Newton's method for solving PDEs with neural network discretization and applications

Wenrui Hao

Penn State University

CBMS Conference: Deep Learning and Numerical PDEs

Morgan State University

Machine learning for solving PDEs

- Neural network discretizations
  - ✓ Xu, J. (2020). The finite neuron method.
  - ✓ He, J., & Xu, J. (2019). MgNet: A unified framework of multigrid and convolutional neural network.
  - ✓ E, W., Yu, B. (2018). The deep Ritz method: a deep learning-based numerical algorithm for solving variational problems.
  - ✓ Mao, Z., Jagtap, A. D., & Karniadakis, G. E. (2020). Physics-informed neural networks for high-speed flows.
  - ✓ Chen, Y., & Koohy, S. (2023). GPT-PINN: Generative Pre-Trained Physics-Informed Neural Networks
  - ✓ Gu, Y., Yang, H., & Zhou, C. (2021). Selectnet: Self-paced learning for high-dimensional partial differential equations.
  - ✓ E, W., Han, J., & Jentzen, A. (2021). Algorithms for solving high dimensional PDEs: from nonlinear Monte Carlo to machine learning.
  - ✓ Guo, R., Cao, S., & Chen, L. (2022). Transformer meets boundary value inverse problems.

- Operator learning in PDEs

✓ Lu, L., Jin, P., & Karniadakis, G. E. (2019). Deeponet: Learning nonlinear operators for identifying differential equations
✓ Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., & Anandkumar, A. (2020). Fourier neural operator
✓ You, H., Zhang, Q., Ross, C. J., Lee, C. H., & Yu, Y. (2022). Learning deep implicit fourier neural operators (IFNOs)

- Training algorithms

✓ Siegel, J. W., Hong, Q., Jin, X., Hao, W., & Xu, J. (2021). Greedy Training Algorithms for Neural Networks for PDEs
✓ Wang, B., & Ye, Q. (2023). Improving Deep Neural Networks’ Training With Nonlinear Conjugate Gradient-Style Adaptive Momentum.

Problem setup

- We consider the following Laplace's equation

$$\begin{cases} -\Delta v = f(v) \quad in\ \Omega \\ \dfrac{\partial v}{\partial n} = 0 \quad on\ \partial\Omega \end{cases}$$

where $\Omega \subset R^d$ and $\partial\Omega$ is the boundary of the domain.

- Energy functional

$$\min_{w} J(\mathrm{w}) = \int_{\Omega} |\nabla w|^2 - G(w)dx$$

- Neural network discretizations

$$U(x;\theta) = W_n\sigma(W_{n-1}\cdots\sigma(W_2\sigma(W_1x + b_1) + b_2)\cdots + b_{n-1}) + b_n,$$

- How to solve $\boldsymbol{\theta}$ to get a numerical approximation?

Three different approaches

- Variational energy minimization

$$\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = \int_{\Omega} |\nabla u(x;\boldsymbol{\theta})|^2 - G(u(x;\boldsymbol{\theta}))dx$$

- L2 residual minimization

$$\min_{\boldsymbol{\theta}} \sum_i \left|\left|\Delta u(x_i;\boldsymbol{\theta}) + f\left(u(x_i;\boldsymbol{\theta})\right)\right|\right|^2 + \sum_j \left|\left| \frac{\partial u(x_j;\boldsymbol{\theta})}{\partial \mathrm{n}} \right|\right|^2 \Leftrightarrow \min_{\boldsymbol{\theta}} \left|\left|\boldsymbol{F}(\boldsymbol{\theta})\right|\right|_2^2$$

- A system of nonlinear equations

$$\boldsymbol{F}(\boldsymbol{\theta}) = \begin{cases} \Delta u(x_i;\boldsymbol{\theta}) + f\left(u(x_i;\boldsymbol{\theta})\right), & i = 1, \cdots, N \\ \dfrac{\partial u\left(x_j;\boldsymbol{\theta}\right)}{\partial \mathrm{n}}, j = 1, \cdots, n \end{cases} = \boldsymbol{0}$$

Gradient descent methods

- Variational energy minimization

$$\nabla_\theta L(\theta) = \int_\Omega \nabla u(x;\theta)\nabla_\theta \nabla u(x;\theta) - f\big(u(x;\theta)\big)\nabla_\theta u(x;\theta)dx$$

- L2 residual minimization

$$\nabla_\theta \big||\boldsymbol{F}(\theta)|\big|_2^2 = 2(\nabla_\theta \boldsymbol{F}(\theta))^T \boldsymbol{F}(\boldsymbol{\theta})$$

- Gradient descent method for solving nonlinear systems is written as

$$\theta^{k+1} = \theta^k - \eta^k (\nabla_\theta \boldsymbol{F}\big(\theta^k\big))^T \boldsymbol{F}\big(\boldsymbol{\theta^k}\big)$$

where $\eta^k = \frac{v^T \boldsymbol{F}\left(\boldsymbol{\theta^k}\right)}{v^T v}$ and $v = \nabla_\theta \boldsymbol{F}(\theta^k)(\nabla_\theta \boldsymbol{F}\big(\theta^k\big))^T \boldsymbol{F}\big(\boldsymbol{\theta^k}\big)$.

Hao, W. (2021). A gradient descent method for solving a system of nonlinear equations. *Applied Mathematics Letters*, *112*, 106739.

Newton’s methods

- Variational energy minimization

$$\nabla_\theta^2 L(\theta) = \int_\Omega \nabla_\theta \nabla u(x;\theta) \nabla_\theta \nabla u(x;\theta)^T - f'\big(u(x;\theta)\big) \nabla_\theta u(x;\theta) \nabla_\theta u(x;\theta)^T dx$$

$$+ \int_\Omega \nabla_\theta^2 \nabla u(x;\theta) \cdot \nabla u(x;\theta) - f\big(u(x;\theta)\big) \nabla_\theta^2 u(x;\theta) dx$$

- Newton’s method becomes

$$\theta^{k+1} = \theta^k - \nabla_\theta^2 L\left(\theta^k\right)^{-1} \nabla_\theta L\left(\theta^k\right)$$

Newton's methods

- The Hessian matrix of $L2\ minimization\ problem$ is

$$\nabla_\theta \boldsymbol{F}(\theta)^T \nabla_\theta \boldsymbol{F}(\theta) + \sum_{i=1}^{m} F_i(\theta) \nabla_\theta^2 F_i(\theta)$$

- Newton's method becomes

$$\theta^{k+1} = \theta^k - \left(\nabla_\theta \boldsymbol{F}\left(\theta^k\right)^T \nabla_\theta \boldsymbol{F}\left(\theta^k\right) + \sum_{i=1}^{m} F_i\left(\theta^k\right) \nabla_\theta^2 F_i\left(\theta^k\right)\right)^{-1} (\nabla_\theta \boldsymbol{F}\left(\theta^k\right))^T \boldsymbol{F}\left(\boldsymbol{\theta^k}\right)$$

Gauss Newton's methods

- The Hessian matrix of $L2\ minimization\ problem$ is

$$\nabla_\theta \boldsymbol{F}(\theta)^T \nabla_\theta \boldsymbol{F}(\theta) + \sum_{i=1}^{m} \boldsymbol{F}_i(\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}}^2 \boldsymbol{F}_i(\boldsymbol{\theta})$$

- Gauss Newton's method becomes

$$\theta^{k+1} = \theta^k - \left(\nabla_\theta \boldsymbol{F}(\theta^k)^T \nabla_\theta \boldsymbol{F}(\theta^k)\right)^{-1} (\nabla_\theta \boldsymbol{F}(\theta^k))^T \boldsymbol{F}(\boldsymbol{\theta}^k)$$

which is equivalent to solving the linearized least squares problem

$$p^k = arg \min_p \frac{1}{2} \left|\left|\nabla_\theta \boldsymbol{F}(\theta^k) p + \boldsymbol{F}(\theta^k)\right|\right|_2^2.$$

Newton’s methods

- Newton’s method for solving nonlinear system is

$$\theta^{k+1} = \theta^k - \left(\nabla_\theta \boldsymbol{F}(\theta^k)\right)^{-1} \boldsymbol{F}(\boldsymbol{\theta^k}).$$

- If the Jacobian matrix is not an invertible square matrix, we have

$$\theta^{k+1} = \theta^k - \left(\nabla_\theta \boldsymbol{F}(\theta^k)\right)^{+} \boldsymbol{F}(\boldsymbol{\theta^k}).$$

where $A^+ = (A^T A)^{-1} A^T$.

Gauss-Newton method

Example $F(x, y) = \begin{pmatrix} x^2 + y^2 - 1 \\ x + y - 1 \\ x + \varepsilon \end{pmatrix}$, GN is the Newton's method for solving nonlinear systems with small residual.

$\boldsymbol{\varepsilon}$	$\boldsymbol{x^0 = \left(\frac{1}{2}, \frac{1}{3}\right)^T}$	$\boldsymbol{x^0 = -\left(\frac{1}{2}, \frac{1}{3}\right)^T}$
0	6 steps	7 steps
10^{-2}	8 steps	10 steps
10^{-1}	11 steps	13 steps
1	19 steps	19 steps
5	647 steps	651 steps
10	Diverge	Diverge

Summary of Newton's method

	Variational formula	L2 minimization	Nonlinear system
Newton	$\theta^{k+1} = \theta^k - \nabla_\theta^2 L(\theta^k)^{-1} \nabla_\theta L(\theta^k)$	$\theta^{k+1} = \theta^k - \left(\nabla_{\theta\theta} \boldsymbol{F}(\theta^k)\right)^{-1} (\nabla_\theta \boldsymbol{F}(\theta^k))^T \boldsymbol{F}(\boldsymbol{\theta^k})$	$\theta^{k+1} = \theta^k - \left(\nabla_\theta \boldsymbol{F}(\theta^k)\right)^{+} \boldsymbol{F}(\boldsymbol{\theta^k})$
Gauss-Newton	??	$\theta^{k+1} = \theta^k - \left(\nabla_\theta \boldsymbol{F}(\theta^k)\right)^{+} \boldsymbol{F}(\boldsymbol{\theta^k})$	N/A

Gauss Newton's method

- Variational energy minimization

$$\boldsymbol{J(\theta)}$$

$$\nabla_\theta^2 L(\theta) = \int_\Omega \nabla_\theta \nabla u(x;\theta) \nabla_\theta \nabla u(x;\theta)^T - f'\big(u(x;\theta)\big) \nabla_\theta u(x;\theta) \nabla_\theta u(x;\theta)^T dx$$

$$+ \int_\Omega \boldsymbol{\nabla_\theta^2 \nabla u(x;\theta) \cdot \nabla u(x,\theta) - f\big(u(x;\theta)\big) \nabla_\theta^2 u(x;\theta) dx}$$

$$\boldsymbol{Q(\theta)}$$

- $Q(\theta) = \int_\Omega \nabla_\theta^2 u(x,\theta)(-\Delta u(x;\theta) - f\big(u(x;\theta)\big) dx + \int_{\partial\Omega} \nabla_\theta^2 u(x,\theta) \frac{\partial u(x,\theta)}{\partial n} ds$
- Since $\big||Q(\theta)|\big| \leq \varepsilon$, Gauss Newton's method becomes

$$\theta^{k+1} = \theta^k - J\big(\theta^k\big)^+ \nabla_\theta L\big(\theta^k\big)$$

Hao, W., Hong, Q., Jin, X. & Wang, Y. (2023). Gauss Newton method for solving variational problems of PDEs with neural network discretizaitons. Submitted, available at *arXiv preprint arXiv:2306.08727*.

