Neural Arithmetic Units

Andreas Madsen[†‡]
amwebdk@gmail.com

Alexander Rosenberg Johansen[†]
aler@dtu.dk

Who else?

[†]Technical University of Denmark [‡]Computationally Demanding

Abstract

Exact addition, subtraction, multiplication and division present a unique challenge for machine learning models. Neural networks can approximate complex functions by learning from labeled data. However, when extrapolating to out-of-distribution samples on arithmetic operations neural networks often fail to learn the underlying logic, which can be the limiting factor for application such as comparing, counting and inferring physical models. Our proposed Neural Addition Unit (NAU) and Neural Multiplication Unit (NMU) rely on constrained weights to learn rules and extrapolate well beyond the training distribution. The proposed NAU and NMU are inspired by the underlying arithmetic units of the Neural Arithmetic Logic Unit (NALU). The NAU learns addition and subtraction through a linear layer of regularized and constrained weights. The NMU learns multiplication through an accumulative product of the input using gating with an identity function to mask out unwanted elements. Through analytic and empirical analysis we justify how the NAU and NMU improve over the Neural Arithmetic Logic Unit (NALU), a linear regression model and a ReLU based multi-layer perceptron (MLP). Our NAU and NMU have fewer parameters, converges more consistently, learns faster and have more meaningful discrete values than the NALU and its underlying units.

1 Introduction

[1]

The ability for neurons to hold numbers and do arithmetic operations has been documented in both humans, non-human primates [**?**], newborn chicks [**?**] and bees [**?**]. In our race to solve intelligence we have put much faith in neural networks, which in turn has provided unparalleled and often superhuman performance in many tasks requiring high cognitive ability [**???**]. However, when using neural networks to solve simple arithmetic problems, such as counting, they systematically fail to extrapolate [**???**].

In this paper, we analyze and improve parts of the recently proposed Neural Arithmetic Logic Unit (NALU) [**?**]. Our contribution is an alternative formulation of the weight constraint with a clipped linear activation, a regularizer that bias throwards sparse solutions, and a reformulation of the mul-

[1] In the interest of scientific integrity, we have made the code for all experiments, and more, available on GitHub: `https://github.com/AndreasMadsen/stable-nalu`.

Preprint. Under review.

tiplication unit to be partially linear. All of which significantly improves upon the existing NAC_+ and $\text{NAC}_\bullet$ units as shown through extensive testing on arithmetic constructions.

The NALU is a neural network layer with two sub-units. The two sub-units, NAC_+ for addition/subtraction and $\text{NAC}_\bullet$ for multiplication/division, are softly gated between with a sigmoid function. By using trainable weights, and restricting the weights towards $\{-1, 0, 1\}$. The weights are learned by observing arithmetic input-output pairs and using backpropagation[**?**].

We focus only on the NAC_+ and $\text{NAC}_\bullet$ as we have found that the gating in NALU can be cumbersome, as shown in table 4 where the NALU performs significantly worse than NAC_+ and $\text{NAC}_\bullet$. This is because of the difficulties in selecting between, and simultaneously training, two vastly different operations.

We will thus assume that the appropriate operation is already known, or can empirically be found by varying the network architecture (oracle gating). We find that the NAC_+ and $\text{NAC}_\bullet$ units poses optimization difficulties. We present the following findings:

- The gradients from the weight matrix construction in NAC_+ and $\text{NAC}_\bullet$, have zero expectation.
- The $\text{NAC}_\bullet$ have a treacherous optimization space with unwanted global minimas (as shown in figure 2) and have exploding/vanishing gradients.
- Using the addition module NAC_+, we observe that the wanted weight matrix values of $\{-1, 0, 1\}$ is rarely found.

Motivated by these convergence and sparsity issue, we propose alternative formulations of the NAC_+ and $\text{NAC}_\bullet$, which we call the Neural Addition Unit (NAU) and Neural Multiplication Unit (NMU).

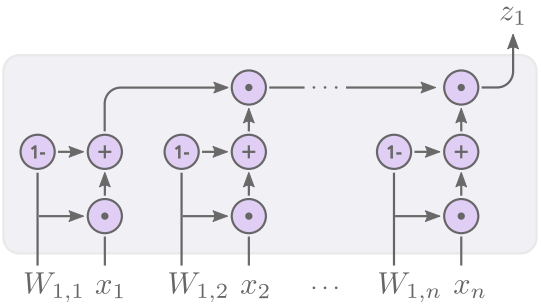


Figure 1: Visualization of NMU for a single output scalar z_1, this construction repeats for every element in the output vector $\mathbf{z}$.

2 Introducing differentiable binary arithmetic operations

Our goal is to achieve arithmetic operations between the elements of a vector. Such that the output is an addition, subtraction, multiplication, or division of arbitrary elements of a vector $\mathbf{x}$ (e.g. $x_5 + x_1 \cdot x_7$). Formally defined as:

$$x_1 \circ_1 x_2 \circ_2 \cdots x_{k-1} \circ_{k-1} x_k \mid (x_1, \ldots, x_k) \in \mathbf{x}, \mathbf{x} \in \mathbb{R}^n, \circ_i \in \{+, -, \times, \div\} \tag{1}$$

The Neural Arithmetic Logic Unit (NALU) [**?**] attempts to solve equation 1 by presenting two sub-units; the NAC_+ and $\text{NAC}_\bullet$ to exclusively represent either the $\{+, -\}$ or the $\{\times, \div\}$ operations. The NALU attempts to have either NAC_+ or $\text{NAC}_\bullet$ selected exclusively, which could require the NALU to be applied multiple times (alternating between NAC_+ and $\text{NAC}_\bullet$) in order to represent the entire space of solutions for equation 1.

The NAC_+ and $\text{NAC}_\bullet$ are defined accordingly,

$$W_{h_\ell,h_{\ell-1}} = \tanh(\hat{W}_{h_\ell,h_{\ell-1}})\sigma(\hat{M}_{h_\ell,h_{\ell-1}}) \tag{2}$$

$$\text{NAC}_+ : \; z_{h_\ell} = \sum_{h_{\ell-1}=1}^{H_{\ell-1}} W_{h_\ell,h_{\ell-1}} z_{h_{\ell-1}} \tag{3}$$

$$\text{NAC}_\bullet : \; z_{h_\ell} = \exp\left(\sum_{h_{\ell-1}=1}^{H_{\ell-1}} W_{h_\ell,h_{\ell-1}} \log(|z_{h_{\ell-1}}| + \epsilon)\right) \tag{4}$$

where $\hat{\mathbf{W}}, \hat{\mathbf{M}} \in \mathbb{R}^{H_\ell \times H_{\ell-1}}$ are trainable weight matrices. The matrices are combined using tanh and sigmoid transformation to bias the parameters towards a $\{-1, 0, 1\}$ solution. Having $\{-1, 0, 1\}$ allows a linear layer to exactly emulate the binary $\{+, -\}$ operation between elements of a vector as used when computing the NAC_+. The $\text{NAC}_\bullet$ extends the NAC_+ by using an exponential log transformation, which, with $\{-1, 0, 1\}$ weight values, becomes the $\{\times, \div\}$ operations (within ϵ precision).

The NALU combines these units with a gating mechanism $\mathbf{z} = \mathbf{g} \odot \text{NAC}_+ + (1 - \mathbf{g}) \odot \text{NAC}_\bullet$ given $\mathbf{g} = \sigma(\mathbf{G}\mathbf{x})$. The idea is that the NALU should be a plug-and-play component in a neural network and has the ability to, with stochastic gradient descent and backpropagation, to learn the functionality in equation 1.

2.1 Challenges of the NALU, NAC_+ and $\text{NAC}_\bullet$

To simplify the problem we have chosen to leave out the gating mechanism and focus on the sub-units, assuming "oracle gating". We have not had any consistent success of convergence using the gating mechanism using the NALU or by combining our own proposed sub-units (NAU, NMU), as shown in table 4. We find that gating between NAC_+ and $\text{NAC}_\bullet$ is challenging. This is likely due to the vastly different gradients, causing addition to be learned much faster than multiplication.

2.1.1 Weight matrix construction

The weight matrix construction $\tanh(\hat{W}_{h_{\ell-1},h_\ell})\sigma(\hat{M}_{h_{\ell-1},h_\ell})$ has the following properties that could make convergence challenging using gradient decent.

The loss gradient with respect to the weight matrices can be derived from equation 2.

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \hat{W}_{h_{\ell-1},h_\ell}} &= \frac{\partial \mathcal{L}}{\partial W_{h_{\ell-1},h_\ell}}(1 - \tanh^2(\hat{W}_{h_{\ell-1},h_\ell}))\sigma(\hat{M}_{h_{\ell-1},h_\ell}) \\
\frac{\partial \mathcal{L}}{\partial \hat{M}_{h_{\ell-1},h_\ell}} &= \frac{\partial \mathcal{L}}{\partial W_{h_{\ell-1},h_\ell}} \tanh(\hat{W}_{h_{\ell-1},h_\ell})\sigma(\hat{M}_{h_{\ell-1},h_\ell})(1 - \sigma(\hat{M}_{h_{\ell-1},h_\ell}))
\end{aligned} \tag{5}$$

The gradient $E\left[\partial\mathcal{L}/\partial\hat{M}_{h_{\ell-1},h_\ell}\right] = 0$ can be problematic as we prefer zero having a zero mean expectation of our output. Something that can only be ensured with $E[\hat{W}_{h_{\ell-1},h_\ell}] = 0$ [**?**].

In our empirical analysis we find that equation 2 does not create the desired bias for $\{-1, 0, 1\}$, as it doesn't converge towards those values.

