SOFTWARE TOOL ARTICLE

REVISED HPAStainR: a Bioconductor and Shiny app to query protein expression patterns in the Human Protein Atlas [version 2; peer review: 3 approved]

Tim O. Nieuwenhuis[1,2], Marc K. Halushka[1]

[1]Department of Pathology, Johns Hopkins University School of Medicine Baltimore, Baltimore, MD, 21205, USA
[2]McKusick-Nathans Institute, Department of Genetic Medicine, Johns Hopkins University School of Medicine, Baltimore, MD, 21205, USA

v2 **First published:** 08 Oct 2020, **9**:1210
https://doi.org/10.12688/f1000research.26771.1
Latest published: 22 Mar 2021, **9**:1210
https://doi.org/10.12688/f1000research.26771.2

Abstract

The Human Protein Atlas is a website of protein expression in human tissues. It is an excellent resource of tissue and cell type protein localization, but only allows the query of a single protein at a time. We introduce HPAStainR as a new Shiny app and Bioconductor/R package used to query the scored staining patterns in the Human Protein Atlas with multiple proteins/genes of interest. This allows the user to determine if an experimentally-generated protein/gene list associates with a particular cell type. We validated the tool using the Panglao Database cell type specific marker genes and a Genotype Expression (GTEx) tissue deconvolution dataset. HPAStainR identified 92% of the Panglao cell types in the top quartile of confidence scores limited to tissue type of origin results. It also appropriately identified the correct cell types from the GTEx dataset. HPAStainR fills a gap in available bioinformatics tools to identify cell type protein expression patterns and can assist in establishing ground truths and exploratory analysis. HPAStainR is available from: https://32tim32.shinyapps.io/HPAStainR/

Keywords

protein staining, Human Protein Atlas, marker genes, marker proteins, exploratory analysis

This article is included in the RPackage gateway.

Open Peer Review

Reviewer Status ✓ ✓ ✓

	Invited Reviewers		
	1	**2**	**3**
version 2 (revision) 22 Mar 2021	✓ report	✓ report	✓ report
version 1 08 Oct 2020	✓ report	? report	? report

1. **Mazdak Salavati**, University of Edinburgh, Edinburgh, UK
2. **Laurent Gatto**, UCLouvain, Brussels, Belgium
3. **Yasin Kaymaz**, Harvard University, Boston, USA

Any reports and responses or comments on the article can be found at the end of the article.

This article is included in the Bioconductor gateway.

Corresponding author: Tim O. Nieuwenhuis (tnieuwe1@jhmi.edu)

Author roles: Nieuwenhuis TO: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Software, Visualization, Writing – Original Draft Preparation; **Halushka MK**: Conceptualization, Funding Acquisition, Project Administration, Resources, Supervision, Visualization, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: This work was supported by the National Heart, Lung, and Blood Institute [1R01HL137811]; and the National Institutes of Health [R01GM130564, T32GM07814].

How to cite this article: Nieuwenhuis TO and Halushka MK. **HPAStainR: a Bioconductor and Shiny app to query protein expression patterns in the Human Protein Atlas [version 2; peer review: 3 approved]** F1000Research 2021, **9**:1210 https://doi.org/10.12688/f1000research.26771.2

First published: 08 Oct 2020, **9**:1210 https://doi.org/10.12688/f1000research.26771.1

REVISED **Amendments from Version 1**

After carefully reading the comments from our reviewers we have made several changes to the HPAStainR package and paper briefly described here. In the package, `HPA_data_downloader()` has been updated with further functionality and arguments allowing individuals to not only date the downloaded data but use data from a specific date of download. In the vignette, we also now suggest the usage of the Bioconductor package `hpar` for its benefit in version control when paired with HPAStainR. The main function, `HPAStainR()` has also been updated, replacing the default Chi-Square analysis with a Fisher's Exact test, as the data is non-parametric and the Fisher's exact test results in more reproducibility when compared to the permuted p-values from the Chi-Square test. The p-value output has also been changed to be numeric instead of a character. `shiny_HPAStainR()` has also been edited to reflect these changes.

The paper also received revisions based on reviewer comments and changes in the package. Due to using Fisher's Exact Test, all p-values have been changed to be the result of this new method including tables and figures. We have cited other important Bioconductor packages relevant to the Human Protein Atlas, such as `hpar` and `HPAnalyze`. Table 1 has two new columns showing the p-values of the HPAStainR results. We described how the "stringency" is based on the "Reliability" column from HPA's data describing how certain HPA is of the staining result. To better explore the arbitrary staining score we generated staining score distributions for random gene groupings with an n ranging from 10 to 100 genes and added them as extended figures. Lastly, we have added further explanations of the figures in their legends for further clarification.

Any further responses from the reviewers can be found at the end of the article

Introduction

The Human Protein Atlas (HPA) has performed immunohistochemistry-based visual proteomics for over 15,313 proteins across 59 tissues. Within each tissue a number of different cell types have been scored for staining patterns by a group of pathologists. Therefore, there is a great amount of visual proteomic data that can be used to classify gene or protein lists into specific cell types[1–3]. Their website is designed to query one protein of interest at a time and there is no option to query a list of proteins to determine if that protein set is enriched in a particular cell type. This would be a useful feature to take advantage of this robust dataset. Other gene list tools such as Enrichr, which query multiple databases for associations, have not incorporated the HPA protein cell expression dataset into their tools[4]. There are other R packages used to incorporate and query HPA data such as hpar[5], which allows for easy loading and querying of version controlled data and HPAnalyze[6] which has powerful visualization tools for protein levels. However, both packages lack the functionality of determining enriched proteins in the database.

We introduce HPAStainR (https://32tim32.shinyapps.io/HPAStainR/), a Bioconductor R package and Shiny app developed to query the cell staining database of the HPA. HPAStainR allows a user to input a list of proteins/genes and returns a rank ordered list of cell types that are stained for the input list. HPAStainR is customizable, allowing the user the ability to include cancer or normal tissue data, change the HPA confidence levels, toggle the identification of what proteins from the list were detected, generate a p-value for how many cell type specific proteins are counted for a given cell type, and allow the downloading of the output as a comma separated (csv) file.

Methods

Implementation

The user interface of Shiny HPAStainR is made of a sidebar where one can input their protein/gene list, along with various options to customize the output of the Shiny app. The main panel consists of two tabs. The first tab is the output tab, where the DataTable from the user's query is output. The second tab is informational giving the user a list of HPA cell types and how many proteins were tested/histologically scored in a given cell type.

The HPAStainR package is available on Bioconductor. The package shares all of the same functionality as the Shiny web application, including the ability to run the Shiny app locally and acquire all of the data to do so. This allows HPAStainR to be used as the Shiny app or incorporated into a local R pipeline.

Operation

HPAStainR is an online Shiny app[7], available at http://shinyapps.io, and as a Bioconductor R Package (https://bioconductor.org/)[8] with its source code available on GitHub (https://github.com/tnieuwe/HPAStainR). The function has been tested on R version 3.6.1 and later. Minimal requirements are the same as RStudio's system requirements [https://bit.ly/2UqwXc6].

Installation: The installation of the HPAStainR package can be done in R using the following commands:

> if (!requireNamespace("BiocManager", quietly = TRUE))

> install.packages("BiocManager")`

> BiocManager::install("HPAStainR")`

The remote-Shiny web application can be accessed via the following link:

https://32tim32.shinyapps.io/HPAStainR/

Note: This analysis uses HPAStainR v.1.1.4 which is available on the HPAStainR Github using:

> install.packages("devtools")

> install_github("tnieuwe/HPAStainR")

And through the devel version of Bioconductor:

> if (!requireNamespace("BiocManager", quietly = TRUE))

> install.packages("BiocManager")

```
> BiocManager::install(version = 'devel ')

> BiocManager::install("HPAStainR")
```

The next Bioconductor version (3.13) is expected in April or May of 2021 and at that time HPAStainR v.1.2.0 will be released with all the changes in the devel. At that time the HPAStainR shiny app will be updated as well.

