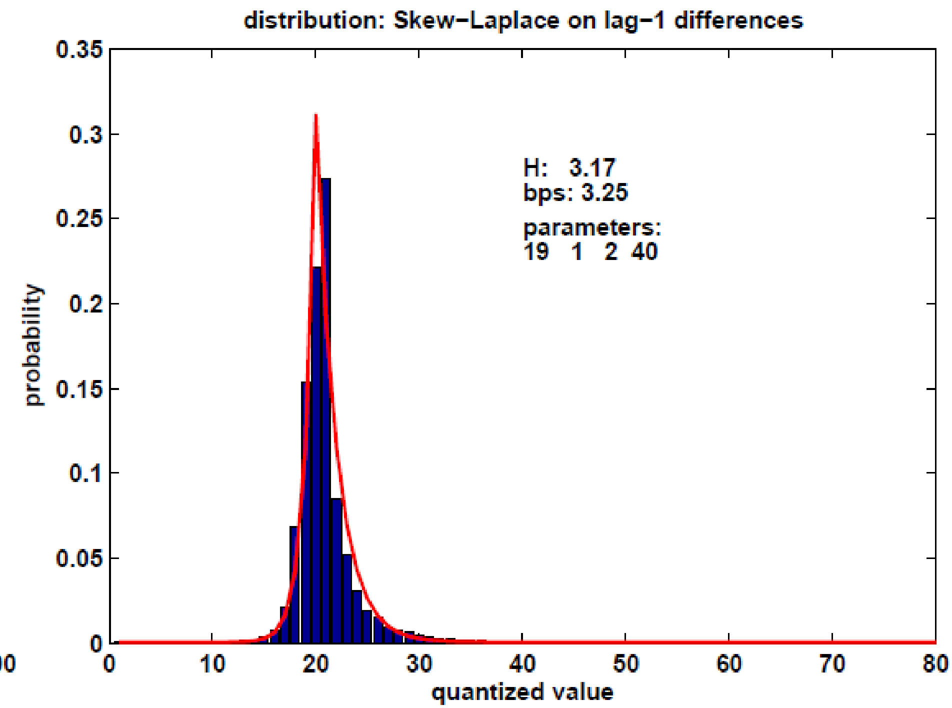
Rainfall compressibility



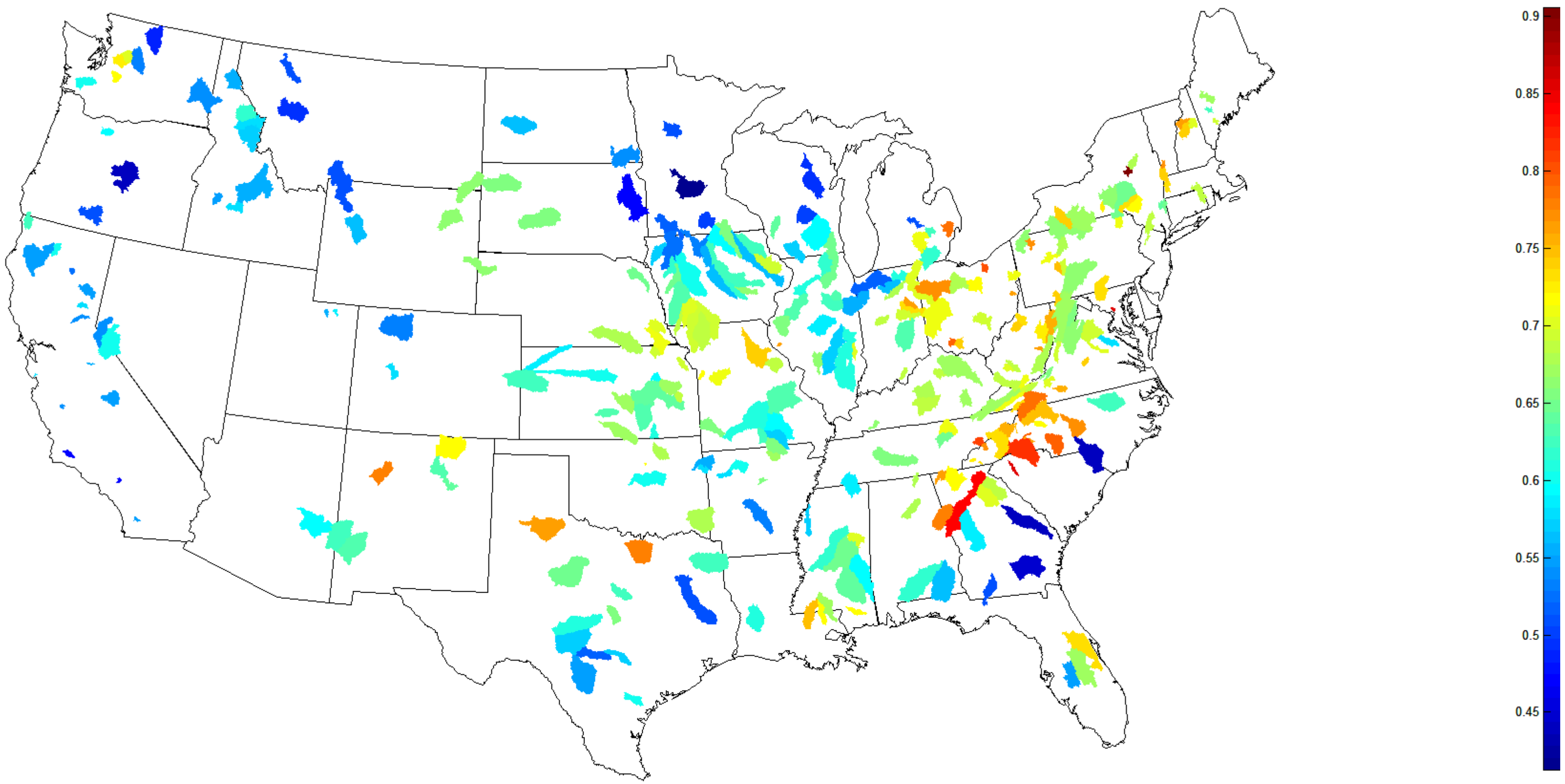
Which algorithm zips Q best?