To create a bias and prevent the gradient challenges of equation 5 we propose a simple clamped linear construction with an out-of-bound regularizer $\mathcal{R}_{\ell,\text{oob}}$ to force $\hat{W}$ to be within $[-1, 1]$ and

ensure that the gradient is always present.

$$
\begin{aligned}
W_{h_{\ell-1},h_\ell} &= \min(\max(\hat{W}_{h_{\ell-1},h_\ell}, -1), 1), \\
\mathcal{R}_{\ell,\text{bias}} &= \frac{1}{H_\ell + H_{\ell-1}} \sum_{h_\ell=1}^{H_\ell} \sum_{h_{\ell-1}=1}^{H_{\ell-1}} \hat{W}^2_{h_{\ell-1},h_\ell} (1 - |\hat{W}_{h_{\ell-1},h_\ell}|)^2 \\
\mathcal{R}_{\ell,\text{oob}} &= \frac{1}{H_\ell + H_{\ell-1}} \sum_{h_\ell=1}^{H_\ell} \sum_{h_{\ell-1}=1}^{H_{\ell-1}} \max(|\hat{W}_{h_{\ell-1},h_\ell}| - 1, 0)^2 \\
\text{NAU}: \; z_{h_\ell} &= \sum_{h_{\ell-1}=1}^{H_{\ell-1}} W_{h_\ell,h_{\ell-1}} z_{h_{\ell-1}} \\
\mathcal{L} &= \hat{\mathcal{L}} + \lambda_{\text{bias}} \mathcal{R}_{\ell,\text{bias}} + \lambda_{\text{oob}} \mathcal{R}_{\ell,\text{oob}}
\end{aligned}
\tag{6}
$$

2.1.2 Challenges of division

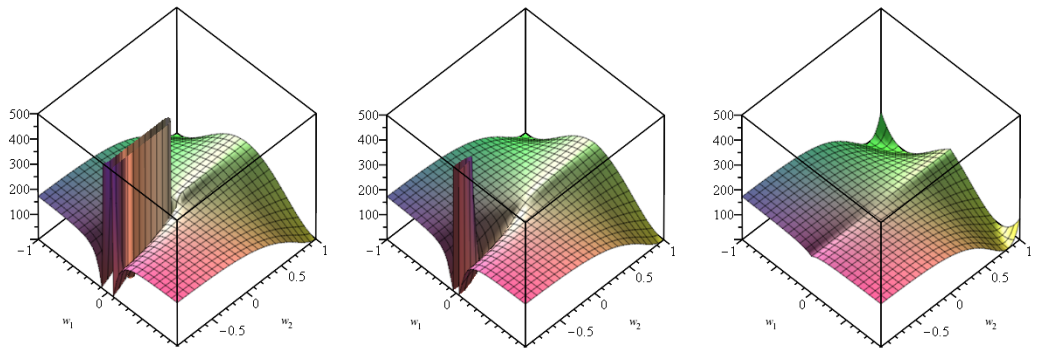
The NAC$_\bullet$, as formulated in equation 4, has the ability to learn exact multiplication and division of elements from a vector if the weights of $W_{h_{\ell-1},h_\ell}$ are one of $\{-1, 0, 1\}$.

However, backpropagation through the NAC$_\bullet$ unit reveals that if $|z_{h_{\ell-1}}|$ is near zero, $W_{h_{\ell-1},h_\ell}$ is negative and ϵ is small, the gradient term will explode and oscillate between large postive and large negative values, which can be problematic in optimization [**?**], as visualized in figure 2.

$$
\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}} = \sum_{h_\ell=1}^{H_\ell} \frac{\partial \mathcal{L}}{\partial z_{h_\ell}} \frac{\partial z_{h_\ell}}{\partial z_{h_{\ell-1}}} = \sum_{h_\ell=1}^{H_\ell} \frac{\partial \mathcal{L}}{\partial z_{h_\ell}} z_{h_\ell} W_{h_\ell,h_{\ell-1}} \frac{\text{sign}(z_{h_{\ell-1}})}{|z_{h_{\ell-1}}| + \epsilon} \tag{7}
$$

(see full derivation in Appendix A.2)

This is not an issue for for positive values of $W_{h_{\ell-1},h_\ell}$ (multiplication), as z_{h_ℓ} and $z_{h_{\ell-1}}$ will be correlated causing the terms z_{h_ℓ} and $\frac{\text{sign}(z_{h_{\ell-1}})}{|z_{h_{\ell-1}}|+\epsilon}$ to partially cancel out.



(a) $\epsilon = 10^{-7}$ (b) $\epsilon = 0.1$ (c) $\epsilon = 1$

Figure 2: RMS loss curvature for a NAC$_+$ layer followed by a NAC$_\bullet$ layer. The weight matrices constrained are to $\mathbf{W}_1 = \begin{bmatrix} w_1 & w_1 & 0 & 0 \\ w_1 & w_1 & w_1 & w_1 \end{bmatrix}$, $\mathbf{W}_2 = \begin{bmatrix} w_2 & w_2 \end{bmatrix}$. The problem is $x = (1, 1.2, 1.8, 2)$, $t = 13.2$. Desired solution is $w_1 = w_2 = 1$, although this problem have additional undesired solutions.

This gradient can be particular problematic when considering that $E[z_{h_{\ell-1}}] = 0$ is a desired property when initializing [**?**]. An alternative multiplication operator must thus be able to not explode for $z_{h_{\ell-1}}$ near zero. To that end we propose a new neural multplication units (NMU):

$$
\begin{aligned}
W_{h_{\ell-1},h_\ell} &= \min(\max(\hat{W}_{h_{\ell-1},h_\ell}, 0), 1), \\
\mathcal{R}_{\ell,\text{bias}} &= \frac{1}{H_\ell + H_{\ell-1}} \sum_{h_\ell=1}^{H_\ell} \sum_{h_{\ell-1}=1}^{H_{\ell-1}} \hat{W}^2_{h_{\ell-1},h_\ell}(1 - \hat{W}_{h_{\ell-1},h_\ell})^2 \\
\mathcal{R}_{\ell,\text{oob}} &= \frac{1}{H_\ell + H_{\ell-1}} \sum_{h_\ell=1}^{H_\ell} \sum_{h_{\ell-1}=1}^{H_{\ell-1}} \max\left(\left|\hat{W}_{h_{\ell-1},h_\ell} - \frac{1}{2}\right| - \frac{1}{2}, 0\right)^2 \\
\text{NMU} : z_{h_\ell} &= \prod_{h_{\ell-1}=1}^{H_{\ell-1}} \left(W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}\right)
\end{aligned}
\tag{8}
$$

Notable is the multiplicative identity for when $W_{h_{\ell-1},h_\ell} = 0$. This unit does not support division, but supporting division is likely infeasible as dividing by $z_{h_{\ell-1}}$ near zero would cause explosions. As shown in [**?**], experiments using the NALU for division does not work well hence very little is lost with this modification. As opposed to the NALU, the NMU can represent input of both negative and positive $z_{h_{\ell-1}}$ values and is not ϵ dependent, which allows the NMU to extrapolate inputs that are negative or smaller than ϵ.

The gradients with respect to the weight and input in the NMU are (see details in Appendix A.3):

$$
\begin{aligned}
\frac{\partial \mathcal{L}}{\partial W_{h_\ell,h_{\ell-1}}} &= \frac{\partial \mathcal{L}}{\partial z_{h_\ell}} \frac{\partial z_{h_\ell}}{\partial W_{h_\ell,h_{\ell-1}}} = \frac{\partial \mathcal{L}}{\partial z_{h_\ell}} \frac{z_{h_\ell}}{W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}} \left(z_{h_{\ell-1}} - 1\right) \\
\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}} &= \sum_{h_\ell=1}^{H_\ell} \frac{\partial \mathcal{L}}{\partial z_{h_\ell}} \frac{\partial z_{h_\ell}}{\partial z_{h_{\ell-1}}} = \sum_{h_\ell=1}^{H_\ell} \frac{z_{h_\ell}}{W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}} W_{h_{\ell-1},h_\ell}
\end{aligned}
\tag{9}
$$

Note that the fraction does not explode for $z_{h_{\ell-1}}$ close to zero, as the denominator simply cancels out a term in z_{h_ℓ}.

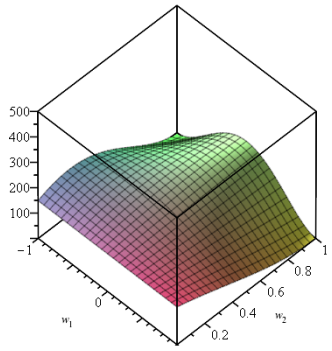


Figure 3: RMS loss curvature (without regularization) for a NAC_+ layer followed by an NMU layer. Otherwise, the setup is identical to that in Figure 2.

2.1.3 Moments and initialization

Initialization is important to consider for fast and consistent convergence [**?**].

Our proposed NAU, can be initialize using Glorot initialzation as it is a linear layer. The NAC_+ unit can also achieve an ideal initialization, although it is less trivial (details in Appendix B.1).

Using second order multivariate Taylor approximation and some assumptions of uncorrelated stochastic variables, the expectation of $\text{NAC}_\bullet$ can be estimated to be

$$
E[z_{h_\ell}] \approx \left(1 + \frac{1}{2} Var[W_{h_\ell,h_{\ell-1}}] \log(|E[z_{h_{\ell-1}}]| + \epsilon)^2\right)^{H_{\ell-1}} \Rightarrow E[z_{h_\ell}] > 1 \tag{10}
$$

(proof in Appendix B.2). An ideal initialization should satisfy $E[z_{h_\ell}] = 0$ [**?**], which the expectation for $\mathrm{NAC}_\bullet$ is infeasible.

Our proposed NMU when initialized with $E[W_{h_\ell,h_{\ell-1}}] = 1/2$ has an expectation of

$$E[z_{h_\ell}] \approx \left(\frac{1}{2}\right)^{H_{\ell-1}} \tag{11}$$

which approaches zero for $H_{\ell-1} \to \infty$ (proof in Appendix B.3).

The $\mathrm{NAC}_\bullet$ can not be input-independent initialization and has an exploding variance in depth (proof in Appendix B.2 and B.3). The NMU can, with the assumption that, $Var[z_{h_{\ell-1}}] = 1$ and $H_{\ell-1}$ is large, be intitialized optimally with $Var[W_{h_{\ell-1},h_\ell}] = \frac{1}{4}$ (see proof in Appendix B.3.3).

3 Experimental results

3.1 Arithmetic datasets

The arithmetic dataset is a replica of the "simple function task" shown in [**?**]. The goal is to sum two subsets of a vector and perform a arithmetic operation as defined below

$$t = \sum_{i=a_{\text{start}}}^{a_{\text{end}}} \mathbf{x}_i \circ \sum_{i=b_{\text{start}}}^{b_{\text{end}}} \mathbf{x}_i \quad \text{where } \mathbf{x} \in \mathbb{R}^n, x_i \sim \text{Uniform}[r_{\text{lower}}, r_{\text{upper}}], \circ \in \{+, -, \times\} \tag{12}$$

where n, r_{lower}, r_{upper}, $\circ$, the subset size and subset overlap are dataset parameters that we use to test the models ability to learn. We define a set of default parameters, see table (table 1). When probing a specific dataset parameter, e.g. subset overlap, the default will be the used for the remaining parameters.

