

Versuch 234

Lichtquellen

Viktor Ivanov

17. März 2024

Inhaltsverzeichnis

1	Einleitung	2
1.1	Motivation	2
1.2	Physikalische Grundlagen	2
1.2.1	Temperaturstrahler	2
1.2.2	Nichttemperaturstrahler	3
1.2.3	Das Natriumspektrum	3
2	Messprotokoll und Durchführung des Versuchs	4
3	Auswertung	7
3.1	Auswertung der unterschiedlichen Lichtquellen	7
3.2	Auswertung des Sonnenspektrums	8
3.3	Auswertung des Natriumspektrums	9
3.4	Zuordnung der gefundenen Linien zu Serien	10
3.4.1	Erwartete Linien für die 1. Nebenserie: $md \rightarrow 3p$	11
3.4.2	Erwartete Linien für die 2. Nebenserie: $ms \rightarrow 3p$	13
3.4.3	Erwartete Linien für die Hauptserie: $mp \rightarrow 3s$	15
3.5	Bestimmung der Serienenergien und der l-abhängigen Korrekturfaktoren	15
3.5.1	1. Nebenserie	15
3.5.2	2. Nebenserie	16
4	Zusammenfassung und Diskussion	17
5	Anhang	19
5.1	Quellen	19
5.2	Python-Code	19

1 Einleitung

1.1 Motivation

Das Ziel dieses Versuchs ist, Sonnen-, Glühlampen-, LED- und Energiesparlampenlicht zu untersuchen und zu vergleichen und schließlich den Natriumspektrum quantitativ zu untersuchen.

1.2 Physikalische Grundlagen

1.2.1 Temperaturstrahler

Die Körper, die eine Temperatur anders als die absolute Null betragen, senden elektromagnetische Strahlung aus, deren Intensität von der Temperatur abhängt. Ein schwarzer Strahler ist ein idealisierter Körper, der die einfallende Strahlung vollständig absorbiert, sodass ein Körper ein maximales Emissionsvermögen von $\epsilon = 1$ hat. Das Plancksche Strahlungsgesetz beschreibt die Intensitätsverteilung der Strahlung, die von einem schwarzen Strahler ausgeht:

$$M_\lambda(\lambda, T) d\lambda = \frac{2\pi hc^2}{\lambda^5} \frac{1}{\exp\left(\frac{hc}{\lambda kT}\right) - 1} d\lambda \quad (1)$$

Wobei hier M_λ die Strahlungsleistung ist, λ die Wellenlänge, h die Plancksche Wirkungsquantum, c die Lichtgeschwindigkeit und k die Boltzmann-Konstante.

In Abbildung 1 kann man die Intensitätsverteilung eines schwarzen Körpers sehen.

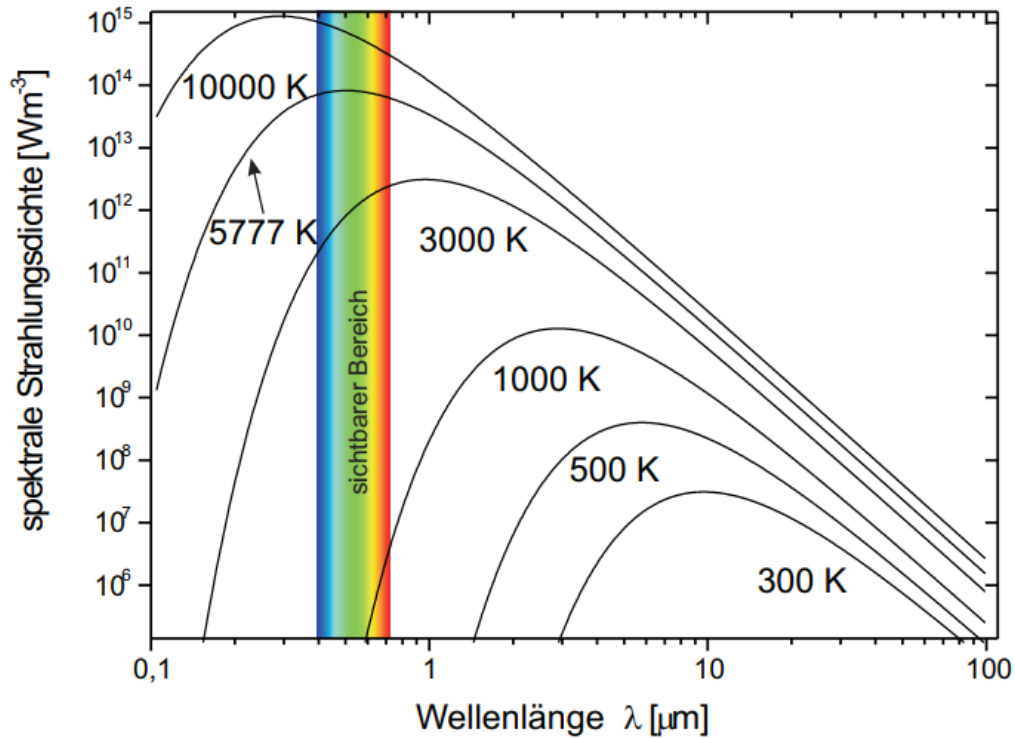


Abbildung 1: Spektrale Intensitätsverteilung eines schwarzen Körpers

Für die Wellenlänge des Intensitätsmaximums, der sich mit zunehmender Temperatur verschiebt, gilt das Wien'sche Verschiebungsgesetz:

$$\lambda_{max} = \frac{2897,8 \mu\text{m}K}{T} \quad (2)$$

Temperaturstrahler erscheinen bei der absoluten Null komplett schwarz, mit Erhöhung der Temperatur sehen sie immer roter aus, bis zu ungefähr $5500K$, wenn alle Wellenlängen mit der ungefähr gleichen Intensität wirken. Dann sehen sie immer blauer aus.

Blaues Licht ist bezeichnet als "kalt" und rotes - "warm". Direktes Sonnenlicht ist warm und gestreute Sonnenlicht ist kalt wegen der Rayleigh-Streuung.

Das Spektrum von Sonnenlicht ist von vielen Absorptionslinien durchgezogen, die von Stoffen als z.B. Wasserdampf, Sauerstoff, Ozon und Kohlendioxid verursacht werden.

1.2.2 Nichttemperaturstrahler

Das Licht aus Nichttemperaturstrahlern ist abhängig von der Anregung von bestimmten Atomzuständen in Gasen oder Festkörpern oder durch Rekombination von Elektron-Loch Paaren in Halbleitern. Im Gegensatz zu Temperaturstrahlern, die ein kontinuierliches Spektrum haben, haben diese ein diskretes Spektrum. Zu diesem Typ von Strahlern gehören Leuchtstoffröhren, Leuchtdioden, oder LASER. Leuchtdioden sind viel effizienter als andere Lampen.

1.2.3 Das Natriumspektrum

Für große Radien können wir den Wasserstoffpotential für den Natriumpotential annähern:

$$V(r) = \frac{e^2}{4\pi\epsilon_0 r} \quad (3)$$

Eine gute Näherung für die Energieterme dann beträgt:

$$E_{n,l} = -13,6eV \frac{1}{(n - \Delta_{l,n})^2} \quad (4)$$

Da $\Delta_{l,n}$ wenig von n abhängt, können wir es auch als Δ_l schreiben. Die Photonübergänge, die mit 4 beschrieben werden können, sind in Abbildung 2 zu finden. Die Spektren der Natriumserien sind in Abbildung 3 zu sehen.

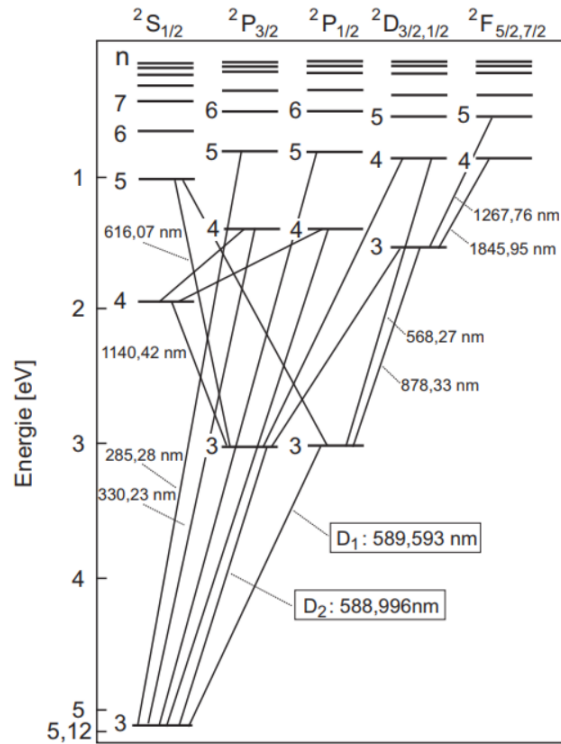


Abbildung 2: Energieschema und Photonübergänge im Natriumatom

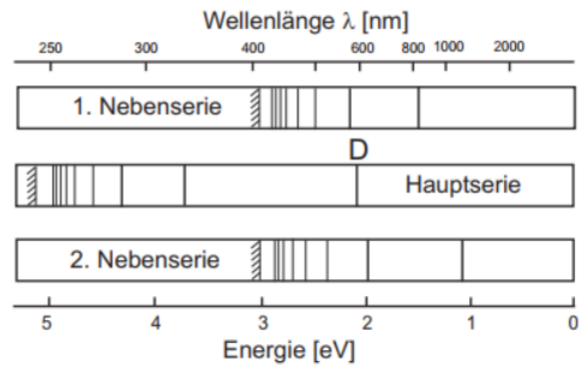


Abbildung 3: Spektren der Natriumserien

2 Messprotokol und Durchführung des Versuchs

Das Messprotokoll befindet sich auf der nächsten Seite.

I Messaufbau

- Gitterspektrometer (GSM)
- Verschiedene Lichtquellen
- PC mit Drucker

II Durchführung

- 1) Im ersten Versuchsteil nehmen wir mit dem GSM das Spektrum der Sonne auf.
Ein Mal direkt und ein Mal hinter einer Festscheibe. Die Intensitätsverteilungen werden mit dem Programm Ocean View analysiert.
- 2) Im zweiten Versuchsteil nehmen wir mit dem GSM das Spektrum von einer Glühlampe, Energiesparlampe, Laser und verschiedenen LED's auf. Auswertung mit OceanView
- 3) Im dritten Versuchsteil nehmen wir mit dem GSM das Spektrum von einer Na-Lampe auf. Auswertung mit OceanView
- 4) Im vierten Versuchsteil öffnen wir die Blende soweit, dass man die weniger intensiven Linien gerade erkennt.

Skizze



11.12.2023
Vanessa Deuschner

3 Auswertung

3.1 Auswertung der unterschiedlichen Lichtquellen

Im ersten Teil der Auswertung haben wir verschiedene Lichtquellen untersucht, darunter verschiedene LED-Farben, eine Energiesparlampe, eine Glühlampe und einen Laser. Die Daten aus den Messungen wurden mithilfe eines Python-Skripts in Intensität-Wellenlänge-Diagramme umgewandelt.

Auf Abbildung 5 (links) sind die LED-Farben blau, orange, gelb und rot dargestellt, während rechts zwei Messungen für weißes Licht einer LED-Lampe zu sehen sind.

Abbildung 4 zeigt die Messungen für weißes und farbiges Licht zusammen.

Die Intensitäten der Energiesparlampe, Glühlampe und des Lasers sind in Abbildung 6 dargestellt.

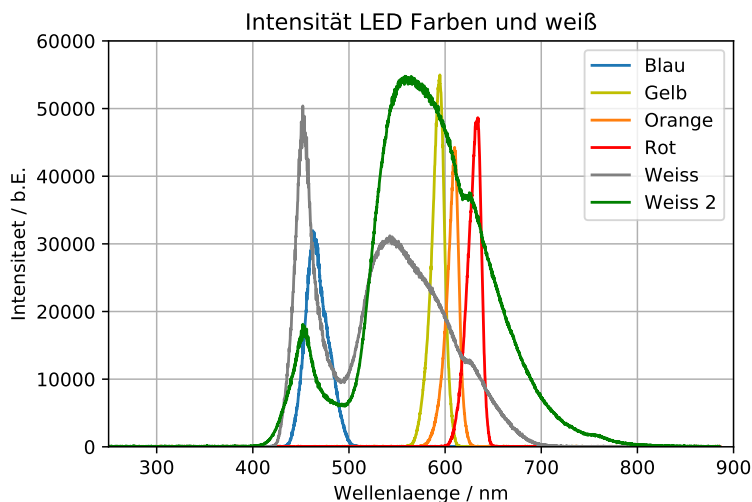


Abbildung 4: LED Intensitäten von allen Farben

In Tabelle 0 habe ich alle Daten aus den Abbildungen 5, 4 und 6 zusammengefasst.

Wir beobachten, dass die LED-Lampen, die nur eine Farbe emittieren, diskret sind und dass ihre Wellenlängendistribution mit dem Farbspektrum übereinstimmen. Die weiße Farbe wird durch die Kombination anderer Farben (additive Farbmischung) erzeugt. Es fällt auf, dass die weißen LED-Lampen im Vergleich zur Energiesparlampe viel mehr Energie verbrauchen, da sie kontinuierlich in vielen Wellenlängen hohe Lichtintensitäten emittieren, im Gegensatz zur Energiesparlampe, die nur drei schmale Peaks aufweist.

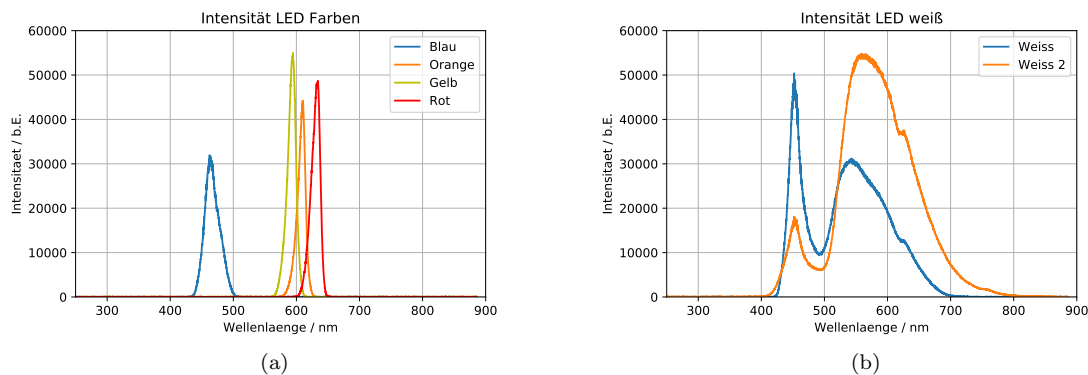


Abbildung 5: LED Farben (links) LED weiß (rechts)

Wir sehen, dass die weiße LED 1 kaltes Licht emittiert, im Vergleich zu der weißen LED 2, die warmes Licht emittiert.

