Machine Learning

Dementia Dataset

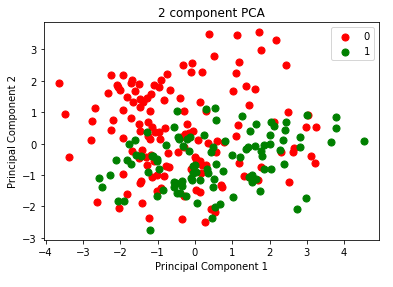
Michael Kwan

Nahom Molla

**Introduction**

The “OASIS-1: Cross-sectional MRI Data in Young, Middle Aged, Nondemented and Demented Older Adults” dataset was from Open Access Series of Imaging Studies (OASIS). As the title describes, this dataset contains MRI Data from various people that are diagnosed as having no dementia to severe dementia. The 416 subjects include both men and women as well as adults from various ages. The one similarity they have is that they are all right handed. The data for each subject includes the ID of the subject, their gender, their age, their education level, their socioeconomic status(SES), their mini mental state examination(MMSE), clinical dementia rating(CDR), estimated total intracranial volume(eTIV), normalize whole brain volume (nWBV), and the atlas scaling factor(ASF). The genders contain only male and female. The age is a number that ranges from 18 to 96 years old. The education level is categorized into numbers 1-5, where 1 is less than high school, 2 is high school grad, 3 is some college, 4 is college grad, and 5 is beyond college. The (SES) is also categorized between numbers 1-5, where each number represents a fifth of the population based on their socioeconomic status. For example, 1 is the lowest fifth and 2 is the next lowest fifth and so on. The CDR is what each subject has been labeled as for their dementia, where it goes from 0 to 2 in increments of 0.5 that show a spectrum of dementia from no dementia to severe dementia. The MMSE is an exam that was conducted on each of the patients that scores between 14 and 30. The eTIV ranges from 1123 to1992, the nWBV ranges from .64 to .89, and the ASF. This data set can be used to determine if any of the variables can be used to predict the CDR of the subjects. If a correlation is found between any of the variables and the CDR, it will mean that there is a way to try to predict dementia and be able to help find people that are more susceptible to dementia. We tried to use some unsupervised models such as the k-means and pca to find some more information. We also tried to use supervised models such as logistic regression, svm, and neural networks. We can find out if any of the models work well by seeing how accurate the models are at labeling or differentiating the different subjects into their dementia categories using the other variables.

**Unsupervised Models**



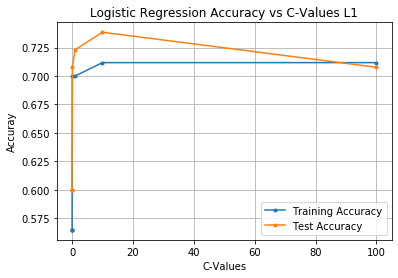
The figure above shows the two principal components that could be found from using PCA from the sklearn library where the red dots indicate the subjects without dementia and the green dots indicate the subjects with dementia. PCA is a dimensionality-reduction technique that can help simplify the problem by reducing the variables, but sometimes at the cost of accuracy. We used PCA to find two components that explain the most variance and try to graph it to see if we can make any new observations. The first principal component was [0.44711953, 0.15408128, 0.59221435, -0.27850459, -0.58996648] and the second principal component was [0.19625851, -0.70454301, 0.16766767, 0.63916591, -0.16868977] where each of the elements in the components correlate to the predictor variables respectively. The predictor variables are as follows: ['M/F', 'Age', 'eTIV', 'nWBV', 'ASF']. The first principal component was able to explain 0.50365773 of the variance and the second principal component was able to explain 0.32848104 of the variance. The first principal component explained the most variance, but since it was only 50%, it would not make sense to eliminate the rest because the first principal component is not enough to explain all the data and would result in significantly less accuracy.I think the most interesting observation that can be made is that the second principal component can be helpful in understanding that most people that end up with dementia are negative when the second principal component is calculated. According to the components, old age and low nWBV are the biggest indicators of dementia.

