CLASSIFICATION OF AUTISTIC SPECTRUM DISORDER SCREENING DATA

IST 707 Final Project

Francisco Nunez-Fondeur
frnunez@syr.edu

Introduction

Autism Spectrum Disorder refers to a range of conditions characterized by challenges with social skills, repetitive behaviors, speech and non-verbal communication According to the Centers for Disease Control, autism affects an estimated 1 in 59 children in the United States. In 2013 the American Psychiatric Association merged four distinct diagnoses into one under the name of Autism Spectrum Disorder, there were previously referred to as autistic disorder, childhood disintegrate disorder, pervasive development disorder-not otherwise specified (PDD-NOS) and Asperger syndrome.

For a long time, ASD was associated with a lengthy process to get properly diagnosed. With an increase in the number of ASD cases around the world, a faster screening tool was created called the Autism Spectrum Quotient (AQ) consisting of 50 questions. A condensed version the AQ-10 was created as a faster self-diagnosis tool to determine an individual's position on the autism-normality spectrum. While the AQ-10 is NOT used for a definitive diagnosis, a score greater that 6/10 would be a flag that you should seek a professional diagnosis.

Objective

While the tool is a great first step, there aren't many available datasets associated with clinical screenings and behavior. Per Fadi Fayez Thabtah, creator of the set, most available datasets on autism are genetic in nature. The set consists of the responses to behavior questions on the AQ-10 tool along with the results. In addition, 10 individual characteristics were made part of this set which have been used by the behavioral sciences experts for ASD detection. I intend to also use classification to determine the effectiveness of not just the AQ-10 tool but also the individual traits provided in diagnosing ASD by using classification and clustering algorithms. I will be using decision trees, Naïve Bayes, k-NN and Support Vector Machines.

Data Description

The data consists of three separate data sets available from the UCI Machine Learning Repository. The sets are divided by age; children (4-11), adolescents, (12-17) and adult (18 and over). The number of observations for each of the sets were 292 (children), 104 (adolescents), and 704 (adults) combining for 1100 total observations. The attributes were: ASD Diagnosis (our classifier), age (in years), gender, ethnicity, born with jaundice (Y/N), Family member with ASD (Y/N), Relation of the person completing the test, Country of Residence, Use of a screening application previously, age description, Questions 1-10 of the AQ tool (each separate), and the Screening Score on the AQ-10. The set was distributed as an arff file.

Data Preparation

There was some heavy amount of preparation that needed to be performed before I could perform any analysis. Because each of the three datasets were small (child, adolescent and adult) my first step was to merge the three data sets into one larger set. There were several NA values that existed in the age column as well as in some of the factor columns (for example ethnicity). For the age NAs, I used the avg age in each of the primary sets (child adolescent, adult) to replace the NA value before merging into one set. For factor columns I created an "Unknown Value". Several columns also needed to be recategorized to numeric value in order to be able to use them in the various algorithms.

Once I had one dataset, I did some additional cleaning of the data. The age column was eliminated since we already had a categorical "age description" column. Since research shows that ASD is not specific to country or ethnicity, I decided to remove those columns as well. Country of origin specifically had 89 different discrete values for this column, so this made the data easier to work with. The remaining values were converted into discrete values, I did this by creating separate variables for each columns option and using 1 and 0 to indicate Yes and No. Had I kept the countries and ethnicities; this would have resulted in over 100 additional variables.

Test and Training Data

I used the caret library to create my training and test sets. The data was randomized and 2/3 of the data was used for training (727 observations), while the remaining 1/3 use to test (373 observations).

Results

1 - Decision Tree

I created a Decision Tree using the c50 library's C5.0 algorithm. I first attempted to create a decision tree based on just the survey and final score. My intention was to see if there were any patterns with the answers, however I was surprised with the results. The resulting decision tree was completely based on the result of the survey, <=6 was no expected diagnosis of autism and >6 was expecting a Yes. Upon using the test data on the model, the model accurately predicted the outcome diagnosis 100% of the time. This raised a red flag as we should never expect 100% accuracy. All other subsequent classification algorithms also produced a 100% Accuracy in prediction, this caused me to want to NOT look at the questionnaire responses and instead look at all the other variables.

