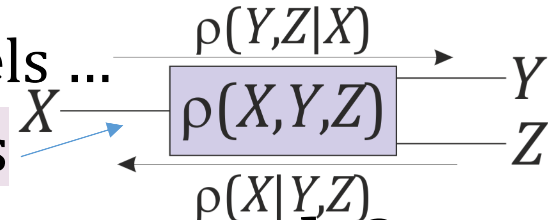
HIERARCHICAL CORRELATION RECONSTRUCTION

for time series, conditional distribution (~~Bayes~~) models …
(nonlinear, adaptive, all-directional) artificial neurons



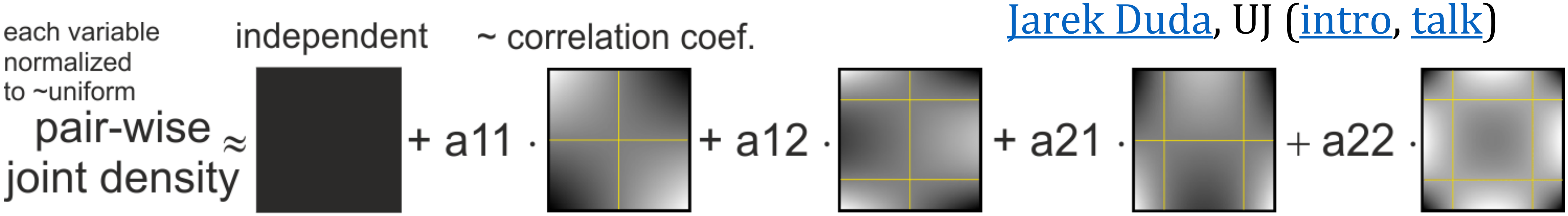
How to model/estimate density from a data sample?

MSE fit polynomial $\rho(x) = \sum_{f \in B} a_f\, f(x)$ (in (f) orthonormal basis)

also for **joint distribution**, **non-stationarity**, **missing data**

	Moments/cumulants	$\boldsymbol{\rho(x) = \sum_f a_f\, f(x)}$	**Machine learning**
# parameters	low – rough	from low to high	high - accurate
estimation	e.g. $m_k = \frac{1}{\lvert X \rvert} \sum_{x \in X} x^k$	$a_f = \frac{1}{\lvert X \rvert} \sum_{x \in X} f(x)$	usually iteration
Interpretable?	yes	Yes: mixed moments	depends
Independently?	yes	Yes (adapt, missing)	depends
Unique?	yes	yes (MSE)	often huge freedom
Accuracy?	controllable	controllable	usually uncontrollable
Density?	moment problem	**YES:** $\boldsymbol{\sum_f a_f\, f(x)}$	depends
→ complete	depends	yes	depends

Jarek Duda, UJ (intro, talk)

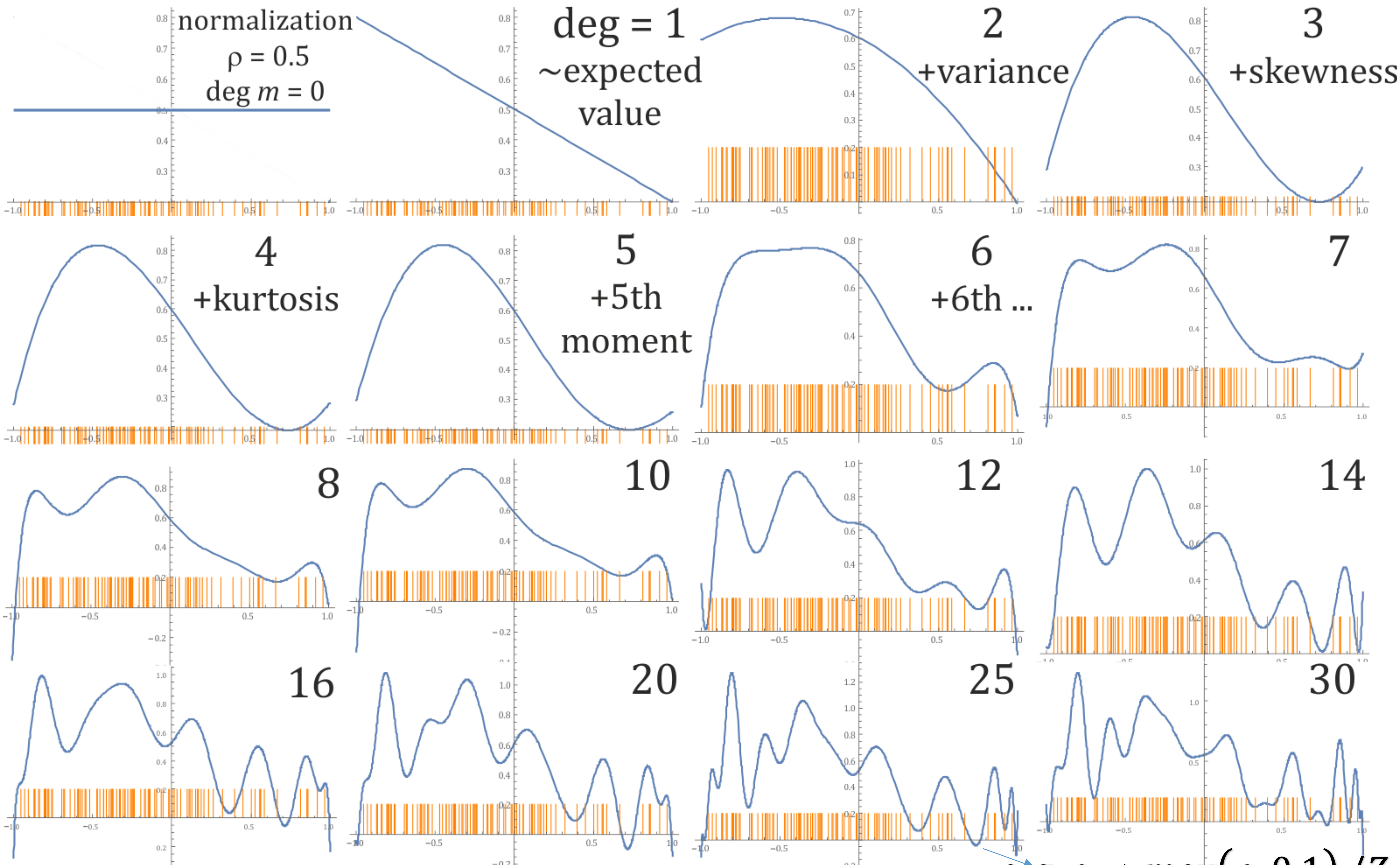


Articles using **hierarchical correlation reconstruction**: introduction with Mathematica code

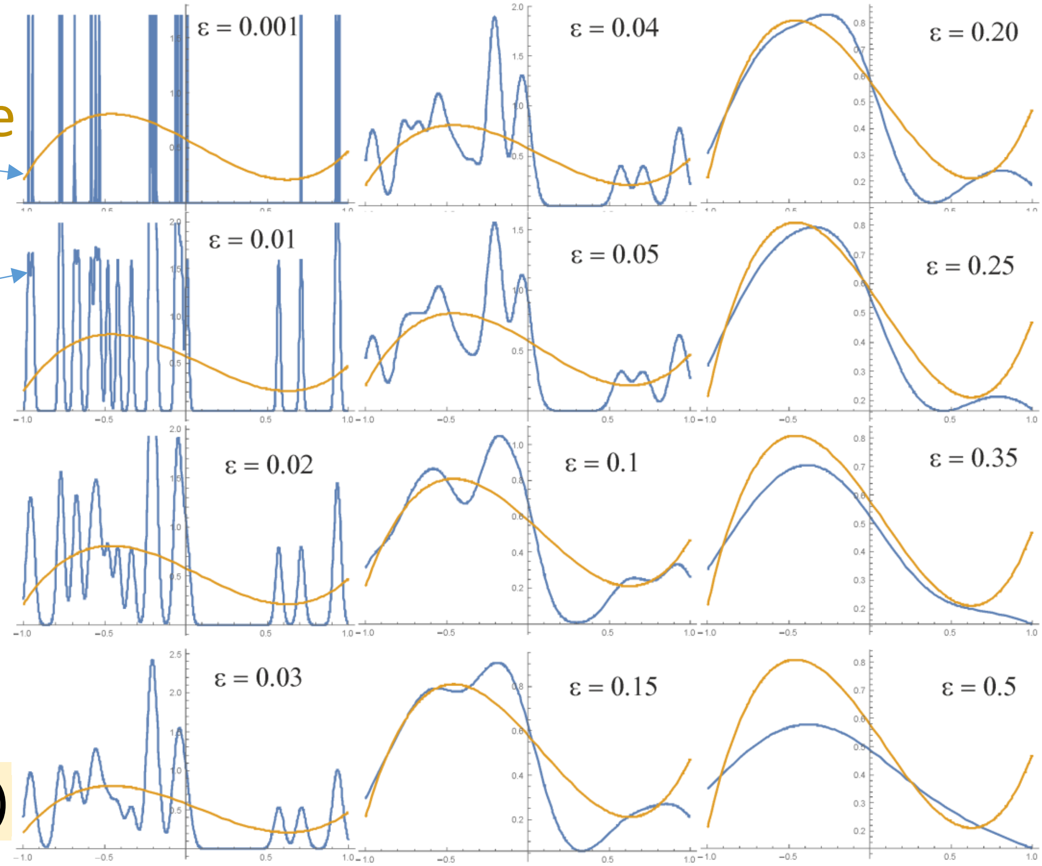
[1] J. Duda, Rapid parametric **density estimation**, arXiv:1702.02144 (2017)
[2] J. Duda, **Hierarchical correlation reconstruction** with missing data, for example for **biology-inspired neuron**, arXiv:1804.06218 (2018)
[3] J. Duda, **Exploiting statistical dependencies of time series** with hierarchical correlation reconstruction, arXiv:1807.04119 (2018)
[4] J. Duda, M. Snarska, Modeling joint probability distribution of **yield curve parameter**s, arXiv:1807.11743 (2018)
[5] J. Duda, A. Szulc, **Credibility evaluation** of income data with hierarchical correlation reconstruction, arXiv:1812.08040 (2018), International Conference on Applied Economics (2020)
[6] J. Duda, R. Syrek, H. Gurgul, Modelling **bid-ask spread conditional distributions** using hierarchical correlation reconstruction, arXiv:1911.02361(2019), Statistics in Transition vol 21 no 4 (2020)
[7] J. Duda, G. Bhatta, Log-stable probability density functions, **non-stationarity evaluation**, and **multi-feature autocorrelation analysis** of the γ-ray light curves of blazars, arXiv:2005.14040 (2020), Monthly Notices of the Royal Astronomical Society Main Journal (2021)
[8] J. Duda, H Gurgul, R. Syrek, Multi-feature evaluation of **financial contagion**, Central European Journal of Operations Research (2021)
[9] J. Duda, Predicting **conditional probability distributions** of redshifts of Active Galactic Nuclei using Hierarchical Correlation Reconstruction, arXiv:2206.06194 (2022), Monthly Notices of the Royal Astronomical Society Main Journal (2024)
[10] J. Duda, S. Podlewska, Low cost **prediction of probability distributions** of molecular properties for early virtual screening, arXiv:2207.11174, Molecular Diversity (2022)
[11] J. Duda, **Time delay multi-feature correlation analysis** to extract subtle dependencies from EEG signals, arXiv:2305.09478 (2023)
[12] J. Duda, **Extracting individual variable information** for their decoupling, direct mutual information and multi-feature Granger causality, *arXiv:2311.13431* (2023)
[13] J. Duda, J. Leśkow, P. Pawlik, W. Cioch, CMAFI — Copula-based Multifeature Autocorrelation Fault **Identification of rolling bearing**, Mechanical Systems and Signal Processing (2024)
[14] J. Duda, Biology-inspired **joint distribution neurons** based on Hierarchical Correlation Reconstruction allowing for **multidirectional neural networks**, *arXiv:2405.05097*

$n = 100$ size **1D** **sample** (from degree = 3), **density estimated as polynomial**:

on [-1,1] $\approx$ (deg $m \to \infty$ leads to sum of Dirac deltas)



e.g. $\rho \to \max(\rho, 0.1)/Z$



Derivation: $n = 25$ size sample

KDE (kernel density estimation):

g_ϵ : ϵ-width Gaussian in each point

Find $\rho_{\boldsymbol{a}}(\boldsymbol{x}) = \sum_j a_j f_j(x)$ minim. MSE

$$\arg\min_{\boldsymbol{a}} \int (\rho_{\boldsymbol{a}} - g_\epsilon)^2 dx =$$

$$\arg\min_{\boldsymbol{a}} \ \|\rho_{\boldsymbol{a}}\|^2 - 2\langle \rho_{\boldsymbol{a}}, g_\epsilon \rangle + \|g_\epsilon\|^2$$

Taking $\epsilon \to 0$, $\langle \rho_{\boldsymbol{a}}, g_\epsilon \rangle = \sum_{x\in X} \rho_{\boldsymbol{a}}(x)$

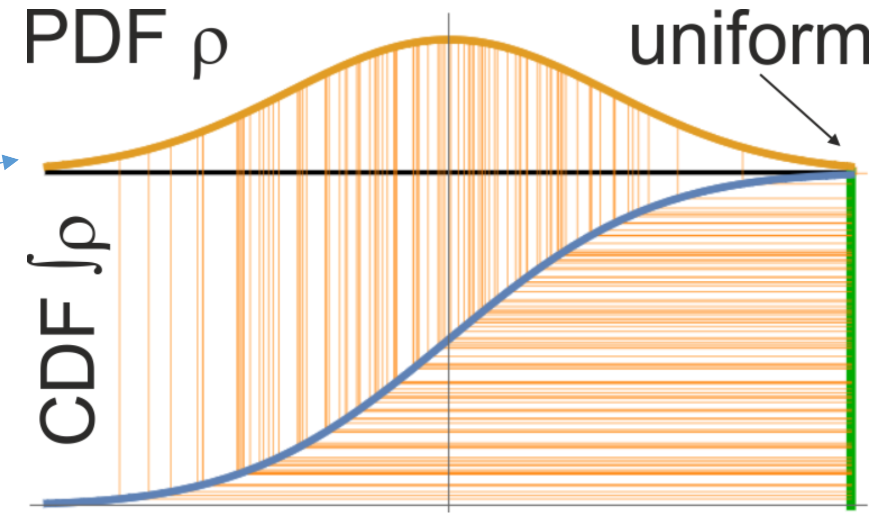
Removing $\lim_{\epsilon\to 0} \|g_\epsilon\|^2 = \infty$ which does not affect parameters $\boldsymbol{a}$

Using **orthonormal**: $\langle f_i, f_j \rangle = \int f_i(x) f_j(x) dx = \delta_{ij}$ e.g. on $[0,1]^d$

$$\arg\min_{\boldsymbol{a}} \|\rho_{\boldsymbol{a}}\|^2 - \frac{2}{n}\sum_{x\in X} \rho_{\boldsymbol{a}}(x) = \arg\min_{\boldsymbol{a}} \sum_j (a_j)^2 - \frac{2}{n}\sum_{x\in X}\sum_{j\in B} a_j f_j(x)$$

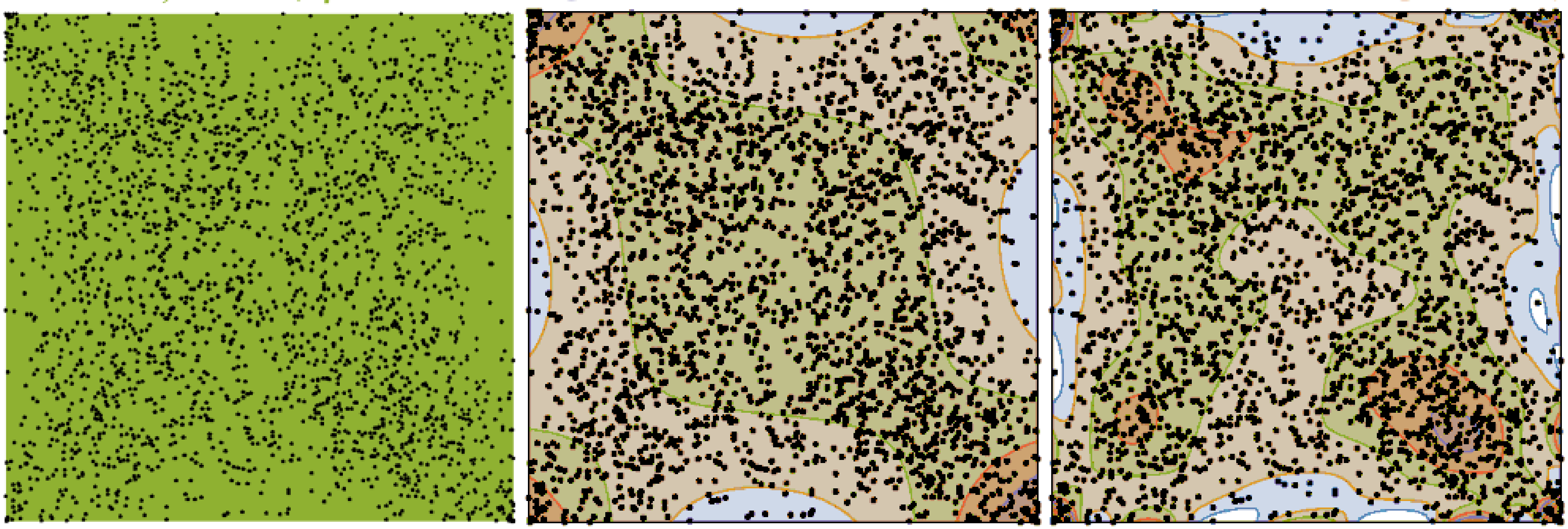
$$\text{minimum:} \quad \partial_{a_j} = 0 \quad \Rightarrow \quad a_j = \frac{1}{n}\sum_{x\in X} f_j(x)$$

In practice: **normalize each variable to ~ uniform distribution:** $\boldsymbol{x^t = \mathrm{CDF}(y^t)}$
(1/2: median, position: quantile, like [copula])
Then fit polynomial as joint distribution
(daily log returns: $\ln(v_{t+1}/v_t)$)

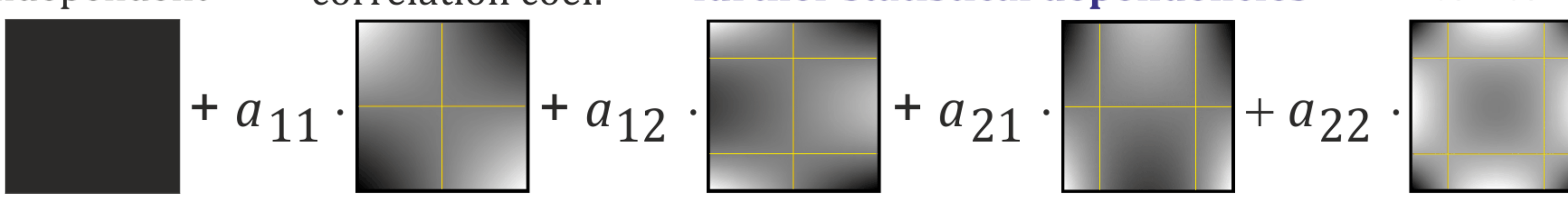


(x^{t-1}, x^t) pairs from TRV series + used estimated density ρ isolines

MLE κ, m=0, ρ=1 HCR m=2 0, 0.5, 1, 1.5, 2, 2.5 HCR m=9

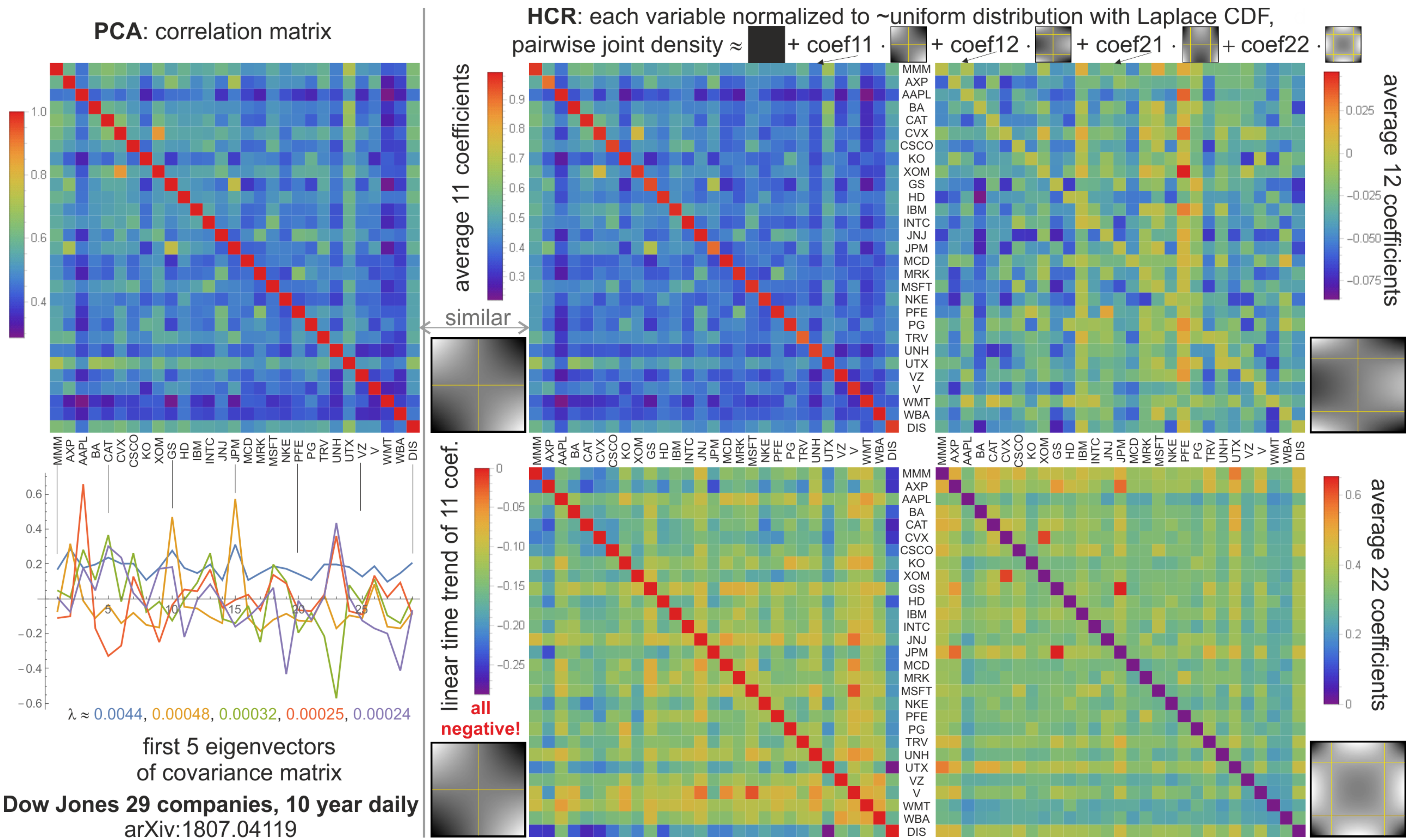


independent ~ correlation coef. **further statistical dependencies** var-var

pair-wise joint density ≈  $+\, a_{11} \cdot$ … $+\, a_{12} \cdot$ … $+\, a_{21} \cdot$ … $+\, a_{22} \cdot$ …

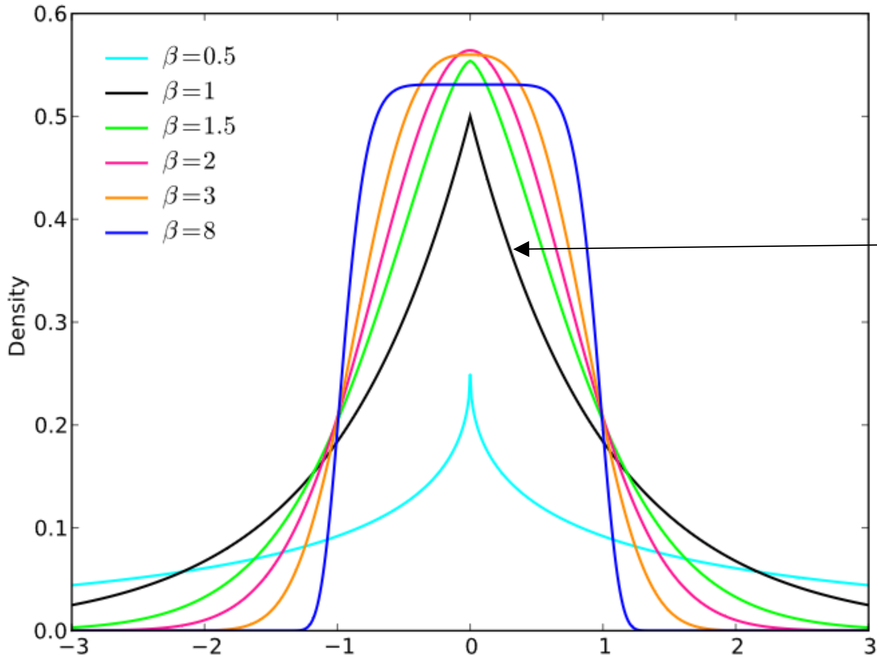
Basic application: **many mixed-moment features** e.g. for time series classification

Standard: pairwise correlation “11”, here: also higher, “triple+”wise, time dependent



Normalization $x = \mathrm{CDF}(y)$ to $x \sim \mathrm{uniform}[0,1]$

Generalized normal distribution/EPD $\rho \sim \exp(-|x|^\kappa)$



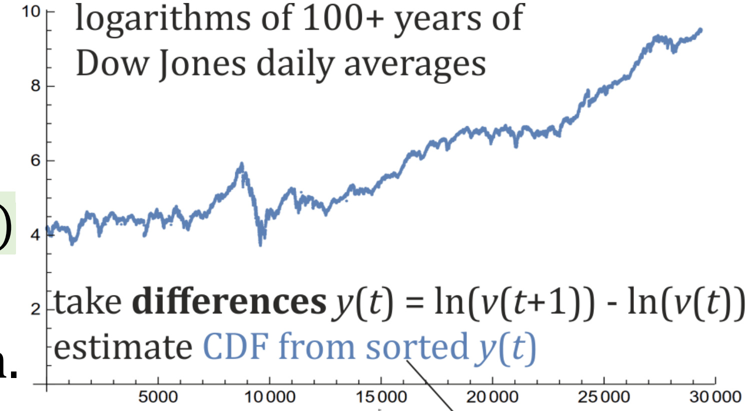
Laplace, MLE estim.

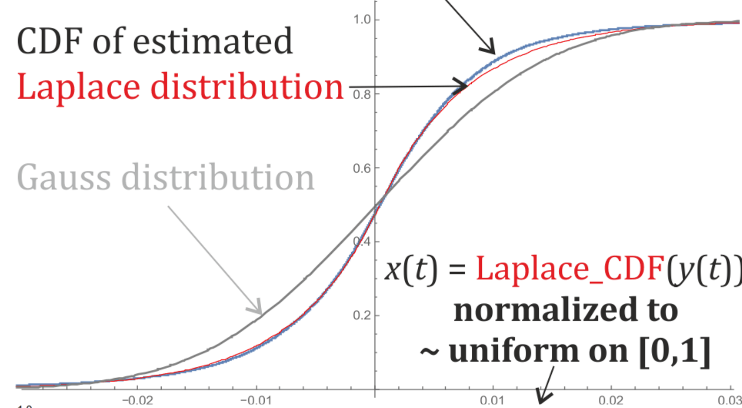
$$\rho = \frac{1}{2b}\exp\left(-\frac{|x-\mu|}{b}\right)$$

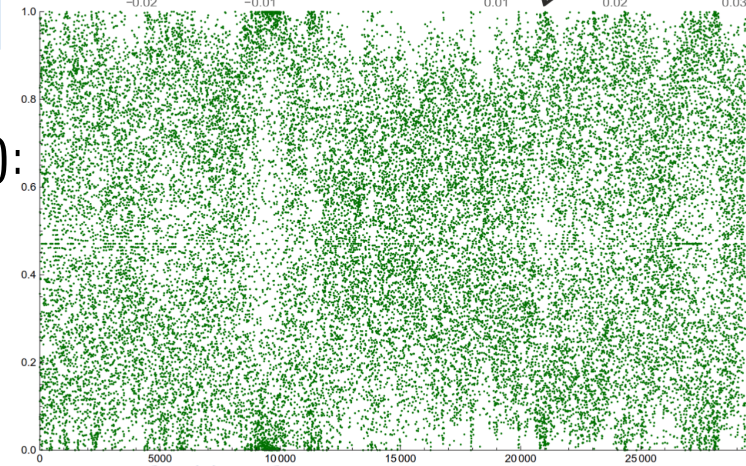
$$\hat{\mu} = \text{median}$$

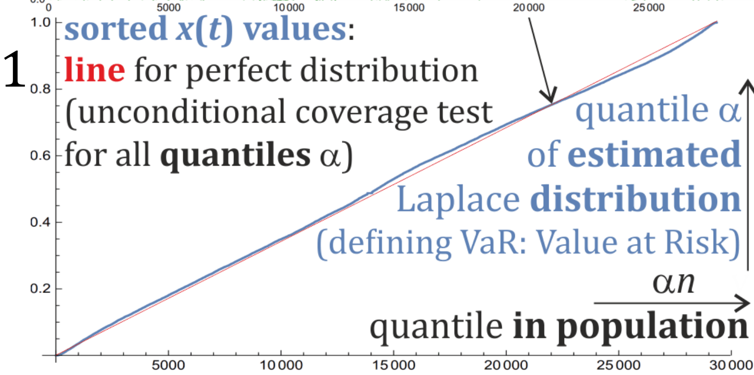
$$\hat{b} = \frac{1}{N}\sum_i |x_i - \hat{\mu}|$$

Normalization contains tail model

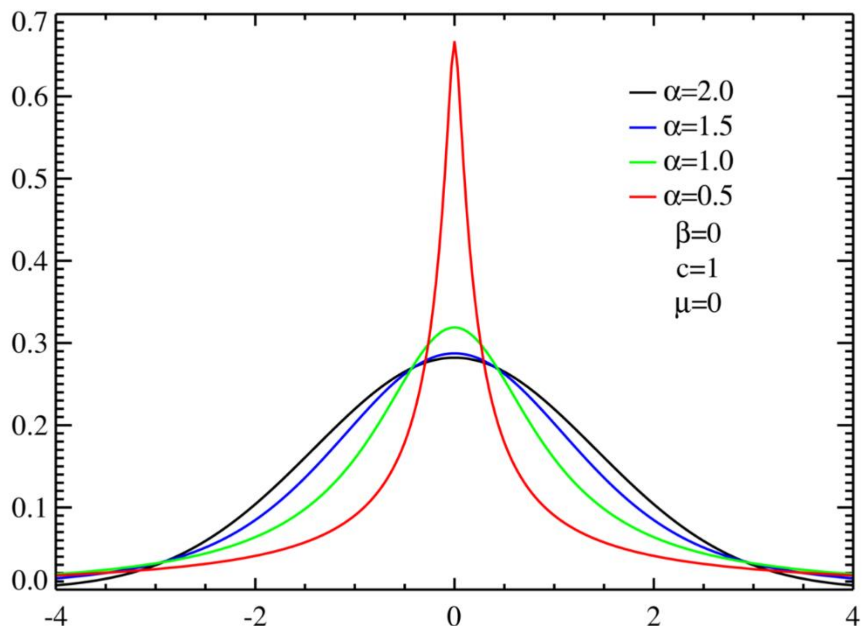








Lévy/**stable distribution** $\rho \sim |x|^{-1-\alpha}$ tail (∞ moments):



Student's t-dist.:

$$\rho \propto \left(1+\frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}}$$

$\widehat{\text{Gauss}}$ of $n = \nu - 1$

Cauchy for $\nu = 1$

∞ moments $\geq \nu$

E.g. for **ARMA/ARCH** enhancement

Gaussian-based, often terrible LL

(8σ: $1/3 \cdot 10^{12}$ yrs … S&P 500: 1/10 yrs)

