Exploiting statistical dependencies of time series with hierarchical correlation reconstruction

Jarek Duda
Jagiellonian University, Golebia 24, 31-007 Krakow, Poland, Email: dudajar@gmail.com

Abstract—While we are usually focused on forecasting future values of time series, it is often valuable to additionally predict their entire probability distributions, e.g. to evaluate risk, Monte Carlo simulations. On example of time series of $\approx 30000$ Dow Jones Industrial Averages, there will be presented application of hierarchical correlation reconstruction for this purpose: mean-square estimating polynomial as joint density for (current value, context), where context is for example a few previous values. Then substituting the currently observed context and normalizing density to 1, we get predicted probability distribution for the current value. In contrast to standard machine learning approaches like neural networks, optimal polynomial coefficients here can be inexpensively directly calculated, have controllable accuracy, are unique and independently calculated, each has a specific cumulant-like interpretation, and such approximation using can approach complete description of any real joint distribution - providing a perfect tool to quantitatively describe and exploit statistical dependencies in time series. There is also discussed application for non-stationary time series like calculating linear time trend, or adapting coefficients to local statistical behavior.

Keywords: time series analysis, machine learning, density estimation, risk evaluation, data compression, non-stationary time series, trend analysis, wallet analysis

I. INTRODUCTION

Modeling spatial or temporal statistical dependencies between observed values is a difficult task required in a countless number of applications. Standard approaches like correlation matrix, PCA (principal component analysis) approximate this behavior with multivariate gaussian distribution. Further corrections can be extracted by approaches like GMM (gaussian mixture model), KDE (kernel density estimation) [1] or ICA (independent component analysis) [2], but they have many weaknesses like lack error control, large freedom of parameters and varying their number, or focusing on a specific types of distributions.

Fitting polynomial to observed data sample is universal approach in many fields of science, can provide as close approximation as needed. It turns out also very advantageous for density estimation, including multivariate joint distribution ([3], [4]), especially if variables are normalized to approximately uniform distribution on $[0, 1]$ with CDF of approximated distribution, to properly model tails, improve performance and standardize coefficients.

Using orthonormal basis $\rho(x) = \sum f_a(x)$, it turns out that mean-square (MSE, $L^2$) optimization leads to estimated coefficients being just averages over the observed sample: $a_f = \frac{1}{n} \sum_{i=1}^{n} f(x^i)$. For multiple variables we can use basis of products of 1D orthonormal polynomials. On example of DJIA time series it

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{Top: degree $m = 5$ polynomials (integrating to 1) on $[0,1]$ range predicting probability density based on length 5 context (previous 5 values) in 100 random positions of analyzed sequence (normalized Dow Jones Industrial Averages): joint density for $d = 1 + 5 = 6$ variables (current value and context) was MSE fitted as polynomial, then substituting the current context and normalizing to integrate to 1, we get predicted density for the current value. We can see that some predicted densities go below 0, what is an artifact of using polynomials, but can be interpreted using below evaluation/calibration curves. Predicted densities are usually close to marked $\rho = 1$ uniform density (obtained if not using context), but often localize improving prediction - for example they usually avoid extreme values beside some predictable conditions. Bottom: sorted densities for the actual current values in all 29349 situations: in $\approx 20\%$ cases it gives worse prediction than $\rho = 1$ (without using context), but in the remaining cases it is essentially better. The number of coefficients in the used basis is $|B| = (m+1)^d$. We can see that prediction generally improves (higher density with growing number of coefficients, however, beside growing computational cost, it comes with overfitting (e.g. negative density) - polynomial approaches sum of Dirac deltas in points of the sample.

1 Source of DJIA time series: http://www.idvbook.com/teaching-aid/data-sets/the-dow-jones-industrial-average-data-set/}
will be used for prediction of current probability distribution based on a few previous values.

