Magnetic Resonance Imaging 59 (2019) 130–136

Contents lists available at ScienceDirect

Magnetic Resonance Imaging

journal homepage: www.elsevier.com/locate/mri

Original contribution

A deep learning approach to estimation of subject-level bias and variance in high angular resolution diffusion imaging

Allison E. Hainline[a], Vishwesh Nath[b], Prasanna Parvathaneni[c], Kurt G. Schilling[d], Justin A. Blaber[b], Adam W. Anderson[d], Hakmook Kang[a,e,*], Bennett A. Landman[c,d,f]

[a] Biostatistics, Vanderbilt University Medical Center, Nashville, TN, USA
[b] Computer Science, Vanderbilt University, Nashville, TN, USA
[c] Electrical Engineering, Vanderbilt University, Nashville, TN, USA
[d] Vanderbilt University Institute of Imaging Science, Vanderbilt University Medical Center, Nashville, TN, USA
[e] Center for Quantitative Sciences, Vanderbilt University Medical Center, Nashville, TN, USA
[f] Department of Psychiatry and Behavioral Sciences, Vanderbilt University School of Medicine, TN, USA

ARTICLE INFO

Keywords:
HARDI
Q-ball
Bias correction
GFA
Measurement error
Neural network

ABSTRACT

The ability to evaluate empirical diffusion MRI acquisitions for quality and to correct the resulting imaging metrics allows for improved inference and increased replicability. Previous work has shown promise for estimation of bias and variance of generalized fractional anisotropy (GFA) but comes at the price of computational complexity. This paper aims to provide methods for estimating GFA, bias of GFA and standard deviation of GFA quickly and accurately. In order to provide a method for bias and variance estimation that can return results faster than the previously studied statistical techniques, three deep, fully-connected neural networks are developed for GFA, bias of GFA, and standard deviation of GFA. The results of these networks are compared to the observed values of the metrics as well as those fit from the statistical techniques (i.e. Simulation Extrapolation (SIMEX) for bias estimation and wild bootstrap for variance estimation). Our GFA network provides predictions that are closer to the true GFA values than a Q-ball fit of the observed data (root-mean-square error (RMSE) 0.0077 vs 0.0082, $p < .001$). The bias network also shows statistically significant improvement in comparison to the SIMEX-estimated error of GFA (RMSE 0.0071 vs. 0.01, $p < .001$).

1. Introduction

Diffusion-weighted magnetic resonance imaging (DW-MRI) harnesses the diffusion of water for use as a proxy for underlying tissue microstructure. Diffusion tensor imaging (DTI) characterizes this microstructure, but cannot discern fibers in more than one direction per voxel, while high angular resolution diffusion imaging (HARDI) is able to discern crossing fibers. Harmonization of DW-MRI acquisitions remains an important, yet largely misunderstood area of research. DTI-derived metrics are known to have bias as a result of imaging noise [1–6].

Here we focus on a particular HARDI metric, generalized fractional anisotropy (GFA). Previous work provided methods for both bias correction and variance estimation of GFA from a single, empirical HARDI scan [7]. Simulation Extrapolation (SIMEX) was used to estimate bias of GFA, while a wild bootstrap technique was used to estimate the standard deviation of GFA. While these methods work well, they tend to be computationally intensive due to the Monte Carlo simulations required. An alternative estimation technique would have to be developed in order to make these methods widely applicable in a clinical setting.

The application of deep neural networks (DNN) is a relatively recent development within the medical imaging literature [8,9] but is growing rapidly in popularity. The ability of DNNs to utilize large datasets in a fast and flexible way makes them very attractive to researchers looking to explore big data. Deep learning has been successfully used in a variety of neuroimaging applications including image registration [10], segmentation [11,12], inter-scanner harmonization in diffusion MRI [13], traumatic brain injury identification [14], and many more.

In this work, we demonstrate a deep neural network approach for learning GFA itself, in addition to the bias and variance of GFA from the observed data. We find that we can take observed data, put it into our network and estimate a GFA value that is closer to the truth than what would result from calculating GFA from the orientation distribution function (ODF) of the original data via a regularized Q-ball fit.

* Corresponding author at: Department of Biostatistics, Vanderbilt University Medical Center, 2525 West End, Ste. 11000, Nashville, TN 37203, USA.
E-mail address: h.kang@vumc.org (H. Kang).

https://doi.org/10.1016/j.mri.2019.03.021
Received 13 September 2018; Received in revised form 23 March 2019; Accepted 23 March 2019

