Applications of a deep convolutional autoencoder to process pulses from a p-type point contact germanium detector

CAP Congress 2023 | Fredericton, NB

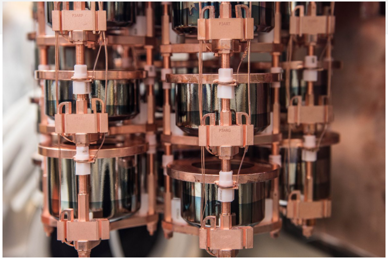
Mark Anderson
anderson.mark@queensu.ca

June 19th, 2023

Queen's University

Canadian Association of Physicists
Association canadienne des physiciens et physiciennes

Introduction



[10]

- High-purity germanium (HPGe) detectors widely used in beyond Standard Model rare event searches ($0\nu\beta\beta$, dark matter, etc.)[2, 3, 4, 5, 6, 7, 8, 9]

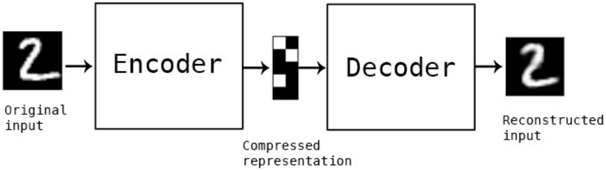
- Electronic noise makes signal identification challenging
  - Rare events in the presence of backgrounds

- Noise removal could help advance these searches
  - Identify low-energy signal events that would otherwise be dominated by electronic noise
  - Improved background rejection based on pulse shapes
  - More accurate measurements of pulse amplitudes $\rightarrow$ better energy resolution

- Deep learning has been successfully used in other fields (typically 2D images)
  - Why not 1D pulses from HPGe detectors?

Autoencoders

- An autoencoder is an algorithm used to learn a useful representation of data
  - Trained to map the inputs to the inputs (with some form of constraint)

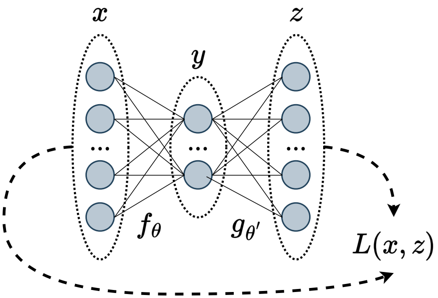


[11]

- By definition, an autoencoder is *lossy*
  - The goal is to retain as much *useful* information as possible
- Typically a (deep) neural network

Denoising autoencoders

- Denoising autoencoders impose the constraint that reconstruction must also remove noise
  - Proposed as a method to extract robust features for other classification tasks[12]
  - Input becomes a corrupted version of x, $\tilde{x}$, by some process $q_{\mathcal{D}}$



[1]

Internal/latent representation, y:

$$f_\theta(x) = y$$

Input reconstruction, z:

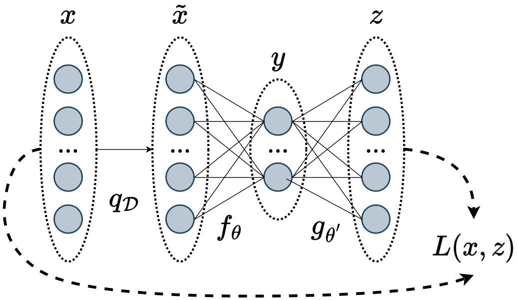
$$g_{\theta'}(y) = g_{\theta'}(f_\theta(x)) = z$$

Minimize some **loss function** quantifying the reconstruction of x, $L(x, z)$

$$L(x, z) = \frac{1}{N} \sum_{i}^{N} \|z_i - x_i\|_2^2$$

Denoising autoencoders

- Denoising autoencoders impose the constraint that reconstruction must also remove noise
  - Proposed as a method to extract robust features for other classification tasks[12]
  - Input becomes a corrupted version of x, $\tilde{x}$, by some process $q_{\mathcal{D}}$



[1]

Internal/latent representation, y:

$$f_\theta(\tilde{x}) = y$$

Input reconstruction, z:

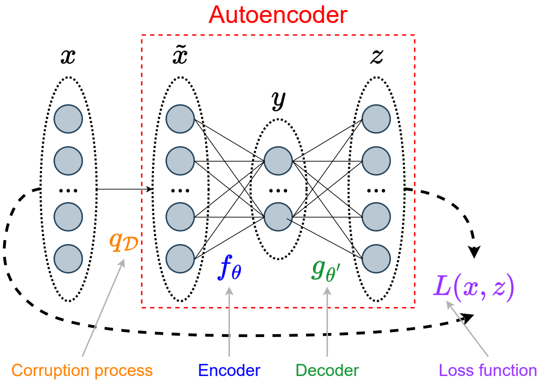
$$g_{\theta'}(y) = g_{\theta'}(f_\theta(\tilde{x})) = z$$

Minimize some **loss function** quantifying the reconstruction of x, $L(x, z)$

$$L(x, z) = \frac{1}{N} \sum_{i}^{N} \|z_i - x_i\|_2^2$$

Denoising autoencoders

- Denoising autoencoders impose the constraint that reconstruction must also remove noise
  - Proposed as a method to extract robust features for other classification tasks[12]
  - Input becomes a corrupted version of x, $\tilde{x}$, by some process $q_{\mathcal{D}}$



[1]

