Karlsruher Institut für Technologie

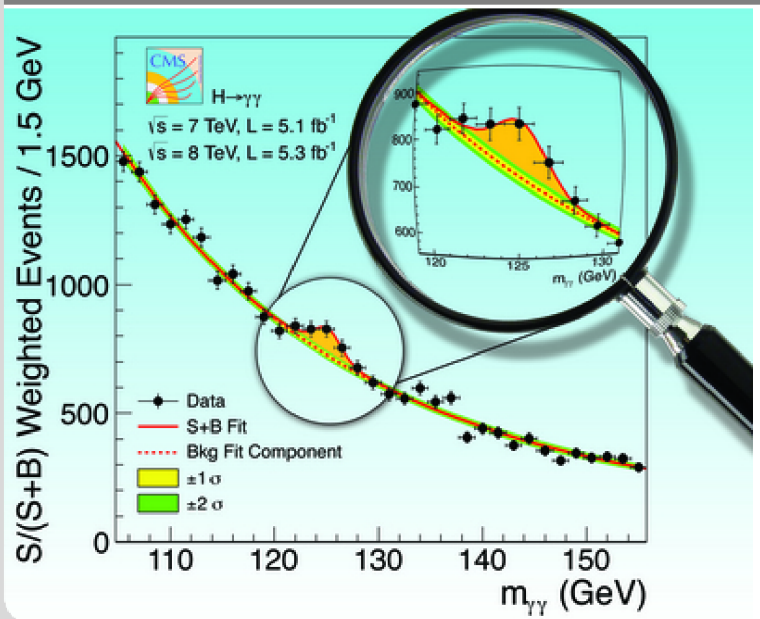
Moderne Methoden der Datenanalyse

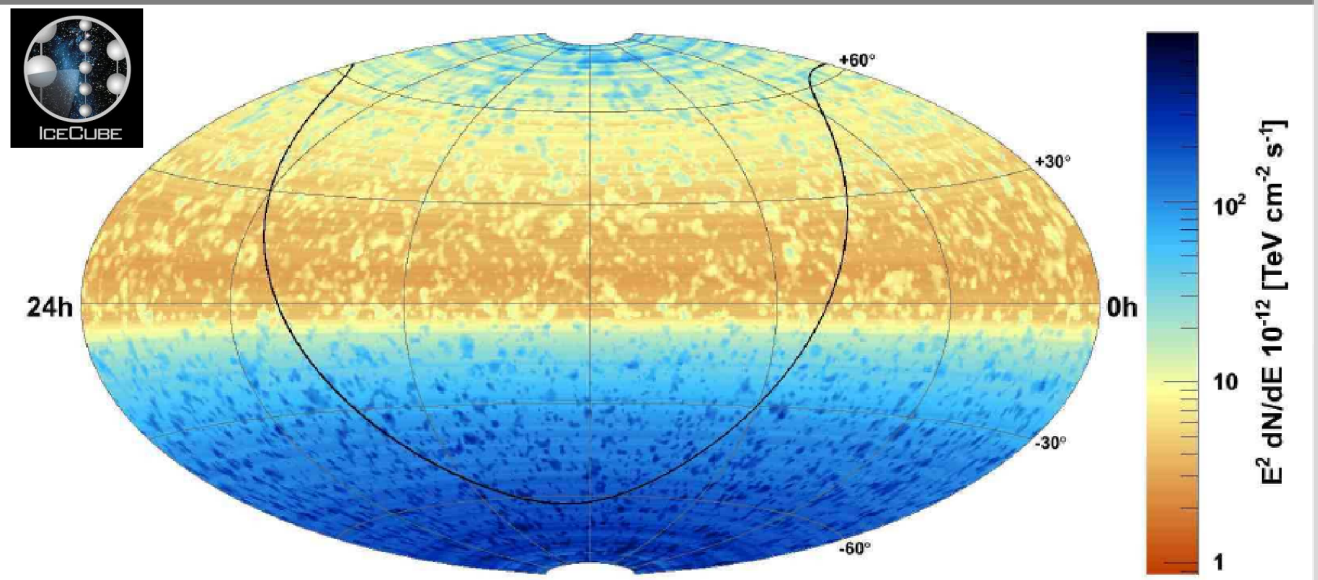
6.6.2017

Ralf Ulrich, Andreas Meyer

Institut für Kernphysik, Institut für Experimentelle Kernphysik

Master-Kurs SS 2017





Themen, Volesungsprogramm

- Einführung: Überblick und grundlegende Konzepte (1)
- Einführung: Überblick und grundlegende Konzepte (2)
- Zufallszahlen und Monte-Carlo Methoden
- **Hypothesentests (1)**
- **Parameterschätzung**
- **Parameterschätzung (Goodness-of-fit)**
- **Optimierungs- und Parametrisierungsmethoden**
- Konfidenzintervalle
- Hypothesentests (2)
- Klassifikation
- Klassifikation
- Klassifikation
- Messen und Entfalten
- Systematische Unsicherheiten

- Legendre Polynome
- Splines
- Wavelets

fallen weg. Die Prioritäten der Vorlesung wurden entsprechend gesetzt.

Wichtiger (eingeschobener) Hinweis!

Bitte beachten Sie, daß es in diesen Folien einige Seiten und Informationen gibt die der Vollständigkeit wegen und um den Gesamtzusammenhang des Themas deutlich zu machen vorhanden sind.
Diese Folien sind als solche markiert und wurden in der Vorlesung nur am Rande betrachtet. Für das Selbststudium sind sie aber durchaus wertvoll. Der spezifisch Stoff auf diesen Folien wird nicht geprüft.

Notwendigkeit von Optimierung

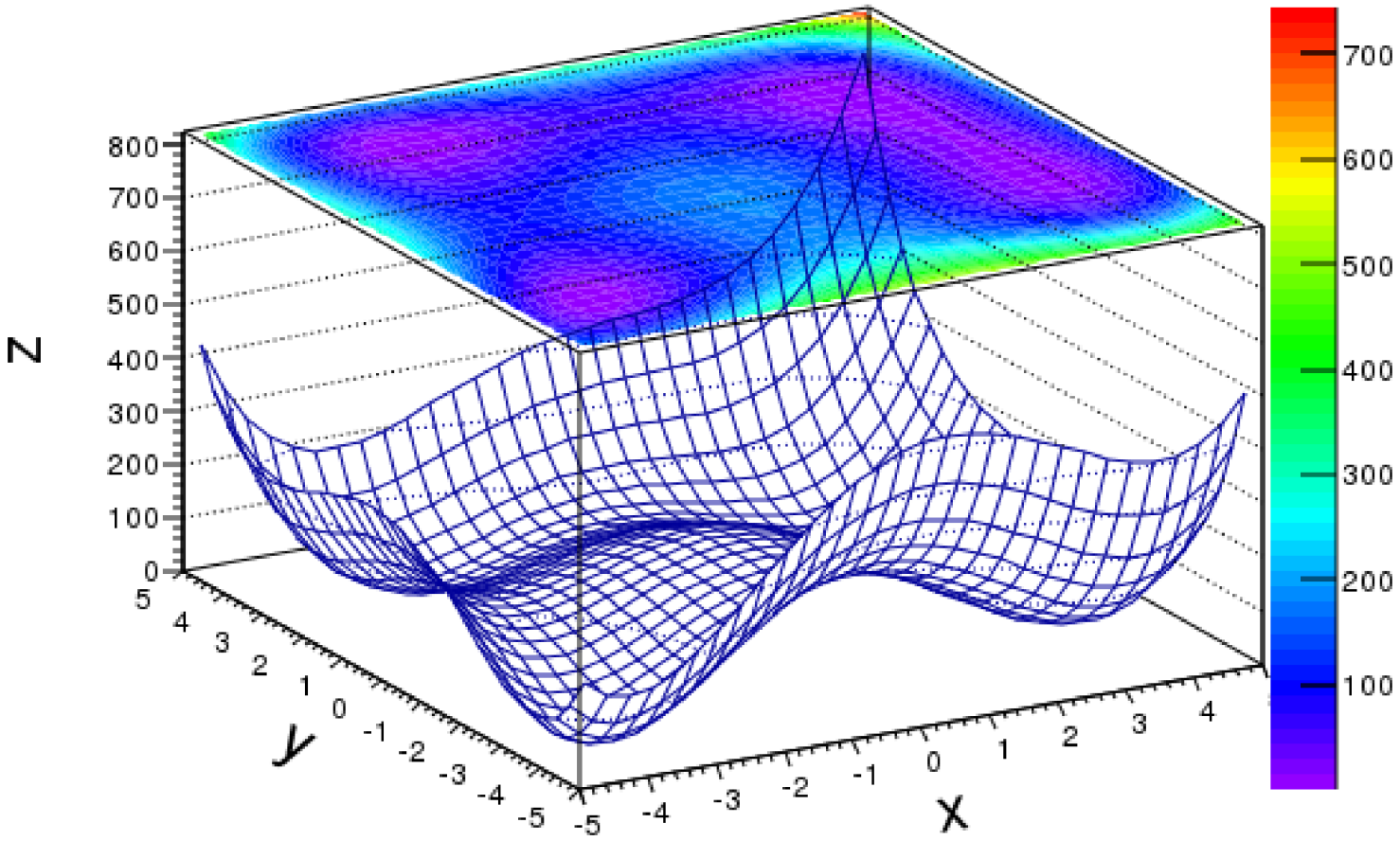
- **Häufige Aufgabe bei Parameterschätzung**

  Minimierung der negativen log-Likelihood Funktion oder der Summe der quadratischen Abweichungen und Berechnung der Unsicherheiten aus der zweiten Ableitung am Minimum.

- **→ Optimierung**

  Bestimmung des Minimums einer Gütefunktion $F(\vec{x})$ (und deren Kovarianzmatrix am Minimum) mit oder ohne Nebenbedingen in Gleichungs- oder Ungleichungsform

Globale und Lokale Minima



$$f(x,y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2$$

- **Optimalität**
  $\vec{x}$ ist ein lokales Minimum, wenn $F(\vec{x}^*) < F(\vec{x})$ für alle $\vec{x} \neq \vec{x}^*$ in der Umgebung von $\vec{x}$

- **Achtung lokales Minimum muss nicht global sein!**

Minimierung in mehreren Dimensionen

- $F(\vec{x})$ sei eine glatte Funktion (1. und 2. Ableitung stetig) oft erfüllt, zumindest in Lösungsnähe
- Gradient

$$\vec{g}(\vec{x}) = \nabla F(\vec{x}) = \begin{pmatrix} \frac{\partial F}{\partial x_1} \\ \vdots \\ \frac{\partial F}{\partial x_n} \end{pmatrix}$$

- Notwendige Bedingung für Minimum: (oder Maximum bzw. Sattelpunkt)

$$\vec{g}(\vec{x}) = 0$$

- Also:

$$\frac{\partial F}{\partial x_i} = 0 \qquad \text{für alle } i$$

Krümmung einer Funktion, Hesse Matrix

- Matrix der 2. Ableitungen, n x n symmetrisch

$$H(\vec{x}) = \begin{pmatrix} \frac{\partial^2 F}{\partial x_1^2} & \cdots & \frac{\partial^2 F}{\partial x_1 \partial x_n} \\ \vdots & & \vdots \\ \frac{\partial^2 F}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 F}{\partial x_n^2} \end{pmatrix}$$

- z.B. Taylor Entwicklung

$$F(\vec{x} + \vec{\Delta x}) = F(\vec{x}) + \vec{g}^{\mathrm{T}} \vec{\Delta x} + \frac{1}{2} \vec{\Delta x}^{\mathrm{T}} H \vec{\Delta x} + \ldots$$

- Liegt vielen Minimierungsalgorithmen zugrunde

- Hinreichende Bedingung für lokales Minimum

  1) $\vec{g}(\vec{x})=0$ und 2) $H(\vec{x})$ positiv definit

Spektrale Zerlegung

eigenes Studium

- Für eine symmetrische n x n Matrix H existieren n orthogonale Eigenvektoren u_i mit Eigenwerten λ_i

$$Hu_i = \lambda_i u_i$$

- Orthogonale Matrix $U=(u_1, ..., u_n)$ mit normierten Eigenvektoren als Spalten transformiert H in eine Diagonalmatrix

$$D = U^{\mathrm{T}} H U = \begin{pmatrix} \lambda_1 & & 0 \\ & \dots & \\ 0 & & \lambda_n \end{pmatrix}$$

- Wegen $U^{-1} = U^{\mathrm{T}}$ gilt

$$H = U D U^{\mathrm{T}} = \sum_{i=0}^{n} \lambda_i u_i u_i^{\mathrm{T}}$$

- H^{-1} hat identische Eigenvektoren mit Eigenwerten $1/\lambda_i$

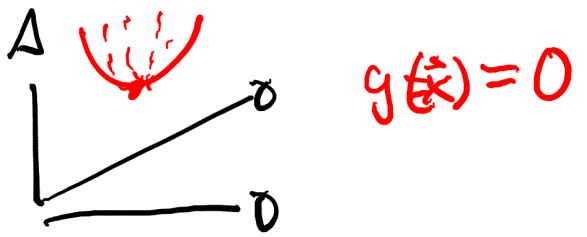
Eigenwerte der Hesse Matrix

- **Konditionszahl** $\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}$

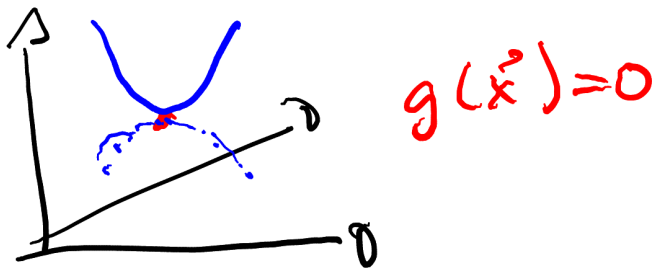
  Numerische Berechnung der inversen Matrix kann problematisch sein wenn die Konditionszahl groß ist (d.h. H „fast“ singulär ist)

- **Hesse Matrix bei $g(\vec{x}) = 0$**

  - positiv definit $\lambda_i > 0$ für alle i
    - Minimum

  - positiv semidefinit $\lambda_i >= 0$ für alle i
    - „Tal“, Lösung oft instabil

