Feature Learning on EEG Data using Autoencoders for Epilepsy Diagnosis

Gentry Atkinson (gma23@txstate.edu)

1. Introduction

Autoencoders, which were first described by Mark Kramer in 1991, were primarily used for dimensionality reduction. Their behavior allowed them to learn the structure of a dataset in the hidden layer of a shallow ANN. Recently researchers have been working with autoencoders as a tool for feature learning and automated feature extraction.

Feature selection is often one of the most difficult phases of modern data analysis. As the focus of data science shifts away from neatly curated academic data sets and towards ses collected "in the wild", researchers are finding new tools to work with data that is much larger and much noisier than what traditional tools can easily handle. Deciding what features should be focused on which ones can be safely ignored can have an enormous impact on the ultimate success or failure of a project. The ability of ANNs to learn non-linear relationships means that they can detect and define features within data that would have been missed by traditional methods. The has been the foundation of Deep Learning.

Autoencoders have already proven to be a valuable tool for feature extraction in many fields. However feature extraction in time series data offers many challenges that static data does not. This project proposes to measure the effectiveness of autoencoders for feature extraction on time-series electroencephalographic data. Many researchers have demonstrated the ability of machine learning techniques to correctly recognize epileptic episodes on EEG data. To gauge the effectiveness of autoencoders for feature extraction a classifier will be trained on the features extracted and its accuracy of the new system against that of the established methods. Achieving a segment-wise accuracy comparable to other research groups will show that autoencoders are effective.

2. Background

2.1 Electroencephalography (EEG) was first described by Richard Caton in 1875. Its purpose is to measure small voltage fluctuations with the neurons of the brain. It is used as a tool for the diagnosis of epilepsy, sleep disorders, coma, encephalopathies, and brain death.

2.2 Autoencoders are a for of simple ANN whose basic structure is a network that is trained from a source set back to itself. A properly trained autoencoder will always output the same set that is provided as input. By reducing the size of the hidden layer (causing a "bottleneck") researchers can create a numerical set which is smaller than the input but contains all of the same information (by virtue of the fact that the input can be losslessly re-derived from the hidden

layer). The values from the hidden layer can be used as a dimensionally reduced representation of the input or (potentially) as a feature extracted from the input.

2.3 Epilepsy is a broad name for several neurological disorders that all share a common symptom, recurrent seizures. Although several forms of trauma and disorder have been shown to result in epilepsy there is no one known root cause. EEG is often employed by trained professionals in the diagnosis of epilepsy.

3. Related Work

Autoencoders were first described in "Nonlinear Principal Component Analysis Using Autoassociative Neural Networks" by Mark Kramer in the AIChe Journal in 1991. As the title suggests the proposed usage was only dimensionality reduction rather than any sort of feature learning. The novel contribution was that autoencoders could learn non-linear relationships while PCAs are purely linear.

The application of autoencoders as deep feature extractors was first proposed by Quoc Le in *"Building High-Level Features Using Large Scale Unsupervised Learning"* at the 2013 IEEE ICASSP. This shows that the usefulness of autoencoders as feature extractors was recognized in the early days of deep learning. The advantage of autoencoders is that their auto-associative property makes them much easier to train than other deep architectures. Effectively they use supervised training techniques and apply them to a problem of unsupervised learning.

An example of feature learning applied to bioinformatics is "Application of Deep Learning in Neuroradiology: Brain Haemorrhage Classification Using Transfer Learning" by Awwal Muhammad Dawud , Kamil Yurtkan , and Huseyin Oztoprak published in Hindawi: Computational Intelligence and Neuroscience in June of 2019. This work shows that there is a specific interest in deep learning applications of medical diagnoses. However, this paper acts of image data rather than time series EEG data which is a different set of challenges altogether.

Autoencoders are mentioned in "A review of unsupervised feature learning and deep learning for time-series modeling" by Martin Längkvist, Lars Karlsson, and Amy Loutfi in Pattern Recognition Letters in 2014. Here, autoencoders are applied to video images and temporal coherence is maintained by manipulation of the loss function in training the autoencoder. The techniques should be usefully applicable to medical data.

4. Contributions

Feature learning on time series data is a difficult challenge. Autoencoders have been shown by other researchers to be a valuable tool for feature extraction. This project will test the ability of autoencoders to extract features which maintain the temporal locality of extracted features. Effective feature extraction on bioinformatic data, such as EEGs, could open up valuable avenues of research and help develop diagnostic techniques for conditions which are more nuanced that epilepsy, such as PTSD and anxiety.

5. Methodology

In a small deviation from the original proposal the project has adopted a convolutional autoencoder rather than training a conventional autoencoder on a series of "windows" sampled from the input data. The new model trains a series of kernels to recognize significant patterns in 16-sample (80ms) portions of the input signal. The new model trains one convolutional layer and one dense layer as an encoder. The decoder layer is composed of a dense layer whose outputs feed into a transposed convolutional layer. The transposed convolutions recompse the 16 sample kernels from their inputs, in effect reversing the first convolutional layer. The convolutional model was adopted to make better use of existing software packages rather than saddling the project with reconstructing tools which do not directly contribute to the novel contributions of the project.

Keras was chosen as a software package to support the construction of the machine learning model employed by this project. Python scripts are being constructed using a text editor. The extracted features are being written to a csv file so that a preliminary analysis can be done. At this time that analysis is being done in MATLAB which provides a good suite of tools for visualizing time series data.

When it has been confirmed that by the preliminary analysis that the autoencoder is producing a usable set of features then another python script will be generated to train an SVM on the feature set. This SVM will be trained to classify "seizure" or "not seizure" in the input data. Seizures are present in exactly 20% of the provided samples. If this SVM is able to achieve similar accuracy to *"Building High-Level Features Using Large Scale Unsupervised Learning",* which has provided the data set that this project is being built on, then we will have shown that convolutional autoencoders are an effective tool for learning locality-sensitive features from time series data.

6. Obstacles

The original data set contained data segments composed of 4098 (20,000ms) samples. Although that is an input size that is well within the realm of possibility for convolutional networks (consider that this is the number of samples in a 64x64 grayscale image) it was substantially slowing down the training of the network. In order to make the network trainable within a manageable time period the samples were divided into 8 512-sample segments (2500 ms each). Two and a half seconds should be more than sufficient to recognize a seizure in the sample data and a model trained on 2.5 second samples rather than 20 second samples could be much more useful in a potential real-time monitoring application.

The initial results of the project have shown that the encoder-decoder pair are having trouble training themselves to the sample data. Some progress has been made by adjusting activation functions and the number of nodes in each layer. Further trials will be necessary to adequately refine the model.