The University Of Sheffield.

MATLAB Basics 7 symbolic algebra

Anthony Rossiter

University of Sheffield

For a neat organisation of all videos and resources

http://controleducation.group.shef.ac.uk/indexwebbook.html

Introduction

1. The previous resources demonstrate how to use basic MATLAB functionality.
2. It is useful to next to consider how MATLAB supports basic algebra, that is, variables which do not contain numerical values.
3. This includes support for operations such as differentiation, integration, solving equations, function description and so forth.

Some times your need to automate mathematical operations, and the symbolic toolbox is an easy way of doing this (e.g. to find an exact tangent.)

Common symbolic algebra options

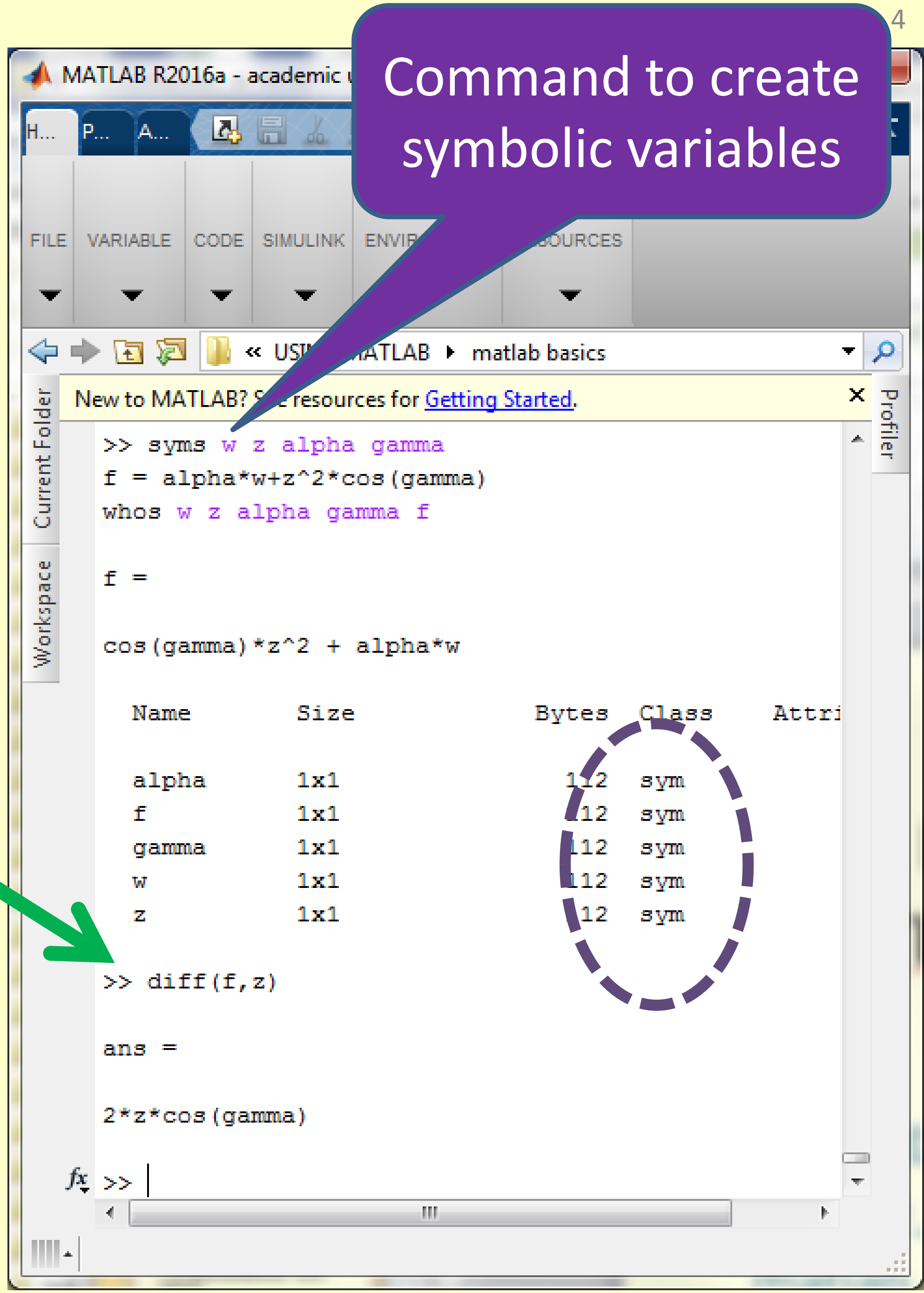
- Defining symbolic variables.
- Defining functions such as ***f(x)***.
- Differentiation and integration.
- Tangents and normals.
- Solution of ODEs.
- Taylor series.
- Solving equations.
- For a more complete list use: ***>> help symbolic***

