A Data-Driven Approach for Detecting Autism Spectrum Disorders

Manika Kapoor
Computer Engineering
San José State University
manika.kapoor@sjsu.edu

David C. Anastasiu[*]
Computer Engineering
San José State University
david.anastasiu@sjsu.edu

Abstract— **Autism spectrum disorders (ASDs) are a group of conditions characterized by impairments in reciprocal social interaction and by the presence of restricted and repetitive behaviors. Current ASD detection mechanisms are either subjective (survey-based) or focus only on responses to a single stimulus. In this project, we develop machine learning methods for predicting ASD and characterizing the type of stimuli needed for its detection, based on electrocardiogram (ECG) and skin conductance (SC) data collected during a sensory challenge protocol (SCP) in which the reactions to eight stimuli were observed from 25 children with ASD and 25 typically developing children between 5 and 12 years of age. Each protocol took 45–90 minutes, resulting in a long time series containing approximately 2 million data points for each subject. The length of the time series makes it difficult and costly to use traditional machine learning algorithms to analyze them due to the time and space constraints of these methods. As a result, we developed feature processing techniques which allow efficient analysis of these types of data. The results of our analysis of the protocol time series confirmed our hypothesis that autistic children are greatly affected by certain sensory stimulation. Moreover, we analyzed the degree with which each stimulus affects autistic children and devised a ensemble prediction strategy that combines outcomes of individual stimuli for the task of ASD prediction. Our ensemble model achieved 93.33% accuracy, which is 13.33% higher than the best of 8 different baseline models we tested. The results show that the feature processing and ensemble techniques we developed are effective tools for analyzing longitudinal ECG and SC time series and can be successfully used to detect ASD in children.**

Keywords: Autism Spectrum Disorders, large time series, sensor data driven autism prediction, feature extraction from time series, sensory challenge protocol

I. Introduction

Autism spectrum disorders (ASD) are a group of conditions characterized by impairments in reciprocal social interaction and communication and the presence of restricted and repetitive behaviors. These neurodevelopmental disorders do not have a cure, but their early detection increases the chances of patients being able to develop coping mechanisms that improve their ability to function in society [1]. Current ASD detection mechanisms are focused on the observation of a subject's social interaction. The instruments used for such assessments are lengthy and require extensive training, which prevents them from being used on the overall population. Before referring the subjects for further evaluation, they are first identified as at-risk via a screening process which is sometimes not accurate [2]. The social responsiveness scale (SRS) test, the most popular of such screening instruments, was shown to only have 0.78 sensitivity and 0.67 specificity [3]. Recent work has identified autonomic and behavioral responses of children with autism to be different from those of typically developing (TD) children in response to auditory [4], [5], [6] or visual stimuli [7], [8]. Activity in the parasympathetic nervous system (PsNS), which is measured by high frequency heart rate variability changes, has also been shown to be a good indicator of the presence of sensory modulation dysfunction (SMD) in children [9], which can lead to similar behavioral markers as in autism.

Our research project utilizes longitudinal physiological data collected from multiple sensors in response to a protocol involving eight stimuli sequentially administered to a mixed group of ASD and TD children. Electrocardiogram (ECG) activity was collected at a frequency of 500Hz by placing sensors on the child's chest. ECG provides an index of PsNS function. Galvanic skin response was measured at a frequency of 40Hz by monitoring skin conductance (SC) using sensors attached to the right hand of the subject. SC provides an index of activity in the sympathetic nervous system (SNS). The ECG and SC data were recorded using a PsychLab machine, which encoded each raw signal as a sequence of 500 integers for every second of the protocol. Each protocol took approximately one hour to execute and resulted in large amounts of time series data consisting of millions of correlated values across the length of the protocol.

We need to consider the time component when analyzing the sensor data resulting from the protocol, as it may provide some discriminatory information. For instance, an autistic subject may be affected by one stimulus and its residual effect may be present during the administration of the next stimulus. These correlations can be taken into account by analyzing the data as a time series. Analyzing such large time series is a challenging task, both in terms of the time and the space requirements of the time series analysis methods. In our research, we use different data preprocessing techniques to transform the time series into a form which can be used for efficient analysis and prediction.

We hypothesized that autistic children would be greatly affected by certain sensory stimulation. While TD children can quickly recover to a normal state after the sensory trial, autistic children may be slower to return to normal. We conducted experiments to test our hypothesis and analyzed

[*] Corresponding Author: David C. Anastasiu

the degree with which each stimulus affects autistic children, in general. We also developed predictive models for autism detection from the ECG and SC response signals recorded during the sensory trials.

II. Literature Review

Current ASD detection mechanisms are based on the observation of a subject's social interaction by either close observers [2] or behavioral therapists [10]. The instruments used for ASD assessment are often lengthy and require extensive training before they can be administered and they are also not very accurate [3].

Some researchers have argued that PsNS activity can be used as an indicator for the presence of autism. Schaaf et al. [9] studied PsNS activity during a sensory challenge protocol (SCP) which included a baseline phase, administration of stimuli in five sensory domains, a recovery phase, and a final prolonged auditory stimulus phase. Laufer and Nemeth [11] used SC to predict user action, based on a neural network model, by collecting SC data while users were playing an arcade game. Some researchers have also utilized machine learning-based approaches to build predictive models for the presence of autism. Changchun et al. [12] designed a therapist-like support vector machine (SVM)-based affective model as part of a computer-based ASD intervention tool for children using physiological responses that predicts autism with an accuracy of 82.9%.

Much of the existing research in the field of time series analysis was relevant for this study. Dynamic time warping (DTW) [13] is a technique used to compare two time-dependent series that automatically accounts for time deformations and different speeds. Muda et al. [14] used DTW to create efficient voice recognition algorithms by doing direct analysis and synthesis of voice signals. Juang [15] used DTW hidden markov models, and linear predictive coding techniques to develop speech recognition models. To optimize DTW, Salvador and Chan introduced FastDTW [16], which is an approximation of DTW with linear time and space complexity and is thus comparatively fast. Hong and Dhupia [17] analyzed vibration signals using FastDTW to create more efficient algorithms that can characterize and localize local gearbox faults in automobiles. Mueen et al. have introduced several variants of DTW, including constrained DTW, multidimensional DTW and asynchronous DTW [18].

Piecewise linear approximation (PLA) is one of the most common ways to process time series. It works by approximating a time series of length l with n straight lines using different algorithms, such as the top-down, bottom-up and sliding window approaches. Keogh at el. [19] developed a sliding window and bottom-up algorithm as a means to derive PLA and perform segmentation of time series.

Some methods represent time series using symbols, or motifs, which are derived by identifying frequently occurring patterns in the time series and replacing each pattern with a symbol. Lonardi et al. introduced an algorithm, called enumeration of motifs (EoM) [20], that uses matrix approximation to efficiently locate repeated patterns in the time series and match them by utilizing the algorithm, devised by Shasha and Wang [21].Lin et al. introduced a more scalable method, called symbolic aggregate approximation (SAX) [22], which discretizes original time series data into strings and defines distance measures on the symbolic string representation. Looking for a way to characterize computer usage evolution, Anastasiu et al. [23] devised an optimal segmentation algorithm that segments users' application-level usage into varying length segments.