  - indefinit $\lambda_i < 0$ für mindestens ein i
    - Sattelpunkt, instabil

Eindimensionale Minimierung: Suchmethode

eigenes Studium

Voraussetzung: Unimodale Funktion (→ eindeutiges Minimum)

- **1) Einschluss des Minimums**
  - Ausgangspunkt: Startwerte x_1 und x_2 mit $F(x_1)>F(x_2)$
  - Iteration: $x_k = x_{k-1} + \alpha\,(x_{k-1} - x_{k-2})$ (z.B. mit $\alpha = 3$)
  - Abbruchbedingung: $F(x_k)>F(x_{k-1})$

- **2) Reduktion des Einschlussintervalls**
  - Ausgangspunkt: Tripel (x_1, x_2, x_3) mit $F(x_1)<F(x_2)<F(x_3)$
  - Testpunkt x_t zwischen x_{k-1} und x_k (oder x_{k-2} und x_{k-1})
  - Falls $F(x_t)<F(x_{k-1})$: neues Tripel (x_{k-1}, x_t, x_k)
  - Falls $F(x_t)>F(x_{k-1})$: neues Tripel (x_{k-2}, x_{k-1}, x_t)

Goldene Schnitt

eigenes Studium

- Wahl des Testpunktes x_t zwischen x_1 und x_2, so dass

$$\frac{x_2 - x_t}{x_2 - x_1} = \frac{x_t - x_1}{x_2 - x_t}$$

- Lange Teilstrecke zu Gesamtstrecke = kurze zu lange Teilstrecke
  - Verhältnis des goldenen Schnitts

$$\tau = \frac{\sqrt{5} - 1}{2} = 0.618034...$$

  - Konstante Reduktion des Einschlussintervalles pro Iteration um Faktor τ
  - Sinnvoll $\alpha = 1/\tau$ zu wählen in 1. Phase der Suche

- Suchmethode ist robust
- Da unabhängig vom Verhalten der Funktion

Goldener Schnitt

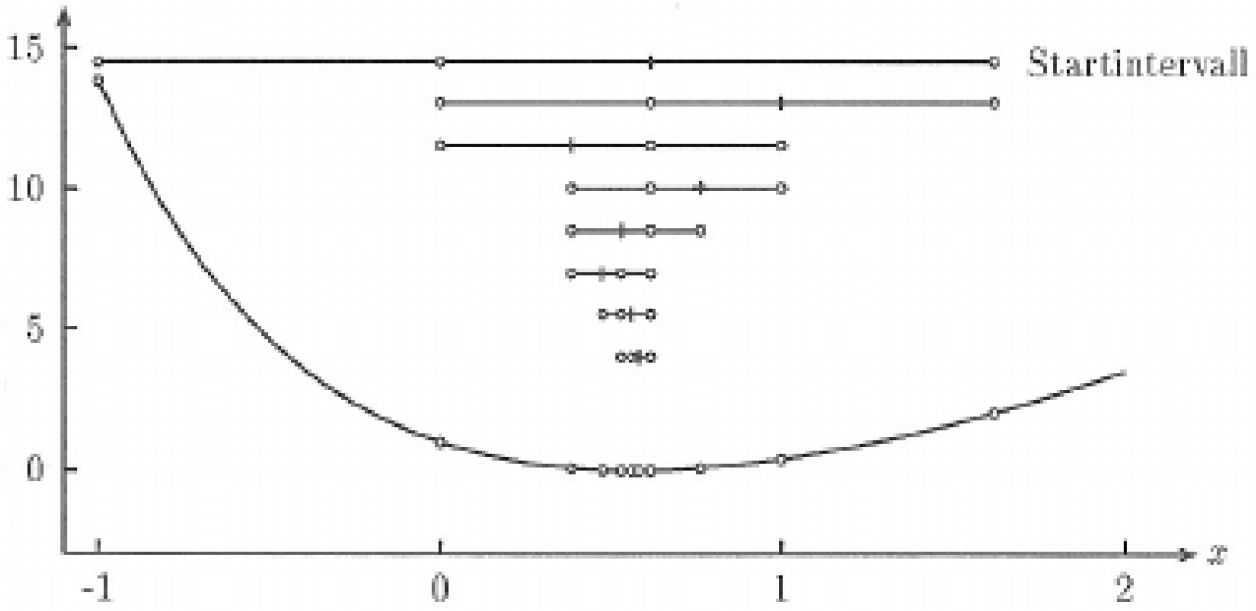
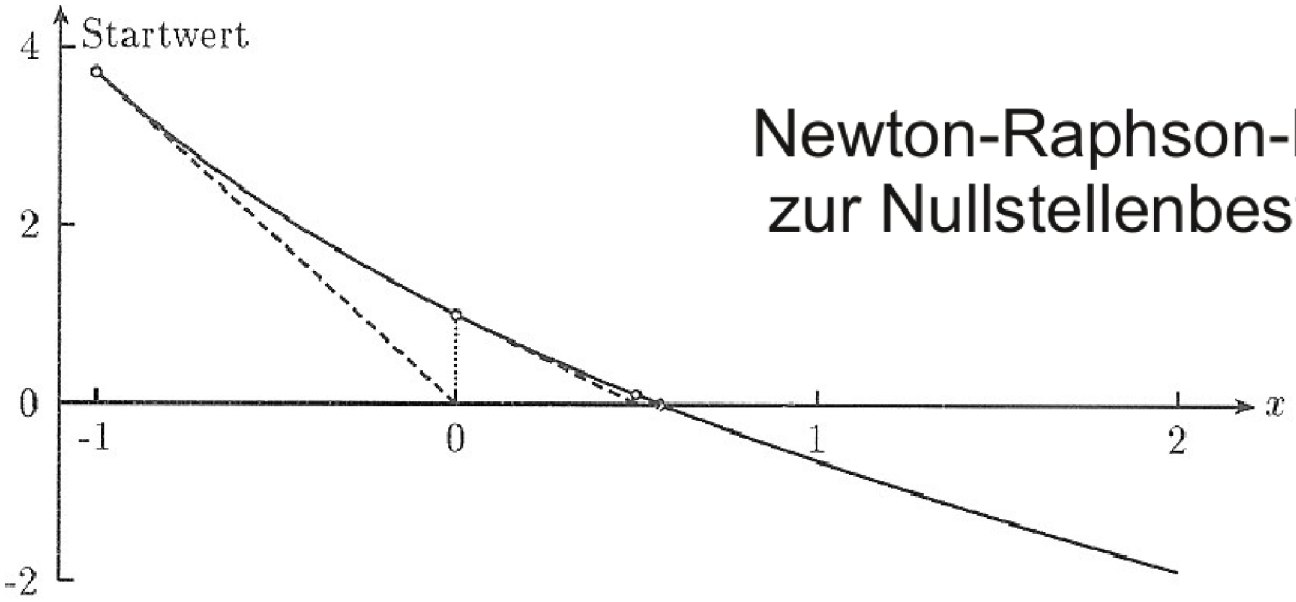


Abbildung 8.3: Bestimmung des Minimums der Funktion $f(x) = (\exp(-x) - x)^2$ nach der Methode des goldenen Schnitts. Die waagerechten Linien zeigen, von oben nach unten, jeweils das durch zwei Punkte begrenzte Intervall, das durch den mittleren Punkt nach dem goldenen Schnitt geteilt wird. Durch einen neuen Punkt kann das Intervall jeweils um den Faktor $\tau = 0.618$ verkürzt werden.

Newton Methode

eigenes Studium



Newton-Raphson-Methode
zur Nullstellenbestimmung

Abbildung 8.4: Bestimmung der Nullstelle der Funktion $f(x) = \exp(-x) - x$ nach der Newton-Methode mit $x_0 = -1$ als Startwert. Gestrichelt gezeichnet für die ersten beiden Iterationen ist die jeweilige Tangente an den Näherungswert, deren Schnittpunkt mit der x-Achse den nächsten Näherungswert ergibt.

- Anwendung auf Ableitung zur Minimum Suche

$$x_{k+1} = x_k - \frac{F'(x_k)}{F''(x_k)}$$

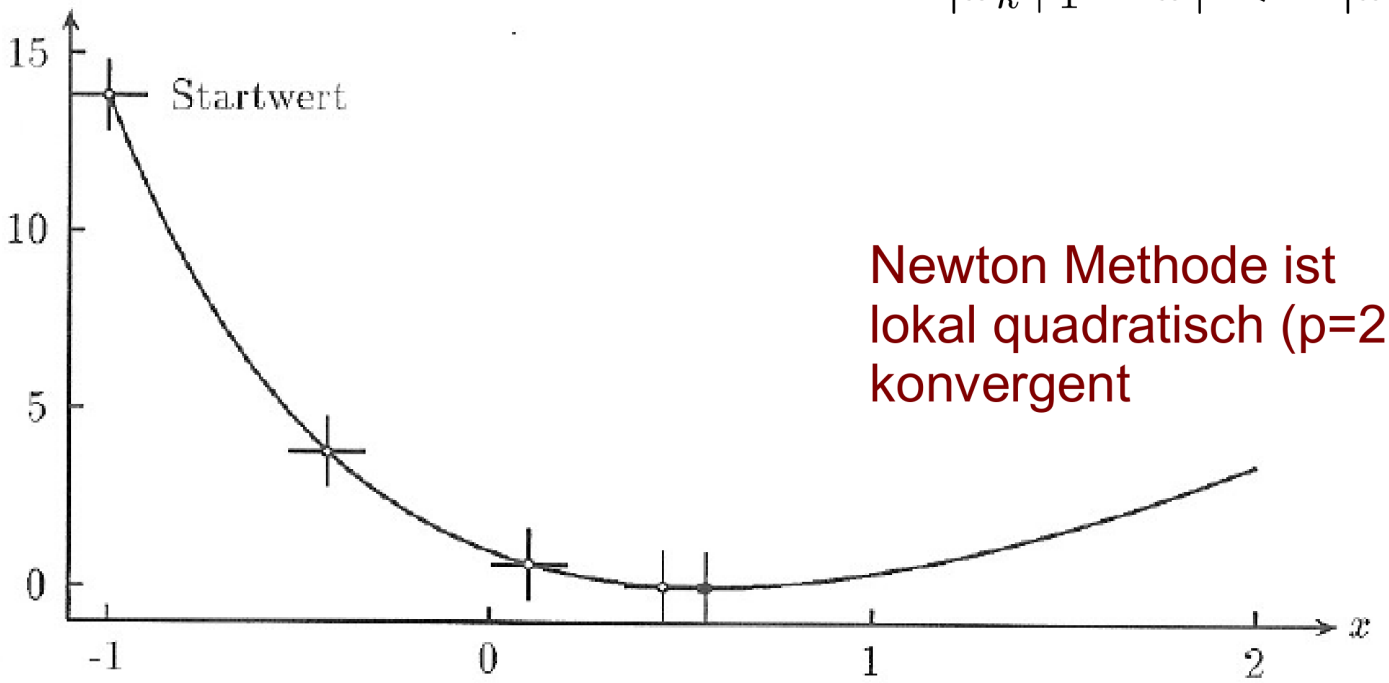
- **Konvergenzverhalten nicht garantiert**

Konvergenzverhalten

eigenes Studium

- Eine Iterationsmethode ist **lokal konvergent** von der **Ordnung p**, wenn es eine positive Konstante K (K<1 für p>=1) gibt, so dass für alle Startwerte x_0 in einer Umgebung des Fixpunktes x* gilt:

$$|x_{k+1} - x| < K|x_k - x^*|^p$$



Newton Methode ist lokal quadratisch (p=2) konvergent

Abbildung 8.5: Bestimmung des Minimums der Funktion $f(x) = (\exp(-x) - x)^2$ nach der Newton-Methode mit $x_0 = -1$ als Startwert. Die Näherungswerte (Kreuze) konvergieren monoton gegen das Minimum bei $x = 0.567143$.

Such- versus Newton-Methode

Suchmethode	Newton Methode
Verwendet nur Funktionswerte, keine Ableitungen	Verwendet nur 1. und 2. Ableitung, keine Funktionswerte
Robust	Konvergenz nicht garantiert
Lokal linear konvergent	Lokal quadratisch konvergent

Robuste und schnelle Methode durch
Kombination beider Methoden

Kombinierte Methode

- **Polynom - Interpolationsmethode**

  Bekanntes Minimum für Polynom durch berechnete Funktionswerte ergibt nächsten Testpunkt

  Parabolische Interpolation entspricht Newton-Methode mit numerisch berechneten Ableitungen

- Bei sehr asymmetrischer Intervallteilung:

  Rückfall auf goldenen Schnitts

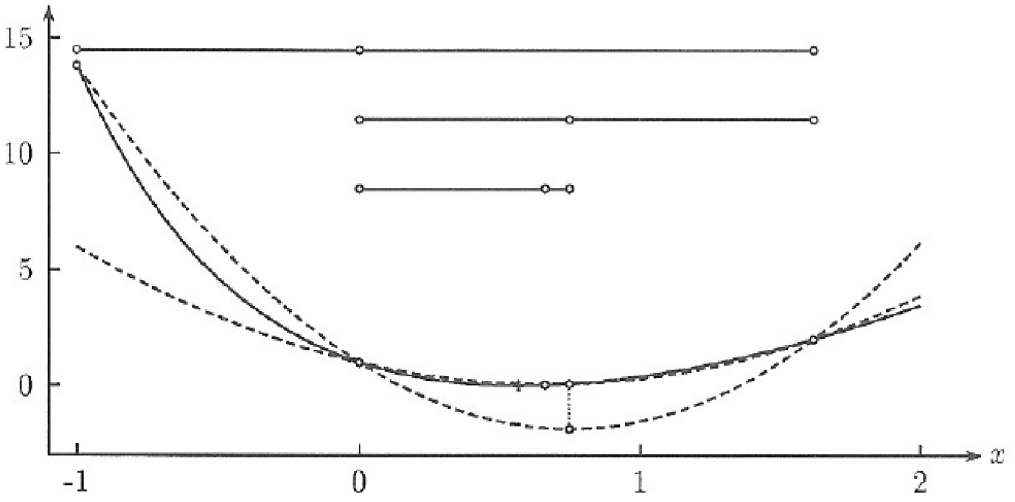


Abbildung 8.6: Bestimmung des Minimums der Funktion $f(x) = (\exp(-x) - x)^2$ mit Hilfe der Parabelinterpolation. Die gestrichelten Kurven sind Parabeln durch drei Punkte der Funktion; das Minimum der Parabel ist jeweils die nächste Testkoordinate. Die waagerechten Linien zeigen, von oben nach unten, jeweils das durch zwei Punkte begrenzte Intervall mit einem mittleren Punkt. Die zweite Parabel kommt der Funktion im Bereich um das Minimum zwar schon sehr nahe, führt jedoch zu einer stark asymmetrischen Teilung des Intervalls.

