Cancer Biology
Computational and Systems Biology

GENIUS: GEnome traNsformatIon and spatial representation of mUltiomicS data

Reviewed Preprint

Revised by authors after peer review.

About eLife's process

Reviewed preprint version 2
August 10, 2023 (this version)

Reviewed preprint version 1
May 31, 2023

Sent for peer review
March 1, 2023

Posted to bioRxiv
February 13, 2023

Mateo Sokač, Asbjørn Kjær, Lars Dyrskjøt, Benjamin Haibe-Kains, Hugo J.W.L. Aerts, Nicolai J Birkbak

Department of Molecular Medicine, Aarhus University Hospital, Aarhus, Denmark • Department of Clinical Medicine, Aarhus University, Aarhus, Denmark • Bioinformatics Research Center, Aarhus University, Aarhus, Denmark • Princess Margaret Cancer Centre, University Health Network, Temerty Faculty of Medicine, University of Toronto, Toronto, ON, Canada • Artificial Intelligence in Medicine (AIM) Program, Mass General Brigham, Harvard Medical School, Boston, MA • Departments of Radiation Oncology and Radiology, Brigham and Women's Hospital, Dana-Farber Cancer Institute, Harvard Medical School, Boston, MA • Radiology and Nuclear Medicine, CARIM & GROW, Maastricht University, The Netherlands

Abstract

The application of next-generation sequencing (NGS) has transformed cancer research. As costs have decreased, NGS has increasingly been applied to generate multiple layers of molecular data from the same samples, covering genomics, transcriptomics, and methylomics. Integrating these types of multi-omics data in a combined analysis is now becoming a common issue with no obvious solution, often handled on an ad-hoc basis, with multi-omics data arriving in a tabular format and analyzed using computationally intensive statistical methods. These methods particularly ignore the spatial orientation of the genome and often apply stringent p-value corrections that likely result in the loss of true positive associations. Here, we present GENIUS (GEnome traNsformatIon and spatial representation of mUltiomicS data), a framework for integrating multi-omics data using deep learning models developed for advanced image analysis. The GENIUS framework is able to transform multi-omics data into images with genes displayed as spatially connected pixels and successfully extract relevant information with respect to the desired output. Here, we demonstrate the utility of GENIUS by applying the framework to multi-omics datasets from the Cancer Genome Atlas. Our results are focused on predicting the development of metastatic cancer from primary tumors, and demonstrate how through model inference, we are able to extract the genes which are driving the model prediction and likely associated with metastatic disease progression. We anticipate our framework to be a starting point and strong proof of concept for multi-omics data transformation and analysis without the need for statistical correction.

eLife assessment

This **valuable** manuscript presents a new approach to transform multi-omics datasets into images and to exploit Deep Learning methods for image analysis of the transformed datasets. As an example, the method is applied to multi-omics datasets on different cancers. While the evidence in this specific case is **solid**, whether the method is working as advertised in other settings is not yet known.

Introduction

The recent advent of Next Generation Sequencing (NGS) has revolutionized research and has been applied extensively to investigate complex biological questions. As the cost of sequencing continues to drop, it has become increasingly common to apply NGS technology to investigate complementary aspects of the biological processes on the same samples, particularly through analysis of DNA to resolve genomic architecture and single nucleotide variants, RNA to investigate gene expression, and methylation to explore gene regulation and chromatin structure. Such multi-omics data provides opportunities to perform integrated analysis, which investigates multiple layers of biological data together. Over the years, this has resulted in the generation of an incredible amount of very rich data derived from the genome itself, either directly or indirectly. The genome is spatially organized, with genes positioned on chromosomes sequentially and accessed by biological processes in blocks based on chromatin organization[1]. However, genome-derived NGS data is usually stored in and analyzed from a tabular format, where the naturally occurring spatial connectivity is lost. Furthermore, while genomic data is rich, the feature space is generally much larger than the number of samples. As the number of features to evaluate in statistical tests increases, the risk of chance associations increases as well. To correct for such multiple hypothesis testing, drastic adjustments of p-values are often applied which ultimately leads to the rejection of all but the most significant results, likely eliminating a large number of weaker but true associations. While this is a significant issue when analyzing a single type of data, the problem is exacerbated with performing multi-omics analysis where different types of data are combined, often in an ad-hoc manner tailored to specific use cases. Importantly, a common theme in multi-omics analytical approaches is that observations are processed individually, thereby discarding potential spatial information that may originate from the organization of genes on individual chromosomes.

Using artificial intelligence methods may help overcome this problem. Over the past decade, the development of artificial intelligence methods, particularly within deep learning architectures, has thoroughly revolutionized several technical fields, such as computer vision, voice recognition, advertising, and finance. Within the medical field, the roll-out of AI-based technologies has been slower, hampered in part by considerable regulatory hurdles that have proven difficult for machine-learning applications where the systems may accurately classify patients or samples by some parameter, but the logical reason behind this is unclear[2]. Nevertheless, AI systems have proven successful in a multitude of medical studies, and in recent years some AI-powered tools have started to move past testing to deployment[3]. A major benefit of deep neural networks is that they can capture nonlinear patterns in the data without necessitating correction for multiple hypothesis testing. Additionally, the use of convolutional layers within the networks has shown to improve performance by decreasing the impact of noise[4,5]. However, the problem with complex deep learning models is not the analysis itself but their interpretation[6]. Simpler models tend to have high interpretability; however, they are unable to capture complex nonlinear connections in data. This often leads to the utilization of “black box” models at the cost of interpretability[7,8]. “Black box” models are popular in the artificial intelligence industry,

especially in computer vision applications, where immense progress is being made in technologies such as self-driving cars and computer interpretation of images. However, in many of those applications, the interpretability of models is not as important as in medicine[9,10].

In medicine, the interpretability of models is crucial since there is a need for discovering new biomarkers as well as identifying underlying biological processes[11]. In addition to advancements in artificial intelligence and NGS, a vast amount of research has been conducted to interpret highly complex machine learning models; frameworks such as DeepLIFT[12], Integrated Gradients[13,14], and DeepExplain [12,15,16] were developed in recent years with the purpose of debugging complicated machine learning models[17]. These frameworks enable the usage of deep learning models for integrated multi-omics analysis through their ability to evaluate input attribution in models that are traditionally considered a "black box". In multiomics analysis, this means that it is possible to combine the entirety of the data from multiple data sources into a high-dimensional data structure and process it with deep learning models without losing interpretability. As output, an attribution score can be produced for every input, which may be interpreted as the relative importance of the feature in the model and used for further analysis.

Here, we present a framework for multi-omics analysis based on a convolutional deep learning network to find hidden, non-linear patterns in spatially connected feature-rich multi-layered data. The spatial connection of the data is made by transforming the data into a multi-channel image in such a way that spatial connections between genes are captured and analyzed using convolutional layers. Using spatial connections between the data showed superior performance when compared to non-spatially data transformations. Furthermore, the trained model is combined with Integrated Gradients, which allows us to evaluate the relative contribution of individual features and thus decipher the underlying biology that drives the classification provided by the deep learning models. Integrated Gradients is a non-parametric approach that evaluates the trained model relative to input data and output label, resulting in attribution scores for each input with respect to the output label. In other words, Integrated Gradients represent the integral of gradients with respect to inputs along the path from a given baseline. By using Integrated Gradients, we provide an alternative solution to the problem posed by performing multiple independent statistical tests. Here, instead of performing multiple tests, a single analysis is performed by transforming multiomics data into genome images, training a model, and inspecting it with Integrated Gradients. Integrated Gradients will output an attribution score for every gene included in the genome image. These can be ranked in order to retrieve a subset of the most associated genes relative to the output variable. We named the framework GENIUS (GEnome traNsformatIon and spatial representation of mUltiomicS data), and the methodology may be split into two parts, classification and interpretation. First, the key feature of GENIUS is that for classification, multi-omics data is transformed into multi-channel images where each gene is presented as a pixel in an image that covers the whole genome (**Figure 1A-B**). We then incorporate multiple types of omics data, such as mutation, expression, methylation, and copy-number data, into the image as distinct layers. These layers are then used as input into the deep learning model for training against a binary or continuous outcome variable. Next, for interpretation, an attribution score is assigned to each feature using integrated gradients, allowing the user to extract information about which feature or features may drive a specific prediction based on deep learning analysis of input from multiple-omics data sources. In this work, we describe the development of the GENIUS framework and demonstrate its utility in predicting the development of metastatic cancer, patient age, chromosomal instability, cancer type, and as proof of concept, loss of TP53.

All predictions are based on multi-omics input through the GENIUS framework. Users may train their own or publicly sourced multi-omics data against a specified endpoint tailored to the user's choice. The GENIUS framework thus overcomes the issue of multiple hypothesis testing and may

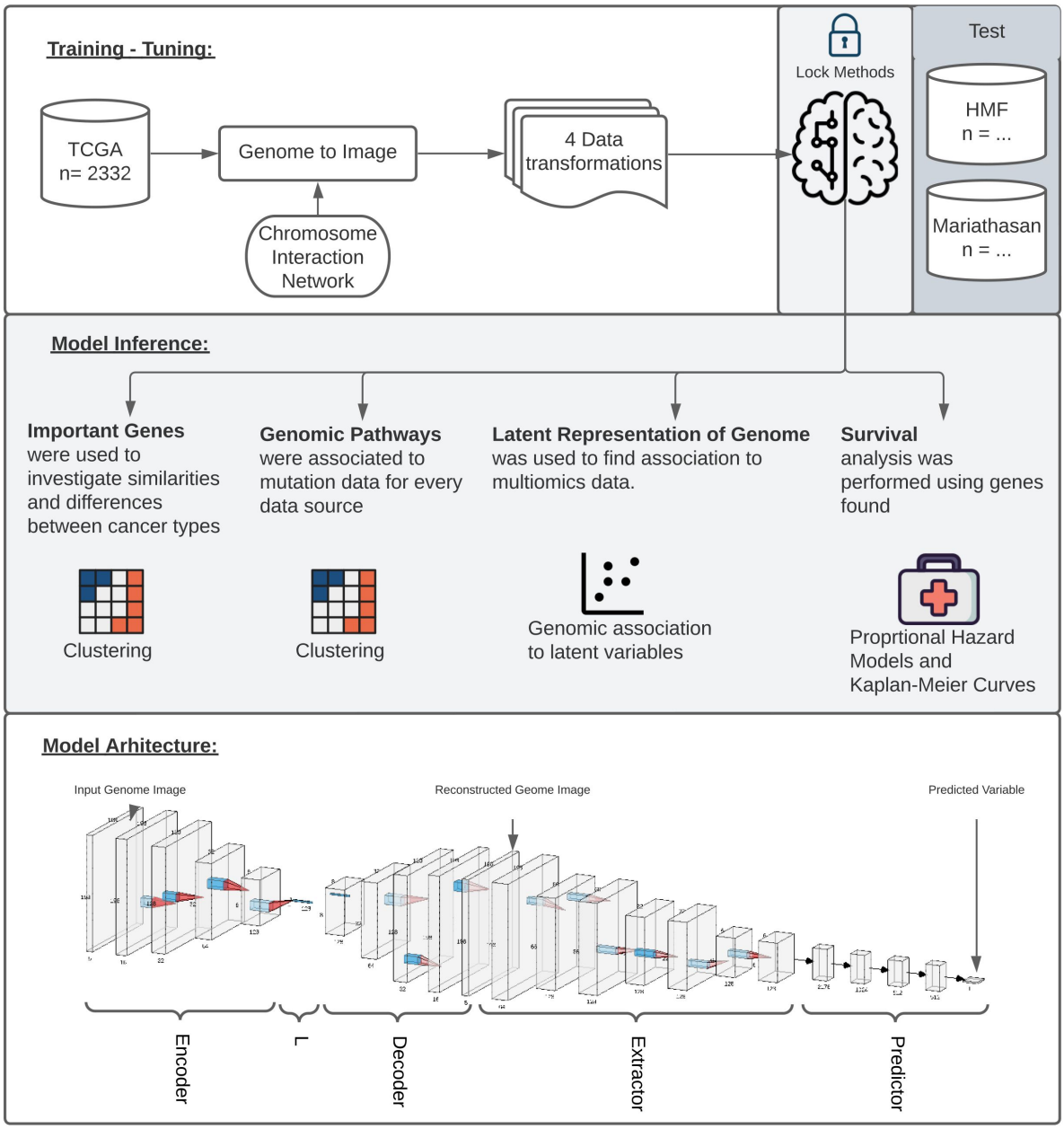


Figure 1.

Study overview.