III. Dataset

Our research is based on examining existing data from a study conducted by Dr. Megan C. Chang [4]. The data were collected from various sensors during a SCP [9] in which the reactions to multiple stimuli were observed from 25 children with ASD and 25 typically developing (TD) children between 5 and 12 years of age. Each protocol took 45–90 minutes including preparation, and had three phases: baseline, sensory challenge, and recovery. The baseline and recovery periods lasted 3 minutes each and did not include any stimulation. The sensory challenge consisted of six different sensory stimuli with a pseudorandom pause of 12–17 seconds between the stimuli. Each stimulus was administered for 3 seconds and was presented at least 8 times. The following are the six stimuli, listed in the order they were administered:

- auditory – continuous sound tone of 84 decibels
- visual – 20W strobe light at 10Hz
- auditory – interrupted sound siren at 78 decibels
- olfactory – wintergreen oil passed under the nose
- tactile – touch along the jaw bone from the mandibular angle on the right to the mandibular angle on the left with a feather
- vestibular – chair tilted back to a 30 degree angle

Physiological ECG and SC data were continuously collected from multiple sensors in response to the eight stimuli (including the baseline and recovery periods). To obtain an index of PsNS function, ECG activity was collected by placing sensors on the child's chest. To measure the SNS activity, galvanic skin response was measured by attaching sensors to the right hand of the child. The sweat glands secrete more sweat as the subject becomes excited or nervous, which in turn increases the skin conductance. The ECG data and SC data were collected at a frequency of 500Hz and 40Hz, respectively. This resulted in a very long time series consisting of approximately 3 million correlated values across the length of the series. Table I provides a description of the dataset that was collected from the 50 subjects, which we analyze in this thesis.

TABLE I
Dataset Description

# Autistic samples	25
# TD samples	25
Average # data points per subject	2,981,476
Average # data points per stimulus	372,682

Figure 1 shows an example of the ECG and SC data for a subject in two different time spans of 2 and 10 seconds. The left y-axis shows the ECG signal, measured in milli-Volts (mV), and the right y-axis shows SC intensities, measured in micro-Siemens (μSiemens).

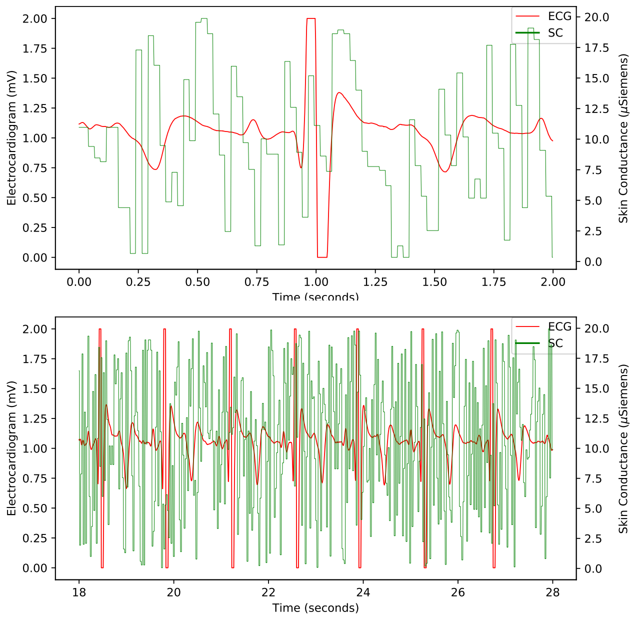


Fig. 1. Time series showing 2 seconds (top) and 10 seconds (bottom) of ECG and SC signal for a subject (best viewed in color).

While Fig. 1 shows ECG and SC values in mV and μSiemens, respectively, the primary data we analyzed is encoded as 16 bit and 24 bit integers, respectively, by the PsychLab machine that was used to record the signal during the SCP. Transforming the primary data into their respective units of measurement is achieved through a simple scaling operation, noting that the machine captures ECG signs in the $0-2$ mV range and SC signals in the $0-20$ μSiemens range. The machine learning models we we describe in the following sections used the primary data as input rather than values in mV or μSiemens.

IV. Hypothesis and Supporting Evidence

We hypothesize that autistic children are greatly affected by certain sensory stimulation and thus may take longer to return to a normal state than TD children, who can quickly recover to a normal state after the sensory trial. To test this, we compared the sensory data recorded during an initial baseline rest stage of the protocol, recorded prior to any stimulus being administered, with data recorded during the final recovery rest stage, 30 seconds after the final stimulus was administered. No stimulus was administered during either rest stage. For each subject, we compared the baseline and recovery rest stages by computing the Euclidean DTW distance of the ECG and SC time series recorded during the rest periods. Euclidean DTW is a measure of distance between two time-dependent sequences which is able to account for different series speeds and lengths. The distance is calculated by doing many-to-one point comparisons between the raw ECG and SC time series data.

To analyze the differences between the baseline/recovery distances of autistic and TD children, we fit a Gaussian probability distribution function (PDF) over the distances between the baseline and recovery sensor time series data for autistic and TD children. Fig. 2 shows these functions for the ECG time series. Results show that autistic (solid green line) children exhibit substantially greater differences between their respective baseline and recovery phases than TD children (dashed red line). The PDF means for autistic and TD children were around 1.25e+9 and 9.07e+8 and their standard deviations were 6.9e+8 and 4.03e+8, respectively. Results suggest that TD children recover faster, which would explain the shorter distances between the respective baseline and recovery phase time series.

While these results support our hypothesis and indicate that stimuli affect autistic children more than TD ones, the remainder of the analysis tries to do two things:

- develop predictive models for autism detection from the ECG and SC response signals recorded during the sensory trials
- analyze the degree with which each stimulus affects autistic children, in general

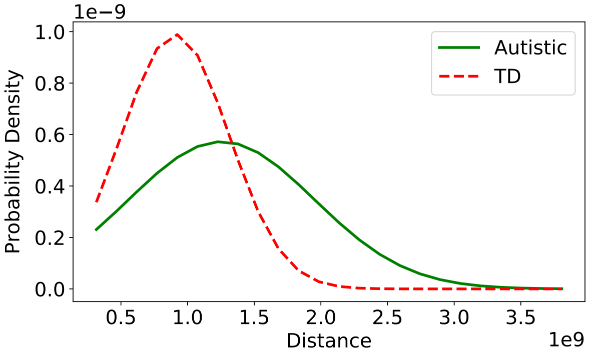


Fig. 2. ECG Gaussian probability density functions of DTW distances between the baseline and recovery stages for autistic and TD subjects.

V. Methods

A list of abbreviations used throughout the rest of the work is provided in the preamble of this thesis. We now present several methods we devised for extracting features from stimulus response time series data, and prediction models we developed for detecting ASD in children. Furthermore, we describe an analysis we conducted for determining the degree with which each stimulus affects autistic children.

