THE DERIVE - NEWSLETTER #10

THE BULLETIN OF THE

DERIVE

USER GROUP

Contents:

revised reprint 2006 **June 1993**

D E R I V E
U s e r G r o u p
1993 U.K. M e e t i n g

in the frame of

The Technology in Mathematics Conference
at
The University of Birmingham

Monday, 20th September 1993
10.00 a.m. till 4.00 p.m.

****

ATM ATM ATM ATM ATM ATM ATM ATM

In the autumn this year MicroMath will be running a special feature on the use of computer algebra systems and we have a number of articles specifically about Derive. Non-ATM members who would like to receive a copy should contact Basil Blackwell Publishers, 108 Cowley Road, Oxford, England.

ATM ATM ATM ATM ATM ATM ATM ATM

Announcing a NEW journal:

The International DERIVE Journal,
Reporting on Research and Innovative Teaching

The International DERIVE Journal is published to disseminate information about research and practice on the use of DERIVE as a tool for doing and learning mathematics. The journal aims to enhance the use of DERIVE by reporting on research and significant innovations.

Editor: Prof. John Berry, Centre for Teaching Mathematics, Plymouth, UK

D E R I V E - B O O K - S H E L F

Two Italian DERIVE books:

Manara-Perotti, Lineare a Geometria con DERIVE, McGraw-Hill, Milano 1992

Bacchelli-Lorenzi-Perotti, Matematica con DERIVE, McGraw-Hill, Milano 1992

Mauve/Moos, Mathematik mit DERIVE (Arbeitsblätter zur experimentellen Mathematik), Heidelberg 1993

****

Ein Service der DUG für den deutschsprachigen Raum:

Alle DUG-Mitglieder können den Tagungsband der DERIVE-Konferenz 1992 "Teaching Mathematics with DERIVE" zum Preis von öS 300.- (Österreich), bzw. DM 45.- (Deutschland, Schweiz, ...) beim Herausgeber beziehen. Ein Anruf oder eine kurze Nachricht genügt.

Liebe DERIVE Anwender!

Sie können in dieser Ausgabe des DNL neben dem Abschluss der beiden physikalischen Beiträge von M.Horbatsch und E.Ahonen vor allem DERIVE-Anwendungen aus den Gebieten der Wahrscheinlichkeitsrechnung und Statistik finden. In einer der nächsten Nummern des DNL möchte ich Ihnen noch vorstellen, wie ich beliebige diskrete Verteilungen darstelle und wie ich Zufallsvariable erzeuge, die einer vorgegebenen diskreten Verteilung gehorchen.

Herr Pröpper, einer der Autoren dieser Ausgabe, war auch Teilnehmer des DUG-Treffens in Schweinfurt. Ich möchte bei dieser Gelegenheit allen Teilnehmern für ihr Interesse und ihr Engagement danken. Besonderer Dank gilt unserem Freund H.Appel, der durch seinen persönlichen Einsatz sehr viel zum Gelingen des Treffens beigetragen hat. Th. Weth aus Würzburg hat versprochen, eine Serie über die bekanntesten algebraischen und transzendenten Kurven für den DNL zu schreiben. Die erste Folge dieses „Kurvenlexikons“ wird im DNL#11 - über die Kissoide und das Delische Problem - erscheinen.

Auf der Heimfahrt von Schweinfurt sprachen B. Wadsack aus Wien und ich über DERIVE und seine Möglichkeiten. Wir bedauerten, dass es nicht möglich sei, die Koordinatenachsen zu beschriften - oder doch? So hatten wir die Idee, dieses Problem zu "derivisieren". Kaum zu Hause angekommen, schon probiert. Auf Seite 35 können Sie eine kleine Utility als Ergebnis meiner Bemühungen finden.

Ich habe für meinen persönlichen Gebrauch in der Schule eine DERIVE-Unterrichtseinheit, bestehend aus fünf Arbeitsblättern über die Sinus-Schwingung (Kreisfrequenz, Periodenlänge, Phasenverschiebung,...) zusammengestellt. Falls Sie Interesse an diesem Papier haben, melden Sie sich bitte bei mir, ich sende Ihnen gerne eine Kopie.

An dieser Stelle ein paar Zeilen über DERIVE 2.55. Der Zeileneditor (Author ...) ist mit den Cursortasten bedienbar. (Zwischen Editor Mode und dem Subexpression Mode kann umgeschaltet werden). DERIVE-Schirme können als Grafikdateien (im TIF-Format) abgespeichert werden. Die schon lange gewünschte Möglichkeit, DERIVE Grafiken als Hardcopy über den Drucker auszugeben wurde in 2.55 realisiert.

Zuletzt möchten wir noch A.Kozubik zur Geburt seiner Tochter gratulieren. Ich wünsche Ihnen allen einen schönen Sommer (auf der Nordhalbkugel) und einen angenehmen Winter (auf der südlichen Hälfte unserer Erde).

Bis zum Herbst (Frühjahr)
mit den besten Grüßen Ihr

Dear DERIVE Users,

In this issue you will find the last part of both physical contributions (M.Horbatsch and E.Ahonen) and mainly applications on Statistics and Probability theory. In one of the next DNLs I will publish a contribution how to plot graphs of discrete distributions and how to produce random variables which are following any given discrete distribution.

Mr Pröpper, one of the authors of this issue, was participant of the German DUG-Meeting in Schweinfurt. I have to thank all participants for their cooperation and interest. My special thank is for our friend H. Appel because his personal efforts helped to make this meeting a success. Mr Th. Weth from Würzburg has promised to write a series about interesting algebraic and transcendental curves for the DNL. So in DNL#11 you will find the first part of this "Lexicon of Curves". (The cissoid and its importance for the problem of Deli).

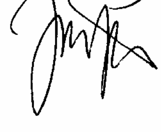
On the ride back from Schweinfurt B. Wadsack and I talked about ..... DERIVE and its possibilities. We were sorry, that it is not possible to designate the axes in DERIVE, or could it be???. Immediately grew an idea. Having arrived at home I turned on the PC and tried to create the letters x and y and the arrowheads. You can find a little utility as my result of our ideas on page 35.

For my use in school I've prepared a teaching unit about the sine function (amplitude, period, translation of the phase,...) with a size of 5 pages. If you are interested in these worksheets then I'll be glad to send them to you if you would ask for.

Now I want to tell you some things about DERIVE 2.55: Heureka, there is a line editor (the arrow keys move the cursor in the Author line, you can toggle between Line Edit Mode and Subexpression Mode). DERIVE-screens can be stored as TIF-files for further use in text processing or desktop publishing. Graphics images can be sent to the printer (with or without the menu) with a keystroke...

At last I'd like to felicitate A.Kozubik on the birth of his daughter Susane. I wish you all a nice summer (northern hemisphere) and a calm winter (southern hemisphere).

Until autumn (spring time)
with my best regards



The *Derive-News-Letter* is the Bulletin of the Derive-User Group. It is published at least three times a year with a contents of 30 pages minimum. The goals of the *D-N-L* are to enable the exchange of experiences made with Derive as well as to create a group to discuss the possibilities of new methodical and didactic manners in teaching Mathematics.

Editor: Mag. Josef Böhm
A-3042 Würmla
D´Lust 1
Austria
Phone/FAX: 43-(0)2275/8207
e-mail: nojo.boehm@pgv.at

Contributions:

Please send all contributions to the above address. Non-English speakers are encouraged to write their contributions in English to reinforce the international touch of the *D-N-L*. It must be said, though, that non-English articles will be warmly welcomed nonetheless. Your contributions will be edited but not assessed. By submitting articles the author gives his consent for reprinting it in the *D-N-L*. The more contributions you will send to the Editor, the more lively and richer in contents the Derive-News-Letter will be.

Preview: (Contributions for the next issues)

Bisection with DERIVE, D.M.Dyer, USA
Fluid Flow in DERIVE, Reuther a.o., BRA
Newton´s Chaos, Graphic Integration, J. Böhm, AUT
Computer Aided Mathematics in School, K.H.Keunecke, GER
Continued Fractions, R. Setif, FRA
DERIVE in Hawaiian Classrooms, E. Sawada, USA
Stability of Systems of ODEs, A. Kozubik, Slovakia
Minimization of a „Flat Function", Lopez, Portugal
Digital Filter Design Using DERIVE, Hood, UK

(DNL#11 will be published September 1993)

Impressum:
Medieninhaber: *DERIVE* User Group, A-3042 Würmla, D'Lust 1, AUSTRIA
Richtung: Fachzeitschrift
Herausgeber: Mag.Josef Böhm
Herstellung: Selbstverlag

M. Hornschuh, Hilter, D

Ich suche *DERIVE*-User im Raum Osnabrück zum Erfahrungsaustausch.
Meine Telefonnummer: 05424 69370

P. Steiner, Klagenfurt, A

Can you tell me, how DERIVE calculates the values for SIN(x), COS(x), EXP(x) and LN(x)?

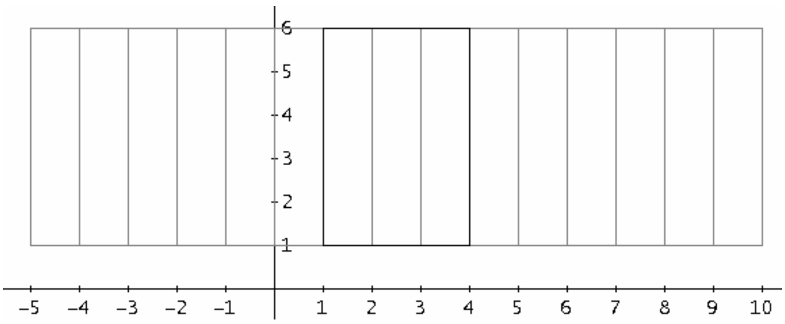
DNL: I hope to get an answer from Hawaii.

R. Mauve, Hirschhorn, D

1. Komplexere Grafiken im R^2 werden erst geplottet, wenn man die, sie erzeugenden Listen vereinfacht hat (Simplify oder approX). Das führt zu erheblichen Problemen bei der Erzeugung umfangreicherer Zeichnungen. Z.B. zeichnet bei geeigneter Definition von VERSCHIEBE, RECHTECK

```
VECTOR(VERSCHIEBE(RECHTECK([1,1],[4,6]),[n,0]),n,-6,6,2)
```

trotz einiger "Bemühungen" nichts. Aber nach Vereinfachung mit Simplify ergibt sich:



Wunsch: DERIVE sollte ohne Vereinfachen ("unausgerechnete Ausdrücke") plotten.

(Wish: DERIVE should plot directly without Simplifying the expressions; it is not able to plot complex graphs without simplifying or approximating the generating lists).

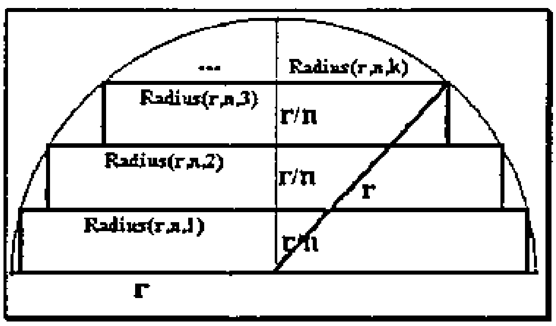
This problem has been resolved since long. Don´t forget to activate Options > Simplify before Plotting or Options > Approximate before Plotting in the Plot Window. Otherwise you will receive an error message:

2. Wir berechneten den Kreisinhalt. (We calculated the area of a circle):

$$\#1: \quad \text{radius}(r, n, k) := \sqrt{r^2 - \left(\frac{k \cdot r}{n}\right)^2}$$

$$\#2: \quad \text{area}(r, n, k) := \frac{2 \cdot \text{radius}(r, n, k) \cdot r}{n}$$

$$\#3: \quad \text{appr\_area}(r, n) := \sum_{k=1}^{n} \text{area}(r, n, k)$$



```
#4:   Precision := Approximate

#5:   Notation := Decimal

#6:   [2·appr_area(1, 10), 2·appr_area(1, 100), 2·appr_area(1, 1000), 2·appr_area(1, 10000)]

#7:   [2.904518326, 3.120417031, 3.139555466, 3.141391477]
```

#8: $2\cdot appr\_area(r, 10000) = 3.141391477\cdot r^2\cdot SIGN(r)$

#9: $\lim_{n\to\infty} appr\_area(1, n) = 2\cdot\lim_{n\to 0+} n\cdot\sum_{k=1}^{FLOOR(1/n)} \sqrt{(1 - k^2\cdot n^2)}$

#10: $\lim_{n\to\infty} appr\_area(r, n) = 2\cdot r^2\cdot SIGN(r)\cdot\lim_{n\to 0+} n\cdot\sum_{k=1}^{FLOOR(1/n)} \sqrt{(1 - k^2\cdot n^2)}$

In 1993 the limit was 0.

R. Wadsack, Vienna, A

There is a little trick to accelerate both the Cursor in text windows and the Cross in Plot-windows. You could include this command into any batch file.

```
mode con rate = 32 delay = 1
```

Marko Horbatsch, North York, CAN

Dear DERIVE'rs
particularly dear MATH teachers,

It is really quite refreshing to see how active high school math teachers are in Europe by trying to do more interesting material in the classroom using DERIVE. Here in North America (I spent 10 years in Toronto, Canada by now after doing all my studies in Frankfurt up to Ph.D. in theoretical physics) matters aren't quite as bright as far as student preparation for university is concerned. While doing some work on Computer Assisted Instruction (CAI) for university kids, I thought that some of this might be useful for high school students as well.

