

Universidad de San Andrés
Departamento de Matemática y Ciencias
Maestría en Ciencia de Datos

TP Final
Asignatura: Estadística Espacial

true true true

14-04-2023

Contents

| | |
|-------------------------|----------|
| Problemática | 1 |
| Librerías | 2 |
| Metodología | 2 |
| Moran - Geary | 8 |
| Variograma | 15 |

Problemática

Para este trabajo, contamos con la base de datos “meuse” de la librería “sp” de R como objeto de análisis. En particular, nos encontramos con datos sobre las locaciones de grandes concentraciones de metales pesados, más precisamente zinc, en la capa superficial del suelo de la llanura del río Meuse. Es por eso que, valiéndonos de los recursos vistos en el curso, decidimos plantear un análisis sistemático que abarque desde una estadística descriptiva, seguido por estudios de estacionariedad, isotropía, análisis estructural y predicción.

Librerías

```
library('geoR')
library('spdep')
library('gstat')
library("leafsync")
library('mapview')
library('leaflet')
library('RColorBrewer')
library('ggplot2')
library('ggmap')
library('tibble')
library('caret')
library('sf')
library('sp')
library('PerformanceAnalytics')
library("rgdal")
library('lattice')
library('grDevices')
library('GGally')
library('dplyr')
library('geostan')
```

Metodología

En un primer lugar, realizamos la carga de los datos y generamos una copia de seguridad de la misma en caso de que necesitemos modificarla para algún análisis particular.

```
data(meuse)
copia_seguridad <- meuse
```

Comenzamos el análisis exploratorio de la base en cuestión. Vemos que consta de 155 registros, y que ninguno cuenta con valores nulos o faltantes.

```
# Vemos la cantidad de registros y columnas
class(meuse[,c('x','y','zinc')]) # Es de clase "data frame"

## [1] "data.frame"

dim(meuse[,c('x','y','zinc')])

## [1] 155    3

# Analizamos duplicados y valores nulos
sum(duplicated(meuse[,c('x','y','zinc')])) # No tenemos registros duplicados

## [1] 0

sum(is.na(meuse[,c('x','y','zinc')])) # No tenemos valores NA.

## [1] 0

# Vemos algunos datos de la base
head(meuse[,c('x','y','zinc')])

##           x           y zinc
## 1 181072 333611 1022
## 2 181025 333558 1141
## 3 181165 333537  640
## 4 181298 333484  257
## 5 181307 333330  269
## 6 181390 333260  281
```

Para entender más la base, hicimos una estadística descriptiva encontrando que existen valores atípicos de zinc, ya que se alcanzan niveles de concentración de 1839 ppm en una muestra donde el tercer cuartil se alcanza a las 674.5 ppm.

```
# Vemos la estadística descriptiva
```

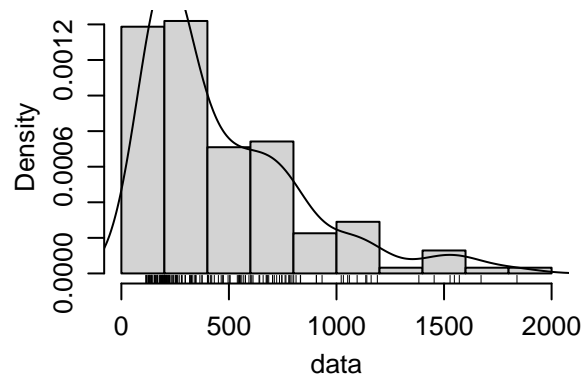
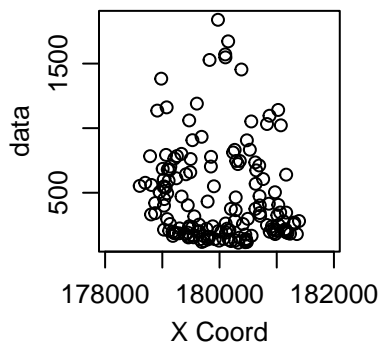
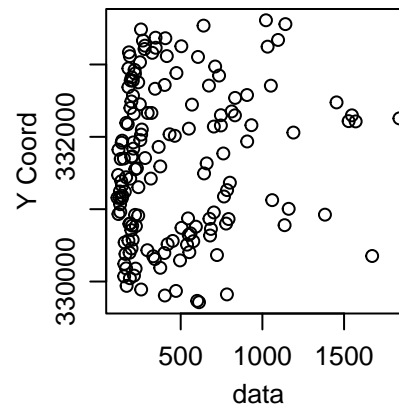
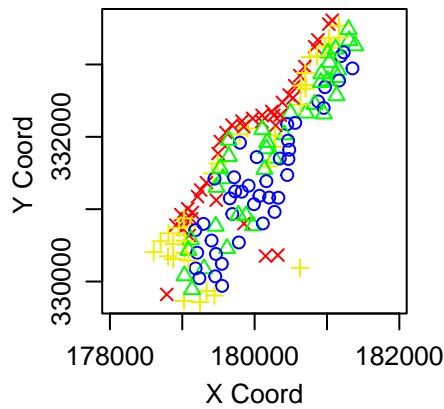
```
summary(meuse[,c('x','y','zinc')]) # Tenemos valores máximos que superan ampliamente la
```

```
##           x           y           zinc
## Min.      :178605   Min.      :329714   Min.      : 113.0
## 1st Qu.:179371   1st Qu.:330762   1st Qu.: 198.0
## Median :179991   Median :331633   Median : 326.0
## Mean    :180005   Mean    :331635   Mean    : 469.7
## 3rd Qu.:180630   3rd Qu.:332463   3rd Qu.: 674.5
## Max.    :181390   Max.    :333611   Max.    :1839.0
```

```
# Sospechamos entonces que habrán eventuales datos atípicos
```

Además, estudiamos la distribución de la concentración del metal y su relación con las coordenadas registradas en la siguiente salida.

```
x<-meuse$x
y<-meuse$y
zinc<-meuse$zinc
data <- cbind(x,y,zinc)
datag <- as.geodata(data) # Lo llevamos a tipo "geo-data"
plot(datag)
```



```
#data(meuse)
#coordinates(meuse) <- ~x+y
#proj4string(meuse) <- CRS("+init=epsg:28992")

#mapview(meuse, zcol = c("zinc"), legend = TRUE)
```

Aquí podemos observar que todos los puntos se encuentran dispersos a lo largo del río, teniendo ciertos lugares donde los niveles de concentración del metal parecen agruparse. Por ejemplo, para el caso de los puntos rojos, notamos que estos son los valores que reflejan el mayor nivel de concentración de zinc, y que están agrupados muy próximos al río; mientras que los puntos que con menores niveles de concentración se hallan más dispersos y a una distancia mayor en relación al río.

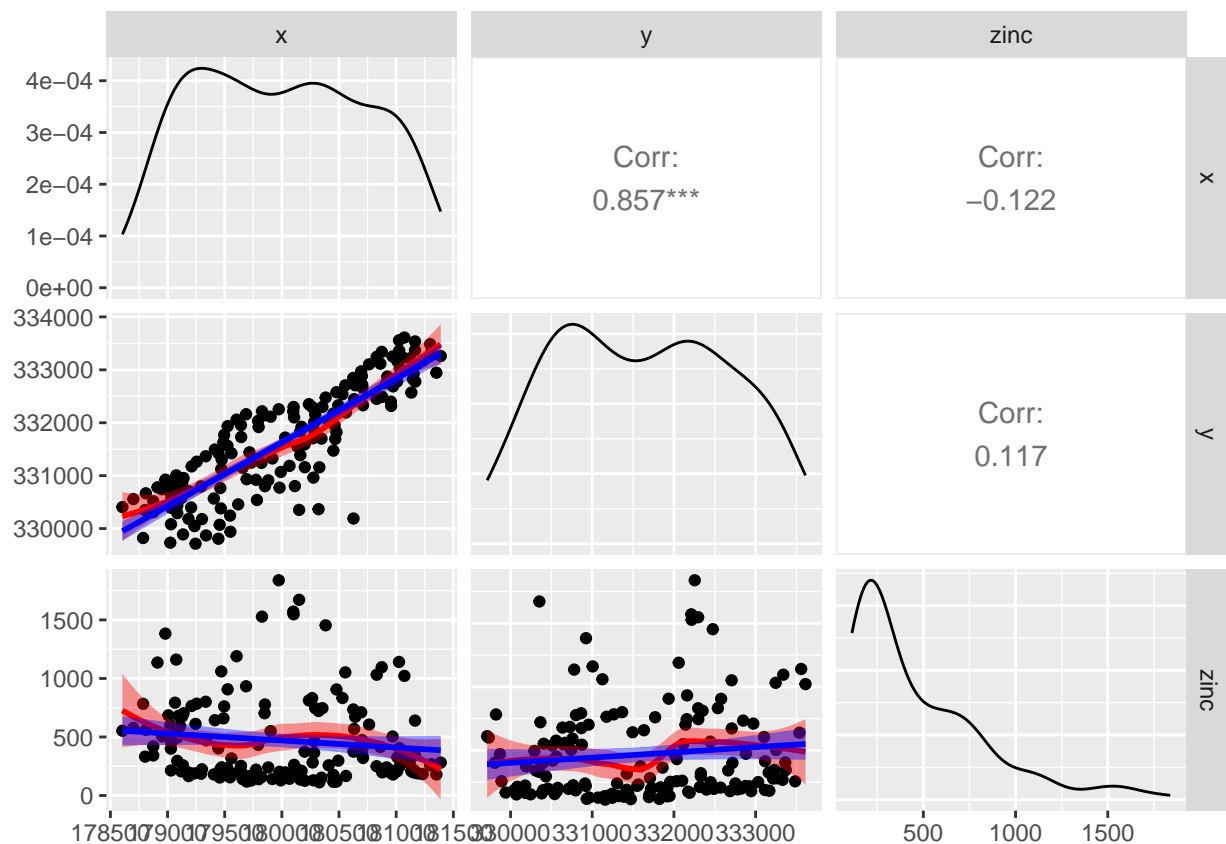
Haciendo referencia a la relación entre la variable con cada coordenada, podemos notar una muy leve correlación entre los niveles de concentración por zinc con las coordenada x e y, tratándose en ambos casos de una magnitud muy chica y sin significancia estadística para los

niveles usualmente utilizados.

```
data(meuse)

# Definimos una función para ver las correlaciones en conjunto con los plots
my_fn <- function(data, mapping, ...){
  p <- ggplot(data = data, mapping = mapping) +
    geom_point() +
    geom_smooth(method=loess, fill="red", color="red", ...) +
    geom_smooth(method=lm, fill="blue", color="blue", ...)
  p
}

g = ggpairs(meuse[,c('x','y','zinc')], columns = 1:3, lower = list(continuous = my_fn))
g
```



