

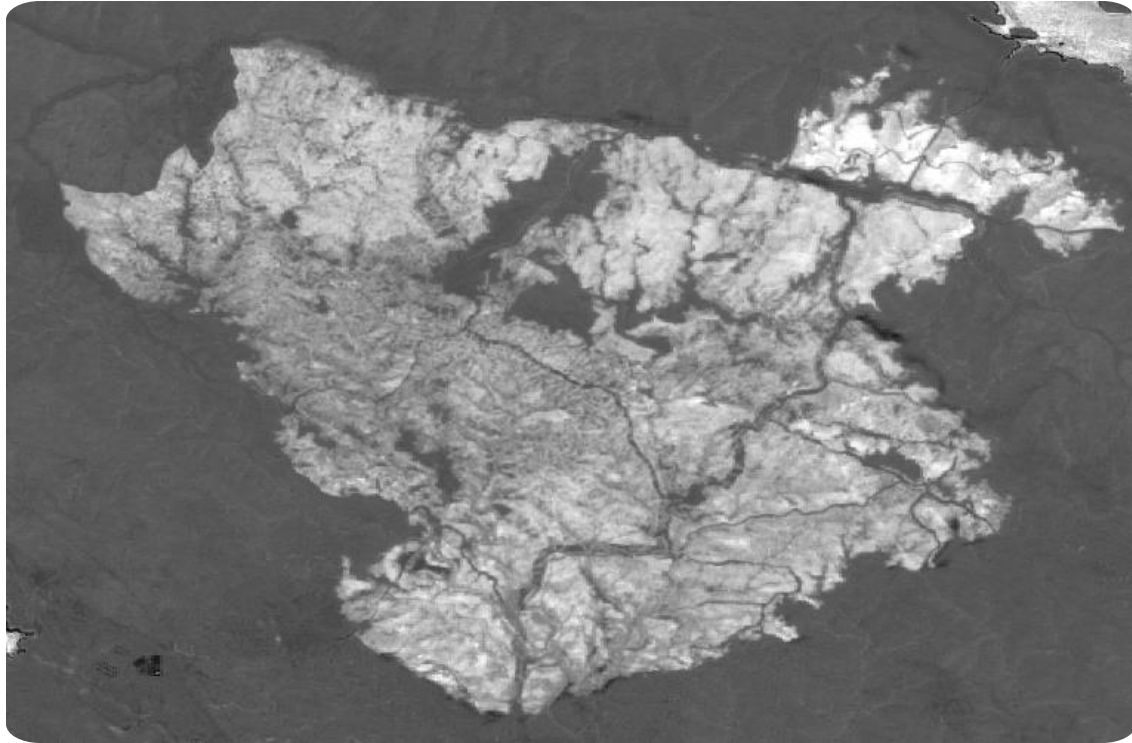


O manexo de imaxes termográficas con R: identificando o perímetro dun incendio forestal

CENTRO DE INVESTIGACIÓN Y TECNOLOGÍA
MATEMÁTICA DE GALICIA
GALICIAN CENTRE FOR MATHEMATICAL
RESEARCH AND TECHNOLOGY

Marta Rodríguez Barreiro
María José Ginzo Villamayor

marta.rodriguez.barreiro@usc.es



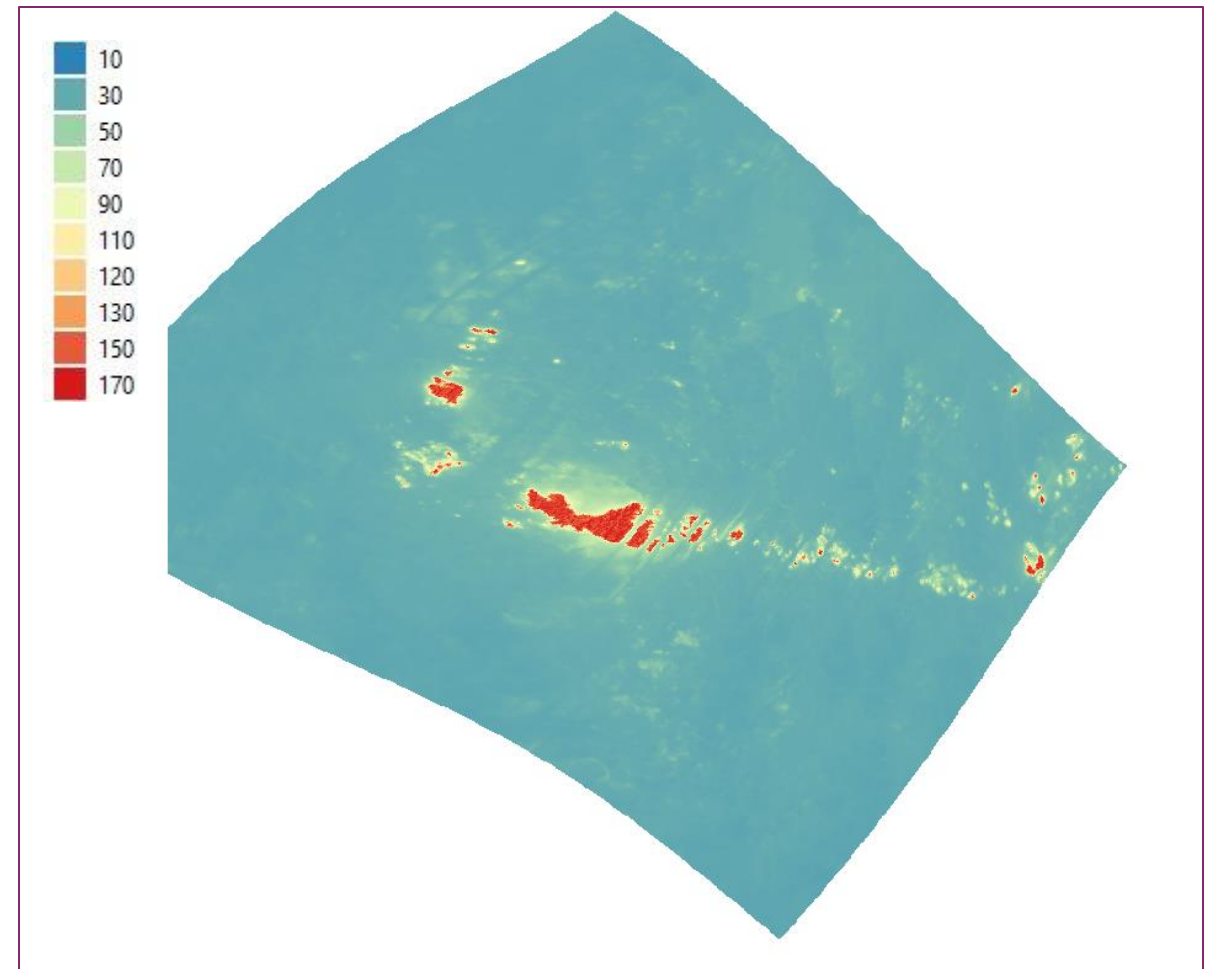
Coñecer o perímetro dun incendio forestal en tempo real é de gran interese para a xestión das tarefas de extinción.

