Cran_Klovan_Final

Jeffrey Yarus

2024-07-25

```
#loading data
data("Klovan_Row80", package = "klovan")
data("Klovan_2D_all_outlier", package = "klovan")

#apply a range transform to your data
klovan <- klovan::range_transform(Klovan_Row80)
```

- Eigenvectors: also called Principal Components
- Eigenvalues: the factor by which the eigenvector is scaled

```
cov_mtrx <- klovan::covar_mtrx(klovan)
cov_mtrx
```

```
##                   rank      P_Elong         P_Fe      P_Fold     P_Fract
## rank       0.103192043 -0.003208257  0.022142820 -0.01090080 -0.02441241
## P_Elong   -0.003208257  0.063067292  0.049123140  0.05301063  0.03890943
## P_Fe       0.022142820  0.049123140  0.089174087  0.02272592  0.02504509
## P_Fold    -0.010900797  0.053010631  0.022725920  0.05138066  0.03326994
## P_Fract   -0.024412411  0.038909429  0.025045086  0.03326994  0.06031784
## P_Mg       0.024037709  0.045108134  0.086553127  0.01928991  0.01925962
## P_Na       0.021634161  0.050110354  0.086658753  0.02481922  0.02422390
## P_Space   -0.013641520  0.047812205  0.010888847  0.04990160  0.02952942
## P_Sulfide  0.008976645  0.063799741  0.084928536  0.04044762  0.04438451
## P_Veins   -0.025773905  0.017171926  0.007238515  0.01501100  0.05540251
## P_XLSize  -0.036426253  0.010915760 -0.043091477  0.02726559  0.03105227
##                   P_Mg         P_Na      P_Space    P_Sulfide      P_Veins
## rank       0.024037709  0.021634161 -0.013641520  0.008976645 -0.025773905
## P_Elong    0.045108134  0.050110354  0.047812205  0.063799741  0.017171926
## P_Fe       0.086553127  0.086658753  0.010888847  0.084928536  0.007238515
## P_Fold     0.019289909  0.024819218  0.049901599  0.040447625  0.015011001
## P_Fract    0.019259624  0.024223904  0.029529417  0.044384511  0.055402511
## P_Mg       0.084689809  0.084102957  0.007827682  0.080427478  0.002092618
## P_Na       0.084102957  0.084460032  0.013469888  0.083227110  0.005727341
## P_Space    0.007827682  0.013469888  0.050093917  0.029474281  0.012896789
## P_Sulfide  0.080427478  0.083227110  0.029474281  0.091029451  0.023803588
## P_Veins    0.002092618  0.005727341  0.012896789  0.023803588  0.060461400
## P_XLSize  -0.046032117 -0.040531637  0.033547435 -0.020059364  0.032605614
##              P_XLSize
## rank      -0.03642625
## P_Elong    0.01091576
## P_Fe      -0.04309148
## P_Fold     0.02726559
## P_Fract    0.03105227
## P_Mg      -0.04603212
## P_Na      -0.04053164
## P_Space    0.03354743
## P_Sulfide -0.02005936
## P_Veins    0.03260561
## P_XLSize   0.06426804
```

```
#calulate Eiegn values
klovan::calc_eigenvalues(cov_mtrx)
```

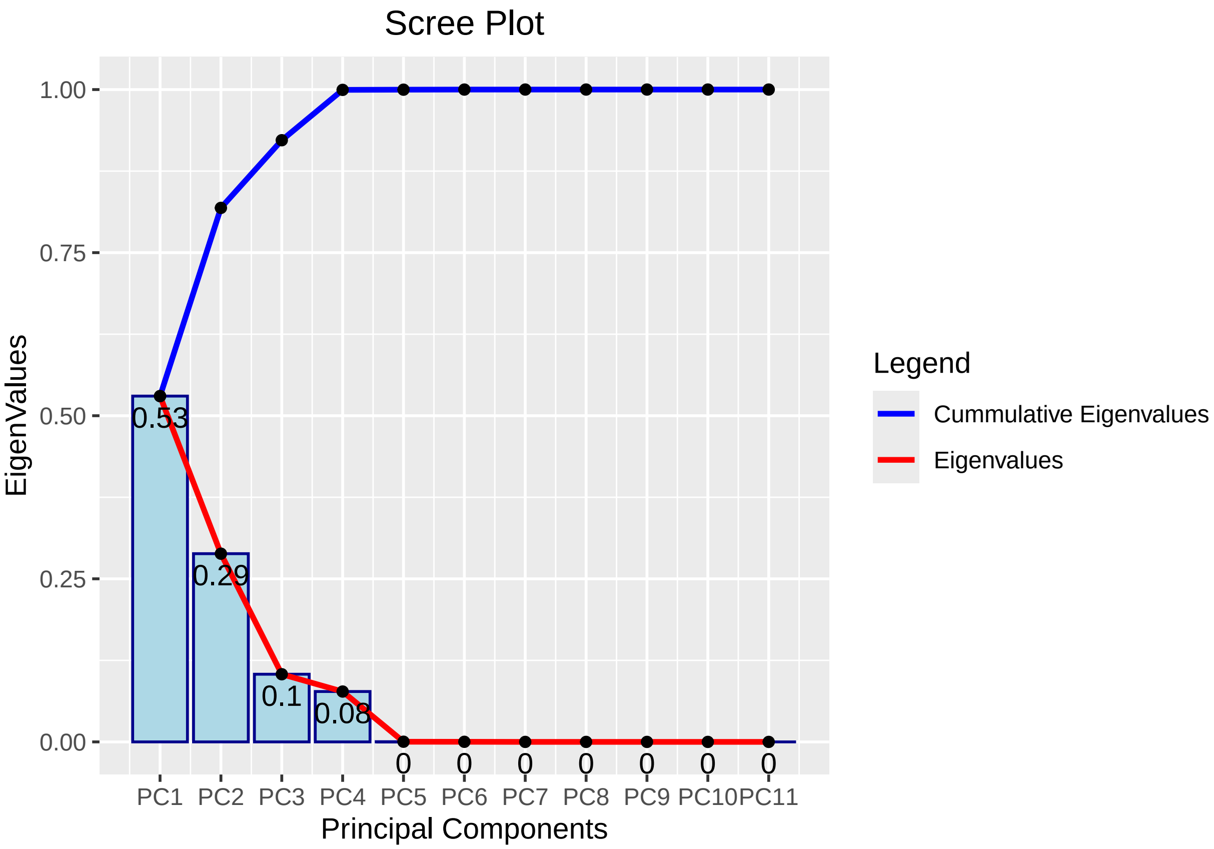
```
##    Cov_Mtrx.eigen.values pc.names1
## 1           4.252014e-01       PC1
## 2           2.314241e-01       PC1
## 3           8.321070e-02       PC1
## 4           6.192743e-02       PC1
## 5           2.191894e-04       PC1
## 6           1.374119e-04       PC1
## 7           1.225983e-05       PC1
## 8           1.104557e-06       PC1
## 9           6.109843e-07       PC1
## 10          2.558324e-07       PC1
## 11          1.203109e-07       PC1
```

```
eigen_data <- klovan::eigen_contribution(klovan)
eigen_data
```

```
##     EigenValues    CumSum CumSumPct pc.names
## 1            NA 0.0000000   0.00000      PC0
## 2  5.300873e-01 0.5300873  53.00873      PC1
## 3  2.885103e-01 0.8185977  81.85977      PC2
## 4  1.037366e-01 0.9223342  92.23342      PC3
## 5  7.720330e-02 0.9995375  99.95375      PC4
## 6  2.732577e-04 0.9998108  99.98108      PC5
## 7  1.713078e-04 0.9999821  99.99821      PC6
## 8  1.528401e-05 0.9999974  99.99974      PC7
## 9  1.377022e-06 0.9999988  99.99988      PC8
## 10 7.616980e-07 0.9999995  99.99995      PC9
## 11 3.189395e-07 0.9999999  99.99999     PC10
## 12 1.499884e-07 1.0000000 100.00000     PC11
```

