DeepJoint: Robust Survival Modelling Under Clinical Presence Shift

Anonymous Author(s)
Affiliation
Address
email

Abstract

Medical data arise from the complex interaction between patients and healthcare systems. This data-generating process often constitutes an informative process. Prediction models often ignore this process, potentially hampering performance and transportability when this interaction evolves. This work explores how current practices may suffer from shifts in this *clinical presence* process and proposes a multi-task recurrent neural network to tackle this issue. The proposed joint modelling performs similarly to state-of-the-art predictive models on a real-world prediction task. More importantly, the approach appears more robust to change in the clinical presence setting. This analysis emphasises the importance of modelling clinical presence to improve performance and transportability.

1 Introduction

Medical observations reflect the complex interaction between patients, clinical staff, and healthcare services. In the context of a hospital stay, each observation from admission to discharge emerges from one of these interactions: observations' times and types reflect the patient's condition and the clinical staff's assessment of what was needed and when. The data-generating process is, therefore, informative — we refer to this process as *clinical presence*.

While the machine learning community has leveraged individual aspects of clinical presence, there are few predictive approaches that explicitly *model* this process. More commonly, modelling pipelines consist of imputation and re-sampling that assume non-informativeness of clinical presence. Although practitioners justify ignoring clinical presence to avoid leveraging a changeable and unpredictable process [1] in the hope of improved robustness, no evidence has demonstrated this property. The challenge of robustness is at the core of clinical presence as changes in this process can occur between countries or over time, due to new medical findings or insurance policy incentives. Ensuring models' transportability under clinical presence shift remains an understudied challenge.

Our work questions the assumption of ignorability for improved generalizability. We explore (i) how state-of-the-art methodologies behave under clinical presence shifts, and (ii) how to leverage clinical presence for improved transportability. Specifically, we propose a multi-task architecture explicitly modelling clinical presence in addition to the survival outcome of interest as shown in Figure 1. This approach links joint modelling, proposed in the statistical literature, to multi-task deep learning by leveraging state-of-the-art neural networks to model three dimensions of clinical presence. These additional components detect potential shifts in the data-generating process and regularise the model to obtain a more robust embedding. Empirical evidence shows strengthened transportability and improved performance.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

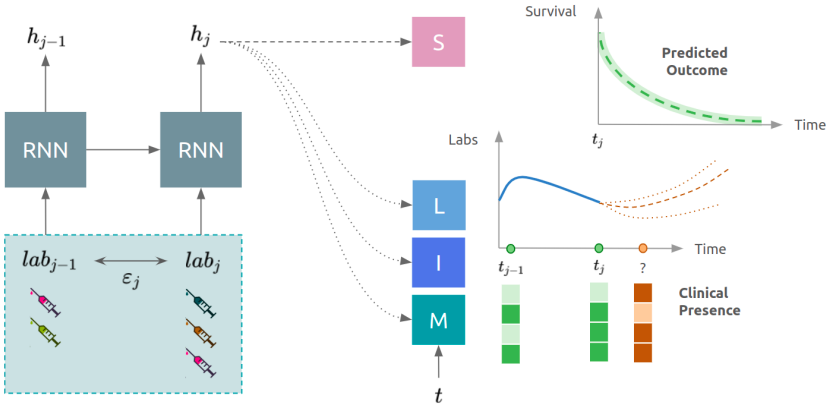

Figure 1: DeepJoint - Joint modelling of clinical presence and survival outcome. *An LSTM network extracts an embedding h which is leveraged to model the clinical presence mechanisms through the networks* ***L*** *– to model the longitudinal evolution,* ***I*** *– to model the inter-observation time,* ***M*** *– to model the missingness patterns, and the survival outcome through the network* ***S****.*

2 Related work

Clinical presence affects Electronic Health Records (EHRs) in multiple ways. First, observation times potentially provide information on the clinical assessment of a patient's condition. For instance, the testing frequency decreases when clinical staff believe that the patient's condition is improving. Second, the order of tests reflects the clinical staff's understanding of the disease, e.g. if a laboratory test is inconclusive, a second would follow. Third, missingness accompanies this previous process, e.g. a missing value reflects its non-importance for diagnosis, or its non-informative value [2]. Leveraging these different dimensions have independently shown predictive edge [3, 4, 5], and reduced biases emerging from this process [6]. For instance, [7] shows improved 3-year-survival predictive performance when inputs contain missingness indicators, and [3] discusses how using this information may improve mortality prediction.

Methodologies to handle clinical presence [8] can be divided into four groups: **(i) Assumed uninformative.** Assume the observed distribution representative of the underlying distribution. **(ii) Leveraged for pre-processing.** Assume a given data-generating process to recover the true distribution from the observed sample. [3] questions these methodologies due to the difficulty of adequately modelling this process. **(iii) Featurised.** Extract measures of clinical presence reflecting medical practitioners' expertise, e.g. missing indicators [3], observation times [7, 9, 10], inter-observation time [11, 12, 13, 14, 15], or frequency [16]. **(iv) Jointly modelled.** Joint models [6, 17, 18], marked point process [19] and Markov models [20] and machine learning extensions [9, 21, 22, 23, 24] model the visit process and the outcome of interest. Through shared effects, state modelling or decaying embedding, the data-generating process informs the modelling of the outcome of interest.

A fundamental feature of clinical presence is its propensity to change as medical practices evolve or due to hospital-level, regional or national practice differences. This is a crucial limitation of the application of machine learning for medical data as models may not generalise and may become outdated as practices and policies evolve [1]. Evidence suggests performance drops across regions and over time [25, 26, 27, 28]. Developing approaches to detect and potentially improve robustness to clinical presence changes is essential to machine learning applicability in clinical practice.