(A) The study utilized 2332 tumor samples representing six cancer types (bladder, uterine, stomach, ovarian, kidney and colon) and transformed multiomics data into images based on chromosome interaction networks. After the model was trained, we validated found genes with two independent cohorts representing early stage BLCA (UROMOL) and late stage BLCA (Mariathasan). (B) The validation included looking at the most important genes driving metastatic disease, similar/different methylation patterns between cancer types, latent representation of genome data and looking at survival data. (C) The model architecture where the first part of the network encodes genome data into latent vector, L, followed by decoding where image is reconstructed. Next layers aim to extract information from the reconstructed image, concat it with L and make a final prediction.

provide new insights into the biology behind classification by deep learning models. The GENIUS framework is made available as a GitHub repository and may be used without restrictions to develop stratification models and inform about genome biology using multi-omics input.

Methods

GENIUS Model Architecture and Hyperparameters

We designed a four-part convolutional neural network with the purpose of extracting the features from multi-dimensional data while minimizing the impact of noise in the data (**Figure 1C**). The network was implemented using the PyTorch framework. The structure of the network is similar to an autoencoder architecture; however, the reconstruction of the genome image is not penalized. The motivation behind the implemented network structure is to use an encoder in order to learn how to compact genomic information into a small vector, L, forcing the network to extract relevant information from up to five data sources. The next module reconstructs the image from vector L, learns which features are important, reorganizes them optimally, and removes noise. The final module of the network uses a series of convolutions and max pooling layers in order to extract information from the reconstructed image and, finally, predicts the outcome variable using a fully connected dense network.

The first part of the network is called the encoder, as its purpose is to encode the entire image to a vector of size 128, representing the latent representation of the input data, "L". Next, the original image is reconstructed from L into its original size using a decoder module in the network. In this step, since we are not using the reconstruction loss, the network reconstructs the image of a genome which is optimal for information extraction. This is followed by the extractor module containing convolution and max-pooling layers aiming to extract relevant information from the reconstructed image. The final part of the network flattens the learned features obtained from previous layers, concatenates them with the L vector, and forwards it to a fully connected dense feed-forward network where the final prediction is made (**Figure 1C**) [18]. During training, the last module of this model was adopted to predict qualitative as well as quantitative types of data.

All models were trained with Adagrad optimizer with the following hyperparameters: starting learning rate = 9.9e-05 (including learning rate scheduler and early stopping), learning rate decay and weight decay = 1e-6, batch size = 256, except for memory-intensive chromosome images where the batch size of 240 was used. Adding chromosome interaction information to the data transformation showed improvement during training; Next question was whether we should penalize the reconstruction of genome image during the training process. After multiple training scenarios and hyperparameter exploration, we concluded that by forcing the network to reconstruct genome images in the process of learning, we are limiting network performance. Instead, we used the appropriate loss function for prediction and allowed the network to reconstruct genome images that are optimal for making predictions.

Evaluating input image design

To evaluate the performance of GENIUS with an image-based transformation of input omics data, we tested four different image layouts of the genome. For each layout, we created a set of images where each sample is represented by one multichannel image and each channel represented a specific type of omics data (gene expression, methylation, mutation, deletion, amplification) (Supplementary Figure 2A). Each data type was encoded for each gene as a continuous value, where each gene was defined by a single pixel in each layer. We then tested the performance of the deep neural network on four different image layouts. First, we assembled the genome as a square image, measuring 198 x 198 pixels in total. Here, all genes were placed on the image sequentially according to their chromosomal locations, and individual chromosomes were

organized by how close they were oriented in 3D space[19]. Second, we tested an image organized by 24 x 3760 pixels, with 3760 pixels representing the most gene-rich chromosome, and each chromosome placed below the other on the image following the same order as in 198x198 images. Chromosomes containing fewer than 3760 genes had black pixels added to the end to create a rectangular image. Third, we tested a random 2D location, with each gene placed as a random pixel in a 198x198 pixel square image. Lastly, we tested an image of a single vector with all genes placed in a randomly ordered sequence. Data transformation we performed and tested:

1. Square image (198 x 198 pixels), each gene represented by one pixel ordered by chromosome position. Chromosomes are ordered by interaction coefficient based on Hi-C sequencing [19].
2. Square image (198 x 198 pixels), each gene is represented by one pixel located on the image in random order; thus, the 2D location carries no information.
3. Rectangular image (24 x 3760 pixels), each gene represented by one pixel ordered by chromosome position. Chromosomes are ordered by interaction coefficient based on Hi-C sequencing [19]
4. A flat, one-dimensional vector containing all features from the five data sources in random order.

By using different image layouts, we wanted to investigate the spatial dependency of observations. Images were created by making a matrix for each source of data where each cell was represented by a single gene (**Figure 1A**, Supplementary Figure 2A-B). The genes in 198x 198 and 24 x 3760 images were ordered by position as well as by chromosome interaction coefficients resulting in the following order of chromosomes: 4, X, 7, 2, 5, 6, 13, 3, 8, 9, 18, 12, 1, 10, 11, 14, 22, 19, 17, 20, 16, 15, 21. Finally, newly created observations for each data source were merged as a multi-channel image where each channel represents a single source of data (Supplementary Figure 2A).

Samples & Training Data

We obtained gene expression, exome mutation, methylation and copy number data from six cancer types from the Cancer Genome Atlas (TCGA). These were picked to filter out cancer types with less than 400 samples. Next, cancer types with an extremely high proportion of metastatic samples (0.85 < Proportion > 0.15) were removed, resulting in ovarian serous cystadenocarcinoma (OV), colon adenocarcinoma (COAD), uterine corpus Endometrial carcinoma (UCEC), kidney renal clear cell carcinoma (KIRC), urothelial bladder carcinoma (BLCA) and stomach adenocarcinoma (STAD) (Supplementary Figure 1). RNAseq was obtained from the UCSC Toil pipeline [20] and summarized to transcript per million (TPM) on the gene level. SNP6 copy number data were segmented using ASCAT v2.4[21,22] and converted into a ploidy and purity normalized logR value by dividing the total copy number with ploidy and taking the log2 value. The weighted genome integrity index (wGII)[23] was calculated on the available segmented copy number data, as previously described. Mutation calls were annotated using Polyphen2 to assess the mutation's impact on the protein structure. Methylation was summarized by the mean methylation score for each gene.

Validation cohorts acquisition and processing

Two independent cohorts of bladder cancer patients were used for validation. The UROMOL cohort [24,25] contains molecular data from 535 tumors from patients with early-stage bladder cancer (Ta and T1) and was included to evaluate the progression to muscle-invasive bladder cancer. The Mariathasan cohort[26] contains molecular data from 348 tumors from patients with advanced or metastatic bladder cancer (stage III and IV), treated with checkpoint immunotherapy. This cohort was included to evaluate the ability of the GENIUS framework to predict the likelihood of developing metastatic disease.

For both cohorts, RNAseq data was aligned against hg38 using STAR[27] version 2.7.2 and processed to generate count and transcript per million (TPM) expression values with Kallisto [28] version 0.46.2. Whole exome sequence (WES) data was processed using GATK [29] version 4.1.5 and ASCAT version 2.4.2 to obtain mutation and allele-specific copy number, purity, and ploidy estimates.

Data transformation

All mutations were ranked by PolyPhen scores, ranging between 0 and 1. LogR segmented copy number data was analyzed as deletion and amplification separately. Copy number deletion was defined as logR scores < log2 of 0.5/2, copy number amplification was defined as logR scores > log2 of 5/2. All data types were defined on the gene level. For copy number alterations, we defined genes as amplified if the entirety of the gene was found within the amplified DNA segment. Genes were defined as deleted if they were partially within the deleted DNA segment (**Figure 1A**, Supplementary Figure 2A-B). Finally, to enable data integration and for more stable training of machine learning models, we generated mathematically equivalent values for each data source ranging from 0 to 1 through a simple linear transformation (min-max scaling). This enabled comparisons between individual data types, and was performed on each data source.

Training scenarios

We used the GENIUS framework to make six models predicting the following conditions:

1. Metastatic cancer (binary classification), defined as Stage IV vs Stages I, II and III.
2. TP53 mutation (binary classification), where the TP53 mutation was removed from the input data and used only as a binary outcome label.
3. The tissue of origin (multi-class classification).
4. Age (continuous variable).
5. Weighted genome integrity index (wGII)[23], a chromosomal instability marker (continuous variable).
6. Randomized tissue of origin (multi-class variable). By randomizing the tissue of origin labels, a negative control was created. The purpose of this negative control was to confirm the model would fail to predict a pattern when none existed.

In order to adapt the network for predicting different variables, we simply changed the output layer and loss function for training. Binary classifications and the multi-class classification used softmax as the output layer and the cross entropy loss function. When predicting continuous values, the model used the output from the activation function with the mean squared error loss function. When predicting multi-class labels, the performance measure was defined by the F1 score, a standard measure for multiclass classification that combines the sensitivity and specificity scores and is defined as the harmonic mean of its precision and recall. To evaluate model performance against the binary outcome, ROC analysis was performed, and the area under the curve (AUC) was used as the performance metric.

Latent representation of genome

The purpose of latent vectors is to capture the most significant information from the entire genome data and compress it into a vector of size 128. This vector was later appended into a feed-forward network when making the final prediction. This way, the model had access to extracted information before and after image reconstruction. After successful model training, we extracted the latent representations of each genome and performed the Uniform Manifold Approximation and Projection (UMAP) of the data for the purpose of visual inspection of a model. The UMAP

projected latent representations into two dimensions which could then be visualized. In order to avoid modeling noise, this step was used to inspect if the model is distinguishing between variables of interest. We observed that all training scenarios successfully utilized genome images to make predictions with the exception of Age, where no pattern was found from the genomic data, and randomized cancer type, which served as negative control where no pattern was expected (**Figure 2B**). Information in latent vectors extracted from Age-Model and randomized cancer type showed no obvious patterns, which is likely the cause of poor performance.

Identifying genes relevant to the tested outcome

Once the model was trained on the data, the appropriate loss function and output layer, including the model weights, were stored in a .pb file. The model and final weights were analyzed using the Integrated Gradients method (IG) implemented by Capture [14]. IG is an attribution method that assigns an "attribution score" to each feature of the input data based on predictions the model makes. The attribution score is calculated based on the gradient associated with each feature of each image channel with respect to the output of the model. This information indicates to the neural network the extent of weight decrease or increases needed for certain features during the backpropagation process. Next, the created attribution images are used to extract information for each image channel and for every pixel. Since the pixels represent individual genes, this information can be reformatted and filtered to show the most important genes from every data source included in the analysis. All attribution scores were scaled using a min-max scaler for every cancer type to address biological differences between cancer types.

Code Availability

All code is available on the public GitHub repository (*https://github.com/mxs3203/GENIUS*), where the framework is easily available for analysis of private or public data. The framework provides tools to transform gene-oriented data into an image, train a model using existing model architecture and infer the most informative genes from the model using integrated gradients. The GitHub repository contains example data and instructions on how to use the GENIUS framework.

Computational Requirements

In order to train the model, we used the following hardware configuration: Nvidia RTX3090 GPU, AMD Ryzen 9 5950X 16 core CPU, and 32Gb of RAM memory. In our study, we used a batch size of 256, which occupied around 60% of GPU memory. Training of the model was dependent on the output variable. For metastatic disease prediction, we trained the model for approximately 4 hours. This could be changed since we used early stopping in order to prevent overfitting. By reducing the batch size to smaller numbers, the technical requirements are reduced making it possible to run GENIUS on most modern laptops.

