Software Impacts 16 (2023) 100508

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

Precision-medicine-toolbox: An open-source python package for the quantitative medical image analysis

Elizaveta Lavrova [a,b,*,1], Sergey Primakov [a,1], Zohaib Salahuddin [a], Manon Beuque [a], Damon Verstappen [a], Henry C. Woodruff [a,c], Philippe Lambin [a,c]

[a] The D-Lab, Department of Precision Medicine, GROW—School for Oncology and Reproduction, Maastricht University, Maastricht, The Netherlands
[b] GIGA Cyclotron Research Centre In Vivo Imaging, University of Liège, Liège, Belgium
[c] Department of Radiology and Nuclear Medicine, Maastricht University Medical Centre, Maastricht, The Netherlands

ARTICLE INFO

Keywords:
Medical imaging research
DICOM
Radiomics
Statistical analysis
Features
Image pre-processing

ABSTRACT

Medical image analysis plays a key role in precision medicine. Data curation and pre-processing are critical steps in quantitative medical image analysis that can have a significant impact on the resulting performance of machine learning models. In this work, we introduce the Precision-medicine-toolbox, allowing clinical and junior researchers to perform data curation, image pre-processing, radiomics extraction, and feature exploration tasks with a customizable Python package. With this open-source tool, we aim to facilitate the crucial data preparation and exploration steps, bridge the gap between the currently existing packages, and improve the reproducibility of quantitative medical imaging research.

Code metadata

Current code version	*v0.11*
Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2023-79
Permanent link to Reproducible Capsule	https://codeocean.com/capsule/0396992/tree/v1
Legal Code License	*BSD-3-Clause*
Code versioning system used	*Git*
Software code languages, tools, and services used	*Python*
Compilation requirements, operating environments & dependencies	*numpy 1.16.2, SimpleITK 0.9.1, PyWavelets 0.4.0, pykwalify 1.6.0, six 1.10.0, tqdm 4.40.2, pydicom 1.3.0, pandas 0.25.1, pyradiomics 2.2.0, scikit-image 0.14.2, ipywidgets 7.4.2, matplotlib 3.0.3, Pillow 5.4.1, scikit-learn 0.21.3, scipy 1.2.1, plotly 4.8.1, mkdocstrings 0.18.0, statsmodels 0.12.2, opencv-python 4.1.2.30, seaborn 0.11.1, pickle-mixin 1.0.2, openpyxl 3.0.7*
If available Link to developer documentation/manual	https://precision-medicine-toolbox.readthedocs.io/en/latest/
Support email for questions	e.lavrova@maastrichtuniversity.nl, s.primakov@maastrichtuniversity.nl, h.woodruff@maastrichtuniversity.nl

1. Introduction

Precision medicine (PM) aims to enhance individual patient care by identifying subgroups of patients within a disease group using genotypic and phenotypic data, consequently targeting the disease with more efficient treatment [1]. Medical image analysis plays a key role in PM as it allows the clinicians to non-invasively identify phenotypes [2].

The number of medical imaging data to analyze is rising rapidly. Hence, there is a need for medical image analysis tools that can aid clinicians in meeting the challenges of rising demand and better clinical performance, while reducing variability and costs. At the heart of

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author at: GIGA Cyclotron Research Centre In Vivo Imaging, University of Liège, Liège, Belgium.
E-mail address: e.lavrova@maastrichtuniversity.nl (E. Lavrova).

[1] Equal contribution.

https://doi.org/10.1016/j.simpa.2023.100508
Received 17 February 2023; Received in revised form 19 April 2023; Accepted 23 April 2023