Table 1: Default dataset parameters

Parameter name	Default value
Input size	100
Subset ratio	0.25
Overlap ratio	0.5
Interpolation range	$U[1, 2]$
Extrapolation range	$U[2, 6]$

3.1.1 Criterion

The goal is to achieve a solution that is acceptably close to a perfect solution. To evaluate if a model instance solves the task, the MSE is compared to a known nearly-perfect solution on the extrapolation range.

If $\mathbf{W}_1, \mathbf{W}_2$ defines the weights of the fitted model, and $\mathbf{W}_1^\epsilon$ is nearly-perfect and $\mathbf{W}_2^*$ is perfect (example in equation 13), the success criteria is $\mathcal{L}_{\mathbf{W}_1,\mathbf{W}_2} < \mathcal{L}_{\mathbf{W}_1^\epsilon,\mathbf{W}_2^*}$, measured on the extrapolation error. Meaning the MSE for the fitted model, should be less than the MSE for a nearly perfect solution.

$$\mathbf{W}_1^\epsilon = \begin{bmatrix} 1-\epsilon & 1-\epsilon & 0+\epsilon & 0+\epsilon \\ 1-\epsilon & 1-\epsilon & 1-\epsilon & 1-\epsilon \end{bmatrix}, \mathbf{W}_2^* = [1 \quad 1] \tag{13}$$

All experiments are evaluated multiple times with different seeds. We define the success rate as the percentage of experiments that achives success.

A sparsity error is also reported, the is defined in equation 14. This is only considered for model instances that did solve the task.

$$E_{\text{sparsity}} = \max_{h_{\ell-1},h_\ell} \min(|W_{h_{\ell-1},h_\ell}|, |1 - |W_{h_{\ell-1},h_\ell}||) \tag{14}$$

The first iteration for which $\mathcal{L}_{\mathbf{W}_1,\mathbf{W}_2} < \mathcal{L}_{\mathbf{W}_1^\epsilon,\mathbf{W}_2^*}$, is also reported. Again, only model instances that did solve the task are considered.

3.1.2 Model setup

To solve the task, we compare the models defined in table 2.

Table 2: Model defintions

Model	Layer 1	Layer 2
NMU	NAU	NMU
NAU	NAU	NAU
$\mathrm{NAC}_\bullet$	NAC_+	$\mathrm{NAC}_\bullet$
NAC_+	NAC_+	NAC_+
NALU	NALU	NALU
Linear	Linear	Linear
ReLU	ReLU	ReLU

For all experiments $\lambda_{\mathrm{oob}} = 1$ and $\lambda_{\mathrm{bias}} = 0.1 \cdot (1 - \exp(-10^5 \cdot t))$. Gradually scaling the bias regularizer $\mathcal{R}_{\ell,\mathrm{bias}}$ is to ensure it does not interfere with early training.

For all experiments Adam optimization [**?**] with default parameters is used.

The training dataset is continuously sampled from the interpolation range, a different seed is used for each experiment. Training is done with a mini-batch size of 128 observations.

A fixed validation dataset with 10000 observations is sampled from the interpolation range. A fixed test dataset with 10000 observations is sample from the extrapolation range.

Validation error, test error and sparsity error is sampled every 1000 iterations. To avoid noise from exploration, the best fit in terms of the validation error among the last 100 samples is used.

3.1.3 Very simple function

To empirically validate the theoretical challenges with $\mathrm{NAC}_\bullet$ consider the very simple problem shown earlier in figure 2. That is, $t = (x_1 + x_2) \circ (x_1 + x_2 + x_3 + x_4)$ for $x \in \mathbb{R}^4$.

Each experiment is conducted 100 times with different seeds, and stopped after 200000 iterations.

The results, in table 3, show that NMU has a higher success rate and converges faster. When inspecting the 6% that did not converge, we found the issue to underflow when $w = 0$ in the NMU layer.

Table 3: Shows the success-rate for $\mathcal{L}_{\mathbf{W}_1,\mathbf{W}_2} < \mathcal{L}_{\mathbf{W}_1^\epsilon,\mathbf{W}_2^*}$, at what global step the model converged at, and the sparsity error for all weight matrices.

Operation	Model	Success	Converged at		Sparsity error
		Rate	Median	Mean	Mean
$\times$	$\mathrm{NAC}_\bullet$	13%	$4.1 \cdot 10^4$	$4.4 \cdot 10^4 \pm 6.6 \cdot 10^3$	$7.5 \cdot 10^{-6} \pm 2.0 \cdot 10^{-6}$
	NALU	26%	$4.7 \cdot 10^4$	$5.4 \cdot 10^4 \pm 8.2 \cdot 10^3$	$9.2 \cdot 10^{-6} \pm 1.7 \cdot 10^{-6}$
	NMU	94%	$1.3 \cdot 10^4$	$1.7 \cdot 10^4 \pm 3.3 \cdot 10^3$	$5.2 \cdot 10^{-5} \pm 4.0 \cdot 10^{-5}$

3.1.4 Arithmetic operation comparison

We compare the models on different arithmetic operation $\circ \in \{+, -, \times\}$ used in equation 12, results are seen in 4.

Each experiment is trained for $5 \cdot 10^6$ iterations.

For multiplication, the NMU success more often and converges faster. For subtraction, the NAU model converges almost 3 orders of magnitude faster. For addition, .

add median to table (DONE) and discuss.

Table 4: Shows the success-rate for $\mathcal{L}_{\mathbf{W}_1,\mathbf{W}_2} < \mathcal{L}_{\mathbf{W}_1^\epsilon,\mathbf{W}_2^*}$, at what global step the model converged at, and the sparsity error for all weight matrices.

Operation	Model	Success	Converged at		Sparsity error
		Rate	Median	Mean	Mean
$-$	NAC_+	100%	$8.0 \cdot 10^3$	$1.5 \cdot 10^6 \pm 1.5 \cdot 10^6$	$4.6 \cdot 10^{-1} \pm 2.9 \cdot 10^{-2}$
	Linear	100%	$1.1 \cdot 10^6$	$1.9 \cdot 10^6 \pm 1.3 \cdot 10^6$	$3.7 \cdot 10^{-1} \pm 1.1 \cdot 10^{-1}$
	NALU	20%	$3.6 \cdot 10^6$	$3.6 \cdot 10^6 \pm 1.3 \cdot 10^7$	$4.7 \cdot 10^{-1} \pm 3.3 \cdot 10^{-1}$
	NAU	100%	$4.0 \cdot 10^3$	$4.2 \cdot 10^3 \pm 3.0 \cdot 10^2$	$1.9 \cdot 10^{-3} \pm 4.2 \cdot 10^{-4}$
	ReLU	0%	—	—	—
$\times$	$\mathrm{NAC}_\bullet$	30%	$2.5 \cdot 10^6$	$2.5 \cdot 10^6 \pm 1.5 \cdot 10^6$	$3.9 \cdot 10^{-4} \pm 9.4 \cdot 10^{-4}$
	Linear	0%	—	—	—
	NALU	0%	—	—	—
	NMU	90%	$1.4 \cdot 10^6$	$1.6 \cdot 10^6 \pm 5.6 \cdot 10^5$	$1.8 \cdot 10^{-3} \pm 1.1 \cdot 10^{-3}$
	ReLU	0%	—	—	—
$+$	NAC_+	100%	$6.0 \cdot 10^4$	$7.1 \cdot 10^4 \pm 2.4 \cdot 10^4$	$4.8 \cdot 10^{-1} \pm 2.0 \cdot 10^{-2}$
	Linear	100%	$4.2 \cdot 10^4$	$4.2 \cdot 10^4 \pm 1.9 \cdot 10^3$	$6.1 \cdot 10^{-1} \pm 1.2 \cdot 10^{-1}$
	NALU	0%	—	—	—
	NAU	100%	$1.8 \cdot 10^4$	$7.0 \cdot 10^5 \pm 9.2 \cdot 10^5$	$1.7 \cdot 10^{-3} \pm 8.0 \cdot 10^{-4}$
	ReLU	80%	$4.2 \cdot 10^4$	$8.4 \cdot 10^5 \pm 1.1 \cdot 10^6$	$7.3 \cdot 10^{-1} \pm 2.3 \cdot 10^{-1}$

3.1.5 Exploration of dataset parameters

Finally, the parameters from which the dataset is constructed are considered for just the multiplication problem ($a \cdot b$). The setup is the same the results from table 4. The results are visualized in in figure 4, 5, 8, and 7. Errors bars show the upper and lower 10% quantile, computed over 10 different seeds for each configuration. The center shows the mean of those 10 observations.

Generally the NMU performs far better than both $\mathrm{NAC}_\bullet$ and NALU. Some important observations to make:

- Input size > 100. The NMU model's success-rate very suddenly decreases when the input size is greater than 100. We have been unable to explain why this happens. We suspect it is a problem with the signal-to-noie ratio of the problem. However the result is also seen if the mini-batch size is dramatically increased.

- Overlap ratio $= 0$: Both the NMU and also the $\mathrm{NAC}_\bullet$ when it does converge, finds a suboptimal solution where in the addition layer $w = 1$ for the overlapping input between a and b, and $w = 0$ for where the input isn't used. However when an input-scalar is only used in either a or b, convergence the corresponding weights is difficult and slow. Thus the lower the overlap ratio is, the harder the problem is.

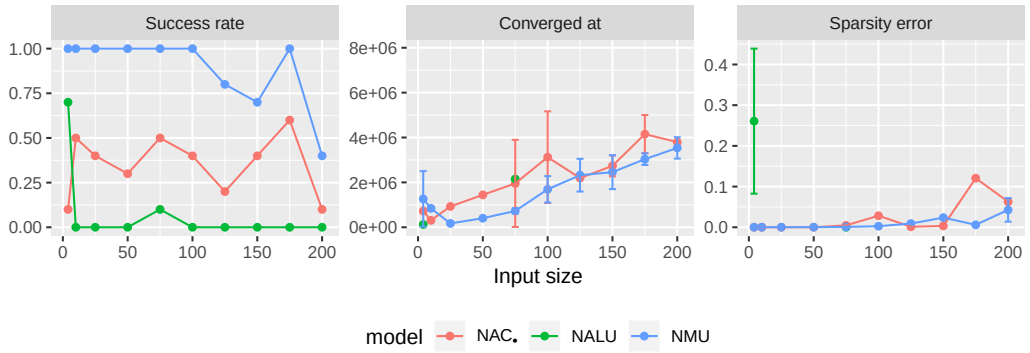


Figure 4: Shows the effect of the input size, on the simple function task problem.

