

# MAPPATURA DEL RADON INDOOR IN TRENTINO CON SOFTWARE LIBERO: UN APPROCCIO GEOSTATISTICO

Stefano Pegoretti

APPA Trento  
Settore Laboratorio

[stefano.pegoretti@provincia.tn.it](mailto:stefano.pegoretti@provincia.tn.it)



AGENZIA PROVINCIALE PER  
LA PROTEZIONE DELL'AMBIENTE

**PROGETTO  
RADON**  
MISURE PER 1000  
FAMIGLIE



RISULTATI E  
APPROFONDIMENTI



giornata di studio  
gratuita e aperta al pubblico  
ingresso consentito fino alla capienza massima della sala

11 OTTOBRE 2018  
Palmanova, teatro Gustavo Modena

# UNA MAPPA... PER CHI?

- facilmente *interpretabile* dal cosiddetto “utente medio”  
(o anche meno)
- possibilmente non “*mal-interpretabile*”

Quale parametro rappresentare?

- *probabilità di superamento* di un dato valore di soglia  
vs. un valore di concentrazione
- *Simulazioni Gaussiane Sequenziali* vs. kriging

# SIMULAZIONI STOCASTICHE (CONDIZIONATE)

... dando per scontata una buona “analisi esplorativa dei dati” e lo loro “pulizia” ...

1. determinare la funzione densità di probabilità cumulativa (c.d.f.) dei dati sperimentali rappresentativa dell’intera area di studio
2. trasformare la c.d.f. in una distribuzione normale standardizzata  
tecnicamente, passare dallo spazio-Z allo spazio-Y
3. determinare il variogramma sperimentale nello spazio-Y
4. definire una griglia di simulazione e un opportuno “vicinaggio” di ricerca  
quali/quanti punto uso per condizionare la simulazione?
5. effettuare N realizzazioni nello spazio-Y toccando ogni punto della griglia secondo un percorso casuale  
stima dei parametri della c.d.f. locale in base al variogramma del punto 3 e ricorrendo a *Simple Kriging*
6. (retro)trasformare i dati dallo spazio-Y allo spazio-Z
7. post-processing dei dati ottenuti

# SI PUÒ FARE IN UN AMBIENTE DI LAVORO BASATO UNICAMENTE SU SOFTWARE LIBERO?

Vorrei poter (facilmente):

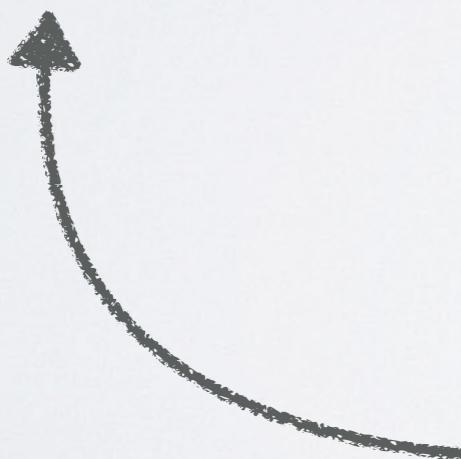
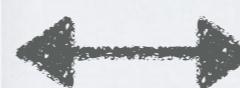
- importare ed esportare facilmente i dati  
(far comunicare i vari sotto-ambienti)
- gestire, manipolare e visualizzare i dati
- effettuare analisi statistiche “tradizionali”
- effettuare analisi di tipo geostatistico
- effettuare post-processing ed elaborazione  
dei risultati



“pagando” con la scrittura  
di qualche riga di  
“codice”...

## R Studio® IDE

- Data analysis scripts
- Interactive web applications
- Documents
- Reports
- Graphs



## R packages for data science

The tidyverse is an opinionated collection of R packages designed for data science. All packages share an underlying design philosophy, grammar, and data structures.

