hpar: The Human Protein Atlas in *R*

Laurent Gatto

Computational Proteomics Unit, University of Cambridge

October 17, 2016

Abstract

The Human Protein Atlas (HPA) is a systematic study oh the human proteome using antibody-based proteomics. Multiple tissues and cell lines are systematically assayed affinity-purified antibodies and confocal microscopy. The *hpar* package is an *R* interface to the HPA project. It distributes three data sets, provides functionality to query these and to access detailed information pages, including confocal microscopy images available on the HPA web page.

Contents

Package

hpar 1.16.0

Report issues on https://github.com/Bioconductor/hpar/issues

Ask questions on https://support.bioconductor.org/

1 Introduction

1.1 The HPA project

From the Human Protein Atlas[1] [**?**, **?**] site:

> The Swedish Human Protein Atlas project, funded by the Knut and Alice Wallenberg Foundation, has been set up to allow for a systematic exploration of the human proteome using Antibody-Based Proteomics. This is accomplished by combining high-throughput generation of affinity-purified antibodies with protein profiling in a multitude of tissues and cells assembled in tissue microarrays. Confocal microscopy analysis using human cell lines is performed for more detailed protein localisation. The program hosts the Human Protein Atlas portal with expression profiles of human proteins in tissues and cells.

[1] http://www.proteinatlas.org/

The *hpar* package provides access to HPA data from the *R*interface. It also distributes the following data sets:

`hpaNormalTissue` **Normal tissue data**: Expression profiles for proteins in human tissues based on immunohistochemisty using tissue micro arrays. The comma-separated file includes Ensembl gene identifier ("Gene"), tissue name ("Tissue"), annotated cell type ("Cell type"), expression value ("Level"), the type of annotation (annotated protein expression (APE), based on more than one antibody, or staining, based on one antibody only) ("Expression type"), and the reliability or validation of the expression value ("Reliability").

`hpaCancer` **Cancer tumor data**: Staining profiles for proteins in human tumor tissue based on immunohistochemisty using tissue micro arrays. The comma-separated file includes Ensembl gene identifier ("Gene"), tumor name ("Tumor"), staining value ("Level"), the number of patients that stain for this staining value ("Count patients"), the total amount of patients for this tumor type ("Total patients") and the type of annotation staining ("Expression type").

`rnaGeneTissue` **RNA gene data**: RNA levels in 45 cell lines and 32 tissues based on RNA-seq. The comma-separated file includes Ensembl gene identifier ("Gene"), analysed sample ("Sample"), fragments per kilobase of transcript per million fragments mapped ("Value" and "Unit"), and abundance class ("Abundance").

`rnaGeneCellLine` **RNA gene data**: RNA levels in 45 cell lines and 32 tissues based on RNA-seq. The comma-separated file includes Ensembl gene identifier ("Gene"), analysed sample ("Sample"), fragments per kilobase of transcript per million fragments mapped ("Value" and "Unit"), and abundance class ("Abundance").

`hpaSubcellularLoc` **Subcellular location data**: Subcellular localization of proteins based on immunofluorescently stained cells. The comma-separated file includes Ensembl gene identifier ("Gene"), main subcellular location of the protein ("Main location"), other locations ("Other location"), the type of annotation (annotated protein expression (APE), based on more than one antibody, or staining, based on one antibody only) ("Expression type"), and the reliability or validation of the expression value ("Reliability").

hpaSubcellularLoc14 Same as above, for version 14.

1.2 HPA data usage policy

The use of data and images from the HPA in publications and presentations is permitted provided that the following conditions are met:

- The publication and/or presentation are solely for informational and non-commercial purposes.
- The source of the data and/or image is referred to the HPA site (www.proteinatlas.org) and/or one or more of our publications are cited.

1.3 Installation

hpar is available through the Bioconductor project. Details about the package and the installation procedure can be found on its page[2]. To install using the dedicated Bioconductor infrastructure, run :

[2] http://bioconductor.org/packages/devel/bioc/html/hpar.html

```
source("http://bioconductor.org/biocLite.R")
## or, if you have already used the above before
library("BiocInstaller") ## and to install the package
biocLite("hpar")
```

After installation, *hpar* will have to be explicitly loaded with

```
library("hpar")
```

```
## This is hpar version 1.16.0,
## based on the Human Protein Atlas
##  Version:  15
##  Release data:  2016.04.11
##  Ensembl build:  78.38
## See '?hpar' or 'vignette('hpar')' for details.
```

so that all the package's functionality and data is available to the user.