(daily log returns for 29 **Dow Jones**)

MLE gives much lower power $\kappa \ll 2$:

Having approximate parametric dist.

we can normalize as in copula theory

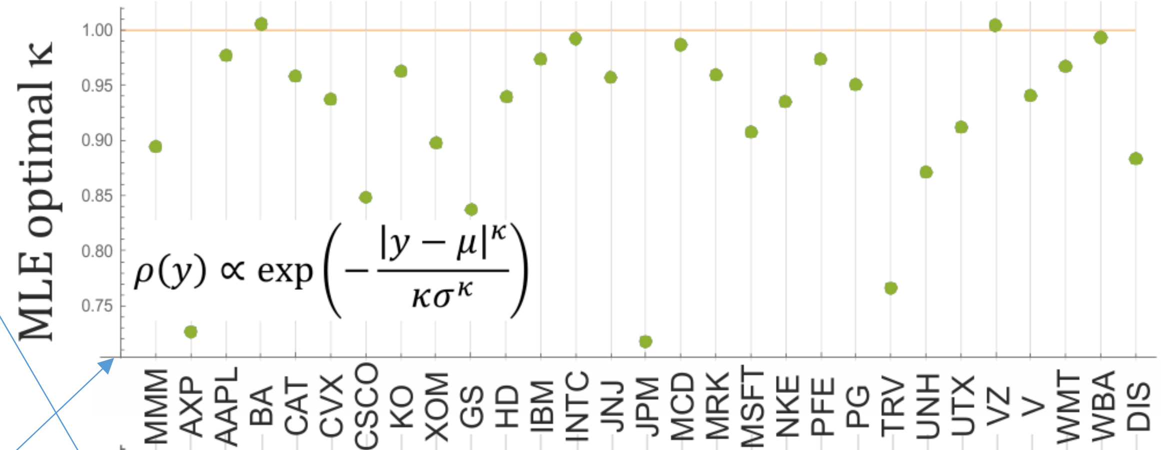
to $x \sim$ uniform on [0,1] distribution:

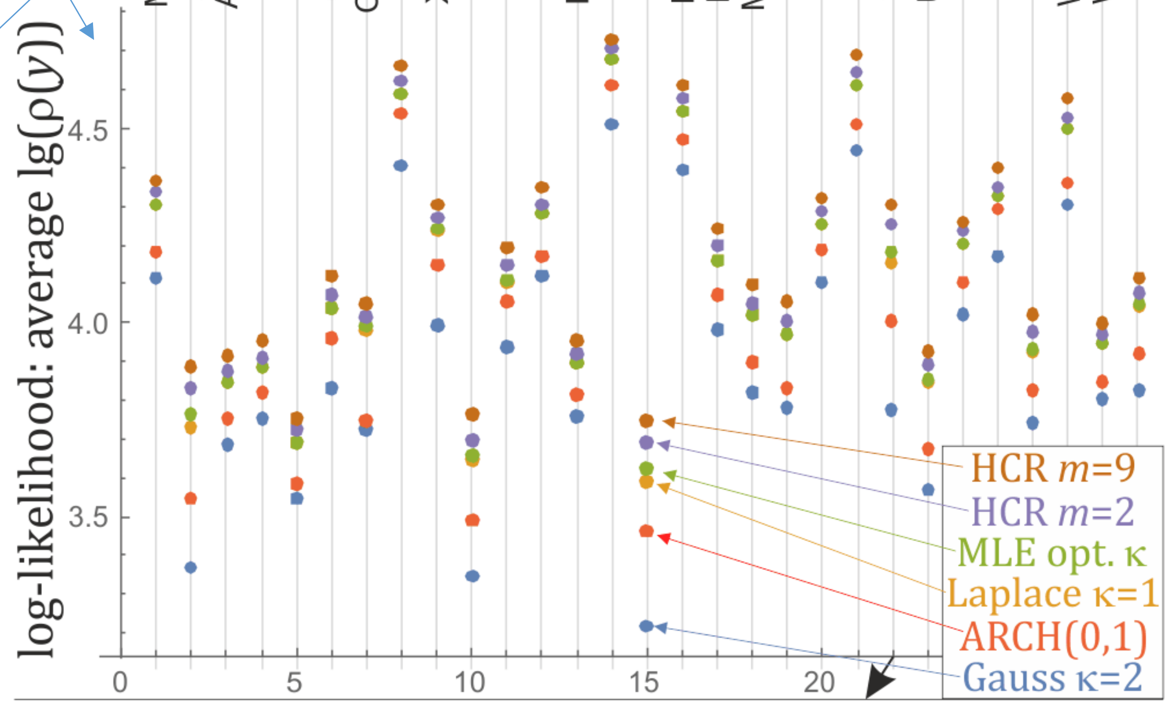
$x^t = \mathrm{CDF}_{\mathrm{parametric}}(y^t)$

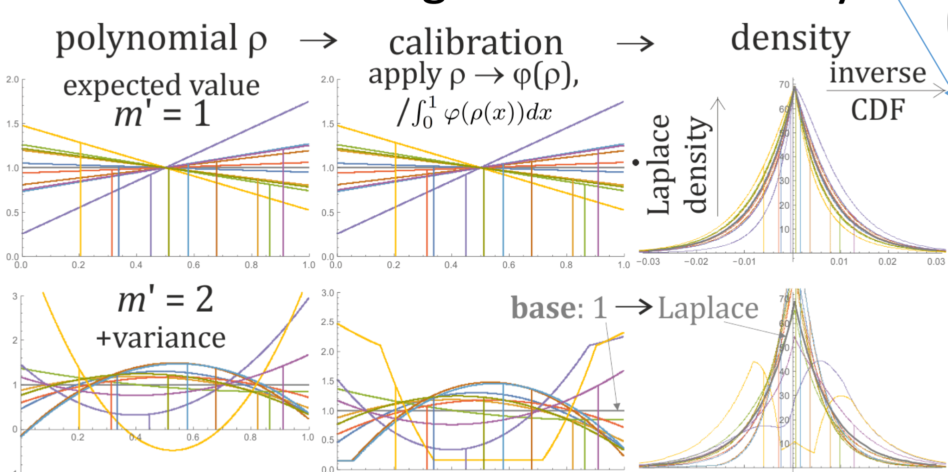
HCR: Fit degree m polynomial

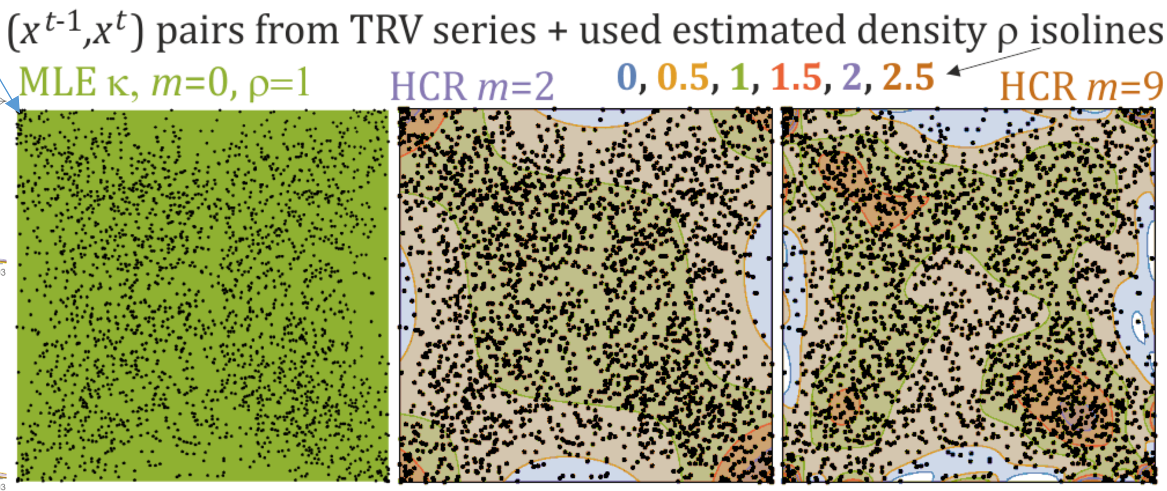
e.g. to $(\boldsymbol{x^{t-1}}, \boldsymbol{x^t})$ **joint distribution**

can be evolving for nonstationary









Adaptivity: models evolving with time

We can **normalize** with $\boldsymbol{x_t} = \mathbf{CDF}_{\boldsymbol{t}}(\boldsymbol{y_t})$

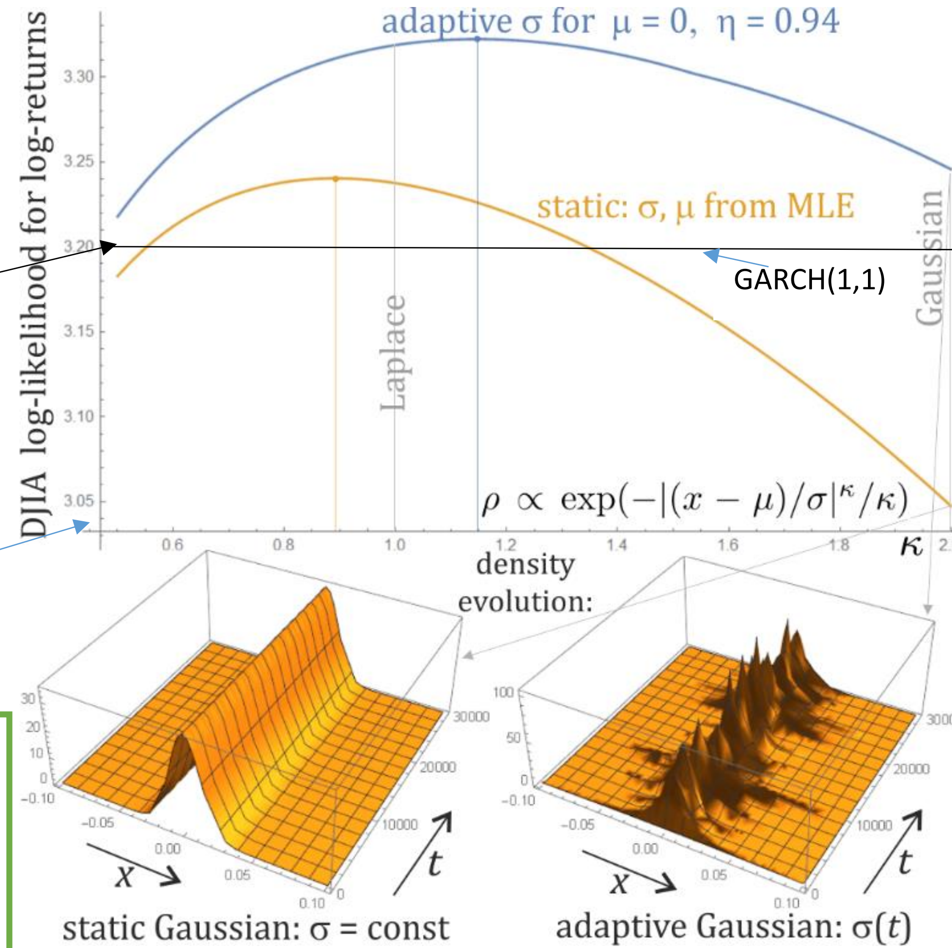
e.g. Gaussian with varying σ like in GARCH

e.g. average → exponential moving average

EPD width: $\widehat{\sigma^\kappa} = \frac{1}{n}\sum_{x\in X}|x-\mu|^\kappa$ →

$$\widehat{\sigma^\kappa}^{T+1} = \eta\,\widehat{\sigma^\kappa}^{T} + (1-\eta)\,|x^T-\mu|^\kappa$$

Student's t (next slide): arXiv:2304.03069



Optimizing **exponential moving estimator**:

log-lik: $\theta^T = \mathrm{argmin}_\theta \sum_{t<T} \eta^{T-t} \ln(\rho_\theta(x^t))$

Preferably $\eta = \mathrm{argmin}_\eta \sum_T \ln(\rho_{\theta^T}(x^T))$

Weighted linear regression: $\beta = \mathrm{argmin}_\beta \sum_i w_i\,((M\beta)_i - x_i)^2$

$$\beta = (M^T M)^{-1} M^T x \quad \Rightarrow \quad \beta = (M^T \operatorname{diag}(w)\, M)^{-1}\, M^T \operatorname{diag}(w)\, x$$

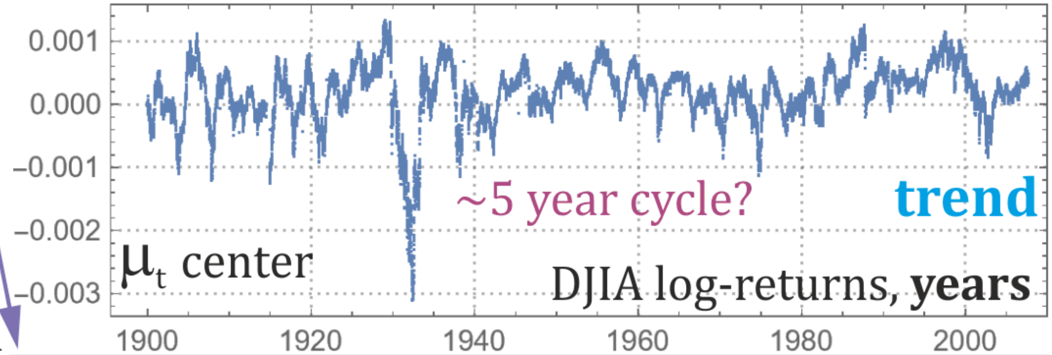
Adaptive linear regression: $\beta^T = \mathrm{argmin}_\beta \sum_{t<T} \eta^{T-t}((M\beta)_t - x_t)^2$

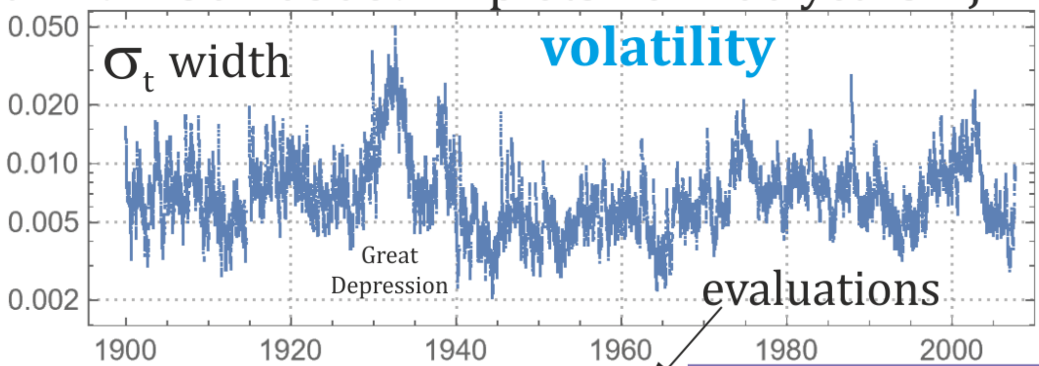
$\beta^T = (\mathcal{M}^T)^{-1} y^T$ for exponential moving averages:

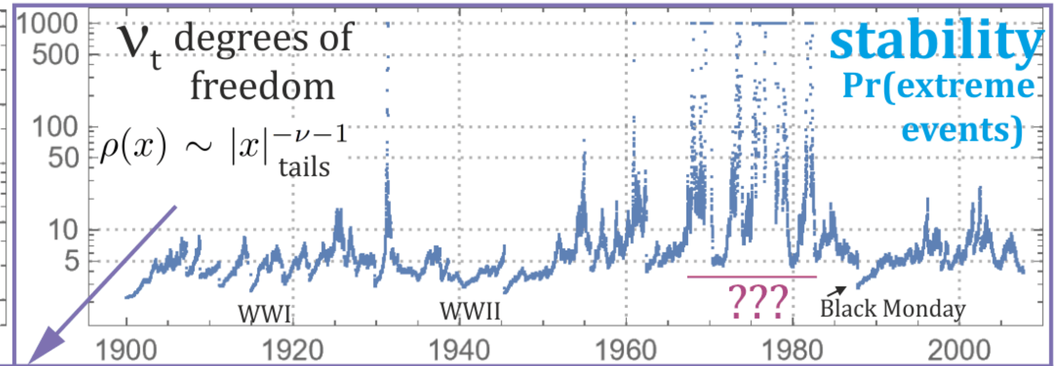
$$y^{T+1} = y^T + \eta(x^T M_{T\cdot} - y^T) \qquad \mathcal{M}^{T+1} = \mathcal{M}^T + \eta\big((M_{T\cdot})(M_{T\cdot})^T - \mathcal{M}^T\big)$$

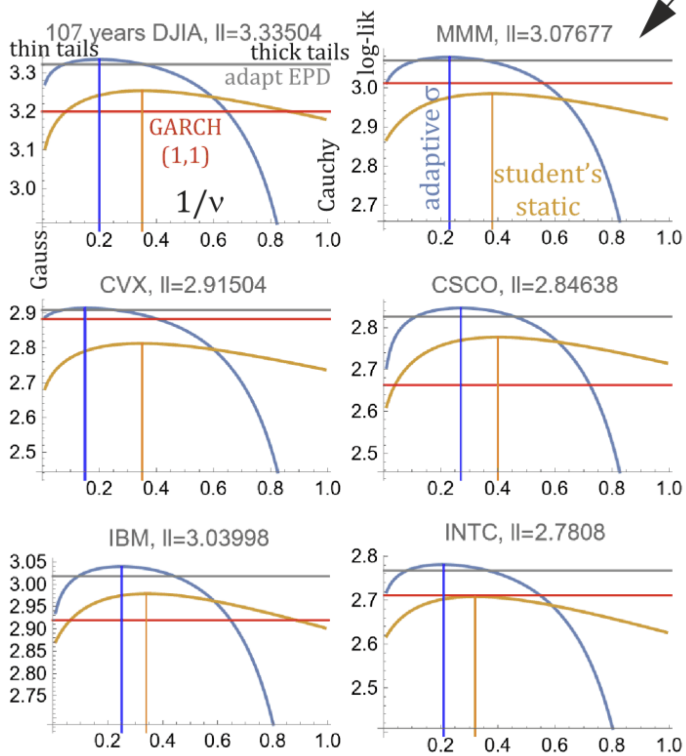
Adaptive Student's t-distribution with moving estimators of $\vec{\mu}, \sigma, \nu$

agnostic alternative to ARMA-ARCH-like:
don't assume arbitrary dependence type,
only **shift local estimators**:
with EMA: exponentially weakening weights
arXiv: 2304.03069 plots: for 100 years DJIA









evaluation: log-likelihoods

Numbers of **extreme events** for 107 years DJIA daily log-returns: data vs expected Student's t-distribution

event	$\sigma=\sqrt{var}$	adapt σ	$\nu=1$ Cauchy	$\nu=2$	$\nu=3$	$\nu=4$	$\nu=5$	$\nu=6$	$\nu=10$	$\nu=\infty$ Gauss
1σ	5204	11191	14674.5	12404.3	11475.5	10973.6	10660.1	10445.8	10004.9	9312.75
2σ	1170	3281	8662.86	5385.64	4089.08	3407.9	2991.82	2712.62	2153.87	1335.39
3σ	422	916	6011.64	2801.83	1692.52	1172.26	883.383	704.617	391.623	79.2363
4σ	194	316	4577.22	1678.5	822.02	473.402	302.982	208.935	73.9105	1.85904
5σ	96	133	3688.17	1107.91	451.753	219.837	120.469	71.9738	15.7702	0.0168259
6σ	58	74	3085.66	782.781	272.145	113.949	54.1823	28.3081	3.87726	0.0000579107
7σ	43	48	2651.23	581.226	175.691	64.3368	26.9056	12.4288	1.09049	7.51224×10^{-8}
8σ	29	37	2323.47	448.103	119.643	38.8551	14.4663	5.97148	0.345583	3.65158×10^{-11}
9σ	28	29	2067.54	355.759	84.9892	24.7656	8.29637	3.08961	0.121448	6.62459×10^{-15}
10σ	23	20	1862.22	289.16	62.4664	16.4942	5.01714	1.69989	0.0466518	4.4727×10^{-19}

Above all 29349 days, **below** 4012 days **1967-1983 - no 6σ (extreme) events, much closer to Gaussian**

event	1967−1983	$\nu=1$ Cauchy	$\nu=2$	$\nu=3$	$\nu=4$	$\nu=5$	$\nu=6$	$\nu=10$	$\nu=\infty$ Gauss
1σ	1078	2006.	1695.67	1568.7	1500.09	1457.23	1427.94	1367.66	1273.05
2σ	204	1184.21	736.216	558.976	465.859	408.981	370.814	294.433	182.547
3σ	38	821.789	383.009	231.368	160.247	120.758	96.3209	53.5347	10.8316
4σ	13	625.705	229.45	112.37	64.7139	41.4175	28.5613	10.1036	0.25413
5σ	3	504.172	151.451	61.7545	30.0516	16.4681	9.8388	2.15578	0.00230009

extreme events - usually $\nu \sim 4$ **heavy tails**, but **~Gauss in 1970s**??

Having modelled joint distribution for **missing data**: $a_{\boldsymbol{j}} = \frac{1}{|X_{\boldsymbol{j}}|}\sum_{\boldsymbol{x}\in X_{\boldsymbol{j}}} f_{\boldsymbol{j}}(\boldsymbol{x})$

substituting known coordinates to $\rho(\boldsymbol{x}) = \sum_{\boldsymbol{j}\in B} a_{\boldsymbol{j}} \; f_{j_1}(x_1)\cdot \ldots \cdot f_{j_d}(x_d)$

we get joint distribution of missing coordinates **(conditionals avoiding Bayes)**

Imputation – modelling missing values, e.g. as expected value for each coordinate

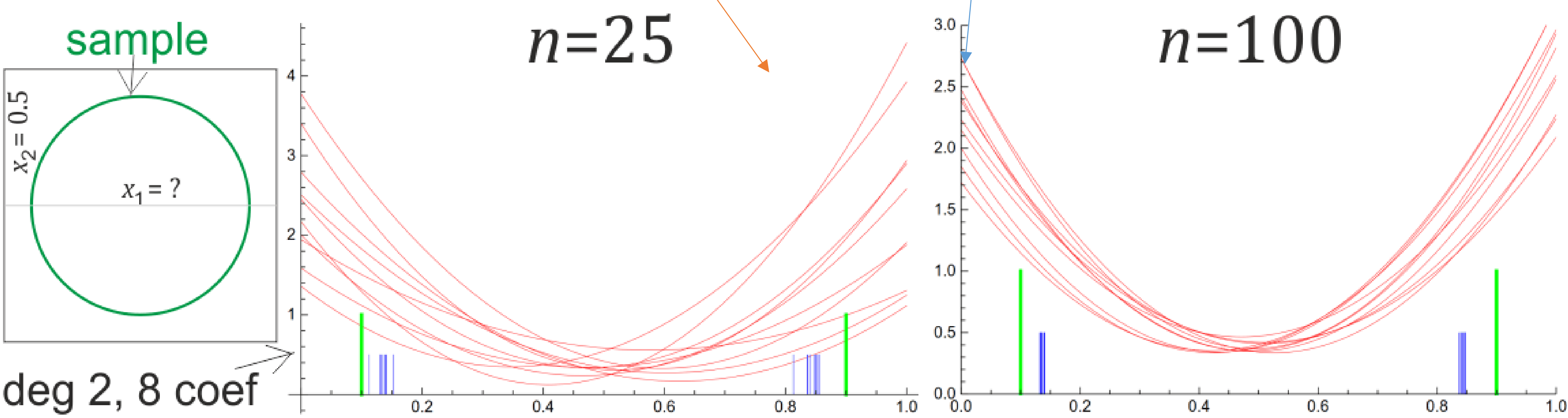
However, sometimes **ambiguity**, e.g. circle as sample below we can handle.

Here we can **model distribution of each missing coordinate** as polynomial,

or even **joint distribution of multiple missing coordinates**

circle (2D) centered in (0.5,0.5), $r = 0.4$ Knowing only $x_2 = 0.5$, $x_1 = ???$

we can get e.g. (joint) **distribution**, or **expected values for (2) clusters** ...



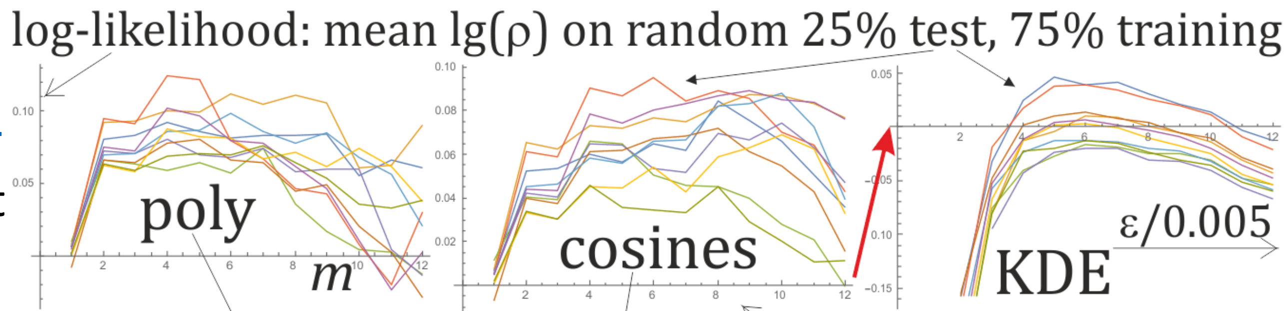
KDE – kernel density estimation

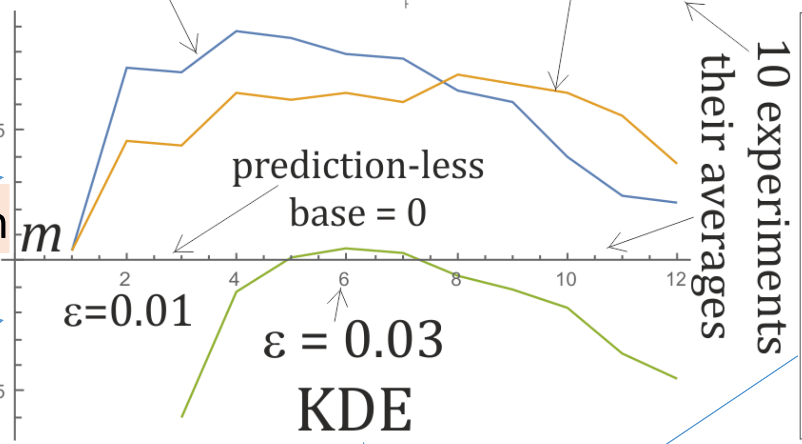
e.g. ϵ-radius Gaussian in each point

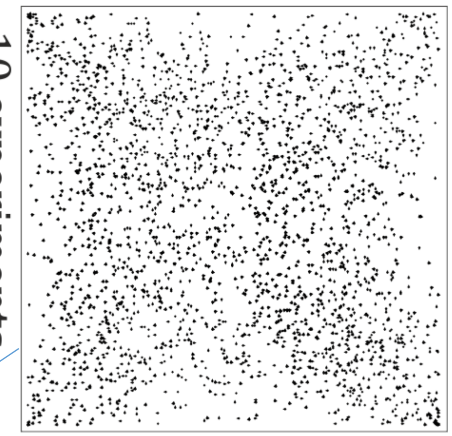
- huge #parameters ~ #points
- how to choose (ellipse?) radii??
- doesn't work in high dimension
- terrible log-likelihood, generalization

  as it localizes in the old points

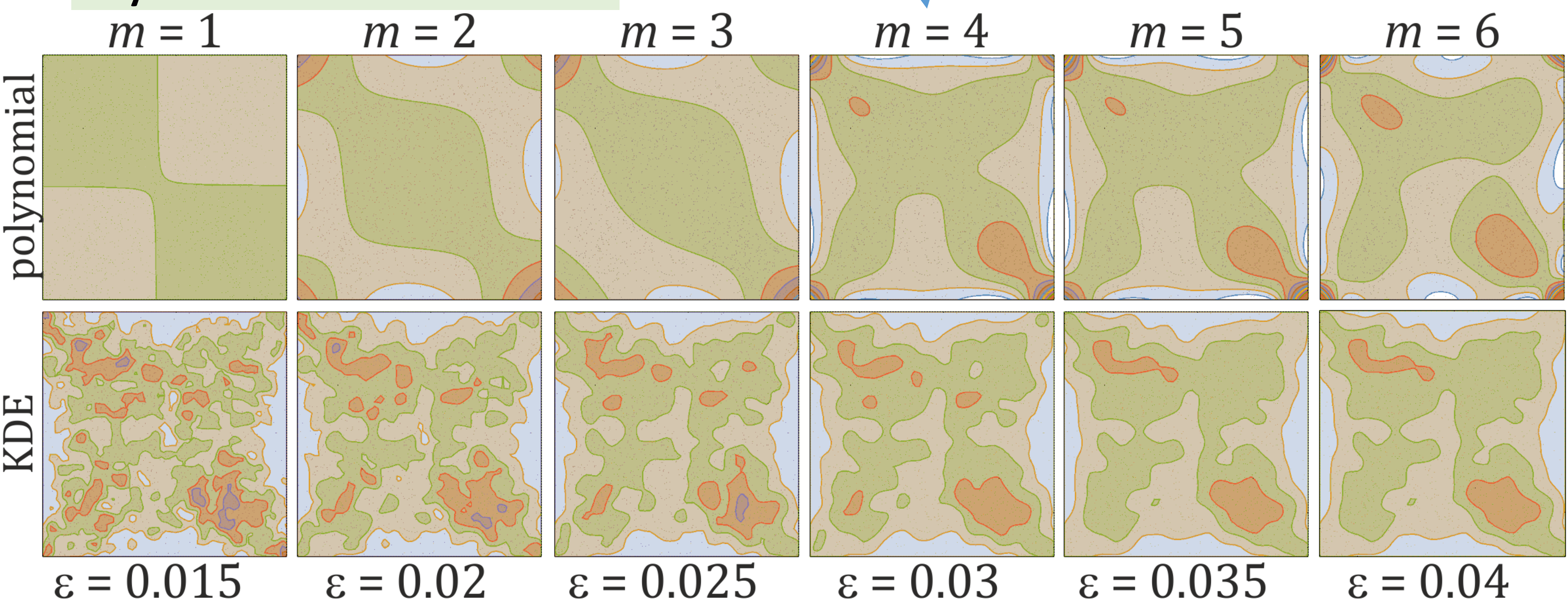
cross-validation:





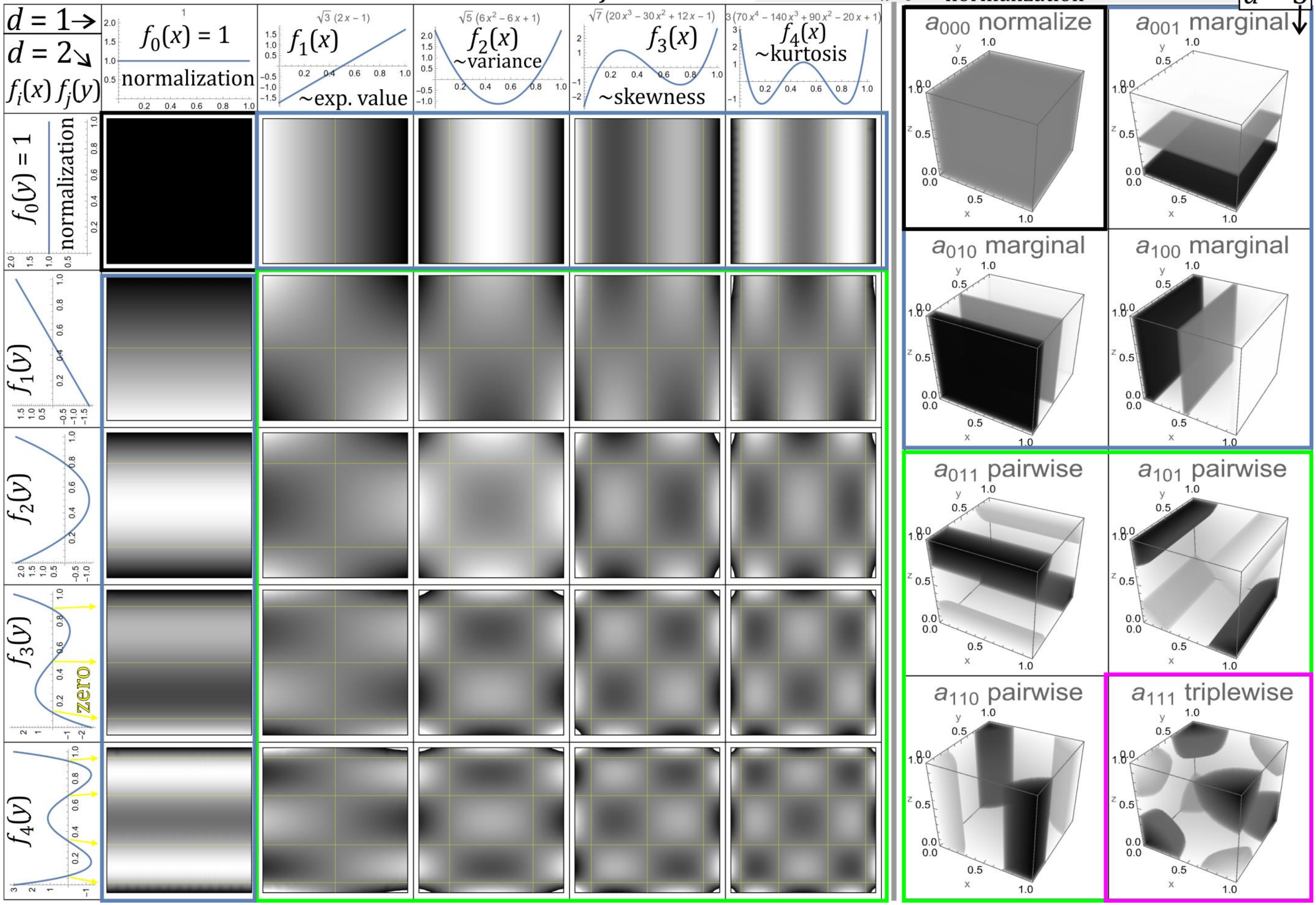


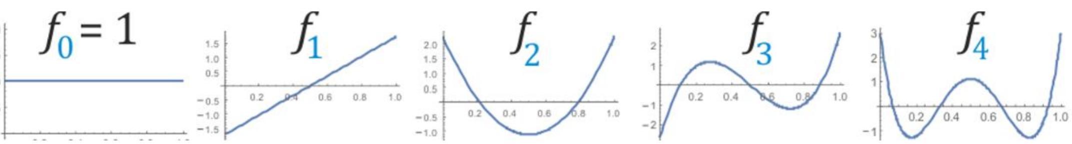
Polynomial: MSE fitted to $\epsilon \to 0$



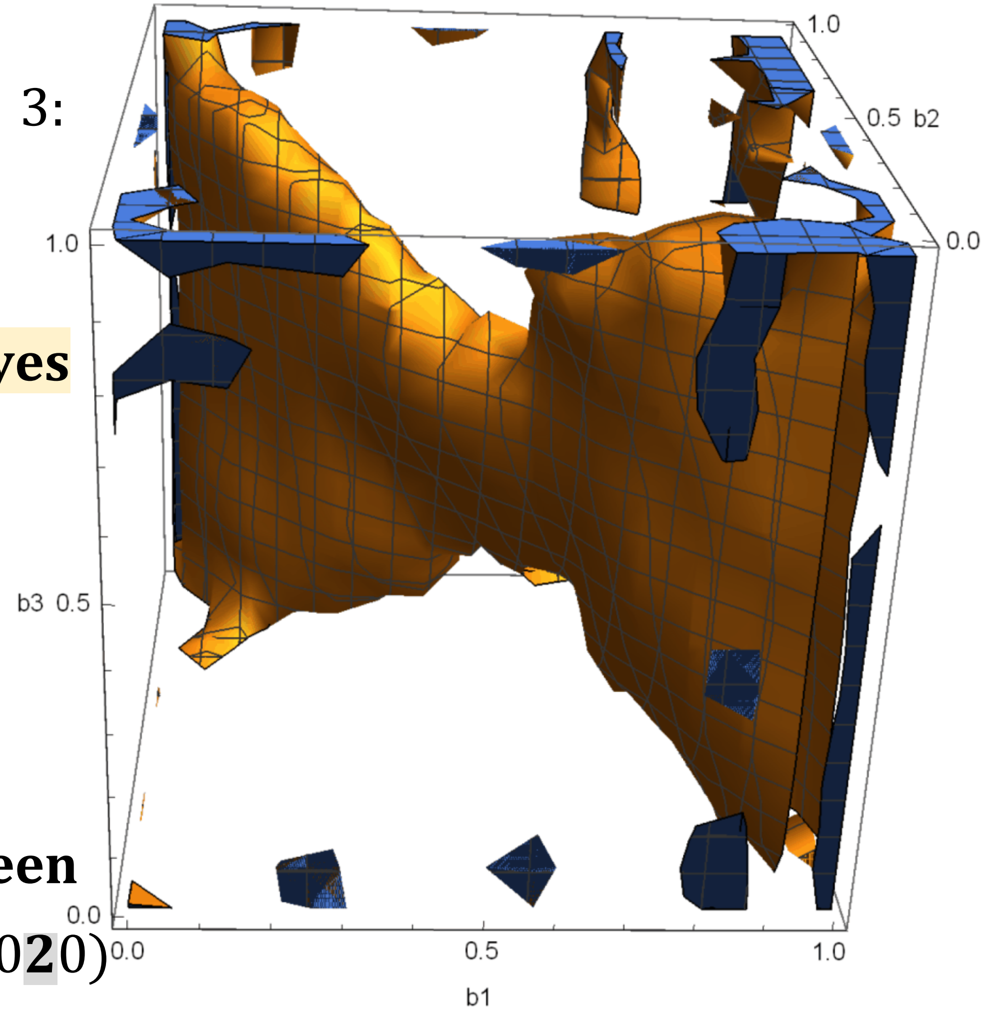
Hierarchical correlation reconstruction: $\rho(\boldsymbol{x}) = \sum_j a_j \boldsymbol{f_j}(\boldsymbol{x})$

d variables, up to m-th moment, number of a_j moments: $(m+1)^d = \sum_{k=0}^{d} \binom{d}{k} m^k = \underbrace{1}_{\text{normalization}} + \overbrace{dm}^{\text{marginal}} + \overbrace{\frac{1}{2}d(d-1)m^2}^{\text{pairwise}} + \boxed{\cdots}$





ρ > 2 region: ~14% of volume, ~62% of cases:



Also $[0,1]^d$ in **higher dimensions**, e.g. $d = 3$:

$$\rho(x_1, x_2, x_3) = \sum_{\boldsymbol{j} \in B} a_{\boldsymbol{j}} \ f_{j_1}(x_1) f_{j_2}(x_2) f_{j_3}(x_3)$$

⇒ conditional distributions without Bayes

MSE estimated from dataset $X \subset \mathbb{R}^3$:

$$a_{\boldsymbol{j}} = \frac{1}{|X|} \sum_{\boldsymbol{x} \in X} f_{j_1}(x_1) f_{j_2}(x_2) f_{j_3}(x_3)$$

For considered **statistical dependencies:**

basis B of **considered mixed moments**

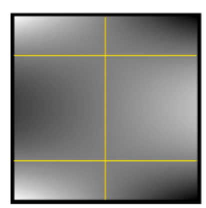
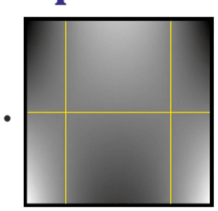
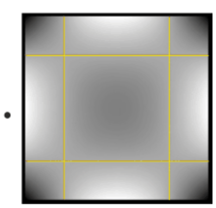
$a_{\boldsymbol{j}}$ describes e.g. **variance-variance between**

hierarchical e.g. $d = 9$: $B \ni \boldsymbol{j} = (000200020)$

$a_{\boldsymbol{j}}$ = **average** of $f_{j_1}(x_1) \ldots f_{j_d}(x_d)$ over dataset - entire or:

- over a subset for **missing data** - we need only $j > 0$ coordinates as $f_0 = 1$
- $a_{\boldsymbol{j}}^{t+1} = \lambda a_{\boldsymbol{j}} + (1 - \lambda) f_{\boldsymbol{j}}(\boldsymbol{x})$ parameter evolution for **nonstationary time**

independent ~ correlation coef. **further statistical dependencies** var-var

pair-wise joint density ≈  $+ a_{11} \cdot$  $+ a_{12} \cdot$  $+ a_{21} \cdot$  $+ a_{22} \cdot$ 

Having density model,

we can cheaply

normalize

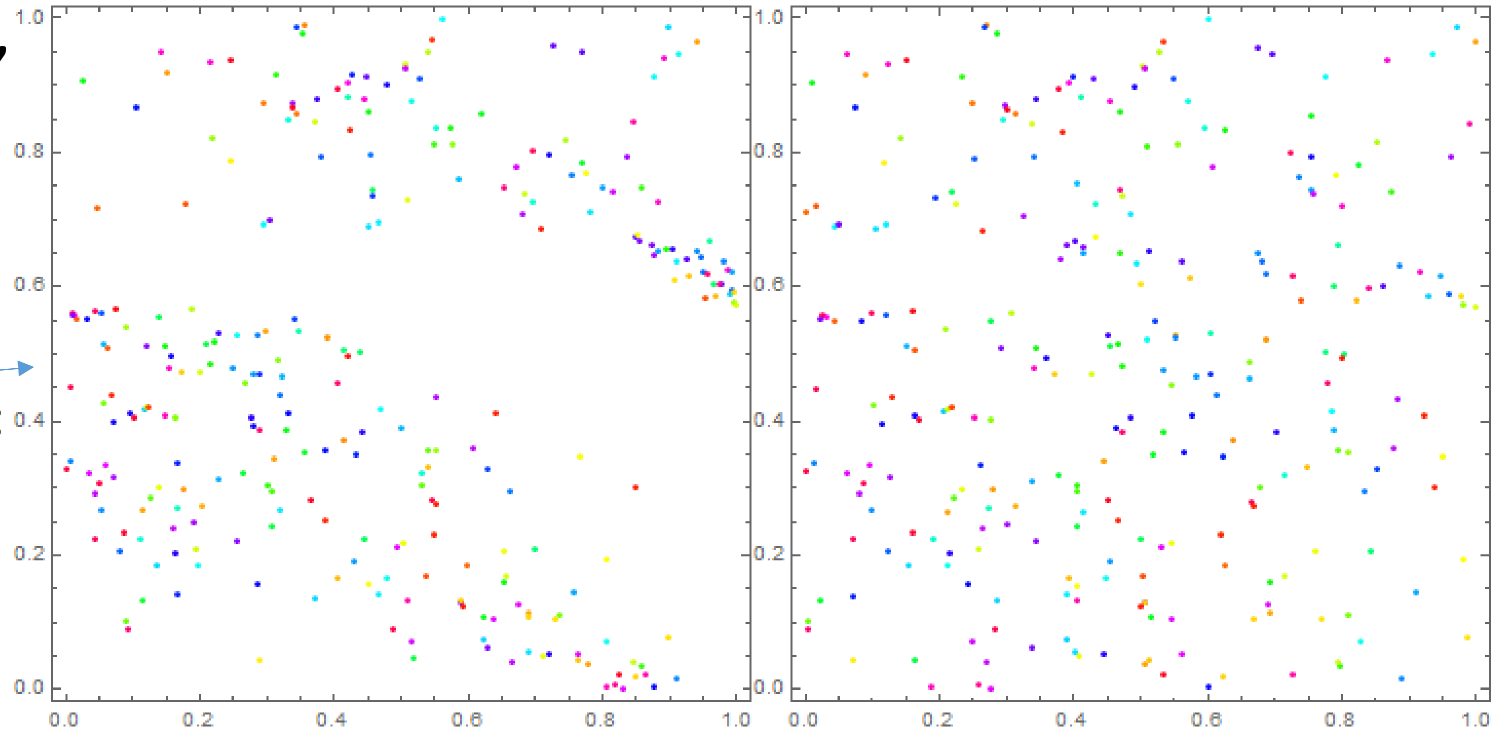
e.g. to uniform

$x \to CDF_y(x)$ by lines:

or **generate random**

sample e.g. for

Monte-Carlo methods



Generalization problem: e.g. could we avoid splitting into training + validation?

X – test, Y – training set, how to choose function basis B to maximize log-lik l ?

$$a_j = \frac{1}{|Y|}\sum_{y\in Y} f_j(y) \qquad\qquad \rho(x) = \sum_{j\in B} a_j\, f_j(x) = \frac{1}{|Y|}\sum_{j\in B}\sum_{y\in Y} f_j(y) f_j(x)$$

$$l = \frac{1}{|X|}\sum_{x\in X} \ln\left(1 + \sum_{j\in B^+} a_j f_j(x)\right)$$

can we ask separately for j about including in B?

Assume training and test set have the same statistics, e.g. value, variance for a_j ...

Economists: ~ copula theory plots

Guess one from usually **single-parameter:**

2D complex formula, tough to estimate

For more variables build tree ("vine") …

HCR – agnostic,

any # of param.

Between any pairs,

also triples

and more variables

Cheap to use,

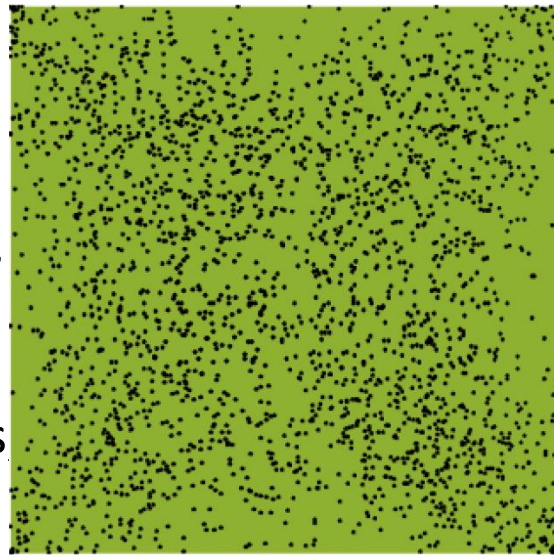
Cheap to estimate

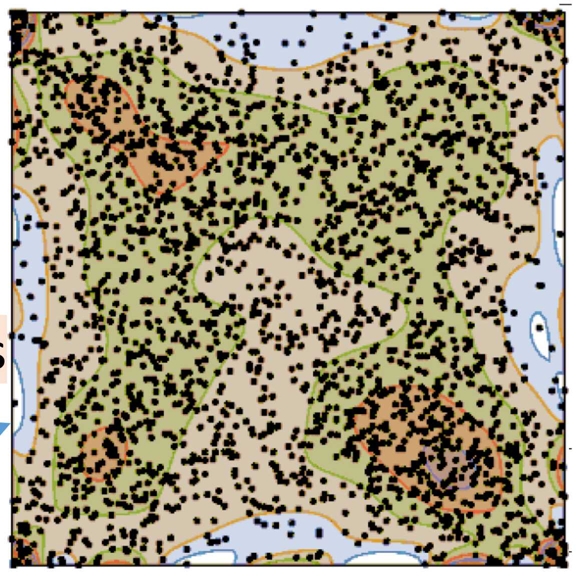
also to adapt,

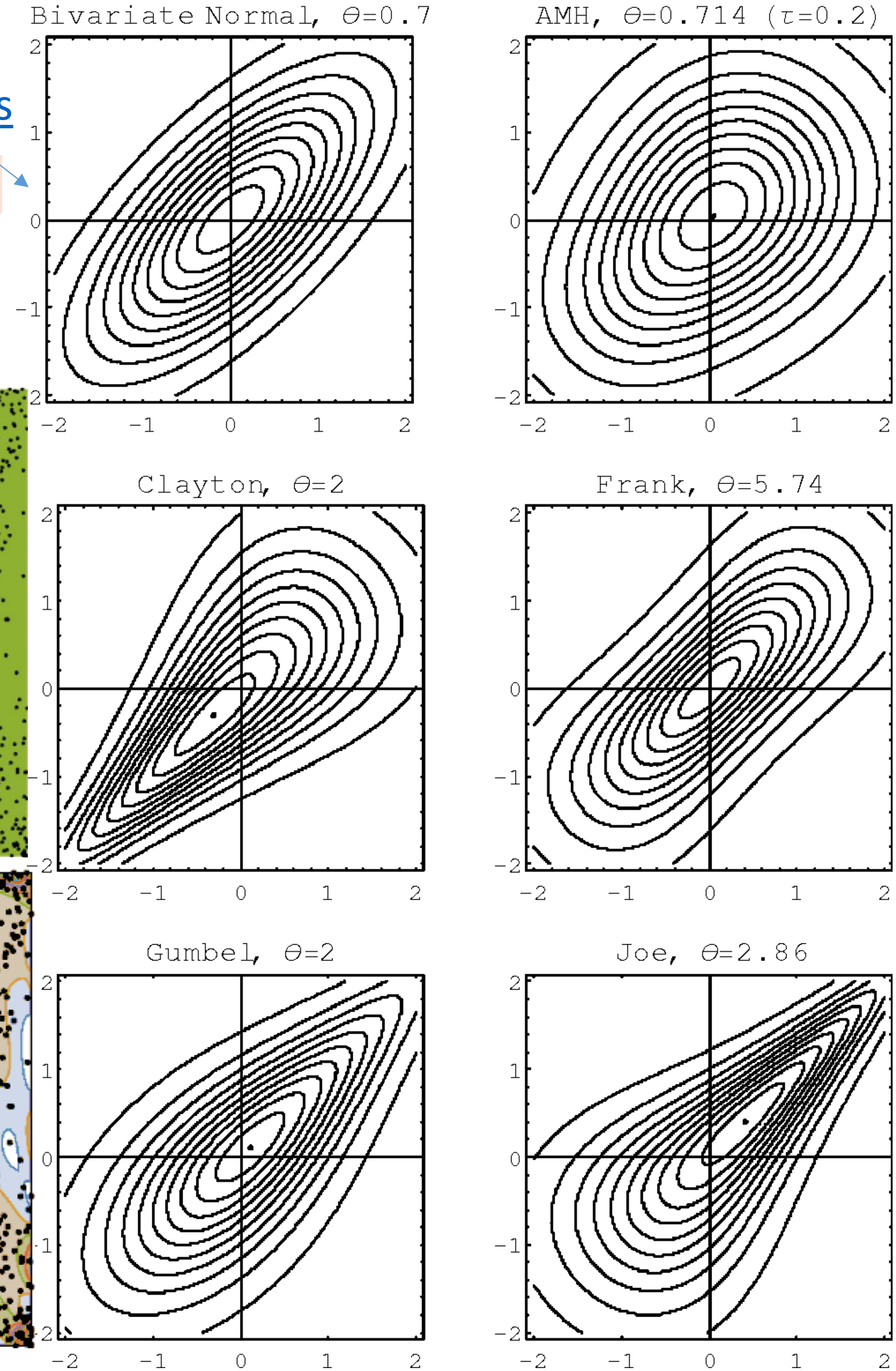
missing data

but $\rho < 0$ happens

$m = 9$, 81 param.







Predict value spread

(**bid-ask**, DAX) from

(price, volume, H-L)

should be diagonal

AMI, HLR – noise

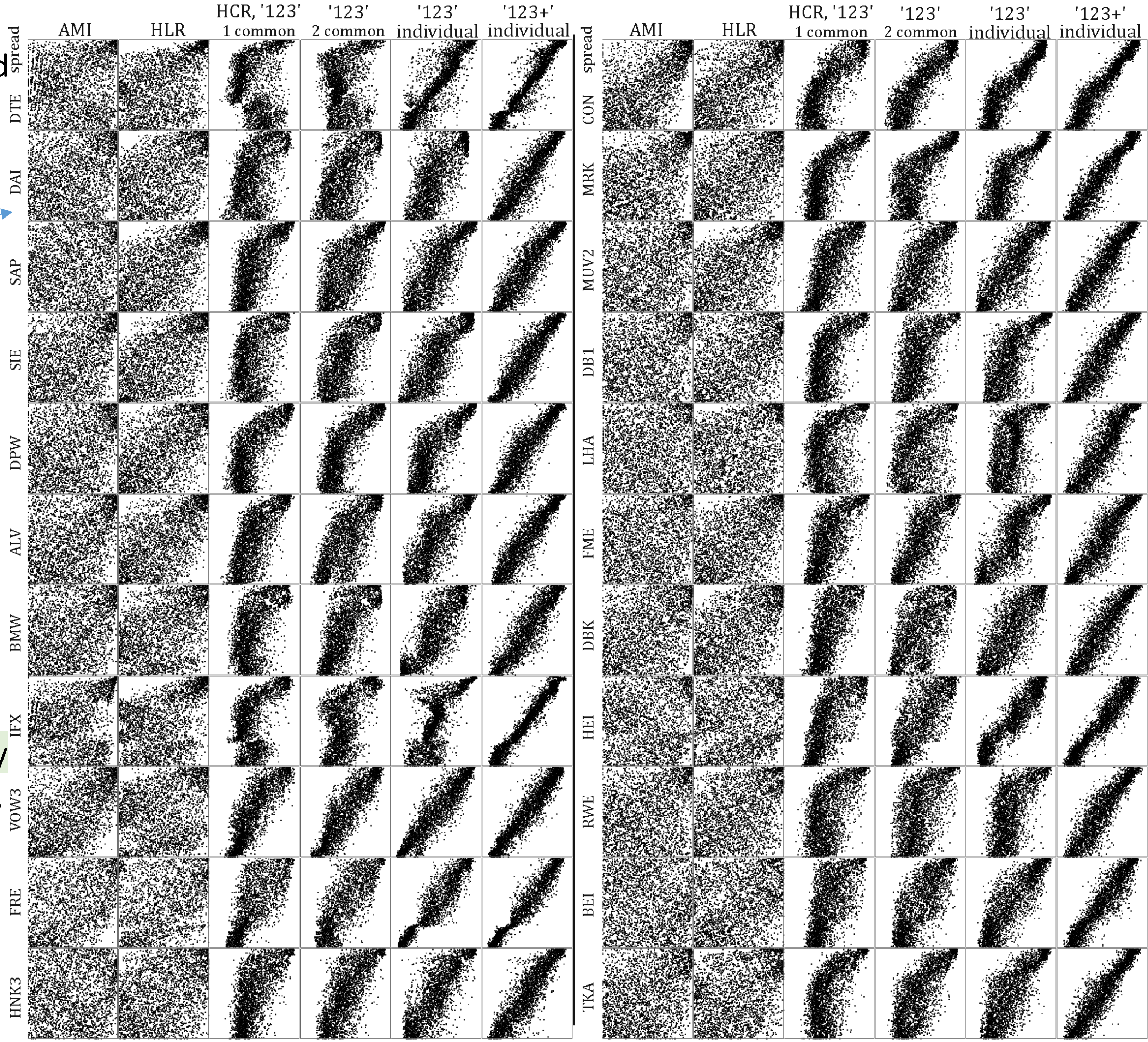
HCR – can handle

predicting density

→ expected value

[aXiv:1911.02361](#)

[Stat. in Transition](#)



Density: additional

variance:uncertainty

skewness, kurtosis…

find quantiles,

Monte Carlo rand.,

Further nonlinear f

$f(E(X)) \neq E(f(X))$

Least-squares linear regression to:

1) predict value as linear combination

2) **HCR**: predict first few moments

each separately as linear combination

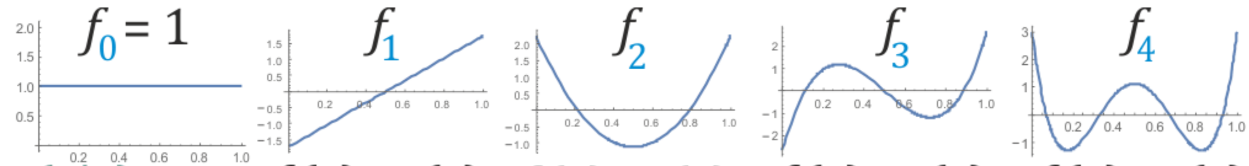
then combine into predicted density.

$$\tilde{\rho}(y|x) = \sum_j f_j(y) \sum_k \beta_{jk} f_k(x)$$

$$\rho(y|x) = \max(\tilde{\rho}(y|x), 0.03)\,/N$$

Examples

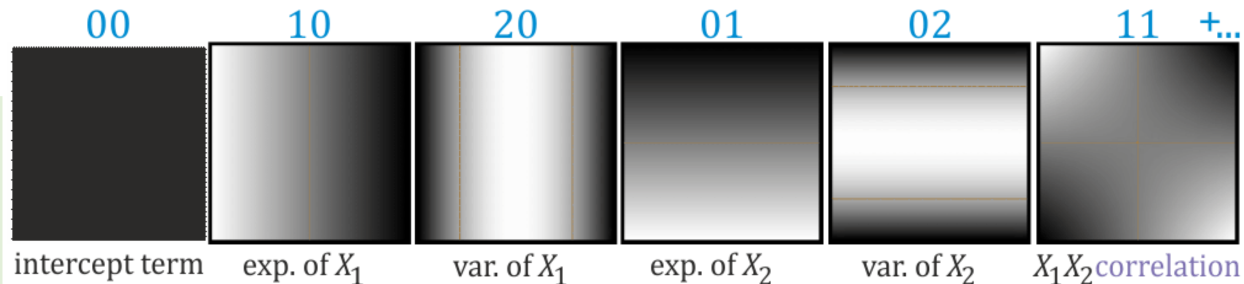
normalization



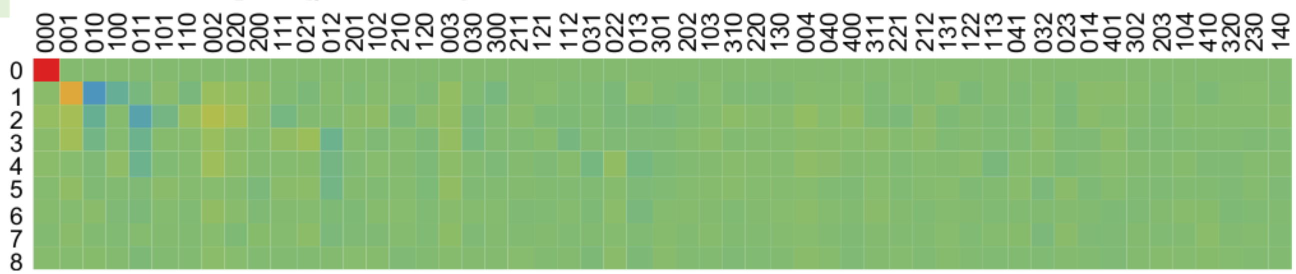
$$\rho(y|x) = 1 + f_1(y)\cdot a_1(x) + f_2(y)\cdot a_2(x) + f_3(y)\cdot a_3(x) + f_4(y)\cdot a_4(x)$$

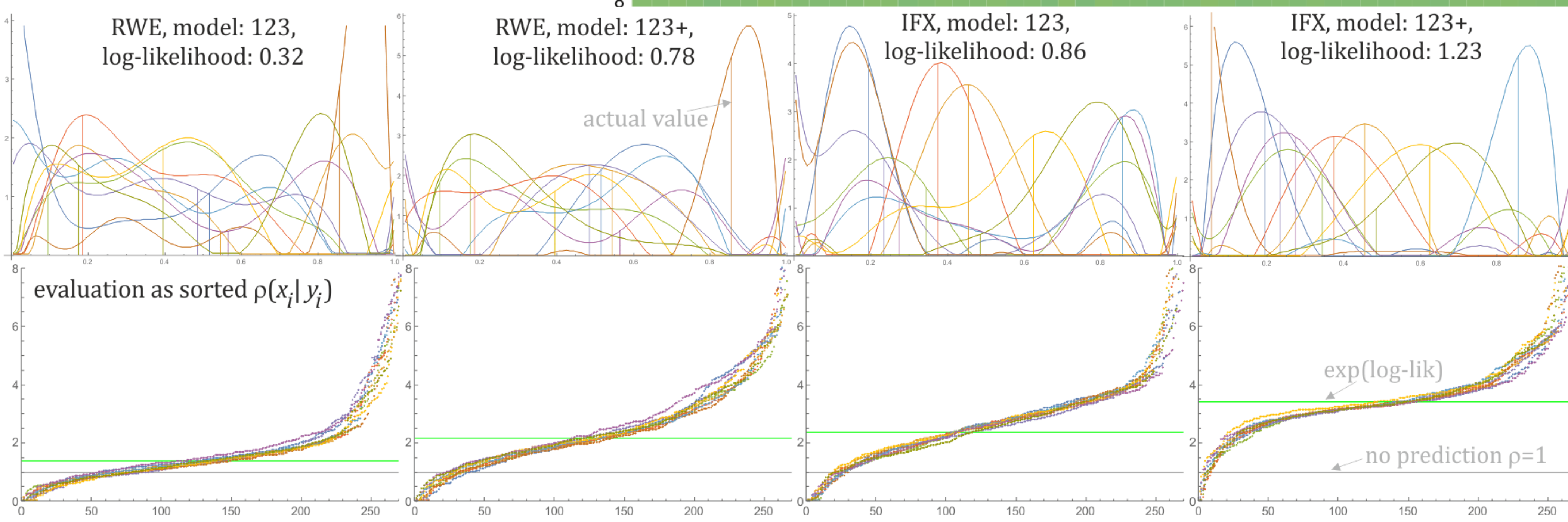
normalization ~exp. value ~variance ~skewness ~kurtosis

each modeled separately using linear regression of mixed moments of x:

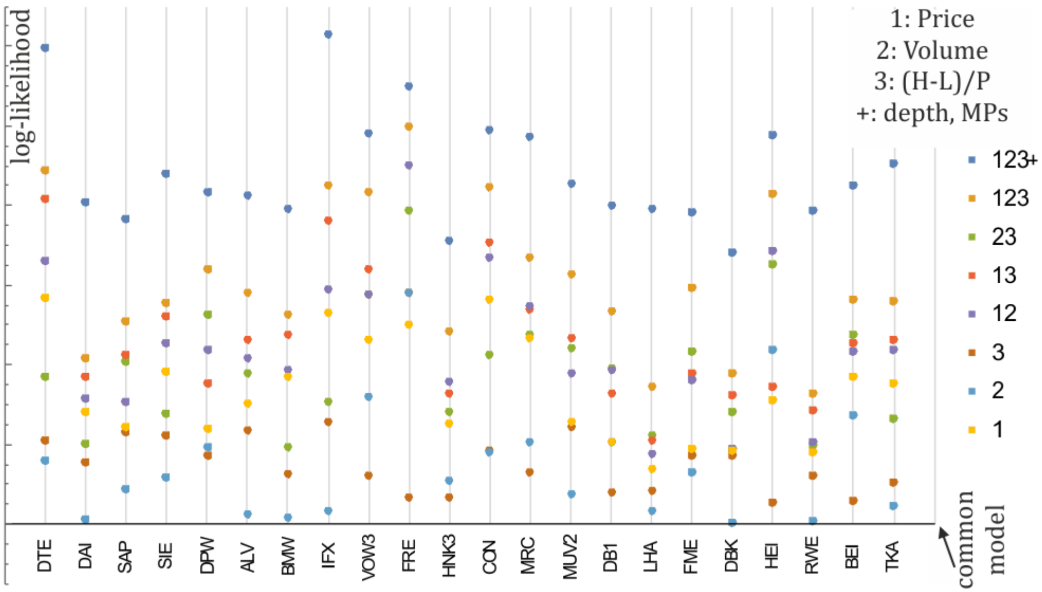


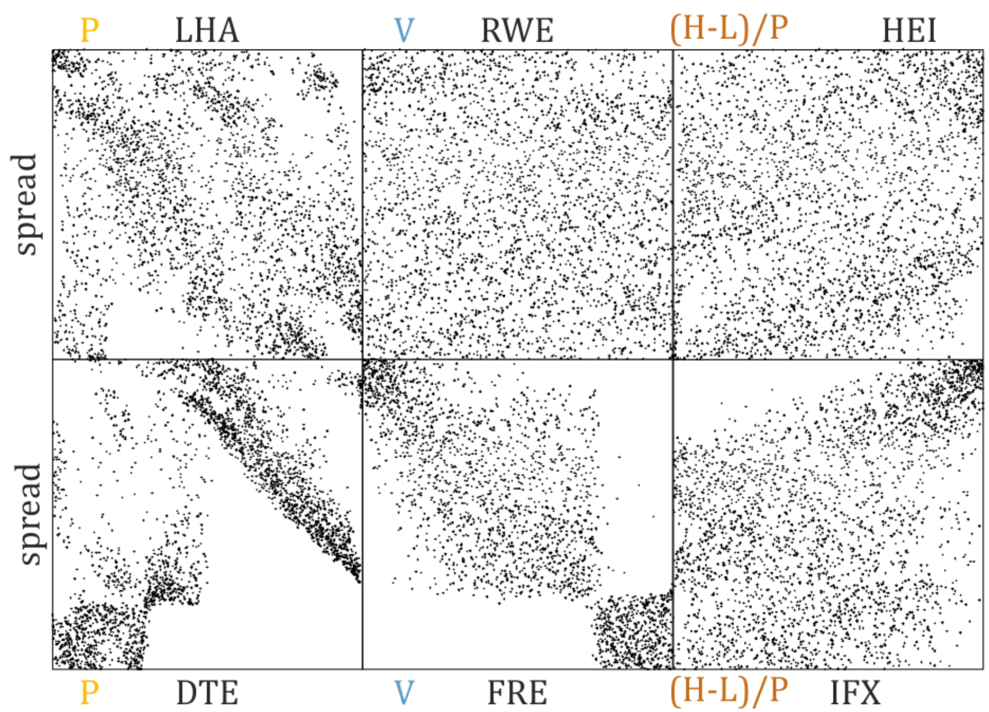
model example (β matrix): 8 moments of Y from 53 mixed moments of x