Respecto a la distribución del zinc, para alcanzar una mayor profundidad, hicimos el siguiente

test de Shapiro-Wilk para evaluar su normalidad de la distribución con mayor rigurosidad.

$$H_0 = \text{Datos} \sim N(\mu, \sigma) \quad vs \quad H_1 = \text{Datos} \not\sim N(\mu, \sigma)$$

Como resultado, obtuvimos un p-valor de 3.28e-12, por lo cual rechazamos la hipótesis nula de normalidad. También hicimos el ejercicio transformando la variable con logaritmo, y obtuvimos un resultado muy similar a favor de la no normalidad de los datos. Esto también se evidencia si comparamos los cuantiles de una normal con la distribución de los datos en un QQ-Plot. Luego, como tanto los test de Moran como el de Geary presuponen una distribución normal de los datos, tendremos que hacer ambos tests considerando sus versiones aleatorizadas.

```
shapiro.test(meuse$zinc)
```

```
##
```

```
##  Shapiro-Wilk normality test
```

```
##
```

```
## data:  meuse$zinc
```

```
## W = 0.82812, p-value = 3.28e-12
```

```
# Vemos como parece quedar con una distribución más cercana a la normal si lo llevamos
```

```
meuse$lnZn <- log(meuse$zinc, base = exp(1))
```

```
shapiro.test(meuse$lnZn)
```

```
##
```

```
##  Shapiro-Wilk normality test
```

```
##
```

```
## data:  meuse$lnZn
```

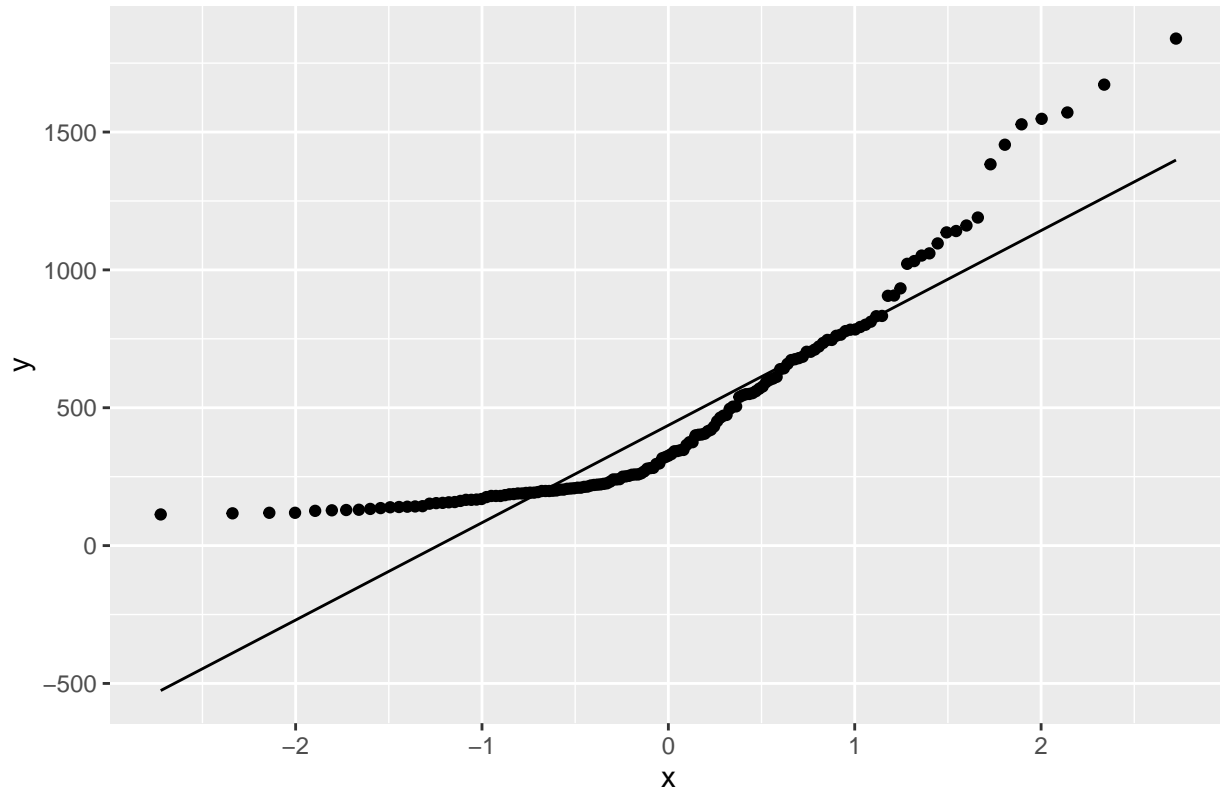
```
## W = 0.95092, p-value = 2.901e-05
```

```
# Igual rechazamos normalidad
```

```
df = as.data.frame(zinc)
```

```
ggplot(df, aes(sample=zinc)) + stat_qq() + stat_qq_line() + labs(title = 'QQ Plot de la
```

QQ Plot de la distribución del Zinc



Moran - Geary

Moral: Análisis global y local

Para continuar con el análisis geoestadístico, apelamos a los índices y tests vistos en clase, analizando así si es que existe -o no- una autocorrelación espacial.

En la primera parte, para poder calcular ambos índices para determinar la autocorrelación de nuestros datos, debemos definir los pesos para cada una de las observaciones para así definir una grilla la cual nos determine el vecindario y los vecinos de cada punto.

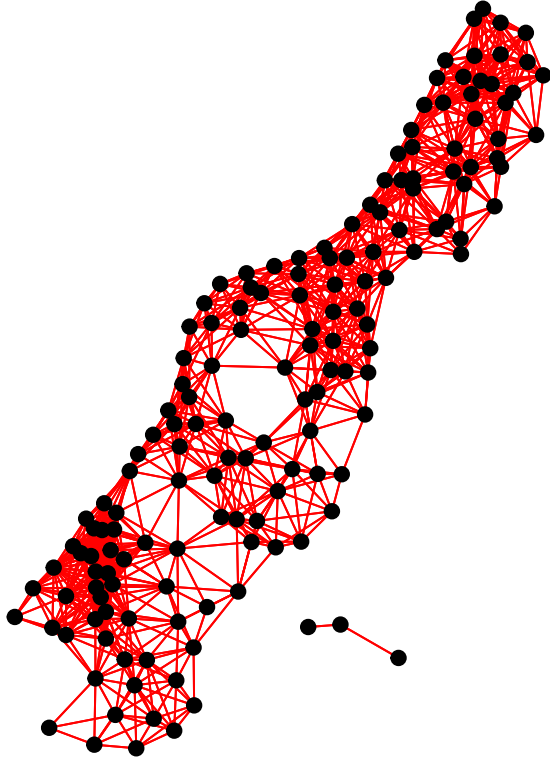
```
data(meuse)
```

```
coordinates(meuse) <- c("x", "y")
```

```
coordenadas <- coordinates(meuse)
```



```
grilla <- dnearneigh(meuse,0,400)
pesos <- nb2listw(grilla, style = "W")
plot(grilla, coordenadas, col = "red", pch = 19, cex = 1)
```



Para evaluar la autocorrelación espacial en mayor profundidad, vamos a calcular los índices local y global de Moran como también el de Geary. Así buscaremos comprender la variación del fenómeno, en este caso del zinc en el río Meuse, en un marco geográfico de análisis. Los posibles resultados son:

- Notar que hay evidencia a favor de la existencia de autocorrelación positiva. En este caso deberíamos notar que el zinc se agrupa en zonas uniformes conformando de esta manera una especie de cluster.
- Hallar evidencia de autocorrelación negativa, notando que el zinc se encuentra disperso, es decir que la presencia de zinc sea disímil en sus lugares aledaños/vecinos.
- No encontrar evidencia de autocorrelación espacial, lo que nos diría que la variable tiene un comportamiento aleatorio en el estudio del fenómeno en el terreno.