A. Feature Extraction

The time series data we are analyzing consist of millions of data points. This poses a major challenge due to the high time and space complexity of existing time series analysis algorithms. As a means to improve analysis efficiency, we propose to transform the data in a form that is representative of the raw time series data but has much smaller dimensionality. We devised three different methods to extract features that can be used to conduct specific experiments. The following sections describe the details of the three methods.

1) Equal Width Partitioning (EWP): As the input to our methods is time series data, we needed to represent them in a standard format that is uniform across all subjects. During the SCP, a particular stimulus is administered in a specific number of contiguous trials at equal intervals. Thus, we can divide the data into sub-series and still capture the pattern or trends in the time series.

In this approach, for each subject, the ECG and SC data were first split into 8 parts representing the 8 stimuli. The data were then standardized using the mean and standard deviation of the baseline stage, i.e., the first of the 8 splits. Since it is recorded prior to any stimulus being administered, the baseline stage captures the normal ECG and SC signal for a subject. After standardization, the data for each stimulus are split into n equal parts. Fig. 3 shows the representation of an SC time series using the EWP approach when n is equal to 7. The time series was divided into 7 equal segments, s1 to s7.

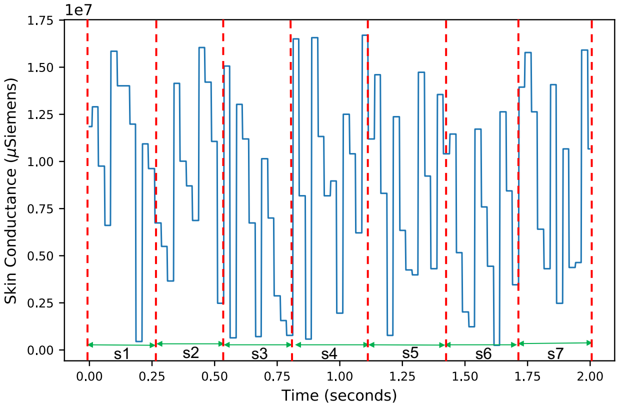


Fig. 3. Example of equal width partitioning of a time series where s1, s2, ..., s7 represent the 7 equal segments.

As each stimulus was administered 8 sequential times during the SCP, with an equal interval between trials, one can split the data into n splits without losing any information. After getting n splits for each stimulus, we used two different approaches to encode the information in each split, as discussed in the following sections.

a) Mean and standard deviation (MSD) representation: In this approach, we represented the n splits for each stimulus using the mean and standard deviation of the data in that split. The final data vector consists of n ECG mean and standard deviation values followed by n SC mean and standard deviation values for each stimulus. For instance, if the value of n is 3, then each stimulus will be split into 3 equal parts and then, for each split, the mean and standard deviation will be computed, leading to 48, i.e., $8 \times 3 \times 2$, values in each of the ECG and SC vectors. This will generate a data vector of $48 + 48 = 96$ values. Thus, the length of the final data vector depends on the number of splits, n, and not on the length of each of the time series. The higher the number of splits, the higher the size of the final data vector will be. We used the vectors obtained using this encoding method to create different machine learning models for ASD prediction in children. For each machine learning method we tested, we created models based on vectors constructed using only ECG data, only SC data, and using both data types.

Fig. 4 shows the ECG mean and standard deviation values for a TD subject (dashed green line) and for an autistic subject (solid red line) chosen at random. One can observe that the ECG mean and standard deviation values of the autistic subject are generally higher than those of the TD subject. The maximum mean value for the autistic subject is 9.52 and that for the TD subject is 5.08.

b) Slope and intercept (SI) representation: We assume that an autistic child gets more excited when a stimulus is administered as compared to a TD child. When a subject gets excited or nervous, his/her ECG values spike, showing higher maximum and minimum values. Also, as the subject becomes excited or nervous, his/her sweat glands secrete more sweat, which in turn increases skin conductance. Thus, we hypothesize that the trend and intensity of the signal contains sensitive information that can be used to predict ASD.

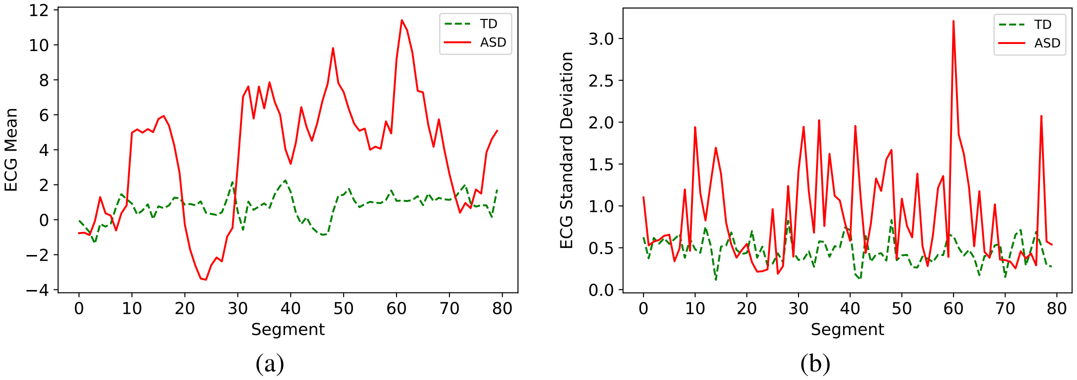
In this approach, we aim to capture the extreme (maximum and minimum) values of the ECG and SC time series and the rate at which they increase or decrease. To do so, we represented ECG data using two different data vectors. For each of the n splits and for each stimulus, we retrieved all peak (maximum) values, denoted as *ekg_p*, and all valleys (minimum) values in a cycle, which we denote by *ekg_v*. A data point is considered a peak value if its value is greater than the value of its neighboring data points. Moreover, a data point is considered a valley if its value is lower than the value of its neighboring data points.

After retrieving all *ekg_p* and *ekg_v* values in a time series, we represented each split as the slope and intercept of the *best fit line* (BFL) for both *ekg_p* and *ekg_v*. The slope of the BLF captures the variation in trend and the intercept captures the intensity of the signal.

SC values fluctuate less than ECG values do, in general. Therefore, we represented the n splits for each stimulus with the slope and intercept of the BFL over the entire SC time series data in that split.

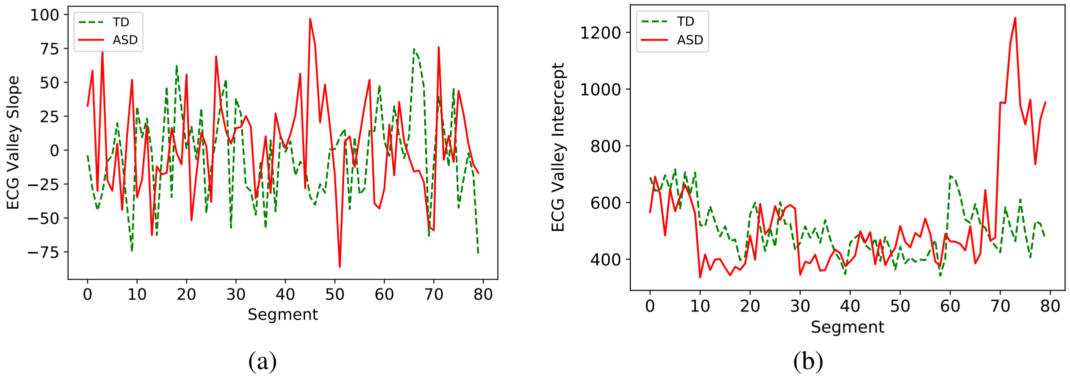
Fig. 5 shows the valley-based slope and intercept representation of the ECG time series, for a TD subject (dashed green line) and for a subject with ASD (solid red line), chosen at random. Time series data represented in these figures were processed using n = 10.

