Machine Learning for Imaging – Coursework Report
Age Regression from Brain MRI

Group Leader: **Stigliano**

Lorenzo Stigliano, Stefanos Ioannou
{ls1121, si122}@imperial.ac.uk

1 Part A

The first step in the age prediction task involved increasing the image size during preprocessing to improve image resolution. Images of size 96x96x96 with spacing of 2 in all dimensions were used to obtain better resolution images since a full image was obtained with the least amount of background as seen in Figure 1. The images were then normalized for consistency in intensity levels across different images. To carry out the brain tissue segmentation task, a U-Net architecture was chosen due to its widespread use and success in medical image segmentation [1]. The architecture had a contracting path to capture context and an expanding path for precise localization, and the number of channels increased from 16 to 256. Skip connections were used to preserve spatial information during downsampling. Various architectures were tested, and the U-Net architecture described above performed the best based on the Dice score. Figure 2 shows the training curve. The model was trained for 70 epochs to avoid overfitting. Table 1 shows the hyperparameters used for the segmentation model.

Figure 4 shows a confusion matrix of the accuracy per region of the final model. As can be seen, there is some difficulty in distinguishing between background and CSF. Classifying CSF seems to be producing a higher range of outliers compared to other regions (see Figure 3). All the regions achieve a median dice score higher than 0.85.

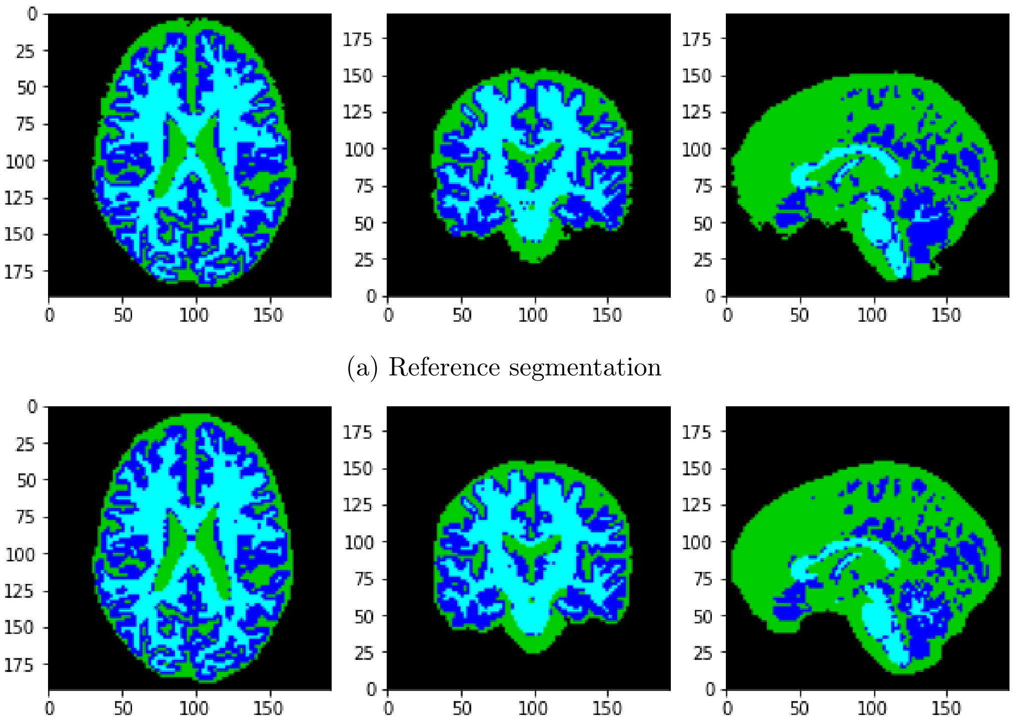
Handcrafted features were calculated from the segmentations and used as input to the regression model for age prediction. Several tests were carried out with different feature combinations and regressors to find the best-performing model. In an attempt to decrease demographic bias because of sex [2], stratified cross-validation was used, ensuring a balance between male and female subjects in the two folds. To predict age, the best model and feature combination identified were used to train a regression model on all 500 subjects (see Figure 2).