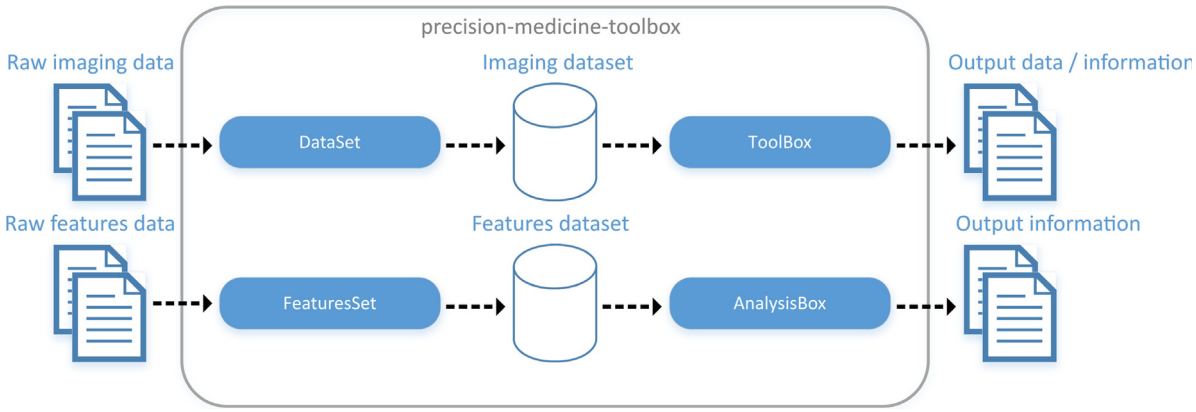

Fig. 1. Organization of the precision-medicine-toolbox: The DataSet class takes an imaging dataset as in input and is inherited by the ToolBox class; the FeatiresSet class takes a features dataset as an input and is inherited by the AnalysisBox class.

these tools will be advanced quantitative imaging analysis, such as handcrafted radiomics and deep learning. Handcrafted radiomics is the high-throughput extraction of pre-defined high-dimensional quantitative image features and their correlation with clinical outcomes using machine learning methods [3]. Deep learning automatically learns representative image features from the high dimensional image data without the need for feature engineering by using non-linear modules that constitute a neural network [4]. The field of quantitative image analysis is expanding [5–7]. Moreover, it has demonstrated promising results in various clinical applications [8–11]. As with many nascent technologies, high-throughput quantitative image analysis suffers from a lack of standardization, e.g. in the image domain (different vendors, acquisition and reconstruction protocols, pre-processing), or different definitions of handcrafted features (such as shape, intensity, and texture features). The spread of widely used open-source software such as Pyradiomics, allows the extraction of standard handcrafted radiomics features [12].

Data curation and the pre-processing of medical images are time-consuming and critical steps in the radiomics workflow that can have a significant impact on the resulting model performance [13–15]. These steps may be performed manually or using lower level python libraries such as Numpy [16], Pandas [17], Pydicom [18], Scikit-image [19], Scikit-learn [20], SimpleITK [21], Nibabel [22], or Scipy [23]. As most current data curation workflows necessitate time-consuming human input, this step becomes an error-prone bottleneck and adds to the current reproducibility problem. Moreover, it is important to perform an exploratory analysis to understand the link between the data used as input in a machine learning model with the outcome it has to predict. While there are tools available for the implementation of the radiomics pipeline such as Nipype [24], Pymia [25], and MONAI [26], there is also the need for a tool that allows for the systematic and standardized data curation, image pre-processing, and feature exploration during the development phase of the study.

We introduce the open-source Precision-medicine-toolbox that facilitates data curation, image pre-processing, and feature exploration using customizable Python scripts.

Implementation and architecture

As illustrated in Fig. 1, dedicated base classes have been implemented for each dataset type to extract the corresponding data, as well as the associated metadata. The functionality classes inherit from the base classes. This approach allows for the separation of reading and processing tasks and makes it readily available for new data formats or functions.

The imaging module allows for pre-processing and exploration of the imaging datasets. It consists of the base DataSet class and the inheriting ToolBox class. The DataSet class reads the imaging data and the corresponding metadata and initializes a dataset object. The ToolBox is an inheriting class that enables functions for working with raw computed tomography (CT) or magnetic resonance (MR) imaging data. Currently, the following functions are implemented: dataset parameter exploration by parsing of the DICOM metadata, dataset basic quality examination by comparing imaging parameters to the user-defined threshold, conversion of DICOM data into volumetric Nearly Raw Rusted Data (NRRD), image basic pre-processing, unrolling NRRD images and region of interest (ROI) masks into Joint Photographic Experts Group (JPEG) slices for a quick check of co-registration between imaging data and masks, radiomics feature extraction from NRRD/MHA data using Pyradiomics [12]. The image and mask co-alignment preview example is illustrated in Fig. 2.

