Final Project

Abstract

Alzhymer's disease can be predicted with a good accuracy by examining thickness of cerebral cortex regions obtained through brain scans. In this report, I try to show different approaches and models that I tried to accurately predict Alzheimer's disease.

Due to the redundant information and high dimensionality of the data, models, even simpler ones tend to overfit, thereby necessitate regularization and dimensionality reduction. I used simple models of Elastic net, Logistic regression, Linear discriminant analysis (LDA), and Support Vector Machines (SVM), as well as ensembled tree based models including XGBoost and Random Forest models in this study to make comparison of best performing models. I also used pca and autoencoders as dimensionality reduction techniques, which then are fit within the models for all models except elastic net.

PCA and variational autoencoders have shown very similar information (variance) preservation and similar scores indicating close to linear relationship among features.

Elastic-net model has shown one of the best performances. Similarly simpler models logistic regression, LDA, SVM showed the best performance, while RF and XGBoost tend to generalize less with high risk of overfitting. Finally, LDA and logistic regression tend to have a high AUC score indicating a balanced accuracy of prediction for both classes, in addition to best overall accuracy.

Introduction

In this analysis, I tried to investigate which models help best predict Alzhymer's disease by investigating cerebral cortex thickness measurements at 360 brain regions (predictors). People with Alzheimer's disease have different cerebral cortex thickness in some regions of the brain as compared to that of the controls. Therefore, machine learning classification methods help in predicting (classifying) Alzheimer's disease using the 360 brain region cerebral cortex thickness measurements.

The dataset has 400 labeled observations with 360 continuous predictor variables, with continuous predictors and outcome class of AD (Alzheimer's disease) and C (Control).

There were no missing values in the dataset, we have a large class imbalance with 303 AD classes and 97 C classes. Therefore, I also considered precision, recall and f1-score as a metric as simple one unit accuracy metric may not show the full picture, especially in these types of cases with higher class imbalance.

Nature of Data:

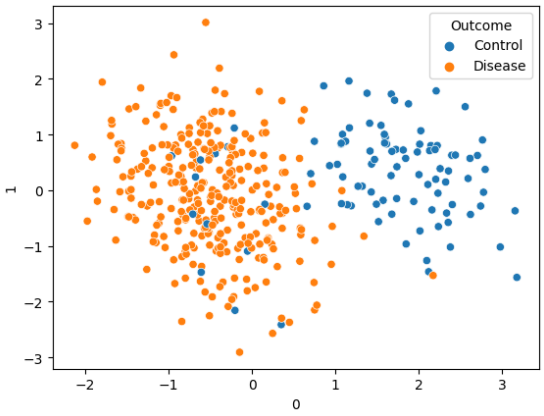


Fig 1: Visualizing nature of data using PCA (2 principal components)

We can see from figure 1 that the data seems to be linearly classifiable, hence we propose first to use simple linear models.

I have used 75% of the data for training set and the remaining 25% for validation set for all models. I did not use the test set provided for this report.

Data Analysis

I started implementing simple linear regression and Linear Discriminant Analysis (LDA) using raw data, and also support vector classifier. These models at first were prone to high overfitting and indicated the need to regularize the models as the data is high dimensional with much redundant information, and simple models like the ones mentioned above could easily overfit the model.

Dimensionality reduction and regularization

First Approach (Regularization)

To do the dimensionality reduction, we followed two steps. The first model we used is the Elastic net, which uses both Lasso and ridge to take advantage of the strengths of both regularization methods.

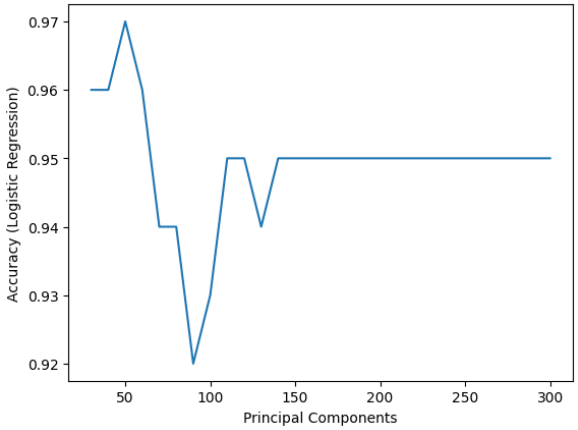
We did hyperparameter tuning for penalizing terms of alpha and L1 proportion.

