Computing for Geometry and Number Theory
Lecture 2

Dr John Armstrong

King's College London

August 22, 2020

Basic Mathematica Continued

- Last week we used Mathematica as a calculator
- Using the workbook, for example to type SHIFT + Enter
- Arrays and lists and wrote a simple function
- We learned built in functions and constants start with a capital letter
- We learned about = and :=

Functions - Revision

```
solveQuadratic[a_,b_,c_]:=
(-b + Sqrt[b^2 - 4 a c ])/(2a)
solveQuadratic[1,2,1]
```

- Note the underscores
- It's usually best to use := when defining functions
- Note the use of parentheses when writing such a complex expression
- Note the way Mathematica colours things in as you type. This can be very helpful.

★ Enhance solve Quadratic so that it returns a list containing both roots of the quadratic. Don't worry about duplicates.

★ Write a Mathematica *function* called `rotationMatrix`. It should take one parameter θ and return the matrix:

$$\begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$$

Use your function to prove the standard formula for $\sin(\theta + \phi)$. Show that Pythagoras's theorem from the fact that $\theta \mapsto \text{rotationMatrix}(\theta)$ is a homomorphism.

Moral

Functions:

- Reduce typing
- Prevent typographical errors
- Make your code much clearer
- Enable reuse of code
- Slogan: "Once and only once".

More complex functions

```
solveQuadratic[a_,b_,c_]:= Module[ {discriminant, value1, value2},
discriminant = b ^2 - 4 a c;
value1 = (-b + discriminant)/(2a);
value2 = (-b - discriminant)/(2a);
{value1, value2 }
]
```

- We have defined "local variables"
- Note the way Mathematica colours things in
- You can use Module whenever you want temporary variables, not just in function definitions.
- The functions Block and Module are almost interchangeable. Block is faster. Module is "safer".

Workbooks

- Open a new workbook
- What is the value of `z` in this workbook?
- Since this quickly becomes irritating you might want to:
    - Open the Options Inspector (`CTRL+SHIFT+ O`)
    - Change the scope from "Selection" to "Global Preferences"
    - Expand `Cell Options -> Evaluation Options`, and change the `CellContext` setting to `Notebook`

Clearing variables

- Type `ClearAll[x,y,z]` to get rid of the definitions for these variables.
- Sometimes you might want to clear everything and start again. I then
    - Select: `Evaluation->Quit Kernel->Local`.
    - Select: `Evaluation->Evaluate Notebook`.
- As you will see from the options, you can evaluate just parts of the document too.

The sensible way of solving equations

```
Solve[ x^2 + 2 x + 2 == 0, {x}]
Solve[ {2 x + 3 y == -1, 2 x - 4 y + 1 == -1}, {x,y}]
Solve[ 2 x + 3 y == -1 && 2 x - 4 y + 1 == -1, {x,y}]
Solve[ 2 x + 3 y == -1 || 2 x - 4 y + 1 == -1, {x,y}]
Reduce[ x^2 + 2 x + 2 == 0, {x}]
```

- Notice the == signs. We've now met =, := and ==.
- Notice the && this means "and"
- Notice the || this means "or"

You can easily specify the domain of the variables if you want:

```
Solve[ x^2 + 1 == 0, {x}, Reals]
```

Exercises

★ What is the general formula for the roots of a quartic equation?

★ Use the `Solve` command over the Integers to show that 2 is irrational.

★ Use the `Solve` command to find all Pythagorean triples. The main challenge is understanding the output. `Reduce` gives a more comprehensible answer.

Plots

The basic command to plot a function is `Plot`.

```
Plot[ Sin[x], {x, -10, 10}]
```

But you can tinker with the output:

```
Plot[Sin[x], {x, -10, 10}, AspectRatio -> Automatic,
PlotStyle -> {Orange, Dashed, Thick}]
```

★ Use `ContourPlot` to plot a hyperbola $x^2 - y^2 = 1$

★ Use `ParametricPlot` to plot the same hyperbola.

★ Use `ContourPlot` to show how the hyperbola $x^2 - y^2 = a$ depends on the parameter a. Use `Plot3D` to examine the surface defined by $x^2 - y^2 = a$.

★ Use `ContourPlot` to plot two touching circles. Do this in two ways: by passing two equations to contour plot, by thinking up a function whose zero set consists of two touching circles.

★ Use `ImplicitPlot3D` to plot a sphere.

★ Use `ParametericPlot3D` to plot a Möbius strip.

Interesting surfaces [Hard]

★ Plot a torus using `ContourPlot3D`

★ Plot a genus 2 surface using the approach of your choice.

★ Plot a Klein bottle using the approach of your choice. You may want to use the option `Opacity` and you may want to switch off the Mesh so that you can understand your picture. Use `PlotStyle->Opacity[0.5]` to create a translucent parametric plot and `ContourStyle->Opacity[0.5]` to create a translucent contour plot.

Morse theory

★ Find a map from the unit square to a vertical torus - i.e. a torus oriented so that you would be looking through it if you held it level with your eye. Use ParametericPlot3D to plot this torus. This defines a "height" function on the unit square. Plot the contours of this height function on the square.

★ A "critical point" of a function on the plane is a point where the gradient is zero. What are the critical points in your contour plot? How do they relate to the the 3D picture?

Morse theory continued

★ A non-degenerate critical point of f is a critical point where the 2×2 matrix of partial derivatives:

$$\frac{\partial^2 f}{\partial x_i \partial x_j}$$

does not vanish. Here x_1, x_2 are coordinates on the plane. Recall the classification of conic sections (up to linear transformation of the plane). Give without proof a classification of non-degenerate critical points up to deformation.

Revision

- [], {}, ()
- ==, =, :=
- &&, ||
- _

The Clebsch Cubic Surface

The Clebsch cubic surface is defined in projective four space as the set of points satisfying:

$$x_1^3 + x_2^3 + x_3^3 + x_4^3 + x_5^3 = 0$$

and

$$x_1 + x_2 + x_3 + x_4 + x_5 = 0$$

So it is isomorphic to the surface in projective three space given by:

$$x_1^3 + x_2^3 + x_3^3 + x_4^3 - (x_1 + x_2 + x_3 + x_4)^3 = 0$$

Let's use Mathematica to find all the lines on this surface.

Idea

- By fundamental theorem of algebra a line lies on the surface if and only if 4 points of the line lie on the surface.

- Pick two planes in 3 space. Then most lines can be parameterized simply by giving the point of intersection on each plane. This parameterization only leaves lines passing through the intersection of the planes to worry about.

- Working in inhomogeneous coordinates $(x, y, z) \mapsto [x, y, z, 1]$ we can take the planes $x = 5$ and $x = -5$. We can parameterize points in these planes as $p_1 = (5, u_1, v_1)$ and $p_2 = (-5, u_2, v_2)$. The line through these points consists of points $\lambda p_1 + (1 - \lambda)p_2$. The 4 conditions given by requiring that the points with λ equal to 0, 1, 2 and 3 give us four equations in our 4 unknowns u_1, v_1, u_2, v_2.