

Zwischenstand Spektraldichten-Testverfahren

Carsten Stahl

January 2, 2026

TU Dortmund University - Department of Computer Science

Performance Verbesserungen

Bivariater MA-Prozess

Quarter- und Halfplane Prozess

Monte Carlo Simulationen

Beweis für bedingte Erwartung und Varianz

Testen auf Isotropie

Performance Verbesserungen

Stand vorher

- Riemann Integral über Fourier Frequenzen gelöst
- Für jede Frequenz die Kernelgewichte neu ausgerechnet (auf dem ganzen Gitter) und dann durch ein das Hadamard Produkt ausgerechnet
 - Das Ergibt für einen Funktionswert eine Laufzeit von $\mathcal{O}(N^2)$
- Wir müssen aber den Funktionswert für alle Fourier Frequenzen ausrechnen und kommen damit auf $\mathcal{O}(N^4)$

Der Trick

- Die Fourier Frequenzen sind äquidistant. Daher bleiben die Kernelgewichte gleich
 - Der Abstand zwischen ω_k und ω_{k-1} ist immer $2\pi/N$ deswegen gilt:

$$\omega_{k+1} - \omega_k = \omega_k - \omega_{k-1}$$

- Wir müssen daher die Kernelgewichte nur einmal erzeugen und anschließend eine Konvolution mit den Kernelgewichten über das Periodogramm laufen lassen.
- Die Laufzeit beträgt dann nur noch $\mathcal{O}(N^3 \log N)$

Laufzeit Verbesserung

- `phi_n_star` für 20×20 Gitter, $B = 100$ lief für 5233.2311 ms^1
- `phi_n_star_fast` für denselben Einstellungen lief für 29 ms
- Dieser neue Algorithmus inspirierte eine Implementierung in **Pytorch**
 - Diese macht eine Monte-Carlo Simulation mit $N = 200$ in $\approx 500\text{ ms}$
 - Die Laufzeit von `phi_n_star` wäre $\approx 1.000.000\text{ ms}$

¹Durchschnitt für 10 Ausführungen

Bivariater MA-Prozess

Bisheriger Stand

Die MA(1) oder MA(2) Prozesse, die wir bisher simuliert hatten, waren eindimensional und hatten eine Spektraldichte:

$$f_{X,Y}(\omega, \tilde{\omega}) = \begin{pmatrix} f_X(\omega, \tilde{\omega}) & 0 \\ 0 & f_Y(\omega, \tilde{\omega}) \end{pmatrix}$$

Sinn von Bivariatem MA Prozess

Jetzt wollten wir wissen, was passiert, wenn:

$$f_{X,Y}(\omega, \tilde{\omega}) = \begin{pmatrix} f_X(\omega, \tilde{\omega}) & \rho(\omega, \tilde{\omega}) \\ \rho(\omega, \tilde{\omega}) & f_Y(\omega, \tilde{\omega}) \end{pmatrix}$$

Mit $\rho(\omega, \tilde{\omega}) > 0$.

Lösung

Für einen möglichst einfachen Ansatz habe ich die Gaussische Weiße Rauschen korreliert:

$$\text{Cov}(\varepsilon_{X,ij}, \varepsilon_{Y,ij}) = \rho \quad \forall (i,j) \in \mathbb{I}$$

Unter H_0 würde das bedeuten, dass:

$$f_{X,Y}(\omega, \tilde{\omega}) = \begin{pmatrix} f(\omega, \tilde{\omega}) & \rho f(\omega, \tilde{\omega}) \\ \rho f(\omega, \tilde{\omega}) & f(\omega, \tilde{\omega}) \end{pmatrix}$$

Ergebnisse

- Ablehnung unter H_0 für $N = 1000$ auf einem 20×20 Gitter mit $B = 1000$ und $\alpha = 5\%$:
 - 5.6%
- Unter der einfachen Hypothese mit (Sonst dieselben Einstellungen):

$$K_0 = \begin{pmatrix} 0 & 0 & 0 \\ .7 & 1 & .7 \\ 0 & 0 & 0 \end{pmatrix} K_1 = \begin{pmatrix} 0 & .7 & 0 \\ 0 & 1 & 0 \\ 0 & .7 & 0 \end{pmatrix}$$

- Wurde in 100% der Fälle abgelehnt.
- Auswirkung auf die Güte? Nächstes mal :)

Quarter- und Halfplane Prozess

Theoretischer Rahmen: Quarterplane

Wir können den Quarterplane Prozess mit einem Hadamard-Produkt ausdrücken:

$$X_{ij} = \begin{pmatrix} \phi_1 & 1 \\ \phi_2 & \phi_3 \end{pmatrix} \odot \begin{pmatrix} X_{i,j-1} & \varepsilon_{ij} \\ X_{i-1,j-1} & X_{i-1,j} \end{pmatrix}$$

