Package 'panstarrs'

October 14, 2022

Title Interface to the Pan-STARRS API

Version 0.1.0

Description An interface to the API for 'Pan-STARRS1', a data archive of the PS1 wide-field astronomical survey. The package allows access to the PS1 catalog and to the PS1 images. (see <https://outerspace.stsci.edu/display/PANSTARRS/> for more information). You can use it to plan astronomical observations, make guidance pictures, find magnitudes in five broadband filters (g, r, i, z, y) and more.

License MIT + file LICENSE

URL https://uskovgs.github.io/PanSTARRS/

BugReports https://github.com/uskovgs/PanSTARRS/issues

Imports attempt, dplyr, glue, httr, jsonlite, magrittr, purrr, RCurl, readr, rlang, stringr

Suggests celestial, FITSio, knitr, magicaxis, magick, rmarkdown

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.1.2

NeedsCompilation no

Author Grigory Uskov [cre, aut]

Maintainer Grigory Uskov <uskov.russia@gmail.com>

Repository CRAN

Date/Publication 2022-02-07 09:20:02 UTC

R topics documented:

`checklegal` *Check legal*

Description

Checks if this combination of table and release is acceptable.

Usage

```
checklegal(table, release)
```

Arguments

`table`	"mean", "stack", "detection"
`release`	"dr2", "dr1"

`ps1_cone` *Do a cone search of the PS1 catalog*

Description

Do a cone search of the PS1 catalog

Usage

```
ps1_cone(
  ra,
  dec,
  r_arcmin = 0.05,
  table = c("mean", "stack", "detection"),
  release = c("dr2", "dr1"),
  columns = NULL,
  verbose = FALSE,
  ...
)
```

Arguments

ra	(degrees) J2000 Right Ascension
dec	(degrees) J2000 Declination
r_arcmin	(arcmins) Search radius (<= 30 arcmins)
table	"mean"(default), "stack", or "detection"
release	"dr1" or "dr2"(default)
columns	list of column names to include (NULL means use defaults)
verbose	print info about request
...	other parameters (e.g., nDetections.min = 2)

Value

data.frame

Examples

```
## Not run:
ps1_cone(ra = 139.334,dec = 68.635,r_arcmin = 0.05, nDetections.gt = 1)

## End(Not run)
```

---

ps1_crossmatch *Do a cross-match with PS1 catalog*

Description

Do a cross-match with PS1 catalog

Usage

```
ps1_crossmatch(
  ra,
  dec,
  r_arcmin = 0.05,
  table = c("mean", "stack", "detection"),
  release = c("dr2", "dr1"),
  verbose = FALSE
)
```

Arguments

ra	(degrees) numeric vector of J2000 Right Ascension
dec	(degrees) numeric vector of J2000 Declination
r_arcmin	(arcmins) Search radius (<= 30 arcmins)
table	"mean"(default), "stack", or "detection"
release	"dr1" or "dr2"(default)
verbose	print info about request

Value

data.frame

Examples

```
## Not run:
ps1_crossmatch(ra = c(268.70342, 168.87258), dec = c(71.54292, 60.75153))

## End(Not run)
```

`ps1_image_color` *Get color image at a sky position*

Description

Get color image at a sky position

Usage

```
ps1_image_color(
  ra,
  dec,
  size = 240,
  output_size = NULL,
  filters = "grizy",
  format = "jpg"
)
```

Arguments

`ra`	ra position in degrees
`dec`	dec position in degrees
`size`	extracted image size in pixels (0.25 arcsec/pixel)
`output_size`	output (display) image size in pixels (default = size). output_size has no effect for fits format images.
`filters`	string with filters to include
`format`	data format (options are "jpg", "png")

Value

the image url

Examples

```
## Not run:
ps1_image_color(ra = 83.633210, dec = 22.014460, size = 1280, filters="grz")

## End(Not run)
```

ps1_image_gray *Get grayscale image at a sky position*

Description

Get grayscale image at a sky position

Usage

```
ps1_image_gray(
  ra,
  dec,
  size = 240,
  output_size = NULL,
  filter = "g",
  format = "jpg"
)
```

Arguments

ra	ra position in degrees
dec	dec position in degrees
size	extracted image size in pixels (0.25 arcsec/pixel)
output_size	output (display) image size in pixels (default = size). output_size has no effect for fits format images.
filter	string with filter to extract (one of grizy)
format	data format (options are "jpg", "png")

Value

the image

Examples

```
## Not run:
ps1_image_gray(ra = 83.633210, dec = 22.014460, size = 1280, filter = "i")

## End(Not run)
```

ps1_image_list *Get list of images*

Description

Query ps1filenames.py service to get a list of images.

