Title

Multimodal prediction of biochemical recurrence in prostate cancer patients using mpMRI and clinical data

Introduction

Multimodal deep learning models that integrate heterogeneous patient data have shown significant promise for improving cancer diagnosis and prognosis (Yuan et al., 2025; Zhang et al., 2025). In prostate cancer, combining MRI and clinical or pathology data has been shown to enhance non-invasive diagnosis and tumour grading (Shao et al., 2025).

This technical report This technical report evaluates a multimodal deep learning survival model that integrates mpMRI and clinical data of prostate cancer patients to estimate biochemical recurrence risk (BCR), where BCR is defined as a post-prostatectomy prostate-specific antigen (PSA) level greater than 0.10 µg/L. Model performance is quantified using the concordance index (C-index), with event times corresponding to time to BCR or the time of the most recent PSA measurement following prostatectomy. C-index values above 0.5 demonstrate a prediction better than random chance.

The multimodal architecture consists of a 3D convolutional neural network (MRI CNN Backbone) for processing mpMRI images and a multilayer perceptron (Clinical MLP Backbone) for processing tabular clinical data. Feature embeddings from each modality branch were fused and passed through a risk prediction MLP to output a risk score. Model performance was evaluated across five folds, and the average C-index across fold was

compared across several variants: the full model (Full Multimodal), the model with the clinical branch ablated (Clinical-Ablated), the model with the MRI branch ablated (MRI-Ablated), a standalone unimodal MRI CNN model (MRI CNN), a standalone unimodal Clinical MLP model (Clinical MLP), and a linear Cox model using only clinical data (Clinical Linear). Additionally, alternative preprocessing strategies and training optimizations were evaluated.

Results

Among the unimodal models, the MRI CNN achieved the highest average C-index (0.754 ± 0.147), outperforming both the Clinical MLP (0.716 ± 0.114) and the Clinical Linear model (0.692 ± 0.112). Surprisingly, the initial Full Multimodal model performed worse (0.589 ± 0.233), with substantial variability across folds (Figure 1).

Ablation studies revealed that removing the clinical branch (Clinical-Ablated) improved performance relative to the initial Full Multimodal model (0.692 ± 0.093), whereas removing the MRI branch (MRI-Ablated) substantially reduced performance (0.589 ± 0.156). Notably, the MRI-Ablated model underperformed the unimodal Clinical MLP, suggesting that the clinical branch inside the Multimodal architecture was not learning effectively (Figure 1).

