Anxiety Disorder Prediction from Virtual Reality Head Movements

Sarah Ciresi **John Hewitt** **Cooper Raterink**

Introduction

We are motivated by the opportunity of using machine learning on novel VR-based mental healthcare datasets to improve understanding and diagnosis of mood and emotional disorders. Recently, researchers have established a relationship between degree, frequency, and type of head movement and the degree of clinical depression experienced by an individual [1]. We use machine learning methods to analyze head movement data of subjects undergoing various virtual reality experiences in order to predict whether they are suffering from an anxiety disorder. Andrea Goldstein, from Stanford's Williams PANLab, advised us and provided the data gathered during the ENGAGE study, which previous to this project has been unexplored for this task. The ENGAGE study focuses on the effects of behavioral intervention on participants suffering from obesity and depression [3].

We are given the time-series data gathered from Oculus sensors for 148 ENGAGE study participants. The time-series represent two channels of yaw, pitch and roll values of the participant's head location. This is our input data, and our output labels are binary - true if the participant's GAD7 (a score for level of anxiety) value is above 10, false otherwise. We featurize this data in two ways - (1) summary statistics across time, and (2) a 30-point Discrete Fourier Transform. We feed both of these input featurizations to three different classifiers - Logistic Regression, Naive Bayes, and Decision Tree. So, in total there are six classification nodes. The outputs of these 6 classifiers are fed into an ensembled learned-weights voting node and the output of that is our final prediction. We compare the efficacy of this approach for several weighting/threshold schemes against a convolutional neural network using the unfeaturized time-series as input, and also against predict-1 and predict-random baselines, finding our best model to improve upon baselines.

Related Work

To our knowledge, this is the first time the task of psychological disorder prediction using machine learning has been explored for time-series head movement datasets gathered from virtual reality experiences. That said, we drew inspiration from discussion with members of the PANLab and chose machine learning methods based on their success with similar tasks. We chose to use a convolutional neural network in hope that it captures patterns that hand-crafted features do not. A study done by Hoppe and Bulling proposes a convolutional neural network for learning featurizations for classification tasks on eye movement data [2]. They use a short-time frequency transform on the data beforehand, which we don't use for the CNN because of resolution (sampling time) differences between our data and the eye movement data, however this idea feeds into our second featurization of the head movement data using the Discrete Fourier Transform (DFT). Another study, on detection of seizure using EEG data, uses a DFT featurization before feeding the data to a decision tree [4]. This is not related to head movement data, however it is also a classification procedure on time-series data in which frequencies are important, and is also fed into a decision tree and so supports our intuition that a DFT featurization would be applicable to our problem. Further, as mentioned above, researchers have shown that type of head movement can be related to neurological disorders [1]. This study in particular uses summary statistics of head movements as featurization of time-series yaw, pitch and roll data.

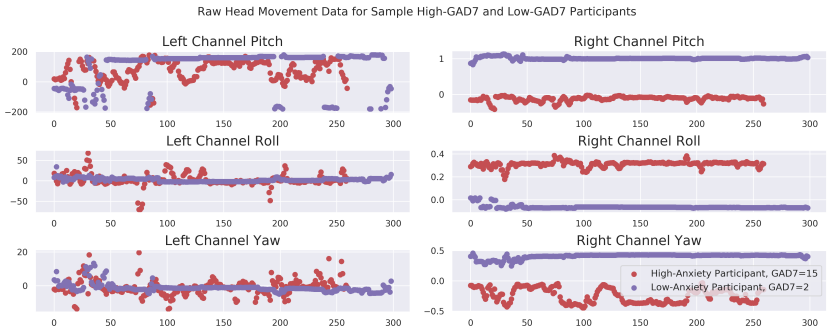

Figure 1: Plot of head movement data across pitch, yaw, and roll for two tracking channels for each of two participants. One participant (red) was recorded to have low anxiety at the time of the survey, while the other participant (purple) was recorded to have high anxiety.

0.1 Dataset and Featurization

The data we have available to us is head angle data recorded over time by two sensors of an Oculus VR. The data is recorded at 0-, 2-, 6- and 12-month sessions and there are five types of virtual reality experiences during each session. An example of the time-series yaw, pitch roll data for a single VR experience can be seen in Figure 1 above. Based on discussions with a team from Williams PANLab, it seems there is no de-facto method for representing head movement time series data for psychological prediction tasks. Additionally, not all of the head movement time series have the same length. So, we decided to use two featurizations computed on the whole time-series: (1) summary statistics and (2) the Discrete Fourier Transform. The convolutional neural network takes in a finite length of the original time-series data - that is, no featurizations are used, we assume the CNN will learn appropriate featurizations.

