OXFORD

A comprehensive review of cancer survival prediction using multi-omics integration and clinical variables

Dao Tran [ID][1], Ha Nguyen [ID][1], Van-Dung Pham[1], Phuong Nguyen[1], Hung Nguyen Luu [ID][2,3], Liem Minh Phan[4],

Christin Blair DeStefano[5], Sai-Ching Jim Yeung[6], Tin Nguyen [ID][1,*]

[1]Department of Computer Science and Software Engineering, Auburn University, 345 W Magnolia Avenue, Auburn, AL 36849, United States
[2]UPMC Hillman Cancer Center, University of Pittsburgh Medical Center, 5150 Centre Avenue, Pittsburgh, PA 15232, United States
[3]Department of Epidemiology, School of Public Health, University of Pittsburgh, 130 De Soto Street, Pittsburgh, PA 15261, United States
[4]David Grant USAF Medical Center—Clinical Investigation Facility, 60th Medical Group, Defense Health Agency, 101 Bodin Circle, Travis Air Force Base, CA 94535, United States
[5]Walter Reed National Military Medical Center, Defense Health Agency, 8901 Rockville Pike, Bethesda, MD 20889, United States
[6]Department of Emergency Medicine, The University of Texas MD Anderson Cancer Center, 1400 Pressler Street, Houston, TX 77030, United States

*Corresponding author. Department of Computer Science and Software Engineering, Auburn University, 345 W Magnolia Avenue, Auburn, AL 36849, USA. E-mail: tinn@auburn.edu

Abstract

Cancer is an umbrella term that includes a wide spectrum of disease severity, from those that are malignant, metastatic, and aggressive to benign lesions with very low potential for progression or death. The ability to prognosticate patient outcomes would facilitate management of various malignancies: patients whose cancer is likely to advance quickly would receive necessary treatment that is commensurate with the predicted biology of the disease. Former prognostic models based on clinical variables (age, gender, cancer stage, tumor grade, etc.), though helpful, cannot account for genetic differences, molecular etiology, tumor heterogeneity, and important host biological mechanisms. Therefore, recent prognostic models have shifted toward the integration of complementary information available in both molecular data and clinical variables to better predict patient outcomes: vital status (overall survival), metastasis (metastasis-free survival), and recurrence (progression-free survival). In this article, we review 20 survival prediction approaches that integrate multi-omics and clinical data to predict patient outcomes. We discuss their strategies for modeling survival time (continuous and discrete), the incorporation of molecular measurements and clinical variables into risk models (clinical and multi-omics data), how to cope with censored patient records, the effectiveness of data integration techniques, prediction methodologies, model validation, and assessment metrics. The goal is to inform life scientists of available resources, and to provide a complete review of important building blocks in survival prediction. At the same time, we thoroughly describe the pros and cons of each methodology, and discuss in depth the outstanding challenges that need to be addressed in future method development.

Keywords: survival prediction; multi-omics integration; clinical variables; deep neural network

Introduction

Recent advancements in high-throughput technologies have revolutionized cancer research by allowing us to evaluate the patients and their cancers at different molecular layers, including DNA single nucleotide polymorphism and variation, gene mutations (insertion, deletion, translocation, copy number variation, point mutation, etc.), transcription, translation, transcriptional regulation (e.g. regulation by DNA methylation), post-transcriptional regulation (e.g. acetylation, lactylation, glycosylation, ubiquitination, etc.), expression of noncoding RNAs, and metabolites, etc. In turn, these technological and research advances led to the discovery of additional cancer hallmarks and broadening of our cancer knowledge bases [1–7]. By analyzing multiple molecular layers together, researchers can obtain a more comprehensive view of cancer evolution and prognosis [8–10]. For instance, Granja *et al.* [11] combined transcriptomic, epigenomic, and proteomic data of leukemic blood cells to identify cancer-specific processes involved in blood differentiation and key regulators of

leukemia-specific genes. Similarly, other multi-omics studies resulted in the discovery of molecular signatures of breast cancer [12, 13], liver cancer [14, 15], lung cancer [16, 17], pancreatic cancer [18, 19], brain cancer [20, 21], and other cancer types [22–24].

