A1	A2	A3	A4	A5	Σ
4	5	—	—	3.25	12.2

SMD-Übungsblatt 11

Abgabe: 24.01.19

Yvonne Kasper yvonne.kasper@udo.edu ,
Robert Appel robert.appel@udo.edu ,
Julian Schröer julian.schroeer@udo.edu

1 Aufgabe1

Wir betrachten zwei Hypotesen:

$$\Delta E_A = 31{,}3\,\mathrm{meV} \quad \text{und} \quad \Delta E_B = 30{,}7\,\mathrm{meV}\,. \tag{1}$$

Der χ^2-Test wird wie folgt durch geführt:

$$t = \sum_{i=1}^{n} \frac{(\Delta E_i - \Delta E_H)^2}{\sigma_{t,i}^2} = \chi^2\,, \tag{2}$$

dabei bezeichnet t die Testgröße, ΔE_i die Messwerte, ΔE_H den Wert der Hypothese und $\sigma t, i = 0,5$ den Fehler dessen. Die Hypothese wird angenommen, wenn t einer χ^2-Verteilung folgt. Das wird überprüft indem die χ^2-Verteilung für die n Freiheitsgerade der Messwerte betrachtet wird und ein Annahme- und Verwurfsbereich abgesteckt wird. Der Annahmebreich enthält $(1-\alpha)$ der Fläche unter der Wahrscheinlichkeitsdichte des Testes. Der Verwurfsbreich enthält α der Fläche und ist die Signifikanz.
Hier ist $n = 7$ und $\alpha = 0,05$, damit ergibt sich $\chi^2_{(1-\alpha)} = 14,07$[1]. ✓ Mit Gleichung (2) ergibt sich

$$t_A \approx 3,04 \quad \text{und} \quad t_B \approx 10,96 \tag{3}$$

−0.5 −0.5

Für beide Testgrößen gilt $\leq 14,07$ damit wären beide Hypotesen angenommen. Bei genauerer Betrachtung ist $t_A < t_B \leq 14,7$ also t_B wesentlich nähr an der Verteilung als t_A. Deshalb sollte die Hypotese A verworfen werden und die Hypotese B weiterverfolgt werden, da diese die Messung besser beschreibt. 4/5P.

2 Aufgabe 2

2.1 Teil a)

Da bei der Poissonverteilung sowohl der Mittelwert als auch die Varianz durch λ gegeben ist, wählen wir für die Gaußverteilung $\mu = \lambda$ und $\sigma^2 = \lambda$. Da ich nicht weiß, wie ich das beweisen soll, habe ich ein Paar Beispielplots gemacht.

0.5P.

[1]Entnommen aus `http://eswf.uni-koeln.de/glossar/chivert.htm`.