Symbolic variables

Symbolic variables do not contain values and allow the user to write algebraic expressions.

This supports operations such as differentiation.

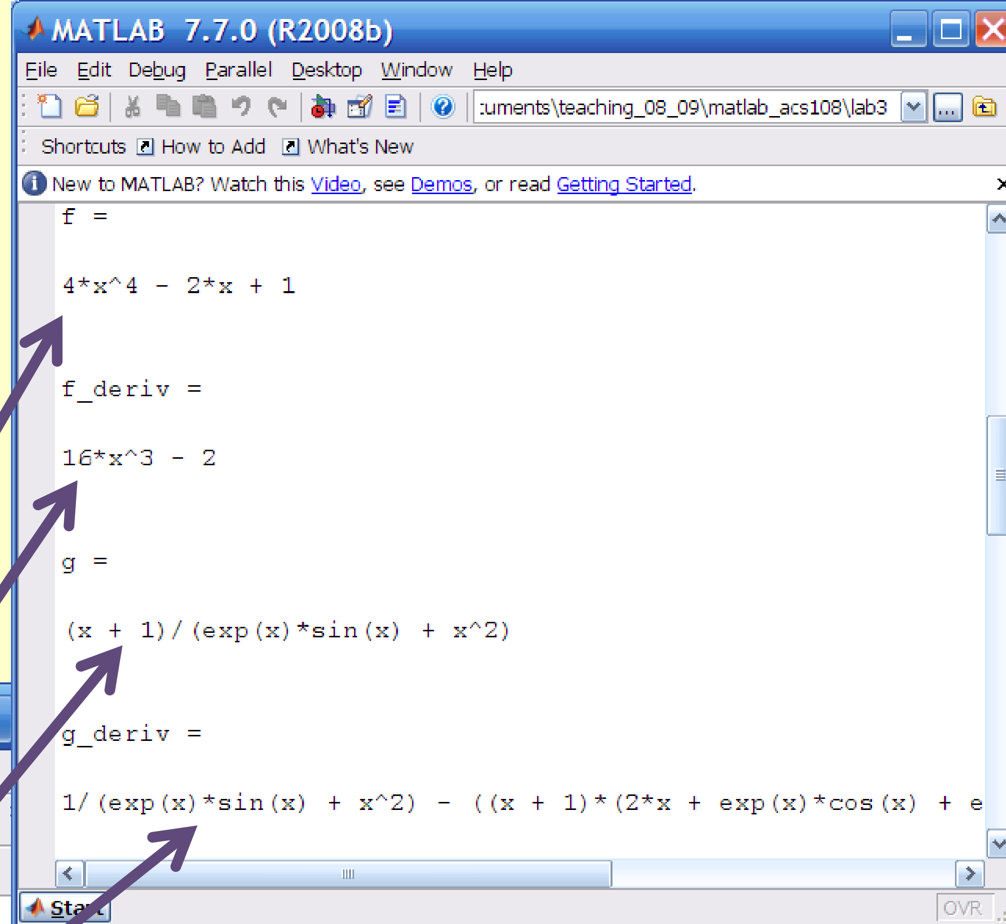
Matrix denotes these with the classification of ***"sym".***

