

# Distribución espacial de los puntos de calor - Hot Spots para los incendios forestales en la República Dominicana y predicción de las zonas mas vulnerables usando el metodo de Kriging universal

**Diego Cordero** *Estudiante de Maestría en Teledetección y Ciencias de la Información Geográfica, Universidad Autónoma de Santo Domingo (UASD)*

---

Esta investigación se basa en información geográfica de libre acceso encontrada en la web y pretende realizar un análisis geoespacial que permita conocer cuál es la distribución espacial de los puntos de calor - Hot Spots para los incendios forestales en la República Dominicana. Para lo cual se descargó la información correspondiente para el periodo 2009-2019 de la pagina de NASA-FIRMS (2020). El resultado final esperado que se busca mediante el desarrollo de un script bajo el software R y la herramienta RStudio es un Cluster LISA que indique la relación espacial de los municipios con los incendios y una predicción de las zonas mas vulnerables usando el metodo de Kriging universal teniendo como base la elevación de un modelo digital de elevación (DEM).

*Keywords:* r, rstudio, fire forest, incendio forestal, kriging, LISA cluster

---

## 1 Introducción

Los incendios forestales son un problema mundial, por lo que mediante este estudio se espera realizar un acercamiento para dar conocer a mediante el uso de análisis geoestadisticos las posibles causas de variables naturales que pueden influir en la propagación de los mismos. Mediante herramientas y data de libre acceso se pretende realizar un acercamiento que permita establecer la relación directa o indirecta sobre la problemática principal que se ha intesificado durante el 2019 según Diario Libre (2019) siendo Los parques nacionales los más quemados, el mismo establece que para los primeros meses de 2019 tuvo un incremento en los incendios Vs 2018. El gobierno dominicano entre las medidas preventivas previstas en cabeza del Ministerio de Medio Ambiente y Recursos Naturales, en coordinación con ministerio de Obras Publicas y el ministerio de Defensa, activó un Plan Preventivo de Emergencia (PPE) en la que se sobrevolará en helicópteros y drones las zonas vulnerables donde se producen conatos de incendio por conuquismo, con el propósito de detectar los daños en los bosques, atrapar a los culpables y actuar contra los responsables asegura EFE (2019).

## 2 Metodología

### 2.1 Librerías Necesarias y adición de data

Inicialmente se usarán las siguientes librerías y el archivo para el desarrollo de los LISA cluster realizado por Martínez Batlle (2019c) para el desarrollo del script para la investigación

```
library(sf)
library(sp)
```

```

library(raster)
library(tidyverse)
library(tmap)
library(RColorBrewer)
library(lmtest)
library(spdep)
library(parallel)
library(ggplot2)
library(gstat)
library(stars)
source('lisaclusters.R')

```

Teniendo en cuenta la información disponible se realizó la carga de la data de **incendios, municipios, y uso de suelo**, la primera de las capas fue trabajada inicialmente en QGis realizando un join de la data de VIIRS y MODIS, sin embargo se percibieron dos problemas. La cantidad de zonas de alta densidad de incendios que correspondían a canteras y zonas industriales a lo largo del país como se presenta en las Figuras 1 y en segunda instancia la notoriedad que no todos los incendios correspondían a incendios forestales.

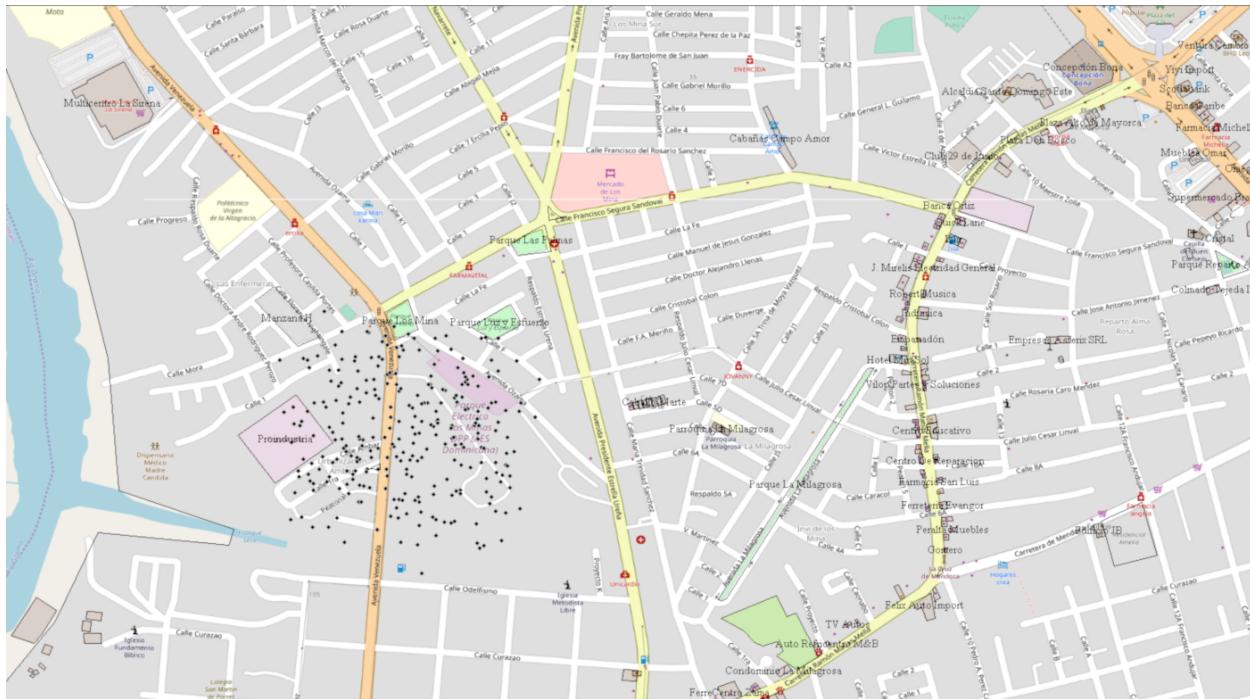


Figure 1: Densidad de Incendios en zonas Industriales

Para el primer problema y haciendo uso de la data de la OpenStreetMaps (2020), se realizó un buffer de los centroides de los polígonos de canteras y edificios de un radio de 1.11 km o 0.01 grados teniendo en cuenta que la data se encuentra en la referencia espacial WGS84:4326 usando el archivo gis\_osm\_landuse\_a\_free\_1 eliminando todos los datos que se encontrarán dentro del mismo como se muestra en la Figura 2.

Para el segundo problema y haciendo uso de la data de GlobCover (2005) se realizó un subset de los usos de suelo contemplando los valores entre 30 y 120 de la siguiente tabla tomada de la

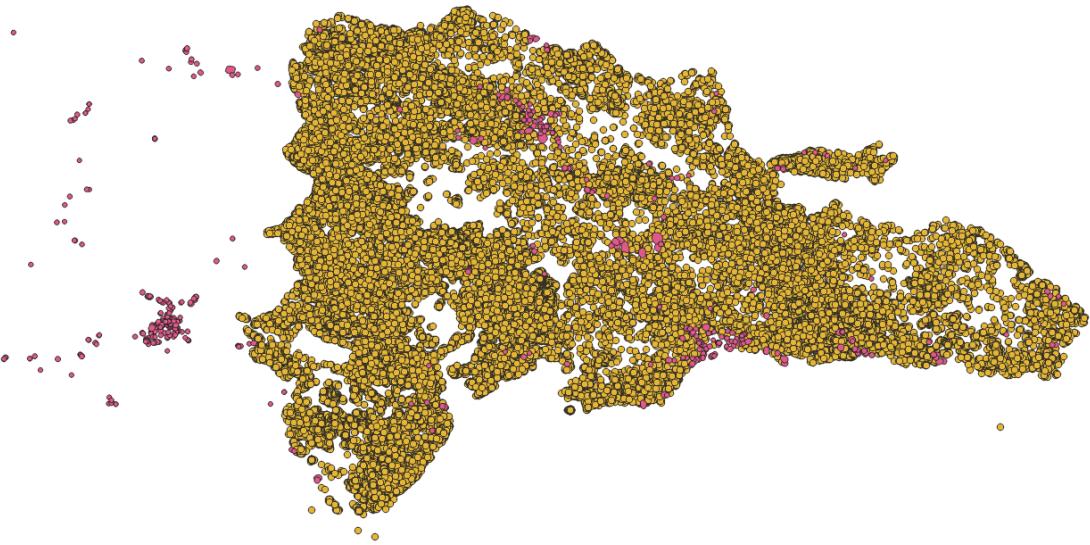


Figure 2: Resultado Buffer

data descargada de GlobCover.

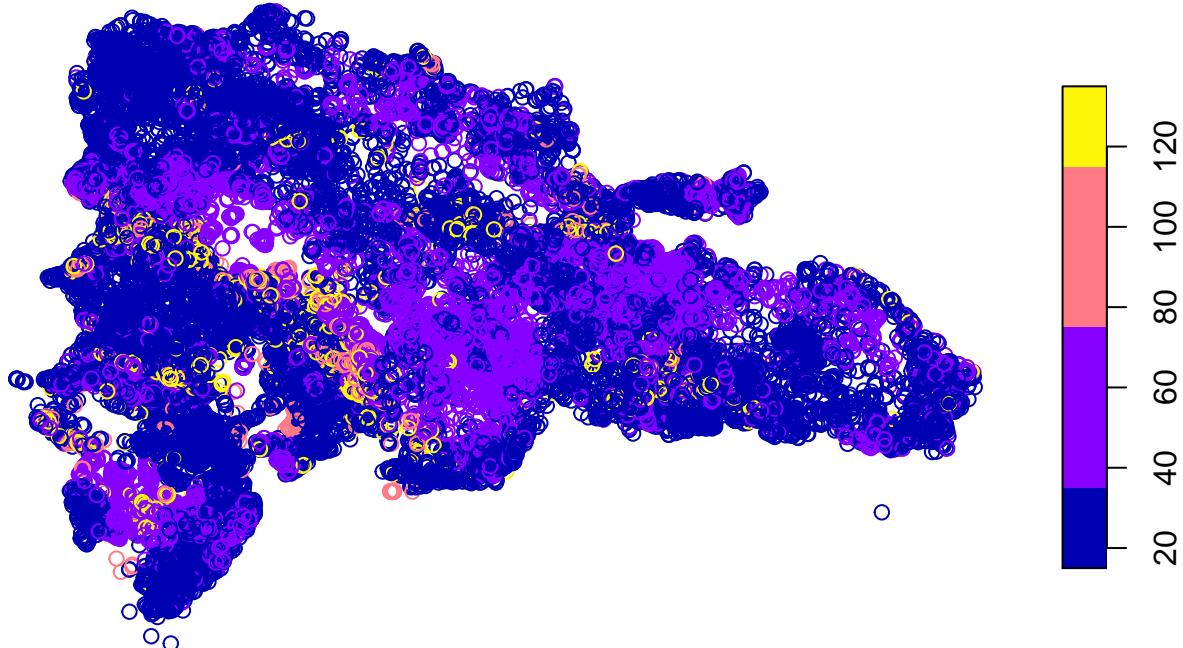
Value	Label
11	Post-flooding or irrigated croplands (or aquatic)
14	Rainfed croplands
20	Mosaic cropland (50-70%) / vegetation (grassland/shrubland/forest) (20-50%)
30	Mosaic vegetation (grassland/shrubland/forest) (50-70%) / cropland (20-50%)
40	Closed to open (>15%) broadleaved evergreen or semi-deciduous forest (>5m)
50	Closed (>40%) broadleaved deciduous forest (>5m)
60	Open (15-40%) broadleaved deciduous forest/woodland (>5m)
70	Closed (>40%) needleleaved evergreen forest (>5m)
90	Open (15-40%) needleleaved deciduous or evergreen forest (>5m)
100	Closed to open (>15%) mixed broadleaved and needleleaved forest (>5m)
110	Mosaic forest or shrubland (50-70%) / grassland (20-50%)
120	Mosaic grassland (50-70%) / forest or shrubland (20-50%)
130	Closed to open (>15%) (broadleaved or needleleaved, evergreen or deciduous) shrubland (<5m)
140	Closed to open (>15%) herbaceous vegetation (grassland, savannas or lichens/mosses)
150	Sparse (<15%) vegetation
160	Closed to open (>15%) broadleaved forest regularly flooded (semi-permanently or temporarily) - Fresh or brackish water
170	Closed (>40%) broadleaved forest or shrubland permanently flooded - Saline or brackish water

Value	Label
180	Closed to open (>15%) grassland or woody vegetation on regularly flooded or waterlogged soil - Fresh, brackish or saline water
190	Artificial surfaces and associated areas (Urban areas >50%)
200	Bare areas
210	Water bodies
220	Permanent snow and ice
230	No data (burnt areas, clouds,...)

Se realizó una comprobación para conocer si los valores correspondían a los usados para la obtención de los datos forestales y se realizó una intersección entre la capa de incendios y la de municipios asignandole a cada dato el valor ENLACE.

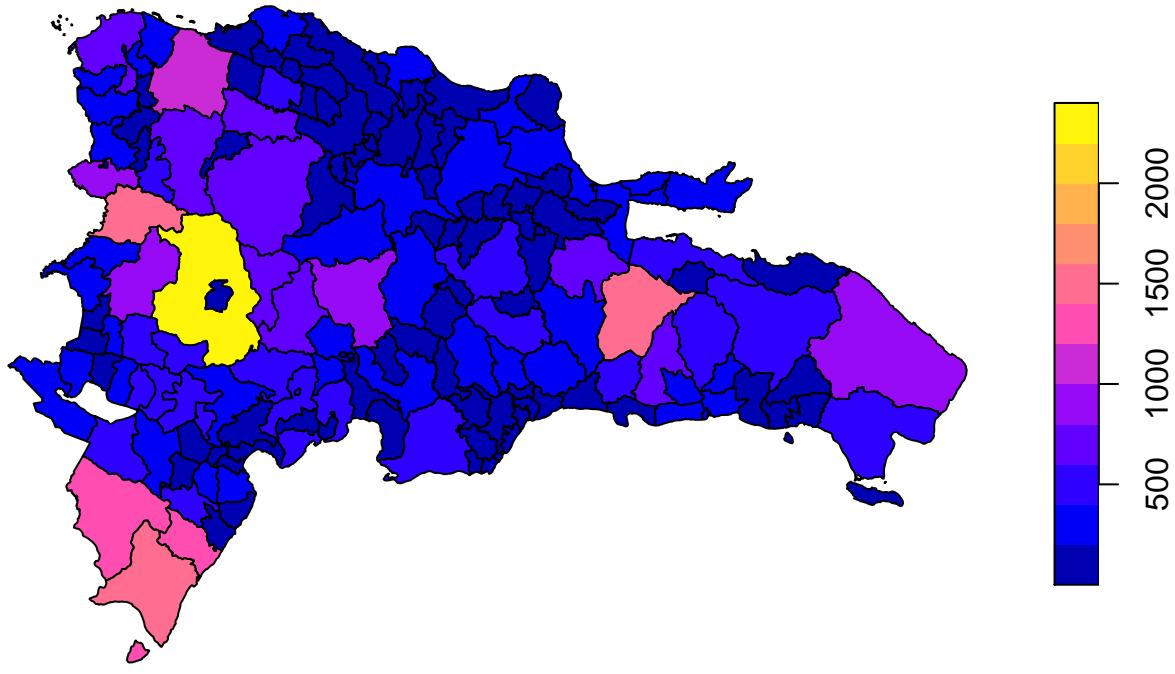
```
##      Min. 1st Qu. Median     Mean 3rd Qu.    Max.
## 30.00   30.00  30.00  37.91  40.00 120.00
```

### GLOBCOVER\_RD.color



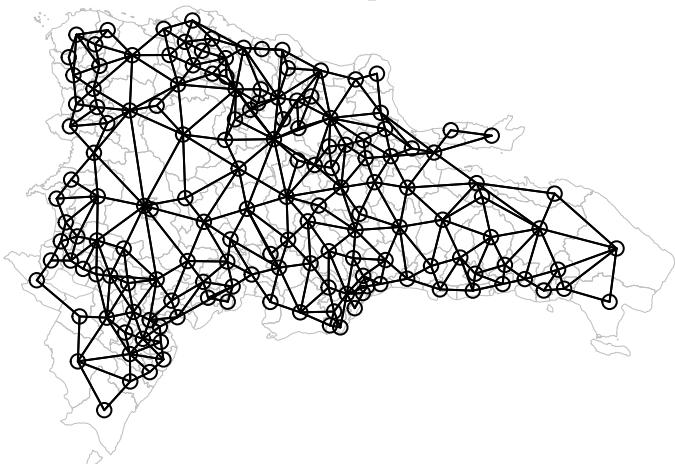
Con la información resulante para cada dato, se realizó un arrange entre el numero de datos con el mismo ENLACE y se concatenó con la data de municipios permitiendo así mostrar un mapa con la cantidad de incendios por municipio.

## NumIncendios

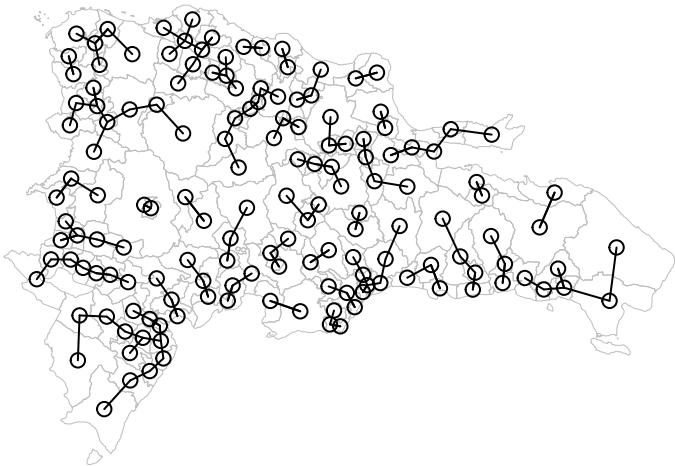


### 2.2 Vecindad

Para hallar los vecinos, se debe convertir la capa de Incendios por Municipio a `SpatialPolygonsDataFrame`, y basado en el anterior se crea un objeto de vecindad por contigüidad usando el criterio queen, dandonos como resultado el mapa de vinculos de vecindad de cada municipio,



Sin embargo se debe hallar los vinculos de vecindad entre los municipios para lo que se usa su vecino mas proximo



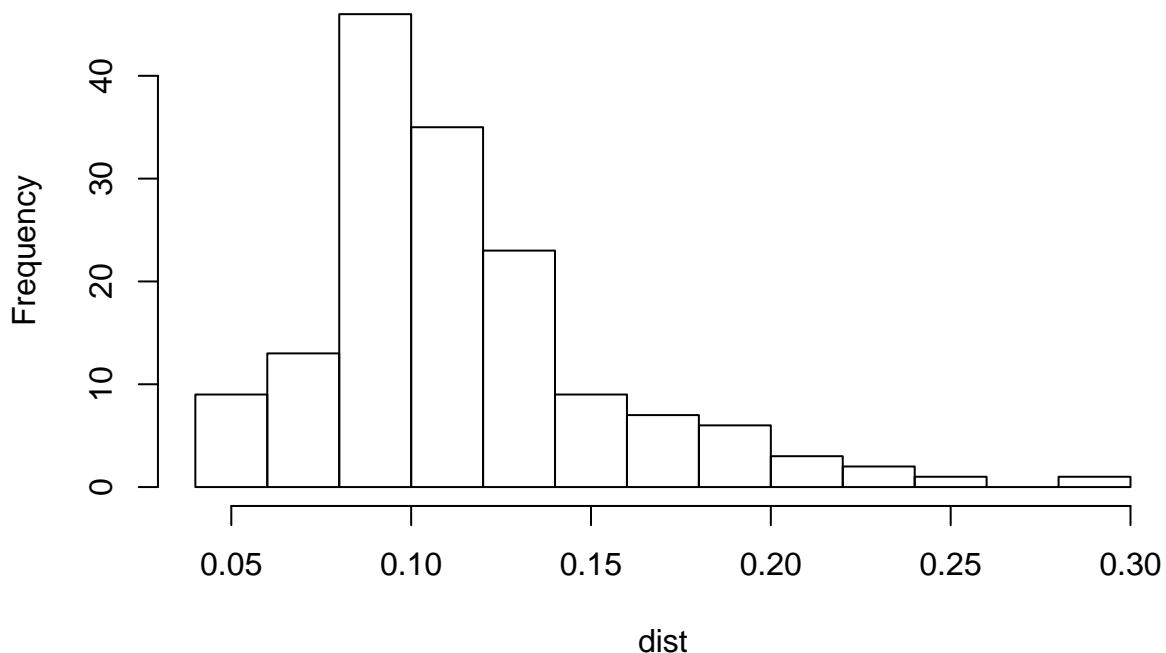
Basandose en el mapa resultante se visualizan los datos estadísticos descriptivos sobre las distancias entre los municipios permitiendo determinar los valores extremos (mínimo y máximo) de distancias a vecinos más próximos.

```
summary(dist)
```

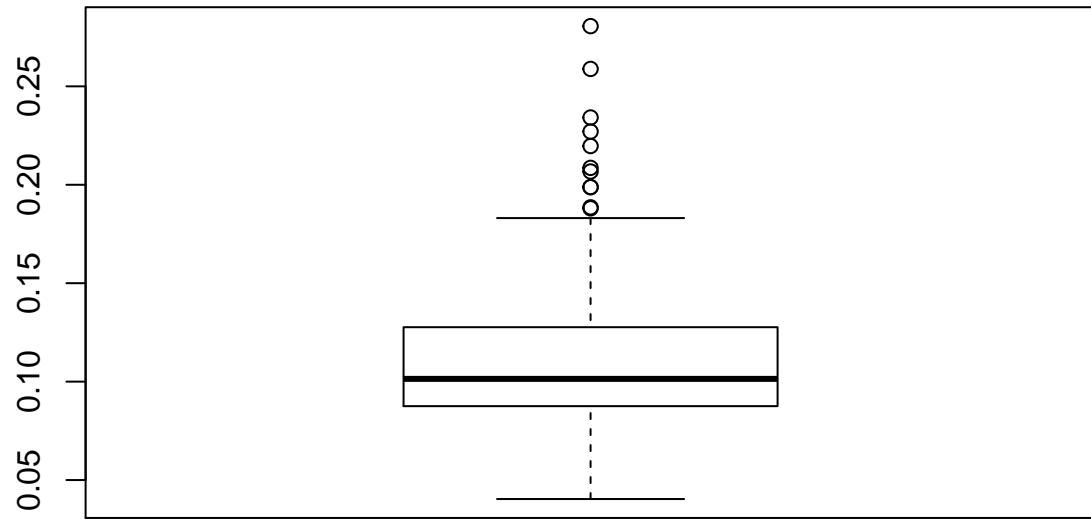
```
##      Min. 1st Qu. Median     Mean 3rd Qu.    Max.
## 0.04036 0.08754 0.10140 0.11396 0.12765 0.28058
```

```
hist(dist)
```

**Histogram of dist**



```
boxplot(dist)
```



A continuación deben asignarse los pesos espaciales utilizando el estilo *weighted* o “W”, donde los pesos de las observaciones vecinas a una observación dada, suman 1, asignando pesos homogéneos a cada vecino, sin embargo se realizará tambien al estilo binario. (Bivand, Pebesma, & Gomez-Rubio, 2008), bajo el cual el objeto  $j$  recibe un peso de 1 ante el objeto  $i$ , siempre que el primero sea vecino del primero; por el contrario, si  $k$  no es vecino de  $i$  recibe un peso de 0 por ante  $i$ . Los pesos son indicativos de la fortaleza de la relación entre dos o más observaciones.

```
munInc.w.W
```

```
## Characteristics of weights list object:  
## Neighbour list object:  
## Number of regions: 155  
## Number of nonzero links: 804  
## Percentage nonzero weights: 3.346514  
## Average number of links: 5.187097  
##  
## Weights style: W  
## Weights constants summary:  
##      n     nn    S0        S1        S2  
## W 155 24025 155 65.94606 650.7687
```

```
munInc.w.B
```

```
## Characteristics of weights list object:  
## Neighbour list object:  
## Number of regions: 155  
## Number of nonzero links: 804  
## Percentage nonzero weights: 3.346514  
## Average number of links: 5.187097  
##  
## Weights style: B
```

```

## Weights constants summary:
##      n    nn   S0    S1    S2
## B 155 24025 804 1608 19520

```

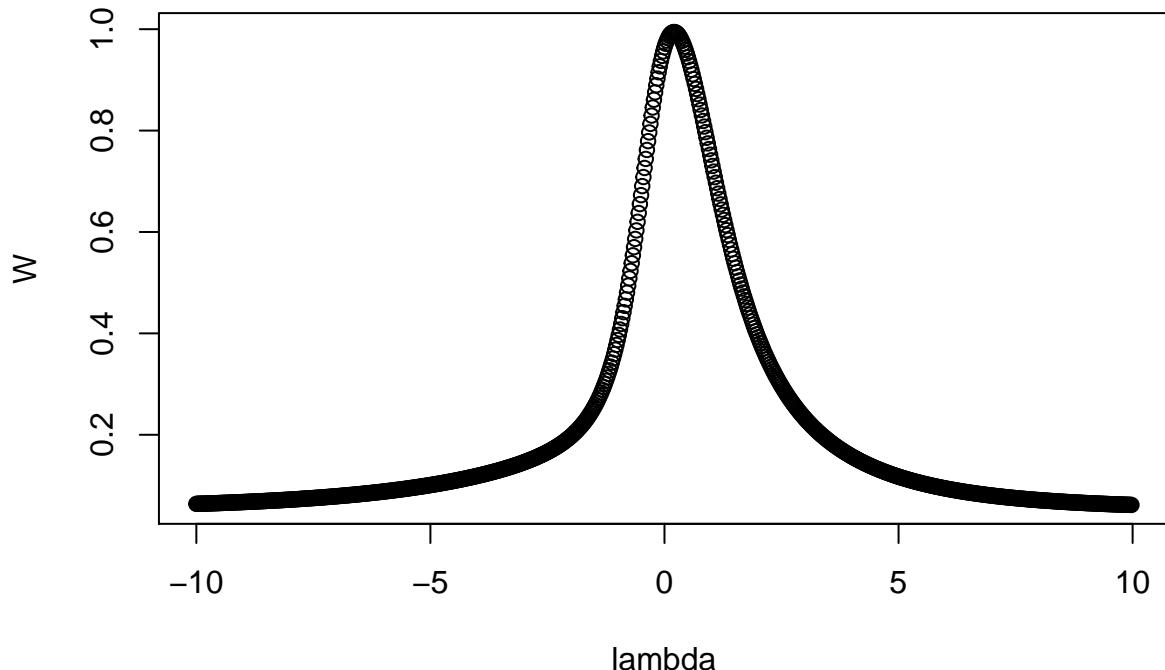
### 2.3 Autocorrelación espacial

Para trabajar en la autocorrelación espacial se decide generar varias columnas entre las que se encuentran *porcentaje de incendios* y *porcentaje por km<sup>2</sup>* a su vez y puesto que existen municipios mas grandes que otros y esto puede generar valores erroneos para el estudio se generan una columna de *incendios por area*, sin embargo para efectos de estudio se genera una columna usando el *logaritmo*, *Ladder of Powers (escalera de potencias de Tukey)* y *tukey\_lambda*

```

##
##      lambda      W Shapiro.p.value
## 409     0.2 0.9944          0.8194
##
## if (lambda > 0){TRANS = x ^ lambda}
## if (lambda == 0){TRANS = log(x)}
## if (lambda < 0){TRANS = -1 * x ^ lambda}

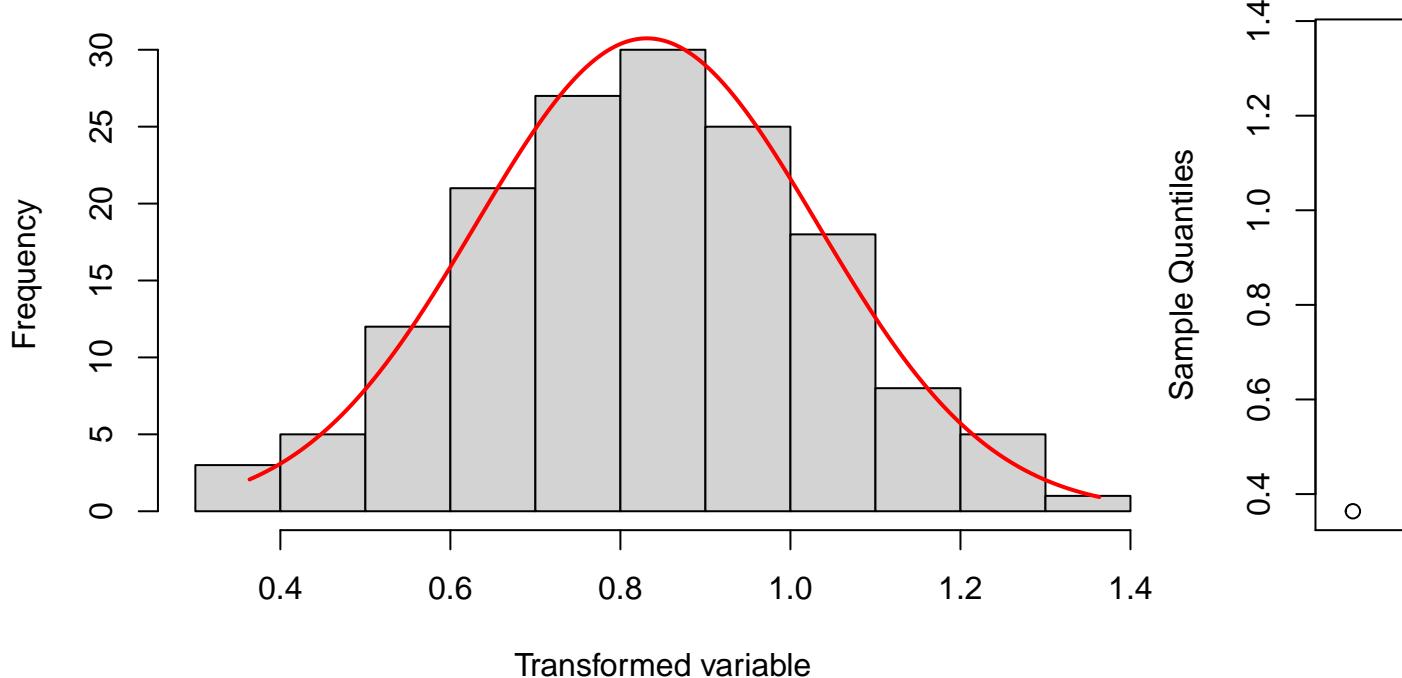
```



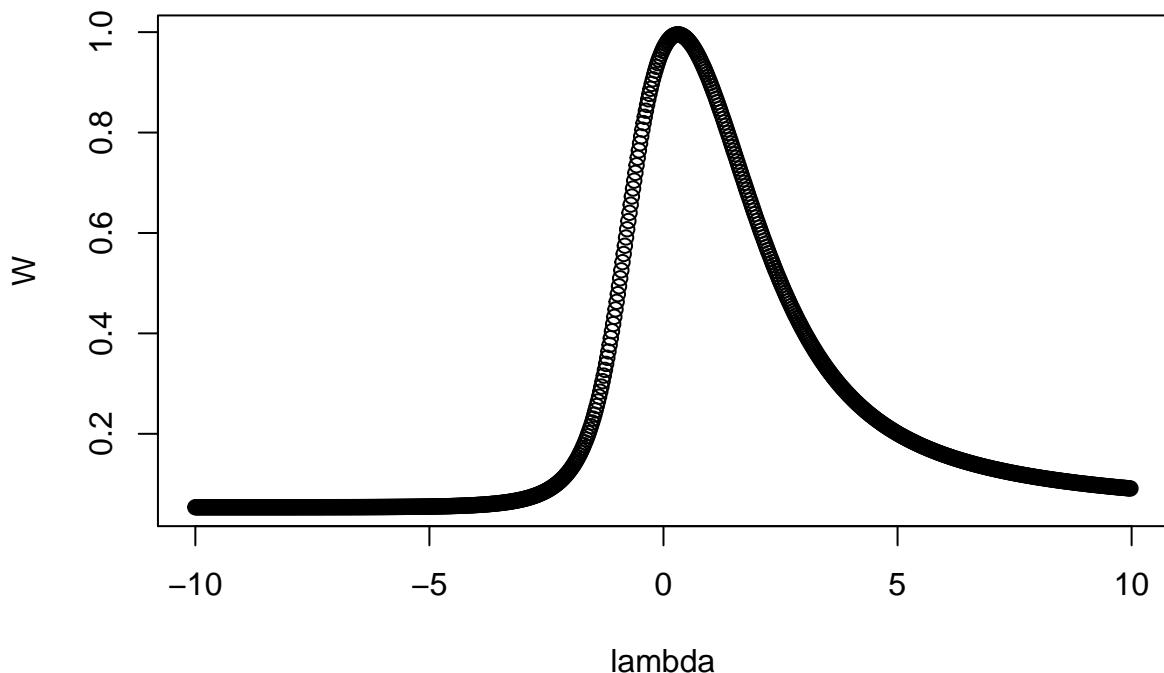
```

##
##      lambda      W Shapiro.p.value
## 409     0.2 0.9944          0.8194
##
## if (lambda > 0){TRANS = x ^ lambda}
## if (lambda == 0){TRANS = log(x)}
## if (lambda < 0){TRANS = -1 * x ^ lambda}

```



```
##  
##      lambda      W Shapiro.p.value  
## 413    0.3 0.9956          0.9253  
##  
## if (lambda > 0){TRANS = x ^ lambda}  
## if (lambda == 0){TRANS = log(x)}  
## if (lambda < 0){TRANS = -1 * x ^ lambda}
```



```

##  

##      lambda      W Shapiro.p.value  

## 413     0.3 0.9956          0.9253  

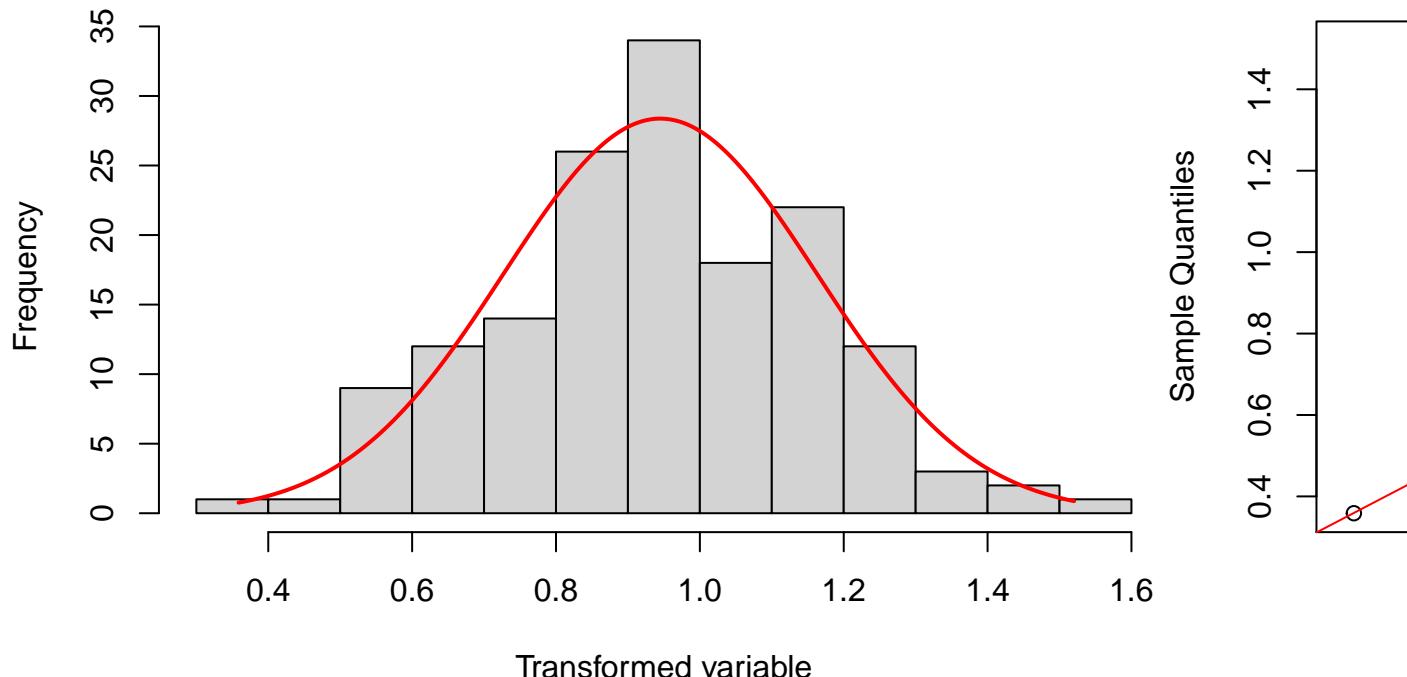
##  

## if (lambda > 0){TRANS = x ^ lambda}  

## if (lambda == 0){TRANS = log(x)}  

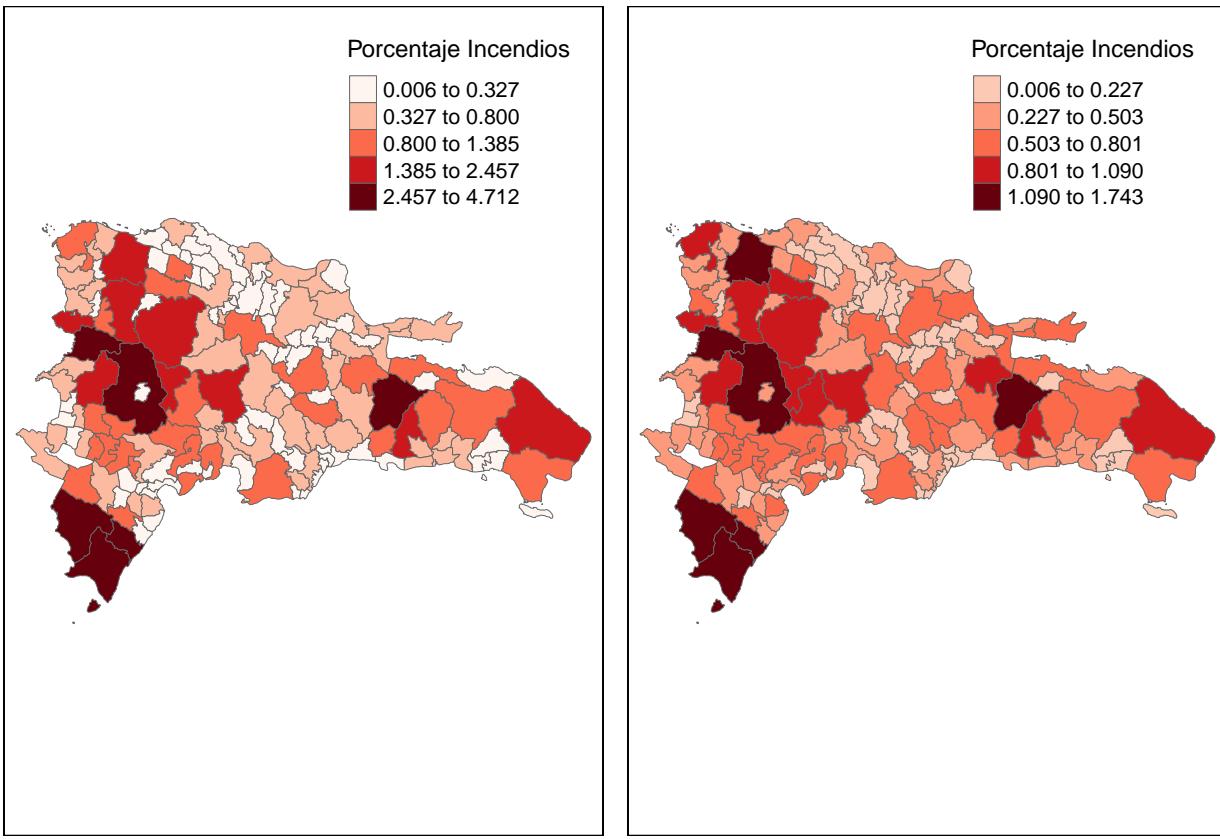
## if (lambda < 0){TRANS = -1 * x ^ lambda}

```



Teniendo en cuenta los resultados se genera el polígono realizando una union entre Incendios por Municipio y la tabla resultante anterior.

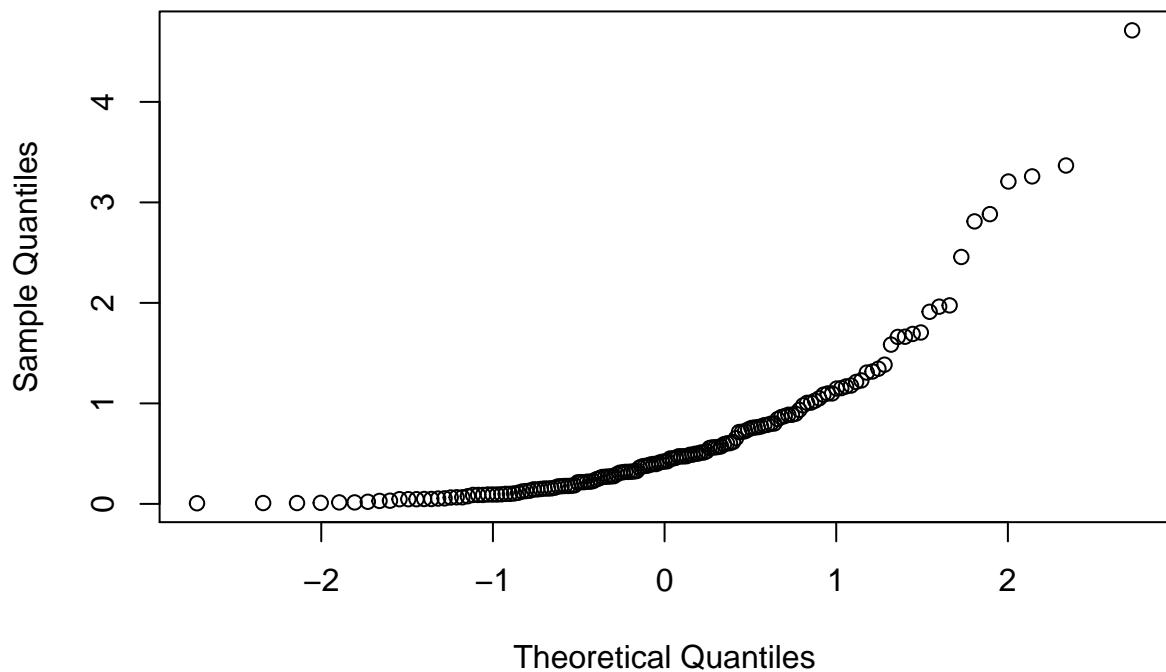
Basandose en la capa resultante se realizan dos mapas que representan los porcentajes de incendios por cada municipio sin importar el área mostrando a los municipios del sur oeste y San Juan como los de mayor tasa de incendios.



Al realizar las pruebas de “Shapiro-Wilk” para las columnas *Porcentaje de Incendios*, *Log Porcentaje de Incendios* y *Tukey Porcentaje de Incendios* se observa que tan solo la ultima tiene valores p mayores a 0.05 por lo que no permite su rechazo y se considera el supuesto de normalidad. Al realizar la prueba “Breusch-Pagan” para cada una de estas todas las variables tienen resultados que pueden ser interpretados con una hipótesis nula que no puede ser rechazada y con presunción de homocedasticidad esto debido a que todas superan el p mayor a 0.05.

```
#qq
qqnorm(munIncPerc$IncPercentage)
```

## Normal Q-Q Plot

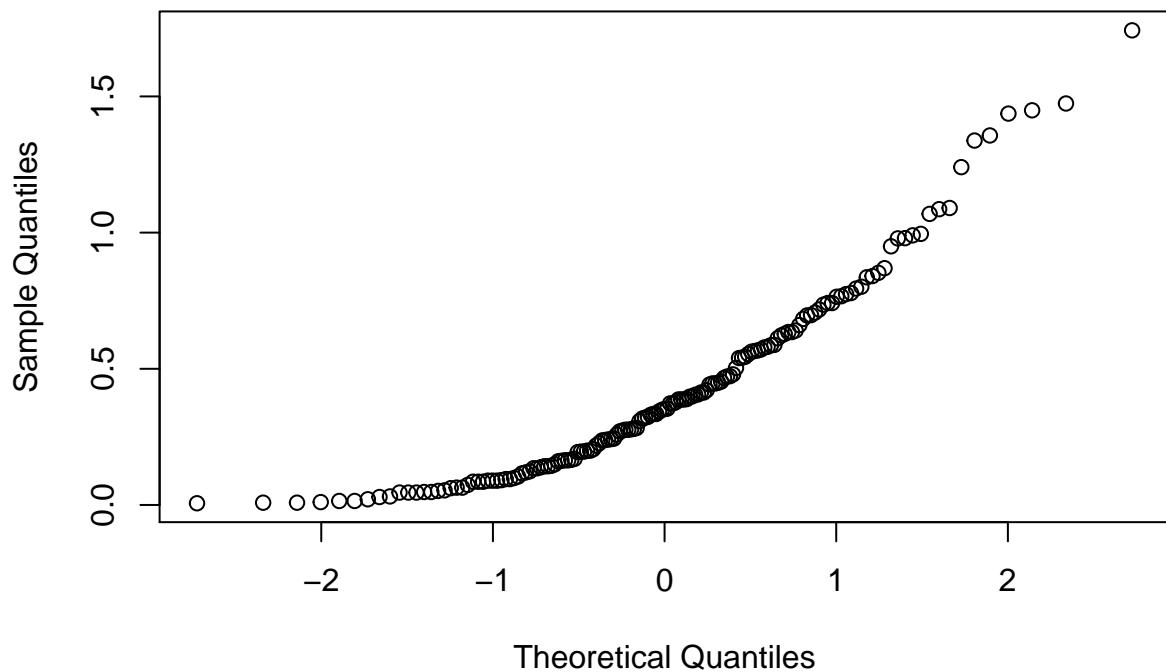


```
shapiro.test(munIncPerc$IncPercentage)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: munIncPerc$IncPercentage  
## W = 0.74127, p-value = 3.202e-15
```

```
qqnorm(munIncPerc$IncPercentage_log)
```

## Normal Q-Q Plot

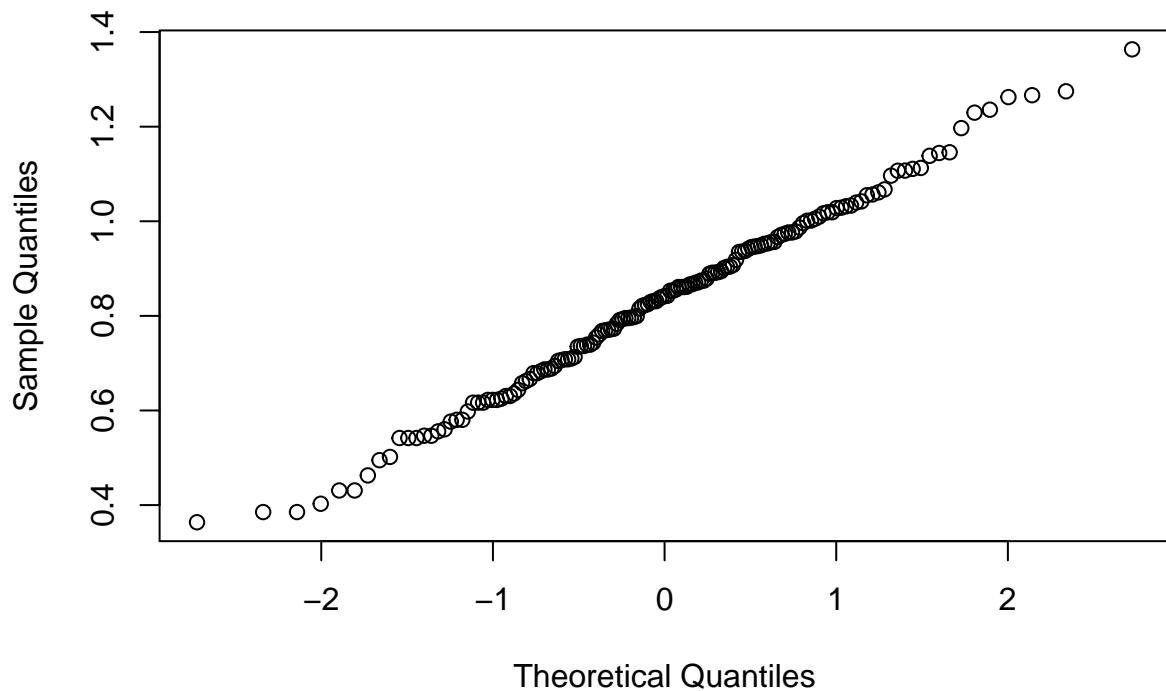


```
shapiro.test(munIncPerc$IncPercentage_log)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: munIncPerc$IncPercentage_log  
## W = 0.89833, p-value = 6.953e-09
```

```
qqnorm(munIncPerc$IncPercentage_tukey)
```

## Normal Q-Q Plot



```
shapiro.test(munIncPerc$IncPercentage_tukey)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: munIncPerc$IncPercentage_tukey  
## W = 0.99441, p-value = 0.8194
```

```
munIncPerc %>% lm(IncPercentage ~ x, .) %>% bptest()
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: .  
## BP = 1.6534, df = 1, p-value = 0.1985
```

```
munIncPerc %>% lm(IncPercentage ~ y, .) %>% bptest()
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: .  
## BP = 1.1426, df = 1, p-value = 0.2851
```

```

munIncPerc %>% lm(IncPercentage_log ~ x, .) %>% bptest()

## 
##   studentized Breusch-Pagan test
## 
## data: .
## BP = 0.61098, df = 1, p-value = 0.4344

```

```

munIncPerc %>% lm(IncPercentage_log ~ y, .) %>% bptest()

## 
##   studentized Breusch-Pagan test
## 
## data: .
## BP = 1.3799, df = 1, p-value = 0.2401

```

```

munIncPerc %>% lm(IncPercentage_tukey ~ x, .) %>% bptest()

## 
##   studentized Breusch-Pagan test
## 
## data: .
## BP = 0.40819, df = 1, p-value = 0.5229

```

```

munIncPerc %>% lm(IncPercentage_tukey ~ y, .) %>% bptest()

## 
##   studentized Breusch-Pagan test
## 
## data: .
## BP = 0.08262, df = 1, p-value = 0.7738

```

Teniendo en cuenta los resultados anteriores, se evalua la *autocorrelación espacial global* teniendo en cuenta la prueba para la “I” de Moran usando los pesos estandarizados y los binarios para la variable de *Tukey* estableciendo que las comprobaciones son menores al alpha estipulado permitiendo afirmar al estudio que existe una autocorrelación espacial.

```

(gmoranw <- moran.test(x = munIncPerc$IncXArea_tukey, listw = munInc.w.W))

## 
##   Moran I test under randomisation
## 
## data: munIncPerc$IncXArea_tukey
## weights: munInc.w.W
## 
## Moran I statistic standard deviate = 8.6616, p-value < 2.2e-16

