

API Documentation

API Documentation

November 30, 2016

Contents

Contents	1
1 Package hal	5
1.1 Modules	5
1.2 Variables	6
2 Package hal.files	7
2.1 Modules	7
2.2 Variables	7
3 Module hal.files.models	8
3.1 Variables	8
3.2 Class FileSystem	8
3.2.1 Methods	8
3.2.2 Properties	11
3.3 Class Document	11
3.3.1 Methods	11
3.3.2 Properties	13
3.4 Class Directory	14
3.4.1 Methods	14
3.4.2 Properties	15
3.5 Class MP3Song	15
3.5.1 Methods	15
3.5.2 Properties	16
4 Package hal.internet	17
4.1 Modules	17
4.2 Variables	17
5 Module hal.internet.engines	18
5.1 Class SearchEngineResult	18
5.1.1 Methods	18
5.1.2 Properties	18
5.2 Class SearchEngine	18
5.2.1 Methods	19
5.2.2 Properties	19
6 Module hal.internet.parser	20

6.1	Functions	20
6.2	Variables	20
6.3	Class HtmlTable	20
6.3.1	Methods	21
6.3.2	Properties	21
7	Module hal.internet.selenium	22
7.1	Variables	22
7.2	Class SeleniumForm	22
7.2.1	Methods	22
8	Module hal.internet.web	24
8.1	Functions	24
8.2	Variables	24
8.3	Class Webpage	24
8.3.1	Methods	25
8.3.2	Properties	26
9	Module hal.internet.youtube	27
9.1	Functions	27
10	Package hal.maths	28
10.1	Modules	28
10.2	Variables	28
11	Module hal.maths.crypt	29
11.1	Class MD5	29
11.1.1	Methods	29
11.1.2	Properties	29
11.2	Class MD6	29
11.2.1	Methods	30
11.2.2	Properties	30
11.2.3	Class Variables	30
11.3	Class SHA	30
11.3.1	Methods	31
11.3.2	Properties	31
11.3.3	Class Variables	32
11.4	Class DES	32
11.4.1	Methods	32
11.4.2	Properties	32
11.4.3	Class Variables	33
11.5	Class ARC	33
11.5.1	Methods	33
11.5.2	Properties	33
11.5.3	Class Variables	34
11.6	Class AES	34
11.6.1	Methods	34
11.6.2	Properties	34
11.7	Class HMAC	34
11.7.1	Methods	35
11.7.2	Properties	35
11.8	Class BLOWFISH	35

11.8.1	Methods	35
11.8.2	Properties	36
11.9	Class IDEA	36
11.9.1	Methods	36
11.9.2	Properties	37
11.10	Class CAST128	37
11.10.1	Methods	37
11.10.2	Properties	37
11.11	Class Dsa	38
11.11.1	Methods	38
11.11.2	Properties	38
12	Module hal.maths.maths	39
12.1	Functions	39
12.2	Variables	39
12.3	Class Integer	39
12.3.1	Methods	39
12.3.2	Properties	40
12.3.3	Class Variables	40
12.4	Class EightQueen	40
12.4.1	Methods	40
12.4.2	Properties	41
13	Module hal.maths.plotter	42
13.1	Class Plot2d	42
13.1.1	Methods	42
13.1.2	Properties	42
13.2	Class Plot3d	43
13.2.1	Methods	43
13.2.2	Properties	43
13.3	Class Plot4d	44
13.3.1	Methods	44
13.3.2	Properties	44
14	Package hal.ml	46
14.1	Modules	46
14.2	Variables	46
15	Package hal.ml.data	47
15.1	Modules	47
15.2	Variables	47
16	Module hal.ml.data.parser	48
16.1	Variables	48
16.2	Class Parser	48
16.2.1	Methods	48
16.2.2	Properties	48
16.3	Class CSVParser	49
16.3.1	Methods	49
16.3.2	Properties	49
17	Module hal.ml.features	50

17.1 Functions	50
18 Package hal.ml.models	51
18.1 Modules	51
18.2 Variables	51
19 Module hal.ml.models.classification	52
19.1 Functions	52
20 Module hal.ml.models.pipelined	53
20.1 Functions	53
21 Module hal.ml.models.regression	54
21.1 Functions	54
22 Module hal.ml.models.time_series	55
22.1 Functions	55
23 Module hal.ml.predict	56
23.1 Class BasePrediction	56
23.1.1 Methods	56
23.1.2 Properties	56
24 Module hal.ml.utils	57
24.1 Functions	57
25 Package hal.profile	59
25.1 Modules	59
25.2 Variables	59
26 Module hal.profile.performance	60
26.1 Variables	60
26.2 Class EightQueenTest	60
26.2.1 Methods	60
26.2.2 Properties	60
27 Package hal.wrappers	62
27.1 Modules	62
27.2 Variables	62
28 Module hal.wrappers.methods	63
28.1 Functions	63
Index	64

1 Package hal

1.1 Modules

- **files** (*Section 2, p. 7*)
 - **models**: Main entities in files, such as documents, folders.
(*Section 3, p. 8*)
- **internet** (*Section 4, p. 17*)
 - **engines**: Abstract search engines.
(*Section 5, p. 18*)
 - **parser**: Parse anything there is on the Internet.
(*Section 6, p. 20*)
 - **selenium**: Some utils methods for a selenium webdriver
(*Section 7, p. 22*)
 - **web**: Deal with webpages.
(*Section 8, p. 24*)
 - **youtube**: Get rss feed for youtube channel.
(*Section 9, p. 27*)
- **maths**: MATHS: important and scalable math functions
(*Section 10, p. 28*)
 - **crypt**: Perform fast hash, encryption and calculations related to cryptography.
(*Section 11, p. 29*)
 - **maths**: A few elegant and powerful mathematical functions.
(*Section 12, p. 39*)
 - **plotter**: Show elegant plots in any dimension.
(*Section 13, p. 42*)
- **ml** (*Section 14, p. 46*)
 - **data** (*Section 15, p. 47*)
 - * **parser**: Parsers for raw databases.
(*Section 16, p. 48*)
 - **features**: Collection of methods to find weights of features and select the best ones.
(*Section 17, p. 50*)
 - **models** (*Section 18, p. 51*)
 - * **classification**: Prediction methods based on classification algorithms.
(*Section 19, p. 52*)
 - * **pipelined**: Prediction methods based on multiple models mixed up.
(*Section 20, p. 53*)
 - * **regression**: Prediction methods based on regression algorithms.
(*Section 21, p. 54*)
 - * **time_series**: Multi-purpose prediction methods to be used in time-series.
(*Section 22, p. 55*)
 - **predict**: " General model to make prediction about everything.
(*Section 23, p. 56*)
 - **utils**: Various tools and utilities to deal with database and machine learning.
(*Section 24, p. 57*)
- **profile** (*Section 25, p. 59*)
 - **performance**: Perform benchmarks and tests on your PC.
(*Section 26, p. 60*)
- **wrappers** (*Section 27, p. 62*)
 - **methods**: Typical (and useful) function wrappers
(*Section 28, p. 63*)