Fig. 6 shows the slope and intercept representation for the SC time series from the same subject as in Figure 5. One can observe that the variation in slopes, especially for ECG valley points and SC data, is higher for the autistic subject as compared to the TD subject. Similar observations can be



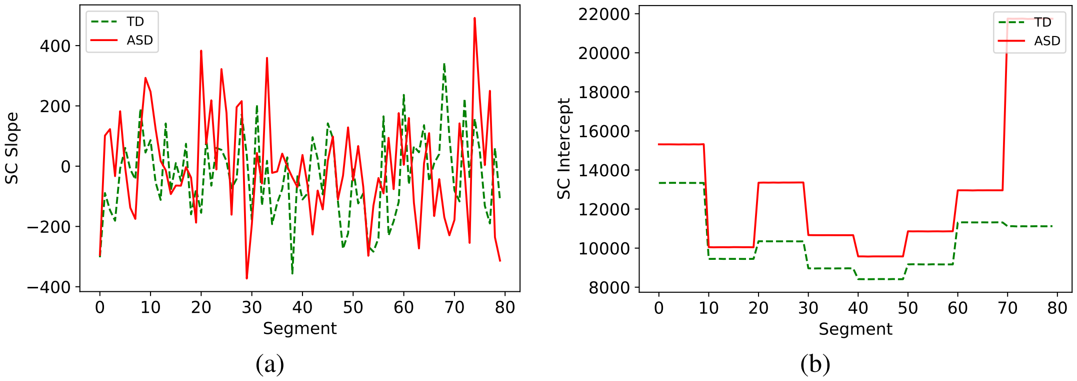
(a) (b)

Fig. 4. Plot showing ECG mean (a) and standard deviation (b) values for a TD subject (dashed green line) and an autistic subject (solid red line), given $n = 10$.



(a) (b)

Fig. 5. Plot showing the valley-based slope (a) and intercept (b) representation of the ECG time series for a TD subject (dashed green line) and an autistic subject (solid red line), given $n = 10$.



(a) (b)

Fig. 6. Plot showing the slope (a) and intercept (b) representation of the SC time series for a TD subject (dashed green line) and an autistic subject (solid red line), given $n = 10$.

seen in other autistic and TD subjects. SC data shows more discriminatory characteristics, with autistic subjects showing higher maximum and minimum slope values. We also plotted the intercept representations for both the ECG and SC time series in order to visualize the intensity of the signal. We observed that the intensity of the signals (ECG and SC) is much higher for autistic subjects as compared to TD subjects.

Similar to models we described in Section V-A.1.a, the length of the considered SI vectors depends on the number of splits, n. Given a specific value of n, we learned machine learning models to predict autism in children using only ECG data, only SC data, and both ECG and SC data encoded as SI vectors. Note that, for the same value of n, the MSD and SI vector representations of time series have the same number of attributes.

2) *Dynamic Time Warping (DTW):* The approach we devised in Section V-A.1 transforms the real time series data into a derived format, which may lead to some loss of information. DTW allows us to compare two time series in their raw format. As DTW automatically accounts for to time deformations, it will identify similar patterns in two time series even if one of them is longer than the other.

In this approach, we used DTW to compare the ECG or SC time series between two subjects. First, we computed the DTW Euclidean distance of every subject to every other subject using both the SC and ECG series. After creating the pairwise distance matrix, we used the k-nearest neighbor (KNN) algorithm for classification. As this method works on the original time series data, which contains more than $100,000$ data points for each stimulus, it is slow and computationally expensive. To increase the speed of the process, we employed a faster version of DTW, called FastDTW, which is an approximation of DTW that has linear time and space complexity [16].

We first conducted an experiment using the basic FastDTW on each stimulus series. However, the experiment could not complete as it ran out of memory on our very large server with 24 GB of random access memory (RAM). The way DTW distance is calculated requires the creation of a 2D matrix of size $l_1 \times l_2$, where l_1 and l_2 are the lengths of the time series being compared. In our case, this matrix would occupy, on average, $100,000 \times 100,000 \times 8$ bytes $= 8$ GB of RAM. Some stimuli have much more than 100,000 data points, leading to out-of-memory errors in executing the algorithm. To address this issue, we split each stimulus into 8 splits, since each stimulus test was repeated 8 times for each subject as part of the protocol. Since the gap between two sequential applications of each stimulus varied between 12 and 19 seconds, as a way to approximate the location of the break between splits, we created eight overlapping splits by including $r\%$ data points from the neighboring splits in each split. For instance, if the second split had 1000 data points, and r was 10%, then the number of extra data points from the neighboring splits would be 0.1×1000. Fig. 7 shows an example representation of the data using overlapping splits.

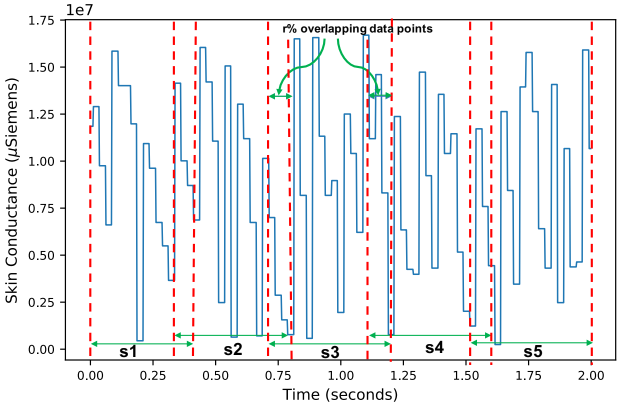


Fig. 7. Example of splitting a time series into 5 overlapping splits.

After creating the splits, we computed pairwise FastDTW distances between all the subjects. For a pair of query and candidate subjects, eight different distances were computed for each stimulus, between each of the 8 stimulus segments in the query subject series and the corresponding segment in the candidate series. We then used the maximum of the eight distance values for the final distance between the two subjects for that stimulus. The pairwise distances were finally used to create a KNN-based machine learning model to predict ASD. We also tried the efficient DTW method introduced by Mueen et al. [18] and compared it with FastDTW. While it was marginally faster than FastDTW, it required more memory and most of our series could not be computed on our server due to lack of available memory.

3) Symbolic Representation of Time Series: In this approach, we used SAX [22] to represent each of the time series using a SAX vector with a given number of symbols and segments. To get the best representation, we tested with numbers of symbols in the range 2 to 10 and numbers of segments from 2 to 14, in increments of 1. After representing the time series using SAX, we computed pairwise Euclidean DTW distances. These distances were then used to create a KNN-based machine learning classification model to predict autism in children.