Eine weitere Besprechung der Ergebnisse erfolgt in der Diskussion.

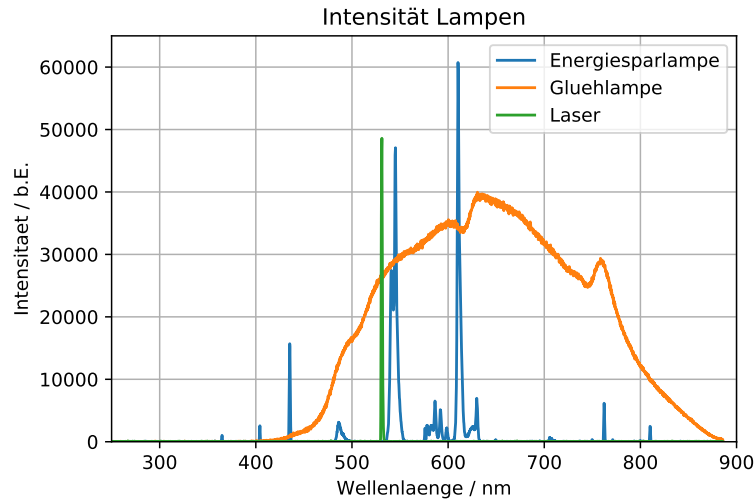


Abbildung 6: Intensitäten der Energiesparlampe, Glühlampe und Laser

Tabelle 0. Lampen und Farbenvergleich

Lampe	Farbe	Spektrum	Farbtemperatur	Peak 1 λ [nm]	Peak 2 λ [nm]	Peak 3 λ [nm]	Peak 4 λ [nm]	Ablesefehler $\Delta\lambda$ [nm]
Glühbirne	weiß	kontinuierlich	warm	495	600	625	755	10
Energiesparlampe	weiß	diskret	warm	420	540	605	-	10
LED weiß 1	weiß	kontinuierlich	kalt	450	520	-	-	10
LED weiß 2	weiß	kontinuierlich	warm	450	525	-	-	10
LED rot	rot	diskret	warm	625	-	-	-	10
LED orange	orange	diskret	warm	610	-	-	-	10
LED gelb	gelb	diskret	warm	595	-	-	-	10
LED blau	blau	diskret	kalt	465	-	-	-	10
LASER	grün	diskret	kalt	535	-	-	-	10

3.2 Auswertung des Sonnenspektrums

Zunächst wollen wir das Sonnenspektrum mit und ohne Fenster vergleichen. Beide Spektren haben wir auf einem Diagramm mithilfe von Python in Abbildung 7 gezeichnet.

Wie erwartet, die Intensität von Licht durch das Fenster ist niedriger als dieser ohne das Fenster. Die Absorption vom Glas A_{Glas} ist mit dieser Formel zu berechnen:

$$A_{Glas} = 1 - \frac{I_{mG}(\lambda)}{I_{OG}(\lambda)} \quad (5)$$

Wobei hier I_{mG} die Intensität ohne Glas ist und I_{OG} diese mit Glas ist. Der Absorptionsspektrum vom Glas ist in Abbildung 8 zu sehen. Es ist zu erkennen, dass der Fenster fast alle Licht mit einer Wellenlänge von unter c.a. $320nm$ absorbiert und auch mehr als die Hälfte vom Licht mit Wellenlänge von unter $380nm$, oder sogenannte UV Licht. Als erwartet, sehen wir eine schwächere Absorption im Teil vom sichtbaren Licht. Die Absorption nimmt dann ab c.a. $700nm$ ab, was schon IR Licht ist.

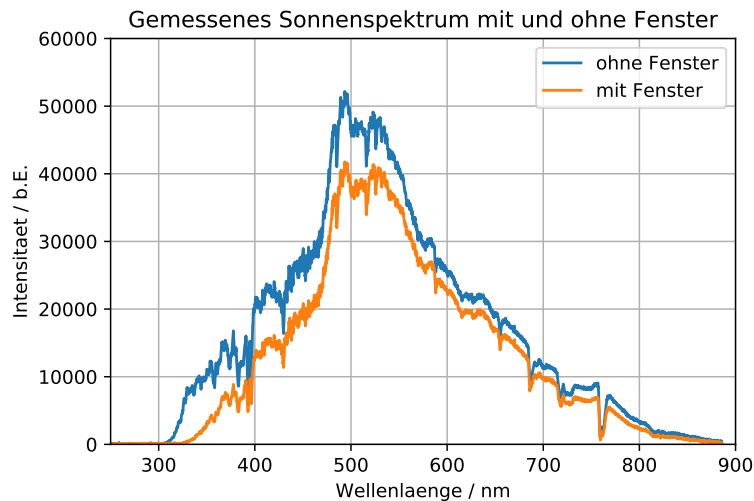


Abbildung 7: Vergleich von Lichtspektrum mit und ohne glas

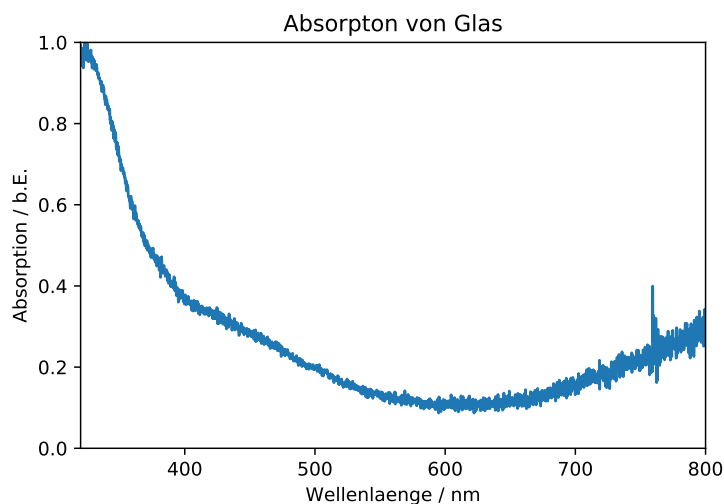


Abbildung 8: Absorptionsspektrum vom Fenster

Aus dem Sonnenlichtdiagramm ohne Glas habe ich die Fraunhoferlinien bestimmt und mit den Werten habe ich eine Tabelle erstellt, um die mit den theoretischen Werten zu vergleichen. Die Fraunhoferlinien auf dem Sonnenlichtdiagramm kann man in Abbildung 9 finden. Die verglichenen experimentellen und theoretischen Werte sind in Tabelle 1 zu finden.

Die gelbe Heliumlinie sollte eigentlich als D3 identifiziert werden, obwohl die drei D-Linien sehr nah und schwierig zu unterscheiden sind.

Die Balmer-Serie Linien sind in Abbildung 10 zu finden. In Tabelle 2 haben wir die gemessenen und theoretischen Werte der Balmer-Serie verglichen.

Alle Ergebnisse besprechen wir in der Diskussion.

3.3 Auswertung des Natriumspektrums

Zur Auswertung des Natriumspektrums haben wir drei während der Messung aufgenommene Bilder analysiert und mithilfe eines Python-Skripts die Wellenlängen der beobachteten Linien grafisch dargestellt. In Abbildung 11 sind die

Tabelle 1: Vergleich zwischen theoretischen und gemessenen Wellenlängen der Fraunhoferlinien

Symbol	Theoretische Wellenlänge λ [nm]	Gemessene Wellenlänge λ [nm]	Fehler Gemessene Wellenlänge $\Delta\lambda$ [nm]	Abweichung zw. theor. und exp. Wert λ [nm]	σ Abweichung [σ]	Element /Molekül
A	759.4	759.5	0.6	0.1	0.2	telluric oxygen
B	686.7	686.3	0.6	0.4	0.7	telluric oxygen
C	656.3	655.5	0.6	0.8	1.3	hydrogen
D_1	589.6	590.0	0.6	0.4	0.7	sodium
D_2	589.0	588.4	0.6	0.6	1.0	sodium
D_3	587.6	587.3	0.6	10.3	0.5	helium
E	527.0	526.1	0.6	0.9	1.5	iron and calcium
b	518.4	516.6	0.6	1.8	3.0	magnesium
F	486.1	485.8	0.6	0.3	0.5	hydrogen
G	430.8	429.7	0.6	1.1	1.8	iron and calcium
H	396.8	396.5	0.6	0.3	0.5	calcium
K	393.4	393.0	0.6	0.4	0.7	calcium

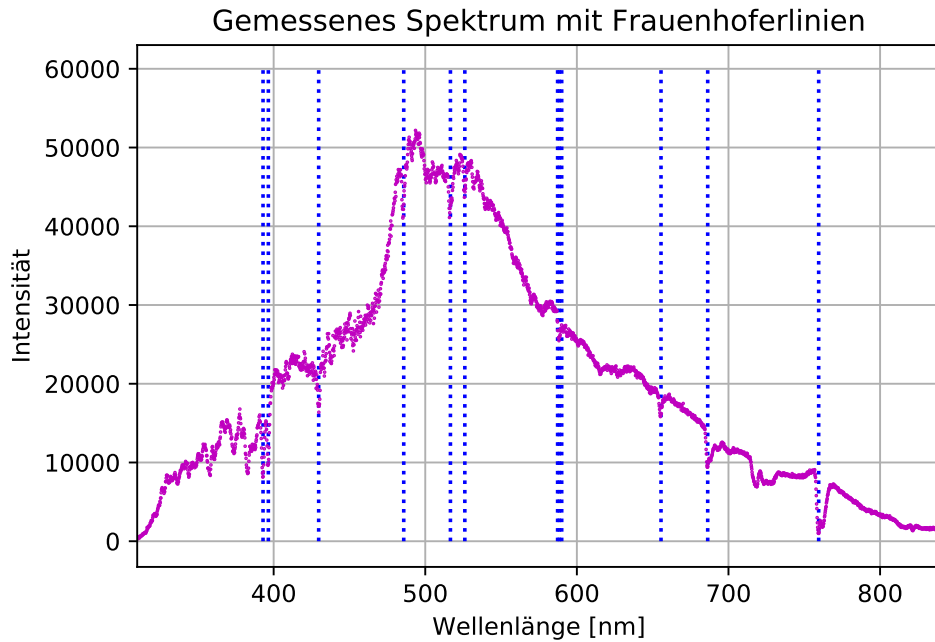


Abbildung 9: Gemessene Fraunhofsche Linien

Tabelle 2: Vergleich zwischen theoretischen und experimentellen Wellenlängen der Balmer serie

Symbol	Theoretische Wellenlänge λ [nm]	Gemessene Wellenlänge λ [nm]	Fehler Gemessene Wellenlänge $\Delta\lambda$ [nm]	Δ Wellenlänge λ [nm]	σ Abweichung [σ]	Farbe
H_α	656.3	656.7	0.6	0.4	0.7	Rot
H_β	486.1	486.3	0.6	0.2	0.3	Blau-Grün
H_γ	434.0	430.7	0.6	3.3	5.5	Violett
H_δ	410.1	410.3	0.6	0.2	0.3	Violett

starken Linien zu sehen, während Abbildung 12 und Abbildung 13 die schwächeren Linien im Wellenlängenbereich von 300nm bis 540nm bzw. 600nm bis 850nm zeigen.

In Tabelle 3 sind alle Wellenlängen eingetragen, wobei "links" der Spektrum zwischen 300nm und 540nm bezeichnet und "rechts" dieser zwischen 600nm und 850nm.

3.4 Zuordnung der gefundenen Linien zu Serien

In diesem Abschnitt ordnen wir die gemessene Linien den drei Natriumserien zu. Zunächst sollen wir die ungefähren Wellenlängen für jede Serie berechnen und danach mit den beobachteten Linien vergleichen.

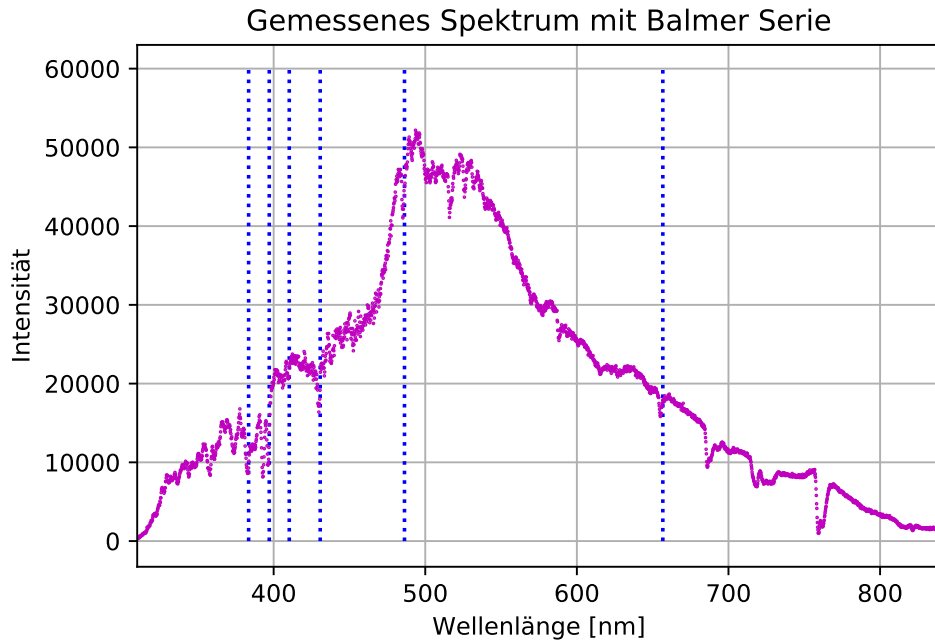


Abbildung 10: Gemessene Balmerlinien

3.4.1 Erwartete Linien für die 1. Nebenserie: $md \rightarrow 3p$

Wir können annehmen, dass der Korrekturterm für die d-Energieniveaus Δ_d gleich Null ist. Für die Wellenlänge λ_m der einzelnen Übergänge gilt dann:

$$\frac{hc}{\lambda_m} = \frac{E_{Ry}}{m^2} - E_{3p} \quad (6)$$

Und nach einer Umformung erhalten wir für die Wellenlänge:

$$\lambda_m = \frac{hc}{\frac{E_{Ry}}{m^2} - E_{3p}} \quad (7)$$

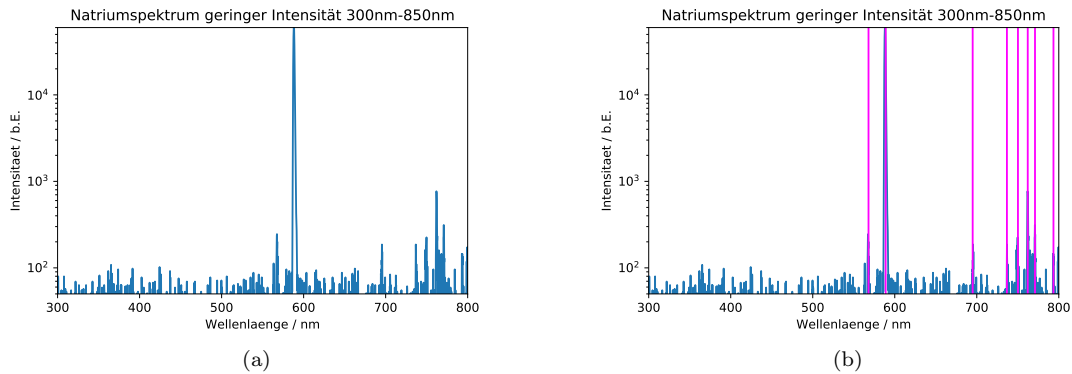
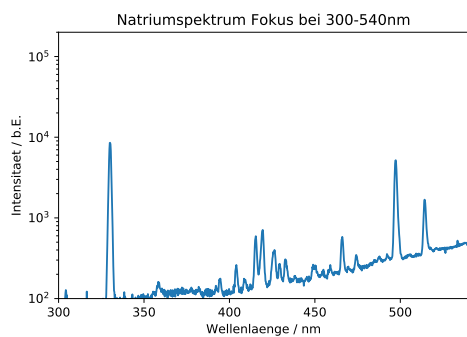
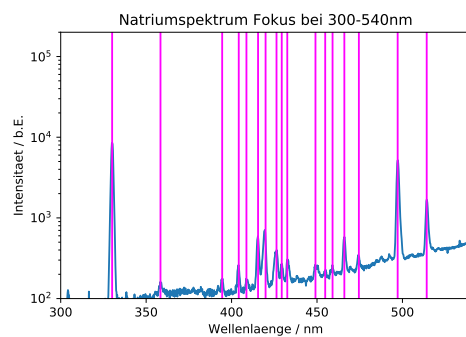


