Focal onset seizure prediction using convolutional networks

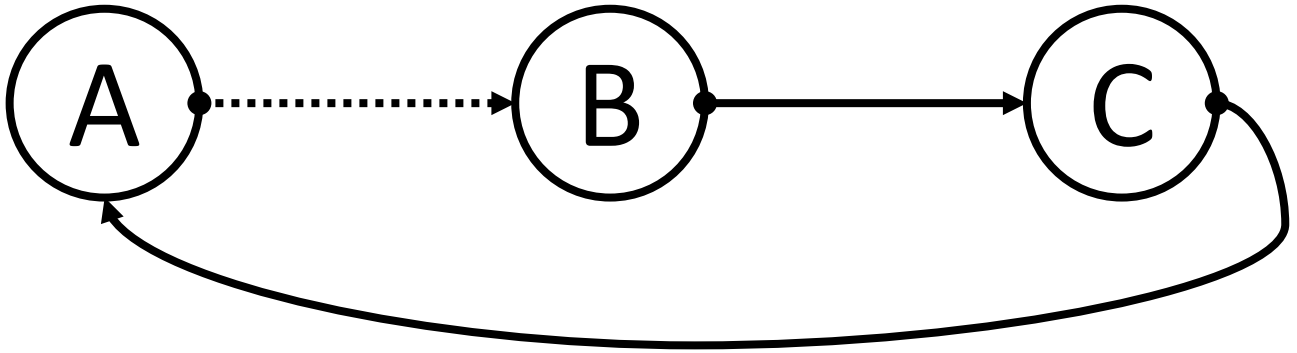
Haidar Khan, Lara Marcuse, Madeline Fields, Kalina Swann, Bülent Yener

Setting and Motivation

- We observe a system that generates a time series signal while transitioning between states
- With a dataset of time series with labeled states, we can train a discriminative model
- Use model to predict next states given previous data.

- Predicting pre-seizure transition in patients with epilepsy*
- Computer network intrusion detection
- Climate change detection

High level problem



- Problem: Only some states labeled.
  - $A \rightarrow B$ transition unknown, only that it occurs some time before $B \rightarrow C$
- To learn a good discriminative model, need to assign labels to the time series.
  - unsupervised/semi-supervised learning

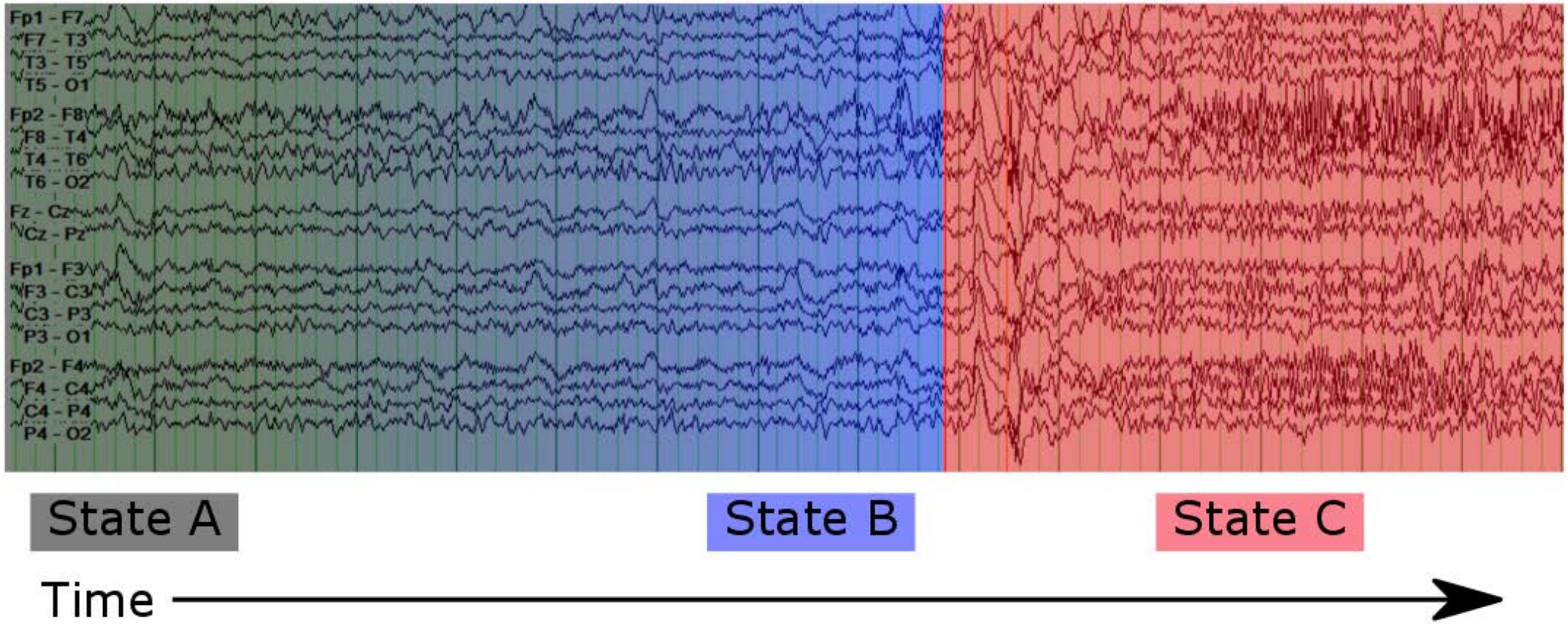


Figure 1. Example of a system generating a multichannel signal transitioning between states. The transition from State B to State C can be easily marked, but the transition from State A to State B cannot be marked. This results in a region of uncertainty about the state of the system.