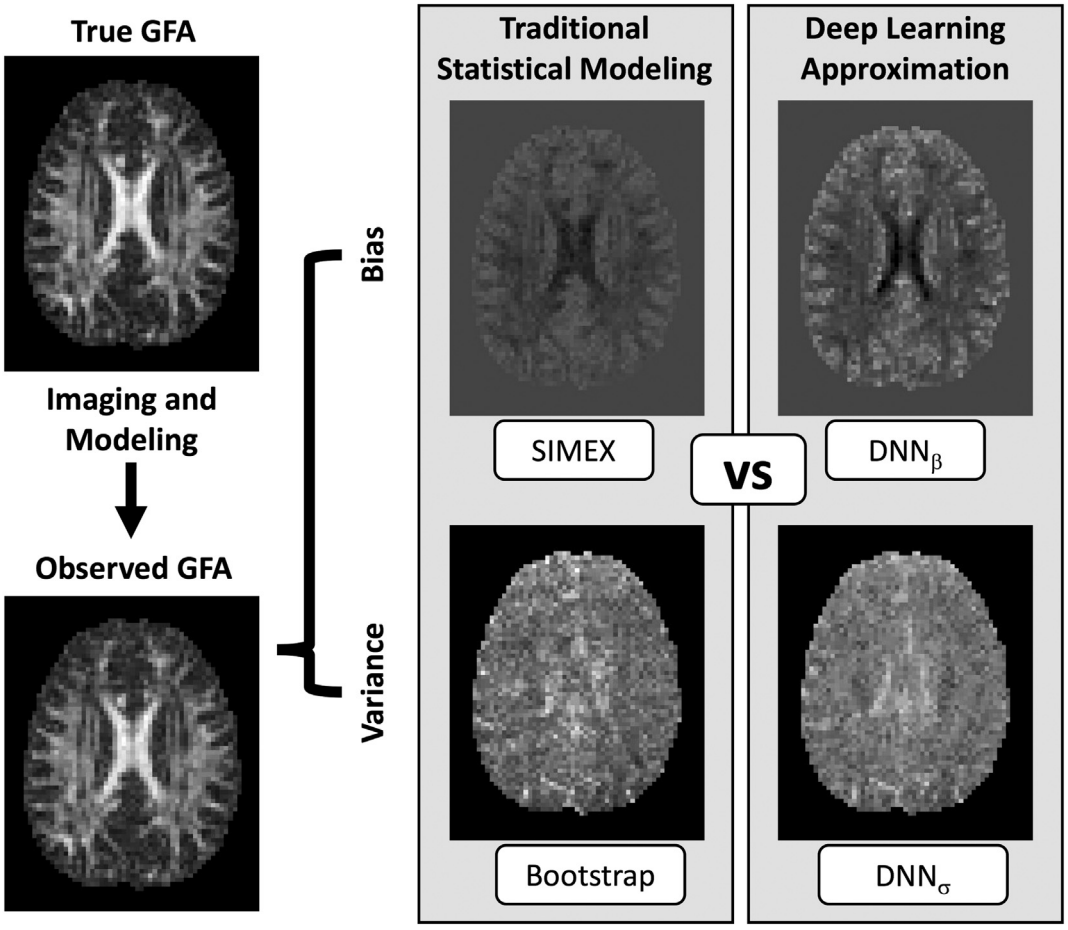

Fig. 1. Broad overview of methodology presented. The true GFA is obscured by noise and artifacts in the imaging process. We previously used traditional statistical modeling to recover the true values through SIMEX and the wild bootstrap. In this paper, we extend this idea and replace the statistical modeling techniques with deep learning approximations for bias and variance.

Fig. 1 maps out the relationship between the traditional statistical modeling techniques and the deep learning approximations proposed herein. The former are relatively computationally intensive, while the latter are much faster to apply to new datasets, though they involve extensive training initially.

2. Methods

2.1. Data acquisition and preprocessing

The empirical data used in this experiment were obtained from a healthy volunteer 3 T Phillips Scanner with a 32-channel head coil after informed consent [16,17]. The session consisted of 96 gradient directions at a b-value of 3000 s/mm^2. The voxel resolution is 2.5 mm × 2.5 mm × 2.5 mm with a matrix of 96 × 96, FOV = 240 × 240 mm^2, and 38 axial slices. The scan parameters were: Multi-Band = 2; SENSE = 2.2; TR = 2650 ms; TE = 94 ms; partial Fourier = 0.7. Fold over direction was A-P with a P fat shift. For each shell, an additional diffusion scan was acquired with reverse phase encoded volumes (i.e., fold over direction A-P with A fat shift) with a minimally weighted volume and 3 diffusion weighting directions with a b-value of 1000 s/mm^2 along the imaging frame cardinal directions, and all other parameters were kept constant. The image was transformed to MNI space, resulting in an image matrix of 78 × 93 × 75. This registered image was used for the analysis.

A truth model is considered as the 'ground truth' data in this experiment. This model is generated from the spherical harmonic coefficients of the DW signal. This truth model is assumed to be less noisy and used as the basis for comparison of the methods.

To represent data from a typical DW acquisition, random Rician noise was added in quadrature to the ground truth data. The resulting 'observed' dataset, $\boldsymbol{X}_{obs}$, represents an empirically observed HARDI acquisition, as the noise value is the standard deviation of the residuals, σ_E. The value of σ_E impacts the signal-to-noise ratio (SNR) of the observed data.

2.2. Preparation for analysis: Calculation of true GFA, bias, variance

2.2.1. Calculating true GFA

The true GFA value is calculated via a regularized Q-ball imaging fit [18] to the true data model. For this work, a spherical harmonic basis was used in the reconstruction of the fiber orientation distribution function (ODF). GFA is given by.

$$GFA\frac{std(\psi)}{rms(\psi)} = \sqrt{\frac{m\sum_{i=1}^{m}(\psi_i - \overline{\psi})^2}{(m-1)\sum_{i=1}^{m}\psi_i^2}} \quad (1)$$

where ψ is the ODF vector, and $\overline{\psi}$ is its mean [19].

2.2.2. Calculating true bias

We used a Monte Carlo approach to determine the true bias of an observed GFA value. This method involves simulating an observed voxel, calculating the GFA value for that voxel, subtracting the true GFA to determine the error for that observed voxel. This process is then repeated 100 times, after which the 100 errors are averaged, resulting in the true, voxel-wise bias of GFA.