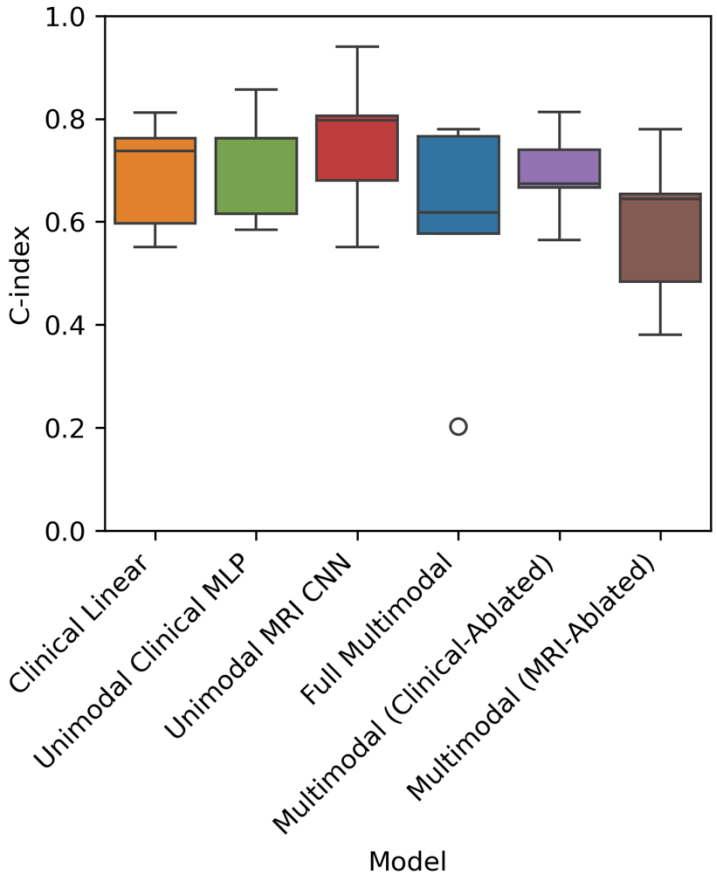


Figure 1. C-index across models. Box plots show the median and IQR, whiskers extend to 1.5 x IQR, with points beyond shown as outliers.

Further interrogation revealed that the discrepancy was due to the Clinical MLP being trained at a lower learning rate (1e-4) inside the Multimodal model, whereas the unimodal Clinical MLP used a higher learning rate (1e-4). When the unimodal Clinical MLP was retrained at 1e-4, its performance dropped sharply (0.496 ± 0.185), approaching random (C-index ~ 0.5) (Figure 2). This confirmed that the clinical branch was under-trained in the Multimodal setting.

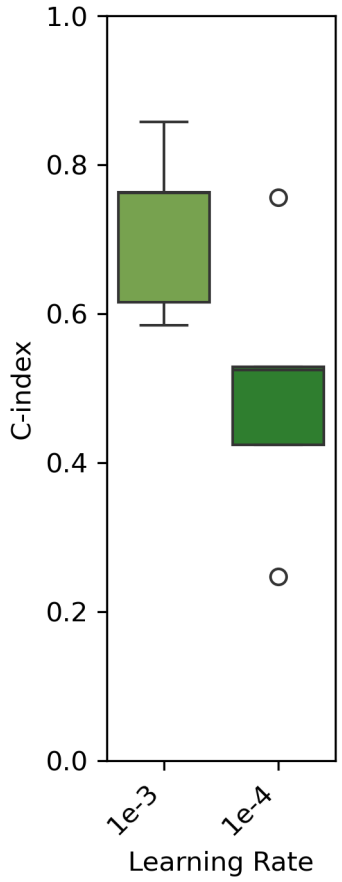


Figure 2. C-index for the Clinical MLP model for different learning rates. Box plots show the median and IQR, whiskers extend to 1.5 x IQR, with points beyond shown as outliers.

To address this, a dual-learning-rate optimizer was introduced, training the MRI CNN branch at a rate of 1e-4 and the Clinical MLP (as well as the risk predictor MLP) at a rate of 1e-3. This modification markedly improved performance, and the updated Full Multimodal model achieved the highest average C-index across all variants (0.786 ± 0.123) (Figure 3).

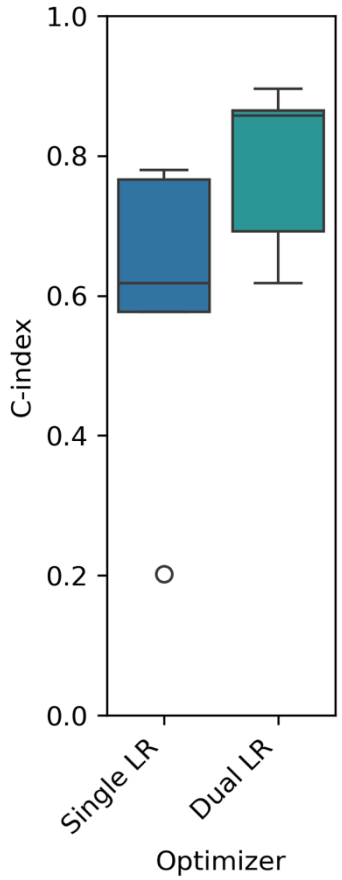


Figure 3. C-index of the Full Multimodal Model under different optimizer learning-rate configurations. Box plots show the median and IQR, whiskers extend to 1.5 x IQR, with points beyond shown as outliers.