Internal/latent representation, y:

$$f_\theta(\tilde{x}) = y$$

Input reconstruction, z:

$$g_{\theta'}(y) = g_{\theta'}(f_\theta(\tilde{x})) = z$$

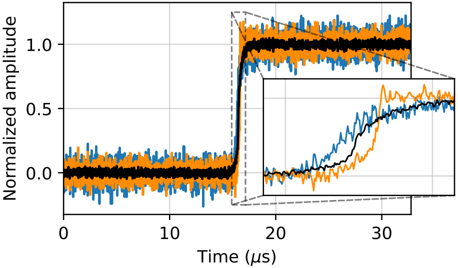
Minimize some **loss function** quantifying the reconstruction of x, $L(x, z)$

$$L(x, z) = \frac{1}{N}\sum_{i}^{N} \|z_i - x_i\|_2^2$$

The detector

- Signals are from a $1\,$kg p-type point contact detector located at Queen's University
  - Cylindrical with a radius of $3\,$cm and height of $5\,$cm
  - Manufactured by ORTEC/AMTEK
  - Operated in a PopTop cryostat

[13]



[1]

- Each signal is a sequence of voltages sampled at a fixed interval
  - Observed noise levels after preprocessing reflect energy of pulse; signal-to-noise ratio (SNR)
  - Different rise times reflect different positions

Datasets: real detector data

Americium-241 source

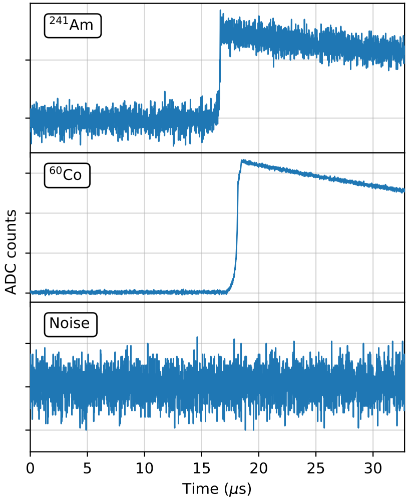
- Produces $60\,\text{keV}$ γs
- Almost entirely single-site events
- Lower energy (higher noise), good for *validation*

Cobalt-60 source

- Produces $1173\,\text{keV}$ and $1332\,\text{keV}$ γs
- Numerous multi-site events from Compton scatters
- Higher energy (lower noise), good for *training*

Detector noise

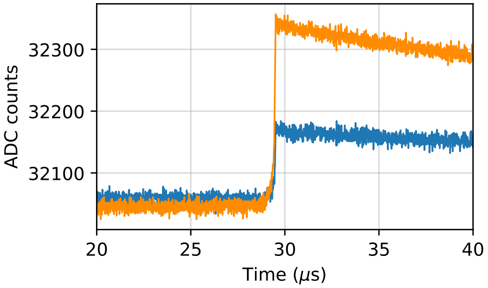
- Collected by randomly triggering the detector (and removing actual signals that occasionally occur)
- Used for *data augmentation*



Datasets: real detector data (preprocessing)

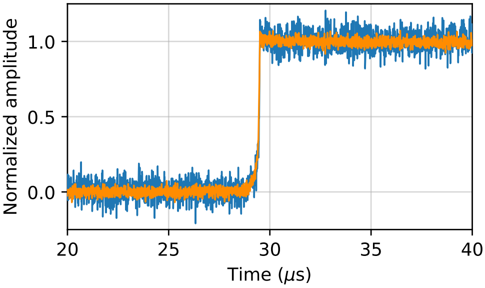
- Data pulses preprocessed to remove baseline
- Data pulses have exponential decay removed with pole zero correction
- Data pulses scaled by amplitude (calculated with a trapezoidal filter)

} **Amplitude normalization**



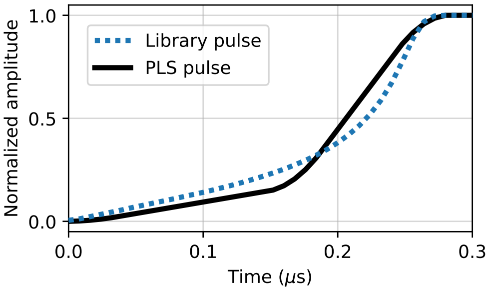
Before preprocessing

⟶



After preprocessing

Datasets: simulated data



[1]

Library pulses

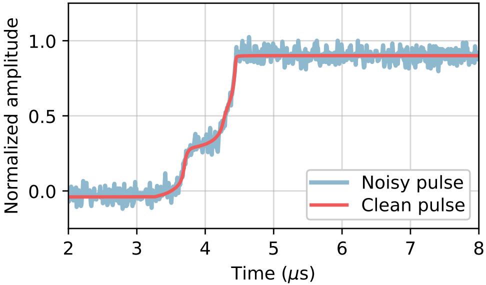
- 1724 simulated "library" pulses[14]
- Each pulse corresponds to point on 1 mm × 1 mm azimuthally symmetric grid
- Created using `siggen` simulation software[15]
- Used to infer position of real events

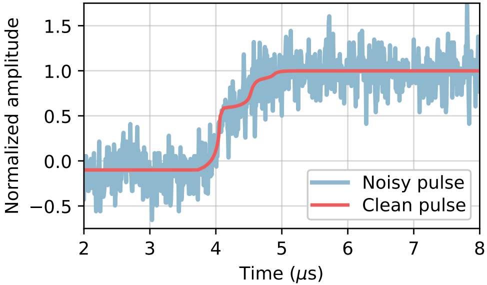
Piecewise linear smoothed (PLS) pulses

- Mimic the general shape of the library pulses without the requirement of complex physics simulations

Datasets: simulated data (data augmentation)

- From the simulated single-site event pulses, can create a diverse training set
  - Combine single-site simulated pulses to create *artificial multi-site events*
  - Apply random *horizontal shifts, vertical shifts, and amplitude scales* to each pulse
  - Add *detector noise* to each pulse with a random standard deviation





Note: no preprocessing required for simulated pulses!