Hyperparameter	Final Value
Epochs	70
Learning Rate	0.001
Batch Size	4
Optimiser	Adam

Table 1: Hyperparameters for segmentation model (Part A).

Model	Features
Linear regression	CSF volume (normalised), GM volume (normalised), CSF^2 volume (normalised), GM^2 volume (normalised), log ratio CSF to WM volume (normalised), log ratio CSF to GM volume (normalised)

Table 2: Best model and features used for the final regression model.



(a) Reference segmentation

(b) Predicted segmentation

Figure 1: Example of final segmentation on test set using our model.

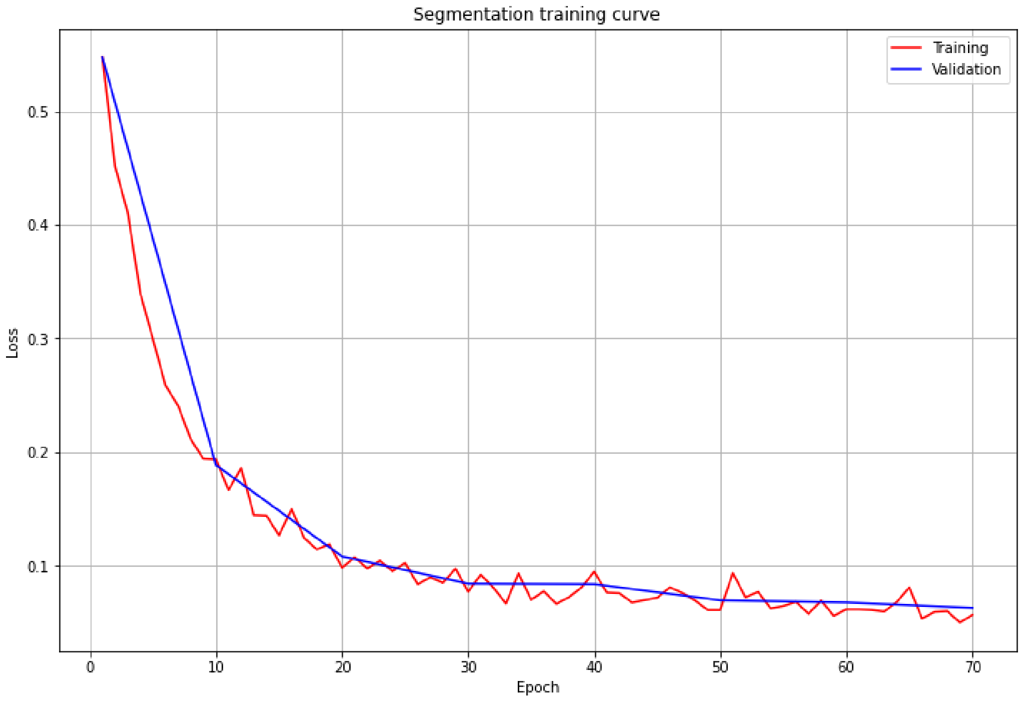


Figure 2: Training curve for segmentation for final U-Net architecture.