B. Developing Prediction Models for Autism Detection

1) Base Models: In our experiments, we aim to classify the subject as either Autistic or TD. To perform this binary classification, we trained and tested models based on the following algorithms:

- decision tree (DT) [24]
- k-nearest neighbor (KNN) [25]
- support vector machine (SVM) [26]
- naïve Bayes (NB) [27]
- random forest (RF) [28]
- XGBoost (XGB) [29]
- DTW-based KNN (DTW-KNN)
- SAX-based KNN (S-KNN)

The first six models consume the features generated using methods specified in Section V-A.1. Separate models were created using the MSD and SI feature generation approaches mentioned in Sections V-A.1.a and V-A.1.b, respectively.

We refer to the DTW-KNN and S-KNN models as the standard time series models, as these models use standard time series analysis algorithms. The DTW-KNN model is based on the method described in Section V-A.2, which utilizes the raw time series for comparison and prediction. The S-KNN model is based on the method described in Section V-A.3, which first transforms the raw time series data into its SAX representation before computing pairwise Euclidean DTW distances between the subjects. Different hyper-parameters were tuned for each model to obtain the best results. For measuring the effectiveness of a model, we measured its accuracy in a 10-fold cross-validation scenario. The dataset was randomly split into 10 folds with 5 folds containing 6 samples each and 5 folds containing 4 samples each, such that each fold contained an equal number of randomly selected autistic and TD samples.

As we have both ECG and SC data, we wanted to understand how different physiological data help in predicting autism. Thus, we create different models either using only ECG data, or only SC data, or both ECG and SC data.

2) Ensemble Models: In Section V-B.1, we executed experiments for each separate stimulus. After building the separate models for all stimuli, we combined them to build ensemble models and make additional predictions. We used three different approaches to create ensemble models.

a) Majority vote: In this approach, we combined the predictions from all the models for different stimuli and chose the majority predicted class as the final prediction. All the model outputs were given the same weight. For instance, for a subject T, if 3 out of 8 models predicted T to be autistic, then the final predicted class would be TD. In the case of an equal number of votes for each class, the final predicted class was randomly chosen.

b) Weighted prediction: n this approach, instead of giving the same weight to all the model outputs, we weighed the classification output of each stimulus model with the prediction confidence of its associated model, which ranges between 0 and 1. Considering a vector $\mathbf{w}_c$ of weights associated with each stimulus and the vector $\mathbf{y}$ representing classification predictions of models associated with each stimulus, we compute the final prediction as the linear combination of vectors $\mathbf{w}_c$ and $\mathbf{y}$, $y_c = \mathbf{w}_c{}^T\mathbf{y}$. The vector $\mathbf{y}$ contains the predicted classes, +1 or -1, representing TD and autistic subjects, respectively. A negative y_c prediction value indicates that the models predicted the subject as autistic with higher confidence.

c) Stochastic gradient descent (SGD): In this approach, instead of using the prediction confidence scores from separate stimuli models as weights, as described in Section V-B.2.b, we learned the contribution of each stimulus towards predicting autism. Some stimuli may contribute in a positively towards correct prediction, while others may contribute negatively.

This can be done by deriving a set of weights such that the linear combination of the weight vector and predictions from different stimulus models results in an accurate binary classification of autistic and TD children. The weight vector $\mathbf{w}_s$, is learned via the SGD algorithm applied to training set predictions. Then, the stimuli predictions in the test set are combined linearly with the weights to generate the final SGD predictions for test samples, computed as $y_s = \mathbf{w}_s{}^T\mathbf{y}_s$.

C. Degree of Influence of Each Stimulus on Autistic Children

One of our goals was to find the impact of different stimuli on autistic children and how much each stimulus contributes towards predicting autism in children. This would be useful towards filtering out the less important stimuli in order to simplify the SCP.

To infer the stimulus weights, we first saved all the predictions from all of the individual stimulus prediction models in all 10 folds of our evaluation protocol. We use all of these predictions, along with the true class of all the samples in the entire dataset, to learn inference SGD weights. These weights capture the relationship between the predicted values of each stimulus model and the class for the subject, thus explaining the importance of the stimulus in the prediction. A positive value of the weight for a stimulus shows that it contributes in a positive way towards the correct prediction, and *vice versa*.

VI. Experiment Design

A. Performance Measure

We used *accuracy* as the performance measure when comparing the prediction models. Accuracy is an appropriate evaluation metric in our setting, as the dataset contains an equal number of samples for both autistic and TD subjects. It is defined as

$$A = \frac{T_p}{T_s} \times 100,$$

where T_p represents the total number of correct predictions and T_s represents the total number of subjects.

B. Efficiency Measure

For comparing the efficiency of different methods, we measure the training and prediction time, in seconds, for each of the different models. Prediction time is given priority over training time, as training can be done offline but prediction must be executed online, in real time, and thus needs to be fast.

C. Execution Environment

We conducted experiments on a computing system equipped with an Intel(R) Xeon(R) E5-2695 v3 CPU with 8 cores, each at 2.30 GHz, and equipped with 24 GB RAM. The prediction models were implemented in Python using the *scikit-learn* and *scipy* packages. The DTW-based experiments and SAX-based experiments were conducted using the *fastdtw* and *tslearn* Python packages, respectively.

D. General Data Preprocessing

The dataset consists of the ECG and SC data for each subject, along with protocol event data that can be used to accurately split the time series based on stimulus. After obtaining a different time series for each stimulus, we cleaned the data by removing spurious values, such as ECG data with values of 0.

E. Model Hyper-Parameters

Different hyper-parameters were tuned for each model, details for which are specified below:

- Decision tree (DT). We tested different choices for the *criterion* parameter, namely *gini* and *entropy*, and for the *splitter* parameter, namely *best* and *random*. We also tested values of *min_samples_split* and *max_depth* in the range 1 to 5 and 2 to 6, respectively, to determine the best performance.
- K-nearest neighbors (KNN). For the KNN model, the only hyper-parameter tuned was *k*, which represents the number of most similar samples considered to classify a query sample. We tested values of *k* between 1 and 14 inclusive, in increments of 1.
- Support vector machine (SVM). To get the best results, we tested SVM with all *kernel* types, namely *linear*, *poly*, *rbf* and *sigmoid*, and *C* values varying from 1 to

1000, in steps of 10. Apart from these, we also tested *degree* values in the range 2 to 6 for the *poly* kernel and *gamma* values of 1e−3 and 1e−4 for the *rbf* kernel.