Another unsupervised model that was used was the k-means clustering, which can also be found in the sklearn library. It was used to find two clusters and the two centers were [-0.61270642 -0.12926271 -0.5826881 0.2027143 0.57914777] and [ 1.13642679 -0.01804274 0.9478226 -0.18305995 -0.93100618], which similar to the previous model used predictor variables as follows: ['M/F', 'Age', 'eTIV', 'nWBV', 'ASF']. The biggest differences in the centers of the two clusters lie in the gender, the eTIV, and the ‘ASF.’ This is interesting because it is the opposite of the results using the PCA because the two clusters are differentiated using different variables. It was able to achieve an accuracy of 0.661538 in determining if someone has dementia, so it is not a bad metric.

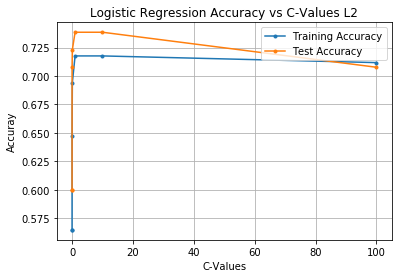
The use of PCA and k-means clustering indicates that there are no variables that are extremely useful in predicting dementia because there are many variables that have an impact on the result. Also, there are many people without dementia that can share all the same statistics and qualities as a person with dementia, so it is still very hard to make an extremely confident decision.

**Supervised Models**

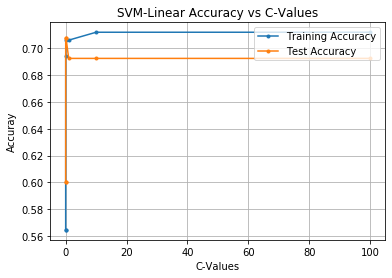
For the supervised part of our project we used Logistic Regression with lasso and ridge regularization, Support Vector Machines (SVMs) with linear, polynomial and radial-basis function kernels and a Neural Network model.



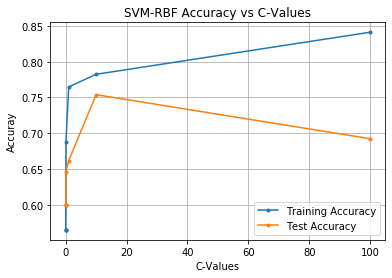
In the figure above, logistic regression with lasso regularization (L1-norm) was utilized using the ‘saga’ solver and these ‘c-values’: [0.0001, 0.001, 0.01, 0.1, 1, 10, 100]. After multiple iterations of this model we found that these are the best ‘c-values’; anything below 0.0001 and above 100 was reducing our accuracy. As done in the homework, the model was trained on the set of training examples provided to it and tested on both the training set and the testing set. As seen in the figure above, this model averages a testing accuracy of about 69%. An interesting aspect to note is that the testing accuracy is higher than the training accuracy for all values of ‘c’ under 100. The highest testing accuracy is achieved with a c-value of 10. This is an interesting trend because the model performs better on data that it has not seen beforehand as opposed to data it was trained on.



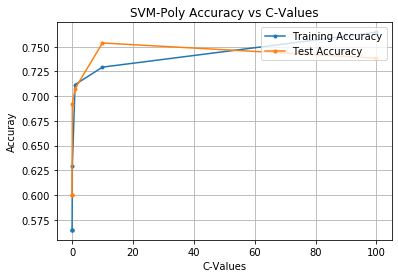
In the figure above, logistic regression with ridge regularization (L2-norm) was utilized using the ‘lbfgs’ solver and the same ‘c-values’ used with the lasso regularization: [0.0001, 0.001, 0.01, 0.1, 1, 10, 100]. Similarly, as observed in the lasso regularization, after multiple iterations of this model we found that these are the best ‘c-values’. This model was then trained on the set of training examples provided to it and tested on both the training set and the testing set. As seen in the figure above, this model also averages a testing accuracy of about 70%. The same interesting aspect to note is that the testing accuracy is higher than the training accuracy for all values of ‘c’ under 100, just like the lasso regularized model. The best testing accuracy for this model is achieved with c-values 0.1 and 10. As mentioned earlier, this is a surprising trend because the model is expected to perform better on the data it was trained on compared to data it has not seen yet.