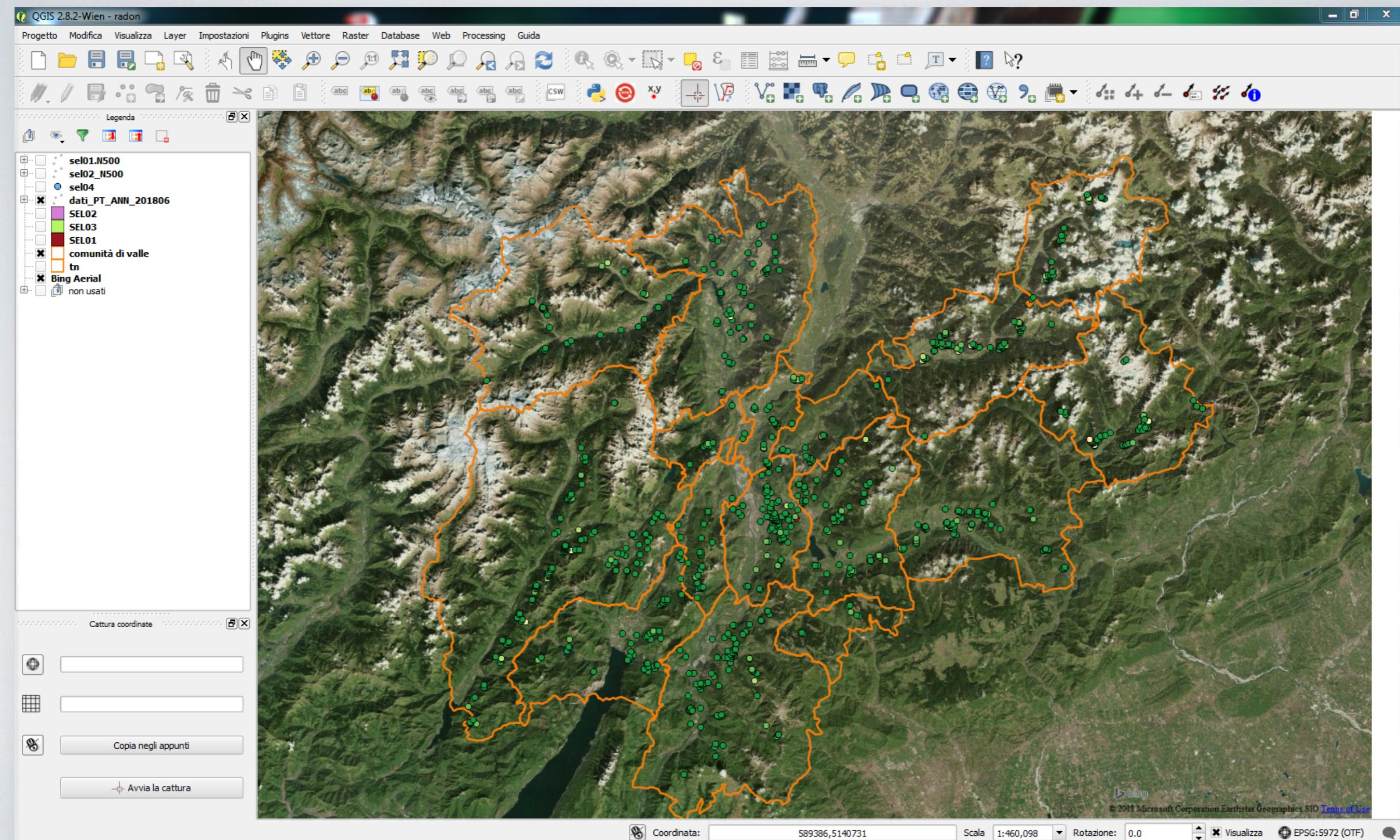
Install the complete tidyverse with:

```
install.packages("tidyverse")
```

# UN ESEMPIO “PRATICO”...

...con un dataset di test

# SELEZIONE DEI DATI



# ANALISI ESPLORATIVA DEI DATI DATI E SELEZIONE

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

cGSim\_SEL01.R x

Source on Save Go to file/function Addins

Environment History Connections

Import Dataset

Global Environment

Data

- g1 List of 9
- neigh Formal class db
- sel01 573 obs. of 6 variables
- sel01.anam Formal class anam
- sel01.db Formal class db
- sel01.db.Y Formal class db
- sel01.grid Formal class db
- sel01.moving.neigh Formal class neigh
- sel01.vg Formal class vario
- sel01.vg.model.Y Formal class model
- sel01.vg.Y Formal class vario

values

lag 2500

Files Plots Packages Help Viewer

Zoom Export Publish

Distribuzione valori di concentrazione Radon indoor

Console Terminal

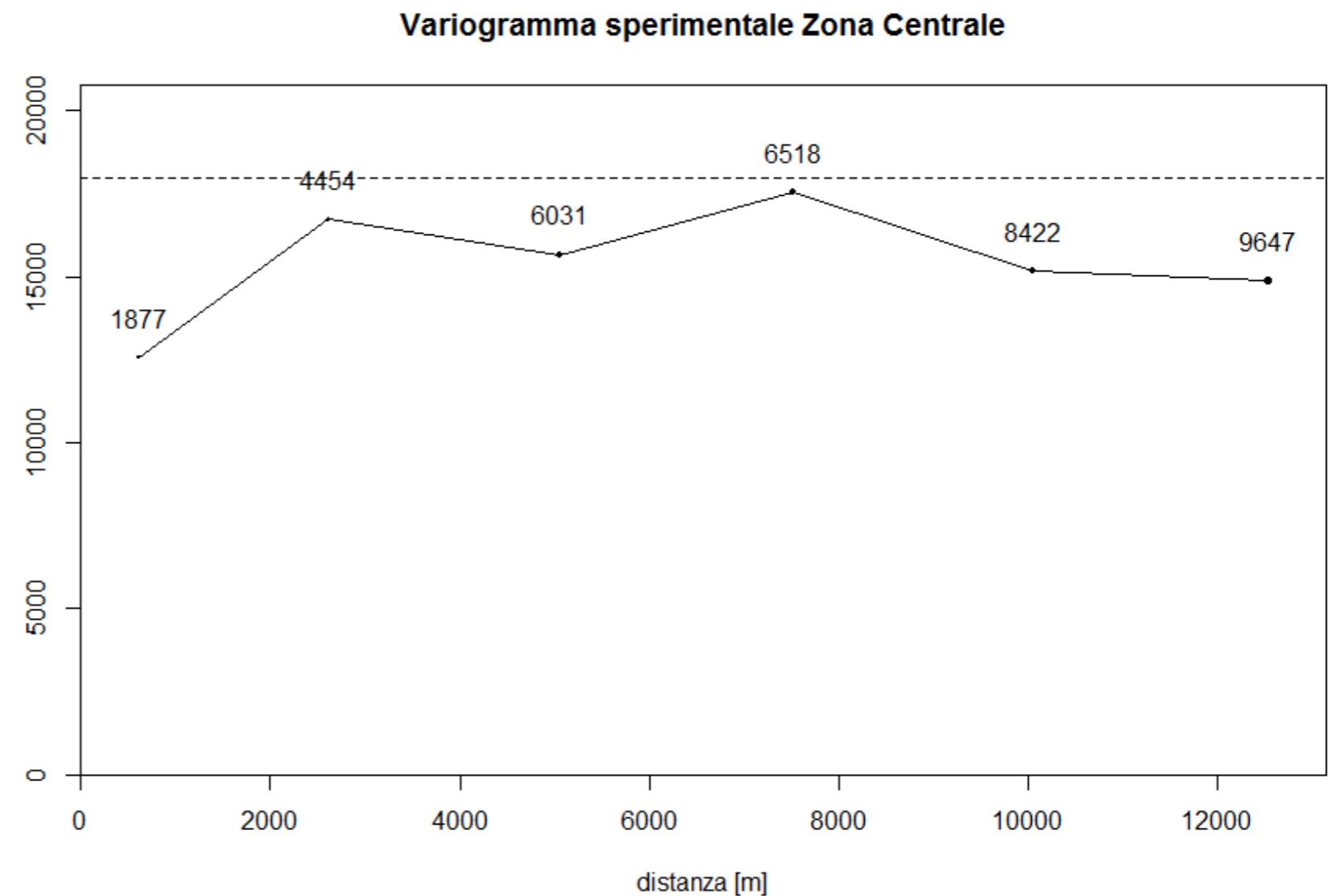
~/elaborazioni/transito R/

```
nb of data = 2915
Nb of active values = 989
Minimum value = 1276.284
Maximum value = 4999.543
Mean value = 4582.577
Standard Deviation = 581.481
Variance = 338119.805
6 - Locator Variable z3 (Name=Nbgh.conc.nbgh3)
Nb of data = 2915
Nb of active values = 989
Minimum value = 30.385
Maximum value = 4834.010
Mean value = 1860.008
Standard Deviation = 1106.181
Variance = 1223636.28
> g1 +labs(x = "concentrazione di attività di radon [Bq/m3]",
+           y = "numerosità",
+           title ="Distribuzione valori di concentrazione Radon indoor") +
+     geom_histogram(binwidth = 15)
> g1 + geom_boxplot(aes(y=conc), notch = T, outlier.colour = "red")
> g1 +labs(x = "concentrazione di attività di radon [Bq/m3]",
+           y = "numerosità",
+           title ="Distribuzione valori di concentrazione Radon indoor") +
+     geom_histogram(binwidth = 15)
>
```