Training procedures

Regular

- Trained to map the noisy pulse to the corresponding clean underlying pulse
- Must know the true pulse – *only works on simulated data*

Noise2Noise[16]

- Trained to map noisy pulse to noisy pulse (different noisy realizations of same underlying signal)
  - An impossible task in practice
  - Model will instead learn to predict the mean, given infinite different noisy realizations
- Can be used with simulations, *but this is not required*
  - For detector data, add even more noise to the already noisy pulse
  - Include a *total variation* penalty[17] to original loss function L_0 to account for the noisy true mean
  - Penalize the absolute difference between given sample (j) and subsequent sample $(j+1)$ in pulse
  - Apply scaling factor λ to control weighting

$$L = L_0 + \frac{\lambda}{N} \sum_{i}^{N} \sum_{j}^{M-1} |z_{i,j+1} - z_{i,j}|$$

Training procedures

Regular

- Trained to map the noisy pulse to the corresponding clean underlying pulse
- Must know the true pulse – *only works on simulated data*

Noise2Noise[16]

- Trained to map noisy pulse to noisy pulse (different noisy realizations of same underlying signal)
  - An impossible task in practice
  - Model will instead learn to predict the mean, given infinite different noisy realizations
- Can be used with simulations, *but this is not required*
  - For detector data, add even more noise to the already noisy pulse
  - Include a *total variation* penalty[17] to original loss function L_0 to account for the noisy true mean
  - Penalize the absolute difference between given sample (j) and subsequent sample ($j+1$) in pulse
  - Apply scaling factor λ to control weighting

$$L = L_0 + \frac{\lambda}{N} \sum_{i}^{N} \sum_{j}^{M-1} |z_{i,j+1} - z_{i,j}|$$

Training procedures

Regular

- Trained to map the noisy pulse to the corresponding clean underlying pulse
- Must know the true pulse – *only works on simulated data*

Noise2Noise[16]

- Trained to map noisy pulse to noisy pulse (different noisy realizations of same underlying signal)
  - An impossible task in practice
  - Model will instead learn to predict the mean, given infinite different noisy realizations
- Can be used with simulations, *but this is not required*
  - For detector data, add even more noise to the already noisy pulse
  - Include a *total variation* penalty[17] to original loss function L_0 to account for the noisy true mean
  - Penalize the absolute difference between given sample (j) and subsequent sample ($j+1$) in pulse
  - Apply scaling factor λ to control weighting

$$L = L_0 + \frac{\lambda}{N} \sum_{i}^{N} \sum_{j}^{M-1} |z_{i,j+1} - z_{i,j}|$$

Training procedures

Regular

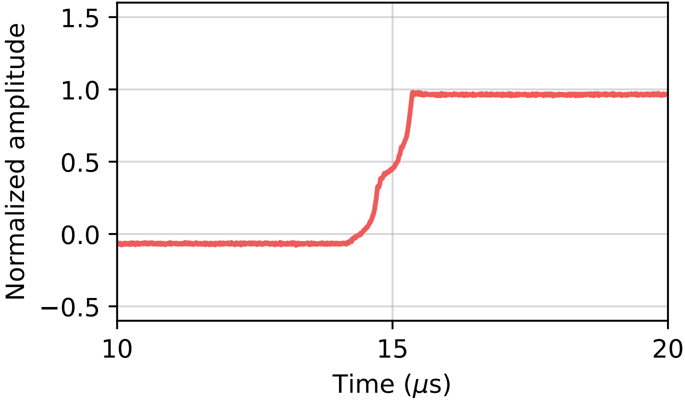
- Trained to map the noisy pulse to the corresponding clean underlying pulse
- Must know the true pulse – *only works on simulated data*

Noise2Noise[16]

- Trained to map noisy pulse to noisy pulse (different noisy realizations of same underlying signal)
    - An impossible task in practice
    - Model will instead learn to predict the mean, given infinite different noisy realizations
- Can be used with simulations, *but this is not required*
    - For detector data, add even more noise to the already noisy pulse
    - Include a *total variation* penalty[17] to original loss function L_0 to account for the noisy true mean
    - Penalize the absolute difference between given sample (j) and subsequent sample ($j+1$) in pulse
    - Apply scaling factor λ to control weighting

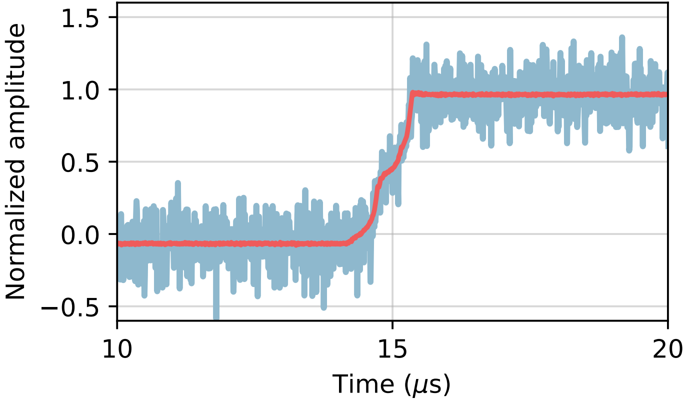
$$L = L_0 + \frac{\lambda}{N} \sum_{i}^{N} \sum_{j}^{M-1} |z_{i,j+1} - z_{i,j}|$$

Original ^{60}Co data pulse (low noise/high energy/large SNR)



