ADHD Prediction in Children Through Machine Learning Algorithms

Daniela Andrea Ruiz Lopez[1], Harun Pirim[2] and David Grewell[3]

[1] University of the Andes, Bogotá, Colombia
[2] North Dakota State University, Fargo ND, United States
[3] Northern Illinois University, DeKalb IL, United States

Abstract. Attention-deficit/hyperactivity disorder (ADHD) is a neurodevelopmental disorder that affects approximately 5% of children worldwide. It is typically diagnosed based on the presence of inattentive and hyperactive symptoms. Our objective is to identify ADHD from a Machine Learning (ML) perspective, utilizing symptom information and features such as socioeconomic status, social behavior, academic competence, and quality of life. We conducted extensive experiments using the CAP dataset and various machine learning algorithms, including logistic regression, k-nearest neighbors, Support Vector Machines (SVMs), Random Forest, XGBoost, and an Artificial Neural Network (ANN). The ANN model demonstrated the highest accuracy, achieving an AUC metric of 0.99. As a result, we conclude that using ML algorithms to predict ADHD provides a better understanding of the etiological factors associated with the disorder and has the potential to form the basis for a more precise diagnostic approach. The code is available at: GitHub Repository.

Keywords: ADHD, Machine Learning, Logistic Regression, kNN, SVMs, Random Forest, Xgboost and Artificial Neural Network.

1 Introduction

Attention-deficit/hyperactivity disorder (ADHD) is a common neurodevelopmental disorder with behavior of high distractibility and impaired executive functions [1]. The global prevalence is around 5%, but at least a further 5% of children have substantial difficulties that placed them just under the threshold to meet full diagnostic criteria for ADHD [1].

Capturing and analyzing electroencephalogram (EEG) and Magnetic Resonance Imaging (MRI) data to diagnose ADHD have increased [2]. However, it is usually diagnosed by the behavioral symptoms exposed on the Diagnostic Statistical Manual criteria [3]. It considers 18 ADHD symptoms divided on 9

Inattentive and 9 Hyperactive/Impulsive. According to them, ADHD can be diagnosed categorically (ADHD hyperactive/impulsive, ADHD inattentive, ADHD combined or non-ADHD) or dimensional, based on the total symptom count [4].

On the other hand, due to ADHD heterogeneity between individuals, computer-aided methods of diagnosing have emerged [2]. These approaches are based on supervised machine learning (ML) algorithms for classification, which learn how to map an input to its output from experience (example input-output pairs) [5]. Among the most used models stand out the logistic regression [6], k-Nearest-Neighbors (kNN) [7], Random decision forests [8], Support Vector Machine (SVM) [9] and Artificial Neural Network (ANN) [10].

To advance in the ADHD diagnosis through ML techniques, several studies have been developed to collect phenotypic data such as gender, age, IQ scores, diagnostic status, and medication status, others even include medical images such as EEG and MRIs from ADHD patients and non-ADHD subjects. The Children's Attention Project (CAP) [11] is one of the studies focused on the phenotypic data. It studied the mental health, social, academic and quality of life outcomes along 3 years for children between 6 and 8 years old with diagnostically-confirmed ADHD in contrast to non-ADHD controls [11].

According to the above, our aim is to distinguish ADHD and control groups using the relevant features reported in the CAP dataset [11] and the interconnections between them through machine learning perspective to better understanding etiological factors associated with the disorder, increasing accuracy in predicting outcomes.

2 Methodology

2.1 Dataset description

The CAP's dataset recorded symptoms and social, economic and academic information for 146 medication naive children with ADHD and 209 controls through parent interviews and teacher reports [12]. The ground-truth labels for CAP's dataset were obtained through a rigorous diagnostic process. First, any child reported by parents to have previously been diagnosed with ADHD was regarded as a positive screen; if not, there was an initial screening phase using the Conners' 3 ADHD index. Then, a face-to-face structured diagnostic interview was conducted using the NIMH Diagnostic Interview Schedule for Children IV (DISC-IV) to confirm the diagnostic status. Trained staff members with psychology degrees interviewed the parents of the participating children. Based on these results, the children were assigned ground-truth labels of ei-

ther ADHD or control, forming the basis for classifying participants in the study [11].
The ADHD group comprised 61 Predominantly Inattentive (ADHD-I), 15 Predominantly Hyperactive/Impulsive (ADHD-H) and 70 Combined types (ADHD-CT). Children were classified as having an internalizing disorder if they met criteria for separation anxiety disorder, social phobia, generalized anxiety disorder, post-traumatic stress disorder, obsessive compulsive disorder, hypomania or manic episode, and an externalizing disorder if they met criteria for oppositional defiant disorder or conduct disorder [12].