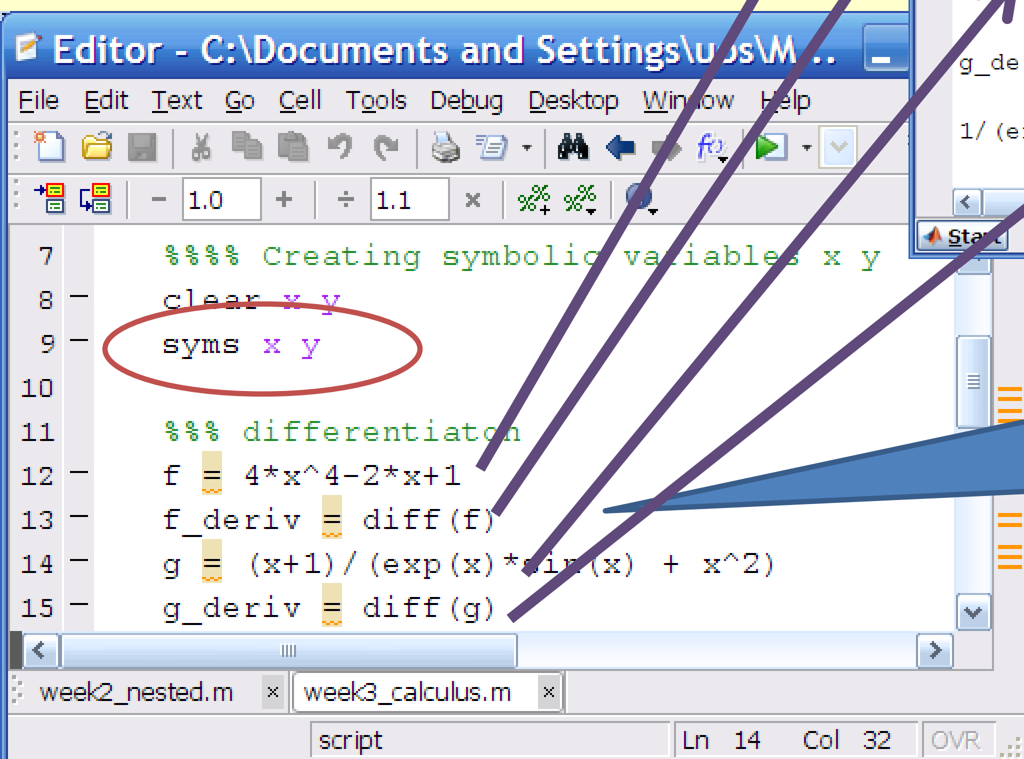


The University Of Sheffield.

Differentiation

- Define variables as symbolic.
- Use diff.m to differentiate.





