Risk Aggregation

Paul Embrechts

Department of Mathematics, ETH Zurich

Senior SFI Professor

www.math.ethz.ch/~embrechts/

Joint work with P. Arbenz and G. Puccetti

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

RiskLab® Switzerland

The background

Query by practitioner (2005):

> *Calculate VaR for the sum of three random variables with given marginals (Pareto, gamma, lognormal) and across a variety of dependence structures (copulas)*

Research project:

> Numerical evaluation of (generalized) copula convolutions, leading to (G)AEP.

Risk aggregation is relevant for:

- portfolio analysis
- understanding diversification & concentration
- for regulatory capital calculations
  - between risk categories
  - within risk categories

  within the Basel III, Solvency 2, SST frameworks
- better understanding of diversification
- we shall only touch upon some aspects

New publication by the Bank for International Settlements

Basel Committee
on Banking Supervision

Joint Forum

Developments in Modelling Risk Aggregation

October 2010

BANK FOR INTERNATIONAL SETTLEMENTS

A canonical set-up

- $X_1, \ldots, X_d$ one-period risks
- $\psi : \mathbb{R}^d \to \mathbb{R}$ aggregation function
- $\mathcal{R}$ a risk measure

Task: calculate $\mathcal{R}(\psi(X_1, \ldots, X_d))$

Example: $\psi(X_1, \ldots, X_d) = X_1 + X_2 + \cdots + X_d$, $\mathcal{R} = \mathrm{VaR}_\alpha$, $\alpha \in (0, 1)$

$$\mathrm{VaR}_\alpha(X_1 + X_2 + \cdots + X_d)$$

At best:

$$R_L \leq \mathcal{R}(\psi(\mathbf{X})) \leq R_U$$

depending on the underlying model assumptions!

Key issues

- Conditions:
  - $X_i \sim F_i, i = 1, \ldots, d$
  - known?/unknown?/unknowable?
  - risk versus uncertainty
  - statistical uncertainty
  - model uncertainty
- Dimensionality:
  - small: $d \leq 5$, say, versus
  - large: $d \sim 100s$
- Extremes matter:
  - in the tails: Extreme Value Theory (EVT)
  - in the interdependence: copulas (may) enter

$$\mathbb{P}[X_1 \leq x_1, \ldots, X_d \leq x_d] = C\left(F_1(x_1), \ldots, F_d(x_d)\right)$$

Return to canonical example:

$$\text{VaR}_\alpha(X_1 + X_2 + \cdots + X_d)$$

Issues:

- Relevance: sense or nonsense?
- Estimation, calculation
- additive ($=$) for comonotonic risks
  subadditive ($\leq$) for elliptical risks
  superadditive ($>$) for
  - very heavy-tailed risks
  - very skewed risks
  - risks with a special interdependence

  does it matter?
- measure of frequency (if), not severity (what if)

VaR in finance and insurance

Concerning VaR-calculations in finance and insurance:

- the VaR-number is just the final-final issue
- getting the risk-factor-mapping, clean-P&L are far more important
- recall: VaR is a statistical estimate
- often upper (lower) bounds can be found
- find (best) worst case VaR given some side conditions

Example for an upper bound for VaR

Theorem (Embrechts-Puccetti)
Let $(X_1, \dots, X_d)$ be continuous with equal margins $F_i = F$, $i = 1, \dots, d$. Then for $\alpha \in (0, 1)$,

$$\mathrm{VaR}_\alpha(X_1 + \cdots + X_d) \leq D_d^{-1}(1 - \alpha),$$

where

$$D_d(s) = \inf_{r \in [0, s/d)} \frac{\int_r^{s-(d-1)r} (1 - F(x))\mathrm{d}x}{s/d - r}$$

This talk (as an example):

Numerically calculate, for α close to 1,

$$\mathrm{VaR}_\alpha(X_1 + X_2 + \cdots + X_d) \tag{1}$$

or equivalently, calculate, typically for s large:

$$\mathbb{P}[X_1 + X_2 + \cdots + X_d \leq s] \tag{2}$$

numerically in terms of $F_1, \ldots, F_d$ and C which are assumed to be known analytically