Dataset is compound of 42 factors (columns) and 355 children (rows). Between the factors is the group to which each child belongs (0 if Control, 1 if ADHD); gender (0 if female, 1 if male), age, Socio-Economic Indexes for Areas (SEIFA), the code of the child, the presence of each of the 18 symptoms, the count of all of them, the count of the hyperactive symptoms, the count of the inattentive ones and the presence of externalizing and internalizing disorder. There is also information about the Clinical Evaluation of Language Fundamentals (CELF), the Wide Range Achievement Test (WRAT) for math and reading, the social problems measure and the academic competence in the first and third age (2011 and 2014) of the study for each factor. Moreover, the sleep problems, the irritability and the Quality of Life in terms of emotion, family and time were also considered inside the dataset.

2.2 Dataset preprocessing

For each of the factors mentioned previously, we estimated descriptive data corresponding to the Mean, Typical Error, Median, Mode, Standard Deviation, Sample Variance, Kurtosis, Skewness Coefficient, Range, Minimum, Maximum, Sum, and Count. Some of them are reported in the table 2.

According to the statistics, most of the children interviewed are males, but the proportion between the groups is very similar (68%-32% in ADHD group and 63%-37% in control one). The mean age was 7.3 years old in both groups. Median and mode were also close to this value. SEIFA mean in children with ADHD is very close to that of the control group, but the median and mode are higher for control children. Children with ADHD have 13 symptoms on average (the inattentive symptoms are the most prevalent), while children in the control group just have 1 (in most cases, a hyperactive symptom). The externalizing disorder is more frequent than internalizing in both groups. It is present in 50% of the children with ADHD and 7% of the control children. Although CELF shows the same performance in the groups, WRATs about maths and reading, academic competence, and all the indicators of quality of life have higher statistics for the control group. On the contrary, irritability and social and sleep problems are more frequent in the ADHD group.

Since some of the data was missing, we made a data recovery using the mode for the categorical variables and an interpolation for continuous ones. Moreover, we considered a factor called *Desertion* that takes a value of 1 if a child had no values reported for data collected in 2014 or 0 if it was complete. A higher proportion of children in the control group dropped out of the study but considering that the difference in Desertion between groups is only 6%, it was not regarded as biased.

On the other hand, as feature values belong to different ranges, data normalization was implemented in both training and validation set. This data preprocessing technique is frequently used in machine learning to use the same scale for all the numerical values in the dataset, which avoids distorting the differences in the ranges of values and losing information. Furthermore, many commonly used algorithms require normalization to model the data correctly.

Correlations between factors are shown in figure 1. We noticed that age was highly correlated only to the CELF language baseline and the same exam after 3 years. SEIFA has low correlations with all factors. The symptoms have a high positive correlation between them and the social problems and irritability indicators. On the contrary, symptoms indicate a high negative correlation with the quality of life, such as with most academic indicators (maths, reading, and academic competence). In this way, the presence of ADHD symptoms seems to be related to lower grades, quality of life, and more behavioral problems.

