
Applied geostatistics

Exercise 5a: Predicting from point samples (Part 3)

Using secondary information

D G Rossiter

*University of Twente, Faculty of Geo-Information Science & Earth
Observation (ITC)*

June 22, 2014

Contents

1	Introduction	1
2	Setup	2
3	Feature-space modelling	2
3.1	Merging undersampled classes	4
3.2	Visualizing feature-space dependence	5
3.3	Modelling feature-space dependence	6
3.4	Answers for feature-space modelling	9
4	Computing and modelling the residual variogram	10
4.1	Answers for Computing and modelling the residual variogram	13
5	The interpolation grid	14
5.1	Answers for Creating the interpolation grid	17
6	Kriging with external drift	17
6.1	KED prediction variance	21
6.2	Answers for Kriging with external drift	25
7	Validation	26

Version 2.2. Copyright © 2007–2014 University of Twente, Faculty ITC
All rights reserved. Reproduction and dissemination of the work as a whole
(not parts) freely permitted if this original copyright notice is included. Sale
or placement on a web site where payment must be made to access this
document is strictly prohibited. To adapt or translate please contact the
author (<http://www.itc.nl/personal/rossiter>).

8	Cleaning up	26
9	*Stratified kriging	27
9.1	Determining the stratification	27
9.2	Per-stratum variograms	29
9.3	* Answers for Stratified Kriging	40
10	Self-test	41
	References	43
	Index of R concepts	44

少小不努力，老大徒伤悲
“If a young person does not strive to learn, in old age he will
only have regrets.”
– Chinese proverb

1 Introduction

In Exercise 4 §3 we saw how a feature-space attribute could be used for prediction. This had the disadvantage that the prediction was the same everywhere in a stratum, and did not take account of local spatial dependence. In Exercise 4 §4 we saw how to predict using Ordinary Kriging; this used a model of local spatial dependence but no information on feature-space attributes other than the target variable. These two approaches can be combined into a **mixed predictor** that uses both sources of information. There are three main approaches for this:

1. Kriging with External Drift (KED): model residual spatial variability, use classes and this model with all samples to predict;
2. Regression Kriging (RK): compute feature-space model, compute residuals from this, model residual spatial variability, use Simple Kriging (with expected value 0) to predict residuals, add in the class means;
3. Stratified Kriging (StK): stratify the area by class, model the variogram in each class separately, predict by OK in each stratum separately, merge the classes for the final map;
4. Stratified Co-Kriging (StCoK): as StK, but use non-collocated cokriging to predict within all the strata from all points (in the same or different stratum); merge for the final map.

These each have advantages and disadvantages:

1. KED models both local and feature-space in one kriging system, so it is the easiest extension to OK. The mathematics are exactly the same as for UK;
2. RK is quite flexible but requires that we model the feature space and residuals separately;
3. StK allows the possibility of different spatial structure (e.g. variograms) in the different strata; however the sample size must be large enough to estimate this structure in each stratum separately;
4. StCoK requires modelling cross-variograms with a linear model of co-regionalization.

In this exercise we will examine KED and (optionally) StK.

After completing this exercise you should be able to:

1. Model feature-space relations between a continuous target variable and a categorical (classified) predictor variable;

2. Model local spatial dependence after accounting for these feature-space relations;
3. Predict attributes over a region using Kriging with External Drift.
4. (Optionally) Predict attributes over a region using Stratified Kriging.

2 Setup

We continue with the Jura dataset from Exercise 4.

Task 1 : Set up this exercise as explained in the following list. •

1. If R is not already running, start it.
2. If you haven't already done so, load the `gstat` and `sp` libraries.
3. If the calibration dataset `jura.cal` is not loaded as a spatial object, do so.
4. If the fitted variogram model `vmf` from Exercise 4 §4.1 is no longer in the workspace, re-create it.

If you followed the instructions in Exercise 4, these should all have been saved in file `JuraEx4.RData`, so they can be restored with the `load` method:

```
> load("JuraEx4.RData")
```

Note: If this file is not in the current directory, you either have to change the directory with the `setwd` method, or add the full path to the argument of the `load` method.

3 Feature-space modelling

The first step in a mixed predictor is to see whether the feature space provides any useful information with respect to the target variable.

Q1 : *What are the categorical attributes in the Jura dataset? Are these expected to be predictors of heavy metals in soils? Why or why not?* [Jump to A1](#) •

```
> str(jura.cal@data)
```

```
'data.frame':      259 obs. of  9 variables:
 $ Rock: Factor w/ 5 levels "Argovian","Kimmeridgian",...: 3 2 3 2 5 5 5 1 1 3 ...
 $ Land: Factor w/ 4 levels "Forest","Pasture",...: 3 2 2 3 3 3 3 3 3 3 ...
 $ Cd  : num  1.74 1.33 1.61 2.15 1.56 ...
 $ Cu  : num  25.72 24.76 8.88 22.7 34.32 ...
 $ Pb  : num  77.4 77.9 30.8 56.4 66.4 ...
 $ Co  : num  9.32 10 10.6 11.92 16.32 ...
 $ Cr  : num  38.3 40.2 47 43.5 38.5 ...
 $ Ni  : num  21.3 29.7 21.4 29.7 26.2 ...
 $ Zn  : num  92.6 73.6 64.8 90 88.4 ...
```

We first see how many observations represent each level of the categorical variables. Further, to be able to explore possible interactions, we need to see their cross-tabulation.

Task 2 : Compute and display a cross-tabulation of the number of observations in each rock category and land use category. •

The `table` function (cross-)tabulates categorical variables.

```
> table(jura.cal$Rock)
```

Argovian	Kimmeridgian	Sequanian	Portlandian
53	85	63	3
Quaternary			
55			

```
> table(jura.cal$Land)
```

Forest	Pasture	Meadow	Tillage
33	56	165	5

```
> table(jura.cal$Rock, jura.cal$Land)
```

	Forest	Pasture	Meadow	Tillage
Argovian	7	6	39	1
Kimmeridgian	22	18	44	1
Sequanian	3	25	33	2
Portlandian	1	1	1	0
Quaternary	0	6	48	1

Note: The factor names of the rock types are not consistent: Quaternary, now called the Pleistocene, is an epoch, which is a higher division than Kimmeridgian, a stage within the Late Jurassic epoch of the Jurassic period. The remaining names are now not used for international correlation. The Sequanian (younger) and Argovian are sub-stages of the Oxfordian stage, which is just below (older than) Kimmeridgian in the stratigraphic column. The Portlandian is roughly equivalent to the middle part of the youngest of the Late Jurassic stage, now called the Tithonian, above the Kimmeridgian¹. So the stratigraphic order, from oldest to youngest is: Argovian (early Oxfordian), Sequanian (later Oxfordian), Kimmeridgian, (discontinuity), Portlandian (middle Tithonian), (gap of ≈ 140 Ma), Quaternary.

Q2 : (1) Which rock types are poorly represented in the sample set?

Which land uses are poorly represented in the sample set?

Which combinations have no observations at all? What does this imply for analysis? Jump to A2 •

¹ http://www.stratigraphy.org/bak/geowhen/stages/Late_Jurassic.html

3.1 Merging undersampled classes

Any analysis using such an unbalanced observations will be very uncertain for the classes with few observations, and impossible for the interactions with no observations. So before proceeding we combine the small classes (Portlandian rocks and Tillage land use) with their most similar classes. This requires expert knowledge from a geologist and land use specialist, respectively.

- The **Tillage** land use class is managed; the other managed class is **Pasture**. **Meadow** and **Forest** are unmanaged. If Co concentration is influenced by land use, that would come from management. So it seems logical to merge the tillage and pasture classes.
- The Portlandian rock type is a Jurassic formation, closest in age to the Kimmeridgian.

Task 3 : Create two new categorical variables in the data frame (1) **Rock4**, combining Portlandian and Sequanian into a class **SeqPortlandian**, and (2) **Land3**, combining **Tillage** and **Pasture** into a class **PasTill**. Keep the other classes as they were. •

We take the chance to use simpler class names. Although the original field is a factor, the new field is created as a string variable, so the `as.factor` function must be applied to it, to convert to a factor (categorical variable).

```
> jura.cal$Rock4 <- ifelse(jura.cal$Rock == "Argovian",
  "Argo", ifelse(jura.cal$Rock == "Sequanian",
    "Seq", ifelse(jura.cal$Rock == "Quaternary",
      "Quat", "KimmPort")))
> class(jura.cal$Rock4)

[1] "character"

> jura.cal$Rock4 <- as.factor(jura.cal$Rock4)
> class(jura.cal$Rock4)

[1] "factor"

> table(jura.cal$Rock4)

      Argo KimmPort      Quat      Seq
      53      88      55      63

> jura.cal$Land3 <- ifelse(jura.cal$Land == "Forest",
  "Forest", ifelse(jura.cal$Land == "Meadow",
    "Meadow", "PasTill"))
> jura.cal$Land3 <- as.factor(jura.cal$Land3)
> table(jura.cal$Land3)

Forest Meadow PasTill
   33    165     61
```

Note: Another way to do this is with the `recode` function of John Fox’s `car` “Companion to Applied Regression” package [1]. This has a somewhat unusual syntax: it first gives the factor to be recoded, and then a so-called “recode specification”; see `?recode` for details.

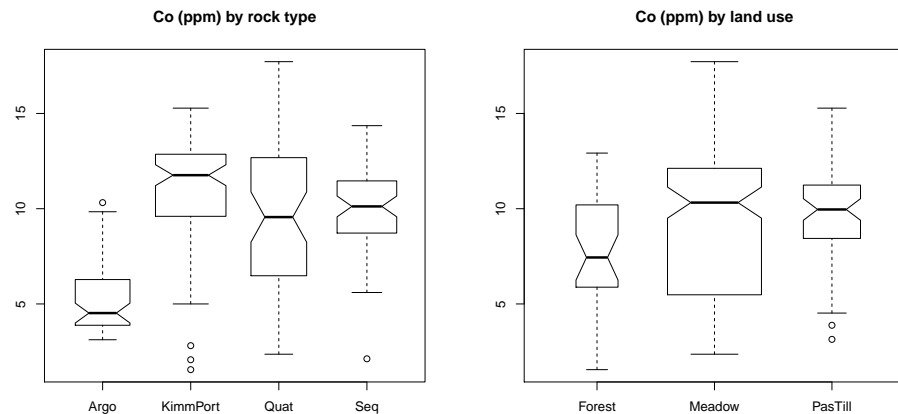
3.2 Visualizing feature-space dependence

We next **visualize** whether there is any difference in the target variable `Co` (cobalt) due to these factors.

Task 4 : Display **classified boxplots** of `Co` by rock type and by land use. •

We use the `boxplot` method with a formula:

```
> par(mfrow=c(1,2))
> boxplot(Co ~ Rock4, data=jura.cal@data,
          varwidth=T, notch=T,
          main="Co (ppm) by rock type")
> boxplot(Co ~ Land3, data=jura.cal@data,
          varwidth=T, notch=T,
          main="Co (ppm) by land use")
> par(mfrow=c(1,1))
```



Note: We have enhanced the boxplot with (1) the optional `varwidth` argument, which scales the width of the box by the number of observations it represents (narrower boxes have fewer observations), and (2) the optional `notch` argument, which draws notches on each side of the median, its approximate confidence interval.

Q3 : Do the areas with different rock types have similar distributions of `Co`? If not, what are the major differences? Jump to A3 •

Q4 : Do the areas with different land uses have similar distributions of `Co`? If not, what are the major differences? Jump to A4 •

Q5 : Which of the two classified variables is expected to provide more differentiation for this target variable? Jump to A5 •

3.3 Modelling feature-space dependence

After visualizing, we can verify our impressions by a **linear model** of the target variable as modelled by the categorical variable; this one-way Analysis of Variance (ANOVA) is computed with the `lm` method and summarized with `summary.lm` (which is automatically called by the generic `summary` method for objects of class `lm`).

