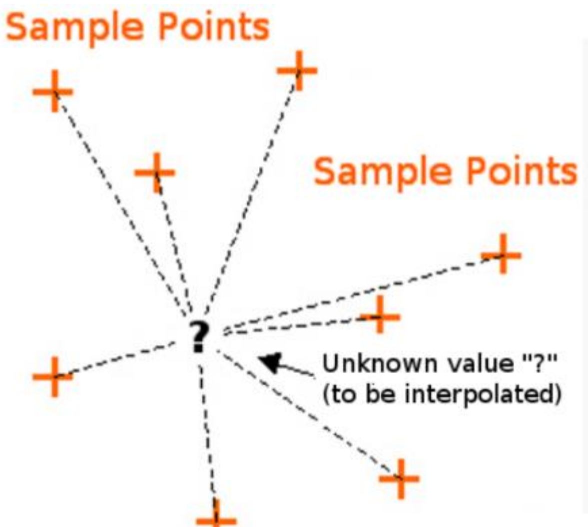
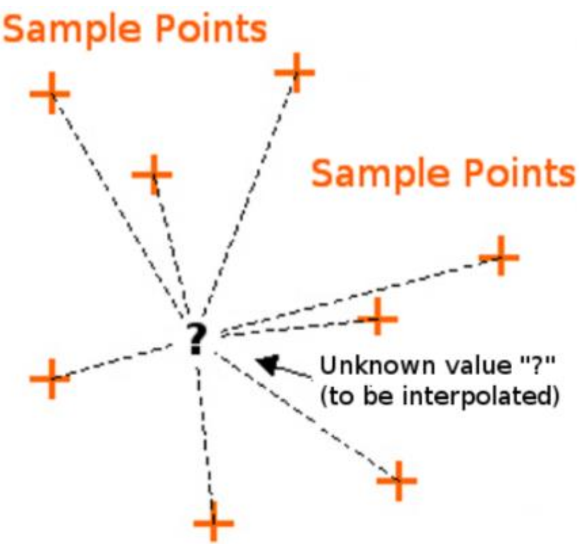
地理統計概述

Geostatistics: A Brief Introduction



授課教師：溫在弘
E-mail: wenthung@ntu.edu.tw

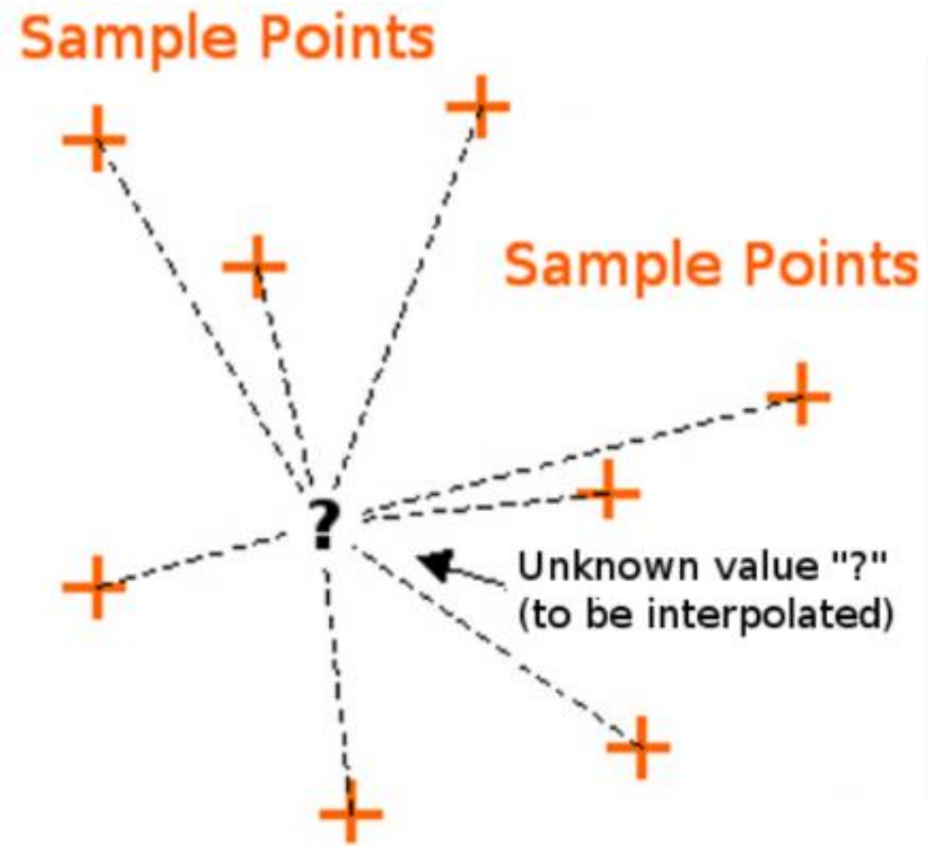
授課大綱:地理統計概述



Spatial Interpolation/Prediction

- Stochastic (geostatistics)
  - Semivariogram and Kriging Method
- Deterministic and Non-parametric
  - Inverse Distance Weighting (IDW)

Geostatistics:
Principles of Spatial Interpolation



Z(x): a regionalized random variable that is associated with a true measurement, z(x), that characterizes the quantity of a variable at point x.

Spatial Covariance vs. Variogram

- **Spatial Covariance**: how that variable is distributed across space, focusing on the degree of similarity among pairs of data points.
- We can define the values of the random variable Z at two locations, Z(x) and Z(x + h), where h represents the distance (spatial lag) between a pair of sampling sites.

$$C(h) = \mathrm{E}[Z(x+h).Z(x)] - \mu^2$$

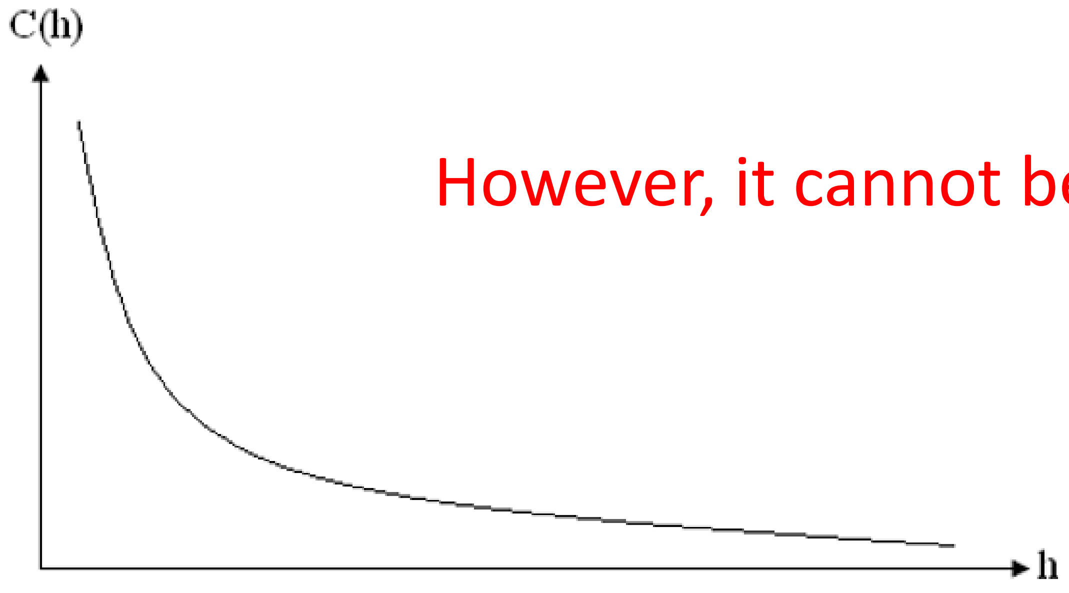
where μ is the stationary mean

Spatial Covariance Function

spatial structure of the data

assumption of stationary covariance

$$C(h) = E[Z(x+h).Z(x)] - \mu^2$$

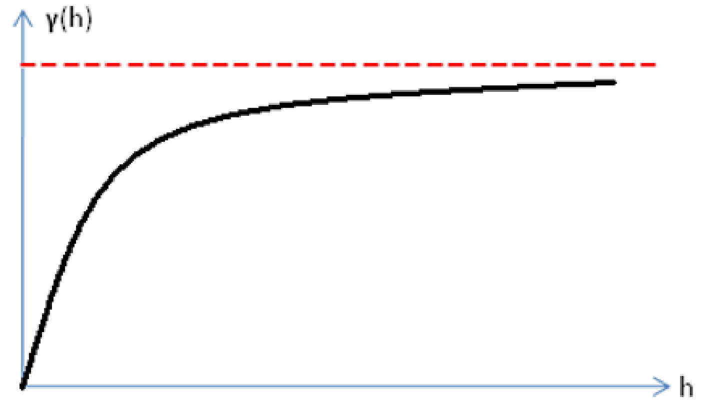


However, it cannot be estimated directly

Variogram

- A variogram might be thought of as "*dissimilarity* between point values as a function of distance", such that the dissimilarity is greater for points that are farther apart

$$2\gamma(h) = E\left\{[Z(x+h) - Z(x)]^2\right\}$$



Variogram γ(h)

Variogram: Mathematical definition

$$2\gamma(h) = E\left\{[Z(x+h) - Z(x)]^2\right\}$$

⬇

$$2\gamma(h) = average\left[\left(Z(i) - Z(j)\right)^2\right]$$

⬇

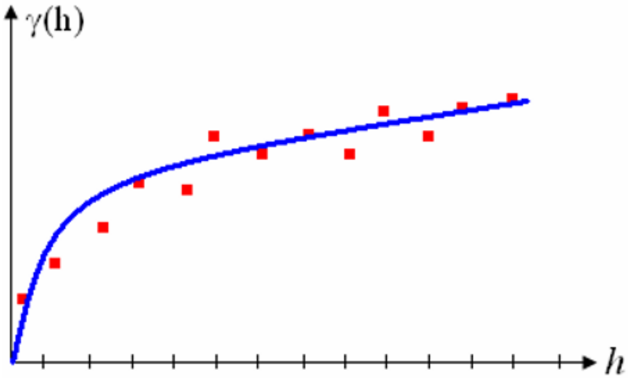
$$2\hat{\gamma}(h) = \frac{1}{N(h)} \sum_{N(h)} (Z(s_i) - Z(s_j))^2,$$

N(h): the number of paired comparisons at lag h.

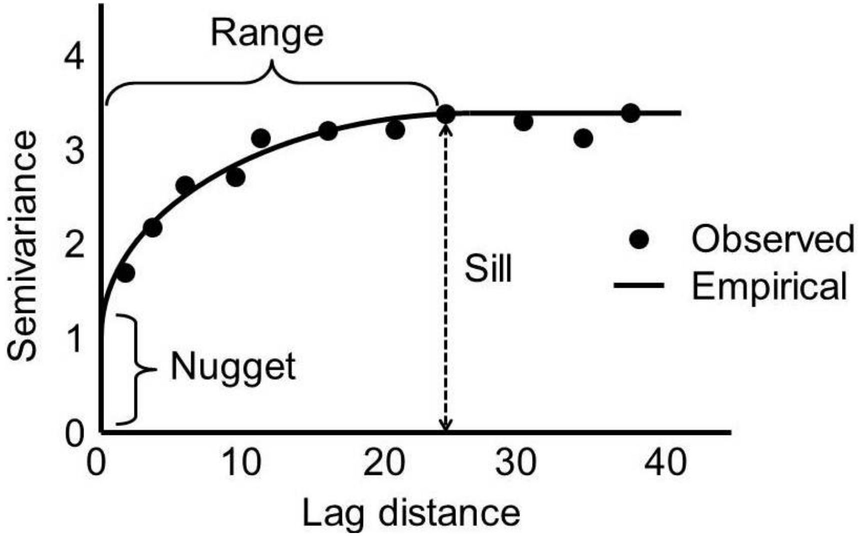
Semivariogram $\gamma(h)$
半變異元圖

$$\gamma(h) = \frac{1}{2n(h)} \sum_{i=1}^{n(h)} \left[z(x_i + h) - Z(x_i)^2 \right]$$

where n is the number of sample points, $Z(x_i)$ is the measured sample value at location x_i, $Z(x_{i+h})$ is the sample value at location x_{i+h}, regionalized variable $Z(x)$, and $n(h)$ is the number of pairs of observations a distance h apart.