The consistency between Gauss-Newton methods

- By applying the divergence theorem, we have

$$J(\theta) = \int_\Omega \nabla_\theta \nabla u(x;\theta) \nabla_\theta \nabla u(x;\theta)^T - f'\big(u(x;\theta)\big) \nabla_\theta u(x;\theta) \nabla_\theta u(x;\theta)^T dx$$

$$= \int_\Omega \nabla_\theta u(x;\theta)(-\nabla_\theta \Delta u(x;\theta) - f'\big(u(x;\theta)\nabla_\theta u(x;\theta)\big) dx + \int_{\partial\Omega} \nabla_\theta u(x;\theta) \frac{\partial \nabla_\theta u(x;\theta)^T}{\partial n} dS$$

- By applying numerical integrations, we have

$$\int_\Omega \nabla_\theta u(x;\theta)(-\nabla_\theta \Delta u(x;\theta) - f'\big(u(x;\theta)\nabla_\theta u(x;\theta)\big) dx$$

$$\approx \sum_i \boxed{w_i \nabla_\theta u(x_i;\theta)}(-\nabla_\theta \Delta u(x_i;\theta) - f'\big(u(x_i;\theta)\nabla_\theta u(x_i;\theta)\big)$$

$$\int_{\partial\Omega} \nabla_\theta u(x;\theta) \frac{\partial \nabla_\theta u(x;\theta)^T}{\partial n} dS \approx \sum_j \boxed{w_j \nabla_\theta u(x_j;\theta)} \frac{\partial \nabla_\theta u\left(x_j;\theta\right)^T}{\partial n}$$

- Then we have $J(\theta) = G \nabla_\theta \mathbf{F}(\theta), where\ G = [w_i \nabla_\theta u(x_i;\theta)]_{i=1}^{N+n}$

The consistency between Gauss-Newton methods

- Similarly, we can rewrite the gradient $\nabla_\theta L(\theta) = G\nabla_\theta F(\theta)$.
- Then we have $J(\theta)^+\nabla_\theta L(\theta)$= $\nabla_\theta F(\theta)^+(G^+G)\boldsymbol{F}(\boldsymbol{\theta})$.
- If G is column full rank, then we have $G^+G = I$.
- This is possible if the number of grid points N + n is less than or equal to the number of parameters $\dim(\theta)$.

Hao, W., Hong, Q., Jin, X. & Wang, Y. (2023). Gauss Newton method for solving variational problems of PDEs with neural network discretizaitons. Submitted, available at *arXiv preprint arXiv:2306.08727*.

Summary of Newton's method

	Variational formula	L2 minimization	Nonlinear system
Newton	$\theta^{k+1} = \theta^k - \nabla_\theta^2 L(\theta^k)^{-1} \nabla_\theta L(\theta^k)$	$\theta^{k+1} = \theta^k - \left(\nabla_{\theta\theta} \boldsymbol{F}(\theta^k)\right)^{-1} (\nabla_\theta \boldsymbol{F}(\theta^k))^T \boldsymbol{F}(\boldsymbol{\theta}^{\boldsymbol{k}})$	$\theta^{k+1} = \theta^k - \left(\nabla_\theta \boldsymbol{F}(\theta^k)\right)^+ \boldsymbol{F}(\boldsymbol{\theta}^{\boldsymbol{k}})$
Gauss-Newton	$\theta^{k+1} = \theta^k - \left(\nabla_\theta \boldsymbol{F}(\theta^k)\right)^+ G^+ G \boldsymbol{F}(\boldsymbol{\theta}^{\boldsymbol{k}})$	$\theta^{k+1} = \theta^k - \left(\nabla_\theta \boldsymbol{F}(\theta^k)\right)^+ \boldsymbol{F}(\boldsymbol{\theta}^{\boldsymbol{k}})$	N/A

Solution structure

All dimensional solutions for a given polynomial system:

- Points + Curves + Surfaces+…

$$f = \begin{pmatrix} (y - x^2)(x^2 + y^2 + z^2 - 1)(x - 2) \\ (z - x)(x^2 + y^2 + z^2 - 1)(y - 2) \\ (y - x^2)(z - x)(x^2 + y^2 + z^2 - 1)(z - 2) \end{pmatrix} = 0$$

The irreducible decomposition for Z=V(f) is:

$$Z = Z_2 \cup Z_1 \cup Z_0 = \{Z_{21}\} \cup \{Z_{11} \cup Z_{12} \cup Z_{13} \cup Z_{14}\} \cup \{Z_{01}\}$$

- Z_{21} is the sphere $x^2 + y^2 + z^2 - 1 = 0$
- Z_{11} is the line $(x = 2, z = 2)$
- Z_{12} is the line $(x = \sqrt{2}, y = 2)$
- Z_{13} is the line $(x = -\sqrt{2}, y = 2)$
- Z_{14} is the curve $(y = x^2, z = x)$
- Z_{01} is the point $(x = 2, y = 2, z = 2)$

Semi-regular zeros (positive dimensional zeros)

Regular zeros (isolated zeros)

Semi-regular zeros of $\nabla_{\theta} L(\theta) = 0$

- Dimension of zeros for a system of nonlinear equations.

Definition 1. *(Dimension of a Zero) Let θ^* be a zero of a smooth mapping $\nabla L : \Omega \subset \mathbb{R}^m \to \mathbb{R}^{N+n}$. If there is an open neighborhood $\Omega_z \subset \Omega$ of θ_* in $\mathbb{R}^m$ such that $\Omega_z \cap (\nabla L)^{-1}(\mathbf{0}) = \phi(\Lambda)$ where $\mathbf{z} \mapsto \phi(\mathbf{z})$ is a differentiable injective mapping defined in a connected open set Λ in $\mathbb{R}^k$ for a certain $k > 0$ with $\phi(\mathbf{z}_*) = \theta_*$ and $rank(\phi_{\mathbf{z}}(\mathbf{z}_*)) = k$, then the dimension of θ_* as a zero of ∇L is defined as*

$$dim_{\nabla L}(\theta_*) := dim(Range(\phi_{\mathbf{z}}(\mathbf{z}_*))) \equiv rank(\phi_{\mathbf{z}}(\mathbf{z}_*)) = k$$

- Example $x^2 + y^2 - 1 = 0$
  - We define $\phi(z) = (\cos(z), \sin(z))^T$. $\phi_z(z) = (-\sin(z), \cos(z))^T$, $rank(\phi_Z(Z)) = 1$

Semi-regular zeros of $\nabla_\theta L(\theta) = 0$

- Regularity of non-isolated zeros

Definition 2. *(Semiregular Zero) A zero $\theta^* \in \mathbb{R}^m$ of a smooth mapping $\theta \mapsto \nabla L(\theta)$ is semiregular if $dim_{\nabla L}(\theta^*)$ is well-defined and identical to nullity $(\mathbf{H}(\theta^*))$. Namely*

$$dim_{\nabla L}(\theta^*) + rank(\mathbf{H}(\theta^*)) = m.$$

- Example $x^2 + y^2 - 1 = 0$
  - $H(x, y) = (2x, 2y)^T$ and $\mathrm{rank}(\mathrm{H}(x, y)) = 1$

Semi-regular zeros of $\nabla_\theta L(\theta) = 0$

- **Finite element method case**: Let us consider the finite element space and define V_N as the set of functions that can be represented as a linear combination of the basis functions $\phi_i(x)$, where $a_i \in R$ and $i = 1 \cdots m$. Then we have $u(x;\theta) = \sum_i \alpha_i \phi_i(x)$. In this case, we have a linear system $\nabla L(\theta) = A\theta - g,$ where $\theta = (\alpha_1, \cdots, \alpha_m)$.
- **Regular zero**: if A is full rank, we have an isolated regular zero.
- **Semiregular zero**: if $\mathrm{rank}(A) = r < m$, ker(A)=span($\theta_1, \cdots, \theta_{m-r}$), then we define

$$\phi(z) = \theta = \theta^* + \delta_1\theta_1 + \cdots \sigma_{m-r}\theta_{m-r} \text{ and } z = (\delta_1, \cdots, \delta_{m-r}).$$

Therefore, $rank(A) + rank\big(\phi_z(z)\big) = m$

Semi-regular zeros of $\nabla_\theta L(\theta) = 0$

- **Neural network discretization**: we consider a simple neural network in 1D with domain $\Omega = (-1,1)$:

$$u(x;\theta) = \sum_i a_i ReLU^k(w_i x + b_i), a_i \in R, b_i \in [-1-\delta, 1+\delta], w_i \in \{-1,1\}.$$

- **Simple case of $m = 1$ and $N = 1$:** we consider $u(x,\theta) = a_1 ReLU(x + b_1)$ when $x_1 + b_1^* < 0$
- Semi-regular zero:

$$\nabla_\theta L(\theta) = \begin{bmatrix} 0 & w_1^b \frac{\partial u(x_1^b,\theta)}{\partial a_1} \\ 0 & w_1^b \frac{\partial u(x_1^b,\theta)}{\partial b_1} \end{bmatrix} \begin{bmatrix} F(x_1,\theta) \\ F(x_1^b,\theta) \end{bmatrix}$$

Let us define

$$\theta = (a_1, b_1)^T, \quad \theta^* = (a_1^*, b_1^*)^T, \quad \theta_1 = (0,1)^T$$

and

$$\phi(\mathbf{z}) = \theta^* + \delta_1\theta_1 = \theta^* + \delta_1\theta_1 = \theta^* + \delta_1(0,1)^T \quad \text{with} \quad \mathbf{z} = \delta_1.$$

$$\text{rank}(\phi_z(\mathbf{z})) + \text{rank}(\mathbf{J}(\theta^*)) = 2,$$

Hao, W., Hong, Q., Jin, X. & Wang, Y. (2023). Gauss Newton method for solving variational problems of PDEs with neural network discretizaitons. Submitted, available at *arXiv preprint arXiv:2306.08727*.

