Classifying Depression with Casper and Genetic Algorithm Metaheuristic for Feature Selection

Barclay Zhang
Research School of Computer Science, Australian National University
Canberra, Australia

u6669143@anu.edu.au

Abstract. Depression as a clinical disorder is difficult to diagnose correctly however we can possibly sense depression in others and give off subconscious physiological responses. CasPer is a neural network algorithm that has shown strength to generalise well on toy problems and Genetic Algorithms can perform strongly on hyperparameter tuning tasks such as feature selection. Pupil Dilation Data of observers of depressed people were collected from a previous experiment and used to train neural networks to help diagnose depression. However the networks showed poor results with the CasPer network achieving only a 37.5% accuracy in classifying depression and CasPer + Genetic Algorithm performing slightly worse at only a 36.5% accuracy. However both of these surpass the baseline result of a simple neural network which only achieved 22% accuracy.

Keywords: Casper, Neural Network, Depression, Genetic Algorithm

1 Introduction

Depression is a chronic, widespread, and internalising mental disorder. The general symptoms involve a constant long term lingering feeling of moodiness, sadness and/or apathy for no apparent reason. It is different to general mood fluctuations and acute emotional responses that people generally face in daily events. Depression can be debilitating to those affected by it, making even the simplest of tasks and responsibilities difficult and draining. The symptoms and severity of depression lie on a wide spectrum from a general feeling of sadness, to at worst self-harm, suicidal thoughts, behavior and even attempts (N. Cummins, S. Scherer, J. Krajewski, S. Schnieder, J. Epps, and T. F. Quatieri, 2015). Due to the severe impact depression can have on people, it is paramount there are effective diagnostic methods available.

Unlike the majority of clinical diseases and disorders which can receive (relatively) conclusive diagnoses from hard evidence such as blood tests, laboratory tests etc. Depression diagnosis using hard evidence is currently limited due to the nature of it as a mental disorder. Therefore, most current techniques for diagnosis lead it to require qualitative judgements from a clinician. These methods are not only subjective and therefore biased but can also be very time consuming. Clinician diagnosis generally involves an interview styled assessment such as the Hamilton Rating Scale for Depression (HAMD) which rely heavily on patients opening up and willingly honestly sharing themselves and their experiences. Theses assessments will also be heavily dependent on individual clinicians' abilities and experience due to the subjective qualitative nature of depression as a clinical disorder.

With a modern boom in machine learning technologies and its proven ability in assisting in difficult classification tasks, it is natural to therefore investigate the possibilities of having machine learning algorithms assist and make possible depression diagnoses for us. There are many existing approaches of using a patient's physiological signals to predict depression. However, in this paper we will investigate and discuss depression classification using an observer's physiological signals, in particular we will attempt the classification of depression from an observer's pupil dilation physiological signal. There are also many different classification, machine learning and hyper parameter tuning techniques to choose from, in this paper we will take a deep dive into using the CasPer Neural Network training algorithm and also extend it with feature selection optimized using a genetic algorithm and compare the results between each other and a general multilayer perceptron model in classifying depression based on an observers pupil dilation.

1.1 The Data

In order to achieve our task, first a dataset is needed. The dataset we will use comes from the paper *detecting emotional reactions to videos of depression* (X. Zhu, T, Gedeon, S Caldwell, R Jones). The paper collects a dataset of various physiological signals of observers observing videos of people with varying levels of depression, these are then fed as inputs to a neural network, where they are then used to help create a classifier for depression, the labels for the dataset are the actual depression levels from 1-4 referring to no depression at 1, mild depression at 2, moderate depression at 3 to severe depression at 4. The paper uses a simple multilayer perceptron model and finds that the Pupil Dilation features provide the strongest and most accurate classifications, hence in this paper we aim to extend on this problem. The pupil dilation dataset was collected from 12 observers each observing 16 videos of people with varying levels of depression giving a total of 192 total datapoints. Pupil dilation metrics were collected during the original experiment and are split into 39 different features. These extracted features are time domain features (for example average pupil size, max, min, standard deviation, and variance etc.) of each individual pupil sizes, since each individual has different pupil sizes and physiological features in general, the data has been normalized towards each individuals maximum and minimum sizes.