En una primera instancia calculamos el Índice de Moran Global mediante la ejecución del test de Moran Global en su versión aleatorizada. Estos, independientemente de la opción que elijamos para calcular los pesos, nos arroja resultados que nos permiten llegar a las mismas conclusiones:

- Obtenemos un p-valor menor a los niveles estadísticos usualmente utilizados, permitiendo concluir que hay evidencia suficiente para rechazar la hipótesis nula de que no existe autocorrelación espacial.
- Se obtiene un valor de índice de Moran (IM) superior a su valor esperado, lo cual reafirma la conclusión de que existe autocorrelación espacial.

```
moran.test(meuse$zinc, nb2listw(grilla, style = "W"), randomisation = TRUE)
```

```
##
```

```
## Moran I test under randomisation
```

```
##
```

```
## data: meuse$zinc
```

```
## weights: nb2listw(grilla, style = "W")
```

```
##
```

```
## Moran I statistic standard deviate = 10.188, p-value < 2.2e-16
```

```
## alternative hypothesis: greater
```

```
## sample estimates:
```

| ## Moran I statistic | Expectation | Variance |
|----------------------|--------------|-------------|
| ## 0.320600991 | -0.006493506 | 0.001030686 |

```
#moran.test(meuse$zinc, nb2listw(grilla, style = "S"), randomisation = TRUE)
```

```
#moran.test(meuse$zinc, nb2listw(grilla, style = "B"), randomisation = TRUE)
```

```
#moran.test(meuse$zinc, nb2listw(grilla, style = "C"), randomisation = TRUE)
```

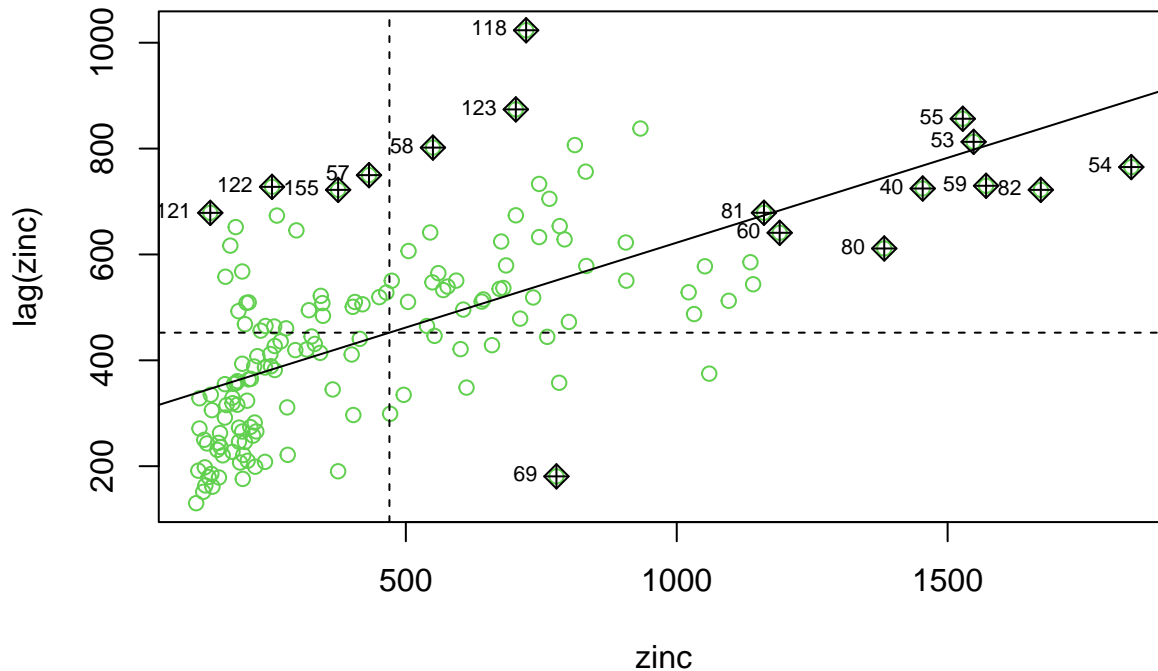
```
#moran.test(meuse$zinc, nb2listw(grilla, style = "U"), randomisation = TRUE)
```

```
#moran.test(meuse$zinc, nb2listw(grilla, style = "minmax"), randomisation = TRUE)
```

De manera grafica, también podemos evaluar cuan similar es cada dato con respecto a sus vecinos. A través de este índice podemos encontrar datos atípicos, es decir posibles outliers

espaciales.

```
M <- moran.plot(meuse$zinc,pesos,zero.policy=F,col=3, quiet=TRUE,labels=T,xlab = "zinc",
```



Encontramos luego que hay eventuales candidatos a outliers espaciales, como son por ejemplo los registros 69 y 118, los cuales muestran un valor muy distinto a sus observaciones cercanas. Además, si vemos las observaciones que tienen un Índice de Moran Local (IML) muy bajo a nivel observación, nos encontramos con las observaciones que sospechamos outliers de manera gráfica muestran efectivamente un IML bajo:

```
ML <- localmoran(meuse$zinc, pesos, alternative ="less")
IML <-data.frame(ML,row.names=elevation$Casos)

#Potenciales outliers
potenciales_outliers = subset(IML, IML$Ii < 0 & IML$Pr.z...E.Ii.. < 0.05)
potenciales_outliers
```

```
##           Ii           E.Ii          Var.Ii          Z.Ii Pr.z...E.Ii..
## 41 -0.22547290 -0.0014302294 0.0109317773 -2.142818 0.0160638455
## 43 -0.32244456 -0.0041844569 0.0299942797 -1.837651 0.0330569422
```

```
## 57 -0.07890771 -0.0000689982 0.0006476965 -3.097803 0.0009748063
## 69 -0.66536542 -0.0046098322 0.0748935167 -2.414453 0.0078794310
## 98 -0.38567470 -0.0039043753 0.0263977957 -2.349732 0.0093934649
## 121 -0.51601760 -0.0053051084 0.0495390824 -2.294576 0.0108787093
## 122 -0.41767062 -0.0022780643 0.0299336116 -2.400928 0.0081767750
## 125 -0.31610960 -0.0020927818 0.0141751997 -2.637475 0.0041762915
```

```
# Nos quedamos con la base sin outliers
```

```
base_sin_outliers = copia_seguridad[!(rownames(copia_seguridad) %in% rownames(potencial
```

```
coordinates(base_sin_outliers) <- c("x", "y")
```

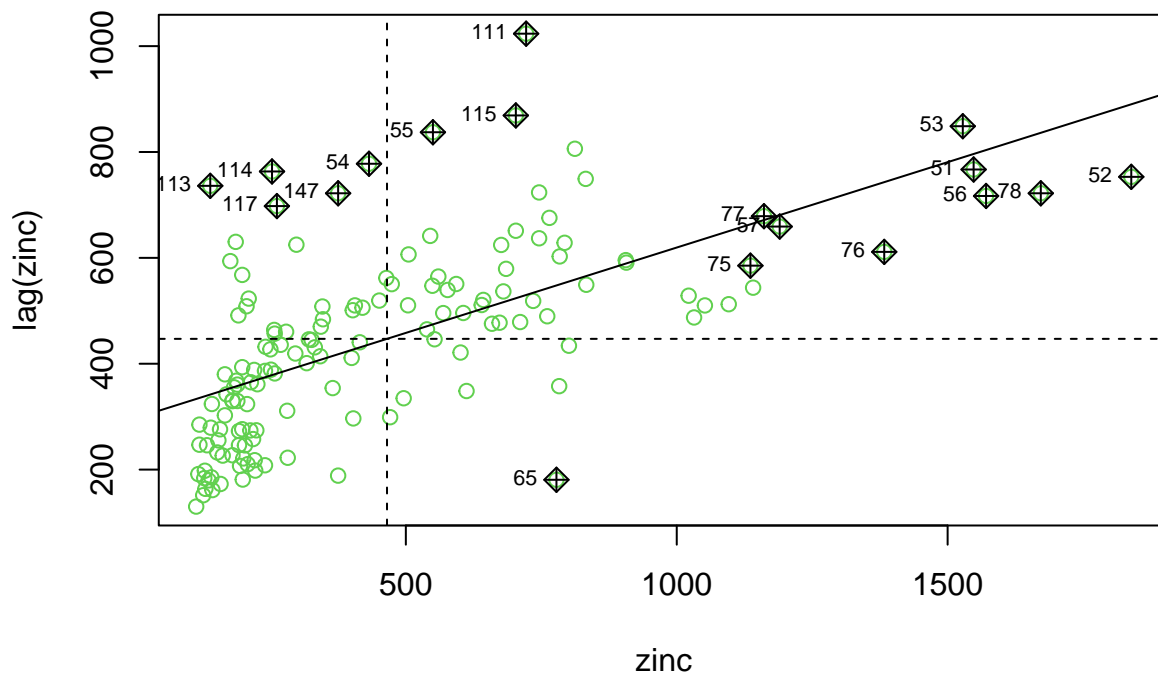
```
coordenadas_sin_outliers <- coordinates(base_sin_outliers)
```

```
grilla_sin_outliers <- dnearneigh(base_sin_outliers,0,400)
```

```
pesos_sin_outliers <- nb2listw(grilla_sin_outliers, style = "W")
```

Si volvemos a hacer el gráfico de Moran, de hecho nos cambia la pendiente, puesto que es un método que es sensible a outliers.

```
M_sin_outliers <- moran.plot(base_sin_outliers$zinc,pesos_sin_outliers,zero.policy=F,col
```



```

moran.test(meuse$zinc, nb2listw(grilla, style = "W"), randomisation = TRUE) # Test con o

##
## Moran I test under randomisation
##
## data: meuse$zinc
## weights: nb2listw(grilla, style = "W")
##
## Moran I statistic standard deviate = 10.188, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      0.320600991      -0.006493506      0.001030686

moran.test(base_sin_outliers$zinc, nb2listw(grilla_sin_outliers, style = "W"), randomisa

##
## Moran I test under randomisation
##
## data: base_sin_outliers$zinc
## weights: nb2listw(grilla_sin_outliers, style = "W")
##
## Moran I statistic standard deviate = 9.7328, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      0.322545398      -0.006849315      0.001145406

```

Geary

Con este índice vamos a medir la asociación espacial a través del cambio y de su varianza. De esta manera evaluaremos si la distribución del zinc en el río Meuse es aleatoria o no,

afirmando que hay una alta autocorrelación espacial si la diferencia de los valores del zinc en diferentes lugares son similares y la distancia entre los lugares es pequeña.

Este índice es más sensible a outliers que el índice de Moran. Toma valores que van entre 0 y 2, donde:

- 1 indica una distribución espacial aleatoria de la variable.
- valores menores que 1 indican una autocorrelación positiva, siendo los valores similares cercanos unos a otros.
- valores mayores que 1 indican una autocorrelación negativa, siendo los valores diferentes cerca unos de otros.