Further, different preprocessing strategies were evaluated for the unimodal MRI CNN model. In particular using the prostate mask as an additional input channel (default approach) was compared to cropping all MRI modalities to the prostate region plus an additional margin using the T2W mask. The results showed that pre-cropping yielded lower performance (0.651 ± 0.120) compared with using the mask as an additional channel (Figure 4).

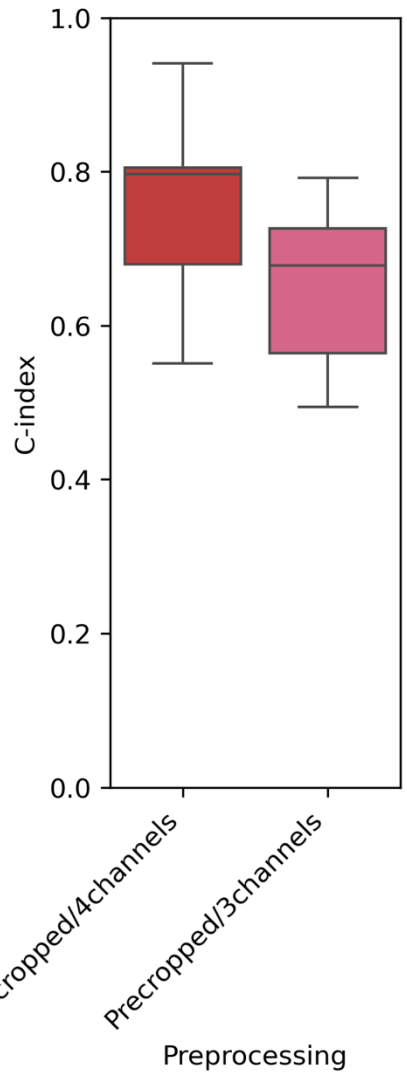


Figure 4. C-index for the unimodal MRI CNN model using different preprocessing strategies. Box plots show the median and IQR, whiskers extend to 1.5 x IQR, with points beyond shown as outliers.

Paired Wilcoxon signed-rank tests across folds showed no statistically significant differences between models after Benjamini–Hochberg correction (α = 0.05).

Discussion

The results demonstrate that multimodal integration of mpMRI and clinical features produced the strongest performance compared to the unimodal models. The initial underperformance of the Full Multimodal model was traced to the Clinical MLP being trained with an excessively low learning rate, which prevented it from contributing meaningfully to the risk prediction. Introducing a dual learning-rate optimizer corrected this imbalance and restored the expected benefit of multimodal fusion.

Preprocessing choices also affected performance of the MRI CNN. Pre-cropping the images to the prostate mask reduced performance despite incorporating an additional margin. This highlights that biochemical recurrence is not necessarily involve local recurrence and features informing recurrence may lie outside the prostate region.

Despite improved average C-index values, high fold-to-fold variability and limited sample size resulted in no statistically significant differences after multiple-testing correction. These findings emphasize that careful optimization of each modality's training dynamics and data preprocessing is essential for maximizing Multimodal performance.

Limitations and Future Directions

The MRI CNN backbone was intentionally kept shallow (three convolutional blocks, maximum 64 channels) to mitigate overfitting in a dataset of only 95 subjects (~76 training samples per fold). This may lack the capacity to capture informative MRI

features. Future work could explore deeper 3D architectures (e.g., 3D ResNet, DenseNet-121).

Additional ablation studies could evaluate the contribution of individual mpMRI modalities by selectively removing each channel (ADC, HBV, T2W) to quantify modality-specific signal. No systematic hyperparameter search was conducted, and more thorough tuning may yield more stable results. Further, preprocessing decisions may also have influenced results. For e.g., intensity clipping the 1st–99th percentiles could improve normalization, and class imbalance between BCR and non-BCR samples may benefit from balancing strategies.

The clinical models were not assessed for multicollinearity or potential feature leakage. High correlations among clinical variables can inflate performance and reduce robustness. Future work should include correlation analysis.