Mit $\varepsilon_{ij} \stackrel{iid}{\sim} \mathcal{N}(0, 1)$

Naiver Algorithmus

- Man könnte jedes X_{ij} nacheinander simulieren (so wie in dem Zeitreihenfall)
- ABER
 - Burn in?
 - Laufzeit?
 - Stationarität?
- Deswegen war es einen Versuch wert, einfach direkt über die Woldzerlegung zugehen

Theoretischer Rahmen: Quarterplane

Unter Stationarität können wir diesen Prozess aber auch als $MA(\infty)$ Prozess darstellen (Über die Wold-Zerlegung):

$$X_{ij} = \begin{pmatrix} \cdots & \psi_{12} & 1 \\ \cdots & \psi_{22} & \psi_{21} \\ \ddots & \vdots & \vdots \end{pmatrix} \odot \begin{pmatrix} \cdots & \varepsilon_{i,j-1} & \varepsilon_{ij} \\ \cdots & \varepsilon_{i-1,j-1} & \varepsilon_{i-1,j} \\ \ddots & \vdots & \vdots \end{pmatrix}$$

Eigenschaften der Wold-Koeffizienten

Die Rand-Koeffizienten ψ_{1i}, ψ_{j1} von der Wold Zerlegung verhalten sich wie im normalen univariaten AR(1) Fall:

$$\psi_{1i} = \phi_1^i \quad \psi_{i1} = \phi_3^i$$

Für alle restlichen Koeffizienten gilt:

$$\psi_{ij} = \phi_1 \psi_{ij-1} + \phi_2 \psi_{i-1,j-1} + \phi_3 \psi_{i-1j} \quad \forall i, j > 1$$

Basierend auf diesen Eigenschaften kann man sich iterativ die Wold Koeffizienten berechnen.

Schritt 1: Erzeugen der Rand-Koeffizienten:

$$\Psi^{(1)} = \begin{pmatrix} \dots & \phi_1^2 & \phi_1 & 1 \\ \dots & 0 & 0 & \phi_3 \\ \dots & 0 & 0 & \phi_3^2 \\ \ddots & \vdots & \vdots & \vdots \end{pmatrix}$$

Schritt 2: Erzeugen der ersten Reihe:

$$\Psi^{(1)} = \begin{pmatrix} \dots & \phi_1^2 & \phi_1 & 1 \\ \dots & 0 & \phi_2 + \phi_3\phi_1 + \phi_1\phi_3 & \phi_3 \\ \dots & 0 & 0 & \phi_3^2 \\ \ddots & \vdots & \vdots & \vdots \end{pmatrix}$$

Schritt 2: Erzeugen der ersten Reihe:

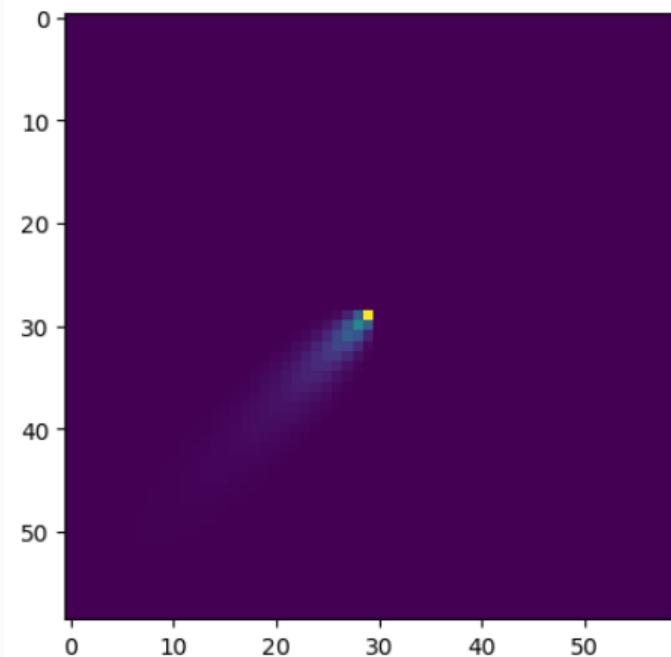
$$\Psi^{(1)} = \begin{pmatrix} \dots & \phi_1^2 & \phi_1 & 1 \\ \dots & \phi_3\phi_2^2 + \phi_2\phi_1 + \phi_1(\phi_2 + \phi_3\phi_1 + \phi_1\phi_3) & \phi_2 + \phi_3\phi_1 + \phi_1\phi_3 & \phi_3 \\ \dots & 0 & 0 & \phi_3^2 \\ \ddots & \vdots & \vdots & \vdots \end{pmatrix}$$

