A deep clustering model for PDAC patient stratification using multi-omics data

INF-8605 Interpretability in deep learning

Alberto Lopez

2023-07-21

Abstract

Pancreatic ductal adenocarcinoma (PDAC) patients face the lowest survival among all cancer patients in Europe. Since Western societies have the highest incidence of pancreatic cancer, it has been projected that PDAC will soon become the second leading cause of cancer-related deaths. The main medical challenges of PDAC are that patients with similar somatic genotypes show a wide range of disease phenotypes, and due to its extreme malignancy, it is difficult to collect large and high-quality datasets that allow systematic analyses of the hidden genotype-phenotype patterns. Artificial Intelligence (AI) is currently transforming the field of healthcare and represents a promising technology to integrate various datasets and optimize evidence-based decision making. However, we cannot interpret directly how and why a decision has been made by most of machine learning models. In this study, we developed a deep clustering model for PDAC patient stratification. We then exploited it in terms of explainability, aiming to unbox how black-box choices are made. We hope this study helps to introduce AI in real clinical applications, where the knowledge to make decisions is crucial. The code of this project is publicly available in GitHub.

1. Introduction

PDAC is an extremely lethal disease, accounts for 85% of all pancreatic cancers and is the most aggressive subtype. In 2020, 495773 new cases and 466003 deaths worldwide were related to pancreatic cancer, that is, 2.6% of all new cancer diagnoses and 4.7% of all cancer deaths, respectively[1]. It ranks as the third highest cause of cancer-related deaths in the United States and the sixth in China[2,3]. Experts predict that by 2030, PDAC will reach the second place in terms of cancer mortality[4]. The five-year survival rate in the United States stands at a 11%, which is the lowest among all cancers.

This poor prognosis is primarily due to the lack of clear and distinctive symptoms, as well as reliable biomarkers for early detection, with only about 20% of patients being diagnosed at an

early stage[5]. Additionally, the cancer tends to spread aggressively, leading to poor responses to treatments[6]. Unfortunately, despite advancements in pancreatic cancer research, there has been little improvement in the ratio of deaths to new cases over the past few decades. However, pancreatic cancer detected by screening has a 5-year survival rate of 73.3% and a median survival time of 9.8 years, compared with 1.5 years for those seen by non-screening[7]. Given these challenges, there is an urgent need to gain a deeper understanding of the molecular characteristics involved in PDAC development. This knowledge is crucial for developing innovative diagnostic and therapeutic strategies to tackle this disease.

AI is a branch of computer science focused on developing intelligent machines that can exhibit human-like intelligence[8]. Currently, numerous researchers are working on applying AI in the field of oncology[9]. AI offers greater adaptability and expandability compared to traditional biometric methods, enabling its use in various tasks. One of these tasks is precisely patient stratification. The definition of PDAC subtypes, currently just based on the histology and genotype[10], from molecular perspectives would enable personalized diagnosis and treatment by providing more effective therapeutic plans.

Most of PDAC patient stratification studies used single-omics analyses. However, analyzing each type of omics data separately may yield inconsistent results[11]. This is because each omics data type carries distinct information and is associated with different mechanisms related to oncogenesis and tumor development. To enhance our understanding further, we require tools that can identify cohesive subtypes using multiple omics data sources. This is another advantage AI holds, the capability to assimilate diverse data types and comprehend intricate connections between variables in a flexible and trainable manner.

Deep Learning (DL) has significantly impacted various fields of machine learning and artificial intelligence[12,13]. It has brought transformational changes and gained popularity in many areas, mostly related to supervised tasks. The success of supervised deep learning has motivated the development of deep learning-based clustering algorithms[14]. These clustering approaches leverage the capabilities of DL to preprocess data, thereby improving the quality of clustering results and obtaining more meaningful clusters[15].

However, despite its undeniable potential, AI, and especially DL, struggles becoming a real application in medical domain. DL methods are perceived as black-box approaches due to limited understanding of how they work internally, and the lack of explainability has been criticized[16]. Therefore, explainability, considered as an important part of transparency (a key to achieve the desired trustworthy AI, according to the European Commision's High-Level Expert Group on AI) is crucial[17]. Explainable Artificial Intelligence (XAI) is an emerging research topic that focuses on unboxing how AI systems' black-box choices are made[18]. Explainability plays a significant role in generating insights about why and how a particular prediction was made by the model. Consequently, explainability becomes critical to understand the features captured by these methods and the attributes of samples on which the clusters are based. Specifically in medicine, interpretability can reveal shared characteristics among the most important biomarkers, and this information can contribute to recommending more precise treatments and exploring possibilities for repurposing drugs.

In this study, we created a PDAC patient stratification model using a deep learning-based clustering algorithm with multi-omics data (RNA-seq and methylation data). Once the model was created, we exploited it in terms of explainability. By understanding the underlying patterns learned by the model, we aimed to provide interpretable and actionable information regarding PDAC patient stratification.

2. Methods

Data

In this study, we obtained PAAD-TCGA data from the Firehose Broad GDAC using the R packages curatedTCGAData[19] and TCGAutils. We analyzed RNA-seq data (Illumina HiSeq, upper quartile normalized RSEM TPM gene expression values) and DNA methylation (Illumina Human Methylation 450). Common patients with all these omics were filtered by primary tumor and PDAC histological type. In total, we had 147 patients.

Preprocessing

Omics data were preprocessed using a Scikit-learn pipeline[20]:

- The gene expression dataset had 20,501 features and the pipeline was composed of: removing features with zero values in more than 20% of patients; keeping 50% most variable features using mean absolute error; removing correlated features with over 0.85 pearson correlation; log2 transformation; feature selection based on non-negative matrix factorization (NMF[21]), with the most important features (the final number was determined with an optimization) of each one of the 512 components; z-score normalization.
- Epigenomics dataset had initially 485,577 features, and the steps in the pipeline included: removing sexual chromosomes; filtering out features with missing values in more than 20% patients; keeping 10% most variable features using mean absolute error; missing value imputation using average values; feature selection based on non-negative matrix factorization (NMF), with the most important features (the final number was determined with an optimization) of the 512 components; z-score normalization.

