Laborübung - Versuchsprotokoll "Name des Versuchs"

Max Mustermann
Stefan Laufmann
Martha Musterfrau

Abgabe: 23. April 2010

Inhaltsverzeichnis

1 Vorbereitungsaufgaben

1.1 Regression vs. Interpolation

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

1.2 Unterabschnitt

1.2.1 rechnerisch

Man will eine Gerade der Form $y = a + bx$ an die Messwerte annähern. Damit ergibt sich die Summe der Fehlerquadrate als $E = \sum (y_i - a - bx_i)^2$. Um E zu minimieren, muss man die partiellen Ableitungen gleich null setzen.

$$\frac{\partial E}{\partial a} = -2 \sum (y_i - a - bx_i) \stackrel{!}{=} 0$$

und

$$\frac{\partial E}{\partial b} = -2 \sum x_i (y_i - a - bx_i) \stackrel{!}{=} 0$$

Nach Lösen des linearen Gleichungssystems erhält man:

$$\begin{aligned} a &= \frac{\sum x_i \sum x_i y_i - \sum x_i^2 \sum y_i}{(\sum x_i)^2 - n \sum x_i^2} \\ b &= \frac{\sum x_i \sum y_i - n \sum x_i y_i}{(\sum x_i)^2 - n \sum x_i^2} \end{aligned}$$

Setzt man nun die Messwerte ein, ergeben sich $a = 0,37$ und $b = 1,16$.

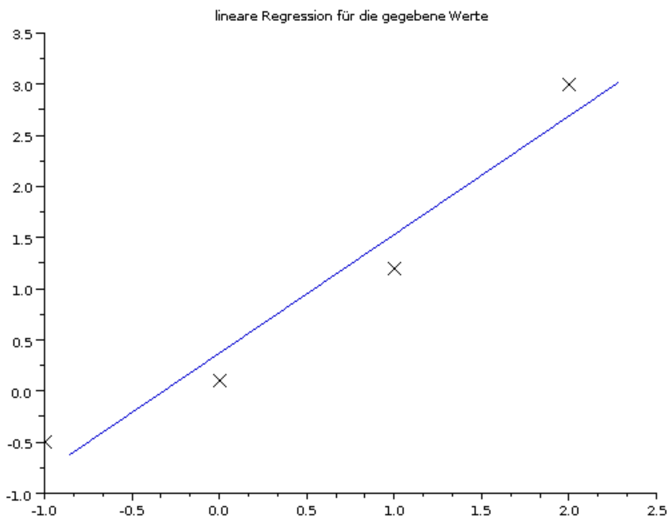
1.2.2 mit Scilab

Die scilab-Funktion "regress(x,y)"führt für zwei Vektoren x und y eine lineare Regressionsanalyse mithilfe der Methode der kleinsten Fehlerquadrate durch.
Als Koeffizienten ergeben sich für die gegebenen Werte:

$$\begin{aligned} a &= 0,37 \\ b &= 1,16 \end{aligned}$$

Daraus folgt für die Näherungsgerade:

$$y = 0,37 + 1,16 \cdot x$$



1.3 Unterabschnitt

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Die Vorgehensweise für dieses Verfahren ist wie folgt:

1. Länge h_i zwischen den Stützstellen $[x_i, x_{ir+1}]$ festlegen
2. y_i'' anhand des Gleichungssystems berechnen
3. Koeffizienten a_i, b_i, c_i, d_i für die Ableitungen berechnen
4. Bestimmung der Polynome s_i zu den Intervallen $[x_i, x_{i+1}]$

2 Praktische Aufgaben

2.1 Widerstandsgerade

In dem vorherigen Praktikum haben wir über einen Spannungsteiler die Spannungen über zwei Widerstände bestimmt. Dabei war uns ein Widerstand als Präzisionswiderstand mit $R_I = 100\Omega$ gegeben und der zweite Widerstand R_x war unbekannt. Bei anliegender Wechselspannung wurde eine Messreihe für die beiden Spannungen erstellt. Die gemessene Spannung U_x in Abhängigkeit zu U_i ist in Abbildung 1 dargestellt.
Zu diesen gegebenen Messwerten wollen wir nun eine lineare Regression durchführen.
Für das Verhältnis der Widerstände zu den gemessenen Spannungen gilt:

$$\frac{U_x}{U_i} = \frac{R_x}{R_i}$$

und es ergibt sich

$$U_x = \frac{R_x}{R_i} U_i$$

Zur Regression eignet sich somit die Funktion

$$U_x = c\,U_i$$
$$\text{wobei } c = \frac{R_x}{R_i}$$

Die Summe der Fehlerquadrate ist

$$E = \sum (U_{x\,i} - c\,U_{i\,i})^2$$

Wenn $\frac{\partial E}{\partial c} \stackrel{!}{=} 0$, ergibt sich $c = 0,977$. Die enststande Regressionsfunktion ist in Abbildung 2 dargestellt.
Somit erhält man mit $R_x = c\,R_i$ einen Widerstandswert von

$$R_x = 97,7\Omega$$

Dieser Wert stimmt mit dem zuvor berechneten empirischen Mittelwert überein.