En esta segunda instancia calculamos el Índice de Geary mediante la ejecución del test de Geary en su versión aleatorizada. Estos, independientemente de la opción que elijamos para calcular los pesos, nos arroja resultados que nos permiten llegar a conclusiones similares:

- Obtenemos un p-valor menor a los niveles estadísticos usualmente utilizados, permitiendo concluir que hay evidencia suficiente para rechazar la hipótesis nula de que no existe autocorrelación espacial.
- Se obtiene un valor de índice de Geary (IG) superior a su valor esperado, lo cual reafirma la conclusión de que existe autocorrelación espacial.

```
geary.test(meuse$zinc, nb2listw(grilla, style = "W"), randomisation = TRUE)
```

```
##
```

```
## Geary C test under randomisation
```

```
##
```

```
## data: meuse$zinc
```

```
## weights: nb2listw(grilla, style = "W")
```

```
##
```

```
## Geary C statistic standard deviate = 9.9183, p-value < 2.2e-16
```

```
## alternative hypothesis: Expectation greater than statistic
```

```
## sample estimates:
```

| ## Geary C statistic | Expectation | Variance |
|----------------------|-------------|-------------|
| ## 0.633355716 | 1.000000000 | 0.001366521 |

```
#geary.test(meuse$zinc, nb2listw(grilla, style = "S"), randomisation = TRUE)
#geary.test(meuse$zinc, nb2listw(grilla, style = "B"), randomisation = TRUE)
#geary.test(meuse$zinc, nb2listw(grilla, style = "C"), randomisation = TRUE)
#geary.test(meuse$zinc, nb2listw(grilla, style = "U"), randomisation = TRUE)
#geary.test(meuse$zinc, nb2listw(grilla, style = "minmax"), randomisation = TRUE)
```

En base a los resultados obtenidos nos indica que en sitios conectados los valores del zinc son similares. Además, obtenemos los mismos resultados en caso de purgar los valores que sospechamos outliers previamente.

```
geary.test(base_sin_outliers$zinc, nb2listw(grilla_sin_outliers, style = "W"), randomisa

##
## Geary C test under randomisation
##
## data: base_sin_outliers$zinc
## weights: nb2listw(grilla_sin_outliers, style = "W")
##
## Geary C statistic standard deviate = 9.4415, p-value < 2.2e-16
## alternative hypothesis: Expectation greater than statistic
## sample estimates:
## Geary C statistic      Expectation      Variance
## 0.626140212          1.000000000          0.001567976
```

Al arrojar valores menores a 1, podemos afirmar que hay autocorrelación positiva, es decir que los valores similares tienden a estar cerca unos de otros.

Variograma

El variograma es una herramienta para el análisis geoestadístico utilizada para describir la variabilidad espacial de un fenómeno, en este caso el zinc. Hay tres tipos de variogramas que

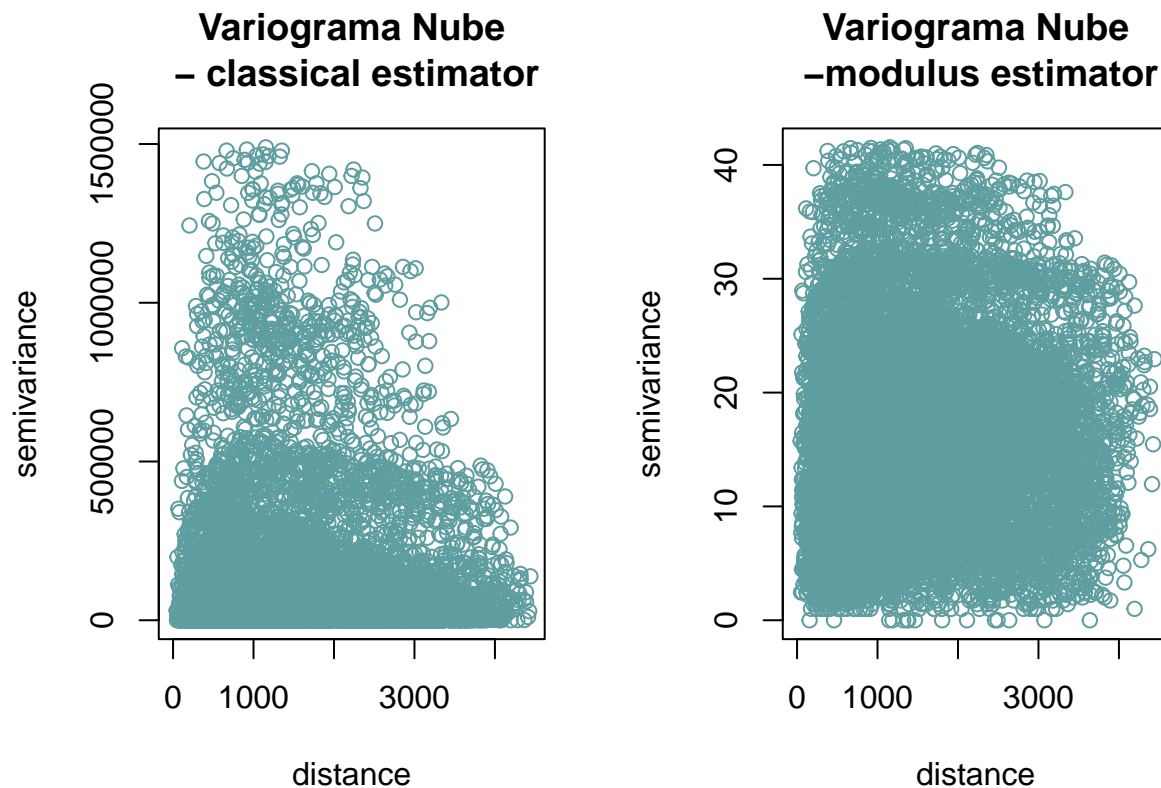
se utilizaran para el análisis:

- **Variograma nube:** podemos graficar la distancia de los puntos en el espacio versus la variable de cambio (semi varianza al cuadrado).
- **Variograma empírico o experimental:** Como con el variograma nube no alcanza vamos a calcular el variograma empírico, el cual va a ser construido a partir del variograma nube, dividiendo el eje de distancias en intervalos, tomando un representante sobre el cual promedió con todos los puntos que caen dentro del intervalo, consiguiendo así la representación del mismo.
- **Variograma teórico:** este será calculado dado que necesitamos de una línea continúa para poder estimar las observaciones que no fueron representadas.

Tras haber realizado tanto los cálculos de los índices como los análisis de autocorrelación y la evaluación de potenciales outliers, procedemos con el **análisis estructural**. Para esto construiremos el **variograma nube**, tomando el dataset sin las observaciones que consideramos como outliers.

A continuación mostraremos cómo es la visualización tanto del variograma nube como del empírico.

```
par(mfrow=c(1,2))
plot(nube_clasica, col = 'cadetblue', main = "Variograma Nube\n - classical estimator")
plot(nube_CH,col = 'cadetblue', main = "Variograma Nube\n -modulus estimator")
```

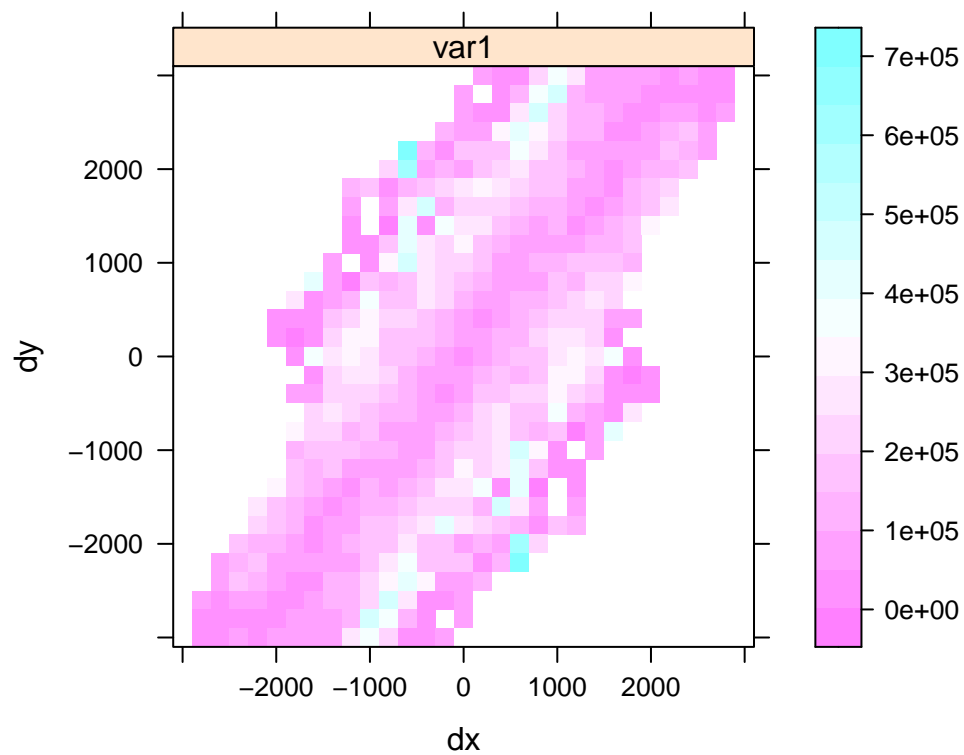



```
par(mfrow = c(1,1))
```

Lo primero que podemos observar es que no parece haber una tendencia clara. Esto era de esperarse luego de lo observado en el análisis exploratorio previo, donde las correlaciones entre las coordenadas y los niveles de concentración de zinc eran débiles.

Luego, para analizar la isotropía y saber si es correcto que nuestro variograma sea omnidireccional, realizamos el variograma mapa.

