22	23	24	
6/6	6/6	7,5/8	19,5/20

blatt08_nitschke_grisard

December 13, 2018

1 Blatt 8

1.1 Aufgabe 22: Fehlerfortpflanzung

```
In [1]: import numpy as np
        import matplotlib.pyplot as plt
        import uncertainties.unumpy as unp
        from uncertainties import ufloat
```

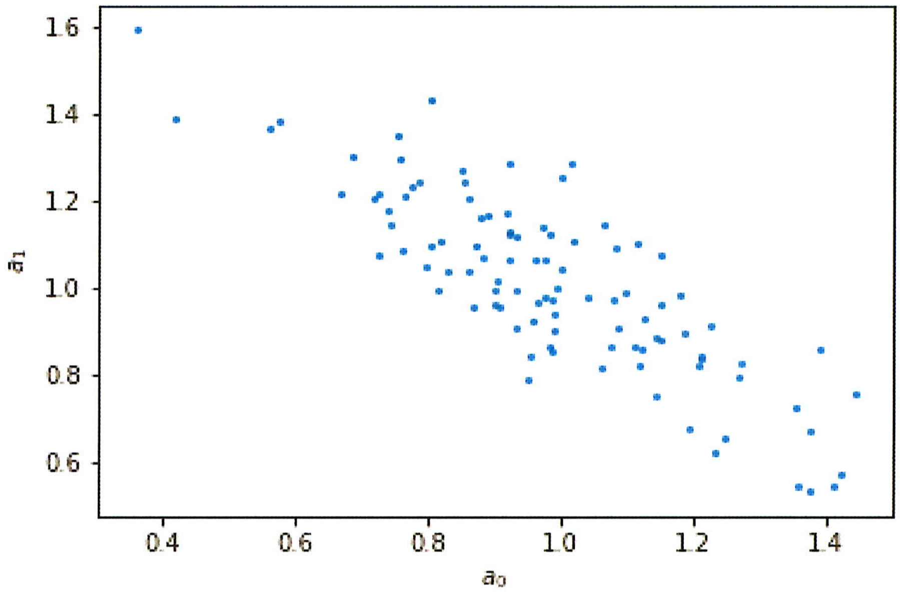
```
In [2]: sig0 =  0.2
        sig1 =  0.2
        rho  = -0.8
        cov = [[sig0**2, rho * sig0 * sig1], [rho * sig0 * sig1, sig1**2]]

        a = np.random.multivariate_normal(mean = [1, 1], cov = cov, size = 100)
```

hier hättet ihr ruhig mehr Werte nehmen können

```
In [3]: plt.scatter(a[:, 0], a[:, 1], alpha = 0.9, s = 5)
        plt.xlabel('$a_0$')
        plt.ylabel('$a_1$')
```

```
Out[3]: Text(0,0.5,'$a_1$')
```

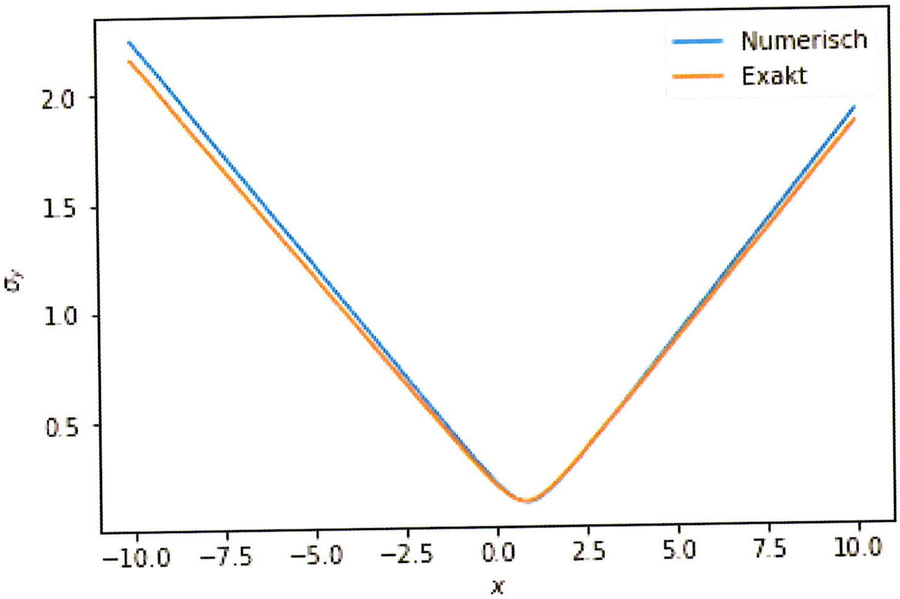


sieht richtig aus :)