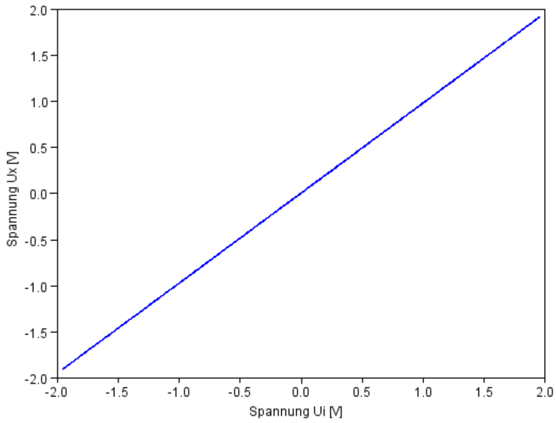


Abbildung 1: Spannung U_x in Abhängigkeit von U_i

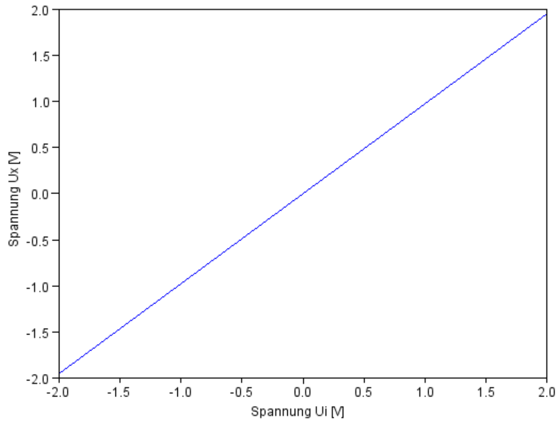


Abbildung 2: Regressionsfunktion der Spannungen

2.2 Blackbox-Kennlinie

2.2.1 Messung

Sieht man sich den Spannungsverlauf der Blackbox zu einer sinusförmigen Eingangsspannung mit dem Oszilloskop an, so erkennt man einen abgeschnittenen, sowie gestauchten, positiven Teil eines Sinussignals (siehe auch Skizze in Abbildung 3).

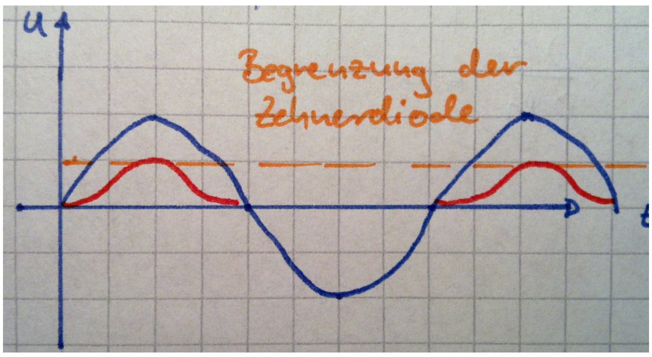


Abbildung 3: Sinus-Spannung U in Abhängigkeit von t

Im XY-Modus des Oszilloskops lässt sich nun die Kennlinie der Schaltung (Abbildung 4) darstellen. Aus dieser lässt sich eine sinnvolle Verteilung der ≥ 20 Messpunkte zwischen $-15V$ und $15V$ ermitteln: Man erkennt grafisch, dass die Werte unterhalb von $0V$ eine Gerade ergeben, eine Kurve zwischen $0V$ und $3V$ entsteht und der Spannungswert anschließend saturiert. Daraus ergibt sich unsere Einteilung für 25 Messungen und die tabellarisch dargestellten Messwerte der Messung mit dem Multimeter:

- $[-15V; 0V]$: $5V$-Schritte
- $]0V; 3V[$: $0,3V$-Schritte
- $[3V, 15V]$: $1V$-Schritte

