

**Universidad Rey Juan Carlos**

GRADO EN MATEMÁTICAS

# PRÁCTICA 10

*Geometría Computacional*

Autores: Guillermo Grande Santi y  
Alejandro López Adrados

Abril, 2022

# Índice

<b>1</b>	<b>Objetivos</b>	<b>1</b>
<b>2</b>	<b>Metodología</b>	<b>1</b>
<b>3</b>	<b>Conclusiones</b>	<b>1</b>
<b>4</b>	<b>Anexos</b>	<b>1</b>
4.1	Código del ejercicio . . . . .	1

## 1 Objetivos

El objetivo de esta práctica es generar aleatoriamente un conjunto de datos, aplicar los conocimientos de clusterización y diagramas de Voronoi trabajados en clase.

## 2 Metodología

Podemos observar que, en primer lugar, se importan las librerías necesarias para el correcto funcionamiento del código (`deldir`, `RColorBrewer`, `tripack`) y se escriben los valores de los vectores `x` e `y`.

Entonces se aplicará la función `deldir()`, a partir de la que se extraerán los centroides de la triangulación. En este punto, se ejemplificará con un diagrama de Voronoi. Para ello se generarán puntos aleatorios distribuidos entre los intervalos de cada 0,2 décimas, y se usará la función `kmeans` para realizar el proceso de clustering, cuyo resultado es que no converge en 10 iteraciones.

Por último, se definen las celdas de Voronoi y se dibuja todo en conjunto, obteniendo el tercer gráfico como resultado del programa, donde se ven cinco celdas diferenciadas por colores.

## 3 Conclusiones

Podemos ver en esta práctica ejemplos gráficos de los diagramas de Voronoi estudiados en la parte teórica de la asignatura, algo que siempre es interesante. Cabe resaltar que el código de esta práctica está extraído de los apuntes de clase, pero esto no ha impedido el entendimiento del código.