Task 5 : Compute and summarize linear models of `Co` predicted by rock type and land use, separately. •

Note that we apply the `summary` method to a linear model object, created by the `lm` method. The linear model object is saved in the workspace with the `<-` assignment operator.

```
> summary(mr <- lm(Co ~ Rock4, data = jura.cal@data))

Call:
lm(formula = Co ~ Rock4, data = jura.cal@data)

Residuals:
    Min       1Q   Median       3Q      Max
-9.438 -1.618  0.087  1.908  8.125

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    5.393      0.404   13.34 < 2e-16 ***
Rock4KimmPort    5.594      0.512   10.93 < 2e-16 ***
Rock4Quat       4.202      0.566    7.42 1.8e-12 ***
Rock4Seq        4.582      0.548    8.35 4.3e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.94 on 255 degrees of freedom
Multiple R-squared:  0.331,    Adjusted R-squared:  0.323
F-statistic: 42.1 on 3 and 255 DF,  p-value: <2e-16

> summary(ml <- lm(Co ~ Land3, data = jura.cal@data))

Call:
lm(formula = Co ~ Land3, data = jura.cal@data)

Residuals:
    Min       1Q   Median       3Q      Max
-7.034 -2.698  0.286  2.586  8.326

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    7.683      0.614   12.50 <2e-16 ***
```



```

Land3Meadow      1.710      0.673      2.54      0.0117 *
Land3PasTill     2.242      0.763      2.94      0.0036 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.53 on 256 degrees of freedom
Multiple R-squared:  0.0338,    Adjusted R-squared:  0.0262
F-statistic: 4.48 on 2 and 256 DF,  p-value: 0.0123

```

Q6 : *How much of the variability is explained by each model? (Hint: Look at the adjusted R^2). Which is thus the better predictor?* [Jump to A6](#) •

Q7 : *In the rock type model, for which rock types is the mean concentration most and least accurately estimated?* [Jump to A7](#) •

Note: The model coefficients show (1) the mean value for the first-listed factor level, here **Argovian** rock type and **Forest** land use. The coefficients for the other factor levels are *differences* from the first-listed factor level. To see the means for each class, use the formula `Co ~ Rock4 - 1` and `Co ~ Land3 - 1`, respectively. The `- 1` removes the intercept from the model. Do *not* pay any attention to the reported R^2 for no-intercept models.

Perhaps there is an interaction between the two predictors.

Task 6 : Compute and summarize a linear model of Co predicted by rock type and land use, together, both as **additive** factors (no interaction) and as **interacting** factors. •

The formula for additive models names the two factors, joined by `+`; the interaction model joins the names by `*`.

First, the additive model:

```
> summary(mrl <- lm(Co ~ Rock4 + Land3, data = jura.cal@data))
```

Call:

```
lm(formula = Co ~ Rock4 + Land3, data = jura.cal@data)
```

Residuals:

```

      Min       1Q   Median       3Q      Max
-7.536 -1.826   0.001   1.739   8.101

```

Coefficients:

```

              Estimate Std. Error t value Pr(>|t|)
(Intercept)    3.133      0.623    5.03 9.3e-07 ***
Rock4KimmPort    5.953      0.502   11.86 < 2e-16 ***
Rock4Quat       3.854      0.551    6.99 2.4e-11 ***
Rock4Seq        4.416      0.544    8.11 2.1e-14 ***
Land3Meadow     2.633      0.571    4.61 6.4e-06 ***
Land3PasTill    2.444      0.640    3.82 0.00017 ***
---

```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.84 on 253 degrees of freedom
Multiple R-squared: 0.384, Adjusted R-squared: 0.372
F-statistic: 31.5 on 5 and 253 DF, p-value: <2e-16

Then, the model with interactions:

```
> summary(mrli <- lm(Co ~ Rock4 * Land3, data = jura.cal@data))
```

Call:

```
lm(formula = Co ~ Rock4 * Land3, data = jura.cal@data)
```

Residuals:

Min	1Q	Median	3Q	Max
-7.642	-1.326	0.001	1.543	7.718

Coefficients: (1 not defined because of singularities)

	Estimate	Std. Error	t value
(Intercept)	5.806	1.027	5.65
Rock4KimmPort	2.345	1.173	2.00
Rock4Quat	-1.037	1.452	-0.71
Rock4Seq	2.674	1.875	1.43
Land3Meadow	-0.927	1.115	-0.83
Land3PasTill	2.040	1.452	1.40
Rock4KimmPort:Land3Meadow	4.781	1.315	3.64
Rock4Quat:Land3Meadow	6.160	1.566	3.93
Rock4Seq:Land3Meadow	2.729	1.982	1.38
Rock4KimmPort:Land3PasTill	1.769	1.673	1.06
Rock4Quat:Land3PasTill	NA	NA	NA
Rock4Seq:Land3PasTill	-0.754	2.201	-0.34

Pr(>|t|)

(Intercept)	4.3e-08 ***
Rock4KimmPort	0.04664 *
Rock4Quat	0.47581
Rock4Seq	0.15502
Land3Meadow	0.40680
Land3PasTill	0.16137
Rock4KimmPort:Land3Meadow	0.00034 ***
Rock4Quat:Land3Meadow	0.00011 ***
Rock4Seq:Land3Meadow	0.16976
Rock4KimmPort:Land3PasTill	0.29135
Rock4Quat:Land3PasTill	NA
Rock4Seq:Land3PasTill	0.73215

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.72 on 248 degrees of freedom
Multiple R-squared: 0.445, Adjusted R-squared: 0.423
F-statistic: 19.9 on 10 and 248 DF, p-value: <2e-16

Another way to compare **hierarchical** feature-space models (i.e. where they are increasingly-complex) is ANOVA. This shows how much more variance is explained by the more complex models, and whether this is a significant difference or could have occurred by chance.

Task 7 : Compare the interaction, additive, and single-factor (rock type) models with an ANOVA. •

We use the `anova` method, specifying the model in descending order of complexity:

```
> anova(mrli, mrl, mr)

Analysis of Variance Table

Model 1: Co ~ Rock4 * Land3
Model 2: Co ~ Rock4 + Land3
Model 3: Co ~ Rock4
  Res.Df  RSS Df Sum of Sq    F Pr(>F)
1     248 1831
2     253 2034 -5      -204  5.52 7.7e-05 ***
3     255 2208 -2      -174 11.77 1.3e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Q8 : *Are the models significantly different? Which increase in complexity gives the most improvement?* [Jump to A8](#) •

Q9 : *In words, what does it mean that the additive and interaction models are significantly better than the single-factor model?* [Jump to A9](#) •

We have seen that the interaction model of both rock type and land use gives the best results; however, for the rest of this exercise we will use only the single-factor model of rock type; this will allow us to visualize the differences in predictions against a single map (i.e. that of rock type) and should help in understanding. In practice you would probably use the best model, since it is quite a bit better than either single-factor model.

Note: However, in this case, there is one combination that does not occur in the sample, so any areas in the map with this combination could not be estimated.

3.4 Answers for feature-space modelling

A1 : *Rock type (field `Rock`) and land use (field `Land`). Both might be associated with soil metals: different rocks have different minerals, which have different proportions of various metals; different land uses may be treated differently (e.g. fertilisation, removal of plant material) which may cause differential concentrations of metals in the soil.* [Return to Q1](#) •

A2 : (1) *There are very few observations on Portlandian rocks.*

(2) *There are very few observations on land use for tillage (arable crops).*

(3) There are no observations in forests on Quaternary rocks or on arable land on soils derived from Portlandian rocks. Further, there are only one or two observations in several other combinations. This implies that we can not analyze all interactions between rock type and land use,

Note that another selection of a calibration set would change these numbers slightly but the overall pattern would not change. [Return to Q2](#) •

A3 : The Argovian rocks have a distribution of Co values that is well below the others (although there is some overlap). The Quaternary sediments have the complete range (probably because they are derived as surface sediments from weathering of all the others).

The Argovian, Sequanian and Kimmeridgian/Portlandian types are all hard rock from the Jurassic period (201.3–145 Ma), whereas Quaternary are soft rocks (sediments) that have been recently deposited. The Co in the soils developed on the weathering products (regolith) of Jurassic rocks is presumably mostly from the original rock, whereas the Co in the soils developed on Quaternary sediments is presumably most from sediments eroded from these. [Return to Q3](#) •

A4 : There is not much difference between land uses, except that pasture/tillage has only two Co values less than 5 mg kg⁻¹ and meadow has all the values greater than 15 mg kg⁻¹. [Return to Q4](#) •

A5 : The rock types appear more discriminating. [Return to Q5](#) •

A6 : Adjusted R^2 are 32.3% (rock type) and 2.6% (land use); clearly rock type is the better predictor. [Return to Q6](#) •

A7 : To answer this, look at the standard error of the coefficients. The minimum (0.404) is for the intercept, which is the first-listed rock type, i.e., Argovian. The maximum (0.566) is for the Quaternary rock type. [Return to Q7](#) •

A8 : Both improvements are highly significant, so it is unlikely that the improvement is due to chance. The improvement from the additive to the interaction model is larger (see the differences in mean sum of squares). [Return to Q8](#) •

A9 : Additive: both factors are significant predictors of Co; Interactive: some combinations of rock type and land use have significantly different Co concentrations than would be expected from the addition of the effects of the two factors separately. [Return to Q9](#) •

4 Computing and modelling the residual variogram

We have seen that the feature space explains a fair amount of the variation. But we also saw that the maximum explained was under half of the total. Where is the rest?

One possibility is the local spatial structure, as we saw for OK (Exercise 4 §4). Now the question is, **after** accounting for the feature-space predictor, is there still local structure?

This is shown by the **residual variogram**, i.e. a plot of the semi-variances of **residuals** (not original values) against separation of point pairs. Fortunately, `gstat` computes these directly, if you provide an appropriate model formula; you do not have to compute the residuals manually.

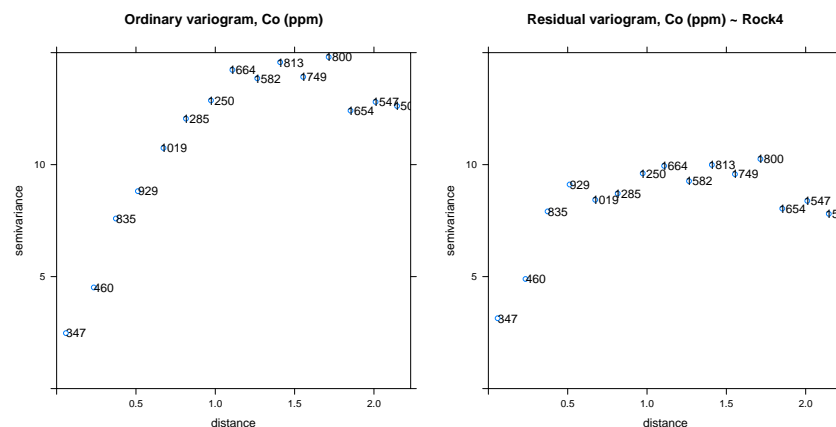
Task 8 : Compute and display the residual variogram for Co, accounting for the feature-space predictor of rock type. Compare it with the ordinary variogram. •

We use the `variogram` method and specify the spatial dependence with the formula `Co ~ Rock4` (as opposed to `Co ~ 1` in the ordinary variogram). This has the same meaning as in the `lm` model specification: the cobalt concentration is to be predicted from the rock type; then the residuals are to be modelled spatially.

