Matlab's Symbolic Toolbox

ChEn 1703

THE UNIVERSITY OF UTAH

Some Capabilities of the Symbolic Toolbox

NOTE: The symbolic toolbox is included in the student version. It must be purchased separately in general.

- Algebra - manipulate expressions
- Solve systems of linear equations
  - Other linear algebra functions also.
- Solve systems of nonlinear equations
- Calculus - integration & differentiation of expressions
- Solve differential equations
- Fourier & Laplace transforms
- "`help symbolic`"

Building Symbolic Variables and Expressions

- Normal variables in MATLAB have a value (number, string, etc)
- Symbolic variables are special, and are declared differently

```
syms a b c x;
```

- Symbolic expressions may also be created:

```
syms x y; % declare symbolic variables.
x=y^2;    % build a symbolic expression.
```

```
x=sym('y^2');
```

Use "sym" to distinguish this from a text string...

Note usage of "sym" versus "syms" - use "sym" for an expression and "syms" to declare a bunch of variables.

- You can do arithmetic with symbolic expressions

```
syms a b c d x;
y1=a*x^2+b*x+c;
y2=d*x^2;
y3=y1+y2;
```

```
y1=sym('a*x^2+b*x+c');
y2=sym('sin(y1)');
```

THE UNIVERSITY OF UTAH

Some Useful Commands

Statement	Description	Example	
syms	Declare a symbolic variable	`syms a b c x;`	
pretty(f)	Print out the symbolic expression **f** with nice formatting	`pretty( sqrt(a*x)*exp(c) );`	
collect(f,v)	Collect coefficients in **f** in terms of powers of **v**.	`syms a b c x;` `collect(a*x^2+c+x^2+b*x+b+x/2);`	

expand(f)	Expands polynomials, trig functions, exponential functions.	`syms a b c x;`
		`expand( exp(a+b) );`
		`expand( (a*x+c)^2 );`
			$\exp(a+b) \rightarrow \exp(a)\exp(b)$
			$(ax^1+c)^2 \rightarrow ax^2+2acx+c^2$

factor(f)	Obtain the prime factors of a given number.
	If **f** is a polynomial, this factors the polynomial.
		`factor(90);`
		
		`syms x;`
		`factor(x^3-6*x^2+11*x-6)`
			2 3 3 5
			
			$(x-1)(x-2)(x-3)$

simplify(f)	*Attempts* to simplify the expression. Not always very useful.	`syms x;` `simplify( sin(x)^2+cos(x)^2 )`	$\sin^2(x)+\cos^2(x) \rightarrow 1$
subs(f,t,v)	Substitute **v** for **t** in the expression **f**.	`syms x a b c;` `subs( sin(pi*x), x, (a+b)/c )`	$\sin(\pi x) \rightarrow \sin\left(\pi\frac{(a+b)}{c}\right)$

THE UNIVERSITY OF UTAH

Simplification

$$
\begin{aligned}
y &= x^3 - 6x^2 + 11x - 6 \\
&= (x-1)(x-2)(x-3) \\
&= x(x(x-6)+11)-6
\end{aligned}
$$

Which form of "y" do you want?
It depends on the situation.

```
syms x;
f = x^3-6*x^2+11*x-6;
g = (x-1)*(x-2)*(x-3);
h = x*(x*(x-6)+11)-6;

pretty( collect(f) );
pretty( collect(g) );
pretty( collect(h) );
```

```
   3       2
  x  - 6 x  + 11 x - 6
```

```
syms x;
f = x^3-6*x^2+11*x-6;
g = (x-1)*(x-2)*(x-3);
h = x*(x*(x-6)+11)-6;

pretty( factor(f) );
pretty( factor(g) );
pretty( factor(h) );
```

```
(x - 1) (x - 2) (x - 3)
```

```
syms x;
f = x^3-6*x^2+11*x-6;
g = (x-1)*(x-2)*(x-3);
h = x*(x*(x-6)+11)-6;

pretty( horner(f) );
pretty( horner(g) );
pretty( horner(h) );
```

```
x (x (x - 6) + 11) - 6
```

"Horner" form is also referred to as "nested"

`simple` - tries to find the *shortest* representation of the expression.

Substitutions: **subexpr**

$$Ax = b$$ Linear system

$$A^{-1}Ax = A^{-1}b$$ Multiply by the "inverse" of A.

$$Ix = A^{-1}b$$ $A^{-1}A=I$ $\quad I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

$$x = A^{-1}b$$ Solution for x.

```
A=sym('[a11 a12 a13; a21 a22 a23; a31 a32 a33]');
Ainv=inv(A);
pretty(Ainv);
  [a22 a33 - a23 a32      a12 a33 - a13 a32    a12 a23 - a13 a22]
  [----------------- ,  - ----------------- ,  -----------------]
  [       %1                    %1                   %1        ]
  [                                                            ]
  [  a21 a33 - a23 a31   a11 a33 - a13 a31     a11 a23 - a13 a21]
  [- ----------------- , ----------------- , - -----------------]
  [         %1                  %1                   %1        ]
  [                                                            ]
  [a21 a32 - a22 a31      a11 a32 - a12 a31    a11 a22 - a12 a21]
  [----------------- ,  - ----------------- ,  -----------------]
  [       %1                    %1                   %1        ]

  %1 := a11 a22 a33 - a11 a23 a32 - a21 a12 a33 + a21 a13 a32 + a31 a12 a23

        - a31 a13 a22
[Ainv,s]=subexpr(Ainv,'s');
```

