

Vorlesung
Statistische Methoden der Datenanalyse
Prof. Dr. Dr. Wolfgang Rhode

Entfaltung

Inhalt

- Einleitung
 - Messung physikalischer Parameter
 - Entfaltung
- Faltungsintegral
 - Fredholm Integral Gleichung
 - Diskretisierung, Responsematrix
- Entfaltungsproblem
 - Lösung durch Invertierung
 - Eigenwerte der Responsematrix
 - Regularisierung
 - Singulärwertzerlegung
- Lösung durch Fit
 - Kleinste Quadrate, Poisson Likelihood Fit, Regularisierung

MOTIVATION

Messung physikalischer Parameter

- Wir sind interessiert an der Messung eines Parameters x
 - Z.B. Energiespektrum der kosmischen Strahlung
- Gemessen wird Stichprobe $\{y\}_N$ mit Umfang N
 - Observablen y entsprechen nicht direkt der gesuchten Größe x

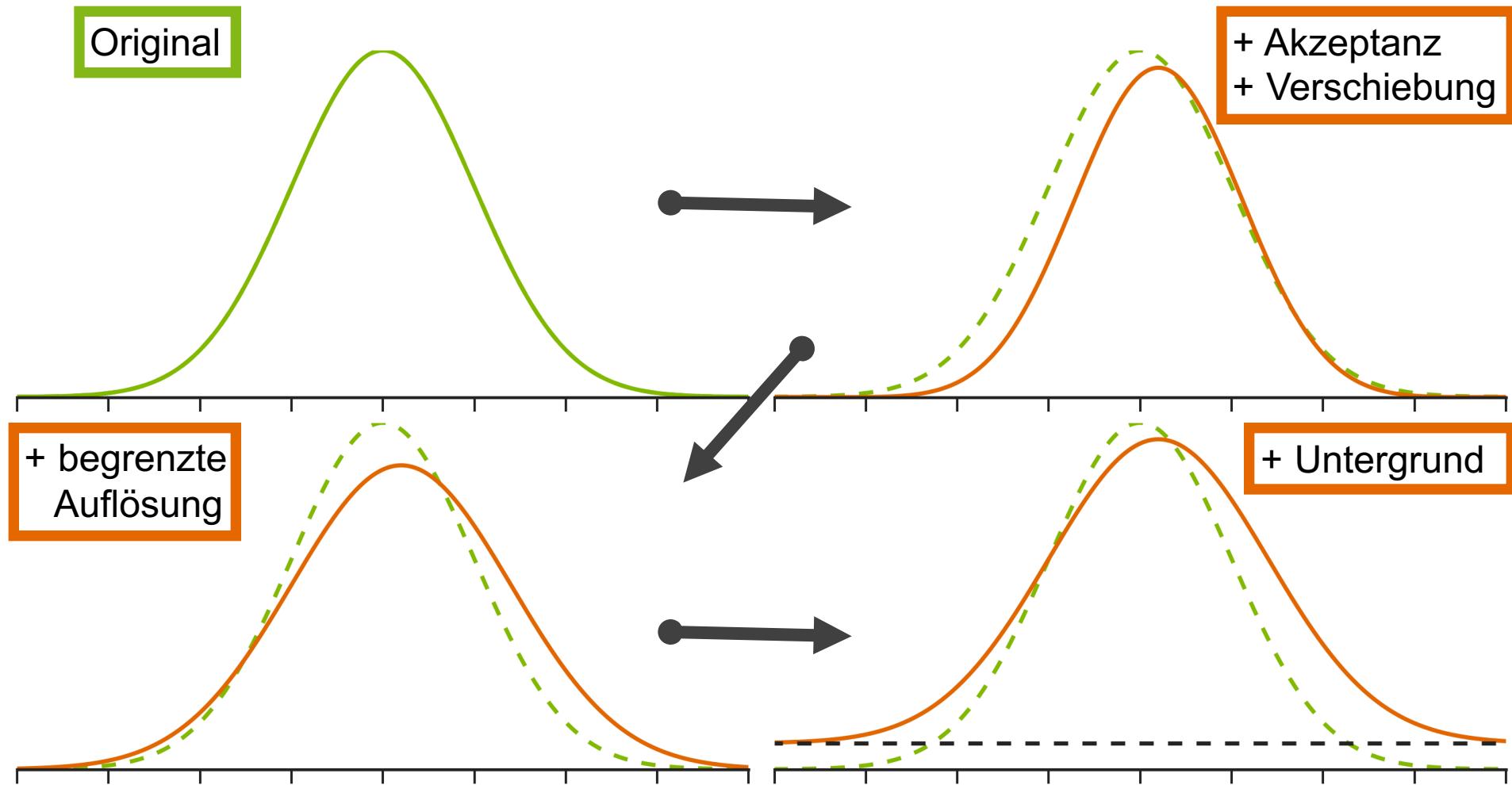
→ Messprozess / Detektor ist nicht perfekt

- *Direkter Prozess:* Wahre Verteilung $f(x) \rightarrow$ Gemessene Verteilung $g(y)$
 - Z.B. durch Monte-Carlo Simulation
 - realer Messprozess
- *Inverser Prozess:* Gemessene Verteilung $g(y) \rightarrow$ Wahre Verteilung $f(x)$
 - **Entfaltung** (engl. auch *Unfolding, Deconvolution*)

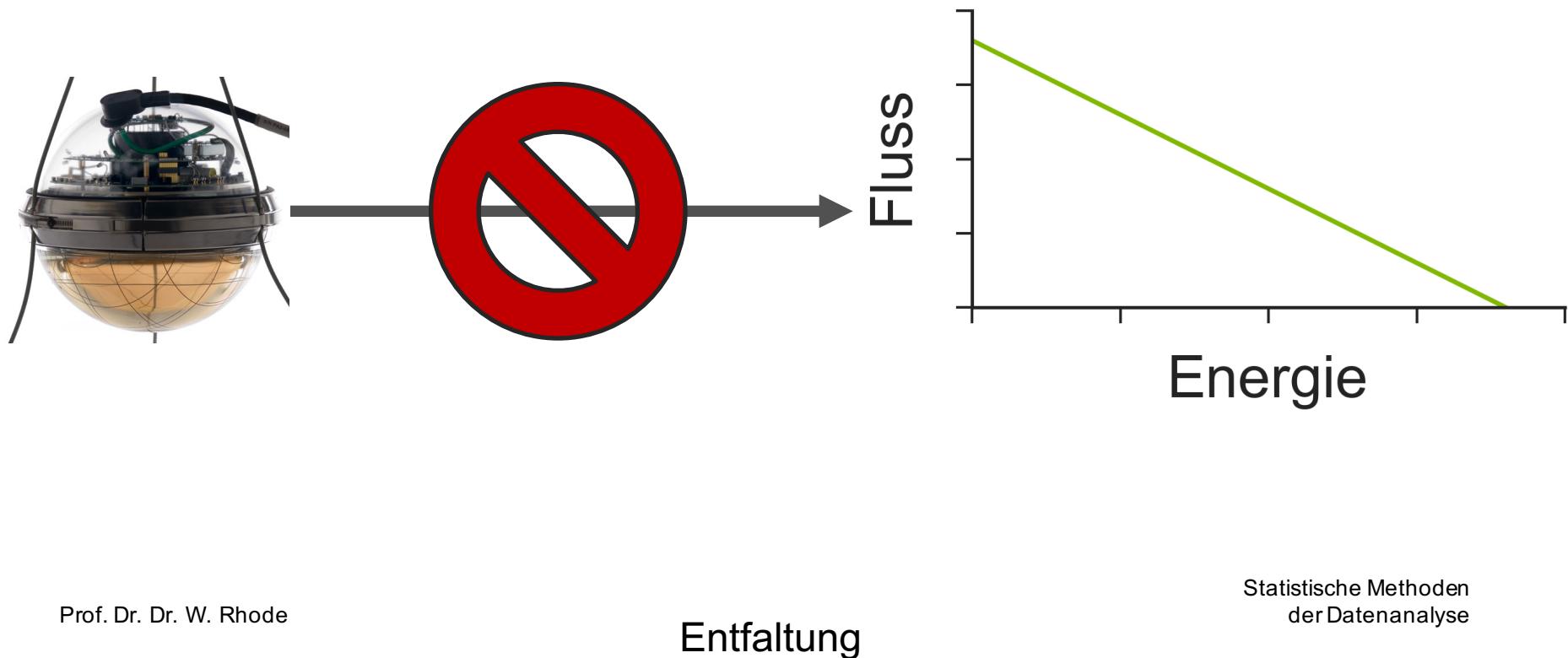
Entfaltung - Messprozess, realer Detektor

- Durch **physikalische Prozesse und den Messprozess** selbst können Parameter nur indirekt gemessen werden, z.B.
 - Wechselwirkung kosmischer Teilchen mit der Atmosphäre
 - Begrenzte Lebensdauer → Zerfall in andere Teilchen
- **Detektoren sind nicht perfekt**, die Messung wird erschwert durch z.B.
 - Begrenzte Auflösung
 - Transformationseffekte
 - Begrenzte Akzeptanz → Ereignisse werden nicht immer detektiert
 - Untergrund durch andere Prozesse, die ebenfalls detektiert werden

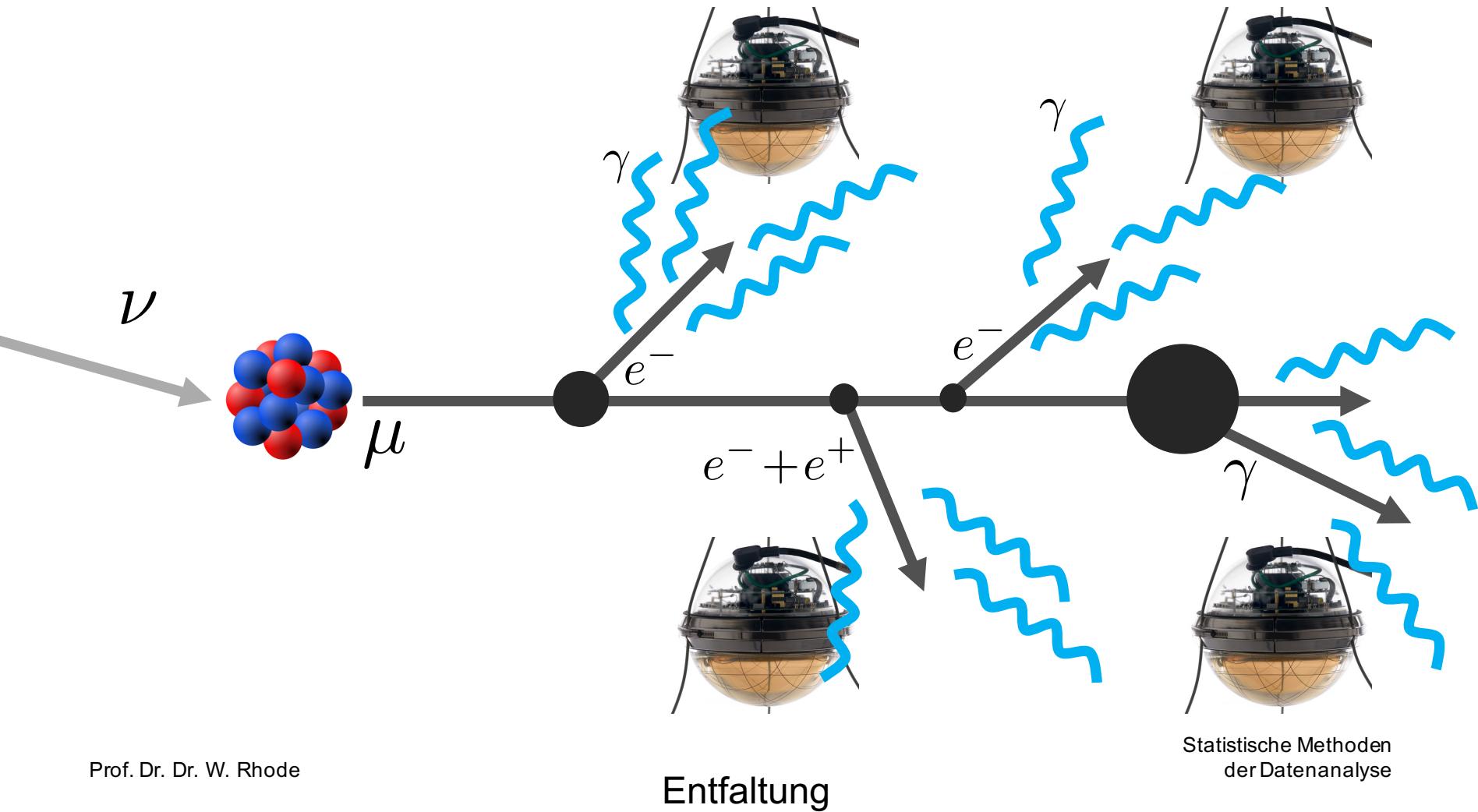
Beispiel A - Messprozess



Beispiel B – Messung bei IceCube (Fortsetzung)