Clustering model

A multi-view DL-based clustering algorithm was built using a multi-view deep autoencoder (employing Pytorch[22]) and a K-Means. A deep autoencoder (DAE) is an artificial neural network used for feature extraction, in a way that it maps the input data into a hidden representation. The layers that create this hidden representation (or embedding) are called encoder, while the layers that try to reconstruct the original input from it are named decoder. The branches of the encoder were merged one layer before the latent representation. Each block was composed of a linear, PReLU activation and a batch normalization layer. A joint loss function as the weighted (with a lambda coefficient) sum of the reconstruction error (mean absolute error, MAE) of the DAE and the sum of squared distances of samples to their closest cluster center was used. Both the multi-view deep autoencoder and the cluster centers

were initialized with a previous indepedent pretraining. This architecture is illustrated in Figure 1.

A hyperparameter optimization was carried out with Optuna[23] to ensure the best possible result, with the silhouette score[24] as the objective function. We executed the Tree-structured Parzen Estimator (TPE) algorithm for 1225 trials (the first 1000 were a random search) in a nested 5-fold cross validation. To avoid data leakage, we applied both the preprocessing pipeline and the model in each iteration. The hyperparameters to optimise were: the number of hidden layers (one or two before merging the branches of each modality), the number of neurons (with a number proportional to the previous layer), the latent space dimensionality (50-200), the number of input features (from 1 to 10 different features for each NMF component), the epochs (20-100), the parameter that controls the joint loss function (0.001-1) and the number of clusters (2-6). In addition, the learning rate was automatically determined using a learning rate finder strategy[25].

After the hyperparameter optimization, we trained a model using all the samples. The hard clustering assignments were transformed to soft assigments by computing the probability of a sample owing to a cluster with the inverse distance weighting.

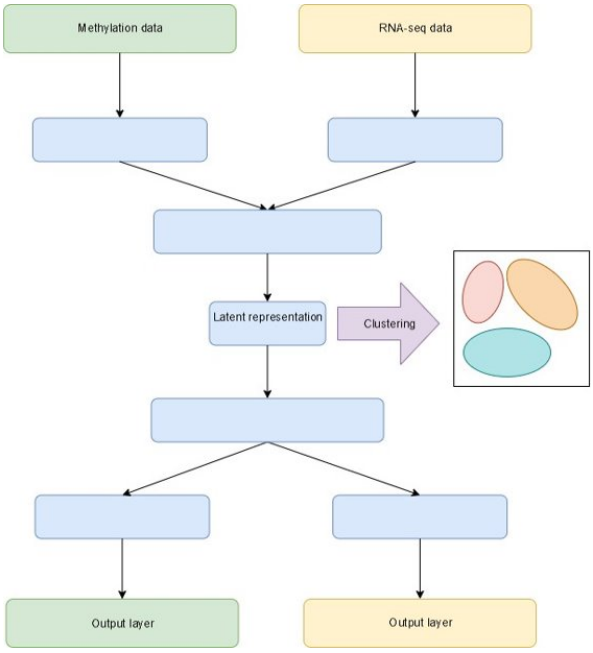

Figure 1: Algorithm architecture: the algorithm is composed of a multi-branch deep autoencoder, with a K-Means applied on the latent representation.

Multi-view contribution measure

To measure the contribution of each omics to the final predictions, we applied MM-SHAP, a performance-agnostic multi-view score based on Shapley values that reliably quantifies in

which proportions a multi-view model uses individual modalities. We adapted their functions for tabular data, and used Deep Shap (SHAP library[26]) to compute shap values.

Feature importance analysis

In order to compute the attributions with respect to the inputs of the model, we applied several algorithms available in Captum[27], with their respective default parameters. We used: integrated gradients[28] and noise tunnel[29] (add gausian noise to each input) to approximate the integral of gradients of the model's output with respect to the inputs along the path; gradient shap[30] to approximate SHAP values by computing the expectations of gradients by randomly sampling from the distribution of baselines; and feature ablation, which replaces each input feature with a baseline and computes the difference in output. We chose zero-valued baselines, which corresponde to, give the z-normalization, the average of the features.

Neuron importance analysis

Besides attributing to the inputs of the model, we also attributed to the layers of the model to understand which neurons appear to be more important. For this purpose, we computed the conductance[31] with respect to the latent representation layer of the model. The conductance of a hidden neuron is the flow of attribution via this hidden unit[32]. The normalized values were compared with the mean weights of each neuron in the embedding layer.

Statistical analysis

Clusters were plotted applying a PCA[33] on the latent representation. Visualizations in 2D were compared with clusters generated with K-Means and PCA, T-SNE[34] and UMAP[35] on a concatenated multi-omics matrix.

To evaluate if the clusters were biologically meaningful, we applied a survival analysis and an enrichment analysis of clinical labels. For survival analysis, log-rank test (lifelines python package[36]) indicates statistically significant difference existence in survival profiles between different cancer subtypes. For enrichment analysis of clinical labels (using Scipy package[37]), we applied Kruskal-Wallis test on the age at initial diagnosis, as well as a chi-square test on gender and three discrete clinical pathological parameters quantifying the progression of the tumor (pathologic T), cancer in lymph nodes (pathologic N) and metastases (pathologic M).

3. Results

Hyperparameter optimization

The results of the hyperparameter optimization are shown in Figure S1. The silhouette score can take values between -1 and 1. The best score of the optimization was 0.28. The nested 5-fold cross-validation allowed us to avoid overfitting, with a mean training, validation and testing score of 0.29, 0.28 and 0.29, respectively. The total weighted losses of the models were 3.25, 2.18 and 2.10 for training, validation and testing. The model was composed of over 72M parameters.

From figure S1b and S1c, we can conclude the most important hyperparameter is the number of clusters, where two is the optimal value. The second most important hyperparameter was the number of epochs, the minimum possible value 20 being the best. This is probably due to the previous pretraining, that allowed us to start the training with better weight inizialization. Also, we observed the best scores came when more features were selected as input, as well as a bigger neural network architecture (number of layers and number of neurons). The model obtained better scores with a 50-dimensional latent space. Lastly, the optimal lambda coefficient, that controls the trade-off between the loss function of the autoencoder and the loss function of the clustering, is near 0.01.