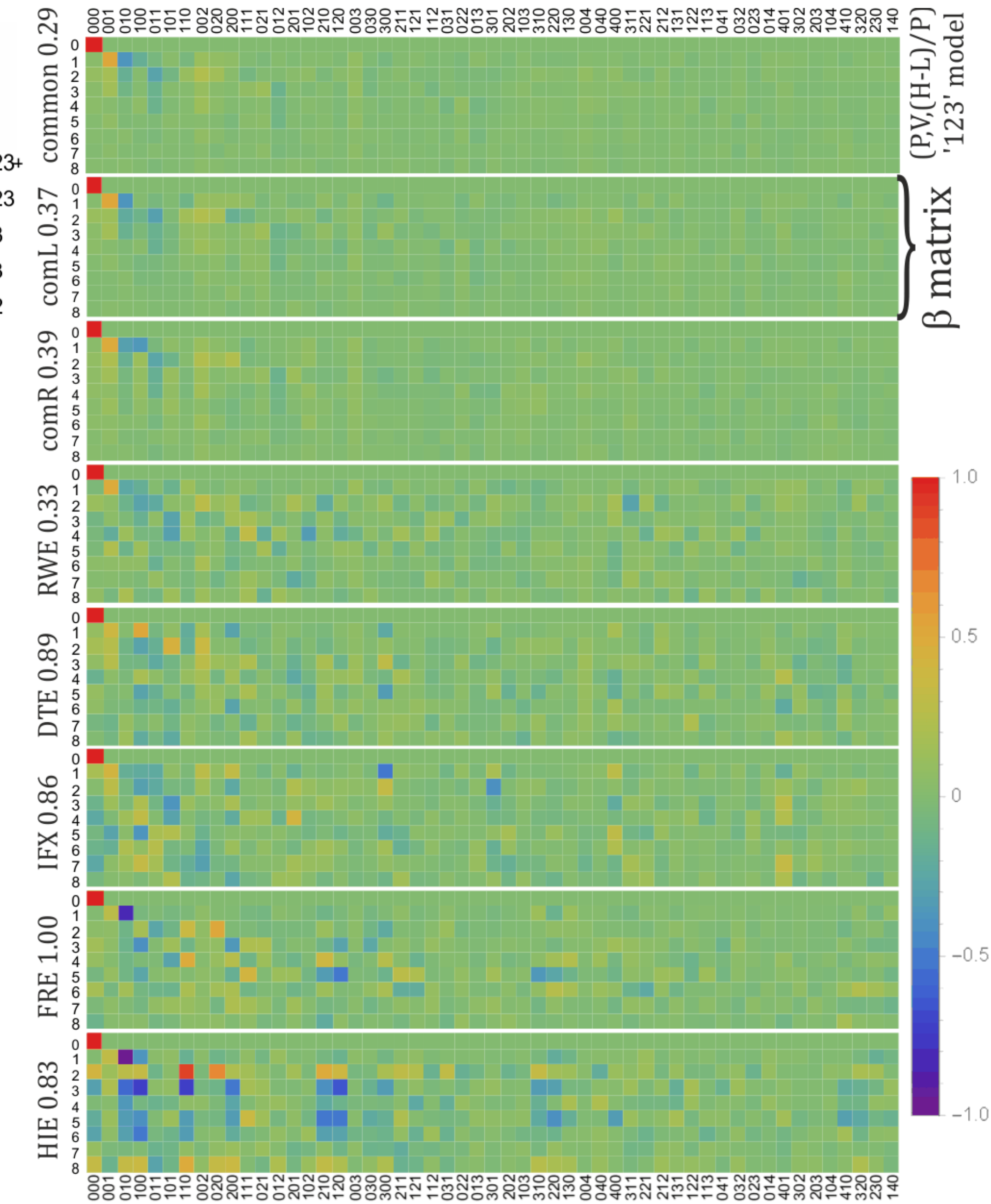




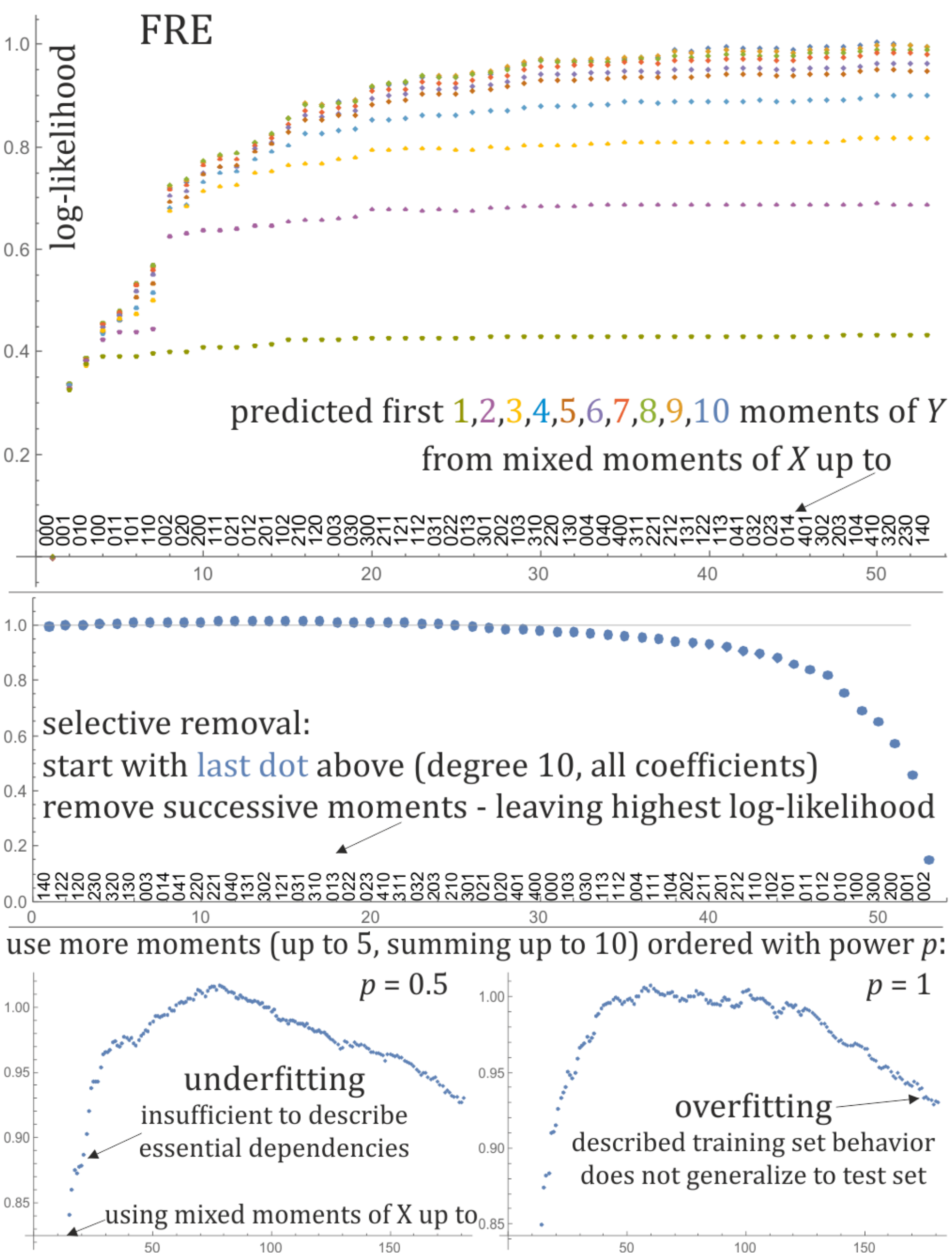
Large differences between companies – individual models give much better evaluation



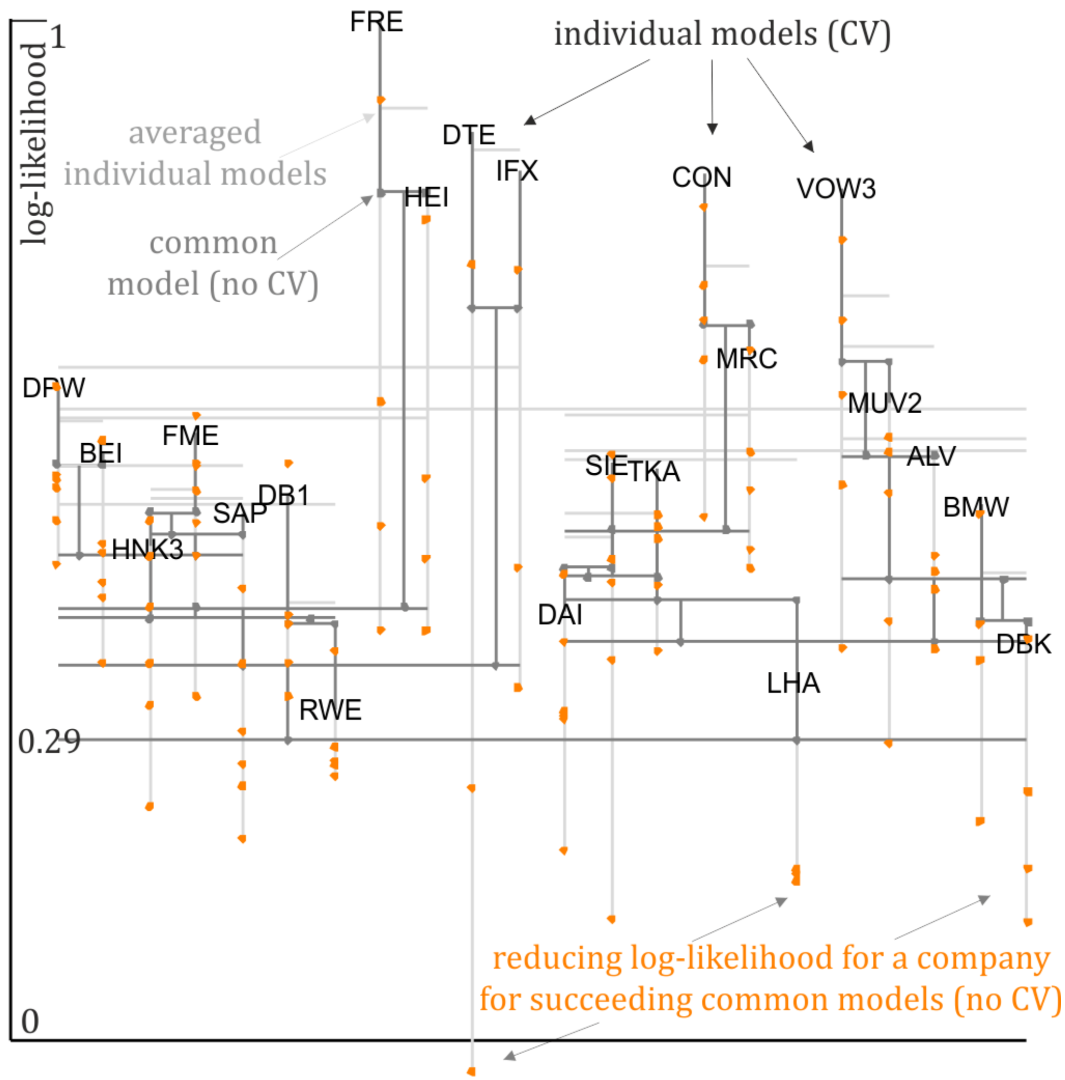




Choosing model size: predict ≈ 8 moments basis of mixed moments – difficult problem



Universality – searching for common models with lowest evaluation loss



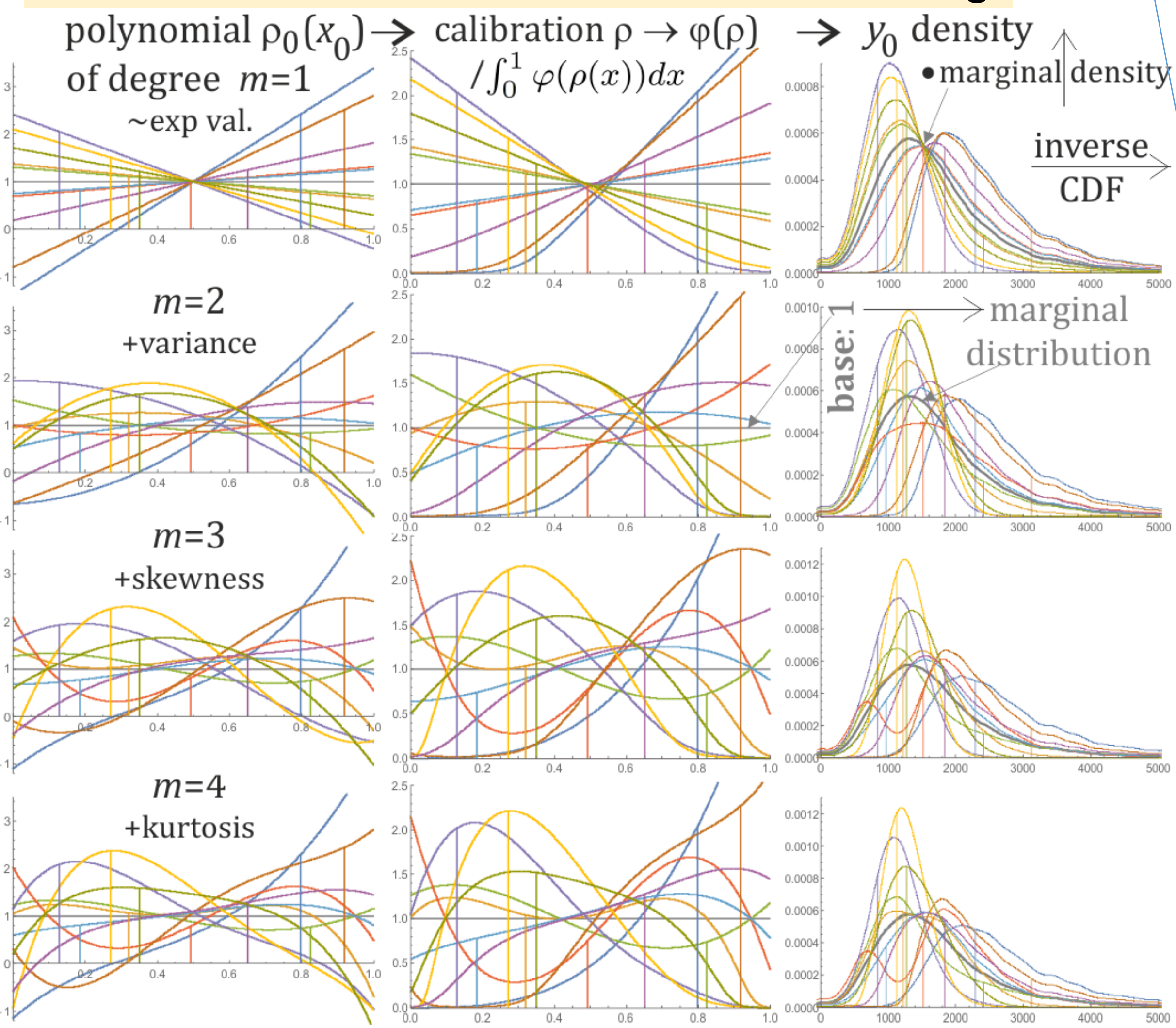
37k households GUS data (arXiv:1812.08040, ICOAE)

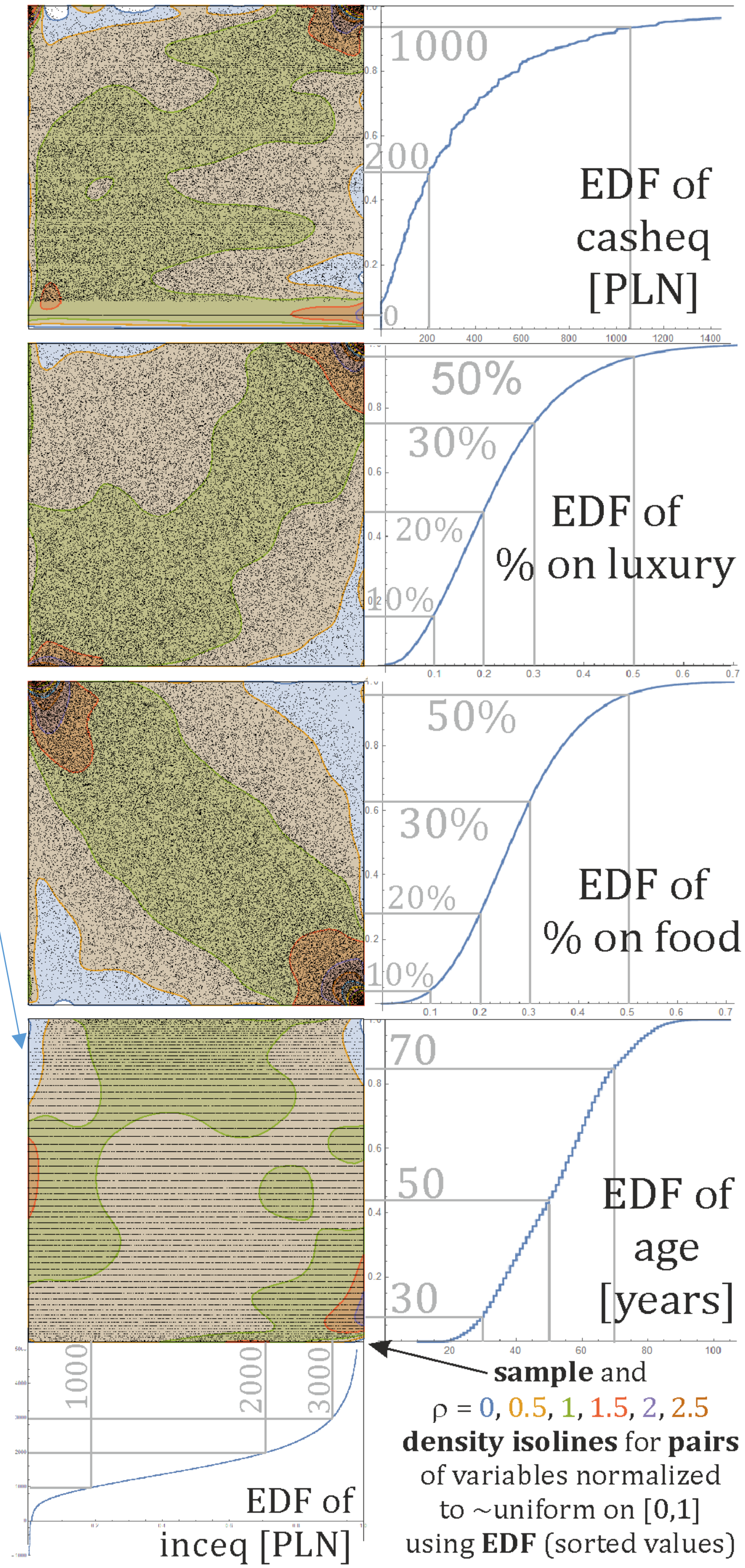
Find **conditional distribution** of equivalent income from **31 discrete variables** and **4 continuous** normalized to uniform on [0,1] by sorting (EDF):

1/2 is median, 10% is 10% of population

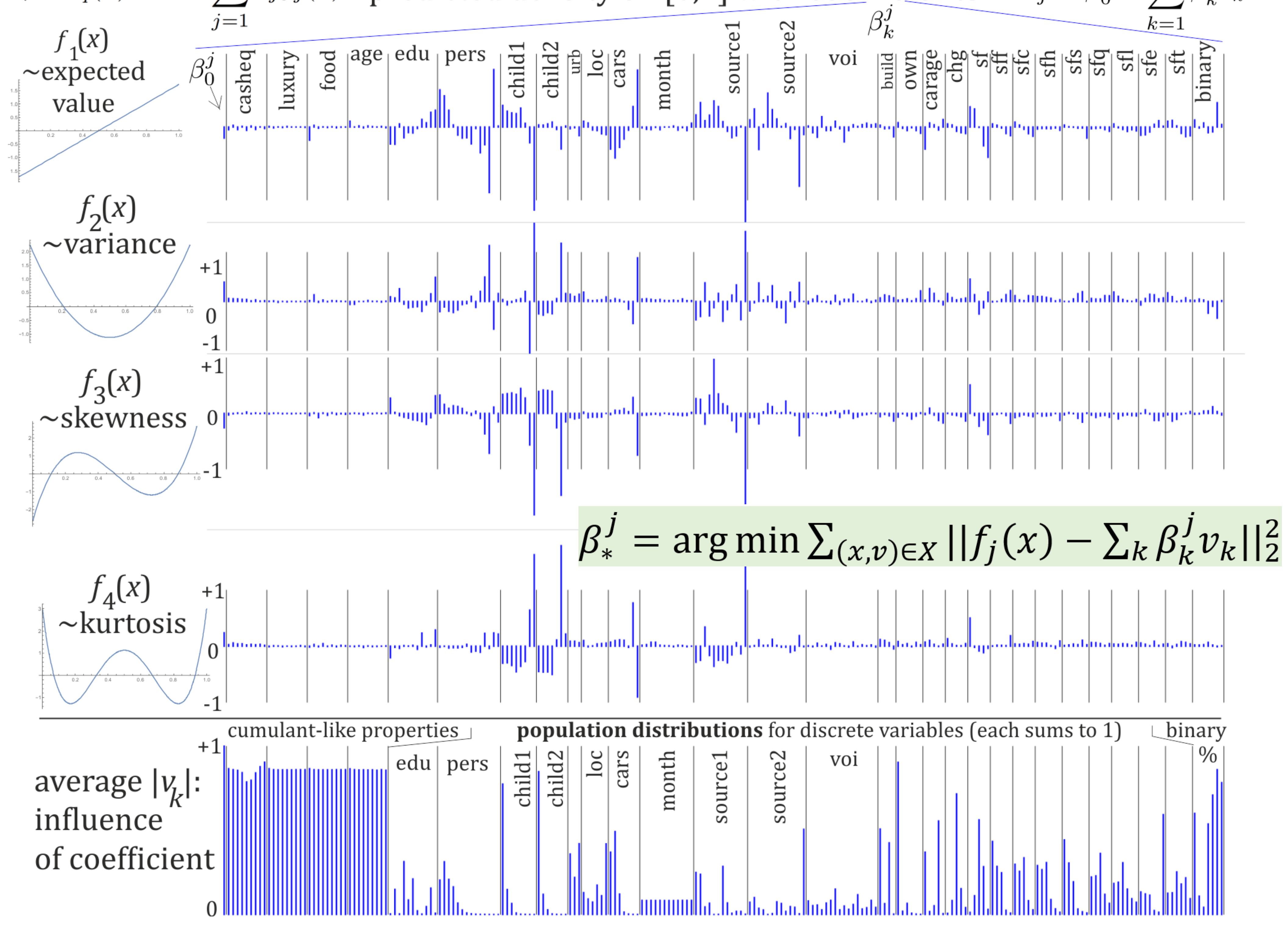
Credibility evaluation e.g. 70 years old close to median

How to model it with standard machine learning?





$\rho_{inceq}(x) = 1 + \sum_{j=1}^{m} a_j f_j(x)$ predicted density on [0,1] with **coefficients:** $a_j = \beta_0^j + \sum_{k=1}^{222} \beta_k^j v_k$



$$\beta_*^j = \arg\min \sum_{(x,v)\in X} ||f_j(x) - \sum_k \beta_k^j v_k||_2^2$$

LL(**log-likelihood**): average $\log_2$ of predicted density in actual value:

$m=$	0	1	2	3	**4**	5	6	7	8	9
	1	+exp.	+var.	+skew.	+kurt.					
LL	0	0.420	0.566	0.576	**0.580**	0.578	0.579	0.578	0.578	0.577
2^{LL}	1	1.338	1.480	1.490	**1.494**	1.493	1.494	1.493	1.493	1.492

Random 75% to train, 25% **evaluation**

(expected value, sqrt(var)) of predicted

Log-likelihood evaluation of 35 variables:

Relevance: from single variable Y_i

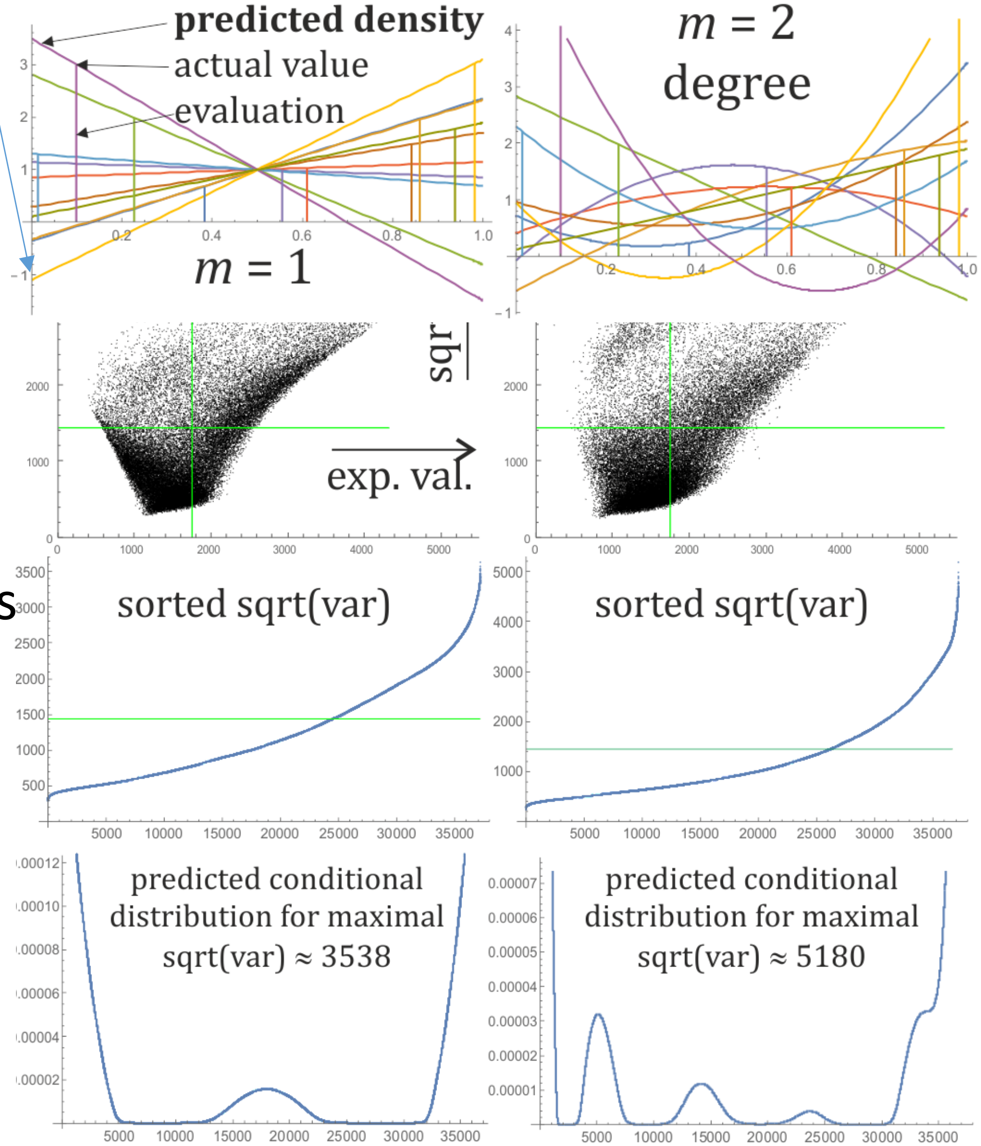
entropy: $E[\ln(\rho(x|y_i)] = -H(X|Y_i)$

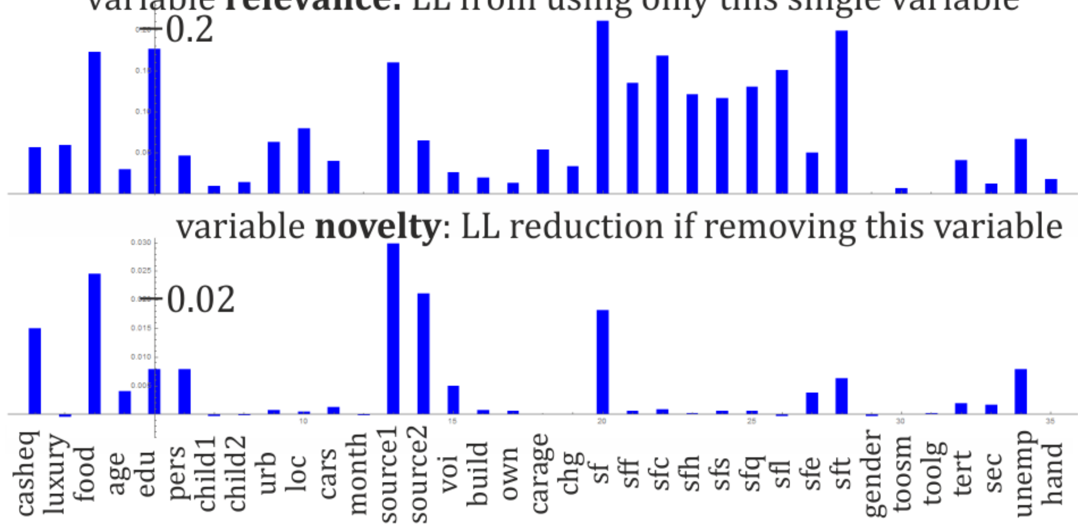
bits of information theoretic evaluation

in linear regression poor evaluation (?)

