Robust Deep Residual Shrinkage Networks for Online Fault Classification

Alireza Salimy*, Imene Mitiche*, Philip Boreham†
Alan Nesbitt* and Gordon Morison*
* School of Computing, Engineering and Built Environment
Glasgow Caledonian University
Glasgow, United Kingdom
† Innovation Centre for Online Systems
Doble Engineering
Bere Regis, United Kingdom

***Abstract*—In this paper, a novel approach to improve signal classification in the presence of noise is presented. Using Stockwell transforms for feature extraction on time-series electromagnetic interference data and deep residual neural networks, containing thresholding functions (shrinkage functions) as non-linear transformation layers for classification. Thresholding functions are commonly used for signal de-noising. Setting thresholds for optimal functionality is often complex and requires expertise, this paper will investigate learned methods of threshold selection along with alternate thresholding functions. Using deep learning methods to select thresholds reduces the dependency on experts for the use of thresholding functions for de-noising and allows for adaptation to alternate noise environments. This paper proposed the novel application of two different threshold functions and introduces an architecture update for learning the threshold parameters for classification in the presence of noise. Several experiments are carried out to compare the performance of the systems with varying signal-to-noise ratio data sets taken from real-world operational high-voltage assets. Experimental results show that the proposed approaches using both Garrote and Firm thresholding achieved improved performance increases over utilizing soft thresholding within deep shrinkage networks in low signal-to-noise ratios.**

I. Introduction

High-voltage (HV) and mechanical equipment used in power generation are prone to faults, if incurred these faults can lead to major losses such as; health and safety hazards, fines, legal issues, and possibly large-scale power outages [1]. To avoid such losses, condition monitoring is carried out on crucial HV assets. Condition monitoring allows early detection of arising faults and quick correction, currently condition monitoring is carried out manually by experts [2]. They observe electromagnetic interference (EMI) data in many forms to classify the faults present. The EMI method is commonly used to detect partial-discharge (PD) in HV systems [3]. The dependency on experts to carry out this essential condition monitoring has many downfalls. If experts are not available condition monitoring cannot be carried out and faults could go unnoticed, allowing them to become malfunctions. If the condition monitoring of HV assets is automated it can be used in a continuous nature preventing faults from going unnoticed, it also provides the ability for condition monitoring to be carried out by concerned parties without the need for expertise, reducing the dependency on experts. This research will observe several proposed autonomous fault classification systems, focusing on thresholding functions used within the systems and their performance in the presence of various levels of noise. Thresholding or Shrinkage functions are implemented in the field of Signal Processing to alter values in ranges corresponding to noise, thresholds are often selected by signal processing experts with respect to the signal data they observe. Previous work in the condition monitoring of mechanical power generation assets using vibration signals implemented Soft thresholding with learned thresholds [4], this study will build upon this work using the Soft thresholding method along with several others to observe which thresholding method is beneficial in the case of EMI fault classification. Learned thresholds implemented in this research allow for thresholding functions to be used without the expert insight into the data at hand. This research will produce systems to ingest data in the form of time-frequency decomposition matrices and produce a fault classification of the input data. The time-frequency decomposition used in this study is the Stockwell (S) transform proposed in [5]. The machine learning (ML) system used in the study to produce fault classifications from the time-frequency decomposition's of signals is based upon the residual neural network (ResNet) an architecture built for image recognition [6], with residual shrinkage blocks, proposed in [4], implemented to carry out thresholding with learned threshold values. The data-set used in this study consisted of EMI fault signals of 7 various classes, collected from real-world operational HV assets and 4 shrinkage functions were observed in the study with 3 containing a single learned thresholding parameter and 1 containing two learned thresholding parameters.

This paper will outline the methods used in this research in the following structure: **Section II**- Introduces the S transform and explores its derivation, **Section III**- Explores the various thresholding functions used in the experiments throughout this research and outlines their learned parameters, **Section IV**- This section will introduce the various models used to implement the thresholding functions and their learned

Thanks to Doble Engineering for supporting this study.

parameters, **Section V**- Explaining the experimental procedure followed through the research, outlining the data used and the results obtained, **Section VI**- Discusses and concludes the research, providing some insight into future work.

