Statistical modelling of spatial data

Andreas Papritz

ETH Zurich

Institute of Terrestrial Ecosystems

Physics of Soils and Terrestrial Ecosystems

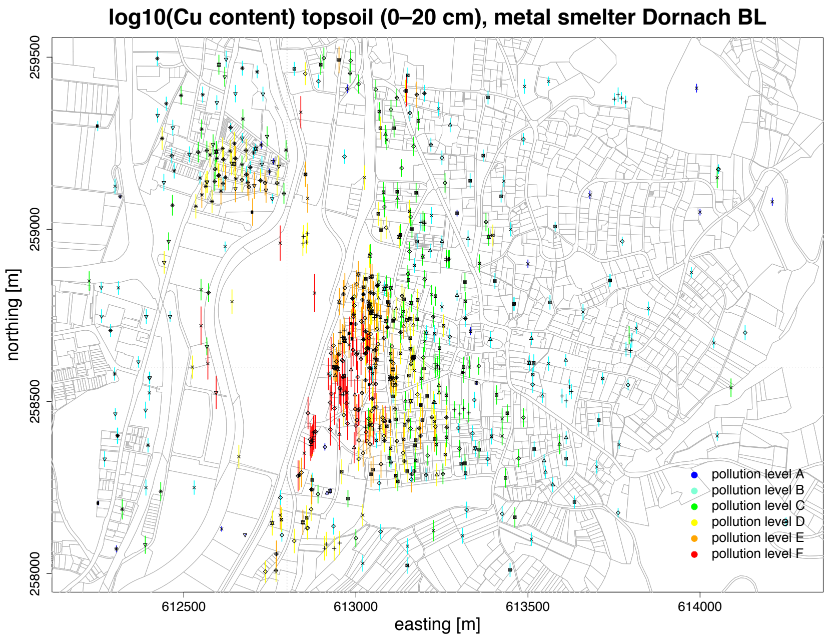
andreas.papritz@env.ethz.ch

Contents

1 geostatistics: data sets and objectives of analyses

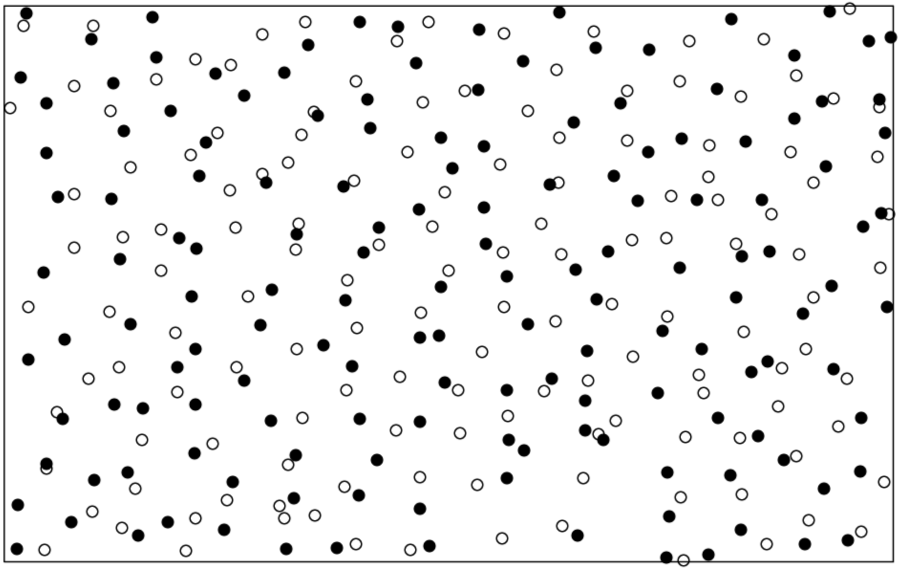
- spatial statistics: statistical analysis of data for which spatial (or spatio-temporal) position where attribute was recorded is known
- depending on type of spatial data one distinguishes:
  - *geostatistics*: analysis of spatial data that refers to a very small area (volume) and that can in principle be recorded at any point in a study domain ($\Rightarrow$ infinite number of locations in study domain where measurements can be taken)
  - *point pattern analysis*: analysis of spatial positions of “points” (or other objects) in a study domain
  - *statistical image analysis*: analysis of spatial data that typically refer to larger areas (volumes) and that are arranged on regular or irregular lattices (with finite number of “cells”)

$\Rightarrow$ geostatistics is just one (important) branch of spatial statistics

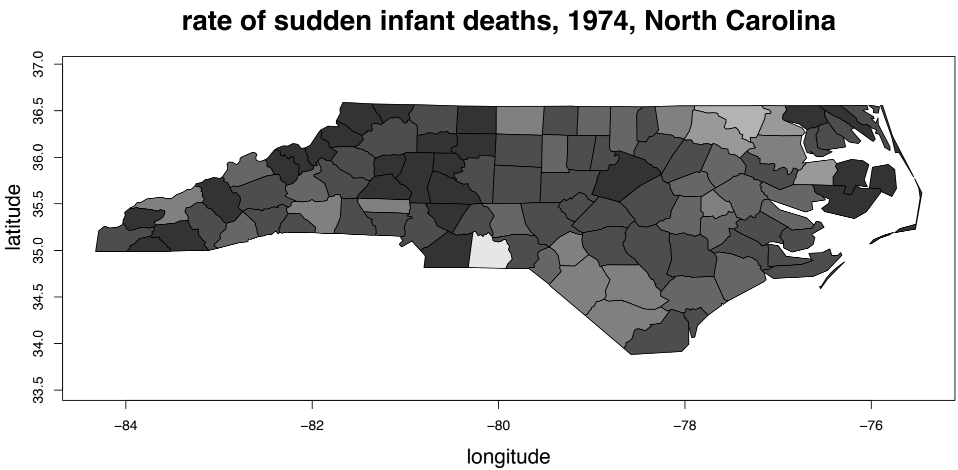


log10(Cu content) topsoil (0–20 cm), metal smelter Dornach BL
northing [m]
258000
258500
259000
259500
easting [m]
612500
613000
613500
614000
pollution level A
pollution level B
pollution level C
pollution level D
pollution level E
pollution level F

a point pattern



a lattice data set



1.2 objectives of geostatistical analyses

1. computing spatial predictions
   - pressure head of Wolfcamp aquifer
   - topsoil heavy metal content in vicinity of metal smelter
2. parameter estimation, significance testing for analysis of experimental data
   - field trial on yield of 56 wheat varieties

1.2.1 Wolfcamp aquifer data set

- data on hydraulic head (pressure) in a confined brine aquifer in NW Texas
- hydro-geological study part of evaluation whether region suited to host nuclear waste repository
- measurement of hydraulic head at 85 locations
- geostatistical analysis by Cressie (1993)
- data set part of R packages `georob` and `geoR`

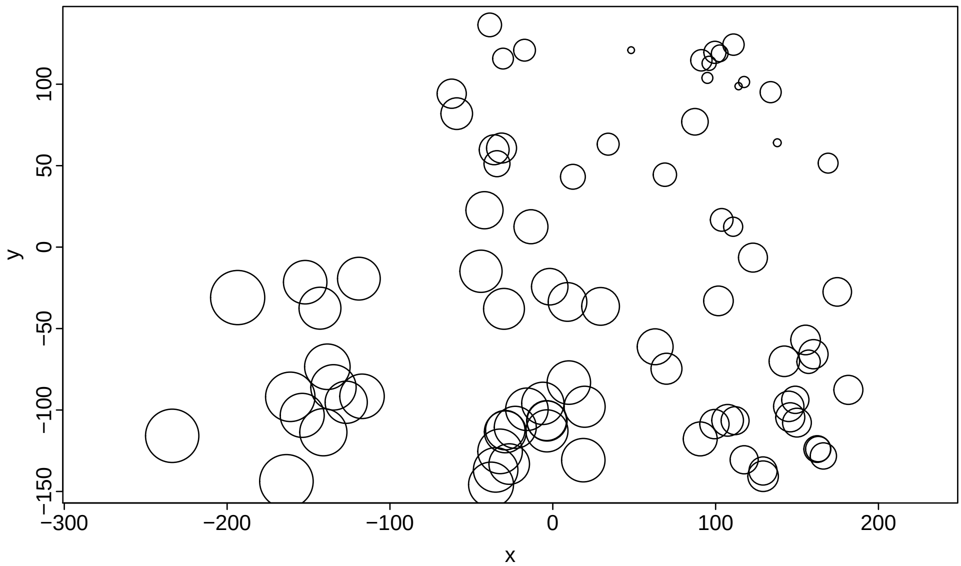
```
> library(georob)
> data(wolfcamp, package="georob")
> str(wolfcamp)
```

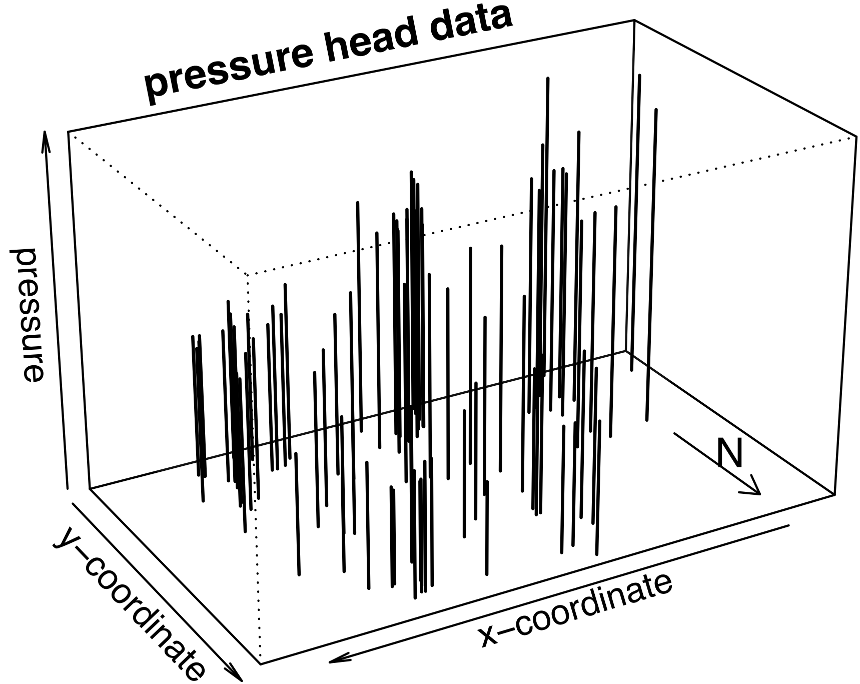
```
'data.frame':        85 obs. of  3 variables:
 $ x       : num  68.85 -44.09 -1.87 -29.96 155.24 ...
 $ y       : num  44.5 -14.8 -24.3 -37.9 -57 ...
 $ pressure: num  446 778 658 748 535 ...
```

```
> summary(wolfcamp)
```

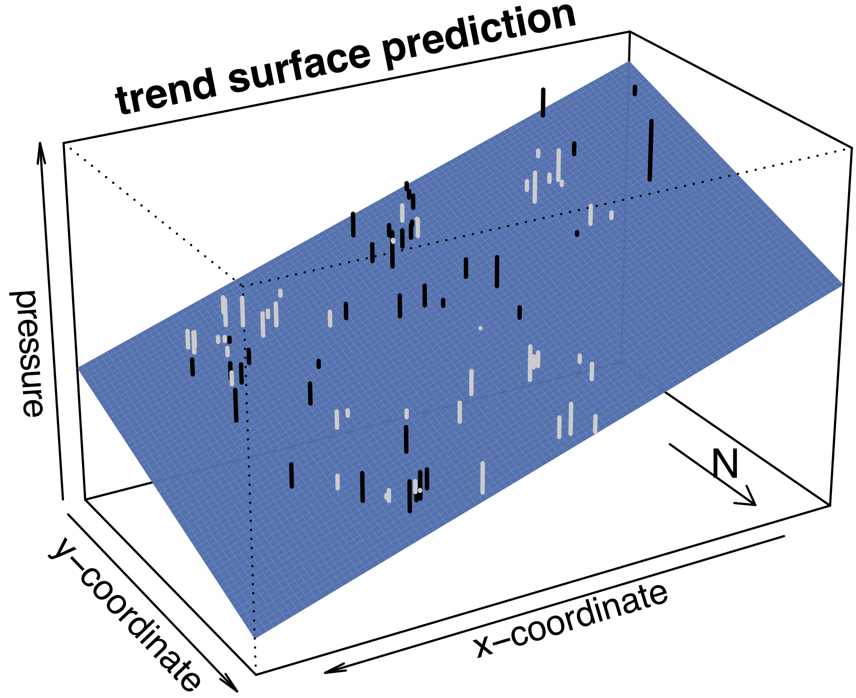
```
       x                  y               pressure
 Min.   :-233.72   Min.   :-145.79   Min.   : 312.1
 1st Qu.: -34.26   1st Qu.:-106.73   1st Qu.: 471.8
 Median :  19.57   Median : -65.74   Median : 547.7
 Mean   :  27.63   Mean   : -33.23   Mean   : 610.3
 3rd Qu.: 114.10   3rd Qu.:  51.21   3rd Qu.: 774.2
 Max.   : 181.53   Max.   : 136.41   Max.   :1088.4
```

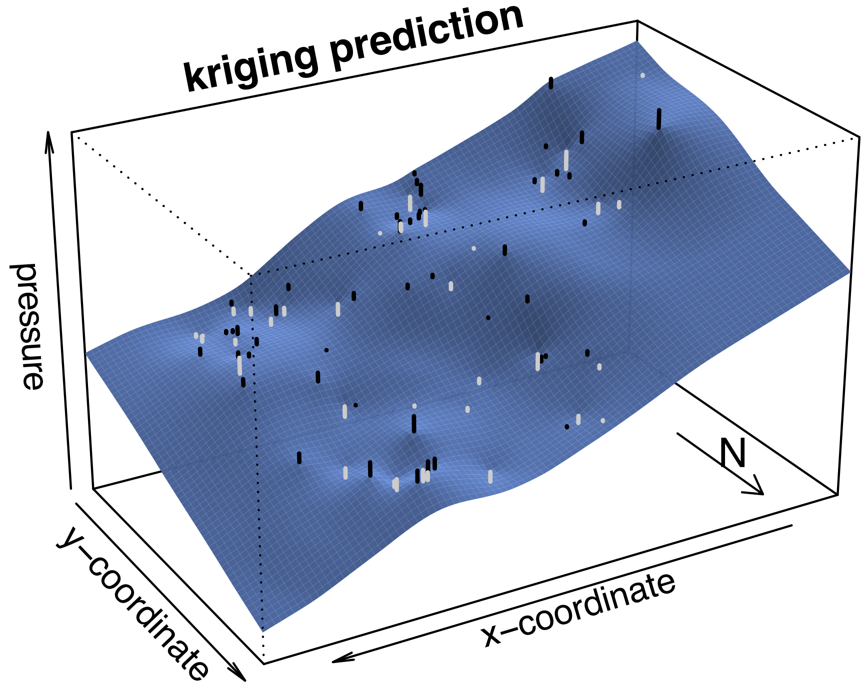
```
> ## bubble plot: symbol area linearly related to data
> plot(y~x, data=wolfcamp, asp=1,
+   cex=sqrt(pressure-300)/5)
```

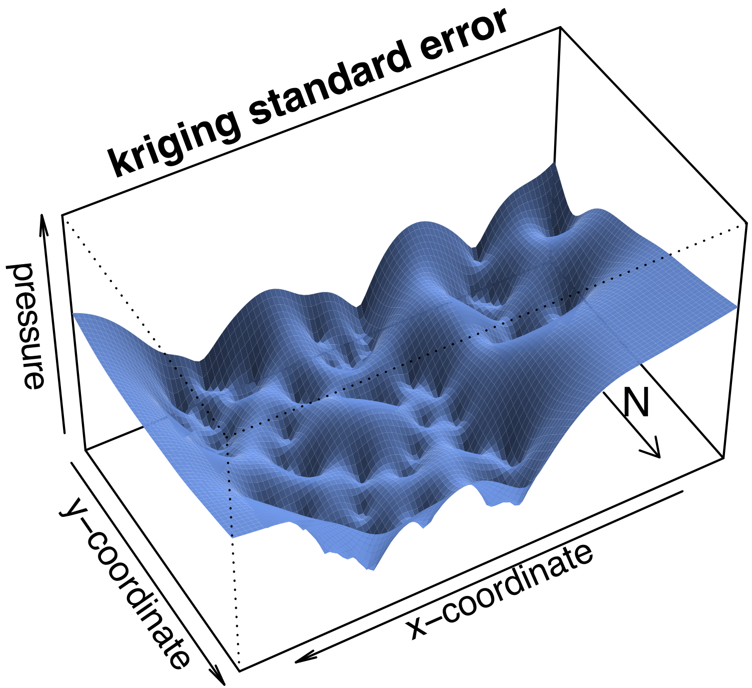




pressure head data
pressure
y–coordinate
x–coordinate
N


trend surface prediction
pressure
y–coordinate
x–coordinate
N


kriging prediction
pressure
y–coordinate
x–coordinate
N


kriging standard error
pressure
y–coordinate
x–coordinate
N

- set of observations $(y_i, \boldsymbol{x}_i)$ where y_i is a datum of a *response* variable and $\boldsymbol{x}_i$ is a *spatial location* in a study domain D

- optional: spatial covariates, say $d_k(\boldsymbol{x}_i)$, used to “explain” the spatial pattern of the response variable

- geostatistical data often show (gradual) *large-scale spatial variation (trend)* and *small-scale local fluctuations*

  - *trend*: commonly modelled as low-order polynomial function of spatial coordinates ($\Rightarrow$ *trend surface*) or as function of external spatial covariates ($\Rightarrow$ *external trend*)
  - local fluctuations usually spatially structured (values at pairs of nearby locations “more similar” than for pairs farther apart): $\Rightarrow$ *auto-correlation*

- spatial data sometimes distorted by independent *measurement errors*

terminology and model notation

- model for data: $Y_i = S(\boldsymbol{x}_i) + Z_i$ where

Y_i	i^{th} datum
$S(\boldsymbol{x}_i)$	"signal" (= true quantity) at location $\boldsymbol{x}_i$
Z_i	iid random measurement error

- decomposition of signal into trend $\mu(\boldsymbol{x}_i)$ and stochastic fluctuation:

$$S(\boldsymbol{x}_i) = \mu(\boldsymbol{x}_i) + E(\boldsymbol{x}_i)$$

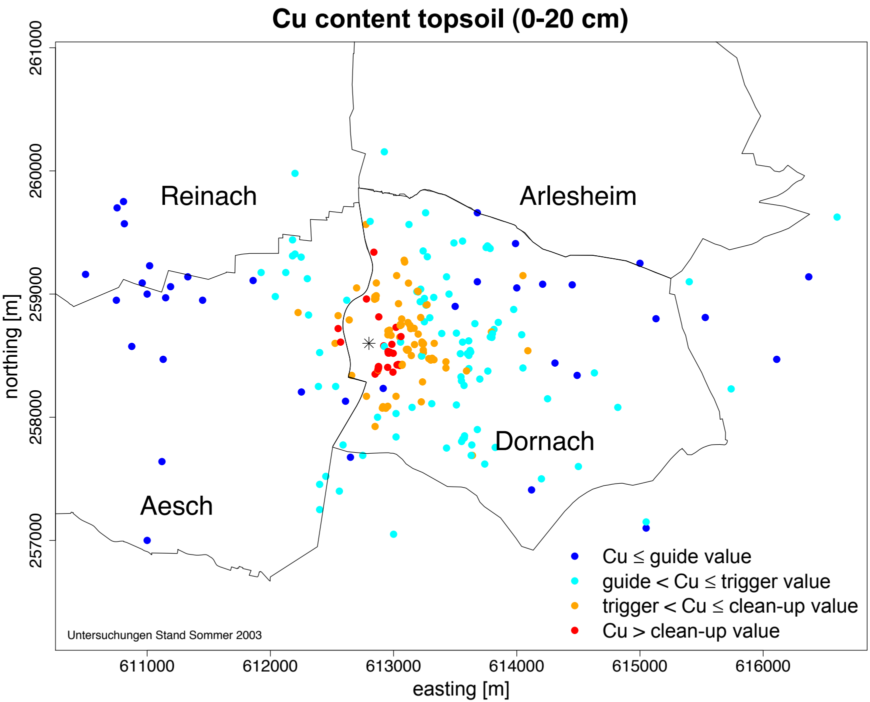
where commonly a linear model is used for $\mu(\boldsymbol{x}_i)$

$$\mu(\boldsymbol{x}_i) = \sum_k d_k(\boldsymbol{x}_i)\beta_k = \boldsymbol{d}(\boldsymbol{x}_i)^{\mathsf{T}}\boldsymbol{\beta}$$

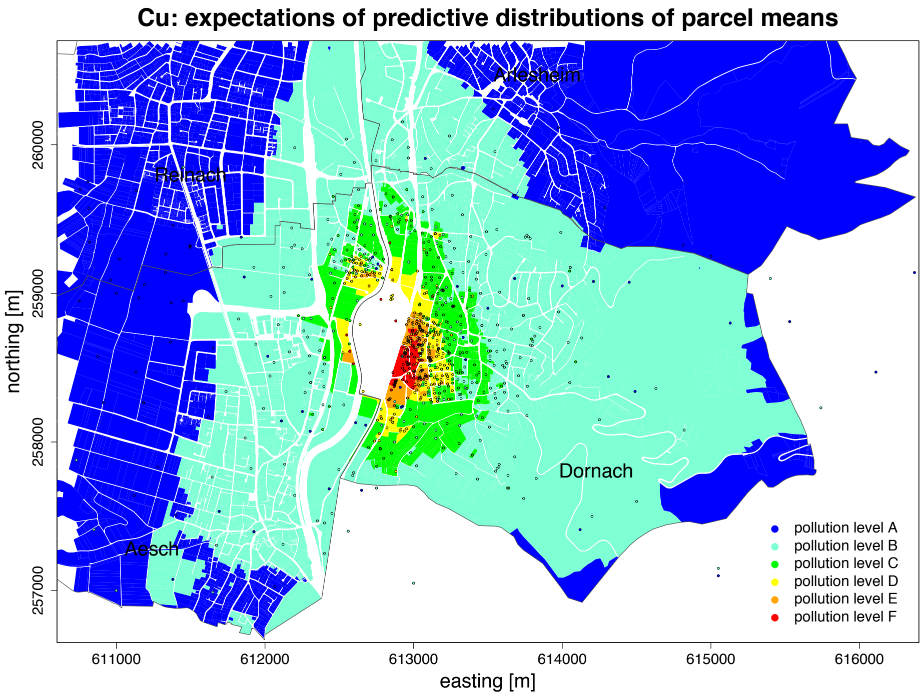
with $d_k(\boldsymbol{x}_i)$ denoting (spatial) covariates and $\{E(\boldsymbol{x}_i)\}$ a zero mean stochastic process (random field)

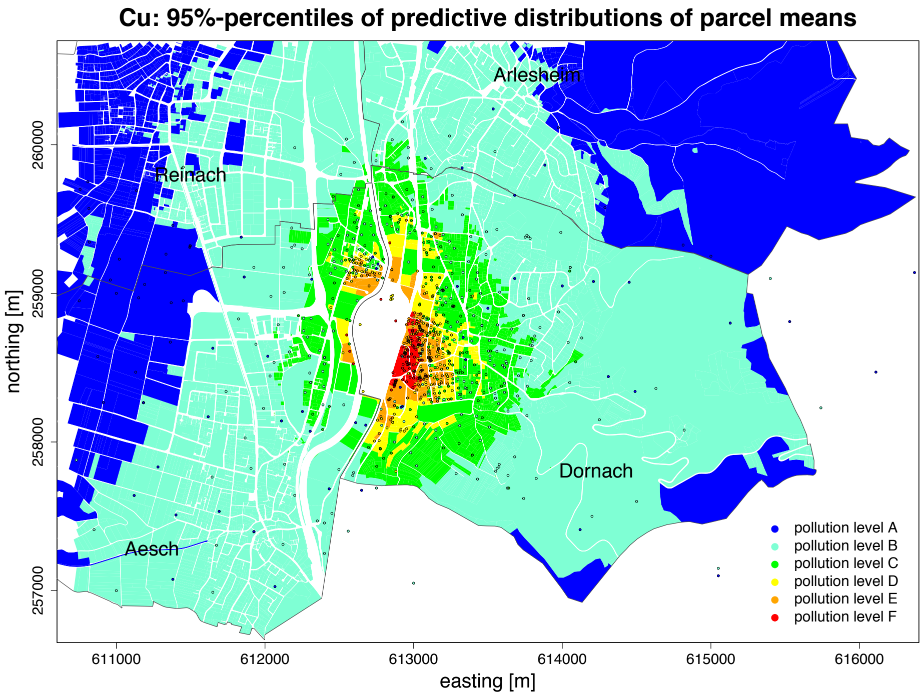
1.2.2 Dornach data: soil pollution by heavy metals

- site: Swissmetal smelter in Dornach (SO), Switzerland
- emission of dust and fumes containing Cu, Cd and Zn from 1870 to 1980s severely contaminated soils in vicinity of smelter
- comprehensive survey 2003–2005 with objective to spatially delimit zones that require mitigating measures
  details → website BBG Dornach
- more details in Hofer *et al*. (2013)



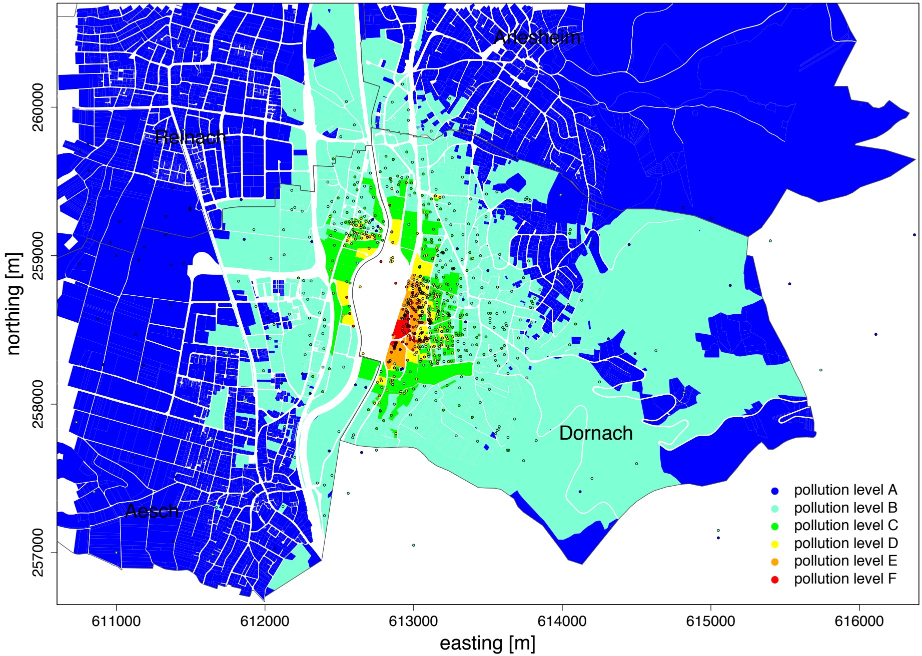
Cu content topsoil (0-20 cm)
261000
260000
259000
258000
257000
northing [m]
611000
612000
613000
614000
615000
616000
easting [m]
Reinach
Arlesheim
Dornach
Aesch
Cu ≤ guide value
guide < Cu ≤ trigger value
trigger < Cu ≤ clean-up value
Cu > clean-up value
Untersuchungen Stand Sommer 2003