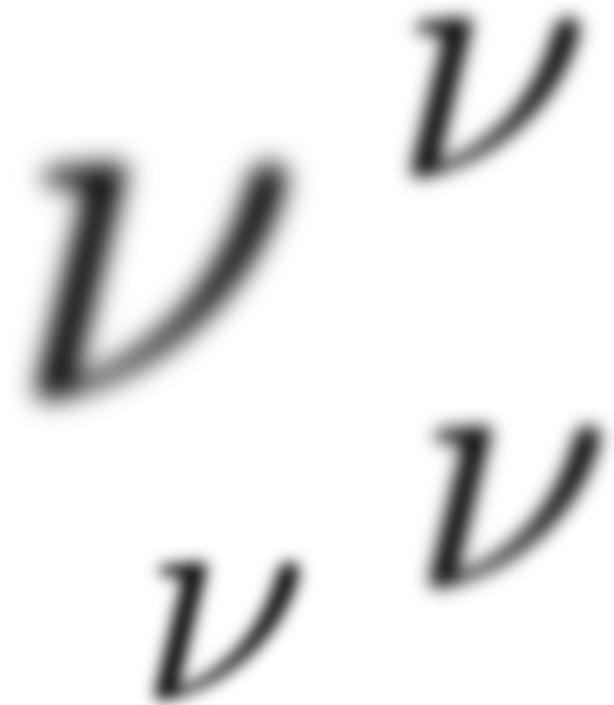
Beispiel B – Messung bei IceCube



Beispiel B – Messung bei IceCube (Fortsetzung)



Begrenzte Auflösung

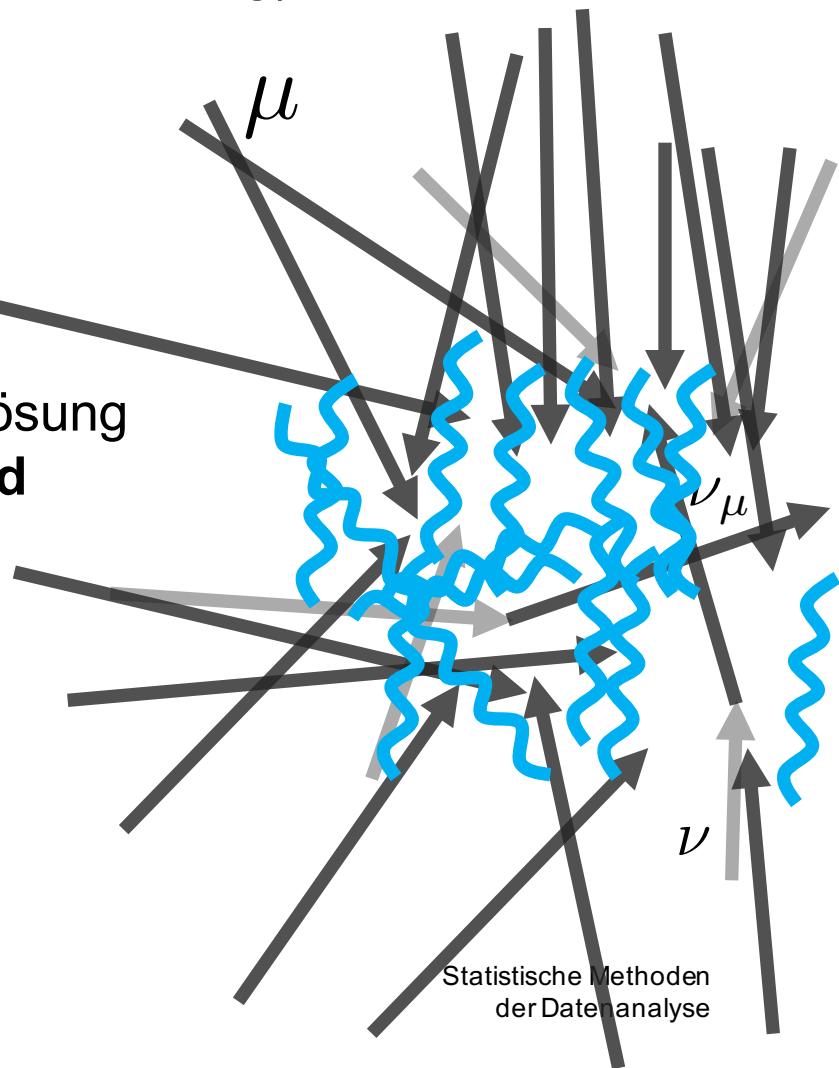


Beispiel B – Messung bei IceCube (Fortsetzung)



Begrenzte Auflösung
Untergrund

Entfaltung

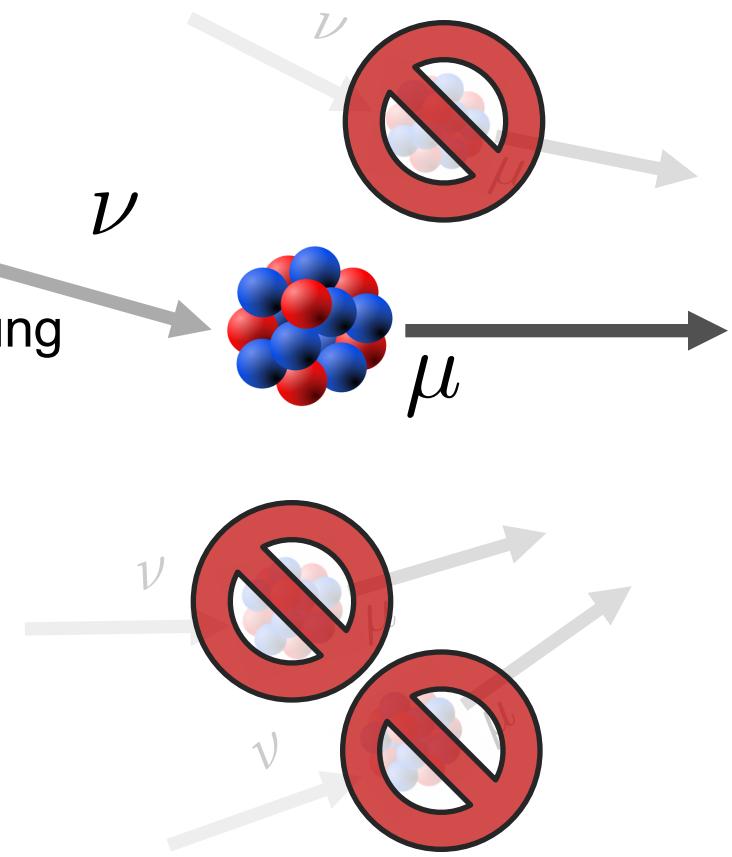


Beispiel B – Messung bei IceCube (Fortsetzung)



Begrenzte Auflösung
Untergrund
Akzeptanz

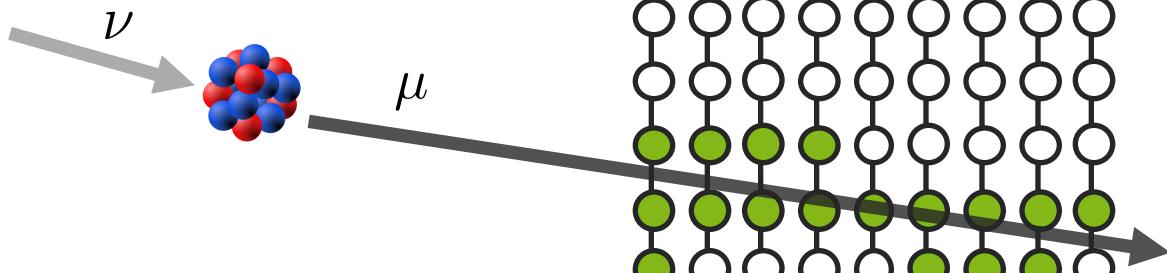
Entfaltung



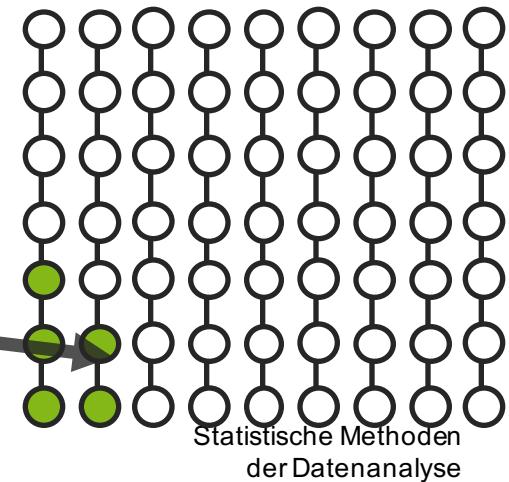
Beispiel B – Messung bei IceCube (Fortsetzung)



Begrenzte Auflösung
Untergrund
Akzeptanz
Unterschiedliche Signatur



Entfaltung



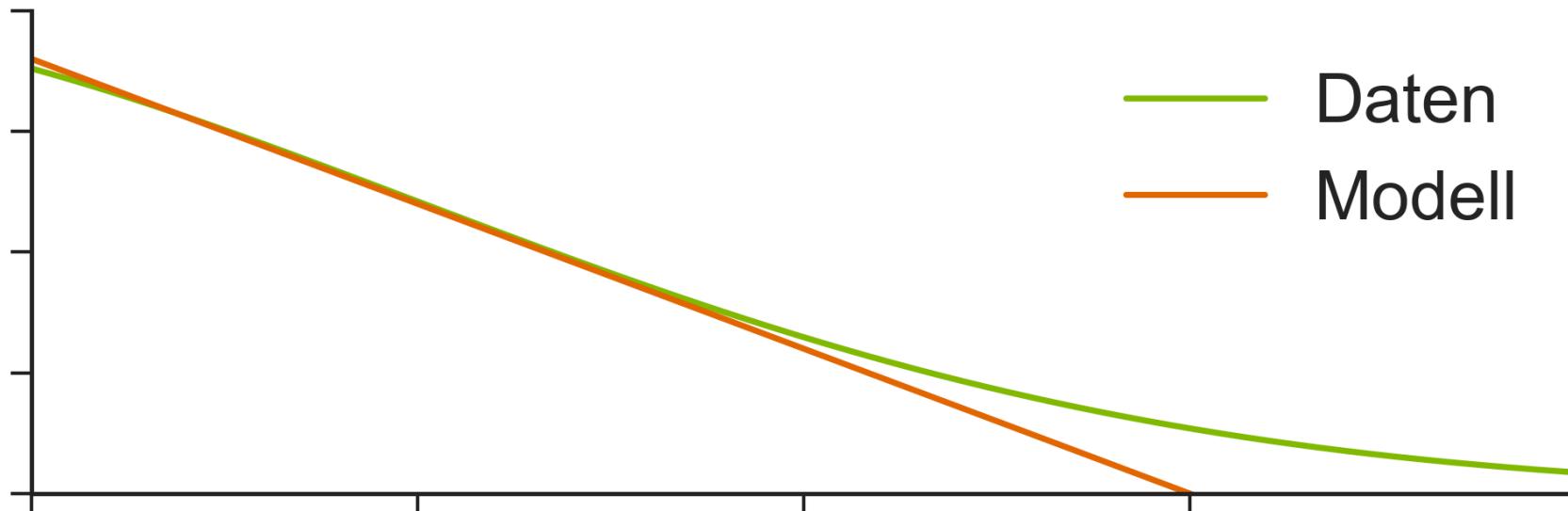
Entfaltung

**Die Rekonstruktion der „wahren“ Verteilung $f(x)$ aus der
gemessenen Verteilung $g(y)$ heißt Entfaltung**

- Entfaltung ist ein **inverses** Problem
 - Schließe von der Messung auf die wahre Verteilung
- Entfaltung ist ein **statistisches** Problem
 - Rekonstruktion fehlerbehafteter Größen und statistischer Prozesse
- Entfaltung ist ein **schlecht konditioniertes** Problem
 - Kleine Fehler in den Anfangsdaten bedingen große Fehler in der Lösung
 - Es kann zu stark oszillierenden, unbrauchbaren Lösungen kommen

Entfaltung – Warum nicht einfach fitten?