I created a new model using all the other variables excluding the survey question and results. The decision tree model created had a probability of being accurate 67% of the time. Upon running my test set. The tree was able to accurately predict the diagnosis 68.63% (237/373) of the time. Below is the resulting decision tree model. As you can see, the attributes that contributed the most to this tree was the age description (12-17 age group) and the relationship of the person who filled out

the form. It also seems that the algorithm is having problems correctly predicted “Yes ASD” outcomes. It correctly identified 80.83% of “No ASD” outcome but only correctly predicted 46.61% of “Yes ASD” outcome. Also, when attempting to boost, the number of trials stopped at 2 since the classifier was not accurate. There was no improvement in boosting. For this data, the decision tree is not a suitable tool for classification. There may possibly be additional variables not accounted for in this data that would aid in better classification via a Decision Tree.

```
C5.0 [Release 2.07 GPL Edition]       Wed Jun 12 03:09:34 2019
-------------------------------
Class specified by attribute `outcome'
Read 727 cases (18 attributes) from undefined.data

-----  Trial 0:  -----

Decision tree:
12to17 > 0: YES (68/26)
12to17 <= 0:
:...18Plus > 0: NO (473/129)
    18Plus <= 0:
    :...RelationRelative > 0: NO (12/3)
        RelationRelative <= 0:
        :...RelationUnkown <= 0: YES (141/66)
            RelationUnkown > 0: NO (33/11)

-----  Trial 1:  -----

Decision tree:
RelationUnkown > 0: NO (95.4/23.8)
RelationUnkown <= 0:
:...JundiceN <= 0: YES (87.2/39.9)
    JundiceN > 0: NO (544.4/212.8)

-----  Trial 2:  -----

Decision tree:
 NO (727/316.2)

*** boosting reduced to 2 trials since last classifier is very inaccurate
*** boosting abandoned (too few classifiers)


Total Observations in Table:  373

                  | predicted diagnosis
actual diagnosis |        NO |       YES | Row Total |
-----------------|-----------|-----------|-----------|
              NO |       194 |        46 |       240 |
                 |     0.520 |     0.123 |           |
-----------------|-----------|-----------|-----------|
             YES |        71 |        62 |       133 |
                 |     0.190 |     0.166 |           |
-----------------|-----------|-----------|-----------|
    Column Total |       265 |       108 |       373 |
-----------------|-----------|-----------|-----------|
```

2 – Naïve Bayes

Next, I applied the Naïve Bayes algorithm via the e1071 package. The Naïve Bayes performed better than the Decision Tree, I ran improved model with a laplace value of 3. The Naïve Bayes model performed significantly better than the decision tree. Since the classifier is not looking for interdependence, it was able to do a better job of classifying the data.

