

Die Methode der kleinsten Quadrate für Linearkombinationen von Funktionen

Zur Anpassung einer Linearkombination aus Funktionen

$$\sum_i^p a_i \cdot f_i(x) \quad (1)$$

an N Datenpunkte (x_j, y_j) kann die analytische Methode der kleinsten Quadrate verwendet werden. Ziel ist es, die Parameter a_i zu bestimmen, bei denen die Summe

$$\sum_j^N \left(y_j - \sum_i^p a_i \cdot f_i(x_j) \right)^2 \quad (2)$$

minimal ist, der Funktionsgraph also minimal von den Datenpunkten abweicht.

Bestimmung der Parameter

1. Zuerst wird die sogenannte Designmatrix $\mathbf{A}$ aufgestellt. Sie enthält die Funktionswerte für jede Funktion f_i ausgewertet an den gemessenen x_j :

$$\mathbf{A} = \begin{pmatrix} f_1(x_1) & f_2(x_1) & \cdots & f_p(x_1) \\ f_1(x_2) & f_2(x_2) & \cdots & f_p(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(x_N) & f_2(x_N) & \cdots & f_p(x_N) \end{pmatrix} . \quad (3)$$

2. Als Nächstes definieren wir den Spalten-Vektor

$$\vec{y} = (y_1, y_2, \dots, y_N)^\top , \quad (4)$$

der die y -Koordinaten unserer Messwerte enthält.

3. Die Kovarianzmatrix der Messwerte sei $\mathbf{W}$ und $\mathbf{Z} = \mathbf{W}^{-1}$.
4. Der Parametervektor $\vec{a}$ ergibt sich dann zu:

$$\vec{a} = (\mathbf{A}^\top \mathbf{Z} \mathbf{A})^{-1} \cdot \mathbf{A}^\top \mathbf{Z} \cdot \vec{y} . \quad (5)$$

5. Die Kovarianzmatrix ergibt sich zu:

$$\mathbf{V} = (\mathbf{A}^\top \mathbf{Z} \mathbf{A})^{-1} . \quad (6)$$

Wer sich für die Herleitung interessiert: Blobel-Lohrmann: „Statistische und numerische Methoden der Datenanalyse“ bzw. SMD-Vorlesung. Alternativ auch bei Wikipedia: [https://en.wikipedia.org/wiki/Linear_least_squares_\(mathematics\)](https://en.wikipedia.org/wiki/Linear_least_squares_(mathematics))

Aufgaben

1. Implementiere eine Funktion, die eine Liste von Funktionen und die x - und y -Werte übergeben bekommt und die lineare Methode der kleinsten Quadrate anwendet. Die Funktion soll den Parametervektor als Array von korrelierten `ufloats` zurückgeben.
2. Teste deine Funktion indem du einen Fit der Form

$$\Psi(x) = a_1 \cos(x) + a_2 \sin(x) \quad (7)$$

an die Daten in der Datei `daten.txt` durchführst. Die Unsicherheiten der y_i sollen als $\sigma = 0,1$ angenommen werden.

Tipps

- Matrix-Multiplikation zwischen numpy arrays mit dem `@` Operator
- Zum invertieren einer Matrix kann `numpy.linalg.inv` genutzt werden.