Results

Building genome images to utilize spatial connections in genomic data

We endeavored to present genomic data as an image with genes represented as individual pixels to be processed by our deep-learning architecture. To evaluate the relevance of the spatial orientation of the genes relative to model performance, we tested four different image layouts (Methods, **Figure 2A**): (1) Square image (198 x 198 pixels), each gene represented by one pixel ordered by chromosome position. Chromosomes are ordered by interaction coefficient based on Hi-C sequencing [9]. (2) Square image (198 x 198 pixels), each gene is represented by one pixel located on the image in random order; thus, the 2D location carries no information. (3)

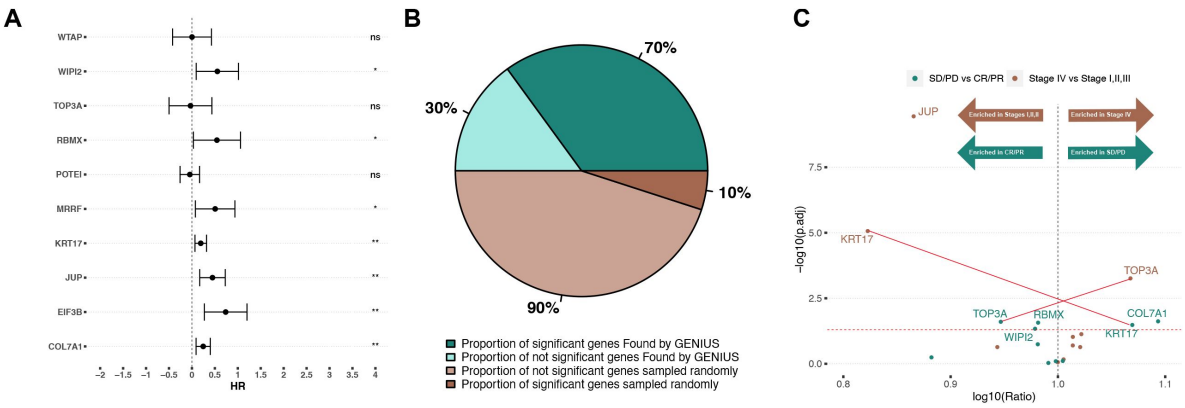


Figure 2.

Data transformation overview.

(A) The multiomics genome data was transformed into 4 image types; square image organized by chromosome interaction network, chromosome image organized by chromosome interaction network, randomly organized image and flat vector containing all multiomics data. (B) The x-axis represents epochs and the y-axis represents AUC score of fixed 25% data we used for accuracy assessment within the TCGA cohort. All four image types were used in training for metastatic disease prediction and the square image organized by chromosome interaction network resulted in best model performance (green color). The red line shows where the model resulted in the best loss. All curves stopped when the loss started increasing, indicating overfitting. The bar plot shows the proportion of correctly predicted (metastatic disease) in every cancer type included in the study. (C) 2-dimensional representation of vector L using UMAP for each predicted variable. Colors indicate the output variable which was used in the specific run.

Chromosome image (24 x 3760 pixels), each gene represented by one pixel ordered by chromosome position. Chromosomes are ordered by interaction coefficient based on Hi-C sequencing [9]. (4) A flat, one-dimensional vector containing all features from the five data sources in random order. To evaluate the image layout, we used each type of layout to train against six biological states: (1) metastatic disease (stage IV vs. I-III), (2) cancer type, (3) burden of copy number alterations (defined by the weighted genome instability index, wGII), (4) patient age, (5) TP53 status (where the TP53 pixel was set to "0" for all samples), and (6) randomized tissue type (negative control). Every model output variable was trained until we observed no change in loss function or until validation loss values started increasing, indicating overfitting, which we handled by implementing early stopping.

While predicting metastatic disease, we observed that the Square Image data transformation outperformed all other data transformations, reaching a validation AUC of 0.87. The chromosome image and shuffled squared image performed similarly with AUC of 0.72 and 0.70, respectively. Interestingly, the flat vector of features scored validation AUC around 0.84; however, the loss function started increasing as training epochs increased, indicating that the model was overfitted (**Figure 2B**, Supplementary Figure 4 C-D). In the second scenario, we tested multiclass prediction using six cancer types in our dataset. Square Image outperformed other image layouts, reaching an F1 score of 0.81. Chromosome Image followed with an F1 score of 0.74, and the flat vector of features performed similarly to the random square image, reaching F1 scores of 0.66 and 0.71, respectively (Supplementary Figure 4 E-F). In order to address the framework's capabilities for predicting numeric output variables, we used wGII and patient age. Predicting wGII showed that the flat vector of features reached the least favorable RMSE score of 0.22, where chromosome image, shuffled square image, and square image reached similar RMSE scores of 0.16, 0.15, and 0.14, respectively (Supplementary Figure 5 C-D).

These results suggest that data layout does not play a major role when predicting wGII as the number of events in the genome would be predictive regardless of location. The age prediction model using square image data transformation outperformed other data transformations and obtained a validation RMSE of 0.19. The shuffled image performed the worst, reaching an RMSE of 0.49, while the chromosome-organized image and flat vector of features scored similar RMSE values of 0.38 and 0.31, respectively. Additionally, the flat vector was inconsistent during training, but it did outperform the chromosome image (Supplementary Figure 5 A-B). In the fifth scenario, we predicted the TP53 mutation status but removed the TP53 mutation itself from the data. Square Image performed the best, reaching a validation AUC of 0.83, whereas no major difference could be observed between a flat vector of features and Chromosome Image, reaching a validation AUC of 0.75 (Supplementary Figure 4 A-B). Finally, we tested the framework by predicting randomized cancer types as the negative control. All data transformations had similar and poor results (Supplementary Figure 6). For each output variable, we trained four different models utilizing the four data transformations. In all cases, the square image (198x198, ordered by chromosomes) outperformed the other transformations and was chosen as the layout for the final GENIUS framework, which was used for all subsequent analyses.

Latent representation of genome captures relevant biology

The model architecture contains an encoder and decoder connected by a latent vector of size 128 (L), which provides the opportunity to inspect model performance (**Figure 1**). The L vector is considered the latent representation of the genome data because it extracts and captures the most relevant data with respect to the output variable. This implies that an optimally trained model would show a perfect latent representation of the genome when overlaid with the output variable. Furthermore, this vector was later appended into a feed-forward network when making the final prediction. This way, the model had access to extracted information before and after image reconstruction. In order to visually inspect patterns captured by the model, we extracted the latent representations of each genome and performed the Uniform Manifold Approximation and Projection (UMAP) of the data to project it into two dimensions. We observed that all training

scenarios successfully utilized genome images to make predictions that clustered into distinct groups, with the exception of Age. As expected, randomized cancer type, which served as negative control, also performed poorly (**Figure 2C**). Information in latent vectors extracted from Age-Model and randomized cancer type-model showed no obvious patterns, which is likely the cause of poor performance.

GENIUS classification identifies tumors likely to become metastatic

To explore the utility of the GENIUS framework to classify tumors from multiomics data and to interpret the biological drivers behind the classification, we further investigated the GENIUS model trained against metastatic disease using the TCGA datasets (**Figure 2B**). This analysis included primary tumors from six cancer types, a total of 2307 tumors, with 53 percent progressing to metastatic disease (bladder cancer [BLCA, 277 metastatic/133 not-metastatic], ovarian cancer [OV, 535 metastatic/47 not-metastatic], colon adenocarcinoma [COAD, 196 metastatic/254 not-metastatic], stomach adenocarcinoma [STAD, 230 metastatic/189 not-metastatic], kidney clear cell renal cancer [KIRC, [208 metastatic/326 not-metastatic], uterine endometrial cancer [UCEC, 117 metastatic/394 not-metastatic]). The omics data types included somatic mutations, gene expression, methylation, copy number gain and copy number loss. Using holdout type cross-validation, where we split the data into training (75%) and validation (25%), we observed a generally high performance of GENIUS, with a validation AUC of 0.83 for predicting metastatic disease (**Figure 2B**). The GENIUS framework allows us to explore the attribution of individual data layers to the final prediction. Across the cohort, gene expression and methylation data were generally the most informative data layers when it comes to classifying metastatic disease (**Figure 3A**). We noted that expression and methylation overall ranked the highest in terms of mean scaled attribution, with the exception of OV, which showed enrichment in methylation followed by copy number gain and loss. The same analysis was performed for cancer type, wGII, patient age, TP53 status and randomized tissue type, (Supplementary Figures 7 - 8).

Interpreting the GENIUS model classifying metastatic cancer biology

Analyzing raw attribution scores we concluded the most informative data type overall regarding the development of metastatic disease was methylation (**Figure 3A**). To identify the individual genes driving the prediction, we pulled the 100 genes with the highest methylation attribution according to the GENIUS classification. We observed that many methylated regions overlapped between the six cancer types. These regions included methylation on specific regions of chromosomes 1, 6, 11, 17, and 19 (Supplementary Figure 9). Additionally, OV showed a unique methylation pattern spanning most of chromosome 7, while KIRC, COAD, and BLCA displayed regions of overlapping methylation on chromosome 22. We also noticed that mutation data often had a single mutation with a large attribution score while expression and methylation showed multiple genes with high attribution scores. To determine the genes that overall across the multi-omics data analysis contributed the most to the GENIUS classification of metastatic disease, we normalized gene attribution by cancer type and compared the top 50 genes for each cancer type (total of 152 genes, **Figure 3B**, Supplementary Table 1). Unsurprisingly, we observed that TP53 mutations held the highest attribution score, followed by mutations to VHL. Both of these genes are well-established drivers of cancer and were previously reported as enriched in metastatic cancer [30,31], likely representing a more aggressive disease. However, of the 152 top genes, we noted only 11 genes previously reported as either oncogenes or tumor suppressor genes in the COSMIC cancer gene census (**Figure 3B**, indicated with a star), leaving 141/152 as potentially novel cancer genes. The highest-scoring gene not previously associated with cancer was SLC3A1, the expression of which was found to be strongly associated with metastatic disease in clear-cell renal cancer. SLC3A1 gene is a protein-coding gene associated with the transportation of amino

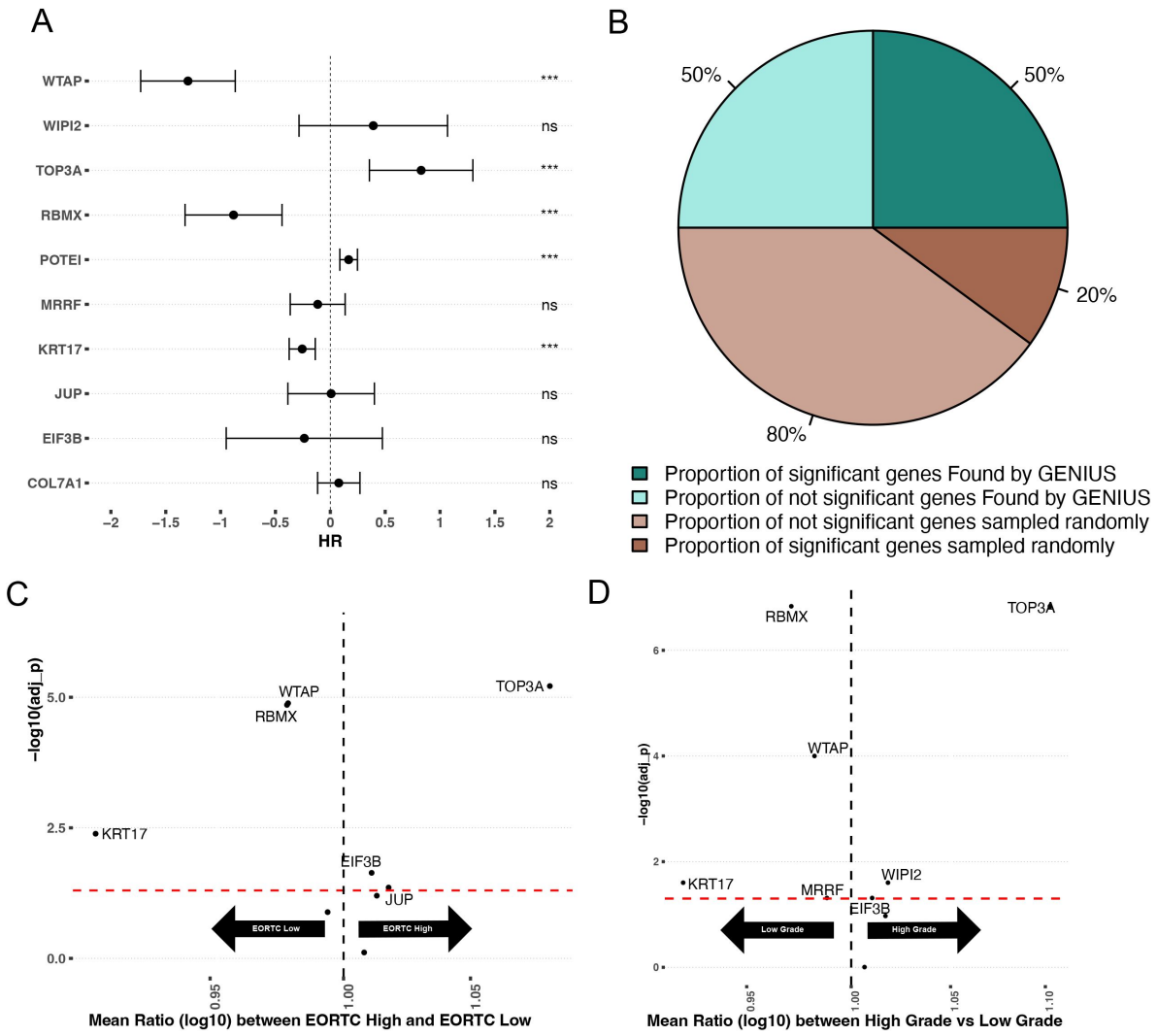


Figure 3.

