

# Package ‘krige’

September 22, 2020

**Version** 0.5.7

**Title** Geospatial Kriging with Metropolis Sampling

**Date** 2020-09-20

**Author** Jason S. Byers [aut, cre], Le Bao [aut], Jamie Carson [aut], Jeff Gill [aut]

**Maintainer** Jason S. Byers <jaybyers55@gmail.com>

**Description** Estimates kriging models for geographical point-referenced data. Method is described in Gill (2020) <doi:10.1177/1532440020930197>.

**License** GPL (>= 2)

**Imports** Rcpp (>= 1.0.4)

**LinkingTo** Rcpp

**Depends** R (>= 3.5.0)

**LazyData** true

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2020-09-22 04:30:08 UTC

## R topics documented:

congCombined . . . . .	2
ContrivedData . . . . .	3
exponential.semivariogram . . . . .	4
geweke . . . . .	6
heidel.welch . . . . .	7
krige.posterior . . . . .	9
krige.pred . . . . .	11
krige.update . . . . .	13
lowerCombined . . . . .	15
metropolis.krige . . . . .	17
NYcity_subset . . . . .	20
NY_subset . . . . .	23
semivariogram . . . . .	26

stateCombined . . . . .	28
upperCombined . . . . .	29
WVwells . . . . .	31

<b>Index</b>	<b>35</b>
--------------	-----------

---

congCombined	<i>Congressional District Public Opinion Ideology in 2010</i>
--------------	---

---

## Description

These data present measures of ideology in 2010 for 434 districts for the U.S. House of Representatives, recorded as the variable `krige.cong`. Forecasts are based on a kriging model fitted over the 2008 Cooperative Congressional Election Survey (CCES), paired with predictive data from the 2010 Census. Each district's public ideology is paired with the DW-NOMINATE common space score of each of its representative in 2011 (update from McCarty, Poole and Rosenthal 1997). Eight districts have repeated observations in order to include the DW-NOMINATE score when a member was replaced mid-term.

## Usage

```
data(congCombined)
```

## Format

The `congCombined` dataset has 442 observations and 12 variables. 434 out of 435 congressional districts are covered, with eight districts duplicated when a member was replaced mid-term.

`stateCD` Unique identifier for each congressional district by state. The first two digits are STATEA, and the second two are cd.

`krige.cong` The ideology of the average citizen in the congressional district.

`krige.state.var` The variance of ideology among the district's citizens.

`cong` The term of Congress studied—112 for this dataset.

`idno` Identification number for the House member—ICPSR numbers continued by Poole & Rosenthal.

`state` The ICPSR code for the state.

`cd` The congressional district number.

`statenm` The first seven letters of the state's name.

`party` Political party of the House member. 100=Democrat, 200=Republican.

`name` Last name of the House member, followed by first name if ambiguous.

`dwnom1` First dimension DW-NOMINATE common space score for the House member. Higher values are usually interpreted as more right-wing, with lower values as more left-wing.

`STATEA` The FIPS code for the state.

## Source

Ansolabehere, Stephen. 2011. "CCES, Common Content, 2008." Ver. 4.  
<http://hdl.handle.net/1902.1/14003>

McCarty, Nolan M., Keith T. Poole and Howard Rosenthal. 1997. *Income Redistribution and the Realignment of American Politics*. American Enterprise Institute Studies on Understanding Economic Inequality. Washington: AEI Press.

Minnesota Population Center. 2011. *National Historical Geographic Information System: Version 2.0*. Minneapolis, MN: University of Minnesota.  
 'https://www.nhgis.org'

## References

Jeff Gill. 2020. Measuring Constituency Ideology Using Bayesian Universal Kriging. *State Politics & Policy Quarterly*.

## Examples

```
# Descriptive Statistics
summary(congCombined)

# Correlate House Members' DW-NOMINATE Scores with Public Opinion Ideology
cor(congCombined$dwnom1, congCombined$krige.cong)

# Plot House Members' DW-NOMINATE Scores against Public Opinion Ideology
plot(y=congCombined$dwnom1, x=congCombined$krige.cong,
     xlab="District Ideology (Kriging)", ylab="Legislator Ideology (1st Dim., Common Space)",
     main="U.S. House of Representatives", type="n")
points(y=congCombined$dwnom1[congCombined$party==200],
       x=congCombined$krige.cong[congCombined$party==200], pch="R", col="red")
points(y=congCombined$dwnom1[congCombined$party==100],
       x=congCombined$krige.cong[congCombined$party==100], pch="D", col="blue")
```

---

ContrivedData

*Simulated Point-Referenced Spatial Data*

---

## Description

These data are a simulated point-referenced geospatial data that serve to provide a clean example of a kriging model. There are 500 observations with coordinates located on a unit square.

## Usage

```
data(NY_subset)
```

**Format**

The ContrivedData dataset has 500 observations and 5 variables.

y The outcome variable. Its true population functional form is  $y_s = 0 + 1x_{1s} + 2x_{2s} + \omega_s + \epsilon_s$ . The true variance of  $\omega$  is  $\sigma^2 = 0.5$  and of  $\epsilon$  is  $\tau^2 = 0.5$ . The decay term that shapes spatial correlation levels is  $\phi = 2.5$ .

x.1 A predictor with a standard uniform distribution.

x.2 A predictor with a standard normal distribution.

s.1 Coordinate in eastings for each observation, distributed standard uniform.

s.2 Coordinate in northings for each observation, distributed standard uniform.

**Examples**

```
#Examine Data
summary(ContrivedData)

#Initial OLS Model
contrived.ols<-lm(y~x.1+x.2,data=ContrivedData);summary(contrived.ols)
```

---

```
exponential.semivariogram
```

*Parametric Exponential Semivariogram*

---

**Description**

This function returns the value of a parametric powered exponential semivariogram given the values of the parameters and the distance between observations.

**Usage**

```
exponential.semivariogram(nugget,decay,partial.sill,distance,power=2)
```

**Arguments**

nugget	The value of the non-spatial variance, or nugget term.
decay	The value of the decay term that sets the level of correlation given distance.
partial.sill	The value of the spatial variance, or partial sill term.
distance	The distance among observations for which the semivariance value is desired.
power	The exponent specified in the powered exponential semivariogram. Defaults to 2, which corresponds to a Gaussian semivariance function.

## Details

The models estimated by the `krige` package assume a powered exponential covariance structure. Each parametric covariance function for kriging models corresponds to a related semivariance function, given that highly correlated values will have a small variance in differences while uncorrelated values will vary widely. More specifically, semivariance is equal to half of the variance of the difference in a variable's values at a given distance. That is, the semivariance is defined as:  $\gamma(h) = 0.5 * E[X(s+h) - X(s)]^2$ , where  $X$  is the variable of interest,  $s$  is a location, and  $h$  is the distance from  $s$  to another location.

The powered exponential covariance structure implies that the semivariance follows the specific functional form of  $\gamma(d) = \tau^2 + \sigma^2(1 - \exp(-|\phi d|^p))$  (Banerjee, Carlin, and Gelfand 2015, 27). A perk of this structure is that the special case of  $p=1$  implies the commonly-used exponential semivariogram, and the special case of  $p=2$  implies the commonly-used Gaussian semivariogram. Upon estimating a model, it is advisable to graph the functional form of the implied parametric semivariance structure. By substituting estimated values of the nugget, decay, and `partial.sill` terms, as well as specifying the correct power argument, it is possible to compute the implied semivariance from the model. The distance argument easily can be a vector of observed distance values.

## Value

Returns an object of class `numeric` with the parametric value of the semivariance corresponding to each input value of distance.

## References

Sudipto Banerjee, Bradley P. Carlin, and Alan E. Gelfand. 2015. *Hierarchical Modeling and Analysis for Spatial Data*. 2nd ed. Boca Raton, FL: CRC Press.

## Examples

```
#Examine Data
summary(ContrivedData)

#OLS Model for Residuals
contrived.ols<-lm(y~x.1+x.2,data=ContrivedData);summary(contrived.ols)

#Examine the Parametric Semivariogram Graphically with Population Parameter Values
raw.semivar<-semivariogram(x=ContrivedData$y,east=ContrivedData$s.1,
  north=ContrivedData$s.2)
resid.semivar<-semivariogram(x=contrived.ols$residuals,east=ContrivedData$s.1,
  north=ContrivedData$s.2,draw.plot=FALSE)
points(resid.semivar,pch=3,col='blue')
lines(exponential.semivariogram(nugget=0.5,decay=2.5,partial.sill=0.5,
  distance=as.numeric(names(resid.semivar))),col='red')
```

---

geweke

*Geweke Diagnostic for MCMC*


---

## Description

This function takes a matrix of MCMC iterations and conducts a Geweke convergence diagnostic on each column.