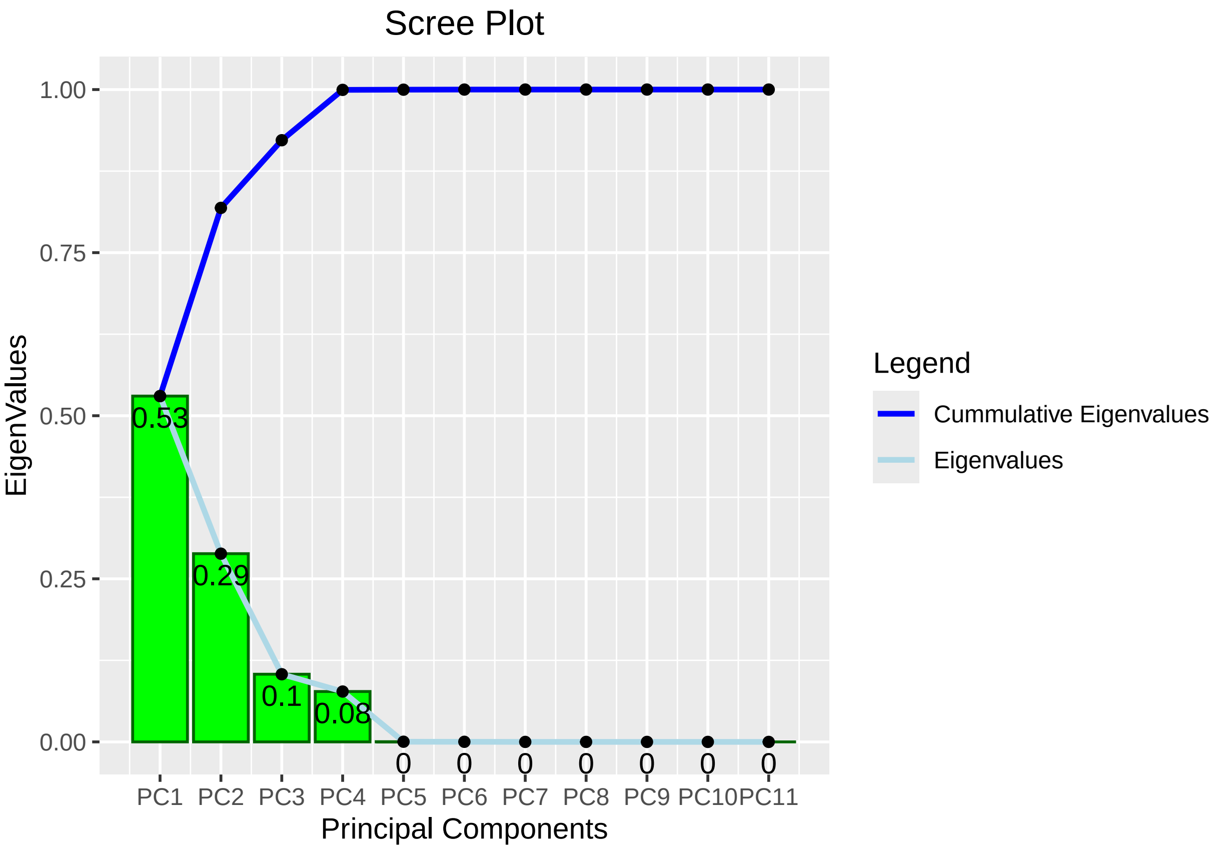
```
klovan::scree_plot(eigen_data)
```

```
## Warning in Ops.factor(left, right): '<' not meaningful for factors
```



```
klovan::scree_plot(eigen_data, bar_fill = "green", outline = "darkgreen", eigen_line = "lightblue")
```

```
## Warning in Ops.factor(left, right): '<' not meaningful for factors
```