2 P.

```
In [4]: x = np.linspace(-10, 10, 100)
        stdy = [np.std(a[:, 0] + a[:, 1] * x_i) for x_i in x]
        plt.plot(x, stdy, label = 'Numerisch' )

        def exact(x):
            return np.sqrt(sig0**2 + x**2 * sig1**2 + 2 * rho * sig0 * sig1 * x)
        plt.plot(x, exact(x), label = 'Exakt')
        plt.xlabel('$x$')
        plt.ylabel('$\sigma_y$')
        plt.legend()
        plt.show()
```



sehr schön !

```
In [5]: x = [-3, 0, 3]
        for x_i in x:
            y = a[:, 0] + a[:, 1] * x_i
            print(f'x = {x_i}:\t y = {np.mean(y)} +/- {np.std(y)}')
```

```
x = -3:         y = -2.055500218300318 +/- 0.8068818144837163
x = 0:          y = 0.9836304809349856 +/- 0.21175884014833143
x = 3:          y = 4.022761180170289 +/- 0.4588214095750574
```

Kommentar: Die numerischen Ergebnisse stimmen recht gut mit den exakt bestimmten Werten überein (siehe Handschriftliches). Die Präzision nimmt für groSSe Beträge von x ab (siehe Plot oben). Durch eine VergröSSerung des Samples könnte die Präzision gesteigert werden.

2P.

1.2 Aufgabe 24: F(uckin')-Praktikum

Famoses

```
In [6]: Asymmetrie = np.matrix([-0.032, 0.01, 0.057,
                                0.068, 0.076, 0.08,
                                0.031, 0.005, -0.041,
                                -0.09, -0.088, -0.074])
        Psi = np.arange(0, 331, 30)
```

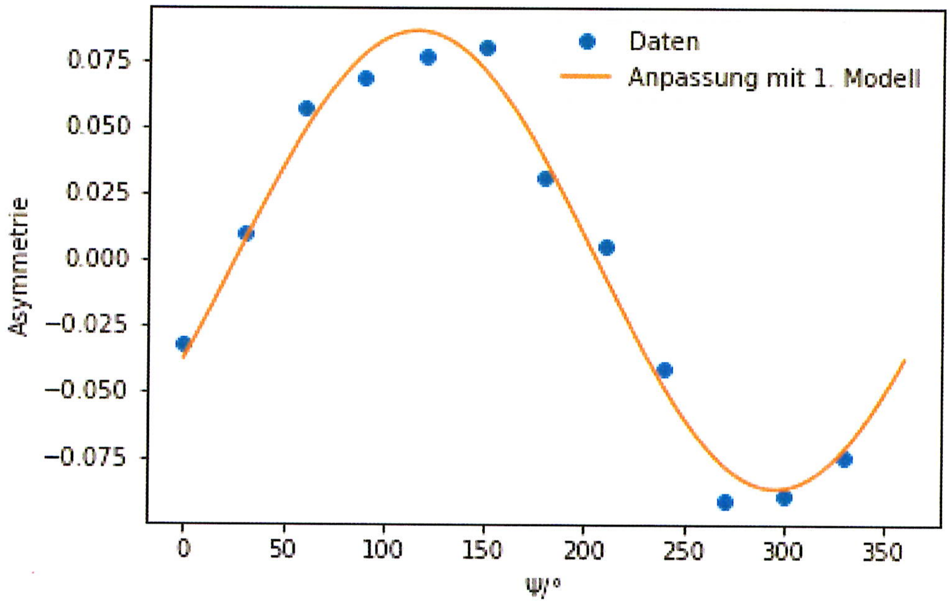
```
In [7]: A = np.matrix([np.cos(Psi / 180 * np.pi), np.sin(Psi / 180 * np.pi)]).T
```

```
In [8]: a = (A.T * A).I * A.T * Asymmetrie.T
        a
```

```
Out[8]: matrix([[-0.0375063 ],
                [ 0.07739978]])
```

```
In [9]: Psi_plot = np.linspace(0, 360, 100)
        plt.plot(Psi, Asymmetrie.T, 'o', label = 'Daten')
        plt.xlabel('$\Psi / ř$')
        plt.ylabel('Asymmetrie')
        plt.plot(Psi_plot,
                 a[0, 0] * np.cos(Psi_plot / 180 * np.pi) + a[1,0] * np.sin(Psi_plot / 180 * np
                 label = 'Anpassung mit 1. Modell')
        plt.legend()
```

```
Out[9]: <matplotlib.legend.Legend at 0x1f69b997c18>
```




```
In [10]: sigma = 0.011
         Cov = np.array(sigma**2 * (A.T * A).I)
```

```
In [11]: rho = Cov[0, 1] / np.sqrt(Cov[0, 0] * Cov[1, 1])
         print(f'a1 = {ufloat(a[0, 0], np.sqrt(Cov[0, 0]))}')
         print(f'a2 = {ufloat(a[1, 0], np.sqrt(Cov[1, 1]))}')
         print(f'Korrelationskoeffizient: {rho}')
```

```
a1 = -0.038+/-0.004
a2 = 0.077+/-0.004
Korrelationskoeffizient: -6.542726332681614e-17
```