3 Deep Joint

To tackle the problem of both performance under clinical presence and robustness to its shift, we propose a recurrent neural network that jointly models multiple dimensions of the data-generating process: (i) the time to the following observation using a temporal point process neural network [29], (ii) the subsequent tests to be performed and (iii) their expected results using neural networks with Bernoulli and Gaussian likelihoods respectively. Finally, a deep-learning extension of Cox regression, known as DeepSurv [30], models the survival outcome. The proposed model allows scalable joint

modelling by an end-to-end gradient descent maximisation of the full likelihood assuming conditional independence given the extracted embedding. Further description of the model and its training strategy is provided in Appendix A.2. In the remainder of this section, we motivate the use of joint modelling to tackle the challenges linked to clinical presence.

DeepJoint for performance. The proposed method is motivated by the statistical literature that leverages joint models to incorporate informative processes into the outcome model through a shared effect [6, 17, 18]. However, these methods suffer from poor scalability or strong parametric assumptions. Our proposed architecture tackles these issues by leveraging neural networks. This method is also inspired by the multi-task literature in which models aim to predict multiple outcomes to improve performance [31] and generalisability [32, 33]. Theoretical foundations are still lacking [34] and rely on the intuition of regularisation of the shared embedding [35]. Our work builds a connection between these two domains by using multi-task learning to model the informative data-generating process. The aim is to detect if a new point is not following the training clinical presence patterns with the hope that regularisation improves generalisability [32].

DeepJoint for robustness. In healthcare, practitioners sample data to provide care. Under the strong assumption of negligible differences between clinical presence processes under *similar medical practices*, we adapt a result from the adversarial attack literature. This provides an intuition on why multi-task may improve robustness to small changes in clinical presence. Specifically, [36] showed the proportionality of the error to the inverse of the number of tasks modelled. Assuming bounded distance between marked point processes resulting from different clinical presence processes, one can adapt these theoretical results (see Appendix A.1 for proof and further discussion of its limitations).

4 Experiments

This paper studies the problem of in-hospital mortality prediction in the Medical Information Mart for Intensive Care III dataset (MIMIC III) [37] using laboratory tests after a 24-hour observation period in the intensive care unit. The proposed models' and experiments' implementations are available on Github[1].

Data We choose to demonstrate the impact of clinical presence on laboratory tests only as other informative modalities might present different clinical presence patterns, e.g. semi-automatic vital signs collection. After pre-processing [38] and subselection using an ECLAT algorithm [39], the resultant cohorts consists of 30,834 patients with 17 shared laboratory tests observed at least once in the observation period (See Tables 1 and 2 in Appendix).

Baselines We compare the proposed methodology (**DeepJoint**) against different methods for handling clinical presence. All methods rely on the same normalised data imputed using last-observations-carried-forward with patient-mean imputation. All compared approaches extract a representation of the laboratory tests time series, then leveraged by a DeepSurv [30] architecture. First, we compared against two non-sequential approaches: **(i) Last observation (Last)**: Extract the last observation l_i as representation for each patient. **(ii) Summarising (Count)**: Add the count of each test performed in the first 24 hours to the previous representation. This assumes the informativeness of the counting process but ignores its temporal evolution. Then, we used RNN-based approaches to take advantage of the longitudinal evolution of the laboratory tests: **(iii) Ignoring clinical presence (Ignore)**: An LSTM is trained on the imputed data leveraging the inputs' temporal order but ignoring their irregularity and missingness patterns. **(iv) Resampling (Resample)**: Data are re-sampled every hour. **(v) Modelling irregularity (GRU-D)**: Data concatenated with missingness indicators serve as inputs to a GRU-D model [9] which leverages inter-observation times to decay the embedding. **(vi) Featurization (Feature)**: An LSTM leverages missingness indicators and time elapsed since previous observation [3] in addition to the lab results as inputs to model survival. Finally, our proposed method **(vii) DeepJoint** leverages the same input as this last method.

Robustness to shift in clinical presence Motivated by the difference in the counts of tests performed on patients admitted on weekends and weekdays (see Appendix Figure 4), we hypothesise that

[1] Link released upon acceptance of the manuscript.

practice might differ between weekends and weekdays due to physicians and laboratory availability. While providing an opportunity to study survival models' transfer, the weekend effect [40] is marked by different outcome distributions. We adopt the evaluation methodology used in the robustness literature. We stratify patients given days of admission (from Monday 8 am to Saturday 8 am vs the rest). Each group is further divided between training and testing. A first model uses the training set of patients admitted on weekends and tested on the testing set of weekdays-admitted patients. Then, a second model uses the training set of patients admitted on weekdays and tested on the testing set of weekdays-admitted patients. The two resulting models are comparable as evaluated on the same testing set, but trained under two different clinical presence processes (See Figure 3 in Appendix for visualisation of the training strategy).

All methods use the same 90%-10% train-test patients split to train the network. Their training relies on gradient backpropagation with an Adam optimiser [41] over 1000 epochs with early stopping on the survival loss computed on a left aside 10% of the training set, i.e. validation set. The entire network is optimised for 500 epochs. The remaining iterations are for fine-tuning the survival component. We perform hyperparameter tuning on the validation set on the grid presented in Appendix Table 3. We compute survival prediction at the last observation in the 24-hour post-admission period. Models are compared using time-dependent C-index [42] and Brier score [43] evaluated at time horizons of 1, 7 and 14 days after the observation period. 95% confidence intervals were obtained using 100 bootstrapped iterations on the testing set predicted values.

5 Results and Conclusions

Figure 2 presents the performance change when a model is transferred from one clinical presence setting to another. A model robust to change is one with the same performance when transferred from another setting and trained under the same clinical presence. Practitioners should select a model close to the diagonal with the best discriminative performance (upper right corner).