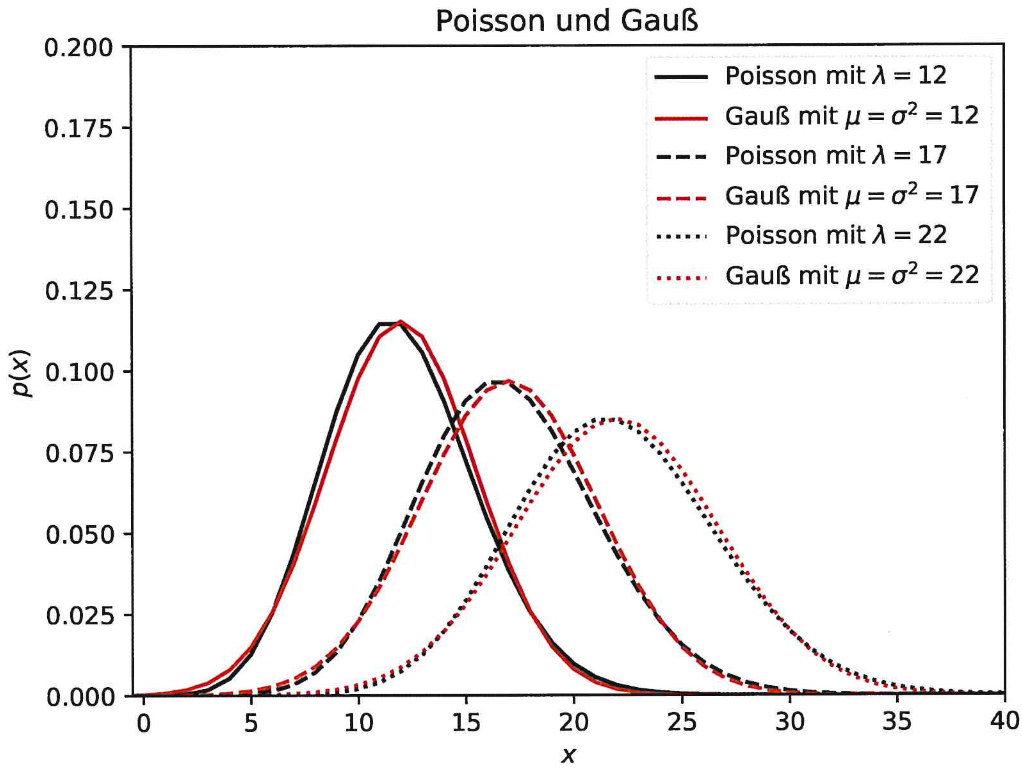

Abbildung 1: Beispielplots der Gauß- und Poissonverteilung mit drei unterschiedlichen λ.

Das sieht doch ganz ähnlich aus. Stimmt :)

2.2 Teil b)

Kolmogorow–Smirnow-Test wurde wie in der Vorlesung beschreiben implementiert.

2.3 Teil c) und d)

Es wurden für die Konfidenzniveaus $\alpha = 0,05$, $\alpha = 0,025$ und $\alpha = 0,001$ alle ganzzahligen $\lambda \in [1, 15]$ getestet.
Für $\alpha = 0,05$ kann der Kolmogorow–Smirnow-Test ab $\lambda = 8$ nicht mehr unterscheiden.
Für $\alpha = 0,025$ kann der Kolmogorow–Smirnow-Test für $\lambda = 6$ nicht mehr unterscheiden, dann für $\lambda = 7$ gehts wieder und ab $\lambda = 8$ dann nicht mehr. Ich nehme an das liegt an den gezogenen Zufallszahlen.
Für $\alpha = 0,001$ kann der Kolmogorow–Smirnow-Test ab $\lambda = 5$ nicht mehr unterscheiden.

4.5

5/5P.

3 Aufgabe 5

a Die Nullhypotese hier besagt, dass der Bin i den Erwartungswert μ_i besitzt. Die Zahl der Eintrage n_i im Bin i sind Zufallsvariablen verteilt gemäß der Poisson-Verteilung. Damit folgt

$$P(n_i|\mu_i) = \frac{\mu_i^{n_i}\exp(-\mu_i)}{n_i!} \quad \text{für} \quad n_i \tag{4}$$

$$P(m_i|\mu_i) = \frac{\mu_i^{m_i}\exp(-\mu_i)}{m_i!} \quad \text{für} \quad m_i \tag{5}$$

die Wahrscheinlichkeitsdichte für ein Eintrag in einem Bin. ABER: $\mu_{i,n} \neq \mu_{i,m}$ 1P.

b Hier bietet es sich an die negative Log-Likelyhood Funktion zu verwenden.

$$\begin{aligned} F(\mu_i) &= -\sum_{i=1}^{r} \log(P(n_i|\mu_i)\cdot P(m_i|\mu_i)) \\ &= -\sum_{i=1}^{r} \log\left(\frac{\mu_i^{n_i}\exp(-\mu_i)}{n_i!}\right) - \sum_{i=1}^{r} \log\left(\frac{\mu_i^{m_i}\exp(-\mu_i)}{m_i!}\right) \\ &= -\sum_{i=1}^{r}(n_i+m_i)\log(\mu_i) + 2\sum_{i=1}^{r}\mu_i + \underbrace{\sum_{i=1}^{r}\log(n_i!\cdot m_i!)}_{=konst,\ \text{da}\ \mu_i\ \text{unabhängig}} \end{aligned} \tag{6}$$

0.25P.

Die Likelyhood muss nun minimiert werden,deshalb

Aufgabe ist $\frac{dF}{d\mu_i} = 0$!

$$\frac{dF(\mu_i)}{d\mu_i} \stackrel{!}{=} 0 \tag{7}$$

$$\frac{dF(\mu_i)}{d\mu_i} = -\sum_{i=1}^{r}\frac{(n_i+m_i)}{\mu_i} + 2\sum_{i=1}^{r} 1 \tag{8}$$

$$\implies -\frac{1}{\mu_i}\sum_{i=1}^{r}(n_i+m_i) + 2r = 0 \tag{9}$$

$$\implies \hat{\mu_i} = \frac{\sum_{i=1}^{r}(n_i+m_i)}{2r} = \frac{N+M}{2r} \tag{10}$$

c Der χ^2-Test ist gegeben über

$$t = \chi^2 = \sum_{i=1}^{r}\frac{(n_i-n_0)^2}{n_0} + \sum_{i=1}^{r}\frac{(m_i-m_0)^2}{m_0} \tag{11}$$

f.

