
Praktikum 7

Simon Stingelin

07.02.2023

Inhaltsverzeichnis

1	Nichtlineare Ausgleichsrechnung	1
1.1	Lernziele	1
1.2	Theorie	1
1.3	Aufträge	2
1.4	Industrielle Anwendung aus der Gas Analytik	3
1.5	Abgabe	5

1 Nichtlineare Ausgleichsrechnung

1.1 Lernziele

- Sie verstehen den Unterschied zwischen linearen und nichtlinearen Ausgleichsproblemen.
- Sie kennen numerische Verfahren zur Lösung eines nichtlinearen Ausgleichsproblems.
- Sie können das
 - Gauss-Newton Verfahren und
 - Levenberg-Marquardt Verfahren (optional)anwenden.

1.2 Theorie

Die Theorie zum Praktikum ist im Skript Kapitel 2.4 gegeben.

Das Praktikum ist wie folgt aufgebaut:

- Implementieren Gauss-Newton Verfahren
 - Testen am Modell Beispiel
Gleichung (2.43) im Skript mit den Daten aus der Tabelle 2.4 (vgl. Jupyter Notebook)
 - Anwenden auf Industrielle Anwendung aus der Gas Analytik (vgl. Praktikum 5)
- Erweitern des Gauss-Newton Verfahrens für eine gedämpfte Iteration

- Vergleichen Konvergenz Verhalten ungedämpft vs gedämpft
- Implementieren Levenberg-Marquardt Verfahren (optional)
 - Testen am Modell Beispiel
 - Anwenden auf Industrielle Anwendung aus der Gas Analytik

1.3 Aufträge

1. Implementieren Sie das Gauss-Newton Verfahren (Algorithmus 2.11)
2. Testen Sie Ihr Programm mit dem Modellproblem

$$u(t; u_0, \tau, \omega, \varphi) = u_0 \cdot e^{-\tau t} \cdot \sin(\omega t + \varphi) \quad (1)$$

Gleichung (2.41) aus dem Skript.

- Benutzen Sie die Daten aus dem Jupyter Notebook.
 - Generieren Sie selber ein Testdatenset und testen Sie Ihren Code.
3. Anwenden auf industrielle Anwendung aus der Gas Analytik

Wir greifen die Anwendung aus dem Praktikum 5 (lineare Ausgleichsrechnung) auf, wobei nun ebenso die Breite s_0 , wie auch die Position x_0 gesucht ist.

- Studieren Sie als Repetition die Einführung *Industrielle Anwendung aus der Gas Analytik*
- Skalieren Sie die x -Achse

Für die nichtlineare Ausgleichsrechnung müssen wir die x -Achse zwingend skalieren. Benutzen Sie dazu die Skalierung τ auf das Einheitsintervall $[0, 1]$

$$\begin{aligned} \tau : [x_{\min}, x_{\max}] &\rightarrow [0, 1] \\ x &\mapsto \tau(x) = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \end{aligned}$$

- Berechnen Sie mit Hilfe der linearen Ausgleichsrechnung gute Startwerte a_0

Die Daten P7 weichen leicht von denen im Praktikum 5 ab. Schätzen Sie die Lage x_0 und Breite s_0 ab und berechnen Sie mit Hilfe der linearen Ausgleichsrechnung Startwerte für das Parameterset. Achten Sie auf die Skalierung.

- Berechnen Sie das nichtlineare Ausgleichsproblem

Benutzen Sie dazu das implementierte Gauss-Newton Verfahren und die Startwerte vom Schritt oben. Kontrollieren Sie die Lösung, in dem Sie das Modell mit den Startwerten und dem optimierten Parametersatz darstellen.