# VARIOGRAMMA Sperimentale (spazio-z)

```
## controllo il variogramma sui dati z -----
lag <- 2500
sel01.vg <- vario.calc(sel01.db,
                        lag = lag,
                        nlag = as.integer(15000/lag),
                        calcul = "vg")

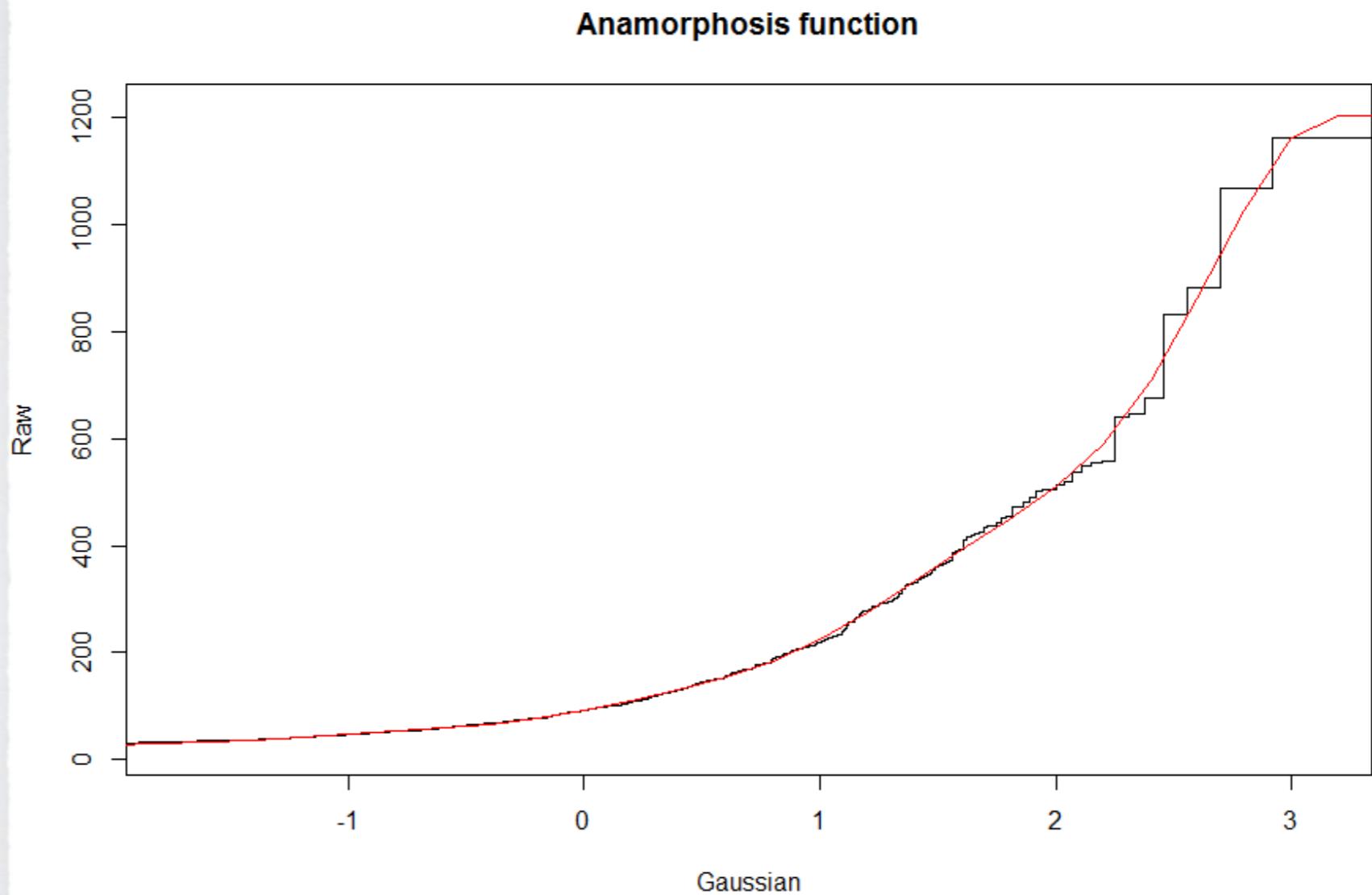
plot(sel01.vg, npairdw=T, npairpt=1,
      title = "Variogramma sperimentale Zona Centrale",
      xlab = "distanza [m]" )
```