Na actualidade, as imaxes satelitais permiten ver a ubicación da fronte do incendio e a súa dirección e velocidade de avance, pero a frecuencia de actualización destas imaxes aínda é moi baixa.

As cámaras térmicas ubicadas en drons ou aeronaves de extinción, permiten tomar imaxes para monitorizar o avance do incendio en tempo real.

Unha imaxe térmica ou termográfica, é unha imaxe que captura a radiación térmica que non é visible para o ollo humano.

A tecnoloxía termográfica permite rexistrar e representar a distribución de temperaturas en superficies e obxectos.





No noso caso as imaxes térmicas obtéñense con cámaras termográficas instaladas nos helicópteros de extinción.

Tamén hai cámaras instaladas en drons, pero aínda non dispoñemos desas imaxes.

O procesamento das imaxes térmicas presenta algunhas dificultades. Os valores da temperatura da imaxe poden verse alterados, o que impide establecer umbrais de temperatura para determinar. En ocasións, atopámonos incluso con valores negativos.



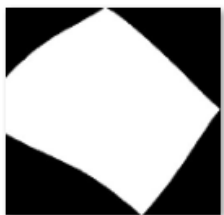
img-2024060511
4534123-ter_modified.tif



img-2024060511
4536125-ter_modified.tif



img-2024060512
0034597-ter_modified.tif



img-2024060512
0139064-ter_modified.tif



img-2024060512
2640325-ter_modified.tif



img-2024060512
2642327-ter_modified.tif



img-2024060512
2644329-ter_modified.tif



img-2024060512
2646332-ter_modified.tif

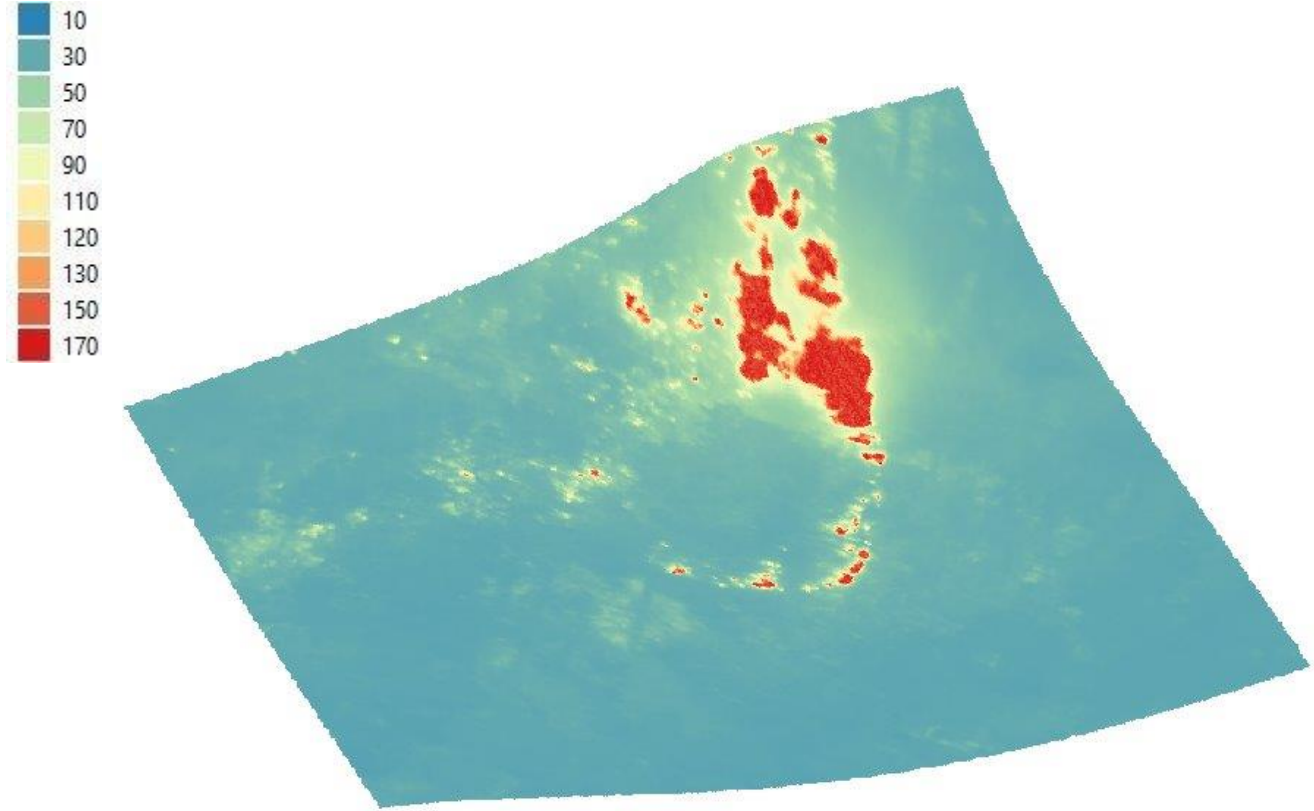


img-2024060512
2648333-ter_modified.tif

Para o desenvolvemento deste algoritmo dispoñemos dunha serie de imaxes térmicas, en formato tif, tomadas por un helicóptero durante as tarefas de extinción do incendio. Só un helicóptero dispón da cámara.

As imaxes están xeorreferenciadas, é dicir, están orientadas cara o norte e cada píxel correspóndese cunhas coordenadas. Deste xeito, se representamos a imaxe nun mapa, vemos exactamente onde está situada.

Ademais, cada imaxe ten asociada unha data e unha hora, na que foi tomada.