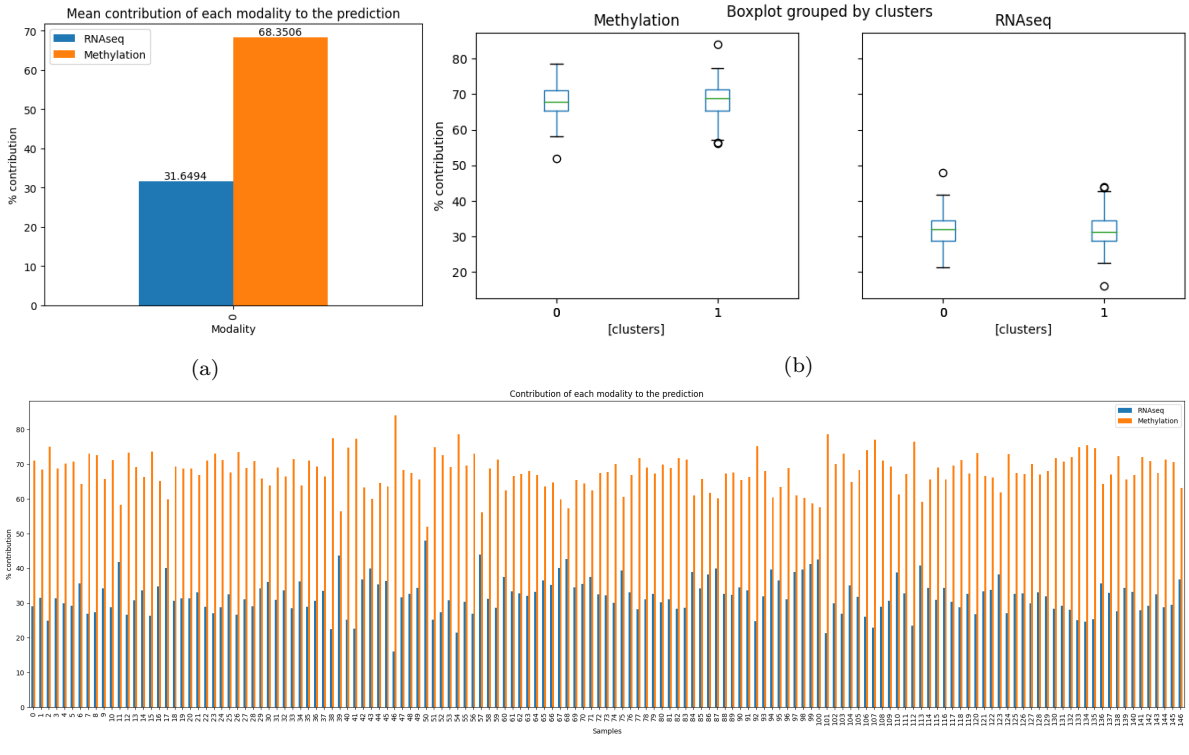

Figure 2: Multi-omics contribution. a) Mean MM-SHAP value of each omics data type; b) Boxplot of MM-SHAP values of each omics data type for both clusters; c) Contribution of each omics modality to the cluster prediction based on MM-SHAP values.

Measuring multi-view contribution

To understand better about the biology of PDAC, we were interested in seeing how gene expression vs methylation patterns affect predictions. The multi-view contribution scores are shown in Figure 2. Figure 2a shows methylation data was the most important omics modality, which contributes 68% on average to the final predictions. The remaining 32% on average was provided by RNA-seq. We also studied if there was a difference in the contributions for the clusters. Figure 2b shows that predictions in both clusters use similar multi-omics

contributions. In addition, as plotted in Figure 2c, the contributions of the modalities for each sample is quite stable.

Feature importance analysis

Most important features based on attributions with respect to the inputs were computed. A visualization of the attribution scores of those with the 10 highest mean absolute values is shown in Figure 3. This figure also compare them with the learned model weights. The values of the learned model weights provide insights into the relationships between the clusters and the features. A weight of zero indicates no correlation, while positive weights suggest positive correlations and negative weights imply negative correlations. However, due to the network's multiple layers, these weights might not have a straightforward connection with the labels.

One of the main observations we can extract is all of them are methylation features. The second one is that attribution algorithms disagree on assigning importance scores. In fact, we observe Gradient Shap behaves differently than the other methods. Furthermore, they are not aligned with weights, probably because of the multiple layers in the network, as we said previously. The most important feature based on the analysis was "cg26680608". Interestingly, the four algorithms agree about the high importance of this feature.

However, figure S2 shows there is no difference in the values of both clusters of the top 25 features based on this strategy.

Neuron importance analysis

As the function of the whole neural system is a combination of the different neurons in each layer working together to get the predictions, we were also interested in understanding which neurons appear to be more important. For this purpose, we analysed the conductance of the latent representation layer.

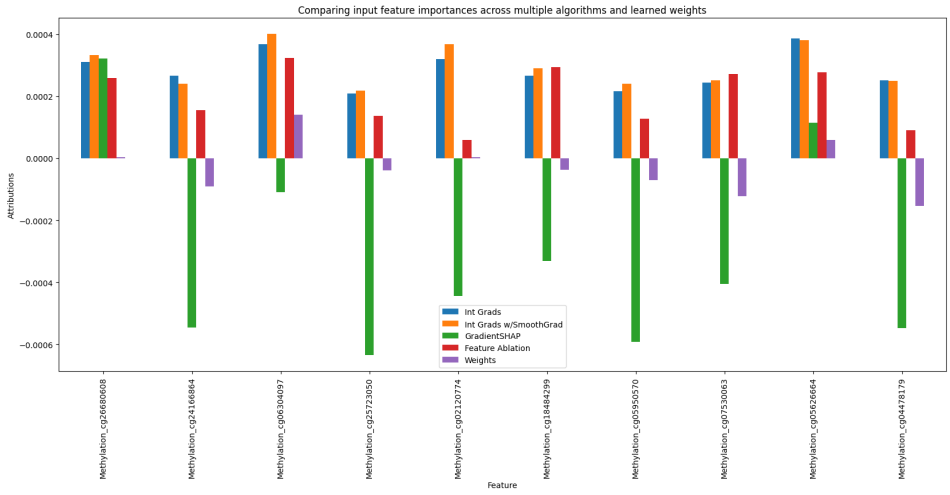

Figure 3: Top 10 features computed using several attribution algorithms.