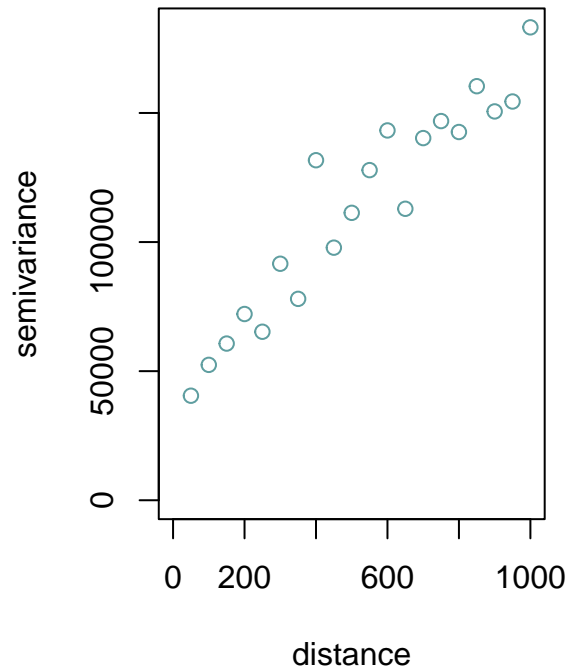
```
vv <- variogram(zinc~1, base_sin_outliers, cutoff = 3000, width = 200, map=T)
plot(vv)
```



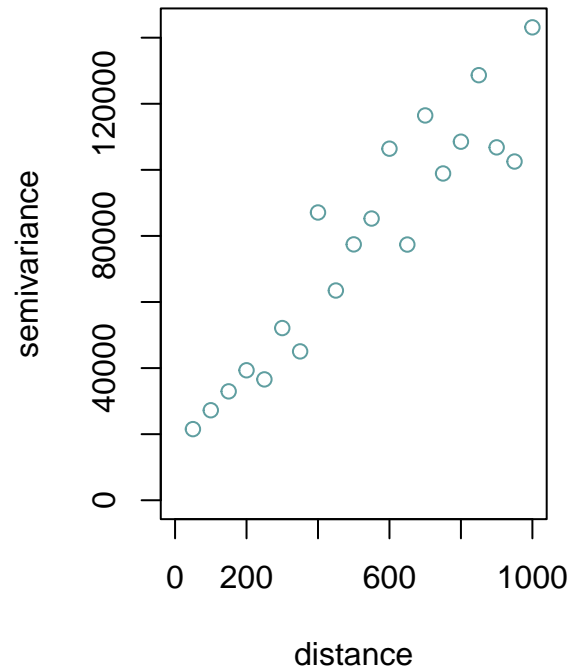
Parece entonces un proceso Anisotrópico, con dirección por continuidad, alcanzando otros valores si corremos la dirección perpendicular sentido de la mancha.

```
par(mfrow=c(1,2))
plot(bin_clasico,col = 'cadetblue', main = "Variograma Empírico\n- classical estimator")
plot(bin_CH,col = 'cadetblue', main = "Variograma Empírico\n- modulus estimator")
```

**Variograma Empírico
– classical estimator**



**Variograma Empírico
– modulus estimator**



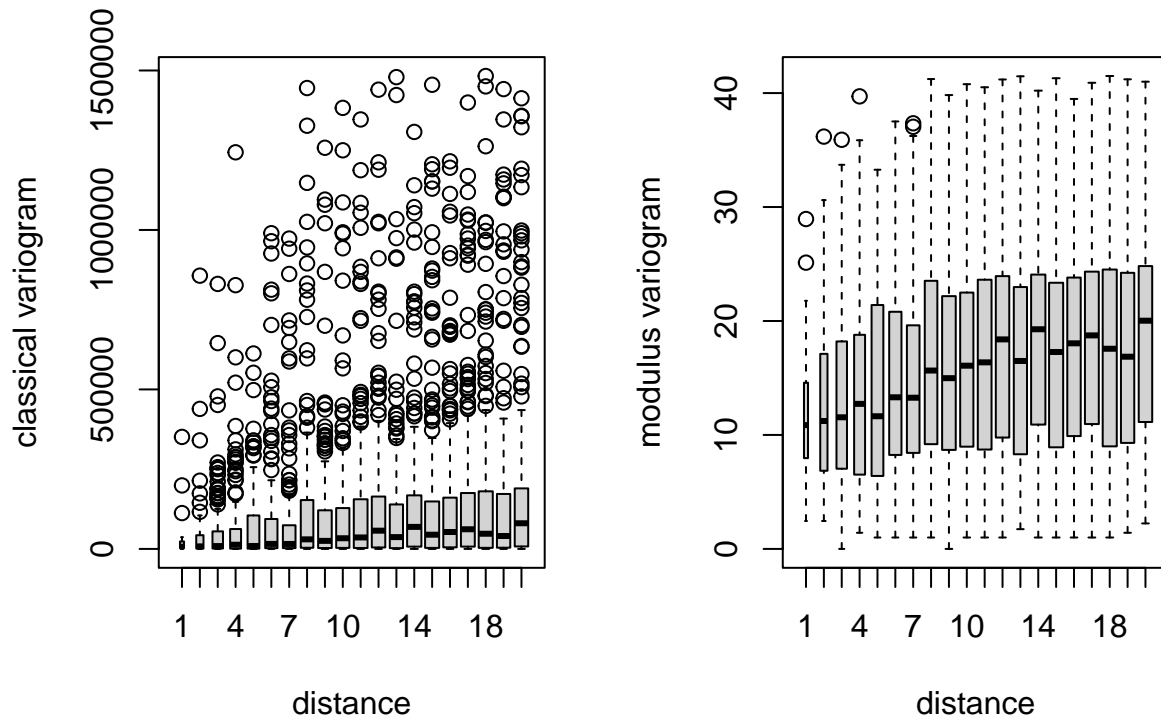
```
par(mfrow = c(1,1))
```

asdsaasdads

```
par(mfrow = c(1,2))
```

```
plot(bin1, bin.cloud = T, main = "classical estimator")
```

```
plot(bin2, bin.cloud = T, main = "modulus estimator")
```



```
par(mfrow = c(1,2))
```

Como podemos observar a través de estos gráficos podríamos observar un ligero comportamiento **anisotrópico** en la variable, dado que para algunos puntos el comportamiento del zinc presenta algunas variaciones en ciertas direcciones, pero las mismas son muy pequeñas.

En este caso queremos entender si estamos frente a una variable **anisotrópica** o **isotrópica**, es decir queremos terminar de confirmar si el comportamiento estadístico varía según la dirección en la que se mida la distancia, es por eso que calculamos el variograma variando las direcciones, de esta manera queremos entender la variabilidad espacial del zinc en diferentes direcciones. Para poder calcularlo vamos a tomar un ángulo y distancia determinados

```
vario.2 <- variog(meuse, coords = coordinates(meuse), data = meuse$zinc, uvec = seq(0, 1
```

```
## variog: computing variogram for direction = 0 degrees (0 radians)
```

```
##      tolerance angle = 22.5 degrees (0.393 radians)
```

```
vario.3 <- variog(meuse, coords = coordinates(meuse), data = meuse$zinc, uvec = seq(0, 1
```

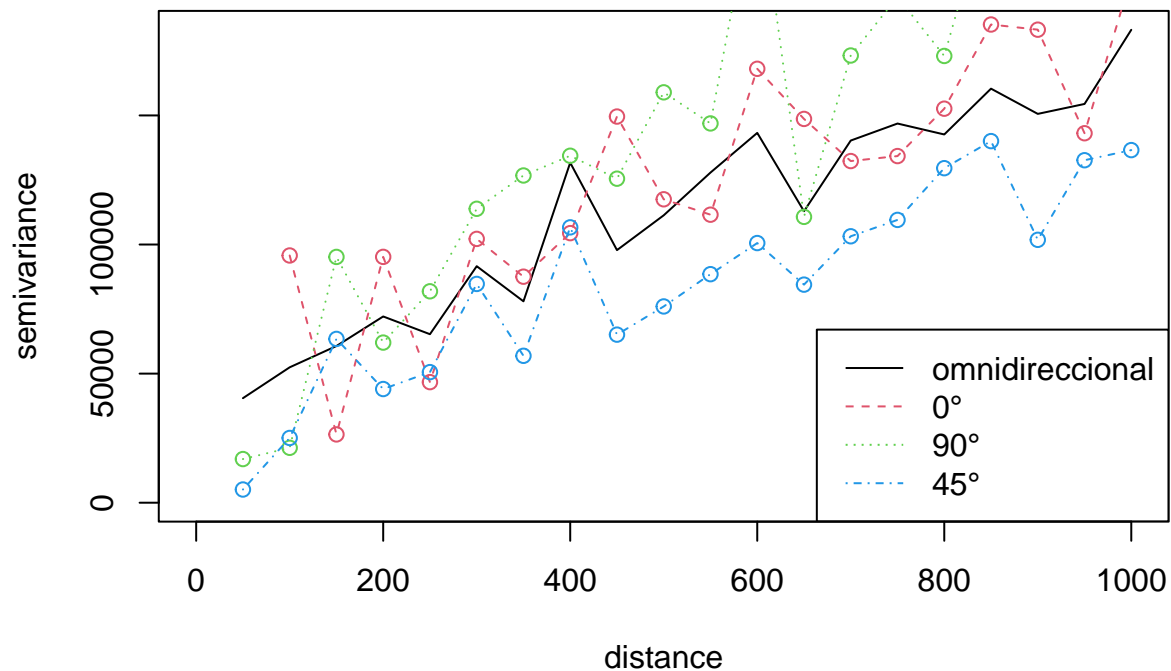
```
## variog: computing variogram for direction = 90 degrees (1.571 radians)
```

```
##          tolerance angle = 22.5 degrees (0.393 radians)

vario.4 <- variog(meuse, coords = coordinates(meuse), data = meuse$zinc, uvec = seq(0, 1

## variog: computing variogram for direction = 45 degrees (0.785 radians)
##          tolerance angle = 22.5 degrees (0.393 radians)

par(mfrow = c(1,1))
plot(bin_clasico, type="l")
lines(vario.2, lty = 2, col = 2)
lines(vario.3, lty = 3, col = 3)
lines(vario.4, lty = 4, col = 4)
legend("bottomright", c("omnidireccional", "0°", "90°", "45°"), col=c(1,2,3,4), lty=c(1,
```



Podemos observar que para diferentes distancias y ángulos la variable en cuestión presenta ciertas variaciones, por lo que estaríamos en presencia de **anisotropía**.