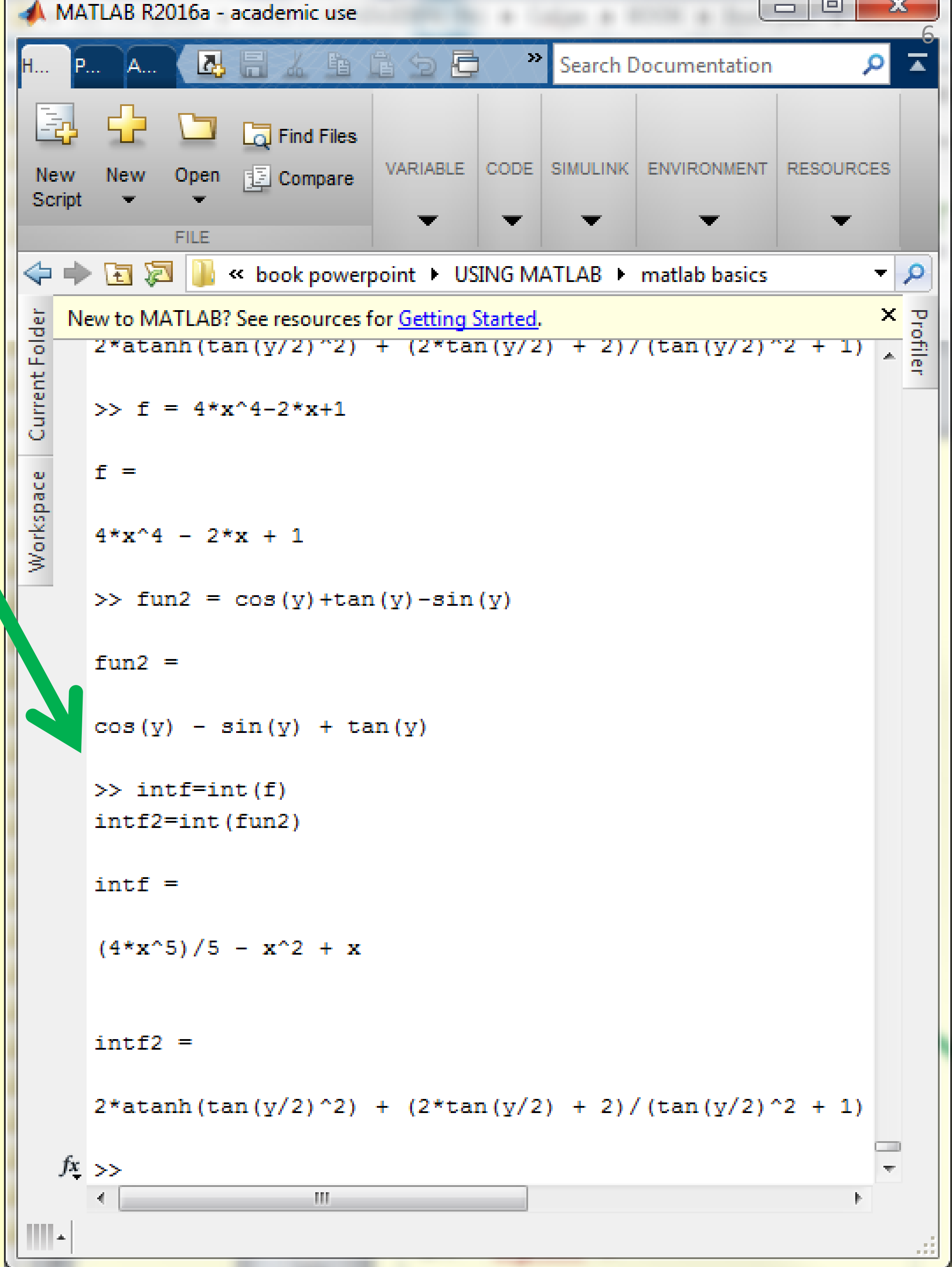
Assume symbolic expressions have just one variable so it is obvious what to differentiate wrt.

Integration

Assuming the function is symbolic, int.m will attempt to integrate.

May fail if integral has a non-simple form.

Also works for functions of many variables – **see 'help int' for instructions.**