Cu: expectations of predictive distributions of parcel means
northing [m]
easting [m]
260000
259000
258000
257000
611000
612000
613000
614000
615000
616000
Arlesheim
Reinach
Dornach
Aesch
pollution level A
pollution level B
pollution level C
pollution level D
pollution level E
pollution level F



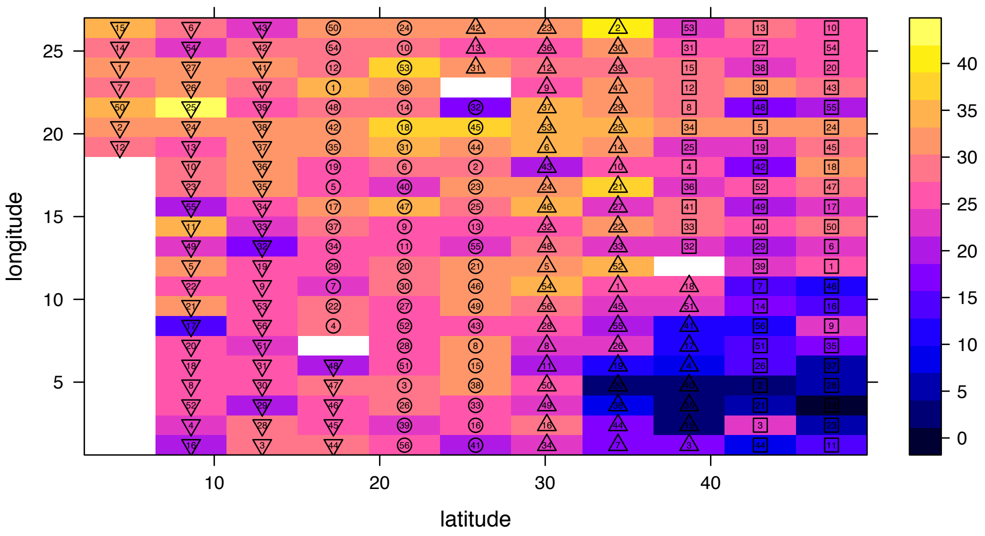
Cu: 95%-percentiles of predictive distributions of parcel means
Arlesheim
Reinach
Dornach
Aesch
pollution level A
pollution level B
pollution level C
pollution level D
pollution level E
pollution level F
northing [m]
260000
259000
258000
257000
easting [m]
611000
612000
613000
614000
615000
616000

Cu: 5%-percentiles of predictive distributions of parcel means



1.2.3 field experiment on yield of wheat varieties

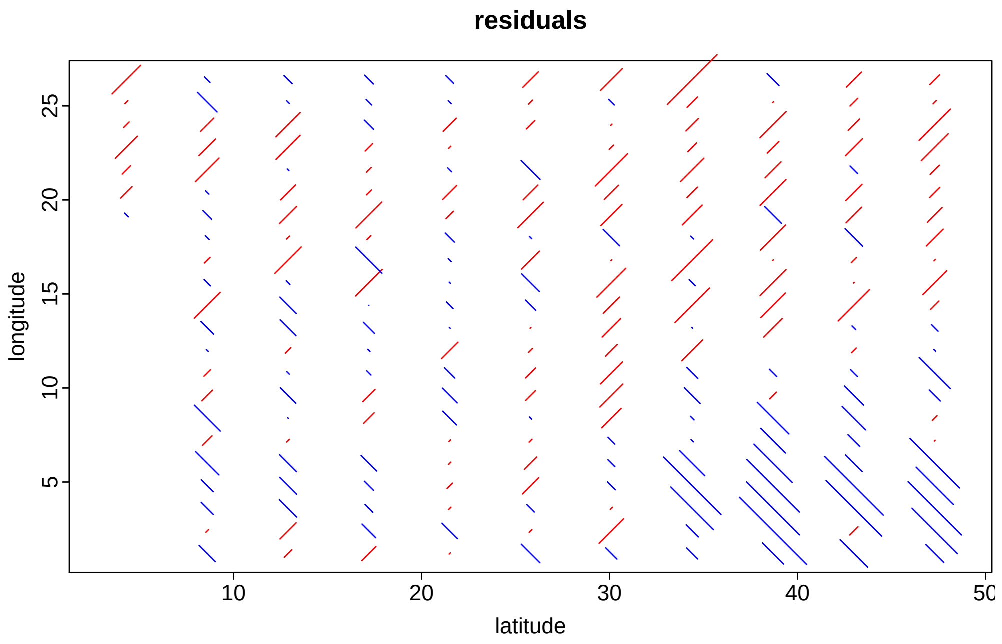
- field experiment to compare the yield of 56 varieties of wheat
- design of experiment: completely randomised block experiment with 4 blocks
- location of centres of each of 224 experimental plots were recorded
- more details in Pinheiro and Bates (2000, pp. 260)

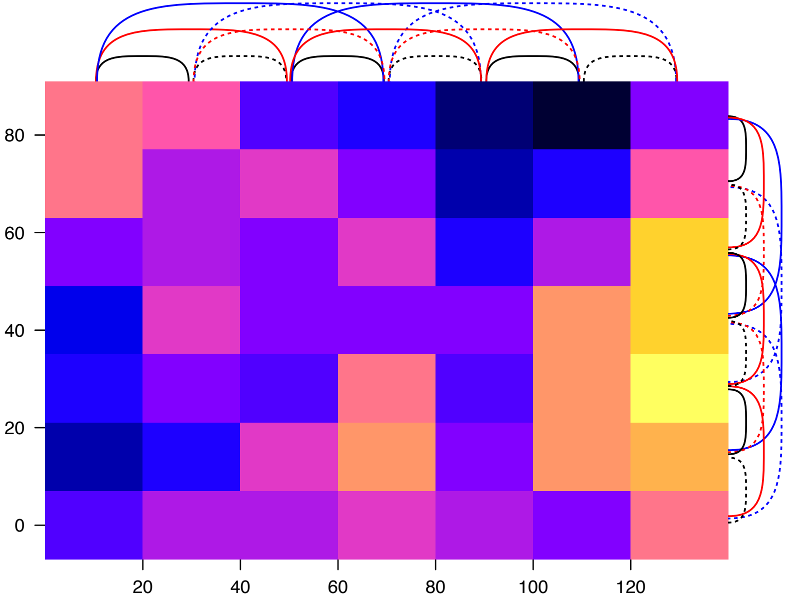

longitude
latitude

```
> library(nlme)
> # analysis as a completely randomised block experiment
> r.lme.means <- lme(yield~variety-1, Wheat2,
+   random=~1|Block)
> summary(r.lme.means)
```

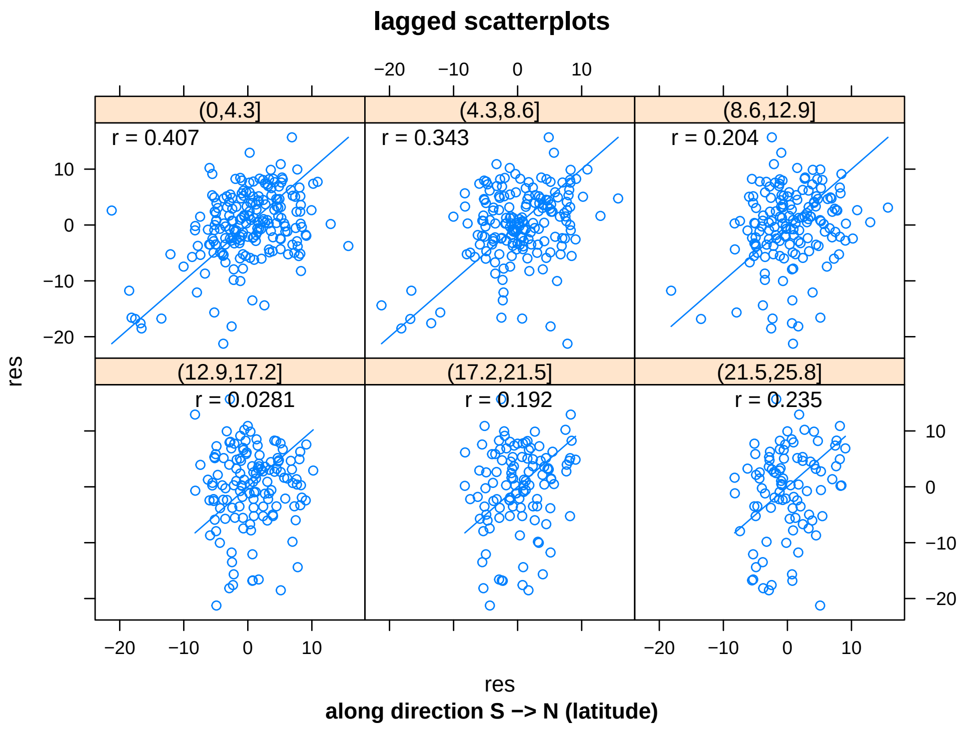
```
Linear mixed-effects model fit by REML
 Data: Wheat2
       AIC      BIC    logLik
  1333.702 1514.891 -608.8508

Random effects:
 Formula: ~1 | Block
        (Intercept) Residual
StdDev:     3.14371 7.041475

Fixed effects: yield ~ variety - 1
                  Value Std.Error  DF  t-value p-value
varietyARAPAHOE   29.4375  3.855687 165 7.634827       0
varietyBRULE      26.0750  3.855687 165 6.762738       0
varietyBUCKSKIN   25.5625  3.855687 165 6.629818       0
...
```

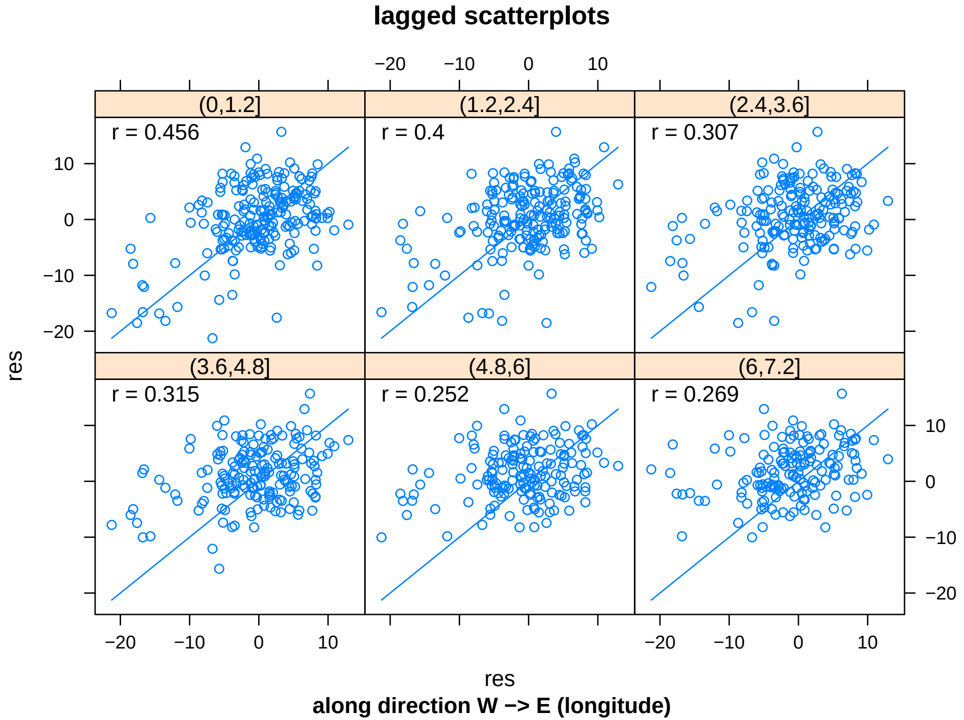
```
> # global F-Test: testing for any significant
> #   treatment effects
> anova(update(r.lme.means, .~. + 1))

            numDF denDF   F-value p-value
(Intercept)     1   165 242.05402  <.0001
variety        55   165   0.87549  0.7119

> # testing particular treatment contrast
> #   (BUCKSKIN vs. ARAPHAHOE)
> anova(r.lme.means, L=c(1, 0, -1))

F-test for linear combination(s)
varietyARAPAHOE varietyBUCKSKIN
              1              -1
  numDF denDF   F-value p-value
1     1   165 0.6056841  0.4375
```


residuals
longitude
latitude
5
10
15
20
25
10
20
30
40
50



80
60
40
20
0
20
40
60
80
100
120


lagged scatterplots
(0,4.3]
(4.3,8.6]
(8.6,12.9]
(12.9,17.2]
(17.2,21.5]
(21.5,25.8]
r = 0.407
r = 0.343
r = 0.204
r = 0.0281
r = 0.192
r = 0.235
res
res
along direction S −> N (latitude)


lagged scatterplots
(0,1.2]
r = 0.456
(1.2,2.4]
r = 0.4
(2.4,3.6]
r = 0.307
(3.6,4.8]
r = 0.315
(4.8,6]
r = 0.252
(6,7.2]
r = 0.269
−20
−10
0
10
res
res
along direction W −> E (longitude)

```
> # analysis using a geostatistical spatial model
> r.gls.means <- gls(yield~variety-1, Wheat2,
+   corr=corRatio(form=~latitude+longitude,
+   nugget=TRUE))
> summary(r.gls.means)
```

```
Generalised least squares fit by REML
  Model: yield ~ variety
  Data: Wheat2
       AIC      BIC    logLik
  1183.278 1367.592 -532.6389

Correlation Structure: Rational quadratic spatial correlation
 Formula: ~latitude + longitude
 Parameter estimate(s):
     range     nugget
13.4613358  0.1935803

Coefficients:
                      Value Std.Error   t-value p-value
varietyARAPAHOE   26.54597  4.970942 5.340229   0e+00
varietyBRULE      26.28374  4.984883 5.272690   0e+00
varietyBUCKSKIN   35.03727  5.007094 6.997526   0e+00
...
```

```
> # global F-Test: testing for any significant
> #   treatment effects
> anova(update(r.gls.means, .~.+1))


Denom. DF: 168
            numDF  F-value p-value
(Intercept)     1 30.39940  <.0001
variety        55  1.85094  0.0015


> # testing particular treatment contrast
> #   (BUCKSKIN vs. ARAPHAHOE)
> anova(r.gls.means, L=c(1, 0, -1))


Denom. DF: 168
 F-test for linear combination(s)
varietyARAPAHOE varietyBUCKSKIN
              1              -1
  numDF F-value p-value
1     1 7.69673  0.0062
```

geostatistical analyses of experimental data

- field experiments in ecology, agriculture, forestry, … often give rise to spatial data
- classical analysis of variance of spatial experimental data ignores spatial structure of the data
- blocking and randomisation sometimes not effective to account for natural heterogeneity within experimental site
- residuals often violate independence assumption of classical analysis of variance methods
- explicit consideration of auto-correlation by generalised least squares estimation:

  $\Rightarrow$ increase in power to detect treatment effects

summary section 1

- geostatistical data $(y_i, \boldsymbol{x}_i, d_k(\boldsymbol{x}_i))$:
  1. response y_i
  2. location $\boldsymbol{x}_i$
  3. covariates $d_k(\boldsymbol{x}_i))$
  4. often approximately infinitesimal support
- models for geostatistical data decompose spatial variation (non-uniquely) into "large-scale" trend and local auto-correlated fluctuations
- trend modelled by linear regression model
- local fluctuations modelled by auto-correlated stochastic process

- objectives of geostatistical analyses:

  1. prediction of response variable
     - at location without measurement
     - for finite support targets (spatial means)
  2. estimation of parameters of (regression) models fitted to geostatistical data
  3. (likely more powerful) analyses of spatial experimental data (than classical ANOVA)

References

Pinheiro, J. C. and Bates, D. M. (2000). *Mixed-Effects Models in S and S-PLUS*. Springer Verlag.

2 an example of a geostatistical analysis with R

typical steps of a geostatistical analysis

$$\boxed{\text{exploratory analysis}}$$
$$\Downarrow$$
$$\boxed{\text{trend modelling}}$$
$$\Downarrow$$
$$\boxed{\text{modelling residual auto-correlation}}$$
$$\Downarrow$$
$$\left(\boxed{\text{statistical inference}}\right)$$
$$\Downarrow$$
$$\boxed{\text{model assessment by cross-validation}}$$
$$\Downarrow$$
$$\boxed{\text{computing spatial predictions}}$$

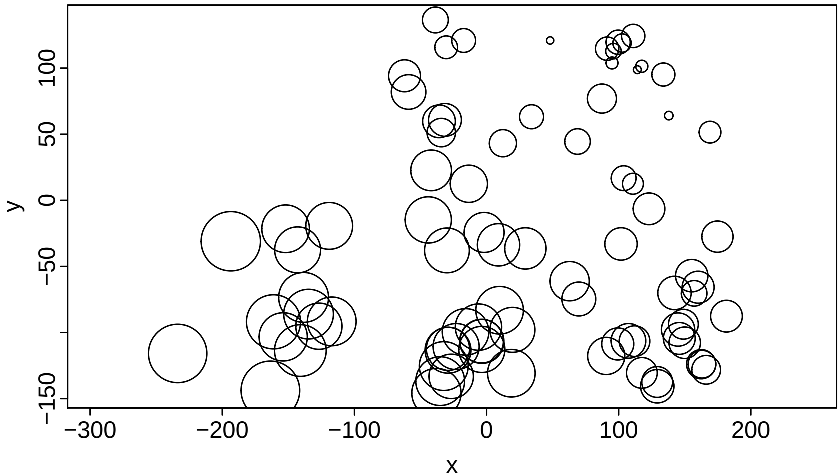
example data set: Wolfcamp aquifer data

```
> library(sp)
> library(georob)
> library(gstat)
> data(wolfcamp, package="georob")
> d.w <- wolfcamp
> coordinates(d.w) <- ~x+y
> summary(d.w)
```

```
Object of class SpatialPointsDataFrame
Coordinates:
        min      max
x -233.7217 181.5314
y -145.7884 136.4061
Is projected: NA
proj4string : [NA]
Number of points: 85
Data attributes:
    pressure
 Min.   : 312.1
 1st Qu.: 471.8
 Median : 547.7
```

```
Mean   : 610.3
3rd Qu.: 774.2
Max.   :1088.4
```

```
> plot(y~x, d.w, asp=1, cex=sqrt(d.w$pressure-300)/5)
```



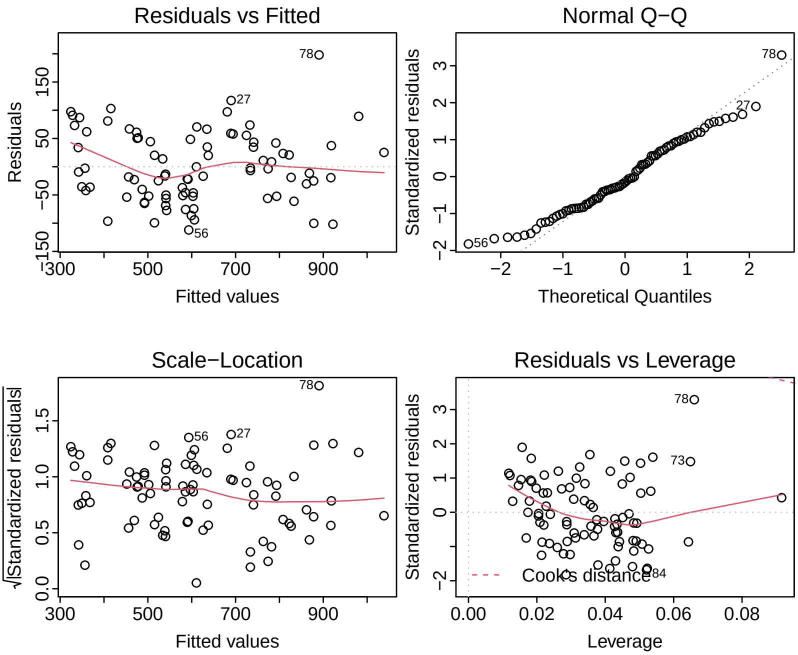
interactive data inspection by R package `rgl`

```
> library(rgl)
> open3d()
> plot3d(x=d.w$x, y=d.w$y,
+   z=d.w$pressure/3,
+   type="s", radius=7, col="red", aspect="iso",
+   xlab="x", ylab="y", zlab="pressure")
> clear3d()
```

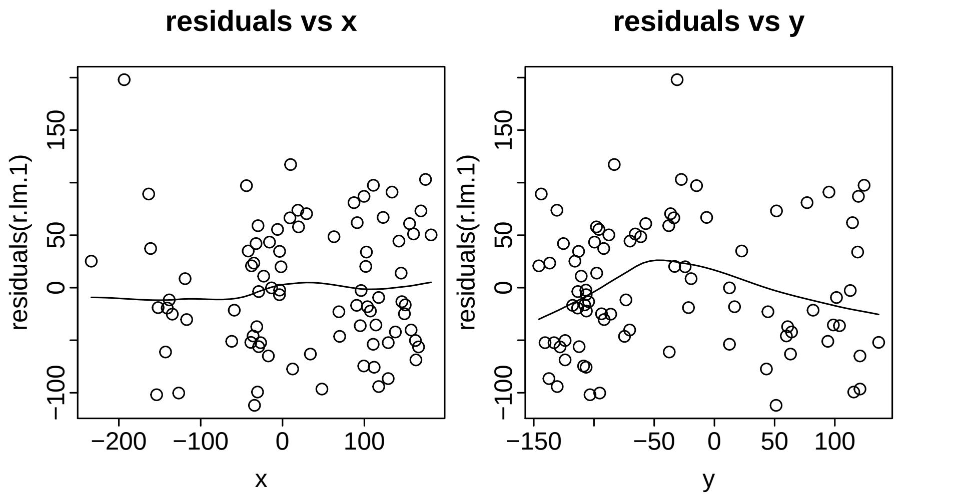
```
> r.lm.1 <- lm(pressure~x+y, d.w)
> summary(r.lm.1)
```

```
Call:
lm(formula = pressure ~ x + y, data = d.w)

Residuals:
     Min       1Q   Median       3Q      Max
-111.989  -50.297   -9.326   48.510  197.986

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 607.77066    7.52219   80.80   <2e-16
x            -1.27844    0.06552  -19.51   <2e-16
y            -1.13874    0.07739  -14.71   <2e-16

Residual standard error: 62.29 on 82 degrees of freedom
Multiple R-squared:  0.8909,        Adjusted R-squared:  0.8882
F-statistic: 334.8 on 2 and 82 DF,  p-value: < 2.2e-16
```

```
> op <- par(mfrow=c(2, 2)); plot(r.lm.1); par(op)
```



```
> op <- par(mfrow=c(1, 2))
> scatter.smooth(d.w$x, residuals(r.lm.1), xlab="x",
+   main="residuals vs x")
> scatter.smooth(d.w$y, residuals(r.lm.1), xlab="y",
+   main="residuals vs y")
> par(op)
```

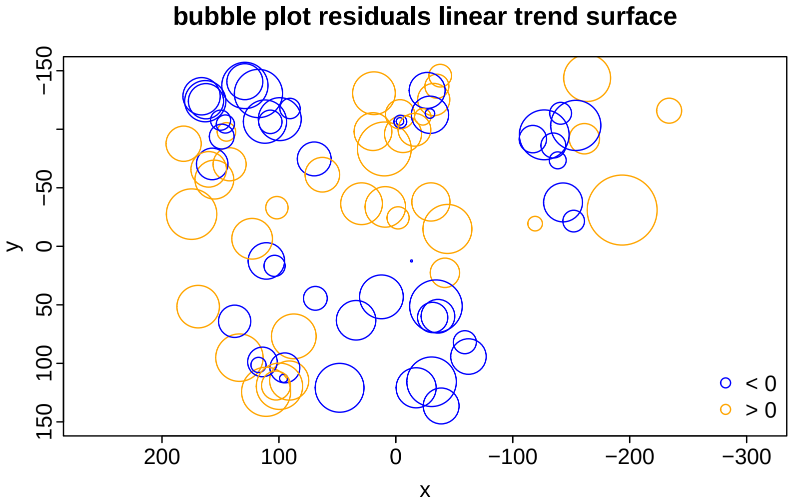


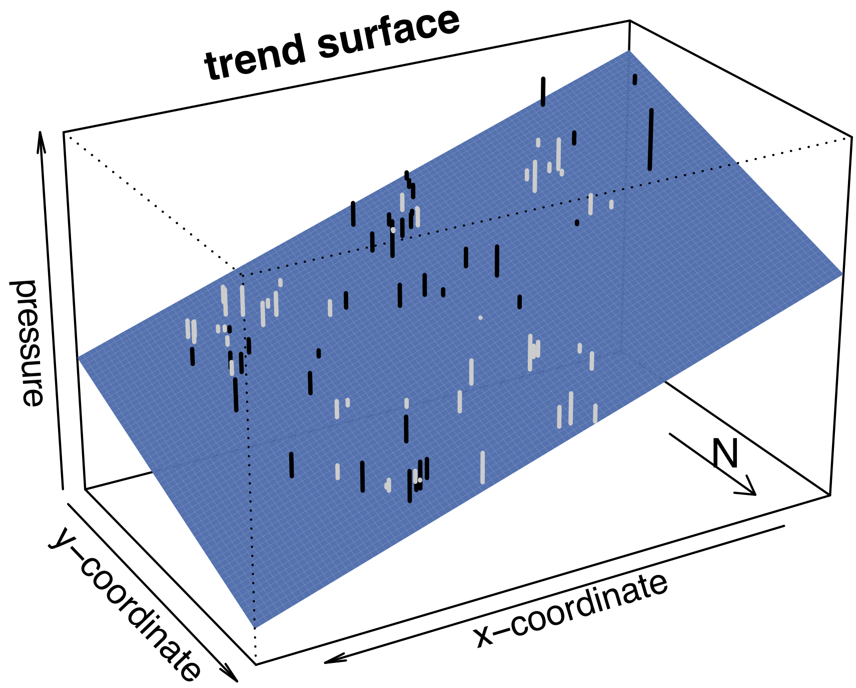
```
> r.lm.2 <- update(r.lm.1, .~.+I(x^2)+I(y^2)+x:y)
> summary(r.lm.2)


Call:
lm(formula = pressure ~ x + y + I(x^2) + I(y^2) + x:y, data = d.w)

Residuals:
     Min       1Q   Median       3Q      Max 
-124.405  -43.662   -2.337   39.017  199.198 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.203e+02  1.295e+01  47.902  < 2e-16
x           -1.075e+00  8.191e-02 -13.128  < 2e-16
y           -1.330e+00  8.861e-02 -15.008  < 2e-16
I(x^2)       8.994e-05  5.908e-04   0.152 0.879388
I(y^2)      -2.929e-03  1.101e-03  -2.659 0.009486
x:y          3.184e-03  8.790e-04   3.622 0.000515

Residual standard error: 57.02 on 79 degrees of freedom
Multiple R-squared:  0.9119,       Adjusted R-squared:  0.9063 
F-statistic: 163.6 on 5 and 79 DF,  p-value: < 2.2e-16
```

```
> anova(r.lm.1, r.lm.2)
```

```
Analysis of Variance Table

Model 1: pressure ~ x + y
Model 2: pressure ~ x + y + I(x^2) + I(y^2) + x:y
  Res.Df    RSS Df Sum of Sq      F   Pr(>F)
1     82 318200
2     79 256887  3     61313 6.2852 0.000702
```

```
> plot(y~x, d.w, asp=1, cex=sqrt(abs(residuals(r.lm.1)))/2,
+   xlim=c(200, -250), ylim=c(150, -150),
+   col=c("blue", NA, "orange")[sign(residuals(r.lm.1))+2],
+   main = "bubble plot residuals linear trend surface")
> legend("bottomright", pch=1, col=c("blue", "orange"),
+   legend=c("< 0", "> 0"), bty="n")
```





trend surface
pressure
y–coordinate
x–coordinate
N

2.3 estimating and modelling auto-correlation

- data: measurements $(y_{1,i}, y_{2,i})$, $i = 1, 2, \ldots, n$ about 2 response variables

