Classification of Autism in Adults using two Machine Learning Algorithms: K Nearest Neighbours and Support Vector Machine

Radiet Zenebe

Abstract

The purpose of this paper is to evaluate the performance of two machine learning algorithms namely, K Nearest Neighbour (KNN) and Support Vector Machine (SVM) on Autism Screening Adult data set uploaded on UC Irvine Machine Learning Repository. Since most screening tools that identify autism are very time consuming, the need to automate the process, so that individuals can screen themselves before diagnosis, is highly beneficial. I have computed multiple statistical measures to evaluate the performance of the two models. My analysis concluded that SVM is the most accurate and reliable model for the data set with an accuracy of 97.5%. Average Kappa for 10-fold cross validation also returned high performance of 0.95. However, further research should be implemented with a larger sample size and a more balanced data set.

1. Introduction

The UK's leading charity for Autism, the National Autistic Society (NAS) puts the number of people in the UK who are on the autism spectrum at 700,000 (NAS website: http://www.autism.org.uk/about.aspx). If one were to add their family members, this number increases to 2.8 million people for which autism is a part of their daily life (Ibid).

Despite autism being classified as a neurodevelopmental disorder, computed tomography (CT) scans or other types of medical tests are not used for its diagnosis. Instead, healthcare professionals diagnose the condition based on the patient's behavioural traits using several behavioural screening and diagnosis questionnaires. Most of these diagnoses and screening tools are extensive, long and, in most cases, require clinically trained staff to observe and carry out handcrafted mathematical algorithms (Thabtah, F., 2018). Therefore, the need to automate this process has emerged and in the past few years researchers have used machine learning for the classification purpose of Autism Spectrum Disorder (ASD) cases (Wall, D.P. et.al. 2012, Kosmicki, J.A. 2015). The main purposes of these machine learning researches can be summarised into three: less diagnosis time so people can access health care quicker, accurate prediction or classification and proposal of less features in the screening tools (Thabtah, F., 2018).

In this paper, I intend to evaluate the performance of two machine learning algorithms on the recently uploaded data set of Adults with Autism on UC Irvine Machine Learning Repository by Thabtah Fadi. The data set is based on Autism-Spectrum Quotient (AQ), a self-administered questionnaire which measures and situates users on a continuum from normality to autism **(**Ecker, C. et. al. 2010**)** and other individual characteristics that are deemed important in classifying ASD cases (Thabtah F., 2017). After a brief critical review of literatures on the application of machine learning algorithms on ASD data set, I will perform exploratory analysis using chi square test and correlation matrix which will be my basis for reducing unnecessary attributes. Subsequently, I will utilize K Nearest neighbour and Support Vector Machine models and evaluate their performance using false negatives, false positives and accuracy. I will also apply Kappa statistics for a better performance measure. To validate their performance ability on future data, I will apply 10-fold cross validation and proceed with a discussion of the outcome and conclude with the limitations of this paper.

2. Method

In this paper, I will apply machine learning models using the free open source statistical software R version 3.4.3. Machine learning uses computer algorithms to change raw data into intelligent knowledge (Lantz, B., 2013). I will use two machine learning algorithms, K nearest neighbour (KNN) and Support Vector Machine (SVM) for distinguishing ASD cases from non ASD cases.

This is a supervised classification problem since the data set already contains the final class of the observation with the aim of accurately classifying the test data set by learning from the training data set. In other words, I will divide my data set into training and test data set for teaching the algorithm the correct classification and testing it with the remaining dataset.

KNN model works by identifying K nearest records in a training data set by calculating mostly Euclidian distance and applying majority vote to classify an unlabelled test data set. It requires a training data set, a test data set, a vector with the correct class of training data set and a value for K (James, G. et.al, 2013).

I opted for the KNN algorithm as it makes no assumption about the data distribution (Lantz, B. 2013). For example, the dataset referred to in this paper has no apparent difference in the weight of each attribute. Hence, a distance-based classification is reasonable considering the varying features which are not necessarily more important than one another. However, KNN is easily affected by noises and outliers. It also requires some pre-processing for nominal data and missing values. (Lantz, B., 2013)

Firstly, I converted the categorical data to numeric and tried several techniques to improve KNN model performance. Two normalization techniques, min-max (MM) and soft-max (SM) normalization techniques, were tried. Normalization is a technique used to put different numerical measurements on the same scale to optimise the learning process of the algorithm. Second, I have tried several K values (K=1,5,9,13,17,21,25) and selected the most accurate K. I have also used a function called *knncat* which takes both categorical and numeric data and identifies the best K value from a set of random Ks supplied for training the model.

I then tabulated a confusion matrix for evaluating accuracy (correctly classified cases), false negative (FN) and false positive (FP) predictions of the model. A confusion matrix for binary classification problems shows the false negatives (Actual ASD cases predicted as not ASD), false positives (actual not ASD cases predicted as ASD cases) as well as correctly predicted ASD cases (true positives) and correctly predicted non ASD cases (true negative). Please see table 1.