Second approach (dimensionality reduction)

The second approach that I took was to perform dimensionality reduction techniques using Principal Components Analysis (PCA), which assumes linear relation among predictors, as well as autoencoders, which assumes nonlinearity of predictors. PCA is an unsupervised learning technique that reduces dimensions by assuming features as linear combinations of a transformed and reduced dimensions by estimating coordinate transformation by calculating Eigenvalues and their corresponding eigenvectors. Autoencoders on the other hand is a unsupervised learning (nonlinear dimensionality reduction) technique that is a deep neural network model that projects information into a latent (bottleneck) reduced dimension with minimal loss of important

information. These dimensionality reduction techniques not only reduces the number of features, but also remove unnecessary and redundant information (noise) and yield better accuracy scores.

To obtain the optimal dimensions that contain highest signal and minimal noise, I tried to tune hyperparameter (number of reduced dimensions) by benchmarking the accuracy score of a logistic regression model fitted over them. The result is found on the figure below.

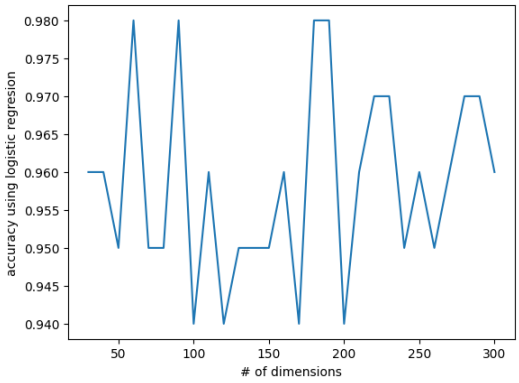


We can see that the accuracy of the mode peaks when using around 57 principal components, with explained variance ratio of about 68%.

This is possibly because the variability of the data (main information or signal) could be well explained with these 57 principal components and the remaining information is just noise.

Fig 1: PCA reduced dimensions parameter tuning.

Assuming nonlinear relationship among features, I also implemented autoencoders to reduce the dimensions with hyperparameters including number of hidden neurons, number of epochs, activation function and learning rate tunned. I also used early stopping to obtain maximum accuracy and reduce overfitting. Once hyperparameters are tuned, I iteratively tuned the bottleneck layer (latent dimension) to how much dimensions shall the features be reduced in order to obtain minimum mean square error and denoise the features.



As can be seen from the figure above, there are multiple dimensions that result in maximum accuracy (98%), i.e. around 66, 80 and between 160 and 190. I chose 66 dimensions resulting in 98% accuracy when piped to a logistic regression model.

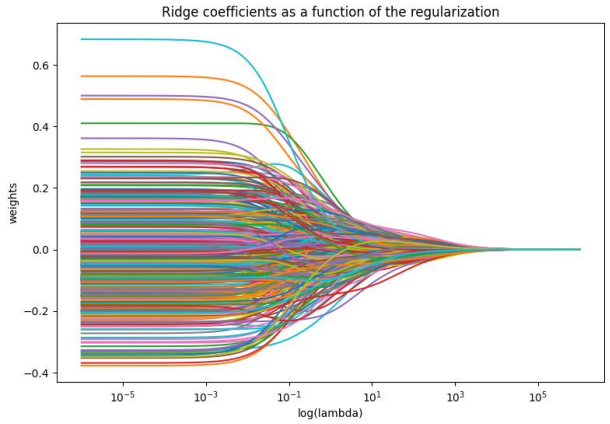
Fig 2: Variational autoencoders reduced dimensions parameter tuning

Once the dimensions are reduced, similar to what I did to the pca for dimensionality reduction, I fit the resulting 190 optimal bottleneck layer reduced dimensions to logistic regression and support vector classifier models.

Ensemble methods

Of the well-known ensemble methods, I implemented XGBoost (boosting) method to perform prediction in an attempt of getting better model prediction using the L1 regularization parameter alpha built within XGBoost, as the model is prone to overfitting. The other analysis I followed is to reduce dimensions using PCA and autoencoders and feed the results to XGBoost model, where comparison is made as well. Hyperparameters tuned were number of estimators, depth, learning rate, and alpha (regularization parameter).

Results



Results from the first approach using elastic net are found to be one of the best from the fitted models. I tuned the alpha and L1 ratio hyperparameters using cross validated grid search hyperparameter optimization and found the optimal hyperparameters to be 0.1 and 0 respectively. Hence, the optimal model is ridge as per the results.

