Übungen zur Vorlesung Modellierung

Hans G. Feichtinger und Team
Wintersemester 2011/12

Teil 1 Stand: November 25, 2011

Allgemeine Hinweise

Es ist wesentlich, sich "in der Welt umzusehen" und eigene Ideen für kleine Simulationen oder mathematische Modelle zu entwickeln.

We will use the abbreviation OML in order to indicate that the text (usually in verbatim-mode) is Octave/MATLAB (pseudo)-code.

For the convenience of English-speaking participants and for the training of local students the examples/exercises are given in English:

Übungsmaterial

1. Multiplication of polynomials at the coefficient level can be realized using the command `conv(a,b)`. Use it to multiply long numbers! (e.g. 12345678^2). Just for training verify in this way the well-known formula $(x-1)(x+1) = x^2 - 1$. Observe that multiplication does not depend on order. Write a short test-file comparing `fliplr(conv(a,b)) = conv(fliplr(a),fliplr(b))` for random sequences. Note: `fliplr(1:4) = [4,3,2,1]`. This implies in particular that it does not matter whether one chooses the basis $\{t^n, \ldots, t, 1\}$ (MATLAB convontion) or the usual increasing order $\{1, t, t^2, \ldots, t^n\}$ for the representation of polynomial functions. CONV always delivers the coefficients of the product matrix.

2. Find different ways to calculate the binomial coefficients $\binom{n}{k}$, as in

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^k b^{n-k}.$$

   What is the range of numbers integers (n, k) for which you can make use of your approach? (some methods are rather limited to relatively small input n, k).

3. (almost all square matrices are diagonalizable)
   Given random square matrices (real or complex), one finds that most/few are invertible (confirm by an experiment), or normal (i.e. satisfies $\boldsymbol{A} * \boldsymbol{A}' = \boldsymbol{A}' * \boldsymbol{A}$). Are the also diagonalizable?

Consider, that $[V, D] = eig(\boldsymbol{A})$ provides (hopefully) a basis of eigen-vectors (columns V), and a diagonal matrix D such that $V * D == \boldsymbol{A} * \mathbf{V}$, or $V * D * V^{-1} = \boldsymbol{A}$.

Interpret the following OML-commands:

```
for jj=1:100;A=rand(5);[V,D]=eig(A);ch(jj)=norm(A*V-V*D);end;plot(ch);
```

4. Try to understand the behaviour of the random variable: sum of several dices (start with 2,3 or 5 dices, but consider also the possibility of simulating 20 dices at once, be it in parallel [20 dices at once] or serial [20 times throughing a single dice and summing up]. In each case these are independent events, both in a natural context or in a strict mathematical sense). (cf. course on probability theory). Produce the corresponding histograms.

   GERMAN: Man versuche das Verhalten der Würfelsumme von mehreren Würfeln experimentell zu verstehen, d.h. man bestimme durch numerische Simulation die entsprechenden Histogramme.

Material for the week Oct. 24-29th

5. Assume that you want to deposit a certain sum (say 100 EUROs) with a bank, for 6 years. Assume that you have a variable interest rate in the range of $1,5\% - -3\%$. What it the maximal outcome after 6 years, what is the minimal outcome, what it the distribution of possible outcomes if one allows the interest rate to vary randomly within this interval, on a monthly basis? Is it relevant whether the interest rate is higher at the beginning and goes down towards the end (because then the higher saving early on are achieved earlier (?)) or is it irrelevant? (give a reasoning for your answer).

6. Simulate a Brownian Motion on the number line $\mathbb{Z}$ in different variations: for example let the "particles" start in 0 with differnt probabilties to the left, to the right or "stay at home". Typical cases: a coin toss should decide whether to go to the left or to the right or equal probability of 1/3 for staying at the initial position or move the left or to the right neighbour. What can you say (experimental or statistical) by reason of experiments? How does the result change if both sides have different probabilities?

   How large is the probability approximately for beeing on the positive side after 50 iterations if the probability to go right is 2/3 (hence the probability to go left eqals 1/3)?

   Courageous can tackle the corresponding simulations $2-$dimensional and move left/right respectively up/down (with equal or different probability).