```

```

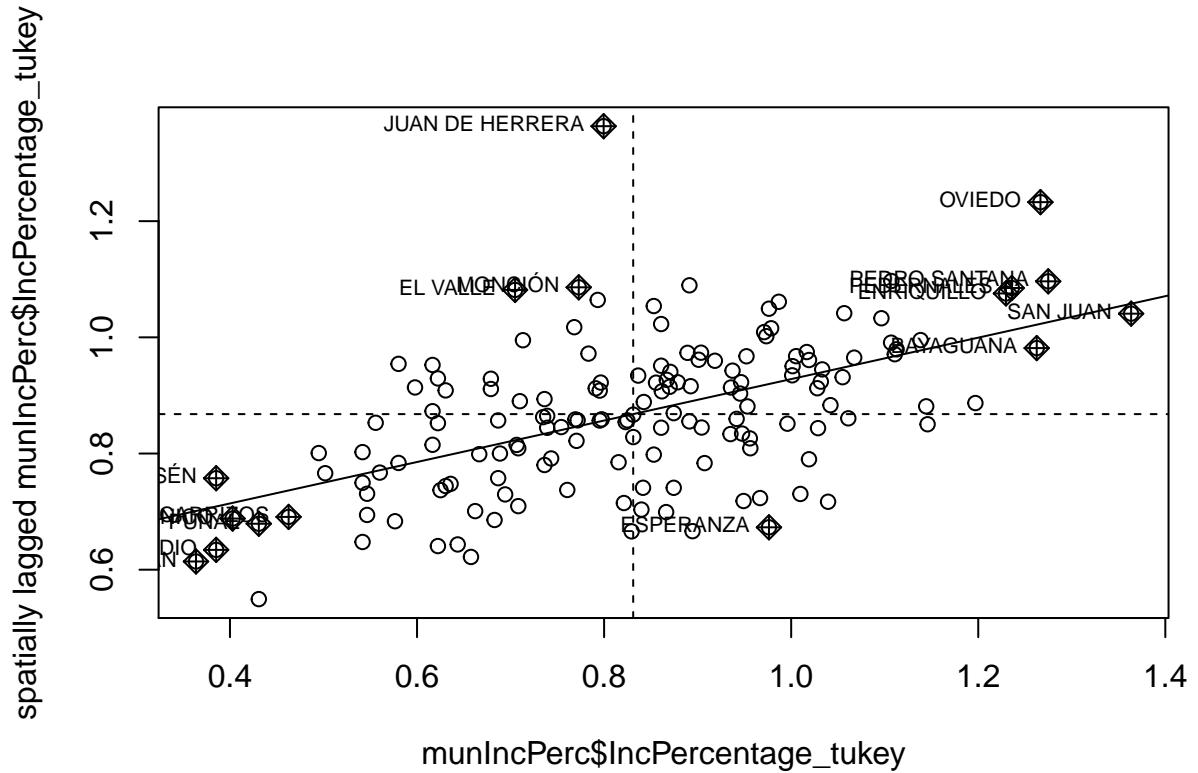
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##          0.439964128     -0.006493506    0.002656854

(gmoranb <- moran.test(x = munIncPerc$IncXArea_tukey, listw = munInc.w.B))

##
## Moran I test under randomisation
##
## data: munIncPerc$IncXArea_tukey
## weights: munInc.w.B
##
## Moran I statistic standard deviate = 8.9868, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##          0.431838654     -0.006493506    0.002379007

moran.plot(x = munIncPerc$IncPercentage_tukey, listw = munInc.w.W)

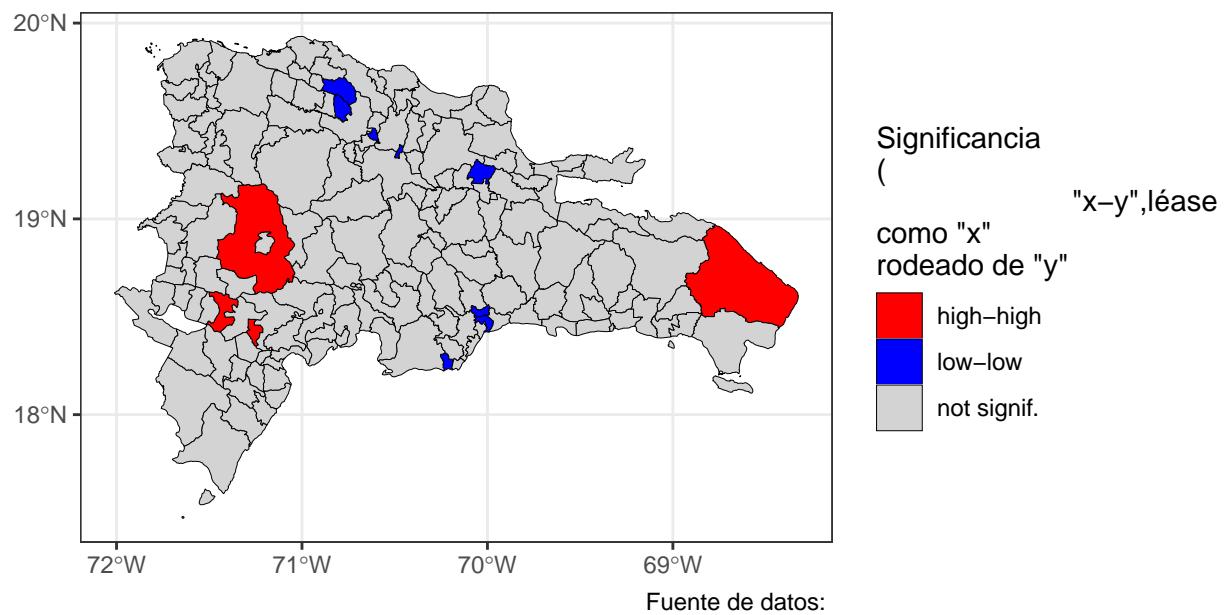
```



Finalmente se realiza el mapa de LISA Cluster para la variable *Porcentaje de incendios Tukey* arrojando el siguiente resultado que permite observar que los municipios con correlación espacial para los incendios son San Juan, Higuey y Neiba.

```
## $grafico
```

## Clusters LISA de Porcentaje de Incendios Forestales (Tukey T)



```
##  

## $objeto  

## Simple feature collection with 155 features and 20 fields  

## geometry type:  MULTIPOLYGON  

## dimension:      XY  

## bbox:            xmin: -72.01147 ymin: 17.47033 xmax: -68.32354 ymax: 19.93211  

## epsg (SRID):   4326  

## proj4string:    +proj=longlat +datum=WGS84 +no_defs  

## First 10 features:  

##   PROV MUN REG          TOPONIMIA ENLACE NumIncendios IncPercentage  

## 1 01 01 10 SANTO DOMINGO DE GUZMÁN 100101 3 0.006364562  

## 2 02 01 05 AZUA 050201 518 1.098947726  

## 3 02 02 05 LAS CHARCAS 050202 68 0.144263408  

## 4 02 03 05 LAS YAYAS DE VIAJAMA 050203 412 0.874066531  

## 5 02 04 05 PADRE LAS CASAS 050204 621 1.317464358  

## 6 02 05 05 PERALTA 050205 262 0.555838425  

## 7 02 06 05 SABANA YEGUA 050206 72 0.152749491  

## 8 02 07 05 PUEBLO VIEJO 050207 25 0.053038018  

## 9 02 08 05 TÁBARA ARRIBA 050208 462 0.980142566  

## 10 02 09 05 GUAYABAL 050209 369 0.782841141  

##   IncPercentage_log IncPercentage_tukey IncPercentage_tukey_lambda  

## 1 0.006344394 0.3637087 0.2  

## 2 0.741436136 1.0190498 0.2  

## 3 0.134761118 0.6789398 0.2  

## 4 0.628110685 0.9734394 0.2  

## 5 0.840473639 1.0566904 0.2  

## 6 0.442014580 0.8891801 0.2  

## 7 0.142149950 0.6867457 0.2  

## 8 0.051679337 0.5557986 0.2
```

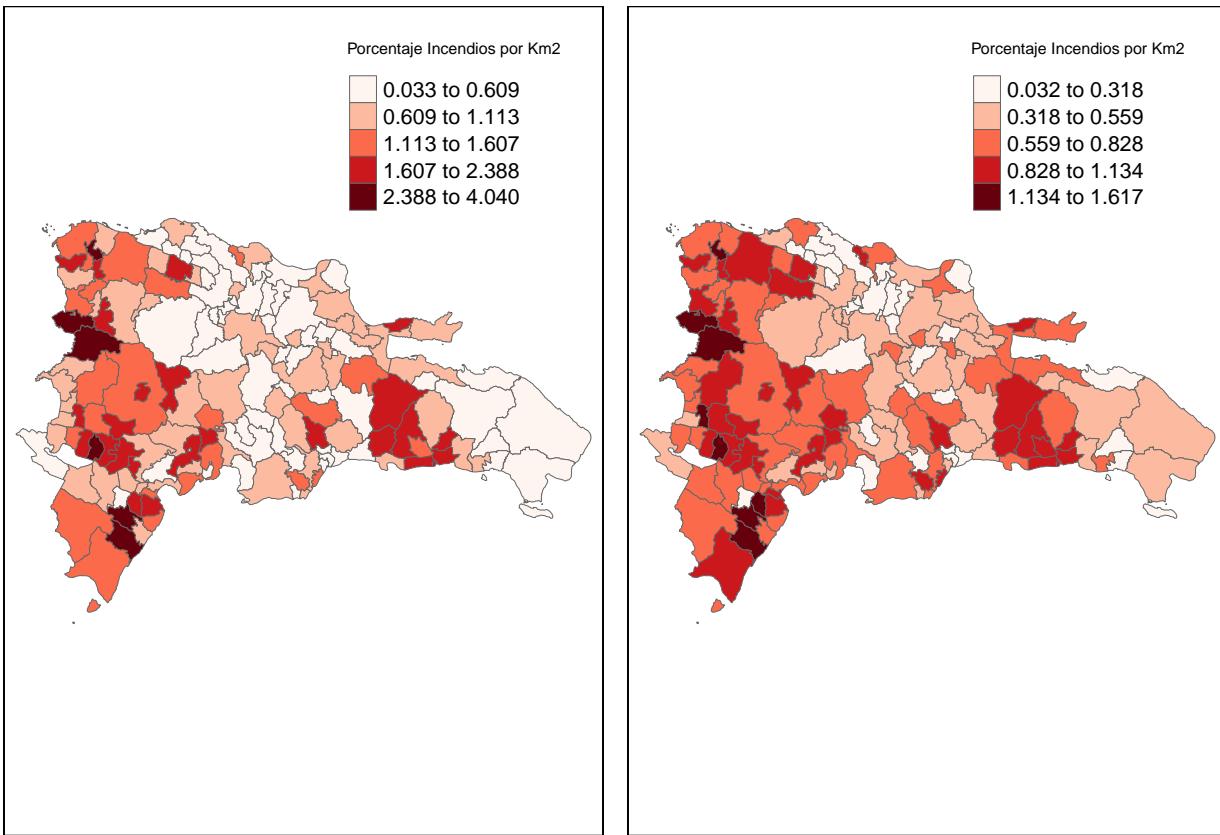
```

## 9      0.683168845      0.9959966      0.2
## 10     0.578208238      0.9522144      0.2
##   AreaKm2  IncXArea  IncXArea_log  IncXArea_tukey  IncXArea_tukey_lambda
## 1  91.49644 0.03278816    0.0322621    0.3586866      0.3
## 2  416.24047 1.24447292    0.8084707    1.0678141      0.3
## 3  246.53555 0.27582229    0.2435909    0.6794990      0.3
## 4  430.81968 0.95631656    0.6710634    0.9866895      0.3
## 5  573.56072 1.08271013    0.7336700    1.0241266      0.3
## 6  129.35998 2.02535594    1.1070288    1.2358063      0.3
## 7  113.90367 0.63211309    0.4898756    0.8714419      0.3
## 8  48.08024 0.51996408    0.4186867    0.8218489      0.3
## 9  274.86478 1.68082650    0.9861251    1.1685757      0.3
## 10 235.34938 1.56788174    0.9430813    1.1444425      0.3
##           x          y                  geometry puntuacionz
## 1 -69.94175 18.48488 MULTIPOLYGON (((-69.89794 1... -2.3235747
## 2 -70.80988 18.42093 MULTIPOLYGON (((-70.71457 1... 0.9344148
## 3 -70.54611 18.39184 MULTIPOLYGON (((-70.50185 1... -0.7564222
## 4 -70.99966 18.60405 MULTIPOLYGON (((-70.85774 1... 0.7076652
## 5 -70.91135 18.80900 MULTIPOLYGON (((-70.77551 1... 1.1215430
## 6 -70.78275 18.60198 MULTIPOLYGON (((-70.73131 1... 0.2887751
## 7 -70.88809 18.41374 MULTIPOLYGON (((-70.83014 1... -0.7176153
## 8 -70.78029 18.39309 MULTIPOLYGON (((-70.79387 1... -1.3686111
## 9 -70.91510 18.49605 MULTIPOLYGON (((-70.83352 1... 0.8198072
## 10 -70.76497 18.71745 MULTIPOLYGON (((-70.68664 1... 0.6021464
##   lagpuntuacionz quad_sig
## 1      0.43036993 not signif.
## 2     -0.76905286 not signif.
## 3      0.20455192 not signif.
## 4     -1.00787376 not signif.
## 5      0.07620910 not signif.
## 6     -0.04071622 not signif.
## 7     -0.16061258 not signif.
## 8      0.42293128 not signif.
## 9     -0.03986659 not signif.
## 10     0.39438601 not signif.

```

Para efectos del estudio se repite el proceso descrito anteriormente para las mismas variables sin embargo teniendo en cuenta el “área” de los municipios esto debido a los resultados vistos en el LISA Cluster.

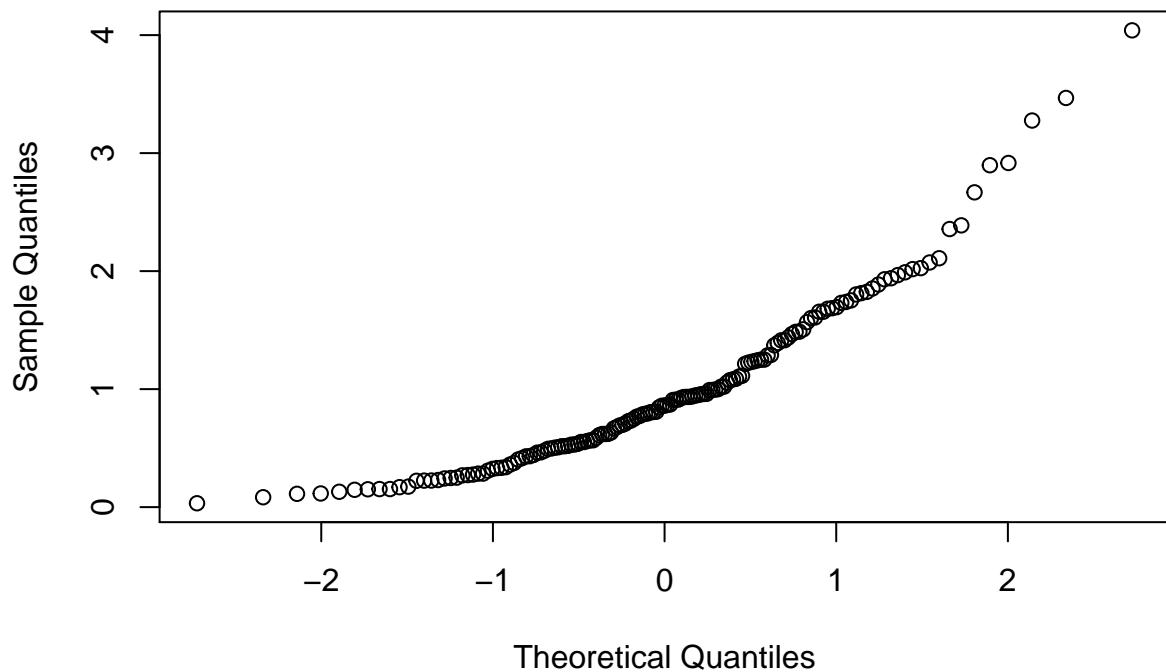
Teniendo en cuenta que los mapas mostrados a continuación se enfocan en la cantidad de incendios por km<sup>2</sup> se puede observar que los municipios con mayor cantidad de incendios son los municipios fronterizos de orografía pronunciada.



Al realizar las pruebas de “Shapiro-Wilk” para las columnas *Porcentaje de Incendios x Area*, *Log Porcentaje de Incendios x Area* y *Tukey Porcentaje de Incendios x Area* se observa que tan solo la ultima tiene valores p mayores a 0.05 por lo que no permite su rechazo y se considera el supuesto de normalidad. Al realizar la prueba “Breusch-Pagan” para cada una de estas todas las variables tienen resultados que pueden ser interpretados con una hipótesis nula que no puede ser rechazada y con presunción de homocedasticidad esto debido a que todas superan el p mayor a 0.05.

```
#qq
qqnorm(munIncPerc$IncXArea)
```

## Normal Q-Q Plot

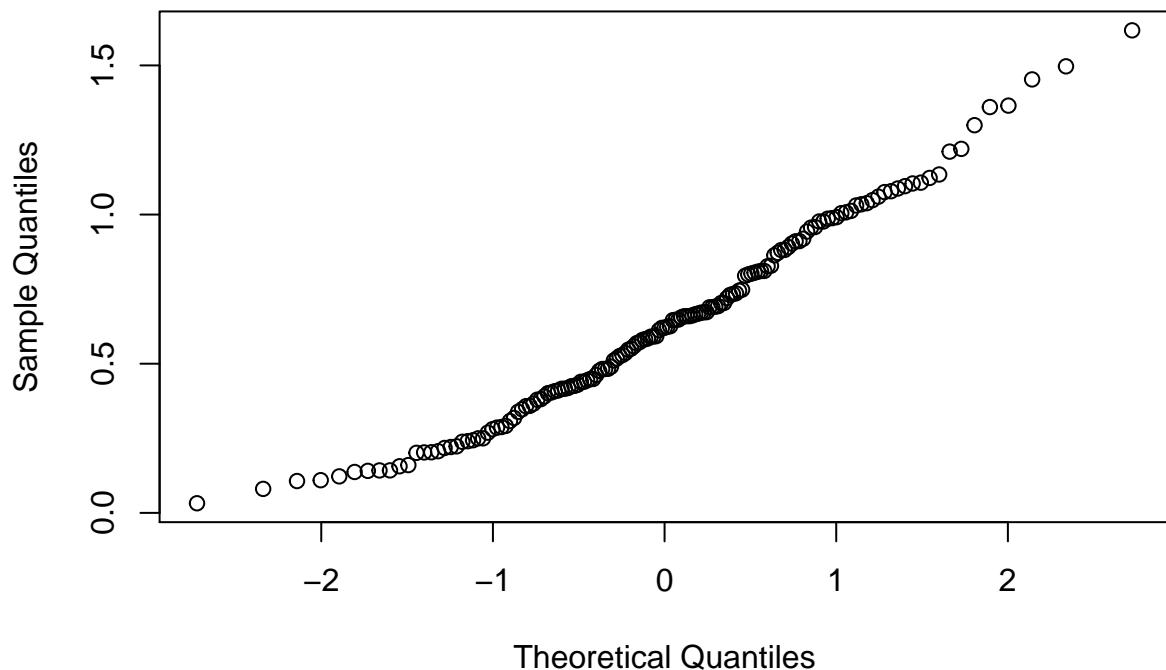


```
shapiro.test(munIncPerc$IncXArea)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: munIncPerc$IncXArea  
## W = 0.89846, p-value = 7.076e-09
```

```
qqnorm(munIncPerc$IncXArea_log)
```

## Normal Q-Q Plot

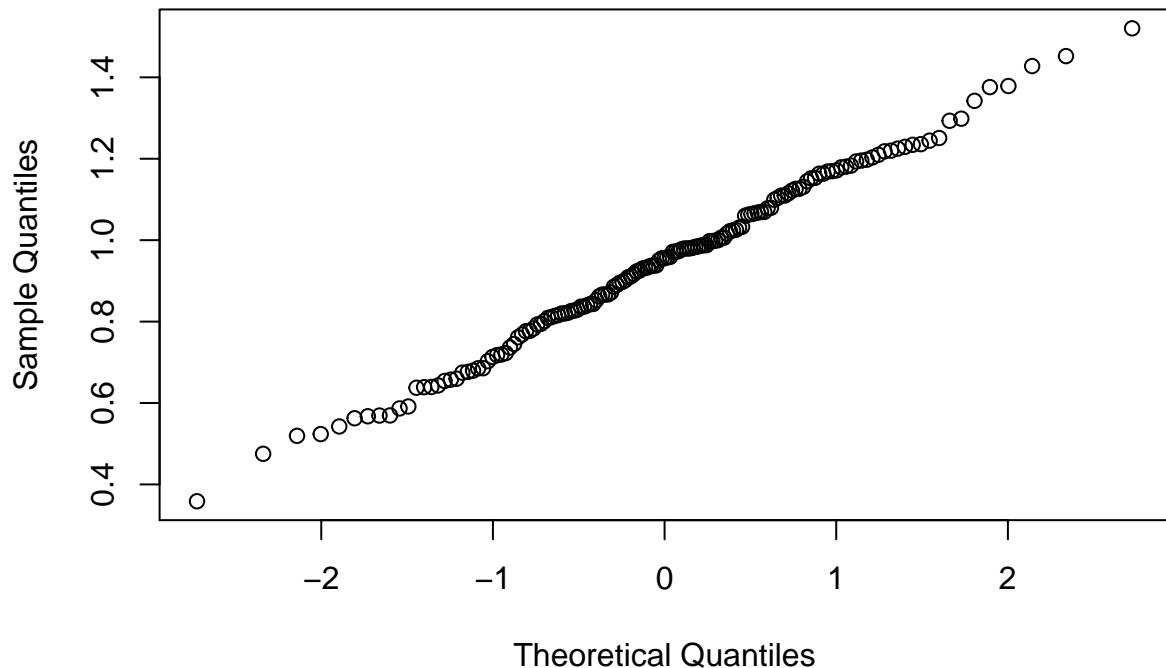


```
shapiro.test(munIncPerc$IncXArea_log)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: munIncPerc$IncXArea_log  
## W = 0.9773, p-value = 0.01161
```

```
qqnorm(munIncPerc$IncXArea_tukey)
```

## Normal Q-Q Plot



```
shapiro.test(munIncPerc$IncXArea_tukey)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: munIncPerc$IncXArea_tukey  
## W = 0.99557, p-value = 0.9253
```

```
munIncPerc %>% lm(IncXArea ~ x, .) %>% bptest()
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: .  
## BP = 3.3994, df = 1, p-value = 0.06522
```

```
munIncPerc %>% lm(IncXArea ~ y, .) %>% bptest()
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: .  
## BP = 0.41244, df = 1, p-value = 0.5207
```

```
munIncPerc %>% lm(IncXArea_log ~ x, .) %>% bptest()
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: .  
## BP = 0.19948, df = 1, p-value = 0.6551
```

```
munIncPerc %>% lm(IncXArea_log ~ y, .) %>% bptest()
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: .  
## BP = 0.0796, df = 1, p-value = 0.7778
```

```
munIncPerc %>% lm(IncXArea_tukey ~ x, .) %>% bptest()
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: .  
## BP = 0.18554, df = 1, p-value = 0.6667
```

```
munIncPerc %>% lm(IncXArea_tukey ~ y, .) %>% bptest()
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: .  
## BP = 0.66037, df = 1, p-value = 0.4164
```

Teniendo en cuenta los resultados anteriores, se evalua la *autocorrelación espacial global* teniendo en cuenta la prueba para la “I” de Moran usando los pesos estandarizados y los binarios para la variable de *Tukey* estableciendo que las comprobaciones son menores al alpha estipulado permitiendo afirmar al estudio que existe una autocorrelación espacial.

```
(gmoranw <- moran.test(x = munIncPerc$IncXArea_tukey, listw = munInc.w.W ))
```

```
##  
## Moran I test under randomisation  
##  
## data: munIncPerc$IncXArea_tukey  
## weights: munInc.w.W  
##  
## Moran I statistic standard deviate = 8.6616, p-value < 2.2e-16
```

```

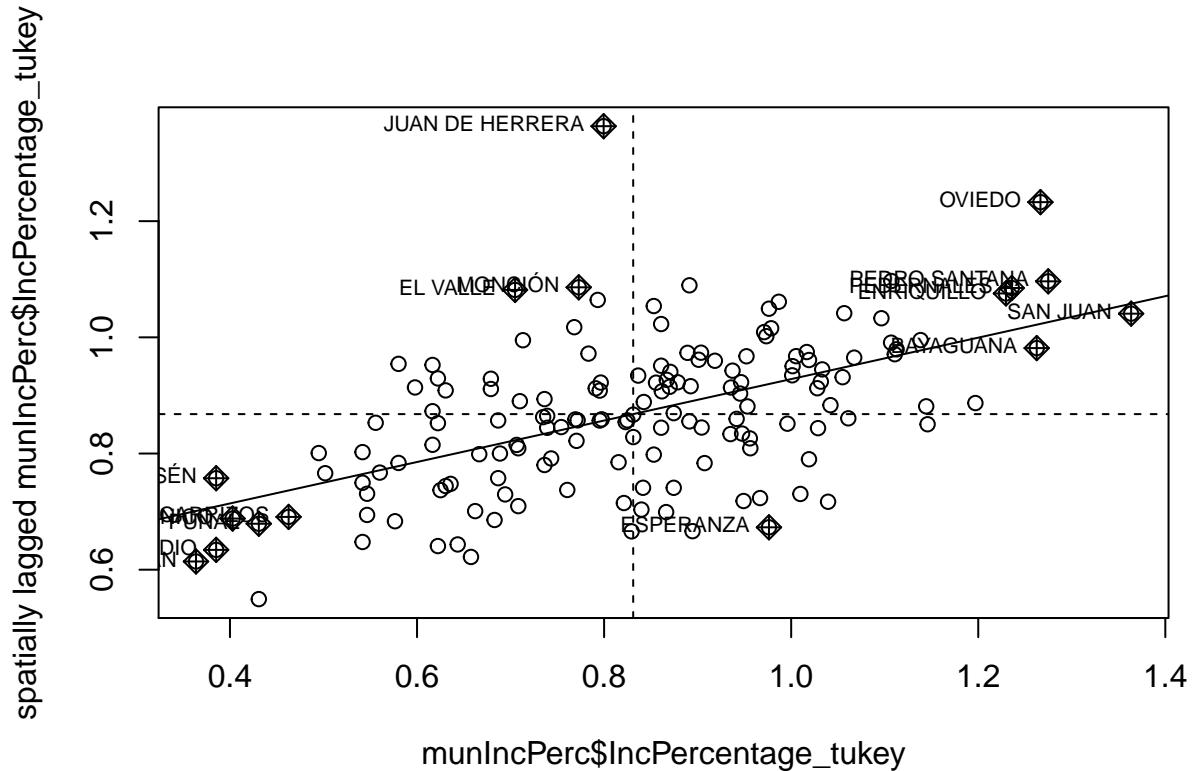
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##          0.439964128     -0.006493506    0.002656854

(gmoranb <- moran.test(x = munIncPerc$IncXArea_tukey, listw = munInc.w.B))

##
## Moran I test under randomisation
##
## data: munIncPerc$IncXArea_tukey
## weights: munInc.w.B
##
## Moran I statistic standard deviate = 8.9868, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##          0.431838654     -0.006493506    0.002379007

moran.plot(x = munIncPerc$IncPercentage_tukey, listw = munInc.w.W)

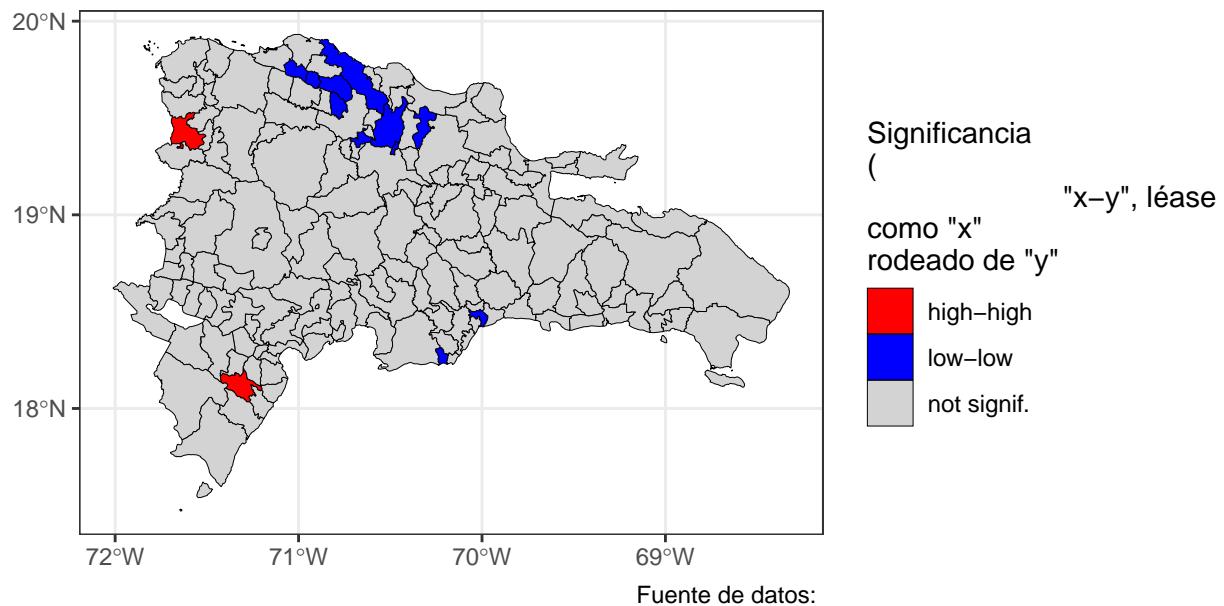
```