Algorithmus für Quarterplane - 3

Schritt 2: Erzeugen der zweiten Reihe (usw):

$$\Psi^{(1)} = \begin{pmatrix} \dots & \phi_1^2 & \phi_1 & 1 \\ \dots & \phi_3\phi_2^2 + \phi_2\phi_1 + \phi_1(\phi_2 + \phi_3\phi_1 + \phi_1\phi_3) & \phi_2 + \phi_3\phi_1 + \phi_1\phi_3 & \phi_3 \\ \dots & \leftarrow & \phi_3(\phi_2 + \phi_3\phi_1 + \phi_1\phi_3) + \phi_2\phi_3 + \phi_1\phi_3^2 & \phi_3^2 \\ \ddots & \vdots & \vdots & \vdots \end{pmatrix}$$

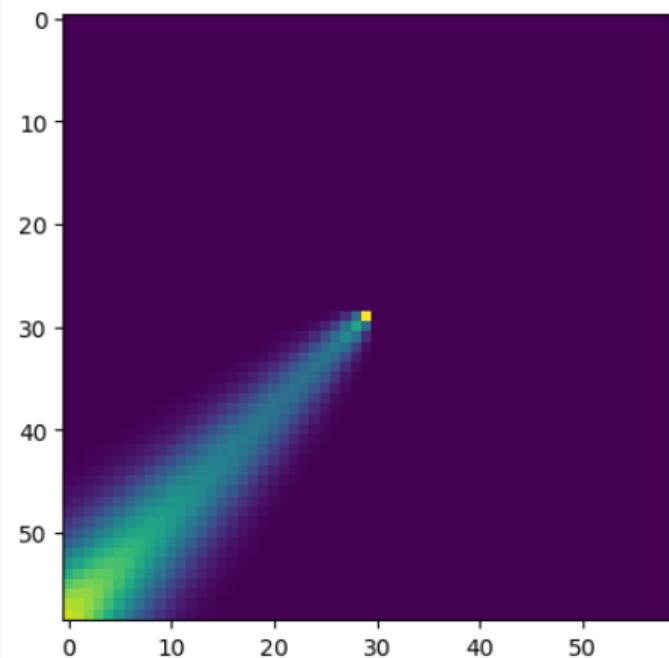
Quarterplane Wold weights



$$\tilde{\Phi}_q = \begin{pmatrix} .3 & 1 \\ .3 & .3 \end{pmatrix}$$

Figure 1: Wold Koeffizienten für Φ

Quarterplane Wold weights (nicht stationär)



$$\tilde{\Phi}_q = \begin{pmatrix} .35 & 1 \\ .35 & .35 \end{pmatrix}$$

Figure 2: Wold Koeffizienten für Φ

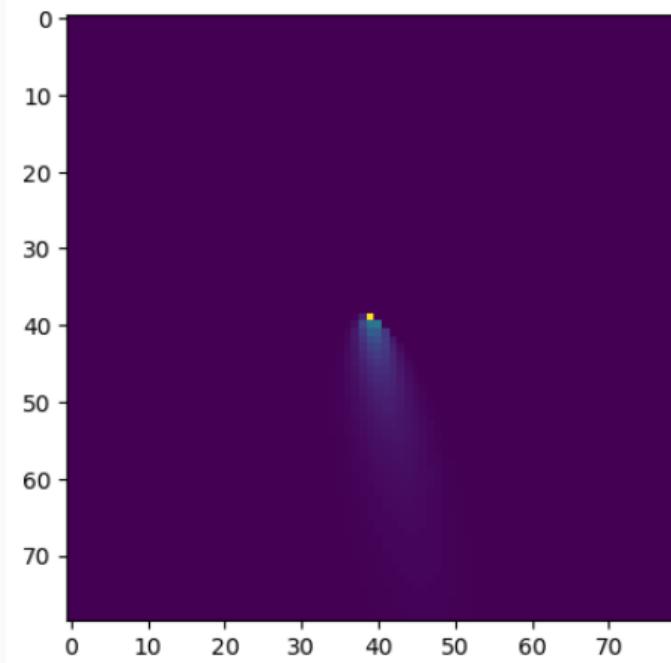
Wold-Zerlegung von Halfplane Prozess

$$X_{ij} = \begin{pmatrix} \dots & \psi_{12} & 1 & 0 & 0 & \dots \\ \dots & \psi_{22} & \psi_{21} & \psi_{20} & 0 & \dots \\ \dots & \psi_{32} & \psi_{31} & \psi_{30} & \psi_{3(-1)} & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} \odot \begin{pmatrix} \dots & \varepsilon_{i,j-1} & \varepsilon_{ij} & 0 & 0 & \dots \\ \dots & \varepsilon_{i-1,j-1} & \varepsilon_{i-1,j} & \varepsilon_{i-1,j+1} & 0 & \dots \\ \dots & \varepsilon_{i-2,j-1} & \varepsilon_{i-2,j} & \varepsilon_{i-2,j+1} & \varepsilon_{i-2,j+2} & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

Algorithmus für Halfplane

- Funktioniert auf dieselbe Weise
- Spezialfall einen Eintrag vor der Diagonalen
- Erzeugen nach Reihe