II. Stockwell Transform

The S transform is proposed to build upon the continuous wavelet transform (CWT) proposed in [7], the S transform is implemented to produce a time-frequency decomposition retaining frequency dependant resolution from the original time-series data. These characteristics of the S transform prove to be desirable in regards to EMI data taken from various real-world assets, due to the non-stationary characteristics of the data.

A. Derivation

The S transform used in this paper is derived by finding the "phase correction" of the CWT as recommended by [5]. First, the CWT of a signal $h(t)$ is found using (1).

$$W(\tau, d) = \int_{-\infty}^{\infty} h(t)u(t-\tau, d)dt \tag{1}$$

Where $W(\tau, d)$ the CWT of $h(t)$, is a two-dimensional function in the time-frequency plane (τ, d) with d representing the dilation term controlling the resolution by determining the width of the wavelet and τ representing the height of the wavelet and $u(t, d)$ representing a scaled replica of the mother wavelet. A constraint is placed on $u(t, d)$ requiring it to have zero mean to produce a CWT. The mother wavelet is defined in (2).

$$u(t, f) = \frac{|f|}{\sqrt{2\pi}} e^{-\frac{t^2 f^2}{2}} e^{-j2\pi f t} \tag{2}$$

$W(\tau, d)$ with specific mother wavelet is then used to find the S transform of the function $h(t)$ by multiplication with the phase factor $e^{j2\pi f\tau}$, with f representing frequency, this relationship is outlined in (3).

$$S(\tau, f) = e^{j2\pi f\tau} W(\tau, d) \tag{3}$$

Written explicitly the S transform of $h(t)$ is:

$$S(\tau, f) = \int_{-\infty}^{\infty} h(t) \frac{|f|}{\sqrt{2\pi}} e^{-\frac{t^2 f^2}{2}} e^{-j2\pi f t} dt \tag{4}$$

If the S transform is shown to be a representation of the local spectrum, averaging the local spectra over time produces the Fourier spectrum. This is shown in (5), where $H(f)$ represents the Fourier spectrum of $h(t)$.

$$\int_{-\infty}^{\infty} S(\tau, f) d\tau = H(f) \tag{5}$$

Thus showing that the S transform can be written as operations on the Fourier spectrum, when $f \neq 0$:

$$S(\tau, f) = \int_{-\infty}^{\infty} H(\eta + f) e^{-\frac{2\pi^2 \eta^2}{f^2}} e^{j2\pi\eta\tau} d\eta \tag{6}$$

B. Discrete Stockwell Transform

The discrete S transform is found by taking the discrete analog of (6). Letting $h[n]$ denote a discrete time-series and $H[k]$ the discrete Fourier transform of this signal, found by (7).

$$H[k] = \sum_{n=0}^{N-1} h[n] e^{-\frac{j2\pi nk}{N}} \tag{7}$$

Using (6) and (7), the S transform of a discrete time-series signal can be found, outlined in (8), when $n \neq 0$.

$$S[\rho, n] = \sum_{m=0}^{N-1} H[m+n] e^{-\frac{2\pi^2 m^2}{n^2}} e^{\frac{j2\pi m\rho}{N}} \tag{8}$$

Where $\rho = m = n = 0, 1, ..., N-1$, (8) is the premise of the S transforms used in the research however, in the case of $n = 0$ the S transform is found by (9). Further processing is carried out on the produced transformations to convert the imaginary product of (8) to a modulus representation by taking the absolute value of the imaginary product.

$$S[\rho, 0] = \frac{1}{N} \sum_{m=0}^{N-1} h[m] \tag{9}$$

III. Thresholding Methods

Thresholding functions are used for signal de-noising, as observed in [8] [9]. Thresholding is a process that observes values within given ranges and alters them to determined values, for example in Soft thresholding proposed by [10], values close to zero i.e. within the given thresholds are deemed unimportant and assigned zero, values deemed not close enough to zero i.e. outwith the thresholds given are given non-zero values. The thresholds determining values to be altered or negated are often chosen by Signal Processing experts, creating artificial filters to denoise signals. Deep learning allows thresholds to be chosen through gradient descent, this allows for filters to be produced relevant to the data being observed. Providing an optimised filter intuitively should lead to systems with greater confidence in classification and give systems the upper hand in classifying data from high noise environments.

