

Fitten eines verrauschten Polynoms

Schreiben Sie ein MATLAB-Script `polyrausch.m`, die folgende Aufgaben erfüllt:

1. Gegeben ist ein Polynom `p_test` und `sigma`, mit deren Hilfe Sie verrauschte Testdaten berechnen:
2. Erzeugen Sie einen x -Vektor $x = [-5 \dots 5]$ mit 10 Werten und entsprechenden y -Werten. (`linspace`)
3. Verrauschen Sie die y -Daten, indem Sie einen gleichverteilten Zufallsvektor, mit der Varianz σ^2 (`sigma * (rand(size(x))-0.5)`), zum y -Vektor addieren. Damit die automatische Überprüfung funktioniert, sollten Sie den Zufallszahlengenerator in diesem Script nur einmal verwenden.
4. Erzeugen Sie eine Grafik, die das verrauschte y als Funktion von x darstellt.
5. Benutzen Sie `errorbar` für der Fehlerbalken. Die Datenpunkte sollen dabei nicht mit Linien verbunden sein. Die Marker sollen Grün sein. Das Markersymbol soll `*` sein. Als Werte für die Fehler Δy verwenden Sie `sigma`.
6. Fitten Sie ein Polynom `p_fit` gleicher Ordnung mit `polyfit`.
7. Zeichnen Sie zuerst das gefittete Polynom in roter Farbe mit `plot` und `y_fit=polyval(p_fit,x_fit)`.
8. Zeichnen Sie dann das gegebene Polynom in blauer Farbe mit `y_test=polyval(p_test,x_fit)`.
9. $x_{fit} = [2 \cdot \min(x) \dots 2 \cdot \max(x)]$ und 100 x_{fit} -Punkten (`linspace`).
10. Schalten sie den `grid` für die Grafik ein.
11. Beschriften Sie die Achsen und der Grafik mit `y` und `x`.
12. Betiteln Sie die Grafik mit `Testen des Polynomfits`
13. Erstellen sie eine Legende mit `Testdaten`, `Fitpolynom` und `Testpolynom`.
14. Experimentieren Sie mit den Werten für `p_test` und `sigma`.
15. Vor der Beispielabgabe sollten Sie dann die Definitionen von `p_test` und `sigma` auskommentieren.

Hinweis:

Das erstellen der `Legende` für `errorbars` funktioniert so:

```
he1=errorbar(...);  
he2=plot(...  
...  
legend([he1,he2,...], 'errorbar-beschriftung', 'plot-beschriftung..', ...);
```

Hinweis:

Vergessen Sie nicht, die Definitionen von `p_test` und `sigma` vor der Abgabe auszukommentieren.

Hinweis:

Eigentlich wäre die Verwendung von Gausverteilten Rauschen `randn` statt `rand` realistischer, da spielt die automatische Überprüfung aber noch nicht mit.

Gesucht: Script `polyrausch.m`

Anschauungsbeispiel:

