Laboratorio 1: Curso Tópicos de Estadística Geoespacial

Katherine Barquero

10/9/2020

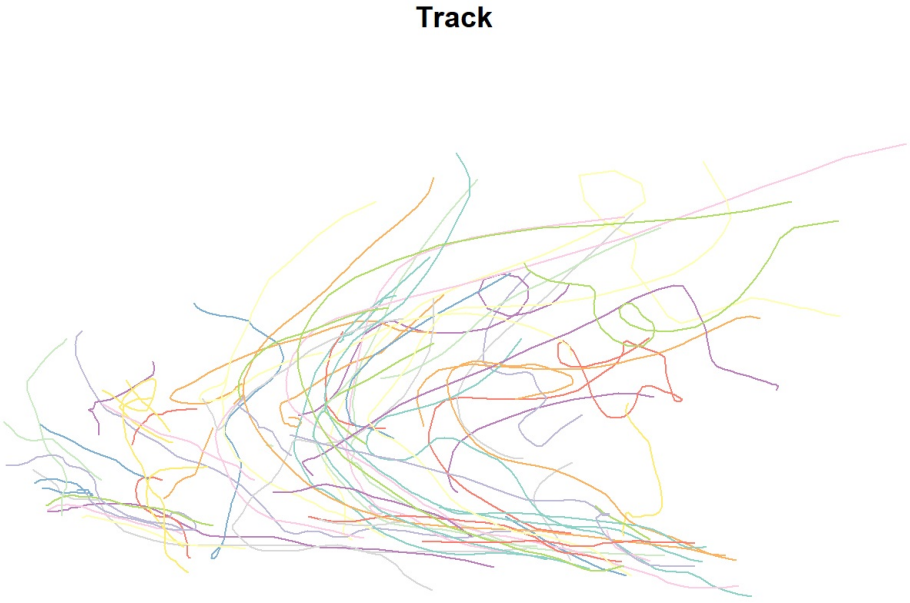
Ejercicios 1.3

1. Read the shapefile storms_xyz_feature from the shape directory in the sf package

```
dataset <- system.file("shape", package="sf") %>%
           st_read("storms_xyz_feature") %>%
           st_zm()
```

```
## Reading layer `storms_xyz_feature' from data source `C:\Users\Katherine\Documents\R\win-library\4.0\sf\shape' u
sing driver `ESRI Shapefile'
## Simple feature collection with 71 features and 1 field
## geometry type:  LINESTRING
## dimension:      XYZ
## bbox:           xmin: -102.2 ymin: 8.3 xmax: 0 ymax: 59.5
## z_range:        zmin: 924 zmax: 1017
## CRS:            NA
```

```
plot(dataset)
```



2. Copy this file to another directory on your computer, and read it from there (note: a shapefile consists of more than one file!)

```
#st_write(dataset, "storms_xyz_feature/storms_xyz_feature.shp")
#
```

3. How many features does this dataset contain?

El set de datos cuenta con 71 features

```
dataset$geometry
```

```
## Geometry set for 71 features
## geometry type:  LINESTRING
## dimension:      XY
## bbox:           xmin: -102.2 ymin: 8.3 xmax: 0 ymax: 59.5
## CRS:            NA
## First 5 geometries:
```

```
## LINESTRING (-50.8 20.1, -51.2 20.4, -51.5 20.8,...
```

```
## LINESTRING (-77.4 14.3, -77.8 13.9, -78.2 13.5,...
```

```
## LINESTRING (-62.7 14.7, -63.2 15, -63.7 15.3, -...
```

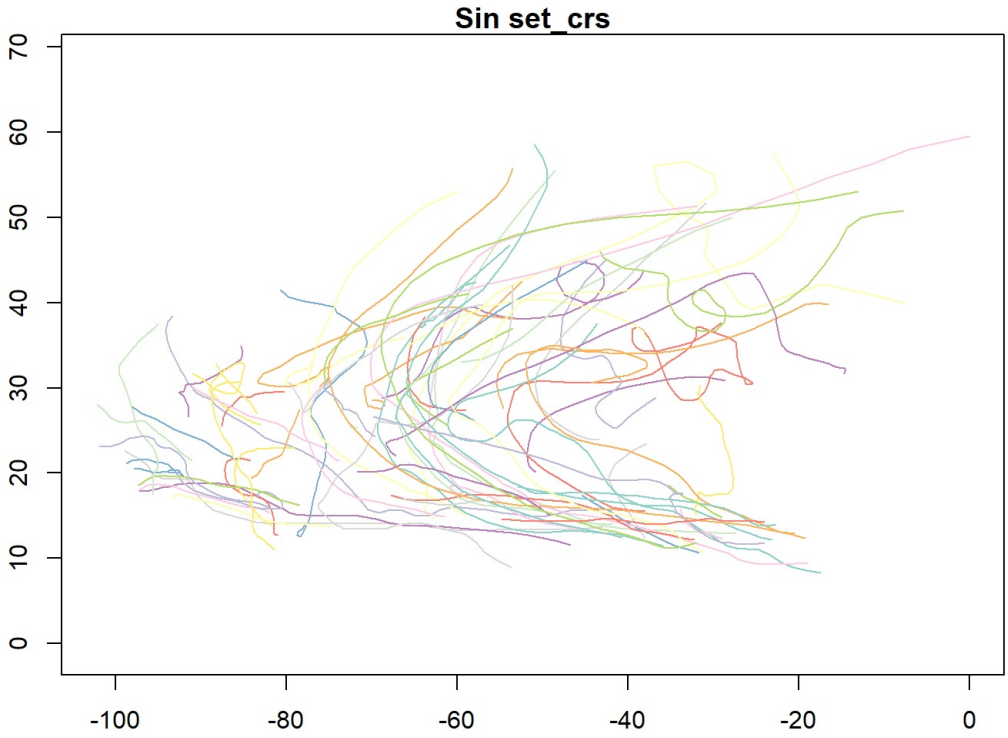
```
## LINESTRING (-72.5 25.5, -72.4 25.4, -72.2 25.4,...
```

```
## LINESTRING (-38 12.4, -38.7 13.6, -39.6 15, -40...
```

```
# Hay 71 features
```

4. Plot the dataset, with axes = TRUE (hint: before plotting, pipe through st_zm to drop Z and M coordinates; more about this in chapter 3).

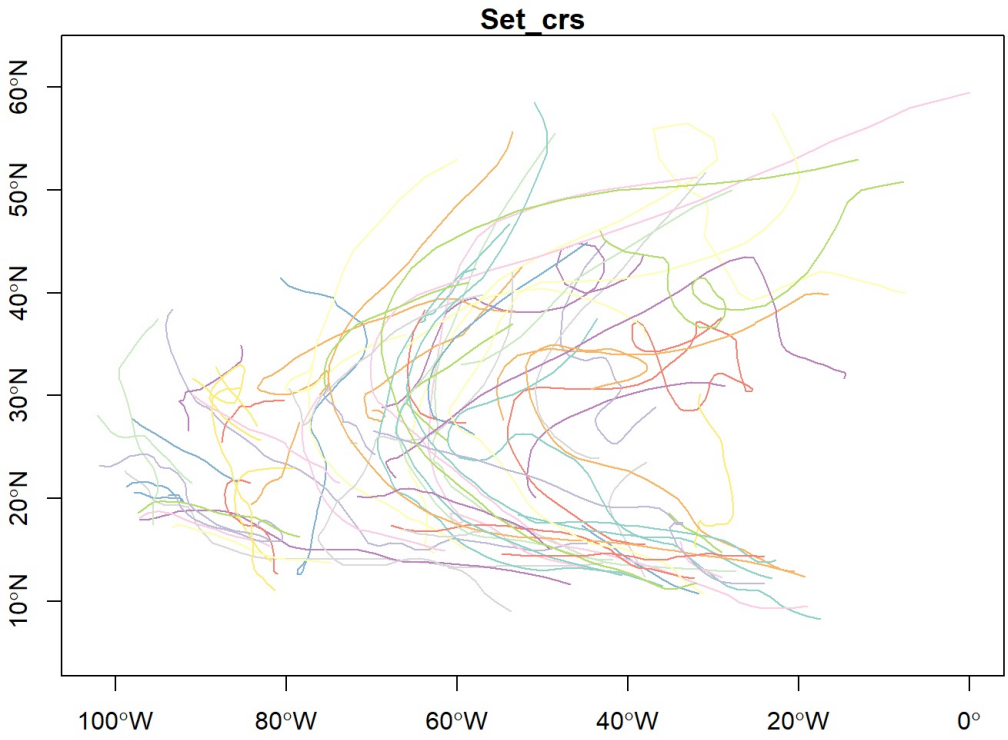
```
par(mfrow=c(2,2))
plot(dataset, axes = T, main ="Sin set_crs")
```



5. Before plotting, pipe the dataset through st_set_crs(4326). What is different in the plot obtained?

La diferencia se presenta en los ejes, el primero en coordenadas cartesianas y el segundo en polares

```
a= dataset %>% st_set_crs(4326)
plot(a, axes = T, main= "Set_crs")
```



Ejercicios 2.6

1. Convert the (x,y) points (10,2), (-10,-2),(10, -2) and (0,10) to polar coordinates

```
cart2pol <- function(x, y)
{
  r <- sqrt(x^2 + y^2)
  t <- atan(y/x)


  c(r,t)
}

X1 = 10
Y1 = 2

X2 = -10
Y2 = -2

X3 = 10
Y3 = -2

X4 = 0
Y4 = 10

XY1<-cart2pol(X1, Y1)
XY2<-cart2pol(X2, Y2)
XY3<-cart2pol(X3, Y3)
XY4<-cart2pol(X4, Y4)

CoordPol<-rbind(XY1, XY2, XY3, XY4)
CoordPol
```

```
##         [,1]       [,2]
## XY1 10.19804  0.1973956
## XY2 10.19804  0.1973956
## XY3 10.19804 -0.1973956
## XY4 10.00000  1.5707963
```

2. Convert the the polar points (10, 45°), (0, 100°) and (5, 359°) to Cartesian coordinates

Para pasar de puntos polares a coordenadas cartesianas se realizan las siguientes operaciones:

- x = r * COS (_0)
- Y = r x sin ()

```
polcart <- function(r, t)
{
  x <- r*cos(t)
  y <- r*sin(t)

  c(x,y)
}


X1 = 10
Y1 = 45

X2 = 0
Y2 = 100

X3 = 5
Y3 = 39



XY1<-polcart(X1, Y1)
XY2<-polcart(X2, Y2)
XY3<-polcart(X3, Y3)


CoordCart<-rbind(XY1, XY2, XY3)
CoordCart
```

```
##         [,1]     [,2]
## XY1 5.253220 8.509035
## XY2 0.000000 0.000000
## XY3 1.333215 4.818977
```

3. Assuming the Earth is a sphere with a radius of 6371 km,compute for (lambda, cita) points the great circle distance between (10,10) and (11,10), between (10,80) and (11,80), between (10,10) and (10,81) (units:degree). What are the distance units?

Primero los puntos se pasan a coordenas polares y para eso se usa cart2pol, utilizada en el punto 1 de este ejercicio

```
Coordpol <- rbind(cart2pol(10,10),
                  cart2pol(11,10),
                  cart2pol(10,80),
                  cart2pol(11,80),
                  cart2pol(10,10),
                  cart2pol(10,11),
                  cart2pol(10,80),
                  cart2pol(10,81)
                  )

Coordpol <- data.frame(Coordpol)
Coordpol
```

```
##         X1        X2
## 1 14.14214 0.7853982
## 2 14.86607 0.7378151
## 3 80.62258 1.4464413
## 4 80.75271 1.4341532
## 5 14.14214 0.7853982
## 6 14.86607 0.8329813
## 7 80.62258 1.4464413
## 8 81.61495 1.4479611
```

Se procede a calcular las distancias

```
d1<-acos(sin(Coordpol[1,2])*sin(Coordpol[2,2]) + cos(Coordpol[1,2])*cos(Coordpol[2,2])*cos(abs(Coordpol[1,1]-Coordpol[2,1])))

d2<-acos(sin(Coordpol[3,2])*sin(Coordpol[4,2]) + cos(Coordpol[3,2])*cos(Coordpol[4,2])*cos(abs(Coordpol[3,1]-Coordpol[4,1])))
d2
```

```
## [1] 0.020898
```

```
d3<-acos(sin(Coordpol[5,2])*sin(Coordpol[6,2]) + cos(Coordpol[5,2])*cos(Coordpol[6,2])*cos(abs(Coordpol[5,1]-Coord
pol[6,1])))
d3
```

```
## [1] 0.4958222
```

```
d4<-acos(sin(Coordpol[7,2])*sin(Coordpol[8,2]) + cos(Coordpol[7,2])*cos(Coordpol[8,2])*cos(abs(Coordpol[7,1]-Coord
pol[8,1])))
d4
```

```
## [1] 0.1174568
```

```
distancias<-cbind(d1, d2, d3, d4)
distancias
```

```
##             d1       d2        d3        d4
## [1,] 0.5203428 0.020898 0.4958222 0.1174568
```

Ejercicios 4.4

1. NDVI

(NIR-R)/(NIR+R) NIR the near infrared and R the red band

Read the L7_ETMs.tif file into object x

```
tif = system.file("tif/L7_ETMs.tif", package = "stars")
  x = read_stars(tif)

length(x)
```

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

```
#class(x[[1]])
dim(x[[1]]) # 6 bandas
```

```
##    x    y band
##  349  352    6
```

```
#A qué se refieren los valores de la matriz
st_dimensions(x)
```

```
##      from  to  offset delta                       refsys point values
## x       1 349  288776  28.5 UTM Zone 25, Southern Hem... FALSE   NULL [x]
## y       1 352 9120761 -28.5 UTM Zone 25, Southern Hem... FALSE   NULL [y]
## band    1   6      NA    NA                           NA    NA   NULL
```

```
st_bbox(x) #EXTENSIÓN ESPACIAL DE LA MATRIZ
```

```
##      xmin      ymin      xmax      ymax
##  288776.3 9110728.8  298722.8 9120760.8
```

Distribute the band dimensions over attributes by split(x, “band”)

```
a<-split(x, "band")
a
```

```
## stars object with 2 dimensions and 6 attributes
## attribute(s):
##       X1              X2              X3              X4
##  Min.   : 47.00   Min.   : 32.00   Min.   : 21.00   Min.   :  9.00
##  1st Qu.: 67.00   1st Qu.: 55.00   1st Qu.: 49.00   1st Qu.: 52.00
##  Median : 78.00   Median : 66.00   Median : 63.00   Median : 63.00
##  Mean   : 79.15   Mean   : 67.57   Mean   : 64.36   Mean   : 59.24
##  3rd Qu.: 89.00   3rd Qu.: 79.00   3rd Qu.: 77.00   3rd Qu.: 75.00
##  Max.   :255.00   Max.   :255.00   Max.   :255.00   Max.   :255.00
##       X5              X6
##  Min.   :  1.00   Min.   :  1.00
##  1st Qu.: 63.00   1st Qu.: 32.00
##  Median : 89.00   Median : 60.00
##  Mean   : 83.18   Mean   : 59.98
##  3rd Qu.:112.00   3rd Qu.: 88.00
##  Max.   :255.00   Max.   :255.00
## dimension(s):
##   from  to  offset delta                       refsys point values
## x    1 349  288776  28.5 UTM Zone 25, Southern Hem... FALSE   NULL [x]
## y    1 352 9120761 -28.5 UTM Zone 25, Southern Hem... FALSE   NULL [y]
```