- sample covariance

$$s_{1,2} = \frac{1}{(n-1)} \sum_{i=1}^{n} (y_{1,i} - \bar{y}_1)(y_{2,i} - \bar{y}_2)$$

  where $\bar{y}_1$ and $\bar{y}_2$ are the (arithmetic) sample means

- (Pearson) correlation coefficient

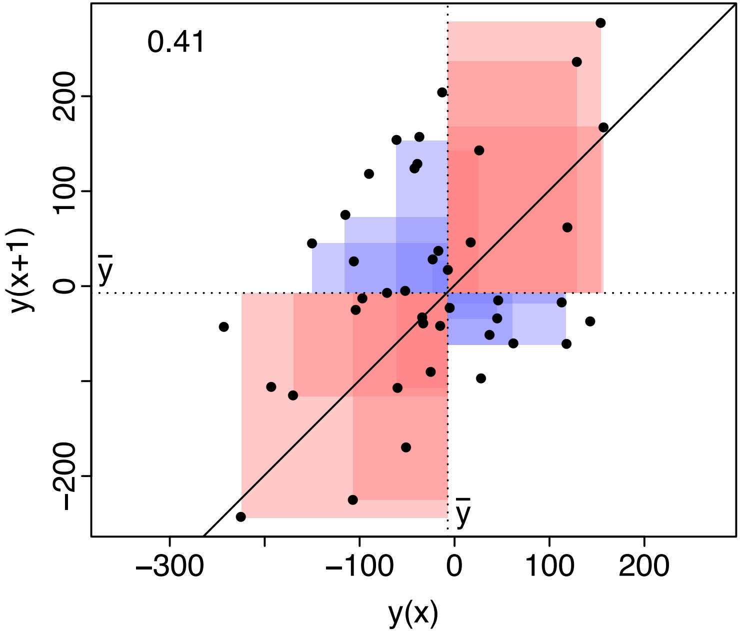
$$\hat{\rho} = \frac{s_{1,2}}{s_1 \; s_2}$$

  where s_1 and s_2 are the sample standard deviations

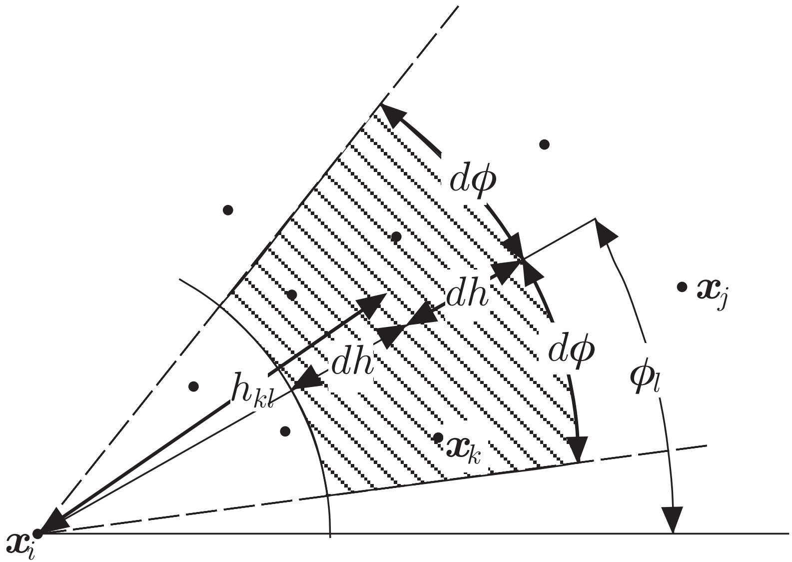
- “plug-in” estimator for auto-correlogram of time series

$$\hat{\rho}(h) = \frac{\sum_{i=1}^{n-h} (y_{i+h} - \bar{y})(y_i - \bar{y})}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

pro memoria: sample covariance and correlation

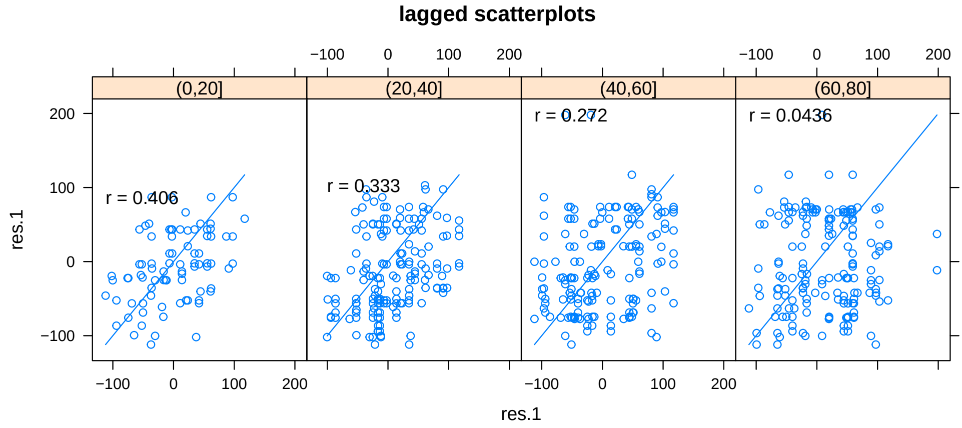


defining lags for irregular sampling grids



lag-scatter plots of trend surface residuals

```
> d.w$res.1 <- residuals(r.lm.1)
> hscat(res.1~1, d.w, breaks=seq(0, 80, by=20))
```



- $(k,l)^{\text{th}}$ lag class, $\boldsymbol{h}_{kl}$, characterized by distance, $(h_k - dh, h_k + dh]$, and angular class, $\phi_l - d\phi, \phi_l + d\phi]$
- N_{kl}: number of pairs of locations $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ with $\boldsymbol{x}_j - \boldsymbol{x}_i \approx \boldsymbol{h}_{kl}$
- estimator for covariance for lag class $\boldsymbol{h}_{kl}$:

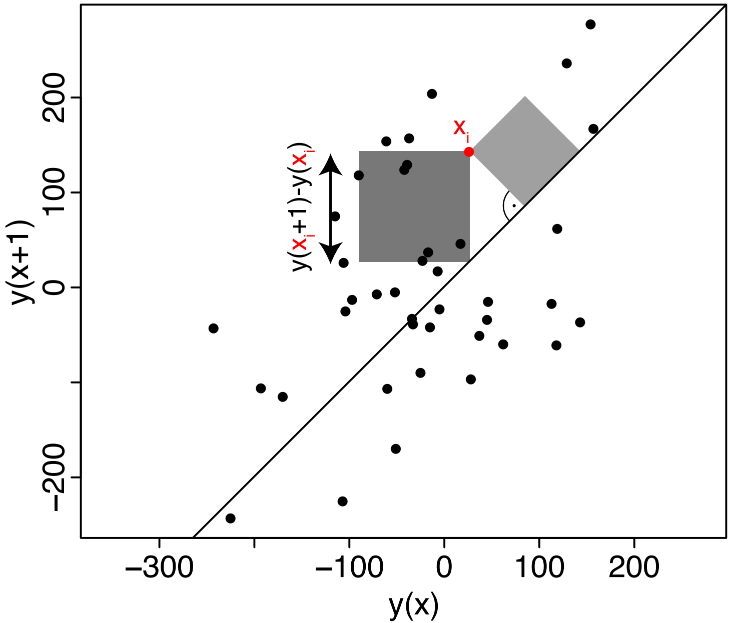
$$\widehat{\gamma}(\boldsymbol{h}_{kl}) = \frac{1}{N_{kl}} \sum_{(i,j)\in\boldsymbol{h}_{kl}} [y(\boldsymbol{x}_i) - \bar{y}][y(\boldsymbol{x}_j) - \bar{y}]$$

$\Rightarrow$ covariogram

- estimator for (semi-)variance for lag class $\boldsymbol{h}_{kl}$:

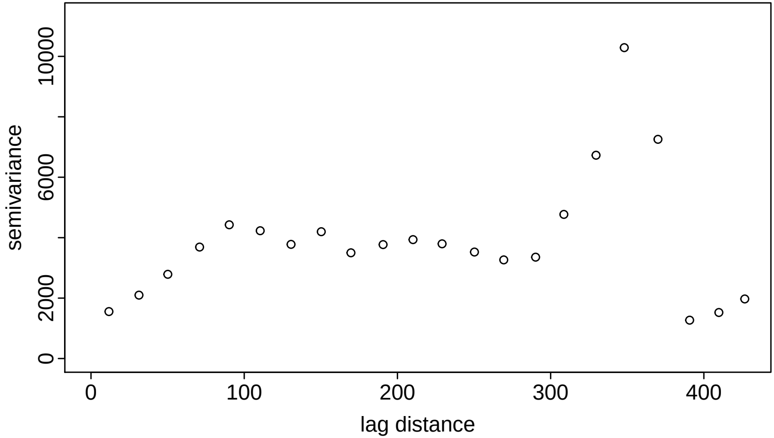
$$\widehat{V}(\boldsymbol{h}_{kl}) = \frac{1}{2\,N_{kl}} \sum_{(i,j)\in\boldsymbol{h}_{kl}} [y(\boldsymbol{x}_i) - y(\boldsymbol{x}_j)]^2$$

$\Rightarrow$ (semi-)variogram

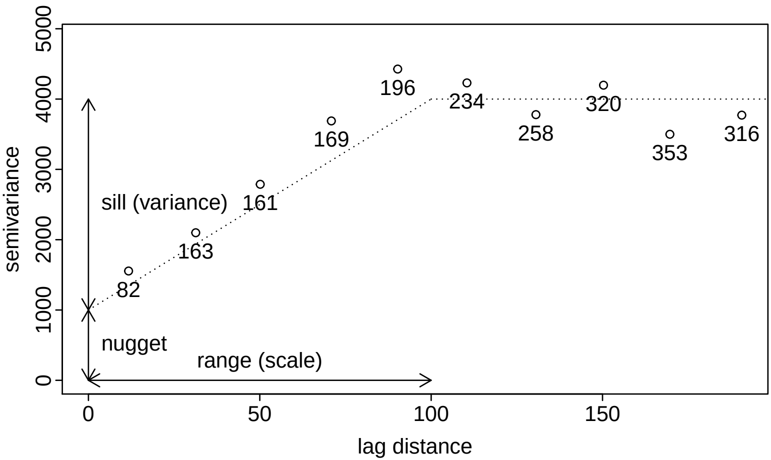

y(x+1)
200
100
0
−200
−300
−100
0
100
200
y(x)
x_i
$y(x_i+1)-y(x_i)$

variogram of trend surface residuals

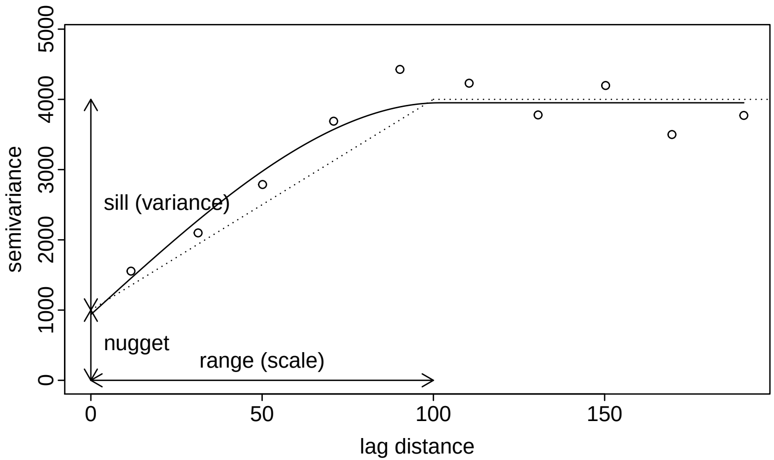

```
> plot(sample.variogram(residuals(r.lm.1),
+   locations=coordinates(d.w), lag.dist.def=20,
+   estimator="matheron"))
```



```
> r.v <- sample.variogram(residuals(r.lm.1),
+   locations=coordinates(d.w), lag.dist.def=20,
+   max.lag=200, estimator="matheron")
> plot(r.v)
> text(gamma~lag.dist, r.v, labels=npairs, pos=1)
```



```
> r.v.sph <- fit.variogram.model(r.v,
+   variogram.model="RMspheric",
+   param=c(variance=3000, nugget=1000, scale=100))
> lines(r.v.sph)
```



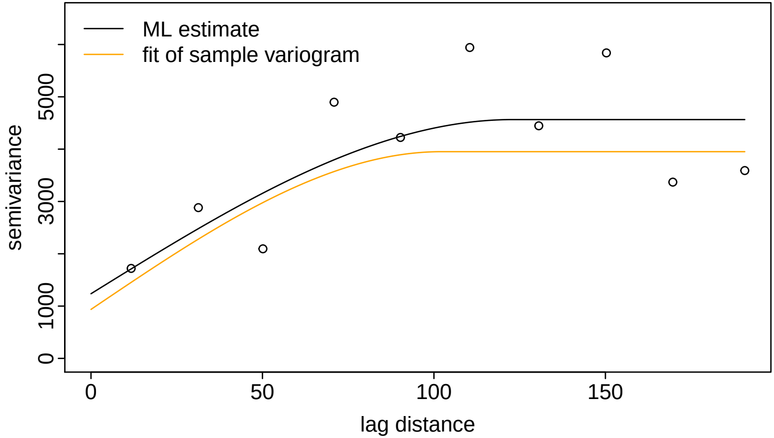
2.4 fitting spatial model by maximum likelihood

```
> r.georob.1 <- georob(pressure~x+y, d.w,
+   locations=~x+y, variogram.model="RMspheric",
+   param=c(variance=3000, nugget=1000, scale=100),
+   tuning.psi=1000, control=control.georob(ml.method="ML"))
> summary(r.georob.1)
```

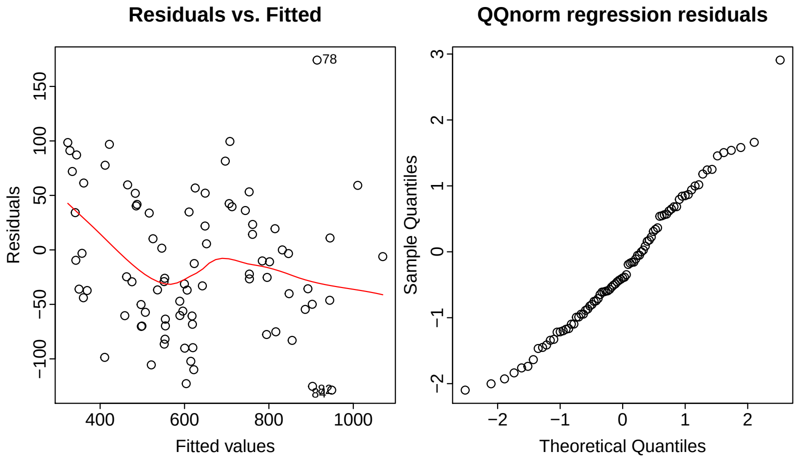
```
...
Maximized log-likelihood: -458.3671
...
Variogram:  RMspheric
              Estimate   Lower  Upper
variance       3328.90 1453.95 7621.7
snugget(fixed)    0.00      NA     NA
nugget         1236.27  616.01 2481.1
scale           122.95   95.13  158.9

Fixed effects coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 620.3550    17.0641  36.354  < 2e-16
x            -1.3256     0.1360  -9.750 2.33e-15
y            -1.2061     0.1793  -6.727 2.16e-09
...
```

```
> summary(r.lm.1)


Call:
lm(formula = pressure ~ x + y, data = d.w)

Residuals:
     Min       1Q   Median       3Q      Max
-111.989  -50.297   -9.326   48.510  197.986

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 607.77066    7.52219   80.80   <2e-16
x            -1.27844    0.06552  -19.51   <2e-16
y            -1.13874    0.07739  -14.71   <2e-16

Residual standard error: 62.29 on 82 degrees of freedom
Multiple R-squared:  0.8909,        Adjusted R-squared:  0.8882
F-statistic: 334.8 on 2 and 82 DF,  p-value: < 2.2e-16
```

```
> plot(r.georob.1, lag.dist.def=20, max.lag=200)
> lines(r.v.sph, col=”orange”)
> legend(”topleft”, lty=1, col=c(”black”, ”orange”), bty=”n”,
+   legend=c(”ML estimate”,
+     ”fit of sample variogram”))
```



```
> op <- par(mfrow=c(1,2))
> plot(r.georob.1, what = "ta")
> qqnorm(rstandard(r.georob.1, level=0),
+   main="QQnorm regression residuals")
> par(op)
```



- data analysis often leads to a set of equally plausible candidate models that use different set of covariates and different variograms

⇒ compare fit of candidate models by hypothesis tests taking auto-correlation properly into account

⇒ use established goodness-of-fit criteria (AIC, BIC) to select a "best" model, again taking auto-correlation into account

⇒ use cross-validation to compare the power of candidate models to *predict new data*

ML fit quadratic trend surface model

```
> r.georob.2 <- update(r.georob.1, .~.+I(x^2)+I(y^2)+x:y)
> summary(r.georob.2)
```

```
...
Maximized log-likelihood: -455.0776
...
Variogram:  RMspheric
              Estimate   Lower  Upper
variance       2060.36  784.17 5413.4
snugget(fixed)    0.00      NA     NA
nugget         1402.08  654.37 3004.1
scale           103.85   30.07  358.7

Fixed effects coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.111e+02  2.165e+01  28.225  < 2e-16
x           -1.168e+00  1.317e-01  -8.868 1.76e-13
y           -1.268e+00  1.566e-01  -8.098 5.61e-12
I(x^2)       1.297e-03  9.419e-04   1.377    0.172
I(y^2)      -2.319e-03  1.652e-03  -1.404    0.164
x:y          2.290e-03  1.500e-03   1.526    0.131
...
```

```
> waldtest(r.georob.2, r.georob.1, test="F")
```

```
Wald test

Model 1: pressure ~ x + y + I(x^2) + I(y^2) + x:y
Model 2: pressure ~ x + y
  Res.Df Df      F   Pr(>F)
1     79
2     82 -3 2.7684 0.04713
```

```
> step(r.georob.2)


Start:  AIC=922.16
pressure ~ x + y + I(x^2) + I(y^2) + x:y

         Df    AIC Converged
- I(x^2)  1 922.05         1
- I(y^2)  1 922.13         1
<none>      922.16
- x:y     1 922.49         1

Step:  AIC=922.05
pressure ~ x + y + I(y^2) + x:y

         Df    AIC Converged
<none>      922.05
+ I(x^2)  1 922.16         1
- I(y^2)  1 922.54         1
- x:y     1 924.61         1

Tuning constant:  1000

Fixed effects coefficients:
(Intercept)           x           y      I(y^2)
 627.526464   -1.148338   -1.347008   -0.002587
        x:y
```

```
   0.003005

Variogram:  RMspheric
 variance(fixed)    snugget(fixed)     nugget(fixed)
          2060.4               0.0            1402.1
    scale(fixed)
           103.8
```

```
> r.cv.1 <- cv(r.georob.1, seed=5426)
> r.cv.2 <- cv(r.georob.2, seed=5426)
```

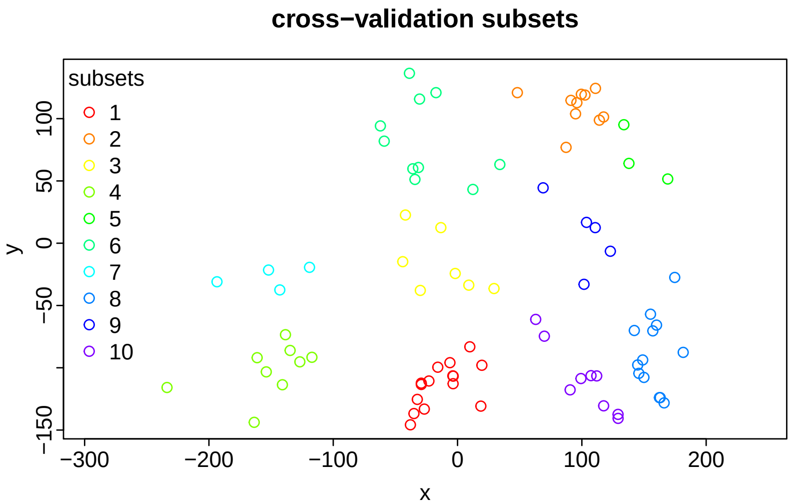
```
> summary(r.cv.1)
```

```
Statistics of cross-validation prediction errors
      me     mede     rmse     made      qne     msse
-13.069  -15.978   66.770   67.132   67.358    1.066
 medsse     crps
  0.474   38.107
```

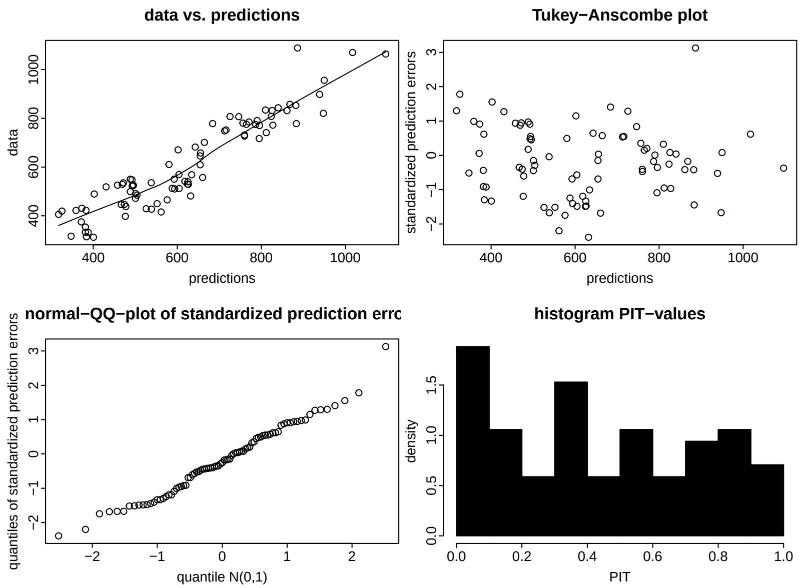
```
> summary(r.cv.2)
```

```
Statistics of cross-validation prediction errors
      me     mede     rmse     made      qne     msse
-3.9202  -2.0977  80.8784  80.0463  76.8096   1.5170
 medsse     crps
 0.7419  44.8059
```

```
> op <- palette(rainbow(12))
> plot(y~x, r.cv.1$pred, asp=1, col=subset,
+   main = "cross-validation subsets")
> legend("topleft", pch=1, col=1:10,
+   title="subsets", legend=1:10,  bty="n")
> palette(op)
```



```
> op <- par(mfrow=c(2, 2))
> plot(r.cv.1, type="sc"); plot(r.cv.1, type="ta")
> plot(r.cv.1, type="qq"); plot(r.cv.1, type="hist.pit")
> par(op)
```



- workhorse: $\Rightarrow$ *universal (or external-drift) kriging*

- prediction of signal $S(\boldsymbol{x}_0)$ at location $\boldsymbol{x}_0$ without measurement

$$\hat{S}(\boldsymbol{x}_0) = \sum_{i=1}^{n} \kappa_i(\boldsymbol{x}_0)\, y(\boldsymbol{x}_i)$$

  where weights $\kappa_i(\boldsymbol{x}_0)$ depend on trend model and variogram

- kriging provides in addition an estimate of the variance of the prediction error $S(\boldsymbol{x}_0) - \hat{S}(\boldsymbol{x}_0)$

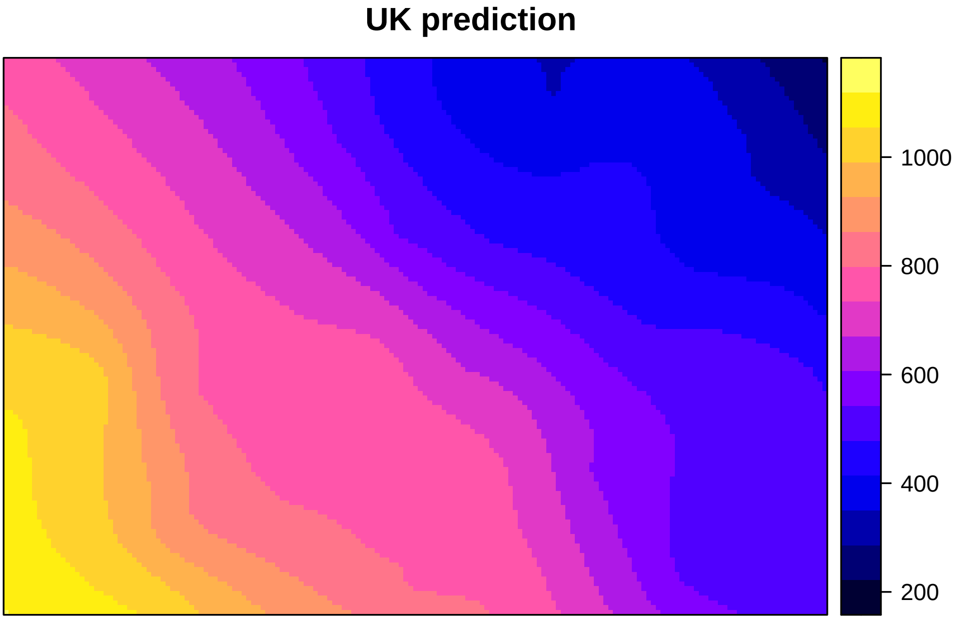
```
> d.w.grid <- expand.grid(
+   x = seq(-240, 190, by= 2.5),
+   y = seq(-150, 140, by= 2.5)
+ )
> r.uk <- predict(r.georob.1, newdata=d.w.grid)
> coordinates(r.uk) <- ~x+y
> gridded(r.uk) <- TRUE
> fullgrid(r.uk) <- TRUE
```

```
> summary(r.uk)
```

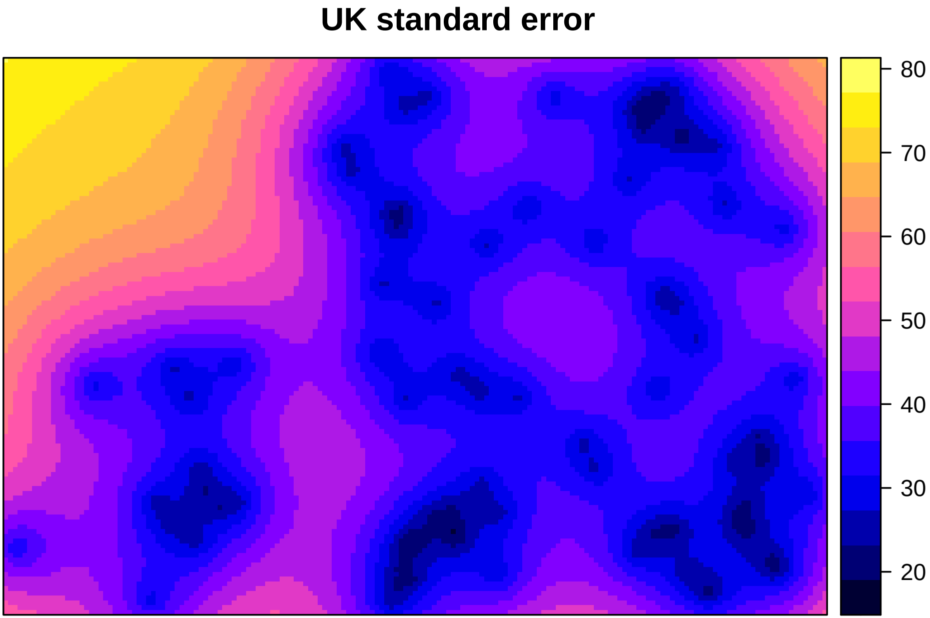
```
Object of class SpatialGridDataFrame
Coordinates:
      min    max
x -241.25 191.25
y -151.25 141.25
Is projected: NA
proj4string : [NA]
Grid attributes:
  cellcentre.offset cellsize cells.dim
x              -240      2.5       173
y              -150      2.5       117
Data attributes:
```

```
      pred              se              lower        
 Min.   : 220.8   Min.   :18.95   Min.   :  92.89  
 1st Qu.: 497.2   1st Qu.:33.21   1st Qu.: 426.74  
 Median : 663.5   Median :38.20   Median : 569.44  
 Mean   : 656.5   Mean   :41.49   Mean   : 575.19  
 3rd Qu.: 789.3   3rd Qu.:46.03   3rd Qu.: 709.57  
 Max.   :1119.7   Max.   :77.22   Max.   :1012.69  
     upper       
 Min.   : 348.7  
 1st Qu.: 563.0  
 Median : 750.3  
 Mean   : 737.8  
 3rd Qu.: 878.7  
 Max.   :1226.8  
```

```
> spplot(r.uk, zcol="pred", main="UK prediction")
```



```
> spplot(r.uk, zcol="se", main="UK standard error")
```



summary section 2

- modelling trend by linear regression model based on insights from exploratory analysis
- modelling residual auto-correlation by variogram
- simultaneous estimation of regression coefficients of trend model and parameters of variogram by (restricted) maximum likelihood
- hypothesis tests for spatial data should take auto-correlation into account