1.2 CasPer

CasPer (T.D. Gedeon & H.K. Treadgold, 2006) is a neural network training algorithm that builds on using the Cascade Correlation Algorithm (Fahlman and Lebiere, 1990). The general idea of Cascade correlation is to automatically find the optimal network structure. Cascade Correlation is a constructive algorithm which begins by training a network that is fully connected from input to output and then a training loop occurs where a new neuron is installed and trained to maximise the correlation between the neurons output and the networks training error, freezing existing weights while doing so. This is repeated, so neurons get added one at a time producing a cascading effect of new layers. In CasPer however, we use RPROP (Resilient back-propagation) to train the whole network at once but with each parameter having different learning rates based on when the weight was added to the network. In the few toy examples in the paper, Casper was shown to produce more compact networks with less redundant parameters while simultaneously being able to generalize better than the normal Cascor training algorithm. Hence, in this paper we will investigate if the CasPer learning algorithm can also generalize to this complicated real word task and assist in helping diagnose depression correctly.

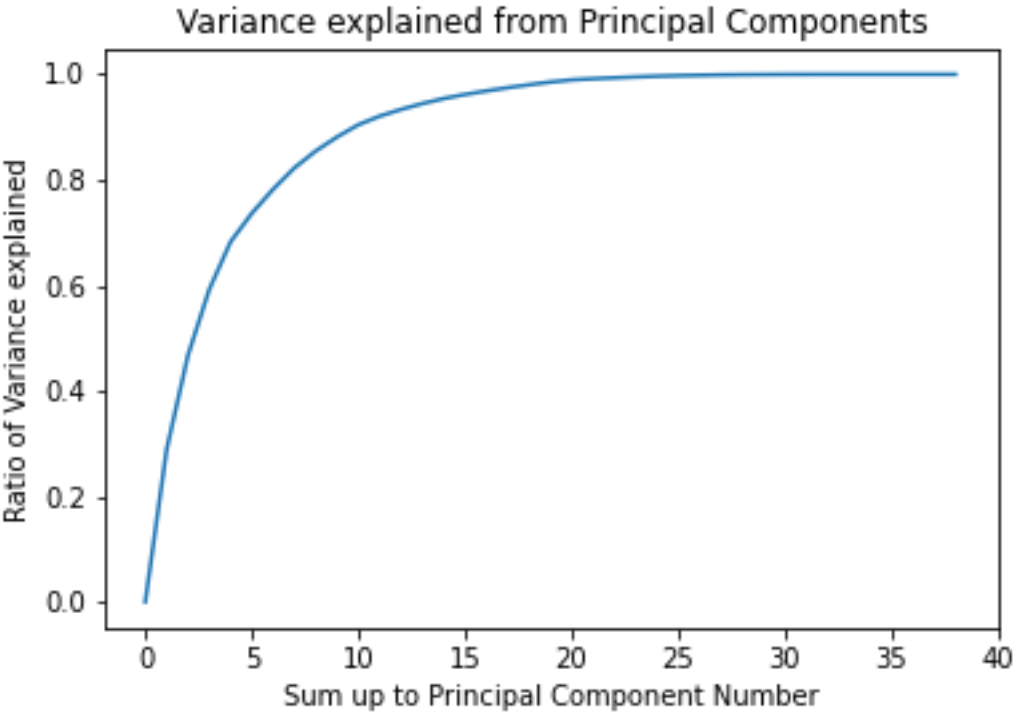
1.3 Genetic Algorithms

One of the difficulties with using neural network architectures is hyper parameter tuning, this includes the problem of feature selection. Genetic algorithms provide a framework and metaheuristic for hyper-parameter tuning and therefore can be used to select the features used by a neural network *(O Babatune, L Armstrong, J Leng, D Diepeveen 2014)*. Genetic algorithms are inspired by biology and evolution, hence the name. An initial pool of candidate solutions are randomly generated and are evaluated on based on a fitness function. A crossover stage then takes strong performing solutions and combines them to provide offspring which then in turn can be used to create future offspring in further iterations. The aim is to then to find the most optimal candidate solutions over multiple generations and by breeding together different solutions. This can be used by neural networks for feature selection by selecting random features which act as the candidate solutions, then through training a neural network for each candidate solution, the strength of the solution can be evaluated by a fitness function which should correlate with the test accuracy of each respective network. Afterwards, the best performing solutions can be crossed over and mutated to provide possible solutions that can give better results with the end goal of finding the best combination of features for the network to use.