Then, compute NDVI by using an expression that uses the NIR (band 4) and R (band 3) attributes directly.

```
NIR<-a$X4
R<-a$X3

NDVI<-(NIR-R)/(NIR+R)
head(NDVI,5)
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## [1,] 0.2640000 0.1538462 0.2066116 0.4000000 0.3846154 0.4380165 0.4000000
## [2,] 0.2096774 0.1840000 0.3170732 0.4000000 0.3225806 0.3223140 0.3333333
## [3,] 0.1891892 0.1960784 0.2941176 0.3384615 0.2824427 0.3281250 0.3650794
## [4,] 0.3069307 0.2235294 0.2421053 0.2421053 0.2500000 0.3277311 0.4153846
## [5,] 0.2666667 0.3518519 0.3535354 0.2954545 0.2619048 0.2929293 0.3606557
##           [,8]      [,9]     [,10]     [,11]     [,12]     [,13]      [,14]
## [1,] 0.2695652 0.2586207 0.2525253 0.2873563 0.3142857 0.1857143 0.06666667
## [2,] 0.2892562 0.1746032 0.2521739 0.2558140 0.3170732 0.2743363 0.18881119
## [3,] 0.3125000 0.1230769 0.3000000 0.3404255 0.2978723 0.2470588 0.22123894
## [4,] 0.4074074 0.3525180 0.2272727 0.3389831 0.3000000 0.3008130 0.20370370
## [5,] 0.3629630 0.3428571 0.4015748 0.3934426 0.2477064 0.2195122 0.25190840
##          [,15]        [,16]       [,17]      [,18]     [,19]     [,20]
## [1,] 0.0000000 -0.040650407  0.07017544 0.20792079 0.4042553 0.4336283
## [2,] 0.1351351  0.006802721 -0.02255639 0.16831683 0.3978495 0.4159292
## [3,] 0.2352941  0.052631579 -0.03355705 0.14285714 0.2125984 0.3786408
## [4,] 0.2173913  0.232142857  0.02158273 0.00000000 0.2238806 0.3225806
## [5,] 0.3084112  0.165217391  0.00000000 0.02189781 0.1520000 0.1718750
##           [,21]       [,22]       [,23]       [,24]     [,25]      [,26]
## [1,] 0.40000000  0.34693878  0.36363636  0.37931034 0.3944954 0.23364486
## [2,] 0.36000000  0.22580645  0.30645161  0.41463415 0.2459016 0.05263158
## [3,] 0.36000000  0.19008264  0.01351351  0.17142857 0.3684211 0.20000000
## [4,] 0.22935780 -0.01587302 -0.19753086 -0.06756757 0.2640000 0.16279070
## [5,] 0.09230769 -0.10967742 -0.28421053 -0.13580247 0.1176471 0.15328467
##           [,27]     [,28]     [,29]     [,30]       [,31]        [,32]
## [1,] 0.28735632 0.4579439 0.5114504 0.3382353 -0.02439024 -0.093023256
## [2,] 0.22000000 0.4716981 0.4710744 0.3565217  0.06870229  0.020689655
## [3,] 0.02097902 0.1652174 0.4867257 0.3228346  0.04347826 -0.126050420
## [4,] 0.15873016 0.3157895 0.4782609 0.2195122  0.03333333  0.000000000
## [5,] 0.12592593 0.1969697 0.2713178 0.1578947  0.06153846 -0.008130081
##            [,33]     [,34]     [,35]     [,36]     [,37]     [,38]     [,39]
## [1,] 0.225806452 0.2758621 0.3153153 0.4495413 0.4876033 0.4210526 0.3442623
## [2,] 0.007518797 0.3070866 0.4482759 0.5000000 0.4954955 0.4754098 0.2824427
## [3,] 0.107692308 0.3846154 0.4915254 0.4285714 0.4821429 0.4240000 0.2535211
## [4,] 0.107692308 0.3740458 0.3781513 0.3162393 0.4017094 0.2519084 0.1125000
## [5,] 0.107692308 0.2125984 0.1962617 0.2869565 0.4126984 0.2500000 0.2380952
##          [,40]     [,41]     [,42]     [,43]     [,44]     [,45]     [,46]
## [1,] 0.2960000 0.3888889 0.4200000 0.4766355 0.4608696 0.3043478 0.2477064
## [2,] 0.2686567 0.3548387 0.4432990 0.4678899 0.4918033 0.4700855 0.4414414
## [3,] 0.2297297 0.3719008 0.4038462 0.4285714 0.4876033 0.4912281 0.4509804
## [4,] 0.1190476 0.4385965 0.4909091 0.4867257 0.4957983 0.4406780 0.3888889
## [5,] 0.3000000 0.4444444 0.4339623 0.4782609 0.5121951 0.4529915 0.4150943
##          [,47]       [,48]      [,49]      [,50]       [,51]      [,52]
## [1,] 0.3333333  0.41071429  0.4375000  0.2689655 -0.03846154  0.1571429
## [2,] 0.4205607  0.33333333  0.1448276 -0.1052632 -0.29523810 -0.2138728
## [3,] 0.4257426  0.08527132 -0.2432432 -0.3240741 -0.35622318 -0.3423423
## [4,] 0.3861386 -0.23636364 -0.3567839 -0.3532338 -0.32038835 -0.3390558
## [5,] 0.1333333 -0.11409396 -0.1927711 -0.2613636 -0.30000000 -0.2500000
```

```
##            [,53]       [,54]        [,55]       [,56]       [,57]       [,58]
## [1,]  0.27407407  0.37096774 -0.095541401 -0.08974359 -0.11111111 -0.13043478
## [2,] -0.14942529 -0.11377246 -0.209876543 -0.21212121 -0.15328467 -0.16546763
## [3,] -0.32291667 -0.26256983 -0.040650407 -0.11564626 -0.18987342 -0.13725490
## [4,] -0.24705882 -0.15862069 -0.008130081 -0.13333333 -0.17365269 -0.03030303
## [5,] -0.05185185  0.05263158 -0.141025641 -0.08219178 -0.05797101 -0.01515152
##           [,59]       [,60]       [,61]     [,62]     [,63]      [,64]
## [1,] -0.1044776 -0.09923664 -0.10606061 0.1525424 0.1639344 0.08196721
## [2,] -0.2409639 -0.15923567  0.03937008 0.2100840 0.3513514 0.25862069
## [3,] -0.2163743 -0.20454545 -0.05263158 0.2136752 0.3027523 0.37614679
## [4,]  0.0720000  0.08450704  0.12676056 0.3500000 0.3565217 0.43362832
## [5,]  0.2115385  0.39622642  0.39344262 0.3983740 0.4067797 0.45614035
##            [,65]       [,66]       [,67]       [,68]       [,69]     [,70]
## [1,] -0.02290076 -0.08974359  0.14925373  0.27586207  0.24074074 0.2653061
## [2,]  0.14049587 -0.08965517  0.12676056  0.19696970  0.08029197 0.1743119
## [3,]  0.29914530 -0.01960784 -0.08888889  0.02631579 -0.02040816 0.1219512
## [4,]  0.29090909 -0.05555556 -0.21428571 -0.18471338 -0.02040816 0.1764706
## [5,]  0.27433628  0.01538462 -0.16312057 -0.13939394  0.05109489 0.2758621
##          [,71]     [,72]     [,73]       [,74]       [,75]       [,76]
## [1,] 0.3982301 0.4634146 0.3445378 0.081967213 -0.10526316 -0.22274882
## [2,] 0.2952381 0.3333333 0.2758621 0.008403361 -0.16312057 -0.09756098
## [3,] 0.2727273 0.2820513 0.2307692 0.130434783 -0.01639344  0.07317073
## [4,] 0.3050847 0.3684211 0.2456140 0.221238938  0.23636364  0.15789474
## [5,] 0.3166667 0.3965517 0.3211009 0.285714286  0.27927928  0.26315789
##           [,77]     [,78]     [,79]     [,80]     [,81]     [,82]     [,83]
## [1,] 0.01840491 0.1408451 0.3906250 0.3613445 0.3333333 0.3396226 0.3703704
## [2,] 0.21538462 0.2137931 0.4366197 0.3281250 0.3162393 0.3518519 0.3962264
## [3,] 0.19379845 0.3146853 0.4701987 0.2700730 0.3448276 0.3750000 0.4150943
## [4,] 0.31200000 0.3636364 0.3793103 0.2631579 0.3559322 0.3451327 0.3333333
## [5,] 0.31782946 0.2173913 0.3237410 0.1450382 0.2295082 0.2586207 0.3217391
##          [,84]      [,85]      [,86]      [,87]     [,88]     [,89]      [,90]
## [1,] 0.2307692 0.12500000 0.15151515 0.17857143 0.2920354 0.3846154 0.32786885
## [2,] 0.1743119 0.04854369 0.01149425 0.03448276 0.2421053 0.4513274 0.41269841
## [3,] 0.3207547 0.25454545 0.21904762 0.17525773 0.2340426 0.3939394 0.46218487
## [4,] 0.3217391 0.33333333 0.39823009 0.38738739 0.3333333 0.3191489 0.30526316
## [5,] 0.3333333 0.29411765 0.32173913 0.26495726 0.2372881 0.2407407 0.05050505
##            [,91]     [,92]     [,93]     [,94]     [,95]     [,96]     [,97]
## [1,]  0.32773109 0.2931034 0.3559322 0.4426230 0.3870968 0.3643411 0.4100719
## [2,]  0.42187500 0.3770492 0.4590164 0.5079365 0.4761905 0.4754098 0.4929577
## [3,]  0.33333333 0.4426230 0.4827586 0.5084746 0.5040000 0.5312500 0.5251799
## [4,] -0.05154639 0.2000000 0.3877551 0.4117647 0.4789916 0.5182482 0.4896552
## [5,]  0.10000000 0.1743119 0.3684211 0.3684211 0.3846154 0.4716981 0.5151515
##          [,98]        [,99]      [,100]     [,101]     [,102]      [,103]
## [1,] 0.2689655 -0.006802721 -0.06329114 0.02702703 0.21739130  0.36956522
## [2,] 0.2751678  0.140939597  0.14666667 0.10810811 0.11278195  0.21428571
## [3,] 0.1621622  0.231884058  0.47586207 0.27741935 0.01910828 -0.01282051
## [4,] 0.1506849  0.272727273  0.54074074 0.40372671 0.22222222  0.02380952
## [5,] 0.4054054  0.238095238  0.30578512 0.44262295 0.45833333  0.48344371
##           [,104]    [,105]    [,106]     [,107]     [,108]     [,109]
## [1,]  0.29670330 0.2527473 0.2421053 0.25454545 0.31782946 0.14482759
## [2,]  0.27835052 0.2527473 0.2631579 0.18644068 0.20967742 0.14705882
## [3,]  0.08396947 0.2307692 0.1940299 0.10638298 0.04000000 0.04918033
## [4,] -0.08496732 0.1056911 0.1081081 0.02631579 0.01538462 0.01538462
## [5,]  0.35099338 0.1586207 0.4393939 0.33333333 0.10967742 0.18421053
##           [,110]      [,111]     [,112]      [,113]     [,114]       [,115]
## [1,]  0.28070175  0.29166667 0.37500000  0.42307692 0.09677419 -0.152000000
## [2,]  0.23966942  0.38297872 0.44827586  0.40186916 0.22580645 -0.008130081
## [3,]  0.10924370  0.35483871 0.35632184  0.31111111 0.26785714  0.203389831
## [4,] -0.03937008 -0.03225806 0.24271845  0.20833333 0.34090909  0.353535354
## [5,] -0.03750000 -0.08270677 0.04201681 -0.04545455 0.15789474  0.359223301
##           [,116]      [,117]      [,118]       [,119]      [,120]     [,121]
## [1,] -0.08928571 -0.11304348 -0.20930233 -0.190082645  0.05357143 0.08256881
## [2,]  0.04424779  0.12500000  0.01123596  0.049504950  0.15789474 0.00000000
## [3,]  0.17241379  0.23076923  0.34831461  0.303370787  0.14942529 0.10588235
## [4,]  0.36842105  0.22834646  0.12500000  0.210084034  0.02439024 0.09473684
## [5,]  0.24137931  0.02189781 -0.04964539 -0.007407407 -0.07575758 0.05660377
##           [,122]     [,123]       [,124]      [,125]       [,126]     [,127]
## [1,] 0.041322314 0.07692308  0.154471545  0.19298246  0.165048544 0.05882353
## [2,] 0.009009009 0.04424779  0.000000000  0.05982906  0.068965517 0.12068966
## [3,] 0.020000000 0.02000000 -0.009009009 -0.06557377  0.008695652 0.04132231
## [4,] 0.152941176 0.11111111  0.029702970  0.02654867 -0.024390244 0.08620690
## [5,] 0.142857143 0.07368421  0.030303030  0.09433962  0.061946903 0.03875969
##           [,128]      [,129]     [,130]     [,131]     [,132]      [,133]
## [1,] -0.09923664 -0.01470588 0.21600000 0.40540541 0.44642857  0.45762712
## [2,]  0.00000000 -0.04918033 0.08333333 0.24786325 0.35537190  0.50000000
## [3,]  0.08928571  0.01694915 0.00000000 0.05882353 0.07317073  0.11111111
## [4,]  0.07407407  0.04424779 0.04132231 0.06451613 0.04065041 -0.01587302
## [5,]  0.13513514  0.09090909 0.03225806 0.06153846 0.11666667  0.02608696
##           [,134]      [,135]     [,136]    [,137]     [,138]       [,139]
```

```
##  [1,]  0.38842975  0.26618705 0.32846715 0.3676471 0.40000000  0.441176471
##  [2,]  0.33333333  0.27819549 0.33834586 0.3383459 0.33823529  0.254901961
##  [3,]  0.37777778  0.10294118 0.14925373 0.2089552 0.13432836 -0.006896552
##  [4,]  0.04411765 -0.08270677 0.02362205 0.1200000 0.00000000  0.051724138
##  [5,] -0.11627907 -0.01612903 0.12820513 0.1735537 0.08474576  0.171171171
##           [,140]    [,141]     [,142]     [,143]     [,144]    [,145]    [,146]
##  [1,] 0.42372881 0.4545455 0.46551724 0.38983051 0.37984496 0.4615385 0.2676056
##  [2,] 0.30769231 0.3392857 0.27433628 0.31623932 0.32758621 0.3613445 0.4796748
##  [3,] 0.09401709 0.1881188 0.02803738 0.04347826 0.14035088 0.2761905 0.3333333
##  [4,] 0.10476190 0.1020408 0.00000000 0.01010101 0.06796117 0.1881188 0.2087912
##  [5,] 0.13131313 0.0200000 0.02702703 0.11111111 0.08256881 0.1111111 0.2244898
##            [,147]      [,148]     [,149]      [,150]      [,151]      [,152]
##  [1,] -0.06586826 -0.14136126 -0.1250000  0.08280255  0.22137405  0.22137405
##  [2,]  0.26086957 -0.10059172 -0.1717172 -0.10734463  0.00000000  0.14503817
##  [3,]  0.16393443 -0.03225806 -0.1055901 -0.03759398 -0.01639344  0.06666667
##  [4,]  0.06666667  0.00000000 -0.1259259  0.00000000 -0.04615385 -0.08527132
##  [5,]  0.12621359 -0.02521008 -0.1311475 -0.11278195 -0.10791367 -0.07575758
##            [,153]      [,154]      [,155]     [,156]     [,157]     [,158]
##  [1,] -0.03571429 -0.16393443 -0.08620690 -0.1428571 -0.1406250 -0.1475410
##  [2,] -0.04918033 -0.17741935 -0.20833333 -0.1908397 -0.1612903 -0.1228070
##  [3,]  0.04000000 -0.02654867 -0.17557252 -0.2195122 -0.1900826 -0.1846154
##  [4,]  0.09243697  0.07017544 -0.14285714 -0.1826087 -0.2500000 -0.2883436
##  [5,]  0.15447154  0.10769231 -0.03333333  0.0000000 -0.2222222 -0.2875000
##           [,159]     [,160]      [,161]     [,162]      [,163]       [,164]
##  [1,] -0.1570248 -0.1200000 -0.09243697 -0.1603053 -0.08620690 -0.084112150
##  [2,] -0.1282051 -0.1162791 -0.12000000 -0.1544715 -0.11290323 -0.113043478
##  [3,] -0.2116788 -0.1875000 -0.13559322 -0.1239669 -0.11290323 -0.025210084
##  [4,] -0.2738854 -0.1525424 -0.12068966 -0.1641791 -0.14285714 -0.008547009
##  [5,] -0.2198582 -0.1130435 -0.12903226 -0.1250000 -0.08064516 -0.026548673
##            [,165]      [,166]       [,167]      [,168]      [,169]      [,170]
##  [1,] -0.13600000 -0.15068493 -0.109375000 -0.01851852  0.00000000 -0.03389831
##  [2,] -0.17355372 -0.11111111 -0.076923077 -0.07692308 -0.12068966 -0.09278351
##  [3,]  0.05555556  0.04950495 -0.032786885 -0.06569343 -0.09230769 -0.07142857
##  [4,]  0.11926606  0.04424779 -0.026548673 -0.04424779 -0.01694915 -0.06153846
##  [5,] -0.02521008 -0.03225806  0.009009009  0.00000000  0.04000000  0.03333333
##            [,171]       [,172]      [,173]     [,174]     [,175]      [,176]
##  [1,] -0.01075269  0.028571429  0.13821138 0.27819549 0.39682540  0.32786885
##  [2,] -0.02272727 -0.037037037 -0.00990099 0.09375000 0.23188406  0.35877863
##  [3,] -0.09565217 -0.061728395 -0.02439024 0.03636364 0.11811024  0.25984252
##  [4,] -0.03333333  0.021739130  0.09859155 0.02380952 0.06122449  0.01785714
##  [5,] -0.01694915 -0.009174312  0.10843373 0.08860759 0.05747126 -0.13513514
##            [,177]      [,178]     [,179]      [,180]       [,181]      [,182]
##  [1,] -0.04964539 -0.21387283 -0.2227979 -0.22994652 -0.108280255  0.05479452
##  [2,]  0.17910448 -0.16167665 -0.2375691 -0.21686747 -0.052631579 -0.07096774
##  [3,]  0.17037037 -0.17880795 -0.2564103 -0.17241379  0.006711409 -0.05806452
##  [4,] -0.03759398 -0.18248175 -0.1200000 -0.09230769  0.007518797  0.01408451
##  [5,] -0.17142857 -0.04477612  0.2222222  0.05357143  0.133333333  0.20312500
##             [,183]    [,184]    [,185]    [,186]    [,187]    [,188]    [,189]
##  [1,]  0.229629630 0.3385827 0.3508772 0.2363636 0.1282051 0.1028037 0.1182796
##  [2,]  0.019867550 0.2575758 0.2920354 0.2800000 0.2190476 0.2321429 0.1724138
##  [3,] -0.006451613 0.1739130 0.1846154 0.1562500 0.2678571 0.4128440 0.3643411
##  [4,]  0.080000000 0.1739130 0.3120000 0.1969697 0.2452830 0.3827160 0.2929293
##  [5,]  0.303030303 0.3821138 0.4821429 0.3666667 0.2222222 0.1645570 0.1428571
##           [,190]     [,191]     [,192]      [,193]      [,194]     [,195]
##  [1,] 0.09677419 0.14285714 0.04761905  0.08163265  0.13461538 0.21276596
##  [2,] 0.10714286 0.02702703 0.22962963  0.10400000  0.04918033 0.09259259
##  [3,] 0.33823529 0.26027397 0.22962963  0.22758621  0.10948905 0.04201681
##  [4,] 0.25000000 0.25352113 0.28571429  0.13235294  0.10294118 0.09090909
##  [5,] 0.13636364 0.09803922 0.12903226 -0.01818182 -0.01666667 0.03508772
##           [,196]      [,197]      [,198]      [,199]      [,200]      [,201]
##  [1,] 0.08163265  0.04950495  0.00000000 -0.05426357 -0.14925373 -0.12307692
##  [2,] 0.03921569  0.00000000  0.05555556 -0.04201681 -0.04201681 -0.06896552
##  [3,] 0.01818182 -0.03278689 -0.08571429 -0.06060606 -0.03816794 -0.09230769
##  [4,] 0.04273504 -0.06349206 -0.13432836 -0.06870229  0.09489051  0.00000000
##  [5,] 0.03333333 -0.09090909 -0.09090909 -0.03092784  0.18811881  0.31092437
##            [,202]      [,203]       [,204]      [,205]     [,206]     [,207]
##  [1,] -0.06086957  0.07812500 -0.006060606  0.06569343 0.08396947 0.20967742
##  [2,]  0.06122449  0.02564103 -0.113300493 -0.02352941 0.04964539 0.14049587
##  [3,]  0.06250000  0.01960784 -0.064102564  0.04000000 0.03703704 0.02479339
##  [4,]  0.04081633  0.14285714  0.151515152  0.19298246 0.04424779 0.01851852
##  [5,]  0.18699187 -0.06796117  0.123595506  0.18072289 0.24000000 0.32608696
##           [,208]      [,209]     [,210]     [,211]      [,212]     [,213]
##  [1,] 0.32812500  0.31884058 0.04347826 -0.1071429 -0.10204082 -0.1111111
##  [2,] 0.31481481  0.51304348 0.40909091  0.1969697 -0.02013423 -0.1860465
##  [3,] 0.23364486  0.30508475 0.38211382  0.2561983 -0.03267974 -0.1529412
##  [4,] 0.03389831  0.06666667 0.21100917  0.2727273  0.10937500  0.1093750
##  [5,] 0.19565217 -0.05882353 0.08771930  0.2844037  0.39449541  0.4181818
##           [,214]     [,215]     [,216]      [,217]       [,218]      [,219]
##  [1,] -0.19444444 0.00990099 0.23404255  0.23364486  0.000000000  0.20689655
```

```
## [2,] -0.15942029 0.03636364 0.20000000  0.12820513  0.007407407  0.00000000
## [3,]  0.08571429 0.09565217 0.23076923  0.18333333  0.055118110 -0.06666667
## [4,]  0.43434343 0.27835052 0.04672897  0.08620690  0.186991870  0.03759398
## [5,]  0.38461538 0.32673267 0.16666667 -0.06422018 -0.107692308 -0.01492537
##           [,220]     [,221]      [,222]      [,223]      [,224]      [,225]
## [1,]  0.19083969 0.20689655  0.04065041 -0.10937500 -0.18840580 -0.13475177
## [2,]  0.04065041 0.08196721 -0.04273504 -0.14285714 -0.22666667 -0.10958904
## [3,]  0.09090909 0.08196721 -0.09523810 -0.17557252 -0.16279070 -0.01639344
## [4,] -0.02068966 0.20895522  0.04918033 -0.07563025 -0.14728682 -0.11475410
## [5,]  0.12000000 0.15447154  0.07017544 -0.10937500 -0.07086614  0.01923077
##           [,226]      [,227]       [,228]    [,229]    [,230]       [,231]
## [1,]  0.17948718  0.23636364 -0.015625000 0.1570248 0.2258065  0.076923077
## [2,]  0.20634921  0.10937500 -0.046153846 0.1791045 0.2153846  0.008403361
## [3,]  0.09523810  0.09523810  0.000000000 0.1840000 0.2093023  0.089430894
## [4,]  0.05454545  0.00000000 -0.007633588 0.3333333 0.2571429  0.015873016
## [5,] -0.01754386 -0.07462687  0.054263566 0.2622951 0.1891892 -0.019607843
##           [,232]      [,233]       [,234]      [,235]     [,236]      [,237]
## [1,] -0.02702703  0.03636364 -0.009174312 -0.03448276 -0.1836735 -0.16030534
## [2,] -0.05263158 -0.04424779 -0.118644068 -0.17142857 -0.1946309 -0.14925373
## [3,] -0.03278689 -0.05600000 -0.056910569 -0.15873016 -0.1044776  0.06382979
## [4,] -0.06976744 -0.12857143 -0.175572519 -0.21804511  0.1317829  0.30827068
## [5,] -0.10489510 -0.10948905 -0.121212121 -0.10937500  0.1818182  0.17187500
##            [,238]       [,239]      [,240]     [,241]      [,242]      [,243]
## [1,] -0.149606299 -0.200000000 -0.22480620 -0.1428571 -0.02173913 -0.03030303
## [2,] -0.129770992 -0.191489362 -0.21985816 -0.1681416 -0.16814159  0.01204819
## [3,]  0.051094891 -0.008264463 -0.11450382 -0.2162162 -0.21367521 -0.01123596
## [4,]  0.169230769  0.008695652 -0.06896552 -0.1782946 -0.19658120 -0.14285714
## [5,] -0.008130081 -0.092436975 -0.07017544 -0.1111111 -0.19047619 -0.14285714
##           [,244]      [,245]      [,246]       [,247]      [,248]      [,249]
## [1,] -0.12962963 -0.13274336 -0.04761905 -0.070707071 -0.10638298 -0.04444444
## [2,]  0.07865169  0.00000000  0.06000000  0.020833333 -0.04081633 -0.14285714
## [3,]  0.12195122  0.11578947  0.08411215 -0.084112150 -0.16239316 -0.20312500
## [4,] -0.02000000 -0.08108108 -0.14728682 -0.194029851 -0.21126761 -0.15492958
## [5,] -0.13114754 -0.15200000 -0.16923077 -0.007633588  0.02040816  0.04000000
##           [,250]      [,251]     [,252]      [,253]     [,254]     [,255]
## [1,] -0.14285714 -0.09433962 -0.1228070 -0.21538462 -0.2352941 -0.1472868
## [2,] -0.13725490 -0.18699187 -0.1428571 -0.10236220 -0.1755725 -0.1967213
## [3,] -0.19327731  0.03174603  0.1360000  0.05511811 -0.1250000 -0.1652174
## [4,] -0.11290323  0.13821138  0.1968504  0.02985075 -0.1532847 -0.1811024
## [5,] -0.03030303 -0.09243697 -0.1219512 -0.11940299 -0.1159420 -0.1641791
##           [,256]      [,257]      [,258]      [,259]     [,260]     [,261]
## [1,] -0.14728682 -0.03937008 -0.11940299 -0.23943662 -0.2380952 -0.2000000
## [2,]  0.00000000  0.05263158 -0.20661157 -0.25373134 -0.2096774 -0.1911765
## [3,] -0.09473684 -0.09473684 -0.07526882 -0.18965517 -0.1612903 -0.1666667
## [4,] -0.18867925 -0.13333333 -0.05882353 -0.15686275 -0.2000000 -0.1578947
## [5,] -0.20000000 -0.12871287 -0.05617978 -0.05747126 -0.1698113 -0.1935484
##          [,262]      [,263]      [,264]      [,265]     [,266]     [,267]
## [1,] -0.2030075 -0.16800000 -0.17037037 -0.18881119 -0.2000000 -0.2061069
## [2,] -0.1798561 -0.13235294 -0.13286713 -0.12592593 -0.1773050 -0.2142857
## [3,] -0.1406250 -0.04918033 -0.08527132 -0.05982906 -0.1278195 -0.1780822
## [4,] -0.1562500 -0.17293233 -0.16129032 -0.10000000 -0.1450382 -0.1159420
## [5,] -0.1818182 -0.20833333 -0.11111111 -0.12592593 -0.1510791 -0.1555556
##           [,268]     [,269]     [,270]     [,271]     [,272]      [,273]
## [1,] -0.16666667 -0.1368421 -0.1546392 -0.1650485 -0.2075472 -0.18750000
## [2,] -0.21481481 -0.1478261 -0.1020408 -0.1730769 -0.1800000 -0.17525773
## [3,] -0.20779221 -0.2056738 -0.1639344 -0.1531532 -0.1224490 -0.11764706
## [4,] -0.09859155 -0.1884058 -0.2408759 -0.1746032 -0.1250000 -0.07692308
## [5,] -0.08029197 -0.1515152 -0.2535211 -0.2317881 -0.1250000 -0.11578947
##          [,274]     [,275]     [,276]     [,277]     [,278]     [,279]
## [1,] -0.1340206 -0.1809524 -0.1923077 -0.1681416 -0.1372549 -0.1702128
## [2,] -0.1157895 -0.1743119 -0.1962617 -0.2068966 -0.1607143 -0.1515152
## [3,] -0.1600000 -0.1809524 -0.1960784 -0.2212389 -0.1780822 -0.1269841
## [4,] -0.1886792 -0.1600000 -0.2156863 -0.2110092 -0.1972789 -0.1492537
## [5,] -0.2000000 -0.1919192 -0.2079208 -0.1515152 -0.1891892 -0.2173913
##          [,280]     [,281]     [,282]     [,283]      [,284]      [,285]
## [1,] -0.2656250 -0.2112676 -0.1654676 -0.1603053 -0.14049587 -0.07547170
## [2,] -0.2033898 -0.1970803 -0.1515152 -0.1239669 -0.13043478 -0.07843137
## [3,] -0.2320000 -0.2516129 -0.2317881 -0.2000000 -0.06666667  0.10091743
## [4,] -0.2575758 -0.3000000 -0.2820513 -0.1653543  0.07692308  0.20000000
## [5,] -0.1650485 -0.2977099 -0.3000000 -0.1641791  0.03125000  0.10447761
##           [,286]      [,287]      [,288]      [,289]      [,290]     [,291]
## [1,] -0.04854369 -0.13559322 -0.13274336 -0.13274336 -0.09615385 -0.1711712
## [2,] -0.07692308 -0.11111111 -0.09734513 -0.07142857 -0.11926606 -0.2000000
## [3,]  0.03921569 -0.09909910 -0.15000000 -0.12307692 -0.12307692 -0.1929825
## [4,]  0.07964602 -0.09090909 -0.10091743 -0.09523810 -0.14925373 -0.1833333
## [5,]  0.07692308 -0.18750000 -0.18260870 -0.19327731 -0.12903226 -0.1228070
##           [,292]       [,293]      [,294]      [,295]       [,296]      [,297]
## [1,] -0.09565217 -0.109090909  0.04950495  0.02000000  0.070707071  0.06422018
## [2,] -0.08928571 -0.009708738 -0.09677419 -0.01724138 -0.009174312 -0.07936508
```

```
## [3,] -0.17431193 -0.125000000 -0.19083969 -0.15447154 -0.065420561 -0.04587156
## [4,] -0.17543860 -0.145161290 -0.08256881 -0.09278351 -0.161904762 -0.14782609
## [5,] -0.17948718 -0.111111111 -0.06194690 -0.07272727 -0.163636364 -0.18918919
##           [,298]      [,299]      [,300]      [,301]      [,302]     [,303]
## [1,] -0.05982906 -0.16363636 -0.22000000 -0.22033898 -0.12962963 -0.1546392
## [2,] -0.08270677 -0.07142857 -0.16981132 -0.18333333 -0.13725490 -0.1818182
## [3,] -0.02564103  0.00000000 -0.10091743 -0.08333333 -0.07272727 -0.1132075
## [4,] -0.04587156 -0.04000000 -0.09259259  0.15714286  0.03508772 -0.1600000
## [5,] -0.16521739 -0.20353982  0.00000000 -0.18620690 -0.22727273 -0.2272727
##          [,304]       [,305]      [,306]       [,307]      [,308]      [,309]
## [1,] -0.1521739 -0.072164948 -0.19685039 -0.123966942  0.06666667 -0.02000000
## [2,] -0.2173913  0.009345794  0.00000000  0.008695652  0.00000000  0.00000000
## [3,] -0.1500000 -0.043478261 -0.02752294 -0.046728972  0.00000000 -0.07272727
## [4,] -0.2000000 -0.130434783 -0.15789474 -0.177777778 -0.08163265 -0.01851852
## [5,] -0.2037037 -0.233644860 -0.20512821 -0.121951220 -0.10256410  0.19565217
##           [,310]      [,311]      [,312]      [,313]      [,314]      [,315]
## [1,] -0.11926606 -0.17073171 -0.16800000 -0.03846154  0.01234568 -0.13333333
## [2,] -0.08256881 -0.13793103 -0.16666667 -0.16949153 -0.05617978 -0.12500000
## [3,] -0.14782609 -0.17355372 -0.17557252 -0.17037037 -0.07317073 -0.07407407
## [4,] -0.07843137 -0.08256881 -0.18400000 -0.17985612 -0.11450382 -0.07142857
## [5,]  0.15596330  0.02857143 -0.05357143 -0.15873016 -0.20634921 -0.13978495
##           [,316]      [,317]      [,318]      [,319]      [,320]      [,321]
## [1,] -0.08235294 -0.02380952  0.03703704  0.36283186  0.15789474 -0.01666667
## [2,] -0.05494505 -0.02127660  0.04000000  0.32231405  0.33944954  0.26923077
## [3,] -0.09259259 -0.14545455 -0.05785124  0.06250000  0.31428571  0.40000000
## [4,] -0.03846154 -0.03846154 -0.11320755 -0.11111111 -0.08943089  0.11711712
## [5,] -0.04950495  0.03030303 -0.03846154 -0.09565217 -0.16030534 -0.03278689
##           [,322]      [,323]       [,324]      [,325]      [,326]      [,327]
## [1,] -0.06666667 -0.01666667 -0.179856115 -0.18620690 -0.10447761 -0.11267606
## [2,]  0.17431193 -0.08695652 -0.251612903 -0.19727891 -0.07200000 -0.09230769
## [3,]  0.21739130 -0.15862069 -0.205479452 -0.12230216 -0.06153846 -0.08148148
## [4,]  0.14285714 -0.06410256  0.007407407  0.08333333  0.13178295  0.11940299
## [5,] -0.07042254 -0.08695652  0.073529412  0.11111111  0.12977099  0.28455285
##           [,328]      [,329]      [,330]       [,331]      [,332]       [,333]
## [1,] -0.13669065 -0.09259259 -0.05154639 -0.126984127 -0.14503817 -0.133858268
## [2,] -0.12820513 -0.12500000  0.15094340  0.000000000 -0.09836066 -0.134615385
## [3,]  0.02752294  0.09259259  0.15789474  0.129629630  0.12962963 -0.084112150
## [4,]  0.10937500  0.28695652  0.21538462  0.008547009  0.09909910 -0.033898305
## [5,]  0.28205128  0.32800000  0.14285714  0.041322314  0.04347826  0.007751938
##           [,334]      [,335]      [,336]       [,337]      [,338]      [,339]
## [1,] -0.08474576 -0.17391304 -0.22666667 -0.125000000  0.03773585  0.12068966
## [2,] -0.04950495 -0.15625000 -0.21495327 -0.243589744 -0.09375000  0.06086957
## [3,] -0.06930693 -0.03636364 -0.11111111 -0.115646259 -0.21379310 -0.20833333
## [4,]  0.01492537 -0.01515152 -0.09836066 -0.115646259  0.03921569 -0.22285714
## [5,] -0.02941176 -0.06153846 -0.11290323  0.008403361  0.04854369 -0.20441989
##           [,340]      [,341]      [,342]      [,343]      [,344]      [,345]
## [1,]  0.09259259  0.06521739  0.02439024 -0.05494505  0.05769231  0.04761905
## [2,]  0.04000000  0.17543860  0.15044248 -0.06796117  0.04504505 -0.04672897
## [3,] -0.14285714 -0.09565217 -0.02127660  0.03225806 -0.01960784  0.01724138
## [4,] -0.28571429 -0.11111111 -0.23404255 -0.16923077 -0.09090909 -0.07692308
## [5,] -0.33990148 -0.29113924 -0.21568627 -0.10588235 -0.07142857 -0.10714286
##           [,346]      [,347]      [,348]      [,349]      [,350]      [,351]
## [1,] -0.06521739 -0.03921569  0.10924370  0.18110236  0.14285714  0.11111111
## [2,] -0.04672897 -0.10091743  0.08510638  0.37815126  0.35384615  0.26050420
## [3,]  0.06250000 -0.02439024 -0.03921569  0.32692308  0.27083333  0.10169492
## [4,]  0.04255319 -0.13432836 -0.23893805 -0.07865169  0.00000000 -0.09836066
## [5,] -0.07070707 -0.11278195 -0.14728682 -0.19626168 -0.09433962 -0.09677419
##           [,352]
## [1,] 0.024390244
## [2,] 0.090909091
## [3,] 0.008130081
## [4,] 0.000000000
## [5,] 0.057692308
```