3 some theory on stochastic processes

- *spatial stochastic process* $\{S(\boldsymbol{x})\}$: collection (= set) of random variables $S(\boldsymbol{x}) : \boldsymbol{x} \in D \subset \mathbb{R}^d$, with a well defined joint distribution

- *strictly stationary process*: joint distributions of arbitrary collections of random variables $\{S(\boldsymbol{x}_1), \ldots, S(\boldsymbol{x}_n)\}$ are invariant to translations by vector $\boldsymbol{h} \in \mathbb{R}^d$ $\Rightarrow$ $\{S(\boldsymbol{x}_1), \ldots, S(\boldsymbol{x}_n)\}$ and $\{S(\boldsymbol{x}_1 + \boldsymbol{h}), \ldots, S(\boldsymbol{x}_n + \boldsymbol{h})\}$ have same joint distribution
$$F(s_1, \ldots, s_n; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = F(s_1, \ldots, s_n; \boldsymbol{x}_1 + \boldsymbol{h}, \ldots, \boldsymbol{x}_n + \boldsymbol{h})$$

- *weakly or second-order stationary process*: distributions of arbitrary pairs of random variables $(S(\boldsymbol{x}), S(\boldsymbol{x} + \boldsymbol{h}))$ satisfy

  1. E $[S(\boldsymbol{x})] =$ constant (independent of $\boldsymbol{x}$)
  2. Cov $[S(\boldsymbol{x} + \boldsymbol{h}), S(\boldsymbol{x}),] = \gamma(\boldsymbol{h})$ (independent of $\boldsymbol{x}$)
  3. Var $[S(\boldsymbol{x})] =$ constant (independent of $\boldsymbol{x}$)

$\Rightarrow$ strict stationarity implies weak stationarity

$\Rightarrow$ stationarity required for estimation/prediction with single realisation of stochastic process

isotropic stochastic process

- weakly stationary process that is invariant to rotations

$$\text{Cov}\left[S(\boldsymbol{x}), S(\boldsymbol{x}+\boldsymbol{h})\right] = \gamma(h) \quad \text{with} \quad h = ||\boldsymbol{h}|| = \sqrt{\sum_{i=1}^{d} h_i^2}$$

$\Rightarrow$ unless stated otherwise, only isotropic and weakly stationary processes are considered

- all finite-dimensional joint distributions are multivariate normal

$$F(s_1, \ldots, s_n; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

  with mean vector $\boldsymbol{\mu}^{\mathsf{T}} = (\mu_1, \ldots, \mu_n); \;\; \mu_i = \mathsf{E}\left[S(\boldsymbol{x}_i)\right]$
  and covariance matrix with elements $[\boldsymbol{\Sigma}]_{ij} = \mathsf{Cov}\left[S(\boldsymbol{x}_i), S(\boldsymbol{x}_j)\right]$

- weakly stationary Gaussian process is also strictly stationary

- definition of *variogram* $V(\boldsymbol{h})$ and *covariance function* $\gamma(\boldsymbol{h})$

$$V(\boldsymbol{h}) = \frac{1}{2}\mathsf{Var}\left[S(\boldsymbol{x}+\boldsymbol{h}) - S(\boldsymbol{x})\right] \qquad \gamma(\boldsymbol{h}) = \mathsf{Cov}\left[S(\boldsymbol{x}+\boldsymbol{h}), S(\boldsymbol{x})\right]$$

- relation between variogram and covariance function

$$V(\boldsymbol{h}) = \gamma(0) - \gamma(\boldsymbol{h}) \quad \text{with} \quad \gamma(0) = \mathsf{Var}\left[S(\boldsymbol{x})\right]$$

- relation between correlogram and covariance function

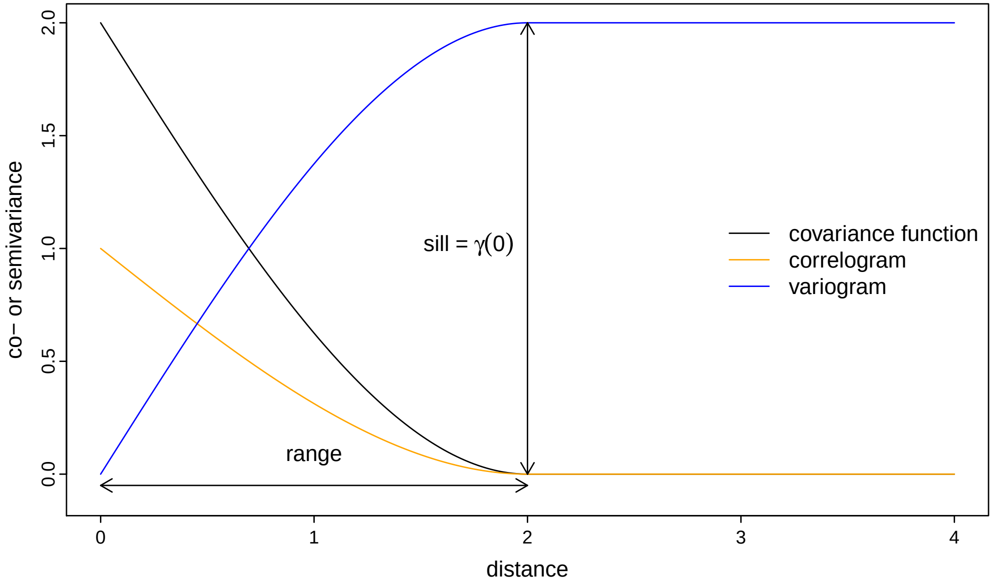
$$\rho(\boldsymbol{h}) = \frac{\gamma(\boldsymbol{h})}{\gamma(0)}$$

- relation between variogram and correlogram

$$V(\boldsymbol{h}) = \gamma(0)\left(1 - \rho(\boldsymbol{h})\right)$$

- symmetry

$$V(\boldsymbol{h}) = V(-\boldsymbol{h}) \quad \gamma(\boldsymbol{h}) = \gamma(-\boldsymbol{h}) \quad \rho(\boldsymbol{h}) = \rho(-\boldsymbol{h})$$


co− or semivariance
0.0
0.5
1.0
1.5
2.0
0
1
2
3
4
distance
range
sill = γ(0)
covariance function
correlogram
variogram

- $\{S(\boldsymbol{x})\}$ is mean square continuous

$$\mathsf{E}\left[(S(\boldsymbol{x}+\boldsymbol{h}) - S(\boldsymbol{x}))^2\right] \to 0 \quad \text{as} \quad \boldsymbol{h} \to 0$$

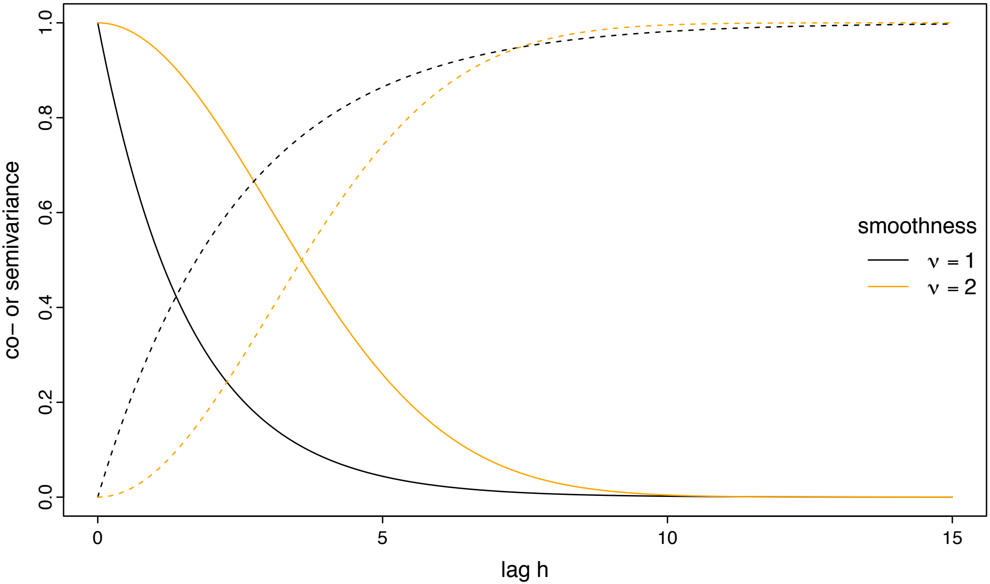
  if $\quad V(h) \to 0 \quad$ and $\quad \gamma(h) \to \gamma(0) \quad$ for $\quad h = ||\boldsymbol{h}|| \to 0$

- $\{S(\boldsymbol{x})\}$ is (once) mean square differentiable with 1st derivative process $\{S'(\boldsymbol{x})\}$

$$\mathsf{E}\left[\left(\frac{S(\boldsymbol{x}+\boldsymbol{h}) - S(\boldsymbol{x})}{\boldsymbol{h}} - S'(\boldsymbol{x})\right)^2\right] \to 0$$

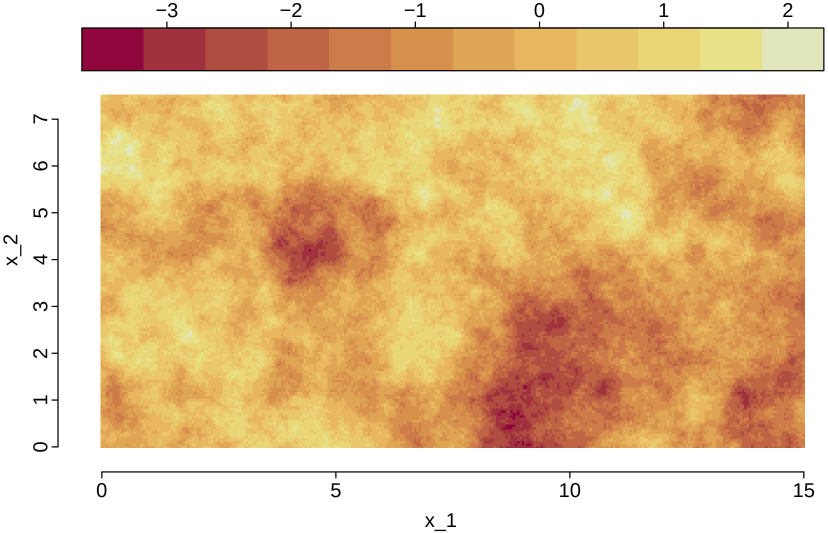
  if $\quad \gamma(h)$ (and $V(h)$) is twice differentiable at $h = 0$

- higher order mean square derivatives analogously defined

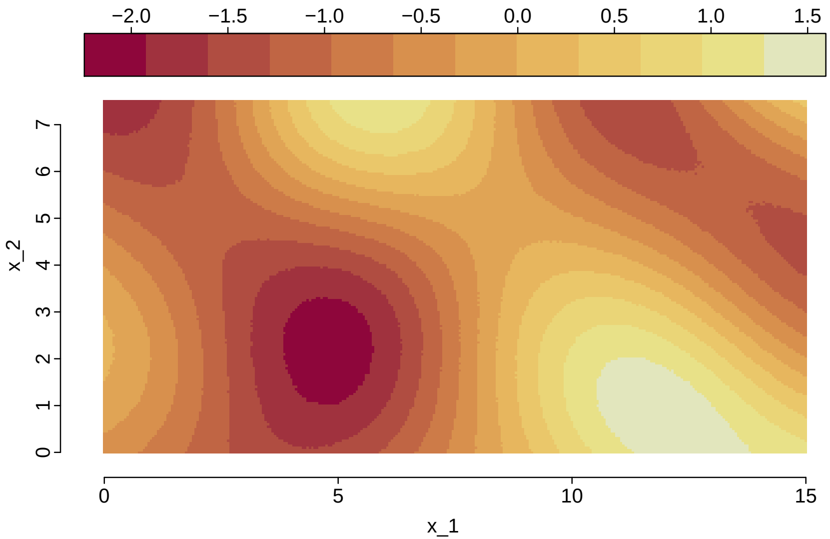


co− or semivariance
lag h
smoothness
ν = 1
ν = 2
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15

```
> RFoptions(spConform=TRUE)
> plot(RFsimulate(RMstable(var=1, scale=2.5, alpha=1),
+   x=x1, y=x2, grid=TRUE))
```



```
> RFoptions(spConform=TRUE)
> plot(RFsimulate(RMstable(var=1, scale=4.3, alpha=2),
+   x=x1, y=x2, grid=TRUE))
```



3.6 examples of isotropic covariance functions

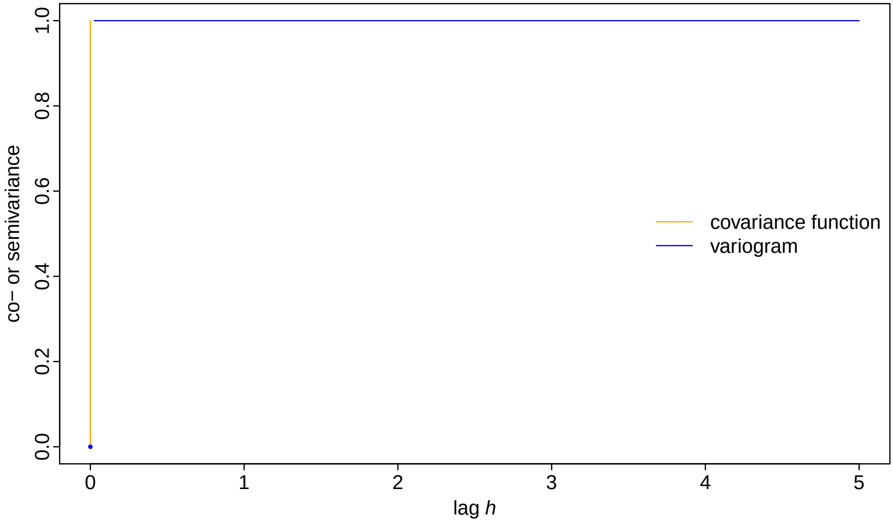
preliminary remark: all models can be used for variograms as well by the relation $V(h) = \gamma(0) - \gamma(h)$; $\Rightarrow$ in the sequel $\gamma(0) = 1$

nugget effect covariance models *absence of auto-correlation*

$$\gamma(h) = \begin{cases} 1 & \text{if } h = 0 \\ 0 & \text{otherwise} \end{cases} \qquad V(h) = \begin{cases} 0 & \text{if } h = 0 \\ 1 & \text{otherwise} \end{cases}$$

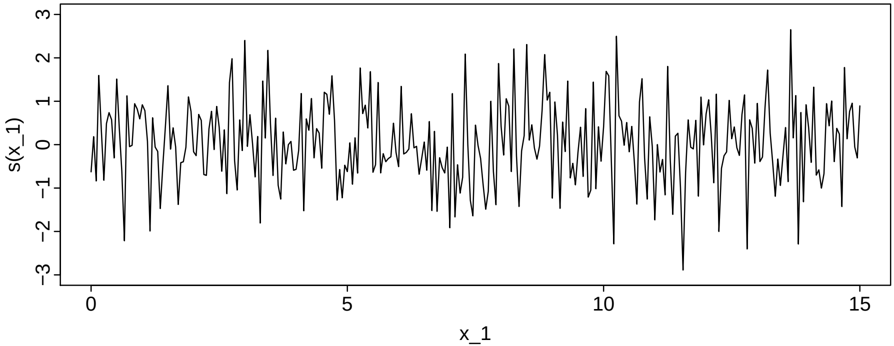
$\Rightarrow$ $\{S(\boldsymbol{x})\}$ spatial white noise process ($p(\boldsymbol{u}) =$ constant)

- mechanism: measurement error and small-scale spatial variation
- valid for all dimensions d of study region
- Gaussian $\{S(\boldsymbol{x})\}$ with nugget effect covariance function has *non-continuous* realisations
- see `?RMnugget` (package `RandomFields`)

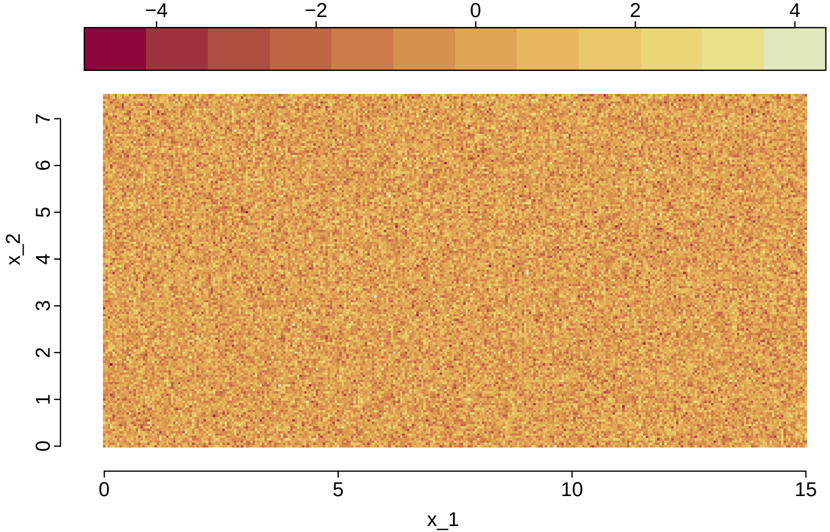


co− or semivariance
0.0
0.2
0.4
0.6
0.8
1.0
0
1
2
3
4
5
lag h
covariance function
variogram

```
> library(RandomFields); RFoptions(spConform=FALSE)
> x1 <- seq(0, 15, length=301)
> set.seed(1)
> plot(x1, RFsimulate(RMnugget(var=1), x=x1), type="l")
```



```
> x2 <- seq(0, 7.5, length=151)
> RFoptions(spConform=TRUE)
> plot(RFsimulate(RMnugget(var=1), x=x1, y=x2, grid=TRUE))
```



examples of isotropic covariance functions

Whitte-Matérn covariance

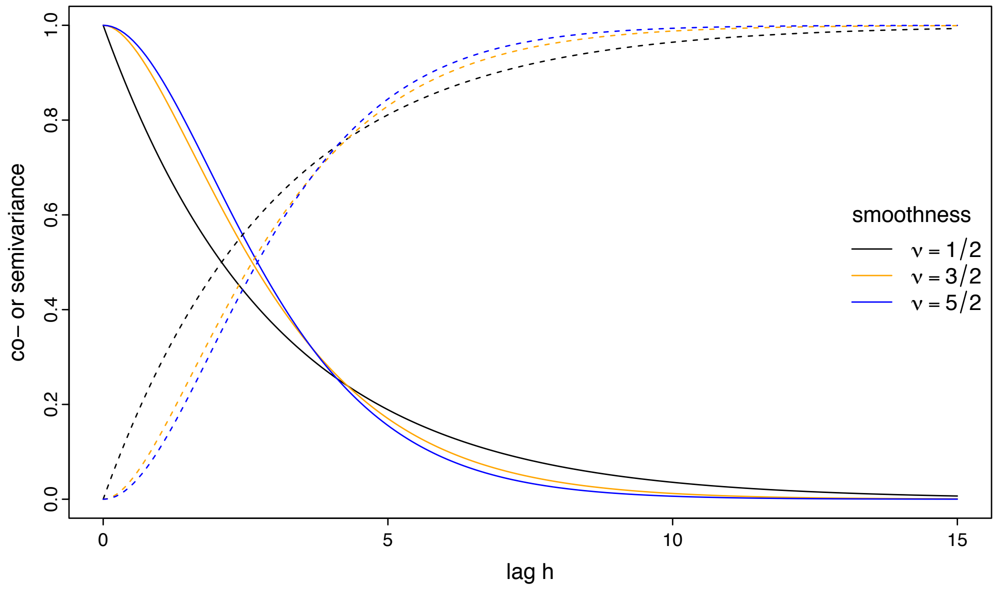
$$\gamma(h) = \frac{2^{1-\nu}}{\Gamma(\nu)} (\sqrt{2\nu}\frac{h}{\alpha})^{\nu} K_{\nu}(\sqrt{2\nu}\frac{h}{\alpha})$$

where $\alpha > 0$ is the range (scale) and $\nu > 0$ the smoothness parameter, K_ν is the modified Bessel function of order ν

- valid for all d
- Gaussian $\{S(\boldsymbol{x})\}$ with Wittle-Matérn covariance have m times differentiable realisations where m is largest integer with $m < \nu$
- special cases

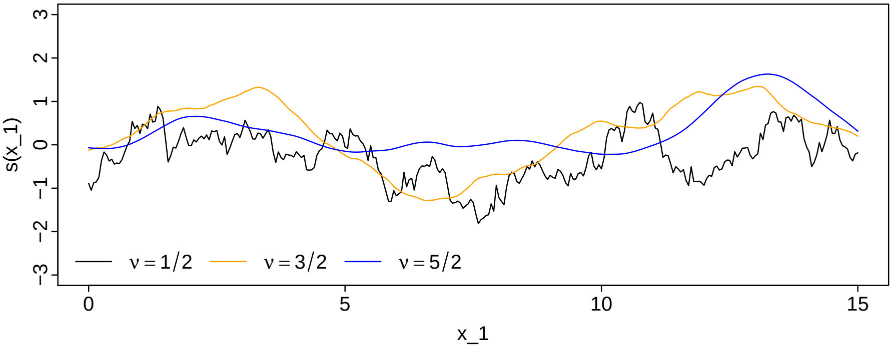
$$\begin{array}{lll} \nu = 1/2 & \gamma(h) = \text{exp}(-h/\alpha) & m = 0 \\ \nu = 3/2 & \gamma(h) = (1 + h/\alpha)\,\text{exp}(-h/\alpha) & m = 1 \\ \nu = 5/2 & \gamma(h) = (1 + h/\alpha + h^2/(3\alpha))\,\text{exp}(-h/\alpha) & m = 2 \end{array}$$

- see `?RMmatern` (package `RandomFields`)

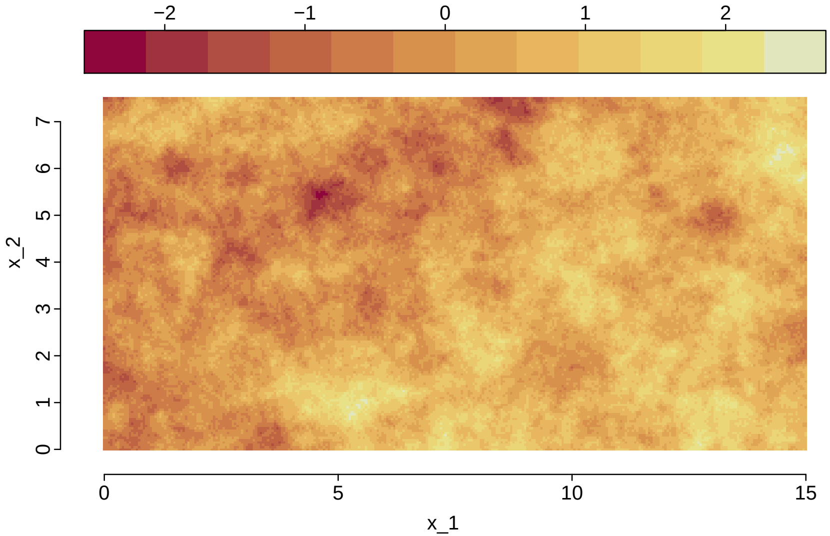


1.0
0.8
0.6
0.4
0.2
0.0
co− or semivariance
0
5
10
15
lag h
smoothness
ν = 1/2
ν = 3/2
ν = 5/2

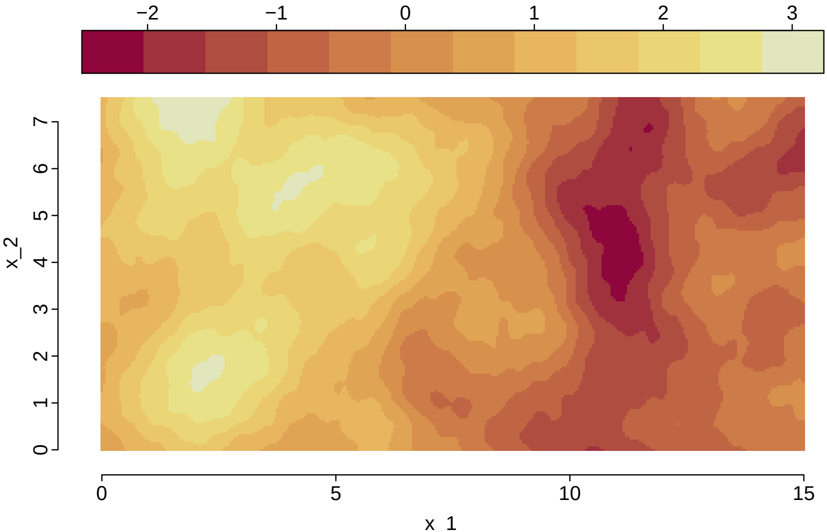
```
> RFoptions(spConform=FALSE)
> plot(x1, RFsimulate(RMmatern(var=1, scale=3, nu=0.5),
+   x=x1), type="l")
> lines(x1, RFsimulate(RMmatern(var=1, scale=2.7,
+   nu=1.5), x=x1), col=2)
> lines(x1, RFsimulate(RMmatern(var=1, scale=2.6,
+   nu=2.5), x=x1), col=3)
```