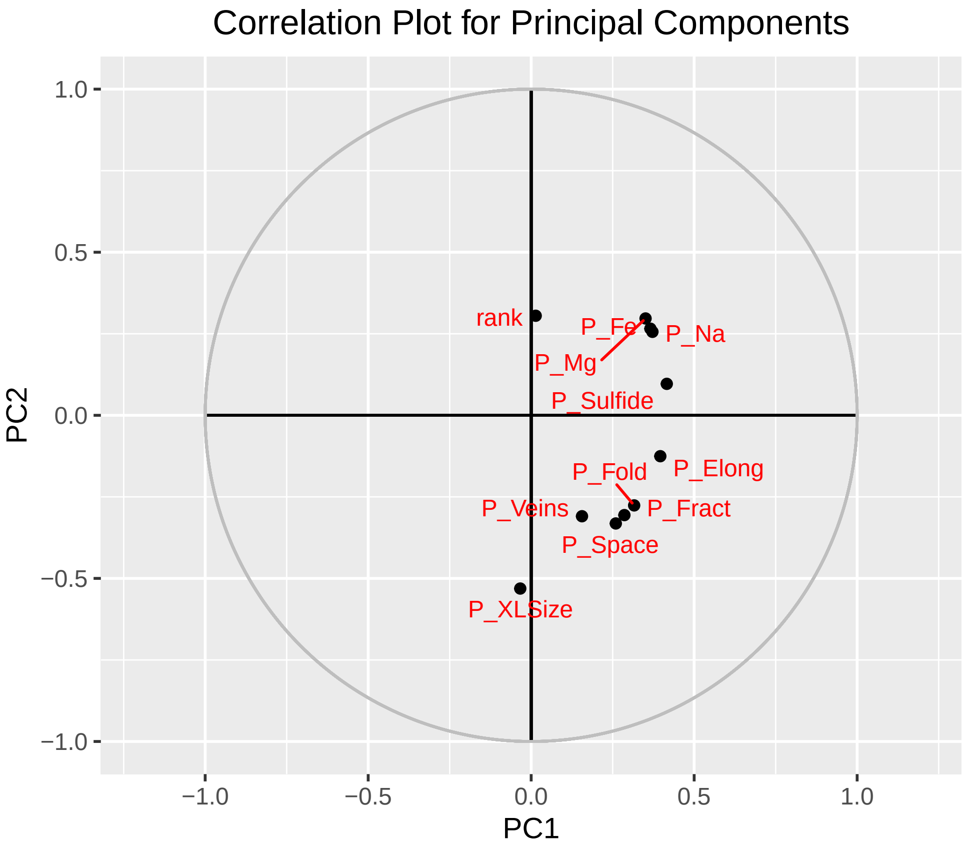
```
#make a correlation Matrix

klovan::cor_mtrx(Klovan_Row80)
```

```
##                  rank         C_X         C_Y     P_Elong        P_Fe
## rank       1.00000000  0.02506345  0.99968586 -0.03976893  0.23082906
## C_X        0.02506345  1.00000000  0.00000000  0.70097957  0.95466075
## C_Y        0.99968586  0.00000000  1.00000000 -0.05735591  0.20696699
## P_Elong   -0.03976893  0.70097957 -0.05735591  1.00000000  0.65503528
## P_Fe       0.23082906  0.95466075  0.20696699  0.65503528  1.00000000
## P_Fold    -0.14970466  0.40728820 -0.15996296  0.93123874  0.33573939
## P_Fract   -0.30943160  0.54374003 -0.32316112  0.63085545  0.34149181
## P_Mg       0.25713098  0.93539066  0.23376029  0.61721603  0.99597365
## P_Na       0.23173484  0.95085566  0.20796845  0.68659362  0.99854463
## P_Space   -0.18973487  0.23983387 -0.19580744  0.85063683  0.16291837
## P_Sulfide  0.09261898  0.96261047  0.06851416  0.84202670  0.94263453
## P_Veins   -0.32630071  0.31994401 -0.33442466  0.27808516  0.09858051
## P_XLSize  -0.44729482 -0.38545137 -0.43777160  0.17145666 -0.56921263
##               P_Fold    P_Fract       P_Mg      P_Na    P_Space  P_Sulfide
## rank      -0.1497047 -0.3094316  0.25713098 0.2317348 -0.1897349 0.09261898
## C_X        0.4072882  0.5437400  0.93539066 0.9508557  0.2398339 0.96261047
## C_Y       -0.1599630 -0.3231611  0.23376029 0.2079684 -0.1958074 0.06851416
## P_Elong    0.9312387  0.6308555  0.61721603 0.6865936  0.8506368 0.84202670
## P_Fe       0.3357394  0.3414918  0.99597365 0.9985446  0.1629184 0.94263453
## P_Fold     1.0000000  0.5976258  0.29242518 0.3767581  0.9836081 0.59142846
## P_Fract    0.5976258  1.0000000  0.26946932 0.3393873  0.5372043 0.59898725
## P_Mg       0.2924252  0.2694693  1.00000000 0.9944205  0.1201780 0.91600515
## P_Na       0.3767581  0.3393873  0.99442050  1.0000000  0.2070837  0.94917925
## P_Space    0.9836081  0.5372043  0.12017804  0.2070837  1.0000000  0.43647540
## P_Sulfide  0.5914285  0.5989872  0.91600515  0.9491792  0.4364754  1.00000000
## P_Veins    0.2693213  0.9174184  0.02924389  0.0801472  0.2343419  0.32085762
## P_XLSize   0.4744795  0.4987385 -0.62394721 -0.5501372  0.5912475 -0.26225796
##               P_Veins   P_XLSize
## rank      -0.32630071 -0.4472948
## C_X        0.31994401 -0.3854514
## C_Y       -0.33442466 -0.4377716
## P_Elong    0.27808516  0.1714567
## P_Fe       0.09858051 -0.5692126
## P_Fold     0.26932130  0.4744795
## P_Fract    0.91741841  0.4987385
## P_Mg       0.02924389 -0.6239472
## P_Na       0.08014720 -0.5501372
## P_Space    0.23434193  0.5912475
## P_Sulfide  0.32085762 -0.2622580
## P_Veins    1.00000000  0.5230651
## P_XLSize   0.52306512  1.0000000
```

```
klovan::pc_cor_plot(Klovan_Row80, "PC1", "PC2")
```

```
## Warning in ggforce::geom_circle(aes(x0 = 0, y0 = 0, r = 1), color = "gray", : All aesthetics have len
## i Please consider using `annotate()` or provide this layer with data containing
##   a single row.
```