Luego de este análisis pasamos a estudiar los residuos de ambos variogramas para entender si existe un margen para analizar la dependencia espacial.

```
res1.v = variog(meuse, coords = coordinates(meuse), data = meuse$zinc, uvec = seq(0, 100

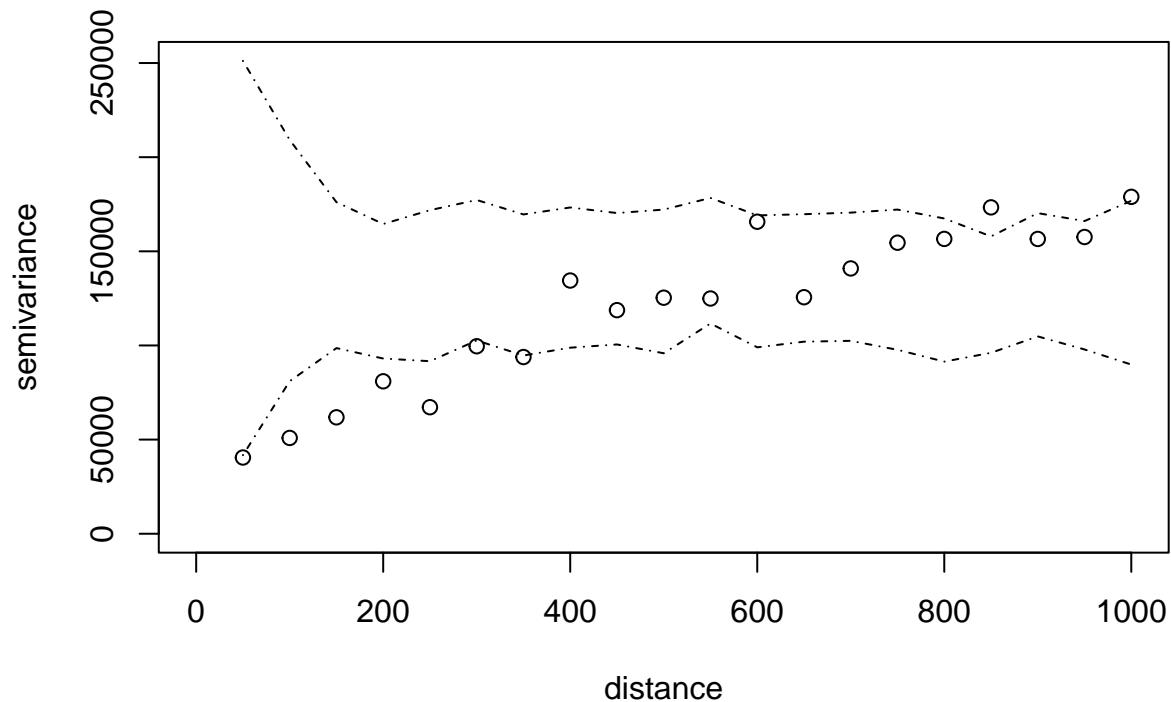
## variog: computing omnidirectional variogram

set.seed(123)

s1 = variog.mc.env(meuse, coords = coordinates(meuse), data = meuse$zinc, obj = res1.v)

## variog.env: generating 99 simulations by permutating data values
## variog.env: computing the empirical variogram for the 99 simulations
## variog.env: computing the envelopes

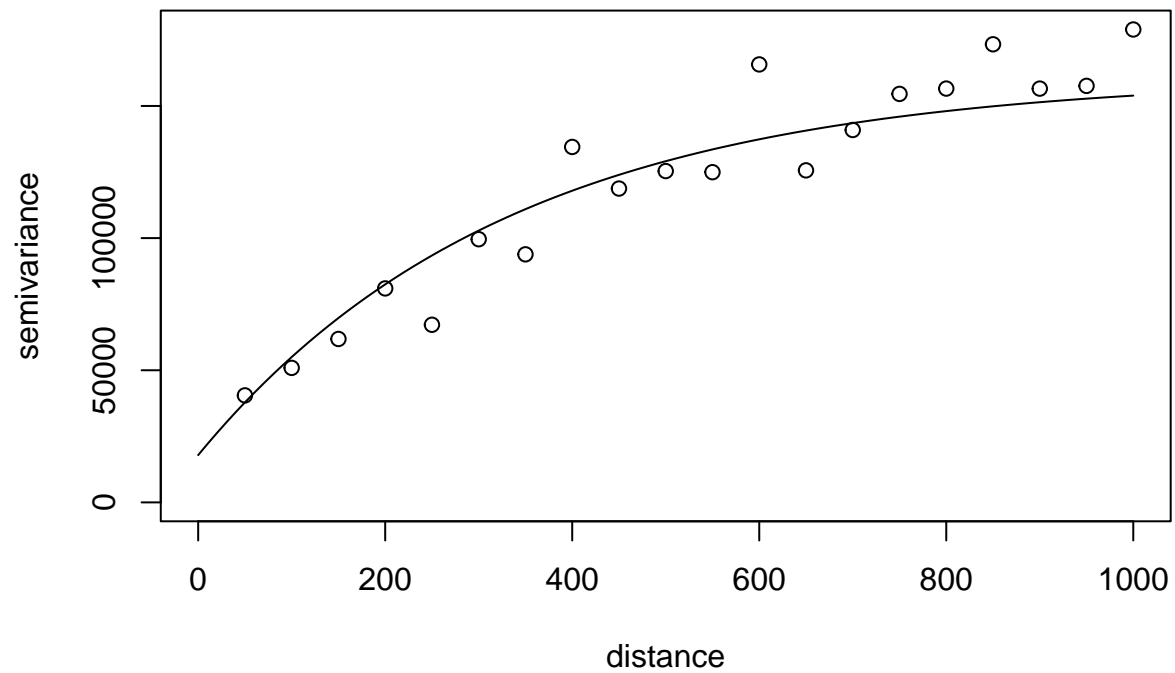
plot(res1.v, env = s1)
```



Podemos observar, usando una simulación simple de Monte Carlo, que existe el margen para estudiar la dependencia espacial, sobre todo para distancias cortas.

Con estos resultados procedemos a ajustar el variograma teórico, para ello consideraremos los siguientes parámetros:

```
res1.v.ef = eyefit(res1.v)
```



```
sigmasq = 143151.79
```

```
phi = 333.33
```

```
nugget = 17893.97
```

```
modelo = "exp"
```

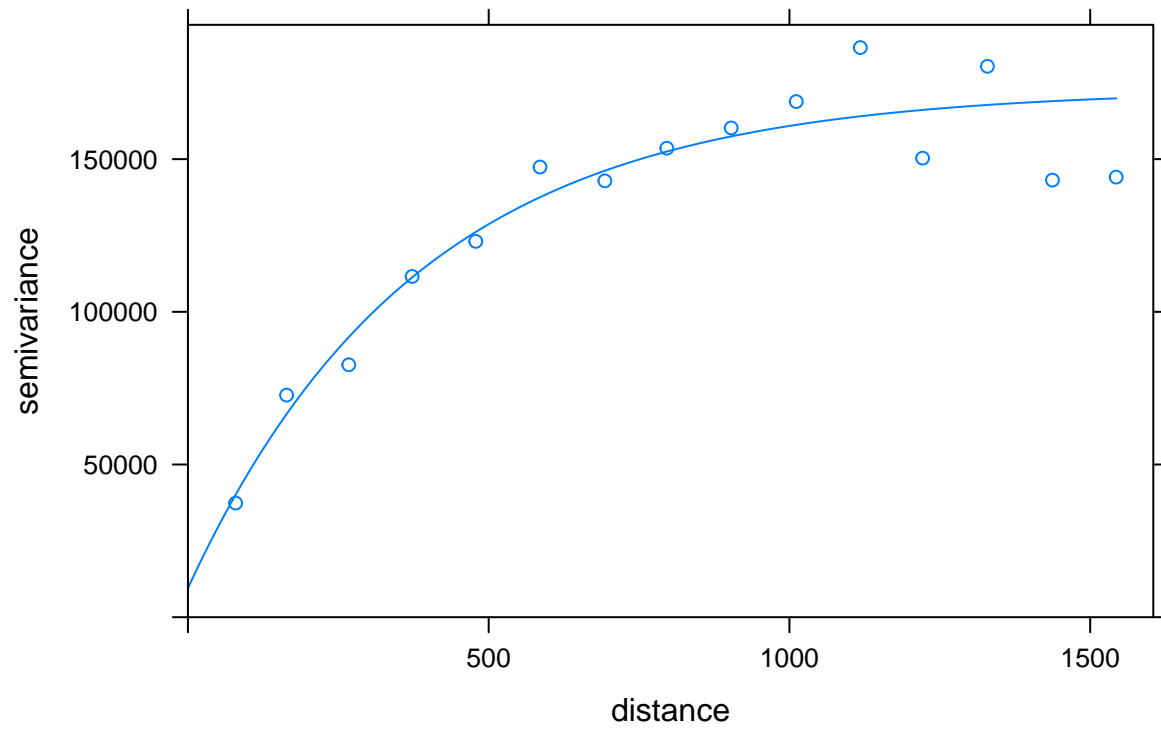
```
v <- variogram(zinc~1, meuse)
```

```
vt_exp = fit.variogram(v, vgm(190000, "Exp", 1400, 30000))
```

```
rango_practico = 381.7098 * 3
```