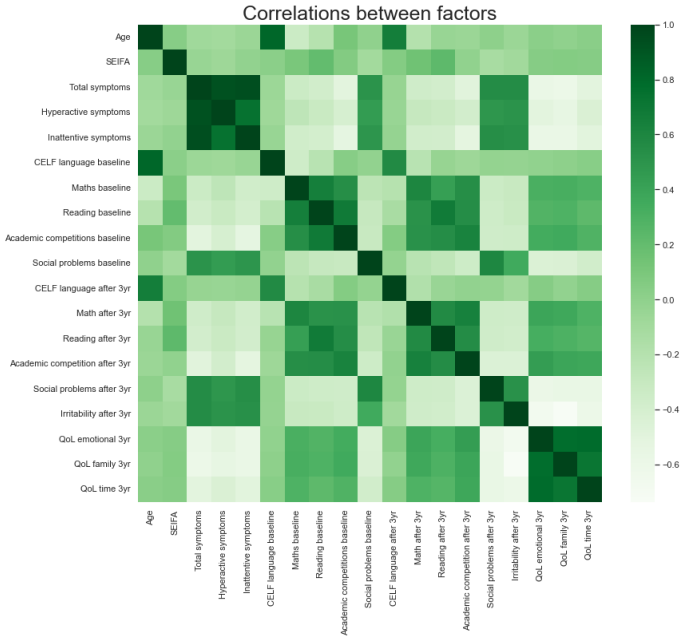

Fig. 1. Pearson correlation between numerical variables. The sign of the correlations indicates the proportionality (direct and inverse), so they are highlighted on a color scale ranging from light colors (highest negative correlations) to dark colors (highest positive correlations).

2.3 Evaluation methodology

We initially divided the data into Train (70%), and test (30%) sets for evaluation. However, to avoid bias induced because of the data split, we used the 5-fold cross-validation method to evaluate. The metrics we worked with were accuracy (2.1), precision (2.2), recall (2.3) and Area Under the Curve (AUC) of the Receiver Operator Characteristic (ROC). The ROC curve plots the Recall (also called True Positive Rate (TPR)) vs. the False Positive Rate (FPR) (2.4).

$$Accuracy=\frac{TP+TN}{TP+FP+FN+TN} \tag{2.1}$$

$$Precision=\frac{TP}{TP+FP} \tag{2.2}$$

$$Recall/TPR=\frac{TP}{TP+FN} \tag{2.3}$$

$$FRP=\frac{TN}{FP+TN} \tag{2.4}$$

True Positives (TP) refer to the correct ADHD predictions, and True Negatives (TN) to the correct control predictions. In contrast, False Positives (FP) refer to the wrong ADHD predictions, and False Negatives (FN) refer to the wrong control predictions. Then, Accuracy is the fraction of correct predictions to total predictions (either positive or negative class) [13]. Precision is the fraction of true positives to total predicted positives (Penalizes false positives) [13]. Recall or TPR is the fraction of true positives to total positives in the data (Penalizes false negatives) [13]. FPR is the fraction of true negatives to total negatives in the data [13]. Finally, AUC measures the ability of a classifier to distinguish between classes [13]. The higher the metrics, the better the model's performance.

2.4 Experiments

Logistic Regression Logistic regression is similar to linear regression but with one critical addition. It analyses the independent variables to estimate a binary dependent variable instead of a continuous one. The logistic function compresses the output of the linear function to a range between 0 and 1 [14]. Then, it is an S-shaped function that gets closer to 1 as the input value increases above 0 and closer to 0 as the input value decreases far below 0 [15].

The hyperparameters tuned were the type of penalty for having too many variables. We tried with *None*, *L1*, and *L2* configurations. We also vary the regularization amount (C) to avoid overfitting between 0.1, 1, and 100. Finally, we changed the class weights from "None" to "balanced" to adjust weights proportionally to class frequencies.

K Nearest Neighbors (kNN) The kNN classifier finds the k-Nearest (k-most similar) instances in the training set to the query point and gets the labels of those training instances. Finally, the algorithm uses the mode of the nearby training labels to predict the label of the new instance [16].

The most important hyperparameter here is the number of neighbors to consider for predicting the class of the query point (K). Likewise, the weight function is crucial; if *uniform* is used, all points in each neighborhood are weighted equally, but if *distance*, points are weighted by the inverse of their distance. We tuned k with values between 1 to 15 for both weight functions.

Linear Support Vector Machine Support Vector Machine is a linear model for classification and regression problems that creates a hyperplane to separate the data into classes. The closest points to the hyperplane from both classes are called support vectors, and the distance between the line and the support vectors is the margin. The goal is to maximize the margin to find the optimal hyperplane [9].

For this algorithm, we change the penalty (*L1* or *L2*) and the loss function (*Hinge* or *Squared hinge*) to measure how good are the train points fitted to the model. We also tried different values of the regularization hyperparameter C (0.1, 1, and 100) and tuned the class weights from unbalanced to balanced.

Kernelized Support Vector Machine Linear support vector machines work well for simple classification problems, where the classes are linearly separable. But in many classification problems, the different classes are located in a way that a line or hyperplane cannot act as an effective classifier. Then, kernelized support vector machines can provide more complex models.

