Useful packages

library(rgdal) # I/O and projections
https://cran.r-project.org/web/packages/rgdal/rgdal.pdf

library(sp) # vector data
https://cran.r-project.org/web/packages/sp/sp.pdf

library(raster) # raster data
https://cran.r-project.org/web/packages/raster/vignettes/Raster.pdf

library(rgeos) # geometry operations
https://cran.r-project.org/web/packages/rgeos/rgeos.pdf

Interpolation:

library(fields) # curve and surface fitting
https://cran.r-project.org/web/packages/fields/fields.pdf

library(gstat) # variogram modelling and kriging
https://cran.r-project.org/web/packages/gstat/gstat.pdf

Plotting:

Plot3d
https://cran.r-project.org/web/packages/plot3D/plot3D.pdf
https://cran.r-project.org/web/packages/plot3D/vignettes/volcano.pdf

~~~~~~~~~~~~~~~~~~

Help:
http://www.maths.lancs.ac.uk/~rowlings/Teaching/UseR2012/cheatsheet.html
http://www.r-bloggers.com/using-r-working-with-geospatial-data/
http://neondataskills.org/R/Raster-Data-In-R/

for geospatial

Reading in and dataframes: http://www.maths.lancs.ac.uk/~rowlings/Teaching/UseR2012/cheatsheet.html

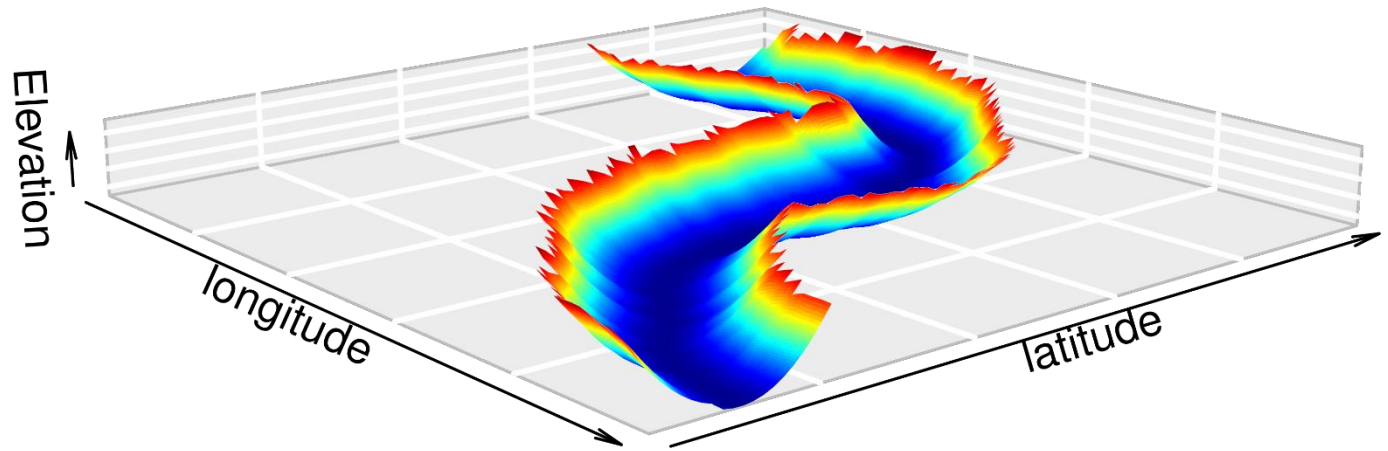
Clipping points in polygons: http://robinlovelace.net/r/2014/07/29/clipping-with-r.html

Spline

```
> library(fields)

> df=read.csv('./test_data.csv', header=FALSE, sep=",", col.names=c("x", "y", "z"))

> coord<-data.frame(x=df$x, y=df$y)
> z<-data.frame(z=df$z_fixed)

> fit<-Tps(coord,z)

> xg<-make.surface.grid(fields.x.to.grid(coord)) # makes a mesh grid of x and y
> plot(xg) # displays the mesh grid
> fhat<- predict( fit, xg) # indexes values of based on the thin plate spline to their xy index locations (i.e. this is
                                    1d and is therefore a length of the dimensions of plot(xg))