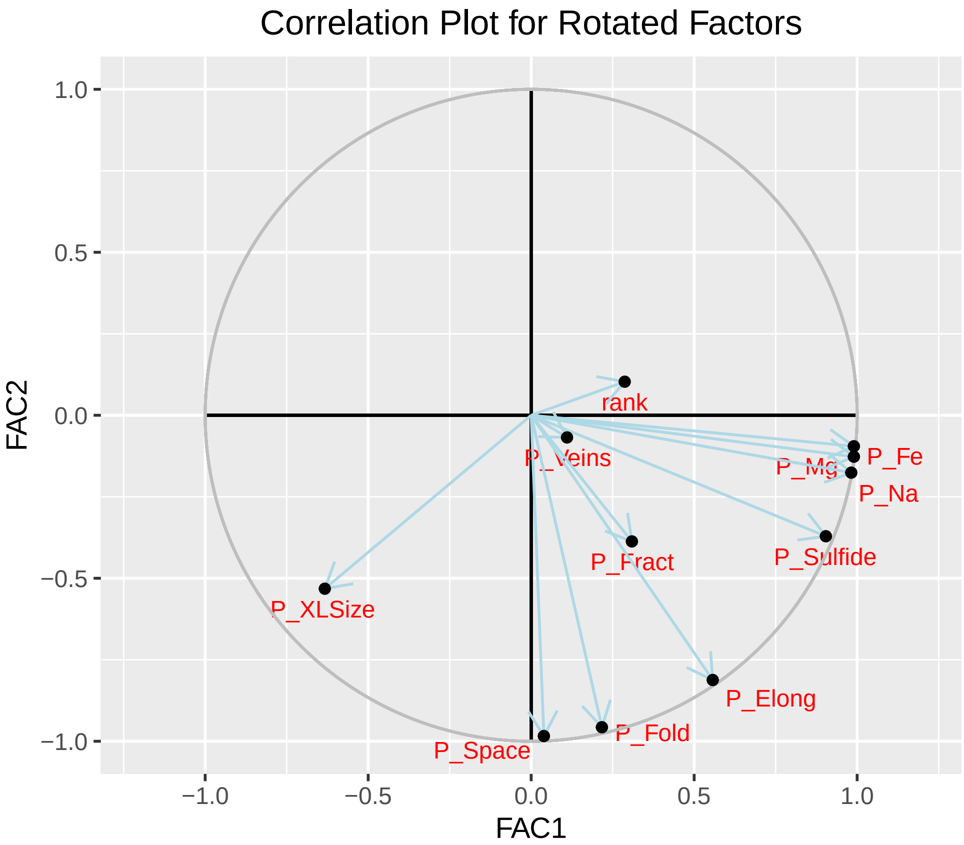


```
#factor analysis
```

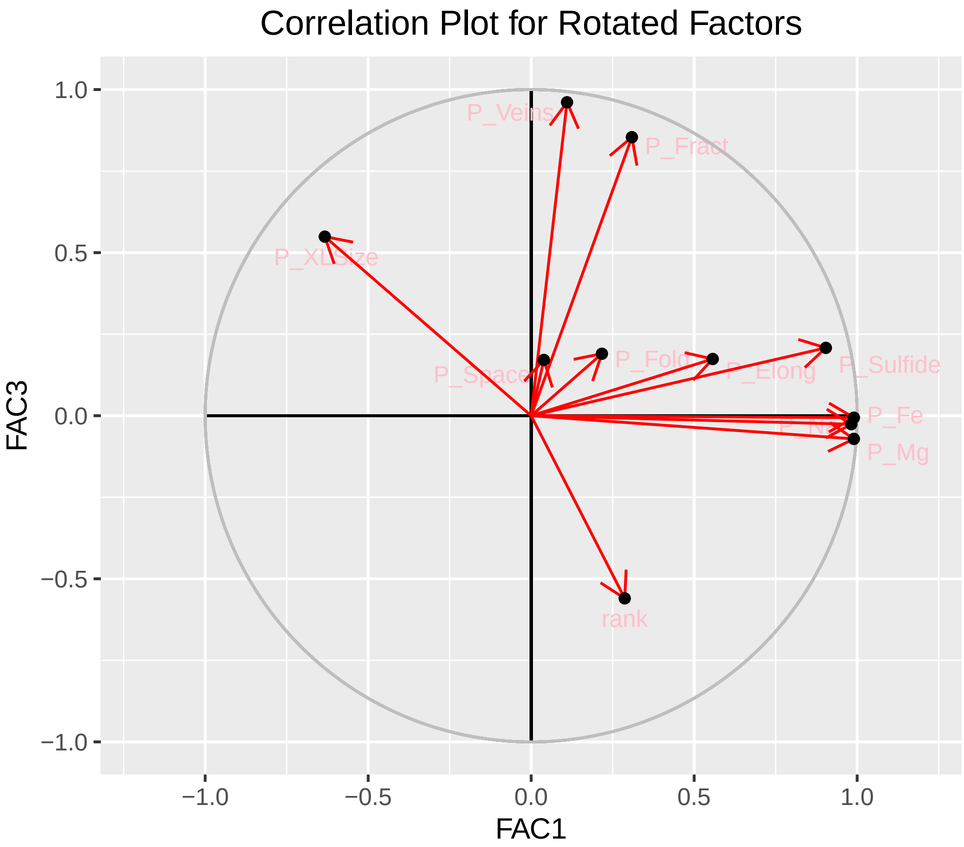
```
klovan::factor_analysis(Klovan_Row80)
```

```
##           VariableName   FAC1   FAC2   FAC3
## rank              rank  0.287  0.103 -0.560
## P_Elong        P_Elong  0.557 -0.812  0.174
## P_Fe              P_Fe  0.990 -0.127 -0.006
## P_Fold          P_Fold  0.217 -0.957  0.190
## P_Fract        P_Fract  0.309 -0.387  0.854
## P_Mg              P_Mg  0.990 -0.095 -0.071
## P_Na              P_Na  0.982 -0.176 -0.026
## P_Space        P_Space  0.039 -0.984  0.171
## P_Sulfide    P_Sulfide  0.904 -0.371  0.208
## P_Veins        P_Veins  0.110 -0.068  0.961
## P_XLSize      P_XLSize -0.633 -0.532  0.549
```

```
#make correlation plot using factor data
klovan::factor_cor_plot(klovan::factor_analysis(Klovan_Row80), "FAC1", "FAC2")
```



```
#customize color choices
klovan::factor_cor_plot(Klovan_Row80, "FAC1", "FAC3", text_col = "pink", line_col = "red")
```



```
#use inverse distance weighted method for interpolation

inv_dis_data <- klovan::inv_dis_wt(Klovan_Row80, 3)
```

```
## [inverse distance weighted interpolation]
## [inverse distance weighted interpolation]
## [inverse distance weighted interpolation]
```

```
summary(inv_dis_data) #view data summary
```

```
##       C_X            C_Y           value                 FA
##  Min.   : 900   Min.   : 900   Min.   :-2.12254   Length:9792
##  1st Qu.:2176   1st Qu.:1872   1st Qu.:-0.54952   Class :character
##  Median :3452   Median :2925   Median : 0.05459   Mode  :character
##  Mean   :3452   Mean   :2925   Mean   : 0.02533
##  3rd Qu.:4727   3rd Qu.:3978   3rd Qu.: 0.60358
##  Max.   :6003   Max.   :4950   Max.   : 2.09071
```