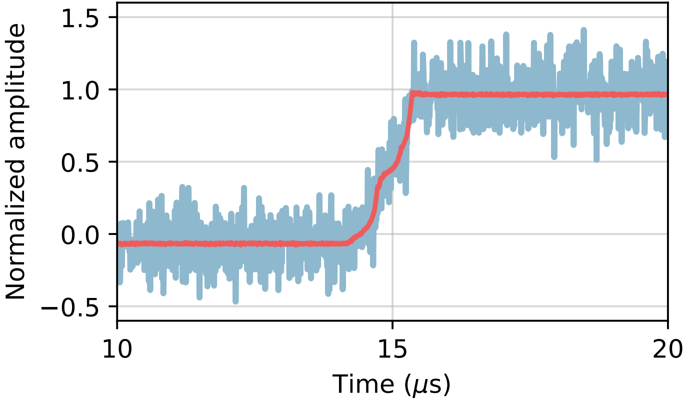
Noise2Noise

Original ^{60}Co data pulse with a random noise pulse

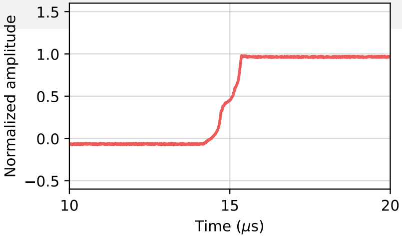


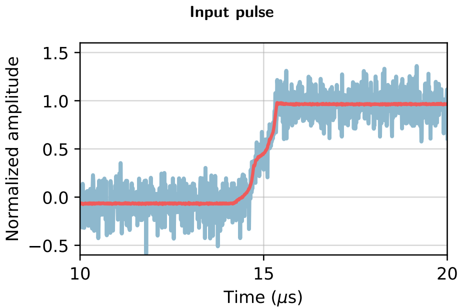
Noise2Noise

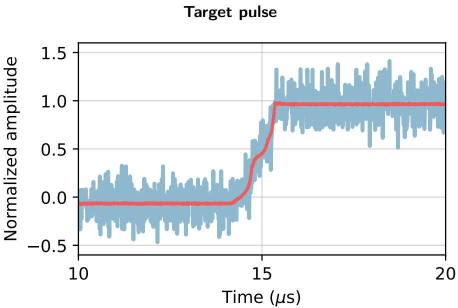
Original ^{60}Co data pulse with *another* random noise pulse



Noise2Noise

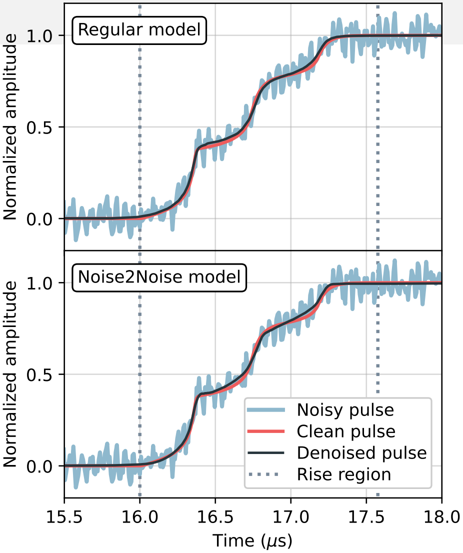






Results: simulations

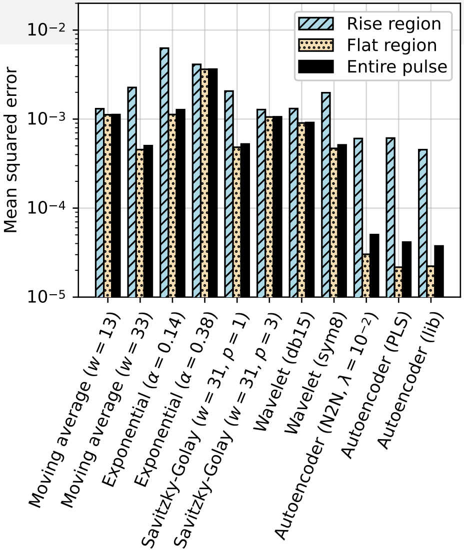
- Qualitatively, denoising with deep learning performs very well on simulations
- Autoencoder is superior to all traditional denoising methods investigated
  - Compared mean squared error on test set containing simulated single-/multi-site events
  - Each method optimized on a separate validation set to select hyperparameters
- Regular training procedure (simulations) outperforms Noise2Noise (^{60}Co data)
  - Still very good performance with Noise2Noise



[1] (modified)

Results: simulations

- Qualitatively, denoising with deep learning performs very well on simulations
- Autoencoder is superior to all traditional denoising methods investigated
    - Compared mean squared error on test set containing simulated single-/multi-site events
    - Each method optimized on a separate validation set to select hyperparameters
- Regular training procedure (simulations) outperforms Noise2Noise (^{60}Co data)
    - Still very good performance with Noise2Noise



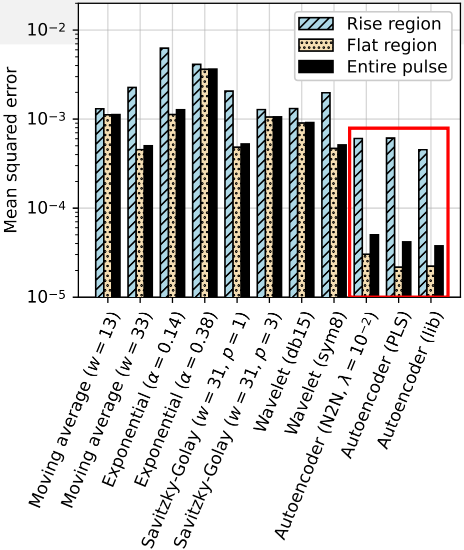
[1]

Results: simulations

- Qualitatively, denoising with deep learning performs very well on simulations

- Autoencoder is superior to all traditional denoising methods investigated
    - Compared mean squared error on test set containing simulated single-/multi-site events
    - Each method optimized on a separate validation set to select hyperparameters