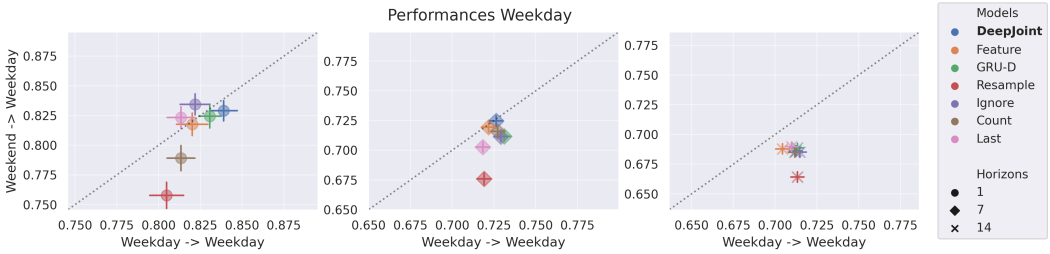

Figure 2: Discriminative performance (time dependent C-index) evaluated on patients admitted on weekdays for a transferred model (*y-axis*) and an oracle model trained on the train weekdays-admitted patients and tested on the testing set of this same group (*x-axis*) (with 95% confidence intervals).

Current practices under-perform under shift. Ignoring the clinical presence process leads to the worst performance and transportability with Resample underperforming in both settings. Last and Count underline that simple featurization may not be an adequate approach, as shown by the detrimental effect of the additional count of events on transportability. Leveraging clinical presence improves performance with explicit modelling outperforming all other approaches.

Clinical presence reflects short-term instability. Note that performances decay with longer horizons with shrinking differences between the different methodologies. This observation shows both the task complexity in intensive care settings and the short-term relevance of clinical presence.

In conclusion, [1, 3] underline how taking advantage of missing data might lead to a mismatch between the training and deployment settings, but ignoring this information might not be possible. Our work proposes a way to leverage this information while informing the user of changes in the observation patterns, resulting in improved performance and robustness. Nonetheless, practitioners should remain wary of informing medical practice with clinical presence as this process may embed not only informative patterns but also potential historical biases [44].

References

[1] M. van Smeden, R. H. Groenwold, and K. G. Moons, "A cautionary note on the use of the missing indicator method for handling missing data in prediction research," *Journal of clinical epidemiology*, vol. 125, pp. 188–190, 2020.

[2] B. J. Wells, K. M. Chagin, A. S. Nowacki, and M. W. Kattan, "Strategies for handling missing data in electronic health record derived data," *Egems*, vol. 1, no. 3, 2013.

[3] Z. C. Lipton, D. Kale, and R. Wetzel, "Directly modeling missing data in sequences with rnns: Improved classification of clinical time series," in *Machine Learning for Healthcare Conference*, pp. 253–270, 2016.

[4] R. H. Groenwold, "Informative missingness in electronic health record systems: the curse of knowing," *Diagnostic and prognostic research*, vol. 4, no. 1, pp. 1–6, 2020.

[5] M. Saar-Tsechansky and F. Provost, "Handling missing values when applying classification models," *Journal of Machine Learning Research*, vol. 8, 2007.

[6] M. Sperrin, E. Petherick, and E. Badrick, "Informative observation in health data: association of past level and trend with time to next measurement," *Stud Health Technol Inform*, vol. 235, pp. 261–265, 2017.

[7] D. Agniel, I. S. Kohane, and G. M. Weber, "Biases in electronic health record data due to processes within the healthcare system: retrospective observational study," *Bmj*, vol. 361, 2018.

[8] R. Sisk, L. Lin, M. Sperrin, J. K. Barrett, B. Tom, K. Diaz-Ordaz, N. Peek, and G. P. Martin, "Informative presence and observation in routine health data: A review of methodology for clinical risk prediction," *Journal of the American Medical Informatics Association*, 2020.

[9] Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu, "Recurrent neural networks for multivariate time series with missing values," *Scientific reports*, vol. 8, no. 1, pp. 1–12, 2018.

[10] R. T. Sousa, L. A. Pereira, and A. S. Soares, "Improving irregularly sampled time series learning with dense descriptors of time," *arXiv preprint arXiv:2003.09291*, 2020.

[11] X. Cai, J. Gao, K. Y. Ngiam, B. C. Ooi, Y. Zhang, and X. Yuan, "Medical concept embedding with time-aware attention," *arXiv preprint arXiv:1806.02873*, 2018.

[12] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun, "Doctor ai: Predicting clinical events via recurrent neural networks," in *Machine learning for healthcare conference*, pp. 301–318, PMLR, 2016.

[13] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart, "Retain: An interpretable predictive model for healthcare using reverse time attention mechanism," in *Advances in Neural Information Processing Systems*, pp. 3504–3512, 2016.

[14] R. Moskovitch, C. Walsh, F. Wang, G. Hripcsak, and N. Tatonetti, "Outcomes prediction via time intervals related patterns," in *2015 IEEE international conference on data mining*, pp. 919–924, IEEE, 2015.

[15] Y. Zhang, "Attain: Attention-based time-aware lstm networks for disease progression modeling.," in *In Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI-2019), pp. 4369-4375, Macao, China.*, 2019.

[16] R. Pivovarov, D. J. Albers, J. L. Sepulveda, and N. Elhadad, "Identifying and mitigating biases in ehr laboratory tests," *Journal of biomedical informatics*, vol. 51, pp. 24–34, 2014.

[17] A. Gasparini, K. R. Abrams, J. K. Barrett, R. W. Major, M. J. Sweeting, N. J. Brunskill, and M. J. Crowther, "Mixed-effects models for health care longitudinal data with an informative visiting process: A monte carlo simulation study," *Statistica Neerlandica*, vol. 74, no. 1, pp. 5–23, 2020.

[18] L. Su, Q. Li, J. K. Barrett, and M. J. Daniels, "A sensitivity analysis approach for informative dropout using shared parameter models," *Biometrics*, vol. 75, no. 3, pp. 917–926, 2019.

[19] K. T. Islam, C. R. Shelton, J. I. Casse, and R. Wetzel, "Marked point process for severity of illness assessment," in *Machine Learning for Healthcare Conference*, pp. 255–270, PMLR, 2017.

[20] A. M. Alaa, S. Hu, and M. Schaar, "Learning from clinical judgments: Semi-markov-modulated marked hawkes processes for risk prognosis," in *International Conference on Machine Learning*, pp. 60–69, PMLR, 2017.