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

subexpr will replace the %1 generated by **pretty**.

$$A^{-1} = \begin{bmatrix} \frac{a_{22}a_{33}-a_{23}a_{32}}{s} & -\frac{a_{12}a_{33}-a_{13}a_{32}}{s} & \frac{a_{12}a_{23}-a_{13}a_{22}}{s} \\ -\frac{a_{21}a_{33}-a_{23}a_{31}}{s} & \frac{a_{11}a_{33}-a_{13}a_{31}}{s} & -\frac{a_{11}a_{23}-a_{13}a_{21}}{s} \\ \frac{a_{21}a_{32}-a_{22}a_{31}}{s} & -\frac{a_{11}a_{32}-a_{12}a_{31}}{s} & \frac{a_{11}a_{22}-a_{12}a_{21}}{s} \end{bmatrix}$$

$$s = a_{11}a_{22}a_{33} - a_{11}a_{23}a_{32} - a_{21}a_{12}a_{33} + a_{21}a_{13}a_{32} + a_{31}a_{12}a_{23} - a_{31}a_{13}a_{22}$$

Substitutions: **subs**

```
B=sym('[b11 b12;b21 b22]');
BI = inv(B);
pretty(BI);

  [        b22                      b12       ]
  [ -----------------     - -----------------]
  [ b11 b22 - b12 b21       b11 b22 - b12 b21]
  [                                           ]
  [         b21                    b11        ]
  [- -----------------     ----------------- ]
  [  b11 b22 - b12 b21     b11 b22 - b12 b21 ]

s=sym('b11*b22-b12*b21');
BIS = subs(BI,s,'s');
pretty(BIS);

  [ b22       b12]
  [ ---     - ---]
  [  s         s ]
  [              ]
  [  b21     b11 ]
  [- ---     --- ]
  [   s       s  ]
```

Create a matrix "B" and solve for its inverse.

`pretty` and **`subexpr`** don't rip the common expression out this time...

Substitute the symbol 's' for the denominator in each element of B^{-1}.

More with **subs**

More on "solve" next.

```
syms a b c x;
y = a*x^2+b*x+c;
xsoln = solve(y,x);
pretty(xsoln);

    [            2         1/2]
    [      b - (b  - 4 a c)   ]
    [- 1/2 -------------------]
    [              a          ]
    [                         ]
    [            2         1/2]
    [      b + (b  - 4 a c)   ]
    [- 1/2 -------------------]
    [              a          ]
```

$$y = ax^2 + bx + c$$

$$x = -\frac{b \pm \sqrt{b^2 - 4ac}}{2a}$$

Substitute variables in `xsoln` with those found in the workspace.

```
a=2; b=-5; c=3;
subs(xsoln)

    ans =
        1.5000
        1.0000

subs(xsoln,'a',pi);

 ans =
  -1/2*(b-(b^2-4*pi*c)^(1/2))/pi
  -1/2*(b+(b^2-4*pi*c)^(1/2))/pi
```

$$y = (2x - 3)(x + 2)$$

$$x = -\frac{b \pm \sqrt{b^2 - 4\pi c}}{2\pi}$$

THE UNIVERSITY OF UTAH

Solving Equations

```
syms a b c x;
y = a*x^2+b*x+c;      % quadratic eqn.
xroot = solve(y,x);   % solve for x.
asolv = solve(y,a);   % solve for a.
```

Solves $f(x)=0$.

Solves $f(a)=0$.

```
solve('y=a*x^2+b*x+c','x')
ans =
 -1/2*(b-(b^2+4*a*y-4*a*c)^(1/2))/a
 -1/2*(b+(b^2+4*a*y-4*a*c)^(1/2))/a

solve('y=a*x^2+b*x+c','a');
ans =
-(-y+b*x+c)/x^2
```

Solves $y = ax^2 + bx + c$ for x.

Solves $y = ax^2 + bx + c$ for a.

NOTE: here we don't need to use "`sym`" since `solve` expects a symbolic expression. Quotes do the trick. Of course, this also works with a `sym` expression (see above).

THE UNIVERSITY OF UTAH

Solving Multiple Equations

Example:

$$\begin{aligned} y_1 &= a_1 x_1 + b_1 \\ y_2 &= a_2 x_2 + b_2 \end{aligned}$$

```
S=solve('y1=a1*x1+b1','y2=a2*x2+b2');
```

Returns a "structure" with the solution variables, x_1, x_2.

```
S.x1 = (y1-b1)/a1
S.x2 = -(-y2+b2)/a2
```

Example:

$$\begin{aligned} x &= \sqrt{\frac{y-3}{2}} \\ y &= x^3 + x^2 - 6x + 3 \end{aligned}$$

Recall we solved this system numerically...

```
S = solve('x=sqrt((y-3)/2)','y=x^3+x^2-6*x+3');
fprintf('x=\n'); disp(S.x);
fprintf('y=\n'); disp(S.y);
```

```
x=  0  3
y=  3  21
```

A generalized version:

```
syms a b c d e f x y;
S = solve('x=sqrt((y-e)/f)',...
          'y=a*x^3+b*x^2+c*x+d',x,y)
disp(S.x);
```

Warning: ugly solution...