The ideal model should perform closer to 100% accuracy with minimal false negatives and false positive classification errors.

	Model prediction	
Actual test data	Not ASD	ASD
Not ASD	TN	FP
ASD	FN	TP

Table 1. Confusion matrix for the binary classification of ASD cases

Accuracy is the percentage of correctly classified cases and can be calculated with the equation: Accuracy (%) = (TP+TN)/(TP+TN+FP+FN). False negatives for an ASD case can be risky for the AQ user as it will lead them to believe that they do not need to seek further diagnosis when they should. In contrast, FP can create a burden for clinical professionals admitting patients who do not need clinical diagnosis. In my view, qualitatively analysing for the least amount of FN and FP cases while optimizing for maximum amount of TN and TP rate gives the best model.

Similar to the KNN model, SVM can be used for classification as a supervised learning algorithm. It works by creating a flat boundary called hyperplane between two different homogenous groups. The power of SVM lies in its ability to create a higher dimension to a nonlinear data set using a process called kernel trick (Lantz, B., 2013). I decided to use SVM because it has been previously applied successfully for the ASD classification problem (Ecker, C. et.al. 2010, Kosmicki,et al., 2015).

Although there are many different kernel functions that could be applied for the ASD data set, I have utilized Gaussian RBF kernel with *"rbfdot"* function as it is the most popular convention (Lantz, B., 2013). Diagonal summation of the result of the selected SVM function gives the accuracy of the model which I subsequently used to compare with the previous model (KNN).

Because of the class imbalance in the data set (see data exploration), I have used *Kappa* statistics to evaluate the performance of the two models. Kappa statistics adjusts the accuracy by considering those correct predictions which could have resulted by chance alone (Ibid). In other words, Kappa statistics boosts a classifier if it has performed better than simply picking up the most frequent class. Kappa ranges between 0 to 1 where values closer to one have very good agreement between model prediction and true values of the data set. The best model for the data set should therefore score Kappa close to 1.

Finally, I have performed model validation for both KNN and SVM using K-fold cross validation (CV) with (k=10). One method to verify the issue of over-fitting to the training data and validate the predictive performance of the machine learning model is to use validation techniques (Rogers, S. et.al, 2016). K fold CV is one of the industry standard techniques for measuring performance of machine learning models on unseen data. (Lantz, B., 2013). It creates random partitions called folds that is then used to train and test the model. I have used 10-fold CV since it is the most common convention. (Ibid). Diagram 1 below depicts the process flow applied in this paper.