in der Klausur müsst ihr das evtl. per Hand berechnen

1,5 P.

1,5 P.

Die Korrelation ist fast 0. Das Modell ist also gut gewählt, was letztlich an der Orthogoalität der Funktionen Sinus und Kosinus liegt.

```
In [12]: delta = unp.arctan(- a[1, 0] / a[0, 0])
         A0 = a[0, 0] / unp.cos(unp.arctan(- a[1, 0] / a[0, 0]))
```

Ihr müsst hier arctan2() benutzen

```
In [13]: a1 = a[0,0]
         a2 = a[1,0]

         J = np.matrix([[a1 / np.sqrt(a1**2 + a2**2), a2 / np.sqrt(a1**2 + a2**2)],
              [a2 / (a1**2 + a2**2), -1 / (a1 + a2**2 / a1)]])
         Cov2 = J * Cov * J.T
         print(f'A_0 = {ufloat(A0, np.sqrt(Cov2[0, 0]))}')
         print(f'delta = {ufloat(delta, np.sqrt(Cov2[1, 1]))}')
         print(f'Korrelation: {Cov2[1, 0] / np.sqrt(Cov2[0, 0] * Cov2[1, 1]) }')
```

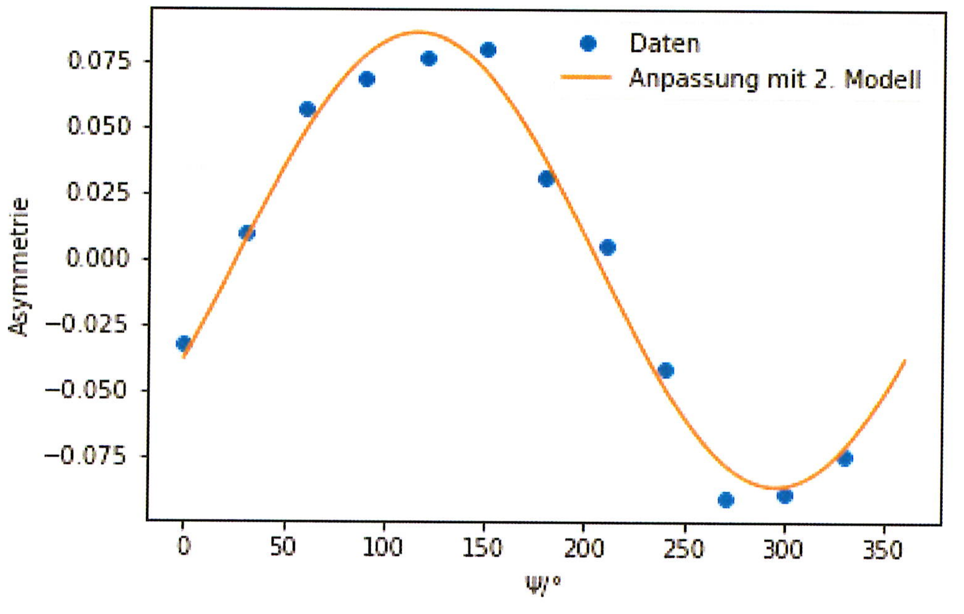
```
A_0 = -0.086+/-0.004
delta = 1.12+/-0.05
Korrelation: -1.155998080162034e-16
```