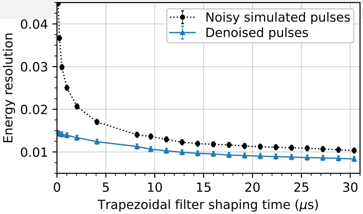
- Regular training procedure (simulations) outperforms Noise2Noise (^{60}Co data)
    - Still very good performance with Noise2Noise



[1]

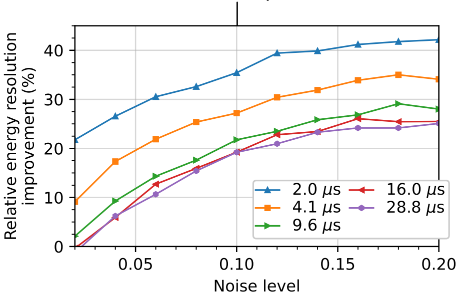
Results: simulations (energy resolution)

- In terms of physics results, allows for an improvement in the energy resolution
  - Energy calculated from the amplitude of a trapezoidal filter with given shaping time
  - FWHM of peak is the energy resolution
- Created test datasets with different noise levels and evaluated the energy resolution of each
- At every noise level and shaping time, the results after denoising with our autoencoder are superior
  - Proportionally larger improvements with increasing noise level, decreasing shaping time



[1]

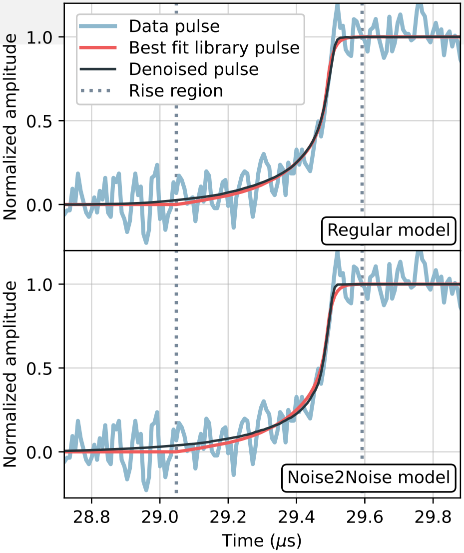
Noise level of 0.10



[1]

Results: data

- Qualitatively, denoising with deep learning performs very well on data
- More difficult to quantify denoising
  - No true underlying pulse to compare to
- However, can make a statistical comparison to evaluate the performance



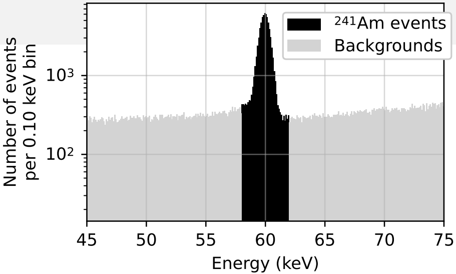
[1] (modified)

Results: data (χ^2 comparison)

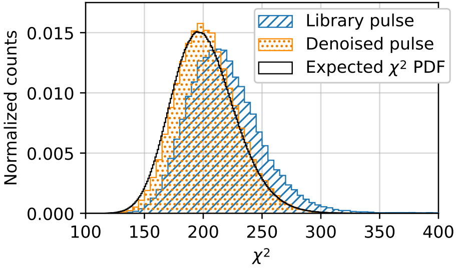
- ^{241}Am dataset contains mostly single-site events from $60\,\text{keV}$ γs
- Use a χ^2 comparison between the original pulse and denoised pulse, best-fit library pulse

$$\chi^2\left(x_i, z_i\right) = \sum_{j=M_1}^{M_2} \frac{\left(z_{i,j} - x_{i,j}\right)^2}{\sigma_i^2}$$

- χ^2 distribution between noisy and denoised pulse is consistent with expected χ^2 distribution of our detector noise
  - Taken over 200 samples containing rise region (M_1, M_2 set appropriately)



[1]



[1]

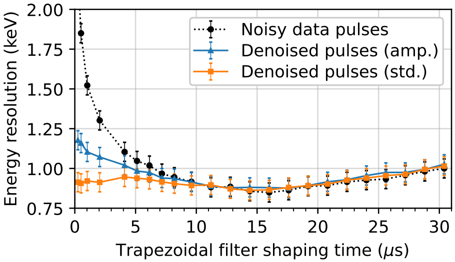
Results: data (energy resolution)

- Can also evaluate the effect of denoising on the energy resolution, compare to simulations

- Using ^{241}Am data, optimal energy resolution is comparable before and after denoising

- Much lower shaping time required to achieve good energy resolution
  - Important for data storage, analysis, etc.

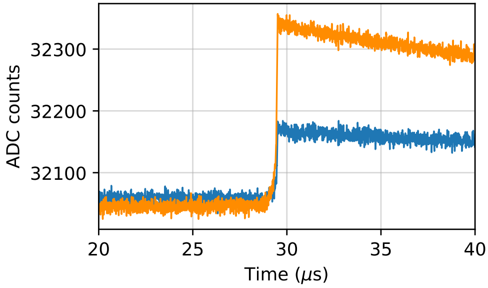
- However, results are not as good as simulations would suggest
  - What's different?