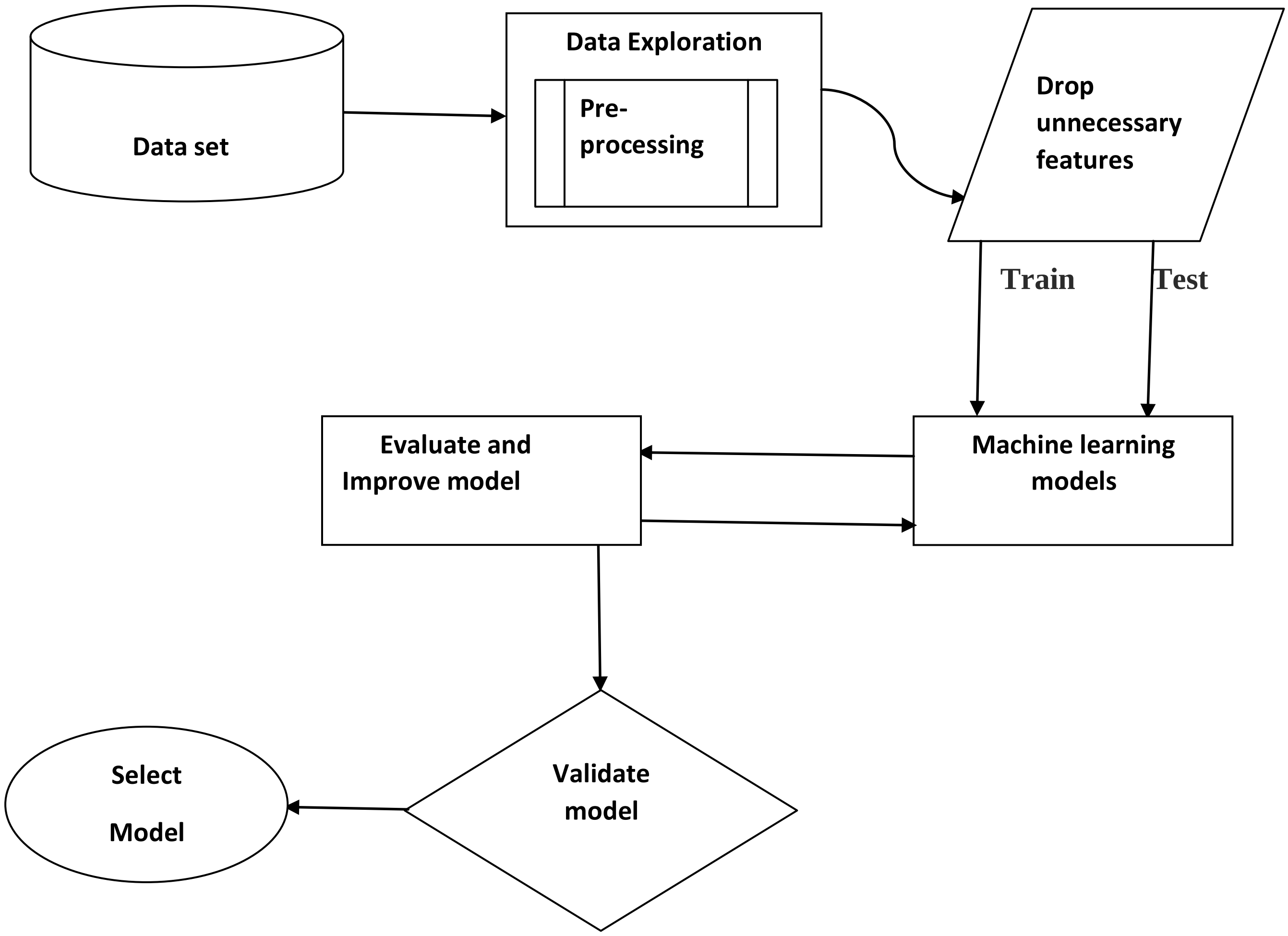


Diagram 1: Flow chart of methods applied.

3. Data

The data is obtained from the UCI Machine Learning Repository on Autism Screening Adults Dataset sourced by Fadi Fayez Thabtah. A data set and a separate description of the dataset was provided by Thabtah. The data set has 704 observations and 21 features. The features in the data set can be grouped into two attribute types: the first ten features are based on the behavioural questionnaire of Autism Spectrum Quotient AQ-10 Adults, available at the University of Cambridge, Autism Research Centre (https://www.autismresearchcentre.com/arc_tests) and the next ten features are individual characteristics which are thought to be significant for ASD classification (Thabtah 2017). The binary classifying feature is also included as part of the data set (See table 2).

The original data set was provided in Attribute-Relation File Format (ARFF) which I converted to CSV file by using Weka open source software. There are some values within the attributes age, ethnicity and relation which were indicated as "?". Before proceeding with R, I have manually replaced these values by "NA" to represent missing values.

To avoid confusion during analysis, I have also manually changed the 15th feature which is typed as "austim" to "Family_ autism" to represent the values of whether a family member has PDD (pervasive developmental disorder) or not, based on the description folder provided.

Attribute	Class	Number of levels
A1_Score	integer	2
A2_Score	integer	2
A3_Score	integer	2
A4_Score	integer	2
A5_Score	integer	2
A6_Score	integer	2
A7_Score	integer	2
A8_Score	integer	2
A9_Score	integer	2
A10_Score	integer	2
age	integer	2
gender	character	2
ethnicity	character	10
jundice	character	2
family_autism	character	2
contry_of_res	character	67
used_app_before	character	2
result	integer	11
age_desc	character	1
relation	character	5
Class.ASD	character	2

Table 2: List of attributes

4. Exploratory Analysis

Originally, there were 704 observations and 21 features in the dataset. The first ten features are classed as integers. These are the AQ questioners with binary values of two levels. Hence, I have changed these features to factor variables. Although the feature “result” is classified as integer, it is the sum of the scores from the first 10 features. The scoring method is clearly explained on the Cambridge University Research Centre website. Consequently, I have changed the “result” to a factor.

Out of the 704 observations, 515 are classified as non ASD cases while 189 are ASD cases. This demonstrates the imbalance in the data set with 73% non ASD cases and only 27% ASD cases. The high imbalance of the outcome variable suggests that caution should be taken selecting the train and test data set before training the model to guarantee a fair distribution of ASD cases.

The only numeric variable in the dataset is “age”. As the dataset is for adults, the minimum age is 17 while the maximum age is 383, clearly an outlier and a noise for the dataset. It was excluded from analysis by applying (n<150) as there is no individual with age 150 or more. The maximum age is now 64.

Box plotting age (figure 1) shows that 50% of the age group is between 20-35 years. To get a more clear understanding of the age distribution, I plotted density graph on a histogram (figure 2). The density plot illustrates that age is highly skewed to the right conveying that most of our data values (more than 50%) are concentrated on the lower level of the age group (n<30).

