Machine Learning
Exercise 2: Manual implementation of linear regression in python

Prof. Dr. Thomas Kopinski

July 10, 2023

Abstract

This week's tasks focus on deepening your knowledge about linear regression by manually implementing the underlying algorithm in python.

Task 1: Understand the algorithm

- Additional information about implementing linear regression in python can again be found here.
- Make sure you understand how the algorithm works before you start implementing any python code.
- Try to structure your code in a clear way, e.g. by using functions and object oriented programming.

Task 2: Implement a general least squares optimizer

- Consider an array of x-values with a variable length with corresponding y-values and statistical errors y_{err}
- x = [1.47,1.50 ,1.52,1.55 ,1.57, 1.60,1.63,1.65,1.68,1.70,1.73,1.75,1.78,1.80,1.83]
- y = [52.21,53.12,54.48,55.84,57.20,58.57,59.93,61.29,63.11,64.47,66.28,68.10,69.92,72.19,74.46]
- y_{err} is 1 for each y
- Define an empty matrix A of shape (len(x), 2)
- Fill the matrix with the values $A_{i,0} = 1$ and $A_{i,1} = x$
- Define a diagonal matrix W from the y_{err} such that $W_{i,i} = \frac{1}{y_{err}^2}$
- Reshape the array of y vectors into a column vector
- Calculate the result of the function $p = (A^T WA)^{-1} A^T y$ and $cov = (A^T WA)^{-1}$ (Hint: use the Invert-Method for numpy matrices)
- Display the coeffictients results and plot the data and fit function (Hint: for errorbars use plt.errorbar)