Application of Mutiple Maching Learning Method in Brain Age Detection

PEIXIAN MA, The Hongkong University of Science and Technology(Guangzhou), Data Science and Analytic Thrust, China

Accurately predicting brain age is a crucial step towards assessing abnormal aging patterns in individuals, as many neurological diseases are associated with deviations from normal brain aging patterns. In this study, we leveraged diverse machine learning methodologies grounded in feature engineering to precisely forecast brain age. Our comparative analysis of these models revealed a salient positive influence of feature engineering on overall model performance. However, individual models demonstrated superior performance on the validation set, indicating the occurrence of overfitting. Contrary to anticipated outcomes, the efficacy of ensemble models fell short of expectations, while residual models exhibited superior performance in a multi-stage configuration. These findings highlight the importance of appropriate feature engineering and model selection in accurately predicting brain age, which could potentially aid in the diagnosis and treatment of neurological diseases associated with abnormal aging patterns.

Additional Key Words and Phrases: Brain Age Detection, Feature Engineering, Machine Learning

ACM Reference Format:
Peixian Ma. 2024. Application of Mutiple Maching Learning Method in Brain Age Detection. 1, 1 (August 2024), 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In recent years, China has entered an aging population society, which will be an important challenge for our country in this century. Various diseases caused by abnormal aging, such as Alzheimer's disease[Wen et al. 2020] and Parkinson's disease[Bloem et al. 2021], have brought about increasing economic and social problems. Brain Age based on structural magnetic resonance is widely used to describe the aging process of the brain. The Predicted Age Difference (PAD)[Cole and Franke 2017] between brain age and actual physiological age, that is, the degree of deviation from the normal aging trajectory of the brain, can be used as an objective indicator to measure the abnormal aging of individuals. Studies have shown that many types of neurological diseases and metabolic diseases are associated with abnormal aging of the brain[Franke et al. 2012]. The greater the PAD value in the elderly, the higher the risk of neuropsychiatric problems.

The present paradigm of brain diagnosis heavily relies on the subjective knowledge and experience of front-line clinicians. This traditional approach poses several challenges such as diagnostic errors, medical delays, and other related issues that can impede the overall health of the patients. Hence, there is a critical need to explore new scientific avenues that can potentially transform the current diagnostic landscape and improve the accuracy and efficacy of brain diagnosis. Brain age prediction provides a new method to explore the abnormal changes of the brain during aging and how neuropsychiatric diseases affect normal aging, and provides a new perspective to study the individual differences of brain aging[Cole and Franke 2017].

In this research project, we have endeavored to explore the application of various machine learning models in the context of brain age detection. Our work encompasses a comprehensive pipeline that can be summarized as follows:

- We conduct a wide range of data depict methods to help us build a visual structure of this data set.
- We utilized correlation analysis to familiarize us with the correlation among features.
- Based on the understanding we obtained from the afore steps, we apply several feature expansion method in order to enhance the information supplied by certain important features.
- We compared a various of models, which could be divided into single model, ensemble learning and residual learning. This research is also one of the questions in the 2023 iflytek Developer Competition. We hope to make an exploration and contribution to the future brain age prediction technology through this work.

2 RELATED WORKS

RVM. Relevance Vector Machine (RVM)[Tipping 1999], a sparse probabilistic model similar to Support Vector Machine (SVM)[Cortes and Vapnik 1995], was proposed by Micnacl E. Tepping in 2000, and it represents a novel supervised learning method. RVM is trained in a Bayesian framework, employing automatic relevance determination (ARD)[Wipf and Nagarajan 2007] to remove irrelevant points under the structure of the prior parameters. This approach allows RVM to exhibit sparsity, efficiency, and robustness even in cases of small sample sizes. The ARD technique is a powerful tool for selecting relevant features and improving the predictive accuracy of a model, making RVM a promising candidate for various applications in machine learning and data mining.

XGBoost. Extreme Gradient Boosting (XGBoost)[Chen and Guestrin 2016] is a robust and widely used decision tree-based ensemble learning model in the field of machine learning. The fundamental concept of XGBoost is to implement the gradient boosting algorithm[Schapire and Freund 2013] in each iteration to rectify the errors of the previous iteration. The model fits the residuals by adding a new decision tree at each iteration, thus enhancing the model's performance. Due to its excellent performance in various applications, XGBoost is considered as one of the most powerful machine learning algorithms in practice.