Convergence analysis

Theorem Let $L(\theta)$ be a sufficiently smooth target function of θ. Then for every open neighborhood Ω_1 of θ^*, the sequence generated by Gauss Newton converges in Ω_1. Furthermore, the sequence has at least linear convergence.

Sketch of proof

h<1

$$\|\theta_{k+1} - \theta_k\| = \|J(\theta_k)^\dagger \nabla_\theta L(\theta_k)\| \leq \|J(\theta_k)^\dagger\| \left(\alpha\|\theta_k - \theta_{k-1}\| + \zeta\|\nabla_\theta L(\theta_{k-1})\| + \epsilon\right) \|\theta_k - \theta_{k-1}\|$$

$$\|\theta_{k+1} - \theta_k\| < h\|\theta_k - \theta_{k-1}\| < \|\theta_k - \theta_{k-1}\|$$

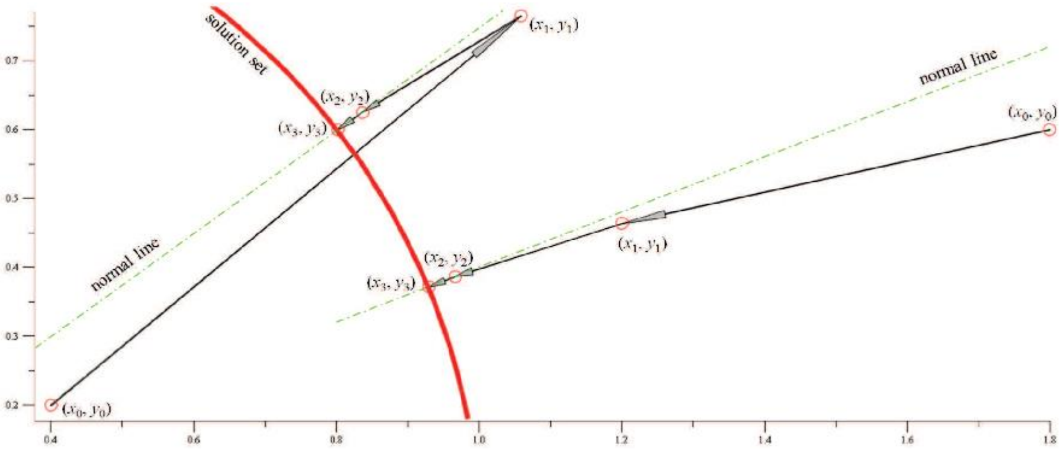
$$\|\theta_{k+1} - \theta^*\| \leq \|\theta_0 - \theta^*\| + \sum_{j=0}^{k} \|\theta_{k-j+1} - \theta_{k-j}\| < \frac{1}{1-h}\frac{h-1}{2}\delta + \frac{1}{2}\delta = \delta,$$

Convergence analysis

Sketch of proof

$$\|\theta_{k+1} - \theta_k\| \le \beta\|\theta_k - \theta_{k-1}\|^2 + \epsilon\|\theta_k - \theta_{k-1}\| = (\beta\|\theta_k - \theta_{k-1}\| + \epsilon)\,\|\theta_k - \theta_{k-1}\|$$

$$\|\theta_{k+1} - \hat{\theta}\| \le \frac{h}{1-h} \le (2\epsilon)^k\|\theta^1 - \theta^0\|.$$



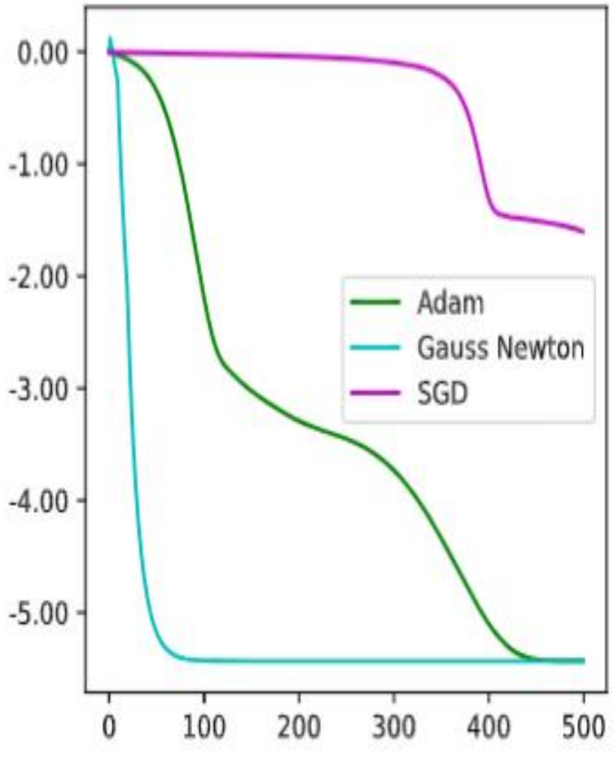
Numerical examples

We consider the following differential equation

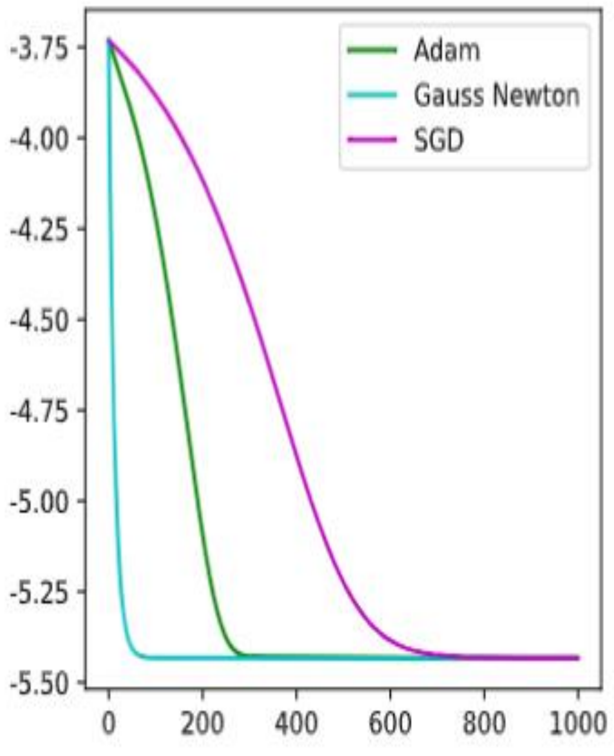
$$\begin{cases} -u_{xx} + u = (\pi^2 + 1)\cos(\pi x) \\ u_x(-1) = u_x(1) = 0 \end{cases}$$

Hidden Layer Width	32	64	128	256	512
L_2 Error	6.33E-3	5.82E-3	3.97E-3	2.50E-3	2.38E-3
H_1 Error	5.44E-2	5.17E-2	4.14E-2	3.55E-2	3.37E-2

Table 1: Errors corresponding to different widths of the hidden layer.



(a) adaptive step size



(b) fixed step size

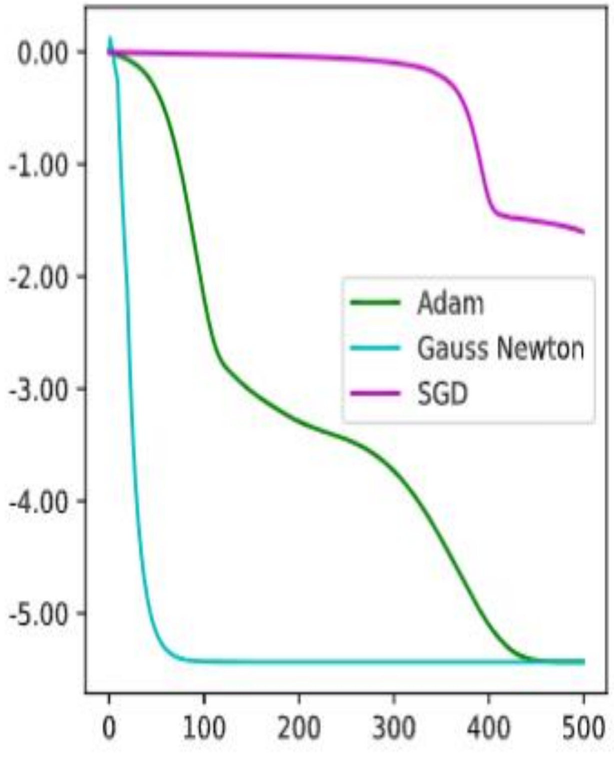
Numerical examples

We consider the following differential equation

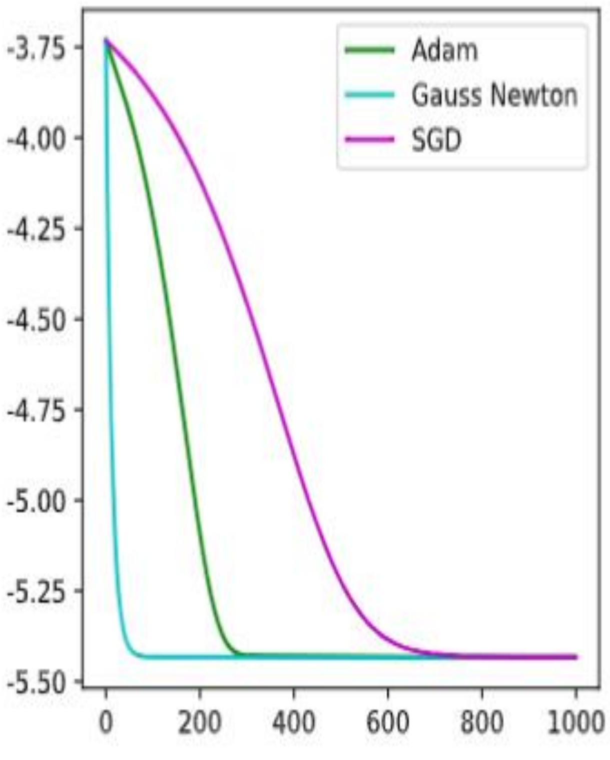
$$\begin{cases} -u_{xx} + u = (\pi^2 + 1)\cos(\pi x) \\ u_x(-1) = u_x(1) = 0 \end{cases}$$

Hidden Layer Width	32	64	128	256	512
L_2 Error	6.33E-3	5.82E-3	3.97E-3	2.50E-3	2.38E-3
H_1 Error	5.44E-2	5.17E-2	4.14E-2	3.55E-2	3.37E-2

Table 1: Errors corresponding to different widths of the hidden layer.