- Variation der Startwerte

Berechnen Sie das nichtlineare Ausgleichsproblem mit den beiden folgenden Startwerte:

- $\tilde{a}_0 = 1.1 \cdot a_0$
- $\hat{a}_0 = 0.8 \cdot a_0$

wobei mit a_0 die Startwerte aus der linearen Ausgleichsrechnung bezeichnet seien. Sie werden feststellen, dass das Gauss-Newton Verfahren im ersten Fall gut konvergiert, im zweiten jedoch nicht. Das Problem kann etwas entschärft werden, in dem eine Dämpfung benutzt wird.

- Erweitern des Gauss-Newton Verfahrens mit einer Dämpfung

Erweitern Sie Ihr Gauss-Newton Verfahren gemäss Algorithmus 2.12 im Skript. Sie können Ihre Funktion um diese Funktionalität erweitern und optional zur Verfügung stellen. Eine mögliche Definition könnte wie folgt aussehen:

```
def GaussNewton(data, x0, y, dy, maxIter=100, tol=1e-12, damped=False,
    maxDampingIter=10):
```

- Testen Sie Ihre Erweiterung mit durch Variation der Startwerte

Sie werden feststellen, dass die Dämpfung eine kleine Verbesserung bringt, jedoch nicht wirklich befriedigend ist.

4. (optional) Implementieren Sie das Levenberg-Marquardt Verfahren analog zum Gauss-Newton Verfahren.

- Benutzen Sie als Test das Modellproblem aus dem Skript
- Wenden Sie das Verfahren auf die industrielle Anwendung.

Sie werden sehen, dass dieses Verfahren eine Verbesserung bezüglich Konvergenz bietet. Der Grund liegt in der *Regularisierung* und der *Trust-Region Strategie* für das Gewicht dieser. Das Verfahren findet daher in der Praxis sehr weitverbreitet Anwendung.

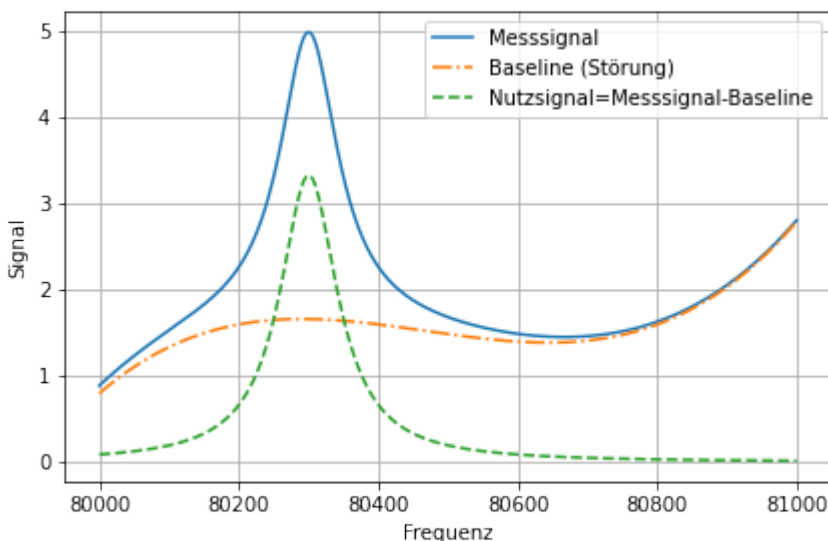
Nicht desto trotz sind gute Startwerte so gut wie immer entscheidend!

Wie sie sehen werden, ist auch hier die Konvergenz stark von den Startwerten abhängig.

1.4 Industrielle Anwendung aus der Gas Analytik

Anwendung aus der Messtechnik: Baseline Fit

In spektrometrischen Anwendungen der Messtechnik hat man oft das Problem, dass die sogenannte Baseline (Untergrund) der Messung unbekannt ist. In der Darstellung unten sehen Sie eine typische Situation:



Für die Interpretation des physikalischen Vorgangs ist man nur am Peak (grün) interessiert. Die Rohsignale entsprechen jedoch der blauen Kurve. Das Baselinesignal (orange) ist die Folge von Hintergrundprozessen oder Eigenschaften des Messgerätes und sollte diskriminiert werden. Die folgende Anleitung zeigt Ihnen, wie das mit Hilfe der Ausgleichsrechnung gelingt.

