
CathPy Documentation

Release 0.3.10

Ian Sillitoe

Mar 07, 2022

Contents

1	Getting Started	3
2	Guide	5
2.1	Tools	5
2.2	API	6
2.3	Need Help	10
3	Code	11
4	Contributors	13
5	Indices and tables	15
	Python Module Index	17
	Index	19

CathPy is a Bioinformatics library and toolkit written for [Python3](#). It is written and maintained by the maintainers of [CATH](#), the Protein Structure Classification Database at [UCL](#), UK.

The code was mainly written for use within the CATH group and covers a range of tasks from manipulating protein 3D structures and sequence alignments. This software has been released to the public domain since some of the functionality may also be useful to groups outside of the CATH group.

CHAPTER 1

Getting Started

This software is released regularly on [PyPI](#) with [semantic versioning](#). The latest stable release can be installed in your local Python environment with:

```
pip install cathpy
```

You can update an existing installation with the latest version by adding the `--update` flag.

```
pip install --update cathpy
```

Note: in very old versions of `pip` you may find it easier to uninstall (`pip uninstall cathpy`) then reinstall (`pip install cathpy`).

2.1 Tools

2.1.1 cath-align-scorecons

```
usage: cath-align-scorecons [-h] --in IN_FILE [--out OUT_FILE] --format
                             {sto,fasta} [--replace] [--verbose]
```

Update the scorecons data **for** a STOCKHOLM alignment

optional arguments:

```
-h, --help            show this help message and exit
--in IN_FILE, -i IN_FILE
                        input alignment file (default: None)
--out OUT_FILE, -o OUT_FILE
                        output STOCKHOLM alignment file (default: None)
--format {sto,fasta}, -f {sto,fasta}
                        alignment format of input file (default: sto)
--replace             overwrite the input alignment file with updated
                        scorecons (default: False)
--verbose, -v         more verbose logging (default: 0)
```

2.1.2 cath-align-summary

```
usage: cath-align-summary [-h] (-d ALN_DIR | -i ALN_FILE) [--suffix SUFFIX]
                           [--skipempty] [--verbose]
```

Provide summary stats **for** alignments

optional arguments:

```
-h, --help            show this help message and exit
```

(continues on next page)

(continued from previous page)

```
-d ALN_DIR, --aln_dir ALN_DIR
                        input alignment directory
-i ALN_FILE, --aln_file ALN_FILE
                        input alignment file
--suffix SUFFIX        suffix to use when matching alignment files
--skipempty            skip empty files
--verbose, -v          more verbose logging
```

2.1.3 cath-sc-merge-alignment

```
usage: cath-sc-merge-alignment [-h] --cv CATH_VERSION --in SC_FILE --out
                                OUT_STO [--ff_dir FF_DIR] [--ff_tmpl FF_TMPL]
                                [--verbose]
```

Merge FunFams using a structure-based alignment

optional arguments:

```
-h, --help            show this help message and exit
--cv CATH_VERSION      cath version
--in SC_FILE           input reference structure-based alignment (FASTA)
--out OUT_STO          output merged alignment file (STOCKHOLM)
--ff_dir FF_DIR        directory in which funfam alignments can be found
                        (default: <sc_dir>)
--ff_tmpl FF_TMPL      template used to help locate funfam file (default:
                        __SFAM__-ff-__FF_NUM__.reduced.sto)
--verbose, -v          more verbose logging
```

2.2 API

2.2.1 cathpy.core.align

```
from cathpy.core.align import Align

aln = Align.from_stockholm('/path/to/align.sto')

aln.count_sequences
# 75

seq = aln.find_seq_by_accession('lcukA01')
seq = aln.find_seq_by_id('lcukA01/1-151')
```

2.2.2 cathpy.core.datafiles

```
from cathpy.core import datafiles

release = datafiles.ReleaseDir('v4.2')

release.get_file('chaingcf', 'lcukA01')
# /cath/data/v4_2_0/chaingcf/lcukA.gcf
```