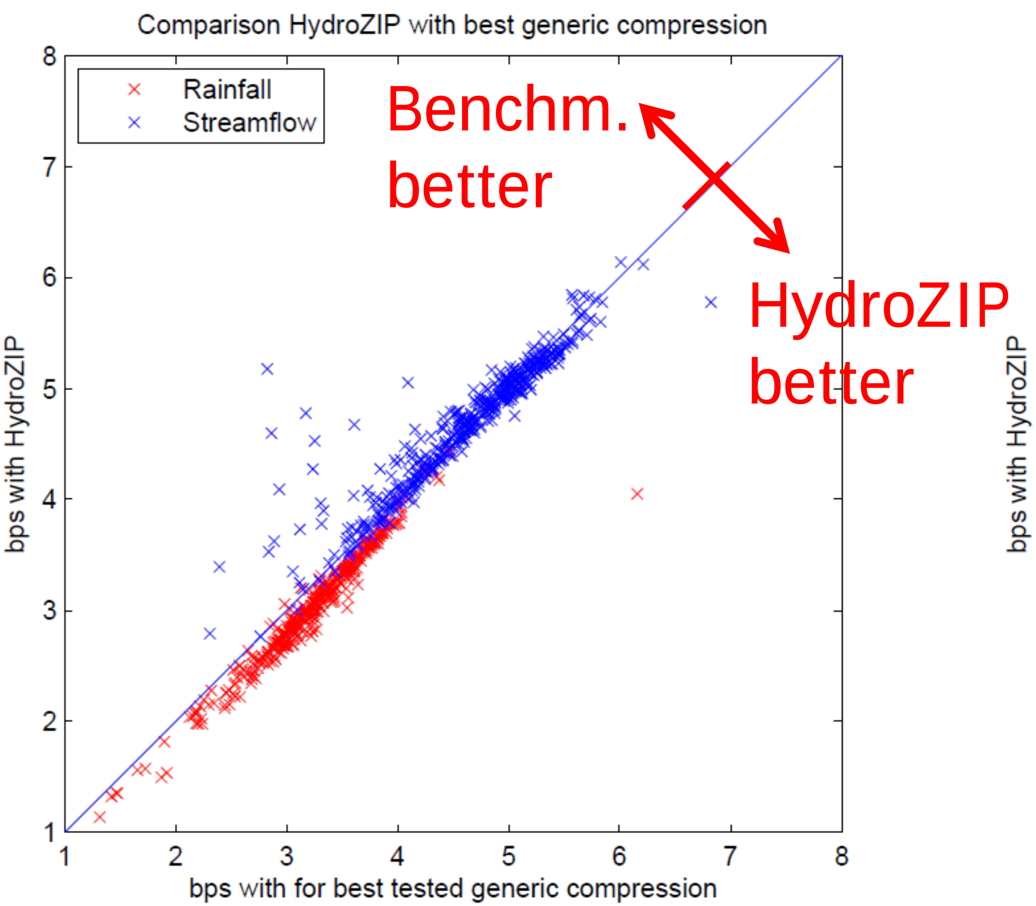

distribution: Normal
0.012
0.01
0.008
0.006
0.004
0.002
0
probability
H: 7.56
bps: 7.71
parameters:
115 50
0
50
100
150
200
250
300
quantized value

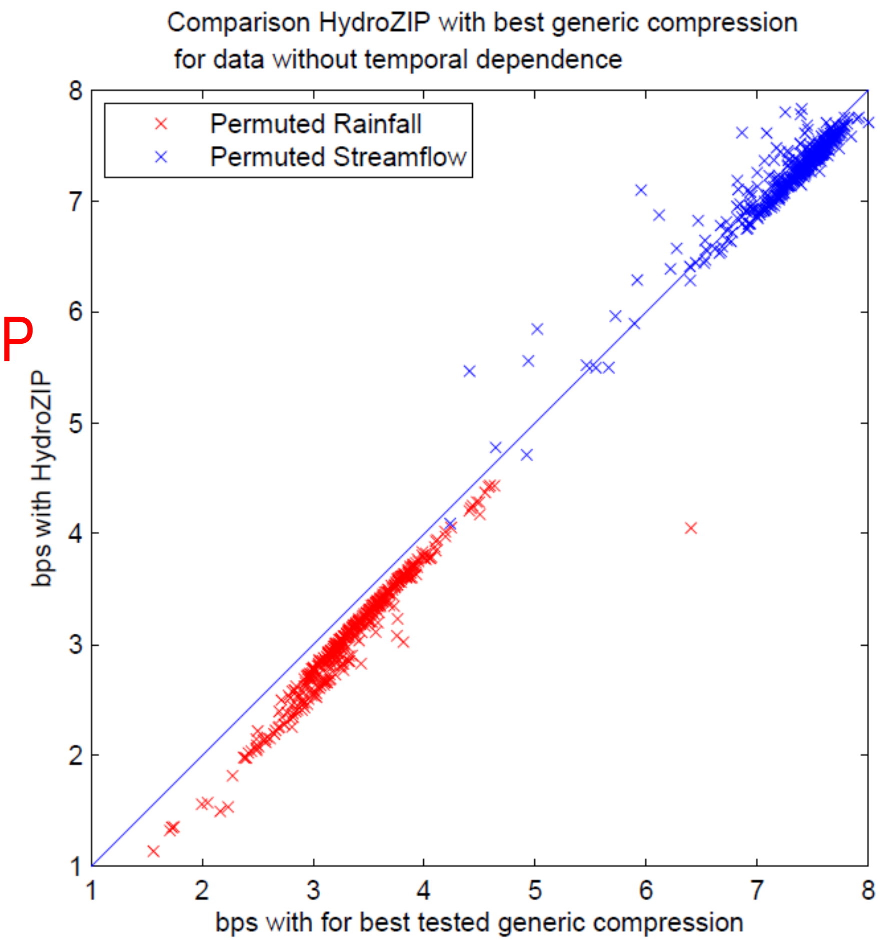

distribution: Skew−Laplace on lag−1 differences
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
probability
H: 3.17
bps: 3.25
parameters:
19 1 2 40
0
10
20
30
40
50
60
70
80
quantized value

temporal compression Q



HydroZIP often beats benchmark !