## Usage

```
geweke(X,early.prop=.1,late.prop=.5,precision=4)
```

## Arguments

<code>X</code>	An object of the <code>matrix</code> class with results from an MCMC algorithm. Each column should designate a parameter, and each row should designate an iteration.
<code>early.prop</code>	Proportion of iterations to use from the start of the chain.
<code>late.prop</code>	Proportion of iterations to use from the end of the chain.
<code>precision</code>	Number of digits of test statistics and p-values to print.

## Details

Geweke's (1992) test for nonconvergence of a MCMC chain is to conduct a difference-of-means test that compares the mean early in the chain to the mean late in the chain. If the means are significantly different from each other, then this is evidence that the chain has not converged. The difference-of-means test is a simple z-ratio, though the standard error is estimated using the spectral density at zero to account for autocorrelation in the chain.

## Value

Returns a `matrix` in which the first row consists of z-scores for tests of equal means for the first and last parts of the chain. The second row consists of the corresponding p-values. Each column of the matrix represents another parameter. For each column, a significant result is evidence that the chain has not converged for that parameter. Thus, a non-significant result is desired.

## References

John Geweke. 1992. "Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments." In *Bayesian Statistics 4*, ed. J.M. Bernardo, J.O. Berger, A.P. Dawid, and A.F.M. Smith. Oxford: Clarendon Press.

## Examples

```
#Examine Data
summary(ContrivedData)

#Initial OLS Model
contrived.ols<-lm(y~x.1+x.2,data=ContrivedData);summary(contrived.ols)

#Define Covariate Matrix
covariates<-cbind(1,ContrivedData$x.1,ContrivedData$x.2)

#set seed
set.seed(1241060320)

#For simple illustration, we set to few iterations.
#In this case, a 10,000-iteration run converges to the true parameters.
#If you have considerable time and hardware, delete the # on the next line.
#10,000 iterations took 39 min. with 8 GB RAM & a 1.5 GHz Quad-Core processor.
M<-8
#M<-10000

#Run the Full Model
contrived.run<-metropolis.krige(y=ContrivedData$y,X=covariates,range.tol=0.05,
  east=ContrivedData$s.1,north=ContrivedData$s.2,mcmc.samples=M)

#Delete 20% for Burn-In
contrived.run<-contrived.run[(ceiling(0.2*M)+1):M,]

#examine results against true coefficients
TRUTH<-c(0.5,2.5,0.5,0,1,2)
rbind(apply(contrived.run,2,quantile,c(.5,.05,.95)),TRUTH)

#Convergence Diagnostic: Geweke
#Note that the second (commented) version of Geweke is more typical
#of a 10,000 iteration run.
geweke(contrived.run,early.prop=0.5)
#geweke(contrived.run)
```

---

heidel.welch

*Heidelberger and Welch Diagnostic for MCMC*


---

## Description

This function takes a matrix of MCMC iterations and conducts a Heidelberger and Welch convergence diagnostic on each column.

## Usage

```
heidel.welch(X,pvalue=0.05)
```

## Arguments

X	An object of the <code>matrix</code> class with results from an MCMC algorithm. Each column should designate a parameter, and each row should designate an iteration.
pvalue	Alpha level for significance tests. Defaults to 0.05.

## Details

This is an adaptation of a function in Plummer et al.'s coda package. Heidelberger and Welch's (1993) test for nonconvergence. This version of the diagnostic only reports a Cramer-von Mises test and its corresponding p-value to determine if the chain is weakly stationary with comparisons of early portions of the chain to the end of the chain.

## Value

Returns a matrix in which the first row consists of the values of the Cramer-von Mises test statistic for each parameter, and the second row consists of the corresponding p-values. Each column of the matrix represents another parameter of interest. A significant result serves as evidence of nonconvergence, so non-significant results are desired.

## References

Philip Heidelberger and Peter D. Welch. 1993. "Simulation Run Length Control in the Presence of an Initial Transient." *Operations Research* 31:1109-1144.

Martyn Plummer, Nicky Best, Kate Cowles and Karen Vines. 2006. "CODA: Convergence Diagnosis and Output Analysis for MCMC." *R News* 6:7-11.

## Examples

```
#Examine Data
summary(ContrivedData)

#Initial OLS Model
contrived.ols<-lm(y~x.1+x.2,data=ContrivedData);summary(contrived.ols)

#Define Covariate Matrix
covariates<-cbind(1,ContrivedData$x.1,ContrivedData$x.2)

#set seed
set.seed(1241060320)

#For simple illustration, we set to few iterations.
#In this case, a 10,000-iteration run converges to the true parameters.
#If you have considerable time and hardware, delete the # on the next line.
#10,000 iterations took 39 min. with 8 GB RAM & a 1.5 GHz Quad-Core processor.
M<-8
#M<-10000

#Run the Full Model
contrived.run<-metropolis.krige(y=ContrivedData$y,X=covariates,range.tol=0.05,
  east=ContrivedData$s.1,north=ContrivedData$s.2,mcmc.samples=M)
```



```
#Delete 20% for Burn-In
contrived.run<-contrived.run[(ceiling(0.2*M)+1):M,]

#examine results against true coefficients
TRUTH<-c(0.5,2.5,0.5,0,1,2)
rbind(apply(contrived.run,2,quantile,c(.5,.05,.95)),TRUTH)

#Convergence Diagnostic: Heidelberger-Welch
heidel.welch(contrived.run)
```

krige.posterior

*Posterior Distribution for the Kriging Process***Description**

This function finds the posterior density of a geospatial linear regression model given a point-referenced geospatial dataset and a set of parameter values. The function is useful for finding the optimum of or for sampling from the posterior distribution.

**Usage**

```
krige.posterior(tau2,phi,sigma2,beta,y,X,east,north,semivar.exp=2,
  p.spatial.share=0.5,p.range.share=0.5,p.range.tol=0.05,p.beta.var=10,
  tot.var=var(y),local.Sigma=NULL,max.distance=NULL)
```

**Arguments**

tau2	Value of the nugget, or non-spatial error variance.
phi	Value of the decay term, driving the level of spatial correlation.
sigma2	Value of the partial sill, or maximum spatial error variance.
beta	Coefficients from linear model.
y	The dependent variable that is used in the kriging model.
X	The matrix of independent variables used in the kriging model.
east	Vector of eastings for all observations.
north	Vector of northings for all observations.
semivar.exp	This exponent, which must be greater than 0 and less than or equal to 2, specifies a powered exponential correlation structure for the data. One widely used specification is setting this to 1, which yields an exponential correlation structure. Another common specification is setting this to 2 (the default), which yields a Gaussian correlation structure.
p.spatial.share	Prior for proportion of unexplained variance that is spatial in nature. Must be greater than 0 and less than 1. Defaults to an even split.

p.range.share	Prior for the effective range term, as a proportion of the maximum distance in the data. Users should choose the proportion of distance at which they think the spatial correlation will become negligible. Must be greater than 0. Values greater than 1 are permitted, but users should recognize that this implies that meaningful spatial correlation would persist outside of the convex hull of data. Defaults to half the maximum distance.
p.range.tol	Tolerance term for setting the effective range. At the distance where the spatial correlation drops below this term, it is judged that the effective range has been met. Users are typically advised to leave this at its default value of 0.05 unless they have strong reasons to choose another level. Must be greater than 0 and less than 1.
p.beta.var	Prior for the variance on zero-meaned normal priors on the regression coefficients. Defaults to 10.
tot.var	Combined variance between the nugget and partial sill. Defaults to the variance of y. The metropolis.krige function inserts the residual variance from a standard linear model.
local.Sigma	The user is advised to ignore this option, or leave it the value of NULL. This option is included to reduce the number of computations required when this function is called by metropolis.krige.
max.distance	The user is advised to ignore this option, or leave it the value of NULL. This option is included to reduce the number of computations required when this function is called by metropolis.krige.

## Details

This function finds the posterior density for a kriging model. The function utilizes information provided about the parameters tau2, phi, sigma2, and beta. It also utilizes the observed data y, X, east, and north. Given a set of parameter values as well as the observed data, the function returns the posterior density for the specified model.

## Value

Returns a single number that is the posterior density of the function, which is stored in object of class matrix.

## References

Jeff Gill. 2020. Measuring Constituency Ideology Using Bayesian Universal Kriging. *State Politics & Policy Quarterly*.

## Examples

```
#Examine Data
summary(ContrivedData)

#Initial OLS Model
contrived.ols<-lm(y~x.1+x.2,data=ContrivedData);summary(contrived.ols)
```

```
#Define Covariate Matrix
covariates<-cbind(1,ContrivedData$x.1,ContrivedData$x.2)

# Find the posterior density for the Contrived Data if all parameters were 1:
s.test <- krige.posterior(tau2=1,phi=1,sigma2=1,beta=rep(1,ncol(covariates)),
  y=ContrivedData$y,X=covariates,east=ContrivedData$s.1,north=ContrivedData$s.2)

# Print posterior density
s.test
```

krige.pred

*Predictions by Kriging*

## Description

This function uses the results of a model estimated by `metropolis.krige` to make kriging-based predictions, the input data for that model, and data with coordinates for the new responses to be predicted.

