Inverting the Radon Transform for GPD modeling using Artificial Neural Networks

Pietro Dall'Olio (Universidad de Huelva)

Collaborators: J. M. Morgado, J. Rodríguez Quintero, C. Mezrag, P. Sznajder

MeV2TeV III: Cordoba 16-17 Feb 2023

Universidad de Huelva

Motivations

- General Parton Distributions (GPD) are non perturbative functions that contain a lot of information on the hadron structure in terms of its constituents, quarks and gluons.

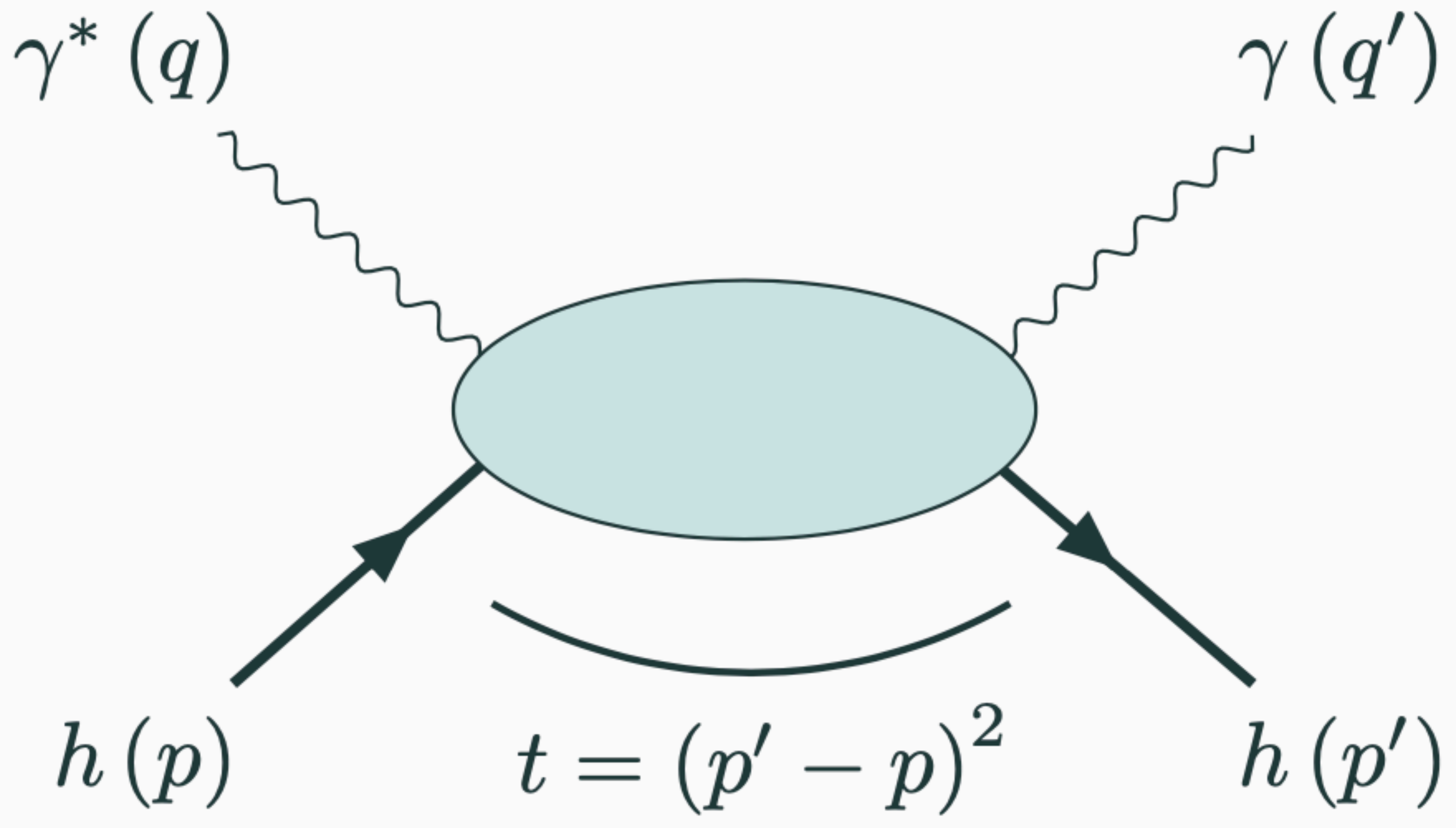
- The derivation from first principles of GPD $H(x, \xi, t)$ in the entire momentum space is challenging: in the overlap representation within the light cone quantization, a consistent truncation that preserves positivity is only possible in the DGALP region ($|x| > |\xi|$).

- Covariant Extension Strategy: exploiting the Radon Transform representation of the GPD that guarantees its polynomiality property:

$$H(x, \xi) = \mathscr{R}[h] \Longleftrightarrow \int_{-1}^{1} dx\, x^m\, H^q(x, \xi, t) = \sum_{\substack{k=0 \\ k\,even}}^{m+1} C_{k,m}(t)\, \xi^k$$

in order to estimate the Double Distribution h, i.e. inverting the Radon Transform, from a partial knowledge of the GPD in the DGALP region and consequently reconstructing it in its whole domain.

Generalized Parton Distribution

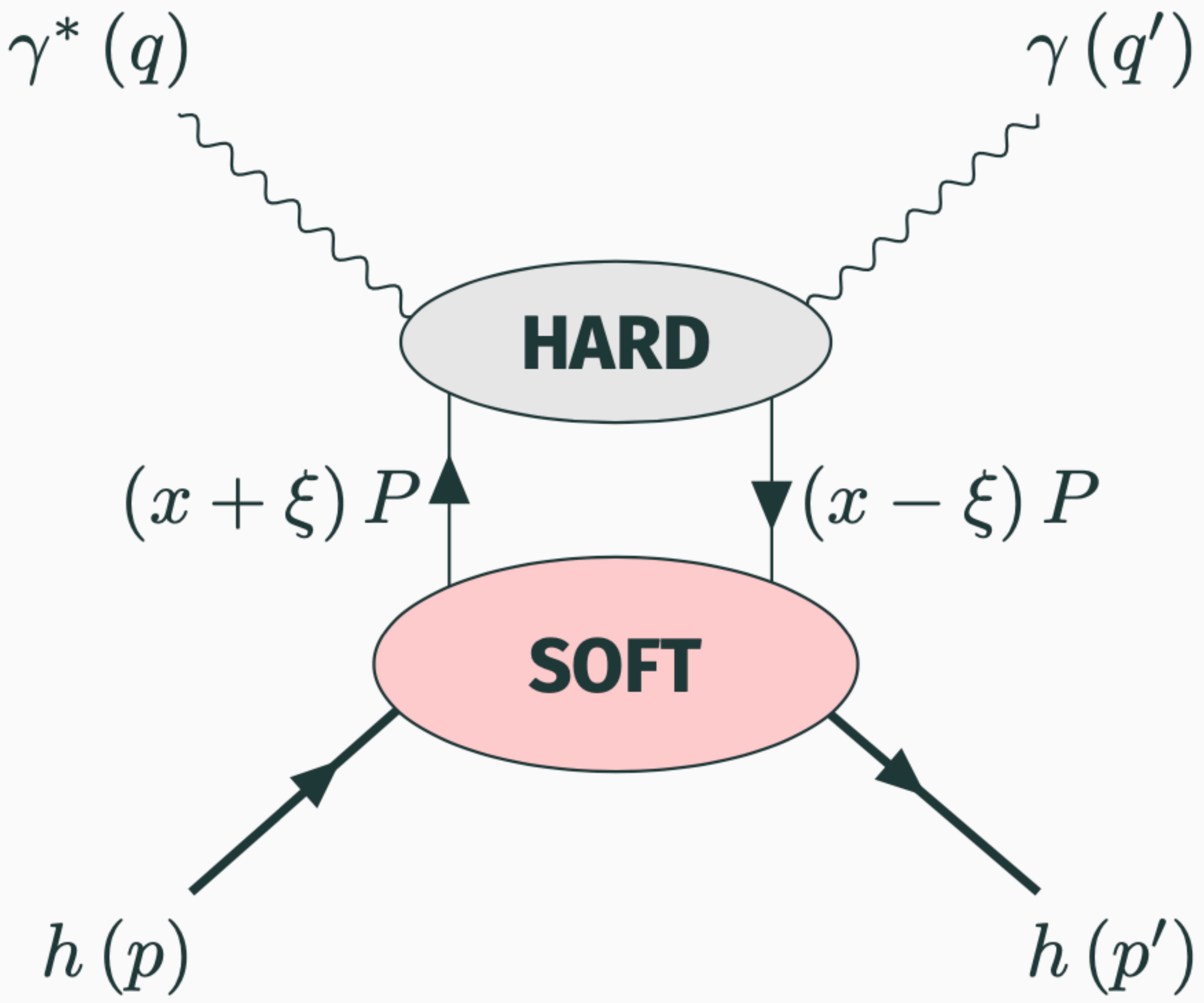


$Q^2 \to \infty$

$$Q = \frac{q + q'}{2}$$

$$P = \frac{p + p'}{2}$$

$$\xi = \frac{(p' - p)^+}{2P^+} = -\frac{\Delta^+}{2P^+}$$



$$\mathcal{M}(\xi, t; Q^2) = \sum_{p=q,g} \int_{-1}^{1} \frac{dx}{\xi} K^p \left(\frac{x}{\xi}, \frac{Q^2}{\mu_F^2}, \alpha_s(\mu_F) \right) H^p \left(x, \xi, t; \mu_F \right)$$