[21] I. M. Baytas, C. Xiao, X. Zhang, F. Wang, A. K. Jain, and J. Zhou, "Patient subtyping via time-aware lstm networks," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 65–74, 2017.

[22] T. Pham, T. Tran, D. Phung, and S. Venkatesh, "Deepcare: A deep dynamic memory model for predictive medicine," in *Pacific-Asia conference on knowledge discovery and data mining*, pp. 30–41, Springer, 2016.

[23] B. Twala, M. Jones, and D. J. Hand, "Good methods for coping with missing data in decision trees," *Pattern Recognition Letters*, vol. 29, no. 7, pp. 950–956, 2008.

[24] P. B. Weerakody, K. W. Wong, G. Wang, and W. Ela, "A review of irregular time series data handling with gated recurrent neural networks," *Neurocomputing*, 2021.

[25] L. L. Guo, S. R. Pfohl, J. Fries, A. E. Johnson, J. Posada, C. Aftandilian, N. Shah, and L. Sung, "Evaluation of domain generalization and adaptation on improving model robustness to temporal dataset shift in clinical medicine," *Scientific reports*, vol. 12, no. 1, pp. 1–10, 2022.

[26] D. Lazer, R. Kennedy, G. King, and A. Vespignani, "The parable of google flu: traps in big data analysis," *science*, vol. 343, no. 6176, pp. 1203–1205, 2014.

[27] B. Nestor, M. B. McDermott, W. Boag, G. Berner, T. Naumann, M. C. Hughes, A. Goldenberg, and M. Ghassemi, "Feature robustness in non-stationary health records: caveats to deployable model performance in common clinical machine learning tasks," in *Machine Learning for Healthcare Conference*, pp. 381–405, PMLR, 2019.

[28] H. Singh, V. Mhasawade, and R. Chunara, "Generalizability challenges of mortality risk prediction models: A retrospective analysis on a multi-center database," *PLOS Digital Health*, vol. 1, no. 4, p. e0000023, 2022.

[29] T. Omi, K. Aihara, *et al.*, "Fully neural network based model for general temporal point processes," in *Advances in Neural Information Processing Systems*, pp. 2122–2132, 2019.

[30] J. L. Katzman, U. Shaham, A. Cloninger, J. Bates, T. Jiang, and Y. Kluger, "Deepsurv: personalized treatment recommender system using a cox proportional hazards deep neural network," *BMC medical research methodology*, vol. 18, no. 1, pp. 1–12, 2018.

[31] A. Maurer, M. Pontil, and B. Romera-Paredes, "The benefit of multitask representation learning," *Journal of Machine Learning Research*, vol. 17, no. 81, pp. 1–32, 2016.

[32] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[33] R. Caruana and J. O'Sullivan, "Multitask pattern recognition for autonomous robots," in *Proceedings. 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems. Innovations in Theory, Practice and Applications (Cat. No. 98CH36190)*, vol. 1, pp. 13–18, IEEE, 1998.

[34] S. Ruder, "An overview of multi-task learning in deep neural networks," *arXiv preprint arXiv:1706.05098*, 2017.

[35] S. Sagawa, P. W. Koh, T. B. Hashimoto, and P. Liang, "Distributionally robust neural networks," in *International Conference on Learning Representations*, 2019.

[36] C. Mao, A. Gupta, V. Nitin, B. Ray, S. Song, J. Yang, and C. Vondrick, "Multitask learning strengthens adversarial robustness," in *European Conference on Computer Vision*, pp. 158–174, Springer, 2020.

[37] A. E. Johnson, T. J. Pollard, L. Shen, H. L. Li-Wei, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. A. Celi, and R. G. Mark, "Mimic-iii, a freely accessible critical care database," *Scientific data*, vol. 3, no. 1, pp. 1–9, 2016.

[38] S. Wang, M. B. McDermott, G. Chauhan, M. Ghassemi, M. C. Hughes, and T. Naumann, "Mimic-extract: A data extraction, preprocessing, and representation pipeline for mimic-iii," in *Proceedings of the ACM Conference on Health, Inference, and Learning*, pp. 222–235, 2020.

[39] M. J. Zaki, S. Parthasarathy, M. Ogihara, and W. Li, "Parallel algorithms for discovery of association rules," *Data mining and knowledge discovery*, vol. 1, no. 4, pp. 343–373, 1997.

[40] L. A. Pauls, R. Johnson-Paben, J. McGready, J. D. Murphy, P. J. Pronovost, and C. L. Wu, "The weekend effect in hospitalized patients: a meta-analysis.," *Journal of hospital medicine*, vol. 12, no. 9, pp. 760–766, 2017.

[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[42] H. Hung and C.-T. Chiang, "Estimation methods for time-dependent auc models with survival data," *Canadian Journal of Statistics*, vol. 38, no. 1, pp. 8–26, 2010.

[43] E. Graf, C. Schmoor, W. Sauerbrei, and M. Schumacher, "Assessment and comparison of prognostic classification schemes for survival data," *Statistics in medicine*, vol. 18, no. 17-18, pp. 2529–2545, 1999.

[44] V. Jeanselme, M. De-Arteaga, Z. Zhang, J. Barrett, and B. Tom, "Imputation strategies under clinical presence: Impact on algorithmic fairness," *arXiv preprint arXiv:2208.06648*, 2022.

[45] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[46] S. Xiao, J. Yan, X. Yang, H. Zha, and S. Chu, "Modeling the intensity function of point process via recurrent neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, 2017.

[47] S. Liu, E. Johns, and A. J. Davison, "End-to-end multi-task learning with attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1871–1880, 2019.

[48] T. Gong, T. Lee, C. Stephenson, V. Renduchintala, S. Padhy, A. Ndirango, G. Keskin, and O. H. Elibol, "A comparison of loss weighting strategies for multi task learning in deep neural networks," *IEEE Access*, vol. 7, pp. 141627–141632, 2019.

