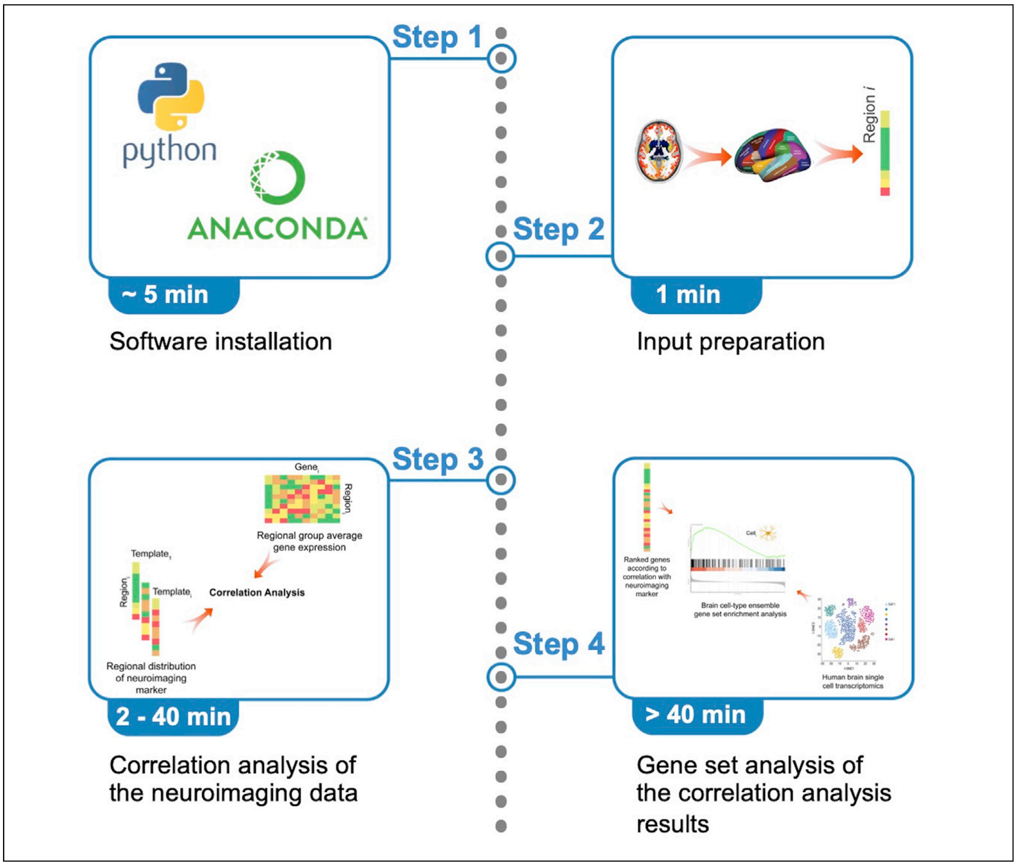
STAR Protocols

CellPress
OPEN ACCESS

Protocol

Integrating neuroimaging and gene expression data using the imaging transcriptomics toolbox


Step 1
python
ANACONDA
~ 5 min
Software installation
Step 2
Region i
1 min
Input preparation
Step 3
Gene$_i$
Region$_i$
Regional group average gene expression
Template$_1$
Region$_i$
Template$_i$
Correlation Analysis
Regional distribution of neuroimaging marker
2 - 40 min
Correlation analysis of the neuroimaging data
Step 4
Ranked genes according to correlation with neuroimaging marker
Cell$_i$
Brain cell-type ensemble gene set enrichment analysis
Human brain single cell transcriptomics
> 40 min
Gene set analysis of the correlation analysis results

The integration of neuroimaging and transcriptomics data, *Imaging Transcriptomics*, is becoming increasingly popular but standardized workflows for its implementation are still lacking. We describe the Imaging Transcriptomics toolbox, a new package that implements a full imaging transcriptomics pipeline using a user-friendly, command line interface. This toolbox allows the user to identify patterns of gene expression which correlates with a specific neuroimaging phenotype and perform gene set enrichment analyses to inform the biological interpretation of the findings using up-to-date methods.

Alessio Giacomel, Daniel Martins, Matteo Frigo, Federico Turkheimer, Steven C.R. Williams, Ottavia Dipasquale, Mattia Veronese

alessio.giacomel@kcl.ac.uk (A.G.)
daniel.martins@kcl.ac.uk (D.M.)

Highlights

Protocol for using the Imaging Transcriptomics toolbox

Identifies transcriptomic correlates of neuroimaging data

Performs gene enrichment analysis to contextualize findings

Standardized and user-friendly pipeline using robust statistics

Giacomel et al., STAR Protocols 3, 101315
June 17, 2022 © 2022 The Author(s).
https://doi.org/10.1016/j.xpro.2022.101315

STAR Protocols

CellPress
OPEN ACCESS

Protocol

Integrating neuroimaging and gene expression data using the imaging transcriptomics toolbox

Alessio Giacomel,[1,5,*] Daniel Martins,[1,6,*] Matteo Frigo,[3,4] Federico Turkheimer,[1] Steven C.R. Williams,[1] Ottavia Dipasquale,[1] and Mattia Veronese[1,2]

[1]Department of Neuroimaging, IoPPN, King's College London, London, UK

[2]Department of Information Engineering, University of Padova, Padova, Italy

[3]Corsmed AB, Stockholm, Sweden

[4]ATHENA Project Team, Inria Sophia Antipolis - Mediterranée, Université Côte d'Azur, Nice, France

[5]Technical contact

[6]Lead contact

*Correspondence: alessio.giacomel@kcl.ac.uk (A.G.), daniel.martins@kcl.ac.uk (D.M.)
https://doi.org/10.1016/j.xpro.2022.101315

SUMMARY

The integration of neuroimaging and transcriptomics data, *Imaging Transcriptomics*, is becoming increasingly popular but standardized workflows for its implementation are still lacking. We describe the Imaging Transcriptomics toolbox, a new package that implements a full imaging transcriptomics pipeline using a user-friendly, command line interface. This toolbox allows the user to identify patterns of gene expression which correlates with a specific neuroimaging phenotype and perform gene set enrichment analyses to inform the biological interpretation of the findings using up-to-date methods.

For complete details on the use and execution of this protocol, please refer to Martins et al. (2021).

BEFORE YOU BEGIN

This toolbox allows the user to identify patterns of gene expression which correlates with a specific neuroimaging phenotype and perform gene set enrichment analyses to inform the biological interpretation of the findings using up-to-date methods.

This section includes all necessary steps to setup a dedicated python environment and install the Imaging Transcriptomics toolbox.

Anaconda python environment

Timing: < 10 min

The imaging transcriptomics package works in Windows and Unix systems (Mac OSX, Linux) with Python 3 (>=3.6). To run the script or use the library without the risk of dependencies conflicts with other scripts or libraries, we recommend installing everything in a dedicated Anaconda environment. The environment hereafter installed, will occupy about 1.1 GB of hard drive space (on a MacBook Pro with 1.4 GHz Quad-Core Intel Core i5 processor and macOS Big Sur). The creation of a dedicated environment allows the user to avoid the accidental generation of conflicts with other software or Python versions.

Note: The occupied space might slightly vary between different systems (i.e., macOS, Linux, Windows) due to the internal filesystem design.

1. Anaconda can be downloaded from https://www.anaconda.com/products/individual and installed following the specific instructions for individual computer specifications.
2. Once Anaconda is installed restart any open terminal and create a dedicated environment using Python (version 3.7) and pip using the following command:

```
> conda create –name transcriptomics python=3.7 pip
```

Follow the prompted instructions until the environment gets successfully created, for more details on the creation of Anaconda environments please refer to the official documentation of anaconda (https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html).