Finally we get asymptotically complete description of statistical dependencies - approaching any real joint distribution of observed variables. Coefficients can be cheaply calculated as just averages, are unique and independently calculated, for stationary time series we can control their accuracy. Each has also a specific interpretation: resembling cumulants, but being much more convenient for reconstructing probability distribution - instead of the difficult problem of moments [5], here they are just coefficients of polynomial. However, disadvantage of using polynomial as density parametrization is that it occasionally leads to negative densities, what can be interpreted as low positive - plot of sorted predicted densities of actually observed values allows for such calibration.

In the discussed here example: analysis of DJIA time series, we will first normalize the variables to nearly uniform probability distribution on \([0, 1]\): by considering differences of logarithms, and transforming them by CDF (cumulative distribution function) of approximated distribution (Laplace) as shown in Fig. 2.

Then looking at \(d\) successive positions of such normalized variable, if uncorrelated they would come from \(\rho \approx 1\) distribution on \([0, 1]^d\). Its corrections as linear combination of orthonormal basis of polynomials can be inexpensively and independently calculated, providing unique and asymptotically complete description of statistical dependencies between these neighboring values. Treating \(d - 1\) of them as earlier context, substituting their values and normalizing to 1, we get predictions of probability distribution for the current value as summarized in Fig. 3.

There will be also proposed handling of non-stationary time series: by replacing \(a_f = \frac{1}{n} \sum_{i=1}^{n} f(x^i)\) global average with local averages over past values with exponentially decaying weights, or using interpolation treating time as additional dimension.

Presented approach can be naturally extended to multivariate time series, e.g. stock prices of separate companies to model their statistical dependencies, what is presented in [6] on example of yield curve parameters, here there will be presented example of modelling various statistical dependencies of 29 of Dow Jones companies.

II. NORMALIZATION TO NEARLY UNIFORM DENSITY

We will discuss on example of Dow Jones Industrial Averages time series \(\{v^t\}_{t=1..n_0}\) for \(n_0 = 29355\). As financial data usually evolve in multiplicative not additive manner, we will work with \(\ln(v^t)\) to make it additive.

Time series are usually normalized to allow assumption of stationary process: such that joint probability distribution does not change when position is shifted. The standard approach, especially for gaussian distribution, is to subtract mean value, then divide by the standard deviation.

However, above normalization does not exploit local dependencies between values, what we are interested in. Using experience from data compression (especially lossless image e.g. JPEG LS [7]), we can use a predictor for the next value based on its local context: for example a few previous values (2D neighbors for image compression), or some more complex features (e.g. using averages over time...
windows, or dimensionality reduction methods like PCA), then model probability distribution of difference from the predicted value (residue).

Considering simple linear predictors: \( y^t \approx \sum_{i=0}^{k} b_i y^{t-i} \) like in ARIMA-like models, we can use optimize \( \{b_j\} \) parameters to minimize mean square error. For 2D image such optimization leads to approximate parameters \( v_{x,y} \approx 0.8 v_{x-1,y} - 0.3 v_{x-1,y-1} + 0.2 v_{x,y-1} + 0.3 v_{x+1,y-1} \). For Dow Jones sequence such optimization leads to nearly negligible weights for all but the previous value. Hence, for simplicity we will just operate on

\[
y^t = \ln(y^{t+1}) - \ln(y^t)
\]

time series, where the number of possible indexes has been reduced by 1 due to shift: \( n_1 = n_0 - 1 \).

Such sequences of differences from predictions (residues) are well known in data compression to have nearly Laplace distribution - density:

\[
g(y) = \frac{1}{2b} \exp\left(-\frac{|y - \mu|}{b}\right)
\]

where maximum likelihood estimation of parameters is just: \( \mu = \text{median of } y, b = \text{mean of } |y - \mu| \). We can see in Fig. 2 that CDF from sorted \( y_t \) values has decent agreement with CDF of Laplace distribution. Otherwise, there can be used e.g. generalized normal distribution, also called exponential power distribution or generalized error distributions, which includes both Gaussian and Laplace distribution. Stable distributions (Lévy) might be also worth considering as they include heavy tail distributions. These distributions have also known asymmetric variants - which should be considered if two directions have essentially different tails.

