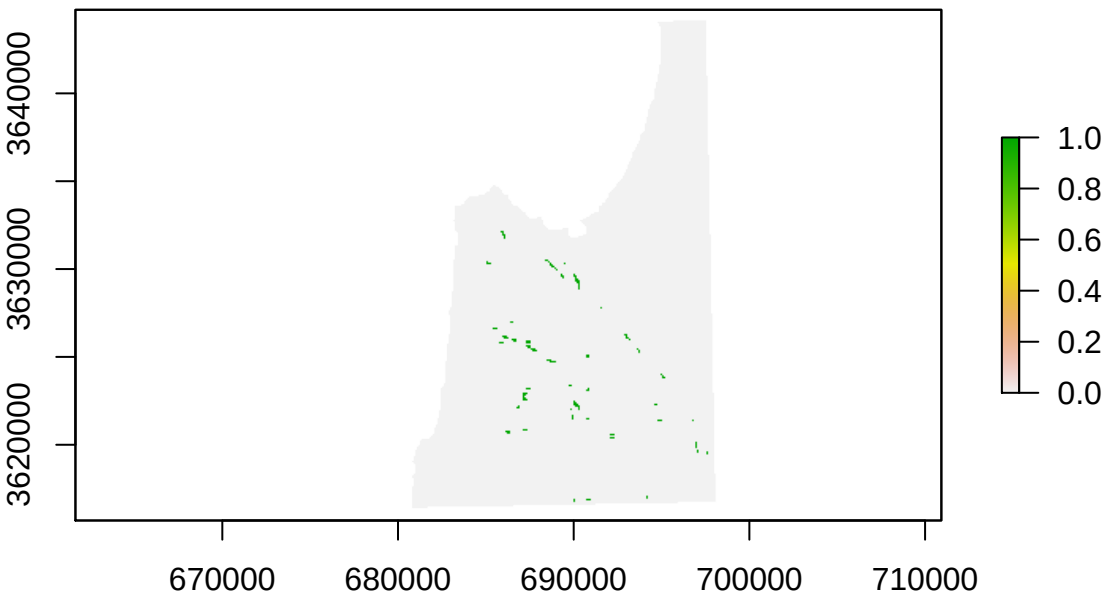
Introduction to Spatial Data Programming
Execrise 5
Geometric operations with rasters
& Working with spatio-temporal data

Last updated: 2019-02-19 15:51:13

Question 1

- Write code to **recreate** the `slope` raster which we calculated in **Lesson 09**
- Note: pay attention that you repeat exactly the same processing steps as in **Lesson 09**, such as reprojection to UTM
- Create a raster of "cliffs" based on the Haifa DEM, where a pixel is defined as a "cliff" if its 3*3 neighborhood -
  - Contains **at least one** pixel with a slope greater than 20 **and** contains **at least one** pixel with a slope less than 2
- Note: to calculate the "cliff" raster you need to apply the right focal filter with a 3*3 neighborhood. As a result, the "cliffs" raster will have a value of `1` or `TRUE` for "cliff" pixels, a value of `0` or `FALSE` for non-"cliff" pixels, and a value of `NA` where there is not enough information to make the decision
- **Plot** the resulting raster
- **Calculate** and **print** the number of pixels which were classified as "cliffs"

```r
library(sf)
library(raster)
dem1 = raster("srtm_43_06.tif")
dem2 = raster("srtm_44_06.tif")
dem = merge(dem1, dem2)
haifa = st_read("haifa.geojson", quiet = TRUE)
haifa_ext = extent(haifa) + 0.25
dem = crop(dem, haifa_ext)
r = brick("modis_south.tif")
dem = projectRaster(from = dem, crs = proj4string(r), method = "ngb", res = 90)
slope = terrain(dem, "slope", "degrees")
m = matrix(1, nrow = 3, ncol = 3)
f = function(x) any(x > 20) & any(x < 2)
cliff = focal(slope, w = m, fun = f)
plot(cliff)
```




```
length(cliff[cliff == 1])
```