- Wenn das Modell bekannt ist, kann ein Fit gemacht werden
 - Liefert die Modell-Parameter, die die Daten am besten beschreiben
- Methode heißt auch „*Forward Folding*“ oder „*Parametrized Unfolding*“
- **Nachteil: Modellabhängigkeit**
 - Wenn das Modell falsch ist, bringt auch der Fit nicht viel



Entfaltung – Bin-by-bin Korrektur?

- Vergleiche die einzelnen Bin-Inhalte eines durch eine Monte Carlo Simulation erhaltenen Histogramms nach den gleichen Analyseschritten wie auf Daten mit der Monte-Carlo Wahrheit
 - Korrekturfaktor $C_i = MC \text{ Wahrheit} / MC \text{ nach Rekonstruktion}$
- Wende den Korrekturfaktor auf das gemessene Histogramm an
- **Probleme:**
 - Berücksichtigt keine Migration in andere Bins („Verschmierung“)
 - Im Prinzip reine Akzeptanzkorrektur
- *“... a HEP-specific heuristic, called bin-by-bin unfolding, which **provably accounts for smearing effects incorrectly** through a multiplicative efficiency correction, is widely used.” [V.M. Panaretos]*

Entfaltung – Vorteil der Entfaltungsmethode

- Die hier vorgestellte Entfaltungsmethode basiert darauf, den gesamten Messprozess durch Monte Carlo zu simulieren
 - Beschreibung durch linearen Operator basierend auf Monte Carlo
- Schließe von gemessenen Daten auf die wahre Verteilung durch Invertierung des linearen Operators
- Diese Methode hat einige **Vorteile**:
 - **Modellunabhängigkeit**
 - Die rekonstruierte Verteilung ist unabhängig von der gewählten Monte Carlo Wahrheit
 - Berücksichtigt **alle simulierten Messprozesse / Detektoreffekte**
 - Die Simulation der Messprozesse muss dafür genau bekannt sein
- Nicht geeignet für die Auflösung von Strukturen unterhalb der Detektorauflösung
- Gut geeignet für Schätzung

FALTUNGSINTEGRAL

Fredholm Integral Gleichung

- Gegeben
 - N Messdaten $\{y\}_N$ beschrieben durch die Verteilung $g(y)$
 - Untergrund beschrieben durch Verteilung $b(y)$
- Gesucht
 - Verteilung des „wahren“ Parameters x , beschrieben durch $f(x)$
- Annahme:
Messprozess wird durch einen linearen Operator A beschrieben: $A f = g$
 - Superposition: $A(f_1 + f_2) = A f_1 + A f_2$
 - Skalierung: $A(a \cdot f) = a \cdot A f$
- Fredholm Integral Gleichung (Faltungsintegral)
$$\int_{\Omega} A(x, y) f(x) dx + b(y) = g(y)$$

Fredholm Integral Gleichung (Fortsetzung)

- Fredholm Integral Gleichung (Faltungsintegral)

$$\int_{\Omega} A(x, y) f(x) dx + b(y) = g(y)$$

- Kern, Antwort- oder **Response-Funktion A(x,y) beschreibt den gesamten Messprozess**
- Untergrund kann (z.B. durch maschinelle Lernverfahren) subtrahiert werden

→ Im Folgenden wird das Problem beschrieben durch

$$\int_{\Omega} A(x, y) f(x) dx = g(y)$$

Diskretisierung

- Für numerische Lösungen muss vorher diskretisiert werden
 - Diskrete Variablen aus Monte Carlo Simulation und Messung
 - Z.B. durch Repräsentation als Histogramm
 - Später im Programm Truee z.B.: Diskretisierung von f durch Splines
- Anstelle der Funktionen f, g treten Vektoren mit den Elementen
 - f_j mit $j = 1, \dots, n$
 - g_i mit $i = 1, \dots, m$
- Die Response-Funktion A wird zu einer $m \times n$ Matrix mit den Einträgen A_{ij}
- Die Integral Gleichung lautet dann diskret

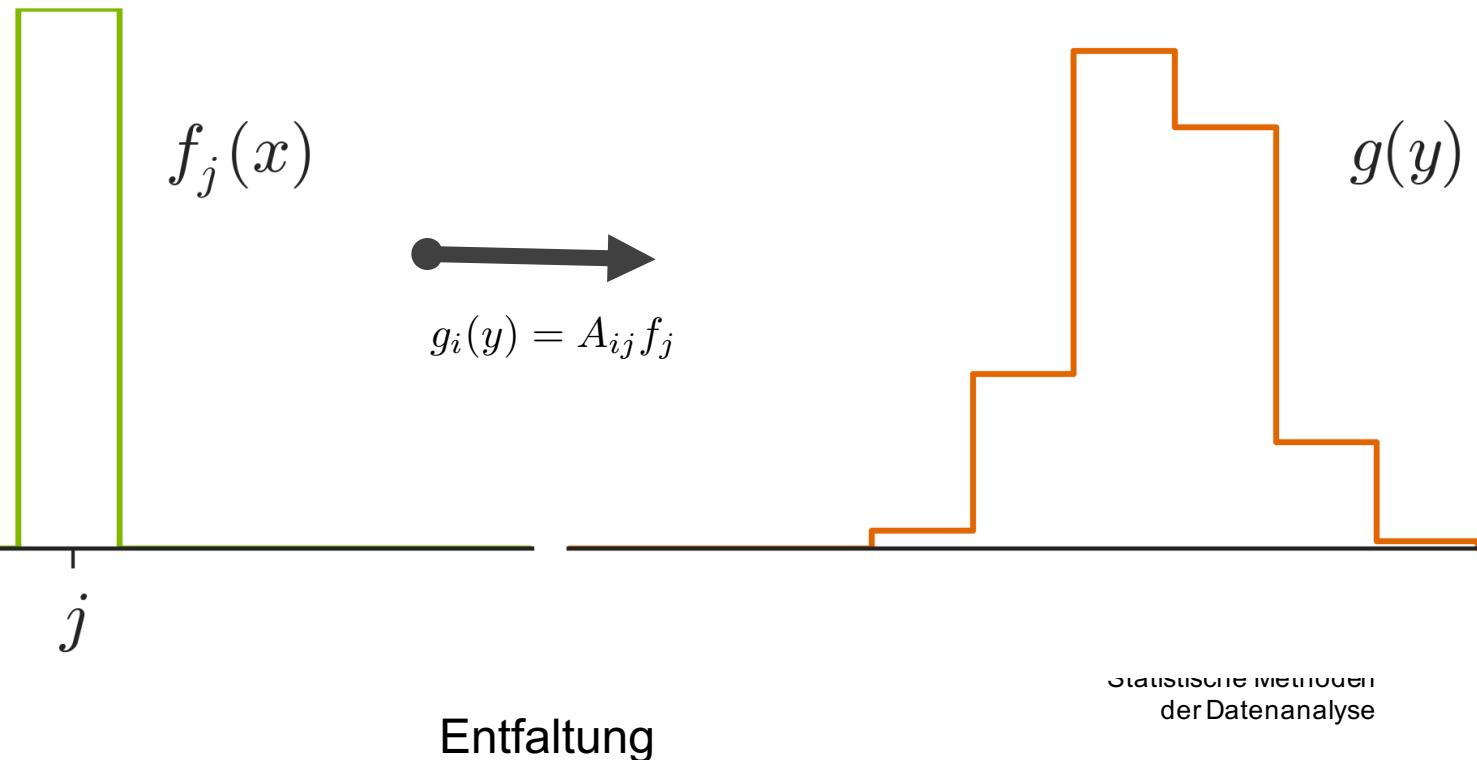
$$g_i = \sum_{j=1}^n A_{ij} f_j$$

Responsematrix

- Durch die Responsematrix (oder Kernel) wird der **gesamte Messprozess** beschrieben
- Bildet die wahre Verteilung f auf die gemessene Verteilung g ab
- **A wird durch Monte-Carlo Simulation bestimmt**
 - Ausgangspunkt: Bekannte Verteilung $f(x)$
 - Z.B. Spektrum der kosmischen Strahlung $\sim E^{-\gamma}$
 - Physikalische Einflüsse / Messprozesse werden durch MC beschrieben
 - Statische Prozesse
- MC Simulation liefert die Verteilung $g(y)$
→ Migration von Bin j der “wahren” Verteilung in Bin i der gemessenen Verteilung ist bekannt
- **Element A_{ij} ist die Wahrscheinlichkeit ein Ereignis x aus Bin j nach dem Messprozess als y im Bin i zu finden**

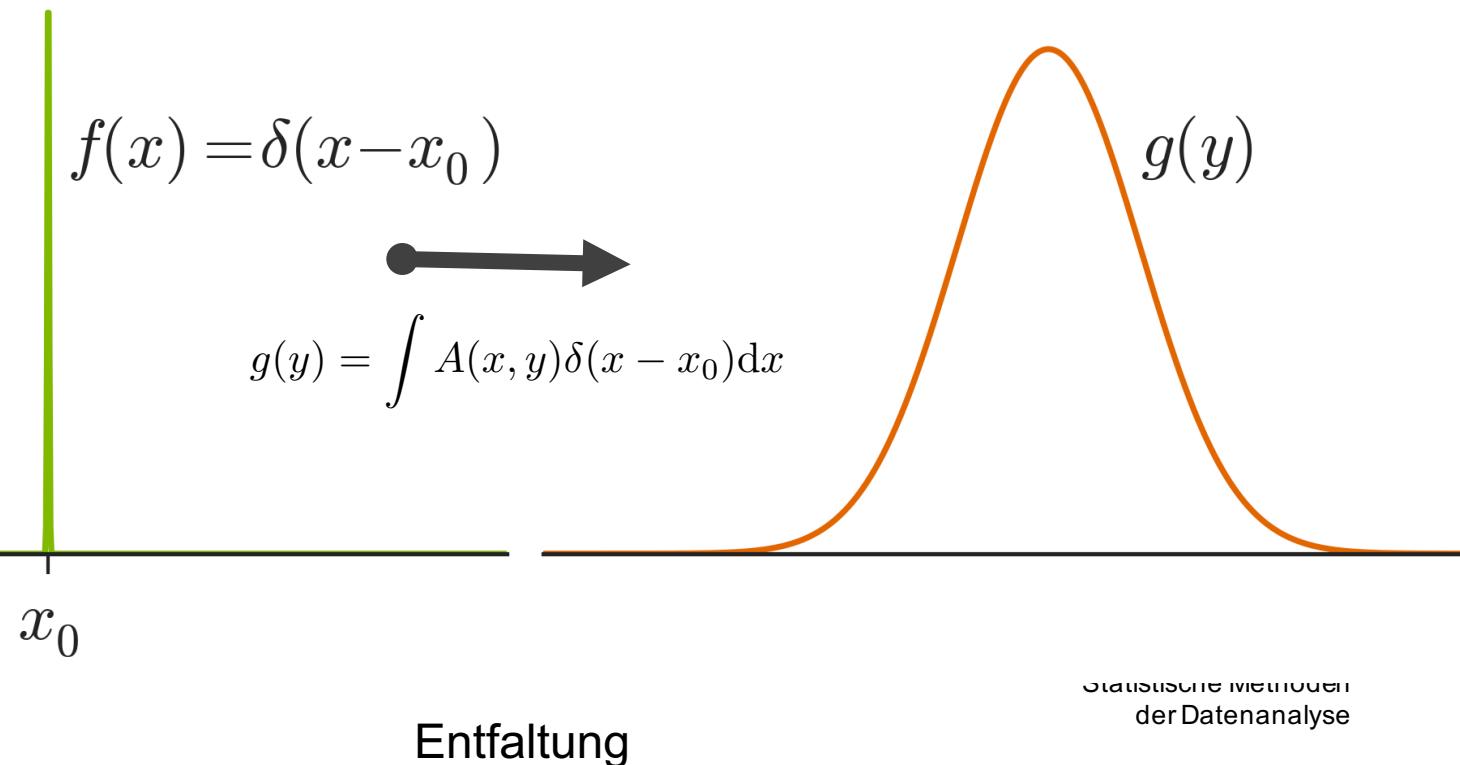
Responsematrix (diskret)