Problem definition

- Given a time series X where $\mathrm{X} = \{x_1, x_2, \ldots x_T\}$
- Assume X is generated by a process which undergoes a transition from state A to state B,
  - with probability distributions P_A and P_B respectively and $P_A \neq P_B$.
- A time t is the change point if:

$$\{x_1, x_2, \ldots x_t\} \sim P_A$$
$$\{x_{t+1}, x_{t+2}, \ldots x_T\} \sim P_B$$

Virtual classifiers (VC) - Theory

- If we consider the change point detection problem as an optimization problem of the form:

$$\max_t D(P_t(x|A), P_t(x|B))$$

- where $D(\cdot,\cdot)$ is a divergence measure between the two distributions.
- Idea is to approximate $D(P_t(x|A), P_t(x|B))$ with classification accuracy

Time series of feature vectors $\{x_k\}_{k=1}^T$ with state space $\mathcal{X} = \mathbb{R}^d$.
Time t defines a split of the time series into disjoint sets $A = \{x_1, x_2, \dots, x_t\}$ and $B = \{x_{t+1}, x_{t+2}, \dots, x_T\}$

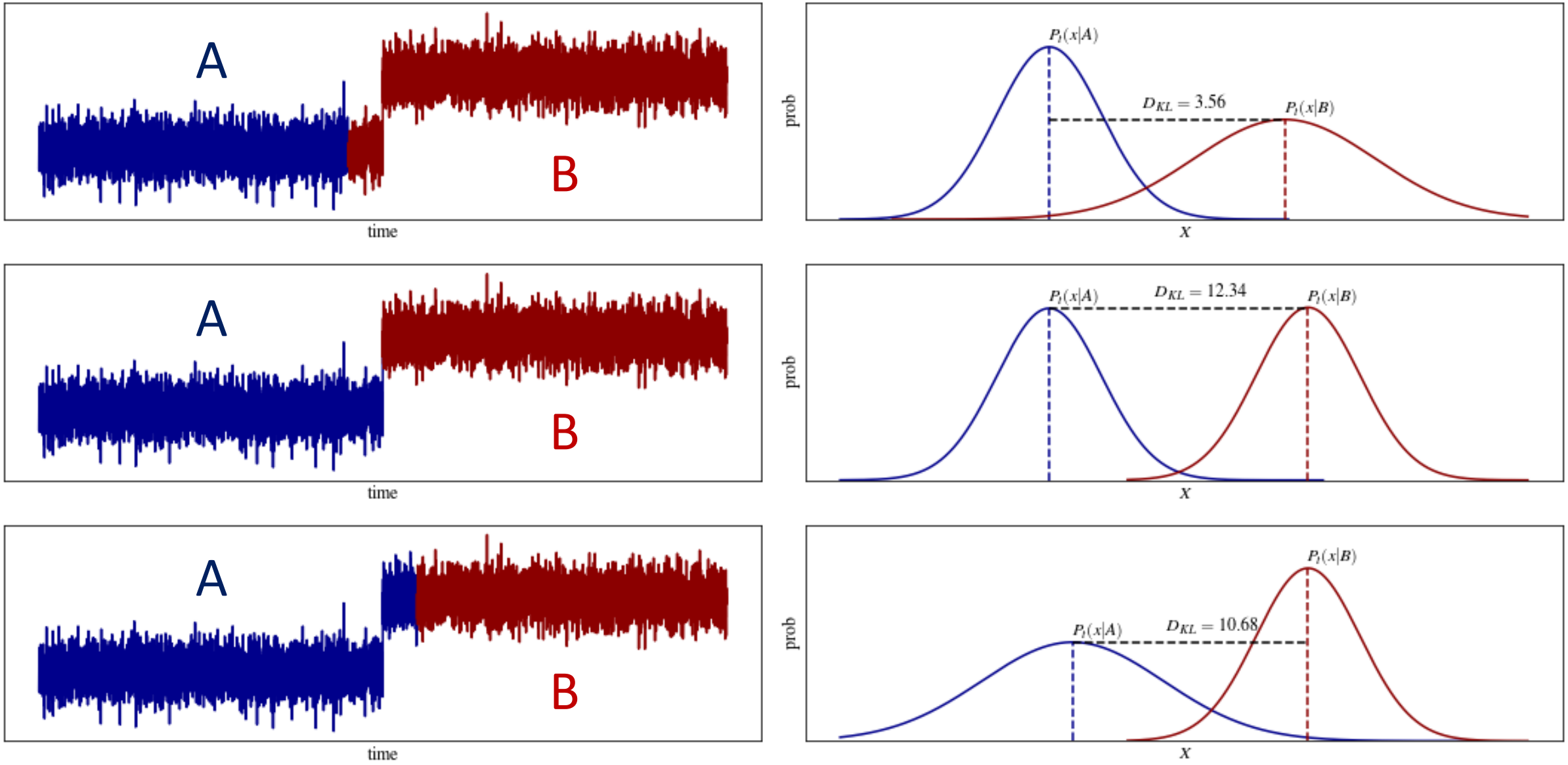


Figure 2. Example of a Gaussian noise signal undergoing a mean shift. By splitting the signal into segments A and B at different time points and approximating the conditional probability distributions with Gaussians, we see the KL-divergence is maximal when the split matches the change point.