3. Activate your newly created Anaconda environment by typing:

```
> conda activate transcriptomics
```

Before you proceed, make sure that you are in the correct environment (troubleshooting 1).

4. We will now install the ENIGMA Toolbox (https://github.com/MICA-MNI/ENIGMA), a package released by the ENIGMA consortium that contains plotting functions used by the Imaging Transcriptomics toolbox to create some plots. To install the ENIGMA Toolbox, run the following commands:

```
> git clone https://github.com/MICA-MNI/ENIGMA.git

> cd ENIGMA

> python setup.py install
```

Note: Some errors may appear during the installation of the toolbox, to resolve them please refer to troubleshooting 4.

5. The last pre-requisite before installing the package is the installation of the pypls library to perform partial least square regression (PLS). This can be done using the following command:

```
> pip install -e git+https://github.com/netneurolab/pypyls.git/#egg=pyls
```

⚠ CRITICAL: Do not install pyls from the python package manager (i.e., Pypi), since that package performs different tasks.

Installation

🕒 Timing: 1 min

After the creation of the dedicated environment and installing all the dependencies that can't be installed automatically, here we will install the core toolbox with its python dependencies. This will make available two scripts, one for the correlation analysis between neuroimaging data and gene expression and one for to perform gene set enrichment analysis (GSEA).

6. To install the Imaging Transcriptomic toolbox, comprising of the python library and command line script, run the following command from your terminal:

```
> pip install imaging-transcriptomics
```

After the process is complete, you can check if the installation was successful by typing the following command in the terminal:

```
> imagingtranscriptomics –help
```

Or by typing:

```
> imt_gsea –help
```

With both the previous commands, if successful, the help for each of the scripts will be displayed.

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
Gene expression data from human brain tissues	Allen Human Brain Atlas (AHBA)	http://human.brain-map.org/
Single cell RNA-seq data	(Lake et al., 2018)	NA
Software and algorithms		
MATLAB	(MATLAB, 2020)	https://uk.mathworks.com/

FMRIB Software Library (FSL)	(Jenkinson et al., 2012;
	Smith et al., 2004;
	Woolrich et al., 2009)
		https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/

Anaconda	Anaconda Inc.	https://www.anaconda.com/products/individual

Abagen toolbox	(Arnatkevičiūtė et al., 2019;
	Hawrylycz et al., 2012;
	Markello et al., 2021)
		https://github.com/netneurolab/abagen (https://doi.org/10.5281/zenodo.5129257)

Alleninf	(Gorgolewski et al., 2014)	https://github.com/chrisfilo/alleninf
ENIGMA Toolbox	(Larivière et al., 2021)	https://enigma-toolbox.readthedocs.io/en/latest/index.html
Netneurotools python library	Network Neuroscience Lab, Brain Imaging Centre, McGill University	https://github.com/netneurolab/netneurotools
Gseapy python library	(Fang, 2020)	https://github.com/zqfang/GSEApy
The Imaging Transcriptomics Toolbox	(Giacomel et al., 2022)	https://github.com/molecular-neuroimaging/Imaging_Transcriptomics
Other		
PET template	(Beliveau et al., 2017)	https://xtra.nru.dk/FS5ht-atlas/

STEP-BY-STEP METHOD DETAILS

In the following sections, we describe step-by-step how to perform an imaging transcriptomics analysis of a neuroimaging map from start to finish. This includes identifying genes whose expression correlate spatially with the neuroimaging map and performing gene set enrichment analysis to inform the biological interpretation of the results. Such analyses require detailed information on gene expression across multiple regions of the *post-mortem* human brain, which right now can only be accessed through the Allen Human Brain Atlas (AHBA).

The toolbox allows implementing two types of analyses to quantify the association between neuroimaging and gene expression data: i) a simple mass-univariate Spearman correlation analysis; ii) a multi-variate PLS regression analysis. The method to be used is defined as an input by the user.

Both methods have been used in previous works applying imaging transcriptomics (Fulcher et al., 2021; Morgan et al., 2019) and have their own strengths and weaknesses that should be considered on a case-by-case basis. Ultimately, the toolbox provides a list of genes ranked by how well they associate with the distribution of a neuroimaging marker input by the user and identifies which genes are significantly associated with the marker using state-of-the-art methods that account for bias induced by the spatial autocorrelation of the data.

In order to perform imaging transcriptomics analyses, both neuroimaging and gene expression should be mapped into the same space. Currently, the analyses implemented in the toolbox are based on the Desikan-Killiany (DK) parcellation (Desikan et al., 2006). For the neuroimaging data, the toolbox implements a simple averaging of the signal across all voxels of each parcel in the atlas. For the gene expression data, the process of mapping the AHBA data to DK parcels was implemented a priori using the *abagen* toolbox (https://www.github.com/netneurolab/abagen). Briefly, genetic probes were reannotated and only probes that could reliable be matched to genes were kept and filtered based on their value relative to the background noise by using a threshold of 50%, yielding a total of 15,633 probes (Arnatkevičiūtė et al., 2019). Next, tissue samples were assigned to brain regions using their corrected MNI coordinates (https://github.com/chrisfilo/alleninf), samples were matched to regions constraining this to hemisphere and cortical/subcortical subdivisions. Samples were assigned to brain regions in the atlas if their coordinates in MNI space were within 2 mm of a given parcel. To reduce the potential for misassignment, sample-to-region matching was constrained by hemisphere and gross structural divisions (i.e., cortex, subcortex/brainstem, and cerebellum). All tissue samples not assigned to a brain region in the provided atlas were discarded (Markello et al., 2021). Samples were then averaged across donors and normalized, resulting in a final single matrix with rows corresponding to brain regions and columns corresponding to the 15,633 genes.

Irrespectively of the statistical method selected by the user (PLS or Spearman correlation), the inferential statistics is calculated using gold-standard methods that are robust to the intrinsic autocorrelation of the imaging data. All significance testing is based on permutation testing, where 1,000 null spatial maps are derived using a combination of spin rotations of the cortical regions and resampling of the subcortical regions. The spin rotations are implemented using the *Vasa* method as in previous studies (Alexander-Bloch et al., 2013a, 2013b; Markello and Misic, 2021; Váša et al., 2018). The same nulls are then used in the ensemble gene set enrichment analyses to control for false positives related to the spatial autocorrelation of the data, as recently recommended (Fulcher et al., 2021).

To illustrate the various steps of the analysis with the toolbox, we will use as an example a publicly available positron emission tomography (PET) average template of the serotonin receptor 5-HT2A ([^{11}C]Cimbi-36) from (Beliveau et al., 2017).

> ***Note:*** The scan downloadable from the online repository (https://xtra.nru.dk/FS5ht-atlas/) must be reshaped since it has a data matrix of 182 × 218 × 182 × 1 (for more see troubleshooting 2). In addition, to avoid problems with the file system the scan should be renamed by replacing the dots (.) in the name with underscores (_). For the scope of the following example the scan has been renamed to *5-HT2A_mean_bmax.nii.gz*.

1. Select the path of your input file.

   Note: For the input, either common neuroimaging scan formats (NIfTI - .nii, .nii.gz) or text files (i.e., .tsv, .csv, .txt) can be used. The path should be provided as an absolute path (e.g., "/home/username/data_folder/myfile" instead of "./data_folder/myfile").

   ⚠ CRITICAL: If the input is a neuroimaging scan, this *must* be already in standard MNI152 space and have a voxel size of 1 mm isotropic (imaging matrix size of 182 × 218 × 182).

On the other hand, if the input is a text file, this *must* have only one column, with no header and a regional value in each row, following the order of the DK atlas (the full list of regions is available in the supplement file Table S1).