2 Methodology

2.1 Preprocessing

The dataset used for this paper was already preprocessed as described in the dataset paper, firstly since each physiological signal is individual dependent the range of each signal could be vastly different from person to person. Therefore, all the signals were scaled between 0 and 1 scaled using each individual's min max ranges. This preprocessing translated to original features presented by the dataset; however, the dataset also included several occurrences-based features which were not transformed, these occurrences and count based features included occurrences of reaching peak pupil dilation etc. and were presented as a integer numbers. Due to this the scale of this data was not consistent with the remaining already pre-scaled features, therefore these features were also scaled, this is possible due to the ordinal nature of the occurrences and the scaling preserving the ordinal information. After the scaling we are left with a dataset with 39 features with values ranging between 0 and 1,



2.2 Feature Creation

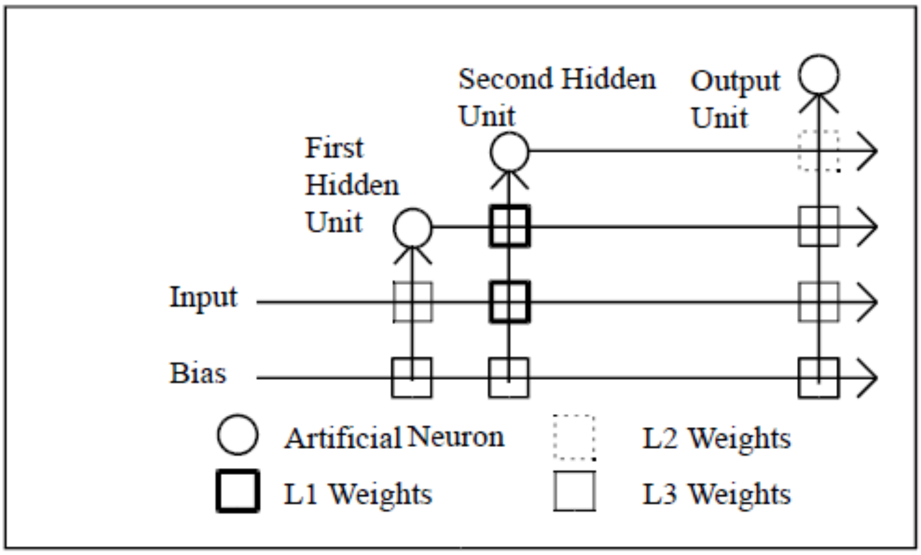
The dataset only has a limited number of data points, in total with only 192 datapoints and even less when using a train test split. Due to the limited number of datapoints, we performed a dimensionality reduction on the data, this was done using principal component analysis. After transforming the data, the variance explained by the sum of the first n principal components was recorded and visualized in figure 1. The first 21 principal components make up for 99.38% of the variance of the data, therefore by discarding the remaining 18 principal components we can reduce the number of features from 39 to 21. The order in which we apply the data preprocessing matters, and it is paramount the test set receives the transformations fitted and transformed on the training set. These 21 principal components therefore then make up the features that we will use for feature selection.

2.3 Neural Network Baseline

A baseline simple neural network architecture is first used to establish a baseline result, the network architecture used is a single fully connected hidden layer with 50 neurons and the sigmoid activation function, using the cross entropy loss and Adam optimizer using backpropagation. This baseline was chosen as it was the specified architecture described in the original dataset paper.

2.4 CasPer Network for classification

The CasPer Algorithm constructs a Cascade network, the algorithm works in a training loop between training the network with RMPROP, adding a neuron when the training is not improving and then readjusting learning rates. The learning rates are adjusted based on different regions of the network, named L1, L2, and L3. L1 is made up of all weights connecting to the newest hidden neuron, L2 refers to all the weights connecting from the newest neuron to the output layer and L3 is all the remaining weights.



In general, L1 >> L2, L3 (T.D. Gedeon & H.K. Treadgold, 2006), for this paper we settled on the following values of 0.02, 0.005,0.001 which were decided on to balance between convergence and optimizing training time. The original CasPer paper also described a P hyperparameter which is used to describe and scale the number of training epochs before checking if a new neuron needs to be added. Due to the complexity of the dataset for this paper, the P value was chosen to be 100 which is higher than the values chosen for the toy problems in the original CasPer paper. In each neuron, a leaky relu was used. This is because by nature cascading networks can get quite deep which greatly amplifies a disappearing gradient problem prevalent when using a sigmoid activation. Due to the task being a classification task a SoftMax function was also applied to the output neurons.