1.2 Variables

Name	Description
__package__	Value: None

2 Package *hal.files*

2.1 Modules

- **models**: Main entities in files, such as documents, folders.
(Section 3, p. 8)

2.2 Variables

Name	Description
<code>__package__</code>	Value: None

3 Module hal.files.models

Main entities in files, such as documents, folders.

3.1 Variables

Name	Description
BAD_CHARS	Value: ['.', ':', '"', '\xe2\x80\x99', '&', '720p', '1080p', 'yi...']
RUSSIAN_CHARS	Value: ['\xd1\x88', '\xd0\xb0', '\xd0\xb1', '\xd0\xbb', '\xd0\xba...']
VIDEO_FORMAT	Value: ['.', '.3g2', '.3gp', '.amv', '.asf', '.avi', '.drc', '.f...']
ARCHIVE_FORMAT	Value: ['.7z', '._?_', '._?Q?', '._?Z?', '.a', '.ace', '.afa', '.a...']
SUBTITLE_FORMAT	Value: ['.srt', '.sub', '.sbv']
TEXT_FORMAT	Value: ['.cnf', '.conf', '.cfg', '.chm', '.epub', '.log', '.asc'...]
IMAGE_FORMAT	Value: ['.ani', '.bmp', '.cal', '.fax', '.gif', '.img', '.jbg', ...]
AUDIO_FORMAT	Value: ['.3gp', '.aa', '.aac', '.aax', '.act', '.aiff', '.amr', ...]
PATH_SEPARATOR	Value: '/'
__package__	Value: 'hal.files'

3.2 Class FileSystem



Known Subclasses: hal.files.models.Directory, hal.files.models.Document, hal.files.models.MP3Song

3.2.1 Methods

__init__ (<i>self</i> , <i>path</i>)
<pre> :param path: string Path to file Overrides: object.__init__ </pre>

fix_raw_path(*path*)

```
:param path: string
    Path to fix
:return: string
    Right path
```

remove_year(*name*)

```
:param name: string
    Name to edit
:return: string
    Given string bu with no years.
```

remove_brackets(*name*)

```
:param name: string
    Name to edit
:return: string
    Given string bu with no barckets.
```

extract_name_max_chars(*name*, *max_chars*=64, *blank*=' ')

```
:param name: string
    Name to edit
:param max_chars: int
    Maximum chars of new name
:param blank: string
    Char that represents the blank between words.
:return: string
    Name edited to contain at most max_chars (truncate to nearest word)
```

prettify(*name*, *bad_chars*=['.', ':', '"', '\xe2\x80\x99', '&', '720p', '1080p', 'yi...', *r*=' ')

```
:param name: string
    Name to edit
:param bad_chars: []
    List of bad strings to remove
:param r: string
    Default blanks in name.
:return: string
    Prettier name from given one: replace bad chars with good ones.
```

ls_dir(*path*, *include_hidden*=False)

:param path: string
 Path to directory to get list of files and folders
:param include_hidden: bool
 Whether to include hidden files in list.
:return: list
 List of paths in given directory.

ls_recurse(*path*, *include_hidden*=False)

:param path: string
 Path to directory to get list of files and folders
:param include_hidden: bool
 Whether to include hidden files in list.
:return: list
 List of paths in given directory recursively.

ls(*path*, *recurse*, *include_hidden*=False)

:param path: string
 Path to directory to get list of files and folders
:param recurse: bool
 Whether to recurse into subdirectories or not.
:param include_hidden: bool
 Whether to include hidden files in list.
:return: list
 List of paths in given directory recursively.

is_archive_mac(*self*)

:return: True iff document is an MACOSX archive.

is_russian(*self*)

:return: True iff document has a russian name.

trash(*self*)

:return: void
 Trash given file/folder

```
rename(self, new_path)
```

```
:param new_path: string
    New path to use
:return: void
    Rename to new path
```

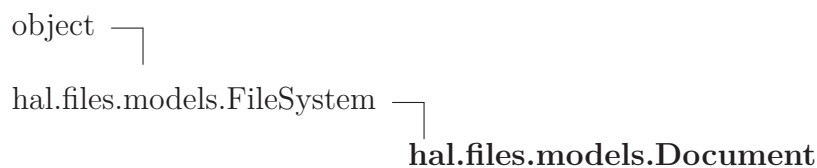
Inherited from object

```
__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),
__reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),
__str__(), __subclasshook__()
```