LightGBM. Light Gradient Boosting Machine (LightGBM)[Ke et al. 2017] is a state-of-the-art gradient boosting model which offers several advantages, such as high-performance, fast, efficient, and

Author's Contact Information: Peixian Ma, The Hongkong University of Science and Technology(Guangzhou), Data Science and Analytic Thrust, Guangzhou, China, pma929@connect.hkust-gz.edu.cn.

ACM XXXX-XXXX/2024/8-ART
https://doi.org/10.1145/nnnnnnn.nnnnnnn

low memory footprint. Compared to traditional Gradient Boosting Decision Tree (GBDT) algorithm, LightGBM employs the histogram-based decision tree algorithm, which discretizes continuous features and transforms the problem into a classification problem. The histogram method is then used to partition the data, which significantly reduces the computational complexity and memory occupation. This approach has demonstrated promising results in handling large-scale data.

3 DATA PREPROCESSING

The feature data assumes a critical role in determining the final results in the context of this study. As demonstrated in Figure 1, the original dataset is first subjected to appropriate pre-processing techniques, followed by feature engineering. The ensuing section expounds upon the details of the dataset and the comprehensive data pre-processing process.

3.1 Demographic

The present training dataset comprises a total of 2000 samples, each of which consists of brain region structure-related indicators such as sex, age, volume, surface area, thickness, mean curvature, Gaussian curvature, and MRI scanner type. These indicators were obtained by medical personnel through whole brain segmentation of the original T1-weighted structural magnetic resonance images[Liang et al. 2015]. The label for each sample corresponds to the patient's actual brain age, as determined by a team of expert physicians. Notably, the dataset also features data pertaining to two distinct regions of the left and right brain, which will be subjected to further processing in the feature engineering phase.

We divide the above data sets with a ratio of 0.8:0.2, and apply five-fold cross validation in the experiments to reduce the influence of the original data distribution on the validation results.

3.2 Data Visualization and Value Check

As explicated in Figure 2 and 3, we employed the missingno package to visualize the presence of missing values in the dataset. The outcome revealed that the provided dataset is complete, with no missing values. To further explore the distribution of data values across individual files, we performed heatmap analysis, which unveiled significant variation. To mitigate this issue, we opted for Min-Max normalization of the data to ensure uniformity and comparability in our subsequent analyses.

3.3 Feature Engineering

In the realm of feature engineering, two primary techniques are employed, namely feature selection and feature augmentation. Given that the features provided by the dataset are evidently inadequate, it is imperative to explore feature augmentation to uncover additional information concealed within the data. To this end, our study employs various feature augmentation methods, which can be broadly categorized into linear and nonlinear expansions.

In the context of linear expansion, it is common practice to apply a manifold linear combination of original features. However, this simple combination of features may not be sufficient to manifest the sophisticated correlation between features that is required for a model to more clearly study them. To address this limitation, researchers often conduct some basic nonlinear feature expansion, such as using power functions. However, such methods may not always lead to clear improvements in model performance. In light of this, more complex feature expansion methods, such as the Truncated Power Basis Function[James et al. 2013], have been proposed. This expansion involves formulating the feature expansion in a more intricate manner, which can better capture the complex relationships between features. This expansion could be formulated as:

$$N_1(X_j) = 1; N_2(X_j) = X_j \tag{1}$$

$$N_{h+2}(X_j) = d_h(X_j) - d_{k_j-1}(X_j); h = 1, 2, ..., (k_j - 2) \tag{2}$$

where

$$d_h(X_j) = \frac{(X_j - \tau_h)_+^3 - (X_j - \tau_{k_j})^3}{\tau_h - \tau_{k_j}} \tag{3}$$

and we denoted our knots as $\tau_i, i = 1, 2, ..., k_j$

This expansion could project the origin features into a space with dimension decided by k. Through this more complicated nonlinear projection the expanded features could supply the model with some nonlinear in advanced, instead of discovering it by the model itself.