Conditional Kolmogorov Complexity

- Estimated by HydroZIPped size
- Takes into account all/some dependencies
- Useful to estimate info content|prior
- Posterior-complexity penalized likelihood

Model complexity

- Is integral part of info content
- Should be accounted for in model selection
- Unless model is prior knowledge
- AIT / compression naturally includes this

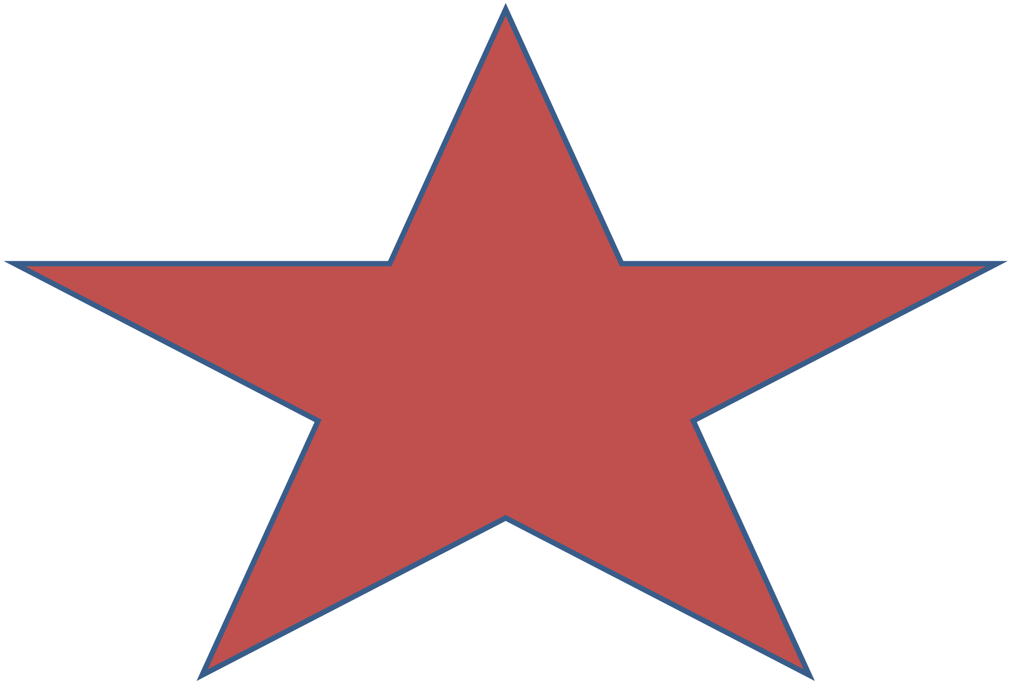
Prior knowledge

- Is free model complexity
- Helps compression
- Influences info content of data

Conclusions case study

- Description length /info content =f(prior knowledge)
- HydroZIP < ZIP, demonstrates this
- Model inference ~ compression

ESTIMATING ENTROPY OF DATA

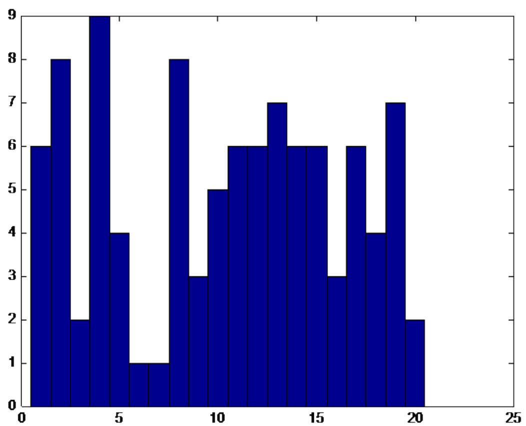


AIT perspective on estimating entropy of data

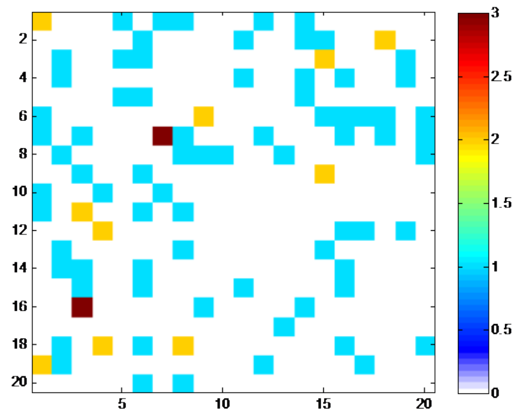
- Entropy is a measure of a probability distribution not a time series
- It measures uncertainty of a state of mind
- Its calculation always defines a question and adds prior knowledge.
- Be careful in multi-dimensional cases!

The Curse of dimensionality

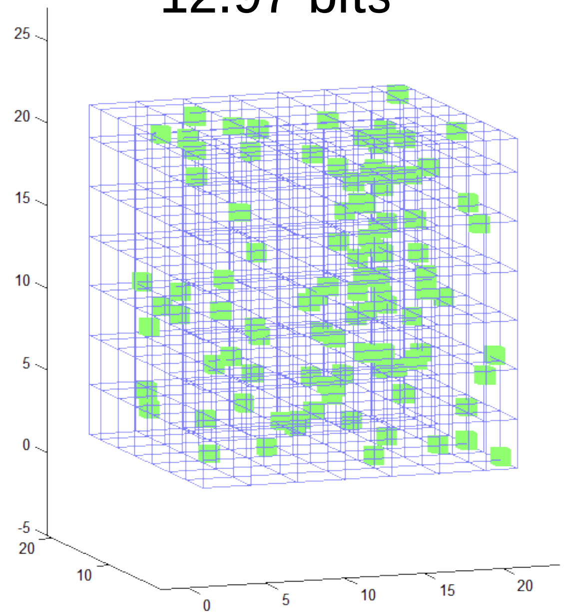
H(X)=
4.18 bits
4.32 bits



H(X,Y)=
6.35 bits
8.64 bits



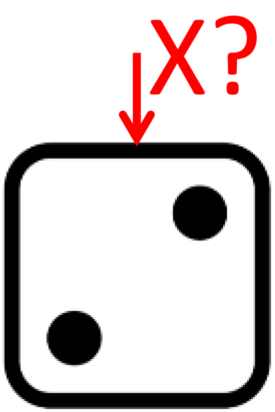
H(X,Y,Z)=
6.64 bits
12.97 bits

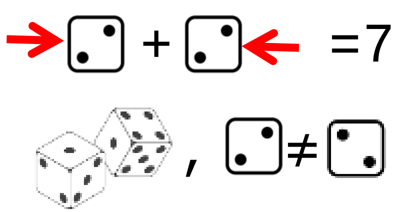


DATA PROCESSING INEQUALITY

Info content and prior knowledge

- What if only one observation?
- Or PUB ?