```
> vr <- variogram(Co ~ Rock4, loc = jura.cal)
> v <- variogram(Co ~ 1, loc = jura.cal)
```

We plot them on the same scale, using the optional `ylim` argument; the maximum is computed by taking the next highest integer (using the `ceiling` method) than the maximum of all the semivariances:

```
> gamma.max <- ceiling(max(v$gamma, vr$gamma))
> plot.1 <- plot(vr, ylim=c(0,gamma.max),
+               pl=T, main="Residual variogram, Co (ppm) ~ Rock4")
> plot.2 <- plot(v, ylim=c(0,gamma.max),
+               pl=T, main="Ordinary variogram, Co (ppm)")
> print(plot.2, split=c(1,1,2,1), more=T)
> print(plot.1, split=c(2,1,2,1), more=F)
> rm(gamma.max, plot.1, plot.2)
```

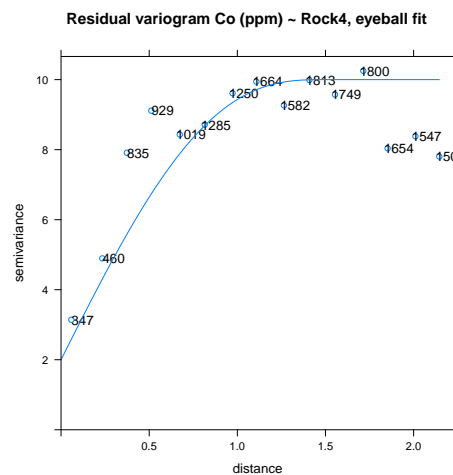


Q10 : Describe the differences between the ordinary and residual variograms (model form, total sill, nugget, range). Jump to A10 •

Task 9 : Model the residual variogram; compare this to the model for the original variogram. •

A pentaspherical model is indicated, since that model was with the original variable. We first make a guess with the `vgm` method:

```
> vrm <- vgm(8, "Pen", 1.5, 2)
> print(plot(vr, pl=T, model=vrm,
             main="Residual variogram Co (ppm) ~ Rock4, eyeball fit"))
```

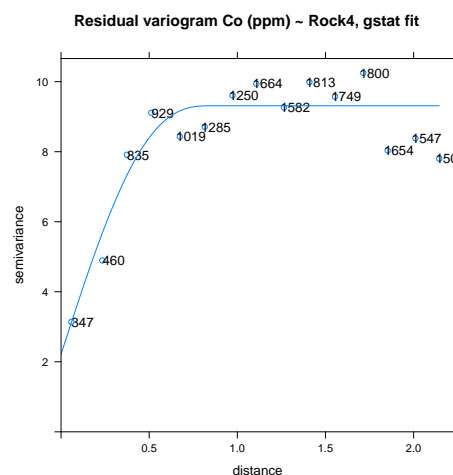


We then ask `gstat` to fit it with the `fit.variogram` method:

```
> (vrmf <- fit.variogram(vr, model=vrm))

model psill range
1  Nug 2.2143 0.00000
2  Pen 7.0973 0.85613

> print(plot(vr, pl=T, model=vrmf,
             main="Residual variogram Co (ppm) ~ Rock4, gstat fit"))
```



Now we can compare the residual model to the original:

```

> vmf

      model  psill  range
1   Nug  1.3712 0.0000
2   Pen 12.9322 1.5239

> vrmf

      model  psill  range
1   Nug  2.2143 0.00000
2   Pen  7.0973 0.85613

> vmf$range[2] - vrmf$range[2]

[1] 0.66777

> 1 - (vmf$range[2] - vrmf$range[2])/vmf$range[2]

[1] 0.5618

> sum(vmf$psill)

[1] 14.303

> sum(vrmf$psill)

[1] 9.3116

> 1 - sum(vrmf$psill)/sum(vmf$psill)

[1] 0.34899

```

Q11 : *How much has the range been reduced? How much has the total sill been reduced?* [Jump to A11](#) •

Q12 : *In words, what is the difference between the models? How can you explain this?* [Jump to A12](#) •

4.1 Answers for Computing and modelling the residual variogram

A10 : *The model form looks more-or-less the same (so the pentaspherical model we used for OK can also be used for KED); the total sill is considerably lower (10 vs. 16 or so) for the residual variogram; the nugget is similar; the range is shorter (0.5 vs. 1.2 or so).* [Return to Q10](#) •

A11 : *The range has been reduced from 1.52 km to 0.87 km; that is 56% of the original. The total sill has been reduced from 14.3 mg kg⁻¹² to 9.3 mg kg⁻¹²; that is a 35% reduction in variability.* [Return to Q11](#) •

A12 : *The spatial dependence is more local (about half the range) and lower overall*

(about 1/3 lower). The rest of the apparent spatial structure has been accounted for in feature space. Return to Q12 •

5 The interpolation grid

Now we want to predict by KED; but first we need an interpolation grid. We already have one which covers a rectangle enclosing the sample points, created in Exercise 4 §4.4. But, this only has point locations where we want to predict. For KED we also need to know the **feature-space** values of the categorical predictor (e.g. rock type and land use) at **every point** to be predicted.

Fortunately, such a grid has been prepared by Pierre Goovaerts, author of a well-known geostatistics text [2] which uses the dataset as its running example. He has kindly provided us this grid, which he created by manually overlaying a 50 m grid (at each intersection of a grid centred every 0.05 km in both directions) of prediction points over printed rock type and land use maps on a light table, and manually recording the category. To save you work, I have prepared this as an R spatial object, as file `Jura50.RData` which can be loaded with the `load` method as workspace object `jura50`.

Note: Since these notes were first written, the same grid has been added to the sample `jura` dataset of the `gstat` package, as object `jura.grid`. This is now automatically loaded, along with the calibration and validation points, by the `data(jura)` command.

Task 10 : Load the file `Jura50.RData` and summarize the newly-created workspace object `jura50`. •

```
> load("Jura50.RData")
> summary(jura50)
```

Object of class SpatialGridDataFrame
Coordinates:
 min max
X 0.275 5.125
Y 0.075 5.925
Is projected: NA
proj4string : [NA]
Grid attributes:
 cellcentre.offset cellsize cells.dim
X 0.3 0.05 97
Y 0.1 0.05 117
Data attributes:

Land	Rock
Forest : 986	Argovian :1185
Pasture:1553	Kimmeridgian:2036
Meadow :3247	Sequanian :1628
Tillage: 171	Portlandian : 316
NA's :5392	Quaternary : 792
	NA's :5392

Q13 : What is the data type of the `jura50` object? What are the dimensions? *Jump to A13*

•

Q14 : Do the proportions of rock types in the grid match those in the calibration data set? *Jump to A14* •

To answer this, we summarize the counts of each class with the `table` method; for an easy-to-interpret table we bind the two results together with the `rbind` “row bind” function:

```
> round(rbind(grid = table(jura50$Rock)/length(jura50$Rock),
  points = table(jura.cal$Rock)/length(jura.cal$Rock)),
  3)
```

	Argovian	Kimmeridgian	Sequanian	Portlandian
grid	0.104	0.179	0.143	0.028
points	0.205	0.328	0.243	0.012

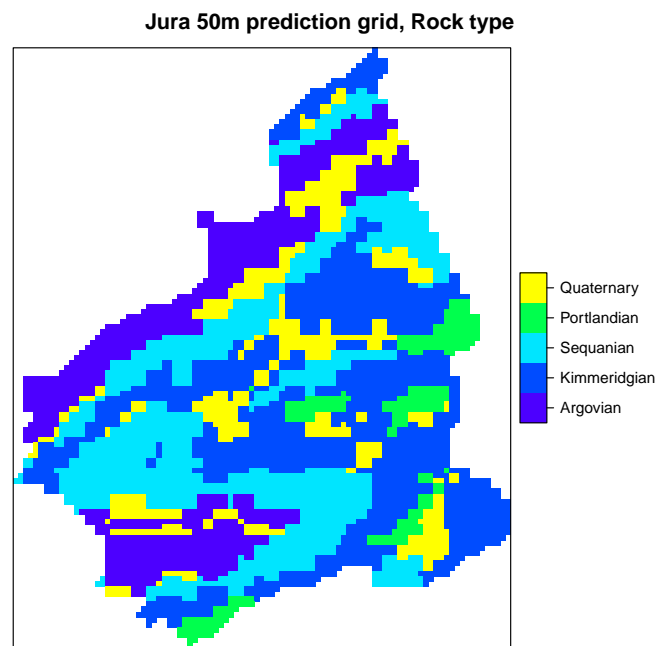
	Quaternary
grid	0.070
points	0.212

Task 11 : Display the distribution of rock types and land uses as maps. •

We use the `spplot` method, saving the two maps as lattice graphics objects for later use.

First the rock type:

```
> map.rock <- spplot(jura50, zcol = "Rock", col.regions = topo.colors(5),
  key.space = "right", main = "Jura 50m prediction grid, Rock type")
> class(map.rock)
> print(map.rock)
```

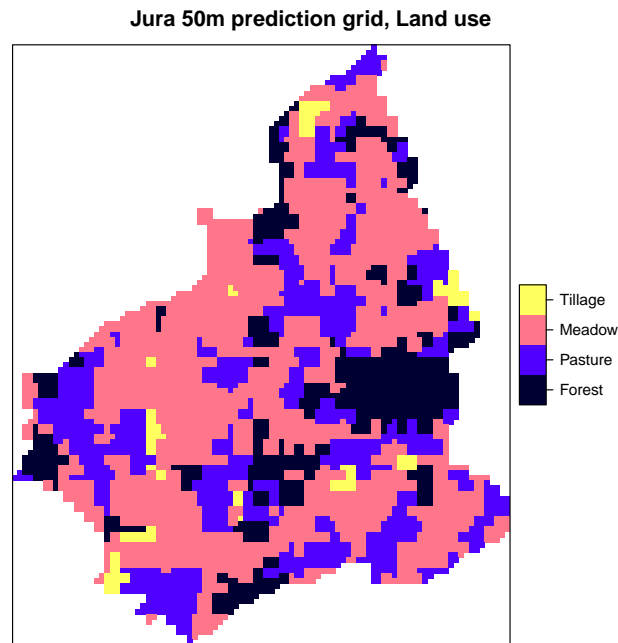


Q15 : *What is the spatial pattern of the rock types? Is there a spatial relation between the two rock types merged in the analysis of the previous section §3.1 (i.e., the under-sampled Portlandian and the Kimmeridgian)?*

Jump to A15 •

Now the land use:

```
> map.land <- spplot(jura50, zcol = "Land", col.regions = bpy.colors(4),
  key.space = "right", main = "Jura 50m prediction grid, Land use")
> print(map.land)
```



Q16 : What is the spatial pattern of the land uses?

[Jump to A16](#) •

5.1 Answers for Creating the interpolation grid

A13 : It is a `SpatialGridDataFrame`, with 97 columns (X-dimension) and 117 rows (Y-dimension).

[Return to Q13](#) •

A14 : Quaternary rocks are substantially over-represented in the calibration set (21% vs. 14%); Portlandian rocks are substantially under-represented (1% vs. 5%); proportions of the other classes are about equal between the grid and calibration set.

[Return to Q14](#) •

A15 : Rock types are generally outcropping following a SW-NE strike. The Portlandian are always adjacent to the Kimmeridgian, as expected from the stratigraphic sequence.

[Return to Q15](#) •

A16 : Land uses form local patches of varying sizes. Most of the area is meadow; there are larger patches of pasture and smaller patches of forest and especially tillage. So most of the area is used for animal production (either natural meadows or managed pasture).

[Return to Q16](#) •

6 Kriging with external drift

We now have a **model of spatial dependence** of the residuals from rock type (model `vrnmf`), and a prediction grid with the rock type known at each location (grid `jura50`). This is what is necessary for KED.

However, the prediction grid still has five rock types, whereas we have merged two of the rock types in the observation points dataset `jura.cal`. Clearly, we have to add a field to the prediction grid to match this merged rock type.

Task 12 : Create a new field in the `jura50` grid for rock type, with the same names, factor levels, and merging the same classes as was done for the observation points in §3.1, above. •

Note: The new field is created as a string variable, so the `as.factor` function must be applied to it, to convert to a factor (categorical variable), to match the data type of the observations.

```
> jura50$Rock4 <- ifelse(jura50$Rock == "Argovian",
  "Argo", ifelse(jura50$Rock == "Sequanian",
    "Seq", ifelse(jura50$Rock == "Quaternary",
      "Quat", "KimmPort")))
> jura50$Rock4 <- as.factor(jura50$Rock4)
> table(jura50$Rock)
```

Argovian	Kimmeridgian	Sequanian	Portlandian
1185	2036	1628	316
Quaternary			
792			

```
> table(jura50$Rock4)
```

Argo	KimmPort	Quat	Seq
1185	2352	792	1628

Task 13 : Predict the Co concentration over the 50 m grid, using kriging with external drift (KED), with rock type as the feature-space predictor. •

The `krige` method will do this, again with a formula `Co ~ Rock` that shows the feature-space dependence:

```
> kr.50 <- krige(Co ~ Rock4, loc = jura.cal, newdata = jura50,
  model = vrmf)
```

```
[using universal kriging]
```

! → **Note carefully** the use of the **residual** variogram model: `model=vrmf`.

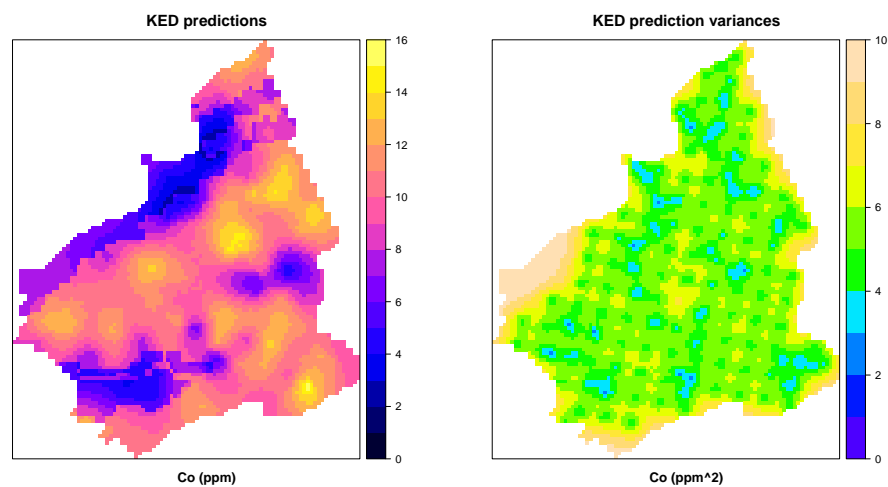
Note: You may have noticed that the `krige` method gives the message `[using universal kriging]` as it works; this is because the form of the kriging equations is exactly the same for KED as for what we have called UK in Exercise 5 §4. We have reserved the term “universal kriging” for the case where the co-predictors are the coördinates; other authors also include feature-space predictors in this term, so they do not use the term “kriging with external drift” at all.

Task 14 : View the kriging predictions and their variances. •

