Deep Learning for Hæmodynamics

Emmanuel Abbé, Federico Betti, Annalisa Buffa, Simone Deparis, Ortal Yona Senouf, *Riccardo Tenderini*, Dorina Thanou

EPFL — October 3, 2022

Goals and motivations

Goals

1. Build an *in silico* **dataset of steady–state solutions** in presence of sub–critical stenotic formations.
2. **Construct an artificial MI risk function** using physics–based measurements (WSS, diameter stenosis, etc.) available from the simulations.
3. **Train machine learning models to predict MI risk** from angiography–like images.
4. **Investigate if/how the inductive bias of physics–based measurements can help** in predicting the MI risk (**transfer learning**, **multitask learning**) with a particular emphasis on the Q–FFR.

Geometry deformation

- The original geometry is a **patient–specific femoropopliteal bypass**, segmented from CT–scans [6]. We reverted the flow and considered it as a **bifurcation**.
- The **presence of a stenosis** is obtained by deforming the vessel boundary. We solve the **linear elasticity problem**:

$$\begin{cases} -\mathrm{div}\,\boldsymbol{\sigma}(\boldsymbol{u}) = 0 & \text{in } \Omega, \\ \boldsymbol{u} = \boldsymbol{0} & \text{on } \partial\Omega \backslash \Gamma_{\text{wall}}, \\ \boldsymbol{u} = \boldsymbol{\phi} & \text{on } \Gamma_{\text{wall}}, \end{cases}$$

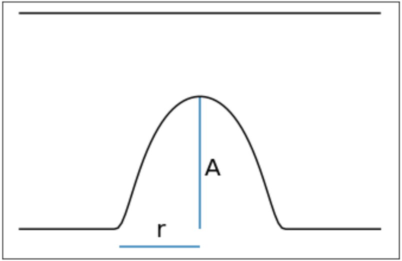
where $\boldsymbol{\sigma}(\boldsymbol{u}) = 2\mu \left(\frac{\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T}{2} \right) + \lambda \mathrm{Tr} \left(\frac{\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T}{2} \right) \boldsymbol{I}$. Let $\boldsymbol{c} \in \Gamma_{\text{wall}}$ (center) and $A, r \in \mathbb{R}^+$ (depth and length); then

$$\boldsymbol{\phi}\left(\boldsymbol{x}; A, r, \boldsymbol{c}\right) = -A\,\hat{\phi}\left(\frac{||\boldsymbol{x} - \boldsymbol{c}||}{r} \right) \boldsymbol{n}(\boldsymbol{c}),$$

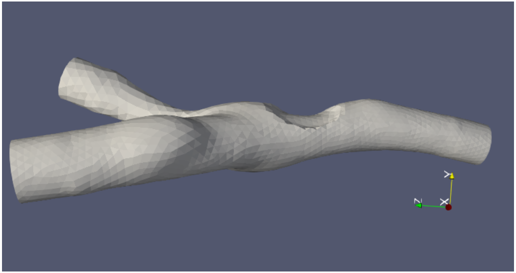
where $\hat{\phi}$ is the standard 1D mollifier.

Geometry deformation — Visualization

Simplified 2D–model of the stenosis



Application to the bifurcation 3D–model



Numerical simulation of hæmodynamics

- On the deformed geometry, we solve the **steady incompressible Navier–Stokes equations** to simulate blood flow.

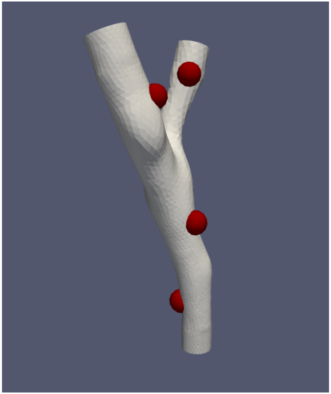
$$
\begin{cases}
\rho\left(\boldsymbol{u}\cdot\nabla\right)\boldsymbol{u} - \operatorname{div}\boldsymbol{\sigma}(\boldsymbol{u},p) = 0 & \text{in } \Omega \\
\operatorname{div}\boldsymbol{u} = 0 & \text{in } \Omega \\
\boldsymbol{\sigma}(\boldsymbol{u},p)\cdot\boldsymbol{n} = \boldsymbol{0} & \text{on } \Gamma_{\text{out}}^{[1]} \cup \Gamma_{\text{out}}^{[2]} \\
\boldsymbol{u} = \boldsymbol{0} & \text{on } \Gamma_{\text{wall}} \\
\int_{\Gamma_{in}} \boldsymbol{u}\cdot\boldsymbol{n} = Q & \text{on } \Gamma_{\text{in}}
\end{cases}
$$