In Figure 4, we can observe both the weights and the attributions have high fluctuations. It seems there is no relation between the weights and the attributions. However, we must remember we aggregated all the weights for each neuron. This figure suggests neurons 31, 11 and 2 are the most important ones. The distribution of the weights of these three neurons are plotted in figure S3, and compared with neurons 26, 37 and 48, that seemed to be the less important ones. As we can see, neurons 31, 11 and 2 appear to be learning strong features, while the weights of the neurons 26, 37 and 48 are very close to 0.

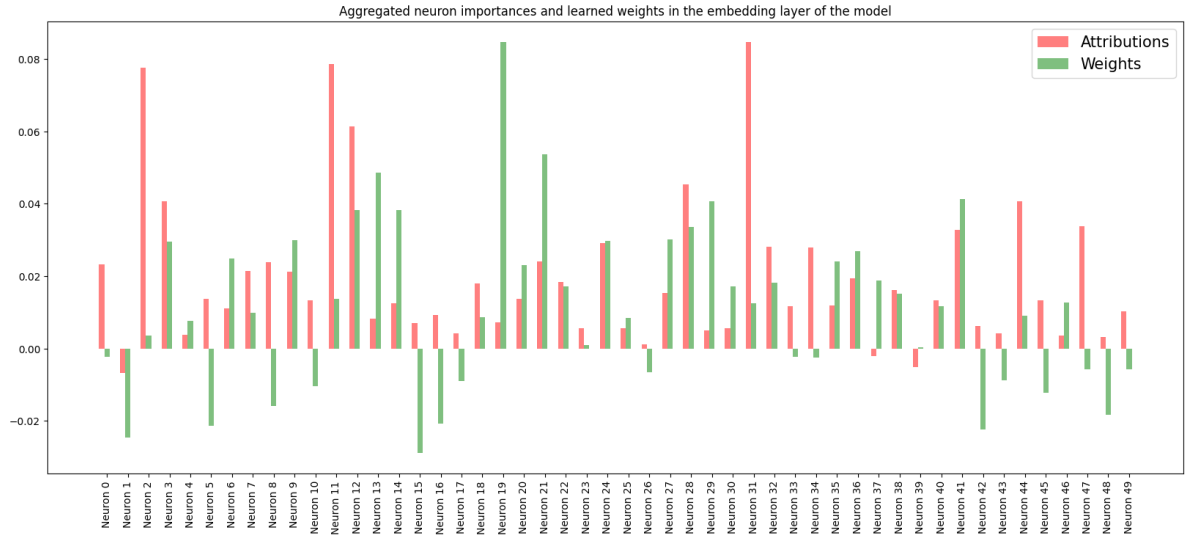

Figure 4: Aggregated neuron importances and learned weights in the embedding layer of the model.

Once we identified the important neurons in the embedding layer, we computed the neuron attributions through the conductance, to understand what each of these important neurons were looking at in the input. Figure S4 shows the most importat features for the neurons. Interestingly, it appears that the neuron 11 is activated mainly by RNA-seq features, while neuron 12 by methylation features. In contrast, both modalities are important for the neuron 31. However, same than before, the values of important features are not able to separate the clusters theirself based on figure S5.

Statistical analysis

The model identified two clusters with 119 and 28 samples. As shown in figure S6, the statistical tests revealed that the clusters were significantly related to neoplasm histologic stage.

Visualizations of the latent representation and comparison with other algorithms, such as PCA and UMAP appear in figure S7. Same than our model, these techniques neither found biologically relevant clusters based on the plot.

4. Discussion

We presented a deep clustering model based on multi-omics data for finding subgroups of PDAC patients. It analysed RNA-seq and methylation data, both high-dimensional datasets. The algorithm is composed of a multi-branch deep autoencoder and a K-Means applied on the embedding layer. The objective of the algorithm was to optimise a joint loss function for both the autoencoder and the clustering.

Through an optimization process, we were able to select the best combination of hyperparameters for the model. The hyperparameter optimization allowed us to find a model avoiding overfitting, as well as analysing the importance of each hyperparameter for the algorithm. Among them, the number of clusters was the most relevant to achieve the optimal model.

The multi-view contribution analysis showed methylation data was more important than RNA-seq for the model. However, there was no difference in the multi-view contribution between the clusters. The fact that the model uses both modalities in a relatively stable way could means it is avoiding unimodal collapse (when a unimodal model achieves similar accuracy on a task than a multimodal one).

We also illustrated that is possible to explain the most important features for an deep clustering model. We analysed these features in different ways: firstly, employing several gradient-based and perturbation-based algorithms; and then, understanding which input features activated the most important neurons in the embedding layer. Some of the features, such as "cg24166864" (the second most important one based on the attributions and the most important feature for the neuron 12) and "cg05626664" (the ninth most important one based on the attributions and the second most important feature for the neuron 12), were common for both techniques.

The fact that the boxplots for the most important features grouped by clusters do not show difference could means there is no an unique feature usefull for classifying the patients. Instead, it is the combination of some of them what allow us to stratify patients. Due to this, a future work would be to study a biomarker panel with the minimum features that allow to classify patients in these subgroups.

Based on the statistical tests, the model found clusters that were associated to the neoplasm histologic stage. It is usually difficult to find clusters that a different in a biologically meaningful way, as illustrated by Bo Yang et al[38], where most of state-of-the-art algorithms found only a few associations to clinical parameters.