```

> plot.ked <- spplot(kr.50, zcol="var1.pred",
  at=seq(0, ceiling(max(kr.50$var1.pred, na.rm=T)),
    by=1),
  col.regions=bpy.colors(64), key.space="right",
  main="KED predictions",
  sub="Co (ppm)")
> plot.ked.v <- spplot(kr.50, zcol="var1.var",
  at=seq(0, ceiling(max(kr.50$var1.var, na.rm=T)),
    by=1),
  col.regions=topo.colors(64), key.space="right",
  main="KED prediction variances",
  sub="Co (ppm^2)")
> print(plot.ked, split=c(1,1,2,1), more=T)
> print(plot.ked.v, split=c(2,1,2,1), more=F)

```



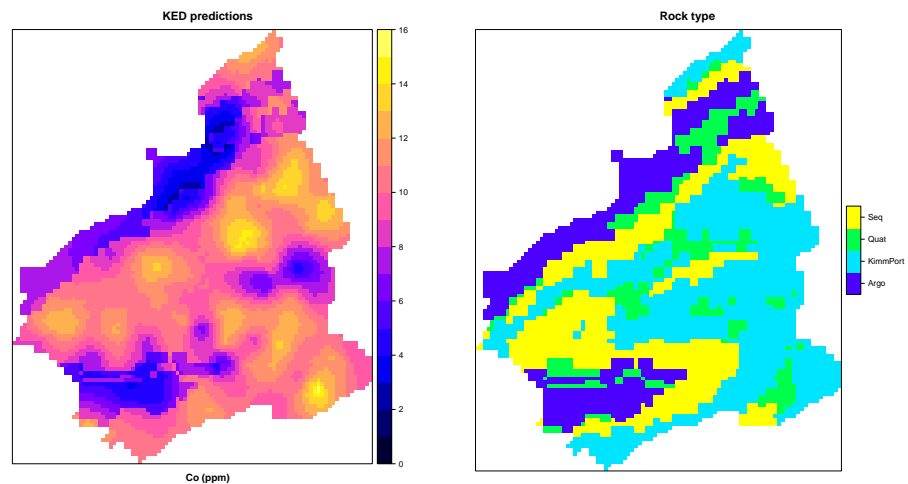
Q17 : *Can you see any evidence of the rock type map in the prediction map? If so, what?* Jump to A17 •

It might be easier to answer the preceding question by showing the KED predictions and the rock types side-by-side:

```

> print(plot.ked, split = c(1, 1, 2, 1), more = T)
> map.rock4 <- spplot(jura50, zcol = "Rock4", col.regions = topo.colors(4),
  key.space = "right", main = "Rock type")
> print(map.rock4, split = c(2, 1, 2, 1), more = F)

```



Task 15 : Compare the KED predictions to the OK predictions, on the same scale. •

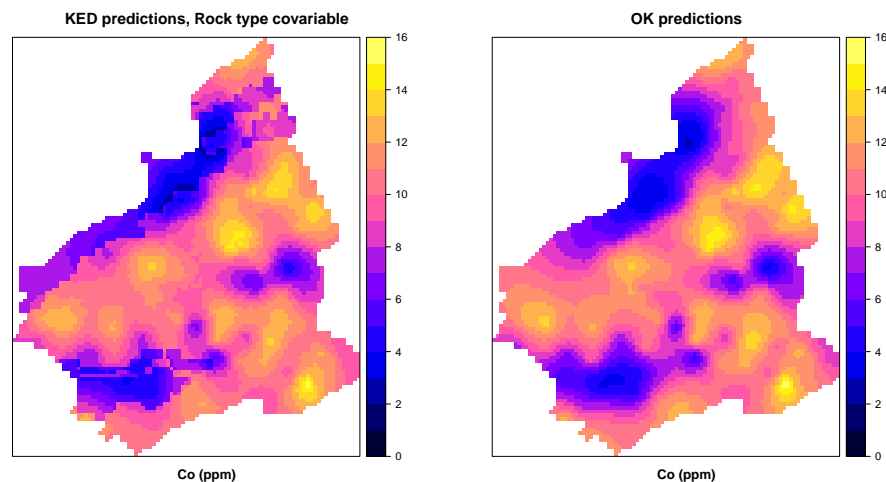
First we compute the OK predictions for this grid:

```
> k.50 <- krige(Co ~ 1, loc = jura.cal, newdata = jura50,
               model = vmf)
```

[using ordinary kriging]

Then we prepare a plot, using a common colour ramp and limits with KED. To compare side-by-side, both maps must have the same stretch; we set up a sequence with `seq` from 0 to the next greater integer, computed with `ceiling`, to the combined maximum of the predictions, computed with `max`. We then use that sequence in both maps.

```
> stretch <- seq(0, ceiling(max(k.50$var1.pred,
                                kr.50$var1.pred, na.rm = T)), by = 1)
> plot.ked <- spplot(kr.50, zcol = "var1.pred",
                    at = stretch, col.regions = bpy.colors(64),
                    main = "KED predictions, Rock type covariable",
                    sub = "Co (ppm)")
> plot.ok <- spplot(k.50, zcol = "var1.pred", at = stretch,
                   col.regions = bpy.colors(64), main = "OK predictions",
                   sub = "Co (ppm)")
> print(plot.ked, split = c(1, 1, 2, 1), more = T)
> print(plot.ok, split = c(2, 1, 2, 1), more = F)
```



Q18 : What are the principal differences between the *spatial pattern* of the two predictions? *Jump to A18* •

6.1 KED prediction variance

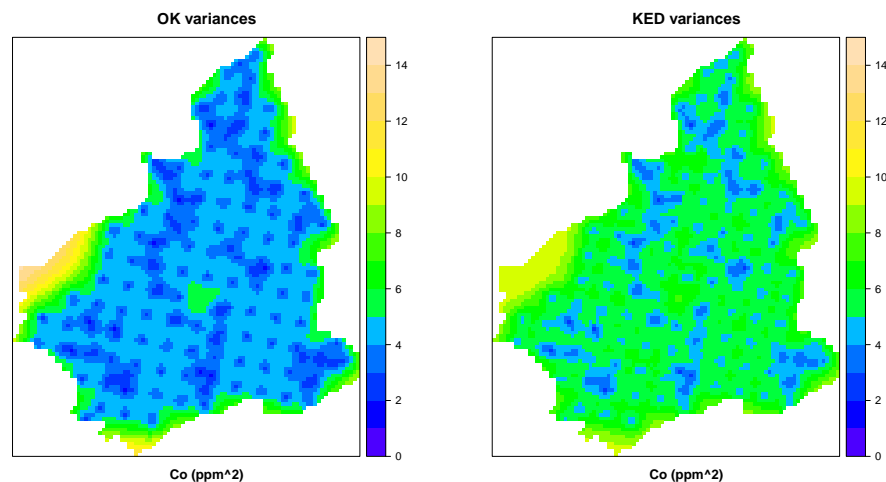
KED has produced a kriging prediction variance, which is a combination of the prediction variance of the linear model and the (simple) kriging prediction variance of the residuals from that model. However, they are solved together in the Universal Kriging system (see formulation of UK). These variances can be lower than OK variances, if the linear model explains a large amount of the variance; however, they can be higher if extrapolating into a part of feature space not covered by the observations. Also, the linear model can give different prediction variances for different classes, and these are included in the overall KED prediction variance.

Task 16 : Compare the kriging variances on the same scale. •

Again we set up a common scale:

```
> stretch <- seq(0, ceiling(max(k.50$var1.var, kr.50$var1.var,
  na.rm = T)), by = 1)
> plot.ok.v <- spplot(k.50, zcol = "var1.var", at = stretch,
  col.regions = topo.colors(64), key.space = FALSE,
  main = "OK variances", sub = "Co (ppm^2)")
> plot.ked.v <- spplot(kr.50, zcol = "var1.var",
  at = stretch, col.regions = topo.colors(64),
  key.space = "right", main = "KED variances",
  sub = "Co (ppm^2)")
> rm(stretch)

> print(plot.ok.v, split = c(1, 1, 2, 1), more = T)
> print(plot.ked.v, split = c(2, 1, 2, 1), more = F)
```



Q19 : *Is there any difference in the **spatial pattern** of the two kriging variances? In the **values**? Explain why in both cases.* Jump to A19 •

A useful way to visualize the differences is with a **difference map**.

Task 17 : Build a spatial data structure with the coordinates of the prediction grid, the differences in prediction and kriging variance between KED and OK, and the covariable (here, rock type). •

We use the `data.frame` method to build a data frame, and then convert this back to a spatial object with the `coordinates` method:

```
> diff <- data.frame(d.pred = kr.50$var1.pred -
  k.50$var1.pred, d.var = kr.50$var1.var - k.50$var1.var,
  Rock4 = jura50$Rock4)
> summary(diff)
```

d.pred		d.var		Rock4	
Min.	:-3.5	Min.	:-4.2	Argo	:1185
1st Qu.	:-0.4	1st Qu.	: 1.0	KimmPort	:2352
Median	:-0.1	Median	: 1.1	Quat	: 792
Mean	:-0.1	Mean	: 1.0	Seq	:1628
3rd Qu.	: 0.2	3rd Qu.	: 1.2	NA's	:5392
Max.	: 3.3	Max.	: 1.9		
NA's	:5392	NA's	:5392		

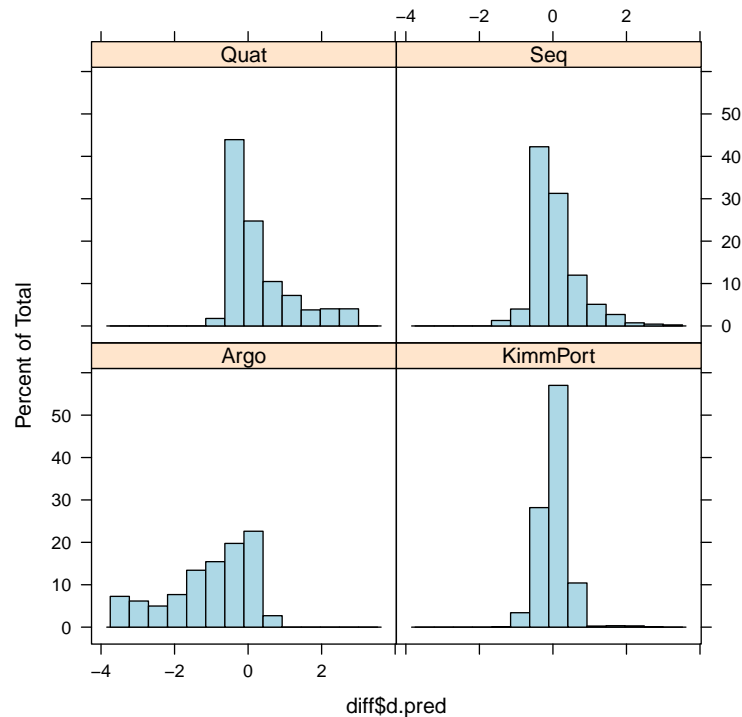
```
> coordinates(diff) <- coordinates(k.50)
> gridded(diff) <- TRUE
> fullgrid(diff) <- TRUE
```

We can visualize these differences both in feature and geographic space.

Task 18 : Display histograms of the differences in prediction for each rock type separately. •

We use the `histogram` method of the `lattice` package; like the `hist` method of the base package, this produces a histogram; but `histogram` has a **formula interface** to specify multiple plots. The so-called **conditioning variable**, which tells which categorical variable(s) to separate the plot by, is written to the right of a vertical bar `|`, and the variable to be plotted after the tilde `~`, to show this is not an x-y plot:

```
> print(histogram(~diff$d.pred | diff$Rock4, col = "lightblue"))
```



Pay close attention to the bin labels, especially the position of the zero point (i.e. zero residual). Note the differences are for $KED - OK$.

Q20 : What are the differences in the distribution of the differences for the different rock types? Jump to A20 •

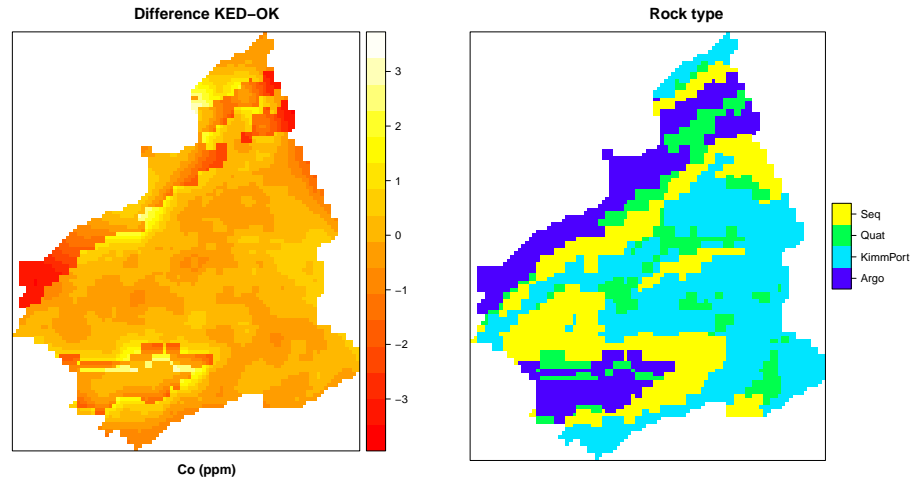
Of course we want to visualize the locations of the differences in geographic space.

Task 19 : Display a map of the prediction differences, along with the map of rock types to aid interpretation. •

Note: Since this is a different graphic variable from the other maps (kriging predictions, their variances, rock type) we use another graphic scale, the heat map, produced by the `heat.colors` function.

```
> plot.diff <- spplot(diff,
  zcol="d.pred", by=seq(-3.5, 3.5, by=0.5),
```

```
col.regions=heat.colors(64), key.space="right",
main="Difference KED-OK", sub="Co (ppm)")
> print(plot.diff, split=c(1,1,2,1), more=T)
> print(map.rock4, split=c(2,1,2,1), more=F)
```



Q21 : Where are the greatest differences (positive and negative) in the predictions? What accounts for these? *Jump to A21* •

Finally, we examine the differences in kriging variance.

Task 20 : Summarize the differences in kriging variance. •

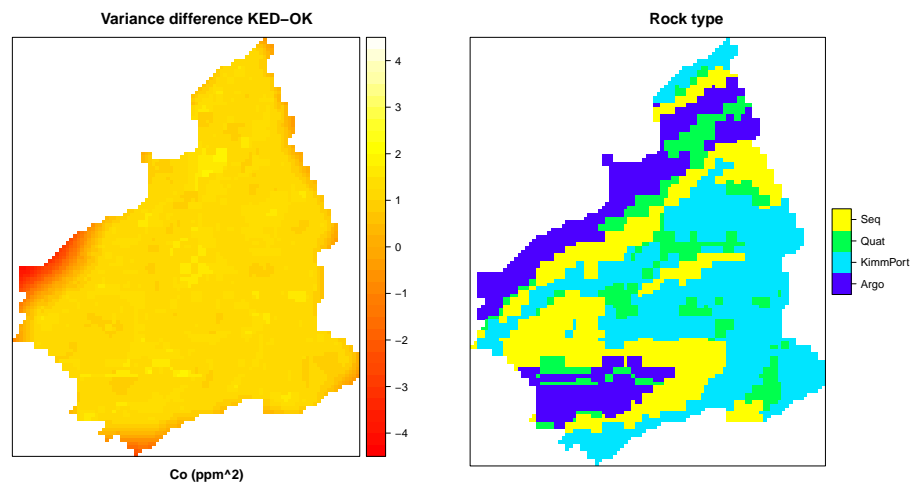
```
> summary(diff$d.var)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
-4.2	1.0	1.1	1.0	1.2	1.9	5392

Task 21 : Display a map of the differences in kriging variance, along with the map of rock types to aid interpretation. •

After some experimentation, the `bpy.colors` colour ramp is chosen for visualization:

```
> plot.diff.v <- spplot(diff, zcol = "d.var", at = seq(-4.5,
4.5, by = 0.25), col.regions = heat.colors(64),
key.space = "right", main = "Variance difference KED-OK",
sub = "Co (ppm^2)")
> print(plot.diff.v, split = c(1, 1, 2, 1), more = T)
> print(map.rock4, split = c(2, 1, 2, 1), more = F)
```



Q22 : Where are the largest differences (positive and negative) in the kriging variance? What accounts for these? [Jump to A22](#) •

6.2 Answers for Kriging with external drift

A17 : There are clear traces of the rock type, especially Argovian, which has substantially lower values than the others. [Return to Q17](#) •

A18 : The OK maps is quite smooth (continuous); the KED map shows discontinuities at boundaries between rock types (especially the Argovian with others). [Return to Q18](#) •

A19 : The spatial pattern in both is clearly related to the distribution of the observations, because all of the OK, and part of the KED, variance depends only on sample point configuration. However, for KED there is also the effect of the covariable, here the rock type. In §3.4 we saw that the mean of the Sequanian/Portlandian rock type values. The variances are lower for KED in areas with no observations, because the linear model explains much of the variability by the rock type. However, in the interior of the prediction area, with many points, the OK variances are lower. [Return to Q19](#) •

A20 : The differences are minimal and symmetric in the Kimmeridgian/Portlandian; OK and KED agree closely. The Argovian has consistently negative differences, i.e. $OK > KED$. This is because the class mean for Argovian is much lower than for the others, and KED takes this into account. The opposite is the case for the Quaternary. [Return to Q20](#) •

A21 : The biggest negative differences (i.e., OK gives higher predictions than KED) are associated with the Argovian rocks. This is because of the different mean concentrations of the rock types. Recall from the linear model \mathbf{m}_r that the other rock

types all had much higher estimates than Argovian. The largest positive differences (i.e., KED gives higher predictions than OK) can be found in all the other rock types, where they border the Argovian rocks. This is because only the residuals from the nearby Argovian observations are used to (partially) adjust the rock type mean, rather than the low-valued original Argovian observations. [Return to Q21](#)

•

A22 : The KED variances tend to be higher overall, i.e., the difference KED–OK tends to be positive, over much of the area about 1 to 1.5 mg kg⁻¹². This is because the standard errors of the linear model coefficients introduce some uncertainty. But, towards the edges OK has higher variance, especially in the area of Argovian rocks at the extreme W, because the standard errors of the linear model coefficients used in KED are small, and the residual variogram has a lower sill (i.e., maximum semi-variance) than the ordinary variogram. Although this edge is far from observations, the lower sill keeps the prediction variance lower, and this is only partly offset by the linear model prediction variance. At a few interior locations KED is much less precise than OK (see yellow spots); these mostly correspond to the Quaternary rocks, with high standard error of the linear model coefficients. [Return to Q22](#)

7 Validation

Both OK and KED can be validated with the 100 extra points in `jura.val` and cross-validated with the 259 points in the sample set `jura.cal`, just as we did for OK; these techniques will be introduced in Exercise 6 §2 and §3, respectively. You can then try them on the KED prediction to assess its quality.

8 Cleaning up

Task 22 : Remove the temporary variables from the workspace, leaving the kriged objects (OK, KED) for the next section. •

```
> rm(map.rock, map.land)
> rm(plot.ok, plot.ked, plot.ok.v, plot.ked.v)
> rm(plot.diff, plot.diff.v)
> ls()

[1] "diff"      "jura.all"   "jura.cal"   "jura.raster"
[5] "jura.val"   "jura50"     "k.50"       "k.grid"
[9] "k.val"      "kr.50"      "map.rock4"  "ml"
[13] "mr"         "mrl"        "mrli"       "v"
[17] "vmf"        "vr"         "vrml"       "vrmlf"
```

Task 23 : Save the workspace from this exercise; it will be used in the self-test. •

```
> save.image(file = "JuraEx5a.RData")
```

9 *Stratified kriging

As shown in §3, different strata may have different mean levels of a target variable. In §4 we showed how to remove this mean feature-space effect from a spatially-distributed variable and model the residuals; in §6 we showed how to then use this variogram model, along with the feature-space linear model of means separation, to predict the target variable as a combination of the feature-space effect (strata) and residual spatial structure; this is Kriging with External Drift (KED).

However, there is no inherent reason that the **structure** of spatial variability within different strata should be the same, whether or not the **means** are different. For example, on a steep hillslope the depth to a root-restricting layer may be not only less (shallower soils), but also variable at shorter ranges, and perhaps more variable overall (higher sill) than the depth on a gently-sloping area. Even the variogram form may differ between strata. If these contrasting areas are delineated as strata, it may be profitable to model their spatial variability separately, use these models to map by ordinary kriging, and finally combine the maps of each stratum into a single map. This process is called **stratified kriging** (StK) [3].

A further consideration is the size and shape of individual polygons of the strata. In StK, each point to be predicted is only kriged from points in the same stratum. Since kriging is a local predictor, points in the stratum but in scattered polygons will typically not get much kriging weight (of course, that depends on the variogram range). So the larger each polygon, the more nearby known points there will be for a point to be predicted.

This approach is not much used (one example is [3]), mainly because of the requirement to model per-stratum variograms. If the number of points is limited, reducing them further by stratification makes variogram model estimation difficult. However, this is a useful approach to keep in the geostatistician's toolkit for those situations where (1) the spatial structure differs between strata, (2) there are enough points to estimate per-stratum variograms, (3) the individual polygons of the strata are large enough to cover most of the range of the per-stratum variogram.

This **optional** section shows how to implement StK using the Jura dataset. This dataset is not ideal for StK, as we will see, but does allow us to illustrate the method and some of its difficulties.

9.1 Determining the stratification

There are two categorical variables which could be used for stratification: rock type and land use. Both of these were simplified for feature-space modelling (§3) and Kriging with External Drift (§6)

We now decide which factor to use for stratification.

Q23 : *What are the possible stratifying factors? How many levels does each have? Which one showed the largest difference in mean values of the*

Co concentration (see §3)? Which one showed the largest difference in spread (as an indicator of variability)? [Jump to A23](#) •

We use the `by` function to apply a function (here, `length`, `mean` and `var`) to a data frame split by factors. This computes the function for each level. We then use the `cbind` “bind by columns” function to make a nice table for display.

```
> levels(jura.cal$Land3)

[1] "Forest" "Meadow" "PasTill"

> levels(jura.cal$Rock4)

[1] "Argo"      "KimmPort" "Quat"      "Seq"

> tmp <- cbind(by(jura.cal$Co, jura.cal$Rock4, length),
               by(jura.cal$Co, jura.cal$Rock4, mean), by(jura.cal$Co,
               jura.cal$Rock4, var))
> colnames(tmp) <- c("n", "mean", "variance")
> print(tmp)

      n    mean variance
Argo   53  5.3932   4.2762
KimmPort 88 10.9876   7.8654
Quat   55  9.5955  18.5529
Seq    63  9.9752   4.8344

> tmp <- cbind(by(jura.cal$Co, jura.cal$Land3, length),
               by(jura.cal$Co, jura.cal$Land3, mean), by(jura.cal$Co,
               jura.cal$Land3, var))
> colnames(tmp) <- c("n", "mean", "variance")
> print(tmp)

      n    mean variance
Forest 33  7.6833  10.5194
Meadow 165 9.3935  14.8521
PasTill 61 9.9256   6.9569
```

Task 24 : Decide on a stratifying factor. •

We choose to use the rock type to define strata, because, although it has one more class, it promises to have different variogram sills (i.e., total variability). We do not yet know if the variogram model, range, or nuggets will be different.

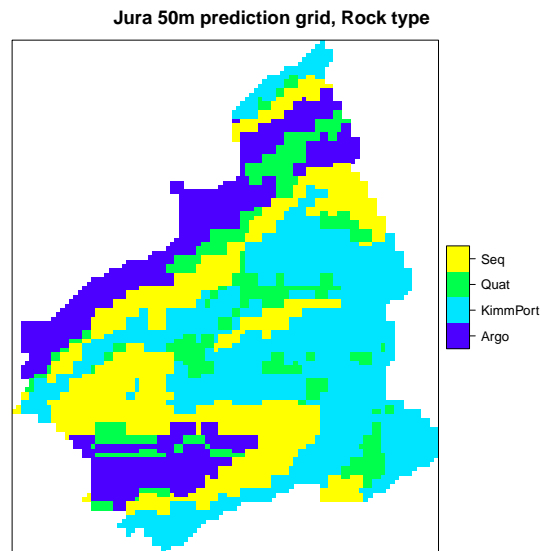
Q24 : Explain how a different sedimentary rock type might have different spatial structure for a trace metal such as Co. [Jump to A24](#) •

A second problem is the spatial continuity and size of individual polygons.