Finally, we fuse the normalized features of the left and right brain to obtain hybrid features:

$$MixedFeature = 0.5 \times leftbrain + 0.5 \times rightbrain \tag{4}$$

and, we concatenate the original features and mixed features to form the feature for the model input:

$$FinalMixedFeature = [AllFeature] \tag{5}$$

4 MODELING

Based on the feature matrices that have been preprocessed, we utilize a variety of machine learning techniques to develop regression and prediction models. To further improve the accuracy of our models, we construct ensemble models and residual models as depicted in FIG 4. In order to evaluate the effectiveness of these models, we employ the mean absolute error (MAE) as the loss function and compare the outcomes. The model with the best performance is then selected for final submission.

4.1 Ensemble Learning

Based on the preprocessed feature matrices, a range of machine learning techniques have been employed to develop regression and prediction models. In order to enhance the accuracy of these models, ensemble models and residual models have been constructed, as illustrated in Figure 6. To evaluate the effectiveness of these models, the mean absolute error (MAE) is utilized as the loss function and the outcomes are compared. The model with the best performance is then selected for final submission. To improve the generalization ability of the model and to mitigate the risk of overfitting, ensemble model techniques are utilized. In addition to the basic Bagging[Lee et al. 2020], Adaboost[Freund et al. 1996] and GBDTGBDT[Friedman 2001] techniques, the predictions generated by the above-mentioned

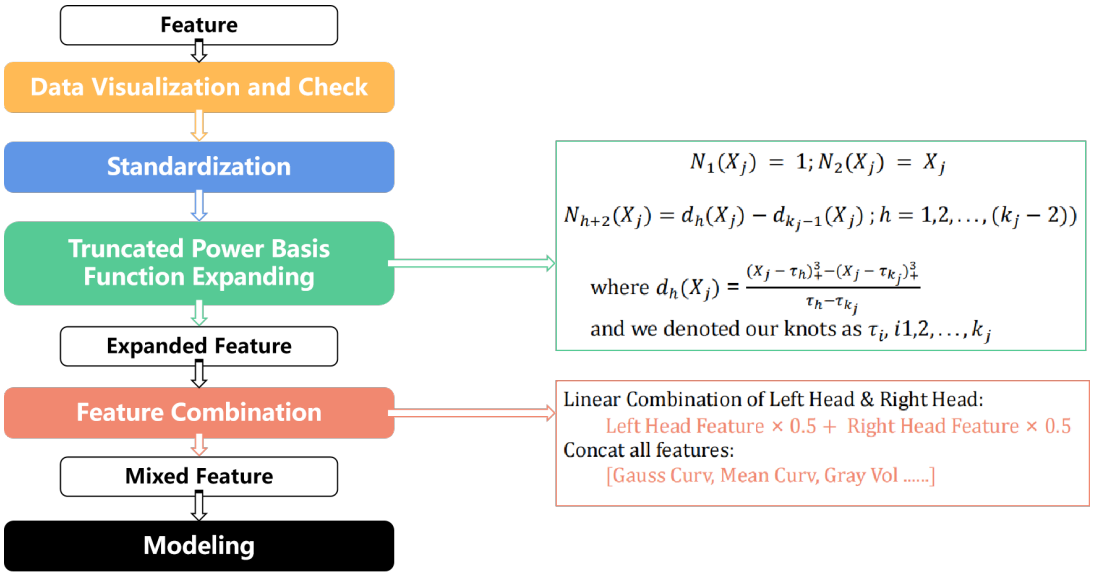

Fig. 1. Data processing flow, including data visualization, data augmentation and feature combination