Input: There are three required R objects for the main HPAStainR function to work and one optional data frame. The first two required objects are the public staining files from the HPA, which can be downloaded using the package and the `HPA_data_downloader()` function. If the argument `save_file` in `HPA_data_downloader()`is set to `TRUE` then the file will be dated and downloaded, and following runs will load the saved file. The third required input is either a vector of proteins or genes or a character list of proteins separated by a space, comma, or newline to be queried in HPAStainR. The optional data frame, used in the Shiny app version of HPAStainR, is a table that contains the percent of proteins that stained the tissue compared to the number of the proteins evaluated in the tissue, represented in Extended Table 1[9], which can be generated using the `hpa_summary_maker()` function. This table demonstrates that not all cell types/tissues have the same number of proteins stained for.

Output: If using `HPAStainR()`, a tibble containing the summarized detection of the input list of proteins or genes for each available cell type customized by the options selected before running the analysis. If using `shiny_HPAStainR()`, a shiny Datatable containing the data previously mentioned in the base `HPAStaiR()` output.

HPA data distribution

Our analysis in this paper uses the 19.3 version of the Human Protein Atlas Data. Staining was scored by cell type in each tissue by a group of pathologists who rated the intensity in each evaluated cell type as "high, medium, low or not detected." Not all cell types in all tissues were scored, nor were all cell types consistently evaluated. As a result, there are some caveats in the HPA data that should be noted. The distribution of how many proteins are histologically scored in each of the 137 cell types varies in HPA, such that not all results are equal. The number of proteins scored in cell types ranges from 1 in four substantia nigra cell types to over 17,000 in endometrial glandular cells (Figure 1A; Extended Table 1[9]), impacting how often a protein is detected in a given cell type (Figure 1B). The percent of stained to scored proteins demonstrates an enrichment at both extremes of the distribution (Figure 1C). To highlight this discrepancy in testing, we have made the information in Extended Table 1 as an available tab on the Shiny app. This data can also be made using the `HPA_summary_maker()` function on normal tissue.

Staining score calculation

The staining score calculation is an arbitrary measure of how well an input list of proteins are enriched for a particular cell type. A formal equation is below, but briefly, it is calculated based on the frequency and intensity of staining within a given cell type. Staining intensity is a percentage of high, medium, low, and not detected counts. The high percentage is multiplied by a value of 100, the medium percentage by 50, and the low percentage by 25, before adding all the results together to generate the final staining score. While arbitrary, we over-weighted high staining as the IHC was more robust and those proteins may better define the cell type. To illustrate the distribution of the staining score we generated 1,000 HPAStainR results on random genes, including the top 10 results from HPAStainR and all results, for random gene lists of sizes 10, 25, 50, and 100. As the number of genes increases the staining score distribution decreases. For the top 10 results from each run, ordered on staining score, the distribution appears to be normal (Extended Figure 1). Analysis of all staining data suggests a right skew (Extended Figure 2).

The model for the staining score equation is below where t is the total number of proteins from the list tested in the cell type, h is the number of proteins with high staining in the cell type, m is the number of proteins with medium staining in the cell type, and l is the number of proteins with low staining in the cell type.

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

Confidence score calculation

The confidence score is unique to this paper, and only used for the comparison of the Panglao Database (PanglaoDB) cell types to HPA cell types. It is a modified version of the staining score adjusting for size of the protein list for each cell type from PanglaoDB. The confidence score calculation weights PanglaoDB cell types based on how many marker genes they have. Like the staining score, this score ranges from 0–100. The model for the equation is below, where p is the number of proteins tested, with a max p being 50 (standardizing the score range), and the staining score of the protein list in the cell type is represented by s.

$$\text{Confidence Score} = \frac{p \times s}{50}$$

Cell type enriched p-value

While we utilized all expressed proteins in our staining score, we recognize that some proteins demonstrate cell type enrichment. For this analysis we generated the "enriched-protein p-value" based on either a Fisher's Exact Test or χ^2 analysis. In this paper we used the results of the Fisher's Exact Test.

To calculate the enriched-protein p-value we generated a list of cell type enriched proteins for each level of stringency, low, normal and high. The stringency parameter filters the `Reliability` column from the normal tissue HPA dataset. This `Reliability` varies from "Enhanced," "Supported," "Approved," to "Uncertain" in decreasing order of certainty (full descriptions of these labels are found here https://www.proteinatlas.org/about/assays+annotation). Low stringency includes all data, normal stringency includes "Enhanced," "Supported," and "Approved," while high stringency only includes "Enhanced" and "Supported". The cell type enriched protein list was generated by calculating a

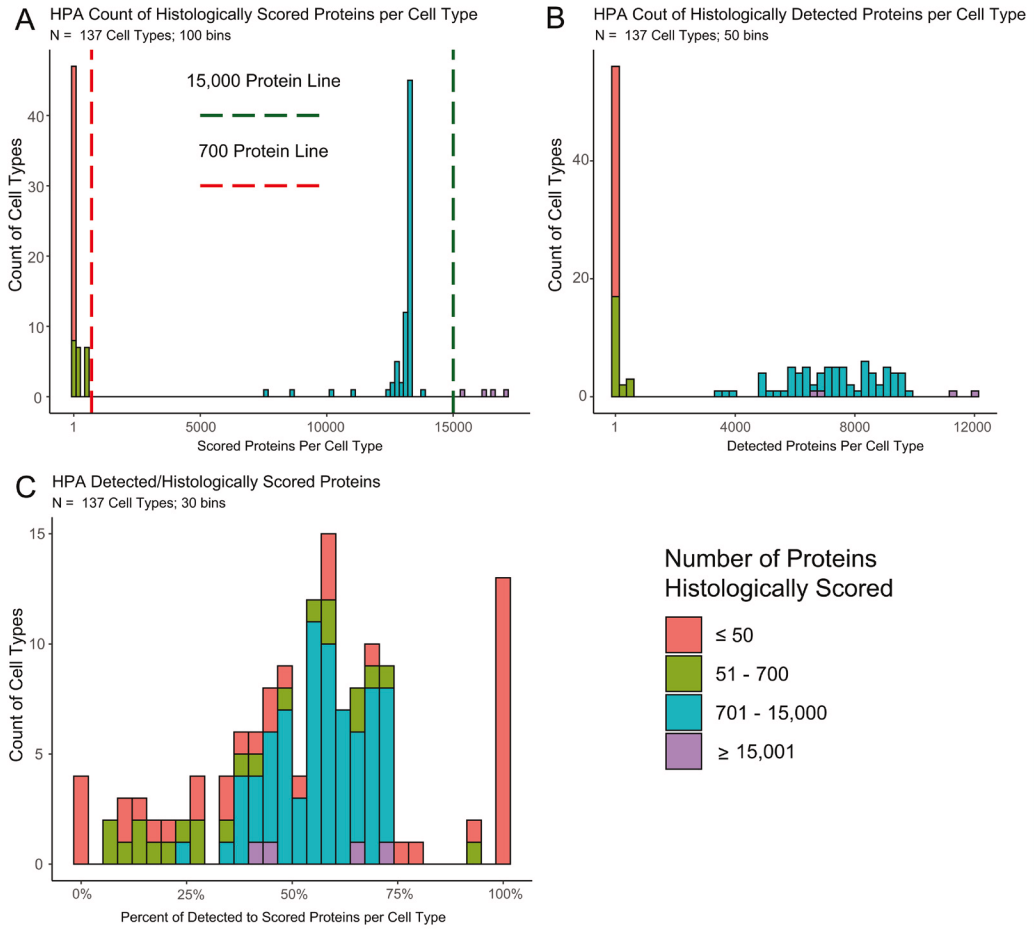

Figure 1. Histograms comparing how often proteins stain to how often they are evaluated in HPA. A. A histogram of the 137 cell types showing the amount of proteins histologically scored in each cell type. Four cell types were evaluated for >15,000 proteins (green line) and 61 for <700 proteins (red line). This bimodal distribution shows that there are nearly separate groups of cell types based on how often they are scored. **B**. A histogram of the 137 cell types on the amount of proteins that had positive staining in each cell type. This distribution reveals an expected skew, cell types that are more often histologically scored, tend to have a higher count of proteins detected. **C**. A histogram of the percent of positively stained proteins to the total amount of histologically scored proteins for the given cell type. The extreme ends of the distribution are populated by samples with less than 700 scored proteins revealing that seldom scored cell types staining frequencies are often artefacts.

percentage of positively stained to evaluated proteins across each cell type to adjust for protein scoring frequency. This percentage generated our 'enriched proteins' list from the top quartile of enriched proteins (the proteins present in <25% of the evaluated cell types. The number of proteins were 3,275, 2,543, and 1,235 for low, normal, and high stringency respectively and 3,818 in cancer) (Extended Tables 2 and 3[9]; Figure 2 and Figure 3). The Fisher's Exact Test analysis was based on the staining presence/absence of 'enriched proteins' for a given HPA cell vs presence/absence of proteins from a protein list query. For all experiments in this paper, stringency was set to normal.