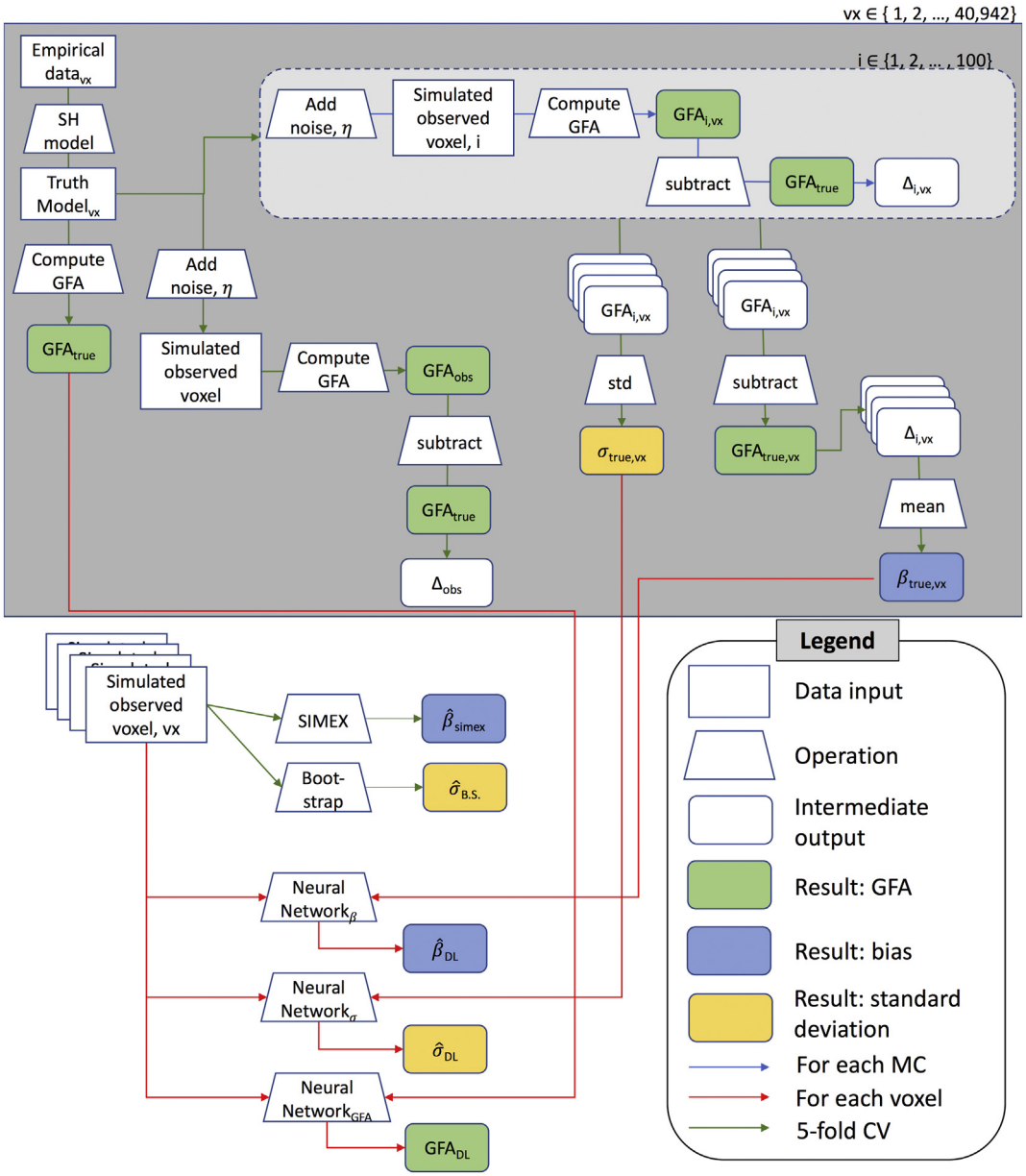

Fig. 2. Overview of the algorithm used in this analysis. Data processing steps and statistical modeling procedures are located within the gray box. The results of the statistical procedures are compared to the deep learning approximations that are shown on the bottom half of the algorithm. In this algorithm outline, the shapes refer to data types, shape colors refer to result metrics and the arrow colors refer to the methodology employed.

2.2.3. Calculating true standard deviation

A similar approach is used for determining the true standard deviation of an observed GFA value per voxel. We simply take the 100 simulated GFA values from the same Monte Carlo procedure used for calculating true bias. Taking the standard deviation of these observed GFA values gives the true, voxel-wise standard deviation of GFA.

2.2.4. SIMEX bias estimation

The SIMEX was developed by Cook and Stefanski [20] to correct for measurement error induced bias. In statistics, it is often used for analysis of electronic medical records to correct human errors, but here we apply it to help correct errors induced by the imaging machinery itself. This method applies as long as the metric of interest changes monotonically as a function of noise and the noise distribution can be estimated [20]. SIMEX utilizes the relationship between the noise level and the metric to estimate the potential, noise- (or error-) free value of the metric of interest. The method is simple and worked sufficiently well to quantify the bias and variance of GFA in an empirical HARDI acquisition. This method was first applied to DTI acquisitions [21] and was recently expanded to apply to HARDI acquisitions in [7].