A Appendix

A.1 Multi task robustness to clinical presence shift

For completeness, we repeat and slightly adapt the results presented in [36] (see original paper for detailed proof). These results motivates the use of multi-task modelling for robust modelling under shift.

First, we introduce the *clinical presence vulnerability* defined as the expected change in loss under two different data-generating processes. Assuming the time series $x \in f(X)$, the target y, a loss $\mathcal{L}$, the data-generating process observed in the original dataset o_s and the target one o_t such that the difference between the marked point processes resulting from these processes lies in a p-norm bounded ball with radius r, i.e. $||o_s(x) - o_t(x)||_p < r$, then the clinical presence vulnerability over the target dataset is:

$$\Delta\mathbb{E} := \mathbb{E}[|\mathcal{L}(o_s(x), y) - \mathcal{L}(o_t(x), y)|] \leq \mathbb{E}[\max_{||\delta||_p < r} |\mathcal{L}(o_s(x), y) - \mathcal{L}(o_s(x) + \delta, y)|]$$

Assuming *infinitesimal changes* in the data-generating processes, i.e. $r \to 0$, one can develop this expression using Taylor expansion:

$$|\mathcal{L}(o_s(x), y) - \mathcal{L}(o_s(x) + \delta, y)| = |\partial_x \mathcal{L}(o_s(x), y)\delta + O(\delta)|$$

Leveraging the property of the dual norm q, one can show that:

$$\Delta\mathbb{E} \propto \partial_x \mathbb{E}[||\mathcal{L}(o_s(x), y)||_q]$$

Theorem 1 (Theorem 2 from [36]) *If the tasks are correlated with each other such that the covariance between the gradient of task i and task j is $Cov(r_i, r_j)$, and the gradient for each task is i.i.d. with zero mean (because the model has converged), then clinical presence vulnerability of the given model is proportional to:*

$$\sqrt{\frac{1 + \frac{2}{M}\sum_{i=1}^{M}\sum_{j=1}^{i-1}\frac{Cov(r_i, r_j)}{Cov(r_i, r_i)}}{M}}$$

where M is the number of output tasks selected.

Remark 1 *Note that we make an assumption on the distance between the marked point process obtained between the source and the target processes, not with the underlying distribution. This observation is key as we assume a bounded distance between these informative processes. This results may therefore suffer when the two considered data-generating processes present strong dissimilarities. From an application point of view, this means, for instance, that the theorem holds when practitioners have similar training and incentive, but may break if the healthcare recommendations differ, e.g. public vs. private systems.*

Remark 2 *Further theoretical work could refine this proof to our specific context in which (i) each additional task aims to model the data-generating process, and (ii) both observational process extract the necessary information for patient's diagnosis. Additionally, one could also provide a bound on the loss difference under a C-Lipschitz loss function.*

A.2 DeepJoint

This section further describes and formalises the proposed model.

A.2.1 Notations

The studied medical setting consists of a population of N individuals with a vector of laboratory tests $lab_{i,j}$ – with missing values – observed through time, in which the notation $j \in [\![0, l_i]\!]$ denotes the jth observation for patient i at time $t_{i,j} \in \mathbb{R}^+$, where $t_{i,0} = 0$ is the patient's admission to the ICU and l_i is the last observation for patient i. The proposed approach focuses on modelling the clinical presence aspect of these data. To do so, we introduce the two following notations. First, we model the elapsed time between two observations: $\epsilon_{i,j} = t_{i,j} - t_{i,j-1}$, note the subscript i underlines that observations may not happen at similar times for different patients. Second, the data-generating process is also characterised by which laboratory tests practitioners perform. Consequently, we introduce $m_{i,j} = [\mathbb{1}_{\text{Lab } k \text{ observed at } t_{i,j}}]_{k \in \text{labs}}$ an indicator vector of observed laboratory tests at observation time $t_{i,j}$.

This work models the survival outcome after the first 24 hours post-admission. Each patient has an associated time of the end of follow-up T_i and the type of observed event d_i for which $d_i = 1$ signifies that the patient died, $d_i = 0$ means right-censoring.

A.2.2 Architecture

The proposed architecture decomposes clinical presence into three dimensions, modelled through three different neural networks. Each relies on the embedding $h_{i,j}$ outputted by a Recurrent Neural Network (RNN) at the observation time $t_{i,j}$. To extract this embedding, we adopted a Long Short Term Memory (LSTM) network [45] that models the sequential nature of the observed data and captures temporal dependencies. Note that the inputs are the series of observations at *irregular times*. The resulting embedding is then leveraged to model the survival outcome.

We now describe the networks used for each of these tasks.

Longitudinal process The latent state $h_{i,j}$ is leveraged through a multi-layer perceptron L to model the future value of each laboratory test to be observed after a period t from the prediction time $t_{i,j}$. Assuming a Gaussian distribution for the observed data, the neural network L models the mean and variance of each future covariate given both the embedding and the prediction time, i.e.

$$\widehat{lab}_i(t) \sim \mathcal{N}(\mu_i(t), \sigma_i^2(t))$$

with $\mu_i(t)$ and $\sigma_i(t)$ the means, variances vectors outputted by $L(h_{i,j}, t)$.

Missingness process A second multi-layer perceptron M leverages $h_{i,j}$ to model the likelihood of observing the different covariates, i.e. which tests are likely to be performed after the period t. One can assume a Bernoulli distribution parameterised by a neural network over the missingness patterns, i.e.

$$\hat{m}_i(t) \sim \mathcal{B}ern(M(h_{i,j}, t))$$

Temporal process A monotonic positive neural network I [46] leverages $h_{i,j}$ to model the recurrent patterns of observations. A monotonic positive neural network is constrained to have positive weights – by using the square of the weights – and a final Softplus layer to guarantee the positivity, monotonicity and boundaries of the outputted likelihood of observing any new laboratory test after time t. This approach allows the computing of the risk of observing an event at any time Δ after t minutes by directly outputting the cumulative hazard I while avoiding any parametric assumption, i.e.

$$\mathbb{P}(\Delta > t) = \exp(-I(h_{i,j}, t))$$

Survival outcome Finally, to model the hazard function $\lambda(t)$, we used the DeepSurv model [30] that leverages a multi-layer perceptron S to extract a non-linear covariates shift used in a standard multiplicative proportional hazards Cox model. Given λ_0, the baseline hazard, the hazard of observing an event at horizon t is expressed as:

$$\lambda(t) = \lambda_0(t) \exp(S(h_{i,j}))$$

The final model, therefore, combines state-of-the-art architectures in a novel way to jointly model clinical presence and survival outcome. This results in a latent representation $h_{i,j}$, which embeds the data-generating process, then leveraged for survival modelling.

