UNIVERSITY OF CAMBRIDGE

Parameter learning enhancement on Biophysics informed Brain Tumour Segmentation

2024 SRIM summer research project

Supervised by Lipei Zhang and Dr. Angelica Aviles-Rivero

Yizhen Chen

Outline

- Background
- Related work
- Motivation
- Method
- Experiment
- Conclusion and contribution
- Future work

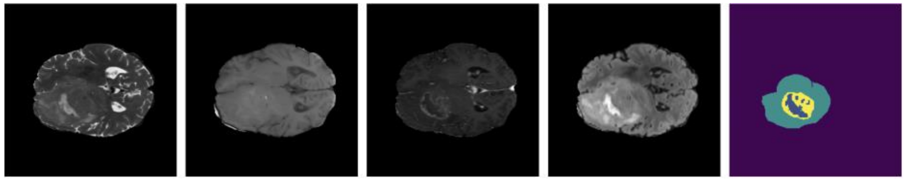
Background

Glioblastoma

1. Approximately 80% of individuals with glioma die within two years
2. Extreme intrinsic heterogeneity of glioma in appearance, shape, and histology
3. Hard to treat and inherently resistant to conventional therapy

BraTS21 Dataset

1. mp-MRI scans of 1251 patients
2. 4 modalities - T1, T1Gd, T2 and T2-FLAIR
3. 4 subregions - normal region , tumour core, and whole tumour, enhancing tumour.

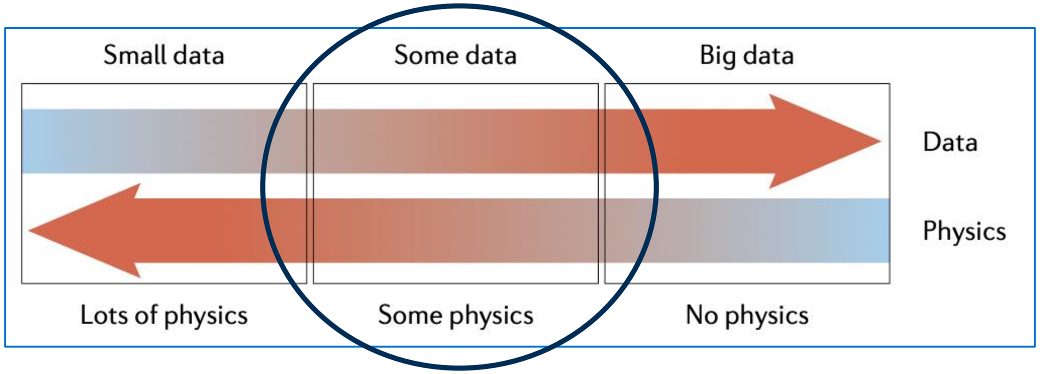


[1] G.E. Karniadakis et al. , Physics-informed machine learning. Nat Rev Phys 3, 2021

Background

Physics Informed Neural Network (PINN)

1. Interaction of physical, chemical and biological processes taking place on spatiotemporal scales that span 17 orders of magnitude.
2. Missing, gappy or noisy boundary conditions through traditional approaches is currently impossible.



[1]

[1] G.E. Karniadakis et al. , Physics-informed machine learning. Nat Rev Phys 3, 2021

Related work

Biophysics Informed Pathological Regularization

Biophysics Informed Pathological Regularization (Zhang, L et al., MICCAI 2024)