A brief explanation of the SIMEX procedure follows and full details of the algorithm can be found in [7]. SIMEX is built upon the idea that our observed data are a function of the true underlying data and random noise. For our application, all calculations are done per voxel, and the observed data are the result of adding stacked Rician noise with standard deviation σ_E to our truth data [22]:

$$\boldsymbol{X}_{obs} = \boldsymbol{X}_{truth} \circ R_{\sigma_E} \tag{2}$$

In order to estimate the bias of our metric, we must discern the relationship between the metric and the noise level. Thus, we generate data values with increasing amounts of noise (indexed by ω), calculate the metric, and observe the relationship between the ω and the metric. Once enough data points have been generated to establish a pattern, we can fit a quadratic curve and find our bias-corrected value by

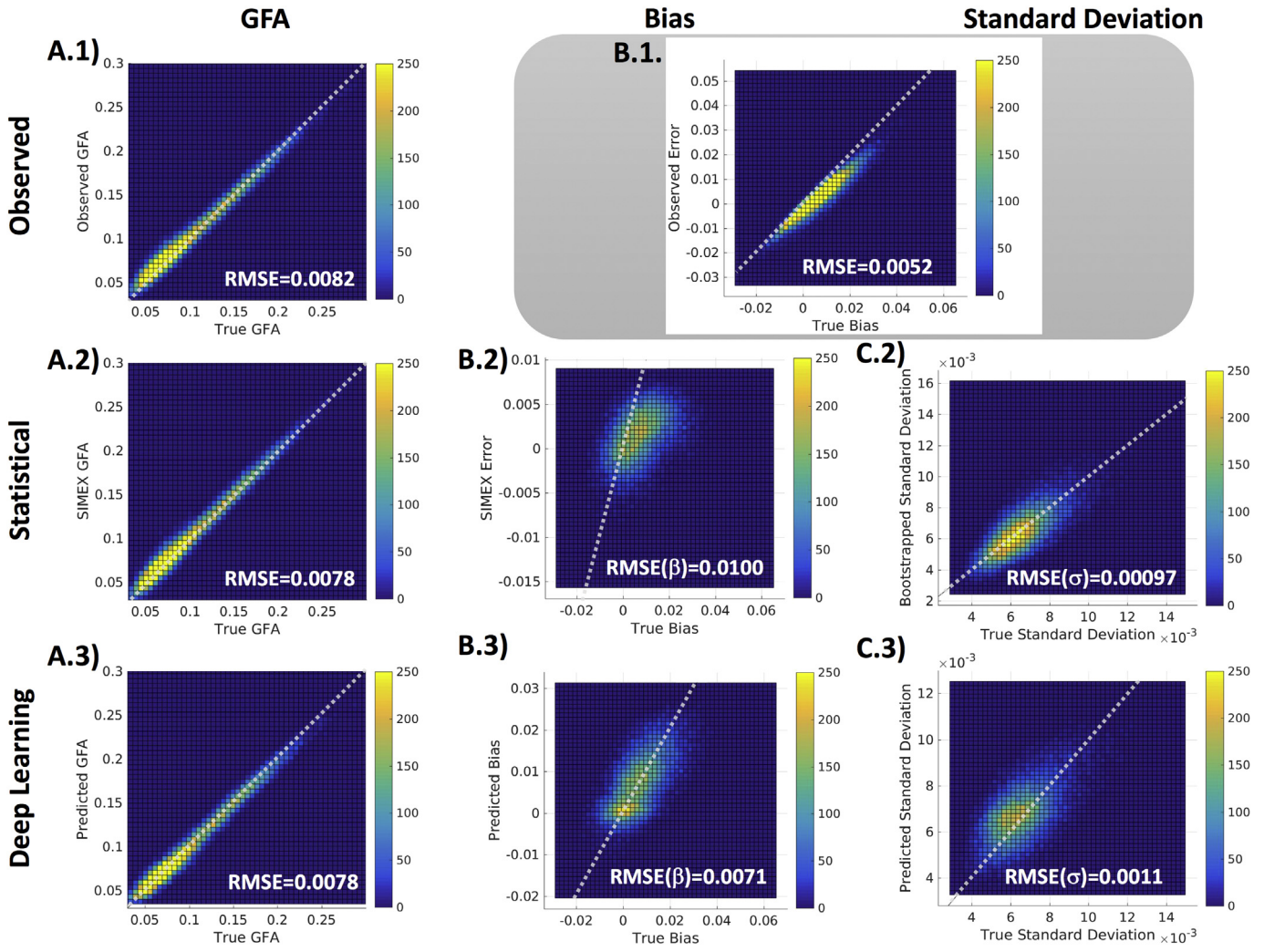

Fig. 3. Quantitative results of the deep learning approximation in comparison to observed results as well as statistical results (SIMEX and bootstrap). The deep learning approach outperforms the Q-ball calculation of GFA from the observed data (A.1 and A.2) and has similar performance to the SIMEX-corrected GFA values (A.2, and A.3). The deep learning technique for bias prediction (B.3) results in a smaller RMSE in comparison to the SIMEX error estimation technique (B.2). The deep learning approach shows larger RMSE for standard deviation prediction (C.2) in comparison to the wild bootstrap technique (C.1).

extrapolating backward. The value of the noiseless metric can be found by extrapolating this curve to the point where $\omega = -1$, i.e. the point where there is no imaging noise.

Once we calculate the SIMEX-extrapolated GFA value, we can estimate the error of GFA by subtracting it from the observed GFA value:

$$\widehat{Bias} = GFA_{obs} - GFA_{SIMEX} \quad (3)$$

2.2.5. Bootstrap estimation of variance

The previously used statistical method for estimating the variance of GFA in HARDI acquisitions was the wild bootstrap. A bootstrap method is ideal, as they do not require the use of several repeated data acquisitions to estimate variance. In particular, the wild bootstrap is chosen due to its ability to estimate variances even when the model tends to have heteroscedastic errors, as is the case with DTI data [23,24].