Here, the most important is to specify the kernel type used in the algorithm. The Radial Basis Function kernel (RBF) uses the squared Euclidean distance, the polynomial kernel represents the similarity of vectors in a feature space over polynomials, and the sigmoid kernel uses the sigmoid function. We set C and the class weight the same as in the linear version.

Decision Trees Decision trees are easy to use and understand and are often an excellent exploratory method for getting a better idea of the influential features in a dataset. In a few words, decision trees learn a series of explicit if-

then rules on feature values that result in a decision that predicts the target value [17].

The hyperparameters we changed are the function to measure the quality of a split (*giny*, *entropy* or *log_loss*), the maximum depth of the tree (integer or None to expand nodes until all leaves are pure), and same as in previous methods, class weight (balanced or not).

Random Forest Random Forest combines several decision trees to improve generalizability. This way, different trees see different portions of the data and combine their results. Some errors are compensated with others, which leads to a prediction that generalizes better.

For this algorithm we set the number of trees in the forest (10, 100, and 1000), the function to measure the quality of a split (*giny*, *entropy* or *log_loss*), and the class weight (balanced or not).

Xgboost Like Random Forest, Xgboost uses an ensemble of multiple trees to create more powerful prediction models for classification. However, this algorithm does not build and combine a forest in parallel; it builds a series of trees. Then, each tree attempts to correct the mistakes of the previous one in the series [18].

In this model, we considered a new hyperparameter called learning rate, which is the step size in updating the weights during training. We tuned this hyperparameter with values of 0.01, 0.1, and 1. Moreover, we varied the maximum depth of the trees (None, 3, and 10) and the number of estimators (2, 10, 100, and 200 trees).

Artificial Neural Network (ANN) The latest algorithm we tried was an Artificial Neural Network, specifically, a multi-layer perceptron (MLP). It is compound of three or more layers of nodes (input, hidden, and output). Each node is a neuron that employs a nonlinear activation function, except for the nodes in the input layer. It uses Backpropagation method for training [19].

The hyperparameters we changed are the activation function for the hidden layer (*relu*, *identity*, *logistic*, *tanh*), the solver for weight optimization (*adam*, *lbfgs* and *sgd*), the learning rate schedule for weight updates (*constant, invscaling, and adaptive*), and the initial learning rate used to control the step-size in updating the weights (0.01, 0.001, 0.0001).

3 Results

We tested different hyperparameter configurations for each model to choose the one with the highest AUC. If two or more experiments had the same value

for this metric, we selected the one with the highest precision. All the models worked better with the original data normalized than with the original data without normalization. Then, table 1 shows the results of the normalized data set.

Table 1. Final metrics and models comparison

Model	Accuracy		Precision		Recall		AUC	
	Train	*Test*	*Train*	*Test*	*Train*	*Test*	*Train*	*Test*
Logistic Regression	0.987	0.983	0.99	0.986	0.978	0.973	1	0.999
kNN	0.994	0.986	0.995	0.986	0.991	0.98	1	0.999
Linear SVM	0.986	0.969	0.993	0.972	0.973	0.952	1	0.997
Kernelized SVM	0.983	0.977	0.962	0.954	0.998	0.993	1	0.999
Decision Trees	1	0.958	1	0.958	1	0.938	1	0.955
Random Forest	1	0.986	1	0.98	1	0.986	1	0.998
XgBoost	0.992	0.966	0.99	0.978	0.99	0.938	0.99	0.999
Neural Network	0.987	0.975	0.976	0.987	0.988	0.979	0.999	0.999

The best configuration for the Logistic Regression model used the L2 penalty and C with a value of 0.1. For kNN, the best k was 3 according to the k sensitivity test, and the k points were weighted uniformly regardless of their distance to the query point. The Linear Support Vector Machine worked better with L1 Penalty, Squared hinge loss function, and C equal to 0.1. Between the Kernelized SVMs, “RBF” showed a better AUC. As for Decision Trees, the criterion chosen was entropy with the maximum depth necessary to expand the nodes until all leaves are pure. In the best experiment of Random Forest, we used the default number of estimators (100) and the “gini” criterion. To XgBoost, the Learning Rate used was 0.01, the number of estimators was 100, and the maximum depth of the trees was 6. Finally, for the Neural Network, we selected “relu” as the activation function, “sgd” as the solver, and 0.001 for the initial learning rate (we decided it to be constant throughout the training).