Novelty: loss if without this variable

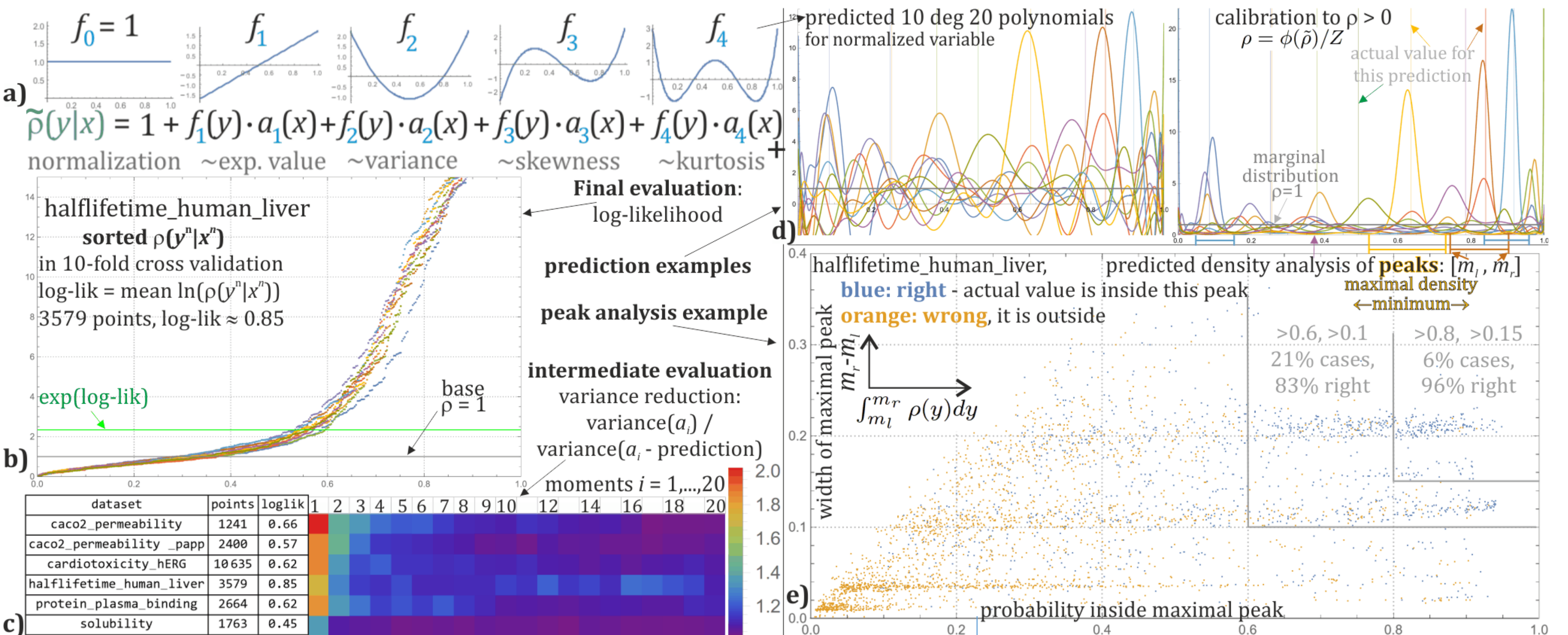
Survey design – choosing best few variables





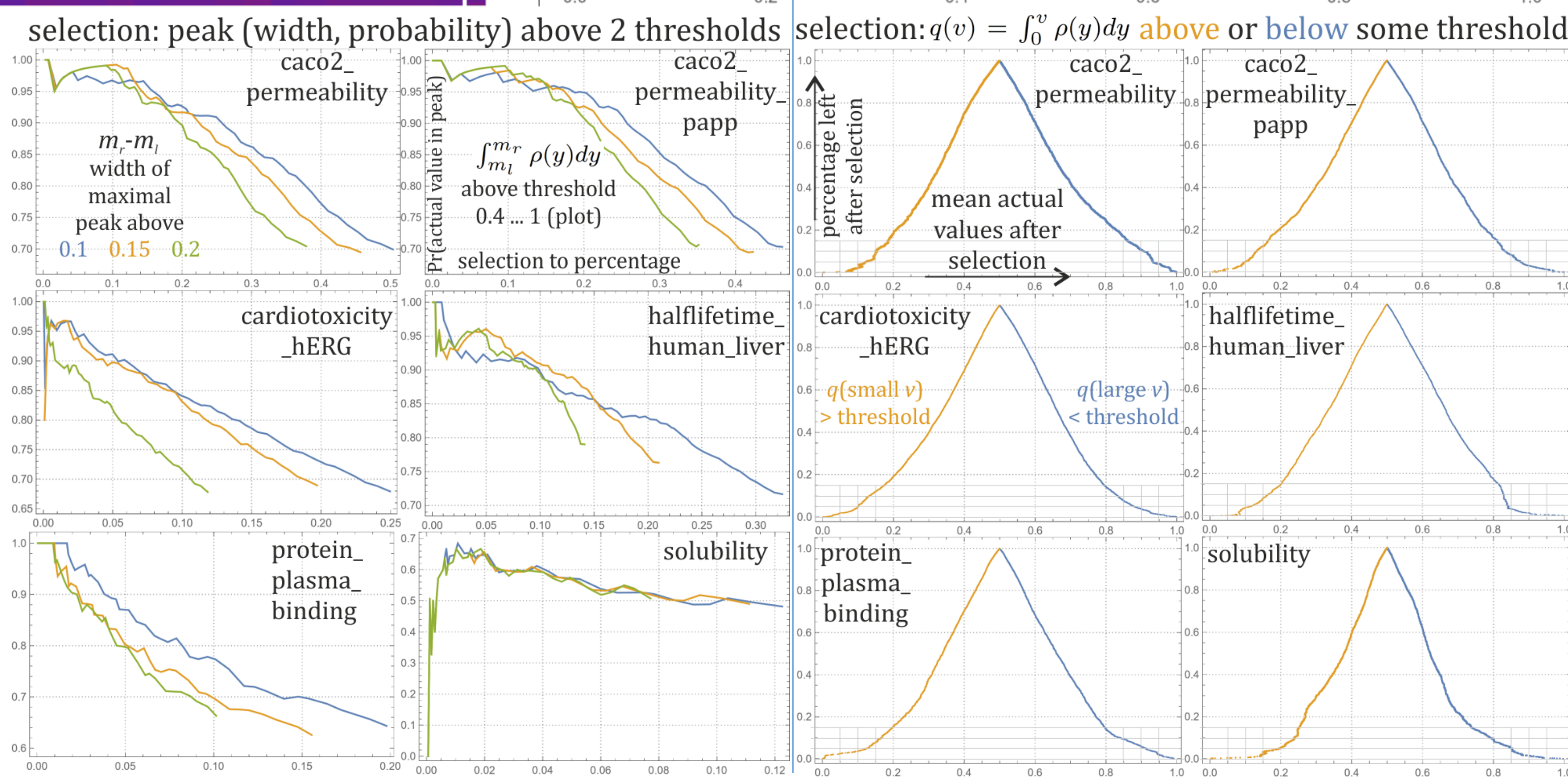
survey design: LL from chosen first few variables for m=4 (all: 0.574)

	sf	source1	food	pers	source2	edu	sft	casheq	unemp	urb
LL	0.217	0.321	0.398	0.435	0.467	0.493	0.512	0.527	0.539	0.545



dataset	points	loglik
caco2_permeability	1241	0.66
caco2_permeability _papp	2400	0.57
cardiotoxicity_hERG	10635	0.62
halflifetime_human_liver	3579	0.85
protein_plasma_binding	2664	0.62
solubility	1763	0.45

Predict probability distributions of chemical properties for virtual screening ([arXiv: 2207.11174](https://arxiv.org/abs/2207.11174)) [Molecular Diversity]() **from 4860 Klek fingerprints**



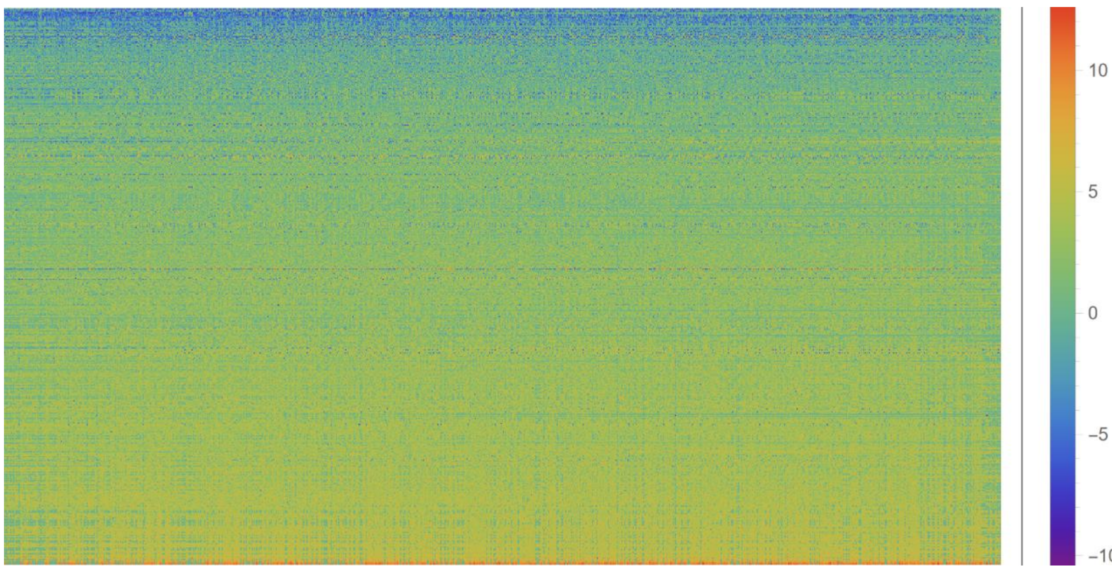
Drug selection to test (arXiv: 2109.06211)

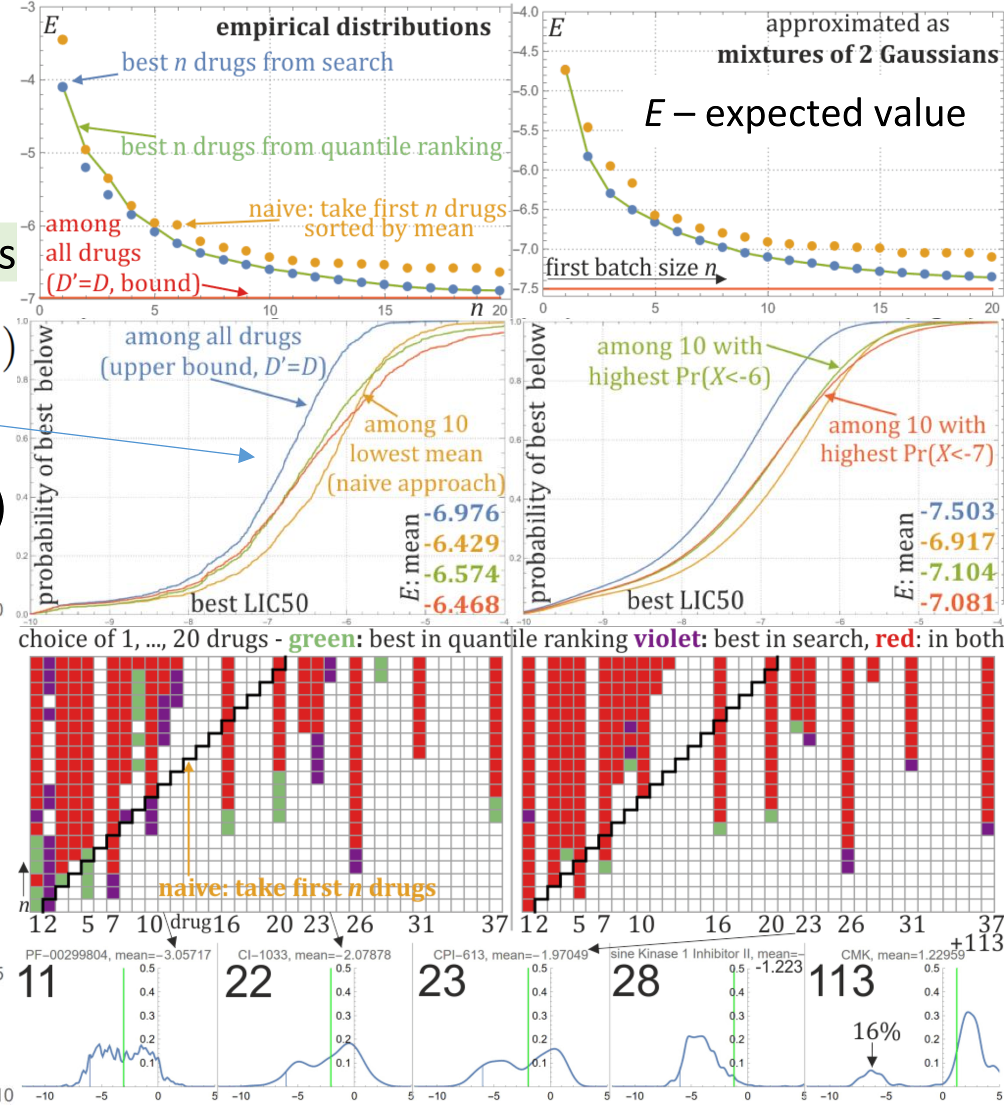
extreme statistics optimization – select for not average, but the best of e.g. n = 10 chosen drugs

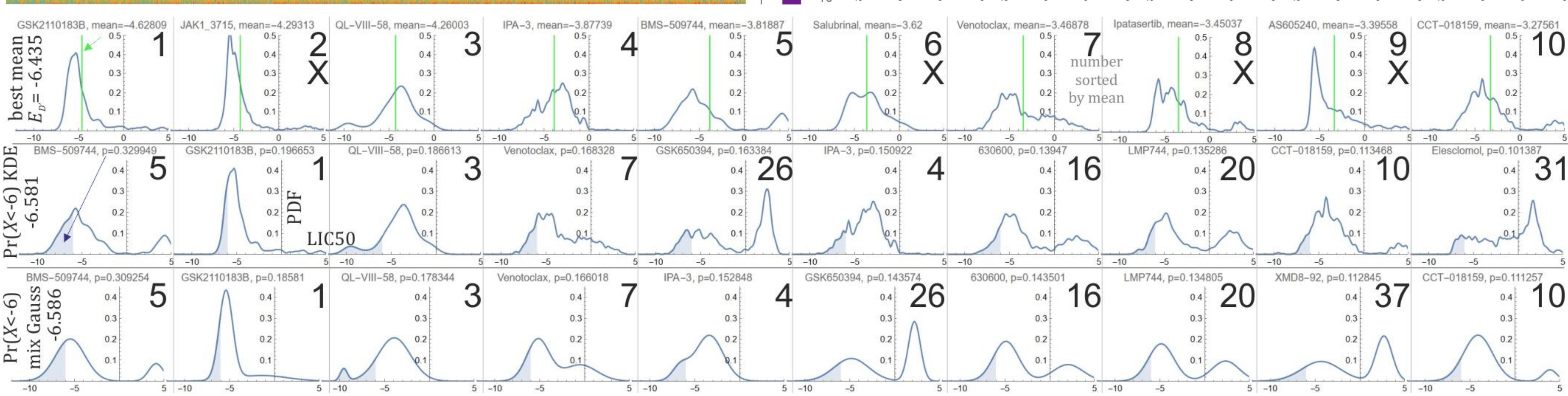
$$C_{D'}(x)=\Pr\left(\left(\min_{i\in D'} X_i\right)\le x\right)=1-\Pi_{i\in D'}(1-C_i(x))$$

ln(IC50): of the needed concentration

GDSC data: **537** drugs x **962** cell lines (~10% missing)





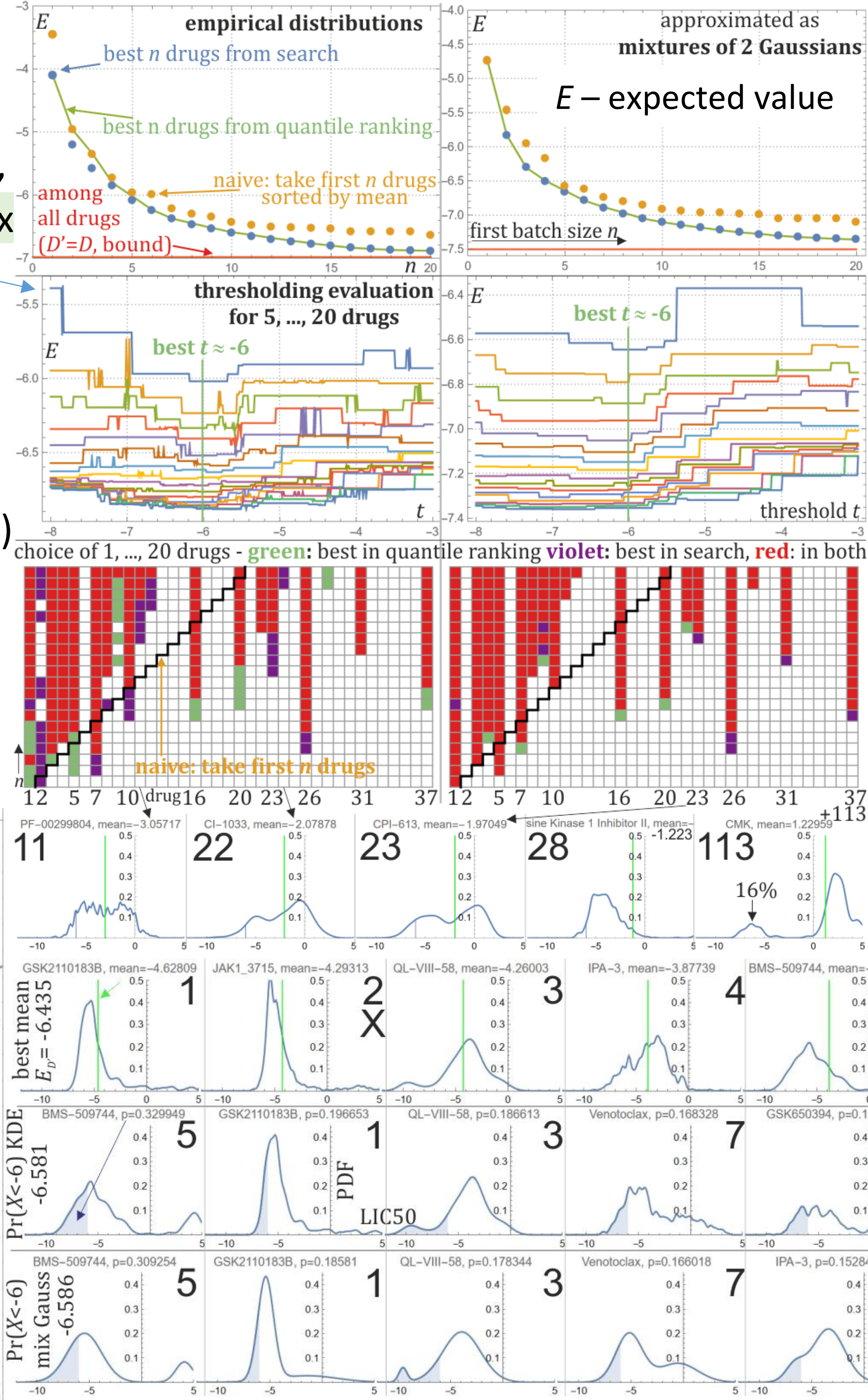


How to optimize for **the lowest values** (not mean)?

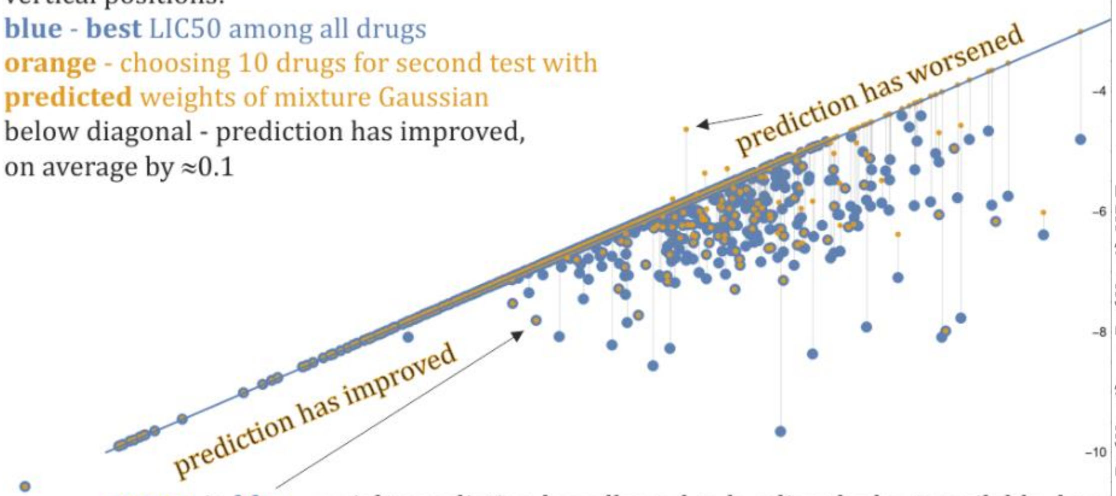
- search enlarging e.g. 1000 most promising subsets,
- take best for some (which?) quantile: cheap approx
- are there some better ways? Literature?

Then **prediction of probability distribution** (data?)
(based on tissue type, genetic … previous tests)
As mixture of A-B Gaussians (binomial: on-off gene?)
predict w – probability of being in left Gaussian (A)
regression from $\Pr(A|X=x) = \frac{w\,\rho_A(x)}{w\,\rho_A(x)+(1-w)\rho_B(x)}$

Finally: normalize + predict polynomials (HCR)?



after the first test of 10 drugs (optimized t=-6), **evaluation of second test of 10 drugs**
each of 962 points represents cell line
horizontal position: lowest LIC50 if using fixed mixture Gaussians for each drug
vertical positions:
blue - **best** LIC50 among all drugs
orange - choosing 10 drugs for second test with **predicted** weights of mixture Gaussian
below diagonal - prediction has improved,
on average by ≈0.1



orange in **blue** - weight prediction has allowed to localize the best available drug

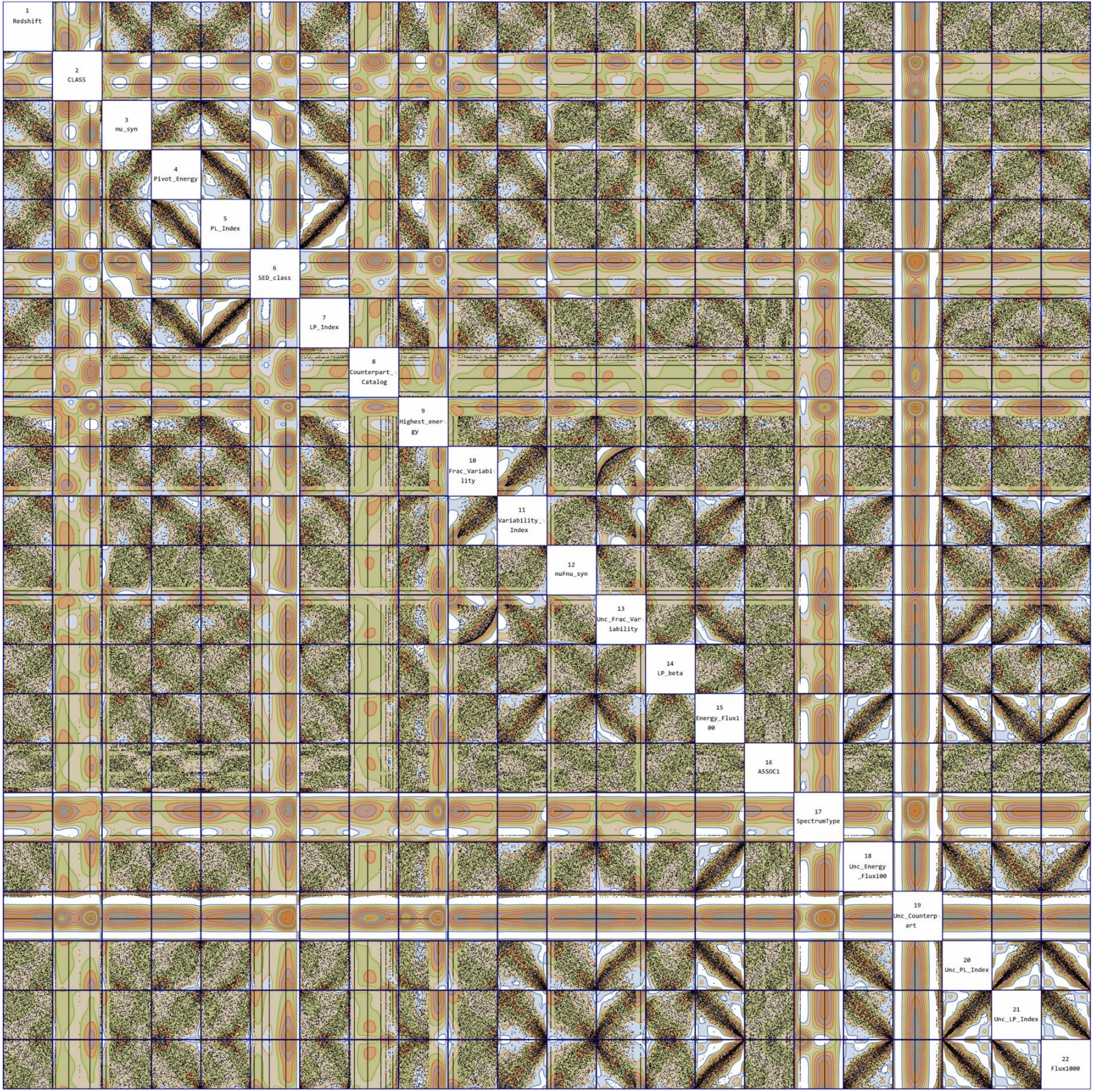
Of probability distribution of **redshift** of **Active Galactic Nuclei**

from
21 variables:
discrete,
continuous,
combined

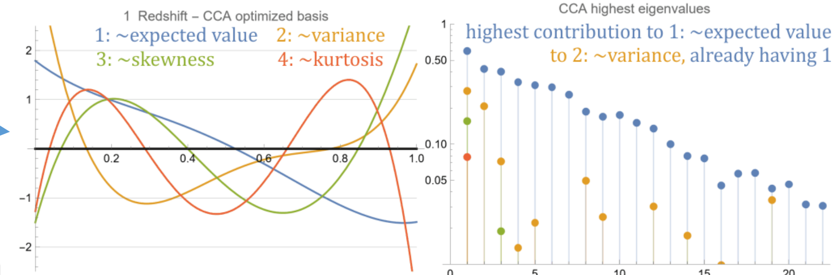
mostly describing spectrum

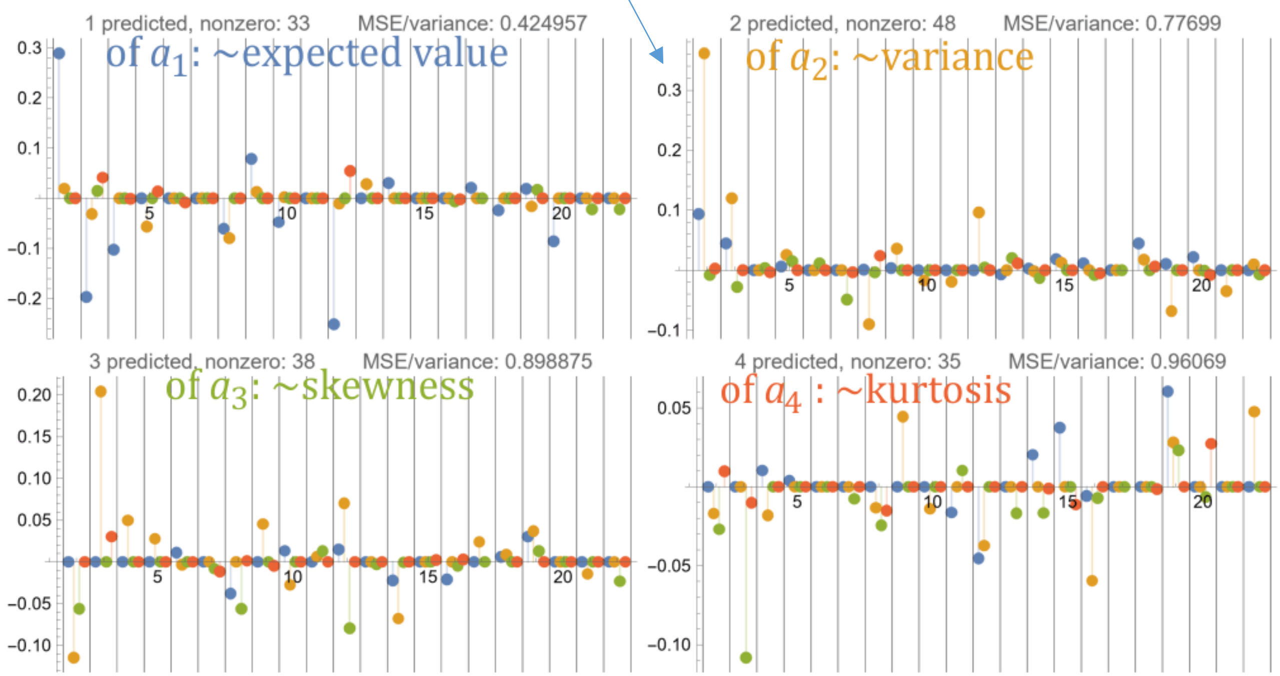
[arXiv: 2206.06194]

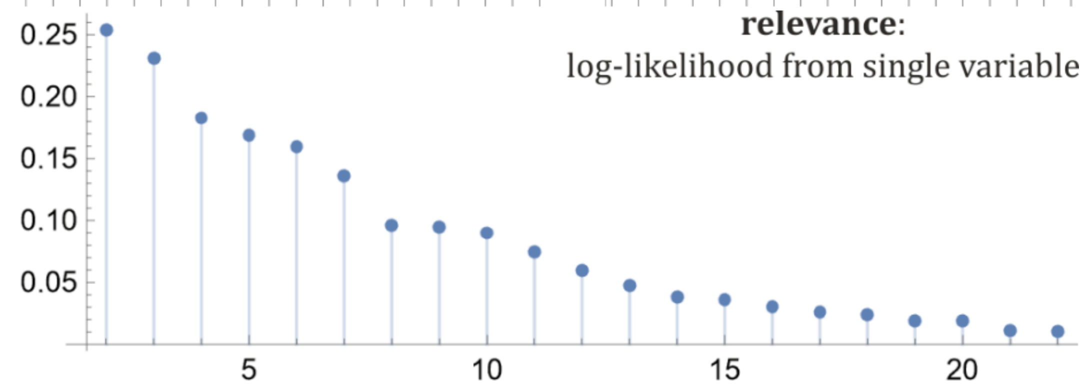
[MNRAS 2024]