## Usage

```
krige.pred(pred.x,pred.east,pred.north,train.y,train.x,train.east,train.north,
  mcmc.iter,powered.exp=2,credible=NULL,inv.Sigma=NULL)
```

## Arguments

<code>pred.x</code>	The matrix of independent variables for observations to be predicted.
<code>pred.east</code>	Vector of eastings for observations to be predicted.
<code>pred.north</code>	Vector of northings for observations to be predicted.
<code>train.y</code>	The dependent variable that was used to estimate the original model.
<code>train.x</code>	The matrix of independent variables that was used to estimate the original model.
<code>train.east</code>	Vector of eastings for all observations that was used to estimate the original model.
<code>train.north</code>	Vector of northings for all observations that was used to estimate the original model.
<code>mcmc.iter</code>	The output matrix created by <code>metropolis.krige</code> containing the iterations of the MCMC sampler from the model.
<code>powered.exp</code>	This value should be set to the same value from the original model. It defaults to 2, which is also the default in the <code>metropolis.krige</code> function. Thus, if this option is ignored in both commands, they appropriately will be equal.
<code>credible</code>	If a credible interval on predictions is desired, a user may specify a proportion between 0 and 1 to indicate the interval probability. For example, a value of 0.9 would create a 90% credible interval. If <code>NULL</code> , then no credible interval will be produced.

`inv.Sigma` Users will almost always want to leave this parameter at its default value of `NULL`. If `inv.Sigma` is specified, the user is providing the inverse of an estimated covariance matrix of errors that will be used when making point predictions of forecasts. A matrix specified here will be ignored when credible intervals are created. This option only exists to facilitate computation of many one-at-a-time point predictions.

## Details

Analysts should use this function if they want to make kriged predictions for observations at new locations. To do this, researchers first must estimate a model using the `metropolis.krige` function.

After estimating the model, the first three arguments present covariate and location information for the observations to be predicted. The next five arguments provide information from the model fitted with `metropolis.krige`, including the outcome variable, predictor variables, coordinates, and the model results themselves. Users also should ensure that `powered.exp` has the same value in both `metropolis.krige` and `krige.pred`. (Note that the defaults of both functions are the same, so omission in both is acceptable.)

By default, the function uses median values of parameters to make a single point prediction for every kriged data point. However, if the user specifies a probability with the `credible` option, then the function will determine the predictions for all iterations of the MCMC sample. The point estimates will then be a median of these predictions, and a credible interval will be returned based on percentiles. Note that estimating a credible interval is substantially more intensive computationally, but has the benefit of reporting uncertainty in predictions.

## Value

Returns an object of class `matrix` with one prediction per row. By default the matrix has one column, as only point predictions are returned. If the `credible` option is specified, there are three columns respectively indicating a point estimate (median prediction from MCMC), lower bound of the credible interval, and upper bound of the credible interval.

## References

Sudipto Banerjee, Bradley P. Carlin, and Alan E. Gelfand. 2015. *Hierarchical Modeling and Analysis for Spatial Data*. 2nd ed. Boca Raton, FL: CRC Press.

## Examples

```
#Examine Data
summary(ContrivedData)

#Initial OLS Model
contrived.ols<-lm(y~x.1+x.2,data=ContrivedData);summary(contrived.ols)

#Define Covariate Matrix
covariates<-cbind(1,ContrivedData$x.1,ContrivedData$x.2)

#set seed
set.seed(1241060320)
```

```

#For simple illustration, we set to few iterations.
#In this case, a 10,000-iteration run converges to the true parameters.
#If you have considerable time and hardware, delete the # on the next line.
#10,000 iterations took 39 min. with 8 GB RAM & a 1.5 GHz Quad-Core processor.
M<-8
#M<-10000

#Run the Full Model
contrived.run<-metropolis.krige(y=ContrivedData$y,X=covariates,range.tol=0.05,
  east=ContrivedData$s.1,north=ContrivedData$s.2,mcmc.samples=M)

#Delete 20% for Burn-In
contrived.run<-contrived.run[(ceiling(0.2*M)+1):M,]

#examine results against true coefficients
TRUTH<-c(0.5,2.5,0.5,0,1,2)
rbind(apply(contrived.run,2,quantile,c(.5,.05,.95)),TRUTH)

#Predictive Data for Three Hypothetical people
euler<-c(1,0.2,0.7)
archimedes<-c(1,0.3,0.1)
pythagoras<-c(1,0.1,0.4)
mathematicians<-rbind(euler,archimedes,pythagoras)
basel<-c(0.1,0.8)
sicily<-c(0.4,0.1)
samos<-c(0.1,0.4)
new.locations<-rbind(basel,sicily,samos)
colnames(new.locations)<-c("eastings","northings")

# Make predictions from median parameter values:
median.pred<-krige.pred(pred.x=mathematicians,
  pred.east=new.locations[, "eastings"],pred.north=new.locations[, "northings"],
  train.y=ContrivedData$y,train.x=covariates,train.east=ContrivedData$s.1,
  train.north=ContrivedData$s.2,mcmc.iter=contrived.run)
median.pred

# Make predictions with 90% credible intervals:
cred.pred<-krige.pred(pred.x=mathematicians,
  pred.east=new.locations[, "eastings"],pred.north=new.locations[, "northings"],
  train.y=ContrivedData$y,train.x=covariates,train.east=ContrivedData$s.1,
  train.north=ContrivedData$s.2,mcmc.iter=contrived.run,credible=0.9)
cred.pred

```

---

krige.update

---

*Update Kriging Models*


---

## Description

This function updates the MCMC iterations associated with the kriging models.

**Usage**

```
krige.update(krige.mat, mcmc.samples=100, accepted.beta, accepted.nugget,
             accepted.decay, accepted.psill, y, X, east, north, powered.exp=2,
             spatial.share=0.5, range.share=0.5, beta.var=10, range.tol=0.05, b.tune=1.0,
             nugget.tune=10.0, psill.tune=1.0)
```

**Arguments**

<code>krige.mat</code>	The output MCMC matrix from <code>metropolis.krige()</code>
<code>mcmc.samples</code>	Number of MCMC iterations to be run for updating
<code>accepted.beta</code>	The <code>accepted.beta</code> from the previous run
<code>accepted.nugget</code>	The <code>accepted.nugget</code> from the previous run
<code>accepted.decay</code>	The <code>accepted.decay</code> from the previous run
<code>accepted.psill</code>	The <code>accepted.psill</code> from the previous run
<code>y</code>	The dependent variable that was used in the previous model. The same as in <code>metropolis.krige()</code>
<code>X</code>	The matrix of independent variables used in the previous model. The same as in <code>metropolis.krige()</code>
<code>east</code>	Vector of eastings for all observations. The same as in <code>metropolis.krige()</code>
<code>north</code>	Vector of northings for all observations. The same as in <code>metropolis.krige()</code>
<code>powered.exp</code>	A powered exponential correlation structure for the data. The same as in <code>metropolis.krige()</code>
<code>spatial.share</code>	Prior for proportion of unexplained variance that is spatial in nature. The same as in <code>metropolis.krige()</code>
<code>range.share</code>	Prior for the effective range term, as a proportion of the maximum distance in the data. The same as in <code>metropolis.krige()</code>
<code>beta.var</code>	Prior for the variance on zero-meanded normal priors on the regression coefficients. The same as in <code>metropolis.krige()</code>
<code>range.tol</code>	Tolerance term for setting the effective range. The same as in <code>metropolis.krige()</code>
<code>b.tune</code>	Tuning parameter for candidate generation of regression coefficients. The same as in <code>metropolis.krige()</code>
<code>nugget.tune</code>	Tuning parameter for candidate generation of the nugget term ( $\tau^2$ ). The same as in <code>metropolis.krige()</code>
<code>psill.tune</code>	Tuning parameter for candidate generation of the partial sill term ( $\sigma^2$ ). The same as in <code>metropolis.krige()</code>

**Details**

This function updates the previous run by `metropolis.krige` since MCMC iterations need to run typically long to gurantee convergence.

`krige.mat` is the output MCMC matrix of `metropolis.krige()` from the previous run. Analysts can specify the number of iterations for update run by `mcmc.samples`. The acceptance rates for *beta*, *nugget*, *decay*, and *psill* from the previous model also need to be specified to make sure

consistency. Other arguments should also be specified as the same as in the `metropolis.krige()`, including `y`, `X`, `east`, `north`, `powered.exp`, `spatial.share`, `range.share`, `beta.var`, `range.tol`, `b.tune`, `nugget.tune`, `psill.tune`.