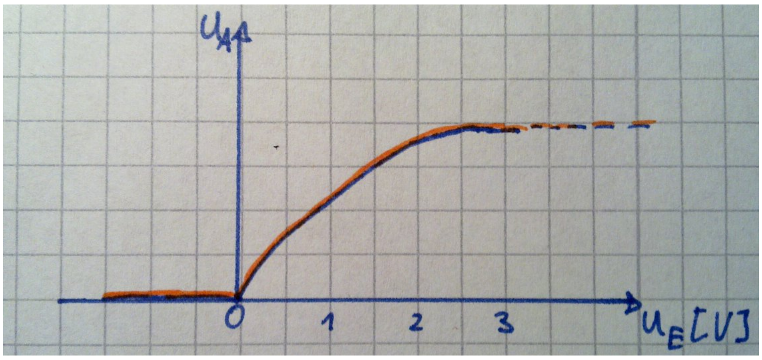


Abbildung 4: Kennlinie mit U_A über U_E

$U_E[V]$	$U_A[V]$	$U_E[V]$	$U_A[V]$
-15	0	Weiterführung...	
- 5	0	4	1.50
0	0	5	1.74
0.3	0.00037	6	1.90
0.6	0.03695	7	2.02
0.9	0.15	8	2.11
1.2	0.28	9	2.19
1.5	0.42	10	2.25
1.8	0.57	11	2.31
2.1	0.71	12	2.36
2.4	0.85	13	2.40
2.7	0.99	14	2.44
3.0	1.12	15	2.47

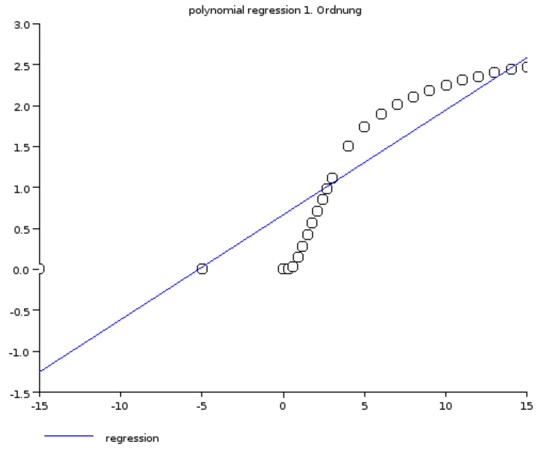
2.2.2 Regressionspolynome

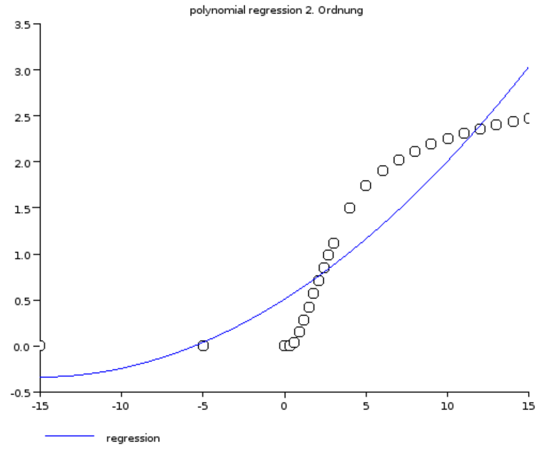
Der folgende Scilab-Code zur Interpolation erzeugt Regressionspolynome mit Hilfe der Funktion *mdt_regression*:

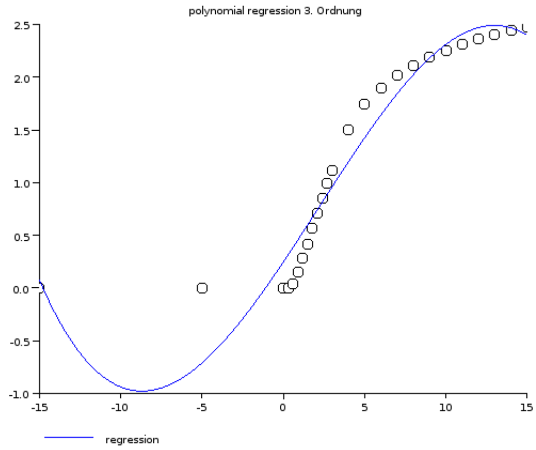
```
// Regression für verschiedene Polynomgrade
// durchführen und die Plots abspeichern
for i=1:6
    filename=["polyRegress", string(i)]
    txt=strcat(filename)
    scf(i)
```

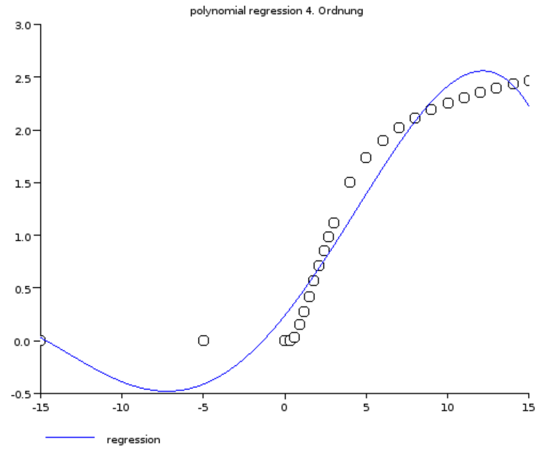
```
    mdt_regression(U_E,U_A,i);
    xs2png(i, txt);
end
```

