PAP 2.2
Versuch 256: "Röntgenfluoreszens"

Jonas Schweer

15. Januar 2025

Inhaltsverzeichnis

1 Einleitung **2**
- 1.1 Ziel des Versuchs . . . 2
- 1.2 Versuchsaufbau . . . 2
- 1.3 Physikalische Grundlagen . . . 3
  - 1.3.1 Vielkanalsystem . . . 3
  - 1.3.2 Röntgenfluoreszenz . . . 3
  - 1.3.3 Benennung der charakteristischen Strahlung . . . 3
  - 1.3.4 Moseleysches Gesetz . . . 3
  - 1.3.5 Halbleiterdetektor/Röntgenenergiedetektor . . . 4

2 Messprotokoll **4**

3 Auswertung **11**
- 3.1 K_α-Linien Auswertung . . . 11
- 3.2 K_β-Linien Auswertung . . . 12
- 3.3 Überarbeitung der K_β-Linienmessung . . . 14

4 Zusammenfassung und Diskussion **15**

5 Anhang **16**

1 Einleitung

1.1 Ziel des Versuchs

In diesem Versuch wollen wir uns das Phänomen der Röntgenfluoreszenz zu nutzen machen um mit den K_α- und K_β-Linien verschiedener Elemente das Moseleysche Gesetz zu überprüfen. Wir wollen vor allem die Rydberg-Energie E_R bestimmen. Zuletzt versuchen wir noch qualitativ die Zusammensetzung einiger Legierungen zu ermitteln, indem wir ihr Spektrum analysieren.

1.2 Versuchsaufbau

Für die Messreihen verwenden wir ein Röntgengerät. Dieses besteht aus einer Röntgenröhre, welche ein kontinuierliches Röntgenspektrum erzeugt. Diese wird mit einem Kollimator auf eine Halterung fokussiert, auf welcher wir die Metalle und Legierungen platzieren können. Oberhalb der Halterung ist ein Energiedetektor angebracht, welcher mit Halbleitern und eines Vielkanalanalysators die einfallenden Energien auswertet. Diese Daten gibt er an einen beistehenden Computer weiter, an welchem wir sie mit einem speziellen Programm auswerten können.

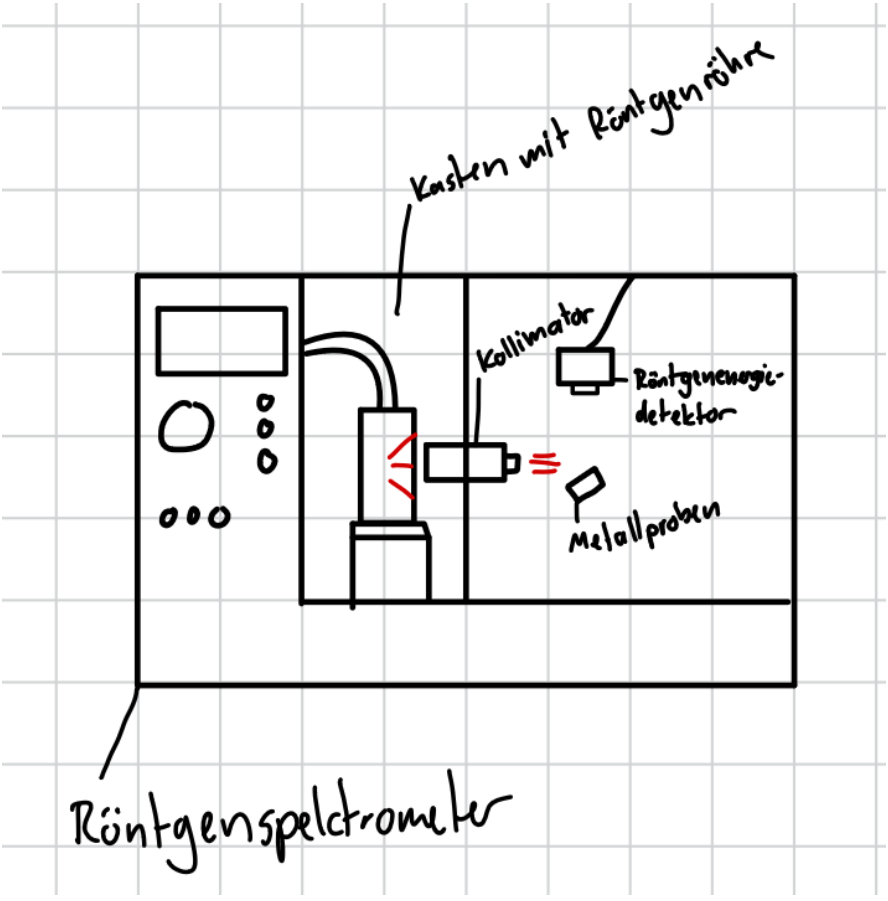


Abbildung 1: Messaufbau

1.3 Physikalische Grundlagen

1.3.1 Vielkanalsystem

Das Vielkanalsystem ist (wie der Name bereits verrät) aus vielen verschiedenen Kanälen aufgebaut. Jeder dieser Kanäle ist für eine unterschiedliche Impulshöhe verantwortlich und misst die Häufigkeit dieser. Nun kann man mit einem bekannten Energiespektrum diese Kanäle kalibrieren, in denen man ihnen eine 'charakteristische Energie' zuordnet. Damit werden die Kanalachsen zu Energieachsen umskaliert und man kann die Energiespektren von beliebigen Stoffen analysieren.

1.3.2 Röntgenfluoreszenz

Wenn die ausgesendete Röntgenstrahlung auf ein Atom des zu analysierenden Stoffes trifft, kommt es zur Energieübertragung. Das kann zum Beispiel über Streuung bei einem Stoß oder durch Absorption passieren. Es kann dabei dazu kommen, dass ein Elektron aus den inneren Schalen des Atoms aus seiner Schale geschlagen wird, und ein Elektron der höheren Schalen nachrückt. Da das Elektron der höheren Schale nun aber Energie abgeben muss, um die neue Schale zu besetzen, strahlt sie diese Energie in Form von Strahlung ab. Da die Energie, welche bei einem Übergang freigegeben wird, immer die selbe ist (quantisierte Energie) und alle gemeinsam charakteristisch für einen Stoff sind, nennt man diese charakteristische Strahlung 'Röntgenfluoreszenz'.

1.3.3 Benennung der charakteristischen Strahlung

Für jeden Übergang der Elektronen in eine neue Schale wird eine charakteristische Energie abgestrahlt. Diese machen sich als Peaks in dem Spektrum sichtbar. Wenn man nun also weiß, welcher Peak zu welcher Energie gehört, kann man davon herausfinden, welcher Übergang stattgefunden hat. Die Benennung besteht aus einem Buchstaben, welcher zu der Schale gehört, in welcher das Elektron endet. Endet es beispielsweise in der K-Schale, so wäre der Buchstabe K. Auch in der Benennung ist ein griechischer Buchstabe, welcher angibt, 'wie weit' das Elektron gewandert ist. Ist es eine Schale gewandert (z.B von L-Schale zu K-Schale), so ist der Buchstabe α. Wandert es zwei Schalen ist er β und so weiter. So würde man einen Übergang von O-Schale in M-Schale mit M_β bezeichnen.

1.3.4 Moseleysches Gesetz

Das Moseleysche Gesetz dient dazu, die charakteristischen Energien der Übergänge zu nähern. Das Gesetz lautet:

$$E = E_R(Z - \sigma_{1,2})^2 \left(\frac{1}{n_1^2} - \frac{1}{n_2^2}\right) \tag{1}$$

E_R bezeichnet die Rydberg-Energie, welche man mittels $E_R = hcR_\infty$ berechnen kann. h ist hier die Planck-Konstante, c die Lichtgeschwindigkeit und R_∞ die Rydberg-Konstante. Die Rydberg-Energie beträgt ungefähr 13.6eV. Auch in dem Gesetz verwendet wird Z, was die Ordnugnszahl des Elementes ist, und $\sigma_{1,2}$, was die mittlere Abschirmkonstante ist. n_1 und n_2 sind die Schalen, auf denen das Elektron beginnt (n_2) und endet (n_1).