Remark: in order to calculate (1) for a given α, use a root-finding procedure based on (2)

Standard solution

Monte Carlo: simulate i.i.d.

$$(X_1^i, X_2^i, \ldots, X_d^i), \qquad i = 1, \ldots, n$$

and estimate

$$\mathbb{P}\left[X_1 + X_2 + \cdots + X_d \leq s\right] \approx \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\left\{X_1^i + X_2^i + \cdots + X_d^i \leq s\right\}$$

(Dis)advantages:

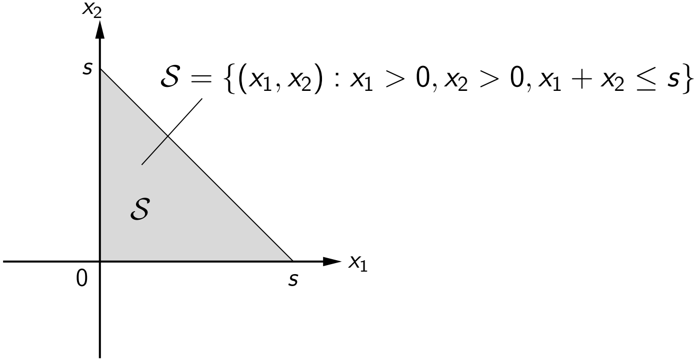
- A sampling algorithm must be available
- The convergence rate is relatively slow: $O(1/\sqrt{n})$
- The convergence rate is independent of the dimension d

The AEP algorithm: First assumption

First assumption:
The components of $(X_1, X_2, \ldots, X_d)$ are positive: $\mathbb{P}[X_i > 0] = 1$ (or bounded from below)

Consequence: Suppose $d = 2$. Due to $X_1 > 0$ and $X_2 > 0$ we get

$$\mathbb{P}[X_1 + X_2 \leq s] = \mathbb{P}[(X_1, X_2) \in \mathcal{S}]$$



The AEP algorithm: Second assumption

Second assumption:
The joint distribution function (df)

$$H(x_1, \dots, x_d) = \mathbb{P}[X_1 \le x_1, X_2 \le x_2, \dots, X_d \le x_d]$$

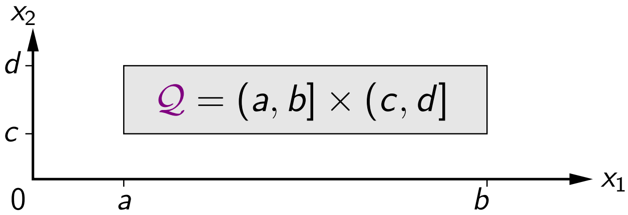
is known analytically or can be numerically evaluated

Example: H is given by a copula model:

$$H(x_1, \dots, x_d) = C(F_1(x_1), \dots, F_d(x_d))$$

The probability mass of a rectangle is easy to calculate

For $d = 2$



Then

$$\mathbb{P}[(X_1, X_2) \in \mathcal{Q}] = H(b, d) - H(a, d) - H(b, c) + H(a, c)$$

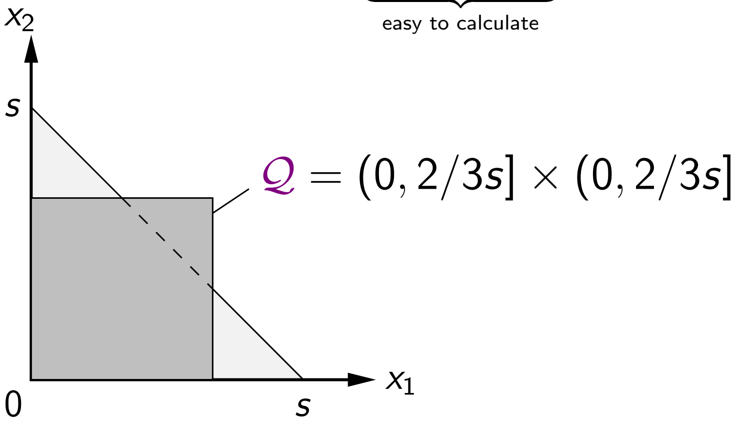
Idea behind the AEP algorithm: approximate the triangle $\mathcal{S}$ by rectangles!

