

ExoSpec

Generated by Doxygen 1.8.13

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	Class Documentation	5
3.1	exospec.lc_class.DifferentFileSizes Class Reference	5
3.1.1	Detailed Description	5
3.2	exospec.lc_class.DifferentParamNum Class Reference	5
3.2.1	Detailed Description	6
3.3	exospec.read_input.EmptyFile Class Reference	6
3.3.1	Detailed Description	6
3.4	exospec.lc_class.EmptyFile Class Reference	6
3.4.1	Detailed Description	6
3.5	exospec.lc_class.EmptyFolder Class Reference	7
3.5.1	Detailed Description	7
3.6	exospec.lc_class.IncorrectNameFormat Class Reference	7
3.6.1	Detailed Description	7
3.7	exospec.lc_class.LightCurve Class Reference	7
3.8	exospec.lc_class.LightCurveData Class Reference	8
3.9	exospec.mcmc.MCMC Class Reference	9
3.9.1	Detailed Description	9
3.9.2	Constructor & Destructor Documentation	9

3.9.2.1	<code>__init__()</code>	10
3.9.3	Member Function Documentation	10
3.9.3.1	<code>get_mean_acceptance_fraction()</code>	10
3.9.3.2	<code>get_median_and_errors()</code>	11
3.9.3.3	<code>light_curve_plot()</code>	11
3.9.3.4	<code>run()</code>	12
3.9.3.5	<code>save_chain()</code>	12
3.9.3.6	<code>triangle_plot()</code>	13
3.9.3.7	<code>walker_plot()</code>	13
3.10	<code>exospec.read_input.NoInput</code> Class Reference	14
3.10.1	Detailed Description	14
3.11	<code>exospec.read_input.read_input</code> Class Reference	14
3.12	<code>exospec.TransitModel.TransitModel</code> Class Reference	15
3.12.1	Detailed Description	16
3.12.2	Constructor & Destructor Documentation	16
3.12.2.1	<code>__init__()</code>	16
3.12.3	Member Function Documentation	17
3.12.3.1	<code>kernelnc()</code>	17
3.12.3.2	<code>Inlike_gp()</code>	17
3.12.3.3	<code>Inprior_base()</code>	18
3.12.3.4	<code>Inprior_gp()</code>	18
3.12.3.5	<code>Inprob_gp()</code>	18
3.12.3.6	<code>Inprob_mcmc()</code>	19
3.12.3.7	<code>meannc()</code>	19
3.12.3.8	<code>model()</code>	19
3.12.3.9	<code>read_errors_data()</code>	20
3.12.3.10	<code>read_limb_dark_params()</code>	20
3.12.3.11	<code>sample_conditional()</code>	20
3.12.3.12	<code>set_values()</code>	21
3.12.3.13	<code>update_data()</code>	21
3.12.3.14	<code>update_kernel_params()</code>	22
3.12.3.15	<code>update_transit_params()</code>	22
3.12.3.16	<code>updateTransitMode()</code>	22

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Exception	
exospec.lc_class.DifferentFileSizes	5
exospec.lc_class.DifferentParamNum	5
exospec.lc_class.EmptyFile	6
exospec.lc_class.EmptyFolder	7
exospec.lc_class.IncorrectNameFormat	7
exospec.read_input.EmptyFile	6
exospec.read_input.NoInput	14
exospec.lc_class.LightCurve	7
exospec.lc_class.LightCurveData	8
object	
exospec.mcmc.MCMC	9
exospec.TransitModel.TransitModel	15
exospec.read_input.read_input	14

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

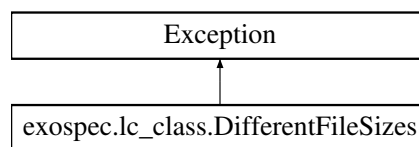
exospec.lc_class.DifferentFileSizes	5
exospec.lc_class.DifferentParamNum	5
exospec.read_input.EmptyFile	6
exospec.lc_class.EmptyFile	6
exospec.lc_class.EmptyFolder	7
exospec.lc_class.IncorrectNameFormat	7
exospec.lc_class.LightCurve	7
exospec.lc_class.LightCurveData	8
exospec.mcmc.MCMC	
Class to run MCMC to fit curve and produce basic diagnostic plots and statistics Uses emcee (to run MCMC) and corner (to produce triangle plots)	9
exospec.read_input.NoInput	14
exospec.read_input.read_input	14
exospec.TransitModel.TransitModel	
Class to estimate the Transit Model with the customized kernel	15