Finally, performance was evaluated solely using the C-index. Although appropriate for survival analysis, additional metrics such as computational cost, memory efficiency, and training stability are relevant for practical deployment and could provide further insight into model behavior.

Methods:

Preprocessing Clinical Data

Clinical tabular data were preprocessed prior to model training as follows. The variable *earlier_therapy* was excluded because only three patients had unique non-null entries (radiation+chemotherapy, radiation+cryotherapy, or “unknown”), while the remaining

patients reported no earlier therapy. The variable *BCR_PSA* was also removed because it directly defines biochemical recurrence (BCR) and therefore introduces strong collinearity with the event label.

For the *capsular_penetration* variable, entries recorded as “x” were converted to “0”, to reflect the absence of documented penetration. Missing values in the variable *tertiary_gleason* score were replaced with “0” to maintain the ordinal scale (1–5). Pathological T-stage covariate, variable *pT-mapping*, was converted to an ordinal scale according to the following scheme:

Pathological staging	Scale
1	1
1a	2
1b	3
1c	4
2	5
2a	6
2b	7
2c	8
3	9
3a	10
3b	11
4/4b*	12

* One patient had a stage listed as “4b,” which is not defined in the European Association of Urology (EAU) guidelines (*EAU Guidelines on Prostate Cancer - CLASSIFICATION AND STAGING SYSTEMS - Uroweb*, n.d.) and was therefore treated as stage “4.”

Prior to model training, clinical data were split into training and validation sets according to the fold assignments in data_split_5fold.csv. For each fold, preprocessing was performed only on the training split, and the resulting transformations were applied to both the training and validation sets to prevent data leakage. Specifically, continuous variables (*age_at_prostatectomy* and *pre_operative_PSA*) were standardized using StandardScaler from scikit-learn (Pedregosa et al., 2011) and categorical variables were one-hot encoded using OneHotEncoder from scikit-learn (Pedregosa et al., 2011). The positive_lymph_nodes variable, included “x” entries, indicating patients whose lymph nodes were not resected. i.e. lymph node metastasis status is unknown. This is different from patients with entry “0”, indicating no cancer is detected in the resected lymph node. As such this variable was treated as a categorical variable.

Preprocessing of mpMRI imaging data

Preprocessing of all mpMRI data was performed using MONA (Cardoso et al., 2022). For each fold, the training and validation splits defined in *data_split_5fold.csv* were transformed independently. The preprocessing pipeline consisted of the following steps.

First, all image modalities (ADC, HBV, T2W) and the corresponding prostate mask were loaded as PyTorch tensors (*LoadImaged*) and converted to a channels-first format

(*EnsureChannelFirstd*). Image orientation was standardized to the RAS (Right–Anterior–Superior) convention (*Orientationd*). The T2W and mask volumes were then resampled to a target voxel spacing of (1.0, 1.0, 3.0) mm using trilinear and nearest-neighbor interpolation, respectively (*Spacingd*). All ADC and HBV volumes were subsequently resampled to match the geometry of the resampled T2W image using trilinear interpolation (*ResampleToMatchd*). All modalities were then resized or padded to a consistent spatial size of 160 × 160 × 48 voxels (*ResizeWithPadOrCropd*).

Data augmentation was applied only to the training set. A spatial affine augmentation (RandAffined) was used with a rotation range of approximately ±5° in all three axes, translation of ±5 voxels in x/y and ±1 voxel in z axes, and isotropic scaling of ±5%. Affine transformations were applied with trilinear interpolation and a probability of 0.5.

Following augmentation, all modalities were normalized independently (*NormalizeIntensityd*), and the four MRI channels (ADC, HBV, T2W, Mask) were concatenated into a single 4-channel tensor (*ConcatItemsd*) for input to the models. An alternative preprocessing procedure was assessed for the MRI CNN unimodal model, which included cropping around the mask with an additional margin of (16, 16, 2) voxels in the (x, y, z) dimensions (*CropForegroundd*), prior to resizing or padding. This procedure also concatenated only the 3 cropped MRI channels (ADC, HBV and T2W).