For simplicity we use Laplace distribution here to normalize our variables to nearly uniform in \((0, 1]\), what allows to compactify the tails, improve performance and normalize further coefficients:

\[
x_t = G(y_t) \quad \text{where} \quad G(y) = \int_{-\infty}^{y} g(y') dy'
\]

is CDF of used distribution (Laplace here). We can see in Fig. 2 that this final \( x_t \) sequence has nearly uniform probability distribution. Its corrections will be included in further estimation of polynomial as (joint) probability distribution, like presented later in Fig. 4.

We will search for \( \rho_X(x) \) density. To remove transformation \( (f, g) = \int f g \, dx \): \( j = 0 \) coefficient guards normalization, the remaining functions integrate to 0, and their coefficients describe perturbation from uniform distribution. These coefficients have similar interpretation as cumulants, but are more convenient for density reconstruction. Center: 2D product basis for \( j \in \{0, 1, 2\} \). The \( j = 0 \) coordinates do not change with corresponding perturbation. Bottom: sorted calculated coefficients (without \( \alpha_{000000} = 1 \)) for DJIA sequence, \( m = 5 \) and length 5 context \((d = 6)\) modelling. Assuming stationarity, for uniform distribution their standard deviation would be \( \sigma \approx 1/\sqrt{n} \approx 0.006 \), exceeded here more than tenfold by many coefficients - allowing to conclude that they are essential: not just a noise.

![Sorted Dow Jones coefficients](image)

**III. HIERARCHICAL CORRELATION RECONSTRUCTION**

After normalization we have \( \{x_t\} \) sequence with nearly uniform density, marked green in Fig. 2 here. Taking its \( d \) succeeding values, if uncorrelated they would come from nearly uniform distribution on \([0, 1]^d\) - difference from uniform distribution describes statistical dependencies in our time series. We will use polynomial to describe them: estimate joint density for \( d \) succeeding values of \( x \).

Define \( x_t^i = x^{t+i-1} \) for \( i = 1, \ldots, d \) and \( t = 1, \ldots, n \), \( n = n_1 - d + 1 \). They form \( x^t = \{x^t_i\}_{i=1..d} \in [0, 1]^d \) vectors containing value with its context - we will model probability density of these vectors. Generally we can also use more sophisticated contexts, for example average of a few earlier values (e.g. \( (x_{t-5} + x_{t-6})/2 \)) as a single context value to include correlations of longer range. Normalization to nearly uniform density is recommended for the predicted values \( x_t^i \), for context values it might be better to omit it, especially when absolute values are important like for image compression.

Finally assume we have \( \{x^t\}_{t=1...n} \subset [0, 1]^d \) vector sequence of value with its context, we would like to model density of such vectors as polynomial. It turns out that using orthonormal basis, which for multidimensional case can be products of 1D orthonormal polynomials, mean square \((L^2)\) optimization leads to extremely simple formula...
for estimated coefficients:
\[ \rho(x) = \sum_{j \in \{0, \ldots, m\}^d} a_j f_j(x) = \sum_{j_1 \ldots j_d=0}^{m} a_j f_{j_1}(x_1) \cdots f_{j_d}(x_d) \]

with estimated coefficients: \( a_j = \frac{1}{n} \sum_{i=1}^{n} f_j(x_i) \) (4)

The basis used this way has \(|B| = (m + 1)^d\) functions, generally it seems worth to consider different \(m_i\) for separate coordinates \((|B| = \prod_{i=1}^{d}(m_i + 1))\). Beside inexpensive calculation, this simple approach has also very convenient property of coefficients being independently calculated, giving each \(j\) unique value and interpretation, and controllable error. Independence also allows for flexibility of considered basis - instead of using all \(k\), we can focus on more promising ones: with larger absolute value of coefficient, replacing negligible \(a_j\). Instead of mean square optimization, we can use often preferred: likelihood maximization [4], but it requires additional iterative optimization and introduces dependencies between coefficients.