I am working on a set of course notes that will hopefully be published in not more than a year's time. Here is a little demonstration of how to use DERIVE to demonstrate functions of a complex variable. The idea that functions like sin and cos can be analytically continued in the complex plane is not evident even to some university math students. The lack of intuition about this should be easily overcome by plotting sequences of z- and f(z)-points in separate screens (screens 2 and 3 respectively). Discrete point plotting was preferred over a continuous plot to show how each point z maps onto its f(z). For plotting purposes the following DERIVE objects were created.

line 1: CV(z) generates an [x,y] vector from a complex number;

line 3: ZV(r,th) generates a circle of radius r that will contain a parameter th(eta) to allow parametric plots in th(eta) once the radius r is specified.

line 2: defines our function of interest that has poles at $z = \pm i$.

The three choices of circles in the z-plane z = 0.5, 1, 2 produce the very intriguing images f(z) and show the blow-up near the pole locations.

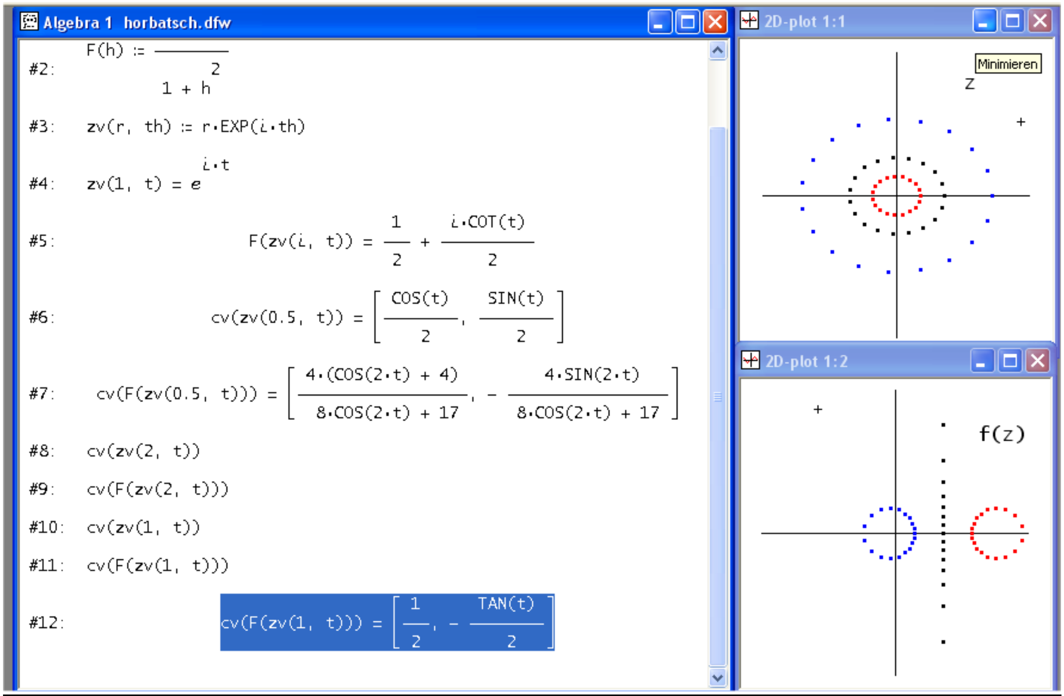
I am convinced that your clever high school students will love this! This subject area has received attention lately with the interest in maps for producing fractal sets.

We have also developed some stand-alone CAI programs for linear algebra, complex variables and Newtonian mechanics. They will run on any EGA/VGA type PC with 256 kB of memory. You can obtain the whole package by sending a cheque over $ 20 (or Eurocheque for DM 30) to cover distribution costs. Unlimited copying/use of the programs is granted.

Specify the media type (5.25"(360/1.2M) or 3.5"(720/1.4M)).

If you want more detailed into first, just write to me:

Prof. Marko Horbatsch
Physics & Astronomy
YORK University, 4700 Keele St
NORTH YORK, Ont. M3J 1P3
C A N A D A

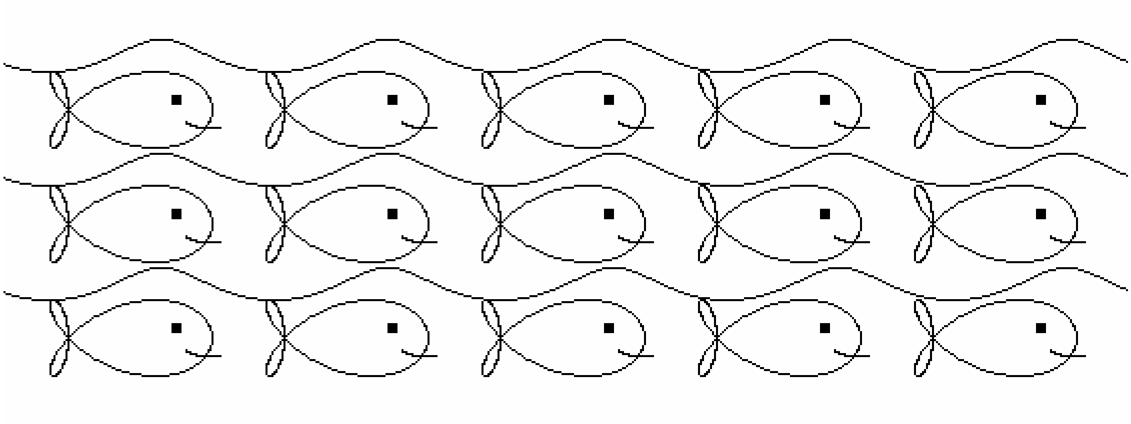


E. Sawada, Hawaii, USA

I once attended a DERIVE session where David Stoutemyer was presiding. He drew - see the graph - using parametric equations. I used his idea and extended it. I used vectors to initiate vertical and horizontal translations. The following program illustrates this translation. Aloha.

```
VECTOR(VECTOR([[COS(x)*(COS(x)+1)+k,SIN(2*x)+m],
[2+1/2*COS(x/6-2.4)+k,1/2*SIN(x/6-2.4)+m],[3/2+k,1/4+m],
[SEC(SIN(x))+m]],k,-6,6,3),m,-3,3,3)
```

DNL: Try this nice plot!



An idea came to me when I was working with translation. If we can translate about some given point, why not translate about some given function. The following program will illustrate how we can move a function f(x) about another function g(x) using parametric equations.

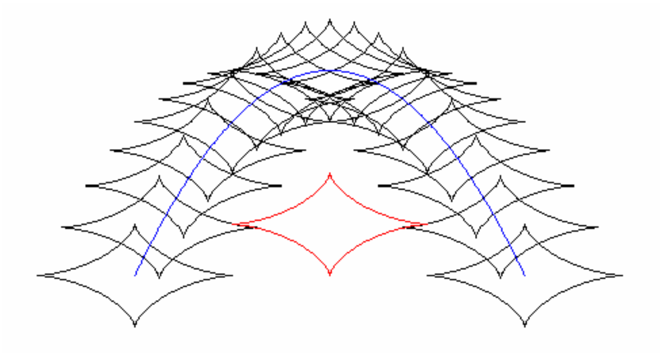
Moving a hypocycloid of four cups about a parabola:

Hypocycloid:

```
[SIN(A)^3,COS(A)^3]
```

Parabola:

```
[t, 3 - t^2]
```



```
VECTOR([SIN(A)^3+t,COS(A)^3+3-t^2],t,-2,2,0.25)
```

Solving Physics Problems in the Classroom with DERIVE 2.0, Part 2

Marko Horbatsch, York, Ontario

Another attempt to make the differential equation – eigenvalue problem more understandable to physics undergraduates is through the solution of the differential equation as an initial problem with the eigenvalue as a trial parameter. In particular, it is emphasized how only by the choice of discrete energy eigenvalues one can obtain localizable eigenfunctions.

The stationary Schrödinger equation with a quartic potential is solved for a selected eigenstate using the Runge-Kutta algorithm supplied in one of the support files. Any other symmetric potential could be substituted instead of the quartic one and higher eigenstates can be solved for as well with this technique. The aim of the demonstration is to show the following: if one solves the equation as an initial-value problem starting at $x = 0$ with appropriate conditions for a symmetric (or alternatively antisymmetric) solution and treats the eigenvalue as a trial parameter, one can satisfy the requirement of normalizability, dictated by physics, only for very special energy values (the eigenvalues of the Hamiltonian). Schrödinger´s equation in one dimension (cf. eq. (3)) is converted to a pair of first order equations (in units where $\hbar = m = 1$):

$$\phi' = \chi \ , \ \ \chi' = 2(V(x) - E)\ \phi \ . \qquad (5)$$

This makes the problem solvable by a standard technique for systems of first order differential equations. The coding of the equations is shown below, the functions ϕ and χ are denoted by `y` and `yp`.

```
#1:   ODEApproximation.mth -- Copyright (c) 1990-2003 by Texas Instruments Incorporated

                                        c1 + (   lim    p) + 2·(c2 + c3)
                                             v→u_ + c3
#2:   RK_AUX3(p, v, u_, c1, c2, c3) := ─────────────────────────────────
                                                     6

      RK_AUX2(p, v, u_, c1, c2) := RK_AUX3(p, v, u_, c1, c2,      lim      p)
#3:                                                           v→u_ + c2/2

      RK_AUX1(p, v, u_, c1) := RK_AUX2(p, v, u_, c1,      lim      p)
#4:                                                 v→u_ + c1/2

      RK_AUX0(p, v, v0, n) := ITERATES(u_ + RK_AUX1(p, v, u_, lim  p), u_, v0, n)
#5:                                                            v→u_

#6:   RK(r, v, v0, h, n) := RK_AUX0(h·APPEND([1], r), v, v0, n)

                     4
#7:   V(x) := 0.5·x

#8:   RHS_(y, yp, et) := [yp, 2·(V(x) - et)·y]

#9:   (RK(RHS_(y, yp, 0.5), [x, y, yp], [0, 1, 0], 0.1, 50))↓↓[1, 2]

#10:  (RK(RHS_(y, yp, 0.6), [x, y, yp], [0, 1, 0], 0.1, 50))↓↓[1, 2]
```

The Runge-Kutta routine `RK(r,v,v0,h,n)` contains in its argument list the derivative of the solution vector (`r`), the vector of the dependent variables itself (`v`), the initial condition (`v0`), the fixed step-size (`h`) and the number of steps (`n`). Note that v contains the independent variable as well.

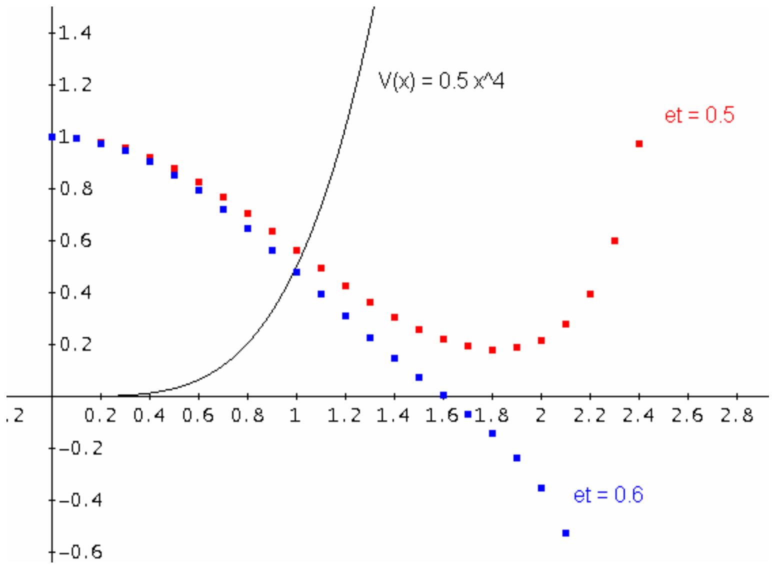


Fig. 3 Numerical solution of the stationary Schrödinger equation for the ground state of a quartic potential as an initial value problem with the energy as a trial value. Upper sequence of points: $E_{t1} = 0.5$, lower sequence: $E_{t2} = 0.6$. The correct ground state eigenfunction is nodeless, decreases to zero asymptotically and has an eigenvalue of $E_{t1} < E_0 < E_{t2}$.

Derive´s Runge-Kutta algorithm provides an example of a task that can be broken up into a few sequential steps and is thus ideally suited for this environment. It also shows that within a symbolic language one can handle automatically exceptions, where a function itself may not exist at a point needed for the calculation, but a limit can be obtained and is sufficient for the calculation (see lines #2 – #6). The main reason for the use of the `LIM(f(x),x,a)` function which calculates the limit of $f(x)$ as x approaches a, is, however, the lack of a command for substitution and LIM in its first attempt just tries that.

Note: Recent versions of DERIVE provide a SUBST-function, so it is not necessary to use the LIM-function.

```
                                      c1 + SUBST(p, v, u_ + c3) + 2·(c2 + c3)
#1:   RK_AUX3(p, v, u_, c1, c2, c3) := ───────────────────────────────────────
                                                         6
```

Another subject area that is being included in modern physics curricula is that of discrete maps. Figure 4 demonstrates that DERIVE 2.0 represents an ideal tool in order to study these. The relatively simple logistic (or quadratic) map is used as an example to show how radically the behavior of a map can change as an order parameter (in this example r) is varied. There are applications of this map in many cases of physics (a model of turbulence, voltage-current characteristics in certain semiconductors, etc.) and other sciences [5]. Derive´s way of allowing one to look both at the numbers and graphs and the fact that one has to look at the numbers and see how they were generated allows one to improve the understanding of this currently very fashionable subject.