Because of the recognized importance of molecular data, many survival prediction models have been developed using either: (1) single-omics data [25, 26], (2) multi-omics data [14, 27], or (3) multi-omics and clinical data combinations [28–34]. Despite the increasing importance of these integrative methods, there is a lack of resources to guide researchers through important concepts in integrative prediction: (1) data processing, (2) modeling of survival time (continuous versus discrete time), (3) modeling of observable covariates (multi-omics measurement and clinical variables), (4) coping with censored and missing data, (5) data integration strategies, (6) effective prediction methodologies, and (7) validation and assessment metrics. Many review articles exist but they are often tailored toward a specific type of cancer or data type (gene expression, clinical variables, or image data, etc.) [35–39].
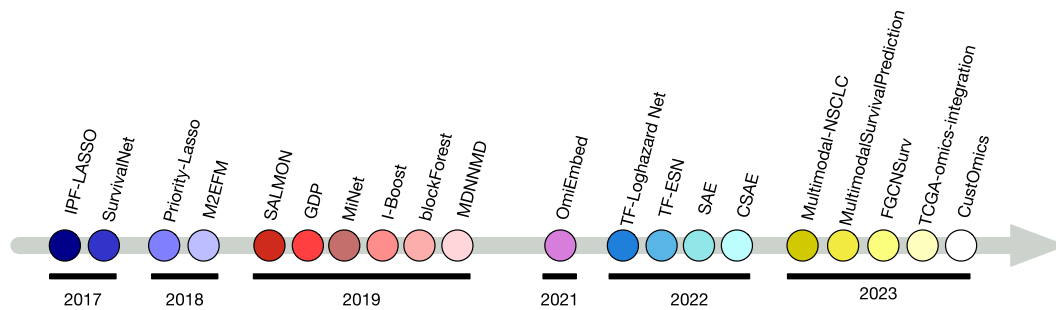
Figure 1. Timeline of computational methods developed for cancer survival prediction using multi-omics data and clinical variables.

In this article, we review 20 methods capable of integrating multi-omics data and clinical variables for survival prediction. We focus on tools that have working source code and are well maintained. There are other tools for the analysis of single-omics [40–43] and image data [44–46] but those methods are excluded from our review because they are not designed for multi-omics integration. Figure 1 shows the 20 methods developed between 2017 and 2023.

The manuscript is organized as follows. Section Integration and validation describes the high-level workflow of survival prediction methods, recapitulating common techniques in data processing, multi-omics data integration, and validation metrics. Section Modeling and censoring discusses the modeling of survival time, the incorporation of observable covariates into survival prediction, and the strategies of handling censored data. Section Technical details of prediction methods categorizes the machine learning techniques and discusses their pros and cons. Section Summary and practical guideline summarizes the key characteristics of each method and provides a practical guideline for users to choose a suitable method. Section Outstanding challenges discusses the outstanding challenges that need to be addressed in future research. Finally, section Conclusion concludes the article.

Integration and validation

Figure 2A shows the general workflow that prediction approaches follow. The input includes multi-omics data and clinical variables. The output can be survival time, survival probability, vital status with probability, hazard ratio, or cumulative hazard risk. These supervised approaches first learn from the training set before they can predict the survival of new patients. The training data consist of data from a set of patients in which both input and survival outcomes are known. After learning from the training data, the models can predict the outcomes of new patients. The development of a survival prediction model usually consists of three stages: data processing, training and integration, and validation, which will be explained in the following subsections.

Data processing

Data processing consists of filtering, normalization, imputation, gene mapping, gene-level aggregation, and sample intersection. The goal of filtering is to remove features/variables that have none to weak association with patients' outcomes. Data normalization is tailored to each data type and includes log2, min-max, z-score transformation, and one-hot encoding [47, 48]. For sequencing data, depth normalization is an important step. OmiEmbed, TF-ESN, TF-LogHazardNet, and I-Boost convert read counts into FPKM/RPKM or TPM. The remaining methods validate their approaches using pre-processed and normalized data from TCGA and NCBI GEO. Data imputation is often necessary because of

missing values. For example, if a gene has only some missing values for a specific patient, it is desirable to impute the missing values instead of removing the gene from the analysis. FGCNSurv, MiNet, TCGA-omics-integration, M2EFM, MDNNMD, and blockForest use KNN imputation [49] to substitute missing values of a patient with mean values of those features from the nearest neighbors of the patients. Other methods simply replace missing values with mean, median, or zero. Another important step is batch correction, but is neglected by most approaches. M2EFM is the only method that uses the *ComBat* function in the sva package [50] to correct for batch effects of both training and testing datasets.