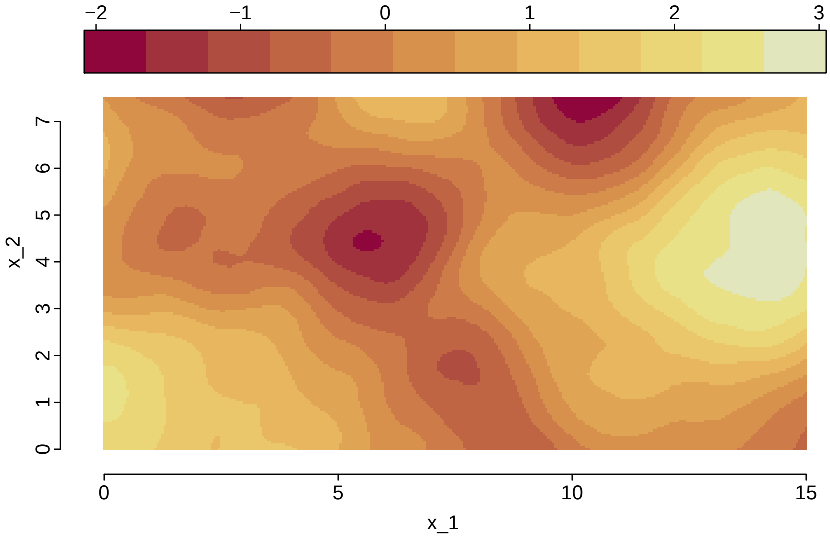
```
> RFoptions(spConform=TRUE)
> plot(RFsimulate(RMmatern(var=1, scale=3,
+   nu=0.5), x=x1, y=x2, grid=TRUE))
```



```
> RFoptions(spConform=TRUE)
> plot(RFsimulate(RMmatern(var=1, scale=2.7,
+   nu=1.5), x=x1, y=x2, grid=TRUE))
```



```
> RFoptions(spConform=TRUE)
> plot(RFsimulate(RMmatern(var=1, scale=2.6,
+   nu=2.5), x=x1, y=x2, grid=TRUE))
```

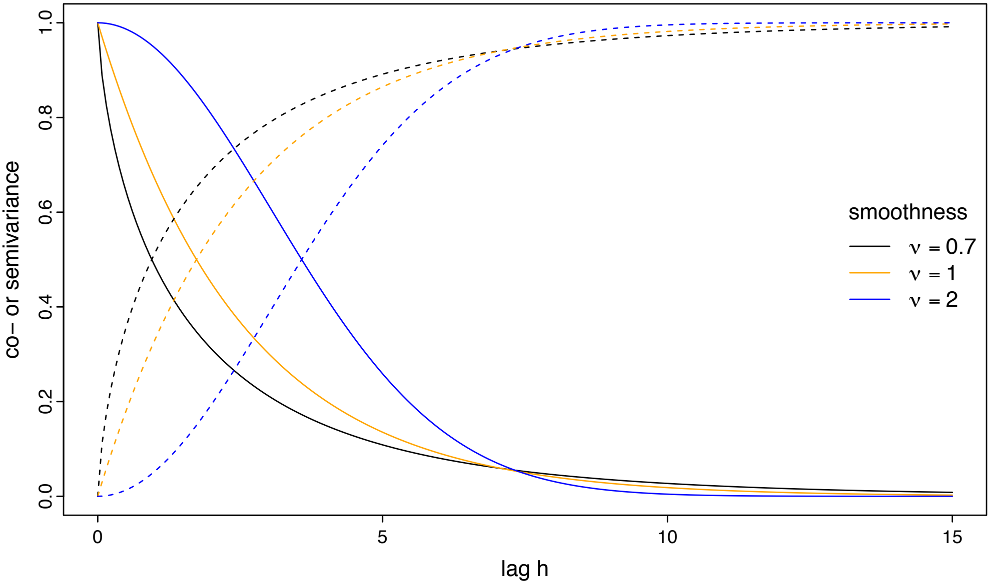


powered exponential or stable covariance

$$\gamma(h) = \text{exp}(-(h/\alpha)^{\nu})$$

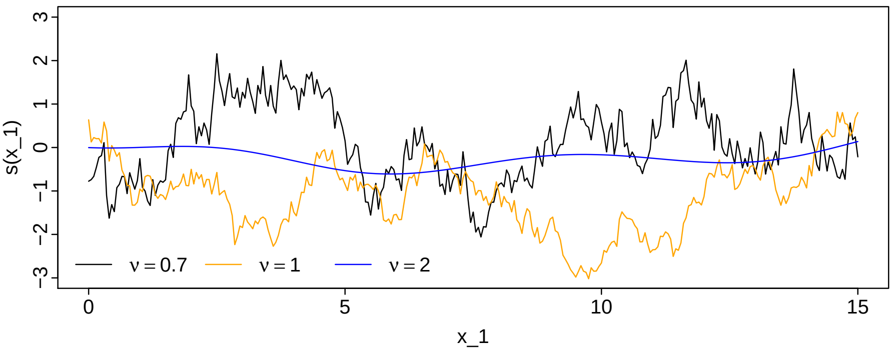
where $\alpha > 0$ is the range (scale) and $0 < \nu \leq 2$ the smoothness parameter

- valid for all d
- Gaussian $\{S(\boldsymbol{x})\}$ with stable covariance with $\nu = 2$ (Gaussian covariance) have realisations that are an infinite number of times differentiable but non-differentiable for $\nu < 2$
- see `?RMstable` (package `RandomFields`)



co– or semivariance
lag h
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
smoothness
ν = 0.7
ν = 1
ν = 2

```
> RFoptions(spConform=FALSE)
> plot(x1, RFsimulate(RMstable(var=1, scale=1.6, alpha=0.7),
+   x=x1), type="l")
> lines(x1, RFsimulate(RMstable(var=1, scale=2.5, alpha=1),
+   x=x1), col=2)
> lines(x1, RFsimulate(RMstable(var=1, scale=4.3, alpha=2),
+   x=x1), col=3)
```



examples of isotropic covariance functions

spherical covariance family with compact support: $\gamma(h) = 0$ for $h > \alpha$

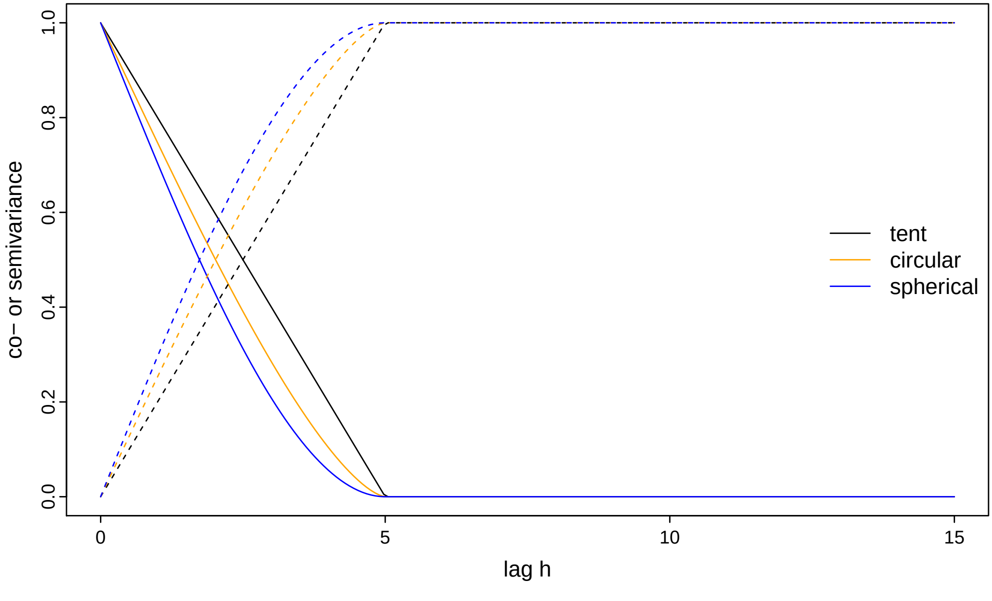
- generated by computing average of spatial white noise within a moving ball with radius $\alpha/2$ in $\mathbb{R}^d$
- models for $d \leq 3$

$$d = 1 \qquad \gamma(h) = \begin{cases} 1 - \frac{h}{\alpha} & \text{if } h \leq \alpha \\ 0 & \text{otherwise} \end{cases} \qquad \text{triangle}$$

$$d = 2 \qquad \gamma(h) = \begin{cases} \frac{2}{\pi}(\text{arccos}(\frac{h}{\alpha}) - \frac{h}{\alpha}\sqrt{1 - \frac{h^2}{\alpha^2}}) & \text{if } h \leq \alpha \\ 0 & \text{otherwise} \end{cases} \qquad \text{circular}$$

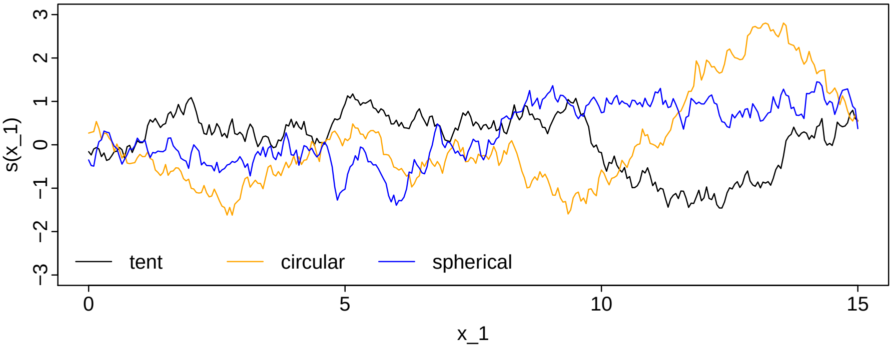
$$d = 3 \qquad \gamma(h) = \begin{cases} 1 - \frac{3h}{2\alpha} + \frac{h^3}{2\alpha^3} & \text{if } h \leq \alpha \\ 0 & \text{otherwise} \end{cases} \qquad \text{spherical}$$

- all models with non-differentiable Gaussian realisations
- “moving average” covariance functions exist also for $d > 3$
- see `?RMtent`, `?RMcircular`, `?RMspheric` (package `RandomFields`)



1.0
0.8
0.6
0.4
0.2
0.0
co− or semivariance
0
5
10
15
lag h
tent
circular
spherical

```
> RFoptions(spConform=FALSE)
> plot(x1, RFsimulate(RMtent(var=1, scale=5), x=x1),
+ type="l")
> lines(x1, RFsimulate(RMcircular(var=1, scale=5), x=x1),
+ col=2)
> lines(x1, RFsimulate(RMspheric(var=1, scale=5), x=x1),
+ col=3)
```

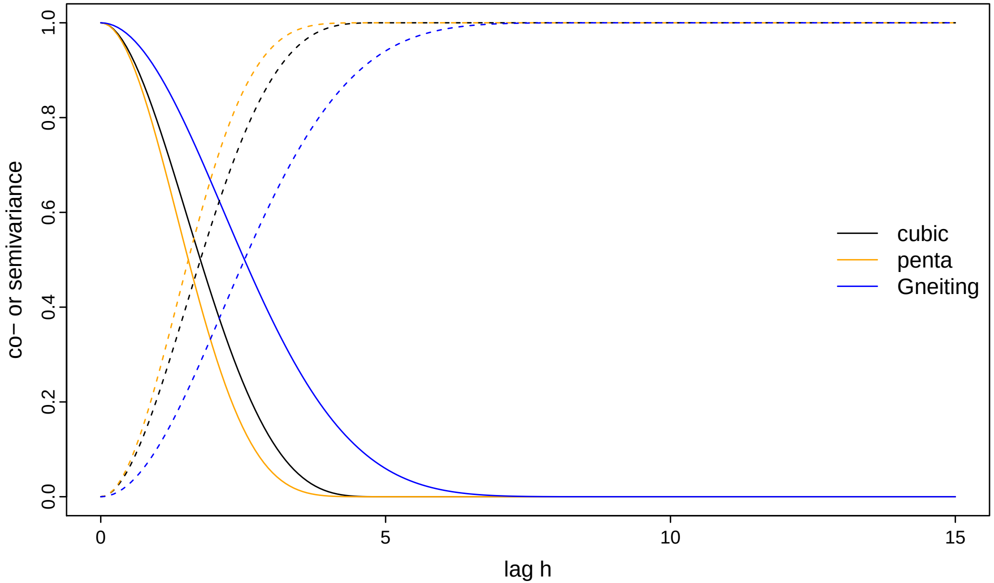


compact support covariance functions with differentiable Gaussian realisations

- ***cubic covariance***: see `?RMcubic` (package `RandomFields`); valid for $d \leq 3$; twice differentiable

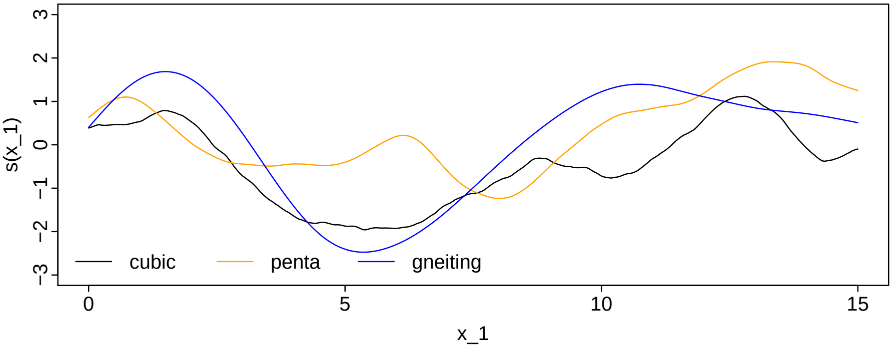
- ***penta covariance***: see `?RMpenta` (package `RandomFields`); valid for $d \leq 3$; 4 times differentiable

- ***Gneiting covariance***: see `?RMgneiting` (package `RandomFields`); valid for $d \leq 3$; 6 times differentiable



1.0
0.8
0.6
0.4
0.2
0.0
co− or semivariance
0
5
10
15
lag h
cubic
penta
Gneiting

```
> RFoptions(spConform=FALSE)
> plot(x1, RFsimulate(RMcubic(var=1, scale=5),
+   x=x1), type="l")
> lines(x1, RFsimulate(RMpenta(var=1, scale=5),
+   x=x1), col=2)
> lines(x1, RFsimulate(RMgneiting(var=1, scale=3.01),
+   x=x1), col=3)
```



summary section 3

- stochastic process: generalisation of multidimensional random variable
- stationarity assumption required for estimation from single realisation of stochastic process
- in practice assumption of weak stationarity:
  1. constant mean
  2. constant variance
  3. covariance and semivariance depends only on lag distance but not on location
- often additional assumption of isotropic auto-correlation
- Gaussian stochastic process: all joint and conditional distributions are normal

summary section 3

- relation between covariance and semi-variance for weakly stationary processes

$$V(\boldsymbol{h}) = \gamma(0) - \gamma(\boldsymbol{h})$$

- variogram generally preferred over covariance function
- shape of variogram close to origin controls smoothness of realisations of Gaussian processes:

1. variogram with nugget: realisations non-continuous
2. variogram grows linearly at origin: realisations continuous but not everywhere differentiable
3. variogram grows at at least quadratically at origin: realisations everywhere at least once differentiable

4 ad-hoc estimation of parameters of model for spatial data

- model for data: $Y_i = S(\boldsymbol{x}_i) + Z_i = \mu(\boldsymbol{x}_i) + E(\boldsymbol{x}_i) + Z_i$ where

Y_i	i^{th} datum
$S(\boldsymbol{x}_i)$	"signal" (= true quantity) at location $\boldsymbol{x}_i$
$\mu(\boldsymbol{x}_i)$	trend
$\{E(\boldsymbol{x}_i)\}$	a zero mean Gaussian process, parametrized by covariance function $\gamma(\boldsymbol{h};\boldsymbol{\theta})$ or variogram $V(\boldsymbol{h};\boldsymbol{\theta})$
Z_i	iid Gaussian measurement error with variance τ^2

- trend $\mu(\boldsymbol{x}_i)$ modelled by linear regression model
$$\mu(\boldsymbol{x}_i) = \sum_k d_k(\boldsymbol{x}_i)\beta_k = \boldsymbol{d}(\boldsymbol{x}_i)^{\mathsf{T}}\boldsymbol{\beta}$$
with $d_k(\boldsymbol{x}_i)$ denoting (spatial) covariates
- unknown elements of model:
  1. structure and parameters $\boldsymbol{\beta}$ of trend model
  2. covariance (or variogram) parameters $\boldsymbol{\theta}$
  3. nugget variance τ^2

exploratory analysis

$\Downarrow$

trend estimation by linear regression analysis

$\Rightarrow$ estimate of trend parameters $\boldsymbol{\beta}$

$\Downarrow$

modelling residual auto-correlation

computing sample variogram of residuals; fitting model function to it

$\Rightarrow$ estimate of variogram parameters $\boldsymbol{\theta}$ and of nugget variance τ^2

$\Downarrow$

(statistical inference)

$\Downarrow$

model assessment by cross-validation

$\Downarrow$

computing spatial predictions

- Gaussian model in vector notation $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{E} + \boldsymbol{Z}$
- estimation of trend parameters β by *ordinary least squares*

$$\widehat{\boldsymbol{\beta}}_{\text{OLS}} = (\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathsf{T}}\boldsymbol{Y}$$

- for spatially uncorrelated data ($\boldsymbol{E} = \boldsymbol{0}$; Cov $\left[\boldsymbol{Y}, \boldsymbol{Y}^{\mathsf{T}}\right] = \tau^2\boldsymbol{I}$)

$$\widehat{\boldsymbol{\beta}}_{\text{OLS}} \sim \mathcal{N}\left(\boldsymbol{\beta}, \tau^2(\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})^{-1}\right)$$

- for spatially auto-correlated data (Cov $\left[\boldsymbol{Y}, \boldsymbol{Y}^{\mathsf{T}}\right] =$ Cov $\left[\boldsymbol{Z}, \boldsymbol{Z}^{\mathsf{T}}\right] +$ Cov $\left[\boldsymbol{E}, \boldsymbol{E}^{\mathsf{T}}\right] = \boldsymbol{\Gamma}_{\boldsymbol{\theta}} = \tau^2\boldsymbol{I} + \boldsymbol{\Sigma}_{\boldsymbol{\theta}}$)

$$\widehat{\boldsymbol{\beta}}_{\text{OLS}} \sim \mathcal{N}\left(\boldsymbol{\beta}, \tau^2(\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})^{-1} + (\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathsf{T}}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\boldsymbol{X}(\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})^{-1}\right)$$

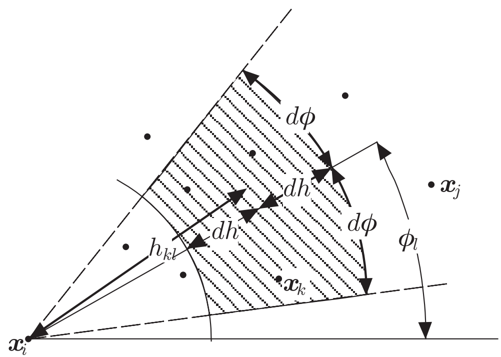
$\Rightarrow$ ignoring auto-correlation: $\widehat{\boldsymbol{\beta}}_{\text{OLS}}$ unbiased, but standard errors too small $\Rightarrow$ tests based on OLS fit biased!

- generalized least squares estimates

$$\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}} = (\boldsymbol{X}^{\mathsf{T}} \boldsymbol{\Gamma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{X})^{-1} \boldsymbol{X}^{\mathsf{T}} \boldsymbol{\Gamma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{Y}$$

- sampling distribution $\quad \widehat{\boldsymbol{\beta}}_{\mathsf{GLS}} \sim \mathcal{N}\left(\boldsymbol{\beta}, (\boldsymbol{X}^{\mathsf{T}} \boldsymbol{\Gamma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{X})^{-1}\right)$
- for spatially uncorrelated data ($\boldsymbol{\Gamma}_{\boldsymbol{\theta}} = \tau^2 \boldsymbol{I}$) $\quad \widehat{\boldsymbol{\beta}}_{\mathsf{GLS}} = \widehat{\boldsymbol{\beta}}_{\mathsf{OLS}}$
- $\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}}$ has among all linear estimators smallest standard errors (Gauss-Markov theorem)
  $\Rightarrow$ *BLUE* (**B**est **L**inear **U**nbiased **E**stimator)
- $\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}}$ is maximum likelihood estimate for Gaussian $\boldsymbol{Y}$

- extract residuals $\boldsymbol{R} = \boldsymbol{Y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}$ of fitted linear model (or use data $\boldsymbol{Y}$ if model has constant $\mu(\boldsymbol{x})$)
- choose bin width dh (and width of angular classes $d\phi$) to define $(k, l)^{\text{th}}$ lag class, $\boldsymbol{h}_{kl}$, characterised by distance, $(h_k - dh, h_k + dh]$ (and angular class, $\phi_l - d\phi, \phi_l + d\phi]$)



computing sample variogram of residuals

- form all N_{kl} pairs (i,j) with $\boldsymbol{x}_i - \boldsymbol{x}_j \approx \boldsymbol{h}_{kl}$ and compute for each lag class $\boldsymbol{h}_{kl}$ the semivariance

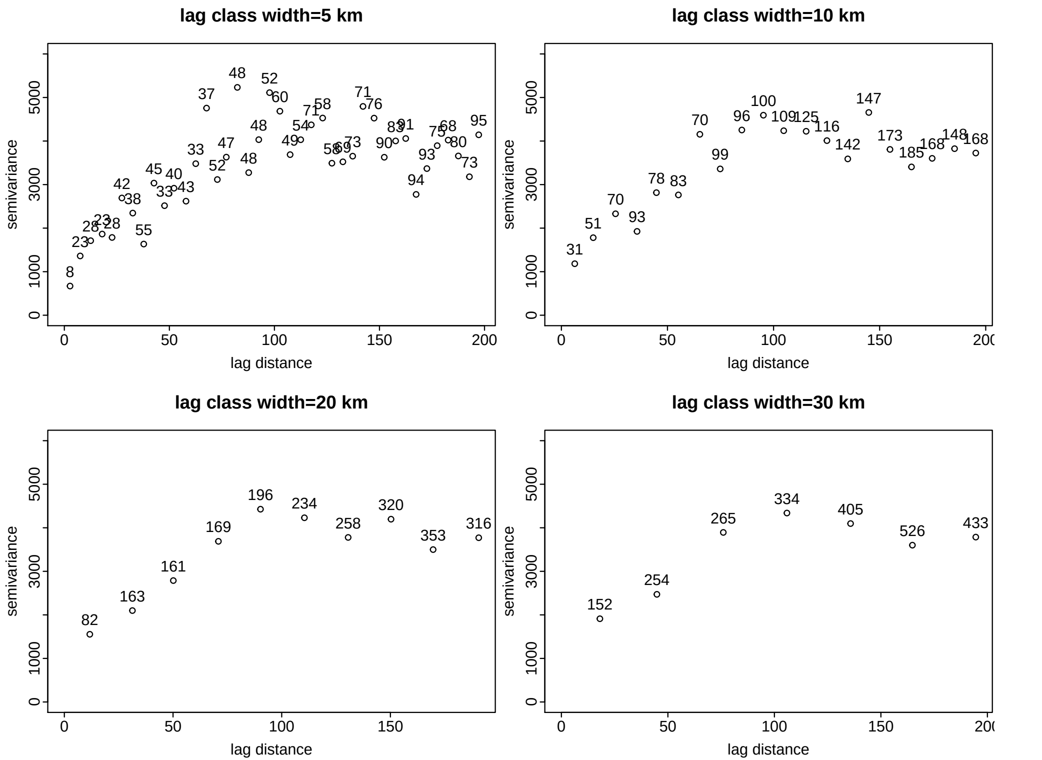
$$\widehat{V}(\boldsymbol{h}_{kl}) = \frac{1}{2\,N_{kl}} \sum_{(i,j)\in \boldsymbol{h}_{kl}} [R(\boldsymbol{x}_i) - R(\boldsymbol{x}_j)]^2$$

- sample variogram   plot of $\widehat{V}(\boldsymbol{h}_{kl})$ vs. $\boldsymbol{h}_{kl}$

- rules of thumb:

  1. choose dh (and $d\phi$) such that $N_{kl} > 30 - 50$
  2. largest $\boldsymbol{h}_{kl} \leq 0.5\,\text{max}(\boldsymbol{x}_i - \boldsymbol{x}_j)$

```
> library(georob)
> r.sv.5 <- sample.variogram(residuals(r.ols),
+   locations=coordinates(d.w), lag.dist.def=5,
+   max.lag=200, estimator="matheron")
> plot(r.sv.5, main="lag class width=5 km")
> text(gamma~lag.dist, r.sv.5, labels=npairs, pos=3)
> ...
```


lag class width=5 km
lag class width=10 km
lag class width=20 km
lag class width=30 km
semivariance
0
1000
3000
5000
lag distance
0
50
100
150
200

fitting variogram model to sample variogram

- semivariance required for arbitrary lag distances when computing predictions

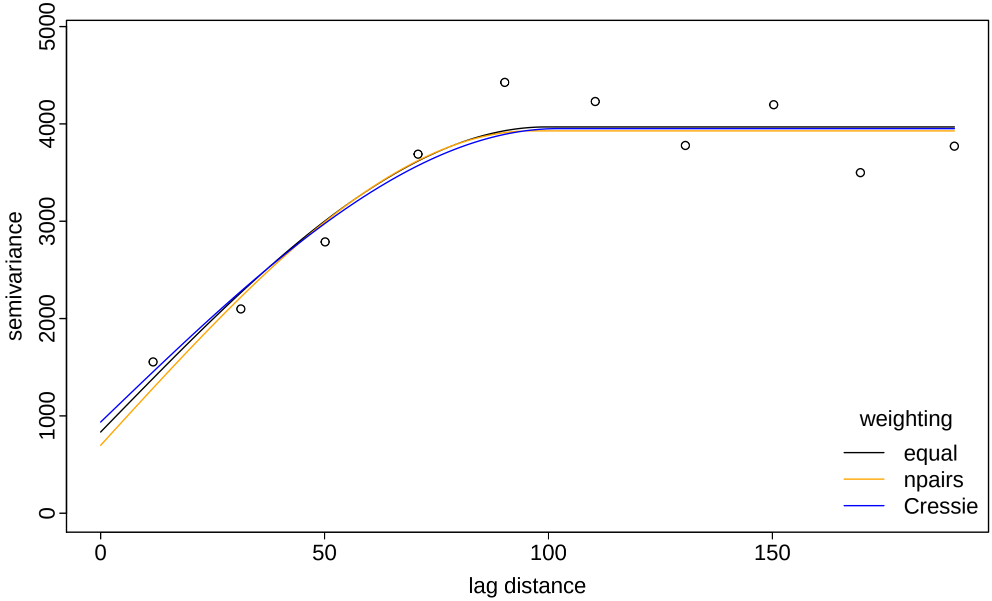
⇒ smoothing sample variogram by fitting a parametric variogram function $V(\boldsymbol{h}, \boldsymbol{\theta})$

- choose a variogram function that approximates shape of sample variogram well (in particular close to origin)
- fit parameters $\boldsymbol{\theta}$ by (weighted) non-linear least squares

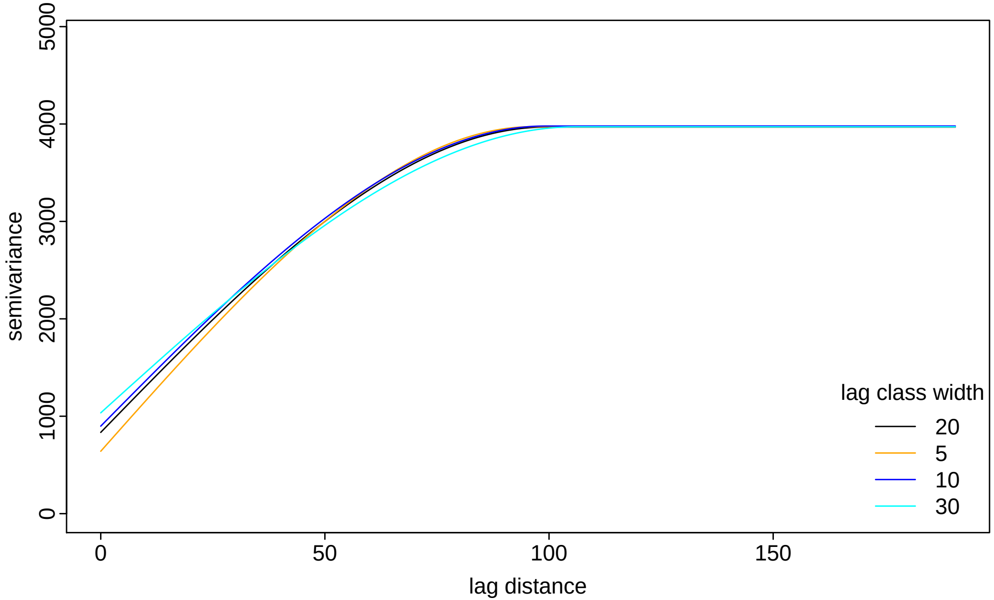
$$\widehat{\boldsymbol{\theta}} = \text{argmin}_{\boldsymbol{\theta}} \sum_{kl} w(\boldsymbol{h}_{kl}) \left(\widehat{V}(\boldsymbol{h}_{kl}) - V(\boldsymbol{h}_{kl}, \boldsymbol{\theta}) \right)^2$$

