

# Package ‘HPAStainR’

April 12, 2022

**Title** Queries the Human Protein Atlas Staining Data for Multiple Proteins and Genes

**Version** 1.4.1

**Description** This package is built around the HPAStainR function. The purpose of the HPAStainR function is to query the visual staining data in the Human Protein Atlas to return a table of staining ranked cell types. The function also has multiple arguments to personalize to output as well to include cancer data, csv readable names, modify the confidence levels of the results and more. The other functions exist exclusively to easily acquire the data required to run HPAStainR.

**License** Artistic-2.0

**Encoding** UTF-8

**LazyData** FALSE

**Depends** R (>= 4.1.0), dplyr, tidyr

**Imports** utils, stats, scales, stringr, tibble, shiny, data.table

**Suggests** knitr, BiocManager, qpdf, hpar, testthat, rmarkdown

**RoxygenNote** 7.1.1

**SystemRequirements** 4GB of RAM

**biocViews** GeneExpression, GeneSetEnrichment

**BugReports** <https://github.com/tnieuwe/HPAStainR>

**VignetteBuilder** knitr

**git\_url** <https://git.bioconductor.org/packages/HPAStainR>

**git\_branch** RELEASE\_3\_14

**git\_last\_commit** 70ebd09

**git\_last\_commit\_date** 2021-11-25

**Date/Publication** 2022-04-12

**Author** Tim O. Nieuwenhuis [aut, cre] (<<https://orcid.org/0000-0002-1995-3317>>)

**Maintainer** Tim O. Nieuwenhuis <[tnieuwe1@jhmi.edu](mailto:tnieuwe1@jhmi.edu)>

## R topics documented:

HPAStainR	2
HPA_data_downloader	4
HPA_summary_maker	5
shiny_HPAStainR	6

<b>Index</b>	<b>7</b>
--------------	----------

---

HPAStainR	<i>HPAStainR</i>
-----------	------------------

---

### Description

Uses a protein/gene list to query Human Protein Atlas (HPA) staining data.

### Usage

```
HPAStainR(
  gene_list,
  hpa_dat,
  cancer_dat = data.frame(),
  cancer_analysis = c("normal", "cancer", "both"),
  tissue_level = TRUE,
  stringency = c("normal", "high", "low"),
  scale_abundance = TRUE,
  round_to = 2,
  csv_names = TRUE,
  stained_gene_data = TRUE,
  tested_protein_column = TRUE,
  percent_or_count = c("percent", "count", "both"),
  drop_na_row = FALSE,
  test_type = c("fisher", "chi square"),
  adjusted_pvals = TRUE
)
```

### Arguments

<code>gene_list</code>	A list of proteins or genes that you want to query the HPA staining data with.
<code>hpa_dat</code>	The data frame of normal HPA staining data data, required to run HPAStainR.
<code>cancer_dat</code>	The data frame of pathologic HPA staining data, required to run HPAStainR.
<code>cancer_analysis</code>	A character string indicating inclusion of cancer data in the result, must be one of 'normal' (default), 'cancer', or 'both'.
<code>tissue_level</code>	A boolean that determines whether tissue level data for the cell types are included. Default is TRUE

stringency	A character string indicating how stringent the confidence level of the staining findings have to be. Must be 'normal' (default), 'high', or 'low'. This stringency is based on the 'Reliability' column from the hpa_dat object which varies from "Enhanced", "Supported", "Approved", to "Uncertain" in decreasing order of certainty. Low stringency includes all data, normal stringency includes "Enhanced", "Supported", and "Approved", while high stringency only includes "Enhanced" and "Supported". Further information about these categorizations can be found in the following link <a href="https://www.proteinatlas.org/about/assays+annotation">https://www.proteinatlas.org/about/assays+annotation</a>
scale_abundance	A boolean that determines whether you scale Staining Score based on the size of the gene list. Default is TRUE.
round_to	A numeric that determines how many decimals in numeric outputs are desired. Default 2.
csv_names	A Boolean determining if you want names suited for a csv file/pipeline, or for presentation. Default is TRUE giving csv names.
stained_gene_data	A boolean determining if there is a list of which proteins stained, TRUE is default.
tested_protein_column	A boolean determining if there is a column listing which proteins were tested, TRUE is default.
percent_or_count	A character string determining if percent of proteins stained, count of proteins stained, or both are shown for high, medium, and low staining. Must be 'percent' (default), 'count', or 'both'.
drop_na_row	A boolean that determines if cell types with no proteins tested are kept or dropped, default is FALSE.
test_type	A character vector for either "fisher" or "chi square", used to select the statistical test for determining cell type enrichment. The two options are Fisher's Exact Test and a Chi Square test. The original version of HPAStainR defaulted to the Chi Square test, however because this requires simulated values to run correctly, we suggest the usage of the Fisher's Exact Test for consistency.
adjusted_pvals	A boolean indicating if you want the p-values corrected for multiple testing. Default is TRUE.