Partial least square regression analysis

⏱ Timing: < 5 min

This step is the first of two alternative ways to run the analysis, and it employs PLS regression to identify latent components that maximize the correlation between neuroimaging and gene expression data.

2. Run the script *imagingtranscriptomics* using the pls option:

```
> imagingtranscriptomics -input <input_path> [-output <output-path>] [-regions <all|cort+sub|cort>] [-no-gsea] [-genest] pls <pls_options>
```

The arguments to be provided are:

a. –input <input_path>: the path to the input file (the path from step 1).
b. –output <output_path> (optional): the path where the results should be saved.

Note: If this is not provided the results will be saved in the path of the input file.

c. –regions <all|cort+sub|cort>(optional): Allows the user to select which regions to use in the analysis. This is particularly useful with certain types of data (e.g., EEG) where subcortical regions might not be available. The available options are *all* (or equivalently *cort+sub*), which specifies that all regions should be used, or *cort* where only cortical regions are used.
d. –no-gsea (optional): this flag allows running the script without performing GSEA.

Note: If this is not provided, the script will also run the GSEA step (described below).

e. –geneset (optional): Name of the gene set to be used in the GSEA analysis.

Note: If the –no-gsea flag is provided, this option will be ignored. If you also want to perform GSEA (i.e., excluding the –no-gsea flag), a gene set should be selected - for more information on the available gene sets refer to the GSEA step.

f. pls <pls_options>: uses PLS regression to analyze the data. After the pls keyword, only one of the following inputs is required:
   i. –ncomp <n>: number of components to use in the PLS regression (this must be an integer between 1 and 15).
   ii. –var <n>: percentage of the variance to explain. With this option, the optimal number of components will be automatically calculated by the script (this must be a float between 0 and 1).

   For instance, with the example data, we can run the command:

```
> imagingtranscriptomics -input 5-HT2A_mean_bmax.nii.gz -regions all -no-gsea pls -ncomp 1
```

Which will run the analysis with one PLS component on the example scan, without running GSEA, and save the results the same directory as the input scan (the results will be in a folder named *Imt_5-HT2A_mean_bmax_pls*).

Mass-univariate correlation analysis

Timing: > 30 min

As an alternative to PLS regression, the toolbox also offers the option to run the analysis using mass-univariate Spearman correlations. This option will simply calculate Spearman correlations between the neuroimaging vector and the expression of each gene.

Note: if you want to analyze the data with PLS regression, you can skip this step.

3. Run the script *imagingtranscriptomics* using the correlation option:

```
> imagingtranscritomics -input <input_path> [-output <output-path>][-regions <all|cort+-
sub|cort>] [-no-gsea] [-genest] corr [-cpu <n_cpu>]
```

The first optional input is described in the PLS analysis section (points 2a-2e); the additional optional input for the script in this case is:

a. `-cpu <n_cpu>`: number of cpu to be used for the calculation of the correlations (the default number is 4).

Note: This step takes a considerably longer time compared to the PLS analysis, since the number of correlations to estimate is much greater.

With the example data we can run the command:

```
> imagingtranscriptomics -input 5-HT2A_mean_bmax.nii.gz -regions all -no-gsea corr
```

This command will run the analysis on all brain regions, without running GSEA, with mass univariate correlation and save the results in the same directory as the input scan (the results will be saved in a folder named *Imt_5-HT2A_mean_bmax_corr*).

Note: Irrespective of the method selected (PLS regression or mass-univariate correlation), the toolbox produces lists of genes ranked according to the strength of the spatial alignment between the neuroimaging phenotypes (e.g., regional distribution of a PET tracer, statistical map reflecting effects of a drug or case-control differences for a certain neuroimaging metric) and their expression. Please, note that when the user does not have *a priori* hypotheses about specific genes or pathways, interpreting the output in biological terms can be challenging. For instance, one might be interested in understanding if the top genes positively associated with a certain neuroimaging phenotype belong to specific biological pathways or brain cell-types. Answers to this type of questions can be provided by gene set enrichment analyses, which we will describe in the next section.

Ensemble gene set enrichment analysis (GSEA)

Timing: > 1 h

GSEA uses a statistical hypothesis-testing framework to assess which categories of genes (i.e., set of genes sharing a certain biological function, such as neuronal genes or astrocytic genes) are most strongly related to a given phenotype, leveraging annotations of genes to categories from open ontologies, like the Gene Ontology (GO). Performing GSEA in the context of imaging transcriptomics is associated with methodological challenges that the application of the same algorithms in other circumstances do not necessarily raise, mainly, within-category gene–gene co-expression and spatial autocorrelation are now known to drive false-positive bias, which requires particular attention in the way it is dealt with (for further information on this topic, please see Fulcher et al., 2021). In this toolbox, we implement the recently introduced ensemble GSEA framework, which overcomes false-positive gene-category enrichment in the analysis of spatially resolved transcriptomic brain atlas data, using a pre-ranked approach. These analyses are implemented through the imt_gsea script, which requires the .pkl file generated as a result of the previous step.

> *Note:* This step can be run as part of a single command as explained above; this is equivalent to omitting the –no-gsea flag and specifying the input –geneset in the previous script.

4. Define which gene set you want to use for the analysis; as an example, we will use the "Lake" brain cell-type gene set included with the toolbox (this set includes genes expressed in 30 brain cell-types as identified in a previous single-cell transcriptomic study (Lake et al., 2018)). Other available gene sets are provided and can be searched by running the command:

```
> imt_gsea -geneset avail
```

> *Note:* The toolbox offers the users the possibility to select their own gene set file; this should nevertheless be in a compatible format, i.e., *gmt* (see here the instructions on how to create your own gene set file https://www.gsea-msigdb.org/gsea/doc/GSEAUserGuideFrame.html). If the users decide to use their own genes, the file must be provided as an absolute path as the argument of the –geneset flag.

5. Run the ensemble GSEA using the command:

```
> imt_gsea -input /path_to_yourfile/file.pkl -geneset lake
```

The imt_gsea script accepts the following arguments:

a. –input: path to the .pkl file generated by the previous step.
b. –output (optional): path where the results will be saved; if none is provided, the parent directory of the input file will be used instead.
c. –geneset: name of the gene set to be used in the analysis.

> *Note:* Depending on the gene set and analysis used, the GSEA will take longer to run, i.e., running the ensemble GSEA on an analysis with 2 PLS components will take twice the amount of time as running GSEA on an analysis with 1 PLS component.

With the results from either step 2 we can run the GSEA analysis by running the command (similar for the results from step 3):

```
> imt_gsea -input Imt_5-HT2A_mean_bmax_pls/pls_analysis.pkl -geneset lake
```

To run the GSEA analysis using the lake gene set.