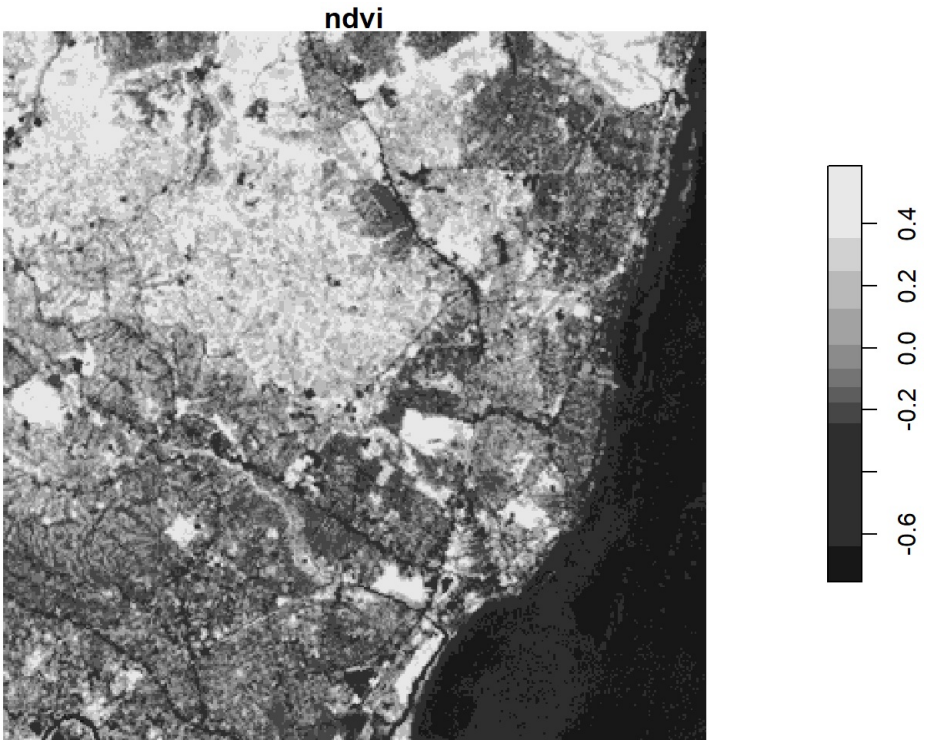
2. Compute NDVI for the S2 image, using st_apply and an a function ndvi = function(x) (x[4]-x[3])/(x[4]+x[3]). Plot the result, and write the result to a GeoTIFF. Explain the difference in runtime between plotting and writing.

st_apply and function ndvi

```
ndvi = function(x) (x[4]-x[3])/(x[4]+x[3])
b<-st_apply(x, c("x", "y"), ndvi)
```

Plot the result and write the result to a GeoTIFF

```
plot(b)
```



Explain the difference in runtime between plotting and writing.