The most important events in metastatic disease development.

(A) Pieplot showing the relative importance of each data source when predicting metastatic disease for each cancer type included in the study. (B) Top 50 genes for every cancer type scale by cancer type. The star symbol below the gene names indicates that the gene is part of COSMIC gene consensus. The color of the gene name indicates the data source and color of the bar indicates the cancer type.

acids in the renal tubule and intestinal tract, and aberrations in this gene have been associated with cystinuria, a metabolic disorder of the kidneys, bladder, and ureter [32,33]. Furthermore, we identified PLVAP, often involved in MAPK cascades as well as in cellular regulatory pathways and the tumor necrosis factor-mediated signaling pathway. In BLCA, one of our most significant findings was increased expression of KRT17, a gene associated with a cytoskeletal signaling pathway, glucocorticoid receptor regulatory network, and MHC class II receptor activity [34,35]. KRT17 has previously been reported as a potential cancer gene, but with an uncertain role[36]. Across cancer types, TOP3A was found to be commonly methylated in BLCA, COAD, STAD and UCEC. TOP3A is associated with homology-directed repair and methylation may lead to increased chromosomal instability, a hallmark of cancer[37]. The top ten most important events driving the prediction of every output variable included in the study are summarized in Supplementary Table 2-3.

Validation of bladder cancer metastasis-associated genes in an independent cohort of advanced and metastatic bladder cancer

To investigate if the genes with the highest attribution score in the TCGA bladder cancer analysis were indeed associated with metastatic bladder cancer, we utilized an immunotherapy-treated predominantly late-stage (mainly stage III and IV) bladder cancer cohort with gene expression data available for 348 tumors[26]. For this analysis, we considered only the methylation and gene-expression-associated genes from the TCGA analysis. For methylation, we restricted the analysis to genes showing a significantly negative correlation between gene expression and gene-specific methylation levels (Supplementary Figure 10). We then combined the methylation and gene-expression-based attribution scores and took the top 10 genes: RBMX, COL7A1, KRT17, JUP, WIPI2, TOP3A, EIF3B, WTAP, POTEI, and MRRF. Next, we implemented 10 multivariate Cox proportional hazard models (one for each gene), including available clinical parameters such as tumor stage, gender, neoantigen burden and baseline performance status (Supplementary Table 4). This showed that in multivariate analysis, 7/10 genes had a significant association with outcome (**Figure 4A**). To evaluate the results of this analysis, we compared it to an identical model run 1,000 times, but where the 10 genes were randomly picked. In 1,000 runs, not one returned at least 7 significant genes ($P < 0.001$) (Supplementary Figure 11A). The median percentage of significant genes for each run is reported in **Figure 4B**. Next, we performed two independent analyses, comparing the expression values of the top 10 genes between either (1) tumors defined as stage IV versus stage I-III, and (2) patients that responded to immunotherapy (CR and PR) versus patients that did not respond to immunotherapy (SD and PD). Following correction for multiple hypothesis testing, we observed that TOP3A showed significantly increased expression in stage IV tumors, while JUP and KRT17 were significantly increased in stage I-III tumors (**Figure 4C**, brown dots). When comparing gene expression to response to immunotherapy, TOP3A, RBMX, and WIPI2 were significantly more expressed in CR/PR while KRT17 and COL7A1 were significantly more expressed in SD/PD. Interestingly, we observed increased expression of TOP3A in stage IV tumors, suggesting a role in metastatic disease, yet we also observed that the same gene was more expressed in tumors that responded to immunotherapy. This suggests that TOP3A is associated with the development of metastatic disease, but its expression may result in the development of a bladder cancer phenotype that is more sensitive to checkpoint immunotherapy.

Validation of metastasis-associated genes in an independent cohort of early-stage bladder cancer

To investigate if the metastasis-associated genes found through the GENIUS framework also plays a role in the development of aggressive features in early stage bladder cancer, we acquired the UROMOL dataset[25], which includes gene expression data from 535 low-stage tumors. We again investigated the top 10 methylated or expressed genes found in the TCGA analysis of BLCA,

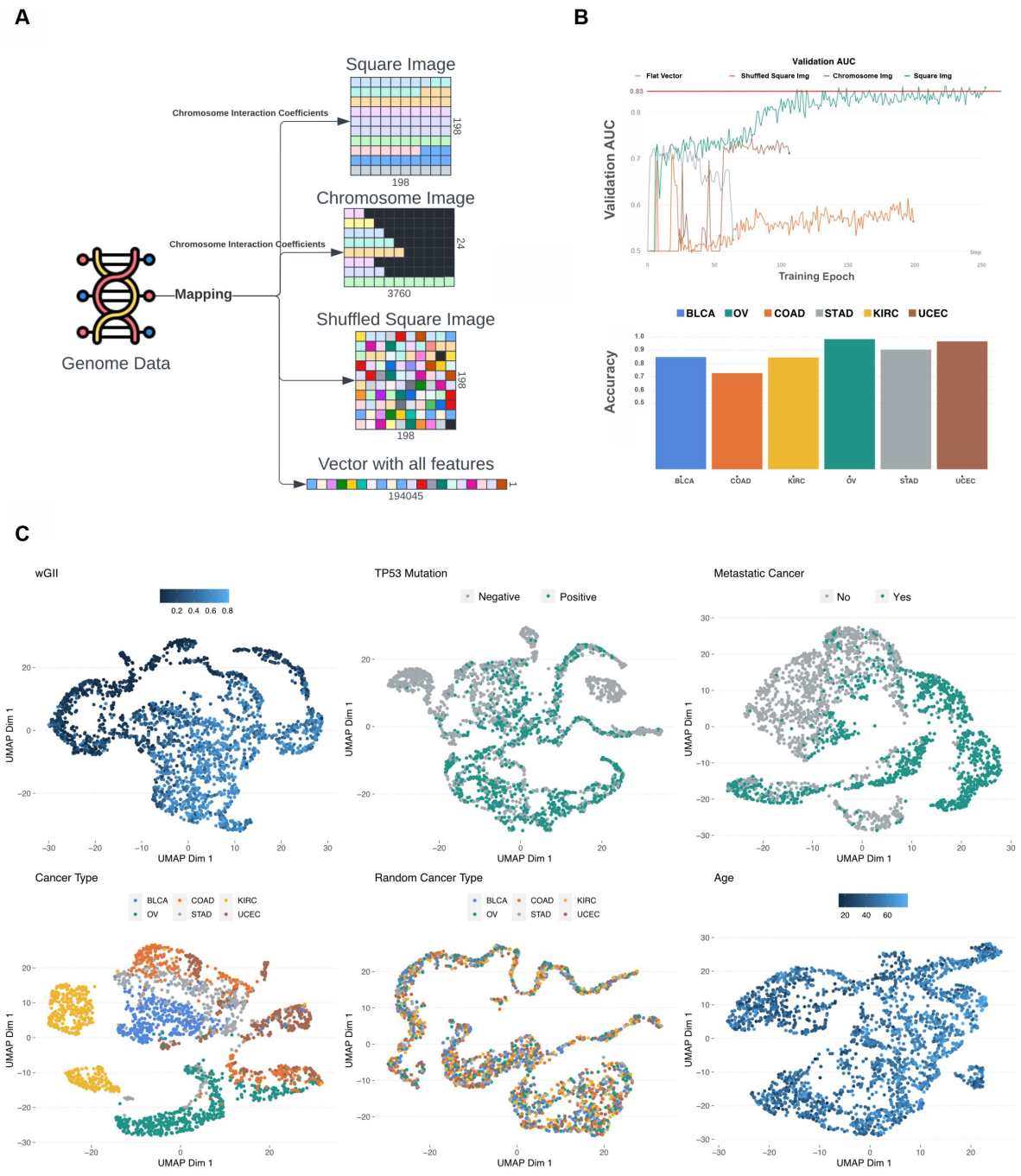


Figure 4.

Validation on late stage immunotherapy treated bladder cancer (Mariathasan).

(A) Forest plot showing top 10 expressed/methylated genes in multivariate cox proportional hazard model. (B) Comparison of median percent of randomly selected genes vs. genes picked by GENIUS in cox proportional hazard model. (C) Volcano plot showing top 10 expressed/methylated genes and their enrichment in two comparisons; Stages I-III vs Stage IV and immunotherapy response (CR: complete response, PR: partial response) versus no response (SD: stable disease, PD: progressive disease). Two genes show association in opposite directions, indicated by red lines (KRT17, associated with low stage and poor immunotherapy response, and TOP3A, associated with high stage and improved immunotherapy response).

using the gene expression data from UROMOL. First, we performed Cox proportional hazard analysis with progression-free survival (PFS) using the top 10 genes found by the GENIUS framework, again creating 10 individual models containing the selected genes and available clinical factors such as age, tumor stage, and sex. This showed that in multivariate analysis, 5/10 genes had a significant association with outcome (**Figure 5A**). The results were compared with cox proportional hazard models utilizing random sets of ten genes, repeated 1,000 times. Of these, 216 runs showed at least five significant genes (P = 0.216) (Supplementary Figure 11B), indicating that in early stage bladder cancer, the genes found by GENIUS to be associated with cancer metastasis were not uniquely relevant for disease progression. However, when we computed the median percentage of significant genes and compared it to the top 10 genes picked by the GENIUS framework, by random chance, only 20% of genes overall were found to be significantly associated with PFS compared to 50% of GENIUS genes (**Figure 5B**). To further investigate the top 10 genes picked by GENIUS, we compared the mean expression of each gene between different clinical risk groups (EORTC status [38]) and tumor grade. In this analysis, six of the 10 genes were significantly associated with EORTC status (**Figure 5C**, Supplementary Table 4), and seven with grade (**Figure 5D**, Supplementary Table 5).

Discussion

In this work, we explored multiple options on how to transform multi-omics data into an image, leading to the utilization of deep learning models, which are often described as “black box” models. The model architecture was evaluated in six different training scenarios, with a focus on validating the prediction of metastatic cancer. In this process, we also evaluated four different image layouts, concluding that of these, projecting the genome into a 198x198 square image with genes organized based on chromosome interaction[19] performed the best. While that spatial organization improved the prediction, we recognize that there may exist a more optimal representation of multi-omics data which should be explored further in future work. Potential methods for organizing gene orientation in a multi-channel image could consider integrating topologically associating domains[39] along with the spatial information from HiC. With the current implementation of GENIUS, gene layout can be set manually by the user to explore this issue further. For GENIUS, we have also included an auto-encoder in the network to recreate the input information without reconstruction loss. In this manner, the model itself can reconstruct the image of a genome in a format that is optimal for the prediction it is trying to make. The model also produces a latent representation of multi-omics data in a shape of a vector of a size 128 (L), which is later concatenated in a model when making final predictions. In order to investigate training effectiveness, we performed a UMAP clustering analysis of the L vector, where we compared the 2D representation of L with the variables of interest (**Figure 2C**). It is clear from this analysis that the L vector itself holds information that may be particularly relevant for multi-class prediction, but further analysis is needed to decipher what information is encoded in the L vector.