(a) adaptive step size

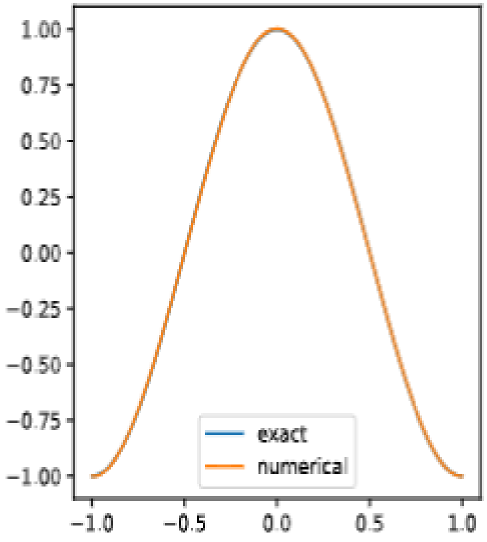


(b) fixed step size

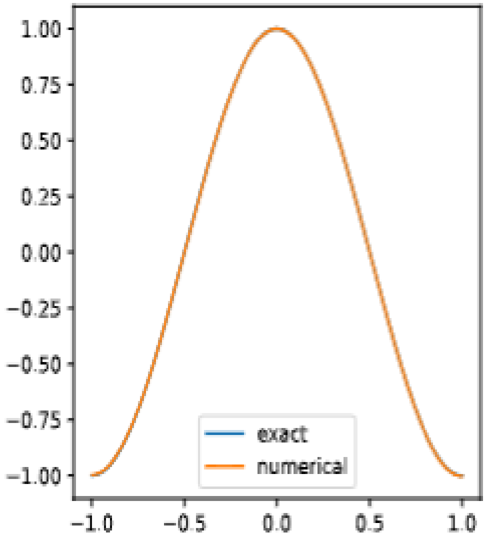
Numerical examples

We confirm the consistency of two Gauss-Newton methods for variational and L2 minimization problems.