Above \(f_j\) 1D polynomials are orthonormal in \([0, 1]\):
\[ f_j^1 f_j(x) f_k(x) \, dx = \delta_{jk} \], getting (rescaled Legendre): \( f_0 = 1 \) and for \(j = 1, 2, 3, 4, 5\) correspondingly:
\[ \sqrt{3}(2x-1), \sqrt{5}(6x^2-6x+1), \sqrt{7}(20x^3-30x^2+12x-1), \]
\[ 3(70x^4-140x^3+90x^2-20x+1), \]
\[ \sqrt{11}(252x^5-630x^4+560x^3-210x^2+30x-1). \]

Their plots are in top of Fig. [3] \( f_0 \) corresponds to normalization. The \(j = 1\) coefficient decides about reducing or increasing the mean - have similar interpretation as expected value. Analogously \(j = 2\) coefficient decides about focusing or spreading given variable, similarly as variance. And so on: further \(a_j\) have similar interpretation as cumulants, however, while reconstructing density from moments is a difficult problem, presented description is directly coefficients of polynomial estimating the density.

For multiple variables, \(a_j\) describes only correlations between \(C = \{i : j_i > 0\}\) coordinates, does not affect \(j_i = 0\) coordinates, as we can see in the center of Fig. [3] Each coefficient has also a specific interpretations here, for example \(a_{11}\) decides between increase and decrease of second variable with increase of the first, \(a_{12}\) analogously decides focus or spread of the second variable.

Assuming stationary time series (fixed joint distribution in \([0, 1]^d\), errors of such estimated coefficients come from approximately gaussian distribution:
\[ \tilde{a} - a \sim \mathcal{N}\left(0, \frac{1}{\sqrt{n}} \int (f_j - a_j)^2 \rho \, dx \right) \] (5)

For \(\rho = 1\) the integral has value 1, getting \(\sigma = 1/\sqrt{n} \approx 0.006\) in our case. As we can see in bottom of Fig. [3] a few percents of coefficients here are more that tenfold larger: can be considered as essential, not a result of noise.

Here is a list of the largest \(|a_j| > 0.14\) coefficients for Dow Jones normalized series (beside \(a_{000000} = 1\) in \(d = 6, m = 5\) case. It neglects shifted sequences, for example \(a_{020020} \approx a_{020020} \approx a_{002002}.\)

Positive:
\[ a_{020020} \approx 0.184867 \quad a_{020002} \approx 0.183297 \]
\[ a_{200200} \approx 0.178384 \quad a_{202000} \approx 0.177606 \]
\[ a_{555455} \approx 0.176333 \quad a_{220000} \approx 0.176184 \]
\[ a_{555435} \approx 0.169778 \quad a_{554535} \approx 0.161684 \]
\[ a_{545445} \approx 0.156764 \quad a_{555555} \approx 0.149727 \]
\[ a_{555555} \approx 0.147934 \quad a_{454523} \approx 0.145962 \]

Negative:
\[ a_{555552} \approx -0.170723 \quad a_{344444} \approx -0.166773 \]
\[ a_{455235} \approx -0.156860 \quad a_{342554} \approx -0.149314 \]
\[ a_{452555} \approx -0.147201 \quad a_{555451} \approx -0.146523 \]
\[ a_{555532} \approx -0.145356 \quad a_{555345} \approx -0.143087 \]
\[ a_{555552} \approx -0.142076 \quad a_{554545} \approx -0.140343 \]

Each such unique coefficient describes a specific correction from uniform density: by \(a_j f_j(x_1) \cdots f_{j_d}(x_d)\). For example we can see large positive coefficients for all pairs of \(j = 2\), what means upward directed parabola for both variables: quantitatively describes how large change in a given day increases probability of large changes in neighboring days. Further coefficients have more complex interpretations, for example large positive \(a_{555555}\) means that 6 large increases in a row are preferred, but 6 large decreases are less likely. In contrast, large negative \(a_{555552}\) means that larger change 5 days earlier reduces probability of 5 large increases in a row.