Gene-level aggregation is performed by some methods. GDP, MiNet, TF-Loghazard Net, and TF-ESN require users to perform gene-level aggregation for omics types other than gene expression (e.g. DNA methylation, copy number variation). MiNet, TF-Loghazard Net, and TF-ESN further intersect the gene-level features among all molecular types. Finally, all approaches intersect samples/patients among multi-omics, clinical, and survival data to keep patients that have all types of data for the training process.
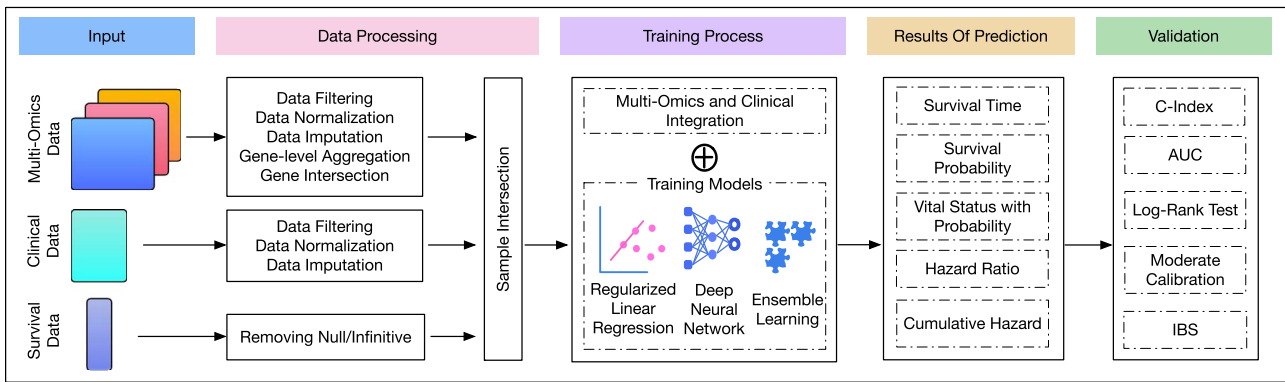
Training and data integration

The training process generally consists of model training and data integration, which can be executed either simultaneously or separately. For training, the methods implement three distinctive types of models: regularized linear regression, deep neural network, and ensemble learning. Details about model training and machine learning techniques are described in section Technical details of prediction methods.

Figure 2B shows the integration strategies, which can be categorized as early, middle, mixed, and late integration [51, 52]. Early integration involves concatenating the input data matrices into a single matrix, and potential dimension reduction (low-variance filtering, univariate survival models, etc.) to reduce noise and computational burden. Methods following the early integration strategy include IPF-LASSO, Priority-Lasso, M2EFM, SALMON, GDP, MiNet, SurvivalNet, I-Boost, blockForest, and TCGA-omics-integration.

Middle and mixed integration strategies use representation learning to capture shared information. They assume that all input matrices can be decomposed into a common latent space, revealing the underlying mechanisms [52, 53]. Middle integration approaches include Multimodal_NSCLC, TF-Loghazard Net, TF-ESN, SAE, CSAE, OmiEmbed, and FGCNSurv. These methods use an autoencoder (AE) [54] to encode multi-omics data into a common representation, which is then concatenated with clinical data. In some cases, middle integration methods use an AE to simultaneously encode multi-omics and clinical data into a common representation. Mixed integration methods use AEs or multilayer perceptrons (MLPs) [55] to transform each data type into a lower

A - Overall Pipeline
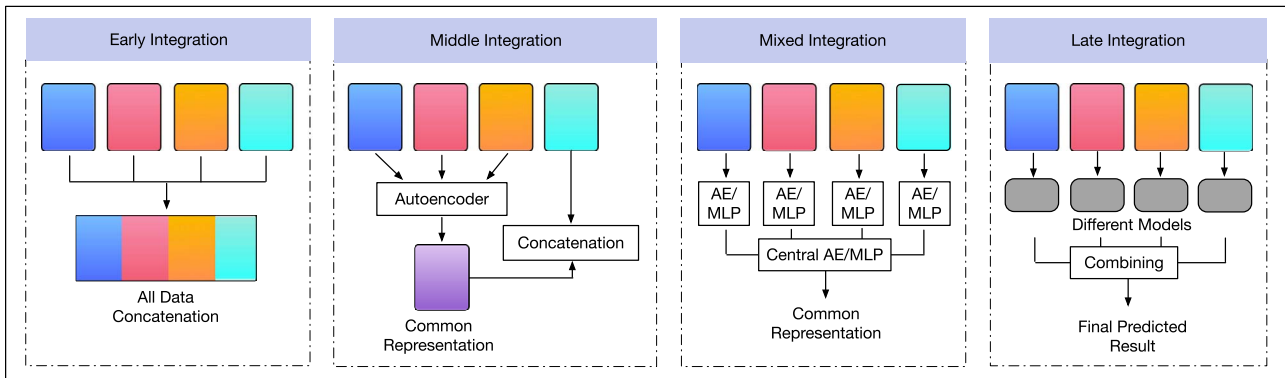

B - Integration Strategies

Figure 2. High-level workflow (A) and data integration strategies (B) of survival prediction methods.

dimensional representation before combining them. Multimodal-SurvivalPrediction and CustOmics adapt the mixed integration strategy.

Late integration is a meta-analysis approach that trains separate survival prediction models for each input type and then quantitatively combines the predicted values using an ensemble learning strategy such as applying a weighted aggregation function. MDNNMD is the sole method that uses the late integration strategy.