The wild bootstrap procedure is a modified residual bootstrap, where the first step is to compute the residuals between the model and the observed data. The signs of the residuals are then flipped randomly and added back to the observed data, resulting in a new, bootstrapped data set. This step is then repeated 100 times and the GFA is calculated for each of the 100 bootstrapped acquisitions. The standard deviation is then taken across all n simulated datasets as an estimate of the true standard deviation [25]. Please refer to [7] for full details on the wild bootstrap procedure for HARDI data acquisitions.

2.3. Data processing for deep network

The data require further processing for input to our deep learning networks. The inputs for the deep neural networks are 6th order spherical harmonic (SH) coefficients for each voxel. These coefficients were calculated as described in [18,26]. The input for a single voxel is thus a 28 × 1 vector. The outputs for the deep neural networks are the true values of GFA, bias of GFA, and standard deviation of GFA as defined in section 2.2. The outputs are thus single values for each of the 3 networks.

The HARDI truth data consists of a single brain volume with 75 axial slices. First, we separated the training data from the validation data. Training data were defined as the first 41 axial slices of the volume, and the validation data were defined as the remaining 34 slices. In terms of number of voxels, this amounts to approximately 65% of the data for training and 35% for validation. The training and validation data sets were handled separately from this point on. Note that the partition is anatomically distinct, leaving no spatial consistency between training and validation sets. Separation of the training and validation data sets in this fashion yields results that are the worst-case scenario for our models. Models trained with a true mix of voxels from all brain regions

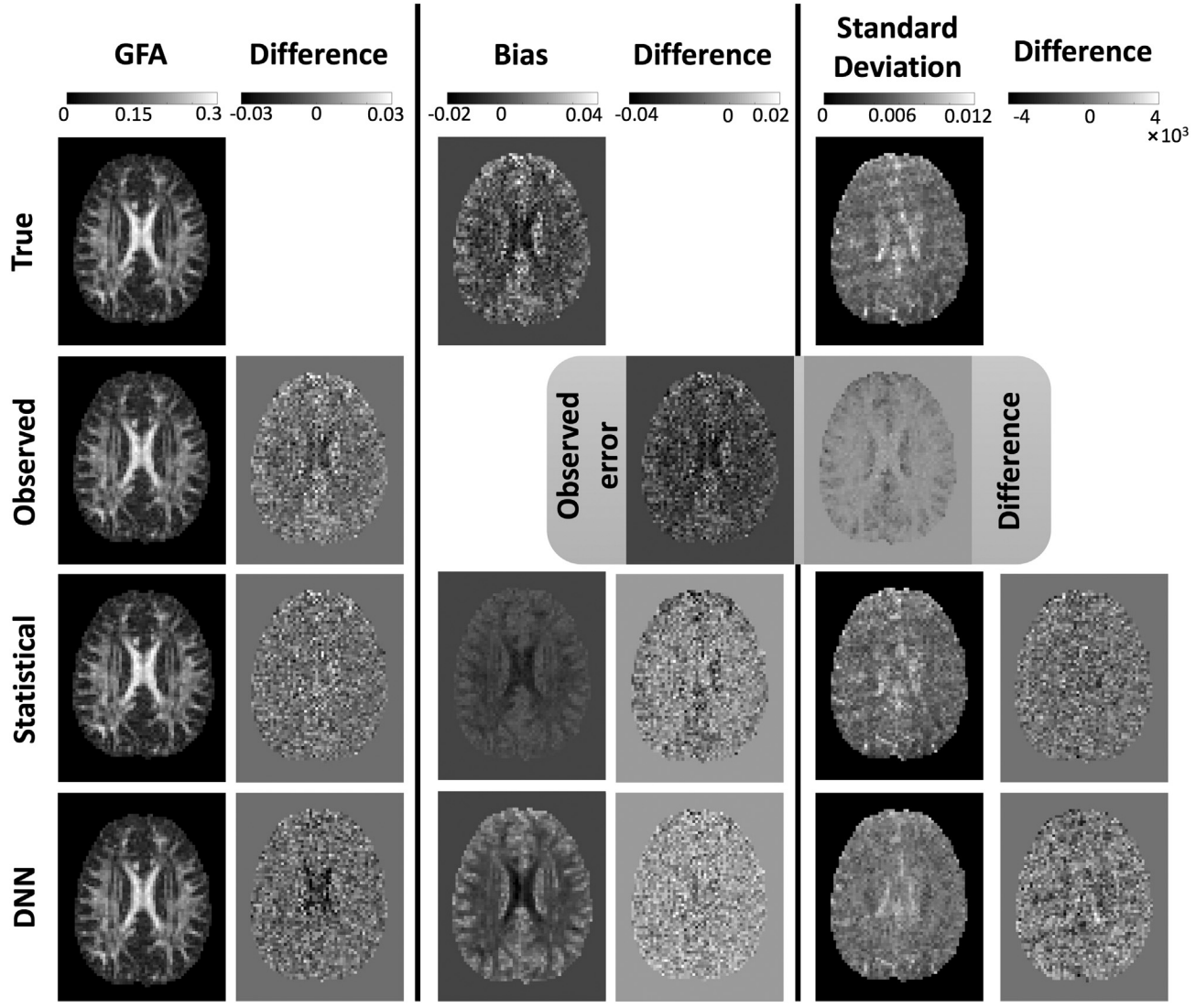

Fig. 4. Qualitative results demonstrate the performance of the GFA, bias, and standard deviation estimation methods when compared to the ground truth values of each. We see that the DNN maintains the structural qualities of the brain and maintains comparable error when compared to the statistical techniques. Note that each column of images maintains the same color scale.

