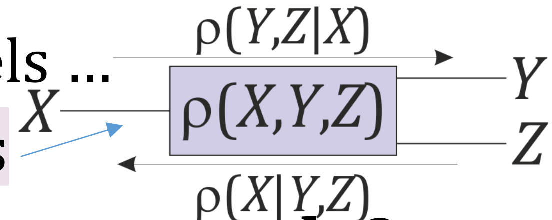
HIERARCHICAL CORRELATION RECONSTRUCTION

for time series, conditional distribution (~~Bayes~~) models …

(nonlinear, adaptive, all-directional) artificial neurons



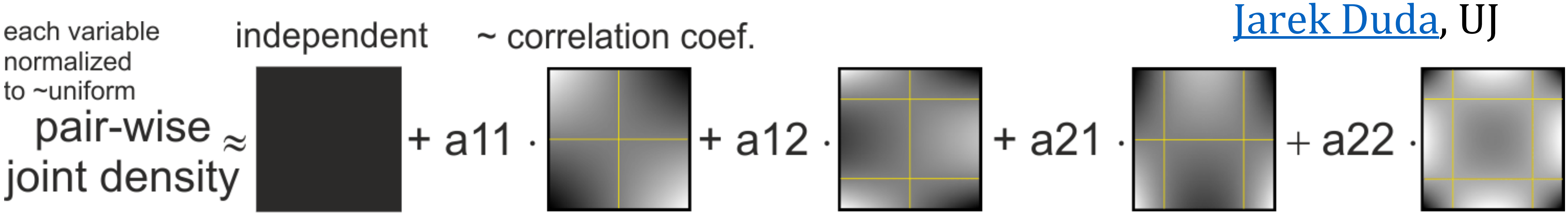
How to model/estimate density from a data sample?

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

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

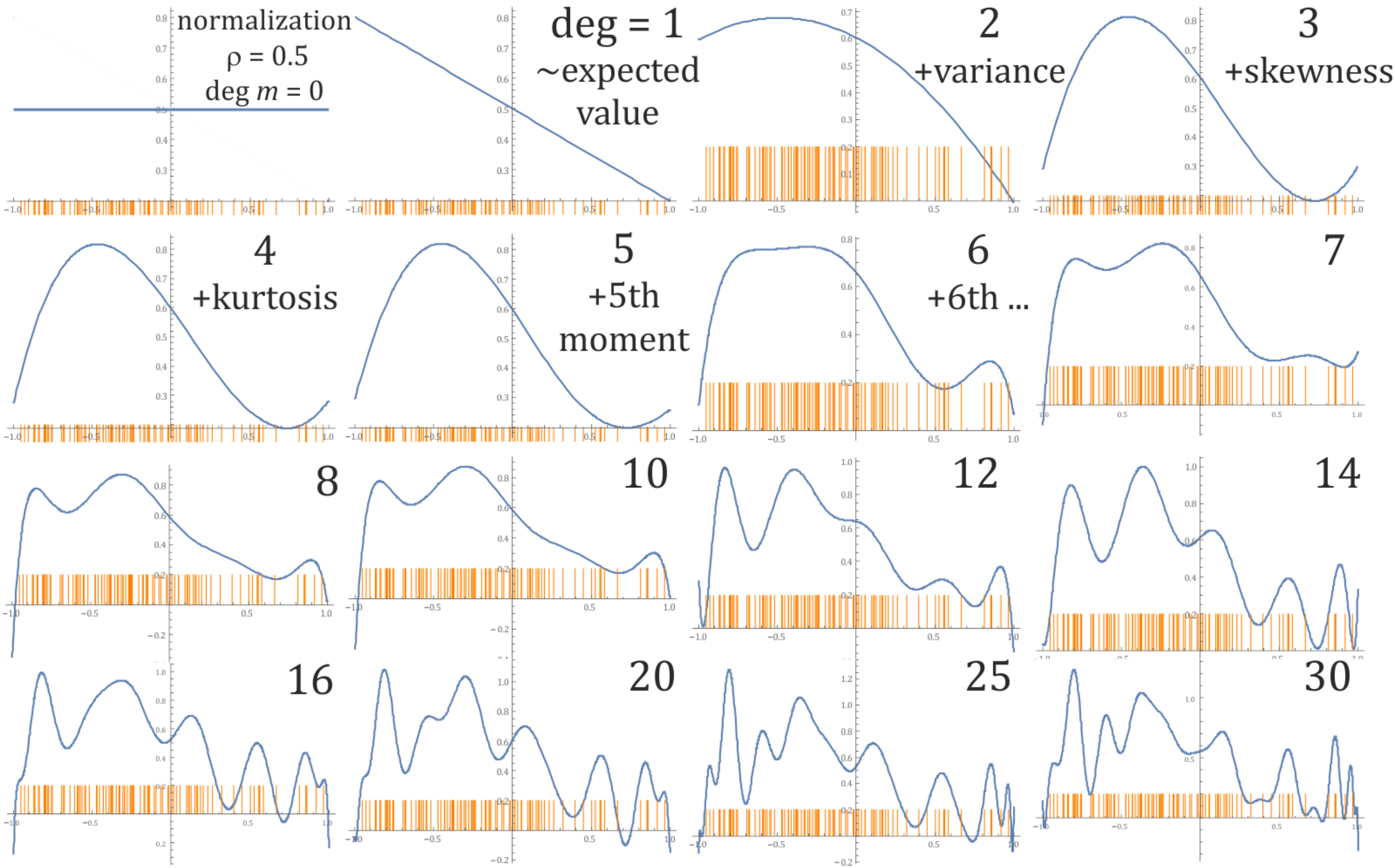
	Moments/cumulants	$\boldsymbol{\rho(x) = \sum_f a_f\, f(x)}$	**Machine learning**
# parameters	low – rough	from low to high	high - accurate
estimation	e.g. $m_k = \frac{1}{\lvert X\rvert}\sum_{x\in X} x^k$	$a_f = \frac{1}{\lvert X\rvert}\sum_{x\in X} f(x)$	usually iteration
Interpretable?	yes	Yes: mixed 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

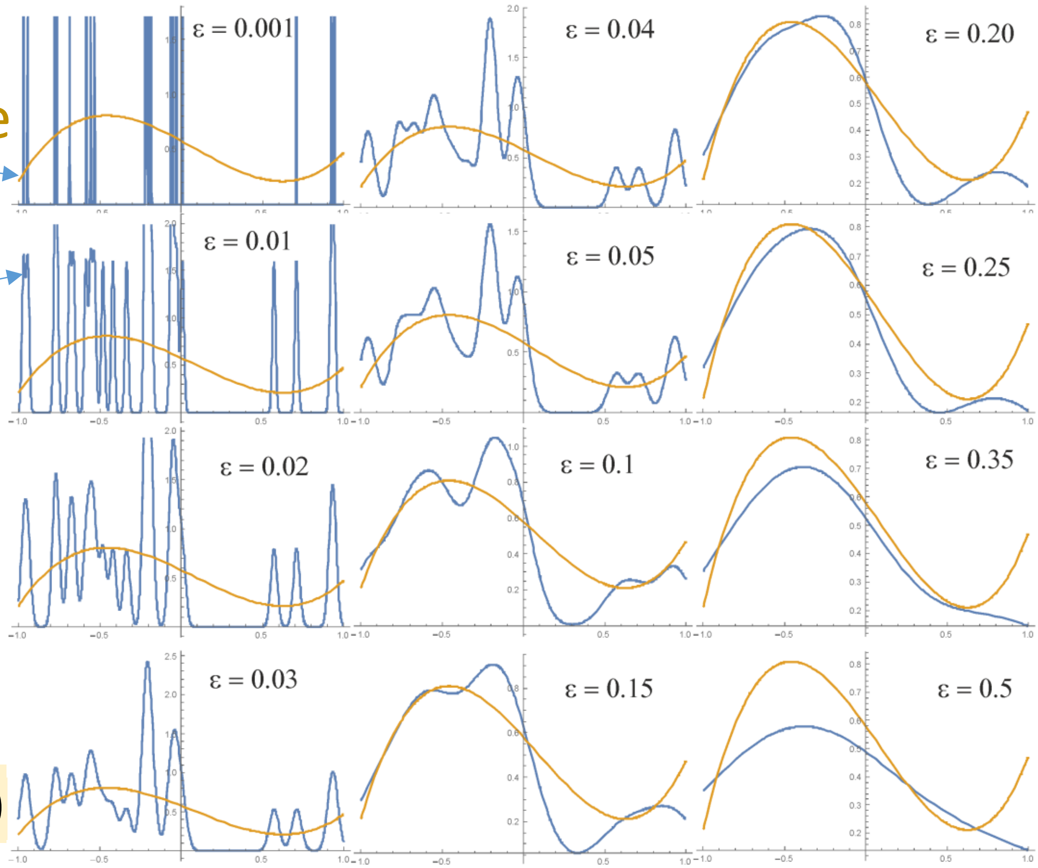


$n = 100$ size **1D sample** (from degree = 3), **density estimated as polynomial**:

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



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

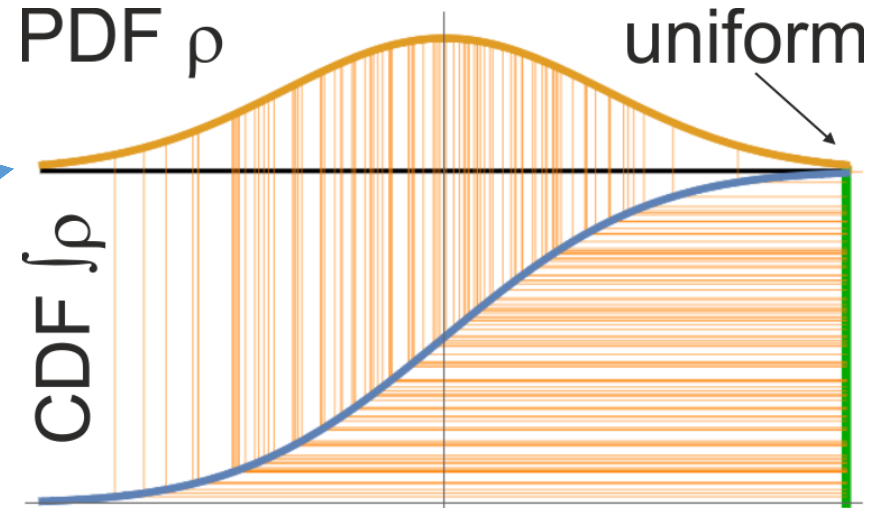
Removing $\lim_{\epsilon\to 0} \|g_\epsilon\|^2 = \infty$ which does not affect parameters $\boldsymbol{a}$

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

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

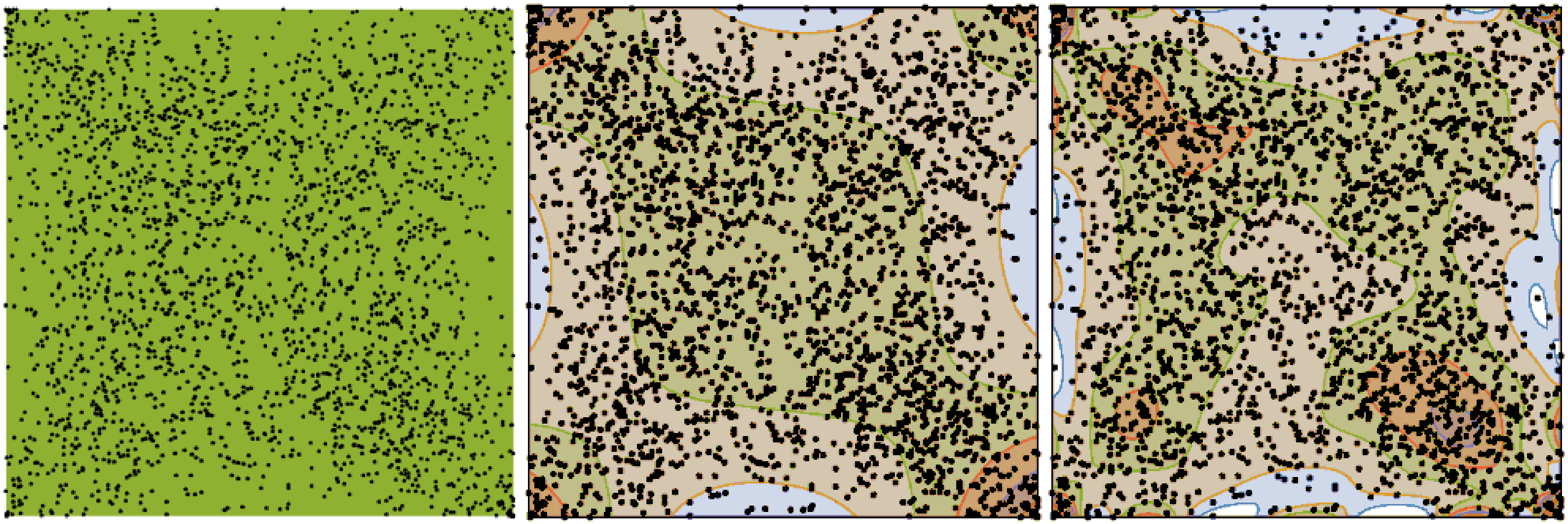
$$\text{minimum:} \quad \partial_{a_j} = 0 \quad \Rightarrow \quad a_{\boldsymbol{j}} = \frac{1}{n}\sum_{x\in X} f_{\boldsymbol{j}}(x)$$

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: $\ln(v_{t+1}/v_t)$)



$(x^{t\text{-}1}, x^t)$ 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

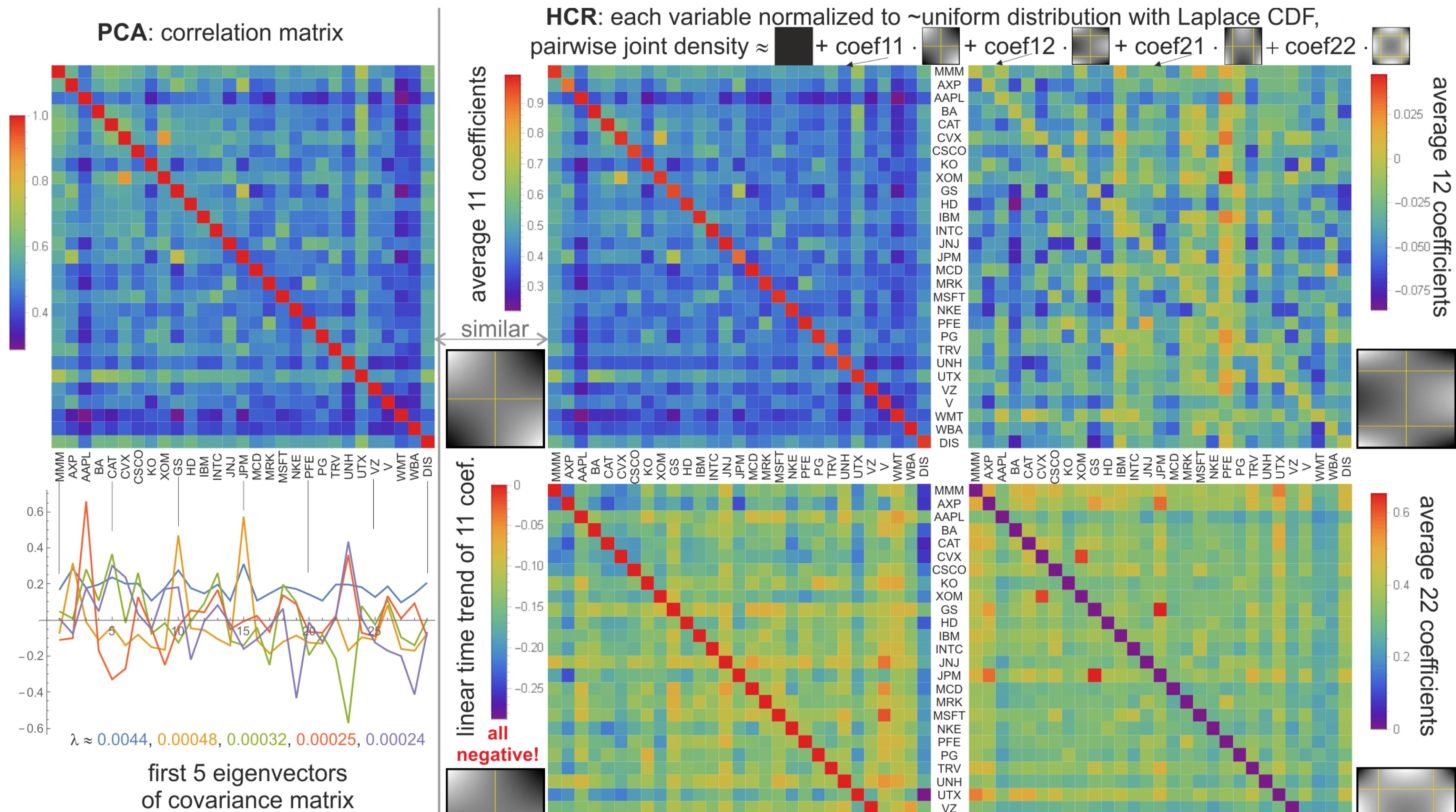


independent ~ correlation coef. **further statistical dependencies** var-var

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

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

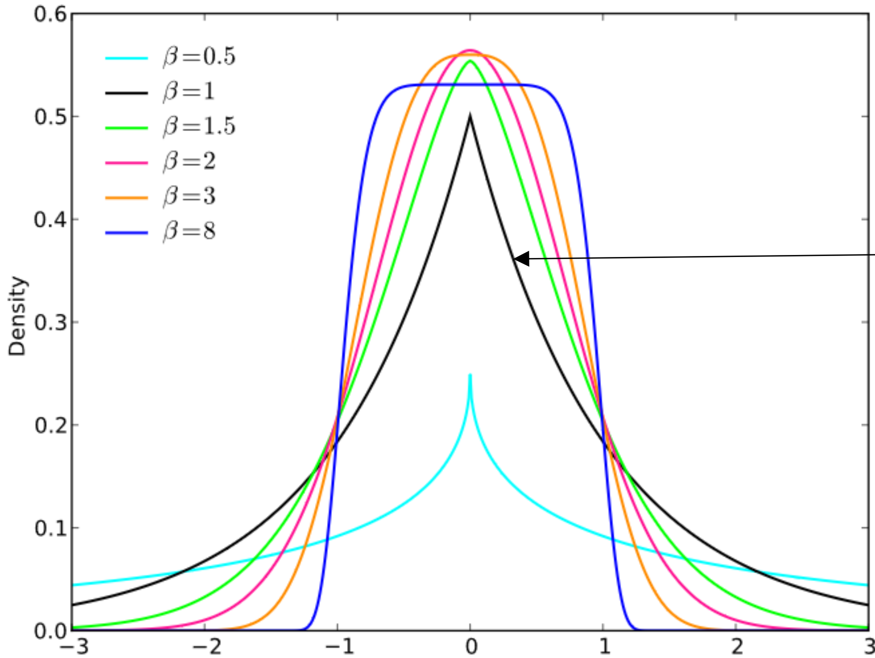
Standard: pairwise correlation "11", here: also higher, "triple+"wise, time dependent