In general, we could say we obtained a lower silhouette score and fewer significant associations to clinical parameters that other models usually find in other cancer types. This is probably related to the high dimensions of the datasets, and the very low sample size that we were working with. It is widely known that deep learning algorithms struggle to be applied on this type of datasets. So far, research community have not been able to find PDAC subgroups with good properties that can be applied in real cases, showing how difficult this field is. This study was just a proof-of-concepts for a bigger project, however, we succesfully applied several explainability methods, that will be able to use in next projects for a better understanding

of the models. Therefore, in the future we should introduce some modifications that allow the algorithm to be more effective with high-dimensional, low sample size datasets, and find biologically relevant clusters for PDAC patient stratification.

References

1. Sung, H. *et al.* Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians* **71**, 209–249 (2021).
2. Siegel, R. L., Miller, K. D., Fuchs, H. E. & Jemal, A. Cancer statistics, 2022. *CA: A Cancer Journal for Clinicians* **72**, 7–33 (2022).
3. Xia, C. *et al.* Cancer statistics in china and united states, 2022: Profiles, trends, and determinants. *Chinese Medical Journal* **135**, 584–590 (2022).
4. Vespasiani-Gentilucci & Antonelli-Incalzi. Pancreatic cancer: A review of clinical diagnosis, epidemiology, treatment and outcomes. (2018) doi:https://dx.doi.org/10.3748/wjg.v24.i43.4846.
5. Hruban, R. H. *et al.* Why is pancreatic cancer so deadly? The pathologist's view. *The Journal of Pathology* **248**, 131–141 (2019).
6. Maitra, A. & Hruban, R. H. Pancreatic cancer. *Annual Review of Pathology: Mechanisms of Disease* **3**, 157–188 (2008).
7. Dbouk, M. *et al.* The multicenter cancer of pancreas screening study: Impact on stage and survival. *Journal of Clinical Oncology* **40**, 3257–3266 (2022).
8. Rajkomar, A., Dean, J. & Kohane, I. Machine learning in medicine. *New England Journal of Medicine* **380**, 1347–1358 (2019).
9. Shimizu, H. & Nakayama, K. I. Artificial intelligence in oncology. *Cancer Science* **111**, 1452–1460 (2020).
10. Makohon-Moore, A. & Iacobuzio-Donahue, C. Pancreatic cancer biology and genetics from an evolutionary perspective. *Nature Reviews Cancer* **16**, (2016).
11. Sinkala, M., Mulder, N. & Martin, D. Machine learning and network analyses reveal disease subtypes of pancreatic cancer and their molecular characteristics. *Scientific Reports* **10**, 1212 (2020).
12. Goodfellow, I., Bengio, Y. & Courville, A. *Deep learning*. (MIT Press, 2016).
13. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *nature* **521**, 436 (2015).
14. Karim, M. R. *et al.* Deep learning-based clustering approaches for bioinformatics. *Briefings in Bioinformatics* **22**, 393–415 (2020).
15. Nutakki, G. C., Abdollahi, B., Sun, W. & Nasraoui, O. An introduction to deep clustering. in *Clustering methods for big data analytics: Techniques, toolboxes and applications* (eds. Nasraoui, O. & Ben N'Cir, C.-E.) 73–89 (Springer International Publishing, 2019). doi:10.1007/978-3-319-97864-2_4.
16. Amann, J., Blasimme, A., Vayena, E., Frey, D. & Madai, V. Explainability for artificial intelligence in healthcare: A multidisciplinary perspective. *BMC Medical Informatics and Decision Making* **20**, (2020).
17. Amann, D. A. B., Julia AND Vetter. To explain or not to explain?—artificial intelligence explainability in clinical decision support systems. *PLOS Digital Health* **1**, 1–18 (2022).
18. Yang, G., Ye, Q. & Xia, J. Unbox the black-box for the medical explainable AI via multi-modal and multi-centre data fusion: A mini-review, two showcases and beyond. *Information Fusion* **77**, 29–52 (2022).
19. Ramos, M. *et al.* Multiomic integration of public oncology databases in bioconductor. *JCO Clinical Cancer Informatics* 958–971 (2020) doi:10.1200/CCI.19.00119.

20. Pedregosa, F. *et al.* Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

21. Févotte, C. & Idier, J. Algorithms for nonnegative matrix factorization with the beta-divergence. *CoRR* **abs/1010.1763**, (2010).

22. Paszke, A. *et al.* PyTorch: An imperative style, high-performance deep learning library. *CoRR* **abs/1912.01703**, (2019).

23. Akiba, T., Sano, S., Yanase, T., Ohta, T. & Koyama, M. Optuna: A next-generation hyperparameter optimization framework. *CoRR* **abs/1907.10902**, (2019).

24. Rousseeuw, P. J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* **20**, 53–65 (1987).

25. Smith, L. N. No more pesky learning rate guessing games. *CoRR* **abs/1506.01186**, (2015).

26. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. *CoRR* **abs/1705.07874**, (2017).

27. Kokhlikyan, N. *et al.* Captum: A unified and generic model interpretability library for PyTorch. *CoRR* **abs/2009.07896**, (2020).

28. Sundararajan, M., Taly, A. & Yan, Q. Axiomatic attribution for deep networks. *CoRR* **abs/1703.01365**, (2017).

29. Adebayo, J. *et al.* Sanity checks for saliency maps. *CoRR* **abs/1810.03292**, (2018).

30. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. in *Advances in neural information processing systems* (eds. Guyon, I. et al.) vol. 30 (Curran Associates, Inc., 2017).

31. Shrikumar, A., Su, J. & Kundaje, A. Computationally efficient measures of internal neuron importance. *CoRR* **abs/1807.09946**, (2018).

32. Dhamdhere, K., Sundararajan, M. & Yan, Q. How important is a neuron? *CoRR* **abs/1805.12233**, (2018).

33. Halko, N., Martinsson, P.-G. & Tropp, J. A. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. (2010).

34. Maaten, L. van der & Hinton, G. Viualizing data using t-SNE. *Journal of Machine Learning Research* **9**, 2579–2605 (2008).