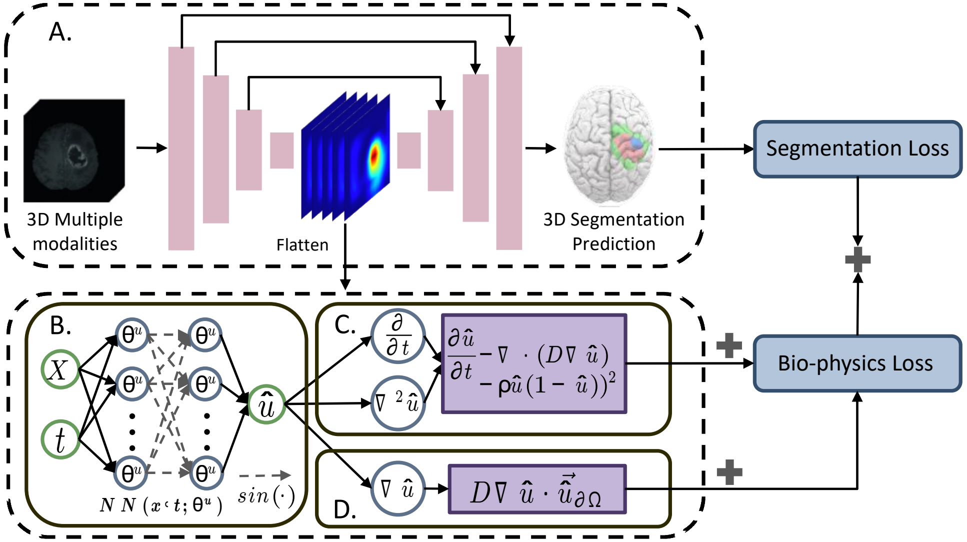
Reaction-diffusion PDE:

[1] $$\frac{\partial u}{\partial t} = \underbrace{\nabla \cdot (d \nabla u)}_{\text{Diffusion}} + \underbrace{f(u)}_{\text{Reaction}}$$

$$f(u) = \rho u (1 - u), \quad d \nabla u \cdot \vec{u}_{\partial \Omega} = 0$$

Loss function:

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{Dice}} + \underbrace{\lambda_1 \mathcal{L}_{\text{PDE}} + \lambda_2 \mathcal{L}_{\text{BC}}}_{\text{Biophysics Informed Regularisation}}$$



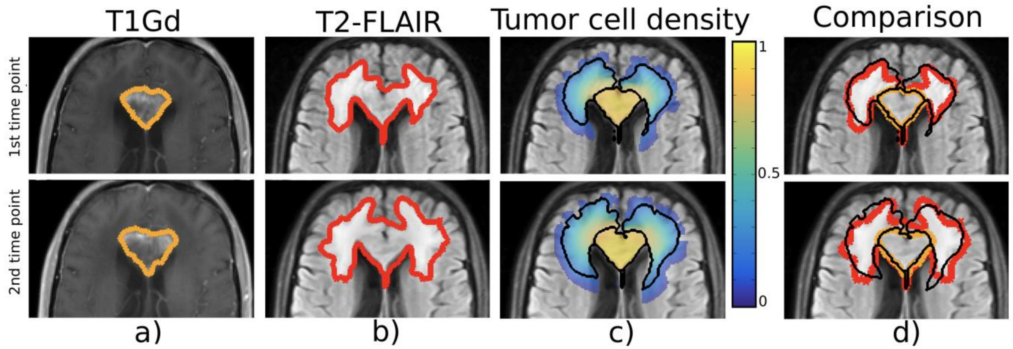
Tumor density estimator:

Multiple fully connected layers with periodic activation functions to disentangle high-level feature maps from encoder concatenated with an assumed temporal dimension t to potential tumour cell density u.

[1] Lê, M et al., Bayesian Personalization of Brain Tumor Growth Model, MICCAI 2015

Motivations

Parameter learning neural networks for d and ρ

[1] 

