



Introduction to Scientific and Engineering Computing, BIL108E

Karaman

INTRODUCTION TO SCIENTIFIC & ENGINEERING COMPUTING BIL 108E, CRN24023

Dr. S. Gökhan Karaman

Technical University of Istanbul

March 29, 2010



Introduction to Scientific and Engineering Computing, BIL108E

Karaman

TENTATIVE SCHEDULE

Week	Date	Topics
1	Feb. 10	Introduction to Scientific and Engineering Computing
2	Feb. 17	Introduction to Program Computing Environment
3	Feb. 24	Variables, Operations and Simple Plot
4	Mar. 03	Algorithms and Logic Operators
5	Mar. 10	Flow Control, Errors and Source of Errors
6	Mar. 17	Functions
6	Mar. 20	Exam 1
7	Mar. 24	Arrays
8	Mar. 31	Solving of Simple Equations
9	Apr. 07	Polynomials Examples
10	Apr. 14	Applications of Curve Fitting
11	Apr. 21	Applications of Interpolation
11	Apr. 24	Exam 2
12	Apr. 28	Applications of Numerical Integration
13	May 05	Symbolic Mathematics
14	May 12	Ordinary Differential Equation (ODE) Solutions with Built-in Functions



Introduction to Scientific and Engineering Computing, BIL108E

Karaman

LECTURE # 8

LECTURE # 8

1 POLYNOMIALS

2 APPROXIMATION OF DATA



Introduction to Scientific and Engineering Computing, BIL108E

Karaman

POLYNOMIALS

POLYNOMIALS

- Definition: n^{th} degree polynomial

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_2 x^2 + a_1 x + a_0$$

- Coefficients of the polynomial

$$a_n, a_{n-1}, \dots, a_2, a_1, a_0$$

- n : degree of the polynomial



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

POLYNOMIALS

MATLAB toolbox: **polyfun**

- `polyval`: returns the value of the polynomial at a point x .
Input arguments are vector p and vector x
 $p \rightarrow a_n, a_{n-1}, \dots, a_1, a_0$
 x is the abscissae, where the polynomials is evaluated.
- `y = polyval(p, x)`



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

- Given: $p(x) = x^7 + 3x^2 - 1$,
 $x_k = -1 + k/4$ for $k = 0, \dots, 8$
- Find: Plot the given function.



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /home/sept/Desktop
Command Window
>> help polyval
POLYVAL Evaluate polynomial.
Y = POLYVAL(P,X) returns the value of a polynomial P evaluated at X. P
is a vector of length N+1 whose elements are the coefficients of the
polynomial in descending powers.

      Y = P(1)*X^N + P(2)*X^(N-1) + ... + P(N)*X + P(N+1)

If X is a matrix or vector, the polynomial is evaluated at all
points in X. See POLYVALM for evaluation in a matrix sense.

[Y,DELTA] = POLYVAL(P,X,S) uses the optional output structure S created
by POLYFIT to generate prediction error estimates DELTA. DELTA is an
estimate of the standard deviation of the error in predicting a future
observation at X by P(X).

If the coefficients in P are least squares estimates computed by
POLYFIT, and the errors in the data input to POLYFIT are independent,
normal, with constant variance, then Y +/- DELTA will contain at least

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /home/sept/Desktop
Command Window
If the coefficients in P are least squares estimates computed by
POLYFIT, and the errors in the data input to POLYFIT are independent,
normal, with constant variance, then Y +/- DELTA will contain at least
50% of future observations at X.

Y = POLYVAL(P,X,[ ],MU) or [Y,DELTA] = POLYVAL(P,X,S,MU) uses XHAT =
(X-MU(1))/MU(2) in place of X. The centering and scaling parameters MU
are optional output computed by POLYVALM.

Class support for inputs P,X,S,MU:
Float: double, single

See also polyfit, polyvalm.