Usage

```
ps1_image_list(ra, dec, size = 240, filters = "grizy")
```

Arguments

ra	ra position in degrees
dec	dec position in degrees
size	image size in pixels (0.25 arcsec/pixel)
filters	string with filters to include

Details

src: https://ps1images.stsci.edu/ps1image.html

Value

table with the results

Examples

```
## Not run:
# Crab nebulae image
ps1_image_list(ra = 83.633210, dec = 22.014460, size = 1280, filters = "grz")

## End(Not run)
```

ps1_image_url *Get URL of images*

Description

Get URL of images

Usage

```
ps1_image_url(
  ra,
  dec,
  size = 240,
  output_size = NULL,
  filters = "grizy",
  format = "jpg",
  color = FALSE
)
```

Arguments

ra	ra position in degrees
dec	dec position in degrees
size	extracted image size in pixels (0.25 arcsec/pixel)
output_size	output (display) image size in pixels (default = size). output_size has no effect for fits format images.
filters	string with filters to include
format	data format (options are "jpg", "png" or "fits")
color	if TRUE, creates a color image (only for jpg or png format). Default is return a list of URLs for single-filter grayscale images.

Value

string with the URL

Examples

```
## Not run:
ps1_image_url(
ra = 83.633210,
dec = 22.014460,
size = 1280,
format = "jpg",
filters = "grz",
color = T)

## End(Not run)
```

ps1_mast_query *Perform a MAST query.*

Description

Perform a MAST query.

Usage

```
ps1_mast_query(request)
```

Arguments

request	(list): The MAST request json object

Value

Returns response

ps1_mast_resolve *Get the RA and Dec for an object using the MAST name resolver*

Description

Get the RA and Dec for an object using the MAST name resolver

Usage

```
ps1_mast_resolve(name)
```

Arguments

name	Name of object

Value

list of ra, decl

Examples

```
## Not run:
ps1_mast_resolve('Acrux')

## End(Not run)
```

ps1_metadata *Metadata from PS1*

Description

Return metadata for the specified catalog and table

Usage

```
ps1_metadata(
  table = c("mean", "stack", "detection"),
  release = c("dr2", "dr1")
)
```

Arguments

table	"mean", "stack", or "detection"
release	"dr1" or "dr2"(default)

Value

Returns data.frame with columns: name, type, description

Examples

```
## Not run:
ps1_metadata()

## End(Not run)
```

ps1_resolve *Get the RA and Dec for objects from PanSTARRS catalog.*

Description

Only works for "north" objects with decl > -30. For all objects see function ‘ps1_mast_resolve‘.

Usage

```
ps1_resolve(target_names, full_table = FALSE, verbose = FALSE)
```

Arguments

target_names	character vector of target names (see example)
full_table	show full cross-matched table or only main columns.
verbose	print info about request

Value

data.frame

Examples

```
## Not run:
ps1_resolve(c('Andromeda', "SN 2005D", 'Antennae', 'ANTENNAE'))

## End(Not run)
```

`ps1_search` *Do a general search of the PS1 catalog (possibly without ra/dec/radius)*

Description

Do a general search of the PS1 catalog (possibly without ra/dec/radius)

Usage

```
ps1_search(
  table = c("mean", "stack", "detection"),
  release = c("dr2", "dr1"),
  columns = NULL,
  verbose = FALSE,
  ...
)
```

Arguments

`table`	"mean", "stack", or "detection"
`release`	"dr1" or "dr2"(default)
`columns`	list of column names to include (NULL means use defaults)
`verbose`	print info about request
`...`	other parameters (e.g., nDetections.min = 2).

Value

data.frame

Examples

```
## Not run:
ps1_search(
table='detection',
release='dr2',
objid = '190361393344112894')

ps1_search(
table='mean',
release='dr2',
objid = '190361393344112894',
columns = c('objName', 'raMean', 'decMean', 'rMeanPSFMag'))

## End(Not run)
```

Index