where $\boldsymbol{\sigma}(\boldsymbol{u},p) = -p\boldsymbol{I} + 2\mu\left(\frac{\nabla\boldsymbol{u}+(\nabla\boldsymbol{u})^T}{2}\right)$. Namely, we impose homogeneous Neumann BCs at the two outlets, an **inflow rate** Q at the inlet and homogeneous Dirichlet BCs on the wall.

- Concerning numerical discretization, we employ the **finite elements method**, with P1–P1 elements and SUPG stabilization term [1].

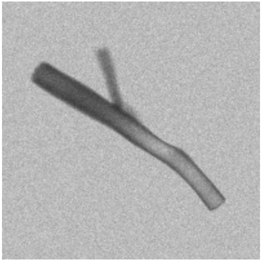
Postprocessing & Dataset construction

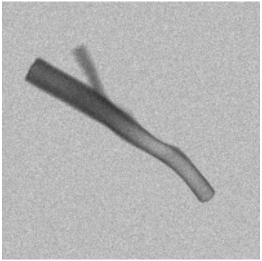
- The generated dataset consists of **8'100 numerical solutions**, obtained by solving the NS equations for different values of the stenosis parameters A and r and of the inflow rate Q. Furthermore, 4 distinct locations for the stenosis center $\boldsymbol{c}$ were considered.
- For each simulation, we store:
  - The **velocity, pressure and WSS fields**.
  - The characteristic **parameter values** (Q, A, r, $\boldsymbol{c}$).
  - The **Q–FFR** in the outlet branches.



Postprocessing & Dataset construction

- For each simulation, we take **2 BW snapshots** (200x200 pixels) at $\approx 30^\circ$ of distance and we add **Poisson random noise**. The snapshots show the **velocity heatmap** and they are as reminiscent as possible of **angiography images**.
- We further enrich the dataset by considering **5 different camera angles**. In total, the dataset is made of **40'500 images**. *Caveat: there may be views from which the stenosis is completely hidden.*



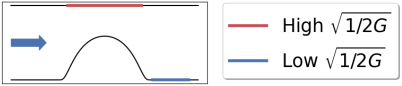


MI risk: function design

The main goal is to **predict the MI risk** associated to the presence of the stenosis. We designed it as follows:

$$MI := \tanh\left(\sqrt{\frac{R^2}{2G}}\exp(A)\right) \qquad \in (0,1)$$

- A is the **diameter stenosis**.
- R is a **risk factor**, exponentially decreasing from the inlet to the outlets, hence **forcing the model to learn the stenosis location**.
- $\sqrt{\frac{1}{2G}}$, with $G = \left(1 + \left|\frac{WSS}{WSS_0}\right|^2\right)^{-1}$, is a common factor to many plaque growth models [9]. $|WSS|$ ($|WSS_0|$) is an **average of the WSS norm at plaque throat** with (without) stenosis.



Auxiliary tasks

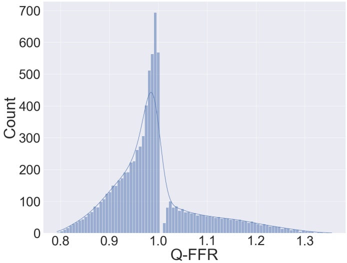
We consider the Q–FFR, the diameter stenosis and the stenosis position as auxiliary tasks.

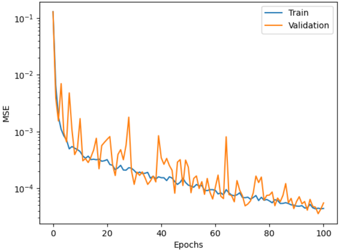
- For the Q–FFR, because a large part of the flow goes into the bigger branch even in the absence of stenosis, the main focus is on the Q-FFR measured at the outlet of the small branch.
- For the stenosis position we perform classification on the 4 possible locations ('Inlet', 'Bifurcation', 'Big branch', 'Small branch').

