Novel Reconstruction Algorithms for Magnetic Localization for Capsule Endoscopy

Arnav Kethana - kethana.arnav@gmail.com
Dougherty Valley High School, Class of 2022
USC Viterbi | ACME Lab | Prof. Sideris | Ming Hsieh Department of Electrical Engineering, SHINE 2021

Introduction

Digestive tract issues account for millions of cases yearly. However, doctors lack a reliable localization method, which forces an attempted colonoscopy, which is often unneeded, expensive, and has adverse side effects. To combat this, reconstruction algorithms for magnetic localization of a "smart pill" were investigated. The pill is intended, when swallowed, to non-invasively collect information as it passes through the digestive tract. We model and track this pill by gathering magnetic field data of it in three dimensions amongst various positions in space. Deep Neural Networks, Tree Algorithms, and Support Vector Regressions were employed to find the most optimal reconstruction algorithm which predicts the position (x,y,z) in space, showing exactly where the health problem lies without needing colonoscopy.

Methods

I. Building Dataset

In order to model our pill and gather data around its space for the magnetic field, we utilize a commercially available electromagnetic simulation software, HFSS (High-Frequency Structure Simulator) ANSYS Electronics Desktop. In HFSS, 1D case simulation was performed to get data of the magnetic field H (A/m) versus distance (mm). We then extended our simulation case to 3D with multiple coils on each axis to be able to retrieve H-field data in a given cubic region of interest as shown in Fig.1.

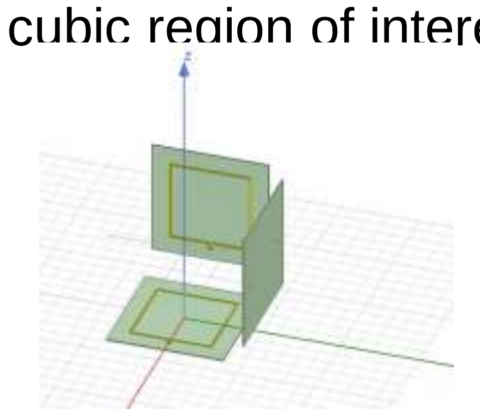


Fig. 1 - *3 coils surrounding the center(pill) in space in HFSS PC: Arnav Kethana*

Methods (Continued)

II. Machine Learning Algorithms

Various algorithms were designed and utilized to reconstruct angular position.

A. *Deep Neural Network*

A Deep Neural Network(DNN) consists of multiple layers in which matrices are curved within each layer and results are produced through each layer to come to a final result. DNNs are optimal for non-linear tasks and is very quick in regards to training. However, DNN results are complicated and hypertuning parameters is computationally very challenging.

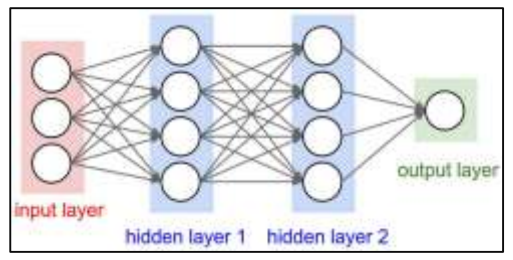


Fig. 2 - *A DNN consists of an input layers and embedded to produce a result output* PC: J. Johnson, BMC

A. *Tree-Based Algorithms*

We utilize multiple tree-based algorithms: Extra Trees and Random Forest. The Random Forest algorithm utilizes multiple trees while utilizing the average outputs of each tree to come to a result with use of an optimal split of each tree. Extra Trees utilizes a similar method but splits each tree randomly to average a result. Both are very fast, with Extra Trees 3x faster. Disadvantages are that tree-based algorithms are prone to overfitting.

$$RFfi_i = \frac{\sum_j normfi_{ij}}{\sum_{j \in all\ features, k \in all\ trees} normfi_{jk}}$$

Fig. 3 - *Random Forest Equation PC: S. Ronaghan*

A. *Support Vector Regression*

An SVR finds a hyperplane between data between two subsets of data utilizing support vectors to output results. Advantages include working well with high dimensional spaces. However, training time is the slowest and we must utilize several cross-validations which makes hypertuning challenging.