2.5 CasPer + Genetic Algorithm

Genetic Algorithm will also then be used for hyper parameter selection for a third test. The candidate solutions are represented as a 21 long length chromosome with binary encodings where the i^{th} gene represents using (1) or not using (0) the i^{th} feature, the population is initially initialized randomly. The population is set to 20 and the algorithm runs over 100 generations with a size 3 tournament. A two-point crossover is used with 80 percent probability and a 20 percent mutation probability is used for a bitflip mutation. The fitness function used to evaluate each candidate solution will therefore be the test accuracy result after each neural network has been trained using the respective chromosome indicating the features to be used, hence the objective direction will be to maximize the accuracy. These hyperparameters were used to allow large enough sample population breeding and generational growth to occur while noting the limited computing resources available, due to the nature and time required for executing genetic algorithms with a backpropagation trained network, hyper parameter tuning for the genetic algorithm was infeasible to do.

2.6 Leave one out Validation

For our dataset, the normal K-fold validation used is not sufficient due to each individual being represented by multiple datapoints. This means that there is correlation that exists outside the data. To account for this we use Leave one out validation in which the data points for one individual becomes the testing set and everyone else is used as the training set. We repeat for all combinations and average out to calculate the final results, this means we total to running an average over 12 training and test cycles.

2.7 Evaluation and benchmarking

To evaluate the effectiveness of the models, we will use the multiclass precision, recall and F1-scores as evaluation measures. For a depression class D, the precision refers to the proportion of individuals with D that are predicted to be of class D. Recall refers to the percentage of individuals correctly predicted to be of D among all individuals classified as D. The F1 score acts as a medium between the two taking the harmonic mean of recall and precision. An overall score can then be calculated by taking the average over each depression level to produce a final metric to score each model by.

3 Results

As seen from figure 3 and figure 4, the simple network was not a good classifier for detecting depression levels achieving only a 22% accuracy which is even worse than the baseline random chance classifier of 25% given from having 4 evenly distributed classes. Despite most of the simple networks scores being consistently low, the big standouts were for moderate depression achieving only a precision of 0.12, recall of 0.06 and F1 of 0.08 which are all extremely low. Contrasts this with the Recall of 0.5 for Severe depression it shows that the Simple network was very biased towards severe depression and was completely unable to capture the information on moderate depression.

Figure 3		Simple Network	Casper Network	Casper Network + Genetic Algorithm
No Depression	Precision	0.27	0.28	0.28
	Recall	0.20	0.42	0.40
	F1	0.23	0.35	0.34
Mild Depression	Precision	0.21	0.34	0.33
	Recall	0.20	0.35	0.34
	F1	0.21	0.35	0.33
Moderate Depression	Precision	0.12	0.50	0.47
	Recall	0.06	0.28	0.31
	F1	0.08	0.39	0.39
Severe Depression	Precision	0.28	0.41	0.40
	Recall	0.5	0.43	0.45
	F1	0.36	0.42	0.42

Figure 4	Simple Network	Casper Network	Casper Network + Genetic Algorithm
Overall Accuracy	22%	37.5%	36.5%

The standalone Casper Network did however perform better than the simple multilayer perceptron did, it ended up with an overall accuracy of 37.5% which shows it did perform better than random chance would in this case. The Casper network showed very consistent results compared to the wild varying results of the baseline simple model. In particular its precision score for Moderate depression was 0.5 which soars above the measly 0.06 of the Simple MLP Network. For almost all categories, the Casper Network proved to be a stronger classifier, by a healthy margin of 15.5%. There are no standout deficits or strong points for Casper in classifying between the different levels showing it had very little bias towards a particular depression level. The results also showed the genetic algorithm CasPer network having very similar performance to the CasPer Network without the genetic algorithm. Almost all scores coincided completely within a small delta when compared to the standalone CasPer results however in almost all cases they were slightly slower than standalone CasPer results.