The main purpose behind the study was to demonstrate the feasibility of leveraging the power of deep learning techniques optimized for image analysis to interpret genome-derived multi-omics data. A key element of this approach includes the transformation of genomic data into images with genes arranged as pixels organized by chromosomal location. Beyond the readout from multi-omics data, this approach provides spatial information to the deep-learning framework, which significantly improves the performance of the models (**Figure 2B**). To the best of our knowledge, we are the first to demonstrate the utility of spatial information and to provide a ready-to-use framework that incorporates spatial information and deep learning for the analysis of genome-derived multi-omics data. Furthermore, within the GENIUS framework, we facilitate the interpretation of the trained model in order to explore the biology behind the prediction

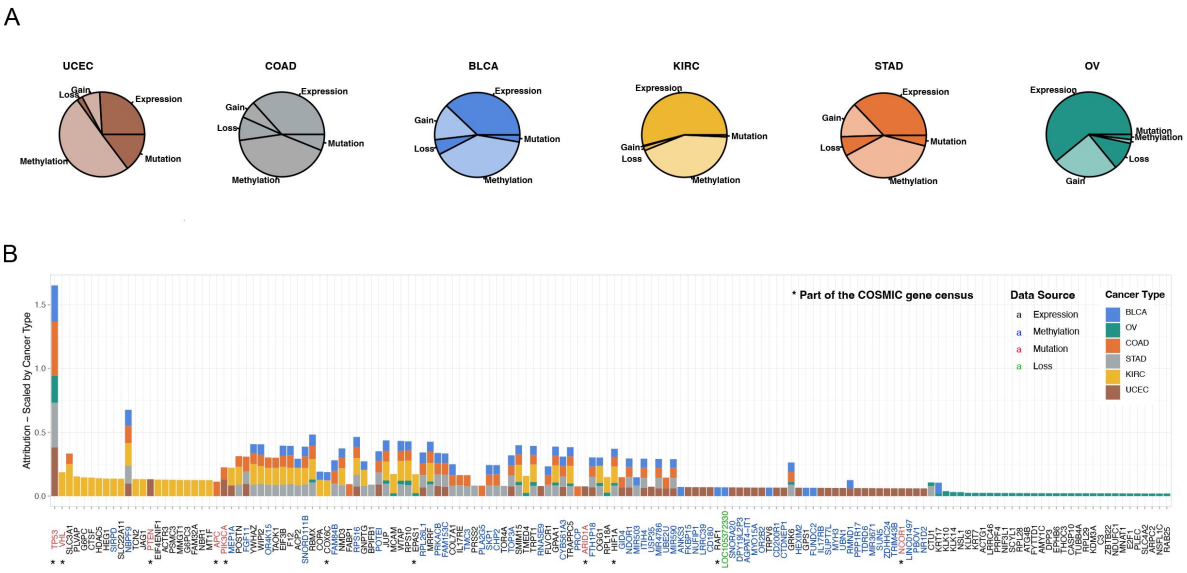


Figure 5.

Validation on early stage bladder cancer (UROMOL).

(A) Forest plot showing top 10 expressed/methylated genes picked by GENIUS for BLCA. (B) Comparison of median percent of randomly selected genes vs. genes picked by GENIUS in cox proportional hazard model. (C) Volcano plot showing association of the top 10 expressed/methylated genes relative to EORTC-Low and EORTC-High groups. (D) Volcano plot showing association of the top 10 expressed/methylated genes relative to low grade and high grade BLCA tumors.

without the need for data preprocessing and multiple hypothesis correction. This was achieved by combining a deep learning network with Integrated Gradients[14], allowing us to infer the attribution score for the input, resulting in non-parametric, ready-to-analyze output.

For every cancer type included in the dataset, we listed the top 10 genes driving metastatic disease and investigated in detail genes associated with BLCA metastasis and aggressiveness. For this, we used two independent cohorts, one representing late-stage and metastatic cancer, and one representing early stage cancer. In both cohorts, we tested if methylation and expression of genes found by the GENIUS framework were associated with survival at higher rates than when compared to randomly picked genes. In the late stage BLCA cohort, seven out of 10 genes were significantly associated with overall survival, while in the early stage BLCA cohort, we found that five out of 10 were significantly associated with progression-free survival. That the results in the early stage bladder cancer cohort (UROMOL) are less significant may relate to the model being trained to predict metastatic cancer. It is likely that the drivers of malignancy are different in early relative to late stage disease, thus the top 10 genes found by GENIUS might not be prognostic in early stage setting. In this regard it is also worth noting that two of the top 10 genes (RBMX and KRT17) were associated with poor outcome in late stage disease, while they were associated with improved outcome in early stage disease. Interestingly, in the late stage bladder cancer cohort, we observed that high expression of TOP3A associated with stage IV disease (**Figure 4C**). However, we also observed that high expression associated with improved response to immunotherapy. It is known that TOP3A has an important role in homology-directed repair and loss may be associated with chromosomal instability, which has shown a positive association with immunotherapy response [40–42], potentially offering a likely explanation for this finding. Similarly, we observed that KRT17 was enriched in stages I-III, suggesting it may be associated with a less aggressive disease type. However, in the immunotherapy-treated cohort, KRT17 is associated with poor response to immunotherapy. In previous studies, KRT17 has been reported as associated with the development of metastatic disease, MHC type II receptor activity and angiogenesis [36,43]. This indicates that the KRT17 gene plays an important role as tumor suppressor gene in early-stage cancer, and that loss may further promote the development of aggressive, metastatic disease. While further research in this field is required to properly assess the utility of the reported genes, this work provides a framework that unlocks powerful machine-learning for more direct analysis of multi-omics data.

Taken together, we provide here the GENIUS framework along with analysis demonstrating the utility in multi-omics analysis. While we have focused on cancer analysis here, we believe GENIUS may find utility in a diverse range of genome-based multi-omics analyses. We have provided a github repository that can be used to transform data into images and train the same model predicting variables of user's interest and inferring the importance of input with respect to the desired output.

Acknowledgements

N.J.B. is a fellow of the Lundbeck Foundation (R272-2017-4040), and acknowledges funding from Aarhus University Research Foundation (AUFF-E-2018-7-14), and the Novo Nordisk Foundation (NNF21OC0071483). The results published here are in whole or part based upon data generated by the TCGA Research Network: *https://www.cancer.gov/tcga*.

Gene	Early Stage	Late Stage (Immunotherapy)	Description
TOP3A	HR > 0 (PFS), High Grade, High EORTC	HR = 0 (OS), Enriched in Stage IV, Enriched in CR/PR	Catalyses the transient breaking and rejoining of a single strand of DNA, involved in regulation of recombination and homology directed repair. Positive association to OS in OV [44]
RBMX	HR < 0 (PFS), Low Grade, Low EORTC	HR > 0 (OS), Enriched in CR/PR	Associated with translational control and DNA damage pathways. Reported to be negatively correlated with tumour stage, histological grade and poor patient prognosis in BLCA [45]
POTEI	HR = 0 (PFS)	HR = 0 (OS)	POTE family of proteins is associated with apoptotic cells[46]
KRT17	HR < 0 (PFS) Low Grade, Low EORTC	HR > 0 (OS), Enriched in Stages I-III Enriched in SD/PD	Associated with structural molecule activity and MHC class II receptor activity. Associated with metastasis and angiogenesis in variety of tumour types [36,43]
WIPI2	HR = 0 (PFS), High Grade	HR > 0 (OS), Enriched in CR/PR	Component of the autophagy machinery that controls the major intracellular degradation process. WIPI2 is suggested as a biomarker for predicting colorectal cancer prognosis [46]
MRRF	HR = 0 (PFS) Low Grade	HR > 0 (OS)	Associated with the ribosome recycling factor, which is a component of the mitochondrial translational machinery. High expression is associated with poor outcome in ovarian cancer[45]
EIF3B	HR = 0 (PFS) High EORTC	HR > 0 (OS)	Eukaryotic Translation Initiation Factor 3 Subunit B Is a Promoter associated with Pancreatic Cancer [44]
JUP	HR = 0 (PFS)	HR > 0 (OS) Enriched in Stages I-III	Common junctional plaque protein. Controversial role in different malignancies. Knockdown of JUP in epithelium-like GC cells causes EMT and promotes GC cell migration and invasion [47]
WTAP	HR < 0 (PFS), Low EORTC Low Grade	HR = 0 (OS)	Wilms' tumor 1-associating protein is associated to RNA methylation modifications, which regulate biological processes such as RNA splicing, cell proliferation, cell cycle, and embryonic development [47]
COL7A1	HR = 0 (PFS)	HR > 0 (OS) Enriched in SD/PD	Associated with metabolism of proteins and integrins in angiogenesis. Aberrant gene expression is associated with distinct tumour environment, metastasis and survival in multiple cancer types [48]

Table 1

Summary of BLCA genes in two validation cohorts

References

1. Franke M, Ibrahim DM, Andrey G, Schwarzer W, Heinrich V, Schöpflin R, et al. (2016) **Formation of new chromatin domains determines pathogenicity of genomic duplications** *Nature* **538**:265–269

2. Wiens J, Saria S, Sendak M, Ghassemi M, Liu VX, Doshi-Velez F, et al. (2019) **Do no harm: a roadmap for responsible machine learning for health care** *Nature Medicine* :1337–1340 https://doi.org/10.1038/s41591-019-0548-6

3. Benjamens S, Dhunnoo P, Meskó B. (2020) **The state of artificial intelligence-based FDA-approved medical devices and algorithms: an online database** *NPJ Digit Med* **3**

4. Jang H, McCormack D, Tong F (2021) **Noise-trained deep neural networks effectively predict human vision and its neural responses to challenging images** *PLoS Biol* **19**

5. Du R, Liu W, Fu X, Meng L, Liu Z (2022) **Random noise attenuation via convolutional neural network in seismic datasets** *Alex Eng J* **61**:9901–9909

6. Rudin C (2019) **Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead** *Nat Mach Intell* **1**:206–215

7. Elmarakeby HA, Hwang J, Arafeh R, Crowdis J, Gang S, Liu D, et al. (2021) **Biologically informed deep neural network for prostate cancer discovery** *Nature* **598**:348–352

8. Wolfe JC, Mikheeva LA, Hagras H, Zabet NR (2021) **An explainable artificial intelligence approach for decoding the enhancer histone modifications code and identification of novel enhancers in Drosophila** *Genome Biol* **22**

9. Yang G, Ye Q, Xia J (2022) **Unbox the black-box for the medical explainable AI via multi-modal and multi-centre data fusion: A mini-review, two showcases and beyond** *Inf Fusion* **77**:29–52

10. Petch J, Di S, Nelson W (2022) **Opening the Black Box: The Promise and Limitations of Explainable Machine Learning in Cardiology** *Can J Cardiol* **38**:204–213

11. Picard M, Scott-Boyer M-P, Bodein A, Périn O, Droit A (2021) **Integration strategies of multi-omics data for machine learning analysis** *Comput Struct Biotechnol J* **19**:3735–3746

12. Shrikumar A, Greenside P, Kundaje A (2017) **Learning Important Features Through Propagating Activation Differences** https://doi.org/10.48550/arXiv.1704.02685

13. Ancona M, Ceolini E, Öztireli C, Gross M (2017) **Towards better understanding of gradient-based attribution methods for Deep Neural Networks** https://doi.org/10.48550/arXiv.1711.06104

14. Sundararajan M, Taly A, Yan Q (2017) **Axiomatic Attribution for Deep Networks**

15. Samek W, Montavon G, Vedaldi A, Hansen LK, Müller K-R. (2019) **Explainable AI: Interpreting, Explaining and Visualizing Deep Learning** *Springer Nature*

16. Bach S, Binder A, Montavon G, Klauschen F, Müller K-R, Samek W (2015) **On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation** *PLoS One* **10**

17. Despraz J, Gomez S, Satizábal HF, Peña-Reyes CA (2017) **Towards a Better Understanding of Deep Neural Networks Representations using Deep Generative Networks** https://doi.org/10.5220/0006495102150222

18. LeNail A (2019) **NN-SVG: Publication-Ready Neural Network Architecture Schematics** *Journal of Open Source Software* https://doi.org/10.21105/joss.00747

19. Sarnataro S, Chiariello AM, Esposito A, Prisco A, Nicodemi M (2017) **Structure of the human chromosome interaction network** *PLoS One* **12**

20. Vivian J, Rao AA, Nothaft FA, Ketchum C, Armstrong J, Novak A, et al. (2017) **Toil enables reproducible, open source, big biomedical data analyses** *Nat Biotechnol* **35**:314–316

21. Adzhubei IA, Schmidt S, Peshkin L, Ramensky VE, Gerasimova A, Bork P, et al. (2010) **A method and server for predicting damaging missense mutations** *Nat Methods* **7** https://doi.org/10.1038/nmeth0410-248

22. Raine KM, Van Loo P, Wedge DC, Jones D, Menzies A, Butler AP, et al. (2016) **ascatNgs: Identifying Somatically Acquired Copy-Number Alterations from Whole-Genome Sequencing Data** *Curr Protoc Bioinformatics* **56**

23. Burrell RA, McClelland SE, Endesfelder D, Groth P, Weller M-C, Shaikh N, et al. (2013) **Replication stress links structural and numerical cancer chromosomal instability** *Nature* **494**:492–496

24. Zuiverloon TCM, Beukers W, van der Keur KA, Nieuweboer AJM, Reinert T, Dyrskjot L, et al. (2013) **Combinations of urinary biomarkers for surveillance of patients with incident nonmuscle invasive bladder cancer: the European FP7 UROMOL project** *J Urol* **189**:1945–1951

25. Lindskrog SV, Prip F, Lamy P, Taber A, Groeneveld CS, Birkenkamp-Demtröder K, et al. (2021) **An integrated multi-omics analysis identifies prognostic molecular subtypes of non-muscle-invasive bladder cancer** *Nat Commun* **12**