All code for the package and the analysis can be found on GitHub at https://github.com/tnieuwe/HPAStainR and https://github.com/tnieuwe/HPAstainR_dev_paper, respectively.

Use cases

HPA functionality

HPAStainR uses the publicly available HPA cell type histologically scored staining data to identify the top cell type matches to a queried protein/gene list. It ranks cell types on a 0 to 100 "staining score" (Figure 4). This score is based on the pathologist annotated staining intensity (high, medium, low) of each protein/gene in the query list for each HPA cell type, as a percent of the total number of proteins/genes queried (see Methods). For example, a query of the pancreatic enzymes PRSS1, PNLIP, and CELA3A, along with the protein PRL, would identify "pancreas exocrine glandular cells" as the top hit with a staining score of 75 due to the high staining intensity in three proteins and negative staining of the fourth protein. The second hit would be the "pituitary gland cells in anterior" due to PRL's high expression in that cell type

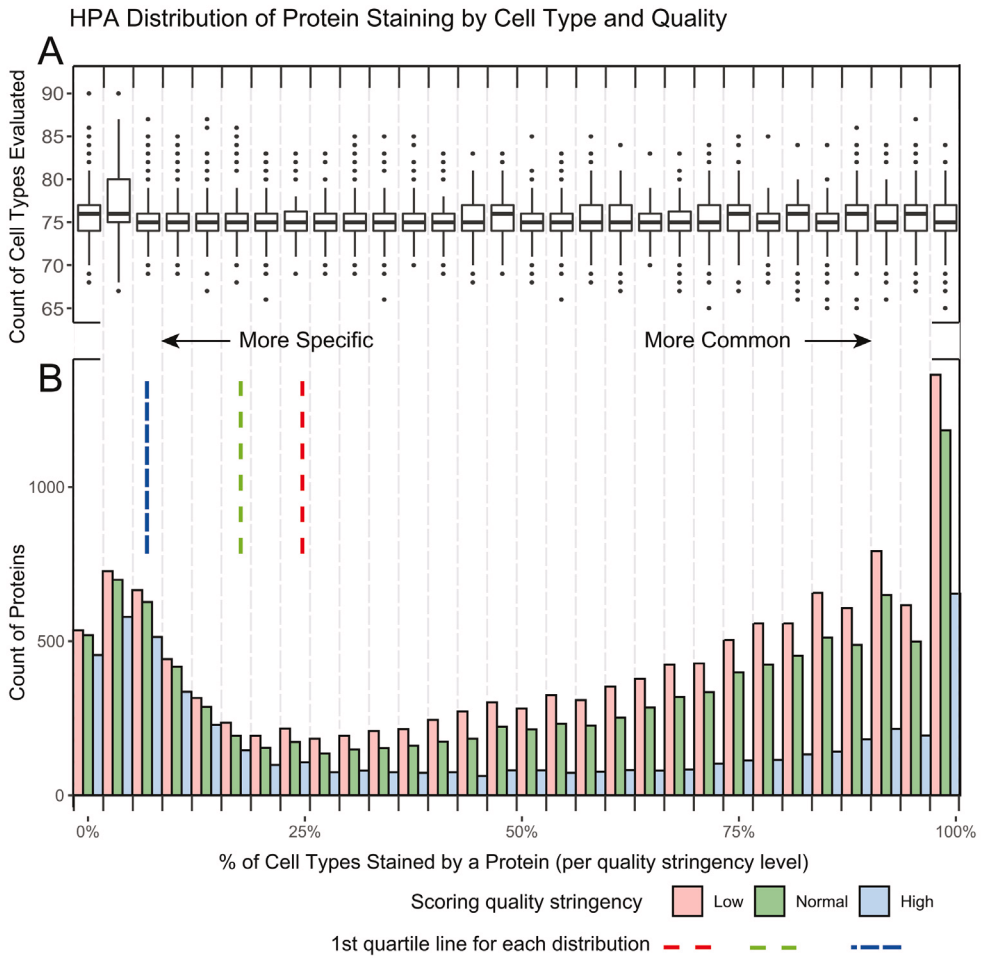

Figure 2. HPA distribution of protein staining by cell type and quality. A. A boxplot of the number of cell types evaluated per protein in each of the 30 histogram bins. The overall median of cell types is 75 proteins with the 1st and 3rd quartile being 74 and 77 proteins respectively. This illustrates that tissue enrichment of a protein is not an artifact of how often a protein is scored, as there is a similar level of testing across bins. **B**. A histogram demonstrating the percent of positive staining cells per protein. Three quality stringencies are given. The 1st quartile lines demonstrate the specificity cut off of the distribution used at each stringency level. As expected, the number of proteins in the 1st quartile increase with lower stringency. More commonly detected genes are more greatly affected by stringency when compared to rarely detected genes. There also appear to be a larger number of specific proteins (10%) than there are of semi-specific proteins (25%) in HPA, as the distribution from common to specific decreases before peaking again.

(score of 25), followed by "intestinal glandular cells" which only have medium staining of PRSS1 (score of 12.5).

The Panglao Database

To show the functionality of the Shiny app we applied HPAStainR to the Panglao Database, a hub of community-curated cell type markers from single cell data[10]. We wanted to investigate how well HPAStainR would mark the cell types based on PanglaoDB's annotations. We downloaded a tsv file of PanglaoDB's cell type gene marker data, and parsed it down to only human protein coding marker genes. We assayed 146 human cell types and their 3,661 marker genes through HPAStainR. The number of marker genes per cell type in PanglaoDB are variable, ranging from one marker in trophoblast stem cells to 216 in interneurons. A histogram of markers per cell type showing the distribution can be found in Figure 5.

HPAStainR identified many of the cell types in PanglaoDB

To perform analyses between multiple runs of HPAStainR and PanglaoDB, we generated a "confidence score," a value (theoretical 0–100) that corrected for the staining score's determination using an additional feature of how many proteins were evaluated (see Methods). This score weighted cell types with multiple marker proteins staining over cell types with a single or fewer marker proteins. Thus, the confidence score allowed us to rank the cell types based on both staining and depth of data.

HPAStainR is agnostic to the source of a protein/gene list. Therefore, an identification of equivalent cell types across two methods provides strong evidence of HPAStainR's usefulness. Specific protein lists, corresponding to the 146 cell types were evaluated from PanglaoDB in HPAStainR with the

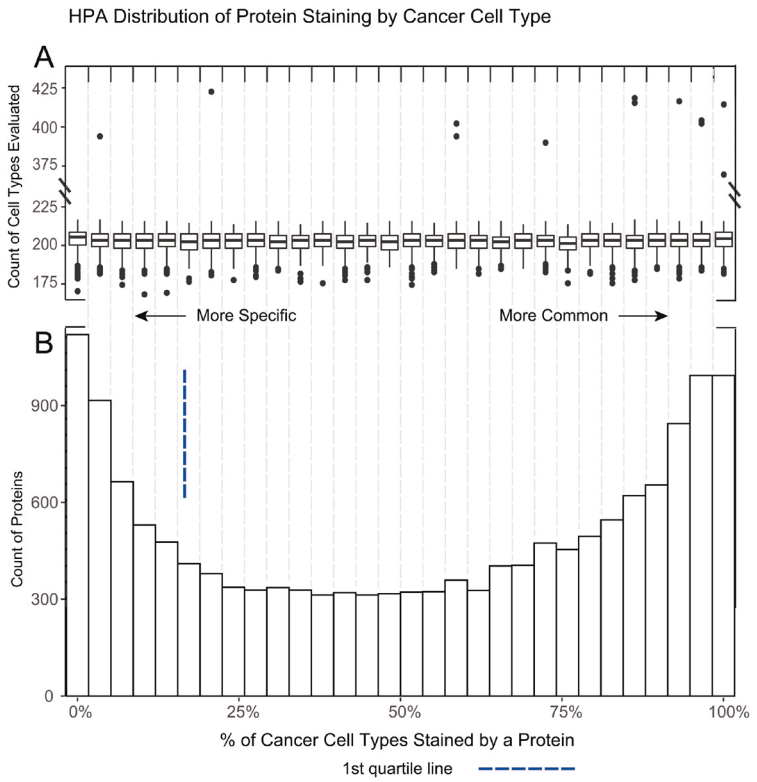

Figure 3. HPA distribution of protein staining by cancer cell type. A. A boxplot of the number of cell types evaluated per protein in each of the 30 histogram bins. The overall median of cancer cell types is 203 with the 1st and 3rd quartile of 199 and 207 respectively. A number of outliers with ~2x more cell types evaluated are noted. Besides frequently tested cancers, the distribution reveals proteins are evenly scored across cancers, regardless of how frequently they are tested. **B**. A histogram demonstrating the percent of positive staining cancer cells per protein. This distribution is different from normal tissue as some cancer samples of the same cancer type can positively stain for a protein while other samples will not. Similar to normal tissue there seems to be an increase of proteins in the extreme ends of the specificity distribution.