# DALLO SPAZIO-Z ALLO SPAZIO-Y

```
##Simulazioni Gaussiane Condizionate :: Turnign Bands Algorithm -----
#creo la funzione anamorfica di trasformazione dallo spazio z allo spazio Y
sel01.anam <- anam.fit(sel01.db, "conc",
                         type = "gaus",
                         ndisc = 30,
                         nbpoly= 30,
                         title = "Anamorphosis function")
plot(sel01.anam, col = "red", add = T)

#passo dallo spazio z allo spazio Y
sel01.db.Y <- anam.z2y(sel01.db, "conc", anam = sel01.anam)
sel01.db.Y
print(sel01.db.Y, flag.stats = T, names = "Gaussian.conc")
```



# VARIOGRAMMA Sperimentale e Modello (spazio-Y)

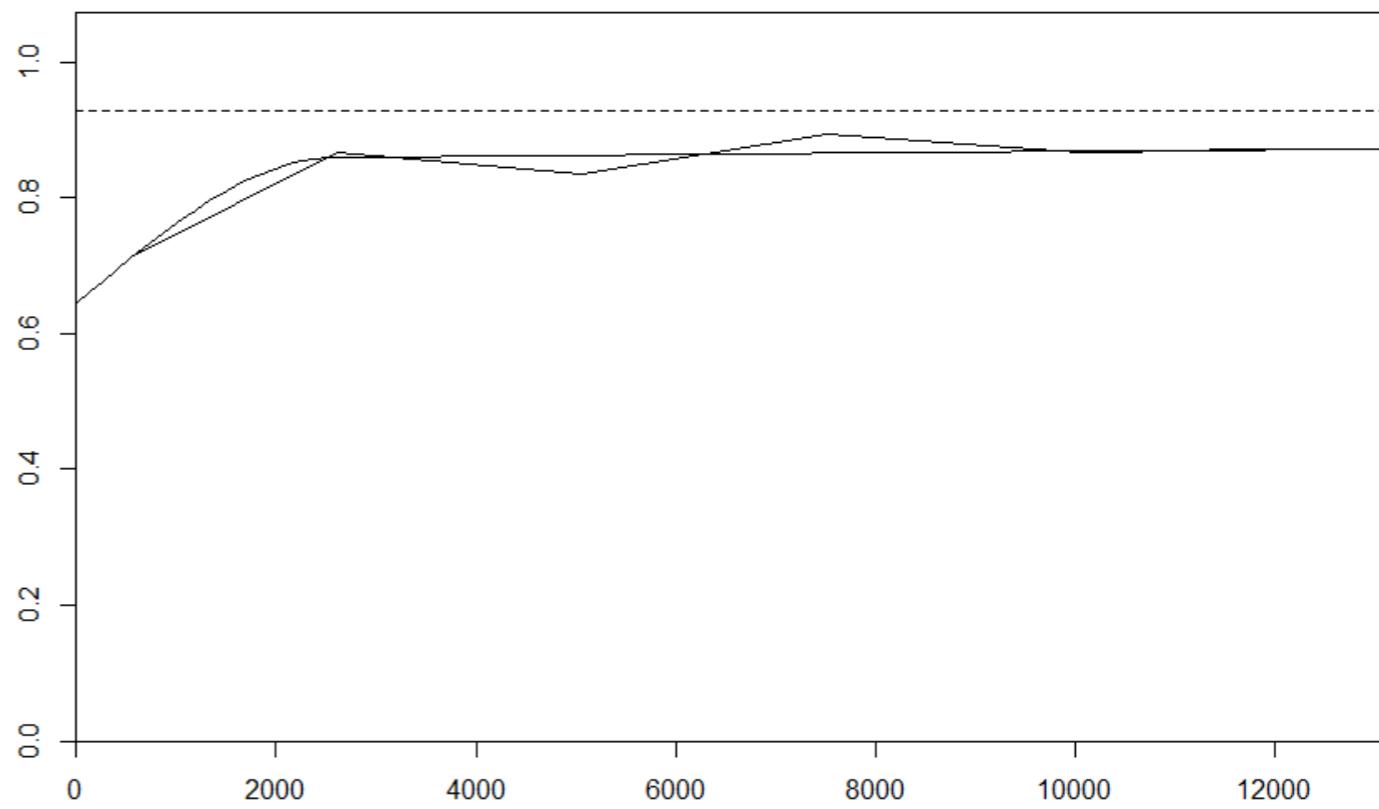
```
#determino il variogramma sperimentale nello spazio Y
lag <- 2500
sel01.vg.Y <- vario.calc(sel01.db.Y,
                           lag = lag,
                           nlag = as.integer(15000/lag),
                           calcul = "vg")
plot(sel01.vg.Y, npairdw=T, npairpt=1,
      title = "Variogramma sperimentale nello spazio Y",
      xlab = "distanza [m]")

#determino il modello di variogramma nello spazio Y
sel01.vg.model.Y <- model.auto(sel01.vg.Y,
                                   struct = c("Nugget Effect",
                                             "Spherical",
                                             "Exponential"),
                                   flag.goulard = T,
                                   flag.noreduce = T,
                                   flag.norm.sill = T,
                                   draw = T,
                                   verbose = 0)
sel01.vg.model.Y
```

```
Model characteristics
-----
Space dimension          = 2
Number of variable(s)   = 1
Number of basic structure(s) = 3
Number of drift function(s) = 1
Number of drift equation(s) = 1

Covariance Part
-----
- Nugget Effect
  sill      = 0.644
- Spherical
  Range     = 2583.257
  sill      = 0.212
- Exponential
  Range     = 315177.450
  Theo. Range = 105208.827
  sill      = 0.144
Total sill    = 1.000

Drift Part
-----
Universality condition
```