Overloaded methods:
gf/polyval

Reference page in Help browser
doc polyval

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

% 7 6 5 4 3 2 1 0
p = [1 0 0 0 0 3 0 -1];
x = [-1:0.25:1];
y = polyval(p,x);
plot(x, y)

```

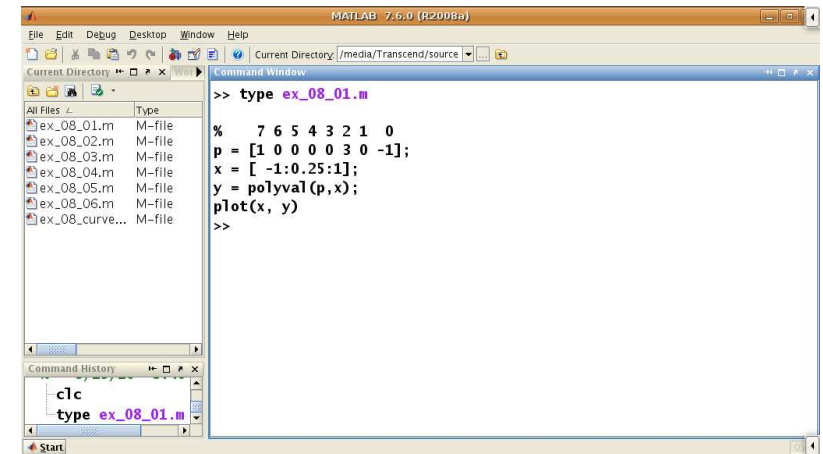


POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

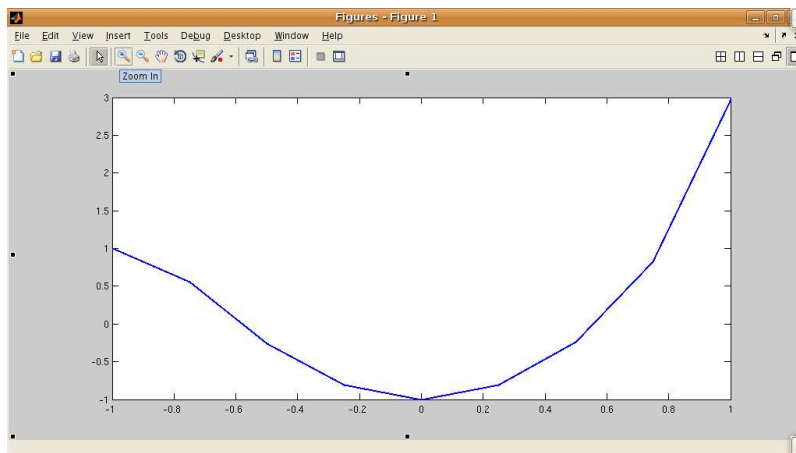


POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

POLYNOMIALS

- roots: provides an approximation of the zeros of a polynomial.
- Usage: r = roots(p)
- poly: returns the coefficients of the polynomial, whose zeros are given.
- Usage: p = poly(r)



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

- Given: $p(x) = x^3 - 6x^2 + 11x - 6$
- Find: Compute the zeros of the polynomial.



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```
p = [1 -6 11 -6]; format long;
roots(p)
```

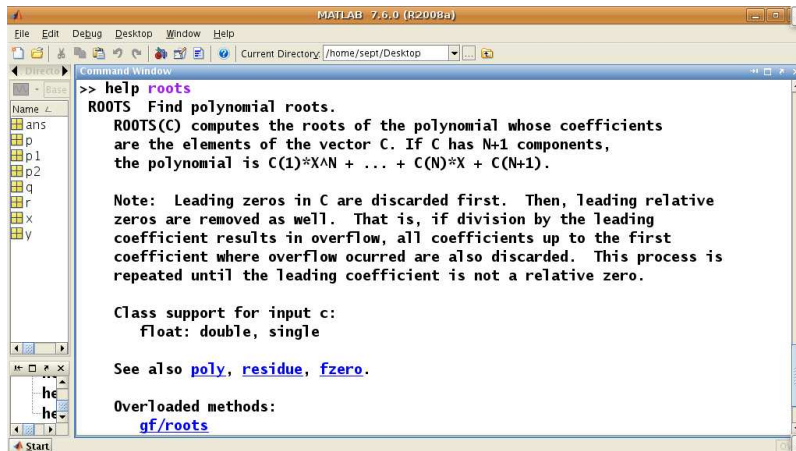


POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:



```
MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /home/sept/Desktop
Command Window
>> help roots
ROOTS Find polynomial roots.
ROOTS(C) computes the roots of the polynomial whose coefficients
are the elements of the vector C. If C has N+1 components,
the polynomial is C(1)*X^N + ... + C(N)*X + C(N+1).

Note: Leading zeros in C are discarded first. Then, leading relative
zeros are removed as well. That is, if division by the leading
coefficient results in overflow, all coefficients up to the first
coefficient where overflow occurred are also discarded. This process is
repeated until the leading coefficient is not a relative zero.

Class support for input c:
Float: double, single

See also poly, residue, fzero.

Overloaded methods:
gf/roots
```