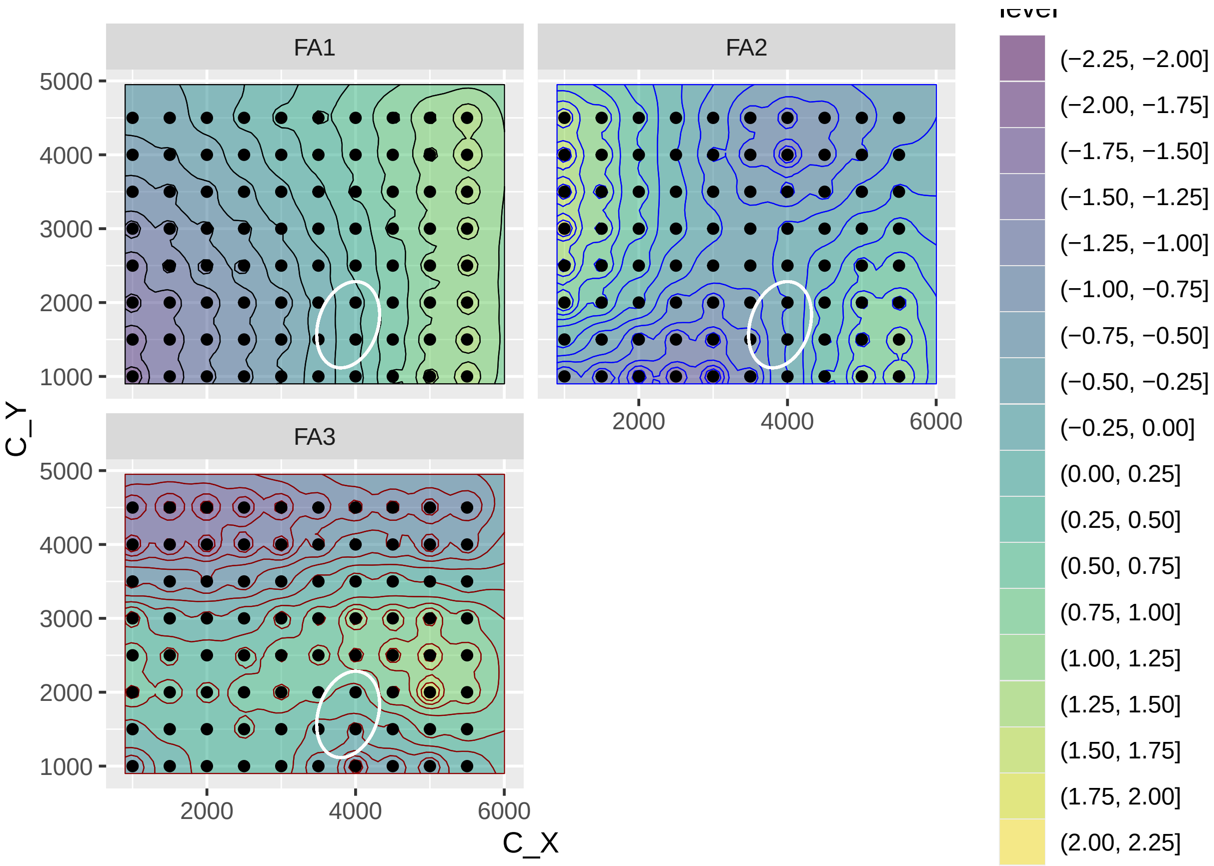
```
library(ggforce)
```

```
## Loading required package: ggplot2
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.4     v tibble    3.2.1
## v purrr     1.0.4     v tidyr     1.3.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

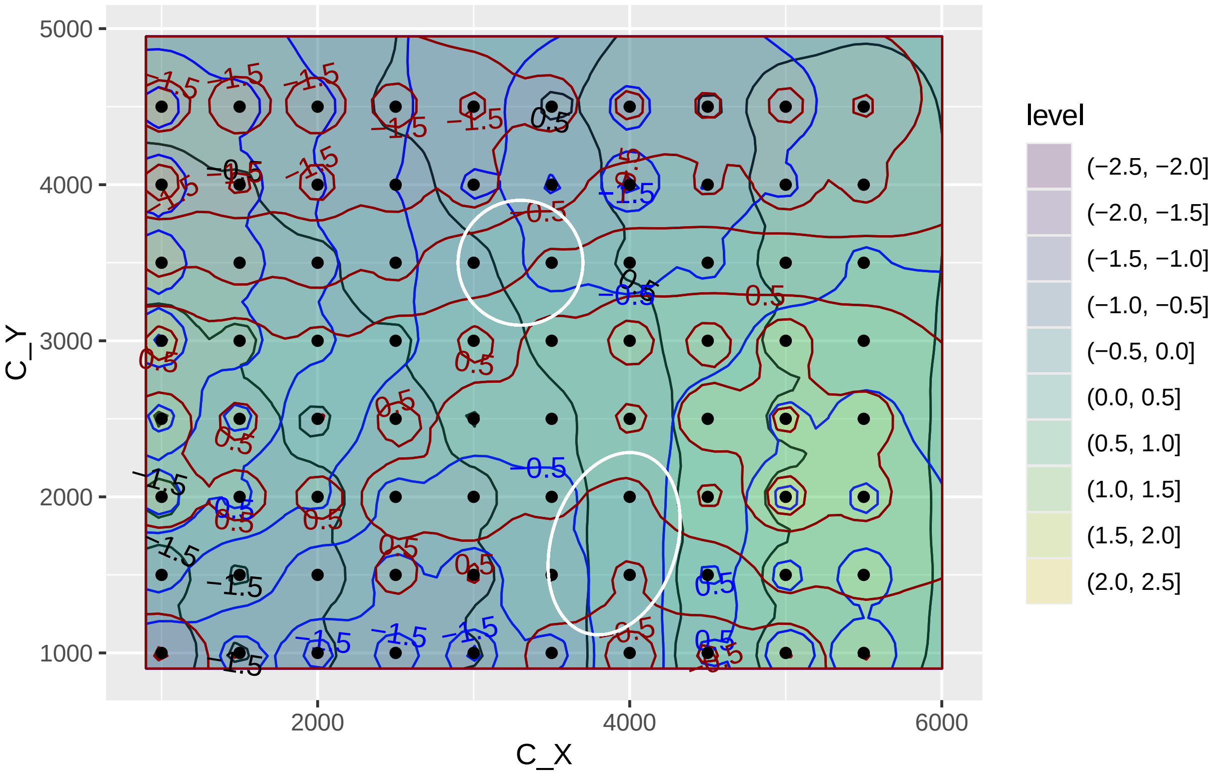
```
klovan::factor_score_plot(inv_dis_data, FALSE, data = Klovan_Row80) + ggforce::geom_ellipse(
    aes(x0 = 3900, y0 = 1700, a = 600, b = 400, angle = pi/2.5),
    color = "white")
```