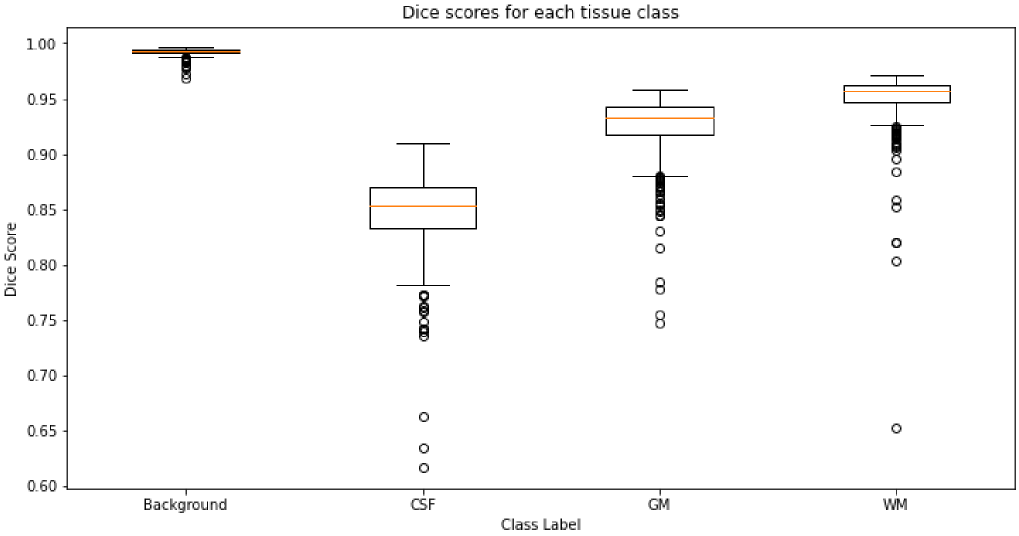


Figure 3: Segmentation test results - Box plots of dice scores for each tissue class.

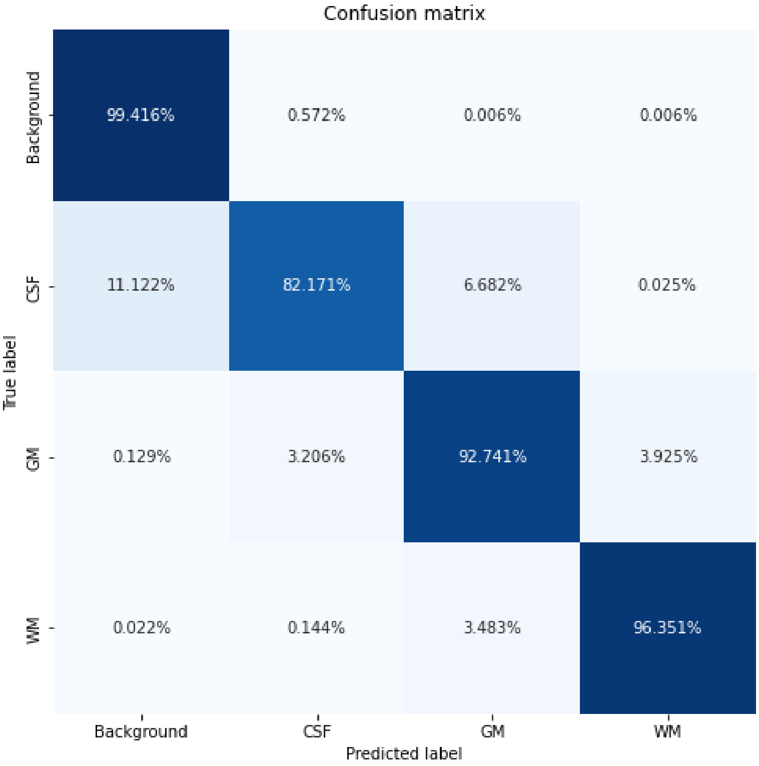


Figure 4: Segmentation test results - Confusion matrix for each tissue class.

2 Part B

The same preprocessing steps as with Part A were used. Initially, the task was tackled with architectures inspired by ResNet [3], which proved to be too large for the amount and complexity of the data given. Adding dropout layers and changing to different pooling functions (e.g. Average Pooling) showed little improvement.

For our final model, we took inspiration from AlexNet [4], in their use of Convolution, Batch Normalisation and Relu blocks. We used 5 of these. Batch Normalisation was used to stabilise learning and prevent overfitting. Contrary to the AlexNet architectures, we used smaller kernels (3x3x3) similar to VGG [5]. To further decrease the number of parameters. The Convolutional Layers and MaxPooling operations were constructed in such a way that only a single fully connected linear layer is needed.

Mean Square Error (MSE) was used as a loss function over Mean Absolute Error (MAE), since the former resulted in better performance, as it penalises outliers more. Table 3 shows the hyperparameters used to train the final model, following hyperparameter tuning using stratified 2-fold cross-validation (similar to Part A).

Figure 5 shows the learning curve of the model trained under stratified 2-fold cross-validation (similar to Part A). As can be seen, the validation loss is lower than the training loss. We think this is because of selection bias, i.e. the validation set was composed of 'easier' images than the training set. Changing to a different random seed solved this issue.