Dow Jones 29 companies, 10 year daily
arXiv:1807.04119

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

Generalized normal distribution/EPD $\rho \sim \exp\left(-|x|^{\beta}\right)$



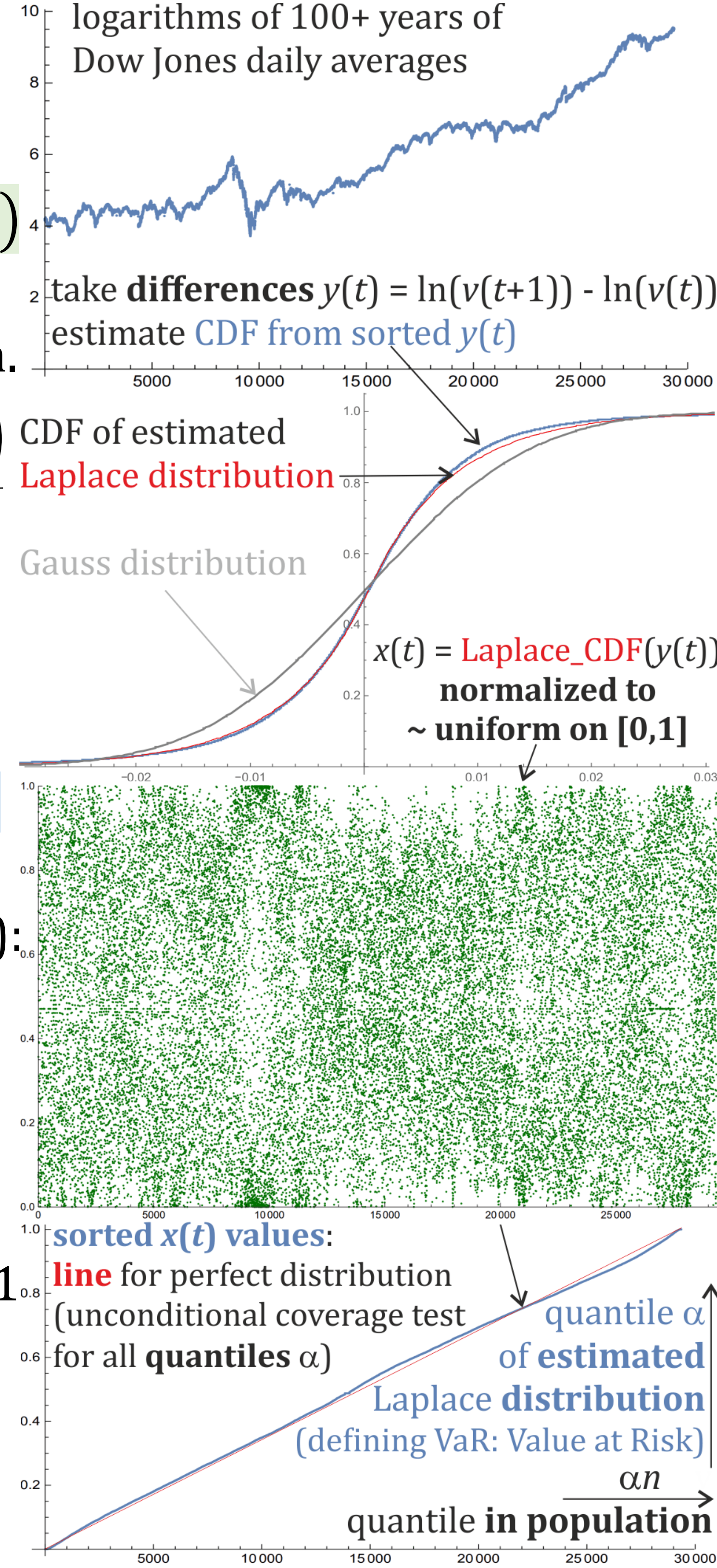
Laplace, MLE estim.

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

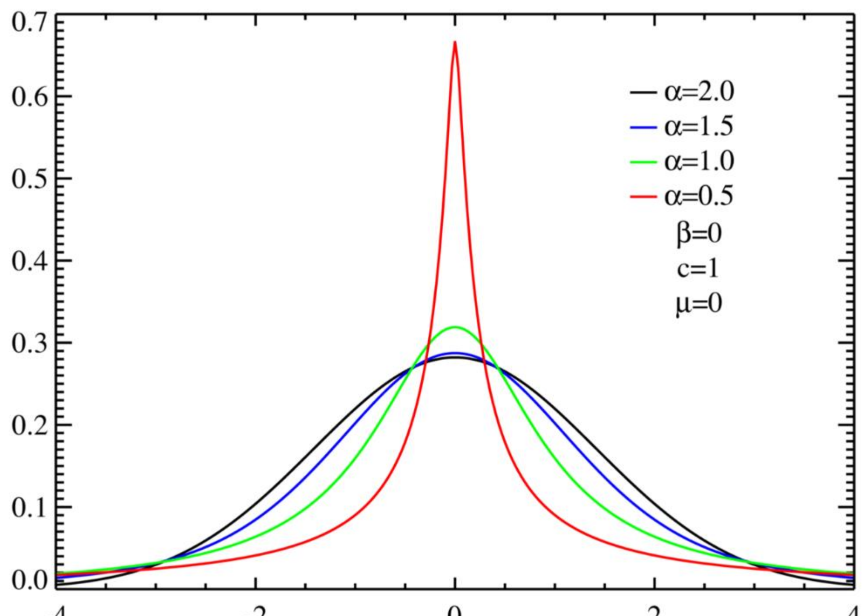
$$\hat{\mu} = \text{median}$$

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

Normalization contains tail model



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



Student's t-dist.:

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

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

Cauchy for $v = 1$

∞ moments $\geq v$

Adaptivity: models evolving with time

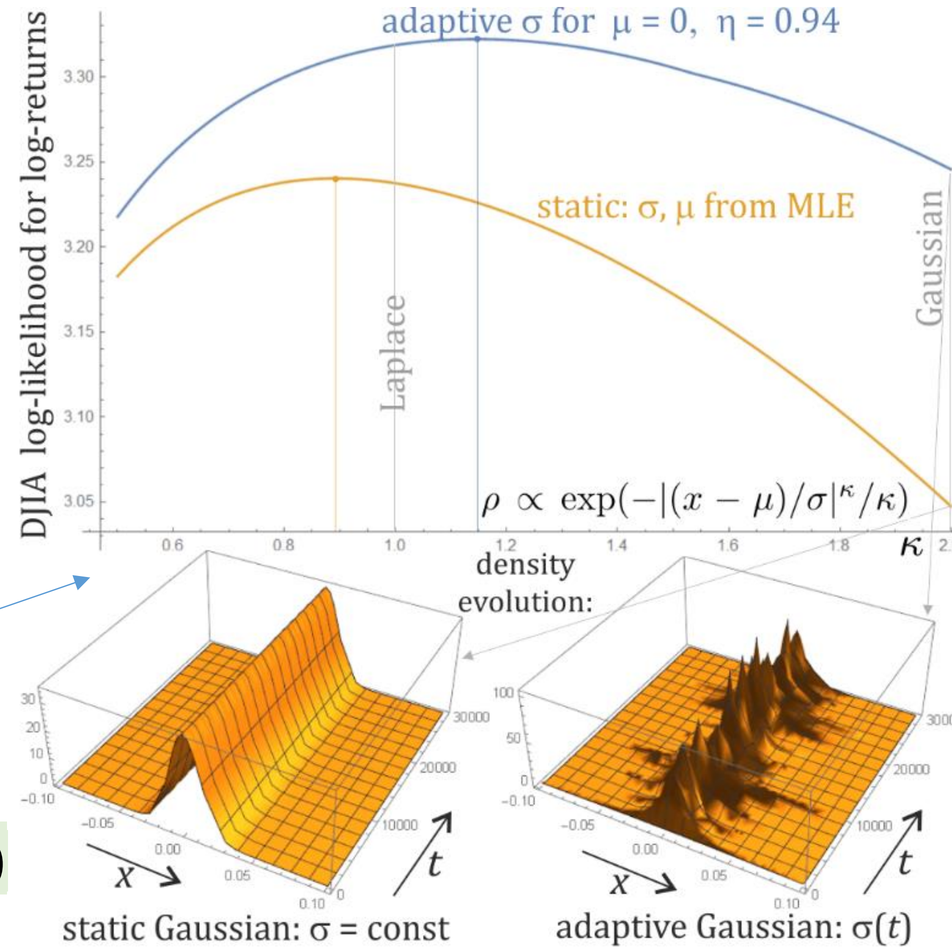
We can **normalize** with $x_t = \mathrm{CDF}_t(y_t)$

e.g. Gaussian with varying σ like in ARCH

e.g. average → exponential moving average

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

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



Optimizing **exponential moving criterion**:

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

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

[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:

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

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

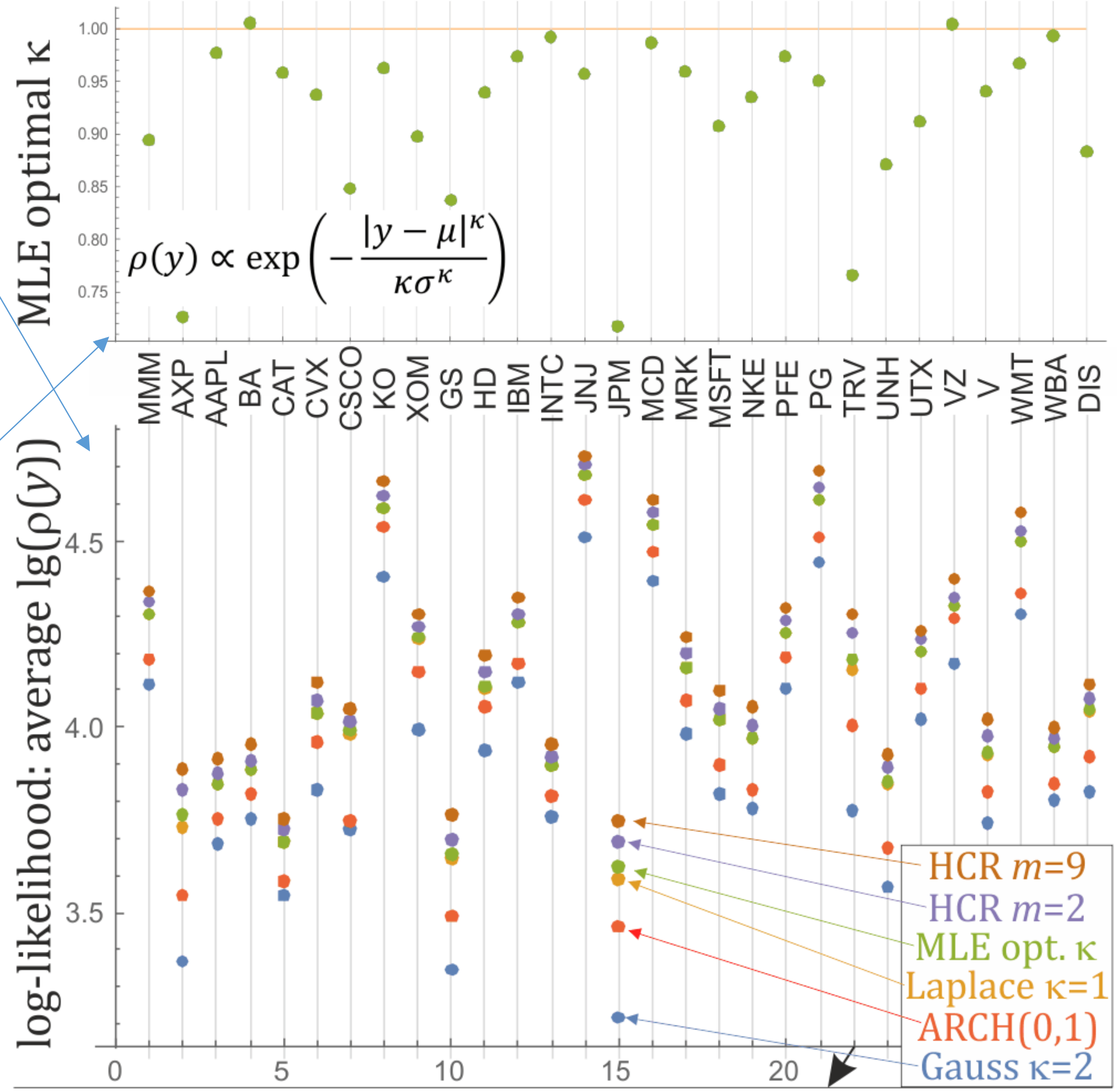
to $x \sim$ uniform on [0,1] distribution:

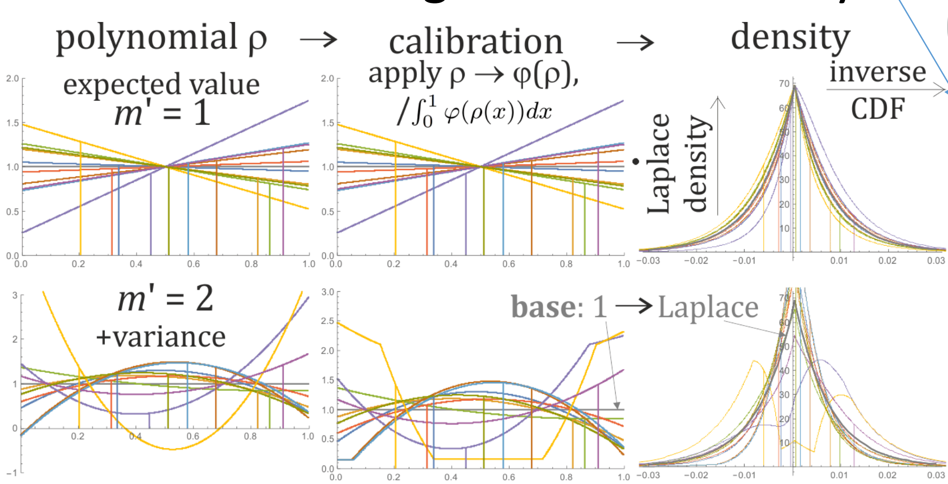
$x^t = \mathrm{CDF}_{\text{parametric}}(y^t)$

HCR: Fit degree m polynomial

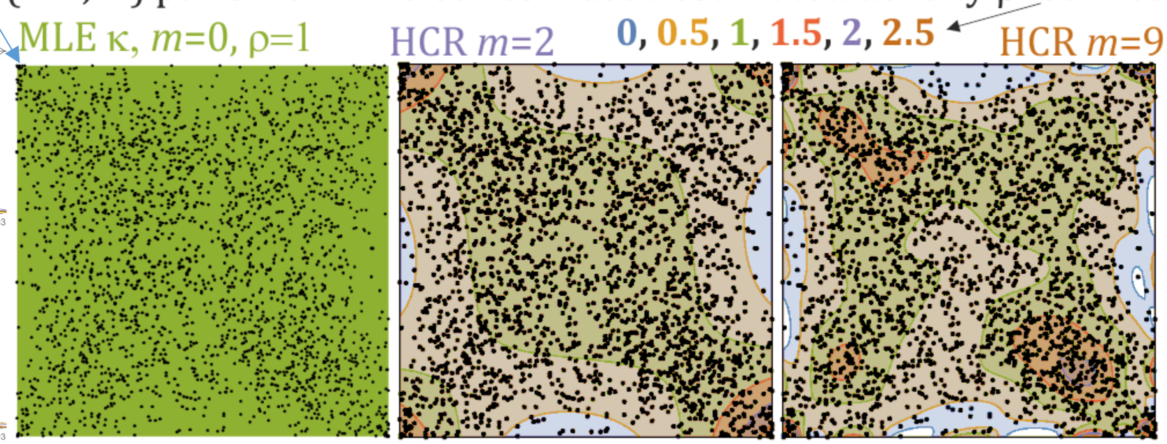
e.g. to $(\boldsymbol{x^{t-1}}, \boldsymbol{x^t})$ **joint distribution**

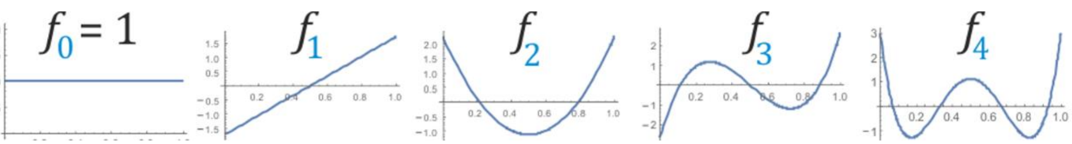
can be evolving for nonstationary





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





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

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

⇒ conditional distributions without Bayes

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

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

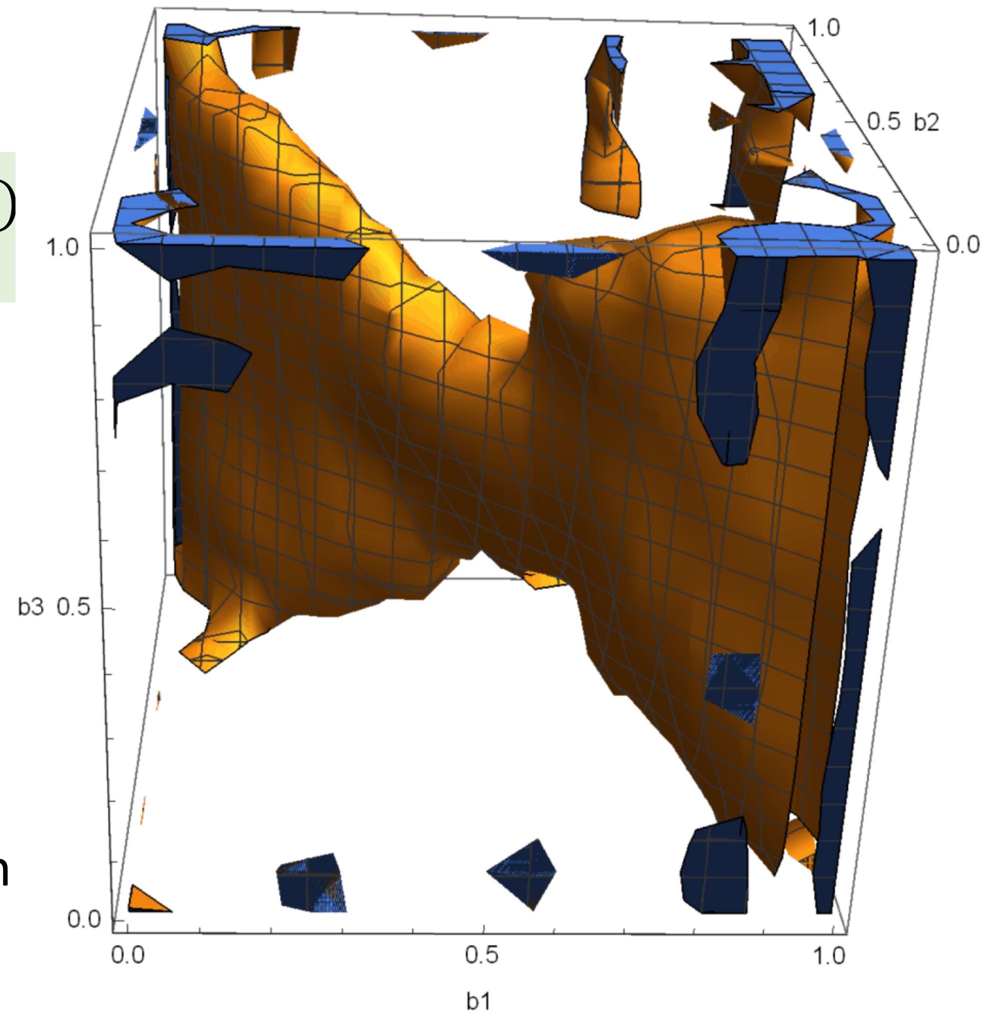
For considered **statistical dependencies:**

basis B of considered **mixed moments**

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

$$B\ni \boldsymbol{j}=(000\mathbf{2}000\mathbf{2}000)$$

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

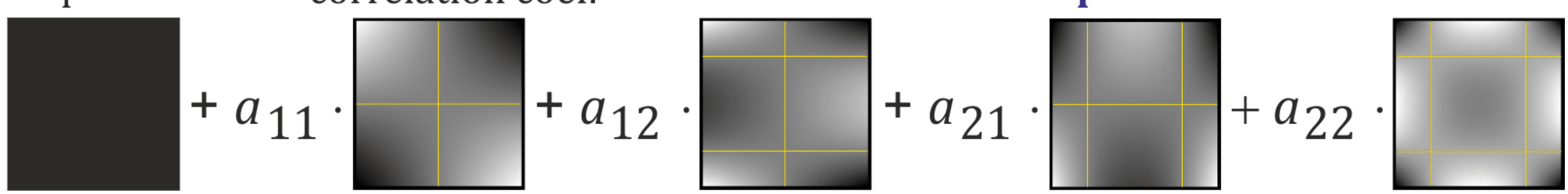


$a_{\boldsymbol{j}}=$ **average** ...

