Patch2Self2: Self-supervised Denoising on Coresets via Matrix Sketching [Supplement]

Shreyas Fadnavis*
Johnson and Johnson R&D
Cambridge, MA
sfadnavi@its.jnj.com

Agniva Chowdhury
Oak Ridge National Laboratory
Oak Ridge, TN
chowdhurya@ornl.gov

Joshua Batson
Anthropic
San Francisco, CA
joshb@anthropic.com

Petros Drineas
Purdue University
West Lafayette, IN
pdrineas@purdue.edu

Eleftherios Garyfallidis
Indiana University Bloomington
Bloomington, IN
elef@iu.edu

Acknowledgments

SF and EG were supported by the National Institute of Biomedical Imaging And Bioengineering (NIBIB) of the National Institutes of Health (NIH) under Award Numbers R01EB027585 and R01EB017230. PD and AC were partially supported by NSF grants CCF-2209509, CCF-1814041, DMS-1760353, and DOE grant DE-SC0022085. AC was also partially supported by the U.S. Department of Energy, through the Office of Advanced Scientific Computing Research's "Data-Driven Decision Control for Complex Systems (DnC2S)" project, FWP ERKJ368. Oak Ridge National Laboratory is operated by UT-Battelle LLC for the U.S. Department of Energy under contract number DEAC05-00OR22725. This manuscript has been authored in part by UT-Battelle, LLC. The US government retains, and the publisher, by accepting the article for publication, acknowledges that the US government retains, a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to the results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

1. Model Fitting Comparisons

In order to compare the model fitting gains of P2S2 against P2S, an analysis similar to P2S was performed where the goodness of fit for the diffusion tensor model [1] and the diffusion kurtosis [4] model was computed. To do this a k-fold cross-validation approach was used where signal data corresponding to the k^{th} fold were held-out and predicted using the held-in data from the rest of the folds. An R^2 metric was computed to see changes in the goodness of fit by the procedure described in [6] and implemented in DIPY [3]. This goodness of fit was computed for each voxel of 80^{th} slice in the HCP 7T [7] data after masking the data (17433 voxels were used). The improvement in the R^2 scores over noisy data was shown using strip-plots after subtracting the R^2 values of the noisy data from the R^2 values of both P2S and P2S2. As one can note from the strip-plots shown in

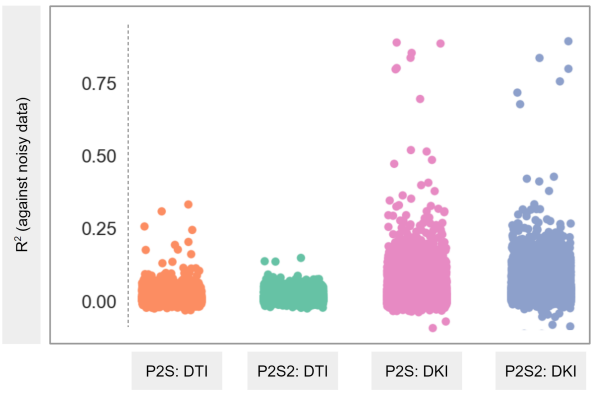

Figure S1: Depicts improvement in the R^2 scores to show a voxel-wise improvement in the goodness of fit via DTI and DKI models both via P2S and P2S2 for each model. The improvement is a subtraction of the R^2 scores obtained from fitting the model to the noisy data from the R^2 scores after fitting to P2S and P2S2 denoised data.

Fig. S1, the model fitting performance improved approximately the same after denoising with P2S2 in comparison with P2S. Some variability in the strip-plots shows is expected but the overall tendency remains the same both for

DTI and DKI model representations. This explains why the DKI metrics shown in the main text result in the same denoising performance and similar noise suppression leading to reduced fitting degeneracies (number of black voxels in the Radial Kurtosis and Mean Kurtosis maps).

2. Residual Maps Comparison

To show that the noise suppressed by P2S and P2S2 are very similar to one another, we show the residual maps for a random slice from a random diffusion weighted volume of the HCP 7T denoised data. Here volume #139 and slice #77 has been used for qualitative comparison. The same data was denoised both by P2S and P2S2 and the residuals were compared by computing root mean squared error for each individual voxel in the data shown in Fig. S2. Note

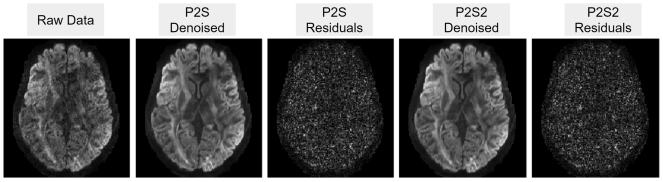

Figure S2: Shows the residual maps computed for after denoising the same subject via P2S and P2S2. Note that both show very similar noise suppression resulting in approximately the same denoised data and residual maps.

that both P2S and P2S2 show very similar performance and do not result in smoothing or suppression of signal relating to anatomy. This can be seen by visually inspecting the residual maps shown in Fig. S2 for structure. If the data was smoothed, anatomical structure would start becoming visible in the residual maps. It is also important to note that P2S2 was denoised by training the denoiser only on 50K of the 61M available training samples, i.e. only 0.083% of the data, and yielded exactly the same denoising performance.