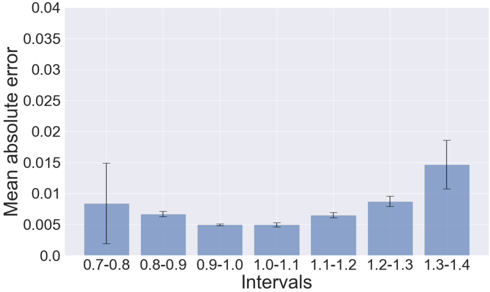
We employ **ResNet18** [3] as architecture. The next slides show for the three auxiliary tasks:

- Labels distribution
- Training trend
- Error distribution: for every interval, the mean error of prediction is shown with the width $\frac{\sigma_N}{\sqrt{\alpha N}}$ of a 95% confidence interval (i.e. $\alpha = 0.05$) as error estimate.

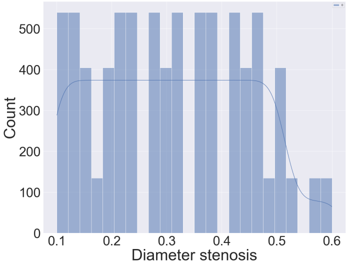
Single tasks results - Q–FFR

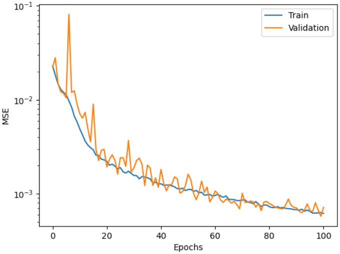


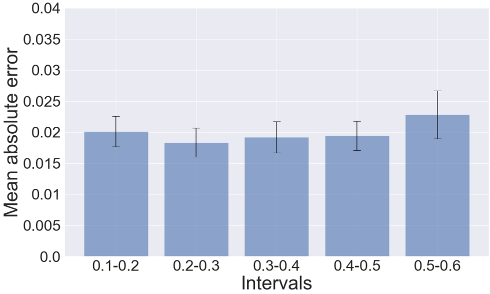




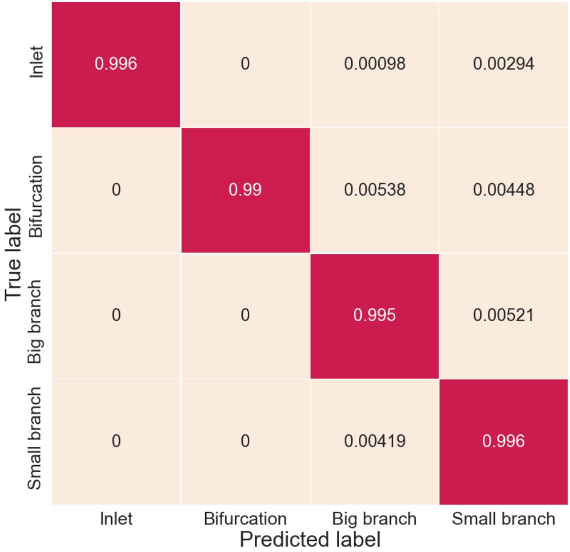
Single tasks results - Diameter stenosis





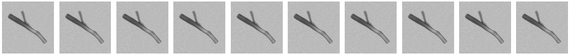


Single tasks results - Stenosis position

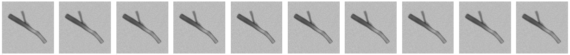


Scenarios for MI risk prediction

To perform MI risk prediction, we compare **three different scenarios** and **three different approaches**, based on the type and amount of data at disposal:

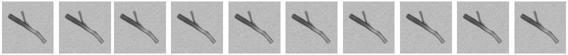
- **Scenario 1:** 

  100% all labels

- **Scenario 2:** 

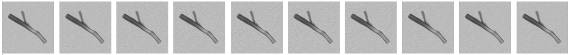
  20% MI labels

  100% Q–FFR labels

- **Scenario 3:** 

  20% all labels

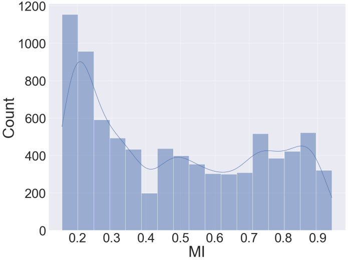
Scenario 1 – Single Task Learning

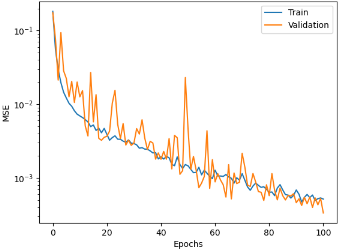
Scenario: 