Abbildung 11: Natriumspektrum (links) mit gezeichnete Starke Linien (rechts)

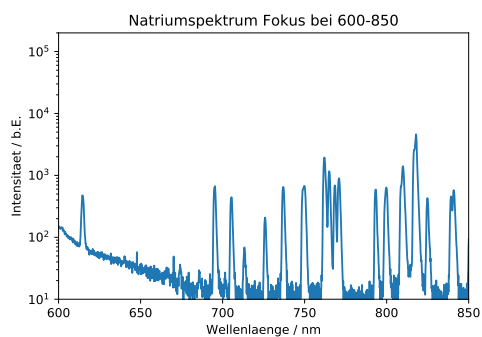


(a)

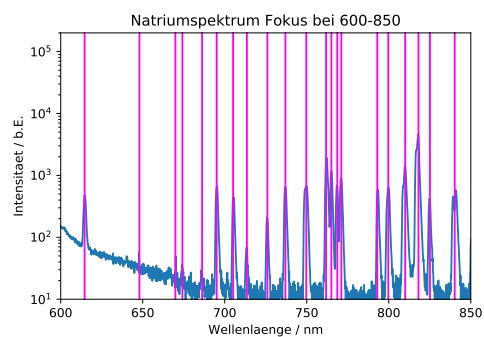


(b)

Abbildung 12: Natriumspektrum 300nm-540nm (links) mit gezeichnete Linien (rechts)



(a)



(b)

Abbildung 13: Natriumspektrum 600nm-850nm (links) mit gezeichnete Linien (rechts)

Tabelle 3: Wellenlängen Natrium			
Starke Linien [nm]	Linien links [nm]	Linien rechts [nm]	Fehler [nm]
568.0	330.1	614.6	0.6
588.9	358.4	648.0	0.6
695.0	394.5	669.9	0.6
736.8	404.2	674.2	0.6
750.2	408.7	686.2	0.6
762.1	415.5	695.1	0.6
771.0	419.9	705.1	0.6
793.5	426.3	713.5	0.6
	429.4	726.0	0.6
	432.6	737.0	0.6
	449.1	749.7	0.6
	454.8	761.8	0.6
	459.1	765.0	0.6
	466.0	768.6	0.6
	474.5	771.1	0.6
	497.2	793.0	0.6
	514.2	799.7	0.6
		810.0	0.6
		818.1	0.6
		825.1	0.6
		840.2	0.6

Nach der Gauß'schen Fehlerfortpflanzung berechnen wir den Fehler:

$$\Delta\lambda_m = \frac{hc}{\left(\frac{E_{Ry}}{m^2} - E_{3p}\right)^2} \Delta E_{3p} \quad (8)$$

Hier ist h die Planksche Wirkungsquantum, c die Lichtgeschwindigkeit, $E_{Ry} = -13,605\text{eV}$ die Rydbergenergie; m ist das Hauptquantenzahl des d -Niveaus. E_{3p} ist die Energie für den Zustand $md \rightarrow 3p$.

Die Wellenlänge, die wir für $m = 3$ gemessen haben beträgt $\lambda_{m=3} = (818,1 \pm 0,6)\text{nm}$. Die Energie lautet:

$$E_{3p} = \frac{E_{Ry}}{3^2} - \frac{hc}{\lambda_{m=3}} \quad (9)$$

Der fehler beträgt:

$$\Delta E_{3p} = \frac{hc}{(\lambda_{m=3})^2} \Delta\lambda_{m=3} \quad (10)$$

Die Energie kommt zu:

$$E_{3p} = -(3,0261 \pm 0,0011)\text{eV} \quad (11)$$

In Tabelle 4 habe ich alle erwarteten Werte mit den experimentell gemessenen verglichen und die Abweichungen berechnet. Für die Linie bei $\lambda_k = 440,48\text{nm}$ und $\lambda_k = 425,35$ habe ich keine passende Zuordnungen gefunden.

Die Ergebnisse besprechen wir ausführlich in der Diskussion.

3.4.2 Erwartete Linien für die 2. Nebenserie: $ms \rightarrow 3p$

Wir werden ausnutzen, dass die gelbe D-Linie des Natriums bei $\lambda_{gelb} = 589\text{nm}$ dem Übergang $ms \rightarrow 3p$ entspricht. Für die Bindungsenergie des grundzustands gilt:

$$E_{3s} = E_{3p} - \frac{hc}{\lambda} \quad (12)$$

Tabelle 4: Vergleich zwischen erwarteten und experimentellen Linien 1.Nebenserie

m	$\lambda_{erw}[nm]$	$\Delta\lambda_{erw}[nm]$	$\lambda_{exp}[nm]$	$\Delta\lambda_{exp}[nm]$	σ -Abweichung $[\sigma]$
3	818.64	0.6	818.1	0.6	0.6
4	569.52	0.29	568.0	0.6	2.3
5	499.31	0.22	497.2	0.6	3.3
6	467.97	0.2	466.0	0.6	3.1
7	450.90	0.18	449.1	0.6	2.9
8	440.48	0.17	x	0.6	x
9	433.61	0.17	432.6	0.6	1.6
10	428.82	0.16	426.3	0.6	4.1
11	425.35	0.16	x	0.6	x
12	422.74	0.16	419.9	0.6	4.6

Der Fehler betragt:

$$\Delta E_{3s} = \sqrt{\Delta E_{3p}^2 + \left(hc \frac{\Delta \lambda}{\lambda^2} \right)^2} \quad (13)$$

Es folgt, dass:

$$\Delta \lambda = 0 \Rightarrow \Delta E_{3s} = |\Delta E_{3p}| \quad (14)$$

Das Ergebnis der Bindungsenergie kommt zu:

$$E_{3s} = -(5,1322 \pm 0,0011)eV \quad (15)$$

Der Korrekturfaktor Δ_s können wir mithilfe der folgenden Formel berechnen:

$$E_{3s} = \frac{E_{Ry}}{(3 - \Delta_s)^2} \quad (16)$$

Nach einer Umformung betragt den korrekturfaktor:

$$\Delta_s = 3 - \sqrt{\frac{E_{Ry}}{E_{3s}}} \quad (17)$$

Nach der Gauß'schen Fehlerfortpflanzung erhalten wir fur den Fehler vom Korrekturfaktor:

$$\Delta \Delta_s = \frac{1}{2} \sqrt{\frac{E_{Ry}}{E_{3s}^3} \Delta E_{3s}^2} \quad (18)$$

Das Endergebnis des Korrekturfaktors kommt zu:

$$\Delta_s = (1,3714 \pm 0,00017)eV \quad (19)$$

Die Wellenlange berechnet man mit:

$$\lambda \approx \frac{hc}{\frac{E_{Ry}}{(m - \Delta_s)^2} - E_{3p}} \quad (20)$$

Und nach der Gauß'schen Fehlerfortpflanzung erhalten wir fur den Fehler:

$$\Delta \lambda_m = \sqrt{\left(\frac{hc}{\left(\frac{E_{Ry}}{(m - \Delta_s)^2} - E_{3p} \right)^2} \Delta E_{3p} \right)^2 + \left(\frac{2hcE_{Ry}(m - \Delta_s)}{(E_{Ry} - E_{3p}(m - \Delta_s)^2)^2} \Delta \Delta_s \right)^2} \quad (21)$$

In Tabelle 5 sind alle erwarteten Werte eingetragen und zu den gemessenen geordnet. Mit x habe ich die Linien mit keine passende Zuordnung gezeichnet.

Wir sehen, dass die Abweichungen ziemlich gro sind, was zu einem systematischen Fehler weist. Die Ergebnisse besprechen wir in der Diskussion.

Tabelle 5: Vergleich zwischen erwarteten und experimentellen Linien 2. Nebenserie

m	$\lambda_{erw,ns}[nm]$	$\Delta\lambda_{erw,ns}[nm]$	$\lambda_{exp,ns}[nm]$	$\Delta\lambda_{exp,ns}[nm]$	σ -Abweichung $[\sigma]$
4	1172.32	0.44	x	0.6	x
5	621.86	0.15	614.6	0.6	11.8
6	518.86	0.12	514.2	0.6	7.6
7	477.26	0.1	474.5	0.6	4.5
8	456.22	0.1	454.8	0.6	2.3
9	443.83	0.09	x	0.6	x

3.4.3 Erwartete Linien für die Hauptserie: $mp \rightarrow 3s$

Für den Korrekturfaktor Δ_p haben wir den Formel:

$$\Delta_p = 3 - \sqrt{\frac{E_{Ry}}{E_{3p}}} \quad (22)$$

Und den Fehler:

$$\Delta\Delta_p = \frac{1}{2} \sqrt{\frac{E_{Ry}}{E_{3p}^3}} \Delta E_{3p} \quad (23)$$

Das Endergebnis für den Korrekturfaktor ergibt sich auf:

$$\Delta_p = (0,8800 \pm 0,0004)eV \quad (24)$$

Der Vergleich zwischen den erwarteten und den experimentellen Linienwellenlängen der Hauptserie liegt in Tabelle 6. Da wir nur im Bereich zwischen $300nm$ und $850nm$ untersucht haben, wurde es nicht möglich zu der erwarteten Linie bei $286,28nm$ eine experimentelle Linie zuzuordnen.

Tabelle 6: Hauptserievergleich zwischen erwarteten und experimentellen Werte

m	$\lambda_{erw,hs}[nm]$	$\Delta\lambda_{erw,hs}[nm]$	$\lambda_{exp,hs}[nm]$	$\Delta\lambda_{exp}[nm]$	σ -Abweichung $[\sigma]$
4	331.98	0.23	330.1	0.6	2.9
5	286.28	0.29	x	0.6	x

3.5 Bestimmung der Serienenergien und der l-abhängigen Korrekturfaktoren

3.5.1 1. Nebenserie

In diesem Abschnitt verwenden wir die experimentell gemessenen Wellenlängen der Serien um die Rydbergenergie E_{Ry} , E_{3p} sowie die Korrekturterme Δ_d und Δ_s zu bestimmen.

Wir haben für die erste Nebenserie die folgende Funktion gewählt:

$$\lambda_m[nm] \approx \frac{hc}{\frac{E_{Ry}}{(m-\Delta_d)^2} - E_{3p}} \quad (25)$$

Die Funktion für die Wellenlänge in Abhängigkeit von der Quantenzahl für die erste Nebenserie des Natriumatoms ist in Abbildung 14 dargestellt.

In Tabelle 7 sind die zusammengefassten Energien und Korrekturfaktor für die erste Nebenserie sowie die Abweichungen zwischen den erwarteten Werten dargestellt.

Um die Güte des Fits diskutieren zu können, haben wir die χ^2 -Summe mit dieser Formel berechnet:

$$\chi^2 = \sum_i^N \left(\frac{Funktionswert_i - Messwert_i}{Fehler_i} \right)^2 \quad (26)$$

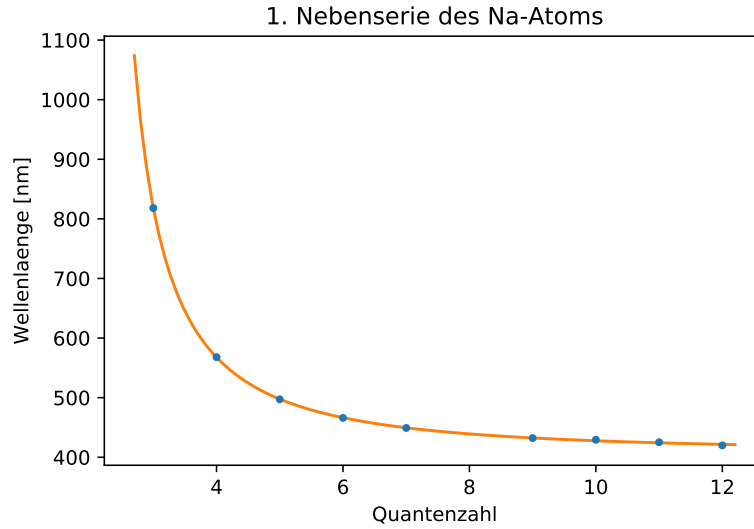


Abbildung 14: 1. Nebenserie des Na-Atoms

Tabelle 7: 1. Nebenserie zusammenfasste Ergebnisse

	Exp. Wert [eV]	Fehler [eV]	Erwartete Wert [eV]	Fehler[eV]	σ -Abweichung [σ]
E_{Ry}	-13.90	0.18	-13.605	-	1.7
E_{3p}	-3.0449	0.0028	-3.0272	0.0011	6.0
Δ_d	-0.015	0.017	0	-	0.9

Die χ^2_{red} -Wert berechnen wir mit diesem Formel:

$$\chi^2_{red} = \frac{\chi^2}{Freiheitsgrad} \quad (27)$$

Wobei der Freiheitsgrad sich aus der Anzahl der Messwerte abzüglich der Zahl der Fitparameter berechnet. Für die Güte des Fits der 1. Nebenserie erhalten wir:

$$\chi^2_1 = 3,35 \quad (28)$$

$$\chi^2_{red,1} = 0,67 \quad (29)$$

$$\sigma_{1.ns} = 65\% \quad (30)$$

Wobei $\sigma_{1.ns}$ die Fitwahrscheinlichkeit ist.

