Peak Detection and Baseline Correction using a Convolution Neural Network

Mikkel N. Schmidt, **Tommy S. Alstrøm**, Marcus Svendstorp and Jan Larsen

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science

In memory of Prof. Jan Larsen, 1965–2018

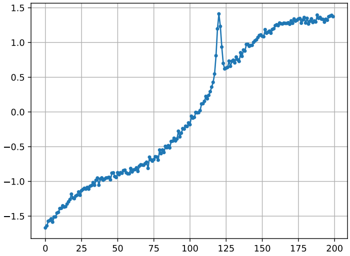
Honory special session Friday 8.30–10.30, organized by
Professor Tülay Adali, University of Maryland Baltimore
Professor Zheng-Hua Tan, Aalborg University

Abstact

- *Peak detection* and *baseline suppression* in a noisy signal with an unknown baseline.
- In practical applications, one of the most *successful* approaches to *joint* baseline suppression and peak localization is based on the *continuous wavelet transform*.
- Reformulate this as a *convolutional neural network*.
- Demonstrate that with sufficient training data, the approach consistently *compares* to (and often outperforms) the *optimized continuous wavelet method*.

Peak detection problem

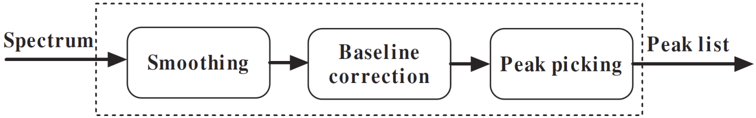
- Peak finding – detect the existence of peak and locate the position.
- Baseline suppression – carry out this task robustly in the presence of a baseline.



Background

Study by Yang et al[1] – compares:

- 7 smoothing methods.
- 5 baseline correction methods.
- 8 peak finding criterions.



- Alternative – joint baseline correction and peak detection/localization.
- "*Results show that CWT provides the best performance*".

[1] Chao Yang, Zengyou He, and Weichuan Yu. "Comparison of public peak detection algorithms for MALDI mass spectrometry data analysis". In: BMC Bioinformatics 10 (2009). DOI: 10.1186/1471-2105-10-4.

Signal model

$$s(f) = b(f) + v(f - f_0) + s \cdot e(f)$$

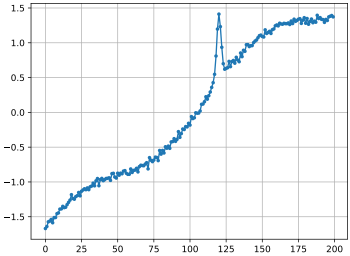
where

$s(f)$: Measured spectrum

$b(f)$: Baseline

$v(f)$: Peak line-shape

$e(f)$: i.i.d Gaussian noise



Continuous wavelet peak localization

The mexican hat wavelet

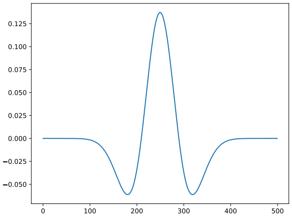
$$\psi_a(f) = \frac{2}{\sqrt{3a}\pi^{1/4}}\left(1 - \frac{f^2}{a^2}\right)\exp\left(-\frac{f^2}{2a^2}\right)$$

Write as a convolutional sum

and pick $c[j]$

$$c[j] = \sum_{f=1}^{W} s[f+j]\psi_a[f]$$

Mexican hat example:



Suppressing the baseline

The continuous wavelet peak localization scheme suppresses a locally smooth baseline, i.e. baseline is modelled as constant plus an odd signal:

$$b(f) = \delta + g(f), g(f) = g(-f)$$

The convolution with the baseline then vanishes:

$$(b * \psi)(f) = \int_{-\infty}^{\infty} b(f')\psi_a(f' - f)\mathrm{d}f' = 0$$

This is due to the CW begin a zero–mean symmetric function.

As a convolution network

1–d convolutional layer:

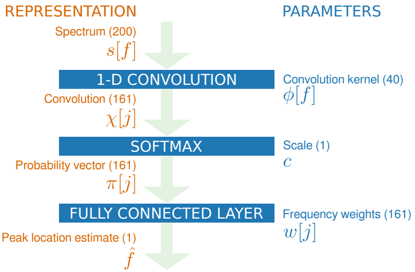
$$\chi[j] = \sum_{f=1}^{W} s[f+j]\phi[f]$$

Softmax layer:

$$\pi[j] = \frac{\exp(c \cdot \chi[j])}{\sum_{k=0}^{F-W} \exp(c \cdot \chi[k])}$$