A.2.3 Training

Training consists in maximising the likelihood of observations and the survival likelihood. Each objective is back-propagated simultaneously by averaging the loss of the different tasks: survival, longitudinal, timing and missingness processes. We ensure that no objective is over-represented in the average by using a dynamic weighting average scheme [47], as imbalances can greatly impact performance [48]. Each loss is weighted by its relative change at iteration e as follows:

$$\forall\, task \in \{L, I, M\}, w_{task}(e) = \frac{l_{task}(e)}{l_{task}(e-1) \cdot \theta}$$

which is then normalised between the three clinical presence tasks using a Softmax. l is the validation log-likelihood for the given task and θ is a temperature hyperparameter that controls softness, i.e. larger values would lead to equal weights.

Longitudinal process This first neural network aims to maximise the likelihood of the next *observed* laboratory test values at time $\epsilon_{i,j}$. The assumption of normality leads to the use of the Gaussian log-likelihood loss:

$$\begin{aligned} l_L &= \sum_i \sum_{j \in [\![1, l_i - 1]\!]} m_{i,j+1} \cdot \log \mathcal{N}(lab_{i,j+1} | \mu_{i,j}, \sigma^2_{i,j}) \\ &= -\sum_i \sum_{j \in [\![1, l_i - 1]\!]} m_{i,j+1} \cdot \left(\frac{(lab_{i,j+1} - \mu_{i,j})^2}{2{\sigma_{i,j}}^2} + \log \sqrt{2\pi {\sigma_{i,j}}^2} \right) \end{aligned}$$

with $\mu_{i,j}$ and $\sigma_{i,j}$ the means, variances vectors outputted by $L(h_{i,j}, \epsilon_{i,j+1})$ and $*$ the element-wise multiplication. Note the filtering through $m_{i,j+1}$ to avoid a penalty on unobserved data.

Missingness process Similarly, the loss for the missingness process results in the binary cross entropy loss:

$$\begin{aligned} l_M &= \sum_i \sum_{j \in [\![1, l_i - 1]\!]} \log \mathcal{B}ern(m_{i,j+1} | M(h_{i,j}, \epsilon_{i,j+1})) \\ &= -\sum_i \sum_{j \in [\![1, l_i - 1]\!]} (m_{i,j+1} \cdot \log[M(h_{i,j}, \epsilon_{i,j+1})] + (1 - m_{i,j+1}) \cdot \log[1 - M(h_{i,j}, \epsilon_{i,j+1})]) \end{aligned}$$

Temporal process Monontonic networks allow the exact computation of the likelihood without the need for distributional assumptions on the inter-observation distribution. The model outputs the cumulative intensity at horizon t, and automatic differentiation results in the instantaneous intensity at no extra computational cost in the context of the neural networks' training procedure.

$$l_I = \sum_i \sum_{j \in [\![1, l_i - 1]\!]} \left(I(h_{i,j}, \epsilon_{i,j+1}) - \log \frac{\partial I(h_{i,j}, t)}{\partial t}_{t = \epsilon_{i,j+1}} \right)$$

Survival outcome DeepSurv relies on the proportional hazards assumption made by the Cox model resulting in the optimisation of the partial log-likelihood:

$$l_S = \sum_{i, d_i = 1} \left(S(h_{i,j}) - \log \sum_{k, T_k > T_i} \exp(S(h_{i,j})) \right)$$

Finally, the population baseline hazard, $\lambda_0(t)$, is estimated as a piece-wise constant based on the training population.

The final loss is defined at iteration e as

$$l(e) = (1 - \alpha) l_S + \alpha \sum_{task \in \{L, I, M\}} w_{task}(e) \cdot l_{task}(e)$$

With α, a hyperparameter balancing between survival and clinical presence. This loss is computed on a validation set for early stopping of the multi-task training. Then, we perform a fine-tuning of the network S with all other weights fixed with early stopping.

A.3 MIMIC III - Experiments

A.3.1 Data characteristics

Table 1 presents the demographic characteristics of the studied population and Table 2 summarises the set of tests selected with the mean number of tests performed during the 24 hours post admission and their mean values. The results are presented at the population level and differentiated by the subgroups used to study the impact of clinical presence shift.

Table 1: MIMIC III - Population characteristics between patient admitted on weekdays and weekends.

			Population		
			Overall	Weekday	Weekend
		Number of patients	30,834	23,359	7,475
Outcome	Length of stay (in days*)		10.05 (10.49)	10.05 (10.60)	10.03 (10.15)
	Death	Overall (%)	13.86	13.50	15.00
		1 day+ (%)	1.15	1.13	1.20
		7 days+ (%)	7.05	6.79	7.88
		14 days+ (%)	10.33	9.94	11.52
Demographics	Gender	Male (%)	56.43	56.38	56.62
		Female (%)	43.57	43.62	43.38
	Ethnicity	White (%)	71.84	72.52	69.74
		Other (%)	14.52	14.03	16.04
		Black (%)	7.88	7.87	7.92
		Hispanic (%)	3.32	3.19	3.73
		Asian (%)	2.43	2.39	2.57
	Insurance	Public (%)	66.15	65.68	67.59
		Private (%)	32.65	33.31	30.58
		Self Pay (%)	1.21	1.01	1.82

* Mean (std)
+ After first day of observation

Table 2: MIMIC III - List of laboratory tests used with associated mean number of tests and values (and standard deviations).