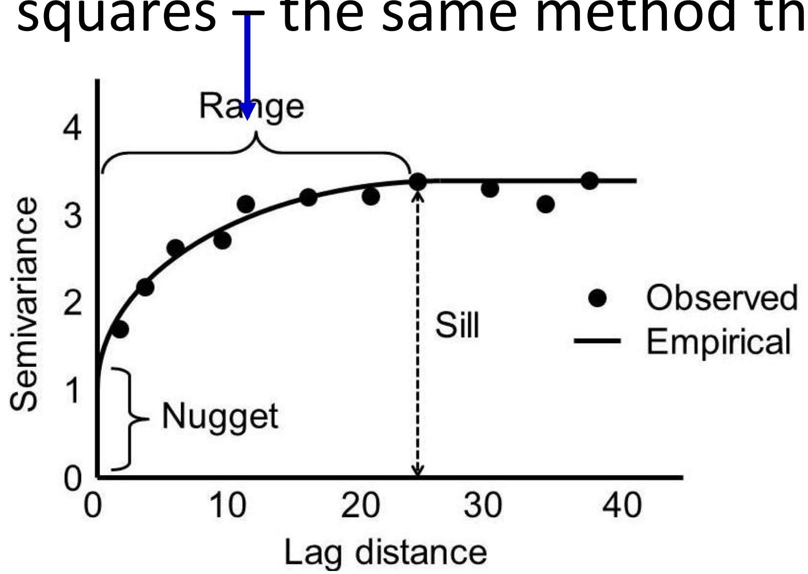


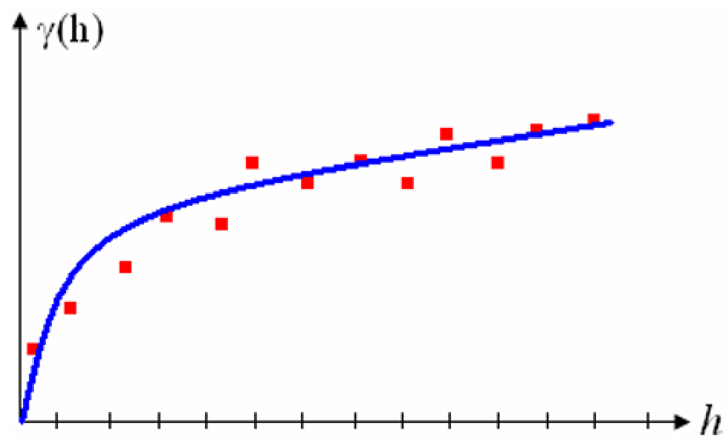
Concept of Semivariogram



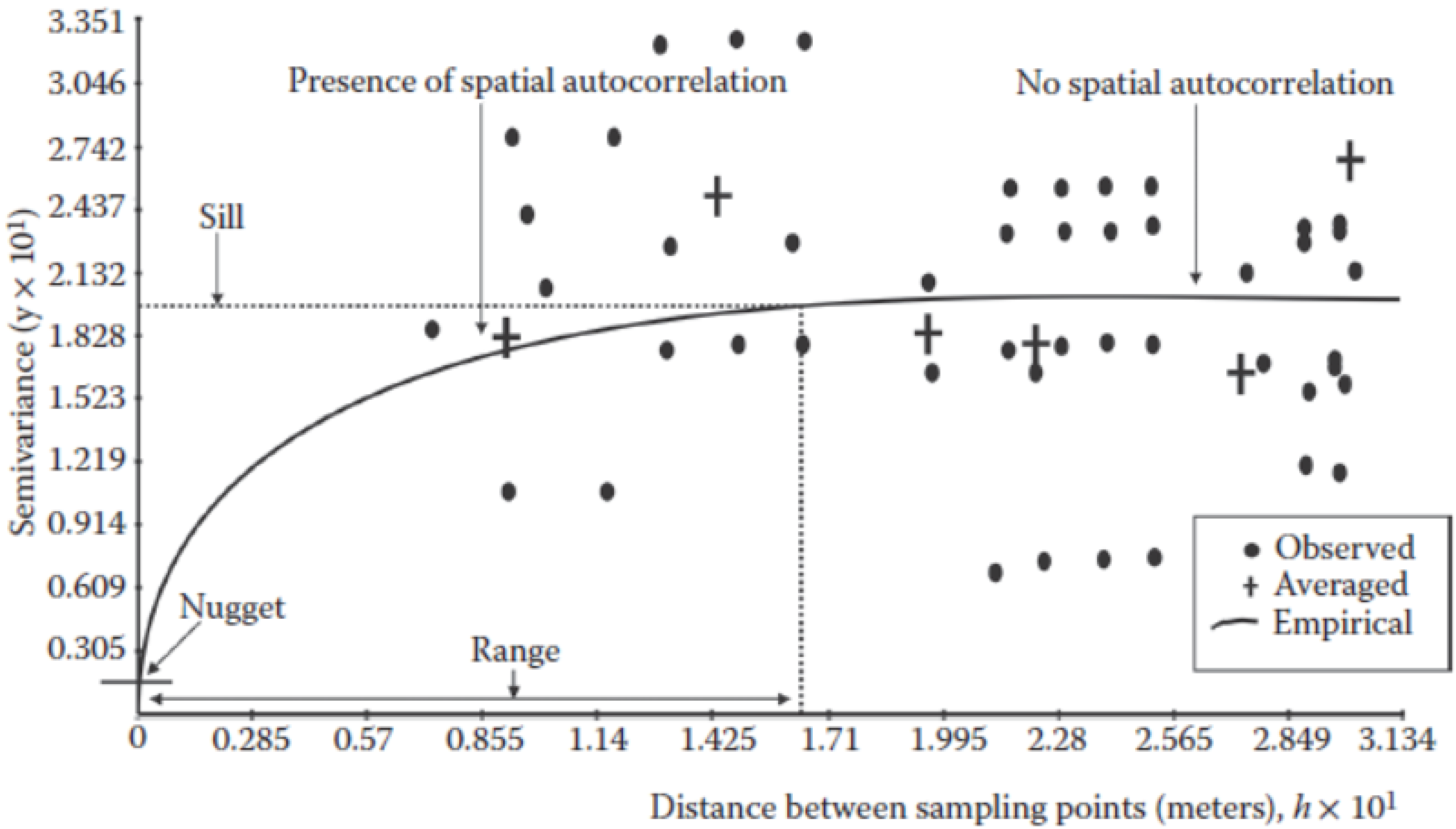
Fitting a Variogram Model

- Now, we're going to fit a variogram model (i.e., curve) to the empirical variogram
- That is, based on the shape of the empirical variogram, different variogram curves might be fit
- The curve fitting generally employs the method of least squares – the same method that's used in regression analysis





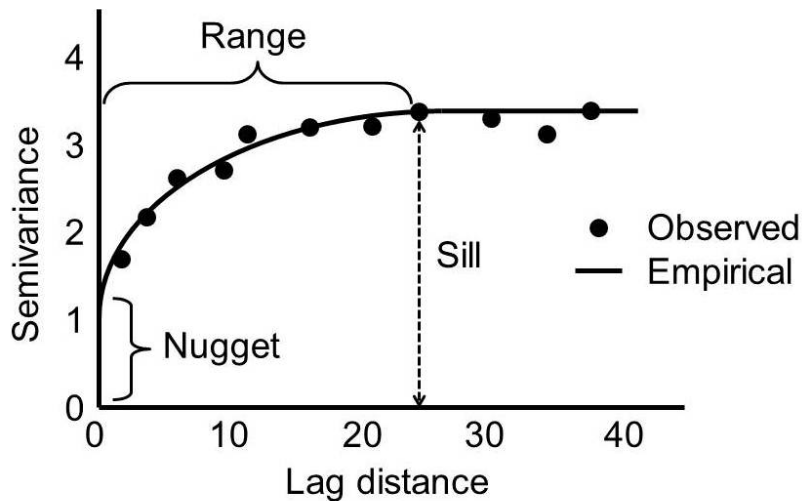
Variogram Model



The Variogram Parameters

- The variogram models are a function of three parameters, known as the range, the sill, and the nugget.
- Semivariance value where it flattens out is called a “sill.”
- The distance range for which there is a slope is called the “*neighborhood*”; this is where there is positive spatial structure
- The intercept is called the “nugget” and represents random noise that is spatially independent

$$\tilde{\gamma}(h) = \frac{1}{2N(h)} \sum_{u=1}^{N} (z(u) - z(u+h))^2$$



Variogram Models

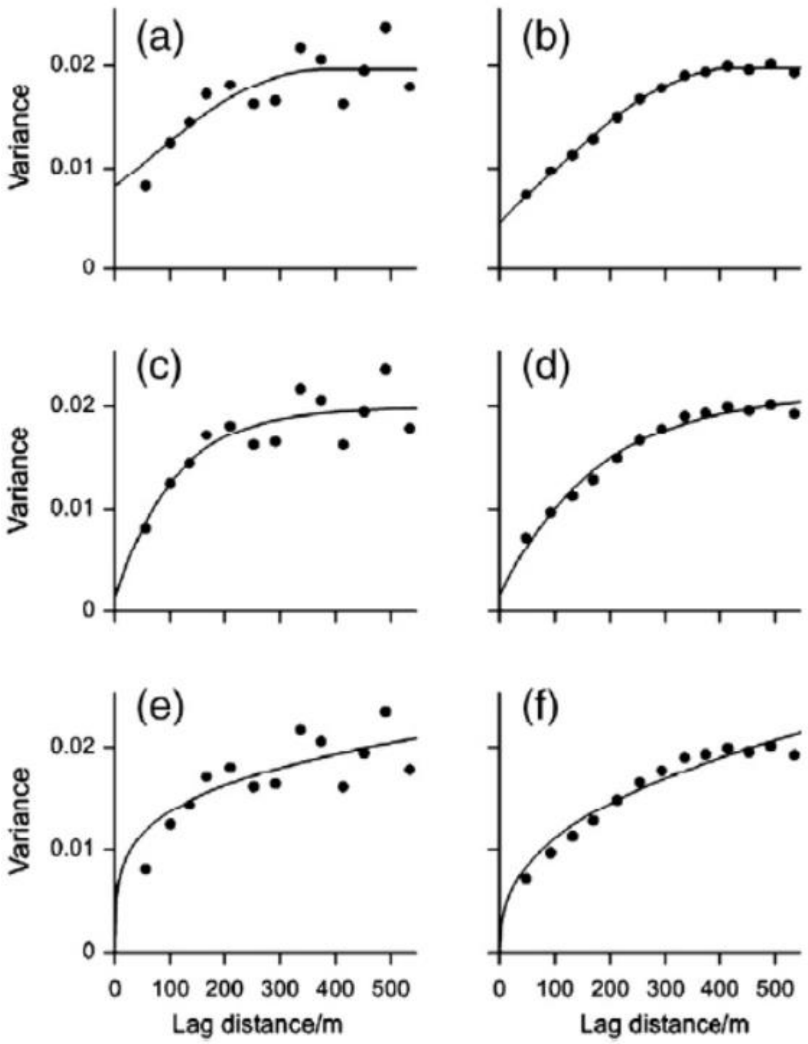
spherical model

$$\gamma(h) = c_0 + c_1\left[\frac{3h}{2a} - \frac{1}{2}\left(\frac{h}{a}\right)^3\right], \quad \text{for } 0 < h < a,$$

$$\gamma(h) = c_0 + c_1, \quad \text{for } h \geq a,$$

exponential model

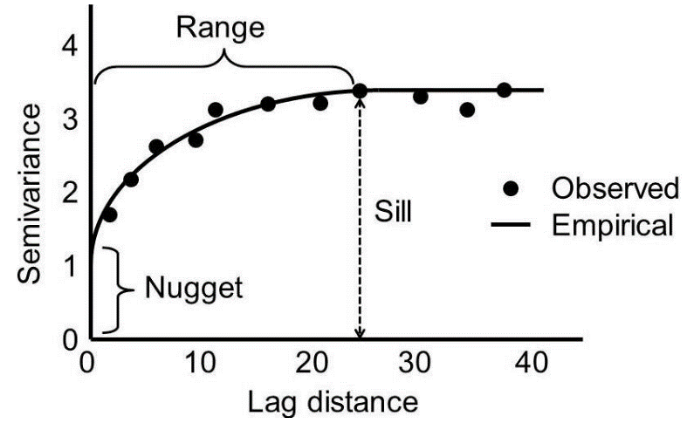
power function



Variogram Models

spherical model

- the most widely used.
- Monotonically non-decreasing: as h increases, the value of $\gamma(h)$ does not decrease - i.e., it goes up (until $h \le r$) or stays the same ($h > r$)



nugget (sill-nugget)

$$\gamma(h) = c_0 + c_1 \left[\frac{3h}{2a} - \frac{1}{2} \left(\frac{h}{a} \right)^3 \right], \quad \text{for } 0 < h < a,$$

range

$$\gamma(h) = c_0 + c_1, \quad \text{for } h \geq a,$$

Variogram Models

$$\gamma(h) = c\left[1 - \exp\frac{-3h}{a}\right]$$

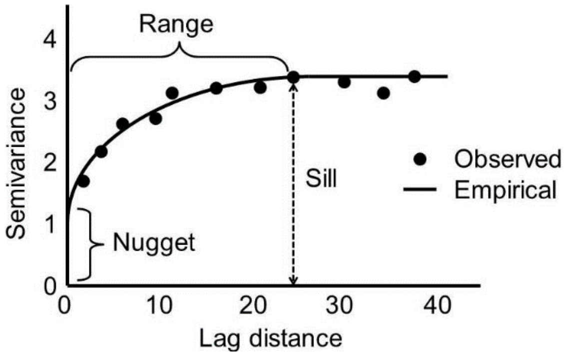
exponential model

- similar to the spherical model, but assumes that the correlation never reaches exactly zero, regardless of how great the distances between points are
- In other words, the variogram approaches the value of the sill asymptotically
- Because the sill is never actually reached, the range is generally considered to be the smallest distance after which the covariance is 5% or less of the maximum covariance
- The model is monotonically increasing
  - I.e., as h goes up, so does $\gamma(h)$

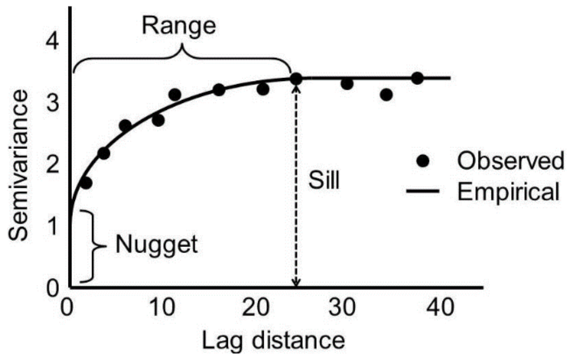
Variogram Models

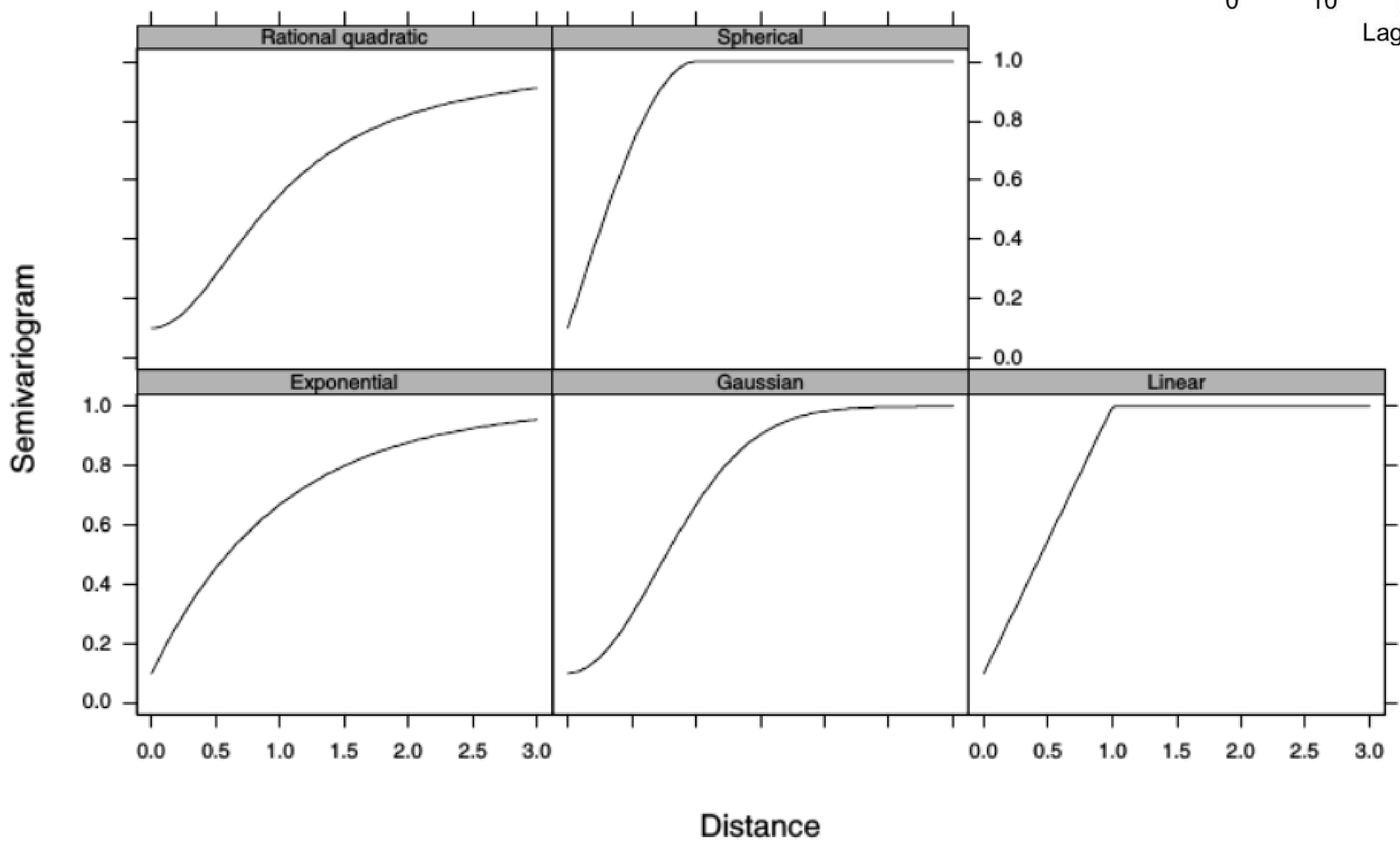
Model Variogram Types

Value	Model variogram type (from VarModel)	Equation
1	Spherical	$\gamma(h) = c\left[1.5\frac{h}{a} - 0.5\left(\frac{h}{a}\right)^3\right]$
2	Exponential	$\gamma(h) = c\left[1 - \exp\frac{-3h}{a}\right]$
3	Gaussian	$\gamma(h) = c\left[1 - \exp\left(\frac{-3h}{a}\right)^2\right]$