3.2.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

3.3 Class Document



3.3.1 Methods

```
__init__(self, path)
```

```
:param path: string
    Path to file
Overrides: object.__init__
```

move_file_to_directory(*file_path*, *directory_path*)

```
:param file_path: string
    Path to file to move
:param directory_path: string
    Path to target directory where to move file
:return: void
    Move file to given directory
```

move_file_to_file(*old_path*, *new_path*)

```
:param old_path: string
    Old path of file to move
:param new_path: string
    New path (location) of file
:return: void
    Move file from old location to new one
```

write_data_to_file(*data*, *out_file*)

```
:param data: string
    Data to write to file.
:param out_file: string
    Path to output file.
:return: void
    Writes given data to given path file.
```

extract_name_extension(*file_name*)

```
:param file_name: string
    Name of file
:return: tuple string, string
    Name of file, extension of file
```

```
get_path_name(self)
```

```
:return: tuple string, string  
        Name of path, name of file (or folder)
```

```
is_video(self)
```

```
:return: True iff document is a video.
```

```
is_subtitle(self)
```

```
:return: True iff document is a subtitle.
```

```
is_text(self)
```

```
:return: True iff document is a text file.
```

```
is_image(self)
```

```
:return: True iff document is an image.
```

```
is_audio(self)
```

```
:return: True iff document is an audio.
```

```
is_hidden(self)
```

```
:return: bool  
        True iff path is hidden
```

*Inherited from **hal.files.models.FileSystem**(Section 3.2)*

```
extract_name_max_chars(), fix_raw_path(), is_archive_mac(), is_russian(), ls(),  
ls_dir(), ls_recurse(), prettify(), remove_brackets(), remove_year(), rename(),  
trash()
```

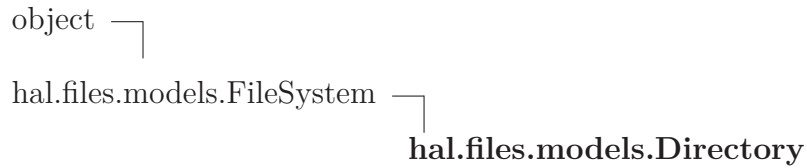
*Inherited from **object***

```
__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),  
__reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),  
__str__(), __subclasshook__()
```

3.3.2 Properties

Name	Description
<i>Inherited from object</i> __class__	

3.4 Class Directory



3.4.1 Methods

__init__ (<i>self</i> , <i>path</i>) <hr/> :param path: string Path to file Overrides: object.__init__
create_new (<i>path</i>) <hr/> :param path: string Path to directory to create :return: void Creates new directory
get_path_name (<i>self</i>) <hr/> :return: tuple string, string Name of path, name of file (or folder)

```
is_empty(self)
```

```
:return: Bool  
        True iff empty
```

Inherited from `hal.files.models.FileSystem` (Section 3.2)

```
extract_name_max_chars(), fix_raw_path(), is_archive_mac(), is_russian(), ls(),
ls_dir(), ls_recurse(), prettify(), remove_brackets(), remove_year(), rename(),
trash()
```

Inherited from object

```

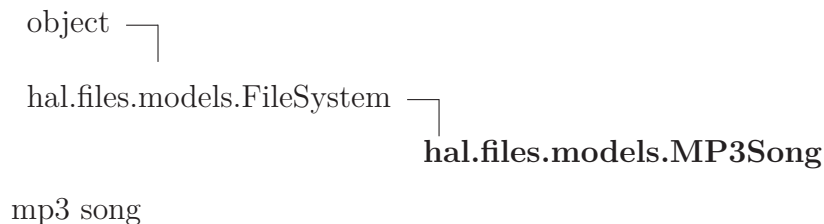
__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),
__reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),
__str__(), __subclasshook__()

```

3.4.2 Properties

Name	Description
<i>Inherited from object</i> __class__	

3.5 Class MP3Song



3.5.1 Methods

```

__init__(self, path)

:param path: string
    Path to file

Overrides: object.__init__ exit(herited documentation)

```

<code>set__name(self, name)</code>

<code>set__artist(self, artist)</code>

<code>set__album(self, album)</code>

<code>set__nr__track(self, nr__track)</code>

<code>set__year(self, year)</code>

Inherited from hal.files.models.FileSystem(Section 3.2)

`extract_name_max_chars()`, `fix_raw_path()`, `is_archive_mac()`, `is_russian()`, `ls()`,
`ls_dir()`, `ls_recurse()`, `prettify()`, `remove_brackets()`, `remove_year()`, `rename()`,
`trash()`

Inherited from object

`__delattr__()`, `__format__()`, `__getattr__()`, `__hash__()`, `__new__()`,
`__reduce__()`, `__reduce_ex__()`, `__repr__()`, `__setattr__()`, `__sizeof__()`,
`__str__()`, `__subclasshook__()`

3.5.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

4 Package *hal.internet*

4.1 Modules

- **engines:** Abstract search engines.
(Section 5, p. 18)
- **parser:** Parse anything there is on the Internet.
(Section 6, p. 20)
- **selenium:** Some utils methods for a selenium webdriver
(Section 7, p. 22)
- **web:** Deal with webpages.
(Section 8, p. 24)
- **youtube:** Get rss feed for youtube channel.
(Section 9, p. 27)

4.2 Variables

Name	Description
<code>__package__</code>	Value: None

5 Module `hal.internet.engines`

Abstract search engines.

5.1 Class `SearchEngineResult`



5.1.1 Methods

```
__init__(self, title, link, description="")
```

`x.__init__(...)` initializes `x`; see `help(type(x))` for signature

Overrides: `object.__init__` `extit`(inherited documentation)

```
__str__(self)
```

`str(x)`

Overrides: `object.__str__` `extit`(inherited documentation)

Inherited from object

```
__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),
__reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),
__subclasshook__()
```

5.1.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

5.2 Class `SearchEngine`



5.2.1 Methods

__init__ (<i>self</i> , <i>url</i> , <i>blank_replace</i> ="+")
<p>:param url: string Url of search engine used in all query.</p> <p>:param blank_replace: Every search engine has to replace blanks in query</p> <p>Overrides: object.__init__</p>

parse_query (<i>self</i> , <i>query</i>)
<p>:param query: string Query to search engine.</p> <p>:return: string Parse given query in order to meet search criteria of search engine.</p>

get_search_page (<i>self</i> , <i>query</i>)
<p>:param query: string Query to search engine.</p> <p>:return: string Get HTML source of search page of given query.</p>

Inherited from object

__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),
 __reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),
 __str__(), __subclasshook__()

5.2.2 Properties

Name	Description
<i>Inherited from object</i>	
__class__	

6 Module *hal.internet.parser*

Parse anything there is on the Internet.