Finalmente se realiza el mapa de LISA Cluster para la variable *Porcentaje de incendios x Area Tukey* arrojando el siguiente resultado que permite observar que los municipios con correlación espacial para los incendios teniendo en cuenta su área son Loma de Cabrera y Polo.

```
## $grafico
```

## Clusters LISA de Porcentaje de Incendios Forestales x Area (



```
##  
## $objeto  
## Simple feature collection with 155 features and 20 fields  
## geometry type:  MULTIPOLYGON  
## dimension:      XY  
## bbox:            xmin: -72.01147 ymin: 17.47033 xmax: -68.32354 ymax: 19.93211  
## epsg (SRID):   4326  
## proj4string:    +proj=longlat +datum=WGS84 +no_defs  
## First 10 features:  
##   PROV MUN REG          TOPONIMIA ENLACE NumIncendios IncPercentage  
## 1 01 01 10 SANTO DOMINGO DE GUZMÁN 100101 3 0.006364562  
## 2 02 01 05 AZUA 050201 518 1.098947726  
## 3 02 02 05 LAS CHARCAS 050202 68 0.144263408  
## 4 02 03 05 LAS YAYAS DE VIAJAMA 050203 412 0.874066531  
## 5 02 04 05 PADRE LAS CASAS 050204 621 1.317464358  
## 6 02 05 05 PERALTA 050205 262 0.555838425  
## 7 02 06 05 SABANA YEGUA 050206 72 0.152749491  
## 8 02 07 05 PUEBLO VIEJO 050207 25 0.053038018  
## 9 02 08 05 TÁBARA ARRIBA 050208 462 0.980142566  
## 10 02 09 05 GUAYABAL 050209 369 0.782841141  
##   IncPercentage_log IncPercentage_tukey IncPercentage_tukey_lambda  
## 1 0.006344394 0.3637087 0.2  
## 2 0.741436136 1.0190498 0.2  
## 3 0.134761118 0.6789398 0.2  
## 4 0.628110685 0.9734394 0.2  
## 5 0.840473639 1.0566904 0.2  
## 6 0.442014580 0.8891801 0.2  
## 7 0.142149950 0.6867457 0.2  
## 8 0.051679337 0.5557986 0.2
```

```

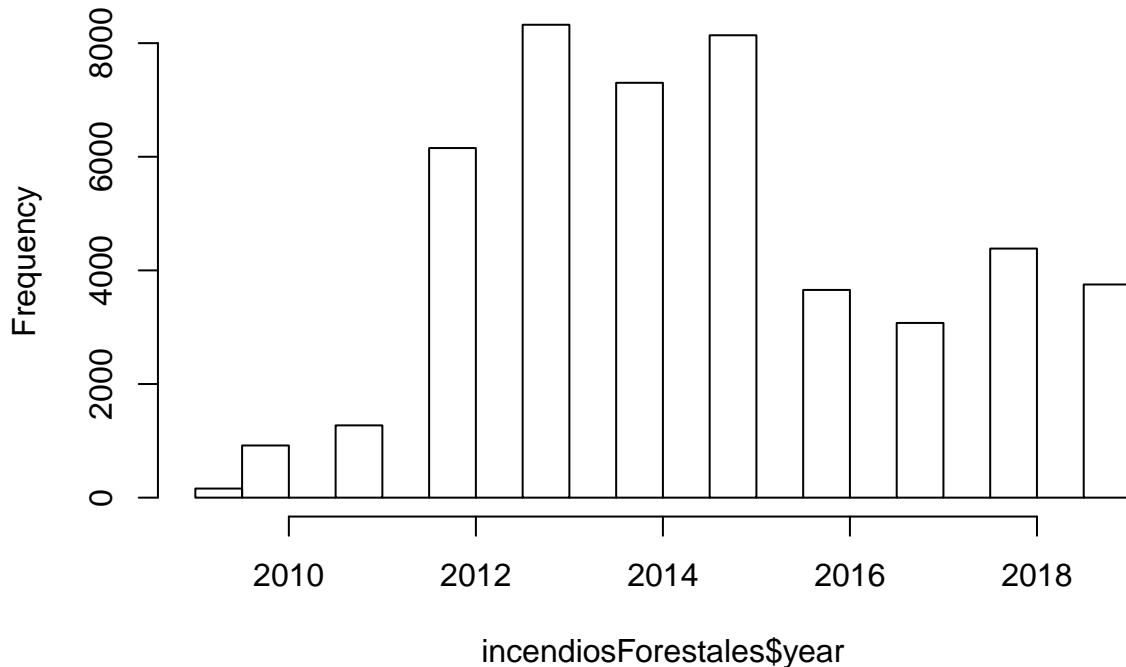
## 9      0.683168845      0.9959966      0.2
## 10     0.578208238      0.9522144      0.2
##   AreaKm2  IncXArea  IncXArea_log  IncXArea_tukey  IncXArea_tukey_lambda
## 1  91.49644 0.03278816    0.0322621    0.3586866      0.3
## 2  416.24047 1.24447292    0.8084707    1.0678141      0.3
## 3  246.53555 0.27582229    0.2435909    0.6794990      0.3
## 4  430.81968 0.95631656    0.6710634    0.9866895      0.3
## 5  573.56072 1.08271013    0.7336700    1.0241266      0.3
## 6  129.35998 2.02535594    1.1070288    1.2358063      0.3
## 7  113.90367 0.63211309    0.4898756    0.8714419      0.3
## 8  48.08024 0.51996408    0.4186867    0.8218489      0.3
## 9  274.86478 1.68082650    0.9861251    1.1685757      0.3
## 10 235.34938 1.56788174    0.9430813    1.1444425      0.3
##           x          y                  geometry puntuacionz
## 1 -69.94175 18.48488 MULTIPOLYGON (((-69.89794 1... -2.6897115
## 2 -70.80988 18.42093 MULTIPOLYGON (((-70.71457 1... 0.5638173
## 3 -70.54611 18.39184 MULTIPOLYGON (((-70.50185 1... -1.2178007
## 4 -70.99966 18.60405 MULTIPOLYGON (((-70.85774 1... 0.1916118
## 5 -70.91135 18.80900 MULTIPOLYGON (((-70.77551 1... 0.3633762
## 6 -70.78275 18.60198 MULTIPOLYGON (((-70.73131 1... 1.3345780
## 7 -70.88809 18.41374 MULTIPOLYGON (((-70.83014 1... -0.3371526
## 8 -70.78029 18.39309 MULTIPOLYGON (((-70.79387 1... -0.5646887
## 9 -70.91510 18.49605 MULTIPOLYGON (((-70.83352 1... 1.0261192
## 10 -70.76497 18.71745 MULTIPOLYGON (((-70.68664 1... 0.9153943
##   lagpuntuacionz    quad_sig
## 1      0.56769173 not signif.
## 2      -0.17377395 not signif.
## 3       0.03623403 not signif.
## 4      -1.24703680 not signif.
## 5       0.34551910 not signif.
## 6       0.28475890 not signif.
## 7       0.48035889 not signif.
## 8       0.66046303 not signif.
## 9       0.19931002 not signif.
## 10      0.52734478 not signif.

```

## 2.4 correlación variables world clim

Para la segunda parte de esta investigación, se añaden las variables de Version2 (2000) *Precipitación Viento Temperatura Promedio y Radiación Solar*, a la información de los Incendios Forestales así como las columnas de *mes*, *año* y *Id unico*. Cabe notar que la data de "WorldClim" se establece basandose un promedio mensual en el periodo 1970-2000

### Histogram of incendiosForestales\$year



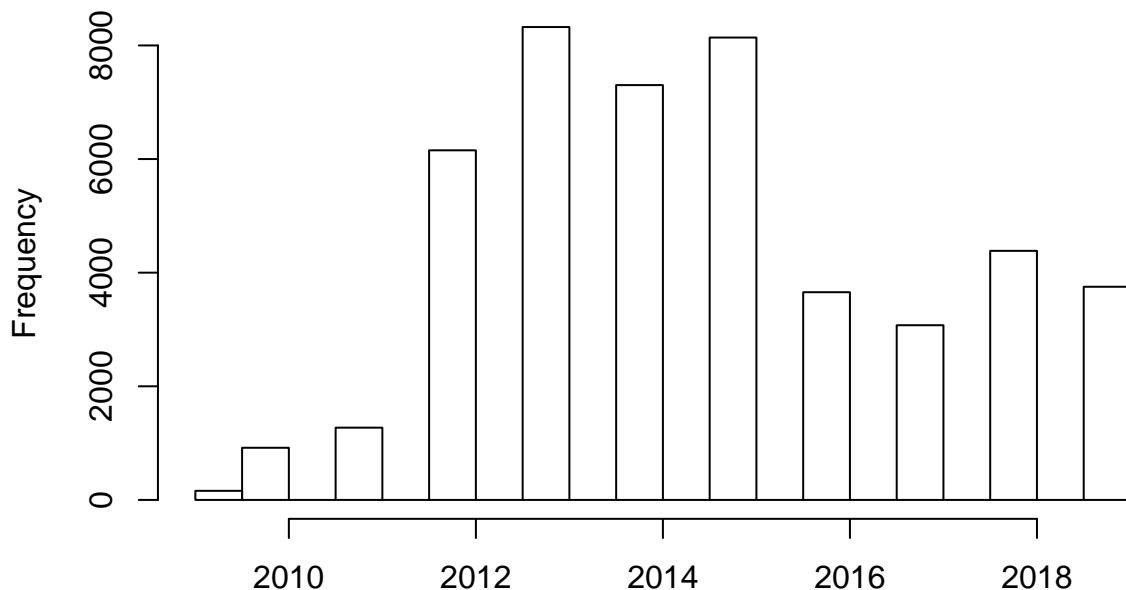
Martínez Batlle (2019b) realizó el script de adición de data basandose en la localización del incendio y el mes de adquisición por parte de la NASA por cada una de las variables a estudiar dando como resultado el archivo `puntos_calor_fuegos_y_worldclim_viento_temp_radiacion`. Sin embargo este script necesita de una unidad de computo de altas prestaciones por lo que es recomendable no realizar dicho procedimiento en equipos con menos de 64Gb de memoria RAM y multiples cores para su procesamiento.

Para efectos prácticos, se realizó el mismo procedimiento usando la herramienta QGis mediante el plugin `point sampling tool` estableciendo el valor de todas las variables climáticas para cada incendio, el *mes* y *ID unico* obteniendo una tabla que fue exportada con formato “CSV”. La misma se abre en el software “Openoffice Calc” y por medio la función “Buscar” para la matriz estableciendo el valor para cada incendio tomando en cuenta el valor del mes.

Con el resultado anterior se realiza un *left join* con la tabla inicial de incendios forestales usando como punto de enlace el valor “ID Unico” Posteriormente se quitan los valores que obtuvieron como valor 0 debido a que estos son errores puesto que se conoce que en el país no existe ese valor para ninguna de las variables.

Teniendo en cuenta el siguiente gráfico se decide un valor promedio de basandose en el año, puesto que el manejo de todos los datos se torna complicado con una máquina con prestaciones insuficientes ademas para poder realizar las pruebas “Sapiro-Wilk” las cuales no se pueden realizar con mas de 5000 datos.

## Histogram of incendiosForestales\$year



Basandose en lo descrito anteriormente y se establece el 2018 como el año con el que se puede realizar una mejor estimación ademas cuenta con 4374 datos.

Se obtienen los datos estadísticos para el año 2018

```
#Año 2018
#EDA
nrow(incForWC2018)

## [1] 4374

summary(incForWC2018$prec)

##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##    11.00   55.00  89.00    95.05 129.00  298.00

summary(incForWC2018$wind)

##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##    1.304   2.100  2.460    2.625  3.088   4.600

summary(incForWC2018$srad)

##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##  12794   18651  20113   19446  20914   22359
```

```
summary(incForWC2018$temp)
```

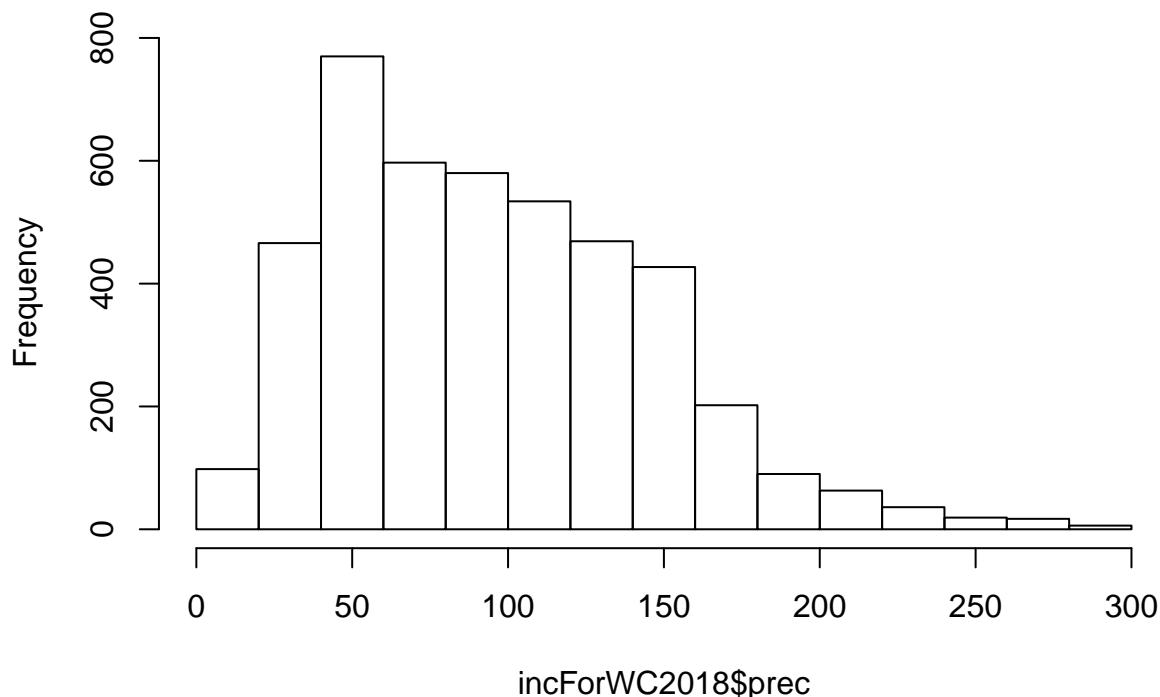
```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##    11.33   22.72   24.90   24.34   26.65   29.42
```

Se realiza la prueba “Shapiro-Wilk” para las cuatro variables y su logaritmo sin embargo se puede observar que ninguna pasa la prueba puesta que ninguna supera el valor de 0.05, sin embargo se realiza un ploteo informativo de la variable basado en la localización de los incendios.

```
#Temperatura
```

```
hist(incForWC2018$prec)
```

**Histogram of incForWC2018\$prec**



```
shapiro.test(incForWC2018$temp)
```

```
##
## Shapiro-Wilk normality test
##
## data: incForWC2018$temp
## W = 0.93385, p-value < 2.2e-16
```

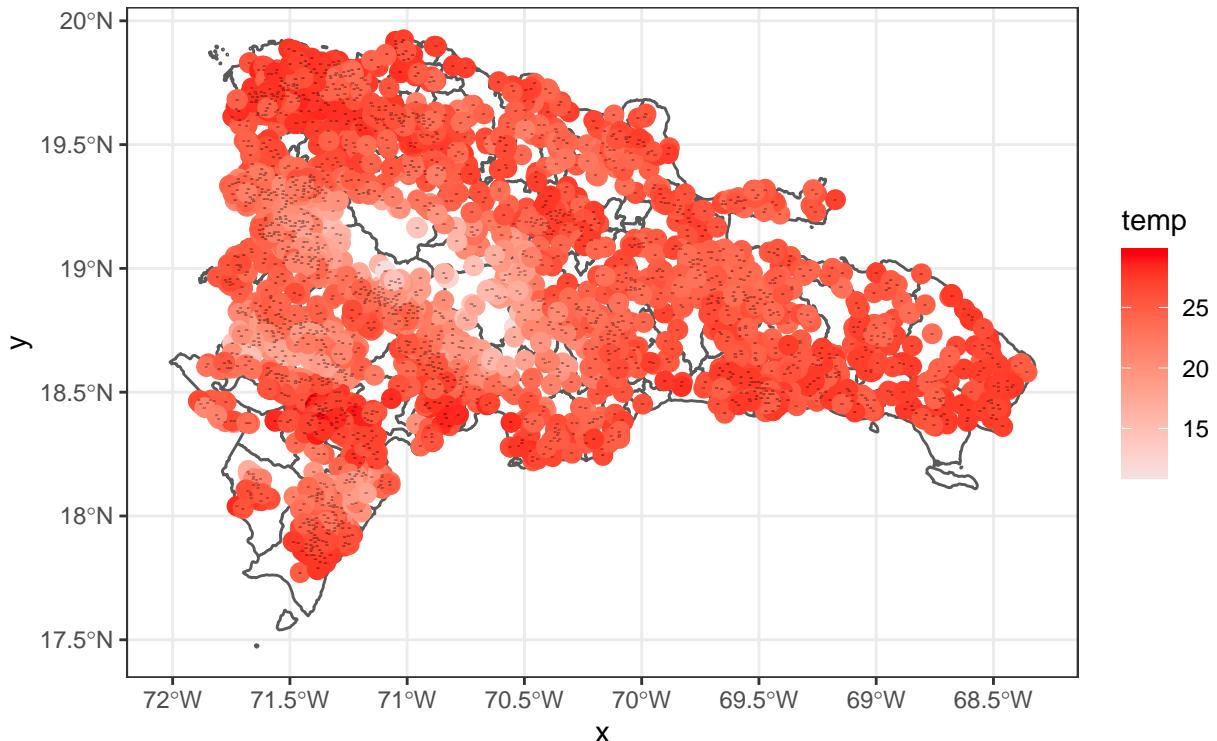
```
shapiro.test(log(incForWC2018$temp))
```

```
##
## Shapiro-Wilk normality test
##
## data: log(incForWC2018$temp)
## W = 0.89088, p-value < 2.2e-16
```

```

ggplot() +
  geom_sf(data = mun4326, fill = 'white') +
  geom_sf(data = incForWC2018, aes(col = temp), size = 3) +
  scale_colour_gradient(low="#f7dede", high="#ff0011") +
  geom_sf_text(data = incForWC2018, aes(label=MUN),
               check_overlap = T, size = 0.5) +
  theme_bw()

```



```

#Radiacion Solar
shapiro.test(incForWC2018$srad)

```

```

##
## Shapiro-Wilk normality test
##
## data: incForWC2018$srad
## W = 0.82435, p-value < 2.2e-16

```

```

shapiro.test(log(incForWC2018$srad))

```

```

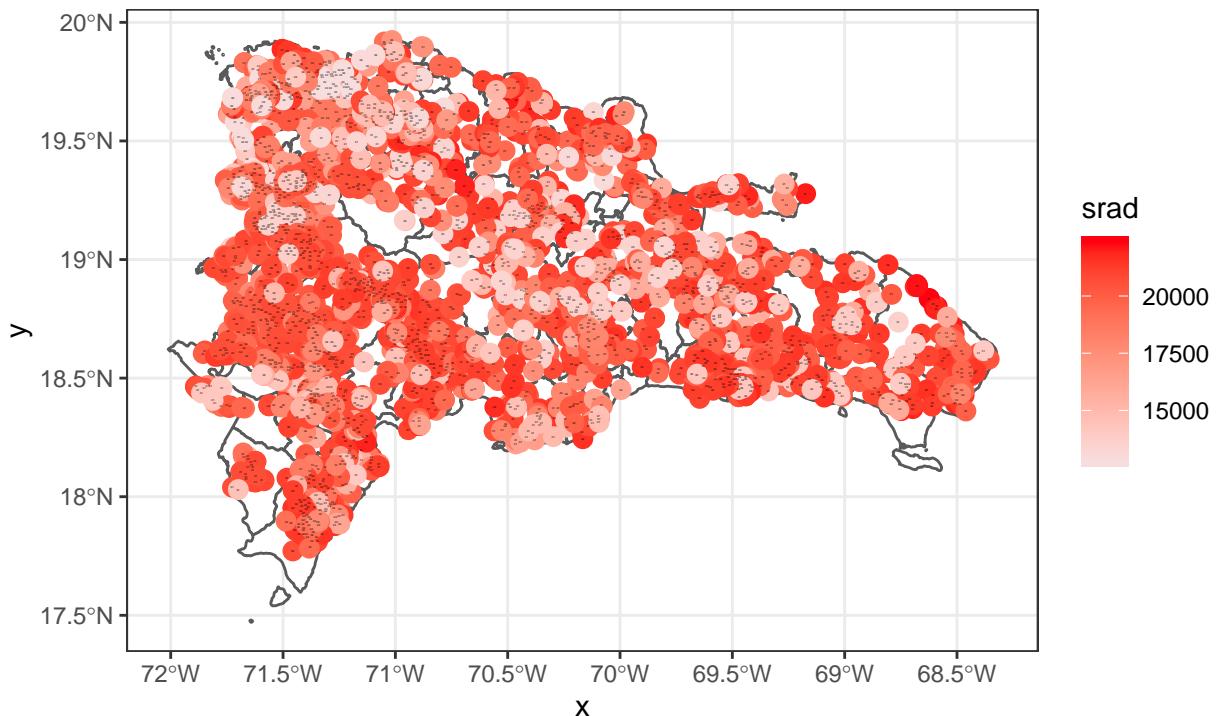
##
## Shapiro-Wilk normality test
##
## data: log(incForWC2018$srad)
## W = 0.78429, p-value < 2.2e-16

```

```

ggplot() +
  geom_sf(data = mun4326, fill = 'white') +
  geom_sf(data = incForWC2018, aes(col = srad), size = 3) +
  scale_colour_gradient(low="#f7dede", high="#ff0011") +
  geom_sf_text(data = incForWC2018, aes(label=MUN),
               check_overlap = T, size = 0.5) +
  theme_bw()

```



```

#Viento
shapiro.test(incForWC2018$wind)

```

```

##
## Shapiro-Wilk normality test
##
## data: incForWC2018$wind
## W = 0.94401, p-value < 2.2e-16

```

```

shapiro.test(log(incForWC2018$wind))

