

ANALYSIS OF POPULARITY FROM ASCII MAPS

This script calculates the mean popularity of paths around the studied sites.

The results are printed in the console rather than saved to a file.

```
library(raster)
library(ggplot2)
library(reshape)
library(dplyr)
library(rgeos)
library(rgdal)
```

Clears the environment to avoid errors from old files

```
rm(list=ls())
```

Browse to the main DEM

```
DEM <- raster(file.choose(), sep=",")
```

Browse to the sites' shapefile (for which to get popularity values) **Make sure it is in the same projection as the DEM

```
shp <- readOGR(file.choose())
```

Browse to the raster map to analyse

```
map <- raster(file.choose(), sep=",")
```

Browse to the map to use as a buffer for possible paths ("Exploration" for site-pair paths, "DEM" for FETE paths)

```
buffer <- raster(file.choose(), sep=",")
```

```
buffer[buffer > 0] <- 0
```

```
buffer[buffer < 0] <- NA
```

Merge map and buffer so that everything in map that is NA and has a value in buffer is set to 0

```
map2 <- resample(map, buffer, "bilinear")
```

```
full <- merge(map2, buffer) # For all NA in "map2," it takes the values of "buffer"
```

Transform the raster into a matrix of coordinates and values

```
whole <- rasterToPoints(full)
```

Iterate over the different site coordinates to obtain the mean popularity values of the cells in a 1km buffer

```
for(i in 1:nrow(shp)) {
```

```
  # Transform the site into a raster point
```

```
  site.ras <- rasterize(shp[i,], map, field = 1)
```

```
  # Create a polygon of 1km radius around the site raster
```

```
  pol <- rasterToPolygons(site.ras)
```

```
  pol.buf <- buffer(pol, width=1000)
```

```
  # Use the buffer polygon to clip the Popularity map uploaded
```

```
  mask.ras <- mask(map, pol.buf)
```

```
  site.list <- rasterToPoints(mask.ras)
```

```
  # If the site's buffer falls onto at least 2 cells walked on by an agent,
```

```
# this calculates the mean popularity of the cells within the buffer and uses a T-test to evaluate if these are
significantly more popular
# than all the cells walked on.

if (nrow(site.list) > 1){
  print(shp[i,]$str_3) # The value here represents the name of the site, but can be changed for other shapefiles.
  print(t.test(site.list[,3],whole[,3], alternative = "greater"))

}else{
  # If the site buffer is outside of the paths, nothing happens.
  print(s)
  print("NA")
}
}
```