Validation and assessment metrics

To validate the methods, developers apply them to another set of patients with known outcomes that are not included in the training set. The performance of the methods is measured by comparing the predicted outcomes with the ground truth using several metrics. These metrics measure the following capabilities of prediction models: (1) **discrimination**, (2) **calibration**, and (3) **overall performance**.

Discrimination refers to the ability to separate and rank patients according to their survival probability. Concordance index (C-Index) is a metric that compares the rank order of patients based on predicted hazard or survival probability against the true ranking according to survival time [56]. Time-independent C-Index, such as Harrell's C-Index [57] or Uno's C-Index [58], validates outcomes such as prognostic index, or risk score of patients. In contrast, the time-dependent C-Index is used for time-varying outcomes, such as survival probability of patients across time intervals. In total, 18 out of 20 methods (all but IPF-LASSO and MDNNMD) use C-Index as their main assessment metric. Priority-Lasso uses Uno's C-Index, while

TF-Loghazard Net and TF-ESN use the time-dependent C-Index. The remaining 15 approaches use Harrell's C-Index.

Two other metrics for discrimination power are the area under the ROC curve (**AUC**) and **Log-Rank** test. One can define a threshold and transform predicted values or survival probabilities into binarized vital status. In this case, the AUC value measures how well the prediction model distinguishes between the patients experiencing the event and those who do not. Priority-Lasso, FGCNSurv, and MDNNMD use AUC. Based on the predicted probability of hazard or survival, one can also stratify patients into two different groups and utilize the Log-Rank test to examine significant differences between the true survival probability in these groups [59]. M2EFM, SALMON, MiNet, SAE, CSAE, FGCNSurv, CustOmics, and MDNNMD use the Log-Rank test.

Calibration refers to the accuracy of the predicted risk in comparison with the true probability of the event. There are three levels of calibration that are listed in an increasingly stringent order: mean calibration, weak calibration, and moderate calibration [60–62]. Mean calibration requires that the average predicted risk equals to the overall event rate which can be computed based on Kaplan–Meier estimation [63]. Weak calibration requires that the model generally does not overestimate or underestimate the predicted risk for any patient. One can evaluate moderate calibration by separating patients into different groups based on their predicted risks and comparing the predicted against the true probability of the event within each group. Priority-Lasso and M2EFM evaluate their models using moderate calibration.

The **overall performance** indicates both discrimination and calibration power of prediction models. Integrated Brier Score (IBS) [64] is a standard metric to assess the overall performance.

Brier Score is calculated for each observed time as the average of the sum of squared differences between the predicted probability of event and the observed outcomes (e.g. 1 is dead and 0 is alive) of patients [65]. IBS is obtained by combining the Brier scores for each observed time. Five methods, IPF-LASSO, Priority-Lasso, OmiEmbed, TCGA-omics-integration, and CustOmics, assess their overall performance using IBS. One can also use both discrimination and calibration metrics listed above in their performance assessment.

Modeling and censoring
Continuous versus discrete-time modeling

Survival prediction methods generally formulate the random variable T that describes the length of time until the occurrence of a well-defined event of interest (e.g. death, disease recurrence) using the **survival function** and **hazard function**. The **survival function** $S(t)$ is defined as $S(t) = P(T > t)$, which is the probability that an individual will survive past time t. In contrast, the **hazard function** $h(t)$ represents how likely a patient will experience the event given that the individual has already survived past time t. The hazard can be considered as mortality rate or instantaneous risk at the time t. Crucially, specifying the survival function allows the hazard function to be ascertained and vice versa (see Fig. 3).

In continuous-time survival representation, T is a continuous variable with a cumulative distribution function (CDF) F and a probability density function (PDF) f. The **survival function** $S(t)$ is defined as $S(t) = P(T > t) = 1 - F(t)$. The **hazard function** is defined as $h(t) = \lim_{\Delta t \to 0} \frac{P(t < T < t + \Delta t | T > t)}{\Delta t} = \frac{f(t)}{S(t)} = -\frac{d}{dt} \ln S(t)$. The cumulative hazard function $H(t)$ is defined as $H(t) = \int_0^t h(x) dx$. The relationship between $H(t)$ and $S(t)$ is as follows: $H(t) = -\ln S(t)$ and $S(t) = e^{-H(t)}$.

Discrete-time approaches partition the survival time into contiguous intervals $\{(t_0, t_1], (t_1, t_2], ..., (t_{j-1}, t_j]\}$, where $t_0 = 0$. The PDF f is discretized so that $f(I_j) = P(T \in I_j) = P(t_{j-1} < T \le t_j)$, where I_j represents a specific time interval: $(t_{j-1}, t_j]$. The survival probability is given as $S(I_j) = P(T > t_j) = 1 - \sum_{k:t_k \le t_j} f(I_k)$, and the hazard is calculated as $h(I_j) = \frac{f(I_j)}{S(I_{j-1})}$. Note that the hazard can also be written as a conditional probability $h(I_j) = P(t_{j-1} < T \le t_j | T > t_{j-1})$, which is the probability of experiencing the event during the interval given that the patient has survived up to the start of that interval. The survival probability $S(t)$ can also be rewritten as $S(I_j) = P(T > t_j) = \prod_{k=1}^{j} (1 - h(I_k))$.