6.1 Functions

is_string_well_formatted(*string*)

```
:param string: string
    String to parse
:return: bool
    True iff string is good formatted
```

html_stripper(*string*)

```
:param string: string
    String to parse
:return: string
    Given string with raw HTML elements removed
```

6.2 Variables

Name	Description
<code>__package__</code>	Value: <code>'hal.internet'</code>

6.3 Class *HtmlTable*



6.3.1 Methods

<code>__init__(self, html_source)</code>
<p><code>:param html_source: string</code> Html source of table</p> <p>Overrides: <code>object.__init__</code></p>
<code>parse(self)</code>
<p><code>:return: list of list</code> List of list of values in table</p>

Inherited from str

`__add__()`, `__contains__()`, `__eq__()`, `__format__()`, `__ge__()`, `__getattr__()`, `__getitem__()`, `__getnewargs__()`, `__getslice__()`, `__gt__()`, `__hash__()`, `__le__()`, `__len__()`, `__lt__()`, `__mod__()`, `__mul__()`, `__ne__()`, `__new__()`, `__repr__()`, `__rmod__()`, `__rmul__()`, `__sizeof__()`, `__str__()`, `capitalize()`, `center()`, `count()`, `decode()`, `encode()`, `endswith()`, `expandtabs()`, `find()`, `format()`, `index()`, `isalnum()`, `isalpha()`, `isdigit()`, `islower()`, `isspace()`, `istitle()`, `isupper()`, `join()`, `ljust()`, `lower()`, `lstrip()`, `partition()`, `replace()`, `rfind()`, `rindex()`, `rjust()`, `rpartition()`, `rsplit()`, `rstrip()`, `split()`, `splitlines()`, `startswith()`, `strip()`, `swapcase()`, `title()`, `translate()`, `upper()`, `zfill()`

Inherited from object

`__delattr__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`, `__subclasshook__()`

6.3.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

7 Module hal.internet.selenium

Some utils methods for a selenium webdriver

7.1 Variables

Name	Description
<code>__package__</code>	Value: None

7.2 Class SeleniumForm

Great and simple static methods to deal with selenium webdrivers.

7.2.1 Methods

```
fill_form_field(browser, field_name, field_value)
```

```
:param browser: webdriver
    Browser to use to submit form.
:param field_name :string
    Name of field to fill
:param field_value: string
    Value with which to fill field.
:return: void
    Fill given field wiht given value.
```

```
fill_login_form(browser, username, username_field, userpassword,  
userpassword_field)
```

```
:param browser: webdriver  
    Browser to use to submit form.  
:param username: string  
    Username of user to login.  
:param username_field: string  
    Name of field to fill with username.  
:param userpassword: string  
    Password of user to login.  
:param userpassword_field: string  
    Name of field to fill with userpassword.  
:return: void  
    Form filled with given information.
```

```
submit_form(browser, button_name)
```

```
:param browser: webdriver  
    Browser to use to submit form.  
:param button_name: string  
    Name of button to press to submit form  
:return: void  
    Submit form.
```

8 Module *hal.internet.web*

Deal with webpages.

8.1 Functions

download_url (<i>url</i> , <i>local_file</i>) <hr/> <pre> :param url: string Url to download :param local_file: string Save url as this path :return: void Download link to local file </pre>

8.2 Variables

Name	Description
CHROME_USER_AGE-NT	Value: ["Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US) AppleWe...

8.3 Class Webpage

object —
hal.internet.web.Webpage

representation of URL (web page)

8.3.1 Methods

<code>__init__(self, url, using_tor=True)</code> <hr/> :param url: string Url of webpage :param using_tor: bool Whether using tor or not to fetch source page Overrides: object.__init__
<code>parse_url(raw_url)</code> <hr/> :param raw_url: url to parse :return: parses correctly url
<code>get_scheme(self)</code> <hr/> :return: get scheme (HTTP, HTTPS, FTP ..) from given url
<code>get_hostname(self)</code> <hr/> :return: extract hostname from given url
<code>get_domain(self)</code> <hr/> :return: get domain from given url
<code>get_html_source(self, tor=False)</code> <hr/> :return: BeautifulSoup to parse
<code>get_links(self, recall, timeout)</code> <hr/> :param recall: max time to attempt to fetch url :param timeout: max time (s) to wait for web_page response :return: array of out_links
<code>open_in_browser(self, times)</code> <hr/> :param times: int Times to open webpage in browser :return: void Open a wendrive and go to webpage

Inherited from object

`__delattr__()`, `__format__()`, `__getattr__()`, `__hash__()`, `__new__()`,
`__reduce__()`, `__reduce_ex__()`, `__repr__()`, `__setattr__()`, `__sizeof__()`,
`__str__()`, `__subclasshook__()`

8.3.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

9 Module *hal.internet.youtube*

Get rss feed for youtube channel.

9.1 Functions

```
get_channel_page(channel_name,  
youtube_channel_url="https://www.youtube.com/user/")
```

```
@param channel_name: string  
    name of channel.  
@param youtube_channel_url: string  
    base url of youtube channels.  
@return string  
    source page of youtube channel.
```

```
get_channel_id(channel_name,  
channel_id_field="data-channel-external-id")
```

```
@param channel_name: string  
    channel_name name of channel.  
@param channel_id_field: string  
    default field to get channel id.  
@return string  
    id of youtube channel.
```

```
get_channel_feed_url(channel_name,  
base_feed_url="https://www.youtube.com/feeds/videos.xml?channel_id=")
```

```
@param channel_name: string  
    channel_name name of channel.  
@param base_feed_url: string  
    default base url for rss feed of youtube channels.  
@return string  
    rss url feed of youtube channel.
```

10 Package **hal.maths**

MATHS: important and scalable math functions

10.1 Modules

- **crypt**: Perform fast hash, encryption and calculations related to cryptography.
(Section 11, p. 29)
- **maths**: A few elegant and powerful mathematical functions.
(Section 12, p. 39)
- **plotter**: Show elegant plots in any dimension.
(Section 13, p. 42)

10.2 Variables

Name	Description
<code>__package__</code>	Value: None

11 Module *hal.maths.crypt*

Perform fast hash, encryption and calculations related to cryptography.