- Random forest (RF). As RF is an ensemble model using DT, most of its parameters are the same as for DT. We tested different values for the *criterion*, *splitter*, *max_depth* and *min_samples_split* parameters, as mentioned earlier. Apart from these parameters, we also tuned the *n_estimators* parameter, testing values in the range 1 to 11, in increments of 1.
- XGBoost (XGB). We tested different *booster* choices, including *gbtree* and *gblinear*, along with different *eval_metric* choices, such as *logloss* and *error*. Apart from these parameters, we also tuned the *min_child_weight* and *max_depth* parameters using values in the range 1 to 5 and 3 to 10, respectively, in increments of 1.
- DTW-based KNN (DTW-KNN). As any other KNN model, the only hyper-parameter tuned was *k*, which represents the number of most similar samples considered to classify a query sample. We tested values of *k* between 1 and 30, inclusive, in increments of 1.
- SAX-based KNN model (S-KNN). We tested values for the numbers of symbols parameter in the range 2 to 10 and for the numbers of segments parameter from 2 to 14 when transforming the time series into its SAX representation. Then, we tuned the KNN hyper-parameter *k*, testing values between 1 and 30, inclusive, in increments of 1.

VII. Results And Discussion

A. Effectiveness Results

1) Base Models: We created eight different models, as described in Section V-B, one for each of the eight stimuli. The first six models, namely, DT, KNN, SVM, NB, RF and XGB, were built using the features extracted based on the two approaches mentioned in the EWP method described in Section V-A.1, which splits the time series into a specified number of sections. We created different dataset representations with number of splits, *n*, ranging from 2 to 13, inclusive. For each value of *n*, after further splitting the training set into training and validation subsets, we trained different instances of all the six models using different combinations of hyper-parameters. Then, we chose the best model instance (i.e., specific hyper-parameter values as described in Section VI-E) based on its validation accuracy. Finally, we re-trained the best model for each algorithm using the chosen best hyper-parameters and the entire original training set.

The DTW-KNN model utilizes the features extracted using the DTW approach mentioned in Section V-A.2, which computes the Euclidean DTW distance between different subjects. Higher distance values imply lower similarity, and *vice versa*. For creating the overlapping splits, we chose r = 10%. The S-KNN model was then built using the SAX feature construction method described in Section V-A.3.

Fig. 8 shows the comparison of the best performing model instances for different algorithms, created using different feature extraction methods and using baseline stage data. We observed that, in almost all cases, the models created using SI features perform better than those created using MSD features. Also, among the two standard time series approaches, the models created using SAX features (S-KNN) perform much better as compared to those based on DTW distances (DTW-KNN).

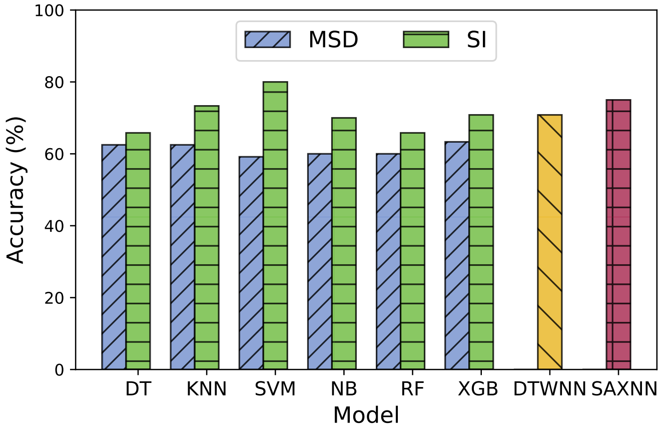


Fig. 8. Comparison of the best base models for the auditory (tones) stage. The description of the methods being compared can be found in Section V-B.1.

Table II shows the accuracy scores of the best models for each stimulus. Auditory (tones) and visual stimuli data result in the best performing models, with an accuracy of 80.00% (highlighted in bold). We also observed that two of the best performing models utilize both ECG and SC data for making predictions, showing that both types of sensor data are important in predicting autism.

TABLE II
Best Base Model Accuracy Values Using Each Stimulus

	Accuracy(%)	Model	Data Used
Baseline	75.83	SAXNN	SC
Auditory (Tones)	**80.00**	SVM	Both
Visual	**80.00**	XGB	SC
Auditory (Siren)	77.50	RF	ECG
Olfactory	77.50	SAXNN	SC
Tactile	74.17	SAXNN	SC
Vestibular	78.33	RF	Both
Recovery	73.33	SAXNN	Both

2) Ensemble Models: We combined the results from the models generated using different stimuli, presented in Section VII-A.1, to create ensemble models. We compared the accuracy of the ensemble models with the best base models. Ensemble models were created using the three approaches described in Section V-B.2.

Fig. 9 shows the comparison of the best SGD ensemble models. We observed that models constructed from SI features outperformed those using MSD ones in almost all cases. The best performing model using SI features is an SGD ensemble XGB model that achieved in an accuracy of 93.33%, which is 7.50% higher than the best performing model using MSD features.

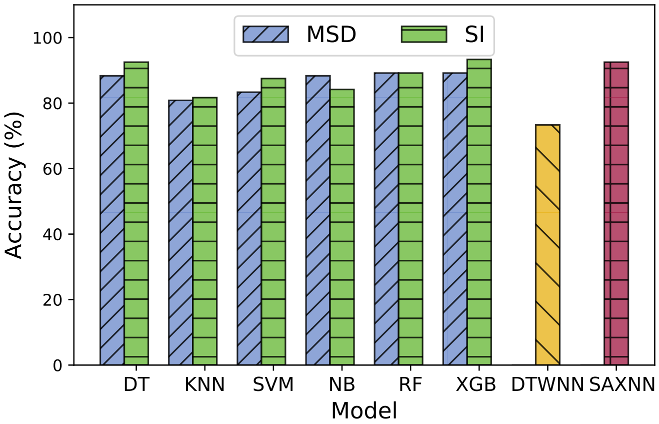


Fig. 9. Comparison of the best SGD ensemble models. The description of the methods being compared can be found in Section V-B.1.

As SI features performed better than the MSD ones, further comparisons with DTW and SAX-based approaches were done using only SI features. As mentioned in Sections V-A.2 and V-A.3, both DTW and SAX-based models are KNN models. Table III shows the best model accuracies for the different tested data processing and modeling methods. One can observe that all the models give the best accuracy using the SGD ensemble method. In this ensemble approach, as described in Section V-B.2.c, the SGD algorithm is applied on the training set to learn the weights of each stimulus towards making correct predictions. Different models had different weights for each stimulus.

TABLE III
BEST ENSEMBLE MODEL ACCURACY VALUES

	Accuracy(%)	Ensemble Type	Data Used
DT	92.50	SGD	Both
KNN	81.67	SGD	SC
SVM	87.50	SGD	Both
NB	88.33	SGD	SC
RF	89.17	SGD	Both
XGB	**93.33**	**SGD**	**Both**
DTWNN	77.50	SGD	Both
SAXNN	92.50	SGD	ECG

The best performing model was the SGD ensemble XGB model, built using both ECG and SC data, which resulted in an accuracy of 93.33%. The value is approximately 4.16% greater than that achieved using either the majority vote or weighted prediction vote ensemble methods. This model was built using both ECG and SC data.

As the best accuracy is achieved using both ECG and SC data, we can infer that both types of sensors are important in accurately predicting autism. Additionally, we observed that the next best performing models were DT and S-KNN, which were built using either only ECG data or both ECG and SC data This further highlights the importance of ECG data in predicting autism in children. In comparison to the best performing base model, the ensemble models performed much better in general. The best performing ensemble model (93.33%) had an accuracy that was approximately 13.33% higher than the best performing base model (80.00%). Even ensemble models built using majority vote (89.17%) and weighted prediction (89.17%) decisions performed better than the base models.