N=100, m=128 ,$\dim(\theta) = 384$

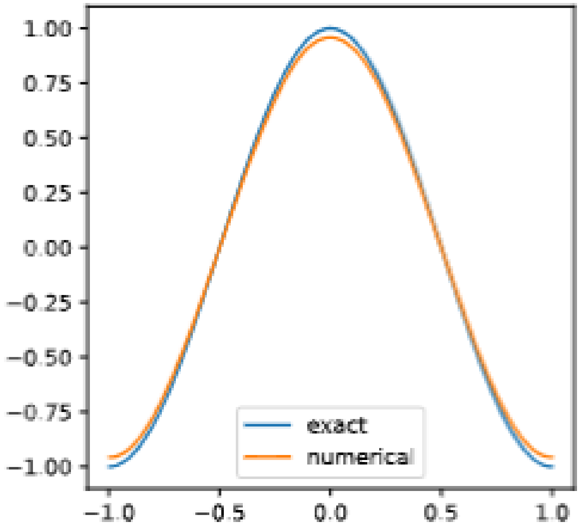


Variational form

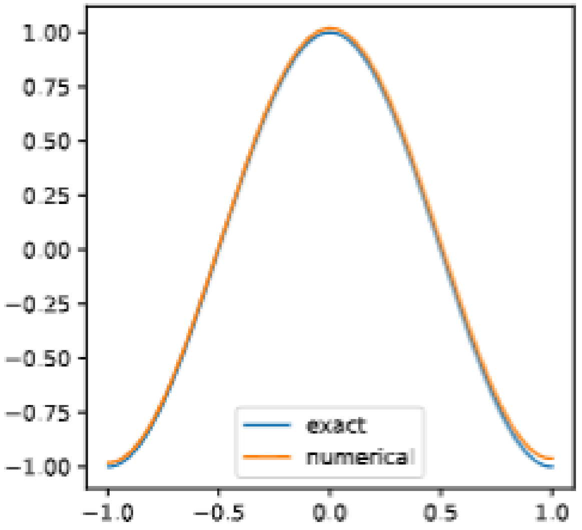


L2 minimization

N=500, m=128,$\dim(\theta) = 384$



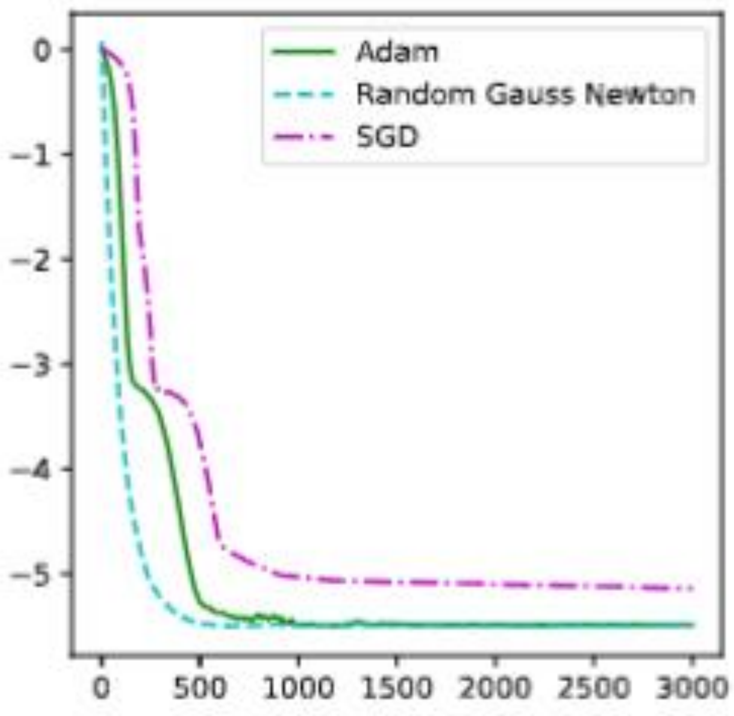
Variational form



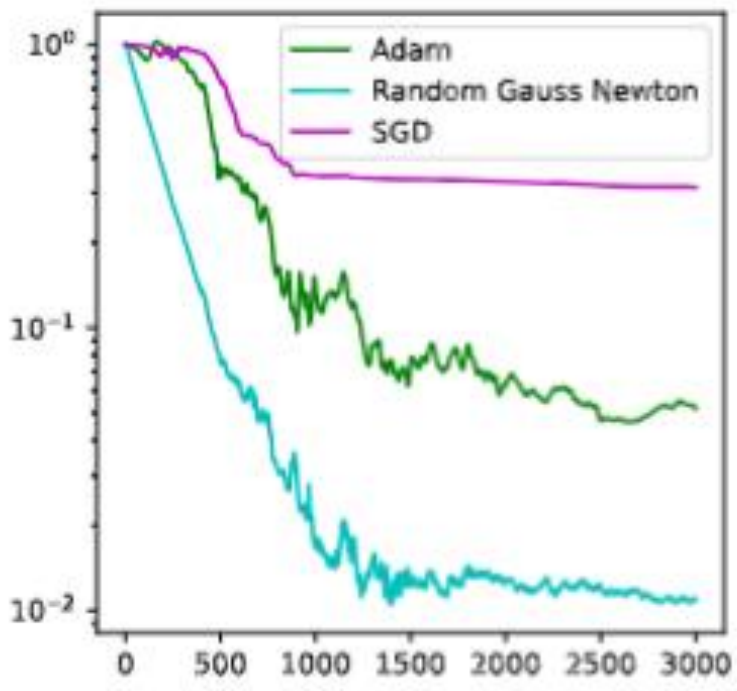
L2 minimization

Numerical examples

We consider the random Gauss-Newton's method

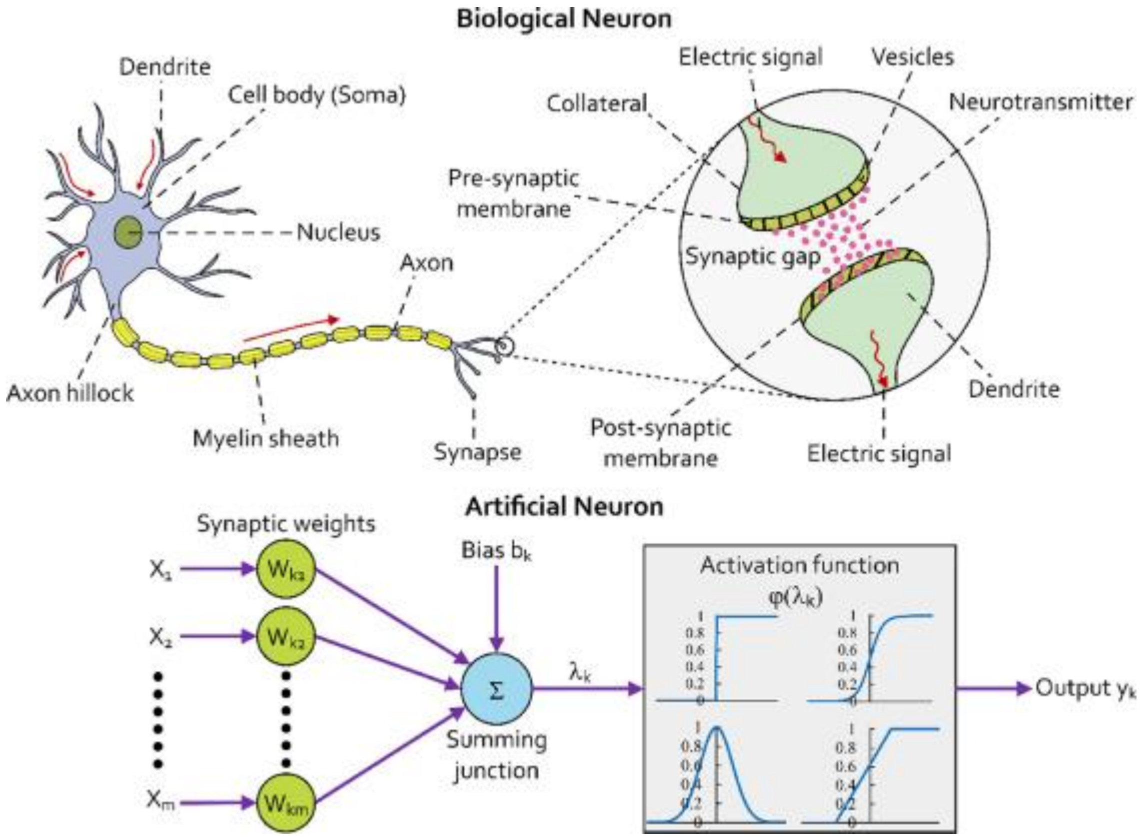


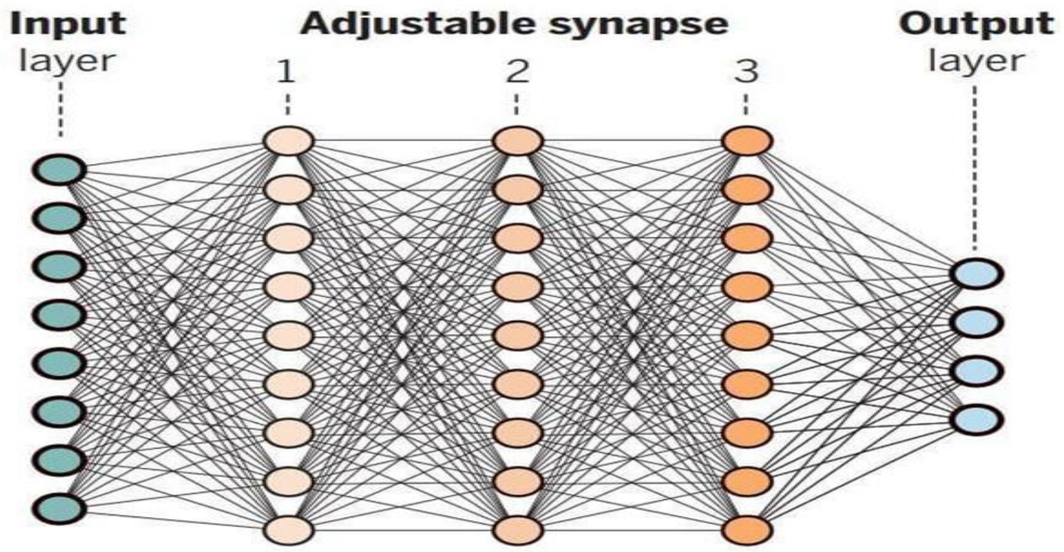
(a) Variational loss



(b) L2 error

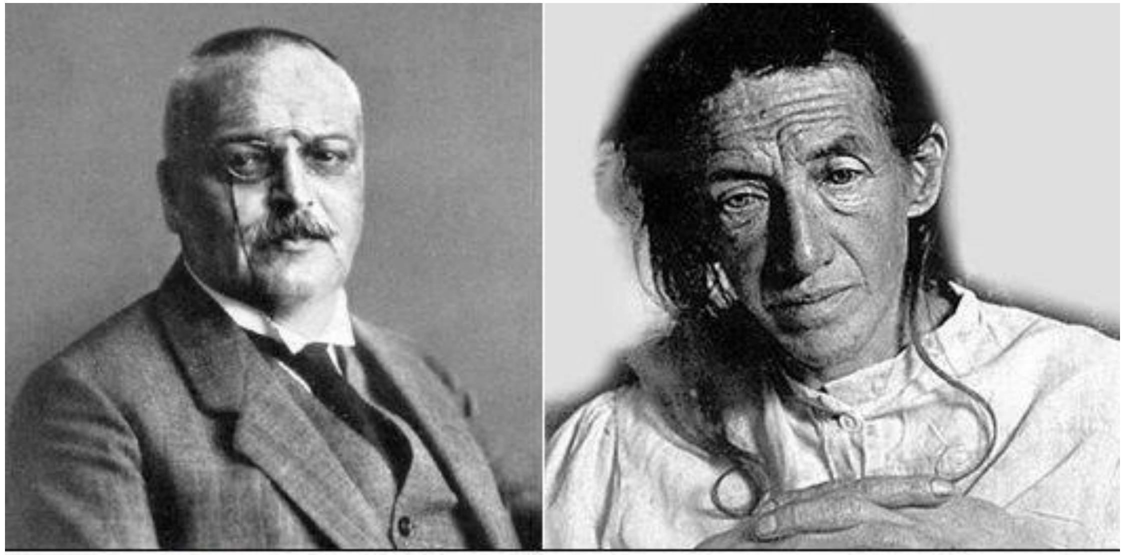
Artificial neural networks



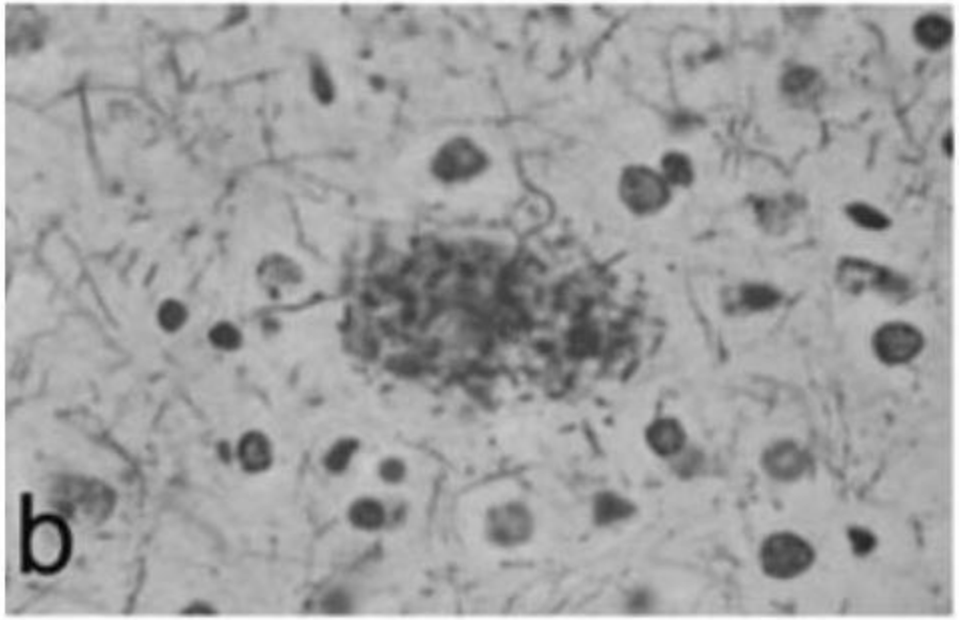


Aghbashlo, Mortaza, et al. "Machine learning technology in biodiesel research: A review." *Progress in Energy and Combustion Science* 85 (2021): 100904.

Application to Alzheimer’s disease

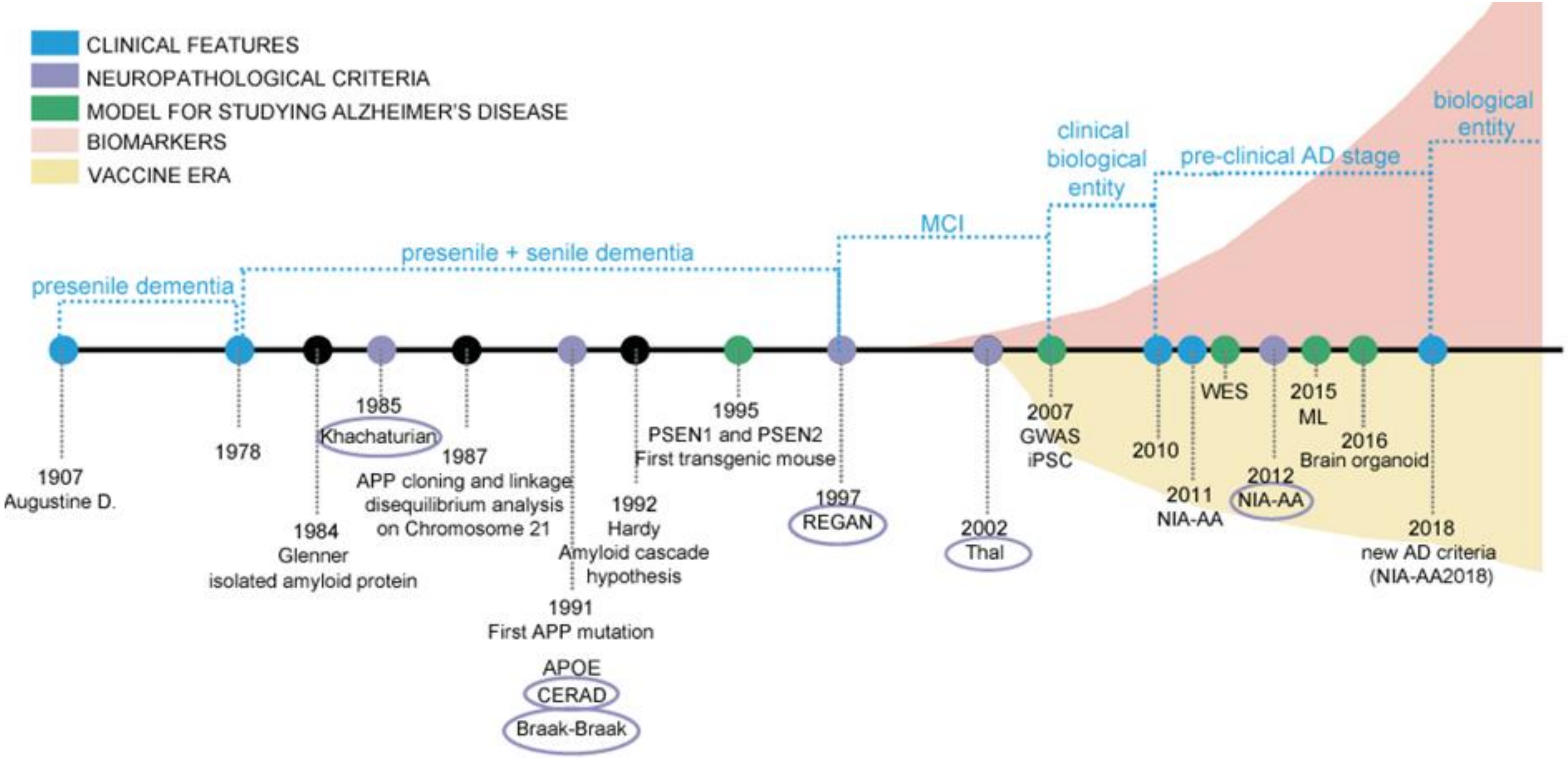


Alois Alzheimer Auguste Deter



Dr. Alois Alzheimer (1864-1915) was the physician who first reported on a patient with dementia, later termed as "Alzheimer's Disease."