Laboratory test	Number of tests	Value
Anion gap	1.84 (1.00)	13.87 (3.23)
Bicarbonate	1.88 (1.00)	24.15 (4.34)
Blood urea nitrogen	1.90 (1.00)	24.91 (20.36)
Chloride	1.92 (1.03)	105.09 (5.74)
Creatinine	1.91 (1.00)	1.33 (1.38)
Glucose	2.84 (2.59)	136.77 (49.89)
Hematocrit	2.89 (2.36)	32.89 (5.34)
Hemoglobin	2.39 (2.04)	11.09 (1.91)
Magnesium	1.81 (0.97)	1.98 (0.37)
MCH*	1.76 (0.96)	30.29 (2.52)
MCH concentration	1.76 (0.96)	33.82 (1.52)
Mean corpuscular volume	1.76 (0.96)	89.63 (6.65)
Platelets	1.88 (1.10)	224.07 (109.73)
Potassium	2.01 (1.08)	4.13 (0.55)
Red blood cell count	1.87 (1.02)	5.23 (15.24)
Sodium	1.93 (1.08)	138.89 (4.38)
White blood cell count	1.88 (1.03)	11.83 (13.08)

* Mean corpuscular hemoglobin.

A.3.2 Hyperparameters tuning

All models' hyper-parameters were selected over the following grid of hyperparameters (if appropriate) following 100 iterations of random search. All experiments were ran on a A100 GPU over 100 hours for each experiment.

Table 3: Grid for hyperparameters search

	Hyperparameter	Values
Training	Learning rate	$10^{-3}, 10^{-4}$
	Batch size	100, 250
	α	$0.1, 0.3, 0.5$
	θ	2^*
RNN	Layers	1, 2, 3
	Hidden nodes	10, 30
Survival	Layers	0, 1, 2, 3
	Nodes	50
Clinical Presence	Longitudinal	Same parameters explored as survival
	Temporal	Same parameters explored as survival
	Missing	Same parameters explored as survival

* Following the results from [47].

A.3.3 Weekend - Weedays differences

Figure 3: Robustness evaluation between patients admitted on weekdays and weekends.

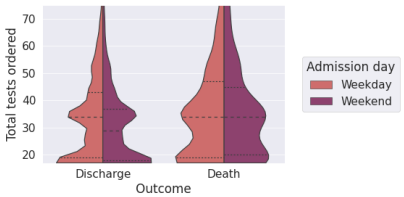

Figure 4: Total number of tests performed in the 24 hours after admission to the ICU.

A.3.4 Modelling clinical presence

Table 4 presents the discriminative results on the testing set of a random split of the population for which patients are randomly assigned to train and testing sets. The proposed methodology presents similar performance to Feature that is based on the same inputs. However, as shown Deepjoint presents better transportability. Similarly, Count seems to achieve better than Last in this setting but presents strong overfitting when transfered.

Table 4: Time Dependent C-Index - Mean (95%-CI) – *Higher is better.*

	Evaluation horizons (*in days after last observation*)		
Models	1	7	14
Last	0.851 (0.004)	0.734 (0.002)	0.691 (0.002)
Count	0.860 (0.004)	0.746 (0.002)	**0.699** (0.002)
Ignore	0.857 (0.003)	0.740 (0.002)	0.695 (0.002)
Resample	0.845 (0.004)	0.740 (0.002)	0.695 (0.002)
GRU-D	0.855 (0.004)	0.743 (0.002)	0.697 (0.002)
Feature	**0.874** (0.003)	**0.749** (0.002)	0.698 (0.002)
DeepJoint	0.871 (0.003)	0.748 (0.002)	0.695 (0.002)

Figure 5 and Table 5 present the Brier score obtained on a random split of patients, showing little difference in the calibration of the different models at shorter time horizons.

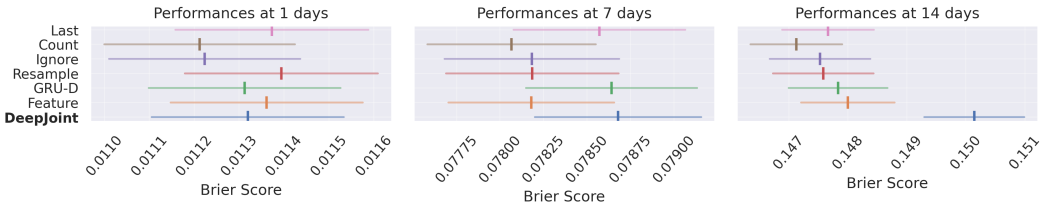

Figure 5: Brier score obtained on a random subset of the population.

Table 5: Time dependent Brier score - Mean (95%-CI) – *Lower is better.*

Models	Evaluation horizons (*in days after last observation*) 1	7	14
Last	0.011 (0.000)	0.079 (0.000)	0.148 (0.001)
Count	0.011 (0.000)	0.078 (0.000)	0.147 (0.001)
Ignore	0.011 (0.000)	0.078 (0.001)	0.148 (0.001)
Resample	0.011 (0.000)	0.078 (0.000)	0.148 (0.001)
GRU-D	0.011 (0.000)	0.079 (0.000)	0.148 (0.001)
Feature	0.011 (0.000)	0.078 (0.000)	0.148 (0.001)
DeepJoint	0.011 (0.000)	0.079 (0.000)	0.150 (0.001)

A.3.5 Transfer performances

Figure 6 shows the performance on patients admitted on weekends. This echoes the conclusion made in Section 5 that the proposed model is more robust to change in the clinical presence at shorter time horizons. However, this advantage fades as clinical presence reflects short term changes.