Approximating KL-divergence with VC

- Using the KL-divergence for $D(\cdot,\cdot)$ yields:

$$\max_t \sum_{x \in \mathcal{X}} P_t(x|A) \log \left(\frac{P_t(x|A)}{P_t(x|B)} \right)$$

$$\max_t \sum_{x \in \mathcal{X}} P_t(x|A) \log P_t(x|A) - \sum_{x \in \mathcal{X}} P_t(x|A) \log P_t(x|B)$$

- Assuming the entropy of $P_t(x|A)$ is fixed with respect to t:

$$\max_t - \sum_{x \in \mathcal{X}} P_t(x|A) \log P_t(x|B)$$

Bayes rule to isolate posterior distribution

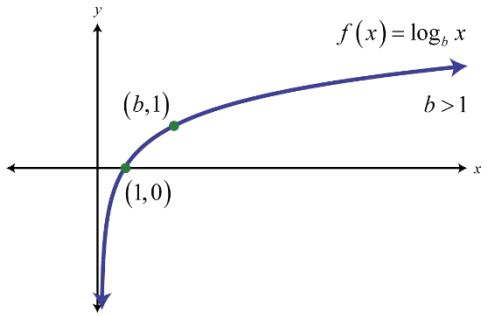
- Applying Bayes rule to $P_t(x|B)$ yields:

$$\max_t -\sum_{x\in\mathcal{X}} P_t(x|A)\log P_t(B|x) - \sum_{x\in\mathcal{X}} P_t(x|A)\log P(x) + \sum_{x\in\mathcal{X}} P_t(x|A)\log P(B)$$

- Simplifying:

$$\max_t -\sum_{x\in\mathcal{X}} P_t(x|A)\log P_t(B|x)$$

- Model posterior $P_t(B|x)$ as a classifier and $P_t(x|A)$ as an assumed set of labels



$$-\sum_{x\in\mathcal{X}} P_t(x|A)\log P_t(B|x) \approx \min -\frac{1}{T}\sum_{i=1}^{T} y_i \log(z_i)$$

Virtual classifiers summary

- Given:
  - a set of candidate change points $\{\tau_1, \tau_2, \ldots \tau_m\}$
  - a set of time series $\{X_i\}_{i=1}^{n}$
- Construct a set of binary labels $\{Y_j\}_{j=1}^{m}$
- Each Y_j is a vector of length T with:

$$Y_{jk} = \begin{cases} -1 \; if \; k \leq \tau_j \\ 1 \; if \; k > \tau_j \end{cases} \; for \; k = 1,2, \ldots T$$

- Copies of each of these label vectors Y_j are paired with every time series in $\{X_i\}_{i=1}^{n}$ forming the pseudo-labeled dataset $D_j = \{(X_i, Y_j)\}_{i=1}^{n}$.
- A classifier is trained on each dataset D_j, resulting in m classifiers each trained on a different labeling of the data.
- Accuracy on a validation set of each of the classifiers is measured as $p_1, p_2, \ldots p_m$.

Learn a predictor

1. Determine when the change point occurs in each time series X_i of the dataset $\{X_i\}_{i=1}^{n}$
2. Train a predictor, using the result of step 1, to predict the current state of the system given a sample from a time series
3. On a previously unseen time series X' generated by the same system, predict the change point prospectively.

Application – Seizure prediction

- Changes occur in the brain prior to seizure onset that make the seizure inevitable.
  - **Seizure prediction horizon (SPH), preictal state/period**
- Central question: When do the pre-seizure changes occur?

Learning the preictal period

- Use Change Point Detection (CPD) to determine preictal period
- Combine CPD with automatic feature extraction