2 The *hpar* package

2.1 Data sets

The data sets described above can be loaded with the `data` function, as illustrated below for `hpaNormalTissue` below. Each data set is a `data.frame` and can be easily manipulated using standard *R* functionality. The code chunk below illustrates some of its properties.

```
data(hpaNormalTissue)
dim(hpaNormalTissue)
```

```
## [1] 1159341       7
```

```
names(hpaNormalTissue)
```

```
## [1] "Gene"            "Gene.name"       "Tissue"
## [4] "Cell.type"       "Level"           "Expression.type"
## [7] "Reliability"
```

```
## Number of genes
length(unique(hpaNormalTissue$Gene))
```

```
## [1] 14578
```

```
## Number of cell types
length(unique(hpaNormalTissue$Cell.type))

## [1] 44

head(levels(hpaNormalTissue$Cell.type))

## [1] "adipocytes"                  "bile duct cells"
## [3] "cells in endometrial stroma" "cells in glomeruli"
## [5] "cells in granular layer"     "cells in molecular layer"

## Number of tissues
length(unique(hpaNormalTissue$Tissue))

## [1] 48

head(levels(hpaNormalTissue$Tissue))

## [1] "adrenal gland" "appendix"      "bone marrow"   "breast"
## [5] "bronchus"      "cerebellum"

table(hpaNormalTissue$Expression.type)

##
##      APE Staining
##   752442   406899
```

2.2 HPA interface

The package provides a interface to the HPA data. The `getHpa` allows to query the data sets described in section **??**. It takes three arguments, `id`, `hpadata` and `type`, that control the query, what data set to interrogate and how to report results respectively. The HPA data uses Ensembl gene identifiers and `id` must be a valid identifier. `hpadata` must be one of available dataset. `type` can be either `"data"` or `"details"`. The former is the default and returns a `data.frame` containing the information relevant to `id`. It is also possible to obtained detailed information, (including cell images) as web pages, directly from the HPA web page, using `"details"`.

We will illustrate this functionality with using the TSPAN6 (tetraspanin 6) gene (ENSG00000000003) as example.

```
id <- "ENSG00000000003"
head(getHpa(id, hpadata = "hpaNormalTissue"))

##              Gene Gene.name        Tissue           Cell.type
## 1 ENSG00000000003    TSPAN6 adrenal gland     glandular cells
## 2 ENSG00000000003    TSPAN6      appendix     glandular cells
## 3 ENSG00000000003    TSPAN6      appendix     lymphoid tissue
## 4 ENSG00000000003    TSPAN6   bone marrow hematopoietic cells
## 5 ENSG00000000003    TSPAN6        breast          adipocytes
```

```
## 6 ENSG00000000003    TSPAN6        breast     glandular cells
##          Level Expression.type Reliability
## 1 Not detected             APE   Uncertain
## 2       Medium             APE   Uncertain
## 3 Not detected             APE   Uncertain
## 4 Not detected             APE   Uncertain
## 5 Not detected             APE   Uncertain
## 6         High             APE   Uncertain
```

```
getHpa(id, hpadata = "hpaSubcellularLoc")
```

```
##              Gene Gene.name Main.location Other.location
## 1 ENSG00000000003    TSPAN6     Cytoplasm
##   Expression.type Reliability Main.location.GO.id
## 1             APE   Uncertain          GO:0005737
##   Other.location.GO.id
## 1
```

```
head(getHpa(id, hpadata = "rnaGeneCellLine"))
```

```
##              Gene Gene.name  Sample Value Unit Abundance
## 1 ENSG00000000003    TSPAN6   A-431  17.5 FPKM       Low
## 2 ENSG00000000003    TSPAN6    A549  27.2 FPKM    Medium
## 3 ENSG00000000003    TSPAN6  AN3-CA  32.9 FPKM    Medium
## 4 ENSG00000000003    TSPAN6    BEWO  26.0 FPKM    Medium
## 5 ENSG00000000003    TSPAN6  CACO-2  56.2 FPKM      High
## 6 ENSG00000000003    TSPAN6 CAPAN-2  30.0 FPKM    Medium
```

If we ask for `"detail"`, a browser page pointing to the relevant page is open (see figure **??**)

```
getHpa(id, type = "details")
```