III. Analysis Metrics

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

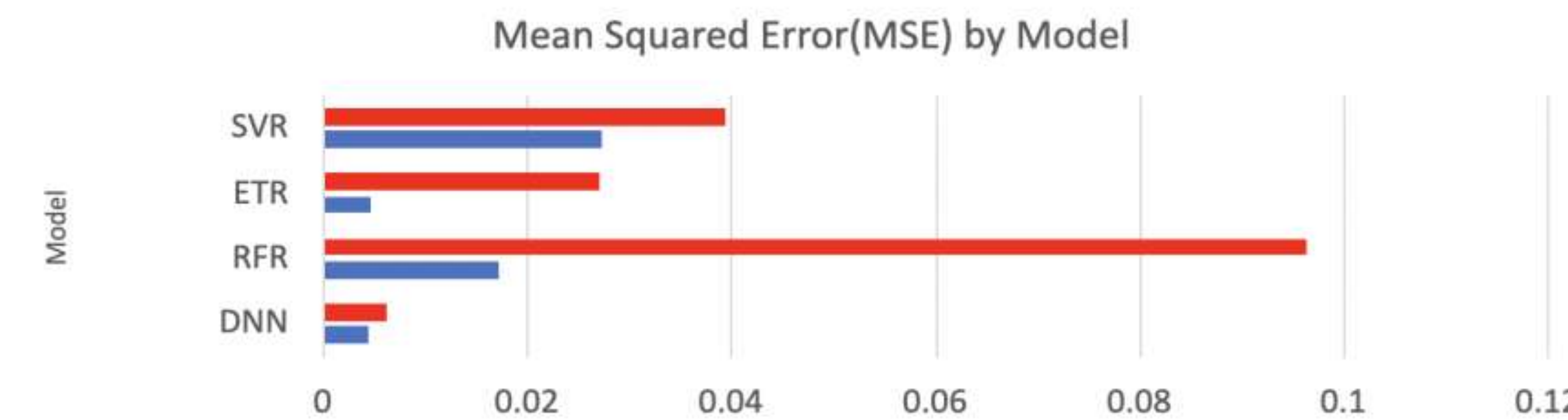
Fig. 4 - *Euclidean Distance equation used to calculated distance between data and predicted values PC: Google*

$$\mathrm{MSE} = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2$$

Fig. 5 - *Mean Squared Error used to analyze average predictive error (variation of Euclidean Dist.) PC: Google*

Results

The range in regards to MSE was around approximately less than 0.01 for DNN testing data to 0.09 for RFR testing data.



	DNN	RFR	ETR	SVR
MSE on Testing	0.0063	0.0965	0.0271	0.0394
MSE on Training	0.0045	0.0173	0.0048	0.0273