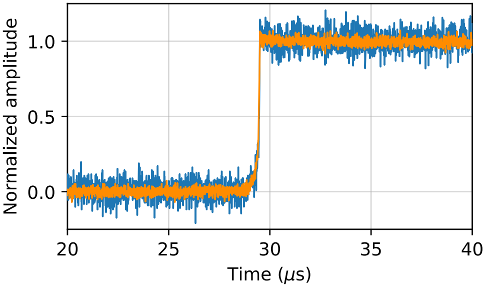
[1]

Results: data (energy resolution)

- We assume data pulses have *one* exponential decay and correct for that



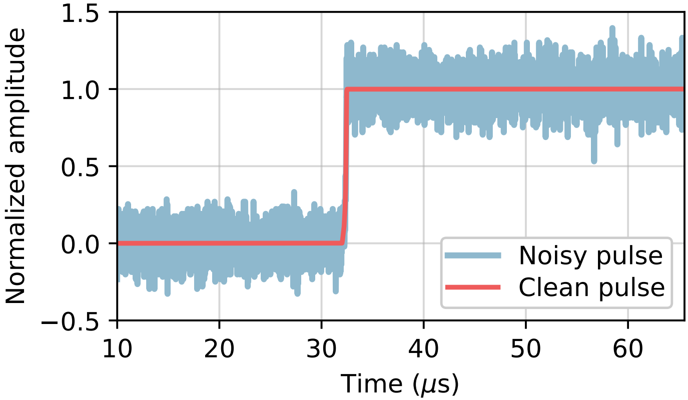
⟶



- In reality, there are *multiple* sources of exponential decay, usually small, but still contribute
- A single "effective" pole zero correction is thus imperfect and leaves residual effects from the other exponential decays

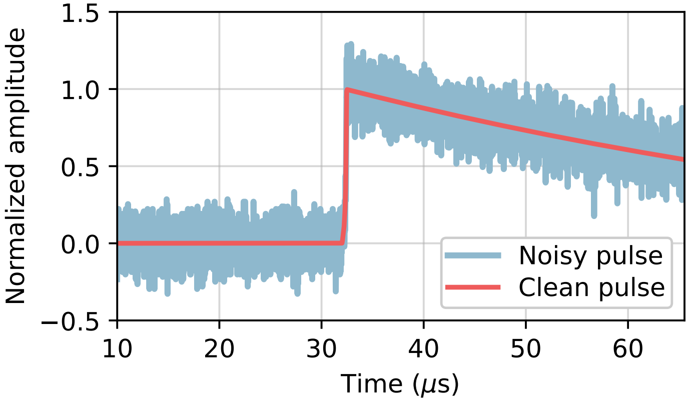
Results: data (energy resolution)

Simulated pulse with noise



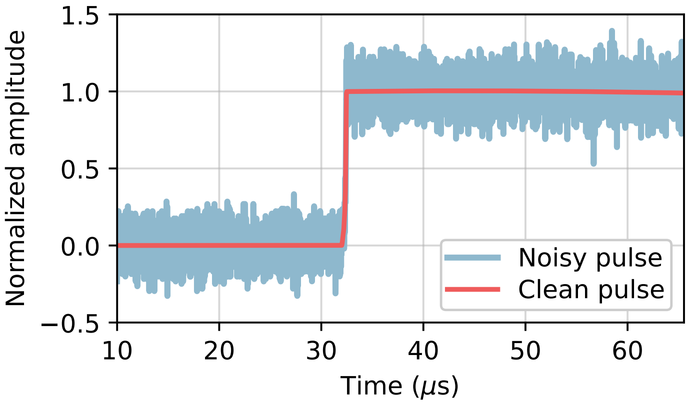
Results: data (energy resolution)

Simulated pulse with noise (convolved with multiple exponentials)



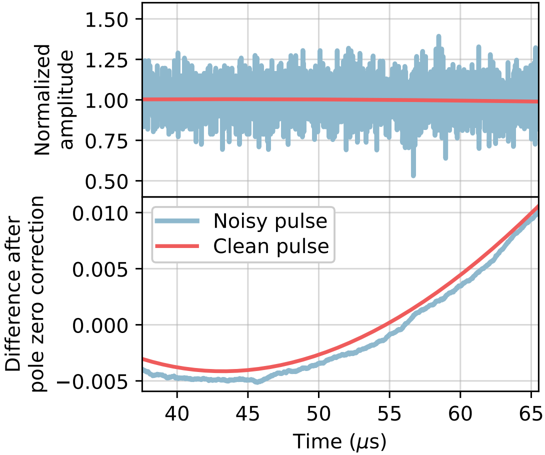
Results: data (energy resolution)

Simulated pulse with noise (deconvolved with one "effective" exponential)



Results: data (energy resolution)

Simulated pulse with noise (deconvolved with one "effective" exponential)



Conclusions and future work

- Deep convolutional autoencoders are effective at removing electronic noise from HPGe detector pulses
  - Outperforms various traditional denoising methods
  - Denoised pulses are statistically consistent with data pulses
  - Can reach optimal energy resolution with a lower shaping time
    - Simulations suggest improvements in the overall optimal energy resolution are possible
    - Accounting for effects present in real data could improve results

- Models can be trained *without* the need for detailed detector simulations
  - PLS pulses are a very rough approximation to library pulses
  - Noise2Noise method requires *only* noisy detector data
    - Results could likely be improved with more (diverse) data

Conclusions and future work

- Results presented here are focused on HPGe detector data
  - Noise removal is beneficial in many contexts
  - Our group is applying these methods to signals from other detector technologies
    - Gaseous proportional counters (e.g., see talk from Noah Rowe), bubble chambers

- Our group is also exploring various extensions of this research
  - New network architectures such as CycleGAN[18] for improved performance
    - Potential to improve modelling of multiple exponential decays, and thus energy resolution, due to the less stringent requirement of unpaired simulated and detector pulses
  - Continuous inline denoising before triggering to reduce trigger thresholds
    - Useful to identify low SNR events otherwise dominated by electronic noise

- Work is broadly applicable to the particle astrophysics community and has great potential to be expanded on

Thank You!