   Fassung auf Deutsch: Man simuliere eine Brownsche Bewegung auf der Zahlengeraden $\mathbb{Z}$, in verschiedenen Variationen: Beispielsweise lasse man die "Partikel" von

0 startend mit unterschiedlichen Wahrscheinlichkeiten nach links, rechts oder “daheim” bleiben. Typische Fälle: Münzwurf soll entscheiden ob man nach links oder nach rechts geht, oder gleiche Wahrscheinlichkeit 1/3 für eine Bewegung von der Ausgangsposition, zu verweilen oder zum linken oder rechten Nachbarn zu wandern. Was kann man (experimentell bzw. statistisch) aufgrund von Experimenten sagen. Wie ändert sich das Ergebnis, wenn die beiden Seiten unterschiedliche Wahrscheinlichkeit haben.

Wie gross in etwa ist die Wahrscheinlichkeit, dass man nach 50 Wiederholungen auf der positiven Seite ist, wenn die Wahrscheinlichkeit nach rechts zu gehen 2/3 ist (und also die, nach links zu gehen, dementsprechend 1/3).

Mutige können entsprechende Simulationen 2−dimensional angehen, und nach links/rechts bzw. oben/unten wandern lassen (mit gleicher oder unterschiedlicher Wahrscheinlichkeit).

7. Consider the multiplication of numbers in different number systems. recall that the decimal expression 123 is the same (by definition) as `polyval([1,2,3],10) ==` $1 \cdot 10^2 + 2 \cdot 10^1 + 3 \cdot 10^0$. In a similar way one can work with binary systems. The sequence $cc = [1, 1, 0, 1, 0, 1]$ represents the decimal number `polyval(cc,2)` which equals 53. Find out how once can find the converse, i.e. a representation of a natural number using a special digit (e.g. $k = 10$, or $k = 2$ or $k = 6$). Test the correctness by applying $polyval(cc, k)$ to the result, and show that multiplication can be expressed with these methods.

8. How can you find out, what the probability of reaching by at least 100 points by throwing the dice 16 times is, if the special rule is applied, that a **6** implies that one is allowed to throw the dice once more (for simplicity no further repetition is assumed!).

```
n= 360; dic6=zeros(1,n);dic6(2:6)= 1/6; dic6(8:13) = 1/36;
dic666 = real(ifft( fft(dic6).^6)); sum(dic666(101:n)),
```

9. (complementary example) Try to figure out how to simulate a team-race, with 8 teams and 6 participants in a running competition. Generate a table of times in the format of an 48×3 matrix with times in the format such as $[1, 12, 34]$ for 1 hour, 12 minutes and 34 seconds (for an individual participant).

   Let the different teams have slightly different strength (athletic quality) and also different variance (the effective times obtained by a random experiment should be considered as a realization of a random process).

   After setting up the experiment try to check that sometimes the “stronger team” can beat the weaker team. Try to argue that e.g. for a given expectation value the observed situation may depend on the chosen variances. So far a relatively informal procedure is OK.

   Hint: Conversion from the given format into seconds can be carried out using the command `polyval(aa,60)`!

Material for the week Nov. 1-6th

10. What does the command `vander()` do in Octave/Matlab? The Vandermonde matrix describes the point evaluation of a given polynomial. Write down the structure of the matrix and how would the interpretation change if one deletes some of the columns? Check the literature (e.g. Wikipedia) for a formula for the determinant of the matrix. When is it invertible?

11. Assume the following setting: We get measurements and we want to fit them using polynomials. For simplicity we are only given 7 values of the function f we do not know (i.e. $f(x_i) = y_i, \quad i = 1, \cdots, 7$). Furthermore the measurements could contain some noise.

    Try to fit polynomials of different order to the points given below. Hint: use the command `vander()` to compute the Vandermonde matrix. Taking only certain columns of it gives the right matrix to compute the corresponding linear system with. Use the pseudo-inverse (`pinv()`) to solve the problem.