Chapter 3

Class Documentation

3.1 `exospec.lc_class.DifferentFileSizes` Class Reference

Inheritance diagram for `exospec.lc_class.DifferentFileSizes`:



3.1.1 Detailed Description

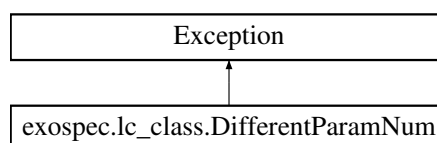
Raise when one the light curve file does not have the same size with light curve file for the lowest wavelength

The documentation for this class was generated from the following file:

- `/Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/lc_class.py`

3.2 `exospec.lc_class.DifferentParamNum` Class Reference

Inheritance diagram for `exospec.lc_class.DifferentParamNum`:



3.2.1 Detailed Description

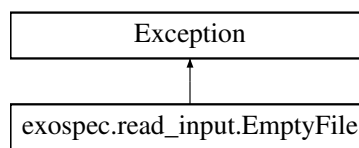
Raise when one the light curve file does not have the same number of parameters with light curve file for the

The documentation for this class was generated from the following file:

- /Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/lc_class.py

3.3 `exospec.read_input.EmptyFile` Class Reference

Inheritance diagram for `exospec.read_input.EmptyFile`:



3.3.1 Detailed Description

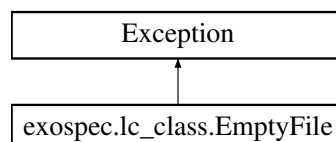
Raise when the input file is empty or only has comments

The documentation for this class was generated from the following file:

- /Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/read_input.py

3.4 `exospec.lc_class.EmptyFile` Class Reference

Inheritance diagram for `exospec.lc_class.EmptyFile`:



3.4.1 Detailed Description

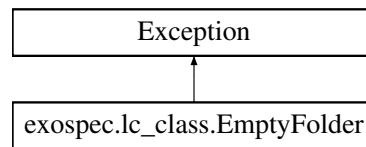
Raise when the light curve file is empty

The documentation for this class was generated from the following file:

- /Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/lc_class.py

3.5 exospec.lc_class.EmptyFolder Class Reference

Inheritance diagram for exospec.lc_class.EmptyFolder:



3.5.1 Detailed Description

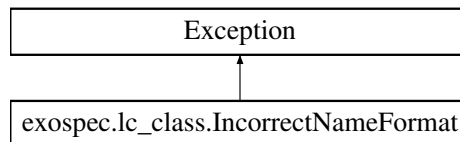
Raise when the light curve folder is empty

The documentation for this class was generated from the following file:

- `/Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/lc_class.py`

3.6 exospec.lc_class.IncorrectNameFormat Class Reference

Inheritance diagram for exospec.lc_class.IncorrectNameFormat:



3.6.1 Detailed Description

Raise when the light curve file name is not under the expected format `sample_lc_<wavelength>.txt`

The documentation for this class was generated from the following file:

- `/Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/lc_class.py`

3.7 exospec.lc_class.LightCurve Class Reference

Public Member Functions

- `def __init__(self, PathToLC, wave_bin_size)`
- `def LC_dic(self)`
- `def wave_length(self)`
- `def new_wave_length(self)`
- `def store_transit_model(self, transit_model)`

Public Attributes

- **files_list**
- **files_num**
- **wave_length**
- **new_wave_length**
- **LC_dic**
- **obj_mcmc**
- **obj_chain**
- **obj_mcmcGP**
- **obj_chainGP**
- **transit_model**

The documentation for this class was generated from the following file:

- /Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/lc_class.py

3.8 exospec.lc_class.LightCurveData Class Reference

Public Member Functions

- **def __init__** (self, Path_to_files)
- **def len_file** (self)
- **def time** (self)
- **def flux** (self)
- **def ferr** (self)
- **def param_num** (self)
- **def param_name** (self)
- **def param_list** (self)
- **def new_time_bin** (self, bin_size)
- **def plot_flux_time** (self, bin_size)
- **def plot_flux_param** (self, param_index)

Public Attributes

- **len_file**
- **time**
- **flux**
- **ferr**
- **param_num**
- **param_name**
- **param_list**