- Jede Spalte j der Responsematrix A_{ij} entspricht der Detektorantwort für ein „wahres“ Event x aus Bin $j \rightarrow$ Migrations-Wahrscheinlichkeit in Bin i der gemessenen Verteilung



Responsefunktion (kontinuierlich)

- Responsefunktion $A(x=x_0, y)$ entspricht der Detektorantwort für einen isolierten Wert $x_0 \rightarrow$ Detektorantwort auf δ -Funktion



Beispiel - Responsematrix

- Die Matrix beschreibt hier nur eine **Verschmierung in die jeweiligen Nachbar-Bins mit Wahrscheinlichkeit ϵ**
 - Mit Wahrscheinlichkeit $1-2\epsilon$ bleibt das Ereignis im gleichen Bin
 - A ist hier quadratisch → Gleiches Binning von f, g

$$A = \begin{pmatrix} 1 - \epsilon & \epsilon & 0 & 0 & \dots & 0 \\ \epsilon & 1 - 2\epsilon & \epsilon & 0 & \dots & 0 \\ 0 & \epsilon & 1 - 2\epsilon & \epsilon & \dots & 0 \\ \vdots & & & \ddots & & \vdots \\ 0 & \dots & 0 & \epsilon & 1 - 2\epsilon & \epsilon \\ 0 & \dots & 0 & 0 & \epsilon & 1 - \epsilon \end{pmatrix}$$

ENTFALTUNGSPROBLEM

Einleitung

- In diesem Abschnitt geht es um konzeptionelle Probleme, die bei dieser Entfaltungsmethode auftreten
- Probleme werden anhand eines Beispiels mit quadratischer Responsematrix diskutiert
 - Was passiert, wenn naiv die Faltungsgleichung invertiert wird?
 - Wo kommen die Probleme mit diesem Ansatz her?
 - Wie kann ich die Probleme umgehen?
- Vorarbeit, um zu verstehen, was bestimmte Einstellungen in TRUEE bewirken

Lösung durch Invertierung - Spezialfall quadratischer Matrizen

- Im Spezialfall $\dim(A) = n \times n$ kann die Matrixgleichung durch einfache Invertierung gelöst werden

$$f_{\text{unf}} = A^{-1}g$$

- Dieses Verfahren liefert eine konsistente Schätzung

$$E[f_{\text{unf}}] = E[A^{-1}g] = A^{-1}E[g] = A^{-1}Af = f$$

- Und eine Fehlerabschätzung durch die Kovarianzmatrix

$$V[f_{\text{unf}}] = A^{-1}V[g](A^{-1})^T$$

- Allerdings: Lösung durch Invertierung führt zu starken Oszillationen
 - Problem ist schlecht konditioniert

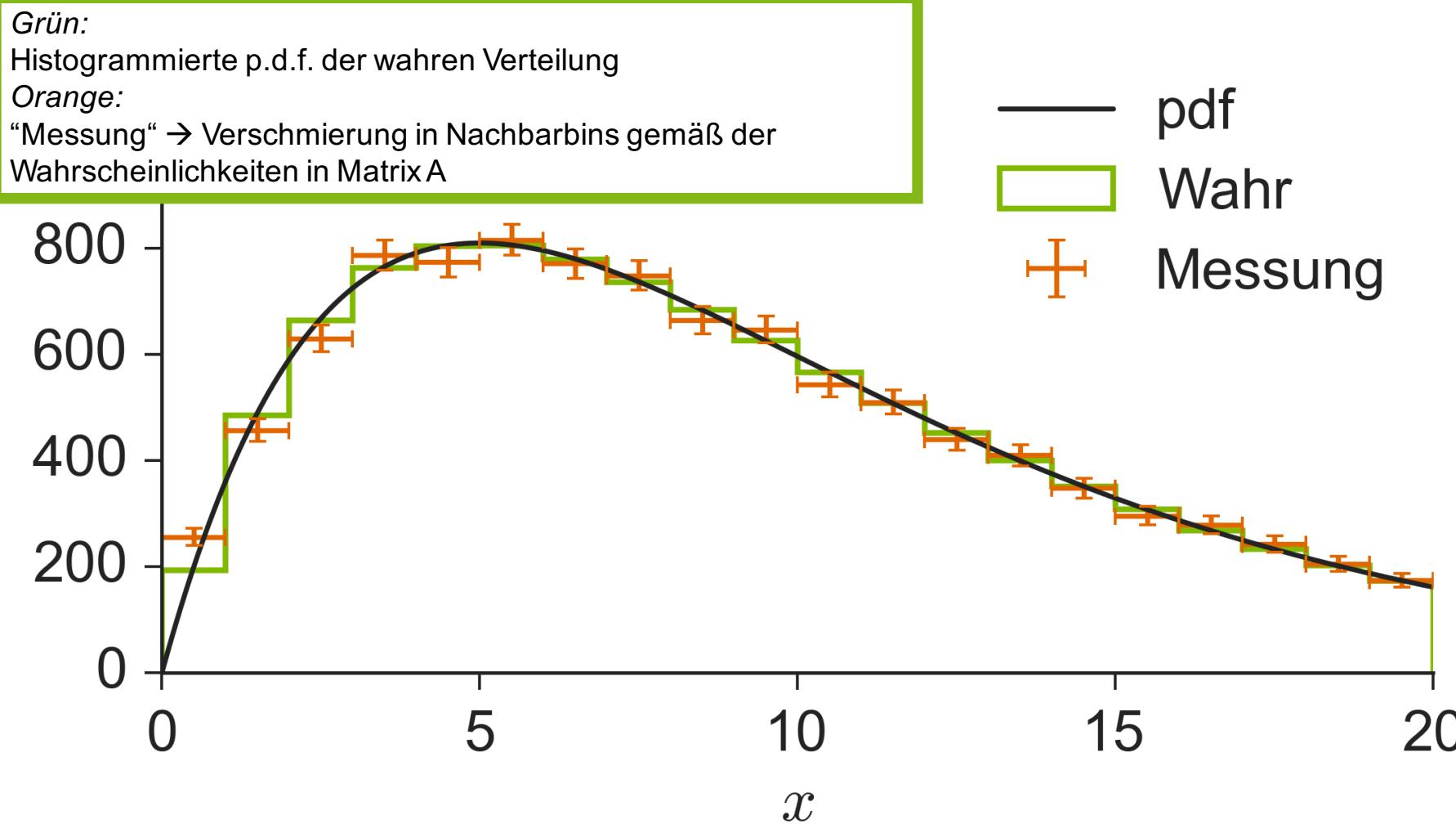
Beispiel - Lösung durch Invertierung - Quadratische Matrizen

- P.d.f. der wahren Verteilung ist hier $f(x) = N \cdot x e^{-ax}$
- Responsematrix: Verschmierung in Nachbar-Bins mit Wahrscheinlichkeit ϵ

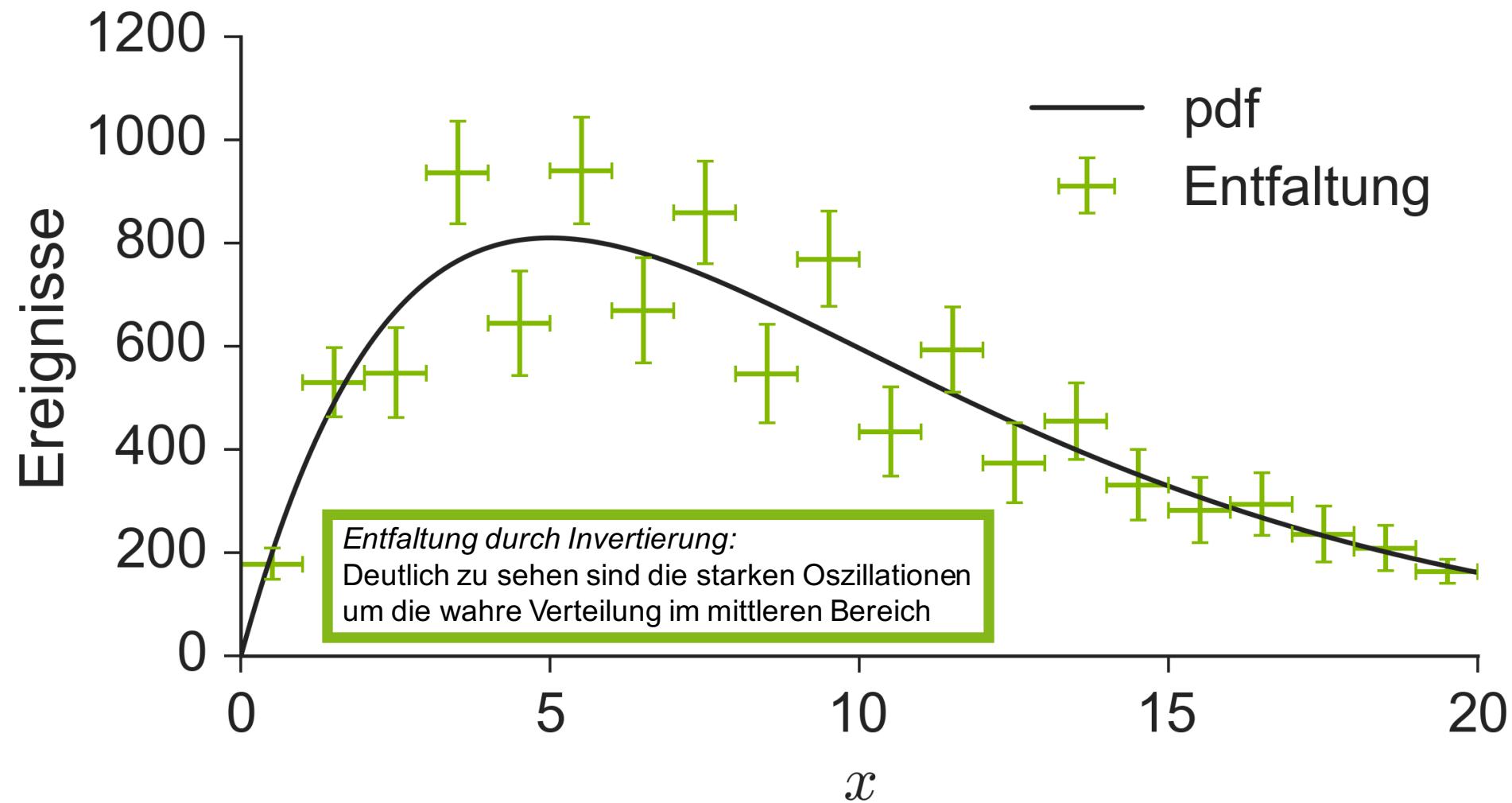
$$A = \begin{pmatrix} 1 - \epsilon & \epsilon & 0 & 0 & \dots & 0 \\ \epsilon & 1 - 2\epsilon & \epsilon & 0 & \dots & 0 \\ 0 & \epsilon & 1 - 2\epsilon & \epsilon & \dots & 0 \\ \vdots & & & \ddots & & \vdots \\ 0 & \dots & 0 & \epsilon & 1 - 2\epsilon & \epsilon \\ 0 & \dots & 0 & 0 & \epsilon & 1 - \epsilon \end{pmatrix}$$

- Hier je 20 Bins für $f, g \rightarrow \dim(A) = 20 \times 20$
- Entfaltung durch einfache Invertierung: $f_{\text{unf}} = A^{-1} g$

Beispiel - Lösung durch Invertierung (Fortsetzung)