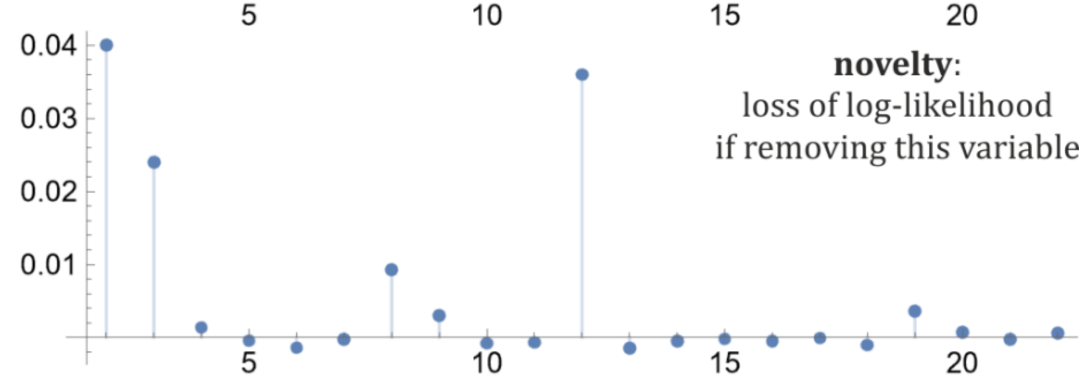


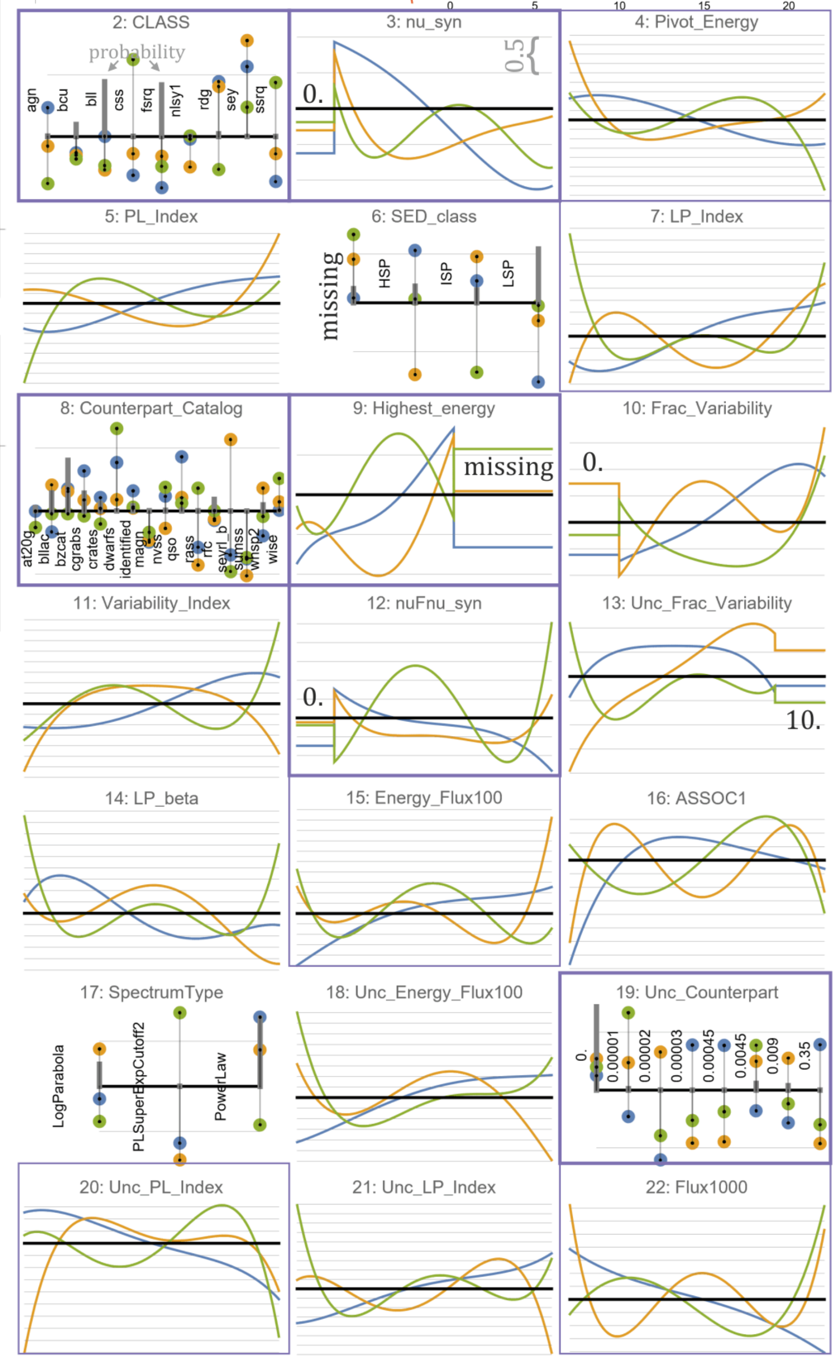
Canonical correlation analysis to optimize features for 21+1 variables, model (**l1 regular.**), var. evaluation











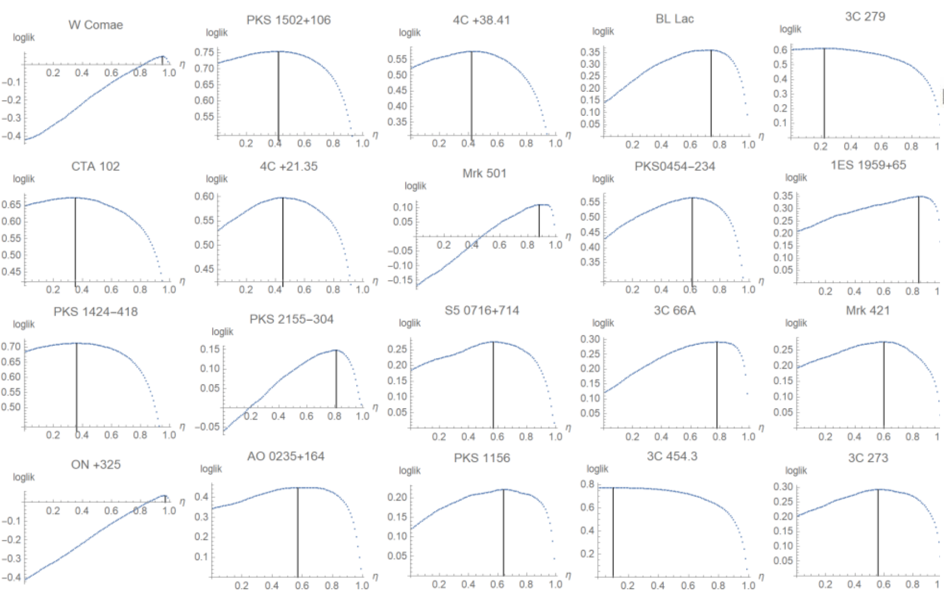
Non-stationarity analysis for blazars
https://arxiv.org/pdf/2005.14040
[MNRAS (Royal Astronomical Society)](MNRAS)

Evolving density for normalized

$$\rho_t(x) = \sum_{j \in B} a_j(t)\ f_j(x)$$

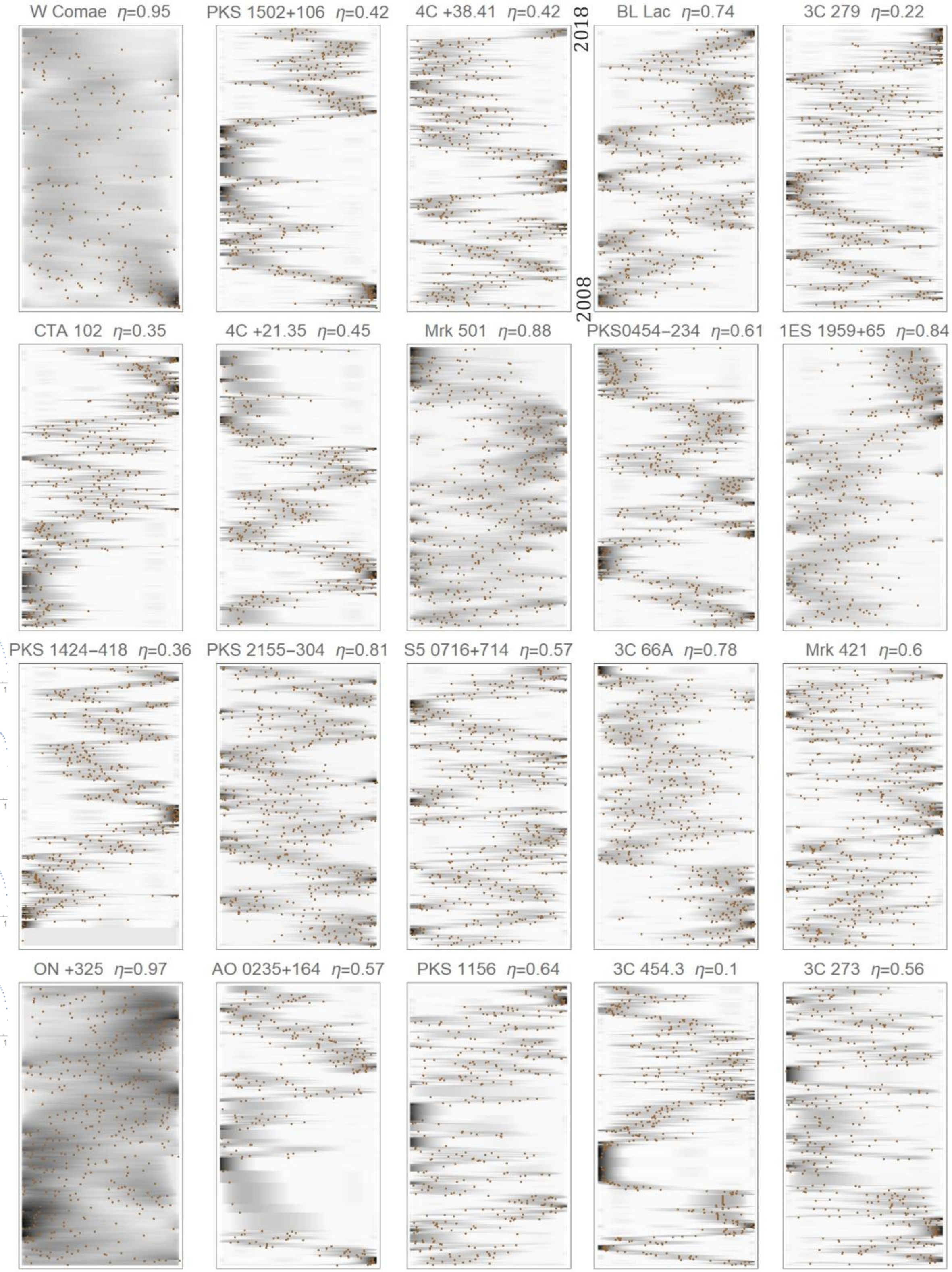
$$a_j(t+1) = a_j(t) + (1-\eta)\left(f_j(x_t) - a_j(t)\right)$$

η to maximize log-likelihood:

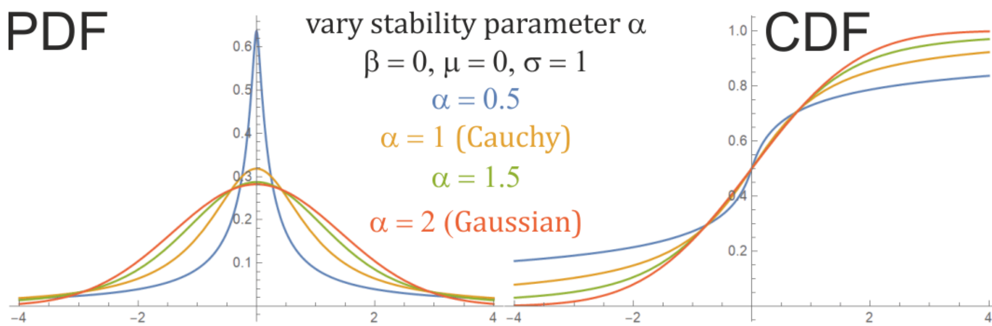


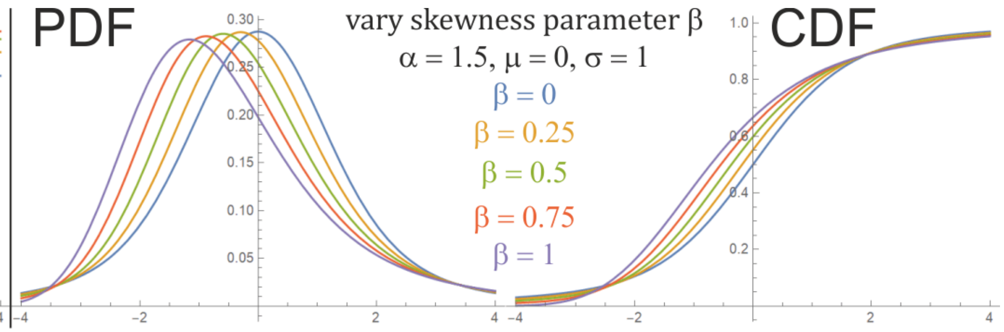
(η, loglik): nonstationarity evaluation

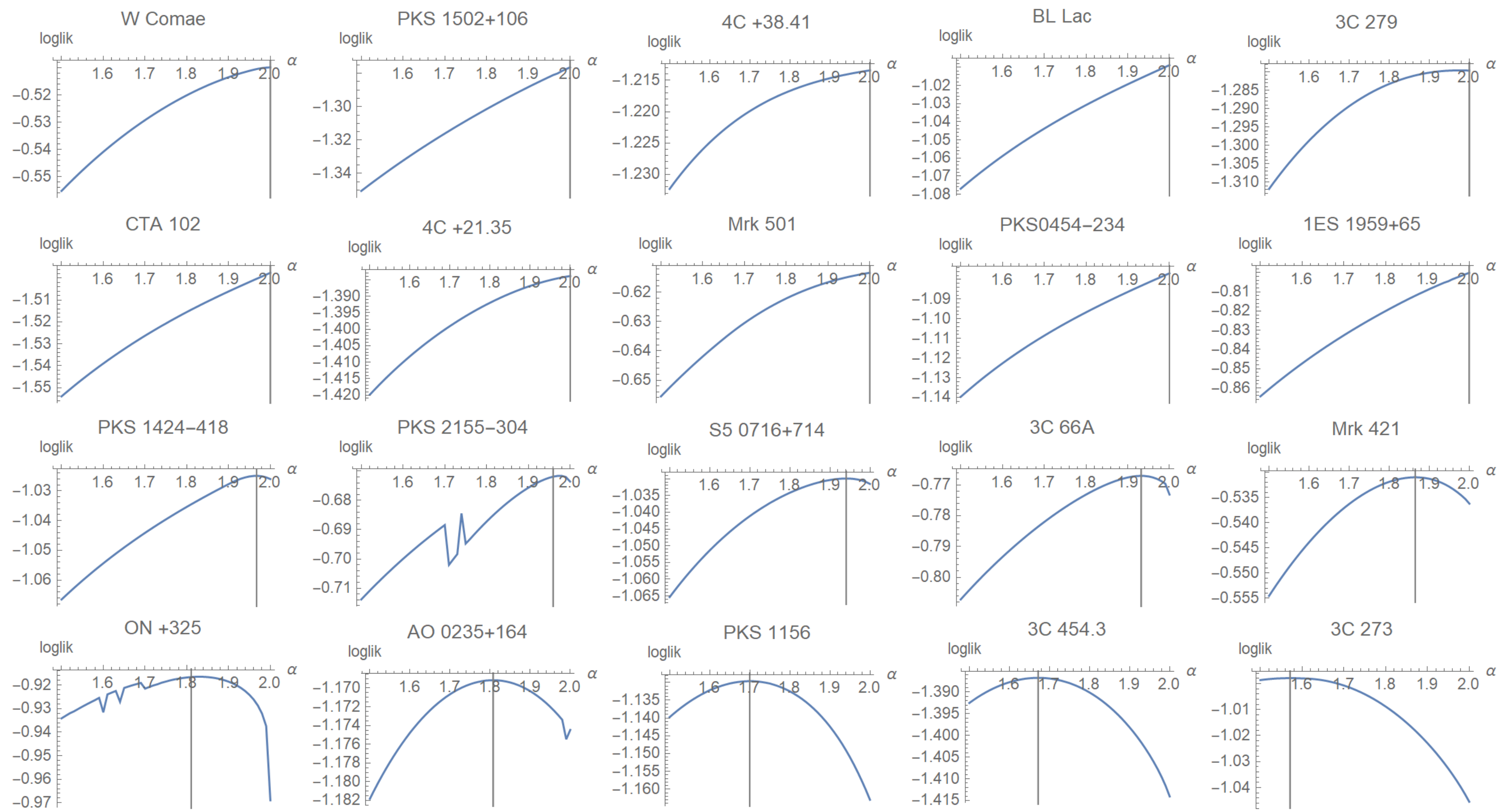
1/time, "localization"



Generalized central limit thm: sum of i.i.d. $\rho \sim |x|^{-\alpha-1}$ **infinite variance** variables lead to stable distribution, **product for log-stable here**







Multi-feature autocorrelation analysis (MNRAS):

all (y_t, y_{t+l}) pairs

static 2D for each l

$\rho(x) = \sum_{j\in B} a_j \; f_j(x)$

$a_j = \frac{1}{|X|}\sum_{x\in X} f_j(x)$

Basis up to 4th moment

$B = \{(j,k): 0 \le j,k \le 4\}$

some $f_{jk}(y_t, y_{t+l})$

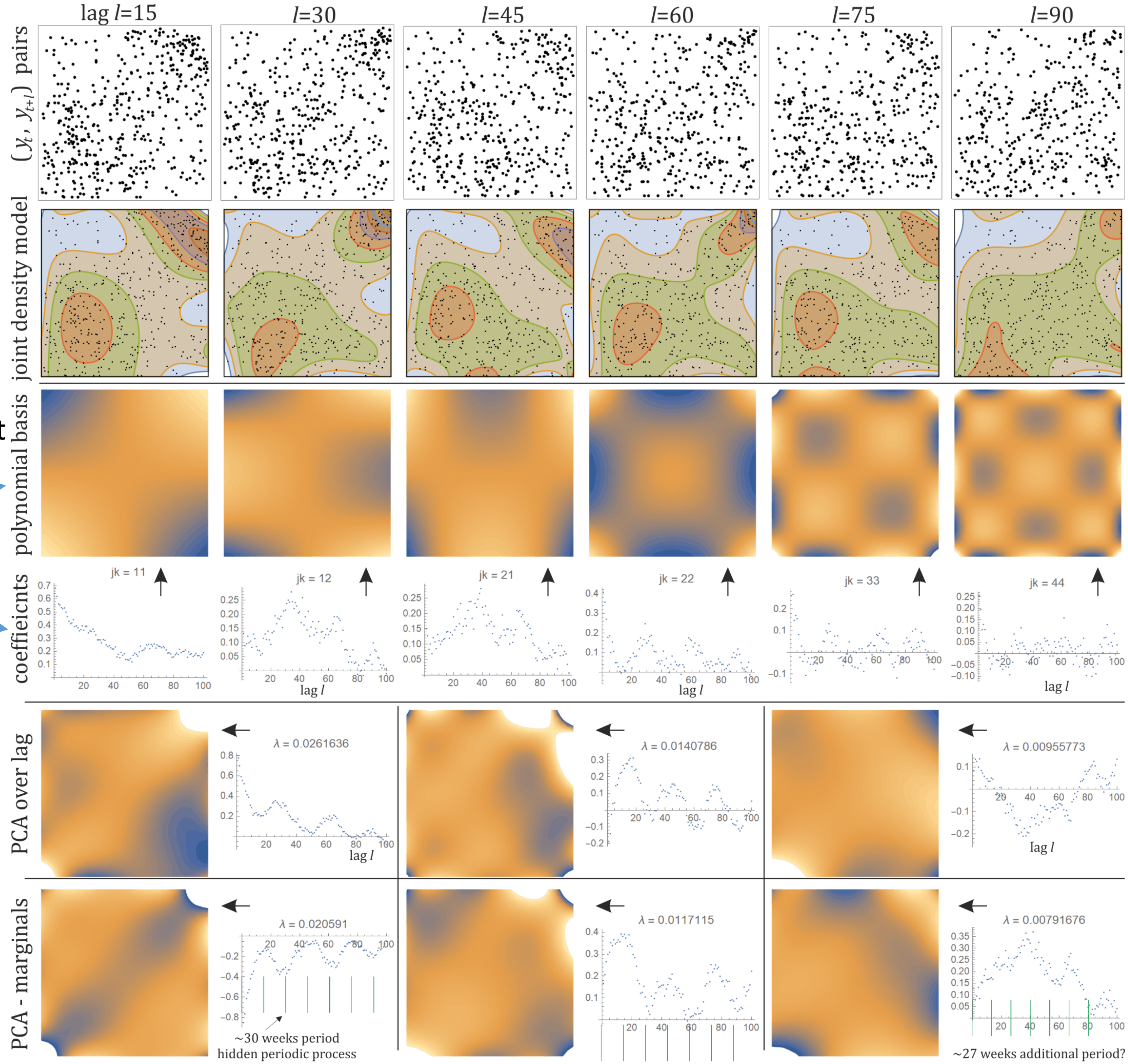
Some $a_{jk}(l)$ sequences

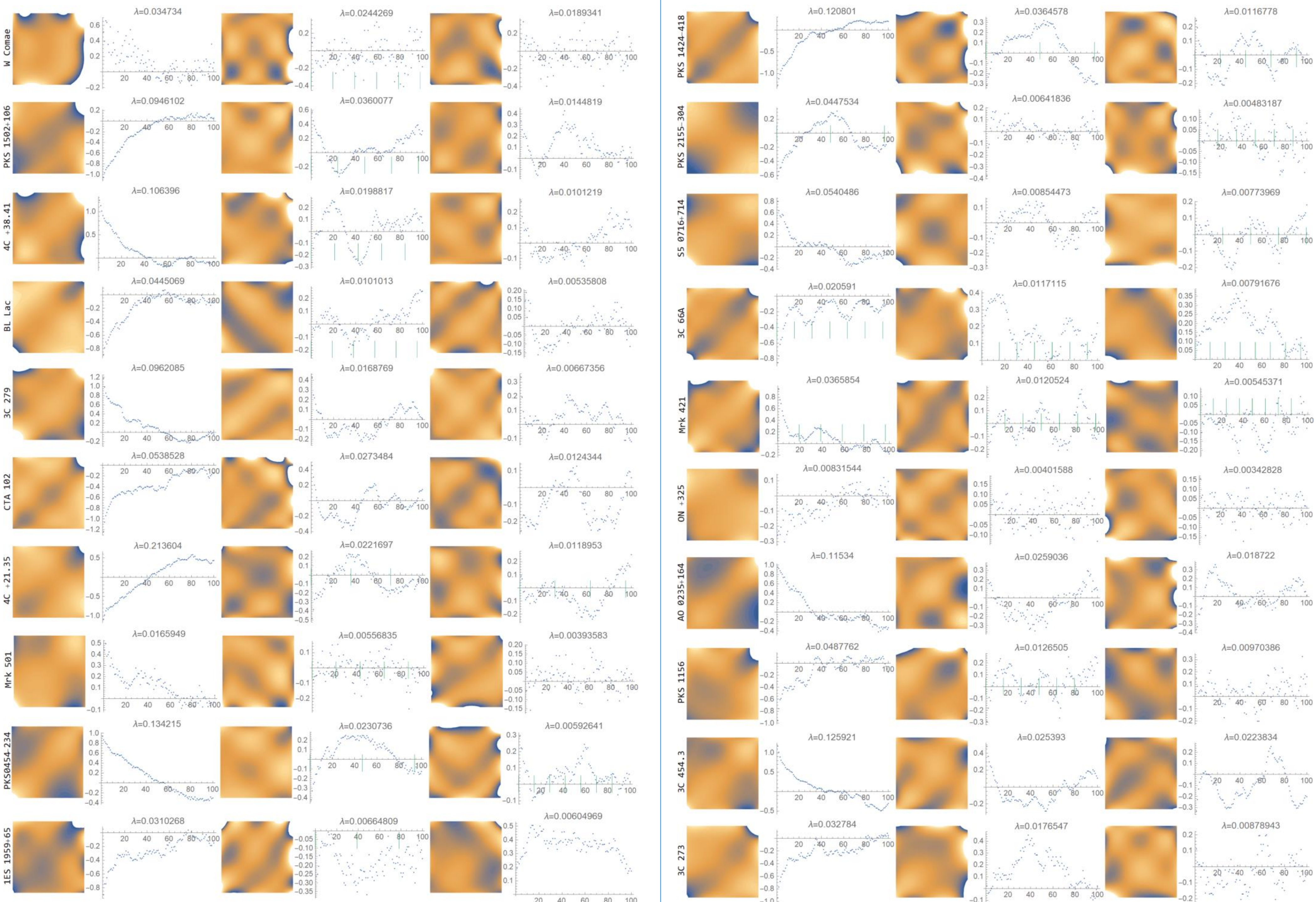
PCA feature select. to reduce basis $25 \to 3$, interpret.

Minus marginals:

$\tilde{a}_{jk} = a_{jk} - a_{j0}a_{0k}$

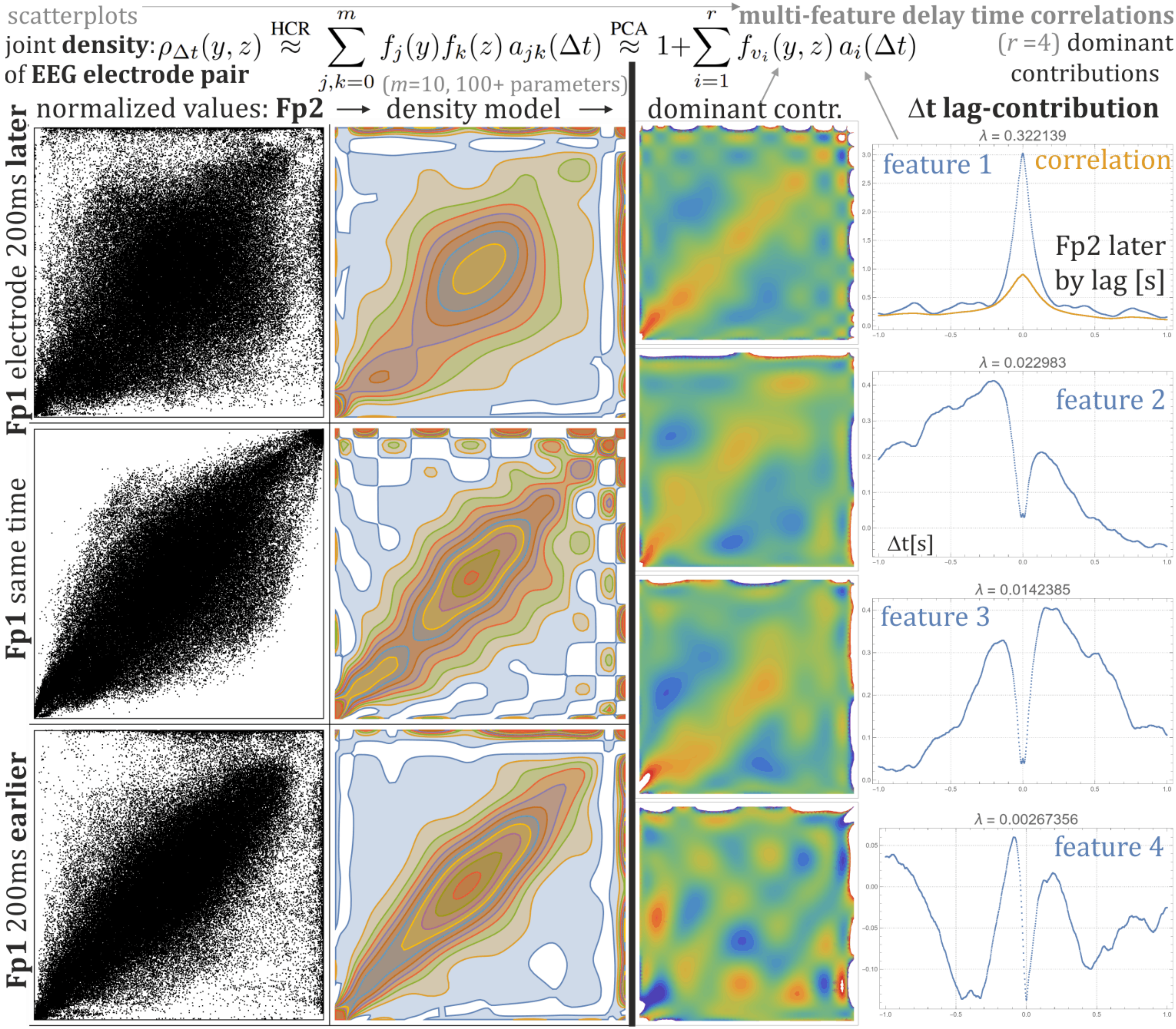
for $\quad j,k \ge 1$




W Comae
λ=0.034734
λ=0.0244269
λ=0.0189341
PKS 1502+106
λ=0.0946102
λ=0.0360077
λ=0.0144819
4C +38.41
λ=0.106396
λ=0.0198817
λ=0.0101219
BL Lac
λ=0.0445069
λ=0.0101013
λ=0.00535808
3C 279
λ=0.0962085
λ=0.0168769
λ=0.00667356
CTA 102
λ=0.0538528
λ=0.0273484
λ=0.0124344
4C +21.35
λ=0.213604
λ=0.0221697
λ=0.0118953
Mrk 501
λ=0.0165949
λ=0.00556835
λ=0.00393583
PKS0454-234
λ=0.134215
λ=0.0230736
λ=0.00592641
1ES 1959+65
λ=0.0310268
λ=0.00664809
λ=0.00604969
PKS 1424-418
λ=0.120801
λ=0.0364578
λ=0.0116778
PKS 2155-304
λ=0.0447534
λ=0.00641836
λ=0.00483187
S5 0716+714
λ=0.0540486
λ=0.00854473
λ=0.00773969
3C 66A
λ=0.020591
λ=0.0117115
λ=0.00791676
Mrk 421
λ=0.0365854
λ=0.0120524
λ=0.00545371
ON +325
λ=0.00831544
λ=0.00401588
λ=0.00342828
AO 0235+164
λ=0.11534
λ=0.0259036
λ=0.018722
PKS 1156
λ=0.0487762
λ=0.0126505
λ=0.00970386
3C 454.3
λ=0.125921
λ=0.025393
λ=0.0223834
3C 273
λ=0.032784
λ=0.0176547
λ=0.00878943

Multi-feature cross-correlation analysis

EEG

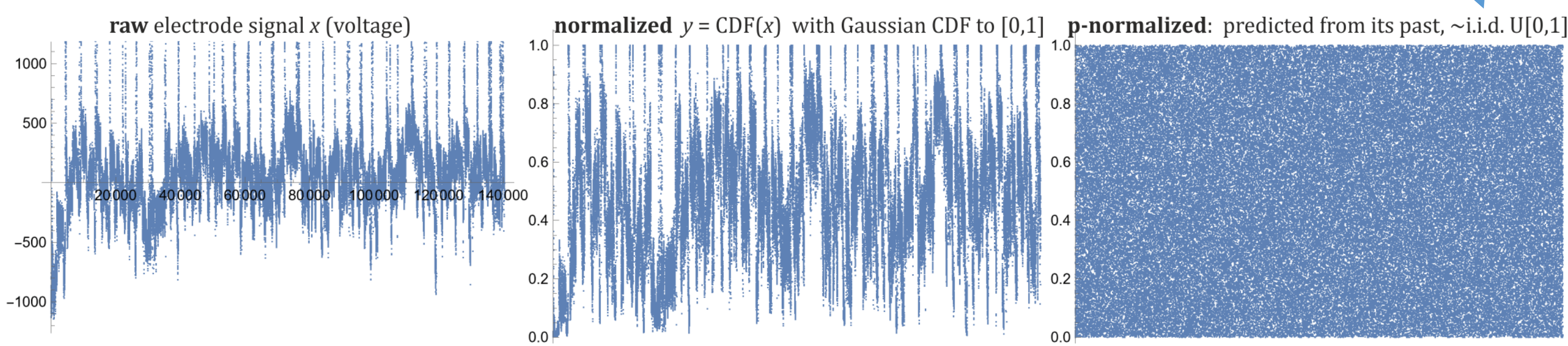


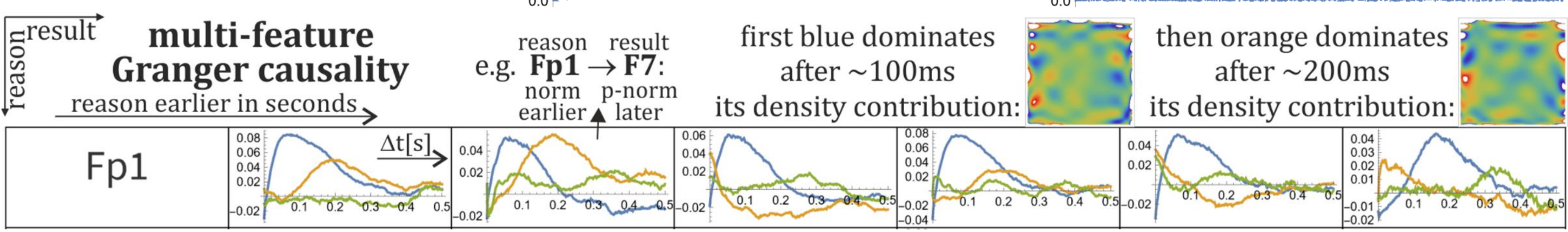
"If a signal $\boldsymbol{Y} = (y_t)$ "Granger-causes" (or "G-causes") a signal $\boldsymbol{X}$, then **past values of Y should contain information that helps predict X** above and beyond the information contained in **past values of X alone**"

