ELEMENTWISE LAYER NORMALIZATION

Felix Stollenwerk
AI Sweden
felix.stollenwerk@ai.se

ABSTRACT

A recent paper proposed Dynamic Tanh (DyT) as a drop-in replacement for Layer Normalization. Although the method is empirically well-motivated and appealing from a practical point of view, it lacks a theoretical foundation. In this work, we derive DyT mathematically and show that a well-defined approximation is needed to do so. By dropping said approximation, an alternative element-wise transformation is obtained, which we call Elementwise Layer Normalization (ELN). We demonstrate that ELN resembles Layer Normalization more accurately than DyT does.

1 Introduction

Most modern neural network architectures contain normalization layers. These have been shown to have beneficial effects on model training, such as faster and more stable convergence and better results (see e.g. Huang et al. [2023]). The most widely used normalization layers nowadays, especially in transformers, are Layer Normalization [Ba et al., 2016] and RMSNorm [Zhang and Sennrich, 2019]. Both employ activation statistics across the channels (or hidden dimensions) of the neural network. More concretely, consider a single token representation $x \in \mathbb{R}^C$. Layer normalization computes the mean and standard deviation,

$$\mu = \frac{1}{C} \sum_{k=1}^{C} x_k \tag{1}$$

$$\sigma^2 = \frac{1}{C} \sum_{k=1}^{C} (x_k - \mu)^2 \tag{2}$$

and acts on x by centering and scaling:

$$y = \frac{x - \mu}{\sqrt{\sigma^2}} \tag{3}$$

Note that $\mu, \sigma \in \mathbb{R}$ and $y \in \mathbb{R}^C$. In RMSNorm, the data centering is skipped, $\mu = 0$.

Recently, Zhu et al. [2025] have suggested a paradigm shift by employing an element-wise, non-linear transformation called Dynamic Tanh (DyT):

$$y = \tanh(\alpha x) \tag{4}$$

They have shown empirically for transformer-based architectures that DyT resembles Layer Normalization in the sense that it linearly transforms small values of x while squashing large values. However, it uses a learnable parameter $\alpha \in \mathbb{R}$, instead of relying on activation statistics like traditional normalization methods. DyT can be used as a drop-in replacement for normalization layers, leading to performance on par with normalization layers while being significantly faster. A potential caveat of the approach is that the initial values for α may require fine-tuning in certain cases. Although DyT is well-motivated empirically, the authors did not provide a theoretical justification for why it resembles Layer Normalization. In the present work, we aim to enhance the theoretical understanding of DyT. In Sec. 2, we find that it can be mathematically derived using the approximation that the variance σ^2 is a constant, independent of a single element x_i. By lifting this approximation in Sec. 3, we find an alternative element-wise transformation that we call *Elementwise Layer Normalization (ELN)*. In Sec. 4, it is shown that ELN resembles Layer Normalization more accurately than DyT does. Finally, our conclusions are presented in Sec. 5.

2 Dynamic Tanh (DyT)

In this section, we provide a mathematical derivation of the DyT function as an element-wise approximation of Layer Normalization. This is done in three steps:

- The derivative of Layer Normalization with respect to its input is computed, resulting in a differential equation.
- The differential equation is simplified by using an approximation.
- The simplified differential equation is solved, leading to the DyT function.

Theorem 1 (Layer Normalization Derivative)**.** *Let* $x \in \mathbb{R}^C$ *and*

$$y = \frac{x - \mu}{\sqrt{\sigma^2}} \tag{3}$$

with

$$\mu = \frac{1}{C} \sum_{k=1}^{C} x_k \tag{1}$$

$$\sigma^2 = \frac{1}{C} \sum_{k=1}^{C} (x_k - \mu)^2 \tag{2}$$

Then $\forall i \in [1, \ldots, C]$*:*

$$\frac{dy_i}{dx_i} = F(x) \left(C - 1 - y_i^2 \right) \tag{5}$$

with

$$F(x) = \frac{1}{C\sqrt{\sigma^2}} \tag{6}$$

The proof can be found in App. A.1. Note that Eq. (5) implies

$$y_i = \pm\sqrt{C-1} \quad \Rightarrow \quad \frac{dy_i}{dx_i} = 0 \tag{7}$$

Approximation We now assume that $F(x)$ can be approximated by a constant F that is independent of x:

$$F(x) \equiv F \tag{8}$$

Based on this, the following theorem shows that the DyT function can be obtained by solving the differential equation in Eq. (5).