$$P(X|y = \text{⚁}) = \left(\frac{1}{5}, 0, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}\right)$$

$$P\left(X|y = \text{⚁}\right) = \left(\frac{1}{4}, 0, \frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}\right)$$

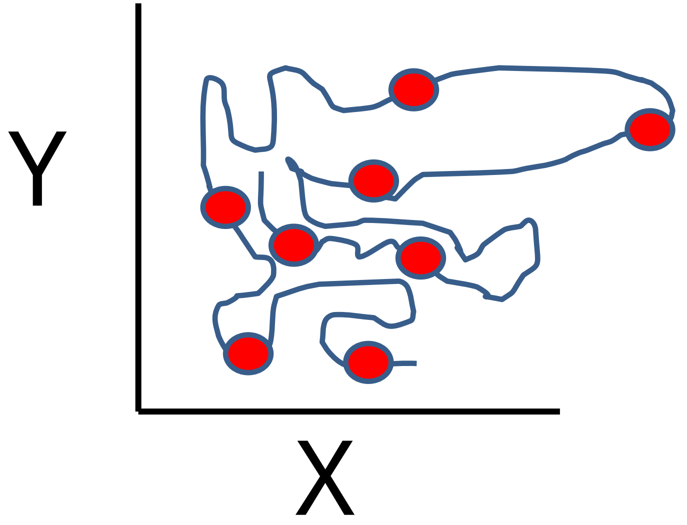
$$P\left(X|y = \text{⚁}\right) = \left(0, 0, \frac{1}{2}, \frac{1}{2}, 0, 0\right)$$

What we learn from die example

- Adding observed variable helps
- model adds (on average) true information to predictions, if true “physics” added
- importance fades with more data
- But was information already there in predictor?
- $H(Y|X,\text{model}) < H(Y|X)$??

H(Y|X) ill defined for data

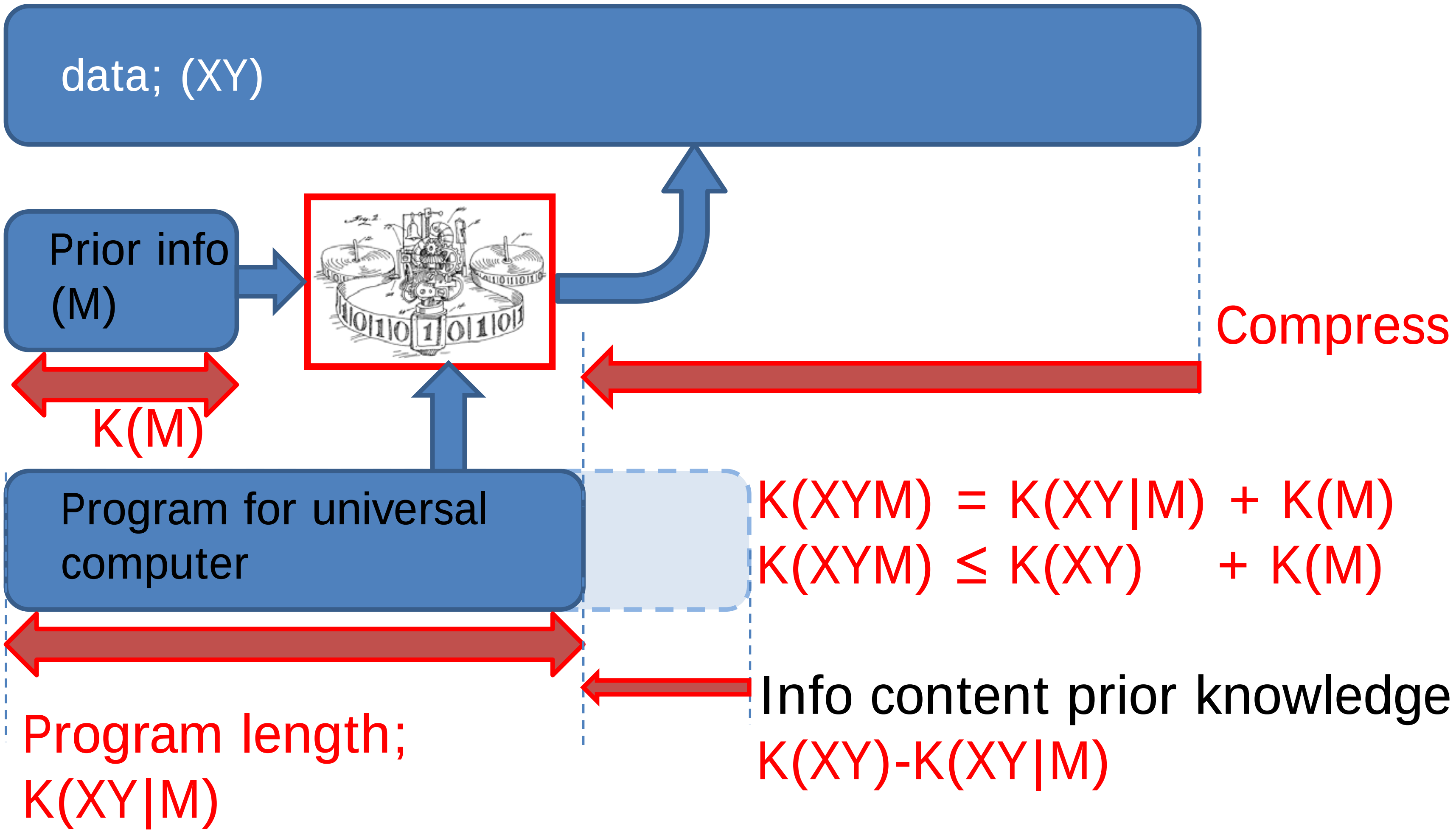
- Not defined for data, but for distribution
- Loads of data needed
- Or involves model
- Can model be arbitrarily complex?



