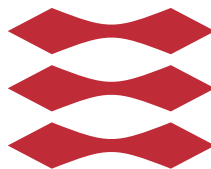


Causal discovery and inference of data from production flows

Jonas Bruun Hubrechts

DTU



Kongens Lyngby 2024

Technical University of Denmark
Department of Applied Mathematics and Computer Science
Richard Petersens Plads, building 324,
2800 Kongens Lyngby, Denmark
Phone +45 4525 3031
compute@compute.dtu.dk
www.compute.dtu.dk

Summary (English)

The goal of the thesis is to ...

Summary (Danish)

Målet for denne afhandling er at ...

Preface

This thesis was prepared at DTU Compute in fulfilment of the requirements for acquiring an M.Sc. in Engineering.

The thesis deals with ...

The thesis consists of ...

Lyngby, 31-July-2024



Not Real

Jonas Bruun Hubrechts

Acknowledgements

I would like to thank my....

Contents

Summary (English)	i
Summary (Danish)	iii
Preface	v
Acknowledgements	vii
1 Time to Level of Brownian motion	1
1.1 Arrivals of batches	1
1.1.1 Joint distribution of Brownian and its running maximum	2
1.1.2 Joint distribution with drift and arbitrary variance	4
1.1.3 Distribution of maximum of Brownian motion with drift .	6
1.1.4 Cumulative distribution of maximum	6
1.1.5 Distribution of time to level	7
1.1.6 MGF	8
2 Problemformulering / Introduktion	13
3 Ideer til hvad der skal laves	15
4 Data	17
4.1 Basic statistics	18
4.2 Incompleteness on trailing batches	19
4.3 Production phases	21
4.3.1 Correlations	24
4.4 Cleaning operations	29

5	Method	37
5.1	Causal discovery	37
5.1.1	Setup and assumptions	38
5.2	Information measures and computation	41
5.2.1	Copula	41
5.2.2	Mutual information and Copula entropy	43
5.2.3	Correlation	47
5.3	Copula based network discovery	47
5.4	Algorithms	48
5.5	Examples	50
5.5.1	Ensuring convergence and the effect of β	57
5.5.2	Robustness to noise	58
5.5.3	Quick made up example	59
A	Stuff	61
	Bibliography	63

CHAPTER 1

Time to Level of Brownian motion

1.1 Arrivals of batches

Assuming that the in-flow from the previous section in the production obeys the following SDE

$$dS_t = rdt + \sigma dB_t$$

I.e. Brownian motion with drift. And assuming that every time the accumulated mass hits a level l , the batch is ready to be processed by the next step, we wish to first find the distribution for these times. Note that the above model allows for negative flow and thus also negative accumulated mass. However, for $\sigma \ll r$ this becomes very unlikely as

$$\mathbb{P}(S_t \leq 0) = \Phi\left(\frac{-r\sqrt{t}}{\sigma}\right)$$

Thus, only for small t this is probable, as otherwise it is dominated by $\frac{r}{\sigma}$ which

is large and thus the probability very low.

Furthermore, if one allows periods without inflow, the running maximum could be a good model. Either way, the probability distribution for between batch times is the same.

To derive the distribution for the between batch times, T , we shall use the Girsanov Theorem as well as the joint distribution of the maximum of a standard Brownian motion and its running maximum. Thus, let B_t be a standard Brownian motion, and M_t the running maximum defined as

$$M_t := \sup_{s \in [0, t]} \{B_s\}$$

- Udlædning af joint fordeling mellem M og B
- Change of measure for at opnå med drift og sigma
- Marginal fordeling for M

1.1.1 Joint distribution of Brownian and its running maximum

To derive the joint density of a standard Brownian motion and its running maximum, consider the following probability

$$\mathbb{P}(M_t \geq m, B_t \leq w)$$

Let T_m be defined as the first time B_t hits the level m , i.e. $T_m := \inf_t (B_t = m)$. Then $M_t \geq m \iff T_m \leq t$. Thus, the above probability is reexpressed as

$$\mathbb{P}(M_t \geq m, B_t \leq w) = \mathbb{P}(T_m \leq t, B_t \leq w)$$

To proceed, we use the principle of reflection which is admissible due to B_t being a martingale. In particular, we define \tilde{B}_t as follows

$$\tilde{B}_t := \begin{cases} B_t & t \leq T_m \\ 2m - B_t & t > T_m \end{cases}$$

It follows that \tilde{B}_t is also a standard Brownian motion. By the definition of \tilde{B}_t , we then have that

$$\mathbb{P}(T_m \leq t, B_t \leq w) = \mathbb{P}(T_m \leq t, 2m - w \leq \tilde{B}_t)$$

Notice that the original expression is only sensible for $m \geq w$ as $w > m$ is a contradiction to the definition of M_t . Thus, $2m - w \geq m$ hence $\tilde{B}_t \geq 2m - w$ implies that the original Brownian motion B_t has hit the level m and thus $T_m \leq t$. This means that

$$\mathbb{P}(T_m \leq t, 2m - w \leq \tilde{B}_t) = \mathbb{P}(2m - w \leq \tilde{B}_t) = 1 - \Phi\left(\frac{2m - w}{\sqrt{t}}\right)$$

Thus, in total we have found that

$$\mathbb{P}(M_t \geq m, B_t \leq w) = 1 - \Phi\left(\frac{2m - w}{\sqrt{t}}\right)$$

And thus, the joint distribution is obtained by differentiation

$$\begin{aligned} f_{M_t, B_t}(m, w) &= \frac{\partial^2}{\partial m \partial w} \mathbb{P}(M_t \leq m, B_t \leq w) \\ &= \frac{\partial^2}{\partial m \partial w} (\mathbb{P}(B_t \leq w) - \mathbb{P}(M_t \geq m, B_t \leq w)) \\ &= \frac{\partial^2}{\partial m \partial w} \Phi\left(\frac{2m - w}{\sqrt{t}}\right) \\ &= \frac{2(2m - w)}{t^{3/2}} \phi\left(\frac{2m - w}{\sqrt{t}}\right), \quad m \leq w, \quad m \geq 0 \end{aligned}$$

Note:

Now, define instead $\tilde{B}_t = \sigma B_t$. We then find a similar expression for the joint density of ... and its running maximum. Namely, as

$$\mathbb{P}(\tilde{M}_t \geq m, \tilde{B}_t \leq w) = \mathbb{P}(\sigma M_t \geq m, \sigma B_t \leq w)$$

Same formula, but with m and w divided by σ

1.1.2 Joint distribution with drift and arbitrary variance

Let B_t be a standard Brownian motion defined on the probability space, $(\Omega, \mathcal{F}, \mathbb{P})$. Furthermore, define \tilde{B}_t to be a Brownian motion with drift as follows

$$\tilde{B}_t := \tilde{\mu}t + B_t$$

To derive the joint density $f_{\tilde{M}_t, \tilde{B}_t}(m, w)$ on measure \mathbb{P} , we use a corollary of the Girsanov theorem. Namely, suppose B_t is Brownian motion under measure \mathbb{P} , then there exists a measure \mathbb{Q} such that $\tilde{B}_t = B_t - \langle B, X \rangle_t$ is a Brownian motion (without drift) under this new measure given that X_t is an adapted process. Furthermore, as \tilde{B}_t is a martingale, the Radon-Nikodym derivative is equal to the stochastic exponential $Z_t = \exp(X_t - \frac{1}{2} \langle X \rangle_t)$.

Now, if X_t is of the form $\int_0^t Y_s dB_s$ where $\mathbb{E}_{\mathbb{P}} \left[\exp \left(\frac{1}{2} \int_0^T Y_s^2 ds \right) \right] < \infty$, a special case, the Cameron-Martin-Girsanov implies that $\tilde{B}_t = B_t - \int_0^t Y_s ds$ is then a \mathbb{Q} Brownian motion. This can easily be shown when Y_s fulfills Noviko's condition, then Z_t is a martingale and the Girsanov theorem applies as clearly X_t is also adapted to B_t . Then, from the above corollary,

$$\begin{aligned} \tilde{B}_t &= B_t - \langle B, X \rangle_t \\ &= B_t - \lim_{||P|| \rightarrow 0} \sum_i (B_{t_{i+1}} - B_{t_i}) \left(\int_{t_i}^{t_{i+1}} Y_s dB_s \right) \\ &= B_t - \lim_{||P|| \rightarrow 0} \sum_i (B_{t_{i+1}} - B_{t_i})^2 Y_{t_i}^* \\ &= B_t - \int_0^t Y_s ds \end{aligned}$$

As it has now been shown that there exists as measure \mathbb{Q} under which \tilde{B}_t is a Brownian motion as choosing $Y_s = -\tilde{m}u$ we reproduce the initial definition of \tilde{B}_t . To then derive the joint distribution of \tilde{B}_t and its running maximum \tilde{M}_t ,

we compute the Radon-Nikodym derivative, Z_t , hence given by

$$\begin{aligned} \left. \frac{d\mathbb{Q}}{d\mathbb{P}} \right|_{\mathcal{F}_t} &= Z_t = \exp \left(\int_0^t Y_s dB_s - \frac{1}{2} \int_0^t Y_s^2 ds \right) \\ &= \exp \left(-\tilde{\mu} \int_0^t dB_s - \frac{1}{2} \tilde{\mu}^2 \int_0^t ds \right) \\ &= \exp \left(-\tilde{\mu} B_t - \frac{1}{2} \tilde{\mu}^2 t \right) \\ &= \exp \left(-\tilde{\mu} \tilde{B}_t + \frac{1}{2} \tilde{\mu}^2 t \right) \end{aligned}$$

With the above derivative, we have that

$$\mathbb{Q}(A) = \int_A Z_t d\mathbb{P}$$

And thus also

$$\mathbb{P}(A) = \int_A Z_t^{-1} d\mathbb{Q}$$

as $Z_t : X \rightarrow (0, \infty)$. It then simply follows that

$$f_{\tilde{M}_t, \tilde{B}_t}(m, w) = \tilde{f}_{\tilde{M}_t, \tilde{B}_t}(m, w) e^{\tilde{\mu}w - \frac{1}{2}\tilde{\mu}^2 t}$$

where \tilde{f} is the probability distribution under measure \mathbb{Q} . Hence,

$$f_{\tilde{M}_t, \tilde{B}_t}(m, w) = \frac{2(2m - w)}{t^{3/2}} e^{\tilde{\mu}w - \frac{1}{2}\tilde{\mu}^2 t} \phi \left(\frac{2m - w}{\sqrt{t}} \right)$$

To introduce the standard deviation σ , first define $\tilde{\mu} = \mu/\sigma$ and $\hat{B}_t = \sigma \tilde{B}_t$. Then, \hat{B}_t is also a Brownian with drift, μ , but with variance $\sigma^2 t$. Furthermore, the joint distribution is

$$f_{\hat{M}_t, \hat{B}_t}(m, w) = \frac{2(2m - w)}{\sigma^3 t^{3/2}} e^{\frac{1}{\sigma^2}(\mu w - \frac{1}{2}\mu^2 t)} \phi \left(\frac{2m - w}{\sigma \sqrt{t}} \right)$$

1.1.3 Distribution of maximum of Brownian motion with drift

The distribution of the running maximum \hat{M}_t is given by the marginal of the above, namely

$$f_{\hat{M}_t}(m) = \int_{-\infty}^m f_{\hat{M}_t, \hat{B}_t}(m, w) dw$$

Integration by parts admits

$$f_{\hat{M}_t}(m) = \frac{2}{\sigma\sqrt{t}} \phi\left(\frac{m - \mu t}{\sigma\sqrt{t}}\right) - \frac{2\mu}{\sigma^2} e^{\frac{2m\mu}{\sigma^2}} \Phi\left(-\frac{m + \mu t}{\sigma\sqrt{t}}\right)$$

1.1.4 Cumulative distribution of maximum

As we shall later need the survival function of \hat{M}_t , we first compute the cumulative distribution. Namely

$$\mathbb{P}(\hat{M}_t \leq m) = \int_0^m \int_{-\infty}^{\eta} f_{\hat{M}_t, \hat{B}_t}(\eta, w) dw d\eta$$

To compute the above, we split the inner integral over the line $w = 0$ in the η, w plane and reformulate

$$\mathbb{P}(\hat{M}_t \leq m) = \underbrace{\int_0^m \int_w^m f_{\hat{M}_t, \hat{B}_t}(\eta, w) d\eta dw}_{I_1} + \underbrace{\int_{-\infty}^0 \int_0^m f_{\hat{M}_t, \hat{B}_t}(\eta, w) d\eta dw}_{I_2}$$

The antiderivative of $f_{\hat{M}_t, \hat{B}_t}(m, w)$ w.r.t. m is simple and calculated to be

$$\int f_{\hat{M}_t, \hat{B}_t}(m, w) dm = -\frac{1}{\sigma\sqrt{2\pi t}} e^{\frac{1}{\sigma^2}(\mu w - \frac{1}{2}\mu^2 t)} e^{-\frac{1}{2}\left(\frac{2m-w}{\sigma\sqrt{t}}\right)^2}$$

The first of the above integrals, I_1 , is then

$$I_1 = -\frac{1}{\sigma\sqrt{2\pi t}} e^{-\frac{1}{2\sigma^2}\mu^2 t} \int_0^m e^{\frac{\mu w}{\sigma^2} - \frac{1}{2}\left(\frac{2m-w}{\sigma\sqrt{t}}\right)^2} - e^{\frac{\mu w}{\sigma^2} - \frac{1}{2}\left(\frac{w}{\sigma\sqrt{t}}\right)^2} dw$$

And similar for the second integral I_2

$$I_2 = -\frac{1}{\sigma\sqrt{2\pi t}} e^{-\frac{1}{2\sigma^2}\mu^2 t} \int_{-\infty}^0 e^{\frac{\mu w}{\sigma^2} - \frac{1}{2}\left(\frac{2m-w}{\sigma\sqrt{t}}\right)^2} - e^{\frac{\mu w}{\sigma^2} - \frac{1}{2}\left(\frac{w}{\sigma\sqrt{t}}\right)^2} dw$$

It is observed that the integrands are the same, thus

$$\mathbb{P}(\hat{M}_t \leq m) = -\frac{1}{\sigma\sqrt{2\pi t}} e^{-\frac{1}{2\sigma^2}\mu^2 t} \int_{-\infty}^m e^{\frac{\mu w}{\sigma^2} - \frac{1}{2}\left(\frac{2m-w}{\sigma\sqrt{t}}\right)^2} - e^{\frac{\mu w}{\sigma^2} - \frac{1}{2}\left(\frac{w}{\sigma\sqrt{t}}\right)^2} dw$$

From simple substitution, and a few calculations, one gets that

$$\mathbb{P}(\hat{M}_t \leq m) = \Phi\left(\frac{m - \mu t}{\sigma\sqrt{t}}\right) - e^{\frac{2m\mu}{\sigma^2}} \Phi\left(-\frac{m + \mu t}{\sigma\sqrt{t}}\right)$$

1.1.5 Distribution of time to level

As $\mathbb{P}(M_t \geq l) = \mathbb{P}(T_l \leq t)$. It thus follows that $f_{T_l}(t) = \frac{d}{dt}\mathbb{P}(M_t \geq l)$ which is easily calculated from the above. Namely

$$\begin{aligned} f_{T_l}(t) &= \frac{d}{dt} (1 - \mathbb{P}(M_t \leq l)) \\ &= -\frac{d}{dt} \left(\Phi\left(\frac{l - \mu t}{\sigma\sqrt{t}}\right) - e^{\frac{2l\mu}{\sigma^2}} \Phi\left(-\frac{l + \mu t}{\sigma\sqrt{t}}\right) \right) \\ &= \frac{\mu t + l}{2\sigma t^{3/2}} \phi\left(\frac{l - \mu t}{\sigma\sqrt{t}}\right) + \frac{l - \mu t}{2\sigma t^{3/2}} e^{\frac{2\mu l}{\sigma^2}} \phi\left(-\frac{\mu t + l}{\sigma\sqrt{t}}\right) \end{aligned}$$

Note that although the distribution above is parameterized by 3 parameters, it can be completely specified by $\tilde{\mu} = \mu/\sigma$ and $\tilde{l} = l/\sigma$ which is clear also from the following

Let $Z_t = \mu t + \sigma B_t$ and similarly $\tilde{Z}_t = Z_t/\sigma = \tilde{\mu} + B_t$. Then $\mathbb{P}(T_l \leq t) = \mathbb{P}(M_t \geq l) = \mathbb{P}(\tilde{M}_t \geq \tilde{l}) = \mathbb{P}(\tilde{T}_{\tilde{l}} \leq t)$ where \tilde{M}_t and $\tilde{T}_{\tilde{l}}$ are the running maximum and time to level of \tilde{Z}_t . Thus, equivalent to a probability of non-scaled Brownian motion with drift.