11.1 Class MD5

object └─
 hal.maths.crypt.MD5
md5 hash

11.1.1 Methods

<code>__init__(self, string)</code>

x.`__init__`(...) initializes x; see `help(type(x))` for signature

Overrides: object.`__init__` `exit`(inherited documentation)

<code>hash(self)</code>

:return: hash plaintext

Inherited from object

`__delattr__`(), `__format__`(), `__getattr__`(), `__hash__`(), `__new__`(),
`__reduce__`(), `__reduce_ex__`(), `__repr__`(), `__setattr__`(), `__sizeof__`(),
`__str__`(), `__subclasshook__`()

11.1.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

11.2 Class MD6

object └─
 hal.maths.crypt.MD6
md6 hash

11.2.1 Methods

__init__(*self*, *string*, *size*)

x.__init__(...) initializes *x*; see help(type(*x*)) for signature

Overrides: object.__init__ *exitit*(inherited documentation)

hash(*self*)

:return: return md6 hash

hex(*self*, *data*, *size*)

:param *data*: plaintext :param *size*: bytes :return: hex representation

raw(*self*, *data*, *size*)

:param *data*: plaintext :param *size*: bytes :return: raw representation

Inherited from object

__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),
__reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),
__str__(), __subclasshook__()

11.2.2 Properties

Name	Description
<i>Inherited from object</i>	
__class__	

11.2.3 Class Variables

Name	Description
ALLOWED_SIZE	Value: [64, 128, 224, 256, 384, 512]

11.3 Class SHA

object —
 hal.maths.crypt.SHA
general SHA hash

11.3.1 Methods

<code>__init__(self, string, size, salt=None)</code>

`x.__init__(...)` initializes `x`; see `help(type(x))` for signature

Overrides: `object.__init__` `exitit` (inherited documentation)

<code>hash(self)</code>

:return: hash of given size

<code>hash_sha1(self)</code>

:return: sha1 hash

<code>hash_sha224(self)</code>

:return: sha224 hash

<code>hash_sha256(self)</code>

:return: sha256 hash

<code>hash_sha384(self)</code>

:return: sha384 hash

<code>hash_sha512(self)</code>

:return: sha512 hash

<code>hash_shasalted(self)</code>

:return: sha512 hash

Inherited from object

`__delattr__()`, `__format__()`, `__getattr__()`, `__hash__()`, `__new__()`,
`__reduce__()`, `__reduce_ex__()`, `__repr__()`, `__setattr__()`, `__sizeof__()`,
`__str__()`, `__subclasshook__()`

11.3.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

11.3.3 Class Variables

Name	Description
ALLOWED_SIZE	Value: [1, 224, 256, 384, 512]

11.4 Class DES

object └─ **hal.maths.crypt.DES**

DES hash

11.4.1 Methods

__init__(*self*, *string*, *key*, *size*)

x.**__init__**(...) initializes x; see help(type(x)) for signature

Overrides: object.**__init__** extit(inherited documentation)

hash(*self*)

:return: hash of given size

hash_des(*self*)

:return: des hash

hash_des3(*self*)

:return: des3 hash

Inherited from object

__delattr__(), **__format__**(), **__getattr__**(), **__hash__**(), **__new__**(),
__reduce__(), **__reduce_ex__**(), **__repr__**(), **__setattr__**(), **__sizeof__**(),
__str__(), **__subclasshook__**()

11.4.2 Properties

Name	Description
<i>Inherited from object</i> __class__	

11.4.3 Class Variables

Name	Description
ALLOWED_SIZE	Value: [1, 3]

11.5 Class ARC

object └─ **hal.maths.crypt.ARC**

ARC hash

11.5.1 Methods

__init__(*self*, *string*, *key*, *size*)

x.**__init__**(...) initializes x; see help(type(x)) for signature

Overrides: object.**__init__** extit(inherited documentation)

hash(*self*)

:return: hash of given size

hash_ar2(*self*)

:return: des hash

hash_arc4(*self*)

:return: des3 hash

Inherited from object

__delattr__(), **__format__**(), **__getattr__**(), **__hash__**(), **__new__**(),
__reduce__(), **__reduce_ex__**(), **__repr__**(), **__setattr__**(), **__sizeof__**(),
__str__(), **__subclasshook__**()

11.5.2 Properties

Name	Description
<i>Inherited from object</i> __class__	

11.5.3 Class Variables

Name	Description
ALLOWED_SIZE	Value: [2, 4]

11.6 Class AES

object —
 hal.maths.crypt.AES
 aes hash

11.6.1 Methods

__init__ (<i>self</i> , <i>string</i> , <i>key</i>)
x. __init__ (...) initializes x; see help(type(x)) for signature
Overrides: object. __init__ extit(inherited documentation)

hash (<i>self</i>)
:return: hash plaintext

Inherited from object

__delattr__(*self*), **__format__**(*self*), **__getattr__**(*self*), **__hash__**(*self*), **__new__**(*cls*, *args, **kwargs),
__reduce__(*self*), **__reduce_ex__**(*self*), **__repr__**(*self*), **__setattr__**(*self*, name, value), **__sizeof__**(*self*),
__str__(*self*), **__subclasshook__**(*self*)

11.6.2 Properties

Name	Description
<i>Inherited from object</i>	
__class__	

11.7 Class HMAC

object —
 hal.maths.crypt.HMAC

hmac hash

11.7.1 Methods

`__init__(self, string, key)`

`x.__init__(...)` initializes `x`; see `help(type(x))` for signature

Overrides: `object.__init__` `exitit`(inherited documentation)

`hash(self)`

:return: hash plaintext

Inherited from object

`__delattr__()`, `__format__()`, `__getattr__()`, `__hash__()`, `__new__()`,
`__reduce__()`, `__reduce_ex__()`, `__repr__()`, `__setattr__()`, `__sizeof__()`,
`__str__()`, `__subclasshook__()`