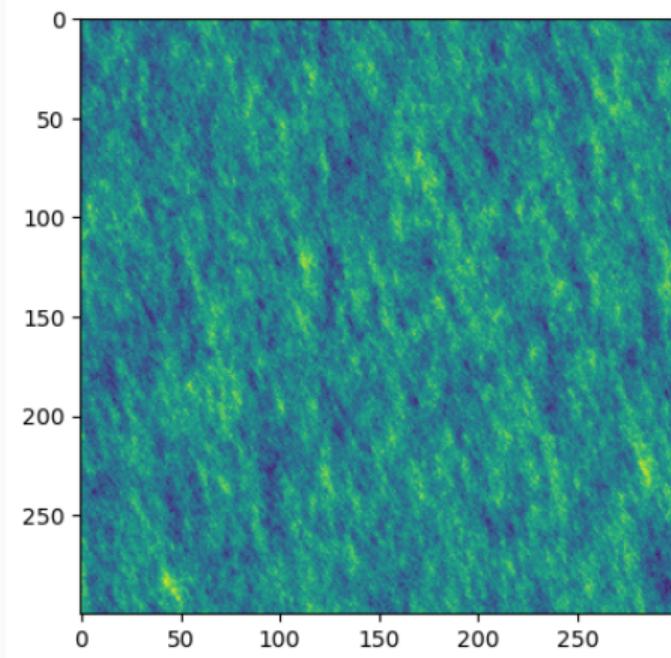
Halfplane Wold Koeffizienten



$$\Phi_h = \begin{pmatrix} .1 & 1 & 0 \\ .15 & .3 & .4 \end{pmatrix}$$

Figure 3: Wold Koeffizienten für Φ_h

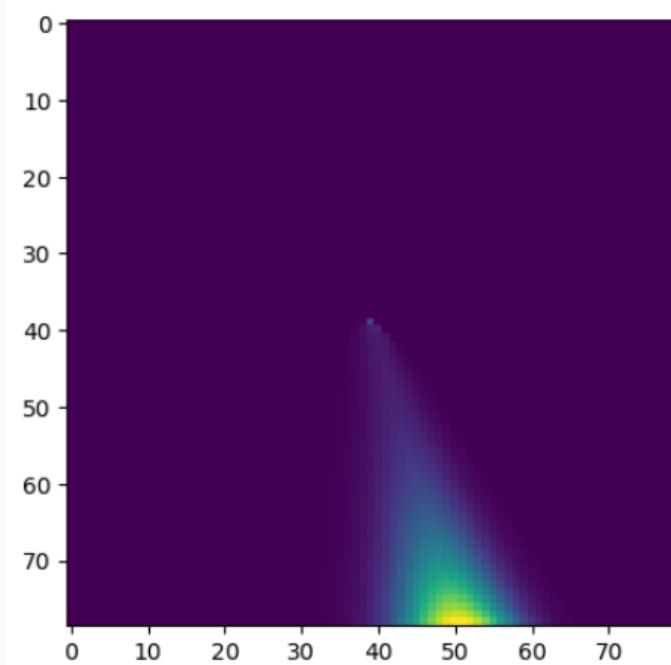
Halfplane Wold Koeffizienten



$$\Phi_h = \begin{pmatrix} .1 & 1 & 0 \\ .15 & .3 & .4 \end{pmatrix}$$

Figure 4: Beispielrealisierung von Φ_h

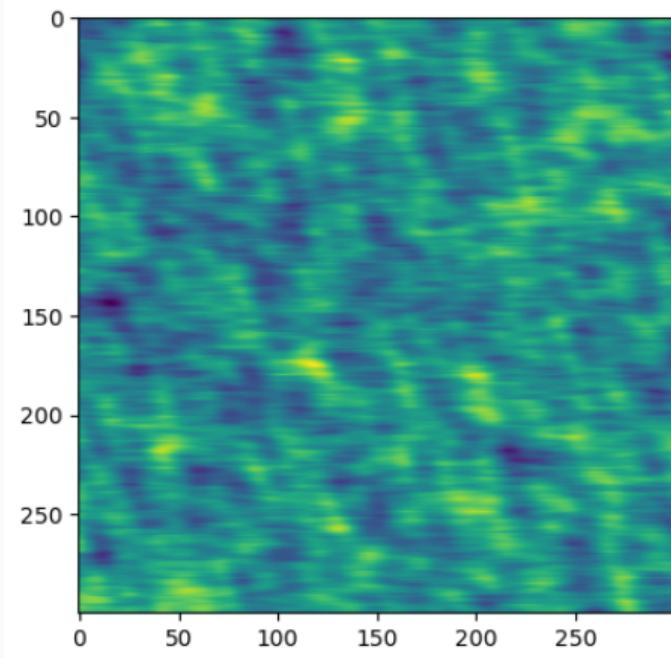
Halfplane Wold Koeffizienten (nicht stationär)



$$\tilde{\Phi}_h = \begin{pmatrix} .1 & 1 & 0 \\ .15 & .3 & .55 \end{pmatrix}$$

Figure 5: Wold Koeffizienten für $\tilde{\Phi}_h$

Halfplane Wold Koeffizienten (nicht stationär)