```

```

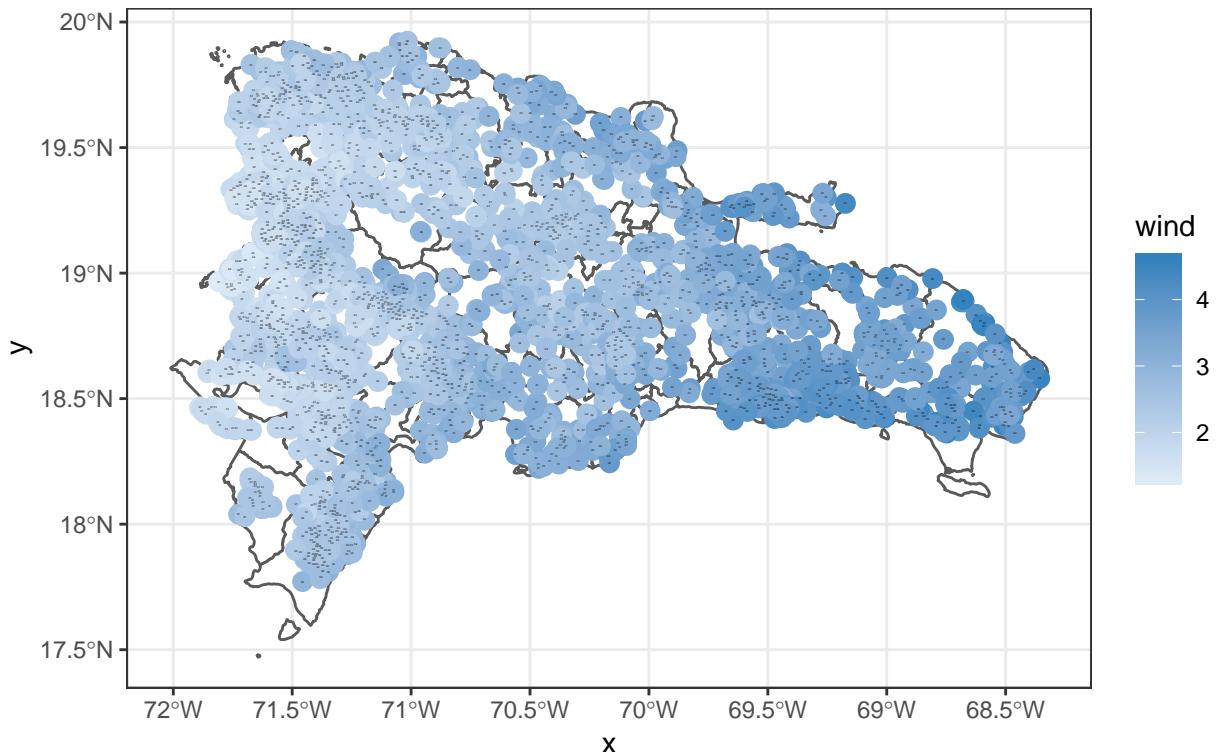
##
## Shapiro-Wilk normality test
##
## data: log(incForWC2018$wind)
## W = 0.97691, p-value < 2.2e-16

```

```

ggplot() +
  geom_sf(data = mun4326, fill = 'white') +
  geom_sf(data = incForWC2018, aes(col = wind), size = 3) +
  scale_colour_gradient(low="#deebf7", high="#3182bd") +
  geom_sf_text(data = incForWC2018, aes(label=MUN),
               check_overlap = T, size = 0.5) +
  theme_bw()

```



```

#Precipitacion
shapiro.test(incForWC2018$prec)

```

```

##
## Shapiro-Wilk normality test
##
## data: incForWC2018$prec
## W = 0.95828, p-value < 2.2e-16

```

```

shapiro.test(log(incForWC2018$prec))

```

```

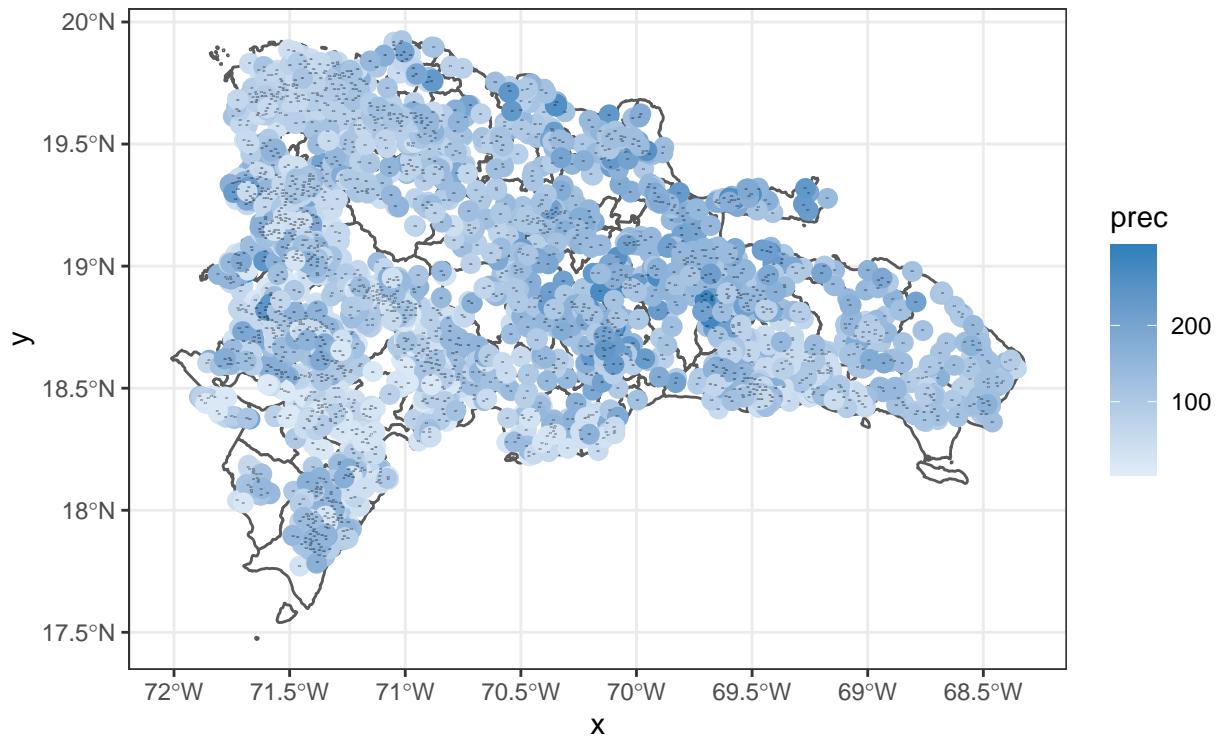
##
## Shapiro-Wilk normality test
##
## data: log(incForWC2018$prec)
## W = 0.97345, p-value < 2.2e-16

```

```

ggplot() +
  geom_sf(data = mun4326, fill = 'white') +
  geom_sf(data = incForWC2018, aes(col = prec), size = 3) +
  scale_colour_gradient(low="#deebf7", high="#3182bd") +
  geom_sf_text(data = incForWC2018, aes(label=MUN),
               check_overlap = T, size = 0.5) +
  theme_bw()

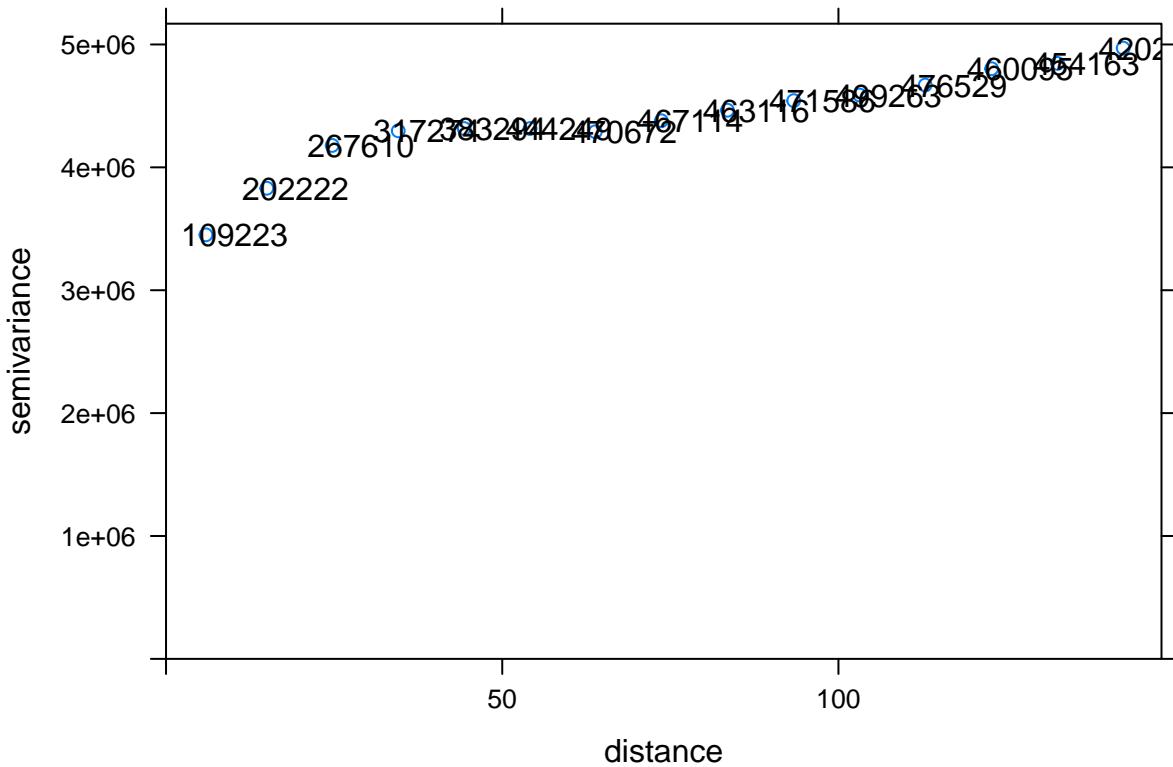
```



Se realiza un variograma del modelo para cada una de las variables y de esta manera establecer cual podría ser el que se utilizará para el desarrollo de la interpolación “krige”. A continuación se detalla para cada una de las variables.

Radiacion Solar

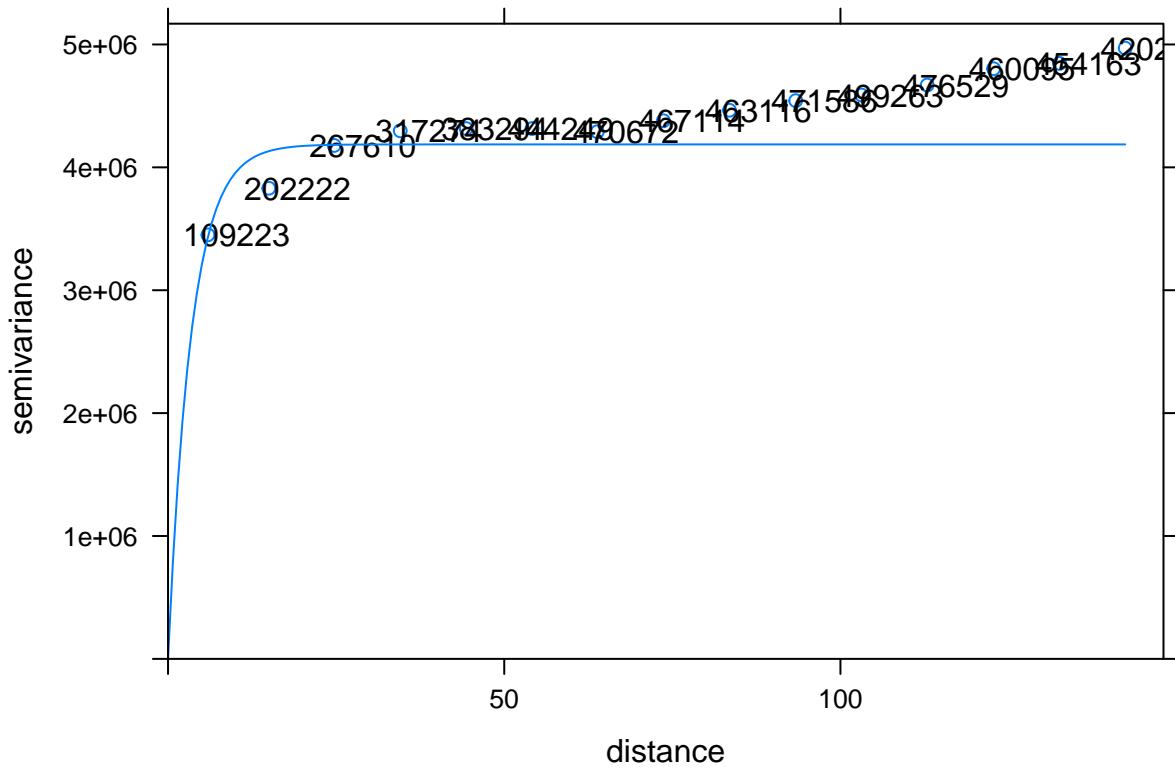
```
plot(vsrad18, plot.numbers = T)
```



```
vsrad18_m1
```

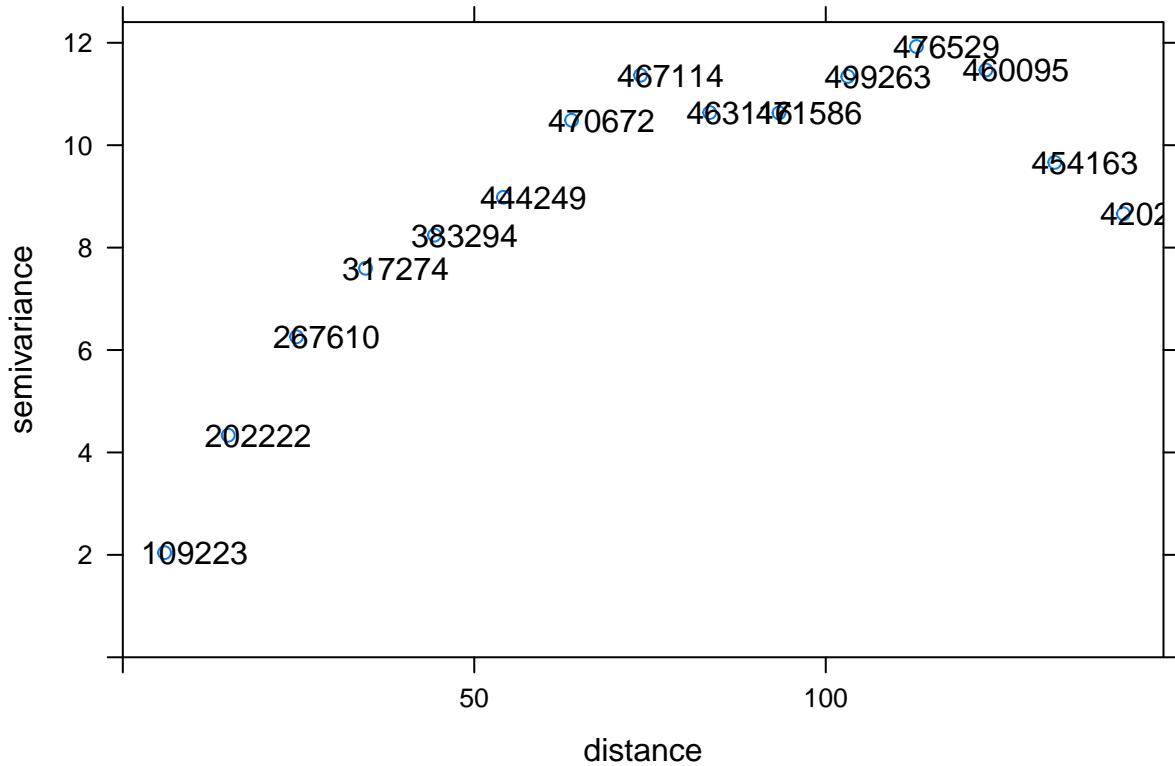
```
##   model   psill     range
## 1   Exp 4186621 3.456893
```

```
plot(vsrad18, vsrad18_m1, plot.numbers = T)
```



Temperatura

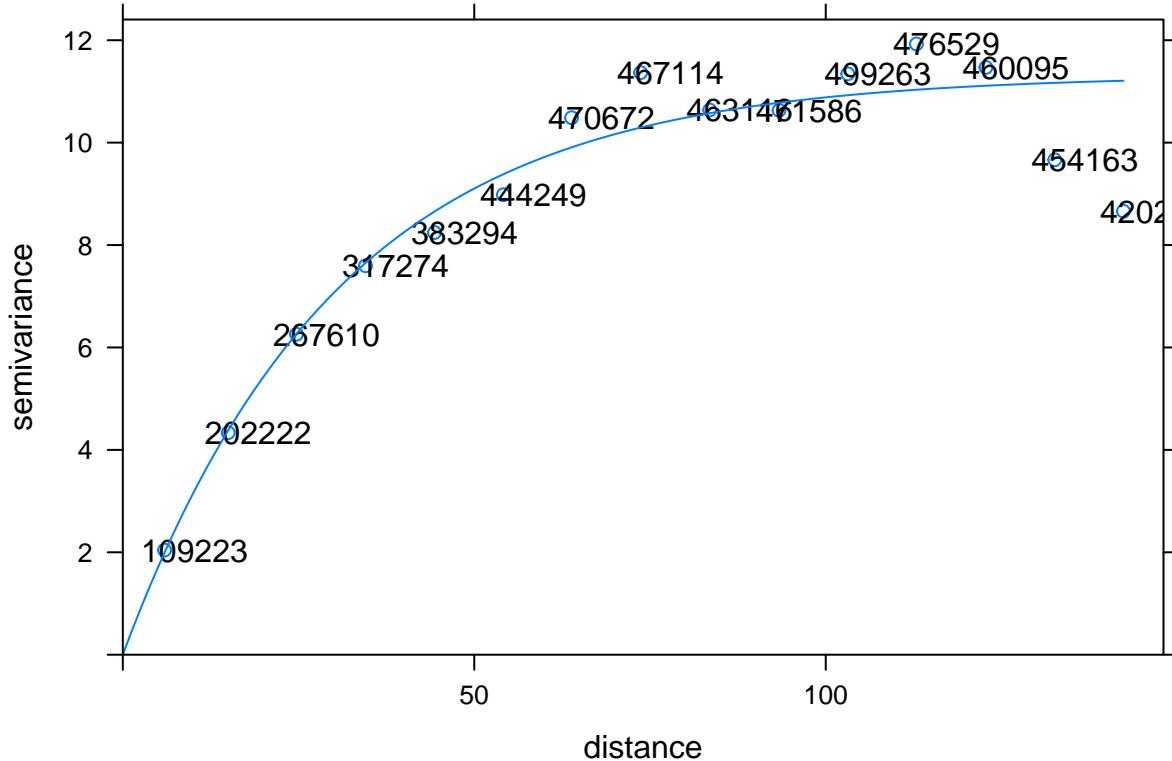
```
plot(vtemp18, plot.numbers = T)
```



```
vtemp18_m1
```

```
##   model    psill    range
## 1  Exp 11.31686 30.64284
```

```
plot(vtemp18, vtemp18_m1, plot.numbers = T)
```

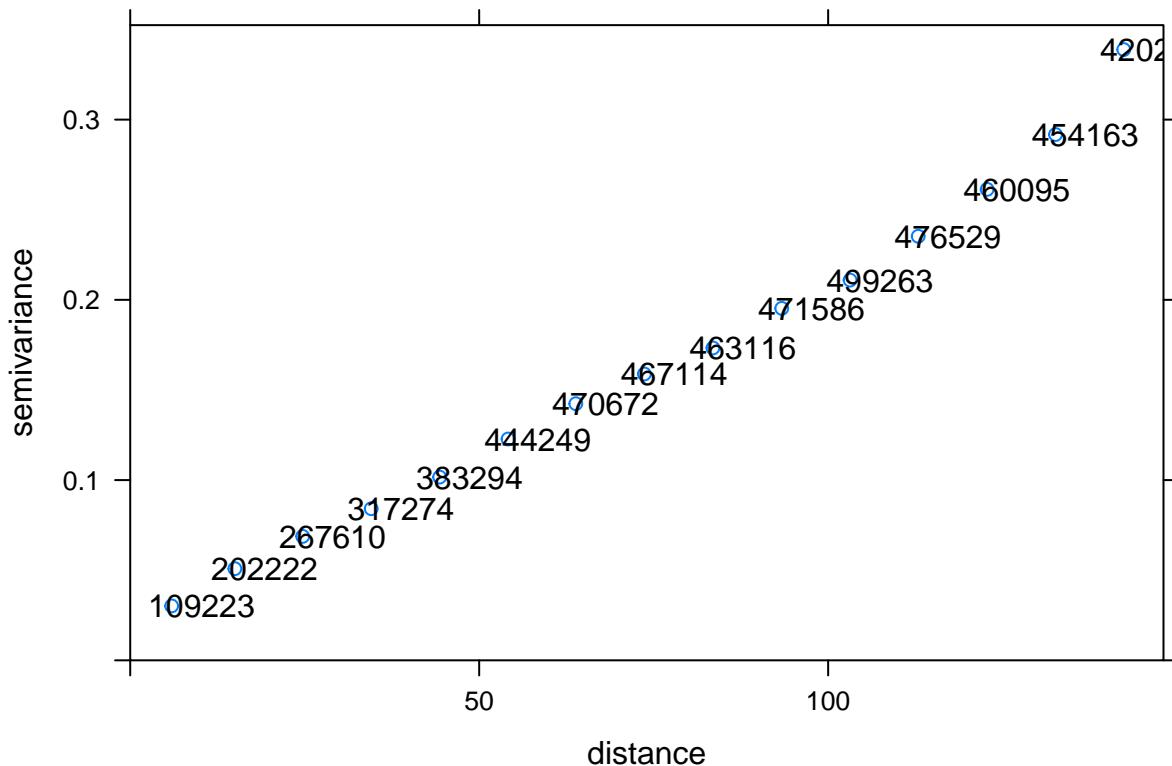


```
attr(vtemp18_m1, 'SSErr')
```

```
## [1] 441.3533
```

Viento

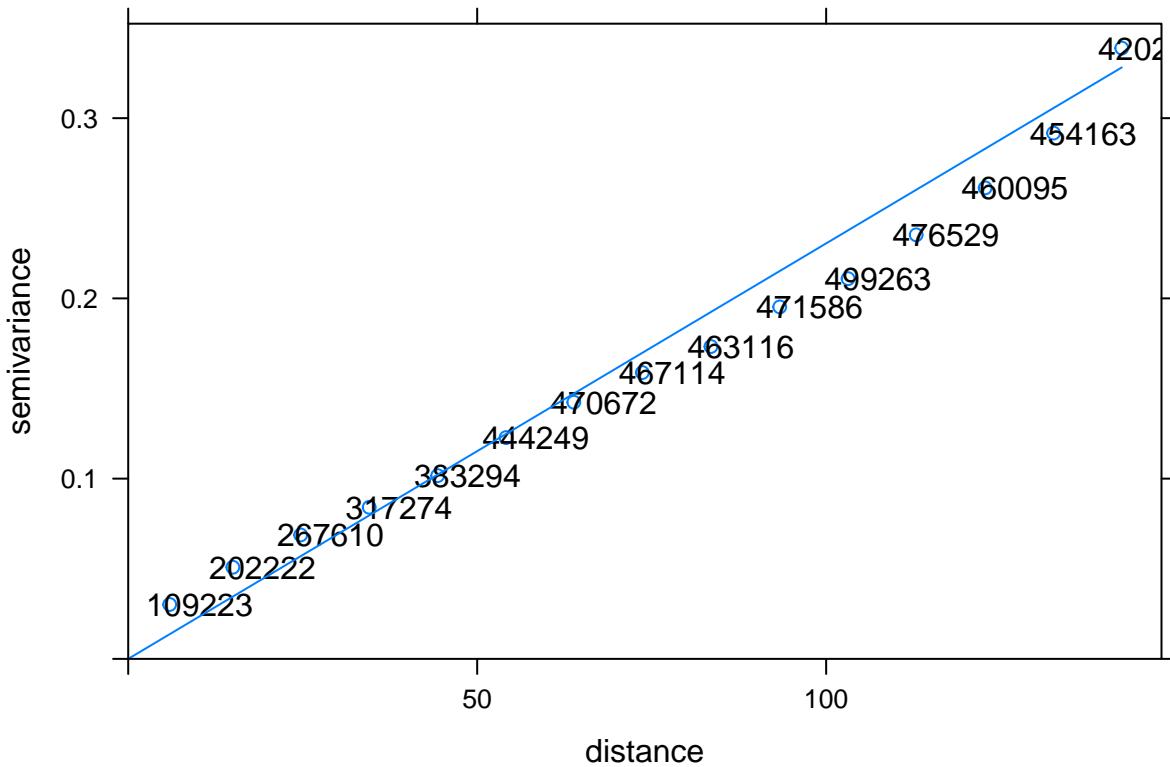
```
plot(vwind18, plot.numbers = T)
```



```
vwind18_m1
```

```
##   model      psill    range
## 1   Lin 0.3388659 147.004
```

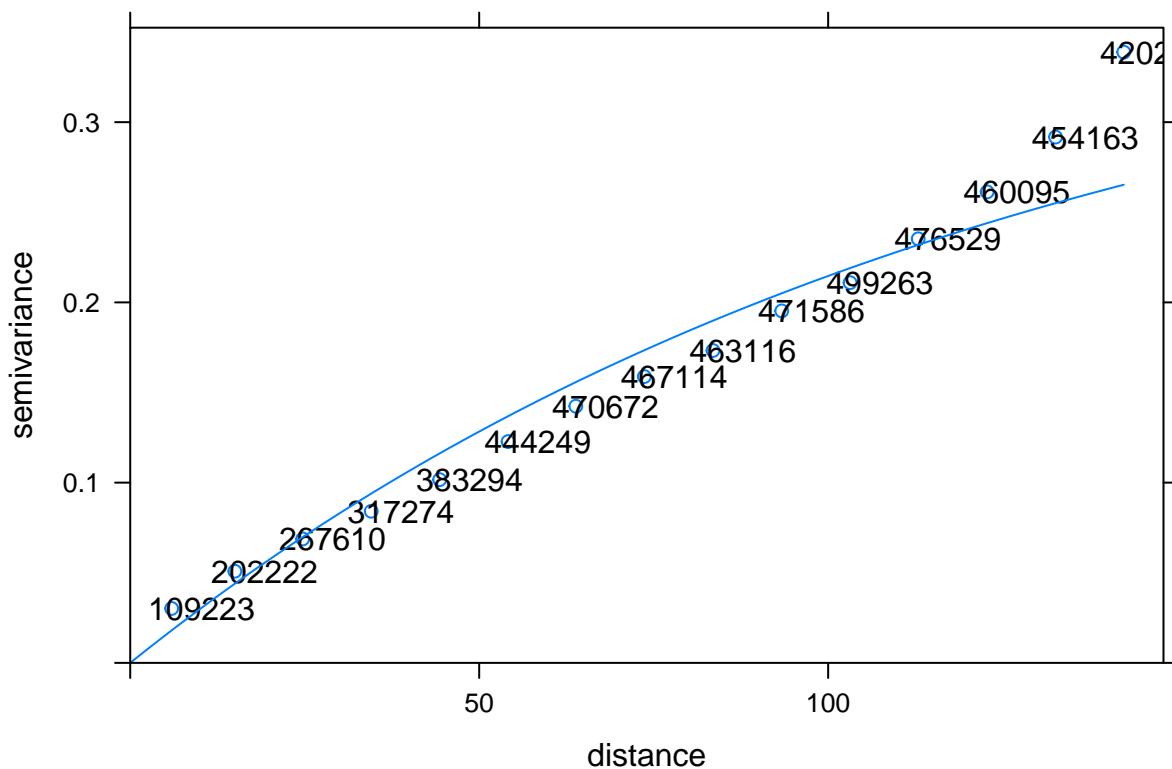
```
plot(vwind18, vwind18_m1, plot.numbers = T)
```



```
vwind18_m2
```

```
##   model      psill     range
## 1   Exp 0.3920755 126.1115
```