- 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 series**

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

pair-wise joint density $\approx$ [independent] $+\ a_{11}\cdot$ [ ] $+\ a_{12}\cdot$ [ ] $+\ a_{21}\cdot$ [ ] $+\ a_{22}\cdot$ [ ]



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

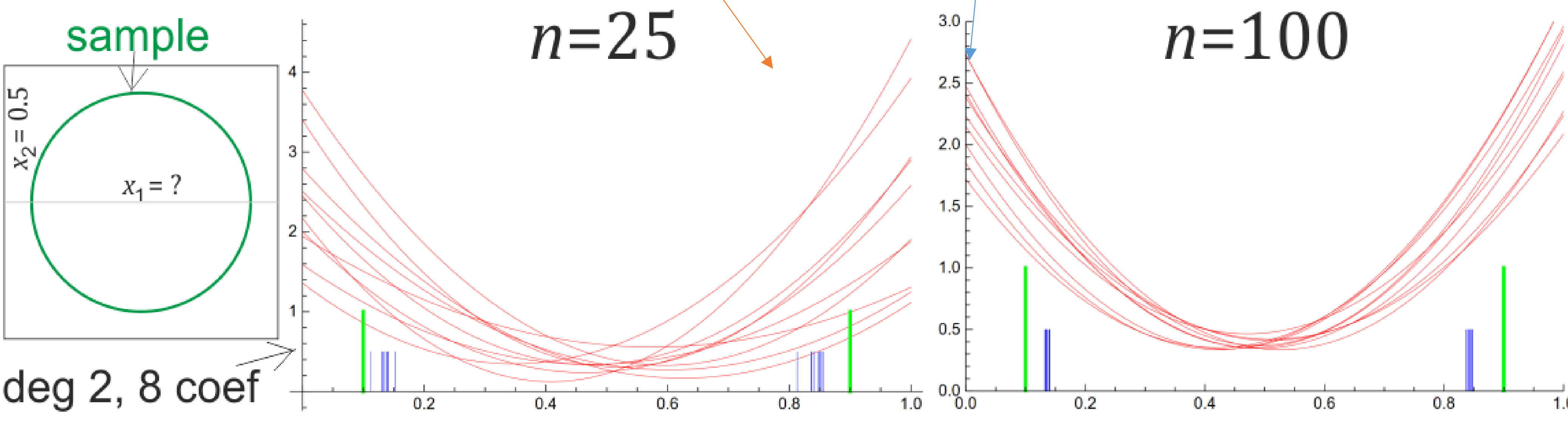
However, sometimes **ambiguity**, e.g. circle as sample below we can handle.

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** ...

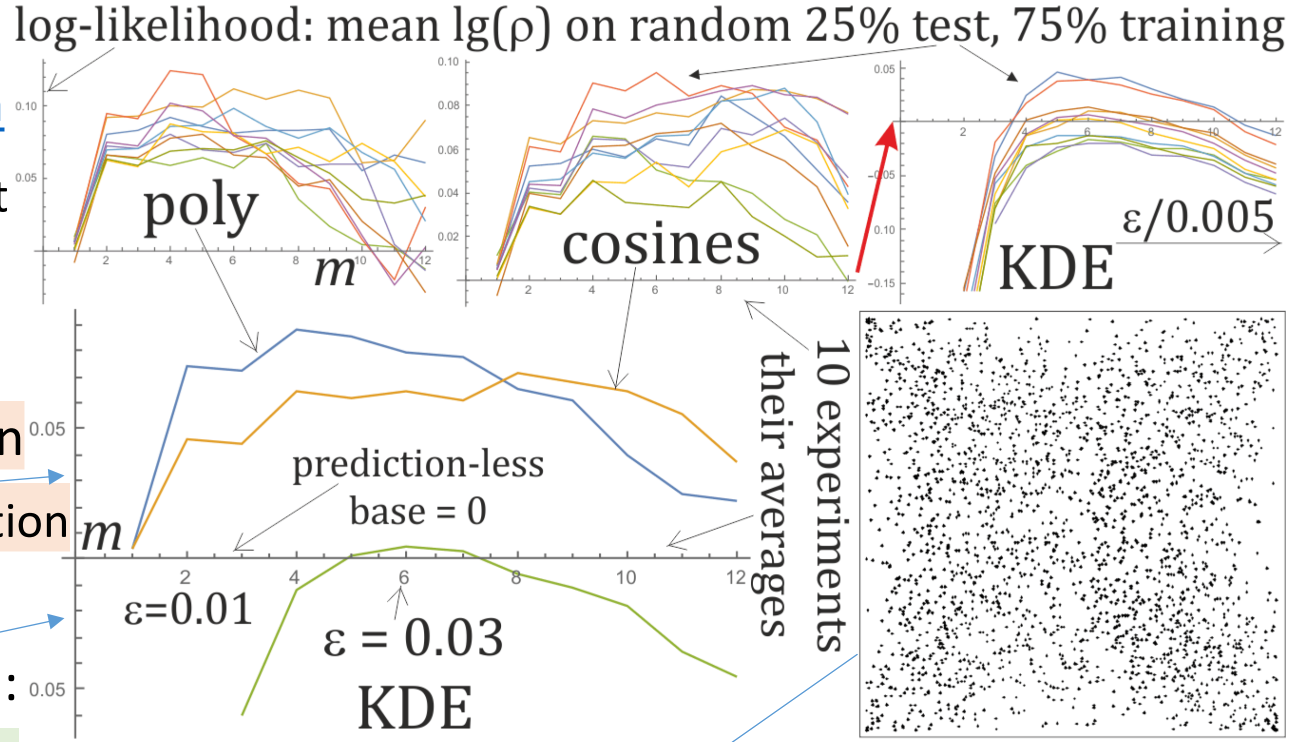


KDE – [kernel density estimation]

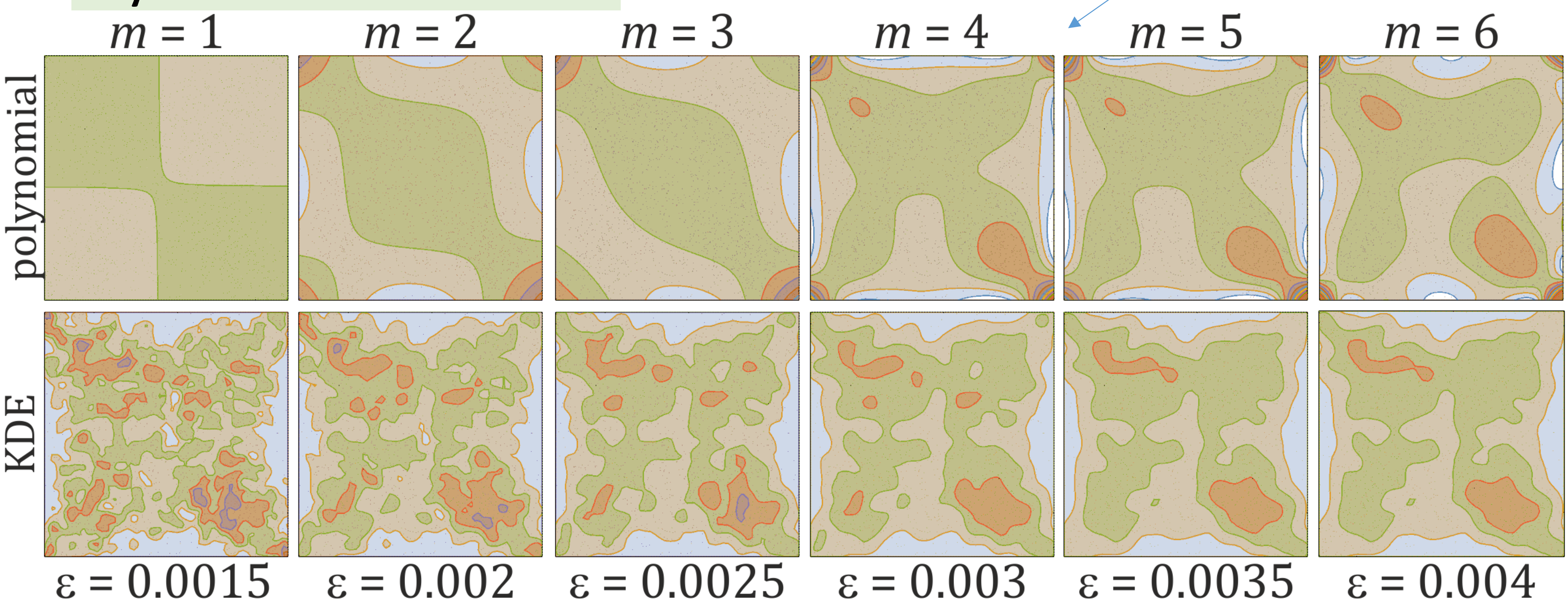
e.g. ϵ-radius Gaussian in each point

- 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 $\epsilon \to 0$



Having density model,

we can cheaply

normalize

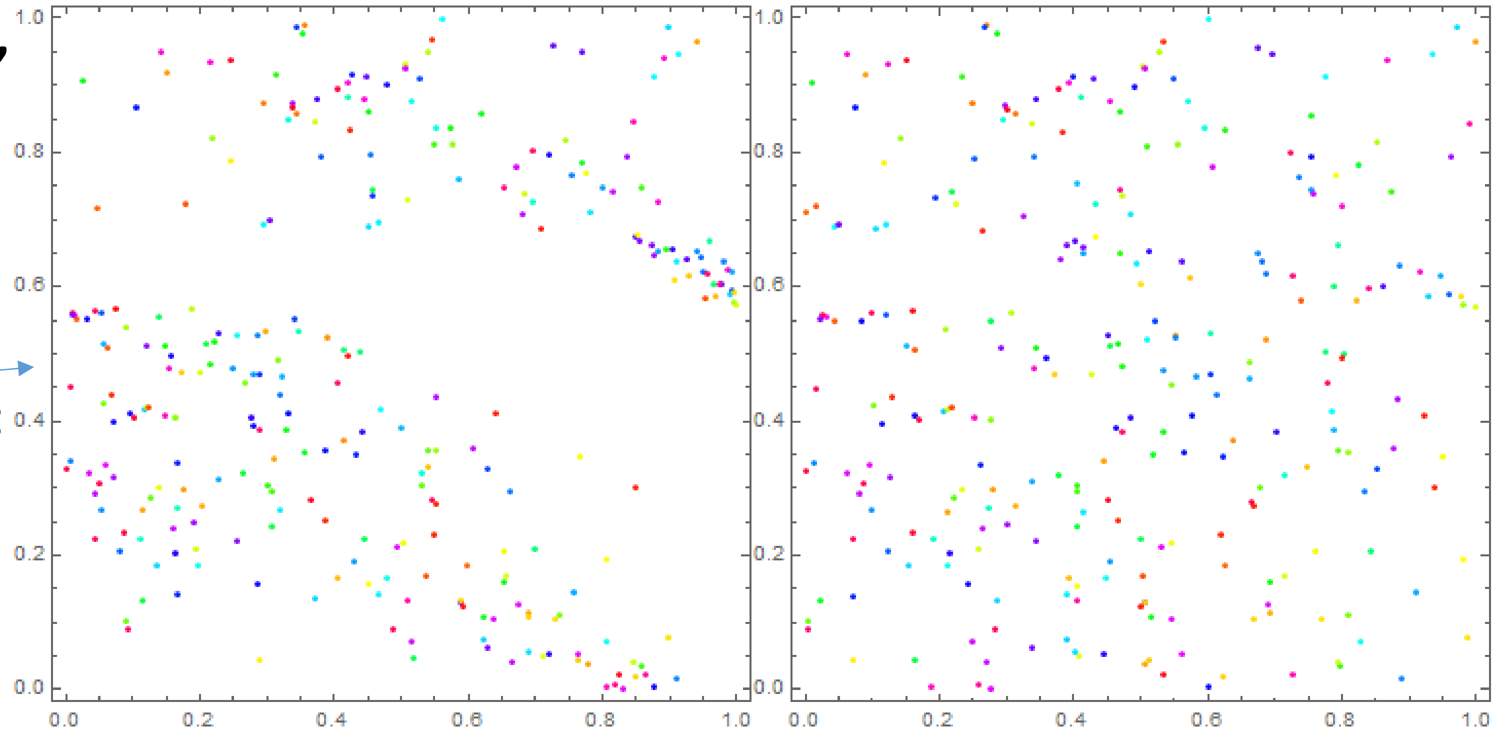
e.g. to uniform

$x \to CDF_y(x)$ 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 ?