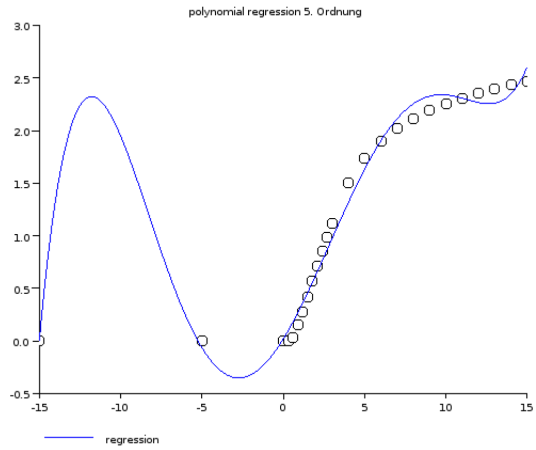
Erzeugt werden folgende Ergebnissplots der Polynomregressionen für Polynomgrade von eins bis sechs:

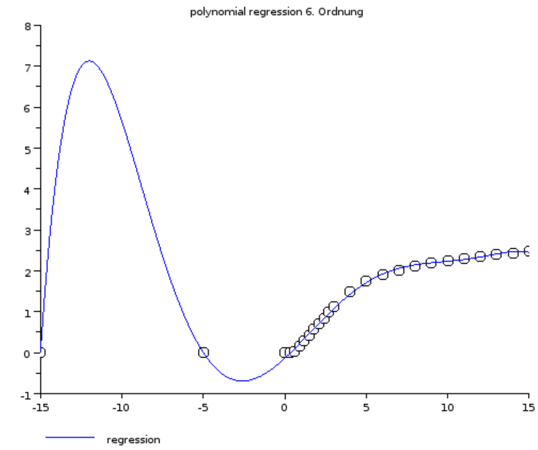












Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

2.2.3 Kubische Splines

Mit folgendem Scilab-Code kann man unter Zuhilfenahme der Funktion *mdt_kubicspline* die kubischen Spline-Funktionen des gesamten Datensatzes (Abbildung 5) und des halben Datensatzes (Abbildung 6) erzeugen.

```
// Kubischen Spline ueber Gesamtdatensatz und Plots speichern
scf(0);
mdt_kubicspline(U_E,U_A);
xs2png(0, strcat(["../images/kubicspline_full"]));

// Kubischen Spline ueber dem halben Datensatz und Plots speichern
U_E=[-15,0,0.6,1.2,1.8,2.4,3.0,5,7,9,11,13,15];
U_A=[0,0,0.03695,0.28,0.57,0.85,1.12,1.74,2.02,2.19,2.31,2.40,2.47];
scf(1);
mdt_kubicspline(U_E,U_A);
xs2png(1, strcat(["../images/kubicspline_half"]));
```

Die Graphen vergleichend, funktioniert die Interpolation durch kubische Splines offensichtlich besser als mit Regressionspolynomen. Dennoch besteht auch hier, allerdings mit abnehmender Messdatenzahl, das Problem des Überschwingens (vgl. Abb. 6).

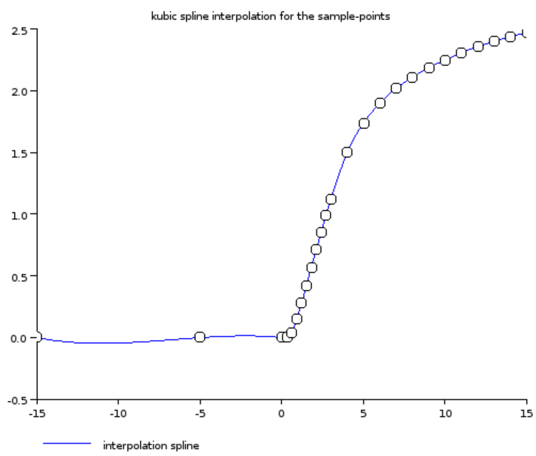


Abbildung 5: Kubische Spline über gesamten Datensatz

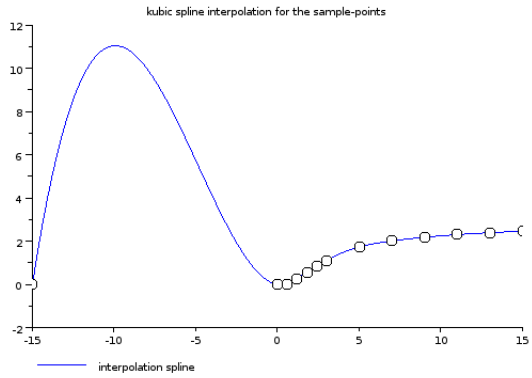


Abbildung 6: Kubische Spline über halben Datensatz