For r-values up until ≈ 2.9 the map is not too exciting: after some number of iterations the (n+1)st iterate of

$$x_{n+1} = r\, x_n\, (1 - x_n) \tag{6}$$

essentially has forgotten what its starting value x_0 was. (This fact itself is worth observing in Derive). Fig. 4 shows as a function of r the last 10 iterates of a sequence with typically 50 points. Up to $r \approx 2.9$ these last 10 iterates practically fall on top of each other, such that they are displayed as a single point. For $2.9 < r < 3.4$ one finds such sequences show bistability for large n, i.e. independent of the starting value x_0 they will oscillate between two values, which, again, are only determined by the value of r. What happens in the vicinity of $r_1 \approx 2.9$ is called a pitchfork bifurcation and it can be seen that this phenomenon repeats itself at $r_2 \approx 3.44$ at both parts of the fork and the sequence will oscillate between four values. This behavior of further bifurcation continues repeating itself, but is happening faster as r increases. A study of the r_n – values at which bifurcations occur leads to a critical value r_∞, beyond which no regularity can be observed (chaos) and a law which relates all bifurcation points to a fundamental constant (the Feigenbaum number δ):

$$r_n = r_\infty - \frac{const.}{\delta^n}\,. \tag{7}$$

From the generated sequences of points one can obtain an estimate of the Liapunov exponent, which can be defined as the limit of the expression given in line 4 as $\varepsilon \to 0$ and $n \to \infty$. The convergence of the exponent with respect to both parameters could be studied. The sign of the Liapunov exponent can be used as a measure of the fact whether the behavior is chaotic or regular.

The details of the implementation shown below provide examples of three ways to generate the iteration with the functions `FUN` (line 2), `FUN1` (line 3) and `FUN0` (line 10) depending on whether one wants pairs of iteration index and function value, just the last iterate or simply a sequence of the iterates. The function `BIFURC` in line 13 takes a sequence generated for given *r* and provides a vector of points to plot the last 10 elements at given *r*. To produce figure 4 one has to generate sequences for the various *r*-values, simplify line 13 and plot.

#1: $\mathrm{lmap}(r, x) := r \cdot x \cdot (1 - x)$

#2: $\mathrm{FUN}(r, x0, n) := \mathrm{ITERATES}([v_1 + 1, \mathrm{lmap}(r, v_2)], v, [0, x0], n)$

#3: $\mathrm{FUN1}(r, x0, n) := (\mathrm{ITERATE}([v_1 + 1, \mathrm{lmap}(r, v_2)], v, [0, x0], n))_2$

#4: $$\mathrm{liap}(r, x0, n, \varepsilon) := \frac{\mathrm{LN}\left(\left|\dfrac{\mathrm{FUN1}(r, x0 + \varepsilon, n) - \mathrm{FUN1}(r, x0, n)}{\varepsilon}\right|\right)}{n}$$

#5: $$\mathrm{xbarr}(w) := \frac{1}{\mathrm{DIM}(w)} \cdot \sum_{i\_=1}^{\mathrm{DIM}(w)} w_{i\_}$$

#6: FUN(1, 0.1, 20)

0	0.1
1	0.09
2	0.0819

. . . . . .

#8: FUN1(1, 0.1, 20)

#9: 0.03207763711

#10: FUN0(r, x0, n) := ITERATES(lmap(r, v), v, x0, n)

#11: w := FUN0(1, 0.1, 20)

#12: w := [0.1, 0.09, 0.0819, 0.07519239000, 0.06953849448, 0.06470289227, 0.06051642800, 0.05685418994, 0.05362179103, 0.05074649455, 0.04817128784, 0.04585081487, 0.04374851764, 0.04183458485, 0.04008445236, 0.03847768904, 0.03699715648, 0.03562836689, 0.03435898637, 0.03317844642, 0.03207763711]

#13: $\mathrm{BIFURC}(w, r, n) := \mathrm{VECTOR}([r, w_i], i, n - 9, n)$

#14: BIFURC(w, 1, 20)

1	0.04817128784
1	0.04585081487
1	0.04374851764

#16: $\mathrm{BIFURC\_}(r, x0, n) := \mathrm{VECTOR}([r, (\mathrm{FUN0}(r, x0, n))_i], i, n - 9, n)$

#17: VECTOR(BIFURC_(a, 0.1, 20), a, 1, 5, 0.1)

#18: VECTOR(BIFURC_(a, 0.1, 20), a, 1.05, 4.95, 0.1)

#19: VECTOR(BIFURC_(a, 0.1, 20), a, 1.025, 4.975, 0.1)

#20: VECTOR(BIFURC_(a, 0.1, 20), a, 1.075, 4.975, 0.1)

I added expressions #16 through #20 to make the plotting procedure easier, Josef

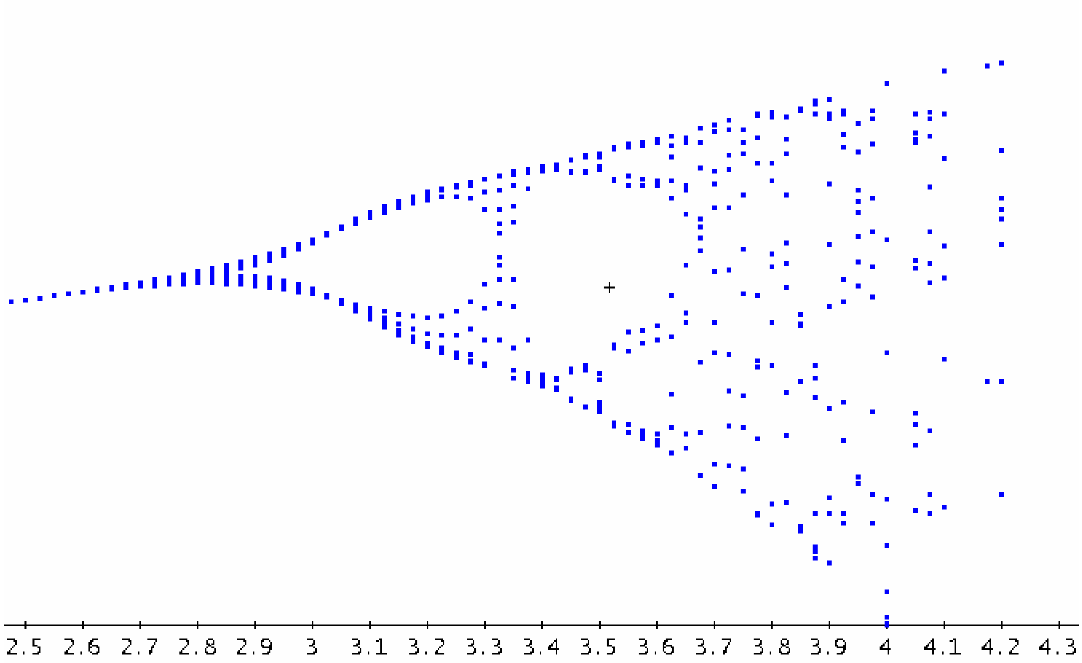


Fig. 4 Iteration of the logistic map as a function of the order parameter *r*. The display shows bifurcations in the accumulation points as *r* passes through certain critical values. At $r_\infty \approx 3.57$ chaotic behavior (loss of regularity) can be observed.

Another important topic in the upper year undergraduate curriculum is the solution of partial differential equations. Most equations discussed in the classroom context are separable and thus reduced to ordinary equations that can be handled with Derive. In particular, Fourier representations of the solutions are commonly used and Derive´s capabilities to show the result are given in the last example. To prepare the students for situations where this technique does not work one has to study numerical techniques. Finite difference methods are easily applied to elliptic problems, such as Poisson's or Laplace´s equation and the resulting systems of linear equations can be solved in Derive. For time evolution problems various combinations of finite difference representations of the temporal and spatial derivatives lead to either explicit or implicit methods [6]. Explicit methods result in recursion relations for many coupled parameters that represent the value of the unknown function at discrete points in space-time.

The example shown in figure 5 deals with the numerical solution of the diffusion equation in one dimension:

$$\frac{\partial}{\partial t} f(x,t) = c \frac{\partial^2}{\partial x^2} f(x,t)\,. \qquad (8)$$

This equation has many applications in the sciences, e.g., fort he description of heat conduction. Here it serves merely the purpose of demonstration as to what can occur in the discretization by an explicit finite difference algorithm. Replacement of the time derivative by a forward difference and of the spatial second order derivative by a central difference approximation [6] results in the recursion relation ($f_{i,j} = f(x_i, t_j)$):

$$f_{i,j+1} = f_{i,j} + r\left(f_{i+1,j} + f_{i-1,j} - 2f_{i,j}\right). \qquad (9)$$

Here the independent variables x and t have been made dimensionless by appropriate scaling ($0 \le x \le 1$, $t \ge 0$) and the new diffusion constant r has also absorbed a $\Delta t/\Delta x^2$ factor coming from the finite difference operators. After imposition of the usual boundary condition $f_{i,0} = f_{N,0} = 0$ and choice of an initial condition such as, e.g., $f(x, t = 0) = x(1 - x)$ the recursion (9) can be solved.

```
#1:   n := 5

                        1
#2:   xvek := VECTOR(———————·i, i, 0, n + 1)
                      n + 1

#3:   U0(x) := - x·(x - 1)

      uinit := VECTOR(U0(xvek ), i, n + 2)
#4:                          i

      UPLUS(u, i, n_) :=
        If i < n_
#5:        u↓(i + 1)
           0

      UMINUS(u, i) :=
        If i > 1
#6:        u↓(i - 1)
           0

      RHS_(u, r) := VECTOR(IF(i < n + 2 ∧ i > 1, u  + r·(UPLUS(u, i, n + 2) +
#7:                                                i

        UMINUS(u, i) - 2·u ), 0), i, n + 2)
                          i

#8:   solution(r, m) := ITERATES(RHS_(u, r), u, uinit, m)

      plot(w, j) := VECTOR([xvek , w   ], i, n + 2)
#9:                             i   j,i

                                   y2 - y1
#10:  cutline(x1, y1, x2, y2) := ———————————·(x - x1) + y1
                                   x2 - x1

                                    y2 - y1
#11:  cutline(x1, y1, x2, y2) := (———————————·(x - x1) + y1)·χ(x1, x, x2)
                                    x2 - x1

      cutl(a, b) := cutline(a , a , b , b )
#12:                          1   2   1   2

               6
#13:  VECTOR(  Σ  cutl((plot(solution(0.5, 10), j)) , (plot(solution(0.5, 10),
              i=1                                   i

        j))       ), j, 1, 10, 2)
           i + 1

               6
#14:  VECTOR(  Σ  cutl((plot(solution(1, 5), j)) , (plot(solution(1, 5), j))
              i=1                                i                           i +

        ), j, 5)
      1
```

The first four statements serve to define a mesh and to set up the initial condition on this mesh. Lines 5 and 6 define functions that will return the right hand and left hand neighbors of a vector element provided that the index i is not at the boundary in which case zero is returned. The special handling of the endpoints is not necessary for the implementation of (9) as given on line 7, as the boundary condition is incorporated there directly. It is merely included to demonstrate a simple use of the `IF`-function, which is a new addition in *DERIVE* 2.0.

`RHS_(u,r)` defines the updating for the vector of solution points, that is it defines equation (9). `solution(r,m)` generates the iteration for given constant r and number of steps m. `plot(w,j)` is a function which generates a vector of points representing the solution at time-level j. In order to plot straight line segments between these points the function `cutline` is defined. It makes use of Derive´s intrinsic `CHI`-function which returns one for $x1 < x < x2$ and zero elsewhere. Shown subsequently is a statement which upon simplification leads to a vector of functions that consists of straight line segments connecting the solution points. A single plot call to the result provides the graphs shown in the windows of figure 5. It is shown that multiple loops can be generated in Derive, but it is also clear that such deep nesting of calls require carefulness of the programmer.

The plot windows show such attempted solutions to (8) with 5 steps and $r = 1$ (right window) and using 10 steps and $r = 0.5$ (left window) for the same time domain. It turns out that this instability has nothing to do with round-off, as this calculation remains in the rational numbers, which Derive handles exactly. The discrete solution points have been connected by straight lines for convenience. Parabolic interpolation would have been more consistent with the second order differencing in space, but would have required more sophisticated programming for display purposes. The instability is seen to not appear immediately, but only after the solution begins to deviate from the originally parabolic form. The reason for this behavior is due to the fact that initially the spatial finite operator does not produce any discretization error.