hard / perturbative

soft / non perturbative GPD

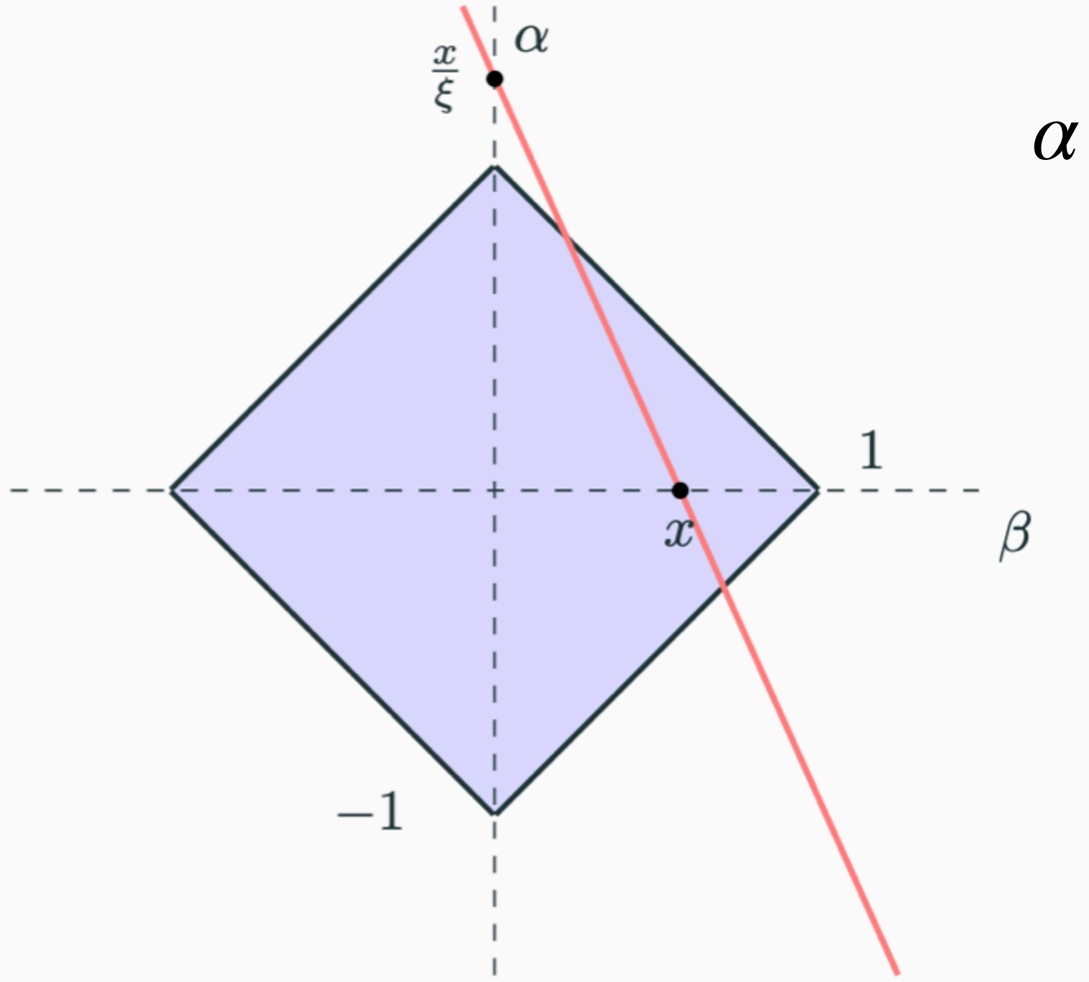
quark GPD

$$H^q(x,\xi,t) = \frac{1}{2}\int \frac{dz^-}{2\pi} e^{ixP^+z^-} \left\langle P+\frac{\Delta}{2}\middle|\bar{\psi}^q\left(-\frac{z}{2}\right)\gamma^+\psi^q\left(\frac{z}{2}\right)\middle|P-\frac{\Delta}{2}\right\rangle\Bigg|_{z^+=0,\ z^\perp=0}$$

$$= \int_\Omega d\beta d\alpha\,\delta\left(x-\beta-\alpha\xi\right)h(\beta,\alpha) + \theta(|\xi|-|x|)\ \text{D-terms}$$

$h(\beta,\alpha)$ → Double Distribution

$$\Omega = \{(\beta,\alpha)\,|\,|\beta|+|\alpha|\le 1\}$$



$$\alpha = \frac{1}{\xi}(-\beta+x)$$

$(x,\xi)\in[-1,1]\otimes[-1,1]$

DGLAP: $|x|>|\xi|$ **ERBL:** $|x|<|\xi|$

- $H(x,-\xi) = H(x,\xi) \leftrightarrow h(\beta,-\alpha) = h(\beta,\alpha)$
- $H(x,0,0) = q(x)\theta(x) - \bar{q}(-x)\theta(-x)$ $\rightarrow \Omega = \Omega_{\beta>0} + \Omega_{\beta<0}$

$\Omega_{\beta>0}$ → quark, $\Omega_{\beta<0}$ → antiquark

Uniqueness theorem

[F. Natterer, *The Mathematics of Computerized Tomography*]

If $H(x,\xi)$ is known for a subset of its domain whose points correspond to lines that span the entire $h(\beta,\alpha)$ domain (except for the axis $\beta = 0$ that corresponds to the D-term), then $h(\beta,\alpha)$ is unique.

→ $H(x,\xi)$ restricted to the DGLAP region $|x| > |\xi|$ corresponds to a unique $h(\beta,\alpha)$

Even a smaller subset of the DGALP region, where the skewness parameter is bounded from above $0 < \xi < \xi^{max}$ corresponds to a unique $h(\beta,\alpha)$

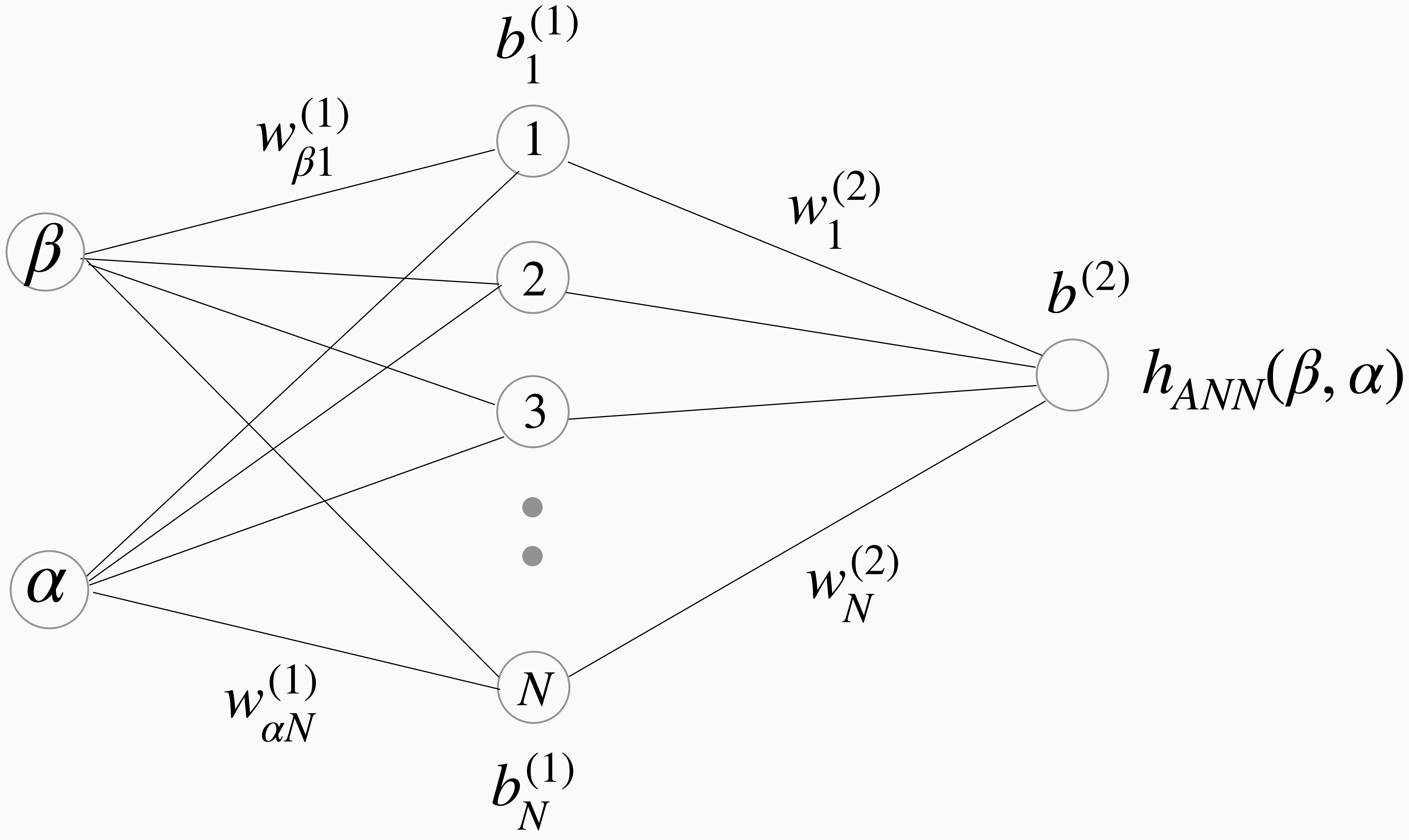
In experiments only a limited range of ξ available

Artificial Neural Networks

H. Dutrieux et al., *Eur.Phys.J.C 82 (2022)*

<u>Universal approximation theorem:</u> continuous function on compact approximated by ANN with one hidden layer

ANN used to approximate the DD $h(\beta, \alpha)$



Output of neuron i at layer j

$$o_i^{(j)} = \varphi^{(j)}\left(\sum_k w_{ki}^{(j)} o_k^{(j-1)} + b_i^{(j)}\right)$$

+ terms in order to impose constraints

$$\varphi^{(1)}(x) = \frac{1}{1+e^{-x}}, \quad \varphi^{(2)}(x) = x$$

Algorithm

- Initialize the NN parameters (randomly).
- Given a sampling set of GPD values $H_i(x_i, \xi_i)$ in the DGALP region, numerically evaluate the RT along each linea (x_i, ξ_i) using $h_{ANN}(\beta, \alpha)$ as DD. $\mathscr{R}h_{ANN}(x_i, \xi_i) = \hat{H}(x_i, \xi_i)$.
- Update the NN parameters using some optimization algorithm (gradient descent, genetic algorithm…) in order to minimize some loss function, e.g. $$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(H_i - \hat{H}_i\right)^2$$
- Iterate until convergence.