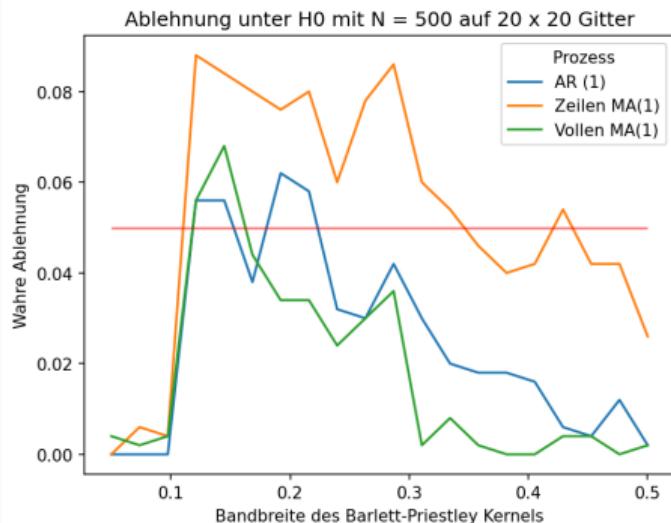


- Rand von dem berechneten $\tilde{\Psi}_h$ zeigt sich durch Streifen
- Viel verschwommener, da die Wold Koeffizienten in der breite Wachsen.

Figure 6: Beispielrealisierung von $\tilde{\Phi}_h$

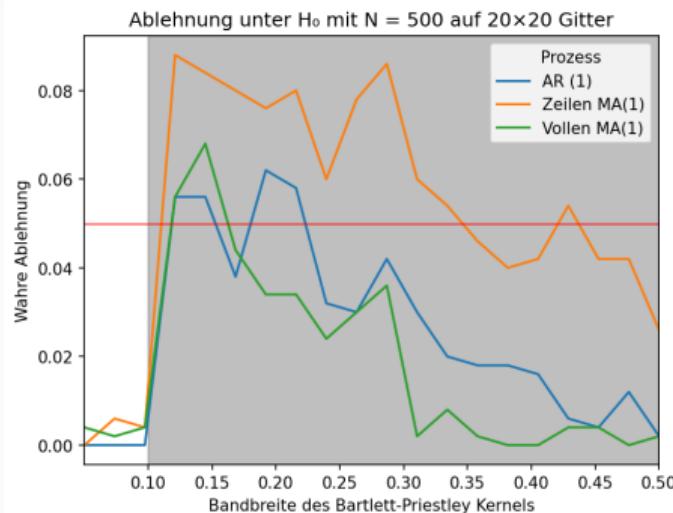
Monte Carlo Simulationen

Ablehnung unter H_0



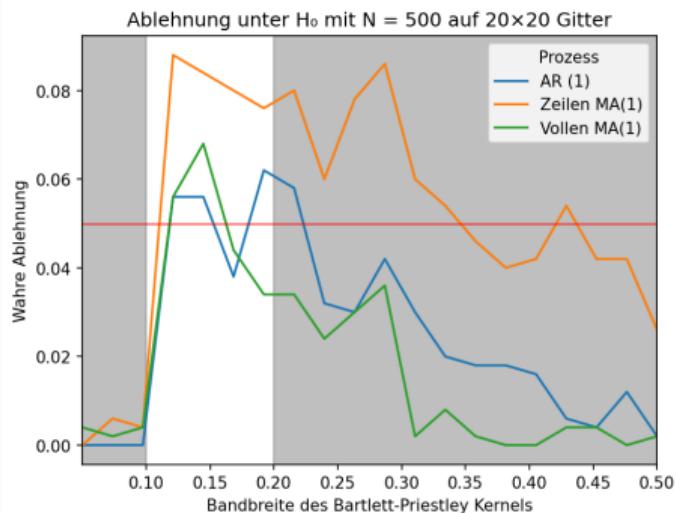
- Wir wollen den Einfluss von verschiedenen Bandbreiten auf die wahre Größe (Size)
- Parameter:
 - $N = 500$ Simulationen
 - 20×20 Gitter
 - $B = 100$ und $\alpha = 5\%$
 - $H = 20$ Verschiedene Bandbreiten zwischen .05 bis .5