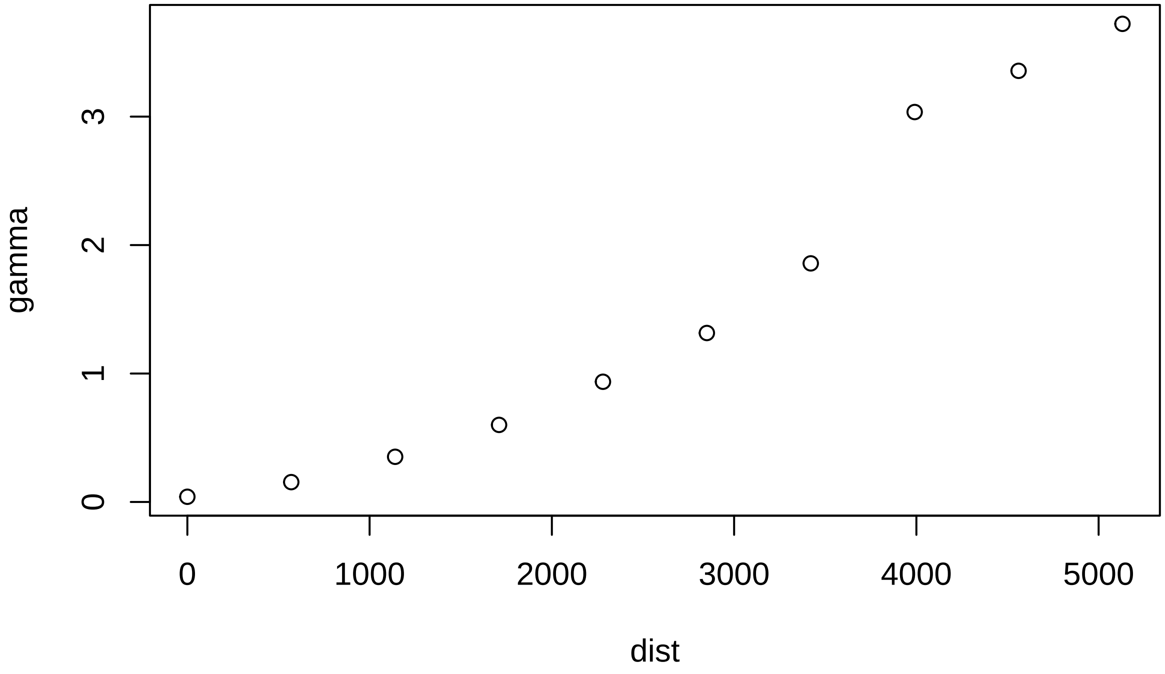
```
klovan::factor_score_plot(inv_dis_data, TRUE, data = Klovan_Row80) + ggforce::geom_ellipse(
    aes(x0 = 3900, y0 = 1700, a = 600, b = 400, angle = pi/2.5),
    color = "white") +
  ggforce::geom_circle(
    aes(x = NULL, y = NULL, x0 = 3300, y0 = 3500, r = 400),
  color = "white",
  inherit.aes = FALSE)
```

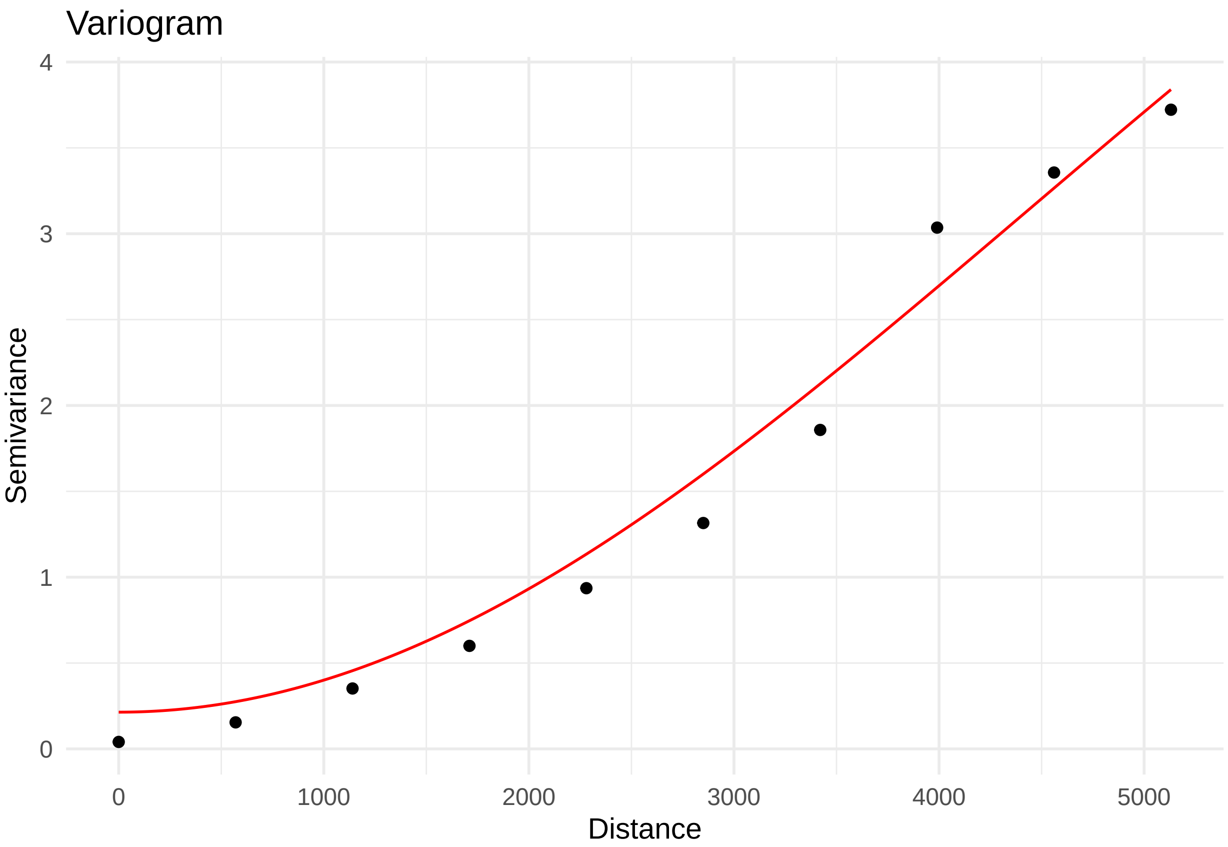
```
## Warning in metR::geom_text_contour(min.size = 5, binwidth = 0.5, label.size =
## 0.5, : Ignoring unknown parameters: `label.size`