Beschreibung des Untergrundes (Baseline)

Wir gehen davon aus, dass sich der Untergrund durch ein unbekanntes **Polynom 3. Grades** beschreiben lässt, welches **additiv** zum eigentlichen **Peak der Resonanz** dazu kommt. Damit können wir die Messdaten durch eine *lineare* Kombination von Funktionsansätzen fitten.

Lorentz-Shape einer Resonanz

Die Systemreaktion wird mit Hilfe der Lorentz-Shape $l(x')$ beschrieben:

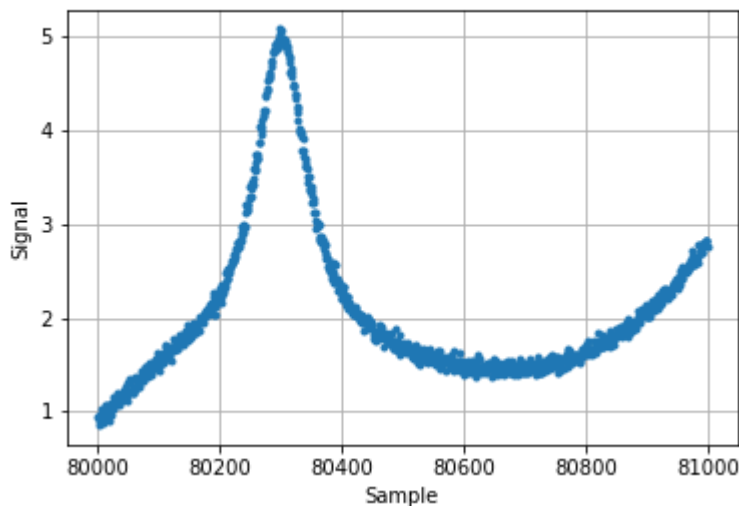
$$l(x') = \frac{1}{1 + (x')^2}, \quad x' = \frac{x - x_0}{s_0/2}.$$

wobei x_0 die Resonanzfrequenz und s_0 die Breite des Peaks bezeichnen. In vielen Anwendungen ist die Resonanzfrequenz, also x_0 zum vornherein bekannt, aber nicht der Wert von s_0 . Mit Hilfe der nichtlinearen Ausgleichsrechnung können wir nun sämtliche Parameter in den Datenfit miteinbeziehen.

Die Breite s_0 und die Position x_0 sei nun ebenfalls unbekannt.

Rauschenanteil

Im allgemeinen wird die Aufgabe erschwert durch Rauschen im Messsignal, d.h. realistische Daten sehen folgendermassen aus:



Sie finden diese Daten im Downloadbereich als “data.txt”. Eingeladen werden können diese Werte via:

```
import numpy as np
np.loadtxt('dataP7.txt')
```

Separation von Lorentz-Peak und Untergrund

- Implementieren Sie das nichtlineare Modell.
- Skalieren Sie die x -Achse.
- Berechnen Sie die Jakobi-Matrix bezüglich den Modellparameter für das Gauss-Newton Verfahren.
- Wenden Sie Ihre Gauss-Newton Methode zur Bestimmung der Modellparameter für die im Downloadbereich gegebenen Messdaten. Benutzen Sie geeignete Startparameter, welche Sie ggf. aus einem linearen Datenfit bestimmen.
- Stellen Sie den Lorentz-Peak als Funktion mit der gefitteten Amplitude graphisch dar, zusammen mit den verauschten Daten von welchen Sie den Untergrund *abziehen*.
- Vergleichen Sie das Konvergenzverhalten zwischen dem ungedämpften und gedämpften Gauss-Newton Verfahren.

1.5 Abgabe

Bitte geben Sie Ihre Lösungen bis spätestens vor dem nächsten Praktikum 8 ab.

Downloads:

- PDF-Dokumentation:
 - Anleitung Praktikum 7
 - dataP7.txt