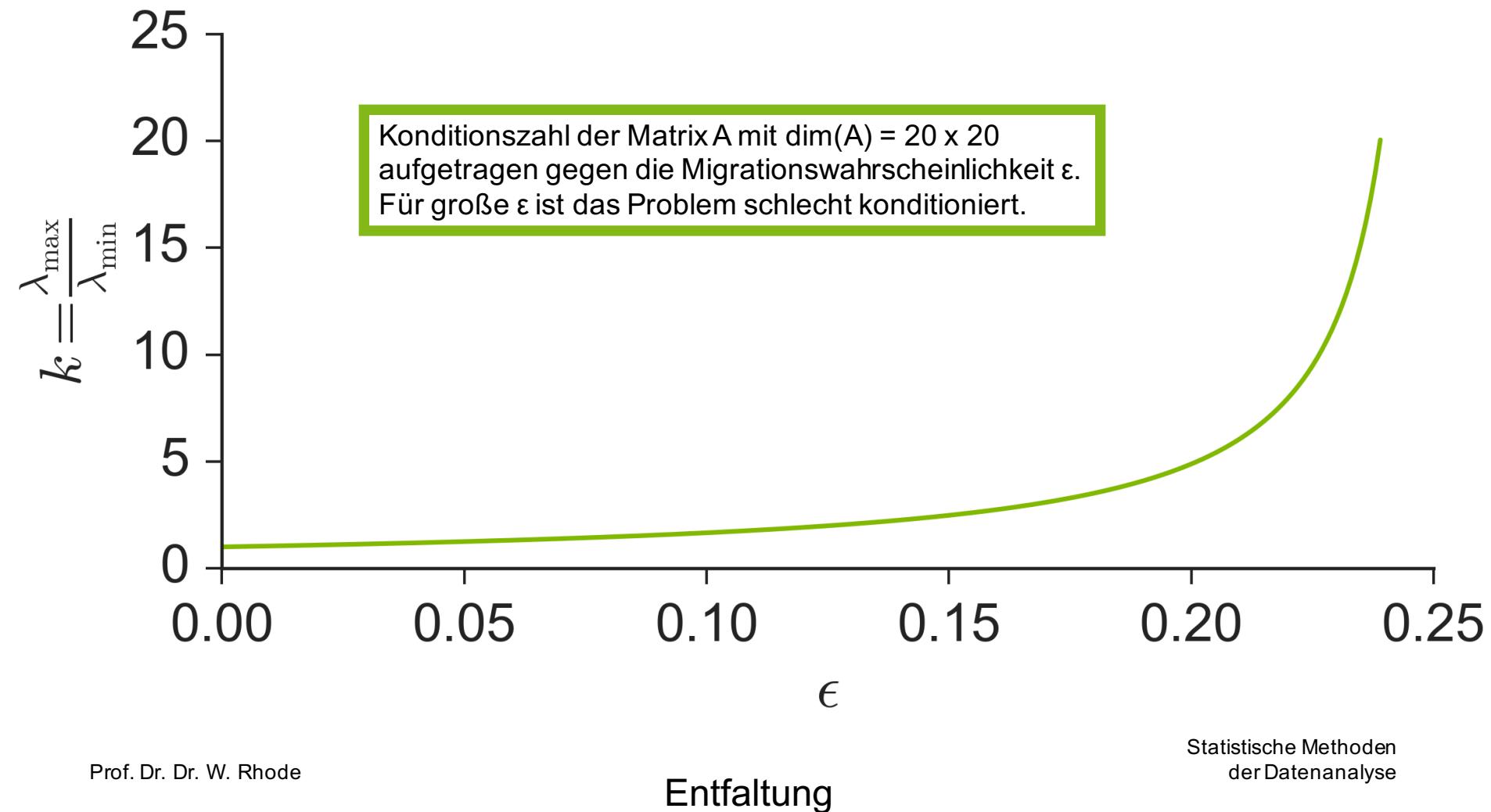
Beispiel - Lösung durch Invertierung (Fortsetzung)



Eigenwerte der Responsematrix

- Im Beispiel gesehen: Naiver Lösungsansatz führt zu unbrauchbaren Ergebnissen
→ Woher kommen die Oszillationen?
- Betrachte Matrix A aus dem Beispiel
 - Konditionszahl $k = \text{Größter Eigenwert} / \text{kleinster Eigenwert von } A$
 - Für größere ϵ wird die Konditionszahl schnell größer (siehe Plot)
→ Hohe Verstärkung anfänglicher Störungen
- *Erinnerung:* Kondition ist eine Eigenschaft des Problems und beschreibt die Verstärkung des Fehlers bei kleinen Störungen der Eingangsdaten

Eigenwerte der Responsematrix - Konditionszahl



Eigenwerte der Responsematrix – Eigenwertzerlegung

- **Wir wollen den Einfluss der Eigenwerte auf die Entfaltung verstehen**
→ Transformation der Faltungsgleichung in die Eigenbasis von A

$$A = UDU^{-1}$$

- D ist diagonal mit den Eigenwerten λ von A auf der Diagonalen
- U ist die Transformationsmatrix in das Eigensystem von A
 - Spaltenvektoren in U sind die Eigenvektoren von A mit $U^{-1}U=1$
- Die Faltungsgleichung ist in der neuen Basis dann

$$g = Af = UDU^{-1}f \Leftrightarrow \underbrace{U^{-1}g}_{:=c} = D \underbrace{U^{-1}f}_{:=b}$$

Eigenwerte der Responsematrix – Eigenwertzerlegung (Forts.)

- Es ergeben sich die neuen Vektoren b, c in der Diagonalbasis zu

$$f \rightarrow b = U^{-1}f \Leftrightarrow f = Ub \quad \text{und} \quad g \rightarrow c = U^{-1}g \Leftrightarrow g = Uc$$

- Und die **Faltungsgleichung in der Eigenbasis von A** wird zu

$$c = Db$$

- Da D diagonal ist, wird jeder Eintrag b_j, c_j **unabhängig transformiert**

$$\text{Faltung } f \rightarrow g : \quad b_j \rightarrow b_j \lambda_j = c_j$$

$$\text{Entfaltung } g \rightarrow f : \quad c_j \rightarrow \frac{c_j}{\lambda_j} = b_j$$

Eigenwerte der Responsematrix – Eigenwertzerlegung (Forts.)

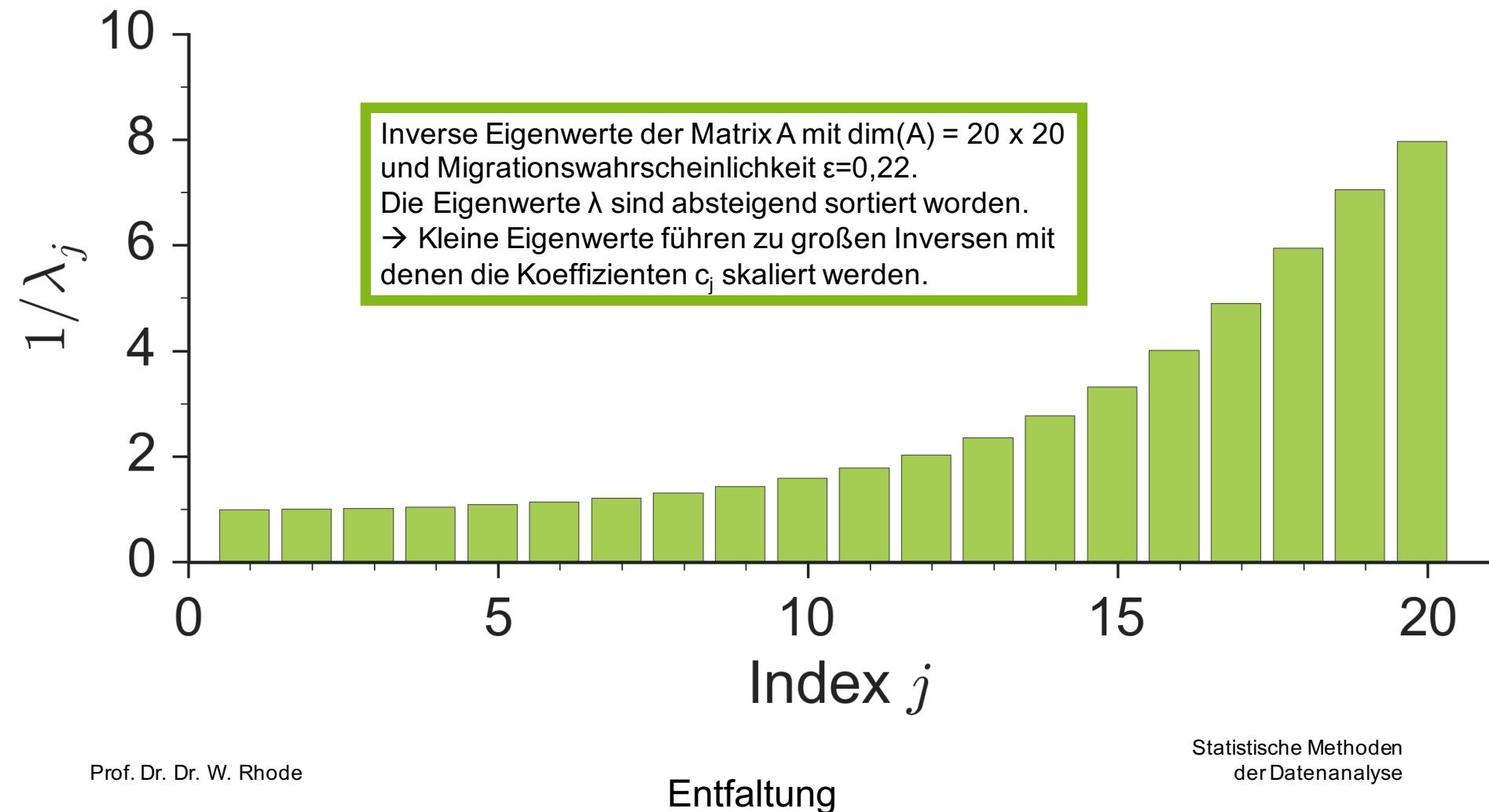
- Das Problem wird in der transformierten Entfaltung deutlich

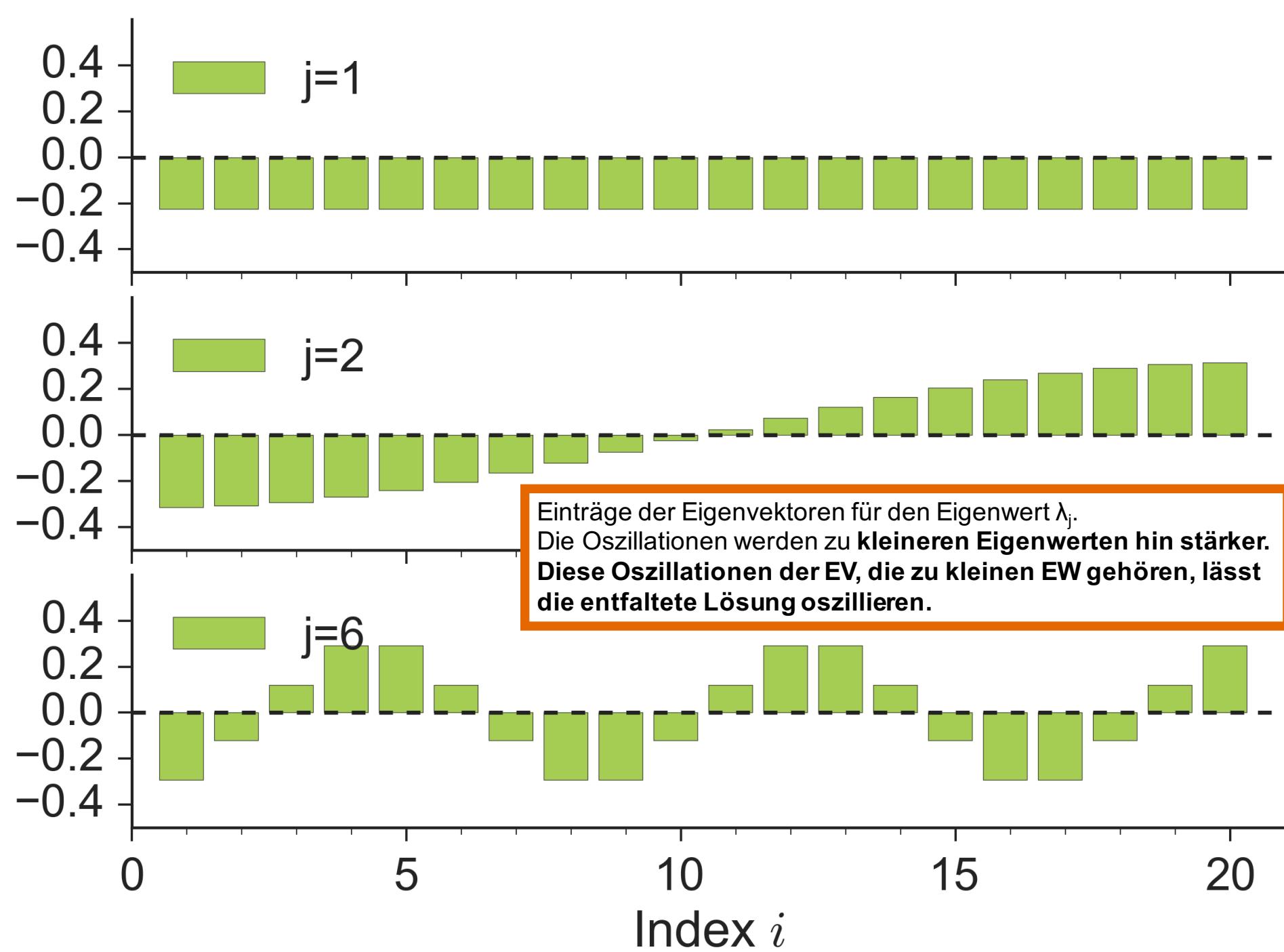
$$\text{Entfaltung } g \rightarrow f : c_j \rightarrow \frac{c_j}{\lambda_j} = b_j$$