The experiments carried out in this research will observe various methods of thresholding; Soft thresholding and Hard thresholding proposed by [10], Firm thresholding introduced by [11] and Garrote thresholding from [12]. All of the thresholds required for the alternative forms of thresholding functions will be learned using deep learning. Graphical representations of all thresholding methods implemented are found in Fig. 1.

A. Soft Thresholding

The Soft thresholding function is outlined in (10), values within given thresholds are set to zero, and values outwith this range are converted to their original and the threshold value γ is subtracted from them.

$$\delta(w) = \begin{cases} (|w| - \gamma) \cdot sgn(w), & |w| \geq \gamma \\ 0, & |w| < \gamma \end{cases} \tag{10}$$

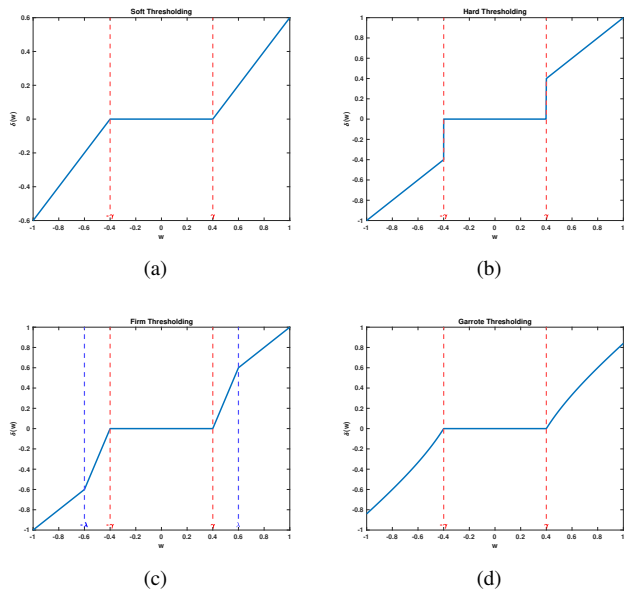

Fig. 1. Graphical representation of thresholding methods (a) - Soft thresholding, (b) - Hard thresholding, (c) - Firm thresholding, (d) - Garrote thresholding.

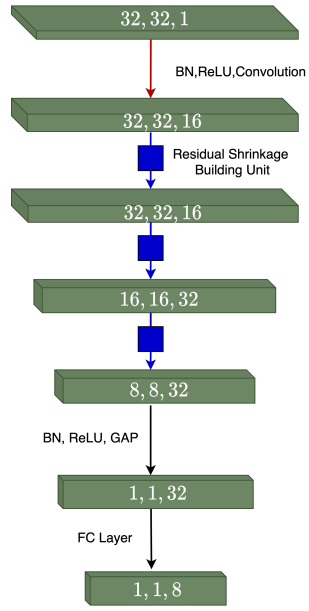

Fig. 2. Overall deep residual shrinkage network.

B. Hard Thresholding

Hard thresholding shown in (11), assigns values within a given threshold to zero and original values are retained if the value is outwith the threshold γ range.

$$\delta(w) = \begin{cases} w, & |w| \geq \gamma \\ 0, & |w| < \gamma \end{cases} \tag{11}$$

C. Firm Thresholding

Firm thresholding is the only thresholding method in this study to contain two thresholding parameters, shown in (12), as γ and λ. Values within a given threshold are set to zero much like the previously mentioned thresholding methods, although values are altered depending on their location in respect to the thresholds.

$$\delta(w) = \begin{cases} 0, & |w| \leq \gamma \\ \frac{\lambda(|w|-\gamma)}{\lambda-\gamma} \cdot sgn(w), & \gamma \leq |w| \leq \lambda \\ w, & |w| \geq \lambda \end{cases} \tag{12}$$

D. Garrote Thresholding

Garrote thresholding was originally introduced to overcome the downfalls of Soft and Hard thresholding, similar to these methods, Garrote assigns values below a given threshold to zero. However, the major difference in Garrote thresholding arises from the non-linear assignment of values outwith the thresholds as observed in (13).

$$\delta(w) = \begin{cases} 0, & |w| < \gamma \\ w - \frac{\gamma^2}{w}, & |w| \geq \gamma \end{cases} \tag{13}$$

IV. Model Architecture

The models used in the experiments throughout this paper are based on the channel-wise deep residual shrinkage network (DRSN) proposed in [4], implementing a deep ResNet architecture with residual shrinkage building units (RSBU). ResNets, developed in [6], are a variation of the standard convolutional neural network (CNN) and have become very popular in tackling image classification problems and have been found to produce state-of-the-art results in this field. The main variation of ResNets from standard CNN's are identity skip connections, they are implemented to avoid the exploding or vanishing gradient problem, in turn reducing training error and loss.