Clinical Linear Model

The clinical linear model was implemented as an elastic-net regularized Cox proportional hazards model (*CoxnetSurvivalAnalysis*) from scikit-survival (Pölsterl, 2020) with an L1

ratio of 0.0001, resulting in a dominantly L2-regularized (ridge-like) model. This improves model stability and reduces overfitting given the low dimensionality of the clinical data.

Clinical MLP Backbone

The Clinical MLP backbone maps tabular patient features into a low-dimensional embedding. The input is a vector of clinical covariates of length *input_dim* (default: 14). The backbone consists of a single fully connected layer projecting the input to a feature space of dimension *feature_dim* (default: 32), followed by a Rectified Linear Unit (ReLU) activation and dropout (default: 0.1). The output is a dense clinical feature embedding of size *feature_dim* (default: 32).

Clinical MLP Model

The Clinical MLP model extends the backbone with a prediction head for survival risk estimation. The clinical embedding of dimension *feature_dim* (default: 32) is passed through one final fully connected layer with output dimension 1, producing a single log-risk score suitable for Cox proportional hazards modelling.

MRI CNN Backbone

The MRI CNN backbone is a 3D convolutional neural network that extracts feature embeddings from multi-parametric MRI volumes. The input consists of three imaging channels (ADC, HBV, and T2W). The backbone contains three sequential convolutional blocks implemented using MONAI's Convolution module. The first block maps the 3-channel input to a feature embedding of size *feature_dim* (default: 16) using a 3×3×3

kernel with stride 1, instance normalization, and ReLU activation. The second and third blocks progressively downsamples the spatial dimensions using stride 2 while expanding the channel depth to *2* × *feature_dim* (default: 32) and *4* × *feature_dim* (default: 64), respectively.

After the convolutional blocks, a global average pooling layer (*AdaptiveAvgPool3d(1)*) aggregates each feature map into a single scalar value. Dropout (default: 0.3) is applied to the pooled embedding. The backbone outputs a flattened MRI feature vector of size *4* × *feature_dim* (default: 64).

MRI CNN Model

The MRI CNN model extends the backbone with a prediction head for survival risk estimation. The flattened embedding of size *4* × *feature_dim* (default: 64) is passed through one final fully connected layer with output dimension 1, producing a single log-risk score suitable for Cox proportional hazards modelling.

Multimodal Model

The multimodal model integrates features from both the clinical tabular data and the multi-parametric MRI volumes. The Clinical MLP backbone produces an embedding of size *clinical_feature_dim* (default**:** 32), while the MRI CNN backbone produces an embedding of size *4* × *feature_dim* (default: 64). These embeddings are concatenated to form a joint representation (default: 64 + 32 = 96). The concatenated embedding is passed through a prediction head consisting of a single fully connected layer projecting the input to a feature

space of dimension *predictor_feature_dim* (default: 32), followed by ReLU activation and dropout (default: 0.3). The intermediate representation is passed through one final fully connected layer with output dimension 1, producing a single log-risk score suitable for Cox proportional hazards modelling.

Multimodal Ablation

During ablation experiments, either the Clinical MLP branch or the MRI CNN branch was disabled. This was implemented by replacing the corresponding feature embedding with a zero vector prior to concatenation, ensuring that only the remaining modality contributed to the fused representation.

Training and Validation

Training and validation datasets were split using the fold assignments in “data_split_5fold.csv”. For each fold, the model was validated on the samples assigned to the current fold and trained on all remaining samples.

For the Clinical MLP model, the Adam optimizer (Kingma & Ba, 2017) was used with a learning rate of 1e-3 or a learning rate of 1e-4 and weight decay 1e-4. For the MRI CNN model, Adam was used with a learning rate of 1e-4 and the same weight decay. The Multimodal model was trained using both a learning rate of 1e-4 and a dual optimizer setting that trained the CNN branch with 1e-4 and MLP segments (Clinical MLP and risk prediction MLP) with 1e-3. Models were trained with a batch size of 4, and optimizer gradients were zeroed at every iteration.