Note: This numerical approach is complementary to the Finite Element Methods implemented by J.M. Morgado (see his talk on Friday)

Nakanishi based model for pion

N. Chouika et al., *Phys.Lett B:780 (2018)*

$$H(x,\xi,t=0) = \begin{cases} 30\,\dfrac{(1-x)^2(x^2-\xi^2)}{(1-\xi^2)^2}, & |x| > \xi \\ 15\,\dfrac{(1-x)(\xi^2-x^2)(x+2x\xi+\xi^2)}{2\xi^3(1+\xi)^2}, & |x| < \xi \end{cases}$$

$$H(x,\xi,t=0) = (1-x)\int_{\Omega^+} d\beta d\alpha \delta(x-\beta-\alpha\xi)\, h(\beta,\alpha)$$

$$h(\beta,\alpha) = \frac{15}{2}\left(1-3(\alpha^2-\beta^2)-2\beta\right)$$

Output of the ANN hidden layer:

$$o_i = \varphi\left(w_{\beta i}\,\beta + w_{\alpha i}\,\alpha' + b_i\right) + w_{\alpha i} \to -w_{\alpha i}$$

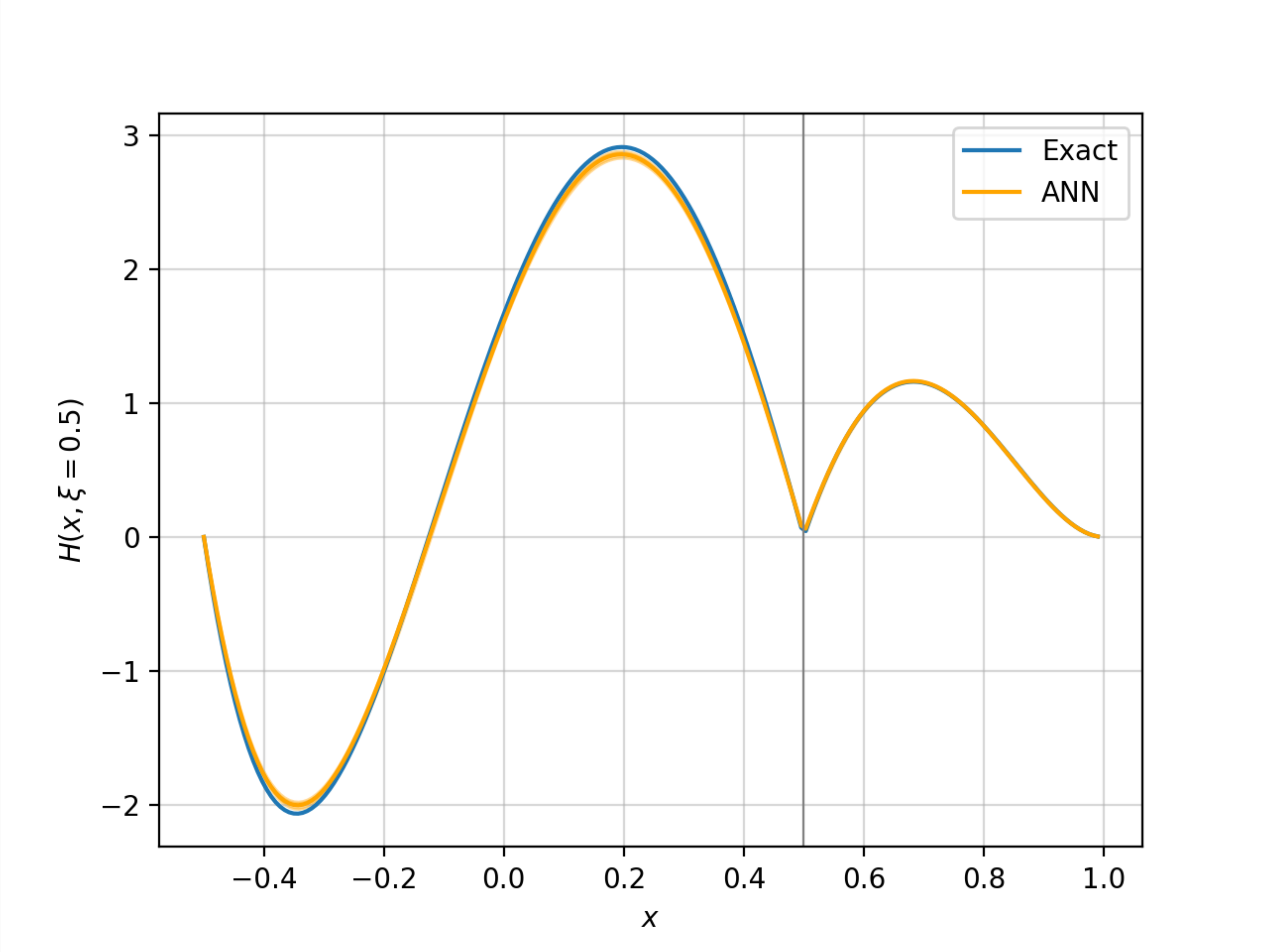
$$\alpha' = \frac{\alpha}{1-|\beta|} \in [-1,1]$$

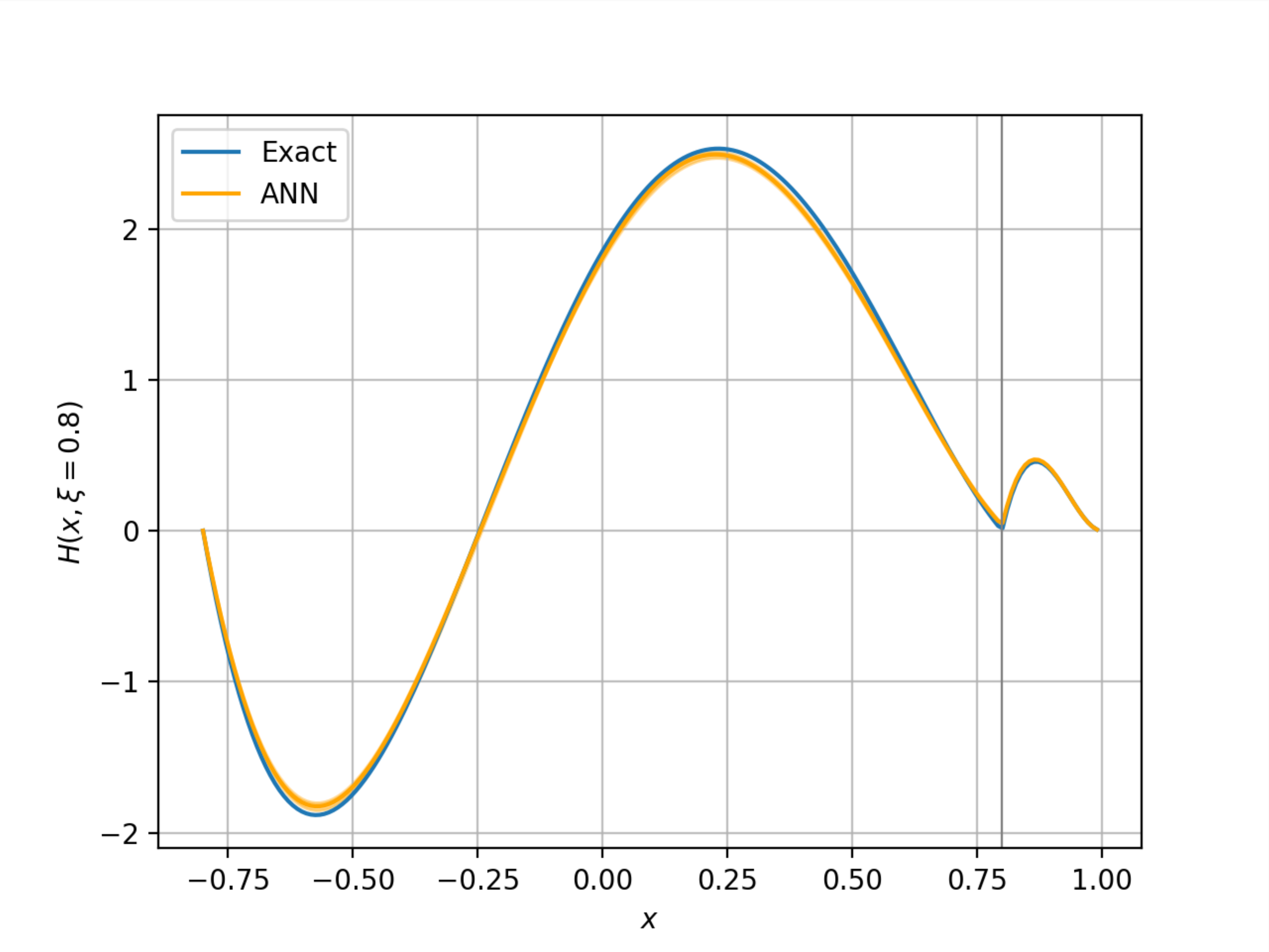
Imposes $h_{ANN}(\beta,-\alpha) = h_{ANN}(\beta,\alpha)$

from $H(x,-\xi) = H(x,\xi)$

$N_{points} = 10^4\,, \quad N_{neurons} = 100\,,$ Dropout regularization $r = 0.03\,,$ ADAM optimizer $lr = 0.001$