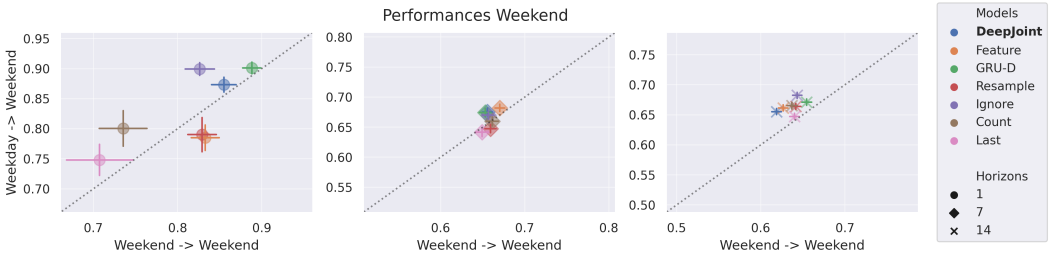

Figure 6: Robustness evaluation between patients admitted on weekdays and weekends.

Tables 6, 7 and 8 present the discriminative difference on the weekends-admissions testing set between the model trained on weekends-admissions and the model transferred from weekdays to weekends (and the opposite scenario).

At short time horizons, methodologies that leverage clinical presence present more robust result. Note that the proposed methodology present better performance than Feature based on the same inputs. Additionally, for weekends-admissions evaluation, **GRU-D** presents state-of-the-art predictive performance and robustness but this result didn't generalise to the other settings. Interestingly, ignoring the clinical process (**Resample** or **Ignore**) seems to be less robust to a change in the data-generating process, echoing the remark made by [3] about the difficulty of ignoring it.

Table 6: Performance of models trained and tested on the same type of patients (Internal), transferred from the other setting (Transfer) and difference in performance between these settings (Difference) at one day after observation period - Mean (95%-CI). *Dashed lines separate models based on different inputs.*

	Horizon: 1 day after observation					
	Evaluated on weekends			Evaluated on weekdays		
Models	Internal	Transfer	Difference	Internal	Transfer	Difference
Last	0.707 (0.201)	0.748 (0.132)	0.041 (0.021)	0.813 (0.043)	0.823 (0.046)	0.010 (0.004)
Count	0.735 (0.145)	0.801 (0.151)	0.065 (0.010)	0.813 (0.043)	0.789 (0.054)	-0.024 (0.006)
Ignore	0.826 (0.089)	0.899 (0.053)	0.073 (0.013)	0.822 (0.045)	**0.834 (0.045)**	0.012 (0.006)
Resample	0.829 (0.086)	0.790 (0.145)	-0.039 (0.022)	0.805 (0.052)	0.758 (0.057)	-0.047 (0.004)
GRU-D	**0.888** (0.055)	**0.901** (0.050)	**0.013** (0.005)	0.831 (0.034)	0.825 (0.051)	-0.006 (0.005)
Feature	0.833 (0.085)	0.785 (0.108)	-0.048 (0.006)	0.820 (0.048)	0.818 (0.048)	**-0.003** (0.006)
DeepJoint	0.855 (0.074)	0.874 (0.063)	0.018 (0.008)	**0.839 (0.042)**	0.829 (0.044)	-0.010 (0.005)

Table 7: Performance of models trained and tested on the same type of patients (Internal), transferred from the other setting (Transfer) and difference in performance between these settings (Difference) at 7 days after observation period - Mean (95%-CI)

	Horizon: 7 days after observation					
	Evaluated on weekends			Evaluated on weekdays		
Models	Internal	Transfer	Difference	Internal	Transfer	Difference
Last	0.649 (0.038)	0.642 (0.037)	-0.008 (0.004)	0.718 (0.021)	0.703 (0.021)	-0.016 (0.002)
Count	0.662 (0.036)	0.660 (0.038)	**-0.002** (0.004)	0.727 (0.019)	0.716 (0.023)	-0.011 (0.003)
Ignore	0.657 (0.034)	0.670 (0.034)	0.013 (0.004)	0.729 (0.021)	0.711 (0.022)	-0.018 (0.003)
Resample	0.659 (0.033)	0.648 (0.034)	-0.012 (0.005)	0.719 (0.022)	0.676 (0.025)	-0.043 (0.003)
GRU-D	0.653 (0.036)	0.675 (0.036)	0.021 (0.003)	**0.731** (0.019)	0.712 (0.022)	-0.019 (0.002)
Feature	**0.670** (0.030)	**0.682** (0.031)	0.012 (0.004)	0.722 (0.021)	0.719 (0.023)	**-0.002** (0.004)
DeepJoint	0.656 (0.030)	0.675 (0.033)	0.020 (0.004)	0.726 (0.020)	**0.725** (0.022)	**-0.002** (0.003)

Table 8: Performance of models trained and tested on the same type of patients (Internal), transferred from the other setting (Transfer) and difference in performance between these settings (Difference) at 14 days after observation period - Mean (95%-CI)

	Horizon: 14 days after observation					
	Evaluated on weekends			Evaluated on weekdays		
Models	Internal	Transfer	Difference	Internal	Transfer	Difference
Last	0.640 (0.029)	0.647 (0.027)	**0.007** (0.003)	0.710 (0.019)	**0.690** (0.020)	-0.020 (0.002)
Count	0.637 (0.031)	0.665 (0.029)	0.028 (0.003)	0.712 (0.019)	0.685 (0.020)	-0.027 (0.003)
Ignore	0.643 (0.028)	**0.682** (0.026)	0.039 (0.003)	**0.715** (0.020)	0.685 (0.021)	-0.030 (0.003)
Resample	0.642 (0.029)	0.664 (0.024)	0.022 (0.004)	0.713 (0.020)	0.664 (0.022)	-0.049 (0.003)
GRU-D	**0.655** (0.030)	0.671 (0.028)	0.016 (0.003)	0.713 (0.019)	0.689 (0.019)	-0.025 (0.002)
Feature	0.627 (0.028)	0.662 (0.026)	0.035 (0.004)	0.704 (0.021)	0.688 (0.022)	**-0.017** (0.003)
DeepJoint	0.619 (0.028)	0.655 (0.026)	0.037 (0.004)	0.712 (0.019)	0.686 (0.022)	-0.026 (0.003)