	predicted diagnosis		
actual diagnosis	NO	YES	Row Total
NO	230	10	240
	0.958	0.042	0.643
	0.987	0.071	
YES	3	130	133
	0.023	0.977	0.357
	0.013	0.929	
Column Total	233	140	373
	0.625	0.375	

```
Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = trainAut, y = trainAut$ClassASD, laplace = 3)

A-priori probabilities:
trainAut$ClassASD
       NO       YES
0.6423659 0.3576341

Conditional probabilities:
                ClassASD
trainAut$ClassASD         NO         YES
              NO  0.993657505 0.006342495
              YES 0.011278195 0.988721805

                f
trainAut$ClassASD      [,1]      [,2]
              NO  0.4175589 0.4936855
              YES 0.4461538 0.4980508

                m
trainAut$ClassASD      [,1]      [,2]
              NO  0.5824411 0.4936855
              YES 0.5538462 0.4980508

                JundiceN
trainAut$ClassASD      [,1]      [,2]
              NO  0.8693790 0.3373468
              YES 0.7807692 0.4145232

                JundiceY
trainAut$ClassASD      [,1]      [,2]
              NO  0.1306210 0.3373468
              YES 0.2192308 0.4145232

                FamAutismN
```

```
trainAut$ClassASD       [,1]       [,2]
              NO  0.9079229 0.2894448
              YES 0.8346154 0.3722439

                 FamAutismY
trainAut$ClassASD       [,1]       [,2]
              NO  0.09207709 0.2894448
              YES 0.16538462 0.3722439

                 RelationHealthcarePro
trainAut$ClassASD       [,1]       [,2]
              NO  0.01713062 0.1298972
              YES 0.03461538 0.1831562

                 RelationOther
trainAut$ClassASD        [,1]       [,2]
              NO  0.006423983 0.07997757
              YES 0.007692308 0.08753632

                 RelationParent
trainAut$ClassASD       [,1]       [,2]
              NO  0.2226981 0.4165034
              YES 0.3884615 0.4883404

                 RelationRelative
trainAut$ClassASD       [,1]       [,2]
              NO  0.03854390 0.1927117
              YES 0.05384615 0.2261492

                 RelationSelf
trainAut$ClassASD       [,1]       [,2]
              NO  0.5396146 0.4989627
              YES 0.4576923 0.4991677

                 RelationUnkown
trainAut$ClassASD       [,1]       [,2]
              NO  0.17558887 0.3808780
              YES 0.05769231 0.2336104

                 UsedAppN
trainAut$ClassASD       [,1]       [,2]
              NO  0.9785867 0.1449128
              YES 0.9730769 0.1621708

                 UsedAppY
trainAut$ClassASD       [,1]       [,2]
              NO  0.02141328 0.1449128
              YES 0.02692308 0.1621708

                 12to17
trainAut$ClassASD       [,1]       [,2]
              NO  0.04710921 0.2120996
              YES 0.17692308 0.3823396

                 18Plus
trainAut$ClassASD       [,1]       [,2]
              NO  0.7344754 0.4420857
              YES 0.4538462 0.4988255

                 4to11
trainAut$ClassASD       [,1]       [,2]
              NO  0.2184154 0.4136139
              YES 0.3692308 0.4835273
       'Positive' Class : NO
```

3 – Known Nearest Neighbor (k-NN)

Next I used the class package to apply the K-NN classification algorithm. The original unscaled version of the model I created only produced a 62.4% accuracy when tested. Like the Decision Tree Model, it has an issue with over reporting the “No-ASD” classification. I attempted to improve the model by rescaling using z-score standardization. This improved the accuracy slightly to 68.63%. The issue is still with the misclassification of the “Yes-ASD” outcome.

	knnAutSpread		
testAutlabel	NO	YES	Row Total
NO	186	54	240
	0.775	0.225	0.643
	0.684	0.535	
	0.499	0.145	
YES	86	47	133
	0.647	0.353	0.357
	0.316	0.465	
	0.231	0.126	
Column Total	272	101	373
	0.729	0.271	

	knnAutPrediction		
Total2z_test_label	NO	YES	Row Total
NO	247	26	273
	0.905	0.095	0.744
	0.744	0.743	
	0.673	0.071	
YES	85	9	94
	0.904	0.096	0.256
	0.256	0.257	
	0.232	0.025	
Column Total	332	35	367
	0.905	0.095	

4 – Support Vector Machine (SVM)