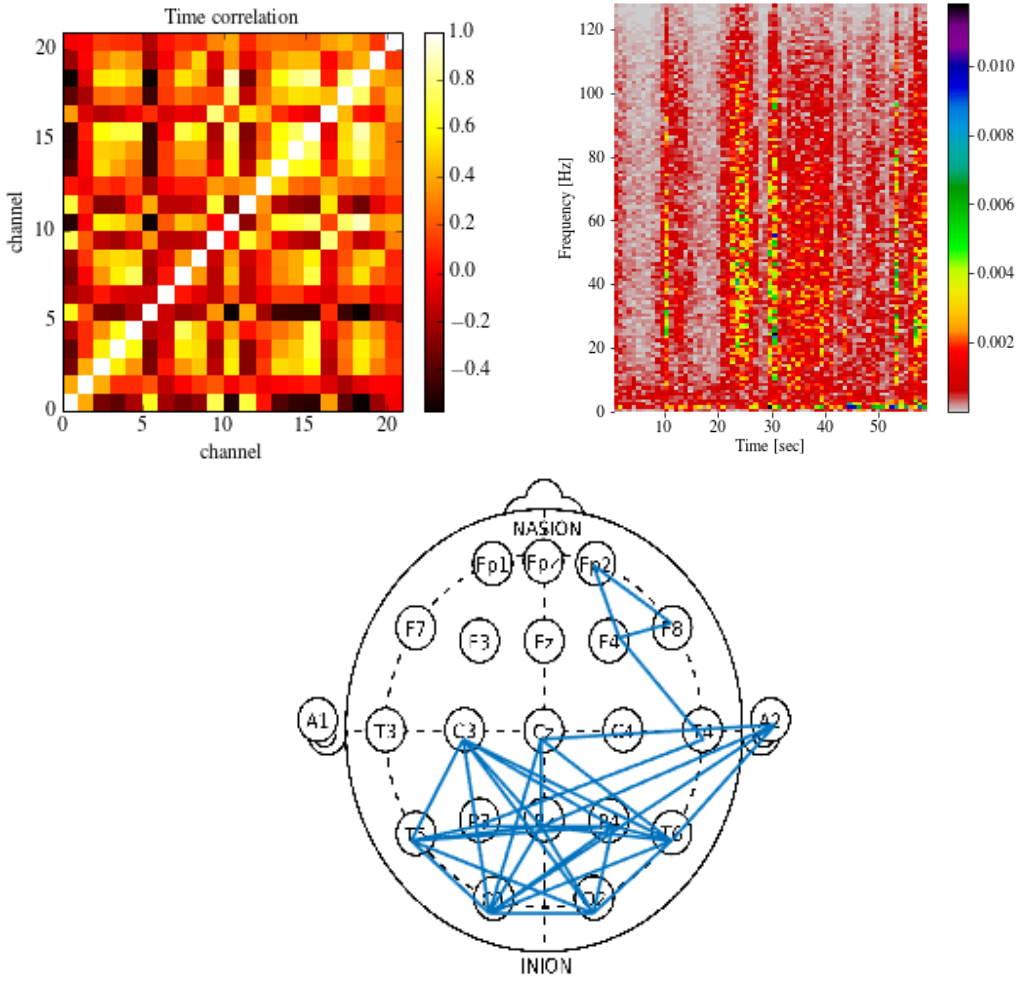


Figure 4. Examples of features extracted for seizure prediction.

CNN for feature extraction

- Use CNN (LeCun et al., 1998) as virtual classifiers to detect change point and learn features from EEG
- Convolutions over time and frequency domain via wavelets

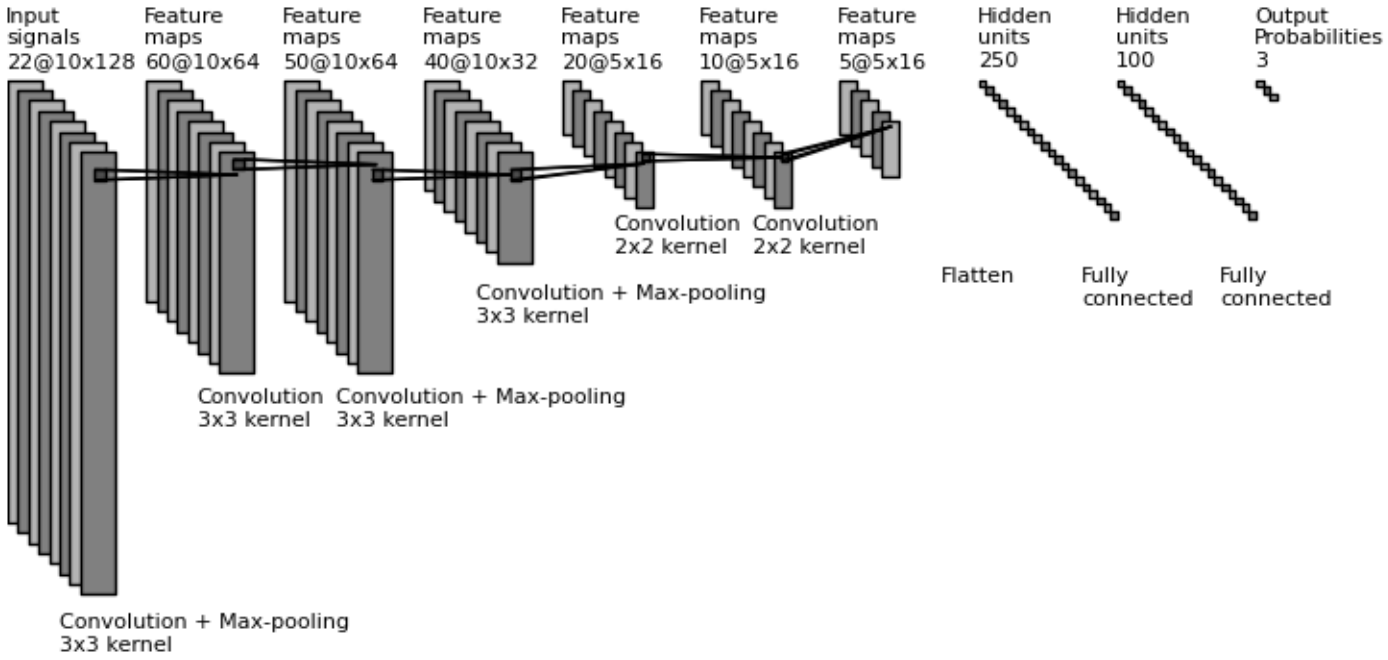
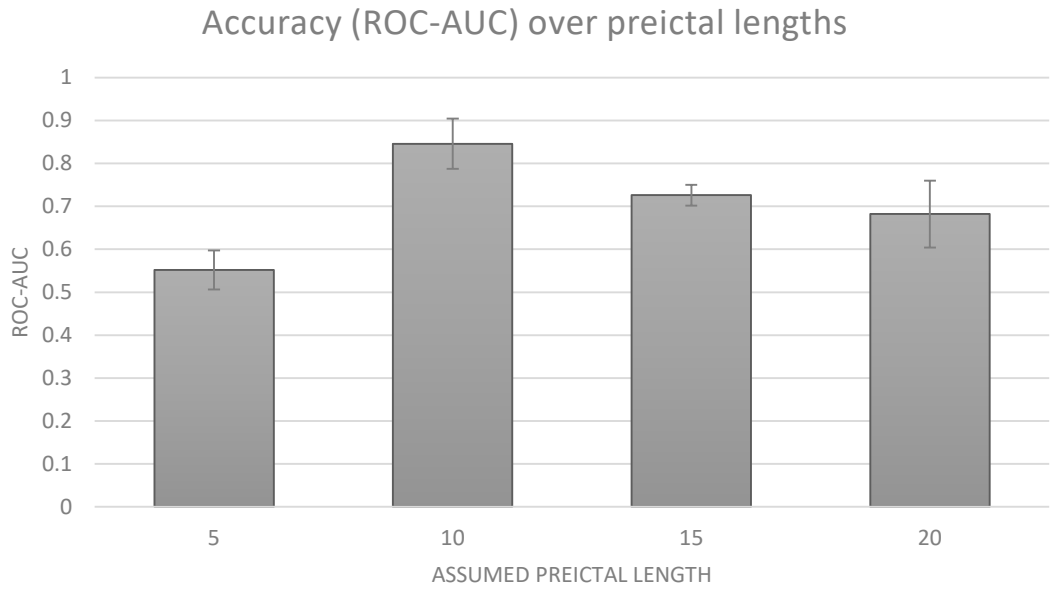