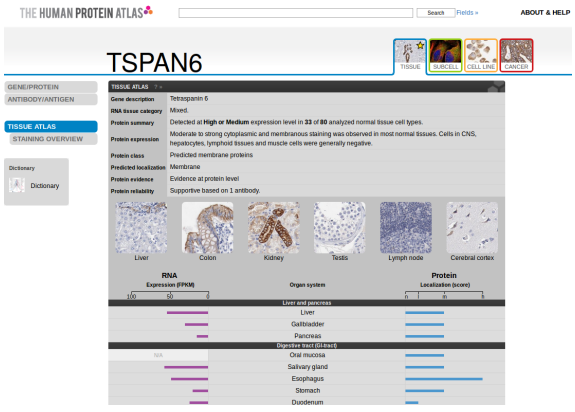

Figure 1: The HPA web page for the tetraspanin 6 gene (ENSG00000000003).

If a user is interested specifically in one data set, it is possible to set `hpadata` globally and omit it in `getHpa`. This is done by setting the *hpar* options `hpardata` with the `setHparOptions` function. The current default data set can be tested with `getHparOptions`.

```
getHparOptions()

## $hpar
## $hpar$hpadata
## [1] "hpaNormalTissue"

setHparOptions(hpadata = "hpaSubcellularLoc")
getHparOptions()

## $hpar
## $hpar$hpadata
## [1] "hpaSubcellularLoc"

getHpa(id)

##              Gene Gene.name Main.location Other.location
## 1 ENSG00000000003    TSPAN6     Cytoplasm
##   Expression.type Reliability Main.location.GO.id
## 1             APE   Uncertain          GO:0005737
##   Other.location.GO.id
## 1
```

2.3 HPA release information

Information about the HPA release used to build the installed *hpar* package can be accessed with `getHpaVersion`, `getHpaDate` and `getHpaEnsembl`. Full release details can be found on the HPA release history[3] page.

[3] http://www.proteinatlas.org/about/releas

```
getHpaVersion()

## version
##    "15"

getHpaDate()

##         date
## "2016.04.11"

getHpaEnsembl()

## ensembl
## "78.38"
```

3 A small use case

Let's compare the subcellular localisation annotation obtained from the HPA subcellular location data set and the information available in the Bioconductor annotation packages.

```
id <- "ENSG00000001460"
getHpa(id, "hpaSubcellularLoc")
```

```
##              Gene Gene.name Main.location   Other.location
## 8 ENSG00000001460     STPG1       Nucleus Nuclear membrane
##   Expression.type Reliability Main.location.GO.id
## 8             APE  Supportive          GO:0005634
##   Other.location.GO.id
## 8           GO:0031965
```

Below, we first extract all cellular component GO terms available for ENSG00000001460 from the *org.Hs.eg.db* human annotation and then retrieve their term definitions using the *GO.db* database.

```
library("org.Hs.eg.db")
library("GO.db")
ans <- select(org.Hs.eg.db, keys = id,
              columns = c("ENSEMBL", "GO", "ONTOLOGY"),
              keytype = "ENSEMBL")
```

```
## 'select()' returned 1:many mapping between keys and columns
```

```
ans <- ans[ans$ONTOLOGY == "CC", ]
ans
```

```
##           ENSEMBL         GO EVIDENCE ONTOLOGY
## 1 ENSG00000001460 GO:0005634      IEA       CC
## 2 ENSG00000001460 GO:0005737      IEA       CC
```

```
sapply(as.list(GOTERM[ans$GO]), slot, "Term")
```

```
##  GO:0005634  GO:0005737
##   "nucleus" "cytoplasm"
```

Session information

- R version 3.3.1 (2016-06-21), `x86_64-apple-darwin13.4.0`
- Locale: `C/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8`
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, stats4, utils

- Other packages: AnnotationDbi 1.36.0, Biobase 2.34.0, BiocGenerics 0.20.0, GO.db 3.4.0, IRanges 2.8.0, S4Vectors 0.12.0, hpar 1.16.0, org.Hs.eg.db 3.4.0
- Loaded via a namespace (and not attached): BiocStyle 2.2.0, DBI 0.5-1, RSQLite 1.0.0, evaluate 0.10, formatR 1.4, highr 0.6, knitr 1.14, magrittr 1.5, stringi 1.1.2, stringr 1.1.0, tools 3.3.1