OmiEmbed, TF-Loghazard Net, TF-ESN, and MDNNMD formulate survival time using the discrete-time strategy. IPF-LASSO, Priority-Lasso, and blockForest allow users to choose one among different implemented models in which each model follows either the continuous-time or discrete-time modeling strategy. MultimodalSurvivalPrediction simultaneously trains two different models: one model formulates survival time using the continuous-time strategy and the other adapts the discrete-time strategy. All the remaining methods follow the continuous-time strategy.

The methods following the discrete-time strategy can model the survival/hazard of patients for each interval using a different formula. However, discrete-time methods encounter the potential issue of losing information of survival time. Segregating survival times into intervals results in the model being unable to account for variability in survival duration among patients experiencing the event within the same interval. On the contrary, the adaptation of continuous-time strategy typically requires strict assumptions about the distribution of survival quantities (e.g. survival time, survival function, hazard function, hazard ratio, etc.) across all observed times.

Note that none of the reviewed methods allows users to input variables for multiple time points. In order to incorporate time-varying variables, each sample should be represented by multiple records indicating different time points/intervals with corresponding covariate values [66–68]. Although many methods can be extended to process time-varying variables, all accompanied tools input only one value per variable per sample.

Parametric versus nonparametric modeling

Nonparametric models do not assume any parametric form for the survival function or the hazard function. The estimators such as Kaplan–Meier [63] and Nelson–Aalen [69, 70] estimate the survival probability and cumulative hazard directly from the set of observed time points. For example, Kaplan–Meier sorts survival times into an increasing order $\{t_1, t_2, ..., t_m\}$ and calculates the survival function as $S_{KM}(t) = \prod_{i:t_i \le t}(1 - \frac{d_i}{n_i})$, in which n_i is the number of alive patients at the beginning of t_i and d_i represents the number of patients experiencing the event at t_i. Nonparametric models can be used in comparing the survival among two or more groups of patients, but they often lack inference for effects of important covariates.

Parametric models can address some shortcomings of nonparametric models by formulating the parameters as a function of observable covariates (measurable variables). For example, we can model the survival time to follow the exponential distribution, such that $f(t) = \lambda e^{-\lambda t}$. The rate parameter λ represents the mortality rate (hazard) at time t because $h(t) = \frac{f(t)}{S(t)} = \lambda$. The parameter λ can depend on measurable covariates using $\lambda = e^{\beta X}$, where β is a row vector of coefficients and X is a column vector of covariates (e.g. gene expression of a person). Other commonly used models include Weibull, log-normal, log-logistic, and generalized gamma [71, 72]. Generally, parametric models are favored by researchers over nonparametric techniques because of the aforementioned advantages. However, parametric modeling requires users to ascertain the accurate distribution of survival time in their data, which can be a challenging task in practical scenarios. Moreover, false assumptions about data distribution may produce biases in the results of an analysis.

Consequently, semi-parametric models were developed to address these problems. In comparison to parametric techniques, semi-parametric models do not specify the distribution of survival time or the hazard function. Instead, these models specify the effects of the measurable covariates. Cox proportional hazards (CPH) [73]—one of the most widely used models—formulates the hazard function as $h(t) = h_0(t)e^{\beta X}$, in which $h_0(t)$ is an arbitrary **baseline hazard**, and $\frac{h(t)}{h_0(t)}$ is called the **hazard ratio**. The predicted survival probability is calculated as $S(t) = e^{-H_0(t)e^{\beta X}}$, where $H_0(t) = \int_0^t h_0(x) dx$. CPH models often estimate β using maximum likelihood estimation and estimate $H_0(t)$ using a nonparametric method named Breslow estimator [74, 75].