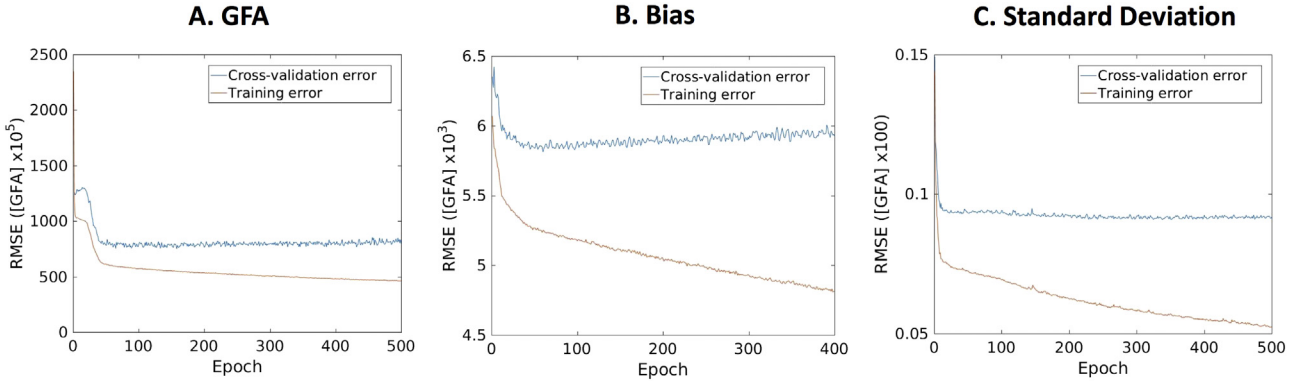

Fig. 5. Training and cross-validation error curves for each of the three deep neural networks. The GFA network (A) shows no overfitting, while the other two networks demonstrate increasing overfitting with a larger number of epochs. All three networks train within 100 epochs. The final model is taken from the epoch with the lowest cross-validation error within the training cohort.

are likely to have even better performance than those tested herein.

The next step was to remove any background voxels, as our data was not masked prior to separation of training and validation data. This was done by removing all voxels with no diffusion information. After removal of these background voxels, the data are ready for fitting.

2.4. Network design

We train three distinct fully connected deep neural networks to predict (1) GFA, (2) the bias of GFA and (3) the variance of GFA. Each of the three networks takes 6th order spherical harmonic (SH) coefficients as input. The GFA network uses the true GFA as the output, the bias network uses the true bias as the output, and the variance network uses the true standard deviation as the output. Each network uses a mean squared error loss and Adam optimizer [27]. Training data were voxels from the observed data, $\boldsymbol{X}_{obs}$. The first 41 slices of our observed data were used as training data and the remaining 34 slices were reserved as a validation set. For each network, 40,942 voxels were included for the network training and 27,798 voxels were included for network validation. For training, 5-fold cross-validation was used to assess performance with 20% validation in each set. Root mean squared error was used to evaluate network performance for each. The validation error was computed at the empirically chosen epoch where the testing error was the smallest.

2.4.1. GFA network

The output of the GFA network is the voxel-wise true GFA. This network consists of 5 fully connected layers with 1200, 400, 200, 100, 66 neurons, respectively, with a single output neuron. The first three layers use a 'ReLU' activation function. The first two layers are followed by a 30% dropout layer to help prevent overfitting. A batch size of 1000 was used.

2.4.2. Bias network

The output of the bias network is the voxel-wise true bias of GFA. This network consists of 5 fully connected layers with 400, 300, 200, 100, 66 neurons, respectively, with a single output neuron. The first two layers use a 'ReLU' activation function. The activation function is not used in the later layers to avoid constraint to positive values. The first two layers are followed by a 30% dropout layer to help prevent overfitting. A batch size of 1000 was used.

2.4.3. Variance network

The output of the variance network is the voxel-wise true standard deviation of GFA. This network consists of 5 fully connected layers with 1200, 400, 200, 100, 66 neurons, respectively, with a single output neuron. The first three layers use a 'ReLU' activation function. The first two layers are followed by a 30% dropout layer to help prevent overfitting. A batch size of 1000 was used.

2.5. Statistical analysis

Statistical significance of the proposed method is determined via a comparison of the squared errors of each method. For the GFA network, a two-way ANOVA was fit for the squared errors of the observed GFA, SIMEX-corrected GFA, and the DNN-predicted GFA. Following a significant result for the ANOVA, pairwise two-sample *t*-tests were conducted to determine pairwise significance after a Bonferroni correction. For both the bias and variance networks, two-sample t-tests were conducted between the statistical techniques and their DNN counterparts. All tests were conducted at a 5% significance level.

2.6. Characterization on an independent dataset

In order to evaluate the generalizability of the pre-trained deep neural networks, we have applied the methods described above on an additional independent dataset without retraining. This additional dataset is from the 2017 ISMRM TraCED challenge (https://my.vanderbilt.edu/ismrmtraced2017/). These data were acquired on a 3 T Philips scanner and consisted of 64 gradient directions at a b-value of 3000 s/mm^2 with a voxel resolution of 2.5 mm × 2.5 mm × 2.5 mm with a matrix of 96 × 96, FOV = 240 × 240 mm^2, and 44 slices. The scan parameters were: Multi-Band = 2; SENSE = 2.2; TE = 99 ms; partial Fourier = 0.755. Fold over direction was A-P with a P (posterior) fat shift. This additional dataset was processed the same way as the first and used as input into the three proposed deep neural networks.