Kolmogorov complexity

- K(X)
- K(Y)
- K(Y*)
- K(Model)
- K(DEM)

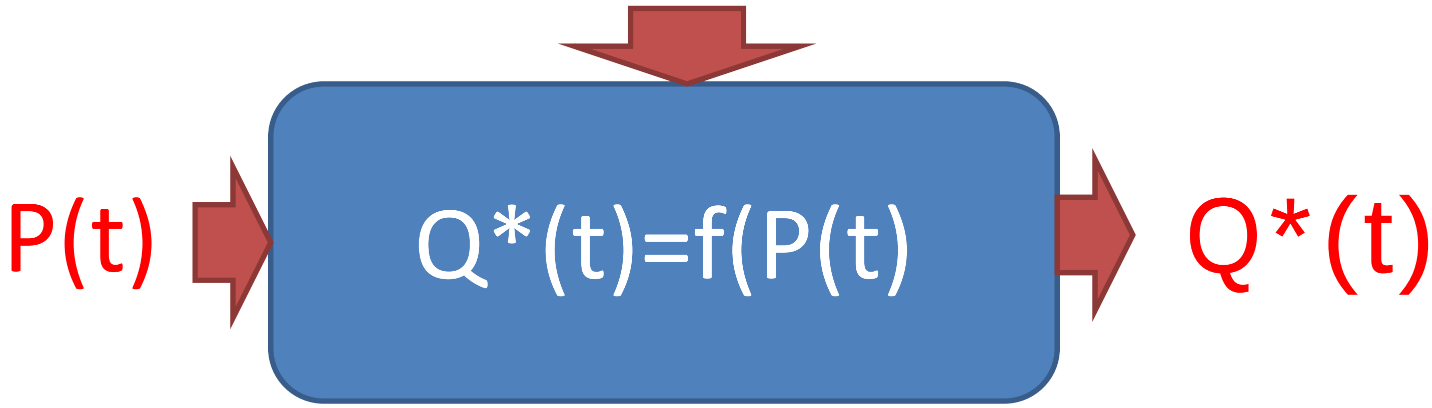
AIT view


data; (XY)
Prior info
(M)
K(M)
Compress
Program for universal computer
K(XYM) = K(XY|M) + K(M)
K(XYM) ≤ K(XY) + K(M)
Info content prior knowledge:
K(XY)-K(XY|M)
Program length;
K(XY|M)

DPI in AIT

- $K(Y^*)<K(X)+K(\text{model})$
- $K(Y)- K(Y|Y^*) \leq K(Y^*)$
- $K(Y|Y^*) = K(Y|X,\text{model}) \geq K\ (Y|X,\text{system})$

