Karlsruher Institut für Technologie

Due Date: Computerpraktikum am 29.06. und 30.06.2023

Verantwortlich für Vorlesung bzw. Computerpraktikum:
Dr. P. Goldenzweig, P.-D. Dr. R. Wolf, Dr. S. Stefkova
Tutoren: Dr. G. De Pietro, Dr. R. Quishpe, Dr. S. Mitra, G. Heine, A. Brusamolino, T. Voigtländer

Computerpraktikum zur Vorlesung Moderne Methoden der Datenanalyse - Blatt 6

Exercise 6.1: Hypothesis Testing

"Is this a new discovery or just a statistical fluctuation?" Statistics offers some methods to give a quantitative answer. But these methods should not be used blindly. In particular one should know exactly what the obtained numbers mean and what they don't mean.

- **Exercise 6.1.1:** **obligatory to solve either 6.1.1 or 6.1.2**

  The following table shows the number of winners in a horse race for different track numbers:

track	1	2	3	4	5	6	7	8
#winners	29	19	18	25	17	10	15	11

  Use a χ^2 test to check the hypothesis that the track number has *no* influence on the chance to win. Define a significance level, e.g., $\alpha = 5\,\%$ or $\alpha = 1\,\%$, *before* you do the test.

- **Exercise 6.1.2:** **obligatory to solve either 6.1.1 or 6.1.2**

  In a counting experiment 5 events are observed while $\mu_B = 1.8$ background events are expected. Is this a significant ($= 3\,\sigma$) excess? Calculate the probability of observing 5 or more events when the expectation value is 1.8 using Poisson statistics.

Exercise 6.2: Parameter Estimation

- **Exercise 6.2.1:** **voluntary**

  Consider the following set of values approximately following a Gaussian distribution. The set of values can be found in the repository in the file `exercise_6_2_1.csv`.

x_i	y_i	σ_i	x_i	y_i	σ_i	x_i	y_i	σ_i	x_i	y_i	σ_i
0.46	0.19	0.05	0.69	0.27	0.06	0.71	0.28	0.05	1.04	0.62	0.01
1.11	0.68	0.05	1.14	0.70	0.07	1.17	0.74	0.08	1.20	0.81	0.09
1.31	0.93	0.10	2.03	2.49	0.03	2.14	2.73	0.04	2.52	3.57	0.01
3.24	3.90	0.07	3.46	3.55	0.03	3.81	2.87	0.03	4.06	2.24	0.01
4.93	0.65	0.10	5.11	0.39	0.07	5.26	0.33	0.05	5.38	0.26	0.08

$$y(x) = a_1 e^{-\frac{1}{2}\left(\frac{x-a_2}{a_3}\right)^2}$$

with σ_i being the uncertainty on y_i.

Determine the values of the three parameters a_1, a_2 and a_3 as well as their uncertainties.

Afterwards, use the transformation $z = \ln y$ to obtain the linear function $z(x) = b_1 + b_2 x + b_3 x^2$. Determine the new parameters b_1, b_2, and b_3 and uncertainties in two ways and compare the results:

1. Fit the new function $z(x) = b_1 + b_2 x + b_3 x^2$ to the transformed data.
2. Calculate the new parameters using the transformation $z = \ln y$ and the values for a_j which you obtained before.

- **Exercise 6.2.2:** **obligatory**

  This exercise aims at constructing the error band around a function $f(x)$, fitted to data points (x, y) - i.e. the errors on the fitted parameters are transformed into errors on the value of the function at each value of x.

  Let us attack the problem in steps:

  - First, define a function for our problem, a straight line $f(x) = a + bx$ which is used both for creating the data and fitting
  - Next, consider $n = 11$ data points in the interval $[10, 20]$ and $f(x) = x$. To simulate measurement errors, shift the data points in y-direction by a random shift, drawn from a Gaussian distribution with $\sigma = 0.5$ and $\mu = 0.0$.
  - Now fit the straight line defined above to your data points using a fitting tool of your choice (for example *scipy.optimize* or *ROOT*) and store both the fit result and the corresponding covariance matrix.
  - Draw the data points and the fit result and print the correlation coefficient of the errors on the parameters a and b.
  - Try to give an intuitive argument why the two parameters a (axis intercept) and b (slope) are strongly correlated.
  - Next, construct the error band around the fit function: To do so, write a function *make_band(f, cov, withCorr=False)*, which takes the function f and the covariance matrix as input arguments and calculates for each value of x the error on y, $\Delta_y(x)$. As a first approach, use the simple formula for error propagation, which, in this case, results in $\Delta y(x)^2 = {\Delta_a}^2 + (x * \Delta_b)^2$ if correlations are neglected. Draw the error band $y(x) \pm \Delta_y(x)$ on top of the data points and the fit. Does this look correct?
  - Derive the appropriate formula for error propagation taking into account the correlation of the errors. Re-calculate $\Delta_y(x)$ and plot the corresponding error band. Compare with the above result which was obtained without taking correlations into account.

  To get a better idea of what happens here and what the effect of the correlations is, you might want to repeat the whole exercise setting in the range $[-5, 5]$, meaning now that the mean of the x-values of the data points is approximately at 0.