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

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

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,

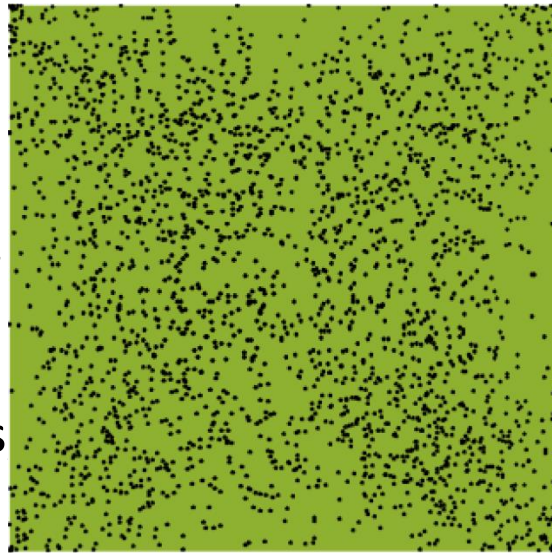
Cheat to estimate

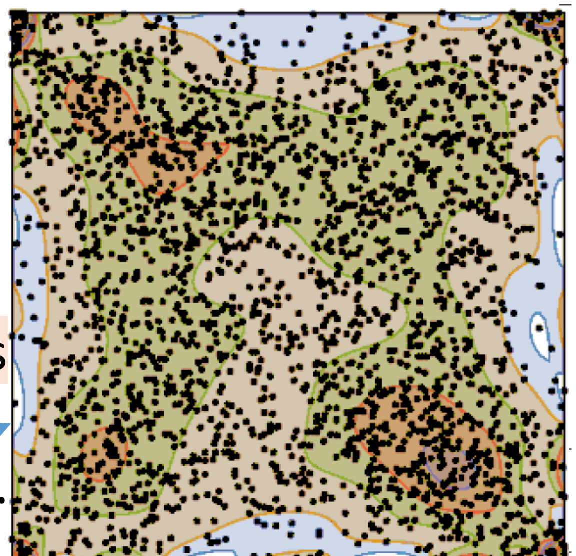
also to adapt,

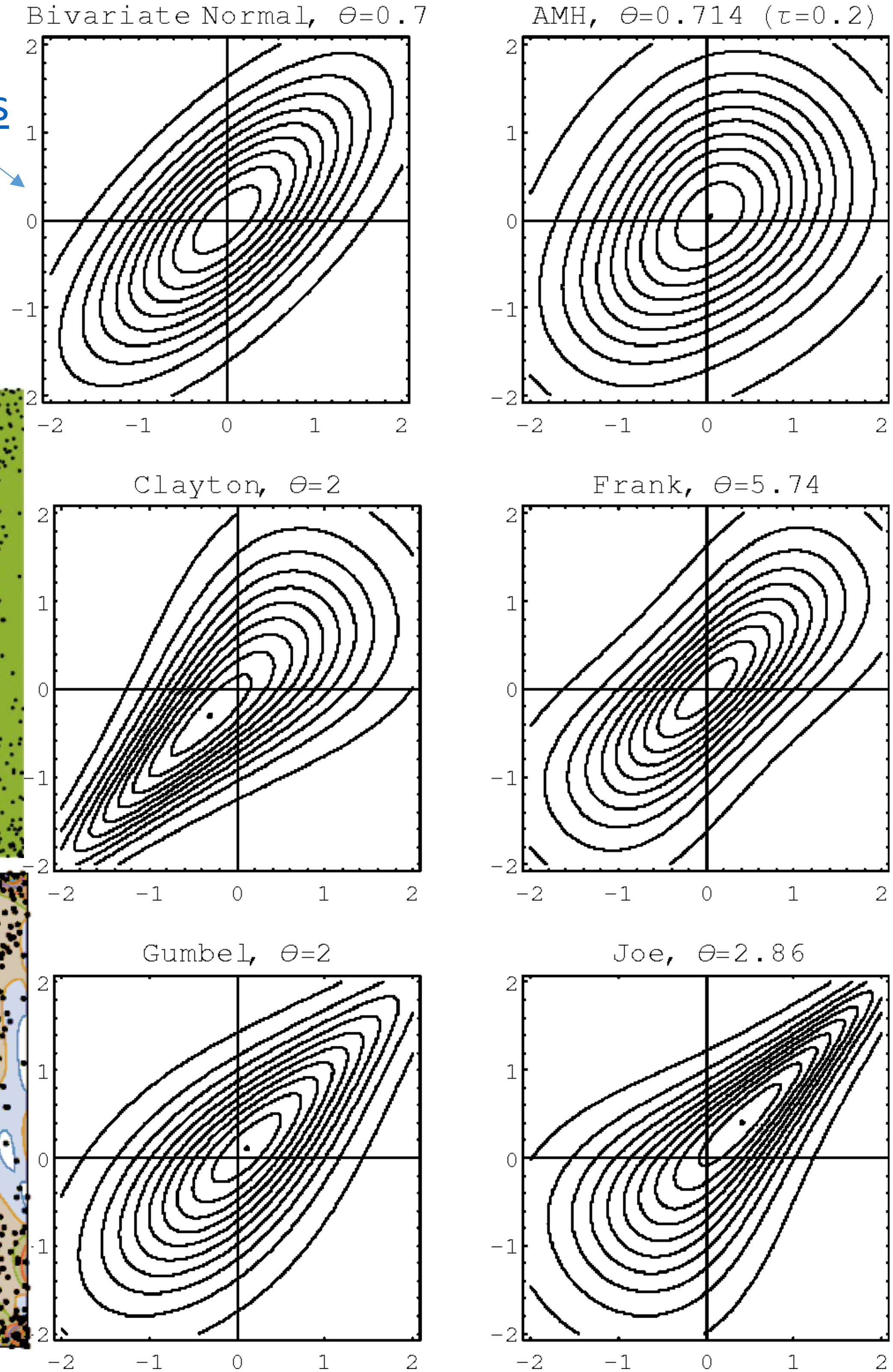
missing data

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

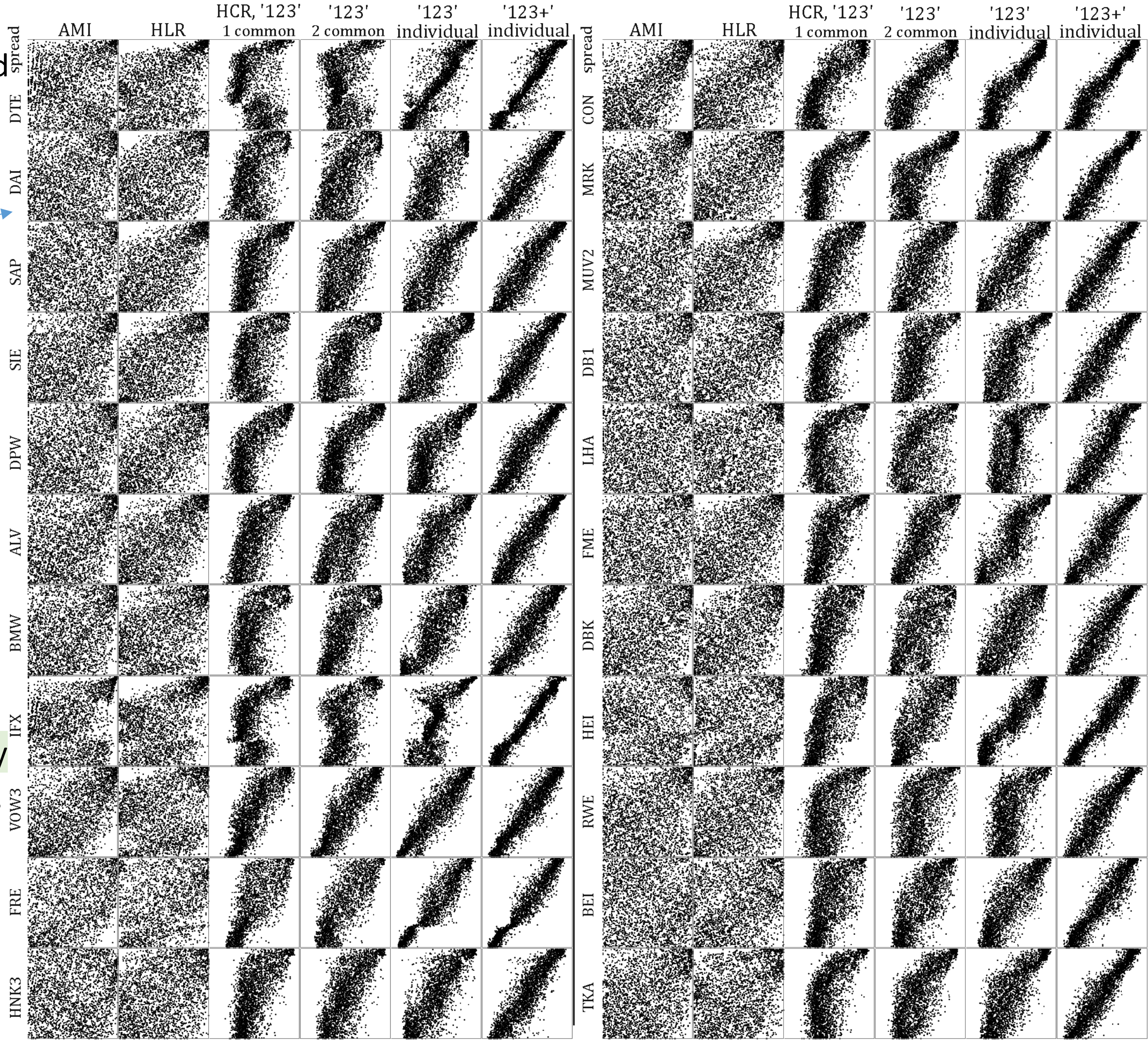
skewness, kurtosis…

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

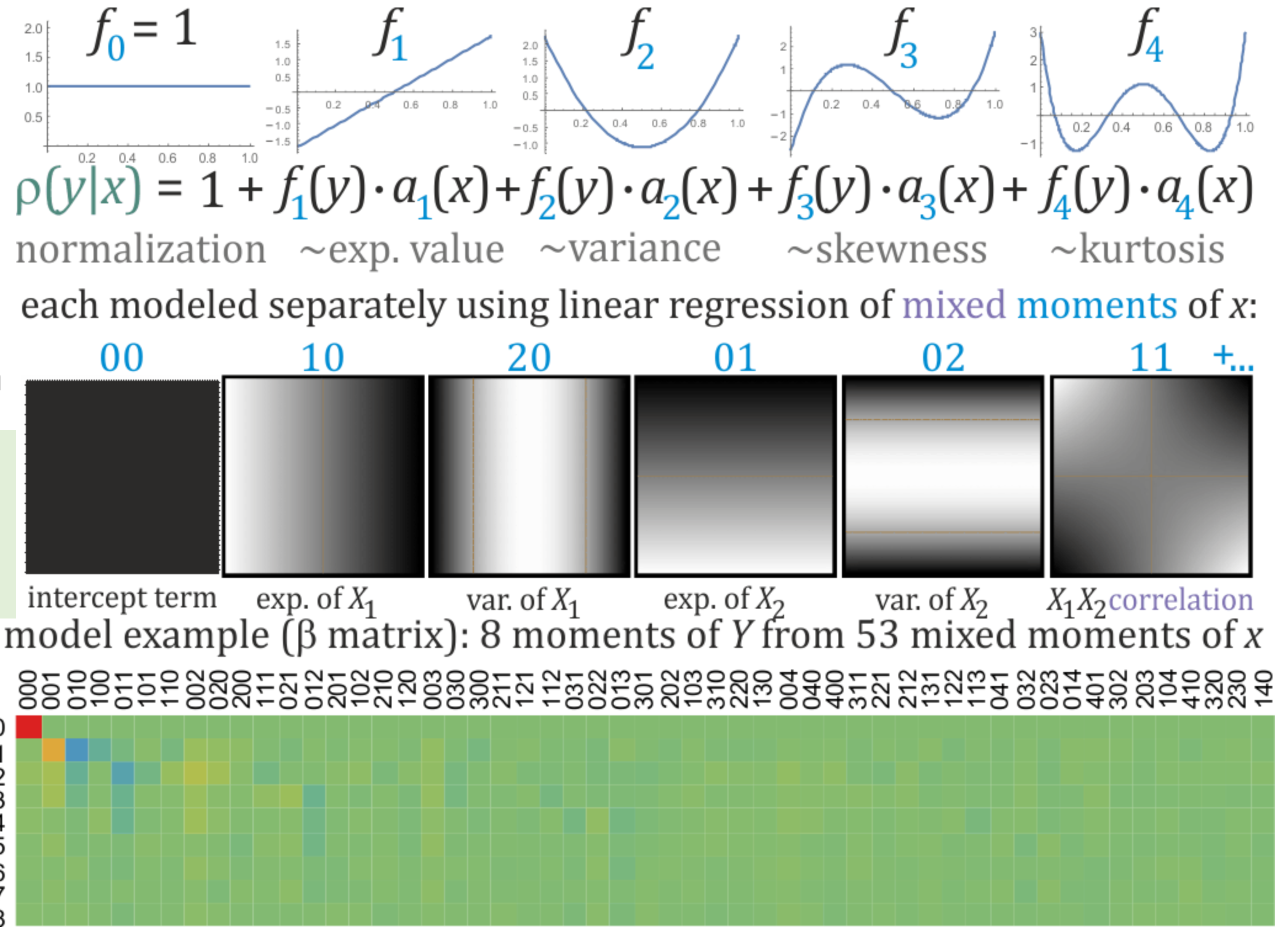
each separately as linear combination

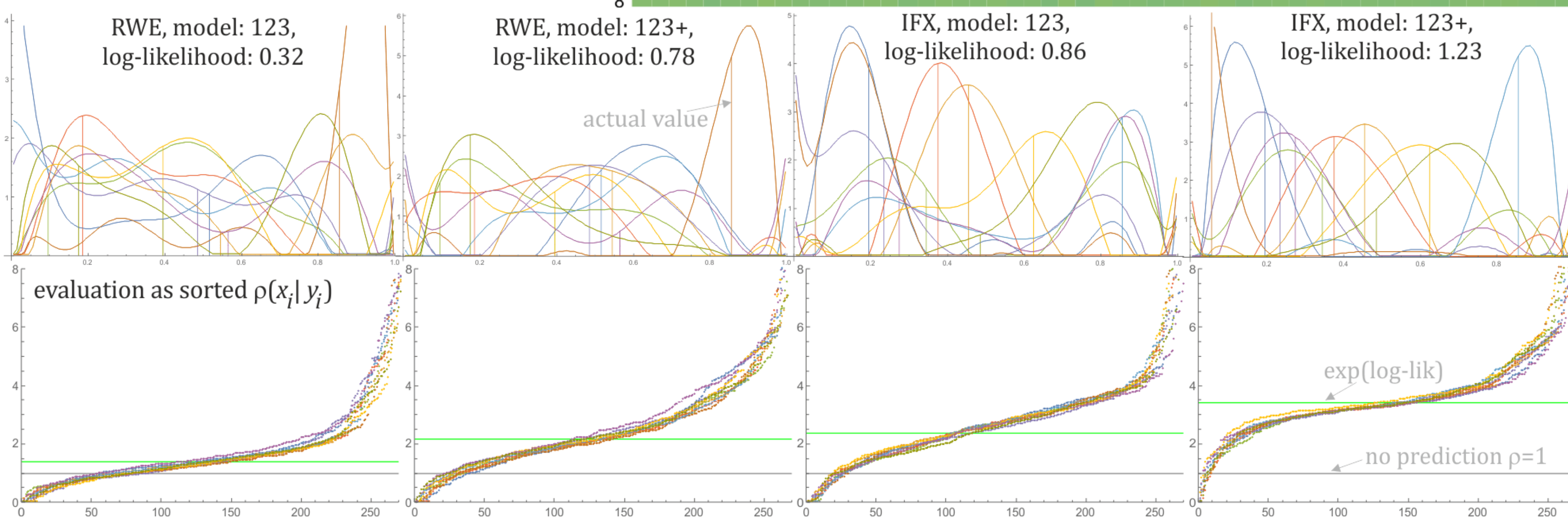
then combine into predicted density.

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

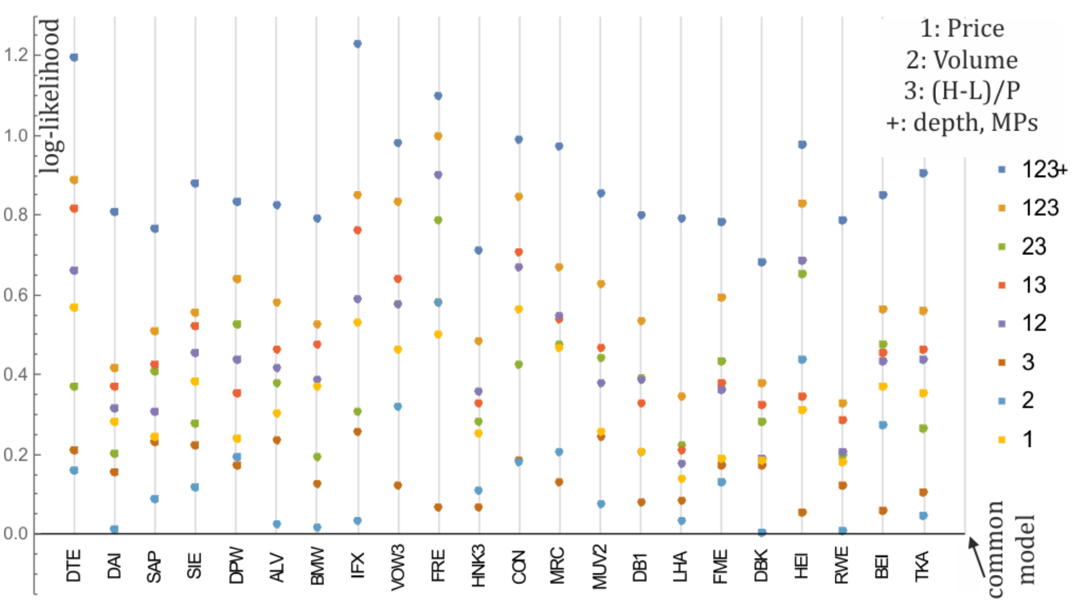
$$\rho(y|x) = \max(\tilde{\rho}(y|x), 0.03)\,/N$$

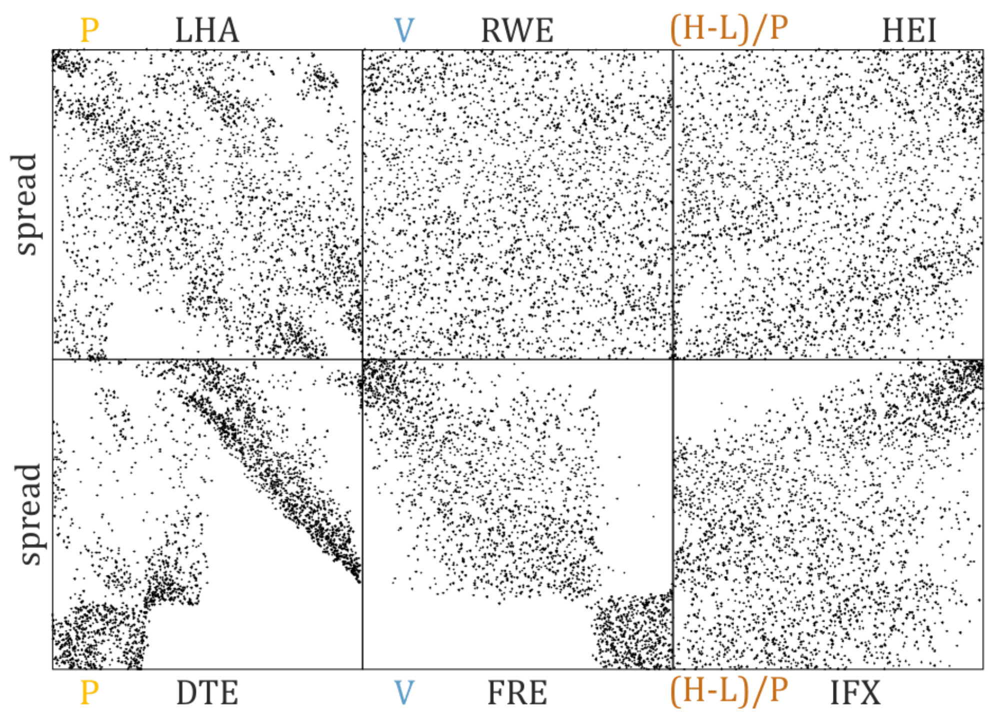
Examples normalization

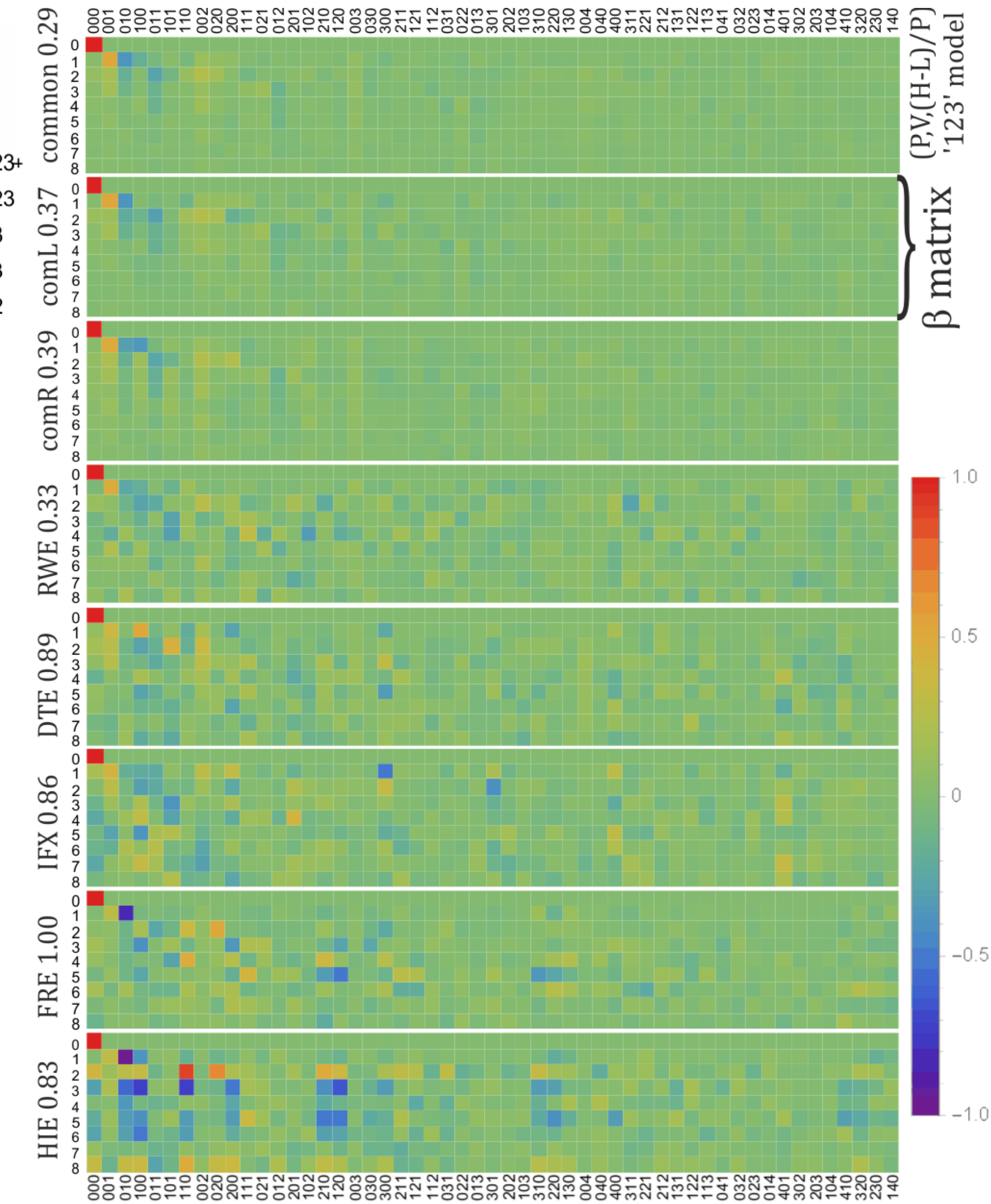




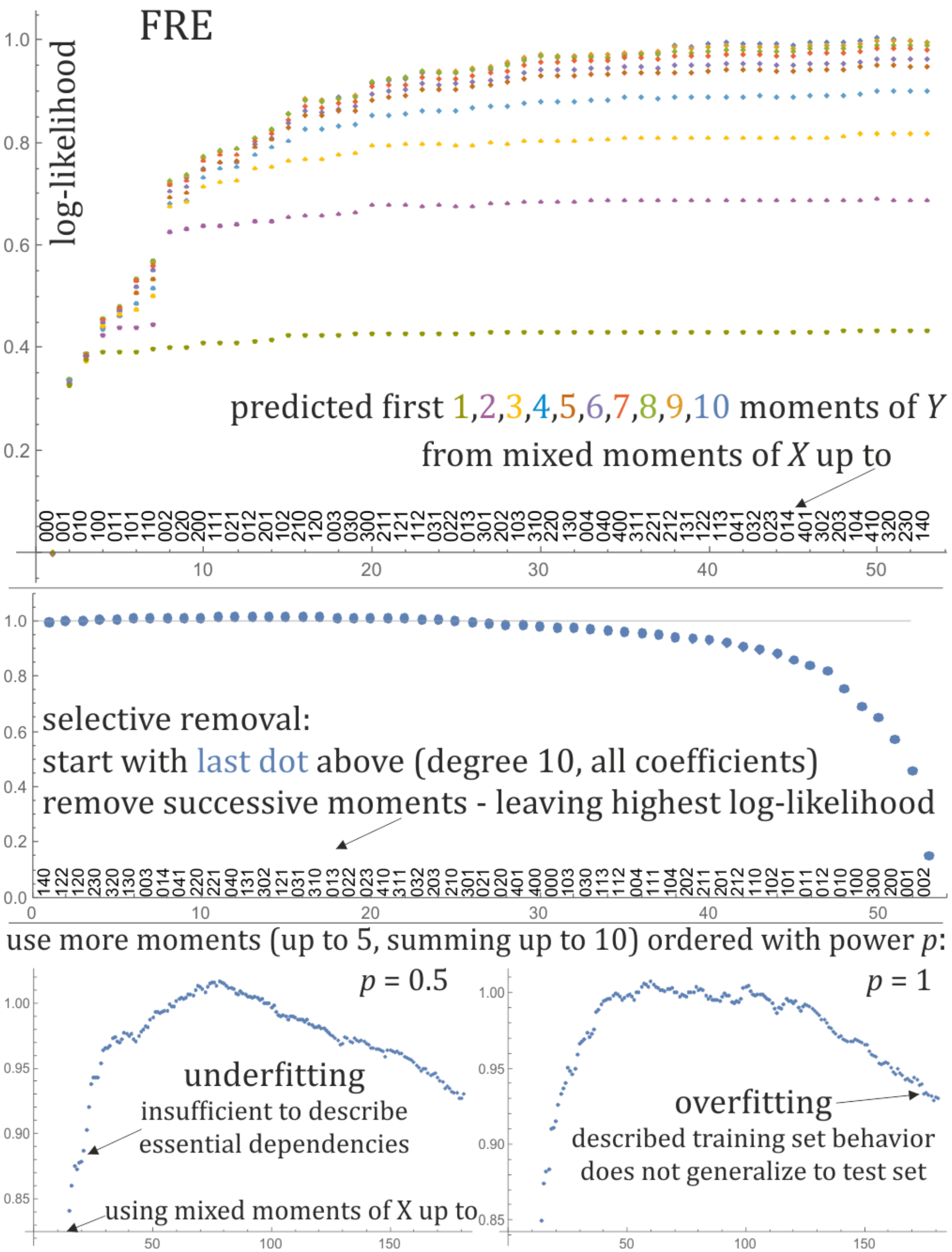
Large differences between companies – individual models give much better evaluation