3. Results

3.1. Network results

Full details of the algorithm used in this analysis are shown in Fig. 2. The results of the neural networks are compared to the results from the single observation as well as the statistical approaches (SIMEX and wild bootstrap). All calculations for the traditional statistical approaches were performed in Matlab version R2016a (MathWorks, Natick, MA) [28] and with the Camino Diffusion MRI toolkit [29]. The deep learning networks were trained using Python version 3.6.4 (Python Software Foundation) [30] and the Keras deep learning library [31]. *All code, data, and trained models are available here:* www.nitrc.org/projects/masimatlab under "Deep learning bias and variance (HARDI)."

The deep learning results are compared to the observed values as well as the SIMEX-corrected values for both GFA and bias, while the deep learning variance estimation is only compared to our SIMEX estimate.

Fig. 3 plots the true values for GFA, bias of GFA and standard deviation of GFA against the observed, estimated, and predicted values. The observed GFA plot (A.1) shows a small amount of error, though only referring to a single scan, not an average across all possible scans. We found that our deep learning approach more effectively approximated the true GFA in comparison to the observed GFA (RMSE 0.0078 vs. 0.0082, $p < .001$). The deep learning approach approximates the SIMEX estimate of GFA very well (RMSE 0.0078 vs 0.0078, $p = .987$). Finally, the SIMEX estimate showed statistically significant improvement over the observed estimate (RMSE 0.0078 vs. 0.0082, $p < .001$).

The second column of Fig. 3 refers to the performance of SIMEX and the deep learning network on the bias of GFA. The observed error of GFA is plotted against the true bias of GFA in B.1. We find that the neural network predicted bias is superior to the estimated bias of SIMEX (RMSE 0.0071 vs. 0.01, $p < .001$).

The third column of Fig. 3 demonstrates the performance of the wild bootstrap and the deep learning network on the standard deviation of GFA. The RMSE of the wild bootstrap is 12% lower than that of the deep neural network (RMSE 0.0011 vs. 0.00097, p < .001).

Fig. 4 provides a qualitative look at the comparative performance of each method for GFA, bias of GFA and standard deviation of GFA. The true GFA, bias, and standard deviation are shown in the first row, with subsequent rows demonstrating the fitted values along with difference images. We find that our deep learning networks return appropriate values for GFA, bias, and variance in compliance with tissue microstructural differences. The gray box shows the error between the observed GFA and the true GFA as well as the difference between this observed error and the true bias of GFA.

Fig. 5 shows the training and cross-validation error curves for each of the three networks. The GFA network required 147 epochs of training and showed the least amount of overfitting. The bias network required much less training, reaching a minimum error before 100 epochs and demonstrated a significant amount of overfitting in later epochs. Finally, the standard deviation network trained very quickly and saw similar overfitting to the bias network as it trained for more epochs.

3.2. Performance on an independent dataset

The following are the results of the application of our pretrained networks to the independent TraCED dataset. We found that the deep learning approach was not as effective as the SIMEX approach in estimating GFA (RMSE 0.0208 vs. 0.0091, $p < .001$). In addition, the deep learning approach failed to show significant improvement over the observed estimate of GFA (RMSE 0.0208 vs. 0.0102). The SIMEX approach, however, showed a statistically significant improvement when compared to the observed estimate (RMSE 0.0091 vs. 0.0102, $p < .001$). For the bias network, we found that the neural network predicted bias resulted in a larger RMSE, it is not statistically different from the SIMEX estimated bias (RMSE 0.0182 vs. 0.0091, $p = .26$). The deep learning network on the standard deviation of GFA resulted in a larger RMSE than the estimates obtained from the wild bootstrap (RMSE 0.0018 vs. 0.0013, $p < .001$).

4. Discussion

Our previous work using SIMEX and the wild bootstrap to characterize HARDI data acquisitions have a variety of important applications to harmonization and data quality analyses. However, these methods can be time consuming and complicated for the casual user. The ability to evaluate and correct imaging metrics can allow for better inference and more replicable results. These methods can be especially useful in cases where brain changes are of interest, as the changes in brain microstructure are often very small and can be either diminished or magnified by bias.

In this paper we have demonstrated the potential of a deep neural network to predict the true GFA value of a voxel more accurately than a regularized Q-ball fit on the observed data without a considerable increase in computation time. The SIMEX and wild bootstrap method can take up to ten hours per acquisition, where the trained neural networks provide results in 2–3 s. The neural networks explored herein are excellent candidates for inclusion in standard data processing procedures, as monitoring bias and variance of well-known metrics such as GFA is valuable for understanding data quality. While the networks demonstrated here are not perfect, these results are encouraging and reveal the potential of similar networks to enhance and support the traditional methods in diffusion imaging.

As discussed in Section 3.2, the inability of the deep learning networks to outperform the SIMEX and wild bootstrap methods demonstrates that these networks, once trained, may only be useful for scans within the exact same study parameters or for single subject data. The two datasets used herein were acquired on different scanners and with a different number of gradient directions, though both were acquired with the same b-value of 3000 s/mm^2. While more analyses will be required to make any definitive statements about the performance and generalizability of these methods, we believe that the results within a study are compelling enough to warrant further research and deep neural network model development for HARDI data.