1.3.5 Halbleiterdetektor/Röntgenenergiedetektor

Um einen Halbleiterdetektor herzustellen, verbindet man einen n-Halbleiter und einen p-Halbleiter. Ein n-Halbleiter ist ein Halbleiter mit Elektronenüberschuss, ein p-Halbleiter ein Halbleiter mit Elektronenmangel. An der Grenzfläche der beiden wandern nun Elektronen von dem n-Halbleiter zu den 'Löchern' im p-Halbleiter (positive Atome). Somit breitet sich von der Grenzfläche ausgehend eine so genannte 'Verarmungszone' aus, welche neutral geladen ist. In dieser Zone bleiben jedoch die fest gebundenen, ionisierten Donatoren im n-Halbleiter und die ionisierten Akzeptoren im p-Halbleiter zurück, wodurch ein internes elektrisches Feld aufgebaut wird. Dieses Feld wirkt der weiteren Diffusion von Ladungsträgern entgegen, bis ein Gleichgewicht erreicht ist. Wenn nun eine äußere Spannung angelegt wird, mit der man den pn-Halbleiter in Sperrrichtung betreibt, so vergrößert sich besagte Verarmungszone durch das 'absaugen' weiterer Ladungen.
Wenn nun ein Röntgenquant auf diese trifft, kann es sein, dass ein Elektron aus der Verarmungszone herausgelöst wird. Dieses verliert seine Energie durch Stöße mit umliegenden Atomen, wodurch weitere Löcher-Elektronen-Paare entstehen. Diese Ladungen werden mit der angelegten Spannung abgesaugt und gemessen. Die Signale sind proportional zur Energie der einfallenden Photonen.

2 Messprotokoll

Messprotokoll: Röntgenfluoreszens

26.11.24
Harish Mahendran
Jonas Schweer

Messgeräte

- Röntgengerät mit Röntgenröhre (Molybdän-Anode)
- Röntgenenergiedetektor
- Vielkanalanalysator
- Metallproben
- Computer

Versuchsdurchführung

Wir nehmen in dem digitalen Rechenprogramm zur Auswertung elektronischer Datenströme folgende Einstellungen vor:

- 512 Kanäle
- negative Pulse
- Verstärkung = Faktor -2,5
- Messzeit = 180s

An unserem Röntgenstrahl-analysierapparatus nehmen wir zudem folgende Einstellungen vor:

- 35 kV
- 1 mA

Wir legen nun verschiedene Platten in den Analysator und stellen die Hochspannung an. Mit dem PC (Persönlicher Computer, verbaut: Intel Core i5-3470 4GB DDR 3 Ram) nehmen wir die Messwerte auf. Wir messen die Metalle:

Fe, Cu, Ag, Zn, Zr, Ni, Ti, Mo

Wir legen an die K_α-Linie des Fe- und Mo-Spektrums eine Gaußverteilung an und notieren den Mittelwert und die Standardabweichung:

Fe: $\mu = 81{,}943$ $\sigma = 3{,}083$

Mo: $\mu = 231{,}3$ $\sigma = 2{,}715$

Mit diesen Messwerten kalibrieren wir unsere Energieskala

Nun legen wir weitere Gaußkurven an die restlichen Linien an und notieren die Werte:

Cu: $\mu_\alpha = 7{,}84\,keV$, $\sigma_\alpha = 0{,}33\,keV$ (blau)
$\mu_\beta = 12{,}02\,keV$, $\sigma_\beta = 0{,}83\,keV$

Ag: $\mu_\alpha = 21{,}9\,keV$, $\sigma_\alpha = 0{,}19\,keV$ (Lila)
$\mu_\beta = 24{,}5\,keV$, $\sigma_\beta = 0{,}2\,keV$

Zn: $\mu_\alpha = 8{,}44\,keV$, $\sigma_\alpha = 0{,}34\,keV$ (grün)
$\mu_\beta = 12{,}83\,keV$, $\sigma_\beta = 1{,}22\,keV$

Zr: $\mu_\alpha = 15{,}84\,keV$, $\sigma_\alpha = 0{,}2\,keV$ (dunkelblau)
$\mu_\beta = 17{,}67\,keV$, $\sigma_\beta = 0{,}23\,keV$

Ni: $\mu_\alpha = 7{,}23\,keV$, $\sigma_\alpha = 0{,}38\,keV$ (magenta)
$\mu_\beta = 11{,}19\,keV$, $\sigma_\beta = 0{,}37\,keV$

Ti: $\mu_\alpha = 4{,}96\,keV$ $\sigma_\alpha = 0{,}19\,keV$ (braun)
$\mu_\beta = 7{,}38\,keV$ $\sigma_\beta = 0{,}26\,keV$

Mo: $\mu_\alpha = 17{,}51\,keV$ $\sigma_\alpha = 0{,}17\,keV$ (schwarz)
$\mu_\beta = 19{,}57\,keV$ $\sigma_\alpha = 0{,}2\,keV$

Fe: $\mu_\alpha = 6{,}4\,keV$ $\sigma_\alpha = 0{,}3\,keV$ (rot)
$\mu_\beta = 9{,}81\,keV$ $\sigma_\beta = 0{,}26\,keV$

Nun vermessen wir drei unbekannte Legierungen mit dem Programm um die chem. Zusammensetzung herauszufinden.