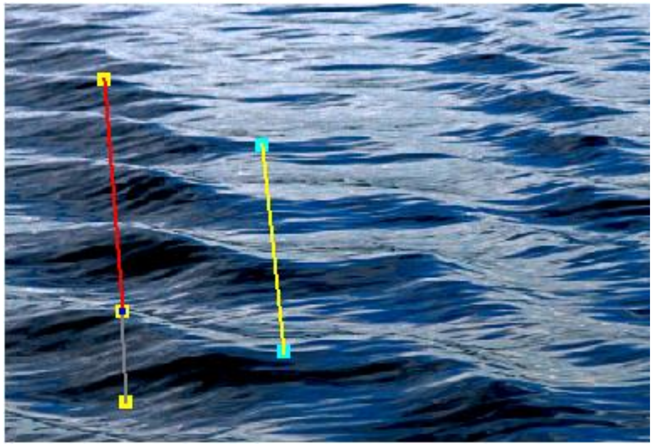
Variogram Models



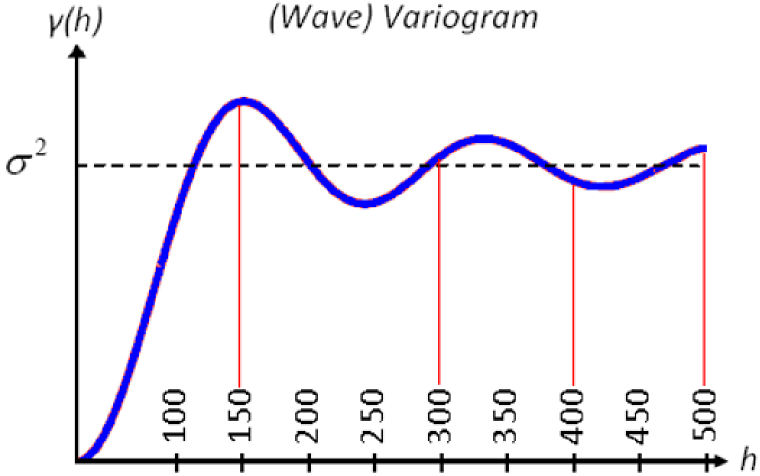


Variogram Models

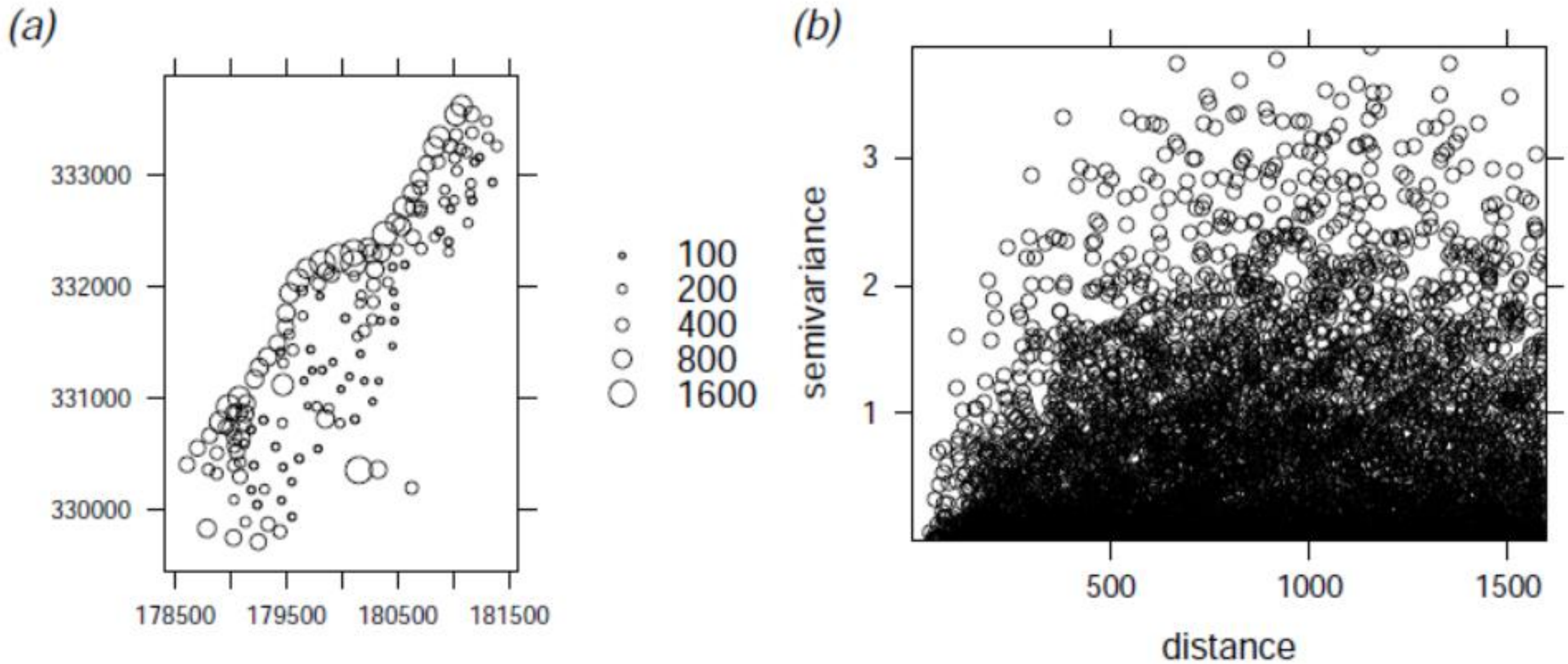
The Wave (Hole-Effect) Model



the waves exhibit a periodic pattern. A non-standard form of spatial autocorrelation applies. Peaks are similar in values to other peaks, and troughs are similar in values to other troughs. However, note the *dampening* in the covariogram and variogram below: That is, *peaks* that are closer together have values that are more correlated than peaks that are father apart (and same holds for troughs).



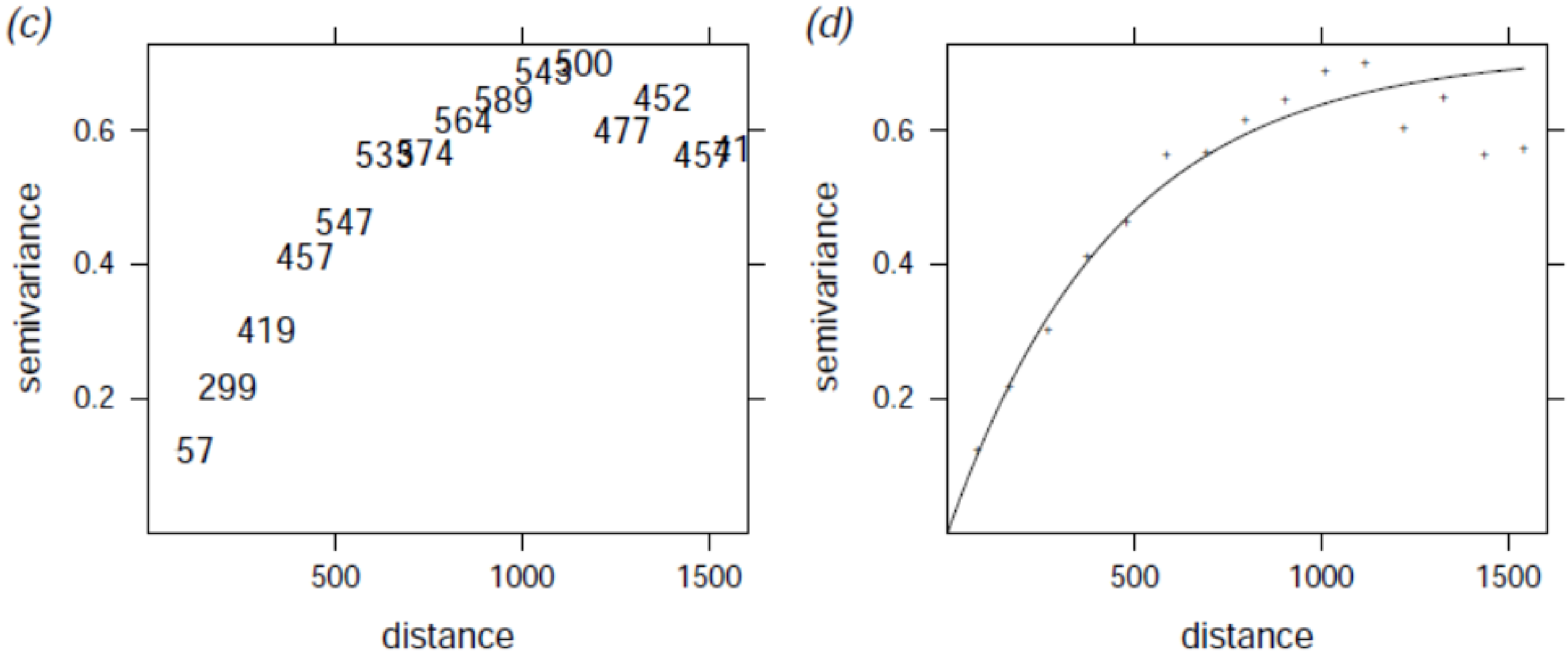
Steps of Variogram Modeling



(a) sampling locations (n=155) and measured variable

(b) variogram cloud showing semivariances for all pairs

Steps of Variogram Modeling (cont'd)



(c) semivariances aggregated to lags of about 100 m

(d) the final variogram model fitting

Lab: Variogram (Exploring data)

- 安裝R套件 gstat

gstat v1.1-6

Other versions ▾

NaN Monthly downloads › 76th Percentile

by Edzer Pebesma

View Source

https://www.rdocumentation.org/packages/gstat Copy

Spatial and Spatio-Temporal Geostatistical Modelling, Prediction and Simulation

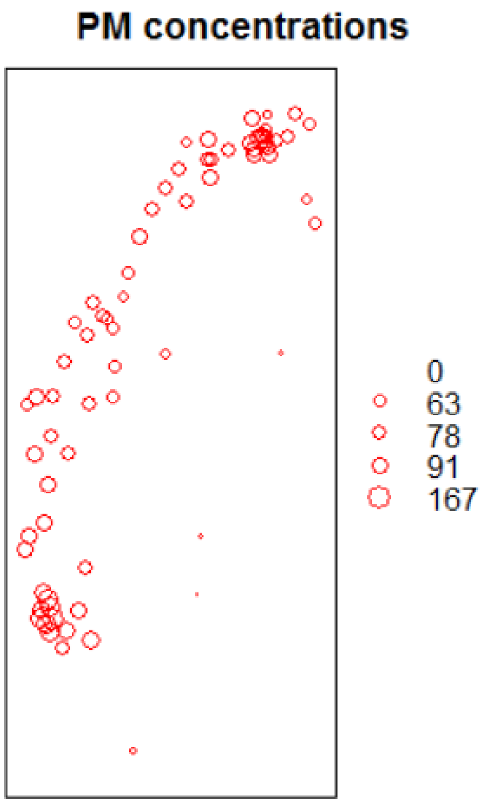
Variogram modelling; simple, ordinary and universal point or block (co)kriging; spatio-temporal kriging; sequential Gaussian or indicator (co)simulation; variogram and variogram map plotting utility functions.

Data: 台灣環保署空氣品質測站資料 (shape file)

73 obs. x 16 variables

	SiteName	SiteEngNam	AreaName	County	Township	SiteAddres	TWD97Lon	TWD97Lat	SiteType	Name	PSI	PM	O3	SO2	CO	NO2
1	二林	Erlin	中部空品區	彰化縣	二林鎮	彰化縣二林鎮萬合里江山巷1號	120.4097	23.92517	一般測站	二林	62	75	40	5.8	0.47	12.0
2	三重	Sanchong	北部空品區	新北市	三重區	新北市三重區三和路重陽路交口	121.4938	25.07261	交通測站	三重	68	102	0	3.0	1.37	36.0
3	三義	Sanyi	竹苗空品區	苗栗縣	三義鄉	苗栗縣三義鄉西湖村上湖61-1號	120.7588	24.38294	一般測站	三義	45	56	32	1.9	0.36	6.3
4	土城	Tucheng	北部空品區	新北市	土城區	新北市土城區學府路一段241號	121.4519	24.98253	一般測站	土城	62	84	30	1.9	0.51	16.0
5	士林	Shilin	北部空品區	臺北市	北投區	臺北市北投區文林北路155號	121.5154	25.10542	一般測站	士林	50	61	32	1.8	0.41	11.0
6	大同	Datong	北部空品區	臺北市	大同區	臺北市大同區重慶北路三段2號	121.5133	25.06320	交通測站	大同	61	78	0	2.3	0.84	20.0
7	大里	Dali	中部空品區	臺中市	大里區	臺中市大里區大新街36號	120.6777	24.09961	一般測站	大里	42	50	37	2.4	0.62	19.0
8	大園	Dayuan	北部空品區	桃園市	大園區	桃園市大園區中正東路160號	121.2018	25.06034	一般測站	大園	62	85	35	3.3	0.37	12.0
9	大寮	Daliao	高屏空品區	高雄市	大寮區	高雄市大寮區潮寮路61號	120.4251	22.56575	一般測站	大寮	72	109	51	8.2	0.91	38.0
10	小港	Xiaogang	高屏空品區	高雄市	小港區	高雄市小港區平和南路185號	120.3377	22.56583	一般測站	小港	86	132	34	7.1	0.72	31.0
11	中山	Zhongshan	北部空品區	臺北市	中山區	臺北市中山區林森北路511號	121.5265	25.06236	一般測站	中山	52	81	14	2.4	1.06	33.0
12	中壢	Zhongli	北部空品區	桃園市	中壢區	桃園市中壢區延平路622號	121.2217	24.95328	交通測站	中壢	67	83	26	2.4	0.91	23.0
13	仁武	Renwu	高屏空品區	高雄市	仁武區	高雄市仁武區八卦里永仁街555號	120.3326	22.68906	一般測站	仁武	91	144	32	2.7	0.70	26.0
14	斗六	Douliu	雲嘉南空品區	雲林縣	斗六市	雲林縣斗六市民生路224號	120.5450	23.71185	一般測站	斗六	59	71	41	2.5	0.49	14.0
15	冬山	Dongshan	宜蘭空品區	宜蘭縣	冬山鄉	宜蘭縣冬山鄉南興村照安路26號	121.7929	24.63220	一般測站	冬山	50	49	29	2.1	0.33	12.0
16	古亭	Guting	北部空品區	臺北市	大安區	臺北市大安區羅斯福路三段153號	121.5296	25.02061	一般測站	古亭	45	87	21	0.0	0.67	24.0
17	左營	Zuoying	高屏空品區	高雄市	左營區	高雄市左營區翠華路687號	120.2929	22.67486	一般測站	左營	81	117	40	2.5	0.59	17.0
18	平鎮	Pingzhen	北部空品區	桃園市	平鎮區	桃園市平鎮區文化街189號	121.2040	24.95279	一般測站	平鎮	60	71	31	1.6	0.44	12.0