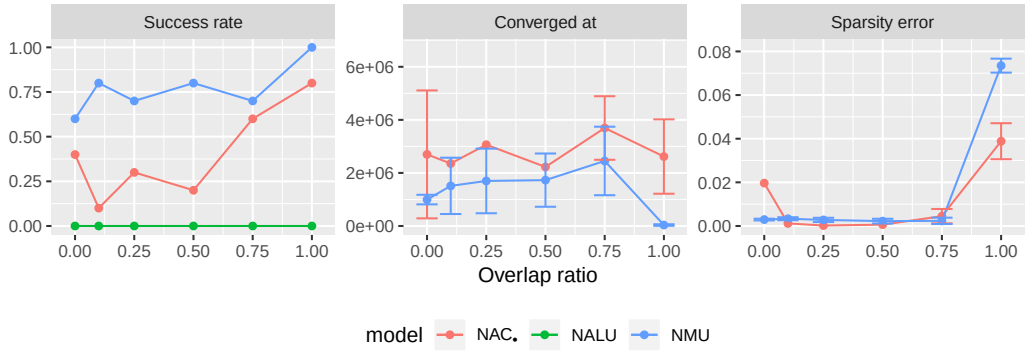


Figure 5: Shows the effect of the overlap ratio, on the simple function task problem.

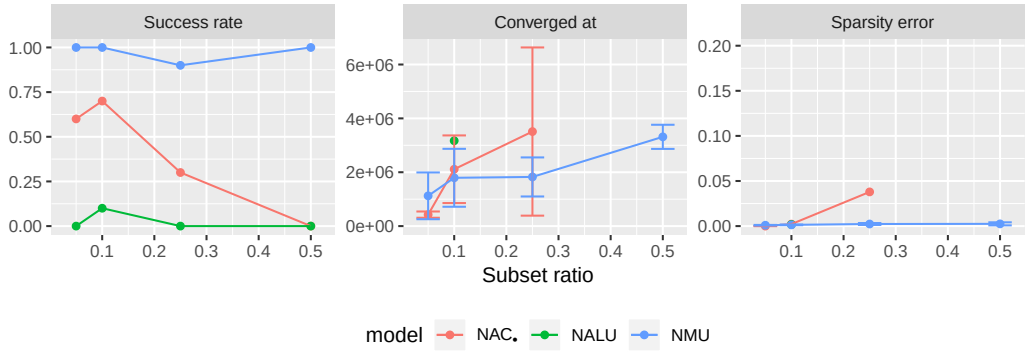


Figure 6: Shows the effect of the subset ratio, on the simple function task problem.

4 Related work

- Lorem Ipsum

5 Conclusion

Acknowledgments

We would like to thank Andrew Trask and the other authors of the NALU paper, for highlighting the importance and challenges of etrapolation in Neural Networks. We would also like to thank

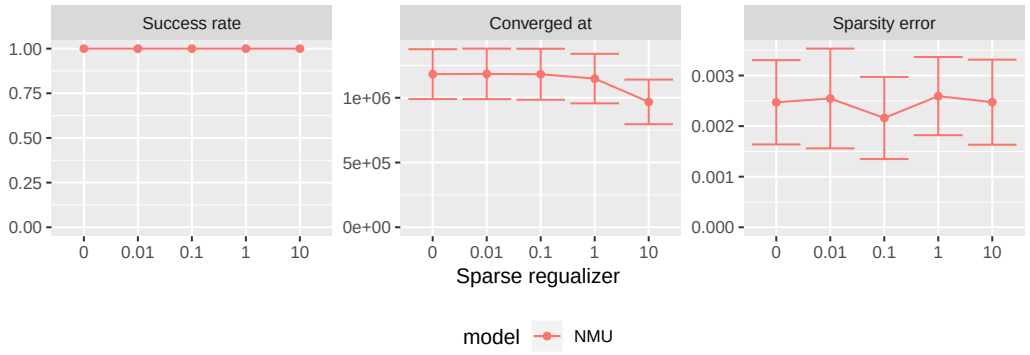


Figure 7: Shows the effect of the subset ratio, on the simple function task problem.

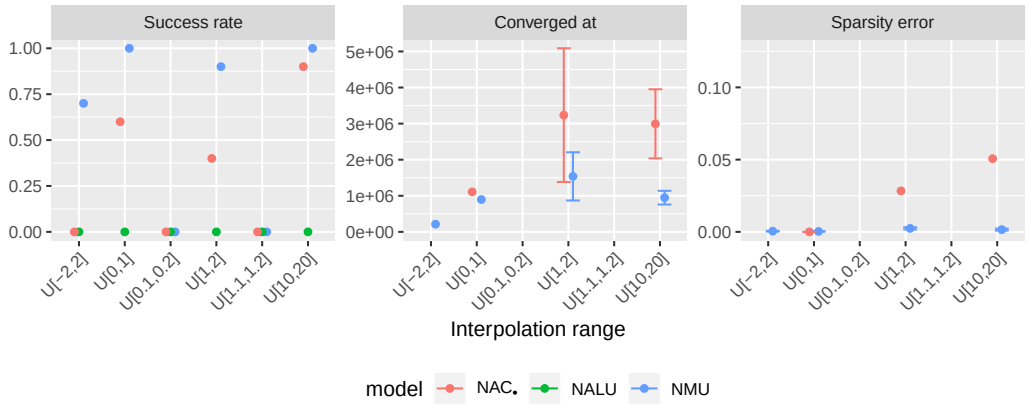


Figure 8: Shows the effect of the interpolation range. For each interpolation range, the following extrapolation ranges are used: $\mathrm{U}[-2, 2] \to \mathrm{U}[-6, 6]$, $\mathrm{U}[0, 1] \to \mathrm{U}[0, 5]$, $\mathrm{U}[0.1, 0.2] \to \mathrm{U}[0, 2]$, $\mathrm{U}[1, 2] \to \mathrm{U}[1, 6]$, $\mathrm{U}[10, 20] \to \mathrm{U}[1, 40]$.

the students Raja Shan Zaker Kreen and William Frisch Moller from The Technical University of Denmark, who showed us that the NALU does not converge consistently.

A Gradient derivatives

A.1 Weight matrix construction

For clarity the weight matrix construction is defined using scalar notation

$$W_{h_\ell,h_{\ell-1}} = \tanh(\hat{W}_{h_\ell,h_{\ell-1}})\sigma(\hat{M}_{h_\ell,h_{\ell-1}}) \tag{15}$$

The of the loss with respect to $\hat{W}_{h_\ell,h_{\ell-1}}$ and $\hat{M}_{h_\ell,h_{\ell-1}}$ is then straight forward to derive.

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \hat{W}_{h_\ell,h_{\ell-1}}} &= \frac{\partial \mathcal{L}}{\partial W_{h_\ell,h_{\ell-1}}}\frac{\partial W_{h_\ell,h_{\ell-1}}}{\partial \hat{W}_{h_\ell,h_{\ell-1}}} \\
&= \frac{\partial \mathcal{L}}{\partial W_{h_\ell,h_{\ell-1}}}(1-\tanh^2(\hat{W}_{h_\ell,h_{\ell-1}}))\sigma(\hat{M}_{h_\ell,h_{\ell-1}}) \\
\frac{\partial \mathcal{L}}{\partial \hat{M}_{h_\ell,h_{\ell-1}}} &= \frac{\partial \mathcal{L}}{\partial W_{h_\ell,h_{\ell-1}}}\frac{\partial W_{h_\ell,h_{\ell-1}}}{\partial \hat{M}_{h_\ell,h_{\ell-1}}} \\
&= \frac{\partial \mathcal{L}}{\partial W_{h_\ell,h_{\ell-1}}}\tanh(\hat{W}_{h_\ell,h_{\ell-1}})\sigma(\hat{M}_{h_\ell,h_{\ell-1}})(1-\sigma(\hat{M}_{h_\ell,h_{\ell-1}}))
\end{aligned} \tag{16}$$

As seen from this result, one only needs to consider $\frac{\partial \mathcal{L}}{\partial W_{h_\ell,h_{\ell-1}}}$ for NAC_+ and $\mathrm{NAC}_\bullet$, as the gradient with respect to $\hat{W}_{h_\ell,h_{\ell-1}}$ and $\hat{M}_{h_\ell,h_{\ell-1}}$ is just a multiplication on $\frac{\partial \mathcal{L}}{\partial W_{h_\ell,h_{\ell-1}}}$.