Suchmethoden in d Dimensionen (d>1)

- **Gittermethode**
  - k gleichverteilte Testwerte pro Dimension
  - Erfordert k^d Berechnungen
  - Ungeeignet für große d

eigenes Studium

- **Monte Carlo Methode**
  - Funktionsberechnung an zufällig verteilten Testpunkten
  - Auch bei großen d möglich
  - Nicht effizient, aber geeignet für Bestimmung von Startwerten

- **Einfache Parametervariation**
  - Eindimensionale Minimierung jeweils für einen Parameters
  - Dann Minimierung für den nächsten Parameter → Iteration
  - i.A. nur schnelle Konvergenz, wenn Minimierung entlang der Eigenvektoren der Hesse-Matrix

Suchmethoden für d>>1 sehr ineffizient und nicht zu empfehlen.

Beispiel der einfachen Parametervariation

eigenes Studium

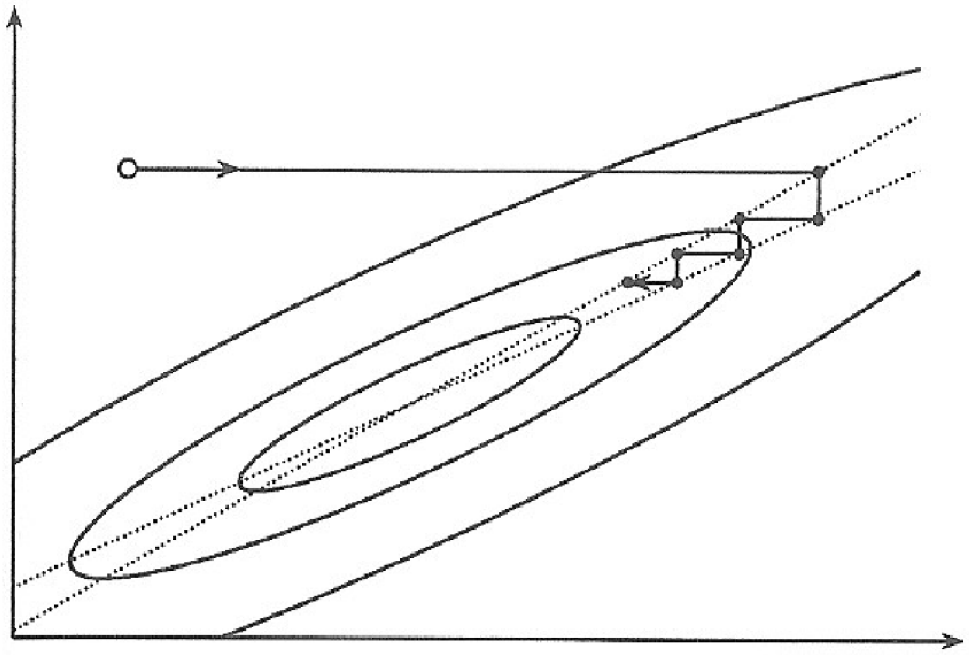


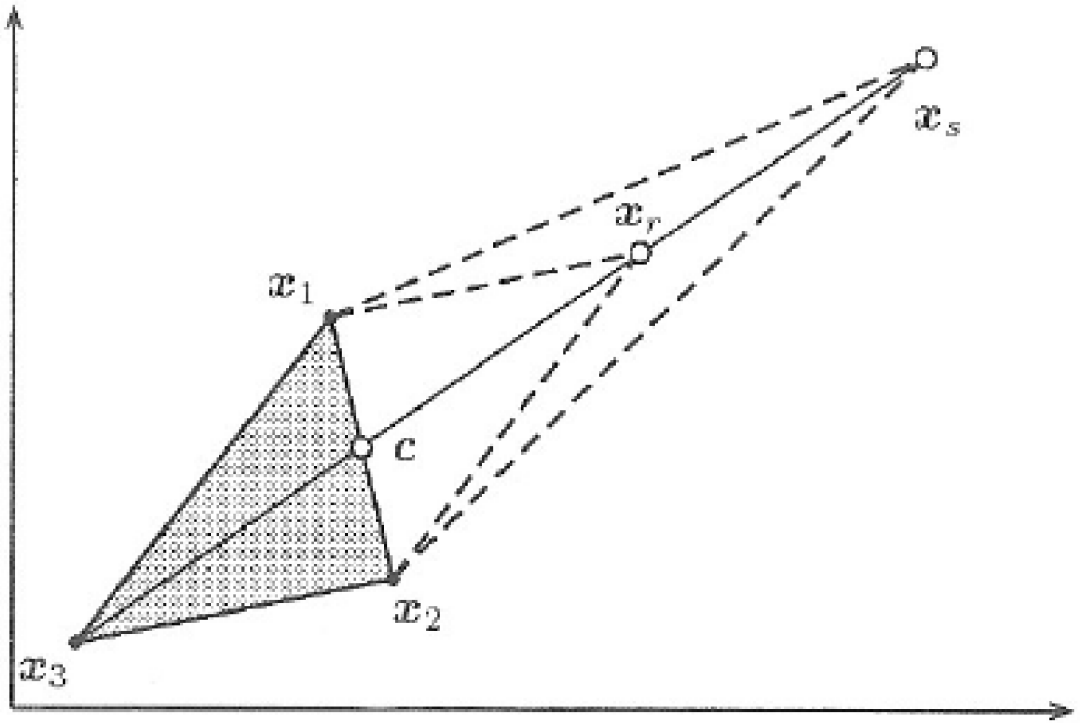
Abbildung 8.7: Die Methode der einfachen Parametervariation bei einer quadratischen Funktion von zwei Variablen. Durch die Korrelation der Parameter ist die Konvergenz langsam.

Langsame Konvergenz, da Suchrichtung nicht der Richtung der Eigenvektoren der Hesse Matrix entspricht

Simplex Methode

eigenes Studium

- d+1 Punkte im Raum R^d $\rightarrow$ d - dimensionaler Polyeder oder Simplex
- Sortierung, so daß $F(\vec{x}_1) < \cdots < F(\vec{x}_{d+1})$
- Schwerpunkt der d besten Punkte $\vec{c} = \sum_{i=1}^{d} \vec{x}_i / d$
- Spiegelung des schlechtesten Punktes an $\vec{c}$: $\vec{x}_r = \vec{c} + \alpha(\vec{c} - \vec{x}_{d+1})$



Simplex Methode

eigenes Studium

- **Falls** $F(\vec{x}_1) < F(\vec{x}_r) < F(\vec{x}_d)$ x_r ersetzt x_{d+1}

- **Falls** $F(\vec{x}_r) < F(\vec{x}_1)$
  gute Richtung $\rightarrow$ Streckung
  $\vec{x}_s = \vec{c} + \beta(\vec{x}_r - \vec{c})$ $\beta > 1$

- **Falls** $F(\vec{x}_r) > F(\vec{x}_d)$
  Simplex zu groß $\rightarrow$ Abflachung
  $\vec{x}_s = \vec{c} - \gamma(\vec{c} - \vec{x}_{d+1})$ $0 < \gamma < 1$

- **Falls** $F(\vec{x}_s) < F(\vec{x}_{d+1})$ $\vec{x}_s$ ersetzt $\vec{x}_{d+1}$

- **Ansonsten** Kontraktion um $\vec{x}_1$:
  $\vec{x}_j = \vec{x}_1 + \delta(\vec{x}_j - \vec{x}_1)$ $0 < \delta < 1$

Beispiel, Simplex in 2 Dimensionen

eigenes Studium

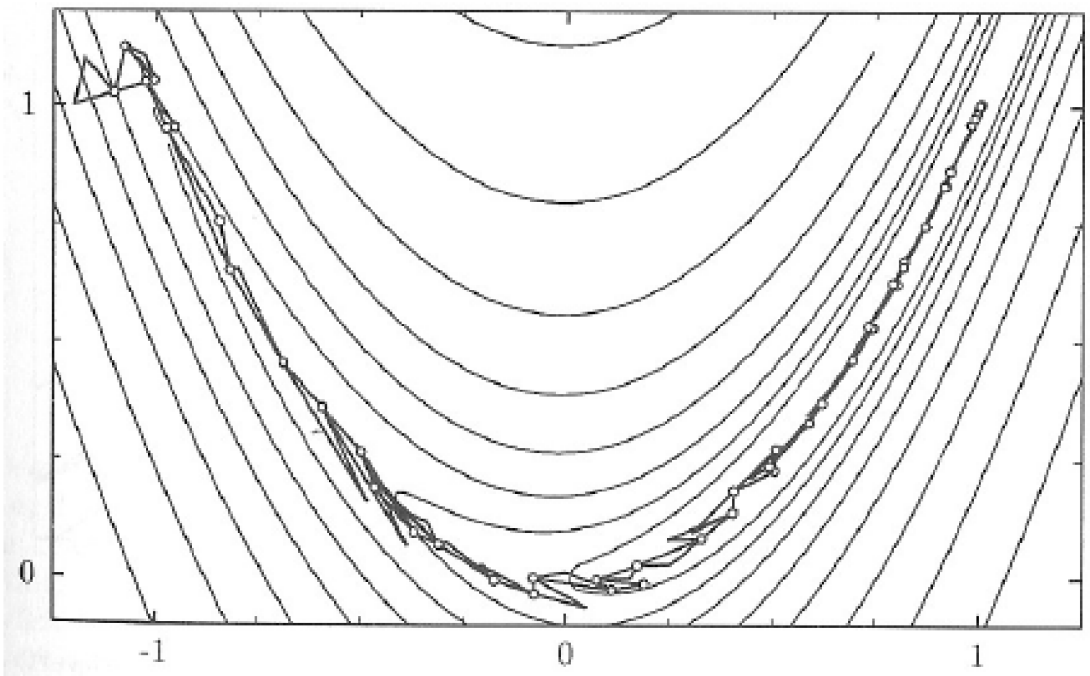


Abbildung 8.9: Minimierung der Rosenbrook-Funktion $f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$ mit der Simplex-Methode vom Startwert $(-1.2, 1.0)$ bis zum Minimum bei $(1.0, 1.0)$. Die niedrigsten Funktionswerte bei jedem Zyklus sind durch Kreise, und das Simplex ist bei jedem zweiten Zyklus durch Linien angegeben. Die Konturlinien der Funktion sind für die Funktionswerte 1, 2, 4, 8, 16, 32, 64 und 128 gezeichnet. Die Funktion hat ein sehr enges parabelförmiges Tal.

Simplex passt sich Verlauf
Der Funktion an.
Informationen aus vorhergehenden Funktions-Berechnungen werden genutzt, ohne Ableitungen zu verwenden.

Methode des steilsten Abstieges

eigenes Studium

$$\Delta x = -g$$

- Einfach, aber ernsthafte Nachteile
  - Keine natürliche Schrittweite gegeben
  - Nur lineare Konvergenz
  - Insbesondere langsam wenn Konditionszahl von H groß

$$K \approx \left(\frac{\kappa - 1}{\kappa + 1} \right)^2$$

- **Besser: Newton Methode**

Newton Methode in mehreren Dimensionen

- **Quadratische Näherung**

$$F(\vec{x}+\vec{\Delta x}) \approx F(\vec{x}) + \vec{g}^T\vec{\Delta x} + \frac{1}{2}\vec{\Delta x}^T H \vec{\Delta x}$$

$$\vec{g}(\vec{x}+\vec{\Delta x}) \approx \vec{g} + H\vec{\Delta x}$$

- **Bedingung für Minimum**

$$\vec{g}(\vec{x}+\vec{\Delta x}) = 0$$

- **$\rightarrow$ Newton Schritt**

$$\vec{\Delta x} = -H^{-1}\vec{g}$$

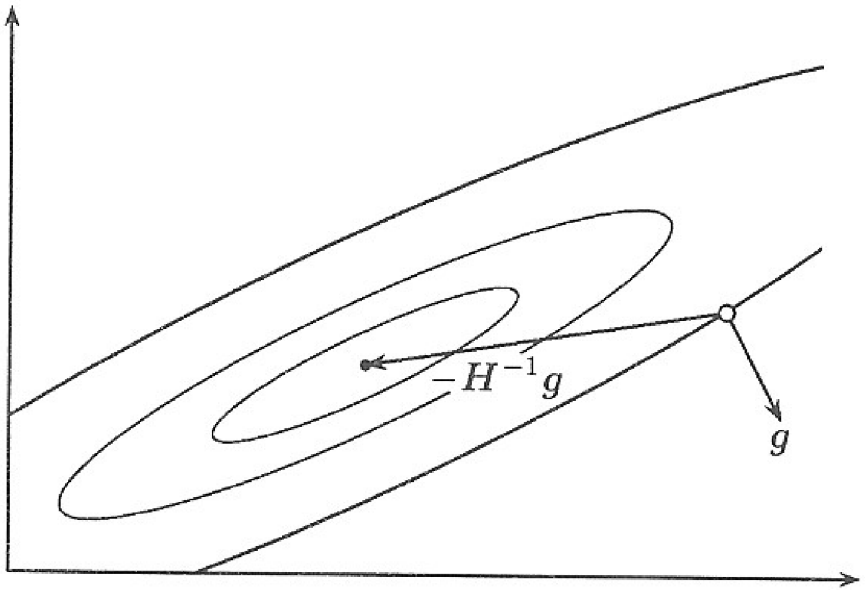


Abbildung 8.11: Ein Schritt der Newton-Methode bei einer quadratischen Funktion von zwei Variablen, deren Konturlinien Ellipsen sind. Gezeigt ist der Gradientenvektor g am Startpunkt, und der Newton-Schritt $-H^{-1}g$, der in diesem Fall genau auf das Minimum zeigt.