6. Check the results in the folder where the .pkl file was stored if no output path was specified.

The interpretation of the ensemble GSEA output does not differ much from the standard GSEA analysis. The primary result of the gene set enrichment analysis is the enrichment score (ES), which reflects the degree to which a gene set is overrepresented at the top (positive score) or bottom (negative score) of a ranked list of genes. Significant enrichment is identified by p-values, corrected for

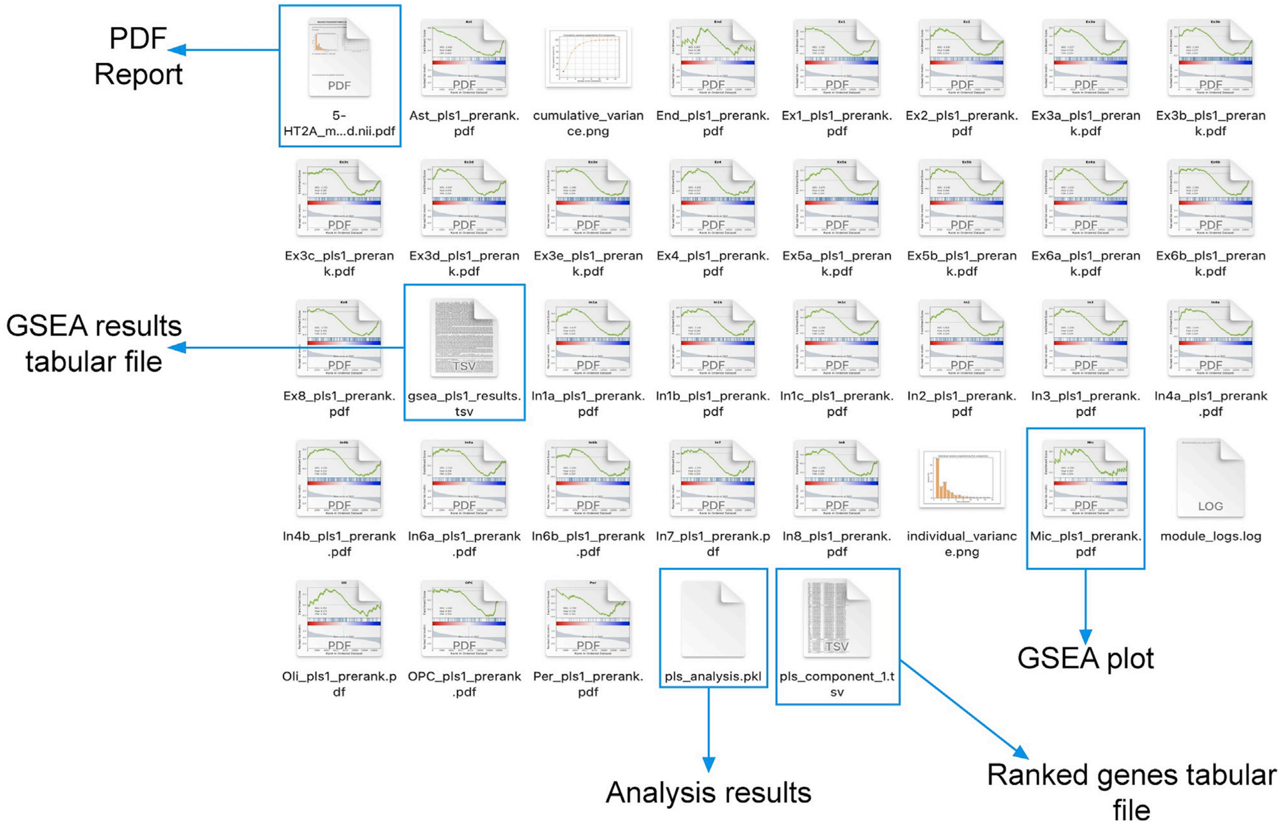


Figure 1. Example of the structure of an output folder

The folder includes tabular files with the results of both the correlation analysis and the GSEA analysis, plots for the variance explained by each component in case of a PLS analysis and enrichment plots for the results from the GSEA analysis.

multiple comparisons, less than 0.05 (i.e., $p_{FDR}<0.05$). In ensemble GSEA, this means that the enrichment observed is higher than one would expect for a null neuroimaging phenotype with the same embedded spatial autocorrelation.

EXPECTED OUTCOMES

Once the analysis is completed, the toolbox creates a folder where the output files are stored (a typical example can be seen in Figure 1). The output files can be summarized as 1) a tabular file (i.e., pls_component_1.tsv Figure 1) with the results from the correlation analysis containing a list of ranked genes, the coefficient of correlation (z-score in the case of PLS analysis) and the uncorrected and FDR-corrected p values (Figure 3). Note that in the case of a PLS regression analysis, a different file is created for each of the PLS components. 2) A pkl file which contains null ranked list of genes to be used for a different enrichment analysis without having to re-run the entire analysis (i.e., pls_analysis.pkl, Figure 1); 3) A PDF file with a report of the analysis performed. In the case of the PLS analysis, the PDF will include plots of the individual and cumulative variance explained by the first 15 components, alongside with the R^2 and p value for each of the components used in the analysis (plots are also available in the output folder as graphics, i.e., *cumulative_variance.png* and *individual_variance.png*, Figure 2). 4) A tabular file with the GSEA results (*gsea_pls1_results* containing the term of the gene set, enrichment score (ES) and normalized enrichment score (NES) scores, uncorrected and FDR-corrected p-values, the size of the gene set term, the number of matched genes, the list of all matched genes and the list of edge genes (i.e., genes contributing the most to the enrichment signal) (Figure 4). 5) Enrichment plots for each individual term of the gene set used (all the files terminating in *_prerank.pdf*, Figure 5).

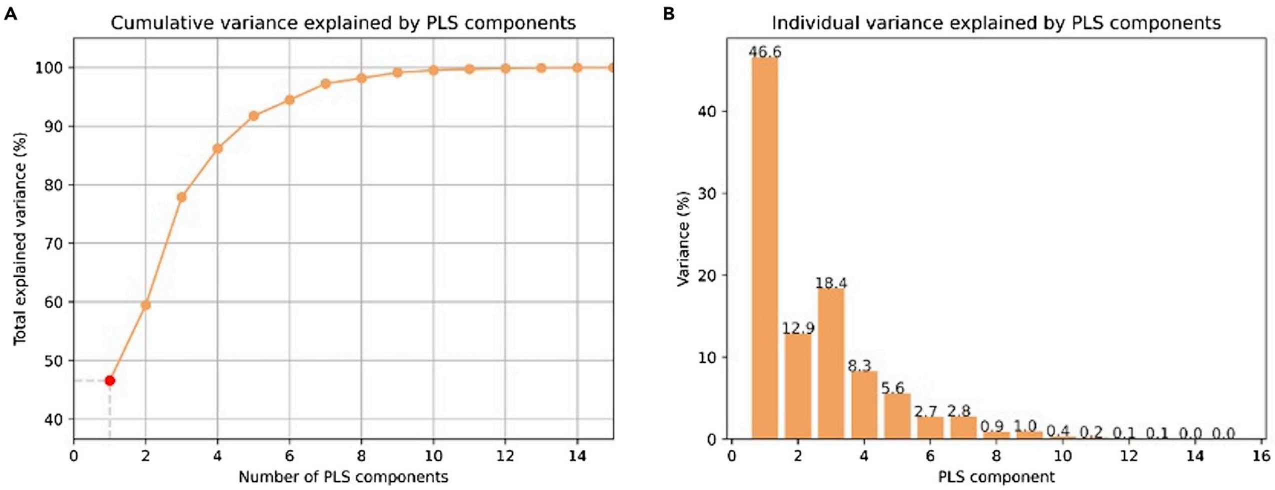


Figure 2. Example of variance plots produced in case of a PLS analysis

(A) Cumulative variance explained by different PLS models with increasing number of components; (B) Individual variance explained by each of the first 15 components.

LIMITATIONS

One of the main limitations of the current implementation of the toolbox is the lack of flexibility regarding the parcellation used to map the neuroimaging and gene expression data. We setup the pipeline to use a standard and widely used parcellation (DK atlas), which provides a fairly coarse coverage of cortical and subcortical regions. However, we acknowledge that specific research questions might require other parcellations, which for now are not readily available to the user. In that case, the user can modify the original code to use other parcellations, but the gene expression matrix will have to be recalculated (e.g., by using *abagen* to remap gene expression to the parcellation chosen by the user). Moreover, all the analyses are based on data from the left hemisphere because the AHBA includes gene expression data of the right hemisphere for two donors only. While a general limitation of the field and not of this specific work, this aspect might raise issues when a certain neuroimaging phenotype is strongly lateralized to the right hemisphere.