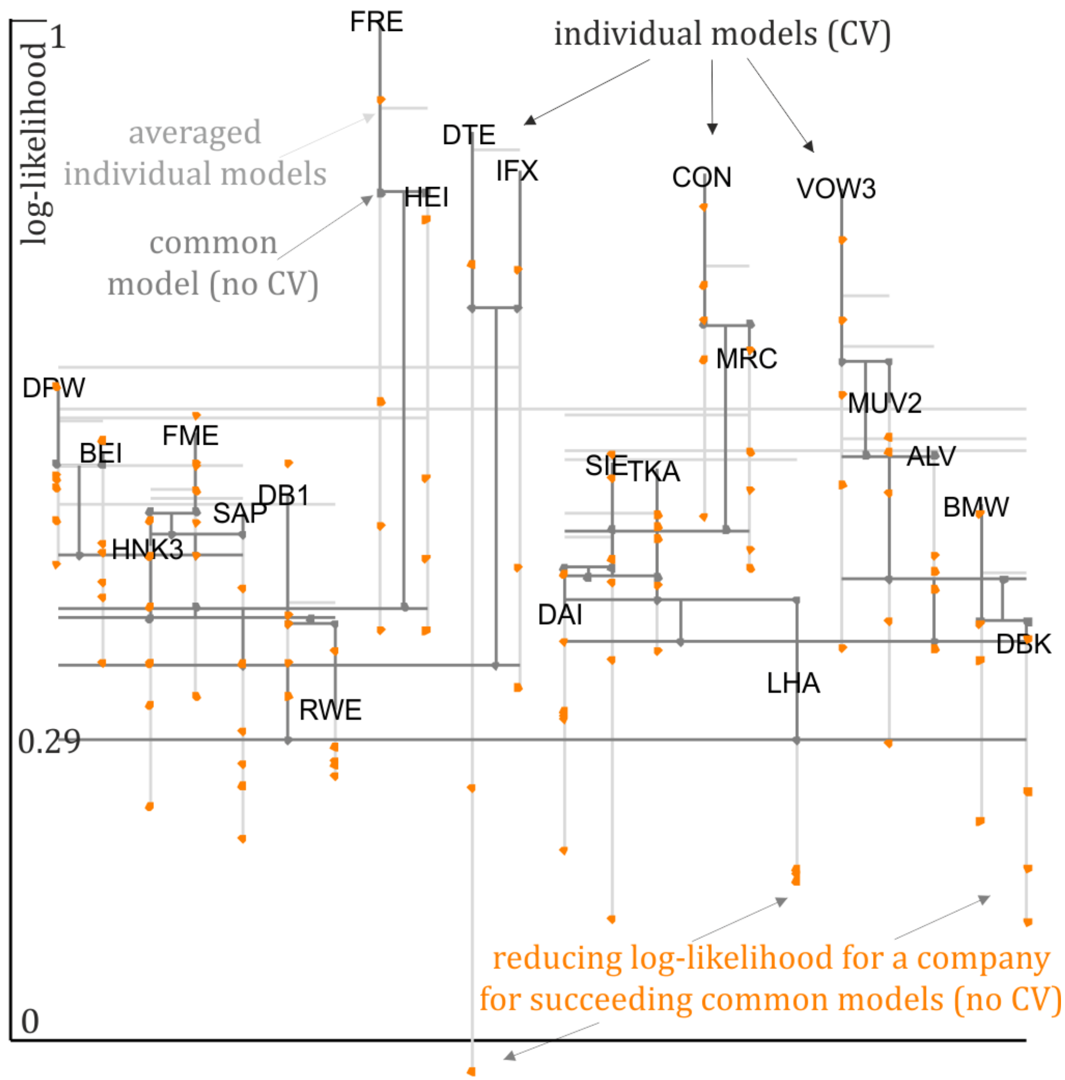




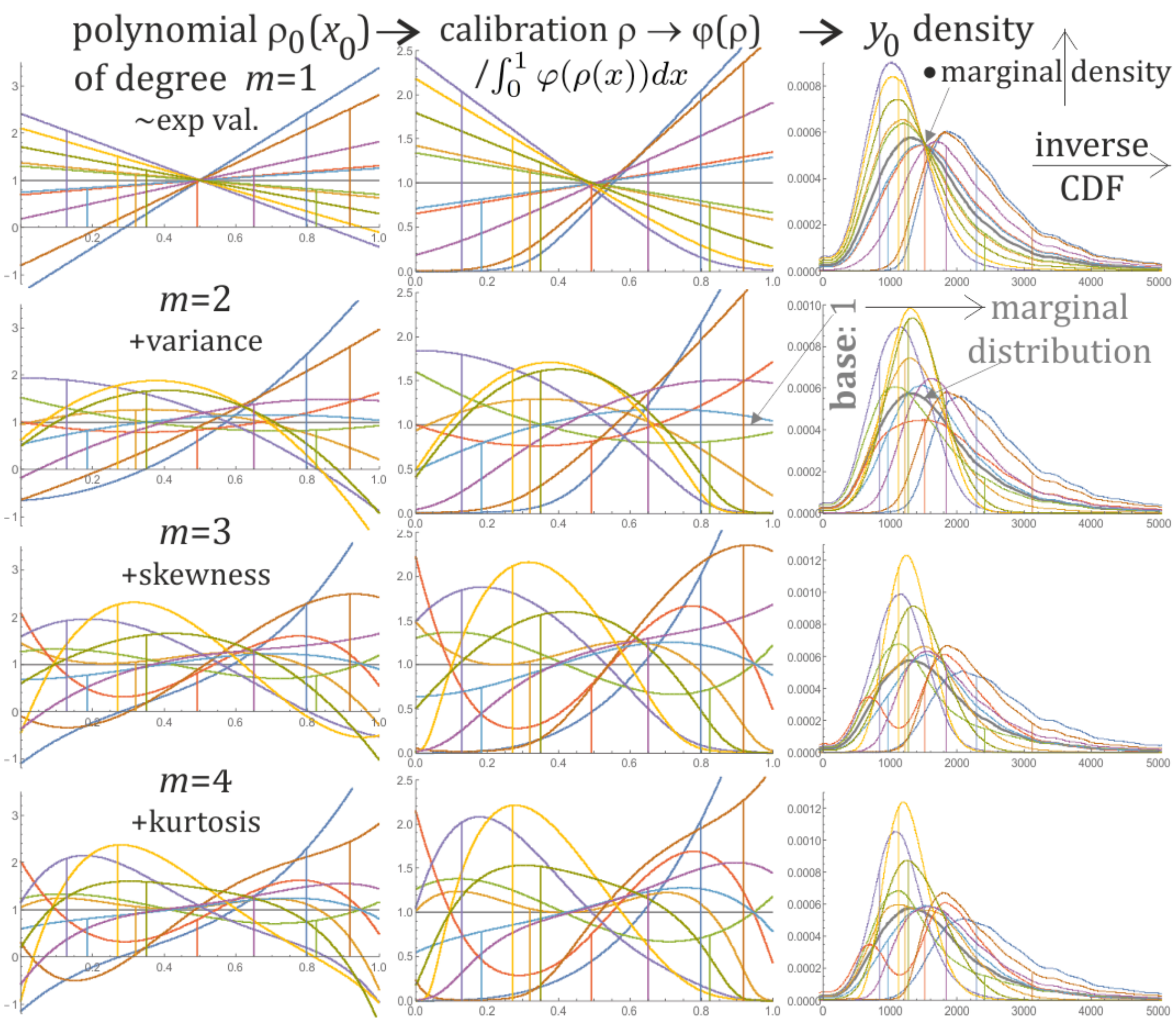
Choosing model size: predict ≈ 8 moments basis of mixed moments – difficult problem

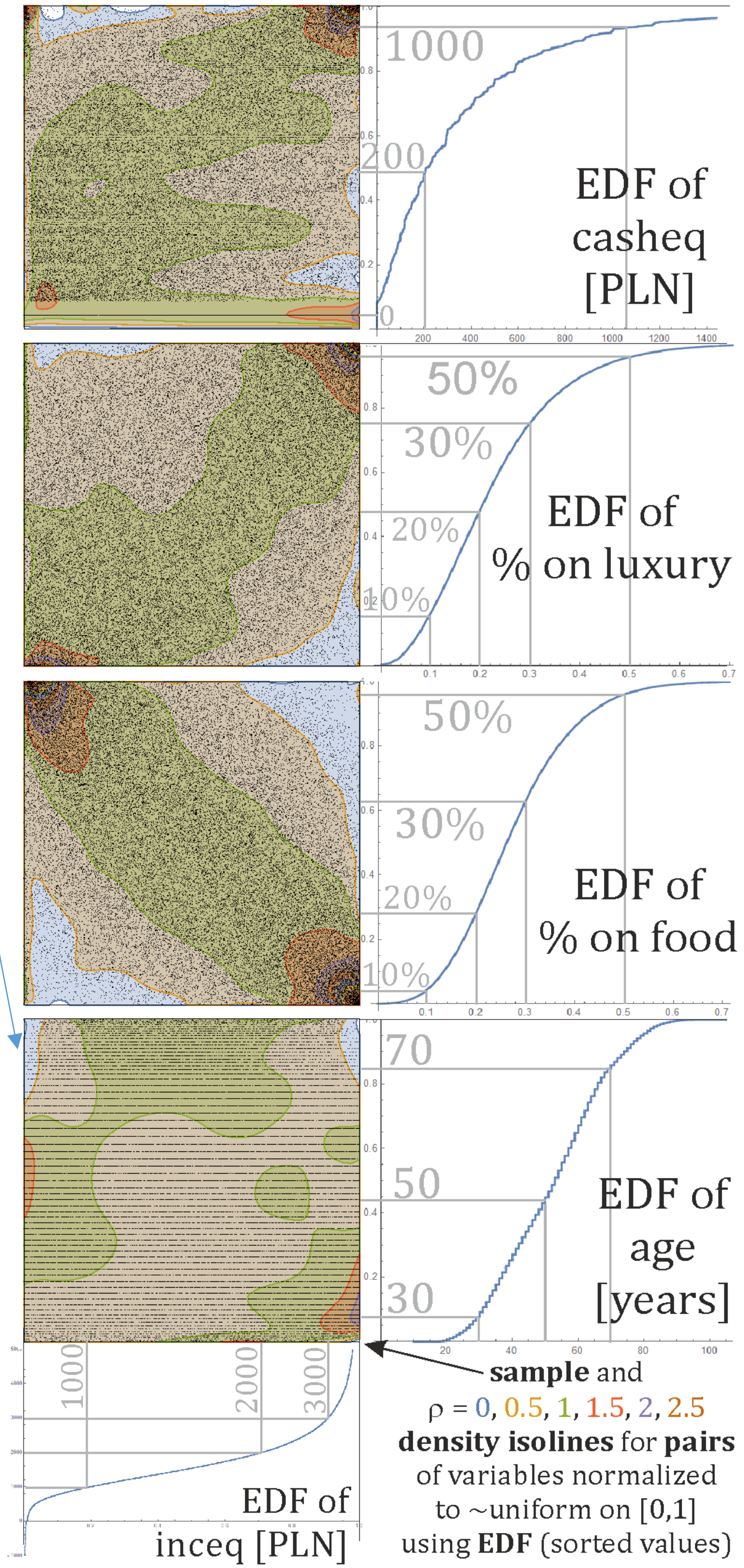


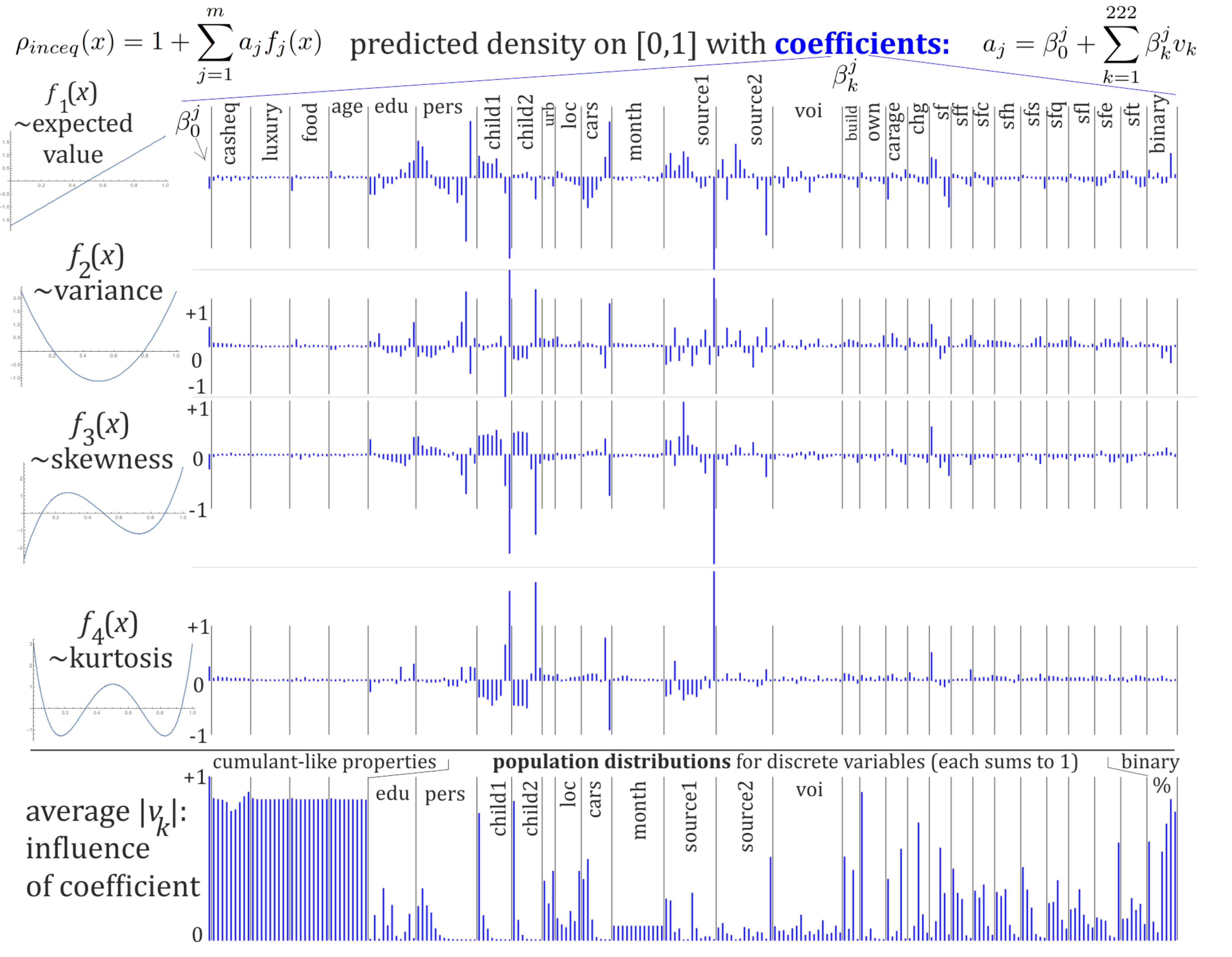
Universality – searching for common models with lowest evaluation loss



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: $a_j = \beta_0^j + \sum_{k=1}^{222} \beta_k^j v_k$
β_k^j
β_0^j
$f_1(x)$ ~expected value
$f_2(x)$ ~variance
$f_3(x)$ ~skewness
$f_4(x)$ ~kurtosis
casheq
luxury
food
age
edu
pers
child1
child2
urb
loc
cars
month
source1
source2
voi
build
own
carage
chg
sf
sff
sfc
sfh
sfs
sfq
sfl
sfe
sft
binary
+1
0
-1
average $|v_k|$: influence of coefficient
cumulant-like properties
population distributions for discrete variables (each sums to 1)
binary
%

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

(expected value, sqrt(var)) of predicted

Log-likelihood evaluation of 35 variables:

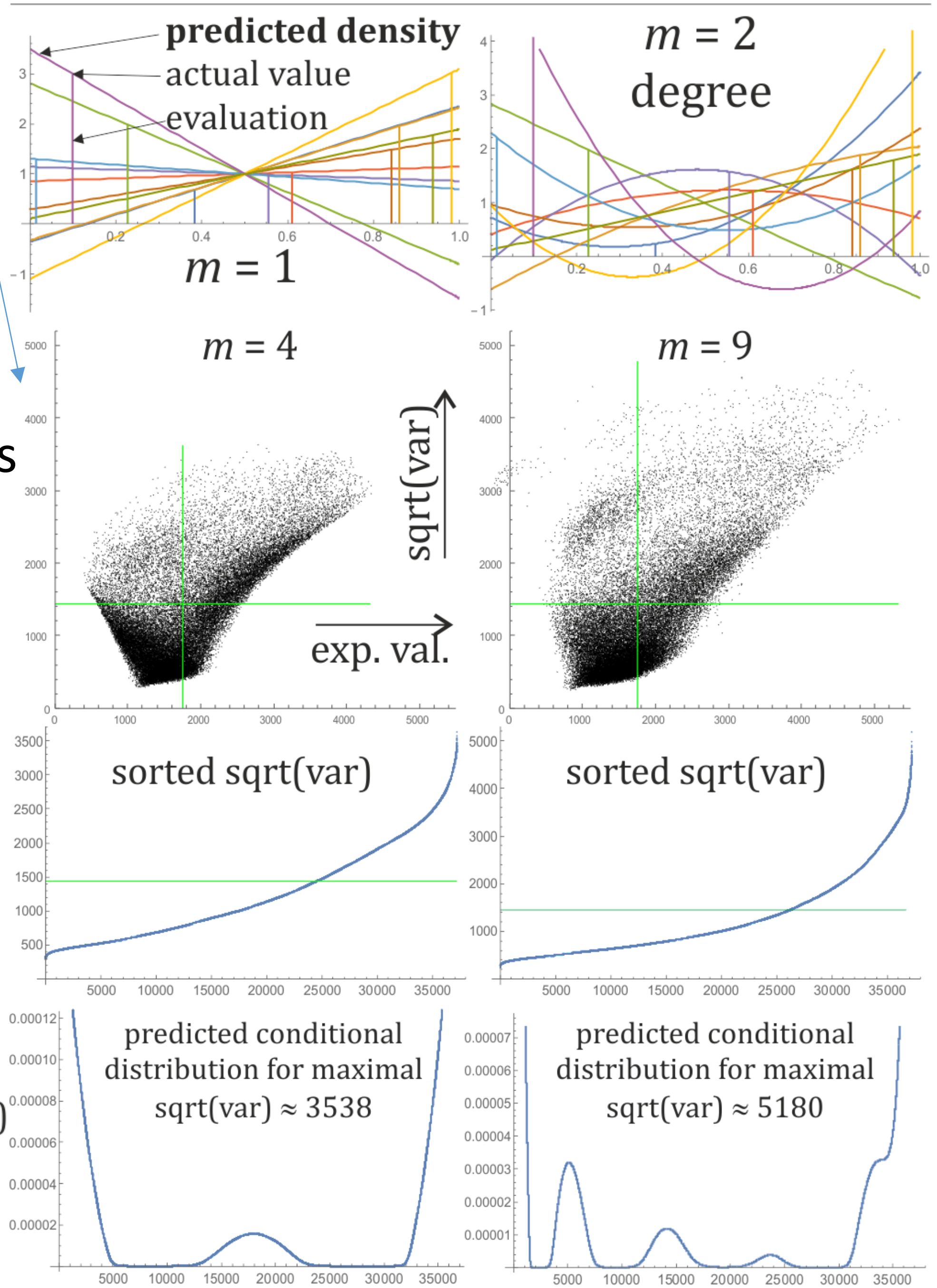
Relevance: predicting from single variable

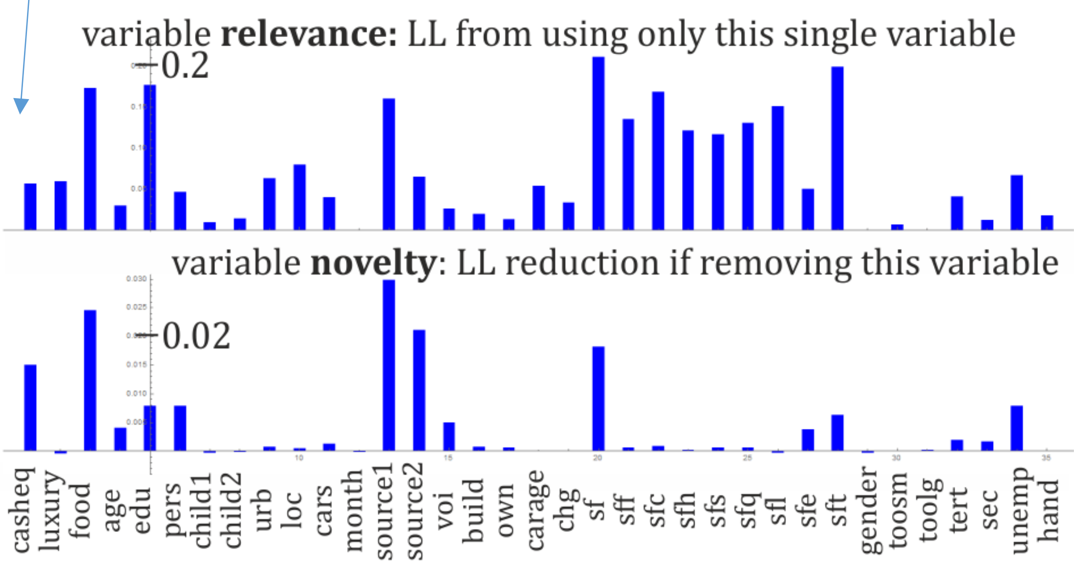
Novelty: loss if without this variable

Survey design – choosing best few variables

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

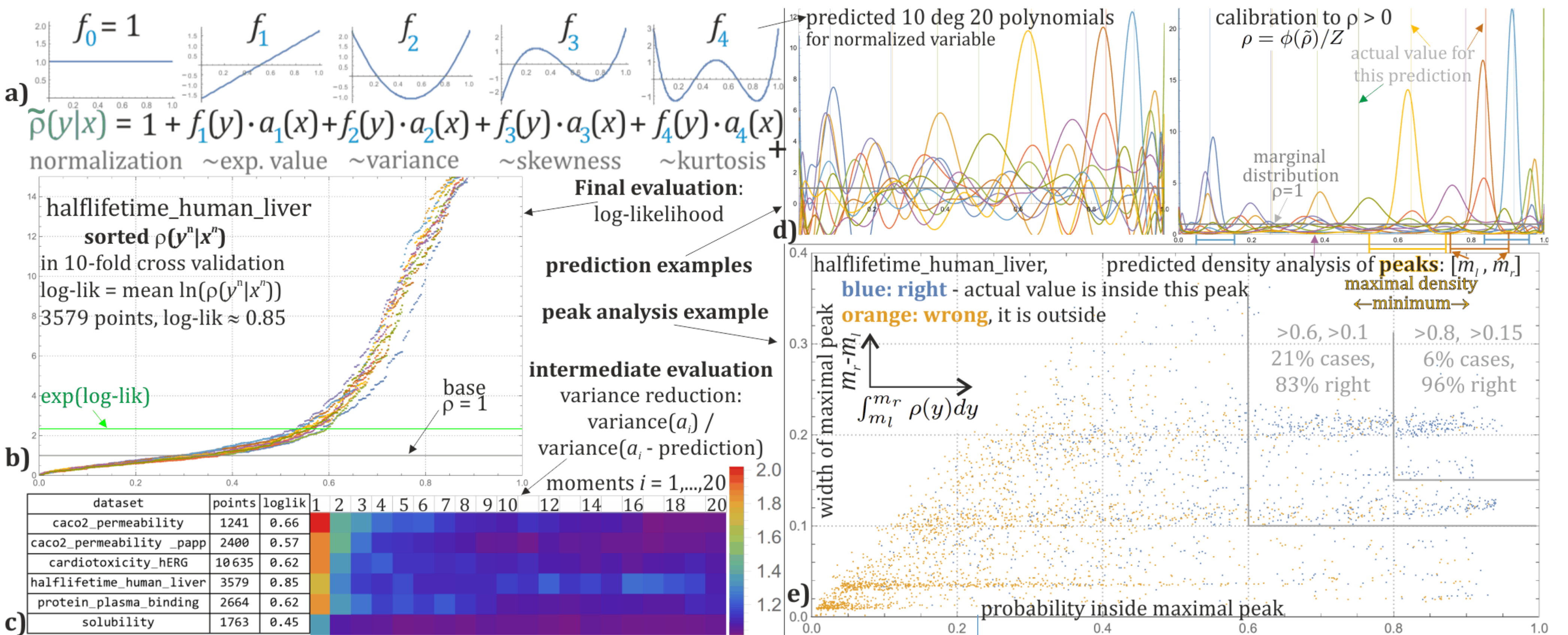
$m=$	0	1	2	3	**4**	5	6	7	8	9
	1	+exp.	+var.	+skew.	+kurt.					
LL	0	0.420	0.566	0.576	**0.580**	0.578	0.579	0.578	0.578	0.577
2^{LL}	1	1.338	1.480	1.490	**1.494**	1.493	1.494	1.493	1.493	1.492





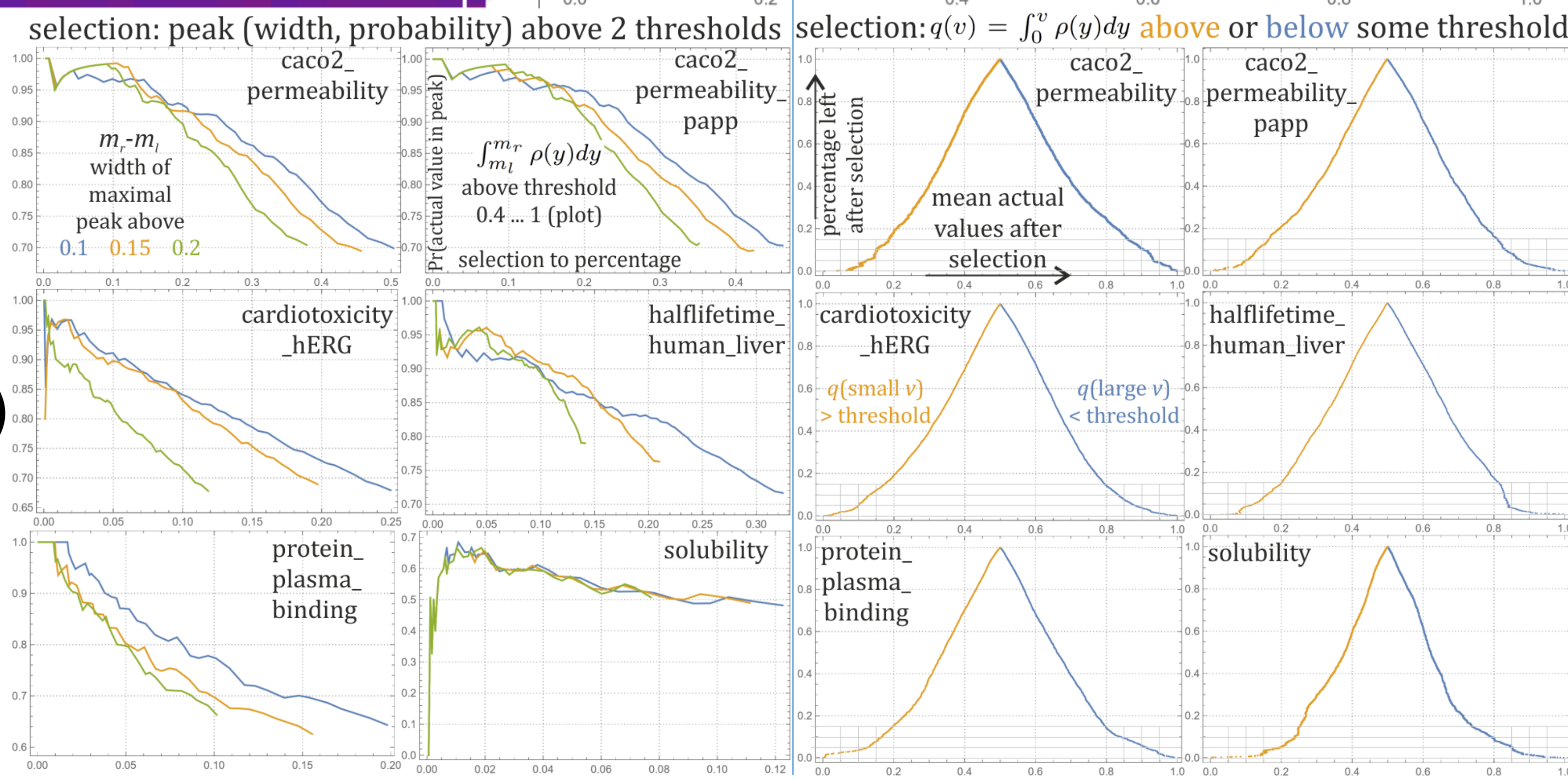
survey design: LL from chosen first few variables for m=4 (all: 0.574)

	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

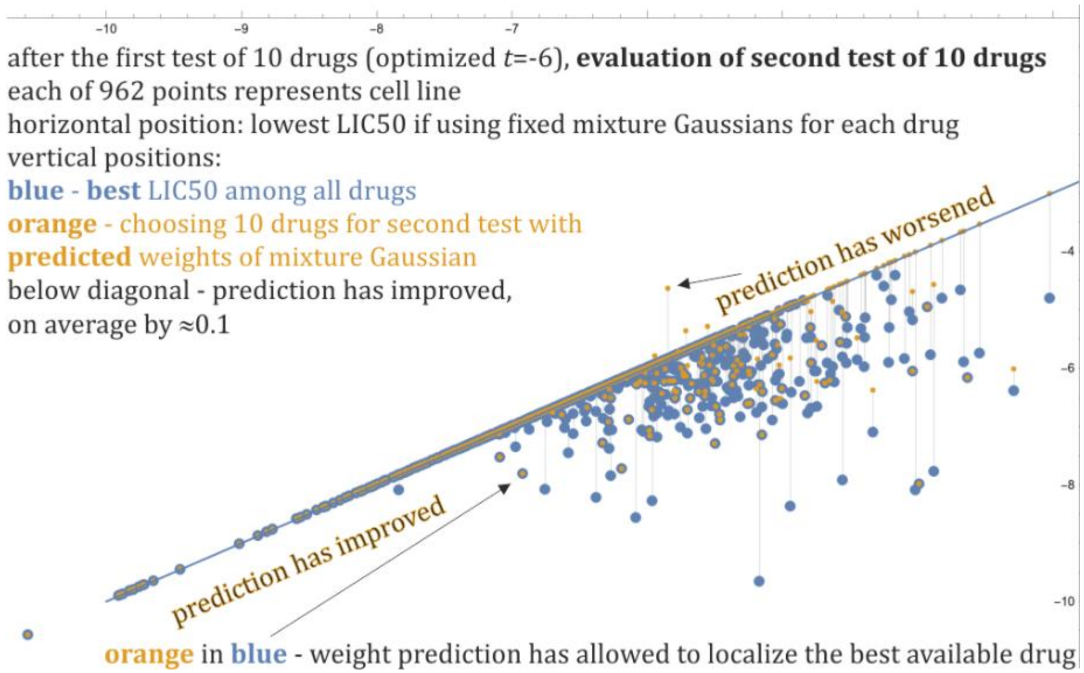
Predict probability distributions of **chemical properties for virtual screening** (**[arXiv: 2207.11174]**) **from 4860 Klek fingerprints**

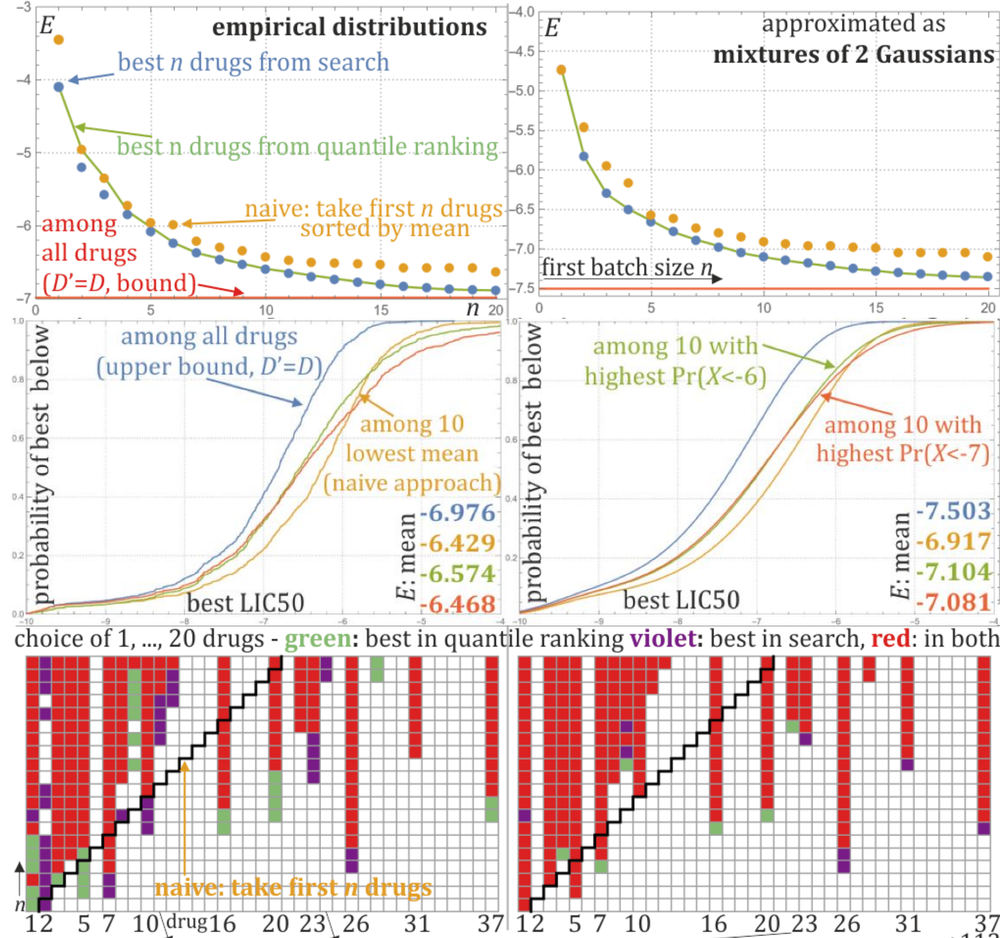


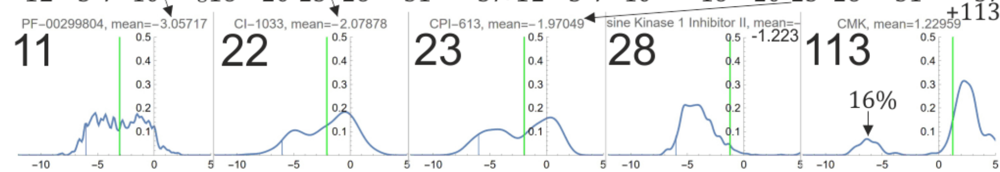
MSE value prediction: of expected value (only)

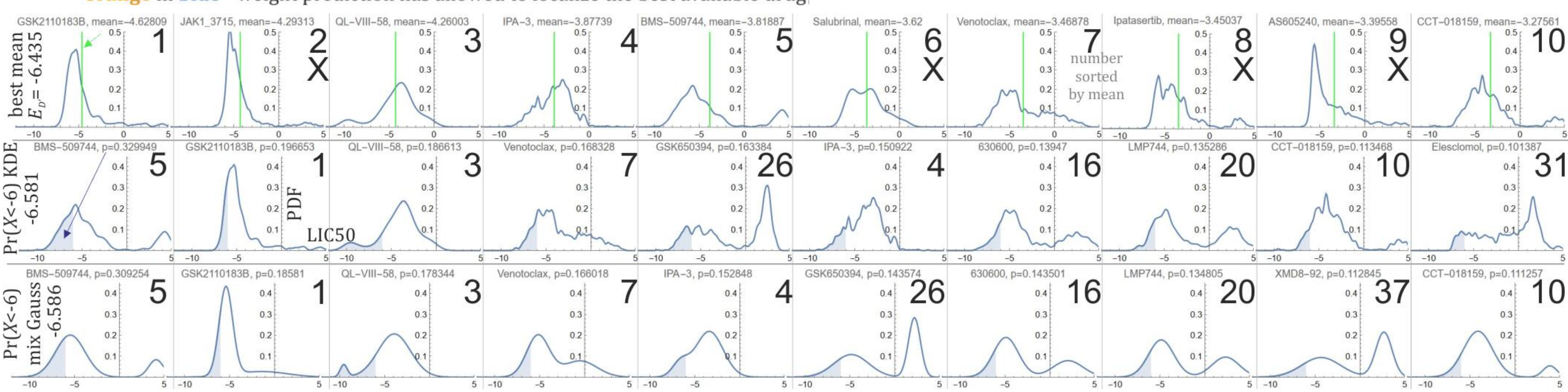
Superiority of probability density modelling, prediction for **extreme subset selection** e.g. subsets of drugs most likely containing the best one