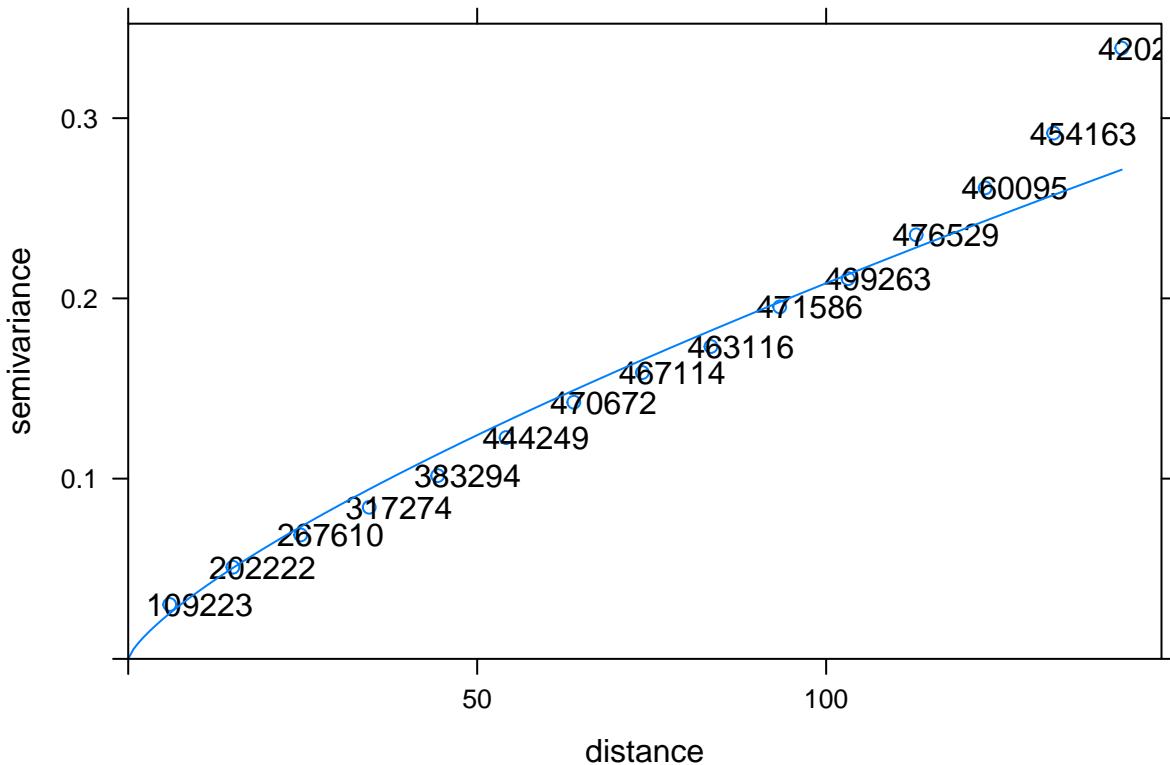
```
plot(vwind18, vwind18_m2, plot.numbers = T)
```



```
vwind18_m3
```

```
##   model      psill      range
## 1 Pow 0.006660904 0.7477137
```

```
plot(vwind18, vwind18_m3, plot.numbers = T)
```



```

attr(vwind18_m1, 'SSErr')

## [1] 1.287008

attr(vwind18_m2, 'SSErr')

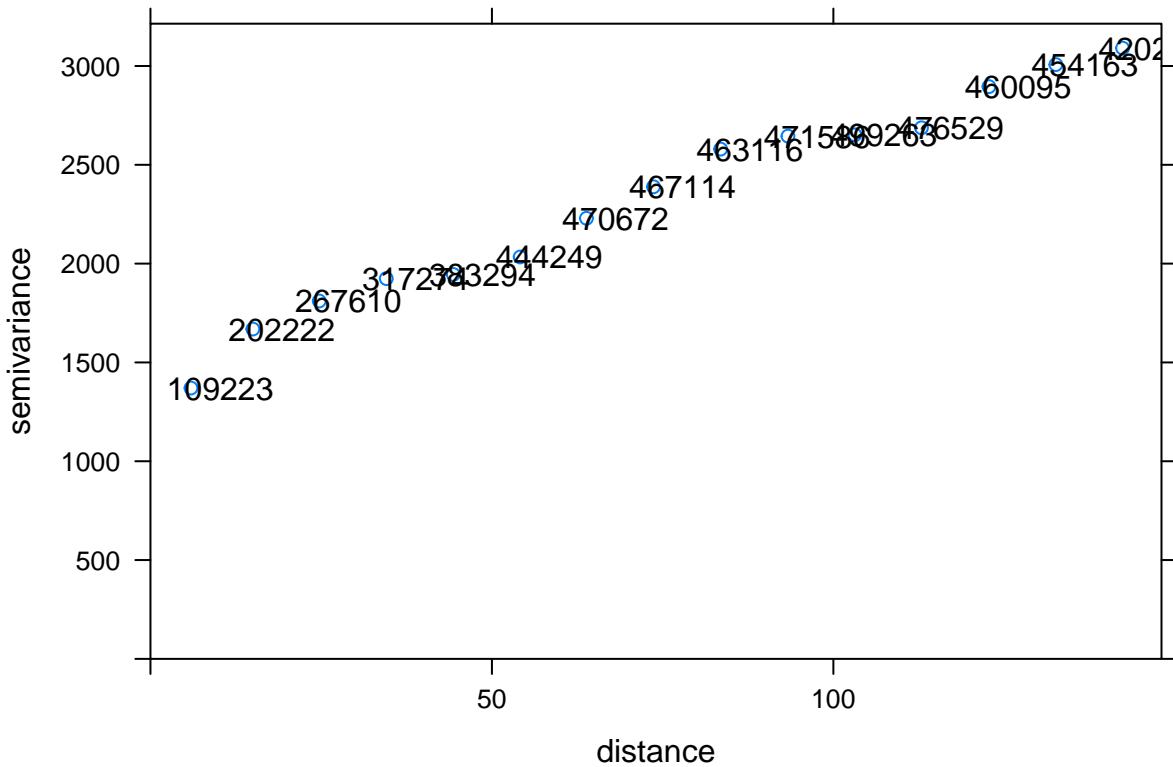
## [1] 0.8203171

attr(vwind18_m3, 'SSErr') #####
## [1] 0.3033476

```

Precipitación

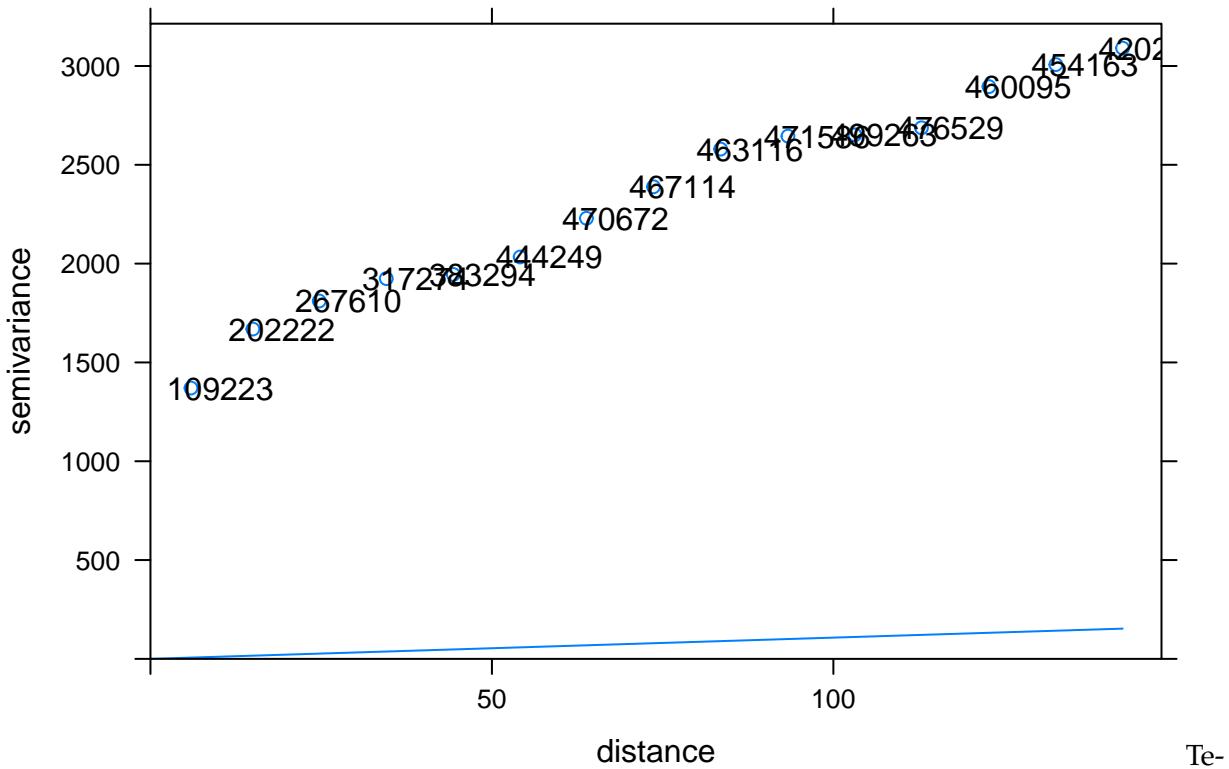
```
plot(vprec18, plot.numbers = T)
```



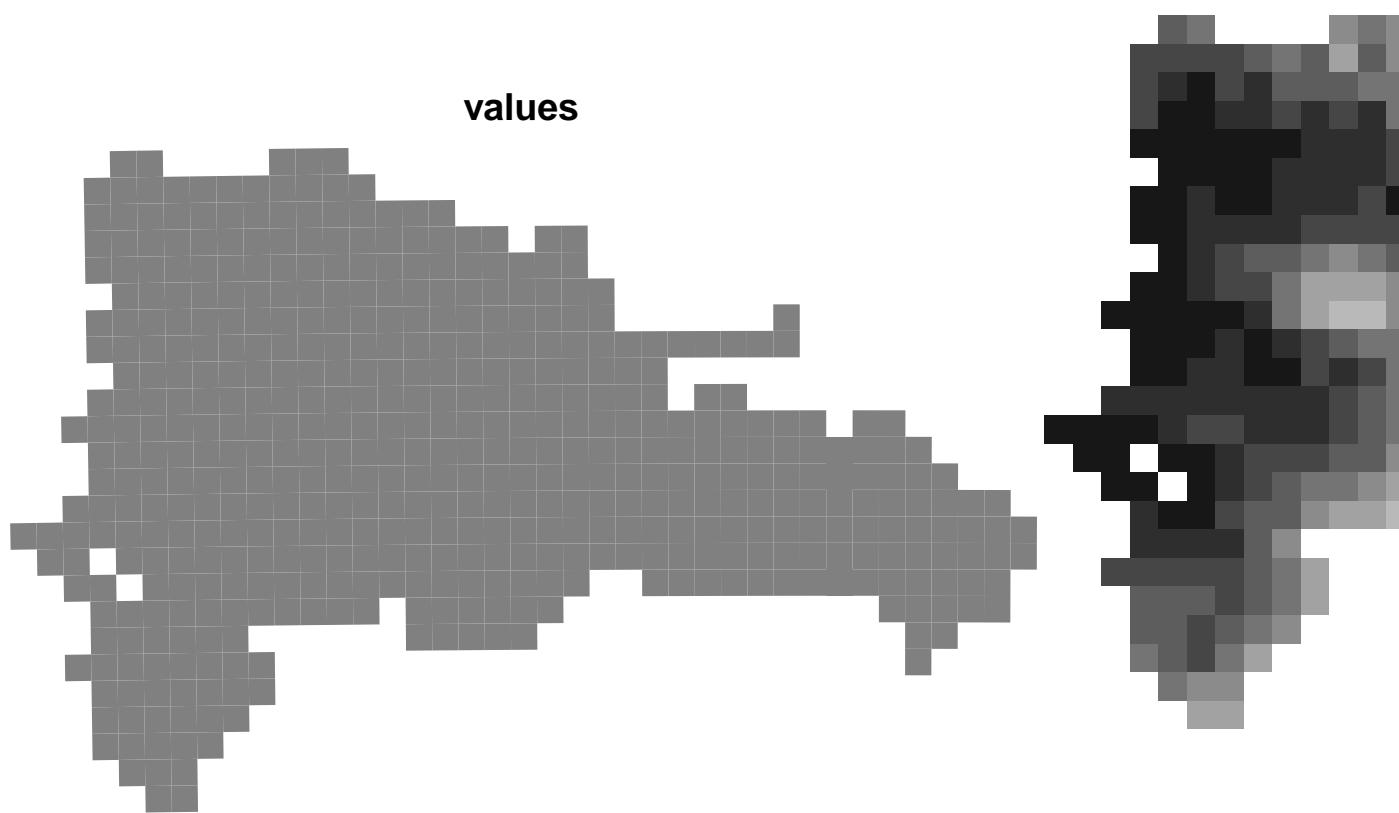
```
vprec18_m1
```

```
##   model    psill    range
## 1   Pen 2864.107 4995.939
```

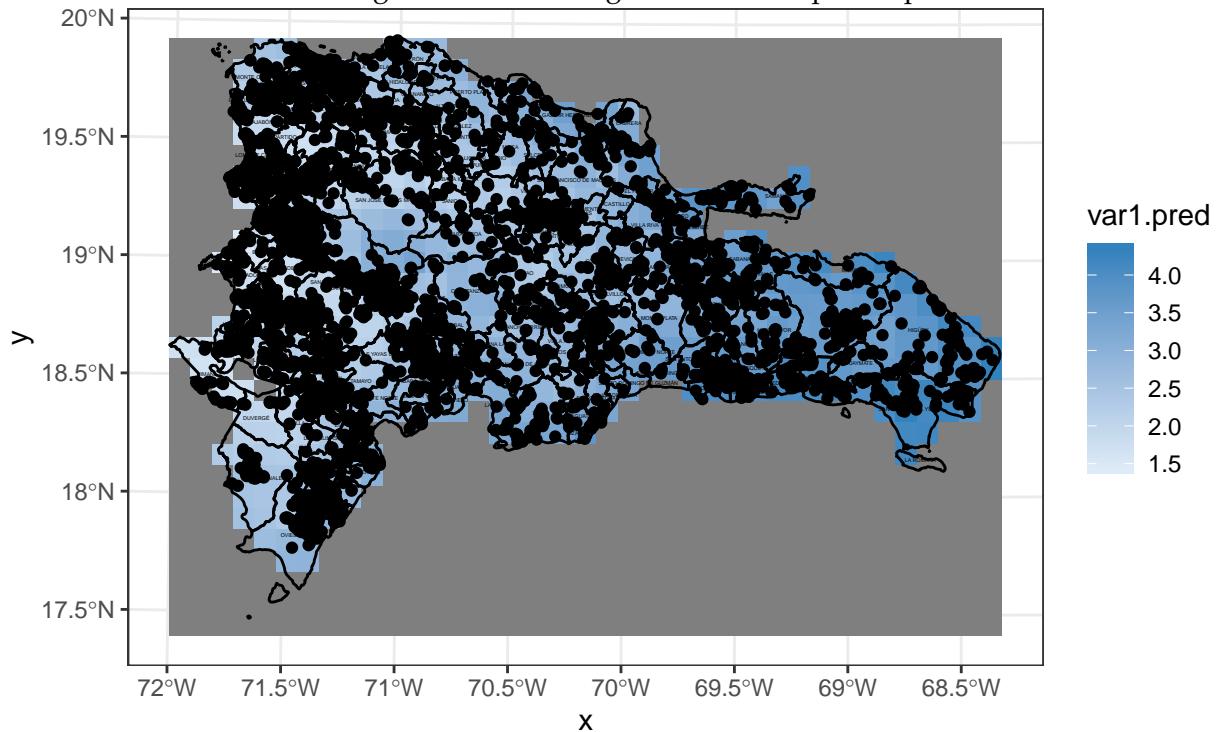
```
plot(vprec18, vprec18_m1, plot.numbers = T)
```



niendo en cuenta los resultados obtenidos, se visualiza que el modelo No.3 para el viento basado en el modelo *Pow (power)* es el que da mejor resultado por lo cual será utilizado para el modelo de interpolación “Krigue” para esto se genera una grilla de 10km y se realiza una transformación a UTM19N:32619 para poder realizar el modelo.



Como resultado del modelo “Krigue” se obtiene el siguiente modelo para el pais basado en los vien-

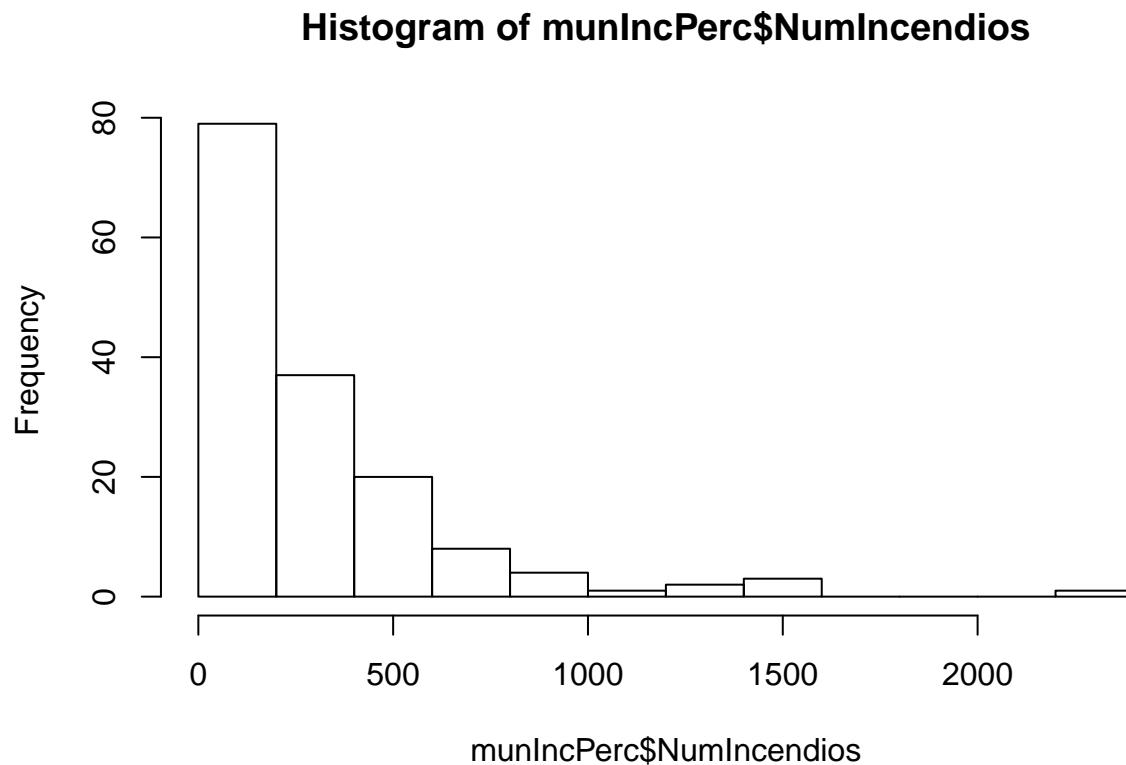


tos.

Por ultimo se realiza un modelo de prediccion teniendo en cuenta el modelo basandose en un modelo digital de elevaciones para buscar las zonas mas propensas teniendo en cuenta los incendios y la elevación para cada uno de los puntos.

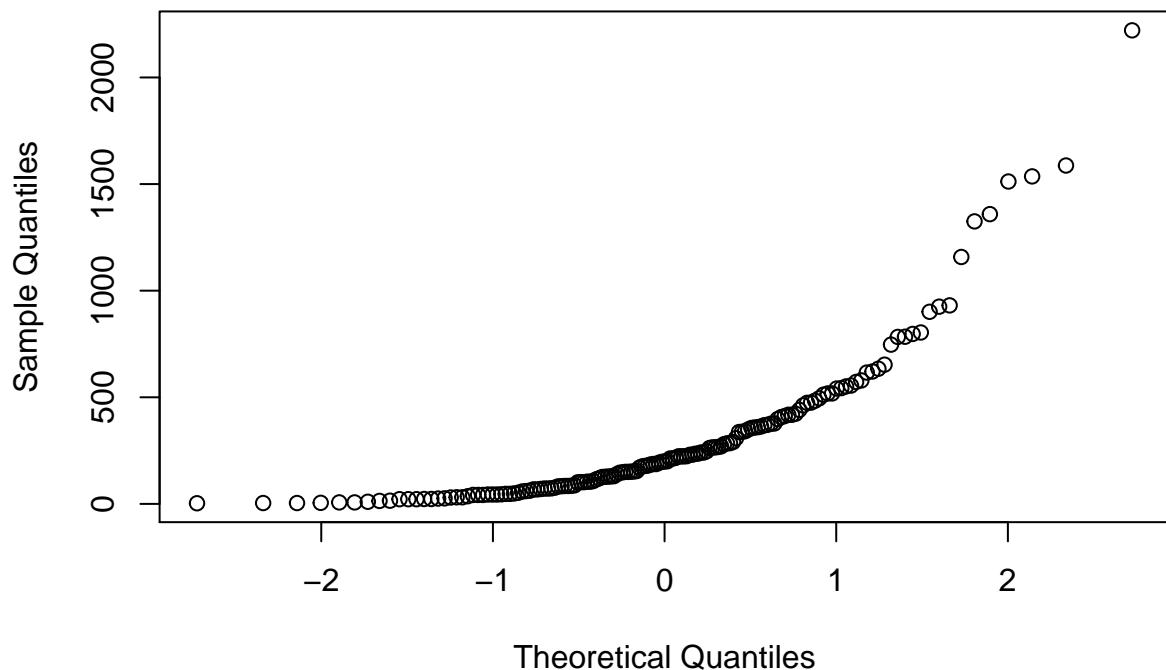
Teniendo en cuenta los datos para el numero de incendios se observa que mediante la modificación de los datos usando un logaritmo se estandarizan un poco permitiendo realizar un modelo mas cercano a la realidad.

```
hist(munIncPerc$NumIncendios)
```



```
qqnorm(munIncPerc$NumIncendios)
```

## Normal Q-Q Plot

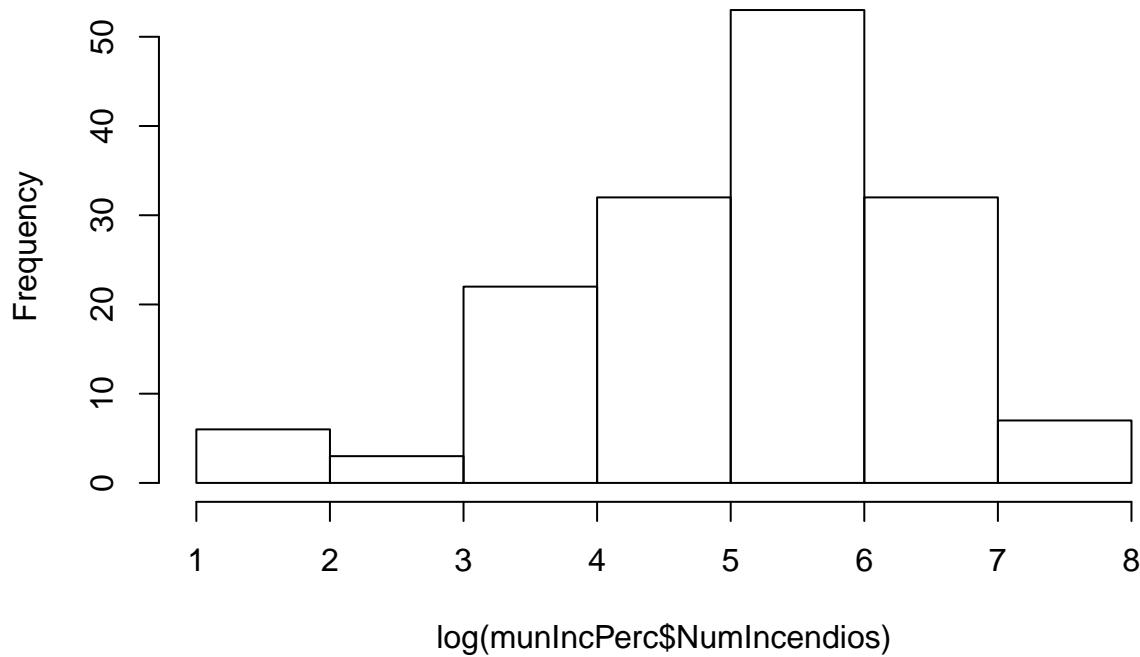


```
shapiro.test(munIncPerc$NumIncendios)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: munIncPerc$NumIncendios  
## W = 0.74127, p-value = 3.202e-15
```