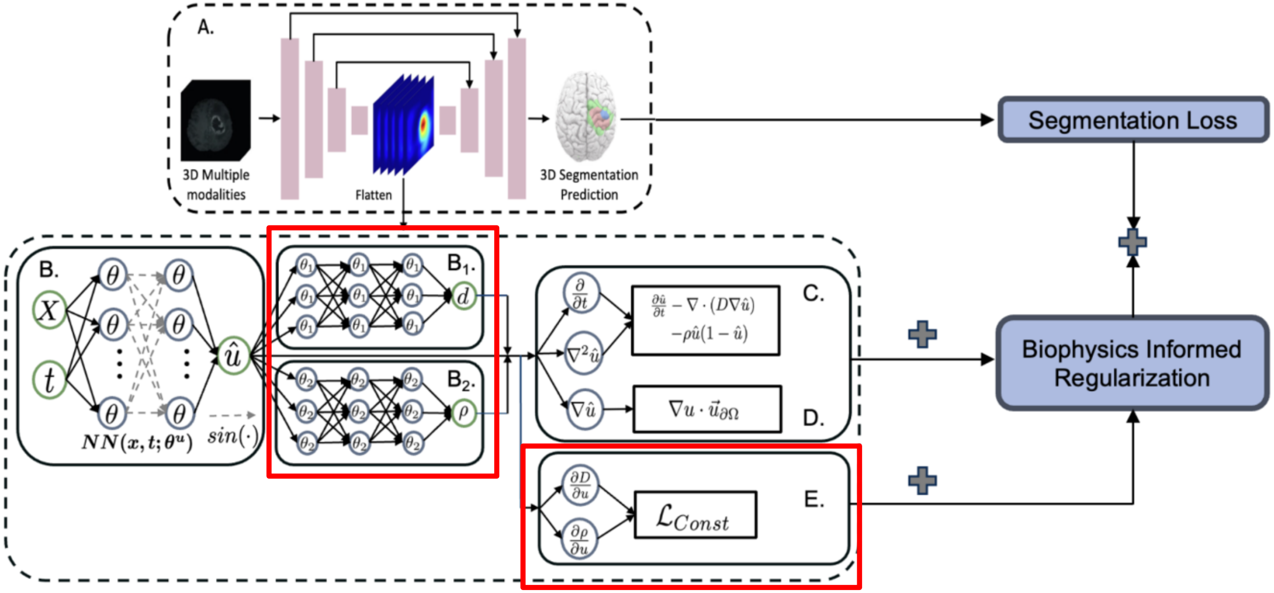
- Diffusion coefficient and the proliferation rate can be related to clinically relevant information, but difficult to estimate.
- d and ρ$are randomly initialized in the previous model$
- Learn the parameters within the governing function with two separate networks
- Leverage the property of MLP that can assimilate any continuous functions

UNIVERSITY OF CAMBRIDGE

[1] Lê, M et al., Bayesian Personalization of Brain Tumor Growth Model, MICCAI 2015

Method

Neural network Pipeline



Part A. – Segmentation model

Part B. – Tumor density estimator

Part B_1. – Diffusion coefficient estimator

Part B_2. – Proliferative rate estimator

Part C. – PDE Loss

Part D. – Boundary Loss

Part E. – Constraint Loss

Prediction: $P = \{p_0, p_1, p_2, p_3\}$ ***Label:*** $Y = \{y_0, y_1, y_2, y_3\}$

Dimension of feature maps: $(B, C, H, W, D) \longrightarrow (B, C, H \times W \times D) \longrightarrow (B, C, H \times W \times D \times 2)$

Embedding temporal dimension T

Method

Part A: $P = f_{\theta_{seg}}(I)$

Part B: $\hat{u} = \Gamma(y) = W_n\left(\gamma_{n-1} \circ \gamma_{n-2} \circ \cdots \circ \gamma_0\right)(y) + b_n, \quad y_i \mapsto \gamma_i\left(y_i\right) = \sin\left(W_i y_i + b_i\right)$ [1]

Part B_1: $\hat{d} = \Gamma_1(\hat{u}) = W_{n_1}^{(1)}\left(\gamma_{n_1-1}^{(1)} \circ \gamma_{n_1-2}^{(1)} \circ \cdots \circ \gamma_0^{(1)}\right)(\hat{u}) + b_n^{(1)}, \quad \hat{u}_i \mapsto \gamma_i^{(1)}\left(\hat{u}_i\right) = \sigma_i^{(1)}\left(W_i^{(1)} \hat{u}_i + b_i^{(1)}\right)$

Part B_2: $\hat{\rho} = \Gamma_2(\hat{u}) = W_{n_2}^{(2)}\left(\gamma_{n_2-1}^{(2)} \circ \gamma_{n_2-2}^{(2)} \circ \cdots \circ \gamma_0^{(2)}\right)(\hat{u}) + b_n^{(2)}, \quad \hat{u}_i \mapsto \gamma_i^{(2)}\left(\hat{u}_i\right) = \sigma_i^{(2)}\left(W_i^{(2)} \hat{u}_i + b_i^{(2)}\right)$

Here γ_i denotes the ith layer $\mathbb{R}^{M_i} \mapsto \mathbb{R}^{N_i}$ transformation, by the weight matrix $W_i \in \mathbb{R}^{M_i \times N_i}$ and bias vector $b_i \in \mathbb{R}^{M_i}$.