Task 25 : Examine the prediction grid of rock types, which was created in

§5.1 and simplified in §6. •

```
> print(splot(jura50, zcol="Rock4", col.regions=topo.colors(4),
             key.space="right",
             main="Jura 50m prediction grid, Rock type"))
```



Q25 : Which classes have reasonably large and compact polygons? *Jump to A25* •

Although the Quaternary has scattered small polygons, its spatial structure promises to be different from the others, so we do not eliminate it.

9.2 Per-stratum variograms

The first step in Stratified Kriging is to model the spatial structure within each stratum.

Task 26 : Compute the empirical variogram for each of the four strata. •

Recall the the fitted model of the ordinary variogram (all strata, original values) in a previous exercise was pentaspherical with a range of about 1.5 km. This range could be the cutoff for computing the stratified variogram. However, the polygons of the strata are smaller, as can be appreciated from the map of rock types above; thus much of the longer-range dependence comes from the inter-polygon similarity. When kriging within a polygon the short-range structure is most important; therefore we limit the cutoff of the empirical variogram.

For efficiency and clarity, we will do the variogram computation, variogram modelling, kriging, and printing in parallel for the four strata, using R **lists**. These are arbitrary collections of R objects, joined together by the **list** function.

Note: They are also the result of many R functions which return hybrid data structures; see for example objects of class `lm`, returned by the `lm` and some other functions.

The first step is to split the dataset into the four strata, using the `split` function and the just-created `Rock4` field as the factor on which to split. This function only works on dataframes with a factor field; since we have a `SpatialPointsDataFrame` object, we must first convert to a dataframe with the `as.data.frame` function, split that dataframe with `split`, and recover the coordinates with the `coordinates` method, to finally have a list of four `SpatialPointsDataFrame` objects:

```
> jura.cal.split <- split(as.data.frame(jura.cal),
  jura.cal$Rock4)
> for (i in 1:4) coordinates(jura.cal.split[[i]]) <- c("X",
  "Y")
```

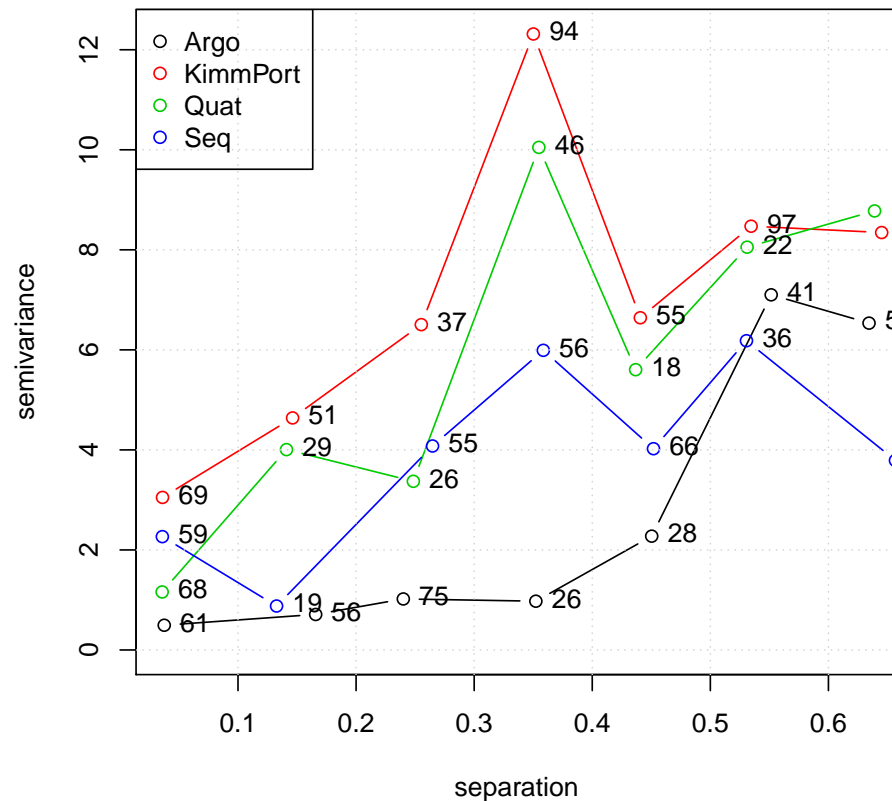
Now compute the four variograms, and display them on one graph, with the number of point-pairs per bin:

```
> v <- NULL; for (i in 1:4) {
>   v[[i]] <- variogram(Co ~ 1, loc=jura.cal.split[[i]],
  cutoff=0.7, width=0.1 );
> ylim <- c(0, max(unlist(lapply(v, function(x) max(x$gamma)))))
> plot(v[[1]]$gamma ~ v[[1]]$dist, type="n", ylim=ylim,
  xlab="separation", ylab="semivariance",
  main="Empirical variogram, Co, per stratum")
> for (i in 1:4) lines(v[[i]]$dist, v[[i]]$gamma, type="b", col=i)
> grid()
> for (i in 1:4) text(v[[i]]$dist, v[[i]]$gamma, v[[i]]$np, pos=4)
> names(jura.cal$Rock4)

NULL

> legend("topleft", levels(jura.cal$Rock4), pch=1, col=1:4)
```


Empirical variogram, Co, per stratum



Note: The `unlist` function is used to return the results of the `lapply` “apply a function to a list” function as a vector, suitable for the `max` function.

Q26 : Do the four variograms have the same shape, total sill, nugget, and range? If not, describe the differences. Jump to A26 •

With few point-pairs per variogram bin, and irregular shape, the automatic fit of `fit.variogram` may not be reliable. We first model the short-range structure by eye and then compare the automatic fit. This shows a weakness of stratified kriging: modelling variograms in each stratum, often with not much evidence.

Task 27 : Fit the short-range part of each variogram with a reasonable model. •

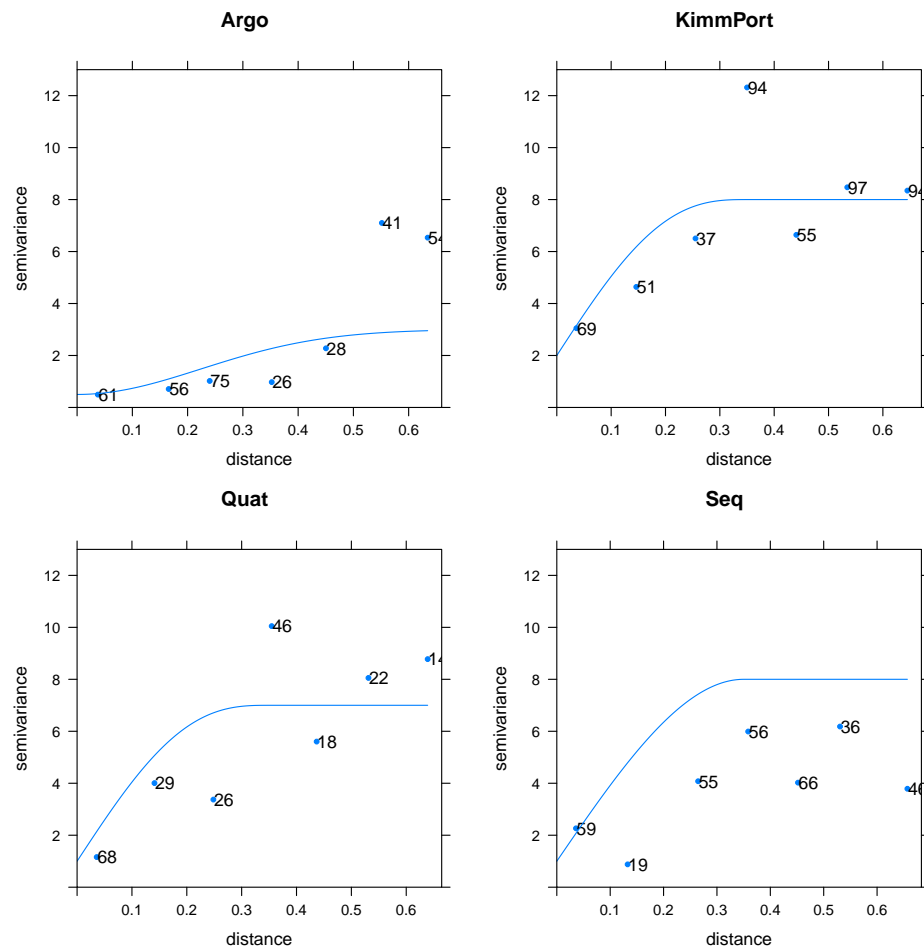
We first define a list of starting models from visual inspection of the empirical variograms on the above plot:

```
> start.model <- list(vgm(2.5, "Gau", 0.55/sqrt(3), 0.5),
                      vgm(6, "Pen", 0.35, 2),
```

```
vgm(6, "Pen", 0.35, 1),
vgm(7, "Sph", 0.35, 1))
```

We then display these starting models on the empirical variograms, using the `print` method of the `lattice` package; recall we can place several lattice plots on one page with the `split` and `more` arguments:

```
> p <- NULL
> for (i in 1:4) {
  p[[i]] <- plot(v[[i]], plot.numbers=T, pch=20,
    model=start.model[[i]],
    main=levels(jura.cal$Rock4)[i], ylim=c(0,13)) }
> print(p[[1]], split=c(1,1,2,2), more=T)
> print(p[[2]], split=c(2,1,2,2), more=T)
> print(p[[3]], split=c(1,2,2,2), more=T)
> print(p[[4]], split=c(2,2,2,2), more=F)
```



We then fit them with `fit.variogram`'s default weighted least squares fit:

```
> for (i in 1:4) {
  vmf[[i]] <- fit.variogram(v[[i]], start.model[[i]])
  print(vmf[[i]])
}

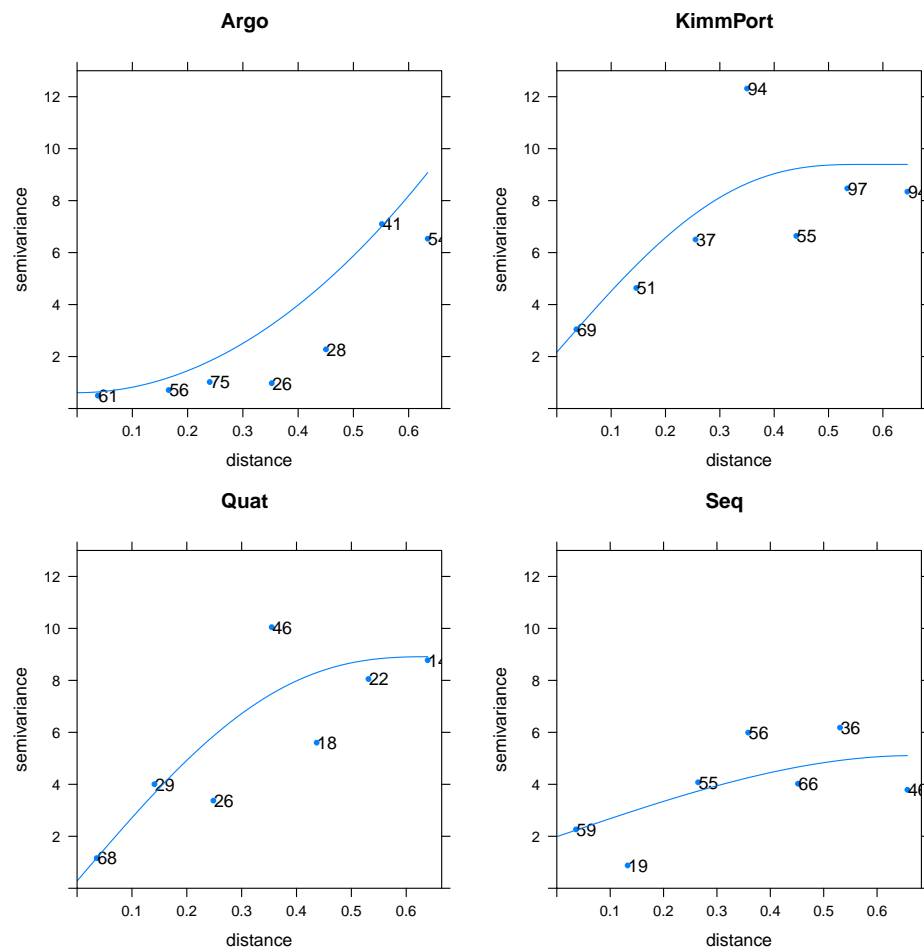
model      psill  range
```

```

1  Nug    0.60884 0.0000
2  Gau 1385.65454 8.1018
  model psill range
1  Nug 2.1661 0.00000
2  Pen 7.2267 0.56527
  model psill range
1  Nug 0.2798 0.00000
2  Pen 8.6325 0.65518
  model psill range
1  Nug 1.9853 0.000
2  Sph 3.1207 0.669

> p <- NULL
> for (i in 1:4) {
  p[[i]] <- plot(v[[i]], plot.numbers=T, pch=20,
    model=vmf[[i]], main=levels(jura.cal$Rock4)[i],
    ylim=c(0,13)) }
> print(p[[1]], split=c(1,1,2,2), more=T)
> print(p[[2]], split=c(2,1,2,2), more=T)
> print(p[[3]], split=c(1,2,2,2), more=T)
> print(p[[4]], split=c(2,2,2,2), more=F)