Line-Search in Newton-Richtung

eigenes Studium

- **Funktionsverlauf in Newton Richtung**

$$f(z) = F(\vec{x} + z\Delta\vec{x})$$

- **Quadratische Näherung**

$$f(z) = F(\vec{x}) + d\left(\frac{z^2}{2} - z\right)$$

mit $d = -\vec{g}^T \Delta\vec{x}$

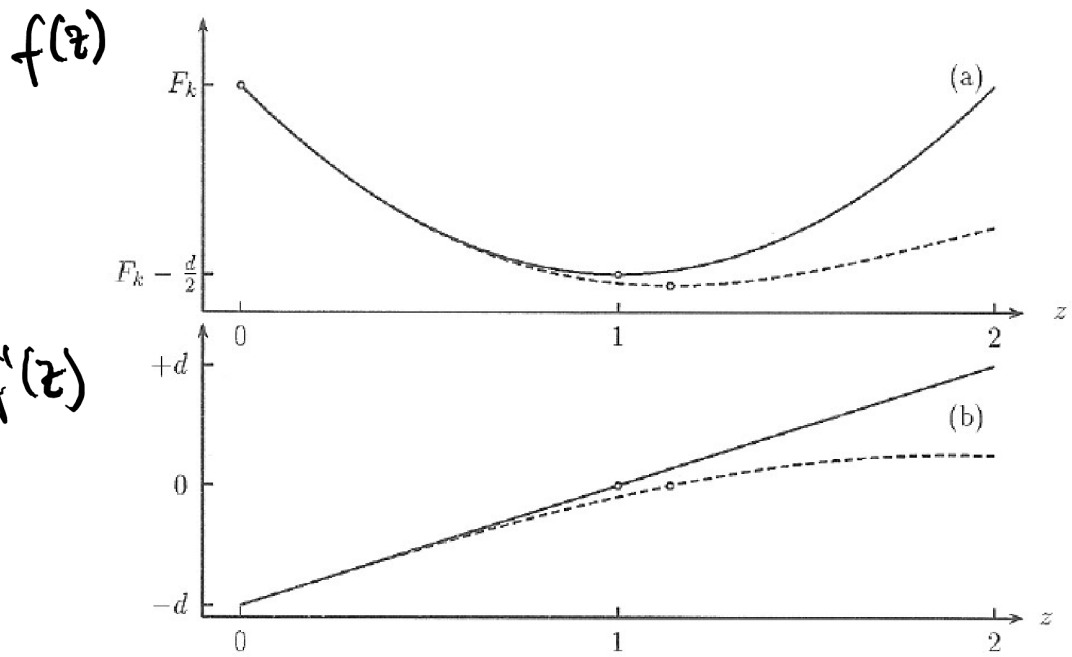


Abbildung 8.12: Die Funktion $f(z) = F_k + d(z^2/2 - z)$ (a) und die Ableitung $f'(z) = d(z - 1)$ (b) als durchgezogene Kurven. Gestrichelt gezeigt ist die mögliche Abhängigkeit innerhalb einer Minimierung, die vom idealisierten Verlauf abweicht.

Minimierung von f(z) bei jedem Schritt

Algorithmus mit Line-Search

eigenes Studium

- Definiere Startwert x_0
- **Berechne Suchvektor Δx,** z.B. Newton-Vektor $\vec{\Delta x} = -H^{-1}\vec{g}$
- **Minimiere Funktion in Richtung des Suchvektors** (line-search)
  - Eindimensionale Minimierung von $f(z) = F(\vec{x} + z\,\vec{\Delta x})$
- **Iteration**: $x_{k+1} = x_k + z_{min}\,\Delta x$
- **Konvergenztest**:
  - x_{k+1} ist Lösung bei erfolgreichem Konvergenztest
  - z.B. $dist = |x_k - x_{k+1}| < \varepsilon$ oder $F_k - F_{k+1} < \varepsilon$
  - Empfehlung Blobel: ε=0.01
  - **Oder**: Abbruch bei Erreichen einer Maximalzahl von Iterationen

Beispiel: Anpassen einer Exponentialfunktion

eigenes Studium

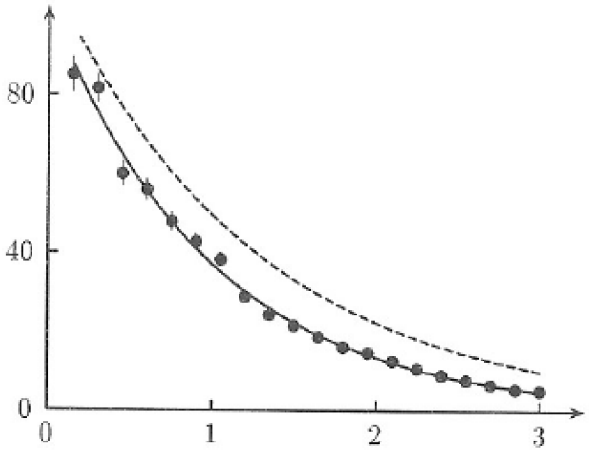


Abbildung 8.13: Anpassung einer Exponentialfunktion an 20 Datenpunkte. Die gestrichelte Kurve zeigt die Exponentialfunktion mit den Startwerten der Parameter.

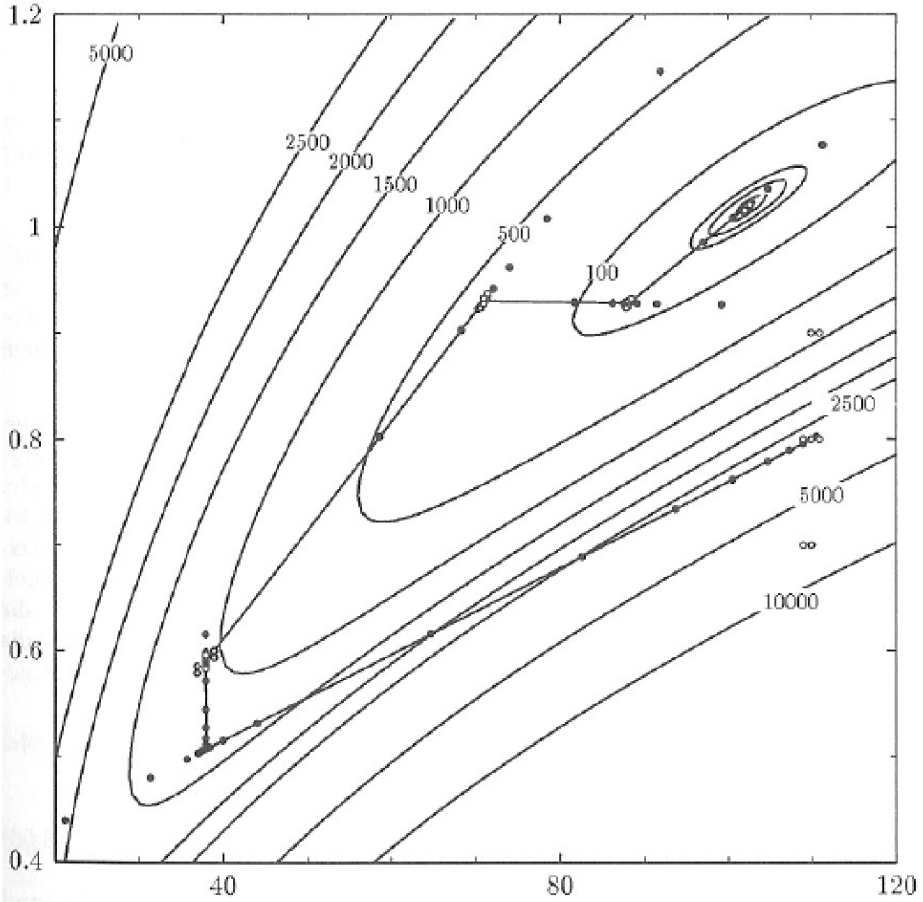


Abbildung 8.14: Minimierung der χ^2-Funktion für die Anpassung einer Exponentialfunktion mit dem Startwert $(x_1, x_2) = (110, 0.8)$. Die ausgefüllten Kreise sind Punkte, bei denen bei der eindimensionalen Minimierung längs der Newton-Richtung die χ^2-Funktion berechnet wurde. Die offenen Kreise sind Punkte, bei denen bei der numerischen Differentiation die χ^2-Funktion berechnet wurde. Um das Minimum mit einem Wert der χ^2-Funktion von 15.56 bei $(x_1, x_2) = (102.4, 1.018)$ sind die Konturlinien für eine, zwei und drei Standardabweichungen gezeichnet. Weiterhin sind die Konturlinien für Funktionswerte 100 ... 10000 gezeichnet.

Modifizierte Newton Methode

eigenes Studium

- Falls Hesse Matrix **nicht positiv definit**
- Verwenden modifizierter Hesse Matrix H' zur Berechnung des Newton-Vektors

entweder:

- Spektrale Zerlegung
  - Setze $\lambda_i' = max(|\lambda_i|, \delta)$ (+ Rücktransformation)
  - sehr aufwendig

oder:

- Addition der Einheitsmatrix I_n $H' = H + \alpha I_n$

- H` positiv definit, falls $\alpha > |\lambda_{min}|$
- α klein $\rightarrow$ nahe Newton Richtung
- α groß $\rightarrow$ nahe steilstem Abstieg

Methoden mit variabler/adaptiver Metrik

- **Statt numerischer Berechnung der Hesse Matrix**
  → $O(n^2)$ Berechnungen erforderlich

- **Iterative Schätzung** der Hesse Matrix aus Änderung der Gradientenvektors

- Stichwort: „Broyden-Fletcher-Goldfarb-Shannon (BFGS)-Methode“

- MIGRAD in MINUIT

Nebenbedingung als Gleichung

Minimierung von F(x) mit m Bedingungen $f_1(x) = \cdots = f_m(x) = 0$

- z.B. Energie- und Impulserhaltung

- **Parametertransformation**
  - z.B. φ mit r=const statt x und z

- Methode der **Lagrange´schen Multiplikatoren**

$$\Phi(\vec{x}, \vec{\lambda}) = F(\vec{x}) + \sum_{i=1}^{m} \lambda_i f_i(\vec{x})$$

  - Minimierung von ϕ
    - Nebenbedingung erfüllt

$$\frac{\partial \Phi}{\partial \lambda_i} = 0 = f_i(\vec{x})$$

  - Zurückführen auf Minimierung ohne Nebenbedingungen, aber mit m zusätzlichen Dimensionen

- **Alternativ**: Minimierung, die Nebenbedingungen berücksichtigen, z.B. durch projizierten Gradient und Hesse Matrix bei linearen Nebenbedingungen

Nebenbedingung als Ungleichung

- Minimierung von $F(\vec{x})$, mit m Bedingungen $h_i(\vec{x})>0$, für i=1,...,m
  - z.B. Masse > 0 oder Wahrscheinlichkeit 0<p<1

- Am günstigsten
  - Konvergenz weit weg von den Grenzen
  - Geeignete Wahl der Startparameter

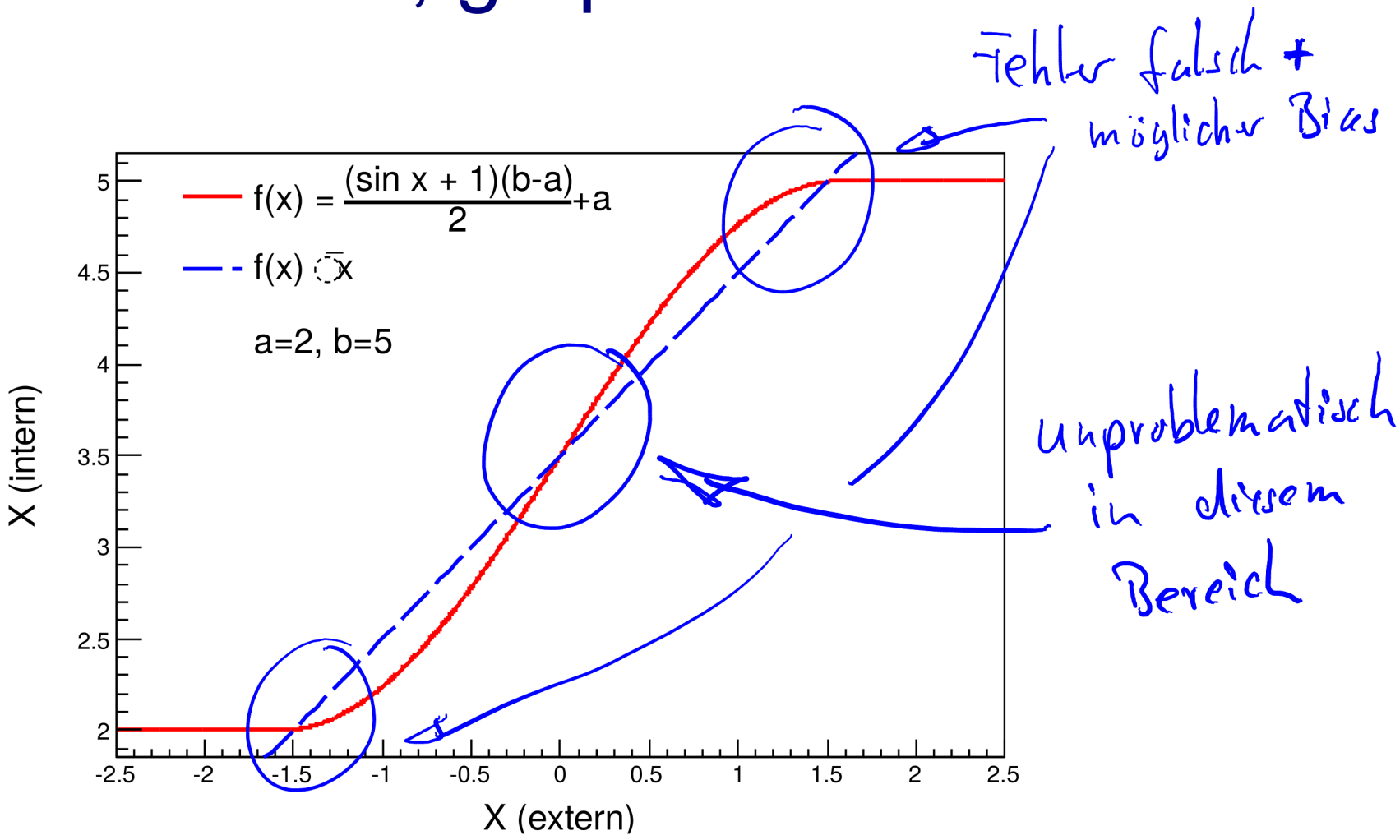
- Parametertransformation
  - z.B. für a<x<b:
  - Kann zu numerischen Problemen führen

$$x' = \arcsin\left(\frac{2(x-a)}{b-a} - 1\right)$$

$$x = \frac{(\sin(x')+1)\,(b-a)}{2} + a$$

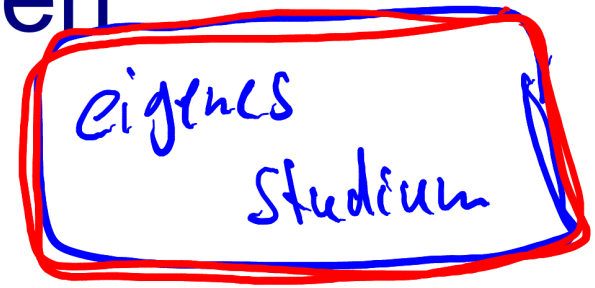
- Falls möglich → vermeiden !