R code: loading data



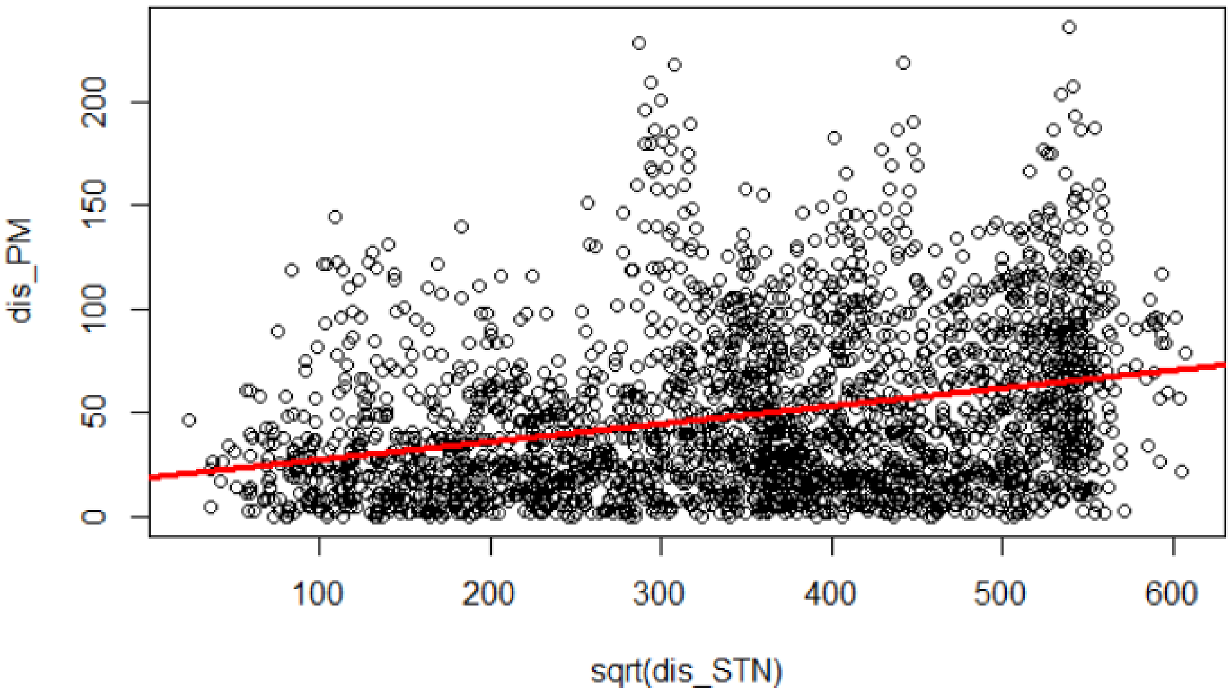

```
library(rgdal)

rm(list=ls())
setwd("D:/R_Labs")

EPA_STN <- readOGR(dsn = "EPA", layer = "EPA_STN2", encoding="utf-8")
plot(EPA_STN); head(EPA_STN)

data= EPA_STN@data
PMdata=EPA_STN@data["PM"]

bubble(EPA_STN, "PM", col="red", fill=FALSE, maxsize = 1.5, main = "PM concentrations")
```

Exploring distance vs. variance

$$2\gamma(h) = average\left[\left(Z(i) - Z(j)\right)^2\right]$$

Dist.(h) vs. $[z(x+h)-z(x)]^2$

variogram cloud



Exploring distance vs. variance (R code)

$$2\gamma(h) = average\left[\left(Z(i) - Z(j)\right)^2\right]$$

Dist.(h) vs. $[z(x+h)-z(x)]^2$

```
x= coordinates(EPA_STN)[,1]
y= coordinates(EPA_STN)[,2]

STNDF = cbind(x,y)
dis_STN= dist(STNDF)

pm= EPA_STN@data[,12]

PMDF= cbind(pm,pm)
dis_PM = dist(PMDF)

plot(dis_PM~sqrt(dis_STN))
abline(lm(dis_PM~sqrt(dis_STN)), lwd=3, col='red')
```

Using variogram() function in R

variogram {gstat} R Documentation

Calculate Sample or Residual Variogram or Variogram Cloud

Description

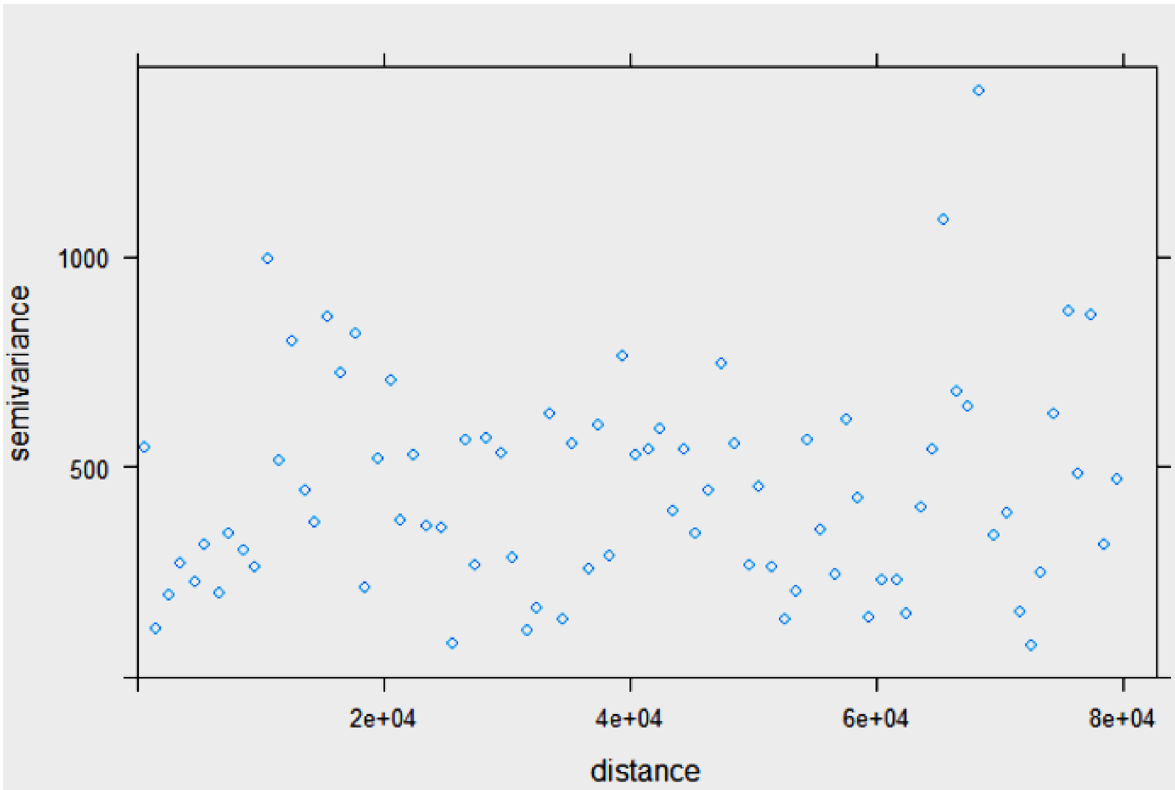
Calculates the sample variogram from data, or in case of a linear model is given, for the residuals, with options for directional, robust, and pooled variogram, and for irregular distance intervals.

In case spatio-temporal data is provided, the function `variogramST` is called with a different set of parameters.

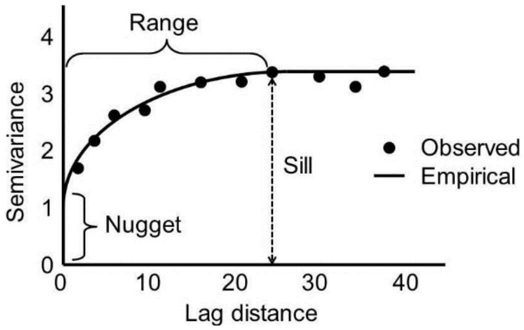
```
library(gstat)
pm.vgm = variogram(PM~1, EPA_STN,cutoff=80000, width=1000)
```

Using variogram() function in R

```
library(gstat)
pm.vgm = variogram(PM~1, EPA_STN,cutoff=80000, width=1000)
plot(pm.vgm)
```



Fitting a Variogram Model



fit.variogram {gstat}

Fit a Variogram Model to a Sample Variogram

Description

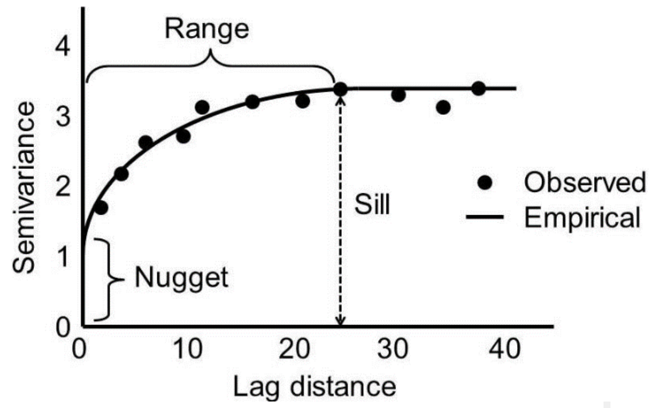
Fit ranges and/or sills from a simple or nested variogram model to a sample variogram

Usage

```
fit.variogram(object, model, fit.sills = TRUE, fit.ranges = TRUE,
        fit.method = 7, debug.level = 1, warn.if.neg = FALSE, fit.kappa = FALSE)
```

Sill, func. range, nugget

```
pm.fit = fit.variogram(pm.vgm, model = vgm(1000, "Exp",2000,1) )
```



vgm {gstat}

Generate, or Add to Variogram Model

Description

Generates a variogram model, or adds to an existing model. `print.variogramModel` prints the essence of a variogram model.

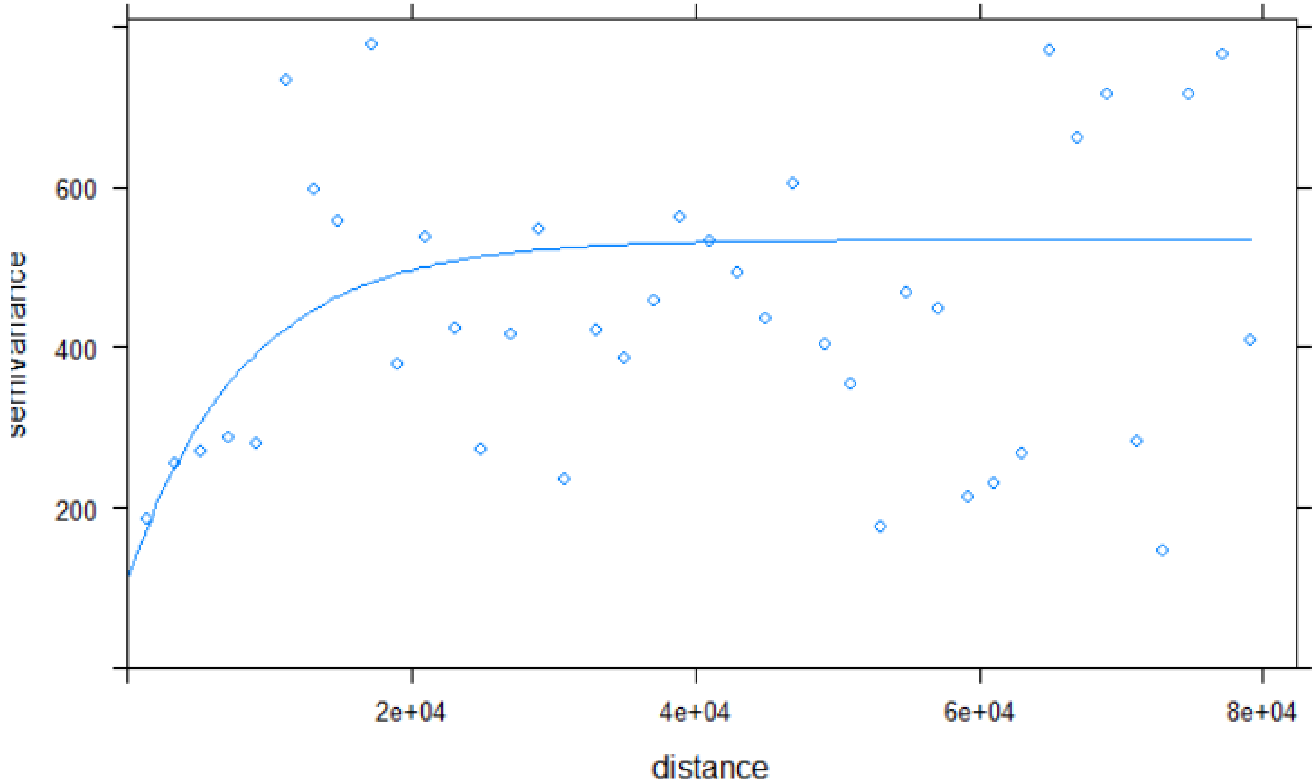
Usage

```
vgm(psill = NA, model, range = NA, nugget, add.to, anis, kappa = 0.5, ..., covtabl
        Err = 0)
## S3 method for class 'variogramModel'
print(x, ...)
as.vgm.variomodel(m)
```

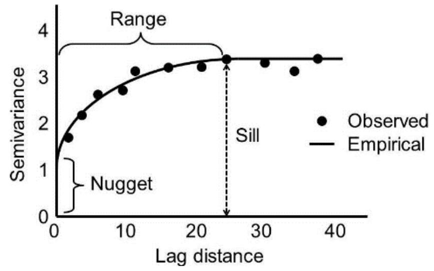
Sill, func. range, nugget

```
pm.fit = fit.variogram(pm.vgm, model = vgm(1000, "Exp",2000,1) )
```