- options for weighing
  1. equal weights: $w(\boldsymbol{h}_{kl}) = 1$
  2. by number of pairs: $w(\boldsymbol{h}_{kl}) = N_{kl}$
  3. Cressie's weights: $w(\boldsymbol{h}_{kl}) = N_{kl}/V(\boldsymbol{h}_{kl}, \boldsymbol{\theta})^2$

```
> r.sph.e <- fit.variogram.model(r.sv.20,
+   variogram.model="RMspheric",
+   param=c(variance=3000, nugget=100, scale=100),
+   weighting="equal")
> plot(r.sv.20)
> lines(r.sph.e)
> ...
```



semivariance
lag distance
0
1000
2000
3000
4000
5000
0
50
100
150
weighting
equal
npairs
Cressie



semivariance
0
1000
2000
3000
4000
5000
0
50
100
150
lag distance
lag class width
20
5
10
30

- subjective choice of lag class width and weighting method for model fitting
- estimates of semivariance for different lag classes mutually correlated; choice of variogram function based on sample variogram problematic
- auto-correlation of residuals

$$\widehat{\boldsymbol{R}} = \boldsymbol{Y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}_{\mathsf{OLS}} = \boldsymbol{Y} - \underbrace{\boldsymbol{X}(\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathsf{T}}}_{\boldsymbol{H}}\boldsymbol{Y} = (\boldsymbol{I} - \boldsymbol{H})\boldsymbol{Y}$$

differs from auto-correlation of underlying stochastic process

$$\mathsf{Cov}\left[\widehat{\boldsymbol{R}}, \widehat{\boldsymbol{R}}^{\mathsf{T}}\right] = (\boldsymbol{I}-\boldsymbol{H})\mathsf{Cov}\left[\boldsymbol{Y}, \boldsymbol{Y}^{\mathsf{T}}\right](\boldsymbol{I}-\boldsymbol{H}) = (\boldsymbol{I}-\boldsymbol{H})\Gamma_{\boldsymbol{\theta}}(\boldsymbol{I}-\boldsymbol{H}) \neq \Gamma_{\boldsymbol{\theta}}$$

⇒ estimate of variogram based on sample variogram of OLS residuals biased

⇒ estimate trend and variogram parameters simultaneously by maximum likelihood

- generalised least squares (GLS) method of choice for estimating coefficients of trend model (BLUE)
- GLS requires variogram
- subjective choice of lag class definition for computing sample variogram
- sample variogram susceptible to outliers $\Rightarrow$ robust estimators
- fitting model function to sample variogram requires further subjective choices
- ad-hoc approach provides biased estimates of variogram of underlying stochastic process if trend is modelled

5 maximum likelihood (ML) estimation of parameters of Gaussian model for spatial data

- principle of ML estimation: find parameters that maximize joint probability for observed data
- properties of ML estimates: asymptotically unbiased and fully efficient; asymptotically normally distributed
- bias matters for estimating variance parameters from small samples
- profile likelihood useful for exploring shape of likelihood surface and for computing confidence intervals based on likelihood ratio test

- consider now a Gaussian stochastic process $\{Y(\boldsymbol{x})\}$ with a linear trend function
- any arbitrary set of random variables $\boldsymbol{Y} = (Y(\boldsymbol{x}_1), \ldots, Y(\boldsymbol{x}_n))$ has a multivariate Gaussian distribution with expectation

$$\mathsf{E}\left[\boldsymbol{Y}\right] = \boldsymbol{X}\boldsymbol{\beta}$$

  and covariance matrix

$$\mathsf{Cov}\left[\boldsymbol{Y}, \boldsymbol{Y}^{\mathsf{T}}\right] = \boldsymbol{\Gamma}_{\boldsymbol{\theta}}$$

- joint probability density for $\boldsymbol{Y}$ given by

$$f(\boldsymbol{y}; \boldsymbol{\beta}, \boldsymbol{\theta}) = (2\pi)^{-\frac{n}{2}} |\boldsymbol{\Gamma}_{\boldsymbol{\theta}}|^{-\frac{1}{2}} \exp(-\frac{1}{2}\{\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\}^{\mathsf{T}} \boldsymbol{\Gamma}_{\boldsymbol{\theta}}^{-1} \{\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\})$$

- unknown model parameters:
  1. regression coefficients $\boldsymbol{\beta}$
  2. covariance (or variogram) parameters $\boldsymbol{\theta}$
- log-likelihood function (up to a constant) given by

$$L(\boldsymbol{\beta}, \boldsymbol{\theta}; \boldsymbol{y}) = -\frac{1}{2}\log(|\boldsymbol{\Gamma}_{\boldsymbol{\theta}}|) - \frac{1}{2}\{\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\}^{\mathsf{T}}\boldsymbol{\Gamma}_{\boldsymbol{\theta}}^{-1}\{\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\}$$

- for known $\boldsymbol{\theta}$ MLE for $\boldsymbol{\beta}$ equal to GLS estimator

$$\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}} = (\boldsymbol{X}^{\mathsf{T}}\boldsymbol{\Gamma}_{\boldsymbol{\theta}}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathsf{T}}\boldsymbol{\Gamma}_{\boldsymbol{\theta}}^{-1}\boldsymbol{Y}$$

- plugging $\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}}$ for $\boldsymbol{\beta}$ into $L(\boldsymbol{\beta}, \boldsymbol{\theta}; \boldsymbol{y})$ gives profile likelihood function for $\boldsymbol{\theta}$

$$L_{\mathsf{p}}(\boldsymbol{\theta}; \boldsymbol{y}) = -\frac{1}{2}\log(|\boldsymbol{\Gamma}_{\boldsymbol{\theta}}|) - \frac{1}{2}\{\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}}\}^{\mathsf{T}}\boldsymbol{\Gamma}_{\boldsymbol{\theta}}^{-1}\{\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}}\}$$

ML estimation for Gaussian spatial model

- set of estimating equations

$$\frac{\partial L_{\mathsf{p}}}{\partial \boldsymbol{\theta}} = \mathbf{0}$$

  forms a system of non-linear equations $\Rightarrow$ in general difficult to solve

$\Rightarrow$ maximize $L_{\mathsf{p}}(\boldsymbol{\theta}; \boldsymbol{y})$ numerically by a non-linear optimization method to find MLE $\widehat{\boldsymbol{\theta}}$

$\Rightarrow$ numerical optimization requires initial values of $\boldsymbol{\theta}$

example: ML estimates Wolfcamp data

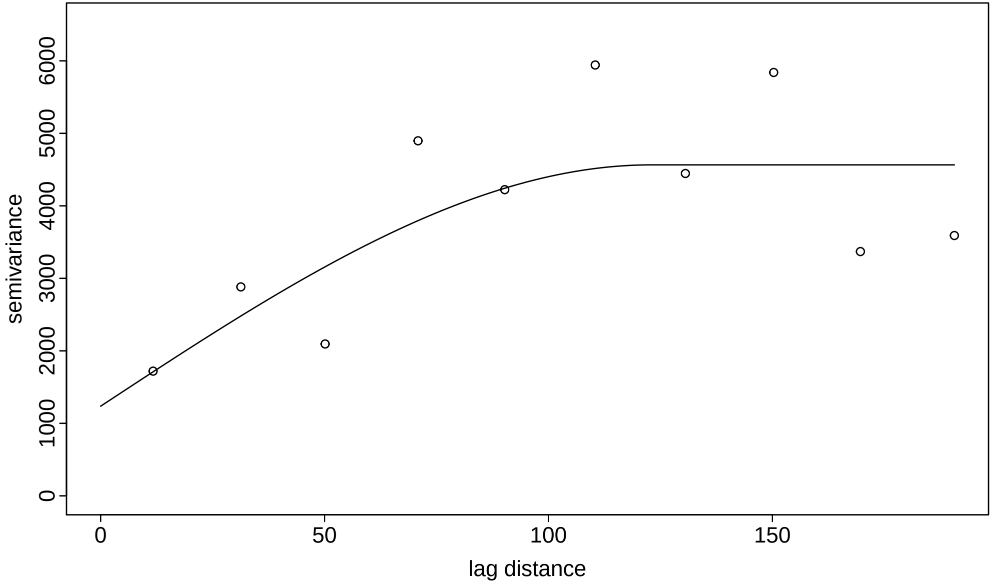

```
> library(gstat)
> library(georob)
> d.w <- wolfcamp
> coordinates(d.w) <- c("x", "y")
> r.georob.ml <- georob(pressure~x+y, d.w,
+   locations=~x+y, variogram.model="RMspheric",
+   param=c(variance=3000, nugget=1000, scale=100),
+   tuning.psi=1000, control=control.georob(ml.method="ML"))
> summary(r.georob.ml)
```

```
Call:georob(formula = pressure ~ x + y, data = d.w, locations = ~x +
    y, variogram.model = "RMspheric", param = c(variance = 3000,
    nugget = 1000, scale = 100), tuning.psi = 1000, control = control.georob

Tuning constant:  1000

Convergence in 12 function and 7 Jacobian/gradient evaluations

Estimating equations (gradient)
```

```
                                      eta          scale
  Gradient            : -8.880367e-05  7.654524e-04

Maximized log-likelihood: -458.3671

Predicted latent variable (B):
   Min     1Q Median     3Q    Max
-89.37 -54.98 -16.64  24.43 111.41

Residuals (epsilon):
    Min      1Q  Median      3Q     Max
-62.789 -19.775   6.311  18.032  62.776

Standardized residuals:
    Min      1Q  Median      3Q     Max
-2.3778 -0.7413  0.2249  0.6534  3.2793


Gaussian ML estimates

Variogram:  RMspheric
               Estimate   Lower  Upper
variance        3328.90 1453.95 7621.7
snugget(fixed)     0.00      NA     NA
nugget          1236.27  616.01 2481.1
scale            122.95   95.13  158.9
```

```
Fixed effects coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 620.3550    17.0641  36.354  < 2e-16
x            -1.3256     0.1360  -9.750 2.33e-15
y            -1.2061     0.1793  -6.727 2.16e-09

Residual standard error (sqrt(nugget)): 35.16

Robustness weights:
 All 85 weights are ~= 1.
```

```
> plot(r.georob.ml, lag.dist.def=20, max.lag=200)
```

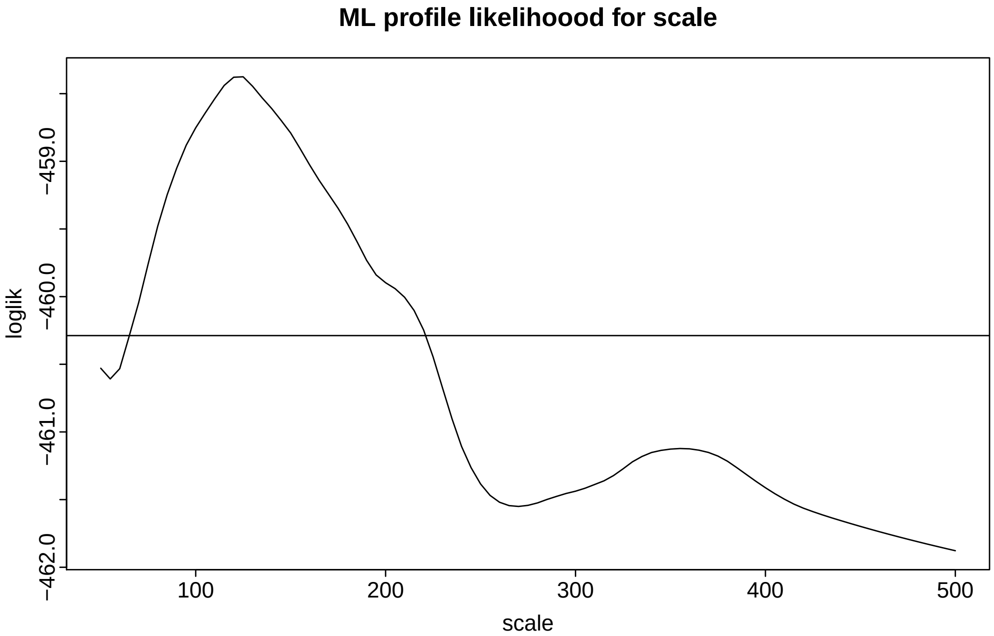


computing profile log-likelihood for range parameter along with 95%-confidence interval

```
> r.proflik.ml <- profilelogLik(r.georob.ml,
+  values=data.frame(scale=seq(50, 500, by=5)))
> str(r.proflik.ml)
```

```
'data.frame':        91 obs. of  9 variables:
 $ scale          : num  50 55 60 65 70 75 80 85 90 95 ...
 $ loglik         : num  -461 -461 -461 -460 -460 ...
 $ variance       : num  3641 3508 3393 3315 3240 ...
 $ nugget         : num  535 668 780 860 933 ...
 $ (Intercept)    : num  617 618 618 618 618 ...
 $ x              : num  -1.29 -1.29 -1.3 -1.3 -1.31 ...
 $ y              : num  -1.24 -1.25 -1.25 -1.25 -1.26 ...
 $ gradient.nugget: num  0.018148 -0.006713 -0.00423 -0.00..
 $ converged      : num  1 1 1 1 1 1 1 1 1 1 ...
```

```
> plot(loglik~scale, r.proflik.ml, type="l",
+   main="ML profile likelihoood for scale")
> abline(v=r.georob.ml$param["scale"])
> abline(h=r.georob.ml$loglik - qchisq(0.95, 1)/2)
```


ML profile likelihoood for scale
loglik
−459.0
−460.0
−461.0
−462.0
100
200
300
400
500
scale

- for small sample size MLEs of variogram parameters often negatively biased when trend is simultaneously estimated
- for auto-correlated data this problem is more severe because *effective sample size* usually much smaller than nominal sample size n
- effective sample size given by *equivalent number of independent observations*

$$n_{\text{eq}} = \frac{\text{Var}\left[Y(\boldsymbol{x})\right]}{\text{Var}\left[\bar{Y}\right]} \leq n$$

where

$$\text{Var}\left[\bar{Y}\right] = \frac{\text{Var}\left[Y(\boldsymbol{x})\right]}{n} + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1;j\neq i}^{n}\text{Cov}\left[Y(\boldsymbol{x}_i), Y(\boldsymbol{x}_j)\right]$$

```
> library(RandomFields)
> Gamma <- RFcovmatrix(
+   RMspheric(var=3329, scale=123)+RMnugget(var=1236),
+   x=coordinates(d.w))
> var.y <- sum(c(variance=3329, nugget=1236))
> (n <- nrow(d.w))
```

```
[1] 85
```

```
> var.ybar <- sum(Gamma)/n^2
> var.y/var.ybar
```

```
[1] 13.67365
```

- bias of MLEs of variogram parameters $\boldsymbol{\theta}$ can be reduced by *restricted maximum likelihood estimation*

- principle of restricted maximum likelihood estimation (REML)

  1. form linear combinations $\boldsymbol{Z} = \boldsymbol{A}\boldsymbol{Y}$ of data $\boldsymbol{Y}$ that have zero expectation (and do no longer depend on β)

$$\mathsf{E}\left[\boldsymbol{Z}\right] = \boldsymbol{A}\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{0}$$

$\Rightarrow$ matrix $\boldsymbol{A}$ must satisfy condition $\quad \boldsymbol{A}\boldsymbol{X} = \boldsymbol{0}$

$\Rightarrow$ $\boldsymbol{A}$ non-unique; many possibility, e.g.

$$\boldsymbol{A} = \boldsymbol{I} - \boldsymbol{H}_{\mathsf{OLS}} = \boldsymbol{I} - \boldsymbol{X}(\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathsf{T}}$$

$\Rightarrow$ $\boldsymbol{Z}$ is an error contrast or a generalized increment

- principle of REML (continued)

  2. estimate $\boldsymbol{\theta}$ by maximizing likelihood function for $n - p$ elements of $\boldsymbol{Z}$

  ⇒ this is equivalent to maximizing the restricted log-likelihood function

$$
\begin{aligned}
L_{\mathsf{r}}(\boldsymbol{\theta}; y) \;&=\; -\frac{1}{2}\,\mathsf{log}(|\boldsymbol{\Gamma}_{\boldsymbol{\theta}}|) - \frac{1}{2}\,\mathsf{log}(|\boldsymbol{X}^{\mathsf{T}}\boldsymbol{\Gamma}_{\boldsymbol{\theta}}^{-1}\boldsymbol{X}|) \\
&\quad -\frac{1}{2}\{\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}}\}^{\mathsf{T}}\boldsymbol{\Gamma}_{\boldsymbol{\theta}}^{-1}\{\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}}\}
\end{aligned}
$$

  ⇒ REML estimate $\widehat{\boldsymbol{\theta}}_{\mathsf{REML}}$ has same properties (asymptotic normal distribution, likelihood ratio statistic) as ML estimate

  3. given $\widehat{\boldsymbol{\theta}}_{\mathsf{REML}}$ compute $\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}}$ and $\mathsf{Cov}\left[\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}}, \widehat{\boldsymbol{\beta}}_{\mathsf{GLS}}^{\mathsf{T}}\right] = (\boldsymbol{X}^{\mathsf{T}}\boldsymbol{\Gamma}_{\widehat{\boldsymbol{\theta}}_{\mathsf{REML}}}^{-1}\boldsymbol{X})^{-1}$

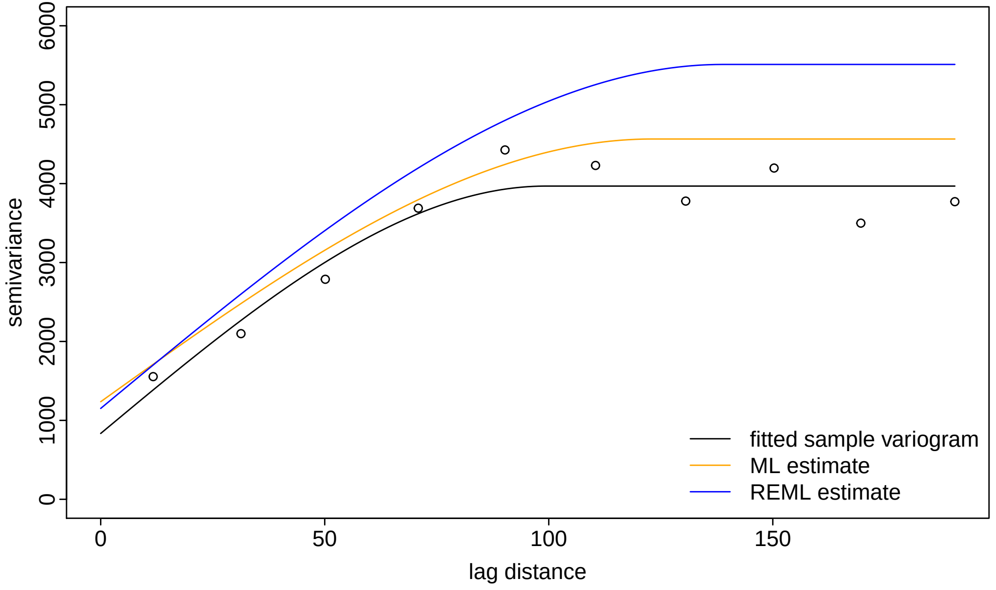
example: REML estimates Wolfcamp data

```
> r.georob.reml <- georob(pressure~x+y, d.w,
+   locations=~x+y, variogram.model="RMspheric",
+   param=c(variance=3000, nugget=1000, scale=100),
+   tuning.psi=1000)
> summary(r.georob.reml)
> plot(r.sv.20, ylim=c(0, 6000))
> lines(r.sph.e)
> lines(r.georob.ml, col=2)
> lines(r.georob.reml, col=3)
> legend("bottomright", lty=1, col=1:3,
+   legend=c("fitted sample variogram", "ML estimate",
+   "REML estimate"), bty="n")
```

```
Call:georob(formula = pressure ~ x + y, data = d.w, locations = ~x +
    y, variogram.model = "RMspheric", param = c(variance = 3000,
    nugget = 1000, scale = 100), tuning.psi = 1000)

Tuning constant:  1000
```

```
Convergence in 6 function and 5 Jacobian/gradient evaluations

Estimating equations (gradient)

                                        eta          scale
  Gradient           : -2.248651e-04 -1.070402e-01

Maximized restricted log-likelihood: -456.3802

Predicted latent variable (B):
   Min     1Q Median     3Q    Max
-94.58 -60.99 -17.59  23.10 115.72

Residuals (epsilon):
    Min      1Q  Median      3Q     Max
-59.148 -18.009   6.251  15.982  54.620

Standardized residuals:
    Min      1Q  Median      3Q     Max
-2.4030 -0.7131  0.2282  0.6937  3.1932


Gaussian REML estimates

Variogram:  RMspheric
               Estimate
variance         4358.8
```

```
snugget(fixed)       0.0
nugget            1151.3
scale              138.9


Fixed effects coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 624.3287    20.7961  30.021  < 2e-16
x            -1.3291     0.1611  -8.248 2.25e-12
y            -1.1804     0.2115  -5.581 3.00e-07

Residual standard error (sqrt(nugget)): 33.93

Robustness weights:
 All 85 weights are ~= 1.
```



semivariance
0
1000
2000
3000
4000
5000
6000
0
50
100
150
lag distance
fitted sample variogram
ML estimate
REML estimate

- likelihood ratio test can only be used to test hypotheses and build confidence regions for $\boldsymbol{\theta}$
- LRT for regression for $\boldsymbol{\beta}$ in general biased, (too small p-values Pinheiro and Bates, 2000, pp. 87)

$\Rightarrow$ use conditional F-tests for testing hypotheses about $\boldsymbol{\beta}$:

1. fit covariance parameters of “largest” regression model
   $\Rightarrow \widehat{\boldsymbol{\theta}}$
2. compute covariance matrix $\Rightarrow \boldsymbol{\Gamma}_{\widehat{\boldsymbol{\theta}}}$
3. compute $\Rightarrow \boldsymbol{L}_{\widehat{\boldsymbol{\theta}}}$ by Cholesky decomposition of $\boldsymbol{\Gamma}_{\widehat{\boldsymbol{\theta}}}$
4. orthogonalize response vector and design matrix
   $\Rightarrow \tilde{\boldsymbol{Y}} = \boldsymbol{L}_{\widehat{\boldsymbol{\theta}}}^{-1}\boldsymbol{Y}$, $\tilde{\boldsymbol{X}} = \boldsymbol{L}_{\widehat{\boldsymbol{\theta}}}^{-1}\boldsymbol{X}$
5. conventional F-test with orthogonalized items $\tilde{\boldsymbol{Y}}$ and $\tilde{\boldsymbol{X}}$

example: hypothesis tests trend Wolfcamp data

```
> d.w$xs <- d.w$x - mean(d.w$x)
> d.w$ys <- d.w$y - mean(d.w$y)
> r.georob.full <- georob(
+   pressure~xs+ys+I(xs^2)+I(ys^2)+xs:ys, d.w,
+   locations=~x+y, variogram.model="RMspheric",
+   param=c(variance=3000, nugget=1000, scale=100),
+   tuning.psi=1000)
> summary(r.georob.full)
```

```
Call:georob(formula = pressure ~ xs + ys + I(xs^2) + I(ys^2) + xs:ys,
    data = d.w, locations = ~x + y, variogram.model = "RMspheric",
    param = c(variance = 3000, nugget = 1000, scale = 100), tuning.psi = 100

Tuning constant:  1000

Convergence in 10 function and 8 Jacobian/gradient evaluations

Estimating equations (gradient)

                                 eta         scale
```

```
  Gradient           :  3.590344e-04 -4.553394e-03

Maximized restricted log-likelihood: -470.3894

Predicted latent variable (B):
   Min     1Q Median     3Q    Max
-89.22 -46.81 -11.06  20.80  94.07

Residuals (epsilon):
    Min      1Q  Median      3Q     Max
-59.664 -18.086   6.783  16.245  49.986

Standardized residuals:
    Min      1Q  Median      3Q     Max
-2.4096 -0.7213  0.2500  0.6830  3.1042


Gaussian REML estimates

Variogram:  RMspheric
               Estimate   Lower  Upper
variance        3740.02 1469.60 9518.1
snugget(fixed)     0.00      NA     NA
nugget          1152.79  532.29 2496.6
scale            123.93   93.96  163.5
```

```
Fixed effects coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 613.105482  30.364674  20.191  < 2e-16
xs           -1.161184   0.167717  -6.923 1.05e-09
ys           -1.040147   0.214430  -4.851 6.06e-06
I(xs^2)       0.001592   0.001230   1.294    0.199
I(ys^2)      -0.001731   0.002133  -0.811    0.420
xs:ys         0.002067   0.001968   1.050    0.297

Residual standard error (sqrt(nugget)): 33.95

Robustness weights:
 All 85 weights are ~= 1.
```

```
> waldtest(r.georob.full, .~.-xs:ys, test="F")
```

```
Wald test

Model 1: pressure ~ xs + ys + I(xs^2) + I(ys^2) + xs:ys
Model 2: pressure ~ xs + ys + I(xs^2) + I(ys^2)
  Res.Df Df      F Pr(>F)
1     79
2     80 -1 1.1032 0.2968
```

```
> waldtest(r.georob.full, .~.-I(xs^2)-I(ys^2)-xs:ys,
+   test="F")
```

```
Wald test

Model 1: pressure ~ xs + ys + I(xs^2) + I(ys^2) + xs:ys
Model 2: pressure ~ xs + ys
  Res.Df Df      F Pr(>F)
1     79
2     82 -3 1.6284 0.1895
```

- given estimates of covariance parameters $\widehat{\theta}$ and keeping them fixed, the usual stepwise procedures for selecting covariates can be used

- selecting models based on AIC and BIC

```
> # model building based on AIC
> step(r.georob.full)

Start:  AIC=922.16
pressure ~ xs + ys + I(xs^2) + I(ys^2) + xs:ys

          Df    AIC Converged
- I(xs^2)  1 922.05         1
- I(ys^2)  1 922.13         1
<none>       922.16
- xs:ys    1 922.49         1

Step:  AIC=922.05
pressure ~ xs + ys + I(ys^2) + xs:ys
```

```
          Df    AIC Converged
<none>       922.05
+ I(xs^2)  1 922.16         1
- I(ys^2)  1 922.54         1
- xs:ys    1 924.61         1

Tuning constant:  1000

Fixed effects coefficients:
(Intercept)           xs           ys     I(ys^2)
 634.939862    -1.248212    -1.092032    -0.002587
      xs:ys
   0.003005

Variogram:  RMspheric
 variance(fixed)    snugget(fixed)     nugget(fixed)
          2060.4               0.0            1402.1
    scale(fixed)
           103.8
```

```
> # model building based on BIC
> step(r.georob.full, k=log(nrow(d.w)))
```

```
Start:  AIC=936.81
pressure ~ xs + ys + I(xs^2) + I(ys^2) + xs:ys

          Df    AIC Converged
- I(xs^2)  1 934.27         1
- I(ys^2)  1 934.34         1
- xs:ys    1 934.70         1
<none>       936.81

Step:  AIC=934.27
pressure ~ xs + ys + I(ys^2) + xs:ys

          Df    AIC Converged
- I(ys^2)  1 932.31         1
<none>       934.27
- xs:ys    1 934.38         1
+ I(xs^2)  1 936.81         1

Step:  AIC=932.31
pressure ~ xs + ys + xs:ys

          Df    AIC Converged
- xs:ys    1 931.79         1
<none>       932.31
+ I(ys^2)  1 934.27         1
+ I(xs^2)  1 934.34         1
```

```
Step:  AIC=931.79
pressure ~ xs + ys

          Df     AIC Converged
+ I(xs^2)  1  931.70         1
<none>        931.79
+ xs:ys    1  932.31         1
+ I(ys^2)  1  934.38         1
- ys       1 1006.98         1
- xs       1 1085.42         1

Step:  AIC=931.7
pressure ~ xs + ys + I(xs^2)

          Df     AIC Converged
<none>        931.70
- I(xs^2)  1  931.79         1
+ xs:ys    1  934.34         1
+ I(ys^2)  1  934.70         1
- ys       1  991.37         1
- xs       1 1021.51         1

Tuning constant:  1000

Fixed effects coefficients:
(Intercept)          xs          ys     I(xs^2)
 596.183822   -1.181578   -1.143500    0.001875
```

```
Variogram:  RMspheric
 variance(fixed)    snugget(fixed)     nugget(fixed)
          2060.4                0.0           1402.1
    scale(fixed)
           103.8
```

- no closed form expressions for ML estimates of parameters of Gaussian model for spatial data; estimates obtained by numerically maximizing likelihood function

- equivalent number of independent observations of a sample of spatial data often much smaller than nominal sample size: $\Rightarrow$ bias of ML estimates of variance parameters important

$\Rightarrow$ restricted maximum likelihood estimation (REML) method of choice

- use of conditional F-tests for testing hypotheses about trend function coefficients

- use of standard stepwise model building procedures for finding structure of trend function

References

Pinheiro, J. C. and Bates, D. M. (2000). *Mixed-Effects Models in S and S-PLUS*. Springer Verlag.