On the other hand, we analyzed the features’ importance based on their relevance in the decision trees-based methods (Decision Trees, Random Forest, and XgBoost). Figure 3 shows that the most important symptoms are difficulty sustaining attention (“susatt”), not following instructions (“instruct”), not listening (“listen”), and avoiding mental effort (“avoid”) while being quiet (“quiet”) and often talking excessively (“talks”) had the lowest importance. In

general, the inattention category shows higher average importance than the category of hyperactivity. Among the social-economic-academic features, we found the Academic Competence taken at the end of the study is the most relevant. In contrast, sleep problems and language tests were of low importance.

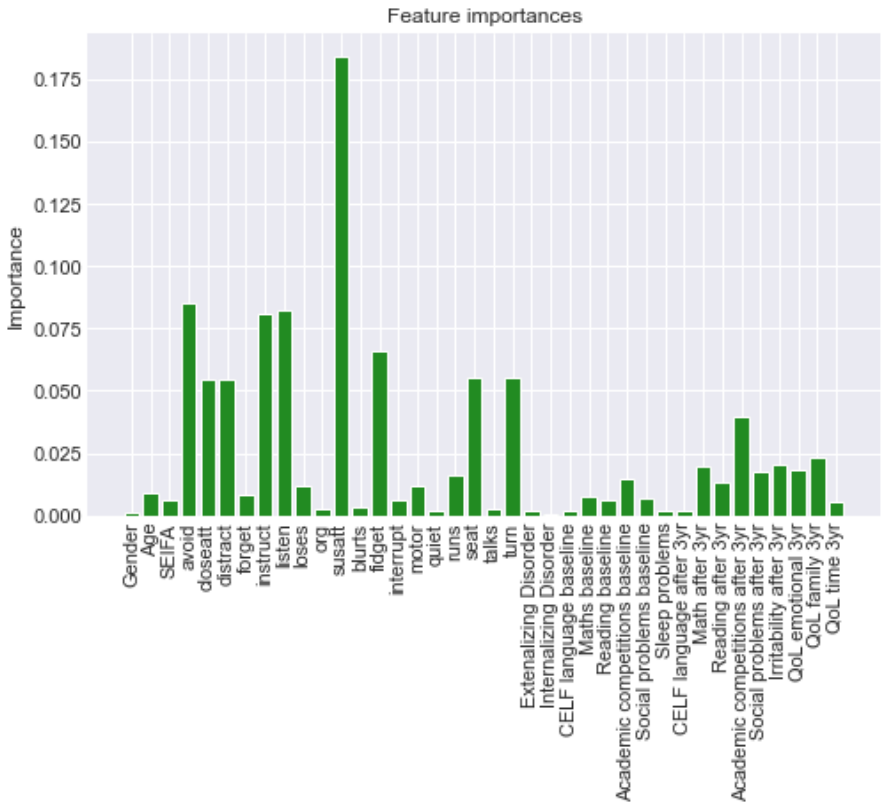

Fig. 2. Features importance in decision trees-based methods.

4 Analysis and conclusions

As mentioned before, the original data showed worse metrics than after normalization. This shows the importance of scaling the data so that each feature has the same contribution. As this procedure can improve the data quality and the performance of the machine learning algorithms, it should be the first step of the data pre-processing.

On the other hand, it is important to mention why the AUC metric was preferred instead of accuracy. The AUC metric uses probabilities of predictions, which is essential to evaluate imbalanced data and make the evaluation more precise. Moreover, we chose Precision over Recall because it penalizes false

positives, the error we want to avoid most because of the consequences of diagnosing a child with ADHD when it is invalid.