Die Korrelation ist ebenfalls sehr klein aber etwa 10x gröSSer als bei dem anderen Modell.

```
In [14]: Psi_plot = np.linspace(0, 360, 100)
         plt.plot(Psi, Asymmetrie.T, 'o', label = 'Daten')
         plt.xlabel('$\Psi / ř$')
         plt.ylabel('Asymmetrie')
         plt.plot(Psi_plot, A0 * unp.cos(Psi_plot / 180 * np.pi + delta), label = 'Anpassung m:
         plt.legend()

Out[14]: <matplotlib.legend.Legend at 0x1f69be37278>
```

3P.


Daten
Anpassung mit 2. Modell
Asymmetrie
0.075
0.050
0.025
0.000
−0.025
−0.050
−0.075
0
50
100
150
200
250
300
350
ψ/°

8. Übung zur SMD

13.12.18

22

a) $y = a_0 + a_1 x$

$$\sigma_y^2 = \left(\frac{\partial y}{\partial a_0}\sigma_{a_0}\right)^2 + \left(\frac{\partial y}{\partial a_1}\sigma_{a_1}\right)^2 + 2\frac{\partial y}{\partial a_0}\frac{\partial y}{\partial a_1}\operatorname{cov}(a_0, a_1)$$

ohne Korrelation:

$$\sigma_y^2 = (\sigma_{a_0})^2 + (\sigma_{a_1} x)^2$$

$$\rightarrow \sigma_y = \left((0,2)^2 + (0,2)^2 x^2\right)^{1/2}$$

$$= 0,2\,(1 + x^2)^{1/2} \checkmark$$

mit Korrelation

$$\sigma_y = \left\{(\sigma_{a_0})^2 + (\sigma_{a_1} x)^2 + 2x \operatorname{cov}(a_0, a_1)\right\}^{1/2}$$

$$= \left\{(\sigma_{a_0})^2 + (\sigma_{a_1} x)^2 + 2x\, \varrho\, \sigma_{a_0}\sigma_{a_1}\right\}^{1/2}$$

$$= 0,2\,\left\{x^2 + 2\varrho x + 1\right\}^{1/2} \checkmark$$

2P.

b) im Notebook

c) analytisch:

$y(-3) \approx -2,000 \pm 0,769$

$y(0) \approx 1,0 \pm 0,2$

$y(3) \approx 4,000 \pm 0,456$ ✓

numerisch + Vergleich im Notebook

23

a) Designmatrix

$$A = \begin{pmatrix} f_1(\varphi_1) & f_2(\varphi_1) \\ f_1(\varphi_2) & f_2(\varphi_2) \\ \vdots & \vdots \\ f_1(\varphi_N) & f_2(\varphi_N) \end{pmatrix} = \begin{pmatrix} \cos(\varphi_1) & \sin(\varphi_1) \\ \cos(\varphi_2) & \sin(\varphi_2) \\ \vdots & \vdots \end{pmatrix}$$ ✓

lass ich mal durchgehen :) 1,5P.

