**Software paper for submission to MEDSCI 736 University of Auckland**

To complete this template, please replace the blue text with your own. The paper has three main sections: (1) Overview; (2) Availability; (3) Reuse potential.

Please submit the completed paper to: f.kubke@auckland.ac.nz

(1) Overview

Project IE - An analysis tool for guinea-pig cochlear MRI data

Paper Authors

*Lee, Sang Eun (Bibiana)*

Paper Author Roles and Affiliations

*Development Team, Faculty of Medical and Health Sciences UoA*

Abstract

Project IE is a python supported module which parses, stores and analyses MRI data (text files) obtained through the imaging of guinea pig cochlear from an ongoing replication project. This prototype currently allows the user to input one raw file at a time to be tidied and passed through pandas DataFrame for easy data management. The resulting DataFrames also enable further processing in the form of a one-way ANOVA analysis and can alternatively output a graph which is then saved as an image (.png).

This module is licensed under the MIT license and is available for modification and use at:

<https://github.com/UOA-MEDSCI-736/BibianaLee-medsci736>

Keywords

*cochlear inflammation; lipopolysaccharide; MRI; vascular permeability; ultrasmall superparamag-netic iron oxide; blood-labyrinth barrier; Python; Software; Statistical Analysis; Pandas Library*

Introduction

Deafness and hearing loss is increasingly becoming a topical public health issue. Damage or disease to the cochlear (an auditory sensory organ) can result in hearing loss and affects around 538 million people worldwide (1). The increasing burden of hearing impairment is setting the precedence for prevention and treatment options.

The inner ear (IE) is a mostly fluids based organ surrounded by a blood barrier. The cochlea which is a spiral shaped cavity and is the auditory portion of the inner ear is associated with hearing loss. The degeneration of the cochlear can be caused by a plethora of reasons such as injury, ageing, noise exposure, trauma, infection or genetic abnormalities (2).

Recent research suggests that inflammation of the middle ear or inner ear is a major contributing factor to cochlear degeneration. Although, there is scarce direct evidence to link these two factors in the human ear, there is evidence of this relationship in animal models. In these studies treatment with steroids have been shown to mitigate injury and hearing loss. A current study replicating another project previously conducted at the University of Auckland is looking to understand the inflammatory processes in the inner ear. The investigation of inflammatory responses may help in the diagnosis of cochlear injury. This could in turn aid in the development of treatments which target and reduce inflammation in the inner ear, thereby reducing the risk of hearing loss (2,3).

Project IE was designed as a prototype to assist a team conducting the ongoing replication study at the University of Auckland. This study looks to quantify spatial and temporal inflammation-induced changes in the capsular permeability and macrophage infiltration in guinea-pig cochlea using MRI data. This study could provide reference data that can quantitatively assess the treatment of ear disease in animal models and establish a platform from which such techniques can be transferred into clinical practice (2).However, the modelling of such exchanges in blood and different inner ear compartments require the analysis of a substantial amount of data.

Currently, there are a number of software packages that can be used to both parse and analyze large datasets. Software such as R, MATLAB, SAS and even Excel are commonplace in scientific communities. These data analysis packages can be difficult to use and require familiarity with the programme before any extensive analysis can be completed (4). Furthermore, a lot of the packages are not open-source and therefore do not have a high reuse potential outside of their respective licenses permit.

Project IE utilizes python, an open source programming language that is easy, distributable and modifiable. This programme is a free open-source prototype which enables the parsing and analysis of MRI text data obtained through this replication study.

**Implementation and architecture**

Project IE was designed as a prototype to assist a team conducting an ongoing replication study at the University of Auckland. This software utilizes python 2.7.12, Pandas version u'0.18.1 (an open source data analysis library), SciPy library and Matplotlib in order to parse and analyze a single MRI text file.

MRI files were obtained through the University of Auckland and are around 7 kilobytes in size and in total there are 200 files making the dataset around 1.4 gigabytes.

Each file contained the unstructured or "raw" values for inputs (time after fixed captures in seconds and arterial input function values) and outputs (K trans value, pixel enhancement averages along time, pixel enhancement STD value along time and values for the diffusion coefficient).

The general implementation of Project IE is illustrated in *Figure 1.* Project IE was designed to be a command line programme and incorporates a simple workflow which splits into three arms after parsing.

The user is welcomed to the interface and asked to give the name of a single file (which should be located in the same folder as the script). Documentation on how to run the programme can be found in the readme file.



Figure 1: Project IE implementation diagram

ARM 1

ARM 2

ARM 3

All filenames contained a number followed by a 'D' for 'Day' and another number. The first number referred to the number of images taken of the animal and the second number referred to the number of days after treatment or non-treatment.

The values in the 'raw' file consist of strings which begin with the specific string of characters 'local-cochlea-bm.mha'. These characters were used to identify a new line and the beginning of a new label.

Figure 2: Label Hierarchy

Each label contained the string of characters which were then followed by an output label, ID1 and ID2 which were separated by underscores ('\_'). The first ID referring to either the right ear (ID tag: 294) or left ear (ID tag: 295) and finally the height of the inner ear being analyzed which ranged from 0 to 7.

*"local-cochlea-bm.mha\_ktrans\_294\_0"*

All files contained a three level label hierarchy system for outputs which is illustrated in **Figure 2** and a single level one for inputs.

Following user input of the file, the initial clean up consisted of a code which separated values via the delimiter ",", removed blank lines and strings which started with’#'. Labels were then identified using Python's *str.find* method and matching strings were grouped into lists corresponding to one of the outputs or input labels.

Due to the hierarchical nature of the labels, pandas an open source data analysis tool was utilized to multi-index, sort, structure and manage labels and values. Each list was passed through pandas Dataframe, a two-dimensional tabular data structure. Once all the data had been passed through pandas DataFrame, the values are converted into floats to enable analysis of the data. The programme will then print the DataFrames as an output and these can be viewed through command line.