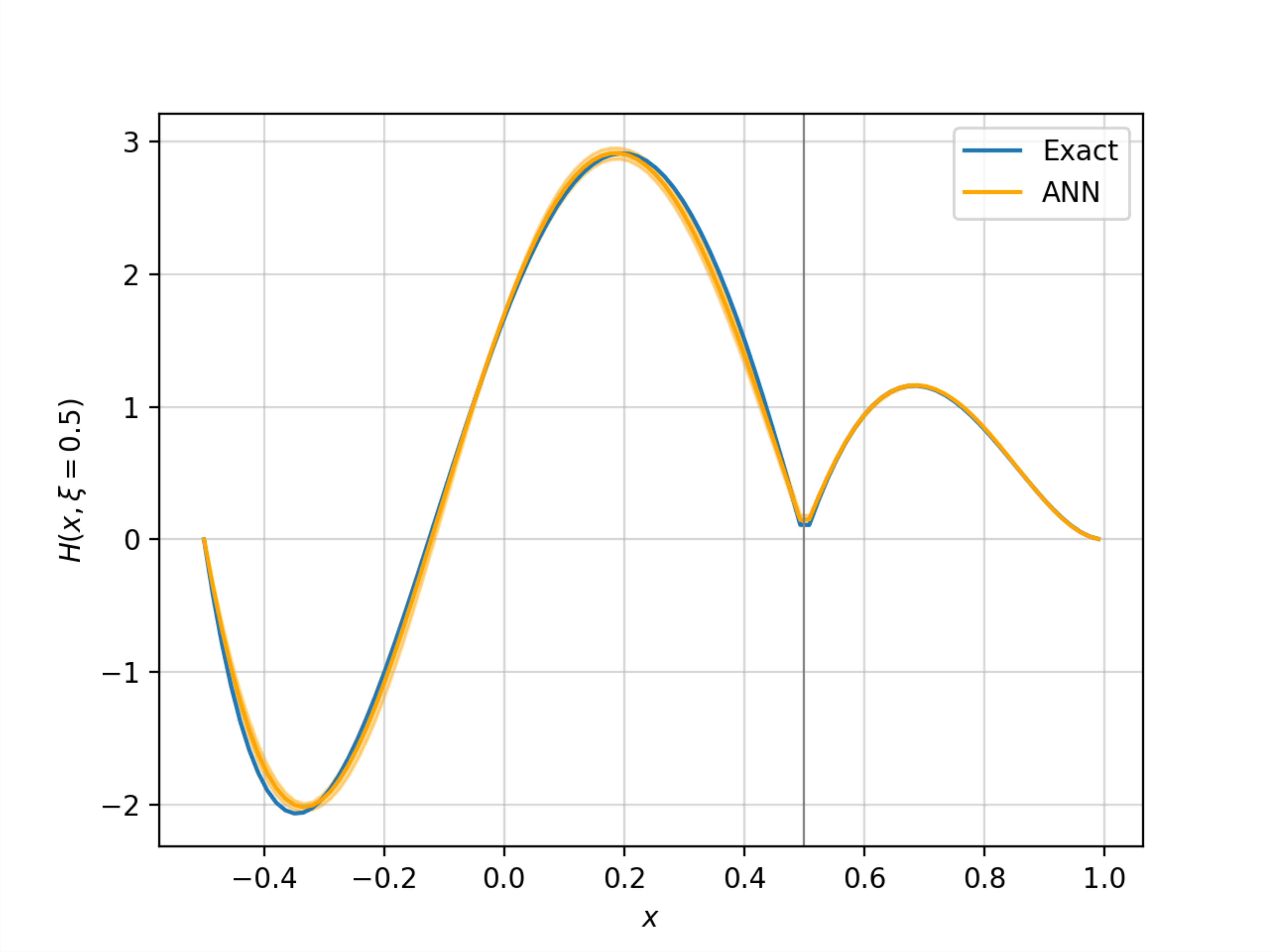
$$\xi^{max} = 1$$

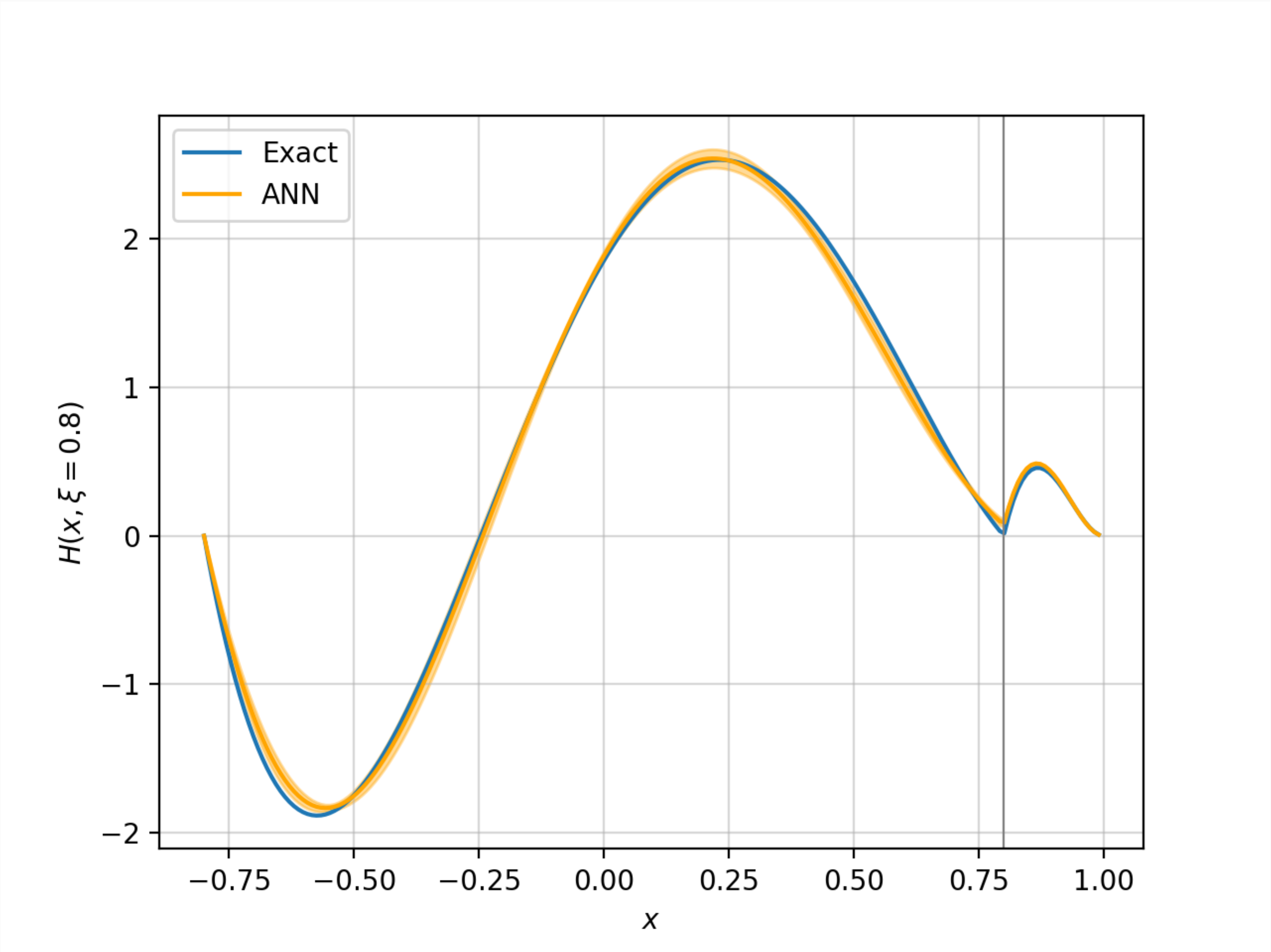




$N_{points} = 10^4\,, \quad N_{neurons} = 100\,,$ Dropout regularization $r = 0.0005\,,$ ADAM optimizer $lr = 0.0005$