Figure 5. Convolutional neural network trained on EEG to predict brain states from wavelet transformed EEG.

VC for preictal period length

- Candidate preictal lengths were 5, 10, 15, and 20 mins
- A CNN was trained for each labelling of the data
- Preictal length of 10 mins was chosen based on significant improvement in accuracy

Table 1. ROC-AUC between interictal and preictal classes for different assumed preictal lengths. Averaged over 10-folds of validation data, error bar shows 1 standard deviation.



Results and Comparison

- We compared our results to:
  - 2 top performing algorithms from Kaggle (Brinkmann et al., 2016)
  - Cook group's algorithm (Cook et al., 2013)

Table 2. Seizure prediction results

Method	SPH (mins)	Sensitivity	FPr (FP/h)	Random pred. $\sigma_{low} - \sigma_{high}$
Kaggle1	60	72.7%	0.285	15.1% - 27.2%
Kaggle 2	60	75.8%	0.230	12.1% - 24.2%
Cook et al.	PS*	66.7%	0.186	12.1% - 21.2%
This work	10	**87.8%**	**0.142**	9.1% - 15.1%

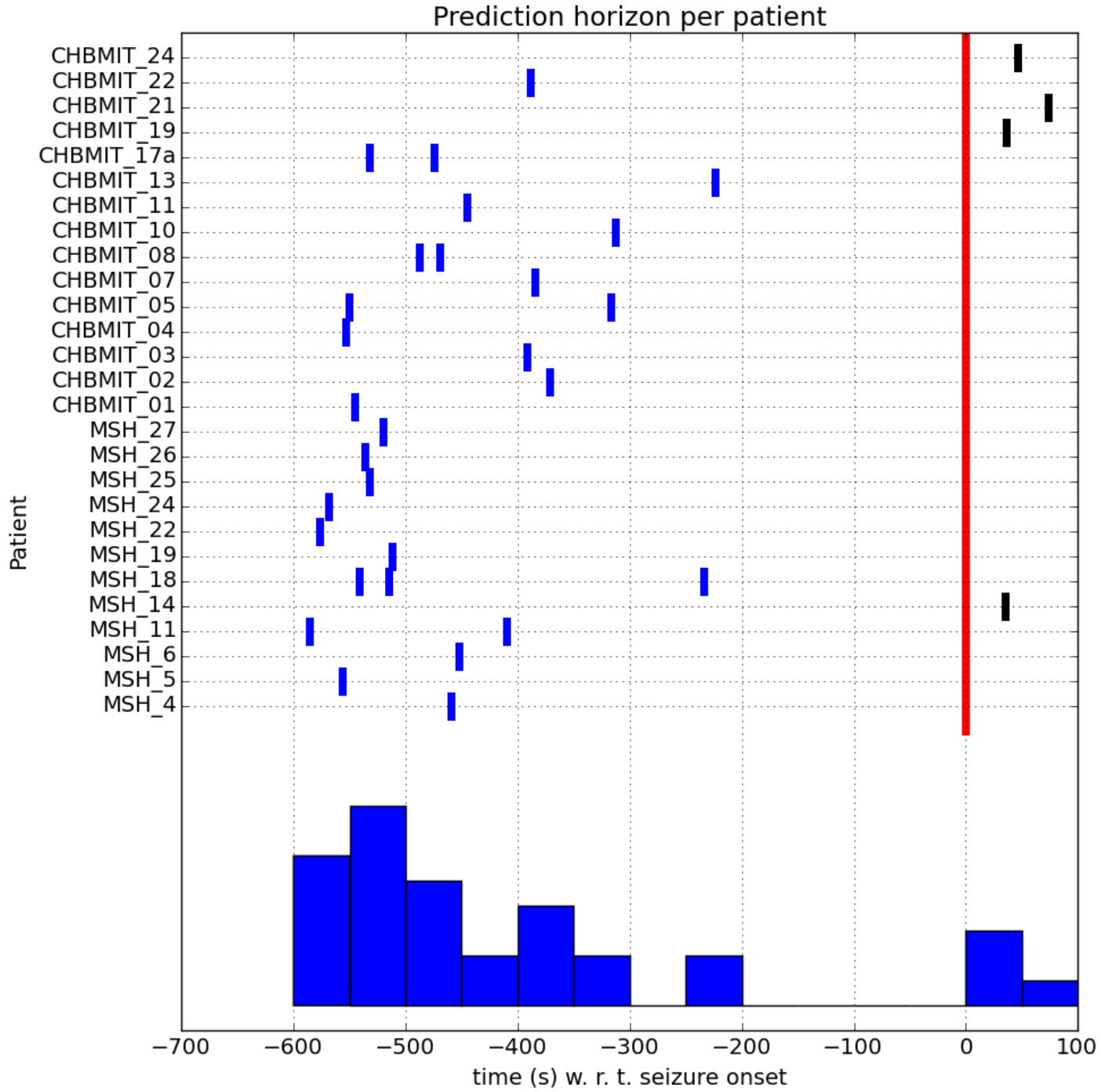


Figure 6. Prediction times generated by the CNN for all test set recordings with seizures grouped by patient. The spread of the prediction times is large indicating a non-uniform transition time within patients and between patients.