Optimization scheme:

$$\arg \min_{\substack{\theta_{Unet}, \theta \\ \theta_1, \theta_2}} \mathcal{L}(I, Y, \theta) = \arg \min_{\substack{\theta_{Unet}, \theta \\ \theta_1, \theta_2}} \Bigg(\underbrace{\mathcal{L}_{Dice}\left(f_{\theta_{Unet}}(I)\right)}_{\text{Segmentation Loss}} + \underbrace{\mathcal{L}_{PDE}\left(\hat{u}_\theta(x,t), \hat{d}_{\theta_1}(\hat{u}), \hat{\rho}_{\theta_2}(\hat{u})\right)}_{\text{PDE Loss}} +$$
$$\underbrace{\mathcal{L}_{BC}\left(\hat{u}_\theta(x,t)\right)}_{\text{BC Loss}} + \underbrace{\mathcal{L}_{Constr}\left(\hat{d}_{\theta_1}(\hat{u}), \hat{\rho}_{\theta_2}(\hat{u})\right)}_{\text{Constraint Loss}} \Bigg)$$

[1] Sitzmann V. et al: Implicit neural representations with periodic activation functions. Advances in neural information processing systems 33, 7462–7473 (2020)

Method

Loss function

Segmentation Loss: *Dice Loss*

$$\mathcal{L}_{Dice} = \sum_{n \in \{0,1,2,3\}} \left(1 - \frac{2 y_n \hat{p}_n + 1}{y_n + \hat{p}_n + 1}\right)$$

PDE Loss:

$$\mathcal{L}_{\text{PDE}} = \left\| \frac{1}{N_{\text{biophy}}} \sum_{i=1}^{N_{\text{biophy}}} \left(\frac{\partial \hat{u}_i}{\partial t_i} - \hat{d}_i \nabla^2 \hat{u}_i - \rho_i u_i (1 - u_i) \right) \right\|^2$$

Laplacian: *3D Laplacian Kernel*

Partial derivatives: *Automatic [illegible]tiation*

$$K = \left[\begin{array}{ccc|ccc|ccc} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & -6 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{array}\right]$$

Boundary Loss:

$$\mathcal{L}_{\text{BC}} = \left\| d \sum_{x=0}^{H} \sum_{y=0}^{W} \sum_{z=0}^{D} \left[\frac{\mathbb{1}_{\{0,H\}}(x)}{WD} \hat{u}_x^2(y,z) + \frac{\mathbb{1}_{\{0,W\}}(y)}{HD} \hat{u}_y^2(x,z) + \frac{\mathbb{1}_{\{0,D\}}(z)}{HW} \hat{u}_z^2(x,y) \right] \right\|$$

Constraint Loss:

$$\mathcal{L}_{\text{Constr}} = \frac{1}{N_{biophy}} \sum_{x=0}^{H} \sum_{y=0}^{W} \sum_{z=0}^{D} \Big[\hat{d}^2 (1 - \mathbb{1}_{(d_{min}, d_{max})})(\hat{d})(x,y,z) + \hat{\rho}^2 (1 - \mathbb{1}_{(\rho_{min}, \rho_{max})})(\hat{\rho})(x,y,z)$$
$$+ \left(\frac{\partial \hat{d}}{\partial \hat{u}} \mathbb{1}_{(-\infty,0)} (\frac{\partial \hat{d}}{\partial \hat{u}})(x,y,z) \right)^2 + \left(\frac{\partial \hat{\rho}}{\partial \hat{u}} \mathbb{1}_{(0,\infty)} (\frac{\partial \hat{\rho}}{\partial \hat{u}})(x,y,z) \right)^2 \Big]$$

Experiment

Hyperparameter and network fine-tuning

- **Segmentation model** = U-net
- **Number of hidden layers in three MLPs** = 3
- **Size of hidden layers in three MLPs** =256
- **Activation function in three MLPs:** sine function in hidden layer and softplus/linear function in the output layer
- **Learning rate** = 0.0002-0.0003 with a cosine decay schedule
- **Mini-batch size** = 8
- **Epochs** = 150 - 200

Experiment

Data, device and biophysics parameters

- Randomly split the **1251 datasets into 7:1:2**
- MONAI v1.3.2 and Pytorch 2.0.1+cu118
- Cambridge CSD3 HPC with NVIDIA A100-SXM-80GB GPUs
- $[d_{min}, d_{max}] = [0.02, 0.15]\ mm^2/day$
- $[\rho_{min}, \rho_{max}] = [0.002, 0.2]\ /day$

Methods	Dice ↑ TC/WT/ET	Hausdorff Distance (mm) ↓ TC/WT/ET
Unet(biophy)	90.58/90.89/87.88	9.18/8.33/10.89
Unet(Parameters learning)	91.36/91.83/88.01	7.07/7.48/10.43

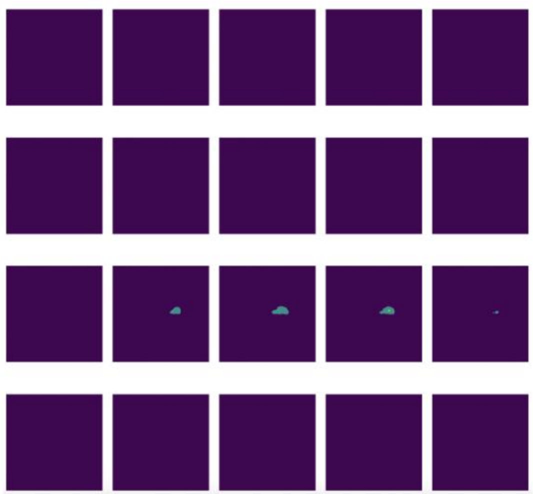
Conclusion and contribution

- **Constructing** two MLPs to learn the diffusion coefficient and proliferative rate
- **Devise** the Constraint loss function to regularize the range of $d$and ρ.
- **Fine-tuned** by experimenting with various hyperparameters to optimize performance.
- **Tested** The final model on the BraTS21 dataset, demonstrating improvements over the previous version.

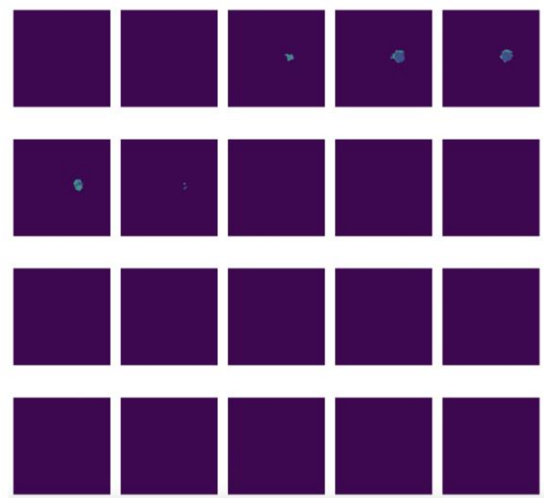
Future work

Bad performance with missing or negligible subregion

```
Dice [[0.0, 0.866, 0.0]] ([[0.93, 0.928, 0.896]])   Hausdorff_Distance95 [[373.129, 4.123, 373.129]] ([[7.295, 8.031, 7.189]])
Dice [[0.3, 0.93, 0.291]] ([[0.933, 0.925, 0.903]]) Hausdorff_Distance95 [[16.248, 4.359, 13.077]] ([[5.511, 7.427, 4.601]])
Dice [[0.0, 0.929, 1.0]] ([[0.92, 0.92, 0.887]])    Hausdorff_Distance95 [[373.129, 3.0, 0.0]] ([[6.918, 7.45, 7.791]])
Dice [[0.0, 0.49, 0.0]] ([[0.916, 0.918, 0.883]])   Hausdorff_Distance95 [[31.464, 20.421, 373.129]] ([[7.029, 7.509, 9.444]])
Dice [[0.053, 0.782, 0.0]] ([[0.927, 0.92, 0.894]]) Hausdorff_Distance95 [[28.46, 20.717, 373.129]] ([[5.273, 7.572, 7.85]])
Dice [[0.8, 0.961, 0.0]] ([[0.932, 0.921, 0.899]])  Hausdorff_Distance95 [[20.876, 2.236, 373.129]] ([[5.123, 7.502, 6.116]])
```



i



ii

UNIVERSITY OF CAMBRIDGE

Thanks for your listening!