In our experiments, we attempt to predict the GAD7 score, an indicator of anxiety, based on the feature vector discussed above. Computing these features requires that we have the associated score labels and the tracking data for each experiment type for a given participant and month.

$|V_{GAD7}| = 148$ is the number of pairs of (participant, month) for which we have the required data. Because of the small number of examples, we use a hold-one-out cross-validation scheme and for testing we set aside 30 test examples. This leaves us with a training matrix X_{GAD7} and a label matrix Y_{GAD7} whose rows correspond respectively to the feature and label for all (p, m) leftover for training. In our experiments, X_{GAD7} has 118 observations and 120 summary statistics features or 360 frequency domain features, depending on the featurization scheme used.

0.1.1 Summary Statistics

We compute summary statistics on both the time series and the differences between subsequent elements of the time series as our features.

Each piece of head-tracking data is a matrix T whose columns are the roll, pitch, yaw gathered by sensor 1 concatenated with that of sensor 2. Now we compute a difference matrix:

$$D_{i,j} = |T_{i,j} - T_{i-1,j}|$$

Then for every column of T we compute the mean and variance, and every column of D we compute the sum and variance. We concatenate these statistics across all experience types to form a feature vector for each (participant, month) pair.

0.1.2 Discrete Fourier Transform

We use a 30-pt Discrete Fourier Transform(DFT) computed on each time axis for our second featurization. The N-point DFT is defined as follows:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-2\pi jkn/N}$$

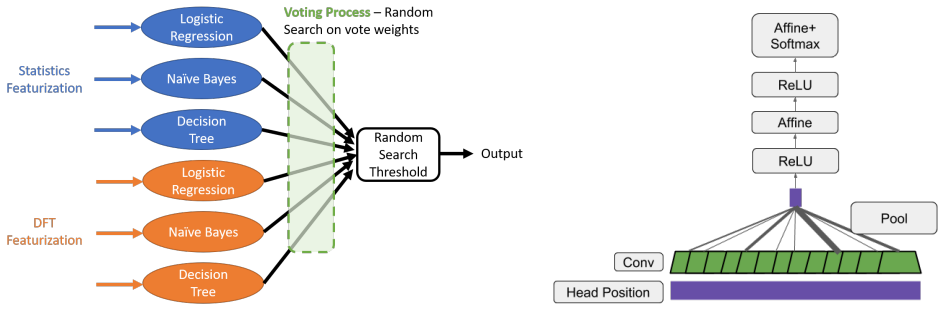

Figure 2: Diagram of the ensemble voting model and convolutional neural network

Where X_k is called the Discrete Fourier Transform of the sequence x_n. X_k can be thought of as periodic with period N or as of length N - hence it is called the N-point DFT and for our purposes is used to compute N features.

There exists an aptly named algorithm called the Fast Fourier Transform(FFT) which is a computationally efficient implementation of an N-point DFT. We use numpy's FFT implementation to featurize our time-series head movement data across each each channel's yaw, pitch and roll time axis.

0.2 Methods

The primary challenge of our task, beyond uncertainty in featurization due to the novelty of the task, is data scarcity. We have only 118 participants on which to train our systems and a complex phenomenon to model. We pursue three broad avenues to tackle these challenges: (1) simple classifiers with different inductive biases and trained off of each of our two featurizations, (2) ensembles of simple classifiers with vote weighting determined through random search, and (3) a small convolutional neural network with pooling across time.

Simple Models We train Multinomial Naive Bayes classifiers, Decision Trees, and logistic regression classifiers on each of our 2 featurization types, for a total of 6 simple classifiers. In short, the naive bayes models each observation as having been generated by sampling the class, and then sampling all features independently given the class; the decision tree iteratively splits the data to minimize the gini loss based on individual parameter differences, and the logisitic regression learns a linear combination of the features to minimize the difference between the true and predicted probabilities of anxiety disorder for each patient. We ran manual search in development, deciding to use a max tree depth of 5, and a regularization precision of 1 for our logistic regression.