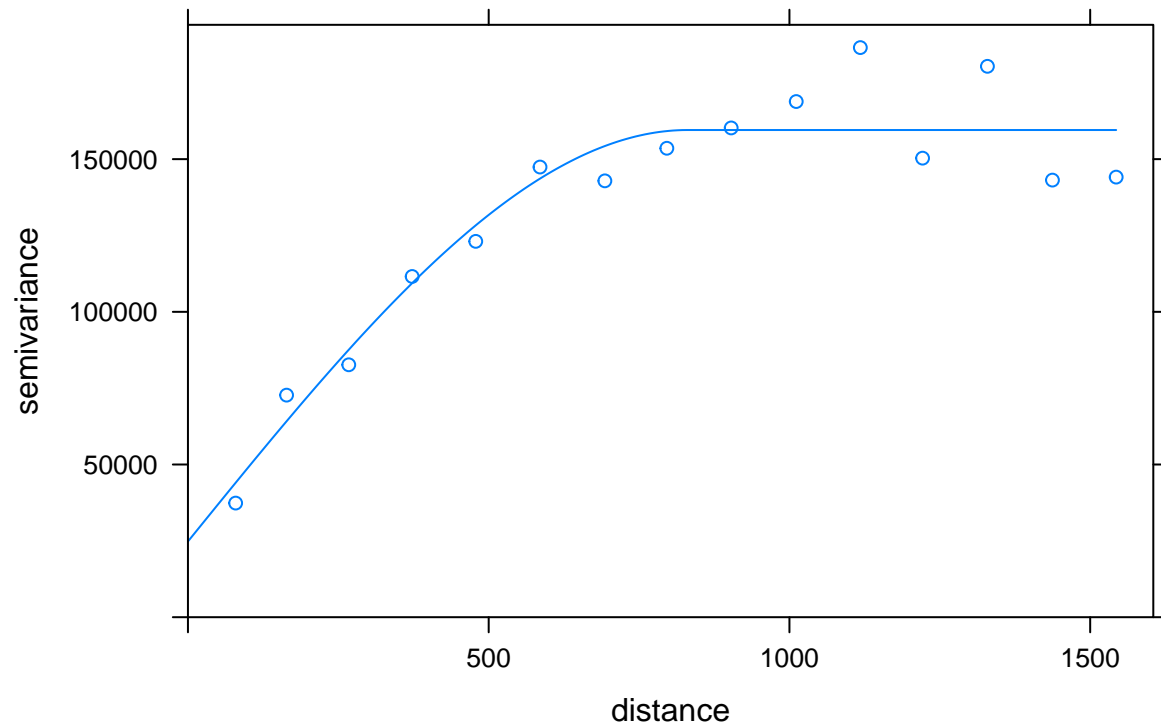
```
plot(v , vt_exp, main = 'Variograma Teórico Exponencial sin tendencia')
```

Variograma Teórico Exponencial sin tendencia



```
vt_sph = fit.variogram(v, vgm(190000, "Sph", 1400, 30000))  
plot(v , vt_sph, main = 'Variograma Teórico Esférico sin tendencia')
```


Variograma Teórico Esférico sin tendencia

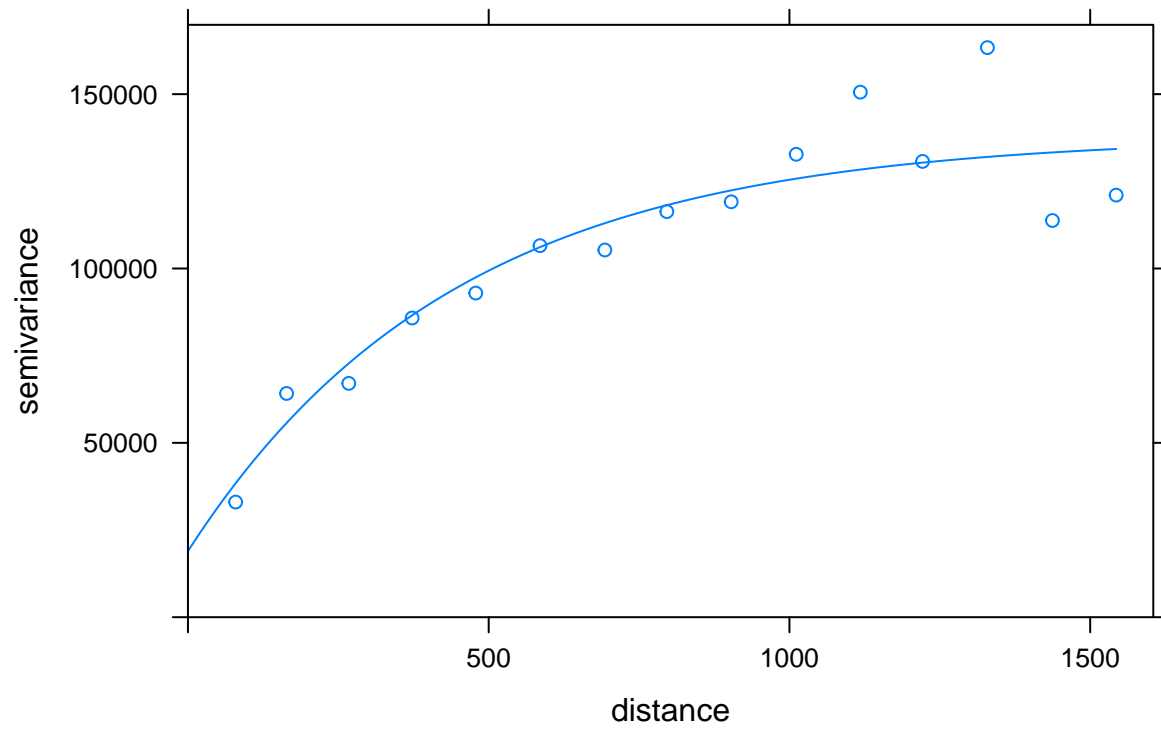


```
vten <- variogram(zinc~x+y, meuse)
```

```
vtent_exp = fit.variogram(vten, vgm(190000, "Exp", 1400, 30000))
```

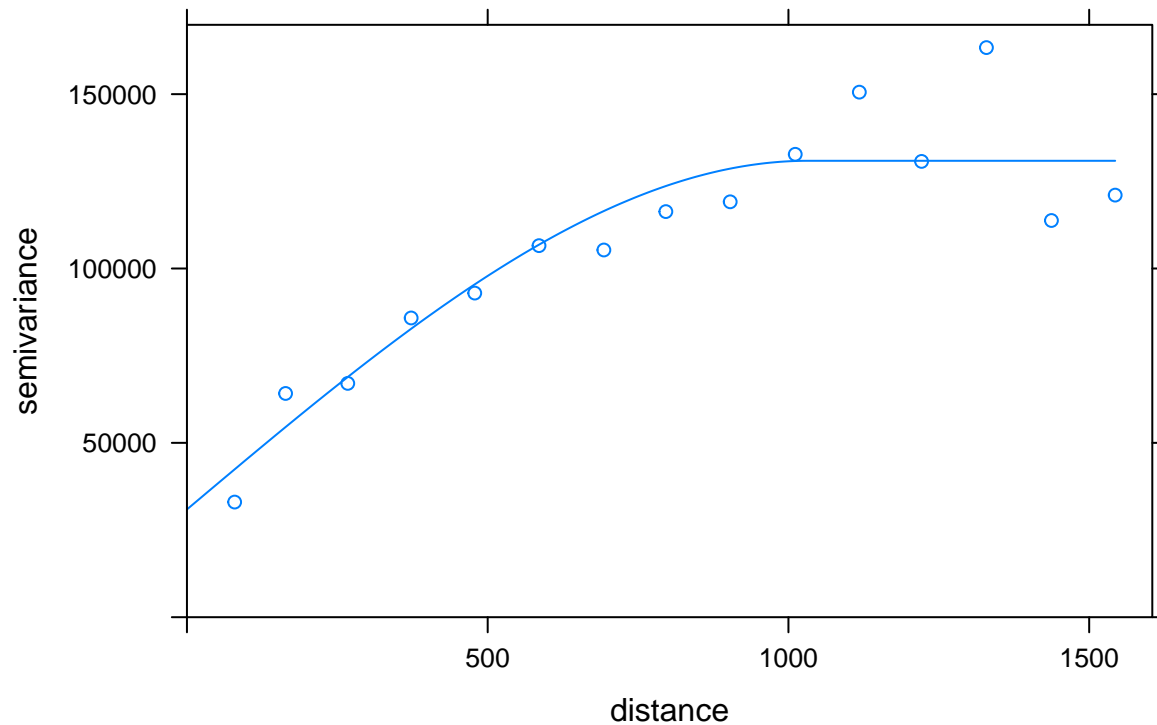
```
plot(vten , vtent_exp, main = 'Variograma Teórico Exponencial con tendencia')
```

Variograma Teórico Exponencial con tendencia



```
vtent_sph = fit.variogram(vten, vgm(190000, "Sph", 1400, 30000))  
plot(vten , vtent_sph, main = 'Variograma Teórico Esférico con tendencia')
```

Variograma Teórico Esférico con tendencia



Como podemos observar tanto para el caso de los modelos con tendencia como sin tendencia el exponencial es el que mejor se ajusta a nuestro variograma. Ahora bien, la pregunta es: ¿el modelo correcto es con o sin tendencia?, no podemos mezclar los errores cuadráticos entre modelos con y sin tendencia. Para ello, hay que analizar los plots, pero como vimos antes, no hay patrones claros. Además, al haber hecho variogramas con y sin tendencia y obtener resultados tan parecidos, se refuerza nuestra hipótesis de que no tienen tendencia.

```
modelo<- c('Modelo Exponencial con tendencia','Modelo Esférico con tendencia','Modelo Exponencial sin tendencia')
error <- c(attr(vtent_exp, 'SSErr'),attr(vtent_sph, 'SSErr'),attr(vt_exp, 'SSErr'),attr(vt_sph, 'SSErr'))
df_error <-data.frame(modelo,error)
df_error
```

```
##              modelo    error
## 1 Modelo Exponencial con tendencia 1975144
## 2   Modelo Esférico con tendencia 2668378
## 3 Modelo Exponencial sin tendencia 1791466
```