First approximation ($d = 2$)

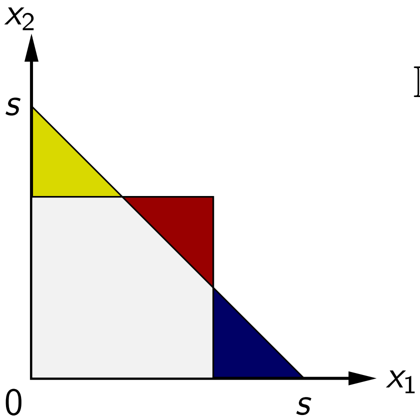
- Recall: $\mathcal{S} = \{(x_1, x_2) \in \mathbb{R}^2 : x_1 > 0, x_2 > 0, x_1 + x_2 \leq s\}$
- Set: $\mathcal{Q} = (0, 2/3s] \times (0, 2/3s]$ (later: why 2/3)

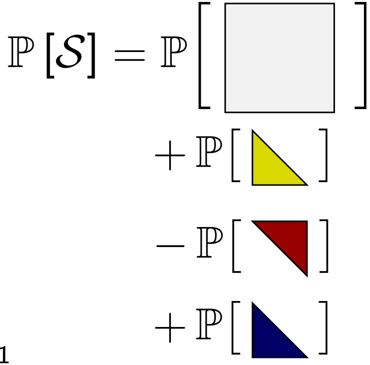
Use $\mathcal{Q}$ as a first approximation of $\mathcal{S}$

$$\mathbb{P}[(X_1, X_2) \in \mathcal{S}] \approx \underbrace{\mathbb{P}[(X_1, X_2) \in \mathcal{Q}]}_{\text{easy to calculate}}$$



Error of the first approximation

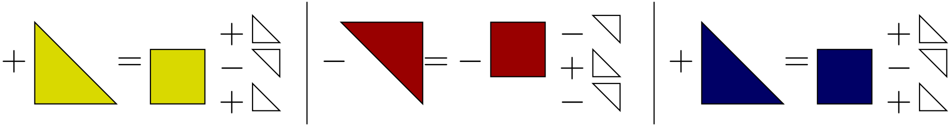




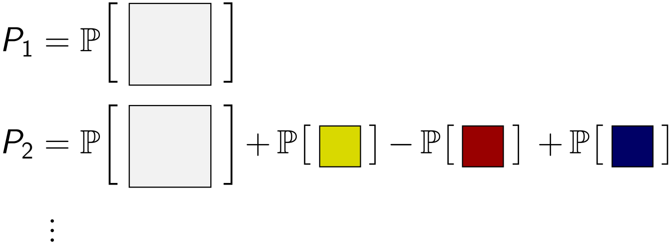
The error of the first approximation $\mathbb{P}[(X_1, X_2) \in \mathcal{Q}]$ can again be expressed in terms of triangles!

Idea: again approximate those triangles by squares!

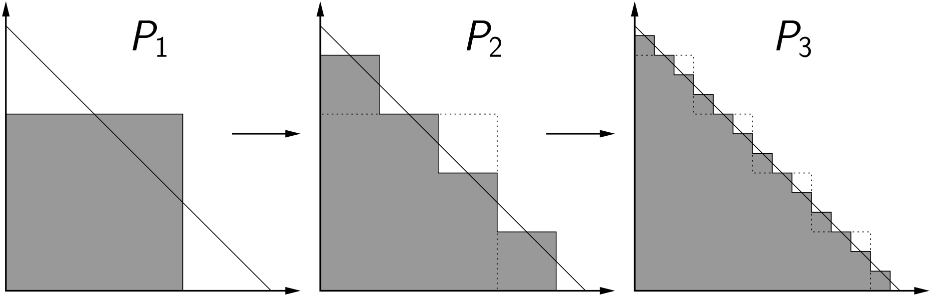
Approximate new triangles by squares



With these geometric approximations of $\mathcal{S}$, define a sequence P_n of approximations of $\mathbb{P}[X_1 + X_2 \leq s] = \mathbb{P}[(X_1, X_2) \in \mathcal{S}]$:



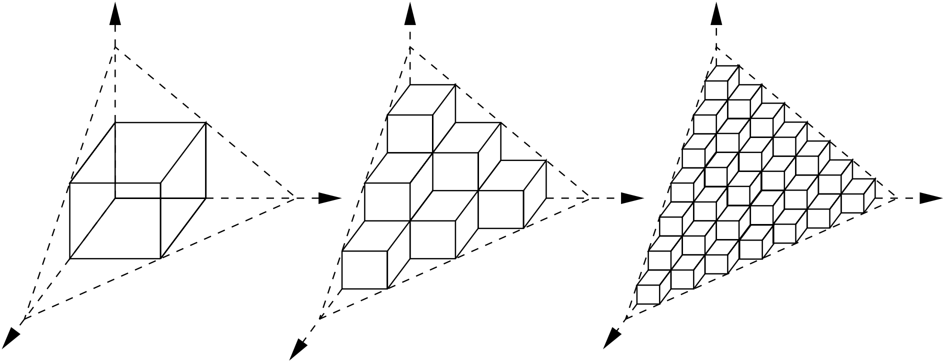
Set representation of P_1, P_2 and P_3



Triangles are iteratively approximated by squares and the left over triangles are then passed on to the next iteration

AEP algorithm for $d = 3$

In higher dimensions, the AEP can also be used.
For instance, for $d = 3$, the set representation of P_1, P_2 and P_3 is



Analogous decomposition possible in any dimension $d \in \mathbb{N}$,
but resulting simplexes are *overlapping* for $d \geq 4$!

Choice of the sidelengths of the approximating hypercubes

How to choose the sidelengths of the approximating hypercubes?

Answer: For an optimal rate of convergence, take a hypercube with sidelength

$$h = \frac{2}{d+1} \times (\text{sidelength of the triangle})$$

Hence the choice of $\mathcal{Q} = (0, 2/3s] \times (0, 2/3s]$ before for $d = 2$

Convergence

Theorem
Let $d \leq 5$ and suppose $(X_1, \ldots, X_d)$ has a density in a neighbourhood of $\{\mathbf{x} \in \mathbb{R}^d : \sum x_i = s\}$, then

$$\lim_{n \to \infty} P_n = \mathbb{P}\left[X_1 + \cdots + X_d \leq s\right]$$

Remark: reason for convergence problems in high dimensions: simplex decomposition is overlapping for $d \geq 4$

Richardson extrapolation

Define the extrapolated estimator P_n^* of $\mathbb{P}[X_1 + \cdots + X_d \leq s]$ by

$$P_n^* = P_n + a\,(P_n - P_{n-1}),$$

where $a = 2^{-d}(d+1)^d/d! - 1$.
The additional term cancels the dominant error term of P_n

Theorem
Let $d \leq 8$ and suppose $(X_1, \ldots, X_d)$ has a twice continuously differentiable density in a neighbourhood of $\{\mathbf{x} \in \mathbb{R}^d : \sum x_i = s\}$, then

$$\lim_{n\to\infty} P_n^* = \mathbb{P}[X_1 + \cdots + X_d \leq s]$$

Remark: for $d > 8$, higher order extrapolation may be useful for proving convergence

Convergence rates

Theorem

- Let $d \leq 5$ and suppose $(X_1, \ldots, X_d)$ has a density in a neighbourhood of $\{\mathbf{x} \in \mathbb{R}^d : \sum x_i = s\}$, then

$$|P_n - \mathbb{P}[X_1 + \cdots + X_d \leq s]| = O\big((A_d)^n\big)$$

- Let $d \leq 8$ and suppose $(X_1, \ldots, X_d)$ has a twice continuously differentiable density in a neighbourhood of $\{\mathbf{x} \in \mathbb{R}^d : \sum x_i = s\}$, then

$$|P_n^* - \mathbb{P}[X_1 + \cdots + X_d \leq s]| = O\big((A_d^*)^n\big)$$

	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$	$d=7$	$d=8$
A_d	0.333	0.500	0.664	0.925	–	–	–
A_d^*	0.037	0.125	0.234	0.358	0.498	0.656	0.8314

Convergence rates, cont.