Die χ^2_{red} Wert sollte idealerweise bei 1 liegen, wenn unsere Dateien der Fit genau beschreiben. Da der Wert kleiner ist, deutet dies darauf hin, dass unsere Daten den Fit besser beschreiben als erwartet. Das könnte bedeuten, dass wir eigentlich zu große Fehlern bei den Wellenlängen von Spektrallinien abgeschätzt haben. Die Fitwahrscheinlichkeit sollte im besten Fall bei ungefähr 50% liegen. Wir haben ein Zusammenhang, der größer als 50% ist, was auch wahrscheinlich an der große relative Fehler liegt.

Die Ergebnisse besprechen wir ausführlicher in der Diskussion.

3.5.2 2. Nebenserie

Analog bestimmen wir die Energien und Korrekturfaktor und ihre Abweichungen von den erwarteten Werten für die zweite Nebenserie. Die Daten sind in Tabelle 8 dargestellt. Die Funktion der zweite Nebenserie ist in Abbildung 15 dargestellt. Anschließend berechnen wir die χ^2 , χ^2_{red} und Fitwahrscheinlichkeit.

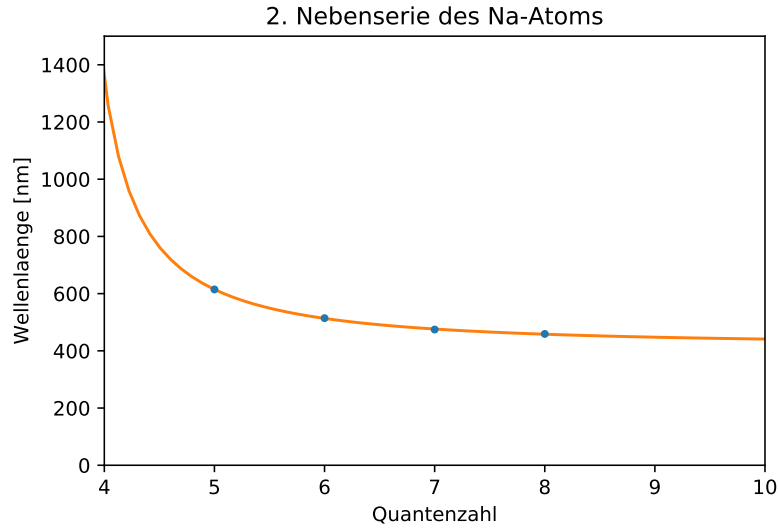


Abbildung 15: 2. Nebenserie des Na-Atoms

Tabelle 8: 2. Nebenserie zusammengefasste Ergebnisse

	Exp. Wert [eV]	Fehler [eV]	Erwartete Wert [eV]	Fehler[eV]	σ -Abweichung [σ]
E_{Ry}	-13.3	0.4	-13.605	-	0.8
E_{3p}	-3.018	0.007	-3.0272	0.0011	1.3
Δ_s	1.49	0.05	1.37184	0.00017	2.4

Für die Güte des Fits erhalten wir:

$$\chi^2_2 = 0,21 \quad (31)$$

$$\chi^2_{red,2} = 0,21 \quad (32)$$

$$\sigma_{2.ns} = 64\% \quad (33)$$

Die Ergebnisse besprechen wir in der Diskussion.

4 Zusammenfassung und Diskussion

In diesem Versuch haben wir zunächst die Lichtintensität verschiedener LED-Farben, einer Energiesparlampe, einer Glühlampe und eines Lasers ausgewertet. Die Daten sind in Tabelle 0 zusammengefasst. Dabei haben wir beobachtet, dass die Energiesparlampe und der Laser diskretes Licht in bestimmten Wellenlängen mit höheren Intensitäten emittieren und daher effizienter sind. Im Gegensatz dazu emittieren die weißen LEDs kontinuierliches Licht mit zwei Maxima bei $450nm$ und $520nm$. Die Glühlampe ähnelt dem Sonnenlicht am meisten, ist jedoch auch die ineffizienteste, da sie das gesamte sichtbare Spektrum emittiert.

Es wurde festgestellt, dass Weißlicht verschiedene Intensitäten von verschiedenen Wellenlängen umfassen kann, was dazu führt, dass es entweder als weiß oder als warm eingestuft wird, wie bei den beiden weißen LEDs beobachtet wurde.

Anschließend haben wir das Sonnenspektrum untersucht und festgestellt, dass die Intensität des direkten Lichts höher ist als die des Lichts, das durch ein Fenster durchgelaufen ist. In Abbildung 8 haben wir das Absorptionsspektrum von Glas dargestellt und erkannt, dass Glas sichtbares Licht schwächer absorbiert, dafür aber mehr Infrarot (IR) und Ultraviolettlicht (UV). Dies entspricht unseren Erwartungen, da wir durch die Fenster sehen können und sie uns vor den schädlichen Auswirkungen von IR- und UV-Licht schützen, ähnlich wie die Ozonschicht. Im gleichen Abschnitt haben wir auch die Wellenlängen der Fraunhoferlinien bestimmt und mit den theoretischen Werten verglichen. Mehr als die Hälfte der Abweichungen zwischen den theoretischen und experimentellen Werten liegen im

1σ -Bereich, etwa ein Drittel im 2σ -Bereich, und ein Ergebnis liegt genau bei 3σ . Dies sind im Allgemeinen präzise Ergebnisse, insbesondere wenn man die vielen Fehlerquellen berücksichtigt.

Zum einen gibt es Stoffe und Moleküle in der Atmosphäre, die die Ergebnisse teilweise verändern und das Bild verwascht machen können. Eine weitere Fehlerquelle besteht darin, dass wir nur ein Bild aufgenommen haben und nicht jedes kleine, gezoomte Intervall einzeln sorgfältig untersucht haben. Dies könnte zu systematischen Fehlern führen, da menschliche Fehler bei der Identifizierung der Spektrallinien auftreten können. Auch das Gitterspektrometer hat eine Messgerätsfehler.

Wir haben auch die Linien der Balmer-Serie bestimmt, in einem Diagramm dargestellt und mit den theoretischen Werten verglichen. Alle Abweichungen außer einer liegen unter 1σ . Die violette H_γ -Linie liegt sehr nahe an der Eisen- und Calcium-Fraunhoferlinie, und deshalb konnte ich die wahrscheinlich nicht identifizieren und daher die große Abweichung.

Im zweiten Teil des Versuchs haben wir uns mit dem Natriumspektrum beschäftigt. Zunächst haben wir unsere drei Messungen analysiert und die Wellenlängen der Spektrallinien ermittelt. Die entsprechenden Diagramme sind in den Abbildungen 11, 12 und 13 dargestellt, während die Daten in Tabelle 2 dargestellt sind.

Wir haben angenommen, dass der Korrekturterm für die d-Energieniveaus, Δ_d , gleich null ist. Mithilfe von Formel 9 haben wir die Energie E_{3p} berechnet, die sich auf $-(3,0261 \pm 0,0011)$ eV beläuft. Durch die Anwendung der in Anhang gegebenen Wellenlänge $\lambda_{m=3} = (818,1 \pm 0,6)$ nm und Formel 7 haben wir die erwarteten Wellenlängen für die Nebenserie bis $m = 12$ berechnet. Diese spektrale Linien haben wir dann in Tabelle 4 zugeordnet. Zwei von den Linien konnte ich nicht zuordnen, deshalb sind sie mit einem 'x' markiert.

Die gemessene $\lambda_{m=3}$ weicht um $1,5\sigma$ von dem in der Praktikumsanleitung angegebenen Wert ab, deshalb bin ich unsicher, ob ich die richtige Spektrallinie identifiziert habe. Alle Abweichungen außer dieser von $\lambda_{m=3}$ liegen im Bereich von $1,6\sigma$ bis $4,6\sigma$. Das sind sehr große Abweichungen, die zu einer systematischen Fehler hinweisen. Im Abschnitt 3.5.1 haben wir die χ^2_1 und $\chi^2_{1,red}$ unserer gemessenen Werte bestimmt. Wie schon in der Auswertung erwähnt, unsere Werte passen nicht so schlecht zu der Fitgeraden, was möglicherweise auf Fehler bei der Bestimmung der erwarteten Wellenlängen oder auf eine falsche Identifizierung der $\lambda_{m=3}$ -Linie hindeutet.

Wir haben die gemessenen Wellenlängen dieser Serie verwendet, um die Rydberg-Energie E_{Ry} , die Bindungsenergie E_{3p} und den Korrekturfaktor Δ_d zu bestimmen. Die Ergebnisse sind in Tabelle 7 dargestellt und mit den erwarteten Werten verglichen. Die Abweichung der Rydberg-Energie beträgt $1,7\sigma$ mit einem relativen Fehler von $1,3\%$. Obwohl das eine signifikante Abweichung ist, bleibt der relative Fehler moderat. Für die Energie E_{3p} haben wir eine Abweichung von 6σ und einen relativen Fehler von $0,09\%$. Der Korrekturfaktor Δ_d weicht um weniger als 1σ ab. Da wir in diesem Teil des Experiments keine systematischen Fehler berücksichtigt haben und nur statistische Fehler vorliegen, sind die kleinen relativen Fehler unrealistisch und daher die große Abweichung. Der Korrekturfaktor ist eine Näherung und hängt von der Quantenzahl ab, deshalb war eine gewisse Abweichung schon erwartet.

Analog dazu haben wir dies für die zweite Nebenserie $m_s \rightarrow 3p$ durchgeführt. Hierbei hatten wir eine geringere Anzahl von Linien, und unsere χ^2_2 und $\chi^2_{red,2}$ Summen liegen näher an Null, mit Werten von $\chi^2_2 = \chi^2_{red,2} = 0,21$. Das bedeutet, dass unsere Ergebnisse den Fit sehr genau beschreiben. Das ist größtenteils darauf zurückzuführen, dass ich für zwei Linien keine gemessenen Wellenlängen zuordnen konnte, und somit nur vier Linien verblieben sind. Die σ -Abweichungen zwischen den gemessenen und den erwarteten Wellenlängen liegen alle im 2σ -Bereich, höchstwahrscheinlich aufgrund der zuvor diskutierten Gründe.

Die Energien E_{Ry} und E_{3p} wurden in Tabelle 8 dargestellt und mit den erwarteten Werten verglichen. Hier haben wir größere relative Fehler und dabei kleinere σ -Abweichungen. Bei kleinen Stichproben, wie in diesem Experiment, mit nur vier bis neun Messungen, kann der statistische Fehler sowohl sehr klein als auch sehr groß sein, und man kann nicht ausschließlich auf diesen verlassen. Die oben genannten Messgerät-, systematische- und weitere Fehler sollen ebenfalls zu dem statistischen Fehler hinzugefügt werden, um ein umfassenderes Bild zu zeichnen. In diesem Fall war der statistische Fehler wahrscheinlich näher am Gesamtfehler, jedoch können wir aus nur vier Messungen keine endgültigen Schlussfolgerungen ziehen. Für den Korrekturfaktor Δ_s haben wir eine Abweichung von $2,4\sigma$ und einen relativen Fehler von $3,3\%$.

Für die Hauptserie haben wir den Korrekturfaktor Δ_p gemessen und die Wellenlänge der Serienlinien mit den theoretischen Werten verglichen. Da wir nur den Bereich zwischen 300nm und 850nm untersucht haben, konnten wir nur einer Wellenlänge zuordnen. Diese weist eine Abweichung von $2,9\sigma$ auf, was mit den zuvor besprochenen

größeren Abweichungen übereinstimmt.

In diesem Versuch traten zahlreiche Fehlerquellen und Näherungen auf, darunter wahrscheinlich auch systematische Fehler, deshalb haben wir für den Natrium Atom weniger präzisen Ergebnisse, außer den χ^2 Summen und den Fit-wahrscheinlichkeiten, die besser als erwartet waren. Um genauere Ergebnisse zu erzielen, bedürfte es möglicherweise umfassender Änderungen in Versuchsdurchführung. Zum Beispiel könnte eine bessere Methode zur Identifizierung der Balmer- und Fraunhoferlinien verwendet werden, um die relativen Fehler zu verringern. Es wäre auch ratsam, in einer komplett dunklen Umgebung zu arbeiten, um Störungen durch Rauschen zu vermeiden. Weniger Näherungen zu verwenden würde auch die Ergebnisse genauer machen. Jedoch liegen die Ergebnisse im Allgemeinen nicht weit von der Realität entfernt.

Der erste Teil des Versuchs, insbesondere die Auswertung des Sonnenspektrums, verlief erfolgreicher, mit nur einem Fehler bei der violetten Balmerreihe.

Trotz der nicht optimalen Ergebnisse hat mir dieser Versuch gefallen, da ich viel über das Thema gelernt habe. Die Arbeit mit χ^2 -Summen war für mich eine neue Erfahrung und sehr informativ.

5 Anhang

5.1 Quellen

Alle Informationen, die ich im Protokoll verwendet habe, stammen aus der Praktikumsanleitung, Ausgabe 4.2023.

5.2 Python-Code

Der Python-Code befindet sich auf der nächsten Seite.