The features module allows for the exploration of the feature datasets. It consists of the base FeaturesSet class and the inheriting AnalysisBox class. The FeaturesSet class reads the features data and the corresponding metadata and initializes a FeaturesSet object. The AnalysisBox class allows for the primary analysis of the features. Currently, the following functions are implemented: visualization of feature value distributions in classes and mutual Spearman correlation matrix, calculation of corrected p-values for Mann–Whitney U-test for features mean values in groups, visualization of univariate receiver operating characteristic (ROC) curves for each feature and calculation of the area under the curve (AUC), volumetric analysis, calculation of basic statistics for every feature. Features distribution in classes visualization is illustrated in Fig. 3.

The binary classification metrics reporting module allows for the generation of binary classification performance metrics given true labels and predicted probabilities.

Quality control

To ensure that precision-medicine-toolbox meets the requirements, continuous integration workflow is built in GitHub actions. Tests are run automatically after every new commit is pushed. Every time, the project is built and unit tests are performed for the latest Windows system on Python 3.7. Quick start and running software examples are described in the documentation. Additionally, code quality is reviewed with CodeFactor (http://codefactor.io). The API specifications for all the classes and methods are generated automatically from the source code annotations with Mkdocstrings (https://mkdocstrings.github.io/). This enables keeping documentation up to date with the latest developments of the package.

Software impacts

The functionality of the toolbox aims to meet some challenges that are specific to the radiomics field. One of these challenges is the lack of data and pipelines standardization. Therefore, reproducibility is one of the key criterias for the radiomics studies.

The toolbox is mostly dedicated to radiomics analysis, as it allows for the handling of both raw imaging data and derivative features. Nevertheless, its modules can be used separately for other medical imaging research applications. The imaging module is applicable for

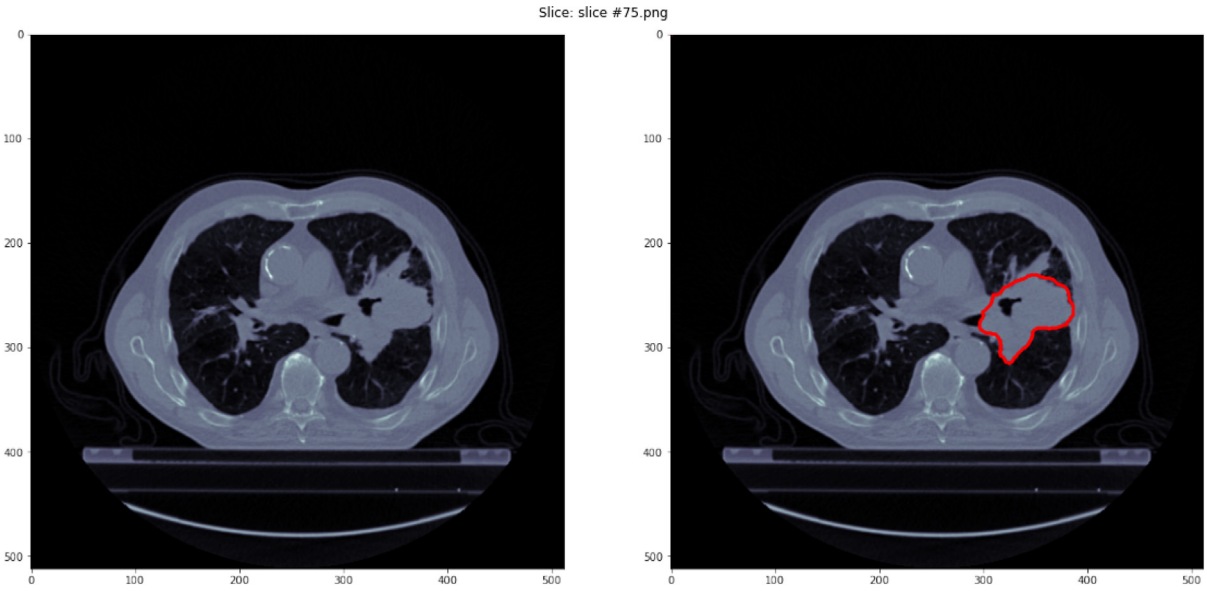

Fig. 2. Example of the quick check of the segmentation alignment to the original scan by visualizing CT axial slices.