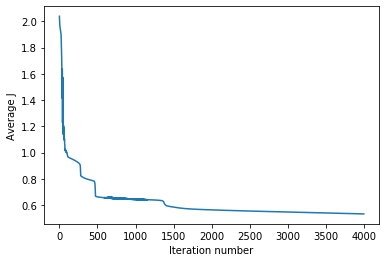
In the figure above, a linear kernel SVM was utilized using the same ‘c-values’ used with the lasso and ridge regularizations: [0.0001, 0.001, 0.01, 0.1, 1, 10, 100]. Similarly, as observed in both the lasso and ridge regularized models, multiple iterations of this model helped us conclude that these are the best ‘c-values’. This model was then trained on the same set of training examples we used for the previous 2 models. The model was then tested on both the training set and the testing set. As seen in the figure above, this model averages a testing accuracy of about 67% and a training accuracy of about 70%. The best testing accuracy of this model is attained with a c-value of 0.1. As opposed to the lasso and ridge regularized models, this model follows the expected trend since it performs better on the data it was trained on as opposed to data it has not seen yet.



In the figure above, a Support Vector Machine with a radial-basis function kernel with the gamma value ‘scale’ was utilized using the same ‘c-values’ used with the previous 3 models: [0.0001, 0.001, 0.01, 0.1, 1, 10, 100]. This model was then trained on the same set of training examples we used for the previous 3 models. The model was then tested on both the training set and the testing set. As seen in the figure above, this model achieves the best testing accuracy of about 75% with a c-value of 10. It averages a higher training accuracy compared to the testing accuracy. This model affirms expectations since it performs better on the data it was trained on as opposed to data it has not seen yet.



In the final SVM model we implemented, we used a polynomial kernel support vector machine with the gamma value ‘scale’. The same ‘c-values’ used with the previous 4 models were used with this model as well. The model was then trained on the same set of training examples used for the previous 4 models. The model was then tested on both the training set and the testing set. As seen in the figure above, this model attains the highest testing accuracy with a c-value of 10. An interesting trend noticed in this model is the fact that the testing accuracy is higher than the training accuracy for c-values less than 100. The fact that this model performs better on data it has not seen as opposed to the data it trained on, is still rather peculiar even though that trend is reversed when the c-value is 100.



In the figure above, a neural network was used and was implemented similar to the way it was done in the homework. A hyperbolic activation function was used with a regularization term because this yielded the best accuracy. The neural network is able to reduce the Average J, which was calculated using our cost function, and can be understood to be error. As seen in the figure, when there are more iterations in a neural network, the Average J decreases, but after a certain number of iterations, the Average J levels off. Various different neural networks structures were tested and the most accurate structure with 6 layers had an accuracy of .63. However, it should be noted that the neural network was used to predict more than just whether a subject had dementia or not. It was used to predict the severity of the dementia using the 0.5 increments that were used to label each of the subjects in the dataset. So, the results using the neural network were more precise than the results that only predicted whether or not the subject had dementia.

**Table of Results**

|  |  |  |
| --- | --- | --- |
| Lasso Regularized Model | | |
| C-values | Training Accuracy | Testing Accuracy |
| 0.0001 | 0.435 | 0.4 |
| 0.001 | 0.565 | 0.6 |
| 0.01 | 0.565 | 0.6 |
| 0.1 | 0.7 | 0.708 |
| 1 | 0.7 | 0.723 |
| 10 | 0.712 | 0.738 |
| 100 | 0.712 | 0.708 |

|  |  |  |
| --- | --- | --- |
| Ridge Regularized Model | | |
| C-values | Training Accuracy | Testing Accuracy |
| 0.0001 | 0.565 | 0.6 |
| 0.001 | 0.565 | 0.6 |
| 0.01 | 0.647 | 0.708 |
| 0.1 | 0.694 | 0.723 |
| 1 | 0.718 | 0.738 |
| 10 | 0.718 | 0.738 |
| 100 | 0.712 | 0.708 |