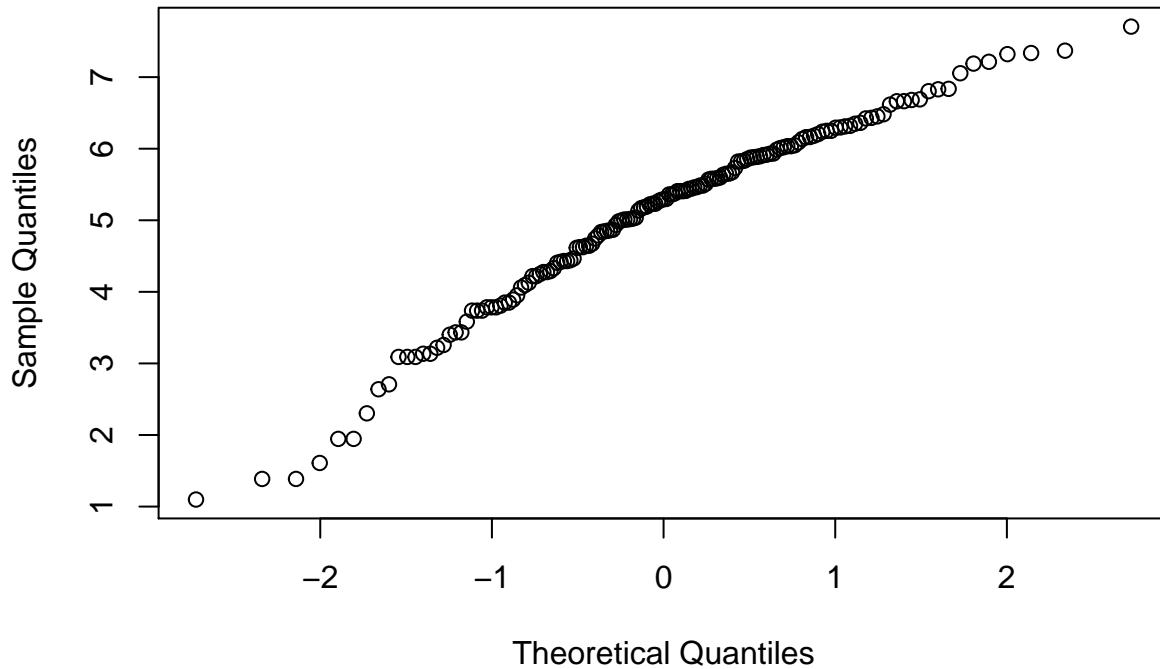
```
hist(log(munIncPerc$NumIncendios))
```

**Histogram of  $\log(\text{munIncPerc\$NumIncendios})$**



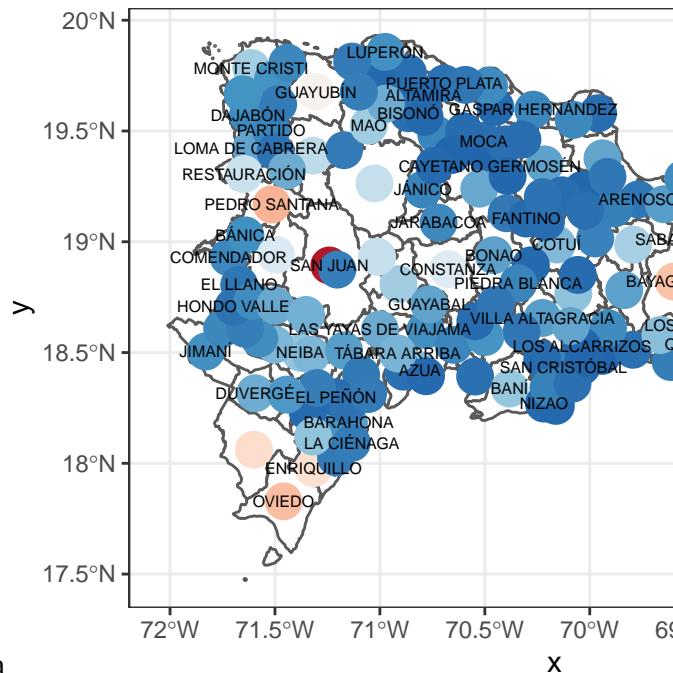
```
qqnorm(log(munIncPerc$NumIncendios))
```

**Normal Q-Q Plot**



```
shapiro.test(log(munIncPerc$NumIncendios))
```

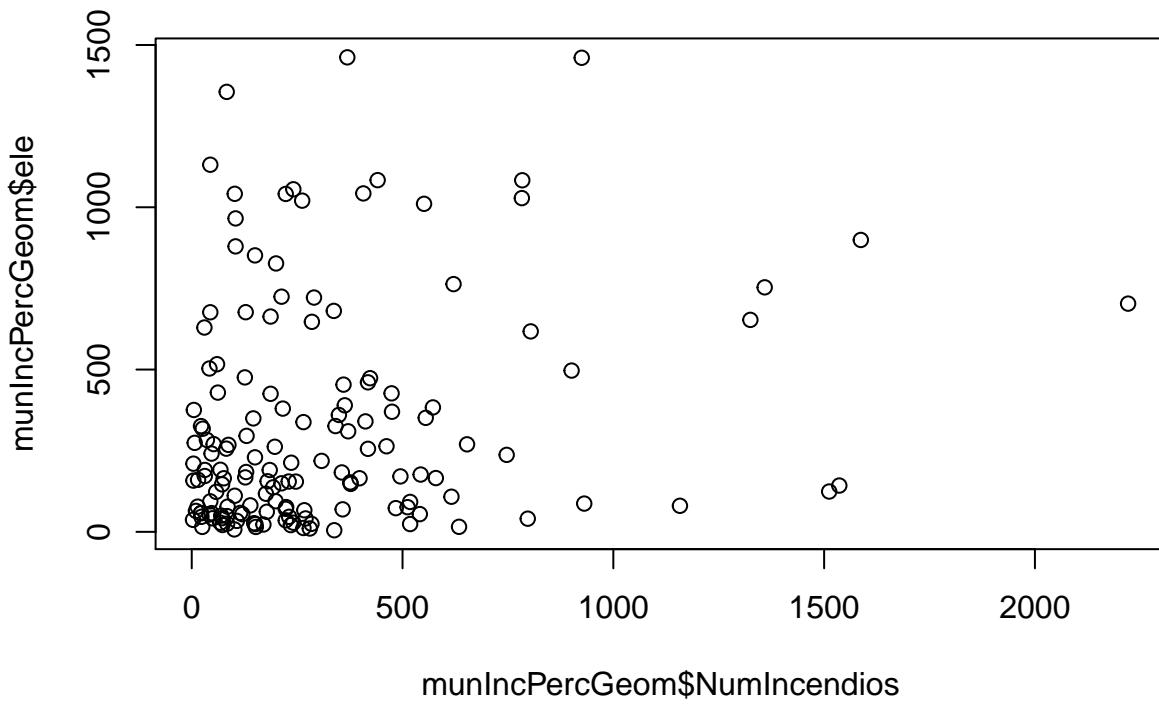
```
##  
## Shapiro-Wilk normality test  
##  
## data: log(munIncPerc$NumIncendios)  
## W = 0.96419, p-value = 0.0004752
```



Visto desde un mapa se representa de la siguiente manera

Se carga el modelo digital de elevación y se asignan los valores de elevación para cada incendio, ademas se remuestrea el valor basandose en la cuadricula de 10km realizada anteriormente, esta misma sirve para predecir los valores de incendios.

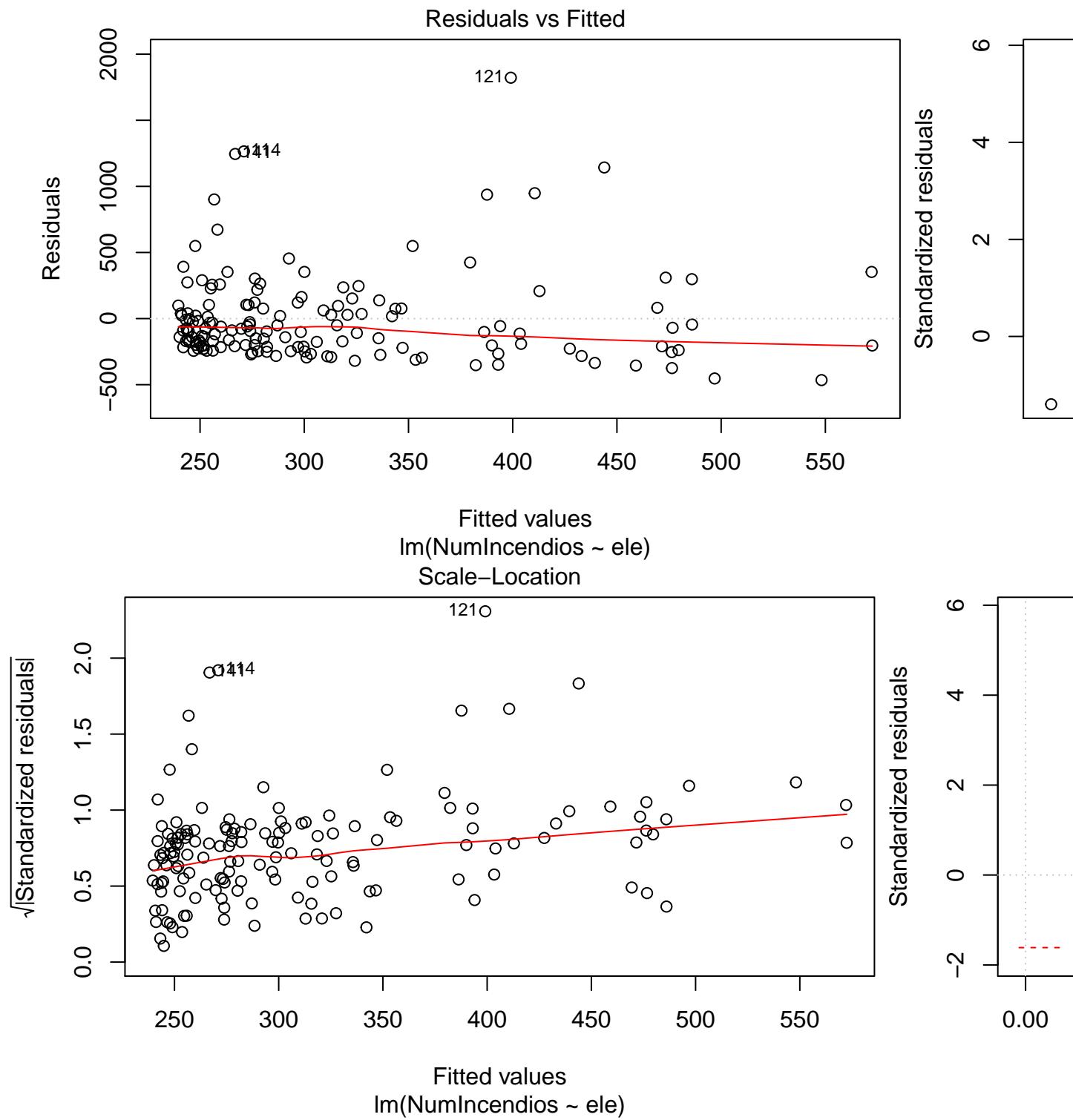
Se observa en el siguiente gráfico que la mayoría de incendios se presentan en elevaciones menores a 500m.



```
summary(munIncPercGeom_lm)
```

```
##
## Call:
## lm(formula = NumIncendios ~ ele, data = munIncPercGeom)
##
## Residuals:
##    Min      1Q  Median      3Q     Max 
## -465.17 -213.78 - 93.47   97.76 1821.90 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 238.45656   38.84970   6.138 7.3e-09 ***
## ele          0.22849    0.08273   2.762 0.00648 ** 
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 344.6 on 148 degrees of freedom
## Multiple R-squared:  0.04901, Adjusted R-squared:  0.04258 
## F-statistic: 7.627 on 1 and 148 DF,  p-value: 0.006479
```

```
plot(munIncPercGeom_lm)
```

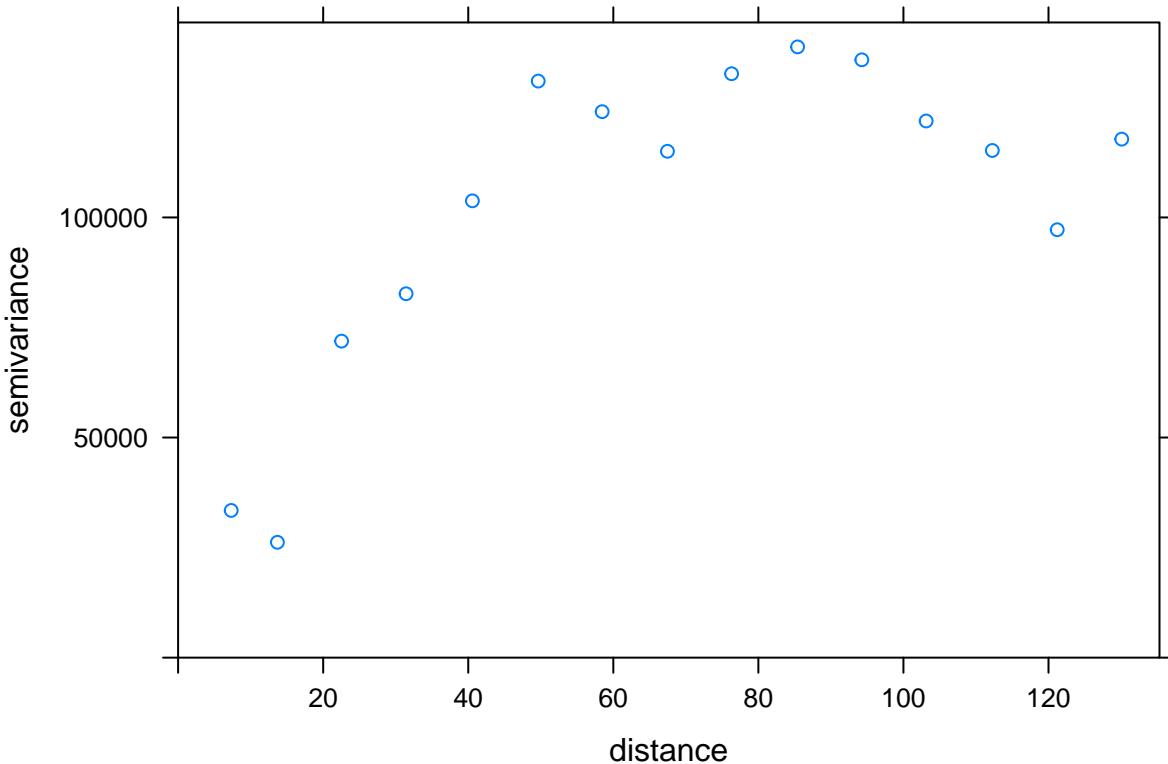


Se necesita que los datos de elevación pasen al objeto Incendios, para probar un modelo lineal que ponga en relación a los datos por lo que se prueban diferentes modelos.

```
vt <- variogram(NumIncendios ~ ele, munIncPercGeom)
vt
```

```
##      np      dist     gamma dir.hor dir.ver   id
## 1    12  7.317683 33441.30      0      0 var1
## 2   180 13.690184 26197.13      0      0 var1
## 3   283 22.538770 71902.26      0      0 var1
## 4   322 31.428440 82660.61      0      0 var1
## 5   384 40.572943 103773.04      0      0 var1
## 6   426 49.646198 130980.71      0      0 var1
## 7   458 58.462264 124034.04      0      0 var1
## 8   477 67.463910 115032.32      0      0 var1
## 9   515 76.307155 132645.72      0      0 var1
## 10  546 85.406215 138749.38      0      0 var1
## 11  556 94.268669 135813.05      0      0 var1
## 12  573 103.132259 121910.10      0      0 var1
## 13  574 112.257804 115212.33      0      0 var1
## 14  614 121.202576  97209.55      0      0 var1
## 15  577 130.099268 117794.99      0      0 var1
```

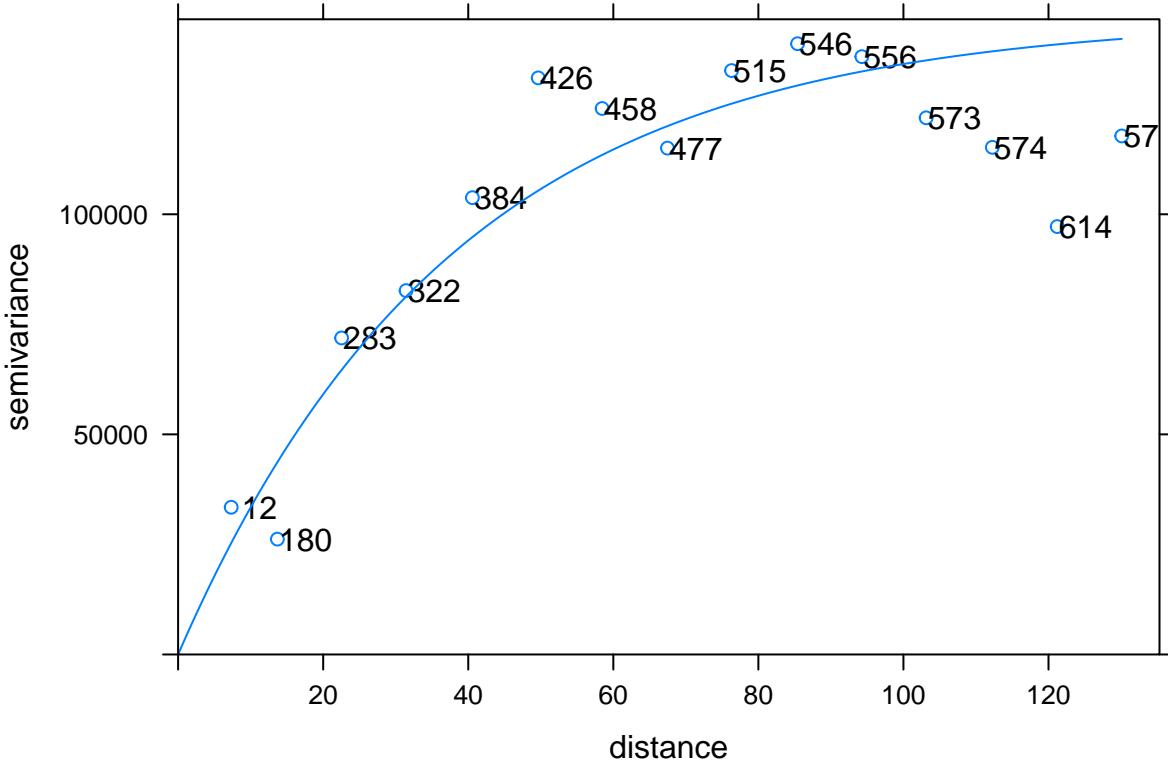
```
plot(vt)
```



```
vt_m <- fit.variogram(vt, vgm(model = "Exp", range = 50))
vt_m
```

```
##   model  psill    range
## 1   Exp 144570 38.04227
```

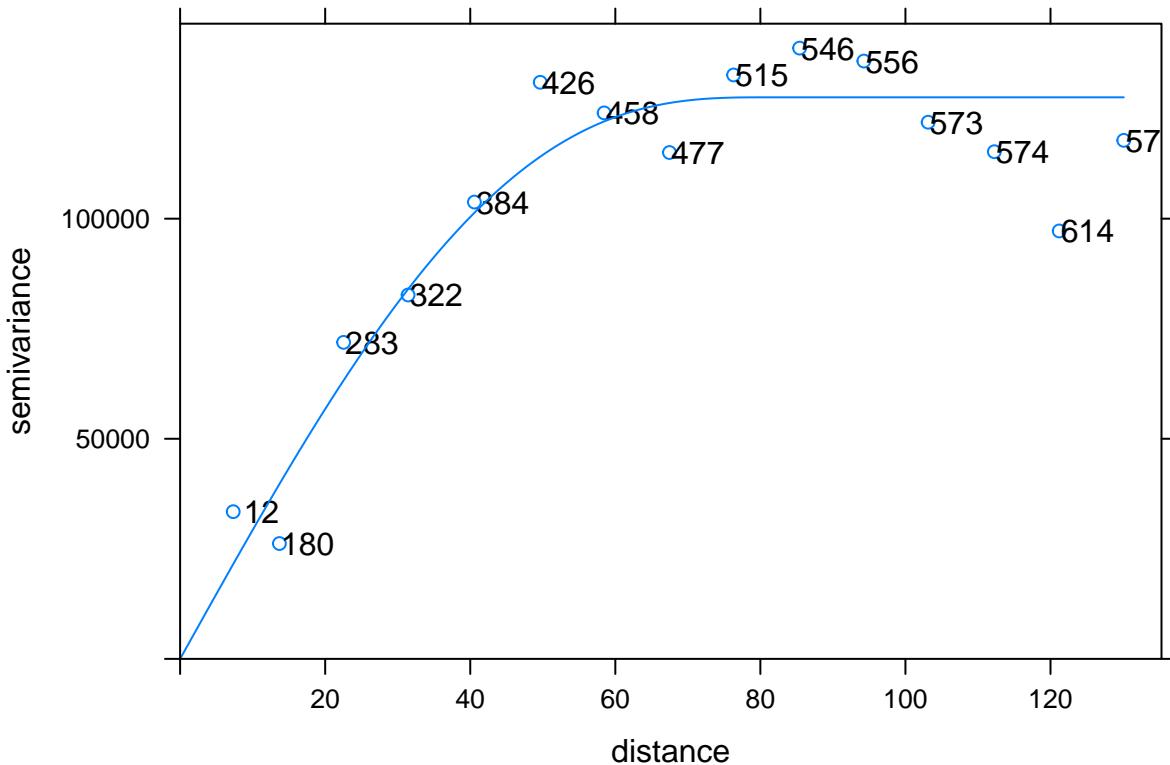
```
plot(vt, vt_m, plot.numbers = T)
```



```
vt_m2 <- fit.variogram(vt, vgm(model = "Pen", range = 50))
vt_m2
```

```
##   model  psill    range
## 1   Pen 127570.9 80.92264
```

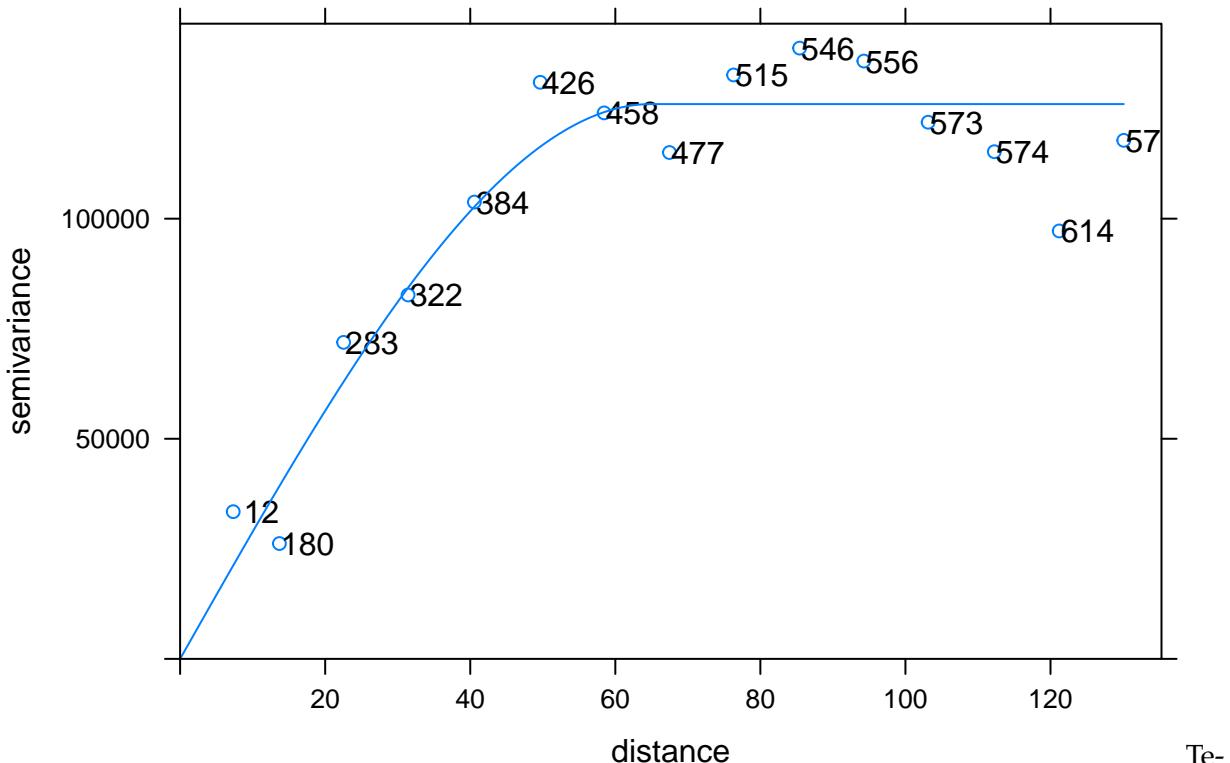
```
plot(vt, vt_m2, plot.numbers = T)
```



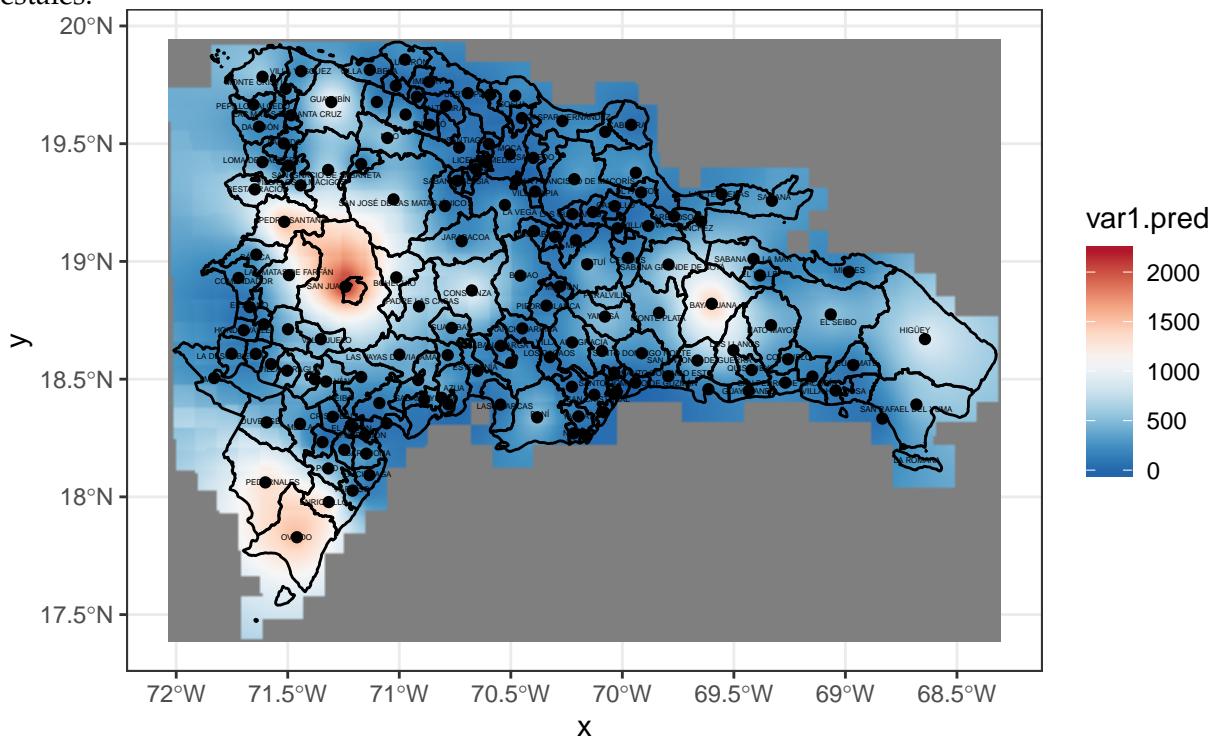
```
vt_m3 <- fit.variogram(vt, vgm(model = "Sph", range = 50))
vt_m3
```

```
##   model    psill    range
## 1   Sph 126034.7 64.97345
```

```
plot(vt, vt_m3, plot.numbers = T)
```



niendo en cuenta que el modelo mas ajustado es el modelo basado al modelo “Exponencial” se realiza el modelo de predicción de las zonas mas vulnerables para la propagación de incendios forestales.