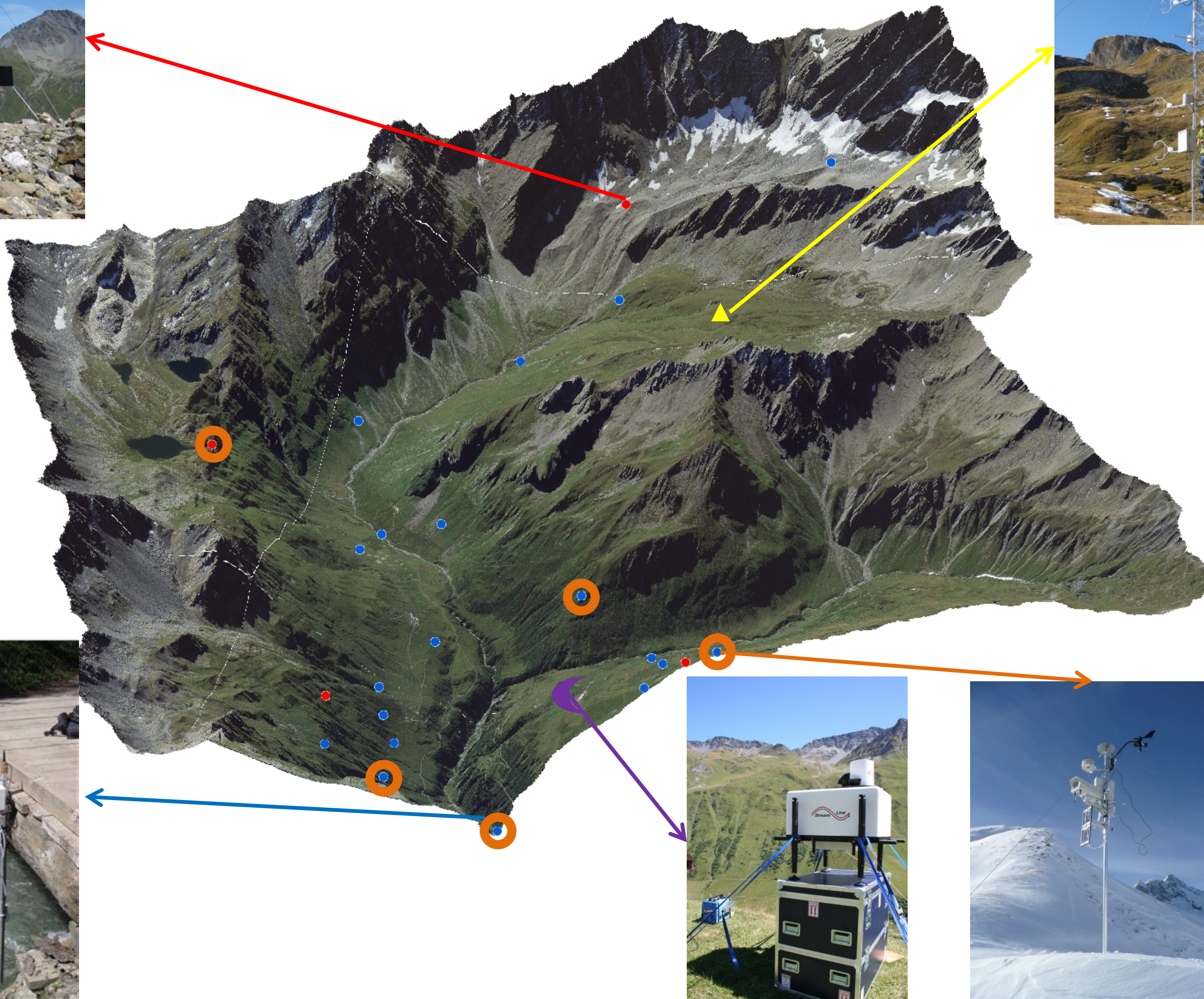
K(M):Mass balance, DEM, Land-use



Solution to paradox?

1. Yes, info is always lost
   - Compared to imaginary infinite data set
   - Or compared to imaginary perfect model
2. No, model adds info
   - When prior justified complexity present
   - Most important with small datasets
   - Model must be more than hypothesis

Small dataset!



Conclusions

- Info can only be destroyed not created
- DPI holds, but model can contain info
- Data → PMF not straightforward
- So benchmark info in data dubious
- AIT formulation for DPI = more general

Thanks!