Figure Figure 6 shows the learning curves for the model trained on the full dataset ($n = 500$).

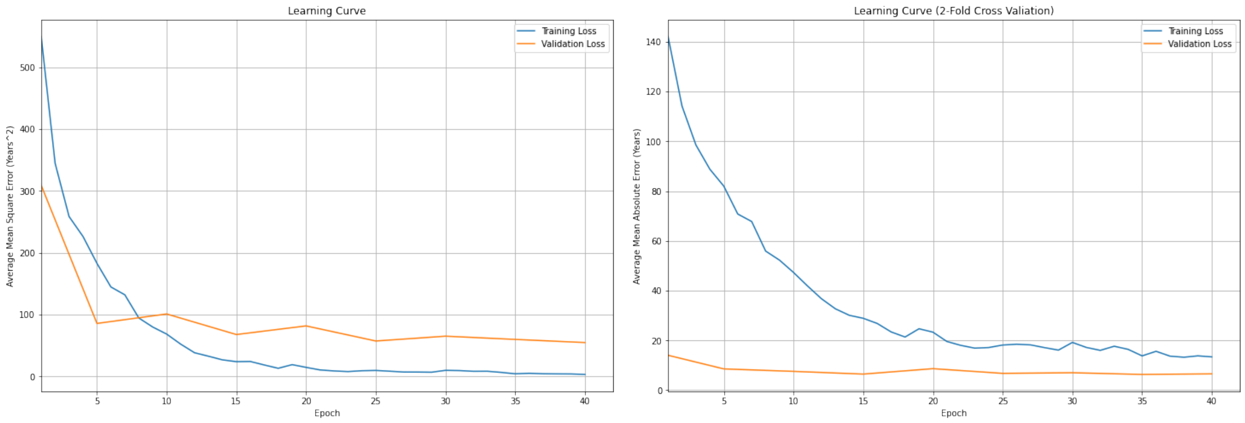


Figure 5: Learning curves for Average MSE and MAE obtained via two cross-validation on 500 subjects.

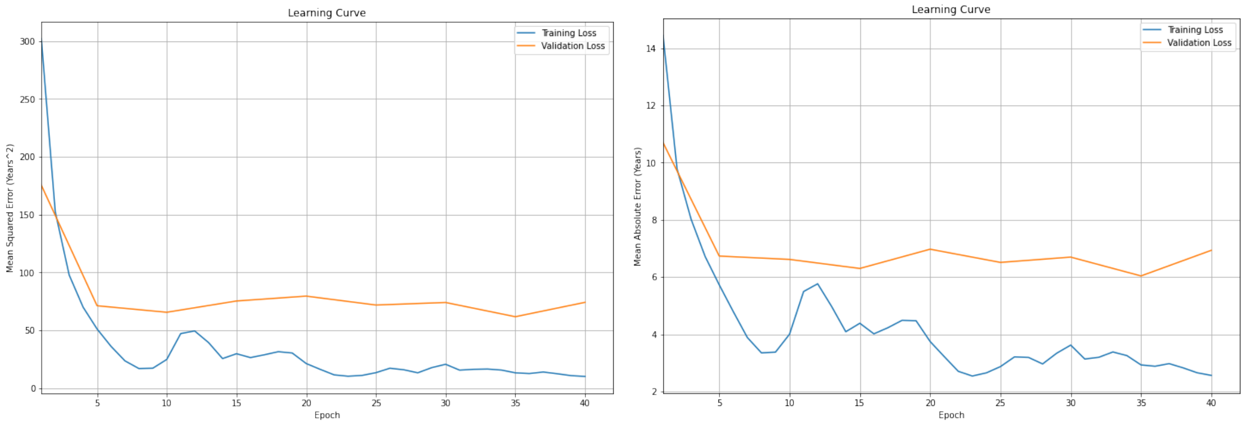
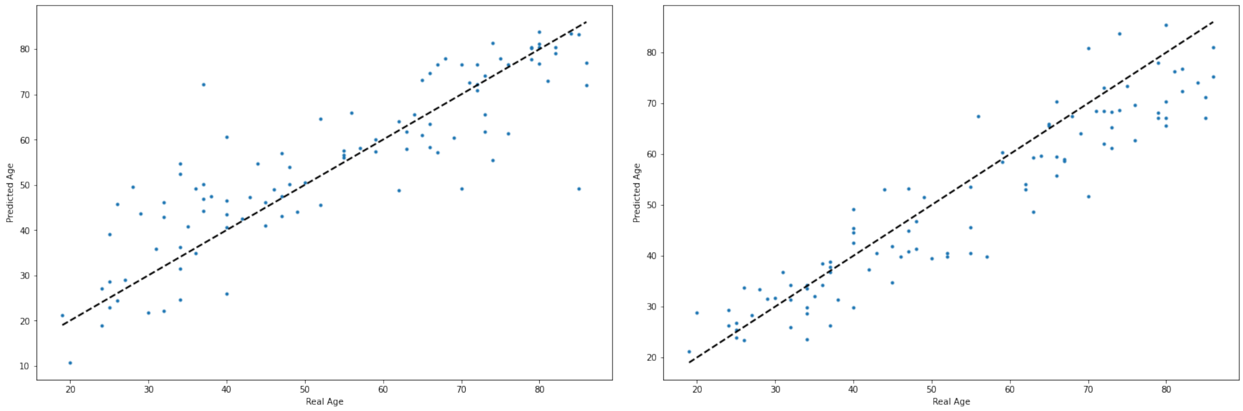