- Kleine Eigenwerte λ verstärken die Koeffizienten c_j

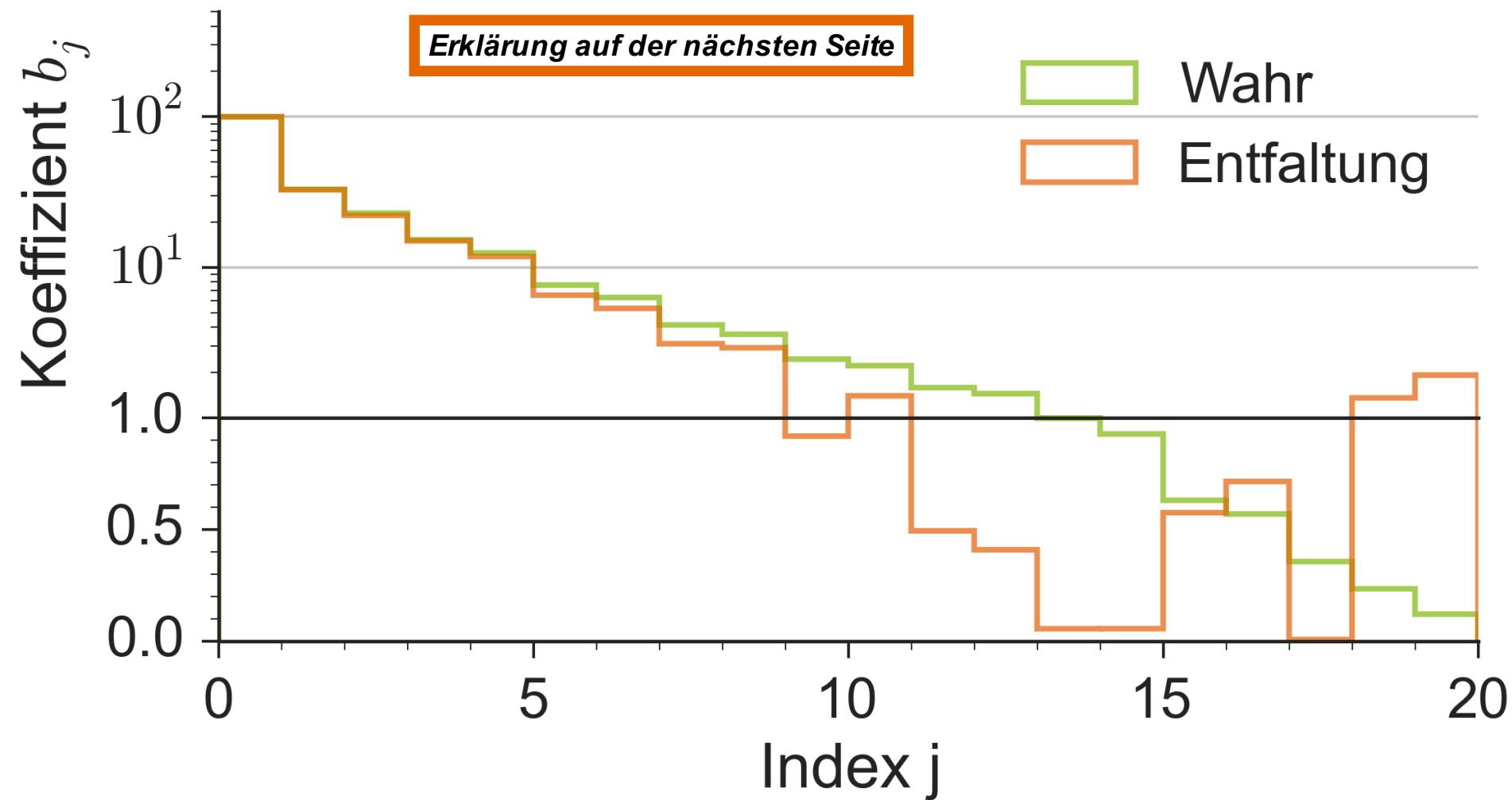
- Problem: Koeffizienten c_j stammen aus einer Messung und sind mit Fehlern behaftet
→ Kleine Eigenwerte λ verstärken statistische Fluktuationen in den Koeffizienten c_j
- Für sehr kleine Eigenwerte wird die entfaltete Verteilung $f_{\text{unf}} = \text{Ub}$ von wenigen Koeffizienten b_j dominiert sein, die aus kleinen Eigenwerten und hohen statistischen Fehlern stammen
 - Die Lösung fängt an zu oszillieren

Beispiel – Inverse Eigenwerte der Responsematrix





Beispiel – Inverse Eigenwerte der Responsematrix (Forts.)



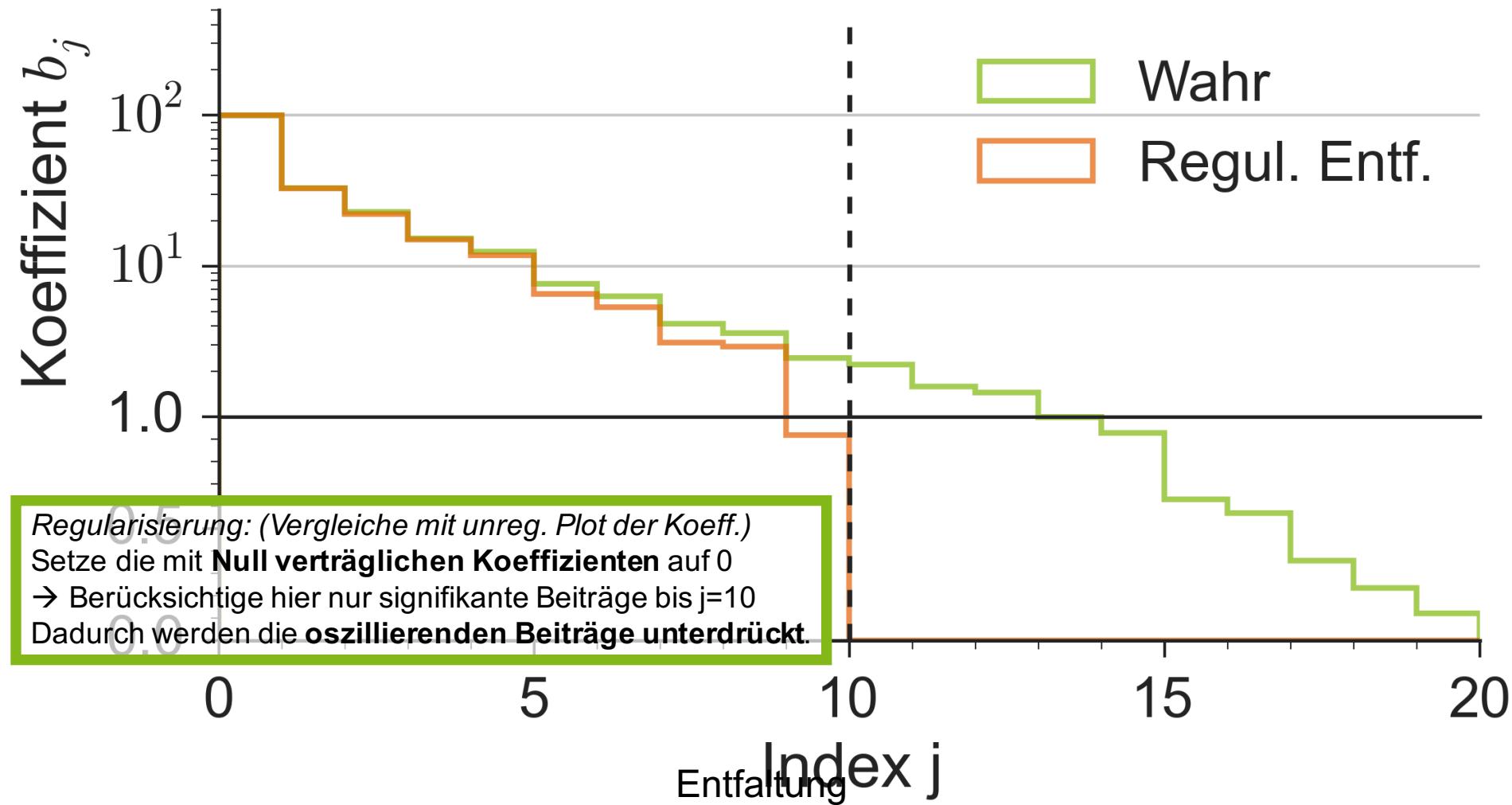
Beispiel – Inverse Eigenwerte der Responsematrix (Forts.)

- Gezeigt sind die Koeffizienten b_j jeweils für die wahre Verteilung (grün) und die entfaltete Verteilung (orange)
 - Koeffizienten sind skaliert auf die propagierten Standardabweichungen
- Die Koeffizienten werden erzeugt aus den Trafos
 - Wahre Verteilung: $b = U^{-1} f$
 - Entfaltete Verteilung: $b_{\text{unf}} = D^{-1} c$
- Durch die Skalierung auf die Standardabweichung markiert die Eins die 1σ Grenze der statistischen Fehler auf die entfalteten Koeffizienten
 - **Ab einem bestimmten Koeffizienten j fluktuiieren die Koeffizienten b_j zufällig um 0 mit der Standardabweichung 1**
- Diese **Werte bieten keinerlei Information** und sollten in der Lösung nicht berücksichtigt werden, daher wird regularisiert
→ **Regularisierung = Abschneiden dieser Eigenwerte**

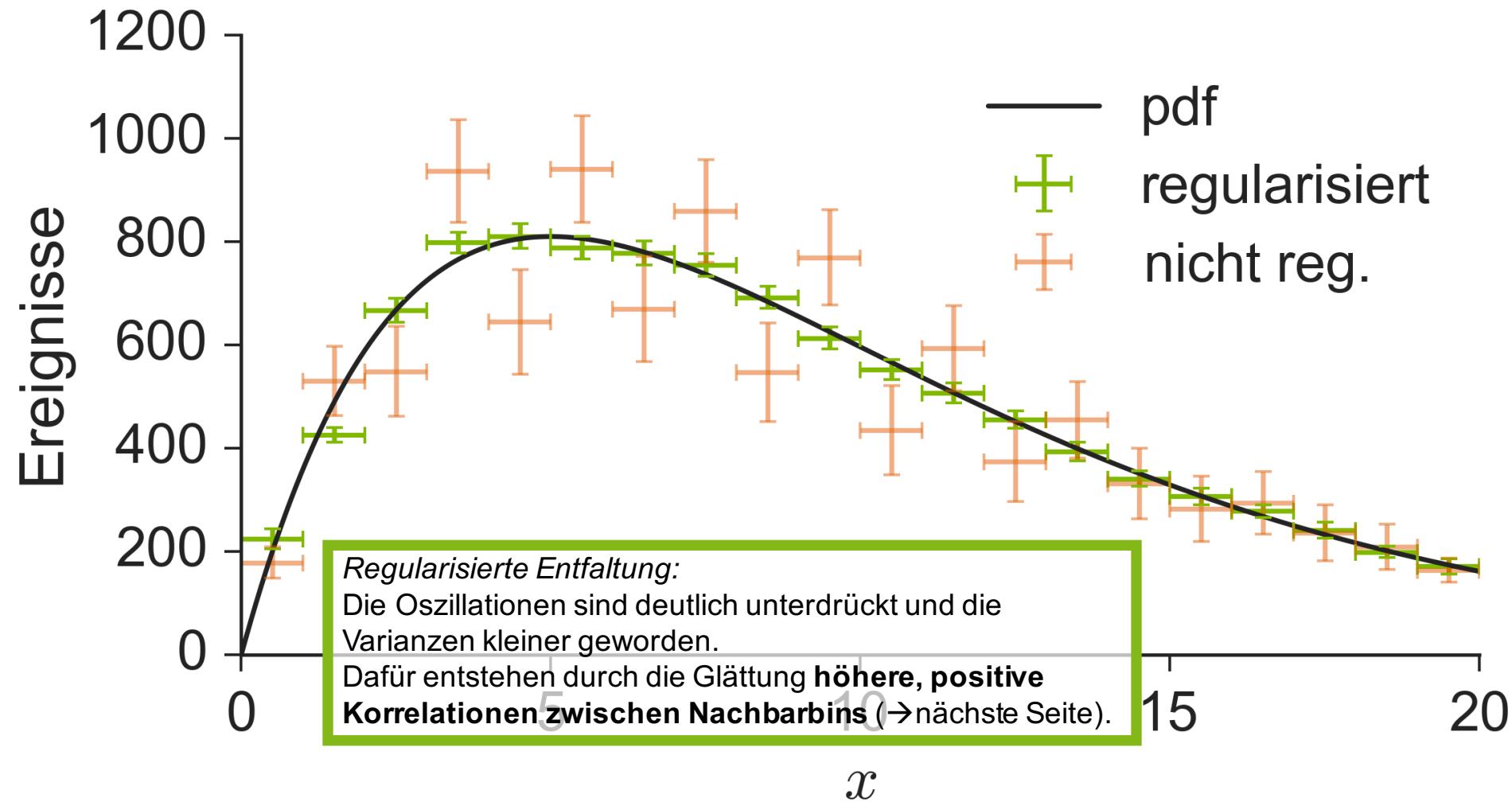
Regularisierung

- Bereits bekannt aus dem „Schätzen“-Teil
- Es werden **zusätzliche Information** in die Lösung mit eingebracht
- Gefordert wird hier z.B., dass die Lösung „glatt“ sein soll und nicht stark oszilliert
- Durch **Abschneiden der kleinen Eigenwerte werden die Oszillationen unterdrückt**
→ Siehe Plots auf den nächsten Seiten