Theorem 2 (Scaled DyT)**.** *The differential equation*

$$\frac{dy_i}{dx_i} = F \left(C - 1 - y_i^2 \right) \tag{9}$$

together with the boundary condition

$$y_i(x_i = 0) = 0 \tag{10}$$

is solved by the function

$$y_i = \sqrt{C-1} \cdot \tanh\left(\alpha x_i\right) \tag{11}$$

The proof can be found in App. A.2. Eq. (11) represents the **scaled DyT** function. Note that in contrast to the original formulation of DyT [Zhu et al., 2025], it explicitly contains the minimum and maximum value of y_i in terms of the scaling factor $\sqrt{C-1}$.

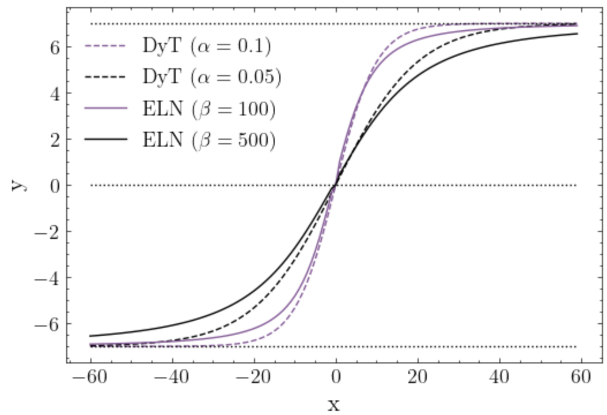

Figure 1: Functions DyT from Eq. (11) and ELN from Eq. (15) with different parameters α and β. The number of channels is set to $C = 50$. Hence, the dashed lines refer to the extrema $y = \pm\sqrt{C-1} = \pm 7$.

3 Elementwise Layer Normalization (ELN)

The results of the previous section raise the question whether it is possible to find an element-wise transformation akin to DyT that emerges from the differential equation Eq. (5) without the approximation from Eq. (8). Such a function could be a more accurate element-wise replacement for Layer Normalization. The question is answered by the following theorem.

Theorem 3 (General ELN)**.** *The differential equation*

$$\frac{dy_i}{dx_i} = F(x)\left(C - 1 - y_i^2\right) \tag{5}$$

together with the boundary condition

$$\frac{dy_i}{dx_i} \geq 0 \tag{12}$$

is solved by the function

$$y_i = \sqrt{C-1} \cdot \frac{(x_i - \mu)}{\sqrt{\beta + (x_i - \mu)^2}} \tag{13}$$

The proof can be found in App. A.3. Similarly to the (scaled) DyT function in Eq. (11), this function explicitly contains the minimum and maximum values of y_i in terms of $\sqrt{C-1}$. Furthermore, note that Eq. (13) is very similar to the original Layer Normalization formulation, Eq. (3). This is of course to be expected since Theorem 3 simply uses the inverse operation of Theorem 1. In fact, matching the two equations leads to an analytic expression for β, as shown by the following theorem.

Theorem 4 (Learnable Parameter β)**.**

$$y_i \overset{(3)}{:=} \frac{x_i - \mu}{\sqrt{\sigma^2}} \overset{(13)}{=} \sqrt{C-1} \cdot \frac{(x_i - \mu)}{\sqrt{\beta + (x_i - \mu)^2}} \quad \Longleftrightarrow \quad \beta \equiv \beta_i = (C-1) \cdot \sigma^2_{\neq i} - \sigma^2 \tag{14}$$

Here, $\sigma^2_{\neq i} := \frac{1}{C-1} \sum_{k \neq i} (x_k - \mu)^2$ denotes the variance without the contribution from x_i.