Polo tipo de cámara e a altura á que voa o helicóptero, as imaxes tomadas son imaxes parciais do incendio, nas que atopamos unha zona que se corresponde co incendio e unha gran parte do terreo

Un dos grandes retos deste algoritmo foi establecer qué temperaturas corresponden coa presenza de incendio, xa que debido ás incidencias nos valores de temperatura, e que non todos os incendios acadan as mesmas temperaturas, non podemos establecer un valor fixo.

O primeiro paso do algoritmo consiste en ler os nomes de todas as imaxes que se atopan na ruta indicada polo usuario. Para isto usamos o comando `list.files` coa ruta indicada polo usuario.

```
> list.files(path=path, full.names=TRUE)
```

Esto permite obter un vector co nome de todas as imaxes que temos que analizar.

A continuación, analizamos as imaxes unha a unha iterando no vector do nome das imaxes. Non podemos unir as imaxes e tratalas todas xuntas porque, polas incidencias da cámara, pode que unha ou máis imaxes teñan algún erro no rango de valores.

Para cada imaxe creamos un obxecto raster da librería `terra` para ler os valores de dita imaxe.

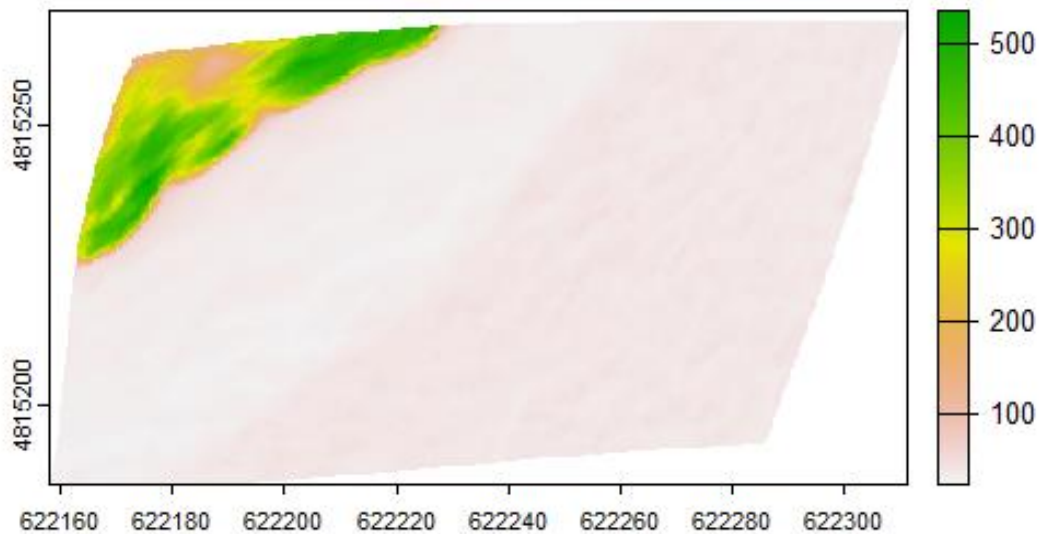
```
> imagen <- rast(im)
```

```
> imagen
```

```
class       : SpatRaster
dimensions  : 266, 487, 1  (nrow, ncol, nlyr)
resolution  : 0.3145875, 0.3145875  (x, y)
extent      : 622158, 622311.2, 4815186, 4815270  (xmin, xmax, ymin, ymax)
coord. ref. : WGS 84 / UTM zone 29N (EPSG:32629)
source      : img-20220826130852303-ter_modified.tif
name        : img-20220826130852303-ter_modified
```

Coa función `plot` podemos representar a imaxe raster.

```
> plot(imagen)
```



Tamén podemos ler os valores de cada píxel coa función `values`.

```
> val <- values(imagen)
```

```
> summary(val)
```

```
img-20220826130852303-ter_modified
```

```
Min.      : 23.89
```

```
1st Qu.: 32.04
```

```
Median : 35.83
```

```
Mean     : 64.00
```

```
3rd Qu.: 38.32
```

```
Max.     :534.32
```

```
NA's     :24027
```


A idea é obter unha temperatura que nos sirva como umbral para discernir que píxeles pertencen ó incendio e cales non.

Escóllese como umbral a temperatura media do segundo grupo creado co k-means.

```
> t_incendio <- sort(grupos$centers)[2]
```

```
> grupos$centers
```

```
      [,1]  
1  35.30975  
2 254.76677  
3 417.55425
```

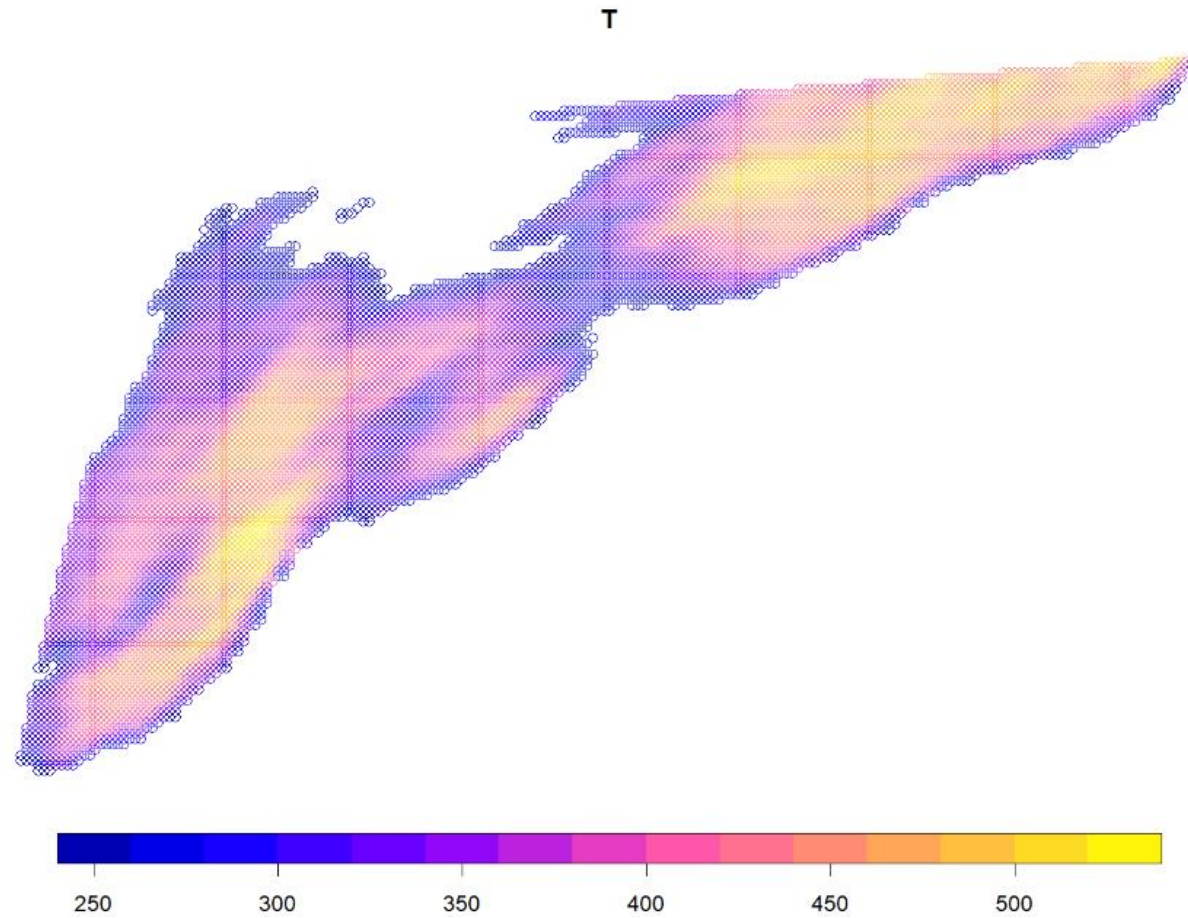
Creamos un obxecto espacial, coa librería sf, cos valores do dataframe superiores ó umbral de temperatura.

```
> p <- sf::st_as_sf(d, coords = c("x","y"))
```

```
> p  
Simple feature collection with 7710 features and 1 field  
Geometry type: POINT  
Dimension:      XY  
Bounding box:   xmin: 622162.8 ymin: 4815226 xmax: 622227.6 ymax: 4815268  
CRS:            NA
```