Regularisierung – Abschneiden der Eigenwerte



Regularisierung – Abschneiden der Eigenwerte (Fortsetzung)



Regularisierung – Abschneiden der Eigenwerte (Fortsetzung)

- Die Korrelationsmatrix unregularisiert: Negative Nachbarbinkorrelation
 - Hier nur $\dim(A) = 5$ damit die Matrizen hierhin passen. Cutoff bei $j=3$

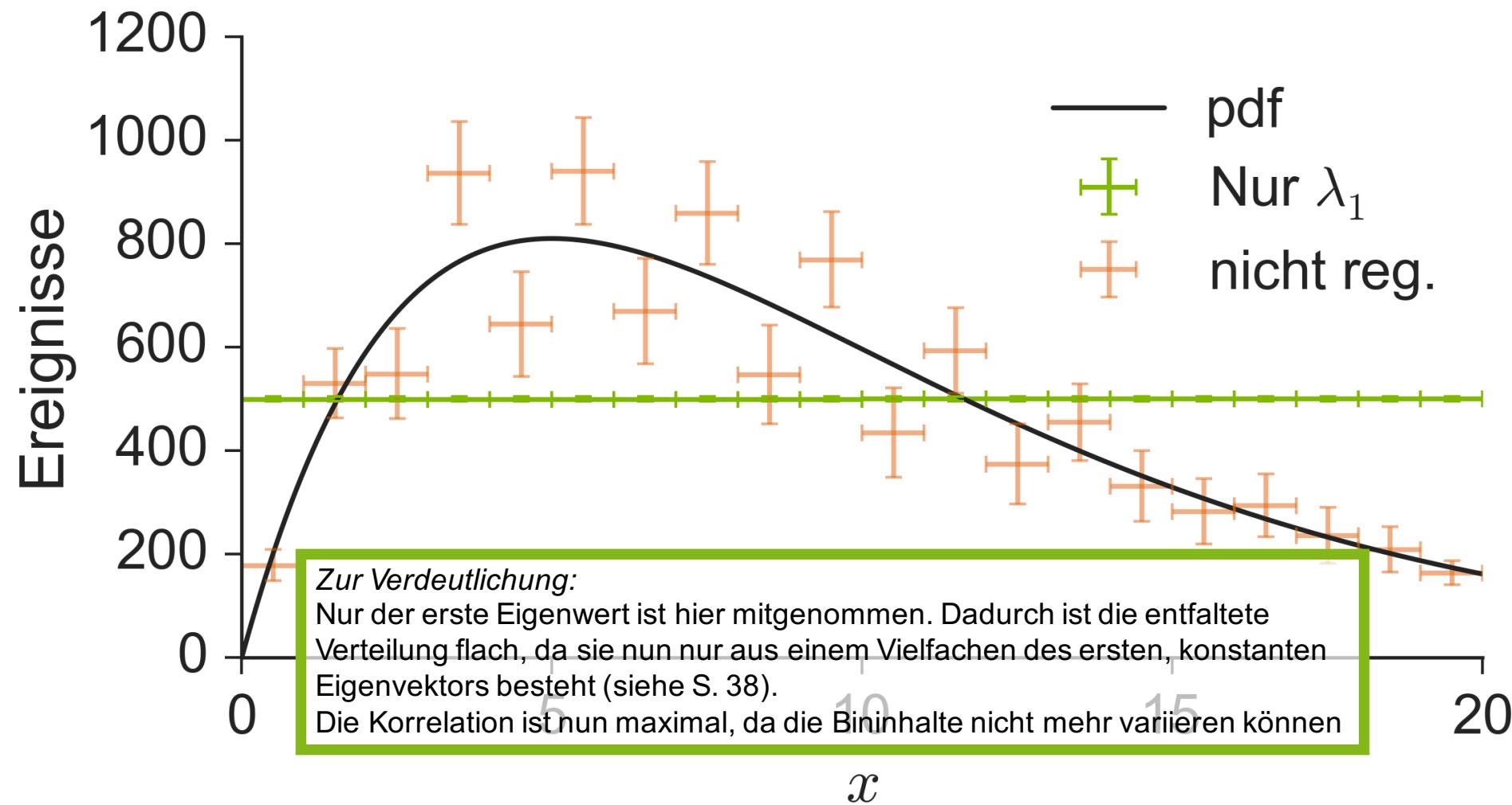
$$\text{Corr}[f_{\text{unf}}] = \begin{pmatrix} 1 & -0,7 & 0,5 & -0,3 & 0,2 \\ -0,7 & 1 & -0,8 & 0,5 & -0,3 \\ 0,5 & -0,8 & 1 & -0,8 & 0,5 \\ -0,3 & 0,5 & -0,8 & 1 & -0,7 \\ 0,2 & -0,3 & 0,5 & -0,7 & 1 \end{pmatrix}$$

- Regularisiert: Positive Korrelation durch Einschränkung der Lösung

$$\text{Corr}[f_{\text{unf}}] = \begin{pmatrix} 1 & 0,4 & -0,4 & -0,4 & 0,3 \\ 0,4 & 1 & 0,7 & 0,4 & -0,4 \\ -0,4 & 0,7 & 1 & 0,8 & -0,5 \\ -0,4 & 0,4 & 0,8 & 1 & 0,1 \\ 0,3 & -0,4 & -0,5 & 0,1 & 1 \end{pmatrix}$$

Statistische Methoden
der Datenanalyse

Regularisierung – Abschneiden der Eigenwerte (Fortsetzung)



Regularisierung – Abschneiden der Eigenwerte (Fortsetzung)

- Korrelationsmatrix wenn nur der erste Eigenwert benutzt wird
→ Korrelation überall 1
 - Es gibt nur noch einen Freiheitsgrad, jedes Bin hat den gleichen Inhalt
 - Hier auch wieder für $\dim(A) = 5$

$$\text{Corr}[f_{\text{unf}}] = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Zusammenfassung bisher

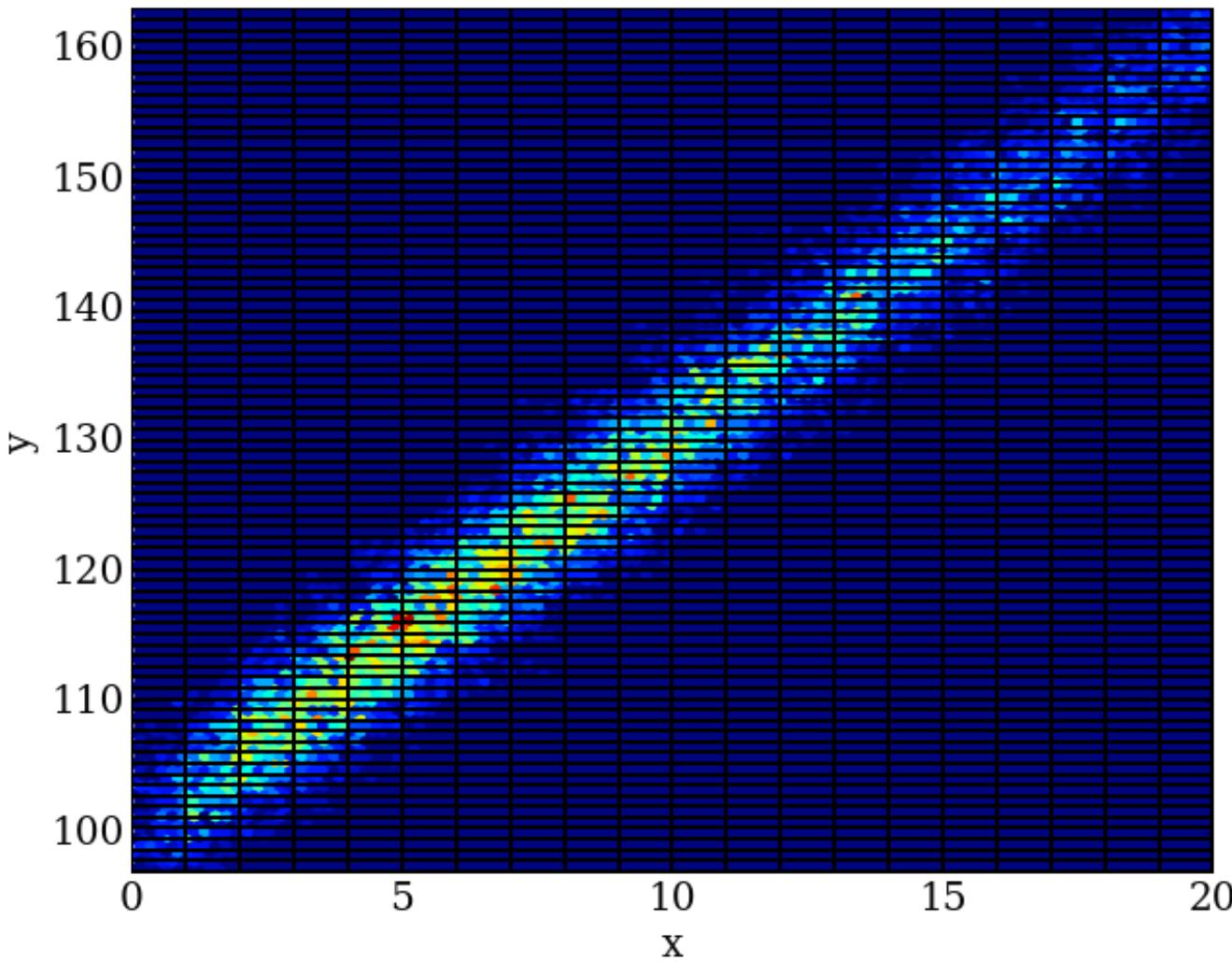
- Naive Invertierung führt zu **starken Oszillationen**.
- Der Grund sind die **kleinen Eigenwerte von A**, die statistische Schwankungen in der Messung verstärken
- **Regularisierung** verwirft die Koeffizienten, die mit Null verträglich sind und somit keinerlei Information enthalten
- Oszillationen werden unterdrückt und die **Varianzen werden kleiner, ABER die Korrelation zwischen den Bins wird dafür größer**
- Einfachste Form der Regularisierung, harter Cutoff
→ Später: „weiches“ Abschneiden der Koeffizienten

LÖSUNG DURCH FIT

Motivation - Lösung mit rechteckiger Responsematrix

- Bisher: Quadratische Matrizen
 - Unnötige Einschränkung → Versuche soviel Information aus der Messung g zu erhalten, wie es die Statistik zulässt
- Wähle ausreichend feines Binning in g
 - Responsematrix wird rechteckig
 - Entfaltung kann durch die zusätzlichen Information verbessert werden
- Nachfolgendes Beispiel: Wahre Verteilung $f(x) = N \times \exp(-ax)$
 - Messung: Aus $f(x)$ zufällig gezogen und mit Gaußverteilung verschmiert