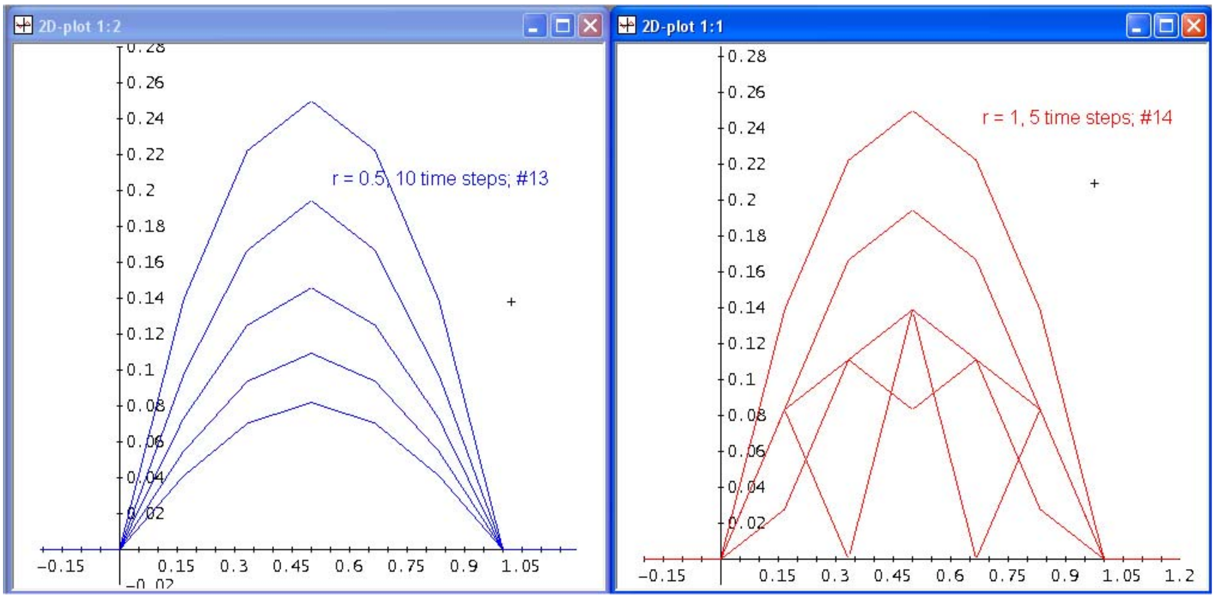


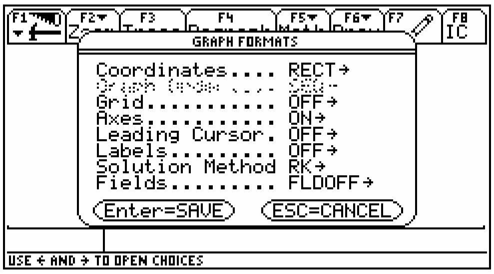
Fig. 5 Numerical solution of the dimensionless diffusion equation in one dimension by an explicit finite difference scheme. The calculation is done in exact precision rational number arithmetic and displays an instability of the algorithm for $r > 0.5$. Window left: $r = 0.5$ and 10 time-steps taken, solutions for every second time-step are displayed in descending order; window right: same time-steps displayed, but calculated within 5 steps ($r = 1.0$).

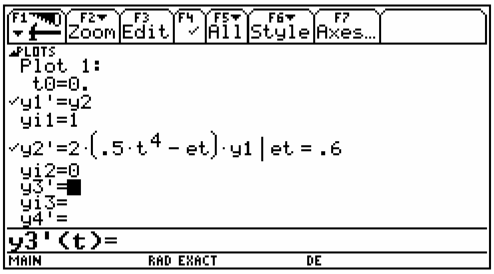
Marko Horbatsch
Department of Physics and
Applied & Computational Mathematical Science Programme
York University, Toronto, Ontario, M3J 1P3, CANADA

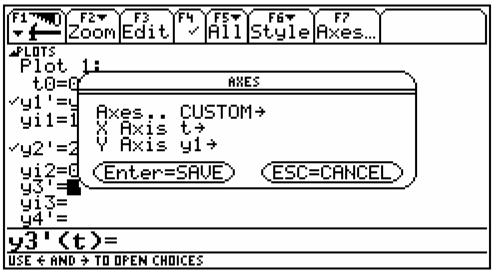
Physics Problems on the TIs

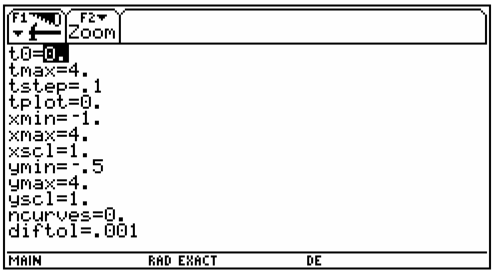
It is not too difficult to solve the system of differential equations on the TIs – Runge-Kutta-method is implemented. What one has to do is to rename the variables: the derivatives are y1 and y2 and the independent variable x has to be changed to t.

In the DE-mode choose the settings as shown below and enter the two equations together with the initial conditions:

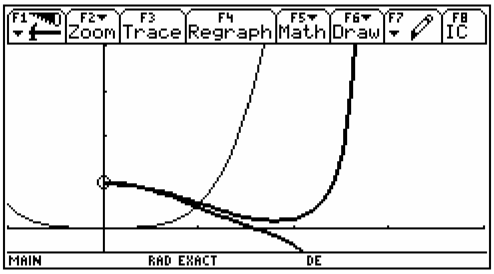








As one cannot plot simultaneously two solutions (for et = 0.5 and 0.6) one has to save the graphs and superimpose them. The graph of function $0.5x^4$ can be added via F6.

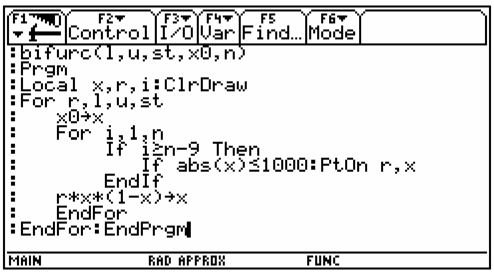


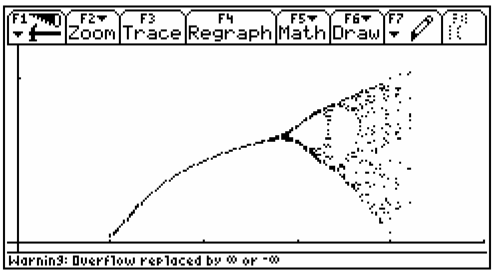
Recommended websites for treating DEs with the TI:

www.seg.etsmtl.ca/ti/edgraphe.pdf (in French!) recommended by Michel Beaudin

www.math.armstrong.edu/ti92/de/de.html recommended by Nils Hahnfeld

A short program shows the bifurcation on the TI-graph screen:





The Quantum Motion

E. Ahonen, Salo, Finland

1: "PLOTEXAMPLE / DERIVE V.1.22"

2: "r,θ: u = dr/dθ: a = du/dθ "

3: "a = -u^-1 √(d^2 - u^2)^-5"

4: "u = ±√(d^2 - (7 θ)^(2/7))"

5: $\int \sqrt{d^2 - (7\,\theta)^{2/7}}\, d\theta$

6: "Simplify #5: the result"

7: $\int - \sqrt{d^2 - (7\,\theta)^{2/7}}\, d\theta$

8: "Simplify #7: the result"

9: "Sum: #6 + #8: the result"

10: "Substitute d=7 in #9: the result"

11: "Try rect. plot in #10:"

12: "Cross:x=2 y=0; Scale:x=1 y=1"

13: "Try polar plot in #10:"

14: "Cross:x=0 y=0; Scale:x=5 y=5"

15: "Plot from 0 to 5*117649"

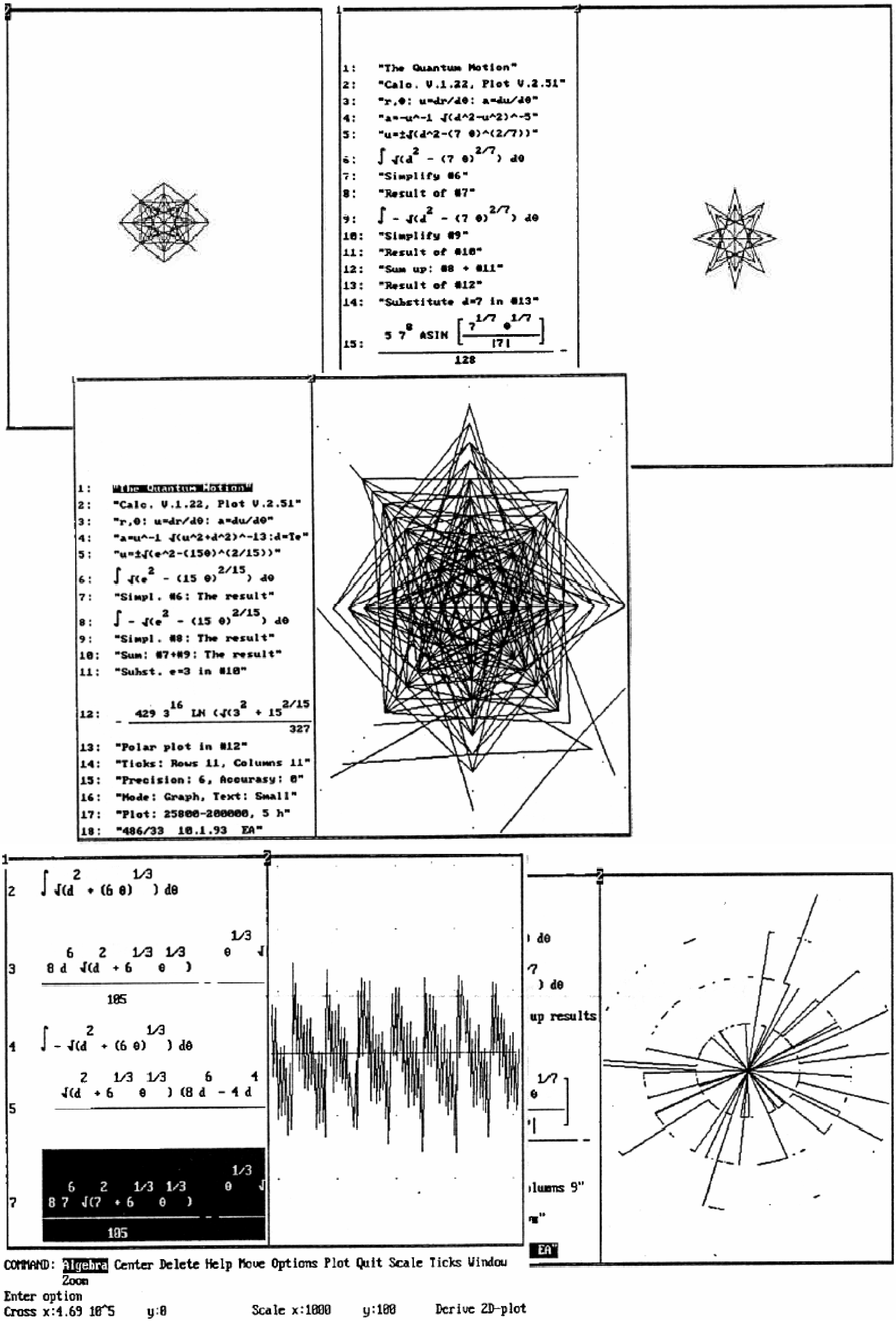
16: "Plot 30 000-100 000; Acc. 0-7"

17: $\int \pm \sqrt{d^2 + (7\,\theta)^{2/7}}\, d\theta$

18: $\int \pm \frac{1}{d^2 \pm (8\,\theta)^{1/4}}\, d\theta$

Simpl. and Plot #17;#18 as earlier; V.2.51 simpl. and plot #18 Plotted values are compressed; variations of settings of Place, Time, Ticks values, Accuracy, Precision, and so on, change the plotted figures.
Equations can be multiplied, divided, etc, by another equation.
Plotts tend to expand as wave from the centre.

Original screen shots from DERIVE Versions 1.22 and 2.51



THE QUANTIZATION OF MATHEMATICS 1 & 2

2: "a-a0(u/u0)^p √(d^2±(u/u0)^k)^n=0"

3: "a=d^2r/dΘ^2=du/dΘ: u=dr/dΘ"

4: "a0=-1:u0=1:p=-1:k=2:n=-7:± neg."

5: "du/dΘ=-u^-1√(d^2-u^2)^-7"

6: $\int - u \sqrt{(d^2 - u^2)}^{7}\, du$

7: $\dfrac{(d^2 - u^2)^{9/2}}{9} = \Theta$

8: "real solutions of #7"

9: $u = \sqrt{(d^2 - (9\,\Theta)^{2/9})}$

10: $u = - \sqrt{(d^2 - (9\,\Theta)^{2/9})}$

11: $\int \sqrt{(d^2 - (9\,\Theta)^{2/9})}\, d\Theta$

12: "the result of #11"

13: "substitute d=5 in #12"

14: "the result of #13"

15: "rect. plot in #14:cross x:2,y:0:"

16: "scale x:1 y:2"

17: $\int - \sqrt{(d^2 - (9\,\Theta)^{2/9})}\, d\Theta$

18: "the result of #17"

19: "subst. d=5 in #18"

20: "the result of #19"

21: "plot #20 as earlier"

22: "no mirror image in plotting"

23: "#14 + #20"

24: "the result of #23"

25: "plot #24 as earlier"

26: "polar plot #24: cross r:0,Θ:0"

27: "scale x:5,y:5"

28: "plot #24 from 0 to 28500-100000"

29: "plot some time"

2: "a-a0(u/u0)^p √(d^2±(u/u0)^k)^n=0"

3: "a=d^2r/dΘ^2=du/dΘ: u=dr/dΘ"

4: "a0=-1:u0=1:p=-1:k=2:n=-29: ± neg."

