Colorado State University
Computer Science Department

Modeling and Classification of EEG using Recurrent Neural Networks

Elliott Forney and Charles Anderson

Why Recurrent Artificial Neural Networks?

Patterns in EEG are have a very temporal structure. Most current approaches use one of two methods to account for the temporal nature of EEG.

Frequency based approaches create features using power or energy spectral densities. These features are then used as inputs to a static classifier. This method is limited in that spacial patterns, i.e. correlations between channels, are not considered.

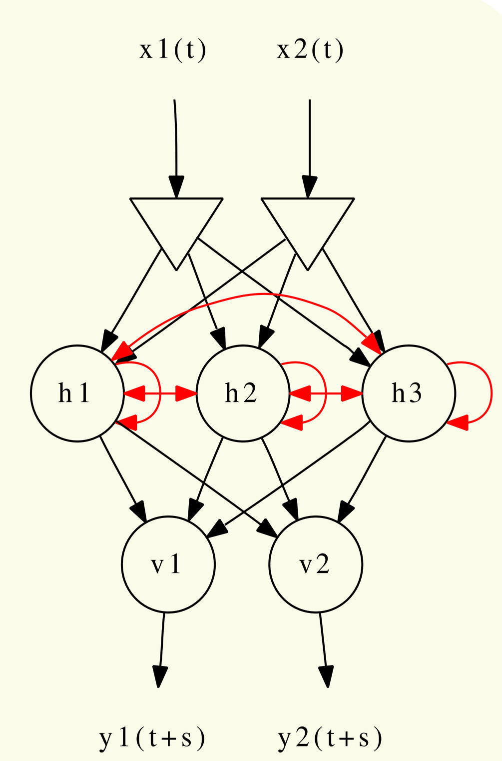
Lag based approaches embed a number of timesteps into a single input to a static classifier. This method is limited by the number of embedded time steps and a very high dimensional input space.

Neither of these methods are very biologically plausible or capable of forecasting or predicting EEG.

We propose a different approach. First, an Artificial Recurrent Neural Network (RNN) is used to build a model for each type of EEG by forecasting the signal several steps ahead in time. Classification is performed by applying each RNN and selecting the class associated with the model that performs best.

Since RNN's have feedback connections they have an intrinsic state. This allows RNN's to learn complex nonlinear spatio-temporal patterns.

Elman Recurrent Networks



Currently, we are using Elman Recurrent Networks.

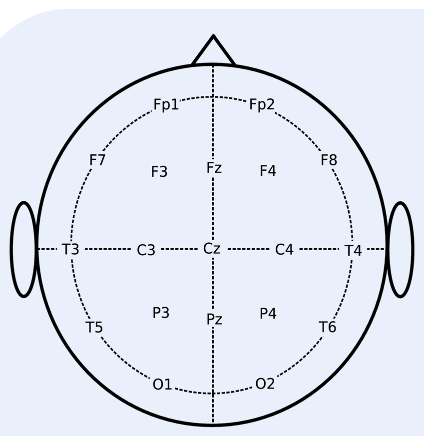
Elman Networks contain a single hidden layer with a hyperholic tangent transfer function and a visible layer with a linear transfer function.

The hidden layer is fully connected, including recurrent connections. This allows the network to maintain a state and utilize previous input values

Backpropagation through time (BPTT) is used to approximate the error gradient and a scaled conjugate gradient (SCG) algorithm is used to adjust connection weights, minimizing training error.

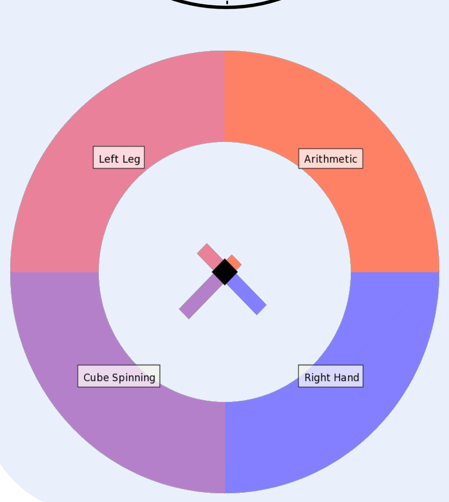
Experiments with other network architectures and training algorithms are ongoing, including Echo State Networks, Fully Connected Networks, ALOPEX, and CMA-ES

Data Acquisition



Electroencephalogram is sampled at 256Hz using a Mindset-24 digital amplifier. A 19 channel subset of the 10-20 system is used with earlobe references.

Subjects are presented with a visual queue that selects one of several mental tasks to perform, such as imagined motor movement, visualization, arithmetic tasks.