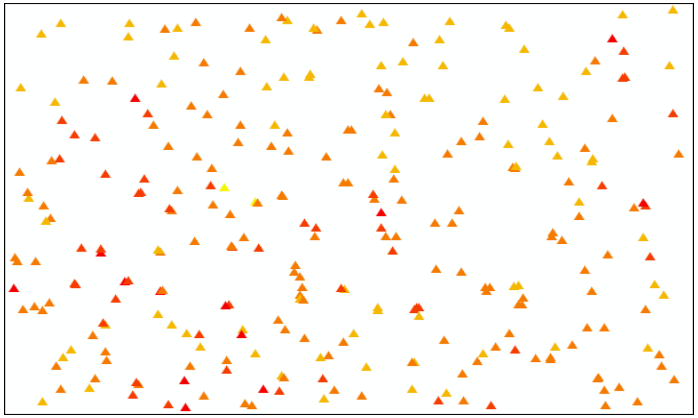
```r
library(gstat)
pm.vgm = variogram(PM~1, EPA_STN,cutoff=80000, width=2000)
pm.fit = fit.variogram(pm.vgm, model = vgm(1000, "Exp",20000,1) )
plot(pm.vgm,pm.fit)
```



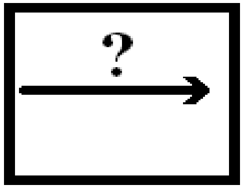
Geostatistical Approach to Spatial Interpolation: using semivariogram



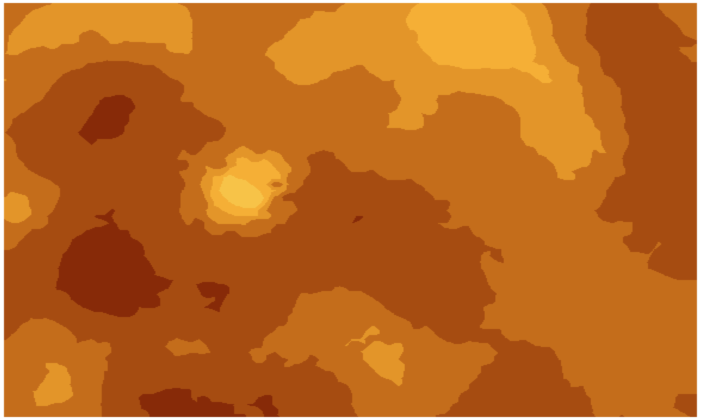
Input



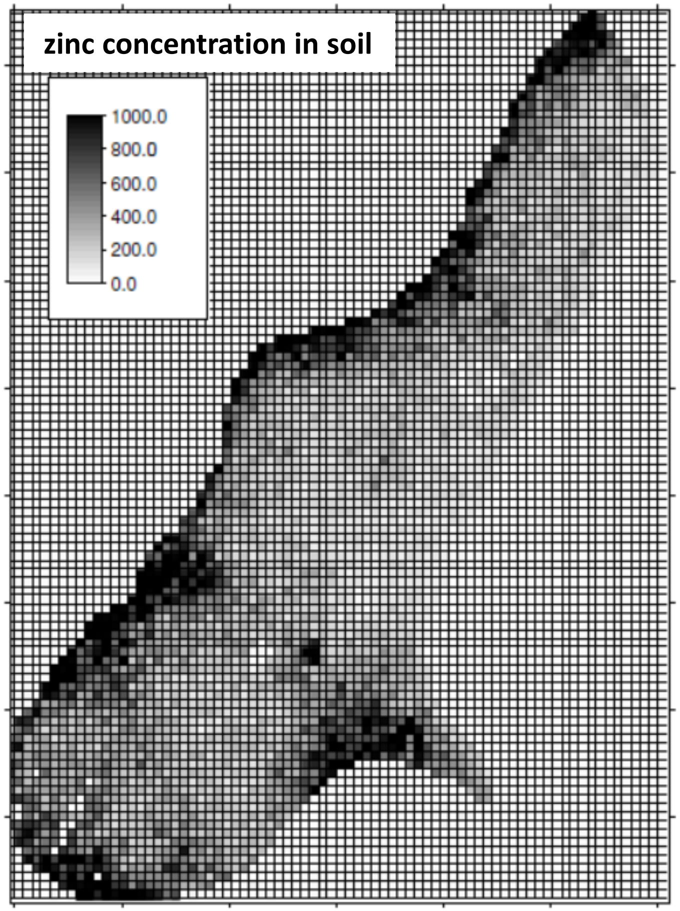
Process



Output



Basics of Geostatistics: Understanding underlying spatial structures



- there seems to be a spatial pattern of how the values change;
- values that are closer together are more similar;
- locally, the values can differ without any systematic rule (randomly)

Universal Model of Variation

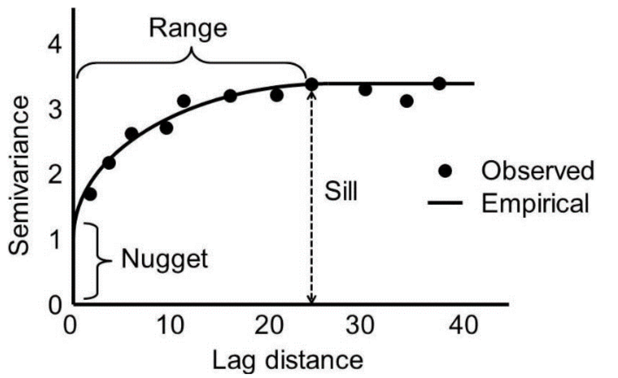
$$Z(\mathbf{s}) = Z^*(\mathbf{s}) + \varepsilon'(\mathbf{s}) + \varepsilon''$$

- $Z^*(s)$ is the deterministic component
- $\varepsilon'(s)$ is the **spatially correlated random component**
- ε'' is the pure noise, e.g. the measurement error.

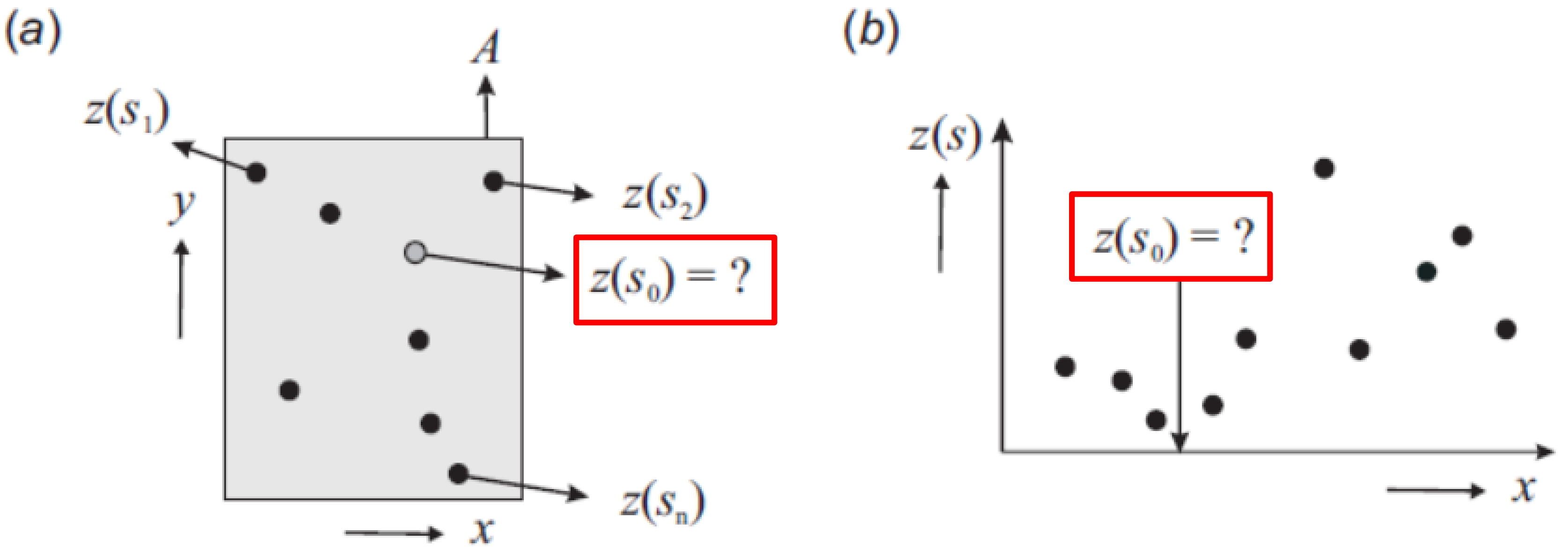
Ordinary Kriging

$$Z(\mathbf{s}) = Z^*(\mathbf{s}) + \varepsilon'(\mathbf{s}) + \varepsilon''$$

- Assumptions
  - *Z(s)* should be normally distributed
  - The global structure $Z^*(s)$ is constant and unknown
  - Covariance between values *of* $\varepsilon'(s)$ depends *only* on distance between the points



Principle of Spatial Prediction



$$\hat{z}_{OK}(\mathbf{s}_0) = \sum_{i=1}^{n} w_i(\mathbf{s}_0) \cdot z(\mathbf{s}_i) = \lambda_0^T \cdot \mathbf{z}$$

where λ_0 is the vector of kriging weights (w_i),

$\mathbf{z}$ is the vector of n observations at primary locations.

不偏估計 unbiased

The weighted linear estimator for location s_0 is: $\hat{Z}_0 = \sum_{i=1}^{n} w_i Z_i$ (*)

The estimation error at location s_0 is the difference between the predictor and the random variable modeling the true value at that location:

$$R_0 = \hat{Z}_0 - Z_0 = \sum w_i Z_i - Z_0$$

The bias is:

$$\begin{aligned} E(R_0) &= E\left(\sum w_i Z_i - Z_0\right) = \sum E(w_i Z_i) - E(Z_0) \\ &= \sum w_i E(Z_i) - E(Z_0) = \sum w_i \mu - \mu = \mu\left(\sum w_i - 1\right) \end{aligned}$$

So, as long as $\sum w_i = 1,$ the weighted linear estimator (*) is unbiased.

However, how to estimate the weight?

Minimizing the mean squared error (MSE)

Kriging is such a method that determines the weights so that the mean squared error (MSE) is minimized:

$$MSE = E\left((\hat{Z}_0 - Z_0)^2\right)$$

subject to the unbiasedness constraint $\sum w_i = 1.$

The final ordinary kriging system is:

$$\underset{(n+1)\times(n+1)}{\underbrace{\begin{bmatrix} C_{11} & \dots & C_{1n} & 1 \\ \dots & \dots & \dots & \dots \\ C_{n1} & \dots & C_{nn} & 1 \\ 1 & \dots & 1 & 0 \end{bmatrix}}_{C}} \bullet \underset{(n+1)\times 1}{\underbrace{\begin{bmatrix} w_1 \\ \dots \\ w_n \\ \lambda \end{bmatrix}}_{w}} = \underset{(n+1)\times 1}{\underbrace{\begin{bmatrix} C_{10} \\ \dots \\ C_{n0} \\ 1 \end{bmatrix}}_{D}} \Longrightarrow w = C^{-1}D$$

如何求解極端值?

Minimizing $f(x_1, x_2)$

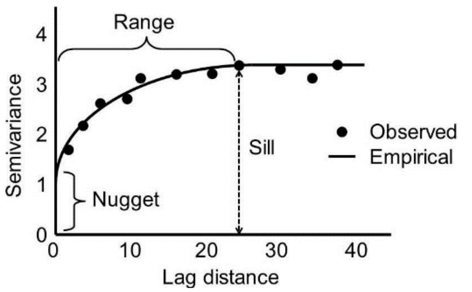
Subject to (受限於) $g(x_1, x_2) = 0$

Lagrangian function (L) $L(x_1, x_2, \lambda) = f(x_1, x_2) + \lambda g(x_1, x_2)$

Lagrange multiplier

$$\frac{\partial L}{\partial x_1} = 0, \ \frac{\partial L}{\partial x_2} = 0, \ \frac{\partial L}{\partial \lambda} = 0$$

Kriging Method



Assume we have a model:

$$Z(\mathbf{s}) = \mu + \varepsilon(\mathbf{s}),$$

where $\varepsilon(\mathbf{s})$ is a zero mean second-order stationary random field with covariogram function $C(h)$ and variogram $\gamma(h)$. Also $\sigma^2 = C(0)$.

The weighted linear estimator for location $\mathbf{s}_0$ is: $\hat{Z}_0 = \sum_{i=1}^{n} w_i Z_i$

$$\underset{(n+1)\times(n+1)}{\Gamma} \quad \underset{(n+1)\times 1}{w} = \underset{(n+1)\times 1}{D}$$

$$\begin{bmatrix} 0 & \gamma_{12} & \dots & \gamma_{1n} & 1 \\ \gamma_{21} & 0 & \dots & \gamma_{2n} & 1 \\ \dots & \dots & \dots & \dots & \dots \\ \gamma_{n1} & \gamma_{n2} & \dots & 0 & 1 \\ 1 & 1 & \dots & 1 & 0 \end{bmatrix} \bullet \begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_n \\ -\lambda \end{bmatrix} = \begin{bmatrix} \gamma_{10} \\ \gamma_{20} \\ \dots \\ \gamma_{n0} \\ 1 \end{bmatrix} \Longrightarrow w = \Gamma^{-1} D$$