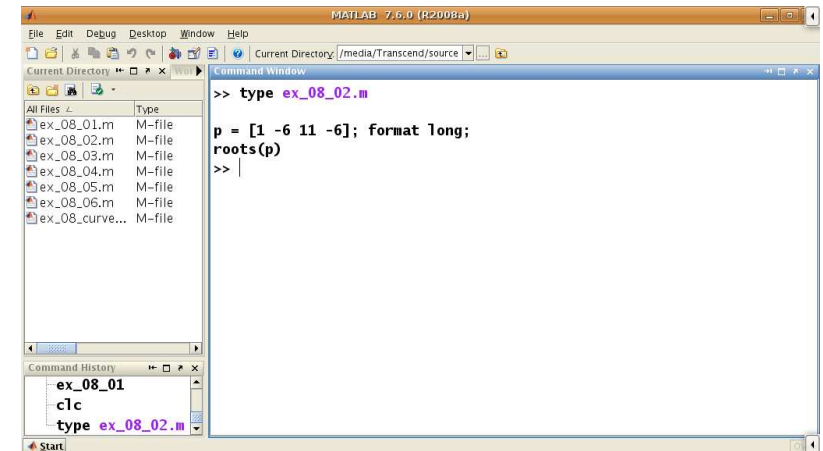


POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:



```
MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_08_02.m
p = [1 -6 11 -6]; format long;
roots(p)
>> |

Command History
ex_08_01
clc
type ex_08_02.m
```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transsend/source
Command Window
All Files z
Type
ex_08_01.m M-file
ex_08_02.m M-file
ex_08_03.m M-file
ex_08_04.m M-file
ex_08_05.m M-file
ex_08_06.m M-file
ex_08_curve... M-file

>> type ex_08_02.m
p = [1 -6 11 -6]; format long;
roots(p)
>> ex_08_02

ans =

3.000000000000002
1.999999999999998
1.000000000000000

>> |

Command History
clc
type ex_08_02.m
ex_08_02
Start

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE: poly

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /home/sept/Desktop
Command Window
>> help poly
POLY Convert roots to polynomial.
POLY(A), when A is an N by N matrix, is a row vector with
N+1 elements which are the coefficients of the
characteristic polynomial, DET(lambda*EYE(SIZE(A)) - A) .

POLY(V), when V is a vector, is a vector whose elements are
the coefficients of the polynomial whose roots are the
elements of V . For vectors, ROOTS and POLY are inverse
functions of each other, up to ordering, scaling, and
roundoff error.

ROOTS(POLY(1:20)) generates Wilkinson's famous example.

Class support for inputs A,V:
Float: double, single

See also roots, conv, residue, polyval.
Start

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /home/sept/Desktop
Command Window
>> poly(1:20)

ans =

Columns 1 through 6
1.0000e+00 -2.1000e+02 2.0615e+04 -1.2568e+06 5.3328e+07 -1.6723e+09

Columns 7 through 12
4.0172e+10 -7.5611e+11 1.1310e+13 -1.3559e+14 1.3075e+15 -1.0142e+16

Columns 13 through 18
6.3031e+16 -3.1133e+17 1.2066e+18 -3.6000e+18 8.0378e+18 -1.2871e+19

Columns 19 through 21
1.3804e+19 -8.7529e+18 2.4329e+18
Start

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /home/sept/Desktop
Command Window
>> roots(poly(1:20))

ans =

2.0000e+01
1.8998e+01
1.8006e+01
1.6989e+01
1.6011e+01
1.4996e+01
1.3996e+01
1.3008e+01
1.1994e+01
1.1002e+01
9.9998e+00
8.9998e+00
8.0001e+00
7.0000e+00
6.0000e+00
Start

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

The result is not always accurate.

Given: $p(x) = (x + 1)^7$

Find: Compute the zeros of the polynomial.

Answer: $\alpha = -1$

In fact, numerical methods for the computation of the polynomial roots with multiplicity larger than one are particularly subject to roundoff errors.



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```
p = [1 7 21 35 35 21 7 1];
roots(p)
```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_08_03.m
p = [1 7 21 35 35 21 7 1];
roots(p)
>>

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_08_03.m
p = [1 7 21 35 35 21 7 1];
roots(p)
>> ex_08_03
ans =
-1.008976439902066
-1.005627465048996 + 0.007012224774421i
-1.005627465048996 - 0.007012224774421i
-0.998023795747401 + 0.008801238285198i
-0.998023795747401 - 0.008801238285198i
-0.991860519252570 + 0.003934467943721i
-0.991860519252570 - 0.003934467943721i
Command History
ex_08_02
clc
type ex_08_03.m
>>

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

ADDITION AND SUBTRACTION OF POLYNOMIALS

- Polynomial addition and subtraction is the same as vector addition/subtraction operators.
- If the order of two polynomials (size of two vectors) does not match, zero should be added in order to match size of vectors.



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

- Given: $p_1(x) = x^4 - 1$,
 $p_2(x) = x^3 - 1$
- Find: Compute the sum of two polynomials.
- Answer: $p = x^4 + x^3 - 2$



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```
p1 = [1 0 0 0 -1];
p2 = [1 0 0 -1];
disp(p1);
disp(p2);
p = p1 + [0 p2]
```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```
MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_08_06.m
p1 = [1 0 0 0 -1];
p2 = [1 0 0 -1];
disp(p1);
disp(p2);
p = p1 + [0 p2]
>> |