Lastly, I attempted classify using the kernlab package to use the SVM algorithm. When running the model using the vanilladot (basic linear) kernel, the model had a 65.68% accuracy in predicting the outcome. I attempted to improve the model's performance by using the rbfdot (Radial basis/Gaussian), however this resulted in less accuracy (64.87%)

```
Support Vector Machine object of class "ksvm"

SV type: C-svc  (classification)
 parameter : cost C = 1

Linear (vanilla) kernel function.

Number of Support Vectors : 504

Objective Function Value : -491
Training error : 0.337001
              | predicted number
actual number |        NO |       YES | Row Total |
--------------|-----------|-----------|-----------|
           NO |       220 |        20 |       240 |
              |     0.590 |     0.054 |           |
--------------|-----------|-----------|-----------|
          YES |       108 |        25 |       133 |
              |     0.290 |     0.067 |           |
--------------|-----------|-----------|-----------|
 Column Total |       328 |        45 |       373 |
--------------|-----------|-----------|-----------|


Gaussian Radial Basis kernel function.
 Hyperparameter : sigma =  0.0479670116236142

Number of Support Vectors : 510

Objective Function Value : -465.9183
Training error : 0.295736


             | predicted number
actual number |        NO |       YES | Row Total |
--------------|-----------|-----------|-----------|
           NO |       183 |        57 |       240 |
              |     0.491 |     0.153 |           |
--------------|-----------|-----------|-----------|
          YES |        74 |        59 |       133 |
              |     0.198 |     0.158 |           |
--------------|-----------|-----------|-----------|
 Column Total |       257 |       116 |       373 |
--------------|-----------|-----------|-----------|
```

Results

I found the results very odd. It seems that when taking into account the AQ-10 responses and score, all the algorithms were able to classify. ASD outcome 100% of the time. When looking at all the behavioral factors, there was a steep decline in the accuracy, with the exception of Naïve Bayes.

Including AQ-10 Reponses & Score		
Model	**Predicted Accuracy %**	**Actual Accuracy %**
Decision Tree	100.00%	100.00%
Naïve Bayes	100.00%	100.00%
k-NN	100.00%	100.00%
SVM	100.00%	100.00%
Excluding AQ-10 Reponses & Score		
Model	**Predicted Accuracy %**	**Actual Accuracy %**
Decision Tree	67.00%	68.64%
Naïve Bayes	98.93%	95.98%
k-NN	68.09%	69.75%
SVM	68.09%	68.09%

Conclusion

So how is this possible? Is the AQ-10 really that good of a predicter? Are there more factors at play than the variables in this data set? I looked up a few research articles, to see what information was available regarding the AQ tool. A study in the *Journal of Autism and Development Disorders* concluded that there is little difference in results of the AQ-10 and AQ-50 and the AQ-10 can be a potentially useful screening tool (Booth, etal 2013). A later study in *Psychological Medicine* concluded that not were AQ scores not a significant predictor of diagnosis for ASD, but that the 64% of those who scored below the cut off and had a "No-ASD" outcome were actually false negatives who were later in fact diagnosed with ASD (Ashwood, etal, 2016). This led me to look at the provided data sets more. Upon further inspection, I noticed that the data was bias. All individuals with an AQ score of >=6 later received a No diagnosis. There were no signs of any false positives or negatives. This explains why there was a 100% accurate prediction when using using the AQ-10 responses and score.

With regards to the other factors, I believe there are two issues at play. First there needs to be more data as there isn't a variance in results. Second, since ASD is still being studied I believe that there are several other variables at play which aren't included in this data set, and may prove as better predictors to classify ASD. I believe more data should be collected, including expanding the variables being recorded.

References

Ashwood, K. L., Gillan, N., Horder, J., Hayward, H., Woodhouse, E., McEwen, F. S., . . . Murphy, D. G. (2016). Predicting the diagnosis of autism in adults using the autism-spectrum quotient (AQ) questionnaire. Psychological Medicine, 46(12), 2595-2604. doi:10.1017/S0033291716001082

Booth, T., Murray, A. L., McKenzie, K., Kuenssberg, R., O'Donnell, M., & Burnett, H. (2013). Brief report: An evaluation of the AQ-10 as a brief screening instrument for ASD in adults. Journal of Autism and Developmental Disorders, 43(12), 2997-3000. doi:10.1007/s10803-013-1844-5

Dataset:

From UCI Machine Learning Repository

https://archive.ics.uci.edu/ml/datasets/Autistic+Spectrum+Disorder+Screening+Data+for+Children++#

https://archive.ics.uci.edu/ml/datasets/Autistic+Spectrum+Disorder+Screening+Data+for+Adolescent+++

https://archive.ics.uci.edu/ml/datasets/Autism+Screening+Adult