## 4 Anexos

### 4.1 Código del ejercicio

```

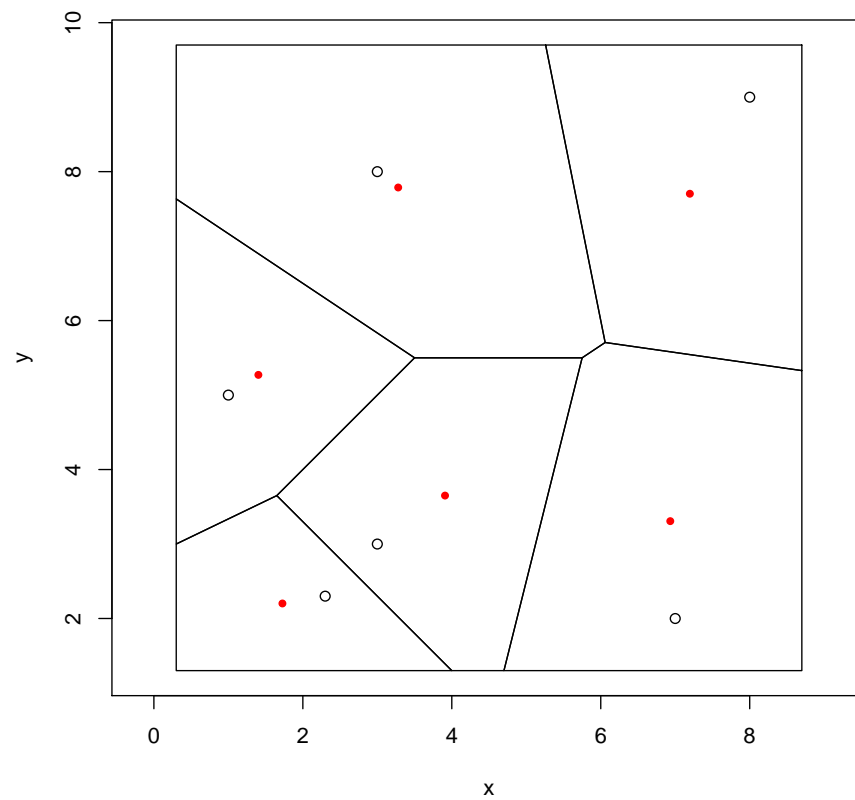
# Introducimos los datos
# dpl estructura de los dummy nodes para la teselacion
# rw coordenadas del recuadro de la teselacion
# Se instalan las librerías deldir, tripack, RColorBrewer
library(deldir)

## deldir 0.2-10      Nickname: "Morpheus and Euripides"
##
##      Note 1: As of version 0.2-1, error handling in this
##      package was amended to conform to the usual R protocol.
##      The deldir() function now actually throws an error
##      when one occurs, rather than displaying an error number
##      and returning a NULL.
##
##      Note 2: As of version 0.1-29 the arguments "col"
##      and "lty" of plot.deldir() had their names changed to
##      "cmpnt_col" and "cmpnt_lty" respectively basically
##      to allow "col" and "lty" to be passed as "..."
##      arguments.
##
##      Note 3: As of version 0.1-29 the "plotit" argument
##      of deldir() was changed to (simply) "plot".
##
##      See the help for deldir() and plot.deldir().

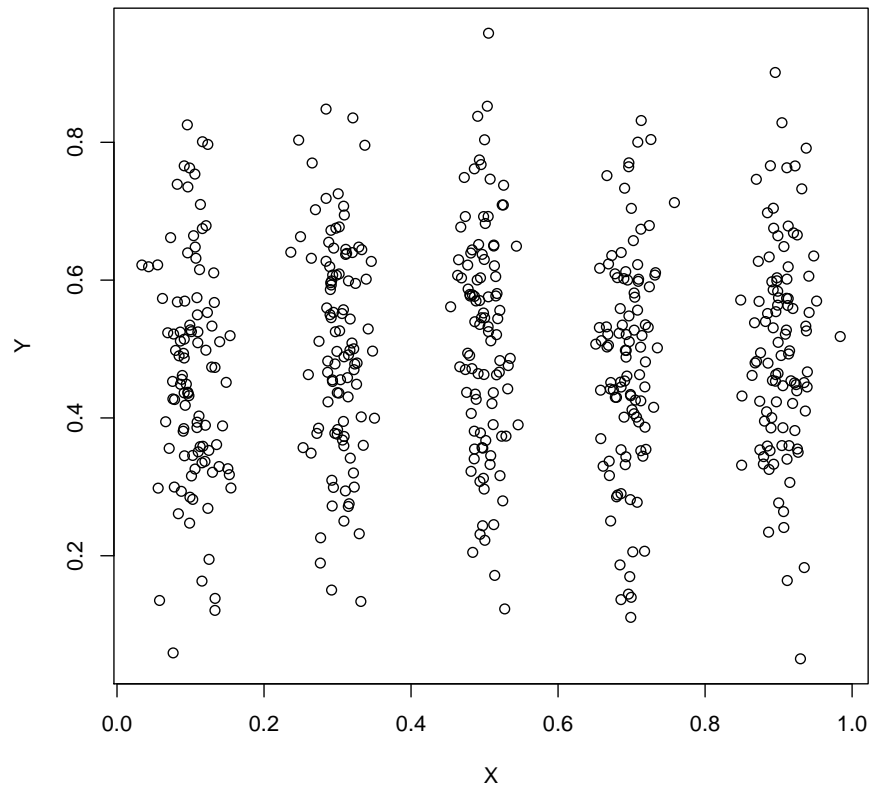
x <- c(2.3,3.0,7.0,1.0,3.0,8.0)
y <- c(2.3,3.0,2.0,5.0,8.0,9.0)
dxy1 <- deldir(x,y,dpl=NULL, rw=NULL, plotit=TRUE)
# Centroides de la triangulacion
l<-tile.list(dxy1)
g<-tile.centroids(l)
plot(l,close=TRUE)

```

```
points(g,pch=20,col="red")
```



```
# Ejemplo con Voronoi  
set.seed(1)  
pts <- cbind(X=rnorm(500,rep(seq(1,9,by=2)/10,100),.022),Y=rnorm(500,.5,.15))  
plot(pts)
```



```
km1 <- kmeans(pts, centers=5, nstart = 1, algorithm = "Lloyd")

## Warning: did not converge in 10 iterations

#library(tripack)
library(RColorBrewer)
CL5 <- brewer.pal(5, "Pastel1")
#V <- voronoi.mosaic(km1$centers[,1],km1$centers[,2])
#P <- voronoi.polygons(V)
#plot(pts,pch=19,xlim=0:1,ylim=0:1,xlab="",ylab="",col=CL5[km1$cluster])
#points(km1$centers[,1],km1$centers[,2],pch=3,cex=1.5,lwd=2)
#plot(V,add=TRUE)
```