Provides access to the location of CATH data files

```
class cathpy.core.datafiles.AtomFastaFileType
    Represents a FASTA file type based on ATOM records (registered as 'atomfasta').

class cathpy.core.datafiles.CombsFastaFileType
    Represents a FASTA file type based on COMBS mapping (registered as 'combsfasta').

class cathpy.core.datafiles.FileTypes
    Provides convenient access to all the different file types

    cls
        alias of CombsFastaFileType

    classmethod get (file_type)
        Returns the FileType class corresponding to the provided key.

        Parameters file_type – index used to register the class
```

Example

```
FileTypes.get('combsfasta') # CombsFastaFileType

class cathpy.core.datafiles.GcfFileType
    Represents a GCF file type (registered as 'chaingcf').

class cathpy.core.datafiles.GenericFileType
    Represents a type of CATH Data file.

class cathpy.core.datafiles.ReleaseDir (cath_version, *, base_dir='/cath/data')
    Provides access to files relating to an official release of CATH.

    Parameters

        • cath_version – version of CATH (eg 'v4_2_0')

        • base_dir – root directory for all data files (default: '/cath/data')

    get_file (file_type, entity_id)
        Returns the path for the given file type and identifier.

        Parameters

        • file_type – type of file (eg 'chaingcf')

        • entity_id – identifier of the CATH entity (eg '1cukA')
```

2.2.3 cathpy.core.error

```
from cathpy.core import error as err

raise err.OutOfBoundsError('error message')
```

CATH exception classes

```
exception cathpy.core.error.DuplicateSequenceError
    More than one sequence in an alignment has the same id

exception cathpy.core.error.FileEmptyError
    File is empty.
```

exception `cathpy.core.error.GapError`

Exception raised when trying to find residue information about a gap position.

exception `cathpy.core.error.GeneralError`

General Exception class within the cathpy package.

exception `cathpy.core.error.HttpError`

Problem getting/sending data over HTTP

exception `cathpy.core.error.InvalidInputError`

Exception raised when an error is encountered due to incorrect input.

exception `cathpy.core.error.JsonError`

Problem parsing JSON

exception `cathpy.core.error.MergeCheckError`

Exception raised when an error is encountered when checking the merge.

exception `cathpy.core.error.MergeCorrespondenceError` (*, *seq_id*, *aln_type*, *seq_type*,
ref_no_gaps, *corr_no_gaps*)

Exception raised when failing to match correspondence sequences during alignment merge.

exception `cathpy.core.error.MissingExecutableError`

Missing an external executable.

exception `cathpy.core.error.MissingGroupsimError`

Failed to find groupsim executable

exception `cathpy.core.error.MissingLockError`

Missing a lock object.

exception `cathpy.core.error.MissingScoreconsError`

Failed to find scorecons executable

exception `cathpy.core.error.MissingSegmentsError`

Exception raised when segment information is missing.

exception `cathpy.core.error.NoMatchesError`

No matches.

exception `cathpy.core.error.OutOfBoundsError`

Exception raised when code has moved outside expected boundaries.

exception `cathpy.core.error.ParamError`

Incorrect parameters.

exception `cathpy.core.error.ParseError`

Failed to parse information.

exception `cathpy.core.error.SeqIOError`

General Exception class within the SeqIO module

exception `cathpy.core.error.TooManyMatchesError`

Found more matches than expected.

2.2.4 cathpy.core.funfhmmer

```
from cathpy.core import funfhmmer

api = funfhmmer.Client()
```

(continues on next page)

(continued from previous page)

```
res = api.search_fasta(fasta_file='/path/to/seq.fa')

res.funfam_scan.as_tsv()

res.funfam_resolved_scan.as_tsv()
```

2.2.5 cathpy.core.models

Provides access to classes that representing general entities such as amino acids, db identifiers, etc.

```
from cathpy.core.models import (
    AminoAcid,
    AminoAcids,
    CathID,
    ClusterFile,
    FunfamID, )

aa = AminoAcids.get_by_id('A')

aa.one           # 'A'
aa.three         # 'ala'
aa.word          # 'alanine'

AminoAcids.is_valid_aa('Z') # False

cathid = CathID("1.10.8.10.1")

cathid.sfam_id   # '1.10.8.10'
cathid.depth     # 5
cathid.cath_id_to_depth(3) # '1.10.8'

funfam_file = ClusterFile("/path/to/1.10.8.10-ff-1234.reduced.sto")

funfam_file.path           # '/path/to/'
funfam_file.sfam_id        # '1.10.8.10'
funfam_file.cluster_type   # 'ff'
funfam_file.cluster_num    # 1234
funfam_file.desc           # '.reduced'
funfam_file.suffix         # '.sto'
```