Command History
conv(p2,q)+r
clc
type ex_08_06.m
```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transend/source
Command Window
>> type ex_08_06.m
p1 = [1 0 0 0 -1];
p2 = [1 0 0 -1];
disp(p1);
disp(p2);
p = p1 + [0 p2]
>> ex_08_06
1 0 0 0 -1
1 0 0 -1
p =
1 1 0 0 -2
>> |
Command History
clc
type ex_08_06.m
ex_08_06

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MULTIPLICATION AND DIVISION OF POLYNOMIALS

- `conv`: returns the coefficients of the polynomial given by the product of two polynomials.
- Usage: `conv(p1, p2)`
- `deconv`: provides the coefficients of the polynomials obtained on dividing p1 by p2.
- Usage: `[q, r]=deconv(p1, p2)`
q: quotient of the division,
r: remainder of the division
 $p_1(x) = q(x)p_2(x) + r(x)$



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

- Given: $p_1(x) = x^4 - 1$,
 $p_2(x) = x^3 - 1$
- Find: Compute the product of two polynomials.
- Answer: $p = x^7 - x^4 - x^3 + 1$



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit View Graphics Debug Desktop Window Help
Current Directory: /home/sept/Desktop
Workspace
Command Window
>> help conv
CONV Convolution and polynomial multiplication.
C = CONV(A, B) convolves vectors A and B. The resulting
vector is length LENGTH(A)+LENGTH(B)-1.
If A and B are vectors of polynomial coefficients, convolving
them is equivalent to multiplying the two polynomials.

Class support for inputs A,B:
float: double, single

See also deconv, conv2, convn, filter and, in the signal
Processing Toolbox, xcorr, convmtx.

Overloaded methods:
gf/conv

Reference page in Help browser
doc conv

```




POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```
p1 = [1 0 0 0 -1];
p2 = [1 0 0 -1];
p=conv(p1 ,p2)
```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_08_04.m
p1 = [1 0 0 0 -1];
p2 = [1 0 0 -1];
p=conv(p1 ,p2)
>> |

Command History
ex_08_03
clc
type ex_08_04.m
Start

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_08_04.m
p1 = [1 0 0 0 -1];
p2 = [1 0 0 -1];
p=conv(p1 ,p2)
>> ex_08_04
p =
    1     0     0    -1    -1     0     0     1
>>

Command History
clc
type ex_08_04.m
ex_08_04
Start

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

- Given: $p_1(x) = x^4 - 1$,
 $p_2(x) = x^3 - 1$
- Find: Compute the division of two polynomials.
- Answer: $q(x) = x$, $r(x) = x - 1$



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
Current Directory: /home/sept/Desktop
Command Window
>> help deconv
DECONV Deconvolution and polynomial division.
[Q,R] = DECONV(B,A) deconvolves vector A out of vector B. The re
is returned in vector Q and the remainder in vector R such that
B = conv(A,Q) + R.

If A and B are vectors of polynomial coefficients, deconvolution
is equivalent to polynomial division. The result of dividing B b
A is quotient Q and remainder R.

Class support for inputs B,A:
float: double, single

See also conv, residue.

Overloaded methods:
gf/deconv

Command History
c1c
help conv
help deconv

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

p1 = [1 0 0 0 -1];
p2 = [1 0 0 -1];
[q,r]= deconv(p1 ,p2)

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
Current Directory: /media/Transcend/source
Command Window
>> type ex_08_05.m
p1 = [1 0 0 0 -1];
p2 = [1 0 0 -1];
[q,r]= deconv(p1 ,p2)
>>

Command History
ex_08_04
c1c
type ex_08_05.m

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
Current Directory: /media/Transcend/source
Command Window
>> type ex_08_05.m
p1 = [1 0 0 0 -1];
p2 = [1 0 0 -1];
[q,r]= deconv(p1 ,p2)
>> ex_08_05
q =
    1     0

r =
    0     0     0     1    -1

Command History
c1c
type ex_08_05.m
ex_08_05
>>

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transsend/source
Command Window
p1 = [1 0 0 0 -1];
p2 = [1 0 0 -1];
[q,r]= deconv(p1 ,p2)
>> ex_08_05

q =

    1     0

r =

    0     0     0     1    -1

>> conv(p2,q)+r

ans =

    1     0     0     0    -1

Command History
type ex_08_05.m
ex_08_05
conv(p2,q)+r

```



POLYNOMIALS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

POLYNOMIALS

- `polyint` returns the coefficients of the primitive of the polynomial.
Usage: $y = \text{polyint}(p)$,
 y : coefficients of $\int_0^x p(t) dt$
- `polyder`: returns the derivative of the polynomial, whose coefficients are given by the components of the vector p .
Usage: $y = \text{polyder}(p)$,
 y : coefficients of $p'(x)$



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

CURVE FITTING

- Approximating a function f consists of replacing it by another function \tilde{f} .
- A function f can be replaced in a given interval by its Taylor polynomial.
- It requires the knowledge of f and its derivatives up to the order n at a given point x_0 .



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

CURVE FITTING

- Taylor polynomial may fail to accurately represent f far enough from the point x_0 .
- Use `taylorTool` for the computation of Taylor's polynomial of arbitrary degree for any given function f .
- The agreement between the function and its Taylor polynomial is very good in a small neighborhood of x_0 .