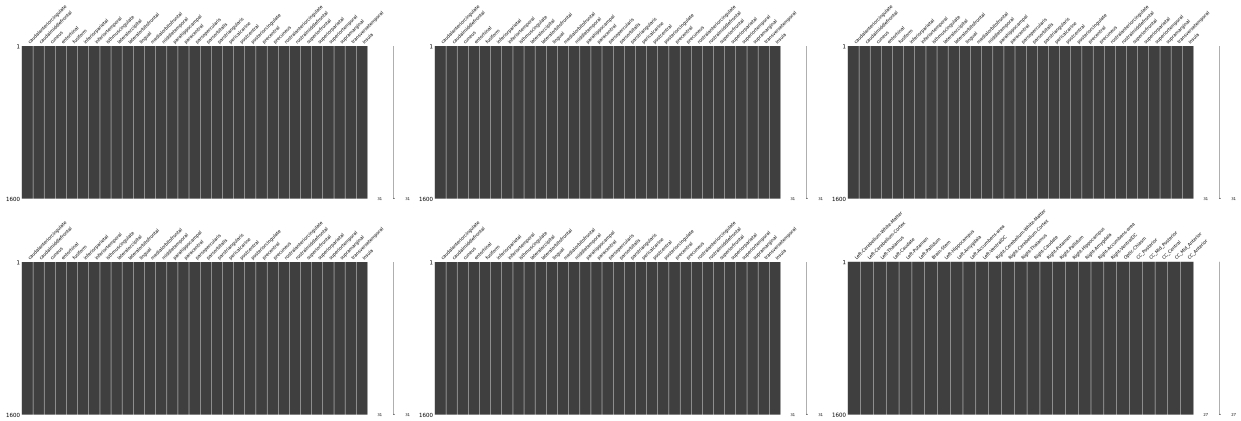

Fig. 2. Missing Value Check of the dataset

major machine learning models are also combined directly to obtain more robust and generalized results.

4.2 Residual Learning

Since there is often a certain gap between the predicted data and the real data, we want to make reasonable use of these residuals to further optimize the performance of our model. For a given model prediction y' and ground truth y of the first stage model, the residual can be expressed as:

$$r = y' - y \tag{6}$$

Then, we again build a new machine learning model in the second stage to predict the gap between the true value and the predicted value r' in the first stage, and finally the predicted output of the model can be represented as:

$$y_{final} = y' + r' \tag{7}$$

5 EXPERIMENTS

In the present study, a plethora of experiments were conducted to evaluate the model's performance on the validation set and test set. The outcomes presented in Table 1 highlighted the significance of feature engineering on the model's efficacy. The findings revealed that models utilizing truncated power basis function expansion and feature combination outperformed those without feature engineering. In the single model performance experiment, XGBoost and LightGBM exhibited impressive performance on the validation set but showed a certain overfitting phenomenon in the test data. Several attempts were made to modify the model's parameters and alter the feature engineering configuration, but the aforementioned issues persisted.

Thereafter, we constructed an ensemble model and a residual model based on the above model for comparison. For the ensemble