# DEFINIZIONE DEI PARAMETRI DELLA GRIGLIA DI SIMULAZIONE E DEL “VICINAGGIO” ➡ SIMULAZIONI ED ELABORAZIONI

```
#definisco la griglia di simulazione e il vicinaggio di ricerca
sel01.grid <- db.grid.init(sel01.db, nodes = c(53,55), margin = 2)
sel01.grid
plot(sel01.grid, asp = 1);plot(sel01.db, col = "black", add=T)

sel01.moving.neigh <- neigh.create(ndim = 2, type = 2,
                                      flag.sector = F,
                                      nmini = 4, #4
                                      nmaxi = 40, #40
                                      radius = 5000) #5000

#analisi del vicinaggio :: N samples, max search radius, min neigh distance
neigh <- neigh.test(dbin = sel01.db,
                     dbout = sel01.grid,
                     model = sel01.vg.model.Y,
                     neigh = sel01.moving.neigh)
print(neigh, flag.stats = T, names = c("*h1", "*h2", "*h3"))

#eseguo le simulazioni
nsim <- 500
sel01.grid <- simtub(dbin = sel01.db.Y,
                      dbout = sel01.grid,
                      model = sel01.vg.model.Y,
                      neigh = sel01.moving.neigh,
                      seed = 4, #0 = non inizializzato
                      nbsimu = nsim,
                      nbtuba = 100)
sel01.grid

#back transform
sel01.grid <- anam.y2z(anam = sel01.anam,
                        db = sel01.grid,
                        names = "Simu.*")

#estratto per elaborazioni dei risultati
sim.results <- db.extract(sel01.grid, c(2,3,(nsim+4):(nsim+nsim+3)))
```

```
#elaboro
#per ogni punto della griglia di simulazione, calcolo
sopra soglia
soglia <- 300
percent <- vector()
sim.results <- na.omit(sim.results)

for (i in 1:dim(sim.results)[1]) {#sulle righe
  over <- 0
  for (j in 3:dim(sim.results)[2]) {#sulle colonne
    if (sim.results[i,j] >= soglia) over <- over + 1
    percent[i] <- (over/(dim(sim.results)[2]-2))*100
  }
}
```