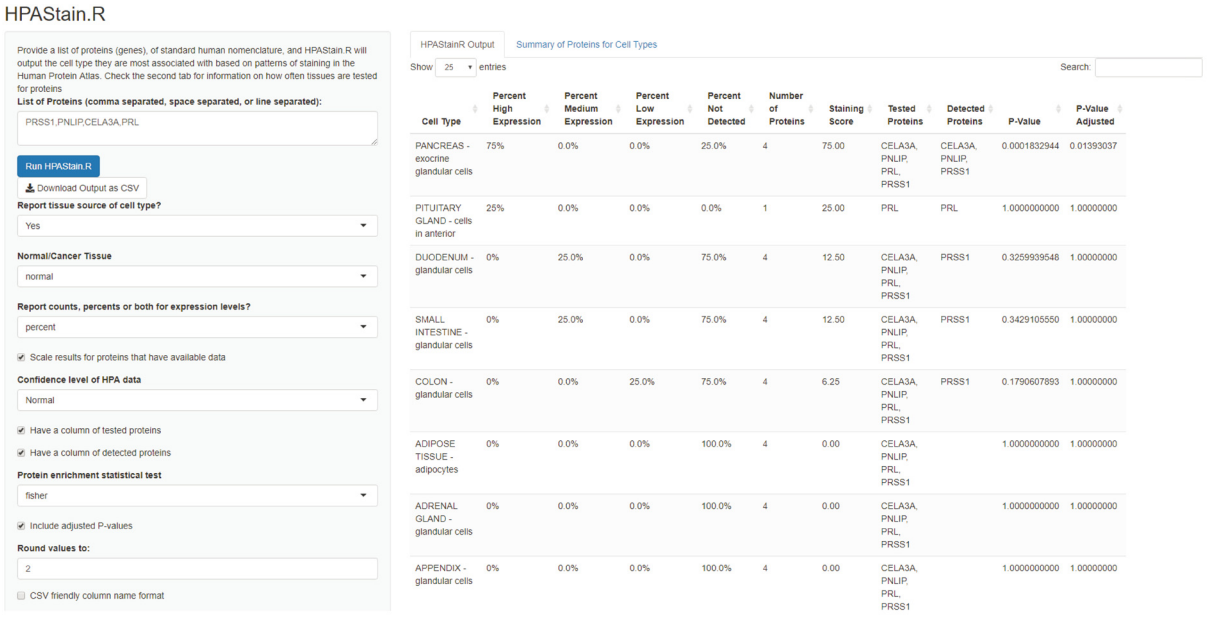

Figure 4. A screen shot of the user interface of HPAStainR. A list of comma, space, or line separated proteins or genes are inputted on the left column. Multiple customizations are available for users below to optimize the search parameters for their query of interest. Data is outputted to the right, and further information about the cell types and how many proteins were histologically scored per cell type are available as a second tab.

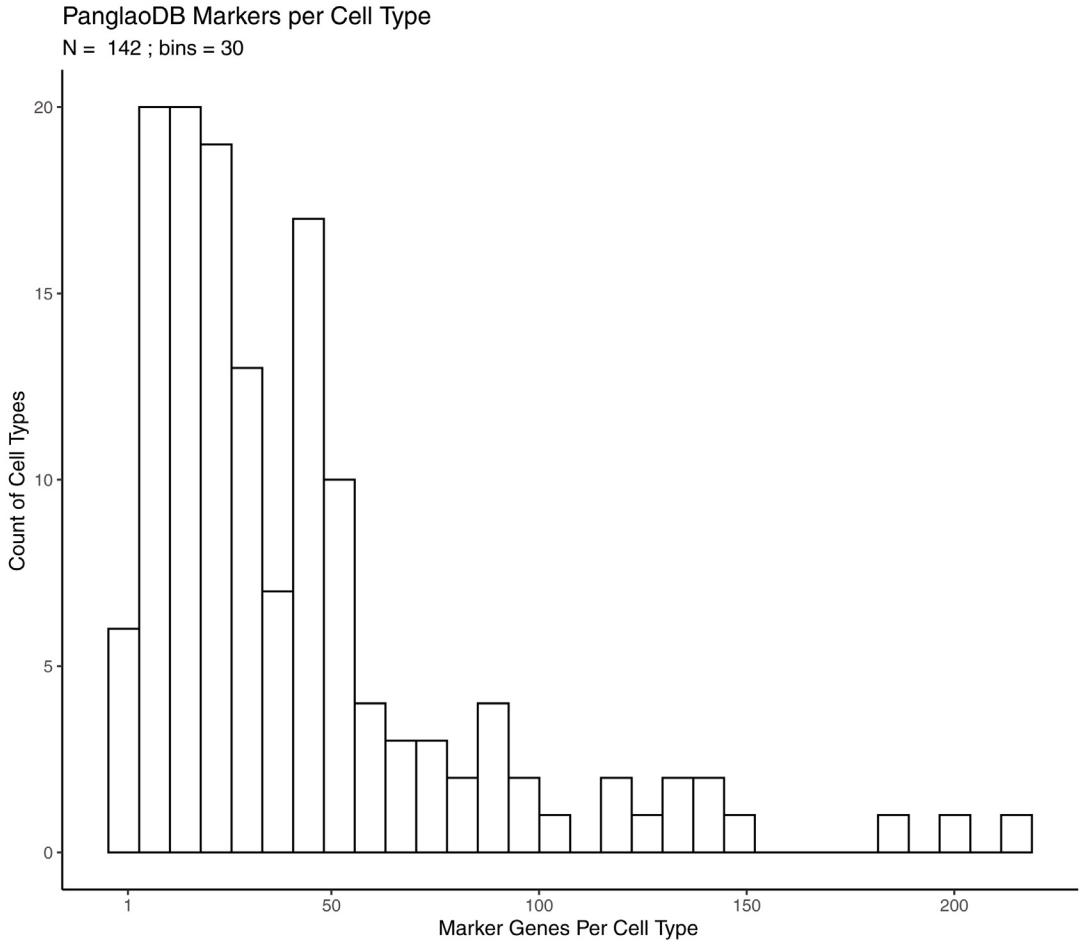

Figure 5. PanglaoDB markers per cell type. Histogram of the number of marker genes used to define 142 different Panglao cell types.

top HPA cell types identified for each. To cover both potential user needs, we included in our PanglaoDB output both the top result of HPAStainR and the top result in the appropriate tissue. The confidence score across these comparisons, generated on HPAStainR data, ranged from 1.5 to 66.75. The results of the 146 cell types were divided into quartiles (Qs) based on the confidence score. The average number of proteins associated with a PanglaoDB cell type used to identify the top HPA cell type strongly correlated with the quartile (76.3; 55.7; 23.9; 7.7 proteins in Q1 to Q4, respectively). In the top quartile of scores, 75% (27/36) of cells matched between PanglaoDB and HPA. Of the nine that were not a perfect match, six matched the top hit when limited by tissue type. Of the remaining Q1 PanglaoDB cell types; liver kupffer cells (a type of macrophage), mesothelial cells, and embryonic stem cells, none had matching cell types in the HPA[11].