TROUBLESHOOTING

Problem 1

The toolbox is installed but the scripts fail to launch from the command line (before you begin step 3).

Potential solution

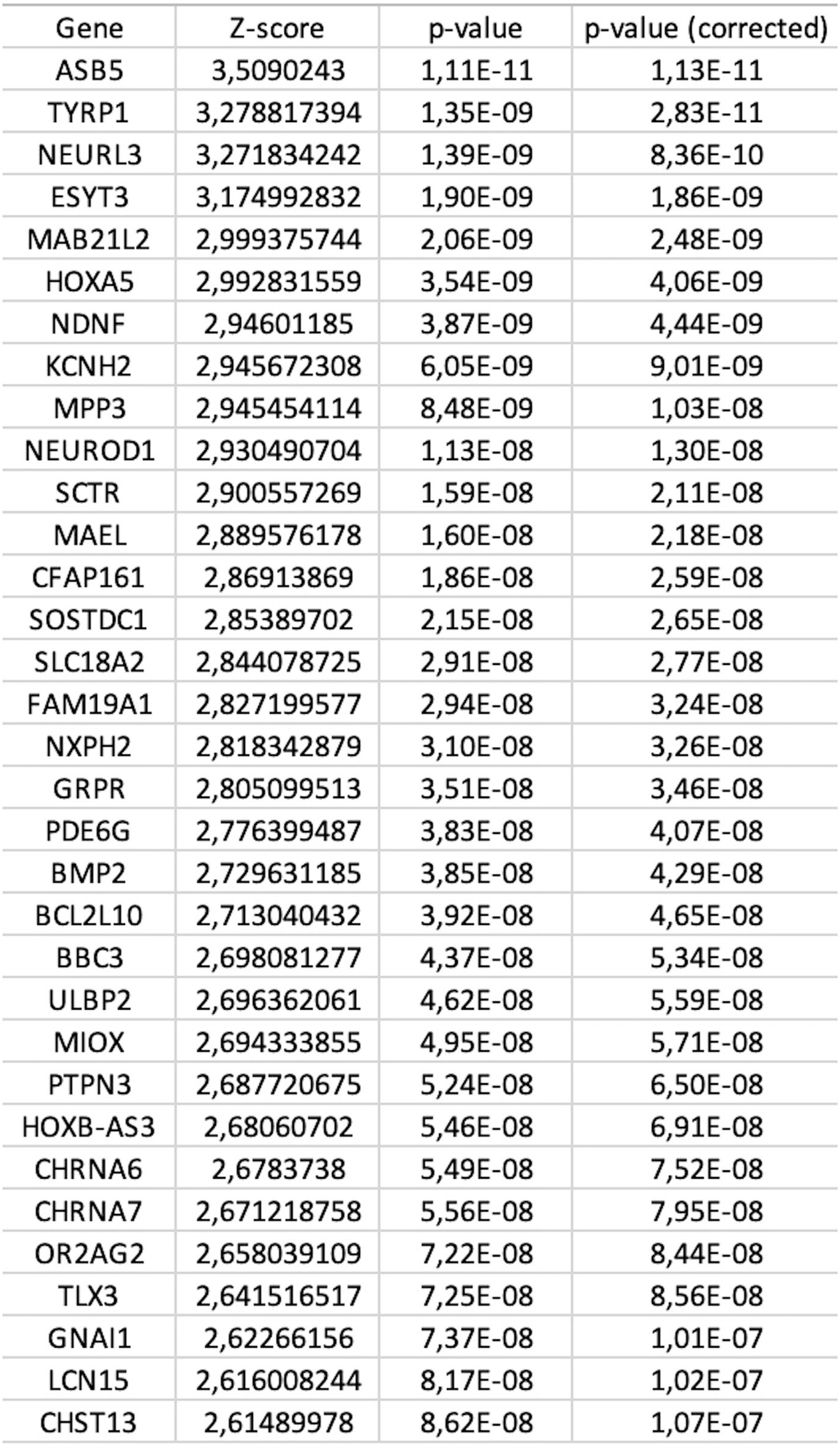
Make sure that the virtual environment where you have installed the toolbox is activated. This can be seen in the terminal or by typing the command:

```
> which python
```

From the command line.

Problem 2

When running the command on an existing image I get an InvalidSizeError, e.g., *imaging_transcriptomics.errors.InvalidSizeError: The provided file has a wrong shape. The file has shape: (182, 218, 182, 1).* (step-by-step method details steps 2 or 3).



Gene	Z-score	p-value	p-value (corrected)
ASB5	3,5090243	1,11E-11	1,13E-11
TYRP1	3,278817394	1,35E-09	2,83E-11
NEURL3	3,271834242	1,39E-09	8,36E-10
ESYT3	3,174992832	1,90E-09	1,86E-09
MAB21L2	2,999375744	2,06E-09	2,48E-09
HOXA5	2,992831559	3,54E-09	4,06E-09
NDNF	2,94601185	3,87E-09	4,44E-09
KCNH2	2,945672308	6,05E-09	9,01E-09
MPP3	2,945454114	8,48E-09	1,03E-08
NEUROD1	2,930490704	1,13E-08	1,30E-08
SCTR	2,900557269	1,59E-08	2,11E-08
MAEL	2,889576178	1,60E-08	2,18E-08
CFAP161	2,86913869	1,86E-08	2,59E-08
SOSTDC1	2,85389702	2,15E-08	2,65E-08
SLC18A2	2,844078725	2,91E-08	2,77E-08
FAM19A1	2,827199577	2,94E-08	3,24E-08
NXPH2	2,818342879	3,10E-08	3,26E-08
GRPR	2,805099513	3,51E-08	3,46E-08
PDE6G	2,776399487	3,83E-08	4,07E-08
BMP2	2,729631185	3,85E-08	4,29E-08
BCL2L10	2,713040432	3,92E-08	4,65E-08
BBC3	2,698081277	4,37E-08	5,34E-08
ULBP2	2,696362061	4,62E-08	5,59E-08
MIOX	2,694333855	4,95E-08	5,71E-08
PTPN3	2,687720675	5,24E-08	6,50E-08
HOXB-AS3	2,68060702	5,46E-08	6,91E-08
CHRNA6	2,6783738	5,49E-08	7,52E-08
CHRNA7	2,671218758	5,56E-08	7,95E-08
OR2AG2	2,658039109	7,22E-08	8,44E-08
TLX3	2,641516517	7,25E-08	8,56E-08
GNAI1	2,62266156	7,37E-08	1,01E-07
LCN15	2,616008244	8,17E-08	1,02E-07
CHST13	2,61489978	8,62E-08	1,07E-07

Figure 3. Example of tabular file containing the results of gene ranking according to alignment with neuroimaging phenotype

The toolbox outputs a tabular file containing: i) gene ID, ranked according to strength of correlation; ii) z-score of gene weight in PLS component (or Spearman's coefficient in case of mass-univariate correlation analysis); and iii) uncorrected and FDR corrected p values for each gene.

Potential solution

Reshape the image to match the correct input shape. In the case of 4D scans, select the image you want to analyze (i.e., a single volume or their average).

Problem 3

After the installation, the program fails to run because of a *ModuleNotFoundError: No module named 'sklearn.datasets.base'*. (step-by-step method details step 2).