More details in the published paper.
Check it out!

doi:**10.1140/epjc/s10052-022-11000-w**

arXiv:**2204.06655**

Acknowledgements

- Thanks to all other authors of the paper
  - V. Basu, R. D. Martin, C. Z. Reed, N. J. Rowe, M. Shafiee, T. Ye
- This work was supported by the Natural Sciences and Engineering Research Council of Canada, the Arthur B. McDonald Canadian Astroparticle Physics Research Institute, and the Canada Foundation for Innovation
- The author held a Walter C. Sumner Memorial Fellowship

Queen's UNIVERSITY

NSERC CRSNG

Arthur B. McDonald
Canadian Astroparticle Physics Research Institute

INNOVATION
Canada Foundation for Innovation
Fondation canadienne pour l'innovation

References I

[1] M. R. Anderson *et al.*, "Performance of a convolutional autoencoder designed to remove electronic noise from p-type point contact germanium detector signals," *Eur. Phys. J. C*, vol. 82, no. 12, p. 1084, 2022.

[2] S. I. Alvis *et al.*, "Search for neutrinoless double-β decay in $^{76}\mathrm{Ge}$ with $26\,\mathrm{kg-yr}$ of exposure from the Majorana Demonstrator," *Phys. Rev. C*, vol. 100, p. 025501, 2019.

[3] M. Agostini *et al.*, "Final results of GERDA on the search for neutrinoless double-β decay," *Phys. Rev. Lett.*, vol. 125, p. 252502, 2020.

[4] N. Abgrall *et al.*, "LEGEND-1000 preconceptual design report," *arXiv preprint arXiv:2107.11462*, 2021.

[5] C. E. Aalseth *et al.*, "CoGeNT: A search for low-mass dark matter using p-type point contact germanium detectors," *Phys. Rev. D*, vol. 88, no. 1, p. 012002, 2013.

[6] M. Agostini *et al.*, "First search for bosonic superweakly interacting massive particles with masses up to $1\,\mathrm{MeV/c^2}$ with GERDA," *Phys. Rev. Lett.*, vol. 125, no. 1, p. 011801, 2020.

[7] N. Abgrall *et al.*, "Search for Pauli Exclusion Principle violating atomic transitions and electron decay with a p-type point contact germanium detector," *Eur. Phys. J. C.*, vol. 76, no. 11, pp. 1–5, 2016.

References II

[8] S. I. Alvis *et al.*, "First limit on the direct detection of lightly ionizing particles for electric charge as low as $e/1000$ with the Majorana Demonstrator," *Phys. Rev. Lett.*, vol. 120, no. 21, p. 211804, 2018.

[9] S. I. Alvis *et al.*, "Search for trinucleon decay in the Majorana Demonstrator," *Phys. Rev. D*, vol. 99, no. 7, p. 072004, 2019.

[10] "Underground neutrino experiment could provide greater clarity on matter-antimatter imbalance," accessed: 2023-05-13. [Online]. Available: https://newscenter.lbl.gov/2018/03/26/neutrino-experiment-probes-matter-antimatter-imbalance/

[11] "Building autoencoders in keras," accessed: 2023-05-13. [Online]. Available: https://blog.keras.io/building-autoencoders-in-keras.html

[12] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[13] "P-type point-contact (PPC) germanium detectors," accessed: 2023-05-13. [Online]. Available: https://www.npl.washington.edu/majorana/design-technologies

References III

[14] Vasundhara, "Pulse fitting for event localization in high purity germanium point contact detectors," Master's thesis, Queen's University, 2020.

[15] D. C. Radford, "`siggen`," 2017. [Online]. Available: https://github.com/radforddc/icpc_siggen

[16] J. Lehtinen *et al.*, "Noise2Noise: Learning image restoration without clean data," in *Proc. Int. Conf. Mach. Learn.*, vol. 80, 2018, pp. 2965–2974.

[17] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1-4, pp. 259–268, 1992.

[18] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2223–2232.

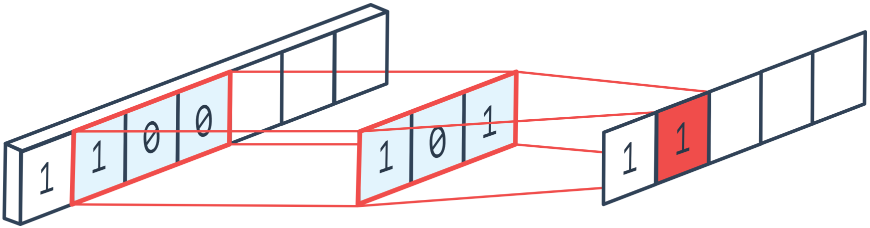
[19] "1D convolution," accessed: 2022-08-10. [Online]. Available: https://peltarion.com/knowledge-center/documentation/modeling-view/build-an-ai-model/blocks/1d-convolution

[20] "Med Phys 4RA3, 4RB3/6R03 - Chapter 6: Pulse Processing," accessed: 2023-05-13. [Online]. Available: https://www.science.mcmaster.ca/radgrad/images/6R06CourseResources/4RA34RB3_Lecture_Note_6_Pulse-Processing.pdf

Additional Slides

Model architecture

- Fully convolutional autoencoder
  - Weight sharing provides consistent noise removal across pulse
  - Feature locality and shift equivariance
  - Allows for a variable input shape (subject to some restrictions)
  - Significant reduction in the number of trainable parameters



[19]

Model architecture

Layer	Stride	Window	Output
Input			4096, 1
Convolution	1	1	4096, 8
Convolution	1	9	4088, 16
Average Pooling	2	2	2044, 16
Convolution	1	17	2028, 32
Average Pooling	2	2	1014, 32
Convolution	1	33	982, 64
Average Pooling	2	2	491, 64
Convolution	1	33	459, 32
Transpose Convolution	1	33	491, 32
Upsampling	2	2	982, 64
Transpose Convolution	1	33	1014, 64
Upsampling	2	2	2028, 64
Transpose Convolution	1	17	2044, 32
Upsampling	2	2	4088, 32
Transpose Convolution	1	9	4096, 16
Convolution (output)	1	1	4096, 1