Fig 3: Coefficients vs alpha plotted in a log scale.

I preferred to see accuracy together with classification report, which includes precision recall and f1 score. This is because in case of class imbalance, the accuracy score alone might be misleading and seeing class specific accuracy, f1 score as well as precision/recall might be important.

Table 1: Classification report for comparison of model performance

Model	Class	precision	recall	f1-score	accuracy
Elastic net	C	0.9	1	0.95	
	AD	1	0.98	0.99	
	Overall				0.98
Logistic Regression (Autoencoder)	C	0.9	1	0.95	
	AD	1	0.98	0.99	
	Overall				0.98
Linear Discriminant Analysis (Autoencoder)	C	0.9	0.95	0.92	
	AD	0.99	0.98	0.98	
	Overall				0.97
Support Vector Classifier (Autoencoder)	C	1	0.9	0.95	
	AD	0.98	1	0.99	
	Overall				0.98
Random Forest (Autoencoder)	C	1	0.7	0.82	
	AD	0.93	1	0.96	
	Overall				0.94
xgboost (Autoencoder)	C	0.94	0.8	0.86	
	AD	0.95	0.99	0.97	
	Overall				0.95

Table 1 above shows the model performance using classification report. Three approaches were taken as for the data to be used within each model as mentioned above, namely PCA reduced data, autoencoder reduced data, and non-reduced data. The performance summary of model comparison is list of best performing models as compared to different (dimensionally reduced/ unreduced) data.

The accuracy for training and validation set of the ridge regression (elastic-net with L1 ratio = 0) is **97% and 98%.** The training accuracy for the autoencoder reduced data fitted in logistic regression is **98%** whereas for the PCA is about **97%**. LDA using autoencoder has found to have a train and test accuracy of 95% and 97% respectively. SVM scored accuracy of **96%** for the training set and **98%** for the test set. Considering boosting ensemble technique (XGBoost) method, autoencoder reduction prior to fitting the model has been found to perform better with a test accuracy of 95%. Similar results were obtained with pca reduced model. Finally, random forest model was also implemented with autoencoder reduced data with tuned hyperparameter using and accuracy of **98%** and **94%** were obtained for training and test set. For all models, there is a very small to no difference in prediction with PCA and autoencoder, with no reduction and regularization performs worst.

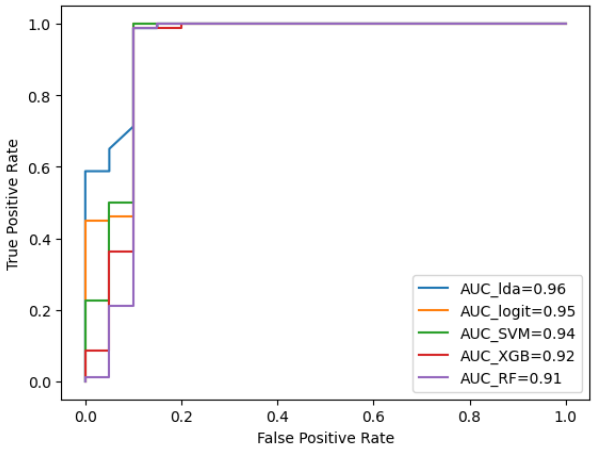


Figure above shows the AUC score for best performing models for the test set, which are, Logistic (logit) model, random forest, SVM, and xgboost classifier and lda, all reduced dimensions using autoencoder. LDA, logistic and SVM has a larger AUC score, followed by xgboost and Random Forest.

Fig: Model comparison with ROC curve with AUC score for best performing models with reduced data.

Conclusion

In this analysis, I tried to investigate which models help best predict Alzhymer's disease by investigating cerebral cortex thickness measurements at 360 brain regions (predictors). Due to the irrelevant and redundant features, and high dimension of the data, simple models tend to overfit and regularization and/or dimensionality reduction has been found useful. Therefore, I used two dimensionality reduction techniques, pca from linear reduction, and variational autoencoders from nonlinear reduction techniques were used before fitting candidate models. As the relation between features were almost linear, we have found similar results in both dimensionality reduction techniques. Simple models tend to perform better (LDA, Logistic regression and SVC) while complex ensemble methods tend to overfit and perform less on generalization.

Finally, LDA and logistic regression piped with autoencoder have the highest accuracy as well as AUC score, which indicates best balanced class-specific predictions (best precision and recall combination) for this data.