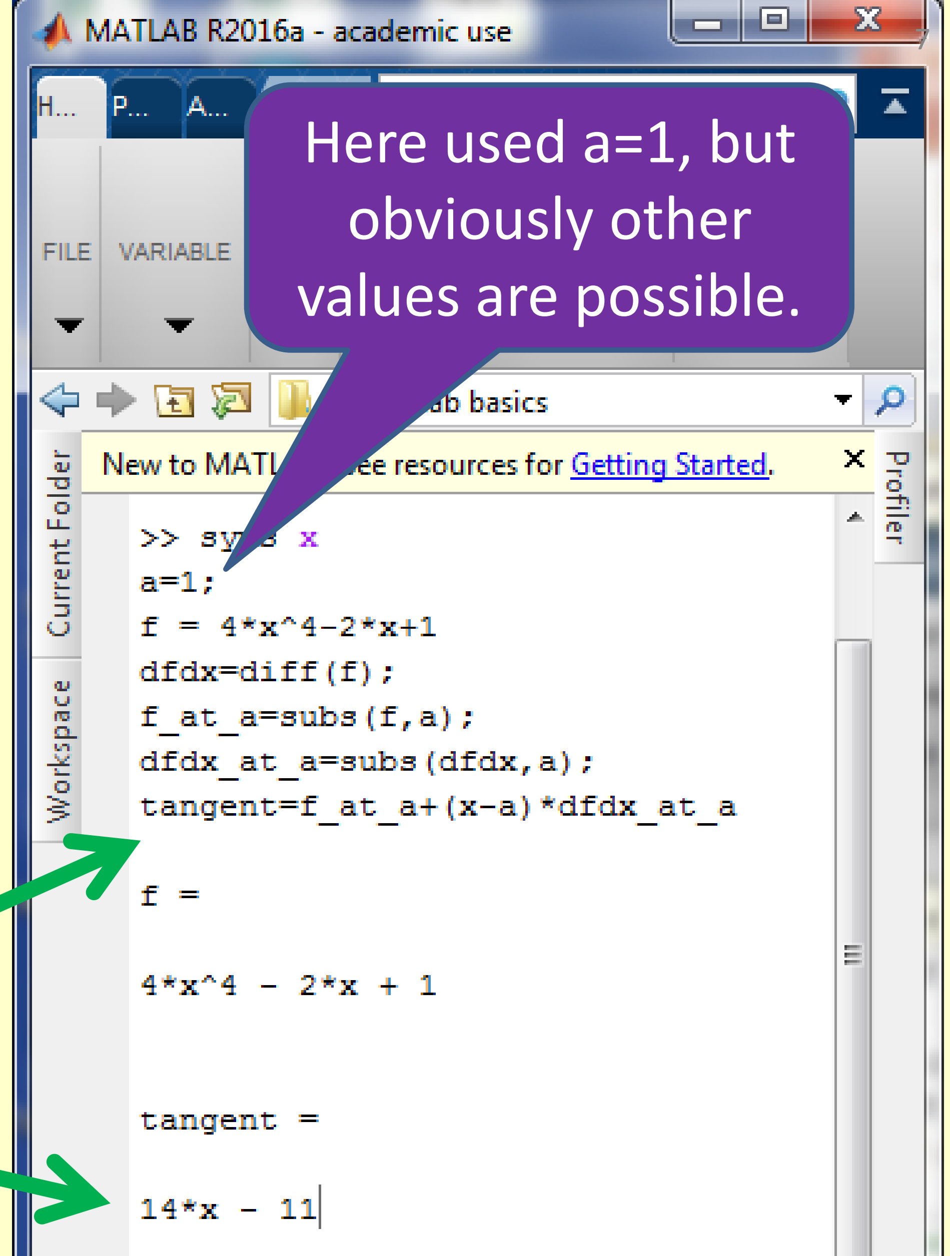
The University Of Sheffield.

Tangents

The expression for a tangent curve is known and all the terms can be obtained with the symbolic toolbox.

$$T(x) = f(a) + (x-a)\frac{df}{dx}(a)$$

Hence tangent line is determined automatically.



Here used a=1, but obviously other values are possible.