$(n+1)\times(n+1)$ $(n+1)\times 1$ $(n+1)\times 1$

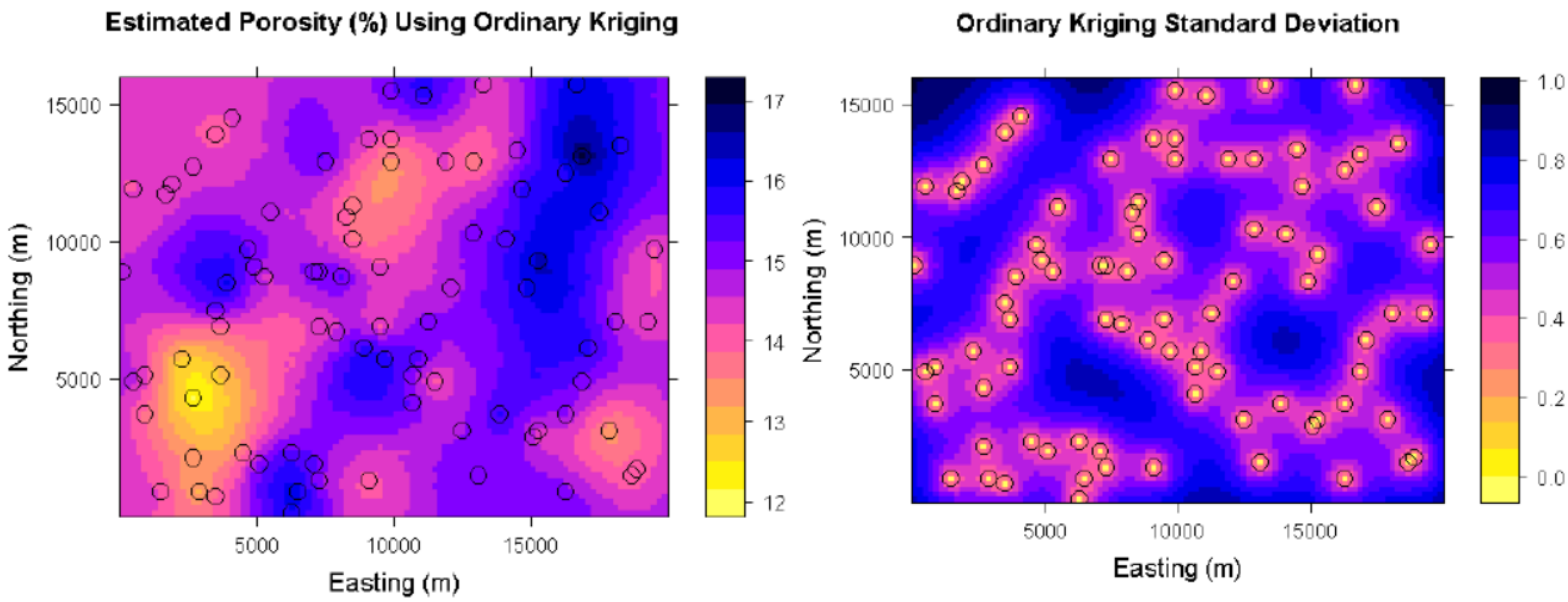
kriging Variance

$$\underset{(n+1)\times(n+1)}{\overset{\Gamma}{\begin{bmatrix} 0 & \gamma_{12} & \dots & \gamma_{1n} & 1 \\ \gamma_{21} & 0 & \dots & \gamma_{2n} & 1 \\ \dots & \dots & \dots & \dots & \dots \\ \gamma_{n1} & \gamma_{n2} & \dots & 0 & 1 \\ 1 & 1 & \dots & 1 & 0 \end{bmatrix}}} \bullet \underset{(n+1)\times 1}{\overset{w}{\begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_n \\ -\lambda \end{bmatrix}}} = \underset{(n+1)\times 1}{\overset{D}{\begin{bmatrix} \gamma_{10} \\ \gamma_{20} \\ \dots \\ \gamma_{n0} \\ 1 \end{bmatrix}}} \Rightarrow w = \Gamma^{-1} D$$

kriging variance in terms of variogram

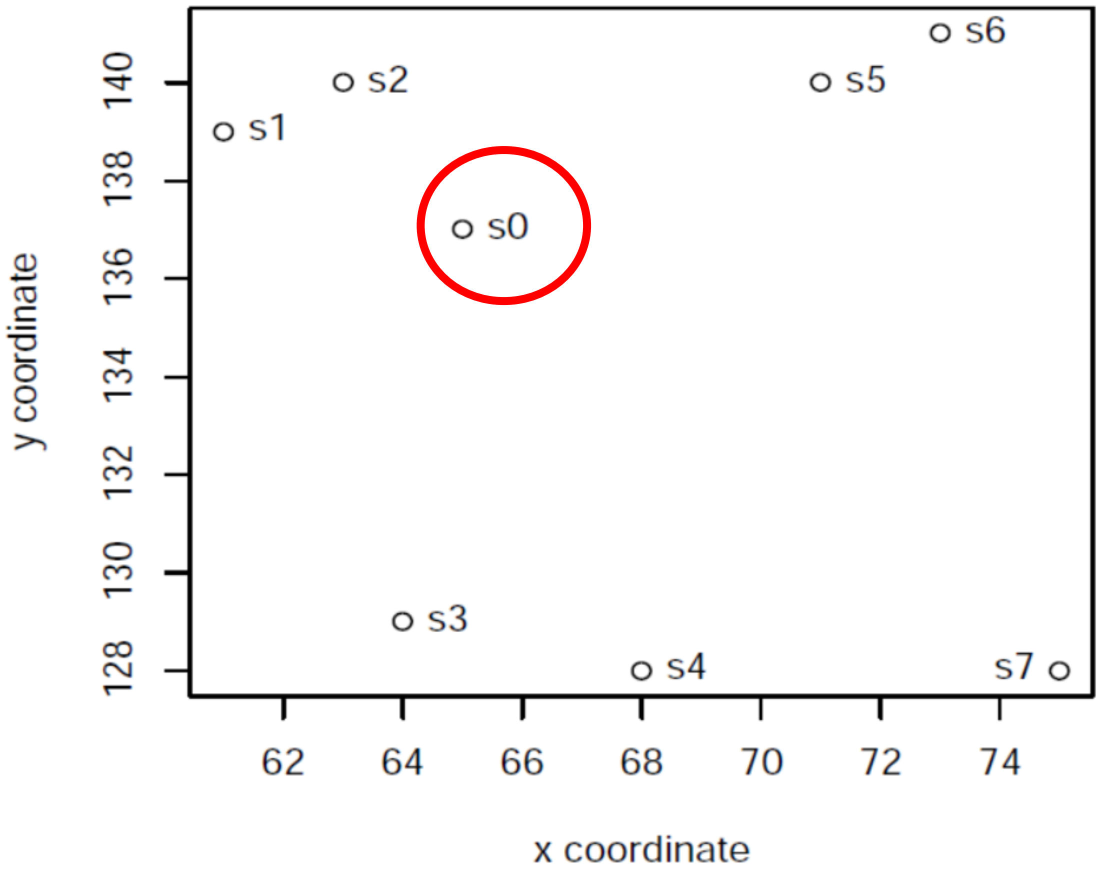
$$\sigma_{OK}^2 = \sum w_i \gamma_{i0} - \lambda = w' D,$$

Ordinary Kriging Estimate and Standard Deviation

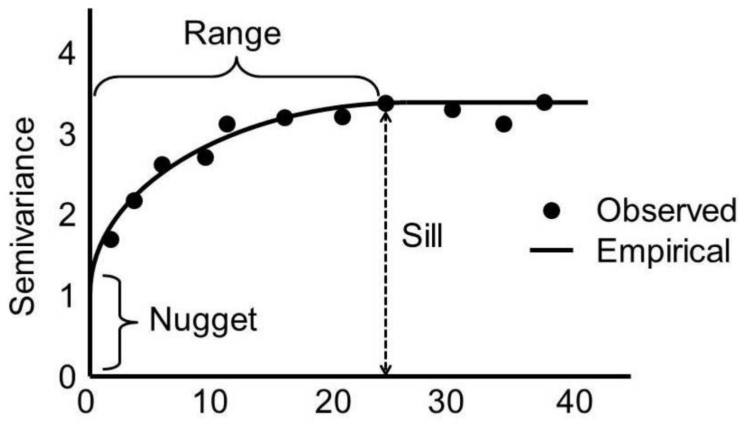


Ordinary Kriging 計算實例

s_i	x	y	$z(s_i)$
s_1	61	139	477
s_2	63	140	696
s_3	64	129	227
s_4	68	128	646
s_5	71	140	606
s_6	73	141	791
s_7	75	128	783
$\mathbf{s_0}$	**65**	**137**	**???**



計算實例 Exponential semivariogram model with parameters $c_0 = 0, c_1 = 10, \alpha = 3.33.$



$$\gamma(h) = 10(1 - e^{-\frac{h}{3.33}}).$$

計算實例 Distance Matrix

$$\underset{(n+1)\times(n+1)}{\Gamma} \cdot \underset{(n+1)\times 1}{w} = \underset{(n+1)\times 1}{D}$$

$$\begin{bmatrix} 0 & \gamma_{12} & \cdots & \gamma_{1n} & 1 \\ \gamma_{21} & 0 & \cdots & \gamma_{2n} & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \gamma_{n1} & \gamma_{n2} & \cdots & 0 & 1 \\ 1 & 1 & \cdots & 1 & 0 \end{bmatrix} \bullet \begin{bmatrix} w_1 \\ w_2 \\ \cdots \\ w_n \\ -\lambda \end{bmatrix} = \begin{bmatrix} \gamma_{10} \\ \gamma_{20} \\ \cdots \\ \gamma_{n0} \\ 1 \end{bmatrix}$$

$(n+1)\times(n+1)$ $\quad$ $(n+1)\times 1$ $\quad$ $(n+1)\times 1$

$$\Rightarrow \quad w = \Gamma^{-1} D$$

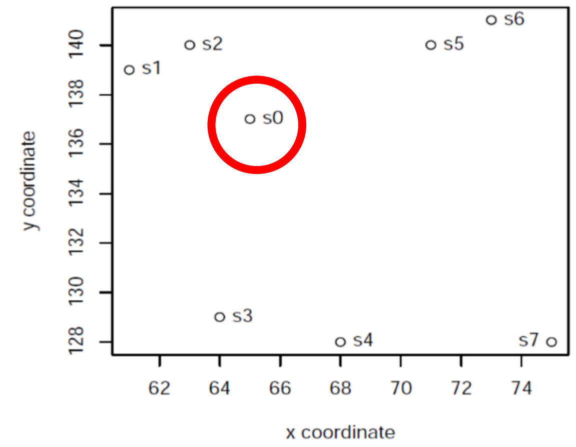
$$\gamma(h) = 10(1 - e^{-\frac{h}{3.33}}).$$

Distance matrix =

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7
s_0	0.00	4.47	3.61	8.06	9.49	6.71	8.94	13.45
s_1	4.47	0.00	2.24	10.44	13.04	10.05	12.17	17.80
s_2	3.61	2.24	0.00	11.05	13.00	8.00	10.05	16.97
s_3	8.06	10.44	11.05	0.00	4.12	13.04	15.00	11.05
s_4	9.49	13.04	13.00	4.12	0.00	12.37	13.93	7.00
s_5	6.71	10.05	8.00	13.04	12.37	0.00	2.24	12.65
s_6	8.94	12.17	10.05	15.00	13.93	2.24	0.00	13.15
s_7	13.45	17.80	16.90	11.05	7.00	2.65	13.15	0.00

計算實例 Exponential semivariogram

$$\gamma(h) = 10(1 - e^{-\frac{h}{3.33}}).$$



$$\Gamma = \begin{pmatrix} 0 & 4.893 & 9.564 & 9.800 & 9.510 & 9.740 & 9.952 & 1 \\ 4.893 & 0 & 9.637 & 9.798 & 9.093 & 9.510 & 9.938 & 1 \\ 9.564 & 9.637 & 0 & 7.095 & 9.800 & 9.889 & 9.637 & 1 \\ 9.800 & 9.798 & 7.095 & 0 & 9.755 & 9.847 & 8.775 & 1 \\ 9.510 & 9.093 & 9.800 & 9.755 & 0 & 4.893 & 9.775 & 1 \\ 9.740 & 9.510 & 9.889 & 9.847 & 4.893 & 0 & 9.806 & 1 \\ 9.952 & 9.938 & 9.637 & 8.775 & 9.775 & 9.806 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix} \qquad \gamma = \begin{pmatrix} 7.384 \\ 6.614 \\ 9.109 \\ 9.420 \\ 8.664 \\ 9.316 \\ 9.823 \\ 1 \end{pmatrix}$$

計算實例 Kriging Weights

$$\boldsymbol{W} = \boldsymbol{\Gamma}^{-1}\gamma = \begin{pmatrix} 0 & 4.893 & 9.564 & 9.800 & 9.510 & 9.740 & 9.952 & 1 \\ 4.893 & 0 & 9.637 & 9.798 & 9.093 & 9.510 & 9.938 & 1 \\ 9.564 & 9.637 & 0 & 7.095 & 9.800 & 9.889 & 9.637 & 1 \\ 9.800 & 9.798 & 7.095 & 0 & 9.755 & 9.847 & 8.775 & 1 \\ 9.510 & 9.093 & 9.800 & 9.755 & 0 & 4.893 & 9.775 & 1 \\ 9.740 & 9.510 & 9.889 & 9.847 & 4.893 & 0 & 9.806 & 1 \\ 9.952 & 9.938 & 9.637 & 8.775 & 9.775 & 9.806 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}^{-1} \begin{pmatrix} 7.384 \\ 6.614 \\ 9.109 \\ 9.420 \\ 8.664 \\ 9.316 \\ 9.823 \\ 1 \end{pmatrix}$$

The answer is:

$$\boldsymbol{W} = \begin{pmatrix} 0.174 \\ 0.317 \\ 0.129 \\ 0.086 \\ 0.151 \\ 0.057 \\ 0.086 \\ 0.906 \end{pmatrix}.$$

Lagrange multiplier

計算實例 Kriging Estimate and Variance

The predicted value at location s_0 is equal to:

$$\hat{z}(s_0) = \sum_{i=1}^{n} w_i z(s_i) = 0.174(477) + \cdots + 0.086(783) = 592.59.$$

And the variance:

$$\sigma_e^2 = \sum_{i=1}^{n} w_i \gamma(s_i - s_0) + \lambda = 0.174(7.384) + \cdots + 0.086(9.823) + 0.906 = 8.96.$$

95% confidence Interval

$$592.59 \pm 1.96\sqrt{8.96}$$

Or

$$577.09 \leq Z(s_0) \leq 588.83$$

R Lab: Ordinary Kriging

krige {gstat} R Documentation

Simple, Ordinary or Universal, global or local, Point or Block Kriging, or simulation.

Description

Function for simple, ordinary or universal kriging (sometimes called external drift kriging), kriging in a local neighbourhood, point kriging or kriging of block mean values (rectangular or irregular blocks), and conditional (Gaussian or indicator) simulation equivalents for all kriging varieties, and function for inverse distance weighted interpolation. For multivariable prediction, see gstat and predict

```
pm.kriged = krige(PM~1, EPA_STN, grid, model = pm.fit)
```

R Lab: Ordinary Kriging

台灣大學的位置 台北市羅斯福路四段1號

X = 304023

Y = 2767886

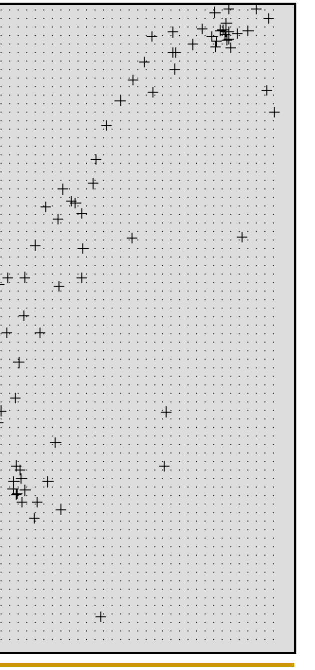
```
ntu_pts <-SpatialPoints(cbind(304023,2767886), proj4string = CRS(proj4string(EPA_STN)))
krige(PM~1, EPA_STN, ntu_pts, model = pm.fit)
```

```
> krige(PM~1, EPA_STN, ntu_pts, model = pm.fit)
[using ordinary kriging]
        coordinates var1.pred var1.var
1 (304023, 2767886)  81.61952 215.2997
```

R Lab: Ordinary Kriging

建立涵蓋台灣範圍的估計點。 makegrid()

