Technische Universität Berlin
Institut für Softwaretechnik und Theoretische Informatik
Methoden der Künstlichen Intelligenz

Bayes'sche Inferenz in dynamischen Systemen mit Gaußprozessen

Diplomarbeit

von
Marcel Uhlich

Betreuer :
Professor Dr. Manfred Opper
Dr. Andreas Ruttor

Mitberichter :
Professor Dr. Klaus-Robert Müller

Eidesstattliche Erklärung

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Diplomarbeit selbständig verfasst und keine weiteren als die angegebenen Hilfsmittel verwendet habe.

Berlin, den 13. August 2010
Marcel Uhlich

Inhaltsverzeichnis

Kapitel 1

Grundlagen

1.1 Einleitung und Motivation

Modelle von physikalischen oder biologischen Prozessen, die vor allem dynamische Eigenschaften besitzen, lassen sich oft durch Differenzialgleichungssysteme beschreiben. Ein grundsätzliches Prinzip dieser Arbeit ist es, dass wir lediglich eine endliche Anzahl von Beobachtungen dieser Prozesse betrachten können. In der Regel sind diese Beobachtungen dann auch durch Rauschen gestört. Sind die Differenzialgleichungssysteme abhängig von Parametern, so sind wir daran interessiert, diese aus den Beobachtungen zu bestimmen.

Ein naiver Ansatz, dieses Inferenzproblem zu lösen, besteht darin die parameterabhängige Lösung des Systems zu bestimmen und dann den entsprechenden quadratischen Fehler zu minimieren. Kann man die Lösung des Differenzialgleichungssystems exakt formulieren und ist diese Lösungsfunktion bezüglich der Zeitkomplexität schnell zu berechnen, so kann der quadratische Fehler ein gute Methode zur Lösung des Inferenzproblems darstellen. Sind die Differenzialgleichungssysteme jedoch nicht exakt lösbar, so muss in jedem Iterationsschritt eines Lösungsalgorithmus die Lösung der Differenzialgleichung numerisch bestimmt

werden. Damit wird die Zeitkomplexität erheblich größer. Dies gilt für alle Verfahren, die die Lösung der Gleichungen benötigen.
Das Ziel dieser Arbeit ist ein Verfahren vorzustellen und zu erweitern, welches die Inferenz vollständig ohne explizite Lösung der Differenzialgleichungssysteme ermöglicht. Wir werden zunächst gewöhnliche Differenzialgleichungen betrachten und anschließend das Verfahren auf partielle Differenzialgleichungen erweitern. Danach wird versucht, mithilfe der zugrunde liegenden Ideen des Verfahrens eine Modifikation dessen herzuleiten.

1.2 Das Inferenzproblem

Wir nehmen an, wir haben eine Menge von Beobachtungen

$$X = \left\{X^{(t)} \in \mathbb{R}^n\right\}, Y = \left\{Y^{(t)} \in \mathbb{R}^m\right\} \text{ mit } t \in \{1, 2, ..., T\} \tag{1.1}$$

und

$$Y^{(t)} = f(X^{(t)}) + \varepsilon \,. \tag{1.2}$$

Die Funktion $f : \mathbb{R}^n \to \mathbb{R}^m$ beschreibt den beobachteten Prozess. Der Term ε beschreibt normalverteiltes Rauschen mit Mittelwert 0 und Kovarianz C. Dabei wird angenommen, dass das Rauschen unabhängig bezüglich der Komponenten ist. Es gilt also $p(Y|X) = \prod_{i=1}^{m} p(Y_i|X_i)$. Weiterhin wissen wir, dass der zugrunde liegende Prozess f einem Differenzialgleichungssystem gehorcht. Wir schreiben[1]

$$D^{k_l} Y_l(x) = g_l\left(f, \Theta, x\right) \; l \in \{1, ..., n\} \,. \tag{1.3}$$

Dabei bezeichnet D^{k_l} die Multiindexschreibweise für partielle Ableitungen. k_l ist ein Vektor mit nicht negativen ganzen Zahlen und es ist

$$D^{k_l} f(x) = \frac{\partial^{|k_l|} f(x)}{\partial^{(k_l)_1} x_1 \partial^{(k_l)_2} x_2 ... \partial^{(k_l)_n} x_n} \,. \tag{1.4}$$

[1] Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes, S. 2

Θ ist die Menge der Parameter des Differenzialgleichungssystems und $x \in D \subset \mathbb{R}^n$. Die Funktion $Y(x) = f(x)$ ist dann eine Lösung des Differenzialgleichungssystems. Der Ausdruck $g_l\left(f, \Theta, x\right)$ ist sehr allgemein gehalten und ist je nach Differenzialgleichung verschieden. Es können auch weitere Ableitungen in g vorkommen. Dies tritt vor allem bei partiellen Differenzialgleichungen auf.
Die Aufgabe ist es nun, die Parameter $\Theta = \{\theta_1, ..., \theta_p\}$ aus den Daten X, Y zu schätzen. Ein Beispiel ist

$$f'(t) = \begin{pmatrix} 0 & 1 \\ -\omega^2 & -\lambda \end{pmatrix} f(t)\,, \tag{1.5}$$

wobei die Parameter ω, λ zu schätzen sind. $f(t)$ ist hier eine vektorwertige Funktion $\mathbb{R} \to \mathbb{R}^2$. Die Daten der Menge X sind dann bestimmte Zeiten t, an denen die Beobachtungen gemacht werden. Y besteht aus den Auswertungen der Funktion f an den Stellen t mit additivem, normalverteiltem Rauschen. Die Funktion f ist im Allgemeinen nur durch die Daten, beziehungsweise durch das Differenzialgleichungssystem implizit beschrieben.

1.3 Gaußprozesse und Ableitungen

Gaußprozesse

Ein stochastischer Prozess $\left(X^{(t)}\right)_{t\in T}$ heißt Gaußprozess, wenn jede endliche Auswahl $X^{(t_1)}, ..., X^{(t_d)}$ multivariat normalverteit ist[2]. Wir schreiben $\mathcal{N}(\mu, C)$ für eine Normalverteilung mit Mittelwert μ und Kovarianz C. Die Kovarianzmatrix C wird durch eine parameterabhängige Kernfunktion $K : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ erzeugt. Es

[2]Gaussian Processes for Machine Learning, S. 13

ist $C_{ij} = K\left(X^{(i)}, X^{(j)}\right)$ und [3]

$$p\left(\phi_n, \sigma_n | Y_n^{(1:T)}\right) = \mathcal{N}\left(0, C_n + \sigma_n^2 I\right) \tag{1.6}$$

$$p\left(f_n\left(X_n^{(1:T)}\right) | Y_n^{(1:T)}, \phi_n, \sigma_n\right) = \mathcal{N}\left(\mu_n, \tilde{C}_n\right) \tag{1.7}$$

$$\mu_n = C_n(C_n + \sigma_n^2 I)^{-1} Y_n^{(1:T)} \tag{1.8}$$

$$\tilde{C}_n = \sigma_n^2 C_n (C_n + \sigma_n^2 I)^{-1} . \tag{1.9}$$

Dabei sind ϕ_n die Hyperparameter der Kernfunktion K und $Y_n^{(1:T)}, X_n^{(1:T)}$ diejenigen Vektoren, die die n-te Komponente der Daten X, Y über alle Zeiten $t \in \{1, ..., T\}$ enthalten. Der Mittelwert μ_n liefert dann eine Regressfunktion der n-ten Komponente des Prozesses $f(X)$ mit den zugehörigen Unsicherheiten $\tilde{C}_n$.

Kernfunktion

Innerhalb der gesamten Arbeit wird ausnahmslos der RBF-Kern

$$K(X_1, X_2) = \alpha \exp\left(-\frac{1}{2}(X_1 - X_2)^T D (X_1 - X_2)\right) \tag{1.10}$$

verwendet. Dabei ist D eine Diagonalmatrix mit $D_{ii} = \frac{1}{\beta_i^2}$. Die Menge $\{\alpha, \beta_1, ..\beta_n\}$ enthält dann die Hyperparameter des Kerns. Alpha ist ein Skalierungsfaktor, die verschiedenen Betas sind Längenskalen.
Bei großen Längenskalen ist die Kovarianz benachbarter Beobachtungspunkte groß, daher schwanken entsprechende Schätzungen oder Samples vom Gaußprozess in Umgebungen benachbarter Beobachtungen weniger stark als bei kleinen Längenskalen.

[3] Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes, S. 2

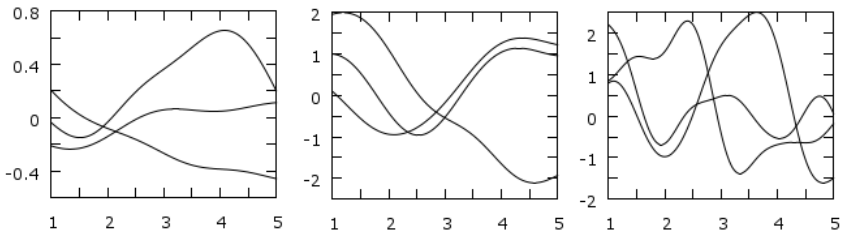

Abbildung 1.1: Samples vom Gaußprior

Abbildung (1.1) zeigt Samples vom Gaußprior für verschiedene α und β. Das erste Bild von links zeigt Samples mit $\alpha = 0.1, \beta = 1$, für das zweite Bild ist $\alpha = 1, \beta = 1$ und im dritten Bild ist $\alpha = 1, \beta = 0.5$. Man sieht deutlich, dass für eine kleinere Längenskala die Samplefunktionen stärker schwanken als für größere. Der Unterschied der Alpha macht sich im Maximum der Funktionswerte bemerkbar.

Ableitungen

Der Ableitungsoperator ist linear, und da lineare Transformation von Gaußprozessen wieder einen Gaußprozess liefert, ist die Ableitung eines Gaußprozesses ebenfalls ein Gaußprozess (siehe Anhang A). Mit Hilfe der Ableitung von Gaußprozessen kann man so auch Regression und Vorhersagen für die Ableitungen der beobachteten Funktionen mittels Gaußprozessen betrachten.
Es sei K der RBF-Kern, dann lassen sich die die Kovarianzen zwischen f und der partiellen Ableitungen von f wie folgt beschreiben[4]

$$\left(C_{f_n}\right)_{ij} = \quad D_1^{k_n} K\left(X^{(i)}, X^{(j)}\right) \tag{1.11}$$

$$\left(C^{f_n}\right)_{ij} = \quad D_2^{k_n} K\left(X^{(i)}, X^{(j)}\right) \tag{1.12}$$

$$\left(C_{f_n}^{f_n}\right)_{ij} = \quad D_1^{k_n} D_2^{k_n} K\left(X^{(i)}, X^{(j)}\right) \ . \tag{1.13}$$

[4]Gaussian Processes for Machine Learning, S. 191, (9.1)

Hierbei bezeichnet $D_k^{k_n}$ die partiellen Ableitungen nach dem ersten oder entsprechend dem zweiten Argument des Kerns. Mit Hilfe dieser Matrizen können wir die bedingte Verteilung von $D^{k_n} f_n$ gegeben f_n ausdrücken[5] :

$$p\left(D^{k_n} f_n(X^{1:T}) | f_n(X^{1:T})\right) = \mathcal{N}(m_n, K_n) \tag{1.14}$$

$$m_n = C_{f_n} \left(C_n + \sigma_n I\right)^{-1} f_n(X^{1:T}) \tag{1.15}$$

$$K_n = C_{f_n}^{f_n} - C_{f_n} \left(C_n + \sigma_n^2 I\right)^{-1} C^{f_n} . \tag{1.16}$$

Der Mittelwert m_n liefert dann eine Regressfunktion der entsprechenden partiellen Ableitung der n-ten Komponentenfunktion von f mit den zugehörigen Unsicherheiten K.

Ableitungen des RBF-Kerns

Wir betrachten zunächst nur partielle Ableitungen bis zum Grad 2. Es sei $s \in \{1, 2\}$ und $k \in \{1, ..., n\}$

$$\begin{aligned}
\frac{\partial K\left(X_1, X_2\right)}{\partial (X_s)_k} &= \frac{\partial \alpha \exp\left(-\frac{1}{2}\left(X_1 - X_2\right)^T D \left(X_1 - X_2\right)\right)}{\partial (X_s)_k} \\
&= K\left(X_1, X_2\right) \frac{\partial\left(-\frac{1}{2}\left(X_1 - X_2\right)^T D \left(X_1 - X_2\right)\right)}{\partial (X_s)_k} \\
&= K\left(X_1, X_2\right) \frac{\partial - \frac{1}{2} \sum_{p=1}^{n} \frac{1}{\beta_p^2} \left((X_1)_p - (X_2)_p\right)^2}{\partial (X_s)_k} \\
&= \frac{(-1)^s}{\beta_k^2} K\left(X_1, X_2\right) \left((X_1)_k - (X_2)_k\right) .
\end{aligned} \tag{1.17}$$

[5] Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes, S. 3

Sei nun weiterhin $t \in \{1,2\}, l \in \{1,2,...,n\}$, dann ist gemäß der Produktregel :

$$\begin{aligned}
\frac{\partial^2 K(X_1,X_2)}{\partial (X_s)_k \partial (X_t)_l} &= \frac{\partial \frac{(-1)^s}{\beta_k^2} K(X_1,X_2)\left((X_1)_k-(X_2)_k\right)}{\partial (X_t)_l} \\
&= \frac{(-1)^s}{\beta_k^2}\frac{(-1)^t}{\beta_l^2} K(X_1,X_2)\left((X_1)_l-(X_2)_l\right)\left((X_1)_k-(X_2)_k\right) \\
&+ \delta_{lk}\frac{(-1)^s}{\beta_k^2} K(X_1,X_2)(-1)^{t-1} \\
&= K(X_1,X_2)\left[\frac{(-1)^{s+t}}{\beta_k^2\beta_l^2}d + \delta_{kl}\frac{(-1)^{s+t-1}}{\beta_k^2}\right] \\
&= \frac{(-1)^{s+t}}{\beta_k^2} K(X_1,X_2)\left[\frac{1}{\beta_l^2}d - \delta_{kl}\right]. \qquad (1.18)
\end{aligned}$$

Dabei ist δ_{kl} das Kroneckerdelta und $d = \left((X_1)_l-(X_2)_l\right)\left((X_1)_k-(X_2)_k\right)$.

Testpunkte

Sei $x^* \in \mathbb{R}^n$ ein Testpunkt und sei k^* ein Vektor mit $k_i^* = K\left(x^*, X^{(i)}\right)$, dann ist[6]

$$\begin{aligned}
f_n^* &= (k^*)^T (C_n + \sigma_n^2 I)^{-1} Y^{1:T} \qquad (1.19) \\
\mathcal{V}(f_n^*) &= K(x^*,x^*) - (k^*)^T (C_n + \sigma_n^2 I)^{-1} k^* \qquad (1.20)
\end{aligned}$$

Die gemeinsame Verteilung von $f_n(X^{1:T})$ und $D^{k_n} f(x^*)$ ist gegeben durch

$$\mathcal{N}\left(0, \begin{pmatrix} C_{f_n}^{f_n} & C^{f_n} \\ C_{f_n} & C_n + \delta^2 I \end{pmatrix}\right).$$

Hierbei ist C_{f_n} ein Zeilenvektor und C^{f_n} ein Spaltenvektor mit

$$\begin{aligned}
(C_{f_n})_i &= D_1^{k_n} K\left(x^*, X^{(i)}\right) \\
(C^{f_n})_i &= D_2^{k_n} K\left(X^{(i)}, x^*\right).
\end{aligned}$$

[6]Gaussian Processes for Machine Learning, S.17

Weiterhin ist $C_{f_n}^{f_n} = D_1^{k_n} D_2^{k_n} K(x^*, x^*)$. Die Verteilung $p\left(D^{k_n} f_n(x^*) | f_n(X^{1:T})\right)$ ist dann wie in (1.15) und (1.16) gegeben durch[7]

$$\begin{aligned} p\left(D^{k_n} f_n(x^*) | f_n(X^{1:T})\right) = &\quad \mathcal{N}\left(D^{k_n} f_n(x^*), \mathcal{V}(D^{k_n} f_n(x^*))\right) \\ D^{k_n} f_n(x^*) = &\quad C_{f_n}(C_n + \sigma_n^2 I)^{-1} f_n(X^{1:T}) \qquad (1.21) \\ \mathcal{V}(D^{k_n} f_n(x^*)) = &\quad C_{f_n}^{f_n} - C_{f_n}(C_n + \sigma_n^2 I)^{-1} C^{f_n} \,. \qquad (1.22) \end{aligned}$$

In Abbildung (1.2) werden Regressfunktionen der Funktion $f(x) = \sin(x)$ (links)

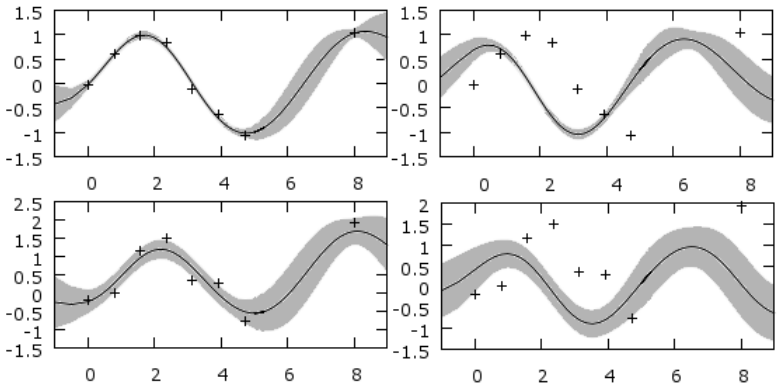

Abbildung 1.2: Regression und Vorhersagen mit Ableitungen

und die Vorhersagen für die Ableitungen gezeigt (rechts). Die Daten, dargestellt durch $+$, sind gemäß $y(x) = \sin(x) + \mathcal{N}(0, \sigma^2)$ erzeugt worden, wobei für die oberen Bilder $\sigma^2 = 0.1$ gilt und für die unteren $\sigma^2 = 0.5$. Für eine leichtere Vergleichbarkeit werden die Daten auch für die Vorhersagen der Ableitungen gezeigt. Die Vorhersagen wurden gemäß (1.19) und (1.21) berechnet, die Standardabweichung ist durch die Wurzel der Ausdrücke in (1.20) und (1.22) gegeben. Mit Hilfe der Standardabweichung wurden die Fehlerbalken (graue Schattierung) erzeugt indem zu den Vorhersagen jeweils die Standardabweichungen addiert und subtrahiert wurden. Die Ausdrücke in (1.21) und (1.22) sind gegeben $f(X^{1:T})$ berechnet

[7] Pattern Recognition and Machine Learning, S. 87, (2.81),(2.82)

worden, für die Vorhersage der Ableitung wurde daher $f(X^{1:T}) = \mu$ wie in (1.8) gesetzt.