Even though DTW is an important metric for comparing time series, we observed that classification models based on DTW failed to outperform other classification models in our problem. The best accuracy achieved by the DTW-KNN models was 77.50%, which is approximately 18% lower than that of the best performing model.

B. *Efficiency Results*

We measured the efficiency of the models based on the time taken to train and perform predictions. Fig. 10 shows the comparison of natural log transformed training and prediction times, in seconds. The log scaling in the figure is necessary due to the very wide range of values, which would otherwise hide most results in the graph.

The best performing model in terms of accuracy was the XGB model, which was the third slowest method, taking approximately 49,300 seconds to train and 1.23e−4 seconds to predict. On the other hand, the DTW-based model took approximately 4.40 times longer to train and 10^8 times longer to predict in comparison to the S-KNN model. The high execution time for training and prediction makes it difficult to utilize DTW-based models in real-world applications. On the other hand, the DT model achieved the second highest accuracy (92.50%) and predicts 7 times faster than the best performing XGB model.

C. *Inference Results*

Before studying the effect of each stimulus on autistic children, we also studied the results achieved without using any stimuli. This refers to the performance of the models built on the data collected during the *baseline* stage of the SCP. Our assumption was that sensor data generated as a result of stimulus application help in highlighting the difference between autistic and TD subjects. To verify this assumption, we compared the accuracy of the models built using only the *baseline* stage data with that of the ensemble models built in Section VII-A.2, the results of which are shown in Table IV. We observed that, in all cases, the accuracy using the baseline stage models is much lower than that of the models using a combination of stimuli, which supports our assumption. The best accuracy achieved by the best ensemble model is 93.33%, which is 17.5% higher than the one achieved using only data from the baseline stage (75.83%).

To study the effect of each stimulus, we learned the weights of an SGD model using the entire dataset, as discussed in Section V-C. We performed this experiment using the classification method that achieved the highest accuracy, i.e., XGB. We observed that the model resulted in different weights for each stimulus. These weights provide

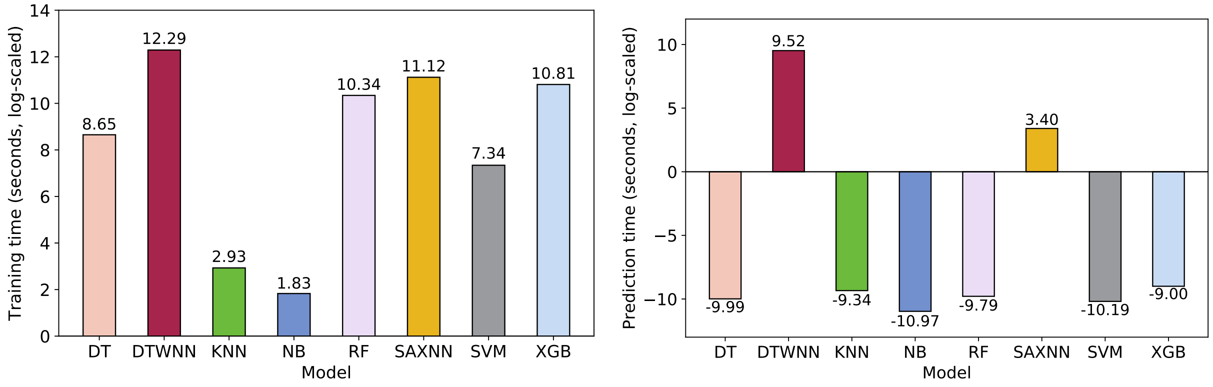


Fig. 10. Comparison of training time (left) and prediction time (right) for all methods.

TABLE IV
ACCURACY COMPARISON OF THE BASELINE STAGE MODELS AND THE BEST ENSEMBLE MODELS

	Baseline Stage	Ensemble Model
DT	72.50	92.50
KNN	70.83	81.67
SVM	69.16	87.50
NB	72.50	88.33
RF	72.50	89.17
XGB	74.17	**93.33**
DTW-KNN	63.33	77.50
S-KNN	**75.83**	92.50

an idea of how important a stimulus is to predict autism. The higher the weight of a stimulus, the higher its contribution is towards accurately predicting autism in children.

Fig. 11 shows the weights learned by the best performing XGB model. One can observe that the XGB model gives very high weights for the *auditory (tones), visual, olfactory, tactile*, and *recovery* stages.

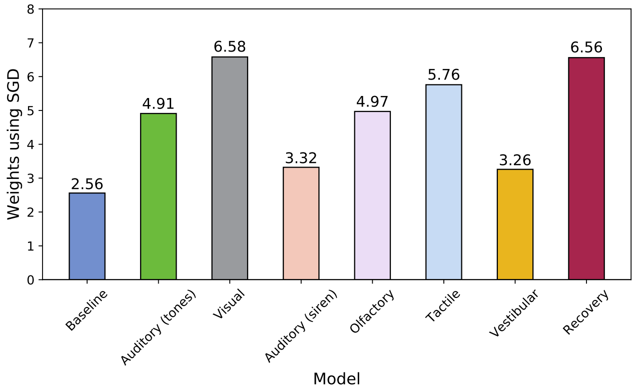


Fig. 11. Stimuli weights for the best performing XGB model.

VIII. Future Work

In the future, we plan to work on developing additional time series-based analysis techniques for predicting autism in children. Similar to models developed by Anastasiu et al. for characterizing computer usage evolution [30], our models will characterize the SNS and PsNS changes over the time of the SCP. In these time series models, an unsupervised optimization procedure will be used to automatically identify prototypical SNS and PsNS states (protos). This procedure seeks to minimize the approximation error of representing the original time series as a sequence of protos. Then, nearest neighbor classification models can be built using this proto sequence representation, which may lead to a greater prediction accuracy by ignoring noise in the data and may provide invaluable insights into common states for ASD and TD children.

IX. Conclusions

In this thesis, we described novel techniques we developed for analyzing very large time series of ECG and SC sensor data derived from a sensory trial administered to 50 autistic and TD children. Our analysis showed that autistic children are affected to a higher degree by some stimuli as compared to TD children and take longer to recover. Moreover, the feature extraction methods we developed were both effective and efficient in analyzing multivariate time series with over 2 million values. A KNN model built using SAX features we extracted from both SC and ECG time series performed quite well when classifying subjects as autistic or TD, achieving an accuracy of 93.33%. We also observed that some stimuli are more significant than others in predicting autism in children. Inference of an ensemble model based on the best performing classifier in our experiments showed increased reliance on the *auditory (tones), visual, olfactory, tactile*, and *recovery* stimuli time series.

An XGB-based model trained on vectors constructed using a feature engineering method we developed (SI) achieved the best performance (93.33% accuracy) taking only a millisecond to predict samples. While DTW is one of the best approaches to compare time series data in general, it does not perform well when working with very large time series data as the ones in our experiments. Models built using DTW were computationally very expensive, taking 4.4 times longer to train and 10^8 times longer to predict as compared to the best model in our experiments.