Untitled

March 17, 2024

```
[1]: import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
import scipy.integrate as integrate
from scipy.stats import chi2
from uncertainties import ufloat
import uncertainties.unumpy as unp
from IPython.display import Image
from colorama import Back, Style
```

```
[2]: def comma_to_float(valstr):
    return float(valstr.decode("utf-8").replace(',', '.'))
```

```
[3]: #LED Leiste
```

```
[4]: #Blau
lamb_bl, inten_bl=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A2/LEDblau.txt',␣
    ↪skiprows=17,
converters= {0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
```

```
[5]: #Gelb
lamb_ge, inten_ge=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A2/LEDgelb.txt',␣
    ↪skiprows=17,
converters= {0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
```

```
[6]: #Orange
lamb_or, inten_or=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A2/LEDorange.txt',␣
    ↪skiprows=17,
converters= {0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
```

```
[7]: #Rot
lamb_rot, inten_rot=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A2/LEDrot.txt',␣
    ↪skiprows=17,
```

```
converters= {0:comma_to_float, 1:comma_to_float}, comments='>', unpack=True)
```

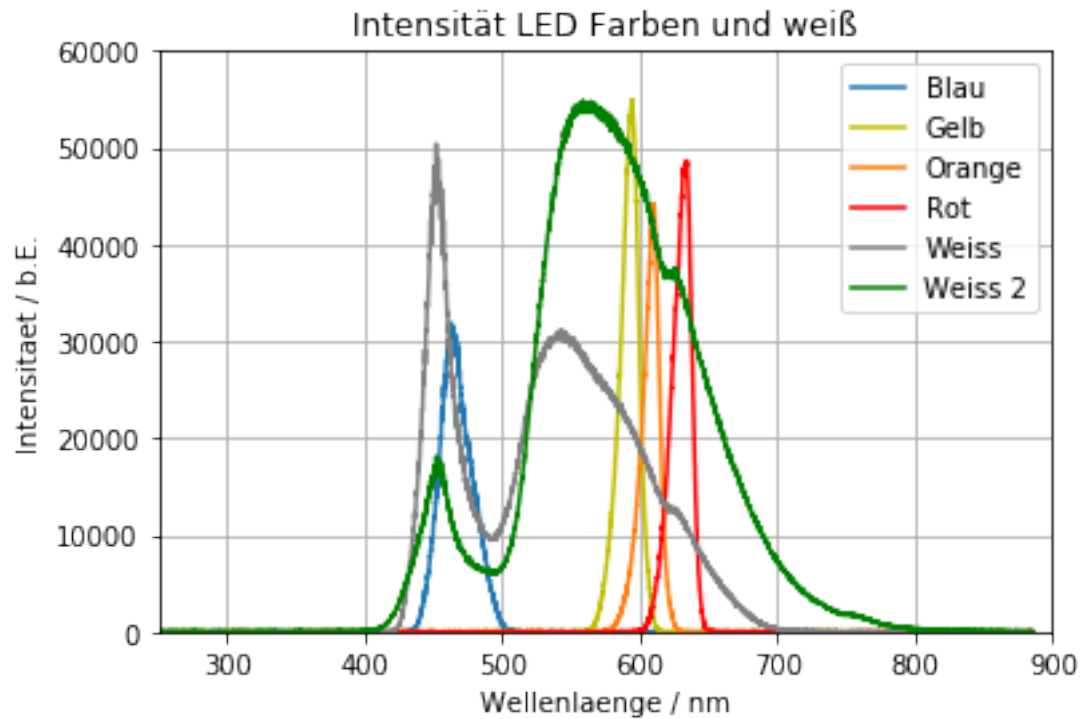
```
[8]: #Weiss
```

```
lamb_weiss, inten_weiss=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A2/LEDweiss.  
↳txt', skiprows=17,  
converters= {0:comma_to_float, 1:comma_to_float}, comments='>', unpack=True)
```

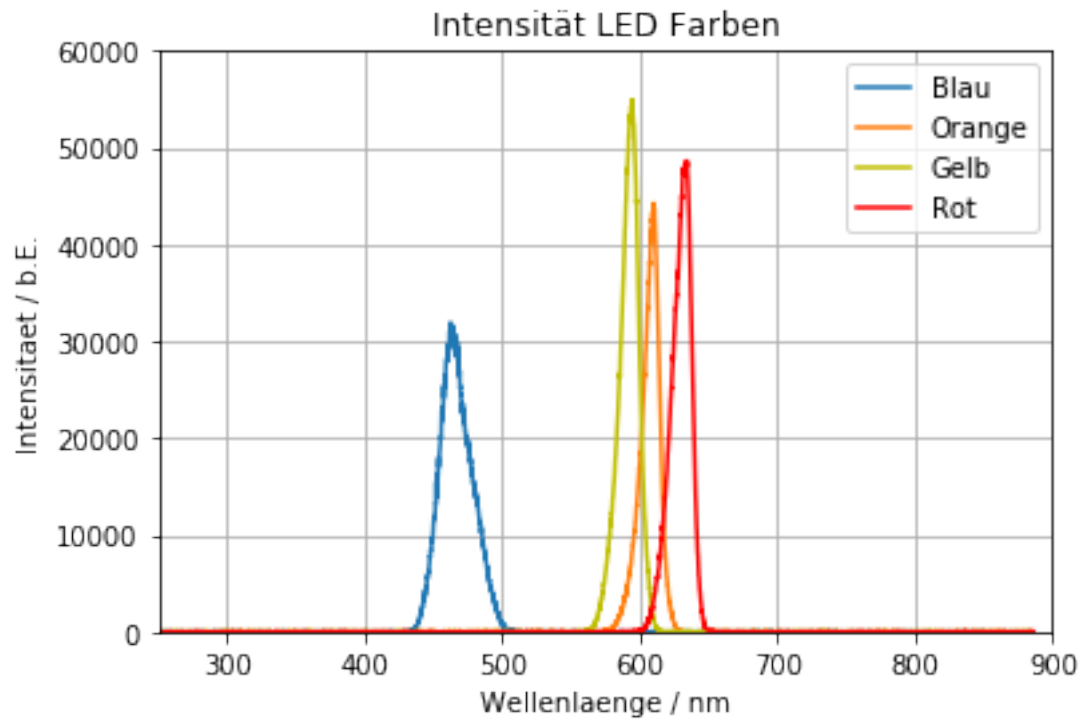
```
[9]: #Weiss2
```

```
lamb_weiss2, inten_weiss2=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A2/  
↳LEDweiss2.txt', skiprows=17,  
converters= {0:comma_to_float, 1:comma_to_float}, comments='>', unpack=True)
```

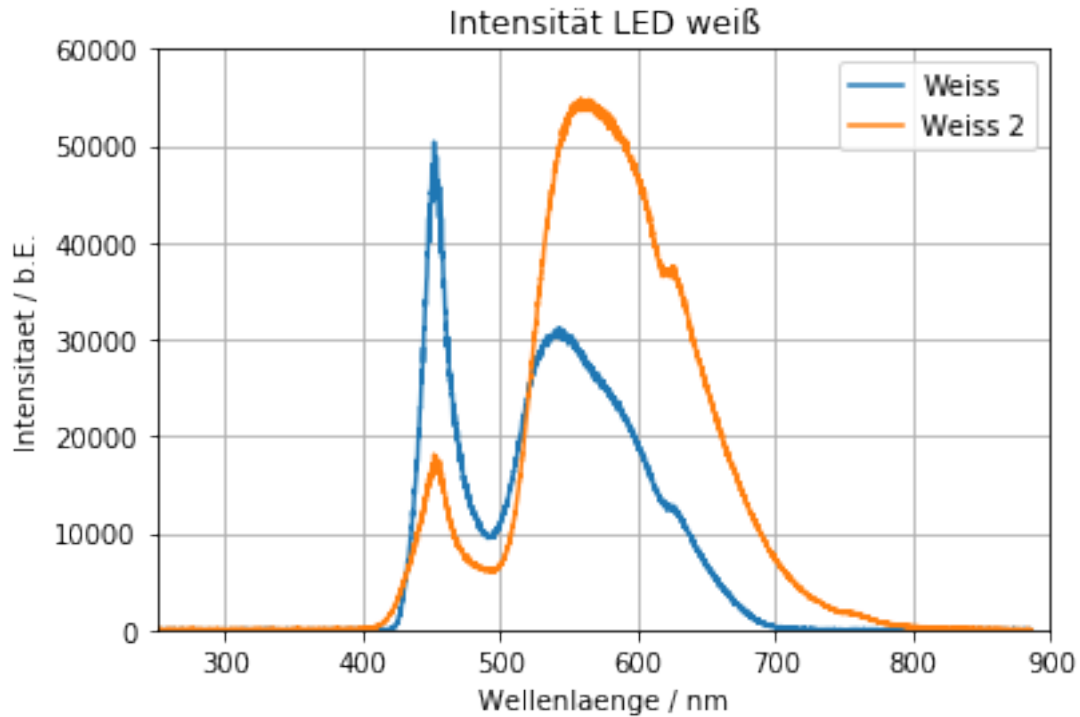
```
[10]: plt.plot(lamb_bl, inten_bl, label='Blau')  
plt.plot(lamb_ge, inten_ge, label='Gelb', color='y')  
plt.plot(lamb_or, inten_or, label='Orange')  
plt.plot(lamb_rot, inten_rot, label='Rot', color="r")  
plt.plot(lamb_weiss, inten_weiss, label='Weiss', color="gray")  
plt.plot(lamb_weiss2, inten_weiss2, label='Weiss 2', color="green")  
  
plt.title('Intensität LED Farben und weiß')  
plt.xlabel('Wellenlaenge / nm')  
plt.ylabel('Intensitaet / b.E.')  
plt.legend()  
plt.grid()  
plt.ylim((0,60000))  
plt.xlim((250,900))  
plt.savefig("LEDalle.pdf", format="pdf")
```



```
[11]: plt.plot(lamb_bl, inten_bl, label='Blau')
plt.plot(lamb_or, inten_or, label='Orange')
plt.plot(lamb_ge, inten_ge, label='Gelb', color='y')
plt.plot(lamb_rot, inten_rot, label='Rot', color="r")
plt.title('Intensität LED Farben')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Intensitaet / b.E.')
plt.legend()
plt.grid()
plt.ylim((0,60000))
plt.xlim((250,900))
plt.savefig("LED.pdf", format="pdf")
```



```
[12]: plt.plot(lamb_weiss, inten_weiss, label='Weiss')
plt.plot(lamb_weiss2, inten_weiss2, label='Weiss 2')
plt.title('Intensität LED weiß')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Intensitaet / b.E.')
plt.legend()
plt.grid()
plt.ylim((0,60000))
plt.xlim((250,900))
plt.savefig("LEDweiss.pdf", format="pdf")
```



```
[13]: #Lampen
```

```
[14]: #Energiesparlampe
lamb_en, inten_en=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A2/
↳Energiesparlampe.txt', skiprows=17,
converters={0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
```

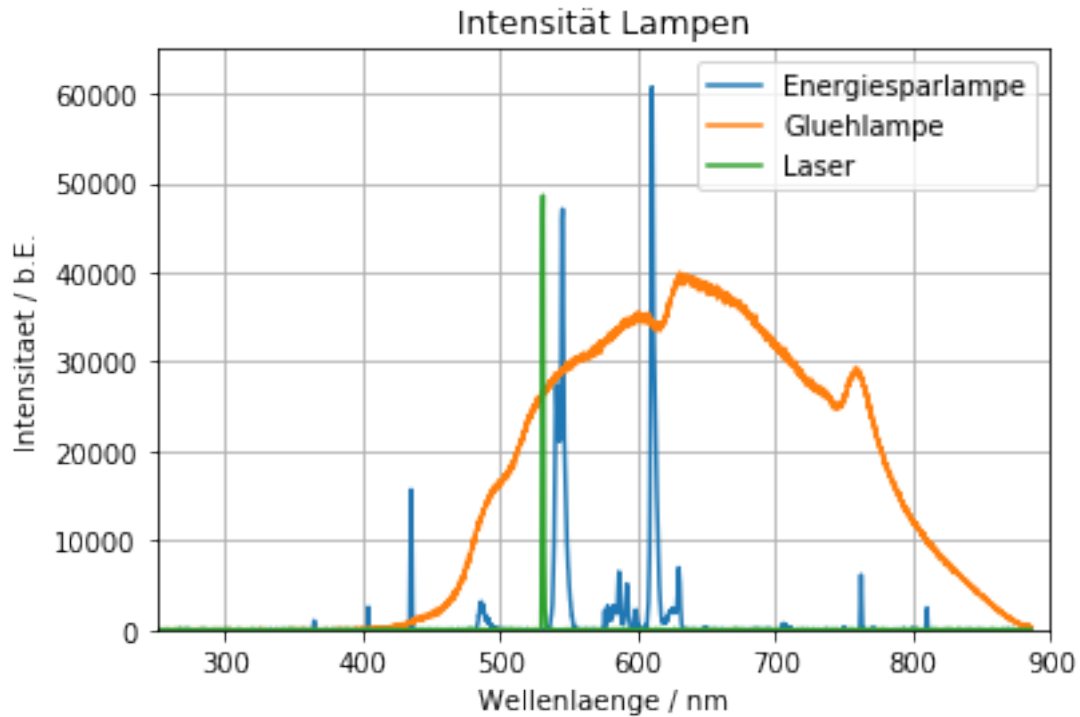
```
[15]: #Gluehlampe
lamb_glu, inten_glu=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A2/Gluehlampe.
↳txt', skiprows=17,
converters={0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
```

```
[16]: #Laser
lamb_las, inten_las=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A2/LASER.txt',
↳skiprows=17,
converters={0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
```

```
[17]: plt.plot(lamb_en, inten_en, label='Energiesparlampe')
plt.plot(lamb_glu, inten_glu, label='Gluehlampe')
plt.plot(lamb_las, inten_las, label='Laser')
```



```
plt.title('Intensität Lampen')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Intensitaet / b.E.')
plt.legend()
plt.grid()
plt.ylim((0,65000))
plt.xlim((250,900))
plt.savefig("Lampen.pdf", format="pdf")
```

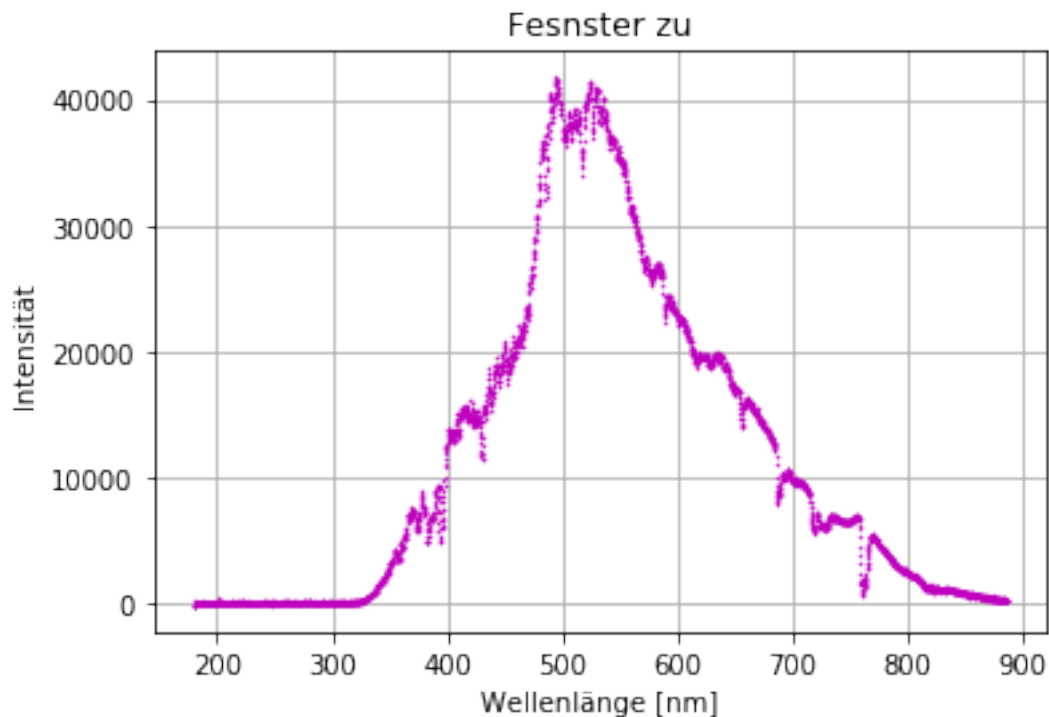


```
[18]: #Sonnenspektrum
def comma_to_float(valstr):
    return float(valstr.decode("utf-8").replace(',', '.'))

x = np.array([np.loadtxt("/home/urz/fphys/eh301/Pap2/234/A11/mitfenster.txt",
    ↳ skiprows = 14, converters= {0:comma_to_float, 1:comma_to_float}, usecols =
    ↳ (0))])
y = np.array([np.loadtxt("/home/urz/fphys/eh301/Pap2/234/A11/mitfenster.txt",
    ↳ skiprows = 14, converters= {0:comma_to_float, 1:comma_to_float}, usecols =
    ↳ (1))])

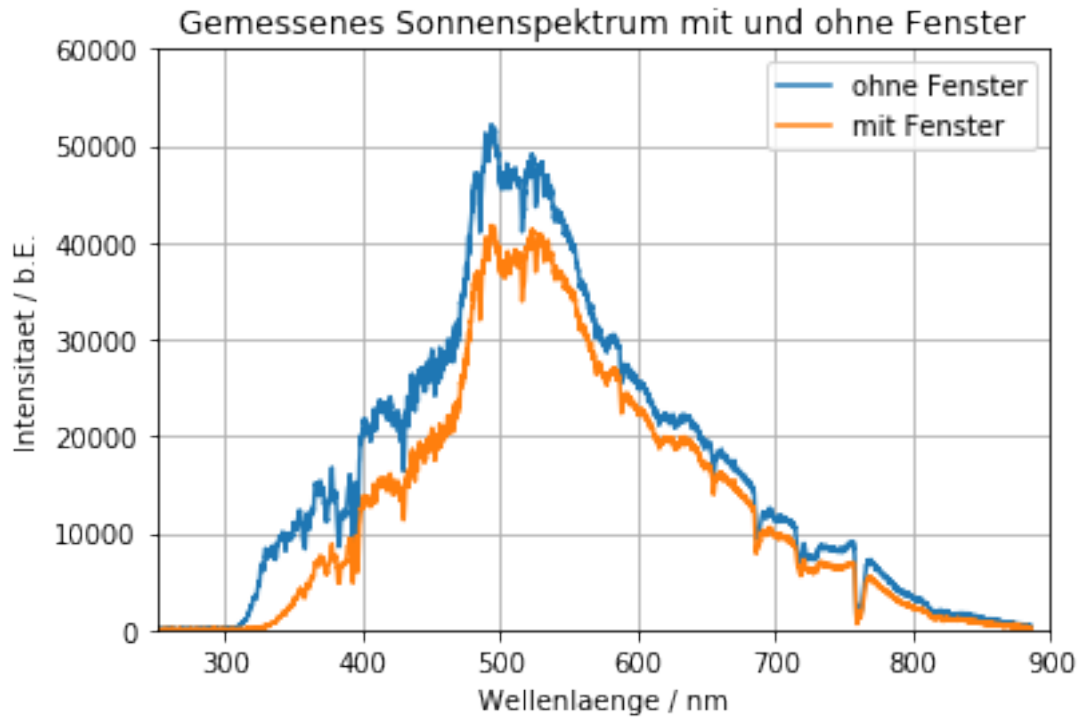
plt.plot(x, y, "m.", ms = 1)
plt.title(r"Fesnster zu")
plt.xlabel(r"Wellenlänge [nm]")
```

```
plt.ylabel(r"Intensität")
plt.grid()
```



```
[19]: def comma_to_float(valstr):
        return float(valstr.decode("utf-8").replace(',','.'))
```