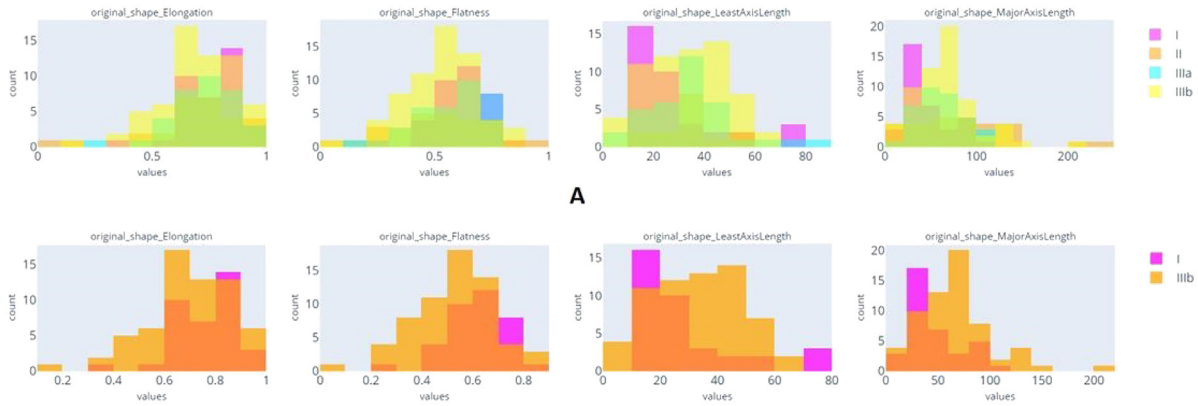

Fig. 3. Feature value distributions in multiple classes: A - for all the presented classes, B - for the selected classes I and IIIb.

deep learning tasks to prepare the imaging data and get information regarding the metadata. The features module can be used for any tabular data analysis, such as health records or histology-derived features.

The toolbox was utilized and tested during the development of multiple projects including automatic lung tumor segmentation on the CT [27], repeatability of breast MRI radiomic features [28], and radiomic-based diagnosis of multiple sclerosis [29].

The development of precision-medicine-toolbox aims for the democratization of the machine learning and deep learning pipelines for researchers without strong programming skills. Additionally, it drives a programming community effort to improve this package and add its own variables and methods. Therefore, user contributions are very welcome.

Conclusions and future works

The development of the precision-medicine-toolbox aims to lower the entry barrier for researchers who are starting to work in medical imaging. Moreover, it provides an open-source solution for the researchers who already have their inhouse workflow of managing data to increase the reproducibility of the quantitative medical imaging research. We would also like to encourage the community to improve this open-source toolbox by contributing to it.

Declaration of competing interest

Philippe Lambin reports a relationship with Radiomics that includes: consulting or advisory, funding grants, speaking and lecture fees, and travel reimbursement. Philippe Lambin reports a relationship with ptTheragnostics that includes: consulting or advisory, equity or stocks, funding grants, speaking and lecture fees, and travel reimbursement. Philippe Lambin reports a relationship with Health Innovation Ventures that includes: funding grants. Philippe Lambin reports a relationship with Convert Pharmaceuticals that includes: consulting or advisory, equity or stocks, speaking and lecture fees, and travel reimbursement. Philippe Lambin reports a relationship with Communicare Solutions that includes: equity or stocks. Philippe Lambin reports a relationship with LivingMed Biotech that includes: equity or stocks. Philippe Lambin reports a relationship with BMS that includes: consulting or advisory, speaking and lecture fees, and travel reimbursement. Philippe Lambin reports a relationship with Elekta that includes: consulting or advisory, speaking and lecture fees, and travel reimbursement. Philippe Lambin reports a relationship with Varian Medical Systems Inc that includes: consulting or advisory, speaking and lecture fees, and travel reimbursement. Philippe Lambin reports a relationship with Merck that includes: consulting or advisory, speaking

and lecture fees, and travel reimbursement. Philippe Lambin reports a relationship with BHV that includes: consulting or advisory, speaking and lecture fees, and travel reimbursement. Henry C. Woodruff reports a relationship with Radiomics that includes: equity or stocks.