El tiempo en graficar es más lento que el de exportar los archivos, la exportación guarda las característica pero los gráficos reconstruyen la información apartir de los cálculos maltriciales que constituyen el objeto

```
tf = tempfile(fileext=".tif")
write_stars(b, tf)
```

3. Use st_transform to transform the stars object read from L7_ETMs.tif to EPSG 4326. Print the object. Is this a regular grid? Plot the first band using arguments axes=TRUE and border=NA, and explain why this takes such a long time.

Transform the stars object read from L7_ETMs.tif to EPSG 4326.Print the object? Is this a regular grid?

Es una grilla curvilínea, que se grafica con las coordenadas polares.

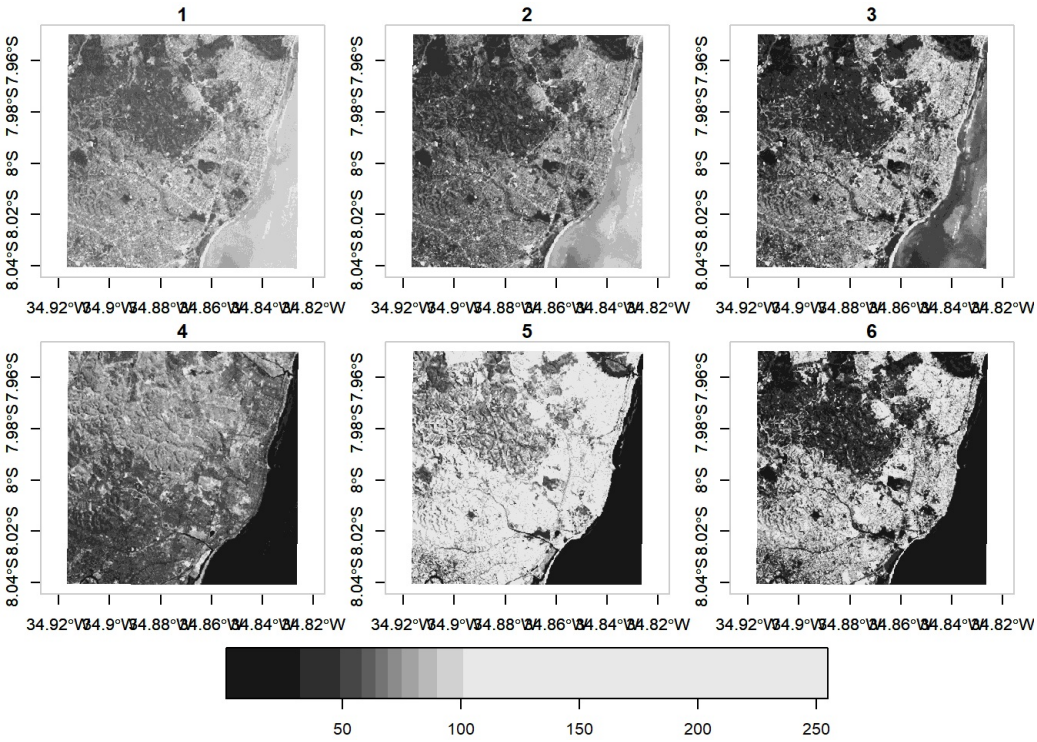
```
c=st_transform(x,4326)
print(c)
```

```
## stars object with 3 dimensions and 1 attribute
## attribute(s):
##   L7_ETMs.tif
##  Min.   :  1.00
##  1st Qu.: 54.00
##  Median : 69.00
##  Mean   : 68.91
##  3rd Qu.: 86.00
##  Max.   :255.00
## dimension(s):
##      from  to offset delta refsys point                          values
## x       1 349     NA    NA WGS 84 FALSE [349x352] -34.9165,...,-34.8261 [x]
## y       1 352     NA    NA WGS 84 FALSE  [349x352] -8.0408,...,-7.94995 [y]
## band    1   6     NA    NA     NA    NA                             NULL
## curvilinear grid
```

Plot the first band using arguments axes=TRUE and border=NA, and explain why this takes such a long time.

Porque su elaboración involucra colocar en un plano de coordenadas polares todos los elementos que conforman este objeto espacial.Además, no tiene proyección lo que repercute en la estiamción solicitada

```
plot(c,axes=TRUE, border=NA)
```



4. Use st_warp to warp the L7_ETMs.tif object to EPSG 4326, and plot the resulting object with axes=TRUE. Why is the plot created much faster than after st_transform?

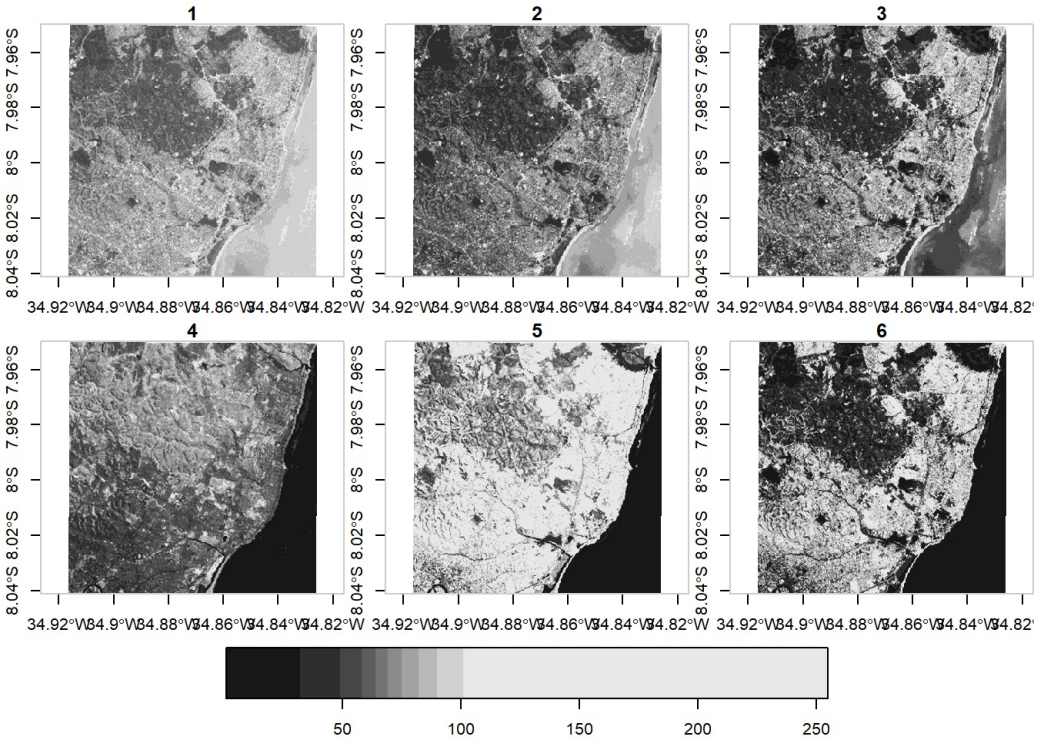
Porque transforma una grilla en una nueva con formato stars y lo hace en un nuevo sistema de coordenadas geográficas. Es decir, incluye la proyección

```
c_warp = st_warp(x, crs=4326)
```

Plot the resulting object with axes=TRUE. Why is the plot created much faster than after st_transform?**

En este caso el objeto ya incluyo la transformación indicada en el punto 4, lo que facilita la estimación.

```
plot(c_warp, axes=TRUE)
```



Ejercicios 6.6

1. Add a variable to the nc dataset by nc$State = “North Carolina”. Which value should you attach to this variable for the attribute-geometry relationship (agr)?

```
nc <- system.file("gpkg/nc.gpkg", package="sf") %>%
    read_sf() %>%
    st_transform(32119)

nc <- nc %>%
  mutate(State = "North Carolina")

st_agr(nc)
```

```
##      AREA PERIMETER     CNTY_   CNTY_ID      NAME      FIPS    FIPSNO  CRESS_ID
##      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>
##     BIR74     SID74   NWBIR74     BIR79     SID79   NWBIR79     State
##      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>
## Levels: constant aggregate identity
```

2. Create a new sf object from the geometry obtained by st_union(nc), and assign “North Carolina” to the variable State. Which agr can you now assign to this attribute variable?

```
nc2 <-st_union(nc)
```

3. Use st_area to add a variable with name area to nc. Compare the area and AREA variables in the nc dataset. What are the units of AREA? Are the two linearly related? If there are discrepancies, what could be the cause?

Al comparar ambos se observa que AREA no está en las medidas adecuados para un objeto geográfico, mientras area con si pues al utilizar st_area se expresa en metros cuadros. Las mismas si están asociadas linealmente, como se presenta en el gráfico de dispersión

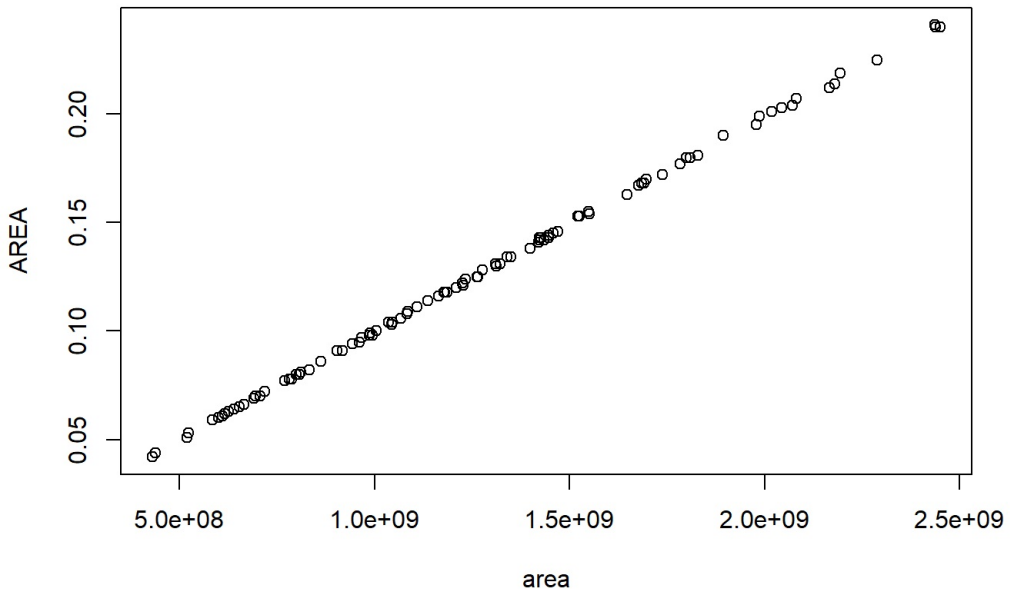
```
nc$area <- st_area(nc)
names(nc)
```

```
##  [1] "AREA"      "PERIMETER" "CNTY_"     "CNTY_ID"   "NAME"      "FIPS"
##  [7] "FIPSNO"    "CRESS_ID"  "BIR74"     "SID74"     "NWBIR74"   "BIR79"
## [13] "SID79"     "NWBIR79"   "geom"      "State"     "area"
```

```
var_area <- nc[,c(1,17)]
var_area
```

```
## Simple feature collection with 100 features and 2 fields
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: 123829.8 ymin: 14740.06 xmax: 930518.6 ymax: 318255.5
## projected CRS:  NAD83 / North Carolina
## # A tibble: 100 x 3
##     AREA       area                                                    geom
##    <dbl>      [m^2]                                       <MULTIPOLYGON [m]>
##  1 0.114 1137590142 (((387344.7 278382.4, 381334.1 282769, 379438.2 282938.4, 3~
##  2 0.061  611196991 (((408601.4 292419.4, 408564.7 293980.2, 406642.5 296868, 4~
##  3 0.143 1423728763 (((478715.7 277484.2, 476934.3 278861, 471501.4 279166.5, 4~
##  4 0.07   694661115 (((878193.4 289118.8, 877381 291107.7, 875993.5 290871.3, 8~
##  5 0.153 1520991764 (((769834.9 277787.4, 768364 274833.2, 762615.4 274392, 763~
##  6 0.097  967855261 (((812327.7 277867.5, 791157.9 277003.1, 789882 277570.4, 7~
##  7 0.062  616042081 (((878193.4 289118.8, 883270.1 275304.7, 881179.6 272571.7,~
##  8 0.091  903814094 (((828444.5 290086.5, 824767.3 287156.9, 820818.1 287099.3,~
##  9 0.118 1179531734 (((671746.3 278680.2, 674042.4 282229.8, 670407.3 310353.5,~
## 10 0.124 1232968370 (((517435.1 277851.9, 479039 279092, 481101.2 311686.7, 515~
## # ... with 90 more rows
```

```
plot(var_area$area,var_area$AREA, xlab="area", ylab="AREA")
```



4.Is the area variable intensive or extensive? Is its agr equal to constant, identity or aggregate?