Coa función `plot` podemos representar o obxecto espacial.

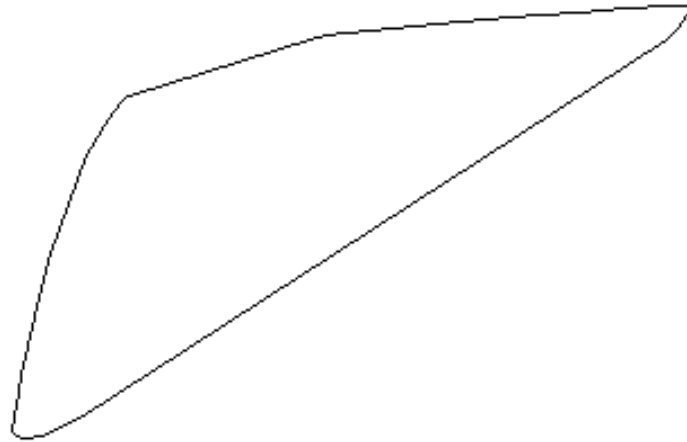
```
> plot(p)
```



A continuación calculamos a envolvente convexa de todos os puntos do obxecto espacial, que son os que consideramos parte do incendio.

Para isto, empregamos as funcións `st_convex_hull` e `st_union` do paquete `sf`.

```
> per <- st_convex_hull(st_union(p))  
> plot(per)
```

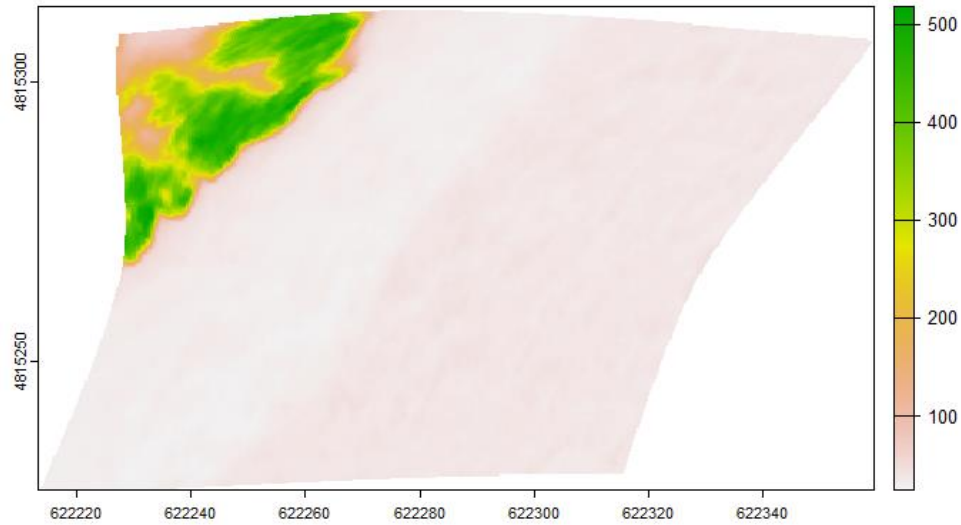


Ademais, de cara a facer un mosaico con todas as imaxes de entrada e representar todo o incendio, gardamos os valores do marco da imaxe, empregando a función `ext` da librería `terra`. Gardamos tamén o número de columnas e filas, empregando `ncol` e `nrow` respectivamente,