## 4 Resultados

Es de notar que a lo largo del estudio realizado se observa que los municipios mas cercanos a la frontera tienen un comportamiento similar sin importar la variable a correlacionarse, se observó que al realizar el procesamiento mediante el método LISA Cluster los municipios de San Juan, Higuey y Neiba inicialmente presentan un alto indice de incendios, sin embargo al tener en cuenta el área de los municipios, los municipios del este del pais no tienen una tasa tan alta y arroja que Loma de Cabrera y Polo son los que tienen mayor indice de incendios. Por otro lado, al realizar el Kriging universal teniendo en cuenta la relación elevación/incendios se observa que los municipios cercanos a la frontera y en zona montañosa tienen un alto indice de propagación de incendios, ademas de estos se tienen dos datos facilmente explicables, para al municipio de Bayaguana el cual basa su economía en ganadería y campos arroceros se puede establecer que teniendo en cuenta que la forma mas sencilla y económica para expandir las áreas para dichas labores es la quema de bosque, asi mismo se comporta el municipio de Higuey en donde los complejos turísticos pueden tener un comportamiento similar.

## 5 Discusión o Conclusiones

Es recomendable realziar un estudio teniendo en cuenta los factores sociales puesto que estos pueden ser un factor influyente en la propagación de incendios para la República Dominicana

## 6 Información de soporte

Para el desarrollo de este trabajo se usó la siguiente información

- Incendios forestales NASA (2020) (Información de los ultimos 10 años para el pais)
- Variables climaticas (???) (2.5m)
  - Temperatura Promedio (C°)
  - Precipitacion (mm)
  - Radiación Solar (Kj m<sup>-2</sup> day<sup>-1</sup>)
  - Velocidad del viento (m s<sup>-1</sup>)
- Elevaciones DTM entregado por Martínez Batlle (2019a)
- Uso de suelo GlobCover (2005)
- OpenStreetMaps OpenStreetMaps (2020)

## 7 Script reproducible

```
library(sf)
library(sp)
library(raster)
library(tidyverse)
library(tmap)
library(RColorBrewer)
library(lmtest)
```

```

library(spdep)
library(parallel)
library(ggplot2)
library(gstat)
library(stars)
source('lisaclusters.R')

#carga de capas y transformación
incendios <- st_read(dsn = "data/Incendios/FiredataBuffer.shp")
mun <- st_read(dsn = 'data/DivisionRD/divisionRD.gpkg', layer = 'MUNCENS02010')
mun4326 <- st_transform(mun, crs = 4326)
usoSuelo <- raster('data/UsoSuelo/GLOBCOVER_RD.color.tif')

#extracción de datos raster
incUsoSuelo <- raster::extract(usoSuelo, incendios, sp=TRUE)
incForestales <- subset(incUsoSuelo, incUsoSuelo$GLOBCOVER_RD.color>20
                         & incUsoSuelo$GLOBCOVER_RD.color<130)
incForestales.df <- data.frame(incForestales)
summary(incForestales.df[[18]])

#Adición de la cobertura
incForestales.sf <- st_as_sf(incForestales)
incendiosForestales <- st_intersection(incForestales.sf, mun4326)
plot(incendiosForestales['GLOBCOVER_RD.color'])

#Conversion a Polígono
table(incendiosForestales$ENLACE)
munInc <- arrange(mun4326, ENLACE)
munInc$NumIncendios <- c(table(incendiosForestales$ENLACE))
plot(munInc['NumIncendios'])

#Vecindad
munInc.sp <- as_Spatial(munInc)
colnames(munInc.sp@data)
row.names(munInc.sp) <- as.character(munInc.sp$TOPONIMIA)
munInc.nb <- poly2nb(munInc.sp, queen = TRUE)
summary(munInc.nb)
plot(munInc.sp, border="grey", lwd=0.5)
plot(munInc.nb, coordinates(munInc.sp), add=T)

#Num Vecinos
coords <- coordinates(munInc.sp)
ident <- row.names(munInc.sp)
munInc.nb.k1 <- knn2nb(knearneigh(coords, k = 1), row.names = ident)
summary(munInc.nb.k1)

```

```

card(munInc.nb.k1)
plot(munInc.sp, border="grey", lwd=0.5)
plot(munInc.nb.k1, coordinates(munInc.sp), add=T)
is.symmetric.nb(munInc.nb.k1)
dist <- unlist(nbdist(munInc.nb.k1, coords))
summary(dist)
hist(dist)
boxplot(dist)
(distmin <- min(dist))
(distmax <- max(dist))
indicemin <- which(distmin==dist)
ident[indicemin]
indicemax <- which(distmax==dist)
ident[indicemax]
ident[order(dist)]
```

*#Ponderados espaciales*

```

munInc.w.W <- nb2listw(munInc.nb)
munInc.w.W
```

```

munInc.w.B <- nb2listw(munInc.nb, style = 'B')
munInc.w.B
```

*#Correlacion Incendios*

```

munIncPercGeom <- munInc %>% st_centroid() %>% mutate(
  'IncPercentage' = munInc$NumIncendios/sum(munInc$NumIncendios)*100,
  'IncPercentage_log' = log1p(munInc$NumIncendios/sum(munInc$NumIncendios)*100),
  'IncPercentage_tukey' = rcompanion::transformTukey(
    NumIncendios/sum(munInc$NumIncendios)*100, plotit = F),
  'IncPercentage_tukey_lambda' = rcompanion::transformTukey(
    NumIncendios/sum(munInc$NumIncendios)*100, returnLambda = T),
  'AreaKm2' = as.numeric((st_area(munInc)/1000000)),
  'IncXArea' = (munInc$NumIncendios/AreaKm2),
  'IncXArea_log' = log1p(munInc$NumIncendios/AreaKm2),
  'IncXArea_tukey' = rcompanion::transformTukey(NumIncendios/
    AreaKm2, plotit = F),
  'IncXArea_tukey_lambda' = rcompanion::transformTukey(
    NumIncendios/AreaKm2, returnLambda = T),
  x=unlist(map(geom,1)), y=unlist(map(geom,2)))
```

```

munIncPerc <- munIncPercGeom %>% st_drop_geometry()
```

*#Join*

```

munIncPercPol <- munInc %>%
  merge(munIncPerc, all.y=TRUE)
```

```

#Mapa Porcentajes
p1 <- tm_shape(munIncPercPol) +
  tm_fill(col = "IncPercentage", style = 'jenks',
          palette = brewer.pal(9, name = 'Reds'),
          title = 'Porcentaje Incendios') +
  tm_borders(lwd = 0.5)

p2 <- tm_shape(munIncPercPol) +
  tm_fill(col = "IncPercentage_log", style = 'jenks',
          palette = brewer.pal(9, name = 'Reds'),
          midpoint = NA,
          title = 'Porcentaje Incendios') +
  tm_borders(lwd = 0.5)

tmap_arrange(p1,p2)

#qq
qqnorm(munIncPerc$IncPercentage)
shapiro.test(munIncPerc$IncPercentage)

qqnorm(munIncPerc$IncPercentage_log)
shapiro.test(munIncPerc$IncPercentage_log)

qqnorm(munIncPerc$IncPercentage_tukey)
shapiro.test(munIncPerc$IncPercentage_tukey)

munIncPerc %>% lm(IncPercentage ~ x, .) %>% bptest()
munIncPerc %>% lm(IncPercentage ~ y, .) %>% bptest()
munIncPerc %>% lm(IncPercentage_log ~ x, .) %>% bptest()
munIncPerc %>% lm(IncPercentage_log ~ y, .) %>% bptest()
munIncPerc %>% lm(IncPercentage_tukey ~ x, .) %>% bptest()
munIncPerc %>% lm(IncPercentage_tukey ~ y, .) %>% bptest()

match(attr(munInc.w.W$neighbours, "region.id"), munIncPerc$TOPONIMIA)==1:155

(gmoranw <- moran.test(x = munIncPerc$IncXArea_tukey, listw = munInc.w.W ))
(gmoranb <- moran.test(x = munIncPerc$IncXArea_tukey, listw = munInc.w.B))

moran.plot(x = munIncPerc$IncPercentage_tukey, listw = munInc.w.W)

lisamap(objesp = munIncPercPol,
        var = 'IncPercentage_tukey',
        pesos = munInc.w.W,
        tituloleyenda = 'Significancia\n" x-y ", léase \n como "x" \n rodeado de "y" ',
        leyenda = T,

```

```

anchuratitulo = 1000,
tamanotitulo = 16,
fuentedatos = '',
titulomap = paste0('Clusters LISA de Porcentaje
de Incendios Forestales (Tukey Trans.))')

# Mapa Porcentaje por Km2
p3 <- tm_shape(munIncPercPol) +
  tm_fill(col = "IncXArea", style = 'jenks',
  palette = brewer.pal(9, name = 'Reds'),
  title = 'Porcentaje Incendios por Km2') +
  tm_borders(lwd = 0.5)
p4 <- tm_shape(munIncPercPol) +
  tm_fill(col = "IncXArea_log", style = 'jenks',
  palette = brewer.pal(9, name = 'Reds'),
  title = 'Porcentaje Incendios por Km2') +
  tm_borders(lwd = 0.5)

tmap_arrange(p3,p4)

#qq
qqnorm(munIncPerc$IncXArea)
shapiro.test(munIncPerc$IncXArea)

qqnorm(munIncPerc$IncXArea_log)
shapiro.test(munIncPerc$IncXArea_log)

qqnorm(munIncPerc$IncXArea_tukey)
shapiro.test(munIncPerc$IncXArea_tukey)

munIncPerc %>% lm(IncXArea ~ x, .) %>% bptest()
munIncPerc %>% lm(IncXArea ~ y, .) %>% bptest()
munIncPerc %>% lm(IncXArea_log ~ x, .) %>% bptest()
munIncPerc %>% lm(IncXArea_log ~ y, .) %>% bptest()
munIncPerc %>% lm(IncXArea_tukey ~ x, .) %>% bptest()
munIncPerc %>% lm(IncXArea_tukey ~ y, .) %>% bptest()

match(attr(munInc.w.W$neighbours, "region.id"), munIncPerc$TOPONIMIA)==1:155

(gmoranw <- moran.test(x = munIncPerc$IncXArea_tukey, listw = munInc.w.W ))
(gmoranb <- moran.test(x = munIncPerc$IncXArea_tukey, listw = munInc.w.B))

moran.plot(x = munIncPerc$IncPercentage_tukey, listw = munInc.w.W)

lisamap(objesp = munIncPercPol,
       var = 'IncXArea_tukey',
       pesos = munInc.w.W,

```

```

tituloleyenda = 'Significancia\n'
                  "x-y", léase\ncomo "x"\nrodeado de "y",
leyenda = T,
anchuratitulo = 1000,
tamanotitulo = 16,
fuenedatos = '',
titulomap = paste0('Clusters LISA de Porcentaje
                  de Incendios Forestales (Tukey Trans.)')

#correlación variables world clim

# * Stack
wcayerspath <- list.files(path = 'data/WorldClim/',
                           pattern = '*.tif', recursive = T, full.names = T)
wcstack <- stack(wcayerspath)
# * Add month and Year field
incendiosForestales$month <- format(as.Date(incendiosForestales$ACQ_DATE), "%m")
incendiosForestales$year <- strtoi(format(as.Date(
                           incendiosForestales$ACQ_DATE), "%Y"))
# * Add unique field
incendiosForestales$unique <- 1:nrow(incendiosForestales)

# * Extract values from WC corresponding to the month of each point
system.time(
  foo <- sapply(1:20, function(x) {
    sp <- incendiosForestales[x,]
    m <- ifelse(nchar(sp$month)==1, paste0('0', sp$month), sp$month)
    e <- raster::extract(wcstack[[grep(paste0(m, '$'), names(wcstack))]],
                          sp, sp=T)
    d <- e@data[,c('unique', 'month',
                  grep('RD_wc2.*', colnames(e@data), value = T))]
    return(d)
  }, simplify = F)
)
#Laptop 8 cores, 8 GB
# user system elapsed
# 20.801 0.008 20.807
#Workstation, 8 cores, 64 GB
# user system elapsed
# 12.036 0.012 12.047
bar <- inner_join(incendiosForestales, bind_rows(foo), by = c('unique', 'month'))
bar

#Parallel
UseCores <- detectCores() - 1
cl <- makeCluster(UseCores, outfile=paste0('info_parallel.log'))
clusterExport(cl, list('incendiosForestales', 'wcstack'))

```

```

system.time(
  foo <- parSapply(cl, 1:20, function(x) {
    sp <- sf::as_Spatial(incendiosForestales[x,])
    m <- ifelse(nchar(sp$month)==1, paste0('0', sp$month), sp$month)
    e <- raster::extract(wcstack[[grep(paste0(m,'$'), names(wcstack))]], 
      sp, sp=T)
    d <- e@data[,c('unique', 'month', grep('RD_wc2.*',
      colnames(e@data), value = T))]
    return(d)
  }, simplify = F)
)
#Laptop 8 cores, 8 GB
# user system elapsed
# 0.009  0.008  2.775
#Workstation, 8 cores, 64 GB
# user system elapsed
# 0.009  0.000  1.552
#Estimated entire job:
1.552/20*nrow(incendiosForestales)/60
# [1] 60.96256
stopCluster(cl)
bar <- inner_join(incendiosForestales, bind_rows(foo),
  by = c('unique', 'month'))
bar
#
#Parallel
UseCores <- detectCores() - 1
cl <- makeCluster(UseCores, outfile=paste0('info_parallel.log'))
clusterExport(cl, list('incendiosForestales', 'wcstack'))

## Not run:
system.time(
  foo <- parSapply(cl, 1:nrow(incendiosForestales), function(x) {
    sp <- sf::as_Spatial(incendiosForestales[x,])
    m <- ifelse(nchar(sp$month)==1, paste0('0', sp$month), sp$month)
    e <- raster::extract(wcstack[[grep(paste0(m,'$'), names(wcstack))]], 
      sp, sp=T)
    d <- e@data[,c('unique', 'month', grep('RD_wc2.*',
      colnames(e@data),
      value = T))]
    return(d)
  }, simplify = F)
)
## End(Not run)
#Workstation, 8 cores, 64 GB

```

```

stopCluster(cl)
bar <- inner_join(incendiosForestales, bind_rows(foo),
                  by = c('unique', 'month'))
bar
#
puntos_calor_fuegos_y_worldclim_viento_temp_radiacion

##### EASYMODE
#extracción de datos raster
#st_write(incendiosForestales, "data/Incendios/incendiosForestales.shp")

tablaWorldClim <- read.table("data/Incendios/Test2.csv",
                             header = TRUE,
                             sep = ",")
#####
END EASYMODE

incForWorldClim <- left_join(incendiosForestales,
                             select(tablaWorldClim,
                                    c(unique, month, wind, temp, srad, prec)),
                             by = 'unique')

#quitar valores 0
incForestWorldClim <- incForWorldClim[incForWorldClim$srad != 0
                                         & incForWorldClim$prec != 0, ]

#Segmentacion por año
hist(incendiosForestales$year)
incForestWC2018 <- incForWorldClim[incForWorldClim$year == 2018, ]
incForWC2018 <- incForestWC2018[incForestWC2018$srad != 0
                                   & incForestWC2018$prec != 0, ]

#Año 2018

#EDA
nrow(incForWC2018)
summary(incForWC2018$prec)
summary(incForWC2018$wind)
summary(incForWC2018$srad)
summary(incForWC2018$temp)

#Temperatura

```

```

hist(incForWC2018$prec)
shapiro.test(incForWC2018$temp)
shapiro.test(log(incForWC2018$temp))
ggplot() +
  geom_sf(data = mun4326, fill = 'white') +
  geom_sf(data = incForWC2018, aes(col = temp), size = 3) +
  scale_colour_gradient(low="#f7dede", high="#ff0011") +
  geom_sf_text(data = incForWC2018, aes(label=MUN),
               check_overlap = T, size = 0.5) +
  theme_bw()

#Radiacion Solar
shapiro.test(incForWC2018$srad)
shapiro.test(log(incForWC2018$srad))
ggplot() +
  geom_sf(data = mun4326, fill = 'white') +
  geom_sf(data = incForWC2018, aes(col = srad), size = 3) +
  scale_colour_gradient(low="#f7dede", high="#ff0011") +
  geom_sf_text(data = incForWC2018, aes(label=MUN),
               check_overlap = T, size = 0.5) +
  theme_bw()

#Viento
shapiro.test(incForWC2018$wind)
shapiro.test(log(incForWC2018$wind))
ggplot() +
  geom_sf(data = mun4326, fill = 'white') +
  geom_sf(data = incForWC2018, aes(col = wind), size = 3) +
  scale_colour_gradient(low="#deebf7", high="#3182bd") +
  geom_sf_text(data = incForWC2018, aes(label=MUN),
               check_overlap = T, size = 0.5) +
  theme_bw()

#Precipitacion
shapiro.test(incForWC2018$prec)
shapiro.test(log(incForWC2018$prec))
ggplot() +
  geom_sf(data = mun4326, fill = 'white') +
  geom_sf(data = incForWC2018, aes(col = prec), size = 3) +
  scale_colour_gradient(low="#deebf7", high="#3182bd") +
  geom_sf_text(data = incForWC2018, aes(label=MUN),
               check_overlap = T, size = 0.5) +
  theme_bw()

#variograma
#Radiacion Solar
vsrad18 <- variogram(srad~1, incForWC2018)

```

```

vsrad18_m1 <- fit.variogram(vsrad18, vgm(model = "Exp", range = 5))
plot(vsrad18, plot.numbers = T)
vsrad18_m1
plot(vsrad18, vsrad18_m1, plot.numbers = T)

#Temperatura
vtemp18 <- variogram(temp~1, incForWC2018)
vtemp18_m1 <- fit.variogram(vtemp18, vgm(model = "Exp", range = 50))
plot(vtemp18, plot.numbers = T)
vtemp18_m1
plot(vtemp18, vtemp18_m1, plot.numbers = T)
attr(vtemp18_m1, 'SSErr')

#Viento
vwind18 <- variogram(wind~1, incForWC2018)
vwind18_m1 <- fit.variogram(vwind18, vgm(model = "Lin", range = 50))
vwind18_m2 <- fit.variogram(vwind18, vgm(model = "Exp", range = 50))
vwind18_m3 <- fit.variogram(vwind18, vgm(model = "Pow", range = 1))
plot(vwind18, plot.numbers = T)
vwind18_m1
plot(vwind18, vwind18_m1, plot.numbers = T)
vwind18_m2
plot(vwind18, vwind18_m2, plot.numbers = T)
vwind18_m3
plot(vwind18, vwind18_m3, plot.numbers = T)

attr(vwind18_m1, 'SSErr')
attr(vwind18_m2, 'SSErr')
attr(vwind18_m3, 'SSErr') #***

#Precipitacion
vprec18 <- variogram(prec~1, incForWC2018)
vprec18_m1 <- fit.variogram(vprec18, vgm(model = "Pen", range = 5000))
plot(vprec18, plot.numbers = T)
vprec18_m1
plot(vprec18, vprec18_m1, plot.numbers = T)

## Loading required package: abind
grd <- st_bbox(mun) %>%
  st_as_stars(dx = 10000) %>% #10000 metros=10km de resolución espacial
  st_crop(mun)
grd
grd4326 <- st_transform(grd, crs = 4326)
plot(grd4326)

incForWC2018.32619 <- st_transform(incForWC2018, crs = 32619)

```

```

#krigging
k <- krige(formula = wind~1, locations = incForWC2018.32619,
            newdata = grd, model = vwind18_m3)
plot(k)

ggplot() +
  geom_stars(data = k, aes(fill = var1.pred, x = x, y = y)) +
  scale_fill_gradient(low="#deebf7", high="#3182bd") +
  geom_sf(data = st_cast(mun, "MULTILINESTRING")) +
  geom_sf(data = incForWC2018) +
  geom_sf_text(data = mun, aes(label=TOPONIMIA), check_overlap = T, size = 0.8) +
  theme_bw()

#DEM

nrow(munIncPerc)
summary(munIncPerc$NumIncendios)
hist(munIncPerc$NumIncendios)
qqnorm(munIncPerc$NumIncendios)

hist(log(munIncPerc$NumIncendios))
qqnorm(log(munIncPerc$NumIncendios))
shapiro.test(munIncPerc$NumIncendios)
shapiro.test(log(munIncPerc$NumIncendios))

ggplot() +
  geom_sf(data = mun4326, fill = 'white') +
  geom_sf(data = munIncPercGeom, aes(col = NumIncendios), size = 6) +
  scale_colour_gradientn(colours = rev(brewer.pal(9, name = 'RdBu'))) +
  geom_sf_text(data = munIncPercGeom, aes(label=TOPONIMIA),
               check_overlap = T, size = 2) +
  theme_bw()

#dem

dem <- read_stars('data/DTM/dem_srtm_remuestreado.tif')
names(dem) <- 'ele'
dem4326 <- st_transform(dem, crs = 4326)
plot(dem4326)

grdcovars <- aggregate(dem, grd, mean, na.rm=T) %>% st_transform(crs = 4326)
plot(grdcovars)

munIncPercGeom$ele <- st_as_sf(aggregate(grdcovars, munIncPercGeom, mean))[[1]]

```

```

munIncPercGeom <- munIncPercGeom [!is.na(munIncPercGeom$ele),]
plot(munIncPercGeom$NumIncendios, munIncPercGeom$ele)

munIncPercGeom_lm <- lm(NumIncendios ~ ele, munIncPercGeom)
summary(munIncPercGeom_lm)
plot(munIncPercGeom_lm)

vt <- variogram(NumIncendios ~ ele, munIncPercGeom)
vt
plot(vt)

vt_m <- fit.variogram(vt, vgm(model = "Exp", range = 50))
vt_m
plot(vt, vt_m, plot.numbers = T)

vt_m2 <- fit.variogram(vt, vgm(model = "Pen", range = 50))
vt_m2
plot(vt, vt_m2, plot.numbers = T)

vt_m3 <- fit.variogram(vt, vgm(model = "Sph", range = 50))
vt_m3
plot(vt, vt_m3, plot.numbers = T)

k_u <- krige(NumIncendios ~ ele , munIncPercGeom,
              st_rasterize(st_as_sf(grdcovars)), vt_m)

ggplot() +
  geom_stars(data = k_u, aes(fill = var1.pred, x = x, y = y)) +
  scale_fill_gradientn(colours = rev(brewer.pal(9, name = 'RdBu'))) +
  geom_sf(data = st_cast(mun4326, "MULTILINESTRING")) +
  geom_sf(data = munIncPercGeom) +
  geom_sf_text(data = mun4326, aes(label=TOPONIMIA), check_overlap = T,
               size = 1) +
  theme_bw()

```

## Referencias

- Bivand, R. S., Pebesma, E. J., & Gomez-Rubio, V. (2008). *Applied spatial data analysis with R* (Vol. 747248717). Springer.
- Diario Libre, T. M. -. (2019). *Incendios forestales se intensifican en el país; van más de 40 este año*. Retrieved from <https://www.diariolibre.com/actualidad/van-mas-de-40-fuegos-forestales-en-este-2019-LE123441>
- EFE, A. (2019). *República dominicana busca prevenir incendios forestales en medio de sequía*. Retrieved from <https://www.eldinero.com.do/79275/republica-dominicana-busca-prevenir-incendios-forestales-e>
- GlobCover. (2005). *European space agency globcover portal*. Retrieved from <http://due.esrin.esa>.

[int/page\\_globcover.php](#)

Martínez Batlle, J. R. (2019a). *Clase de análisis espacial (uasd)*.

Martínez Batlle, J. R. (2019b). *Script adición data raster*.

Martínez Batlle, J. R. (2019c). *Script adición lisa cluster*.

NASA. (2020). *Página de descargas – fire information for resource management system (firms)*. Retrieved from <https://firms.modaps.eosdis.nasa.gov/download/create.php>

NASA-FIRMS. (2020). *Fire information for resource management system (firms)*. Retrieved from <https://firms.modaps.eosdis.nasa.gov/>

OpenStreetMaps. (2020). *OpenStreetMaps*. Retrieved from <https://download.geofabrik.de/>

Version2, W. (2000). *Data worldclim version2 2.5m*. Retrieved from <http://www.worldclim.org/>