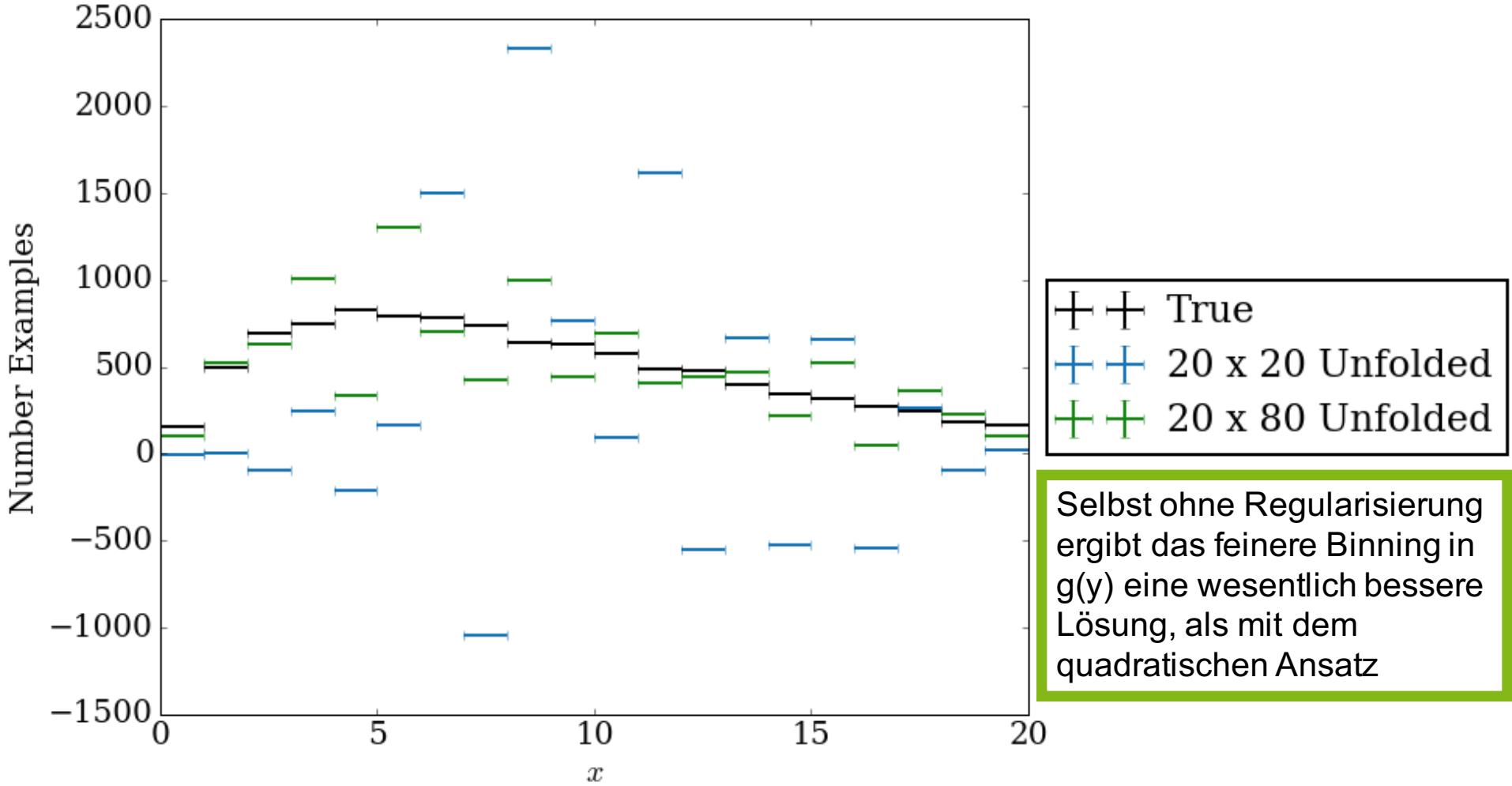
Beispiel – Feineres Binning der Messung



Darstellung der
Responsematrix für
das Problem.
Das Gitter ist das
Binning in $f(x)$, $g(y)$:
20 Bins in $f(x)$
80 Bins in $g(y)$

Beispiel – Feineres Binning der Messung (Fortsetzung)

A: (20 x 20) vs (20 x 80) Unfolding unreg.



Einschub – Singulärwertzerlegung

- Bisher ist alles am Beispiel quadratischer Probleme gezeigt worden
→ Kein Problem, die Argumentation trifft genauso auf allgemeine $m \times n$ Probleme zu
- Erinnerung
 - Quadratische Matrizen können mit $A = U D U^{-1}$ in ihre Diagonalform gebracht werden
 - Die Inverse ergibt sich dann aus $A^{-1} = (U D U^{-1})^{-1} = U D^{-1} U^{-1}$
- Nun: kurzer Einschub, wie **nicht-quadratische Probleme** behandelt werden
→ **Singulärwertzerlegung**

Einschub – Singulärwertzerlegung (Fortsetzung)

- Singulärwertzerlegung: Eine $m \times n$ Matrix M mit Rang r kann **immer** in ein Matrix Produkt der Form

$$M = U S V^H$$

zerlegt werden

- U ist eine unitäre $m \times m$ Matrix mit $U U^H = U^H U = 1$
 - $H = \text{adjungiert} = \text{transponiert} + \text{komplex konjugiert}$
- V^H ist die Adjungierte einer unitären $n \times n$ Matrix V mit $V V^H = V^H V = 1$

Einschub – Singulärwertzerlegung (Fortsetzung)

- Die Matrix S ist eine reelle $m \times n$ Matrix der Gestalt

$$S = \begin{pmatrix} \sigma_1 & & & & & \\ & \ddots & & & & \\ & & \sigma_r & & & \\ \hline & & & \dots & 0 & \dots \\ & \vdots & & & \vdots & \\ \dots & 0 & \dots & \dots & 0 & \dots \\ & \vdots & & & \vdots & \end{pmatrix}$$

- $\sigma_1 \geq \dots \geq \sigma_r > 0$ heißen **Singulärwerte der Matrix M**

Einschub – Singulärwertzerlegung (Fortsetzung)

- Wie funktioniert die Zerlegung?
 - Finde die Matrizen U und V^H und die Singulärwerte

$$M_{m \times n} = U_{m \times m} S_{m \times n} V_{n \times n}^H$$
- Bestimme die Eigenwerte und -Vektoren von $A^H A$ und $A A^H$
 - Die Eigenvektoren von $(A^H A)_{n \times n}$ sind die Spalten von V
 - Die Eigenvektoren von $(A A^H)_{m \times m}$ sind die Spalten von U

- Die Wurzeln der Eigenwerte von $A^H A$ bzw. $A A^H$ sind die Singulärwerte zu den entsprechenden Spaltenvektoren in U, V
 - Singulärwerte kommen absteigend sortiert auf die Diagonale von
 - Beispiel: https://www.ling.ohio-state.edu/~kbaker/pubs/Singular_Value_Decomposition_Tutorial.pdf

Singulärwertzerlegung – Verbindung zu kleinsten Quadraten

- Erinnerung:
Lösung eines überbestimmten linearen Gleichungssystems $m > n$
 - n Unbekannte x und m Gleichungen

$$Ax = b$$

- Minimiere das Quadrat der Residuen r mit

$$r = Ax - b$$

- Dann ist die analytische Lösung gegeben durch

$$x = (A^T A)^{-1} A^T b$$

- Bereits bekannt → siehe Vorlesungsteil „Schätzen“

Singulärwertzerlegung - Pseudoinverse

- **Pseudoinverse** ist definiert als

$$M = USV^H \rightarrow M^+ = VS^+U^H$$

- Wobei $S^+ = \text{diag}(1/\sigma_1, \dots, 1/\sigma_r, 0, \dots, 0)$ ist
- Es kann gezeigt werden, dass die Pseudoinverse A^+ der Matrix A im kleinsten Quadrate Problem ebenfalls die optimale Lösung darstellt

$$x = A^+b$$

- **Beide Ansätze sind somit äquivalent**

Entfaltung - Rechteckige Responsematrizen

- Für rechteckige Responsematrizen bleibt das Prinzip der Entfaltung gleich

$$g_m = A_{m \times n} f_n$$

- Unterschied:

Nutze die Pseudoinverse A^+ statt der normalen Inversen

$$f_{\text{unf}} = A^+ g$$

- Die Regularisierungsmethode bleibt gleich**

- Abschneiden von Singulärwerten ohne statistische Signifikanz

Entfaltung – Lösung durch kleinste Quadrate Fit

- Bereits gezeigt: kleinste Quadrate und Pseudoinverse sind äquivalent
- Entfaltung kann somit auch durch einen Fit mittels kleinsten Quadrate bestimmt werden. Minimiere

$$S = (Ax - b)^T (Ax - b)$$

- Die **Lösung ist analytisch bekannt**
 - Kein Unterschied zur Lösung mittels Singulärwertzerlegung **ohne** Regularisierung

Entfaltung – Gewichtete kleinste Quadrate

- Es können im kleinste Quadrate Fit zusätzlich noch Gewichte der Messung berücksichtigt werden
- Die Gleichung wird dann (unregularisiert) zu

$$S^{\text{reg}} = (Ax - b)^T V_b^{-1} (Ax - b)$$

- Und der Schätzer wird zu

$$x = (A^T V_b^{-1} A)^{-1} A^T V_b^{-1} b$$

Entfaltung – Regularisierter kleinste Quadrate Fit

- Kleinstes Quadrate mit Regularisierungsmatrix Γ
 - Ebenfalls im „Schätzen“-Teil bereits gezeigt

$$S^{\text{reg}} = (Ax - b)^T(Ax - b) + (\Gamma x)^T(\Gamma x)$$

- Der regularisierte Schätzer wird dann zu

$$x^{\text{reg}} = (A^T A + \Gamma^T \Gamma)^{-1} A^T b$$

Entfaltung – Regularisierter kleinste Quadrate Fit (Forts.)

- Oft wird die 2. Ableitung zur Regularisierung benutzt → Matrix C
 - Damit werden **glatte Lösungen ohne Oszillation bevorzugt**
 - Regularisierungsparameter τ steuert die Stärke der Regularisierung

$$S^{\text{reg}} = (Ax - b)^T(Ax - b) + \frac{\tau}{2} x^T(C^T C)x$$

- Es kann gezeigt werden, dass diese Art der Regularisierung die **Koeffizienten b_j allmählich abschwächt**

$$c_j = \frac{1}{1 + \tau S_{jj}} b_j$$

- Die Werte S_{jj} sind die aufsteigend sortierten Eigenwerte der Matrix $S = C^T C$
- **Die größeren Eigenwerte $S_{jj} \gg \tau^{-1}$ sorgen für eine stärker werdende Abschwächung der b_j und bedingen somit eine glatte Funktion f_{unf}**

Entfaltung – Numerischer kleinste Quadrate Fit

- Bisher konnte alles analytisch gelöst werden
 - Bekannte Methoden aus dem „Schätzen“-Teil
- Ein Fit mit der Methode der kleinsten Quadrate kann natürlich auch numerisch durchgeführt werden
→ Kompliziertere Fits müssen ebenfalls numerisch behandelt werden
- Vorteil: **Nebenbedingungen** können mit eingebracht werden
 - Z.B. können nur positive Werte zugelassen werden: $x_{\text{unf},i} \geq 0$
 - Oder die Anzahl der Ereignisse kann fixiert werden: $\sum_i f_{\text{unf},i} = N$

Entfaltung – Lösung durch Likelihood-Fit

- Wenn nur wenig Statistik in den Bins ist, kann die **zugrundeliegende Poissonverteilung durch einen Likelihood Fit berücksichtigt** werden
 - Kleinste Quadrate Methode mit Poisson Fehlern nur gültig, wenn die Statistik ausreichend hoch ist
- Einträge g_i in den gemessenen Bins folgen einer Poissonverteilung
- Die negative Log-Likelihood F wird zu

$$F(f_{\text{unf}}) = \sum_i [g_i^*(f_{\text{unf}}) - g_i \log(g_i^*(f_{\text{unf}}))]$$

- $g_i^* = \sum_j A_{ij} f_{\text{unf},j}$ sind die für den aktuellen Vektor f_{unf} erwarteten Bineinträge
 - Muss numerisch gelöst werden
- Die **Regularisierung** über die zweite Ableitung funktioniert wie bei den kleinsten Quadraten gezeigt: $F \rightarrow F + \frac{\tau}{2} x^T (C^T C) x$