5: "a=-u^-1√(d^2-u^2)^-29"

6: $\int - u \sqrt{(d^2 - u^2)}^{29}\, du$

7: $\dfrac{(d^2 - u^2)^{31/2}}{31} = \Theta$

8: $u = \sqrt{(d^2 - (31\,\Theta)^{2/31})}$

9: $u = - \sqrt{(d^2 - (31\,\Theta)^{2/31})}$

10: $\int \sqrt{(d^2 - (31\,\Theta)^{2/31})}\, d\Theta$

11: "the result of #10"

12: $\int - \sqrt{(d^2 - (31\,\Theta)^{2/31})}\, d\Theta$

13: "the result of #12"

14: "#11 + #13"

15: "the result of #14"

16: "subst. #15 d=2"

17: "the result of #16"

18: "rect. plot in #17:"

19: "cross x:2 y:0: scale x:1,y:1"

20: "again:cross x=2*10^-31,y=-4:"

21: "scale x:10^-31,y:1"

22: "polar plot in #17"

23: "cross r=0, Θ=0: scale x=5,y=5"

24: "plot from 30 000-100 000"

25: $\int \pm \dfrac{1}{d^2 + (30\,\Theta)^{1/15}}\, d\Theta$

26: "Simpl. and plot #25."

27: "Change in Eq.(1) parameters and "

28: "simpl. and plot youself."

[Erkki Ahonen: Kappaleen liikkeen kuvaaminen x,t-tasolla,
Halikko 1983, ISBN 951-99465-7-8]

Independent Repeated Experiments

Ales Kozubik, Bratislava

This paper is dedicated to my daughter Susane which was born at the same night when I was typing it.

In this short paper I would like to present one of possibilities of using DERIVE in teaching the Probability Theory. The lectures on Probability Theory have been presented in University of Economics in Bratislava since the school year 1990/91. During the years 1990/91 and 1991/92 the programs Eureka and MathCAD were used and since 1992/93 the DERIVE program is used.

Very important topic of this lectures are the repeated random experiments. In this article I deal with the independent repeated random experiments i.e. if the probability of the random events remains the same at each of repeated experiments

At first I define the function `pnom(k,n)`

```
                              n!
      pnom(k, n) := ─────────────────────
#1:                    DIMENSION(k)
                           ∏          k !
                          i=1          i

                                                 10!
#2:               [pnom([3, 3, 4], 10),  ─────────── ] = [4200, 4200]
                                            3!·3!·4!
```

which gives for any natural number n and any vector $k = [k_1, ..., k_s]$ the polynomial number

$$\binom{n}{k_1, k_2, ..., k_s} = \frac{n!}{k_1!k_2!...k_s!}$$

At first I deal with n independent repetitions of an experiment with two possible results (we can denote them as success and failure). In order to compute the probability that in these n repetitions the success will be set in i-times the Bernoulli formula

$$\binom{n}{i} p^i (1-p)^{n-1}$$

is used, where p is the probability of the success in one experiment. This formula is realized in function `ir(i,n,p)`:

```
                                 i        n - i
#3:   ir(i, n, p) := COMB(n, i)·p  ·(1 - p)
```

The next question which we can answer is the question: What is the most likely count of success during n repetitions of the experiment? The answer is given by formula

$$i \in \langle (n+1) \cdot p - 1; (n+1) \cdot p \rangle$$

realized in the function

```
      mlc(n, p) :=
        If (n + 1)·p - 1 = FLOOR((n + 1)·p - 1)
#4:        [(n + 1)·p - 1, (n + 1)·p]
           FLOOR((n + 1)·p)
```

In case the endpoints of the interval are integers, the function `mlc(n,p)` gives the vector containing both of them and in the opposite case it gives the integer contained in the interval.

The function

```
      LNR(α, p) :=
        If LN(1 - α)/LN(1 - p) = FLOOR(LN(1 - α)/LN(1 - p))
#5:        LN(1 - α)/LN(1 - p)
           FLOOR(LN(1 - α)/LN(1 - p)) + 1
```

answers the least number of repetitions for success to set in at least one with probability greater or equal to α. That is the least integer greater or equal than

```
                                 i - 1
#6:   af(i, p) := p·(1 - p)
```

gives the answer to the question: What is the probability the success will be set in at first in the i-th repetition?

The last function

$$\#7: \quad irs(k, n, p) := PNOM(k, n)\cdot IF\left(DIM(p) = DIM(k), \prod\left(p_i^{k_i}, i, DIM(k)\right), 0\right)$$

solves the problem of independent repeated events in the case of s possible results of the experiment. In this function n is the number of repetitions, $k = [k_1, ..., k_s]$ is a vector such that $k_1+...+k_s = n$ and $p = [p1, ..., p_s]$ is a vector such that $p_1+ ... +p_s = 1$.

Now, let $A_1, ..., A_s$ are the possible results of the experiment with probabilities $p_1, ..., p_s$ respectively. The probability that in n repetitions of that experiment the results $A_1, ..., A_s$ will be set in $k_1, ..., k_s$ - times respectively is given by formula

$$\binom{n}{k_1, k_2, ..., k_s} \cdot p_1^{k_1} p_2^{k_2} \dots p_s^{k_s}$$

which is realized in the function `irs(k,n,p)`.

Using the functions defined above we will solve the following elementary problems from probability theory:

We will throw a die 21 times.

a) What is the probability that number six will be thrown six times?

b) What is the most likely count of throwing number six in these twenty replications?

c) How many times do we have to throw the die to reach at least one number six with a probability greater than 0.9?

d) What is the probability that the numbers 1,2,3,4,5,6 will be thrown 1,2,3,4,5,6 times respectively?

#8: Notation := Decimal

Answer to question a)

#9: $ir\left(6, 21, \frac{1}{6}\right) = 0.07548933698$

Answer to question b)

#10: $mlc\left(21, \frac{1}{6}\right) = 3$

Answer to question c)

#11: $lnr\left(0.9, \frac{1}{6}\right) = 13$

Answer to question d)

#12: $irs\left([1, 2, 3, 4, 5, 6], 21, \left[\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}\right]\right)$

#13: 0.00009359689105

Let my try solving the problems in the "traditional" way (Josef):

#14: $BINOMIAL\_DENSITY\left(6, 21, \frac{1}{6}\right) = 0.07548933698$

#15: $SELECT\left(v_2 = MAX\left(\left(TABLE\left(BINOMIAL\_DENSITY\left(x, 21, \frac{1}{6}\right), x, 1, 21, 1\right)\right)\downarrow\downarrow 2\right), v, TABLE\left(BINOMIAL\_DENSITY\left(x, 21, \frac{1}{6}\right), x, 1, 21, 1\right)\right)$

#16: [[3, 0.2312786059]]

#17: $CEILING\left(NSOLUTIONS\left(1 - \left(\frac{5}{6}\right)^x = 0.9, x\right)\right) = [13]$

#18: $21\cdot\frac{1}{6}\cdot COMB(20, 2)\cdot\left(\frac{1}{6}\right)^2\cdot COMB(18, 3)\cdot\left(\frac{1}{6}\right)^3\cdot COMB(15, 4)\cdot\left(\frac{1}{6}\right)^4\cdot COMB(11, 5)\cdot\left(\frac{1}{6}\right)^5\cdot COMB(6, 6)\cdot\left(\frac{1}{6}\right)^6 = 0.00009359689105$

Bibliography

[1] Eisen, M.: Introduction to mathematical probability theory, Prentice Hall, New Jersey 1969.

[2] Feller, W.: An introduction to probability theory and its applications, Vol.1, Wiley, New York 1950, Vol 2, 1966.

[3] Rényi, A.: Wahrscheinlichkeitsrechnung mit einem Anhang über Informationstheorie, Deutscher Verlag der Wissenschaften, Berlin 1962.

Bivariate Normal Distribution

Dr. G.Marinell, Innsbruck, A

Die Zufallsvariable *X* hat eine univariate Normalverteilung mit dem Erwartungswert μ und der Varianz σ^2, wenn sie folgende Dichtefunktion besitzt:

$$f_N\left(x \mid \mu, \sigma^2\right) = \frac{1}{\sqrt{2\pi\sigma}} \cdot e^{-(x-\mu)^2/\left(2\sigma^2\right)}.$$

Die Werte der Verteilungsfunktion dieser Normalverteilung

$$W\left(X \le a\right) = F_N\left(a\right) = \int_{-\infty}^{a} f_N\left(x \mid \mu, \sigma^2\right) dx$$

können in *DERIVE* mit dem Befehl

```
NORMAL(z, m, s)
```

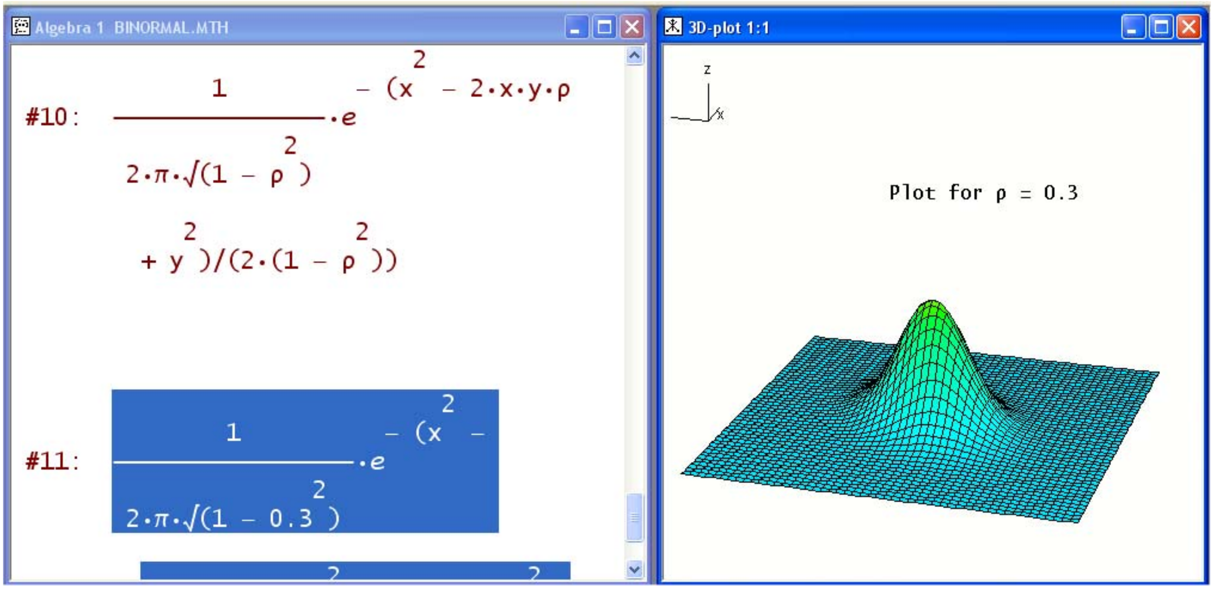
bestimmt werden, wobei $\mu = m$ und $\sigma^2 = s^2$ ist. (siehe auch BÖHM J., (1992), Normal Distribution & Inverse, DNL#8).

Die zweidimensionale Zufallsvariable (*X,Y*) hat eine (nichtsinguläre) bivariate Normalverteilung, wenn sie folgende Dichtefunktion aufweist:

$$f_{2N}\left(x, y \mid \mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \rho\right) =$$

$$= \frac{1}{2\pi\,\sigma_1\,\sigma_2\,\sqrt{1-\rho^2}} \cdot exp\left\{-\frac{1}{2\left(1-\rho^2\right)} \cdot \left[\left(\frac{x-\mu_1}{\sigma_1}\right)^2 - \right.\right.$$

$$\left.\left. -2\rho \cdot \left(\frac{x-\mu_1}{\sigma_1}\right) \cdot \left(\frac{y-\mu_2}{\sigma_2}\right) + \left(\frac{y-\mu_2}{\sigma_2}\right)^2\right]\right\}$$



Für diese bivariate Normalverteilung sind die Werte der Verteilungsfunktion

$$W\left(X \leq a, Y \leq b\right) = F_{2N}\left(a,b\right) = \int_{-\infty}^{a}\int_{-\infty}^{b} f_{2N}\left(x,y\right) dy\, dx$$

numerisch schwierig zu berechnen. Tong schlägt folgenden Ausdruck für die Berechnung vor (Y. L. Tong (1990): The Multivariate Normal Distribution, New York, Springer, p. 15):

$$F_{2N}\left(a,b\right) = \int_{-\infty}^{\infty} \Phi\left(\frac{\sqrt{|\rho|}\cdot z + u}{\sqrt{1-|\rho|}}\right)\cdot \Phi\left(\frac{\delta_\rho\sqrt{|\rho|}\cdot z + v}{\sqrt{1-|\rho|}}\right)\cdot \frac{1}{\sqrt{2\pi}}\cdot e^{-\frac{z^2}{2}} dz$$

$\Phi(z)$ ist die Verteilungsfunktion der Standardnormalverteilung.

$$\Phi(z) = F_Z\left(z\right) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{z} e^{-\frac{t^2}{2}} dt,$$

und δ_ρ ist gleich

$$\delta_\rho = \begin{cases} 1 & \text{für } \rho \geq 0, \\ -1 & \text{für } \rho < 0. \end{cases}$$

u und *v* sind die standardisierten Merkmalsausprägungen der beiden Zufallsvariablen *X* und *Y*, die wie folgt definiert sind:

$$u = \left(a - \mu_1\right)/\sigma_1,$$
$$v = \left(b - \mu_2\right)/\sigma_2.$$

In dem unten angegebenen Programm kann mit dem Befehl

```
BINORMAL(u, v, r)
```

der Wert der Verteilungsfunktion der bivariaten Normalverteilung näherungsweise berechnet werden. Dabei werden 17 äquidistante Stützstellen und zwar von -4 bis +4 berücksichtigt.
u und *v* sind die standardisierten Merkmale und $r \equiv \rho$ der Korrelationskoeffizient.