The history of Alzheimer's disease



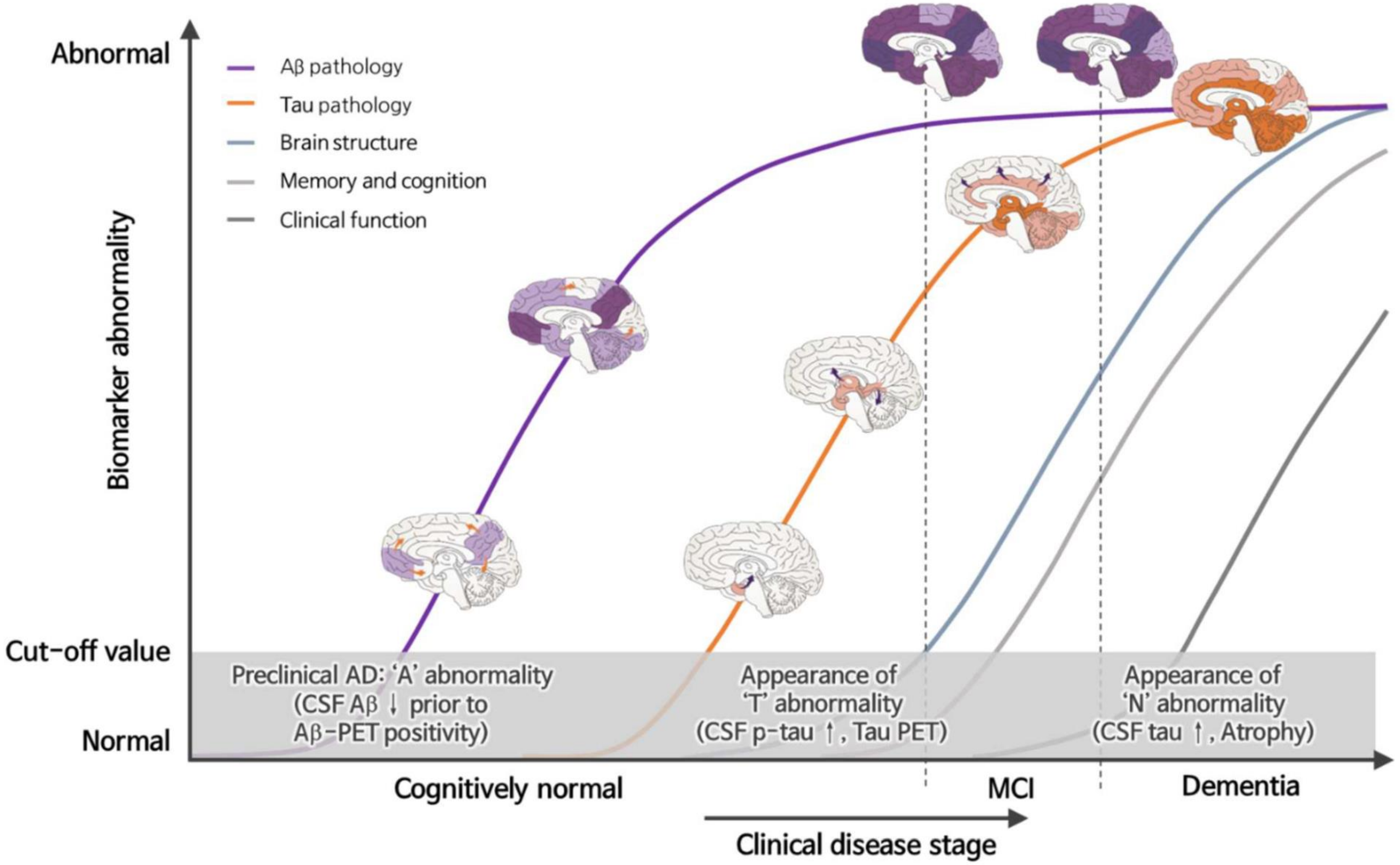
Ferrari, C, Sorbi, S **(2021)**. The complexity of Alzheimer's disease: an evolving puzzle. *Physiol Rev*, 101, 3:1047-1081.

AD clinical biomarkers

	A Amyloid	B Tau	N Neuroinflammation
CSF	↓ $A\beta_{42}$ ↓ $A\beta_{42}/A\beta_{40}$ ratio	↑ T-Tau ↑ p-Tau	↑ Neurofilament light chain (NFL)
Plasma	Controversial	↑ T-Tau	↑ Neurofilament light chain (NFL)
Imaging	Amyloid PET (PIB, ^{18}F-florbetapir)	Tau PET (18F-flortaucipir, 18F-RO-948)	• FDG-PET (hypometabolism) • MRI (atrophy)

Janeiro, M. H., Ardanaz, C. G., Sola-Sevilla, N., Dong, J., Cortés-Erice, M., Solas, M., ... & Ramírez, M. J. (2021). Biomarkers in Alzheimer's disease. *Advances in Laboratory Medicine/Avances en Medicina de Laboratorio*, *2*(1), 27-37.

Biomarkers dynamics

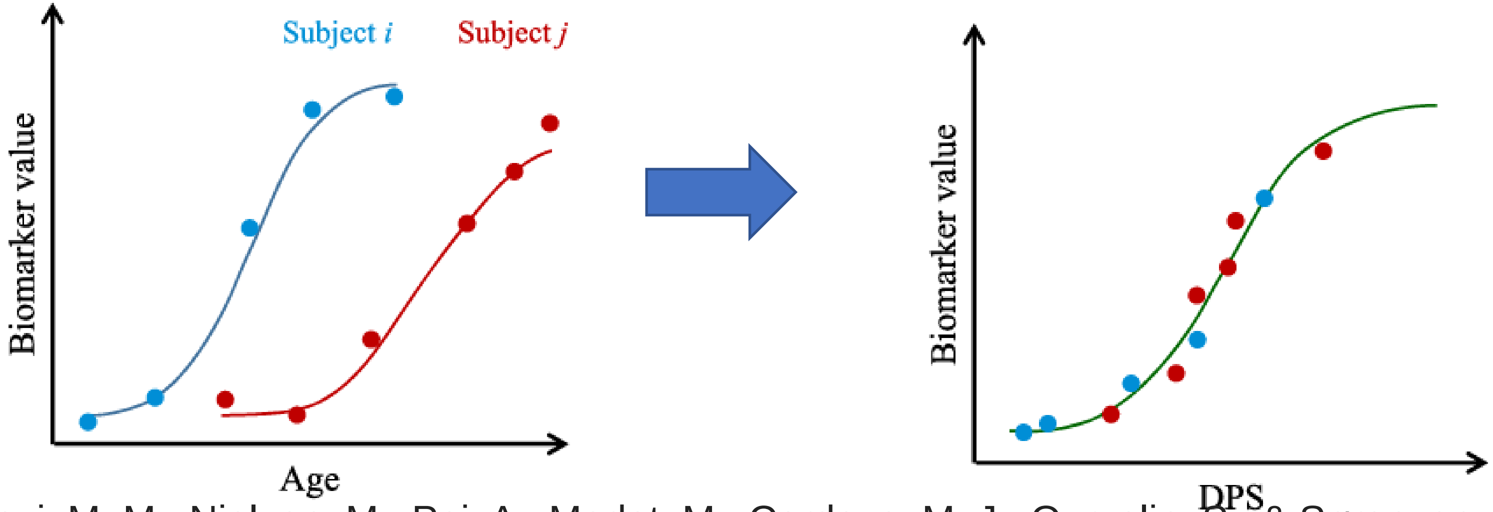


Model the dynamics of biomarkers

- $x(t) = sigmoid(t)$
- Logistic growth model

$$\frac{dx}{dt} = \lambda x\left(1 - \frac{x}{K}\right)$$

Verhulst	$[1+e^{-b(s-c)}]^{-1}$
Gompertz	$e^{-e^{-b(s-c)}}$
Richards	$[1+\gamma e^{-b(s-c)}]^{-1/\gamma}$
Modified Stannard	$\left[1+\frac{1}{\gamma}e^{-\frac{b}{\gamma}(s-c)}\right]^{-\gamma}$

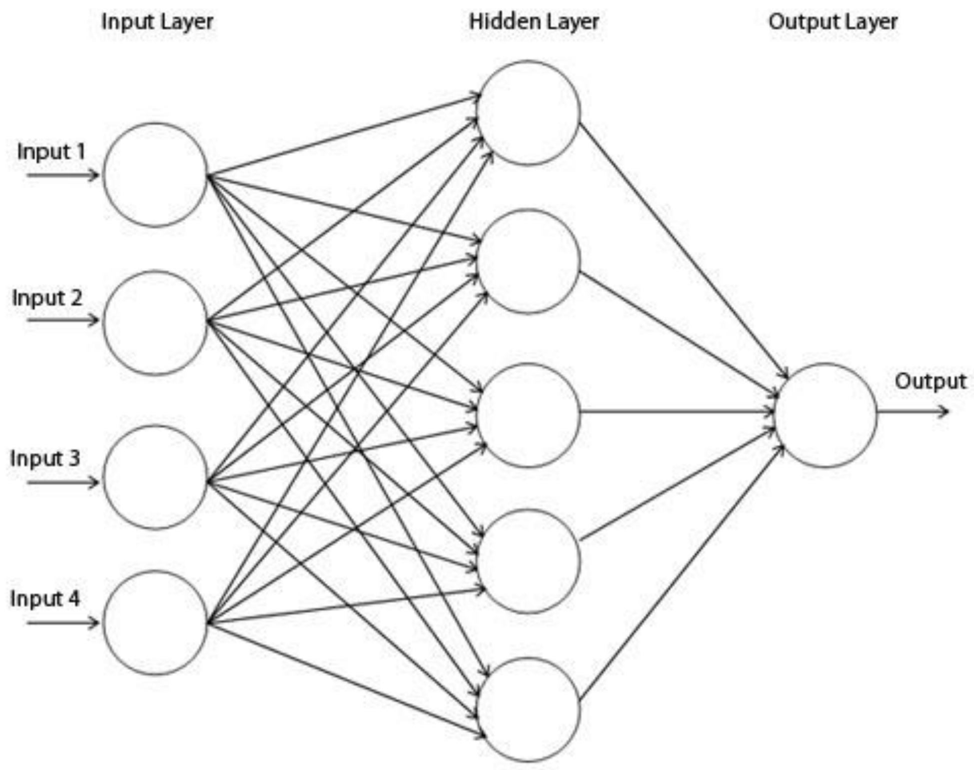


$$DPS_i = \alpha_i t + \beta_i$$

Ghazi, M. M., Nielsen, M., Pai, A., Modat, M., Cardoso, M. J., Ourselin, S., & Sørensen, L. (2021). Robust parametric modeling of Alzheimer's disease progression. *NeuroImage*, *225*, 117460.