1.4 Inferenz in Differenzialgleichungssystemen

Die Methode zur Parameterschätzung ist Hauptgegenstand der Arbeit *Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes*. Das Verfahren besteht aus drei Schritten.

Zunächst werden die Hyperparameter des Gaußprozesses gemäß (1.6) gelernt. Zusätzliches Wissen kann man durch ergänzende Faktoren $\pi(\phi_n), \pi(\sigma_n)$ in die Optimierung mit einbringen. Im Allgemeinen werden aber flache Priorverteilungen über die Hyperparameter angenommen.

Für die Schätzung der Hyperparameter wird, wenn nicht anders erwähnt, stets die negative log-likelihood von (1.6)

$$\begin{aligned} L = & -\ln\left(\frac{1}{(2\pi)^{\frac{T}{2}}\det(C_n+\sigma_n^2 I)^{\frac{1}{2}}}\exp\left[-\tfrac{1}{2}\left(Y^{1:T}\right)^T\left(C_n+\sigma_n^2 I\right)^{-1}Y^{1:T}\right]\right) \\ = & \quad \tfrac{1}{2}\left(\left(Y^{1:T}\right)^T\left(C_n+\delta^2 I\right)^{-1}Y^{1:T}+\tfrac{1}{2}\ln\det\left[C_n+\delta^2 I\right]\right)+c \end{aligned} \tag{1.23}$$

für alle n minimiert. In der Regel wird dazu die Funktion *fminsearch* von Matlab benutzt, welche den Simplexalgorithmus nach Nelder-Mead verwendet. Sollte ein anderes Verfahren verwendet werden, wird dieses an der entsprechenden Stelle erwähnt.

Nach der Bestimmung der Hyperparameter werden die Regressfunktionen für die Funktion f und die betrachteten Ableitungen berechnet. Als Erstes wird der Mittelwert μ_n gemäß (1.8) bestimmt und dann der Mittelwert (1.15) m_n gegeben μ_n berechnet.

In Schritt 3 werden nun die Modellparameter des Differenzialgleichungssystems

gelernt. Dazu wird die Verteilung

$$P(\Theta,\gamma) = \prod_{i=1}^{n} \exp\left(-\tfrac{1}{2}\left(g_i - m_i\right)^T \left(K_i + \gamma_i^2 I\right)^{-1} \left(g_i - m_i\right)\right) \quad (1.24)$$

betrachtet. γ ist ein Präzisionsparameter. Das Inverse der Präzision liefert dann ein zusätzliches, normalverteiltes Ableitungsrauschen. Auch hier kann zusätzliches Wissen durch entsprechende Priorverteilungen $\pi(\Theta)\pi(\gamma)$ modelliert werden. g_n ist wie in (1.3) definiert und bildet einen Vektor über alle X. Die Schätzung der Modellparameter ist abhängig vom Modell leicht bis schwierig. Es werden daher verschiedene Methoden verwendet, um die Parameter zu bestimmen. Die verwendeten Verfahren werden an den entsprechenden Stellen erläutert. Sämtliche Verfahren und Algorithmen in dieser Arbeit wurden in Matlab (R2007a) implementiert.

Kapitel 2

Gewöhnliche Differenzialgleichungen

Da sich jede gewöhnliche Differenzialgleichung n-ter Ordnung in ein äquivalentes Differenzialgleichungssystem erster Ordnung transformieren lässt, werden hier nur Differenzialgleichungssysteme erster Ordnung betrachtet. Im Falle der gewöhnlichen Differenzialgleichungen wird die Matlabfunktion *ode45* genutzt, um die Daten zu erzeugen.

2.1 Gedämpfte Schwingung

2.1.1 Likelihood

Wir betrachten das Differenzialgleichungssystem (1.5) mit Frequenzparameter ω und Dämpfungsparameter λ. Die Matrix in (1.5) bezeichnen wir mit A. Dann ist $g(t) = Af(t)$ mit $g_1(t) = f_2(t), g_2(t) = -\omega^2 f_1(t) - \lambda f_2(t)$. Die Parameterschätzung wird durch Minimierung der Log-likelihood von (1.24) durchgeführt. Diese

ist gegeben durch

$$\begin{aligned} l(\omega, \lambda, \gamma_1, \gamma_2) = & \ \tfrac{1}{2} (G_1 - m_1)^T (K_1 + \gamma_1^2 I)^{-1} (G_1 - m_1) \\ + & \ \tfrac{1}{2} (G_2 - m_2)^T (K_2 + \gamma_2^2 I)^{-1} (G_2 - m_2) \end{aligned}$$

mit $G_1 = \mu_2$ und $G_2 = -\omega^2 \mu_1 - \lambda \mu_2$. K_1, K_2, m_1, m_2 werden entsprechend (1.21) und (1.22) definiert. Man sieht, dass der erste Summand der negativen Log-likelihood nicht von den Parametern $\omega, \lambda, \gamma_2$ abhängt. Für die Optimierung ist daher nur der zweite Summand relevant. Damit reduziert sich die Likelihood zu

$$l(\omega, \lambda, \gamma) = \ \tfrac{1}{2} (G - m)^T (K + \gamma^2 I)^{-1} (G - m) \ . \tag{2.1}$$

Dabei ist $G = G_2, m = m_2, \gamma = \gamma_2$ und $K = K_2$. Diese Funktion wird durch einen Gradientenabstieg minimiert. Dazu berechnen wir zunächst den Gradienten der negativen Log-Likelihood

$$L(\Theta, \Gamma) = \frac{1}{2} \left(\sum_{i=1}^{n} (g_i(f, \Theta, X) - m_i)^T \left(K + \gamma_i^2 I \right)^{-1} (g_i(f, \Theta, X) - m_i) \right) \tag{2.2}$$

2.1.2 Gradient

Zunächst ist für eine quadratische Matrix A und einer differenzierbaren Funktion $f : \mathbb{R}^n \to \mathbb{R}^n$

$$\begin{aligned} \frac{\partial f(x)^T A f(x)}{\partial x_k} = & \quad \frac{\partial \sum_{i=1}^{n} \sum_{j=1}^{n} f_i(x) A_{ij} f_j(x)}{\partial x_k} \\ = & \quad \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial f_i(x)}{\partial x_k} A_{ij} f_j(x) + f_i(x) A_{ij} \frac{\partial f_j(x)}{\partial x_k} \\ = & \quad \frac{\partial f(x)^T}{\partial x_k} A f(x) + f(x)^T A \frac{\partial f(x)}{\partial x_k} \\ = & \quad f(x)^T A^T \frac{\partial f(x)}{\partial x_k} + f(x)^T A \frac{\partial f(x)}{\partial x_k} \\ = & \quad f(x)^T \left(A + A^T \right) \frac{\partial f(x)}{\partial x_k} \ . \end{aligned} \tag{2.3}$$

Für die Ableitung nach den Modellparametern $\theta_k \in \Theta$ gilt daher

$$\begin{aligned}\frac{\partial L}{\partial \theta_k} &= \frac{1}{2}\sum_{i=1}^{n}(g_i(f,\Theta,X)-m_i)^T \\ &* \left[\left(K+\gamma_i^2 I\right)^{-1}+\left(\left(K+\gamma_i^2 I\right)^{-1}\right)^T\right]\left(\frac{\partial g_i(f,\Theta,X)}{\partial \theta_k}\right).\end{aligned}$$

Da die Matrix K symmetrisch ist, ist auch $(K+\gamma_i^2 I)^{-1}$ symmetrisch, daher folgt für die Ableitung

$$\frac{\partial L}{\partial \theta_k} = \sum_{i=1}^{n}(g_i(f,\Theta,X)-m_i)^T\left(K+\gamma_i^2 I\right)^{-1}\left(\frac{\partial g_i(f,\Theta,X)}{\partial \theta_k}\right). \quad (2.4)$$

Für die Ableitung der Inversen gilt[1] $\frac{\partial A^{-1}}{\partial x} = -A^{-1}\frac{\partial A}{\partial x}A^{-1}$. Insbesondere ist mit $\frac{\partial A_{ij}^{-1}}{\partial x} = -\left(A^{-1}\frac{\partial A}{\partial x}A^{-1}\right)_{ij}$ also

$$\begin{aligned}\frac{\partial y^T A^{-1} y}{\partial x} &= \frac{\partial}{\partial x}\sum_{i=1}^{n} y_i (A^{-1}y)_i = \sum_{i=1}^{n}\sum_{j=1}^{n} y_i \frac{\partial A_{ij}^{-1}}{\partial x} y_j \\ &= -\sum_{i=1}^{n}\sum_{j=1}^{n} y_i \left(A^{-1}\frac{\partial A_{ij}}{\partial x}A^{-1}\right)_{ij} y_j = -y^T A^{-1}\frac{\partial A}{\partial x}A^{-1}y\,. \quad (2.5)\end{aligned}$$

Es wurde implizit angenommen, dass der Vektor y nicht von x abhängt. Insgesamt ergibt sich damit für die Ableitungen von (2.1)

$$\frac{\partial l}{\partial \omega} = -(G-m)^T\left(K+\gamma^2 I\right)^{-1}(2\omega\mu_1) \quad (2.6)$$

$$\frac{\partial l}{\partial \lambda} = -(G-m)^T\left(K+\gamma^2 I\right)^{-1}(\mu_2) \quad (2.7)$$

$$\frac{\partial l}{\partial \gamma} = -(G-m)^T\left(K+\gamma^2 I\right)^{-1}(K+2\gamma I)\left(K+\gamma^2 I\right)^{-1}(G-m)\,. \quad (2.8)$$

2.1.3 Versuchsaufbau und Auswertung

Für jeden Parametersatz werden 100 Durchläufe berechnet. In jedem Durchlauf werden die Daten gemäß der Lösung der Differenzialgleichung erzeugt und der

[1] Gaussian Processes for Machine Learning, S. 202

Gaußprozess an die Daten angepasst, wobei die Anfangsbedingungen $f_1(0) = 1$ und $f_2(0) = 1$ sind. Die Daten werden äquidistant von $T = 0$ bis $T = 10$ berechnet. Die Startwerte für die Anpassung der Hyperparameter werden festgehalten bei $(0.1, 0.1, 0.1)^T$. Für die Optimierung der Modellparameter werden die Startwerte für den Gradientenabstieg $(1, 1, 1)^T$ gesetzt. Die Wahl der Startwerte ist bei einem Gradientenabstiegsverfahren kritisch. In den Tests hat sich obiger Startwert als sinnvoll für dieses Problem herausgestellt. Alternativ kann man den Gradientenabstieg auch mit mehreren, verschiedenen Startwerten und anschließendem Vergleich durchführen.

ω	$\hat{\omega}$	$\mathcal{S}(\hat{\omega})$	λ	$\hat{\lambda}$	$\mathcal{S}(\hat{\lambda})$	Rauschen	Daten
1.0	0.9922	0.0633	0.1	0.0409	0.2813	$(0.05, 0.05)$	20
1.0	1.1731	2.9915	0.1	-0.3154	1.5114	$(0.5, 0.5)$	20
2.0	2.0604	0.0405	0.5	0.4950	0.0788	$(0.05, 0.05)$	20
2.0	8.9074	0.6055	0.5	7.6371	0.7592	$(0.05, 0.05)$	10
2.0	2.0661	0.0506	0.5	0.5117	0.1078	$(0.05, 0.05)$	16

Abbildung 2.1: Versuchsauswertung - gedämpfte Schwingung

Abbildung (2.1) zeigt Testläufe mit verschiedenen Parametersätzen. Die Variablen $\hat{\omega}, \hat{\lambda}$ sind die Mittelwerte des entsprechenden Parameters über 100 Durchgänge. Es ist zu beachten, dass der Parameter ω für einige Versuche gleich dem entsprechenden Startwert ist. Die Likelihood ist symmetrisch bezüglich ω, daher wurde von den negativen Approximationen von ω stets der Betrag betrachtet.
Negative Werte für λ wurden nicht verändert und sind entsprechend in die Mittelwerte mit eingegangen. Priorverteilungen über λ können aber ohne Weiteres die Negativität verhindern. Die Testläufe sind jedoch ohne Priorverteilungen berechnet worden. Problematisch sind negative λ, da die Dämpfung dann zu einer Verstärkung wird und das Verhalten der Lösung sich vollständig umkehrt.
Die Parameter in der Rauschenspalte bezeichnen die Standardabweichung des additiven Rauschens. Die Spalte Daten beschreibt die Anzahl der verwendeten Da-

tenpunkte.

Für kleines Beobachtungsrauschen sieht man gute Ergebnisse. Für größeres Rauschen sind die Ergebnisse wie erwartet schlechter. Vor allem der negative Mittelwert der Schätzungen λ im Versuch der zweiten Zeile ist problematisch.

Höhere Präzision

Verbesserungen für die Schätzung der Parameter bei stärkerem Beobachtungsrauschen können erzielt werden, indem man die Startwerte für die Präzision höher wählt.

Parameter	Mittelwert	Stichprobenstandardabweichung
$\omega = 1.0$	0.9905	0.0205
$\lambda = 0.1$	0.0947	0.0478
$\sigma_1 = 0.1$	0.0964	0.0213
$\sigma_2 = 0.1$	0.0893	0.0328
$\omega = 1.0$	0.6479	0.2679
$\lambda = 0.1$	0.0116	0.1574
$\sigma_1 = 1.0$	0.6123	0.4234
$\sigma_2 = 1.0$	0.6338	0.3957
$\omega = 2.0$	2.0135	0.0487
$\lambda = 0.2$	0.1794	0.0897
$\sigma_1 = 0.1$	0.1218	0.003
$\sigma_2 = 0.1$	0.0743	0.0331

Abbildung 2.2: Versuchsauswertung - gedämpfte Schwingung - $\gamma_0 = 100$

Abbildung (2.2) zeigt Versuche mit höherer Präzision. Der Startwert für die Präzision ist 100. Auch für kleineres Beobachtungsrauschen wurde beobachtet, dass die Qualität der Approximation weit höher als für niedrigere Präzision war.

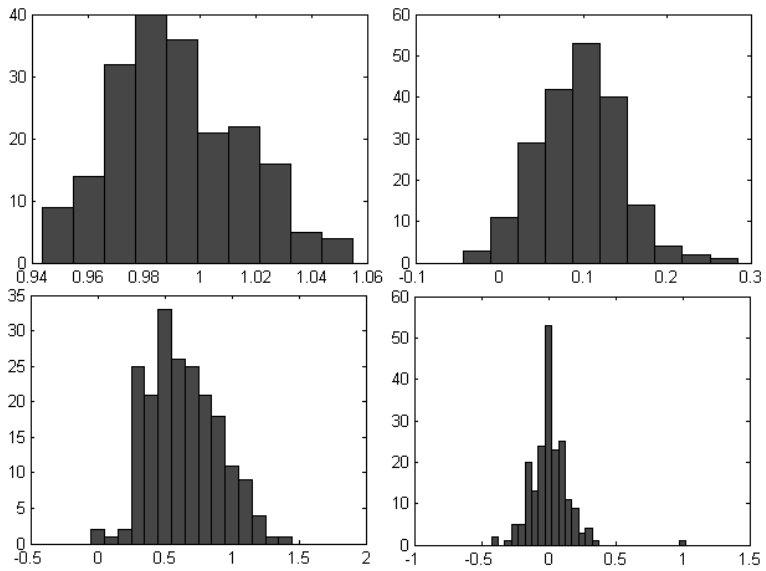

Abbildung 2.3: Histogramme - höhere Präzision

In Abbildung (2.3) werden 4 Histogramme für die Parameter $\lambda = 0.1, \omega = 1$ gezeigt. Die linken Bilder zeigen jeweils die Histogramme für den Parameter ω, während die rechten Bilder λ zeigen. Die oberen Histogramme wurden mit einem Beobachtungsrauschen mit Standardabweichung 0.1 erzeugt, während die unteren mit 1.0 erzeugt wurden. Alle Berechnungen wurden mit 20 Datenpunkten durchgeführt. Für die Histogramme wurden 200 Durchläufe berechnet.

Ist das Beobachtungsrauschen klein genug, werden die Parameter sehr gut gefunden. Für stärkeres Rauschen ist erwartungsgemäß die Schätzung der Parameter weit schwieriger. Allerdings ist eine Standardabweichung von 1 für die Lösung der Differenzialgleichung auch ein sehr großes Rauschen. Trotzdem sind die Ergebnisse auch für starkes Beobachtungsrauschen weit besser bei hoher Präzision als im vorherigen Versuch.

Die Schätzung der Parameter hängt entschieden von der Güte der Approximationen μ des beobachteten Prozesses ab. Nehmen wir an, μ und m wären exakte

Schätzungen des zugrunde liegenden Prozesses und dessen Ableitung. Für die echten Modellparameter Θ ist dann $m - g(\mu, \Theta, x) = 0$. Da K die Kovarianzmatrix eines Gaußprozesses ist, ist K positiv definit und damit auch $K + \gamma^2 I$. Daher sind die Modellparameter in diesem Fall ein Minimum der negativen Log-Likelihood. Jedoch sind es, wie wir später noch sehen werden, nicht die einzigen Minima.