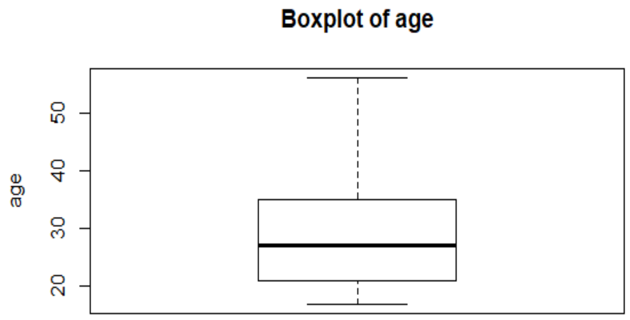


Figure 1: Boxplot of age

Figure 2: Density plot of age

An exploration of the feature ethnicity reveals that white European is the most common ethnicity group at 33%. The rest of the ethnic groups vary between 0.9 and 17.9%. There are 95 counts of missing values for this variable which need to be removed before applying the KNN and SVM models.

Gender is fairly distributed with 47.8% females and 52.2% males. However, a larger proportion of females (30.74%) were classified ASD compared to males (23.49%).

I computed binary cross tabulation and chi-square test to explore the correlation of each attribute to the outcome feature. Chi square is a statistical test for categorical data which can be used as a correlation test between two nominal variables. P value <0.05 suggests that a feature is not an independent feature but somehow associated with our outcome feature.

Feature	**Chi-Square**	**P-Value**	**Heterogeneous Correlation Matrix (Independent variable to Outcome)**
A1_Score	60.262	8.30E-15	0.44
A2_Score	66.676	3.20E-16	0.41
A3_Scroe	133.63	2.20E-16	0.52
A4_Score	151.91	2.20E-16	0.58
A5_Score	200.27	2.20E-16	0.68
A6_Score	242.26	2.20E-16	0.63
A7_Score	85.241	2.20E-16	0.42
A8_Score	37.528	9.01E-10	0.33
A9_Score	279.55	2.20E-16	0.66
A10_Score	102.64	2.20E-16	0.5
age	numeric	numeric	0.16
gender	4.3065	0.03797	-0.11
ethnicity	86.645	7.65E-15	-0.15
Jundice	6.461	0.01103	-0.15
Family with Autism	20.697	5.38E-06	-0.2
Country of residence	181.14	1.13E-12	-0.25
Used app before	0.68856	0.4067	Dropped
result	701	2.20E-16	Dropped
Age description	148.83	2.20E-16	Dropped
relation	1.1723	0.8826	Dropped

Table 3: Correlation matrix

Most features have a significant association with the outcome feature except the “used app before” and “relation” features which have a P value greater than 0.05, indicating no association between the two features and the outcome class “Class.ASD”. Therefore, I have dropped both features when training the model. I have also dropped the feature “age description” as it has only one value for all instances i.e. “18 and more”. Choosing appropriate features and dropping duplicate and unrelated features will help our model focus on the significant associations and improve its performance (Valiant, L., 2013).

From the cross tabulation, the feature “result” is unambiguously classified as indicated by the University of Cambridge, Autism Research Centre (https://www.autismresearchcentre.com/arc_tests), meaning that a total score (result) of less than or equal to six is classified as non ASD while a result greater than six out of 10 is classified as ASD. Even though “result” has a significant P value, I have omitted it from the analysis as it is measuring the same character as the outcome feature.

Since both KNN and SVM utilize a distance function to select the best learning model, it is necessary to convert categorical variables to numeric. For variables which are mutually exclusive with only two levels, I have applied dummy coding (0,1). However, features “ethnicity” and “country of residence” have more than two levels: ten and sixty-seven levels respectively. While “ethnicity” is feasible for manual conversation, it is not practical to convert “country of residence” individually. Therefore, I have grouped them into their respective continents which reduced the levels to 6.

To avoid subjectivity during conversion of these variables, I have initiated the conversion with the most frequent level, scaled as 1, and building up the scale accordingly. See table 4 for conversation of categorical data to numeric.

Ethnicity			**Gender**	
Levels	**Number of instances**	**numerical conversion**	**Levels**	**numerical conversion**
White-European	233	1	Female	0
Asian	123	2	Male	1
Middle Eastern	92	3	**Jundice**	
Black	43	4	**Levels**	**numerical conversion**
South Asian	36	5	Yes	0
Others	31	6	No	1
Latino	20	7	**Family Autism**	
Hispanic	13	8	**Levels**	**numerical conversion**
Pasifika	11	9	Yes	0
Turkish	6	10	No	1
Country of Residence			**Used app before**	
Levels	**Number of Instances**	**numerical conversion**	**Levels**	**numerical conversion**
Asia	276	1	Yes	0
Europe	148	2	No	1
North America	140	3		
Australia/Oceania	110	4		
South America	16	5		
Africa	11	6		

Table 4: Numeric conversion of categorical data

After converting the categorical variables to numeric, I computed heterogonous correlation matrix using *hetcor()* function to assure that there is no identical feature measuring the same character as outcome feature (see table 3). Correlation greater than 0.9 is an almost similar measurement to the outcome feature. For this data set however, there is no variable with correlation more than 0.9 indicating that no more feature reduction is required.

