matk Documentation

Release 0

Dylan R. Harp

November 12, 2013

Contents:

class matk.**matk**(***kwargs*)

   Class for Model Analysis ToolKit (MATK) module

   add_obs(*name*, ***kwargs*)

      Add observation to problem

      Parameters

      - **name** (*str*) – Name of observation

      - **kwargs** – keyword arguments passed to observation class

   add_par(*name*, ***kwargs*)

      Add parameter to problem

      Parameters

      - **name** (*str*) – Name of parameter

      - **kwargs** – keyword arguments passed to parameter class

   add_sampleset(*name*, *samples*, *responses=None*, *indices=None*, *index_start=1*)

      Add sample set to problem

      Parameters

      - **name** (*str*) – Name of sample set

      - **samples** (*list(fl64),ndarray(fl64)*) – Matrix of parameter samples with npar columns in order of [p.name for p in matkobj.parlist]

      - **responses** (*list(fl64),ndarray(fl64)*) – Matrix of associated responses with nobs columns in order of [o.name for o in matkobj.obslist] if observation exists (existence of observations is not required)

      - **indices** (*list(int),ndarray(int)*) – Sample indices to use when creating working directories and output files

   calibrate(*workdir=None*, *reuse_dirs=False*, *report_fit=True*)

      Calibrate MATK model

      Parameters

      - **workdir** (*str*) – Name of directory where model will be run. It will be created if it does not exist

      - **reuse_dirs** (*bool*) – If True and workdir exists, the model will reuse the directory

      - **report_fit** (*bool*) – If True, parameter statistics and correlations are printed to the screen

      Returns lmfit minimizer object

   forward(*pardict=None*, *workdir=None*, *reuse_dirs=False*)

      Run MATK model using current values

      Parameters

      - **pardict** (*dict*) – Dictionary of parameter values keyed by parameter names

      - **workdir** (*str*) – Name of directory where model will be run. It will be created if it does not exist

      - **reuse_dirs** (*bool*) – If True and workdir exists, the model will reuse the directory

      Returns int – 0: Successful run, 1: workdir exists

get_obs_names ()
    Get observation names

get_obs_values ()
    Get observation values

get_par_dist_pars ()
    Get parameters needed by parameter distributions

get_par_dists ()
    Get parameter probabilistic distributions

get_par_maxs ()
    Get parameter lower bounds

get_par_mins ()
    Get parameter lower bounds

get_par_names ()
    Get parameter names

get_par_nvals ()
    Get parameter nvals (number of values for parameter studies)

get_par_values ()
    Get parameter values

get_residuals ()
    Get least squares values

get_sims ()
    Get the current simulated values :returns: lst(fl64) – simulated values in order of matk.obslist

make_workdir (*workdir=None*, *reuse_dirs=False*)
    Create a working directory

> **Parameters**
>
> > • **workdir** (*str*) – Name of directory where model will be run. It will be created if it does not exist
> >
> > • **reuse_dirs** (*bool*) – If True and workdir exists, the model will reuse the directory
>
> **Returns** int – 0: Successful run, 1: workdir exists

model None
    Python function that runs model

model_args None
    Tuple of extra arguments to MATK model expected to come after parameter dictionary

model_kwargs None
    Dictionary of extra keyword arguments to MATK model expected to come after parameter dictionary and model_args

ncpus None
    Set number of cpus to use for concurrent model evaluations

parameters_file None
    Set the name of the parameters_file for parallel runs

results_file None
    Set the name of the results_file for parallel runs

run_samples(*name=None*, *ncpus=1*, *templatedir=None*, *workdir_base=None*, *save=True*, *reuse_dirs=False*)

Run model using values in samples for parameter values If samples are not specified, LHS samples are produced

> **Parameters**
>
> - **name** – Name of MATK sample set object
>
> - **ncpus** (*int*) – number of cpus to use to run models concurrently
>
> - **templatedir** (*str*) – Name of folder including files needed to run model (e.g. template files, instruction files, executables, etc.)
>
> - **workdir_base** (*str*) – Base name for model run folders, run index is appended to workdir_base
>
> - **save** (*bool*) – If True, model files and folders will not be deleted during parallel model execution
>
> - **reuse_dirs** (*bool*) – Will use existing directories if True, will return an error if False and directory exists
>
> **Returns** tuple(ndarray(fl64),ndarray(fl64)) - (Matrix of responses from sampled model runs siz rows by npar columns, Parameter samples, same as input samples if provided)

save_sampleset(*outfile*, *sampleset*)