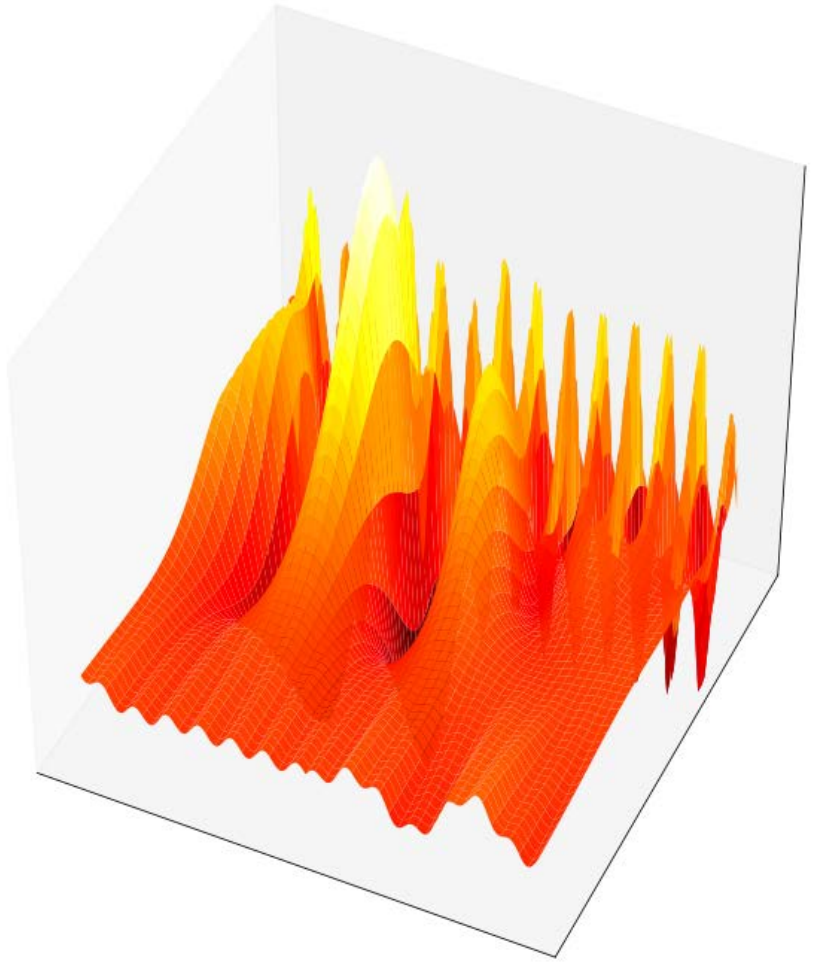
Conclusions

- VC requires a uniform transition time over all time series
  - Otherwise combinatorial explosion of m^n occurs
- VC is also very computationally expensive
  - Requires training m neural nets multiple times.
- Want a method that allows variable transition time between time series, while preserving benefits of VC.

Wavelet Deconvolutions

Spectral decompositions

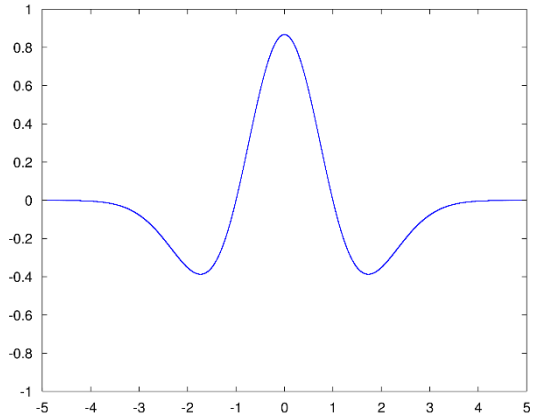
- Many models for time series use a spectral decomposition of the signals as input
  - Many parameters to pick
- Typically use cross validation to pick parameters
  - Can be time consuming and data hungry
- Used in applications such as automatic speech recognition (Hinton et al., 2012), biological signal analysis (Andreao et al., 2006), and financial time series (Cao et al., 2003)