A subset of this analysis can be seen below in Table 1 with the full results being in Extended Table 4[9]. Results were ranked by confidence score, with a strong correlation of higher confidence scores to more accurate cell type assignments between PanglaoDB and HPA. An interesting example are chondrocytes, where the top stained score (27.25) was to TONSIL squamous epithelial cells and the top tissue specific cell type was SOFT TISSUE - chondrocytes (20.75). In addition to the stain score, HPAStainR provides a p-value (and Holm adjusted p-value) based on a separate metric based on cell type specific/enriched protein expression (see Methods). Although tonsil squamous epithelial cells is the top HPAStainR result, the adjusted enriched protein p-value was 1.0 (nonsignificant) while it was p=2.5E-04 for the chondrocytes, indicating cell-enriched proteins favored the correct match.

HPAStainR can help determine cell type populations in bulk RNA sequencing

We then demonstrated the functionality of HPAStainR in bulk datasets. We utilized the variable gene expression data from the Genotype Expression (GTEx) dataset that we had previously uncovered as being driven by variation in pneumocytes or the presence of bronchial epithelium[12]. There were 33 genes identified in the pneumocyte cluster and 70 genes in the bronchial epithelium cluster. HPAStainR was applied

Table 1. A subset of 10 HPAStainR results of PanglaoDB cell type marker queries. Both the overall top HPAStainR result and a tissue-specific result is given. The "Select Tissues" results are from a search performed for the PanglaoDB cell type only within the matched tissue type (ALL CAPITALIZED) in HPA.

PanglaoDB Cell Type	Confidence Score	Top HPAStainR Result	Top Result Stained Score	Top Adjusted P-value	Select Tissue Top Result	Select Tissue Top Stained Score	Select Tissue Adjusted P-value
KIDNEY proximal tubule cells	66.75	KIDNEY - cells in tubules	66.75	1.59E-18	KIDNEY - cells in tubules	66.75	1.59E-18
HEART MUSCLE cardiomyocytes	61	HEART MUSCLE - myocytes	61	1.29E-33	HEART MUSCLE - myocytes	61	1.29E-33
IMMUNE SYSTEM neutrophils	60	BONE MARROW - hematopoietic cells	60	9.99E-13	BONE MARROW - hematopoietic cells	60	9.99E-13
OLFACTORY SYSTEM olfactory epithelial cells	51.25	BRONCHUS - respiratory epithelial cells	51.25	2.17E-05	NASOPHARYNX - respiratory epithelial cells	39.75	0.00076
CONNECTIVE TISSUE adipocytes	33.75	KIDNEY - cells in tubules	33.75	1	ADIPOSE TISSUE - adipocytes	24.25	0.02353
CONNECTIVE TISSUE chondrocytes	27.25	TONSIL - squamous epithelial cells	27.25	1	SOFT TISSUE - chondrocytes	20.75	0.00025
REPRODUCTIVE granulosa cells	13.6	PLACENTA - trophoblastic cells	42.5	0.00174	OVARY - follicle cells	25	1
HEART MUSCLE purkinje fiber cells	5.775	CAUDATE - neuronal cells	57.75	1	HEART MUSCLE - myocytes	0	1
BRAIN cholinergic neurons	5.25	DUODENUM - glandular cells	37.5	1	CEREBRAL CORTEX - neuronal cells	31.25	1
EMBRYO trophoblast progenitor cells	3	PLACENTA - trophoblastic cells	50	1	tissue not found	NA	NA

separately to both lists and found the top results to be lung pneumocytes and bronchus respiratory epithelial cells respectively (Figure 6A and 6B; Extended Tables 5 and 6[9]). Therefore, across both single cell and bulk gene expression data, we have identified useful functionality to HPAStainR.

Conclusion

HPAStainR fills a small gap in our knowledge base by allowing for the query of gene/protein lists against the cellular protein expression pattern data of HPA. As datasets of single cell RNA sequencing analysis become available, it is useful to have a tool to correlate these individual cellular transcriptomic gene profiles with translated protein expression patterns. We have also shown the tool can recapitulate bulk RNA sequencing findings making it a valuable tool to understand the cellular composition of a sample. The HPA is an excellent resource to observe staining patterns within cells across tissues for proteins of interest. The limitations of the study are the quality of the staining across all HPA tissues and the quality/consistency of the pathology scoring of the tissues[13]. Both of these may impact the scoring achieved for any given query. HPAStainR is a new valuable resource to accelerate exploratory and ground truth queries in the HPA cell type protein staining data.

Data availability

Underlying data

The data from PanglaoDB was downloaded at https://panglaodb.se/markers/PanglaoDB_markers_27_Mar_2020.tsv.gz (last updated March 27th 2020).

Human Protein Atlas normal tissue and cancer tissue data was acquired from the website: https://www.proteinatlas.org/about/download (last visited March 28th 2020)

Extended data

Harvard Dataverse: HPAStainR – A Bioconductor and shiny app to query protein expression patterns in the Human Protein Atlas, https://doi.org/10.7910/DVN/CL5ZTA[9].

A

HPAStainR Output | Summary of Proteins for Cell Types

Show 25 entries Search:

Cell Type	Percent High Expression	Percent Medium Expression	Percent Low Expression	Percent Not Detected	Number of Proteins	Staining Score	Tested Proteins	Detected Proteins	P-Value	P-Value Adjusted
LUNG - pneumocytes	17.0%	13.0%	17.0%	52.0%	23	27.75	ALOX15B, C4BPA, CPB2, CRTAC1, DMBT1, FGA, FGFBP1, FGG, FUT3, HKDC1, LAMP3, LRP2, PCSK2, PGC, PIGR, PLA2G3, PLA2G4F, SFTPA1, SFTPA2, SFTPC, SLC26A9, SLC6A14, SLC6A20	ALOX15B, CRTAC1, DMBT1, LAMP3, PIGR, PLA2G3, SFTPA1, SFTPA2, SFTPC, SLC6A14, SLC6A20	7.207277e-08	5.909967e-06

B

HPAStainR Output | Summary of Proteins for Cell Types

Show 25 entries Search:

Cell Type	Percent High Expression	Percent Medium Expression	Percent Low Expression	Percent Not Detected	Number of Proteins	Staining Score	Tested Proteins	Detected Proteins	P-Value	P-Value Adjusted
BRONCHUS - respiratory epithelial cells	56.0%	32.0%	6.0%	6.0%	34	73.50	ADGB, ALOX15, ARMC3, BPIFA1, BPIFB1, CAPSL, CCDC114, CDHR4, CHST9, DNAAF1, DNAH12, DNAH9, DNAI1, DNAI2, DYDC2, EFCAB1, ERN2, FAM183A, FAM92B, GSTA1, MS4A8, MUC16, MUC5B, RSPH1, SCGB1A1, SNTN, SPAG6, STMND1, STOML3, TEKT1, TEKT2, TEKT4, VWA3B, ZBBX	ADGB, ALOX15, ARMC3, BPIFB1, CAPSL, CCDC114, CDHR4, CHST9, DNAAF1, DNAH12, DNAH9, DNAI1, DNAI2, DYDC2, EFCAB1, FAM183A, FAM92B, GSTA1, MS4A8, MUC16, MUC5B, RSPH1, SCGB1A1, SNTN, SPAG6, STMND1, STOML3, TEKT1, TEKT2, TEKT4, VWA3B, ZBBX	2.141124e-23	1.584432e-21

Figure 6. Top HPAStainR results from high variance lung clusters. A. Represents the output when a list of genes associated with pneumocytes are used as input. **B**. Represents the output when a list of genes associated with the bronchial epithelium are used as input.

This project contains the following extended data:

- **Extended Figure 1. The distribution of the top 10 results from 1,000 permutations of HPAStainR. A**. A histogram of the top 10 results selecting 10 genes in each permutation. **B**. A histogram of the top 10 results selecting 25 genes in each permutation. **A**. A histogram of the top 10 results selecting 50 genes in each permutation. **A**. A histogram of the top 10 results selecting 100 genes in each permutation.
- **Extended Figure 2. The distribution of all results from 1,000 permutations of HPAStainR**. **A**. A histogram of all results selecting 10 genes in each permutation. **B**. A histogram of all results selecting 25 genes in each permutation. **A**. A histogram of all results selecting 50 genes in each permutation. **A**. A histogram of all results selecting 100 genes in each permutation.
- **Extended Table 1**. **The number of proteins evaluated and positively stained for each cell type.** For

each tissue cell combination the number of proteins being positively scored over the number of times proteins are evaluated. These are categorically grouped on the amount of proteins evaluated.