26. Mariathasan S, Turley SJ, Nickles D, Castiglioni A, Yuen K, Wang Y, et al. (2018) **TGFβ attenuates tumour response to PD-L1 blockade by contributing to exclusion of T cells** *Nature* **554**:544–548

27. Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, et al. (2013) **STAR: ultrafast universal RNA-seq aligner** *Bioinformatics* **29**:15–21

28. Ayers M, Lunceford J, Nebozhyn M, Murphy E, Loboda A, Kaufman DR, et al. (2017) **IFN-γ-related mRNA profile predicts clinical response to PD-1 blockade** *J Clin Invest* **127**:2930–2940

29. Van der Auwera GA, O'Connor BD (2020) **Genomics in the Cloud: Using Docker** *GATK, and WDL in Terra. O'Reilly Media*

30. Pandey R, Johnson N, Cooke L, Johnson B, Chen Y, Pandey M, et al. (2021) **TP53 Mutations as a Driver of Metastasis Signaling in Advanced Cancer Patients** *Cancers* **597** https://doi.org/10.3390/cancers13040597

31. Christensen DS, Ahrenfeldt J, Sokač M, Kisistók J, Thomsen MK, Maretty L, et al. (2022) **Treatment represents a key driver of metastatic cancer evolution** *Cancer Res* https://doi.org/10.1158/0008-5472.CAN-22-0562

32. Jiang Y, Cao Y, Wang Y, Li W, Liu X, Lv Y, et al. (2017) **Cysteine transporter SLC3A1 promotes breast cancer tumorigenesis** *Theranostics* **7**:1036–1046

33. Woodard LE, Welch RC, Veach RA, Beckermann TM, Sha F, Weinman EJ, et al. (2019) **Metabolic consequences of cystinuria** *BMC Nephrol* **20**

34. Wu J, Xu H, Ji H, Zhai B, Zhu J, Gao M, et al. (2021) **Low Expression of Keratin17 is Related to Poor Prognosis in Bladder Cancer** *Onco Targets Ther* **14**:577–587

35. Li C, Su H, Ruan C, Li X (2021) **Keratin 17 knockdown suppressed malignancy and cisplatin tolerance of bladder cancer cells, as well as the activation of AKT and ERK pathway** *Folia Histochem Cytobiol* **59**:40–48

36. Zhang H, Zhang Y, Xia T, Lu L, Luo M, Chen Y, et al. (2022) **The Role of Keratin17 in Human Tumours** *Front Cell Dev Biol* **10**

37. Hanahan D, Weinberg RA (2011) **Hallmarks of cancer: the next generation** *Cell* **144**:646–674

38. European Organisation For Research And Treatment Of Cancer (2017) **European Organisation For Research And Treatment Of Cancer. In: EORTC [Internet]. 17 Jan 2017 [cited 13 Jul 2022]. Available: https://www.eortc.org/** *EORTC [Internet]*

39. Beagan JA, Phillips-Cremins JE (2020) **On the existence and functionality of topologically associating domains** *Nat Genet* **52**:8–16

40. Bakhoum SF, Cantley LC (2018) **The Multifaceted Role of Chromosomal Instability in Cancer and Its Microenvironment** *Cell* **174**:1347–1360

41. Chen M, Linstra R, van Vugt MATM (2022) **Genomic instability, inflammatory signaling and response to cancer immunotherapy** *Biochim Biophys Acta Rev Cancer* **1877**

42. Sokač M, Ahrenfeldt J, Litchfield K, Watkins TBK, Knudsen M, Dyrskjøt L, et al. (2022) **Classifying cGAS-STING Activity Links Chromosomal Instability with Immunotherapy Response in Metastatic Bladder Cancer** *Cancer Research Communications* **2**:762–771

43. Ji R, Ji Y, Ma L, Ge S, Chen J, Wu S, et al. (2021) **Keratin 17 upregulation promotes cell metastasis and angiogenesis in colon adenocarcinoma** *Bioengineered* **12**:12598–12611

44. de Nonneville A, Salas S, Bertucci F, Sobinoff AP, Adélaïde J, Guille A, et al. (2022) **TOP3A amplification and ATRX inactivation are mutually exclusive events in pediatric osteosarcomas using ALT** *EMBO Mol Med* **14**

45. Song Y, He S, Ma X, Zhang M, Zhuang J, Wang G, et al. (2020) **RBMX contributes to hepatocellular carcinoma progression and sorafenib resistance by specifically binding and stabilizing BLACAT1** *Am J Cancer Res* **10**:3644–3665

46. Yu L, Luo Y, Ding X, Tang M, Gao H, Zhang R, et al. (2023) **WIPI2 enhances the vulnerability of colorectal cancer cells to erastin via bioinformatics analysis and experimental verification** *Front Oncol* **13**

47. Chen Y, Yang L, Qin Y, Liu S, Qiao Y, Wan X, et al. (2021) **Effects of differential distributed-JUP on the malignancy of gastric cancer** *J Advert Res* **28**:195–208

48. Oh SE, Oh MY, An JY, Lee JH, Sohn TS, Bae JM, et al. (2021) **Prognostic Value of Highly Expressed Type VII Collagen (COL7A1) in Patients With Gastric Cancer** *Pathol Oncol Res* **27**

Author information

Mateo Sokač
Department of Molecular Medicine, Aarhus University Hospital, Aarhus, Denmark, Department of Clinical Medicine, Aarhus University, Aarhus, Denmark, Bioinformatics Research Center, Aarhus University, Aarhus, Denmark
ORCID iD: 0000-0001-9896-1544

Asbjørn Kjær
Department of Molecular Medicine, Aarhus University Hospital, Aarhus, Denmark, Department of Clinical Medicine, Aarhus University, Aarhus, Denmark, Bioinformatics Research Center, Aarhus University, Aarhus, Denmark

Lars Dyrskjøt
Department of Molecular Medicine, Aarhus University Hospital, Aarhus, Denmark, Department of Clinical Medicine, Aarhus University, Aarhus, Denmark
ORCID iD: 0000-0001-7061-9851

Benjamin Haibe-Kains
Princess Margaret Cancer Centre, University Health Network, Temerty Faculty of Medicine, University of Toronto, Toronto, ON, Canada
ORCID iD: 0000-0002-7684-0079

Hugo J.W.L. Aerts
Artificial Intelligence in Medicine (AIM) Program, Mass General Brigham, Harvard Medical School, Boston, MA, Departments of Radiation Oncology and Radiology, Brigham and Women's Hospital, Dana-Farber Cancer Institute, Harvard Medical School, Boston, MA, Radiology and Nuclear Medicine, CARIM & GROW, Maastricht University, The Netherlands
ORCID iD: 0000-0002-2122-2003

Nicolai J Birkbak
Department of Molecular Medicine, Aarhus University Hospital, Aarhus, Denmark, Department of Clinical Medicine, Aarhus University, Aarhus, Denmark, Bioinformatics Research Center, Aarhus University, Aarhus, Denmark
For correspondence: nbirkbak@clin.au.dk
ORCID iD: 0000-0003-1613-9587

Editors

Reviewing Editor
Detlef Weigel
Max Planck Institute for Biology Tübingen, Germany

Senior Editor
Detlef Weigel
Max Planck Institute for Biology Tübingen, Germany

Reviewer #1 (Public Review):

This study by Sokač et al. entitled "GENIUS: GEnome traNsformatIon and spatial representation of mUltiomicS data" presents an integrative multi-omics approach which maps several genomic data sources onto an image structure on which established deep-learning methods are trained with the purpose of classifying samples by their metastatic disease progression signatures. Using published samples from the Cancer Genome Atlas the authors characterize the classification performance of their method which only seems to yield results when mapped onto one out of four tested image-layouts.

A few remaining issues are unclear to me:

1. While the authors have now extended the documentation of the analysis script they refer to as GENIUS, I assume that the following files are not part of the script anymore, since they still contain hard-coded file paths or hard-coded gene counts:

- https://github.com/mxs3203/GENIUS/blob/master/GenomeImage/make_images_by_chr.py
- https://github.com/mxs3203/GENIUS/blob/master/GenomeImage/randomize_normal_imgs.py
- https://github.com/mxs3203/GENIUS/blob/master/GenomeImage/utils.py

If these files are indeed not part of the script anymore, then I would recommend removing them from the GitHub repo to avoid confusion. If, however, they are still part of the script, the authors failed to remove all hard-coded file paths and the software will fail when users attempt to use their own datasets.

1. The authors leave most of the data formatting to the user when attempting to use datasets other than their own presented for this study:

Script arguments:

- a. clinical_data: Path to CSV file that must contain ID and label column we will use for prediction
- b. ascat_data: Path to output matrix of ASCAT tool. Check the example input for required columns
- c. all_genes_included: Path to the CSV file that contains the order of the genes which will be used to create Genome Image
- d. mutation_data: Path CSV file representing mutation data. This file should contain Polyphen2 score and HugoSymbol
- e. gene_exp_data: Path to the csv file representing gene expression data where columns=sample_ids and there should be a column named "gene" representing the HugoSymbol of the gene
- f. gene_methyl_data: Path to the csv file representing gene methylation data wherecolumns=sample_ids and there should be a column named "gene1" representing the HugoSymbol of the gene

While this suggests that users will have a difficult time adjusting this analysis script to their own data, this issue is exacerbated by the fact that their analysis script has almost no internal checks whether data format standards were met. Thus, the user will be left with cryptic error

messages and will likely give up soon after. I therefore strongly recommend adding internal data format checks and helpful error or warning messages to their script to guide users in the input data adoption process.

Reviewer #2 (Public Review):

In this manuscript, Birkbak and colleagues use a novel approach to transform multi-omics datasets in images and apply Deep Learning methods for image analysis. Interestingly they find that the spatial representation of genes on chromosomes and the order of chromosomes based on 3D contacts leads to best performance. This supports that both 1D proximity and 3D proximity could be important for predicting different phenotypes. I appreciate that the code is made available as a github repository. The authors use their method to investigate different cancers and identify novel genes potentially involved in these cancers. Overall, I found this study important for the field.

In the original submission there were several major points with this manuscript could be grouped in three parts:

1. While the authors have provided validation for their model, it is not always clear that best approaches have been used. This has now been addressed in the revised version of the manuscript.

2. Potential improvement to the method

a. It is very encouraging the use of HiC data, but the authors used a very coarse approach to integrate it (by computing the chromosome order based on interaction score). We know that genes that are located far away on the same chromosome can interact more in 3D space than genes that are relatively close in 1D space. Did the authors consider this aspect? Why not group genes based on them being located in the same TAD? In the revised version of the manuscript, the authors discussed this possibility but did not do any new additional analysis.

b. Authors claim that "given that methylation negatively correlates with gene expression, these were considered together". This is clearly not always the case. See for example https://genomebiology.biomedcentral.com/articles/10.1186/s13059-022-02728-5. In the revised version of the manuscript, the authors addressed fully this comment.

3. Interesting results that were not explained.

a. In Figure 3A methylation seems to be most important omics data, but in 3B, mutations and expression are dominating. The authors need to explain why this is the case. In the revised version of the manuscript, the authors have clarified this.

Author Response

The following is the authors' response to the original reviews.

> ### *Reviewer #1 (Public Review):*
>
> *This study by Sokač et al. entitled "GENIUS: GEnome traNsformatIon and spatial representation of mUltiomicS data" presents an integrative multi-omics approach which maps several genomic data sources onto an image structure on which established deep-learning methods are trained with the purpose of classifying samples by their metastatic disease progression signatures. Using published samples from the Cancer Genome Atlas*

> *the authors characterize the classification performance of their method which only seems to yield results when mapped onto one out of four tested image-layouts.*
>
> *Major recommendations:*
>
> - *In its current form, GENIUS analysis is neither computationally reproducible nor are the presented scripts on GitHub generic enough for varied applications with other data. The GENIUS GitHub repository provides a collection of analysis scripts and not a finished software solution (e.g. command line tool or other user interface) (the presented scripts do not even suffice for a software prototype). In detail, the README on their GitHub repository is largely incomplete and reads analogous to an incomplete and poorly documented analysis script and is far from serving as a manual for a generic software solution (this claim was made in the manuscript).*

We apologize for this oversight, and we have now invested considerable resources into making the documentation more detailed and accurate. We have created a new GitHub repository (https://github.com/mxs3203/GENIUS) that contains a small set of example data and all the necessary scripts to run GENIUS. The README file guides the user through each step of the GENIUS framework but it also contains a bash script that runs all the steps at once. When a user would like to use it on their own data, they need to replace the input data with their data but in the same format as the example input data. This is now fully documented in the README file. All scripts have arguments that can be used to point to custom data. The entire pipeline using example data can be run using run_genius.sh (http://genius.sh/) script. This script will produce CSV files and PNG files inside the ExtractWithIG folder containing attribution scores for every cancer type tested.