During training, each model produced a continuous risk score for each sample. The survival loss was computed using the negative partial log-likelihood of the Cox proportional hazards model, implemented using TorchSurv (Monod et al., 2024). The clinical variable BCR was used as the event indicator, and Time_to_follow-up/BCR variable was used as the time-to-event or censoring time. The fold-wise training loss was calculated as the mean loss across all batches.

During validation, models were evaluated in inference mode (no gradient updates). All validation samples were processed to obtain risk scores, which were combined with their corresponding event and time values. Model performance was quantified using the concordance index (C-index), implemented in TorchSurv. Each fold was trained for 10 epochs. Training and validation metrics (loss and C-index) were tracked using Weights & Biases (wandb) (Please see “Training and Validation Curves” section).

Statistical Analysis

To compare average C-index across folds for all pairs of model variants, a non-parametric two-sample, two-sided, paired Wilcoxon Signed Rank test was conducted followed by a Benjamini-Hochberg multiple testing correction.

Hardware

Training and validation was conducted on Google Colab using the A100 GPU runtime

References

Cardoso, M. J., Li, W., Brown, R., Ma, N., Kerfoot, E., Wang, Y., Murrey, B., Myronenko, A., Zhao, C., Yang, D., Nath, V., He, Y., Xu, Z., Hatamizadeh, A., Myronenko, A., Zhu, W., Liu, Y., Zheng, M., Tang, Y., … Feng, A. (2022). *MONAI: An open-source framework for deep learning in healthcare* (No. arXiv:2211.02701). arXiv. https://doi.org/10.48550/arXiv.2211.02701

EAU Guidelines on Prostate Cancer—CLASSIFICATION AND STAGING SYSTEMS - Uroweb. (n.d.). Retrieved November 20, 2025, from https://uroweb.org/guidelines/prostate-cancer/chapter/classification-and-staging-systems#4_1

Kingma, D. P., & Ba, J. (2017). *Adam: A Method for Stochastic Optimization* (No. arXiv:1412.6980). arXiv. https://doi.org/10.48550/arXiv.1412.6980

Monod, M., Krusche, P., Cao, Q., Sahiner, B., Petrick, N., Ohlssen, D., & Coroller, T. (2024). *TorchSurv: A Lightweight Package for Deep Survival Analysis* (No. arXiv:2404.10761). arXiv. https://doi.org/10.48550/arXiv.2404.10761

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.*, *12*(null), 2825–2830.

Pölsterl, S. (2020). scikit-survival: A Library for Time-to-Event Analysis Built on Top of scikit-learn. *Journal of Machine Learning Research*, *21*(212), 1–6.

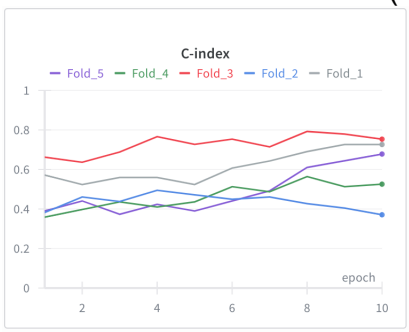
Shao, L., Liang, C., Yan, Y., Zhu, H., Jiang, X., Bao, M., Zang, P., Huang, X., Zhou, H., Nie, P., Wang, L., Li, J., Zhang, S., & Ren, S. (2025). An MRI-pathology

foundation model for noninvasive diagnosis and grading of prostate cancer.