La variable area es una variable extensiva porque depende de la cantidad en la que se mide, pues tiene la propiedad de ser porporcional al tamaño del sistema que describe. Por esta razón, se adiciona debe utilizarse el argumento “constant”, el valor de este atributo es válido en toda la geometría que se esté analizando.

5. Find the name of the county that contains POINT(-78.34046 35.017)

```
#Se carga el objeto usando la proyección 4326
nc2 <- system.file("gpkg/nc.gpkg", package="sf") %>%
       read_sf() %>%
       st_transform(4326)


point<- st_point( c( -78.34046, 35.017 ) ) # Se crea el punto
iden <- st_contains(nc2, point) # se identifican el pinto en el objeto sf
```

```
## although coordinates are longitude/latitude, st_contains assumes that they are planar
```

```
iden<-summary(iden)


# Se procede a transformar el objeto en númerico para filtrar el que cumple con la condición 1, que indica que si lo contiene.

iden_Num <- as.numeric(iden[,1])
log <- which( iden_Num==1 )
nc2$NAME[log]
```

```
## [1] "Sampson"
```

6. Find the names of all counties with boundaries that touch county Sampson.

```
samp <- nc2$geom[log]
samp
```

```
## Geometry set for 1 feature
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: -78.66615 ymin: 34.55387 xmax: -78.11346 ymax: 35.31522
## geographic CRS: WGS 84
```

```
## MULTIPOLYGON (((-78.1135 34.7211, -78.11346 34....
```

```
intersect <- st_intersection(nc2, samp)
```

```
## although coordinates are longitude/latitude, st_intersection assumes that they are planar
```

```
## Warning: attribute variables are assumed to be spatially constant throughout all
## geometries
```

```
intersect  <- intersect [ intersect $NAME!='Sampson',]; intersect $NAME
```

```
## [1] "Johnston"   "Wayne"      "Harnett"    "Cumberland" "Duplin"
## [6] "Bladen"     "Pender"
```

7. List the names of all counties that are less than 50 km away from county Sampson.

```
all <- st_is_within_distance(nc2, samp, dist = 50000)
all<- summary(all)
all <- as.numeric( all[,1])
log_all<- which( all==1 )
nc2$NAME[log_all]
```

```
##  [1] "Wake"        "Chatham"     "Wilson"      "Johnston"    "Greene"
##  [6] "Lee"         "Wayne"       "Harnett"     "Moore"       "Lenoir"
## [11] "Sampson"     "Cumberland"  "Jones"       "Hoke"        "Duplin"
## [16] "Onslow"      "Robeson"     "Bladen"      "Pender"      "Columbus"
## [21] "New Hanover" "Brunswick"
```

Ejercicios 8.7 Lovelace

```
africa = world %>%
  filter(continent == "Africa", !is.na(iso_a2)) %>%
  left_join(worldbank_df, by = "iso_a2") %>%
  dplyr::select(name, subregion, gdpPercap, HDI, pop_growth) %>%
  st_transform("+proj=aea +lat_1=20 +lat_2=-23 +lat_0=0 +lon_0=25")
```

```
zion = st_read((system.file("vector/zion.gpkg", package = "spDataLarge")))
```

```
## Reading layer `zion' from data source `C:\Users\Katherine\Documents\R\win-library\4.0\spDataLarge\vector\zion.g
pkg' using driver `GPKG'
## Simple feature collection with 1 feature and 11 fields
## geometry type:  POLYGON
## dimension:      XY
## bbox:           xmin: 302903.1 ymin: 4112244 xmax: 334735.5 ymax: 4153087
## projected CRS:  UTM Zone 12, Northern Hemisphere
```

```
data(nlcd, package = "spDataLarge")
```

Create a map showing the geographic distribution of the Human Development Index (HDI) across Africa with base graphics (hint: use plot()) and tmap packages (hint: use tm_shape(africa) + …).

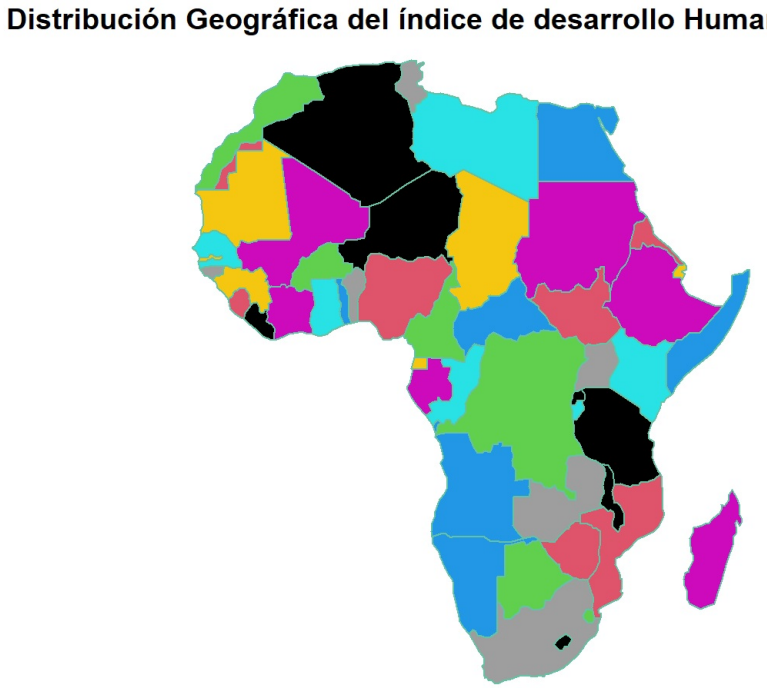
- Name two advantages of each based on the experience.
- Name three other mapping packages and an advantage of each.
- Bonus: create three more maps of Africa using these three packages.

**Map with base graphics (plot)

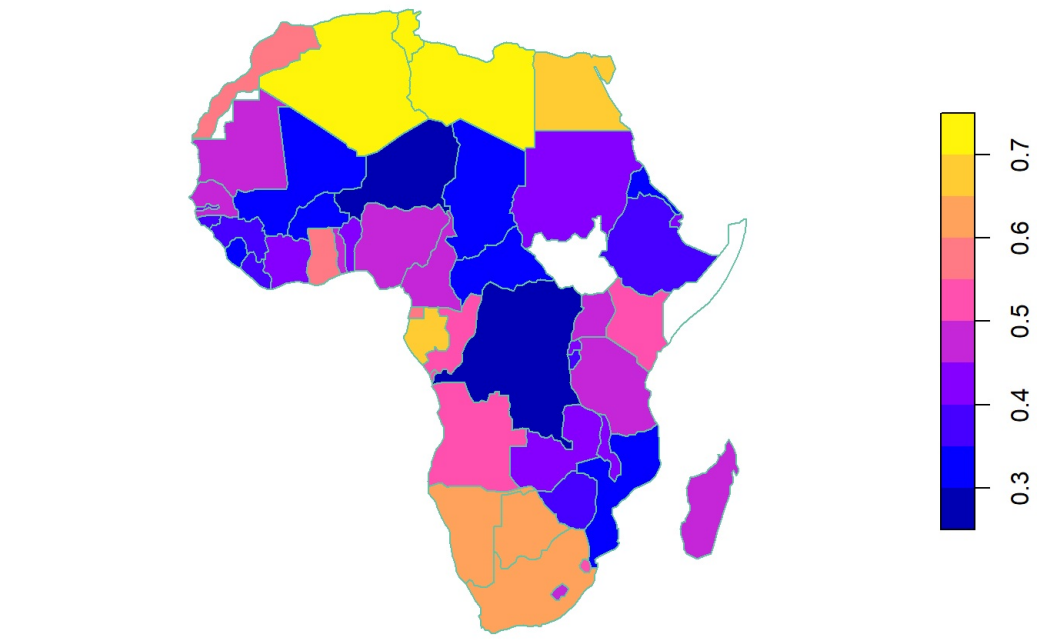
```
HDI =  africa%>%
       dplyr::select( HDI)

plot(HDI, main="Distribución Geográfica del índice de desarrollo Humano en África", col=palette("Set 2"))
```

```
## Warning in plot.sf(HDI, main = "Distribución Geográfica del índice de desarrollo
## Humano en África", : col is not of length 1 or nrow(x): colors will be recycled;
## use pal to specify a color palette
```



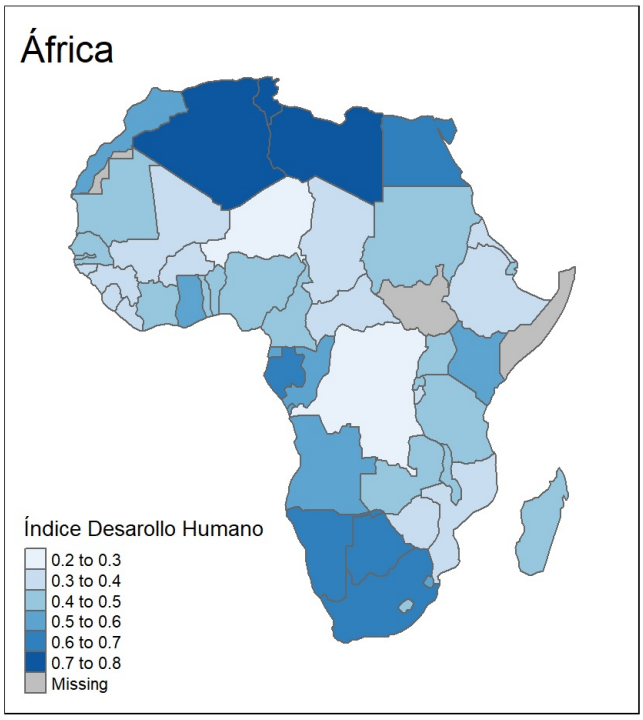

```
plot(HDI, main="Distribución Geográfica del índice de desarrollo Humano en África")
```



Map with tmap packages

```
map_hdi1 = tm_shape(africa) +
  tm_fill(col = "HDI",palette = "Blues", title = "Índice Desarollo Humano")+
  tm_borders()+
  tm_lines+
  tm_layout(title = "África", title.size = 1.5,
            legend.position = c("left", "bottom"), inner.margins = 0.1)


map_hdi1
```



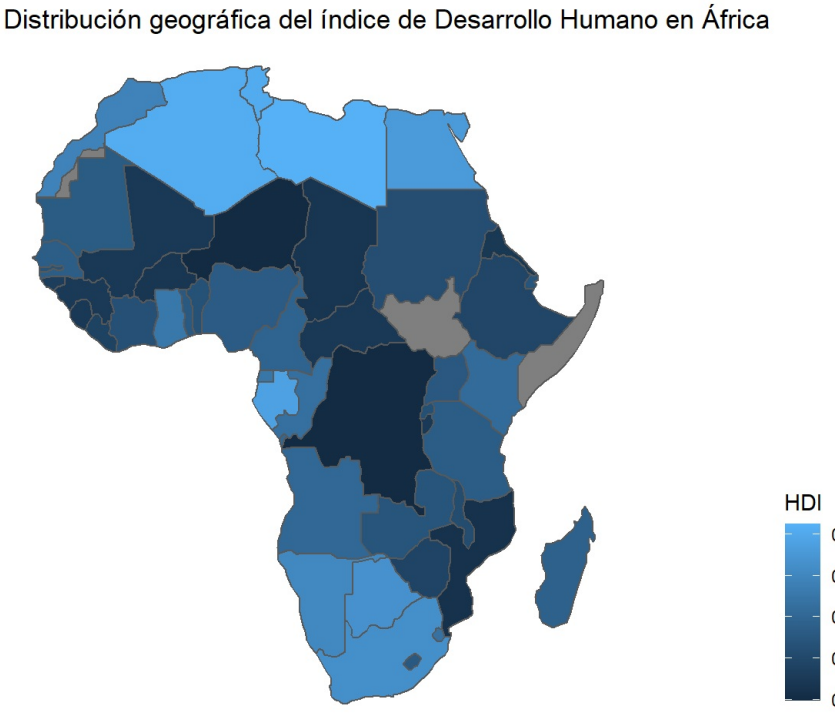
Name three other mapping packages and an advantage of each.