A.2 Gradient of $\mathrm{NAC}_\bullet$

First the $\mathrm{NAC}_\bullet$ is defined using scalar notation.

$$z_{h_\ell} = \exp\left(\sum_{h_{\ell-1}=1}^{H_{\ell-1}} W_{h_\ell,h_{\ell-1}}\log(|z_{h_{\ell-1}}|+\epsilon)\right) \tag{17}$$

The gradient of the loss with respect to $W_{h_\ell,h_{\ell-1}}$ is straight forward to derive.

$$\begin{aligned}
\frac{\partial z_{h_\ell}}{\partial W_{h_\ell,h_{\ell-1}}} &= \exp\left(\sum_{h'_{\ell-1}=1}^{H_{\ell-1}} W_{h_\ell,h'_{\ell-1}}\log(|z_{h'_{\ell-1}}|+\epsilon)\right)\log(|z_{h_{\ell-1}}|+\epsilon) \\
&= z_{h_\ell}\log(|z_{h_{\ell-1}}|+\epsilon)
\end{aligned} \tag{18}$$

We now wish to derive the backpropergation term $\delta_{h_\ell} = \frac{\partial \mathcal{L}}{\partial z_{h_\ell}}$, because z_{h_ℓ} affects $\{z_{h_{\ell+1}}\}_{h_{\ell+1}=1}^{H_{\ell+1}}$ this becomes:

$$\delta_{h_\ell} = \frac{\partial \mathcal{L}}{\partial z_{h_\ell}} = \sum_{h_{\ell+1}=1}^{H_{\ell+1}}\frac{\partial \mathcal{L}}{\partial z_{h_{\ell+1}}}\frac{\partial z_{h_{\ell+1}}}{\partial z_{h_\ell}} = \sum_{h_{\ell+1}=1}^{H_{\ell+1}}\delta_{h_{\ell+1}}\frac{\partial z_{h_{\ell+1}}}{\partial z_{h_\ell}} \tag{19}$$

To make it easier to derive $\frac{\partial z_{h_{\ell+1}}}{\partial z_{h_\ell}}$ we re-express the z_{h_ℓ} as $z_{h_{\ell+1}}$.

$$z_{h_{\ell+1}} = \exp\left(\sum_{h_\ell=1}^{H_\ell} W_{h_{\ell+1},h_\ell}\log(|z_{h_\ell}|+\epsilon)\right) \tag{20}$$

The gradient of $\frac{\partial z_{h_{\ell+1}}}{\partial z_{h_\ell}}$ is then:

$$\begin{aligned}\frac{\partial z_{h_{\ell+1}}}{\partial z_{h_\ell}} &= \exp\left(\sum_{h_\ell=1}^{H_\ell} W_{h_{\ell+1},h_\ell} \log(|z_{h_\ell}| + \epsilon)\right) W_{h_{\ell+1},h_\ell} \frac{\partial \log(|z_{h_\ell}| + \epsilon)}{\partial z_{h_\ell}} \\ &= \exp\left(\sum_{h_\ell=1}^{H_\ell} W_{h_{\ell+1},h_\ell} \log(|z_{h_\ell}| + \epsilon)\right) W_{h_{\ell+1},h_\ell} \frac{\mathrm{abs}'(z_{h_\ell})}{|z_{h_\ell}| + \epsilon} \\ &= m_{h_{\ell+1}} W_{h_{\ell+1},h_\ell} \frac{\mathrm{abs}'(z_{h_\ell})}{|z_{h_\ell}| + \epsilon}\end{aligned} \tag{21}$$

$\mathrm{abs}'(z_{h_\ell})$ is the gradient of the absolute function. In the paper we denote this as $\mathrm{sign}(z_{h_\ell})$ for brevity. However, depending on the exact defintion used there may be a difference for $z_{h_\ell} = 0$, as $\mathrm{abs}'(0)$ is undefined. In practicality this doesn't matter much though, although theoretically it does mean that the expectation of this is theoretically undefined when $E[z_{h_\ell}] = 0$.

A.3 Gradient of NMU

In scalar notation the NMU is defined as:

$$z_{h_\ell} = \prod_{h_{\ell-1}=1}^{H_{\ell-1}} \left(W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}\right) \tag{22}$$

The gradient of the loss with respect to $W_{h_{\ell-1},h_\ell}$ is fairly trivial. Note that every term but the one for $h_{\ell-1}$, is just a constant with respect to $W_{h_{\ell-1},h_\ell}$. The product, expect the term for $h_{\ell-1}$ can be expressed as $\frac{z_{h_\ell}}{W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}}$. Using this fact, it becomes trivial to derive the gradient as:

$$\frac{\partial \mathcal{L}}{\partial w_{h_\ell,h_{\ell-1}}} = \frac{\partial \mathcal{L}}{\partial z_{h_\ell}} \frac{\partial z_{h_\ell}}{\partial w_{h_\ell,h_{\ell-1}}} = \frac{\partial \mathcal{L}}{\partial z_{h_\ell}} \frac{z_{h_\ell}}{W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}} \left(z_{h_{\ell-1}} - 1\right) \tag{23}$$

Similarly, the gradient $\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}$ which is essential in backpropergation can equally easily be derived as:

$$\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}} = \sum_{h_\ell=1}^{H_\ell} \frac{\partial \mathcal{L}}{\partial z_{h_\ell}} \frac{\partial z_{h_\ell}}{\partial z_{h_{\ell-1}}} = \sum_{h_\ell=1}^{H_\ell} \frac{z_{h_\ell}}{W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}} W_{h_{\ell-1},h_\ell} \tag{24}$$

B Moments

B.1 Expectation and variance for weight matrix construction in NAC layers

The weight matrix construction in NAC, is defined in scalar notation as:

$$W_{h_\ell,h_{\ell-1}} = \tanh(\hat{W}_{h_\ell,h_{\ell-1}}) \sigma(\hat{M}_{h_\ell,h_{\ell-1}}) \tag{25}$$

Simplifying the notation of this, and re-expressing it using stochastic variables with uniform distributions this can be written as:

$$\begin{aligned} W &\sim \tanh(\hat{W}) \sigma(\hat{M}) \\ \hat{W} &\sim U[-r, r] \\ \hat{M} &\sim U[-r, r] \end{aligned} \tag{26}$$

Since $\tanh(\hat{W})$ is an odd-function and $E[\hat{W}] = 0$, deriving the expectation $E[W]$ is trivial.

$$\mathrm{E}[W] = \mathrm{E}[\tanh(\hat{W})]\mathrm{E}[\sigma(\hat{M})] = 0 \cdot \mathrm{E}[\sigma(\hat{M})] = 0 \tag{27}$$

The variance is more complicated, however as $\hat{W}$ and $\hat{M}$ are independent, it can be simplified to:

$$\mathrm{Var}[W] = \mathrm{E}[\tanh(\hat{W})^2]\mathrm{E}[\sigma(\hat{M})^2] - \mathrm{E}[\tanh(\hat{W})]^2\mathrm{E}[\sigma(\hat{M})]^2 = \mathrm{E}[\tanh(\hat{W})^2]\mathrm{E}[\sigma(\hat{M})^2] \tag{28}$$

These second moments can be analyzed independently. First for $\mathrm{E}[\tanh(\hat{W})^2]$:

$$\begin{aligned}
\mathrm{E}[\tanh(\hat{W})^2] &= \int_{-\infty}^{\infty} \tanh(x)^2 f_{U[-r,r]}(x)\,\mathrm{d}x \\
&= \frac{1}{2r}\int_{-r}^{r} \tanh(x)^2\,\mathrm{d}x \\
&= \frac{1}{2r}\cdot 2\cdot(r-\tanh(r)) \\
&= 1-\frac{\tanh(r)}{r}
\end{aligned} \tag{29}$$

Then for $\mathrm{E}[\tanh(\hat{M})^2]$:

$$\begin{aligned}
\mathrm{E}[\sigma(\hat{M})^2] &= \int_{-\infty}^{\infty} \sigma(x)^2 f_{U[-r,r]}(x)\,\mathrm{d}x \\
&= \frac{1}{2r}\int_{-r}^{r} \sigma(x)^2\,\mathrm{d}x \\
&= \frac{1}{2r}\left(r-\tanh\left(\frac{r}{2}\right)\right)
\end{aligned} \tag{30}$$

Finally this gives the variance:

$$\mathrm{Var}[W] = \frac{1}{2r}\left(1-\frac{\tanh(r)}{r}\right)\left(r-\tanh\left(\frac{r}{2}\right)\right) \tag{31}$$

B.2 Expectation and variance of $\mathrm{NAC}_\bullet$

B.2.1 Forward pass

Assuming that each $z_{h_{\ell-1}}$ are independent the expectation can be simplified to:

$$\begin{aligned}
E[z_{h_\ell}] &= E\left[\exp\left(\sum_{h_{\ell-1}=1}^{H_{\ell-1}} W_{h_\ell,h_{\ell-1}}\log(|z_{h_{\ell-1}}|+\epsilon)\right)\right] \\
&= E\left[\prod_{h_{\ell-1}=1}^{H_{\ell-1}} \exp(W_{h_\ell,h_{\ell-1}}\log(|z_{h_{\ell-1}}|+\epsilon))\right] \\
&= \prod_{h_{\ell-1}=1}^{H_{\ell-1}} E[\exp(W_{h_\ell,h_{\ell-1}}\log(|z_{h_{\ell-1}}|+\epsilon))] \\
&= E[\exp(W_{h_\ell,h_{\ell-1}}\log(|z_{h_{\ell-1}}|+\epsilon))]^{H_{\ell-1}} \\
&= E\left[(|z_{h_{\ell-1}}|+\epsilon)^{W_{h_\ell,h_{\ell-1}}}\right]^{H_{\ell-1}} \\
&= E\left[f(z_{h_{\ell-1}},W_{h_\ell,h_{\ell-1}})\right]^{H_{\ell-1}}
\end{aligned} \tag{32}$$

Here we define f as a non-linear transformation function of two independent stocastic variables:

$$f(z_{h_{\ell-1}},W_{h_\ell,h_{\ell-1}}) = (|z_{h_{\ell-1}}|+\epsilon)^{W_{h_\ell,h_{\ell-1}}} \tag{33}$$

We then take the second order taylor approximation of f, around $(E[z_{h_{\ell-1}}], E[W_{h_\ell,h_{\ell-1}}])$.

$$
\begin{aligned}
&E[f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})] \approx E\Bigg[\\
&f(E[z_{h_{\ell-1}}], E[W_{h_\ell,h_{\ell-1}}]) \\
&+ \begin{bmatrix} z_{h_{\ell-1}} - E[z_{h_{\ell-1}}] \\ W_{h_\ell,h_{\ell-1}} - E[W_{h_\ell,h_{\ell-1}}] \end{bmatrix}^T \begin{bmatrix} \frac{\partial f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})}{\partial z_{h_{\ell-1}}} \\ \frac{\partial f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})}{\partial W_{h_\ell,h_{\ell-1}}} \end{bmatrix} \Bigg|_{\substack{z_{h_{\ell-1}} = E[z_{h_{\ell-1}}] \\ W_{h_\ell,h_{\ell-1}} = E[W_{h_\ell,h_{\ell-1}}]}} \\
&+ \frac{1}{2} \begin{bmatrix} z_{h_{\ell-1}} - E[z_{h_{\ell-1}}] \\ W_{h_\ell,h_{\ell-1}} - E[W_{h_\ell,h_{\ell-1}}] \end{bmatrix}^T \\
&\bullet \begin{bmatrix} \frac{\partial^2 f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})}{\partial^2 z_{h_{\ell-1}}} & \frac{\partial^2 f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})}{\partial z_{h_{\ell-1}} \partial W_{h_\ell,h_{\ell-1}}} \\ \frac{\partial^2 f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})}{\partial z_{h_{\ell-1}} \partial W_{h_\ell,h_{\ell-1}}} & \frac{\partial^2 f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})}{\partial^2 W_{h_\ell,h_{\ell-1}}} \end{bmatrix} \Bigg|_{\substack{z_{h_{\ell-1}} = E[z_{h_{\ell-1}}] \\ W_{h_\ell,h_{\ell-1}} = E[W_{h_\ell,h_{\ell-1}}]}} \\
&\bullet \begin{bmatrix} z_{h_{\ell-1}} - E[z_{h_{\ell-1}}] \\ W_{h_\ell,h_{\ell-1}} - E[W_{h_\ell,h_{\ell-1}}] \end{bmatrix} \Bigg]
\end{aligned} \tag{34}
$$