To verify the above probability distribution, a Monte-Carlo simulation is carried out for 100.000 simulations with parameters $l = 10$, $\mu = 0.1$, $\sigma = 0.5$. As the shape resembles a gamma distribution, a simple fit, matching the mean and variance is also plotted. Although the gamma family of probability distributions is also a two-parameter family, they do not quite overlap as can be seen in the following plot.

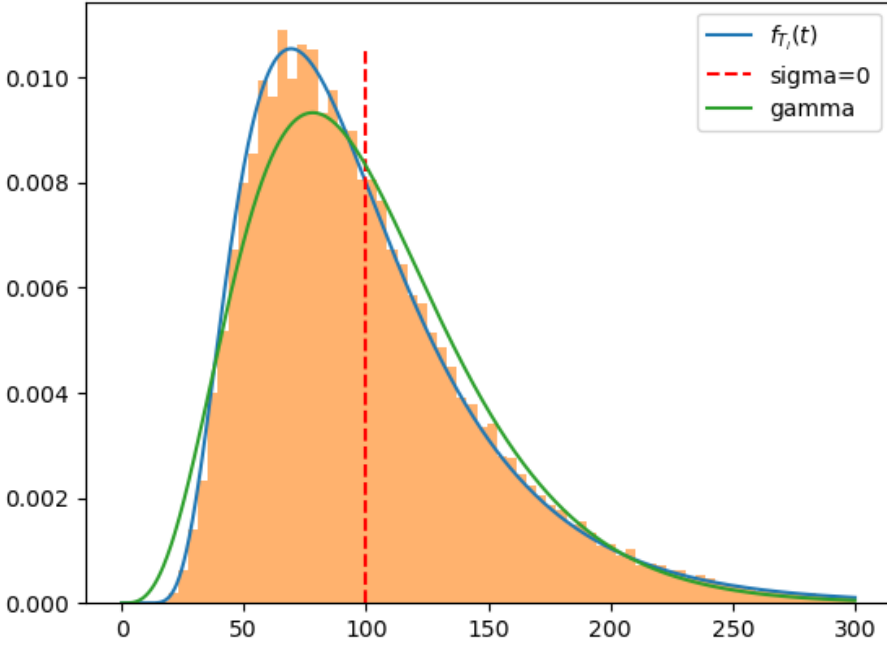


Figure 1.1: Example of simulation and actual distribution. The marked $\sigma = 0$ shows the limit as $\sigma \rightarrow 0$ corresponding to no noise on the input flow

1.1.6 MGF

$$\begin{aligned}
 \mathbb{E} [e^{\theta T_l}] &= \int_0^\infty e^{\theta t} f_{T_l}(t) dt \\
 &= \underbrace{\int_0^\infty e^{\theta t} \frac{\mu t + l}{2\sigma t^{3/2}} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{l - \mu t}{\sigma \sqrt{t}} \right)^2} dt}_{I_1} + e^{\frac{2\mu l}{\sigma^2}} \underbrace{\int_0^\infty e^{\theta t} \frac{l - \mu t}{2\sigma t^{3/2}} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left(-\frac{\mu t + l}{\sigma \sqrt{t}} \right)^2} dt}_{I_2}
 \end{aligned}$$

We shall only consider the first integral I_1 as the second follows directly from

the result of the first by substituting μ with $-\mu$

$$\begin{aligned}
 I_1 &= \int_0^\infty \frac{\mu t + l}{2\sigma t^{3/2}} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \frac{(l-\mu t)^2 - 2\theta\sigma^2 t^2}{\sigma^2 t}} dt \\
 &= \int_0^\infty \frac{\mu t + l}{2\sigma t^{3/2}} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \frac{(\mu^2 - 2\theta\sigma^2)t^2 - 2l\mu t + l^2}{\sigma^2 t}} dt \\
 &= e^{\frac{l\mu - l\sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma^2}} \int_0^\infty \frac{\mu t + l}{2\sigma t^{3/2}} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{\sqrt{\mu^2 - 2\theta\sigma^2} t - l}{\sigma\sqrt{t}} \right)^2} dt \\
 &= e^{\frac{l\mu - l\sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma^2}} \int_0^\infty \left(\frac{\sqrt{\mu^2 - 2\theta\sigma^2} t + l}{2\sigma t^{3/2}} + \frac{\mu - \sqrt{\mu^2 - 2\theta\sigma^2}}{2\sigma\sqrt{t}} \right) \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{\sqrt{\mu^2 - 2\theta\sigma^2} t - l}{\sigma\sqrt{t}} \right)^2} dt
 \end{aligned}$$

Once again, we split the integral, now as follows

$$\begin{aligned}
 I_1 &= e^{\frac{l\mu - l\sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma^2}} \left(\underbrace{\int_0^\infty \frac{\sqrt{\mu^2 - 2\theta\sigma^2} t + l}{2\sigma t^{3/2}} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{\sqrt{\mu^2 - 2\theta\sigma^2} t - l}{\sigma\sqrt{t}} \right)^2} dt}_{I_{11}} \right. \\
 &\quad \left. + \frac{\mu - \sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma} \underbrace{\int_0^\infty \frac{1}{2\sqrt{t}} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{\sqrt{\mu^2 - 2\theta\sigma^2} t - l}{\sigma\sqrt{t}} \right)^2} dt}_{I_{12}} \right)
 \end{aligned}$$

For the first integral, the substitution $u = \frac{\sqrt{\mu^2 - 2\theta\sigma^2} t - l}{\sigma\sqrt{t}}$ reveals that

$$I_{11} = \int_{-\infty}^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} u^2} du = 1$$

As for the second integral I_{12} it can be rewritten as

$$I_{12} = e^{\frac{l\sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma^2}} \int_0^\infty \frac{1}{2\sqrt{t}} \frac{1}{\sqrt{2\pi}} e^{-\left(\frac{\mu^2}{2\sigma^2} - \theta\right)t - \frac{l^2}{2\sigma^2} t^{-1}} dt$$

Then, substituting $u = \sqrt{\frac{\mu^2}{2\sigma^2} - \theta} \sqrt{t}$

$$I_{12} = e^{\frac{l\sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma^2}} \frac{1}{\sqrt{2\pi} \sqrt{\frac{\mu^2}{2\sigma^2} - \theta}} \int_0^\infty e^{-u^2 - \frac{l^2}{2\sigma^2} \left(\frac{\mu^2}{2\sigma^2} - \theta\right) u^{-2}} du$$

To solve the above integral, we consider the following family of integrals, parameterized by s

$$I(s) = \int_0^\infty e^{-u^2 - s^2 u^{-2}} du$$

It follows that

$$I'(s) = -2 \int_0^\infty e^{-u^2 - s^2 u^{-2}} s u^{-2} du$$

letting $z = s/u$, it follows that $dz = -s/u^2 du$ and (assuming $s > 0$)

$$I'(s) = -2 \int_0^\infty e^{-s^2 z^{-2} - z^2} dz = -2I(s)$$

Also,

$$I(0) = \int_0^\infty e^{-u^2} du = \frac{\sqrt{\pi}}{2}$$

Hence

$$I(s) = \frac{\sqrt{\pi}}{2} e^{-2s} \quad , \text{ for } s \geq 0$$

Note that due to symmetry, $I(s) = I(-s)$, and hence

$$I(s) = \frac{\sqrt{\pi}}{2} e^{-2|s|}$$

Thus, letting $s = \frac{l}{2\sigma} \sqrt{\frac{\mu^2}{\sigma^2} - 2\theta}$ i.e. resulting in the integral from I_{12} , the integral I_{12} is simply

$$\begin{aligned} I_{12} &= e^{\frac{l\sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma^2}} \frac{1}{\sqrt{2\pi}\sqrt{\frac{\mu^2}{2\sigma^2} - \theta}} \frac{\sqrt{\pi}}{2} e^{-2\frac{l}{2\sigma}\sqrt{\frac{\mu^2}{\sigma^2} - 2\theta}} \\ &= \frac{\sigma}{2\sqrt{\mu^2 - 2\theta\sigma^2}} \end{aligned}$$

Combining the above results, we finally have I_1

$$\begin{aligned} I_1 &= e^{\frac{l\mu - l\sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma^2}} \left(1 + \frac{\mu - \sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma} \frac{\sigma}{2\sqrt{\mu^2 - 2\theta\sigma^2}} \right) \\ &= e^{\frac{l\mu - l\sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma^2}} \left(\frac{1}{2} + \frac{\mu}{2\sqrt{\mu^2 - 2\theta\sigma^2}} \right) \end{aligned}$$

Similar calculations results in the integral I_2 or simply by letting $\mu = -\mu$ as discussed before. In total, we find the moment generating function to be

$$\begin{aligned} \mathbb{E} [e^{\theta T_l}] &= e^{\frac{l\mu - l\sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma^2}} \left(\frac{1}{2} + \frac{\mu}{2\sqrt{\mu^2 - 2\theta\sigma^2}} \right) \\ &\quad + e^{\frac{2\mu l}{\sigma^2}} e^{\frac{-l\mu - l\sqrt{\mu^2 - 2\theta\sigma^2}}{\sigma^2}} \left(\frac{1}{2} - \frac{\mu}{2\sqrt{\mu^2 - 2\theta\sigma^2}} \right) \\ &= e^{\frac{l}{\sigma^2} (\mu - \sqrt{\mu^2 - 2\theta\sigma^2})} \end{aligned}$$

From the above calculations, this is clearly defined for θ in some neighborhood of 0, thus the above is indeed a proper moment generating function. Furthermore, all derivatives exists at $\theta = 0$.

This also shows that T_l does not belong to neither the Gamma family nor Phase-Type

The first 3 moments are given by

$$\mathbb{E} [T_l] = \frac{l}{\mu}, \quad \mathbb{E} [T_l^2] = \frac{l\sigma^2}{\mu^3} + \frac{l^2}{\mu^2}, \quad \mathbb{E} [T_l^3] = \frac{3l\sigma^4}{\mu^5} + \frac{3l^2\sigma^2}{\mu^4} + \frac{l^3}{\mu^3}$$

Interestingly, the average is exactly what one would expect if no stochasticity was present. Furthermore, the variance has a simple nice form, namely $\frac{l\sigma^2}{\mu^3}$.

Continuing the simulation from above, the theoretical mean evaluates to 100 whereas the simulated mean evaluated to 100.343. The theoretical variance is 2500 whereas the variance from simulation was 2468.224.

CHAPTER 2

Problemformulering / Introduktion

In many production facilities, planning is a big part of maximizing some index. Whether this is production throughput over some time period and thus often also the economic surplus or some other key index, it is of great importance to have an underlying model to describe the observed variation. In particular in operational research, the schedules may drift in suboptimal ways if the variation is not considered.

Furthermore, from a salesman point of view, expected production and time intervals can be of great use when planning and also building production facilities. Namely, one might find that increasing the volume or efficiency of some part of the facility would increase the production throughput and profitability. This is also known as bottleneck analysis and require some understanding of the underlying mechanics and a stochastic model of this could improve the strength of such results.

Therefore, the primary objective of this paper/thesis is to investigate and model the yield and time of a production flow with focus on the pharmaceutical and chemical production industry. More precisely, we will be building a statistical model for a single process, with the purpose of being able to describe the variation in the yield of the production cycle and production times. This will then be used to analyze potential bottlenecks.

Furthermore, it will be interesting to construct a network of such processes as is typically the case in industry. We shall see how much can be said about such a network and what obstacles one may encounter when trying to analyze such networks which is this thesis will initially be treated as networks of queues.

CHAPTER 3

Ideer til hvad der skal laves

Overall model for throughput of system. I.e. model the system as e.g. a system of queues and how much is produced at each step and this propagate. The important aspect is breakdown (extra processing time) and possibility of having to throwing out some production along the way, either due to error or some other (unforeseen) causes.

Need to investigate different ways of modelling this (starting with a simple system with no queuing, i.e. a single batch; this is what is done above). Discuss the pros and cons and how much information they preserve (aggregation models etc. may need to model some part of the system by throwing away)

- Petri Net
- ODE Stochastic Chemical Reaction (first order)
- Database of pharmacokinetic time-series data
- Chemical Manufacturing Process Data'

In this section, we will describe and analyze the data used in this thesis. The data stems from a simulation study on a chemical batch production system and comprises six time series from which the time stamps (in hours) and level in a tank is of the most interest. The goal of this section is to describe how different phases of the production covary. The data is chosen as it resembles an actual production data set but is also implemented with what at first glance seem to be fairly realistic variation and noise in measurements. Overall, the point of this example data is to exemplify what one may encounter in a real batch production system and from this try to build a model in order to predict or quantify the behavior of the system or learn hidden (causal) structure important for optimization etc.

We define a set of phases/units \mathcal{U} that each batch comprises. In this case, units are identified with IDs 1 through 10 (with subunits such as 3.1) described further in Table 4.2. Then, for each unit, we define the stochastic variables X_u and X_u^D to be the duration and delay after a unit respectively. It is also important to keep track of the level in the tank after each unit is finished, and we thus define variables M_u and M_u^D to be the level in the tank after unit u and its associated delay respectively. As the units are executed in sequence, as is the case in this data, a simple representation of the variables can easily be visualized and is shown in Figure 4.1 below.

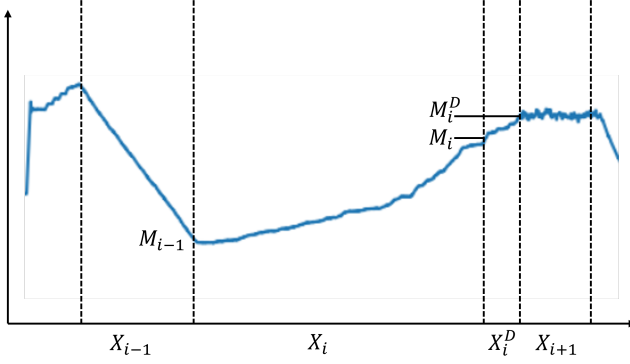


Figure 4.1: Exemplification of the variables X_i , X_i^D , M_i , M_i^D .