Figure 6: Learning curves for MSE and MAE obtained on all 500 subjects used to train final model.

Hyperparameter	Final Value
Epochs	40
Learning Rate	0.0001
Batch Size	2
Optimiser	RAdam

Table 3: Hyperparameters for Deep Learning model (Part B).

3 Age Regression Results



(a) Part A

(b) Part B

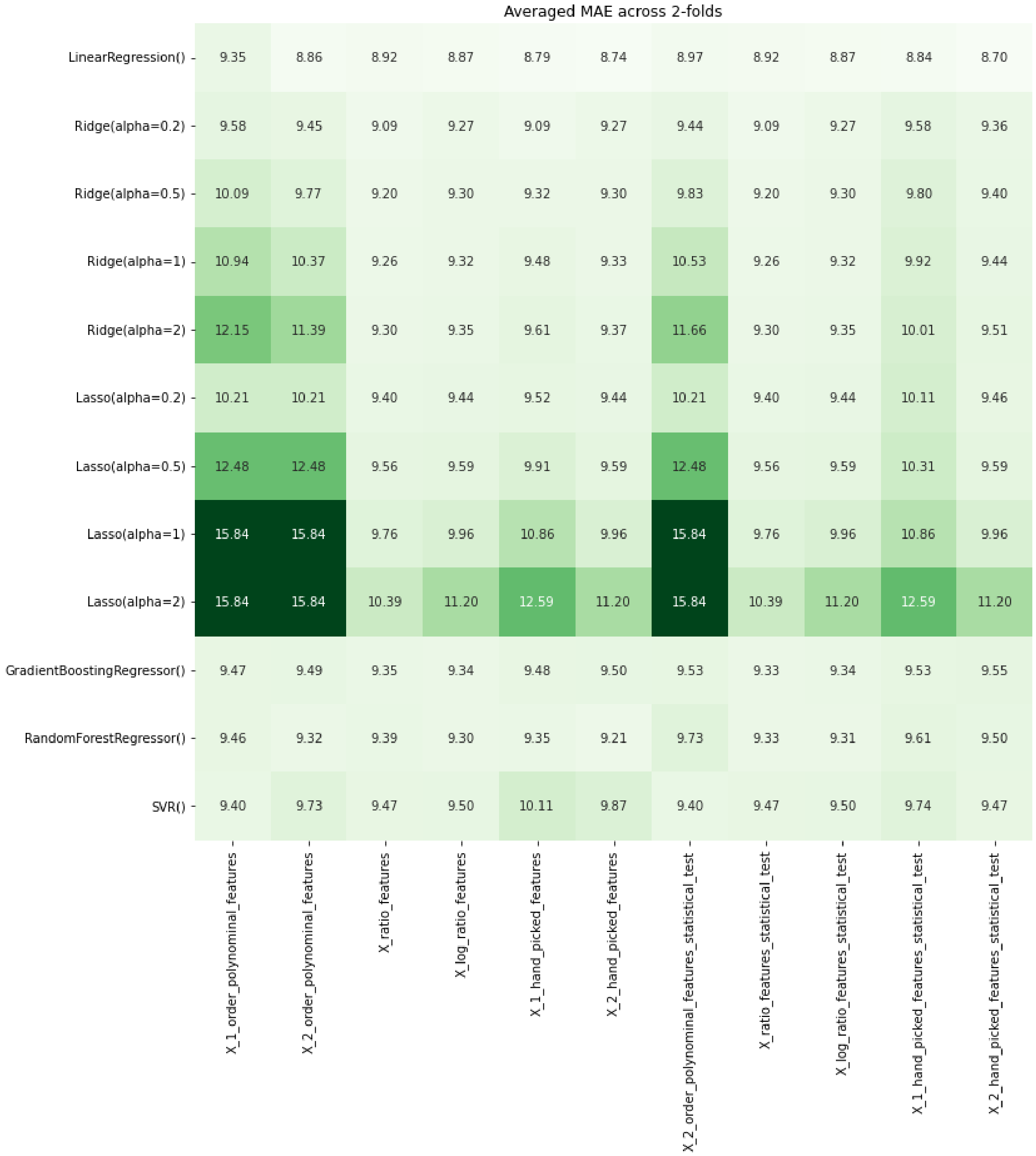
Figure 7: Scatter plots for parts A and B on final held-out dataset for predicted against real age.