## Warning in ggforce::geom_circle(aes(x = NULL, y = NULL, x0 = 3300, y0 = 3500, : All aesthetics have
## i Please consider using `annotate()` or provide this layer with data containing
##   a single row.
```



```
#plot variogram for use in kriging
klovan::vario_plot(Klovan_Row80, factor = 1, nugget = .214, nlags = 10, sill = 7.64507, range_val = 627
```





```
# Load required packages
library(tidyverse)
library(klovan)

# Use tryCatch to suppress errors and continue execution
krig_data <- tryCatch({
  klovan::kriging.auto(Klovan_Row80, 3)
}, error = function(e) {
  message("An error occurred but was suppressed.")
  return(NULL)  # Return NULL or any value if needed in case of error
})

# If kriging succeeds, view summary
if (!is.null(krig_data)) {
  summary(krig_data)
} else {
  message("Kriging failed but continued.")
}
```

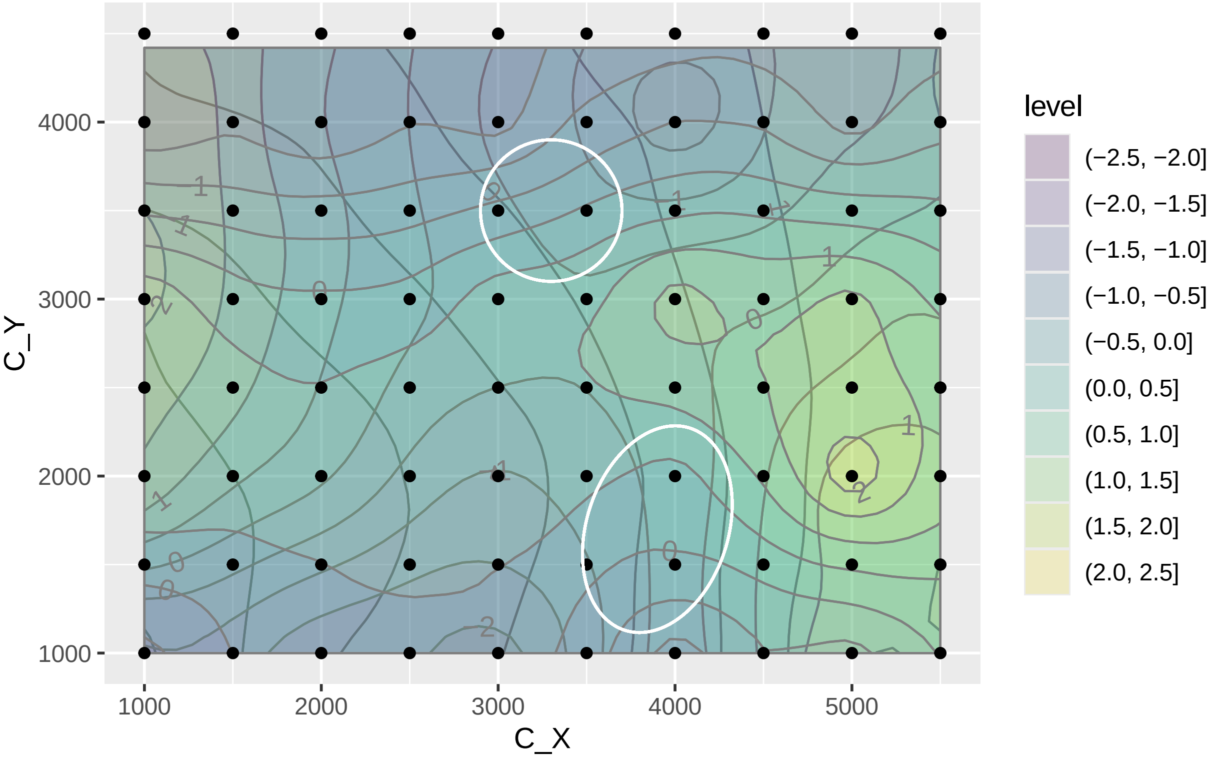
```
##      value               C_X           C_Y            FA
##  Min.   :-2.16399   Min.   :1000   Min.   :1000   Length:5967
##  1st Qu.:-0.67943   1st Qu.:2080   1st Qu.:1810   Class :character
##  Median : 0.04949   Median :3250   Median :2710   Mode  :character
##  Mean   : 0.02175   Mean   :3250   Mean   :2710
##  3rd Qu.: 0.73944   3rd Qu.:4420   3rd Qu.:3610
##  Max.   : 2.24103   Max.   :5500   Max.   :4420
```

```
klovan::factor_score_plot(krig_data, TRUE, data = Klovan_Row80) + ggforce::geom_ellipse(
    aes(x0 = 3900, y0 = 1700, a = 600, b = 400, angle = pi/2.5),
    color = "white") +
  ggforce::geom_circle(
    aes(x = NULL, y = NULL, x0 = 3300, y0 = 3500, r = 400),
  color = "white",
  inherit.aes = FALSE)
```

```
## Warning in metR::geom_text_contour(min.size = 5, binwidth = 0.5, label.size =
## 0.5, : Ignoring unknown parameters: `label.size`
```

```
## Warning in ggforce::geom_circle(aes(x = NULL, y = NULL, x0 = 3300, y0 = 3500, : All aesthetics have
## i Please consider using `annotate()` or provide this layer with data containing
##   a single row.
```

```
## Warning: No shared levels found between `names(values)` of the manual scale and the
## data's colour values.
## No shared levels found between `names(values)` of the manual scale and the
## data's colour values.
```

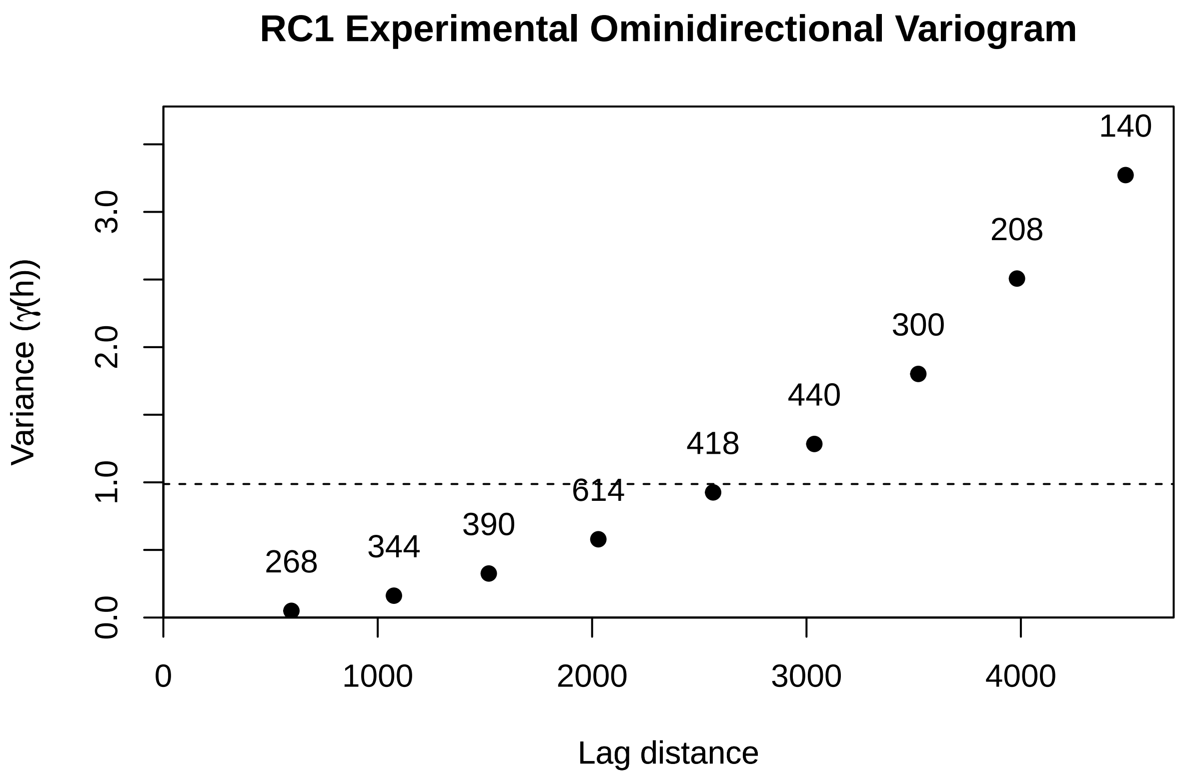