The calculation of P_n and P_n^* requires $N(n) = O\left((B_d)^n\right)$ evaluations of the joint distribution function

	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$	$d=7$	$d=8$
B_d	3	4	15	21	63	92	255

Both convergence rate and numerical complexity of P_n and P_n^* are exponential. Combining both, we get

$$|P_n - \mathbb{P}[X_1 + \cdots + X_d \leq s]| = O\left(N(n)^{-\gamma_d}\right)$$
$$|P_n^* - \mathbb{P}[X_1 + \cdots + X_d \leq s]| = O\left(N(n)^{-\gamma_d^*}\right)$$

where γ_d and γ_d^* determine the rate of convergence.

Convergence rates, cont.

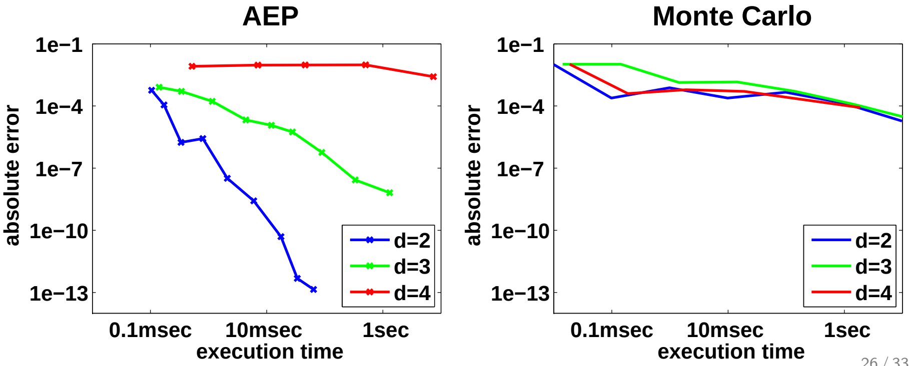
The following table shows γ_d and γ_d^*

	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$	$d = 7$
γ_d	1	0.5	0.15	0.05	–	–
γ_d^*	3	1.5	0.54	0.34	0.17	0.09

- Convergence rate of *Monte Carlo*: $O\left(N^{-0.5}\right)$,
  where N is the number of simulations.
  BUT: a (complex?) sampling algorithm must be available.
- Convergence rate of *Quasi Monte Carlo* $O\left(N^{-1}(\log N)^d\right)$.
  BUT: the algorithm must be tailored for each application.
- AEP does not need any tailoring or simulation.
  Only requirement: able to evaluate the joint distribution function of $(X_1, \ldots, X_d)$.

Numerical example

- $d = 2, 3, 4$
- X_i are Pareto(i) distributed ($\mathbb{P}[X_i \leq x] = 1 - (1 + x)^{-i}$)
- Clayton copula with $\theta = 2$ (pairwise Kendall's tau $= 0.5$)
- $s = 100$
- plot shows logarithm absolute errors: difference between estimate (extrapolated AEP & MC) and reference value x-axis, execution time on log scale



Numerical example: Conclusion

- In two and three dimensions, AEP is much faster than Monte Carlo
- For $d \geq 4$, Monte Carlo beats AEP
- Memory requirements to calculate P_n with AEP grow exponentially in n and in the dimension d, hence only low dimensions are numerically feasible

AEP in general:
INPUT:

- marginal dfs F_i
- copula C
- threshold s

OUTPUT:

- sequence P_n of estimates of $\mathbb{P}[X_1 + \cdots + X_d \leq s]$

SOFTWARE: available in C++

Open problem

Recall: using Richardson extrapolation,

$$P_n^* = P_n + a(P_n - P_{n-1})$$

for some $a \in \mathbb{R}$ converges faster and in higher dimensions than P_n

Further work:
Extend Richardson extrapolation to cancel higher order error terms!
Possibly through estimators of the following form?

$$\begin{aligned} P_n^{**} &= P_n + b_1(P_n - P_{n-1}) + b_2(P_{n-1} - P_{n-2}) \\ P_n^{***} &= P_n + c_1(P_n - P_{n-1}) + c_2(P_{n-1} - P_{n-2}) + c_3(P_{n-2} - P_{n-3}) \\ &\vdots \end{aligned}$$

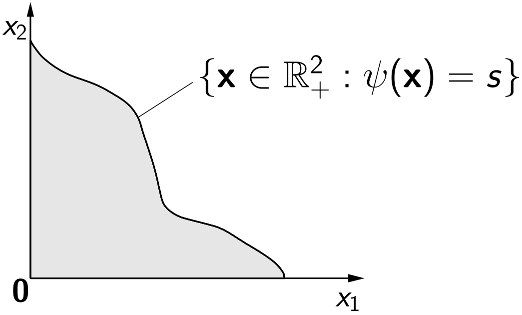
The GAEP algorithm

GAEP (Generalized AEP) concerns more general aggregation functionals, i.e. the estimation of

$$\mathbb{P}[\psi(X_1, \ldots, X_d) \leq s],$$

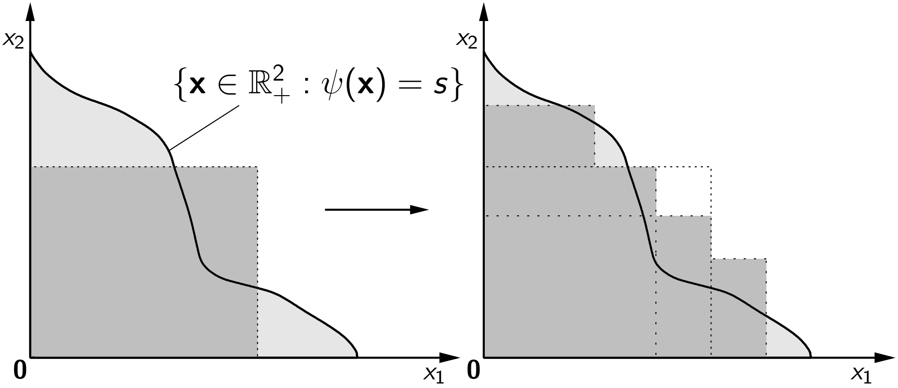
where $\psi : \mathbb{R}^d \to \mathbb{R}$ is a continuous function that is strictly increasing in each coordinate.

This probability can be represented as the mass of some "generalized triangle":



GAEP generalized triangle decomposition

Analogous to the AEP algorithm, we can decompose a generalized triangle into a rectangle and further generalized triangles:



GAEP: short summary

- **Issue:** how to choose the sidelengths of the approximating hypercubes (rectangles)? Paper proposes different possibilities
- **Performance:** Similar to AEP, very good for $d = 2, 3$, acceptable for $d = 4$ and not competitive for $d \geq 5$
- **Open problems:**
  - A proof for an optimal choice of the hypercube sidelengths
  - Extension of the extrapolation technique as used for AEP

References

- P. Arbenz, P. Embrechts, G. Puccetti: *The AEP algorithm for the fast computation of the distribution of the sum of dependent random variables*. Bernoulli **17**(2), 2011, 562–591

- P. Arbenz, P. Embrechts, G. Puccetti: *The GAEP algorithm for the fast computation of the distribution of a function of dependent random variables*. (Forthcoming in Stochastics, 2011)

- P. Embrechts, G. Puccetti: *Risk Aggregation*. In: *Copula Theory and its Applications*, P. Jaworski, F. Durante, W. Haerdle, and T. Rychlik (Eds.). Lecture Notes in Statistics - Proceedings 198, Springer Berlin/Heidelberg, pp. 111-126

- Software (C++ code) to be obtained through the authors

Thank you