The proof can be found in App. A.4. We emphasize that for the equations to match, $\beta \equiv \beta_i$ needs to be channel-specific. This leads us to an important insight. By promoting β_i from a channel-specific parameter to a global learnable parameter β in Eq. (13), the direct equivalence of Layer Normalization and ELN is broken[1]. In accordance with the fact that the importance of normalization can be attributed to the effect it has on outliers [Zhu et al., 2025], we hypothesize that the model automatically learns a global β that describes outliers well (instead of data points close to the mean).

[1]Note that the same logic applies to DyT. In Eq. (11), α was tacitly promoted to a global learnable parameter.

Based on this, we can safely focus on outliers, $|x_i| \gg |\mu|$, in which case Eq. (13) can be approximated by the function

$$y_i = \sqrt{C-1} \cdot \frac{x_i}{\sqrt{\beta + x_i^2}} \tag{15}$$

We call this function **Elementwise Layer Normalization (ELN)**, as it represents an element-wise approximation of LN for outliers. In contrast to Layer Normalization, it does not rely on activation statistics. While this is a feature that ELN shares with DyT, its action on outliers resembles more closely the original Layer Normalization, as we will see in Sec. 4.

The two discussed solutions, DyT from Eq. (11) and ELN from Eq. (15), are compared in Fig. 1. Their shapes are quite similar, but DyT converges faster to the extrema ($\pm\sqrt{C-1}$) than ELN.

4 Simulations

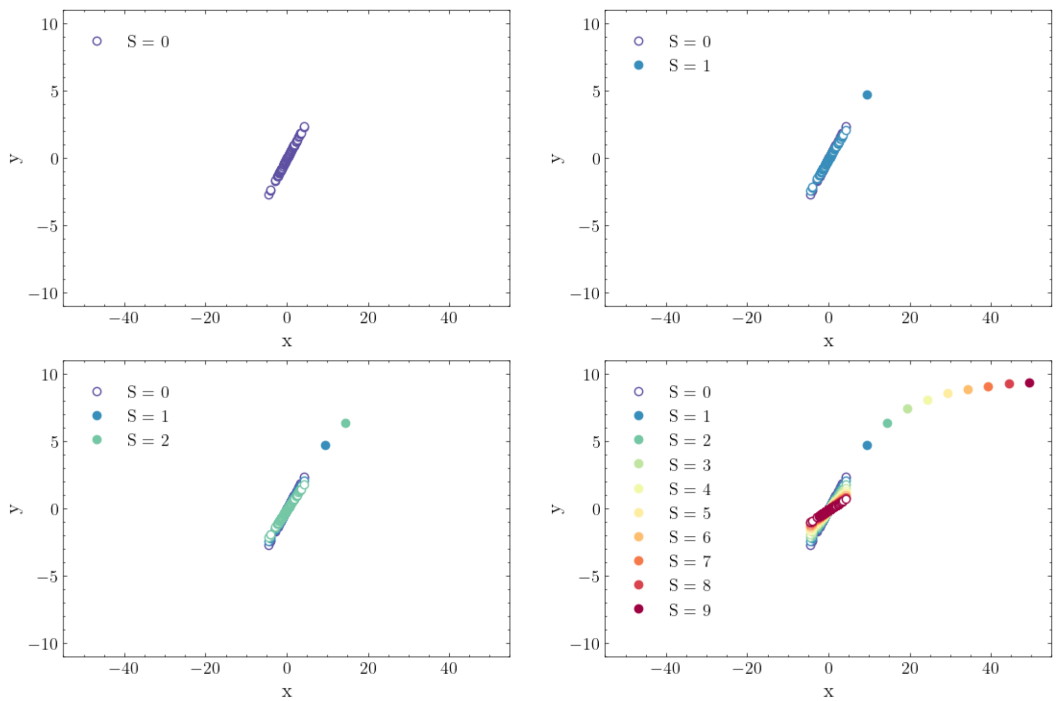

Figure 2: Stepwise outlier simulation. *Top left:* The original sample x and its normalized counterpart y, shown as empty circles ($S = 0$). *Top right:* The largest element of x was increased by one step, making it a modest outlier, shown as a filled circle ($S = 1$). A slight change in the slope of the linear function can be observed. *Bottom left:* The largest element of x was increased by two steps, making it a slightly more significant outlier, shown as a filled circle ($S = 2$). The function that connects the outliers indicates non-linearity. *Bottom right:* The largest element of x was increased by nine steps, resulting in an extreme outlier, shown as a filled circle ($S = 9$). The function that connects the outliers is clearly non-linear.