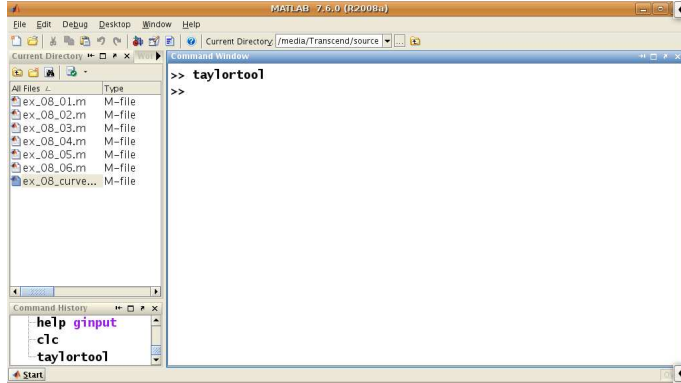
CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

taylortool

EXAMPLE



Comparison between the function $f(x) = 1/x$ and its Taylor polynomial of degree 10 for the point $x_0 = 1$.

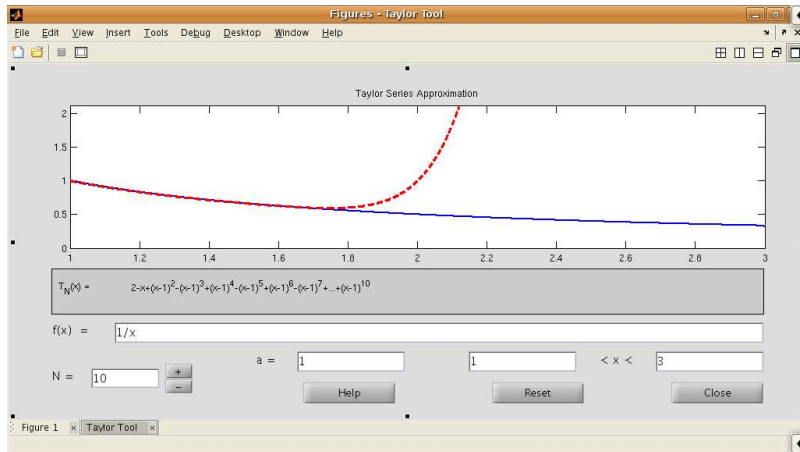


CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

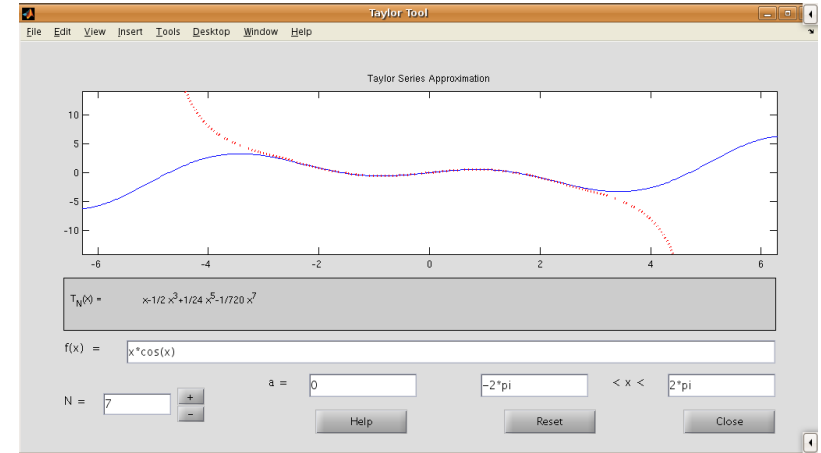


CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

CURVE FITTING

- A function is known only through its values at some given points.
- $n + 1$ couples (x_i, y_i) are given, $i = 0, 1, 2, \dots, n + 1$.
- Approximate function \tilde{f} :
 $\tilde{f}(x_i) = y_i$, where $i = 0, 1, 2, \dots, n$
- \tilde{f} is called **interpolant** of the set of data
- There exist different types of interpolant:
 - Polynomial interpolant
 - Trigonometric interpolant
 - Rational interpolant

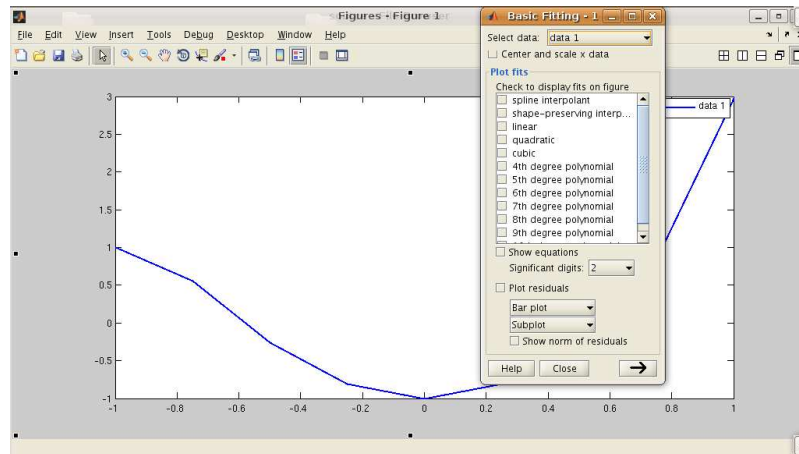


CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

BASIC FITTING:



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

CURVE FITTING

- The most common method of finding the best fit to data point is the least squares method.
- polyfit function uses least squares method.
- polyfit function returns the coefficients of a polynomial for a given data set (x_i, y_i) .
- Usage: $p = \text{polyfit}(x, y, n)$
 n : degree of the polynomial.
 x and y : data set (x_i, y_i) .



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE

In the table below we report the values of the sea water density ρ (in kg/m^3) corresponding to different values of the temperature T (in degrees Celsius):

T	4°	8°	12°	16°	20°
ρ	1000.7794	1000.6427	1000.2805	999.7165	998.9700



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

T = [4 8 12 16 20];
rho = [1000.7794 1000.6427 1000.2805 999.7165 998.9700];
px = polyfit(T, rho, 3)
Tx = linspace(4,20,100);
rhox=polyval(px, Tx)
plot(T, rho, 'o')
hold('on')
plot(Tx, rhox, '-')
grid('on')
xlabel('T')
ylabel('rho')
legend('data', 'curve fit')

```



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transsend/source
Workspace
Command Window
>> type ex_08_36.m
Name Value
ans [1.3967, 3]
p [1, 1, 0, 0, -]
p1 [1, 0, 0, 0, -]
p2 [1, 0, 0, -1]
q [1, 0]
r [0, 0, 0, 1, -]
rho [1.0008e]
x [-1, -0.75]
y [1, 0.5540]
T = [4 8 12 16 20];
rho = [1000.7794 1000.6427 1000.2805 999.7165 998.9700];
px = polyfit(T, rho, 3);
Tx = linspace(4,20,100);
rhox=polyval(px, Tx);
plot(T, rho, 'o')
hold('on')
plot(Tx, rhox, '-');
grid('on')
xlabel('T')
ylabel('rho')
legend('data', 'curve fit')
>>
Command History
c1c
type ex_08_36
Start

```

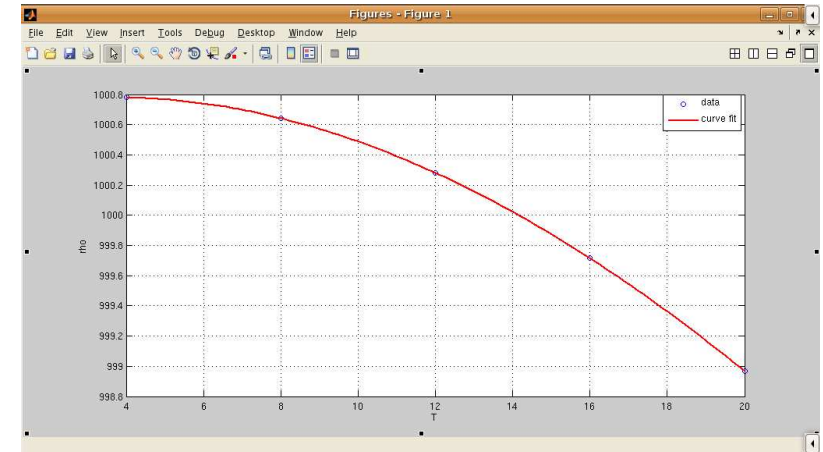


CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

LEAST SQUARES METHOD

- If the degree of the polynomial increases, interpolation does not guarantee a better approximation of a given function.
- In least – squares approximation we look for an approximant \tilde{f} which is a polynomial of degree m (typically, $m \ll n$) that minimizes the mean-square error $\frac{1}{n} \sum_{i=0}^n [y_i - \tilde{f}(x_i)]^2$. The same minimization criterion can be applied for a class of functions that are not polynomials.



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

CURVE FITTING WITH FUNCTIONS OTHER THAN POLYNOMIALS

Rewrite the function in a first degree polynomial form.

- Power function:
 $y = b x^m \rightarrow \ln y = m \ln x + \ln b$
- Exponential function:
 $y = b e^{mx} \rightarrow \ln y = mx + \ln b$
- Logarithmic function:
 $y = m \ln x + b \rightarrow y = m \ln x + b$
- Reciprocal function:
 $y = 1/(mx + b) \rightarrow 1/y = mx + b$



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

FUNCTION SELECTION

- For a given data it is possible to foresee which of the functions have the potential for providing a good fit.
- This is done by plotting the data using different combinations of linear and logarithmic axes.

x-axis	y-axis	function
linear	linear	$y = mx + b$
logarithmic	logarithmic	$y = bx^m$
linear	logarithmic	$y = be^{mx}$
logarithmic	linear	$y = m \ln x + b$
linear	linear	$y = 1/(mx + b)$



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

FUNCTION SELECTION

- Exponential functions can not pass through the origin.
- Exponential functions can only fit data with all positive y's or all negative y's.
- Logarithmic functions cannot model $x = 0$, or negative values of x .
- For the power function $y = 0$ when $x = 0$.
- The reciprocal equation cannot model $y = 0$.