Wavelet transform

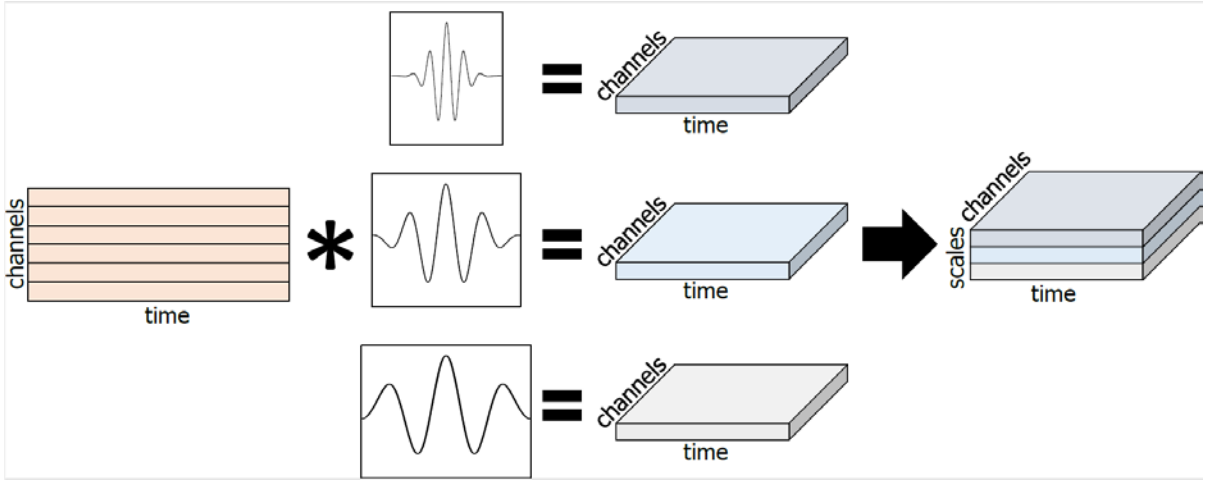
- The wavelet transform reveals spatial and spectral information (Daubechies, 1990)
- Scale the mother wavelet and convolve with the signal
- Reveals frequency content of the signal at that scale and each time point.

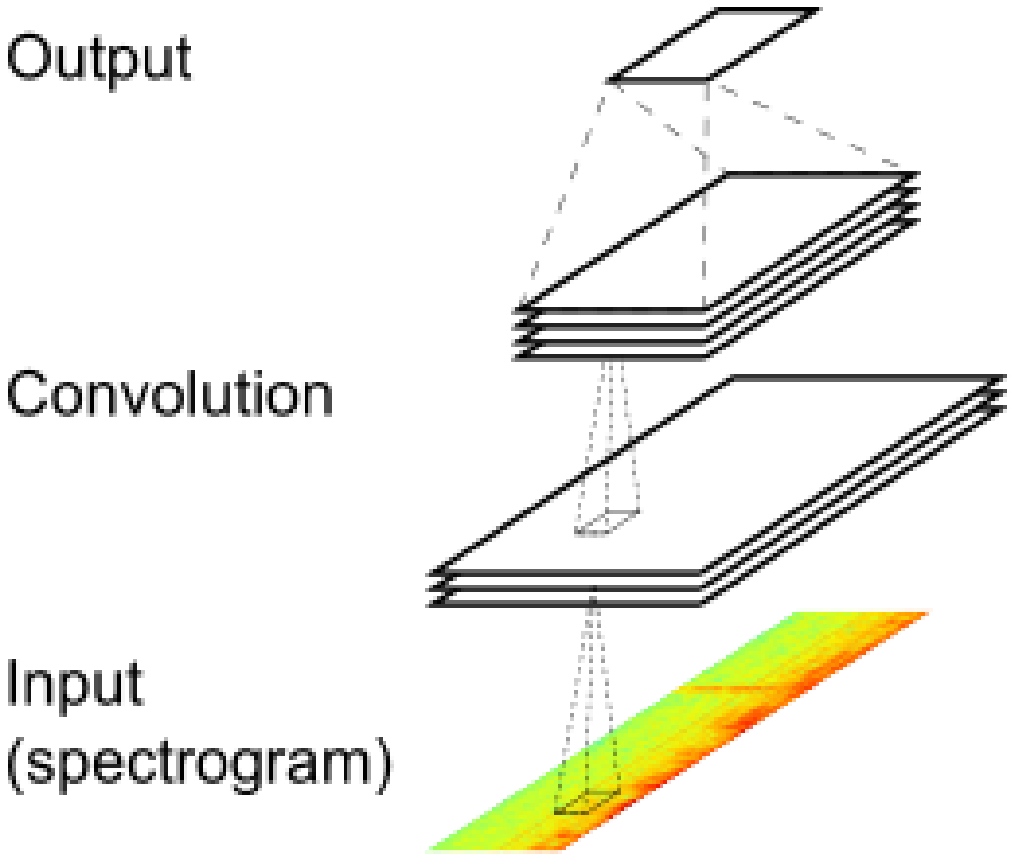
$$\Psi_w(t) = \frac{2}{\sqrt{3w}\pi^{\frac{1}{4}}}\left(1 - \frac{t^2}{w^2}\right)e^{-\frac{t^2}{2w^2}}$$