Parameter Limits, graphisch



Falls möglich → vermeiden

Lösung von Gleichungssystemen



Lösen des Gleichungssystems

$$f_1(x) = f_2(x) = \cdots = f_m(x) = 0$$

entspricht Minimierungsproblem **nur** mit Nebenbedingungen

- Zu minimierende Gütefunktion $F(\vec{x}) = \sum_{i=1}^{m} f_i(x)$
  - F=0 am Minimum bei lösbaren Gleichungssystemen
  - Auch anwendbar bei überbestimmten Gleichungssystemen
    - Ergebnis hängt von der Gewichtung ab
- Methode kann recht **ineffizient** gegenüber angepassten Algorithmen sein

Kostenmethode

- **Häufige Nebenbedingung**
  Parameterwert und Unsicherheit bekannt aus anderen Messungen

$$x_i = x_i^0 \pm \sigma_i$$

Addition einer Kosten oder Straf/Penalty-Funktion

χ^2 Fit $\quad \chi^{2\prime}(\vec{x}) = \chi^2(\vec{x}) + (x_i - x_i^0)^2/\sigma_i^2$

mit: $p(x_i, x_0, \sigma_i) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(x_i - x_0)^2}{2\sigma_i^2}}$

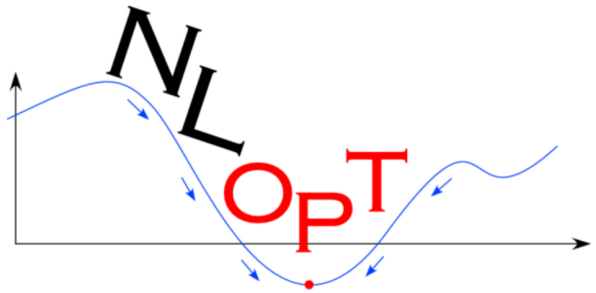
ML Fit $\quad -\ln \mathcal{L}'(x) = -\ln \mathcal{L} - \ln p(x_i, x_0, \sigma_i) = -\ln \mathcal{L} + \frac{1}{2}(x_i - x_i^0)/\sigma_i^2$

- Methode kann auch zur näherungsweise Berücksichtigung von Nebenbedingugen in Gleichungs- oder Ungleichungsform verwendet werden

Optimierung, Zusammenfassung

- Die Information, welche in Messdaten vorhanden ist kann durch die Struktur (hochdimensionaler) Funktionen (χ^2 oder Likelihood) analysiert werden.

- Viele Implementierungen, aber nicht alle empfehlenswert.
  Verwende z.B.



Scilab

MINUIT2

- Nötige Bedingung für Minimum:
  - Gradient Null
  - Hesse-Matrix positiv definit

- Bestes Verfahren: Polynom-Line-Search (Newton) in vielen Dimensionen mit iterativer Approximation der Hesse Matrix

Empfehlungen

Nutzen Sie existierende, gut getestete Optimierungs-algorithmen

- Achten Sie darauf, dass es keine Unstetigkeiten gibt
- Versuchen Sie numerische Limitierungen zu vermeiden (z.B. durch geeignete Skalierungen, numerische Genauigkeit)
- Verifizieren Sie analytisch berechnete Ableitungen durch numerische
- Probieren Sie unterschiedliche Startwerte aus
- Verwenden Sie physikalische oder logische Nebenbedingungen

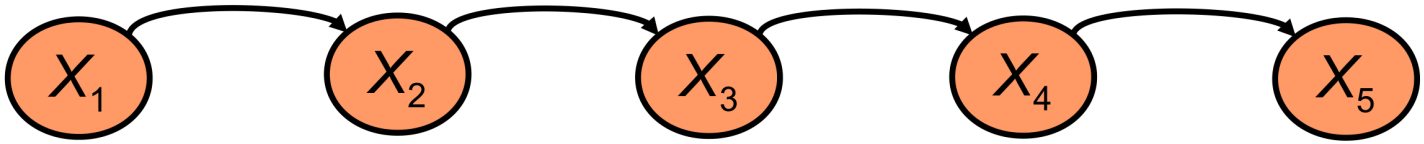
Markov Ketten

Markov Kette

- Eine Serie von Zufallsvariablen / Übergängen
  - Zustand bei i+1 hängt nur von Zustand i ab

- Wenn folgendes gilt:

$$P(X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, X_{n-2} = i_{n-2}, \ldots, X_0 = i_0) = P(X_{n+1} = j | X_n = i)$$



Übergänge zwischen diskreten Zuständen

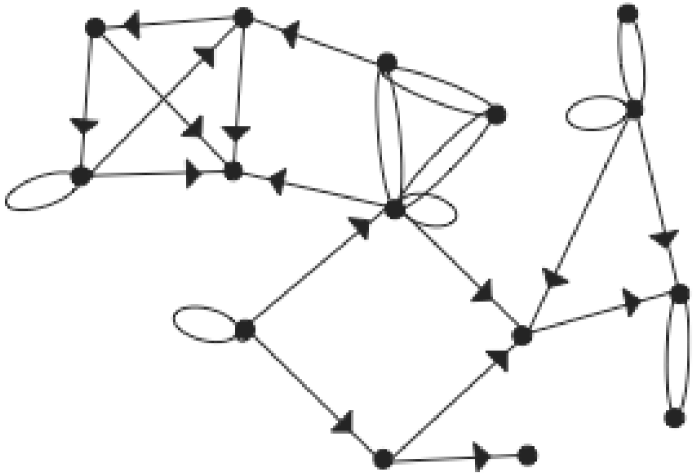
$$\mathrm{P}(\mathrm{X}_{n+1} = j | X_n = i) = p_{ij} \qquad p_{ij} \geq 0$$

$$P = \begin{pmatrix} p_{00} & p_{01} & \cdots & p_{0M} \\ p_{10} & p_{11} & \cdots & p_{1M} \\ \vdots & \vdots & & \\ p_{M0} & p_{M1} & \cdots & p_{MM} \end{pmatrix} \qquad n_i = \sum_{j=1}^{M} p_{ij} = 1$$

- Übergang eines Zustandes $\vec{x}$ nach einem Schritt in $P\vec{x}$
- Nach n Schritten: $P^n\vec{x}$

Markov Ketten

Di(rected)-Graph



Eigenschaften von Markov Ketten (einige):

- Homogen $p_{ij} \neq f(t)$
- Irreduzibel: positive Wahrscheinlichkeit nach n bzw. k Schritten von Zustand i nach j bzw. j nach i zu gelangen.
- Aperiodic: Für jeden Zustand i gilt, dass er nach einer irregulären Zeit wieder auftreten kann.
- Regulär: Eine Matrix P ist regulär, wenn es eine Potenz von P gibt in welcher alle Elemente >0 sind.

Stationärer Zustand

Frobenius Theorem

Eine Markov Kette ist *ergodisch* wenn es eine positive Wahrscheinlichkeit gibt nach n Schritten von jedem Zustand i nach j zu gelangen. Dies gilt z.B. für *reguläre* Ketten. (per Definition)

Erhaltung der Wahrscheinlichkeit

- 1 ist ein Eigenwert von P $P\pi = \pi$
- Alle Eigenwerte von P sind $|\lambda| \leq 1$ (Frobenius Theorem)
- Der Eigenvektor π zum Eigenwert 1 hat keine Elemente gleich 0 und kann ohne Einschränkungen auf eine Länge von 1 normiert sein.
- π ist der stationäre Zustand der Kette $\lim_{n\to\infty} P^n X = \pi$

Geschichte

- Konsonanten und Vokale in russischen Texten

  A. A. Markov, 1913

  An Example of Statistical Analysis of the Text of Eugene Onegin Illustrating the Association of Trials into a Chain

- Struktur und Analyse der Englischen Sprache

  C. E. Shannon and W. Weaver, 1964

  The Mathematical Theory of Communication

- Fundamentalere und allgemeinere Anwendung auf generelle physikalische Fragestellungen

  Erzeugung von Zufallszahlen, Parameter Optimierung

- Und noch vieles mehr...

Konsonanten

- Entwicklung der Markov Kette und die erste „praktische“ Anwendung
  - Wie oft folgt auf einen Vokal ein Vokal bzw. Konsonant und anders herum ?

Vokal Konsonant

$$P = \begin{pmatrix} 0.128 & 0.872 \\ 0.663 & 0.337 \end{pmatrix}$$

Vokal

Konsonant

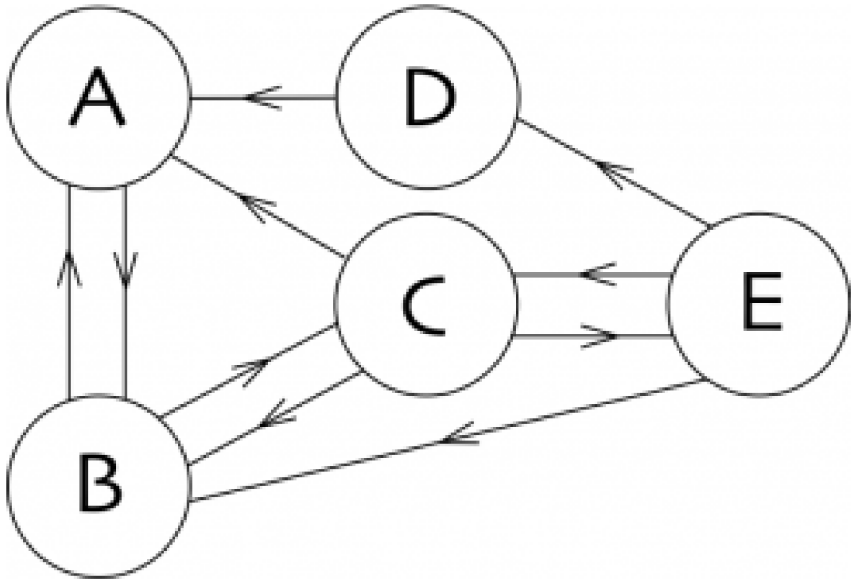
  - Stationärer Zustand dieser Kette:

$$\pi^T = (0.432,\ 0.568)$$

⇒ Wahrscheinlichkeitsdichte der Vokale und Konsonanten!

Google, PageRang Algorithmus

- Das Internet IST ein Netzwerk,
  das durch eine Übergangsmatrix
  beschrieben werden kann:



$$P = \begin{array}{c} \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{3} & 1 & 0 \\ 1 & 0 & \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & \frac{1}{2} & 0 & 0 & \frac{1}{3} \\ 0 & 0 & 0 & 0 & \frac{1}{3} \\ 0 & 0 & \frac{1}{3} & 0 & 0 \end{pmatrix} \end{array} \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix}$$

- $\lim_{n \to \infty} P^n X = \pi$

$$P^{32} = \begin{array}{c} \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{pmatrix} 0.293 & 0.293 & 0.293 & 0.293 & 0.293 \\ 0.390 & 0.390 & 0.390 & 0.390 & 0.390 \\ 0.220 & 0.220 & 0.220 & 0.220 & 0.220 \\ 0.024 & 0.024 & 0.024 & 0.024 & 0.024 \\ 0.073 & 0.073 & 0.073 & 0.073 & 0.073 \end{pmatrix} \end{array} \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix}$$

Wahrscheinlichkeits-Verteilung des Aufenthaltsortes eines „Zufalls-Surfers" nach ∞ Schritten

- $P\pi = \pi$ → $\pi^t = (0.293, 0.390, 0.220, 0.024, 0.073)$

(Weitere Informationen, z.B.: https://de.wikipedia.org/wiki/PageRank)

Google, PageRang Algorithmus

Falls Transformationsmatrix P nicht regulär ist, kann dies durch leichte Deformation erreicht werden

eigenes Studium

$$P_\beta = (1-\beta)P + \beta Q$$

- Google hat z.B. β = 0.15 verwendet
- Wobei Q eine NxN Matrix mit q_{ij}=1/N ist
- Außerdem kann Q die „Vorlieben“ der google Benutzer widerspiegeln
- Viele Details zum PageRank sind Firmen-Geheimnis
- Ohne Klassifikation ist das Internet quasi unbrauchbar, allerdings kann eine Klassifikation auch spezifische Informationen „verstecken“, falls diese nicht im allgemeinen Fokus steht.