|  |  |  |
| --- | --- | --- |
| Linear Kernel SVM | | |
| C-values | Training Accuracy | Testing Accuracy |
| 0.0001 | 0.565 | 0.6 |
| 0.001 | 0.565 | 0.6 |
| 0.01 | 0.694 | 0.708 |
| 0.1 | 0.706 | 0.708 |
| 1 | 0.706 | 0.692 |
| 10 | 0.712 | 0.692 |
| 100 | 0.712 | 0.692 |

|  |  |  |
| --- | --- | --- |
| Radial-Basis Function Kernel SVM | | |
| C-values | Training Accuracy | Testing Accuracy |
| 0.0001 | 0.565 | 0.6 |
| 0.001 | 0.565 | 0.6 |
| 0.01 | 0.565 | 0.6 |
| 0.1 | 0.688 | 0.646 |
| 1 | 0.765 | 0.662 |
| 10 | 0.782 | 0.754 |
| 100 | 0.841 | 0.692 |

|  |  |  |
| --- | --- | --- |
| Polynomial Kernel SVM | | |
| C-values | Training Accuracy | Testing Accuracy |
| 0.0001 | 0.565 | 0.6 |
| 0.001 | 0.565 | 0.6 |
| 0.01 | 0.565 | 0.6 |
| 0.1 | 0.629 | 0.692 |
| 1 | 0.712 | 0.708 |
| 10 | 0.729 | 0.754 |
| 100 | 0.765 | 0.738 |

|  |  |
| --- | --- |
| Neural Network | |
| Layers | Accuracy |
| 2 | 0.523 |
| 3 | 0.553 |
| 4 | 0.584 |
| 5 | 0.615 |
| 6 | 0.63 |
| 7 | 0.553 |

**Why**

The unsupervised and supervised models yielded some interesting findings about the dataset that were able to illuminate some of the relationships between the various metrics that were recorded about the subjects in attempting to determine if they have dementia. The unsupervised models were used to try to break up all the complicated data into more simplified categories to see if we could separate the data into groups, but the k-means and PCA methods didn’t find perfect clusters or principal components that simplified the data perfectly. The k-means method was able to achieve an accuracy of 66% and the first principal component found in PCA only accounted for 50% of the variance. Depending on the way you looked at the data, you could make different categorizations, which shows that all the predictor variables play a role in determining the target variable. For example, the k-means clustering method implies that gender, eTIV, and ASF are the most distinguishing factors and the PCA method implies that age and nWBV are the most distinguishing factors.

The supervised learning models were also used to identify parameters that could be used to predict target variables, but using labels for each data point. A lasso regularized model and a ridge regression model were used for logistic regression and both yielded a test accuracy of .738, which indicates that both of the models are confident in making the right prediction most of the time. The linear kernel SVM, radial basis kernel SVM, and the polynomial kernel SVM yielded test accuracies of .708, .754, and .754. These accuracies are all pretty good, but the most interesting conclusion is that both radial basis kernel SVM and polynomial kernel SVM had the best accuracies, which may mean that the actual split between the data is a combination of the two transformations. In addition, there is the neural network, which was able to differentiate between the different levels of dementia with an accuracy of 63% when 6 layers were used, which displays how complicated the border is between the two classes. In all of the examples, the data collected also shows how having a margin that was too hard or too soft was not optimal in accuracy and it is important to not underfit or overfit the model since that would be reflected in a poor test accuracy even though the training accuracy is higher.

With this current dataset, the best models were able to achieve an accuracy of .754. However, keeping error types such as false positives in mind, perhaps it is better to assess someone as having dementia when they do not have dementia instead of assessing someone as not having dementia when they do have dementia. So, if we cannot find a model that can predict whether or not someone has dementia, we should find one that leans positive to make sure that people who have dementia are more often than not given the care that they need. For future work on predicting dementia, more predictor variables and more subjects would be useful to make more confident conclusions, especially for models that require more data such as the neural network.