A-B mixture: $\Pr(A|X=x) = \frac{w\,\rho_A(x)}{w\,\rho_A(x)+(1-w)\rho_B(x)}$







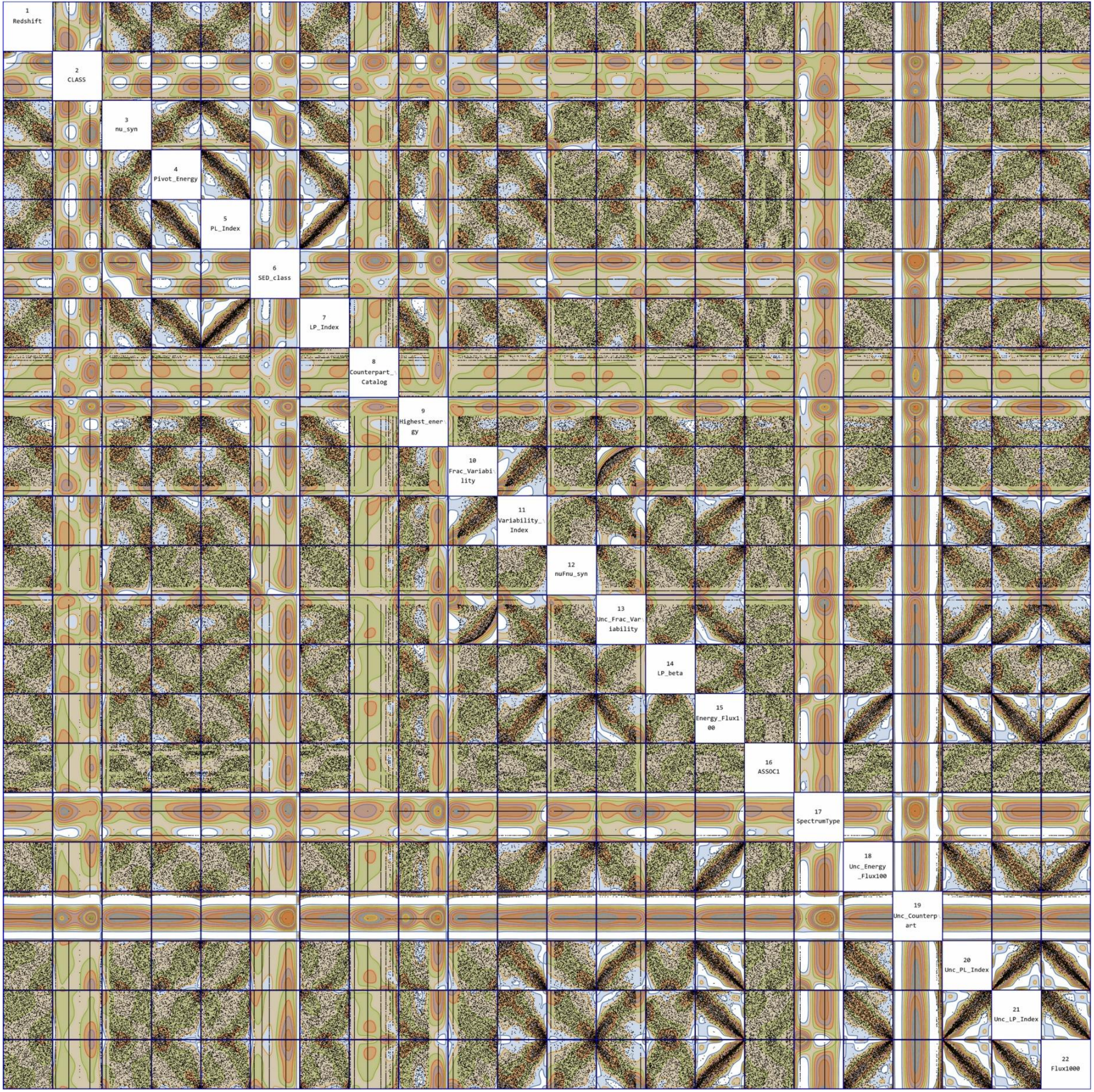


Of probability distribution of **redshift** of **Active Galactic Nuclei**

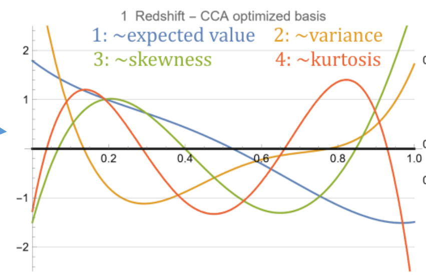
from
21 variables: discrete, continuous, combined

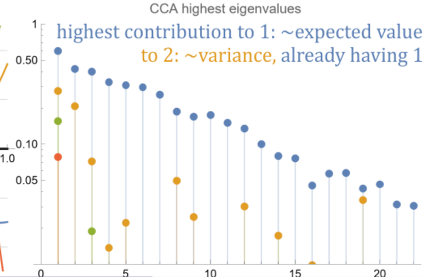
mostly describing spectrum

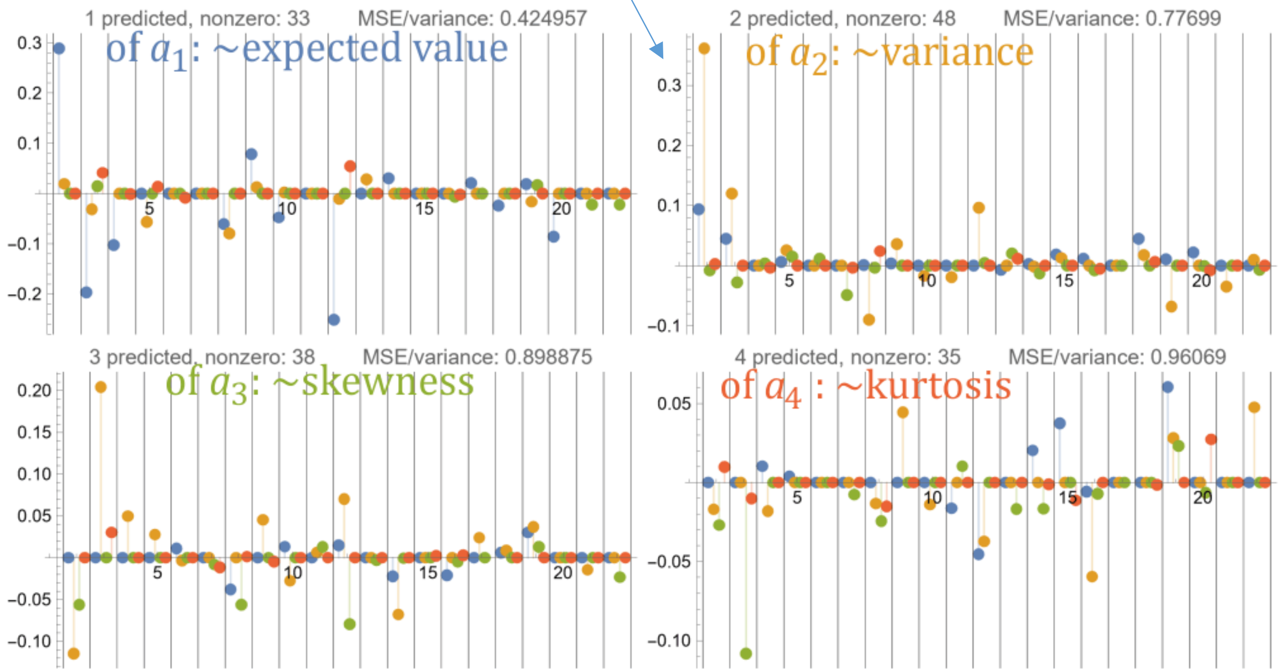
arXiv: 2206.06194

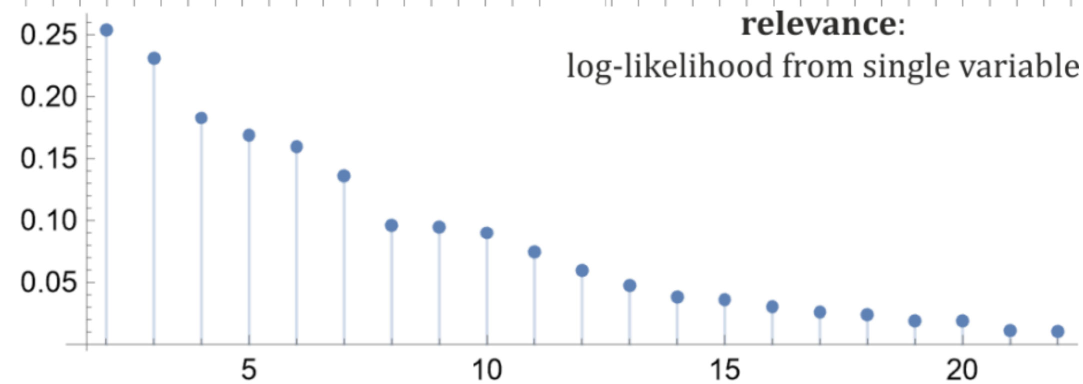


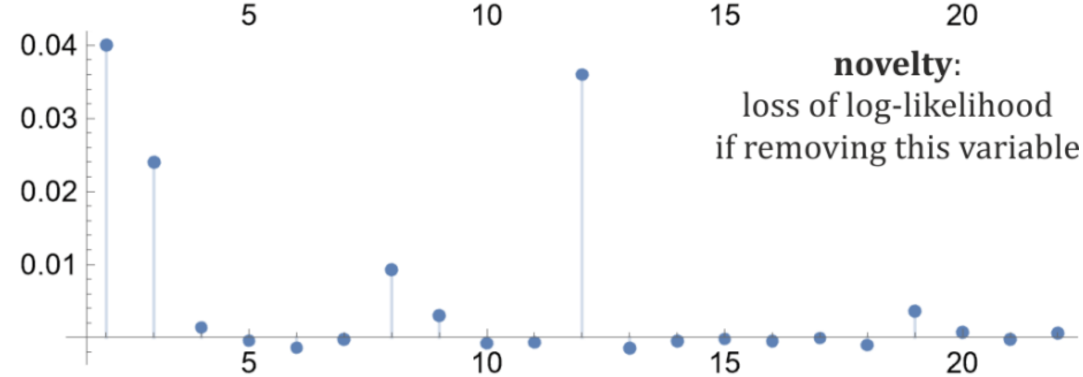
Canonical correlation analysis to optimize features for 21+1 variables, model (**l1 regular.**), var. evaluation

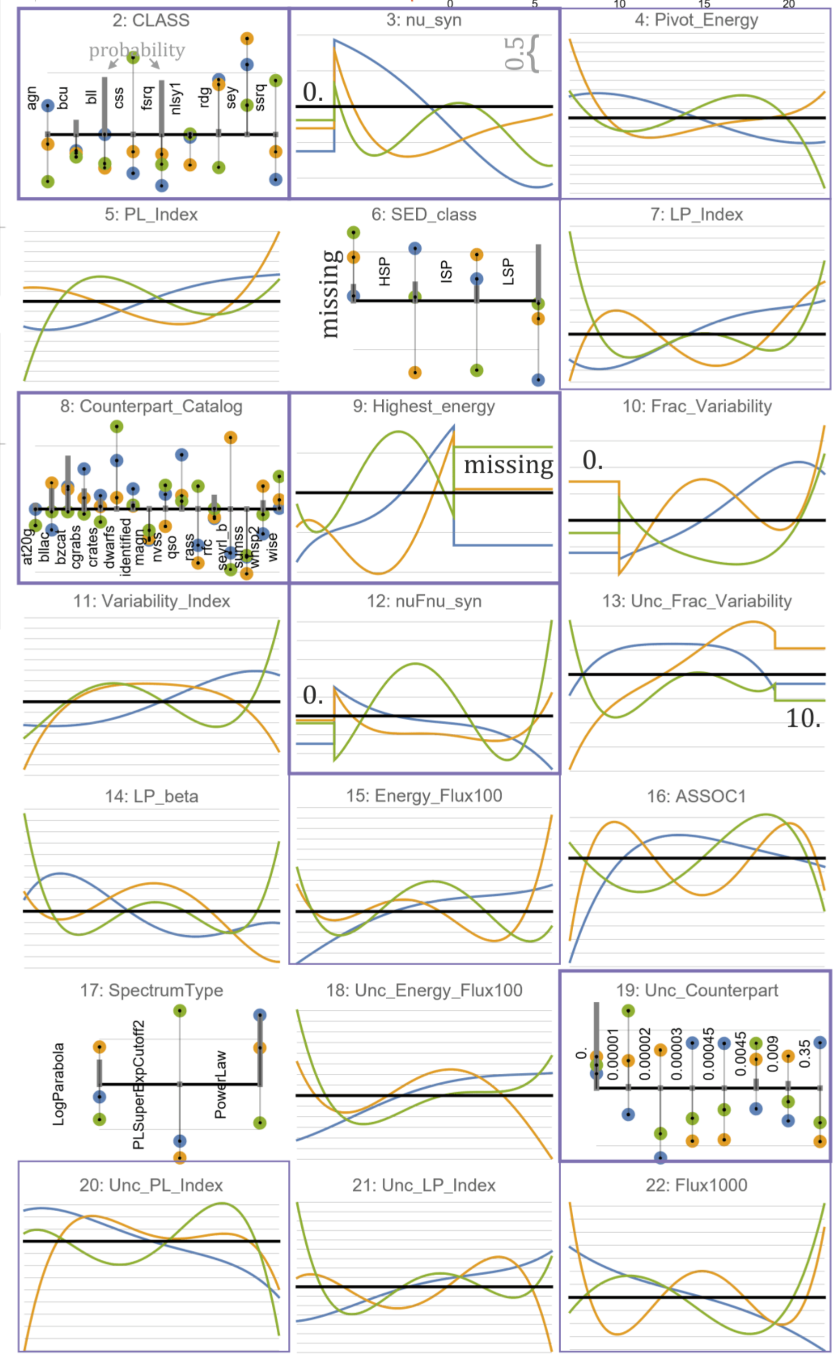












Non-stationarity analysis for [blazars]

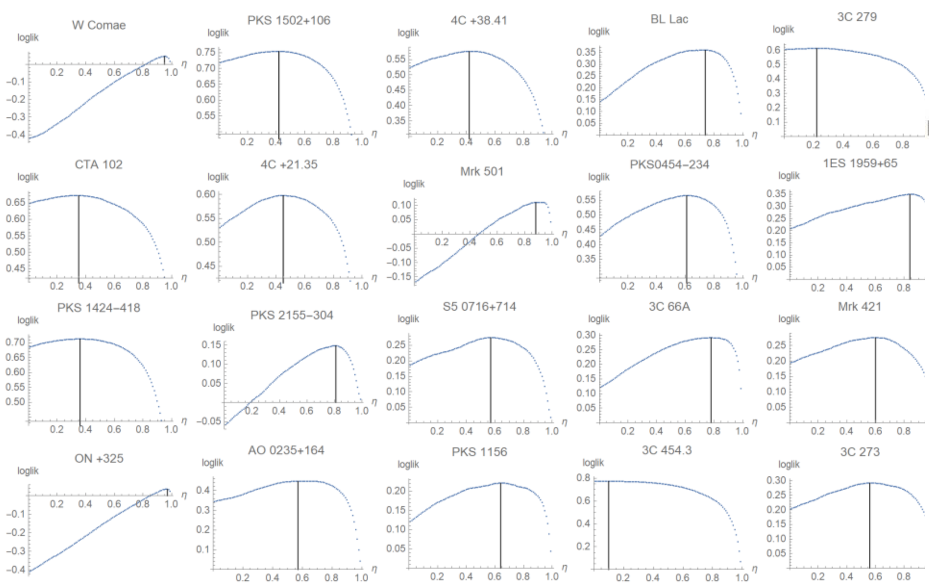
https://arxiv.org/pdf/2005.14040

Evolving density for EPD normalized

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

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

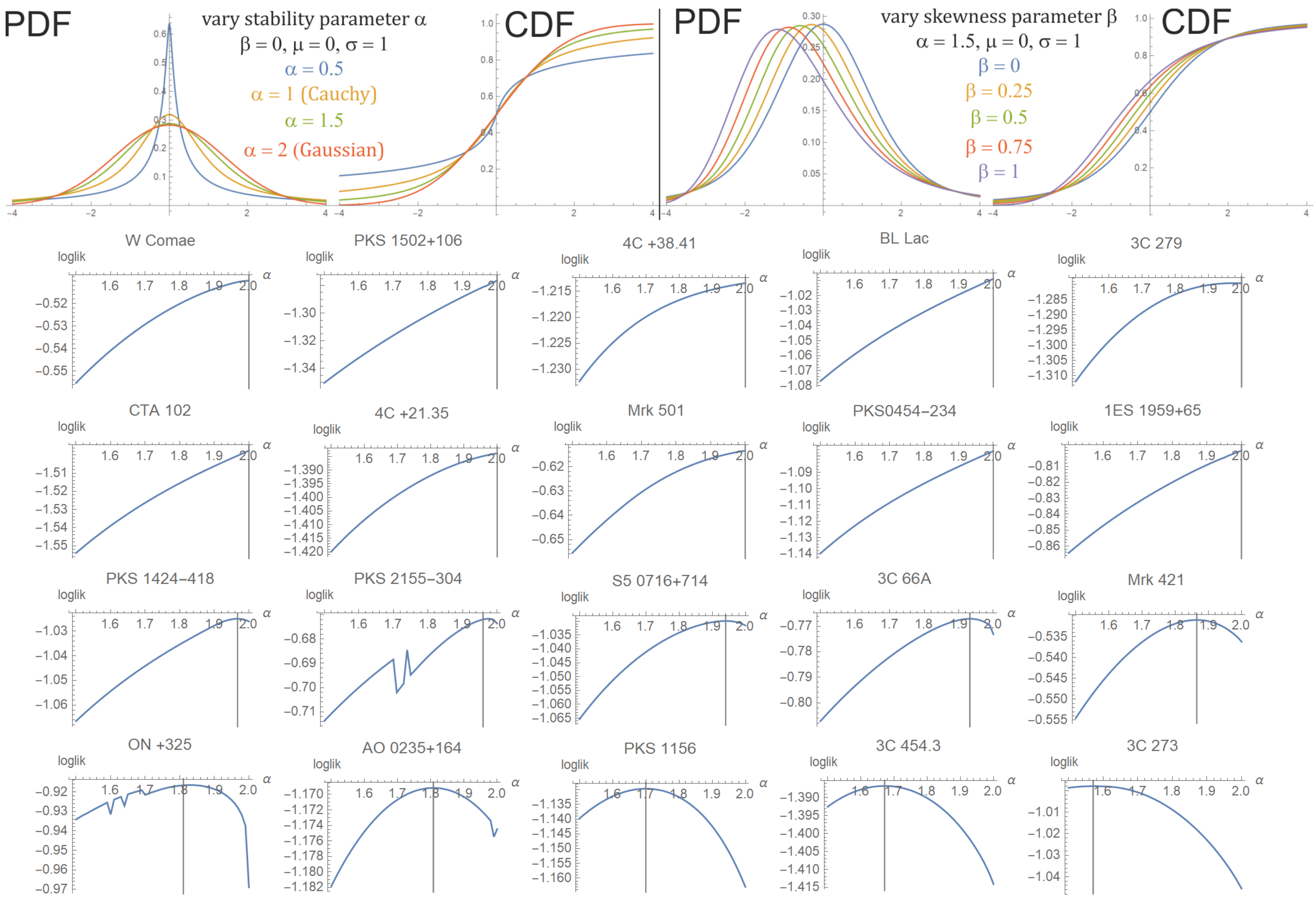
η to maximize log-likelihood:



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

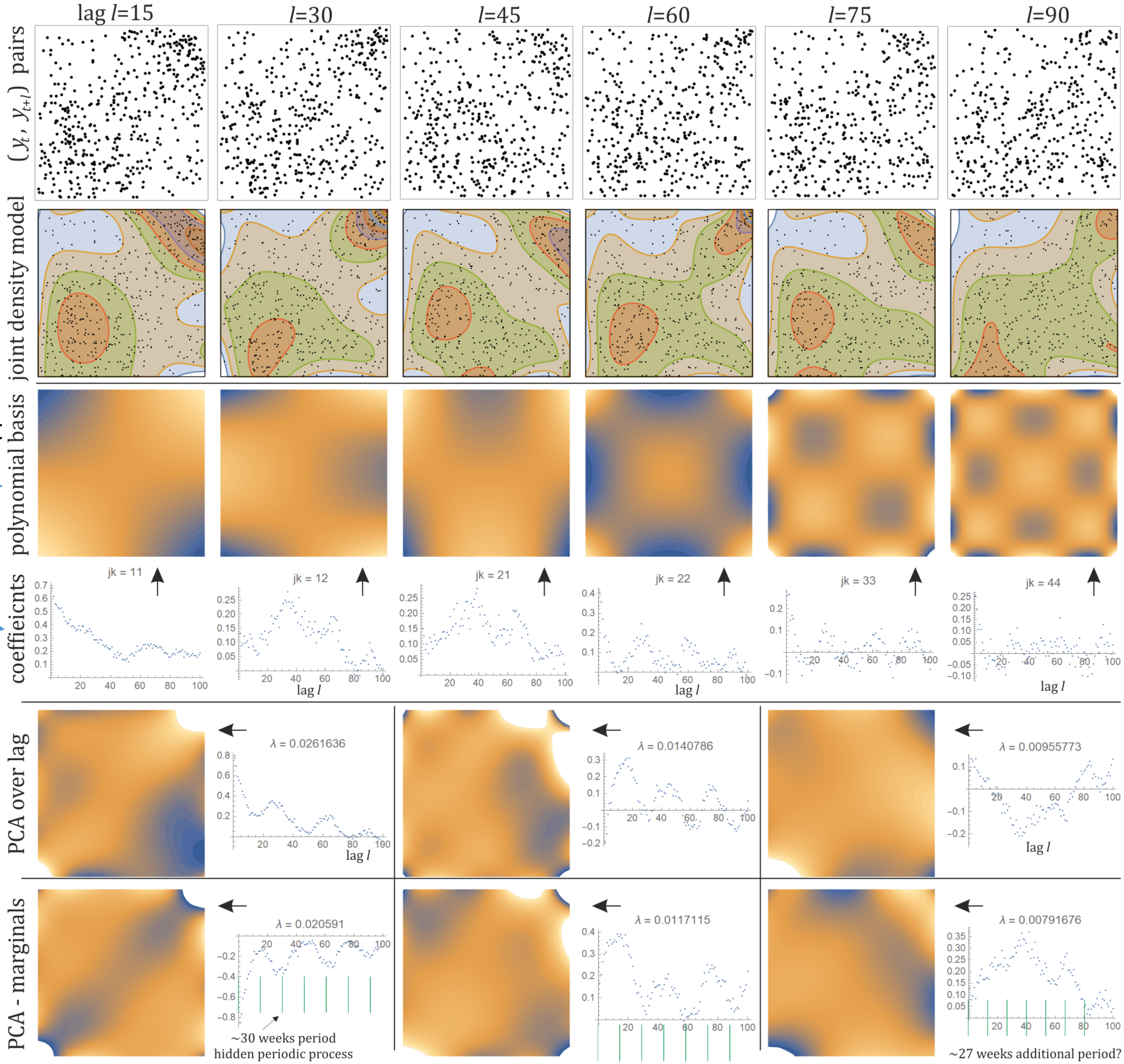
Some $a_{jk}(l)$ sequences

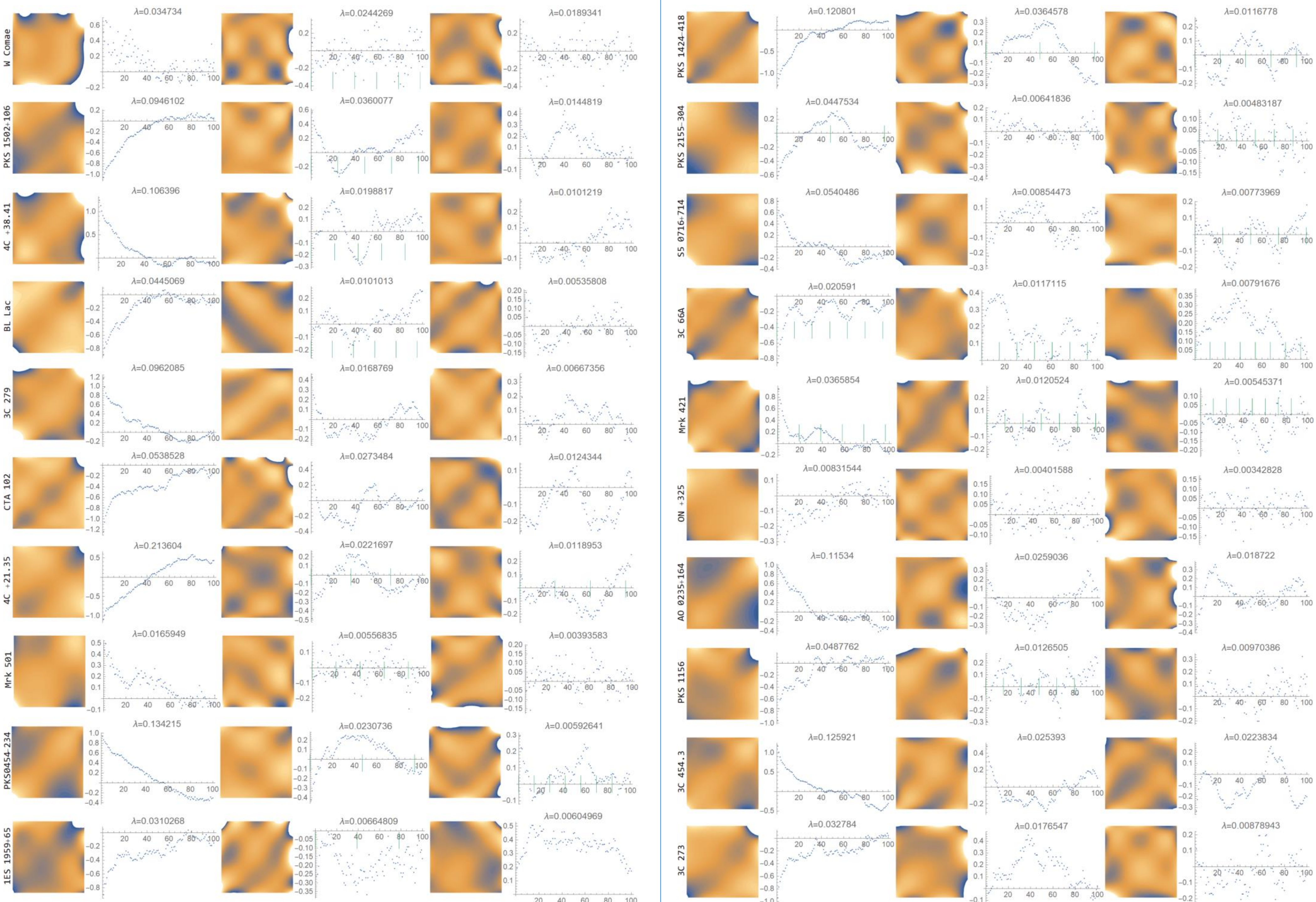
PCA feature select. to reduce basis $25 \to 3$, interpret.

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

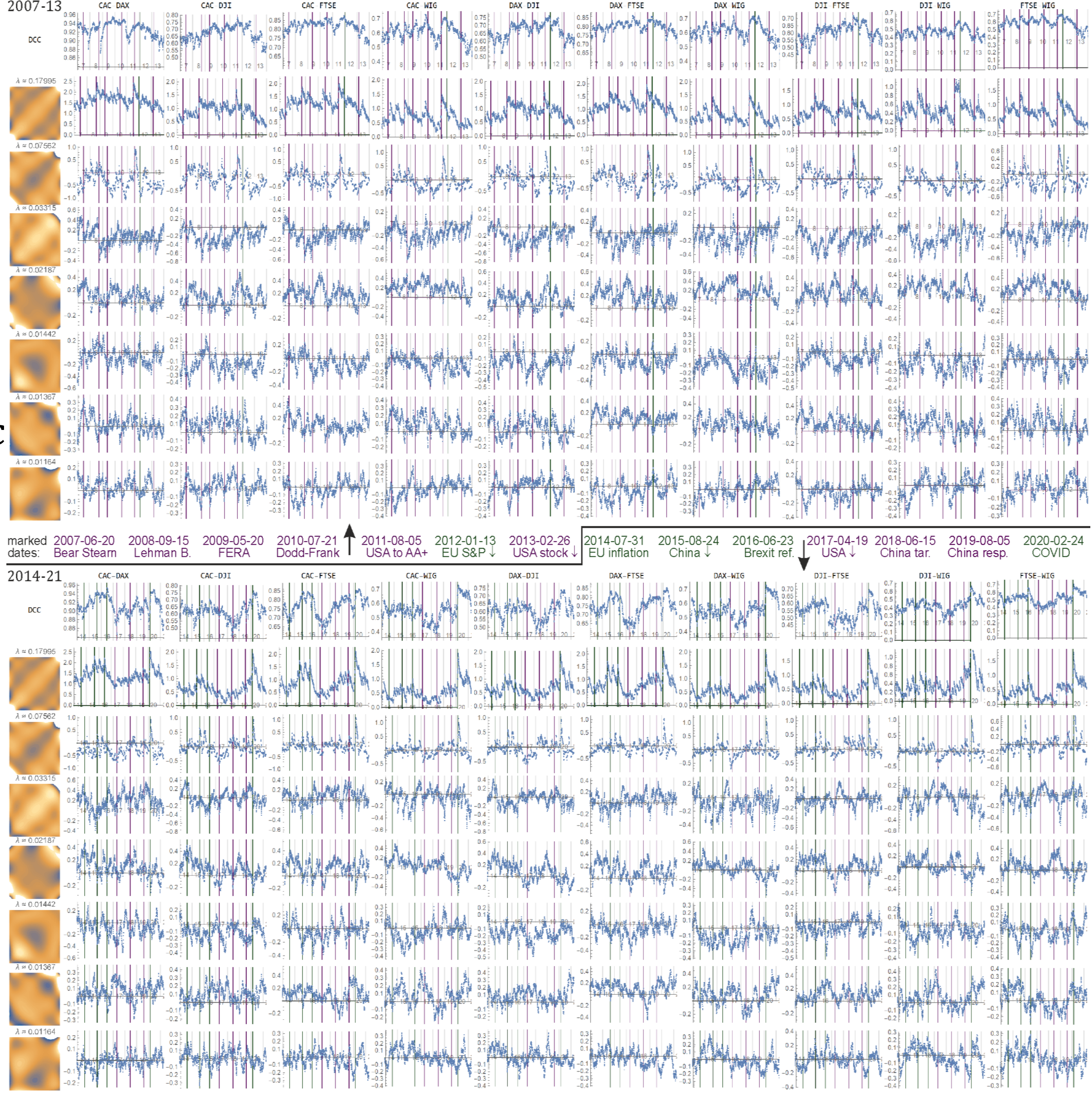
Multi-feature correlation analysis (CEJOR)

evolving in time like

[DCC](DCC) – dynamic conditional correlations

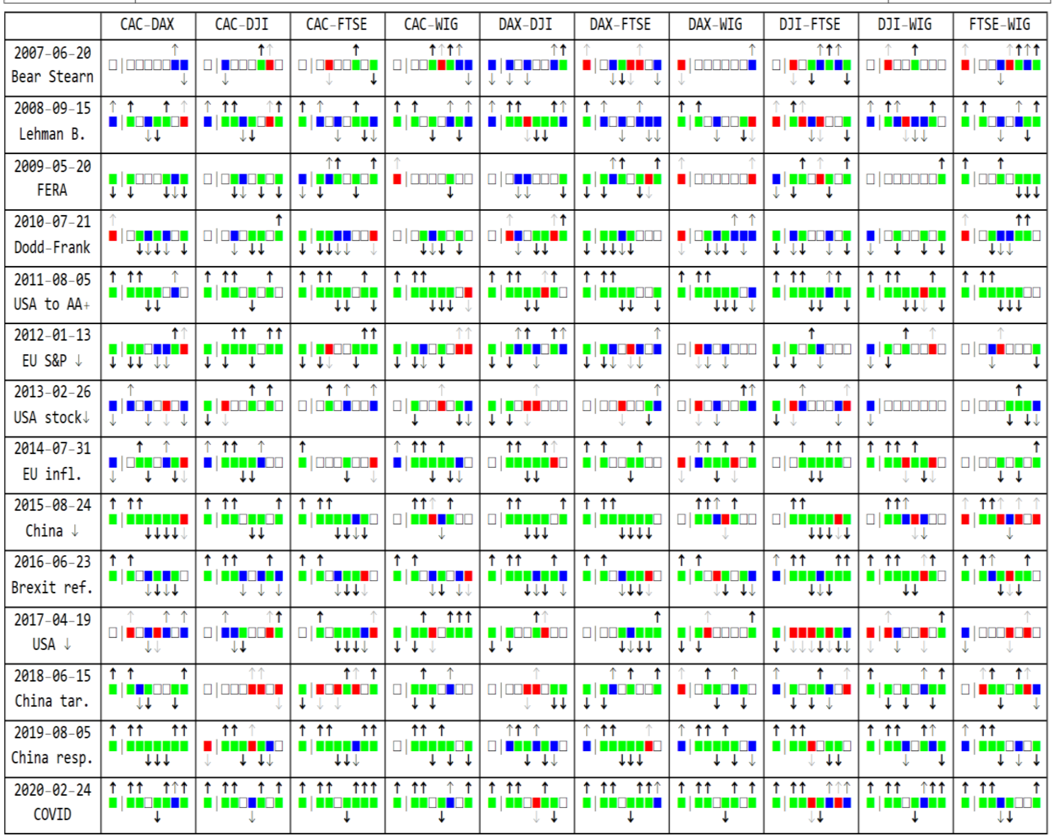
E.g. for **Contagion analysis between markets**

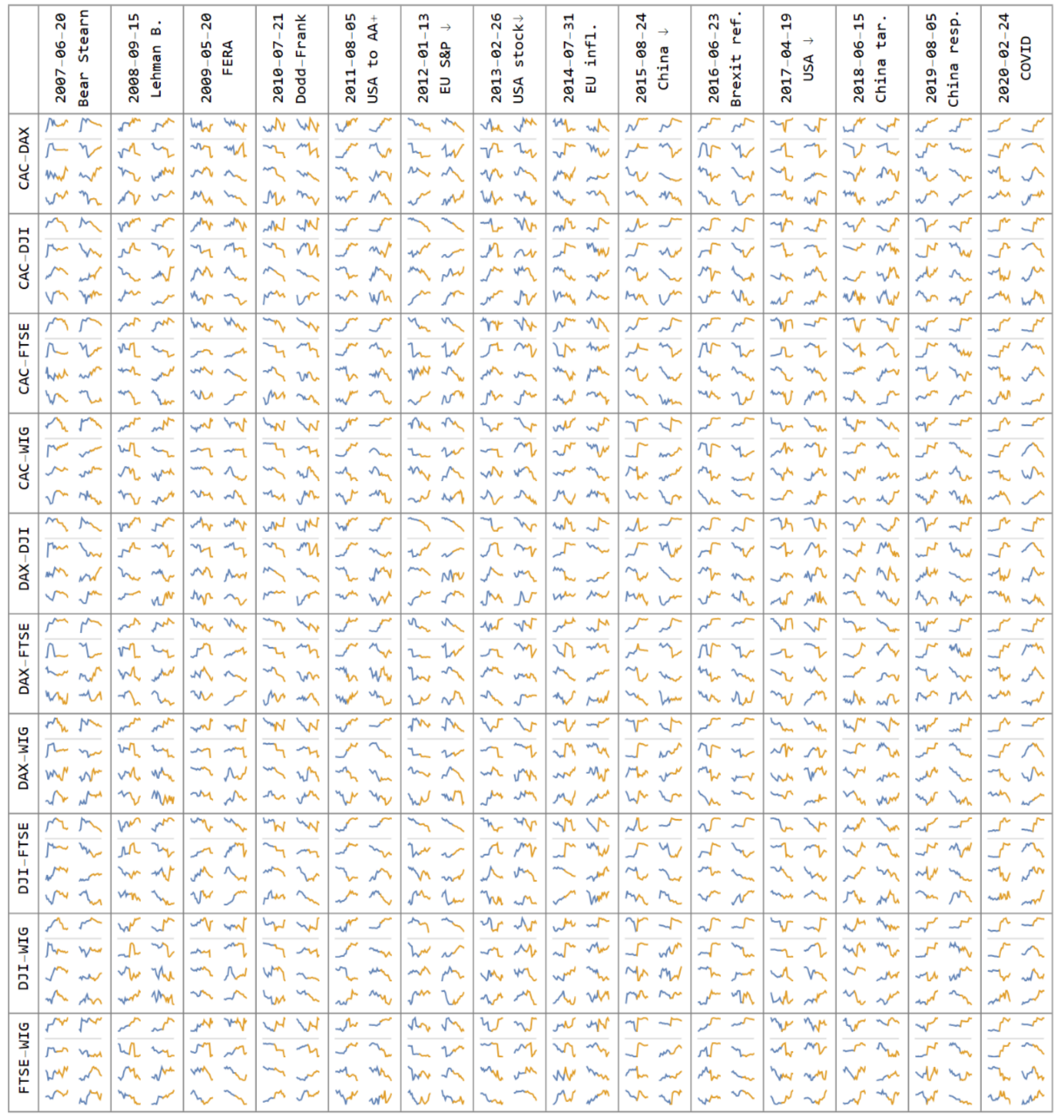
e.g. to detects crucial events



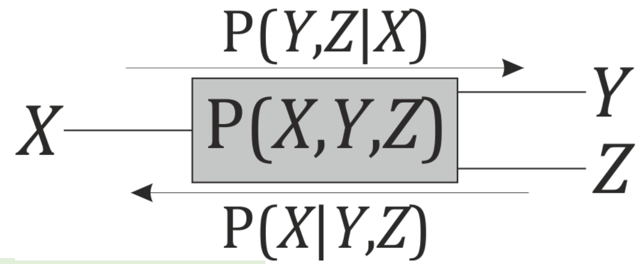
1/mean P-values of event detection:

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





Biology-inspired artificial neuron? (HCR):



Directly learns conditional probability distributions

can exploit them by predicting in flexible directions

How to do **deep learning** with them – learn **long correlation chains**?

… how to learn intermediate layers of distinguishing features?

Biological neuron: accumulate inputs until trigger, then **impulse**: sends modelled predictions of inputs.