After the DataFrames have printed, the user is asked for input again. At this point the code splits into three separate arms. One arm allows the user to simply quit the module, another one enables visualization of the pixel enhancement averages along time as a graph and the final arm give the option to statistically analyze the enhancement averages using a one-way ANOVA.

Visualization of the project data was achieved using a plotting library called Matplotlib. The one-way ANOVA analysis was accomplished using the SciPy library in conjunction with the Pandas DataFrame.

After a complete run of the programme the user will have two different outputs depending on which arm they chose. If they choose ARM 1, then they will have the all the pandas DataFrames on their screen as well as an image (.png) of their graph saved in their folder. If they chose the ARM 2 they will have all the DataFrames and a one-way ANOVA analysis with a p value.

Aesthetic functionality was achieved through the use of blank lines and pandas DataFrame. The DataFrame gives a clear overview of the data and a functional multi-indexing system.

**Quality control**

Quality control was achieved primarily through manual testing of the functions with some unit tests to verify that input functions were working. Unit testing was done retrospectively after the completion of the programme. Unit tests can be run using the Test\_Project\_IE script and running pytest. Users can manual test using the dummy data which can be found in the Test\_Project\_IE folder and comparing the results to the graph, text file and which can also be found in the same folder. Instructions on how to run tests can be found in the readme file.

**(2) Availability**

***Operating system***

Ubuntu 16.04 LTS

***Programming language***

Python 2.7.12

***Additional system requirements***

N/A

***Dependencies***

Pandas version : u'0.18.1

Matplotlib 1.5.3

SciPy 0.18.1

***List of contributors***

Pireva, Kreshnik, staff, Faculty of Medical and Health Sciences UoA

Callum Chalmers, Student, Faculty of Medical and Health Sciences UoA

Plumat, Jerome, Research Fellow, Student, Faculty of Medical and Health Sciences UoA

***Software location:***

***Archive***

***Name:*** Figshare

***Persistent identifier:*** https://figshare.com/articles/Project\_IE/4093371

***Licence: MIT license***

***Publisher: n/a***

***Version published:*** *1*

***Date published:*** 27/10/2016

**Code repository**

***Name:*** UOA-MEDSCI-736/BibianaLee-medsci736

***Persistent identifier:*** <https://github.com/UOA-MEDSCI-736/BibianaLee-medsci736>

***Licence: MIT license***

***Publisher: Bibiana Lee***

***Version published:*** *1*

***Date published:*** 27/10/2016

***Language***

New Zealand English

**(3) Reuse potential**

Project IE was coded specifically for the MRI data text files resulting from the replication experiment currently conducted by the University of Auckland. However, as the software was scripted using Python using open source libraries the reuse potential is high. Scripting language in Python is flexible and fairly easy to learn. Additionally, python is an open-source platform utilized in the wider scientific community.

The programme also comes with comprehensive documentation on how to run it and install the various programmes it uses. This means that the end-user does not require prior knowledge of scripting language or command line in order successfully run the programme.

However, as the code is specifically for the MRI text files provided by the replication study, it is limited to reading these files unless otherwise edited. As previously discussed, the labelling system in each file is hierarchical and specific to the MRI inputs and outputs for this study. This means that a lot of the structure and indexing is hard coded into the script. This may limit the reuse potential of the code for users who are not familiar with scripting as they cannot modify it. For users who are familiar with scripting language or python, reuse of the code in segments or as whole can be achieved through simple editing of the *str.find* segments and pandas DataFrames.

The software is available at <https://github.com/UOA-MEDSCI-736/BibianaLee-medsci736> under an MIT license. Any and all contributions are welcomed.

**Acknowledgements**

Special thanks to the MEDSCI 736 team and mentor, Kreshnik Pireva all of whom were pivotal in the completion of this project.

**Funding statement**

Vice Chancellor’s Strategic Development Fund (University of Auckland 2015)

**Competing interests**

The authors declare no competing interests

**References**

(1) Olusanya BO, Neumann KJ, Saunders JE. The global burden of disabling hearing impairment: a call to action. Bull World Health Organ 2014;92(5):367-373.

(2) Floc'h JL, Tan W, Telang RS, Vlajkovic SM, Nuttall A, Rooney WD, et al. Markers of cochlear inflammation using MRI. Journal of Magnetic Resonance Imaging 2014;39(1):150-161.

(3) Pakdaman M, Ishiyama G, Ishiyama A, Peng K, Kim H, Pope W, et al. Blood-Labyrinth Barrier Permeability in Menière Disease and Idiopathic Sudden Sensorineural Hearing Loss: Findings on Delayed Postcontrast 3D-FLAIR MRI. Am J Neuroradiol 2016.

(4) O'Connor B. **Comparison of data analysis packages: R, Matlab, SciPy, Excel, SAS, SPSS, Stata**. 2009; Available at: <https://brenocon.com/blog/2009/02/comparison-of-data-analysis-packages-r-matlab-scipy-excel-sas-spss-stata/>.

**Copyright Notice**

Authors who publish with this journal agree to the following terms:

Authors retain copyright with the work simultaneously licensed under a [Creative Commons Attribution License](http://creativecommons.org/licenses/by/3.0/) that allows others to share the work with an acknowledgement of the work's authorship and initial publication in this journal.

Authors are able to enter into separate, additional contractual arrangements for the non-exclusive distribution of the journal's published version of the work (e.g., post it to an institutional repository or publish it in a book), with an acknowledgement of its initial publication in this journal.

By submitting this paper you agree to the terms of this Copyright Notice, which will apply to this submission if and when it is published by this journal.