Because $E[z_{h_{\ell-1}} - E[z_{h_{\ell-1}}]] = 0$, $E[W_{h_\ell,h_{\ell-1}} - E[W_{h_\ell,h_{\ell-1}}]] = 0$, and $Cov[z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}}] = 0$. This similifies to:

$$
\begin{aligned}
&E[f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})] \approx f(E[z_{h_{\ell-1}}], E[W_{h_\ell,h_{\ell-1}}]) \\
&+ \frac{1}{2} Var \begin{bmatrix} z_{h_{\ell-1}} \\ W_{h_\ell,h_{\ell-1}} \end{bmatrix}^T \begin{bmatrix} \frac{\partial^2 f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})}{\partial^2 z_{h_{\ell-1}}} \\ \frac{\partial^2 f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})}{\partial^2 W_{h_\ell,h_{\ell-1}}} \end{bmatrix} \Bigg|_{\substack{z_{h_{\ell-1}} = E[z_{h_{\ell-1}}] \\ W_{h_\ell,h_{\ell-1}} = E[W_{h_\ell,h_{\ell-1}}]}}
\end{aligned} \tag{35}
$$

Inserting the derivatives and computing the inner products yields:

$$
\begin{aligned}
&E[f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})] \approx (|E[z_{h_{\ell-1}}]| + \epsilon)^{E[W_{h_\ell,h_{\ell-1}}]} \\
&+ \frac{1}{2} Var[z_{h_{\ell-1}}] (|E[z_{h_{\ell-1}}]| + \epsilon)^{E[W_{h_\ell,h_{\ell-1}}]-2} E[W_{h_\ell,h_{\ell-1}}](E[W_{h_\ell,h_{\ell-1}}] - 1) \\
&+ \frac{1}{2} Var[W_{h_\ell,h_{\ell-1}}] (|E[z_{h_{\ell-1}}]| + \epsilon)^{E[W_{h_\ell,h_{\ell-1}}]} \log(|E[z_{h_{\ell-1}}]| + \epsilon)^2 \\
&= 1 + \frac{1}{2} Var[W_{h_\ell,h_{\ell-1}}] \log(|E[z_{h_{\ell-1}}]| + \epsilon)^2
\end{aligned} \tag{36}
$$

This gives the final expectation:

$$
\begin{aligned}
E[z_{h_\ell}] &= E\left[f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})\right]^{H_{\ell-1}} \\
&\approx \left(1 + \frac{1}{2} Var[W_{h_\ell,h_{\ell-1}}] \log(|E[z_{h_{\ell-1}}]| + \epsilon)^2\right)^{H_{\ell-1}}
\end{aligned} \tag{37}
$$

As this expectation is of particular interrest, we evaluate the error of the approximation, where $W_{h_\ell,h_{\ell-1}} \sim U[-r_w, r_w]$ and $z_{h_{\ell-1}} \sim U[0, r_z]$. These distributions are what is used in the simple function task is done. The result is seen in figure 9.

The variance can be derived using the same assumptions about expectation and no correlation.

$$
\begin{aligned}
Var[z_{h_\ell}] &= E[z_{h_\ell}^2] - E[z_{h_\ell}]^2 \\
&= E\left[\prod_{h_{\ell-1}=1}^{H_{\ell-1}} (|z_{h_{\ell-1}}| + \epsilon)^{2 \cdot W_{h_\ell,h_{\ell-1}}}\right] - E\left[\prod_{h_{\ell-1}=1}^{H_{\ell-1}} (|z_{h_{\ell-1}}| + \epsilon)^{W_{h_\ell,h_{\ell-1}}}\right]^2 \\
&= E\left[f(z_{h_{\ell-1}}, 2 \cdot W_{h_\ell,h_{\ell-1}})\right]^{H_{\ell-1}} - E\left[f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})\right]^{2 \cdot H_{\ell-1}}
\end{aligned} \tag{38}
$$

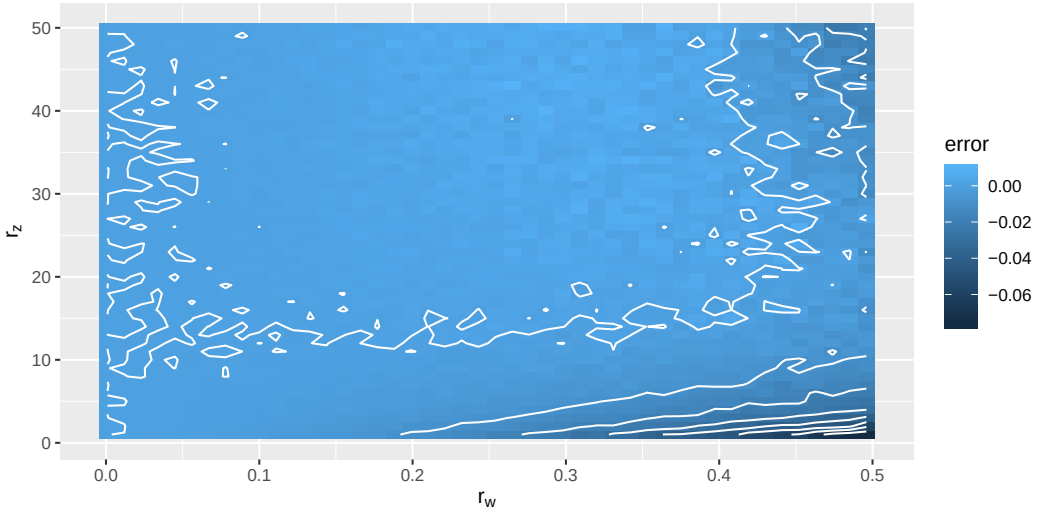


Figure 9: Error between theoretical approximation and the numerical approximation estimated by random sampling of 100000 observations at each combination of r_z and r_w.

We already have from the expectation result that:

$$E\left[f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})\right] \approx 1 + \frac{1}{2} Var[W_{h_\ell,h_{\ell-1}}] \log(|E[z_{h_{\ell-1}}]| + \epsilon)^2 \tag{39}$$

By substitution of variable we have that:

$$\begin{aligned} E\left[f(z_{h_{\ell-1}}, 2 \cdot W_{h_\ell,h_{\ell-1}})\right] &\approx 1 + \frac{1}{2} Var[2 \cdot W_{h_\ell,h_{\ell-1}}] \log(|E[z_{h_{\ell-1}}]| + \epsilon)^2 \\ &=\approx 1 + 2 \cdot Var[W_{h_\ell,h_{\ell-1}}] \log(|E[z_{h_{\ell-1}}]| + \epsilon)^2 \end{aligned} \tag{40}$$

This gives the variance:

$$\begin{aligned} Var[z_{h_\ell}] &= E\left[f(z_{h_{\ell-1}}, 2 \cdot W_{h_\ell,h_{\ell-1}})\right]^{H_{\ell-1}} - E\left[f(z_{h_{\ell-1}}, W_{h_\ell,h_{\ell-1}})\right]^{2 \cdot H_{\ell-1}} \\ &\approx \left(1 + 2 \cdot Var[W_{h_\ell,h_{\ell-1}}] \log(|E[z_{h_{\ell-1}}]| + \epsilon)^2\right)^{H_{\ell-1}} \\ &- \left(1 + \frac{1}{2} \cdot Var[W_{h_\ell,h_{\ell-1}}] \log(|E[z_{h_{\ell-1}}]| + \epsilon)^2\right)^{2 \cdot H_{\ell-1}} \end{aligned} \tag{41}$$

B.2.2 Backward pass

The expectation of the backpropagation term:

$$E[\delta_{h_\ell}] = E\left[\sum_{h_{\ell+1}=1}^{H_{\ell+1}} \delta_{h_{\ell+1}} \frac{\partial z_{h_{\ell+1}}}{\partial z_{h_\ell}}\right] = H_{\ell+1} E[\delta_{h_{\ell+1}}] E\left[\frac{\partial z_{h_{\ell+1}}}{\partial z_{h_\ell}}\right] \tag{42}$$

Where we have that:

$$E\left[\frac{\partial z_{h_{\ell+1}}}{\partial z_{h_\ell}}\right] = E[h_{\ell+1}] E[W_{h_{\ell+1},h_\ell}] E\left[\frac{\text{abs}'(z_{h_\ell})}{|z| + \epsilon}\right] = E[m_{h_{\ell+1}}] \cdot 0 \cdot E\left[\frac{\text{abs}'(z_{h_\ell})}{|z| + \epsilon}\right] = 0 \tag{43}$$

Deriving the variance is more complicated as:

$$Var\left[\frac{\partial m_{h_{\ell+1}}}{\partial z_{h_\ell}}\right] = Var\left[m_{h_{\ell+1}} W_{h_{\ell+1},h_\ell} \frac{\text{abs}'(z_{h_\ell})}{|z_{h_\ell}| + \epsilon}\right] \tag{44}$$

Assuming independence between each term this can be simplified to as:

$$\begin{aligned}
Var\left[\frac{\partial z_{h_{\ell+1}}}{\partial z_{h_\ell}}\right] &= E[z_{h_{\ell+1}}^2]E[W_{h_{\ell+1},h_\ell}^2]E\left[\left(\frac{\text{abs}'(z_{h_\ell})}{|z_{h_\ell}|+\epsilon}\right)^2\right] \\
&\quad - E[z_{h_{\ell+1}}]^2E[W_{h_{\ell+1},h_\ell}]^2E\left[\frac{\text{abs}'(z_{h_\ell})}{|z_{h_\ell}|+\epsilon}\right]^2 \\
&= E[z_{h_{\ell+1}}^2]Var[W_{h_{\ell+1},h_\ell}]E\left[\left(\frac{\text{abs}'(z_{h_\ell})}{|z_{h_\ell}|+\epsilon}\right)^2\right] \\
&\quad - E[z_{h_{\ell+1}}]^2 \cdot 0 \cdot E\left[\frac{\text{abs}'(z_{h_\ell})}{|z_{h_\ell}|+\epsilon}\right]^2 \\
&= E[z_{h_{\ell+1}}^2]Var[W_{h_{\ell+1},h_\ell}]E\left[\left(\frac{\text{abs}'(z_{h_\ell})}{|z_{h_\ell}|+\epsilon}\right)^2\right]
\end{aligned} \tag{45}$$