2.2.6 cathpy.core.release

```
from cathpy.core.release import (
    CathDomainList,
    CathNamesList,
    CathDomall, )

domainlist = CathDomainList.from_file('/path/to/CathDomainList.list')

domainlist[5].domain_id = '3b89A01'
domainlist[5].cath_id = '1.10.8.10.2.1.1.2.1'
```

(continues on next page)

(continued from previous page)

```
sreps = domainlist.filter_reps(depth=5)
sreps.to_file('/path/to/CathDomainList.S35.list')
```

2.2.7 cathpy.core.util

```
from cathpy.core import util
```

2.2.8 cathpy.core.version

```
from cathpy.core.version import CathVersion

cv = CathVersion("v4.2") # or "v4_2_0", "current"

cv.dirname
# "4_2_0"

cv.pg_dbname
# "cathdb_v4_2_0"

cv.is_current
# False
```

Provides access to common functions related to CATH versions

class `cathpy.core.version.CathVersion(*args, **kwargs)`
Object that represents a CATH version.

dirname
Return the version represented as a directory name (eg 'v4_2_0').

classmethod from_string(version_str)
Create a new CathVersion object from a string.

is_current
Returns whether the version corresponds to 'current' (eg HEAD)

join(join_char='.')
Returns the version string (with an optional join_char).

pg_dbname
Return the version represented as a postgresql database (eg 'cathdb_v4_2_0').

to_string()
Returns the CATH version in string form.

2.3 Need Help

Problems? Please contact i.sillitoe@ucl.ac.uk

CHAPTER 3

Code

The code is maintained on GitHub:

```
git clone https://github.com/UCL/cathpy
```

Please submit any feature requests and bug reports as [GitHub Issues](#) (accompanying Pull Requests are welcome!).

CHAPTER 4

Contributors

- Ian Sillitoe @sillitoe
- Sayoni Das @sayonidas03

If you want to see your name here, then submit a pull request to the code base. . .

CHAPTER 5

Indices and tables

- `genindex`
- `modindex`
- `search`

C

`cathpy.core.datafiles`, [7](#)

`cathpy.core.error`, [7](#)

`cathpy.core.version`, [10](#)

A

AtomFastaFileType (class in *cathpy.core.datafiles*), 7

C

cathpy.core.datafiles (module), 7
cathpy.core.error (module), 7
cathpy.core.version (module), 10
CathVersion (class in *cathpy.core.version*), 10
cls (*cathpy.core.datafiles.FileTypes* attribute), 7
CombsFastaFileType (class in *cathpy.core.datafiles*), 7

D

dirname (*cathpy.core.version.CathVersion* attribute), 10
DuplicateSequenceError, 7

F

FileEmptyError, 7
FileTypes (class in *cathpy.core.datafiles*), 7
from_string() (*cathpy.core.version.CathVersion* class method), 10

G

GapError, 7
GcfFileType (class in *cathpy.core.datafiles*), 7
GeneralError, 8
GenericFileType (class in *cathpy.core.datafiles*), 7
get() (*cathpy.core.datafiles.FileTypes* class method), 7
get_file() (*cathpy.core.datafiles.ReleaseDir* method), 7

H

HttpError, 8

I

InvalidInputError, 8

is_current (*cathpy.core.version.CathVersion* attribute), 10

J

join() (*cathpy.core.version.CathVersion* method), 10
JsonError, 8

M

MergeCheckError, 8
MergeCorrespondenceError, 8
MissingExecutableError, 8
MissingGroupsimError, 8
MissingLockError, 8
MissingScoreconsError, 8
MissingSegmentsError, 8

N

NoMatchesError, 8

O

OutOfBoundsError, 8

P

ParamError, 8
ParseError, 8
pg_dbname (*cathpy.core.version.CathVersion* attribute), 10

R

ReleaseDir (class in *cathpy.core.datafiles*), 7

S

SeqIOError, 8

T

to_string() (*cathpy.core.version.CathVersion* method), 10
TooManyMatchesError, 8