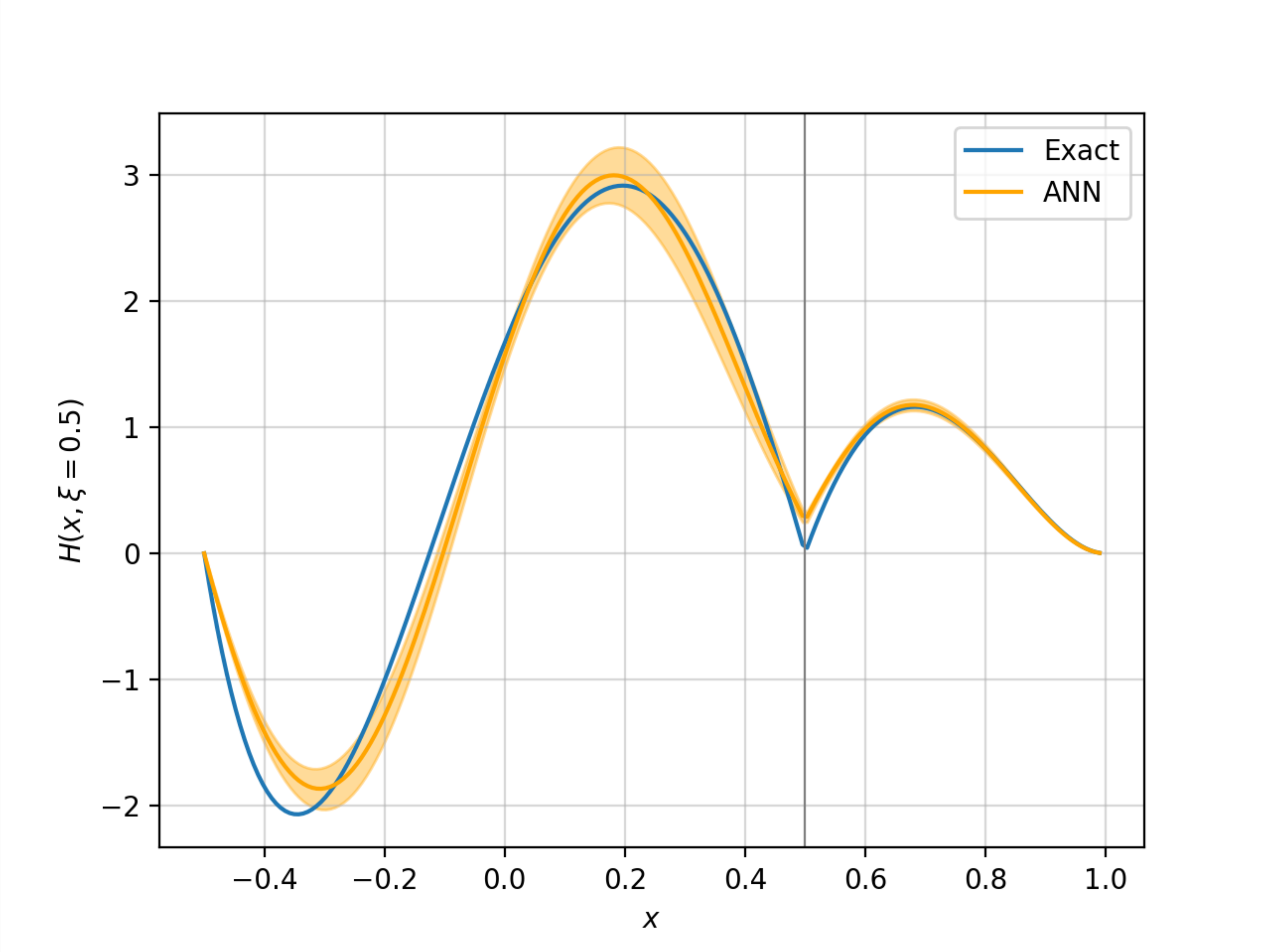
$$\xi^{max} = 0.75$$

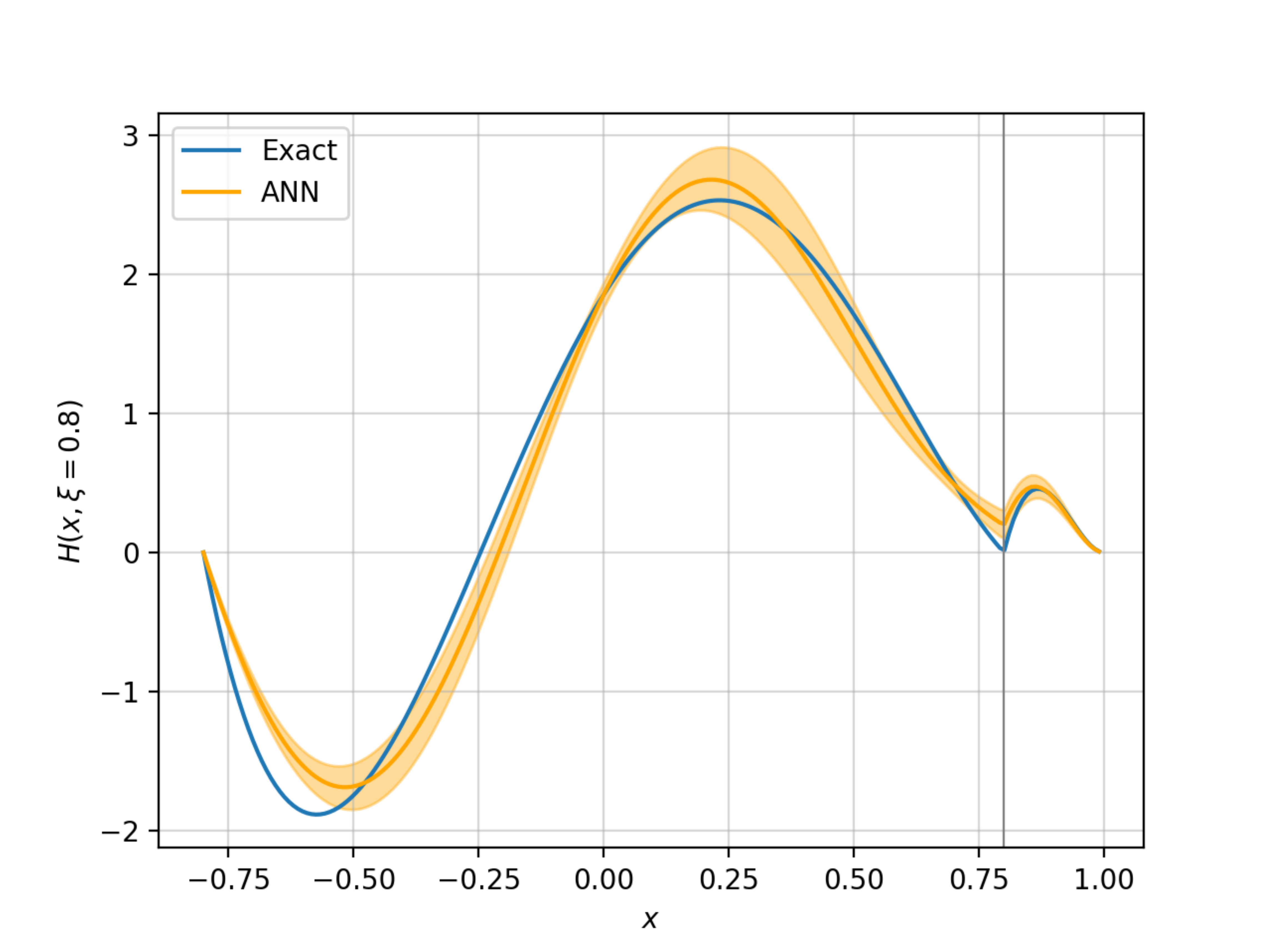




$N_{points} = 10^4\,, \quad N_{neurons} = 100\,,$ Dropout regularization $r = 0.00001\,,$ ADAM optimizer $lr = 0.0001$

$$\xi^{max} = 0.5$$





GK model

S.V. Goloskokov, P. Kroll, *Eur.Phys.J.C 50 (2007)*

$$H(x,\xi) = \int_{\Omega} d\beta d\alpha\, \delta\left(x - \beta - \alpha\xi\right) q(\beta)\, h_{GK}(\beta,\alpha)\,,$$

$$h_{GK}(\beta,\alpha) = \frac{\Gamma(n+2)}{2^{n+1}\Gamma^2(n+1)} \frac{\left[(1-\beta)^2 - \alpha^2\right]^n}{(1-\beta)^{2n+1}}$$

H. Dutrieux et al., *Eur.Phys.J.C 82 (2022)*

Output of the ANN hidden layer:

$$o_i = \varphi\left(w_{\beta i}\,\beta + w_{\alpha i}\frac{\alpha}{1-|\beta|} + b_i\right) - \varphi\left(w_{\beta i}\,\beta + w_{\alpha i} + b_i\right) + w_{\alpha i} \to -w_{\alpha i}$$

Imposes $h_{ANN}(\beta,\alpha) = 0$ on the boundary

(no bias parameter on the last layer)

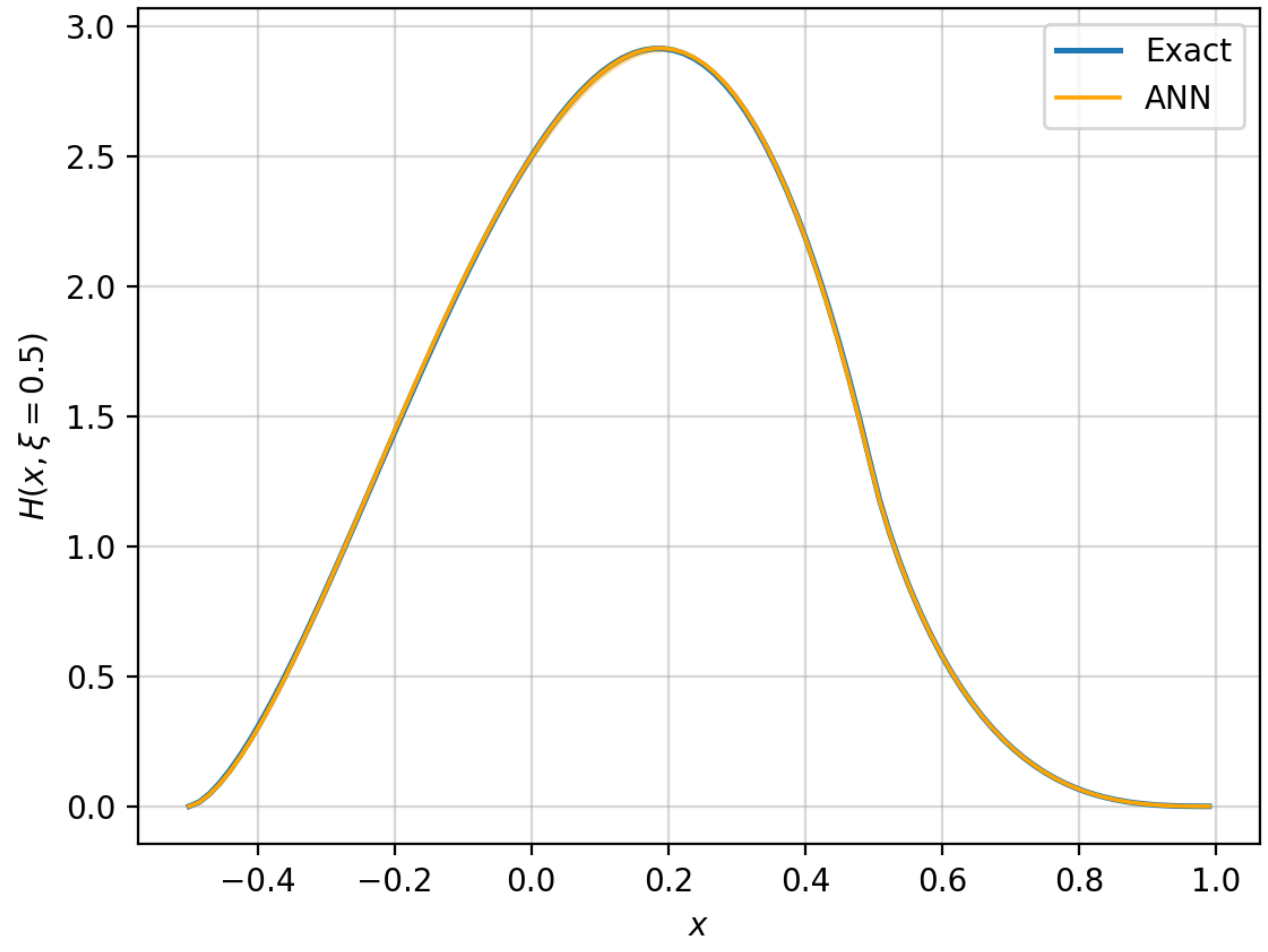
$$h_{GK}(\beta,\alpha) \simeq \frac{h_{ANN}(\beta,\alpha)}{\int_{-1+|\beta|}^{1-|\beta|} d\alpha\, h_{ANN}(\beta,\alpha)}$$