Having such density we can use it to predict probability distribution of the current symbol basing on the context (Fig. [1]): by substituting context to the polynomial and normalizing the remaining 1D polynomial to integrate to 1.

![Figure 4. Modelling probability distribution as independent variable (d = 1) using degree m polynomials: ρ(x) = ∑_{j=0}^{m} a_j f_j(x). After normalization with CDF of Laplace distribution, we should have ρ ≈ 1. Here we repair its inaccuracy with estimated polynomial (left column), corresponding to the plot in the bottom of Fig. 3-right column contains differences between empirical CDF and such fitted polynomial. Obviously this difference reduces with degree m, however, we can see that it contains a growing number (≈ m) of oscillations (Runge’s phenomenon).](image-url)
Unfortunately such density can sometimes go below zero, what needs a separate interpretation as low positive.

IV. ADAPTIVITY FOR NON-STATIONARY TIME SERIES

We have previously assumed stationary time series: that joint probability distribution within length $d$ moving time windows is fixed, what is often only approximation for real time series. As coefficients here are just averages over values: $a_f = \frac{1}{n} \sum x f(x)$, for coefficients describing local behavior we can use (known in data compression) averaging with exponentially decaying weights [4]:

$$a_{f+1}^t = \lambda a_f^t + (1 - \lambda) f(x^t)$$

for $m = 4$ with two different learning rates $\lambda = 0.9997$ or $0.999$.

The blue curve for $\lambda = 0.9997$ is smoother, however, it is at cost of delay (shifted right) - needs more time to adapt to new behavior.
for some learning rate $\lambda$: close but smaller than 1 (e.g. $\lambda = 0.999$), starting for example with $a_f(0) = 0$. Its proper choice is a difficult question: larger $\lambda$ gives smoother behavior, but needs more time to adapt (delay).

For modeling of time trends or a posteriori analysis of historical data (with known future), we can alternatively estimate polynomial for multi-dimensional variable with time as one of coordinates, rescaled to $[0,1]$ range, e.g. $(t/n, x^t)$. This way we estimate behavior of each coefficient as polynomial, allowing e.g. to interpolate to real time, or try to forecast that future trend (as low degree polynomial) will be similar as in the earlier period. It might be tempting to use this approach also for extrapolation to predict future trends, e.g. rescale time to $[0,1-\epsilon]$ range instead, and look at behavior in time 1. However, such polynomial often has some uncontrollable behavior at the boundaries, suggesting caution while such extrapolation.

Other orthonormal families (e.g. sines and cosines) have better boundary behavior - might be more appropriate for such extrapolation, however, discussed earlier modelling of joint distribution with context representing the past is generally a safer approach.

The last 3 figures present such analysis for discussed DJIA sequence. Figure 4 contains estimation of density as polynomial using stationarity assumption (inaccuracy of Laplace used in normalization). Figure 5 contains its time evolution for non-stationary models: adaptive or interpolation. Figure 6 evaluates these approaches and shows time evolution for first 4 cumulant-like coefficients.