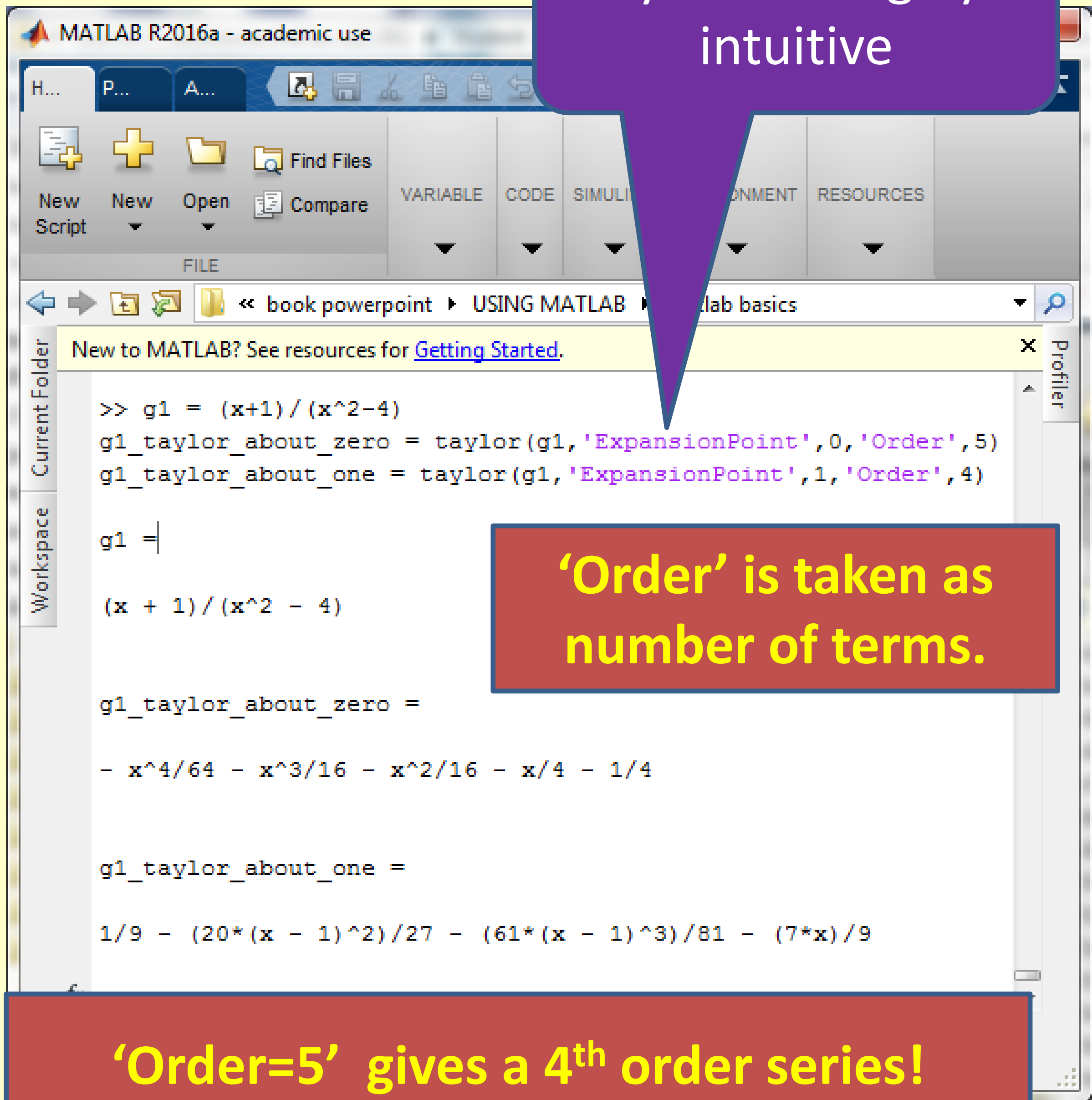
Normal calculations follow a similar methodology.

Taylor series

Automatic generation of Taylor series is very useful.

MATLAB provides the file ***taylor.m*** for this.

Use ***help taylor*** to explore functionality in more depth.