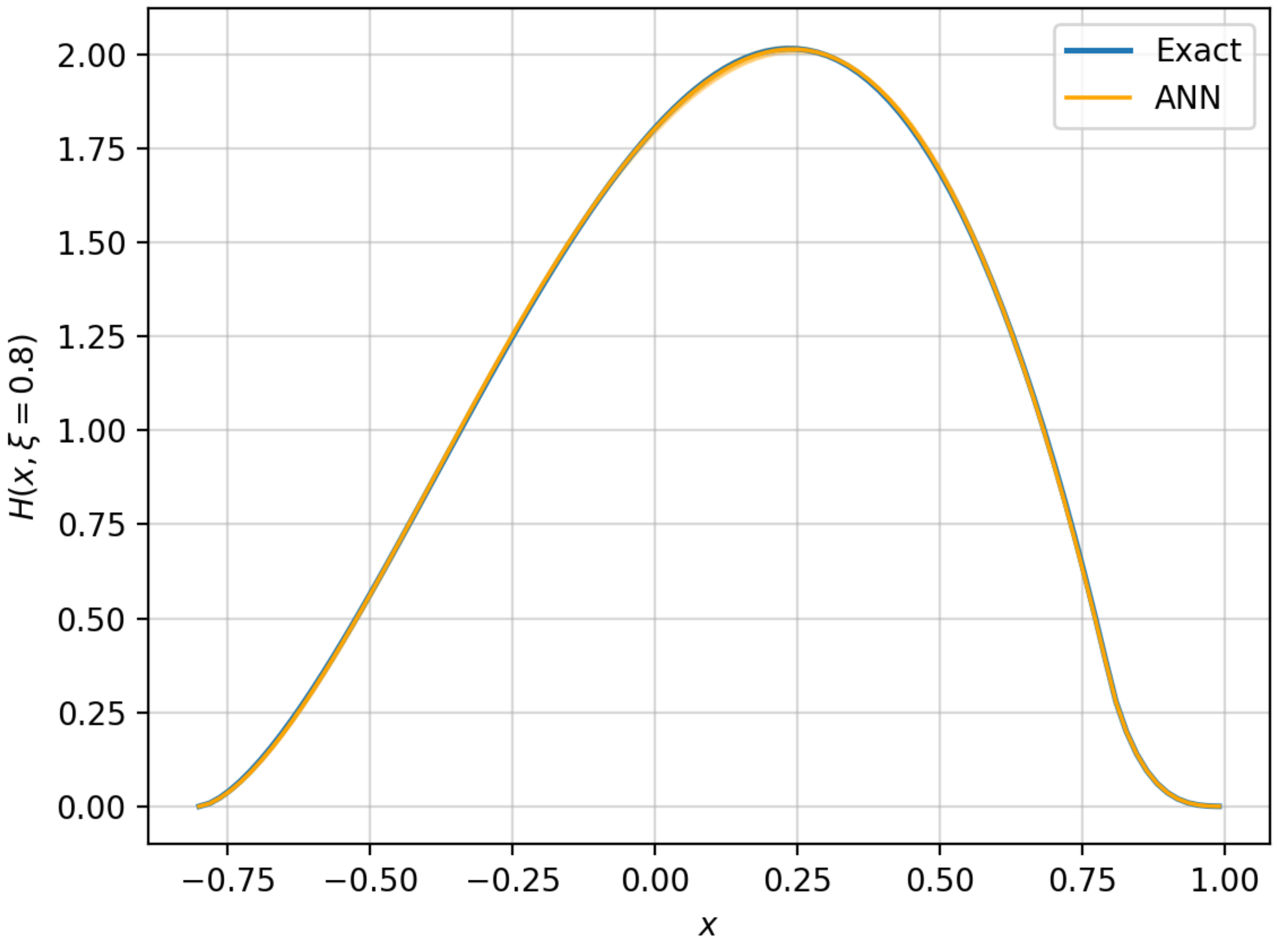
Imposes the correct normalization to recover the forward limit

Valence distribution $q_{val}^{u}(\beta) = \beta^{-\delta}(1-\beta)^{2n+1}\sum_{j=0}^{2} c_j\, \beta^{j/2}\,, \quad n = 1,\ \ \delta = 0.48$

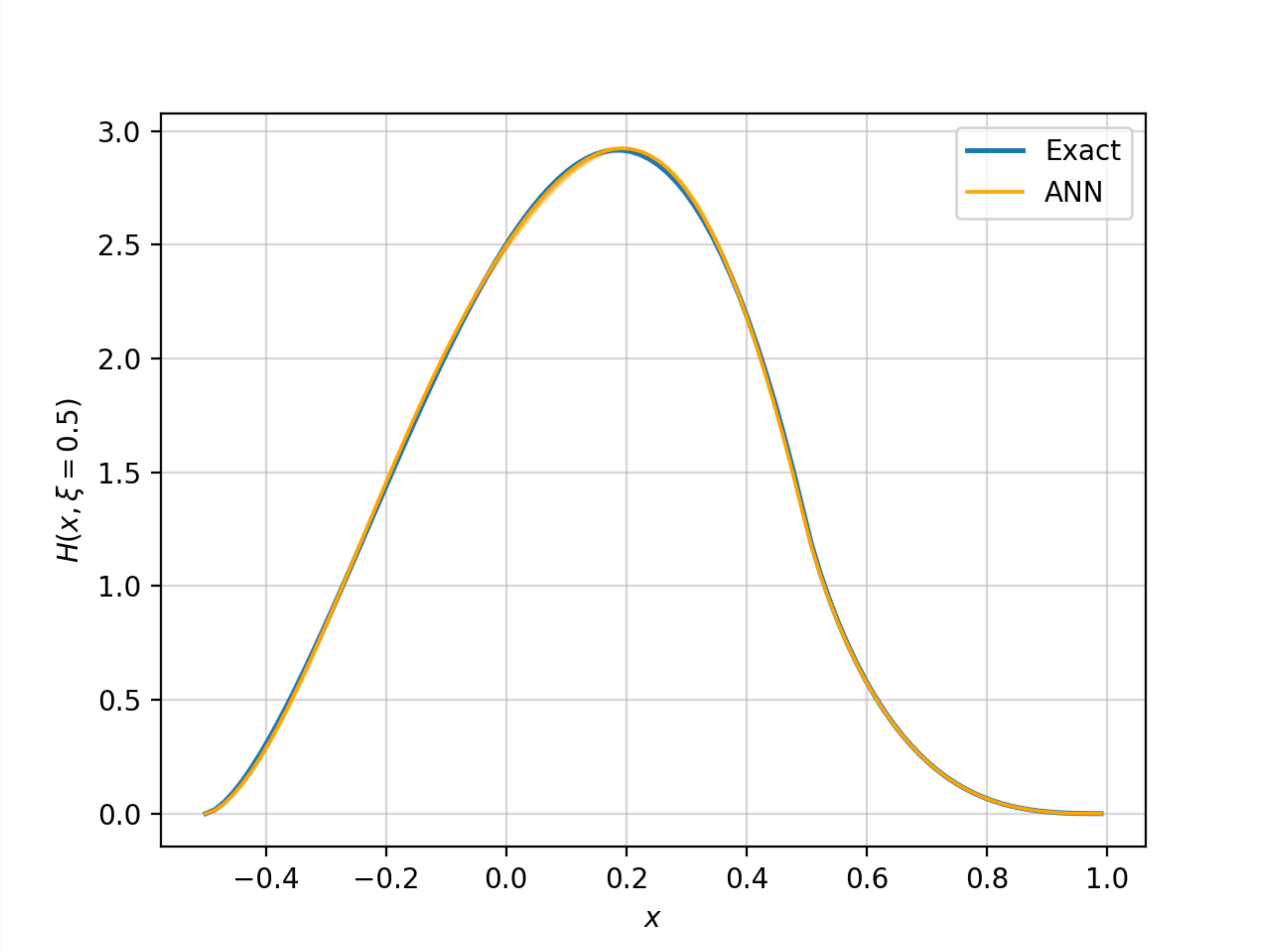
$N_{points} = 10^3\,, \quad N_{neurons} = 25\,,$ Dropout regularization $r = 0.1\,,$ ADAM optimizer $lr = 0.001$