In this section, we use randomly sampled data for x and apply Layer Normalization to obtain y. We then simulate outliers of different degrees of severity in order to gain an intuitive understanding of how Layer Normalization squashes the input x. Afterwards, we employ DyT and ELN with optimal parameters α and β to see how well they describe the data generated by Layer Normalization.

4.1 Layer Normalization

We use C channels and take a normally distributed sample of N values $x = (x_1, x_2, \dots, x_N)$ with mean μ and standard deviation σ:

$$x \sim \mathcal{N}(\mu, \sigma^2) \tag{16}$$

The exact values of the variables do not matter, but we use $C = 100$, $N = 50$ and $\sigma = 2$ throughout all simulations. The mean is set to $\mu = 0$. First, we apply layer normalization and compute y according to Eq. (3). The result is plotted in the top left panel of Fig. 2. Next, we simulate outliers by increasing the largest value of x in steps of 5. The number of steps is denoted by S:

$$x_o \to x_o + 5 \cdot S \qquad \text{with} \qquad o = \underset{k}{\operatorname{argmax}}\, x_k \tag{17}$$

As before, layer normalization is applied to the resulting vector x. The results are plotted in the top right and bottom left panels of Fig. 2 for $S = 1$ and $S = 2$, respectively. Repeating the process up to $S = 9$ yields the plot on the bottom right of Fig. 2. In accordance with Zhu et al. [2025], we observe that

1. The slope of the linear function $y_i(x_i)$ decreases with the variance of x.
2. The outliers, considered separately, follow a non-linear function. The larger the outlier x_o, the more squashed the function is.

4.2 DyT and ELN

In the next step, our aim is to describe the simulated data by DyT and ELN as defined in Eq. (11) and (15), respectively. We only use the outliers (filled circles in Fig. 2) as data points for the fit[2] since—as discussed in Sec. 3—it is primarily those we want to reproduce the Layer Normalization behavior for. Performing fits yields the optimal parameters

$$\alpha = 0.049 \tag{18}$$
$$\beta = 301.1 \tag{19}$$

for DyT and ELN, respectively. Both fitted functions are displayed in Fig. 3 together with the data and the residuals.

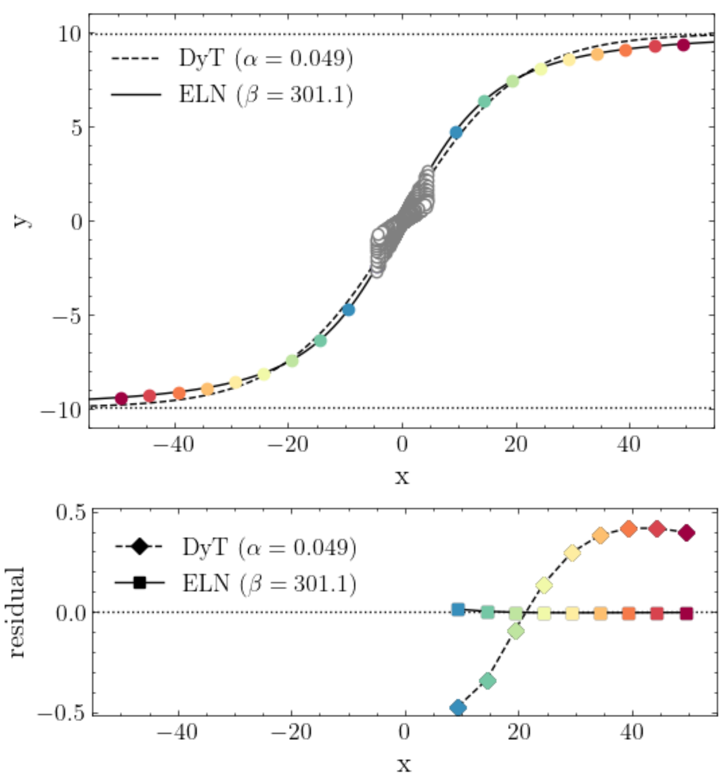

Figure 3: *Top panel:* Functions DyT and ELN with optimal parameters α and β, respectively, fitted on the outliers shown as colored, filled circles. They non-outlier data are shown in gray, empty circles. *Bottom panel:* Residuals for the the functions DyT and ELN with respect to the outlier data. As the residuals are antisymmetric (like the data and the functions), only positive outliers are displayed for the sake of simplicity.