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

w	0.64	0.73	0.96	1.21	1.49	1.83	2.41	3.15	3.70	4.83	6.00
t	5.0	4.5	4.0	3.5	3.0	2.5	2.0	1.5	1.0	0.5	0.0

- Choose the function for the given data.



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
All Files - Type
ex_08_01.m M-file
ex_08_02.m M-file
ex_08_03.m M-file
ex_08_04.m M-file
ex_08_05.m M-file
ex_08_06.m M-file
ex_08_curve... M-file
ex_08_funcs... M-file
ex_08_funcs... M-file
ex_08_funcs... M-file
ex_08_funcs... M-file
ex_08_funcs... M-file
ex_08_36.m M-file
ex_08_36.m~ Editor A
ex_08_polyfi... M-file

Command Window
>> type ex_08_funcse11.m
t=0:0.5:5;
w=[6 4.83 3.7 3.15 2.41 1.83 1.49 1.21 0.96 0.73 0.64];
plot(t, w, 'o')
xlabel('linear t')
ylabel('linear w')
>> |

Command History
c1c
type ex_08_fur
Start

```

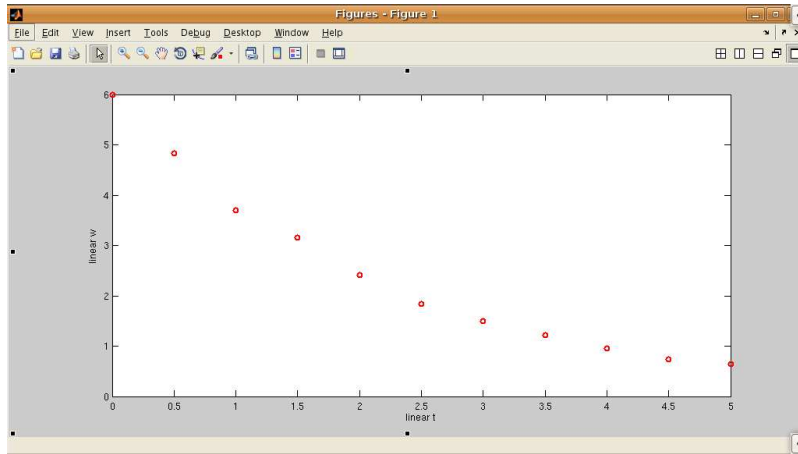


CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
Current Directory: /media/Transcend/source
Command Window
>> type ex_08_funcse12.m
t=0:0.5:5;
w=[6 4.83 3.7 3.15 2.41 1.83 1.49 1.21 0.96 0.73 0.64];
semilogy(w, t, 'o')
ylabel('log w')
xlabel('linear t')
>>
Command History
c1c
type ex_08_fun
Start
  