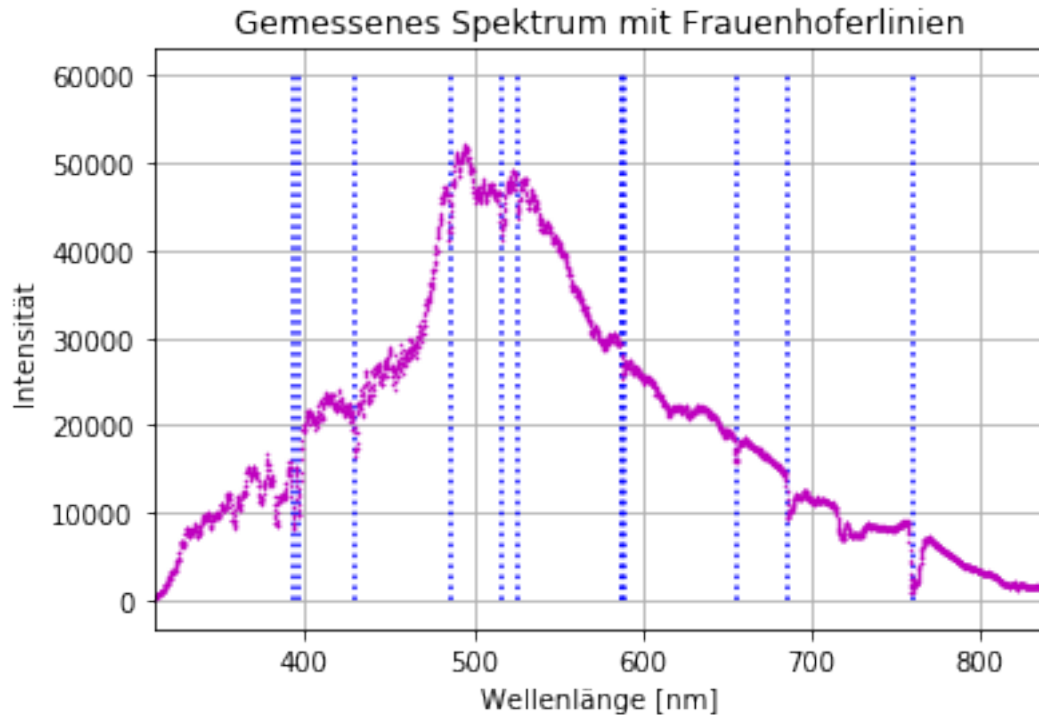
```
[20]: #Sonnenspektrum mit offenen und geschlossenen Fenster
lamb_og, inten_og=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A11/ohnefenster.
↳txt', skiprows=17, converters= {0:comma_to_float, 1:comma_to_float},
↳comments='>', unpack=True)
lamb_mg, inten_mg=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/A11/mitfenster.
↳txt', skiprows=17, converters= {0:comma_to_float, 1:comma_to_float},
↳comments='>', unpack=True)
plt.plot(lamb_og, inten_og, label='ohne Fenster')
plt.plot(lamb_mg, inten_mg, label='mit Fenster')
plt.title('Gemessenes Sonnenspektrum mit und ohne Fenster')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Intensitaet / b.E.')
plt.legend()
plt.grid()
plt.ylim((0,60000))
plt.xlim((250,900))
plt.savefig("Himmel_m_o_G.pdf", format="pdf")
```



```
[21]: #Fraunhoferlinien
x = np.array([np.loadtxt("/home/urz/fphys/eh301/Pap2/234/A11/ohnefenster.txt",
    ↳ skiprows = 14, converters= {0:comma_to_float, 1:comma_to_float}, usecols =
    ↳ (0))])
y = np.array([np.loadtxt("/home/urz/fphys/eh301/Pap2/234/A11/ohnefenster.txt",
    ↳ skiprows = 14, converters= {0:comma_to_float, 1:comma_to_float}, usecols =
    ↳ (1))])

plt.plot(x, y, "m.", ms = 1)
plt.title(r"Gemessenes Spektrum mit Fraunhoferlinien")
plt.xlabel(r"Wellenlänge [nm]")
plt.ylabel(r"Intensität")
plt.xlim((310,840))
plt.vlines(393, 0, 60000, "b", "dotted", "K")
plt.vlines(396.5, 0, 60000, "b", "dotted", "H")
plt.vlines(429.7, 0, 60000, "b", "dotted", "G")
plt.vlines(485.8, 0, 60000, "b", "dotted", "F")
plt.vlines(516.6, 0, 60000, "b", "dotted", "b1")
plt.vlines(526.1, 0, 60000, "b", "dotted", "E")
plt.vlines(587.3, 0, 60000, "b", "dotted", "D3")
plt.vlines(588.4, 0, 60000, "b", "dotted", "D2")
plt.vlines(590, 0, 60000, "b", "dotted", "D1")
plt.vlines(655.5, 0, 60000, "b", "dotted", "C")
```

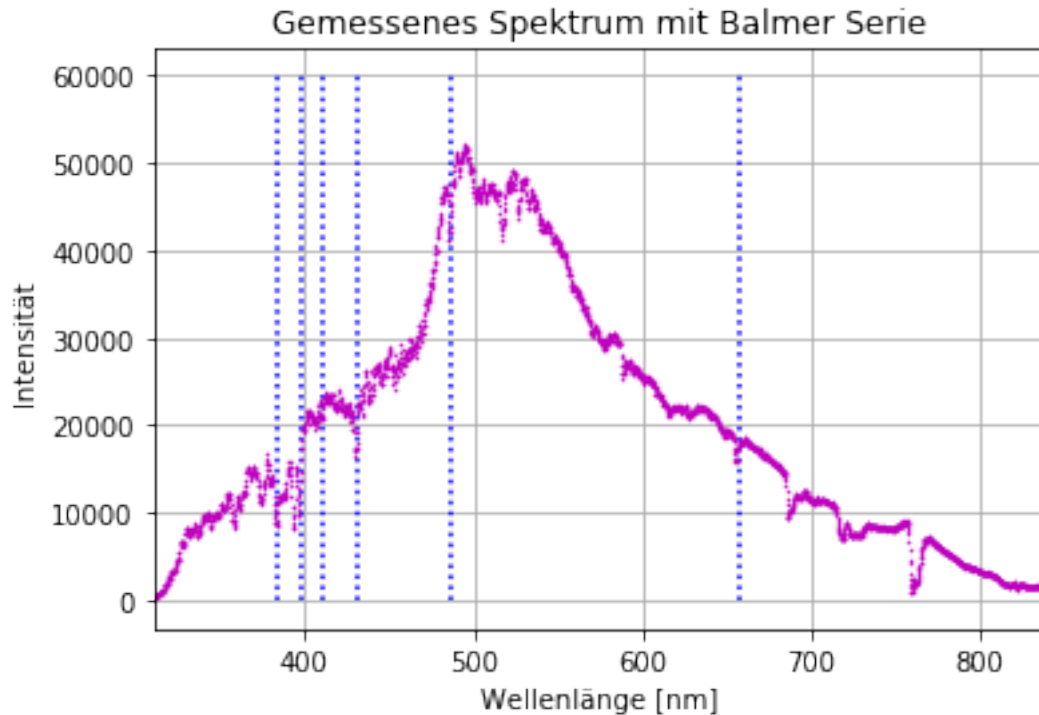
```
plt.vlines(686.3, 0, 60000, "b", "dotted", "B")
plt.vlines(759.5, 0, 60000, "b", "dotted", "A")
plt.grid()
plt.savefig("Frauenhof.pdf", format="pdf")
```



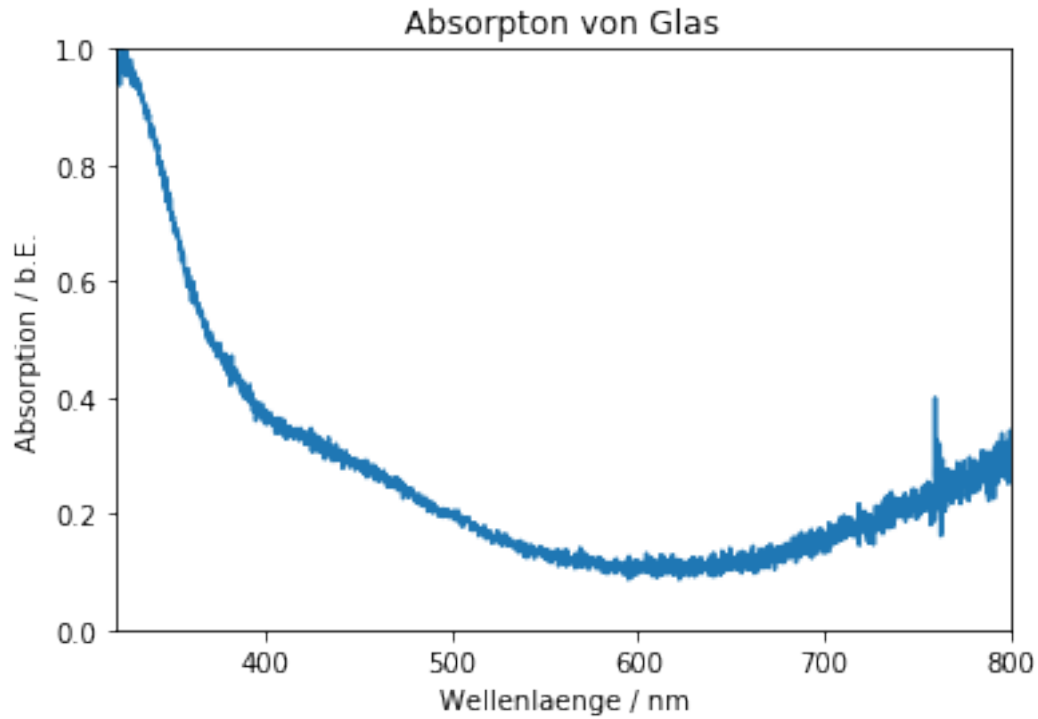
```
[22]: #Balmerserie
x = np.array([np.loadtxt("/home/urz/fphys/eh301/Pap2/234/A11/ohnefenster.txt",
    ↳skiprows = 14, converters= {0:comma_to_float, 1:comma_to_float}, usecols =
    ↳(0))])
y = np.array([np.loadtxt("/home/urz/fphys/eh301/Pap2/234/A11/ohnefenster.txt",
    ↳skiprows = 14, converters= {0:comma_to_float, 1:comma_to_float}, usecols =
    ↳(1))])

plt.plot(x, y, "m.", ms = 1)
plt.title(r"Gemessenes Spektrum mit Balmer Serie")
plt.xlabel(r"Wellenlänge [nm]")
plt.ylabel(r"Intensität")
plt.xlim((310,840))
plt.vlines(656.7, 0, 60000, "b", "dotted", "H_a")
plt.vlines(486.3, 0, 60000, "b", "dotted", "H_b")
plt.vlines(430.7, 0, 60000, "b", "dotted", "H_g")
plt.vlines(410.3, 0, 60000, "b", "dotted", "H_A")
plt.vlines(397.1, 0, 60000, "b", "dotted", "H_e")
```