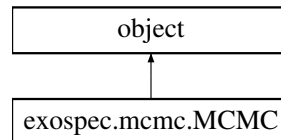
The documentation for this class was generated from the following file:

- /Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/lc_class.py

3.9 exospec.mcmc.MCMC Class Reference

Class to run [MCMC](#) to fit curve and produce basic diagnostic plots and statistics Uses [emcee](#) (to run [MCMC](#)) and [corner](#) (to produce triangle plots)

Inheritance diagram for exospec.mcmc.MCMC:



Public Member Functions

- `def __init__ (self, t, val, err, ln_prob_fn, transit_params, hyper_params, num_walkers, num_threads)`
The constructor.
- `def run (self, pos, burnin_steps, production_run_steps)`
Runs the [MCMC](#) should run [emcee](#) given a log probability function result is the [MCMC](#) chains which are saved as an object attribute.
- `def save_chain (self, filename)`
Saves the chain as a numpy array.
- `def get_mean_acceptance_fraction (self)`
Allows the user to access the mean acceptance fraction, which should be around 1/2.
- `def get_median_and_errors (self)`
Best fit parameters and 1 sigma errors.
- `def triangle_plot (self, extra_burnin_steps=0, theta_true=None, plot_transit_params=True, plot_hyper_↵
params=True, save_as_dir=".", save_as_name="triangle.png")`
Makes a triangle plot If an error is encountered the function returns 1 but does not raise an exception.
- `def walker_plot (self, extra_burnin_steps=0, theta_true=None, plot_transit_params=True, plot_hyper_↵
params=True, save_as_dir=".", save_as_name="walkers.png")`
Plots the chains of each walker and a histogram showing how each parameter was sampled If an error is encountered the function returns 1 but does not raise an exception.
- `def light_curve_plot (self, model, extra_burnin_steps=0, theta_true=None, plot_transit_params=True, plot_↵
hyper_params=True, save_as_dir=".", save_as_name="light_curve")`
Plots the chains of each walker and a histogram showing how each parameter was sampled If an error is encountered the function returns 1 but does not raise an exception.

3.9.1 Detailed Description

Class to run [MCMC](#) to fit curve and produce basic diagnostic plots and statistics Uses [emcee](#) (to run [MCMC](#)) and [corner](#) (to produce triangle plots)

3.9.2 Constructor & Destructor Documentation

3.9.2.1 `__init__()`

```
def exospec.mcmc.MCMC.__init__ (
    self,
    t,
    val,
    err,
    ln_prob_fn,
    transit_params,
    hyper_params,
    num_walkers,
    num_threads )
```

The constructor.

Parameters

<i>self</i>	The object pointer
<i>t</i>	A numpy array of the independent variable for the data to be fitted
<i>val</i>	A numpy array of the dependent variable for the data to be fitted
<i>err</i>	A numpy array of the errors on the dependent variable
<i>ln_prob_fn</i>	The log probability function to be sampled by the MCMC chain
<i>transit_params</i>	A list of strings giving the names of the curve's parameter's
<i>hyper_params</i>	A list of strings giving the names of noise parameters
<i>num_walkers</i>	Integer giving the number of walkers for the MCMC run
<i>num_threads</i>	An integer giving the number of threads to use on each core

3.9.3 Member Function Documentation

3.9.3.1 `get_mean_acceptance_fraction()`

```
def exospec.mcmc.MCMC.get_mean_acceptance_fraction (
    self )
```

Allows the user to access the mean acceptance fraction, which should be around 1/2.

Parameters

<i>self</i>	The object pointer
-------------	--------------------

Returns

Mean acceptance fraction

3.9.3.2 get_median_and_errors()

```
def exospec.mcmc.MCMC.get_median_and_errors (
    self )
```

Best fit parameters and 1 sigma errors.

Parameters

<i>self</i>	The object pointer
-------------	--------------------

Returns

Three numpy arrays giving the median and one sigma errors for each parameter

3.9.3.3 light_curve_plot()

```
def exospec.mcmc.MCMC.light_curve_plot (
    self,
    model,
    extra_burnin_steps = 0,
    theta_true = None,
    plot_transit_params = True,
    plot_hyper_params = True,
    save_as_dir = ".",
    save_as_name = "light_curve" )
```

Plots the chains of each walker and a histogram showing how each parameter was sampled. If an error is encountered the function returns 1 but does not raise an exception.