The contributions in this paper build upon the DRSN architecture recommended by [4], by introducing further learned thresholding functions and increasing the size of the architecture, by adding to the RSBU, to learn these additional thresholds and implementing alternative methods of thresholding to observe the results they obtain. The overall DRSN architecture used in this study can be found in Fig. 2.

The work in this research deploys the use of two alternate RSBUs; RSBU-1 representing the architecture for learning a single threshold parameter and RSBU-2 representing the architecture for learning two threshold parameters. RSBUs are stacked in the overall architecture to gradually reduce noise-related features.

A. RSBU-1

The RSBU-1 architecture, shown in Fig. 3 is designed to apply an individual threshold to each channel in the feature map, this architecture is based on finding a single thresholding parameter. It can be seen that the threshold value γ is calculated by following several steps; first, the feature map is

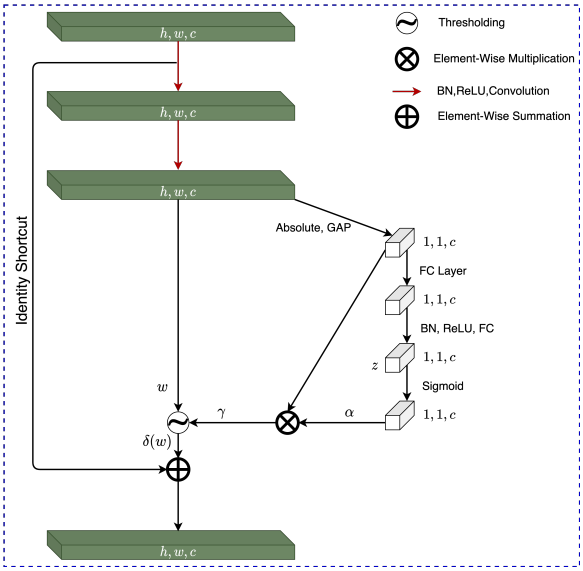

Fig. 3. Residual shrinkage building unit-1 architecture.

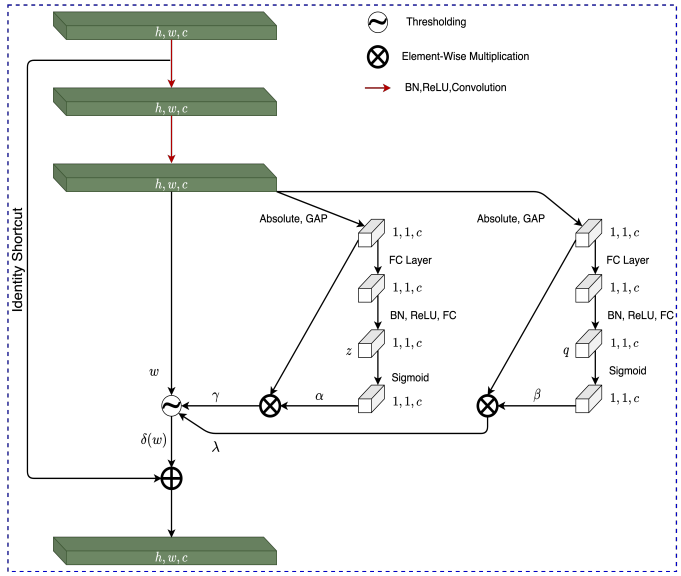

Fig. 4. Residual shrinkage building unit-2 architecture.

reduced to a 1-D vector using global-average-pooling (GAP) and then taking the absolute value of this result. This 1-D vector is then propagated to a two-layer fully connected (FC) network, the output of this FC network α_c is scaled to the range $(0, 1)$ using (14) portrayed as the Sigmoid layer in Fig. 3. Where $z(c)$ represents the feature of the cth neuron and $\alpha(c)$ represents the cth scaling parameter.

$$\alpha(c) = \frac{1}{1 + e^{-z(c)}} \tag{14}$$

$\alpha(c)$ is then used to calculate the threshold of the cth channel of the feature map $\gamma(c)$, using (15), with h, w and c corresponding to the indexes of height, width and channels of the feature map w.

$$\gamma(c) = \alpha(c) \cdot \underset{(h,w)}{E}[w(h, w, c)] \tag{15}$$

The calculated threshold value for all channels γ is then used in the relevant thresholding method to produce a feature map that has underwent thresholding, $\delta(w)$. RSBU-1 was used for Soft, Hard and Garrote thresholding.