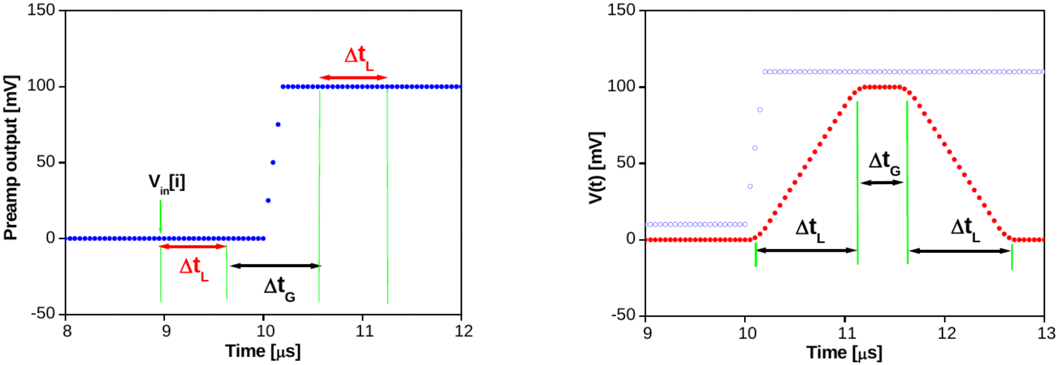
[1]

Results on simulations

Training procedure and data			Mean squared error ($\times 10^{-5}$)			
			Gaussian noise		Detector noise	
Procedure	**Data**	**Noise**	**Library**	**PLS**	**Library**	**PLS**
Regular	Library	Detector	4.12	4.72	3.76	4.21
Regular	Library	Gaussian	3.40	3.82	4.50	4.77
Regular	PLS	Detector	5.10	4.48	4.15	3.57
Regular	PLS	Gaussian	3.93	3.36	5.02	4.31
N2N ($\lambda = 0$)	Library	Detector	3.90	4.37	3.86	4.20
N2N ($\lambda = 0$)	Library	Gaussian	3.46	3.87	4.57	4.82
N2N ($\lambda = 0$)	PLS	Detector	5.11	4.48	4.14	3.55
N2N ($\lambda = 0$)	PLS	Gaussian	3.85	3.46	4.97	4.43
N2N ($\lambda = 0$)	Detector	Detector	6.54	6.30	7.78	7.40
N2N ($\lambda = 10^{-2}$)	Detector	Detector	4.17	4.54	5.04	5.26

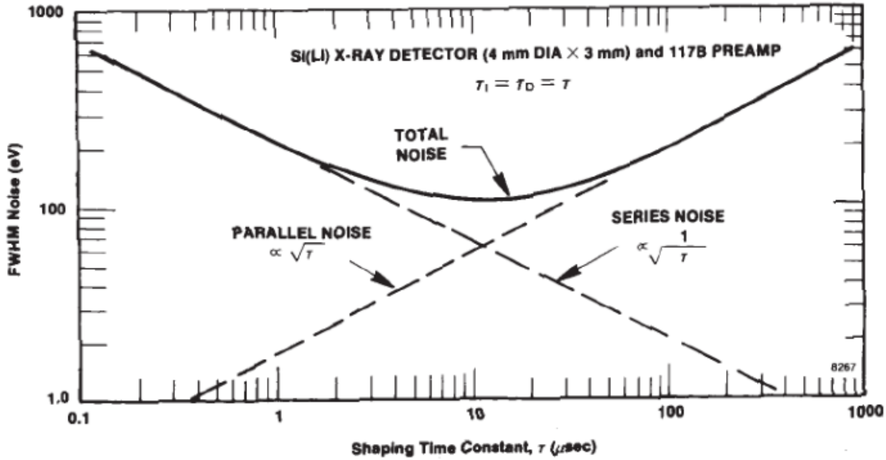
[1]

Trapezoidal filter



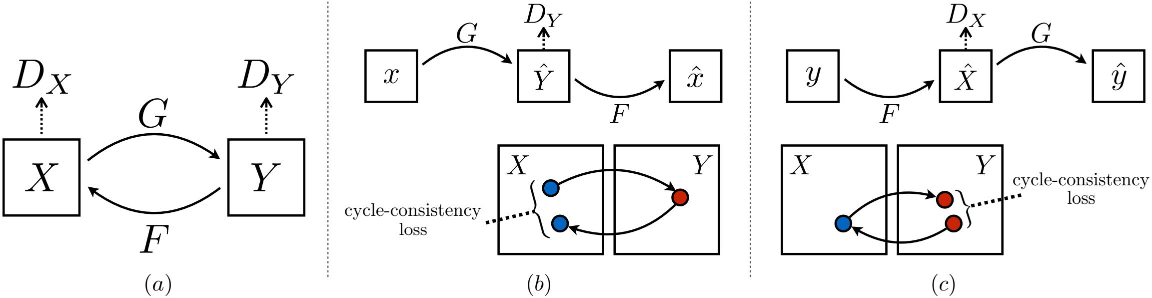
[20]

Noise curve



[20]

CycleGAN



D_X
D_Y
G
X
Y
F
(a)
D_Y
G
x
$\hat{Y}$
$\hat{x}$
F
cycle-consistency
loss
X
Y
(b)
D_X
G
y
$\hat{X}$
$\hat{y}$
F
X
Y
cycle-consistency
loss
(c)

[18]