Acknowledgments

The authors would like to thank the Precision Medicine department colleagues and external users for the feedback, Mart Smidt for testing the tool on the different data, PyRadiomics for a reliable open-source tool for features extraction, Hugo Aerts et al. for the Lung1 dataset we used to demonstrate our functionality, and The Cancer Imaging Archive for the publically available data.

References

[1] T. Niu, X. Sun, P. Yang, G. Cao, K.K. Tha, H. Shirato, K. Horst, L. Xing, Pathways to radiomics-aided clinical decision-making for precision medicine, in: Radiomics and Radiogenomics, Chapman and Hall/CRC, 2019, pp. 193–201.

[2] U.R. Acharya, Y. Hagiwara, V.K. Sudarshan, W.Y. Chan, K.H. Ng, Towards precision medicine: from quantitative imaging to radiomics, J. Zhejiang Univ. Sci. B 19 (1) (2018) 6–24.

[3] P. Lambin, E. Rios-Velazquez, R. Leijenaar, S. Carvalho, R.G.P.M. van Stiphout, P. Granton, C.M.L. Zegers, R. Gillies, R. Boellard, A. Dekker, H.J.W.L. Aerts, Radiomics: extracting more information from medical images using advanced feature analysis, Eur. J. Cancer 48 (4) (2012) 441–446.

[4] J. Schmidhuber, Deep learning in neural networks: an overview, Neural Netw. 61 (2015) 85–117.

[5] O. Oren, B.J. Gersh, D.L. Bhatt, Artificial intelligence in medical imaging: switching from radiographic pathological data to clinically meaningful endpoints, Lancet Digit Health 2 (9) (2020) e486–e488.

[6] R. Aggarwal, V. Sounderajah, G. Martin, D.S.W. Ting, A. Karthikesalingam, D. King, H. Ashrafian, A. Darzi, Diagnostic accuracy of deep learning in medical imaging: a systematic review and meta-analysis, NPJ Digit Med. 4 (1) (2021) 65.

[7] S.K. Zhou, H. Greenspan, C. Davatzikos, J.S. Duncan, B. Van Ginneken, A. Madabhushi, J.L. Prince, D. Rueckert, R.M. Summers, A review of deep learning in medical imaging: Imaging traits, technology trends, case studies with progress highlights, and future promises, Proc. IEEE Inst. Electr. Electron. Eng. 109 (5) (2021) 820–838.

[8] A.S. Tagliafico, M. Piana, D. Schenone, R. Lai, A.M. Massone, N. Houssami, Overview of radiomics in breast cancer diagnosis and prognostication, Breast 49 (2020) 74–80.

[9] W. Mu, L. Jiang, J. Zhang, Y. Shi, J.E. Gray, I. Tunali, C. Gao, Y. Sun, J. Tian, X. Zhao, X. Sun, R.J. Gillies, M.B. Schabath, Non-invasive decision support for NSCLC treatment using PET/CT radiomics, Nature Commun. 11 (1) (2020) 5228.

[10] Y. Zhang, A. Oikonomou, A. Wong, M.A. Haider, F. Khalvati, Radiomics-based prognosis analysis for Non-Small cell lung cancer, Sci. Rep. 7 (2017) 46349.

[11] S. Wang, F. Xiao, W. Sun, C. Yang, C. Ma, Y. Huang, D. Xu, L. Li, J. Chen, H. Li, H. Xu, Radiomics analysis based on magnetic resonance imaging for preoperative overall survival prediction in isocitrate dehydrogenase Wild-Type glioblastoma, Front. Neurosci. 15 (2021) 791776.

[12] J.J.M. van Griethuysen, A. Fedorov, C. Parmar, A. Hosny, N. Aucoin, V. Narayan, R.G.H. Beets-Tan, J.-C. Fillion-Robin, S. Pieper, H.J.W.L. Aerts, Computational radiomics system to decode the radiographic phenotype, Cancer Res. 77 (21) (2017) e104–e107.

[13] X. Fave, L. Zhang, J. Yang, D. Mackin, P. Balter, D. Gomez, D. Followill, A.K. Jones, F. Stingo, L.E. Court, Impact of image preprocessing on the volume dependence and prognostic potential of radiomics features in non-small cell lung cancer, Transl. Cancer Res. 5 (4) (2016) 349–363.