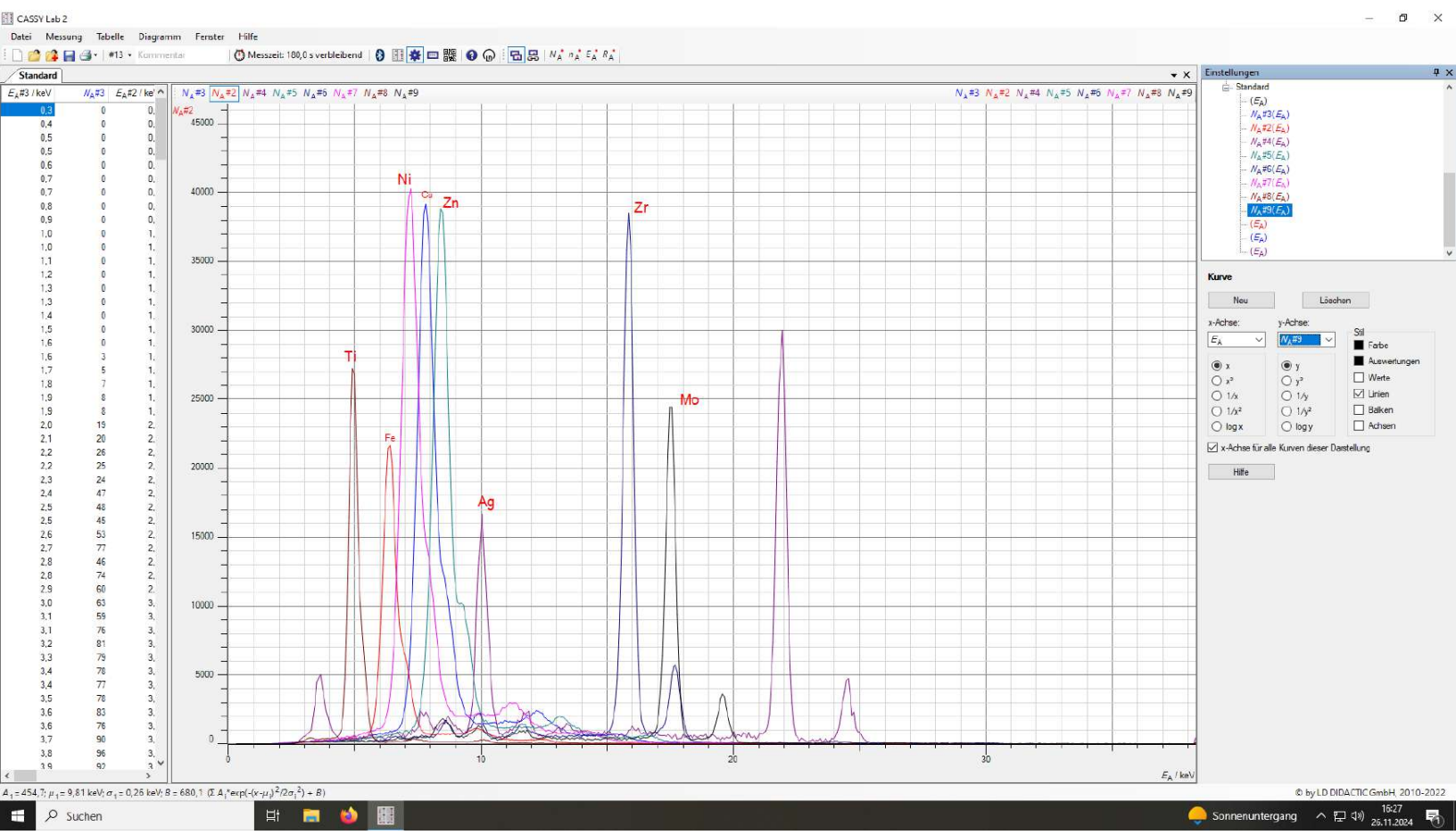
Legierung 5: W + Ga

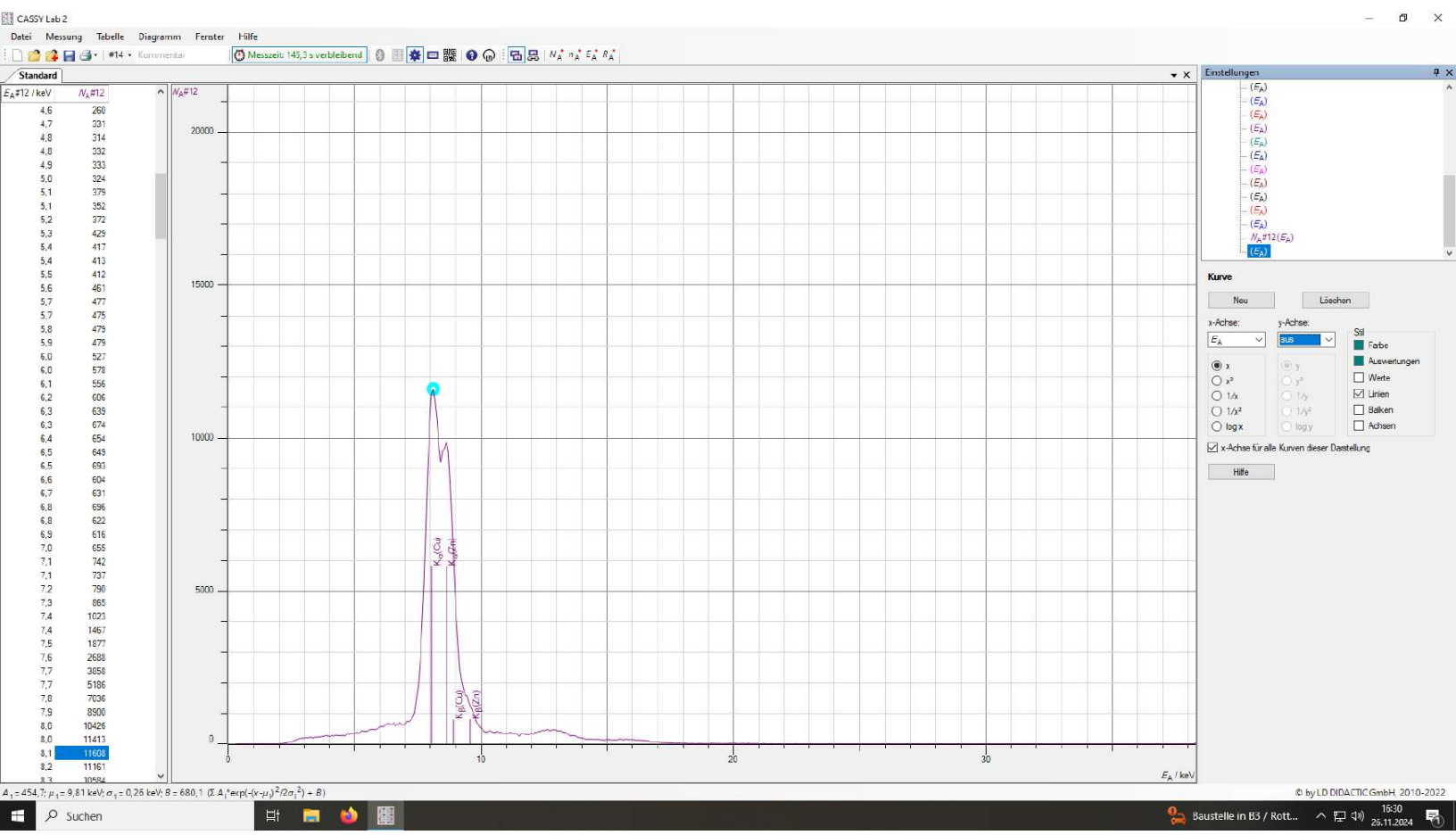
Legierung 3: Cu + Zn (Messing)

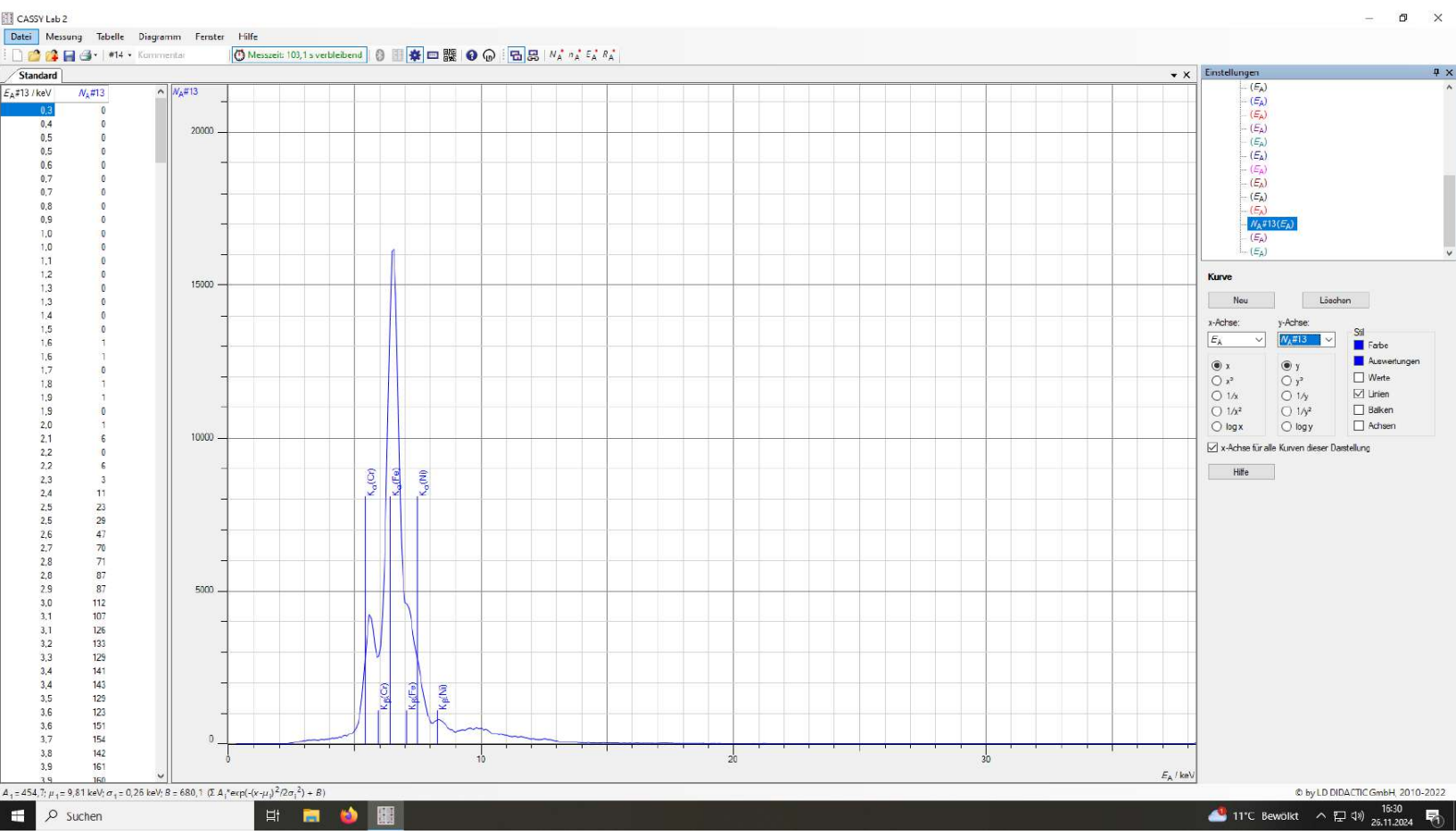
Legierung 1: Fe + Cr + Ni (Edelstahl)