```



Q27 : *Comment on the fitted variograms. Are they reasonable to use for kriging?* *Jump to A27 •*

Task 28 : Predict by kriging within each stratum, each with its own model: •

Note how each stratum's points are selected as the known points (the `loc` argument) and the same stratum's grid cells are selected to be predicted (the `newdata` argument). Thus, each grid cell is predicted from only points in its same stratum, ignoring even close-by known points in other strata.

```
> (k <- list(NULL, NULL, NULL, NULL))
> for (i in 1:4) {
  k[[i]] <- krige(Co ~ 1,
                  loc=jura.cal[as.numeric(jura.cal$Rock4) == i,],
                  newdata=jura50[as.numeric(jura50$Rock4) == i,],
                  model=vmf[[i]], nmax=36) ;
  gridded(k[[i]]) <- T
}
```

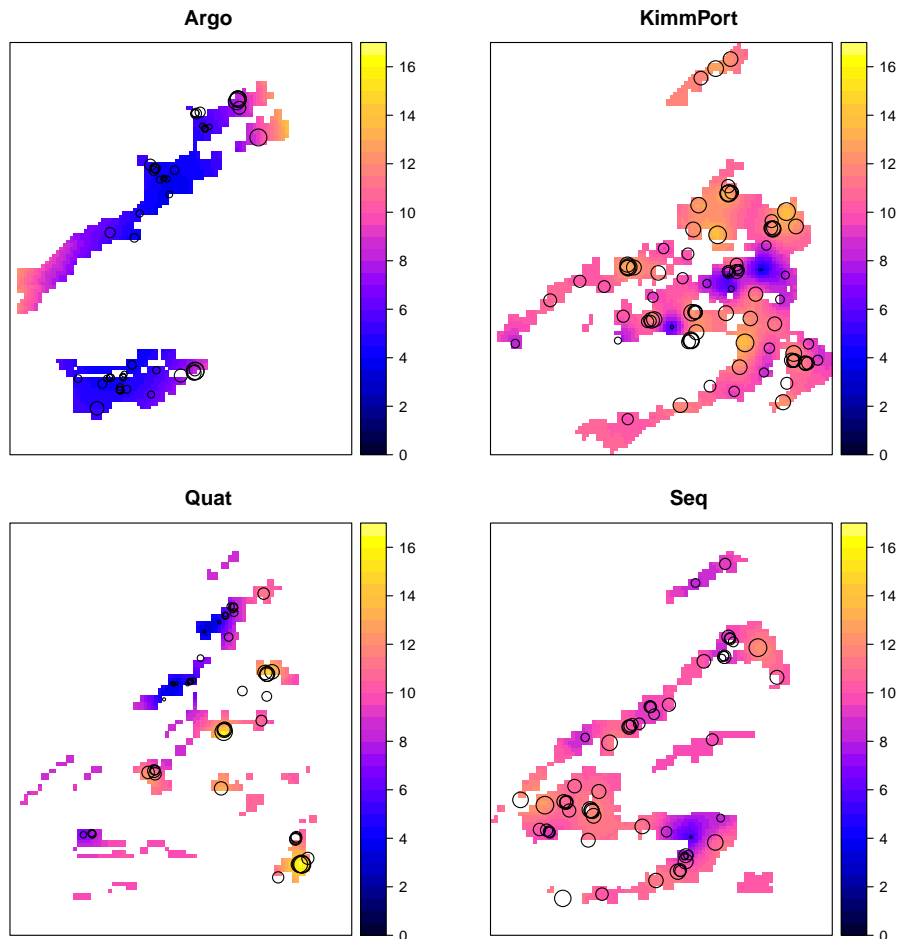
Task 29 : Display the per-stratum kriging predictions side-by-side, on the same scale •

```
> zlim <- ceiling(max(unlist(lapply(k, function(x) max(x$var1.pred,
  na.rm = T))))))
> p <- NULL
> for (i in 1:4) {
  pts <- jura.cal[as.numeric(jura.cal$Rock4) ==
    i, ]
  layout.1 <- list("sp.points", pts, pch = 1,
    col = "black", cex = 2 * pts$Co/max(pts$Co))
  p[[i]] <- spplot(k[[i]], zcol = "var1.pred",
    col.regions = bpy.colors(64), at = seq(0,
    zlim, by = 0.5), main = levels(jura.cal$Rock4)[i],
    xlim = bbox(jura50)["X", ], ylim = bbox(jura50)["Y",
    ], sp.layout = list(layout.1))
}
```

```

> print(p[[1]], split=c(1,1,2,2), more=T);
> print(p[[2]], split=c(2,1,2,2), more=T);
> print(p[[3]], split=c(1,2,2,2), more=T)
> print(p[[4]], split=c(2,2,2,2), more=F)

```



Task 30 : Display the per-stratum kriging prediction variances side-by-side, on the same scale. •

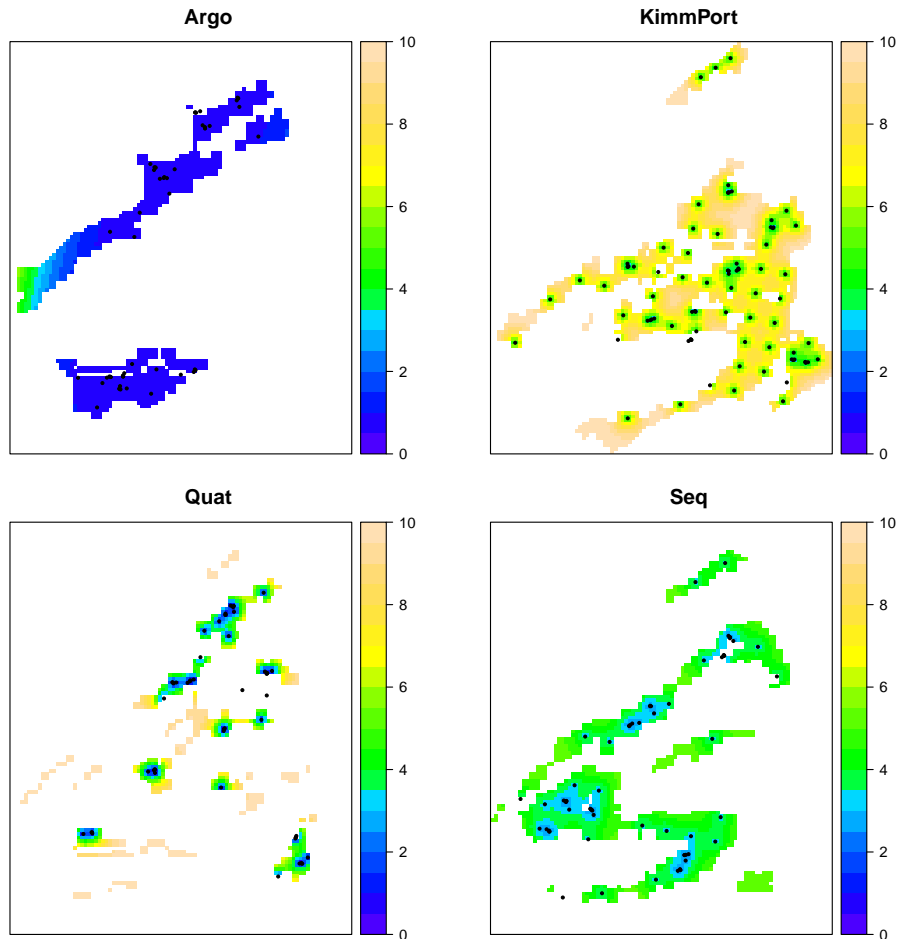
```

> zlim <- ceiling(max(unlist(lapply(k, function(x) max(x$var1.var,
  na.rm = T))))))
> p <- NULL
> for (i in 1:4) {
  pts <- jura.cal[as.numeric(jura.cal$Rock4) ==
    i, ]
  layout.2 <- list("sp.points", pts, pch = 20,
    col = "black", cex = 0.5)
  p[[i]] <- spplot(k[[i]], zcol = "var1.var",
    col.regions = topo.colors(64), at = seq(0,
    zlim, by = 0.5), main = levels(jura.cal$Rock4)[i],
    xlim = bbox(jura50)["X", ], ylim = bbox(jura50)["Y",
    ], sp.layout = list(layout.2))

```

}

```
> print(p[[1]], split=c(1,1,2,2), more=T);  
> print(p[[2]], split=c(2,1,2,2), more=T);  
> print(p[[3]], split=c(1,2,2,2), more=T)  
> print(p[[4]], split=c(2,2,2,2), more=F)
```



Task 31 : Combine the four prediction grids into one. •

We use the `rbind` method to concatenate the four maps; however, this works on dataframes, so we first convert the `SpatialGridDataFrame` objects into `SpatialPointsDataFrame` objects, using the generic `as` function. We then combine them, and convert back to `SpatialGridDataFrame`.

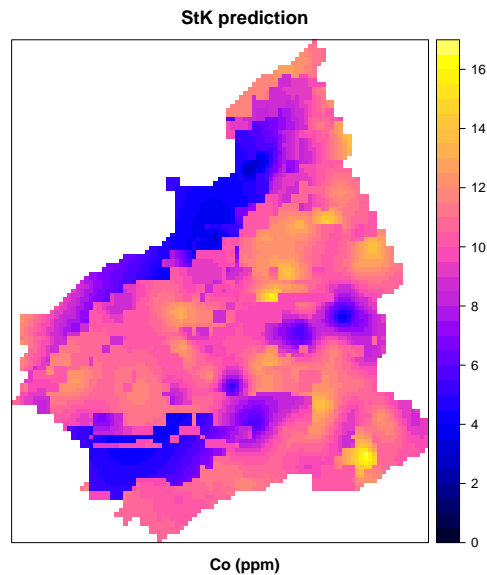
```
> for (i in 1:4) k[[i]] <- as(k[[i]], "SpatialPointsDataFrame")  
> k.st <- rbind(k[[1]], k[[2]], k[[3]], k[[4]])  
> gridded(k.st) <- T  
> fullgrid(k.st) <- T
```

Task 32 : Print the combined predictions. •

```

> plot.stk <- spplot(k.st, zcol="var1.pred",
  col.regions=bpy.colors(64),
  at=seq(0, ceiling(max(k.st$var1.pred, na.rm=T)),
    by=0.5),
  main="StK prediction", sub="Co (ppm)"
)
> print(plot.stk)

```



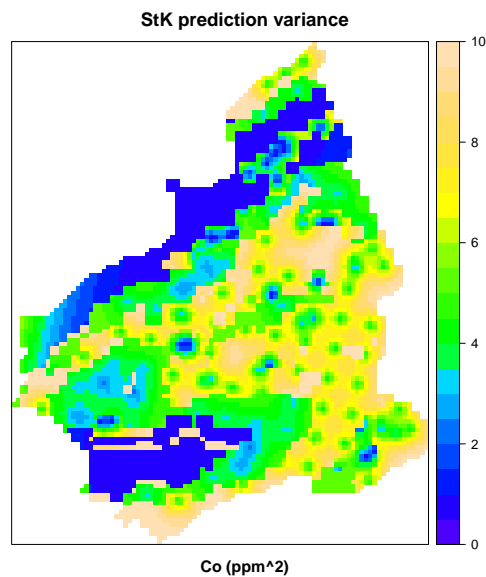
Q28 : Describe and explain the features of the StK predictions. [Jump to A28](#) •