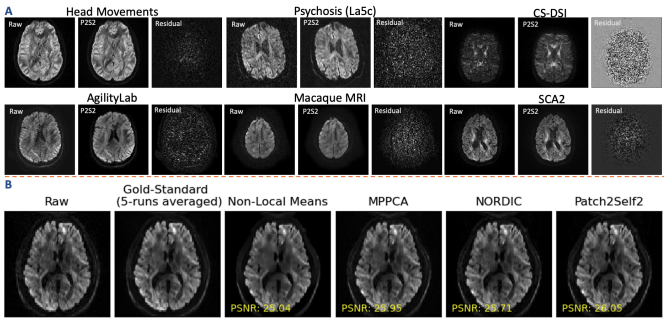

Figure S3: Benchmarking results showing Peak Signal-to-Noise Ratio (PSNR) comparisons across different datasets, illustrating the superior performance of P2S2 over SOTA algorithms NORDIC and MPPCA.

3. Comparison against other baseline methods

To benchmark against unsupervised methods, we used Peak Signal-to-Noise Ratio (PSNR) for comparison, employing the EDDEN dataset. PSNR calculations were based on an average of 5 runs, serving as a standard for clean data. Notably, P2S2 outperformed even in this benchmark, tailored for SOTA algorithms like NORDIC [5] and MPPCA [8]. Additionally, we included 6 datasets (denoised via P2S2 in $\approx$ 20 seconds) from OpenNeuro, where SOTA algorithms either fail or are inapplicable. Names of the datasets are included in the figure.

4. Code

All comparisons in the paper that have been done against P2S are based on the DIPY implementation [3] [2]. The instructions in the tutorial in the documentation [here: https://dipy.org/documentation/1.4.1./examples_built/denoise_patch2self/#example-denoise-patch2self] were followed to use P2S optimally.

P2S2 has been implemented keeping a similar API to P2S for broad adoption by the community. The anonymous code implemented for matrix sketching [File Path: /Patch2Self2/matrix_sketching] and consequently the Patch2Self2 module [File Path: /Patch2Self2/models/patch2self2.py] are available in the link here: https://figshare.com/s/078c34809ce3e39a236e.

To reproduce the results in the paper, jupyter notebooks [File Path: /Patch2Self2/notebooks] have been added to the codebase with the data available in the anonymous link below. A README file has been made available along with the code to explain how to reproduce the results, details of the jupyter notebooks and explanations of the different modules.

NOTE: All experiments in the paper and supplement were performed on a standard i7 CPU with 16 GB RAM Laptop.

5. Data

All the data that is used in the main text and the supplement has been made available in an anonymous link here: https://figshare.com/s/87f6ffee972510bfda76. The data do not have any license on usage and is openly available. The names of the data used in the experiments match exactly the examples in the jupyter notebook for reproducibility. The porcine ex-vivo data used was obtained from the link here: https://med.stanford.edu/cmrgroup/data/ex_vivo_dt_mri.html.

6. Licenses

DIPY: The Patch2Self code used for comparison and as the base code for implementing Patch2Self2 has a BSD-3 License available here: https://github.com/dipy/dipy/blob/master/LICENSE. This code implicitly also relies on Scikit-learn, Numpy and Scipy for different internal operations. Their licenses are as below:
Scikit-learn (BSD-3): https://github.com/scikit-learn/scikit-learn/blob/main/COPYING
Numpy (BSD-3): https://github.com/numpy/numpy/blob/main/LICENSE.txt
Scipy (BSD-3): https://github.com/scipy/scipy/blob/master/LICENSE.txt

The matrix sketching code used for implementing Patch2Self2 has been adopted and modified from PyRLA which has MIT License available here: https://github.com/wangshusen/PyRLA/blob/master/LICENSE.

References

[1] P.j. Basser, J. Mattiello, and D. Lebihan. MR diffusion tensor spectroscopy and imaging. *Biophysical Journal*, 66(1): 259–267, 1994. 1

[2] Shreyas Fadnavis, Joshua Batson, and Eleftherios Garyfallidis. Patch2self: Denoising diffusion MRI with self-supervised learningâ€‹. In *Advances in Neural Information Processing Systems*, pages 16293–16303. Curran Associates, Inc., 2020. 2

[3] Eleftherios Garyfallidis, Matthew Brett, Bagrat Amirbekian, Ariel Rokem, Stefan Van Der Walt, Maxime Descoteaux, and Ian and Nimmo-Smith. Dipy, a library for the analysis of diffusion MRI data. *Frontiers in Neuroinformatics*, 8, 2014. 1, 2

[4] Jens H. Jensen and Joseph A. Helpern. MRI quantification of non-gaussian water diffusion by kurtosis analysis. *NMR in Biomedicine*, 23(7):698–710, 2010. 1

[5] Steen Moeller, Pramod Pisharady Kumar, Jesper Andersson, Mehmet Akcakaya, Noam Harel, Ruoyun Ma, Xiaoping Wu, Essa Yacoub, Christophe Lenglet, and Kamil Ugurbil. Diffusion imaging in the post hcp era. *Journal of Magnetic Resonance Imaging*, 2020. 2

[6] Ariel Rokem, Jason D. Yeatman, Franco Pestilli, Kendrick N. Kay, Aviv Mezer, Stefan van der Walt, and Brian A. Wandell. Evaluating the accuracy of diffusion MRI models in white matter. *PLOS ONE*, 10(4):e0123272, 2015. 1

[7] David C Van Essen, Kamil Ugurbil, Edward Auerbach, Deanna Barch, Timothy EJ Behrens, Richard Bucholz, Acer Chang, Liyong Chen, Maurizio Corbetta, Sandra W Curtiss, et al. The human connectome project: a data acquisition perspective. *Neuroimage*, 62(4):2222–2231, 2012. 1

[8] Jelle Veraart, Dmitry S. Novikov, Daan Christiaens, Benjamin Ades-Aron, Jan Sijbers, and Els Fieremans. Denoising of diffusion MRI using random matrix theory. *NeuroImage*, 142: 394–406, 2016. 2