4.1 Basic statistics

Initially, we present some basic statistics in Table 4.1 for batches and proceed to discuss the nature of the batches, removing outliers and faulty observations etc.

Cycle	#batch	μ	σ^2	σ	σ/μ
A	66	14.776	3.641	1.908	0.1291
B	64	15.644	3.915	1.979	0.1265
C	61	17.714	2.330	1.526	0.08617
D	60	18.069	6.922	2.631	0.1456
E	60	18.088	9.613	3.100	0.1714
F	63	17.227	7.766	2.787	0.1618

Table 4.1: Per cycle batch duration statistics

Each batch comprises several states. These include adding materials (IDs 1 through 4), centrifugation (ID 5), product transfer (the precipitate generated from the centrifugation, ID 6), chemical reaction (ID 7), a post operation state (Probably to let it cool down to a point where it is ready for further processing, ID 8), Cooling of the product (ID 9), material transfer (transfer the gained product before cleaning of the reaction vessel and/or prepare for the next reaction batch, ID 10). Notice that there is a total of 374 batches throughout the 6 observed cycles.

4.2 Incompleteness on trailing batches

As it may be of interest to investigate the correlation structure of different metrics and variables later on, it is important to understand how each of the batches across the cycles behave. Initially, when looking through the dataset, we observe a few negative phase IDs which will need investigation. However, before we do so, we check that each of the batches actually go through all the states mentioned in [1]. Thus, we take the absolute value of the negative phase IDs to ease the analysis prior to the analysis of the negative phase IDs. After this is done, we observe that not all batches go through all the phases and that some seem to have extra phases not described by [1]. Namely, from Table 4.2, we see that IDs 3 and 4 (which are not described in [1]) have significantly fewer batches going through this phase. But perhaps even more interesting is the final 4 phases where almost all batches goes through these phases.

ID	Count	Description
1.0	374	Addition of liquid raw material Educt1
2.0	374	Addition of liquid raw material Educt2
3.0	181	Addition of liquid raw material Educt3
3.1	374	
3.2	374	Agitation
4.0	163	Waiting for field operation
4.1	374	
4.2	374	Addition of solids
4.3	374	Waiting for control operator
5.0	374	centrifugation
6.0	374	Product transfer
7.0	370	Reaction
8.0	369	Post reaction
9.0	369	Cooling
10.0	368	Material transfer

Table 4.2: The number of batches across all cycles that contains at least one observation for each different absolute phase ID.

Investigating when these inadequacies occur reveals that they are the last batch from each of the cycles. For example, the final batch from cycle A only goes to phase 6 (the product transfer). This can however be explained from the fact that simulation only last for 1100 hours for each cycle and is thus simply cut-off here. As we do not know if these final operations were done at the time the simulations were cut off (which is likely not the case), the final phase for each of the final batches should be disregarded. The cut-off can also be observed in Figure 4.2. Furthermore, throwing away 6 incomplete batches out of the total

374 will likely not harm the analysis and is thus thrown away as this will make the analysis much simpler later on.

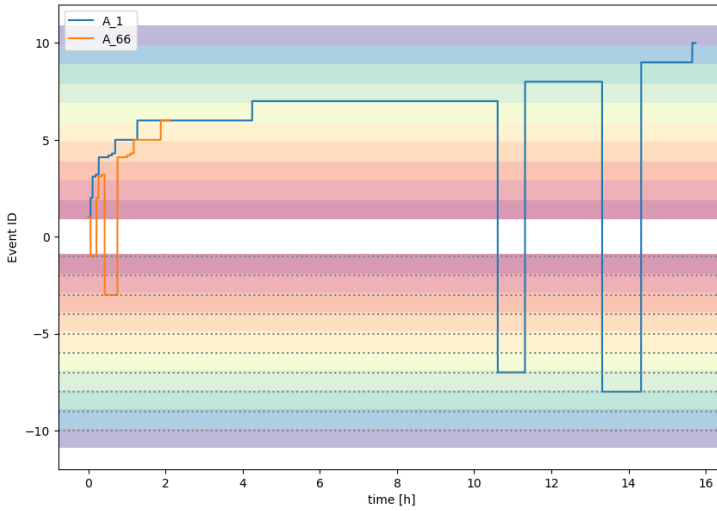


Figure 4.2: The first and last batch from cycle A. It is clear that the final batch is cut-off even before the current phase it finished.

After cutting of the final 6 batches, we have a total of 368 batches of which each goes through all the phases. We thus proceed to discuss the negative phase IDs in the following section, where we also discuss the first four phases.

4.3 Production phases

This part of the process corresponds to events tagged with ID 1 through 10 but will initially concern itself with ID 1 through 4 as much can be learned from the data set here. In Figure 4.3 an example of how the process evolves over time through the different phases is shown. Immediately, we observe something weird, namely the negative event IDs.

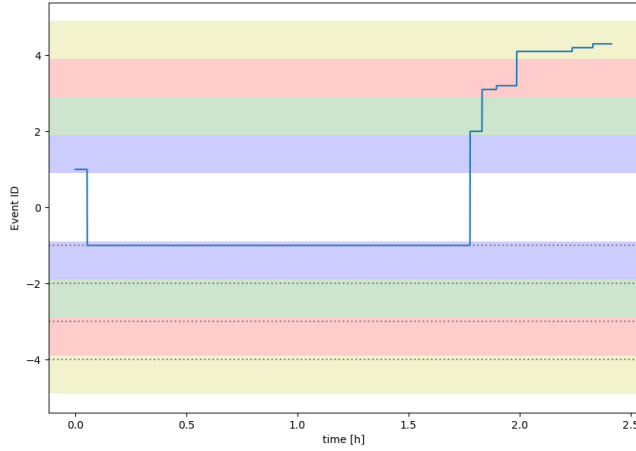


Figure 4.3

To see what is going on here, from data we can see that negative values occur throughout all the six cycles. More specifically, for each negative phase observed, we log in which cycles this occurs. The result is shown in Table 4.3. Notice that -4.1, -4.2 and -4.3 only show up in cycle F, which from [1] is known to be the only one with wrongly labelled phases. We thus suspect that this is indeed the case for these labels and might just have supposed to be the original 4.1, 4.2 and 4.3. To see if nothing funny goes on with these values, these batches are plotted as in Figure 4.3 in Figure 4.4.

Figure 4.4 shows that nothing weird is going on except for the negation of the sub phase's ID. The same can be said for the remaining of the cases where phase ID -4.1, -4.2 and/or -4.3 is used. We thus conclude that these may simply be wrongly labelled thus we convert every such instance to its absolute value and continue with this modified data set from this point on.

Event \ Cycle						
	A	B	C	D	E	F
-1						
-2						
-3						
-4						
-4.1						
-4.2						
-4.3						
-5						
-6						
-7						
-8						
-9						
-10						

Table 4.3: Occurrences of negative phases IDs. It is observed that sub phases 4.1, 4.2, 4.3 only occur in cycle F which is known to be the only cycle with wrongly labelled phases.

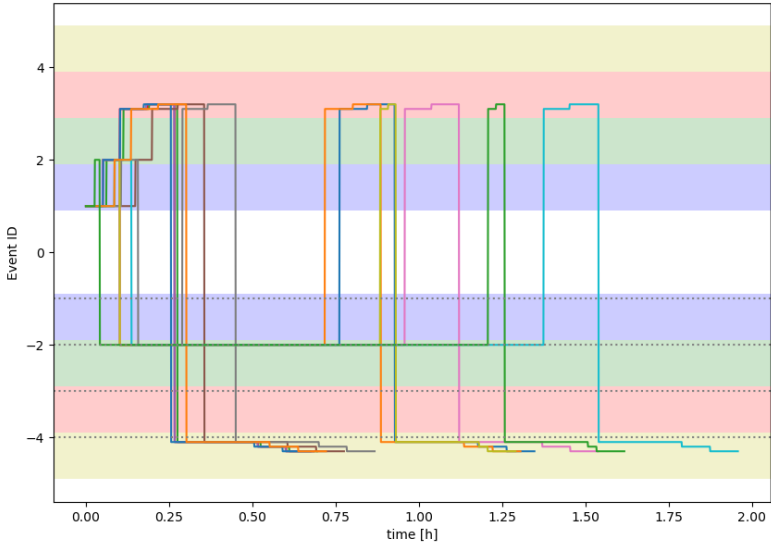


Figure 4.4: 13 of the 48 batches with at least one of the sub phases 4.1, 4.2 4.3 negative.

Having converted the above sub phase IDs we summarize the current situation in regard to negative phase IDs in the following table, Table 4.4. Now all the remaining occurrences of negative phase IDS does not seem to exhibit any structure from looking at Table 4.4. We thus proceed to understand what is going on with the remaining negative phase IDs.

Event \ Cycle	A	B	C	D	E	F
-1						
-2						
-3						
-4						
-5						
-6						
-7						
-8						
-9						
-10						

Table 4.4: Occurrences of negative phases IDs. It is observed that sub phases 4.1, 4.2, 4.3 only occur in cycle F which is known to be the only cycle with wrongly labelled phases.

When plotting different batches, it is clear that the negative phase IDs only occur at the end of a phase e.g. -1 only happens after 1 and so on. This together with the fact that -3 and -4 also only happen after 3.1, 3.2 and 4.1, 4.2, 4.3 respectively (and we never see a phase labelled 3 or 4) indicate that the negative phase IDs could very well correspond to delays at the end of a phase, which both makes sense from a production point of view but also from [1] where they note that all simulated cycles have been implemented with delays.

At this point, it would seem that the labels of the processes are understood for phases 1 through 10 corresponding to the actual production in each batch. Thus, we proceed by searching relationships and otherwise quantifying the durations of each phase, both delays and duration for each of the phases. As a beginning, histograms for each of the phases and delays are plotted in Figure 4.5. Notice that phases 4.1, 4.3 and 8 are not shown, this is because they always last 15 min, 5 min and 2 hours respectively with the only derivation being in machine precision either when loaded or during calculations. Furthermore, notice that for the negative IDs i.e. the delays, the orange bar. This bar represents the cases where no delay was observed which is thus modelled as an atom at 1.

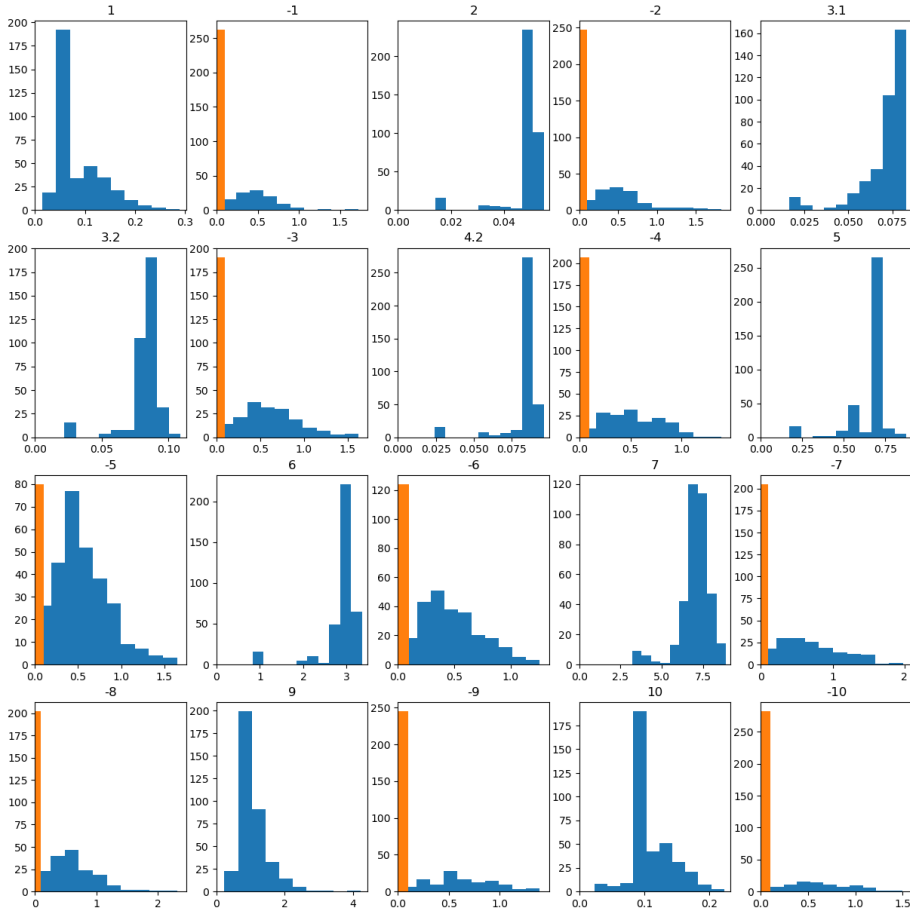


Figure 4.5: Histograms of all phases and delays which are non-constant.

Apart from the above comments, not much catches the eye when looking at Figure 4.5, and we thus proceed by checking if any correlation is immediately present.

4.3.1 Correlations

Lige en korrelationsmatrix

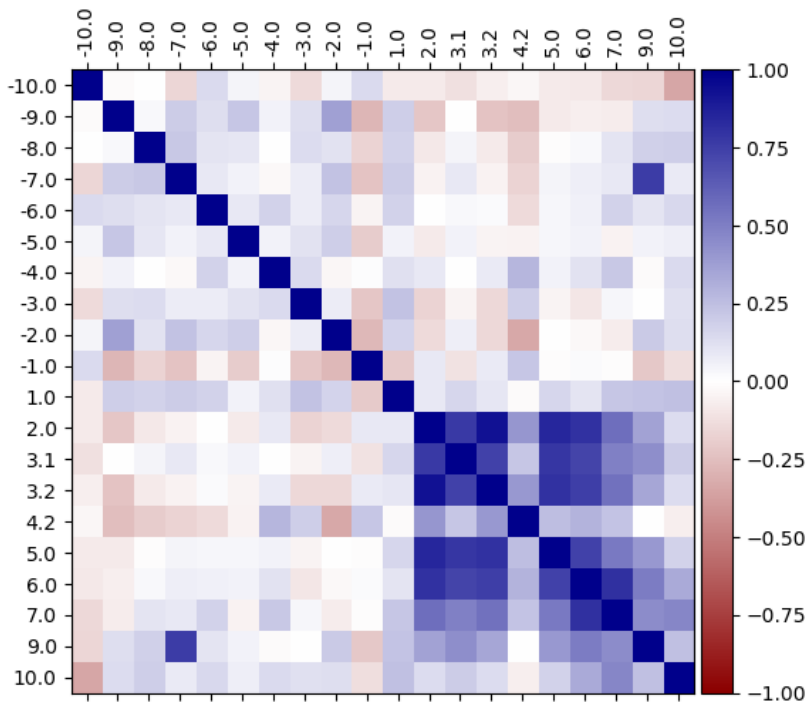


Figure 4.6: Correlation matrix for all phases with non-constant duration.