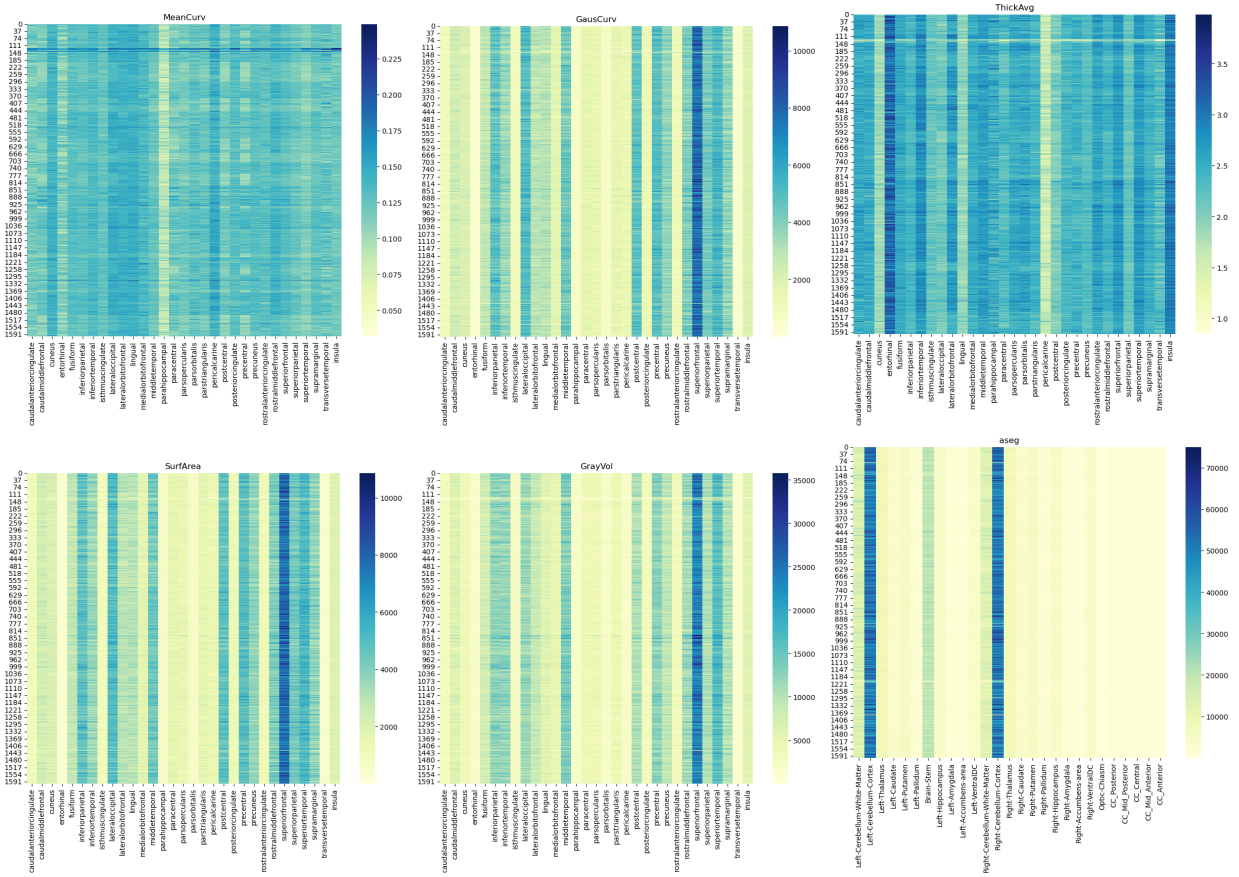

Fig. 3. Heat map visualization of the distribution of the dataset

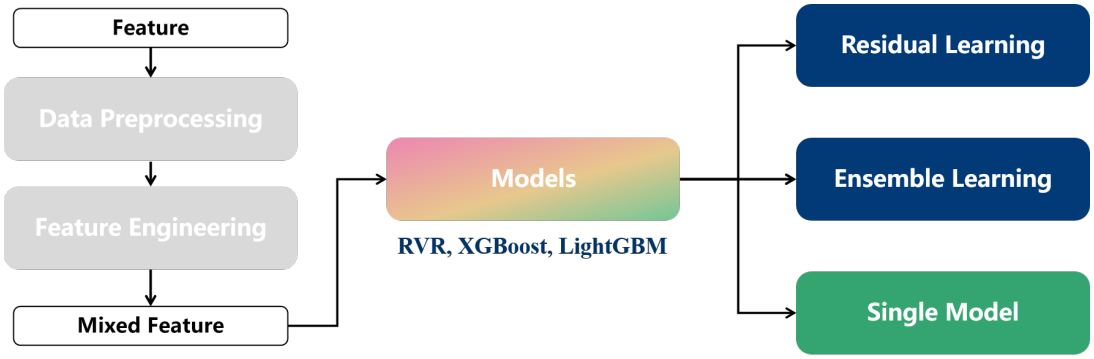

Fig. 4. Modeling flow, including single model, ensemble model and residual model

model, the results demonstrate that it does not significantly improve the prediction performance. Since we only use the simple averaging strategy in the ensemble model, the relatively poor base model will reduce the overall performance of the whole ensemble model. For the residual model, we build multiple models with different stages and embed different base models for each stage for comparison. As shown in Table 1, the residual model with 2 stages significantly outperforms the single model, while the model using all RVM as stage kernels achieves the best performance in test dataset. We also tried to add more residual stages to the residual model, but the results showed that the model performance became worse.