Metric	Part A	Part B
Mean Absolute Error (MAE)	7.07	6.21
R^2 score	0.74	0.84

Table 4: Final regression results from methods in Part A and B on final held-out dataset.

Figure 7 shows a scatter plot of the predicted ages of the two models. Table 4 shows the MAE and R^2 score of the models. In part A, we tested different combinations of features and models to predict the outcome variable, models (see Figures 8 and 9). The features considered were; normalised volumes, squared, ratios and log ratios between all of the tissue types. In addition, we constructed subsets of features and picked the best k features using the F-test per subset. Cross-validation was then conducted, and the R^2 score and MAE were computed for each combination of features and regressors. The lowest MAE was 8.70 with R^2 score of 0.62 obtained using linear regression (see Table 2). The final model was trained with all 500 subjects, the MAE decreased to 7.07 and the R^2 score increased to 0.74. The slight decrease in MAE may be because the smaller dataset in the former experiment may be inducing bias which has a regularising effect on the model.

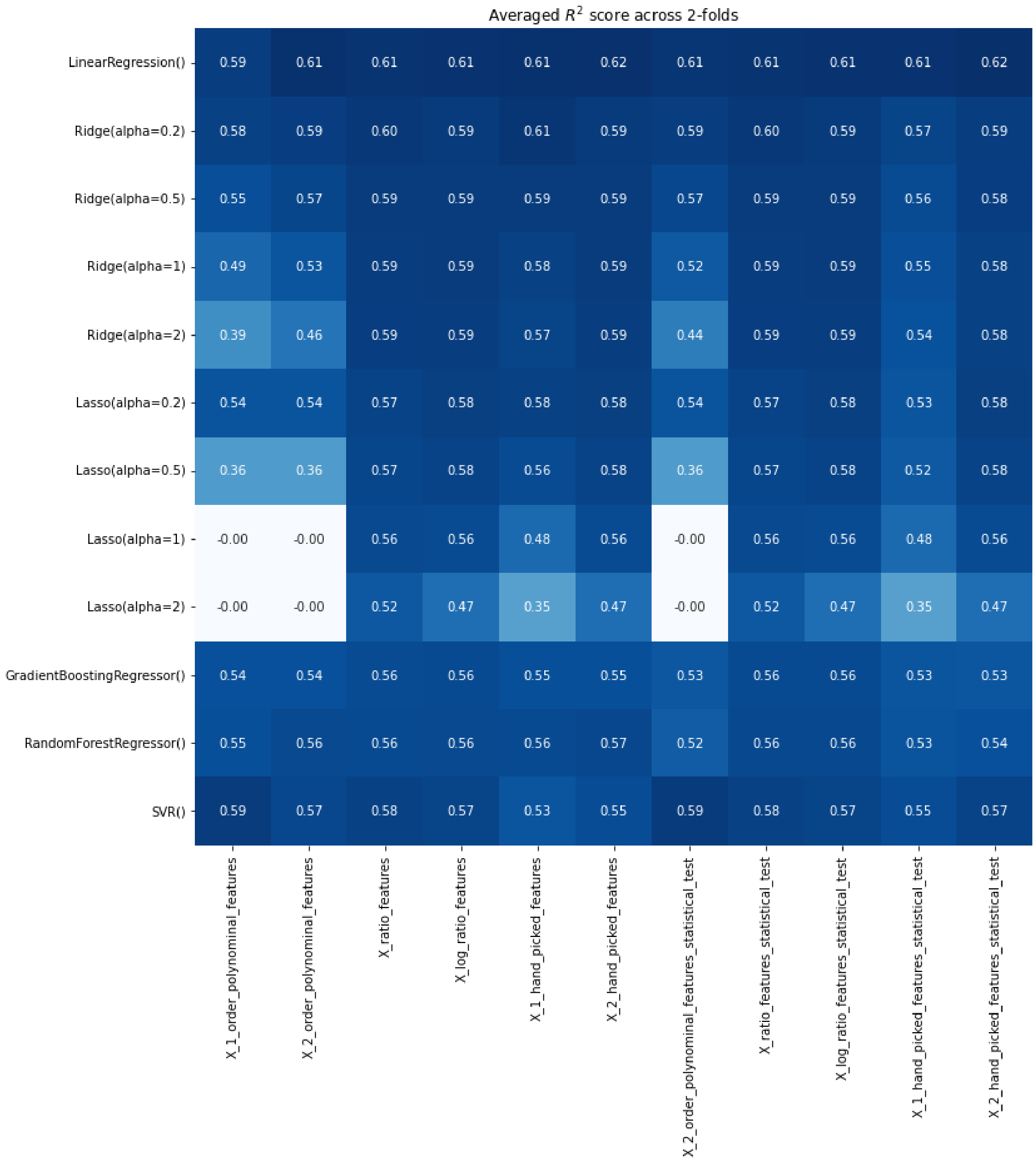
The later model (Part B) performs better compared to the model of Part A (MAE=6.21, R^2=0.84). This may be because the former automatically learns the features needed to predict age, whereas the later model pipeline (Part A), was based on user-selected features, which may lack usefulness if the user does not have domain knowledge or the features are not ideal for this task. The model in Part A is also susceptible to errors carried forward from segmentation to regression which may affect the final prediction. However, it is not clear what features the fully-deep-learning pipeline learn and whether these are 'legal'. In other words, the model may learn features like noise, which will not generalise well to other datasets. The model pipeline in Part A is more explainable since the regression models are smaller compared to the fully-deep-learning pipeline and the features are user-selected.