MSE on Testing ■ MSE on Training

Fig. 6- *MSE on training and testing data for all algorithms PC: Arnav Kethana*

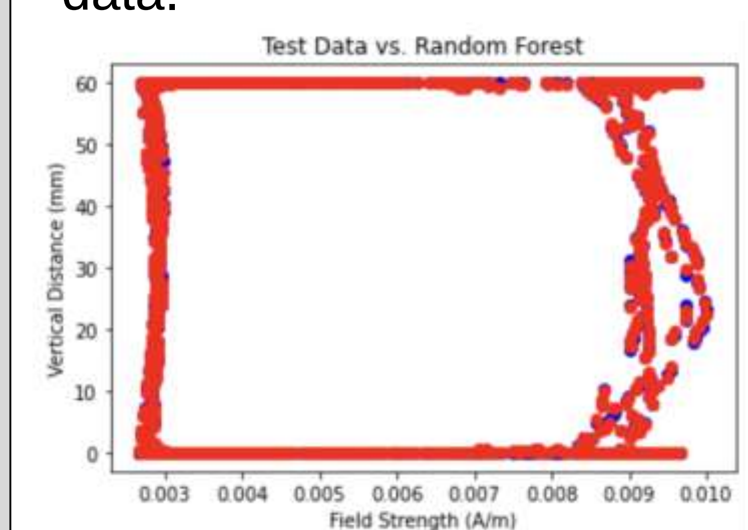


Fig. 7 - *Random Forest output(red) vs testing data(blue) PC: Arnav Kethana*

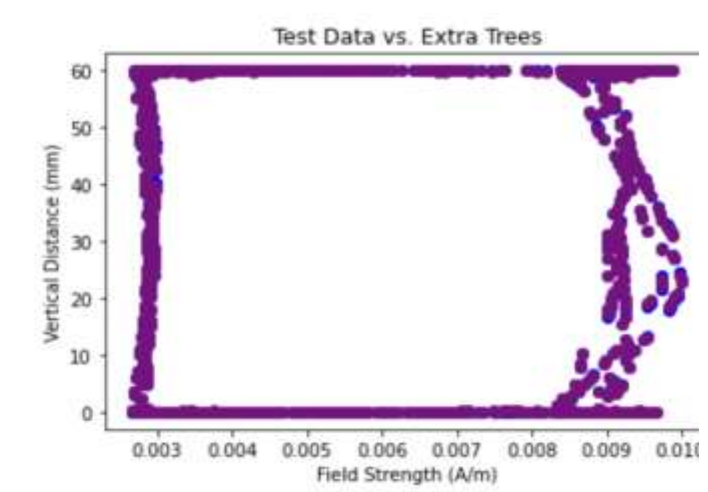


Fig. 8 - *Extra Trees output(purple) vs testing data(blue) PC: Arnav Kethana*

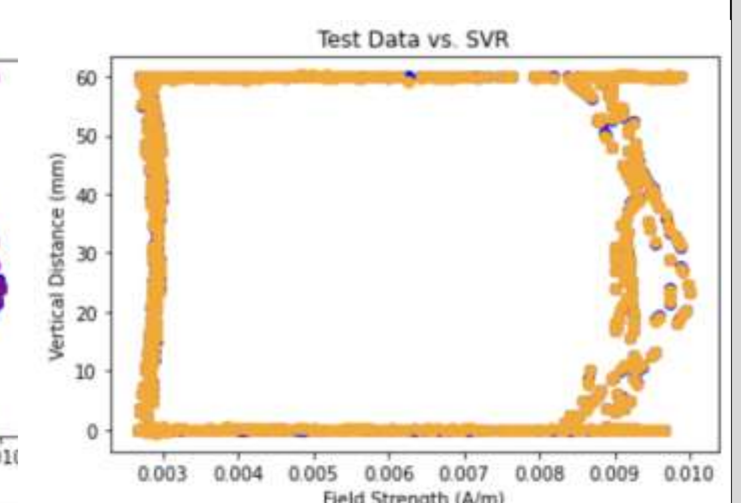


Fig. 9 - *SVR output(orange) vs testing data(blue) PC: Arnav Kethana*

Discussion/Next Steps

The DNN slightly outperformed the other models in terms of MSE, though marginally. This allows for the conclusion that a successful reconstruction algorithm has been implemented with respect to the data. Despite successful results, the dataset was relatively small, which means data augmentation is required to further validate the ability of each reconstruction algorithm. One way this can be implemented is automating a script through HFSS. This should be the next step in regards to anointing the ability of our algorithms.

Acknowledgements

I would like to thank Prof. Constantine Sideris for giving me this wonderful opportunity to contribute to his project. I would also like to thank Michella Rustom, my mentor who spent countless hours helping me alongside my research - it was amazing working with her. Finally, I would like to thank Dr. Mills and the entire SHINE team for running an amazing program.