```
#install.packages("Klovan_0.0.9.tar.gz", repos = NULL, type = "source")
library(klovan)
```

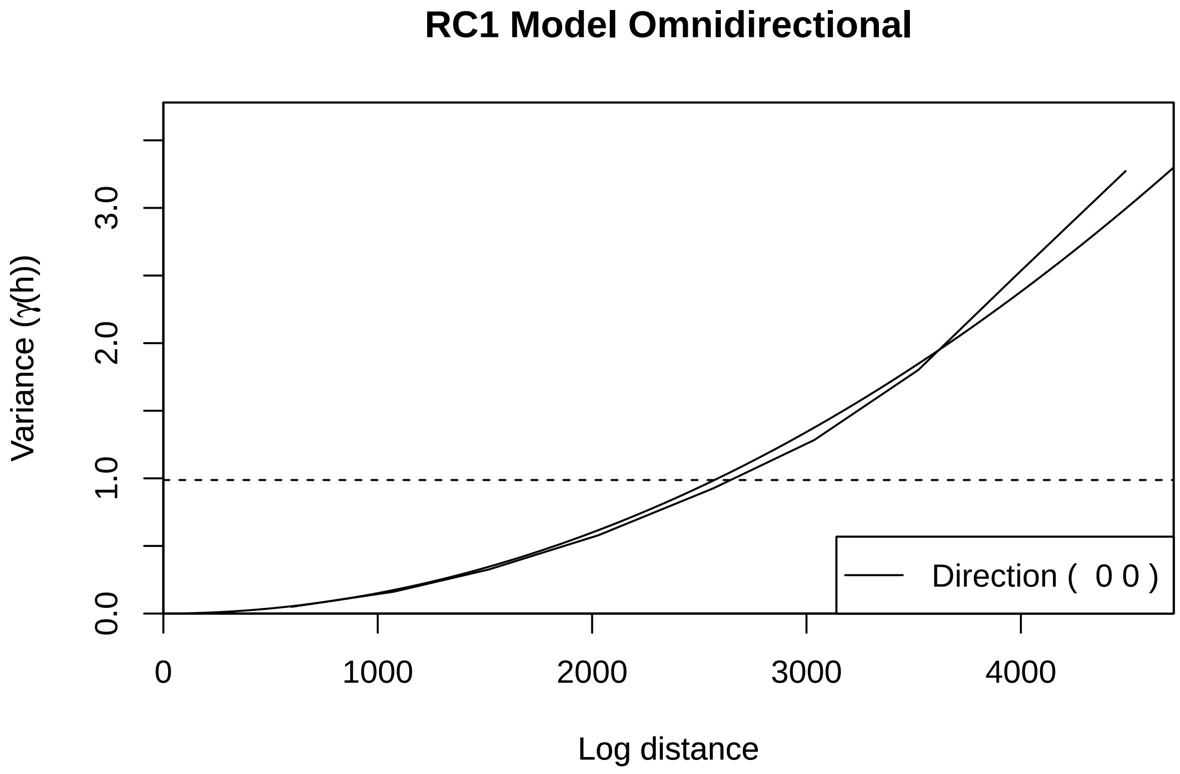
```
library(RGeostats)
data("Klovan_Row80", package = "klovan")
#Klovan_Row80 <- load(file = "~/CSE_MSE_RXF131/cradle-members/sdle/jeg165/git/klovan/packages/Klovan0.0
```

```
# Building a database based on RC1 factor
db <- Rgeo_database(Klovan_Row80, 3, "RC1")
```

```
# Construct and plot the experimental variogram
Rgeo_vario_construct_plot(db, 3, "RC1", lag = 500)
```



```
# Fit the variogram model based on experimental variogram
model <- Rgeo_vario_model(db, 3, "RC1", lag = 500, model = 13)
```



```
print(model)
```

```
##
```

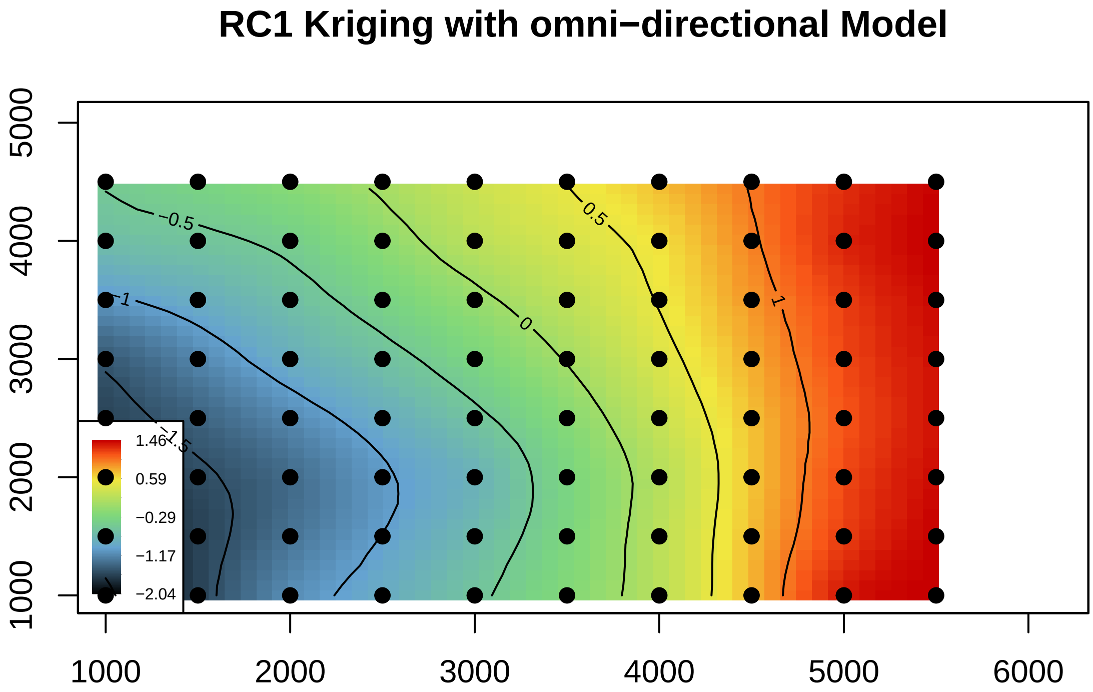
```
## Model characteristics
## =====================
## Space dimension              = 2
## Number of variable(s)        = 1
## Number of basic structure(s) = 1
## Number of drift function(s)  = 1
## Number of drift equation(s)  = 1
##
## Covariance Part
## ---------------
## Power (Third Parameter = 1.99)
## - Slope        =      0.001
##
## Drift Part
## ----------
## Universality Condition
```

```r
krig <- Rgeo_kriging(db, model)
```

```r
print(krig)
```

```
##
## Data Base Grid Characteristics
## ==============================
##
## Data Base Summary
## -----------------
## File is organized as a regular grid
## Space dimension              = 2
## Number of Columns            = 5
## Maximum Number of UIDs       = 5
## Total number of samples      = 2173
##
## Grid characteristics:
## ---------------------
## Origin :   1000.000  1000.000
## Mesh   :     86.000    86.000
## Number :         53        41
##
## Variables
## ---------
## Column = 0 - Name = rank - Locator = NA
## Column = 1 - Name = x1 - Locator = x1
## Column = 2 - Name = x2 - Locator = x2
## Column = 3 - Name = Omni.RC1.estim - Locator = z1
## Column = 4 - Name = Omni.RC1.stdev - Locator = z2
```

```r
# Plot the kriging estimation results
Rgeo_kriging_plot(krig, db, "RC1")
```



RC1 Kriging with omni−directional Model