```

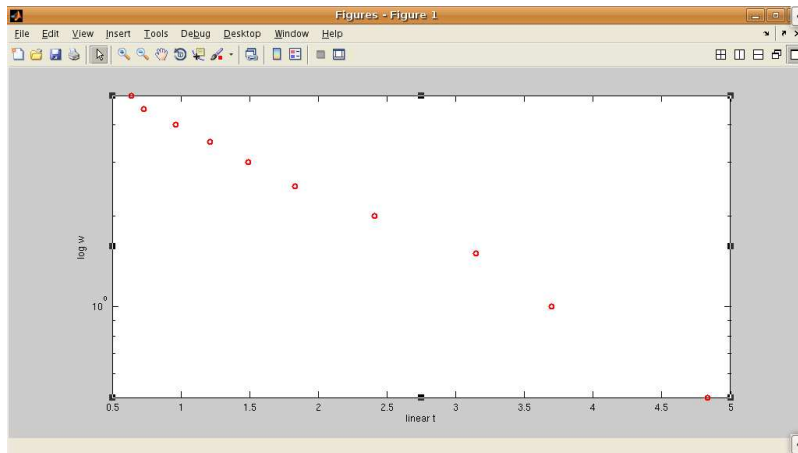


CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
Current Directory: /media/Transcend/source
Command Window
>> type ex_08_funcse13.m
t=0:0.5:5;
w=[6 4.83 3.7 3.15 2.41 1.83 1.49 1.21 0.96 0.73 0.64];
plot(t, 1./w, 'o')
xlabel('linear t')
ylabel('linear w')
>> |
Command History
c1c
type ex_08_fur
Start
  
```

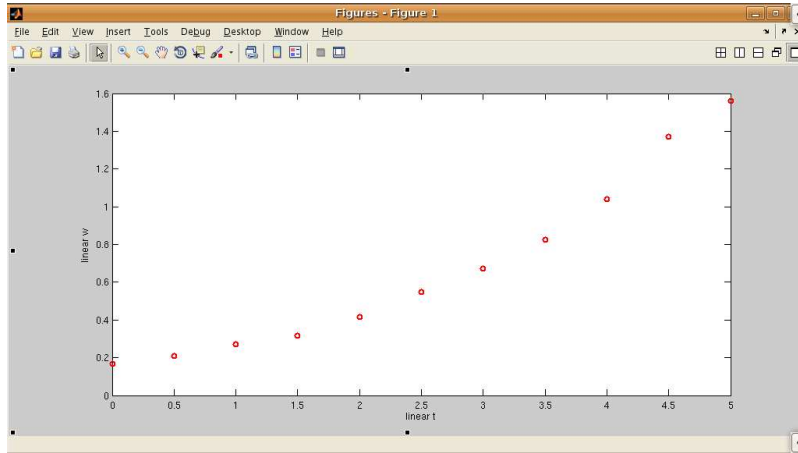



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

1- t = 0:0.5:5;
2- w = [6.00 4.83 3.70 3.15 2.41 1.83 1.49 1.21 0.96 0.73 0.64];
3- p = polyfit(t,log(w),1);
4- m = p(1);
5- b = exp(p(2)) %determine the coefficient b
6- ti = 0:0.1:5;
7- wi = b * exp(m * ti); %calculate the fnc. value at each element of tm
8- plot(t,w,'o', ti, wi) %plot the daya points and the function
9- grid
10- xlabel('t, ti')
11- ylabel('w, wi')
12- legend('data','polyfit')

```



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:

```

MATLAB 7.6.0 (R2008a)
>> clf
>> ex_08_funcsel4

m =

    -4.5801e-01

b =

    5.9889e+00

```

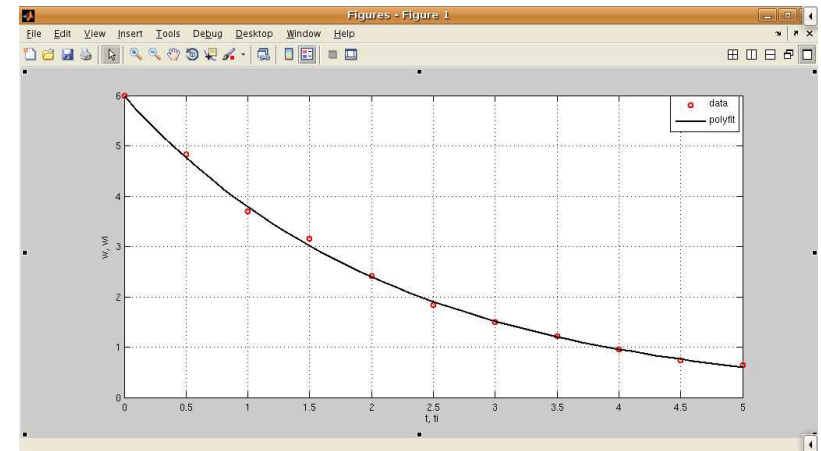


CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE cont'd.:





CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

Latitude	δ_K			
	$K = 0.67$	$K = 1.5$	$K = 2.0$	$K = 3.0$
65	-3.1	3.52	6.05	9.3
55	-3.22	3.62	6.02	9.3
45	-3.3	3.65	5.92	9.17
35	-3.32	3.52	5.7	8.82
25	-3.17	3.47	5.3	8.1
15	-3.07	3.25	5.02	7.52
5	-3.02	3.15	4.95	7.3
-5	-3.02	3.15	4.97	7.35
-15	-3.12	3.2	5.07	7.62
-25	-3.2	3.27	5.35	8.22
-35	-3.35	3.52	5.62	8.8
-45	-3.37	3.7	5.95	9.25
-55	-3.25	3.7	6.1	9.5

Variation of the average yearly temperature on the Earth for four different values of the concentration K of carbon acid at different latitudes.

- Compute the least-squares polynomial of degree 4 that approximates the values of K reported in the Table given above.



CURVE FITTING

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

- The price (in euros) of a magazine has changed as follows:

Nov.87	Dec.88	Nov.90	Jan.93	Jan.95	Jan.96	Nov.96	Nov.00
4.5	5.0	6.0	6.5	7.0	7.5	8.0	8.0

Estimate the price in November 2002 by extrapolating these data.



References

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

References for Week 8

- 1 Alfio Quarteroni, Fausto Saleri, Scientific Computing with Matlab and Octave, Springer, 2006.
- 2 Brian Hahn, Daniel T.Valentine, Essential Matlab for Engineers and Scientists, Elsevier, 2010.