→ expl. Berechnung im Notebook

~~b)~~ c) im Notebook

d) Drücke A_0 und δ durch a_1 und a_2 aus:

$$A_0 \cos(\omega t + \delta) = \underbrace{A_0 \cos(\delta)}_{a_1} \cos \omega t \underbrace{- A_0 \sin \delta}_{a_2} \sin \omega t$$

$$\rightarrow \quad A_0 = \frac{a_1}{\cos\left(\arctan\left(-\frac{a_2}{a_1}\right)\right)}$$

$$\delta = \arctan\left(-\frac{a_2}{a_1}\right)$$

Kovarianzmatrix von A_0, δ mit

$$\mathrm{Cov}(A_0, \delta) = J V J^T \quad \text{mit}$$

$$V = \begin{pmatrix} \sigma_{a_1}^2 & \mathrm{cov}(a_1, a_2) \\ \mathrm{cov}(a_1, a_2) & \sigma_{a_2}^2 \end{pmatrix}$$

und Jacobimatrix J:

$$J = \begin{pmatrix} \frac{\partial A_0}{\partial a_1} & \frac{\partial A_0}{\partial a_2} \\ \frac{\partial \delta}{\partial a_1} & \frac{\partial \delta}{\partial a_2} \end{pmatrix}$$

$$= \begin{pmatrix} \frac{a_1}{\sqrt{a_1^2 + a_2^2}} & \frac{a_2}{\sqrt{a_1^2 + a_2^2}} \\ \frac{a_2}{a_1^2 + a_2^2} & -\frac{1}{a_1 + \frac{a_2^2}{a_1}} \end{pmatrix}$$

Berechnung im Notebook

A23

a) $x_1 = a z_1 + b \qquad \begin{pmatrix} z_1 & 1 \\ z_2 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \underbrace{\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}}_{\vec{x}}$

$x_2 = a z_2 + b$

$$A^T = \begin{pmatrix} z_1 & z_2 \\ 1 & 1 \end{pmatrix} \qquad A^T A = \begin{pmatrix} z_1^2 + z_2^2 & z_1 + z_2 \\ z_1 + z_2 & 2 \end{pmatrix}$$

$$\det(A^T A) = 2(z_1^2 + z_2^2) - (z_1 + z_2)^2 = (z_1 - z_2)^2$$

$$(A^T A)^{-1} = \frac{1}{\det(A^T A)} \begin{pmatrix} 2 & -(z_1+z_2) \\ -(z_1+z_2) & z_1^2 + z_2^2 \end{pmatrix}$$

$$(A^T A)^{-1} A^T = \frac{1}{\det(A^T A)} \begin{pmatrix} 2z_1 - (z_1+z_2) & 2z_2 - (z_1+z_2) \\ -z_1(z_1+z_2) + z_1^2 + z_2^2 & -z_2(z_1+z_2) + z_1^2 + z_2^2 \end{pmatrix}$$

$$= \frac{1}{\det(A^T A)} \begin{pmatrix} z_1 - z_2 & z_2 - z_1 \\ z_2^2 - z_1 z_2 & z_1^2 - z_2 z_1 \end{pmatrix}$$

$$(A^T A)^{-1} A^T \vec{x} = \frac{1}{\det(A^T A)} \begin{pmatrix} x_1(z_1 - z_2) + x_2(z_2 - z_1) \\ x_1(z_2^2 - z_1 z_2) + x_2(z_1^2 - z_2 z_1) \end{pmatrix}$$

$$a = \frac{x_1 - x_2}{z_1 - z_2} \checkmark \qquad b = x_2 \frac{z_1}{z_1 - z_2} - x_1 \frac{z_2}{z_1 - z_2} \checkmark \qquad 1P.$$

$$\begin{pmatrix} a \\ b \end{pmatrix} = \underbrace{\begin{pmatrix} \frac{1}{z_1 - z_2} & -\frac{1}{z_1 - z_2} \\ -\frac{z_2}{z_1 - z_2} & \frac{z_1}{z_1 - z_2} \end{pmatrix}}_{B \checkmark} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \qquad V(x) = \begin{pmatrix} \sigma_{x_1}^2 & 0 \\ 0 & \sigma_{x_2}^2 \end{pmatrix}$$