> out.p<- as.surface( xg, fhat) # creates a surface from predictions onto the xy mesh points (xg), components of
                                        which can be called as out.p$x (cols), out.p$y (rows), out.p$z (z value) etc...
> surface(out.p) # display it
```

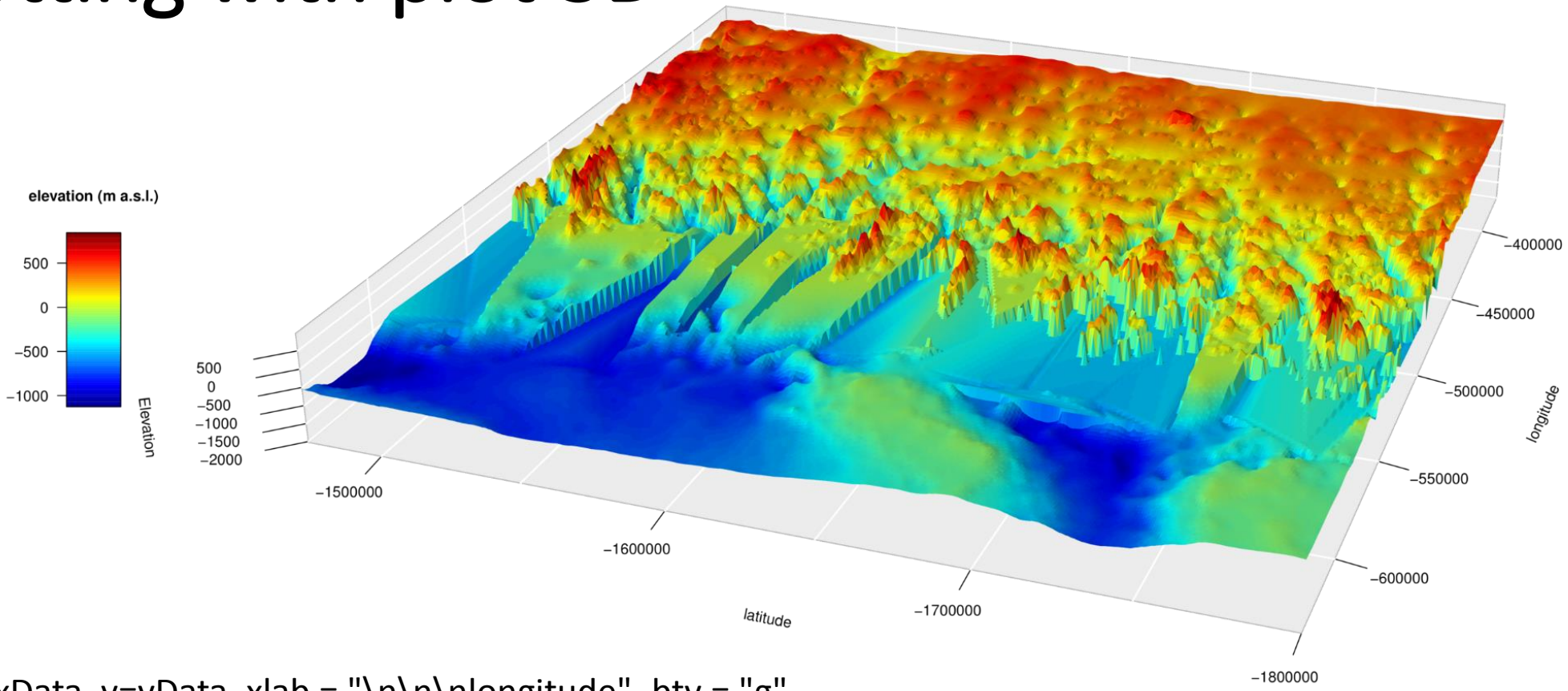
Pass in x, y and z data as matrix

3D plotting with plot 3D




```
> persp3D(z=zData, x=xData, y=yData, xlab = "\n\n\nlongitude", bty = "g",
          ylab = "\n\n\nlatitude", zlab = "Elevation", cex=10, expand = 0.14,
          d = 2, phi = 25, theta = 15, resfac = 6, shade=0.3, col = jet.col(101),
          ticktype="simple", cex.axis=1.8,  cex.title=1.8, colkey=FALSE,
          clim = c(-2000, 2000), cex.axis=1.0,  cex.title=2.0)
```

3D plotting with plot 3D




```
> persp3D(z=zData, x=xData, y=yData, xlab = "\n\n\nlongitude", bty = "g",
          ylab = "\n\n\nlatitude", zlab = "Elevation", cex=10, expand = 0.14,
          d = 2, phi = 25, theta = 15, resfac = 6, shade=0.3, col = jet.col(101),
          ticktype="simple", cex.axis=1.8,  cex.title=1.8, colkey=FALSE,
          clim = c(-2000, 2000), cex.axis=1.0,  cex.title=2.0)
```

Variogram modelling with gstat