V. MODELLING MULTIVARIATE DEPENDENCIES

Example of predicting probability distribution for multivariate time series is presented in [6]. To continue the Dow Jones topic, there will be presented application of the discussed methodology to better understand complex statistical dependencies between stock prizes of 30 companies it is calculated from, for example for more accurate wallet analysis. The selection of these companies has changed throughout the history, but for the last decade there can be downloaded daily prices from NASDAQ webpage (www.nasdaq.com) for all but DowDuPont (DWDP) - there will be used daily close values for 2008-08-14 to 2018-08-14 period ($n_0 = 2518$ values) for the remaining 29 companies: 3M (MMM), American Express (AXP), Apple (AAPL), Boeing (BA), Caterpillar (CAT), Chevron (CVX), Cisco Systems (CSCO), CocaCola (KO), ExxonMobil (XOM), Goldman Sachs (GS), The Home Depot (HD), IBM (IBM), Intel (INTC), Johnson&Johnson (JNJ), JPMorgan Chase (JPM), McDonald’s (MCD), Merck&Company (MRK), Microsoft (MSFT), Nike (NKE), Pfizer (PFE), Procter&Gamble (PG), Travelers (TRV), UnitedHealth Group (UNH), United Technologies (UTX), Verizon (VZ), Visa (V), Walmart (WMT), Wargreens Boots Alliance (WBA) and Walt Disney (DIS).

The standard approach to quantitatively describe statistical dependencies and their time trends, each of them having unique specific interpretation.

The simplest application is modelling pairwise statistical dependencies. After normalization of all variables to nearly uniform distribution (with Laplace CDF like previously), pairwise joint distribution would be nearly uniform on $[0,1]^2$ if they are uncorrelated. To this $p_0 = 1$ initial density we add independently calculated correction as products of two orthonormal polynomials - coefficients of the first three for all pairs are presented in Fig. 8 and 9. There are also presented linear time trends for two of them: using first coefficient with time rescaled to $[0,1]$ as additional variable like in the previous section. The diagonal terms have no meaning in this methodology - are filled with a constant value.

These additional coefficients allow for deeper understanding of statistical dependencies, like between growth of one
variable and variance of the second (12) or between their variances (22). Time trends (calculated for previous decade) may suggest further evolution of these dependencies. These coefficients are similar to multivariate mixed cumulants, but having a direct translation to probability density.

Figure 7 also contains first 5 eigenvectors of covariance matrix as in standard PCA technique. They correspond to largest variance directions and often turn out to group dependent variables. Here we usually estimate density as idealized distribution (e.g. multivariate Gaussian from PCA, or Laplace) by polynomial, estimating approach is to multiply idealized distribution (e.g. multivariate Gaussian from PCA, or Laplace) by polynomial, estimating approach is to multiply idealized distribution (e.g. multivariate Gaussian from PCA, or Laplace) by polynomial, estimating approach is to multiply idealized distribution (e.g. multivariate Gaussian from PCA, or Laplace) by polynomial, estimating approach is to multiply idealized distribution (e.g. multivariate Gaussian from PCA, or Laplace) by polynomial, estimating

\[ \sum_{\alpha \beta} c_{\alpha \beta} f_2(x_\alpha) f_2(x_\beta) = \sum_{k=1}^{n} \lambda_k \left( \sum_{\alpha} v_{\alpha}^k f_2(x_\alpha) \right)^2 \]

Presented matrices describe only pair-wise statistical dependencies, which in real models can be extended, for example: multivariate Gaussian from PCA, or Laplace) by polynomial, estimating approach is to multiply idealized distribution (e.g. multivariate Gaussian from PCA, or Laplace) by polynomial, estimating approach is to multiply idealized distribution (e.g. multivariate Gaussian from PCA, or Laplace) by polynomial, estimating approach is to multiply idealized distribution (e.g. multivariate Gaussian from PCA, or Laplace) by polynomial, estimating approach is to multiply idealized distribution (e.g. multivariate Gaussian from PCA, or Laplace) by polynomial, estimating approach is to multiply idealized distribution (e.g. multivariate Gaussian from PCA, or Laplace) by polynomial, estimating

\[ \sum_{\alpha \beta} c_{\alpha \beta} f_2(x_\alpha) f_2(x_\beta) = \sum_{k=1}^{n} \lambda_k \left( \sum_{\alpha} v_{\alpha}^k f_2(x_\alpha) \right)^2 \]

VI. EXTENSIONS

The used example presented basic methodology for educative reasons, which in real models can be extended, for example:

- Selective choice of basis: we have used complete basis of polynomials, what makes its \((m+1)^d\) size impractically large especially for high dimensions. However, usually only a small percentage of coefficients is above noise - we can selectively choose and use a sparse basis of significant values instead - describing real statistical dependencies. A simpler option is to selectively reduce polynomial degree for some of variables, or for example restrict the real degree of the polynomial: \( \sum_i j_i \) instead of each coordinate.