Ablehnung unter H_0



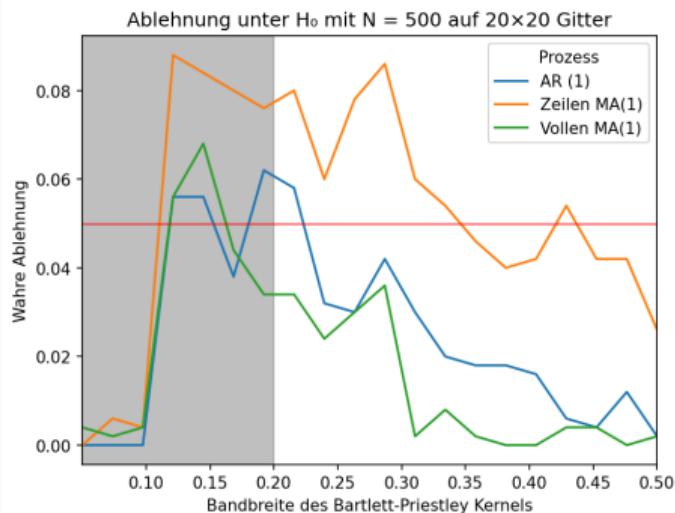
- Für $h_r = h_c < .1$ ist der Kernel kleiner $\frac{2\pi}{20}$ auf jeder Achse
 - Das führt dazu, dass nur ein Element von dem Kernel geglättet wird
 - Tauschen auf den Frequenzen macht hier keinen Unterschied mehr (Kommutativgesetz)

Ablehnung unter H_0



- Unter $h_r = h_c \in [.1, .2]$ werden dann 5 Elemente von dem Kernel geglättet
 - Welche Elemente geglättet werden, macht jetzt einen großen Unterschied
 - Das bedeutet T_n^* reagiert stärker auf kleine Unterschiede zwischen den Spektraldichten
 - Ablehnung schließt für alle Prozesse nach oben.

Ablehnung unter H_0



- Unter $h_r = h_c > .2$ werden immer mehr Elemente geglättet
 - Einzelne Elemente haben immer weniger Einfluss auf die Glättung
 - Varianz von T_n^* wird geringer und leichte Unterschiede lassen sich nicht mehr erkennen
 - Sollte man die Bandbreite weiter erhöhen wird irgendwann gar nicht mehr abgelehnt

Ablehnung unter H_1

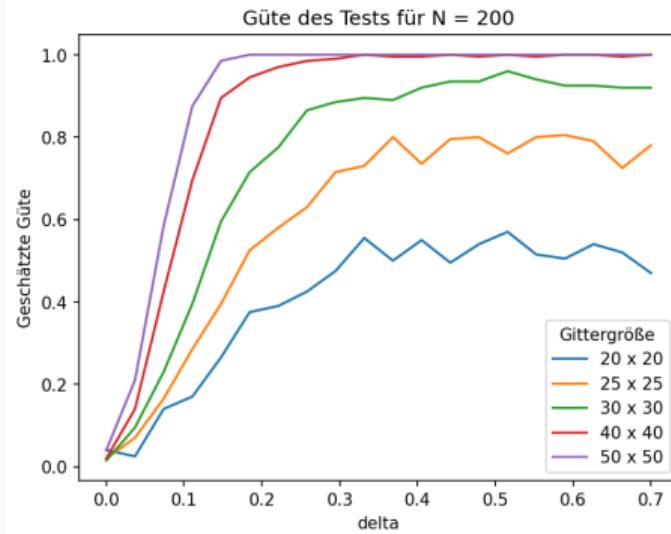
Unser erstes setup ist ein MA Prozess ohne Korrelation zwischen den Stichproben:

$$K_0 = \begin{pmatrix} .7 & .7 & .7 \\ .7 & 1 & .7 \\ .7 & .7 & .7 \end{pmatrix} \quad (1)$$

Ablehnung unter H_1 : Simulationseinstellungen

- $N = 200$ Simulation pro Datenpunkt
- $D = 20$ Verschiedene $\delta \in [0, .7]$
- Verschiedene Gittergrößen von 20×20 bis 50×50

Ablehnung unter H_1



- Sehr sinnvolle Powerfunktion
- Für das 20×20 - 30×30 Gitter scheint ein Erhöhen des δ in keiner größeren Güte zu resultieren

Ablehnung unter H_1

Die nächste Alternative wollen wir auf den AR-Prozessen definieren:

$$\Phi_0 = \begin{pmatrix} .1 & 1 \\ .03 & .8 \end{pmatrix} \Phi_1 = \begin{pmatrix} .1 + \delta & 1 \\ .03 & .8 - \delta \end{pmatrix}$$

- $N = 200$ Simulationen
- $D = 20$ verschiedene $\delta \in [0, .7]$
- 5 verschiedene Gittergrößen