mit $n_0 = \frac{N}{r}$ und $m_0 = \frac{M}{r}$. ← Nimmt in jedem Bin gleichen Wert n_i, m_i an?

d Die χ^2-Verteilung hat $k = r$ Freiheitsgerade, vorrausgesetzt die Einträge sind nicht normiert. Dann wäre der Freiheitsgerade $k = r - 1$. Sie sind normiert.
Für den hier verwendeten χ^2-Test wird die Gaußsche-Nährung $\sigma^2 = n_0$ verwendet, diese ist aber nur für n_o hinreichend groß (> 10) erfüllt. Ist $n_0 < 10$ kann dem Test nicht mehr vertraut werden, da er nicht mehr einer χ^2-Verteilung folgt. 1P.

e Für den angegeben χ^2-Test ergibt sich $t \approx 8.43$. Nun werden die Signifikanzen $\alpha = 0,1; 0,05; 0,01$ betraachtet, damit ergibt sich $(1-\alpha) = 0,9; 0,95; 0,99$. Die dazu gehörigen χ^2-Werte für $k = 3$ sind $\chi^2(3) = 6,25; 7,81; 11,34$ damit würde die Nullhypotese nur für $\alpha = 0,01$ nicht verworfen werden. 1P.

$k=2$, ansonsten ok.

3.25/5P.

Code fuer Blatt11

Kasper, Appel, Schroeer

28. Januar 2019

../B/1/Blatt11_Kasper_Appel_Schroeer/Blatt11.py

```python
def Aufgabe2():
    def G(x, mu, sigma):
        return ((1/np.sqrt(2 * np.pi * sigma**2)) * np.exp(-((x - mu)**2)/(2 * sigma**2)))

    l_values = [12, 17, 22]
    linestyles_values = ['-', '--', ':']

    for lamdahh, ls_v in zip(l_values, linestyles_values):
        dist = poisson(lamdahh)
        x = np.arange(-1, 300)

        plt.plot(x, dist.pmf(x), ls=ls_v, color='black',
                 label=r'Poisson mit $\lambda=%i$' % lamdahh)
        plt.plot(x, G(x, lamdahh, np.sqrt(lamdahh)),ls=ls_v, color='red', label=r'Gauß mit $\mu=\sigma^2=%i$'%lamdahh )

    plt.xlim(-0.5, 40)
    plt.ylim(0, 0.2)
    plt.legend()
    plt.xlabel('$x$')
    plt.ylabel(r'$p(x)$')
    plt.title('Poisson und Gauß')
    plt.savefig('testplot.pdf')
    plt.clf()


    def Kologomorow(A, B, alpha):
        KulSummeA = np.cumsum(A[0])
        KulSummeB = np.cumsum(B[0])
        Abstand = max(np.abs(KulSummeA - KulSummeB))
        langeA, langeB = len(A[0]), len(A[0])

        d = np.sqrt(langeA * langeB / (langeA + langeB)) * Abstand
        K_alpha = np.sqrt(1/2 * np.log(2 / alpha))

        ablehnen = False
        if(d > K_alpha):
            ablehnen = True
        return ablehnen
    #--test- Kologomorow spuckt True aus.
    # lamdahh = 8
    # mu = lamdahh
    # sigma = np.sqrt(lamdahh)
    # p = np.random.poisson(lamdahh, 10000)
    # g = np.random.normal(mu, sigma, 10000)
    # g = np.floor(g)
    #
    # bins = np.linspace(lamdahh - 5*np.sqrt(lamdahh), lamdahh + 5*np.sqrt(lamdahh), 100)
    #
    # phist = np.histogram(p, bins=bins, density=True)
    # ghist = np.histogram(g, bins=bins, density=True)
    # print(Kologomorow(phist,ghist,0.25))

    for alpha in (0.05, 0.025, 0.001):
        print('alpha ist:', alpha)
        for lamdahh in range(1, 15, 1):
            mu = lamdahh
```

```python
        sigma = np.sqrt(lamdahh)
        p = np.random.poisson(lamdahh, 10000)
        g = np.random.normal(mu, sigma, 10000)
        g = np.floor(g)

        bins = np.linspace(lamdahh - 5*np.sqrt(lamdahh), lamdahh + 5*np.sqrt(lamdahh), 100)

        phist = np.histogram(p, bins=bins, density=True)
        ghist = np.histogram(g, bins=bins, density=True)

        Kolo = Kologomorow(phist, ghist, alpha)

        print('bei alpha =', alpha, 'und lambdah=', lamdahh, 'sagt -KolmogorowSmirnow:', Kolo)

    print('Sorry für den Spam')

if __name__ == '__main__':
    import numpy as np
    from scipy.stats import poisson
    import matplotlib.pyplot as plt

    Aufgabe2()
```