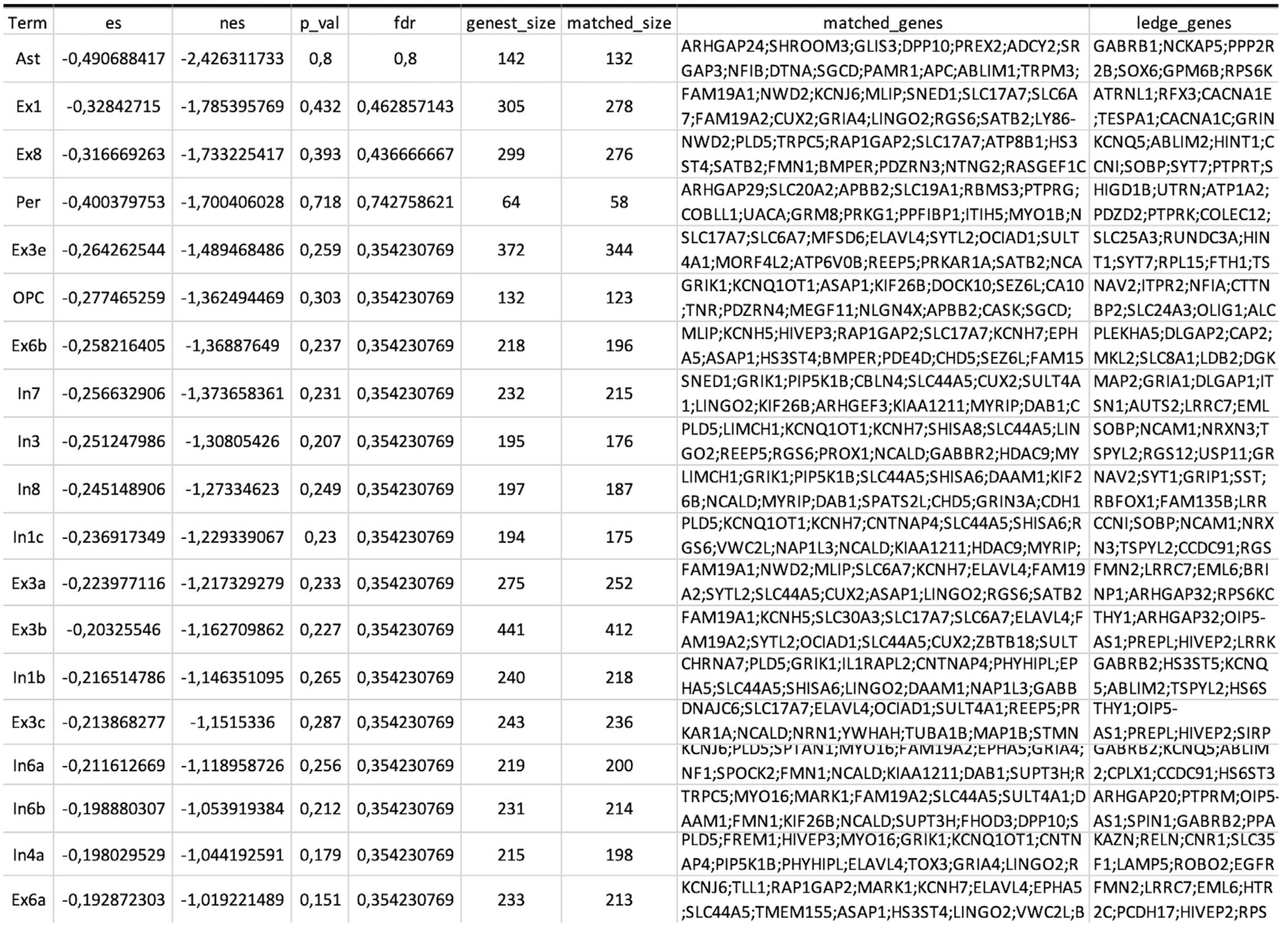
Potential solution

Re-install the sklearn python dependency by running the command:

```
> pip uninstall sklearn
```

Followed by the command:

```
> pip install sklearn
```



Term	es	nes	p_val	fdr	genest_size	matched_size	matched_genes	ledge_genes
Ast	-0,490688417	-2,426311733	0,8	0,8	142	132	ARHGAP24;SHROOM3;GLIS3;DPP10;PREX2;ADCY2;SRGAP3;NFIB;DTNA;SGCD;PAMR1;APC;ABLIM1;TRPM3;	GABRB1;NCKAP5;PPP2R2B;SOX6;GPM6B;RPS6K
Ex1	-0,32842715	-1,785395769	0,432	0,462857143	305	278	FAM19A1;NWD2;KCNJ6;MLIP;SNED1;SLC17A7;SLC6A7;FAM19A2;CUX2;GRIA4;LINGO2;RGS6;SATB2;LY86-	ATRNL1;RFX3;CACNA1E;TESPA1;CACNA1C;GRIN
Ex8	-0,316669263	-1,733225417	0,393	0,436666667	299	276	NWD2;PLD5;TRPC5;RAP1GAP2;SLC17A7;ATP8B1;HS3ST4;SATB2;FMN1;BMPER;PDZRN3;NTNG2;RASGEF1C	KCNQ5;ABLIM2;HINT1;CCNI;SOBP;SYT7;PTPRT;S
Per	-0,400379753	-1,700406028	0,718	0,742758621	64	58	ARHGAP29;SLC20A2;APBB2;SLC19A1;RBMS3;PTPRG;COBLL1;UACA;GRM8;PRKG1;PPFIBP1;ITIH5;MYO1B;N	HIGD1B;UTRN;ATP1A2;PDZD2;PTPRK;COLEC12;
Ex3e	-0,264262544	-1,489468486	0,259	0,354230769	372	344	SLC17A7;SLC6A7;MFSD6;ELAVL4;SYTL2;OCIAD1;SULT4A1;MORF4L2;ATP6V0B;REEP5;PRKAR1A;SATB2;NCA	SLC25A3;RUNDC3A;HINT1;SYT7;RPL15;FTH1;TS
OPC	-0,277465259	-1,362494469	0,303	0,354230769	132	123	GRIK1;KCNQ1OT1;ASAP1;KIF26B;DOCK10;SEZ6L;CA10;TNR;PDZRN4;MEGF11;NLGN4X;APBB2;CASK;SGCD;	NAV2;ITPR2;NFIA;CTTNBP2;SLC24A3;OLIG1;ALC
Ex6b	-0,258216405	-1,36887649	0,237	0,354230769	218	196	MLIP;KCNH5;HIVEP3;RAP1GAP2;SLC17A7;KCNH7;EPHA5;ASAP1;HS3ST4;BMPER;PDE4D;CHD5;SEZ6L;FAM15	PLEKHA5;DLGAP2;CAP2;MKL2;SLC8A1;LDB2;DGK
In7	-0,256632906	-1,373658361	0,231	0,354230769	232	215	SNED1;GRIK1;PIP5K1B;CBLN4;SLC44A5;CUX2;SULT4A1;LINGO2;KIF26B;ARHGEF3;KIAA1211;MYRIP;DAB1;C	MAP2;GRIA1;DLGAP1;ITSN1;AUTS2;LRRC7;EML
In3	-0,251247986	-1,30805426	0,207	0,354230769	195	176	PLD5;LIMCH1;KCNQ1OT1;KCNH7;SHISA8;SLC44A5;LINGO2;REEP5;RGS6;PROX1;NCALD;GABBR2;HDAC9;MY	SOBP;NCAM1;NRXN3;TSPYL2;RGS12;USP11;GR
In8	-0,245148906	-1,27334623	0,249	0,354230769	197	187	LIMCH1;GRIK1;PIP5K1B;SLC44A5;SHISA6;DAAM1;KIF26B;NCALD;MYRIP;DAB1;SPATS2L;CHD5;GRIN3A;CDH1	NAV2;SYT1;GRIP1;SST;RBFOX1;FAM135B;LRR
In1c	-0,236917349	-1,229339067	0,23	0,354230769	194	175	PLD5;KCNQ1OT1;KCNH7;CNTNAP4;SLC44A5;SHISA6;RGS6;VWC2L;NAP1L3;NCALD;KIAA1211;HDAC9;MYRIP;	CCNI;SOBP;NCAM1;NRXN3;TSPYL2;CCDC91;RGS
Ex3a	-0,223977116	-1,217329279	0,233	0,354230769	275	252	FAM19A1;NWD2;MLIP;SLC6A7;KCNH7;ELAVL4;FAM19A2;SYTL2;SLC44A5;CUX2;ASAP1;LINGO2;RGS6;SATB2	FMN2;LRRC7;EML6;BRINP1;ARHGAP32;RPS6KC
Ex3b	-0,20325546	-1,162709862	0,227	0,354230769	441	412	FAM19A1;KCNH5;SLC30A3;SLC17A7;SLC6A7;ELAVL4;FAM19A2;SYTL2;OCIAD1;SLC44A5;CUX2;ZBTB18;SULT	THY1;ARHGAP32;OIP5-AS1;PREPL;HIVEP2;LRRK
In1b	-0,216514786	-1,146351095	0,265	0,354230769	240	218	CHRNA7;PLD5;GRIK1;IL1RAPL2;CNTNAP4;PHYHIPL;EPHA5;SLC44A5;SHISA6;LINGO2;DAAM1;NAP1L3;GABB	GABRB2;HS3ST5;KCNQ5;ABLIM2;TSPYL2;HS6S
Ex3c	-0,213868277	-1,1515336	0,287	0,354230769	243	236	DNAJC6;SLC17A7;ELAVL4;OCIAD1;SULT4A1;REEP5;PRKAR1A;NCALD;NRN1;YWHAH;TUBA1B;MAP1B;STMN	THY1;OIP5-AS1;PREPL;HIVEP2;SIRP
In6a	-0,211612669	-1,118958726	0,256	0,354230769	219	200	KCNJ6;PLD5;SPTAN1;MYO16;FAM19A2;EPHA5;GRIA4;NF1;SPOCK2;FMN1;NCALD;KIAA1211;DAB1;SUPT3H;R	GABRB2;KCNQ5;ABLIM2;CPLX1;CCDC91;HS6ST3
In6b	-0,198880307	-1,053919384	0,212	0,354230769	231	214	TRPC5;MYO16;MARK1;FAM19A2;SLC44A5;SULT4A1;DAAM1;FMN1;KIF26B;NCALD;SUPT3H;FHOD3;DPP10;S	ARHGAP20;PTPRM;OIP5-AS1;SPIN1;GABRB2;PPA
In4a	-0,198029529	-1,044192591	0,179	0,354230769	215	198	PLD5;FREM1;HIVEP3;MYO16;GRIK1;KCNQ1OT1;CNTNAP4;PIP5K1B;PHYHIPL;ELAVL4;TOX3;GRIA4;LINGO2;R	KAZN;RELN;CNR1;SLC35F1;LAMP5;ROBO2;EGFR
Ex6a	-0,192872303	-1,019221489	0,151	0,354230769	233	213	KCNJ6;TLL1;RAP1GAP2;MARK1;KCNH7;ELAVL4;EPHA5;SLC44A5;TMEM155;ASAP1;HS3ST4;LINGO2;VWC2L;B	FMN2;LRRC7;EML6;HTR2C;PCDH17;HIVEP2;RPS