- ggplot2: genera muy buenas visualizaciones de datos pero no está diseñada para manejar datos espaciales, debe combinarse con el paquete sf.

```
# Mapa base sencillo de La Rioja
#install.packages('ggspatial')
library(ggspatial)
```

```
## Warning: package 'ggspatial' was built under R version 4.0.2
```

```
ggplot(data = africa) +
  geom_sf(aes(fill = HDI))+
    ggtitle("Distribución geográfica del índice de Desarrollo Humano en África")+
  theme_void() +##Elimina escalas, marcas de coordenadas, etc.
   theme(legend.position = c(1.3, 0.2))
```



- leaflet: Facilidad para crear cartogramas y mapas interactivos

A continuación se escribe el código dado que no hace mapas interactivos.

```
#leaflet() %>%
  #addTiles() %>%
  # addPolygons(data = st_transform(africa, 4326))
```

- Cartograhy: Utilidad para crear cartogramas

```
library(cartogram)
```

```
## Warning: package 'cartogram' was built under R version 4.0.2
```

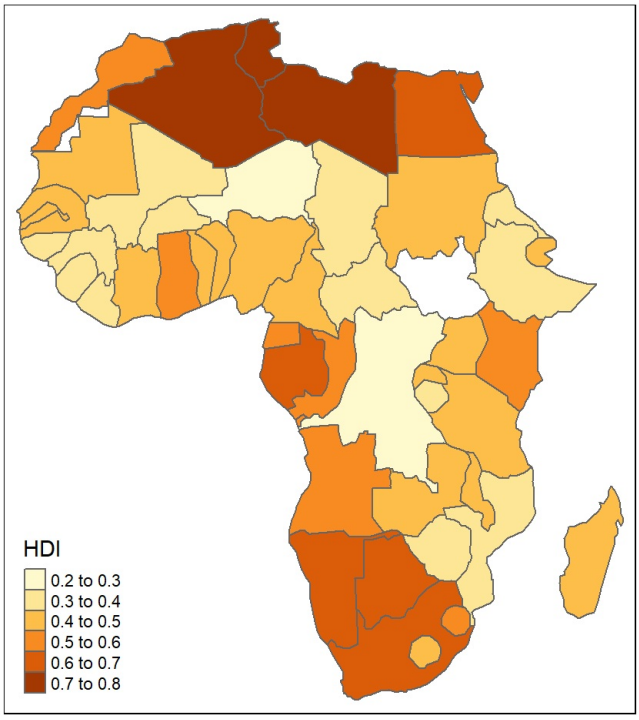
```
carto = cartogram_cont(africa, "HDI", itermax = 2)
```

```
## Warning in cartogram_cont.sf(africa, "HDI", itermax = 2): NA not allowed in
## weight vector. Features will be removed from Shape.
```

```
## Mean size error for iteration 1: 6.03537025368672
```

```
## Mean size error for iteration 2: 3.64410634493783
```

```
tm_shape(carto) + tm_polygons("HDI")
```



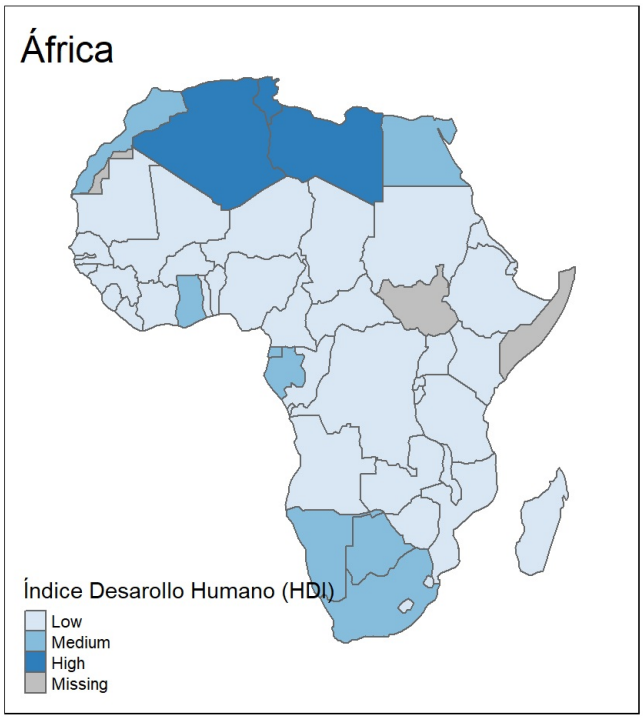
2. Extend the tmap created for the previous exercise so the legend has three bins: “High” (HDI above 0.7), “Medium” (HDI between 0.55 and 0.7) and “Low” (HDI below 0.55).

Bonus: improve the map aesthetics, for example by changing the legend title, class labels and color palette.

```
map_hdi2 = tm_shape(africa) +
  tm_polygons(col = "HDI", palette = "Blues",
              title = "Índice Desarollo Humano (HDI)",
              breaks = c(0, 0.55, 0.7, 1),
              labels = c("Low", "Medium", "High"))+
  tm_borders()+
  tm_lines+
  tm_layout(title = "África", title.size = 1.5,
            legend.position = c("left", "bottom"), inner.margins = 0.1,
            legend.width = 400000)


map_hdi2
```

```
## Warning: One tm layer group has duplicated layer types, which are omitted. To
## draw multiple layers of the same type, use multiple layer groups (i.e. specify
## tm_shape prior to each of them).
```

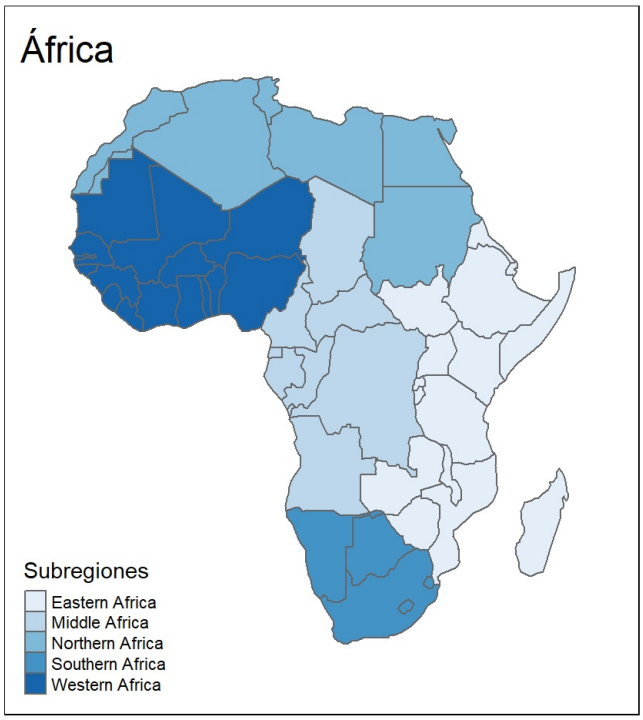


Represent africa's subregions on the map. Change the default color palette and legend title. Next, combine this map and the map created in the previous exercise into a single plot.

Mapa de subrregiones de África

```
map_subr = tm_shape(africa) +
  tm_fill(col = "subregion",palette = "Blues", title = "Subregiones")+
  tm_borders()+
  tm_lines+
  tm_layout(title = "África", title.size = 1.5,
            legend.position = c("left", "bottom"), inner.margins = 0.1,
            legend.width = 400000)


map_subr
```



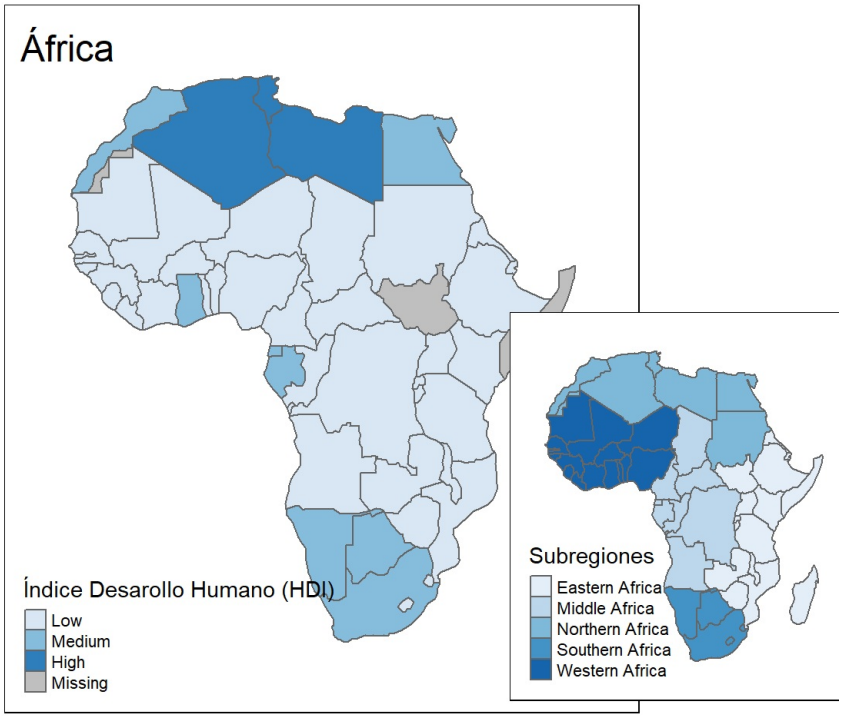
Combinar dos mapas: HDI y Subregiones

```
map_subr2 = tm_shape(africa) +
  tm_fill(col = "subregion",palette = "Blues", title = "Subregiones")+
  tm_borders()+
  tm_lines+
  tm_layout(title = "", title.size = 1.5,
            legend.position = c("left", "bottom"), inner.margins = 0.1,
            legend.width = 400000)


 #junto los dos
     map_hdi2
```

```
## Warning: One tm layer group has duplicated layer types, which are omitted. To
## draw multiple layers of the same type, use multiple layer groups (i.e. specify
## tm_shape prior to each of them).
```

```
      print(map_subr2, vp = grid::viewport(0.85, 0.3, width = 0.35, height = 2.5))
```



4. Create a land cover map of the Zion National Park.

- Change the default colors to match your perception of the land cover categories
- Add a scale bar and north arrow and change the position of both to improve the map's aesthetic appeal
- Bonus: Add an inset map of Zion National Park's location in the context of the Utah state. (Hint: an object representing Utah can be subset from the us_states dataset.)

Mapa de zion utilizando el raster nlcd (National Land Cover Database) y con los colores por default

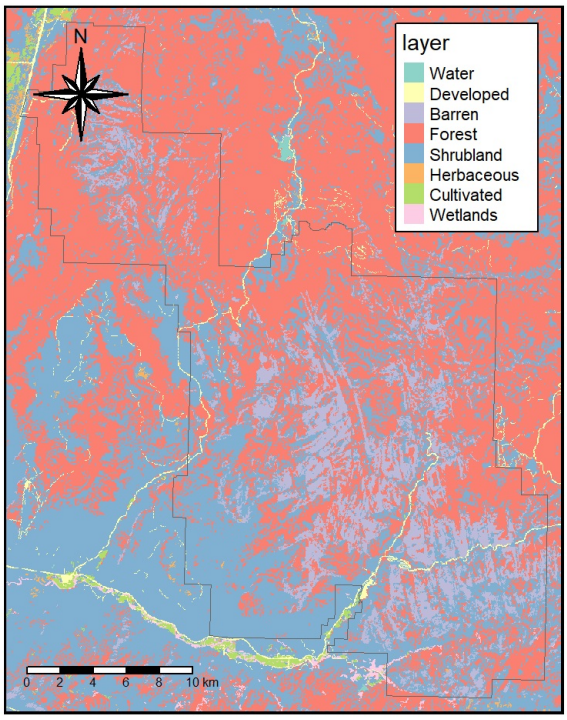
```
map_zion1 = tm_shape(nlcd) + tm_raster() +
  tm_shape(zion) + tm_borders(lwd = 0.5) +
  tm_scale_bar(size = 0.5, position = c("left", "bottom")) +
  tm_compass(type = "8star", position = c("left", "top")) +
  tm_layout(legend.frame = TRUE, legend.position = c(0.7, "top")) +
  tm_layout(frame.lwd = 3)
```

```
## Warning: The argument size of tm_scale_bar is deprecated. It has been renamed to
## text.size
```

```
map_zion1
```

```
## stars object downsampled to 888 by 1125 cells. See tm_shape manual (argument raster.downsample)
```

```
## Warning in rep(attr(shp[[1]], "colors"), length.out = length(lvls)): 'x' is NULL
## so the result will be NULL
```



Mapa de zion utilizando el raster nlcd (National Land Cover Database) y con los colores por default

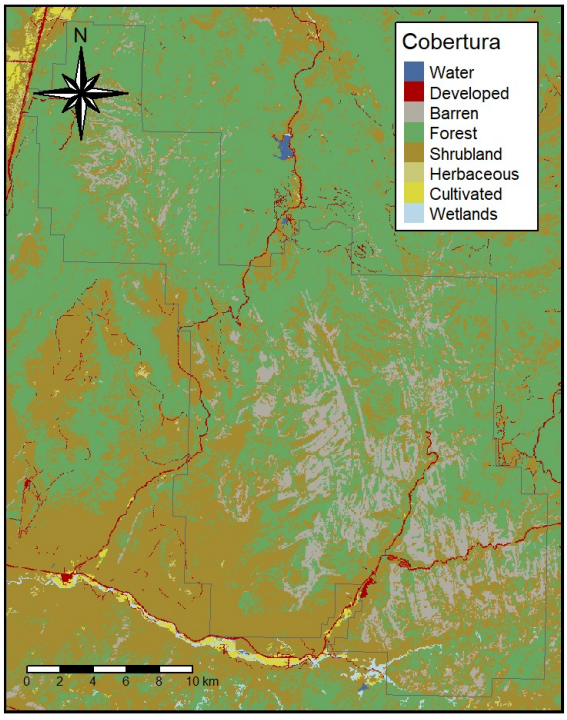
```
c1 = c("#476ba0", "#aa0000", "#b2ada3", "#68aa63", "#a58c30",
               "#c9c977", "#dbd83d", "#bad8ea")
map_zion2 = tm_shape(nlcd) + tm_raster(palette = c1, title= "Cobertura") +
  tm_shape(zion) + tm_borders(lwd = 0.5) +
  tm_scale_bar(size = 0.5, position = c("left", "bottom")) +
  tm_compass(type = "8star", position = c("left", "top")) +
  tm_layout(legend.frame = TRUE, legend.position = c(0.7, "top")) +
  tm_layout(frame.lwd = 3)
```

```
## Warning: The argument size of tm_scale_bar is deprecated. It has been renamed to
## text.size
```

```
map_zion2
```

```
## stars object downsampled to 888 by 1125 cells. See tm_shape manual (argument raster.downsample)
```

```
## Warning in rep(attr(shp[[1]], "colors"), length.out = length(lvls)): 'x' is NULL
## so the result will be NULL
```