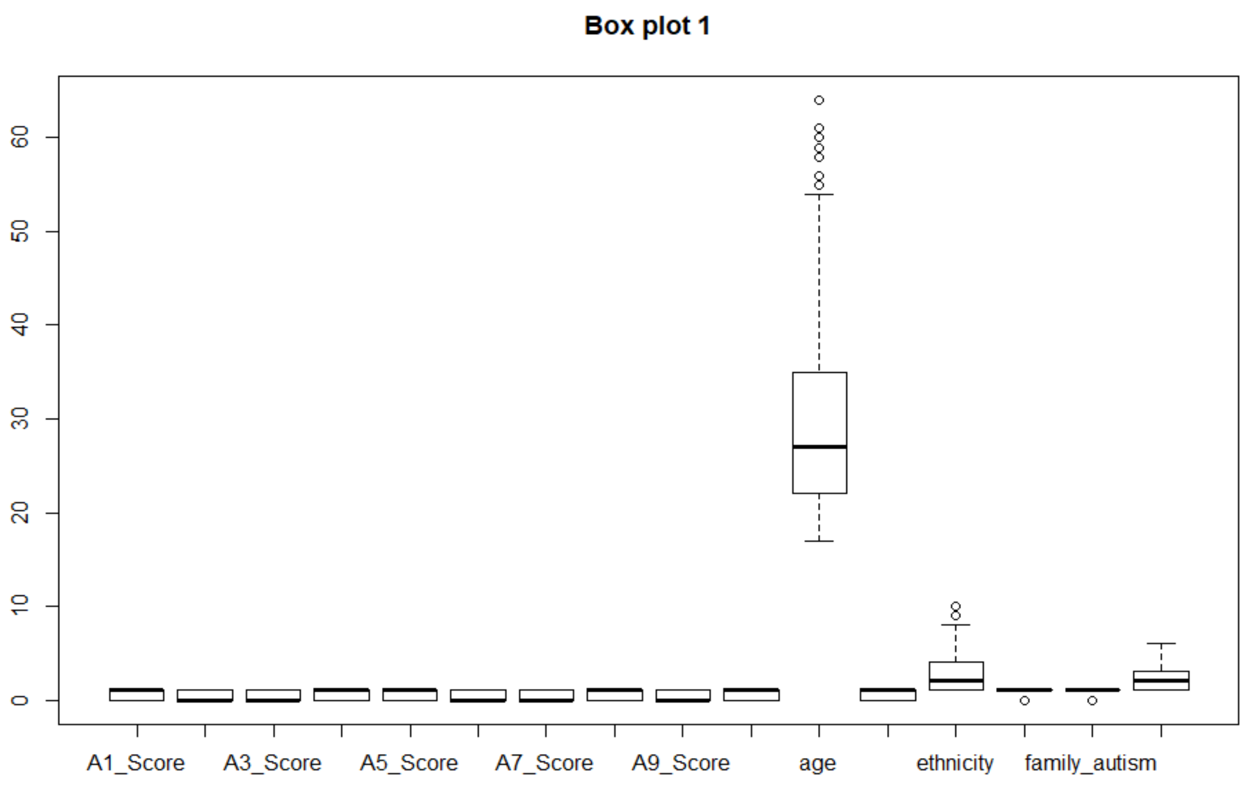
5. Application of Machine Learning Algorithms

K nearest neighbours (KNN)

I have taken the following steps to prepare the dataset for KNN model training:

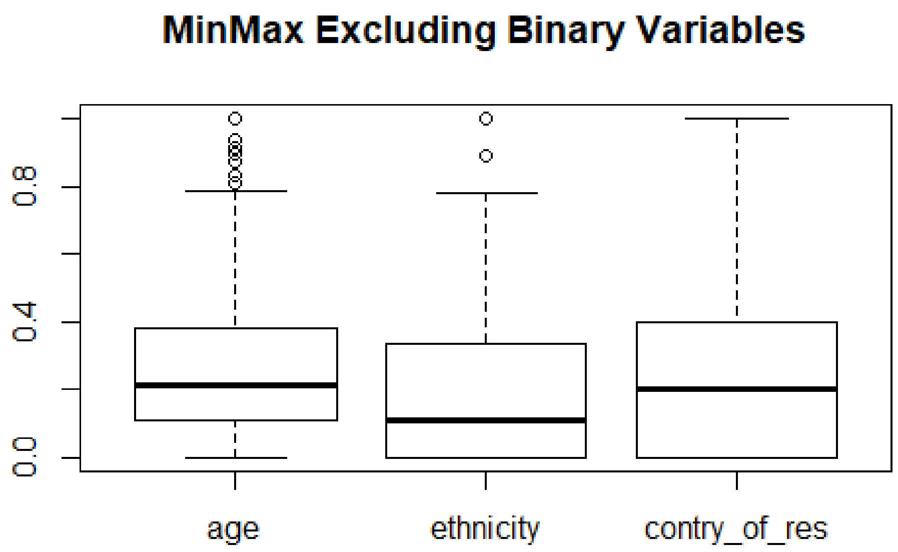
1. Exclusion of all missing values. Altogether, 192 instances with “NA” were omitted reducing the observations to 608 instances.
2. Since there is no information about the sampling technique used for this dataset, I have first randomized the data using the *set.seed (12345)* function in R.
3. Because box plotting showed variables with uneven distribution, I have normalized all the features using min-max normalization.