Figure 4. Example of tabular file with the results from the GSEA analysis

The tabular file contains data about the enrichment score (ES), normalized enrichment score (NES), uncorrected p value (p_val), FDR corrected p value (fdr), number of genes in the gene set term (geneset_size), number of matched genes from the correlation results (matched_size), label of the matched genes (matched_genes) and ledge genes (ledge_genes) for each of the terms included in a certain gene set.

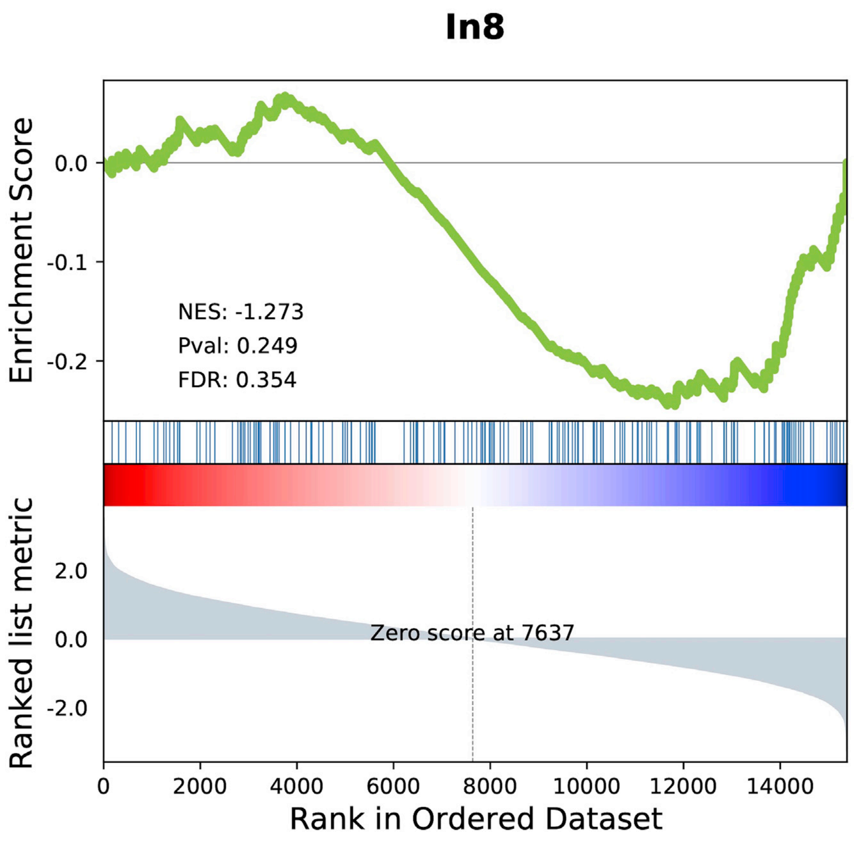


Figure 5. Example of an enrichment plot from the GSEA analysis

The analysis produces a plot for each term of the gene set used. The top portion of the plot shows the running enrichment score (ES) for the gene set as the analysis walks down the ranked list. The score at the peak of the plot (the score furthest from 0.0) is the ES for the gene set. The middle portion of the plot shows where the members of the gene set appear in the ranked list of genes. The bottom portion of the plot shows the value of the ranking metric as you move down the list of ranked genes. The ranking metric measures a gene's correlation with a phenotype. The value of the ranking metric goes from positive to negative as you move down the ranked list.

Problem 4

The installation of the ENIGMA Toolbox fails, raising some errors (before you begin step 4).

Potential solution

To overcome the errors in the ENIGMA Toolbox installation you need to manually install some python libraries which the toolbox depends on (e.g., Cython and NumPy). In addition you can look at specific versions of the packages listed on the environment file (i.e., https://github.com/molecular-neuroimaging/Imaging_Transcriptomics.git).

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Dr. Daniel Martins (daniel.martins@kcl.ac.uk).

Materials availability

This study did not generate new unique reagents.

Data and code availability

The Imaging Transcriptomics toolbox is available on GitHub at https://github.com/molecular-neuroimaging/Imaging_Transcriptomics (Giacomel et al., 2022). The data used as example in the protocol are available in the Neurobiology Research Unit's website https://xtra.nru.dk/FS5ht-atlas/.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.xpro.2022.101315.