9 og 10 er ikke specielt korreleret med noget (afkøling og materiale overførsel). Ellers er 2 fremt il og med reaktionen alle korrelerede med hinanden. Eftersom rent fysisk det udvikles i tid, må handlingen i 2 påvirke de næste osv.

Umiddelbart lidt spøjst hvis delay på 7 (reaktion) skulle have noget med tiden for afkøling at gøre, især at den skulle være positiv (ville man ikke tro delay efter produktion ville afkøle mere og dermed reducere behov for afkøling, medmindre varmt steam bliver tilføjet også under delay på 7)

Herunder er samme korrelationsmatrix, dog hvor delay og phases varighed lagt sammen (også med sub phases såsom 3.1, 3.2 og -3 tilsammen bliver 3)

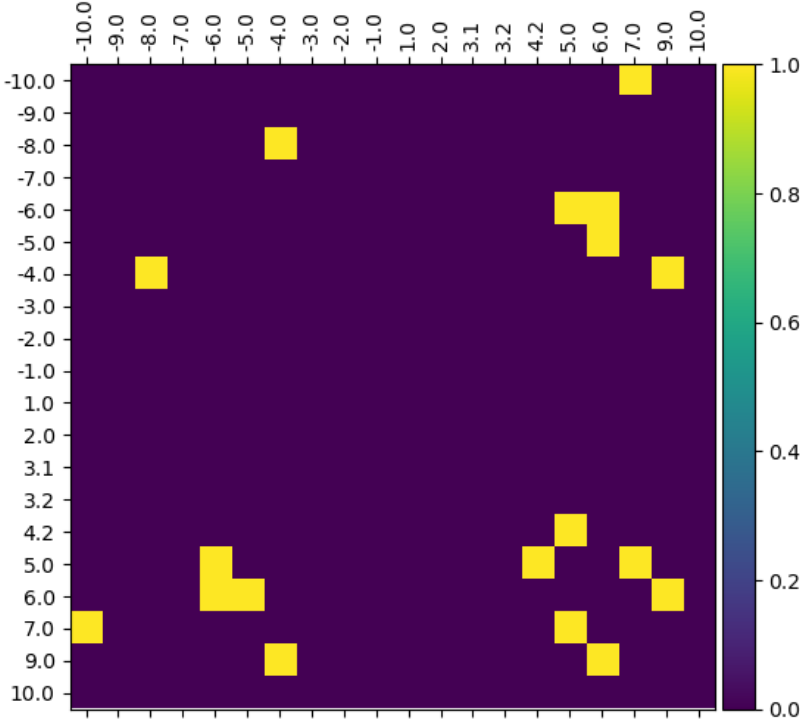


Figure 4.7: Permuteringstest med $\alpha = 0.05$. Also run with less simulations but same result at 1 mil and 10k sims. The Benjamin-Hochberg procedure on the upper (or lower) triangle reveals that none of the correlations are significant.

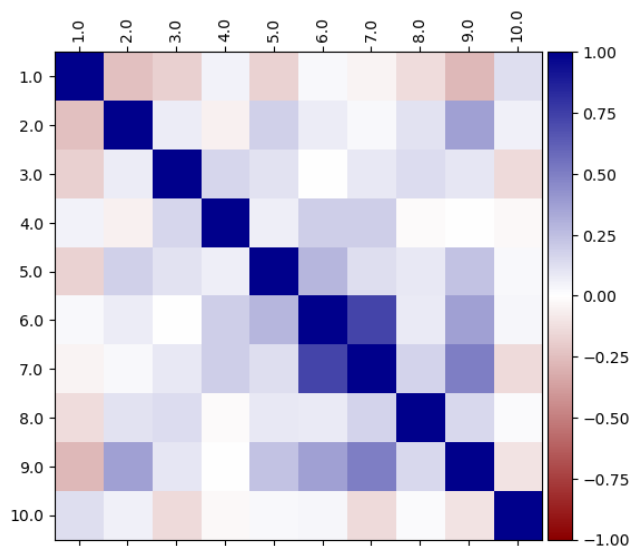


Figure 4.8: Correlation matrix for all phases collapsed

Ligeledes scatter plots for superdiagonalen i ovenstående matrix. Altså phase 1 overfor phase 2, phase 2 overfor phase 3 osv. Er farvelagt efter hvilken cycle de kommer fra. Table 4.4 forklarer hvorfor nogle af de horisontale fremkommer sammen med Figure 4.5 (selve produktionstiden er ret kort sammenlignet med delay.)

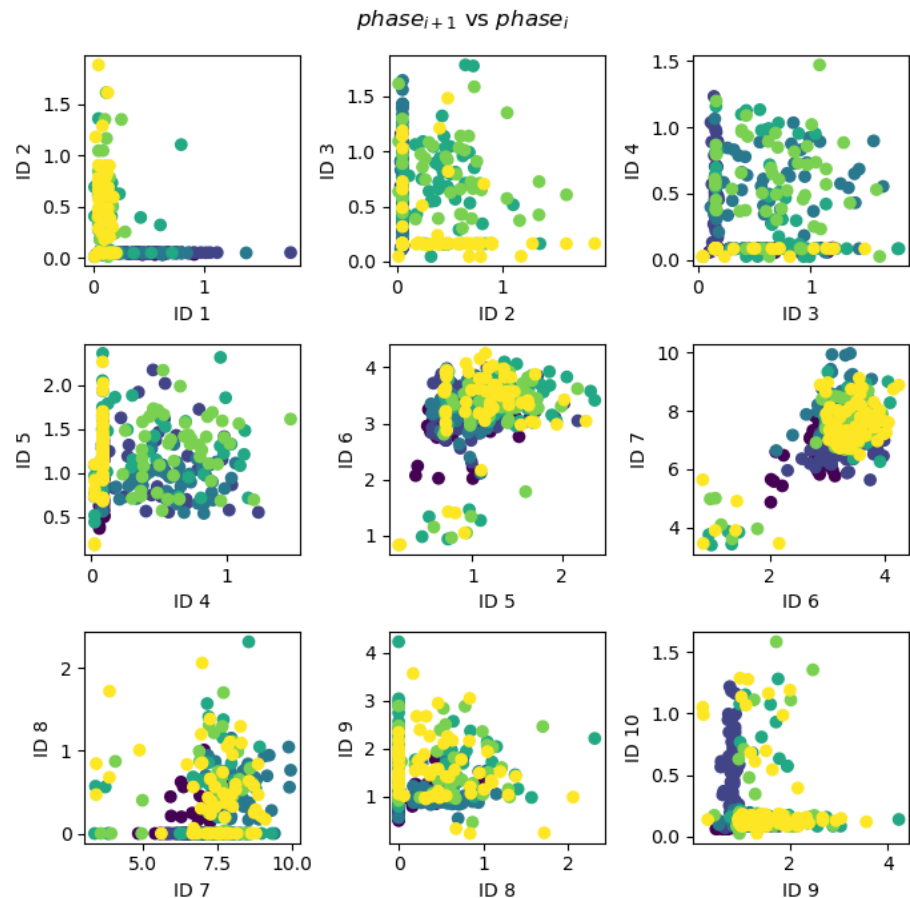


Figure 4.9: Phases vs their next phase when collecting everything regarding a single phase into a total time duration

4.4 Cleaning operations

Sometimes, the vessel is cleansed. This is however not every time after a batch so might be interesting to investigate further. Initially, per cycle, the cleanings are summarized in the following table with basic statistics. As can be seen, there is quite some differences.

The most notifiable differences per batch are the number of cleanses especially when comparing to Table 4.1. For the first two cycles, the cleanses seem to be in between every batch, which is indeed also the while the later four are only sometimes. Furthermore, although the cleanses are between every batch for cycles A and B, the variances are extremely different. For the last four cycles, they seem to be grouped further, E and F are very alike while cleanses in C and D are generally longer although D has a substantially smaller variance than C.

Cycle	#ops	min	max	μ	σ^2	σ	σ/μ
A	65	1.113	3.067	1.917	0.269	0.518	0.270
B	63	1.324	1.751	1.566	0.00883	0.0939	0.0600
C	9	1.544	3.306	2.153	0.277	0.526	0.245
D	10	1.474	2.009	1.581	0.0212	0.146	0.0922
E	10	0.827	1.584	1.465	0.0462	0.215	0.147
F	10	0.748	1.610	1.466	0.0595	0.244	0.166

Table 4.5: Per cycle cleansing statistics

Cycle	A	B	C	D	E	F
$\mathbb{E}[\sum X_u]$	13.993	13.898	15.343	14.471	14.589	14.418
$\text{Var}(\sum X_u)$	0.95636	0.46587	0.76111	4.9589	4.2678	5.3545
$\sum \text{Var}(X_u)$	0.50590	0.31182	0.36667	1.8322	1.5788	1.9696
$\mathbb{E}[\sum X_u^D]$	0.96398	1.9402	2.4503	3.6050	3.7390	3.0041
$\text{Var}(\sum X_u^D)$	0.31843	0.39117	0.90187	1.2468	1.2787	1.0462
$\sum \text{Var}(X_u^D)$	0.34921	0.53198	0.74914	1.4357	1.2454	1.3099
$\mathbb{E}[\sum X_u X_u^D]$	1.9321	1.5001	4.8191	6.4225	6.0405	6.3343
$\text{Var}(\sum X_u X_u^D)$	3.7798	0.89920	16.870	22.133	12.660	16.194

Table 4.6: Each of the time related variables X_i and D_i and variance description.

To verify these observations and potentially discovering more important facts of their probability distributions, histograms are plotted in the following Figure 4.10. We indeed again observe the likeliness between the cycles A and B, C

and D, E and F respectively. Also, for the first two cycles and more so cycle B, the cleaning times are somewhat normally distributed although cycle A has a very heavy right tail in that case. The later four cycles only have 10 observations but the mode (i.e. peak) seem to be about the same.

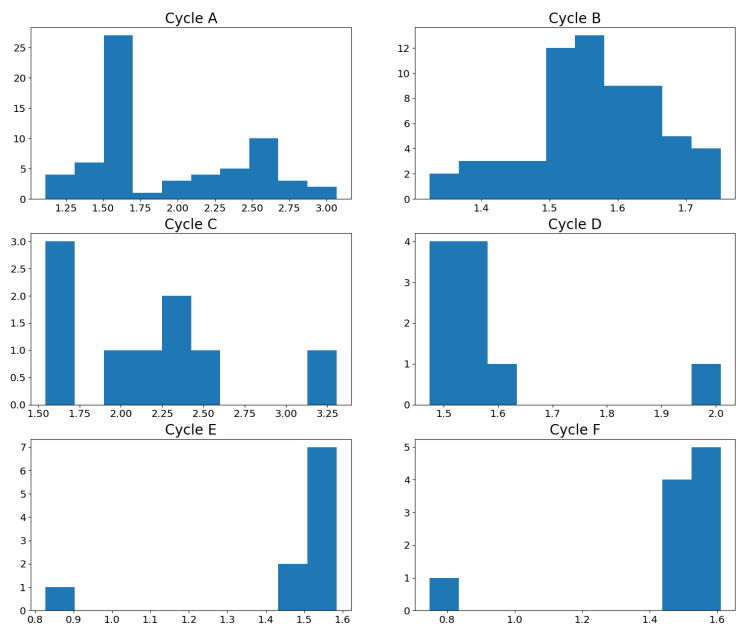


Figure 4.10: Each of the 6 cycles, cleaning operations histograms.

From the above observation of like modes one may want to observe the histogram of the combined set of cleaning times. In particular, under the hypothesis that the durations are actually from the same probability distributions and realized independently within each cycle a histogram of all the observations are of interest and is shown in Figure 4.11 below.

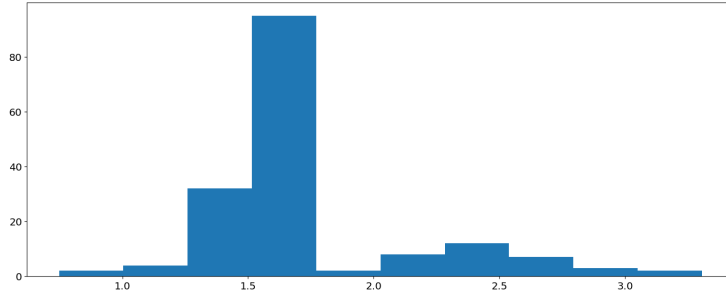


Figure 4.11: Combined cleaning operations histograms.

Finally, to get a better overview of the irregularities in the number of cleaning periods (mostly concerning cycles C through F), each cleaning operation is shown in the following Figure 4.12. The vertical shaded rectangles signify the period in which a cleaning operation is taking place. Furthermore, the event IDs are shown but to get a clearer view on what is going on, a single rectangle (zoomed in) is shown in Figure 4.13.

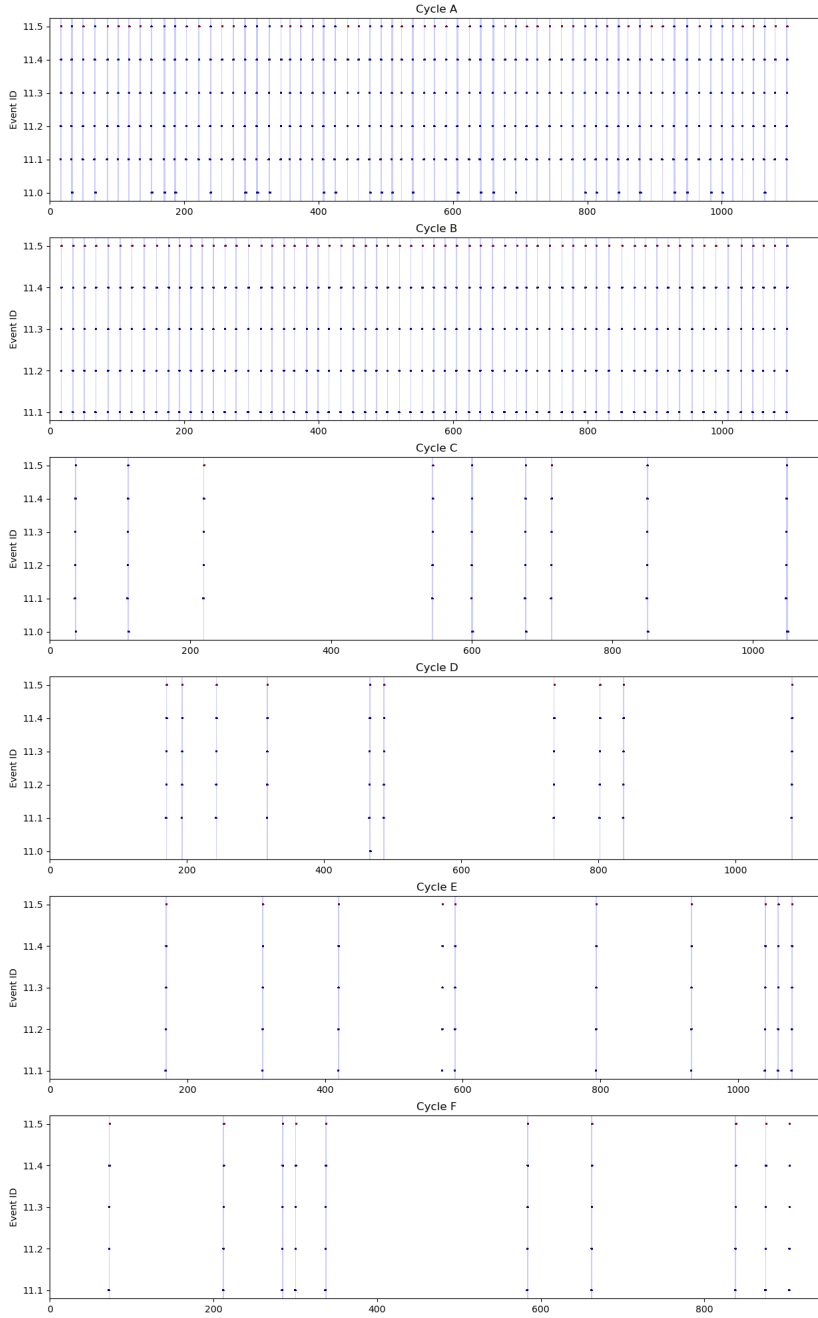


Figure 4.12: Each of the 6 cycles, cleaning (corresponding to `BatchID = 0`). Each (Cleaning Procedure), CIP, is highlighted with an opaque interval (the blue rectangles). The dots marked with red (only ID 11.5, but not all of these are red), is if the Cleaning ID is 0.