An approximated bandpass filter is used to attenuate noise and artifacts. The Best classification results have been achieved with a very narrow passband, i.e. 8-12Hz or 46-50Hz

The data is then downsampled and normalized to zero-mean and unit standard deviation in order to improve training speed.

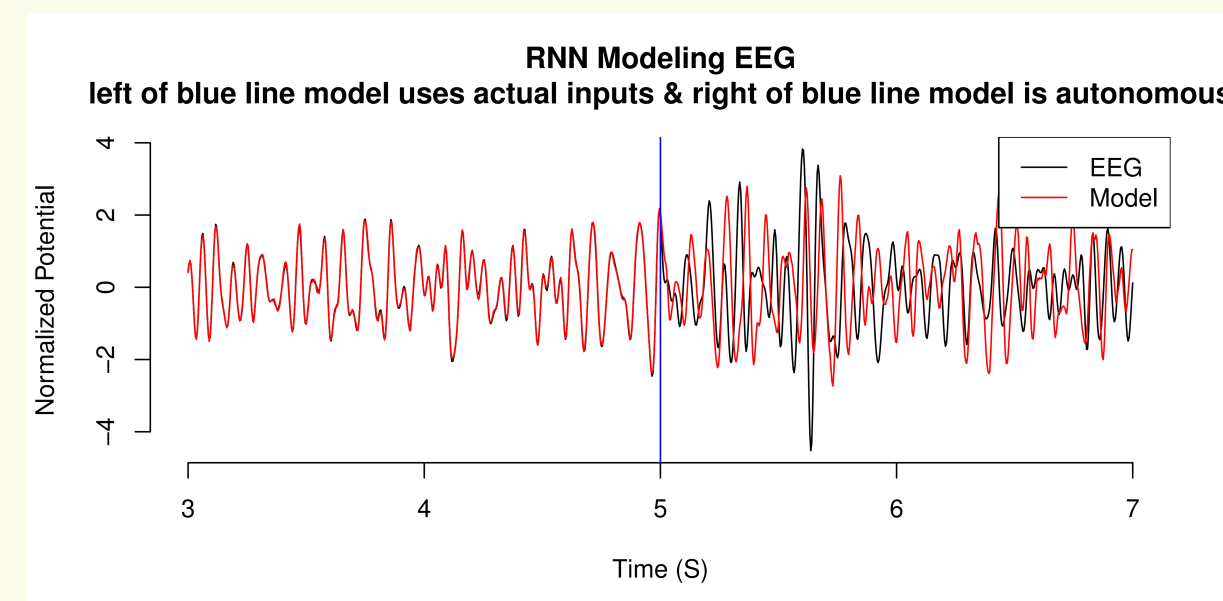
Modeling EEG

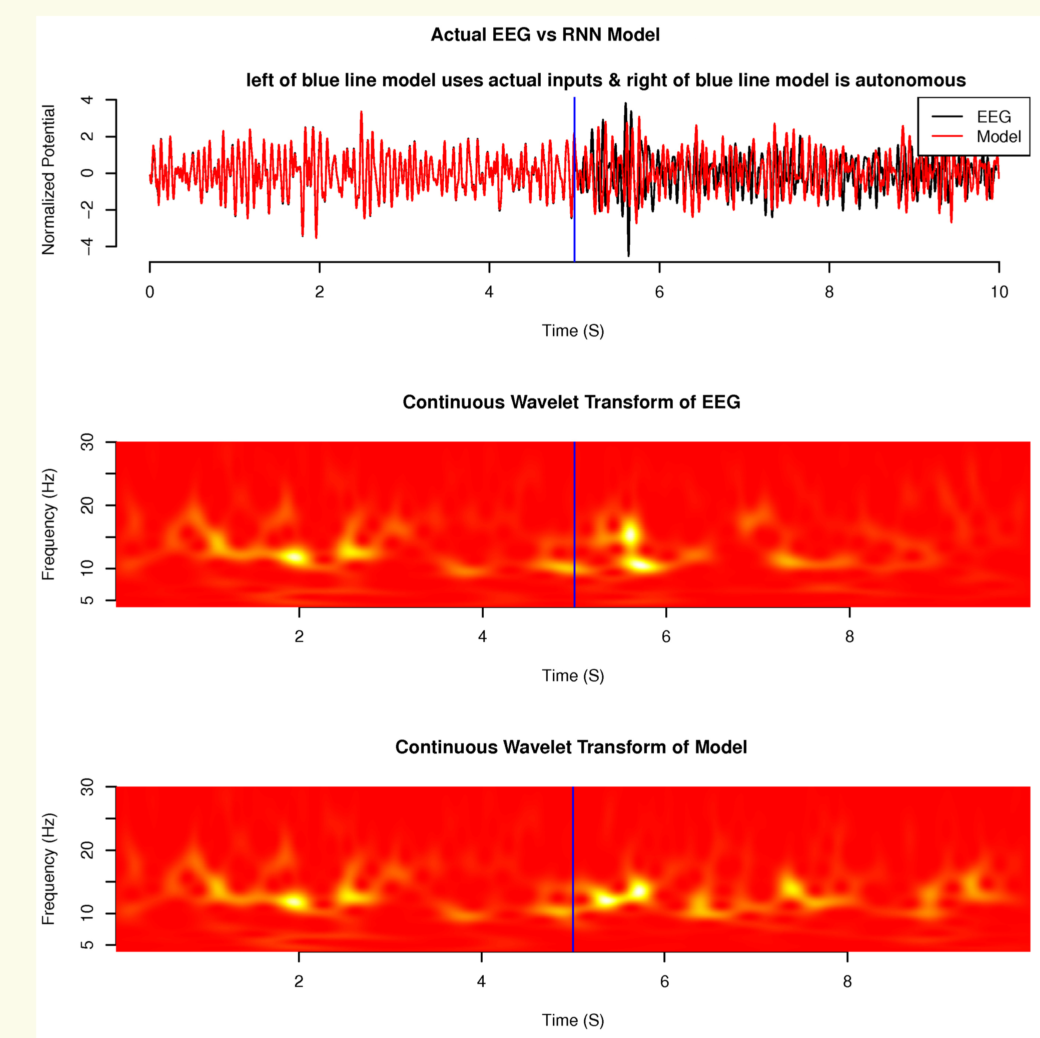
An EEG sequence is modeled by training an RNN to forecast the sequence one or more steps ahead in time.