Je stärker das Beobachtungsrauschen ist, um so schwieriger wird es, eine gute Regressfunktion für den Prozess und damit auch die Modellparameter zu bestimmen.

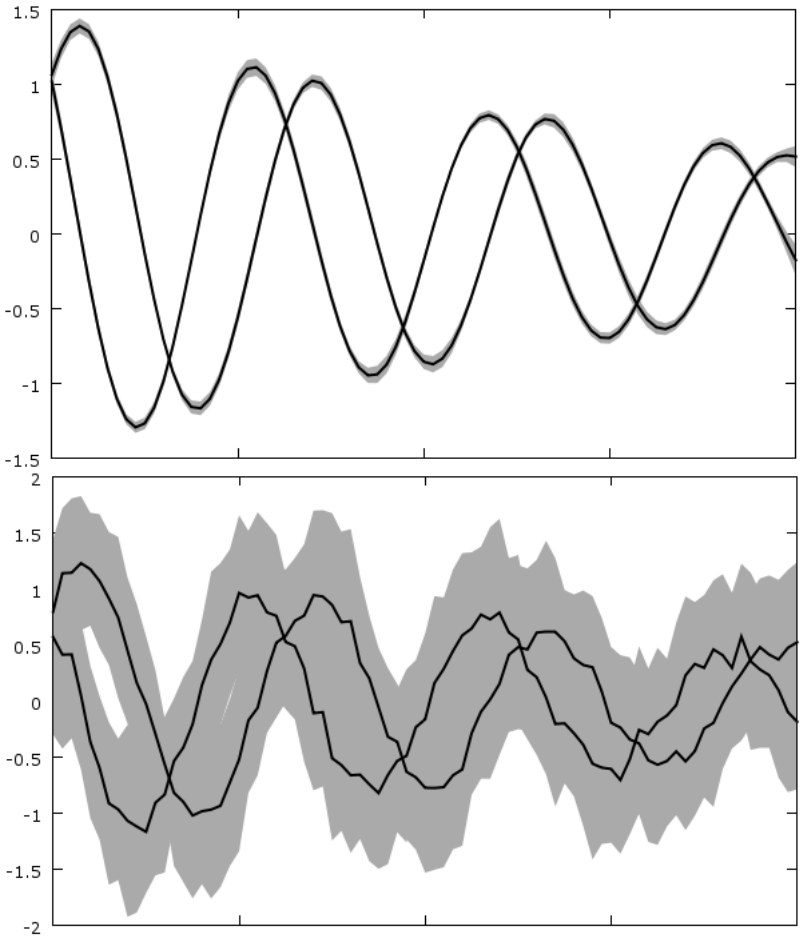

Abbildung 2.4: Schätzungen des zugrundeliegenden Prozesses

Abbildung (2.4) zeigt Mittelwerte und Standardabweichung der Regressionsfunktionen über 100 Testläufe. Dabei wurde das obere Bild mit einem Beobachtungsrauschen mit Standardabweichung von 0.1 berechnet und das untere entsprechend mit Standardabweichung 1.0. Die schwarzen Kurven beschreiben den Mittelwert der Schätzfunktionen über die Testläufe. Die Mittelwerte wurden für jeden Beobachtunszeitpunkt $t \in T$ bestimmt. Die Fehlerbalken wurden erzeugt, indem die Stichprobenstandardabweichungen für jeden Zeitpunkt $t \in T$ zu den entsprechenden Mittelwert addiert bzw. davon subtrahiert wurden. Wie erwähnt ist die Schätzung der μ_n für das Verfahren enorm wichtig. Daher sind die schwachen Ergebnisse für hohes Beobachtungsrauschen leicht durch die stark schwankenden Schätzungen μ_n zu erklären.

2.2 Lorenzgleichungen

Edward N. Lorenz formulierte ein Differenzialgleichungssystem auf der Grundlage von idealisierter Konvektion von Flüssigkeiten nach Rayleigh[2]. Das Modell beschreibt eine rotierende Konvektionsströmung einer Flüssigkeit, die gleichmäßig erhitzt wird. Das Differenzialgleichungssystem ist gegeben durch

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} a(y-x) \\ x(b-z)-y \\ xy-cz \end{pmatrix} . \tag{2.9}$$

Der Parameter a beschreibt die Prandtlzahl, b und c repräsentieren ein Verhältnis von $\triangle T$ zu einer Dämpfung, die durch Viskosität und thermaler Leitfähigkeit erzeugt wird. Es ist $g_1(t) = a(f_2(t) - f_1(t)), g_2(t) = f_1(t)(b - f_3(t)) - f_2(t)$ und $g_3(t) = f_1(t)f_2(t) - cf_3(t)$. Die Lösungen dieses Systems können abhängig

[2]Nonlinear dynamics and Chaos, S. 208

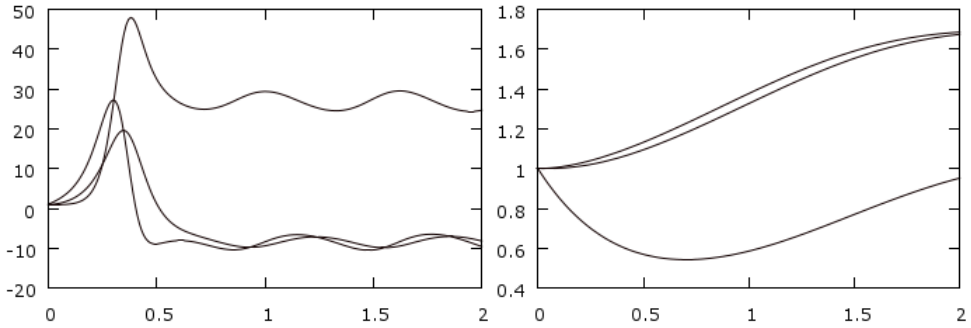

Abbildung 2.5: Lösungen der Lorenzgleichung

von den Parametern stark variieren. Abbildung (2.5) zeigt zwei Lösungen für die Parameter $a = 10, b = 28, c = 8/3$ (links) und $a = 10, b = 2, c = 8/3$ (rechts).

2.2.1 Likelihood und Gradient

Die negative Log-Likelihood gemäß (1.24) ist gegeben durch

$$\begin{aligned} l(a,b,c,\gamma_1,\gamma_2,\gamma_3) = & \quad \tfrac{1}{2}\left((G_1-m_1)^T\left(K_1+\gamma_1^2 I\right)^{-1}(G_1-m_1)\right) \\ & + \quad \tfrac{1}{2}\left((G_2-m_2)^T\left(K_2+\gamma_2^2 I\right)^{-1}(G_2-m_2)\right) \\ & + \quad \tfrac{1}{2}\left((G_3-m_3)^T\left(K_3+\gamma_3^2 I\right)^{-1}(G_3-m_3)\right) \ , \end{aligned} \tag{2.10}$$

wobei $G_1 = a * (\mu_1 - \mu_2), G_2 = \mu_1 \otimes (b - \mu_3) - \mu_2, G_3 = \mu_1 \otimes \mu_2 - c\mu_3$ ist. Mit $\otimes$ wird hier das komponentenweise Produkt von Vektoren bezeichnet. Ist weiterhin x eine Zahl und v ein Vektor, so definieren wir die verkürzte Schreibweise $x \pm v$ als $x * 1_v \pm v$, wobei 1_v der Vektor mit der gleichen Dimension wie v ist, dessen Komponenten überall 1 sind. Die Likelihood wird wieder durch einen Gradientenabstieg minimiert. Der Gradient ist nach (2.4) und (2.5) gegeben durch

$$\begin{aligned} \frac{\partial l}{\partial a} &= (G_1-m_1)^T\left(K_1+\gamma_1^2 I\right)^{-1}(\mu_1-\mu_2) \\ \frac{\partial l}{\partial b} &= (G_2-m_2)^T\left(K_2+\gamma_2^2 I\right)^{-1}\mu_1 \\ \frac{\partial l}{\partial c} &= -(G_3-m_3)^T\left(K_3+\gamma_3^2 I\right)^{-1}\mu_3 \ . \end{aligned}$$

Die partiellen Ableitungen nach γ_i berechnet man analog zu (2.8).

2.2.2 Versuchsaufbau und Auswertung

Es werden wieder feste Parametersätze definiert und für jeden Durchgang die Daten gemäß (2.9) erzeugt, wobei die Anfangsbedingungen $x(0) = 1, y(0) = 1$ und $z(0) = 1$ sind. Die Daten werden äquidistant zwischen $T = 0$ und $T = 1$ erzeugt. Abhängig von den Parametern wird die Größenordnung des Beobachtungsrauschens angepasst, da die Lösungen der Differenzialgleichung, wie man in Abbildung (2.4) sieht, in der Größenordnung auch stark schwanken.
Insgesamt werden 100 Durchgänge pro Parametersatz berechnet. Die Startwerte der Modellparameter werden normalverteilt gezogen, während die Startwerte für die Präzision wieder auf 100 gesetzt werden. Abbildung (2.6) zeigt einige

a	$\hat{a}$	$\mathcal{S}(\hat{a})$	b	$\hat{b}$	$\mathcal{S}(\hat{b})$	c	$\hat{c}$	$\mathcal{S}(\hat{c})$	Rauschen
10	9.95	0.46	28	28.05	0.72	$8/3$	2.65	0.47	(0.1,0.1,0.1)
10	8.22	2.2	28	23.87	6.62	$8/3$	2.48	1.05	(1,1,1)
20	19.96	0.51	15	14.99	0.04	10	10.03	0.04	(0.1,0.1,0.1)
20	10.67	5.24	15	12.73	5.05	10	8.5	3.73	(1,1,1)
10	9.63	1.05	2	2.01	0.01	$8/3$	2.63	0.03	(0.01,0.01,0.01)
10	3.41	6.95	2	2.62	6.7	$8/3$	2.61	1.6	(0.1,0.1,0.1)

Abbildung 2.6: Versuchsauswertung - Lorenzgleichungen

Versuche mit den Lorenz-Gleichungen. Sämtliche dieser Versuche wurden mit 20 Datenpunkten berechnet. Insgesamt sind die Resultate der Parameterschätzungen als gut einzustufen.

2.3 Das Lotka Volterra Modell

Bei dem Lotka Volterra Modell handelt es sich um ein Räuber-Beute Modell. Wir betrachten den Fall, dass es nur zwei Spezies gibt. Die Differenzialgleichung ist

gegeben durch[3]

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x(a-by) \\ -y(c-dx) \end{pmatrix} \tag{2.11}$$

x beschreibt hier die Population der Beute, während y die Population der Raubtiere als Funktion der Zeit bezeichnet. Der Parameter a beschreibt die Zuwachsrate der Spezies x pro Zeit, b ist eine Zerfallsrate, die sowohl Zerfall durch Spezies y als auch andere Umstände beschreibt. c beschreibt eine Zerfallsrate der Population y und d entsprechend eine Zuwachsrate.
Es ist $g_1(t) = f_1(t)(a - bf_2(t))$ und $g_2(t) = -f_2(t)(c - df_1(t))$.

2.3.1 Likelihood und Gradient

Die negative Log-Likelihood gemäß (1.24) ist gegeben durch

$$\begin{aligned} l(a,b,c,d,\gamma_1,\gamma_2) = & \quad \tfrac{1}{2}\left((G_1-m_1)^T\left(K_1+\gamma_1^2 I\right)^{-1}(G_1-m_1)\right) \\ & + \quad \tfrac{1}{2}\left((G_2-m_2)^T\left(K_2+\gamma_2^2 I\right)^{-1}(G_2-m_2)\right) \ , \end{aligned} \tag{2.12}$$

wobei wie üblich $G_1 = \mu_1 \otimes (a - b\mu_2)$ und $G_2 = -\mu_2 \otimes (c - d\mu_1)$ sind. Die Likelihood wird durch einen Gradientenabstieg minimiert. Der Gradient ist nach (2.4) und (2.5) gegeben durch

$$\begin{aligned} \frac{\partial l}{\partial a} &= \quad (G_1-m_1)^T\left(K_1+\gamma_1^2 I\right)^{-1}\mu_1 \\ \frac{\partial l}{\partial b} &= -(G_1-m_1)^T\left(K_1+\gamma_1^2 I\right)^{-1}\mu_1 \otimes \mu_2 \\ \frac{\partial l}{\partial c} &= \quad -(G_2-m_2)^T\left(K_2+\gamma_2^2 I\right)^{-1}\mu_2 \\ \frac{\partial l}{\partial d} &= \quad (G_2-m_2)^T\left(K_2+\gamma_2^2 I\right)^{-1}\mu_1 \otimes \mu_2 \ . \end{aligned}$$

Die partiellen Ableitungen nach γ_i berechnet man analog zu (2.8).

[3] Elements of physical biology, S.92

2.3.2 Versuchsaufbau und Auswertung

Auch hier werden wieder Mittelwert und Standardabweichung über 100 Durchläufe pro Parametersatz bestimmt. Abbildung (2.7) zeigt die Versuche mit dem

R = (0.1,0.1), S = (20,4)	$a = 1$	$b = 0.2$	$c = 0.5$	$d = 0.1$
Mittelwert	1.1203	0.21804	0.3976	0.0795
Standardabweichung	0.4435	0.0468	0.0533	0.0104
R = (1,1), S = (20,4)	$a = 1$	$b = 0.2$	$c = 0.5$	$d = 0.1$
Mittelwert	0.5724	0.1161	0.1948	0.0380
Standardabweichung	0.2326	0.0414	0.0661	0.0129
R = (0.1,0.1), S = (8,8)	$a = 0.8$	$b = 0.1$	$c = 0.6$	$d = 0.4$
Mittelwert	0.7985	0.1069	0.6355	0.4189
Standardabweichung	0.1860	0.0201	0.0370	0.0188
R = (1,1), S = (8,8)	$a = 0.8$	$b = 0.1$	$c = 0.6$	$d = 0.4$
Mittelwert	0.3808	0.0506	0.5641	0.3318
Standardabweichung	0.3139	0.0393	0.2264	0.1032

Abbildung 2.7: Versuchsauswertung - Lotka Volterra Modell

Lotka Volterra Modell. R steht dabei für das Beobachtungsrauschen und S bezeichnet die Anfangwerte der Differenzialgleichung. In diesen Versuchen wurde mit 40 Datenpunkten gerechnet. Auch diese Versuche zeigen gute Resultate, für einige Parameter gab es aber auch bei kleinem Beobachtungsrauschen leichte Abweichungungen. Zusammenfassend kann man sagen, dass für sämtliche Differenzialgleichungssysteme die Modellparameter gut geschätzt werden konnten. Im Verhältnis zu den Parametern b, c, d konnten wir allerdings den Parameter a weniger genau schätzen. Dies wird sich auch in späteren Versuchen zeigen.

Kapitel 3

Partielle Differenzialgleichungen

Wir sprechen im Folgenden wieder von Gaußprozessen. Hängt der Gaußprozess von mehr als nur der Zeit ab spricht man aber auch von *Gaussian Random Fields*.

3.1 Die Likelihoodfunktion

Im Gegensatz zu gewöhnlichen Differenzialgleichungen ist bei partiellen Differenzialgleichungen nicht von vornherein klar, wie die negative Log-Likelihood von (1.24) aufzustellen ist. Man betrachte beispielsweise die partielle Differenzialgleichung

$$\frac{\partial f}{\partial t} = -\lambda \frac{\partial^2 f}{\partial^2 x} \,. \tag{3.1}$$

Es stellt sich nun die Frage, welche Ableitung man für die Likelihood verwendet. Es ist möglich, sowohl $m_t \approx \frac{\partial f}{\partial t}$ als auch $m_x \approx \frac{\partial^2 f}{\partial^2 x}$ zu wählen. Dann sind $G_t = -\lambda m_x$ und $G_x = -\frac{1}{\lambda} m_t$. Für die Möglichkeiten der negativen Log-Likelihood

ergeben sich

$$
\begin{aligned}
l_1(\lambda,\gamma) &= (G_t - m_t)^T \left(K_t + \gamma^2 I\right)^{-1} (G_t - m_t) && (3.2)\\
l_2(\lambda,\gamma) &= (G_x - m_x)^T \left(K_x + \gamma^2 I\right)^{-1} (G_x - m_x) && (3.3)\\
l_3(\lambda,\gamma_1,\gamma_2) &= l_1(\lambda,\gamma_1) + l_2(\lambda,\gamma_2)\,. && (3.4)
\end{aligned}
$$

Ableitung der Kernfunktion

Die Kovarianzmatrix K_x in Ausdruck (3.3) wird nicht durch die bisherige Betrachtung der Ableitungen des RBF-Kerns beschrieben. Es ist nach (1.16) $K_x = C_{fx}^{fx} - C^{fx}(C+\sigma^2 I)^{-1} C_{fx}$. Die Matrizen C_{fx}, C^{fx} sind durch (1.18) gegeben. Mit $K_{ij} := K\left(\begin{pmatrix} t_i \\ x_i \end{pmatrix}, \begin{pmatrix} t_j \\ x_j \end{pmatrix}\right)$ gilt für die Matrix C_{ft}^{ft}

$$
\begin{aligned}
\left(C_{fx}^{fx}\right)_{ij} &= \frac{\partial^4 K_{ij}}{\partial^2 x_i \partial^2 x_j}\\
&\overset{1.17}{=} \frac{\partial^2}{\partial^2 x_j}\frac{1}{\beta_2^2} K_{ij}\left(\frac{(x_i-x_j)^2}{\beta_2^2} - 1\right)\\
&\overset{1.16}{=} \frac{\partial}{\partial x_j}\frac{1}{\beta_2^4} K_{ij}(x_i-x_j)\left(\frac{(x_i-x_j)^2}{\beta_2^2} - 1\right)\\
&- \frac{\partial}{\partial x_j}\frac{1}{\beta_2^2} K_{ij}\frac{2(x_i-x_j)}{\beta_2^2}\\
&= \frac{\partial}{\partial x_j}\frac{1}{\beta_2^2} K_{ij}\left(\frac{(x_i-x_j)^3}{\beta_2^4} - \frac{3(x_i-x_j)}{\beta_2^2}\right)\\
&\overset{1.16}{=} \frac{1}{\beta_2^4} K_{ij}\left(\frac{(x_i-x_j)^4}{\beta_2^4} - \frac{3(x_i-x_j)^2}{\beta_2^2}\right)\\
&+ \frac{1}{\beta_2^2} K_{ij}\left(\frac{3}{\beta_2^2} - \frac{3(x_i-x_j)^2}{\beta_2^4}\right)\\
&= \frac{1}{\beta_2^2} K_{ij}\left(\frac{(x_i-x_j)^4}{\beta_2^6} - \frac{6(x_i-x_j)^2}{\beta_2^4} + \frac{3}{\beta_1^2}\right)\,. \qquad (3.5)
\end{aligned}
$$