35. McInnes, L., Healy, J. & Melville, J. UMAP: Uniform manifold approximation and projection for dimension reduction. (2020).

36. Davidson-Pilon, C. Lifelines: Survival analysis in python. *Journal of Open Source Software* **4**, 1317 (2019).

37. Virtanen, P. *et al.* SciPy 1.0-fundamental algorithms for scientific computing in python. *CoRR* **abs/1907.10121**, (2019).

38. Yang, B., Yang, Y., Wang, M. & Su, X. MRGCN: cancer subtyping with multi-reconstruction graph convolutional network using full and partial multi-omics dataset. *Bioinformatics* **39**, btad353 (2023).

Supplementary

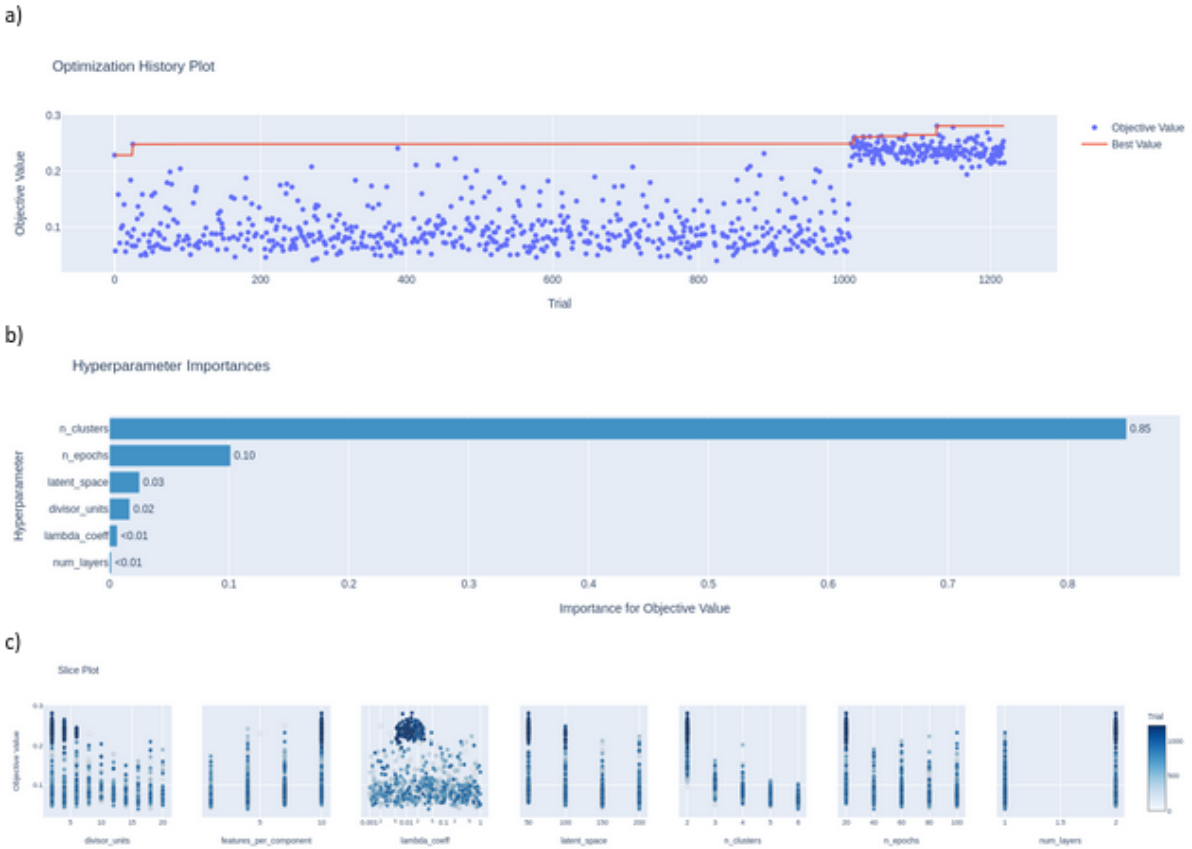

Figure S1. a) Optimization history plot of all trials in the study. The best score of 0.28 is reached at trial 1127. Before the trial 1000, a random search was used, while the TPE algorithm worked from that time; b) Hyperparameter importance plot; c) Parameter relationship plot.