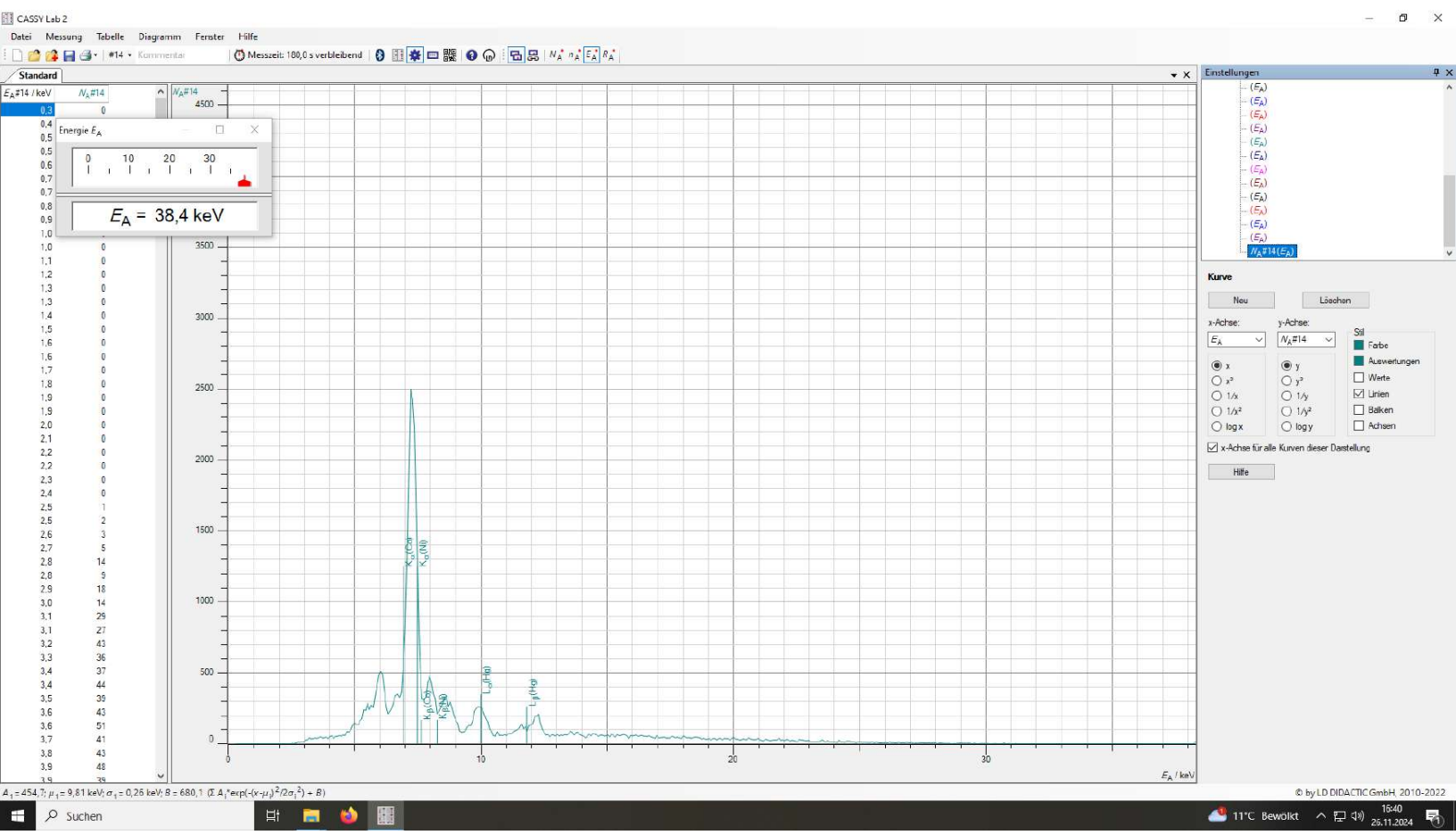
Magnet: Co + Ni (+ Hg ? Ergibt aber wenig Sinn)

26.11 AGB


CASSY Lab 2
Datei Messung Tabelle Diagramm Fenster Hilfe
Standard
Einstellungen
Kurve
Neu
Löschen
x-Achse:
y-Achse:
Stil
Farbe
Auswertungen
Werte
Linien
Balken
Achsen
x-Achse für alle Kurven dieser Darstellung
Hilfe
Ni
Co
Zn
Zr
Ti
Fe
Mo
Ag
E_A / keV
© by LD DIDACTIC GmbH, 2010-2022


CASSY Lab 2
Datei Messung Tabelle Diagramm Fenster Hilfe
Messzeit: 145,3 s verbleibend
Standard
E_A#12 / keV
N_A#12
4,6 260
4,7 331
4,8 314
4,8 332
4,9 333
5,0 324
5,1 379
5,1 352
5,2 372
5,3 429
5,4 417
5,4 413
5,5 412
5,6 461
5,7 477
5,7 475
5,8 479
5,9 479
6,0 527
6,0 578
6,1 556
6,2 606
6,3 639
6,3 674
6,4 654
6,5 649
6,5 693
6,6 604
6,7 631
6,8 696
6,8 622
6,9 616
7,0 655
7,1 742
7,1 737
7,2 790
7,3 865
7,4 1023
7,4 1467
7,5 1877
7,6 2688
7,7 3658
7,7 5186
7,8 7036
7,9 8900
8,0 10426
8,0 11413
8,1 11608
8,2 11161
N_A#12
20000
15000
10000
5000
0
0
10
20
30
E_A / keV
K_α(Cu)
K_α(Zn)
K_β(Cu)
K_β(Zn)
Einstellungen
(E_A)
N_A#12 (E_A)
Kurve
Neu
Löschen
x-Achse:
y-Achse:
aus
Stil
Farbe
Auswertungen
Werte
Linien
Balken
Achsen
x
x^2
1/x
$1/x^2$
log x
log y
x-Achse für alle Kurven dieser Darstellung
Hilfe
A_1= 454,7; μ_1= 9,81 keV; σ_1 = 0,26 keV; B = 680,1 (Σ A_i*exp(-(x-μ_i)^2/(2σ_i^2)) + B)
© by LD DIDACTIC GmbH, 2010-2022
Suchen
Baustelle in B3 / Rott...
16:30
26.11.2024


CASSY Lab 2
Datei
Messung
Tabelle
Diagramm
Fenster
Hilfe
Messzeit: 103,1 s verbleibend
Standard
E_A#13 / keV
N_A#13
N_A#13
20000
15000
10000
5000
0
0
10
20
30
E_A / keV
K_α(Cr)
K_α(Fe)
K_α(Ni)
K_β(Cr)
K_β(Fe)
K_β(Ni)
Einstellungen
N_A#13(E_A)
Kurve
Neu
Löschen
x-Achse:
y-Achse:
E_A
N_A#13
Stil
Farbe
Auswertungen
Werte
Linien
Balken
Achsen
x
x^2
1/x
$1/x^2$
log x
y
y^2
1/y
$1/y^2$
log y
x-Achse für alle Kurven dieser Darstellung
Hilfe
A_1= 454,7; μ_1= 9,81 keV; σ_1 = 0,26 keV; B = 680,1 (Σ A_i*exp(-(x-μ_i)^2/(2σ_i^2)) + B)
© by LD DIDACTIC GmbH, 2010-2022
Suchen
11°C Bewölkt
16:30
26.11.2024


CASSY Lab 2
Datei Messung Tabelle Diagramm Fenster Hilfe
Standard
Energie E_A
E_A = 38,4 keV
N_A#14
E_A / keV
Einstellungen
Kurve
Neu
Löschen
x-Achse:
y-Achse:
Stil
Farbe
Auswertungen
Werte
Linien
Balken
Achsen
x-Achse für alle Kurven dieser Darstellung
Hilfe
© by LD DIDACTIC GmbH, 2010-2022
11°C Bewölkt
Suchen
16:40
26.11.2024

3 Auswertung

3.1 K_α-Linien Auswertung

Nun wollen wir zuerst einmal unsere K_α-Linien verwenden, um das Moseleysche Gesetz zu überprüfen und die Rydberg-Energie zu bestimmen. Dafür plotten wir die Wurzel der Energien der K_α-Linien in Abhängigkeit der Ordnungszahl. Den Fehler der Energien haben wir in dem Programm ablesen können. Demnach ist der Fehler der Wurzel der Energie:

$$\Delta\left(\sqrt{E_\alpha}\right) = \frac{\Delta E_\alpha}{2\sqrt{E_\alpha}} \tag{2}$$

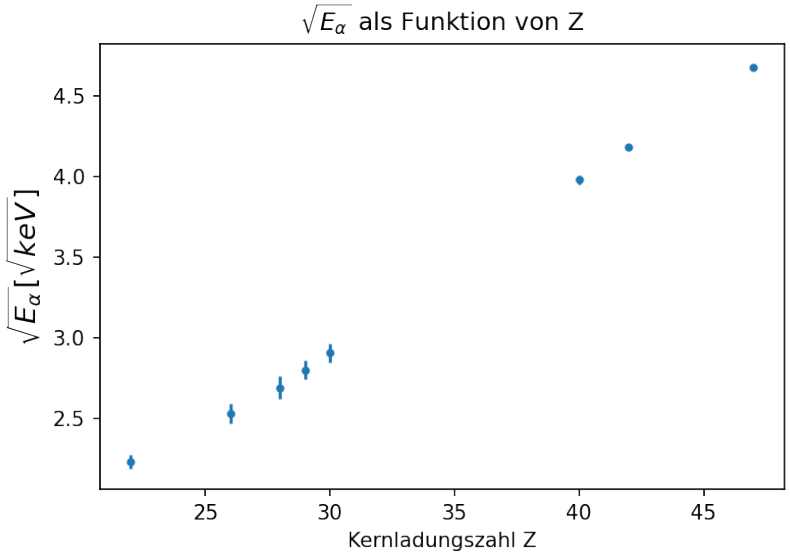


Abbildung 2: Wurzel der Energien der K_α-Linien

Wir legen nun mit Hilfe der curve-fit-Funktion von Python eine Gerade durch die Messwerte, um so die Wurzel der Rydberg-Energie und den Abschwächungskoeffizienten $\sigma_{1,2}$ zu erhalten. Gefittet sieht die Funktion so aus:

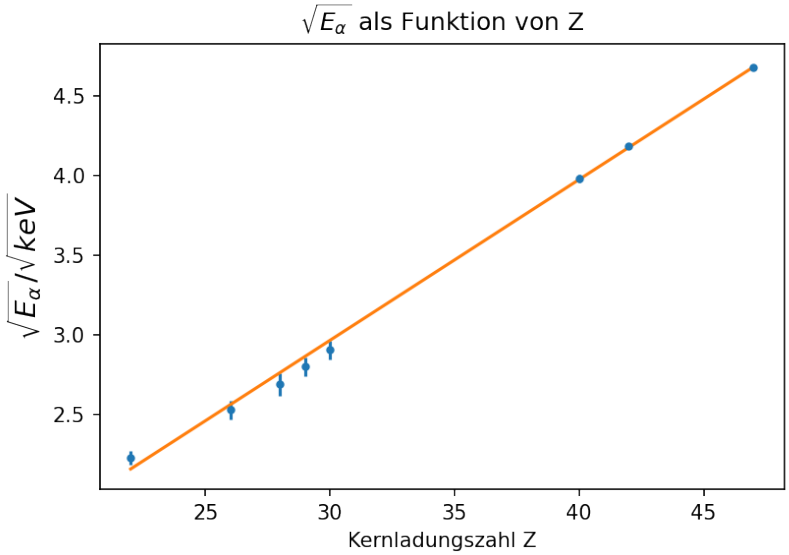


Abbildung 3: Gefittetes Moseleysches Gesetz der α-Linien

Die Parameter, die uns Python ausgibt, lauten:

$$\boxed{\sqrt{E_\alpha} = (0.1166 \pm 0.0018)\sqrt{\text{keV}}}$$
$$\boxed{\sigma_{1,2} = (0.6 \pm 0.6)}$$

Wenn wir $\sqrt{E_\alpha}$ quadrieren, erhalten wir die Rydberg-Energie. Der Fehler verhält sich wie folgt:

$$\Delta E_\alpha = 2\sqrt{E_\alpha} \cdot \Delta\left(\sqrt{E_\alpha}\right) \tag{3}$$

Wir erhalten schließlich also einen Wert für die Rydberg-Energie von:

$$\boxed{E_\alpha = (13.6 \pm 0.4)\text{eV}}$$

3.2 K_β-Linien Auswertung

Wir wiederholen den gesamten Vorgang erneut, jedoch nutzen wir diesmal unsere Werte für die K_β-Linien. Zuerst plotten wir wieder die Messwerte. Der Fehler ergibt sich wieder nach Formel(2).

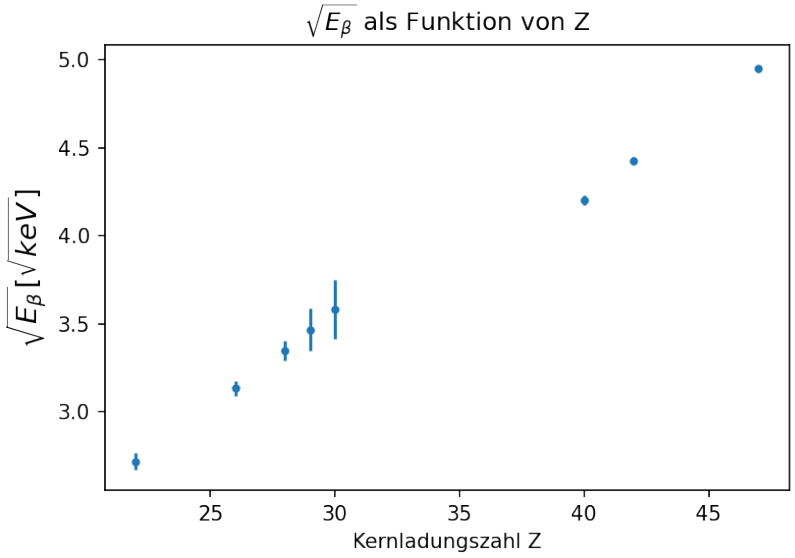


Abbildung 4: Wurzel der Energien der K_β-Linien

Auch hier legen wir wieder eine Fit-Gerade hindurch uns lassen uns $\sqrt{E_\beta}$ und $\sigma_{1,2}$ ausgeben. Wichtig ist, dass wir jetzt für $n_1 = 1$ und $n_2 = 3$ verwenden.

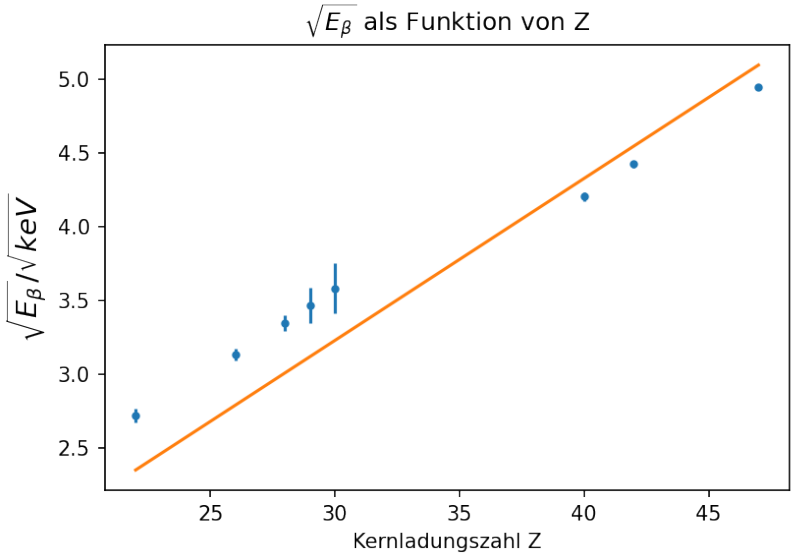


Abbildung 5: Gefittetes Moseleysches Gesetz der β-Linien

Die Werte lauten:

$$\boxed{\sqrt{E_\beta} = (0.092 \pm 0.003)\sqrt{\text{keV}}}$$
$$\boxed{\sigma_{1,2} = (-9.7 \pm 1.8)}$$

Den Energiewert erhalten wir nun genau wie in der Auswertung der K_α-Linien:

$\boxed{E_\beta = (8.4 \pm 0.6)\text{eV}}$

3.3 Überarbeitung der K_β-Linienmessung

Beim betrachten der Fitgerade der K_β-Messwerte und Ergebnisse fallen zwei Dinge stark auf. Zum einen ergibt eine negative Absorptionskonstante relativ wenig Sinn und zum anderen sieht man an dem Graphen, dass zwar die ersten fünf Messwerte schön in einer Reihe liegen, und die letzten drei Messwerte auch sehr gerade verlaufen, jedoch die Fitgerade keine der beiden wirklich gut erfassen kann, da die beiden geraden Verläufe nicht ineinander übergehen. Es wäre also naheliegend zu überprüfen, was passieren würde, wenn man eine Fitgerade durch die ersten fünf, sowie die letzten drei Werte legt. Der Plot würde so aussehen:

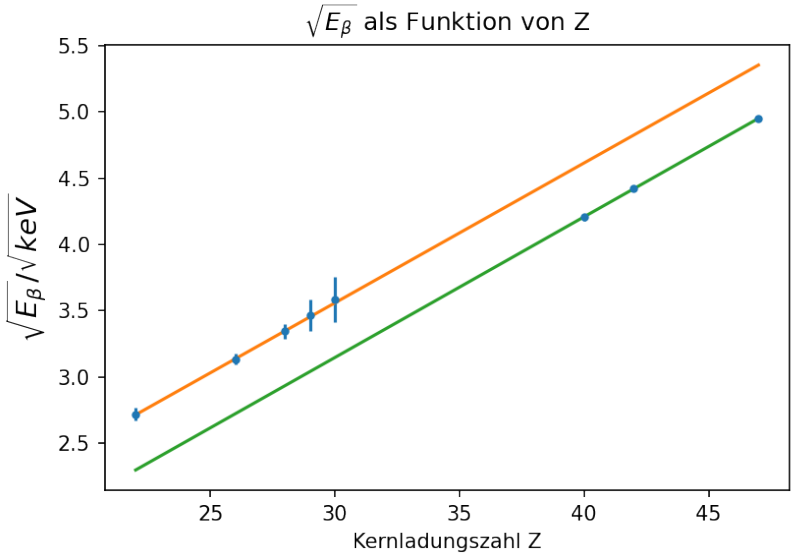


Abbildung 6: Fitgerade durch die ersten fünf (orange) und letzten drei (grün) Messwerte der K_β-Messung

Die Parameter lauten (E_β bereits durch quadrieren analog zu vorher berechnet):

$\boxed{E_{\beta,\text{orange}} = (12.5 \pm 0.3)\text{eV}}$
$\boxed{\sigma_{1,2,\text{orange}} = (-3.7 \pm 0.4)}$
$\boxed{E_{\beta,\text{grün}} = (12.67 \pm 0.25)\text{eV}}$
$\boxed{\sigma_{1,2,\text{grün}} = (0.4 \pm 0.4)}$

Zwar sind wir zumindest bei der grünen Fit-Gerade den negativen Abschirmungskonstante nicht losgeworden, jedoch scheint die gerade besser durch die Messwerte zu gehen.