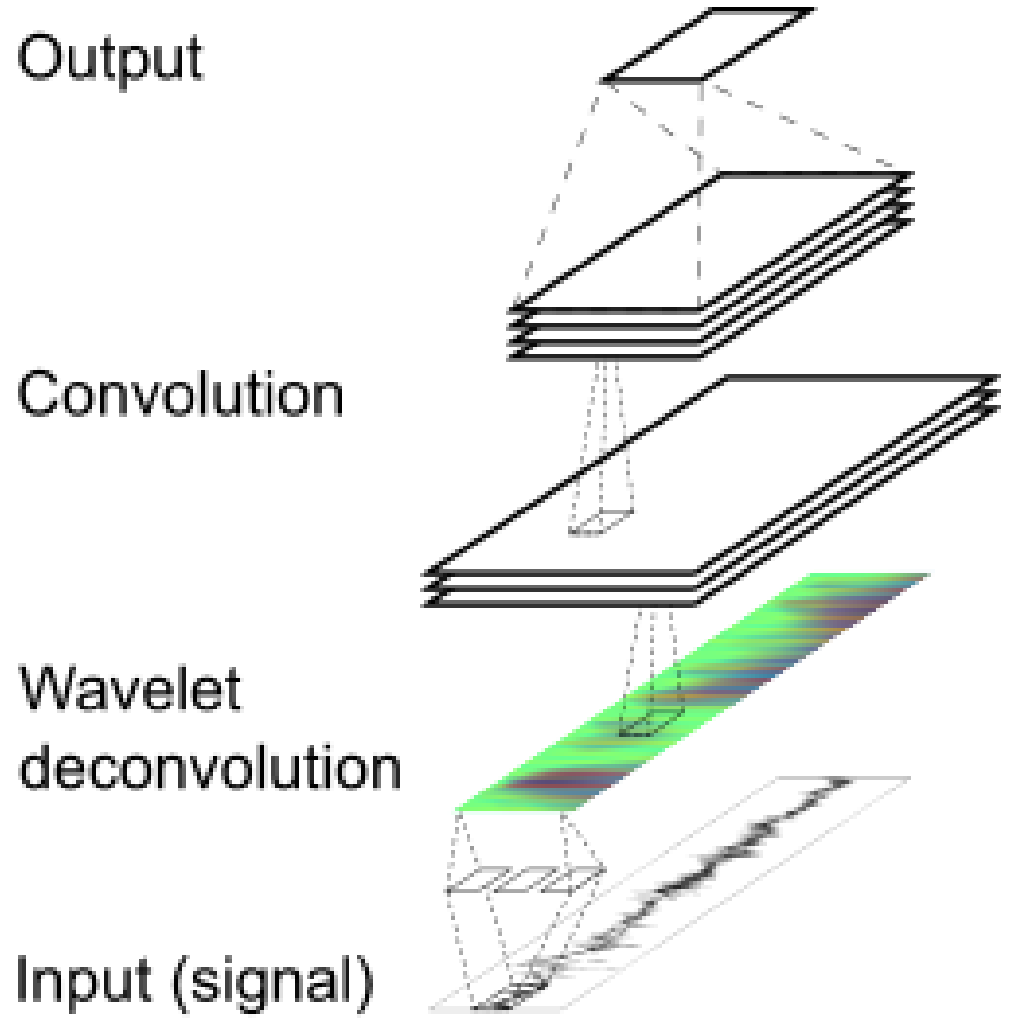
Automatically extracting time/freq domain features

- Combine the Wavelet Transform and CNN
- Use backpropagation to learn the scale parameters
- This enables learning the “width” of the kernel with gradient descent
  - CNN’s have fixed kernel sizes
- Also a reduction in the number of parameters




Output
Convolution
Input
(spectrogram)

(a)


Output
Convolution
Wavelet
deconvolution
Input (signal)

(b)

Learn scales with backpropagation

- Wavelet transform with learnable scales

The output of the wavelet layer is given by:

$$y_{ij} = \sum_{a=1}^{P} s_{ia}\, x_{j+a} \quad \forall i = 1 \dots M$$

Where the wavelet filter $s_i \in \mathbb{R}^{1xP}$ is the discretized wavelet function over the grid $k = \left\{-\frac{P-1}{2} \dots \frac{P-1}{2}\right\}$:

$$s_{ia} = \frac{2}{\sqrt{3w_i}\pi^{\frac{1}{4}}}\left(1 - \frac{k_a^2}{w_i^2}\right) e^{-\frac{k_a^2}{2w_i^2}} \quad \forall a = 1 \dots P$$

For backpropagation, we want $\frac{\delta E}{\delta w_i}$ where E is some error function:

$$\frac{\delta E}{\delta w_i} = \sum_{a=1}^{P} \frac{\delta E}{\delta s_{ia}} \frac{\delta s_{ia}}{\delta w_i} = \sum_{a=1}^{P} \frac{\delta E}{\delta s_{ia}} \left[A\left(M\frac{\delta G}{\delta w_i} + G\frac{\delta M}{\delta w_i}\right) + MG\frac{\delta A}{\delta w_i}\right]$$

$$\frac{\delta E}{\delta s_{ia}} = \sum_{j=1}^{N} \frac{\delta E}{\delta y_{ij}} \frac{\delta y_{ij}}{\delta s_{ia}} = \sum_{j=1}^{N} \frac{\delta E}{\delta y_{ij}} x_{j+a}$$

$$A = \frac{2}{\pi^{\frac{1}{4}}}(3w_i)^{-\frac{1}{3}}$$

$$M = 1 - \frac{k_a^2}{w_i^2}$$

$$G = e^{-\frac{k_a^2}{2w_i^2}}$$

$$\frac{\delta A}{\delta w_i} = -\frac{6}{\pi^{\frac{1}{4}}}(3w_i)^{-\frac{3}{2}}$$

$$\frac{\delta M}{\delta w_i} = \frac{2k_a^2}{w_i^3}$$

$$\frac{\delta G}{\delta w_i} = \frac{k_a^2}{w_i^3} e^{-\frac{k_a^2}{w_i^2}}$$

$w_i > 0$

Results

- TIMIT Phone recognition dataset
- UCR Haptics dataset

Table 3. Best reported PER on the Timit dataset without context dependence

Method	PER (Phone Error Rate)
DNN with ReLU units [96]	20.8
DNN + RNN [110]	18.8
CNN [97]	18.9
WD + CNN (this work)	18.1
LSTM RNN [111]	17.7
Hierarchical CNN [97]	**16.5**

Table 4. Test error on the Haptics dataset

Method	Test Error
DTW [113]	0.623
BOSS [114]	0.536
ResNet [105]	0.495
COTE [115]	0.488
FCN [105]	0.449
WD + CNN (this work)	**0.425**