Ablehnung unter H_1

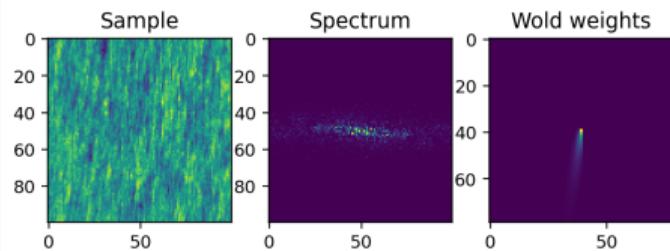


Figure 7: Angaben für Φ_0

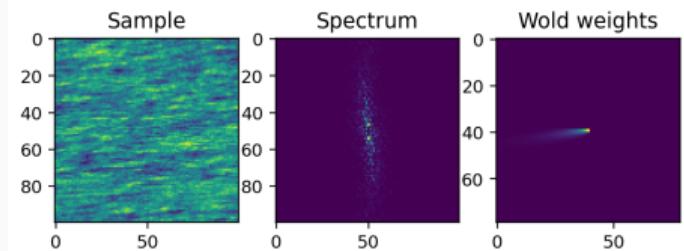
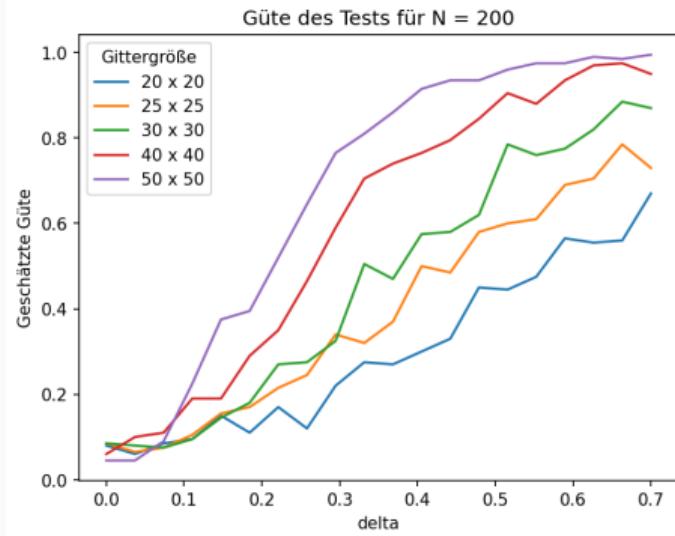


Figure 8: Angaben für Φ_1 mit $\delta = .7$

Ablehnung unter H_1



- Weniger Güte für den AR Prozess
- Bandbreite $h = .18$ und nicht $h = N^{2(-1/5)}$ wie im Paper empfohlen

Ablehnung unter H_1

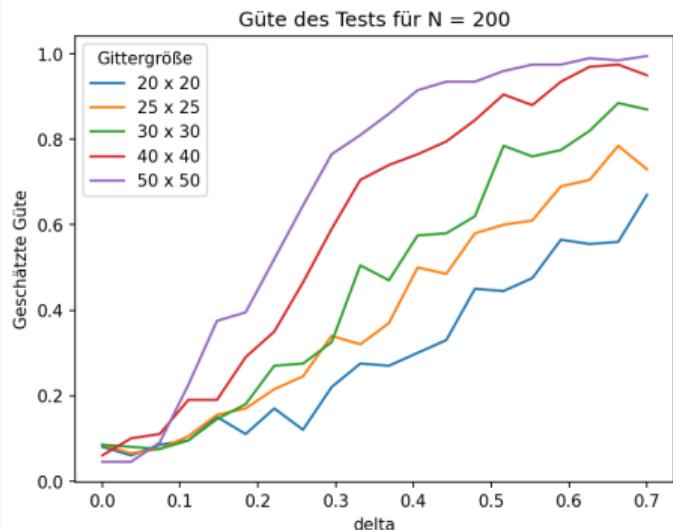


Figure 9: Güte unter $h_r = h_c = .18$

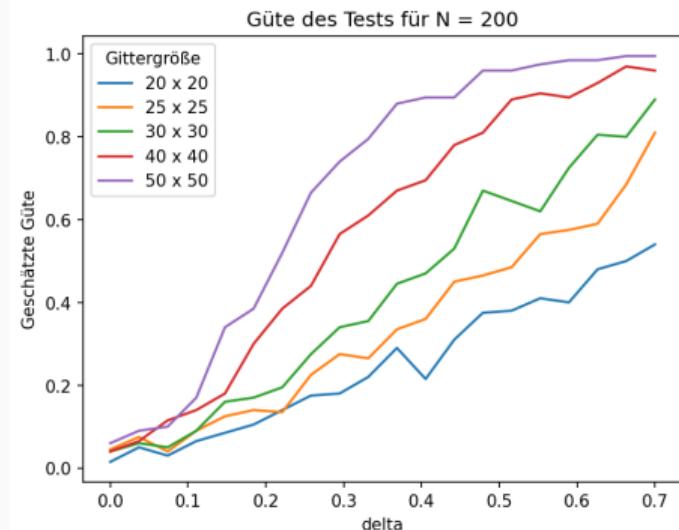


Figure 10: Güte unter $h_r = h_c = N^{2(-1/5)}$

Ablehnung unter H_1

Eine weitere Simulation war der Übergang von einem Quarterplane zu einem Halfplane Prozess:

$$\Phi_0 = \begin{pmatrix} .04 & 1 & 0 \\ .03 & .8 & 0 \end{pmatrix} \quad \Phi_1 = \begin{pmatrix} .04 & 1 & 0 \\ .03 & .8 - \delta & \delta \end{pmatrix}$$