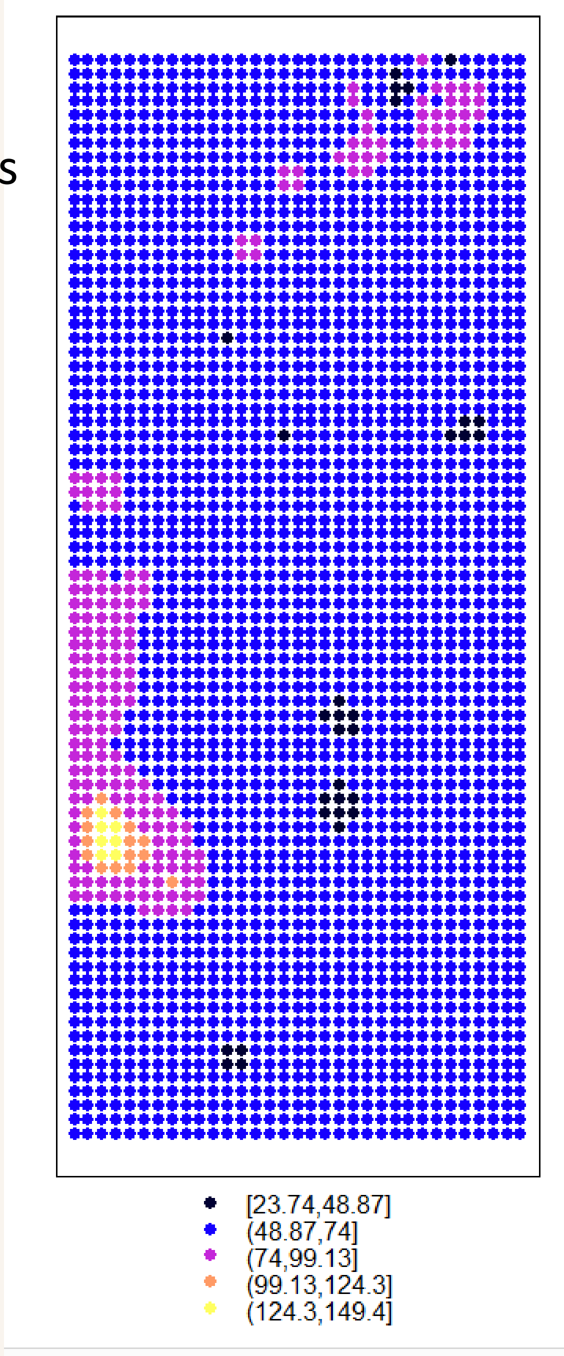
```
# generating grid
proj4string(EPA_STN)
grid <- makegrid(EPA_STN, cellsize = 5000)
grid <- SpatialPoints(grid, proj4string = CRS(proj4string(EPA_STN)))

plot(EPA_STN)
plot(grid, pch = ".", add = T)
```



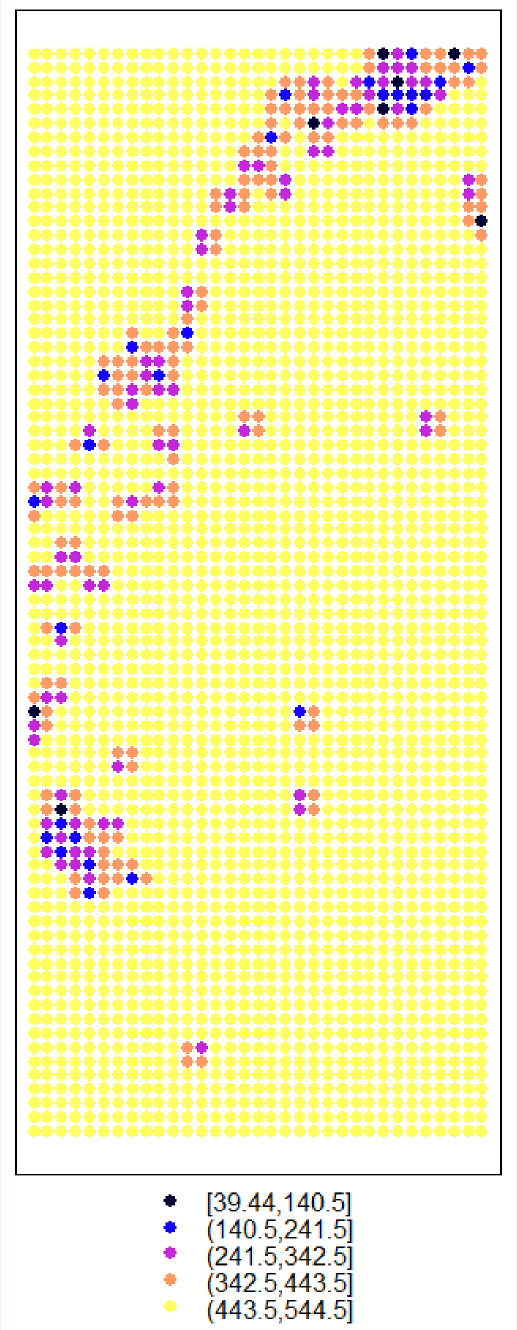
R code: Ordinary Kriging

```
pm.kriged = krige(PM~1, EPA_STN, grid, model = pm.fit)


splot(pm.kriged["var1.pred"])
spplot(pm.kriged["var1.var"])
```

Predicted PM values



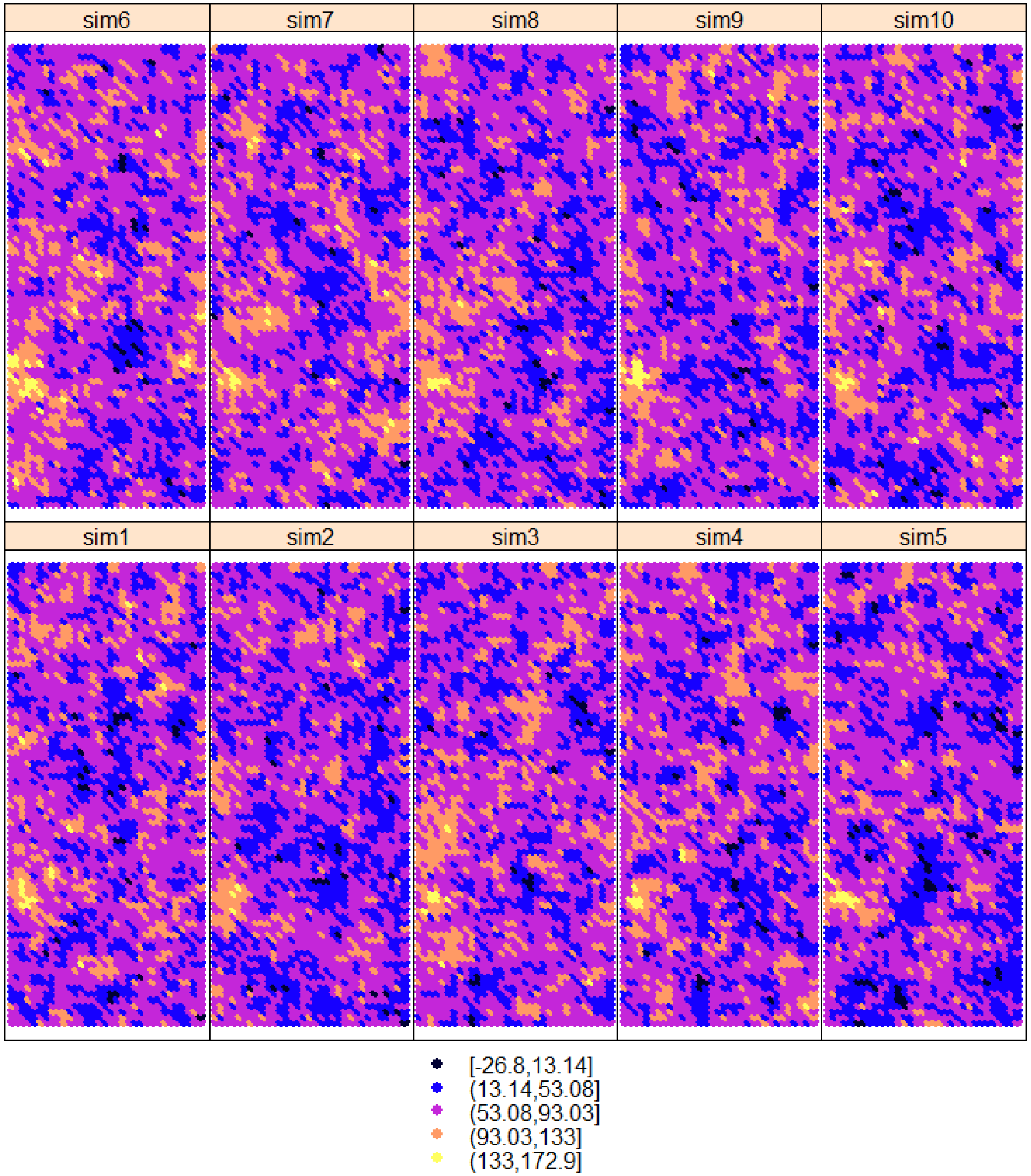
Variance



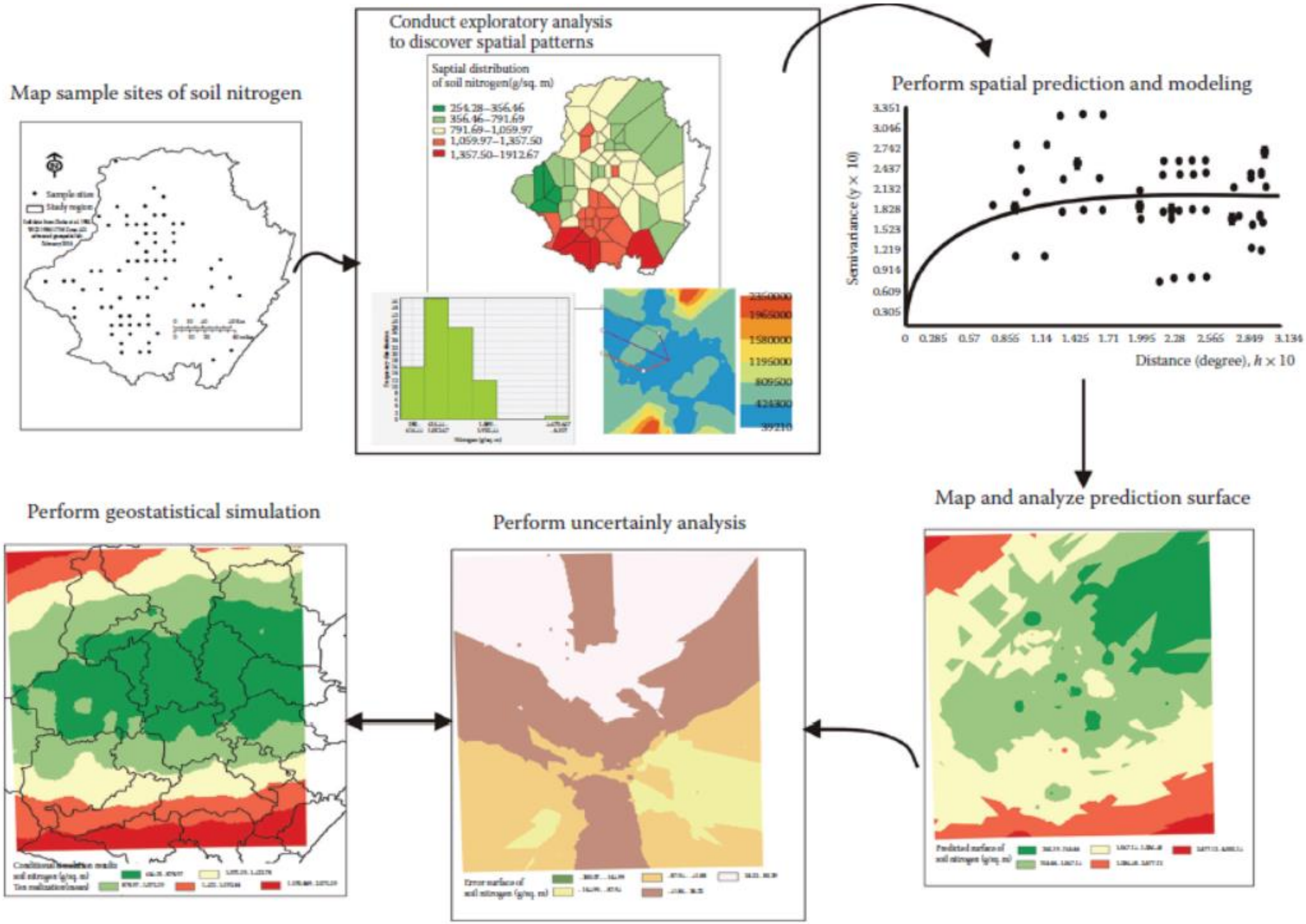
Conditional Simulation

- Spatial mean estimated by kriging.
- Uncertain/stochastic aspect simulated.
- Offers a realistic representation of the variable.
- Gaussian simulations - simulated portion follows a normal distribution.

```
pm.condsim = krige(PM~1, EPA_STN, grid, model = pm.fit, nmax = 30, nsim = 10)
spplot(pm.condsim)
```

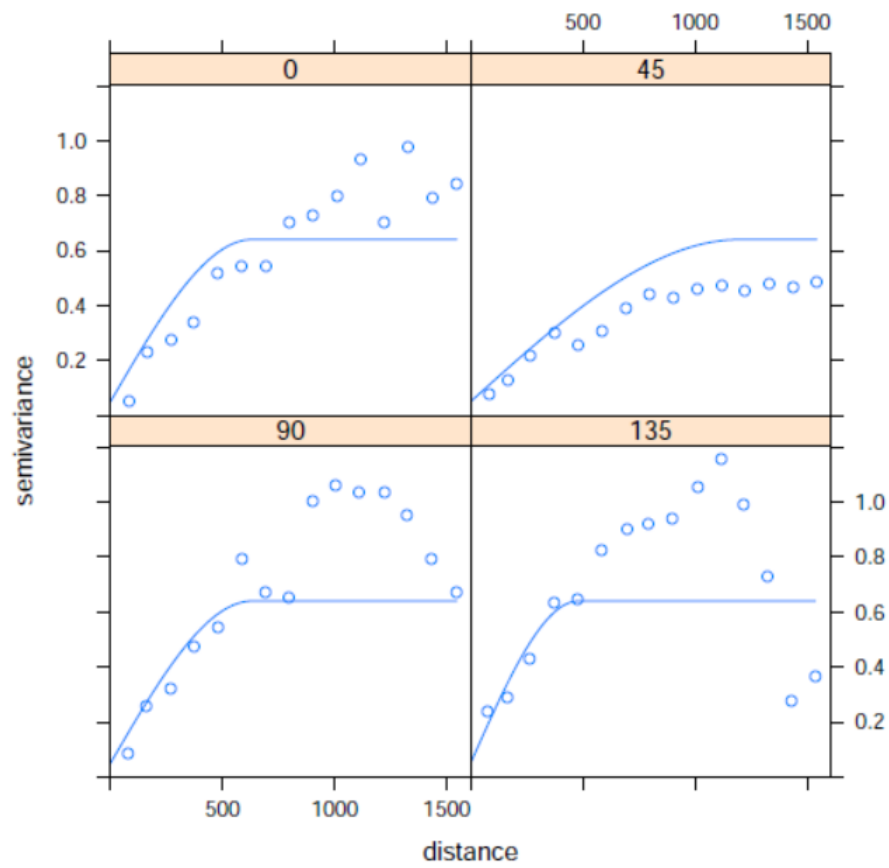

sim6
sim7
sim8
sim9
sim10
sim1
sim2
sim3
sim4
sim5
[-26.8,13.14]
(13.14,53.08]
(53.08,93.03]
(93.03,133]
(133,172.9]