4 Zusammenfassung und Diskussion

Um noch einmal Zusammenzufassen haben wir zuerst unser Vielkanalsystem kalibriert um qualitative Energiemessungen der fluoreszierten Röntgenstrahlen vornehmen zu können. Wir haben unsere Energiemessung der K_α- und K_β-Peaks genutzt, um so das Moseleysche Gesetz nachzuweisen. Wir haben mit Hilfe einer Fitgeraden die Absorptionskonstanten ermittelt sowie die Rydberg-Energie. Anschließend haben wir noch versucht, einen sinnvolleren Fit der Geraden bei den Energiemessungen der K_β-Linien durchzuführen. Unsere Ergebnisse für die Rydberg-Energie lauteten wie folgt:

$$\boxed{E_R^\alpha = (13.6 \pm 0.4)\text{eV}}$$
$$\boxed{E_R^\beta = (8.4 \pm 0.6)\text{eV}}$$
$$\boxed{E_{R,\text{fix1}}^\beta = (12.5 \pm 0.3)\text{eV}}$$
$$\boxed{E_{R,\text{fix2}}^\beta = (12.67 \pm 0.25)\text{eV}}$$

Vergleichen wir diese Werte mit dem im Skript angegebenen Referenzwert von $E_R \approx 13.6\text{eV}$ (ohne Fehler), so erhalten wir Abweichungen von:

$$\boxed{\sigma_\alpha = 0}$$
$$\boxed{\sigma_\beta = 8.7}$$
$$\boxed{\sigma_{\beta,fix1} = 3.7}$$
$$\boxed{\sigma_{\beta,fix2} = 3.7}$$

Wir sehen also, dass wir für die Messung der K_α-Linien keine Abweichung von dem gegebenen Referenzwert vorliegt. Diese Messung war also ein voller Erfolg. Unser Wert für die unveränderte K_β-Linien-Auswertung weist eine Abweichung von 8.7σ auf uns ist somit (wie erwartet) stark signifikant. Es scheint also tatsächlich ein Fehler bei der Messung unterlaufen zu sein. Bei den neu gelegten Geraden kommt beide male eine Abweichung von 3.7 heraus. Diese Abweichung ist immer noch signifikant, jedoch bei weitem nicht so sehr wie die der unveränderten Messung. Die Abweichung hat sich mehr als halbiert, obwohl sich die Unsicherheit des Ergebnisses ebenfalls halbiert hat. Wir sind alles in allem also näher an das eigentliche Ergebnis gekommen. Die restliche Abweichung ist vermutlich ein Nebenprodukt des selben Durchführungsfehlers, der uns auch schon dazu bewegt hat, zwei Geraden einzuführen.
Als Abschirmungskonstanten haben wir folgende bestimmt:

$$\boxed{\sigma_{1,2,\alpha} = (0.6 \pm 0.6)}$$
$$\boxed{\sigma_{1,2,\beta} = (-9.7 \pm 1.8)}$$
$$\boxed{\sigma_{1,2,\beta,fix1} = (-3.7 \pm 0.4)}$$
$$\boxed{\sigma_{1,2,\beta,fix2} = (0.4 \pm 0.4)}$$

Verglichen mit den Literaturwerten, welche sich auf Wikipedia einsehen lassen von $\sigma_{1,2,\alpha} \approx 1.0$, $\sigma_{1,2,\beta} \approx 1.8$ lauten die Abweichungen:

$\boxed{\sigma_\alpha = 0.67}$
$\boxed{\sigma_\beta = 6.4}$
$\boxed{\sigma_{\beta,fix1} = 13.75}$
$\boxed{\sigma_{\beta,fix2} = 3.5}$

Wir sehen also erneut, dass unsere Messung der K_α-Linien nicht signifikant abweicht, und gut den erwarteten Wert widerspiegelt. Die Abschirmungskonstanten der β-Messungen wiederum weichen alle signifikant von dem Richtwert ab. Nach unseren vorherigen Erkenntnissen war dies allerdings auch eigentlich zu erwarten. Die Abschirmungskonstante stellt quasi den y-Achsenabschnitt unseres $\sqrt{E}(Z)$-Plots da. Offensichtlich war unser erster Plot keiner, aus dem man irgendwelche Informationen ziehen kann, also ist es nicht verwunderlich, dass dort auch die Abschirmungskonstante unsinnig ist. Sobald wir allerdings die zwei verschiedenen Geraden eingeführt haben (die nahezu parallel verliefen), war die Idee, einen y-Achsenabschnitt zu messen, bereits hinfällig. Alles, was wir sagen können, ist, dass die grüne Gerade scheinbar näher an dem echten Verlauf der Gerade ist, da dort die Abweichung kleiner ist. Die Messwerte mit größerer Kernladungszahl Z waren also wohl die, bei denen weniger Fehler passiert sind. Was konkret bei der Durchführung schiefgelaufen ist, kann leider nur vermutet werden, da uns beim Durchführen nichts aufgefallen ist. Mögliche Probleme können zum Beispiel beim Kalibrieren des Gerätes vorgekommen sein, jedoch ist es dann erstaunlich, dass die K_α-Messung doch so genau war. Andererseits könnten natürlich immer die Platten nicht richtig in dem Gerät gelegen haben.
Bei einer erneuten Durchführung des Versuchs würde ich vermehrt darauf achten, dass unser Gerät korrekt kalibriert wird. Außerdem würde ich allgemein beim einlegen der Plättchen besser aufpassen um sicherzustellen, dass diese richtig liegen. Mehrere Messungen wären sicher auch hilfreich, jedoch wäre das zeitintensiver und würde eventuell nicht mal einen großen Unterschied machen.

5 Anhang

Versuch_256_Roentgenfluoreszenz

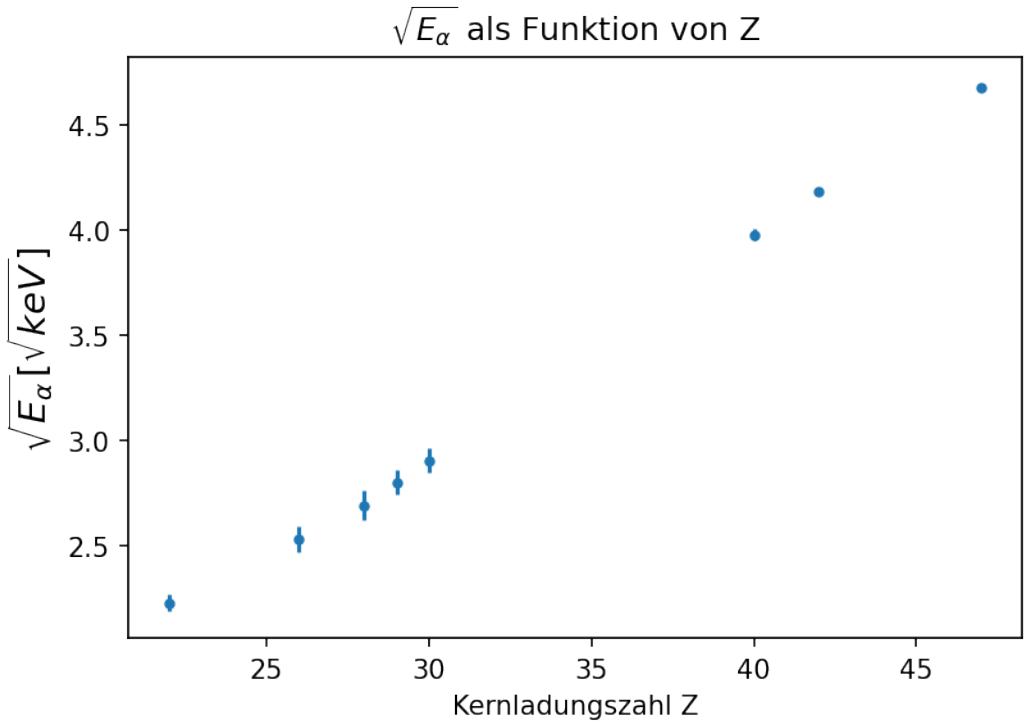
January 15, 2025

[211]:
```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
from scipy.optimize import curve_fit
```