The compared models have both advantages and disadvantages. Logistic regression makes no assumptions about distributions of classes in feature space and can interpret model coefficients as indicators of feature importance. However, it assumes linearity between the dependent and independent variables; then, it has a good accuracy just for simple datasets that are linearly separable. kNN is intuitive and straightforward, does not have training steps, and has just one hyperparameter (k). In contrast, as the dataset grows, the algorithm's speed declines; features must have the same scale, and the algorithm is very susceptible to k and outliers. Linear SVM is simple and easy to train, scales well to large datasets, and works well with sparse data. Still, it does not generalize well for lower-dimensional data and assumes that data is linearly separable. Kernelized SVMs are more versatile and work well for low and high-dimensional data, but they need careful normalization of input data and hyperparameter tuning. The Decision Trees algorithm is easy to visualize and interpret, does not need feature normalization, and works well with datasets using a mixture of feature types. However, Decision Trees tend to overfit, and an ensemble of trees is usually needed for better generalization performance. Random Forest solves this problem but requires a higher computational cost and time, just like XgBoost, which is why it is not recommended for problems with high dimensional sparse features. By last, Artificial Neural Networks form the basis of state-of-the-art models. They can be assembled into advanced architectures that effectively capture complex features given enough data and computation. However, more complex models require more training time, data, and customization.

As noticeable in table 1, all the methods report metrics above 0.9. Indeed, the most straightforward methods, such as logistic regression and kNN, had a performance comparable with the multilayer perceptron. These results may be because the dataset is well-behaved and the classes are well-separated, so the differences in the models' capabilities might be less significant. Also, the dataset does not have complex or non-linear patterns, so a simple model like logistic regression could perform as well as a more complex model like MLP. Then, it would be necessary to try the same model with a larger dataset to confirm the proposed model's efficiency and generalization. This new dataset should contain information about worldwide children with and without ADHD to reach a better abstraction. Also, considering more biological data, such as brain connectivity shown in Magnetic Resonance Images (MRIs), would make the classification model more robust.

Finally, it is crucial to analyze the relevance of features to diagnose ADHD according to figure 3. The higher the difference between groups for a feature, the higher its importance. For example, the feature with the highest importance in predicting ADHD is the disability of sustaining attention since

few children in the control group present this symptom. Although the most important features belong to the symptoms category, not all are relevant. Even some symptoms have lower relevance than academic and social factors. In general, the socioeconomic-academic aspects measured 3 years later had higher importance than those measured at the beginning of the study. This behavior could indicate that the differences between both groups increase with the years.

Current ADHD diagnosis assumes every symptom has the same importance, then the combination of them is irrelevant, and the diagnosis can be made based only on the count of symptoms. However, mentioned above contradicts that hypothesis and shows the different relevance values of each feature. It also demonstrates that some social-economic and academic information correlates highly with ADHD, which is helpful in diagnosis.

In conclusion, it is possible to predict ADHD through Machine Learning techniques. Furthermore, ML allows a better understanding of the etiological factors associated with the disorder, which increases the precision of diagnosis. We demonstrated that symptoms do not have the same importance and are not the only relevant features to recognize the presence of ADHD. In this way, predicting ADHD with ML could allow us to identify that 5% of children worldwide are just under the threshold to be diagnosed because they do not fulfill the count of symptoms required, but they show the most important ones.

References

1. K. Sayal, V. Prasad, D. Daley, T. Ford, and D. Coghill, “Adhd in children and young people: prevalence, care pathways, and service provision,” The Lancet Psychiatry, vol. 5, no. 2, pp. 175-186, 2018.
2. N. Sethu and R. Vyas, “Overview of machine learning methods in adhd prediction,” in Advances in Bioengineering, pp. 51–71, Springer, 2020.
3. A. P. Association, Diagnostic and Statistical Manual of Mental Disorders: DSM-5-TR. G - Reference, Information and Interdisciplinary Subjects Series, American Psychiatric Association Publishing, 2022.
4. K. L. Mueller and J. B. Tomblin, “Diagnosis of adhd and its behavioral, neurologic and genetic roots,” Topics in language disorders, vol. 32, no. 3, p. 207, 2012.
5. B. Mahesh, “Machine learning algorithms-a review,” International Journal of Science and Research (IJSR).[Internet], vol. 9, pp. 381–386, 2020.
6. R. E. Wright, “Logistic regression.,” 1995.
7. G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, “Knn model-based approach in classification,” in OTM Confederated International Conferences” On the Move to Meaningful Internet Systems”, pp. 986–996, Springer, 2003.
8. L. Breiman, “Random forests,” Machine learning, vol. 45, no. 1, pp. 5–32, 2001.