11.7.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

11.8 Class BLOWFISH

object —
 hal.maths.crypt.BLOWFISH

blowfish hash

11.8.1 Methods

`__init__(self, string, key)`

`x.__init__(...)` initializes `x`; see `help(type(x))` for signature

Overrides: `object.__init__` `exitit`(inherited documentation)

hash (<i>self</i>)

:return: hash plaintext

Inherited from object

```
__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),
__reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),
__str__(), __subclasshook__()
```

11.8.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

11.9 Class IDEA

```
object └─ hal.maths.crypt.IDEA
```

IDEA hash

11.9.1 Methods

__init__ (<i>self</i> , <i>string</i> , <i>key</i>)

x. __init__ (...) initializes x; see help(type(x)) for signature

Overrides: object. __init__ extit(inherited documentation)

hash (<i>self</i>)

:return: IDEA hash

change_key (<i>self</i> , <i>key</i>)

:param key: new key :return: change key

encrypt (<i>self</i>)

:return: encrypt with key

Inherited from object

```

__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),
__reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),
__str__(), __subclasshook__()

```

11.9.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

11.10 Class CAST128

```

object └─
          hal.maths.crypt.CAST128

```

CAST 128 hash

11.10.1 Methods

<code>__init__(self, string, key)</code> x. <code>__init__</code> (...) initializes x; see <code>help(type(x))</code> for signature Overrides: <code>object.__init__</code> <code>exitit</code> (inherited documentation)
<code>encrypt(self)</code>
<code>decrypt(self)</code>

Inherited from object

```

__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),
__reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),
__str__(), __subclasshook__()

```

11.10.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

11.11 Class Dsa

object —
 hal.maths.crypt.Dsa
 dsa hash

11.11.1 Methods

__init__ (<i>self</i> , <i>string</i>)

x.**__init__**(...) initializes x; see help(type(x)) for signature

Overrides: object.**__init__** extit(inherited documentation)

hash (<i>self</i>)

:return: hash plaintext

Inherited from object

__delattr__(*self*), **__format__**(*self*), **__getattr__**(*self*), **__hash__**(*self*), **__new__**(*cls*, *args, **kwargs),
__reduce__(*self*), **__reduce_ex__**(*self*, protocol), **__repr__**(*self*), **__setattr__**(*self*, name, value), **__sizeof__**(*self*),
__str__(*self*), **__subclasshook__**(*self*)

11.11.2 Properties

Name	Description
<i>Inherited from object</i>	
__class__	

12 Module *hal.maths.maths*

A few elegant and powerful mathematical functions.

12.1 Functions

get_prime (<i>bits</i>)

:param bits: size of number to generate (bits) :return: prime number of given size

blumblumshub (<i>seed, amount, prime0, prime1</i>)

:param seed: seeder :param amount: amount of number to generate :param prime0: one prime number :param prime1: the second prime number :return: pseudo-number generator

12.2 Variables

Name	Description
__package__	Value: 'hal.maths'

12.3 Class Integer

```

object └─
        hal.maths.maths.Integer

```

12.3.1 Methods

__init__ (<i>self, string</i>)

x.__init__(...) initializes x; see help(type(x)) for signature

Overrides: object.__init__ extit(inherited documentation)

is_probably_prime (<i>self</i>)

:return: test with miller-rabin

```
test_miller_rabin(self, precision)
```

```
:param precision: number of rounds to perform (higher -> better precision)
:return: True iff probably prime
```

Inherited from object

```
__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),
__reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),
__str__(), __subclasshook__()
```

12.3.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

12.3.3 Class Variables

Name	Description
LOW_PRIMES	Value: [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, ...]

12.4 Class EightQueen

```
object └─ hal.maths.maths.EightQueen
```

8 queen problem solver

12.4.1 Methods

```
__init__(self, board_size)
```

```
x.__init__(...) initializes x; see help(type(x)) for signature
Overrides: object.__init__ extit(inherited documentation)
```

```
under_attack(self, col, queens)
```

```
solve(self, n)
```


Inherited from object

```
__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),  
__reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),  
__str__(), __subclasshook__()
```

12.4.2 Properties

Name	Description
<i>Inherited from object</i> __class__	

13 Module hal.maths.plotter

Show elegant plots in any dimension.

13.1 Class Plot2d

object —
 hal.maths.plotter.Plot2d

2d plot

13.1.1 Methods

scatter(*vectorx*, *vectory*)

:param vectorx: vector in x axis :param vectory: vector in y axis :return: 2d scatter plot

param(*self*, *functionx*, *functiony*, *min*, *max*, *points*)

:param functionx: function in x value :param functiony: function in y value
 ::param min: minimum value :param max: maximum value :param points:
 number of points to display :return: 2d parametric graph of given function
 from min to max

plot(*self*, *function*, *min*, *max*, *points*)

:param function: function to plot :param min: minimum value :param max:
 maximum value :param points: number of points :return: plot 2d function

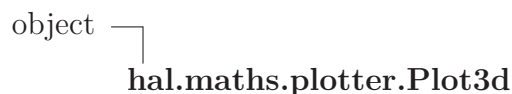
Inherited from object

__delattr__(), __format__(), __getattr__(), __hash__(), __init__(),
 __new__(), __reduce__(), __reduce_ex__(), __repr__(), __setattr__(),
 __sizeof__(), __str__(), __subclasshook__()

13.1.2 Properties

Name	Description
<i>Inherited from object</i>	
__class__	

13.2 Class Plot3d



13.2.1 Methods

scatter(*vectorx*, *vectory*, *vectorz*)

:param vectorx: vector in x axis :param vectory: vector in y axis :param vectorz: vector in z axis :return: plot 3d scattered points

param(*self*, *functionx*, *functiony*, *functionz*, *min*, *max*, *points*)

:param functionx: function in x :param functiony: function in y :param functionz: function in z :param min: minimum :param max: maximum :param points: number of points :return: 3d parametric graph of given function from min to max

plot(*self*, *function*, *minx*, *maxx*, *pointsx*, *miny*, *maxy*, *pointsy*)