    The values were approximately taken from the function $f(x) = xe^{-x}$. Which orders give a good approximation?

x_i	y_i
0.0176	0.0168
0.1514	0.1309
0.3554	0.2528
0.6789	0.3443
1.0589	0.3669
1.5444	0.3296
2.1001	0.2548
2.6206	0.1879
3.4085	0.1112

12. Repeat 10 for the point set below. In which sense are these measurements better/worse than the ones from above? Compare the condition numbers and results obtained in the previous exercise.

x_i	y_i
0.0107	0.0109
0.5453	0.3178
0.6157	0.3345
0.7354	0.3502
0.8901	0.3659
1.0519	0.3669
1.3263	0.3541
1.4809	0.3345
3.4226	0.1112

13. Given an invertible matrix $A \in \mathbb{R}^{n\times n}$. Show the following equation for the condi-

tion numbers:

$$\kappa(A) = \kappa(A^{-1})$$

Hint: Use the singular value decomposition of a matrix. You should not use Octave/Matlab to solve this exercise.

14. In preparation for the discussion of spline-functions: Recall the definition of the standard box function $bx(t) := \mathbf{1}_{[-1/2,1/2]}$, the indicator function of $[-.5, .5]$, which has $\boldsymbol{L}^p$-norm equal to 1 for $1 \leq p \leq \infty$.

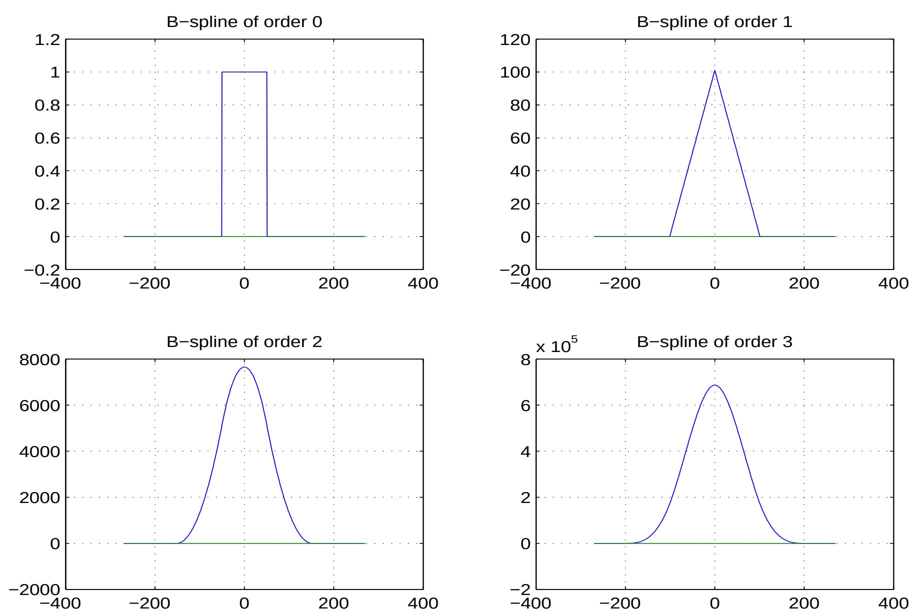
    The convolution of two integrable functions f, g on $\mathbb{R}^d$ is given by

    Definition 1.

$$f * g(x) := \int_{\mathbb{R}^d} g(x-y)f(y)dy = \int_{\mathbb{R}^d} T_y g(x) f(y) dy = \left(\int_{\mathbb{R}^d} T_y g f(y) dy \right)(x), \quad (1)$$

    In our case the functions depend only on one variable and therefore $\mathbb{R}^d$ can be replaced by $\mathbb{R} = (-\infty, \infty)$.

    Define the **B-spline of order** k for for $k \geq 1$ recursively in the following way: $B_0(t) = bx(t)$, and $B_{k+1} = bx * B_k$. A discrete variant looks like this:



    Task: Calculate explicitly B_1 and B_2 (try B_3) and demonstrate in this way that it is a piecewise polynomial function (of degree k), which is smooth of order $k-1$, i.e. $k-1$-time continuously differentiable.

15. A simple initial example for quadratic spline-functions, i.e. functions which coincide with a polynomial of fixed degree (we use degree 2) and continuously differentiable at the node.

Let us consider a quadratic spline function, given by $\varphi(t) = p_1(t)$ on $I_1 = [-1, 0]$ and $\varphi(t) = p_2(t)$ on $I_2 = [0, 1]$.