# VISUALIZZAZIONE DEI RISULTATI

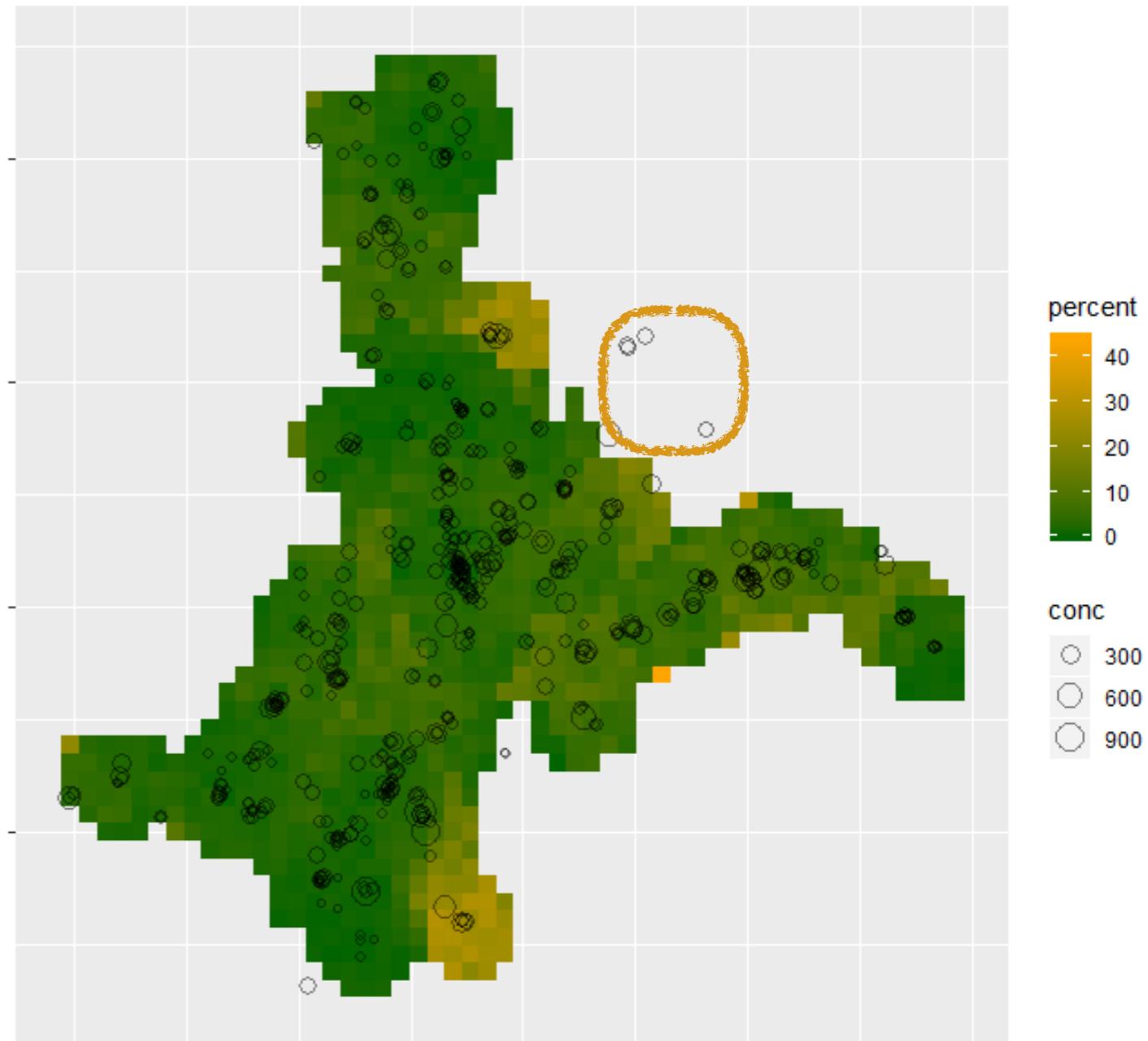
```
#osservo la singola mappa  
ggplot(data = cbind(sim.results[,1:2],percent), aes(x1, x2)) +  
  geom_raster(aes(fill = percent)) +  
  scale_fill_gradient(low = "darkgreen", high = "orange") +  
  coord_equal() +  
  labs(title = "Probabilità di superamento di 300 Bq/m3", x = NULL, y = NULL) +  
  theme(axis.text = element_blank()) +  
  geom_point(data = sel01, aes(X.1, Y.1, size = conc), alpha = 0.5, shape = 1)
```

... punti campionati  
ma senza valori  
simulati nell'intorno?



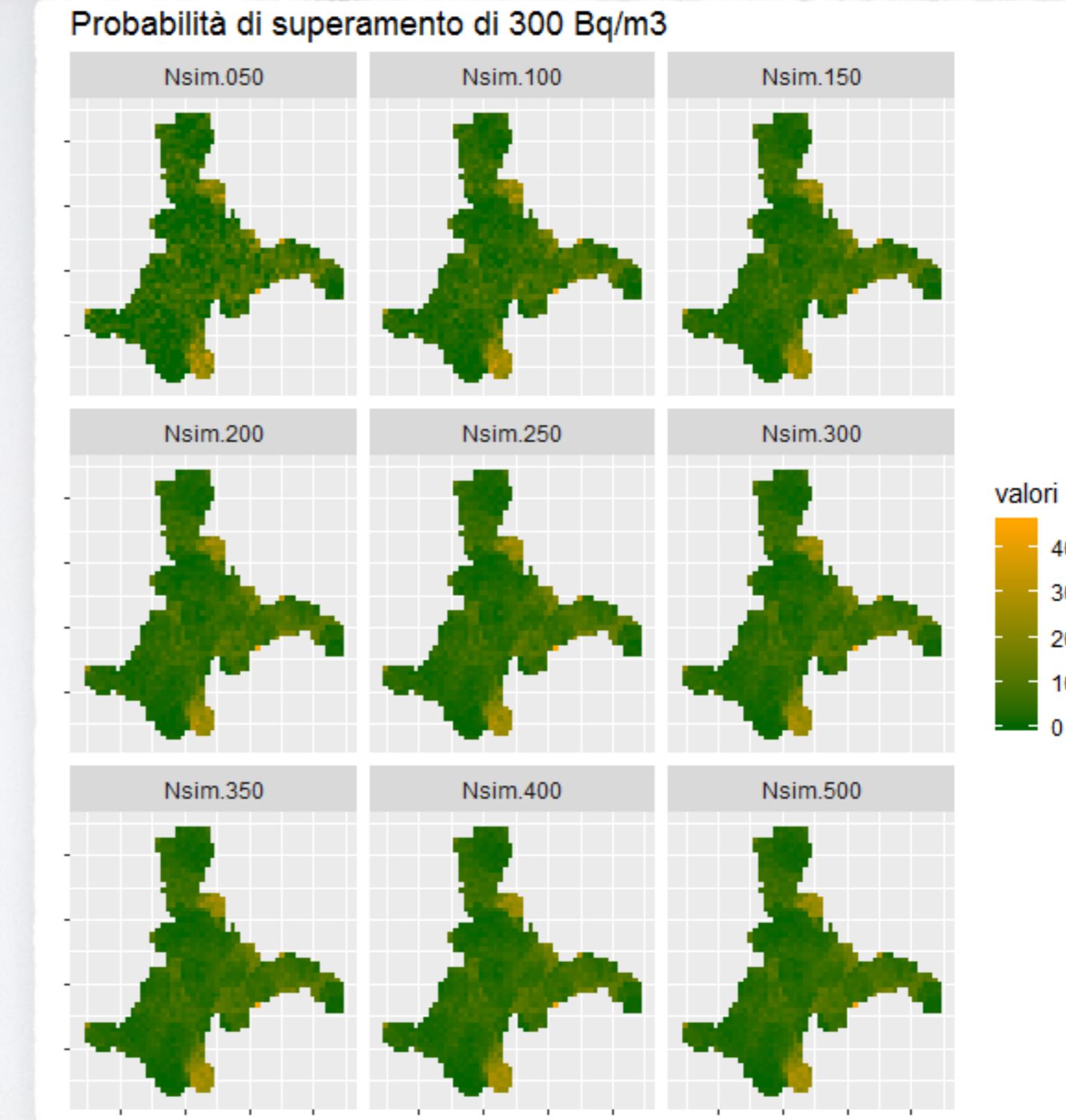
sì, perché per simulare un  
punto della griglia si è  
impostato un numero  
minimo di punti pari a 4  
nel raggio di 5km