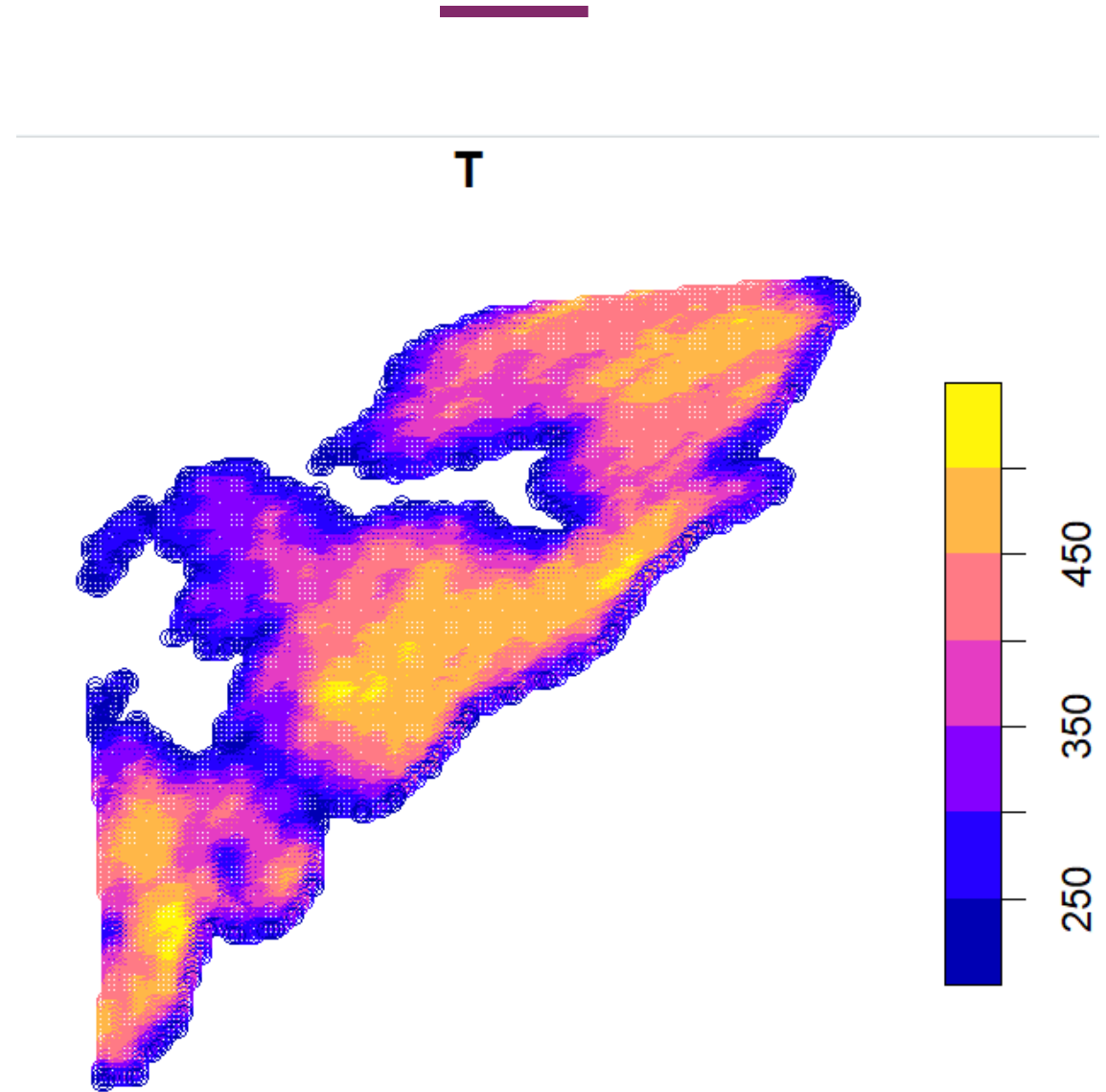
```
> xmin <- ext(imagen)$xmin  
> xmax <- ext(imagen)$xmax  
> ymin <- ext(imagen)$ymin  
> ymax <- ext(imagen)$ymax  
> nrow <- nrow(imagen)  
> ncol <- ncol(imagen)
```

```
> ext(imagen)  
SpatExtent : 622157.955987931, 622311.160112661, 4815186.269  
42874, 4815269.94971042 (xmin, xmax, ymin, ymax)
```

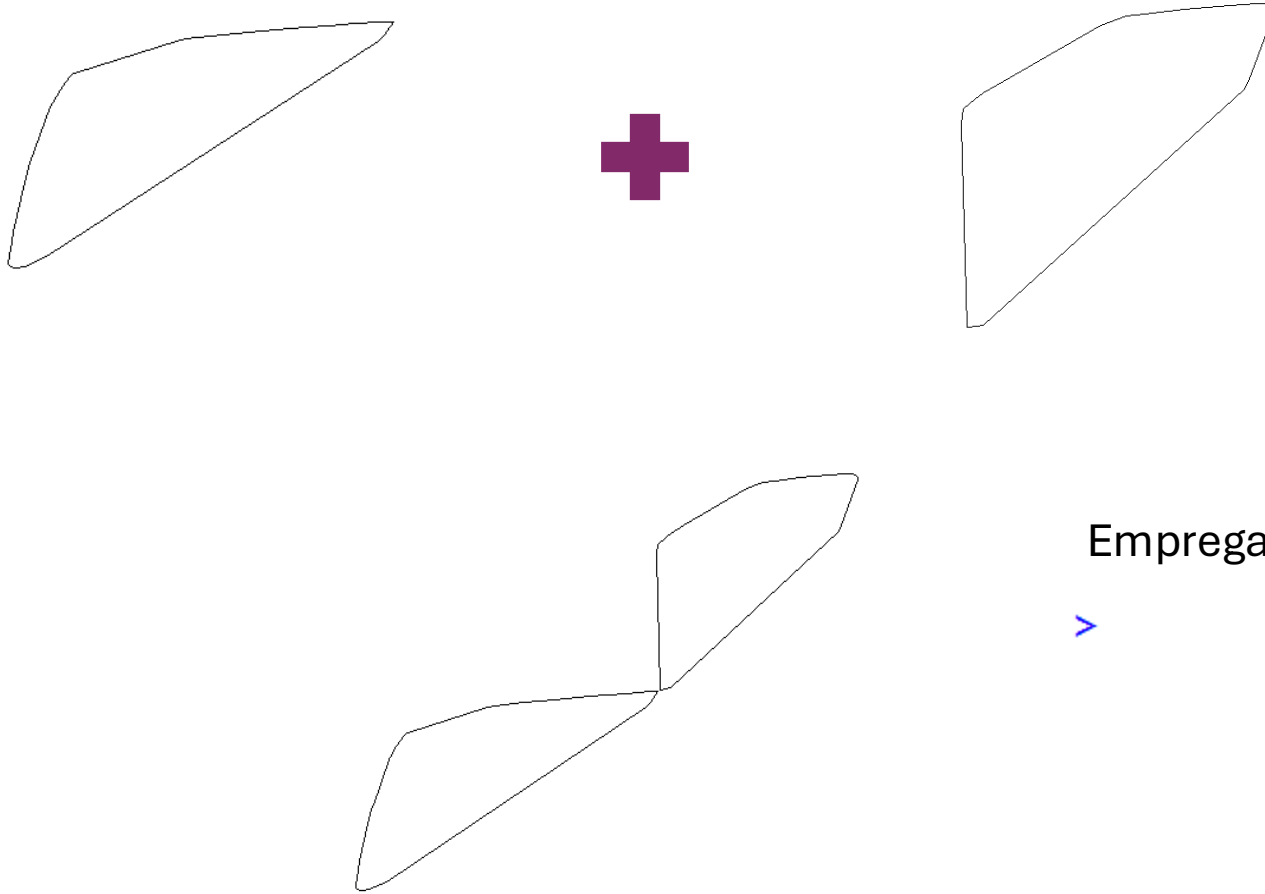
```
> mosaico <- imagen
```



Unha vez terminado este proceso coa primeira imaxe, teríamos un trozo de perímetro do incendio e tamén un mosaico que consiste nunha única imaxe. O bucle segue a iterar, e repite o procedemento coa seguinte imaxe da lista.