REFERENCES

[1] "The importance of early detection." [Online]. Available: https://www.parents.com/health/autism/symptoms/importance-of-early-detection-autism/

[2] M. Norris and L. Lecavalier, "Screening accuracy of level 2 autism spectrum disorder rating scales: A review of selected instruments," *Autism*, vol. 14, no. 4, pp. 263–284, 2010, doi:10.1177/1362361309348071.

[3] J. Constantino and C. Gruber, "Social responsive scale (srs) manual," *Los Angeles, CA: Western Psychological Services*, 2005, doi:10.1177/1534508410380134.

[4] M. C. Chang, L. D. Parham, E. I. Blanche, A. Schell, C.-P. Chou, M. Dawson, and F. Clark, "Autonomic and behavioral responses of children with autism to auditory stimuli," *American Journal of Occupational Therapy*, vol. 66, no. 5, pp. 567–576, 2012, doi:10.5014/ajot.2012.004242.

[5] R. J. Palkovitz and A. R. Wiesenfeld, "Differential autonomic responses of autistic and normal children," *Journal of Autism and Developmental Disorders*, vol. 10, no. 3, pp. 347–360, 1980, doi:10.1007/BF02408294.

[6] L. N. Stiegler and R. Davis, "Understanding sound sensitivity in individuals with autism spectrum disorders," *Focus on Autism and Other Developmental Disabilities*, vol. 25, no. 2, pp. 67–75, 2010, doi:10.1177/1088357610364530.

[7] T. Chaspari, M. Goodwin, O. Wilder-Smith, A. Gulsrud, C. A. Mucchetti, C. Kasari, and S. Narayanan, "A non-homogeneous poisson process model of skin conductance responses integrated with observed regulatory behaviors for autism intervention," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1611–1615, doi:10.1109/ICASSP.2014.6853870.

[8] W. Liu, M. Li, and L. Yi, "Identifying children with autism spectrum disorder based on their face processing abnormality: A machine learning framework," *Autism Research*, vol. 9, no. 8, pp. 888–898, 2016, doi:10.1002/aur.1615.

[9] R. C. Schaaf, T. W. Benevides, E. Blanche, B. A. Brett-Green, J. Burke, E. Cohn, J. Koomar, S. J. Lane, L. J. Miller, T. A. May-Benson *et al.*, "Parasympathetic functions in children with sensory processing disorder," *Frontiers in Integrative Neuroscience*, vol. 4, p. 4, 2010, doi:10.3389/fnint.2010.00004.

[10] S. Chandler, T. Charman, G. Baird, E. Simonoff, T. Loucas, D. Meldrum, M. Scott, and A. Pickles, "Validation of the social communication questionnaire in a population cohort of children with autism spectrum disorders," *Journal of the American Academy of Child & Adolescent Psychiatry*, vol. 46, no. 10, pp. 1324–1332, 2007, doi:10.1097/chi.0b013e31812f7d8d.

[11] L. Laufer and B. Németh, "Predicting user action from skin conductance," in *Proceedings of the 13th international conference on Intelligent user interfaces*. ACM, 2008, pp. 357–360, doi:10.1145/1378773.1378829.

[12] C. Liu, K. Conn, N. Sarkar, and W. Stone, "Physiology-based affect recognition for computer-assisted intervention of children with autism spectrum disorder," *International journal of human-computer studies*, vol. 66, no. 9, pp. 662–677, 2008, doi:10.1016/j.ijhsc.2008.04.003.

[13] M. Müller, "Dynamic time warping," *Information retrieval for music and motion*, pp. 69–84, 2007, doi:10.1007/978-3-540-74048-3_4.

[14] L. Muda, M. Begam, and I. Elamvazuthi, "Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques," *arXiv preprint arXiv:1003.4083*, 2010. [Online]. Available: http://arxiv.org/abs/1003.4083

[15] B.-H. Juang, "On the hidden markov model and dynamic time warping for speech recognitiona unified view," *Bell Labs Technical Journal*, vol. 63, no. 7, pp. 1213–1243, 1984, doi:10.1002/j.1538-7305.1984.tb00034.x.

[16] S. Salvador and P. Chan, "Toward accurate dynamic time warping in linear time and space," *Intelligent Data Analysis*, vol. 11, no. 5, pp. 561–580, 2007. [Online]. Available: https://content.iospress.com/articles/intelligent-data-analysis/ida00303

[17] L. Hong and J. S. Dhupia, "A time domain approach to diagnose gearbox fault based on measured vibration signals," *Journal of Sound and Vibration*, vol. 333, no. 7, pp. 2164–2180, 2014, doi:10.1016/j.jsv.2013.11.033.

[18] A. Mueen and E. Keogh, "Extracting optimal performance from dynamic time warping," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 2129–2130.

[19] E. Keogh, S. Chu, D. Hart, and M. Pazzani, "Segmenting time series: A survey and novel approach," in *Data mining in time series databases*. World Scientific, 2004, pp. 1–21.

[20] J. Lonardi and P. Patel, "Finding motifs in time series," in *Proc. of the 2nd Workshop on Temporal Data Mining*, 2002, pp. 53–68.

[21] D. Shasha and T.-L. Wang, "New techniques for best-match retrieval," *ACM Transactions on Information Systems (TOIS)*, vol. 8, no. 2, pp. 140–158, 1990.

[22] J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing sax: a novel symbolic representation of time series," *Data Mining and knowledge discovery*, vol. 15, no. 2, pp. 107–144, 2007.

[23] D. Anastasiu, A. Rashid, A. Tagarelli, and G. Karypis, "Understanding computer usage evolution," in *2015 IEEE 31st International Conference on Data Engineering, ICDE 2015*, vol. 2015-May. IEEE Computer Society, 5 2015. doi: 10.1109/ICDE.2015.7113424 pp. 1549–1560.

[24] C. Jin, L. De-Lin, and M. Fen-Xiang, "An improved id3 decision tree algorithm," in *Computer Science & Education, 2009. ICCSE'09. 4th International Conference on*. IEEE, 2009, pp. 127–130.

[25] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE transactions on information theory*, vol. 13, no. 1, pp. 21–27, 1967.

[26] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik, "Support vector clustering," *Journal of machine learning research*, vol. 2, no. Dec, pp. 125–137, 2001.

[27] K. P. Murphy *et al.*, "Naive bayes classifiers," *University of British Columbia*, vol. 18, 2006.

[28] T. K. Ho, "Random decision forests," in *Document analysis and recognition, 1995., proceedings of the third international conference on*, vol. 1. IEEE, 1995, pp. 278–282.

[29] 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*. ACM, 2016, pp. 785–794.

[30] D. C. Anastasiu, A. M. Rashid, A. Tagarelli, and G. Karypis, "Understanding computer usage evolution," in *Data Engineering (ICDE), 2015 IEEE 31st International Conference on*. IEEE, 2015, pp. 1549–1560, doi:10.1109/ICDE.2015.7113424.