- Long-range value prediction: combination with state-of-art prediction models exploiting long-range dependencies, for example using a more sophisticated (than just the previous value) predictor of the current value.

- Improving information content of context used for prediction: instead of using a few previous values as the context, we can use some features e.g. describing long-range behavior like average over a time window, or for example obtained from dimensionality reduction methods like PCA (principal component analysis).

- Multivariate time series usually allow for much better prediction, as presented in [6]. Using for example macroeconomic variables should improve prediction.

APPENDIX

This appendix contains Wolfram Mathematica source for discussed procedures for stationary process, optimized to use built-in vector operations:

```mathematica
im = Import["c:/djia-100.xls"];
v = Log[Transpose[im[[1]]]][[2, 2 ;; -1]]; Print[ListPlot[v]]; n0 = Length[v]; yt = Table[v[[i + 1]] - v[[i]], {i, n1 = n0 - 1}]; syt = Sort[yt]; (* for approximated CDF *) mu = Median[yt]; (* Laplace estimation *) b = Mean[Abs[yt - mu]]; cdfL = If[y < mu, Exp[(y - mu)/b]/2, 1 - Exp[-(y - mu)/b]/2]; Print["Laplace distribution: mu = " <> mu <> ", b = " <> b <> "; ]; Print[Show[ListPlot[Table[[syt[[1]], {i, 0.5}/n1, {i, n1}]], Plot[cdfL, {y, -0.1, 0.1}, PlotStyle -› {Thin, Red}]]; xt = Table[cdfL/. y -» yt[[i]], {i, n1}]; (* normalized *) Print[ListPlot[Sort[xt]]]; Print[ListPlot[xt]]; cl = 3; d = d + 1 + cl; (* dimension = 1 + context length *) m = 4; (* maximal degree of polynomial *) coef = Power[m + 1, d]; Print[coef, " coefficients"]; p = Table[Power[x, k], {k, 0, m}]; p = Simplify[Orthogonalize[p, Integrate[#1 #2, {x, 0, 1}] &]]; Print["used orthonormal polynomials: ", p]; n = n1 cl; (* final number of data points *) (* table of contexts and their polynomials: *) ct = Transpose[Table[xt[[i = cl = i ;; -1]], {i, n1}]]; ctp = Table[If[j = i, Power[ct, 0], j]] /; x -» ct, {j, m + 1}]; (* calculate coefficients *) coef = Table[ct[[j = IntegerDigits[jn, m + 1, d] + 1; Mean[Product[ctp[[j = c] = cj]], {c, d}]]], {jn, 0, coefn - 1}]; (* find ID polynomials for various times: *) pt = Table[0, {i, m + 1}, {i, n}]; Do[jt = IntegerDigits[jn, m + 1, d] + 1; pt[[jt = IntegerDigits[jn, m + 1, d] + 1; pt[[j = IntegerDigits[jn, m + 1, d] + 1; pt[[j = IntegerDigits[jn, m + 1, d] + 1; pt[[j = IntegerDigits[jn, m + 1, d] + 1; (* probability normalization to 1: *) Do[pt[[i]] = pt[[i]], {i, m + 1, 1, -1}]; (* predicted densities for observed values: *) rho = Sum[ctp[[i = 1]], pt[[i]], {i, m + 1}], Print[ListPlot[Sort[rho]]]; (* densities in 10 random times: *) plst = RandomInteger[{1, n}, 10]; pl = Table[1 = plst[[k]]]; Sum[pt[= j, k], {j, m + 1}, {k, Length[plst]}]; ListPlot[pt, {x, 0, 1}, PlotRange -› {{0, 1}, {0, 5}}]
```

REFERENCES

[1] M. Rosenblatt et al., “Remarks on some nonparametric estimates of a density function,” The Annals of Mathematical Statistics, vol. 27, no. 3, pp. 832–837, 1956.