9. S. Suthaharan, “Support vector machine,” in Machine learning models and algorithms for big data classification, pp. 207–235, Springer, 2016.
10. J. Zupan, “Introduction to artificial neural network (ann) methods: what they are and how to use them,” Acta Chimica Slovenica, vol. 41, pp. 327–327, 1994.
11. E. Sciberras, D. Efron, E. J. Schilpzand, V. Anderson, B. Jongeling, P. Hazell, O. C. Ukoumunne, and J. M. Nicholson, “The children’s attention project: a community-based longitudinal study of children with adhd and non-adhd controls,” BMC psychiatry, vol. 13, no. 1, pp. 1–11, 2013.
12. T. J. Silk, C. B. Malpas, R. Beare, D. Efron, V. Anderson, P. Hazell, B. Jongeling, J. M. Nicholson, and E. Sciberras, “A network analysis approach to adhd symptoms: More than the sum of its parts,” PLoS One, vol. 14, no. 1, p. e0211053, 2019.
13. Heeswijk, “Precision and recall — a comprehensive guide with practical examples,” Jan 2022.
14. L. Valencia, “[algorithms] — logistic regression - towards dev,” Sep 2021.
15. S. Sharma, “Activation Functions in Neural Networks - Towards Data Science,” Medium, Sep. 06, 2017. https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6.
16. O. Harrison, “Machine learning basics with the k-nearest neighbors’ algorithm,” Sep 2018.
17. Trees, “module2 - decision trees - python,” 2022.
18. T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” in Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pp. 785–794, 2016.
19. M. Fuchs, “multilayer perceptron classifier,” 2021.

5 Annexes

Table 2. Descriptive data: Mean, median, mode and standard deviation for each factor.

	Mean		**Median**		**Mode**		**Standard Deviation**	
Factors	*ADHD*	*Control*	*ADHD*	*Control*	*ADHD*	*Control*	*ADHD*	*Control*
Gender	0.68	0.63	1.00	1.00	1.00	1.00	0.47	0.48
Age	7.27	7.33	7.25	7.34	7.44	7.59	0.44	0.38
SEIFA	1014.9	1015.6	1012.0	1022.0	997.0	1067.0	41.10	45.61
Total symptoms	12.65	1.84	13.00	1.00	12.00	0.00	2.97	2.26
Hyperactive symptoms	5.62	0.99	6.00	0.00	6.00	0.00	2.42	1.39
Inattentive symptoms	7.03	0.85	7.00	0.00	8.00	0.00	1.58	1.33
Externalizing disorder	0.51	0.08	1.00	0.00	1.00	0.00	0.50	0.27
Internalizing disorder	0.25	0.05	0.00	0.00	0.00	0.00	0.44	0.21
CELF language baseline	15.49	15.64	16.00	16.00	16.00	16.00	1.20	1.12
Math baseline	91.30	102.83	92.00	101.00	96.00	92.00	14.33	13.48
Reading baseline	98.10	111.93	97.50	111.00	105.0	108.00	17.34	13.59
Academic competitions baseline	86.17	103.56	87.00	105.00	64.00	107.00	14.18	11.87
Social problems baseline	2.99	1.09	3.00	1.00	2.00	0.00	2.17	1.34
Sleep problems	0.24	0.07	0.00	0.00	0.00	0.00	0.43	0.25
CELF language after 3yr	17.97	18.01	18.00	18.00	18.00	18.00	0.60	0.55
Math after 3yr	87.66	99.09	88.00	97.00	88.00	102.00	14.54	13.16
Reading after 3yr	95.81	106.41	94.00	106.00	89.00	106.00	14.36	13.52
Academic competitions after 3yr	88.87	103.54	91.00	104.00	91.00	121.00	13.84	12.26
Social problems 3yr	2.94	0.99	3.00	0.00	1.00	0.00	2.41	1.48
Irritability after 3yr	4.83	1.61	4.00	1.00	3.00	0.00	3.32	2.32
QoL emotional 3yr	54.40	85.34	62.50	87.50	75.00	100.00	30.01	18.24
QoL family 3yr	66.70	90.20	68.33	95.83	100.0	100.00	23.87	15.48
QoL time 3yr	74.14	96.86	83.33	100.00	100.0	100.00	29.61	10.76
Desertion	0.18	0.24	0.00	0.00	0.00	0.00	0.39	0.43