Ensemble Weighting through Random Search Because of the high-variance nature of the data and the different inductive biases in each of our featurizations and simple model architectures, we hypothesized that an ensemble of models may improve over the performance of any individual model. We start with a simple majority-vote ensembling baseline. To leverage the intuition that (1) the individual simple models are not of the same quality, and (2) the tradeoff between recall and precision may be controlled through the voting threshold for predicting anxiety, we run a random search on our development set to find high-quality model weights and decision thresholds for precision, recall, and F1.

Specifically, for each of the 6 simple models we draw a value from a gamma distribution with parameters (shape=2, scale=1), and then normalize the weights by the sum of all. We chose this distribution because it should give some variation between model weights, without deviation in extremes. For the threshold, we sampled from a uniform between .4 and .6, as we found that sampling from a greater range led to degenerate results like extremely high recall by predicting all participants to have high anxiety. For each sampled set of hyperparameters, we ran all models 5 times using hold-one-out evaluation and averaged the scores. We then picked the set of hyperparameters for each of F1, precision, and recall that led to the best development result to run on the test set. This model is visualized in Figure 2.

Model	F1	Precision	Recall
All-True baseline	**33.3**	**20.0**	**100.**
Coin flip	28.0	19.9	48.3
LogReg sstat	23.5	18.2	33.3
LogReg dft	33.3	25.0	50.0
MultinomialNB sstat	37.0	23.8	**83.3**
MultinomialNB dft	**43.5**	29.4	**83.3**
DecisionTree sstat	40.1	**32.3**	53.0
DecisionTree dft	28.3	22.0	40.3
Precision-Weighted Ensemble	39.2	**32.4**	50.0
Recall-Weighted Ensemble	37.0	23.8	**83.3**
F1-Weighted Ensemble	**40.5**	26.8	**83.3**
Equal-Weighted Ensemble	35.9	28.1	50.0
CNN	21.6	17.0	18.6

Table 1: High-anxiety patient classification results on 30 experiences from the held-out patient test set. Best model in each category (baselines, simple classifiers, ensembles+CNN) is bolded.

1-dimensional Convolutional Neural Network For our second model, we consider a small 1-dimensional convolutional neural network that uses the unfeaturized raw head movement data across the 6 channels of roll, pitch, yaw, for both the right and left sensors. We felt that a 1D CNN model was well-suited for our data given that we were working with time series data where the exact time of head movement may not be as important as the amount or speed of movement in short intervals for predicting anxiety levels. Our CNN architecture consists of five layers. The first layer is a 1-dimensional convolutional layer, followed by an average pooling layer, a ReLU activation layer, and two dense layers that also have ReLU activations. We included dropout as a form of regularization as well, and chose the dropout rate during our hyperparameter search on the development set. This model's layout is visualized in Figure 2.

0.3 Experiments & Results

Because the split of high-anxiety to low-anxiety participants was roughly .20% to 80%, we report precision, recall, and F1 score across all experiments.

0.3.1 Ensemble Weighting through Random Search

In our random search, we sampled 50 ensemble configurations. A few interesting patterns emerged. The top-recall ensemble simply chose to ignore all models but the Naive Bayes on summary statistic features. The top-precision ensemble, as might be expected, used the highest threshold of the three, at 60% of voting weight required to predict "high anxiety." The top-F1 ensemble assigned almost all its weight equally across the summary statistic models.

0.3.2 1D Convolutional Neural Network Hyperparameter Tuning

A convolutional neural network layer passes the same feature detector across all spatial steps of the data, and in our case uses a pooling function to aggregate features across all timesteps. For our CNN, we found that the set of hyperparameters that resulted in the highest F1 score on the development set was a filter count of 16, a kernel size of 10, and a dropout rate of 0.5, and ultimately used this choice of values for our final model. We report numbers on CNNs trained only the first 400 timesteps of the data as the results were robust to the number of timesteps used. One interesting takeaway from our hyperparameter search was that smaller models tended to do best, reflecting small-data problem, but our smallest models started to degrade as well, perhaps signaling limitations of raw head movement feature format with so little data.