Mathematical models on the population level

- We can write the model as follows
- $x(t) = \sum w_i \sigma(\alpha_i t + \beta_i)$



Model verification

- We learn a system of ODEs by using the clinical data

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{G}(\boldsymbol{x}), \boldsymbol{x} = \left(A_\beta, \tau_p, \tau_o, N, C\right)^{\boldsymbol{T}}.$$

- We use a polynomial basis to approximate the right-hand side, namely,

$$\boldsymbol{G}(\boldsymbol{x}, \boldsymbol{p}) = \sum_{\ell} w_\ell \boldsymbol{\phi}(\boldsymbol{x}).$$

- The system of ODEs can be rewritten as

$$\boldsymbol{x}(t) - \boldsymbol{x}(0) = \sum_{\ell} w_\ell \int_0^t \boldsymbol{\phi}\big(\boldsymbol{x}(s)\big) ds.$$

Zheng, H, Petrella, JR, Doraiswamy, PM, Lin, G, Hao, W **(2022)**. Data-driven causal model discovery and personalized prediction in Alzheimer's disease. *NPJ Digit Med*, 5, 1:137.

Model verification

- By choosing $t = t_i$, and denoting

$$D = \left(\int_0^{t_i} \boldsymbol{\phi}\big(\boldsymbol{x}(s)\big)ds\right), b = (\boldsymbol{x}(t_i) - \boldsymbol{x}(0)),$$

we have the following Lasso optimization

$$\min_{w} \big||Dw - b|\big| + \lambda\big||w|\big|_1,$$

where $\big||w|\big|_1$ enforces the sparsity.

Order	Biomarker					
	Abeta	Tau	Hippo	High-risk ADAS	Low-risk ADAS	Sum of MSE
1	4.84E-5	5.60E-4	2.41E-4	8.57E-5	8.06E-5	1.02E-03
2	**1.11E-6**	**4.86E-5**	**1.96E-5**	**1.99E-5**	**2.11E-4**	**3.00E-04**
3	4.71E-7	7.46E-4	2.34E-5	5.81E-5	5.36E-4	1.36E-03
4	4.24E-7	1.59E-3	2.24E-5	8.07E-5	1.49E-2	1.66E-02
5	4.25E-7	2.58E-4	1.45E-2	3.45E-3	4.05E+0	4.07E+00

Disease progression scores

- For different subjects in ADNI, the onset of disease and rate of progression are different within and among subject classes of CN, LMCI and AD.
- We introduce DPS $s_i(t)$ as a linear function of the patient's age t for each patient: $s_i(t) = \alpha_i + \beta_i t$.
- The parameters of the ODE model are fitted based on the ADNI dataset by minimizing the sum of squared differences between the data and the solution of the causal model, namely

$$\min_{w_k} \sum_{(i,j)\in I_k} \left(\tilde{x}_{ijk} - x_k(\alpha_i + \beta_i t; w_k)\right)^2$$

where $\tilde{x}_{ijk}$ is k-th biomarker data for i-th patient at j-th visit.

Disease progression scores

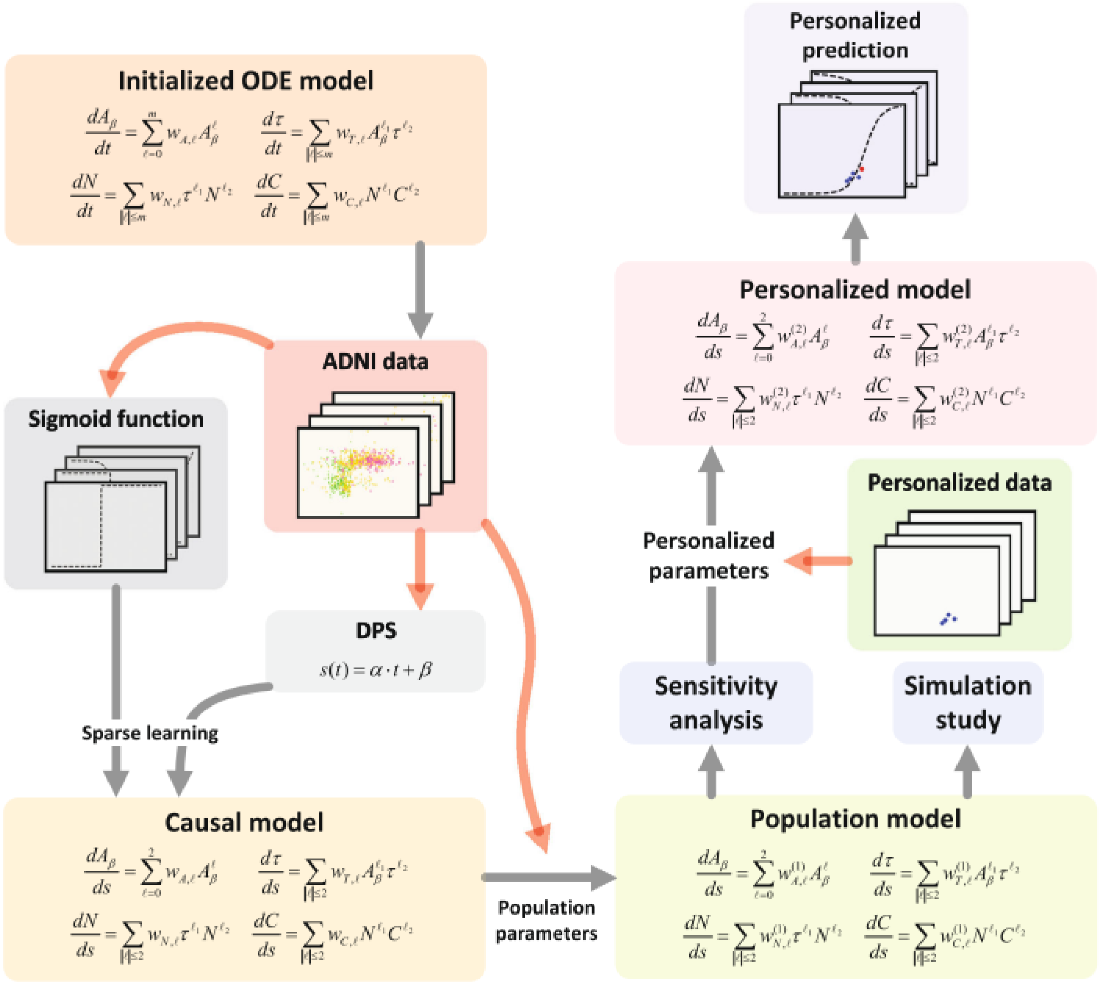
- Since the biomarkers for each patient will generally increases or decreases monotonically, we consider fitting DPSs as a least square linear regression problem, namely,

$$\min_{\alpha_i,\beta_i} \sum_{(j,k)\in I_i} \sigma_k \left(\tilde{x}_{ijk} - x_k(\alpha_i + \beta_i t; w_k)\right)^2$$

where σ_k is the normalization constant for k-th biomarker.

Zheng, H, Petrella, JR, Doraiswamy, PM, Lin, G, Hao, W **(2022)**. Data-driven causal model discovery and personalized prediction in Alzheimer's disease. *NPJ Digit Med*, 5, 1:137.