Is it possible to find for arbitrary values $\alpha, \beta, \gamma \in \mathbb{R}$ a solution $\varphi(t)$ on $[-1, 1]$, with $p_1(t), p_2(t) \in \mathscr{P}_2(\mathbb{R})$ and $p_1(-1) = \alpha, p_1(0) = \beta = p_2(0), p_2(1) = \gamma, p_1'(0) = p_2'(0)$ and $p_1'(-1) = 0$ and $p_2'(+1) = 0$.

16. Find for the unit roots of order 4 and a given sequence of values $\mathbf{d} = [d_1, \ldots, d_4]$ a cubic polynomial, with the given values at the positions $1, i, -1, -i$. What happens, if we prescribe values $d_1, \ldots, d_8$ at the unit roots of order 8? (using the PINV method).

17. (*) What can be said in the situation of exercise 15, if not the values at the integer points $-1, 0, 1$ are given, but rather other values. Is the resulting (quadratic) spline-function (same side constraints as before) uniquely determined (over/under-determined?). Give one set of constraints, such that there exists a unique solution to the problem.

Material for the week Nov. 21-28th

18. Recall the formulas for $B_2(x)$ computed in exercise **??**. Write an Octave function `b2` that evaluates the spline at a given point vector x. It should allow evaluations of the following form:

```
xx = rand(1,1000)*3;
yy = b2(xx);
plot(xx,yy);
```

Plot the written function `b2` and its integer translates on the interval $I = [0, 3]$. For which integer values $z \in \mathbb{Z}$ is the function $B_2(x - z) \neq 0$ on the interval I?

19. (Exercise 18 continued) Given a number of data points on the interval I of the form $(x_i, d_i)_{i=1,\ldots,N}$ for some given N. We are now looking for the B-Spline function of the form

$$f_B(x) = \sum_{j \in \mathbb{Z}} c_j B_2(x - j), \tag{2}$$

that minimizes the following term

$$\sum_{i=1}^{N} (d_i - f_B(x_i))^2. \tag{3}$$

Rewrite this problem in matrix form to obtain a problem of the form

$$\sum_{i=1}^{N} (d_i - (A \cdot c)_i)^2. \tag{4}$$

How does the matrix A look like?

20. (Exercise 18 continued) We fix the function $f(x) = xe^{-x}$ on the interval I. Ten random sampling values can be taken like this:

```
f = @(x) x.*exp(-x);
x = rand(1,10)*3;
xx = linspace(0,3,1000);
d = f(x);
hold on;
plot(xx,f(xx));
plot(x,d,'x');
hold off;
```

Write a function that computes the vector c from the previous exercise using the pseudo-inverse. The function should allow input of the following form (x, d should be the random sampling values of the function f):

```
coeff = bsplappr(x,d);
```

Furthermore the function should plot the resulting approximating function

$$f_B(x) = \sum_{j \in \mathbb{Z}} c_j B_2(x - j). \tag{5}$$

To do this evaluate the function in the points `xx = linspace(0,3,1000)`.

21. For the (possibly non-uniform) knot sequence $[t_1, \dots, t_4]$ compute the B-Spline $b_{1,2}$ explicitly using the De-Boor recursion formula

$$b_{j,0}(t) := \begin{cases} 1 & \text{if} \quad t_j \le t < t_{j+1} \\ 0 & \text{otherwise} \end{cases}, \qquad j = 1, \dots, 3 \tag{6}$$

$$b_{j,n}(t) := \frac{t - t_j}{t_{j+n} - t_j} b_{j,n-1}(t) + \frac{t_{j+n+1} - t}{t_{j+n+1} - t_{j+1}} b_{j+1,n-1}(t), \qquad j = 1, \dots, 3 - n. \tag{7}$$

Material for the week Nov. 29- 4.Dez.