These plots are useful for visualization but should not cause the code to crash, as the main purpose is to create and save the [MCMC](#) chains

Parameters

<i>self</i>	The object pointer
<i>model</i>	A function that returns the lightcurve shape as a function of the light curve parameters and time
<i>extra_burnin_steps</i>	Number of steps (in addition to burnin_steps from run) at the start of each chain to neglect
<i>theta_true</i>	Numpy array of true parameter values if known (used for test data)
<i>plot_transit_params</i>	Boolean value specifying whether or not to plot the transit parameters
<i>plot_hyper_params</i>	Boolean value specifying whether or not to plot the hyper parameters
<i>save_as_dir</i>	Directory where plot should be saved. Default is current working Directory
<i>save_as_name</i>	Name under which plot should be saved

Return values

0	if successful
---	---------------

Return values

1	on failure
---	------------

3.9.3.4 run()

```
def exospec.mcmc.MCMC.run (
    self,
    pos,
    burnin_steps,
    production_run_steps )
```

Runs the [MCMC](#) should run emcee given a log probability function result is the [MCMC](#) chains which are saved as an object attribute.

Parameters

<i>self</i>	The object pointer
<i>pos</i>	A 2D numpy array giving the initial positions of the walkers in parameter space
<i>burnin_steps</i>	An integer giving the number of initial steps to take to start exploring the parameter space before starting to save the chains
<i>production_run_steps</i>	The number of steps to take for each walker after the burnin phase

Returns

A 2D numpy array with all the samples for each of the transit and hyper parameters

3.9.3.5 save_chain()

```
def exospec.mcmc.MCMC.save_chain (
    self,
    filename )
```

Saves the chain as a numpy array.

Parameters

<i>self</i>	The object pointer
<i>filename</i>	The filename including path where the chains should be saved

Return values

0	if successful
1	if an IO error occurs

3.9.3.6 triangle_plot()

```
def exospec.mcmc.MCMC.triangle_plot (
    self,
    extra_burnin_steps = 0,
    theta_true = None,
    plot_transit_params = True,
    plot_hyper_params = True,
    save_as_dir = ".",
    save_as_name = "triangle.png" )
```

Makes a triangle plot If an error is encountered the function returns 1 but does not raise an exception.

These plots are useful for visualization but should not cause the code to crash, as the main purpose is to create and save the [MCMC](#) chains

Parameters

<i>self</i>	The object pointer
<i>extra_burnin_steps</i>	Number of steps (in addition to burnin_steps from run) at the start of each chain to neglect
<i>theta_true</i>	Numpy array of true parameter values if known (used for test data)
<i>plot_transit_params</i>	Boolean value specifying whether or not to plot the transit parameters
<i>plot_hyper_params</i>	Boolean value specifying whether or not to plot the hyper parameters
<i>save_as_dir</i>	Directory where plot should be saved. Default is current working Directory
<i>save_as_name</i>	Name under which plot should be saved

Return values

0	if successful
1	on failure

3.9.3.7 walker_plot()

```
def exospec.mcmc.MCMC.walker_plot (
    self,
    extra_burnin_steps = 0,
    theta_true = None,
    plot_transit_params = True,
    plot_hyper_params = True,
    save_as_dir = ".",
    save_as_name = "walkers.png" )
```

Plots the chains of each walker and a histogram showing how each parameter was sampled If an error is encountered the function returns 1 but does not raise an exception.

These plots are useful for visualization but should not cause the code to crash, as the main purpose is to create and save the [MCMC](#) chains

Parameters

<i>self</i>	The object pointer
<i>extra_burnin_steps</i>	Number of steps (in addition to burnin_steps from run) at the start of each chain to neglect
<i>theta_true</i>	Numpy array of true parameter values if known (used for test data)
<i>plot_transit_params</i>	Boolean value specifying whether or not to plot the transit parameters
<i>plot_hyper_params</i>	Boolean value specifying whether or not to plot the hyper parameters
<i>save_as_dir</i>	Directory where plot should be saved. Default is current working Directory
<i>save_as_name</i>	Name under which plot should be saved

Return values

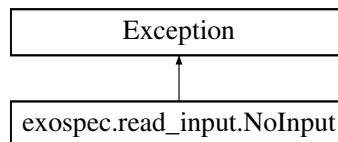
0	if successful
1	on failure

The documentation for this class was generated from the following file:

- /Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/mcmc.py

3.10 exospec.read_input.NoInput Class Reference

Inheritance diagram for exospec.read_input.NoInput:



3.10.1 Detailed Description

Raise when no input is found for a parameter

The documentation for this class was generated from the following file:

- /Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/read_input.py

3.11 exospec.read_input.read_input Class Reference

Public Member Functions

- def **__init__** (self, input_file)
- def **param_dic** (self)
- def **is_float** (self, string)

Public Attributes

- `param_dic`

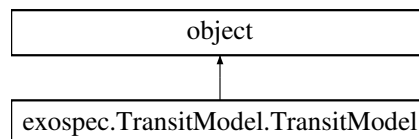
The documentation for this class was generated from the following file:

- `/Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/read_input.py`

3.12 exospec.TransitModel.TransitModel Class Reference

Class to estimate the Transit Model with the customized kernel.

Inheritance diagram for `exospec.TransitModel.TransitModel`:



Public Member Functions

- `def __init__ (self, kwargs)`
The constructor.
- `def set_values (self, dict_of_values, kwargs)`
Set the parameters of the model based on the values provided.
- `def read_limb_dark_params (self, kwargs)`
Forms a list of the limb darkening parameters from the user input In case some values are missed, the default ones will be taken.
- `def read_errors_data (self, kwargs)`
Collects the data about errors that was passed In case some values are missed, the default ones will be taken.
- `def update_data (self, time=None, obs=None, kwargs)`
Updates the data for the given parameters.
- `def update_transit_params (self, rp_new, u_new)`
Updates the parameters of the model.
- `def update_kernel_params (self, a_new=None, gamma_new=None, variance_new=None)`
Updates the hyperparameters of the kernel function.
- `def updateTransitMode (self)`
Updates the transit model parameters.
- `def model (self)`
Returns the flux values array.
- `def model (self, params)`
- `def meanfnc (self, t)`
Mean function for the kernel meaan estimation.
- `def kernelfnc (self, x1, x2, p=None)`
Computes the kernel function for the arbitrary sources of errors in the observations.
- `def lnlike_gp (self)`
Computes the log likelihood from gaussian process.
- `def lnprior_base (self)`

- *Checks if the batman parameters are within the predefined prior ranges.*
- `def lnprior_gp (self)`
Checks if the kernel parameters are within the predefined prior ranges.
- `def lnprob_gp (self)`
Computes the log probability of the parameters for the given data.
- `def sample_conditional (self, p, t, y, yerr)`
For a given set of parameters get predicted y values at t, and separate this into the transit signal component and the noise component.
- `def lnprob_mcmc (self, p, t, y, yerr)`
MCMC API for the Transit Model object.

Public Attributes

- **`batman_default_params`**
- **`transit_default_priors`**
- **`kernel_default_params`**
- **`kernel_default_priors`**
- **`data_defaults`**
- **`n_errors`**
- **`err_names`**
- **`errors_list`**
- **`params`**
- **`batman_model`**
- **`model`**
- **`data_dict`**
- **`model_initialized`**
- **`kernel_type`**
- **`Inprob`**

3.12.1 Detailed Description

Class to estimate the Transit Model with the customized kernel.

More details.

3.12.2 Constructor & Destructor Documentation

3.12.2.1 `__init__()`

```
def exospec.TransitModel.TransitModel.__init__ (
    self,
    kwargs )
```

The constructor.

Takes the disctionary of the parameters and data to customize the object. In case some values are missed, the default one are used.

Parameters

<i>self</i>	The object pointer
<i>**kwargs</i>	Accepts the dictionary of data, transit and kernel parameters

3.12.3 Member Function Documentation

3.12.3.1 kernelfnc()

```
def exospec.TransitModel.TransitModel.kernelfnc (
    self,
    x1,
    x2,
    p = None )
```

Computes the kernel function for the arbitrary sources of errors in the observations.

Parameters

<i>self</i>	The object pointer
<i>x1</i>	First time coordinate
<i>x2</i>	Second time coordinate
<i>p(=None)</i>	Kernel auxiliary parameters

Returns

Covariance between two points in time

3.12.3.2 lnlike_gp()

```
def exospec.TransitModel.TransitModel.lnlike_gp (
    self )
```

Computes the log likelihood from gaussian process.