Ablehnung unter H_1

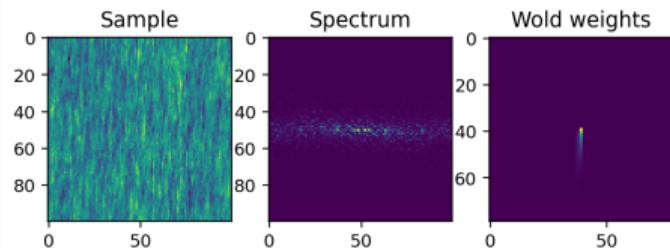


Figure 11: Angaben zu Φ_0

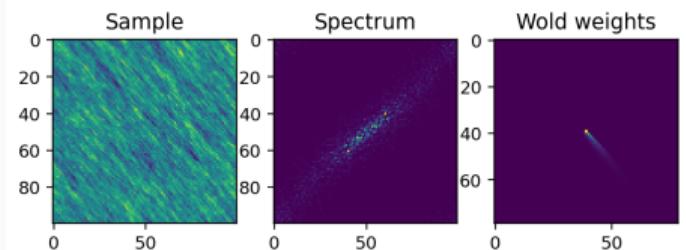
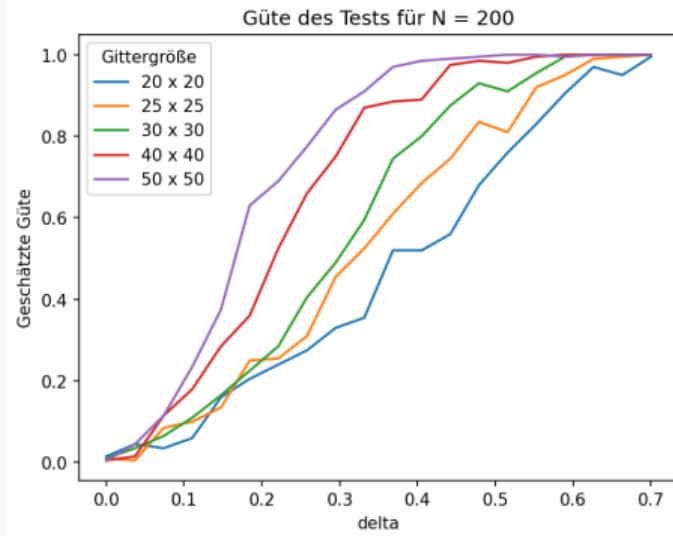


Figure 12: Angaben zu Φ_1 mit $\delta = .7$

Ablehnung unter H_1



- Ein wenig mehr Güte als der AR Fall
- Außerdem scheinen für alle Gittergrößen die Güten zu 100 % zu gehen und nicht irgendwann zu stagnieren

Beweis für bedingte Erwartung und Varianz

Was soll berechnet werden

Gesucht sind diese Größen (für beliebiges q und $p = 1$):

$$\mathbb{E}(T_n^* | I_1, \dots, I_q) \quad (3)$$

$$Var(T_n^* | I_1, \dots, I_q) \quad (4)$$

Die Indexfunktion

- Da sich ein Tauschen auf einem Frequenzpaar schlecht Quantifizieren lässt führen wir diese Indexfunktion als Hilfsmittel ein:

$$1_{\{j\}}(\pi_{k_1 k_2}(r)) = 1_j(\pi_{k_1 k_2}(r))$$

Bedeutung der Indexfunktion für den Beweis

In dem Paper wird diese Indexfunktion verwendet, um die Differenz zwischen dem gepoolten und getauschten Periodogramm einfacher auszudrücken:

$$I_{\pi_{kl}(r)}(\omega_k, \omega_l) - \tilde{I}_{\pi_{kl}(r)}(\omega_k, \omega_l) = -\frac{1}{q} \sum_{j=1}^q (1 - q1_j(\pi_{kl}(r))) I_j(\omega_k, \omega_l)$$

Für den Erwartungswert und Varianz brauchen wir die Ergebnisse aus **Lemma 6.2**, Welches $\mathbb{E}[(1 - q1_j(\pi_{kl}(r)))(1 - q1_{\tilde{j}}(\pi_{\tilde{k}\tilde{l}}(r)))]$ ausrechnet.

Testen auf Isotropie

Es gilt:

$$C(k, l) = C(\|(k, l)\|_2) \iff f(\omega, \tilde{\omega}) = f(\|\omega, \tilde{\omega}\|_2)$$

Wir können also auf Gleichheit der Spektraldichte bei frequenzen mit derselben L_2 Distanz testen.

- Für alle $d > 0$ Tausche $(\omega, \tilde{\omega}) : \|(\omega, \tilde{\omega})\|_2 = d$
- Problem: q nicht mehr Konstant