TF-Loghazard Net, TF-ESN, OmiEmbed, and MDNNMD follow the parametric modeling. IPF-LASSO and Priority-Lasso allow users to choose between parametric and semi-parametric models. MultimodalSurvivalPrediction implements both parametric and semi-parametric models in its analysis pipeline. blockForest is the only approach that utilizes the Nelson–Aalen estimator (nonparametric technique) to predict the survival of patients. The remaining methods implement only semi-parametric prediction models.
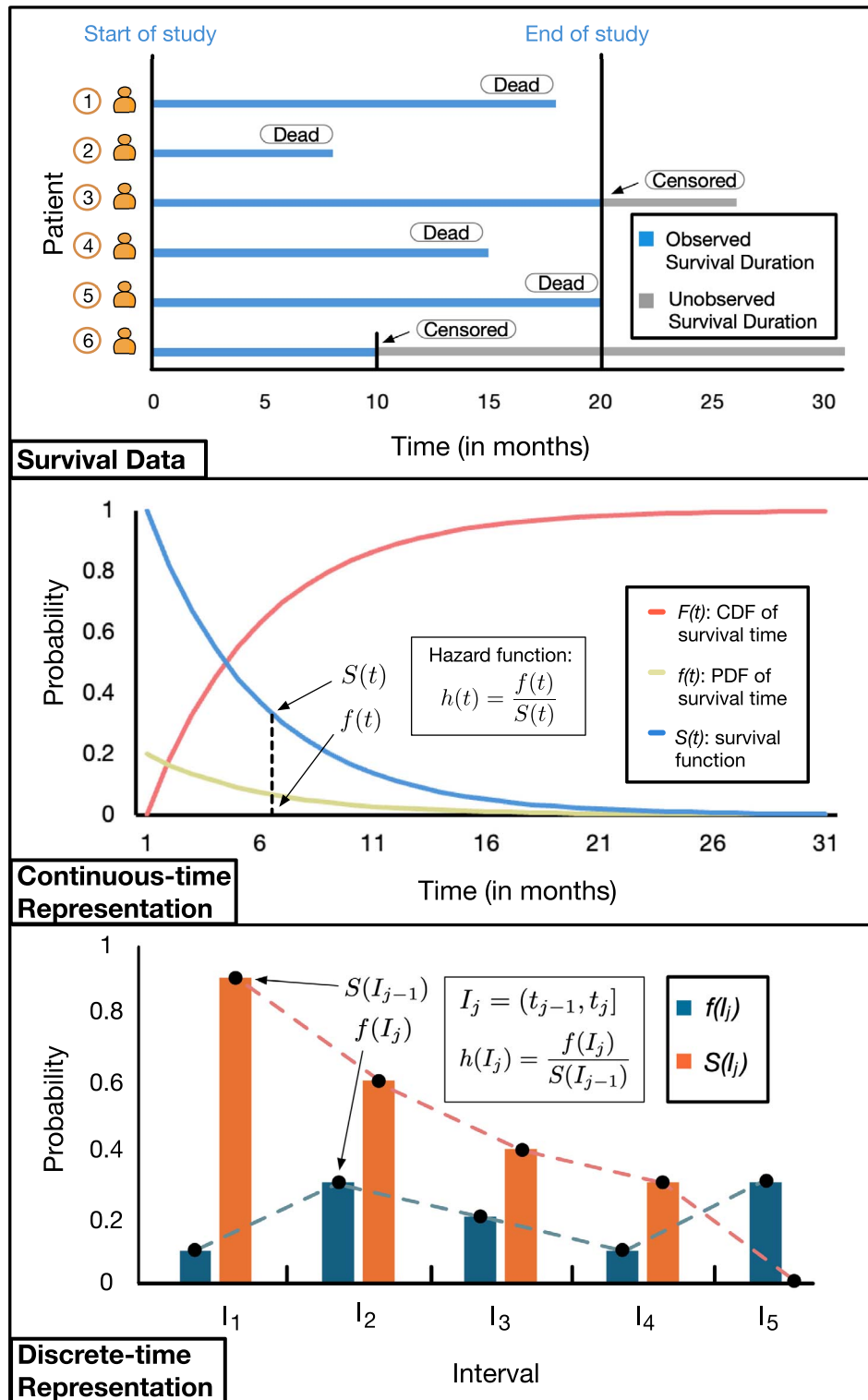
Figure 3. Fundamental concepts related to survival prediction encompassing survival function (**S**), hazard function (**h**), cumulative distribution function – CDF (**F**), and probability density function – PDF (**f**).

Coping with censored data

Censoring occurs when the time of event is unknown for a patient due to the following reasons: (1) study ends without the patient experiencing the event, (2) the patient withdraws from the study, and (3) study fails to follow up after some time. To cope with censored data, prediction methods either (1) remove the observations with censoring, (2) impute censored data, (3) dichotomize the data, or (4) adapt likelihood-based approaches [76]. The first two strategies are straightforward but removing censored data can lead to significant loss of data, while imputation might produce biased or false data, especially when the number of censored observations is large.

The third strategy binarizes the outcomes, i.e., it compares the incidence of occurrence versus nonoccurrence of the event within a fixed period of time. The disadvantages of this strategy include (i) being unable to distinguish between the loss to follow-up and end-of-study censoring, (ii) lacking the capability to model the variability of timing of the event, and (iii) failing to incorporate time-dependent covariates (age, smoking, etc.).