Lösung der partiellen Differenzialgleichung

Sofern möglich werden die untersuchten partiellen Differenzialgleichungssysteme explizit gelöst und die Daten dann entsprechend den Lösungsfunktionen erzeugt. Für die Gleichung (3.1) wird der Produktansatz verfolgt. Nehmen wir also an $f(t,x) = X(t)Y(x)$, dann ist

$$\begin{aligned} & \frac{\partial f}{\partial t} && = -\lambda \frac{\partial^2 f}{\partial^2 x} \\ \Leftrightarrow \quad & X'(t)Y(x) && = -\lambda X(t)Y''(x) \\ \overset{X(t)\neq 0}{\Leftrightarrow} \quad & \frac{X'(t)}{X(t)}Y(x) && = -\lambda Y''(x) \\ \overset{Y(x)\neq 0}{\Leftrightarrow} \quad & \frac{X'(t)}{X(t)} && = -\lambda \frac{Y''(x)}{Y(x)} \,. \end{aligned}$$

Da die linke Seite der Gleichung von t unabhängig ist und die rechte Seite von x unabhängig ist, gibt es eine Konstante c mit $-\lambda\frac{Y''(x)}{Y(x)} = c = \frac{X'(t)}{X(t)}$. Daraus ergibt sich folgendes System gewöhnlicher Differenzialgleichungen

$$\begin{aligned} X'(t) &= cX(t) \\ Y''(x) &= -\frac{c}{\lambda}Y(x) \,. \end{aligned}$$

Lösungen dieses Systems sind gegeben durch $X(t) = \exp(ct)$ und $Y(x) = \sin\left(\sqrt{\frac{c}{\lambda}}x\right)$. Dann ist $f(t,x) = \exp(ct)\sin\left(\frac{c}{\lambda}x\right)$ für $\frac{c}{\lambda} \geq 0$ eine Lösung von Differenzialgleichung (3.1). Zur Erzeugung der Daten wird der Spezialfall $c = \lambda$ betrachtet.

Vergleich der Likelihoodfunktionen

Zum Vergleich der Likelihoodfunktionen werden die Parameter wieder in 100 Versuchen geschätzt. Die Standardabweichung des Beobachtungsrauschens wird auf 0.05 gesetzt. Für die Optimierung wird wieder ein Gradientenabstieg benutzt.

Nach (2.4) sind die Gradienten durch

$$\frac{\partial l_1}{\partial \lambda} = -\left(G_t - m_t\right)\left(K_t + \gamma^2 I\right)^{-1} m_x \tag{3.6}$$

$$\frac{\partial l_2}{\partial \lambda} = \left(G_x - m_x\right)\left(K_x + \gamma^2 I\right)^{-1} \frac{1}{\lambda^2} m_t \tag{3.7}$$

$$\frac{\partial l_3}{\partial \lambda} = \frac{\partial l_1}{\partial \lambda} + \frac{\partial l_2}{\partial \lambda} \tag{3.8}$$

gegeben. Die partiellen Ableitungen nach γ berechnet man analog zu (2.5). In jedem Versuch werden für alle 3 Likelihoods die gleichen Daten und Startwerte für den Gradientenabstieg verwendet. Abbildung (3.1) zeigt einige Ergebnisse

	λ	$\hat{\lambda}$	$\mathcal{S}(\hat{\lambda})$
l_1	1	0.9389	0.0480
l_2	1	-0.3840	1.3310
l_3	1	0.9115	0.0718
l_1	-1	-0.9735	0.3336
l_2	-1	0.5237	1.6350
l_3	-1	-0.9265	0.2224

Abbildung 3.1: Vergleich der Likelihoodfunktionen

der Untersuchung. Die Unterschiede zwischen l_1 und l_2 sind deutlich zu sehen. l_3 hat in den Versuchen nur eine geringfügige Abweichung gegenüber l_1 gezeigt. Die Konsequenz daraus ist zunächst nicht dass l_2 die schlechtere Likelihood ist, vielmehr bedeutet es, dass man die Optimierungsverfahren anpassen muss.
Beispielsweise besitzt die Funktion l_2 an der Stelle $\lambda = 0$ eine Singularität, die bei einem Gradientenabstieg das Ergebnis maßgeblich beeinflussen kann.

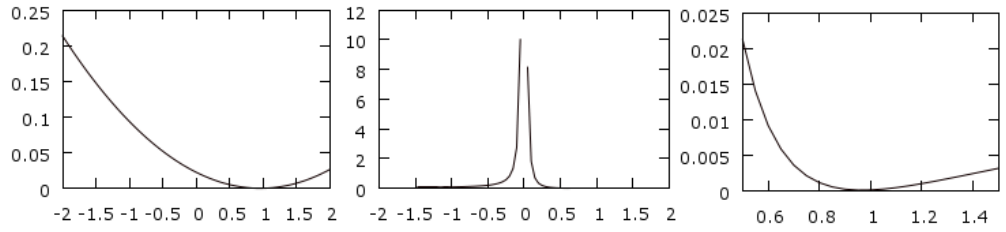

Abbildung 3.2: Likelihoodgraphen

Abbildung (3.2) zeigt die Likelihoodfunktionen l_1 und l_2 als Funktion von λ mit $\gamma = 100$. Die Daten wurden in diesen Beispielen mit $\lambda = 1$ und einem Beobachtungsrauschen mit Standardabweichung 0.1 erzeugt.

Das erste Bild zeigt l_1, das zweite Bild l_2 und das dritte Bild ebenfalls l_2 in einer anderen Skalierung. Man sieht, dass auch l_2 zumindest ein lokales Minimum im Bereich des echten Parameters besitzt.

Wir wiederholen den Versuch mittels Gradientenabstieg, wobei die Startwerte jetzt die Beträge normalverteilter Zufallszahlen bzw. negative Zufallszahlen sind, falls die entsprechenden Parameter negativ sind. Insgesamt wurden die Ergeb-

	λ	$\hat{\lambda}$	$\mathcal{S}(\hat{\lambda})$
l_1	1	0.9391	0.0712
l_2	1	0.6985	0.2627
l_3	1	0.8933	0.1240
l_1	-1	-0.9733	0.0884
l_2	-1	-0.7259	0.3127
l_3	-1	-0.9055	0.1886

Abbildung 3.3: Vergleich der Likelihoodfunktionen (2)

nisse der Parameterschätzungen durch die Likelihood l_2 verbessert, wie man in Abbildung (3.3) sehen kann, während für die anderen beiden Funktionen die Ergebnisse unwesentlich beeinflusst wurden. Festhalten kann man, dass die Likelihoodfunktionen l_1, l_2 und l_3 nicht ohne Weiteres als äquivalent zu betrachten sind.

Simulated Annealing

Als Konsequenz aus den Ergebnissen aus Abbildung (3.1) und (3.3) wird derselbe Versuch mit dem Simulated Annealing Verfahren durchgeführt. Der Vorteil gegenüber dem Gradientenabstieg ist, dass so lokale Minima weniger Probleme bereiten. Der Algorithmus ist gegeben durch[1]

Algorithmus 1

Initialisiere λ_0 , γ_0 *und* T

while *Konvergenzkriterium nicht erfüllt* **do**

wähle $\lambda = \mathcal{N}(\lambda_t, \sigma_1)$ *und* $\gamma = \mathcal{N}(\gamma_t, \sigma_2)$

if $l_i(\lambda, \gamma) < l_i(\lambda_t, \gamma_t)$ **then**

setze $\lambda_{t+1} = \lambda$ *und* $\gamma_{t+1} = \gamma$

else

wähle U *gleichverteilt aus* $[0, 1]$

if $U < \min\left\{1, \exp\left(-\frac{(l_i(\lambda,\gamma) - l_i(\lambda_t,\gamma_t))}{T}\right)\right\}$ **then**

setze $\lambda_{t+1} = \lambda$ *und* $\gamma_{t+1} = \gamma$

else

setze $\lambda_{t+1} = \lambda_t$ *und* $\gamma_{t+1} = \gamma_t$

endif

endif

setze $T \mapsto \varepsilon T$

end while

Für den Testlauf wird $\varepsilon = 0.99$ gesetzt und das Konvergenzkriterium mit $T < 1e - 10$ beschrieben. λ_0, γ_0 werden Normalverteilt und T mit 1 initialisiert, wobei

[1] Simulation and the Monte Carlo Method, S. 191

λ_0 auf das gleiche Vorzeichen wie der echte Parameter gesetzt wird. Für diesen Versuch werden nur noch l_1 und l_2 betrachtet, da die Abweichungen von l_1 und l_3 nur gering sind. Abbildung (3.4) zeigt deutliche Verbesserungen für die Li-

	λ	$\hat{\lambda}$	$\mathcal{S}(\hat{\lambda})$
l_1	1	0.9332	0.0293
l_2	1	0.9501	0.0306
l_1	-1	-0.9499	0.0491
l_2	-1	-0.9948	0.0557

Abbildung 3.4: Vergleich der Likelihoodfunktionen (3)

kelihood l_2, allerdings wurde auch hier wieder A-Priori-Wissen genutzt, indem wir das Vorzeichen der zu schätzenden Parameter festgelegt haben. Diesen Umstand könnte man umgehen, indem man für jeden Parameter sowohl mit negativem als auch mit positivem Startwert optimiert und dann den Schätzwert mit der niedrigeren Likelihood auswählt. Allerdings müsste man dann jede mögliche Vorzeichenkombination der Parameter berechnen, was bei N Parametern in 2^N Optimierungsprozessen enden würde. Für eine große Anzahl an Parametern wäre das also sehr aufwändig.

3.2 Eine Reaktions-Diffusions-Gleichung

Wir betrachten die partielle Differenzialgleichung

$$\frac{\partial C}{\partial t} = D\frac{\partial^2 C}{\partial x^2} - \omega C \tag{3.9}$$

, welche die klassische Reaction-Diffusion Rate Equation beschreibt[2]. D ist dabei der sogenannte Diffusionskoeffizient und ω beschreibt eine Zerfallsrate.

[2]Master equation simulation analysis of immunostained Bicoid morphogen gradient

Lösung der partiellen Differenzialgleichung

Lösungen dieser Differenzialgleichung kann man wieder mit dem Produktansatz bestimmen. Nehmen wir also $C(t,x) = X(t)Y(x)$ an, dann ist

$$\begin{aligned}
& \frac{\partial C}{\partial t} && = D\frac{\partial^2 C}{\partial x^2} - \omega C \\
\Leftrightarrow \quad & X'(t)Y(x) && = DX(t)Y''(x) - \omega X(t)Y(x) \\
\Leftrightarrow \quad & (X'(t) + \omega X(t))Y(x) && = DX(t)Y''(x) \\
\underset{Y(x)\neq 0}{\Leftrightarrow} \quad & X'(t) + \omega X(t) && = DX(t)\frac{Y''(x)}{Y(x)} \\
\underset{X(t)\neq 0}{\Leftrightarrow} \quad & \tfrac{X'(t)}{X(t)} + \omega && = D\frac{Y''(x)}{Y(x)}\,.
\end{aligned}$$

Da die linke Seite von x unabhängig und die rechte Seite von t unabhängig ist, gibt es eine Konstante c mit $\frac{X'(t)}{X(t)} + \omega = c = D\frac{Y''(x)}{Y(x)}$. Dies ergibt folgende gewöhnliche Differenzialgleichungen

$$\begin{aligned}
X'(t) &= X(t)(c-\omega) \\
Y''(x) &= \frac{c}{D}Y(x)
\end{aligned}$$

Die Lösungen der ersten Gleichung sind $A\exp\left((c-\omega)t\right)$ und die der zweiten Gleichung sind $B\sin\left(\sqrt{\tfrac{c}{-D}}x\right) + C\cos\left(\sqrt{\tfrac{c}{-D}}x\right)$ für $c < 0$ und $D > 0$.

Likelihood

Wie in (3.1) beschrieben, gibt es auch hier wieder mehrere Varianten die Likelihoodfunktion aufzustellen. Da es wieder zu Singularitäten kommen kann, entscheiden wir uns für die Likelihood, die dieses Problem vermeidet :

$$l(D,\omega,\gamma) = (Dm_x - \omega\mu - m_t)^T \left(K_t + \gamma^2 I\right)^{-1} (Dm_x - \omega\mu - m_t)\,, \quad (3.10)$$

wobei m_x die Schätzung der zweiten Ableitung von $C(t,x)$ nach x und m_t die Schätzung der ersten Ableitung nach t ist. Diese Likelihood lässt sich weder mit

dem Gradientenabstieg noch mit dem Simulated Annealing Verfahren so minimieren, dass die Schätzparameter in einer Umgebung der echten Parameter liegen.

D	$\hat{D}$	$\mathcal{S}(\hat{D})$	ω	$\hat{\omega}$	$\mathcal{S}(\hat{\omega})$
0.4	0.0266	0.4155	0.5	-0.4423	0.4151
0.2	0.0530	0.2205	2	0.9723	0.6547
0.2	0.7060	2.8432	0	0.5802	0.2674

Abbildung 3.5: Reaction-Diffusion Rate Equation - Simulated Annealing

Abbildung (3.5) zeigt Ergebnisse des ersten Versuchs. Dabei wurden wieder 100 Durchläufe pro Parametersatz berechnet. Die Daten wurden mit $c = -1$ erzeugt und die Standardabweichung des Beobachtungsrauschens wurde auf 0.01 gesetzt. Es ist deutlich zu sehen, dass die Schätzungen der Parameter stark fehlerbehaftet sind. Auch eine Erhöhung der Datenmenge hat keine signifikante Verbesserung zur Folge.

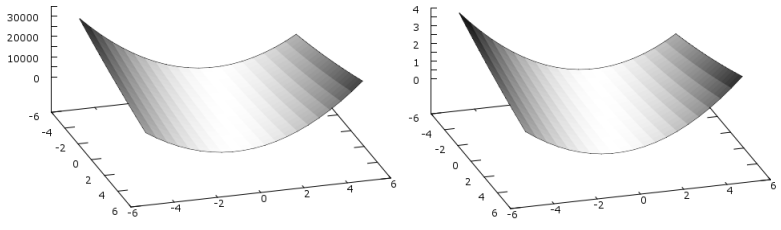

Abbildung 3.6: Likelihoodoberfläche

Abbildung (3.6) zeigt die Oberflächen der Likelihoodfunktion mit festem γ als Funktion von ω und D. Das linke Bild wurde mit $\gamma = 1$ berechnet, während das rechte Bild mit $\gamma = 100$ bestimmt wurde. Die Oberfläche selbst ist geometrisch gesehen nicht kompliziert, allerdings ist die Menge der Minima entlang der weiß gefärbten Strecke sehr ebenmäßig. Insbesondere sind die Minima sehr schwer voneinander zu unterscheiden, was auch die schwachen Ergebnisse des ersten Testdurchlaufs erklärt. Die Daten für die Berechnung der Likelihood wurden mit $D = 0.2, A = B = 1, C = 0$ und $\omega = 0$ und einem Beobachtungsrau-

schen mit Standardabweichung 0.01 erzeugt.