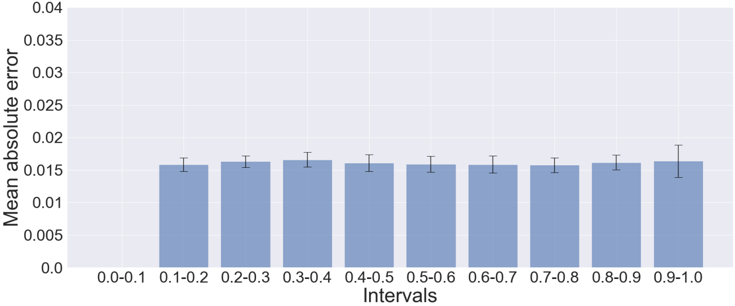
100% all labels

- By single task learning (from random initialization) we train a baseline model on MI predictions using 100 % of the data.
- In the next scenarios, coherently with clinical practice, we have fewer MI labels available. The goal is to obtain a similar performance to this scenario.

Scenario 1 – Single Task Learning

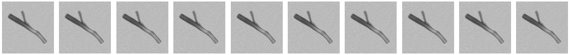






Scenario 2 – Transfer Learning

Scenario:

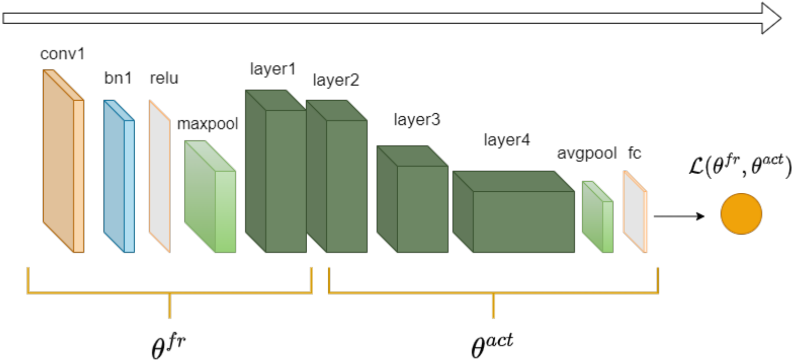


20% MI labels

100% Q–FFR labels

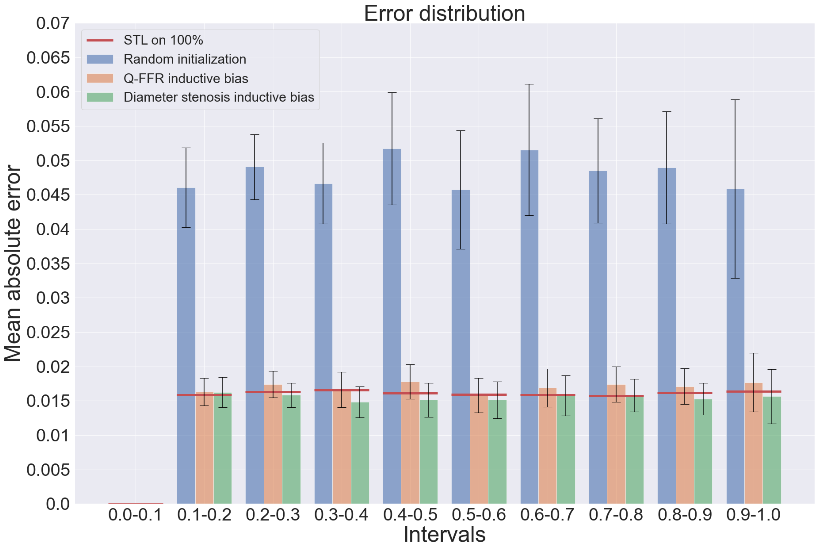
- The goal is to compare the performances of two models trained on MI predictions with 20 % of the data:
  - Starting from **random initialization**.
  - Using as inductive bias a **model pre–trained with Q–FFR predictions** on the whole dataset [7].
- In transfer learning the main hyperparameters are:
  - A suitable partition of the network parameters θ into a set of active parameters θ^{act} and a set of freezed parameters θ^{fr} with corresponding learning rates for the update α^{act} and α^{fr}.
  - The freezed parameters are either kept constant to their initialization value ($\alpha^{fr} = 0$) or they are changed on a much smaller time scale than the active ones ($\alpha^{fr} << \alpha^{act}$).