6 spatial prediction by kriging

- observations $\boldsymbol{y}^{\mathsf{T}} = (y_1, \ldots, y_n)$ available for a set of n locations $\boldsymbol{x}_i$
- $\boldsymbol{y}$ considered as realisation of the multivariate random variable $\boldsymbol{Y}^{\mathsf{T}} = (Y_1, \ldots, Y_n)$
- model: $Y_i = S(\boldsymbol{x}_i) + Z_i$ with

Y_i	i^{th} datum
$S(\boldsymbol{x}_i)$	"signal" (= true quantity) at location $\boldsymbol{x}_i$
$\{S(\boldsymbol{x}_i)\}$	Gaussian process, parametrized by trend $\mu(\boldsymbol{x}_i) = \sum_k d_k(\boldsymbol{x}_i)\beta_k = \boldsymbol{d}(\boldsymbol{x}_i)^{\mathsf{T}}\boldsymbol{\beta}$ and covariance function $\gamma(\boldsymbol{h};\boldsymbol{\theta})$ or variogram $V(\boldsymbol{h};\boldsymbol{\theta})$
Z_i	iid Gaussian measurement error with variance τ^2

- predictions, say $\widehat{\boldsymbol{S}}$, of $\boldsymbol{S}^{\mathsf{T}} = (S(\boldsymbol{x}_1'), \ldots, S(\boldsymbol{x}_m'))$ required for set of m location $\boldsymbol{x}_j'$ without data; $\widehat{\boldsymbol{S}}$ computed from $\boldsymbol{Y}$ $\Rightarrow \widehat{\boldsymbol{S}} = \widehat{\boldsymbol{S}}(\boldsymbol{Y})$

- consider for simplicity case $m = 1$, i.e.

$$S = S(\boldsymbol{x}_1') \text{ and } \widehat{S} = \widehat{S}(\boldsymbol{x}_1'; \boldsymbol{Y})$$

- criterion for optimality of prediction
  ⇒ *mean squared prediction error*

$$\text{MSEP}[\widehat{S}] = \text{E}\left[\{\widehat{S} - S\}^2\right]$$

(expectation taken with respect to joint distribution of $\boldsymbol{Y}$ and S)

mean square prediction

- MSEP can alternatively be written as (e.g. Diggle and Ribeiro, 2007, p. 135)

$$
\begin{aligned}
\mathsf{MSEP}[\widehat{S}] &= \mathsf{E}_{\boldsymbol{Y}}\left[\mathsf{E}_{S|\boldsymbol{Y}}[\{\widehat{S}-S\}^2]\right] = \ldots \\
&= \mathsf{E}_{\boldsymbol{Y}}\left[\mathsf{Var}_{S|\boldsymbol{Y}}[S]\right] + \mathsf{E}_{\boldsymbol{Y}}\left[\left\{\mathsf{E}_{S|\boldsymbol{Y}}[S]-\widehat{S}\right\}^2\right]
\end{aligned}
$$

$\Rightarrow$ *conditional expectation* $\widehat{S}_{\text{opt}} = \mathsf{E}_{S|\boldsymbol{Y}}[S]$ minimises MSEP

- MSEP of $\widehat{S}_{\text{opt}}$ equal to expectation of conditional variance

$$
\mathsf{MSEP}[\widehat{S}_{\text{opt}}] = \mathsf{E}_{\boldsymbol{Y}}\left[\mathsf{Var}_{S|\boldsymbol{Y}}[S]\right]
$$

- evaluation of $\widehat{S}_{\text{opt}}$ and $\mathsf{MSEP}[\widehat{S}_{\text{opt}}]$ requires fully specified parametric model for joint distribution of S and $\boldsymbol{Y}$

- standard results from theory about multivariate normal distributions apply

- joint distribution of $(\boldsymbol{S}^\mathsf{T}, \boldsymbol{Y}^\mathsf{T})$ multivariate normal with mean vector

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_{\boldsymbol{S}} \\ \boldsymbol{\mu}_{\boldsymbol{Y}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}_{\boldsymbol{S}}\boldsymbol{\beta} \\ \boldsymbol{X}_{\boldsymbol{Y}}\boldsymbol{\beta} \end{bmatrix}$$

  and covariance matrix

$$\boldsymbol{\Sigma} = \begin{bmatrix} \mathrm{Cov}\left[\boldsymbol{S}, \boldsymbol{S}^\mathsf{T}\right] & \mathrm{Cov}\left[\boldsymbol{S}, \boldsymbol{Y}^\mathsf{T}\right] \\ \mathrm{Cov}\left[\boldsymbol{Y}, \boldsymbol{S}^\mathsf{T}\right] & \mathrm{Cov}\left[\boldsymbol{Y}, \boldsymbol{Y}^\mathsf{T}\right] \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Sigma}_{\boldsymbol{SS}} & \boldsymbol{\Sigma}_{\boldsymbol{SY}} \\ \boldsymbol{\Sigma}_{\boldsymbol{SY}}^\mathsf{T} & \boldsymbol{\Gamma}_{\boldsymbol{YY}} \end{bmatrix}$$

- note that

  1. $\boldsymbol{\Sigma}$ depends on covariance parameters $\boldsymbol{\theta}, \tau^2$ and
  2. again $\boldsymbol{\Gamma}_{\boldsymbol{YY}} = \boldsymbol{\Sigma}_{\boldsymbol{YY}} + \tau^2\boldsymbol{I}$ where $\boldsymbol{\Sigma}_{\boldsymbol{YY}}$ denotes here the covariance matrix of the signal at the *data locations*

- conditional distribution of $\boldsymbol{S}$ given $\boldsymbol{Y} = \boldsymbol{y}$ normal with mean

$$\mathsf{E}_{\boldsymbol{S}|\boldsymbol{Y}}[\boldsymbol{S}] = \boldsymbol{X}_{\boldsymbol{S}}\boldsymbol{\beta} + \boldsymbol{\Sigma}_{\boldsymbol{S}\boldsymbol{Y}}\boldsymbol{\Gamma}_{\boldsymbol{Y}\boldsymbol{Y}}^{-1}(\boldsymbol{y} - \boldsymbol{X}_{\boldsymbol{Y}}\boldsymbol{\beta})$$

  and covariance matrix

$$\mathsf{Cov}_{\boldsymbol{S}|\boldsymbol{Y}}[\boldsymbol{S}, \boldsymbol{S}^{\mathsf{T}}] = \boldsymbol{\Sigma}_{\boldsymbol{S}\boldsymbol{S}} - \boldsymbol{\Sigma}_{\boldsymbol{S}\boldsymbol{Y}}\boldsymbol{\Gamma}_{\boldsymbol{Y}\boldsymbol{Y}}^{-1}\boldsymbol{\Sigma}_{\boldsymbol{S}\boldsymbol{Y}}^{\mathsf{T}}$$

  (conditional covariance matrix independent of $\boldsymbol{y}$)

- *simple kriging predictor*(= optimal predictor) equal to conditional expectation

$$\widehat{\boldsymbol{S}}_{\mathsf{opt}} = \mathsf{E}_{\boldsymbol{S}|\boldsymbol{Y}}[\boldsymbol{S}]$$

- MSEP of simple kriging predictor equal to

$$\mathsf{MSEP}[\widehat{\boldsymbol{S}}_{\mathsf{opt}}] = \mathsf{Cov}_{\boldsymbol{S}|\boldsymbol{Y}}[\boldsymbol{S}, \boldsymbol{S}^{\mathsf{T}}]$$

- write $\mathbf{\Lambda} = \mathbf{\Sigma}_{\boldsymbol{SY}}\mathbf{\Gamma}_{\boldsymbol{YY}}^{-1}$ ($m \times n$-matrix) and $\boldsymbol{\lambda} = (\boldsymbol{X}_{\boldsymbol{S}} - \mathbf{\Lambda}\boldsymbol{X}_{\boldsymbol{Y}})\boldsymbol{\beta}$ (m-vector) then

$$\widehat{\boldsymbol{S}}_{\text{opt}} = \boldsymbol{\lambda} + \mathbf{\Lambda}\boldsymbol{y}$$

⇒ simple kriging predictor is a heterogeneous linear predictor ($\mathbf{\Lambda}$: matrix with simple kriging weights; $\boldsymbol{\lambda}$: vector with “intercepts”)

- one may show that $\mathbf{\Lambda}$ and $\boldsymbol{\lambda}$ minimize for any heterogeneous linear predictor

$$\text{trace}\left(\mathsf{E}\left[\{\widehat{\boldsymbol{S}} - \boldsymbol{S}\}\{\widehat{\boldsymbol{S}} - \boldsymbol{S}\}^{\mathsf{T}}\right]\right)$$

subject to constraint $\mathsf{E}\left[\widehat{\boldsymbol{S}} - \boldsymbol{S}\right] = \mathbf{0}$

⇒ simple kriging: *BLUP* (**B**est **L**inear **U**nbiased **P**redictor)

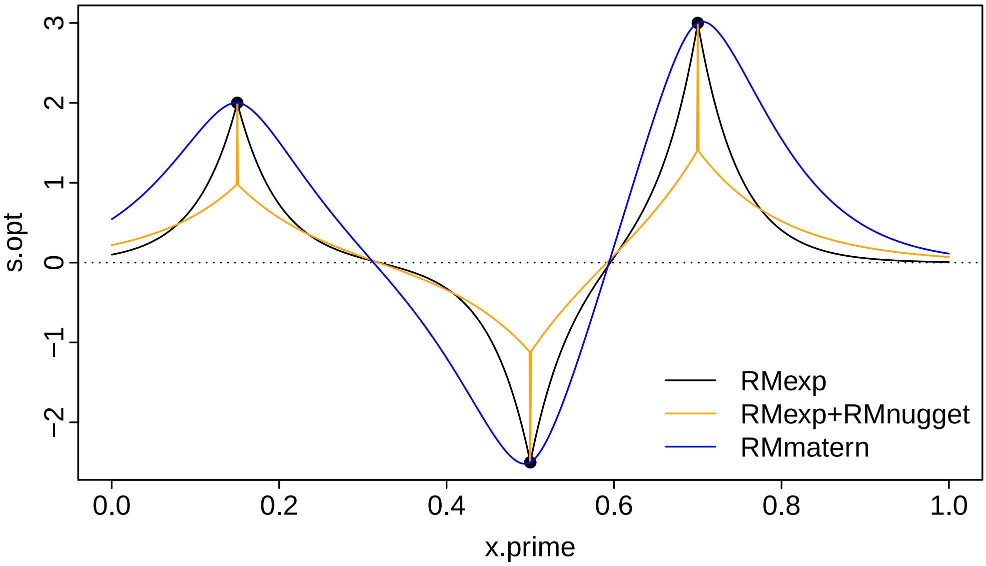
- write $\boldsymbol{\mu} = \boldsymbol{X_S}\boldsymbol{\beta}$ and $\boldsymbol{M} = \boldsymbol{\Gamma}_{\boldsymbol{YY}}^{-1}(\boldsymbol{y} - \boldsymbol{X_Y}\boldsymbol{\beta})$, where $\boldsymbol{M}$ does not depend on the prediction locations $\boldsymbol{x}'_j$; then for each $\boldsymbol{x}'_j$

$$\widehat{S}_{\mathsf{opt}}(\boldsymbol{x}'_j) = \mu_j + \sum_{i=1}^{n} M_i \gamma(\boldsymbol{x}'_j - \boldsymbol{x}_i)$$

⇒ simple kriging predictor for $\boldsymbol{x}'_j$: weighted sum of covariance terms "pinned down" at data locations $\boldsymbol{x_i}$ (dual form of kriging)

⇒ shape of covariance function (or variogram) close to origin determine shape of prediction surface near data locations

⇒ continuity and diffentiability of variogram at origin control geometrical properties of simple kriging prediction surface



3
2
1
0
−1
−2
s.opt
0.0
0.2
0.4
0.6
0.8
1.0
x.prime
RMexp
RMexp+RMnugget
RMmatern

- evaluating $\widehat{\boldsymbol{S}}_{\text{opt}}$ requires a fully specified weakly stationary model:
  1. structure of trend function known
  2. regression coefficients $\boldsymbol{\beta}$ known
  3. type of parametric covariance (variogram) function known
  4. parameters $\boldsymbol{\theta}, \tau^2$ of covariance function known

- relax assumptions: only 1, 3, 4 assumed to be know, $\boldsymbol{\beta}$ implicitly estimated from data by generalized least squares

$\Rightarrow$ *universal (UK) (external drift, EDK) plug-in kriging predictor*

$$\widehat{\boldsymbol{S}}_{\text{k}} = \boldsymbol{X_S}\widehat{\boldsymbol{\beta}}_{\text{GLS}} + \boldsymbol{\Lambda}(\boldsymbol{y} - \boldsymbol{X_Y}\widehat{\boldsymbol{\beta}}_{\text{GLS}}) \quad \text{with} \quad \boldsymbol{\Lambda} = \boldsymbol{\Sigma_{SY}}\boldsymbol{\Gamma}_{\boldsymbol{YY}}^{-1}$$

$\Rightarrow$ MSEP of UK plug-in predictor

$$\begin{aligned}\text{MSEP}[\widehat{\boldsymbol{S}}_{\text{k}}] &= \text{MSEP}[\widehat{\boldsymbol{S}}_{\text{opt}}] \\ &\quad +(\boldsymbol{X_S} - \boldsymbol{\Lambda}\boldsymbol{X_Y})\text{Cov}\left[\widehat{\boldsymbol{\beta}}_{\text{GLS}}, \widehat{\boldsymbol{\beta}}_{\text{GLS}}^{\mathsf{T}}\right](\boldsymbol{X_S} - \boldsymbol{\Lambda}\boldsymbol{X_Y})^{\mathsf{T}}\end{aligned}$$

- substituting $\widehat{\boldsymbol{\beta}}_{\mathsf{GLS}} = (\boldsymbol{X}^{\mathsf{T}}\boldsymbol{\Gamma}_{\boldsymbol{YY}}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathsf{T}}\boldsymbol{\Gamma}_{\boldsymbol{YY}}^{-1}\boldsymbol{y}$ in expression for UK predictor reveals that
$$\widehat{\boldsymbol{S}}_{\mathsf{k}} = \boldsymbol{K}\boldsymbol{Y}$$
is a *homogeneous linear predictor* ($\boldsymbol{K}$: $m \times n$-matrix with UK weights)

- one may show that $\boldsymbol{K}$ minimizes for any homogeneous linear predictor
$$\text{trace}\left(\mathsf{E}\left[\{\widehat{\boldsymbol{S}} - \boldsymbol{S}\}\{\widehat{\boldsymbol{S}} - \boldsymbol{S}\}^{\mathsf{T}}\right]\right)$$
subject to constraint $\mathsf{E}\left[\widehat{\boldsymbol{S}} - \boldsymbol{S}\right] = \mathbf{0}$

⇒ universal kriging: *eBLUP* (**e**mpirical **B**est **L**inear **U**nbiased [homogeneous] **P**redictor)

- condition for unbiasedness of $\widehat{S}_{\mathsf{k}}$ implies

$$(\boldsymbol{X_S} - \boldsymbol{K}\,\boldsymbol{X_Y}) = \mathbf{0}$$

⇒ if trend model includes an intercept then for each $\boldsymbol{x}'_j$

$$\sum_{i=1}^{n} K_{ji} = 1$$

- dual form: UK predictor can again be written as a weighted sum of covariance terms “pinned down” at data locations

- for special case $\mu(\boldsymbol{x}) =$ const. UK is denoted as *ordinary kriging* (OK) fitting the spatial model by REML

```
> library(gstat)
> library(georob)
> library(lattice)
> data(wolfcamp, package="georob")
> d.w <- wolfcamp
> coordinates(d.w) <- c("x", "y")
> r.georob <- georob(pressure~x+y, d.w,
+   locations=~x+y, variogram.model="RMspheric",
+   param=c(variance=3000, nugget=1000, scale=100),
+   tuning.psi=1000)
```

computing UK predictions of signal $S(\boldsymbol{x}')$

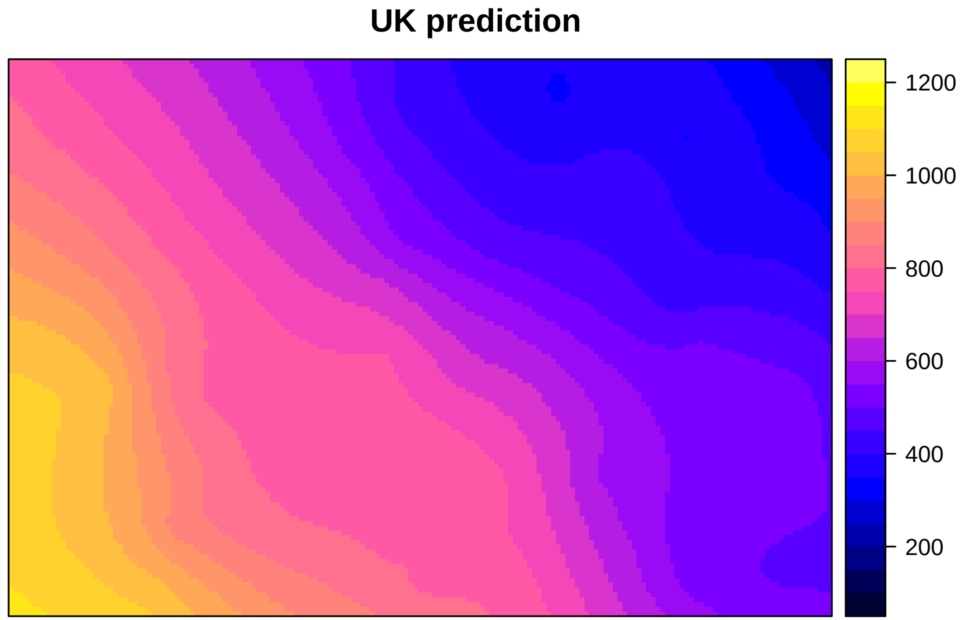
```
> d.w.grid <- expand.grid(
+   x = seq(-240, 190, by= 2.5),
+   y = seq(-150, 140, by= 2.5)
+ )
> r.uk.signal <- predict(r.georob, newdata=d.w.grid)
> str(r.uk.signal)
```

```
'data.frame':        20241 obs. of  6 variables:
 $ x    : num  -240 -238 -235 -232 -230 ...
 $ y    : num  -150 -150 -150 -150 -150 -150 -150 -150 -15..
 $ pred : num  1117 1115 1113 1110 1108 ...
 $ se   : num  58.9 58.3 57.8 57.3 56.8 ...
 $ lower: num  1002 1001 999 998 997 ...
 $ upper: num  1233 1229 1226 1222 1219 ...
 - attr(*, "variogram.object")=List of 1
...
```

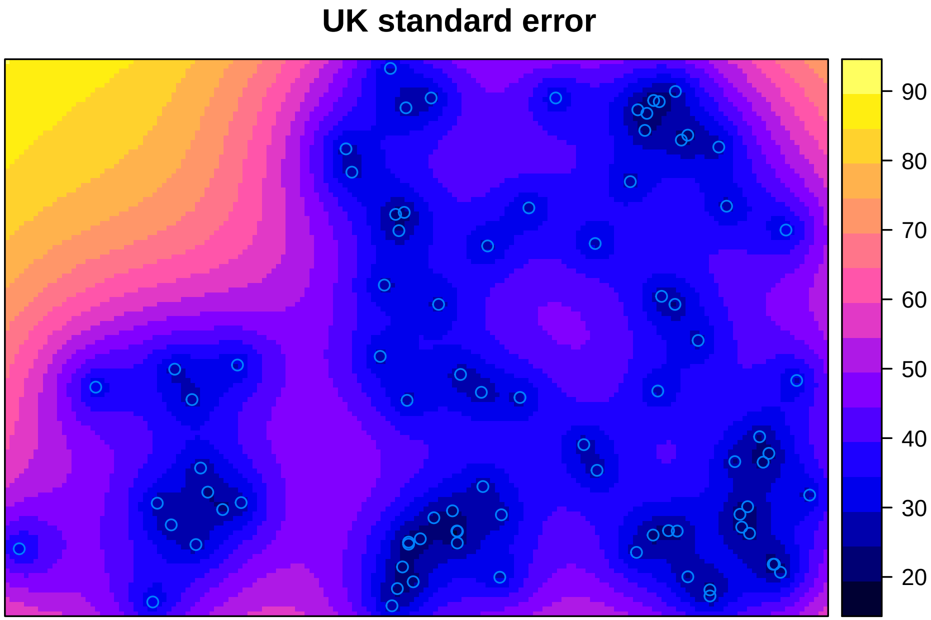
```
> # for plotting convert predictions to SpatialGridDataFrame
> r.uk <- r.uk.signal
> coordinates(r.uk) <- c("x", "y")
> gridded(r.uk) <- TRUE
> fullgrid(r.uk) <- TRUE
> str(r.uk)
```

```
Formal class 'SpatialGridDataFrame' [package "sp"] with 4 ..
  ..@ data       :'data.frame':       20241 obs. of  4 variables:
  .. ..$ pred : num [1:20241] 778 775 771 768 765 ...
  .. ..$ se   : num [1:20241] 89.7 89.5 89.3 89.1 88.9 ...
  .. ..$ lower: num [1:20241] 602 599 596 594 591 ...
  .. ..$ upper: num [1:20241] 954 950 946 943 939 ...
  ..@ grid       :Formal class 'GridTopology' [package "s"..
  .. .. ..@ cellcentre.offset: Named num [1:2] -240 -150
  .. .. .. ..- attr(*, "names")= chr [1:2] "x" "y"
  .. .. ..@ cellsize         : Named num [1:2] 2.5 2.5
  .. .. .. ..- attr(*, "names")= chr [1:2] "x" "y"
  .. .. ..@ cells.dim        : Named int [1:2] 173 117
  .. .. .. ..- attr(*, "names")= chr [1:2] "x" "y"
  ..@ bbox       : num [1:2, 1:2] -241 -151 191 141
  .. ..- attr(*, "dimnames")=List of 2
  .. .. ..$ : chr [1:2] "x" "y"
  .. .. ..$ : chr [1:2] "min" "max"
  ..@ proj4string:Formal class 'CRS' [package "sp"] with 1..
  .. .. ..@ projargs: chr NA
```

```
> # plot UK predictions
> breaks <- seq(50, 1250, by=50)
> spplot(r.uk, zcol="pred", at=breaks,
+   main="UK prediction")
```


UK prediction
1200
1000
800
600
400
200

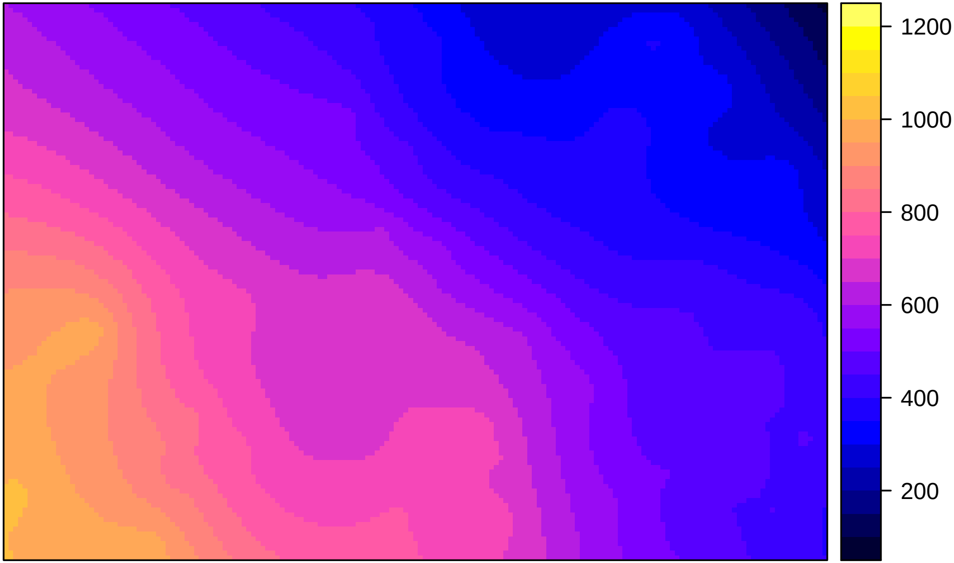
```
> # plot UK prediction standard errors and data locations
> spplot(r.uk, zcol="se", main="UK standard error")
> trellis.focus("panel", row=1, column=1)
> panel.points(x=d.w$x, y=d.w$y)

NULL

> trellis.unfocus()
```



UK standard error
90
80
70
60
50
40
30
20

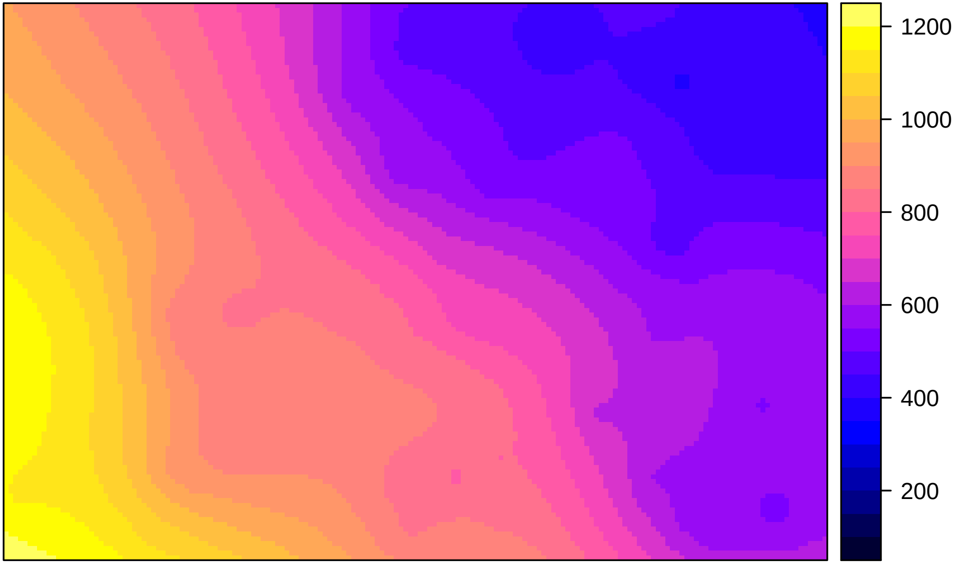
```
> # plot lower limits of 95% prediction intervals
> spplot(r.uk, zcol=”lower”, at=breaks,
+   main=”lower limit 95% prediction interval”)
```

lower limit 95% prediction interval



```
> # plot upper limits of 95% prediction intervals
> spplot(r.uk, zcol="upper", at=breaks,
+   main="upper limit 95% prediction interval")
```

upper limit 95% prediction interval



- Gaussian model fitted to log-transformed response variable $Y(\boldsymbol{x}) = \log\left(U(\boldsymbol{x})\right)$ (e.g. `cu` content Dornach data set)

⇒ computing UK predictions for log-transformed response

⇒ how should we back-transform to original scale of response?