Averaged MAE across 2-folds

	X_1_order_polynominal_features	X_2_order_polynominal_features	X_ratio_features	X_log_ratio_features	X_1_hand_picked_features	X_2_hand_picked_features	X_2_order_polynominal_features_statistical_test	X_ratio_features_statistical_test	X_log_ratio_features_statistical_test	X_1_hand_picked_features_statistical_test	X_2_hand_picked_features_statistical_test
LinearRegression()	9.35	8.86	8.92	8.87	8.79	8.74	8.97	8.92	8.87	8.84	8.70
Ridge(alpha=0.2)	9.58	9.45	9.09	9.27	9.09	9.27	9.44	9.09	9.27	9.58	9.36
Ridge(alpha=0.5)	10.09	9.77	9.20	9.30	9.32	9.30	9.83	9.20	9.30	9.80	9.40
Ridge(alpha=1)	10.94	10.37	9.26	9.32	9.48	9.33	10.53	9.26	9.32	9.92	9.44
Ridge(alpha=2)	12.15	11.39	9.30	9.35	9.61	9.37	11.66	9.30	9.35	10.01	9.51
Lasso(alpha=0.2)	10.21	10.21	9.40	9.44	9.52	9.44	10.21	9.40	9.44	10.11	9.46
Lasso(alpha=0.5)	12.48	12.48	9.56	9.59	9.91	9.59	12.48	9.56	9.59	10.31	9.59
Lasso(alpha=1)	15.84	15.84	9.76	9.96	10.86	9.96	15.84	9.76	9.96	10.86	9.96
Lasso(alpha=2)	15.84	15.84	10.39	11.20	12.59	11.20	15.84	10.39	11.20	12.59	11.20
GradientBoostingRegressor()	9.47	9.49	9.35	9.34	9.48	9.50	9.53	9.33	9.34	9.53	9.55
RandomForestRegressor()	9.46	9.32	9.39	9.30	9.35	9.21	9.73	9.33	9.31	9.61	9.50
SVR()	9.40	9.73	9.47	9.50	10.11	9.87	9.40	9.47	9.50	9.74	9.47

Figure 8: Part A - MAE obtained via cross-validation on 500 subjects.



Averaged R^2 score across 2-folds

	X_1_order_polynomial_features	X_2_order_polynomial_features	X_ratio_features	X_log_ratio_features	X_1_hand_picked_features	X_2_hand_picked_features	X_2_order_polynomial_features_statistical_test	X_ratio_features_statistical_test	X_log_ratio_features_statistical_test	X_1_hand_picked_features_statistical_test	X_2_hand_picked_features_statistical_test
LinearRegression()	0.59	0.61	0.61	0.61	0.61	0.62	0.61	0.61	0.61	0.61	0.62
Ridge(alpha=0.2)	0.58	0.59	0.60	0.59	0.61	0.59	0.59	0.60	0.59	0.57	0.59
Ridge(alpha=0.5)	0.55	0.57	0.59	0.59	0.59	0.59	0.57	0.59	0.59	0.56	0.58
Ridge(alpha=1)	0.49	0.53	0.59	0.59	0.58	0.59	0.52	0.59	0.59	0.55	0.58
Ridge(alpha=2)	0.39	0.46	0.59	0.59	0.57	0.59	0.44	0.59	0.59	0.54	0.58
Lasso(alpha=0.2)	0.54	0.54	0.57	0.58	0.58	0.58	0.54	0.57	0.58	0.53	0.58
Lasso(alpha=0.5)	0.36	0.36	0.57	0.58	0.56	0.58	0.36	0.57	0.58	0.52	0.58
Lasso(alpha=1)	-0.00	-0.00	0.56	0.56	0.48	0.56	-0.00	0.56	0.56	0.48	0.56
Lasso(alpha=2)	-0.00	-0.00	0.52	0.47	0.35	0.47	-0.00	0.52	0.47	0.35	0.47
GradientBoostingRegressor()	0.54	0.54	0.56	0.56	0.55	0.55	0.53	0.56	0.56	0.53	0.53
RandomForestRegressor()	0.55	0.56	0.56	0.56	0.56	0.57	0.52	0.56	0.56	0.53	0.54
SVR()	0.59	0.57	0.58	0.57	0.53	0.55	0.59	0.58	0.57	0.55	0.57

Figure 9: Part A - R^2 scores obtained via cross-validation on 500 subjects.

References

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234–241, Springer, 2015.

[2] S. Ioannou, H. Chockler, A. Hammers, A. P. King, and A. D. N. Initiative, "A study of demographic bias in cnn-based brain mr segmentation," in *Machine Learning in Clinical Neuroimaging: 5th International Workshop, MLCN 2022, Held in Conjunction with MICCAI 2022, Singapore, September 18, 2022, Proceedings*, pp. 13–22, Springer, 2022.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learningfor image recognition," *CoRR, abs/1512*, vol. 3385, p. 2, 2015.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.