Syntax is largely intuitive

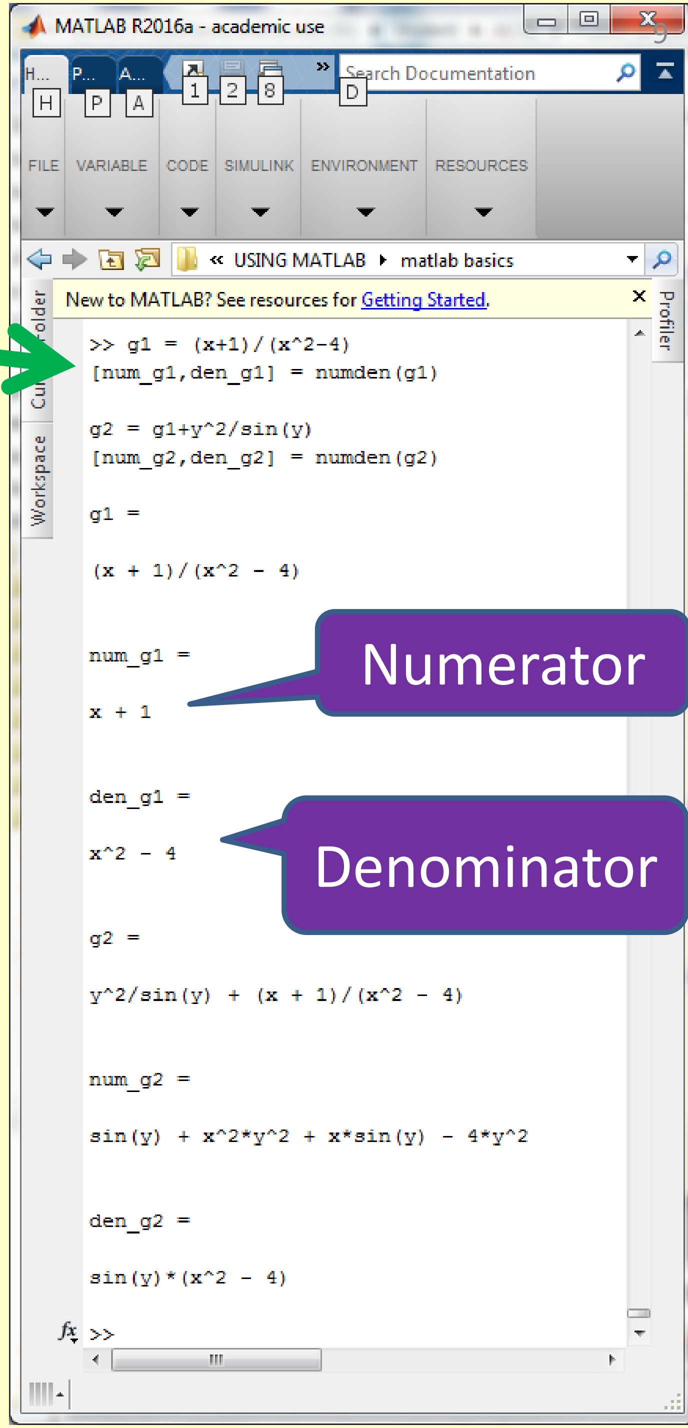
'Order' is taken as number of terms.

'Order=5' gives a 4^{th} order series!

Numerator and denominator

The function *numden.m* allows the user to automatically extract the numerator and denominator components.

This may be useful for some operations and decision making.