[2] In practice, we use the mirrored data points $(-x, -y)$ as well for the sake of numerical stability.

We find that ELN describes the Layer Normalization data much more accurately than DyT. The mean absolute residuals are 0.33 for DyT and < 0.01 for ELN. This reflects the fact that DyT and ELN are approximate and exact solutions to the differential equation provided by Layer Normalization, Eq. (5), respectively.

5 Conclusions

This work provides a theoretical foundation for the empirically observed approximate equivalence of DyT and Layer Normalization. In addition, our analysis reveals that the direct, element-wise counterpart to Layer Normalization is given by a transformation called Elementwise Layer Normalization (ELN). We leave it for future work to investigate how ELN compares to DyT in terms of performance and feasibility. The code used to produce the data, results and figures is available at `https://github.com/flxst/ELN`.

References

Lei Huang, Jie Qin, Yi Zhou, Fan Zhu, Li Liu, and Ling Shao. Normalization techniques in training dnns: Methodology, analysis and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8):10173–10196, 2023. doi:10.1109/TPAMI.2023.3250241.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016. URL `https://arxiv.org/abs/1607.06450`.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper_files/paper/2019/file/1e8a19426224ca89e83cef47f1e7f53b-Paper.pdf`.

Jiachen Zhu, Xinlei Chen, Kaiming He, Yann LeCun, and Zhuang Liu. Transformers without normalization, 2025. URL `https://arxiv.org/abs/2503.10622`.

A Theorem Proofs

A.1 Theorem 1

Theorem 1 (Layer Normalization Derivative)**.** *Let* $x \in \mathbb{R}^C$ *and*

$$y = \frac{x - \mu}{\sqrt{\sigma^2}} \tag{3}$$

with

$$\mu = \frac{1}{C} \sum_{k=1}^{C} x_k \tag{1}$$

$$\sigma^2 = \frac{1}{C} \sum_{k=1}^{C} (x_k - \mu)^2 \tag{2}$$

Then $\forall i \in [1, \ldots, C]$*:*

$$\frac{dy_i}{dx_i} = F(x) \left(C - 1 - y_i^2 \right) \tag{5}$$

with

$$F(x) = \frac{1}{C\sqrt{\sigma^2}} \tag{6}$$

Proof. We start from Eq. (3) and compute the derivative of y_i with respect to x_i:

$$\frac{dy_i}{dx_i} = \frac{d}{dx_i} \left(\frac{x_i - \mu}{\sqrt{\sigma^2}} \right) \tag{20}$$

Defining

$$f = x_i - \mu \tag{21}$$
$$g = \sqrt{\sigma^2} \tag{22}$$

and using the shorthand notation

$$f' := \frac{df}{dx}$$
$$g' := \frac{dg}{dx}$$

the quotient rule states

$$\frac{dy_i}{dx_i} = \frac{f'g - fg'}{g^2} \tag{23}$$

We compute the derivatives in Eq. (23):

$$\begin{aligned}
f' &\overset{(21)}{=} \frac{d}{dx_i}\left(x_i - \mu\right) \\
&\overset{(1)}{=} 1 - \frac{d}{dx_i}\left(\frac{1}{C}\sum_{k=1}^{C} x_k\right) \\
&= 1 - \frac{1}{C} \\
&= \frac{C-1}{C}
\end{aligned} \tag{24}$$