- **Extended Table 2. The rarity for proteins in normal tissue across different filters.** For each gene in normal tissue that was detected, the percent of how often proteins stained compared to how often they were histologically scored based on three quality filters, low, normal and high. Each protein is also labeled if it is considered rare or not in a given tissue based on if its percentage was in the bottom 1st quartile of the distribution for each quality filter. NA indicates that the protein never reached the threshold to be counted in a given filter level. Proteins that never positively stained were not included.
- **Extended Table 3. The rarity of proteins in cancer samples.** The percent positive staining of 15,301 in cancer cells. The quartile of proteins with the lowest values were indicated as rare. Note, there is no quality filter for cancer thus different cancer samples from the same type of cancer can have different staining patterns.
- **Extended Table 4. The extended HPAStainR output from Table 1.**
- **Extended Table 5. HPAStainR output of cluster A from McCall *et al*.** The full results of HPAStainR when running cluster A of McCall *et al*. through the package. Pneumocytes were expected and observed.
- **Extended Table 6. HPAStainR output of cluster B from McCall *et al*.** The full results of HPAStainR when running cluster B of McCall *et al*. through the package. Bronchial epithelial cells were expected and observed.

Software availability

Software available from: https://32tim32.shinyapps.io/HPAStainR/

Bioconductor package available from: https://doi.org/doi:10.18129/B9.bioc.HPAStainR

Source code available from: https://github.com/tnieuwe/HPAStainR

Archived source code as at time of publication: https://doi.org/10.5281/zenodo.4594755[14].

Software license: Artistic-2.0

Analysis code available from: https://github.com/tnieuwe/HPAstainR_dev_paper

Archived analysis code as at time of publication: https://doi.org/10.5281/zenodo.4594672[15].

License: Artistic-2.0

Acknowledgments

We thank Matthew N. McCall, Zachary P. Brehm, Stephanie Y. Yang, and Veronica F. Busa for their consultation on the creation of the software.

References

1. Uhlén M, Fagerberg L, Hallström BM, *et al.*: **Proteomics. Tissue-based map of the human proteome.** *Science.* 2015; **347**(6220): 1260419.
PubMed Abstract | Publisher Full Text
2. Uhlen M, Zhang C, Lee S, *et al.*: **A pathology atlas of the human cancer transcriptome.** *Science.* 2017; **357**(6352): eaan2507.
PubMed Abstract | Publisher Full Text
3. Sjöstedt E, Zhong W, Fagerberg L, *et al.*: **An atlas of the protein-coding genes in the human, pig, and mouse brain.** *Science.* 2020; **367**(6482): eaay5947.
PubMed Abstract | Publisher Full Text
4. Kuleshov MV, Jones MR, Rouillard AD, *et al.*: **Enrichr: a comprehensive gene set enrichment analysis web server 2016 update.** *Nucleic Acids Res.* 2016; **44**(W1): W90–7.
PubMed Abstract | Publisher Full Text | Free Full Text
5. Gatto L, Martin M: **hpar: Human Protein Atlas in R**. R package version 1.32.1. 2020.
Reference Source
6. Tran AN, Dussaq AM, Kennell T, *et al.*: **HPAanalyze: an R package that facilitates the retrieval and analysis of the Human Protein Atlas data.** *BMC Bioinformatics.* 2019; **20**(1): 463.
PubMed Abstract | Publisher Full Text | Free Full Text
7. Chang W, Cheng J, Allaire JJ, *et al.*: **shiny: Web Application Framework for R**. 2020.
Reference Source
8. Gentleman RC, Carey VJ, Bates DM, *et al.*: **Bioconductor: open software development for computational biology and bioinformatics.** *Genome Biol.* 2004; **5**(10): R80.
PubMed Abstract | Publisher Full Text | Free Full Text
9. Nieuwenhuis OT, Halushka KM: **HPAStainR – A Bioconductor and shiny app to query protein expression patterns in the Human Protein Atlas.** Harvard Dataverse, V1, UNF:6:o2EDbY39avbmTP9qswinCA== [fileUNF]. 2020.
http://www.doi.org/10.7910/DVN/CL5ZTA
10. Franzén O, Gan LM, Björkegren JLM: **PanglaoDB: a web server for exploration of mouse and human single-cell RNA sequencing data.** *Database (Oxford).* 2019; **2019**: baz046.
PubMed Abstract | Publisher Full Text | Free Full Text
11. Anene DF, Rosenberg AZ, Kleiner DE, *et al.*: **Utilization of HPASubC for the identification of sinusoid-specific proteins in the liver.** *J Proteome Res.* 2016; **15**(5): 1623–9.
PubMed Abstract | Publisher Full Text
12. McCall MN, Illei PB, Halushka MK: **Complex Sources of Variation in Tissue**

Expression Data: Analysis of the GTEx Lung Transcriptome. *Am J Hum Genet.* 2016; **99**(3): 624–35.
PubMed Abstract | Publisher Full Text | Free Full Text

13. Cornish TC, Chakravarti A, Kapoor A, *et al.*: **HPASubC: A suite of tools for user subclassification of human protein atlas tissue images.** *J Pathol Inform.* 2015; **6**: 36.
PubMed Abstract | Publisher Full Text | Free Full Text

14. Nieuwenhuis T: **tnieuwe/HPAStainR: HPAStainR Dev Release (Version 1.1.4).** *Zenodo.* 2021.
http://www.doi.org/10.5281/zenodo.4594755

15. Nieuwenhuis T: **tnieuwe/HPAStainR_dev_paper: HPAStainR Analysis with adjustments for reviewer comments (Version 1.1.0).** *Zenodo.* 2021.
http://www.doi.org/10.5281/zenodo.4594672

Open Peer Review

Current Peer Review Status: ✔ ✔ ✔

Version 2

Reviewer Report 07 April 2021

https://doi.org/10.5256/f1000research.55429.r81836

✔ **Yasin Kaymaz**
Harvard University, Boston, MA, USA

I would like to thank all authors for addressing my concerns. I have no further comment about the revised version.

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Bioinformatics particularly in RNA-seq, single-cell genomics, and algorithm dev. areas.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Reviewer Report 06 April 2021

https://doi.org/10.5256/f1000research.55429.r81837

✔ **Laurent Gatto**
De Duve Institute, UCLouvain, Brussels, Belgium

I have no further comments to make.

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Computational biology, proteomics, genomics, research software

development, open and reproducible research.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Reviewer Report 22 March 2021

https://doi.org/10.5256/f1000research.55429.r81838

Mazdak Salavati
The Roslin Institute and Royal (Dick) School of Veterinary Studies, University of Edinburgh, Edinburgh, UK

I have no further comments.

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Genetics and Genomics, Bioinfomatics, Cell biology.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Version 1

Reviewer Report 11 January 2021

https://doi.org/10.5256/f1000research.29559.r72699

Yasin Kaymaz
Harvard University, Boston, MA, USA

Authors introduce a new R package called HPAStainR for quickly identifying putative cell types given a list of protein or gene names by searching these through the Human Protein Atlas database. HPAStainR has been developed to allow users to query multiple gene/protein names in the HPA database and return an ordered list of cell types for which the query list might be enriched. They also make use of the Panglao single-cell expression database to validate their

predictions. The main purpose of the tool addresses a need in the field and should be encouraged. However, I would like to mention some of my concerns below about the basis of the tool and the structure of the manuscript.

It has been mentioned that there were two main input files from the HPA database which can be downloaded with the 'HPA_data_downloader' function. Do you need to download all the data from the HPA for each analysis? Is this really necessary? Can't you do this on the fly?

The staining score seems to be constructed a bit arbitrarily. Based on staining intensity (high, medium, and low), scores are weighted with some arbitrary constant values and normalized by the total query number. But I wonder if there is any skew in this scoring scheme in case extreme queries are tried, such as all high with many proteins or all low, etc. Especially, given the highly non-uniform scoring distribution in the HPA database (Figure 1).

Confidence score? Why not combine this equation with the staining score? The point of keeping them as separate metrics seems a bit vague. Also, a bit confusing from the user standpoint.