The returned object is the same as `metropolis.krige()`, a matrix of sampled values from the posterior distribution. Rows represent the iteration number, and ideally the user will discard the first several rows as burn-in. Columns represent the parameter, so summarizing the matrix by column offers summaries of the model's results.

## Value

Returns an object of class `matrix` that includes all iterations of the Metropolis-Hastings sampler. Each column of the matrix represents a different parameter—starting with `tau2`, `phi`, and `sigma2` before listing regression coefficients. Each row represents another iteration of the MCMC sampler. Summarizing the matrix by column offers summaries of the marginal posterior distribution by parameter.

## Examples

```
## Not run:
#Set seed
set.seed(1241060320)

#For simple illustration, we set to few iterations.
#But much more are needed for the realistic estimation.
#In this case, a 10,000-iteration run converges to the true parameters.
M1<-5

contrived.run<-metropolis.krige(y=ContrivedData$y,X=covariates,range.tol=0.05,
                               east=ContrivedData$s.1,north=ContrivedData$s.2,mcmc.samples=M1)

#Acceptance rates for the first 5 iterations.
#"Acceptance percentages: Coefficients=50%. Nugget=75%. Decay=75%.
#Partial sill=75%."

#Update for another 5 iterations.
M2<-5

contrived.run.update<-update.krige(krige.mat=contrived,mcmc.samples=M2,
                                   accepted.beta=0.5,accepted.nugget=0.75,accepted.decay=0.75,
                                   accepted.psill=0.75,y=ContrivedData$y,X=covariates,range.tol=0.05,
                                   east=ContrivedData$s.1,north=ContrivedData$s.2)

## End(Not run)
```

## Description

These data present measures of ideology in 2010 for the districts for lower chambers of state legislatures, recorded as the variable `krige.lower`. 49 states' chambers are covered—the Nebraska Unicameral is omitted here to be included in the file `upperCombined`. Forecasts are based on a kriging model fitted over the 2008 Cooperative Congressional Election Survey (CCES), paired with predictive data from the 2010 Census. Each district's public ideology is paired with a measure of the ideology of the State House member (or members) from the district (update from Shor and McCarty 2011).

## Usage

```
data(lowerCombined)
```

## Format

The `lowerCombined` dataset has 5446 observations and 10 variables.

`st` Two-letter postal abbreviation for the state.

`lower` The state legislative district number (lower chamber).

`STATEA` The FIPS code for the state.

`krige.lower` The ideology of the average citizen in the district.

`lowerKluge` Combined index of `STATEA` followed by `lower`.

`krige.lower.var` The variance of ideology among the district's citizens.

`name` Last name of the state legislator, followed by first name and middle initial.

`party` Political party of the legislator. D=Democrat, R=Republican, X=Other.

`st_id` Temporary identifier variable. DO NOT USE.

`np_score` Ideology score for the state legislator (lower chamber). Higher values are usually interpreted as more right-wing, with lower values as more left-wing.

## Source

Ansolabehere, Stephen. 2011. "CCES, Common Content, 2008." Ver. 4.

<http://hdl.handle.net/1902.1/14003>

Minnesota Population Center. 2011. *National Historical Geographic Information System: Version 2.0*. Minneapolis, MN: University of Minnesota.

'<https://www.nhgis.org>'

Shor, Boris and Nolan M. McCarty. 2011. "The Ideological Mapping of American Legislatures." *American Political Science Review* 105(3):530-551.

## References

Jeff Gill. 2020. Measuring Constituency Ideology Using Bayesian Universal Kriging. *State Politics & Policy Quarterly*.



**Examples**

```
# Descriptive Statistics
summary(lowerCombined)

# Correlate Senators' DW-NOMINATE Scores with Public Opinion Ideology
cor(lowerCombined$np_score, lowerCombined$krige.lower, use="complete.obs")

# Plot Legislators' DW-NOMINATE Scores against Public Opinion Ideology
plot(y=lowerCombined$np_score, x=lowerCombined$krige.lower,
     xlab="District Ideology (Kriging)", ylab="Legislator Ideology (Shor & McCarty)",
     main="State Legislatures: Lower Chambers", type="n")#
points(y=lowerCombined$np_score[lowerCombined$party=="R"],
       x=lowerCombined$krige.lower[lowerCombined$party=="R"], pch=".", col="red")
points(y=lowerCombined$np_score[lowerCombined$party=="D"],
       x=lowerCombined$krige.lower[lowerCombined$party=="D"], pch=".", col="blue")
```

metropolis.krige

*Sampling Technique Using Metropolis-Hastings***Description**

This function performs Metropolis-Hastings sampling for a linear model specified over point-referenced geospatial data. It returns MCMC iterations, with which results of the geospatial linear model can be summarized.

**Usage**

```
metropolis.krige(y,X,east,north,powered.exp=2,mcmc.samples=100,
  spatial.share=0.5,range.share=0.5,beta.var=10,
  range.tol=0.05,b.tune=1.0,nugget.tune=10.0,psill.tune=1.0)
```

**Arguments**

<code>y</code>	The dependent variable that is used in the kriging model.
<code>X</code>	The matrix of independent variables used in the kriging model.
<code>east</code>	Vector of eastings for all observations.
<code>north</code>	Vector of northings for all observations.
<code>powered.exp</code>	This exponent, which must be greater than 0 and less than or equal to 2, specifies a powered exponential correlation structure for the data. One widely used specification is setting this to 1, which yields an exponential correlation structure. Another common specification is setting this to 2 (the default), which yields a Gaussian correlation structure.
<code>mcmc.samples</code>	Number of MCMC iterations.
<code>spatial.share</code>	Prior for proportion of unexplained variance that is spatial in nature. Must be greater than 0 and less than 1. Defaults to an even split, valued at 0.5.

range.share	Prior for the effective range term, as a proportion of the maximum distance in the data. Users should choose the proportion of distance at which they think the spatial correlation will become negligible. Must be greater than 0. Values greater than 1 are permitted, but users should recognize that this implies that meaningful spatial correlation would persist outside of the convex hull of data. Defaults to half the maximum distance, valued at 0.5.
beta.var	Prior for the variance on zero-meanded normal priors on the regression coefficients. Must be greater than 0. Defaults to 10.
range.tol	Tolerance term for setting the effective range. At the distance where the spatial correlation drops below this term, it is judged that the effective range has been met. The default value is the commonly-used 0.05. Must be greater than 0 and less than 1.
b.tune	Tuning parameter for candidate generation of regression coefficients that must be greater than 0. A value of 1 means that draws will be based on the variance-covariance matrix of coefficients from OLS. Larger steps are taken for values greater than 1, and smaller steps are taken for values from 0 to 1. Defaults to 1.0.
nugget.tune	Tuning parameter for candidate generation of the nugget term ( $\tau^2$ ) that must be greater than 0. A value of 1 means that draws will be based on the typical variance of an inverse gamma distribution. <i>Smaller</i> steps are taken for values <i>greater</i> than 1, and <i>larger</i> steps are taken for <i>decimal</i> values from 0 to 1. Defaults to 10.0.
psill.tune	Tuning parameter for candidate generation of the partial sill term ( $\sigma^2$ ) that must be greater than 0. A value of 1 means that draws will be based on the typical variance of an inverse gamma distribution. <i>Smaller</i> steps are taken for values <i>greater</i> than 1, and <i>larger</i> steps are taken for <i>decimal</i> values from 0 to 1. Defaults to 1.0.

## Details

Analysts should use this function if they want to estimate a linear regression model in which each observation can be located at points in geographic space. That is, each observation is observed for a set of coordinates in eastings & northings or longitude & latitude.

Researchers must specify their model in the following manner: `y` should be a column vector of the dependent variable. `X` should be a matrix that includes all independent variables in the model, including a constant vector to estimate an intercept term. `east` should be a vector of all west-east coordinates for observations (ideally eastings but possibly longitude). `north` should be a vector of all north-south coordinates for observations (ideally northings but possibly latitude). `mcmc.samples` is the number of iterations to sample from the posterior distribution using the Metropolis-Hastings algorithm. This defaults to 100 iterations, but many more iterations would normally be preferred. The output of the function prints the proportion of candidate values for the coefficients and for the variance terms accepted by the Metropolis-Hastings algorithm. Particularly low or high acceptance rates respectively may indicate slow mixing (requiring more iterations) or a transient state (leading to nonconvergence), so additional messages will print for extreme acceptance rates. Users may want to adjust the tuning parameters `b.tune`, `nugget.tune`, or `psill.tune`, or perhaps the tolerance parameter `range.tol` if the acceptance rate is too high or too low.