and

$$\begin{aligned}
g' &\overset{(22)}{=} \frac{d}{dx_i}\sqrt{\sigma^2} \\
&= \frac{1}{2\sqrt{\sigma^2}} \cdot \frac{d}{dx_i}\left(\sigma^2\right) \\
&\overset{(2)}{=} \frac{1}{2\sqrt{\sigma^2}} \cdot \frac{d}{dx_i}\left(\frac{1}{C}\sum_{k=1}^{C}\left(x_k - \mu\right)^2\right) \\
&= \frac{1}{2C\sqrt{\sigma^2}} \cdot \frac{d}{dx_i}\left(\left(x_i - \mu\right)^2 + \sum_{k \neq i}^{C}\left(x_k - \mu\right)^2\right) \\
&= \frac{1}{C\sqrt{\sigma^2}} \cdot \left(\left(x_i - \mu\right) \cdot \frac{d}{dx_i}\left(x_i - \mu\right) + \sum_{k \neq i}^{C}\left(x_k - \mu\right) \cdot \frac{d}{dx_i}\left(x_k - \mu\right)\right) \\
&\overset{(24)}{=} \frac{1}{C\sqrt{\sigma^2}} \cdot \left(\left(x_i - \mu\right) \cdot \left(1 - \frac{1}{C}\right) - \sum_{k \neq i}^{C}\left(x_k - \mu\right) \cdot \frac{1}{C}\right) \\
&= \frac{1}{C\sqrt{\sigma^2}} \cdot \left(\left(x_i - \mu\right) - \frac{1}{C}\sum_{k=1}^{C}\left(x_k - \mu\right)\right) \\
&\overset{(1)}{=} \frac{1}{C\sqrt{\sigma^2}} \cdot \left(\left(x_i - \mu\right) - \frac{1}{C}\left(C\mu - C\mu\right)\right) \\
&= \frac{1}{C\sqrt{\sigma^2}} \cdot \left(x_i - \mu\right) \\
&\overset{(3)}{=} \frac{y_i}{C}
\end{aligned} \tag{25}$$

Inserting Eqs. (24) and (25) into Eq. (23), we get

$$\begin{aligned}\frac{dy_i}{dx_i} &\overset{(23)}{=} \frac{f'g - fg'}{g^2} \\ &\overset{(24,25)}{=} \frac{\frac{C-1}{C} \cdot \sqrt{\sigma^2} - (x_i - \mu) \cdot \frac{y_i}{C}}{\sigma^2} \\ &\overset{(3)}{=} \frac{\frac{C-1}{C} \cdot \sqrt{\sigma^2} - \frac{1}{C}\sqrt{\sigma^2} \cdot y_i^2}{\sigma^2} \\ &= \frac{1}{C\sqrt{\sigma^2}} \left(C - 1 - y_i^2\right) \end{aligned} \tag{26}$$

With the abbreviation $F(x)$ from Eq. (6), Eq. (26) can be written in shorthand notation as Eq. (5). ☐

A.2 Theorem 2

Theorem 2 (Scaled DyT)**.** *The differential equation*

$$\frac{dy_i}{dx_i} = F\left(C - 1 - y_i^2\right) \tag{9}$$

together with the boundary condition

$$y_i(x_i = 0) = 0 \tag{10}$$

is solved by the function

$$y_i = \sqrt{C-1} \cdot \tanh\left(\alpha x_i\right) \tag{11}$$

Proof. For the sake of readability, we drop the channel index i, i.e. we use $x_i \to x$ and $y_i \to y$.

First, we separate the variables:

$$\begin{aligned} F \cdot dx &= \frac{dy}{C - 1 - y^2} \\ &= \frac{dy}{(\sqrt{C-1} - y)(\sqrt{C-1} + y)} \\ &= \frac{1}{2\sqrt{C-1}} \frac{\sqrt{C-1} - y + \sqrt{C-1} + y}{(\sqrt{C-1} - y)(\sqrt{C-1} + y)} dy \\ &= \frac{1}{2\sqrt{C-1}} \left(\frac{1}{\sqrt{C-1} + y} + \frac{1}{\sqrt{C-1} - y} \right) dy \end{aligned}$$

Integration yields

$$\begin{aligned} \frac{1}{2\sqrt{C-1}} \log\left(\frac{\sqrt{C-1} + y}{\sqrt{C-1} - y}\right) &= F \cdot x + \frac{c}{2\sqrt{C-1}} \\ \frac{\sqrt{C-1} + y}{\sqrt{C-1} - y} &= \exp(2\sqrt{C-1}Fx + c) \end{aligned}$$