$$\sigma_{x_1}^2 = \sigma_1^2 \; ; \; \sigma_{x_2}^2 = \sigma_2^2$$

$$V(a,b) = B V B^T = \begin{pmatrix} \frac{1}{z_1 - z_2} & -\frac{1}{z_1 - z_2} \\ -\frac{z_2}{z_1 - z_2} & \frac{z_1}{z_1 - z_2} \end{pmatrix} \begin{pmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{pmatrix} \begin{pmatrix} \frac{1}{z_1 - z_2} & -\frac{z_2}{z_1 - z_2} \\ -\frac{1}{z_1 - z_2} & \frac{z_1}{z_1 - z_2} \end{pmatrix}$$

$$= \begin{pmatrix} 1/(z_1 - z_2) & -1/(z_1 - z_2) \\ -z_2/(z_1 - z_2) & z_1/(z_1 - z_2) \end{pmatrix} \begin{pmatrix} \sigma_1^2 \left(\frac{1}{z_1 - z_2}\right) & -\sigma_1^2 \left(\frac{z_2}{z_1 - z_2}\right) \\ -\sigma_2^2 \left(\frac{1}{z_1 - z_2}\right) & \sigma_2^2 \left(\frac{z_1}{z_1 - z_2}\right) \end{pmatrix}$$

$$= \begin{pmatrix} \sigma_1^2 \frac{1}{(z_1 - z_2)^2} + \sigma_2^2 \frac{1}{(z_1 - z_2)^2} \checkmark & -\sigma_1^2 \frac{z_2}{(z_1 - z_2)^2} - \sigma_2^2 \frac{z_1}{(z_1 - z_2)^2} \checkmark \\ -\sigma_1^2 \frac{z_2}{(z_1 - z_2)^2} - \sigma_2^2 \frac{z_1}{(z_1 - z_2)^2} \checkmark & \frac{\sigma_1^2 z_2^2}{(z_1 - z_2)^2} + \frac{\sigma_2^2 z_1^2}{(z_1 - z_2)^2} \checkmark \end{pmatrix} \qquad 1P.$$

$$\rightarrow \sigma_A = \frac{1}{(z_1 - z_2)} \sqrt{\sigma_1^2 + \sigma_2^2} \qquad \sigma_B = \frac{1}{z_1 - z_2} \sqrt{\sigma_1^2 z_2^2 + \sigma_2^2 z_1^2}$$

$$\mathrm{Cov}(a,b) = \frac{1}{(z_1 - z_2)^2} \left(-\sigma_1^2 z_2 - \sigma_2^2 z_1 \right)$$

$$\rho = \frac{Cov(a,b)}{\sigma_A \sigma_B} = \frac{(-\sigma_1^2 z_2 - \sigma_2^2 z_1)}{\sqrt{(\sigma_1^2+\sigma_2^2)(\sigma_1^2 z_2^2 + \sigma_2^2 z_1^2)}}$$

$$= \frac{(-\sigma_1^2 z_2 - \sigma_2^2 z_1)}{\sqrt{z_2^2(\sigma_1^4+\sigma_1^2\sigma_2^2) + z_1^2(\sigma_2^4+\sigma_1^2\sigma_2^2)}}$$ ✓ 1P.

b)

$$x_3 = z_3 \frac{x_1 - x_2}{z_1 - z_2} + \frac{x_2 z_1}{z_1 - z_2} - \frac{x_1 z_2}{z_1 - z_2}$$ ✓ 1P.

$$= \frac{1}{z_1 - z_2}\left[z_3 x_1 - z_3 x_2 + x_2 z_1 - x_1 z_2\right]$$

$$= \frac{1}{z_1 - z_2}\left[x_1[z_3 - z_2] + x_2[z_1 - z_3]\right]$$

$$\sigma_{x_3} = \sqrt{(z_3\sigma_a)^2 + \sigma_b^2 + 2 z_3 Cov(a,b)},$$ 1P.

da $\frac{\partial x_3}{\partial a} = z_3 \quad \frac{\partial x_3}{\partial b} = 1$ ✓

c) $\sigma_{x_3} = \sqrt{(z_3\sigma_a)^2 + \sigma_b^2}$ ← wäre größer als in b), da $Cov(a,b) < 0$

$$= \frac{1}{z_1 - z_2}\sqrt{z_3^2(\sigma_1^2+\sigma_2^2) + z_2^2\sigma_1^2 + \sigma_2^2 z_1^2}$$ ✓

1P.