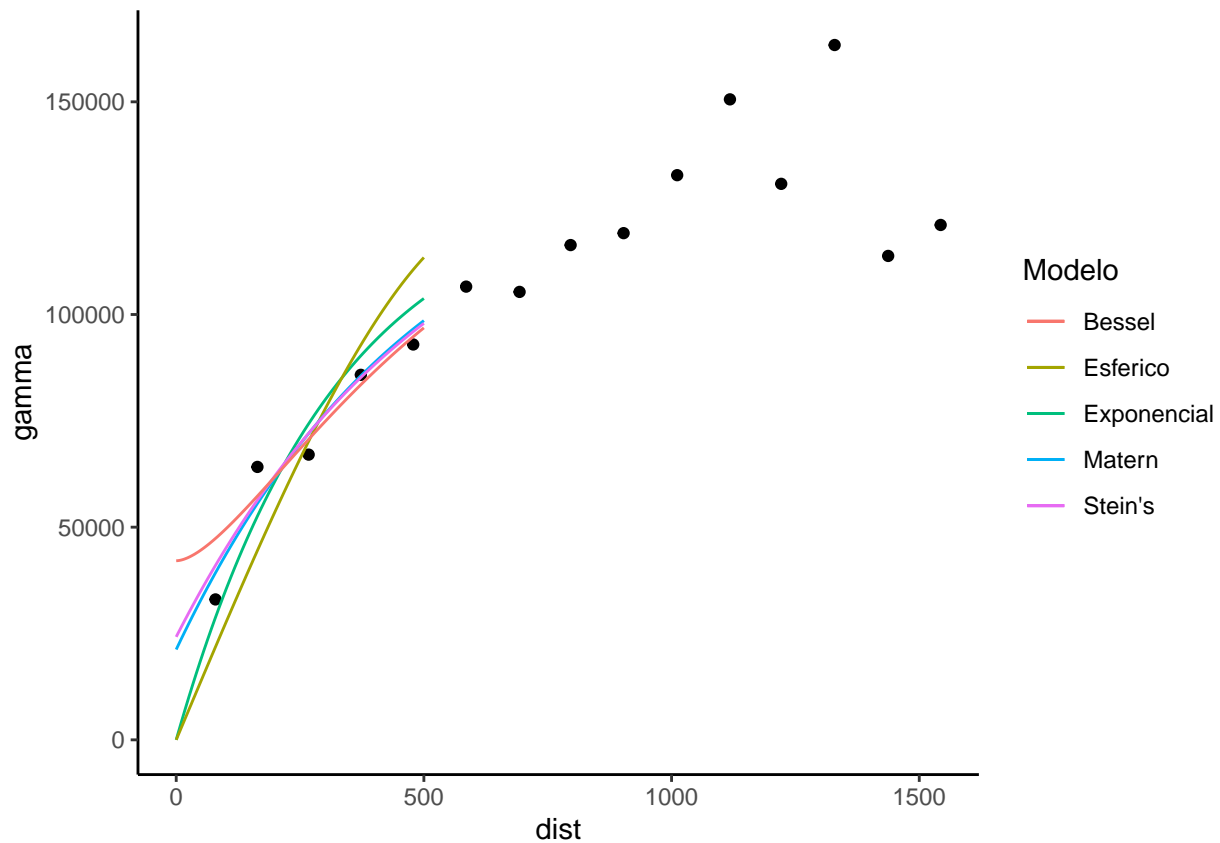
4 Modelo Esférico sin tendencia 2223258

Al probar con otros modelos, el exponencial sigue siendo el mejor ajuste obtenido

```
vExp <- fit.variogram(vten, vgm(model = "Exp"),fit.method = 2)
vSph <- fit.variogram(vten, vgm(model = "Sph"),fit.method = 2)
vMat <- fit.variogram(vten, vgm(model = "Mat", nugget = 1,kappa = 0.5),fit.method = 2)
vBes <- fit.variogram(vten,vgm("Bes"),fit.method = 2)
vSte <- fit.variogram(vten,vgm("Ste"),fit.method = 2)

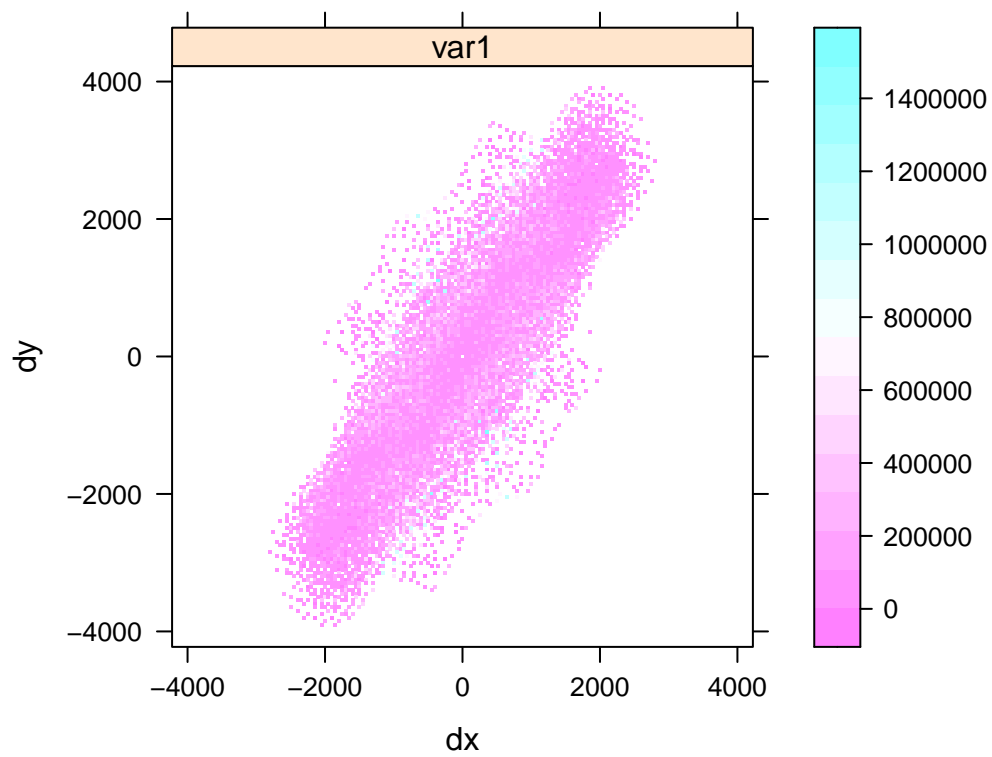
vExpLine=variogramLine(vExp,500)
vSphLine=variogramLine(vSph,500)
vMatLine=variogramLine(vMat,500)
vSteLine=variogramLine(vSte,500)
vBesLine=variogramLine(vBes,500)

ggplot(mapping = aes(dist,gamma))+
  geom_point(data = vten)+
  geom_line(data = vExpLine,aes(color="Exponencial"))+
  geom_line(data = vSphLine,aes(color="Esferico"))+
  geom_line(data = vMatLine,aes(color="Matern"))+
  geom_line(data = vSteLine,aes(color="Stein's"))+
  geom_line(data = vBesLine,aes(color="Bessel"))+
  scale_color_discrete("Modelo")+
  theme_classic()
```

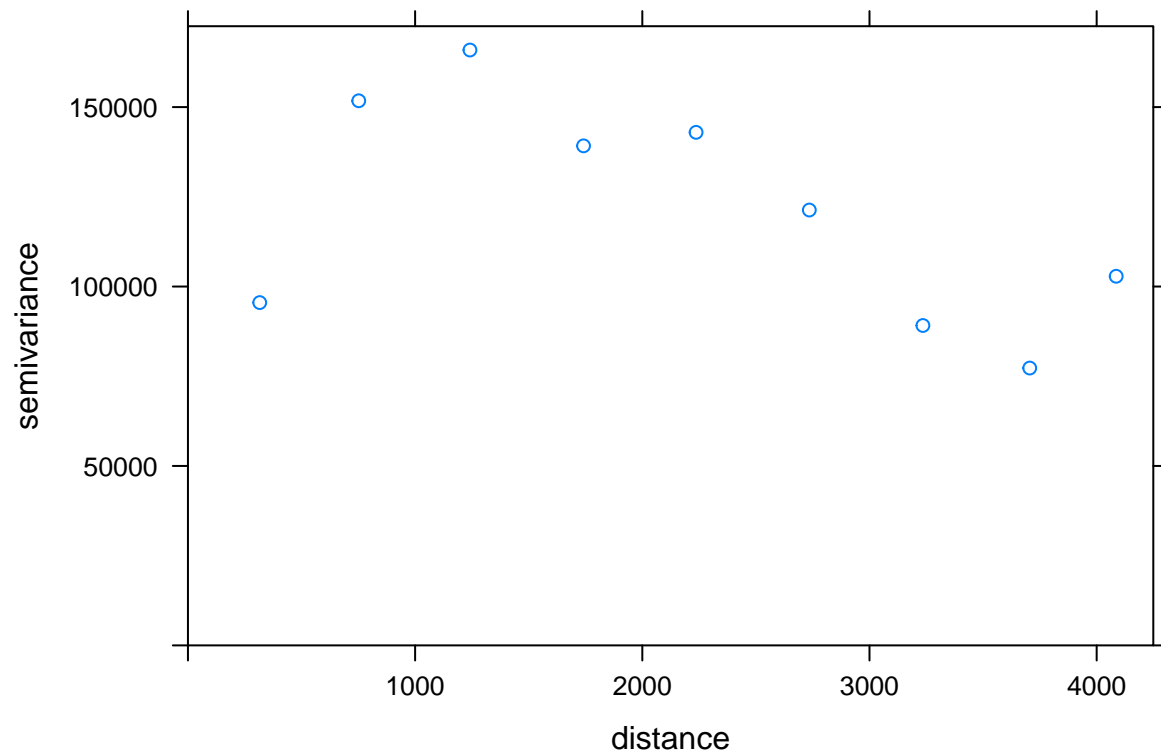


Analizando nuevamente la **isotropía**, consideramos el *cutoff*, el cual es la máxima distancia que tiene sentido considerar entre los puntos, en este caso es 4,200 de distancia.

```
v v <- variogram(zinc~1, meuse, cutoff = 4200, width = 50, map=T)
plot(vv)
```



```
vv <- variogram(zinc~1, meuse, cutoff = 4200, width = 500)
plot(vv)
```



Como mencionamos anteriormente estamos en presencia de un proceso **anisotrópico**, con dirección por continuidad, alcanzando otros valores si corremos la dirección perpendicular al sentido de la mancha.