$$\xi^{max} = 1$$

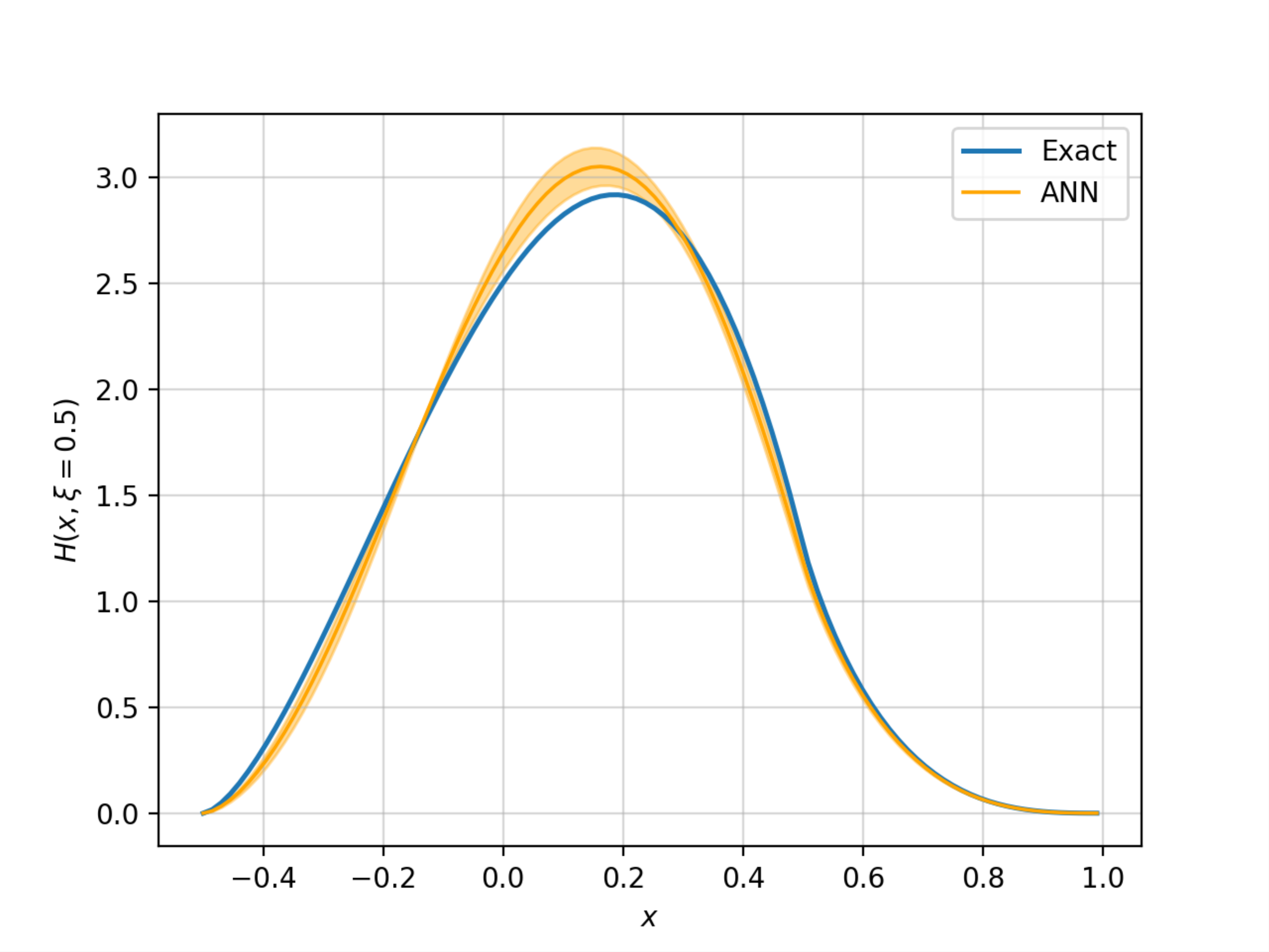




$\xi^{max} = 0.5\,, \quad r = 0.001$

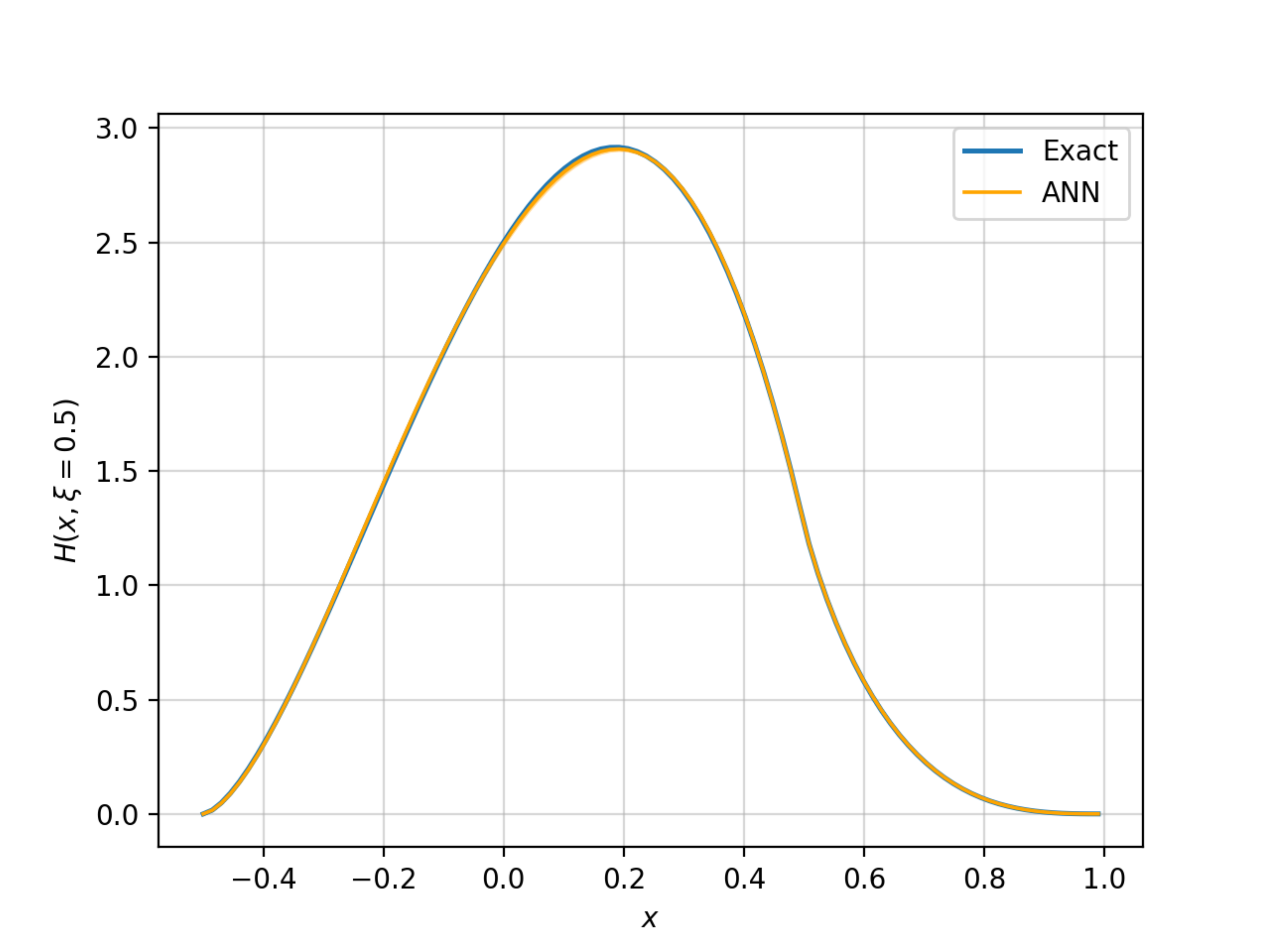


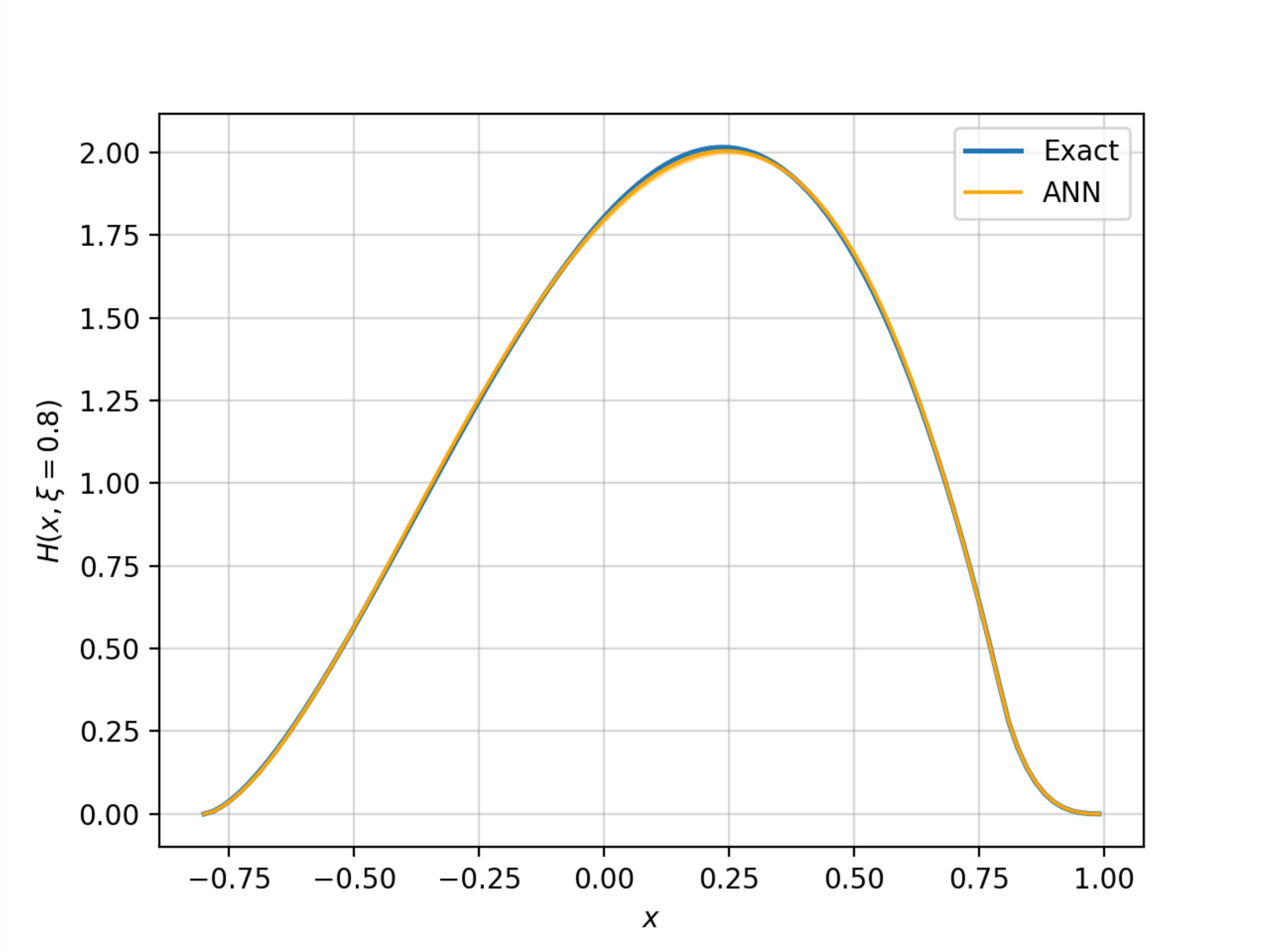
$\xi^{max} = 0.01\,, \quad r = 0$



ANN trained only with $x = \xi$ data (Compton form factor at first order)