Actualizamos o perímetro do incendio unindo o perímetro que xa tiñamos co que acabamos de calcular.



Empregamos a función `st_union` do paquete `sf`.

```
> perimetro <- st_union(perimetro, per)
```

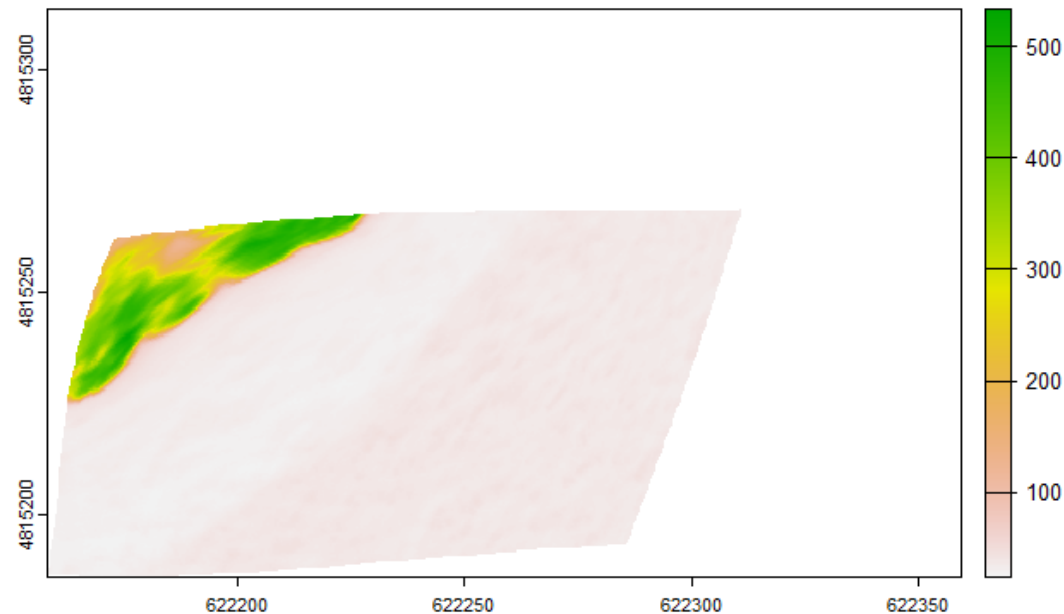
Actualizamos os valores das coordenadas máximas e mínimas, e o número de filas e columnas.

```
> if (ext(imagen)$xmin < xmin){
+   xmin <- ext(imagen)$xmin
+ }
> if (ncol(imagen) > ncol){
+   ncol <- ncol(imagen)
+ }
```

Para facer o mosaico, creamos un ráster baleiro que teña a extensión dos valores que fomos gardando. Empregamos a función `rast` do paquete `terra`.

Logo, adaptamos o mosaico gardado á nova extensión, coa función `resample` do paquete `terra`.

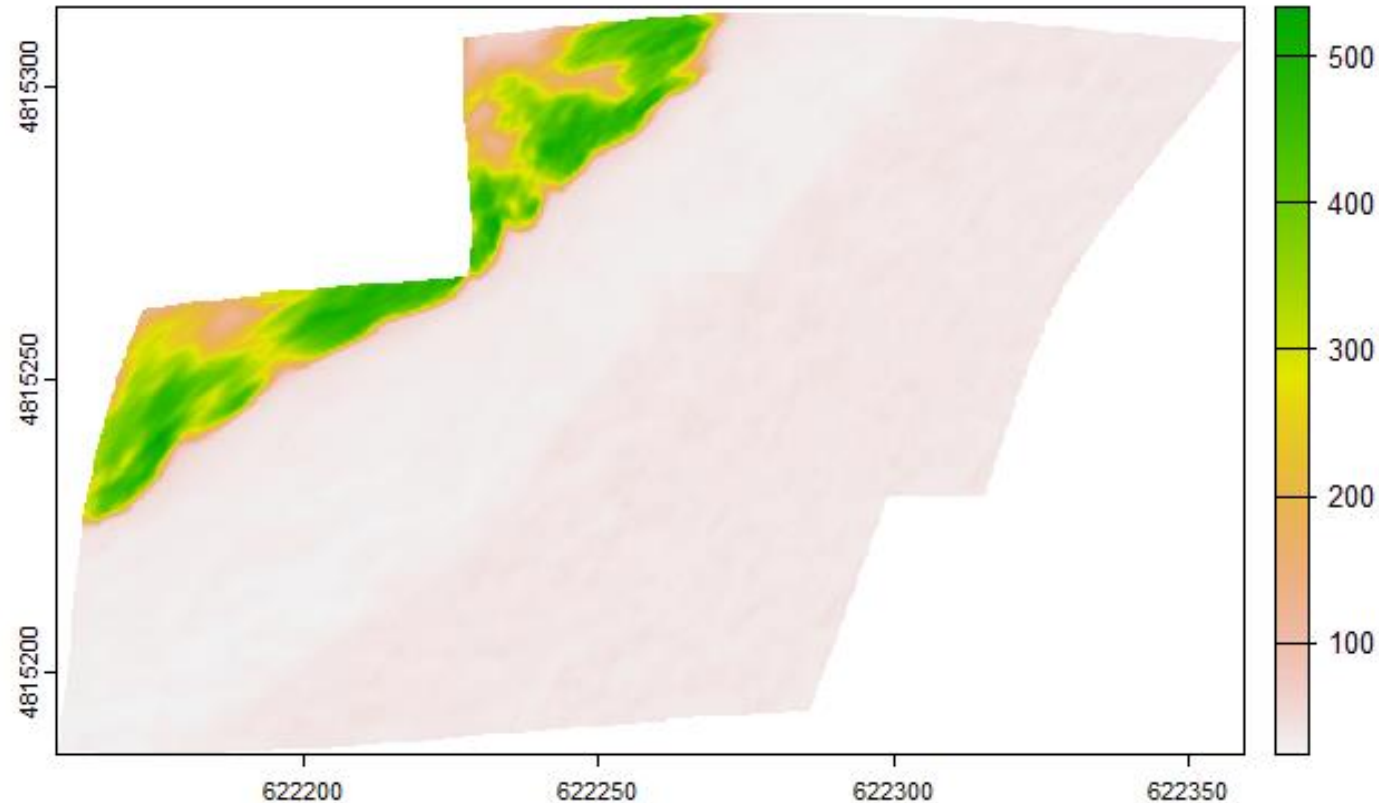
```
> r1 <- rast(nrow=nrow, ncol=ncol, ext=c(xmin, xmax, ymin, ymax), crs=crs(imagen))
> mosaico_old <- resample(mosaico, r1)
```