Using Taylor approximation around $E[z_{h_\ell}]$ we have:

$$\begin{aligned}
E\left[\left(\frac{\text{abs}'(z_{h_\ell})}{|z|+\epsilon}\right)^2\right] &\approx \frac{1}{\left(|E[z_{h_\ell}]|+\epsilon\right)^2} + \frac{1}{2}\frac{6}{\left(|E[z_{h_\ell}]|+\epsilon\right)^4}Var[z_{h_\ell}] \\
&= \frac{1}{\left(|E[z_{h_\ell}]|+\epsilon\right)^2} + \frac{3}{\left(|E[z_{h_\ell}]|+\epsilon\right)^4}Var[z_{h_\ell}]
\end{aligned} \tag{46}$$

Also reusing the result for $E[z_{h_\ell}^2]$ from earlier the variance can be expressed as:

$$\begin{aligned}
Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}}\right] \approx Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right] H_\ell \ \left(1 + 2 \cdot Var[W_{h_\ell,h_{\ell-1}}]\log(|E[z_{h_{\ell-1}}]|+\epsilon)^2\right)^{H_{\ell-1}} \\
\cdot Var[W_{h_\ell,h_{\ell-1}}]\left(\frac{1}{\left(|E[z_{h_{\ell-1}}]|+\epsilon\right)^2} + \frac{3}{\left(|E[z_{h_{\ell-1}}]|+\epsilon\right)^4}Var[z_{h_{\ell-1}}]\right)
\end{aligned} \tag{47}$$

B.3 Expectation and variance of NMU

B.3.1 Forward pass

Assuming that each $z_{h_{\ell-1}}$ are independent, that $E[z_{h_{\ell-1}}] = 0$, and that $E[W_{h_{\ell-1},h_\ell}] = 1/2$ the expectation is:

$$\begin{aligned}
E[z_{h_\ell}] &\approx E\left[\prod_{h_{\ell-1}=1}^{H_{\ell-1}}\left(W_{h_{\ell-1},h_\ell}z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}\right)\right] \\
&\approx E\left[W_{h_{\ell-1},h_\ell}z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}\right]^{H_{\ell-1}} \\
&\approx \left(E[W_{h_{\ell-1},h_\ell}]E[z_{h_{\ell-1}}] + 1 - E[W_{h_{\ell-1},h_\ell}]\right)^{H_{\ell-1}} \\
&\approx \left(\frac{1}{2}\cdot 0 + 1 - \frac{1}{2}\right)^{H_{\ell-1}} \\
&\approx \left(\frac{1}{2}\right)^{H_{\ell-1}}
\end{aligned} \tag{48}$$

Using the same assumptions for the variance one gets:

$$
\begin{aligned}
Var[z_{h_\ell}] &= E[z_{h_\ell}^2] - E[z_{h_\ell}]^2 \\
&\approx E[z_{h_\ell}^2] - \left(\frac{1}{2}\right)^{2 \cdot H_{\ell-1}} \\
&\approx E\left[\prod_{h_{\ell-1}=1}^{H_{\ell-1}} \left(W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}\right)^2\right] - \left(\frac{1}{2}\right)^{2 \cdot H_{\ell-1}} \\
&\approx E[\left(W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}\right)^2]^{H_{\ell-1}} - \left(\frac{1}{2}\right)^{2 \cdot H_{\ell-1}} \\
&\approx \Big(E[W_{h_{\ell-1},h_\ell}^2]E[z_{h_{\ell-1}}^2] - 2E[W_{h_{\ell-1},h_\ell}^2]E[z_{h_{\ell-1}}] \\
&\quad + E[W_{h_{\ell-1},h_\ell}^2] + 2E[W_{h_{\ell-1},h_\ell}]E[z_{h_{\ell-1}}] \\
&\quad - 2E[W_{h_{\ell-1},h_\ell}] + 1\Big)^{H_{\ell-1}} - \left(\frac{1}{2}\right)^{2 \cdot H_{\ell-1}} \\
&\approx \Big(E[W_{h_{\ell-1},h_\ell}^2]E[z_{h_{\ell-1}}^2] + E[W_{h_{\ell-1},h_\ell}^2] \\
&\quad - 2E[W_{h_{\ell-1},h_\ell}] + 1\Big)^{H_{\ell-1}} - \left(\frac{1}{2}\right)^{2 \cdot H_{\ell-1}} \\
&= \Big(E[W_{h_{\ell-1},h_\ell}^2]\left(E[z_{h_{\ell-1}}^2] + 1\right)\Big)^{H_{\ell-1}} - \left(\frac{1}{2}\right)^{2 \cdot H_{\ell-1}} \\
&\approx \left(\left(Var[W_{h_{\ell-1},h_\ell}] + E[W_{h_{\ell-1},h_\ell}]^2\right)\left(Var[z_{h_{\ell-1}}] + 1\right)\right)^{H_{\ell-1}} - \left(\frac{1}{2}\right)^{2 \cdot H_{\ell-1}} \\
&= \left(Var[W_{h_{\ell-1},h_\ell}] + \frac{1}{4}\right)^{H_{\ell-1}} \left(Var[z_{h_{\ell-1}}] + 1\right)^{H_{\ell-1}} - \left(\frac{1}{2}\right)^{2 \cdot H_{\ell-1}}
\end{aligned}
\tag{49}
$$

B.3.2 Backward pass

For the backward pass the expectation can using the same assumptions be derived to:

$$
\begin{aligned}
E\left[\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}}\right] &= H_\ell E\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}} \frac{\partial z_{h_\ell}}{\partial z_{h_{\ell-1}}}\right] \\
&= H_\ell E\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right] E\left[\frac{\partial z_{h_\ell}}{\partial z_{h_{\ell-1}}}\right] \\
&= H_\ell E\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right] E\left[\frac{z_{h_\ell}}{W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}} W_{h_{\ell-1},h_\ell}\right] \\
&= H_\ell E\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right] E\left[\frac{z_{h_\ell}}{W_{h_{\ell-1},h_\ell} z_{h_{\ell-1}} + 1 - W_{h_{\ell-1},h_\ell}}\right] E\left[W_{h_{\ell-1},h_\ell}\right] \\
&= H_\ell E\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right] \left(\frac{1}{2}\right)^{H_{\ell-1}-1} \frac{1}{2} \\
&= E\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right] H_\ell \left(\frac{1}{2}\right)^{H_{\ell-1}} \\
&\approx 0 \cdot H_\ell \cdot \left(\frac{1}{2}\right)^{H_{\ell-1}} \\
&= 0
\end{aligned}
\tag{50}
$$

And finally the variance for the backward pass is derived using the same assumptions.

$$
\begin{aligned}
Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}}\right] &= H_\ell Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\frac{\partial z_{h_\ell}}{\partial z_{h_{\ell-1}}}\right] \\
&= H_\ell\Bigg(Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right]E\left[\frac{\partial z_{h_\ell}}{\partial z_{h_{\ell-1}}}\right]^2 + E\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right]^2 Var\left[\frac{\partial z_{h_\ell}}{\partial z_{h_{\ell-1}}}\right] \\
&+ Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right]Var\left[\frac{\partial z_{h_\ell}}{\partial z_{h_{\ell-1}}}\right]\Bigg) \\
&\approx Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right]H_\ell Var\left[\frac{\partial z_{h_\ell}}{\partial z_{h_{\ell-1}}}\right] \\
&\approx Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right]H_\ell\Bigg(E\left[\left(\frac{z_{h_\ell}}{W_{h_{\ell-1},h_\ell}z_{h_{\ell-1}}+1-W_{h_{\ell-1},h_\ell}}\right)^2\right]E[W_{h_{\ell-1},h_\ell}^2] \\
&- E\left[\frac{z_{h_\ell}}{W_{h_{\ell-1},h_\ell}z_{h_{\ell-1}}+1-W_{h_{\ell-1},h_\ell}}\right]^2 E[W_{h_{\ell-1},h_\ell}]^2\Bigg) \\
&\approx Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right]H_\ell\Bigg(E\left[\left(\frac{z_{h_\ell}}{W_{h_{\ell-1},h_\ell}z_{h_{\ell-1}}+1-W_{h_{\ell-1},h_\ell}}\right)^2\right]E[W_{h_{\ell-1},h_\ell}^2] \\
&- \left(\frac{1}{2}\right)^{2\cdot(H_{\ell-1}-1)}\left(\frac{1}{2}\right)^2\Bigg) \\
&\approx Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right]H_\ell\Bigg(\left(\left(Var[W_{h_{\ell-1},h_\ell}]+\frac{1}{4}\right)\left(Var[z_{h_{\ell-1}}]+1\right)\right)^{H_{\ell-1}-1} \\
&\cdot\left(Var[W_{h_{\ell-1},h_\ell}]+\frac{1}{4}\right)-\left(\frac{1}{2}\right)^{2\cdot H_{\ell-1}}\Bigg) \\
&= Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right]H_\ell\Bigg(\left(Var[W_{h_{\ell-1},h_\ell}]+\frac{1}{4}\right)^{H_{\ell-1}}\left(Var[z_{h_{\ell-1}}]+1\right)^{H_{\ell-1}-1} \\
&- \left(\frac{1}{2}\right)^{2\cdot H_{\ell-1}}\Bigg)
\end{aligned}
\tag{51}
$$

B.3.3 Initialization

The expectation of $W_{h_{\ell-1},h_\ell}$ should be $E[W_{h_{\ell-1},h_\ell}] = \frac{1}{2}$. Using the variance approximations found, the variance should be according to the forward pass:

$$
Var[W_{h_{\ell-1},h_\ell}] = \left((1+Var[z_{h_\ell}])^{-H_{\ell-1}}Var[z_{h_\ell}] + (4+4Var[z_{h_\ell}])^{-H_{\ell-1}}\right)^{\frac{1}{H_{\ell-1}}} - \frac{1}{4} \tag{52}
$$