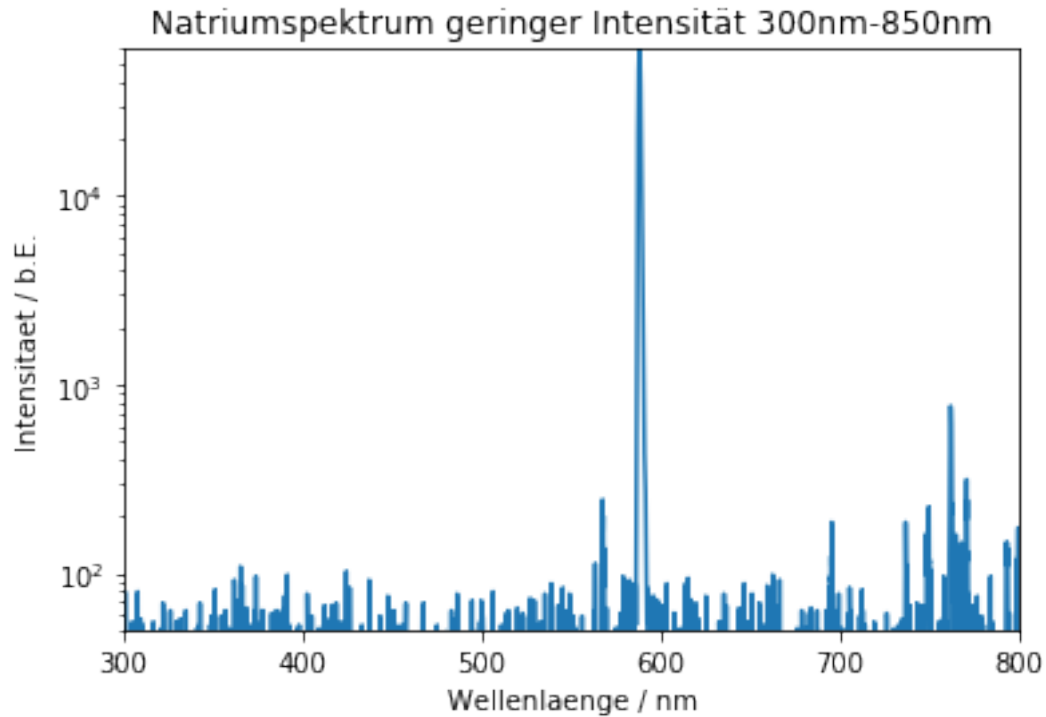
```
#plt.vlines(-, 0, 60000, "b", "dotted", "H_ps")
plt.vlines(383.5, 0, 60000, "b", "dotted", "H_eta")
plt.grid()
plt.savefig("Balmer.pdf", format="pdf")
```



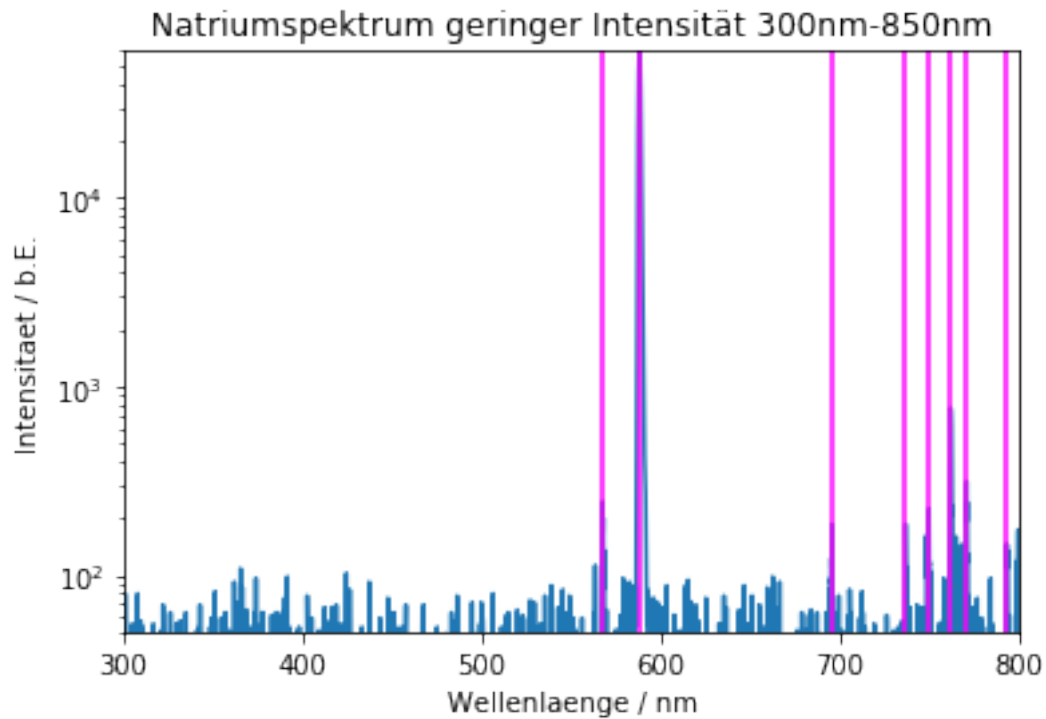
```
[23]: #Glasabsorption
A=1-inten_mg/inten_og
plt.plot(lamb_mg, A)
plt.title('Absorption von Glas')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Absorption / b.E.')
plt.ylim((0,1))
plt.xlim((320,800))
plt.savefig("Absorption_Glas.pdf", format="pdf")
```



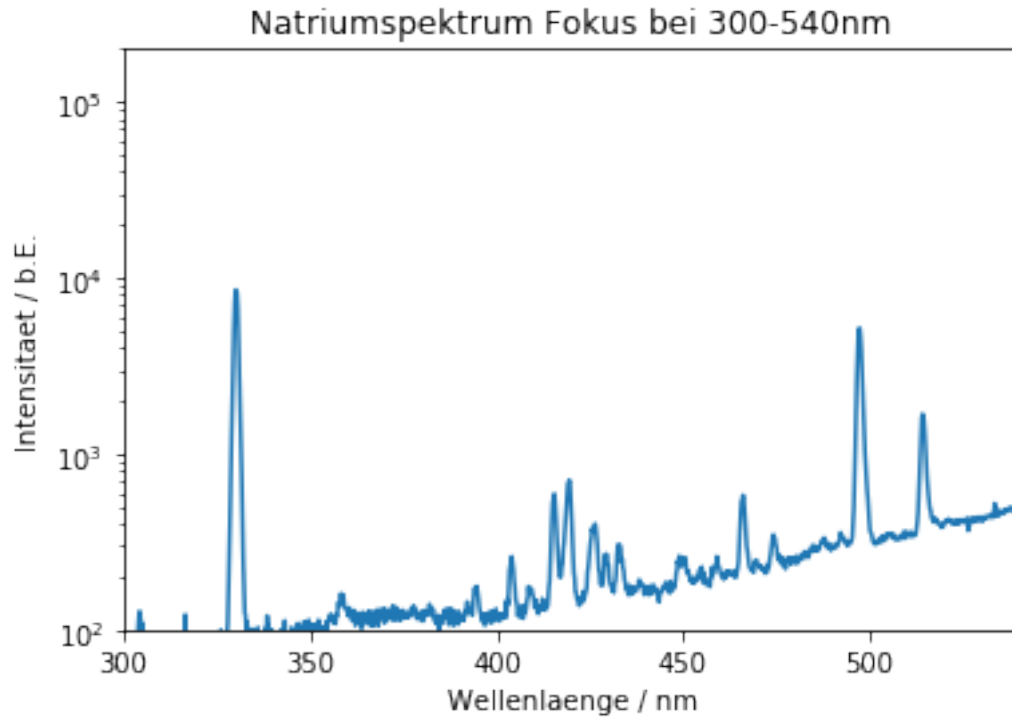
```
[24]: #Natriumspektrum geringer Intensität 300nm-850nm
lamb_og, inten_og=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/Natrium/
↳5mserstebild.txt', skiprows=17,
converters= {0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
plt.plot(lamb_og, inten_og)
plt.title('Natriumspektrum geringer Intensität 300nm-850nm')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Intensitaet / b.E.')
plt.yscale('log')
plt.ylim((50,60000))
plt.xlim((300,800))
plt.savefig("natriumkeinzoom.pdf", format="pdf")
```



```
[25]: #Linien stellen
lamb_og, inten_og=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/Natrium/
↳5mserstebild.txt', skiprows=17,
converters={0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
plt.plot(lamb_og, inten_og)
plt.title('Natriumspektrum geringer Intensität 300nm-850nm')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Intensitaet / b.E.')
plt.yscale('log')
plt.ylim((50,60000))
plt.xlim((300,800))
plt.axvline(x=568.0, ymin=0, ymax=1, color='magenta')
plt.axvline(x=588.9, ymin=0, ymax=1, color='magenta')
plt.axvline(x=695.0, ymin=0, ymax=1, color='magenta')
plt.axvline(x=736.76, ymin=0, ymax=1, color='magenta')
plt.axvline(x=750.2, ymin=0, ymax=1, color='magenta')
plt.axvline(x=762.148, ymin=0, ymax=1, color='magenta')
plt.axvline(x=771.0, ymin=0, ymax=1, color='magenta') #neu
plt.axvline(x=793.51, ymin=0, ymax=1, color='magenta') #neu
plt.savefig("liniennatrium.pdf", format="pdf")
```

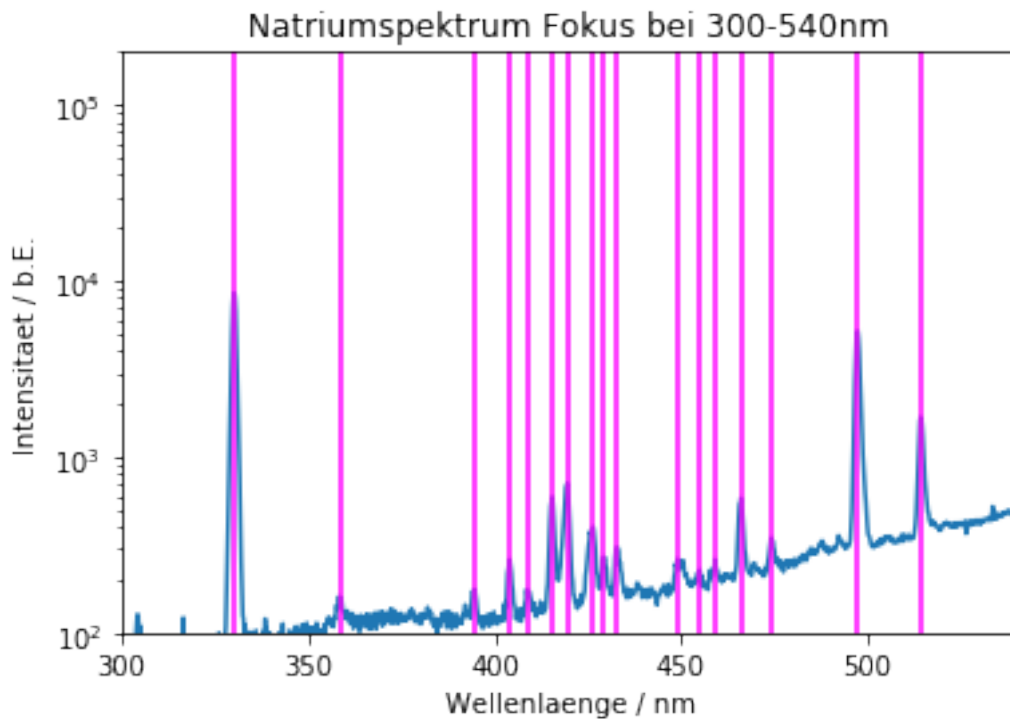


```
[26]: #Natrium linke Seite
lamb_nas, inten_nas=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/Natrium/
↳20msSpektrumlinks.txt', skiprows=17,
converters= {0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
plt.plot(lamb_nas, inten_nas)
plt.title('Natriumspektrum Fokus bei 300-540nm')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Intensitaet / b.E.')
plt.yscale('log')
plt.ylim((100,200000))
plt.xlim((300,540))
plt.savefig("300-540natrium.pdf", format="pdf")
```

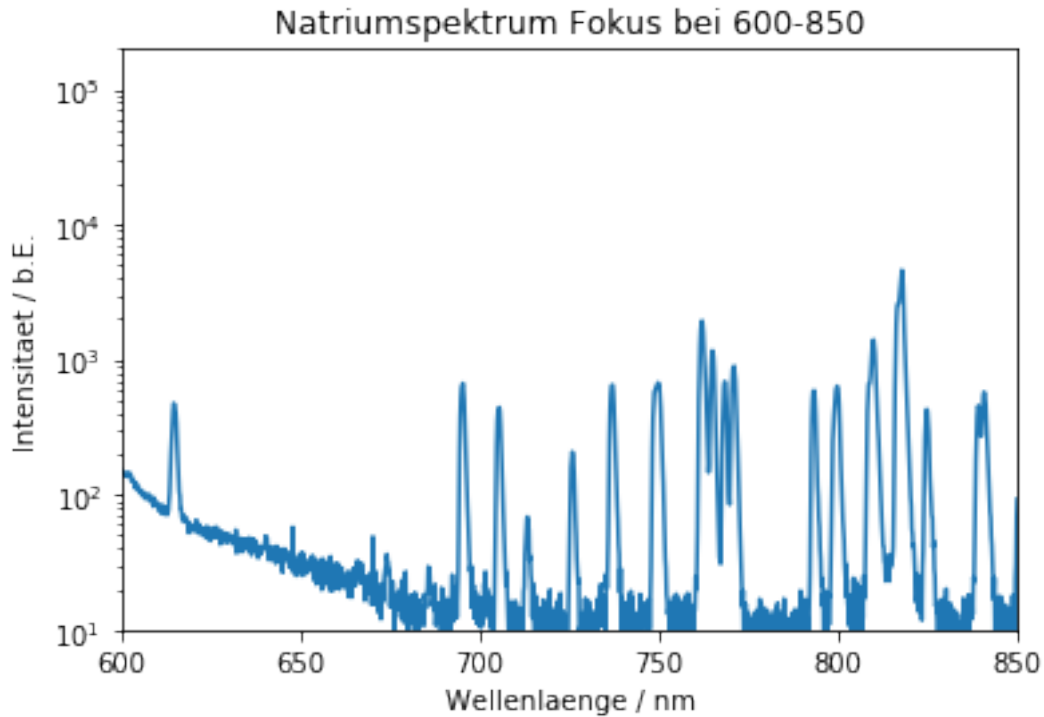



```
[27]: #Linien stellen
lamb_nas, inten_nas=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/Natrium/
↳20msSpektrumlinks.txt', skiprows=17,
converters={0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
plt.plot(lamb_nas, inten_nas)
plt.title('Natriumspektrum Fokus bei 300-540nm')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Intensitaet / b.E.')
plt.yscale('log')
plt.ylim((100,200000))
plt.xlim((300,540))
plt.axvline(x=330.1, ymin=0, ymax=1, color='magenta')
plt.axvline(x=358.395, ymin=0, ymax=1, color='magenta')
plt.axvline(x=394.5, ymin=0, ymax=1, color='magenta')
plt.axvline(x=404.2, ymin=0, ymax=1, color='magenta')
plt.axvline(x=408.7, ymin=0, ymax=1, color='magenta')
plt.axvline(x=415.5, ymin=0, ymax=1, color='magenta')
plt.axvline(x=419.9, ymin=0, ymax=1, color='magenta')
plt.axvline(x=426.3, ymin=0, ymax=1, color='magenta')
plt.axvline(x=429.357, ymin=0, ymax=1, color='magenta')
plt.axvline(x=432.6, ymin=0, ymax=1, color='magenta')
plt.axvline(x=449.1, ymin=0, ymax=1, color='magenta')
```

```
plt.axvline(x=454.8, ymin=0, ymax=1, color='magenta')
plt.axvline(x=459.1, ymin=0, ymax=1, color='magenta')
plt.axvline(x=466.0, ymin=0, ymax=1, color='magenta')
plt.axvline(x=474.5, ymin=0, ymax=1, color='magenta')
plt.axvline(x=497.2, ymin=0, ymax=1, color='magenta')
plt.axvline(x=514.2, ymin=0, ymax=1, color='magenta')
plt.savefig("300-540mitliniennatrium.pdf", format="pdf")
```