The fourth strategy adjusts the likelihood function to account for whether or not an individual observation is censored. An example is the Kaplan–Meier estimator described in the previous section. Another example is the CPH model that aims to maximize the partial likelihood function: $L(\beta) = \prod_{i=1}^{m} \frac{e^{\beta X_i}}{\sum_{j \in R_i} e^{\beta X_j}}$, in which $\{t_1, ..., t_m\}$ are the unique sorted event times, R_{t_i} is the set of patients that experience the event or are censored at a time beyond t_i. Although the likelihood-based strategy can utilize all available information, it still makes assumptions about the censoring mechanism.

MDNNMD applies the third strategy by defining a specific frame of time for its analysis (e.g. 5 years) and binarizes survival times of patients. TF-Loghazard Net, TF- ESN, and OmiEmbed combine the third and fourth strategies by specifying different time intervals that enclose the whole range of survival time and convert each observed time into a binary vector representing the interval that the observed time falls into. Patients experiencing the event in the same interval will be assigned with exactly the same binarized vector. Hazard or survival function across intervals is then estimated using the likelihood-based approach. This approach improves the third strategy, enabling the methods to differentiate between loss to follow-up and end-of-studying as well as account for time-varying covariates. However, the issue of variability in timing of the event remains unsolved. The remaining methods adapt the fourth strategy.

Technical details of prediction methods

We categorize the reviewed methods into three groups: regularized linear regression, deep neural network, and ensemble learning approaches. In each category, we provide a high-level description of the methods and their pros and cons. Throughout the section, we use a number of technical terms that are listed in Table 1. The description of each method can be found in Supplementary Note.

Regularized linear regression

There are three methods in this category: IPF-LASSO [77], M2EFM [29], and Multimodal_NSCLC [78]. They typically use CPH model, a semi-parametric approach that estimates the hazard ratio associated with given covariates, representing the likelihood of the event occurring. Because of the large number of predictors in the model, methods in this category also apply regularization techniques (LASSO, Ridge, elastic net) to select the important predictors and handle potential multicollinearity among them, thereby improving the model's predictive performance.

IPF-LASSO concatenates all data into a single matrix and then applies a model named LASSO with Penalty Factors that employs user-defined penalty factors for each data type to rescale the respective data, thus allowing distinct penalties for each coefficient in the objective function. The second method, M2EFM, performs expression quantitative trait loci (eQTLs) analysis with Matrix eQTL [79] to identify significantly associated probe-gene pairs. Next it trains a CPH model with Ridge regularization on multi-omics data and then integrates predicted values from the

Table 1. A glossary of technical terms

Term	Definition
Adam	Adaptive moment estimation
AE	Autoencoder
AFT	Accelerated failure time
AUC	Area under the ROC curve
CP	Canonical decomposition/parallel factors
C-Index	Concordance index
CNN	Convolutional neural network
CNV	Copy number variation
CPH	Cox proportional hazards
DNN	Deep neural network
FBM	Factorized bilinear model
FC	Fully connected neural network
GCN	Graph convolutional network
GNN	Graph neural network
GradNorm	Gradient normalization
IBS	Integrated brier score
KNN	K nearest neighbor
LASSO	Least absolute shrinkage and selection operator
MAML	Model-agnostic meta-learning
MSE	Mean squared error
MLP	Multilayer perceptron
NSCLC	Non-small cell lung cancer
SGD	Stochastic gradient descent
SVD	Singular value decomposition
VAE	Variational autoencoder

first model with clinical data to train a second CPH model without regularization. The third method, Multimodal_NSCLC, applies a CPH on each molecular feature and then selects features with the most significant effects on survival outcomes. Next, Multimodal_NSCLC uses AE to derive a common representation of multi-omics data (i.e. middle integration) and then concatenates the common representation with clinical variables to obtain a single matrix. Finally, the method trains a CPH model with elastic net regularization using the obtained matrix and patient survival.

Overall, regularized linear regression approaches utilize regularization techniques such as LASSO, Ridge, and elastic net, to avoid overfitting and deal with multicollinearity in high-dimensional data. Methods in this category can provide users with interpretability of the obtained results. However, linear regression models (linear regression, logistic regression and CPH) rely on specific assumptions about the covariate effects (e.g. additive/multiplicative, time-constant effects) on survival response variables. The violation of these assumptions may occur frequently in real-world scenarios, which could result in less ideal prediction [71, 80–83]. Furthermore, IPF-LASSO and Multimodal_NSCLC have high computation time. IPF-LASSO trains its models using a large concatenated data, while the denoising autoencoder implemented by Multimodal_NSCLC may be computationally intensive to train. M2EFM is highly dependent on specific omics types and the method does not allow incorporating additional molecular data in its analysis.