Do the authors use the cell type enrichment p-value in the enrichment at all? If so, it is not obvious from the text.

Figure 1, 2, and 3;
The actual purpose of these figures should be clearly explained in the main text. It is hard to get it unless staring at them for a while. Also, these first three figures are not directly related to the tool explained here. They are rather some statistics showing the key points of HPA data. I would recommend replacing these with more directly related graphs demonstrating the performance of the HPAStainR and including the current ones as supplemental data.

Is the rationale for developing the new software tool clearly explained?
Yes

Is the description of the software tool technically sound?
Partly

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?
Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?
Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?
Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Bioinformatics particularly in RNA-seq, single-cell genomics, and algorithm dev. areas.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 12 Mar 2021

Tim Nieuwenhuis, Johns Hopkins University School of Medicine Baltimore, Baltimore, USA

Thank you for taking the time to review this paper and the associated package, below is a response to your concerns and the changes we've made because of them.

1. ... Do you need to download all the data from the HPA for each analysis? Is this really necessary? Can't you do this on the fly?
In the current version of `HPA_data_downloader()`, as long as the parameter `save_file` is set to `TRUE`, it will only require you to download the files once. The next release of the package (v1.1.4 available on the main branch at https://github.com/tnieuwe/HPAStainR) includes an updated version of the function that marks the download date of the files, and also allows the user to select which downloaded file they want to use in their analysis for backward reproducibility.

2. The staining score seems to be constructed a bit arbitrarily...
We acknowledge the somewhat arbitrary nature of the equation, although it is based on prior histology scoring methods more common in histopathology studies. Additionally, we have generated a distribution of 1,000 HPAStainR results on randomly selected genes, including the top 10 results from HPAStainR and all results, for random gene lists of sizes 10, 25, 50, and 100, these are found as Extended Figures 1 and 2. Our findings show the increased number of genes results in lower staining scores. Regardless, for the top 10 results, the distribution appears to be normal. Analysis of all staining data suggests a right skew. We were unable to create an extreme skew, but we cannot entirely exclude it for some unique queries.

3. Why not combine the confidence score with the staining score?
The reason that they are separate is because the confidence score is simply a scaled staining score that is only used for PanglaoDB-HPAStainR analyses. The context we used the confidence score was strictly for testing how well proteins considered marker genes in PanglaoDB mark the equivalent cell types in HPA. This was described in greater detail to reviewer 2's query.

4. Do the authors use the cell type enrichment pvalue at all?
Yes, in the original manuscript we use the p-value at the end of the **HPAStainR identified many of the cell types in PanglaoDB** section. We also show it in figure 6, and in the updated version of the manuscript, we now include it in table 1. Also, the p-values have changed as the analysis has been updated from a X^2 analysis to a Fisher's Exact Test.

5. The purpose of these figures should be clearly explained in the main text.
We have added further information in the figure legends to better clarify what the figures represent. HPAStainR does not have an output that lends itself to a graphical representation, however, we have added two extended figures to the manuscript showing the distribution of the staining score in various random samplings as noted above.

Competing Interests: We have no competing interests to disclose.

Reviewer Report 11 November 2020

https://doi.org/10.5256/f1000research.29559.r72698

?

Laurent Gatto
De Duve Institute, UCLouvain, Brussels, Belgium

Nieuwenhuis and Halushka describe the HPAStainR package, recently released as part of the Bioconductor project. HPAStainR uses protein immunostraining data from the Human Protein Atlas to asses whether a user-provided list of proteins or genes is associated with a particular cell type.

Introduction

The authors fail to cite other Biocondcutor packages related to the Human Protein Atlas, namely hpar (in Bioconductor for 8 years) and HPAnalyze (in Bioconductor for 2 years). While the functionalities of these packages are different (but see below), citing similar packages in Bioconductor seems very relevant for an paper in the Bioconductor gateway.

(Note that I am the author of the hpar package)

Methods: Operation

- Installation instructions should not refer to the development version of the package, but instruct users to install and use the release version. For two reasons: first the development of a package doesn't guarantee any stability, second it puts additional burden on the user to (1) potential have to install the development version of R and (2) end up with installing _all_ Bioconductor development packages. Given that package is released now, the installation instructions should absolutely be updated accordingly.

- Please fix code formatting.

Input:

- The `HPA_data_downloader()` function is used to download 2 datasets. Note that the 'hpar' package could have been a Bioconductor package to integrate with here, so as to

avoid repeated downloading of the data and/or to provide some reproducibility in the analyses (see below for details on this).

```
> ## Executed on the 10 November at 20:01
> 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')
```

```
> library(hpar)
> packageVersion("hpar") ## '1.32.1'
> data(hpaCancer) ## load the data
> data(hpaNormalTissue)
> all.equal(hpaNormalTissue, HPA_data$hpa_dat)
[1] TRUE
> all.equal(hpaCancer, HPA_data$cancer_dat, check.attributes = FALSE) ## different colnames only
[1] TRUE
> ## BUT see below
> HPA_out2 <- HPAStainR(c('PRSS1', 'PNLIP', 'CELA3A', 'PRL'),
                        hpaNormalTissue,
                        hpaCancer,
                        'both')
```

There are generally two approaches when it comes to using data from remote resources:

1. Download the data on the fly, which allows to use the very latest version of the data, but at the expense of lack of reproducibility/tracking. Indeed, the results can unexpectedly change from one day to another. This would be to option in the HPAStainR package, as well as other Bioconductor packages such as rols (that queries various ontologies, including GO).

2. Packaging and versioning data to guarantee tracking and reproducibility of the analysis. This is for instance the option provided by hpar (latest hpar release, version 1.32.1, provides HPA data version 19.3, dated 2020/03/06). Other Bioconductor package that offer this solution are GO.db (package GO), and many other Bioconductor data packages.

It would be useful for HPAStainR to make these assumptions explicit and to document how to track results: using hpar or manually (and documenting!) storing the tables downloaded using `HPA_data_downloader()`.

Returning to the reproducibility of the results, despite identical input data (except for the cancer data column names), it is intriguing that the results aren't identical. There are three cell types that have different p-values/adjusted p-values.

```
> all.equal(HPA_out, HPA_out2)
[1] "Component “p_val”: 2 string mismatches"
[2] "Component “p_val_adj”: 1 string mismatch"
```

```
> which(HPA_out$p_val != HPA_out2$p_val)
[1] 4 5
> which(HPA_out$p_val_adj != HPA_out2$p_val_adj)
[1] 1
> HPA_out[c(1, 4, 5), c("cell_type", "p_val", "p_val_adj")]
# A tibble: 3 x 3
  cell_type                       p_val  p_val_adj

1 PANCREAS - exocrine glandular cells <0.005 0.076
2 SMALL INTESTINE - glandular cells   0.37   1.000
3 COLON - glandular cells             0.17   1.000
> HPA_out2[c(1, 4, 5), c("cell_type", "p_val", "p_val_adj")]
# A tibble: 3 x 3
  cell_type                       p_val  p_val_adj

1 PANCREAS - exocrine glandular cells <0.005 0.038
2 SMALL INTESTINE - glandular cells   0.34   1.000
3 COLON - glandular cells             0.18   1.000
```

It would be interesting for the authors to investigate this, given that the PANCREAS - exocrine glandular cells change from non-significant to significant.

- 'hpa_summary_maker.R` must be `HPA_summary_maker()'.

Output:

- Running 'HPAStainR()' as indicated in the man page returns a tibble. It is unclear what the "Shiny DataTable" output in the text refers to. I couldn't find any further information in the man page. Did the authors possibly mean the 'shiny_HPAStainR()' function? Anyway, the two function should be mentioned in the manuscript.

- Running that very same example, the output table is incoherent in the mode of the variables: p-values (p_val) and adjusted p-values (p_val_adj) are encoded as characters.

HPA data distribution

The data discussed in 'HPA data distribution' and available in Extended Table 1 seems to be the same one as returned by the 'HPA_summary_maker()' function. Please mention this explicitly, to allow users to easily generate this table for different data.

Confidence score calculation

It isn't clear why the PanglaoDB needs an additional confidence score, or why it wouldn't be relevant or useful in other contexts.

Cell type enriched p-value

It isn't clear what is refereed to by low, normal and medium stringency. Based on the straining score calculation equation, it seems to be related to the low, high, medium staining intensity. Please define the notion of stringency.