Probabilità di superamento di 300 Bq/m3



```
#osservo le mappe - facet
ggplot(data = sim.summary.gather, aes(x1, x2)) +
  geom_raster(aes(fill = valori)) +
  scale_fill_gradient(low = "darkgreen", high = "orange") +
  coord_equal() +
  labs(title = "Probabilità di superamento di 300 Bq/m3", x = NULL, y = NULL) +
  theme(axis.text = element_blank()) +
  facet_wrap(~ tipo, ncol = 3)
```

# ANALISI GRAFICA DELL'INFLUENZA DEI PARAMETRI DI SIMULAZIONE



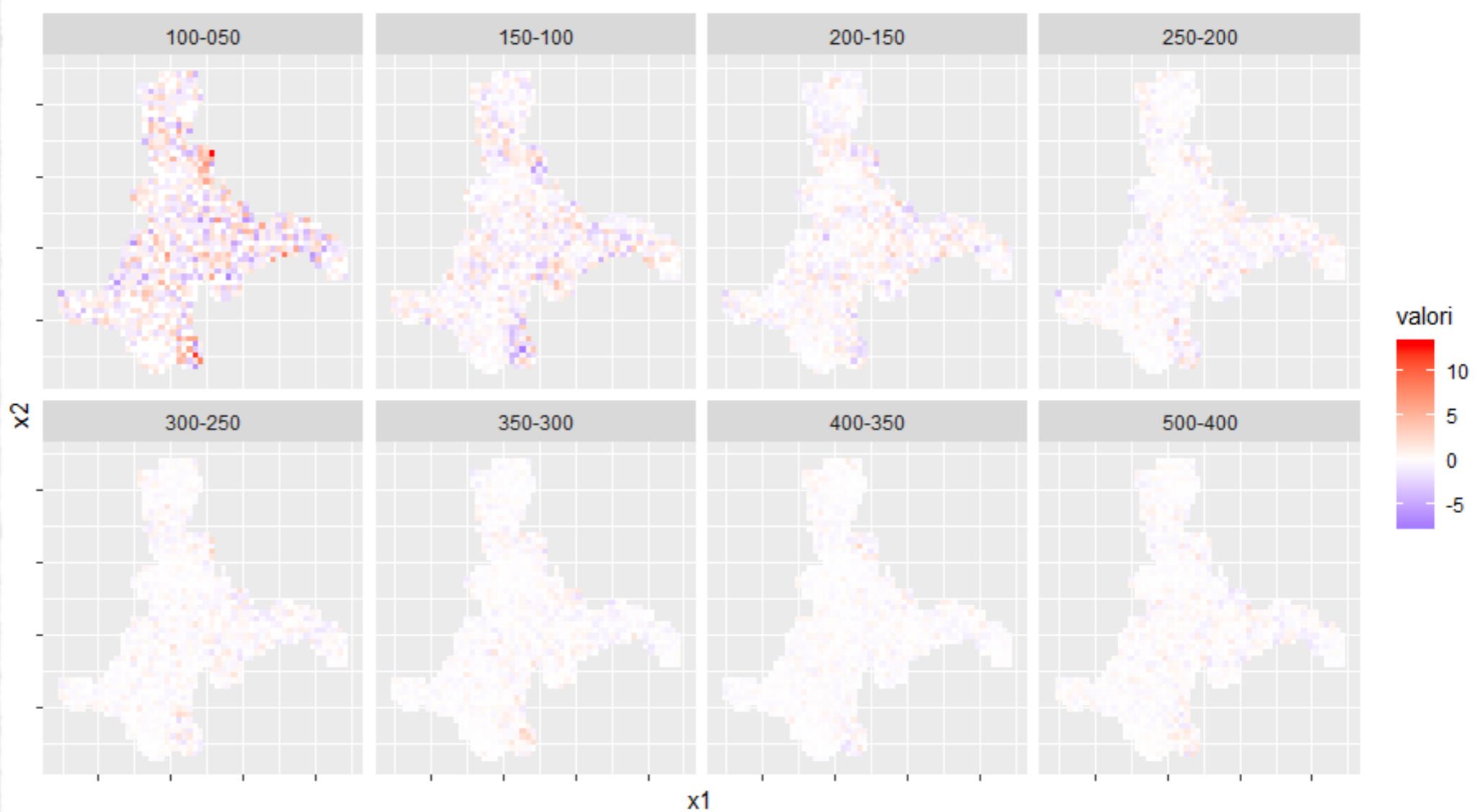
```

diff.pairs <- sim.summary %>%
  transmute("100-050" = Nsim.100 - Nsim.050,
           "150-100" = Nsim.150 - Nsim.100,
           "200-150" = Nsim.200 - Nsim.150,
           "250-200" = Nsim.250 - Nsim.200,
           "300-250" = Nsim.300 - Nsim.250,
           "350-300" = Nsim.350 - Nsim.300,
           "400-350" = Nsim.400 - Nsim.350,
           "500-400" = Nsim.500 - Nsim.400)