And according to the backward pass it should be:

$$
Var[W_{h_{\ell-1},h_\ell}] = \left(H_\ell(1+Var[z_{h_{\ell-1}}])(4+4Var[z_{h_{\ell-1}}])^{-H_{\ell-1}} + (1+Var[z_{h_{\ell-1}}])^{1-H_{\ell-1}}\right)^{\frac{1}{H}} - \frac{1}{4} \tag{53}
$$

These are both dependent on the input variance. If the input variance is know then optimal initialization is possible. However, as this is often not the case one can perhaps assume that $Var[z_{h_{\ell-1}}] = 1$. This is not an unreasonable assumption in many cases, as there may either be a normalization layer somewhere or the input is normalized. If unit variance is assumed, one gets from the forward pass:

$$
Var[W_{h_{\ell-1},h_\ell}] = \left(2^{-H_{\ell-1}} + 8^{-H_{\ell-1}}\right)^{\frac{1}{H_{\ell-1}}} - \frac{1}{4} = \frac{1}{8}\left(\left(4^{H_{\ell-1}}+1\right)^{H_{\ell-1}} - 2\right) \tag{54}
$$

And from the backward pass:

$$Var[W_{h_{\ell-1},h_\ell}] = \left(2H_\ell 8^{-H_{\ell-1}} + 2^{1-H_{\ell-1}}\right)^{\frac{1}{H}} - \frac{1}{4} \tag{55}$$

The variance requirement for the backward pass is hard to satisfy, as this is dependent on two variables. However, the variance requirement from the forward pass quickly $Var[W_{h_{\ell-1},h_\ell}] = \frac{1}{4}$ may be a reasonable initialization.

C Simple function task

C.1 Dataset generation

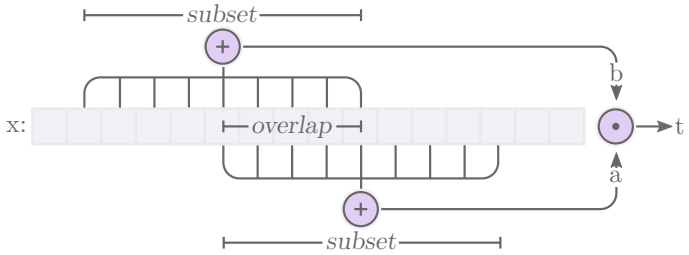


Figure 10: Dataset is parameterized into "Input Size", "Subset Ratio", "Overlap Ratio", an Operation (here showing multiplication), "Interpolation Range" and "Extrapolation Range" from which the data set sampled.

All datasets in the simple function task experiments are generated using the following algorithm:

Algorithm 1 Dataset sampling algorithm

function DATASET(OP($\cdot,\cdot$) : Operation, i : InputSize, s : SubsetRatio, o : OverlapRatio, R : Range)
  $\mathbf{x} \leftarrow$ UNIFORM(R_{lower}, R_{upper}, i) ▷ Sample i elements uniformly
  $k \leftarrow$ UNIFORM($0, 1 - 2s - o$) ▷ Sample offset
  $a \leftarrow$ SUM($\mathbf{x}[ik : i(k+s)]$) ▷ Create sum a from subset
  $b \leftarrow$ SUM($\mathbf{x}[i(k+s-o) : i(k+2s-0)]$) ▷ Create sum b from subset
  $t \leftarrow$ OP(a, b) ▷ Perform operation on a and b
  return x, t

C.2 Ablation study

Table 5: Shows the success-rate for $\mathcal{L}_{\mathbf{W}_1,\mathbf{W}_2} < \mathcal{L}_{\mathbf{W}_1^\epsilon,\mathbf{W}_2^*}$, at what global step the model converged at, and the sparsity error for all weight matrices.

Model	Success	Converged at		Sparsity error
	Rate	Median	Mean	Mean
NAC$_\bullet$	40%	$2.8 \cdot 10^6$	$3.1 \cdot 10^6 \pm 2.0 \cdot 10^6$	$2.8 \cdot 10^{-2} \pm 8.9 \cdot 10^{-2}$
NAC$_\bullet$ ($\mathbf{W} = \sigma(\hat{\mathbf{W}})$)	100%	$1.9 \cdot 10^6$	$1.9 \cdot 10^6 \pm 3.1 \cdot 10^5$	$1.1 \cdot 10^{-4} \pm 1.0 \cdot 10^{-4}$
NMU	100%	$1.2 \cdot 10^6$	$1.2 \cdot 10^6 \pm 2.0 \cdot 10^5$	$1.6 \cdot 10^{-3} \pm 9.2 \cdot 10^{-4}$
NMU ($\mathbf{W} = \hat{\mathbf{W}}$)	100%	$1.3 \cdot 10^6$	$1.2 \cdot 10^6 \pm 1.9 \cdot 10^5$	$3.9 \cdot 10^{-3} \pm 1.2 \cdot 10^{-3}$
NMU ($\mathbf{z} = \mathbf{W} \odot \mathbf{x}$)	100%	$1.2 \cdot 10^6$	$1.2 \cdot 10^6 \pm 2.0 \cdot 10^5$	$1.6 \cdot 10^{-3} \pm 9.2 \cdot 10^{-4}$
NMU (no $\mathcal{R}_{oob}$)	100%	$1.2 \cdot 10^6$	$1.2 \cdot 10^6 \pm 1.9 \cdot 10^5$	$1.7 \cdot 10^{-3} \pm 4.6 \cdot 10^{-4}$
NMU (no $\mathcal{R}_{sparse}$)	100%	$1.2 \cdot 10^6$	$1.2 \cdot 10^6 \pm 1.9 \cdot 10^5$	$1.7 \cdot 10^{-3} \pm 9.0 \cdot 10^{-4}$

Remove both regualizers

C.2.1 Moments and initialization for addition

Initialization is important for fast and consistent convergence. The desired properties are according to Glorot et al. [**?**]:

$$\begin{aligned} E[z_{h_\ell}] &= 0 & E\left[\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}}\right] &= 0 \\ Var[z_{h_\ell}] &= Var\left[z_{h_{\ell-1}}\right] & Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}}\right] &= Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right] \end{aligned} \tag{56}$$

The NAU layer is trivial, as this is just a linear layer. Thus the result from Glorot et al. ($Var[W_{h_{\ell-1},h_\ell}] = \frac{2}{H_{\ell-1}+H_\ell}$) can be used [**?**].

However, the original NAC_+ unit is less trivial as $W_{h_{\ell-1},h_\ell}$ is not sampled directly. Assuming that $\hat{W}_{h_\ell,h_{\ell-1}} \sim \mathrm{Uniform}[-r,r]$ and $\hat{M}_{h_\ell,h_{\ell-1}} \sim \mathrm{Uniform}[-r,r]$ then the variance can be derived (see proof in Appendix B.1) to be:

$$Var[W_{h_{\ell-1},h_\ell}] = \frac{1}{2r}\left(1 - \frac{\tanh(r)}{r}\right)\left(r - \tanh\left(\frac{r}{2}\right)\right) \tag{57}$$

One can the solve for r, given the desired variance.

C.2.2 Moments and initialization for multiplication

Maybe put this entire section in appendix?

put in assumptions in appendix

Using second order multivariate Taylor approximation and some assumptionsof uncorrelated stochastic variables, the expectation and variance of the $\mathrm{NAC}_\bullet$ layer can be estimated to:

$$\begin{aligned} f(c_1,c_2) &= \left(1 + c_1 \frac{1}{2} Var[W_{h_\ell,h_{\ell-1}}] \log(|E[z_{h_{\ell-1}}]| + \epsilon)^2\right)^{c_2\ H_{\ell-1}} \\ E[z_{h_\ell}] &\approx f\left(1,1\right) \\ Var[z_{h_2}] &\approx f\left(4,1\right) - f\left(1,2\right) \\ E\left[\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}}\right] &= 0 \\ Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}}\right] &\approx Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right] H_\ell\ f\left(4,1\right)\ Var[W_{h_\ell,h_{\ell-1}}] \\ &\quad \cdot \left(\frac{1}{\left(|E[z_{h_{\ell-1}}]| + \epsilon\right)^2} + \frac{3}{\left(|E[z_{h_{\ell-1}}]| + \epsilon\right)^4} Var[z_{h_{\ell-1}}]\right) \end{aligned} \tag{58}$$

This is problematic because $E[z_{h_\ell}] \geq 1$, and the variance explodes for $E[z_{h_{\ell-1}}] = 0$ which is normally a desired property.

For our proposed NMU, the expectation and variance can be derived (see proof in Appendix B.3) using the same assumptions as before, although no Taylor approximation is required:

$$\begin{aligned} E[z_{h_\ell}] &\approx \left(\frac{1}{2}\right)^{H_{\ell-1}} \\ E\left[\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}}\right] &\approx 0 \\ Var[z_{h_\ell}] &\approx \left(Var[W_{h_{\ell-1},h_\ell}] + \frac{1}{4}\right)^{H_{\ell-1}} \left(Var[z_{h_{\ell-1}}] + 1\right)^{H_{\ell-1}} - \left(\frac{1}{4}\right)^{H_{\ell-1}} \\ Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_{\ell-1}}}\right] &\approx Var\left[\frac{\partial \mathcal{L}}{\partial z_{h_\ell}}\right] H_\ell \\ &\quad \cdot \left(\left(Var[W_{h_{\ell-1},h_\ell}] + \frac{1}{4}\right)^{H_{\ell-1}} \left(Var[z_{h_{\ell-1}}] + 1\right)^{H_{\ell-1}-1} - \left(\frac{1}{4}\right)^{H_{\ell-1}}\right) \end{aligned} \tag{59}$$

consider throwing in appendix

These expectations are much more well behaved. It is properly unlikely to expect that the expectation can become zero, since the identity for multiplication is 1. However, for a large $H_{\ell-1}$ it will be near zero.

The variance is also more well-behaved, but does not provide a input-independent initialization strategy. We propose initializing with $Var[W_{h_{\ell-1},h_\ell}] = \frac{1}{4}$, as this is the solution to $Var[z_{h_\ell}] = Var[z_{h_{\ell-1}}]$ assuming $Var[z_{h_{\ell-1}}] = 1$ and a large $H_{\ell-1}$ (see proof in Appendix B.3.3). However, feel free to compute more exact solutions.