[14] R. Zhang, L. Zhu, Z. Cai, W. Jiang, J. Li, C. Yang, C. Yu, B. Jiang, W. Wang, W. Xu, X. Chai, X. Zhang, Y. Tang, Potential feature exploration and model development based on 18F-FDG PET/CT images for differentiating benign and malignant lung lesions, Eur. J. Radiol. 121 (2019) 108735.

[15] S.A. Hosseini, I. Shiri, G. Hajianfar, P. Ghafarian, M.B. Karam, M.R. Ay, The impact of preprocessing on the PET-CT radiomics features in non-small cell lung cancer, Front. Biomed. Technol. (2021).

[16] S. van der Walt, S.C. Colbert, G. Varoquaux, The NumPy array: A structure for efficient numerical computation, Comput. Sci. Eng. 13 (2) (2011) 22–30.

[17] W. McKinney, Data structures for statistical computing in Python, in: Proceedings of the 9th Python in Science Conference, SciPy, 2010.

[18] D. Mason, SU-E-T-33: Pydicom: An open source DICOM library, Med. Phys. 38 (6Part10) (2011) 3493.

[19] S. van der Walt, J.L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J.D. Warner, N. Yager, E. Gouillart, T. Yu, scikit-image contributors, Scikit-image: image processing in Python, PeerJ 2 (2014) e453.

[20] O. Kramer, Scikit-Learn, in: O. Kramer (Ed.), Machine Learning for Evolution Strategies, Springer International Publishing, Cham, 2016, pp. 45–53.

[21] Z. Yaniv, B.C. Lowekamp, H.J. Johnson, R. Beare, SimpleITK Image-Analysis notebooks: a collaborative environment for education and reproducible research, J. Digit. Imaging 31 (3) (2018) 290–303.

[22] M. Brett, M. Hanke, C. Markiewicz, M. Côté, P. McCarthy, S. Ghosh, D. Wassermann, S. Gerhard, Y. Halchenko, E. Larson, et al., Nipy/nibabel: 3.2.1, 2020.

[23] P. Virtanen, R. Gommers, T.E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, et al., SciPy 1.0: fundamental algorithms for scientific computing in Python, Nature Methods 17 (3) (2020) 261–272.

[24] K. Gorgolewski, C.D. Burns, C. Madison, D. Clark, Y.O. Halchenko, M.L. Waskom, S.S. Ghosh, Nipype: a flexible, lightweight and extensible neuroimaging data processing framework in Python, Front. Neuroinform. (2011) 13.

[25] A. Jungo, O. Scheidegger, M. Reyes, F. Balsiger, Pymia: A Python package for data handling and evaluation in deep learning-based medical image analysis, Comput. Methods Programs Biomed. 198 (2021) 105796.

[26] M.J. Cardoso, W. Li, R. Brown, N. Ma, E. Kerfoot, Y. Wang, B. Murrey, A. Myronenko, C. Zhao, D. Yang, et al., MONAI: An open-source framework for deep learning in healthcare, 2022, arXiv preprint arXiv:2211.02701.

[27] S.P. Primakov, A. Ibrahim, J.E. van Timmeren, G. Wu, S.A. Keek, M. Beuque, R.W. Granzier, E. Lavrova, M. Scrivener, S. Sanduleanu, et al., Automated detection and segmentation of non-small cell lung cancer computed tomography images, Nature Commun. 13 (1) (2022) 1–12.

[28] R.W. Granzier, A. Ibrahim, S. Primakov, S.A. Keek, I. Halilaj, A. Zwanenburg, S.M. Engelen, M.B. Lobbes, P. Lambin, H. Woodruff, et al., Test–retest data for the assessment of breast MRI radiomic feature repeatability, J. Magn. Resonance Imaging 56 (2) (2022) 592–604.

[29] E. Lavrova, E. Lommers, H.C. Woodruff, A. Chatterjee, P. Maquet, E. Salmon, P. Lambin, C. Phillips, Exploratory radiomic analysis of conventional vs. quantitative brain MRI: Toward automatic diagnosis of early multiple sclerosis, Front. Neurosci. (2021) 838.