Data-driven modeling approach

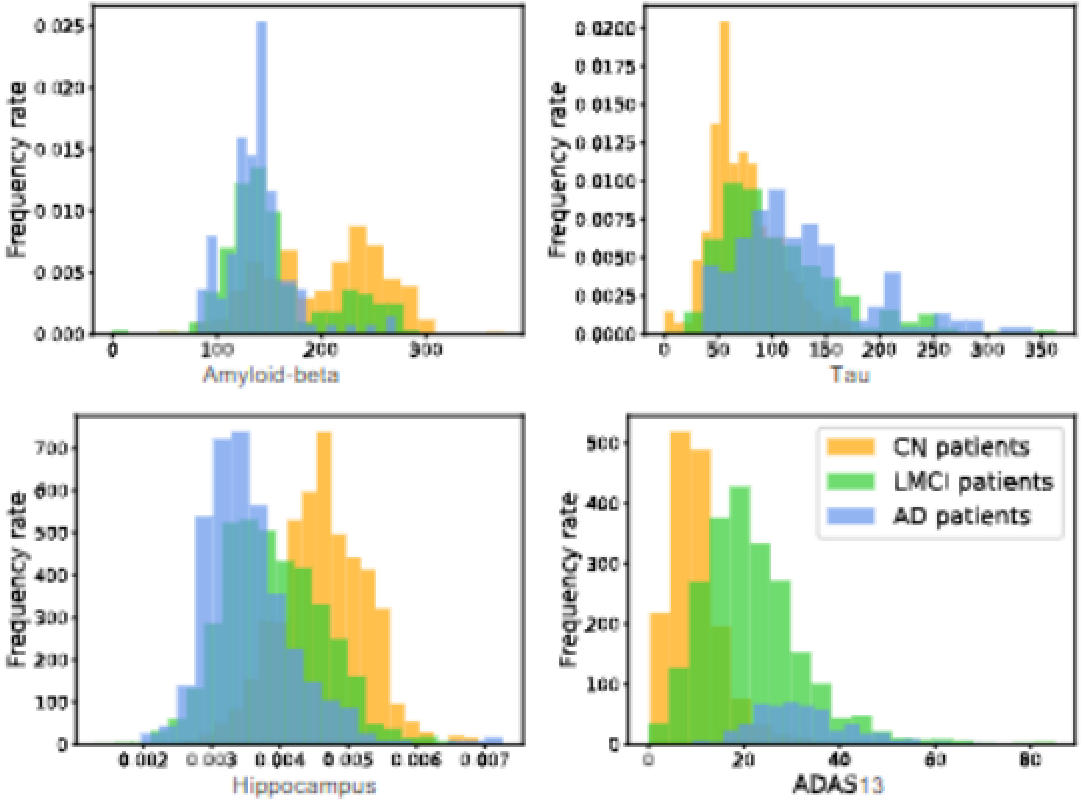


A general causal model is learned from the ADNI dataset

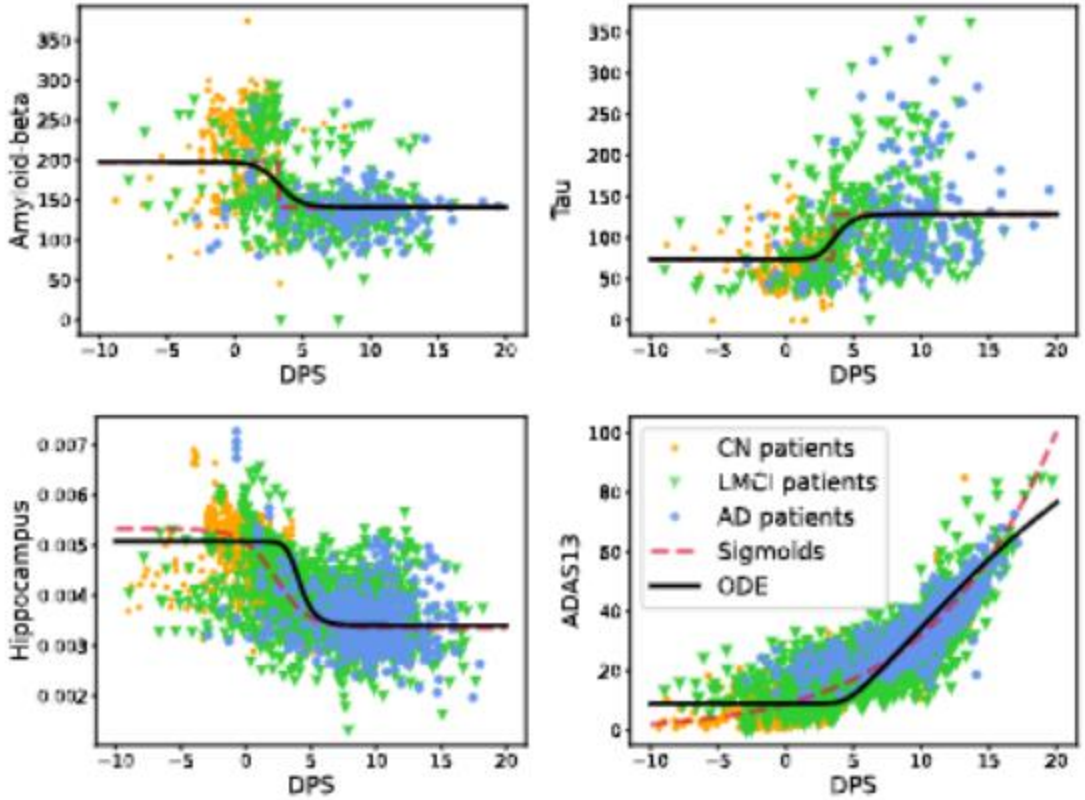
$$
\begin{cases}
\dfrac{dA_\beta}{dt} = w_{10} + w_{11}A_\beta + w_{12}A_\beta^2; \\[2ex]
\dfrac{d\tau}{dt} = w_{20} + w_{21}\tau + w_{22}\tau^2 + w_{23}A_\beta + w_{24}A_\beta^2 + w_{25}A_\beta\tau; \\[2ex]
\dfrac{dN}{dt} = w_{30} + w_{31}N + w_{32}N^2 + w_{33}\tau + w_{34}\tau^2 + w_{35}\tau N; \\[2ex]
\dfrac{dC}{dt} = w_{40} + w_{41}C + w_{42}C^2 + w_{43}N + w_{44}N^2 + w_{45}NC,
\end{cases}
$$

Zheng, H, Petrella, JR, Doraiswamy, PM, Lin, G, Hao, W **(2022)**. Data-driven causal model discovery and personalized prediction in Alzheimer's disease. *NPJ Digit Med*, 5, 1:137.

Results of population model

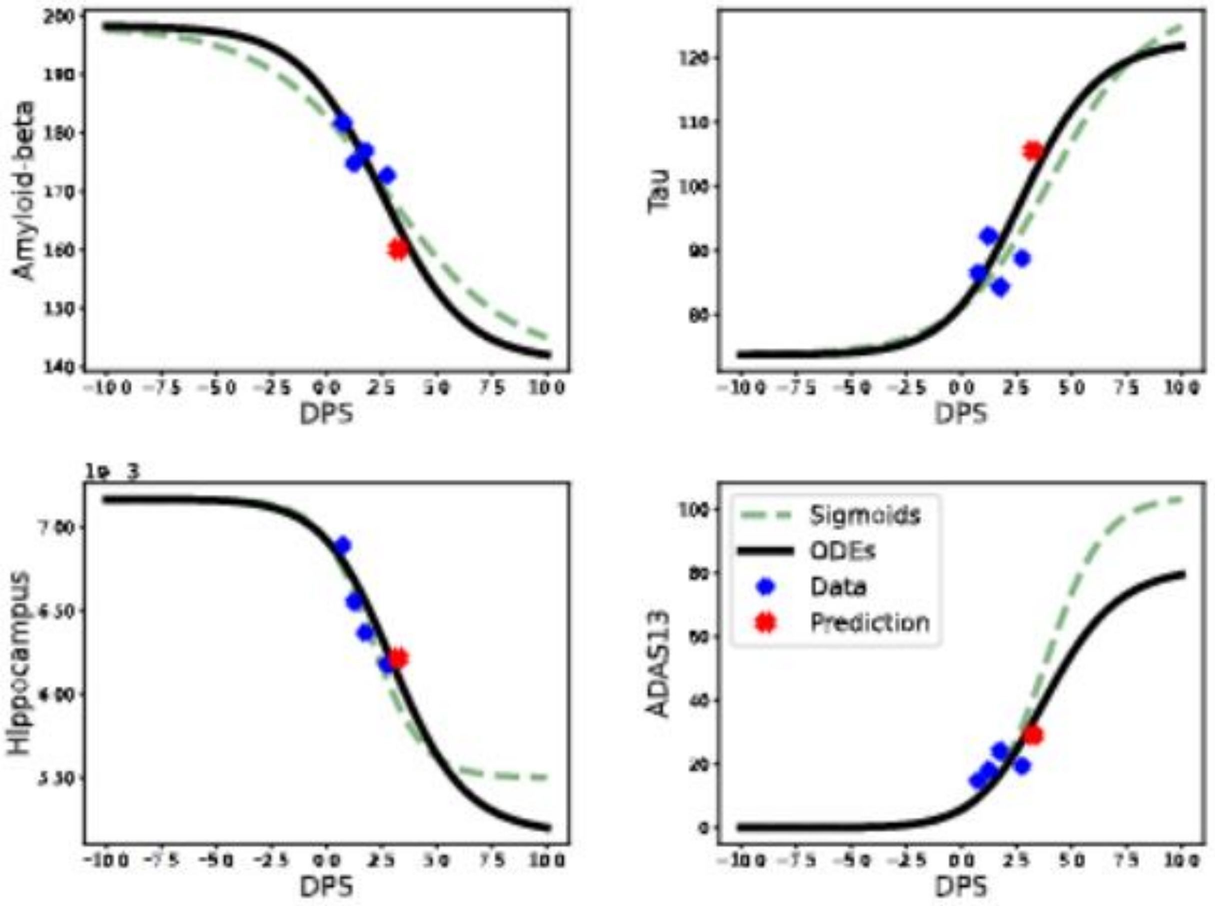


Histogram of biomarkers in ADNI



The population model on three groups

Results of personalized models



The personalized causal model for one AD patient with patient ID =126.

Results of personalized models

Groups	DPS diff	Abeta	Tau	Hippo	ADAS13
CN	0.78 ± 0.64	$97.2 \pm 3.5\%$	$93.3 \pm 3.9\%$	$97.8 \pm 1.9\%$	$88.5 \pm 5.8\%$
MCI & AD	0.58 ± 0.17	$97.4 \pm 1.5\%$	$93.7 \pm 3.8\%$	$96.0 \pm 2.9\%$	$88.6 \pm 5.7\%$
CN	0.78 ± 0.64	$86.8 \pm 7.6\%$	$81.4 \pm 7.4\%$	$82.6 \pm 8.2\%$	$76.3 \pm 10.1\%$
MCI & AD	0.58 ± 0.17	$85.5 \pm 7.8\%$	$81.4 \pm 8.2\%$	$85.8 \pm 7.5\%$	$77.3 \pm 7.9\%$

Zheng, H, Petrella, JR, Doraiswamy, PM, Lin, G, Hao, W **(2022)**. Data-driven causal model discovery and personalized prediction in Alzheimer's disease. *NPJ Digit Med*, 5, 1:137.

Conclusions

- Gauss-Newton method has been developed for the variational formulation of partial differential equations using neural network discretizations.
- The proposed method has been analyzed to demonstrate superlinear convergence.
- A mathematical model of Alzheimer’s disease (AD) is built to describe the progression of AD clinical biomarkers through a public patient dataset.

Acknowledgement

Collaborators

- Dr. Qingguo Hong (Penn State)
- Dr. Xianlin Jin (Peking University)
- Dr. Yiran Wang (Purdue)
- Dr. P. Murali Doraiswamy (Duke)
- Dr. Jeffrey Petrella (Duke)
- Dr. Guang Lin (Purdue)
- Dr. Haoyang Zheng (Purdue)

NSF DMS-2052685

NIH NIGMS R35GM146894