1 Prüfung - Applikationssoftware und Programmierung

- Erlaubt ist jegliche Benutzung Ihrer Unterlagen, Ihrer Übungsbeispiele und der Unterlagen am Web.
- Verboten ist während der Prüfung jedoch der Austausch von Files, E-mails und ähnlichem mit anderen Studierenden oder mit der Aussenwelt.
- Die Dauer des schriftlichen Teils sollte drei Stunden nicht überschreiten.
- Anschließend an den schriftlichen Teil findet ein mündliches Einzelgespräch mit einer Dauer von ca 15-20 Minuten statt. Die Reihung erfolgt nach dem Zeitpunkt der Abgabe.
- Die Abgabe erfolgt wie bei der Übung mit Hilfe des Skripts `pruefungsabgabe`.
- Bitte geben Sie fertiggestellte Beispiele ab und programmieren Sie dann die weiteren. Dies erleichtert die Korrektur erheblich.

1.1 Matrix

1. Schreiben Sie eine Matlab-Funktion `countmat.m`, die mit folgendem Aufruf

   ```
   [c,r] = countmat(m,v)
   ```

   funktioniert. Die Inputgrößen sind eine beliebiege Matrix `m` und ein Vektor `v` (Default `[-1,0,1]`). Der Vektor `v` enthält Intervallgrenzen für die folgenden Intervalle

$$
\begin{aligned}
m &< v_1 && : k = 1 \\
v_1 \leq m &< v_2 && : k = 2 \\
v_2 \leq m &< v_3 && : k = 3 \\
& && \vdots \\
v_{\ell-1} \leq m &< v_\ell && : k = \ell \\
v_\ell \leq m & && : k = \ell + 1
\end{aligned}
$$

wobei ℓ die Länge des Vektors `v` ist. Ein Vektor der Länge ℓ definiert also $\ell + 1$ Intervalle. Der Outputvektor `c` soll nun an der k-ten Position die Anzahl der Zahlen in `m` enthalten, die im jeweiligen Intervall liegen. Die Matrix `r` soll gleich groß wie die Matrix `m` sein und an jeder Position die jeweils richtige Intervallnummer k enthalten (logische Indizierung, `for`-Schleife über die Intervalle, ...).

2. Probieren Sie die Funktion mit Matrizen Ihrer Wahl aus.

1.2 Legendre Polynome

1. Schreiben Sie eine Matlab-Funktion `legendrec.m`, die mit folgendem Aufruf

```
r = legendrec(x,n)
```

die Legendre Polynome $P_k(x)$ der Ordnungen $k = 0$ bis $k = n$ für alle Werte von x berechnet. Die Matrix `r` soll dabei in ihren Spalten die Resultate für $P_0(x)$, $P_1(x) \ldots P_n(x)$ enthalten.

2. Diese Polynome werden durch folgende Vorschrift erzeugt

$$\begin{aligned} P_0(x) &= 1 \ , \\ P_1(x) &= x \ , \\ kP_k(x) &= (2k-1)xP_{k-1}(x) - (k-1)P_{k-2}(x) \ , \quad k \geq 2 \ . \end{aligned}$$

Dies ist eine sogenannte Rekursionsformel, bei der $P_2(x)$ aus $P_0(x)$ und $P_1(x)$ berechnet werden kann, u.s.w. für $P_3(x)$, $P_4(x)$, $\ldots$.

1.3 Lineares Fitten

1. Schreiben Sie ein Matlab-Skript `legendrefit.m`, welches einen linearen Fit mit Legendre Polynomen $P_k(\cos\phi)$ durchführt.

2. Speichern Sie dafür die Datei `legendrefit.dat` in Ihrem Arbeitsverzeichnis (`!pruefungsdaten`). Sie enthält in der ersten Spalte ϕ-Werte und in der zweiten Spalte die jeweiligen Messdaten.

3. Stellen Sie das nötige Gleichungssystem für $n = 5$ auf und bedenken Sie dabei, dass die Spalten der Matrix nicht $P_k(\phi)$ sondern $P_k(\cos\phi)$ enthalten müssen.

4. Lösen Sie das Gleichungssystem.

5. Berechnen Sie mit den so erhaltenen Koeffizienten

$$y(\phi) = \sum_{k=0}^{5} a_k P_k(\cos\phi) \ ,$$

wobei hier angenommen ist, dass die erhaltenen Koeffizienten im Vektor a gespeichert sind.

6. Plotten Sie die Funktion zusammen mit den Daten. Fügen Sie Achsenbeschriftung, Titel, Legende ein. Die erhaltene Kurve sollte dabei exakt durch alle Datenpunkte gehen.

1.4 Funktion

1. Schreiben Sie eine Matlab-Funktion `fp1.m`, die für beliebige Arrays x und das Skalar α folgende Funktion berechnet:

$$f(x,\alpha) = \begin{cases} \cos x & : \quad |x| \leq \pi \\ -\exp\left(-(|x|-\pi)^2/\alpha\right) & : \quad |x| > \pi \end{cases} \ .$$

2. Schreiben Sie ein Matlab-Skript `sfp1.m`, welches die Funktion ausprobiert und einen Plot für 2 verschiedene α-Werte erzeugt (Beschriftung, …).