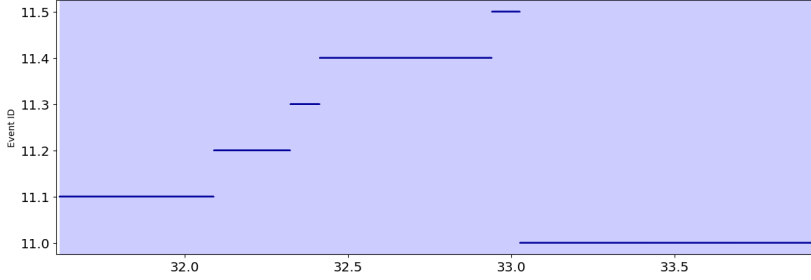


Figure 4.13: A single blue rectangle zoomed in

It is observed that the observations marked with red in figure 4.12 occur exactly when that specific cleaning operation does not go to the state 11.0 after the flush of the tank (event ID 11.5) and vice versa. It is hard to conclude what this may mean, but the cleaning being in state 11.0 may indicate that the system is idle before continuing the next batch like what is observed from the other steps of the process flow. Also, it is noted that while the red dots occur nothing else is happening according to the dataset.

From a modelling point of view, the cycles C through F can be thought of as the cleansing operation having a probability of not happening or equivalently as having a duration of 0. It is thus of interest to observe what the probability of cleaning after an operation is. From Table 4.1 and Table 4.5, we that indeed for cycles A and B, the probability is 100 % when disregarding the possibility of cleaning after the final batch. Hence, we see that for the remaining cycles, the probabilities of cleaning the tank after an operation are as in Table 4.7

Cycle	% cleaning
A	100.00
B	100.00
C	15.00
D	16.95
E	16.95
F	16.13

Table 4.7: Per cycle probability of cleaning

Furthermore, let C_i denote whether the i th batch is followed by a cleaning of the tank or not. It is then of interest if the next batch is followed by a cleaning given whether the current batch is followed by a cleaning. In particular, we count for each of the cycles the transitions which are shown in the following tables. Notice that the number of observations is two less than the total number of batches within each specific cycle. This is due to the last batch is never followed by

a cleaning (nor is the first batch superseded by a cleaning procedure) which results in one less observation and also due to the fact that we are logging transitions and hence lose another observation. To test for randomness, a Chi-squared test is carried out on each of the cycles to check for independence. It is observed all the cycles exhibit independence between the groups i.e. there is no statistical evidence for information is gained about if the next batch is followed by a cleaning operation given whether the current batch is followed by a cleaning operation.

$C_i \backslash C_{i+1}$		No	Yes
		41	9
No		9	0
Yes			
(a) C, $p = 0.3293$			
$C_i \backslash C_{i+1}$		No	Yes
		41	7
No		7	3
Yes			
(c) E, $p = 0.3532$			
$C_i \backslash C_{i+1}$		No	Yes
		41	8
No		7	2
Yes			
(b) D, $p = 0.6456$			
$C_i \backslash C_{i+1}$		No	Yes
		41	9
No		9	1
Yes			
(d) F, $p = 1.0000$			

Table 4.8: Contingency table for Cycle C-F

Thus collecting the observations from all the last four cycles, we may want to model the atom of the cleaning procedure independently of the previous batch and with a probability of 0.8375 corresponding to the cleaning procedure only being carried out 16,25% of cases.

Finally, we show the autocorrelation function for each the four cycles C-F in Figure 4.14 and note that all the ACF stay within the 95% confidence interval.

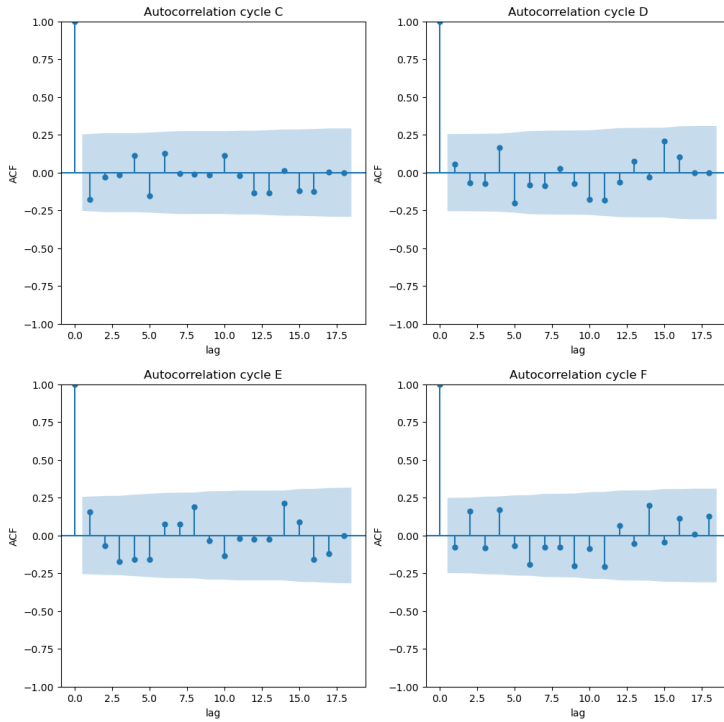


Figure 4.14: Autocorrelation function for each of the final 4 cycles. As can also be seen from this there seem to be no information to be gained of C_i from C_{i-1} .

5.1 Causal discovery

In this section, we shall discuss the method for network deconvolution, originally proposed by [2]. The underlying problem is inferring direct effects and dependencies. From this, using prior information on the production setup, we shall be able to infer causal dependencies by directing the resulting edges from the network deconvolution (ND) algorithm. Particularly, the framework and general algorithm proposed Feizi et al. stems from a graph-theoretic approach to the problem of inferring direct dependencies. Namely, suppose that observations are made of some properties of in this case a chemical process. We shall represent these properties as vertices (nodes) V and dependencies between properties as edges. Initially, when observing the vertices, we observe both direct and indirect effects. Particularly, a vertex v_1 might influence some other vertex v_3 through another vertex v_2 if v_2 depends on v_1 and v_3 of v_2 . In this case, we will observe that v_1 influences v_3 , but actually it is v_2 that has a direct influence on v_3 . In graph-theoretical terms, we thus observe the transitive closure of the information that flows between vertices but want to infer the underlying network structure.

An important note on the algorithm to come is that we only use vertices that we have observed. Namely, the underlying structure might be as in Figure 5.1a

with an unobserved node/variable (named U in this case). However, without any more assumptions or modelling choices we would (ideally) infer the network structure depicted in Figure 5.1b. With these initial comments, we proceed

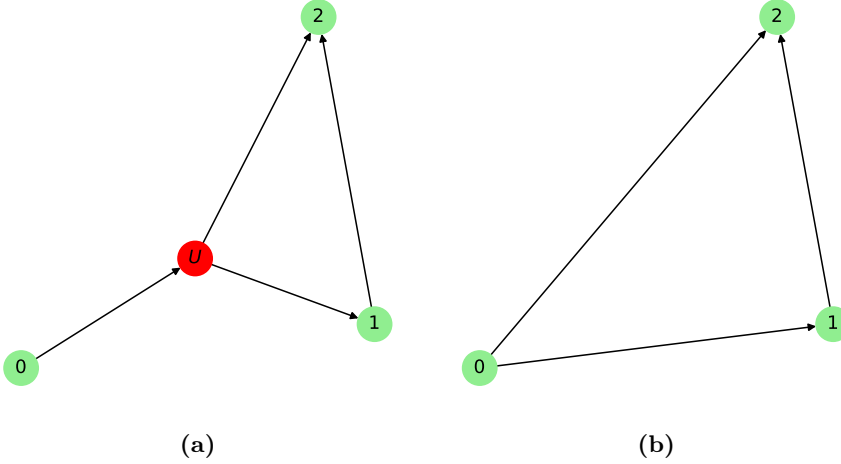


Figure 5.1: (a) An example of a causal structure depicted as a graph. When observing the network, only nodes 0, 1 and 2 are observed/recorded. (b) The resulting inferred graph from observational data. Although this is not a complete picture of the true underlying dynamics of the system, if only the observed variables are of interest, this will be an equally proper representation of the system. Furthermore, in practice this means no further assumptions are made which can and can not be of desire. Namely, if prior information is accessible one might introduce new nodes in the inferred network.

with the general setup and assumptions for network deconvolution based on observations.

5.1.1 Setup and assumptions

Suppose a set of N random variables (X_i) is given. The method presented in this section aims to discover direct relationships between pairs X_i and X_j for $i \neq j$. These relationships will be presented by an undirected graph as in the previous section. In particular, we shall let each random variable X_i be represented by a vertex in a graph. We will later discuss a way of directing edges such that a causal network may be discovered i.e. a directed acyclic graph that may be used for inference.

The method proposed by [2] then works as follows. Given an observed matrix $G_{obs} \in \mathbb{R}^{N \times N}$ of similarities between each pair of variables, we shall deduce a matrix $G_{dir} \in \mathbb{R}^{N \times N}$ of direct similarities between each pair of random variables X_i and X_j . The measure of similarity, can in practice be any desired measure such a correlation, mutual information which we will focus on in this thesis. See Section 5.2 for a further discussion on these two measures. Note that the algorithm presented will in theory work for non-symmetric measures as well such as *Interaction information*, *Directed information* and *Normalized information*.

The (direct) network is then presented by the discovered G_{dir} containing only the direct effects i.e. interaction between pairs of variables which can be viewed as weights on the edges of the complete graph with nodes representing the random variables. As we shall see in section Subsection 5.5.2, the algorithm is somewhat robust to noise in the sense that we can ensure accuracy depending on the level of noise observed present in G_{obs} and on the norm chosen (from a certain, although rather general, set of norms). This hints to that a threshold on the inferred weights on the edges of the network might be a good idea which is further solidified by the facts that often only the most influential variables are of importance when trying to control the process.

The first assumption is that the observed matrix of co-dependence G_{obs} may be expressed as

$$G_{obs} = G_{dir} + G_{indir} \quad (5.1)$$

Namely, that the direct and indirect effects can be added together to get the total and thus observed interdependence between each pair of variables. Often, this is not the case as we shall see later on. However, the error made from this assumption and the ones to be presented seem to be small enough that the discovered network accurately resemble the true underlying network.

The second and final assumption is that the indirect effects G_{indir} can be computed in terms of G_{dir} . Namely, that

$$G_{indir} = G_{dir}^2 + G_{dir}^3 + \dots \quad (5.2)$$

i.e. that the observed *information* exchanged on an edge e_{ij} between nodes X_i and X_j is the sum second, third etc. order effects, each given by the information on the n -path (where n is the order of the (diminishing) indirect effect) again assumed to a sum of products. In other terms, the second order indirect effect between X_i and X_j (given as the (i, j) element of G_{dir}^2) is the sum of products on edges e_{ik} and e_{kj} for all k

$$[G_{dir}^2]_{ij} = \sum_{k=1}^N e_{ik} e_{kj}$$

where e_{ij} is the (i, j) element of G_{dir} . Immediately, we observe that e_{ii} is of interest in terms of its physical meaning. The co-dependence between a random variable and itself might be somewhat ambiguous or even undefined depending on the measure. Thus, the meaning of (non-existing) edges e_{ii} will be of interest later on when using the method on controlled cases to see if any sense can be made of these.

Thus, from the above assumptions, it follows that we can express G_{obs} as

$$G_{obs} = G_{dir} + G_{dir}^2 + G_{dir}^3 + \dots \quad (5.3)$$

Hence, for G_{dir} to exist, it must have spectral radius at most 1 as otherwise, the above sum diverges and thus G_{obs} will not exist. I.e. $\rho(G_{dir}) < 1$. Thus, assuming convergence we can rewrite the infinite series as

$$G_{obs} = G_{dir} (I - G_{dir})^{-1} \quad (5.4)$$

It immediately follows that G_{dir} is given by (can be proved by directly inserting the above expression for G_{obs})

$$G_{dir} = G_{obs} (I + G_{obs})^{-1} \quad (5.5)$$

Furthermore, if the measure of dependence between pairs of variables is symmetric, then so is G_{obs} and hence diagonalizable by some orthogonal matrix U such that $G_{obs} = U \Lambda_{obs} U^T$ (with the columns of U being eigenvectors of G_{obs}). It follows that G_{dir} can be expressed in a simple (and later computationally efficient) way

$$G_{dir} = U \Lambda_{dir} U^T$$

where $\Lambda_{dir} = \Lambda_{obs} (I + \Lambda_{obs})^{-1}$.

We note that from the above one needs $(I + G_{obs})^{-1}$ to be well-defined which is equivalent to $-1 \notin \sigma_{G_{obs}}$ i.e. -1 is not an eigenvalue of G_{obs} . To see that this is indeed the case whenever $\rho(G_{dir}) < 1$, and that $I + G_{obs}$ is thus invertible we use Equation 5.4 and simplify

$$\begin{aligned} I + G_{obs} &= I + G_{dir} (I - G_{dir})^{-1} \\ &= (I - G_{dir}) (I - G_{dir})^{-1} + G_{dir} (I - G_{dir})^{-1} \\ &= (I - G_{dir})^{-1} \end{aligned}$$

which is clearly invertible. Furthermore, we note that under the assumption $\rho(G_{dir}) < 1$ we can not place any bound on the spectral radius of G_{obs} . Namely, if v is a unit eigenvector of G_{dir} with eigenvalue λ such that $|\lambda| < 1$, then v is also an eigenvector of G_{obs} as

$$G_{obs}v = \sum_{k=1}^{\infty} G_{dir}^k v = \sum_{k=1}^{\infty} \lambda^k v = \frac{\lambda}{1 - \lambda} v$$

i.e. $\left(\frac{\lambda}{1-\lambda}, v\right)$ is an eigenpair of G_{obs} and since $\frac{\lambda}{1-\lambda} \in (-1/2, \infty)$ for $\lambda \in (-1, 1)$ we can in general not bound the spectral radius of G_{obs} , although we should never observe an eigenvalue equal to or below $-1/2$ (which again proves that -1 is not an eigenvalue of G_{obs}).