ACKNOWLEDGMENTS

D.M., O.D., F.T., and S.C.R.M. are supported by the NIHR Maudsley's Biomedical Research Centre at the South London and Maudsley NHS Trust. MV is supported by MIUR, Italian Ministry for Education, under the initiatives "Departments of Excellence" (Law 232/2016) and by the National Institute for Health Research Biomedical Research Centre at South London and Maudsley National Health Service Foundation Trust and King's College London. A.G. is supported by the KCL-funded CDT in Data-Driven Health; this represents independent research partly funded by the NIHR Maudsley's Biomedical Research Center at the South London and Maudsley NHS Trust and partly funded by GlaxoSmithKline (GSK).

AUTHOR CONTRIBUTIONS

A.G. wrote the python script and drafted the protocol; D.M. led the conceptual design of the toolbox and drafted the protocol; O.D., M.V., M.F., F.T., and S.C.R.W. revised the protocol for intellectual content. All authors approved the final version of the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests. This manuscript represents independent research.

REFERENCES

Alexander-Bloch, A., Giedd, J.N., and Bullmore, E. (2013a). Imaging structural co-variance between human brain regions. Nat. Rev. Neurosci. *14*, 322–336. https://doi.org/10.1038/nrn3465.

Alexander-Bloch, A., Raznahan, A., Bullmore, E., and Giedd, J. (2013b). The convergence of maturational change and structural covariance in human cortical networks. J. Neurosci. *33*, 2889–2899. https://doi.org/10.1523/JNEUROSCI.3554-12.2013.

Arnatkevičiūtė, A., Fulcher, B.D., and Fornito, A. (2019). A practical guide to linking brain-wide gene expression and neuroimaging data. Neuroimage *189*, 353–367. https://doi.org/10.1016/j.neuroimage.2019.01.011.

Beliveau, V., Ganz, M., Feng, L., Ozenne, B., Højgaard, L., Fisher, P.M., Svarer, C., Greve, D.N., and Knudsen, G.M. (2017). A high-resolution in vivo atlas of the human brain's serotonin system. J. Neurosci. *37*, 120–128. https://doi.org/10.1523/JNEUROSCI.2830-16.2016.

Desikan, R.S., Ségonne, F., Fischl, B., Quinn, B.T., Dickerson, B.C., Blacker, D., Buckner, R.L., Dale, A.M., Maguire, R.P., Hyman, B.T., et al. (2006). An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. Neuroimage *31*, 968–980. https://doi.org/10.1016/j.neuroimage.2006.01.021.

Fang, Z. (2020). GSEApy: Gene Set Enrichment Analysis in Python. Zenodo. https://doi.org/10.5281/zenodo.3748085.

Fulcher, B.D., Arnatkeviciute, A., and Fornito, A. (2021). Overcoming false-positive gene-category enrichment in the analysis of spatially resolved transcriptomic brain atlas data. Nat. Commun. *12*, 2669. https://doi.org/10.1038/s41467-021-22862-1.

Giacomel, A., Martins, D., Frigo, M., Turkheimer, F., Williams, S.C.R., Dipasquale, O., and Veronese, M. (2022). The Imaging Transcriptomics Toolbox. Zenodo. https://doi.org/10.5281/zenodo.6364963.

Gorgolewski, K.J., Fox, A.S., Chang, L., Schäfer, A., Arélin, K., Burmann, I., Sacher, J., and Margulies, D.S. (2014). Tight fitting genes: finding relations between statistical maps and gene expression patterns. https://doi.org/10.7490/f1000research.1097120.1.

Hawrylycz, M.J., Lein, E.S., Guillozet-Bongaarts, A.L., Shen, E.H., Ng, L., Miller, J.A., van de Lagemaat, L.N., Smith, K.A., Ebbert, A., Riley, Z.L., et al. (2012). An anatomically comprehensive atlas of the adult human brain transcriptome. Nature *489*, 391–399. https://doi.org/10.1038/nature11405.

Jenkinson, M., Beckmann, C.F., Behrens, T.E.J., Woolrich, M.W., and Smith, S.M. (2012). FSL. Neuroimage *62*, 782–790. 20 YEARS of fMRI. https://doi.org/10.1016/j.neuroimage.2011.09.015.

Lake, B.B., Chen, S., Sos, B.C., Fan, J., Kaeser, G.E., Yung, Y.C., Duong, T.E., Gao, D., Chun, J., Kharchenko, P.V., and Zhang, K. (2018). Integrative single-cell analysis of transcriptional and epigenetic states in the human adult brain. Nat. Biotechnol. *36*, 70–80. https://doi.org/10.1038/nbt.4038.

Larivière, S., Paquola, C., Park, B., Royer, J., Wang, Y., Benkarim, O., Vos de Wael, R., Valk, S.L., Thomopoulos, S.I., Kirschner, M., et al. (2021). The ENIGMA toolbox: multiscale neural contextualization of multisite neuroimaging datasets. Nat. Methods *18*, 698–700. https://doi.org/10.1038/s41592-021-01186-4.

Markello, R.D., Arnatkeviciute, A., Poline, J.-B., Fulcher, B.D., Fornito, A., and Misic, B. (2021). Standardizing workflows in imaging transcriptomics with the abagen toolbox. eLife *10*, e72129. https://doi.org/10.7554/eLife.72129.

Markello, R.D., and Misic, B. (2021). Comparing spatial null models for brain maps. Neuroimage *236*, 118052. https://doi.org/10.1016/j.neuroimage.2021.118052.

Martins, D., Giacomel, A., Williams, S., Turkheimer, F., Dipasquale, O., and Veronese, M. (2021). Imaging transcriptomics: convergent cellular, transcriptomic, and molecular neuroimaging signatures in the healthy adult human brain. Cell Rep *37*, 110173. https://doi.org/10.1016/j.celrep.2021.110173.

MATLAB (2020). version 9.9.0 (R2020b) (Natick, Massachusetts: The MathWorks Inc).

Morgan, S.E., Seidlitz, J., Whitaker, K.J., Romero-Garcia, R., Clifton, N.E., Scarpazza, C., van Amelsvoort, T., Marcelis, M., van Os, J., Donohoe, G., et al. (2019). Cortical patterning of abnormal morphometric similarity in psychosis is associated with brain expression of schizophrenia-related genes. Proc. Natl. Acad. Sci. U S A *116*, 9604–9609. https://doi.org/10.1073/pnas.1820754116.

Smith, S.M., Jenkinson, M., Woolrich, M.W., Beckmann, C.F., Behrens, T.E.J., Johansen-Berg, H., Bannister, P.R., De Luca, M., Drobnjak, I., Flitney, D.E., et al. (2004). Advances in functional and structural MR image analysis and implementation as FSL. Neuroimage *23*, S208–S219. Mathematics Brain Imaging. https://doi.org/10.1016/j.neuroimage.2004.07.051.

Váša, F., Seidlitz, J., Romero-Garcia, R., Whitaker, K.J., Rosenthal, G., Vértes, P.E., Shinn, M., Alexander-Bloch, A., Fonagy, P., Dolan, R.J., et al. (2018). Adolescent tuning of association cortex in human structural brain networks. Cereb. Cortex *28*, 281–294. https://doi.org/10.1093/cercor/bhx249.

Woolrich, M.W., Jbabdi, S., Patenaude, B., Chappell, M., Makni, S., Behrens, T., Beckmann, C., Jenkinson, M., and Smith, S.M. (2009). Bayesian analysis of neuroimaging data in FSL. Neuroimage *45*, S173–S186. Mathematics Brain Imaging. https://doi.org/10.1016/j.neuroimage.2008.10.055.