FFT und Anwendungen

22. In the course we have heard that the FFT (foward Fourier Transform) is in principle the same as the mapping from coefficients of a polynomial with respect to the MATLAB-ordered *monomial basis*, i.e. $\{t^{n-1}, \dots, t^2, t, 1\}$ to the values of the corresponding polynomials on the unit roots of order n, starting from $1 = \omega^0$, in the *clockwise (!)* sense. For any given value n (e.g. $n = 4$ for computations by hand, or $n = 16$ or some random natural number for MATLAB experiments):

```
bas = 0 : 1/n : (n-1)/n; u =  exp(- 2 * pi * i * bas);  Vn = vander(u);
F = fft(eye(n));  norm(F -  Vn(:,n : -1 : 1),'fro')
```

should give the "numerical value zero". Note: *'fro'* stands for the Frobenius (or Hilbert-Schmidt) norm of the matrix, which is just $(\sum_{k,j} |a_{j,k}|^2)^{1/2}$. One also finds the following (numerical) equality:

```
a = rand(1,n); norm( polyval(fliplr(a),u) - fft(a)),
```

with the hint, that the *fliplr* command is required due to the choice of the basis for the polynomials. The DFT/FFT is based on the usual convention (starting with $1, t, \dots$) while MATLAB is using the opposite order, hence $\mathbf{a}$ has to be flipped! Try to explain why these equalities hold true.

23. Given $p(x)$ a polynomial with coefficients $\mathbf{a}$, e.g. $p(x) = 2x + 3x^2 + 4x^4$. and $q(x) := p(x^2)$. We want to understand how these function look (compare, are similar, etc.) when considered on the torus $\mathbb{U} = \{z \,|\, |z| = 1\}$, by taking their values over the unit roots of some high order, say $n = 512$, because the FFT allows conveniently to carry out this task. The following experiment may be helpful:

```
>> a = rand(1,5) - 0.5;
>> h = zeros(512,1); h(2:6) = a;  px = fft(h);
>> u = zeros(512,1); u(3:2:12)=a; qx = fft(u);
>> subplot(121); plot(px); axis square;
>> subplot(122); plot(qx); axis square;
>> figure; subplot(121); plot(1:512,px(:));
>> axis tight, subplot(122);
>> plot(1:512,qx(:)); axis tight; figure(gcf);
```

It may help to think, what the squares of all the unit roots of order 512 are.

24. The following plots should be done again both using the plotting routine (which does a plot in the complex plane), or by plotting separately the real and imaginary part of the sequences given below:

```
bas=linspace(0,2*pi,512);xyy=0.4*exp(bas*i)-0.05*exp(21*i*bas);
subplot(121); plot(xyy);axis square;
px = zeros(1,512); px(2)=.4; px(22)=-0.05; y = fft(px);
subplot(122); plot(fft(px)); axis square; figure(gcf);
pause;  % or
plot(xyy); hold on; plot(fft(px),'r'); hold off;
```

What happens if one changes the parameters (e.g. replacing the coefficients or the coordinates, say 22 by 36.

25. Describe the behaviour of a sum of two pure frequencies which are close to each other, with equal or comparable coefficients. You can treat the problem in two different ways (ideally in both, but one approach is OK): Either analytically, using the addition theorems for cos and sin functions, or numerically/experimentally. What you find/see is in fact connected to tuning problems for string instruments or pianos. (Schwebungen).

    One way to get started would be this (recall that `fft(eye(n))` contains exactly the pure frequences, sampled over the unit roots of order n):

    ```
    n = 512;  F = fft(eye(n));
    plot(real( F(34,:) + F(36,:)));  % or
    plot(real( F(37,:) - 0.7*F(40,:)));
    ```

    Try to find out what happens if the concrete choices (e.g. the coordinates $34, 37$, or the amplitudes 0.7 etc. are changed!

Further material

26. (special task): The best way to understand and demonstrate the behaviour would be to demonstrate the effect by e.g. visualizing the situation by saying: We have points (one red, one green) representing vectors on circles, moving at different speeds (lock a classical clock): add those to vectors at any given time resp. describe their (oriented) distance as a function of time! I do not mind if you find such demonstrations in the internet and make them public (to the course members).

27. **Denoising, Practical experiment**: Form a trigonometric polynomial with just a few significant spectral components, i.e. start on the Fourier transform side with some $5 - 15$ (not too high) frequencies which you put (randomly or deterministically) to given values. Using IFFT you can obtain the corresponding signal xx. Then add noise to add, e.g. `xxn = xx + alpha * rand(1,n)` for suitable values of `alpha`. Try to recover the signal (i.e. try to remove the *noise*) from the noisy signal `xxn` by identifying even in the noise signal the relevant components.