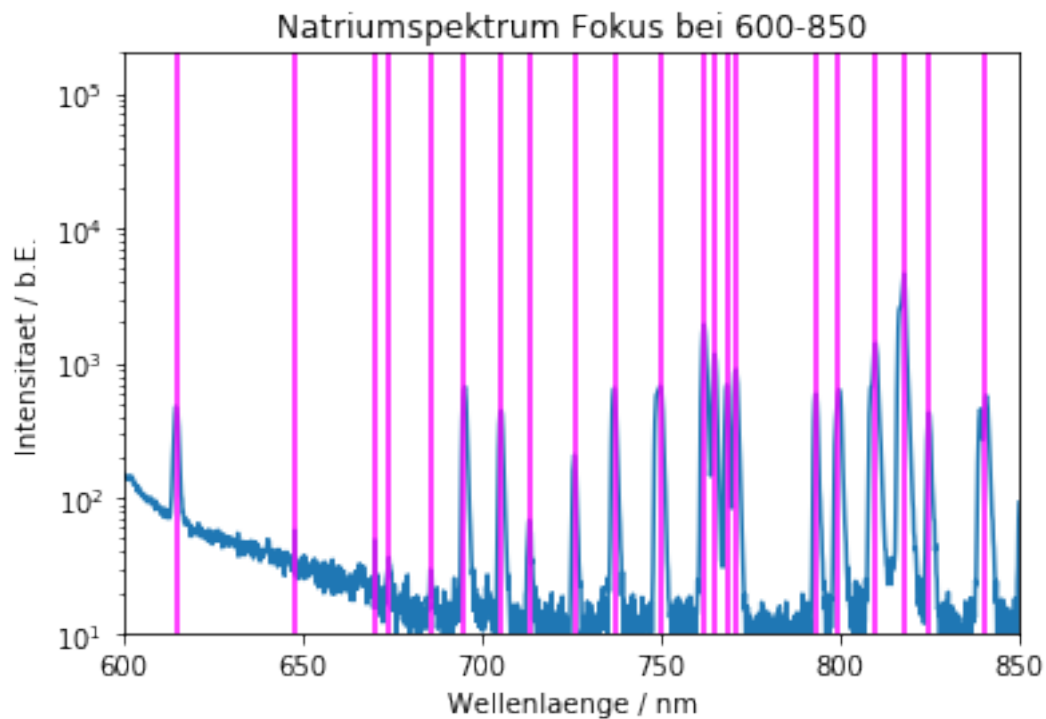


```
[28]: #Natrium rechte Seite
lamb_nass, inten_nass=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/Natrium/
↳Spektrum intensiver linien.txt', skiprows=17,
converters= {0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
plt.plot(lamb_nass, inten_nass)
plt.title('Natriumspektrum Fokus bei 600-850')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Intensitaet / b.E.')
plt.yscale('log')
plt.ylim((10,200000))
plt.xlim((600,850))
plt.savefig("600-850natirum.pdf", format="pdf")
```



```
[29]: #Linien stellen
lamb_nass, inten_nass=np.loadtxt('/home/urz/fphys/eh301/Pap2/234/Natrium/
↳Spektrum intensiver linien.txt', skiprows=17,
converters={0:comma_to_float, 1:comma_to_float},
comments='>', unpack=True)
plt.plot(lamb_nass, inten_nass)
plt.title('Natriumspektrum Fokus bei 600-850')
plt.xlabel('Wellenlaenge / nm')
plt.ylabel('Intensitaet / b.E.')
plt.yscale('log')
plt.ylim((10,200000))
plt.xlim((600,850))
plt.axvline(x=614.55, ymin=0, ymax=1, color='magenta')
plt.axvline(x=648.0, ymin=0, ymax=1, color='magenta')
plt.axvline(x=669.9, ymin=0, ymax=1, color='magenta')
plt.axvline(x=674.2, ymin=0, ymax=1, color='magenta')
plt.axvline(x=686.2, ymin=0, ymax=1, color='magenta')
plt.axvline(x=695.1, ymin=0, ymax=1, color='magenta')
plt.axvline(x=705.1, ymin=0, ymax=1, color='magenta')
plt.axvline(x=713.5, ymin=0, ymax=1, color='magenta')
plt.axvline(x=726.0, ymin=0, ymax=1, color='magenta')
plt.axvline(x=736.99, ymin=0, ymax=1, color='magenta')
plt.axvline(x=749.7, ymin=0, ymax=1, color='magenta')
```

```
plt.axvline(x=761.78, ymin=0, ymax=1, color='magenta')
plt.axvline(x=765.0, ymin=0, ymax=1, color='magenta')
plt.axvline(x=768.6, ymin=0, ymax=1, color='magenta')
plt.axvline(x=771.1, ymin=0, ymax=1, color='magenta')
plt.axvline(x=793.0, ymin=0, ymax=1, color='magenta')
plt.axvline(x=799.7, ymin=0, ymax=1, color='magenta')
plt.axvline(x=810.0, ymin=0, ymax=1, color='magenta')
plt.axvline(x=818.1, ymin=0, ymax=1, color='magenta')
plt.axvline(x=825.1, ymin=0, ymax=1, color='magenta')
plt.axvline(x=840.2, ymin=0, ymax=1, color='magenta')
plt.savefig("600-850mitliniennatirum.pdf", format="pdf")
```



```
[30]: #E_3p bestimmen
E_Ry= -2.179872*10**(-18)
h = 6.626*10**(-34)
c = 299792458
lam = 818.1*10**(-9)
E_3p = (E_Ry/9)-(h*c/lam)
deltaE_3p = h*c/((lam)**2)*0.6*10**(-9)
print(E_3p, "J")
print(deltaE_3p, "J")
```

```
-4.850175375513996e-19 J
1.7807813535122813e-22 J
```

```
[31]: E_3peV=-3.0272414922721053721 #eV
      deltaE_3peV=0.0011114763456250413044 #eV
```

```
[32]: #Wellenlängen bestimmen
      for m in range(3,13):
          l=1.2398E3/(-13.605/m**2-E_3peV)
          dl=(1.2398E3/((-13.605/m**2-E_3p)*(-13.605/m**2-E_3peV))*deltaE_3peV
          print('m={m:2d}, \u03BB={l:6.2f},\u0394={dl:6.2f}'.
                \u2192format(m=m,l=l,dl=dl))
```

```
m= 3, =818.04,\u0394 = -0.60
m= 4, =569.52,\u0394 = -0.74
m= 5, =499.31,\u0394 = -1.02
m= 6, =467.97,\u0394 = -1.38
m= 7, =450.90,\u0394 = -1.81
m= 8, =440.48,\u0394 = -2.30
m= 9, =433.61,\u0394 = -2.87
m=10, =428.82,\u0394 = -3.50
m=11, =425.35,\u0394 = -4.20
m=12, =422.74,\u0394 = -4.97
```

```
[33]: #Wellenlängen bestimmen
      E_3p=-3.0247
      Delta=0.0006 #Fehler3p
      delta=0.0001 #Fehler Delta_s
      k=1.3714 # Delta_s
      for m in range(4,10):
          l=1.2398E3/(-13.605/(m-k)**2-E_3p)
          dl1=1.2398E3/((-13.605/(m-k)**2-E_3p)*(-13.605/m**2-E_3p))*Delta
          dl2=((2*1.2398E3*13.605*(m-k))/((-13.605+3.0221*(m-k)*(m-k))*(-13.605+3.
                \u21920221*(m-k)*(m-k))))*delta
          dl_ges=(dl1**2+dl2**2)**(0.5)
          print('m={m:2d}, \u03BB={l:6.2f}, \u0394_ges={dl_ges:6.2f}'.
                \u2192format(m=m,l=l,dl_ges=dl_ges))
```

```
m= 4, =1174.41, \u0394_ges= 0.36
m= 5, =622.57, \u0394_ges= 0.15
m= 6, =518.82, \u0394_ges= 0.12
m= 7, =477.72, \u0394_ges= 0.10
m= 8, =456.64, \u0394_ges= 0.10
m= 9, =444.23, \u0394_ges= 0.09
```

```
[34]: lam1 = 589*10**(-9)
      E_3s = E_3p-(h*c)/lam1
      deltaE_3s = deltaE_3p
      print(E_3s, "J")
      print(deltaE_3s, "J")
```

-3.0247 J
1.7807813535122813e-22 J

```
[35]: #In eV Umformen
E_3seV=-5.1322141774303338124 #eV
deltaE_3seV=0.0011114763456250413044 #eV
```

```
[36]: #Berechnung del_s
del_s = 3- unp.sqrt((-13.605/E_3seV))
del_del_s=0.5*unp.sqrt((-13.605/((E_3seV)**3)))*deltaE_3seV
print(del_s, "eV")
print(del_del_s, "eV")
```

1.3718407270239386 eV
0.00017630407230286735 eV

```
[37]: k=del_s
for m in range(4,10):
    l=1.2398E3/(-13.605/(m-k)**2-E_3peV)
    dl1=1.2398E3/((-13.605/(m-k)**2-E_3peV)*(-13.605/m**2-E_3peV))*Delta
    dl2=((2*1.2398E3*13.605*(m-k))/((-13.605+3.0221*(m-k)*(m-k))*(-13.605+3.
    ↳0221*(m-del_s)*(m-k)))*del_del_s
    dl_ges=(dl1**2+dl2**2)**(0.5)
    print('m={m:2d}, \u03BB={l:6.2f}, \u0394\u03BB_ges={dl_ges:6.2f}'.
    ↳format(m=m,l=l,dl_ges=dl_ges))
```

m= 4, =1172.32, Δ _ges= 0.44
m= 5, =621.86, Δ _ges= 0.15
m= 6, =518.29, Δ _ges= 0.12
m= 7, =477.26, Δ _ges= 0.10
m= 8, =456.22, Δ _ges= 0.10
m= 9, =443.83, Δ _ges= 0.09

```
[ ]: #Berechnung Hauptserie
#Berechnung del_p
del_p = 3- unp.sqrt((-13.605/E_3peV))
del_del_p=0.5*unp.sqrt((-13.605/((E_3peV)**3)))*deltaE_3peV
print(del_p)
print(del_del_p, "eV")
g=del_p
for m in range(4,6):
    l=1.2398E3/(-13.605/(m-g)**2-E_3seV)
    dl1=1.2398E3/((-13.605/(m-g)**2-E_3seV)*(-13.605/m**2-E_3seV))*Delta
    dl2=((2*1.2398E3*13.605*(m-g))/((-13.605-E_3s*(m-g)*(m-g))*(-13.
    ↳605-E_3s*(m-g)*(m-g)))*del_del_p
    dl_ges=(dl1**2+dl2**2)**(0.5)
```

```

    print('m={m:2d}, \u03BB={l:6.2f}, \u0394\u03BB_ges={dl_ges:6.2f}'.
    ↪format(m=m,l=l,dl_ges=dl_ges))

```

[]:

```

[ ]: #Serienenergien bestimmen
#Erste nebenserie
wellenl=np.array([818.1, 568.0, 497.2, 466, 449.1, 432.16, 426.3,419.9])
fehler=np.array([0.6,0.6,0.6,0.6,0.6,0.6,0.6,0.6])
quantenz=np.array([3,4,5,6,7,9,10,12])
plt.errorbar(quantenz,wellenl,fehler, fmt=".")
plt.xlabel('Quantenzahl')
plt.ylabel('Wellenlaenge [nm]')
plt.title('1. Nebenserie des Na-Atoms')
from scipy.optimize import curve_fit
def fit_func(m,E_Ry,E_3p,D_d):
    return 1.2398E3/(E_Ry/(m-D_d)**2-E_3p)
para = [-13.6,-3,-0.02]
popt, pcov = curve_fit(fit_func, quantenz, wellenl,sigma=fehler ,p0=para)
x=np.linspace(2.7, 12.2,100)
plt.plot(x,fit_func(x,*popt))
print("E_Ry=",popt[0], ", Standardfehler=", np.sqrt(pcov[0][0]))
print("E_3p=",popt[1], ", Standardfehler=", np.sqrt(pcov[1][1]))
print("D_d=",popt[2], ", Standardfehler=", np.sqrt(pcov[2][2]))
plt.savefig("1Nebenserie.pdf", format="pdf")

```

```

[ ]: #chi-Wert bestimmen
chi2_=np.sum((fit_func(quantenz,*popt)-wellenl)**2/fehler**2)
dof=len(quantenz)-3 #dof:degrees of freedom, Freiheitsgrad
chi2_red=chi2_/dof
print("chi2=", chi2_)
print("chi2_red=",chi2_red)
from scipy.stats import chi2
prob=round(1-chi2.cdf(chi2_,dof),2)*100
print("Wahrscheinlichkeit:", prob,"%")

```

```

[ ]: #2 Nebenserie
wellenl2=np.array([614.6, 514.2, 474.5, 454.8])
fehler2=np.array([0.6,0.6,0.6,0.6])
quantenz2=np.array([5,6,7,8])
plt.errorbar(quantenz2,wellenl2,fehler2, fmt=".")
plt.xlabel('Quantenzahl')
plt.ylabel('Wellenlaenge [nm]')
plt.title('2. Nebenserie des Na-Atoms')
from scipy.optimize import curve_fit
def fit_func(m,E_Ry,E_3p,D_d):
    return 1.2398E3/(E_Ry/(m-D_d)**2-E_3p)

```

```

para2 = [-13.6,-3,-0.02]
popt2, pcov2 = curve_fit(fit_func, quantenz2, wellenl2,sigma=fehler2 ,p0=para2)
x2=np.linspace(2.8, 12.2,100)
plt.plot(x2,fit_func(x2,*popt2))
plt.xlim((4,10))
plt.ylim((0,1500))
print("E_Ry=",popt2[0], ", Standardfehler=", np.sqrt(pcov2[0][0]))
print("E_3p=",popt2[1], ", Standardfehler=", np.sqrt(pcov2[1][1]))
print("D_d=",popt2[2], ", Standardfehler=", np.sqrt(pcov2[2][2]))
plt.savefig("2Nebenserie.pdf", format="pdf")

```

```

[ ]: #chi-Wert bestimmen
chi3_=np.sum((fit_func(quantenz2,*popt2)-wellenl2)**2/fehler2**2)
dof3=len(quantenz2)-3 #dof:degrees of freedom, Freiheitsgrad
chi3_red=chi3_/dof3
print("chi3=", chi3_)
print("chi3_red=",chi3_red)
from scipy.stats import chi2
prob=round(1-chi2.cdf(chi3_,dof3),2)*100
print("Wahrscheinlichkeit:", prob,"%")

```