0.3.3 Test Set Results

Our results are summarized in Table 1, averaged over 30 random initializations except the CNN, which was averaged over 10. Our baselines were predicting everyone to have anxiety, and predicting

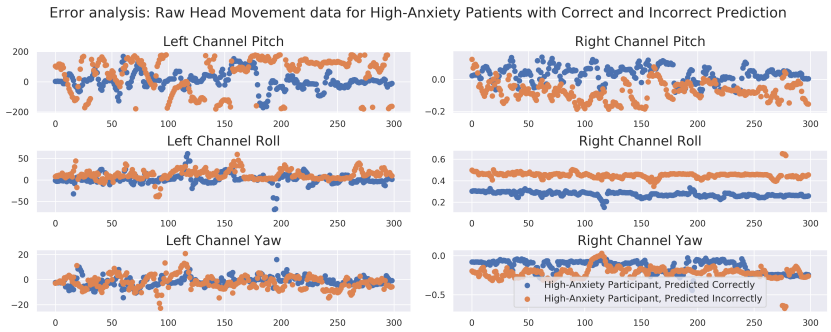

Figure 3: The head movement data of two high-anxiety participants. One participant our best model classified correctly as having high anxiety; the other was misclassified.

a random 50% to have anxiety.[1] Our best model, a multinomial Naive Bayes classifier on Discrete Fourier Transform features, achieves an F1 score of 43.5. This is a 10.2 F1 improvement over the best informationless baseline, showing that there is in fact signal to be had in predicting anxiety disorder through head movement data. Our best precision is achieved by our precision-weighted ensemble, at 32.4, though it is not substantially better than the 32.3 precision of the decision tree on summary statistic information. Our best recall is achieved by Naive Bayes models at 83.3 and some of our ensembles which assigned high weight to them.

Our ensembles underperform our expectations, perhaps due to the high variation in model quality and the bias of our random search to being close to uniform. However, while they do not in general outperform the individual models, we do see the desired behavior of each weighted ensemble (F1, precision, recall) tending to bias towards that metric while not sacrificing too much on the other metrics. Finally, our CNN model underperforms all baselines despite hyperparameter tuning to attempt to avoid overfitting. We expect that this is because the small amount of data, and somewhat abstract problem, make joint feature detection and generalization infeasible.

Finally, we conducted qualitative error exploration on our development set. We sampled two high-anxiety participants, one of which our Naive Bayes classified correctly, and the other incorrectly. Qualitatively, the examples seem quite similar in the amount of movement, as seen in Figure 3, though perhaps our featurization did a poor job of capturing the many discontinuities of the movement of the misclassified high-anxiety participant, since each individual discontinuity contributes little to the macro “variability" of the sequence, but may be highly predictive.

0.4 Discussion & Conclusion

In this work, we explored head movements, a noisy signal in the medical domain which we confirm to be useful for predicting patient anxiety disorder. We faced an inherently small-data problem, since controlled participation in a VR experience is costly to collect. As such, we focused on featurization and model comparison to determine what features and methods are promising for evaluating anxiety through head movement. Our best model improved on an informationless baseline by 10.2 points F1, a modest but potentially useful result when combined with other predictors of anxiety in a hypothetical future system. By ensembling models and running a random search on the ensemble voting weights and decision threshold, we were able to control the tradeoff between precision and recall, but not improve upon the F1 score of individual models, a mixed result. For thoroughness, we compared our featurizations and simple models to a low-parameter CNN, finding as we expected that the CNN underperformed models with hand-crafted features. We hypothesize this was due to the rather small-data setting. Our findings suggest that head movement data has signal for predicting anxiety disorder, and suggest that future work may leverage richer representations of each patient, in combination with head tracking, to improve predictiveness and eventually improve professionals’ ability to care for patients.

[1]We also considered a baseline which predicted a random 20% to have anxiety, which performed worse.

Contributions Cooper Raterink led feature construction, constructing both featurizations used in the final model, as well as data pipeline management and communication with the ENGAGE team. Sarah conducted studies with simple classifiers and wrote and tuned the convolutional neural network. John Hewitt wrote the ensemble classifier and random search procedure, and led experimental design.

References

[1] Sharifa Alghowinem et al. "Head pose and movement analysis as an indicator of depression". In: *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*. IEEE. 2013, pp. 283–288.

[2] Sabrina Hoppe and Andreas Bulling. "End-to-end eye movement detection using convolutional neural networks". In: *arXiv preprint arXiv:1609.02452* (2016).

[3] Benjamin J Li et al. "A Public Database of Immersive VR Videos with Corresponding Ratings of Arousal, Valence, and Correlations between Head Movements and Self Report Measures". In: *Frontiers in psychology* 8 (2017), p. 2116.

[4] Kemal Polat and Salih Güneş. "Classification of epileptiform EEG using a hybrid system based on decision tree classifier and fast Fourier transform". In: *Applied Mathematics and Computation* 187.2 (2007), pp. 1017–1026.

Project code: https://github.com/john-hewitt/cs229-head-tracking