Parameters

<i>self</i>	The object pointer
-------------	--------------------

Returns

Log likelihood of a set of observations under the Gaussian process model.

3.12.3.3 `lnprior_base()`

```
def exospec.TransitModel.TransitModel.lnprior_base (
    self )
```

Checks if the batman parameters are within the predefined prior ranges.

Parameters

<i>self</i>	The object pointer
-------------	--------------------

Returns

Returns 0 in case transit parameters within the prior range and -inf otherwise

3.12.3.4 `lnprior_gp()`

```
def exospec.TransitModel.TransitModel.lnprior_gp (
    self )
```

Checks if the kernel parameters are within the predefined prior ranges.

Parameters

<i>self</i>	The object pointer
-------------	--------------------

Returns

Returns -inf in case parameters out of the range and 0.0 if within the prior range

3.12.3.5 `lnprob_gp()`

```
def exospec.TransitModel.TransitModel.lnprob_gp (
    self )
```

Computes the log probability of the parameters for the given data.

Parameters

<i>self</i>	The object pointer
-------------	--------------------

Returns

Log probability of the parameters

3.12.3.6 lnprob_mcmc()

```
def exospec.TransitModel.TransitModel.lnprob_mcmc (
    self,
    p,
    t,
    y,
    yerr )
```

MCMC API for the Transit Model object.

Parameters

<i>self</i>	The object pointer
<i>p</i>	Parameters of the transit
<i>t</i>	Time data
<i>y</i>	Observations data
<i>yerr</i>	Errors data

Returns

Log probability of the chosen parameters

3.12.3.7 meanfnc()

```
def exospec.TransitModel.TransitModel.meanfnc (
    self,
    t )
```

Mean function for the kernel mean estimation.

Parameters

<i>self</i>	The object pointer
<i>t</i>	The time data

3.12.3.8 model()

```
def exospec.TransitModel.TransitModel.model (
    self )
```

Returns the flux values array.

Parameters

<i>self</i>	The object pointer
-------------	--------------------

Returns

Model-generated observation for the given transit parameters

3.12.3.9 read_errors_data()

```
def exospec.TransitModel.TransitModel.read_errors_data (
    self,
    kwargs )
```

Collects the data about errors that was passed In case some values are missed, the default ones will be taken.

Parameters

<i>self</i>	The object pointer
**kwargs	Dictionary of the parameters to pass

3.12.3.10 read_limb_dark_params()

```
def exospec.TransitModel.TransitModel.read_limb_dark_params (
    self,
    kwargs )
```

Forms a list of the limb darkening parameters from the user input In case some values are missed, the default ones will be taken.

Parameters

<i>self</i>	The object pointer
**kwargs	Dictionary of the parameters to pass

3.12.3.11 sample_conditional()

```
def exospec.TransitModel.TransitModel.sample_conditional (
    self,
    p,
    t,
```



```

        y,
        yerr )

```

For a given set of parameters get predicted y values at t, and separate this into the transit signal component and the noise component.

Parameters

<i>self</i>	The object pointer
<i>p</i>	Parameters of the transit
<i>t</i>	Time data
<i>y</i>	Observations data
<i>yerr</i>	Errors data

Returns

Predicted observations

3.12.3.12 set_values()

```

def exospec.TransitModel.TransitModel.set_values (
    self,
    dict_of_values,
    kwargs )

```

Set the parameters of the model based on the values provided.

In case some values are missed, the default ones will be taken.

Parameters

<i>self</i>	The object pointer
<i>dict_of_values</i>	Dictionary of the parameters to pass

3.12.3.13 update_data()

```

def exospec.TransitModel.TransitModel.update_data (
    self,
    time = None,
    obs = None,
    kwargs )

```

Updates the data for the given parameters.

Parameters

<i>self</i>	The object pointer
<i>time(=None)</i>	Time data
<i>obs(=None)</i>	Observations data
<i>**kwargs</i>	Handles arbitrary number of the errors that was passed

3.12.3.14 `update_kernel_params()`

```
def exospec.TransitModel.TransitModel.update_kernel_params (
    self,
    a_new = None,
    gamma_new = None,
    variance_new = None )
```

Updates the hyperparameters of the kernel function.