Table 1. Selected results of single model, ensemble model and residual model

Model	Feature Engineering	Valid set MAE	Test set MAE
RVR	Original Feature	5.32	7.54
	Feature Combination	4.11	7.22
	Feature Combination + Truncated Power Basis Function Expanding	3.96	7.01
XGBoost	Original Feature	7.94	9.72
	Feature Combination	7.80	9.26
	Feature Combination + Truncated Power Basis Function Expanding	7.89	9.33
LightGBM	Original Feature	8.66	9.81
	Feature Combination	8.44	9.36
	Feature Combination + Truncated Power Basis Function Expanding	8.10	9.07
Ensemble (RVR + LightGBM)	Original Feature	6.89	8.33
	Feature Combination	6.27	8.04
	Feature Combination + Truncated Power Basis Function Expanding	6.20	7.98
Ensemble (RVR + XGBoost)	Original Feature	6.13	8.02
	Feature Combination	5.64	7.86
	Feature Combination + Truncated Power Basis Function Expanding	5.44	7.70
Residual (RVR + RVR)	Original Feature	-	7.01
	Feature Combination	3.27	**6.84 (Rank 52/530)**
	Feature Combination + Truncated Power Basis Function Expanding	3.07	**6.33**

6 CONCLUSION

Based on the results of the aforementioned experiments, several conclusions can be drawn:

Firstly, it is evident that the ensemble and residual models exhibit superior performance when compared to their single-model counterparts. However, it is important to note that the number of models employed does not necessarily have a direct correlation with the mean absolute error (MAE) score.

Secondly, the original model features are relatively limited. Despite the application of various feature engineering techniques to address the overfitting phenomenon of XGBoost and LightGBM, this issue remains unresolved.

Finally, it is important to emphasize the vital role played by data processing and feature engineering in determining the upper limit of the effect. Although adjusting the model hyperparameters can lead to some improvement in the prediction effect, it is limited. On the other hand, utilizing more robust data preprocessing and advanced feature engineering methods can lead to a significant improvement in the score.

REFERENCES

Bastiaan R Bloem, Michael S Okun, and Christine Klein. 2021. Parkinson's disease. *The Lancet* 397, 10291 (2021), 2284–2303.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 785–794.

James H Cole and Katja Franke. 2017. Predicting age using neuroimaging: innovative brain ageing biomarkers. *Trends in neurosciences* 40, 12 (2017), 681–690.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* 20 (1995), 273–297.

Katja Franke, Eileen Luders, Arne May, Marko Wilke, and Christian Gaser. 2012. Brain maturation: predicting individual BrainAGE in children and adolescents using structural MRI. *Neuroimage* 63, 3 (2012), 1305–1312.

Yoav Freund, Robert E Schapire, et al. 1996. Experiments with a new boosting algorithm. In *icml*, Vol. 96. Citeseer, 148–156.

Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics* (2001), 1189–1232.

Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, et al. 2013. *An introduction to statistical learning*. Vol. 112. Springer.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems* 30 (2017).

Tae-Hwy Lee, Aman Ullah, and Ran Wang. 2020. Bootstrap aggregating and random forest. *Macroeconomic forecasting in the era of big data: Theory and practice* (2020), 389–429.

Peipeng Liang, Lin Shi, Nan Chen, Yishan Luo, Xing Wang, Kai Liu, Vincent CT Mok, Winnie CW Chu, Defeng Wang, and Kuncheng Li. 2015. Construction of brain atlases based on a multi-center MRI dataset of 2020 Chinese adults. *Scientific reports* 5, 1 (2015), 18216.

Robert E Schapire and Yoav Freund. 2013. Boosting: Foundations and algorithms. *Kybernetes* 42, 1 (2013), 164–166.

Michael Tipping. 1999. The relevance vector machine. *Advances in neural information processing systems* 12 (1999).

Junhao Wen, Elina Thibeau-Sutre, Mauricio Diaz-Melo, Jorge Samper-González, Alexandre Routier, Simona Bottani, Didier Dormont, Stanley Durrleman, Ninon Burgos, Olivier Colliot, et al. 2020. Convolutional neural networks for classification of Alzheimer's disease: Overview and reproducible evaluation. *Medical image analysis* 63 (2020), 101694.

David Wipf and Srikantan Nagarajan. 2007. A new view of automatic relevance determination. *Advances in neural information processing systems* 20 (2007).