Scenario 2 – Transfer Learning



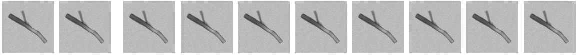
$$\theta^{fr}_{t+1} = \theta^{fr}_t - \alpha^{fr}_t \nabla_{\theta^{fr}_t} \mathcal{L}(\theta^{fr}_t, \theta^{act}_t) \qquad \theta^{act}_{t+1} = \theta^{act}_t - \alpha^{act}_t \nabla_{\theta^{act}_t} \mathcal{L}(\theta^{fr}_t, \theta^{act}_t)$$

Scenario 2 – Transfer Learning



Scenario 3 – Multitask Learning

Scenario:



20% with all labels

- The goal is to compare the performances of models trained on MI predictions with 20 % of the data:
  - Using a standard **single task learning** algorithm.
  - Using a **multitask learning** algorithm to leverage **domain-specific feature sharing**.
- In multitask learning the hyperparameters are:
  - A suitable partition of the network parameters θ into a set of shared parameters θ^{sh} and a set of task-specific parameters θ^{k}.
  - The objective to be optimized.
  - If necessary, the weighting of the single task losses (usually adaptive during training).

Scenario 3 – Multitask Learning

- In standard Multitask Learning (MTL), the objective over the network parameters θ is given by

$$\min_{\substack{\theta^{sh} \\ \theta^1,\cdots,\theta^K}} \sum_{k=1}^{K} w^k \mathcal{L}^k(\theta^{sh}, \theta^k) \quad \text{with} \quad w^k \geq 0, \sum_{k=1}^{K} w^k = 1,$$

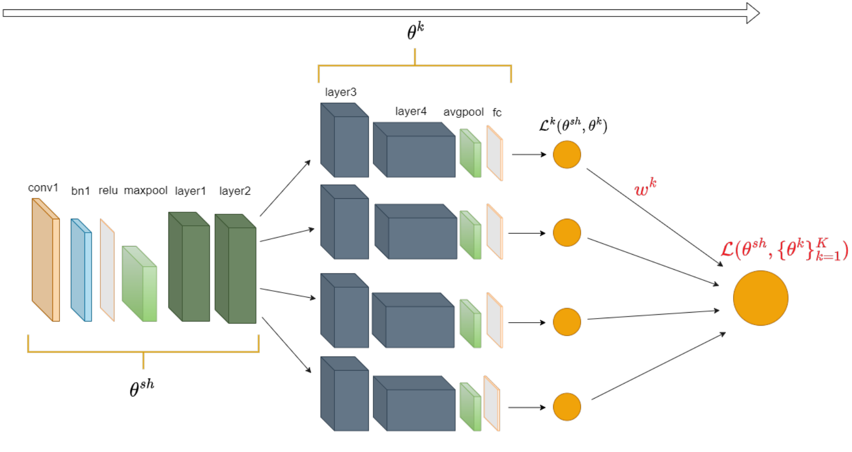
  where $\mathcal{L}_k$ is the individual loss on the k–th task.

- In a multi-objective optimization fashion, the objective is

$$\min_{\substack{\theta^{sh} \\ \theta^1,\cdots,\theta^K}} \left(\mathcal{L}^1(\theta^{sh}, \theta^1), \cdots, \mathcal{L}^K(\theta^{sh}, \theta^K)\right).$$

  From the KKT conditions, the resulting MTL algorithm (**MGDA**) performs standard gradient descent on the task-specific parameters (without any weighting due to task inter-dependencies) and chooses suitably the w^k only for the update of the shared parameters [8].