Parameters

<i>self</i>	The object pointer
<i>a_new</i>	New value of the kernel_a
<i>gamma_new</i>	New value of the kernel_gamma

3.12.3.15 `update_transit_params()`

```
def exospec.TransitModel.TransitModel.update_transit_params (
    self,
    rp_new,
    u_new )
```

Updates the parameters of the model.

Parameters

<i>self</i>	The object pointer
<i>rp_new</i>	New value of the rp parameter
<i>u_new</i>	New list of values for the limb darkening

3.12.3.16 `updateTransitMode()`

```
def exospec.TransitModel.TransitModel.updateTransitMode (
    self )
```

Updates the transit model parameters.

Parameters

<i>self</i>	The object pointer
-------------	--------------------

The documentation for this class was generated from the following file:

- /Users/heatherp/Documents/Courses/Computational/Project/ExoplanetSpectra/exospec/TransitModel.py

Index

- `__init__`
 - `exospec::TransitModel::TransitModel`, 16
 - `exospec::mcmc::MCMC`, 9
- `exospec.lc_class.DifferentFileSizes`, 5
- `exospec.lc_class.DifferentParamNum`, 5
- `exospec.lc_class.EmptyFile`, 6
- `exospec.lc_class.EmptyFolder`, 7
- `exospec.lc_class.IncorrectNameFormat`, 7
- `exospec.lc_class.LightCurve`, 7
- `exospec.lc_class.LightCurveData`, 8
- `exospec.mcmc.MCMC`, 9
- `exospec.read_input.EmptyFile`, 6
- `exospec.read_input.NoInput`, 14
- `exospec.read_input.read_input`, 14
- `exospec.TransitModel.TransitModel`, 15
- `exospec::TransitModel::TransitModel`
 - `__init__`, 16
 - `kernelnc`, 17
 - `Inlike_gp`, 17
 - `Inprior_base`, 17
 - `Inprior_gp`, 18
 - `Inprob_gp`, 18
 - `Inprob_mcmc`, 19
 - `meannc`, 19
 - `model`, 19
 - `read_errors_data`, 20
 - `read_limb_dark_params`, 20
 - `sample_conditional`, 20
 - `set_values`, 21
 - `update_data`, 21
 - `update_kernel_params`, 22
 - `update_transit_params`, 22
 - `updateTransitMode`, 22
- `exospec::mcmc::MCMC`
 - `__init__`, 9
 - `get_mean_acceptance_fraction`, 10
 - `get_median_and_errors`, 10
 - `light_curve_plot`, 11
 - `run`, 12
 - `save_chain`, 12
 - `triangle_plot`, 13
 - `walker_plot`, 13
- `get_mean_acceptance_fraction`
 - `exospec::mcmc::MCMC`, 10
- `get_median_and_errors`
 - `exospec::mcmc::MCMC`, 10
- `kernelnc`
 - `exospec::TransitModel::TransitModel`, 17
- `light_curve_plot`
 - `exospec::mcmc::MCMC`, 11
- `Inlike_gp`
 - `exospec::TransitModel::TransitModel`, 17
- `Inprior_base`
 - `exospec::TransitModel::TransitModel`, 17
- `Inprior_gp`
 - `exospec::TransitModel::TransitModel`, 18
- `Inprob_gp`
 - `exospec::TransitModel::TransitModel`, 18
- `Inprob_mcmc`
 - `exospec::TransitModel::TransitModel`, 19
- `meannc`
 - `exospec::TransitModel::TransitModel`, 19
- `model`
 - `exospec::TransitModel::TransitModel`, 19
- `read_errors_data`
 - `exospec::TransitModel::TransitModel`, 20
- `read_limb_dark_params`
 - `exospec::TransitModel::TransitModel`, 20
- `run`
 - `exospec::mcmc::MCMC`, 12
- `sample_conditional`
 - `exospec::TransitModel::TransitModel`, 20
- `save_chain`
 - `exospec::mcmc::MCMC`, 12
- `set_values`
 - `exospec::TransitModel::TransitModel`, 21
- `triangle_plot`
 - `exospec::mcmc::MCMC`, 13
- `update_data`
 - `exospec::TransitModel::TransitModel`, 21
- `update_kernel_params`
 - `exospec::TransitModel::TransitModel`, 22
- `update_transit_params`
 - `exospec::TransitModel::TransitModel`, 22
- `updateTransitMode`
 - `exospec::TransitModel::TransitModel`, 22
- `walker_plot`
 - `exospec::mcmc::MCMC`, 13