Modifizierte Likelihood

Wir nehmen nun an, dass wir einen der Modellparameter kennen. Wir betrachten also die beiden Likelihoodfunktionen

$$l_1(D,\gamma) = l(D,\omega_0,\gamma) \tag{3.11}$$

$$l_2(\omega,\gamma) = l(D_0,\omega,\gamma) \tag{3.12}$$

Die Daten werden äquidistant in $[-1,1]\times[-1,1]$ berechnet. Abbildung (3.7)

D	$\hat{D}$	$\mathcal{S}(\hat{D})$	ω	Rauschen	Daten
0.2	0.1915	0.002	1	0.01	81
0.2	0.1702	0.0037	1	0.1	81
0.2	0.0127	0.0177	1	1	81
0.2	0.1679	0.0598	1	0.01	49
0.5	0.4698	0.0570	0	0.01	81
ω	$\hat{\omega}$	$\mathcal{S}(\hat{\omega})$	D	Rauschen	Daten
1	0.9591	0.0056	0.2	0.01	81
1	0.787	0.2903	0.2	0.1	81
1	0.1747	0.4268	0.2	1	81
1	0.9437	0.013	0.2	0.01	49
0	−0.0058	0.0156	0.5	0.01	81

Abbildung 3.7: Versuche modifizierte Likelihood

zeigt Versuche mit l_1 (oben) und l_2 (unten). Ist ein Parameter bekannt und das Beobachtungsrauschen nicht zu groß, so können wir den zweiten Parameter relativ sicher schätzen. Daher wird nun versucht, durch verschiedene Priorverteilungen der Parameter D und ω eine Verbesserung für die Likelihood $l(D,\omega,\gamma)$ zu erreichen. Zunächst wird ein normalverteilter Prior über die Parameter betrachtet. Mit $\pi(D)=\mathcal{N}(p_1,s_1)$ und $\pi(\omega)=\mathcal{N}(p_2,s_2)$ ändert sich die Funktion (3.10) zu

$$\begin{aligned} l(D,\omega,\gamma) &= (Dm_x-\omega\mu-m_t)^T\left(K_t+\gamma^2 I\right)^{-1}(Dm_x-\omega\mu-m_t) \\ &+ \frac{(D-p_1)^2}{s_1^2}+\frac{(\omega-p_2)^2}{s_2^2}\,. \end{aligned} \tag{3.13}$$

Dabei ist zu beachten, dass der negative Logarithmus der Priorverteilung zu der negativen Log-Likelihood addiert wurde. Die Normalisierungsfaktoren der Priorverteilungen beeinflussen die Optimierung nicht, da sie konstant sind und in der negativen Log-Likelihood nur additiv auftreten. Sie werden daher ignoriert. Sollte man diese Parameter jedoch auch anpassen wollen, so muss man die Faktoren wieder hinzufügen. Die Wahl der Parameter der Priorverteilungen ist kritisch. Wählt man die Varianzen zu klein, so wird der geschätzte Parameter nahe an dem zugehörigen Mittelwert liegen. Damit werden die Modellparameter prinzipiell vor der Optimierung bereits festgelegt. Abbildung (3.8) zeigt einige Versuche mit nor-

D	$\hat{D}$	$\mathcal{S}(\hat{D})$	ω	$\hat{\omega}$	$\mathcal{S}(\hat{\omega})$	p_1	s_1	p_2	s_2	Rauschen
0.2	0.18	0.05	1	0.99	0.26	0	100	0.5	100	0.01
0.2	0.28	0.07	1	0.49	0.36	0	100	-0.5	100	0.01
0.2	0.22	0.08	1	0.72	0.33	0.5	100	0	100	0.01
0.2	0.24	0.06	1	0.70	0.33	0.2	100	0	100	0.01
0.2	0.25	0.02	2	1.17	0.46	0.5	1	0	9	0.01

Abbildung 3.8: Versuch mit normalverteiltem Prior

malverteiltem Prior. Für einige Parametersätze der Priorverteilungen konnten gute Ergebnisse erzielt werden, allerdings müssen die Mittelwerte verhältnismäßig nahe an den echten Parametern liegen. Insgesamt eignet sich der normalverteilte Prior nur, wenn die Lage der echten Parameter bereits relativ genau bekannt ist.
Als Priorverteilung wählen wir jetzt die Gammaverteilung

$$P_{b,p}(x) = \frac{b^p}{\Gamma(p)} x^{p-1} \exp(-bx)$$

für $x > 0$. Dabei ist zu beachten, dass nur noch nicht negative Modellparameter betrachtet werden, da die Dichte der Gammaverteilung für negative Zahlen 0 ist. Funktion (3.10) ändert sich damit zu

$$\begin{aligned} l(D,\omega,\gamma) &= (Dm_x - \omega\mu - m_t)^T \left(K_t + \gamma^2 I\right)^{-1} (Dm_x - \omega\mu - m_t) \\ &+ b_1 D - (p_1 - 1)log(D) + b_2\omega - (p_2 - 1)log(\omega)\ . \end{aligned} \tag{3.14}$$

Auch hier wurden wieder die negativen Logarithmen der Priorverteilungen zu Funktion (3.10) addiert. Auch hier sind die Modi der Priorverteilungen nahe

D	$\hat{D}$	$\mathcal{S}(\hat{D})$	ω	$\hat{\omega}$	$\mathcal{S}(\hat{\omega})$	p_1	b_1	p_2	b_2	Rauschen
0.2	0.2382	0.0026	1	0.9805	0.0061	1.5	2	2	1	0.01
1	0.9438	0.2777	1	0.8621	0.0928	8	8	8	8	0.01
1	0.8762	0.2183	1	0.9748	0.0796	2	1	2	1	0.01
0.5	0.7697	0.0034	1.5	1.7375	0.0031	2	1	3	1	0.01

Abbildung 3.9: Versuch mit gammaverteiltem Prior

den echten Parametern. Bei dem Versuch in der letzten Zeile wurden die Modi der Priorverteilungen bezüglich der echten Parameter verschoben und man sieht deutlich, dass die Abweichungen von den echten Parametern größer sind als in den anderen Versuchen. Insgesamt ist also auch diese Priorverteilung nur bedingt sinnvoll.

Kapitel 4

Integrierte Ableitungsbedingungen

4.1 Definition als Integralgleichung

Das Ziel dieses Abschnitts ist es, aus der 3-Schritt-Prozedur eine 1-Schritt-Prozedur zu bestimmen. Als Grundlage dienen dazu die Überlegungen aus der wissenschaftlichen Arbeit *Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes*. Es wird versucht, die Ableitungsbedingungen in den Gaußprozess zu integrieren und so die Modellparameter zusammen in einem Schritt mit den Hyperparametern zu schätzen.
Betrachtet man die Verteilung (1.24) (bis auf die Normierungskoeffizienten), ist diese gleich der Marginalverteilung[1]

$$p\left(\Theta,\gamma\right)=\pi\left(\Theta\right)\pi\left(\gamma\right)\prod_{n}\int\mathcal{N}\left(m_{n},K_{n}\right)\mathcal{N}(g_{n}(f,X,\Theta),\gamma_{n}^{2}I)dD^{k_{n}}X\ . \quad (4.1)$$

Das Produkt im Integranden ist ein Produkt von Verteilungen von $D^{k_n}X$. Dieses Produkt wird auch *Product of Experts* genannt. Wir versuchen nun, dieses *Product of Experts* auf den gesamten Gaußprozess zu übertragen. Wir betrachten also die

[1] Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes, S. 3

Verteilung

$$p\left(\Theta|Y^{1:T}\right) = \prod_n \int p(Y_n^{1:T}|X_n^{1:T}) \prod_m \int p(X_m^{1:T}, D^{k_m} X_m^{1:T}) \\ * \quad \beta \exp\left(-\sum_t \frac{1}{2} \left\|D^k X^{(t)} - g_t\right\|_\Gamma\right) dD^{k_m} X_m^{1:T} dX_n^{1:T} \quad (4.2)$$

mit

$$\Gamma = \begin{pmatrix} \gamma_1^2 & 0 & ... & ... & 0 \\ 0 & \gamma_2^2 & 0 & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & ... & ... & 0 & \gamma_N^2 \end{pmatrix} \quad (4.3)$$

und $\beta = \frac{1}{(2\pi)^{\frac{NT}{2}} (\det(\Gamma^{-1}))^{\frac{T}{2}}}$, wobei $g_t = g(f, X^{(t)}, \Theta)$ ist. $\|x\|_A$ ist eine verkürzte Schreibweise für $x^T Ax$. Die Produkte ergeben sich durch die Tatsache, dass wir unabhängige Gaußpriors als Generalvoraussetzung definiert haben. Die Lösung des Integrals für lineare Differenzialgleichungen findet sich im Anhang B. Das Ergebnis ist

$$P(\Theta|Y^{1:T}) = \frac{1}{2}\left(\mu^T C(C+\tilde{\Sigma})^{-1}\tilde{\Sigma}\mu - \ln(\det(\tilde{\Sigma})) + \ln(\det(C+\tilde{\Sigma}))\right) \\ + \quad \ln(|\det(\tilde{A})|) + \frac{1}{2}\sum_{n=1}^{N} \ln(\det(\tilde{C}_n)) \; . \quad (4.4)$$

Die entsprechenden Definitionen der Matrizen und Ausdrücke finden sich ebenfalls im Anhang. Es wurde hier explizit auf die Priorverteilungen für die Parameter und Hyperparameter verzichtet. Diese können aber ohne Weiteres wieder mit in die Gleichungen integriert werden.

4.2 Lineare Differenzialgleichungen

4.2.1 Anwendung auf eine einfache Differenzialgleichung

Wir betrachten zunächst die sehr einfache lineare Differenzialgleichung

$$y' = \lambda y \tag{4.5}$$

Das erste Ergebnis der Testläufe war, dass die Funktion (4.4) nicht ausreicht, um sämtliche Parameter innerhalb eines Optimierungsschrittes zu bestimmen. Eine Modifikation dieser Funktion jedoch ist dazu in der Lage. Zunächst betrachten wir aber die Funktion (4.4), wenn die Hyperparameter bekannt sind. Wir schätzen also zunächst die Parameter des Gaußprozesses, um anschließend die Modellparameter zu bestimmen. Da die Matrizen $\tilde{C}_n$ nur von den Hyperparametern abhängen, ändert sich Ausdruck (4.4) für die Optimierung dann zu

$$\begin{aligned} P(\Theta|Y^{1:T}) = \; & \tfrac{1}{2}\left(\mu^T C(C+\tilde{\Sigma})^{-1}\tilde{\Sigma}\mu - \ln(\det(\tilde{\Sigma})) + \ln(\det(C+\tilde{\Sigma}))\right) \\ + \; & \ln(|\det(\tilde{A})|)\ . \end{aligned} \tag{4.6}$$

Geschätzt werden die Parameter mittels des Simulated Annealing Verfahrens. Allerdings werden die Parameter λ und γ jeweils von Normalverteilungen mit unterschiedlicher Standardabweichung gezogen. Für λ wird eine Standardabweichung von 0.01 angesetzt und für γ eine Standardabweichung von 1.0. Dies soll vor allem für γ eine stärkere Traversierung des Zustandsraumes ermöglichen. Eine hohe Präzision γ hat sich bisher als sinnvoll herausgestellt, daher ermöglicht eine höhere Standardabweichung für die entsprechende Normalverteilung ein schnelleres erreichen von hohen Präzisionswerten, unabhängig vom Startwert. Abbildung (4.1) zeigt erste Versuche mit Differenzialgleichung (4.5).Es werden wieder Mittelwerte und Standardabweichung über 100 Testläufe gezeigt. Interessant ist, dass der Term $\ln(\det(C+\tilde{\Sigma}))$ in Ausdruck (4.6) keinen, oder nur sehr geringen Einfluss

λ	$\hat{\lambda}$	$S(\hat{\lambda})$	Rauschen
-0.5	-0.4997	0.0025	0.01
-0.5	-0.2854	0.2311	0.1
1.0	0.9928	0.0070	0.1
1.0	0.6734	0.1490	1.0

Abbildung 4.1: Versuchsauswertung - lineare Differenzialgleichung

λ	$\hat{\lambda}$	$S(\hat{\lambda})$	Rauschen
-0.5	-0.4991	0.0025	0.01
-0.5	-0.3568	0.3211	0.1
1.0	0.9847	0.0140	0.1
1.0	0.7543	0.0676	1.0

Abbildung 4.2: Versuchsauswertung - lineare Differenzialgleichung (2)

auf die Schätzung der Parameter hat. Abbildung (4.2) zeigt dieselben Versuche wie Abbildung (4.1), nur ohne den beschriebenen Term. Wie man deutlich sieht, sind die Unterschiede beider Versuche nur marginal.

4.2.2 Ein-Schritt-Prozedur

Die Funktion (4.2) wird in diesem Abschnitt modifiziert, um eine Schätzung aller Parameter innerhalb eines Optimierungsablaufes zu ermöglichen. Die Idee ist es, diejenige Verteilung in den Ausdruck (4.2) mit einzufügen, die verwendet wird, um den Gaußprozess an die Daten anzupassen. Diese ist in (1.6) beschrieben. Es ergibt sich

$$\tilde{P}(\Theta|Y^{1:T}) \;=\; P(\Theta|Y^{1:T}) \prod_{n=1}^{N} \mathcal{N}(0, C_n + \sigma_n^2 I_T)\,. \tag{4.7}$$

Die Funktion (4.4) ändert sich dann entsprechend zu

$$\begin{aligned}\tilde{P}(\Theta|Y^{1:T}) = \;& \tfrac{1}{2}\left(\mu^T C(C+\tilde{\Sigma})^{-1}\tilde{\Sigma}\mu - \ln(\det(\tilde{\Sigma})) + \ln(\det(C+\tilde{\Sigma}))\right) \\ +\; & \ln(|\det(\tilde{A})|) + \tfrac{1}{2}\textstyle\sum_{n=1}^{N} \ln(\det(\tilde{C}_n)) + \ln(\det(C_n+\sigma_n I)) \\ +\; & \tfrac{1}{2}\sum_{n=1}^{N}\left(Y_n^{1:T}\right)^T (C_n+\sigma_n^2 I)^{-1} Y_n^{1:T}\,. \end{aligned} \tag{4.8}$$

Abbildung (4.3) zeigt Histogramme für den geschätzten Modellparameter $\lambda = 1$

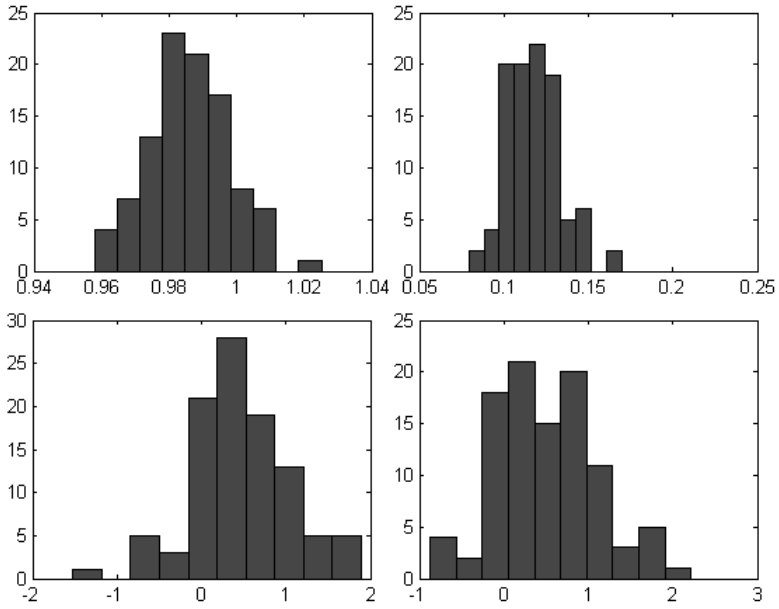

Abbildung 4.3: Histogramm - lineare Differenzialgleichung

(links) und das Beobachtungsrauschen $\sigma = 0.1$ (rechts). Der Unterschied zwischen den oberen und den unteren Histogrammen ist auf die Summanden $\ln \det(\tilde{C})$ zurückzuführen. Bei den Versuchen, die die oberen Histogramme erzeugt haben, wurden diese Terme nicht berechnet, während sie für die unteren Histogramme beachtet wurden. Es ist deutlich zu sehen, dass ohne die entsprechenden Terme die Schätzung der Parameter weitaus stabiler ist.

Die Matrizen $\tilde{C}$ können während der Optimierung numerisch singulär werden oder sogar eine negative Determinante besitzen, die nahe bei 0 liegt. Fügt man einen Störterm $\tilde{C} + sI_T$ additiv hinzu, können die Ergebnisse leicht verbessert werden.

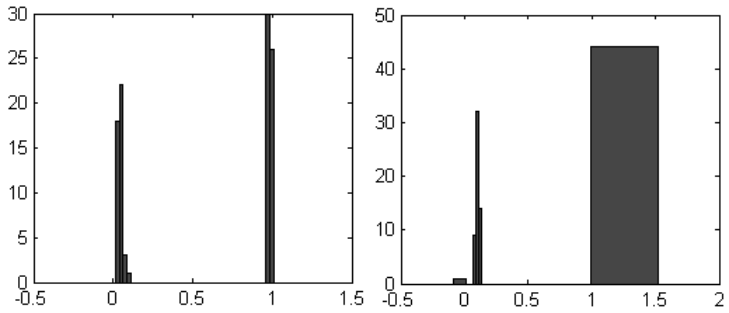

Abbildung 4.4: Histogramm - lineare Differenzialgleichung (2)

Abbildung (4.4) zeigt denselben Versuch mit $s = 1e - 4$. Zwar wird der echte Parameter öfter getroffen, aber es traten auch viele Fehlschätzungen auf. Für das Modell der gedämpften Schwingung (1.5) werden ebenfalls noch einige Versuche durchgeführt.

Parameter	Mittelwert	Stichprobenstandardabweichung
$\omega = 1.0$	0.9734	0.0143
$\lambda = 0.2$	0.1786	0.0194
$\sigma_1 = 0.1$	0.1276	0.0142
$\sigma_2 = 0.1$	0.1333	0.0198
$\omega = 0.5$	0.4874	0.0065
$\lambda = 0.1$	0.0946	0.0125
$\sigma_1 = 0.1$	0.1381	0.0255
$\sigma_2 = 0.1$	0.1227	0.0175
$\omega = 1$	0.2469	0.1078
$\lambda = 0.2$	0.1743	0.2897
$\sigma_1 = 0.5$	0.6565	0.1225
$\sigma_2 = 0.5$	0.7073	0.0758

Abbildung 4.5: Versuchsauswertung - gedämpfte Schwingung

Abbildung (4.5) zeigt diese Versuche für drei Parametersätze. Sowohl das Rauschen als auch die Modellparameter wurden gut appproximiert. Optimiert wurde wieder mittels Simulated Annealing.

Aufwand

Die dominierenden Terme bezüglich der Zeitkomplexität der Funktion (4.8) sind die Inversen- und Determinantenberechnungen. Die größte Matrix ist die Inverse von $A = C + \tilde{\Sigma}$. Sei T die Anzahl der Beobachtungspunkte und N die Anzahl der Komponenten des Gaußprozesses, dann ist $A \in \mathbb{R}^{NT \times NT}$. Die Zeitkomplexität zur Invertierung der Matrix A ist daher $O((NT)^3)$. Die Auswertung der Determinante von A erfordert höchstens denselben Aufwand. Da sämtliche anderen Matrizen in Ausdruck (4.8) höchstens die gleiche Dimension wie A besitzen, ist $O((NT)^3)$ der dominante Term. Bei k Optimierungsschritten ist der Zeitaufwand also asymptotisch $O(k(NT)^3)$. Die in Abschnitt (1.4) vorgestellte Methode hat zum Vergleich eine Zeitkomplexität von[2] $O(kNT^3)$. Die hier hergeleitete Methode ist also für $N \geq 2$ bezüglich der Zeitkomplexität schwächer.