Mit dem Befehl

```
BINO(u, v, r, m, n, s)
```

kann man die Zahl der Stützstellen und damit die Güte der Näherung selbst bestimmen. *m* ist die Untergrenze, *n* die Obergrenze und *s* die Schrittweite. Obwohl der Definitionsbereich der Standardnormalverteilung von -∞ bis +∞ läuft, kommen als Unter- und Obergrenze kaum Werte in Frage, die außerhalb des Intervalls [-5, 5] liegen. Je kleiner die Schrittweite *s* gewählt wird, umso besser ist die Näherung, aber umso länger ist auch die Rechenzeit. Durch

$$\frac{n-m}{s}+1$$

ist die Zahl der Stützstellen gegeben.

BINORMAL calculates the distribution moment of the bivariate Normal Probability function.

μ = (a - μ1)/σ1, v = (b - μ2)/σ2, r = σ12/(σ1 * σ2)

#3:

$$\text{BINORMAL}(u,v,r):=\frac{\left(\left[\text{VECTOR}\left(1,z,1,\frac{4--4}{0.5}+1\right)\right]\cdot\text{VECTOR}\left(\left[z,\frac{1}{\sqrt{(2\cdot\pi)}}\cdot e^{-z^2/2},\text{NORMAL}\left(\frac{\sqrt{r}\cdot z+u}{\sqrt{(1-r)}}\right)\cdot\text{NORMAL}\left(\frac{\sqrt{r}\cdot z+v}{\sqrt{(1-r)}}\right)\cdot\left(\frac{1}{\sqrt{(2\cdot\pi)}}\cdot e^{-z^2/2}\right)\right],z,-4,4,0.5\right)\right)_{1,3}}{\left(\left[\text{VECTOR}\left(1,z,1,\frac{4--4}{0.5}+1\right)\right]\cdot\text{VECTOR}\left(\left[z,\frac{1}{\sqrt{(2\cdot\pi)}}\cdot e^{-z^2/2},\text{NORMAL}\left(\frac{\sqrt{r}\cdot z+u}{\sqrt{(1-r)}}\right)\cdot\text{NORMAL}\left(\frac{\sqrt{r}\cdot z+v}{\sqrt{(1-r)}}\right)\cdot\left(\frac{1}{\sqrt{(2\cdot\pi)}}\cdot e^{-z^2/2}\right)\right],z,-4,4,0.5\right)\right)_{1,2}}$$

BINO calculates the distribution function with the possibility to determine the accuracy by the interval [m,n] and the increment s.
(In BINORMAL these values are: m = -4, n = 4, s = 0.5)

#4:

$$\text{BINO}(u,v,r,m,n,s):=\frac{\left(\left[\text{VECTOR}\left(1,z,1,\frac{n-m}{s}+1\right)\right]\cdot\text{VECTOR}\left(\left[z,\frac{1}{\sqrt{(2\cdot\pi)}}\cdot e^{-z^2/2},\text{NORMAL}\left(\frac{\sqrt{r}\cdot z+u}{\sqrt{(1-r)}}\right)\cdot\text{NORMAL}\left(\frac{\sqrt{r}\cdot z+v}{\sqrt{(1-r)}}\right)\cdot\left(\frac{1}{\sqrt{(2\cdot\pi)}}\cdot e^{-z^2/2}\right)\right],z,m,n,s\right)\right)_{1,3}}{\left(\left[\text{VECTOR}\left(1,z,1,\frac{n-m}{s}+1\right)\right]\cdot\text{VECTOR}\left(\left[z,\frac{1}{\sqrt{(2\cdot\pi)}}\cdot e^{-z^2/2},\text{NORMAL}\left(\frac{\sqrt{r}\cdot z+u}{\sqrt{(1-r)}}\right)\cdot\text{NORMAL}\left(\frac{\sqrt{r}\cdot z+v}{\sqrt{(1-r)}}\right)\cdot\left(\frac{1}{\sqrt{(2\cdot\pi)}}\cdot e^{-z^2/2}\right)\right],z,m,n,s\right)\right)_{1,2}}$$

Example: Let X normal distributed with mean = 40 and stddev = 2.5 and Y normal distributed with mean = 50 and stddev = 5.5.
Correlationcoefficent r = 0.29.

What is the probability of p(X ≤ 38 and Y ≤ 53)?

#5: $\text{BINORMAL}\left(\frac{38-40}{2.5},\frac{53-50}{5.5},0.29\right)$

#6: 0.1769063132

#7: $\text{BINO}\left(\frac{38-40}{2.5},\frac{53-50}{5.5},0.29,-5,5,0.1\right)$

#8: 0.1769116283

I've received some letters and a lot of material to be used in classroom from Keith Eames, Senior Lecturer, Kingsway, Camden College. Here are the main parts of two letters, one worksheet, a question and–- I hope so – a suitable answer.

Part of the 1st letter:

... Though I am interested in the Use of DERIVE as a teaching tool I feel I do not have the time to produce an English DERIVE manual, but thanks for the offer of help. If you feel that the material I have produced is suitable then please feel free to use it as you think fit. A mention in any literature where it is used would be appreciated.

I enclose a copy of a recent assignment given to students on curve fitting population statistics using DERIVE. I will send you copies of any further development work on DERIVE. .....

(see next two pages!)

The second letter:

Dear Josef,

I wonder if anyone in the DERIVE User Group can help me out with a solution to a problem involving transformation of experimental data to a linear form?

I enclose a copy of a recent question given to students on curve fitting of experimental data. I was surprised that DERIVE would not solve it, with the FIT command.

The population of a colony of insects is measured at daily intervals:

Number of days	0	1	2	3	4	5
Population	40	50	80	110	150	220

It is believed that p and t are related by an equation of the form $p = a \cdot b^t$. Estimate the values of a and b.

When I attempted to fit an equation of this form, DERIVE failed to give any answer. So when this did not work, I thought I could try the transformation ln(p) = ln(a) + t ln(b) to the data in the 6 x 2 matrix. I found that I could not take natural logs of column 2 in the data matrix.

A similar problem was encountered in trying to fit a curve to data which was in the form of $T = a.I^n$, where the values of a and n had to be found.

The Physicists and Chemists within the College wish to be able to perform transformations on experimental data, where the data is in matrix form. Is it possible to take, for instance, the log or square root of a single column of data values? They do not really want to take the log or square root of every single element in a column of a 50 x 2 matrix.
(see page 26!)

FOUNDATION CM3 MODULE: CENSUS ASSIGNMENT

When using *Derive* save your work and print out the graphs. Include the graphs in your write up.

Population of USA from 100 to 1970

Year, *T*	Population (in millions)
1900	75 995
1910	91 972
1920	105 711
1930	123 203
1940	131 669
1950	150 697
1960	179 323
1970	203 212

(a) Use *Derive* to plot the given data. Comment on your plot.

Enter the data with T = year values and Y = population values. Since the T-values are so large they can be transformed using the formula:

$$t = \frac{T - 1900}{10} .$$

This transformation results in the year 1980 being equivalent to $t = 8$ and the corresponding year values:

$$t = 0, 1, 2, 3, 4, 5, 6, 7$$

(b) Fit various polynomials to the data starting with a linear fit then a quadratic fit etc. up to a polynomial of degree 7.

A polynomial of degree 7 would be given by:

$$Y = \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \alpha_3 t^3 + \alpha_4 t^4 + \alpha_5 t^5 + \alpha_6 t^6 + \alpha_7 t^7$$

where α_0, $\alpha_1, \alpha_2, ..., \alpha_7$ are constants.

Plot each polynomial and comment on the closeness of each curve to the actual data points.

(c) In each case use your polynomial equation to predict the population of USA in 1980. Find the true value of the population of USA in 1980 and comment on the accuracy of the predictions.

Fitting the linear model

A linear model will be of the form:

$$Y = \alpha_0 + \alpha_1 t$$

You will be required to find the two unknown coefficients α_0 and α_1.

Derive should give you α_0 = 71.4434 and α_1 = 17.5084, with the estimate for the population in 1980 as 211.510 million.

To estimate the population value in 1980 substitute t = 8 in the equation of the line of best fit. Use the <Manage> <Substitute> command in *Derive* to carry out the substitution. (*Now simply Substitute via the menu bar.*)

Fitting the quadratic model

A quadratic model will be of the form:

$$Y = \alpha_0 + \alpha_1 t + \alpha_2 t^2$$

You will be required to find the three unknown coefficients α_0, α_1 and α_2.

Derive should give you α_0 = 79.0375, α_1 = 9.91363 and α_2 = 1.08498, with the estimate for the population in 1980 as 227.7786 million.

As a check the coefficients of the polynomial of degree 7 are:

$$\begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \\ \alpha_7 \end{pmatrix} = \begin{pmatrix} 78.1631 \\ 0.58341 \\ 11.2415 \\ 1.38854 \\ -3.24255 \\ 0.990631 \\ -0.118051 \\ 0.005001 \end{pmatrix}$$

(*There are slight differences to the coefficients obtained by using recent versions of DERIVE. Compare with the results on page 27. Josef*)

#1: data :=
$$\begin{bmatrix} 0 & 40 \\ 1 & 50 \\ 2 & 80 \\ 3 & 110 \\ 4 & 150 \\ 5 & 220 \end{bmatrix}$$

#2: DATLN(d_) := [d_↓↓1, LN(d_↓↓2)]`

#3: DATLN(data) =
$$\begin{bmatrix} 0 & 3.688879454 \\ 1 & 3.912023005 \\ 2 & 4.382026634 \\ 3 & 4.700480365 \\ 4 & 5.010635294 \\ 5 & 5.393627546 \end{bmatrix}$$

#4: FIT_AUX(d_) := FIT([t, a + t·b], DATLN(d_))

#5: EXP_FUNC(d_) := $p = e^{\text{FIT\_AUX(d\_)}}$

#6: EXP_FUNC(data) = $(p = 38.38281493 \cdot e^{0.3468008873 \cdot t})$

#7: COMP(d_, u, t) := VECTOR($[d\__{k,1}, d\__{k,2}, \text{SUBST}(u, t, d\__{k,1})]$, k, DIM(d_))

#8: COMP(data, $38.382 \cdot e^{0.3468 \cdot t}$) =
$$\begin{bmatrix} 0 & 40 & 38.382 \\ 1 & 50 & 54.29263594 \\ 2 & 80 & 76.79876810 \\ 3 & 110 & 108.6344525 \\ 4 & 150 & 153.6671039 \\ 5 & 220 & 217.3673109 \end{bmatrix}$$

#9: 2nd example - census

#10: census :=
$$\begin{bmatrix} 1900 & 75.995 \\ 1910 & 91.972 \\ 1920 & 105.71 \\ 1930 & 123.2 \\ 1940 & 131.66 \\ 1950 & 150.69 \\ 1960 & 179.32 \\ 1970 & 203.21 \end{bmatrix}$$

#11: EXP_FUNC(census) = $(p = 5.546087374 \cdot 10^{-10} \cdot e^{0.01351599389 \cdot t})$

#12: $COMP(census, 5.4835\cdot 10^{-10}\cdot e^{0.013521\cdot t}) =$

1900	75.995	78.71176636
1910	91.972	90.10743253
1920	105.71	103.1529309
1930	123.2	118.0871195
1940	131.66	135.1834376
1950	150.69	154.7549121
1960	179.32	177.1598891
1970	203.21	202.8085951

Polynomial of degree 7:

#13: $transf(t) := FIT([t, a\cdot t^7 + b\cdot t^6 + c\cdot t^5 + d\cdot t^4 + e\cdot t^3 + f\cdot t^2 + g\cdot t + h], [[0, 1, 2, 3, 4, 5, 6, 7], census\downarrow\downarrow 2]')$

#14: $transf(t) := 0.03257063492\cdot t^7 - 0.8285055555\cdot t^6 + 8.293836111\cdot t^5 - 41.26730555\cdot t^4 + 106.0181694\cdot t^3 - 131.0781888\cdot t^2 + 74.80642380\cdot t + 75.995$

#15: TABLE(transf(t), t, 0, 7)

#16:

0	75.995
1	91.972
2	105.711
3	123.2030000
4	131.6690000
5	150.6970000
6	179.3229999
7	203.2120001

Retransform to enter the years 1900 - 1980:

#17: $cens(x) := transf\left(\frac{x}{10} - 190\right)$

#18: $cens(x) := 0.000000003257063491\cdot x^7 - 0.00004414744999\cdot x^6 + 0.2564458850\cdot x^5 - 827.5625641\cdot x^4 + 1602303.135\cdot x^3 - 1861346689.0\cdot x^2 + 1201226335982.3\cdot x - 332225622336045.7$