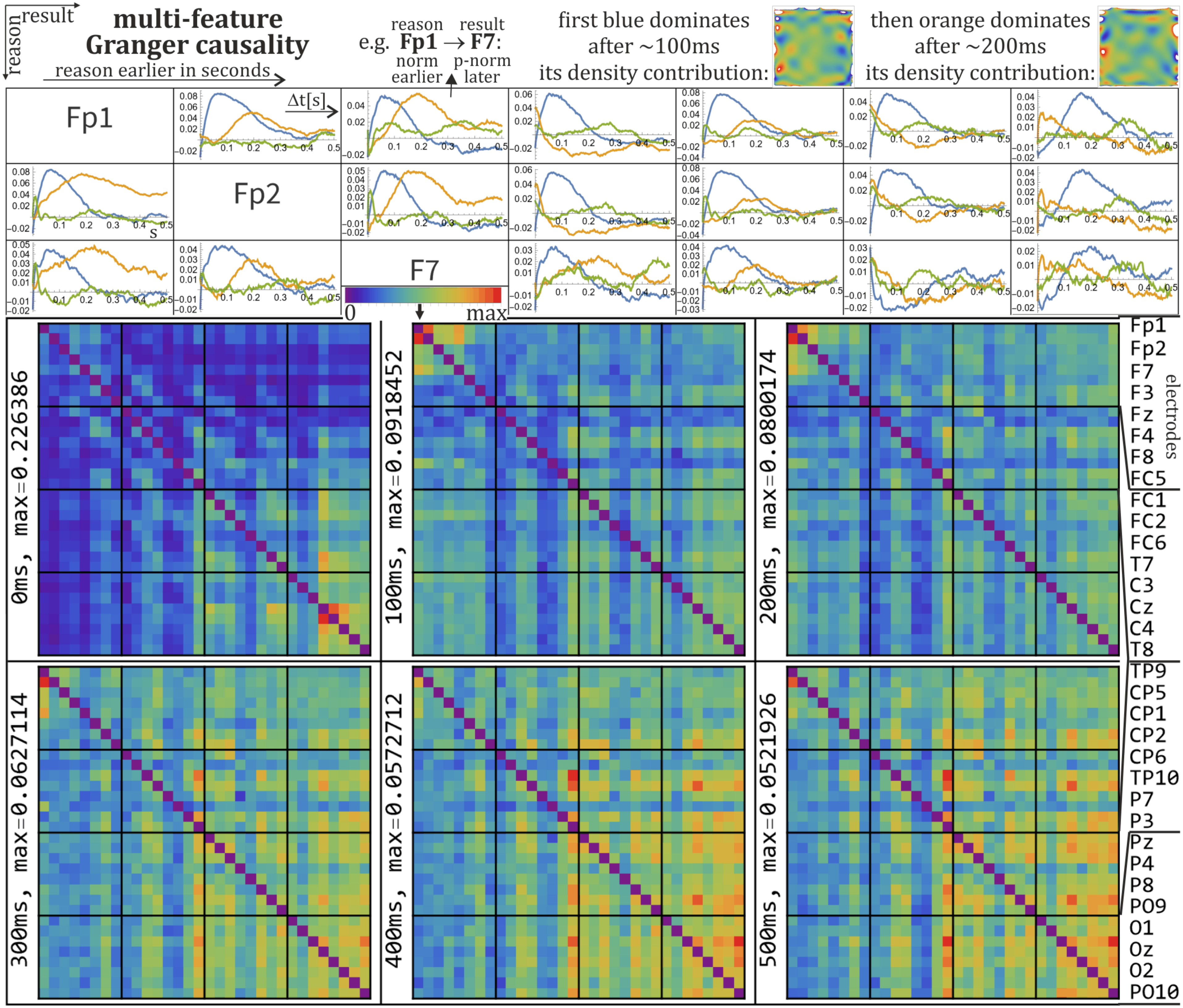
Usually **true/false**: **linear regression of X with/without $(\boldsymbol{y_\tau}: \boldsymbol{\tau} < \boldsymbol{t})$,**

Proposed **multi-feature Granger causality**: multiple, delay dependence:

1) **Residue** $r_t = x_t -$ "prediction of x_t from its past": $(x_\tau: \tau < t)$,
   delay Δt dependence: find correlations in $(\boldsymbol{r_t}, \boldsymbol{y_{t-\Delta t}})_{\textbf{all } t}$
2) **Multiple**: ~i.i.d. residue r_t, **probability prediction**: $r_t = \mathrm{CDF}_{\tau<t}(x_t)$
   multi-feature HCR+PCA: $\rho((r_t, y_{t-\Delta t})_t) \approx 1 + \sum_{i=1}^{r} f_{v_i}(r, y)\, \boldsymbol{a_i}(\boldsymbol{\Delta t})$







result
reason
multi-feature
Granger causality
reason earlier in seconds
reason result
e.g. Fp1 → F7:
norm earlier
p-norm later
first blue dominates
after ~100ms
its density contribution:
then orange dominates
after ~200ms
its density contribution:
Fp1
Fp2
F7
Δt[s]
s
0
max
0ms, max=0.226386
100ms, max=0.0918452
200ms, max=0.0800174
300ms, max=0.0627114
400ms, max=0.0572712
500ms, max=0.0521926
electrodes
Fp1
Fp2
F7
F3
Fz
F4
F8
FC5
FC1
FC2
FC6
T7
C3
Cz
C4
T8
TP9
CP5
CP1
CP2
CP6
TP10
P7
P3
Pz
P4
P8
PO9
O1
Oz
O2
PO10

(a): $P(a,b,c) = P(a)P(b|a)P(c|b)$

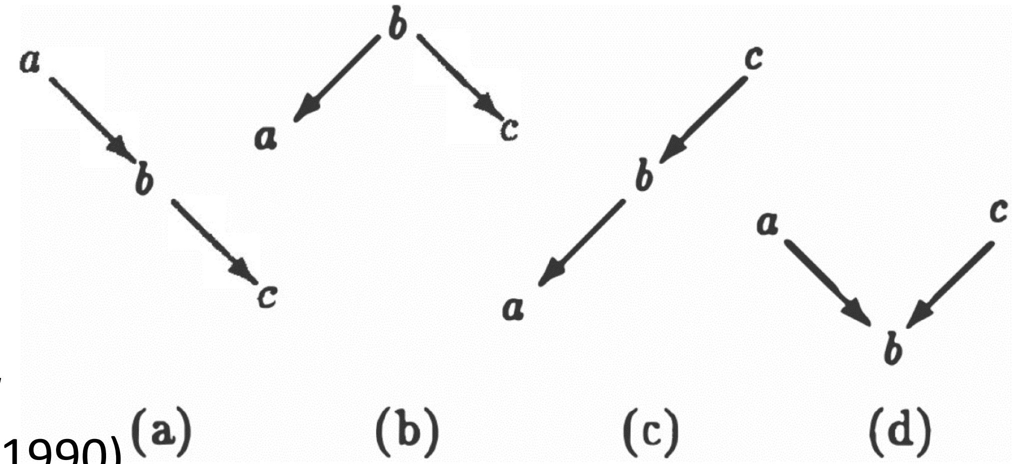
(b): $P(a,b,c) = P(b)P(a|b)P(c|b)$

(c): $P(a,b,c) = P(c)P(b|c)P(a|b)$

(d): $P(a,b,c) = P(a)P(c)P(b|a,c)$

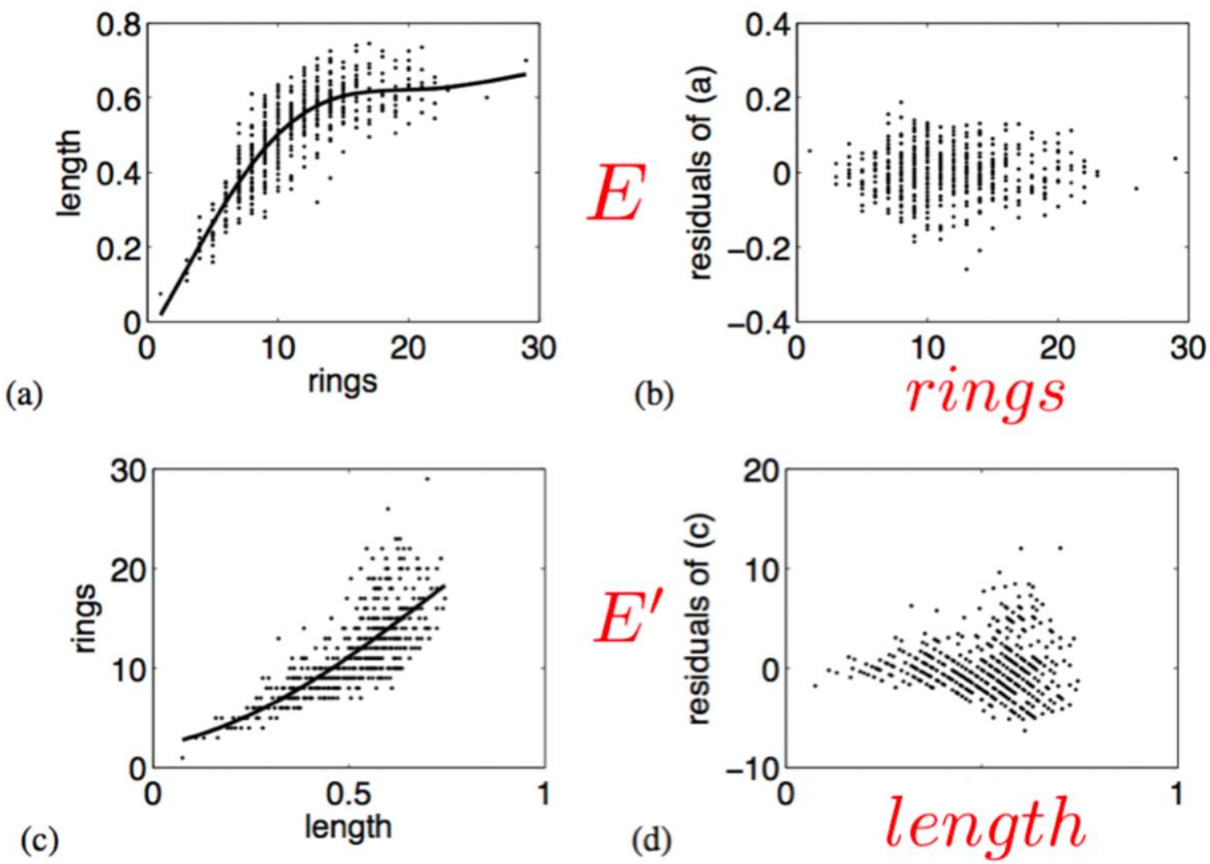
$I(a,b,c)$: $(a) \equiv (b) \equiv (c)$ – b divides a and c

$I(a,\emptyset,c)$: (d) independent   Pearl, Verma (1990)



$\boldsymbol{I(A,B,C)} \equiv \big(P(A|B) = P(A|BC), P(C|B) = P(C|BA)\big)$: ***A* independent of *C* given *B***

Link (article): $\boldsymbol{X \to Y}$: $\boldsymbol{Y = f(X) + E}$ $(\boldsymbol{E \perp X})$ vs $\boldsymbol{Y \to X}$: $\boldsymbol{X = f'(Y) + E'}$ $(\boldsymbol{E' \perp Y})$



length = f(rings) + E

$rings \perp\!\!\!\perp E$

rings = f'(length) + E'

$length \not\perp\!\!\!\perp E'$

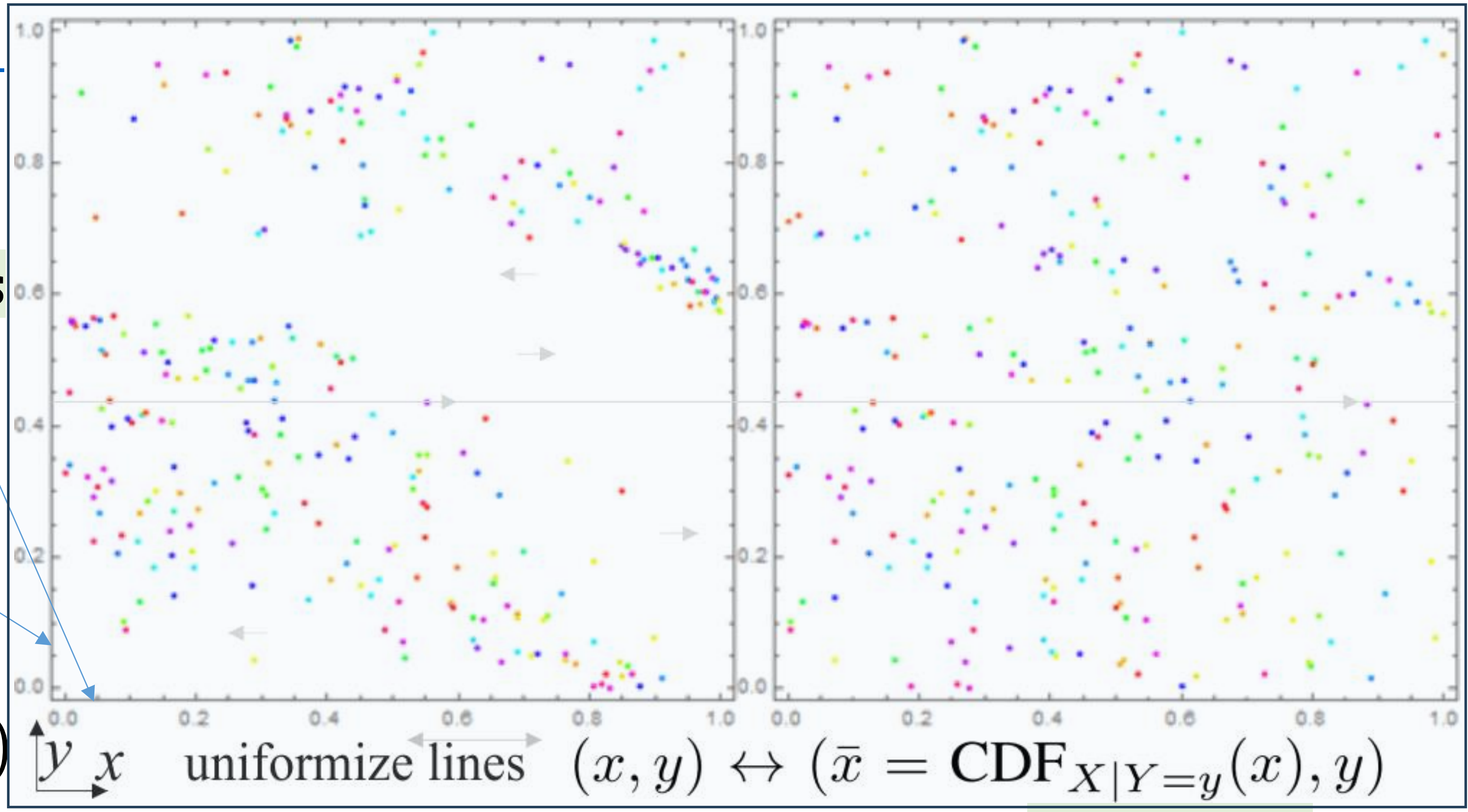
arXiv:2311.13431

$\overline{X|Y} = \mathrm{CDF}_{X|Y}(X)$

normalized variable extracts individual information of X, removing information of Y

$(X,Y) \leftrightarrow \left(\overline{X|Y}, Y\right)$

reversible$\left(\mathrm{CDF}_{X|Y}^{-1}\right)$



direct mutual information (removing intermediate Z): $I\left(\overline{X|Z}; \overline{Y|Z}\right)$

decouple to independent: $(X_1, \ldots, X_n) \leftrightarrow \left(\tilde{X}_1, \ldots, \tilde{X}_n\right)$: $\forall_{i\neq j}\ \tilde{X}_i \perp \tilde{X}_j\ , X_i \perp \tilde{X}_j$

Granger/interpretable models from individual decoupled variables

Bias removal (e.g. gender, age) from data: not to be used by "ethical ML"

How to model conditional CDF from many variables??? HCR ... ?

+ indirect

X — I(X;Y) bits — Z — Y

direct: $I\left(\overline{X|Z}; \overline{Y|Z}\right)$

$(X_1, \ldots, X_n)$ **decouple** to individual information

$\updownarrow$

$(\tilde{X}_1, \ldots, \tilde{X}_n)$ $\forall_{i\neq j} \tilde{X}_i \perp\!\!\!\perp \tilde{X}_j, \tilde{X}_i \perp\!\!\!\perp X_j$

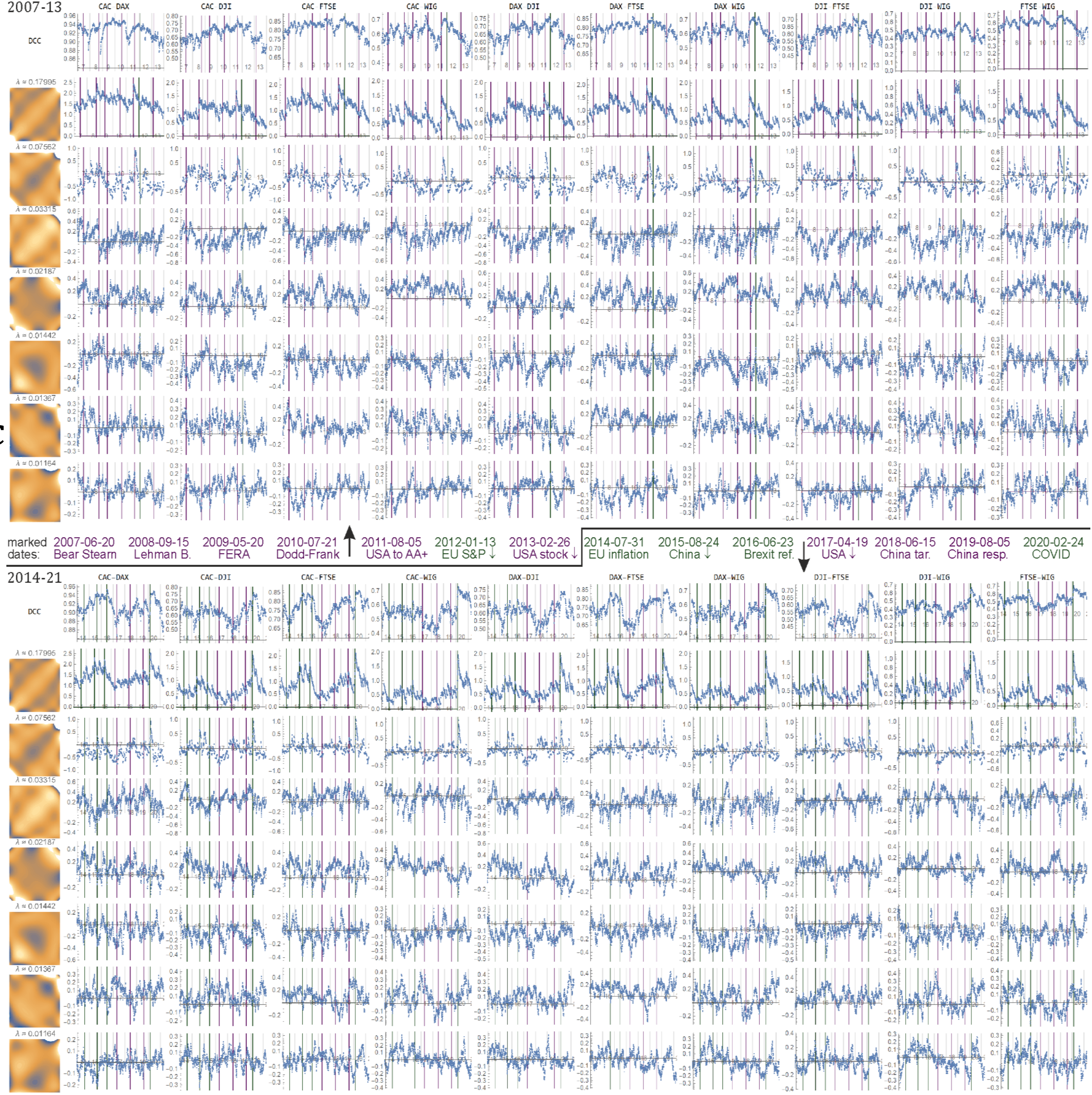
Multi-feature correlation analysis (CEJOR)

evolving in time like

[DCC](DCC) – dynamic conditional correlations

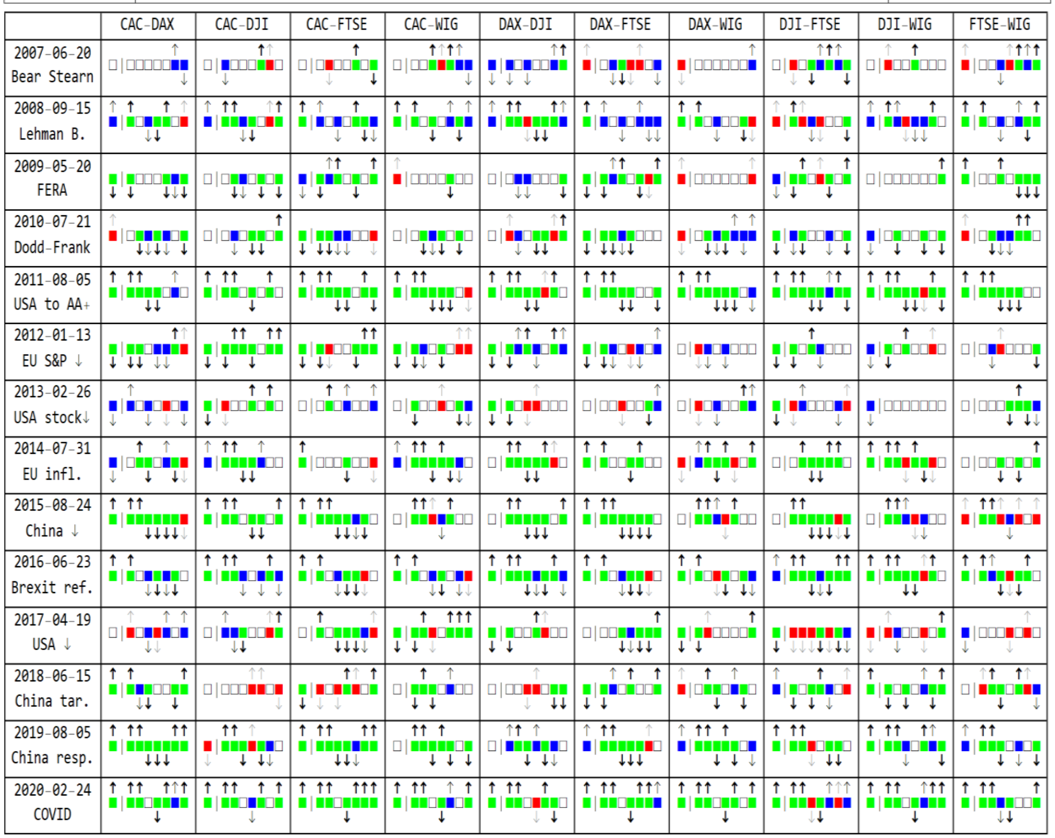
E.g. for **Contagion analysis between markets**

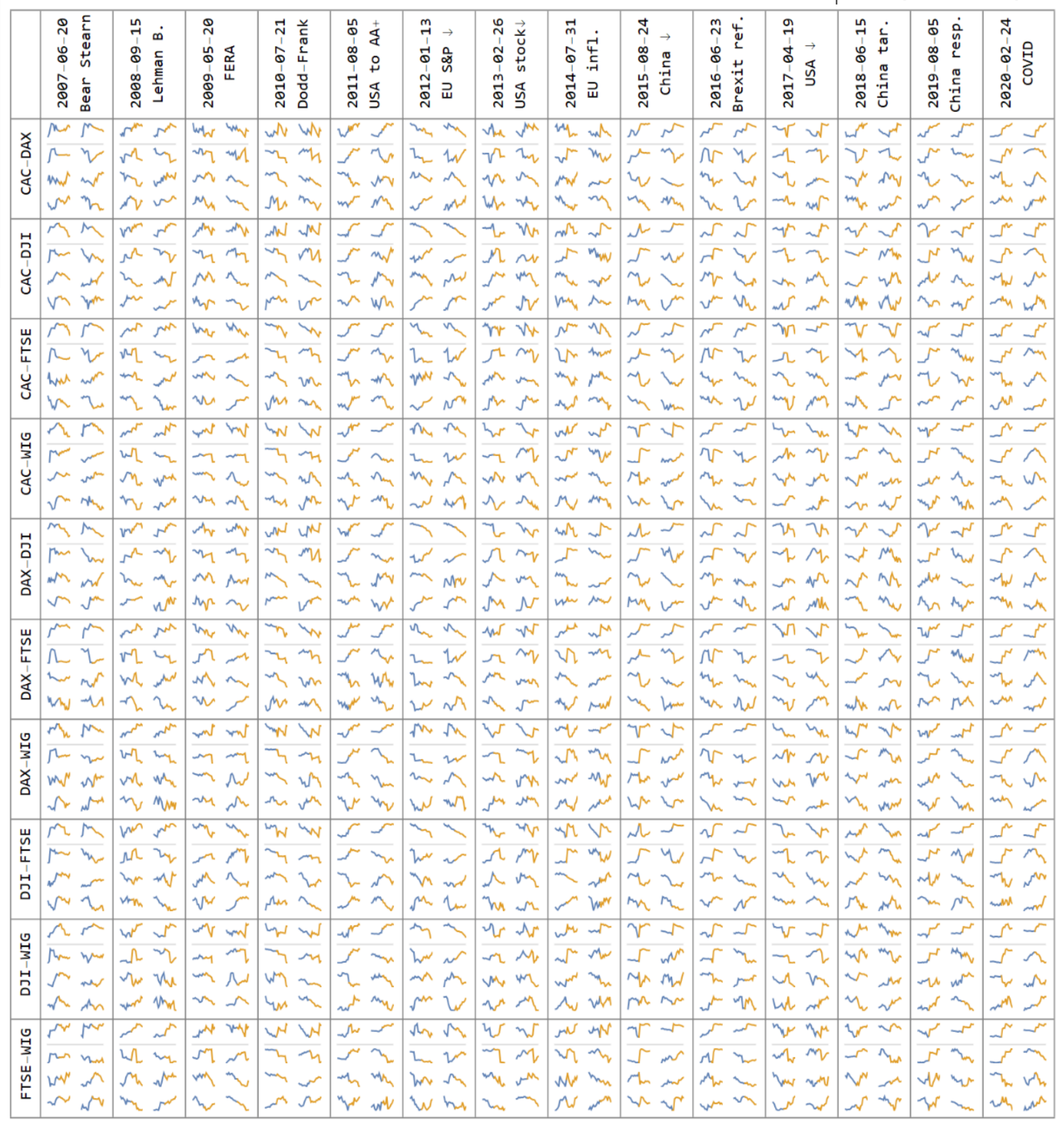
e.g. to detects crucial events



1/mean P-values of event detection:

Date	Event	Abbreviation
2007-06-20	Bear Stearn bailed out 2 of its hedge funds with $20 billion[2]	Bear Stearn
2008-09-15	Bankruptcy of Lehman Brothers[3]	Lehman B.
2009-05-20	President Obama signed the Fraud Enforcement and Recovery Act[4]	FERA
2010-07-21	Dodd–Frank Wall Street Reform and Consumer Protection Act enacted[5]	Dodd–Frank
2011-08-05	S&P downgrade of USA from AAA to AA+[6]	USA to AA+
2012-01-13	Standard & Poor's downgrades France and eight other eurozone countries[7]	EU S&P ↓
2013-02-26	American stock exchanges evaluated results of Italian elections very negatively[8]	USA stock ↓
2014-07-31	Announcement of bad data about inflation in Euro zone[9]	EU inflation
2015-08-24	Announcements of bad economic data from China[10]	China ↓
2016-06-23	Brexit referendum[11]	Brexit ref.
2017-04-19	Announcements of bad economic results of US companies[12]	USA ↓
2018-06-15	Begin U.S. – China Trade War[13]	China tar.
2019-08-05	Halting by China purchases of U.S. agricultural products[14]	China resp.
2020-02-24	The coronavirus outbreak spread worsened substantially outside China[15]	COVID

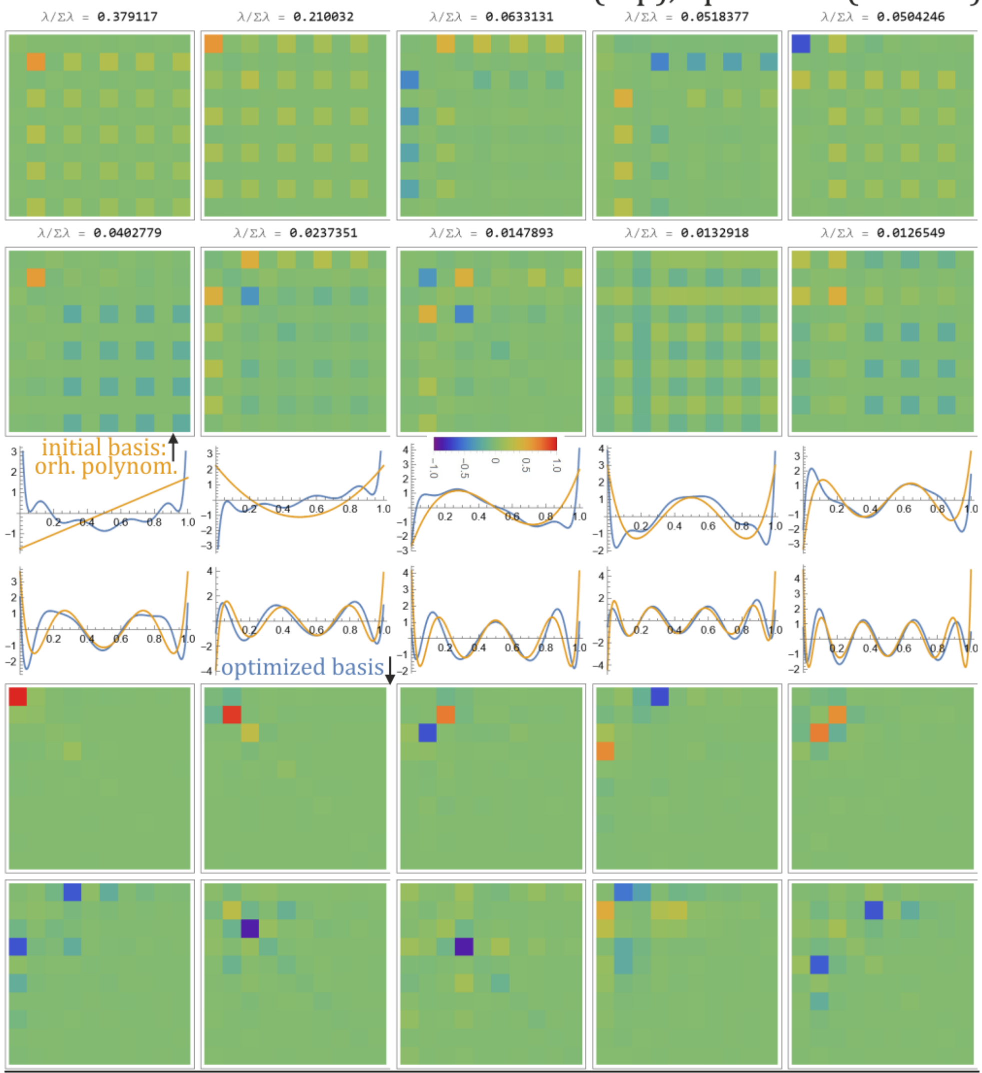




DCT-II (e.g. **JPEG**): $X_k = \sum_{n=0}^{N-1} x_n \cos\left(\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right)$

cheap (~FFT)

10 PCA autocorrelation features: initial(top), optimized (bottom)



Why?