B. RSBU-2

The requirement for a second thresholding parameter λ led to the use of RSBU-2, much like RSBU-1 this architecture applies a threshold value to each channel in the feature map. However, RSBU-2 is used to find two threshold parameters γ as before and a second parameter λ, the RSBU-2 architecture is shown in Fig. 4. RSBU-2 follows the same procedure as RSBU-1 to find the thresholding parameter γ from (15), although a second thresholding parameter λ is also found in RSBU-2. This is done by reducing the feature map into a 1-D vector using GAP and taking the absolute value of the result. The produced 1-D vector is propagated to a two-layer FC network, the output of this layer is also scaled using (16), where $\beta(c)$ represents the cth scaling parameter of the fully connected network after scaling and $q(c)$ represents the feature of the cth neuron.

$$\beta(c) = \frac{1}{1 + e^{-q(c)}} \tag{16}$$

The second threshold value of the cth channel $\lambda(c)$ is then calculated by finding the dot product of the scaling parameter and the global average, of the cth channel, shown in (17).

$$\lambda(c) = \beta(c) \cdot \underset{(h,w)}{E}[w(h, w, c)] \tag{17}$$

Both calculated threshold values γ and λ are then used in the relevant thresholding method to produce a thresholded feature map $\delta(w)$. RSBU-2 was used for Firm thresholding.

V. Experimental Set-up

A. Data-set

The EMI data used in the experiments in this study were collected following the Committee International Special des Perturbations Radioelectriques (CISPR) 16 standard [13] using the EMI technique from [14]. Data was collected in the form of time-resolved signals from operational real-world assets sampled at 24000 samples per second, collected signals were then analysed by EMI experts and labelled accordingly, with the present faults, using their experience and knowledge gained from previous fault diagnosis. The fault classes observed in the span of the data were; Arcing, Data-Modulation, Partial Discharge, Processing Noise, Random Noise, Exciter, and Micro-sparking. The data sets used in the experiments were balanced and contained 261 samples per class, with each signal example containing 4000 sample points. The signals in the data-set underwent further pre-processing to produce various signal data sets with known noise levels by de-noising the raw signals using a symlet4 wavelet with a posterior median threshold rule from [15], carried out using the wavelet de-noising (wdenoise) function from Matlab, noise variance was

TABLE I
PERFORMANCE COMPARISON OF THRESHOLDING MODELS WITH ALTERNATE NOISE LEVEL DATA. BEST PERFORMANCE IS PRESENTED IN **BOLD** FONT.

Data-set dB SNR	Soft	Hard	Garrote	Firm
-5	**59.31** %	54.67 %	57.45 %	59.23 %
-4	56.75 %	55.99 %	56.20 %	**57.41** %
-3	57.81 %	57.99 %	**59.01** %	58.94 %
-2	64.34 %	64.49 %	**66.64** %	65.22 %
-1	66.17 %	67.92 %	67.52 %	**69.23** %
0	72.30 %	71.82 %	73.98 %	**74.60** %
1	**76.16** %	72.96 %	71.06 %	73.10 %
2	75.55 %	73.58 %	75.62 %	**76.20** %
3	83.39 %	80.77 %	**84.27** %	82.77 %
4	**83.36** %	81.82 %	83.32 %	82.52 %
5	86.06 %	84.89 %	**87.77** %	83.94 %

estimated based on the highest-resolution wavelet coefficients. Random noise was then added at desired levels, 11 data sets were obtained; -5, -4, -3, -2, -1, 0, 1, 2, 3, 4 and 5 dB signal-to-noise ratio's (SNR's). The data underwent further splitting to produce 3 sub-sets per data-set for training, validation, and testing, these subsets contained 70%, 15% and 15% of entities of the overall data-set.