Box plot 1 shows that the variable “age” highly influences the other variables and contains outliers to the high-level age group (figure 5.1.1.)



Boxplot 1: Boxplot of all variables

Boxplot 2 depicts a box plot of the three variables with more than two levels which still showed outliers in the variables “age” and “ethnicity”.



Boxplot 2: Boxplot of non-binary variables

Two sampling methods for selecting train and test data set were implemented to confirm proportional distribution of the outcome variable. In the first sample, I have selected the first 80% of the dataset for training and the last 20% for test. In the second sample, I have selected the last 80% of the dataset for training and the remaining for test (see table 5).

	First 80% for training		Last 80% for training	
	Not ASD	ASD	Not ASD	ASD
Training set	71.6	28.4	71.5	28.5
Test set	65.6	34.4	65.8	34.2

Table 5: Two different sampling techniques for train and test split

Since the percentages of ASD and non ASD cases for both training and test data set are similar for the two sampling techniques, I found it reasonable to train the model on the first 80% of the data.

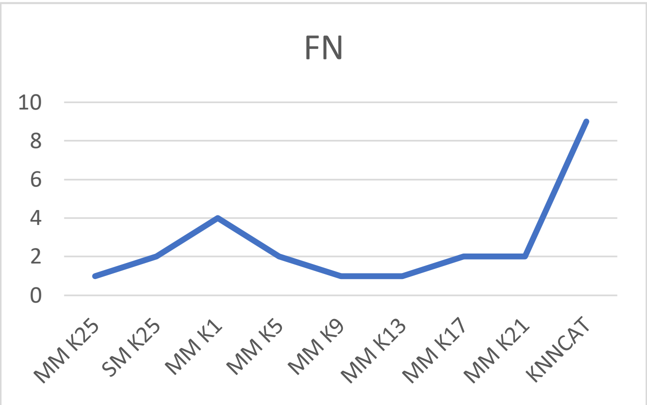
I first used K=25 nearest records taken from the square root of 608 to train and inspect the model. K=25 resulted in 1 FN and 3 FP with 96.72 % accuracy. To improve the model, I applied soft-max normalization (SM) and tested the model again. However, there was no improvement on the accuracy but there was an even greater FN error. I then tried several numbers of K values (K=1,5, 9, 13, 17 and 21) and evaluated FN, FP and accuracy. The best K value is K=9 which resulted with 1 FN, 1 FP, and an accuracy of 98.36%. Kappa for this K was 0.9637, indicating high agreement between model prediction and true actual data (see table 6).

Model optimization	False Negative (FN)	False Positive (FP)	Incorrectly Classified	Accuracy %	Kappa
MM normalization K=25	1	3	4	96.72	0.9282
SM normalization K=25	2	2	4	96.72	0.9282
MM normalization K=1	4	4	8	93.44	0.8548
MM normalization K=5	2	7	9	92.62	0.8411
MM normalization K=9	1	1	2	98.36	0.9637
MM normalization K=13	1	2	3	97.54	0.9458
MM normalization K=17	2	3	5	95.90	0.9097
MM normalization K=21	2	2	4	96.72	0.9282
"Knncat" K=21	9	8	17	83	

Table 6: Testing different K values

Another KNN function that could be used for the data set is *knncat* which takes categorical and numeric values to determine the best K value for a given number of Ks. Hence, I did not convert variables to numeric for this function. I have then followed the same steps as the KNN model except that this time I have only normalized the numeric variable age. This package predicted that K=21 with 9 FN and 8 FP and an accuracy of 83% is the best model. However, this is a poor performance compared to the numeric K=9 value.

The graphs below visualise the results in table 6, depicting relationship of each K value with the four statistical tests. They show a general trend of poor performance of *knncat* and high performance of K=9 in all the statistical measurements.



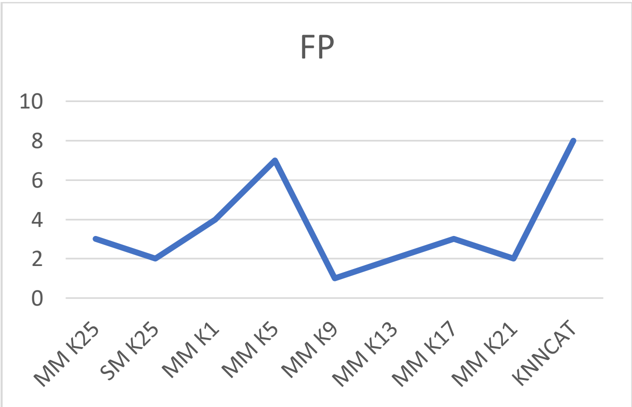
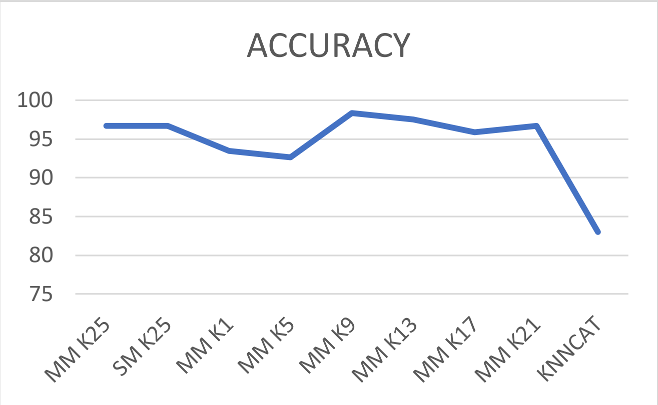