The returned value is a matrix of sampled values from the posterior distribution. Rows represent the iteration number, and ideally the user will discard the first several rows as burn-in. Columns represent the parameter, so summarizing the matrix by column offers summaries of the model's results.

## Value

Returns an object of class `matrix` that includes all iterations of the Metropolis-Hastings sampler. Each column of the matrix represents a different parameter—starting with `tau2`, `phi`, and `sigma2` before listing regression coefficients. Each row represents another iteration of the MCMC sampler. Summarizing the matrix by column offers summaries of the marginal posterior distribution by parameter.

## References

Jeff Gill. 2020. Measuring Constituency Ideology Using Bayesian Universal Kriging. *State Politics & Policy Quarterly*.

## Examples

```
#Examine Data
summary(ContrivedData)

#Initial OLS Model
contrived.ols<-lm(y~x.1+x.2,data=ContrivedData);summary(contrived.ols)

#Define Covariate Matrix
covariates<-cbind(1,ContrivedData$x.1,ContrivedData$x.2)

#set seed
set.seed(1241060320)

#For simple illustration, we set to few iterations.
#In this case, a 10,000-iteration run converges to the true parameters.
#If you have considerable time and hardware, delete the # on the next line.
#10,000 iterations took 39 min. with 8 GB RAM & a 1.5 GHz Quad-Core processor.
M<-8
#M<-10000

#Run the Full Model
contrived.run<-metropolis.krige(y=ContrivedData$y,X=covariates,range.tol=0.05,
  east=ContrivedData$s.1,north=ContrivedData$s.2,mcmc.samples=M)

#Delete 20% for Burn-In
contrived.run<-contrived.run[(ceiling(0.2*M)+1):M,]

#examine results against true coefficients
TRUTH<-c(0.5,2.5,0.5,0,1,2)
rbind(apply(contrived.run,2,quantile,c(.5,.05,.95)),TRUTH)

#Convergence Diagnostics: Geweke and Heidelberger-Welch
#Note that the second (commented) version of Geweke is more typical
```

```

#of a 10,000 iteration run.
geweke(contrived.run,early.prop=0.5)
#geweke(contrived.run)
heidel.welch(contrived.run)

#Examine the Parametric Semivariogram Graphically
raw.semivar<-semivariogram(x=ContrivedData$y,east=ContrivedData$s.1,
  north=ContrivedData$s.2)
resid.semivar<-semivariogram(x=contrived.ols$residuals,east=ContrivedData$s.1,
  north=ContrivedData$s.2,draw.plot=FALSE)
points(resid.semivar,pch=3,col='blue')
lines(exponential.semivariogram(nugget=median(contrived.run["tau2"]),
  decay=median(contrived.run["phi"]),partial.sill=median(contrived.run["sigma2"]),
  distance=as.numeric(names(resid.semivar))),col='red')

#Predictive Data for Three Hypothetical people
euler<-c(1,0.2,0.7)
archimedes<-c(1,0.3,0.1)
pythagoras<-c(1,0.1,0.4)
mathematicians<-rbind(euler,archimedes,pythagoras)
basel<-c(0.1,0.8)
sicily<-c(0.4,0.1)
samos<-c(0.1,0.4)
new.locations<-rbind(basel,sicily,samos)
colnames(new.locations)<-c("eastings","northings")

# Make predictions from median parameter values:
median.pred<-krige.pred(pred.x=mathematicians,
  pred.east=new.locations[, "eastings"],pred.north=new.locations[, "northings"],
  train.y=ContrivedData$y,train.x=covariates,train.east=ContrivedData$s.1,
  train.north=ContrivedData$s.2,mcmc.iter=contrived.run)
median.pred

# Make predictions with 90% credible intervals:
cred.pred<-krige.pred(pred.x=mathematicians,
  pred.east=new.locations[, "eastings"],pred.north=new.locations[, "northings"],
  train.y=ContrivedData$y,train.x=covariates,train.east=ContrivedData$s.1,
  train.north=ContrivedData$s.2,mcmc.iter=contrived.run,credible=0.9)
cred.pred

```

---

NYcity\_subset

New York City CCES Respondents in 2008

---

## Description

These data are a subset of the 2008 Cooperative Congressional Election Survey (CCES) Common Content. Only 568 respondents from New York City are included, with predictors drawn from Gill's (2020) model of self-reported ideology. The CCES data are merged with predictors on geographic location based on ZIP codes (from ArcGIS & TomTom) and county ruralism (from the USDA).

**Usage**

```
data(NYcity_subset)
```

**Format**

The NYcity\_subset dataset has 568 observations and 26 variables.

state The state abbreviation of the respondent's residence.

zip The respondent's ZIP code.

age The age of the respondent in years.

female An indicator of whether the respondent is female.

ideology The respondent's self-reported ideology on a scale of 0 (liberal) to 100 (conservative).

educ The respondent's level of education. 0=No Highschool, 1=High School Graduate, 2=Some College, 3=2-year Degree, 4=4-year degree, 5=Post-Graduate.

race The respondent's race. 1=White, 2=African American, 3=Nonwhite & nonblack.

empstat The respondent's employment status. 1=employed, 2=unemployed, 3=not in workforce.

ownership Indicator for whether the respondent owns his or her own home.

inc14 The respondent's self reported income. 1=Less than \$10,000, 2=\$10,000-\$14,999, 3=\$15,000-\$19,000, 4=\$20,000-\$24,999, 5=\$25,000-\$29,999, 6=\$30,000-\$39,999, 7=\$40,000-\$49,999, 8=\$50,000-\$59,999, 9=\$60,000-\$69,999, 10=\$70,000-\$79,999, 11=\$80,000-\$89,999, 12=\$100,000-\$119,999, 13=\$120,000-\$149,999, 14=\$150,000 or more.

catholic Indicator for whether the respondent is Catholic.

mormon Indicator for whether the respondent is Mormon.

orthodox Indicator for whether the respondent is Orthodox Christian.

jewish Indicator for whether the respondent is Jewish.

islam Indicator for whether the respondent is Muslim.

mainline Indicator for whether the respondent is Mainline Christian.

evangelical Indicator for whether the respondent is Evangelical Christian.

FIPS\_Code FIPS code of the respondent's state.

rural Nine-point USDA scale of the ruralism of each county, with 0 meaning the most urban and 8 meaning the most rural.

zipPop Indicates the population of the respondent's ZIP code.

zipLandKM Indicates the land area in square kilometers of the respondent's ZIP code.

weight Survey weights created by the CCES.

cd The congressional district the respondent resides in.

fipsCD Index that fuses the state FIPS code in the first two digits and the congressional district number in the last two digits.

northings Indicates the geographical location of the respondent in kilometer-based northings.

eastings Indicates the geographical location of the respondent in kilometer-based eastings.

## Source

Ansolabehere, Stephen. 2011. "CCES, Common Content, 2008." Ver. 4.

<http://hdl.handle.net/1902.1/14003>

ArcGIS. 2012. "USA ZIP Code Areas."

<https://www.arcgis.com/home/item.html?id=8d2012a2016e484dafaac0451f9aea24>

United States Department of Agriculture. 2013. "2013 Rural-Urban Continuum Codes."

<https://www.ers.usda.gov/data-products/rural-urban-continuum-codes.aspx>

## References

Jeff Gill. 2020. Measuring Constituency Ideology Using Bayesian Universal Kriging. *State Politics & Policy Quarterly*.

## Examples

```
#Data
ny <- NYcity_subset

#data cleaning
ny$cathOrth<-ny$catholic+ny$orthodox
ny$consRelig<-ny$mormon+ny$evangelical
ny$jewMus<-ny$jewish+ny$islam

# Explanatory Variable Matrix
psrm.data <-cbind(1, ny$age, ny$educ, I(ny$age*ny$educ), as.numeric(ny$race==2),
  as.numeric(ny$race==3), ny$female, I(as.numeric(ny$race==2)*ny$female),
  I(as.numeric(ny$race==3)*ny$female), ny$cathOrth, ny$consRelig,
  ny$jewMus, ny$mainline, ny$rural, ny$ownership,
  as.numeric(ny$empstat==2), as.numeric(ny$empstat==3),ny$inc14)
dimnames(psrm.data)[[2]] <- c("Intercept", "Age", "Education", "Age.education",
  "African.American", "Nonwhite.nonblack","Female",
  "African.American.female", "Nonwhite.nonblack.female",
  "Catholic.Orthodox", "Evang.Mormon", "Jewish.Muslim",
  "Mainline","Ruralism", "Homeowner", "Unemployed",
  "Not.in.workforce","Income")

# Outcome Variable
ideo <- matrix(ny$ideology,ncol=1)

# Set Number of Iterations:
# WARNING: This example was tuned on Amazon Web Services (EC2) over many hours
# with 150,000 iterations--a strain in 2020 for most desktop machines.
# A test with few iterations allows illustration.
M<-5
#M<-150000
set.seed(1,kind="Mersenne-Twister")

# Estimate the Model
out.mat <- metropolis.krige(y=ideo, X=psrm.data, east=ny$eastings, north=ny$northings,
```

*New York State CCES Respondents in 2008*

## Description

These data are a subset of the 2008 Cooperative Congressional Election Survey (CCES) Common Content. Only 1108 respondents from the state of New York are included, with predictors drawn from Gill's (2020) model of self-reported ideology. The CCES data are merged with predictors on geographic location based on ZIP codes (from ArcGIS & TomTom) and county ruralism (from the USDA).