Monte Carlo Markov Chains

Das zugrunde liegende Problem

$F(\vec{x})$ typischerweise eine Gütefunktion ($\chi^2(\vec{x})$, $\mathcal{L}(\vec{x})$, ...) oder ähnlich

Die d-Dimensionale Funktion $F(\vec{x})$ kann an jedem Punkt $\vec{x}$ berechnet werden

(d.h. F ist eine nicht unbedingt normierte Wahrscheinlichkeitsdichte)

- Wie können effektiv Zufallszahlen aus der Verteilung F gezogen werden?
- Wie kann die Funktion F effektiv abgetastet werden, um z.B. Minima/Maxima zu finden?

Anwendungen

Lösung: Algorithmus (Markov Kette), der uns den stationären Zustand $\pi(x) \overset{!}{=} f(x)$ liefert

- **Simulation**: Ziehe (typische) Zustände aus der Wahrscheinlichkeits-verteilung, welche das System bestimmen

$$X \sim \pi(\vec{x})$$

- **Integration / Computing**: In sehr großer Zahl von Dimensionen, z.B. für folgendes:

$$c = E\left[f(\vec{x})\right] = \int \pi(\vec{x}) f(\vec{x}) d\vec{x}$$

- **Optimierung**: Mit einem „Ausglühungs (*Annealing)*“ Schema

$$X^* = \operatorname{argmax}\left[\pi(\vec{x})\right]$$

- **Machine Learning, Bayesian Inference**

  → Abtasten von hoch-dimensionalen Verteilungen

Bayesische Statistik

eigenes Studium

$$p(\theta; \mathrm{data}) = \frac{p(\mathrm{data}; theta)p(\theta)}{p(\mathrm{data})}$$

- Alle interessanten Informationen erfordern Integration von $p(\theta; \mathrm{data})$ oder Teilen davon

  z.B. $E[p(\theta; \mathrm{data})]$ oder $cov_{ij}[p(\theta; \mathrm{data})]$ etc.

- Monte Carlo Markov Chains können genau das leisten. In vielen Fällen ist Bayesische Statistik tatsächlich nur mit MCMC Techniken möglich.

  - Mit MCMC können aus $p(\theta; \mathrm{data})$ Zufallszahlen generiert werden.

Bestimmung der Übergangs-Wahrscheinlichkeit

- Eine (*ergodische*) Markov Kette hat einen einzigen stationären Zustand (→Wahrscheinlichkeitsdichte der Zustände)

$$\pi(x)$$

- Für diesen Zustand muss die „detailed balance“ Bedingung gelten

Wahrscheinlichkeit von Zustand x_1 nach x_2 zu wechseln

$$\pi(x_1)P(x_1 \to x_2) = \pi(x_2)P(x_2 \to x_1)$$

- Wähle Übergangsfunktion folgendermaßen:

freie Wahl:

Vorschlagsfunktion

$$P(x_1 \to x_2) = g(x_1 \to x_2)A(x_1 \to x_2)$$

Akzeptanzfunktion

Definition der Akzeptanzfunktion

- Für $\pi(x) = P(x)$ und $P(x_1 \to x_2) = g(x_1 \to x_2) A(x_1 \to x_2)$
  gilt (durch Umformen der detailed balance Regel):

$$\frac{A(x_1 \to x_2)}{A(x_2 \to x_1)} = \frac{P(x_2) g(x_2 \to x_1)}{P(x_1) g(x_1 \to x_2)}$$

hier und die explizite (sinnvolle) Wahl: $\frac{P(x_2 \to x_1)}{P(x_1 \to x_2)} = \frac{P(x_2)}{P(x_1)}$

- Eine mögliche Wahl: **Metropolis-Hastings Methode**

andere Möglichkeiten: Gibbs-Sampling, Hybrid MC, etc...

beliebig: g

$$A(x_1 \to x_2) = \min\left[1, \frac{P(x_2)}{P(x_1)} \frac{g(x_2 \to x_1)}{g(x_1 \to x_2)}\right]$$

(einsetzen in „detailed balance" Gleichung um zu sehen, daß diese erfüllt ist)

Metropolis-Hastings Algorithmus

eigenes Studium

1. Start mit einem zufälligen Zustand x_1

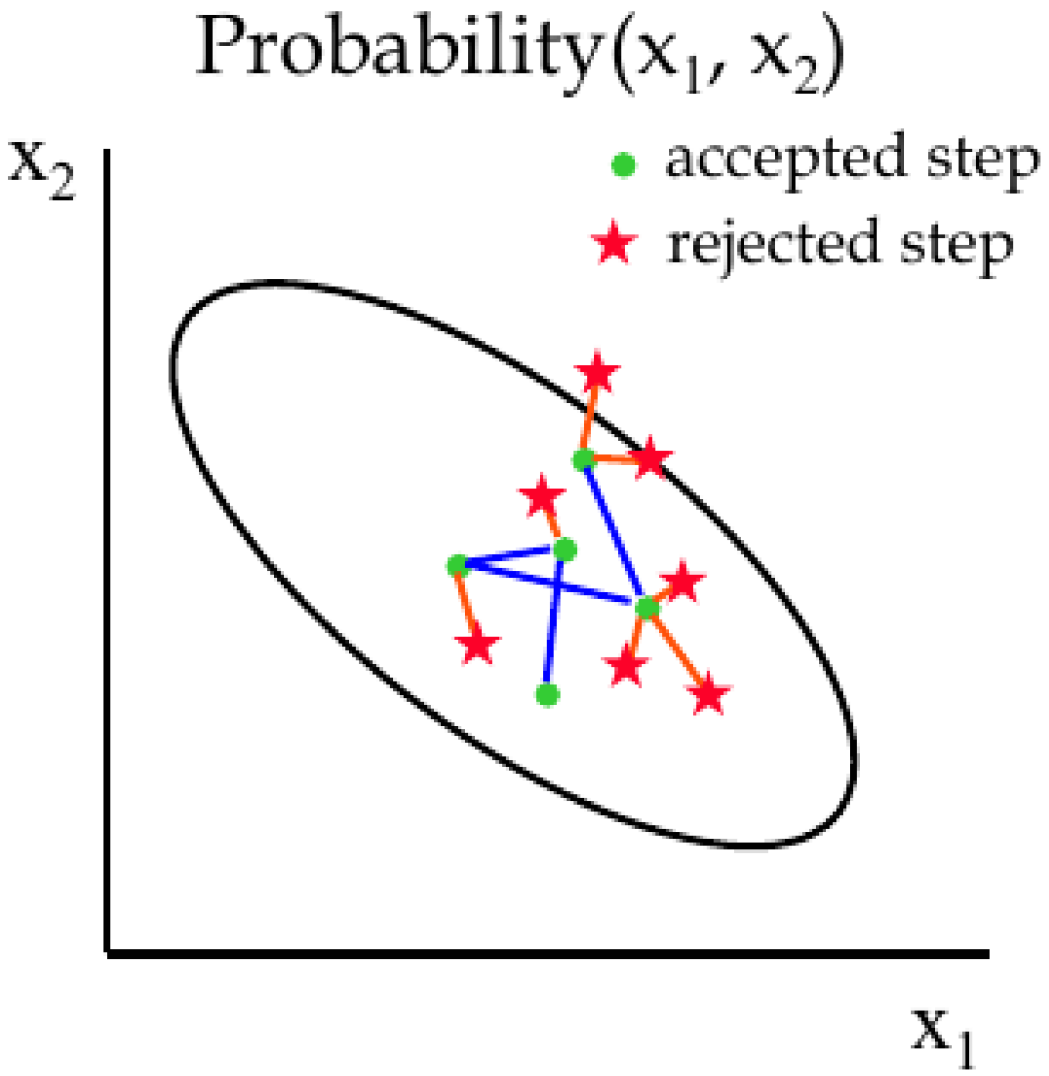
2. Erzeuge Zustand x_{n+1} mit $g(x_n \rightarrow x_{n+1})$

3. Akzeptiere neuen Zustand x_{n+1} mit $A(x_n \rightarrow x_{n+1})$
   Ja: verwende x_{n+1} ; Nein: verwende x_n

4. Wiederhole Schritt 2 bis 3 insgesamt T-Mal um Autokorrelationen zu vermeiden

5. Verwende x_n als Stichprobe aus P, und gehe zu Schritt 2

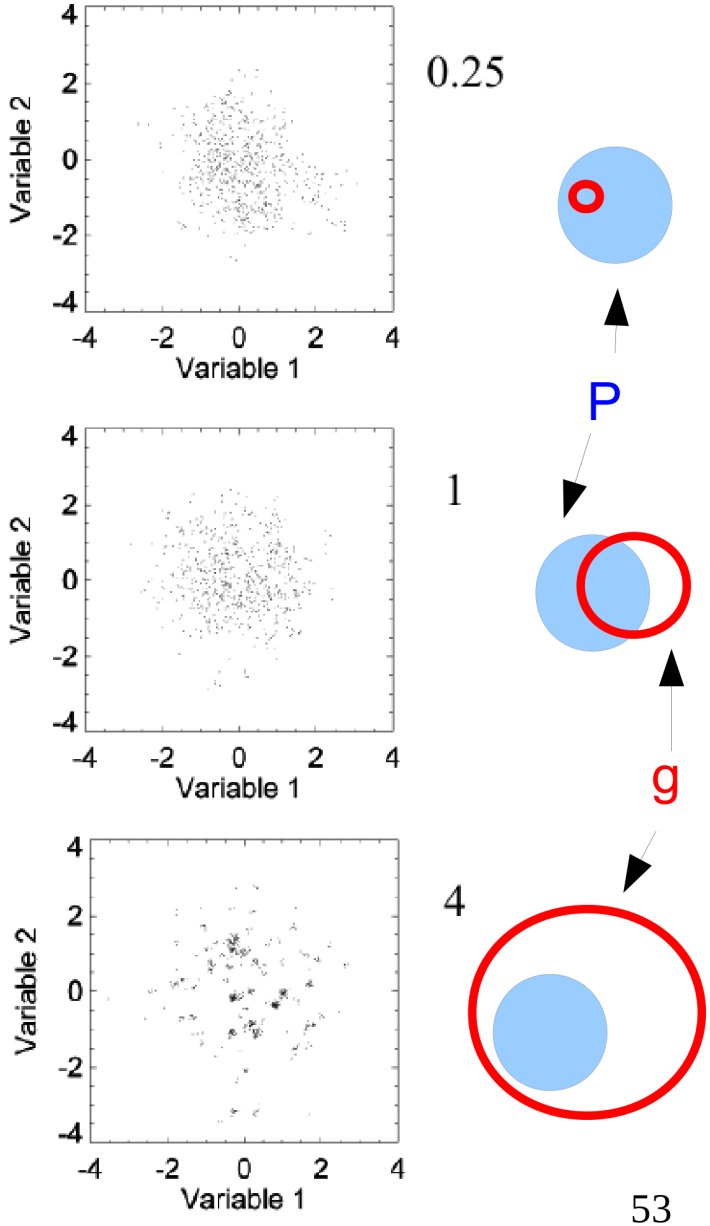
Akzeptanz



Einfluss der Vorschlagsfunktion

- Zu kleine Vorschlagsfunktion
  - große Akzeptanzrate ~1
  - starke Autokorrelationen
  - gute Abtastung
  - **langsame Konvergenz**

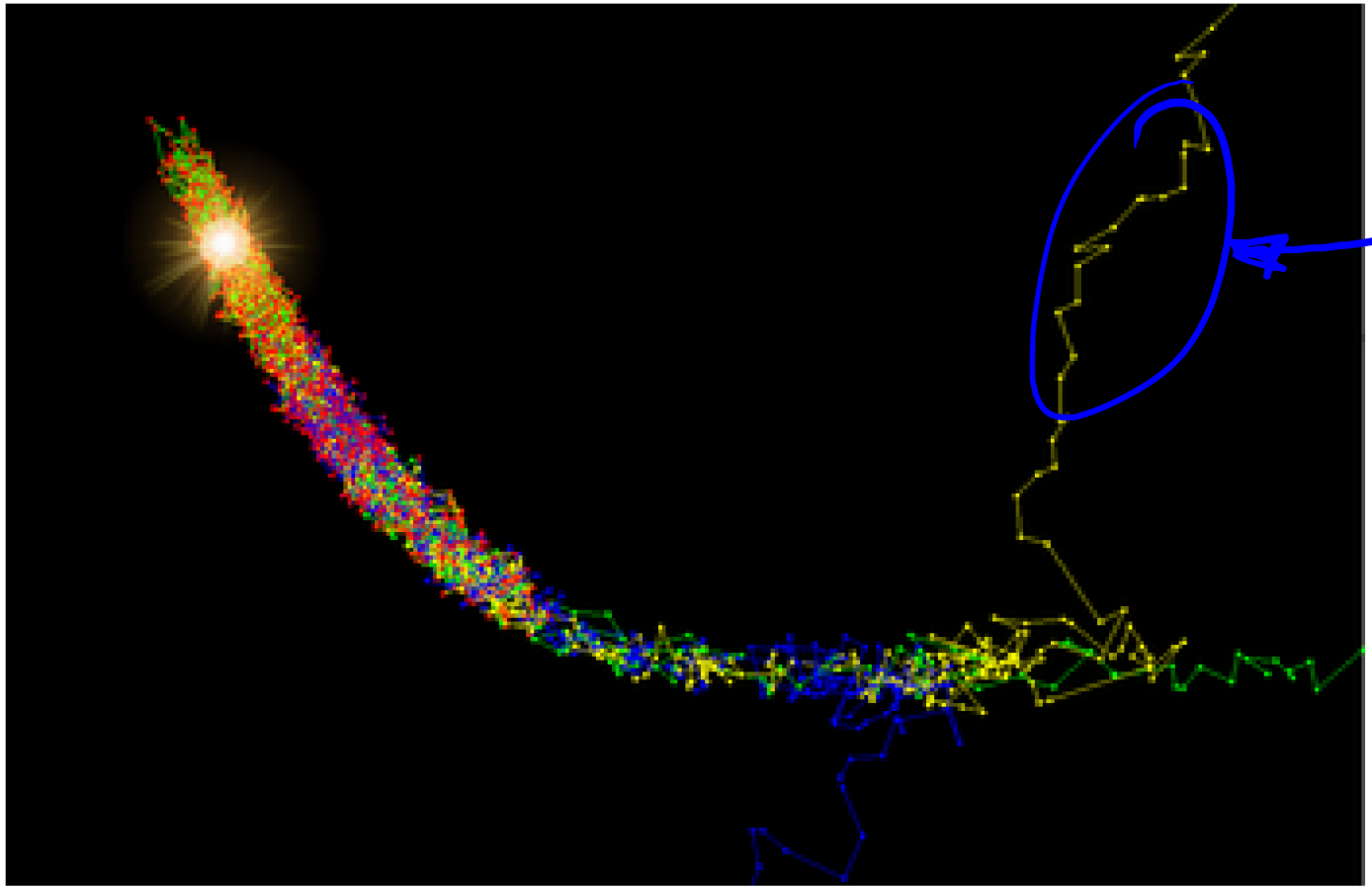
- Vorschlagsfunktion zu groß
  - kleine Akzeptanzrate → 0
  - kleine Autokorrelationen
  - schlechte Abtastung
  - **langsame Konvergenz**