- lognormal distribution

$$Y = \log(U) \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$$

expectation and variance of U

$$\begin{aligned} \mathrm{E}\left[U\right] &= \mu_U = \exp(\mu_Y + 0.5\,\sigma_Y^2) \\ \mathrm{Var}\left[U\right] &= \mu_U^2(\exp(\sigma_Y^2) - 1) \end{aligned}$$

- $\exp\left(\widehat{S}_{\mathsf{k}}(\boldsymbol{x}')\right)$ is a biased predictor of $U(\boldsymbol{x}')$
- unbiased back-transformation

$$\widehat{U}_{\mathsf{lk}}(\boldsymbol{x}') = \exp\left(\widehat{S}_{\mathsf{k}}(\boldsymbol{x}') + 0.5\left\{\mathsf{Var}\left[S(\boldsymbol{x}')\right] - \mathsf{Var}\left[\widehat{S}_{\mathsf{k}}(\boldsymbol{x}')\right]\right\}\right)$$

- limits of prediction intervals can be back-transformed directly by exp$()$
- back-transformation implemented in function `lgnpp` of package `georob`

example: lognormal UK Meuse `zinc` data


```
> library(georob)
> data(meuse); data(meuse.grid)
> coordinates(meuse.grid) <- ~x+y
> meuse.grid <- as(meuse.grid, "SpatialPixelsDataFrame")
> r.logzn <- georob(log(zinc)~sqrt(dist), meuse,
+   locations=~x+y, variogram.model="RMexp",
+   param=c(variance=0.15, nugget=0.05, scale=200),
+   tuning.psi=1000, control=control.georob(
+     cov.bhat=TRUE, cov.bhat.betahat=TRUE,
+     aux.cov.pred.target=TRUE))
> r.logzn
```

```
Tuning constant:  1000

Fixed effects coefficients:
(Intercept)   sqrt(dist)
      6.985       -2.567

Variogram:  RMexp
        variance    snugget(fixed)            nugget
```

```
    0.14910          0.00000          0.04867
      scale
  192.52854
```

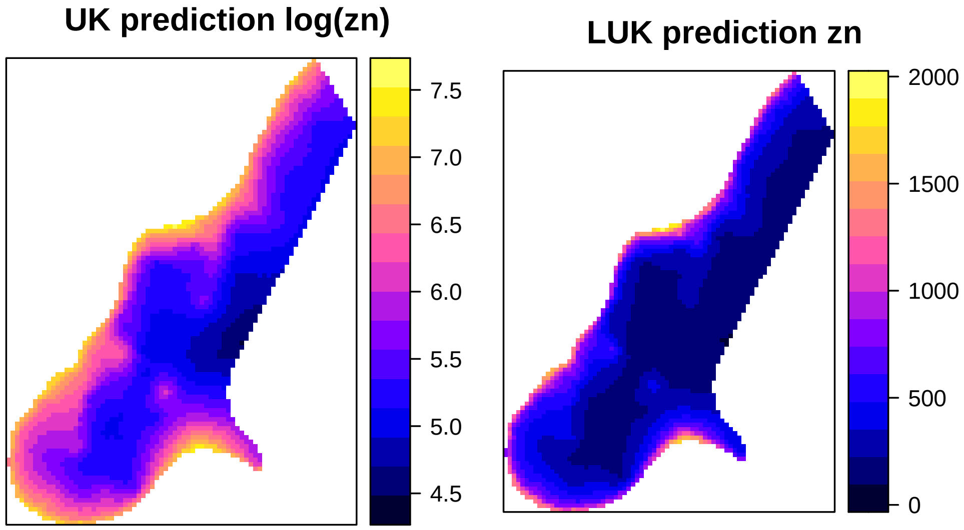
```
> r.luk <- predict(r.logzn, newdata=meuse.grid,
+   control=control.predict.georob(extended.output=TRUE))
> str(r.luk@data)
```

```
'data.frame':        3103 obs. of  8 variables:
 $ pred           : num  7.03 7.05 6.75 6.49 7.07 ...
 $ se             : num  0.362 0.335 0.342 0.349 0.288 ...
 $ lower          : num  6.32 6.39 6.08 5.8 6.5 ...
 $ upper          : num  7.73 7.7 7.42 7.17 7.63 ...
 $ trend          : num  6.99 6.99 6.7 6.45 6.99 ...
 $ var.pred       : num  0.0388 0.0539 0.0437 0.0357 0.078..
 $ cov.pred.target: num  0.0285 0.0453 0.0379 0.0315 0.072..
 $ var.target     : num  0.149 0.149 0.149 0.149 0.149 ...
...
```

```
> r.luk <- lgnpp(r.luk)
> str(r.luk@data)
```

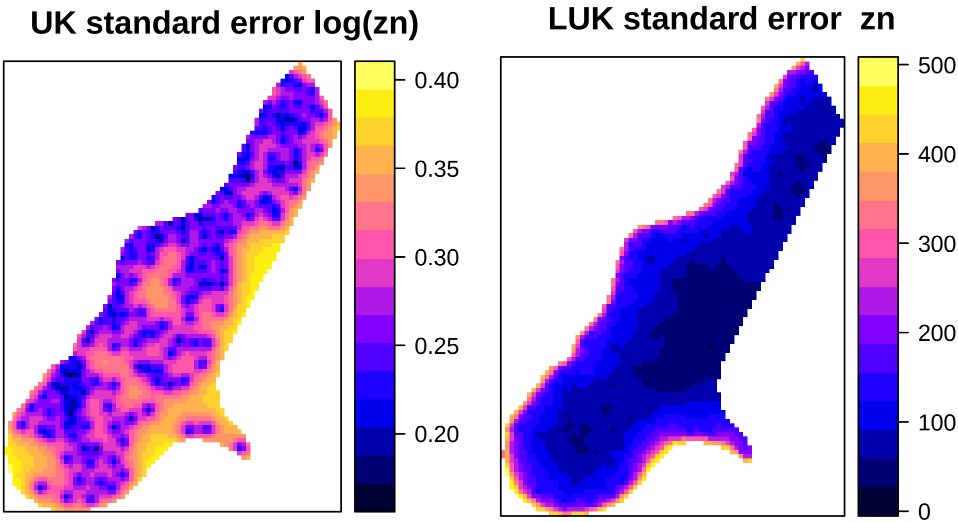
```
'data.frame':        3103 obs. of  12 variables:
 $ pred           : num  7.03 7.05 6.75 6.49 7.07 ...
 $ se             : num  0.362 0.335 0.342 0.349 0.288 ...
 $ lower          : num  6.32 6.39 6.08 5.8 6.5 ...
 $ upper          : num  7.73 7.7 7.42 7.17 7.63 ...
 $ trend          : num  6.99 6.99 6.7 6.45 6.99 ...
 $ var.pred       : num  0.0388 0.0539 0.0437 0.0357 0.078..
 $ cov.pred.target: num  0.0285 0.0453 0.0379 0.0315 0.072..
 $ var.target     : num  0.149 0.149 0.149 0.149 0.149 ...
 $ lgn.pred       : num  1189 1204 902 694 1215 ...
 $ lgn.se         : num  440 409 314 249 354 ...
 $ lgn.lower      : num  554 595 438 331 667 ...
 $ lgn.upper      : num  2286 2215 1673 1299 2064 ...
...
```

```
> print(spplot(r.luk, zcol="pred",
+   main="UK prediction log(zn)"),
+   position=c(0, 0, 0.5, 1), more=TRUE)
> print(spplot(r.luk, zcol="lgn.pred",
+   main="LUK prediction zn"),
+   position=c(0.5, 0, 1, 1))
```



UK prediction log(zn)
7.5
7.0
6.5
6.0
5.5
5.0
4.5
LUK prediction zn
2000
1500
1000
500
0

```
> print(spplot(r.luk, zcol=”se”,
+   main=”UK standard error log(zn)”),
+   position=c(0, 0, 0.5, 1), more=TRUE)
> print(spplot(r.luk, zcol=”lgn.se”,
+   main=”LUK standard error  zn”),
+   position=c(0.5, 0, 1, 1))
```


UK standard error log(zn)
0.40
0.35
0.30
0.25
0.20
LUK standard error zn
500
400
300
200
100
0

- mean squared error (MSE) captures bias and random variation
- mean squared prediction error (MSEP) usual criterion for optimality of predictions
- optimal predictor (which minimises MSEP): conditional expectation of prediction target, given observations
- Gaussian random processes: optimal predictor $\equiv$ simple kriging predictor
- simple kriging predictor: weighted sum of observations with weights equal to $\boldsymbol{\Sigma}_{\boldsymbol{SY}}\boldsymbol{\Gamma}_{\boldsymbol{YY}}^{-1}$; $\boldsymbol{\Sigma}_{\boldsymbol{SY}}$ accounts for auto-correlation between target and observations and $\boldsymbol{\Gamma}_{\boldsymbol{YY}}^{-1}$ for auto-correlation between observations

summary section 6

- simple kriging predictor: weighted sum of covariance (variogram) terms "pinned-down" at observation locations (dual form)
- universal kriging predictor: approximation of simple kriging predictor where $\boldsymbol{\beta}$ is estimated by $\widehat{\boldsymbol{\beta}}_{\text{GLS}}$
- MSEP of universal kriging predictor equal to MSEP of simple kriging predictor plus a term that accounts for the estimation of $\boldsymbol{\beta}$
- computing universal kriging predictor requires:
  1. known structure of trend function
  2. known structure and parameters $\boldsymbol{\theta}, \tau^2$ of covariance function or variogram

$\Rightarrow$ "plug-in" predictor: uncertainty of variogram is ignored when computing predictions

References

Diggle, P. J. and Ribeiro, Jr., P. J. (2007). *Model-based Geostatistics*. Springer, New York.

7 model assessment by cross-validation

- data analysis often leads to a set of equally plausible candidate models that use different sets of covariates and/or different variograms
- covariate selection by `step()` does not lead to a unique set of covariates but depends on the search strategy and on criterion (AIC/BIC) used for model section
- likelihood ratio tests cannot be used to compare the goodness-of-fit of models that have different variograms
- goodness-of-fit not necessarily good criterion for judging quality of predictions

$\Rightarrow$ models should be assessed by their precision to predict new data

- general strategy to assess precision of predictions of new data by a statistical model (Hastie *et al.*, 2009, chap. 7)
- recipe:

1. split data set (randomly) into K subsets (typically $K = 5$ or $K = 10$)
2. for each $k = 1, \ldots, K$
   (a) exclude observations of k^{th} subset and fit model to remaining data
   (b) predict with this model (and excluding again the data of the k^{th} subset) all observations $Y(\boldsymbol{x}_i)$ of the k^{th} subset and compute prediction errors $\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i) - y_i$
3. pool prediction errors for all subsets and compute statistics of $\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i) - y_i$ for evaluating prediction precision and the accuracy of modelling prediction uncertainty (e.g. MSEP$[\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i)]$)

- root mean square error RMSE

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left\{\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i) - y_i\right\}^2}$$

⇒ overall measure of precision (bias and random variation)

- bias

$$\text{BIAS} = \frac{1}{n}\sum_{i=1}^{n}\left\{\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i) - y_i\right\}$$

- robust variants:

$$\begin{aligned} \text{robBIAS} &= \text{median}_i(\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i) - y_i) \\ \text{robRMSE} &= \text{MAD}_i(\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i) - y_i) = 1.4826\,\text{median}_i(|\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i) - y_i|) \end{aligned}$$

- R^2 measures strength of linear dependence between y_i and $\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i)$ and is not a measure of precision

- mean of squared standardized prediction errors

$$\text{MSSE} = \frac{1}{n}\sum_{i=1}^{n}\frac{\{\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i) - y_i\}^2}{\text{MSEP}[\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i)]} \quad \text{should match 1}$$

- robust variant of MSSE for normally distributed prediction errors

$$\text{MEDSSE} = \text{median}_i\left(\frac{\{\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i) - y_i\}^2}{\text{MSEP}[\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}_i)]}\right) \quad \text{should match 0.455}$$

7.3 criteria to assess probabilistic predictions

- for Gaussian stochastic processes kriging provides estimates of mean and variance of conditional distribution of target $Y(\boldsymbol{x}'_j)$ given the data $\boldsymbol{Y}$

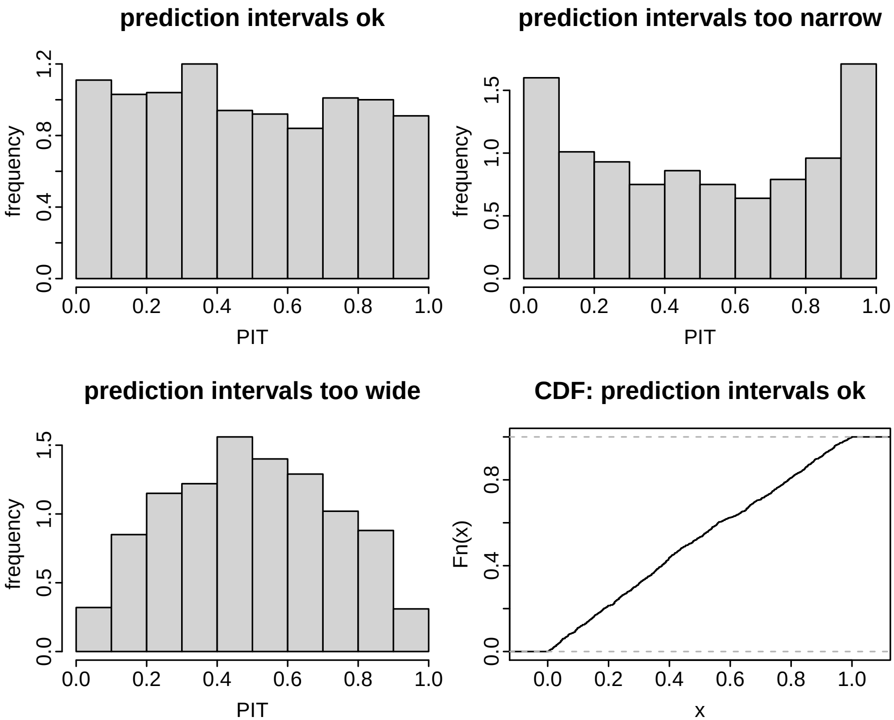
$$Y(\boldsymbol{x}'_j)|\boldsymbol{Y} \sim \mathcal{N}(\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}'_j), \mathsf{MSEP}[\widehat{Y}_{\mathsf{k}}(\boldsymbol{x}'_j)])$$

- denote cdf of predictive distribution by $\widehat{F}_{Y(\boldsymbol{x}'_j)|\boldsymbol{Y}}(y)$
- probability integral transform PIT (Gneiting *et al.*, 2007)

$$\mathsf{PIT}_j = \widehat{F}_{Y(\boldsymbol{x}'_j)|\boldsymbol{Y}}(y_j)$$

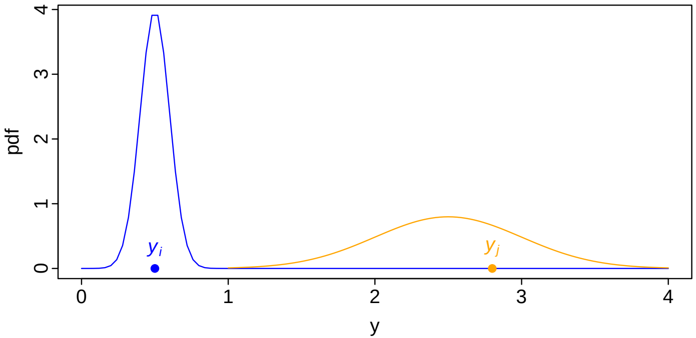
- PIT has a uniform distribution on interval $[0,1]$ if predictive distribution is ok

$\Rightarrow$ histogram of PIT_j should be flat



prediction intervals ok
prediction intervals too narrow
prediction intervals too wide
CDF: prediction intervals ok
frequency
PIT
Fn(x)
x

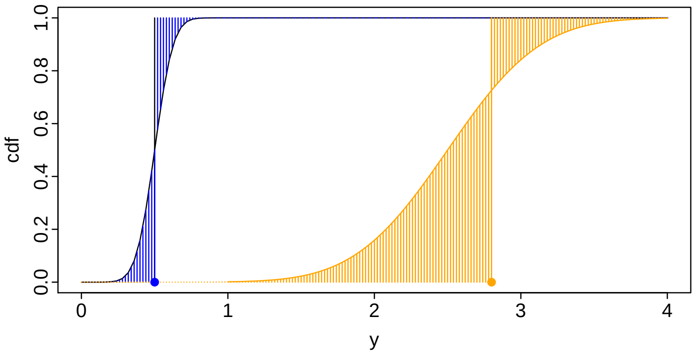
- overall criterion to assess quality of probabilistic predictions
- predictive distribution is "sharp" if it is narrow (small variance) and is centred on true value (no bias)



- measure for sharpness of predictive distribution for single prediction site $\boldsymbol{x}'_j$

$$\int \{\widehat{F}_{Y(\boldsymbol{x}'_j)|\boldsymbol{Y}}(y) - I(y_j \leq y)\}^2 dy$$

  where $I(A)$ is indicator function with value equal to 1 if A is true and zero otherwise



- continuous ranked probability score (CRPS) measures average sharpness of predictive distributions for all sites of a data set

$$\text{CRPS} = \frac{1}{n}\sum_{j=1}^{n}\int_{-\infty}^{\infty}\{\widehat{F}_{Y(\boldsymbol{x}'_j)|\boldsymbol{Y}}(y) - I(y_j \leq y)\}^2 dy$$

- CRPS equal to integral over Brier score (BS = averaged MSEP for predicting that observations y_j do not exceed cutoff y)

$$BS(y) = \frac{1}{n}\sum_{j=1}^{n}\{\widehat{F}_{Y(\boldsymbol{x}'_j)|\boldsymbol{Y}}(y) - I(y_j \leq y)\}^2$$

⇒ CRPS criterion of choice for assessing quality of probabilistic predictions (strictly proper scoring rule, cf. Gneiting *et al.*, 2007)

```
> library(georob)
> d.w <- wolfcamp
> coordinates(d.w) <- c("x", "y")
> d.w$xs <- d.w$x - mean(d.w$x)
> d.w$ys <- d.w$y - mean(d.w$y)
> r.georob.full <- georob(
+   pressure~xs+ys+I(xs^2)+I(ys^2)+xs:ys, d.w,
+   locations=~x+y, variogram.model="RMspheric",
+   param=c(variance=3000, nugget=1000, scale=100),
+   tuning.psi=1000)
> r.georob.bic <- update(r.georob.full,
+   .~xs+ys)
> r.cv.full <- cv(r.georob.full, seed=30)
> r.cv.bic <- cv(r.georob.bic, seed=30)
```

```
> logLik(r.georob.full, REML = TRUE)
'log Lik.' -470.3894 (df=9)
> logLik(r.georob.bic, REML = TRUE)
'log Lik.' -456.3801 (df=6)
> extractAIC(r.georob.full)
[1]   9.0000 929.9218
> extractAIC(r.georob.bic)
[1]   6.0000 929.5165
> extractAIC(r.georob.full, k=log(nrow(d.w)))
[1]   9.0000 951.9057
> extractAIC(r.georob.bic, k=log(nrow(d.w)))
[1]   6.0000 944.1724
```

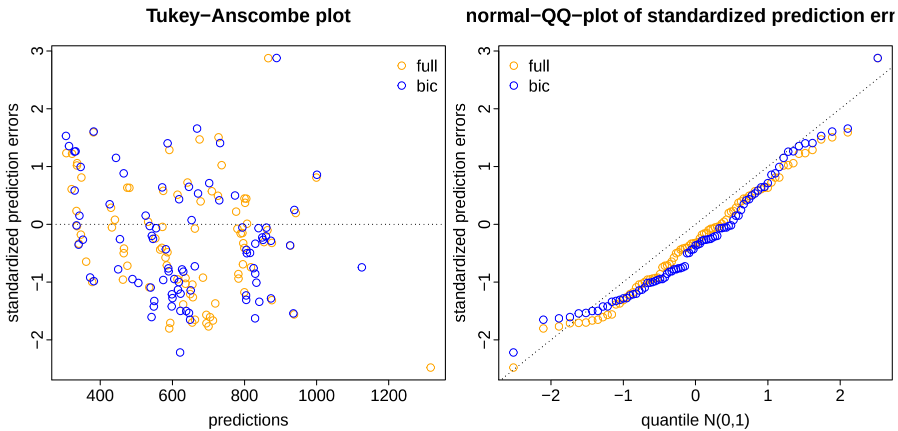
```
> summary(r.cv.full, se=TRUE)
```

```
Statistics of cross-validation prediction errors
    me         mede       rmse       made       qne
    -28.1991   -22.9642    90.4292    73.2048    81.2919
se   29.8780    28.9144    20.8618    33.5121     6.9164
    msse       medsse     crps
      1.0185     0.5056    47.7585
se    0.5546     0.5881    16.5063
```

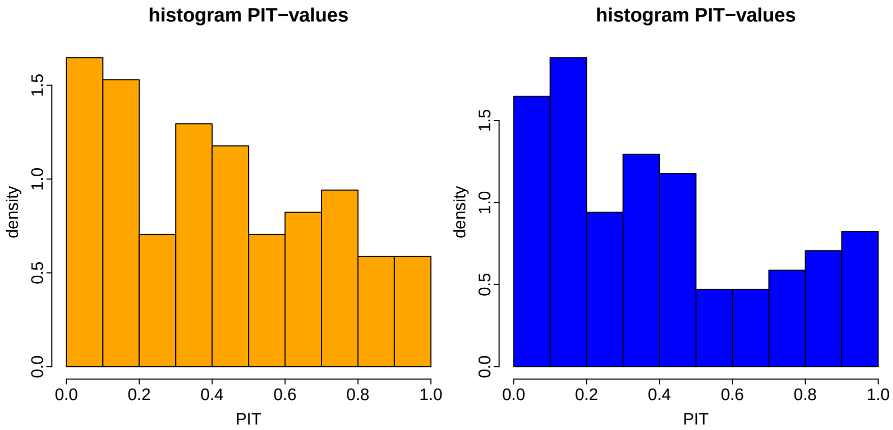
```
> summary(r.cv.bic, se=TRUE)
```

```
Statistics of cross-validation prediction errors
    me         mede       rmse       made       qne
    -23.9936   -26.1068    73.9683    90.1443    72.0825
se   16.1544    14.6219     5.6358     8.6899     6.8800
    msse       medsse     crps
      1.0505     0.7311    42.1020
se    0.1750     0.1057     3.2357
```

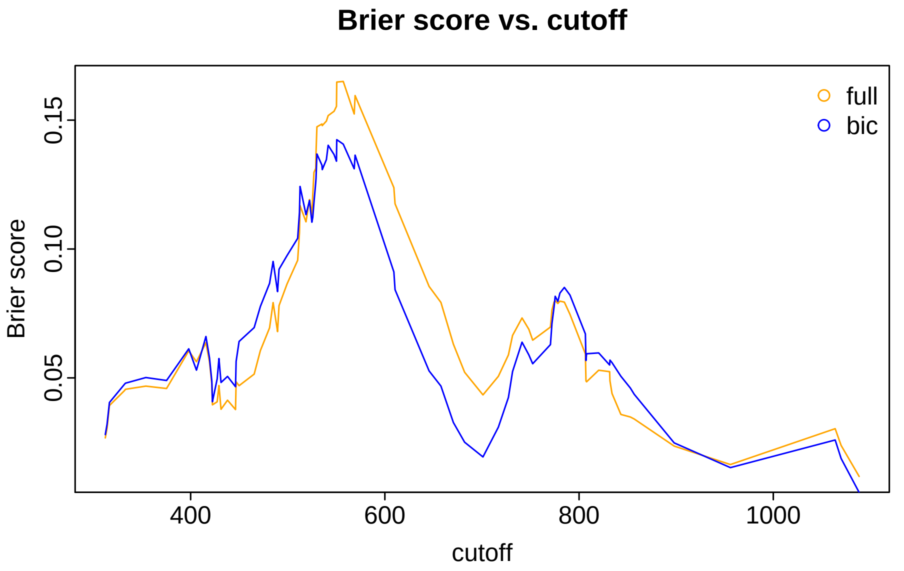
```
> palette(c(”orange”, ”blue”))
> op <- par(mfrow=c(1,2))
> plot(r.cv.full, type=”ta”)
> plot(r.cv.bic, add=T, col=2, type=”ta”)
> abline(h=0, lty=”dotted”)
> legend(”topright”, pch=1, col=1:2,
+   legend=c(”full”, ”bic”), bty=”n”)
> plot(r.cv.full, type=”qq”,
+   ylab=”standardized prediction errors”)
> plot(r.cv.bic, add=T, col=2, type=”qq”, xlab=””)
> abline(0, 1, lty=”dotted”)
> legend(”topleft”, pch=1, col=1:2,
+   legend=c(”full”, ”bic”), bty=”n”)
> par(op)
> palette(”default”)
```


Tukey−Anscombe plot
standardized prediction errors
predictions
400
600
800
1000
1200
−2
−1
0
1
2
3
full
bic
normal−QQ−plot of standardized prediction err
standardized prediction errors
quantile N(0,1)
−2
−1
0
1
2
3
full
bic

```
> palette(c("orange", "blue"))
> op <- par(mfrow=c(1,2))
> plot(r.cv.full, type="hist.pit")
> plot(r.cv.bic, col=2, type="hist.pit")
> par(op)
> palette("default")
```



```
> palette(c("orange", "blue"))
> plot(r.cv.full, type="bs")
> plot(r.cv.bic, add=T, col=2, type="bs")
> legend("topright", pch=1, col=1:2,
+   legend=c("full", "bic"), bty="n")
> palette("default")
```



Brier score vs. cutoff
full
bic
0.05
0.10
0.15
Brier score
400
600
800
1000
cutoff

References

Gneiting, T., Balabdaoui, F., and Raftery, A. E. (2007). Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society Series B*, **69**(2), 243–268.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning; Data Mining, Inference and Prediction*. Springer, New York, second edition.