Figure 3: relationship of K values to FN

Figure 4: relationship of K values to FP



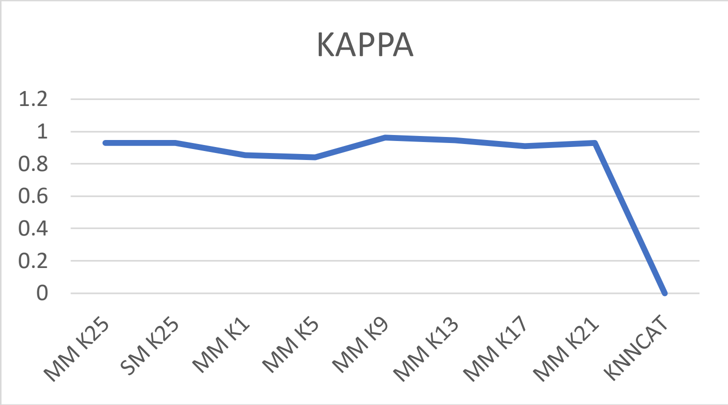


Figure 5: relationship of K values to Accuracy

Figure 6: relationship of K values to Kappa

Support Vector Machine (SVM)

Before using support vector machine, similar pre-processing stages were implemented including omitting missing values, randomization and min-max normalization. However, the outcome feature is converted to factor to train SVM. After pre-processing the data, 608 observations and 16 features were divided into training and test datasets. In parallel to the KNN model, I selected the first 80% for training and the last 20% for test data after randomizing the dataset using *set.seed (123)* function. See table 7 which represents a fair distribution of ASD to Not ASD cases.

Set.seed (123)	Classification		
	ASD	Not ASD	Not ASD to ASD percentage
Actual data set	70.4	29.6	42.04%
Training data set	70.2	29.8	42.45%
Test data set	71.3	28.7	40.25%

Table 7: Distribution of ASD and non ASD cases

Although there are many kernels available, a popular convention is to start with the Gaussian RBF kernel, which has a good performance reputation (Lantz, B., 2013). Consequently, I have applied *"rbfdot"* kernel for the ASD data set. The kernel performed with 97.5% accuracy, 0 FN, 3 FP and a Kappa of 0.9383. Normally, choosing Kernels can be time consuming. However, in this case, the Gaussian RBF kernel performed with a high accuracy and no FN.

Validation

To verify future predictive performance of the two models, I applied 10-fold cross validation to the two best learning models KNN (K=9) and SVM (Guassian rbf kernel). After creating 10 folds and executing the Kappa for each fold, I then computed the average Kappa value for comparison. The

Kappa for SVM returned 0.95, while KNN returned 0.19. Interestingly, the validation reveals that KNN (K=9) has overfitted to the training data set and showed poor predictive performance. This leads to bias variance trade off (Rogers, S. et.al 2016) meaning that finding the optimal model that will generalize well without overfitting to the training set is crucial. To verify if K would perform better with a higher value, I have run cross validation on K=25 as the next best model which resulted with a Kappa of 0.20, almost the same score as K=9 value. See table 8 for multiple statistical scores of both models.

Model	False Negative	False Positive	Accuracy (%)	Kappa	10-fold Cross Validation mean Kappa
KNN(K=9)	1	1	98.36	0.9637	0.19
KNN(K=25)	1	3	96.72	0.9282	0.20
SVM(G. RBF)	0	3	97.5%	0.9383	0.95

Table 8: KNN and SVM model comparison

6. Discussion and Conclusion

Discussion

The main insight from the result of this paper is that high performing models are not necessarily a good predictor in a future dataset. (See average Kappa of 10-fold cross validation for K=9). This paper has utilized multiple statistical tests to evaluate the performance of KNN and SVM for the classification problem of ASD to non ASD cases.

The first, simple but widely used machine learning model utilized was KNN. Despite being simple, KNN can be very powerful and has been used successfully in various applications (Lantz, B., 2013). After trying several K values and normalization techniques, K=9 performed better than any other K values and the SVM model in terms of false negative, false positive and accuracy.

Although trying different K values and selecting the best K value is widely performed for model improvement, I also used *knncat* library, which takes categorical variables without changing them to numeric. This resulted in K=21 being the best classifier but it performed poorly compared to the numerical K values.