4.3 Nichtlineare Differenzialgleichungen

4.3.1 Beschreibung und Sampling

Für nichtlineare Differenzialgleichungen ist das Integral (4.1) nicht exakt zu lösen. Wenn man die Hyperparameter des Gaußprozesses vorher schätzt, lässt sich dieser Ausdruck aber als Erwartungswert ausdrücken. Setzen wir

$$f(X) = \frac{1}{(2\pi)^{\frac{NT}{2}} \det\left(\Gamma^{-1}\right)^{\frac{T}{2}}} \exp\left(-\frac{1}{2}\sum_{t=1}^{T}\left\|\dot{X}^{(t)} - g(X^{(t)})\right\|_{\Gamma}\right) , \tag{4.9}$$

dann ist

$$(4.5) = \mathbb{E}(f(X)) , \tag{4.10}$$

[2] Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes, S. 4

wobei der Erwartungswert bezüglich der Dichte

$$\tilde{p} = p(Y|X^{1:T})p(X^{1:T})p(\dot{X}^{1:T}|\dot{X}) \tag{4.11}$$

betrachtet wird. Für das Sampling Schema zur Approximierung des Integrals spielen die zugehörigen Normierungsfaktoren keine Rolle.

Sampling Schema

Die Idee ist es jetzt, den Erwartungswert (4.10) gemäß[3]

$$\hat{\mathbb{E}} = \frac{1}{L}\sum_{l=1}^{L} f(X^{(l)}) \tag{4.12}$$

zu approximieren. Da die Dichte $\tilde{p}$ nicht von den Modellparametern abhängt, reicht es, einmal genügend unabhängige Samples von $\tilde{p}$ zu ziehen und dann den Erwartungswert bezüglich der Modellparameter zu optimieren. Das Sampling von der Verteilung $\tilde{p}$ ist sehr einfach, da es sich um eine Normalverteilung handelt.
Sei dazu X eine normalverteilte Zufallsvariable mit Mittelwert $\tilde{\mu}$ und Kovarianz C. Sei A eine Matrix, sodass ACA^T definiert und invertierbar ist, dann ist AX Normalverteilt mit Mittelwert $\tilde{\mu}$ und Kovarianz ACA^T.
Betrachten wir nun die Matrizen K und $\tilde{C}$ wie in (1.9) und (1.16). Diese Matrizen sind , da es sich um Kovarianzmatrizen von Normalverteilungen handelt, positiv definit. Daher können wir für beide Matrizen die Choleskyzerlegung berechnen. Es seien $K_c, \tilde{C}_c$ die unteren Dreiecksmatrizen der Choleskyzerlegungen und X eine standardnormalverteilte Zufallsvariable. Dann sind $Y = \tilde{C}_c X + \mu, Z = K_c X + m$ entsprechend $\mathcal{N}(m, K)$ und $\mathcal{N}(\mu, \tilde{C})$ verteilte Zufallsvariablen. Kann man also unabhängige standardnormalverteilte Zufallsvariablen erzeugen, kann man direkt Samples von $\tilde{p}$ ziehen.

[3]Pattern Recognition and Machine Learning, S. 524, 11.2

Die Verteilung $\mathcal{N}(m, K)$ wird gegeben X betrachtet. Man zieht also zuerst Samples von $\mathcal{N}(\mu, \tilde{C})$ und nutzt dieses Sample dann für den Mittelwert m. Zusammengefasst also

Algorithmus 2

Schätze $\alpha, \beta_i, \sigma_i$ *gemäß* $\prod_{n=1}^{N} \mathcal{N}(0, C_n + \sigma_n^2 I_T)$

Ziehe L Samples $X^{(l)}$ *von* $\mathcal{N}(m, K), \mathcal{N}(\mu, \tilde{C})$

Maximiere den Erwartungswert $\frac{1}{L} \sum_{l=1}^{L} f(X^{(l)})$ *bezüglich* Φ

End

4.3.2 Anwendung

Zunächst wenden wir obiges Verfahren auf eine sehr leichte nichtlineare Differenzialgleichung an. Diese ist gegeben durch

$$Y' = \frac{a}{Y} . \tag{4.13}$$

Die Lösungen dieser Gleichung sind Wurzelfunktionen. Abbildung (4.6) zeigt Hi-

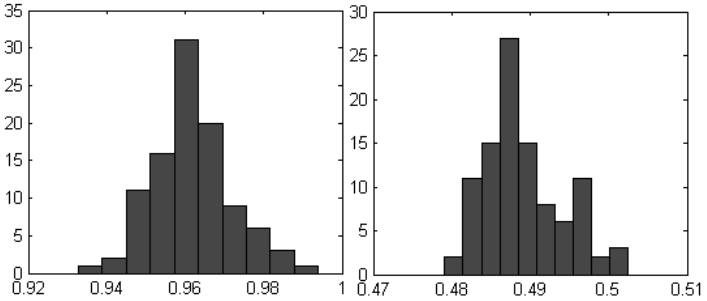

Abbildung 4.6: Versuche für eine nichtlineare Differenzialgleichung

stogramme von zwei Versuchen, deren Daten mit den Parametern $a = 1$ (links)

und $a = 0.5$ (rechts) erzeugt wurden. Zur Berechnung des Erwartungswertes wurden $L = 20$ Samples von $\tilde{p}$ gezogen. Die Standardabweichung des Beobachtungsrauschens betrug 0.01. Der Erwartungswert wurde mittels Simulated Annealing maximiert.

In allen vorherigen Versuchen haben hohe Werte für die Präzision die Stabilität der Inferenz positiv beeinflusst. Für eine hohe Präzision ist der Exponent von $f(x)$ allerdings sehr groß, es sei denn $\dot{X} - g(X)$ ist nahe bei Null. Das führt dazu, dass der Erwartungswert als Funktion der Modellparameter für eine hohe Präzision sehr steile Extrema annimmt. Für die Optimierung sind steile Extrema grundsätzlich nichts Negatives, jedoch kann es zu numerischen Schwierigkeiten kommen.

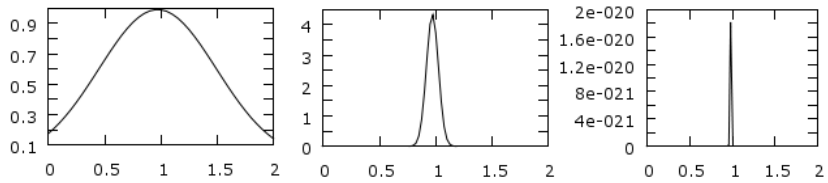

Abbildung 4.7: Erwartungswerte

Abbildung (4.7) zeigt die Erwartungswerte als Funktion von a für $\gamma = 1$ (links), $\gamma = 10$ (Mitte) und $\gamma = 100$ (rechts). Die Daten für die Erwartungswerte wurden mit $a = 1$ erzeugt. Alle 3 Funktionen nehmen in einer kleinen Umgebung um 1 ein Extremum an. Allerdings ist für das rechte Bild die Skala sehr klein. Rechnet man nicht in einer genügend hohen arithmetischen Präzision, ist es nahezu unmöglich, das Extremum zu finden, wenn γ sehr groß ist. Dieses Problem ist vorher nicht aufgetreten, da wir stets den Logarithmus der entsprechenden Likelihood betrachtet haben. Da wir hier aber eine Summe von Exponentialfunktionen haben, bringt die Logarithmierung des Erwartungswertes keine Vorteile.

Lotka Volterra Modell

Wir betrachten nun noch einmal das Lotka Volterra Modell aus Abschnitt 2.3. Die folgenden Tabellen zeigen Versuche mit diesem Modell.

Parameter	Mittelwert	Stichprobenstandardabweichung
$a = 1.0$	0.9959	0.0299
$b = 0.2$	0.1992	0.0037
$c = 0.5$	0.4985	0.0047
$d = 0.1$	0.0998	0.0012
$a = 0.8$	0.7944	0.2080
$b = 0.1$	0.0988	0.0100
$c = 0.6$	0.5922	0.0207
$d = 0.4$	0.3903	0.1908

Abbildung 4.8: Lotka Volterra - Anzahl der Samples : 1, Rauschen $0.1, 0.1$

Parameter	Mittelwert	Stichprobenstandardabweichung
$a = 1.0$	0.7537	0.2968
$b = 0.2$	0.1642	0.0272
$c = 0.5$	0.5116	0.0748
$d = 0.1$	0.0947	0.0106
$a = 0.8$	0.0832	0.5680
$b = 0.1$	0.0314	0.0398
$c = 0.6$	0.5263	0.0671
$d = 0.4$	0.3390	0.0143

Abbildung 4.9: Lotka Volterra - Anzahl der Samples : 1, Rauschen $1, 1$

Parameter	Mittelwert	Stichprobenstandardabweichung
$a = 0.8$	0.0132	0.6253
$b = 0.1$	0.0863	0.0361
$c = 0.6$	0.4926	0.1126
$d = 0.4$	0.3854	0.0374

Abbildung 4.10: Lotka Volterra - Anzahl der Samples : 100, Rauschen $1, 1$

Der Startwert der Präzision wurde auf 0.01 gesetzt. Interessant zu sehen ist vor allem, dass auch geringe Anzahlen von Samples für den Erwartungswert gute Resultate liefern können.

Wir führen nun noch einige Versuche durch, in denen wenig Samples für den Erwartungswert direkt mit mehreren Samples verglichen werden.

Parameter	Mittelwert (100 Samples)	Mittelwert (1 Sample)
$a = 1.0$	0.8390	0.7798
$b = 0.2$	0.1816	0.1731
$c = 0.5$	0.4317	0.4536
$d = 0.1$	0.0867	0.0909
Parameter	Abweichung (100 Samples)	Abweichung (1 Sample)
$a = 1.0$	0.2226	0.2651
$b = 0.2$	0.0282	0.0291
$c = 0.5$	0.0551	0.0894
$d = 0.1$	0.0141	0.0192
Parameter	Mittelwert (100 Samples)	Mittelwert (1 Sample)
$a = 0.8$	-0.035	-0.0423
$b = 0.1$	0.0577	0.0566
$c = 0.6$	0.5777	0.5744
$d = 0.4$	0.3941	0.387
Parameter	Abweichung (100 Samples)	Abweichung (1 Sample)
$a = 0.8$	0.3266	0.4660
$b = 0.1$	0.0171	0.0251
$c = 0.6$	0.0775	0.0833
$d = 0.4$	0.0237	0.0241

Abbildung 4.11: Vergleich - Anzahl der Samples

Abbildung (4.11) zeigt den direkten Vergleich für die Erwartungswertapproximation. In jedem der 100 Durchläufe wurde die Optimierung mit den exakt gleichen Daten und Startwerten durchgeführt. Das Beobachtungsrauschen hatte in den Versuchen eine Standardabweichung von 1. Die Unterschiede zwischen den Schätzungen sind sehr gering ausgefallen.

Nimmt man an, dass ein Sample ausreicht um den Erwartungswert zu approximieren können wir auch den Logarithmus von Ausdruck (4.12) betrachten, um die angesprochenen Probleme der steilen Extrema zu vermeiden. Damit ergibt sich

$$-\log \hat{\mathbb{E}} = \frac{1}{2}\sum_{t=1}^{T} \left\| \dot{X}^{(t)} - g(X^{(t)}) \right\|_{\Gamma} - \frac{T}{2}\log\left(\det\left(\Gamma\right)\right) \ . \tag{4.14}$$

Parameter	Mittelwert	Stichprobenstandardabweichung
$a = 1.0$	0.9966	0.0517
$b = 0.2$	0.1995	0.0055
$c = 0.5$	0.4985	0.0088
$d = 0.1$	0.0998	0.0017
$a = 0.8$	0.7977	0.1752
$b = 0.1$	0.0993	0.0083
$c = 0.6$	0.5953	0.0142
$d = 0.4$	0.3962	0.0077

Abbildung 4.12: Lotka Volterra - Rauschen $0.1, 0.1$

Parameter	Mittelwert	Stichprobenstandardabweichung
$a = 1.0$	0.7043	0.0952
$b = 0.2$	0.1684	0.0094
$c = 0.5$	0.5310	0.0530
$d = 0.1$	0.1019	0.0152
$a = 0.8$	0.3136	0.5961
$b = 0.1$	0.0712	0.0358
$c = 0.6$	0.5398	0.1502
$d = 0.4$	0.4041	0.0601

Abbildung 4.13: Lotka Volterra - Rauschen $1, 1$

Abbildungen (4.12) und (4.13) zeigen Versuche mit Funktion (4.14). Vergleicht man die Ergebnisse mit den vorherigen Versuchen, so sieht man, dass beide Methoden sehr ähnliche Ergebnisse erzielt haben. Der Vorteil hierbei ist jedoch, dass wir auch höhere Präzisionswerte betrachten können.

Betrachtet man Funktion (4.14) mit Präzisionsparameter 1 erhält man in einem gewissen Sinne den quadratischen Fehler als Optimierungsfunktion.

Aufwand

Der erste Schritt des Verfahrens besteht aus der Schätzung der Hyperparameter und hat eine Zeitkomplexität von $O(NT^3)$. Schritt zwei besteht aus dem Sampling der entsprechenden Verteilungen. Da wir nur einmal Samples der Verteilungen ziehen müssen, ist dieser Aufwand mit $O(1)$ abzuschätzen. Insbesondere müssen die Choleskyzerlegungen der Matrizen $\tilde{C}_n, K_n$ auch nur einmal bestimmt werden. Der dominante Term der Funktion (4.9) ist das Matrixprodukt im Exponenten. Die Matrix Γ ist eine Diagonalmatrix, daher haben Invertierung und Determinantenberechnung eine Zeitkomplexität von $O(N)$. Insbesondere lässt sich das Matrixprodukt im Exponenten dadurch optimieren. Es sei $x^t = \dot{X}^{(t)} - g(X^{(t)})$, dann ist

$$(x^t)^T \Gamma x^t \quad = \quad \sum_{i=1}^{n} \gamma_i^2 (x^t)_i^2 \ .$$

Der Aufwand für die Berechnung des gesamten Exponenten ist damit $O(TN^2)$. Für die Bestimmung des Erwartungswertes in Schritt drei ergibt sich damit eine Zeitkomplexität für k Optimierungsschritte von $O(kSTN^2)$. Sowohl Schritt eins als auch Schritt zwei haben dieselbe Zeitkomplexität wie die vorgestellte Methode aus Abschnitt (1.4). Schritt drei nach (1.4) hingegen hat eine Komplexität von $O(kNT^3)$. Ist $T >> N$ und die benötigte Samplemenge S klein , so kann auf diesem Weg eine Verbesserung der Zeitkomplexität erzielt werden. Insbesondere haben wir gesehen, dass für das Lotka Volterra-Modell ein Sample ausreicht, um die Parameter zu schätzen. Dies gilt zumindest für das betrachtete Beobachtungsrauschen.

Kapitel 5

Zusammenfassung und Ausblick

Im Rahmen dieser Arbeit wurden verschiedene Methoden und Verfahren für die Parameterschätzung in Differenzialgleichungssystemen betrachtet. Neben der Implementierung und Auswertung des bereits bekannten Verfahrens aus der Arbeit *Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes* wurde auf der Grundlage desselben versucht, Ableitungsbedingungen in einen Gaußprozess zu integrieren. Das Resultat dieser Betrachtungen waren eine Schätzfunktion für lineare Differenzialgleichungssysteme und ein Samplingverfahren für nichtlineare Systeme.

Gewöhnliche und partielle Differenzialgleichungen

Es wurde festgestellt, dass Inferenz in gewöhnlichen Differenzialgleichungssystemen gut mit den vorgestellten und hergeleiteten Methoden funktionierte. Partielle Differenzialgleichungssysteme sind jedoch nicht so leicht zu handhaben. Während es auch erfolgreiche Versuche wie mit System (3.1) gab, haben wir ein Beispiel einer partiellen Differenzialgleichung gegeben, bei der der Inferenzmechanismus nur bedingt zielführend war.
Auch der Versuch Priorverteilungen für die Modellparameter zu finden, war nur

mäßig erfolgreich, da die entsprechend gewählten Verteilungen ihren Modus stets in der Nähe der echten Modellparameter hatten und somit starkes A-Priori-Wissen verwendet wurde.
Eine weitere partielle Differenzialgleichung wurde während der Untersuchungen betrachtet, aber nicht explizit ausgewertet. Diese ist gegeben durch

$$\frac{\partial f(t,x)}{\partial t} = ax + bt \ .$$

Es sei hier erwähnt, dass auch Inferenz in dieser Gleichung gute Resultate geliefert hat. Grundsätzlich kann man aber davon ausgehen, dass die vorgestellten Methoden nicht ohne weitere Betrachtungen für Inferenz in jeder beliebigen partiellen Differenzialgleichung angewendet werden können. Die Autoren von *Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes* schlagen in diesem Zusammenhang komplexere Samplingverfahren vor, um von der Verteilung (1.24) Samples zu ziehen. Dabei geht es um mehrere, parallele Monte Carlo Sampler, die Informationen austauschen um den Zustandsraum besser zu durchlaufen.

Sampling

Wir haben gesehen, dass man das Inferenzproblem zu einer Erwartungswertbetrachtung umformulieren kann. Insbesondere konnten wir die Zeitkomplexität für die Schätzung der Modellparameter unter gewissen Umständen verbessern.
Das Sampling zur Berechnung des Erwartungswertes (4.12) wurde mit bekannten Hyperparametern durchgeführt. Interessant wäre es, wie im Falle der linearen Differenzialgleichungen, die Optimierung in einem Schritt durchzuführen. Wenn sich allerdings die Hyperparameter ändern, so ändert sich die Verteilung, bezüglich derer der Erwartungswert berechnet wird. Das bedeutet, dass für jede Änderung der Hyperparameter neue Samples von der entsprechenden Verteilung gezogen werden müssen, wenn man den Erwartungswert bestimmen möchte.