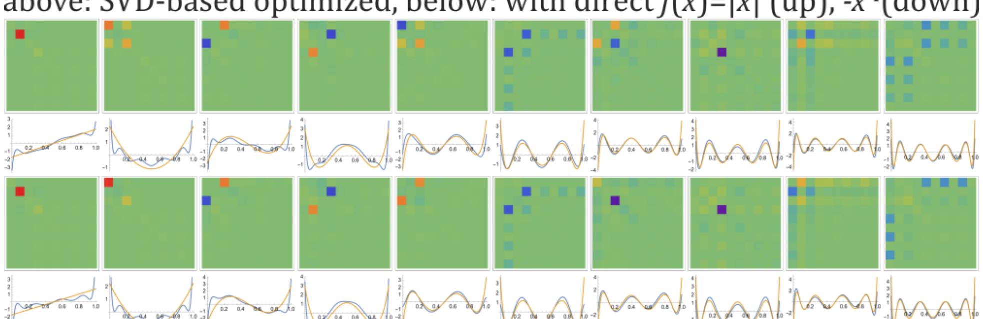
PCA in 8×8 + **optimization of common basis** indeed leads to ~DCT-II

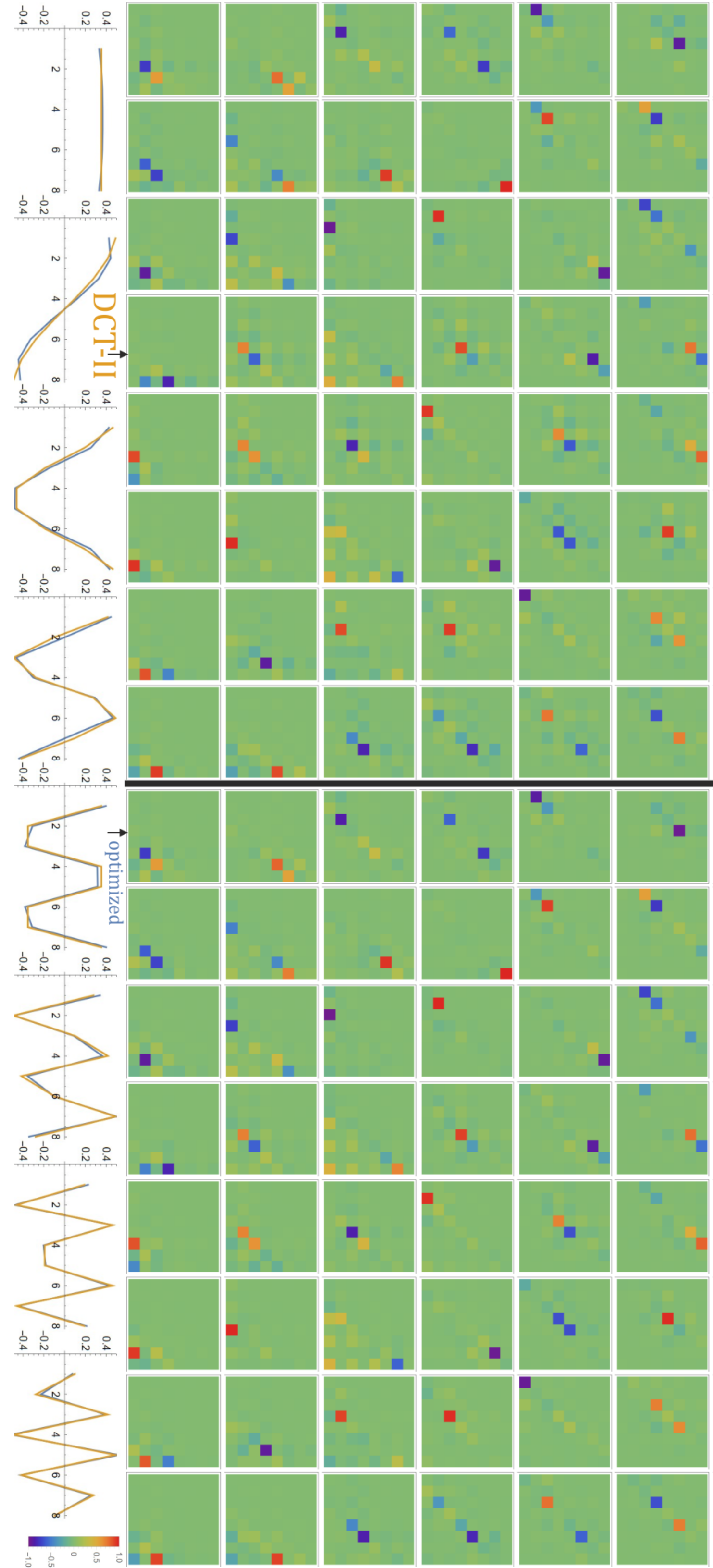
of autocorrelation basis

SVD of

$$\sum_k w_k A_k A_k^T$$

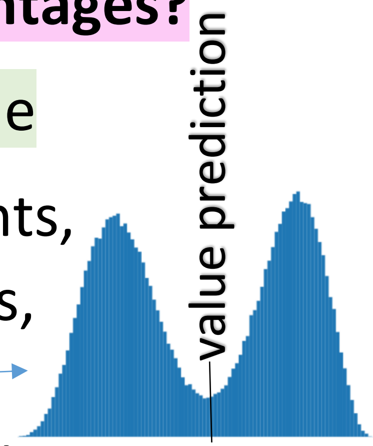
above: SVD-based optimized, below: with direct $f(x)=|x|$ (up), $-x^4$(down)



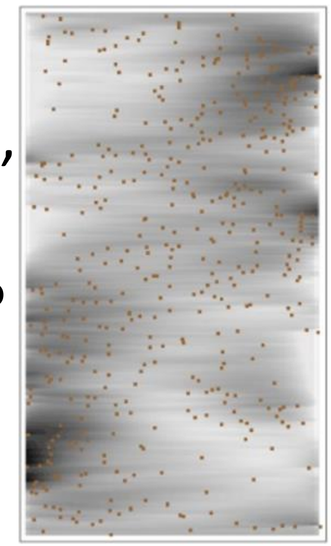


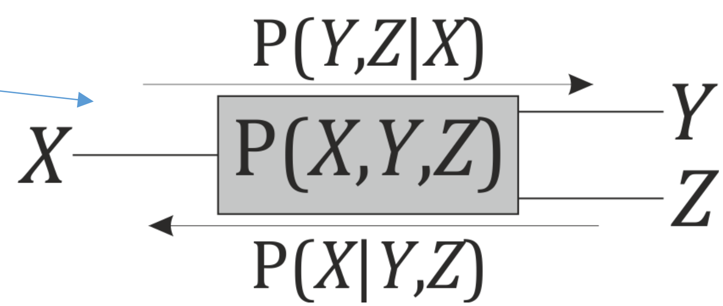
People usually focus on **prediction of values** (→ of moments → density)
where **prediction of probability distributions** gives some **advantages?**

Prediction of value is of e.g. Gaussian around this value

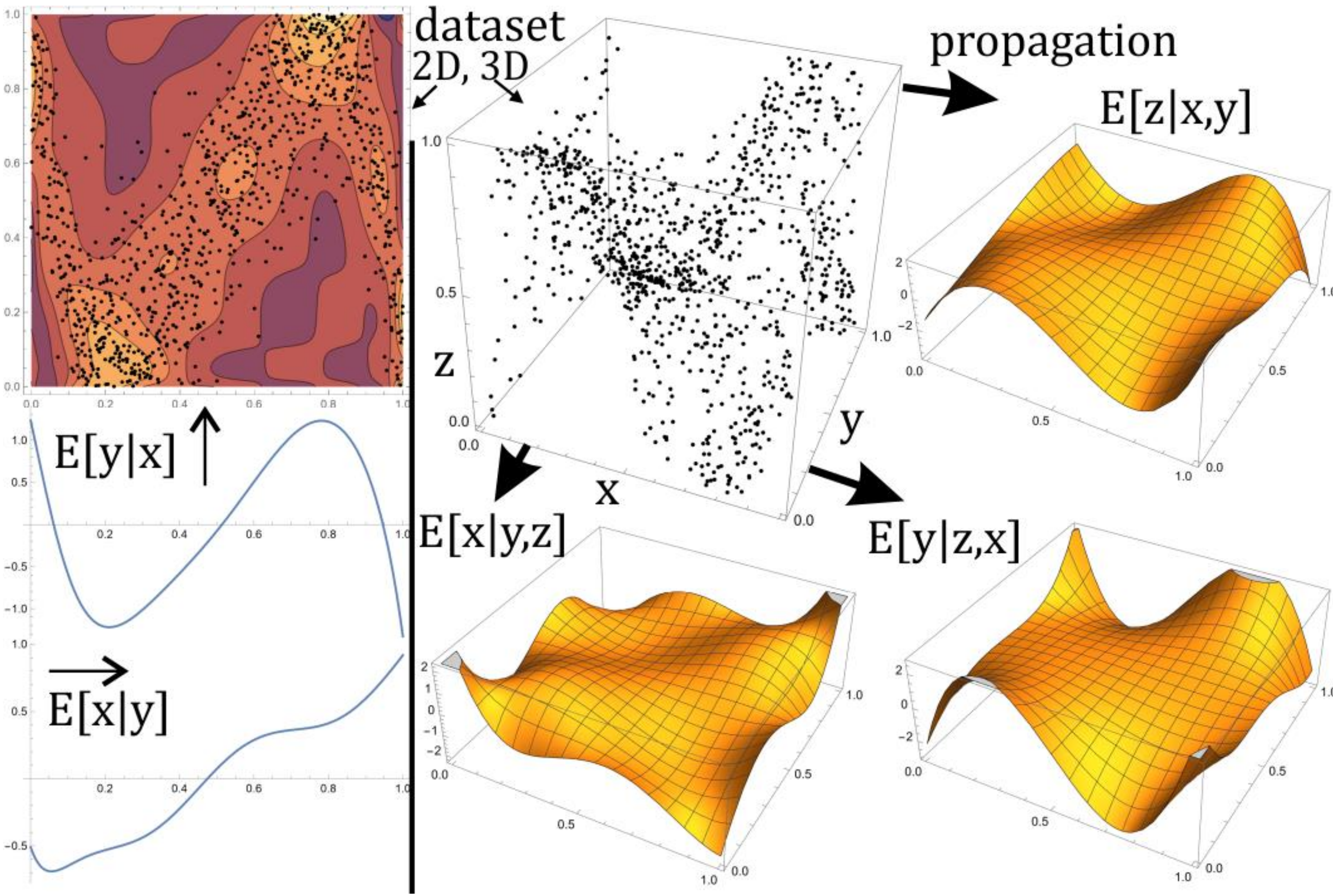


- control of prediction **uncertainty**, prediction of higher moments,
- statistical modelling in **data compression** – needs probabilities,
- controlling **multi-modality**, e.g. Gaussian-mixture,
- proper **variable contribution evaluations** e.g. with conditional entropy,
- **Monte-Carlo**: generate random scenarios with close statistics,
- **credibility evaluation**, find outliers – low probability datapoints,
- nonstationarity analysis: **evolution of probability density**,
- asymmetric correlations e.g. a_{12} – implying **causality direction**?
- **extreme statistics optimization** e.g. best batch of drugs to test,
- **uniformize** data in multiple dimensions e.g. $x' = \mathrm{CDF}_y(x)$,
- **selection for extreme, certain values** – e.g. **virtual screening** of drugs,
- **Bayes scenarios** directly from joint distribution
- **biology-inspired neural networks** …





Multidirectional propagation of conditional distributions, their expected values



Neuron containing local joint distribution model – of its connections with $(a_{\boldsymbol{j}})$ e.g. a_{ij} **matrix** $(d=2)$/**tensor** $(d>2)$ **as neuron parameters propagation as conditional densities/expected values** $E[x|y], E[y|x]$

$$\rho(x,y) = \sum_{i,j\geq 0} a_{ij}\, f_i(x) f_j(y)$$

$$a_{ij} = \frac{1}{|\bar{X}|} \sum_{(x,y)\in\bar{X}} f_i(x)\, f_j(y)$$

direct estimation

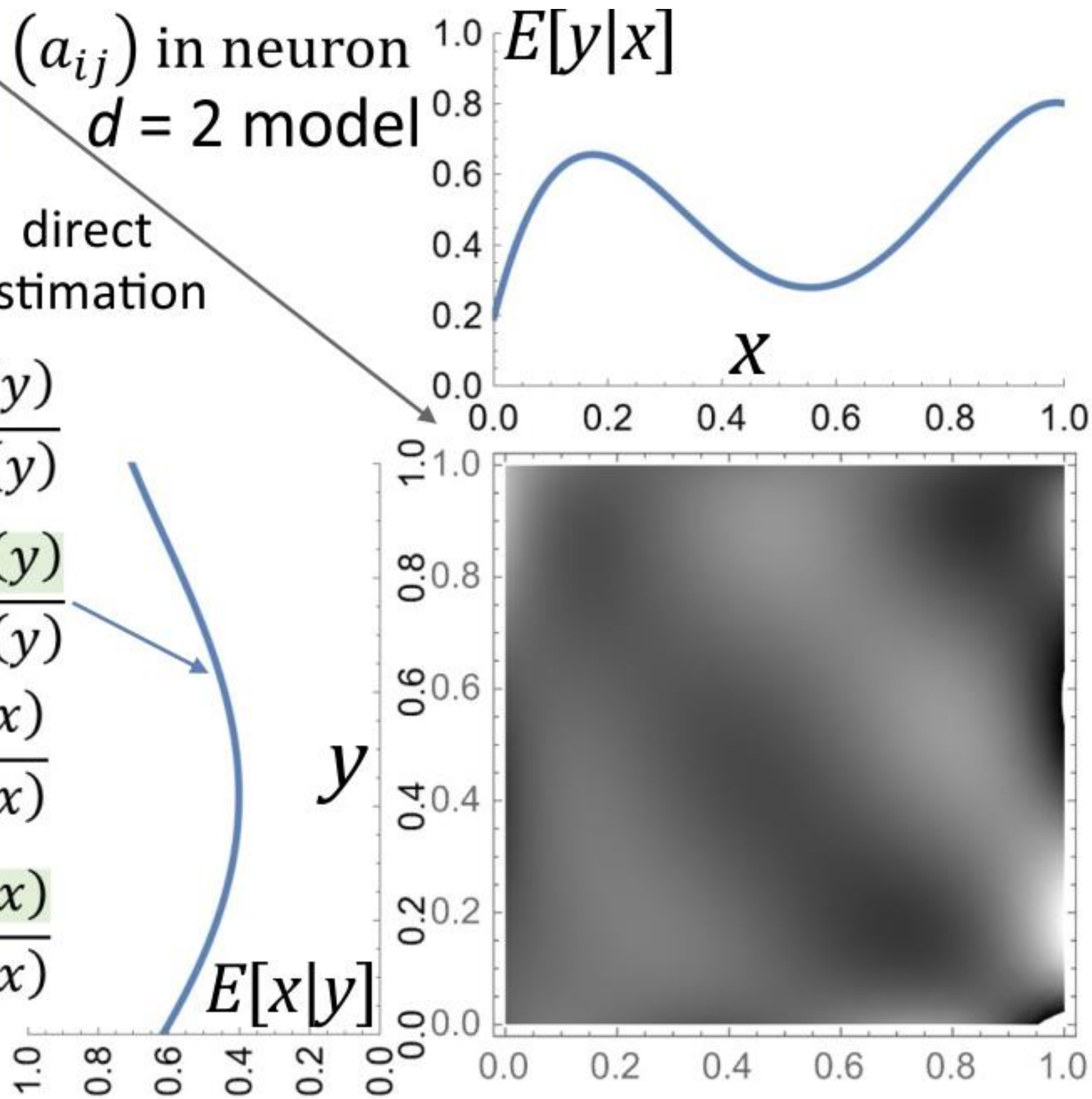
$$\rho(x|y) = \sum_{i\geq 0} f_i(x) \frac{\sum_j a_{ij}\ f_j(y)}{\sum_j a_{0j}\ f_j(y)}$$

$$E[x|y] = \frac{1}{2} + \frac{1}{2\sqrt{3}} \frac{\sum_j a_{1j}\ f_j(y)}{\sum_j a_{0j}\ f_j(y)}$$

$$\rho(y|x) = \sum_{j\geq 0} f_j(x) \frac{\sum_i a_{ij}\ f_i(x)}{\sum_i a_{i0}\ f_i(x)}$$

$$E[y|x] = \frac{1}{2} + \frac{1}{2\sqrt{3}} \frac{\sum_i a_{i1}\ f_i(x)}{\sum_i a_{i0}\ f_i(x)}$$

$$I(X;Y) \approx \sum_{i,j>0} (a_{ij})^2$$



Can we directly train intermediate layers? (video)

Naftali Tishby, **information theoretic view** – permutation, bijection independent

Markov process between layers, **first extract/compress essential information**

reducing **mutual information** [bits] $H(X) \geq I(X;T_1) \geq I(X;T_2) \geq \cdots$

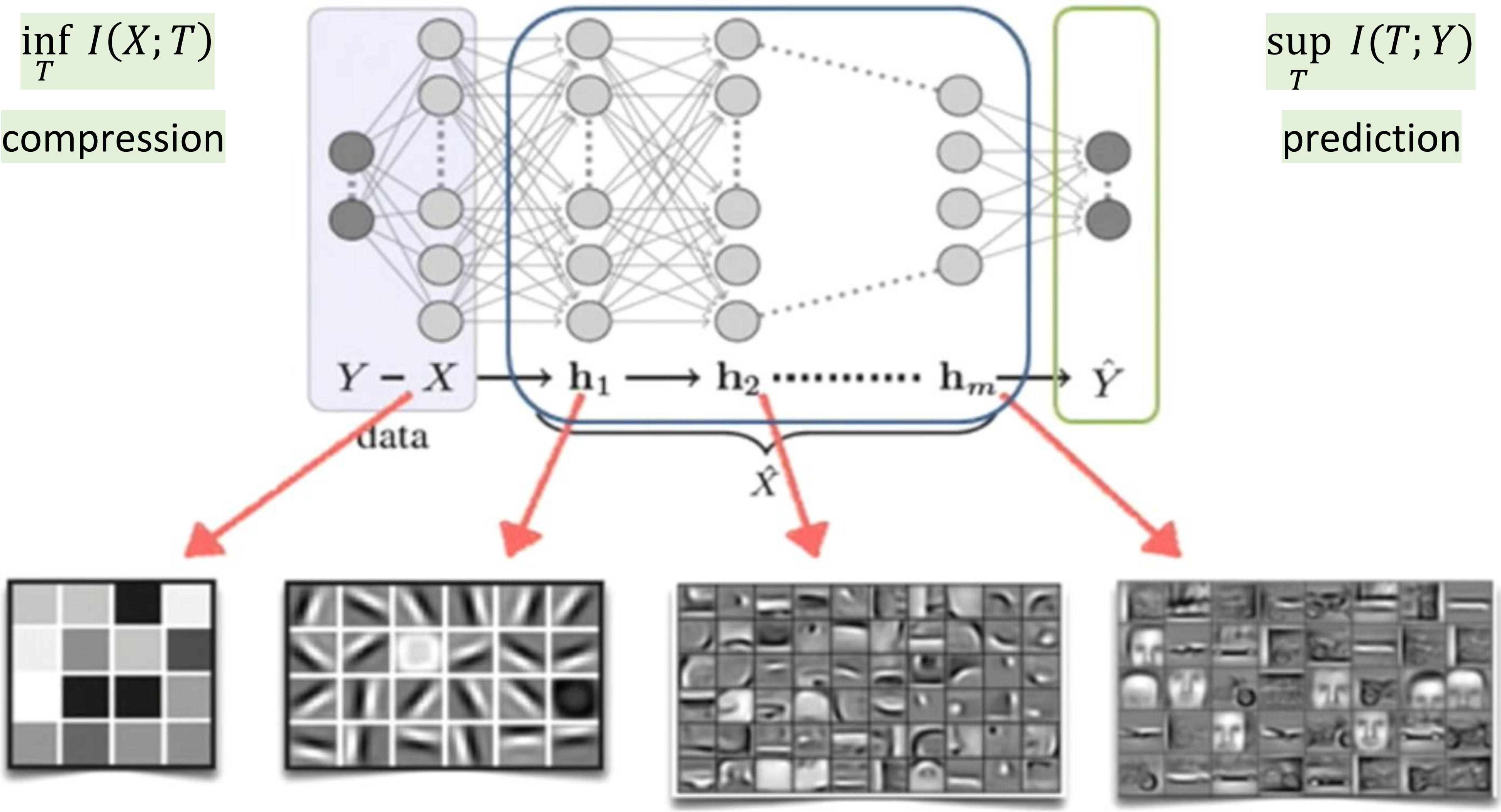
Information bottleneck (Tishby): for $X \to T \to Y$ optimize $\inf_T I(X;T) - \beta I(T;Y)$

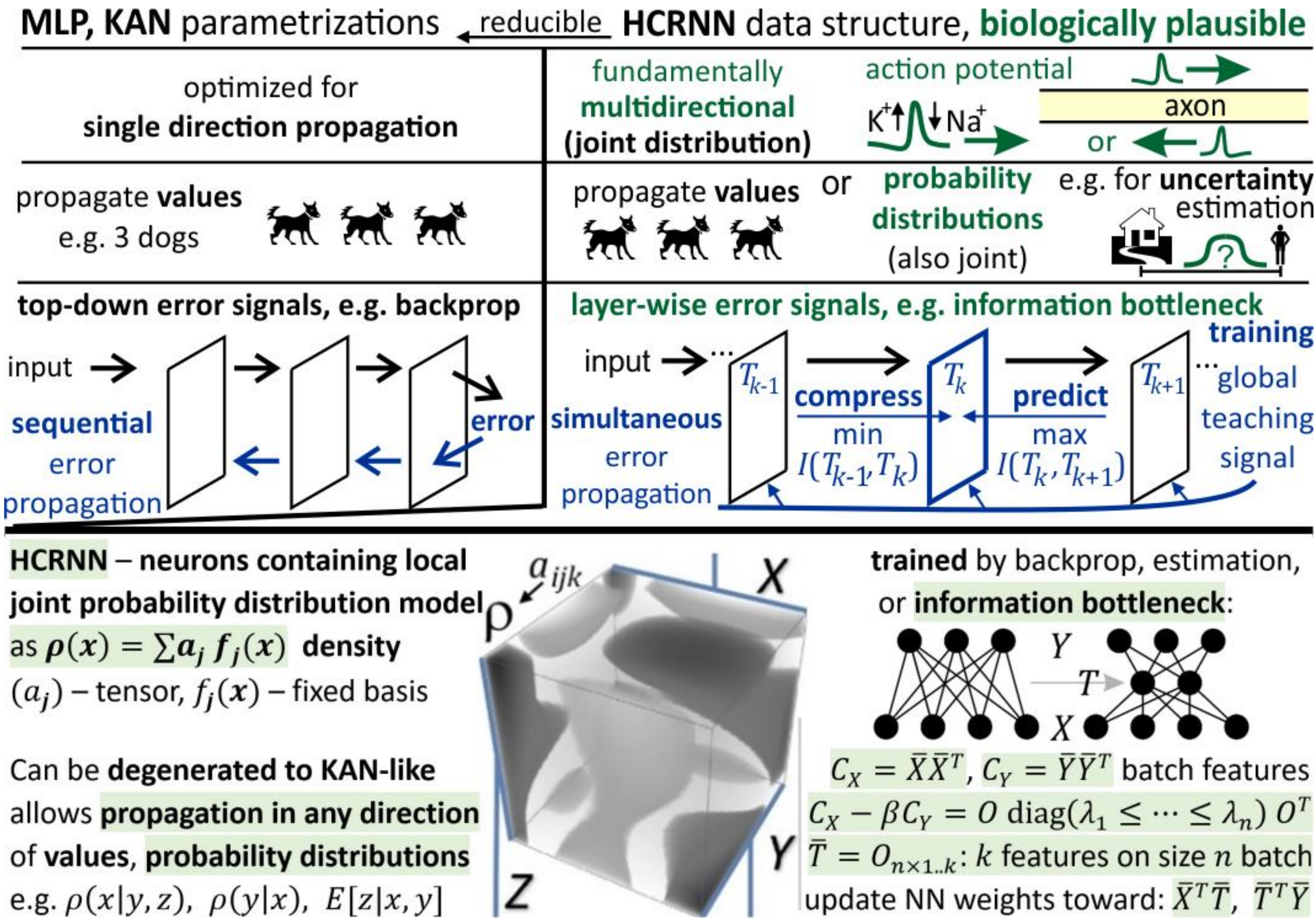
$\inf_T I(X;T)$

compression

$\sup_T I(T;Y)$

prediction



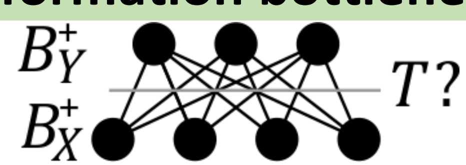

MLP, KAN parametrizations
reducible
HCRNN data structure, biologically plausible
optimized for
single direction propagation
fundamentally
multidirectional
(joint distribution)
action potential
axon
or
propagate values
e.g. 3 dogs
propagate values
or
probability
distributions
(also joint)
e.g. for uncertainty
estimation
?
top-down error signals, e.g. backprop
input
error
sequential
error
propagation
layer-wise error signals, e.g. information bottleneck
training
input
T_{k-1}
T_k
T_{k+1}
compress
predict
global
teaching
signal
simultaneous
error
propagation
min
$I(T_{k-1}, T_k)$
max
$I(T_k, T_{k+1})$
HCRNN – neurons containing local
joint probability distribution model
as $\rho(x) = \sum a_j f_j(x)$ density
(a_j) – tensor, $f_j(x)$ – fixed basis
a_{ijk}
ρ
X
Y
Z
Can be degenerated to KAN-like
allows propagation in any direction
of values, probability distributions
e.g. $\rho(x|y,z)$, $\rho(y|x)$, $E[z|x,y]$
trained by backprop, estimation,
or information bottleneck:
Y
T
X
$C_X = \bar{X}\bar{X}^T$, $C_Y = \bar{Y}\bar{Y}^T$ batch features
$C_X - \beta C_Y = O \operatorname{diag}(\lambda_1 \le \cdots \le \lambda_n)\, O^T$
$\bar{T} = O_{n\times 1..k}$: k features on size n batch
update NN weights toward: $\bar{X}^T\bar{T}$, $\bar{T}^T\bar{Y}$

NN arXiv:2405.05097

Reduced to ~KAN for pairwise-only dependencies

Additionally:

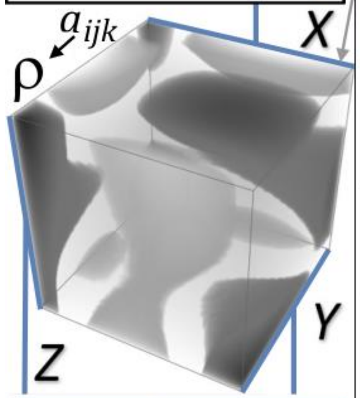
- can extend to **triplewise** or higher,
- **omnidirectional propagation** ↕↔,
- **direct** a_j parameter **estimation**/update,
- propagate values or **probability distributions**,
- **interpretation**: moments,
- cheaply calculate entropy, mutual information,
- additional training e.g. tensor decomposition, **information bottleneck**



d=3 variables

HCR neuron

normalize CDF
CDF^{-1} + calculate $\{f\}$ or $\{g\}$



$\rho(\boldsymbol{x}) = \sum_j a_j f_j(\boldsymbol{x})$

$\rho(x, y, z)$ for $d = 3$

propagation:

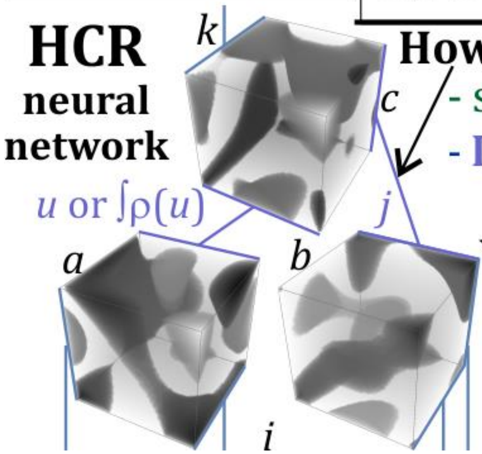
↑ $\rho(x|y,z)$

$\rho(y,z|x)$ ↓

pairwise ~**KAN**: $i \cdot j \cdot k = 0$

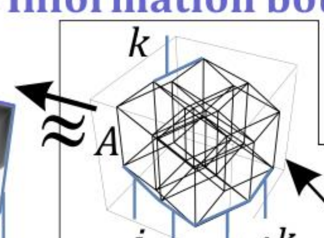
basis in [0,1]	$f_0 = 1, f_1 \propto 2x - 1,\ \int_0^1 f_i(x)\, f_j(x) dx = \delta_{ij}$
↓↑ normalize	$x \leftrightarrow \mathrm{CDF}(x) \sim U[0,1]$ empirical/param.
HCR joint density	$\rho(x,y,z) = \sum_{ijk\in B} a_{ijk}\ f_i(x) f_j(y) f_k(z)$
static **estimation** from $\bar{X}$ dataset	mean: $a_{ijk} = \frac{1}{\lvert\bar{X}\rvert} \sum_{(x,y,z)\in\bar{X}} f_i(x) f_j(y) f_k(z)$
dynamic (EMA) model **update**	$a_{ijk} \xrightarrow{(x,y,z)} (1-\lambda) a_{ijk} + \lambda\, f_i(x) f_j(y) f_k(z)$
$\rho(X = x \mid y,z) \approx$ ↑ conditional	$\sum_i f_i(x) \frac{\sum_{jk} a_{ijk}\, f_j(y) f_k(z)}{\sum_{jk} a_{0jk}\ f_j(y) f_k(z)}$ $\frac{\text{current}}{\text{normal.}}$
$E[X = x \mid y,z] \approx$ ↑ **propagation**?	$\frac{1}{2} + \frac{1}{2\sqrt{3}} \frac{\sum_{jk} a_{1jk}\, f_j(y) f_k(z)}{\sum_{jk} a_{0jk}\ f_j(y) f_k(z)}$ sufficient if norm.
$\rho(y,z \mid x) \approx$ ↓ conditional	$\sum_{jk} f_j(y) f_k(z) \frac{\sum_i a_{ijk}\, f_i(x)}{\sum_i a_{i00}\ f_i(x)}$ $\frac{\text{current}}{\text{normal.}}$
$E[Y = y \mid x] \approx$ ↓ **propagation**?	$\frac{1}{2} + \frac{1}{2\sqrt{3}} \frac{\sum_j a_{1j0}\, f_j(y)}{\sum_j a_{0j0}\ f_j(y)}$ polyn. KAN-like sufficient if normalized
entropy, mutual information	$H(X) \approx -\sum_{\boldsymbol{j}\in B_X^+} (a_{\boldsymbol{j}})^2$ [nits] $I(X;Y) \approx \sum_{\boldsymbol{j}_x\in B_X^+} \sum_{\boldsymbol{j}_y\in B_Y^+} \left(a_{(\boldsymbol{j}_x,\boldsymbol{j}_y)}\right)^2$
basis optimization $(y,z) \to x$ $\{f_i(x)\} \to \{g_i(x)\}$	$M_{i,jk} = a_{ijk}$ SVD: $MM^T = \sum_i \sigma_i\, \boldsymbol{v}_i \boldsymbol{v}_i^T$ $g_i(x) = \sum_j v_{ij}\ f_j(x)$ $\boldsymbol{v}_i \cdot \boldsymbol{v}_j = \delta_{ij}$ $f_i = \sum_l v_{li} g_l$ $a_{ijk} \to \sum_l v_{li}\ a_{ljk}$

HCR neural network



How to train intermediate layers/variables?

- standard **backpropagation** of a_{ijk} gradients
- **Information bottleneck method** for neurons
- **up/down propagation + a_{ijk} estimation/update**
- **tensor decomposition**



$A^k_{i_1 i_2 i_3 i_4} \approx \sum_{j_1,j_2} a^{j_1}_{i_1 i_2}\ b^{j_2}_{i_3 i_4}\ c^k_{j_1 j_2}$