Linear readout layer:

$$\hat{f} = \sum_{j=0}^{F-W} \pi[j] w[j]$$

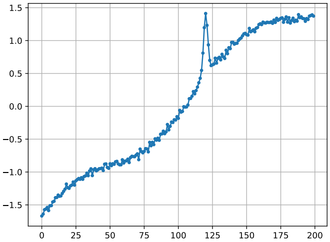


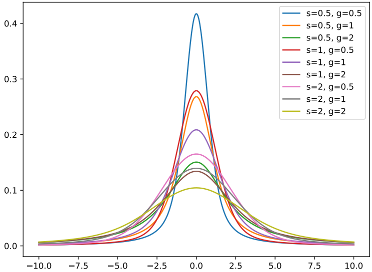
Formulation enables *end–to–end learning*

Data generation

Generate spectra according to our model:
$s(f) = b(f) + v(f - f_0) + s \cdot e(f)$

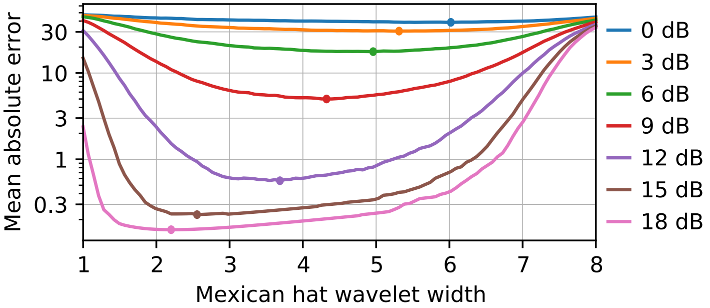
1. Baseline is generated using smoothed Gaussian random walk.
2. Add Voigt shaped peak:
   $v(f) = \frac{1}{\sigma\sqrt{2\pi}} \mathrm{Re}\left[w\left(\frac{f+i\gamma}{\sigma\sqrt{2}}\right)\right]$
3. Add i.i.d Gaussian noise.





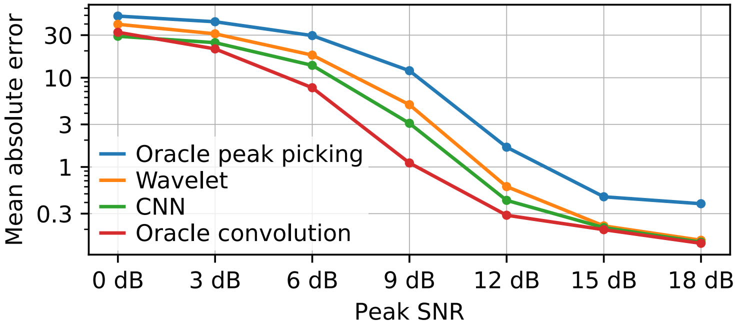
Mexican hat wavelet width

$$\psi_a(f) = \frac{2}{\sqrt{3a}\pi^{1/4}} \left(1 - \frac{f^2}{a^2}\right) \exp\left(-\frac{f^2}{2a^2}\right)$$



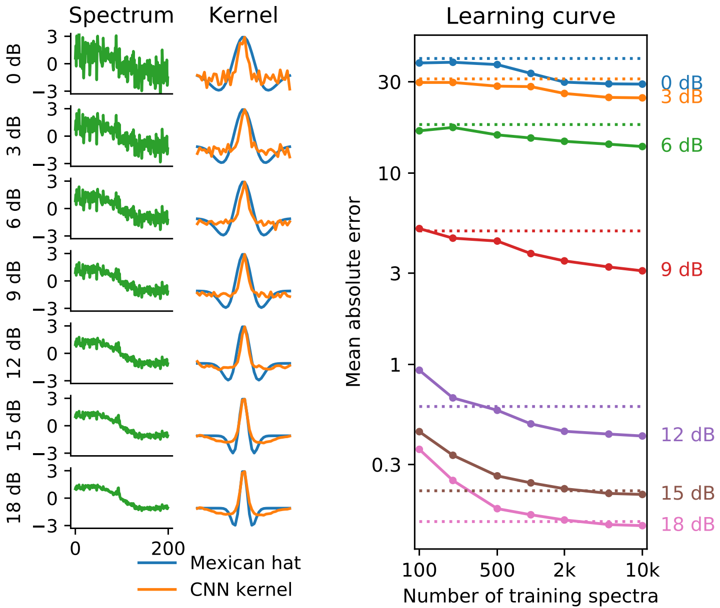
$$\text{Error} = \sum_{i=1}^{N} |f_i - \hat{f}_i|$$

Peak localization



Oracle peak picking: No baseline – pick maximum value.
Oracle convolution: No baseline – convolve with true peak lineshape – pick maximum value.

Learning curves and learned kernels



Limitations and possible extensions

Limitations:

- Spectral peak shape assumed constant.
- Peak signal–to–noise ratio was held constant in any given training.
- It was assumed that a single peak always exists.

Possible extensions:

- Have multiple peak location estimators and endow them with an attention mechanism so that each estimator will focus on a sub-range of frequencies.

Conclusions

The CNN approach to peak localization shows great promise, as it can more *efficiently leverage data* to outperform the current state of the art, and can readily be extended and *incorporated as a module in a larger neural network architecture*.

Thank you