**Value**

A tibble containing the results of HPAStainR.

**Details**

Calculation of the staining score below:

$$\left(\frac{h \times 100}{t}\right) + \left(\frac{m \times 50}{t}\right) + \left(\frac{l \times 25}{t}\right)$$

**Examples**

```
## Below will give you the results found on the shiny app website
## This example also uses HPA_data_downloader output as an example
HPA_data <- HPA_data_downloader(tissue_type = 'both', save_file = FALSE)
HPA_out <- HPASTainR(c('PRSS1', 'PNLIP', 'CELA3A', 'PRL'),
  HPA_data$hpa_dat,
  HPA_data$cancer_dat,
  'both')
```

---

HPA\_data\_downloader     *HPA\_data\_downloader*

---

**Description**

Used to download required data for HPASTainR

**Usage**

```
HPA_data_downloader(
  tissue_type = c("both", "normal", "cancer"),
  save_file = TRUE,
  save_location = "",
  version_date_normal = "last",
  version_date_cancer = "last",
  force_download = FALSE
)
```

**Arguments**

tissue_type	A character string that determines which HPA data you want to download from the website. Has to be 'both' (default), 'normal', or 'cancer'.
save_file	A boolean determining if you want the HPA data downloaded permanently or temporarily. Default is TRUE, meaning the file will be saved in the given 'save_location', default being the current working directory.
save_location	A character string indicating where you want the files to be saved if you are saving them. If the file(s) already exists in that location, those will be loaded instead of re-downloading the files.
version_date_normal	a character string indicating what date of the normal tissue file you want returned if you have saved one. If you say "last" it will return the most recently downloaded file, otherwise give a date in YYYY-MM-DD format.
version_date_cancer	a character string indicating what date of the cancer tissue file you want returned if you have saved one. If you say "last" it will return the most recently downloaded file, otherwise give a date in YYYY-MM-DD format.
force_download	Either a TRUE or FALSE indicating that the HPA files should be re-downloaded from the website, regardless of there being a local recent version. This should be done occasionally to make sure you have up to date data.

**Value**

List of dataframes or dataframe depending on tissue\_type argument. If tissue\_type == 'both' it will be a list of dataframes.

**Examples**

```
HPA_data <- HPA_data_downloader(tissue_type = 'both', save_file = FALSE)
## Access normal data
HPA_data$hpa_dat
## Access cancer data
HPA_data$cancer_dat

## Download only the normal tissue data
HPA_normal_data <- HPA_data_downloader('normal', save_file = FALSE)
```

---

HPA_summary_maker	<i>HPA_summary_maker</i>
-------------------	--------------------------

---

**Description**

Used to generate a summary file used in the second tab of the Shiny app version of HPASTainR

**Usage**

```
HPA_summary_maker(hpa_dat)
```

**Arguments**

hpa\_dat            The dataframe of normal tissue data downloaded by HPA\_data\_downloader()

**Value**

A dataframe summarizing the amount of proteins tested to detected, used for the shiny app.

**Examples**

```
## Load in data from downloader
HPA_data <- HPA_data_downloader(save_file = FALSE)
## Generate the summarized HPA file
hpa_summary <- HPA_summary_maker(HPA_data$hpa_dat)
```

---

shiny\_HPASTainR      *shiny\_HPASTainR*

---

**Description**

Runs HPASTainR as a the shiny app found at <https://32tim32.shinyapps.io/HPASTainR/>

**Usage**

```
shiny_HPASTainR(hpa_dat, cancer_dat, cell_type_data = NULL)
```

**Arguments**

**hpa\_dat**            A required dataframe that has the normal tissue dataframe (see `HPA_data_downloader`).

**cancer\_dat**        A required dataframe that has the cancer tissue dataframe (see `HPA_data_downloader`).

**cell\_type\_data**    An optional dataframe that comes out of the `hpa_summary_maker` function, only needed if you want the second tab of HPASTainR, which shows the ratio of tested proteins to stained proteins, to be functional.

**Value**

A locally ran shiny app

**Examples**

```
## Load in data from downloader
HPA_data <- HPA_data_downloader(save_file = FALSE)
## Generate the summarized HPA file
hpa_summary <- HPA_summary_maker(HPA_data$hpa_dat)
## Run with summary, commented out so example doesn't run indefinitely
## shiny_HPASTainR(HPA_data$hpa_dat, HPA_data$cancer_dat, hpa_summary)
```

# Index

HPA\_data\_downloader, [4](#)

HPA\_summary\_maker, [5](#)

HPAStainR, [2](#)

shiny\_HPASTainR, [6](#)