Save sampleset to file

> **Parameters**
>
> - **outfile** (*str*) – Name of file where sampleset will be written
>
> - **sampleset** (*str*) – Sampleset name

seed None

Set the seed for random sampling

set_lhs_samples(*name*, *siz=None*, *noCorrRestr=False*, *corrmat=None*, *seed=None*, *index_start=1*)

Draw lhs samples of parameter values from scipy.stats module distribution

> **Parameters**
>
> - **name** (*str*) – Name of sample set to be created
>
> - **siz** (*int*) – Number of samples to generate, ignored if samples are provided
>
> - **noCorrRestr** (*bool*) – If True, correlation structure is not enforced on sample, use if siz is less than number of parameters
>
> - **corrmat** (*matrix*) – Correlation matrix
>
> - **seed** (*int*) – Random seed to allow replication of samples
>
> - **index_start** – Starting value for sample indices
>
> **Type** int
>
> **Returns** matrix – Parameter samples

set_obs_values(**args*, ***kwargs*)

Set simulated values using a dictionary or keyword arguments

set_par_values(**args*, ***kwargs*)

Set parameters using values in first argument

set_parstudy_samples(*name*, **args*, ***kwargs*)
  Generate parameter study samples

  Parameters

  - **name** (*str*) – Name of sample set to be created

  - **outfile** (*str*) – Name of file where samples will be written. If outfile=None, no file is written.

  - ***args** – Number of values for each parameter. The order is expected to match order of matk.parlist (e.g. [p.name for p in matk.parlist])

  - ****kwargs** – keyword arguments where keyword is the parameter name and argument is the number of desired values

  Returns ndarray(fl64) – Array of samples

templatedir None
  Set the name of the templatedir for parallel runs

workdir None
  Set the base name for parallel working directories

workdir_base None
  Set the base name for parallel working directories

workdir_index None
  Set the working directory index for parallel runs

class matk.**Parameter**(*name*, ***kwargs*)
  MATK parameter class

  dist None
    Probabilistic distribution of parameter belonging to scipy.stats module

  dist_pars None
    Distribution parameters required by self.dist (e.g. if dist == uniform, dist_pars = (min,max-min))

  expr None
    Mathematical expression to use to evaluate value

  max None
    Parameter upper bound

  mean None
    Parameter mean

  min None
    Parameter lower bound

  name None
    Parameter name

  nvals None
    Number of values the paramter will take for parameter studies

  offset None
    Offset to add to parameter

  scale None
    Scale factor to multiply parameter by

  std None
    Parameter st. dev.

> > **value None**
> > > Parameter value

> > **vary None**
> > > Boolean indicating whether or not to vary parameter

class matk.**Observation**(*name*, ***kwargs*)
> MATK observation class

> > **name None**
> > > Observation name

> > **residual None**
> > > Observation value minus simulated value

> > **sim None**
> > > Simulated value generated by MATK model

> > **value None**
> > > Observation value

> > **weight None**
> > > Weight to apply to simulated values

class matk.**SampleSet**(*name*, *samples*, *index_start=1*, ***kwargs*)
> MATK samples class - Stores information related to a sample includeing parameter samples, associated responses, and sample indices

> > **corr**(*type='pearson'*, *plot=False*)
> > > Calculate correlation coefficients of parameters and responses

> > > > **Parameters type** (*str*) – Type of correlation coefficient (pearson by default, spearman also avaialable)

> > > > **Returns** ndarray(fl64) – Correlation coefficients

> > **index_start None**
> > > Starting integer value for sample indices

> > **indices None**
> > > Array of sample indices

> > **name None**
> > > Sample set name

> > **obsnames None**
> > > Array of observation names

> > **parnames None**
> > > Array of parameter names

> > **responses None**
> > > Ndarray of sample set responses, rows are samples, columns are responses associated with observations in order of MATKobject.obslist

> > **samples None**
> > > Ndarray of parameter samples, rows are samples, columns are parameters in order of MATKobject.parlist

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*

m

T

V

W