#19: TABLE(cens(x), x, 1900, 1980, 10)

#20:

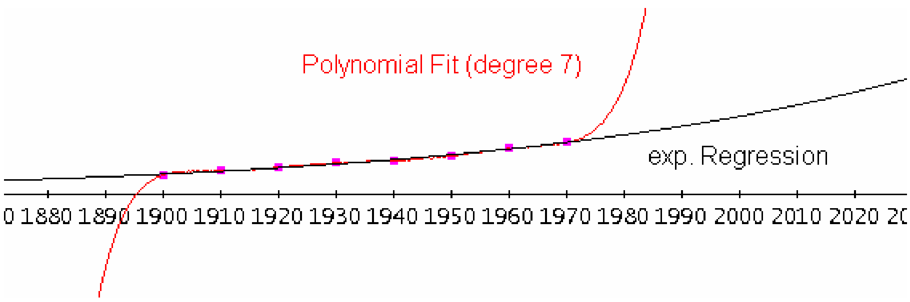
1900	75.995
1910	91.97200000
1920	105.7110000
1930	123.2030000
1940	131.6690000
1950	150.6970000
1960	179.3230000
1970	203.2120000
1980	426.0950000

Quadratic Regression

#21: $\text{quadcens}(t) := \text{FIT}\left(\left[t,\ f\cdot t^2 + g\cdot t + h\right], [[0, 1, 2, 3, 4, 5, 6, 7], \text{census}\downarrow\downarrow 2]'\right)$

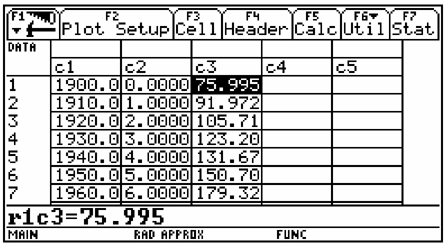
#22: $\text{quadcens}(t) := 1.084285714\cdot t^2 + 9.918309523\cdot t + 79.03366666$

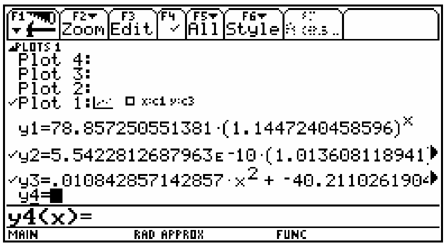
#23: $\text{quadcens}(8) = 227.7744285$

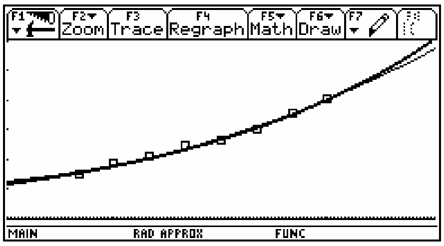


How to do it on the TIs:

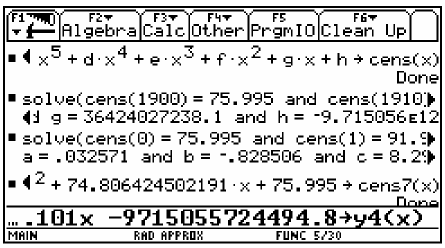
See first the exponential regression line together with the quadratic regression:

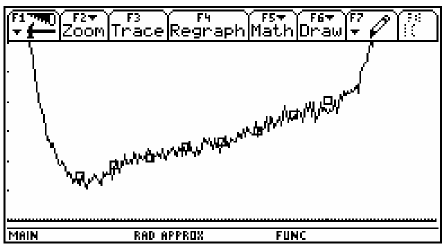


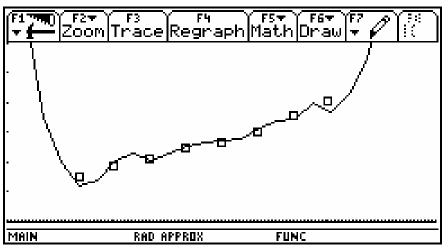




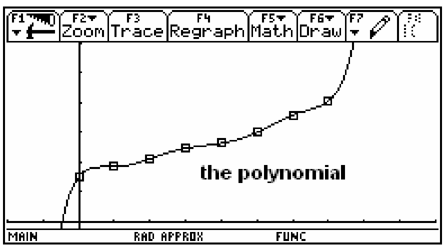
As we have 8 pairs of data we can try to find the polynomial of degree 7 (without transforming the year values) which leads to an interesting graph:







Same graph but xres = 10



The polynomial with transformed years

Von der Binomial- zur Normalverteilung

W. Pröpper, Nürnberg

(This contribution "From the Binomial- to the Normaldistribution" is written in German. So I will try to give short summaries at some points where I find that it could be useful for non-german readers. Editor)

1. Die Binomialverteilung

Jacob Bernoulli (1655 - 1705) wird die Formel

```
                               k       n - k
#1:   B(n, p, k) := COMB(n, k)·p ·(1 - p)
```

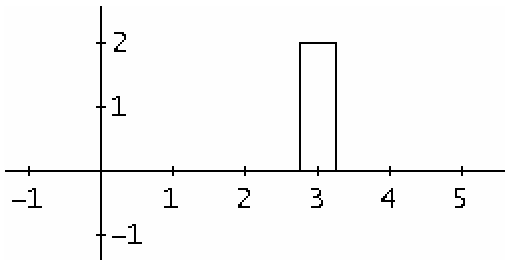
zugesprochen. Mit ihr lässt sich die Wahrscheinlichkeit für genau *k* Treffer in einer Bernoullikette mit der Länge *n* und der Trefferwahrscheinlichkeit *p* berechnen. Für einfache Fälle kann `B(n,p,k)` „zu Fuß" berechnet, bzw. in Tafelwerken nachgeschlagen werden. Bei „krummen" Werten versagen diese Hilfsmittel in der Schule, und der Wunsch nach einer einfachen Näherungsformel liegt nahe.

Zur Veranschaulichung der Binomialverteilung dient das Histogramm. In ihm wird eine Wahrscheinlichkeitsverteilung $x \rightarrow W(x)$ in der Form von Balken über dem Argument *x* derart dargestellt, dass die Balkenfläche gleich (bzw. proportional) der Wahrscheinlichkeit $W(x)$ ist.

Wir konstruieren deshalb zunächst Balken der Breite *d* und der Höhe *h*, die symmetrisch zu *x* auf die Abszissenachse gesetzt werden. Die Funktion `BALK(x,d,h)` leistet dies.

```
                        ⎡ x - 0.5·d   0 ⎤
                        ⎢ x - 0.5·d   h ⎥
#2:   BALK(x, d, h) :=  ⎢ x + 0.5·d   h ⎥
                        ⎣ x + 0.5·d   0 ⎦

#3:   BALK(3, 0.5, 2)
```



(Im Plot-Fenster muss die Options-> Display > Points auf Connected eingestellt sein, damit die Balken mit ausgezeichneten Randlinien erscheinen.)

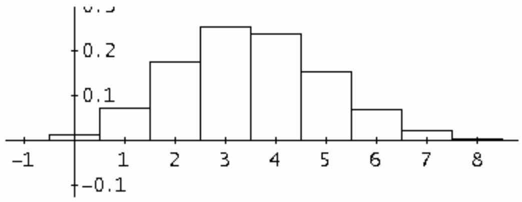
(The students should plot BALK(x=k, d=1, h=B(n,p,k); e.g. n = 20, p = 0.5, $k \in$ {..9,...12,..}.)
Die Bedeutung des Konstrukts `BALK` für (Derive-ungeübte) Schüler kann an wenigen einfachen Beispielen gezeigt werden.

Um zum Thema überzuleiten, lassen wir nun an der Stelle *k* Balken der Höhe `B(n,p,k)` und der Breite 1 nacheinander plotten. Anschauliche Bilder ergeben sich beispielsweise mit n = 20, p = 0.5 und $k \in \{9 .. 12\}$. Dabei legt schon der erste Balken eine Anpassung des Koordinatensystems nahe. Mit einer geeigneten Skalierung und Positionierung des Centers erhält man eine passende Umgebung. Schließlich schalten wir noch über Options > Display > Color den automatischen Farbwechsel aus, um Farbgebungsmöglichkeiten (sofern vorhanden) auf wichtigere Fälle anwenden zu können.

Die Schüler vermuten schnell, wozu die o.g. Plot-Übungen nützlich sein können. Deshalb leuchtet die Syntax der folgenden Funktion `HIST1` leicht ein. Sie liefert ein Histogramm der Binomialverteilung `B(n,p)`.

```
#4:   HIST1(n, p) := VECTOR(BALK(k, 1, B(n, p, k)), k, 0, n)

#5:   HIST1(10, 0.35)
```



Durch geschickte Variation von n und p lassen sich Charakteristika der Binomialverteilung veranschaulichen:

-- Bei festem p und veränderlichem n sieht man das „Auseinanderfließen", der Histogramme.

-- Festes n und variables p zeigt die Symmetrie im Falle $p = 0.5$ und die Verschiebung des Maximums in den Fällen $p < 0.5$ bzw. $p > 0.5$.

-- Schließlich konstruiert man Verteilungen mit konstantem Produkt $n{\cdot}p$.

(Um erträgliche Rechenzeiten zu erhalten, stellen wir die Option-Precision auf Approximate und die Option-Notation auf Scientific. 3 signifikante Stellen sind jeweils genug. Bei der oben vorgeschlagenen Einstellung des Plot-Fensters kann man mit n bis zu Werten von 60 gehen. In jeder Gruppe wird das Aussehen der Plots farblich durch Option > Display >Color-Plot variiert.)

(Change n and p to show the influence of the parameters for the distribution. (p = const. & n = var., p = var. & n = const., $n{\cdot}p$ = const.). The last step - $n{\cdot}p$ = const - prepares for standardisation - the maxima remain on the same position. We transform the coordinates so that the y-axis will become an axis of symmetry.)

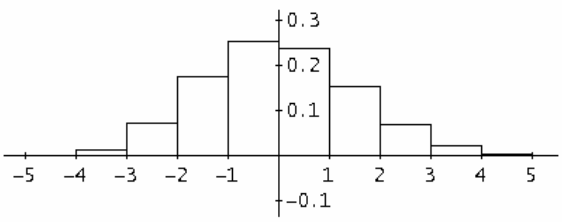
2. Die Standardisierung

Im letzten Schritt (konstantes $n{\cdot}p$) wird der Weg zur Standardisierung vorbereitet, da hier die Lage der Maxima erhalten bleibt. Wir verschieben deshalb das Koordinatensystem so, dass die Maxima aller Histogramme an derselben Stelle liegen. Die Wahl dieser Stelle an sich ist willkürlich, aber wenn die Ordinatenachse Symmetrieachse (für $p = 0.5$) wird, ist auch Ästheten gedient. Die Verschiebung erfolgt also um das Produkt $n{\cdot}p$.

```
#6:   μ(n, p) := n·p

#7:   HIST2(n, p) := VECTOR(BALK(k - μ(n, p), 1, B(n, p, k)), k, 0, n)

#8:   HIST2(10, 0.35)
```



Die Histogramme `HIST2` müssen nicht in so vielen Varianten dargestellt werden, wie `HIST1`, denn sie zeigen das angestrebte und schon vorüberlegte Resultat.

Schwieriger ist es, die Charakteristika der neuen Verteilung herauszuarbeiten. Es handelt sich um eine Wahrscheinlichkeitsverteilung, die jedem $x = k - n \cdot p$ die Wahrscheinlichkeit $W(x) = B(n, p, k)$ zuordnet. Mit der SUM-Funktion lässt sich leicht der Ausdruck

```
#9:   ERW(n, p) :=  Σ   (k - n·p)·B(n, p, k)
                   k=0..n

#10:  ERW(10, 0.35) = 0
```

bilden und mit Simplify vereinfachen. (Achtung: Option-Precision auf Exact umstellen). Entsprechend kann dann auf die Varianz eingegangen werden:

```
#11:  VAR_(n, p) :=  Σ   (k - n·p)^2·B(n, p, k)
                    k=0..n

#12:  VAR_(10, 0.35) = 2.275
```

Als Ergebnis lässt sich festhalten: HIST2 ist im Erwartungswert auf 0 normiert, jedoch die Varianz ist noch abhängig von der Wahl von n und p. (Interpretiert man die Varianz als ein Maß dafür, wie stark sich die Werte einer Zufallsgröße um den Erwartungswert scharen, so lässt sich das mit HIST2 gut veranschaulichen.)

(The expected value is 0 for each (n, p), but the variances differ. Using $\sigma(n, p)$ we create the function NORM. Before plotting NORM its attributes should be discussed.)