The second machine learning algorithm utilized is SVM which also uses distance function in order to select the maximum margin hyperplane and divides the data into two different homogenous groups. Traditionally, SVM has been used for binary classification but recently gained high popularity because of its high performance in various applications (Lantz, B., 2013). For this paper, I applied the Guassian RBF kernel which performed with high accuracy, reducing the need for multiple kernel trial.

Because of the high class imbalance, using only accuracy, false negatives and false positives as a measure of performance can be misleading as a model might perform accurately by just repeating the most common class by chance. My qualitative evaluation of false negative, false positive and accuracy must be backed up by Kappa as it accounts for those classifications occurred only by chance. Both models have performed extremely well in terms of Accuracy and Kappa.

Nonetheless, when verified with 10-fold cross validation, K=9 has overfitted to the training set and performed poorly compared to the SVM. I subsequently validated K=25 with a similar test. There was no substantial difference in the mean kappa of the 10-fold cross validation. In contrast, SVM performed with high average Kappa when tested with 10-folds cross validation. This is a conclusive result that SVM is the best classifier model for the Autism Screening Adult dataset.

Conclusion

In general, both models performed extremely well in terms of accuracy and individual Kappa test. However, it is clear from the 10 folds cross validation that KNN is not the best model for the ASD data set as its predictive performance is poor compared to SVM. In this specific case, SVM performed exceptionally well with 97.5% accuracy, 0 FN, 3 FP and an average Kappa close to one (0.95) when verified with 10 folds cross validation. In contrast, KNN performed with 98.36% accuracy 1 FN, 1 FP and an average 10-folds cross validation kappa close to zero (0.19) implying poor predictive performance on future data sets.

Yet, this paper is not without its limits. Primarily, there is a high class imbalance in the outcome variable which limits the training capacity of the learning model. For this data set, only a third of individuals were classed as ASD.

Another limitation is the actual size of the data was small. Excluding missing values and noises, the models were left with only 608 instances to learn from.

Small data sizes and class imbalance are the main restraints of ASD based researches (Wall, D.P. et.al. 2012, Kosmicki, J.A. 2015, Ecker, C. et.al 2010). Taking these limitations into account, further research should be undertaken with a larger sample size and a more balanced data set for adequately training these models.

References

American Psychiatric Association 2013. *Diagnostic and Statistical Manual of Mental Disorders: DSM-5.* American Psychiatric Association.

Crippa, A., Salvatore, C., Perego, P., Forti, S., Nobile, M., Molteni, M. and Castiglioni, I., 2015. Use of machine learning to identify children with autism and their motor abnormalities. *Journal of autism and developmental disorders*, *45*(7), pp.2146-2156.

Duda, M., Ma, R., Haber, N. and Wall, D.P., 2017. Use of machine learning for behavioral distinction of autism and ADHD. *Translational psychiatry*, *6*(2), p.e732.

Ecker, C., Rocha-Rego, V., Johnston, P., Mourao-Miranda, J., Marquand, A., Daly, E.M., Brammer, M.J., Murphy, C., Murphy, D.G. and MRC AIMS Consortium, 2010. Investigating the predictive value of whole-brain structural MR scans in autism: a pattern classification approach. *Neuroimage*, *49*(1), pp.44-56.

James, G., Witten, D., Hastie, T. and Tibshirani, R. 2014. An Introduction to Statistical Learning (with Applications in R), Springer, 2013

Kosmicki, J.A., Sochat, V., Duda, M. and Wall, D.P., 2015. Searching for a minimal set of behaviors for autism detection through feature selection-based machine learning. *Translational psychiatry*, *5*(2), p.e514.

Lantz, B., 2013. Machine learning with R. Packt Publishing Ltd.

Liu, W., Li, M. and Yi, L., 2016. Identifying children with autism spectrum disorder based on their face processing abnormality: A machine learning framework. *Autism Research*, *9*(8), pp.888-898.

Rogers, S. and Girolami, M., 2016. A first course in machine learning. CRC Press.

Ruzich, E., Allison, C., Smith, P., Watson, P., Auyeung, B., Ring, H. and Baron-Cohen, S., 2015. Measuring autistic traits in the general population: a systematic review of the Autism-Spectrum Quotient (AQ) in a nonclinical population sample of 6,900 typical adult males and females. *Molecular autism*, *6* (1), p.2.

Thabtah, F., 2018. Machine learning in autistic spectrum disorder behavioural research: A review and ways forward. *Informatics for Health and Social Care*, pp.1-20.

Valiant, L., 2013. Probably Approximately Correct: Natures Algorithms for Learning and Prospering in a Complex World. Basic Books (AZ).

Wall, D.P., Kosmicki, J., Deluca, T.F., Harstad, E. and Fusaro, V.A., 2012. Use of machine learning to shorten observation-based screening and diagnosis of autism. *Translational psychiatry*, *2*(4), p.e100.