diff.pairs <- gather(diff.pairs, diff, valori)
diff.pairs <- cbind(sim.summary[,1:2], diff.pairs)

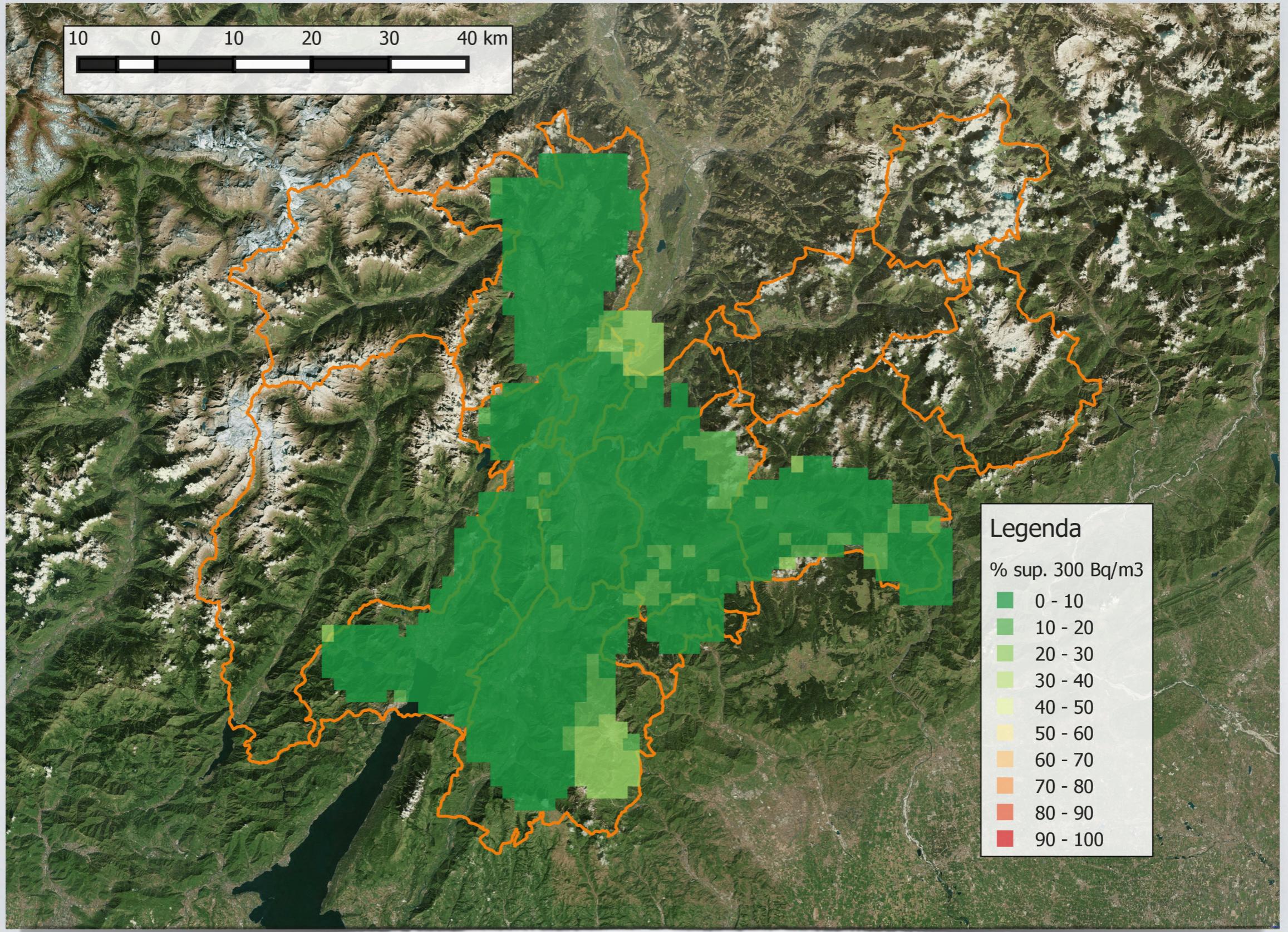
ggplot(data = diff.pairs, aes(x1, x2)) +
  geom_raster(aes(fill = valori)) +
  scale_fill_gradient2(low = "blue", mid = "white", high = "red") +
  coord_equal() +
  facet_wrap(~ diff, ncol = 4) +
  theme(axis.text = element_blank())

```

# ANALISI GRAFICA DELL'INFLUENZA DEI PARAMETRI DI SIMULAZIONE



# ESPORTAZIONE IN QGIS PER UNA INTEGRAZIONE CON ALTRI ASPETTI GEOGRAFICI



תודה Mh'goi .ליתן

Mh'goi Thank you

arigatô danke nandri takk grazie hvala 감사합니다

gracias dziękuję dēkuji kiitos Xièxiè спасибо

euχαριστώ a dank dank u gràcies

tack

•

Сүнрхиақалыңың •

tak ngiyabonga obrigado merci shukran

choukrane Dakujem Баярлалаа diolch blagodaram teşekkür ederim

dhanyavād grazzi faleminderit

dякую Dankie kasih

хвала multumesc

terima

Благодаря köszönöm