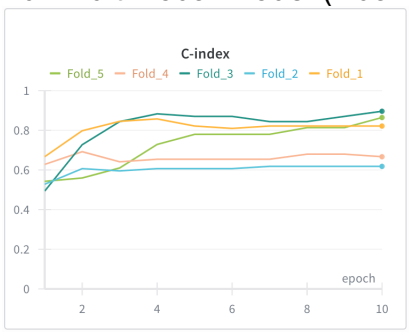
Nature Cancer, *6*(10), 1621–1637. https://doi.org/10.1038/s43018-025-01041-x

Training and Validation Curves

Unimodal MRI CNN Model (Pre-cropped 3-channel Tensor)

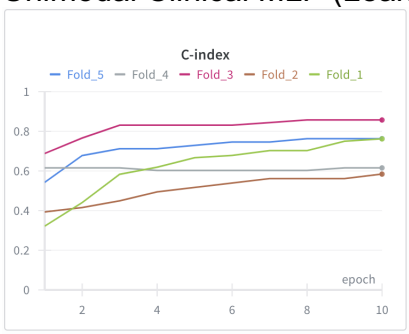


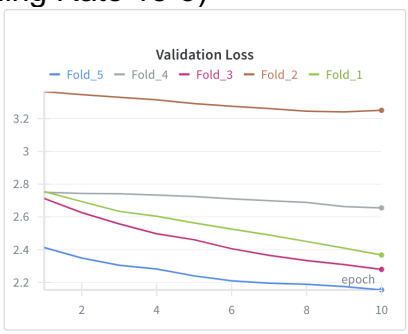
Full Multimodal Model (Dual Learning Rate Optimizer)



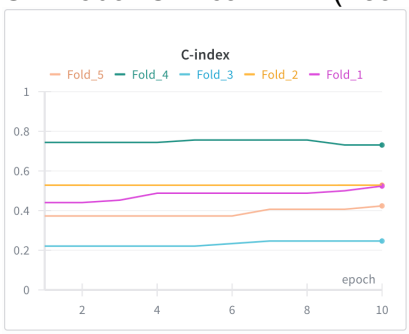
Multimodal Model MRI-Ablated

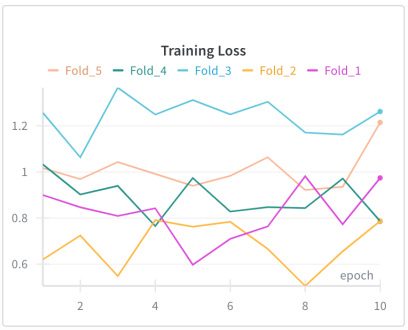
Unimodal Clinical MLP (Learning Rate 1e-3)



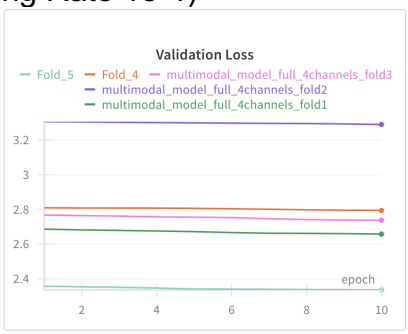


Unimodal Clinical MLP (Learning Rate 1e-4)

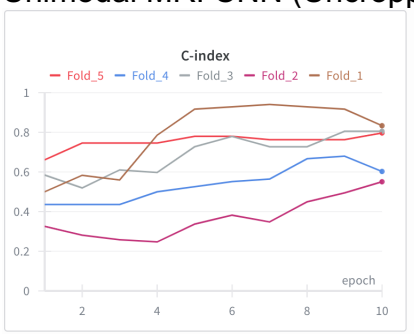


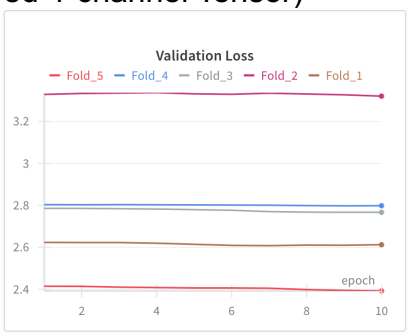


Full Multimodal Model (Learning Rate 1e-4)



Unimodal MRI CNN (Uncropped 4-channel Tensor)





Multimodal Model Clinical-Ablated