## Usage

```
data(NY_subset)
```

## Format

The NY\_subset dataset has 1108 observations and 26 variables.

`state` The state abbreviation of the respondent's residence.

`zip` The respondent's ZIP code.

`age` The age of the respondent in years.

`female` An indicator of whether the respondent is female.

`ideology` The respondent's self-reported ideology on a scale of 0 (liberal) to 100 (conservative).

`educ` The respondent's level of education. 0=No Highschool, 1=High School Graduate, 2=Some College, 3=2-year Degree, 4=4-year degree, 5=Post-Graduate.

`race` The respondent's race. 1=White, 2=African American, 3=Nonwhite & nonblack.

`empstat` The respondent's employment status. 1=employed, 2=unemployed, 3=not in workforce.

`ownership` Indicator for whether the respondent owns his or her own home.

`inc14` The respondent's self reported income. 1=Less than \$10,000, 2=\$10,000-\$14,999, 3=\$15,000-\$19,000, 4=\$20,000-\$24,999, 5=\$25,000-\$29,999, 6=\$30,000-\$39,999, 7=\$40,000-\$49,999, 8=\$50,000-\$59,999, 9=\$60,000-\$69,999, 10=\$70,000-\$79,999, 11=\$80,000-\$89,999, 12=\$100,000-\$119,999, 13=\$120,000-\$149,999, 14=\$150,000 or more.

`catholic` Indicator for whether the respondent is Catholic.

`mormon` Indicator for whether the respondent is Mormon.

`orthodox` Indicator for whether the respondent is Orthodox Christian.

`jewish` Indicator for whether the respondent is Jewish.

`islam` Indicator for whether the respondent is Muslim.

`mainline` Indicator for whether the respondent is Mainline Christian.

`evangelical` Indicator for whether the respondent is Evangelical Christian.

`FIPS_Code` FIPS code of the respondent's state.

`rural` Nine-point USDA scale of the ruralism of each county, with 0 meaning the most urban and 8 meaning the most rural.

`zipPop` Indicates the population of the respondent's ZIP code.

`zipLandKM` Indicates the land area in square kilometers of the respondent's ZIP code.



weight Survey weights created by the CCES.

cd The congressional district the respondent resides in.

fipsCD Index that fuses the state FIPS code in the first two digits and the congressional district number in the last two digits.

northings Indicates the geographical location of the respondent in kilometer-based northings.

eastings Indicates the geographical location of the respondent in kilometer-based eastings.

## Source

Ansolabehere, Stephen. 2011. "CCES, Common Content, 2008." Ver. 4.  
<http://hdl.handle.net/1902.1/14003>

ArcGIS. 2012. "USA ZIP Code Areas."  
<https://www.arcgis.com/home/item.html?id=8d2012a2016e484dafaac0451f9aea24>

United States Department of Agriculture. 2013. "2013 Rural-Urban Continuum Codes."  
<https://www.ers.usda.gov/data-products/rural-urban-continuum-codes.aspx>

## References

Jeff Gill. 2020. Measuring Constituency Ideology Using Bayesian Universal Kriging. *State Politics & Policy Quarterly*.

## Examples

```
#Data
ny <- NY_subset

#data cleaning
ny$cathOrth<-ny$catholic+ny$orthodox
ny$consRelig<-ny$mormon+ny$evangelical
ny$jewMus<-ny$jewish+ny$islam

# Explanatory Variable Matrix
psrm.data <-cbind(1, ny$age, ny$educ, I(ny$age*ny$educ), as.numeric(ny$race==2),
  as.numeric(ny$race==3), ny$female, I(as.numeric(ny$race==2)*ny$female),
  I(as.numeric(ny$race==3)*ny$female), ny$cathOrth, ny$consRelig,
  ny$jewMus, ny$mainline, ny$rural, ny$ownership,
  as.numeric(ny$empstat==2), as.numeric(ny$empstat==3),ny$inc14)
dimnames(psrm.data)[[2]] <- c("Intercept", "Age", "Education", "Age.education",
  "African.American", "Nonwhite.nonblack","Female",
  "African.American.female", "Nonwhite.nonblack.female",
  "Catholic.Orthodox", "Evang.Mormon", "Jewish.Muslim",
  "Mainline", "Ruralism", "Homeowner", "Unemployed",
  "Not.in.workforce", "Income")

# Outcome Variable
ideo <- matrix(ny$ideology,ncol=1)

# Set Number of Iterations:
```

```

# WARNING: 20 iterations is intensive on many machines.
# This example was tuned on Amazon Web Services (EC2) over many hours
# with 20,000 iterations--unsuitable in 2020 for most desktop machines.
M<-20#000 #The comment character constrains this to 20 iterations.
set.seed(1,kind="Mersenne-Twister")

## Not run:
# Estimate the Model
out.mat <- metropolis.krige(y=ideo,X=psrm.data,east=ny$eastings,north=ny$northings,
  powered.exp=1,mcmc.samples=M,spatial.share=0.31,range.share=0.23,beta.var=10,
  range.tol=0.01, b.tune=0.1, nugget.tune=20, psill.tune=5)

# Discard first 20% of Iterations as Burn-In (User Discretion Advised).
out.mat <- out.mat[(ceiling(0.2*M)+1):M,]

# Summarize Results
apply(out.mat,2,quantile,c(0.5,0.05,0.95))

#Convergence Diagnostics: Geweke and Heidelberger-Welch
#Note that the second (commented) version of Geweke is more typical
#of a 20,000 iteration run.
geweke(out.mat,early.prop=0.5)
#geweke(out.mat)
heidel.welch(out.mat)

# Draw Semivariogram
state.ols<-lm(ideo~psrm.data-1)
graph.terms<-apply(out.mat[,1:3],2,quantile,0.5)
raw.semivar<-semivariogram(x=ideo,east=ny$eastings, north=ny$northings)
resid.semivar<-semivariogram(x=state.ols$residuals,east=ny$eastings,
  north=ny$northings,draw.plot=FALSE)
points(resid.semivar,pch=3,col='blue')
lines(exponential.semivariogram(nugget=graph.terms[1],decay=graph.terms[2],
  partial.sill=graph.terms[3],
  distance=as.numeric(names(resid.semivar)),power=1),col='red')

## End(Not run)

```

---

semivariogram

*Empirical Semivariogram*


---

## Description

This function computes the empirical semivariance for a spatially-distributed variable. Based on the user's chosen level of coarsening, the semivariance is presented for various distances.

## Usage

```
semivariogram(x,east,north,bins=13,draw.plot=TRUE)
```

**Arguments**

<code>x</code>	The spatial variable for which the semivariogram is desired.
<code>east</code>	Vector of eastings for the observations of <code>x</code> .
<code>north</code>	Vector of northings for the observations of <code>x</code> .
<code>bins</code>	Number of bins into which distances should be divided. The observed distances will be split at equal intervals, and the semivariance will be computed within each interval. Defaults to 13 intervals.
<code>draw.plot</code>	By default, a graph of the empirical semivariogram appears with a run of the function. Changing this setting to <code>FALSE</code> omits this and only returns semivariogram values.

**Details**

Semivariance is equal to half of the variance of the difference in a variable's values at a given distance. That is, the semivariance is defined as:  $\gamma(h) = 0.5 * E[X(s+h) - X(s)]^2$ , where  $X$  is the variable of interest,  $s$  is a location, and  $h$  is the distance from  $s$  to another location.

An empirical semivariogram is a good tool for diagnosing the kind of spatial relationship that can best describe the data. With a view of the empirical semivariogram, a user can consult images of parametric semivariograms to determine whether an exponential, Gaussian, or other powered exponential function fit the data well, or if another style of semivariogram works better. Examining this also allows the user to develop priors such as the approximate split in variance between the nugget and partial sill as well as the approximate distance of the effective range. Semivariograms are explicitly tied to a corresponding spatial correlation function, so determining the former automatically implies the latter. See Banerjee, Carlin, and Gelfand for a fuller explanation, as well as a guidebook to semivariogram diagnosis (2015, 26-30).