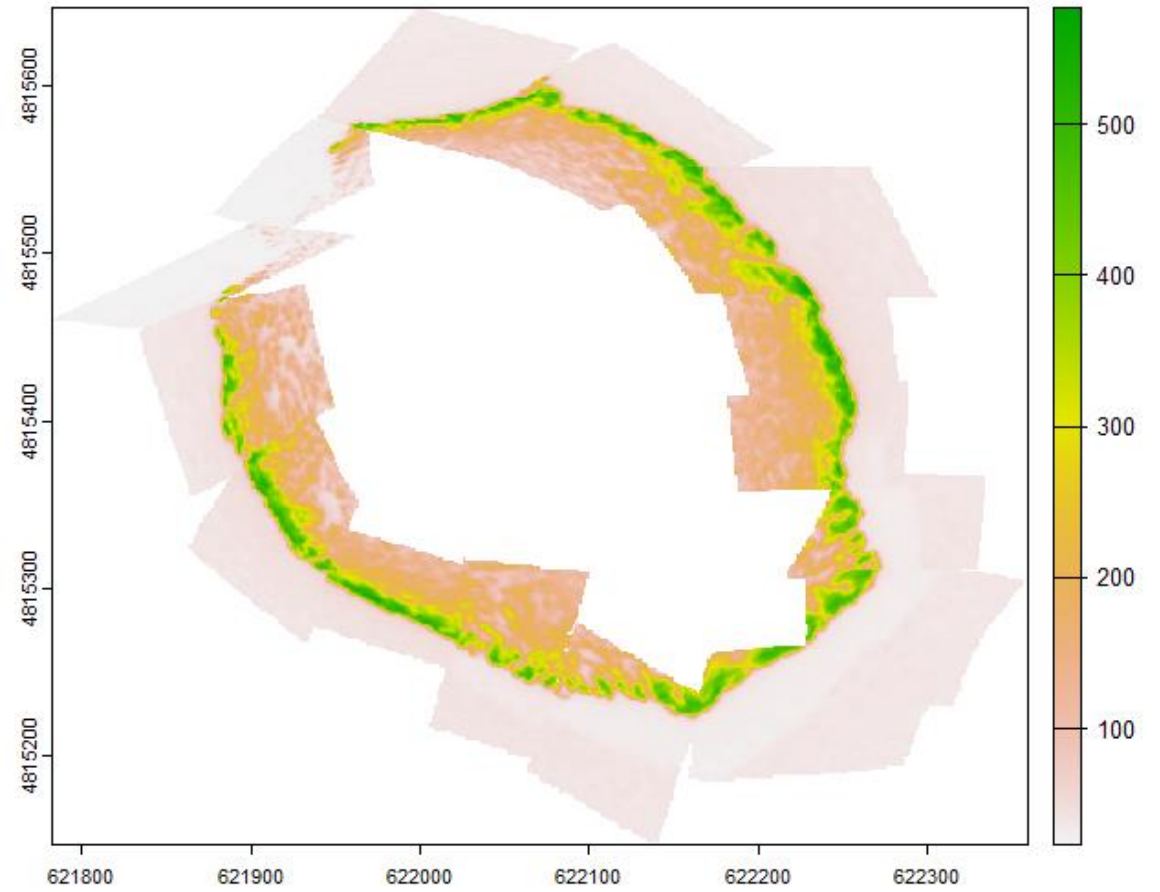
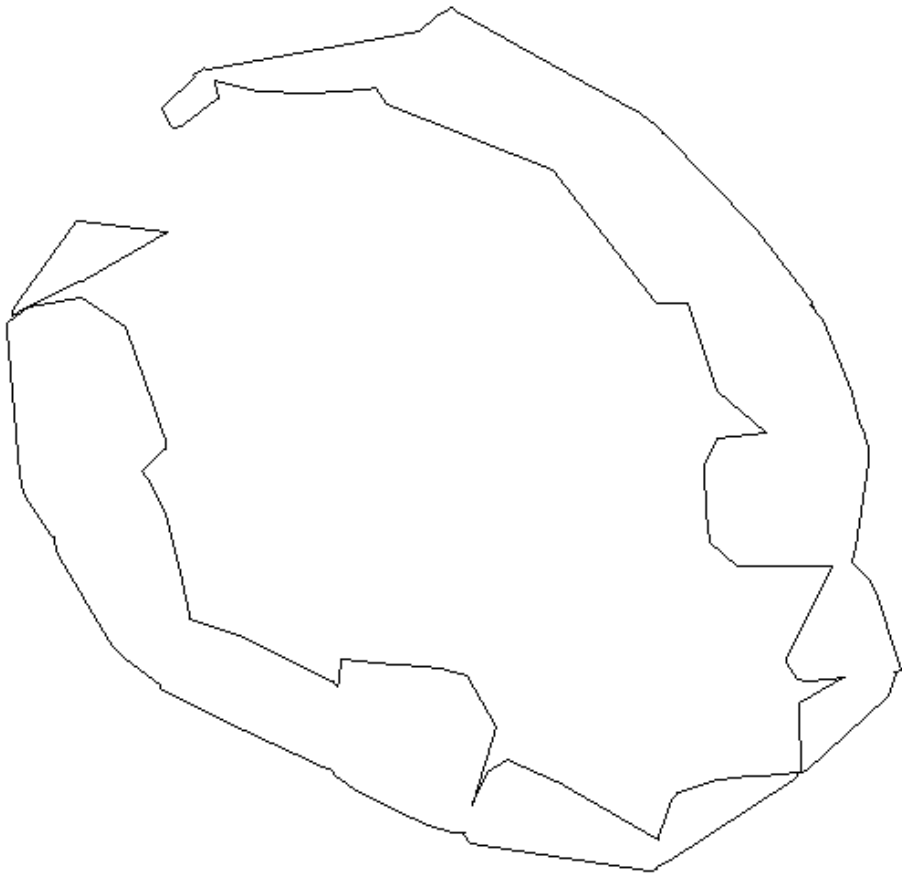
Unha vez que temos as dúas imaxes na mesma extensión, empregamos a función `mosaic` do paquete `terra` para unilas.