comments? : Steven.weijs@epfl.ch

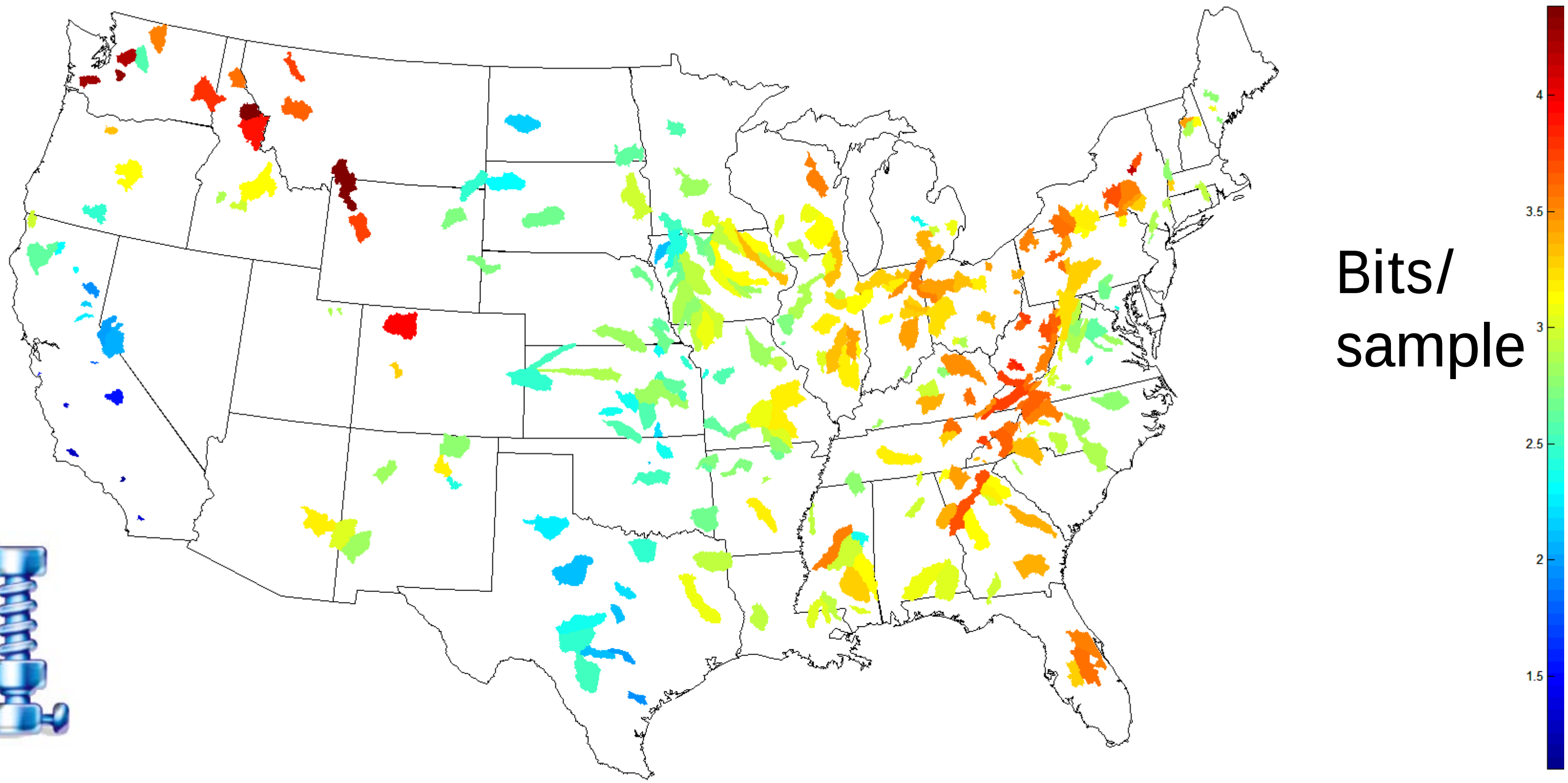
Application: info in data

- Kolmogorov complexity could be seen as measure of info content in data

Zip Q zip P zip PQ

- Explain learning

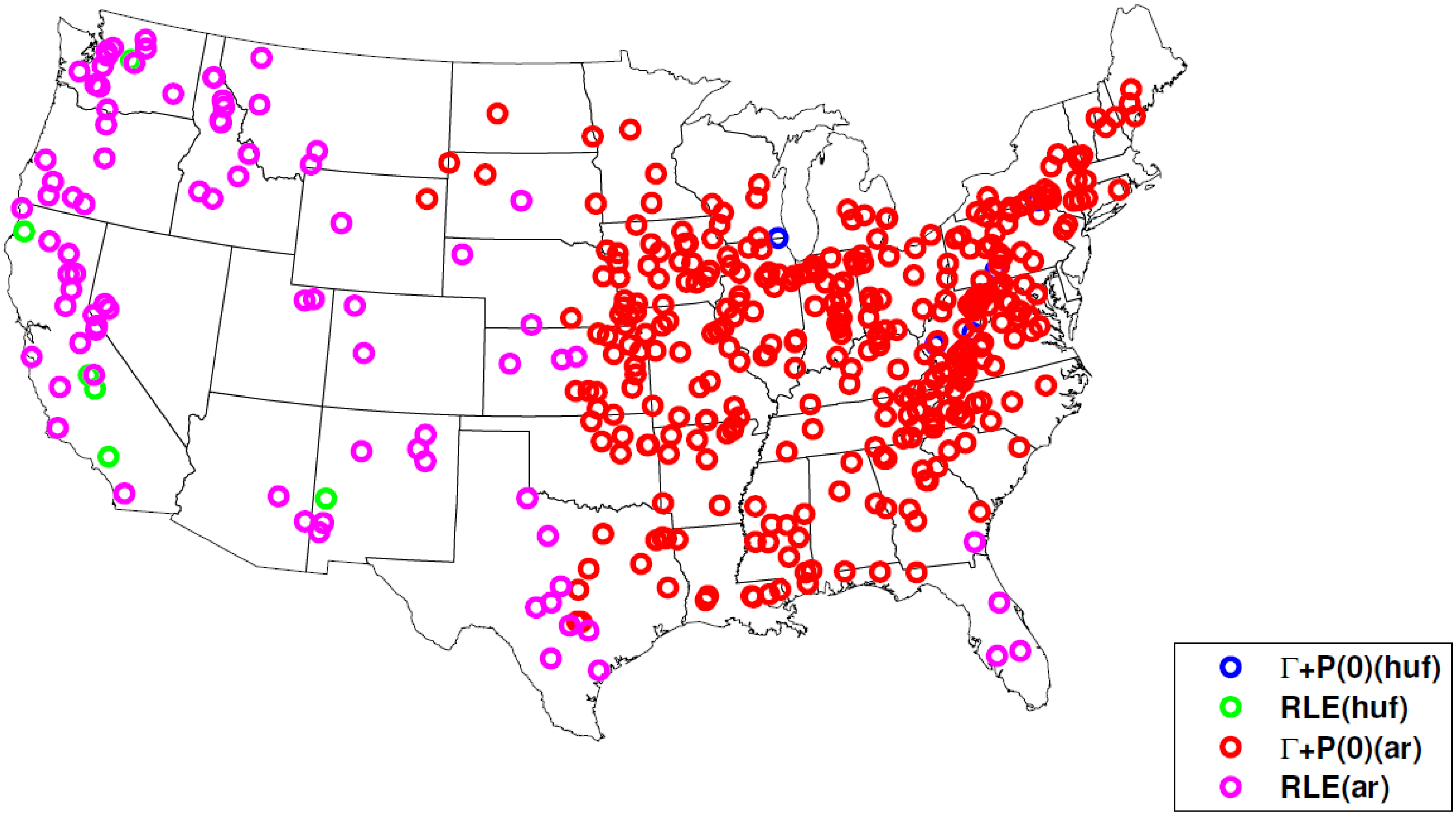
Data vs. information : P



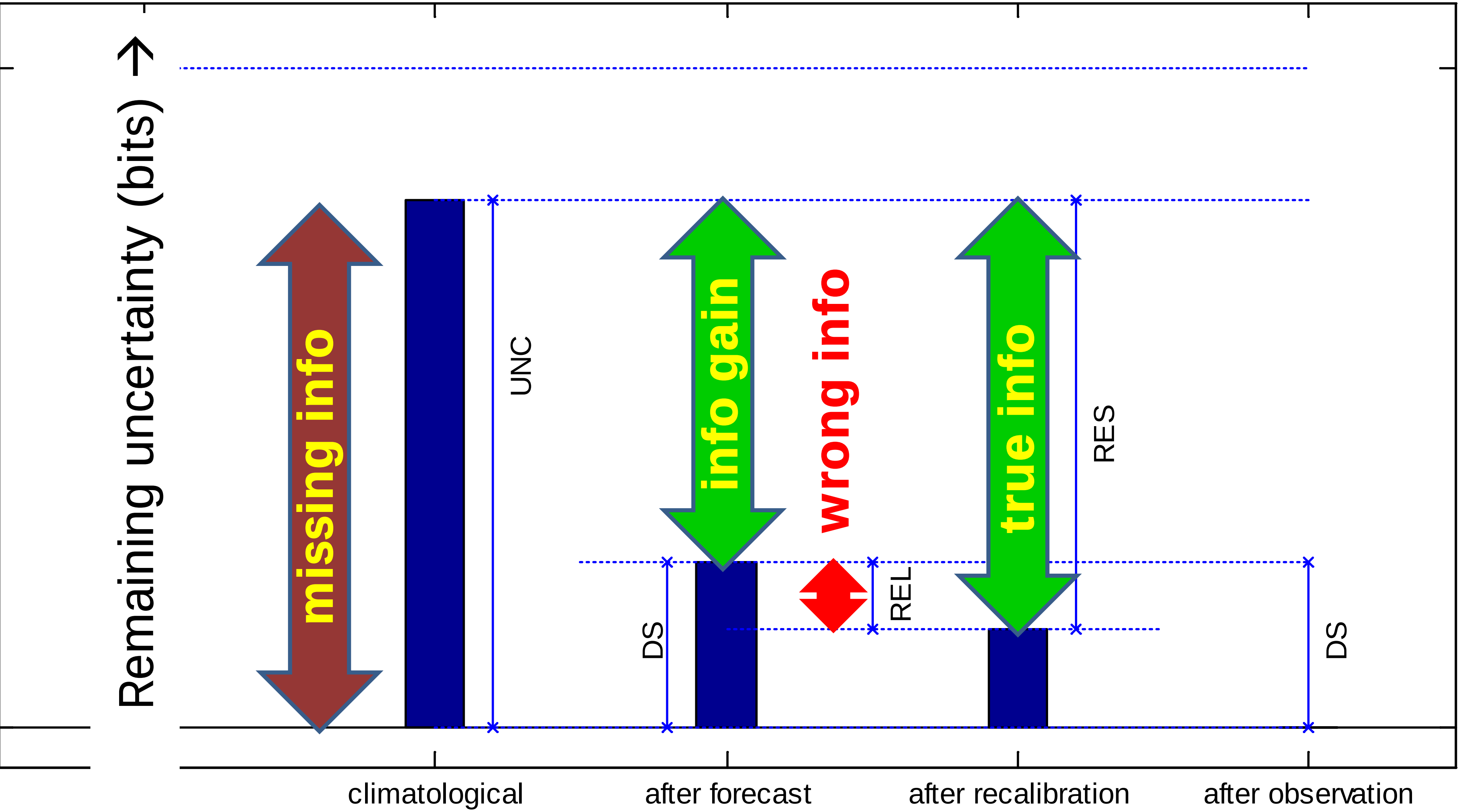
Weijs, S. V.; van de Giesen, N. & Parlange, M. B.
Data compression to define information content of hydrological time series
Hydrology and Earth System Sciences, **2013**, *17*, 3171-3187

