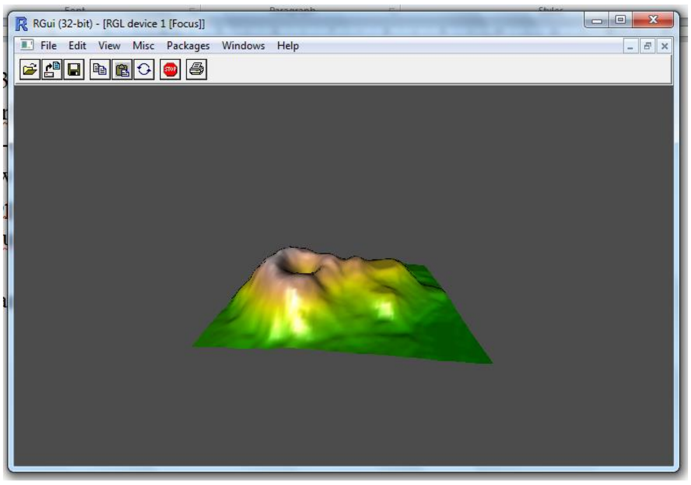
Spatial Analysis Techniques in R

Mid-week bit of fun (4)

3D plots of surface data using rgl

In some applications using geostatistical data there is a lot to be said for being able to interact with the plot so as better to visualize the distribution. Starting from a (digital elevation) matrix (DEM), or a similar object produced by interpolation from scattered data as in this week's lesson, the rgl package makes this very easy. In what follows please note that we use *y* for the height values and reserve (*x, z*) for spatial coordinates generated from the row and column numbers of the matrix. First download and install the rgl package from your CRAN Mirror, then:

```
> library(rgl)
> data(volcano) #for illustration these data come with the package
> #explore how volcano data are coded ...
> class(volcano)
[1] "matrix"
> #we need to exaggerate the relief a bit ...
> y<-2*volcano
> # and space the axes by assigning co-ordinates based on their (r, c) numbers
> x<-10*(1:nrow(y))      #10 units spacing south to north
> z<-10*(1:ncol(y))      # ditto east to west
> ylim<-range(y)
> ylim
[1] 188 390
> ylen<-ylim[2]-ylim[1]+1
> ylen
[1] 203
> colorlut<-terrain.colors(ylen)     #height color look up table
> col<-colorlut[y-ylim[1]+1]       #assign color to each height value
> #now we can plot them
> rgl.open()   #open output channel
> rgl.surface(x,z,y,color=col,back="lines") #plot it
```

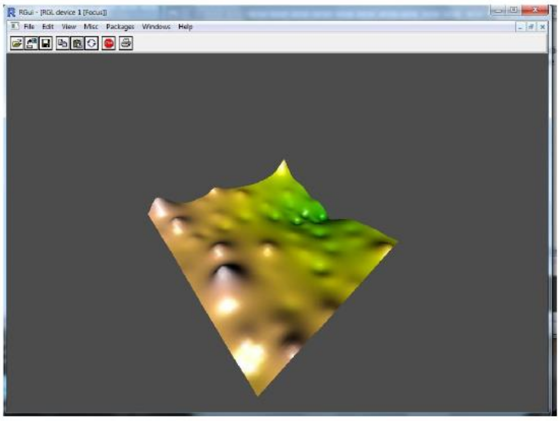


You can now move the image and rotate it using the mouse.

It’s fairly easy to input *any* interpolated values coerced into a matrix into this sequence, for example

```
>vals<-topo_idw$var1.pred   #get the heights from the object created by IDW
>valmat<-matrix(data=vales,nrow=60,ncol=62)
```

Now proceed as before to get the plot:



The above is the rgl version of topo-idw with a distnace exponent of 2 and shows the ‘ring’ contours very cleary. If you rotate the image they can of course be turned into deep pits!

David Unwin
3-12-13, rev 21-11.14, 15-12-15