B. Fault classification using DRSN

The modulus of the Stockwell transform of each signal was calculated, producing 261 2-D time-frequency mappings for every fault class in the data-set. The produced Stockwell entities along with their relevant labels were used to train, validate and test the various models observed in the research. The 4 thresholding methods, outlined in Section III, were all trained, validated, and tested for 10 runs, using the various noise level data sets created producing results for 4 different thresholding models with 7 data sets. Thresholding methods were compared based on their mean test accuracy over these 10 runs. Models were implemented using Tensorflow [16] as multi-class classifiers, categorical cross-entropy loss was used for training all thresholding models with the momentum optimiser being used over 250 epochs.

C. Results

The mean test accuracies produced by each thresholding model concerning the data-set used are outlined in Table I, accuracies are produced using binary accuracy, this being the division of the total number of correct predictions by the number of test samples. It can be seen from the results that the Garrote and Firm thresholding methods provide high accuracies and confident results close to the highest accuracies in low SNR data sets.

VI. CONCLUSION

Our developed systems demonstrated the benefits of implementing alternate thresholding methods on various noise level data sets, it was shown that the implementation of Garrote and Firm thresholding methods produced improved results in low SNR cases in comparison to both Soft and Hard thresholding based approaches. Outlining the benefits learned thresholding parameters can have when observing noisy data and showing that thresholding methods with two learned parameters, found in the RSBU-2 architecture can also improve classification performance. Further work will be carried out to implement further thresholding methods and observing how these will affect the classification of noisy data.

REFERENCES

[1] S. Barrios, D. Buldain, M. Comech, I. Gilbert and I. Orue Sagarduy, "Partial discharge classification using deep learning methods-survey of recent progress," Energies, vol. 12, pp. 2485, 2019.

[2] J. E. Timperley and J. M. Vallejo, "Condition assessment of electrical apparatus with emi diagnostics," IEEE Petroleum and Chemical Industry Committee Conference (PCIC), pp. 1–8, 2015.

[3] J. Timperley, D. Buchanan and J. Vallejo, "Electric generation condition assessment with electromagnetic interference analysis," IEEE Transactions on Industry Applications, vol. 54, pp. 1921–1929, 2017.

[4] M. Zhao, S. Zhong, X. FU, B. Tang and M. Pecht, "Deep Residual Shrinkage Networks for Fault Diagnosis," IEEE Transactions on Industrial Informatics, vol. 16, pp. 4681-4690, 2020.

[5] R. G. Stockwell, L. Mansinha and R. P. Lowe, "Localization of the complex spectrum: the S transform," IEEE Transactions on Signal Processing, vol. 44, pp. 998-1001, 1996.

[6] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, 2016.

[7] P. Goupillaud, A. Grossmann and J. Morlet, "Cycle-octave and related transforms in seismic signal analysis," Geoexploration, vol. 23, pp. 85-102, 1984.

[8] D. L. Donoho, "De-noising by soft-thresholding," IEEE Transactions on Information Theory, vol. 41, pp. 613–627, 1995.

[9] K. Isogawa, T. Ida, T. Shiodera and T. Takeguchi, "Deep shrinkage convolutional neural network for adaptive noise reduction," IEEE Signal Processing Letters, vol. 25, pp. 224–228, 2018.

[10] D. L. Donoho and I. M. Johnstone, "Ideal Spatial Adaptation via Wavelet Shrinkage," Biometrika, vol. 81, pp. 425-455, 1994.

[11] H. Gao and A. Bruce, "Waveshrink with Firm Shrinkage," Statistica Sinica, pp. 855-874, 1997.

[12] H. Gao, "Wavelet Shrinkage Denoising Using the Non-Negative Garrote," Journal of Computational and Graphical Statistics, vol. 7, pp. 469-488, 1998.

[13] CISPR/CIS/A - Radio-interference measurements and statistical methods, EMC 33.100.10 - Emission, 2015.

[14] J. E. Timperley and J. M. Vallejo, "Condition assessment of electrical apparatus with emi diagnostics," IEEE Transactions on Industrial Applications, vol. 53, pp. 693–699, 2017.

[15] I. M. Johnstone and B. W. Silverman, "Needles and Straw in Haystacks: Empirical Bayes Estimates of Possibly Sparse Sequences," Annals of Statistics, vol. 32, pp. 1594-1649, 2004.

[16] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," http://tensorflow.org/, 2015.