:param function: function to plot :param minx: minimum of x-values :param maxx: maximum of x-values :param pointsx: points in x axis :param miny: minimum of y-values :param maxy: maximum of y-values :param pointsy: points in y axis :return: plot 3d function

Inherited from object

`__delattr__()`, `__format__()`, `__getattr__()`, `__hash__()`, `__init__()`,
`__new__()`, `__reduce__()`, `__reduce_ex__()`, `__repr__()`, `__setattr__()`,
`__sizeof__()`, `__str__()`, `__subclasshook__()`

13.2.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

13.3 Class Plot4d

object —
 hal.maths.plotter.Plot4d

13.3.1 Methods

scatter(*vectorx, vectory, vectorz, vectorw*)

:param vectorx: vector in x axis :param vectory: vector in y axis :param vectorz: vector in z axis :param vectorw: vector in w axis :return: plot 4d scattered points

param(*self, functionx, functiony, functionz, functionw, min, max, points*)

:param functionx: function in x :param functiony: function in y :param functionz: function in z :param functionw: function in w :param min: minimum :param max: maximum :param points: number of points :return: 4d parametric graph of given function from min to max

plot(*self, function, minx, maxx, miny, maxy, minz, maxz, precision, kind*)

:param function: function to plot
 :param minx: minimum of x-values
 :param maxx: maximum of x-values
 :param miny: minimum of y-values
 :param maxy: maximum of y-values
 :param minz: minimum of z-values
 :param maxz: maximum of z-values
 :param precision: precision
 :param kind: slice: x cont -> 3d plot with y,z variables in plane and w as "z"-axis
 contour: x cont -> 3d plot with y,z variables in plane and w colored
 :return: plot 4d function

Inherited from object

__delattr__(), __format__(), __getattr__(), __hash__(), __init__(),
 __new__(), __reduce__(), __reduce_ex__(), __repr__(), __setattr__(),
 __sizeof__(), __str__(), __subclasshook__()

13.3.2 Properties

Name	Description
<i>Inherited from object</i> __class__	

14 Package *hal.ml*

14.1 Modules

- **data** (*Section 15, p. 47*)
 - **parser**: Parsers for raw databases.
(*Section 16, p. 48*)
- **features**: Collection of methods to find weights of features and select the best ones.
(*Section 17, p. 50*)
- **models** (*Section 18, p. 51*)
 - **classification**: Prediction methods based on classification algorithms.
(*Section 19, p. 52*)
 - **pipelined**: Prediction methods based on multiple models mixed up.
(*Section 20, p. 53*)
 - **regression**: Prediction methods based on regression algorithms.
(*Section 21, p. 54*)
 - **time_series**: Multi-purpose prediction methods to be used in time-series.
(*Section 22, p. 55*)
- **predict**: " General model to make prediction about everything.
(*Section 23, p. 56*)
- **utils**: Various tools and utilities to deal with database and machine learning.
(*Section 24, p. 57*)

14.2 Variables

Name	Description
__package__	Value: None

15 Package *hal.ml.data*

15.1 Modules

- **parser**: Parsers for raw databases.
(Section 16, p. 48)

15.2 Variables

Name	Description
__package__	Value: None

16 Module *hal.ml.data.parser*

Parsers for raw databases.

16.1 Variables

Name	Description
<code>__package__</code>	Value: None

16.2 Class Parser

object └─ **hal.ml.data.parser.Parser**

Known Subclasses: *hal.ml.data.parser.CSVParser*

16.2.1 Methods

<code>__init__(self, database_file)</code>
:param database_file: a raw .csv file that contains any data about anything
Overrides: object. <code>__init__</code>

<code>get_lines(self)</code>

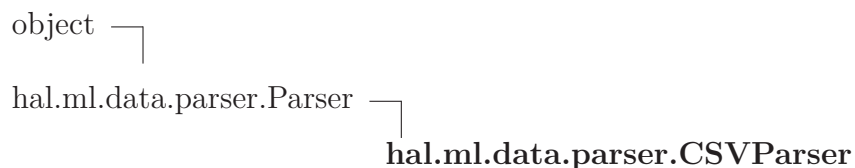
Inherited from object

`__delattr__()`, `__format__()`, `__getattr__()`, `__hash__()`, `__new__()`,
`__reduce__()`, `__reduce_ex__()`, `__repr__()`, `__setattr__()`, `__sizeof__()`,
`__str__()`, `__subclasshook__()`

16.2.2 Properties

Name	Description
<i>Inherited from object</i> <code>__class__</code>	

16.3 Class CSVParser



16.3.1 Methods

__init__ (<i>self</i> , <i>database_file</i>)

:param database_file: a raw .csv file that contains any data about anything

Overrides: object.__init__

parse_data (<i>self</i>)

store values in array, store lines in array; the result is a 2D matrix

Inherited from hal.ml.data.parser.Parser(Section 16.2)

get_lines()

Inherited from object

__delattr__(), __format__(), __getattr__(), __hash__(), __new__(),
 __reduce__(), __reduce_ex__(), __repr__(), __setattr__(), __sizeof__(),
 __str__(), __subclasshook__()

16.3.2 Properties

Name	Description
<i>Inherited from object</i>	
__class__	

17 Module *hal.ml.features*

Collection of methods to find weights of features and select the best ones.

17.1 Functions

select_k_best (x, y, k)
select k best features in dataset

get_best_features (x, y)
finds the optimal number of features

get_features ($x, y, n_features_to_select$)
finds the optimal features

18 Package *hal.ml.models*

18.1 Modules

- **classification**: Prediction methods based on classification algorithms.
(Section 19, p. 52)
- **pipelined**: Prediction methods based on multiple models mixed up.
(Section 20, p. 53)
- **regression**: Prediction methods based on regression algorithms.
(Section 21, p. 54)
- **time_series**: Multi-purpose prediction methods to be used in time-series.
(Section 22, p. 55)

18.2 Variables

Name	Description
<code>__package__</code>	Value: None

19 Module `hal.ml.models.classification`

Prediction methods based on classification algorithms.