Task 33 : Print the combined prediction variances. •

```

> plot.stk.v <- spplot(k.st, zcol="var1.var",
  col.regions=topo.colors(64),
  at=seq(0, ceiling(max(k.st$var1.var, na.rm=T)),
    by=0.5),
  key.space="right",
  main="StK prediction variance", sub="Co (ppm^2)"
)
> print(plot.stk.v)

```



Q29 : Describe and explain the features of the StK prediction variances. [Jump to A29](#) •

Task 34 : Compare the OK, KED, and StK predictions side-by-side. •

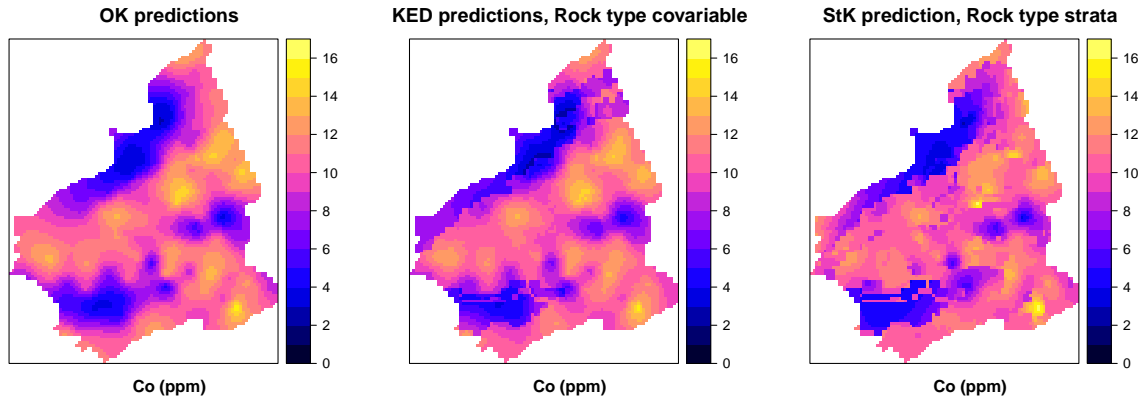
To compare the different predictions visually, they must be on the same visual scale (“stretch”), so we compute the common scale and then repeat the plotting commands.

```
> stretch <- seq(0, ceiling(max(k.50$var1.pred,
                                kr.50$var1.pred,
                                k.st$var1.pred, na.rm=T)),
                  by=1)
> plot.ok <- spplot(k.50, zcol="var1.pred", at=stretch,
                   col.regions=bpy.colors(64),
                   main="OK predictions",
                   sub="Co (ppm)")
> plot.ked <- spplot(kr.50, zcol="var1.pred", at=stretch,
                   col.regions=bpy.colors(64),
                   main="KED predictions, Rock type covariable",
                   sub="Co (ppm)")
> plot.stk <- spplot(k.st, zcol="var1.pred", at=stretch,
                   col.regions=bpy.colors(64),
                   main="StK prediction, Rock type strata",
                   sub="Co (ppm)"
                   )
```

```

> print(plot.ok, split = c(1, 1, 3, 1), more = T)
> print(plot.ked, split = c(2, 1, 3, 1), more = T)
> print(plot.stk, split = c(3, 1, 3, 1), more = F)

```



Task 35 : Compare the OK, KED, and StK prediction variances side-by-side. •

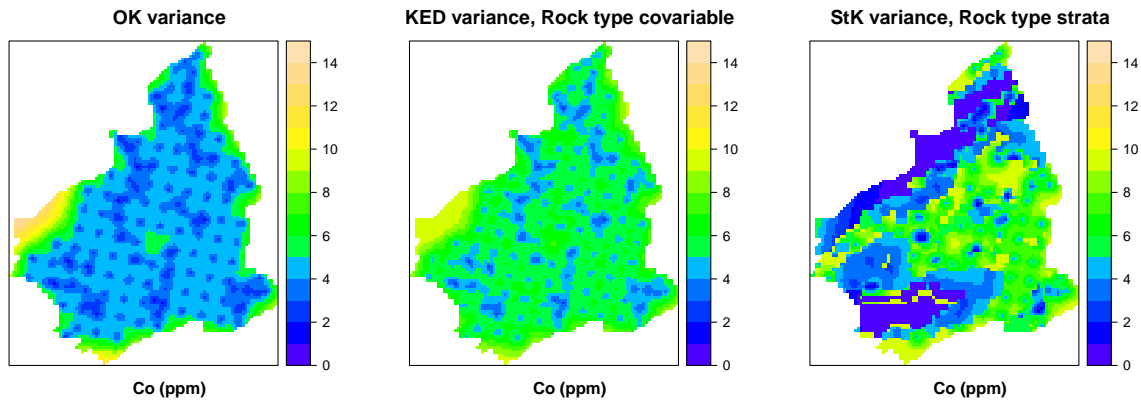
Again we compute the visual scale and display the variances on that scale:

```

> stretch <- seq(0, ceiling(max(k.50$var1.var, kr.50$var1.var,
                                k.st$var1.var, na.rm=T)),
                  by=1)
> plot.ok.v <- spplot(k.50, zcol="var1.var", at=stretch,
                      col.regions=topo.colors(64),
                      main="OK variance",
                      sub="Co (ppm)")
> plot.ked.v <- spplot(kr.50, zcol="var1.var", at=stretch,
                      col.regions=topo.colors(64),
                      main="KED variance, Rock type covariable",
                      sub="Co (ppm)")
> plot.stk.v <- spplot(k.st, zcol="var1.var",
                      col.regions=topo.colors(64), at=stretch,
                      main="StK variance, Rock type strata",
                      sub="Co (ppm)"
                      )

```

```
> print(plot.ok.v, split = c(1, 1, 3, 1), more = T)
> print(plot.ked.v, split = c(2, 1, 3, 1), more = T)
> print(plot.stk.v, split = c(3, 1, 3, 1), more = F)
```



Challenge: Attempt stratified kriging with land use as a stratifying factor.

Task 36 : Remove the temporary variables from the workspace. •

```
> rm(plot.ok, plot.ked, plot.ok.v, plot.ked.v, plot.stk,
     plot.stk.v, stretch)
> rm(k.st, k, p, v, start.model, vmf, jura.cal.split,
     ylim, zlim, layout.1, layout.2)
```

9.3 * Answers for Stratified Kriging

A23 : The two stratifying factors are (simplified) land use (field `Land3`), with three levels, and (simplified) rock type (field `Rock`), with four. In §3 we showed that rock type explained much more variation in Co concentration than land use, 0.32 vs. 0.03. The boxplot in that section shows that the difference in spreads is also wider in rock type: Quaternary has a much wider spread than the other rock types; whereas for land uses the spreads are similar, although the pasture/tillage land use has about half the variance of the other land uses. [Return to Q23](#) •

A24 : The depositional environments in which the sedimentary layers were formed may have had different spatial structure. For example, a limestone may be from very deep, still water, or from shallower, more turbid water; the former would have longer-range spatial structure. In the present case the Quaternary geology is much more recent than the four late Jurassic limestones (approximately 150 Mya), and may result from a shorter-range process, or one with more nugget variance. [Return to Q24](#) •

A25 : No rock type has large, compact polygons; the closest to that ideal are Kimmeridgian/Portlandian and Sequanian, but these both have some SW-NW elongated polygons. The same directional trend is clear in the Argovian. The

Portlandian and Quaternary both have only small, scattered polygons. [Return to Q25](#) •

A26 : The shapes differ substantially: a Gaussian shape seems best for Argovian (strong spatial continuity at the origin), whereas the other three have similar variograms, which could be approximated by a pentaspherical model. The few point-pairs per bin makes for erratic empirical variograms.

The total sill for the Kimmeridgian/Portlandian and Quaternary are the highest; Sequanian and Argovian reach similar sills but with different model forms.

The nuggets also differ, with that for Argovian lower than the others, about $0.5 \text{ (mg kg}^{-1}\text{)}^2$ vs. 1 to 3 $\text{(mg kg}^{-1}\text{)}^2$.

The ranges are roughly similar, about 0.35 km, except for Argovian which appears longer-range, about 0.55 km. [Return to Q26](#) •

A27 : Given the small number of points and the irregular variograms, these fits look reasonable. The extremely high sill and long range of the Gaussian model fit are due to the bin at about 0.55 km separation. [Return to Q27](#) •

A28 : The Argovian areas have the lowest values; the strata are quite recognisable, e.g., the Quaternary within the Argovian in the SW corner. [Return to Q28](#) •

A29 : The Argovian areas have very low prediction variances, because of (1) the low nugget and (2) the strong spatial continuity of the Gaussian model. The Kimmeridgian/Portlandian areas have a bit lower variance, because of the somewhat lower nugget; otherwise the variance is lowest near observation points. [Return to Q29](#) •

10 Self-test

This section is a small self-test of how well you mastered this exercise. You should be able to complete the tasks and answer the questions with the knowledge you have gained from the exercise. Please **submit your answers (including graphical output) to the instructor** for grading and sample answers.

Task 1 : Compute the residual variogram using simplified (3-class) land use as the classifying factor. Plot the variogram, next to the ordinary variogram. •

Task 2 : Model the residual variogram using land use as the classifying factor. Plot the modelled variogram. Compare its parameters to the ordinary variogram from Exercise 4 §4.1. •

Q1 : How much and how does this differ from the ordinary variogram? •

Q2 : *How much does this differ from the residual variogram using rock type as the classifying factor?* •

Q3 : *Explain why the residual variogram from land use does not differ very much from the ordinary variogram, whereas that using rock type is quite different.* •

Task 3 : Predict the Co concentration over the 50 m grid, using kriging with external drift (KED), with simplified land use as the feature-space predictor. View the predictions and the kriging variances. •

Q4 : *Can you see any evidence of the land use map in the prediction map? If so, where and why?* •

Q5 : *Can you see any evidence of the land use map in the prediction variances map? If so, where and why?* •

Task 4 : Compare the KED predictions to the OK predictions, on the same scale. •

Q6 : *What are the principal differences between the **spatial pattern** of the two predictions? Is there a greater or lesser difference using land use as the feature-space predictor vs. using rock type? Why?* •

Task 5 : Compare the KED prediction variance to the OK prediction variance, on the same scale. •

Q7 : *What are the principal differences between the **spatial pattern** of the two prediction variances? Is there a greater or lesser difference using land use as the feature-space predictor vs. using rock type? Why?* •

Task 6 : Clean up the workspace. •

References

- [1] J Fox. *An R and S-PLUS Companion to Applied Regression*. Sage, Newbury Park, 2002. 5
- [2] P Goovaerts. *Geostatistics for natural resources evaluation*. Applied Geostatistics. Oxford University Press, New York; Oxford, 1997. 14
- [3] A. Stein, M. Hoogerwerf, and J. Bouma. Use of soil-map delineations to improve (co)kriging of point data. *Geoderma*, 43:163–177, 1988. 27

Index of R Concepts

`<-` operator, 6

`anova`, 9

`as`, 36

`as.data.frame`, 30

`as.factor`, 4, 18

`boxplot`, 5

`bpy.colors` (sp package), 24

`by`, 28

`car` package, 5

`cbind`, 28

`ceiling`, 11, 20

`coordinates` (sp package), 22, 30

`data.frame`, 22

`fit.variogram` (gstat package), 12, 31, 32

`gstat` package, 2, 11, 12, 14

`heat.colors`, 23

`hist`, 23

`histogram` (lattice package), 23

`jura` dataset, 14

`krige` (gstat package), 18

`lapply`, 31

`lattice` package, 23, 32

`length`, 28

`list`, 29

`lm`, 6, 11, 30

`lm` class, 30

`load`, 2, 14

`loc gstat` argument, 34

`max`, 20, 31

`mean`, 28

`more lattice` graphics argument, 32

`newdata gstat` argument, 34

`notch` argument (boxplot function), 5

`print` (lattice package), 32

`rbind`, 15, 36

`recode` (car package), 5

`seq`, 20

`setwd`, 2

`sp` package, 2

`SpatialGridDataFrame` class, 17, 36

`SpatialPointsDataFrame` class, 30, 36

`split`, 30

`split` lattice graphics argument, 32

`spplot` (sp package), 15

`summary`, 6

`summary.lm`, 6

`table`, 3, 15

`unlist`, 31

`var`, 28

`variogram` (gstat package), 11

`varwidth` argument (boxplot function), 5

`vgm` (gstat package), 12