Now, before discussing the implementation and analyzing the algorithm both analytically and through examples, we will take a closer look at the similarity measures that are to be used with this method and that in the end will make up the matrix G_{obs} . Namely, *mutual information* and *correlation*.

5.2 Information measures and computation

In this section we discuss two measures that can be used to construct the matrices of codependency from the previous section. Namely, we shall touch on correlation and discuss what one might choose to call Copula-based entropy. However, before discussing Copula entropy (CE) we first need to define what a copula is.

5.2.1 Copula

Given a set of N random variables X_1, \dots, X_N , a copula is loosely speaking a distribution function with support $[0, 1]^N$ incorporating the dependence structure between the random variables. Given a joint probability function F and (invertible) marginals F_1, \dots, F_N we define a copula C as

$$F(x_1, \dots, x_N) = \mathbb{P}(X_1 \leq x_1, \dots, X_N \leq x_N) = C(F_1(x_1), \dots, F_N(x_N))$$

Letting $u_i = F_i(x_i) \in [0, 1]$ it is clear that C is a distribution function as described above [3]. Furthermore, it follows that the marginals of C are uniform. We thus define a copula in probabilistic terms as

Definition 5.1 (Copula). *A function $C : [0, 1]^d \rightarrow [0, 1]$ is called a copula if it has uniform marginals and is a probability function for a d -dimensional random vector \mathbf{X} .*

An important and fundamental theorem of copulas for continuous random variables where the marginals are also continuous functions is stated by Sklar:

Theorem 5.2 (Sklar's theorem). *For a random vector \mathbf{X} with CDF F and univariate marginal CDFs F_1, \dots, F_d . There exists a copula C such that*

$$F(x_1, \dots, x_d) = C(F_1(x_1), \dots, F_d(x_d)) \quad (5.6)$$

If X is continuous, C is unique.

Hvis defineret på diskrete punkter i stedet for kontinuert interval er den også unik for diskrete variable?

From Equation 5.6 we see that a copula is thus simply just a function that *couples* the marginals of a random vector to the joint distribution. The following corollary follows immediately

Corollary 5.2.1 (Coordinate transformation). *Under the assumptions of Theorem 5.2, given any set (T_1, \dots, T_d) of strictly increasing functions, if C is a copula of (X_1, \dots, X_d) then it is also a copula of $(T_1(X_1), \dots, T_d(X_d))$.*

Proof. Suppose (X_1, \dots, X_d) permits a copula C and let T_i be given as stated. Consider coordinate wise the result of the transformation $Y_i = T_i(X_i)$ and consider the CDF $F_{Y_i}(y_i)$

$$F_{Y_i}(y_i) = \mathbb{P}(Y_i \leq y_i) = \mathbb{P}(T_i^{-1}(Y_i) \leq T_i^{-1}(y_i)) = \mathbb{P}(X_i \leq x_i) = F_{X_i}(x_i)$$

Thus

$$\begin{aligned} F_{\mathbf{X}}(x_1, \dots, x_d) &= C(F_{X_1}(x_1), \dots, F_{X_d}(x_d)) \\ &= C(F_{Y_1}(y_1), \dots, F_{Y_d}(y_d)) \\ &= F_{\mathbf{Y}}(y_1, \dots, y_d) \end{aligned}$$

where Sklar's theorem have been used for the final equality. \square

The above corollary is actually equivalent with a seemingly stronger statement and follows easily

Proposition 5.3. *Since T_i is strictly increasing, the inverse T_i^{-1} exists and is also strictly increasing. Thus, the above implication is bidirectional and hence for strictly increasing functions T_i , C is a copula of (X_1, \dots, X_d) if and only if it is a copula of $(T_1(X_1), \dots, T_d(X_d))$.*

tilføj remark om hvorfor copula ikke er veldefineret for diskrete fordelinger (henvis til kilde med eksempel. Ret fint eksempel der viser at copula ikke er unik for diskrete fordelinger (Hvertfald set over reelle intervaller. Men hvad med hvis def mængden er diskret??))

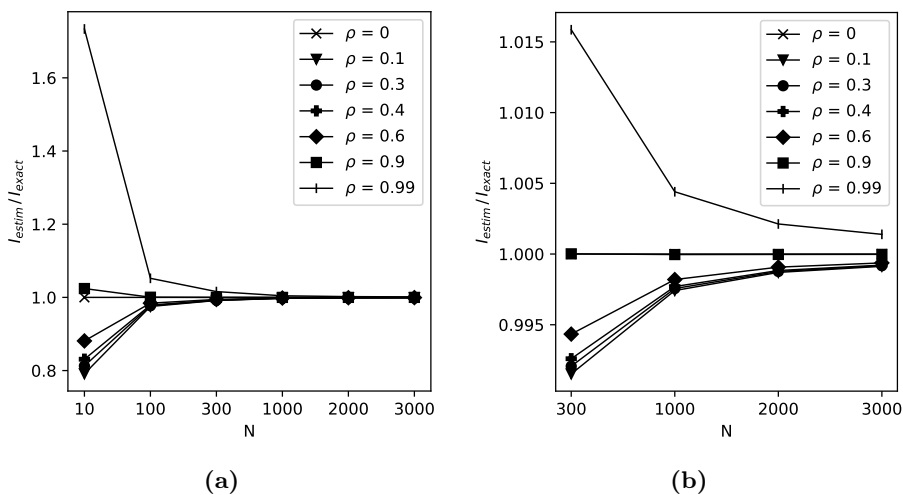


Figure 5.2: Evaluation of MI for new method for different N . Bør sammenlignes med artiekl fundet (har sat i bibtex) og original papers (ikke Kina)

Inkluder flere eksempler end blot gaussian as done by [4]

5.2.2 Mutual information and Copula entropy

In this section we introduce Copula entropy as done in [5]. The name comes from the general definition of (differential) entropy. Namely, for continuous variables X_i and X_j , we define the joint (differential) entropy as

$$h(X_i, X_j) = \int \int_{\mathbb{R}^2} f(x_i, x_j) \log_b(f(x_i, x_j)) \, dx_i, dx_j$$

where the base b , initially, can be chosen at will as all logarithms are equal up to a scalar multiple of each other.

forskellige baser og egentlig ikke et rigtigt valg. Dog er det forskellig skallering af G obs som vi senere viser har en effekt på højere ordens led. Kommenter på/henvis til dette i senere afsnit.

However, to make the calculations more robust and efficient we will use a closely related measure of dependence, namely CE which is defined as follows

$$CE(X_1, \dots, X_n) = - \int \dots \int_{[0,1]^n} c(u_1, \dots, u_n) \log_b c(u_1, \dots, u_n) du_1 \dots du_n$$

where $c(\cdot)$ is the uniquely defined copula of the joint distribution $f_{\mathbf{X}}(\cdot)$. We show that the above is indeed equal to the negative mutual information. First, we realize that from definition,

$$\begin{aligned} c(u_1, \dots, u_n) &= \partial_{\mathbf{u}} C(u_1, \dots, u_n) \\ &= \partial_{\mathbf{u}} F(F_1^{-1}(u_1), \dots, F_n^{-1}(u_n)) \\ &= f(x_1, \dots, x_n) \frac{1}{f_1(x_1) \dots f_n(x_n)} \end{aligned}$$

Thus

$$\begin{aligned} -CE &= \int \dots \int_{[0,1]^n} c(u_1, \dots, u_n) \log c(u_1, \dots, u_n) du_1, \dots, du_n \\ &= \int \dots \int_{[0,1]^n} c(u_1, \dots, u_n) \log c(u_1, \dots, u_n) dF_1(x_1), \dots, dF_n(x_n) \\ &= \int \dots \int_{\mathbb{R}^n} \frac{f(x_1, \dots, x_n)}{f_1(x_1) \dots f_n(x_n)} \log \left(\frac{f(x_1, \dots, x_n)}{f_1(x_1) \dots f_n(x_n)} \right) f_1(x_1) \dots f_n(x_n) dx_1 \dots dx_n \\ &= \int \dots \int_{\mathbb{R}^n} f(x_1, \dots, x_n) \log \left(\frac{f(x_1, \dots, x_n)}{f_1(x_1) \dots f_n(x_n)} \right) dx_1 \dots dx_n \\ &= MI \end{aligned}$$

At this point, we only need G_{obs} and as mentioned earlier we will use mutual information. [ref til afsnit om CE](#) Hence, we obtain G_{obs} from the pairwise copula entropy (CE)

$$G_{obs} = \begin{bmatrix} 0 & -CE_{12} & \dots & -CE_{1n} \\ -CE_{21} & 0 & \dots & -CE_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ -CE_{n1} & -CE_{n2} & \dots & 0 \end{bmatrix}$$

Although theoretically $-NCE_{ii} = \infty$ for all i , we put 0 in the diagonal because we do not want self explanation as these are trivial. The argument for calculating CE instead of MI are due to the finite volume integral and simpler integrand. In particular, using the copulas, we avoid the fraction $\frac{f(x_1, \dots, x_n)}{f_1(x_1) \dots f_n(x_n)}$ which could easily result in numerical instability e.g. when both f and f_i s are close to 0.

Finally, from the deconvoluted information matrix D_{dir} we may choose a threshold t for choosing which edges are significant. The choice of t

5.2.2.1 Entropy and mutual information in the limit

Originally, entropy H is only defined for discrete variables. A closely related but yet very different measure for continuous variables is differential entropy h defined as

$$h(X) = \int_{\mathbb{R}} f(x) \log f(x) dx$$

Although one may think of this as the limit of (discrete) entropy, this is not the case. Namely, consider the support of $f(x)$ (here assumed to be the entire real line) binned into intervals i.e. a discretization of the continuous random variable X . To make notation simpler, we shall bin into equal-sized intervals of width Δ . Then, for each interval $[i\Delta, (i+1)\Delta)$ for $i \in \mathbb{Z}$, there exists an x_i such that the probability mass on this interval is represented by this x_i :

$$f(x_i)\Delta = \int_{i\Delta}^{(i+1)\Delta} f(x) dx$$

Clearly, this discretization is a valid distribution as

$$\sum_{i \in \mathbb{Z}} f(x_i)\Delta = \int_{\mathbb{R}} f(x) dx = 1$$

and in the limit, as $\Delta \rightarrow 0$ we recover the original distribution $f(x)$. However, if we try to calculate the entropy of this discretization, denoted by H^Δ , we get

a diverging limit

$$\begin{aligned}
 H^\Delta &= \sum_{i \in \mathbb{Z}} f(x_i) \Delta \log f(x_i) \Delta \\
 &= \sum_{i \in \mathbb{Z}} f(x_i) \Delta \log f(x_i) + \sum_{i \in \mathbb{Z}} f(x_i) \Delta \log \Delta \\
 &= \sum_{i \in \mathbb{Z}} f(x_i) \Delta \log f(x_i) + \log \Delta
 \end{aligned}$$

Clearly, the first summand in the above expression converges to the differential entropy as $\Delta \rightarrow 0$ whereas $\log \Delta \rightarrow -\infty$.

A similar argument for the joint entropy between the discretization of X_1 and X_2 , denoted by H_{12}^Δ , results in

$$H_{12}^\Delta = \sum_{i,j \in \mathbb{Z}} f(x_1^{(i)}, x_2^{(j)}) \Delta_1 \Delta_2 \log f(x_1^{(i)}, x_2^{(j)}) + \log \Delta_1 + \log \Delta_2$$

where $x_1^{(i)} \in [i\Delta_1, (i+1)\Delta_1)$ and $x_2^{(j)} \in [j\Delta_2, (j+1)\Delta_2)$ are defined such that

$$f(x_1^{(i)}, x_2^{(j)}) \Delta_1 \Delta_2 = \int_{j\Delta_2}^{(j+1)\Delta_2} \int_{i\Delta_1}^{(i+1)\Delta_1} f(x_1, x_2) dx_1 dx_2, \quad \forall i, j \in \mathbb{Z}$$

Note that clearly $(x_1^{(i)}, x_2^{(j)})$ exists for all $i, j \in \mathbb{Z}$. Again, the joint entropy diverges however, when computing the mutual information, we see that the diverging terms cancel. Namely, from [reference til information ud fra entropy proposition](#)

$$\begin{aligned}
 I_{12}^\Delta &= H_1^\Delta + H_2^\Delta - H_{12}^\Delta \\
 &= \sum_{i \in \mathbb{Z}} f_1(\tilde{x}_1^{(i)}) \Delta_1 \log f_1(\tilde{x}_1^{(i)}) + \log \Delta_1 \\
 &\quad + \sum_{j \in \mathbb{Z}} f_2(\tilde{x}_2^{(j)}) \Delta_2 \log f_2(\tilde{x}_2^{(j)}) + \log \Delta_2 \\
 &\quad - \sum_{i,j \in \mathbb{Z}} f(x_1^{(i)}, x_2^{(j)}) \Delta_1 \Delta_2 \log f(x_1^{(i)}, x_2^{(j)}) - \log \Delta_1 \Delta_2 \\
 &\rightarrow h(X_1) + h(X_2) - h(X_1, X_2) \text{ as } \Delta_1, \Delta_2 \rightarrow 0
 \end{aligned}$$

Thus, the limit of the mutual information for discrete random variables is indeed the mutual information defined for continuous random variables and can be computed either as the limit of discretizing the probability density function and then computing entropies or just using the initial definition for mutual information [definition til mutual information](#).

5.2.3 Correlation

introducer correlation hurtigt og kommenter på at det kan bruges som codependence mål og derefter ikke kan regnes ud fra copula

In this section we show that in general, the copula does imply correlation. Namely, given a copula C for some set $\{X_i\}_{i \in I}$ indexed by I , one can not calculate ρ between any pair (X_i, X_j) , $i \neq j$. This is easily shown by the following argument.

First, note that from Corollary 5.2.1, C is also a copula for $Z_i := (X_i - \mu_i) / \sigma_i$ for $i \in I$ where $\mu_i = \mathbb{E}[X_i]$ and $\sigma_i = \sqrt{\text{Var } X_i}$. Clearly, the correlation coefficient for Z_i and Z_j is the same as between X_i and X_j . We thus proceed trying to calculate the correlation between any pair Z_i and Z_j .

$$\begin{aligned} \rho_{ij} &= \int \int_{\mathbb{R}^2} z_i z_j f_{ij}(z_i, z_j) dz_i dz_j \\ &= \int \int_{[0,1]^2} F_i^{-1}(u_i) F_j^{-1}(u_j) c_{ij}(u_i, u_j) du_i du_j \end{aligned}$$

where c_{ij} density version of the copula defined for X_i and X_j and F_i and F_j are the marginals of Z_i and Z_j with mean 0 and variance 1. From the above, it is then clear for a fixed, non-constant copula C , the correlation depends on the marginals of X_i and X_j . Also, we see that a constant copula density (only admissible if $c \equiv 1$ on $[0, 1]^2$ and 0 elsewhere) always results in $\rho_{ij} = 0$ as

$$\int_0^1 F^{-1}(u) du = \int_{\mathbb{R}} z f(z) dz = 0$$

again, under the assumption that Z_i has mean 0.