Scenario 3 – Multitask Learning



$$\theta_{t+1}^{sh} = \theta_t^{sh} - \alpha_t \sum_{k=1}^{K} w^k \nabla_{\theta_t^{sh}} \mathcal{L}^k(\theta_t^{sh}, \theta_t^k) \qquad \theta_{t+1}^k = \theta_t^k - \alpha_t w^k \nabla_{\theta_t^k} \mathcal{L}(\theta_t^{sh}, \theta_t^k)$$

Scenario 3 – Multitask Learning

- Assuming no hierarchical structure exists among the tasks:
  - **Weighted Dynamical Average (WDA): [5]** The weights are updated to ensure that the progress in all the tasks is the same:

$$w^k(t) = \frac{e^{\lambda_k/T}}{\sum_j e^{\lambda_j/T}}, \quad \text{with} \quad \lambda_k = \frac{\mathcal{L}^k(\theta^{sh}_{t-1}, \theta^k_{t-1})}{\mathcal{L}^k(\theta^{sh}_{t-2}, \theta^k_{t-2})} \tag{1}$$

- Assuming the existence of one main task:
  - **Cosine similarity: [2]** It adds $\nabla_{\theta^{sh}_t}\mathcal{L}^k\left(\theta^{sh}_t, \theta^k_t\right)$ to the update of θ^{sh} only if $\cos\left(\nabla_{\theta^{sh}_t}\mathcal{L}^{\text{main}}\left(\theta^{sh}_t, \theta^k_t\right), \nabla_{\theta^{sh}_t}\mathcal{L}^k\left(\theta^{sh}_t, \theta^k_t\right)\right) \geq 0$.
  - **Adaptive Auxiliary Tasks (OL-AUX-N): [4]** The weights of the auxiliary tasks are updated according to

$$\Delta w^k(t) = \frac{\alpha}{N}\sum_{j=0}^{N-1}\left(\nabla_{\theta^{sh}_{t-j}}\mathcal{L}^{\text{main}}\left(\theta^{sh}_{t-j}, \theta^k_{t-j}\right)\right)^T\left(\nabla_{\theta^{sh}_{t-j}}\mathcal{L}^k\left(\theta^{sh}_{t-j}, \theta^k_{t-j}\right)\right)$$

If necessary, one can enforce $\sum_k w^k = 1$ with a softmax.

Scenario 3 – Multitask Learning: gradients correlation

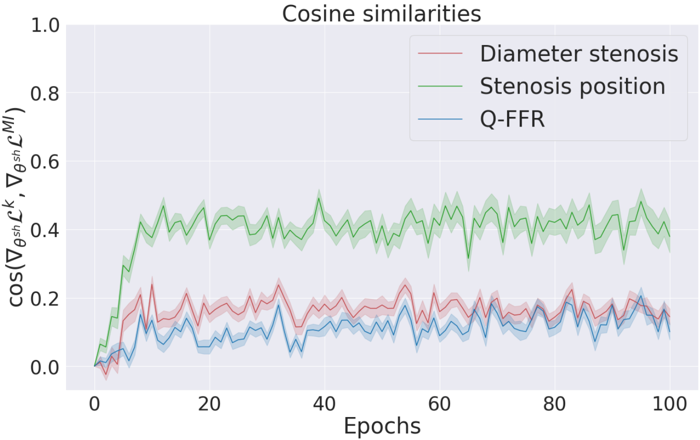
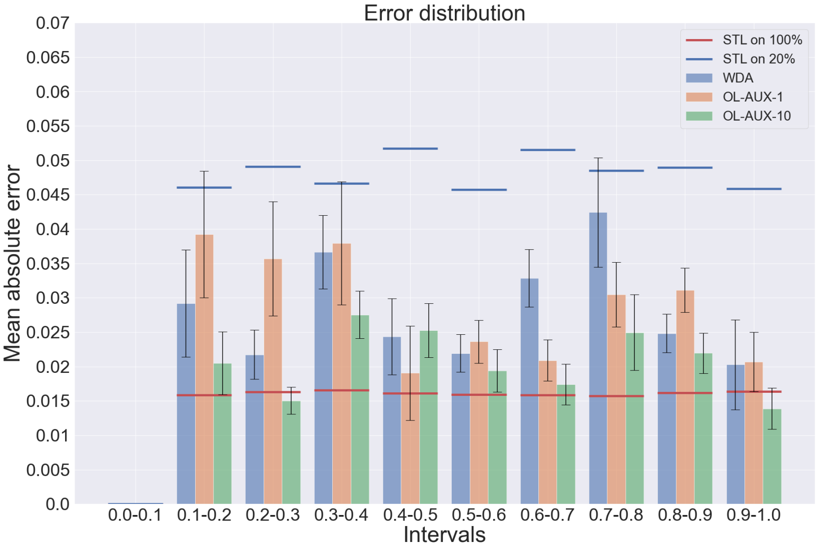
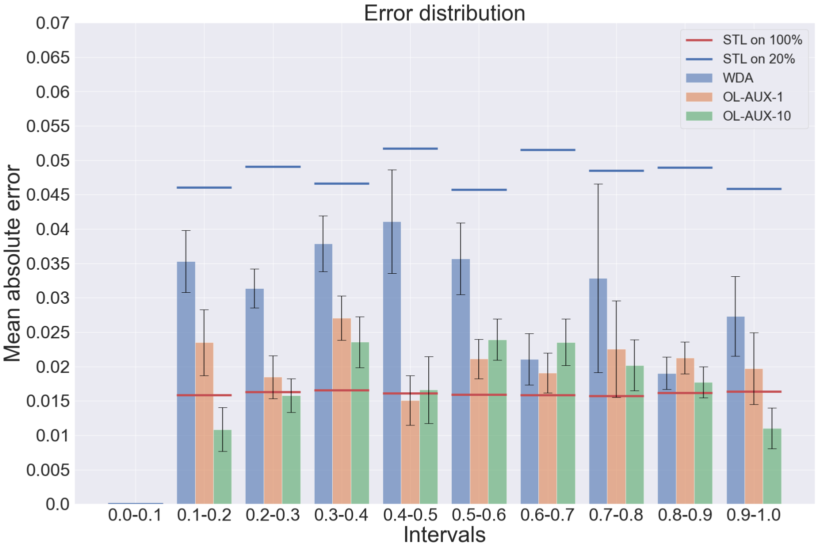