4 Discussion

The results demonstrated that our implementation of CasPer with and without the Genetic Algorithm and the single multilayer perceptron model were both quite incapable of effectively classifying and therefore diagnosing depression through the use of observers' physiological signals. One of the big reasons this could possibly come from, is the wide distribution of the data, in particular it is possible that each individual person will have a completely different physiological reaction towards the same observations, hence the ability for these neural network models to generalize will be heavily limited. It is also possible that only using twelve people alone for the dataset, does not capture enough of the distribution of the general population well enough. Twelve is statistically a very small number and so the dataset used very well could have been biased heavily by the individuals that the models were trained on. In the future, a key improvement would be to use larger datasets with many more data points. Especially such that the number of datapoints is much larger than the dimension of our features.

The CasPer models also on average, averaged adding around 12 hidden neurons and hence layers towards the network architecture. This meant that the models were significantly deeper than the simple model and hence training time was also much longer despite still having an overall smaller number of parameters when compared to the fully connected 50 hidden neuron network. This problem severely limited the ability for the genetic algorithm to converge, genetic algorithms as a whole require large and vast amounts of training time and resources due to the large number of models being trained. This could possibly explain the lackluster results provided by the genetic algorithm boosted model, the epochs used were insufficient to truly reach a converging criterion and instead were required to be stopped due to the extreme computational time and resources that were not available at the time of writing of this paper.

It is therefore also very obvious that with our large networks with many parameters and limited datapoints, that the models themselves ended up overfitting, this can be seen by looking at the training accuracies and losses, the simple model during training consistently reached 100% training accuracy which is a very bright indicator for overfitting. The CasPer network however averaged only reaching 82% training accuracy. The drastic difference in performance between the train and test sets of 78% for the simple model and 44.5% for the CasPer model clearly shows the evidence of two major possible problems, the first being overfitting and secondly the uneven and drastically polarizing distribution of the data. The difference of performance between the models however does possibly show that CasPer as an algorithm provides a method of regularization to prevent overfitting. This could be due to the cascading property where the network size is only grown as needed and hence can reduce the amount of unnecessary saturated parameters. Future work in the space of CasPer can focus on this property.

The similarities in results of the CasPer network with and without the Genetic algorithm can possibly be explained by several phenomena, firstly as mentioned before it is entirely possible the genetic algorithm simply did not converge. Increasing the generations would cause the already long training time to skyrocket, therefore, to prevent this future genetic algorithm attempts will and should require larger allocations of training and computational resources. Another possible be reason would be the preprocessing and feature extraction done using the principal component analysis, this may have already removed unneeded parts of the data and already condensed the features into all the information needed without still maintain overly large number of features. This theory is backed up by the genetic algorithm performing overall slightly worse than just using all 21 features. Since the genetic algorithm solution generally only produced solutions with less than 21 features, the features missed may have missed key important patterns and variance in the data. Future works could aim to perform genetic algorithm on the raw normalized features without the dimensionality reduction.

The results of this paper are contradictory to the parent original dataset paper in which they achieved a result of 88% and 92% test accuracy using the simple network architecture and genetic algorithms to classify depression. The drastic difference in performance shows the likely possibility of a mis-implementation, very likely in the data preprocessing stage. Future work should focus more on understanding the differing distribution of individual observers and looking at differing physiological signals to classify depression.

5 Conclusion

The CasPer algorithm was implemented and used on a dataset consisting of the physiological signals of observers of depressed people, it was found that the baseline method of a simple single 50 neuron hidden layer MLP model achieving 22% accuracy performed considerably worse than the CasPer algorithm at 37.5% which in turn out performed the combined CasPer Genetic Algorithm model which achieved an accuracy of 36.5. Despite CasPer algorithms more accurate classification it was still too inaccurate to provide any substantial benefit to clinical depression diagnosis.

6 References

1. N. Cummins, S. Scherer, J. Krajewski, S. Schnieder, J. Epps, and T. F. Quatieri, “A review of depression and suicide risk assessment using Speech analysis,” Speech Commun. , 2015.

2. T.D. Gedeon & H.K. Treadgold, “A Cascade Network Algorithm Employing Progressive RPROP, 2006”

3. X. Zhu, T, Gedeon, S Caldwell, R Jones “Detecting Emotional reactions to videos of depression” 2019
4. Fahlman, S.E., and Lebiere, C. (1990) The cascade-correlation learning architecture. In *Advances in Neural Information Processing II*, Touretzky, Ed. San Mateo, CA: Morgan Kauffman, 1990.
5. O Babatune, L Armstrong, J Leng, D Diepeveen, “A Genetic Algorithm-Based Feature Selection” *Edith Cowan University* 2014