Visualisierung von Markov Ketten

eigenes Studium

- Markov Ketten finden das Minimum der Rosenbrock Funktion



V7 - Metropolis Hastings.mp4

siehe auch Video

MCMC Vorschlagsfunktion

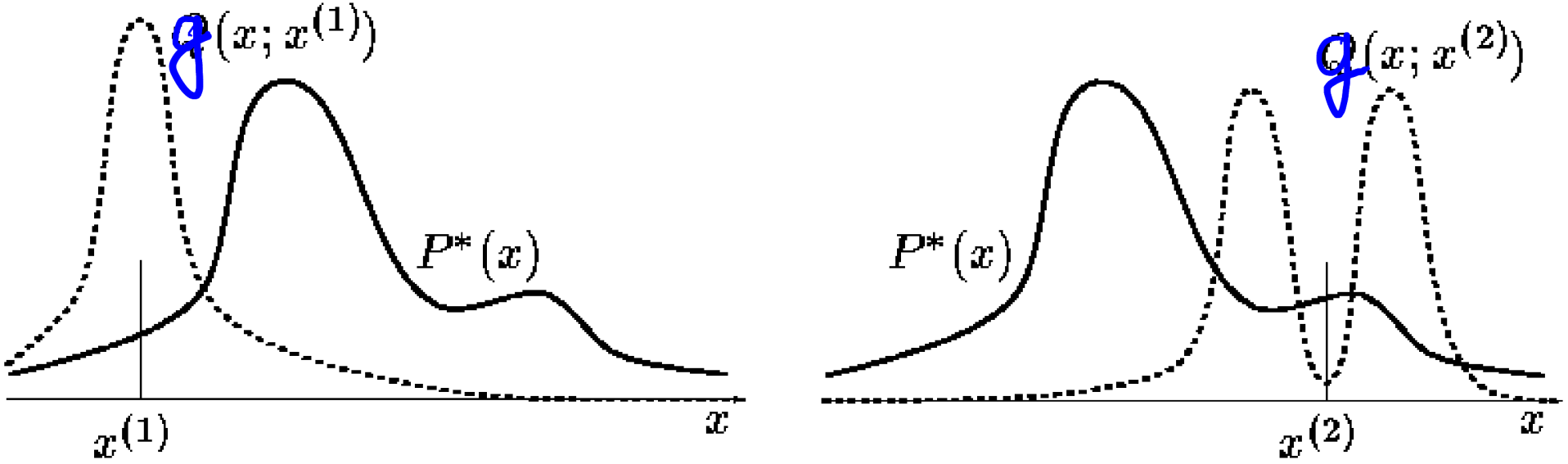
- Könnte man $g(j \to i) = P(i)$ verwenden, wäre die Akzeptanz-Rate:

$$A = \min\left[1,\ \frac{P(j)g(j \to i)}{P(i)g(i \to j)}\right] = \min\left[1,\ \frac{P(j)P(i)}{P(i)P(j)}\right] = 1$$

- Da das im Allgemeinen nicht machbar ist, wird oft eine multi-dimensionale (symmetrische) Gauß Verteilung verwendet. Die Wahl der Kovarianzen ist hier der einzige Freiheitsgrad, ist aber kritisch!

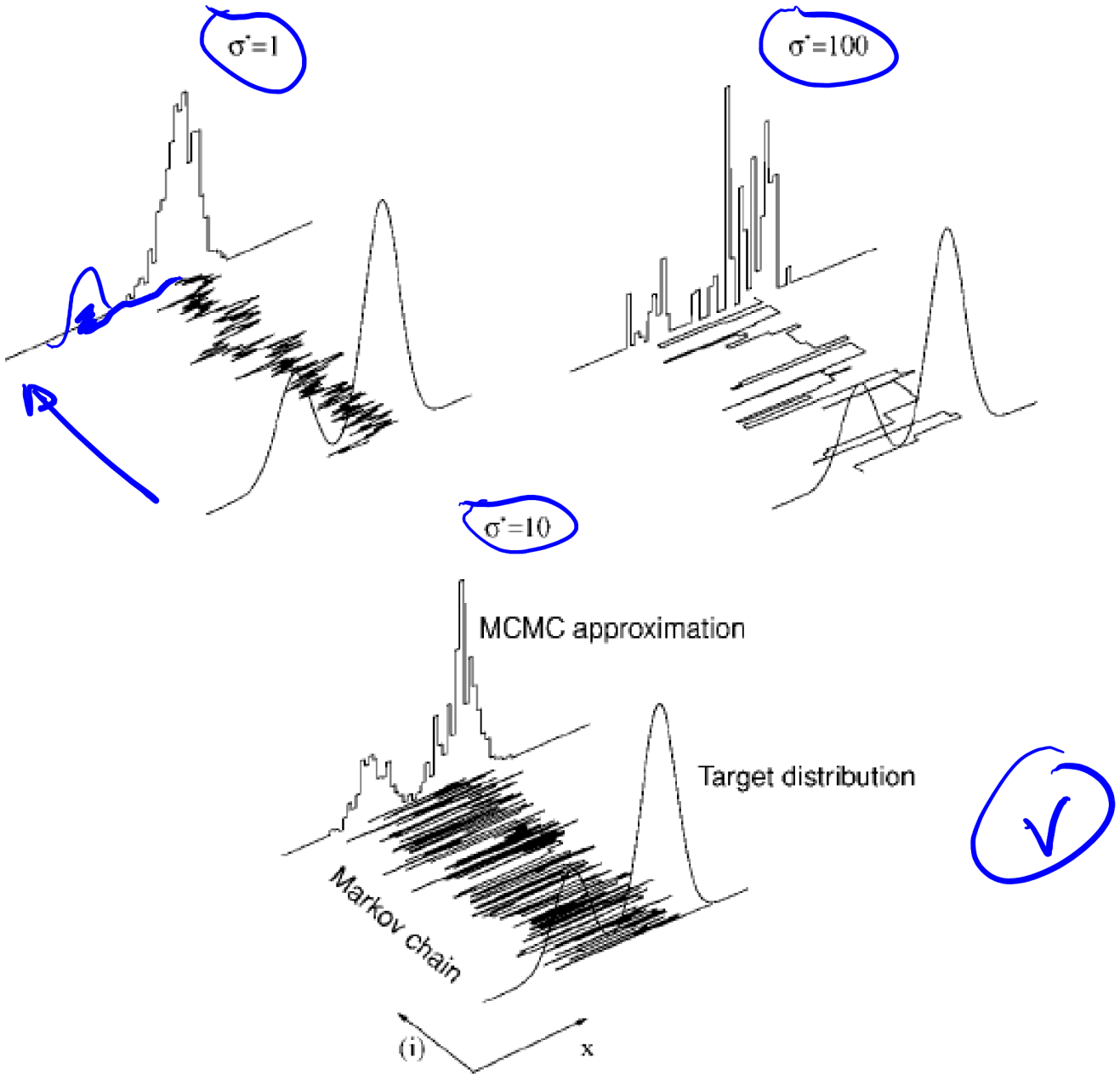
Vorschlagsfunktion $g(x_1 \rightarrow x_2)$

- In einer Dimension:



- Sehr schwierig zu Optimieren. Optimal sollte die Form der Funktion P(x) sehr nahe kommen, sonst ist die Akzeptanzrate schlecht. In vielen Dimensionen ist das schwierig.

Markov Chain Monte Carlo



Beurteilung der Vorschlagsfunktion

Relevante Parameter welche die Qualität der Vorschlagsfunktion quantifizieren können sind unter anderen:

- Akzeptanzrate

  Sollte nicht zu klein sein

- Autokorrelation der Kette

  Sollte möglichst schnell gegen 0 gehen

Parameteroptimierung mit „Ausglühen“

Wie kann eine Markov Kette zur Parameteroptimierung verwendet werden?

- Kirkpatrik, Gelett, Vecci 1983
- Auch: Simulated Annealing, Statistical Science 6 (1993) 10-15
  Bertsimas and Tsitsiklis

- Die „Kostenfunktion“ P($\vec{x}$) in beliebig vielen Dimensionen ist bekannt
- Transformiere die Funktion:

$$P(\vec{x}) \rightarrow P'(\vec{x}) = \exp\left[-\frac{P(\vec{x})}{T}\right]$$

- Und damit:

$$\frac{P_2'(\vec{x})}{P_1'(\vec{x})} = \exp\left[-\frac{P_2(\vec{x}) - P_1(\vec{x})}{T}\right]$$

Ausglüh-Algorithmus „*Simulated Annealing*“

1. Start mit einem zufälligen Zustand x_1 bei Temperatur T

2. Erzeuge Zustand x_{n+1} mit $g(x_n \rightarrow x_{n+1})$

3. Akzeptiere neuen Zustand x_{n+1} wenn $P(x_{n+1}) < P(x_n)$

   oder $c > \exp\left[-\frac{P(x_{n+1}) - P(x_n)}{T}\right]$

   mit c Zufallszahl aus [0,1]

4. Wiederhole Schritt 2 bis 3 insgesamt T-Mal bei gleicher Temperatur

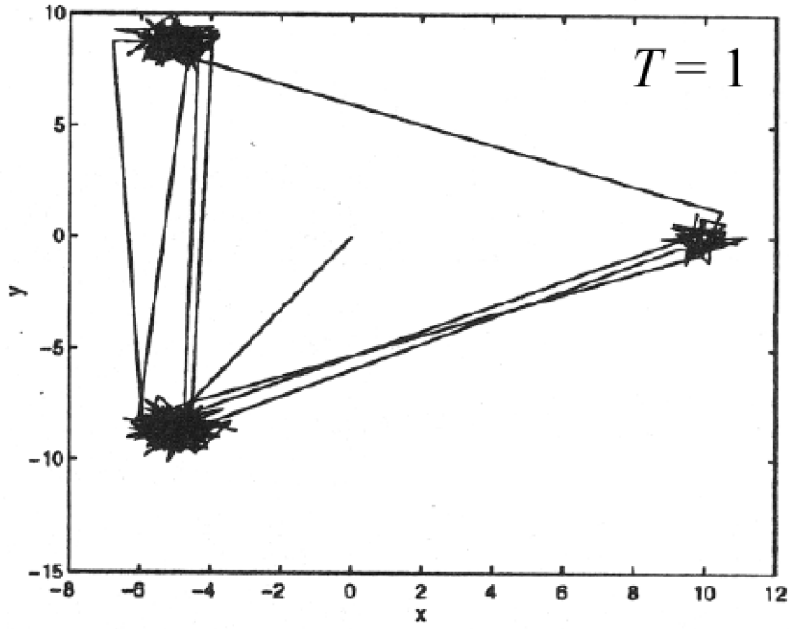
5. Kontrolliere Konvergenz

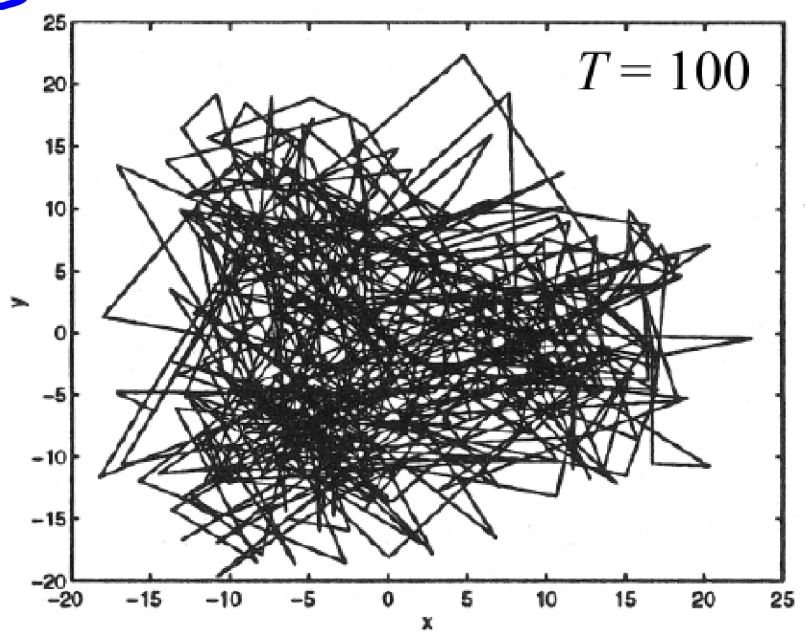
6. Reduziere Temperatur, und gehe zu Schritt 2

Effekt der Temperatur

klein

groß





Temperatur Änderungen „Cooling Schedule“

Viele Methoden in Verwendung. Verschiedene Vor- und Nachteile:

- linear: $T(t) = T_0 - \eta t$

je nach Anwendungsfall zu wählen ...