Workflow of Geostatistical Estimation



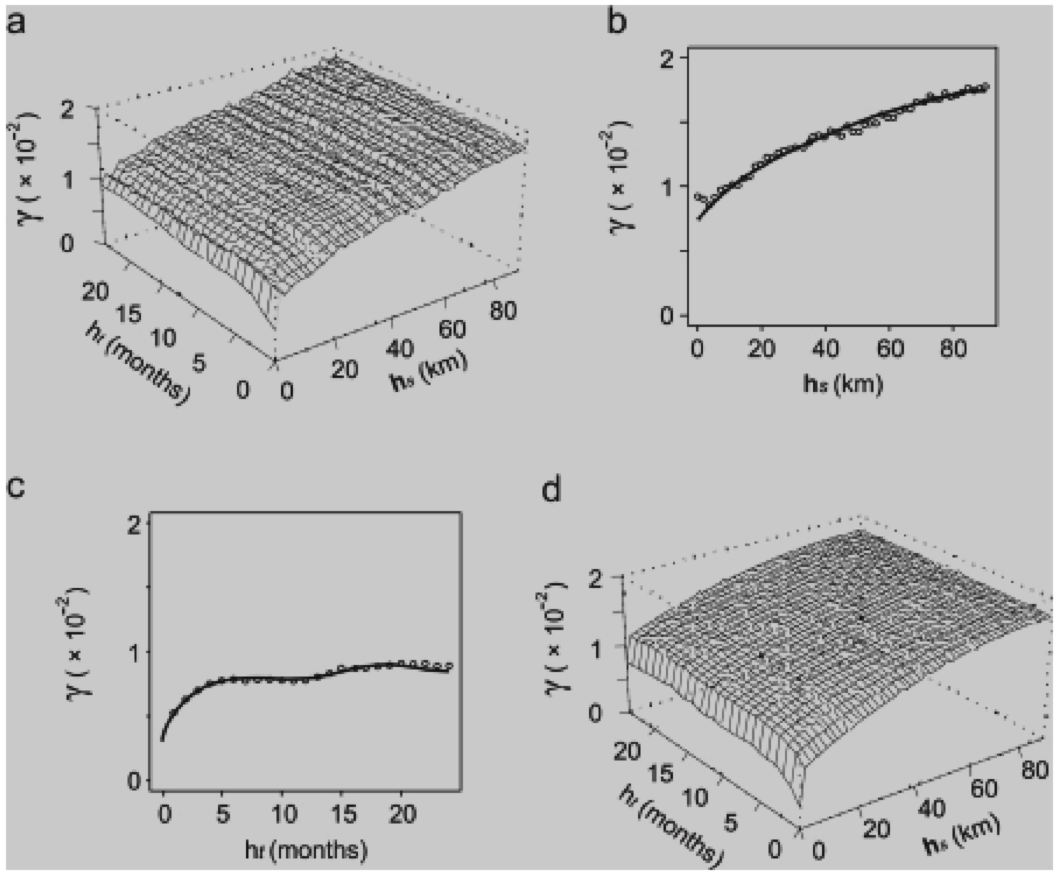
Further Issues

- Directional variogram (e.g. terrain effect)



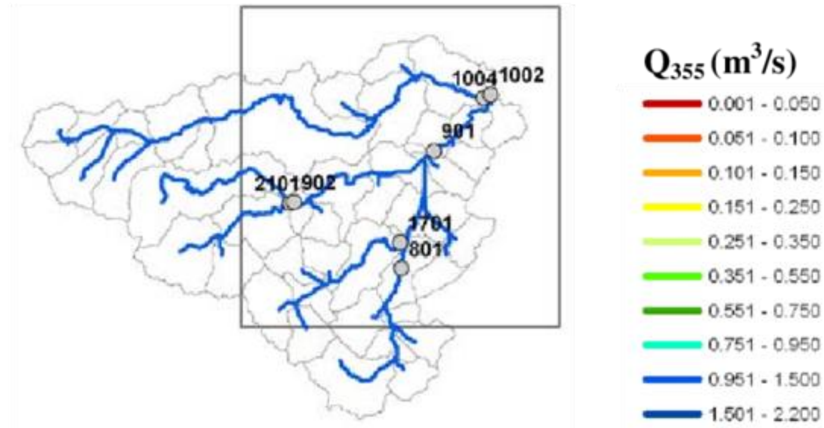
Further Issues

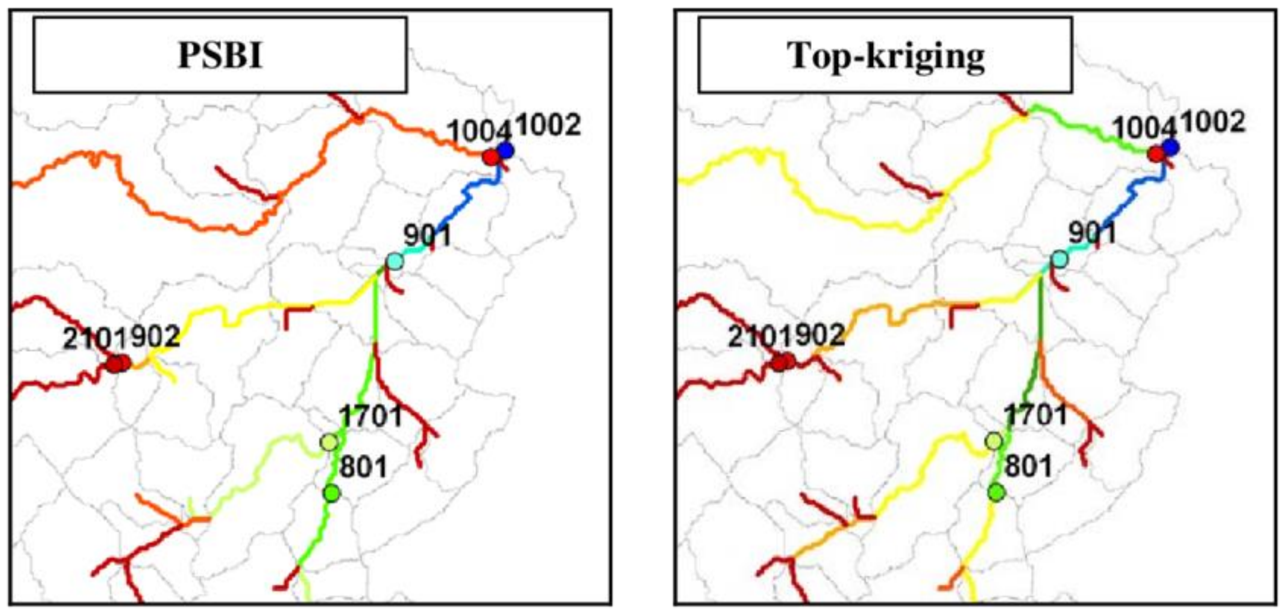
- Space-time variogram



Further Issues

- Network structure (e.g. river, road,…)

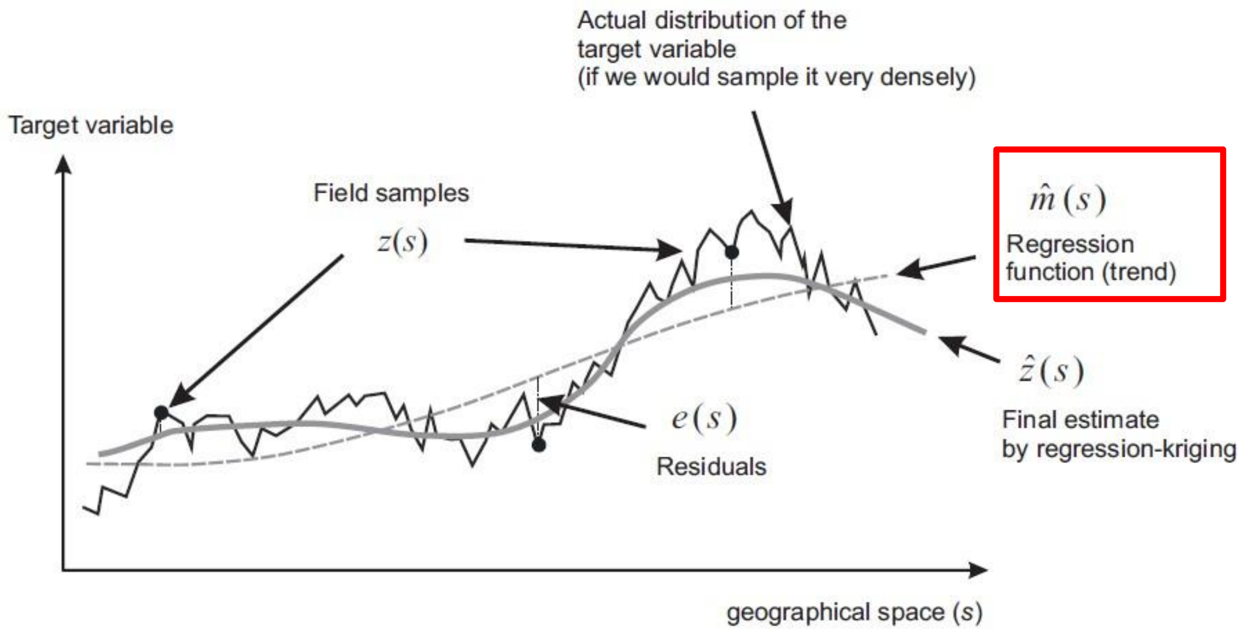




Further Issues

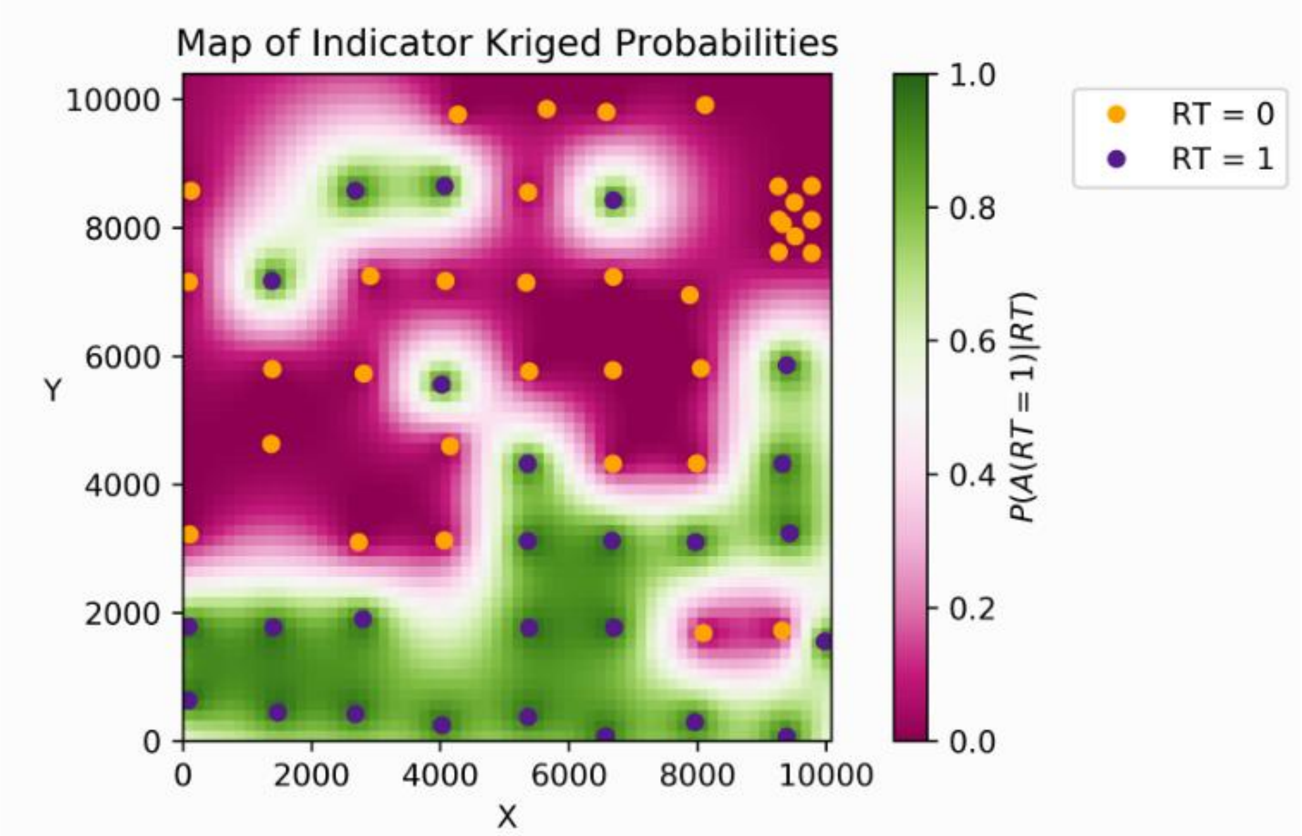
- Regression- Kriging (universal Kriging)

$$\hat{z}(\mathbf{s}_0) = \hat{m}(\mathbf{s}_0) + \hat{e}(\mathbf{s}_0) = \sum_{k=0}^{p} \hat{\beta}_k \cdot q_k(\mathbf{s}_0) + \sum_{i=1}^{n} \lambda_i \cdot e(\mathbf{s}_i)$$



Further Issues

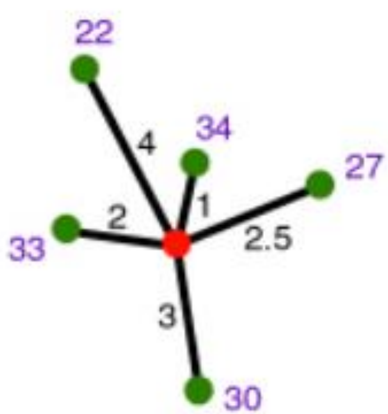
- Binary outcome variable: Indicator Kriging



Deterministic: Inverse Distance Weighting

$$\hat{v}_1 = \frac{\sum_{i=1}^{n} \frac{1}{d^{p}_{i}} v_i}{\sum_{i=1}^{n} \frac{1}{d^{p}_{i}}}$$

Inverse distance weighting (IDW) is a deterministic, nonlinear interpolation technique that uses a weighted average of the attribute (i.e., phenomenon) values from nearby sample points to estimate the magnitude of that attribute at non-sampled locations.



$$Z(x) = \frac{\Sigma w_i z_i}{\Sigma w_i} = \frac{\frac{34}{1^2} + \frac{33}{2^2} + \frac{27}{2.5^2} + \frac{30}{3^2} + \frac{22}{4^2}}{\frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{2.5^2} + \frac{1}{3^2} + \frac{1}{4^2}} = \mathbf{32.38}$$

R code: Inverse Distance Weighting

```r
pm.idw = idw(PM~1, EPA_STN, grid, idp=1)
spplot(pm.idw)
```