[212]:
```
Z=np.array([22,26,28,29,30,40,42,47])
#K_alpha (Ti, Fe, Ni, Cu, Zn, Zr, Mo, Ag) in keV:
K_alpha=np.array([4.96,6.4,7.23,7.84,8.44,15.84,17.51,21.9])
Delta_K_alpha=np.array([0.19,0.3,0.38,0.33,0.34,0.2,0.17,0.19])
sqrt_K_alpha=np.sqrt(K_alpha)
Delta_sqrt_K_alpha= Delta_K_alpha/(2*np.sqrt(K_alpha))
```

[213]:
```
#K_beta (Ti, Fe, Ni, Cu, Zn, Zr, Mo, Ag) in keV:
K_beta=np.array([7.38,9.81,11.19,12.02,12.83,17.67,19.57,24.5])
Delta_K_beta=np.array([0.26,0.26,0.37,0.83,1.22,0.23,0.2,0.2])
sqrt_K_beta=np.sqrt(K_beta)
Delta_sqrt_K_beta= Delta_K_beta/(2*np.sqrt(K_beta))
```

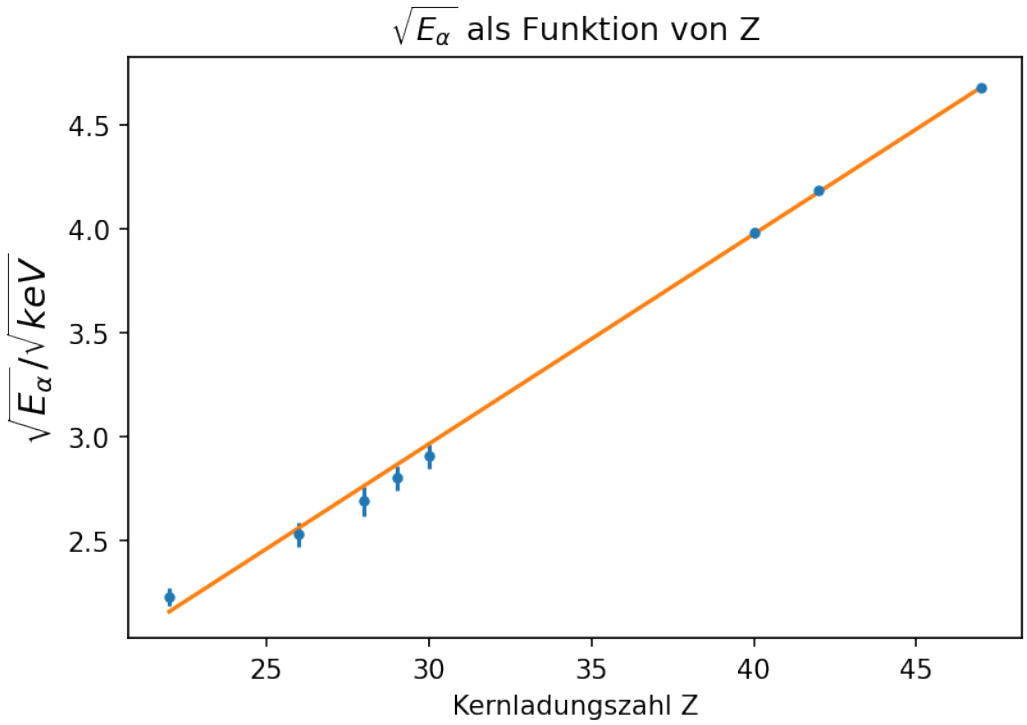
[214]:
```
plt.figure(dpi=150)
plt.errorbar(Z, sqrt_K_alpha, Delta_sqrt_K_alpha, fmt=".")
plt.xlabel('Kernladungszahl Z')
plt.ylabel(r'$\sqrt{E_\alpha}[\sqrt{keV}]$ ', fontsize=14)
plt.title(r'$\sqrt{E_\alpha}$' + ' als Funktion von Z')
plt.savefig("Fluoreszens/Plot1.png")
```



[215]:
```
#sqrt_Er, sig12 sind Fitparameter
n1=1
n2=2
def fit_func(x, sqrt_Er, sig12):
    return sqrt_Er*(x-sig12)*np.sqrt(1/n1**2-1/n2**2)
```

[216]:
```
popt1, pcov1=curve_fit(fit_func, Z, sqrt_K_alpha, sigma=Delta_sqrt_K_alpha)
```

[217]:
```
plt.figure(dpi=150)
plt.errorbar(Z, sqrt_K_alpha, Delta_sqrt_K_alpha, fmt=".")
plt.xlabel('Kernladungszahl Z')
plt.ylabel(r'$\sqrt{E_\alpha}/\sqrt{keV}$ ', fontsize=14)
plt.title(r'$\sqrt{E_\alpha}$' + ' als Funktion von Z')
plt.plot(Z, fit_func(Z,*popt1))
plt.savefig("Fluoreszens/Plot2.png")
```



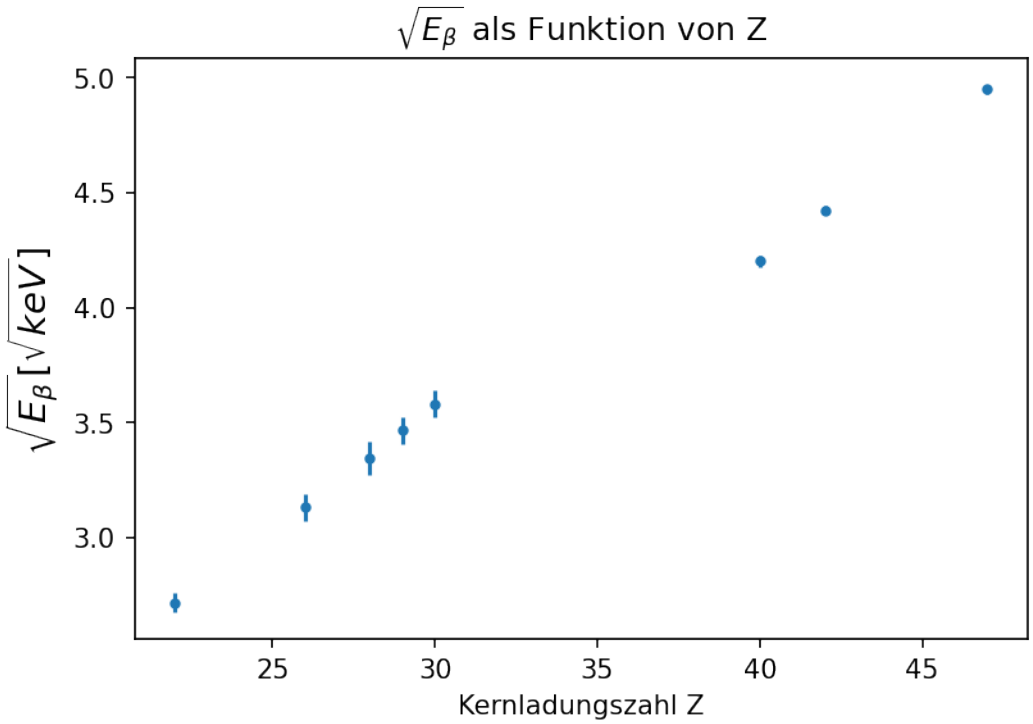
[218]:
```
print("sqrt_Er=",popt1[0], ",Standardfehler=",np.sqrt(pcov1[0][0]))
print("sig12=",popt1[1], ",Standardfehler=",np.sqrt(pcov1[1][1]))
```

```
sqrt_Er= 0.1165847660447819 ,Standardfehler= 0.0017798526345466097
sig12= 0.6307776248737295 ,Standardfehler= 0.6109103758527611
```

[219]:
```
print("Er=",popt1[0]**2, ",Standardfehler=",2*np.sqrt(pcov1[0][0])*popt1[0])
```