Se agrega Utah, primero se filtra el estado de interes, utilizando el objeto sf de us_states

```
#class(us_states)
names(us_states)
```

```
## [1] "GEOID"        "NAME"         "REGION"       "AREA"         "total_pop_10"
## [6] "total_pop_15" "geometry"
```

```
us_states$NAME
```

```
##  [1] "Alabama"              "Arizona"              "Colorado"
##  [4] "Connecticut"          "Florida"              "Georgia"
##  [7] "Idaho"                "Indiana"              "Kansas"
## [10] "Louisiana"            "Massachusetts"        "Minnesota"
## [13] "Missouri"             "Montana"              "Nevada"
## [16] "New Jersey"           "New York"             "North Dakota"
## [19] "Oklahoma"             "Pennsylvania"         "South Carolina"
## [22] "South Dakota"         "Texas"                "Vermont"
## [25] "West Virginia"        "Arkansas"             "California"
## [28] "Delaware"             "District of Columbia" "Illinois"
## [31] "Iowa"                 "Kentucky"             "Maine"
## [34] "Maryland"             "Michigan"             "Mississippi"
## [37] "Nebraska"             "New Hampshire"        "New Mexico"
## [40] "North Carolina"       "Ohio"                 "Oregon"
## [43] "Rhode Island"         "Tennessee"            "Utah"
## [46] "Virginia"             "Washington"           "Wisconsin"
## [49] "Wyoming"
```

```
utah <- us_states %>%
     filter(NAME == "Utah")
utah
```

```
## Simple feature collection with 1 feature and 6 fields
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: -114.0506 ymin: 36.998 xmax: -109.0418 ymax: 42.00157
## geographic CRS: NAD83
##   GEOID NAME REGION            AREA total_pop_10 total_pop_15
## 1    49 Utah   West 219859.8 [km^2]      2657236      2903379
##                         geometry
## 1 MULTIPOLYGON (((-114.0417 4...
```

Se define el área de interes, creando un objeto espacial a partir de la función bbox

```
utah1 <-st_bbox(nlcd)%>%
        st_as_sfc()
utah1
```

```
## Geometry set for 1 feature
## geometry type:  POLYGON
## dimension:      XY
## bbox:           xmin: 301903.3 ymin: 4111244 xmax: 335735.4 ymax: 4154086
## CRS:            +proj=utm +zone=12 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
```

```
## POLYGON ((301903.3 4111244, 335735.4 4111244, 3...
```

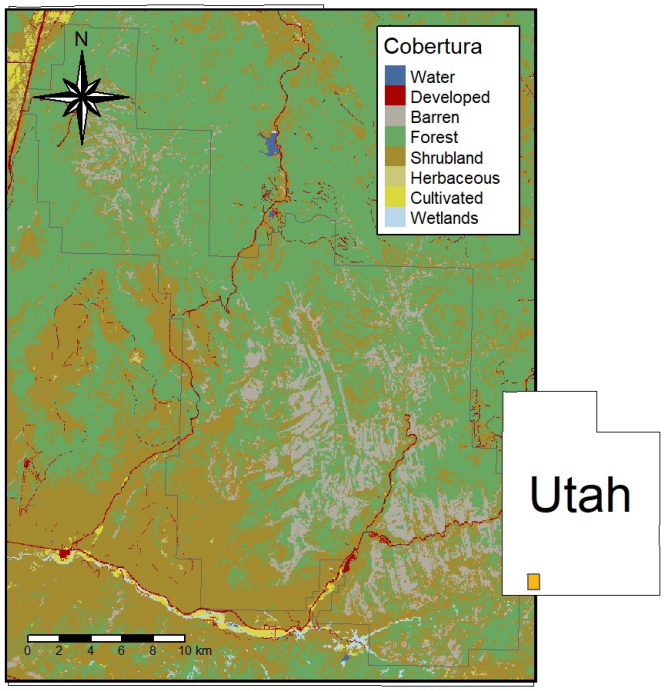
```
map_utah = tm_shape(utah) +
  tm_polygons(lwd = 0.5, border.col = "black", col = "white") +
  tm_shape(utah1) +
  tm_polygons(col = "darkgoldenrod1", lwd = 1) +
  tm_layout(title = "Utah", title.size = 2, title.position = c("center", "center")) +
  tm_layout(frame = FALSE, bg.color = NA)
map_utah

print(map_zion2, vp = grid::viewport(0.5, 0.5, width = 0.95, height = 0.95))
```

```
## stars object downsampled to 888 by 1125 cells. See tm_shape manual (argument raster.downsample)
```

```
## Warning in rep(attr(shp[[1]], "colors"), length.out = length(lvls)): 'x' is NULL
## so the result will be NULL
```

```
print(map_utah, vp = grid::viewport(0.8, 0.3, width = 0.25, height = 0.3))
```



5. Create facet maps of countries in Eastern Africa:

- With one facet showing HDI and the other representing population growth (hint: using variables HDI and pop_growth, respectively)
- With a 'small multiple' per country

```
#class(africa)
names(africa)
```

```
## [1] "name"       "subregion"  "gdpPercap"  "HDI"        "pop_growth"
## [6] "geom"
```

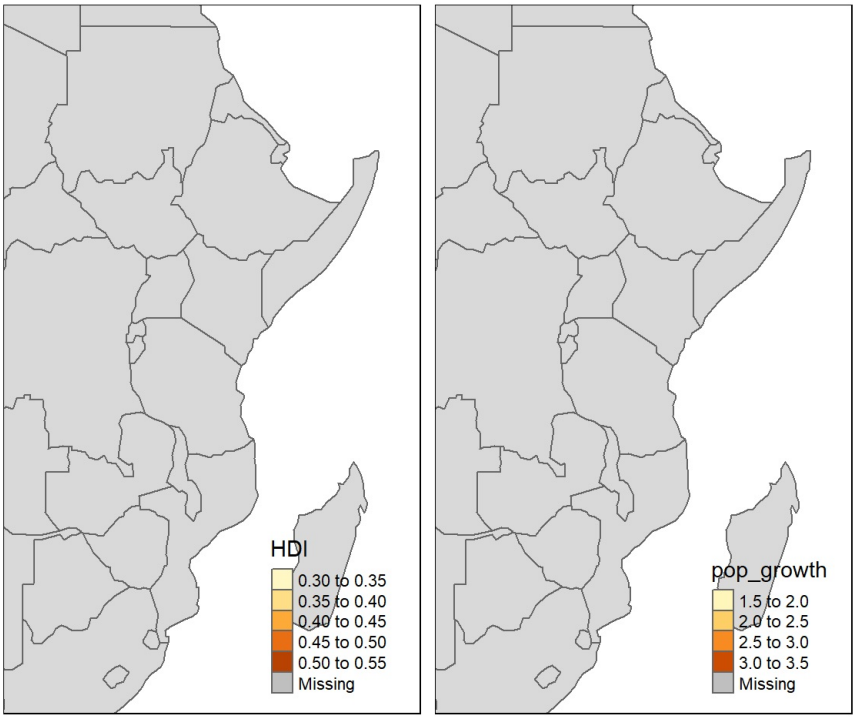
```
africa$subregion
```

```
##  [1] "Eastern Africa"  "Northern Africa" "Middle Africa"   "Eastern Africa"
##  [5] "Eastern Africa"  "Northern Africa" "Middle Africa"   "Southern Africa"
##  [9] "Southern Africa" "Eastern Africa"  "Southern Africa" "Southern Africa"
## [13] "Western Africa"  "Western Africa"  "Western Africa"  "Western Africa"
## [17] "Western Africa"  "Western Africa"  "Middle Africa"   "Western Africa"
## [21] "Western Africa"  "Western Africa"  "Western Africa"  "Western Africa"
## [25] "Western Africa"  "Western Africa"  "Western Africa"  "Middle Africa"
## [29] "Middle Africa"   "Middle Africa"   "Middle Africa"   "Eastern Africa"
## [33] "Eastern Africa"  "Eastern Africa"  "Southern Africa" "Middle Africa"
## [37] "Eastern Africa"  "Eastern Africa"  "Western Africa"  "Northern Africa"
## [41] "Northern Africa" "Eastern Africa"  "Northern Africa" "Northern Africa"
## [45] "Northern Africa" "Eastern Africa"  "Eastern Africa"  "Eastern Africa"
## [49] "Eastern Africa"  "Eastern Africa"
```

```
east_af <- africa %>%
     filter(subregion == "Eastern Africa")
east_af
```

```
## Simple feature collection with 15 features and 5 fields
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: -320102.4 ymin: -3085046 xmax: 2724372 ymax: 2090101
## CRS:            +proj=aea +lat_1=20 +lat_2=-23 +lat_0=0 +lon_0=25
## # A tibble: 15 x 6
##    name   subregion  gdpPercap    HDI pop_growth                          geom
##  * <chr>  <chr>          <dbl>  <dbl>      <dbl>            <MULTIPOLYGON [m]>
##  1 Tanza~ Eastern A~     2402.  0.47        3.11 (((922537.2 -114575.4, 939987.~
##  2 Somal~ Eastern A~       NA  NA          2.86 (((1717805 -205991.8, 1657143 ~
##  3 Kenya  Eastern A~     2753.  0.515       2.64 (((1468850 -559808.8, 1321049 ~
##  4 Zimba~ Eastern A~     1925.  0.387       2.34 (((634977.2 -2592580, 580490.2~
##  5 Zambia Eastern A~     3633.  0.443       3.04 (((592598.3 -989653, 635647.1 ~
##  6 Malawi Eastern A~     1090.  0.415       2.92 (((800730.1 -1095243, 901814.8~
##  7 Mozam~ Eastern A~     1080.  0.322       2.90 (((985449.9 -1364790, 1063050 ~
##  8 Burun~ Eastern A~      803.  0.352       2.99 (((566323.5 -287375.2, 572217.~
##  9 Madag~ Eastern A~     1372.  0.483       2.70 (((2528763 -1487023, 2555589 -~
## 10 Eritr~ Eastern A~       NA   0.346      NA    (((1193005 1686217, 1182131 17~
## 11 Ethio~ Eastern A~     1425.  0.392       2.58 (((2371505 933964.1, 2074454 5~
## 12 Djibo~ Eastern A~       NA   0.442       1.71 (((1809542 1466617, 1854109 14~
## 13 Uganda Eastern A~     1637.  0.454       3.35 (((922537.2 -114575.4, 711396 ~
## 14 Rwanda Eastern A~     1630.  0.429       2.50 (((561438.1 -135409.6, 602405.~
## 15 South~ Eastern A~     1936. NA           3.11 (((605775 415330.8, 514526.9 4~
```

```
map_eastaf = tm_shape(east_af) +
  tm_polygons(col = c("HDI", "pop_growth"))+
  qtm(africa)+
  tm_facets(nrow = 1, sync = TRUE)+
  tm_layout(legend.position = c("right", "bottom"), inner.margins = 0.1,
            legend.width = 40000)
  map_eastaf
```



6. Building on the previous facet map examples, create animated maps of East Africa:

- Showing first the spatial distribution of HDI scores then population growth
- Showing each country in order

```
map_eastafan = tm_shape(east_af) +
  tm_polygons(col = c("HDI", "pop_growth"))+
  qtm(africa)+
  tm_facets(nrow = 1, sync = TRUE)+
  tmap_mode("view")
```

```
## tmap mode set to interactive viewing
```

```
  map_eastafan
```

7. Create

an interactive map of Africa:

- With tmap
- With mapview
- With leaflet Bonus: For each approach, add a legend (if not automatically provided) and a scale bar

Usando tmap

```
map_africa = tm_shape(africa) +
  tm_polygons(col = "pop_growth")+
  qtm(africa)+
  tm_facets(nrow = 1, sync = TRUE)+
  tmap_mode("view")
```

```
## tmap mode set to interactive viewing
```

```
map_africa
```

Usando mapview

Profe,este mapa lo pude generar, pero al convertirlo en html me da el error que coloque en comentarios en el chunk

```
#mapview(africa, zcol = "pop_growth", legend=TRUE)

#ERROR
#List of 10
 #$ name      : chr "PopupTable"
 #$ version   : chr "0.0.1"
 #$ src       :List of 1
  #..$ file: chr ""
# $ meta      : NULL
 #$ script    : NULL
 #$ stylesheet: chr "popup.css"
 #$ head      : NULL
# $ attachment: NULL
# $ package   : NULL
# $ all_files : logi TRUE
 #- attr(*, "class")= chr "html_dependency"
#Error: path for html_dependency not found:
#Ejecución interrumpida
```

Usando leafet : genera gráficos interactivos, se coloca en teclas numerales para poder subirlo a git

```
#pal = colorNumeric("RdYlBu", domain = africa$pop_growth)
#leaflet(data = africa) %>%
  #addProviderTiles(providers$OpenTopoMap) %>%
 # addPolygons(fillColor = ~pal(pop_growth), fillOpacity = .8) %>%
  #addLegend(pal = pal, values = ~pop_growth)
```