Weijs, S. V.; van de Giesen, N. & Parlange, M. B.
HydroZIP: How Hydrological Knowledge can Be Used to Improve Compression of Hydrological Data,
Entropy, **2013**, *15*, 1289-1310

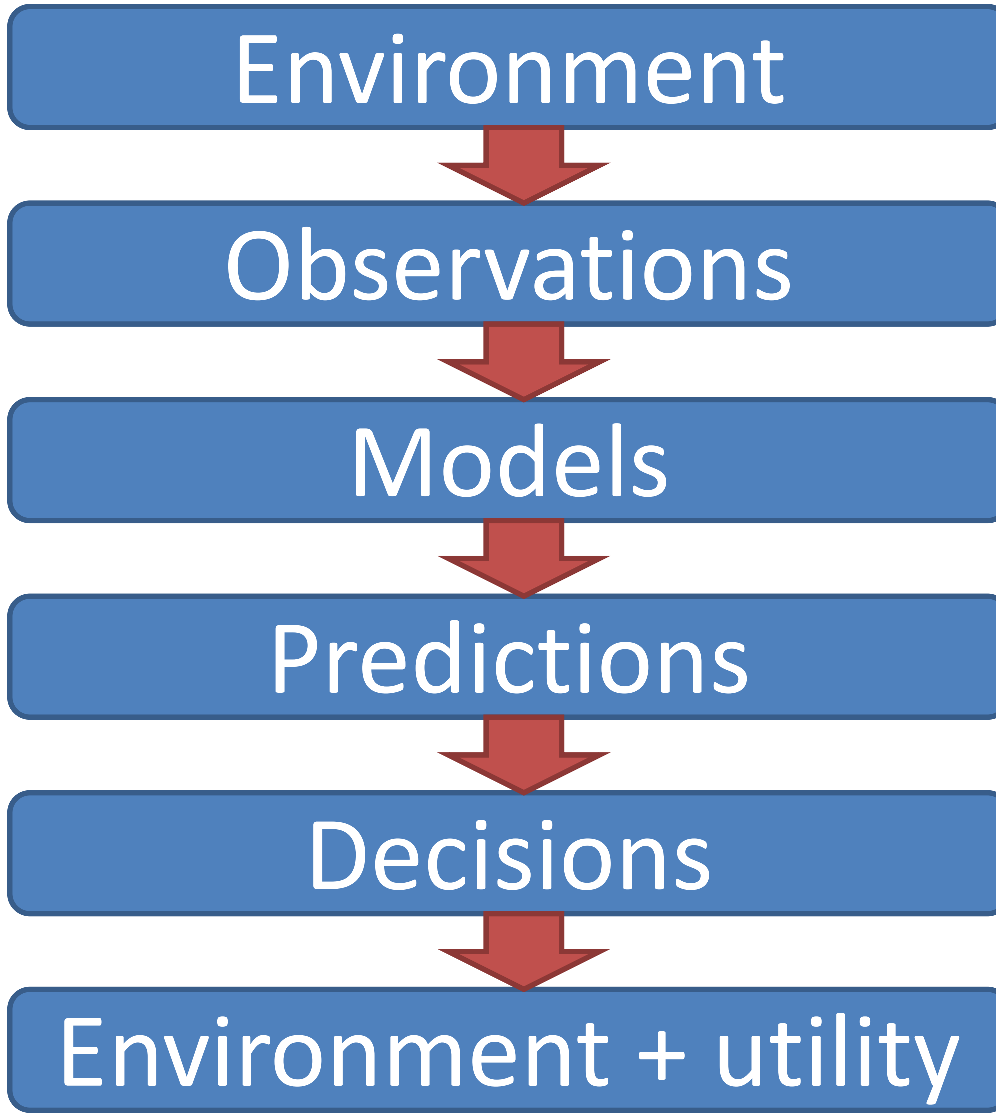
Which algorithm zips P best?



Information interpretation



Information flows



Two types of uncertainty

Perceived uncertainty

- Entropy
- Expected surprise about truth if uncertainty estimate is correct
- Best guess of average actual uncertainty

$$H(\mathbf{p}) = \sum_{i=1}^{n} p_i \log \frac{1}{p_i}$$

True uncertainty

- Actual surprise experienced when truth is revealed
- Only possible to evaluate ex-post
- Ideally equal to perceived uncertainty on average

$$\log \frac{1}{p_i}$$

Two types of information

A message

- Changes pmf (obs)
- Or pmf of pmf's (multhyp)
- E.g. coin is tested and fair
- Does not contain info for bet itself
- But does contain info about future learning from obs.
- Both true and perceived uncertainty might change

True message

- Moves pmf closer to rational one to have with new piece of info xxx??
- Will on average reduce true uncertainty.
- But may not in single instance (good decision can turn out wrong in hindsight)
- Might increase perceived uncertainty (solve previous over-conditioning) (swan)