$N_{points} = 10^3$, $N_{neurons} = 25$, Dropout regularization $r = 0.1$, ADAM optimizer $lr = 0.001$

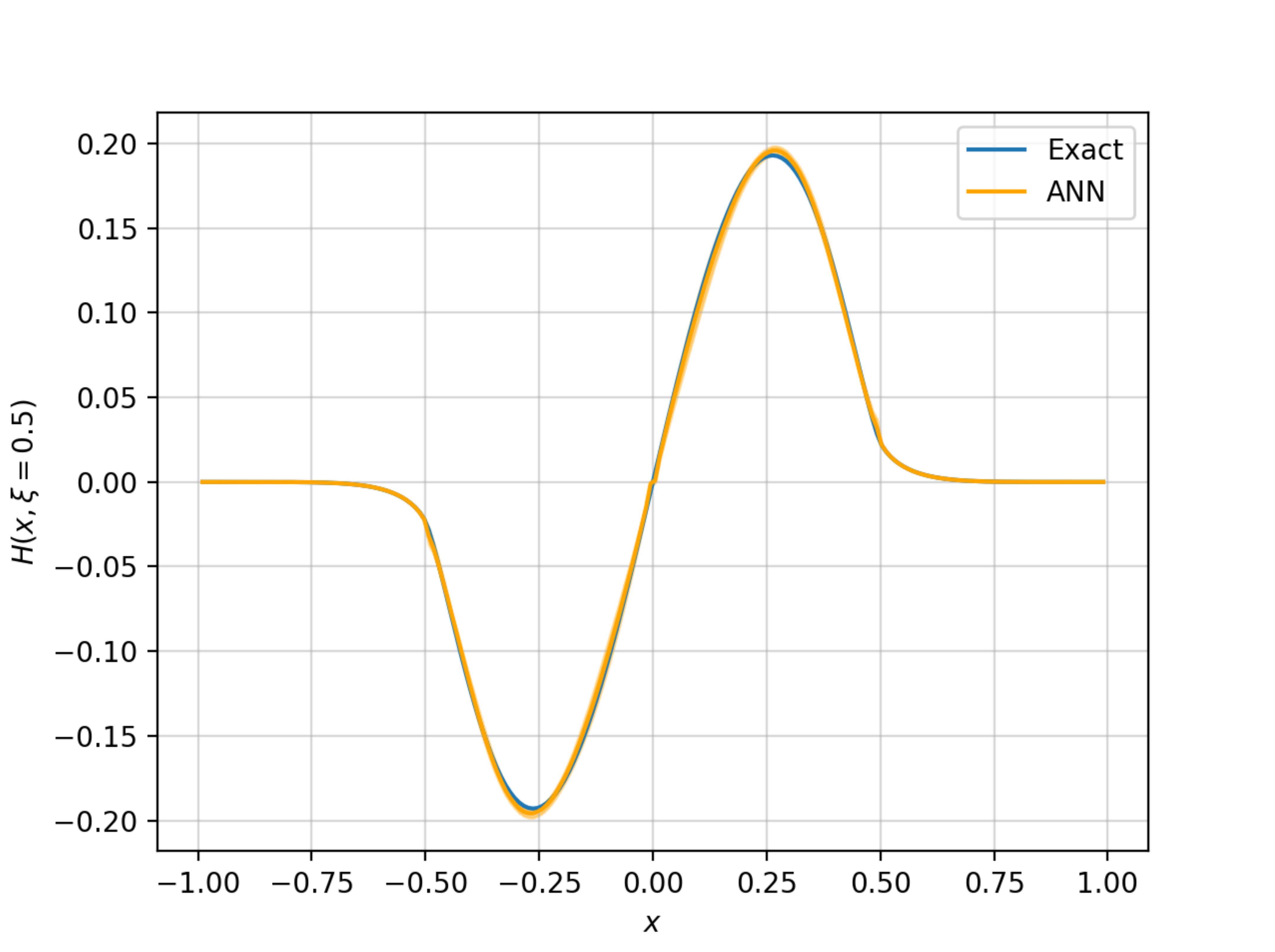


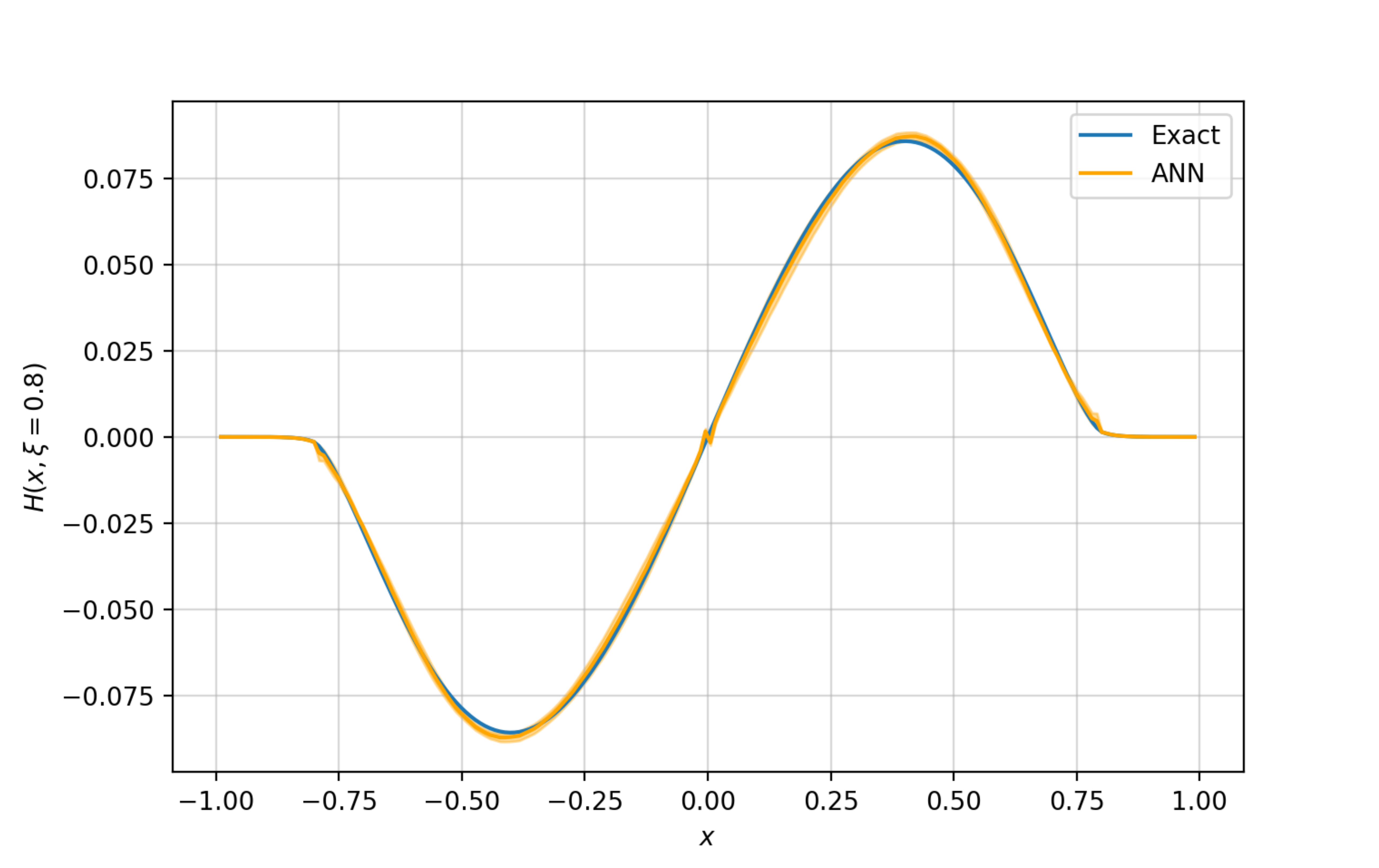


Sea distribution $q_{sea}(\beta) = \beta^{-\delta}(1-\beta)^{2n+1} \sum_{j=0}^{3} c_j\, \beta^{j/2}\,, \quad n = 2, \;\; \delta = 1.1$

$$H(x,\xi) = \int_\Omega d\beta d\alpha\, \delta\left(x - \beta - \alpha\xi\right)\; sign(\beta)\, q(|\beta|)\, h_{GK}(|\beta|, \alpha) \;\rightarrow H(-x,\xi) = -\,H(x,\xi)$$

$$\xi^{max} = 1$$





Conclusions

- ANN are a suitable tool for numerically solving inverse problems such as inverting the Radon Transform of a GPD.
- Different GPD models require different ANN parameters and setups (and their convergence speed can be very different).
- Inverting the Radon Transform using only data from a proper subset of its domain is feasible.
- The uncertainties regarding the GPD reconstructions may be lowered by a more careful tuning of the ANN parameters and by increasing the number of iterations.