Defining $Q := \exp(2\sqrt{C-1}Fx + c)$, we get

$$\begin{aligned} \frac{\sqrt{C-1} + y}{\sqrt{C-1} - y} &= Q \\ \sqrt{C-1} + y &= \left(\sqrt{C-1} - y\right) Q \\ \sqrt{C-1} + y &= \sqrt{C-1}Q - Qy \\ Qy + y &= \sqrt{C-1}\,(Q-1) \\ (Q+1)y &= \sqrt{C-1}\,(Q-1) \\ y &= \sqrt{C-1} \cdot \frac{Q-1}{Q+1} \end{aligned}$$

Replacing Q again, and using $A = \exp(c)$, yields

$$y_i = \sqrt{C-1} \cdot \frac{A \exp(2\sqrt{C-1} F x_i) - 1}{A \exp(2\sqrt{C-1} F x_i) + 1}$$

Note that in the last equation, we have reintroduced the channel index i. We enforce the boundary condition from Eq. (10) which requires $A = 1$. Together with the definition

$$\alpha := \sqrt{C-1} F$$

this leads to the (scaled) DyT function, Eq. (11). □

A.3 Theorem 3

Theorem 3 (General ELN). *The differential equation*

$$\frac{dy_i}{dx_i} = F(x) \left(C - 1 - y_i^2\right) \tag{5}$$

together with the boundary condition

$$\frac{dy_i}{dx_i} \geq 0 \tag{12}$$

is solved by the function

$$y_i = \sqrt{C-1} \cdot \frac{(x_i - \mu)}{\sqrt{\beta + (x_i - \mu)^2}} \tag{13}$$

Proof. For the sake of readability, we drop the channel index i, i.e. we use $x_i \to x$ and $y_i \to y$. First, we separate the variables:

$$F(x) \cdot dx = \frac{dy}{C - 1 - y^2}$$

Expressing the left hand side as

$$\begin{aligned} F(x) \cdot dx &\overset{(6)}{=} \frac{1}{C\sqrt{\sigma^2}} \cdot dx \\ &\overset{(3)}{=} \frac{1}{C} \frac{y}{x - \mu} \cdot dx \end{aligned}$$

we get

$$\frac{1}{C} \frac{dx}{x - \mu} = \frac{dy}{C - 1y - y^3}$$

With

$$\frac{d}{dx} (x - \mu) = \frac{C-1}{C} \tag{24}$$

integration of the left hand side gives

$$\begin{aligned} \int \frac{1}{C} \frac{dx}{x - \mu} &= \int \frac{1}{C} \frac{d(x-\mu)}{x - \mu} \frac{C}{C-1} \\ &= \frac{1}{C-1} \int \frac{d(x-\mu)}{x-\mu} \\ &= \frac{1}{C-1} \log(x - \mu) + \frac{\log(c^{\mathrm{LHS}})}{C-1} \\ &= \frac{1}{C-1} \log(x - \mu) + \frac{\log(c^{\mathrm{LHS}})}{C-1} \end{aligned} \tag{27}$$

where c^{LHS} is an integration constant. For the right hand side, we use the substitution

$$u := y^{-2}$$
$$\frac{du}{dy} = -2y^{-3}$$

Hence, the right hand side becomes

$$\begin{aligned}\frac{dy}{C-1y-y^3} &= \frac{du}{C-1y-y^3}\frac{dy}{du} \\ &= \frac{du}{C-1y-y^3}\left(-\frac{1}{2}y^3\right) \\ &= -\frac{1}{2}\frac{du}{C-1y^{-2}-1} \\ &= -\frac{1}{2}\frac{du}{C-1u-1}\end{aligned}$$