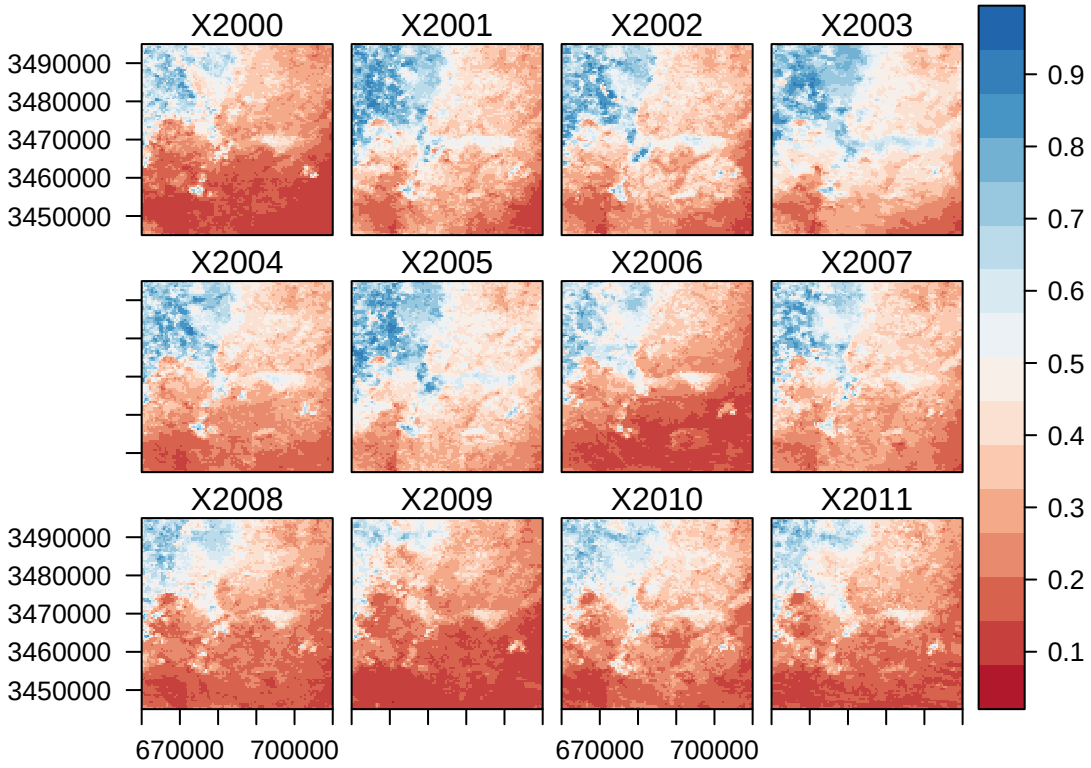
```
## [1] 176
```

(50 points)

Question 2

- **Read** the `modis_south.tif` raster to a multi-band raster object
- **Calculate** a 12-band raster that will contain the maximal NDVI per pixel (excluding `NA`) for each year during 2000-2011. The first band will contain the maximal values per pixel from all images of 2000, the second band will contain the maximal values per pixel in 2001, and so on.
- Note: use the `dates.csv` table to determine the layers corresponding to each year
- **Plot** the resulting raster using the `levelplot` function from package `rasterVis`

```
library(rasterVis)
r = brick("modis_south.tif")
dates = read.csv("dates.csv", stringsAsFactors = FALSE)
years = 2000:2011
a = as.array(r)
template = r[[1]]
year_max = stack()
f = function(x) {max(x, na.rm = TRUE)}
for(i in years) {
  template[] = apply(a[, , dates$year == i], 1:2, f)
  year_max = stack(year_max, template)
}
names(year_max) = years
levelplot(year_max, par.settings = RdBuTheme, contour = FALSE)
```


X2000
X2001
X2002
X2003
X2004
X2005
X2006
X2007
X2008
X2009
X2010
X2011
3490000
3480000
3470000
3460000
3450000
670000
700000
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1

(50 points)