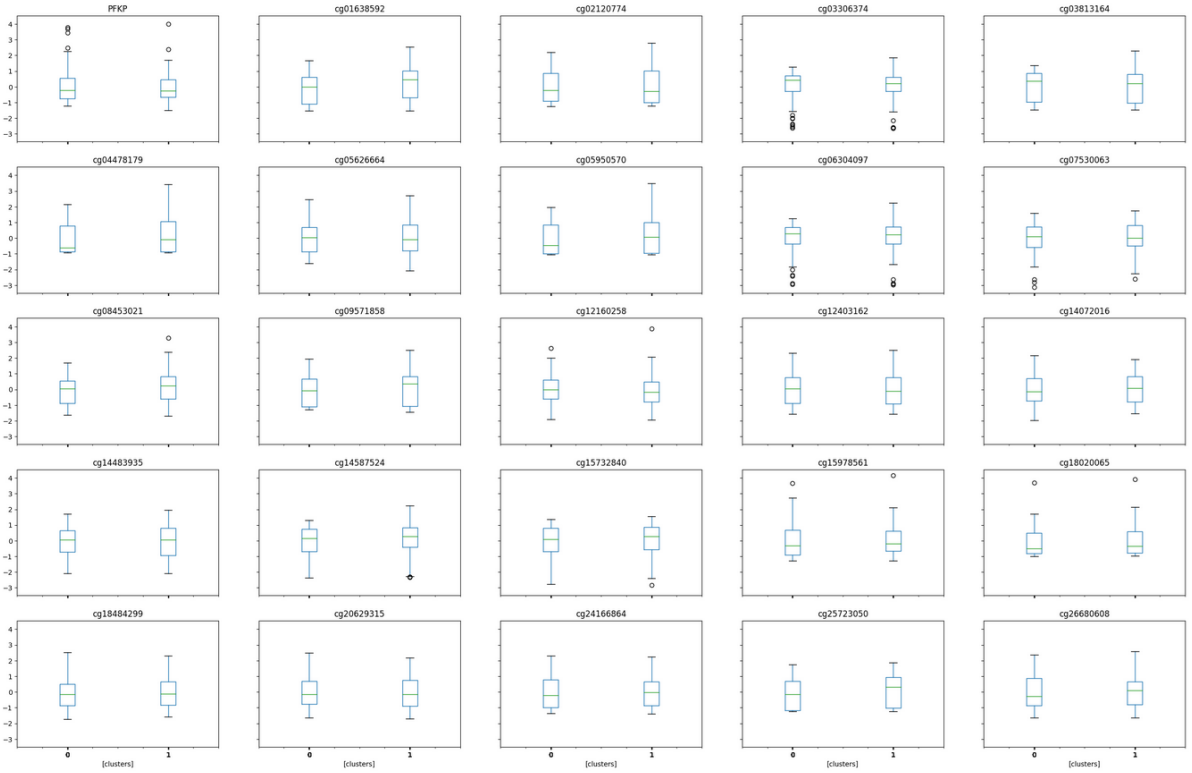

Figure S2. Boxplot for the top 25 features computed using several attribution algorithms grouped by cluster.

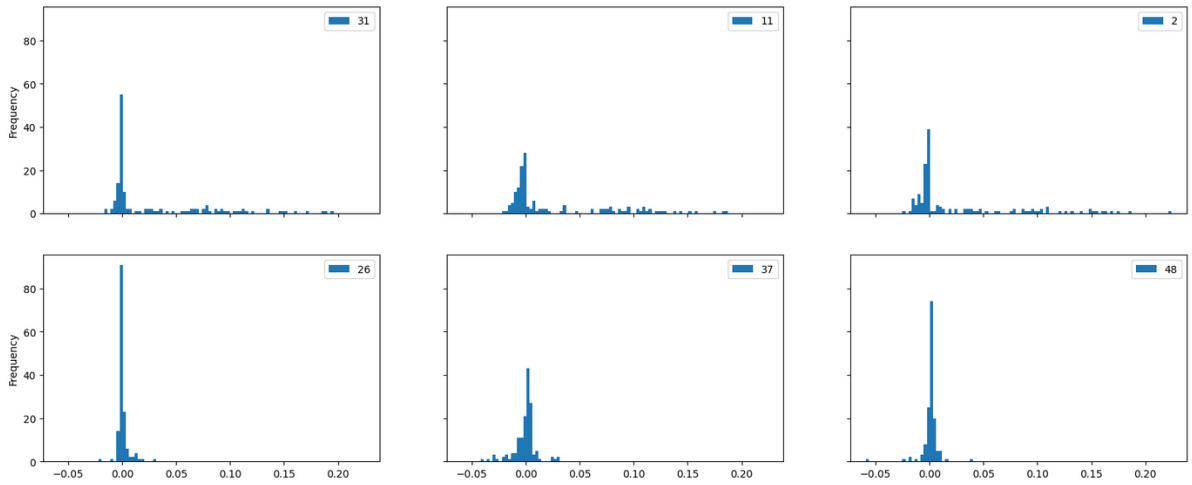

Figure S3. Distribution of weights in most and least important neurons in the embedding layer.