Future work will include training networks for various noise levels and b-values, as this work was done with a single noise level at b = 3000 s/mm^2. The ability to incorporate SNR information can only improve the performance and usefulness of these networks.

Acknowledgements

This work was supported by R01EB017230 & Grant UL1 RR024975-01 & Grant 2 UL1 TR000445-06 and Advanced Computing center for Research and Education (ACCRE). The material presented are the views of the authors listed and do not necessarily reflect the views of the sponsoring entities.

References

[1] Basser PJ. Quantifying errors in fiber-tract direction and diffusion tensor field maps resulting from MR noise. Proceedings of the 5th annual meeting of ISMRM, Vancouver, Canada. 1997.
[2] Bastin ME, Armitage PA, I M. A theoretical study of the effect of experimental noise on the measurement of anisotropy in diffusion imaging. Magn Reson Imaging 1998;16:773–85.
[3] Skare S, et al. Noise considerations in the determination of diffusion tensor anisotropy. Magn Reson Imaging 2000;18:659–69.
[4] Basser PJ, Pajevic S. Statistical artifacts in diffusion tensor MRI (DTMRI) caused by background noise. Magn Reson Med 2000;44:41–50.
[5] Farrell JA, et al. Effects of signal-to-noise ratio on the accuracy and reproducibility of diffusion tensor imaging-derived fractional anisotropy, mean diffusivity, and principal eigenvector measurements at 1.5 T. J Magn Reson Imaging 2007;26(3):756–67.
[6] Hutchinson EB, et al. Analysis of the effects of noise, DWI sampling, and value of assumed parameters in diffusion MRI models. Magn Reson Med 2018;78:1767–80.
[7] Hainline AE, Nath V, Parvathaneni P, Blaber JA, Schilling KG, Anderson AW, et al. Empirical single sample quantification of Bias and variance in Q-ball. Magn Reson Med 2018;80:1666–75.
[8] Greenspan H, Van Ginneken B, Summers RM. Overview and future promise of an exciting new technique. IEEE Trans Med Imaging 2016:1153–9.
[9] Litjens G, et al. A survey on deep learning in medical image analysis. Med Image Anal 2017:60–88.
[10] de Vos BD, et al. A deep learning framework for unsupervised affine and deformable image registration. Med Image Anal 2019:128–43.
[11] de Brebisson A, Montana G. Deep neural networks for anatomical brain segmentation. 2015.
[12] Havaei M, et al. Brain tumor segmentation with deep neural networks. Med Image Anal 2017:18–31.
[13] Nath V, et al. Inter-scanner harmonization of high angular resolution DW-MRI using null space deep learning. 2018. (arXiv preprint, arXiv:1810.04260).
[14] Minaee S, et al. A deep unsupervised learning approach toward MTBI identification using diffusion MRI. 2018. (arXiv preprint, arXiv:1802.02925).
[16] Nath V, et al. Comparison of multi-fiber reproducibility of PAS-MRI and Q-ball with empirical multiple b-value HARDI. Proceedings volume 10133, Medical imaging 2017: image processing. 2017. [101330L].
[17] Nath V, et al. Empirical estimation of intravoxel structure with persistent angular structure and Q-ball models of diffusion weighted MRI. J Med Imag 2018;5:1.
[18] Descoteaux M, et al. Regularized, fast, and robust analytical Q-ball imaging. Magn Reson Med 2007;58(3):497–510.
[19] Tuch DS. Q-ball imaging. Magn Reson Med 2004;52(6):1358–72.
[20] Cook JR, Stefanski LA. Simulation-extrapolation estimation in parametric measurement error models. J Am Stat Assoc 1994;89(428):1314–28.
[21] Lauzon CB, et al. Assessment of bias for MRI diffusion tensor imaging using SIMEX. Med Image Comput Comput Assist Interv 2011;14(Pt 2):107–15.
[22] Lauzon CB, et al. Assessment of bias in experimentally measured diffusion tensor imaging parameters using SIMEX. Magn Reson Med 2013;69(3):891–902.
[23] Basser PJ, Mattiello J, LeBihan D. Estimation of the effective self-diffusion tensor from the NMR spin echo. J Magn Reson B 1994;103(3):247–54.
[24] Whitcher B, et al. Using the wild bootstrap to quantify uncertainty in diffusion tensor imaging. Hum Brain Mapp 2008;29(3):346–62.
[25] Jones DK. Tractography gone wild: probabilistic fibre tracking using the wild bootstrap with diffusion tensor MRI. IEEE Trans Med Imaging 2008;27(9):1268–74.
[26] Descoteaux M, Angelino E, Fitzgibbons S, Deriche R. Apparent diffusion coefficients from high angular resolution diffusion imaging: estimation and applications. Magn Reson Med 2006;56(2):395–410.
[27] Kingma DP, Ba J. Adam: a method for stochastic optimization [arXiv:1412.6980 [cs.LG]]. 2014.
[28] Matlab. MATLAB release. Natick, MA, United States: The MathWorks, Inc.; 2016.
[29] Cook PA, Nedjati-Gilani YBS, Seunarine KK, Hall MG, Parker GJ, Alexander DC. Camino: Open-source diffusion-MRI reconstruction and processing. 14th scientific meeting of the International Society for Magnetic Resonance in medicine. 2006. [Seattle, WA].
[30] PythonCoreTeam. Python: A dynamic, open source programming language Python software foundation. 2015. [version 3.6.4.].
[31] Chollet Fea. Keras. 2015.