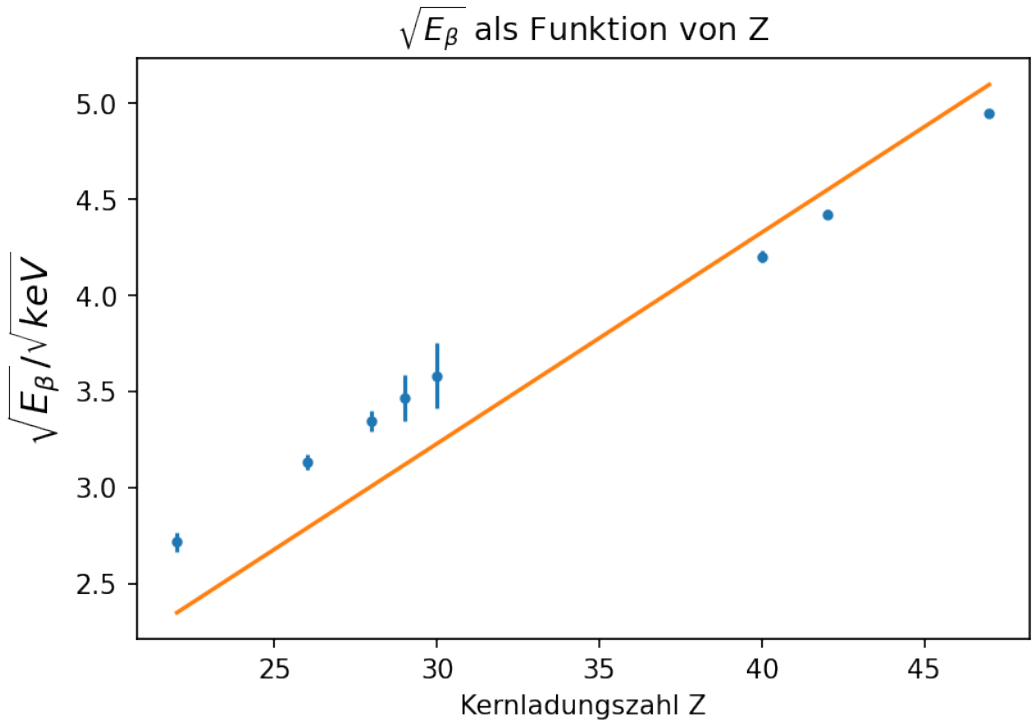
```
Er= 0.013592007673716532 ,Standardfehler= 0.0004150074059856103 7
```

[220]:
```
plt.figure(dpi=150)
plt.errorbar(Z, sqrt_K_beta, Delta_sqrt_K_alpha, fmt=".")
plt.xlabel('Kernladungszahl Z')
plt.ylabel(r'$\sqrt{E_\beta}[\sqrt{keV}]$ ', fontsize=14)
plt.title(r'$\sqrt{E_\beta}$' + ' als Funktion von Z')
plt.savefig("Fluoreszens/Plot3.png")
```



[221]:
```
n1=1
n2=3
popt2, pcov2=curve_fit(fit_func, Z, sqrt_K_beta, sigma=Delta_sqrt_K_beta)
```

[222]:
```
plt.figure(dpi=150)
plt.errorbar(Z, sqrt_K_beta, Delta_sqrt_K_beta, fmt=".")
plt.xlabel('Kernladungszahl Z')
plt.ylabel(r'$\sqrt{E_\beta}/\sqrt{keV}$ ', fontsize=14)
plt.title(r'$\sqrt{E_\beta}$' + ' als Funktion von Z')
plt.plot(Z, fit_func(Z,*popt1))
plt.savefig("Fluoreszens/Plot4.png")
```



[223]:
```
print("sqrt_Er=",popt2[0], ",Standardfehler=",np.sqrt(pcov2[0][0]))
print("sig12=",popt2[1], ",Standardfehler=",np.sqrt(pcov2[1][1]))
```

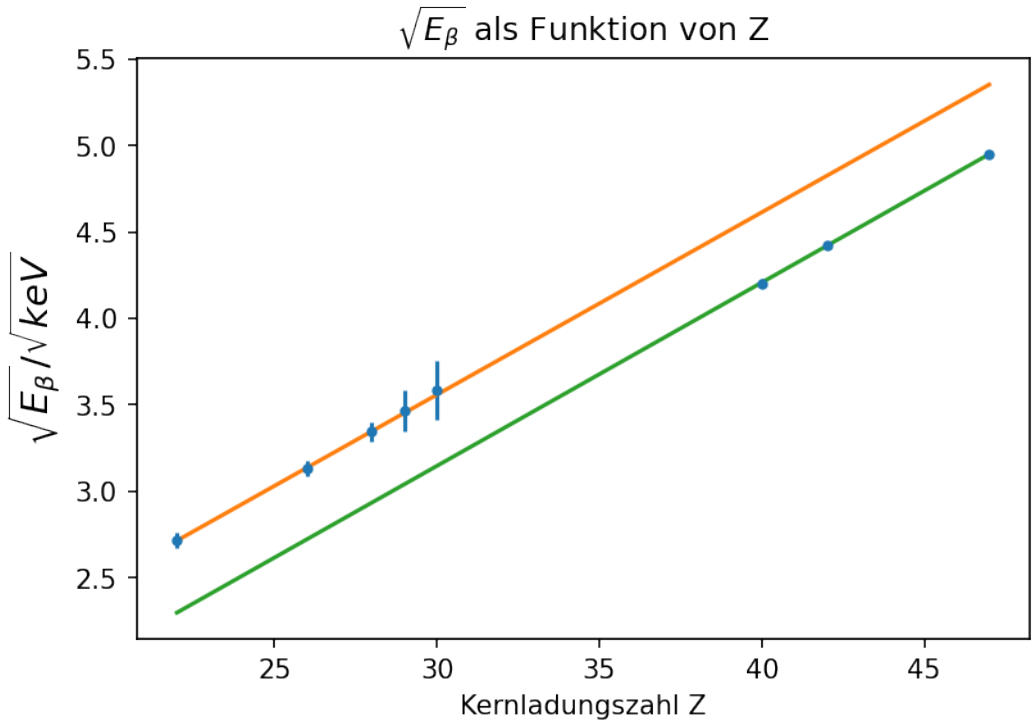
```
sqrt_Er= 0.091641536018317 ,Standardfehler= 0.0033696420592436416
sig12= -9.667331111545536 ,Standardfehler= 1.8490820759243123
```

[224]:
```
print("Er=",popt2[0]**2, ",Standardfehler=",2*np.sqrt(pcov2[0][0])*popt2[0])
```

```
Er= 0.008398171123796492 ,Standardfehler= 0.0006175983482820241
```

[225]:
```
popt3, pcov3=curve_fit(fit_func, Z[:-3], sqrt_K_beta[:-3],␣
 ↪sigma=Delta_sqrt_K_beta[:-3])
popt4, pcov4=curve_fit(fit_func, Z[5:], sqrt_K_beta[5:],␣
 ↪sigma=Delta_sqrt_K_beta[5:])
```

[226]:
```
plt.figure(dpi=150)
plt.errorbar(Z, sqrt_K_beta, Delta_sqrt_K_beta, fmt=".")
plt.xlabel('Kernladungszahl Z')
plt.ylabel(r'$\sqrt{E_\beta}/\sqrt{keV}$ ', fontsize=14)
plt.title(r'$\sqrt{E_\beta}$' + ' als Funktion von Z')
plt.plot(Z, fit_func(Z,*popt3))
plt.plot(Z, fit_func(Z,*popt4))
plt.savefig("Fluoreszens/Plot5.png")
```



[228]:
```
print("sqrt_Erorange=",popt3[0], ",Standardfehler=",np.sqrt(pcov3[0][0]))
print("sig12orange=",popt3[1], ",Standardfehler=",np.sqrt(pcov3[1][1]))
print("Erorange=",popt3[0]**2, ",Standardfehler=",2*np.
 ↪sqrt(pcov3[0][0])*popt3[0])
print("sqrt_Ergrün=",popt4[0], ",Standardfehler=",np.sqrt(pcov4[0][0]))
print("sig12grün=",popt4[1], ",Standardfehler=",np.sqrt(pcov4[1][1]))
print("Ergrün=",popt4[0]**2, ",Standardfehler=",2*np.sqrt(pcov4[0][0])*popt4[0])
```

```
sqrt_Erorange= 0.11196976771279374 ,Standardfehler= 0.001396410354354878
sig12orange= -3.7074963988605525 ,Standardfehler= 0.3654101175231285
Erorange= 0.012537228881656988 ,Standardfehler= 0.00031271148601771134
sqrt_Ergrün= 0.11258184710077514 ,Standardfehler= 0.001089886151999805
sig12grün= 0.3589135918244013 ,Standardfehler= 0.4202494613823439
Ergrün= 0.012674672296622311 ,Standardfehler= 0.00024540279224338844
```