- exponentiell: $T(t) = T_0 \alpha^t$

sehr schneller Übergang zu "kalten" Temperaturen

- logarithmisch: $T(t) = \frac{c}{log(t+d)}$

c: Bis $t = -d$ schneller Temperaturabfall, danach

d: langsam

- Thermodynamic adaptive: $\frac{dT}{dt} = \frac{-v_s T}{\epsilon\sqrt{C} + \dots}$

v_s: Thermodynamic speed, C: heat capacity, e: Relaxation time

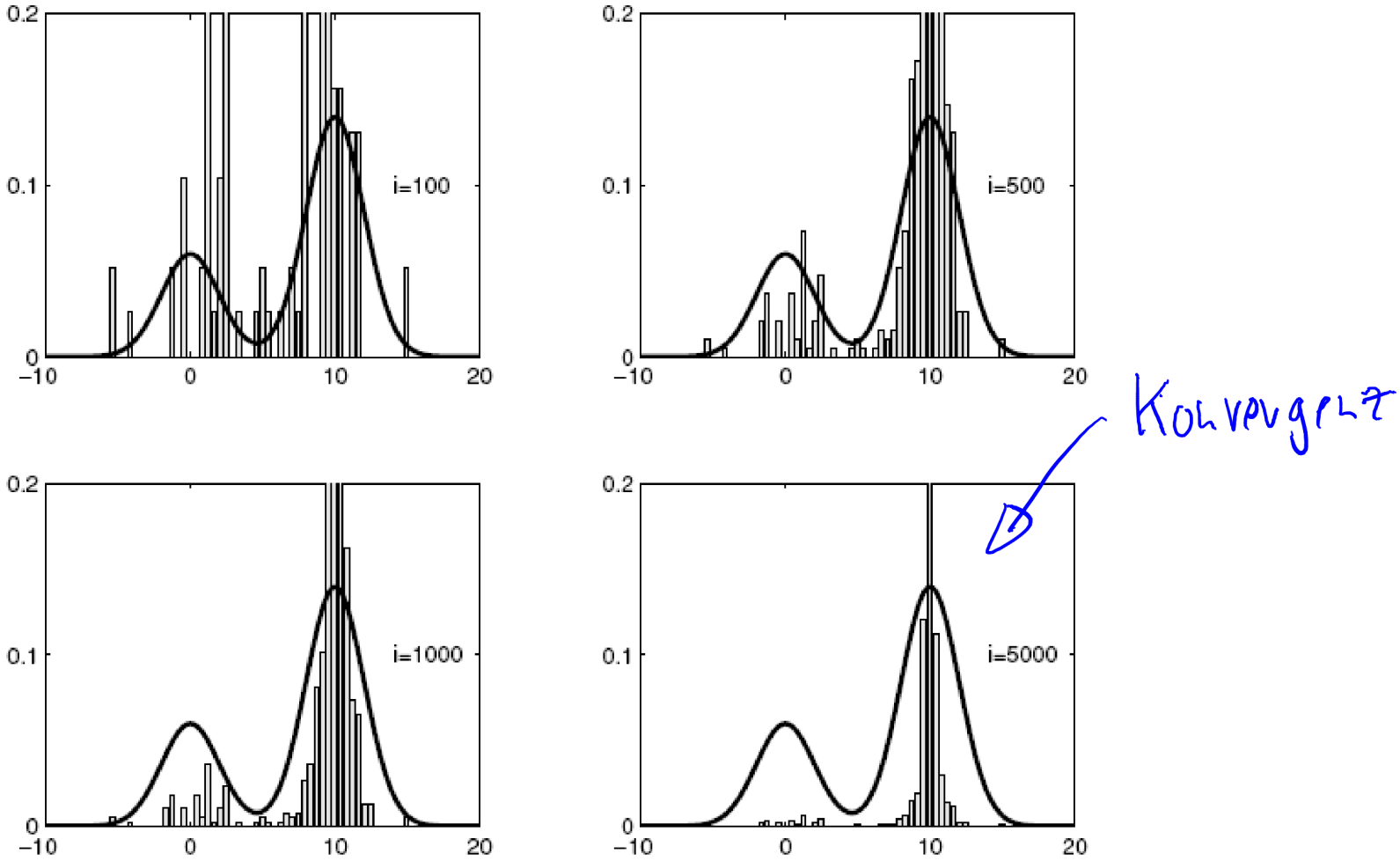
Quantities estimated from Markov Chain

Probleme

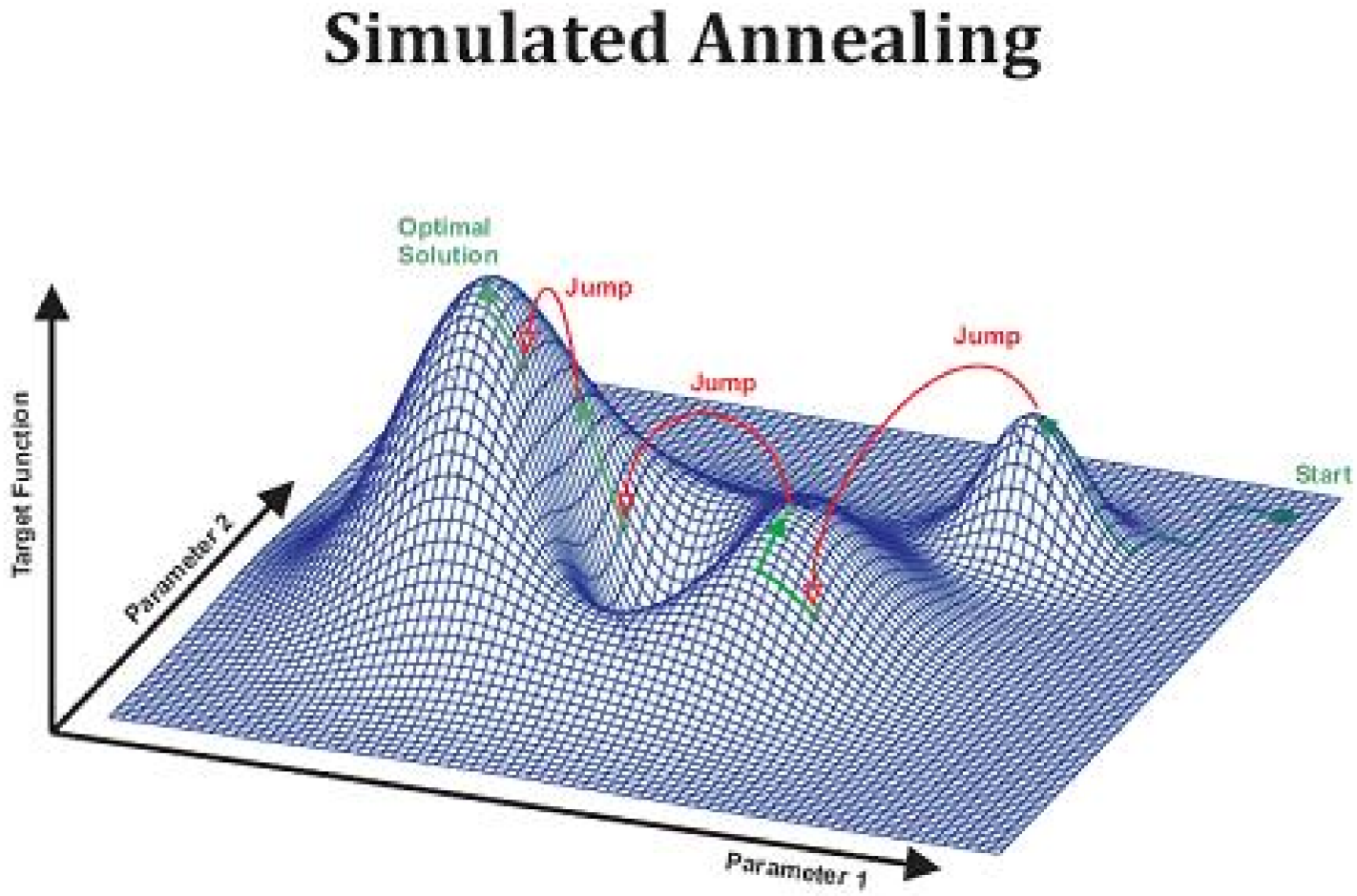
- Parametertuning sehr wichtig
  - Welche Starttemperatur?
    Zum Beispiel „aufheizen“ auf Akzeptanzrate 40-60%
  - Welche Abtastverteilung g verwenden?
  - Wie oft bei gleicher Temperatur Umgebung abtasten? Wie soll sich das mit der Zeit ändern?
  - Mit welcher Geschwindigkeit abkühlen?
  - Abbruchbedingung

Erfolgreiches „annealing“ benötigt Fein-Abstimmung und Kontrolle !

Simuliertes Ausglühen



Finde globales Optimum

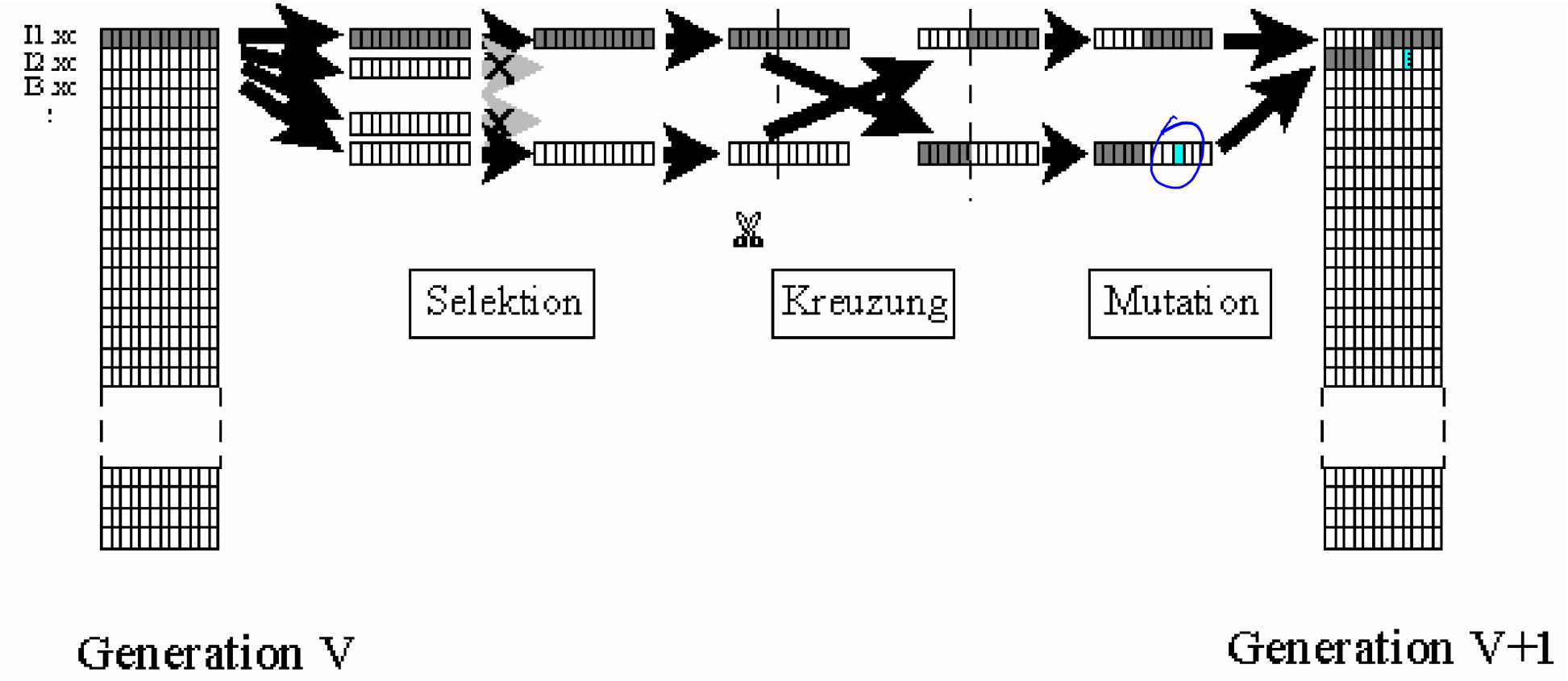


Genetische Algorithmen

An die Genetik angelehnt

- Parameter werden von den Eltern auf die Nachkommen vererbt, dabei kommt es zu:
  - Re-Kombination der Parameter
    Ganze Parameterblöcke werden von einer Generation an die nächste weitergegeben
  - Mutation
    Parameter können auch zufällige Veränderung erfahren
  - Nur die besten Nachkommen können sich weiter vermehren
  - etc.

Genetischer Algorithmus, schematisch



Algorithmus

1. Generiere zufällige Startpopulation P
2. Berechne Fitness der Individuen aus P
3. Selektiere Fitness
4. Wende genetische Operatoren an
5. Wiederhole Schritte 2 bis 4 bis Abbruchkriterium erfüllt
6. Bestes Individuum → Minimum

Kann einfacher zu beherrschen sein als „Annealing“. Allerdings auch hier viel Fein-Arbeit und spezifische Anpassung.

Beispiel

eigenes Studium

- Erzeugung eines Bildes aus einfachen Grundformen. Siehe Video.

V7 - Genetic.mp4

⇒ gleichzeitige Optimierung $\mathcal{O}(100)$ Parameter mit genetischem Algorithmus. Ein "normaler" Algorithmus hat hier keine Chance.

Aktives Forschungsfeld:

- Ant colony optimization
- Bees algorithmn
- Intelligent water drop algorithmn
- Particle swarm optimization
- Firefly algorithmn
-

Oft werden biologische Mechanismen kopiert: die Evolution ist an sich ein Optimierungs-Verfahren und bringt vielfältige selbst-optimierende Systeme zustande.

V7- Particle Swarm.mp4

⇒ gleichzeitige Optimierung eines ganzen Ensembles von Parametersätzen. Hier: kinetische Energie im "Potential" mit Reibungs-Term.

Video

Zusammenfassung, Markov Ketten

- MCMC können von nicht vollständig beschriebenen Verteilungen Zufallszahlen generieren. Damit können hoch-demensionale
  - Integrale gelöst werden
  - Bayesche Wahrscheinlichkeiten berechnet werden
  - Parameter optimiert werden, etc.

- In hoch-dimensionalen Parameterräumen mit teilweise vielen lokalen Minima könnend dies die einzigen brauchbaren Algorithmen sein

- Annealing und Genetische Algorithmen gehören mit zu den modernsten Methoden sehr komplexe Optimierungen durchzuführen

- Bei einer konkreten Anwendung muss sorgfältiges Parameter-Studium betrieben werden