[2] A. Hyvärinen and E. Oja, “Independent component analysis: algorithms and applications,” Neural networks, vol. 13, no. 4, pp. 411–430, 2000.

[3] J. Duda, “Hierarchical correlation reconstruction with missing data,” arXiv preprint arXiv:1804.06218, 2018.

[4] ——, “Hierarchical correlation reconstruction with missing data,” arXiv preprint arXiv:1804.06218, 2018.

[5] J. Shohat, J. Tamarkin, and A. Society, The Problem of Moments, ser. Mathematical Surveys and Monographs. American Mathematical Society, 1943. [Online]. Available: https://books.google.pl/books?id=xGaqjHlOKC

[6] J. Duda and M. Snarska, “Modeling joint probability distribution of yield curve parameters,” arXiv preprint arXiv:1807.11743, 2018.

[7] M. J. Weinberger, G. Seroussi, and G. Sapiro, “The lico-i lossless image compression algorithm: Principles and standardization into jpeg-ls,” IEEE Transactions on Image processing, vol. 9, no. 8, pp. 1309–1324, 2000.

[8] M. K. Varanasi and B. Aazhang, “Parametric generalized gaussian density estimation,” The Journal of the Acoustical Society of America, vol. 86, no. 4, pp. 1404–1415, 1989.

[9] B. Mandelbrot, “The pareto-levy law and the distribution of income,” International Economic Review, vol. 1, no. 2, pp. 79–106, 1960.

[10] L. Qi, “Eigenvalues of a real supersymmetric tensor,” The Annals of Mathematical Statistics, vol. 27, no. 3, pp. 832–837, 1956.
Figure 8. Top: normalization of each variable to nearly uniform distribution like in Fig. 2 using separately estimated Laplace (left) or Gauss (right) distribution for 29 companies. We can see that the later is far from uniform distribution (line), especially the tail behavior - what means (often dangerous) underestimation of probability of extreme events, hence Laplace distribution is further used. Center: pairwise joint distribution for pairs of normalized variables would be nearly uniform on \([0, 1]^2\) if uncorrelated - the presented matrices contain coefficients for ten-year average of \(f_1(x_\alpha) f_2(x_\beta)\) correction to this uniform joint distribution for all pairs (upper) and their linear in time coefficient (lower): \(f_1(x_\alpha) f_1(x_\beta) f_1(t/n)\). Bottom: conformation of this general trend of lowering dependencies by its adaptive calculation for some 2 chosen companies with all 28 remaining.

Figure 9. Ten-year average coefficients for two succeeding corrections to uniform pair-wise joint distributions. Top: \(f_1(x_\alpha) f_2(x_\beta)\) coefficients describing growth or reduction of variance of the second variable with growth of the first variable. Obtained matrix is no longer symmetric, we can clearly see blue columns of companies having lower influence on variance of others. Center: its linear time trend - coefficients of \(f_1(x_\alpha) f_2(x_\beta) f_1(t/n)\). Bottom: \(f_2(x_\alpha) f_2(x_\beta)\) coefficients describing dependencies of variance - it turns out always positive here, meaning that the larger coefficient, the less likely that only one from the pair obtains extreme value - increasing risk for using both. Its linear time trend is not presented, but it turned out always negative like for \(f_1(x_\alpha) f_1(x_\beta)\), meaning weakening of dependencies. Diagonal terms are meaningless so they are filled with 0. While these corrections are formally degree 3 or 4 polynomials, restricting to only some of them allows to use eigenvectors of above matrices.