> *The authors should invest substantially into adding more details on how data can be retrieved (with example code) from the cited databases and how such data should then be curated alongside the input genome to generically create the "genomic image".*

Data for analysis can be sourced from multiple locations, what we have used in our examples and for development was based on data from the TCGA. It can be retrieved from the official TCGA data hub or through Xena Browser (https://xenabrowser.net/). However, the data formats are generic, and similar data types (mutation, expression, methylation, copy number) can be obtained from multiple sources. We have added example data to demonstrate the layout, and we have a script included that creates the layout from standard mutation, expression, methylation and copy number data formats. We have substantially improved the annotations, including detailed descriptions of the data layout along with examples, and we have, as part of our validation, had an independent person test run the scripts using TCGA example data we provided on the new GitHub page.

> *In addition, when looking at the source code, parameter configurations for training and running various modules of GENIUS were hard-coded into the source code and users would have to manually change them in the source code rather than as command line flags in the software call. Furthermore, file paths to the local machine of the author are hard-coded in the source code, suggesting that images are sourced from a local folder and won't work when other users wish to replicate the analysis with other data. I would strongly recommend building a comprehensive command line tool where parameter and threshold configurations can be generically altered by the user via command line flags.*

Apologies, we have changed the code and removed all hard-coded paths. All paths are now relative to the script using them. Furthermore, we made the config file more visible and

easier to use. The example run can be found on the new github repository we linked in the previous comment.

We also inserted the following text in the manuscript

The GitHub repository contains example data and instructions on how to use the GENIUS framework.

> *A comprehensive manual would need to be provided to ensure that users can easily run GENIUS with other types of input data (since this is the claim of the manuscript). Overall, due to the lack of documentation and hard-coded local-machine folder paths it was impossible to computationally reproduce this study or run GENIUS in general.*

Apologies, we have completely reworked the code base, and extensively annotated the code. We have also made highly detailed step-by-step instructions that should enable any user to run GENIUS on their own or public data.

> - *In the Introduction the authors write: "To correct for such multiple hypothesis testing, drastic adjustments of p-values are often applied which ultimately leads to the rejection of all but the most significant results, likely eliminating a large number of weaker but true associations.". While this is surely true for any method attempting to separate noise from signal, their argument fails to substantiate how their data transformation will solve this issue. Data transformation and projection onto an image for deep-learning processing will only shift the noise-to-signal evaluation process to the postprocessing steps and won't "magically" solve it during training.*

The data transformation does not solve the problem of multiple hypothesis testing but it facilitates the use of computer vision algorithms and frameworks on rich multi-omics data. Importantly, transforming the data into genome images, training the model, and inspecting it with integrated gradients can be interpreted as running a single test on all of the data.

Analyzing multiomics data using classical statistical methods typically means that we perform extensive filtering of the data, removing genes with poor expression/methylation/mutation scores, and then e.g. perform logistic regression against a desired outcome, or alternatively, perform multiple statistical tests comparing each genomic feature independently against a desired outcome. Either way, information is lost during initial filtering and we must correct the analysis for each statistical test performed. While this increases confidence in whichever observation remains significant, it also undoubtedly means that we discard true positives. Additionally, classical statistical methods such as those mentioned here do not assume a spatial connection between data points, thus any relevant information relating to spatial organization is lost.

Instead, we propose the use of the GENIUS framework for multiomics analysis. The GENIUS framework is based on deep neural nets and relies on Convolutions and their ability to extract interactions between the data points. This particularly considers spatial information, which is not possible using classical statistical methods such as logistic regression where the most similar approach to this would include creating many models with many interactions.

Furthermore, integrated gradients is a non-parametric approach that simply evaluates the trained model relative to input data and output label, resulting in attribution for each input with respect to the output label. In other words, integrated gradients represent the integral of gradients with respect to inputs along the path from a given baseline to input. The integral is described in Author response image 1:

Author response image 1.

$$\text{IntegratedGrads}_i(x) ::= (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} \, d\alpha$$

More about integrated gradients can be read on the Captum webpage (https://captum.ai/docs/introduction) or in original paper https://arxiv.org/abs/1703.01365.

Since we transformed the data into a data structure (genome image) that assumes a spatial connection between genes, trained the model using convolutional neural networks and analyzed the model using integrated gradients, we can treat the results without any parametric assumption. As a particular novelty, we can sort the list based on attribution score and take top N genes as our candidate biomarkers for the variable of interest and proceed with downstream analysis or potentially functional validation in an in vitro setting. In this manner, the reviewer is correct that the signal-to-noise evaluation is shifted to the post-processing steps. However, the benefit of the GENIUS framework is particularly that it enables integration of multiple data sources without any filtering, and with constructing a novel data structure that facilitates investigation of spatial dependency between data points, thus potentially revealing novel genes or biomarkers that were previously removed through filtering steps. However, further downstream validation of these hits remains critical.

We added the following paragraph to make this more clear

"Integrated Gradients is a non-parametric approach that evaluates the trained model relative to input data and output label, resulting in attribution scores for each input with respect to the output label. In other words, Integrated Gradients represent the integral of gradients with respect to inputs along the path from a given baseline. By using Integrated Gradients, we provide an alternative solution to the problem posed by performing multiple independent statistical tests. Here, instead of performing multiple tests, a single analysis is performed by transforming multiomics data into genome images, training a model, and inspecting it with Integrated Gradients. Integrated Gradients will output an attribution score for every gene included in the genome image and those can be ranked in order to retrieve a subset of the most associated genes relative to the output variable."

> *In addition, multiple-testing correction is usually done based on one particular data source (e.g. expression data), while their approach claims to integrate five very different genomic data sources with different levels and structures of technical noise. How are these applications comparable and how is the training procedure able to account for these different structures of technical noise? Please provide sufficient evidence for making this claim (especially in the postprocessing steps after classification).*

The reviewer is correct that there will be different technical noise for each data source. However, each data source is already processed by standardized pipelines used for interpreting sequence-level data into gene expression, mutations, copy number alterations and methylation levels. Thus, sequence-level technical noise is not evaluated as part of the GENIUS analysis. Nevertheless, the reviewer is correct that sample-level technical noise, such as low tumor purity or poor quality sequencing, undoubtedly can affect the GENIUS predictions, as is true for all types of sequence analysis. As part of GENIUS, an initial data preprocessing step (which is performed automatically as part of the image generation), is that each data source is normalized within that source and linearly scaled in range zero to one (min-max scaling). This normalization step means that the impact of different events within and between data sources are comparable since the largest/smallest value from one data source will be comparable to the largest/smallest value from another data source.

Additionally, deep neural networks, particularly convolutional networks, have been shown to be very robust to different levels of technical noise (Jang, McCormack, and Tong 2021; Du et al. 2022). In the manuscript we show the attribution scores for different cancer types in figure 3B of the paper. Here, the top genes include established cancer genes such as P53, VHL, PTEN, APC and PIK3CA, indicating that the attribution scores based on GENIUS analysis is a valid tool to identify potential genes of interest. Furthermore, when focusing the analysis on predicting metastatic bladder cancer, we were able to show that of the top 10 genes with the highest attribution scores, 7 showed significant association with poor outcome in an independent validation cohort of mostly metastatic patients (shown in figure 4).

> - *I didn't find any computational benchmark of GENIUS. What are the computational run times, hardware requirements (e.g. memory usage) etc that a user will have to deal with when running an analogous experiment, but with different input data sources? What kind of hardware is required GPUs/CPUs/Cluster?*

We apologize for not including this information in the manuscript. We added the following section in to the manuscript:

"Computational Requirements

In order to train the model, we used the following hardware configuration: Nvidia RTX3090 GPU, AMD Ryzen 9 5950X 16 core CPU, and 32Gb of RAM memory. In our study, we used a batch size of 256, which occupied around 60% of GPU memory. Training of the model was dependent on the output variable. For metastatic disease prediction, we trained the model for approximately 4 hours. This could be changed since we used early stopping in order to prevent overfitting. By reducing the batch size to smaller numbers, the technical requirements are reduced making it possible to run GENIUS on most modern laptops."

> - *A general comment about the Methods section: Models, training, and validation are very vaguely described and the source code on GitHub is very poorly documented so that parameter choices, model validation, test and validation frameworks and parameter choices are neither clear nor reproducible.*

Apologies, we have updated the methods section with more details on models, training and validation. Additionally, we have moved the section on evaluating model performance from the methods section to the results section, with more details on how training was performed.

We also agree that the GitHub page is not sufficiently detailed and well structured. To remedy this, we have made a new GitHub page that only has the code needed for analysis, example input data, example runs, and environment file with all library versions. The GitHub repository is also updated in the manuscript.

The new GitHub page can be found on: https://github.com/mxs3203/GENIUS

> *Please provide a sufficient mathematical definition of the models, thresholds, training and testing frameworks.*

We sincerely apologize, but we do not entirely follow the reviewers request on this regard. The mathematical definitions of deep neural networks are extensive and not commonly included in research publications utilizing deep learning. We have used PyTorch to implement the deep neural net, a commonly used platform, which is now referenced in the methods. The design of the deep learning network used for GENIUS is described in figure 1,

and the relevant parameters are described in methods. The hyper parameters are described in the methods section, and are as follows:

"All models were trained with Adagrad optimizer with the following hyperparameters: starting learning rate = 9.9e-05 (including learning rate scheduler and early stopping), learning rate decay and weight decay = 1e-6, batch size = 256, except for memory-intensive chromosome images where the batch size of 240 was used."

> - *In chapter "Latent representation of genome" the authors write: "After successful model training, we extracted the latent representations of each genome and performed the Uniform Manifold Approximation and Projection (UMAP) of the data. The UMAP projected latent representations into two dimensions which could then be visualized. In order to avoid modeling noise, this step was used to address model accuracy and inspect if the model is distinguishing between variables of interest.". In the recent light of criticism when using the first two dimensions of UMAP projections with omics data, what is the evidence in support of the author's claim that model accuracy can be quantified with such a 2D UMAP projection? How is 'model accuracy' objectively quantified in this visual projection?*

We apologize for not clarifying this. The UMAP was done on L, the latent vector, which by assumption should capture the most important information from the “genome image”. In order to confirm this, we plotted the first two dimensions of UMAP transformation and colored the points by the output variable. If the model was capturing noise, there should not be any patterns on the plot (randomized cancer-type panel). Since, in most cases, we do see an association between the first two UMAP dimensions and the output variable, we were confident that the model was not modeling (extracting) noise.

To clarify this, we changed the sentence in the manuscript so it is more clear that this is not an estimation of accuracy but only an initial inspection of the models:

The UMAP projected latent representations into two dimensions which could then be visualized. In order to avoid modeling noise, this step was used to inspect if the model is distinguishing between variables of interest.

> - *In the same paragraph "Latent representation of genome" the authors write: "We observed that all training scenarios successfully utilized genome images to make predictions with the exception of Age and randomized cancer type (negative control), where the model performed poorly (Figure 2B).". Did I understand correctly that all negative controls performed poorly? How can the authors make any claims if the controls fail? In general, I was missing sufficient controls for any of their claims, but openly stating that even the most rudimentary controls fail to deliver sufficient signals raises substantial issues with their approach. A clarification would substantially improve this chapter combined with further controls.*

We apologize for not stating this more clearly. Randomized cancer type was used as a negative control since we expect that model would not be able to make sense of the data if predicting randomized cancer type. As expected, the model failed to predict the randomized cancer types. This can be seen in Figure 2C, where UMAP representations (based on the latent representation of the data, the vector L) are made for each output variable. Not seeing any patterns in UMAP shows that, as expected, the model does not know how to extract useful information from “genome image” when predicting randomized cancer type (as when randomly shuffling the labels there is no genomic information to decipher). Similar patterns

were observed for Age, indicating that patient age cannot be determined from the multi-omics data. Conversely, when GENIUS was trained against wGII, TP53, metastatic status, and cancer type, we observed that samples clustered according to the output label.