Anhang A

Ableitungen von Gaußprozessen

Definition

Die Ableitung von Gaußprozessen liefert wieder einen Gaußprozess, da der Ableitungsoperator linear ist. Zunächst betrachten wir Gaußprozesse, die lediglich von der Zeit abhängen. Es sei $X(t) \in \mathbb{R}$ und $t \in T$ ein Gaußprozess mit zweimal stetig differenzierbarer, symmetrischer Kernfunktion $K(\cdot, \cdot)$ und differenzierbarer Mittelwertsfunktion $\mu(t)$, dann ist der Zufallsprozess $Y(t)$ die Ableitung von $X(t)$, wenn für alle $t \in T$ der Grenzwert [1]

$$\lim_{\tau \to 0} \left\| \frac{X(t+\tau) - X(t)}{\tau} - Y(t) \right\| = 0 \tag{A.1}$$

existiert. Die Konvergenz wird hier im quadratischen Mittel betrachtet. Obiger Grenzwert lässt sich also auch als

$$\lim_{\tau \to 0} \mathbb{E} \left[\left(\frac{X(t+\tau) - X(t)}{\tau} - Y(t) \right)^2 \right] = 0 \tag{A.2}$$

formulieren. Insbesondere ist dann[2] $\frac{dX(t)}{dt} = Y(t)$.

[1] Stochastische Systeme, S. 106 (3.19)

[2] Stochastische Systeme, S. 107 (3.22)

Beweis

Zunächst ist, da $X(t)$ ein Gaußprozess ist, jede endliche Auswahl von Zufallsvariablen des Prozesses stets multivariat normalverteilt. Es gilt also

$$P(X(t+\tau), X(t)) \approx \mathcal{N}(\mu, C) \tag{A.3}$$

für alle τ mit $\mu = (\mu(t+\tau), \mu(t))^T$ und $C = \begin{pmatrix} K(t+\tau, t+\tau) & K(t+\tau, t) \\ K(t, t+\tau) & K(t,t) \end{pmatrix}$.
Ist zudem X eine normalverteilte Zufallsvariable mit Mittelwert μ und Kovarianz C und A eine Matrix, sodass ACA^T regulär ist, so ist AX ebenfalls normalverteilt mit Parametern $\mu^* = A\mu$ und $C^* = ACA^T$.
Wir definieren nun die Matrix $A = \frac{1}{\tau}\begin{pmatrix} 1 & -1 \end{pmatrix}$ und den Zufallsvektor $X_\tau(t) = (X(t+\tau), X(t))^T$, dann ist die Verteilung von $X_\tau(t)$ gleich (A.3) und $AX_\tau(t) = \frac{X(t+\tau)-X(t)}{\tau}$ ist normalverteilt mit Mittelwert und Varianz

$$\mu_\tau = \frac{\mu(t+\tau) - \mu(t)}{\tau} \tag{A.4}$$

$$\begin{aligned} C_\tau &= \frac{K(t+\tau, t+\tau) - K(t, t+\tau) + K(t,t) - K(t+\tau, t)}{\tau^2} \\ &= \frac{K(t+\tau, t+\tau) - K(t, t+\tau) - (K(t+\tau, t) - K(t,t))}{\tau^2} \\ &= \frac{\frac{K(t+\tau,t+\tau)-K(t,t+\tau)}{\tau} - \frac{K(t,t+\tau)-K(t,t)}{\tau}}{\tau}. \end{aligned} \tag{A.5}$$

Betrachtet man nun den Grenzwert $\tau \to 0$, so liegt die Vermutung nahe, dass die Verteilung der Grenzvariablen Y normalverteilt mit Mittelwert $\tilde{\mu} = \mu'(t)$ und Kovarianz $\tilde{C} = \frac{\partial^2 K(t_1,t_2)}{\partial t_1 \partial t_2}$ ist. Hierbei beschreiben die Indizes für t jeweils das erste beziehungsweise das zweite Argument an K. Sei a_n eine Nullfolge, dann bleibt zu zeigen, dass

$$\lim_{n\to\infty} \mathbb{E}\left[(X_{a_n}(t) - Y(t))^2\right] = 0 \tag{A.6}$$

gilt. Dazu betrachten wir zunächst die Kovarianzen von $X_{a_n}(t), X(s)$. Die Kova-

rianz ist eine symmetrische Bilinearform, daher gilt

$$\begin{aligned} C(X_{a_n}(t), X(s)) &= C\left(\frac{X(a_n+t)-X(t)}{a_n}, X(s)\right) \\ &= \frac{1}{a_n}\left(C(X(a_n+t), X(s)) - C(X(t), X(s))\right) \\ &= \frac{1}{a_n}\left(K(a_n+t, s) - K(t, s)\right) \\ &\to \frac{\partial K(t,s)}{\partial t}\,. \end{aligned} \tag{A.7}$$

Damit ist $C(Y(t), X(s)) = \frac{\partial K(t,s)}{\partial t}$. Analog zeigt man $C(X(t), Y(s)) = \frac{\partial K(t,s))}{\partial s}$ und $C(Y(t), Y(s)) = \frac{\partial^2 K(t,s)}{\partial t \partial s}$. Insbesondere ist

$$C(X_{a_n}(t), Y(t)) = \frac{\frac{\partial K(a_n+t,t)}{\partial t_2} - \frac{\partial K(t,t)}{\partial t_2}}{a_n}\,. \tag{A.8}$$

Da wir K als zweimal stetig differenzierbar angenommen haben, ist die Kernfunktion $\frac{\partial^2 K(t,s)}{\partial t \partial s}$ als direkte Folge aus dem Satz von Schwarz symmetrisch. Es ist zu beachten, dass im Allgemeinen $\frac{\partial K(t,s)}{\partial t} \neq \frac{\partial K(t,s)}{\partial s}$ gilt. Allerdings ist $\frac{\partial K(t,s)}{\partial t} = \frac{\partial K(s,t)}{\partial t}$, da K symmetrisch ist.

Wir betrachten nun das zweite quadratische Moment einer normalverteilten Zufallsvariablen X. Dann ist mittels partieller Integration und Substitution $y = x - \mu$

$$
\begin{aligned}
\mathbb{E}(X^2) &= \int_{\mathbb{R}} \frac{x^2}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right) dx \\
&= \int_{\mathbb{R}} \frac{(y+\mu)^2}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\frac{y^2}{\sigma^2}\right) dy \\
&= \int_{\mathbb{R}} \frac{y^2}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\frac{y^2}{\sigma^2}\right) dy + 2\mu \underbrace{\int_{\mathbb{R}} \frac{y}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\frac{y^2}{\sigma^2}\right) dy}_{=0} \\
&+ \mu^2 \underbrace{\int_{\mathbb{R}} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\frac{y^2}{\sigma^2}\right) dy}_{=1} \\
&= \int_{\mathbb{R}} \frac{y^2}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\frac{y^2}{\sigma^2}\right) dy + \mu^2 \\
&= \underbrace{\left[-\frac{y\sigma}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\frac{y^2}{\sigma^2}\right)\right]_{-\infty}^{\infty}}_{=0} + \sigma^2 \underbrace{\int_{\mathbb{R}} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\frac{y^2}{\sigma^2}\right) dy}_{=1} + \mu^2 \\
&= \mu^2 + \underbrace{\sigma^2}_{=\mathbb{V}(X)} . \qquad \text{(A.9)}
\end{aligned}
$$

Mithilfe dieser Gleichung können wir (A.6) umformulieren zu

$$
\begin{aligned}
\mathbb{E}\left[\left(X_{a_n}(t) - Y(t)\right)^2\right] &= \mathbb{E}\left[X_{a_n}^2(t)\right] - 2\mathbb{E}\left[X_{a_n}(t)Y(t)\right] + \mathbb{E}\left[Y^2(t)\right] \\
&= \mu_{a_n}^2 + C_{a_n} - 2\mathbb{E}\left[X_{a_n}(t)Y(t)\right] + \tilde{\mu}^2 + \tilde{C} \,. \qquad \text{(A.10)}
\end{aligned}
$$

Zu Beachten ist, dass $C_{a_n} = \mathbb{V}(X_{a_n}(t))$ gilt. Es bleibt die Auswertung von $\mathbb{E}\left[X_{a_n}(t)Y(t)\right]$. Allgemein gilt für die Kovarianz

$$
C(X_{a_n}(t), Y(t)) = \mathbb{E}(X_{a_n}(t)Y(t)) - \mathbb{E}(X_{a_n}(t))\mathbb{E}(Y(t)) \,.
$$

Damit ist

$$
\begin{aligned}
\mathbb{E}(X_{a_n}(t)Y(t)) &= C(X_{a_n}(t), Y(t)) + \mathbb{E}(X_{a_n}(t))\mathbb{E}(Y(t)) \\
&= \frac{\frac{\partial K(a_n+t,t)}{\partial t_2} - \frac{\partial K(t,t)}{\partial t_2}}{a_n} + \mu_{a_n}\tilde{\mu} \\
&\to \tilde{C} + \tilde{\mu}^2 \,. \qquad \text{(A.11)}
\end{aligned}
$$

Damit folgt insgesamt

$$\begin{aligned}\mathbb{E}\left[(X_{a_n}(t)-Y(t))^2\right] &= \mu_{a_n}^2 + C_{a_n} - 2\frac{\frac{\partial K(a_n+t,t)}{\partial t_2} - \frac{\partial K(t,t)}{\partial t_2}}{a_n} - 2\mu_{a_n}\tilde{\mu} + \tilde{\mu}^2 + \tilde{C} \\ &\to \quad \tilde{\mu}^2 + \tilde{C} - 2\tilde{C} - 2\tilde{\mu}^2 + \tilde{\mu}^2 + \tilde{C} \\ &= \quad 0\,. \end{aligned} \tag{A.12}$$

Also konvergiert die Folge $X_n(t)$ im quadratischen Mittel gegen $Y(t)$ und damit ist $Y(t)$ die Ableitung von $X(t)$.

Als Letztes zeigen wir, dass $Y(t)$ auch tatsächlich ein Gaußprozess ist. Sei also $Y(t_1), ..., Y(t_n)$ eine endliche Auswahl von $Y(t)$. Da $X_\tau(t) \to Y(t)$, gilt auch $(X_\tau(t_1), ..., X_\tau(t_n)) \to (Y(t_1), ..., Y(t_n))$ im quadratischen Mittel.

Betrachte die Matrix $A = \frac{1}{\tau}(I_n \otimes \begin{pmatrix} 1 & -1 \end{pmatrix})$ und den Zufallsvektor

$$X = (X(t_1+\tau), X(t_1), ..., X(t_n+\tau), X(t_n))^T\,.$$

Der Vektor X ist multivariat normalverteilt, da X ein Gaußprozess ist. Das heißt, es gibt einen Vektor μ und eine Kovarianzmatrix C mit $X \approx \mathcal{N}(\mu, C)$. Insbesondere ist $AX \approx \mathcal{N}(A\mu, ACA^T)$. Nach Konstruktion ist jedoch

$$AX = (X_\tau(t_1), ..., X_\tau(t_n))^T\,.$$

Damit ist $(Y(t_1), ..., Y(t_n))$ der Grenzwert im quadratischen Mittel einer multivariaten Normalverteilung und ist damit selbst multivariat normalverteilt und damit ein Gaußprozess. Für Gaußprozesse mit mehreren Abhängigen kann man die partiellen Ableitungen wie in (A.1) als Richtungsableitung definieren. Der Beweis kann dann analog zu dem gezeigten Fall geführt werden.

Anhang B

Lösung der Integralgleichung

Die Integralgleichung ist gegeben durch

$$p\left(\Theta|Y^{1:T}\right) = \prod_n \int p(Y_n^{1:T}|X_n^{1:T}) \prod_m \int p(X_m^{1:T}, D^{k_m}X_m^{1:T})\beta \\ * \quad \exp\left(-\sum_t \frac{1}{2}\left\|D^k X^{(t)} - g_t\right\|_\Gamma\right) dD^{k_m}X_m^{1:T} dX_n^{1:T} \,, \quad \text{(B.1)}$$

wobei β und Γ wie in (4.2) definiert sind. Dann ist mit $p\left(X_m^{1:T}, D^{k_m}X_m^{1:T}\right) = p\left(D^{k_m}X_m^{1:T}|X_m^{1:T}\right)p(X_m)$

$$p\left(\Theta|Y^{1:T}\right) = \beta \prod_n \int p\left(Y_n^{1:T}|X_n^{1:T}\right) p\left(X_n^{1:T}\right) \prod_m \int p\left(D^{k_m}X_m^{1:T}|X_m^{1:T}\right) \\ * \quad \exp\left(-\sum_t \tfrac{1}{2}\left\|D^{k_m}X^{(t)} - g_t\right\|_\Gamma\right) dD^{k_m}X_m^{1:T} dX_n^{1:T} \,. \quad \text{(B.2)}$$

Die Integrale lassen sich nur exakt lösen, wenn die Ableitungen linear eingehen. Daher betrachten wir zunächst nur lineare Differenzialgleichungssysteme $\dot{X}^{(t)} = AX^{(t)}$, also

$$p\left(\Theta|Y^{1:T}\right) = \beta \prod_n \int p\left(Y_n^{1:T}|X_n^{1:T}\right) p\left(X_n^{1:T}\right) \prod_m \int p\left(\dot{X}_m^{1:T}|X_m^{1:T}\right) \\ * \quad \exp\left(-\sum_t \frac{1}{2}\left\|\dot{X}^{(t)} - AX^{(t)}\right\|_\Gamma\right) d\dot{X}_m^{1:T} dX_n^{1:T} \,. \quad \text{(B.3)}$$

Zur Vereinfachung betrachten wir nur quadratische Matrizen. Für die Verteilungen gilt

$$\begin{aligned} & \prod_{n=1}^{N} p\left(Y_n^{1:T}|X_n^{1:T}\right) p\left(X_n^{1:T}\right) \\ \approx & \prod_{n=1}^{N} \mathcal{N}\left(\mu_n, \tilde{C}_n\right) \\ = & \frac{1}{(2\pi)^{\frac{N*T}{2}} \prod_{n=1}^{N} \det\left(\tilde{C}_n\right)^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(X-\mu)^T C (X-\mu)\right) \end{aligned} \tag{B.4}$$

mit

$$C = \begin{pmatrix} \tilde{C}_1^{-1} & 0 & ... & ... & 0 \\ 0 & \tilde{C}_2^{-1} & 0 & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & ... & ... & 0 & \tilde{C}_N^{-1} \end{pmatrix} \tag{B.5}$$

und $X = \left(\left(X_1^{1:T}\right)^T, \left(X_2^{1:T}\right)^T, ..., \left(X_N^{1:T}\right)^T\right)^T$, $\mu = \left(\mu_1^T, \mu_2^T, ..., \mu_N^T\right)^T$. Die Variablen μ_i und $\tilde{C}_i$ sind wie in (1.7) und (1.8) definiert. Für die bedingte Verteilung der Ableitungen gegeben X gilt

$$\begin{aligned} & \prod_{m=1}^{N} p\left(\dot{X}_m^{1:T}|X_m^{1:T}\right) \\ = & \prod_{m=1}^{N} \mathcal{N}\left(m_m, K_m\right) \\ = & \frac{1}{(2\pi)^{\frac{N*T}{2}} \prod_{m=1}^{N} \det\left(K_m\right)^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\left(\dot{X}-m\right)^T K \left(\dot{X}-m\right)\right) \end{aligned} \tag{B.6}$$

mit

$$K = \begin{pmatrix} \left(K_1\right)^{-1} & 0 & ... & ... & 0 \\ 0 & \left(K_2\right)^{-1} & 0 & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & ... & ... & 0 & \left(K_N\right)^{-1} \end{pmatrix}, \tag{B.7}$$

$\dot{X} = \left(\left(\dot{X}_1^{1:T}\right)^T, \left(\dot{X}_2^{1:T}\right)^T, ..., \left(\dot{X}_N^{1:T}\right)^T\right)^T$ und $m = \left(m_1^T, m_2^T, ..., m_N^T\right)^T$. Die Ausdrücke K_i, m_i sind wie in (1.15) und (1.16) deifniert. Weiterhin ist

$$\begin{aligned} & \exp\left(-\tfrac{1}{2}\sum_t \tfrac{1}{2}\left\|\dot{X}^{(t)} - AX^{(t)}\right\|_\Gamma\right) \\ = \quad & \exp\left(-\tfrac{1}{2}\left(\tilde{X} - (I_T \otimes A)\hat{X}\right)^T (I_T \otimes \Gamma)\left(\tilde{X} - (I_T \otimes A)\hat{X}\right)\right) . \end{aligned} \tag{B.8}$$

Mit $\tilde{X} = \left(\dot{X}^{(1)}, \dot{X}^{(2)}, ..., \dot{X}^{(T)}\right)^T$ und $\hat{X} = \left(X^{(1)}, X^{(2)}, ..., X^{(T)}\right)^T$. Der Ausdruck $\oplus$ beschreibt hier das Kroneckerprodukt. Wenn α_1, α_2 die Normalisierungskoeffizienten von (B.4) und (B.6) beschreiben, so ändert sich Ausdruck (B.3) zu

$$\begin{aligned} p\left(\Theta|Y^{1:T}\right) &= \beta\alpha_1\alpha_2 \int \exp\left(-\frac{1}{2}(X-\mu)^T C(X-\mu\right) \\ &\ast \int \exp\left(-\frac{1}{2}\left(\dot{X}-m\right)^T K\left(\dot{X}-m\right)\right) \\ &\ast \exp\left(-\frac{1}{2}\left\|\tilde{X} - (I_T \otimes A)\hat{X}\right\|_{(I_T \otimes \Gamma)}\right) d\dot{X} dX . \end{aligned} \tag{B.9}$$

Das Ziel ist es nun Matrizen zu finden, sodass $P\tilde{X} = \dot{X}$ und $Q\hat{X} = X$ ist. Die Idee dahinter ist es, das innere Integral auf die Form $\exp\left(-\frac{1}{2}X^T\hat{C}X\right)$ zu bringen, um dann die marginale Normalverteilung zu bestimmen.

