Introduction to Plotting in 3D

Read the notes below. Type and evaluate (shift-enter) the items listed with In[#]:= below.
You do not need to retype similar examples. Edit or copy/paste/edit past examples and evaluate.

3-Dimensional Graphing: Plot3D and Options

We will graph this function: $z = \frac{y\,e^x}{x+1}$ also known as $f(x, y) = \frac{y\,e^x}{x+1}$

Within the **Plot3D** command, list the function first (you do NOT need *z* =). The function is followed by a comma and the domain you would like to graph in a list with curly braces { }. The domain list has the variable of the function first followed by the minimum and maximum values of the domain. You need a domain for both *x* and *y* for Plot3D. (A parametric curve in 3D may only need a domain for *t*.) Note that we use "E" for *e* in our code.

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2}]
```

Use the option **BoxRatios** (instead of AspectRatio in 2D) to adjust the scale of the *x*-, *y*-, and *z*-axes. BoxRatios→ Automatic will make the scale for *x*, *y*, and *z* the same. If you put a list of numbers into BoxRatios, that sets the ratio of each axis. For example, BoxRatios→{1,1,1} or just 1 makes it a cube and BoxRatios→{1,1,2} is twice as tall as it is wide. Note with {1,1,2} the *x* and *y* are the same length, not the same scale.

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2}, BoxRatios → Automatic]
```

PlotRange will adjust the range on the *z*-axis of what is displayed for the graph.

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2}, PlotRange → {-4, 4}, BoxRatios → Automatic]
```

A more detailed list of ranges will specify what you see in each direction.

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2},
 PlotRange → {{-2, 2}, {-10, 10}, {-4, 4}}, BoxRatios → Automatic]
```

ImageSize adjusts the horizontal width of the plot in pixels. You may want to use a large size if you plan to copy/paste into another program (such as Word) for printing. Note: **Printing** in *Mathematica* does not always work very well. You can copy/paste into a new *Mathematica* notebook and be sure to use Print Preview...

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2},
 PlotRange → {-4, 4}, BoxRatios → Automatic, ImageSize → 500]
```

It is good to always **label your axes**. The first example gives basic labels. (See my 2D notes for styling.) However, in 3D in *Mathematica*, the labels float in the middle of the axes and sometimes can get in the way of what you want to show or be distracting, so I often skip this in 3D.

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2}, PlotRange → {-4, 4},
 BoxRatios → Automatic, ImageSize → 500, AxesLabel → {"x", "y", "z"}]
```

Often I remove the box (using **Boxed**) and set the axes to the corner we usually draw (using **AxesEdge**). Note that the origin is not necessarily where the axes meet. **Axes→False** will turn the axes off.

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2}, PlotRange → {-4, 4},
 BoxRatios → Automatic, ImageSize → 500, Boxed → False, AxesEdge → {-1, -1}]
```

AxesOrigin will set the cross point of the axes and draw them inside the graphics box. However, they can float awkwardly within the image and I usually prefer them outside as above.

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2}, PlotRange → {-4, 4},
 BoxRatios → Automatic, ImageSize → 500, Boxed → False, AxesOrigin → {0, 0, 0}]
```

Use **Exclusions** to remove asymptotes. Note the use of the double equal sign == for making an equation. You do not want to set *x* to equal -1. (Note: Not using AxesOrigin below.)

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2}, PlotRange → {-4, 4}, BoxRatios → Automatic,
 ImageSize → 500, Boxed → False, AxesEdge → {-1, -1}, Exclusions → {x == -1}]
```

Use **ClippingStyle** to adjust what happens where the surface is cut off by the plot range.

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2}, PlotRange → {-4, 4}, BoxRatios → Automatic,
 ImageSize → 500, Boxed → False, AxesEdge → {-1, -1}, Exclusions → {x == -1}, ClippingStyle → None]
```

To style the surface of the function, use **PlotStyle**.

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2}, PlotRange → {-4, 4},
 BoxRatios → Automatic, ImageSize → 500, Boxed → False, AxesEdge → {-1, -1},
 Exclusions → {x == -1}, ClippingStyle → None, PlotStyle → Blue]
```

There are many configurations for **Lighting** in 3D. Sometimes using “Neutral” gives a nice effect.

```
Plot3D[y E^x / (x + 1), {x, -4, 4}, {y, -2, 2}, PlotStyle → Blue, Lighting → "Neutral"]
```

Note: The order of options does not matter. Options are listed after the domain and are separated by commas.

Try including the following options examples in your code.

```
Mesh → None,
Mesh → Automatic, MeshShading → {{Blue, Red}, {Yellow, Green}}
FaceGrids → All,  FaceGrids → None
Ticks → {Range[-4, 4]}
Ticks → {Range[-4, 4, 1 / 2], Range[-4, 4, 0.5], Range[-4, 4, 1]}
Ticks → {Range[-2 Pi, 2 Pi, Pi / 2], Range[-2, 2], None}
Ticks → None
```

For more details on options for plots refer to the Documentation Center. For more with 2D plots, also see my additional notebook posted on my web site: **www.abbymath.com/Mathematica/BasicOptionsForPlot.nb**