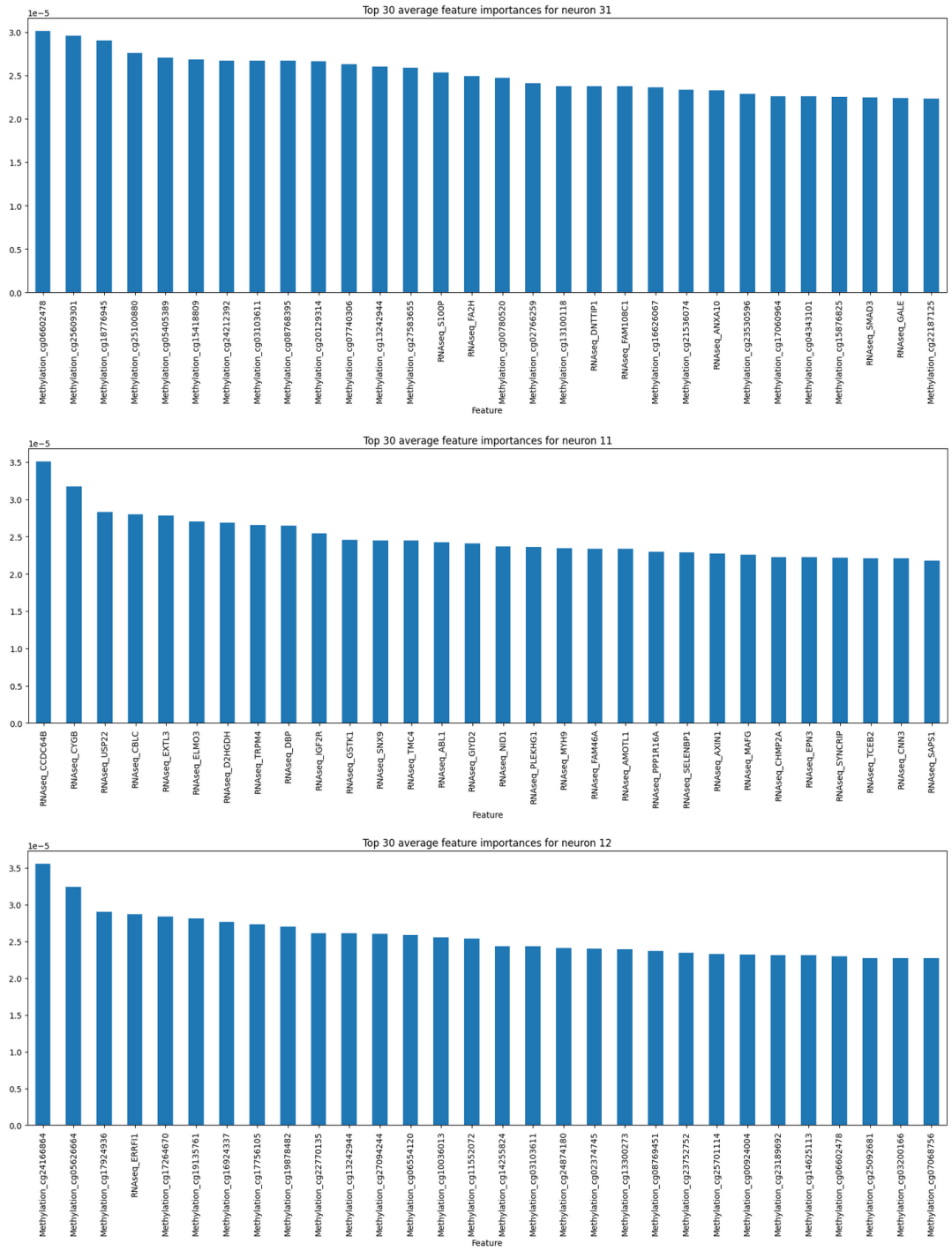

Figure S4. Top 30 features computed using neural conductance for the three most important neurons in the embedding layer.

Figure S5. Boxplot for the top 25 features computed using neural conductance for neuron 31 in the embedding layer grouped by cluster.

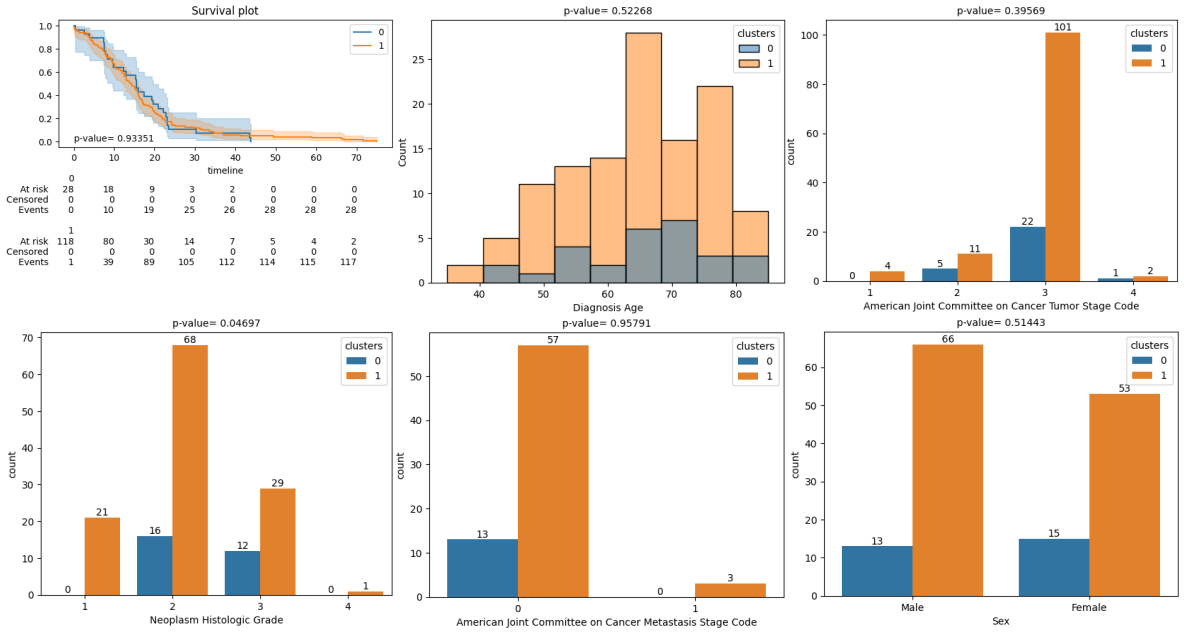

Figure S6. Survival plot, histogram based and bar plot for the clusters and clinical parameters.

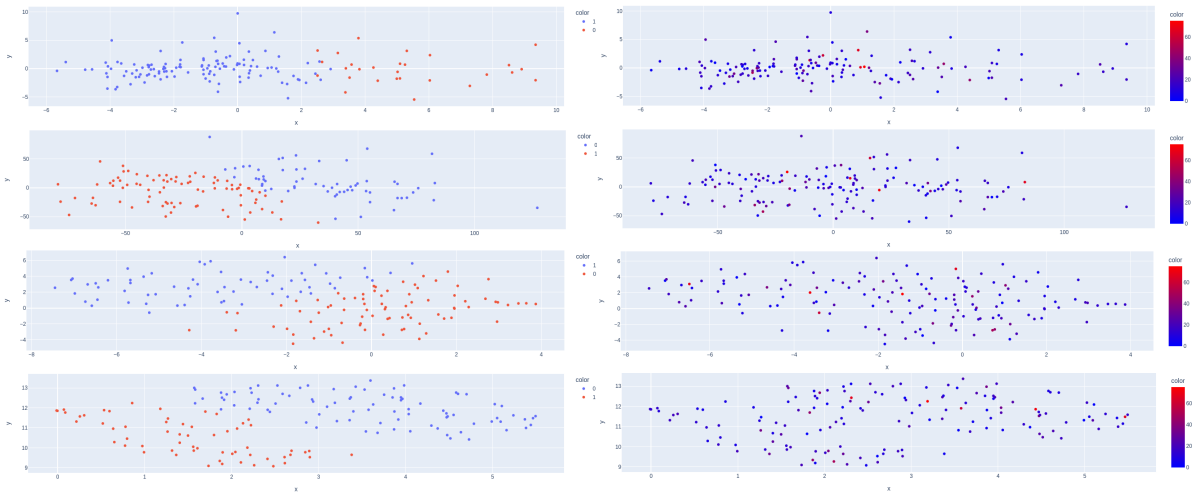

Figure S7. Sample visualization using several techniques. From up to down: latent representation of our model, PCA, T-SNE, UMAP. Left: grouped by clusters; right: colors based on survival.