Die Vektoren $X, \dot{X}$ beinhalten N Vektoren der Länge T und beschreiben die N Komponenten über alle Zeiten, während $\tilde{X}, \hat{X}$ T Vektoren der Länge N beschreiben. Zunächst betrachten wir einige Definitionen

$$x = \begin{pmatrix} X \\ \dot{X} \end{pmatrix} . \tag{B.10}$$

Weiterhin gilt nach (1.15) $m_n = M_n X_n^{1:T}$ mit $M_n = C_{f_n}\left(C_n + \sigma_n^2 I\right)^{-1}$. Wir definieren

$$M = \begin{pmatrix} -M_1 & 0 & ... & ... & 0 \\ 0 & -M_2 & 0 & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & ... & ... & 0 & -M_N \end{pmatrix} \tag{B.11}$$

und setzen $U = (M\ I_T \otimes I_n) \in \mathbb{R}^{NT \times 2NT}$. Damit haben wir alle Matrizen für den Ausdruck $\exp\left(-\frac{1}{2}\left(\dot{X}-m\right)^T K \left(\dot{X}-m\right)\right)$ beschrieben.
Es bezeichne $e_i \in \mathbb{R}^T$ den i-ten kanonischen Einheitszeilenvektor, dann definieren wir weiterhin

$$V = \begin{pmatrix} -A \otimes e_1 & I_N \oplus e_1 \\ -A \otimes e_2 & I_N \oplus e_2 \\ \dots & \dots \\ -A \otimes e_T & I_N \oplus e_T \end{pmatrix} \in \mathbb{R}^{2NT \times 2NT} \tag{B.12}$$

und setzen $\tilde{A} = \begin{pmatrix} U \\ V \end{pmatrix}$, dann sind

$$X - m = Ux \tag{B.13}$$

$$\tilde{X} - (I_T \otimes A)\hat{X} = Vx\,. \tag{B.14}$$

Insgesamt folgt

$$\begin{aligned} & \exp\left(-\frac{1}{2}\left(\dot{X}-m\right)^T K \left(\dot{X}-m\right)\right) \\ * \quad & \exp\left(-\frac{1}{2}\left\|\tilde{X} - (I_T \otimes A)\hat{X}\right\|_{(I_T \otimes \Gamma)}\right) \\ = \quad & \exp\left(-\frac{1}{2} x^T \tilde{A}^T \Sigma \tilde{A} x\right) \end{aligned} \tag{B.15}$$

mit

$$\Sigma = \begin{pmatrix} K & 0 \\ 0 & I_T \otimes \Gamma \end{pmatrix} \tag{B.16}$$

Die Matrix $\tilde{A}^T \Sigma \tilde{A}$ lässt sich als Blockmatrix schreiben mit vier $NT \times NT$ Matri-

zen. Wir setzen dazu

$$O_1 = \begin{pmatrix} -A \otimes e_1 \\ -A \otimes e_2 \\ ... \\ -A \otimes e_T \end{pmatrix} \tag{B.17}$$

$$O_2 = \begin{pmatrix} I_N \otimes e_1 \\ I_N \otimes e_2 \\ ... \\ I_N \otimes e_T \end{pmatrix} . \tag{B.18}$$

Dann ist

$$\tilde{A} = \begin{pmatrix} M & I_T \otimes I_N \\ O_1 & O_2 \end{pmatrix} \tag{B.19}$$

die beschriebene Blockschreibweise von $\tilde{A}$. Für das Produkt $\tilde{A^T}\Sigma\tilde{A}$ gilt dann

$$\begin{aligned} & \tilde{A}^T\Sigma\tilde{A} \\ = & \begin{pmatrix} M^T & O_1^T \\ I_T \otimes I_N & O_2^T \end{pmatrix} \begin{pmatrix} K & 0 \\ 0 & I_T \otimes \Gamma \end{pmatrix} \begin{pmatrix} M & I_T \otimes I_N \\ O_1 & O_2 \end{pmatrix} \\ = & \begin{pmatrix} M^TK & O_1^T(I_T \otimes \Gamma) \\ K & O_2(I_T \otimes \Gamma) \end{pmatrix} \begin{pmatrix} M & I_T \otimes I_N \\ O_1 & O_2 \end{pmatrix} \\ = & \begin{pmatrix} M^TKM + O_1^T(I_T \otimes \Gamma)O_1 & M^TK + O_1^T(I_T \otimes \Gamma)O_2 \\ KM + O_2^T(I_T \otimes \Gamma)O_1 & K + O_2^T(I_T \otimes \Gamma)O_2 \end{pmatrix} . \end{aligned} \tag{B.20}$$

Betrachtung der Matrizen

Sind die entsprechenden Matrixprodukte definiert, so gilt für das Kroneckerprodukt[1] $(A \otimes B)(C \otimes D) = AC \otimes BD$. Daher gilt $-A \otimes e_i = (I_N \otimes e_i)(-A \otimes I_T)$

[1] Topics in Matrix Analysis, Lemma 4.2.10

und damit folgt

$$O_1 = \begin{pmatrix} (I_N \otimes e_1)(-A \otimes I_T) \\ (I_N \otimes e_2)(-A \otimes I_T) \\ \dots \\ (I_N \otimes e_T)(-A \otimes I_T) \end{pmatrix} = O_2 * (-A \otimes I_T) \tag{B.21}$$

Es bezeichne $\tilde{e}_i$ den i-ten kanonischen Einheitszeilenvektor der Länge N, dann gilt sogar

$$\begin{aligned} O_2^T (I_N \otimes \Gamma) O_2 &= \begin{pmatrix} (I_N \otimes e_1^T) & \dots & (I_N \otimes e_T^T) \end{pmatrix} (I_N \otimes \Gamma) O_2 \\ &= \begin{pmatrix} (\gamma_1^2 I_T) \otimes \tilde{e}_1 \\ (\gamma_2^2 I_T) \otimes \tilde{e}_2 \\ \dots \\ (\gamma_N^2 I_T) \otimes \tilde{e}_N \end{pmatrix} \begin{pmatrix} (I_T \otimes \tilde{e}_1^T) & \dots & (I_T \otimes \tilde{e}_N^T) \end{pmatrix} \\ &= \begin{pmatrix} (\gamma_1^2 I_T) \otimes \tilde{e}_1 \tilde{e}_1^T & \dots & (\gamma_1^2 I_T) \otimes \tilde{e}_1 \tilde{e}_N^T \\ (\gamma_2^2 I_T) \otimes \tilde{e}_2 \tilde{e}_1^T & \dots & (\gamma_2^2 I_T) \otimes \tilde{e}_2 \tilde{e}_N^T \\ \dots & \dots & \dots \\ (\gamma_N^2 I_T) \otimes \tilde{e}_N \tilde{e}_1^T & \dots & (\gamma_N^2 I_T) \otimes \tilde{e}_N \tilde{e}_N^T \end{pmatrix} \\ &= \Gamma \otimes I_T \, . \end{aligned} \tag{B.22}$$

Dabei ist zu beachten, dass

$$O_2 = \begin{pmatrix} I_N \otimes e_1 \\ I_N \otimes e_2 \\ \dots \\ I_N \otimes e_T \end{pmatrix} = \begin{pmatrix} (I_T \otimes \tilde{e}_1^T) & \dots & (I_T \otimes \tilde{e}_N^T) \end{pmatrix} \text{ gilt.}$$

Mithilfe dieser Gleichungen folgt

$$\begin{aligned}
O_1^T(I_T \otimes \Gamma)O_1 &= (-A^T \otimes I_T)O_2^T(I_T \otimes \Gamma)O_2(-A \otimes I_T) \\
&= (-A^T \otimes I_T)(\Gamma \otimes I_T)(-A \otimes I_T) \\
&= (A^T\Gamma A \otimes I_T) \\
O_1^T(I_T \otimes \Gamma)O_2 &= (-A^T \otimes I_T)O_2^T(I_T \otimes \Gamma)O_2 \\
&= (-A^T \otimes I_T)(\Gamma \otimes I_T) \\
&= (-A^T\Gamma \otimes I_T) \\
O_2^T(I_T \otimes \Gamma)O_1 &= O_2^T(I_T \otimes \Gamma)O_2(-A \otimes I_T) \\
&= (\Gamma \otimes I_T)(-A \otimes I_T) \\
&= (-\Gamma A \otimes I_T)\ .
\end{aligned}$$

Betrachten wir nun wieder die Blockmatrix (B.20), dann ist

$$\begin{aligned}
\Lambda_{11} &= M^TKM + O_1^T(I_T \otimes \Gamma)O_1 \\
&= M^TKM + (A^T\Gamma A \otimes I_T) && \text{(B.23)} \\
\Lambda_{12} &= M^TK + O_1^T(I_T \otimes \Gamma)O_2 \\
&= M^TK + (-A^T\Gamma \otimes I_T) && \text{(B.24)} \\
\Lambda_{21} &= KM + O_2^T(I_T \otimes \Gamma)O_1 \\
&= (-\Gamma A \otimes I_T) && \text{(B.25)} \\
\Lambda_{22} &= K + O_2^T(I_T \otimes \Gamma)O_2 \\
&= K + (\Gamma \otimes I_T)\ . && \text{(B.26)}
\end{aligned}$$

Bestimmung der Integrale

Wir haben jetzt alle Matrizen definiert, um das innere Integral von Gleichung (B.9) zu lösen. Wir setzen noch $\alpha_3 = (2\pi)^{NT} \det\left(\left(\tilde{A}^T\Sigma A\right)^{-1}\right)$, dann lässt sich (B.9)

wie folgt beschreiben

$$\begin{aligned} & \beta\alpha_1\alpha_2\alpha_3 \int \exp\left(-\tfrac{1}{2}(X-\mu)^T C(X-\mu)\right) \int \\ * \quad & \tfrac{1}{\alpha_3} \exp\left(-\tfrac{1}{2} x^T \tilde{A}^T \Sigma \tilde{A} x\right) d\dot{X} dX \ . \end{aligned}$$

Das innere Integral beschreibt die Marginalverteilung

$$p\left(X^{1:T}\right) = \int p\left(X^{1:T}, \dot{X}^{1:T}\right) d\dot{X}^{1:T}$$

und ist gegeben durch[2]

$$\begin{aligned} & \beta\alpha_1\alpha_2\alpha_3 \int \exp\left(-\tfrac{1}{2}(X-\mu)^T C(X-\mu)\right) \alpha_4 \exp\left(-\tfrac{1}{2} X^T \tilde{\Sigma} X\right) dX \\ = \quad & \beta\alpha_1\alpha_2\alpha_3\alpha_4 \int \exp\left(-\tfrac{1}{2}\left((X-\mu)^T C(X-\mu) + X^T \tilde{\Sigma} X\right)\right) dX \end{aligned} \quad \text{(B.27)}$$

mit $\alpha_4 = \frac{1}{(2\pi)^{\frac{NT}{2}} \det\left(\tilde{\Sigma}^{-1}\right)^{\frac{1}{2}}}$ und $\tilde{\Sigma} = \Lambda_{11} - \Lambda_{12}\Lambda_{22}^{-1}\Lambda_{21}$.
Mit $\mu^* = (C + \tilde{\Sigma})^{-1} C\mu$ ist dann

$$\begin{aligned} & (X-\mu^*)^T (C+\tilde{\Sigma})(X-\mu^*) + \mu^T C (C+\tilde{\Sigma})^{-1} \tilde{\Sigma} \mu \\ = \quad & X^T (C+\tilde{\Sigma}) X - 2\mu^T C X + \mu^T \left(C(C+\tilde{\Sigma})^{-1} C + C(C+\tilde{\Sigma})^{-1}\tilde{\Sigma}\right) \mu \\ = \quad & X^T (C+\tilde{\Sigma}) X - 2\mu^T C X + \mu^T \left(C\left(C+\tilde{\Sigma}\right)^{-1}\left(C+\tilde{\Sigma}\right)\right) \mu \\ = \quad & X^T (C+\tilde{\Sigma}) X - 2\mu^T C X + \mu^T C \mu \\ = \quad & X^T C X - 2\mu^T C X + \mu^T C \mu + X^T \tilde{\Sigma} X \\ = \quad & (X-\mu)^T C (X-\mu) + X^T \tilde{\Sigma} X \ . \end{aligned}$$

Damit lässt sich (B.27) umformulieren zu

$$\begin{aligned} & \beta\alpha_1\alpha_2\alpha_3\alpha_4 \exp\left(-\frac{1}{2}\mu^T C (C+\tilde{\Sigma})^{-1}\tilde{\Sigma}\mu\right) \\ * \quad & \int \exp\left(-\frac{1}{2}(X-\mu^*)^T (C+\tilde{\Sigma})(X-\mu^*)\right) dX \end{aligned}$$

[2]Pattern Recognition and Machine Learning, S. 90 (2.98)

$$
\begin{aligned}
&= \beta\alpha_1\alpha_2\alpha_3\alpha_4 \exp\left(-\frac{1}{2}\mu^T C(C+\tilde{\Sigma})^{-1}\tilde{\Sigma}\mu\right) \\
&* \ (2\pi)^{\frac{TN}{2}} \det\left((C+\tilde{\Sigma})^{-1}\right)^{\frac{1}{2}} \\
&= (2\pi)^{\frac{TN}{2}} \det\left((C+\tilde{\Sigma})^{-1}\right)^{\frac{1}{2}} \exp\left(-\frac{1}{2}\mu^T C(C+\tilde{\Sigma})^{-1}\tilde{\Sigma}\mu\right) \\
&* \ \frac{(2\pi)^{\frac{3NT}{2}} \det\left(\tilde{\Sigma}\right)^{\frac{1}{2}} \det\left((\tilde{A}^T\Sigma\tilde{A})^{-1}\right)^{\frac{1}{2}}}{(2\pi)^{2NT} \det\left(\Gamma^{-1}\right)^{\frac{T}{2}} \prod_{n=1}^{N} \det\left(\tilde{C}_n\right)^{\frac{1}{2}} \det\left(K_n\right)^{\frac{1}{2}}}
\end{aligned}
$$

$$
\begin{aligned}
&= \exp\left(-\frac{1}{2}\mu^T C(C+\tilde{\Sigma})^{-1}\tilde{\Sigma}\mu\right) \\
&* \ \frac{(2\pi)^{-\frac{NT}{2}} \det\left(\Gamma\right)^{\frac{T}{2}} \det\left(\tilde{\Sigma}\right)^{\frac{1}{2}}}{\det\left((\tilde{A}^T\Sigma\tilde{A})\right)^{\frac{1}{2}} \det\left((C+\tilde{\Sigma})\right)^{\frac{1}{2}} \prod_{n=1}^{N} \det\left(\tilde{C}_n\right)^{\frac{1}{2}} \det\left(K_n\right)^{\frac{1}{2}}} \quad \text{(B.28)}
\end{aligned}
$$

Während der Umformungen wurde mehrfach die Gleichung

$$\det(A^{-1}) = \det(A)^{-1}$$

für invertierbares A verwendet. Weiterhin gilt mit dem Determinantenmultiplikationssatz

$$
\begin{aligned}
\det(\tilde{A}^T\Sigma\tilde{A})^{\frac{1}{2}} &= \left(\det(\tilde{A}^T)\det(\Sigma)\det(\tilde{A})\right)^{\frac{1}{2}} \\
&= \left|\det(\tilde{A})\right| \det(\Sigma)^{\frac{1}{2}} \\
&= \left|\det(\tilde{A})\right| \prod_{n=1}^{N} \det(K_n^{-1})^{\frac{1}{2}} \det(\Gamma)^{\frac{T}{2}} \ . \qquad \text{(B.29)}
\end{aligned}
$$

Es wurde dabei benutzt, dass eine Matrix und ihre transponierte Matrix stets die

gleiche Determinante besitzen. Die negative Log-Likelihood von (B.29) ist damit

$$\begin{aligned} & \tfrac{1}{2}\left(\mu^T C(C+\tilde{\Sigma})^{-1}\tilde{\Sigma}\mu - \ln(\det(\tilde{\Sigma})) + \ln(\det(C+\tilde{\Sigma}))\right) \\ + \quad & \ln(|\det(\tilde{A})|) + \tfrac{1}{2}\left(\sum_{n=1}^{N}\ln(\det(\tilde{C}_n)) + \underbrace{\ln(\det(K_n^{-1})) + (\ln\det(K_n))}_{=0}\right) \\ = \quad & \tfrac{1}{2}\left(\mu^T C(C+\tilde{\Sigma})^{-1}\tilde{\Sigma}\mu - \ln(\det(\tilde{\Sigma})) + \ln(\det(C+\tilde{\Sigma}))\right) \\ + \quad & \ln(|\det(\tilde{A})|) + \tfrac{1}{2}\sum_{n=1}^{N}\ln(\det(\tilde{C}_n))\,. \end{aligned} \tag{B.30}$$

Man beachte, dass sich die Terme $\det\Gamma^{\frac{T}{2}}$ ebenfalls durch die Umformung in (B.29) herauskürzen.

Abbildungsverzeichnis

Literaturverzeichnis

[1] Calderhead, Ben, Girolami, Mark, Lawrence, Neil D. : Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes

[2] Rasmussen C.E.,Williams C.K.I. : Gaussian Processes for Machine Learning, The MIT-Press 2006

[3] Bishop, Christopher M. : Pattern Recognition and Machine Learning, Springer 2006

[4] Petersen, Kaare B., Pedersen, Micheal S. : The Matrix Cookbook : http://www.matrixcookbook.com

[5] Rubinstein, Reuven Y., Kroese, Dirk P. : Simulation and the Monte Carlo Method, Wiley 2008

[6] Wu, Yu Feng, Myasnikova, Ekaterina, Reinitz, John : Master equation simulation analysis of immunostained Bicoid morphogen gradient : http://www.biomedcentral.com/1752-0509/1/52

[7] Wunsch, Gerhard, Schreiber, Helmut : Stochastische Systeme, Springer 2006

[8] Thompson, J.M.T., Stewart, H.B., Nonlinear dynamics and Chaos, Wiley 2002

[9] Lotka, J. Alfred, Elements of physical biology, Williams and Wilkins Company 1925

[10] Horn, Roger A.; Johnson, Charles R., Topics in Matrix Analysis, Cambridge University Press 1991