Como parámetro, podemos escoller qué valor queremos manter en caso de que as imaxes que se van unir se solapen nalgún píxel. Neste caso escollemos o valor de temperatura máximo.

```
> mosaico <- mosaic(mosaico_old, imagen_old, fun=max)
```

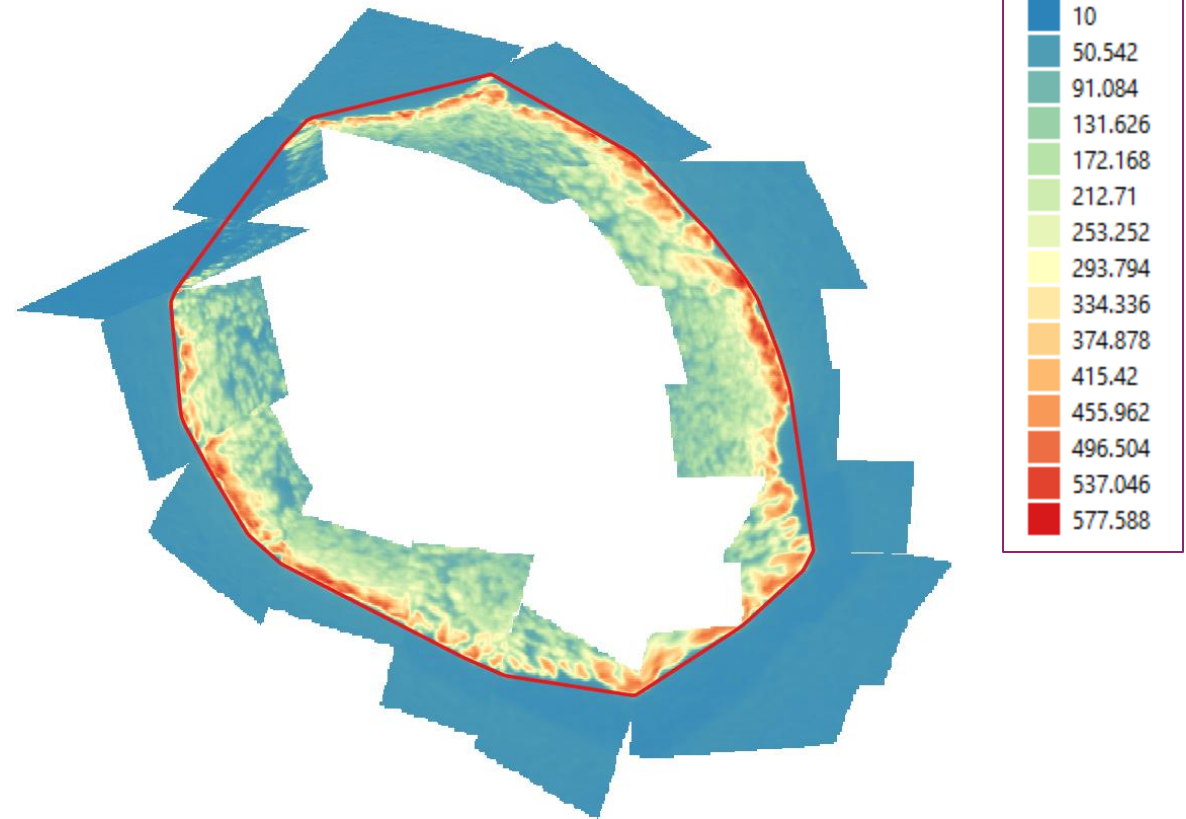
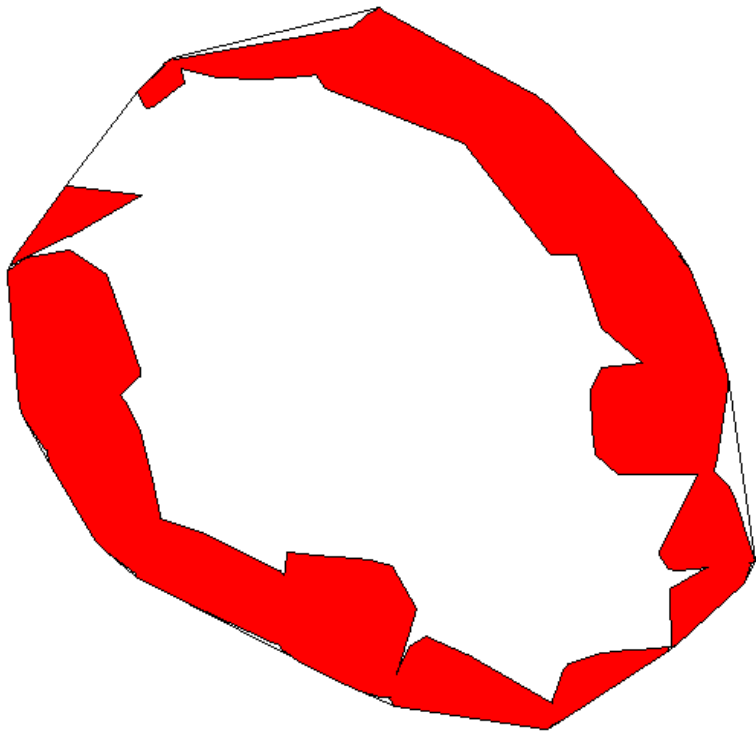


Repetimos esto con todas as imaxes e obtemos o perímetro e mosaicos correspondentes.



Calculamos a envoltura convexa do perímetro obtido, para evitar ter un perímetro discontinuo en caso de que falte algunha imaxe.

```
> peri <-st_convex_hull(perimetro)
```





O manexo de imaxes termográficas con R: identificando o perímetro dun incendio forestal

CENTRO DE INVESTIGACIÓN Y TECNOLOGÍA
MATEMÁTICA DE GALICIA
GALICIAN CENTRE FOR MATHEMATICAL
RESEARCH AND TECHNOLOGY

Marta Rodríguez Barreiro
María José Ginzo Villamayor

marta.rodriguez.barreiro@usc.es