Thus, we conclude that indeed mutual information and correlation is very different measures of codependency and that it does not make much sense to introduce copulas in the setting of correlation.

5.3 Copula based network discovery

5.4 Algorithms

Algorithm 1 G_{obs} computation

Require: $N > 0$ ▷ Number of variables
for $1 \leq i < j \leq N$ **do**
 Estimate F_i and F_j from $x_i^{\mathcal{D}}$ and $x_j^{\mathcal{D}}$
 $u_i^{\mathcal{D}} \leftarrow F_i(x_i^{\mathcal{D}})$
 $u_j^{\mathcal{D}} \leftarrow F_j(x_j^{\mathcal{D}})$
 Estimate C_{ij} from $u_i^{\mathcal{D}}$ and $u_j^{\mathcal{D}}$
 Compute NCE_{ij}
 $G_{ij}, G_{ji} \leftarrow -NCE_{ij}$
end for

Algorithm 2 (ND) Network Deconvolution

Require: G_{obs} ▷ Input observational matrix
 $[G_{obs}]_{ii} \leftarrow 0, \forall 1 \leq i \leq N$ ▷ zero-diagonal
 $Q_p \leftarrow G_{[1-\alpha]}$
 Set $G_{obs} = 0$, where $G_{obs} < Q_p$
 Compute eigendecomposition Q, Λ of G_{obs}
 $\lambda^+ \leftarrow \max(\lambda^{\max}, 0)$
 $\lambda^- \leftarrow -\min(\lambda^{\min}, 0)$
 $m^+ \leftarrow \frac{1-\beta}{\beta} \lambda^+$
 $m^- \leftarrow \frac{1+\beta}{\beta} \lambda^-$
 $m \leftarrow \max(m^+, m^-)$
 $\hat{\Lambda} \leftarrow \Lambda (mI + \Lambda)^{-1}$
return $Q\hat{\Lambda}Q^T$

Remark. The $1 - \alpha$ quantile, denoted by $G_{[1-\alpha]}$ (of the upper triangular matrix), can be computed in many ways. As it is only used to filter the G_{obs} matrix, its precise value does not matter. Only the property α part of the observations are below $Q_{1-\alpha}$ and $1 - \alpha$ are above (or equal to) $Q_{1-\alpha}$. This property is for example fulfilled by the quantile function `quantile` from NumPy (v. 1.26.4). Thus setting $\alpha = 1$ will result in G_{obs} retaining all entries (except for the diagonal entries).

Remark. The $\beta \in (0, 1)$ parameter serves as a sort of regularization. The algorithm above also maps the maximum absolute value of the eigenvalues (the spectral radius) to β as is also pointed out in the implementation of [Source code from Nature paper at https://compbio.mit.edu/nd/index.html](https://compbio.mit.edu/nd/index.html). The proof of this is shown below

Proof. To show that eigenvalues i.e. the diagonal elements of Λ resulting from Algorithm 2 all fall in the interval $[-\beta, \beta]$ (i.e. $\sigma(Q\Lambda Q^T) \subseteq [-\beta, \beta]$) where at least one λ is mapped to either $-\beta$ or β , first notice that clearly the resulting eigenvalues of $G_{dir} = Q\hat{\Lambda}Q^T$ are clearly given by $\frac{\lambda_i}{m+\lambda_i}$ where $(\lambda_i)_{\{1,\dots,N\}}$ are the (real) eigenvalues of G_{obs} from the definition of $\hat{\Lambda}$. We will show the above by first considering $\lambda \geq 0$ and $\lambda < 0$.

For $\lambda \geq 0$, clearly $m \geq \frac{1-\beta}{\beta}\lambda^+$, thus

$$\frac{\lambda}{m+\lambda} = \frac{1}{1+m/\lambda} \leq \frac{1}{1+\frac{\lambda^+}{\lambda}\frac{1-\beta}{\beta}} \leq \frac{1}{1+\frac{1-\beta}{\beta}} = \beta$$

where the final inequality follows from $\lambda \leq \lambda^+$. Hence $[0, \lambda^+] \rightarrow [0, \beta]$.

Furthermore, for $0 > \lambda \geq -\lambda^-$, note that also $m \geq \frac{1+\beta}{\beta}\lambda^-$. Since $\beta \in (0, 1]$, $m+\lambda \geq \frac{1+\beta}{\beta}\lambda^- + \lambda > 0$ and thus $\frac{\lambda}{m+\lambda} < 0$ which implies

$$-\frac{\lambda}{m+\lambda} \leq \frac{-\lambda}{\frac{1+\beta}{\beta}\lambda^- + \lambda} = \frac{1}{\frac{1+\beta}{\beta}\frac{\lambda^-}{-\lambda} - 1} \leq \frac{1}{\frac{1+\beta}{\beta} - 1} = \beta$$

i.e. $[-\lambda^-, 0) \rightarrow [-\beta, 0)$. This shows that indeed all the eigenvalues of G_{dir} is numerically less than or equal to β . Finally, assuming $m \neq 0$ or equivalently that $G_{obs} \neq \mathbf{0}$, either $m = \frac{1-\beta}{\beta}\lambda^+$ (and thus $\lambda^+ \neq 0$ is an eigenvalue of G_{obs}) for which the above shows that indeed λ^+ is mapped to β or $m = \frac{1+\beta}{\beta}\lambda^-$ (and hence $\lambda^- \neq 0$ and thus $-\lambda^-$ is an eigenvalue of G_{obs}) for which $-\lambda^-$ is mapped to $-\beta$. This shows that G_{dir} indeed has an eigenvalue which numerical value is β . \square

5.5 Examples

In this section, we will investigate how the algorithms Algorithm 1 and Algorithm 2 works in junction and, if so, observe how the algorithm can fail and what may be done to correct such cases. Initially, a few simple examples involving exponentiated multivariate Gaussians \mathbf{Y} .

Example 5.1. Exponentiated multivariate Gaussian

Let us consider a simple case with $\mathbf{Y} = e^{\mathbf{X}}$ (element wise exponentiation) where $\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \Sigma)$ where

$$\Sigma = \begin{bmatrix} \sigma_1^2 & 0.9\sigma_1\sigma_2 & 0 \\ 0.9\sigma_1\sigma_2 & \sigma_2^2 & 0 \\ 0 & 0 & \sigma_3^2 \end{bmatrix}$$

It is clear that to Algorithm 1, the mean is non-important as simply corresponds to a scaling of the Y_i variables. Furthermore, because of Corollary 5.2.1, theoretically, due to the uniqueness of the Copula C (as \mathbf{Y} is continuous) we should expect near equal or very similar results for \mathbf{Y} and \mathbf{X} from Algorithm 1. Additionally, different σ corresponds to different scaling of \mathbf{X} , and thus we should observe equal or near equal G_{dir} for all \mathbf{Y} . Initially, we shall see how this hypothesis holds up to the following three examples

$$\boldsymbol{\sigma} = (0.07, 0.3, 0.9), \quad \boldsymbol{\sigma} = (1, 1, 1), \quad \boldsymbol{\sigma} = (1, 2, 3)$$

In order for the sample size to not influence the results, we simulate a generous number of samples, namely, for the following results we have used $n = 10,000$ samples. For $\boldsymbol{\sigma} = (1, 1, 1)$, Algorithm 1 and Algorithm 2 returns the following (using $\alpha = 1$ and $\beta = 0.99$)

$$G_{dir} = \begin{bmatrix} -0.33396 & 0.6660 & 0.02512 \\ 0.6660 & -0.3341 & 0.02730 \\ 0.02512 & 0.02730 & -0.0020583 \end{bmatrix} \quad (5.7)$$

Similarly, for $\boldsymbol{\sigma} = (0.07, 0.3, 0.9)$:

$$G_{dir} = \begin{bmatrix} -0.3335 & 0.6665 & 0.01414 \\ 0.6665 & -0.3335 & 0.01418 \\ 0.01414 & 0.01418 & -0.00060124 \end{bmatrix} \quad (5.8)$$

Finally, for $\boldsymbol{\sigma} = (1, 2, 3)$:

$$G_{dir} = \begin{bmatrix} -0.1490 & 0.09535 & 0.3599 \\ 0.09535 & -0.2989 & 0.5831 \\ 0.3599 & 0.5831 & -0.4037 \end{bmatrix}$$

For $\sigma = (1, 1, 1)$ and $\sigma = (0.07, 0.3, 0.9)$ we observe the most resemblance to the Σ , although the resulting G_{dir} deviate in the final column. The difference is likely produced by Algorithm 1 as if the resulting G_{obs} was the same, then so would G_{dir} and from the above argument, we know that theoretically this should be the case. For the final example, $\sigma = (1, 2, 3)$, we see a completely different result and immediately suspect that there must be some numerical errors. Investigating the partial results of Algorithm 1 we immediately see a flaw in the supposedly uniform variables U_i as shown in figure Figure 5.3

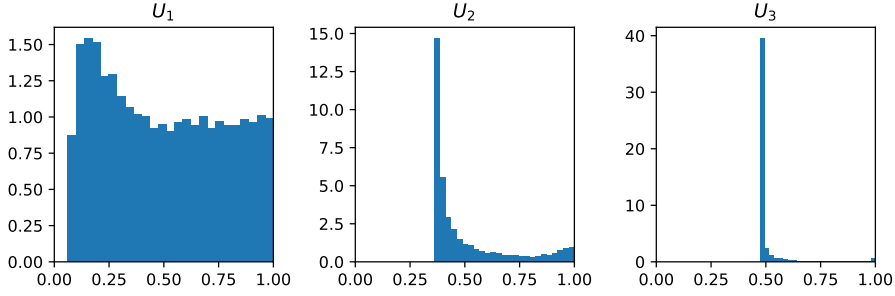


Figure 5.3: The samples transformed using $U_i = F_i(X_i)$ for $\sigma = (1, 2, 3)$. These should be uniformly distributed, but clearly this is not the case for U_2 and U_3 . Even U_1 does not quite resemble 10,000 samples from a uniform distribution.

Before handling this, the non-uniformity of U_1 in Figure 5.3 is likely also present in the case when $\sigma = (1, 1, 1)$. Indeed, Figure 5.4 shows that this is indeed the case.

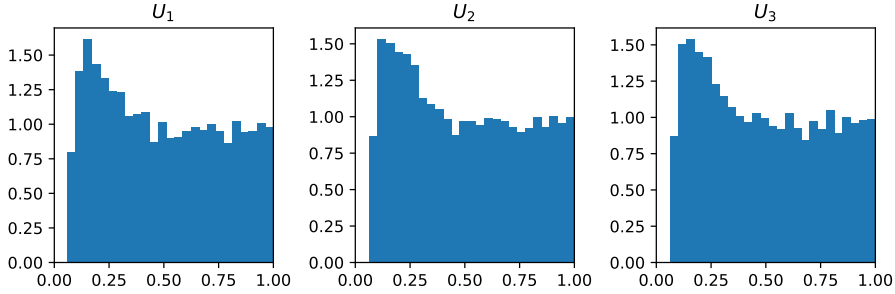


Figure 5.4: The samples transformed using $U_i = F_i(X_i)$ for $\sigma = (1, 1, 1)$.

Finally, just to be sure, $\sigma = (0.07, 0.3, 0.9)$ is also shown in Figure 5.5 and seems very reasonable, except for U_3 .

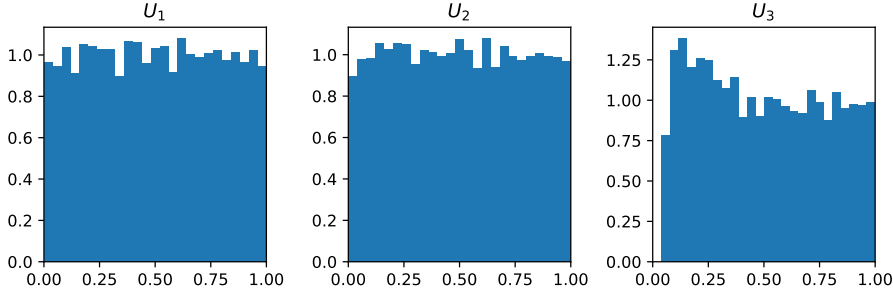


Figure 5.5: The samples transformed using $U_i = F_i(X_i)$ for $\sigma = (0.07, 0.3, 0.9)$.

From the above examples, it seems that the larger the variance, the worse the uniforms turn out. Reasons for this could include numerical issues when trying to calculate $u_i^{(j)}$ from $y_i^{(j)}$ by $u_i^{(j)} = \int_{-\infty}^{y_i^{(j)}} f_i(y) dy$ and bad fitting of the kernel density estimate from observations. In particular, for values similar, which happens in the case for large σ such that we observe large negative realizations of X_i , $y_i^{(j)}$ are almost 0, and when computing the integral could result in identical values. Furthermore, from Figure 5.6 we see that indeed the fit is quite poor. Note that we have zoomed in on the interval $[-200, 200]$ which contains 96.2% of observations. The poor fit is primarily due to the use of Scott's Rule *as discussed above* which in this case overshoots the optimal bandwidth by a lot.

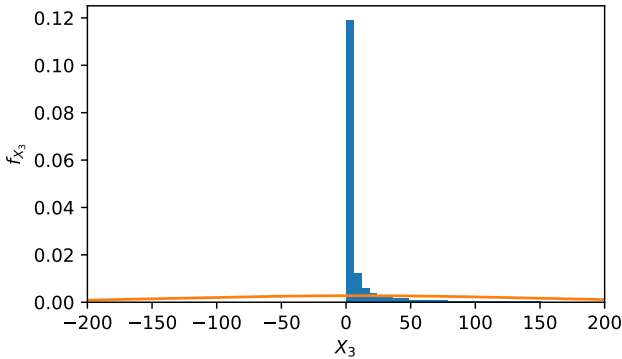


Figure 5.6

The poor fit also explains the high concentration of U_3 around 0.5 in Figure 5.3 as only 54.5% of the probability mass lies above 0.

However, also here Corollary 5.2.1 proves to be useful. Namely, we can get rid of the numerical issues by transforming Y_i using e.g. $\log(\cdot)$ or $(\cdot)^p$ for $p > 0$ to get even out the observations more. As the first simply inverts the initial transformation of X_i , we choose the latter as a more interesting case. In particular, choosing $p < 1$ will result in a more even distribution. In the following, $p = 1/10$ has been used to transform \mathbf{Y} prior to running Algorithm 1 and the resulting $u_i^{(j)}$ is shown in Figure 5.7.

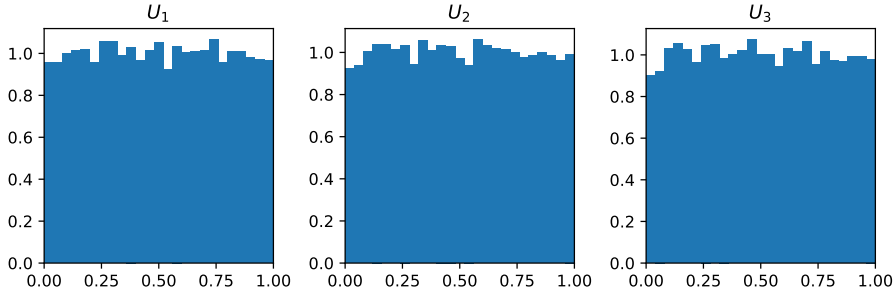


Figure 5.7

The resulting $u_i^{(j)}$ now seem to follow a uniform distribution and indeed the KDE fits much better as seen in Figure 5.8.