Nun ist einleuchtend, dass zur weiteren Vereinheitlichung die Varianz benötigt wird. Wir betrachten jedoch die Standardabweichung σ:

```
#13:  σ(n, p) := √(n·p·(1 - p))
```

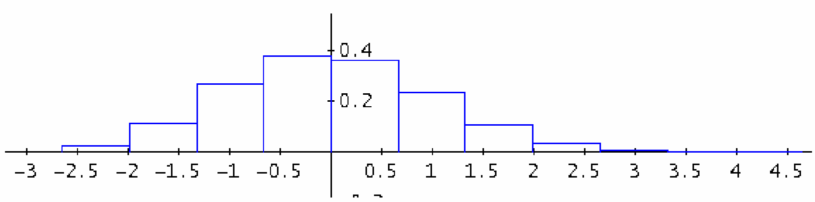
Mit Hilfe von $\sigma(n, p)$ bilden wir nun die Funktion NORM:

```
#14:  NORM(n, p) := VECTOR(BALK((k - μ(n, p))/σ(n, p), 1/σ(n, p), σ(n, p)·B(n, p, k)), k, 0, n)
```

Vor dem Plotten sollten ihre Eigenschaften diskutiert werden:

Bei den vorherigen Beispielen war immer $\sigma > 1$, sodass nun Diagramme zu erwarten sind, deren Balken sich näher um die y-Achse gruppieren ($(k - \mu)/\sigma < k - \mu$), schmäler ($1/\sigma < 1$) und höher sind ($\sigma \cdot B > B$). Gut einleuchtend ist die Erhöhung der Balken um den Faktor σ, wenn gleichzeitig die Breite um diesen Faktor gestaucht wird.
Die Plots von NORM mit verschiedenen Werten von n und p zeigen sofort, dass das Ziel der Vereinheitlichung erreicht ist. Die Histogramme der standardisierten Verteilung sind weitgehend unabhängig von n und p. Mit den nachfolgenden Funktionen ERWW(n,p) und VARI (n,p) lässt sich die Vereinheitlichung numerisch ausdrücken (und ggf. der Wunsch nach einem Beweis wecken).

```
#15:  NORM(10, 0.35)
```



$$\#16: \quad ERWW(n, p) := \sum_{k=0}^{n} \frac{k - \mu(n, p)}{\sigma(n, p)} \cdot B(n, p, k)$$

$$\#17: \quad VARI(n, p) := \sum_{k=0}^{n} \left(\frac{k - \mu(n, p)}{\sigma(n, p)} \right)^2 \cdot B(n, p, k)$$

```
#18:   [ERWW(10, 0.35), ERWW(35, 0.47)] = [0, 0]

#19:   [VARI(10, 0.35), VARI(35, 0.47)] = [1, 1]
```

(`ERWW(n,p)` and `VARI(n,p)` show standardisation numerically)

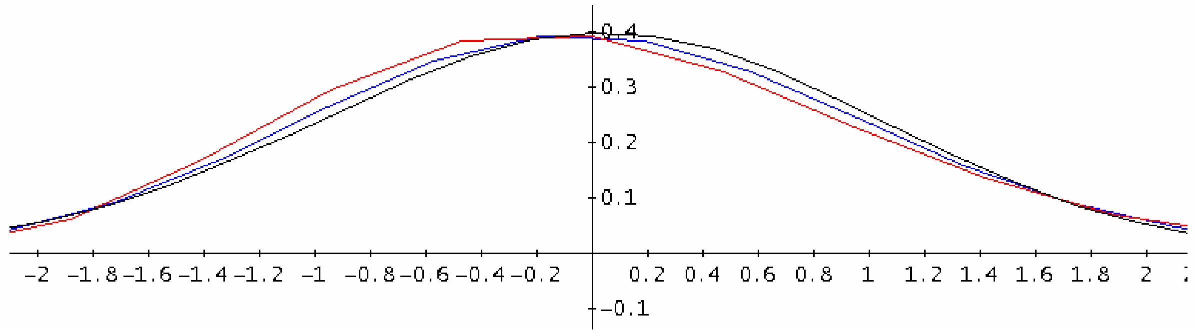
3. Die Gaußkurve

Das nächste Ziel ist nun, von der diskreten Wahrscheinlichkeitsverteilung zu einer in R definierten Näherungsfunktion zu gelangen. Um hier besser experimentieren zu können (mehrere Plots von `NORM` sind wirklich unübersichtlich!), führen wir die Funktion `PKTE(n,p)` ein. Da im Plot-Fenster die Option-State noch immer auf Connected steht, ergeben sich mit $n > 20$ schon sehr ordentliche (abschnittsweise lineare) Näherungen.

```
#21:   PKTE(30, 0.35)

#22:   PKTE(50, 0.1)

#23:   PKTE(100, 0.7)
```



(The next goal is to find an approximation function. As some plots of `NORM` are difficult to be read on one screen so we use PKTE –Options > Display > Points > Connected. Which functions have graphs similar to these plots - as GLO1, GAU1, ...?

Für die Frage, welcher einfache Funktionsgraph diese Punkte verbindet, lassen sich Experimente durchführen, die auf Funktionen basieren, welche aus der Analysis bekannt sind. Sie müssen symmetrisch zur y-Achse liegen, ein Maximum bei (0 / ~0.4) haben und im Unendlichen verschwinden. (Im Plot-Fenster sollte die Zeichenoption auf Connected No umgestellt werden, um unnötige Linien zu vermeiden.) Von den Schülern werden schnell Funktionen der Art `GLO1` und `GAU1` genannt. Weitere Beispiele werden sich daran orientieren.

$$\#24: \quad GLO1(x) := \frac{0.4}{1 + x^2}$$

$$\#25: \quad GAU1(x) := 0.4 \cdot EXP(-x^2)$$

Diese Experimente führen schließlich zur Gaußschen Fehlerkurve oder kurz ***Gaußkurve***.

$$\#26: \quad \mathrm{GAUS}(x) := \frac{1}{\sqrt{(2\cdot\pi)}}\cdot \mathrm{EXP}\left(-\frac{x^2}{2}\right)$$

(Selbstverständlich lässt sich dieser Schritt, besonders wegen √(2π), nicht ohne Hilfe des Lehrers tun. Er lässt sich jedoch mit

$$\#27: \quad \int_{-\infty}^{\infty} e^{-x^2/2}\, dx = \sqrt{2}\cdot\sqrt{\pi}$$

sehr elegant begründen.

Ein Plot der Funktion GAUS lässt sich nun mit weiteren Darstellungen der Funktion PKTE verbinden (wobei weiterhin die Punkte diskret gezeichnet werden sollten). Hier lässt sich gut verdeutlichen, dass die Näherungsfunktion gute Werte liefert, falls $\sigma > 3$ ist.

Für eine Näherungsfunktion der Binomialverteilung muss der bisherige Weg zurückverfolgt werden: Das Argument der Funktion GAUS wird durch $(x - \mu)/\sigma$ ersetzt und der Wert von GAUS wird durch den Faktor $\sigma(n, p)$ reduziert. Als Ergebnis erhalten wir

$$\#28: \quad \mathrm{GAUSS}(n, p, x) := \frac{1}{\sigma(n, p)\cdot\sqrt{(2\cdot\pi)}}\cdot \mathrm{EXP}\left(-\frac{\left(\frac{x - \mu(n, p)}{\sigma(n, p)}\right)^2}{2}\right)$$

(At last we have to consider the transformation x → (x – μ)/σ and the normation - divide by σ (n, p), so we obtain #28.)

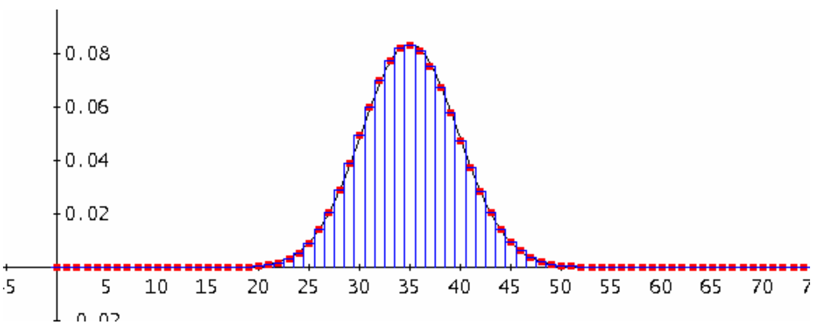
Eine Wahrscheinlichkeitsverteilung, deren Dichte durch die Funktion GAUSS bestimmt ist, heißt normal verteilt. Vergleiche der Plots von GAUSS mit HIST1 (bzw. der nachfolgenden Funktion PKTB) zeigen, dass das Ziel erreicht ist. Entsprechend können numerisch die Werte von B(n,p,k) und GAUSS(n,p,k) verglichen werden.

```
#29:   pktb(n, p) := VECTOR([k, B(n, p, k)], k, 0, n)

#30:   HIST1(100, 0.35)

#31:   pktb(100, 0.35)

#32:   GAUSS(100, 0.35)
```



(Finally we compare the plots of GAUSS with HIST1 and PKTB and the numerical values of BIN(n,p,k) with GAUSS(n,p,k).)

3. Praktische Anwendung

Zum Schluss wäre noch die Frage anzuschneiden, wie sich nun die in der Stochastik häufig auftretenden Problemstellungen "Berechne die Wahrscheinlichkeit für eine Trefferanzahl im Intervall [$k1$; $k2$]" oder auch "Berechne P($X \le k$)" bearbeiten lassen.

Über die Binomialverteilung liefert die Funktion `WBIN` eine Lösung (wobei dies in der schulischen Praxis ohne Computerunterstützung mit der ebenfalls tabellierten kumulativen Binomialverteilung angegangen wird; kritischen Rückfragen kann durch Vergleich von `WBIN(n,p,0,k)` mit `Φ(n,p,k)` begegnet werden.

```
                           k2
#33:  WBIN(n, p, k1, k2) :=  Σ   B(n, p, k)
                          k=k1
```

Aus der Erinnerung an die Herleitung des Riemannschen Integrals kommt schnell die Vermutung, dass bei Verwendung der Funktion `GAUSS` ein Integral der Form

```
                           k2
#34:  WGA1(n, p, k1, k2) := ∫   GAUSS(n, p, x) dx
                           k1
```

das Problem lösen wird. Die Berechnung von `WBIN` und `WGA1` mit einigen konkreten Wertebelegungen zeigt aber, dass die erste Vermutung falsch war. Insbesondere ist immer `WGA1` < `WBIN`. Ein Vergleich mit den letzten Plots zeigt die Ursache auf:

```
#35:  WBIN(100, 0.35, 32, 40) = 0.6418672045

#36:  WGA1(100, 0.35, 32, 40) = 0.5880622500
```

Bei `WBIN` sind die Randstreifen voll berücksichtigt, während sie bei `WGA1` nur etwa zur Hälfte mit eingehen. Deshalb ist die in `WGAU` angebrachte Korrektur erforderlich.

```
                           k2 + 0.5
#37:  WGAU(n, p, k1, k2) := ∫          GAUSS(n, p, x) dx
                           k1 - 0.5

#38:  WGAU(100, 0.35, 32, 40) = 0.6440325550
```

Die Aufgabe "Berechne P ($X \le k$)" führt demnach auf die Funktion

```
                  k + 0.5
#39:  φ(n, p, k) := ∫        GAUSS(n, p, x) dx
                  -∞

#40:  WBIN(100, 0.35, 0, 42) = 0.9405698166

#41:  φ(100, 0.35, 42) = 0.9420742512
```

Hiermit wäre diese Darstellung zu einem gewissen Abschluss gebracht.

(We want to show that there is a good approximation for p(k1 ≤ X ≤ k2) when we use the integral but use the limits k1-0.5 and k2+0.5).

Arrows and Labels for the Axes

Josef Böhm, Würmla

(Later versions of DERIVE allow annotations in all plot windows. So it is no more necessary to design one's own x and y to label the axes. Arrows might be still useful.)

Letters X and Y

$$\#1: \quad LX(m, n, sx, sy) := \begin{bmatrix} m - 0.05\cdot sx & n - 0.05\cdot sy \\ 0.05\cdot sx + m & 0.05\cdot sy + n \\ m & n \\ m - 0.05\cdot sx & 0.05\cdot sy + n \\ 0.05\cdot sx + m & n - 0.05\cdot sy \end{bmatrix}$$

$$\#2: \quad LY(m, n, sx, sy) := \begin{bmatrix} m - 0.05\cdot sx & n - 0.05\cdot sy \\ 0.05\cdot sx + m & 0.05\cdot sy + n \\ m & n \\ m - 0.05\cdot sx & 0.05\cdot sy + n \end{bmatrix}$$

The arrow heads up / down / right / left

$$\#3: \quad AU(m, n, sx, sy) := \begin{bmatrix} m - 0.04\cdot sx & n - 0.08\cdot sy \\ m & n \\ 0.04\cdot sx + m & n - 0.08\cdot sy \end{bmatrix}$$

$$\#4: \quad AD(m, n, sx, sy) := \begin{bmatrix} m - 0.04\cdot sx & n + 0.08\cdot sy \\ m & n \\ 0.04\cdot sx + m & n + 0.08\cdot sy \end{bmatrix}$$

$$\#5: \quad AR(m, n, sx, sy) := \begin{bmatrix} m - 0.08\cdot sx & n + 0.04\cdot sy \\ m & n \\ m - 0.08\cdot sx & n - 0.04\cdot sy \end{bmatrix}$$

$$\#6: \quad AL(m, n, sx, sy) := \begin{bmatrix} m + 0.08\cdot sx & n + 0.04\cdot sy \\ m & n \\ m + 0.08\cdot sx & n - 0.04\cdot sy \end{bmatrix}$$

sx and sy are the values for the scales in x- and y-direction – or multiples for bigger arrows and letters.
(m, n) is the location of the center of the letters and the peaks of the arrows.

$$\#25: \quad y = 4 - \frac{x^2}{2}$$

#26: AU(0, 5, 2, 4)

#27: AR(4, 0, 2, 4)

#28: LX(4.2, 0.3, 1, 2)

#29: LY(0.3, 5.2, 1, 2)