> ***Reviewer #2 (Public Review):***
>
> *In this manuscript, Birkbak and colleagues use a novel approach to transform multi-omics datasets in images and apply Deep Learning methods for image analysis. Interestingly they find that the spatial representation of genes on chromosomes and the order of chromosomes based on 3D contacts leads to best performance. This supports that both 1D proximity and 3D proximity could be important for predicting different phenotypes. I appreciate that the code is made available as a github repository. The authors use their method to investigate different cancers and identify novel genes potentially involved in these cancers. Overall, I found this study important for the field.*
>
> *The major points of this manuscript could be grouped in three parts:*
>
> 1. *While the authors have provided validation for their model, it is not always clear that best approaches have been used.*
>
>   *a) In the methods there is no mention of a validation dataset. I would like to see the authors training on a cancer from one cohort and predict on the same cancer from a different cohort. This will convince the reader that their model can generalise. They do something along those lines for the bladder cancer, but no performance is reported. At the very least they should withhold a percentage of the data for validation. Maybe train on 100 and validate on the remaining 300 samples. They might have already done something along these lines, but it was not clear from the methods.*

Apologize for not being sufficiently clear in the manuscript. We did indeed validate the performance within the TCGA cohort, using holdout cross validation. Here, we trained the network on 75% of the cohort samples (N = 3825), and tested on the remaining 25% (N = 1276).

To make this more clear, we have rewritten section “GENIUS classification identifies tumors likely to become metastatic” as such:

"The omics data types included somatic mutations, gene expression, methylation, copy number gain and copy number loss. Using holdout type cross-validation, where we split the data into training (75%) and validation (25%), we observed a generally high performance of GENIUS, with a validation AUC of 0.83 for predicting metastatic disease (Figure 2B)."

We also added the following sentence in the legend of Figure 2:

"The x-axis represents epochs and y-axis represents AUC score of fixed 25% data we used for accuracy assessment within TCGA cohort."

The accuracy of GENIUS could not be validated on the other two bladder cohorts since they do not contain all the data for the creation of five-dimensional genome images. However, we were able to investigate if the genes with the highest attribution scores towards metastatic bladder cancer obtained based on the TCGA samples also showed a significant association with poor outcome in the two independent bladder cancer cohorts. Here, we observed that of the top 10 genes with the highest attribution scores, 5 were associated with poor outcome in the early stage bladder cancer cohort, and 7 were associated with poor outcome in the late stage/metastatic bladder cancer cohort.

> *b) It was not clear how they used "randomised cancer types as the negative control". Why not use normal tissue data or matched controls?*

In the study, we built six models, one for each variable of interest. One of them was cancer type which performed quite well. In order to assess the model on randomized data, we randomized the labels of cancer type and tried predicting that. This served as “negative control” since we expected the model to perform poorly in this scenario. To make this more clear in the manuscript, we have expanded the description in the main text. We have also added the description of this to each supplementary plot to clarify this further.

While normal tissue and matched controls would have been an optimal solution, unfortunately, such data is not available.

> *c) If Figure 2B, the authors claim they have used cross validation. Maybe I missed it, but what sort of cross validation did they use?*

We apologize for not being sufficiently clear. As described above, we used holdout cross-validation to train and evaluate the model. We clarified this in the text:

"Using holdout type cross-validation, where we split the data into training (80%) and validation (20%), we observed a generally high performance of GENIUS, with a mean validation AUC of 0.83 (Figure 2B)"

> *1. Potential improvement to the method*
>
> *a) It is very encouraging the use of HiC data, but the authors used a very coarse approach to integrate it (by computing the chromosome order based on interaction score). We know that genes that are located far away on the same chromosome can interact more in 3D space than genes that are relatively close in 1D space. Did the authors consider this aspect? Why not group genes based on them being located in the same TAD?*

We thank the reviewer for this suggestion and we will start looking into how to use TAD information to create another genome representation. In this study, we tried several genome transformations, which proved to be superior compared to a flat vector of features (no transformation). We are aware that squared genome transformation might not be optimal, so we designed the network that reconstructs the genome image during the training. This way, the genome image is optimized for the output variable of choice by the network itself. However, we note that the order of the genes themselves, while currently based on HiC, can be changed by the user. The order is determined by a simple input file which can be changed by the user with the argument “all_genes_included”. Thus, different orderings can be tested within the overall square layout. This is now detailed in the instructions on the new GitHub page.

The convolutional neural network uses a kernel size of 3x3, which captures the patterns of genes positioned close to each other but also genes that are far away from each other (potentially on another chromosome). Once convolutions extract patterns from the image, the captured features are used in a feed-forward neural network that makes a final prediction using all extracted features/patterns regardless of their location in the genome image.

We also inserted the following sentence in discussion:

"Given that spatial organization improved the prediction, we recognize that there may exist a more optimal representation of multi-omics data which should be explored further in future work. Potential methods for organizing gene orientation in a 2D image could consider

integrating topologically associating domains[39] along with the spatial information from HiC. This is already possible to explore with the current implementation of GENIUS, where gene layout can be set manually by the user."

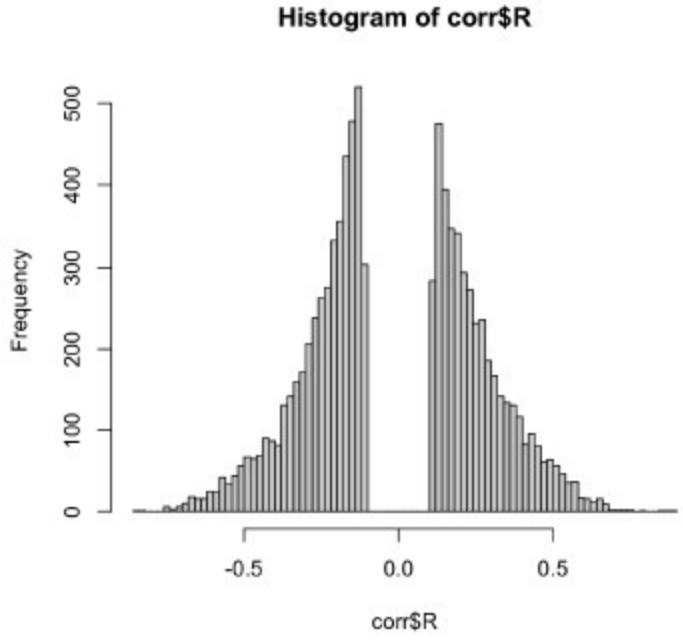
> *b) Authors claim that "given that methylation negatively correlates with gene expression, these were considered together". This is clearly not always the case. See for example https://genomebiology.biomedcentral.com/articles/10.1186/s13059-022-02728-5. What would happen if they were not considered together?*

We thank the reviewer for this insightful comment. We agree with the reviewer that methylation does not always result in lower expression, although methylation levels in most cases should correlate negatively to RNA expression, but with a gene-specific factor. Indeed, there are tools developed that infer RNA expression based on methylation, making use of gene-specific correction factors. E.g. Mattesen et al (Mattesen, Andersen, and Bramsen 2021).

However, upon reflection we agree with the reviewer that we cannot assume for all genes that methylation equals low expression. Therefore, we have performed an analysis where we compared the methylation level to gene expression levels for all tested genes within bladder cancer. We computed Pearson's correlation of 16,456 genes that have both methylation and expression scores. Of these, 8528 showed a negative correlation. After p-value correction, this resulted in 4774 genes where methylation was significantly negatively associated with expression. For these genes we performed the subsequent analysis in bladder cancer, where methylation and expression were considered together. This updated analysis has been included in supplementary figure 10, and the results section has been amended to reflect this. Overall, this analysis resulted in 4 of 10 genes being replaced in the downstream analysis. However, we note that the final results did not materially change, nor did the conclusions.

Author response image 2.

Correlation between gene-level methylation and gene expression in TCGA BLCA cohort



1. Interesting results that were not explained.

a) In Figure 3A methylation seems to be the most important omics data, but in 3B, mutations and expression are dominating. The authors need to explain why this is the case.

We apologize for not explaining this in more detail. Figure 3B shows the attribution scores scaled within the cancer type, where Figure 3A shows raw attribution scores for each data source included. The reason for this is that methylation and expression have in general, smaller attribution scores but more events where a single mutation often is characterized with large attribution scores and the rest of them with very small attribution. In order to make those numbers comparable and take into account biological differences between the cancer type, we scaled the scores within each cancer type.

To make this more clear we modified the first sentence in “Interpreting the GENIUS model classifying metastatic cancer biology” section:

"Analysing raw attribution scores we concluded the most informative data type overall regarding the development of metastatic disease was methylation (Figure 3A). …We also noticed that mutation data often had a single mutation with large attribution score where expression and methylation showed multiple genes with high attribution scores… … The normalization step is crucial to make results comparable as underlying biology is different in each cancer type included in the study."

Reviewer #1 (Recommendations For The Authors):

- *While I appreciate the creative acronym of the presented software solution (GENIUS), it may easily be confused with the prominent software Geneious | Bioinformatics Software for Sequence Data Analysis which is often employed in molecular life science research. I would suggest renaming the tool.*

We appreciate the comment but prefer to keep the name. Given that the abbreviation is not exactly the same and the utility is different, we are confident that there will be no accidental mixup between these tools.

- *A huge red flag is the evaluation of the input image design which clearly shows that classification power after training is insufficient for three out of four image layouts (and even for the fourth AUC is between 0.70-0.84 depending on the pipeline step and application). Could the authors please clarify why this isn't cherry-picking (we use the one layout that gave some form of results)? In light of the poor transformation capacity of this multi-omics data onto images, why weren't other image layouts tried and their classification performance assessed? Why should a user assume that this image layout that worked for this particular input dataset will also work with other datasets if image transformation is performing poorly in most cases?*

We apologize for not describing this further in the manuscript. We wrote in the manuscript that we could not know what genome representation is optimal as it is difficult to know. A flat vector represents a simple (or no) transformation since we simply take all of the genes from all of the data sources and append them into a single list. Chromosome image and square image are two transformations we tried, and we focused on the square image since in our hands it showed superior performance relative to other transformations.

Reviewer #2 (Recommendations For The Authors):

Minor points:

1. Legends of supplementary Figures are missing.

We thank the reviewer for this comment and apologize for missing it. All legends have been added now.

1. For some tests the authors use F1 score while for other AUC, they should be consistent. Report all metrics for all comparisons or report one and justify why that only metric.

We apologize for not being sufficiently clear. AUC is a standard score used for binary classification, while the F1 score is used for multiclass classification. We have now described this in the methods section, and hope this is now sufficiently clear.

"When predicting continuous values, the model used the output from the activation function with the mean squared error loss function. When predicting multi-class labels, the performance measure was defined by the F1 score, a standard measure for multiclass classification that combines the sensitivity and specificity scores and is defined as the harmonic mean of its precision and recall. To evaluate model performance against the binary outcome, ROC analysis was performed, and the area under the curve (AUC) was used as the performance metric."

1. not sure how representation using UMAP in Figure 2C is helping understand the performance.

Apologies for the poor wording in the results section. The purpose of the UMAP representation was to visually inspect if the model was distinguishing between variables of interest, not to estimate model performance. We have rephrased the text in the methods section to make this clear:

"After successful model training, we extracted the latent representations of each genome and performed the Uniform Manifold Approximation and Projection (UMAP) of the data for the purpose of visual inspection of a model."

And

"In order to avoid modeling noise, this step was used to inspect if the model is distinguishing between variables of interest."

And also in the results section:

"In order to visually inspect patterns captured by the model, we extracted the latent representations of each genome and performed the Uniform Manifold Approximation and Projection (UMAP) of the data to project it into two dimensions."

1. Instead of pie chart in 3A, the authors should plot stacked barplots (to 100%) so it would be easier to compare between the different cancer types.

We thank the reviewer for the suggestion; however, since we wanted to compare the relative impact of each data source with each other, we used pie charts. Piecharts are often better for describing relative values, whereas bar plots are better for absolute values.

References

Du, Ruishan, Wenhao Liu, Xiaofei Fu, Lingdong Meng, and Zhigang Liu. 2022. “Random Noise Attenuation via Convolutional Neural Network in Seismic Datasets.” Alexandria Engineering Journal 61 (12): 9901–9.

Jang, Hojin, Devin McCormack, and Frank Tong. 2021. “Noise-Trained Deep Neural Networks Effectively Predict Human Vision and Its Neural Responses to Challenging Images.” PLoS Biology 19 (12): e3001418.

Mattesen, Trine B., Claus L. Andersen, and Jesper B. Bramsen. 2021. “MethCORR Infers Gene Expression from DNA Methylation and Allows Molecular Analysis of Ten Common Cancer Types Using Fresh-Frozen and Formalin-Fixed Paraffin-Embedded Tumor Samples.” Clinical Epigenetics 13 (1): 20.