**Value**

Returns an object of class `numeric` with each bin's value of the semivariance. Each entry is labeled by the upper bound of the bin. The lower bound of each bin is 0 for the first one, and the prior upper bound for all subsequent bins.

**References**

Sudipto Banerjee, Bradley P. Carlin, and Alan E. Gelfand. 2015. *Hierarchical Modeling and Analysis for Spatial Data*. 2nd ed. Boca Raton, FL: CRC Press.

**Examples**

```
#Examine Data
summary(ContrivedData)

#Initial OLS Model
contrived.ols<-lm(y~x.1+x.2,data=ContrivedData);summary(contrived.ols)

#Graphically Examine the Semivariogram of Raw Data and Residuals
raw.semivar<-semivariogram(x=ContrivedData$y,east=ContrivedData$s.1,
  north=ContrivedData$s.2)
```

```
resid.semivar<-semivariogram(x=contrived.ols$residuals,east=ContrivedData$s.1,
  north=ContrivedData$s.2,draw.plot=FALSE)
points(resid.semivar,pch=3,col='blue')
```

---

stateCombined

*State Public Opinion Ideology in 2010*


---

## Description

These data present measures of ideology in 2010 for the 50 American states, recorded as the variable `krige.state`. Forecasts are based on a kriging model fitted over the 2008 Cooperative Congressional Election Survey (CCES), paired with predictive data from the 2010 Census. Each state is listed twice, as each state's public ideology is paired with the DW-NOMINATE common space score of each of its two senators in 2011 (update from McCarty, Poole and Rosenthal 1997).

## Usage

```
data(stateCombined)
```

## Format

The `stateCombined` dataset has 100 observations (2 each for 50 states) and 13 variables.

`STATEA` The FIPS code for the state.

`krige.state` The ideology of the average citizen in the state.

`krige.state.var` The variance of ideology among the state's citizens.

`cong` The term of Congress studied—112 for this dataset.

`idno` Identification number for the senator—ICPSR numbers continued by Poole & Rosenthal.

`state` The ICPSR code for the state.

`cd` The congressional district number—0 for senators.

`statenm` The first seven letters of the state's name.

`party` Political party of the senator. 100=Democrat, 200=Republican, 328=Independent.

`name` Last name of the senator, followed by first name if ambiguous.

`dwnom1` First dimension DW-NOMINATE common space score for the senator. Higher values are usually interpreted as more right-wing, with lower values as more left-wing.

`stateCD` Combined index of `STATEA` followed by `cd`.

`obama` Barack Obama's percentage of the two-party vote in the state in 2012.

## Source

Ansolabehere, Stephen. 2011. "CCES, Common Content, 2008." Ver. 4.  
<http://hdl.handle.net/1902.1/14003>

McCarty, Nolan M., Keith T. Poole and Howard Rosenthal. 1997. *Income Redistribution and the Realignment of American Politics*. American Enterprise Institute Studies on Understanding Economic Inequality. Washington: AEI Press.

Minnesota Population Center. 2011. *National Historical Geographic Information System: Version 2.0*. Minneapolis, MN: University of Minnesota.  
<https://www.nhgis.org>

## References

Jeff Gill. 2020. Measuring Constituency Ideology Using Bayesian Universal Kriging. *State Politics & Policy Quarterly*.

## Examples

```
# Descriptive Statistics
summary(stateCombined)

# Correlate Senators' DW-NOMINATE Scores with Public Opinion Ideology
cor(stateCombined$krige.state, stateCombined$dwnom1)

# Plot Senators' DW-NOMINATE Scores against Public Opinion Ideology
plot(y=stateCombined$dwnom1, x=stateCombined$krige.state,
     xlab="State Ideology (Kriging)", ylab="Legislator Ideology (1st Dim., Common Space)",
     main="U.S. Senate", type="n")
points(y=stateCombined$dwnom1[stateCombined$party==200],
       x=stateCombined$krige.state[stateCombined$party==200], pch="R", col="red")
points(y=stateCombined$dwnom1[stateCombined$party==100],
       x=stateCombined$krige.state[stateCombined$party==100], pch="D", col="blue")
```

---

upperCombined

*State Legislative District (Upper Chambers) Public Opinion Ideology  
in 2010*

---

## Description

These data present measures of ideology in 2010 for the districts for upper chambers of state legislatures, recorded as the variable `krige.upper`. All 50 states' chambers are covered (including the Nebraska Unicameral). Forecasts are based on a kriging model fitted over the 2008 Cooperative Congressional Election Survey (CCES), paired with predictive data from the 2010 Census. Each district's public ideology is paired with a measure of the ideology of the State Senate member from the district (update from Shor and McCarty 2011).

**Usage**

```
data(upperCombined)
```

**Format**

The upperCombined dataset has 1989 observations and 10 variables.

st Two-letter postal abbreviation for the state.

upper The state legislative district number (upper chamber).

STATEA The FIPS code for the state.

krige.upper The ideology of the average citizen in the district.

upperKluge Combined index of STATEA followed by upper.

krige.upper.var The variance of ideology among the district's citizens.

name Last name of the state legislator, followed by first name and middle initial.

party Political party of the legislator. D=Democrat, R=Republican, X=Other.

st\_id Temporary identifier variable. DO NOT USE.

np\_score Ideology score for the state legislator (upper chamber). Higher values are usually interpreted as more right-wing, with lower values as more left-wing.

**Source**

Ansolabehere, Stephen. 2011. "CCES, Common Content, 2008." Ver. 4.

<http://hdl.handle.net/1902.1/14003>

Minnesota Population Center. 2011. *National Historical Geographic Information System: Version 2.0*. Minneapolis, MN: University of Minnesota.

'<https://www.nhgis.org>'

Shor, Boris and Nolan M. McCarty. 2011. "The Ideological Mapping of American Legislatures." *American Political Science Review* 105(3):530-551.

**References**

Jeff Gill. 2020. Measuring Constituency Ideology Using Bayesian Universal Kriging. *State Politics & Policy Quarterly*.

**Examples**

```
# Descriptive Statistics
summary(upperCombined)

# Correlate Senators' DW-NOMINATE Scores with Public Opinion Ideology
cor(upperCombined$np_score, upperCombined$krige.upper, use="complete.obs")

# Plot Legislators' DW-NOMINATE Scores against Public Opinion Ideology
plot(y=upperCombined$np_score, x=upperCombined$krige.upper,
```

```

xlab="District Ideology (Kriging)", ylab="Legislator Ideology (Shor & McCarty)",
main="State Legislatures: Upper Chambers", type="n")
points(y=upperCombined$np_score[upperCombined$party=="R"],
x=upperCombined$krige.upper[upperCombined$party=="R"],pch=".",col="red")
points(y=upperCombined$np_score[upperCombined$party=="D"],
x=upperCombined$krige.upper[upperCombined$party=="D"],pch=".",col="blue")

```

WVwells

*West Virginia Oil and Gas Production in 2012*

### Description

These data are a subset of the West Virginia Geological and Economic Survey of 2014. They contain information on the coordinates of wells that yielded at least some quantity of natural gas in 2012. In addition to coordinates, the data contain information on well ownership and operation, rock pressure at the well, elevation of the well, oil production, and gas production.

### Usage

```
data(WVwells)
```

### Format

The WVwells dataset has 1949 observations and 18 variables.

**APINum** A 10-digit number in the format assigned by the American Petroleum Institute (API), consisting of a 2-digit state code, a 3-digit county code with leading zeroes, and a 5-digit permit number with leading zeroes. Data Source: West Virginia Department of Environmental Protection, Office of Oil & Gas (WVDEP-OOG).

**CntyCode** A 3-digit numeric code, assigned in numeric order by county name. Data Source: The county code for a well is assigned by WVDEP-OOG, based on the well location.

**CntyName** The name of the county. Please see CntyCode (County Code) for a list of all West Virginia county names. Data Source: The county code for a well is assigned by WVDEP-OOG, based on the well location. The county name is a translation of the county code.

**Operator** The name of the operator who owns the well at the time of reporting. Data Source: WVDEP-OOG plat; verified on the WR-35 completion record.

**SurfaceOwn** The name of the owner of the surface property on which the well is located. Data Source: WVDEP-OOG plat; verified on the WR-35 completion record.

**MineralOwn** Mineral Owner: The name of the owner of the mineral rights where the well is located. Data Source: WVDEP-OOG plat.

**CompanyNum** The operator's serial number for the well. Data Source: WVDEP-OOG plat; verified on the WR-35 completion record.

**WellNum** The operator's number for the well on the surface property (farm). Data Source: WVDEP-OOG plat; verified on the WR-35 completion record.