Figure: Gradients correlations with the MI gradients during training (shaded regions show the standard deviation over the training dataset): **cosine similarity** reduces to **uniform weighting** $w^k = \frac{1}{K}$

Scenario 3 – Multitask Learning: all auxiliary tasks



Scenario 3 – Multitask Learning: Q–FFR only auxiliary task



Summary

- **Data**: 40'500 couples of BW noisy images of the velocity field in a bifurcation with stenosis.
- **Goal**: predicting the MI risk in different scenarios.
- **Results**:
  - **Single Task Learning**: if the MI labels are available on the whole dataset, ResNet18 predicts MI risk with absolute errors of roughly 0.015.
  - **Transfer Learning:** using a model trained on the whole dataset with Q–FFR predictions as inductive bias, we retain performances comparable to STL.
  - **Multitask Learning:** with the OL–AUX–10 method, using the domain–specific feature sharing on few data, we attain results comparable to STL. Best results are obtained using only Q–FFR as auxiliary task.

References I

[1] Alexander N Brooks and Thomas JR Hughes. "Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations". In: *Computer methods in applied mechanics and engineering* 32.1-3 (1982), pp. 199–259.

[2] Yunshu Du et al. "Adapting auxiliary losses using gradient similarity". In: *arXiv preprint arXiv:1812.02224* (2018).

[3] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[4] Xingyu Lin et al. "Adaptive auxiliary task weighting for reinforcement learning". In: *Advances in neural information processing systems* (2019).

[5] Shikun Liu, Edward Johns, and Andrew J Davison. "End-to-end multi-task learning with attention". In: *Proceedings of the IEEE/Conference on computer vision and pattern recognition*. 2019.

References II

[6] Emilie Marchandise et al. "Quality open source mesh generation for cardiovascular flow simulations". In: *Modeling of Physiological Flows*. Springer, 2012.

[7] Sinno Jialin Pan and Qiang Yang. "A survey on transfer learning". In: *IEEE Transactions on knowledge and data engineering* (2009).

[8] Ozan Sener and Vladlen Koltun. "Multi-task learning as multi-objective optimization". In: *Advances in neural information processing systems* 31 (2018).

[9] Telma Silva et al. "Modeling of the early stage of atherosclerosis with emphasis on the regulation of the endothelial permeability". In: *Journal of Theoretical Biology* (2020).