19.1 Functions

<code>extra_trees_classifier()</code>

<code>random_forest()</code>

<code>knn()</code>

very fast and slightly more accurate than AdaBoost

<code>ada_boost()</code>

fast, accurate but too uncertainty

<code>bayes_gauss()</code>

slower than svr but equally accurate

<code>bayes_bernoulli()</code>

20 Module `hal.ml.models.pipelined`

Prediction methods based on multiple models mixed up.

20.1 Functions

<code>logistic_rbm()</code>

<code>anova_svm()</code>

21 Module `hal.ml.models.regression`

Prediction methods based on regression algorithms.

21.1 Functions

<code>support_vector_machine()</code>

super fast and precise

<code>logistic_regression()</code>

22 Module *hal.ml.models.time_series*

Multi-purpose prediction methods to be used in time-series.

22.1 Functions

test_stationarity (<i>timeseries</i>)

arma (<i>dates, values, start=None, end=None, plot=False</i>)

Predict days values using ARMA algorithm. :param dates: list of str date :param values: list of float values :param start: start predicting in this day :param end: end of prediction :param plot: whether to plot or not values in graph

arima (<i>dates, values, start=None, end=None</i>)

Predict days values using ARIMA algorithm. :param dates: list of str date :param values: list of float values :param start: start predicting in this day :param end: end of prediction

var (<i>dates, values, start=None, end=None</i>)

Predict days values using ARIMA algorithm. :param dates: list of str date :param values: list of float values :param start: start predicting in this day :param end: end of prediction

dynamic_var (<i>dates, values, start=None, end=None</i>)

Predict days values using ARIMA algorithm. :param dates: list of str date :param values: list of float values :param start: start predicting in this day :param end: end of prediction

23 Module hal.ml.predict

" General model to make prediction about everything.

23.1 Class BasePrediction

object └─
 hal.ml.predict.BasePrediction

23.1.1 Methods

<code>__init__(self, model, rounds)</code>

x.`__init__`(...) initializes x; see `help(type(x))` for signature

Overrides: object.`__init__` extit(inherited documentation)

<code>train(self, x, y)</code>

Inherited from object

`__delattr__()`, `__format__()`, `__getattr__()`, `__hash__()`, `__new__()`,
`__reduce__()`, `__reduce_ex__()`, `__repr__()`, `__setattr__()`, `__sizeof__()`,
`__str__()`, `__subclasshook__()`

23.1.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

24 Module *hal.ml.utils*

Various tools and utilities to deal with database and machine learning.

24.1 Functions

precision(*matrix*)

Calculates accuracy on database

:param matrix: 2x2 matrix that looks like
 True Positive - False Negative
 | - |
 False Positive - True Negative

recall(*matrix*)

Calculates recall on database

:param matrix: 2x2 matrix that looks like
 True Positive - False Negative
 | - |
 False Positive - True Negative

tn_rate(*matrix*)

Calculates true negative rate on database

:param matrix: 2x2 matrix that looks like
 True Positive - False Negative
 | - |
 False Positive - True Negative

accuracy(*matrix*)

Calculates recall on database

:param matrix: 2x2 matrix that looks like
 True Positive - False Negative
 | - |
 False Positive - True Negative

f1_score(*matrix*)

Calculates f1 score on database

:param matrix: 2x2 matrix that looks like

True Positive - False Negative

| - |

False Positive - True Negative

pearson(*x, y*)

Pearson coefficient of arrays

show_correlation_matrix(*feature_list, correlation_matrix*)

Show the given correlation matrix as image

25 Package hal.profile

25.1 Modules

- **performance:** Perform benchmarks and tests on your PC.
(Section 26, p. 60)

25.2 Variables

Name	Description
__package__	Value: None

26 Module *hal.profile.performance*

Perform benchmarks and tests on your PC.

26.1 Variables

Name	Description
<code>__package__</code>	Value: <code>'hal.profile'</code>

26.2 Class *EightQueenTest*

object └─ ***hal.profile.performance.EightQueenTest***

test CPU by solving eight-queen problem

26.2.1 Methods

`__init__(self, size)`

`x.__init__(...)` initializes `x`; see `help(type(x))` for signature

Overrides: `object.__init__` `extit`(inherited documentation)

`welcome()`

:return: introduce script

`introduction()`

:return: introduce 8 queen problem

`run(self)`

Inherited from object

`__delattr__()`, `__format__()`, `__getattr__()`, `__hash__()`, `__new__()`,
`__reduce__()`, `__reduce_ex__()`, `__repr__()`, `__setattr__()`, `__sizeof__()`,
`__str__()`, `__subclasshook__()`

26.2.2 Properties

Name	Description
<i>Inherited from object</i> __class__	

27 Package *hal.wrappers*

27.1 Modules

- **methods:** Typical (and useful) function wrappers
(*Section 28, p. 63*)

27.2 Variables

Name	Description
<code>__package__</code>	Value: None

28 Module *hal.wrappers.methods*

Typical (and useful) function wrappers

28.1 Functions

<code>handle_exceptions(<i>function</i>)</code>
<code>:param function: callback function</code> function to wrap
<code>:return: callback function return type</code> wraps callback function

Index

- hal (*package*), 5–6
 - hal.files (*package*), 7
 - hal.files.models (*module*), 8–16
 - hal.internet (*package*), 17
 - hal.internet.engines (*module*), 18–19
 - hal.internet.parser (*module*), 20–21
 - hal.internet.selenium (*module*), 22–23
 - hal.internet.web (*module*), 24–26
 - hal.internet.youtube (*module*), 27
 - hal.maths (*package*), 28
 - hal.maths.crypt (*module*), 29–38
 - hal.maths.maths (*module*), 39–41
 - hal.maths.plotter (*module*), 42–45
 - hal.ml (*package*), 46
 - hal.ml.data (*package*), 47
 - hal.ml.features (*module*), 50
 - hal.ml.models (*package*), 51
 - hal.ml.predict (*module*), 56
 - hal.ml.utils (*module*), 57–58
 - hal.profile (*package*), 59
 - hal.profile.performance (*module*), 60–61
 - hal.wrappers (*package*), 62
 - hal.wrappers.methods (*module*), 63