UTMESrf Surface Location–Universal Transverse Mercator, Easting: The well location at the surface measured in meters to one decimal point, east of the central meridian in UTM Zone 17; datum: NAD83. Data Source: Taken directly from the plat if given as such. Otherwise, computed from the location reported on the plat. Suspect locations may be adjusted using various additional resources (e.g. topographic maps) if deemed necessary.

UTMNSrf Surface Location–Universal Transverse Mercator, Northing: The well location at the surface measured in meters to one decimal point, north of the equator in UTM Zone 17; datum: NAD83. Data Source: Taken directly from the plat if given as such. Otherwise, computed from the location reported on the plat. Suspect locations may be adjusted using various additional resources (e.g. topographic maps) if deemed necessary.

LonSrf Surface Location–Longitude: The well location at the surface measured to a precision of 6 decimal points, in degrees west of the Prime Meridian. Data Source: Taken directly from the plat if given as such. Otherwise, computed from the location reported on the plat. Suspect locations may be adjusted using various additional resources (e.g. topographic maps) if deemed necessary.

LatSrf Surface Location–Latitude: The well location at the surface measured to a precision of 6 decimal points, in degrees north of the equator. Data Source: Taken directly from the plat if given as such. Otherwise, computed from the location reported on the plat. Suspect locations may be adjusted using various additional resources (e.g. topographic maps) if deemed necessary.

Elevation Elevation: The height of the well in feet above mean sea level. Data Source: WVDEP-OOG plat; verified on the WR-35 completion record.

RockPres Formation Rock Pressure at Surface: The pressure measured at the surface usually after stimulation, in pounds per square inch (psi). Data Source: WVDEP-OOG WR-35 completion record, submitted by the operator to WVDEP-OOG.

GProd2012 2012 Gas Production Reported: The total gas production for the well for 2012 in thousands of cubic feet (MCF); includes all pay zones. Data Source: Production data reported by the operator to the State regulatory authority for Oil and Gas (WVDEP-OOG); WVGES obtained the data from WVDEP-OOG.

OProd2012 2012 Oil Production Reported: The total oil production for the well for 2012 in barrels (Bbl); includes all pay zones. Production data reported by the operator to the State regulatory authority for Oil and Gas (WVDEP-OOG); WVGES obtained the data from WVDEP-OOG.

logElevation Logarithm of Elevation.

logGProd2012 Logarithm of GProd2012.

## Source

West Virginia Geological and Economic Survey. 2014. "WVMarcellusWellsCompleted102014." Morgantown, WV.  
<http://www.wvgs.wvnet.edu/www/datastat/devshales.htm>

Accessed via: FracTracker. 2019. "West Virginia Oil & Gas Activity."  
<https://www.fractracker.org/map/us/west-virginia/>



## References

Jason S. Byers & Jeff Gill. N.D. "Applied Geospatial Data Modeling in the Big Data Era: Challenges and Solutions."

## Examples

```
# Descriptive Statistics
summary(WVwells)

# Record means of predictors:
# These are used BOTH to eliminate the intercept and to recover predictions later.
mean.logGas<-mean(WVwells$logGProd2012);mean.logGas
mean.logElevation<-mean(WVwells$logElevation);mean.logElevation
mean.RockPres<-mean(WVwells$RockPres);mean.RockPres

# Outcome Variable, De-Meaned
logGas <- matrix(WVwells$logGProd2012-mean.logGas,ncol=1)

# Explanatory Variable Matrix: DE-MEANED PREDICTORS AND NO CONSTANT TERM
# Because we deducted the mean from all predictors and the outcome,
# it is valid to do regression through the origin.
wv.pred <-cbind(WVwells$logElevation-mean.logElevation,WVwells$RockPres-mean.RockPres)
dimnames(wv.pred)[[2]] <- c("LogElevation", "RockPressure")

# OLS Model
fracking.ols<-lm(logGas~wv.pred-1)
summary(fracking.ols)

# Set Number of Iterations:
# WARNING: 50 iterations is intensive on many machines.
# This example was tuned on Amazon Web Services (EC2) over many hours
# with 5,000 iterations--unsuitable in 2020 for most desktop machines.
M<-50#00 #The comment character constrains this to 50 iterations.
set.seed(1000,kind="Mersenne-Twister")#SET SEED FOR CONSISTENCY

## Not run:
# Trial Run, Linear Model of Ideology with Geospatial Errors Using Metropolis-Hastings:
wv.mat <- metropolis.krige(y=logGas, X=wv.pred, east=WVwells$UTMESrf, north=WVwells$UTMNSrf,
  powered.exp=0.5, mcmc.samples=M, spatial.share=0.60, range.share=0.31, beta.var=1000,
  range.tol=0.1, b.tune=1, nugget.tune=1, psill.tune=30)

# Discard first 20% of Iterations as Burn-In (User Discretion Advised).
wv.mat <- wv.mat[(ceiling(0.2*M)+1):M,]

# Summarize Results
apply(wv.mat,2,quantile,c(0.5,0.05,0.95))

# Convergence Diagnostics
geweke(wv.mat)
heidel.welch(wv.mat)

# Draw Semivariogram
```

```

graph.terms<-apply(wv.mat[,1:3],2,quantile,0.5)
raw.semivar<-semivariogram(x=logGas,east=WVwells$UTMESrf,north=WVwells$UTMNSrf)
resid.semivar<-semivariogram(x=fracking.ols$residuals,east=WVwells$UTMESrf,
  north=WVwells$UTMNSrf,draw.plot=FALSE)
points(resid.semivar,pch=3,col='blue')
lines(exponential.semivariogram(nugget=graph.terms[1],decay=graph.terms[2],
  partial.sill=graph.terms[3],
  distance=as.numeric(names(resid.semivar)),power=0.5),col='red')

# Predictive Data for Two Wells Tapped in 2013
well.1<-c(log(1110)-mean.logElevation,1020-mean.RockPres)
well.2<-c(log(643)-mean.logElevation,630-mean.RockPres)
wells.2013<-rbind(well.1,well.2)
site.1<-c(557306.0, 4345265)
site.2<-c(434515.7, 4258449)
well.locations<-rbind(site.1,site.2)
colnames(well.locations)<-c("eastings","northings")

# Make predictions from median parameter values:
median.pred<-krige.pred(pred.x=wells.2013,
  pred.east=well.locations[, "eastings"],pred.north=well.locations[, "northings"],
  train.y=logGas,train.x=wv.pred,train.east=WVwells$UTMESrf,
  train.north=WVwells$UTMNSrf,mcmc.iter=wv.mat)
# Prediction of deviation on logged scale:
median.pred
# Prediction in thousands of cubic feet (MCF):
round(exp(median.pred+mean.logGas))

# Make predictions with 90% credible intervals:
cred.pred<-krige.pred(pred.x=wells.2013,
  pred.east=well.locations[, "eastings"],pred.north=well.locations[, "northings"],
  train.y=logGas,train.x=wv.pred,train.east=WVwells$UTMESrf,
  train.north=WVwells$UTMNSrf,mcmc.iter=wv.mat,credible=0.90)
# Prediction of deviation on logged scale:
cred.pred
# Prediction in thousands of cubic feet (MCF) and the true yield in 2013:
Actual.Yield<-c(471171, 7211)
round(cbind(exp(cred.pred+mean.logGas),Actual.Yield))

## End(Not run)

```

# Index

- \* **convergence**
  - geweke, [6](#)
  - heidel.welch, [7](#)
- \* **datasets**
  - congCombined, [2](#)
  - ContrivedData, [3](#)
  - lowerCombined, [15](#)
  - NY\_subset, [24](#)
  - NYcity\_subset, [20](#)
  - stateCombined, [28](#)
  - upperCombined, [29](#)
  - WVwells, [31](#)
- \* **diagnostic**
  - geweke, [6](#)
  - heidel.welch, [7](#)
- \* **estimation**
  - krige.update, [13](#)
  - metropolis.krige, [17](#)
- \* **krige**
  - krige.pred, [11](#)
  - semivariogram, [26](#)
- \* **posterior**
  - krige.posterior, [9](#)
  - krige.update, [13](#)
  - metropolis.krige, [17](#)
- \* **predict**
  - krige.pred, [11](#)
- \* **semivariogram**
  - exponential.semivariogram, [4](#)
  - semivariogram, [26](#)
- \* **update**
  - krige.update, [13](#)

congCombined, [2](#)  
ContrivedData, [3](#)  
  
exponential.semivariogram, [4](#)  
  
geweke, [6](#)  
  
heidel.welch, [7](#)  
  
krige.posterior, [9](#)  
krige.pred, [11](#)  
krige.update, [13](#)  
  
lowerCombined, [15](#)  
  
metropolis.krige, [17](#)  
  
NY\_subset, [23](#)  
NYcity\_subset, [20](#)  
  
semivariogram, [26](#)  
stateCombined, [28](#)  
  
upperCombined, [29](#)  
  
WVwells, [31](#)