The Panglao Database

In Table 1 and in the 'HPAStainR identified many of the cell types in PanglaoDB' section, the authors make use of the confidence score instead of the p-values to support their validation. Why don't they make use of that p-value, advertised in the previous 'Cell type enriched p-value' section?

HPAStainR can help determine cell type populations in bulk RNA sequencing

The authors show the first hits, matching the expected cell types. Are there any other cell types that match with an adjusted p-value < 0.05?

Software availability

The authors mention that the analysis code is available from: https://github.com/tnieuwe/HPAstainR_dev_paper. This repositories however contains a lot of 'old' files ('old_vignette', 'old_files', 'old_versions'), including what appears an old version of the package in 'package_HPAstainR'. Version control is the ideal tool to store and track files over time, and dedicated version can be specifically tagged or released.

Are these old version relevant? What are the differences with the more recent analyses?

Is the rationale for developing the new software tool clearly explained?
Yes

Is the description of the software tool technically sound?
Partly

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?
Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?
Partly

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?
Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Computational biology, proteomics, genomics, research software development, open and reproducible research.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have

significant reservations, as outlined above.

Author Response 12 Mar 2021

Tim Nieuwenhuis, Johns Hopkins University School of Medicine Baltimore, Baltimore, USA

We thank you for your time reviewing our package. Below we have individual responses to your queries formatted the same way as your review.

Introduction
Thank you for pointing out our oversight. We have included, in the newest version of the paper, a citation for both hpar and HPAnalyze. We will discuss the incorporation of hpar into our package later in this response.

Methods
Operation:
In the new version, we have updated this section to properly reflect the release of HPAStainR and the code, properly formatted, to download said library.

Input:
HPAStainR's next release, currently available on the main branch of https://github.com/tnieuwe/HPAStainR and the devel version of Bioconductor, will include a vignette section on how to use hpar's data in HPAStainR. We are currently keeping the function `HPA_data_downloader()` as its one benefit over hpar is that it gives access to the most recent data if the semi-annual Bioconductor release doesn't pair with the HPA release. In response to the remote data comment, we have also updated the package (v1.1.4) on the master branch for future release (https://github.com/tnieuwe/HPAStainR). The changes to `HPA_data_downloader()` includes the following:

1. The name of the file and function has changed from `hpa_data_downloader()` to `HPA_data_downloader()`

2. Every time a user downloads and saves the HPA files, the date of the download is provided.

3. To help in version control there are three new parameters that assist in maintaining reproducibility:
    1. `version_date_normal`: This parameter allows the user to insert a date string in YYYY-MM-DD format to select a normal tissue file downloaded on the respective date. If a date is not supplied the default argument is "last" which will find the latest version of the file.
    2. `version_date_cancer`: The same as version date normal, but for the pathology/cancer file.
    3. `force_download`: An argument forcing the re-downloading of files from the HPA website. The purpose of this argument is to allow the user to update their local files, as `HPA_data_downloader()` by default will use local files over downloading said files again.

We investigated the incongruency between your run of HPAStainR with hpar and our data,

and found the issue was most likely due to the usage of simulated p-values in the chi-square analysis. To fix this, and overall improve the tool, we have changed the base test in HPAStainR() to a Fisher's Exact Test, which works on the non-parametric data we have. From data not shown here, it does not lengthen the run time of HPAStainR(). Therefore all p-values in the paper have been updated to the results of the Fisher's Exact Test.

Output:
We have updated the text to reflect the output of both `HPAStainR()` and `shiny_HPAStainR()` separately. If using the base `HPAStainR ()` function a tibble is returned, while the table returned in the `shiny_HPAStainR` is referred to as a Datatable.
The p-values were character values due to the usage of `format.pvalue().` This has been revised and changed to numerical values for the next release.

##HPA data distribution
We have also clarified Extended Table 1 and its relationship as the output to the 'HPA_summary_maker()' function.

Confidence Score Calculation
The creation of the "confidence scores" was strictly for our PanglaoDB analysis comparison and is not useful outside of validation studies. PanglaoDB had cell types with wildly variable numbers of marker genes. This caused cell types with 1 marker gene that had "high" staining (such as Schwann cells in extended table 4) to become a top hit in PanglaoDB (by staining score), but that does not reveal the accuracy of the tools as the marker genes may just be unique to the cell type in its specific tissue. Therefore we generated the confidence score, which controls and adjusts for the number of marker genes used because having ~40 proteins properly staining is more informative than ~2. The confidence scores are not used in the HPAStainR tool due to the higher consistency of protein staining per cell type, allowing the staining score to be sufficient for ranking.

Cell type enriched p-value
We have added more information explaining how the stringency is based on the "Reliability" column and how each level of stringency functions. The next release of HPAStainR will include this information in the description of the stringency parameter as well.

The Panglao Database
We have revised the table to include p-values of the tissue specificity. The reason we used confidence scores is that the p-values were simulated in the chi-square analysis used in HPAStainR, where there is a smaller range of possible p-values. We felt the confidence score was a better way to order the cell types due to its ability to highlight the staining score while controlling for how many genes each PanglaoDB cell type has.

HPAStainR can help determine cell type populations in bulk RNA Sequencing
In extended table 5 stomach glandular cells have the second-lowest adjusted p-value at 0.306 and in extended table 6 fallopian tube glandular cells, nasopharynx respiratory epithelial cells, endometrium glandular cells, and cervix-uterine glandular cells all have an adjusted p-value <.05. However, the bronchus epithelial cell's p-value is still the smallest at $1.58x10^{-21}$ compared to fallopian tube glandular cells at $3.99x10^{-19}$.

Software availability
All “old” versions have been removed and will remain so in the next version.

Competing Interests: We have no competing interests to disclose

Reviewer Report 05 November 2020

https://doi.org/10.5256/f1000research.29559.r73228

Mazdak Salavati
The Roslin Institute and Royal (Dick) School of Veterinary Studies, University of Edinburgh, Edinburgh, UK

This manuscript describes a novel tool developed for extending access to the visual proteomics dataset produced by Human Protein Atlas (HPA).
The main two features that this shinyApp/R-package is as follows:

1. Enabling batch query of gene or protein name lists for cell type composition identification.
2. Use of bulk RNA data (tissues) in order to unravel cellular composition of the starting RNA sample.

Authors have carried out an external validation with PanglaoDB human cell type dataset in order to confirm the soundness of both staining score and confidence score equations which are largely compatible with HPA cell type groups. They have also studied GTEx RNA-Seq (2 cell types) input gene lists with their pipeline which again was confirm by the top hit returned by the tools (albeit varying Staining score).

This tool has been developed very thoroughly and with a clear demand in the community at its design. However as highlighted by the authors in the conclusion section, one should approach subjective scored histochemistry obtained datasets always with caution. As the scoring bias introduced by scorers will remain as part of the rank outputs.

Suggestions for the authors:

- I would highly recommend to include more data from GTEx tissues and expand the result section of your manuscript.

- Perhaps consider a cross validation procedure in the future once more datasets are available for the same cell type. A 5-10 fold cross validation can immensely improve the reliability of the output ranked results.

- Knowing a package like EnrichR that covers a variety of GSEA and Pathway enrichment databases, would it make sense to collaborate with their development team to expand

functionality for cell type prediction through EnrichR? That's a question or challenge for the authors to answer or decide.

Is the rationale for developing the new software tool clearly explained?
Yes

Is the description of the software tool technically sound?
Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?
Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?
Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?
Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Genetics and Genomics, Bioinfomatics, Cell biology.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Author Response 12 Mar 2021

Tim Nieuwenhuis, Johns Hopkins University School of Medicine Baltimore, Baltimore, USA

Thank you for reviewing our paper, below are our responses to your suggestions:

Suggestion 1, Recommend to include more data from GTEx tissues:
This tool was generated to specifically analyze GTEx data. We are working on a project to use this tool for a more in-depth analysis of GTEx, similar to the lung paper cited.

Suggestion 2, Cross-validation Procedure in the future once more datasets are available:
That is an excellent idea. As more datasets become available, we will work to incorporate them into this tool.

Suggestion 3, Collaborate with EnrichR:
Once HPAStainR package has fully matured through the review process, we will determine if it can be included in the excellent EnrichR tool.

Competing Interests: We have no competing interests to disclose.