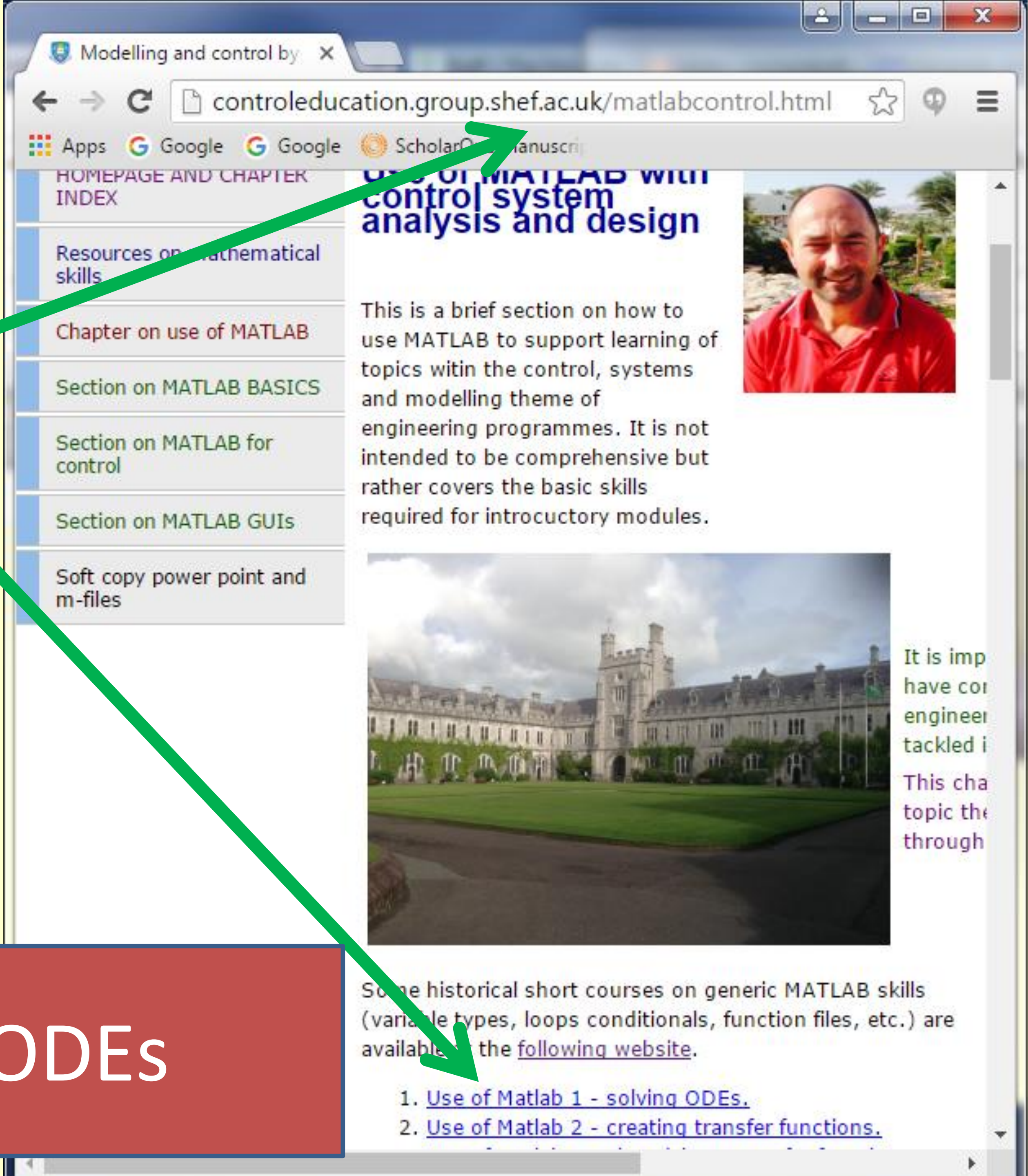


The University Of Sheffield.

There is an entire resource on ODEs in the **MATLAB for control** section.

Also see the direct link if desired:

https://www.youtube.com/watch?v=VcMeG_4BBdQ&feature=youtu.be



SOLVING ODEs

LIVE DEMONSTRATIONS WITH MATLAB

Go through the following to see examples

matlab_basics7.m

matlab_basics7b.m (with plotting)

Conclusions

Demonstrated the usefulness of the symbolic toolbox.

1. Allows complex or messy algebra to be handled by the computer rather than by hand.
2. This includes differentiation, integration, solution of ODEs, matrix inverse and so on.

REMARK: Symbolic expressions can be created from strings and also evaluated with specified numerical values where this is helpful.

For a neat organisation of all videos and resources

http://controleducation.group.shef.ac.uk/indexwebbook.html

Anthony Rossiter
Department of Automatic Control and Systems Engineering
University of Sheffield
www.shef.ac.uk/acse