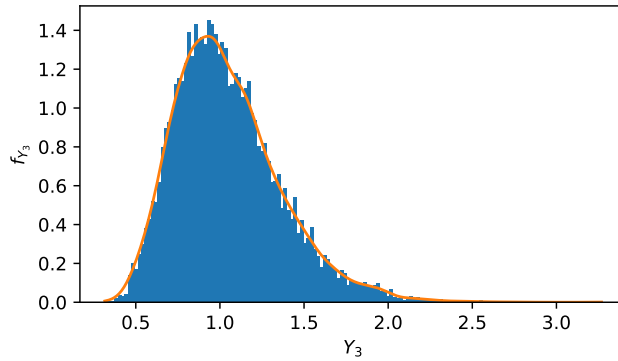


Figure 5.8

Turning to Algorithm 1 and Algorithm 2 we now find that G_{dir} is given by

$$G_{dir} = \begin{bmatrix} -0.3290 & 0.6610 & 0.008440 \\ 0.6610 & -0.3290 & 0.008150 \\ 0.008440 & 0.008150 & -0.0002061 \end{bmatrix}$$

Which is indeed much more comparable with the result from before in Equation 5.7 and Equation 5.8. The difference between G_{dir} from \mathbf{Y} and \mathbf{Y}^p is clearly visible in Figure 5.9 and also Figure 5.9b resembles the original correlation structure.

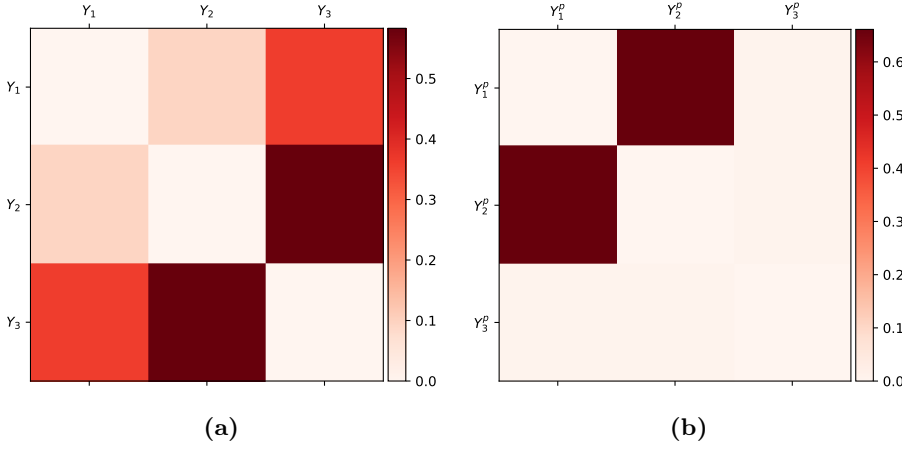


Figure 5.9: G_{dir} resulting from 10,000 samples from multi variate Gaussian with $\sigma = (1, 2, 3)$ in (a) with raw samples from \mathbf{Y} and in (b) the transformed data corresponding to \mathbf{Y}^p .

Finally, to end this example we shall compare with some theoretical results. Namely, the output G_{obs} of Algorithm 1 can also be calculated theoretically. For this, we shall use Proposition 5.4 which permits a theoretical result, namely

$$G_{obs} = \begin{bmatrix} 0 & -\frac{1}{2} \ln(1 - \rho_{12}^2) & -\frac{1}{2} \ln(1 - \rho_{13}^2) \\ -\frac{1}{2} \ln(1 - \rho_{21}^2) & 0 & -\frac{1}{2} \ln(1 - \rho_{23}^2) \\ -\frac{1}{2} \ln(1 - \rho_{31}^2) & -\frac{1}{2} \ln(1 - \rho_{32}^2) & 0 \end{bmatrix}$$

$$\cong \begin{bmatrix} 0 & 0.83037 & 0 \\ 0.83037 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Similarly, prior to deconvolution, using just the sampled \mathbf{X} (i.e. no exponential

transform), Algorithm 1 returns

$$G_{obs} = \begin{bmatrix} 0. & 0.71841756 & 0.01781815 \\ 0.71841756 & 0. & 0.01769672 \\ 0.01781815 & 0.01769672 & 0. \end{bmatrix}$$

Clearly these are not equal, but in this case, the error is suspected to originate from the estimated joint density. For example, considering X_1 and X_2 , we compare the estimated joint copula density and compare to the theoretical *reference til et sted hvor gaussisk copula står* shown in Figure 5.10 and Figure 5.11 respectively.

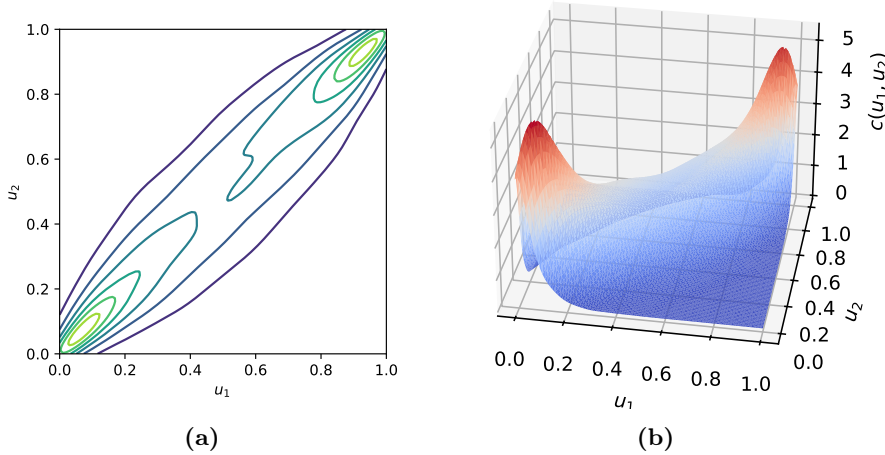


Figure 5.10: Estimated copula density c with $\rho = 0.9$ corresponding to X_1 and X_2 .

The noticeable difference is in the corners $(0,0)$ and $(1,1)$ where the theoretical copula density tends to infinity whereas the estimated density has modes at $(0.1, 0.1)$ and $(0.9, 0.9)$. In particular, simply rescaling the copula density in Algorithm 1 does not resemble the theoretical boundary which is a known issue *reference til artikel om undershoot peaks og boundary conditions for KDE*. A better approach may be to use jackknifing *link til afsnit of jackknifing, som ogsa indeholder reference til artikel hvor dette gøres*.

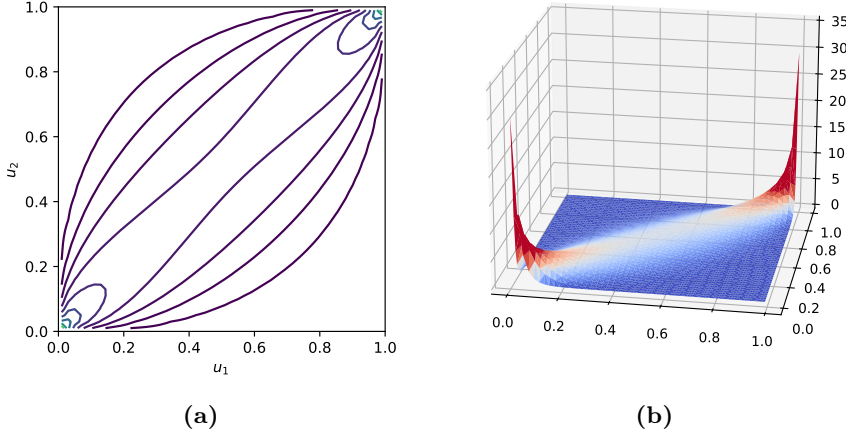


Figure 5.11: Theoretical copula density c with $\rho = 0.9$ corresponding to X_1 and X_2 .

We note however, that the underlying structure is still captured i.e. that Y_1 and Y_2 covary while Y_3 does not inform Y_1 or Y_2 and vice versa.

We continue with a similar example to the previous one. The key difference is the number of variables and a more complicated correlation structure to test the algorithms further.

Example 5.2. From Example 5.1 we saw how one could handle some numerical issues. Thus, in this example we shall not bother ourselves with such computations and merely focus on the correlation structure. In particular, we shall sample \mathbf{X} from a 10 dimensional

Proposition 5.4. *Given a bivariate normal distribution $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ where*

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$$

Then the mutual information $I(X_1, X_2) = -\frac{1}{2} \ln(1 - \rho^2)$.

Proof. This follows by direct computation Using e.g. that $I(X_1, X_2) = h(X_1) + h(X_2) - h(X_1, X_2)$ \square

$$G_{obs} = \sum_{k \geq 1} G_{dir}^k \quad (5.9)$$

5.5.1 Ensuring convergence and the effect of β

From [Network Deconvolution - A General Method to Distinguish Direct Dependencies over Networks - Supplementary Notes](https://compbio.mit.edu/nd/) and their implementation found at their webpage at <https://compbio.mit.edu/nd/> there seem to be an inconsistency between code and theory. In this section, we shall thus investigate from where the discrepancy arises. Initially, from the formulation in Equation 5.9, for the right-hand side to converge, it must have spectral radius at most 1 and to ensure convergence, less than 1. In the latter case,

Noget introduktion til at vi først lige kører nogle basale definition afsted, som kommer til at blive brugt senere

Definition 5.5. *Induced Norm.*

A matrix norm $||| \cdot |||$ is said to be induced by the vector norm $|| \cdot ||$ when

$$|||A||| = \sup_{||x||=1} ||Ax||$$

See [6]

Definition 5.6. *Sub-multiplicative Matrix norm*

A matrix norm $||| \cdot |||$ is said to be sub-multiplicative, if for every $A, B \in \mathbb{F}^{n \times n}$ where \mathbb{F} is either the real or complex field:

$$|||AB||| \leq |||A||| \cdot |||B|||$$

5.5.2 Robustness to noise

They show that the procedure is robust noise by considering that the observed information matrix is influenced by some noise $N \in \mathbb{R}^{n \times n}$ and characterize the noise by its Euclidean norm $\|N\|_2 := \sup_{\|x\|_2=1} \|Nx\|_2$. They show that

$$\|G_{dir} - \hat{G}_{dir}\|_2 \leq \gamma + \mathcal{O}(\delta^2 + \gamma^2 + \delta\gamma)$$

where γ is the spectral norm of N and δ is the spectral norm of \hat{G}_{obs} . However, this upper bound can actually be computed when $\delta + \gamma < 1$ and diverges when $\delta + \gamma > 1$. Furthermore, the result can be generalized to other norms than the spectral norm. In particular, the Frobenius norm admits a similar upper bound on the difference. Consider any matrix norm $||| \cdot |||$ for which $|||AB||| \leq |||A||| \cdot |||B|||$. It then follows that

$$\begin{aligned} |||G_{dir} - \hat{G}_{dir}||| &= |||G_{obs}(I + G_{obs})^{-1} - \hat{G}_{obs}(I + \hat{G}_{obs})^{-1}||| \\ &= |||-\sum_{k \geq 1} (-G_{obs})^k + \sum_{k \geq 1} (-\hat{G}_{obs})^k||| \\ &\leq \sum_{k \geq 1} |||G_{obs}^k - (G_{obs} + N)^k||| \\ &\leq \sum_{k \geq 1} \sum_{i=1}^k \binom{k}{i} |||N|||^i |||G_{obs}|||^{k-i} \\ &= \sum_{k \geq 1} \left((|||N||| + |||G_{obs}|||)^k - |||G_{obs}|||^k \right) \\ &= \frac{|||N||| + |||G_{obs}|||}{1 - (|||N||| + |||G_{obs}|||)} - \frac{|||G_{obs}|||}{1 - |||G_{obs}|||} \end{aligned}$$

Where the final equality assumes that $|||N||| + |||G_{obs}||| < 1$ and by splitting up the sum into 2 geometric series. However, we also show that the series diverge when $|||N||| + |||G_{obs}||| > 1$ which is not directly apparent as it is a difference geometric series. Namely, by the ratio test, letting $\gamma = |||N|||$ and $\delta = |||G_{obs}|||$

as by [5] [supp. notes NetworkDeconvolution-AGeneralMethodtoD...](#)

$$\begin{aligned}
 \lim_{n \rightarrow \infty} \left| \frac{(\gamma + \delta)^{n+1} - \delta^{n+1}}{(\gamma + \delta)^n - \delta^n} \right| &= \lim_{n \rightarrow \infty} \left| \frac{(\gamma + \delta) \left(1 + \frac{\gamma}{\delta}\right)^n - \delta}{\left(1 + \frac{\gamma}{\delta}\right)^n - 1} \right| \\
 &= \lim_{n \rightarrow \infty} \left| \delta + \gamma \frac{\left(1 + \frac{\gamma}{\delta}\right)^n}{\left(1 + \frac{\gamma}{\delta}\right)^n - 1} \right| \\
 &= \lim_{n \rightarrow \infty} \left| \delta + \gamma \frac{1}{1 - \left(1 + \frac{\gamma}{\delta}\right)^{-n}} \right| \\
 &= |\gamma + \delta| = \gamma + \delta
 \end{aligned}$$

as $\gamma, \delta > 0$ unless they are the zero-matrix in which case the above is nonsensical from a perspective of interest. The above shows that indeed the above result diverges when $\gamma + \delta > 1$.

Thus, for the spectral norm, denoted by $|||\cdot|||_2$, one simply needs the fact that it is sub-multiplicative (Definition 5.6) which follows from the fact that the spectral norm is induced by the l_2 norm (see Definition 5.5) on \mathbb{R} , and hence

$$\sup_{||x||_2=1} ||ABx|| \leq ||A||_2 \cdot \sup_{||x||_2=1} ||Bx|| = ||A||_2 ||B||_2$$

which by definition means that $||AB||_2 \leq ||A||_2 ||B||_2$. The Frobenius norm $|||\cdot|||_F$ is also sub-multiplicative and depending on the use case may be very useful.

5.5.3 Quick made up example

As a thought up experiment for the underlying

APPENDIX A

Stuff

This appendix is full of stuff ...

Bibliography

- [1] M. L. C. Vicente, “A benchmark model to generate batch process data for machine learning testing and comparison,” 2021.
- [2] S. Feizi, D. Marbach, M. Médard, and M. Kellis, “Network deconvolution as a general method to distinguish direct dependencies in networks,” *Nature biotechnology*, vol. 31, pp. 726 – 733, 2013.
- [3] D. Kurowicka and H. Joe, *Dependence Modeling: Vine Copula Handbook*. 12 2010.
- [4] A. Kraskov, H. Stögbauer, and P. Grassberger, “Estimating mutual information,” *Phys. Rev. E*, vol. 69, p. 066138, Jun 2004.
- [5] Y.-N. Sun, W. Qin, and Z. Zhuang, “Nonparametric-copula-entropy and network deconvolution method for causal discovery in complex manufacturing systems,” *Journal of Intelligent Manufacturing*, vol. 33, 08 2022.
- [6] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 2 ed., 2012.