Integration yields

$$\begin{aligned}\int\frac{dy}{C-1y-y^3} &= -\frac{1}{2}\int\frac{du}{C-1u-1} \\ &= -\frac{1}{2C-1}\log\left(C-1u-1\right)+\frac{\log(c^{\mathrm{RHS}})}{C-1} \\ &= -\frac{1}{2C-1}\log\left(C-1y^{-2}-1\right)+\frac{\log(c^{\mathrm{RHS}})}{C-1}\end{aligned} \tag{28}$$

with another integration constant c^{RHS}. Comparing Eqs. (27) and (28) leads to

$$\frac{1}{C-1}\log\left(x-\mu\right) = -\frac{1}{2C-1}\log\left(C-1y^{-2}-1\right)+\frac{\log(c)}{C-1}$$

with $c := c^{\mathrm{RHS}}/c^{\mathrm{LHS}}$. This can also be written as

$$\begin{aligned}\log\left(x-\mu\right)-\log(c) &= -\frac{1}{2}\log\left(C-1y^{-2}-1\right) \\ \log\left(\frac{x-\mu}{c}\right) &= -\frac{1}{2}\log\left(C-1y^{-2}-1\right) \\ \log\left(\frac{c^2}{\left(x-\mu\right)^2}\right) &= \log\left(C-1y^{-2}-1\right) \\ \frac{c^2}{\left(x-\mu\right)^2} &= C-1y^{-2}-1 \\ \frac{1}{C-1}\left(\frac{c^2}{\left(x-\mu\right)^2}+1\right) &= y^{-2} \\ \left(\frac{c^2+\left(x-\mu\right)^2}{C-1\left(x-\mu\right)^2}\right) &= y^{-2} \\ \left(\frac{C-1\left(x-\mu\right)^2}{\beta+\left(x-\mu\right)^2}\right) &= y^2\end{aligned}$$

where in the last step, we have used $\beta := c^2 \geq 0$. Hence, we have the two general solutions

$$y_i = \pm\sqrt{C-1}\cdot\sqrt{\frac{\left(x_i-\mu\right)^2}{\beta+\left(x_i-\mu\right)^2}}$$

Note that in the last equation, we have reintroduced the channel index i. The boundary condition, Eq. (12), leads to the solution, Eq. (13). □

A.4 Theorem 4

Theorem 4 (Learnable Parameter β)**.**

$$y_i \overset{(3)}{:=} \frac{x_i - \mu}{\sqrt{\sigma^2}} \overset{(13)}{=} \sqrt{C-1} \cdot \frac{(x_i - \mu)}{\sqrt{\beta + (x_i - \mu)^2}} \quad \Longleftrightarrow \quad \beta \equiv \beta_i = (C-1) \cdot \sigma^2_{\neq i} - \sigma^2 \tag{14}$$

Here, $\sigma^2_{\neq i} := \frac{1}{C-1} \sum_{k \neq i} (x_k - \mu)^2$ *denotes the variance without the contribution from* x_i*.*

Proof.

$$\begin{aligned}
y_i &\overset{(3)}{=} \frac{x_i - \mu}{\sqrt{\sigma^2}} \\
&\overset{(2)}{=} \frac{x_i - \mu}{\sqrt{\frac{1}{C} \sum_{k=1}^{C} (x_k - \mu)^2}} \\
&= \sqrt{C} \cdot \frac{x_i - \mu}{\sqrt{\sum_{k \neq i} (x_k - \mu)^2 + (x_i - \mu)^2}} \\
&= \sqrt{C-1} \cdot \frac{x_i - \mu}{\sqrt{\frac{C-1}{C} \cdot \sum_{k \neq i} (x_k - \mu)^2 + \frac{C-1}{C} \cdot (x_i - \mu)^2}} \\
&= \sqrt{C-1} \cdot \frac{x_i - \mu}{\sqrt{\frac{C-1}{C} \cdot \sum_{k \neq i} (x_k - \mu)^2 - \frac{1}{C} \cdot (x_i - \mu)^2 + (x_i - \mu)^2}}
\end{aligned}$$

Matching with Eq. (13) yields Eq. (14):

$$\begin{aligned}
\beta &= \frac{C-1}{C} \cdot \sum_{k \neq i} (x_k - \mu)^2 - \frac{1}{C} \cdot (x_i - \mu)^2 \\
&= \sum_{k \neq i} (x_k - \mu)^2 - \frac{1}{C} \sum_{k} (x_k - \mu)^2 \\
&= (C-1) \cdot \sigma^2_{\neq i} - \sigma^2
\end{aligned}$$

□