In this way, the network input is the current timestep x(t) and the target is s timesteps in the future x(t+s)

Such models perform well over arbitrary EEG sequences achieving an RMSE on the order of 0.01

The model can also be allowed to take its previous predictions as inputs, forming an autonomous and dynamical system.





With few hidden units, this autonomous system quickly flatlines. With several hundered hidden units, however, very "EEG like" rhythms are produced. A continuous wavelet transform of real and predicted EEG appears to confirm this upon visual inspection.

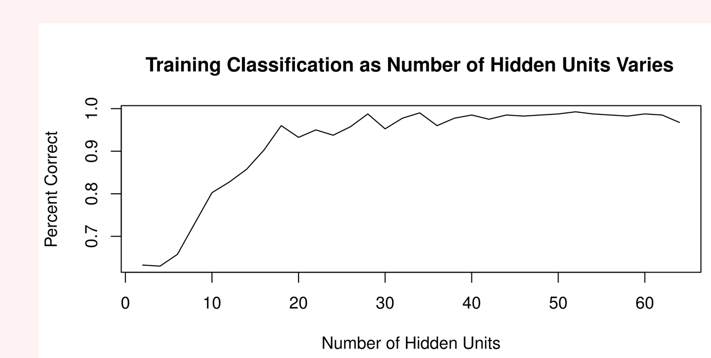
Classification of EEG

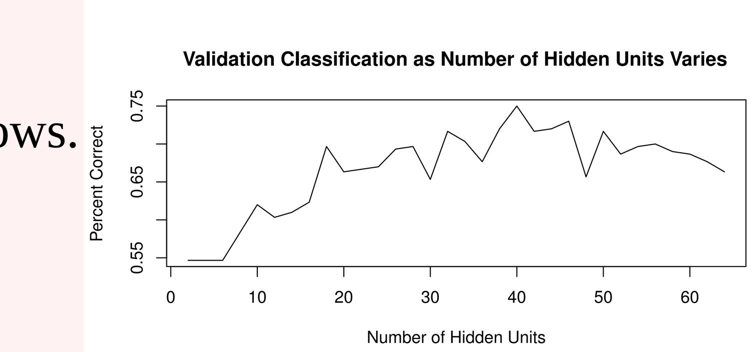
A separate RNN is trained over several sequences recorded during each mental task. This results in an "expert" at predicting each type of EEG.

Classification of previously unseen sequences proceeds by applying each RNN and choosing the class associated with the model that produced the lowest error.

Results are improved by accumulating the errors over half second windows.

Data appears to be overfit with more than about 40 hidden units, despite the rich models produced with many hundreds of hidden units.





Preliminary results achieve 75% correct on a two task problem and 46% correct on a four task problem.

Actual \ Predicted	Motor Right	Motor Left	Visualization	Arithmetic
Motor Right	40%	43%	7%	10%
Motor Left	20%	57%	23%	0%
Visualization	13%	23%	60%	4%
Arithmetic	23%	33%	17%	27%

Actual \ Predicted	Motor	Visualization
Motor	83%	17%
Visualization	33%	67%