Deep neural networks

There are 12 methods in this category: SALMON [31], GDP [84], MiNet [85], SurvivalNet [86], TF-Loghazard Net & TF-ESN [87], SAE & CSAE [88], OmiEmbed [89], CustOmics [90], Multimodal-SurvivalPrediction [91], and FGCNSurv [92]. They integrate multi-omics and clinical data into deep neural network architectures such as MLP and GNN. The objective function of these methods typically integrates multiple data types and estimates survival

simultaneously (middle or mixed integration). This function can be customized to adapt both continuous-time and discrete-time modeling.

SALMON first uses the package lmQCM [93] to identify co-expression modules (mRNA and micro RNA) and then applies SVD [94] to obtain *eigengenes*. Next, SALMON projects matrices of eigengenes to a latent space using perceptron layers. These latent vectors are concatenated with other data types (mutation, clinical data) and are passed through the output layer to compute the survival outcome. To train the model, SALMON uses the Adam optimization algorithm [95] to optimize the negative partial log-likelihood loss function with traditional LASSO regularization.

GDP employs an MLP model that encompasses an input layer, two hidden layers, and an output layer for survival modeling. Users need to provide a vector of group structure of the features, in which each component specifies the group that a feature is assigned to. The loss function is composed of negative partial log-likelihood and a modified LASSO term, in which the group LASSO is applied for the first weight matrix (between input layer and the first hidden layer) and standard LASSO is used for remaining weight matrices. GDP utilizes the Mini-batch Gradient Descent algorithm [96] to minimize the total loss during training.

Similar to GDP, SurvivalNet utilizes a simple MLP with partial log-likelihood objective function for survival prediction. The method first performs data filtering on mutation and copy number variation data using MutSig2CV [97], GISTIC [98], and Sanger Cancer Gene Census [99]. Next, it standardizes all features using z-score transformation and then concatenates them into a single matrix. SurvivalNet incorporates the Bayes Optimization algorithm [100], enabling users to effectively search for the optimal neural network architecture (e.g. number of hidden layers, layer width, activation function) and training configuration (e.g. learning rate, dropout fraction).

MiNet employs a four-layer MLP encoder to embed multi-omics data into a low-dimensional vector, which is concatenated with clinical data to predict patient survival outcomes using a CPH model. MiNet also integrates KEGG and Reactome pathways into their model by introducing a gene layer and a pathway layer within the MLP encoder. During training, MiNet optimizes the MLP weights and CPH coefficients of the model using the log partial likelihood loss function with Ridge regularization and Adam algorithm.

TF-Loghazard Net & TF-ESN aggregate CNV, DNA methylation, and gene expression within each gene into a single value and then merge them into a 3D tensor. They apply canonical decomposition/parallel factors (CP) [101] to decompose the tensor into three latent matrices before concatenating them with clinical data to train DNNs for survival prediction. TF-Loghazard Net applies Logistic-Hazard - a discrete-time survival model [102], utilizing Adam optimization scheme and a modified binary cross-entropy loss which accounts for censoring information of patients. TF-ESN further encodes the concatenated matrix into a lower-dimensional matrix and then passes the obtained matrix to a six-layer MLP. For training its DNN, TF-ESN utilizes Adam optimization and a loss function that is a weighted sum of the loss applied by TF-Loghazard Net and mean squared error.

SAE and CSAE both employ a neural network architecture that consists of an MLP encoder, an MLP decoder, and an MLP network. CSAE also introduces a concrete selection layer in its encoder part [103]. The model applies a reparametrization trick, allowing these weights to be trained along with other weights of the network. To strengthen the ability of the hidden representation to predict survival outcomes, SAE and CSAE utilize a combined loss

function that includes an unsupervised reconstruction loss for AE, a supervised partial log-likelihood loss for survival prediction and a Ridge regularization term.

OmiEmbed supports three different analysis tasks: survival prediction, numerical clinical feature prediction, and tumor type classification. The method allows users to perform one specific task or all of them synchronously. The method trains a DNN that encompasses a VAE for data integration and an MLP. The training process consists of three phases: (i) the method trains only the VAE; (ii) the method trains only the MLP; and (iii) both the deep VAE and MLP are trained collectively. The training loss function is tailored to each training phase, including only the losses corresponding to the trained portions.

CustOmics combines different DNN architectures into a single model, introducing a new training framework with two phases to optimize the use of these architectures. In the first phase, each data type is processed through a DNN consisting of an autoencoder and different MLPs to derive the optimal latent representation for the selected tasks. In the second phase, the latent representations from multiple data types are concatenated and input into central DNN comprising a VAE and different MLPs corresponding to the given tasks. The VAE extracts the common latent representation from the combined data, which is then used to train the MLPs for predicting the desired outcomes.

FGCNSurv uses the DNN architecture that consists of two encoding MLPs, an FBM [104], and a three-layer GCN. Each of the encoding MLPs encompasses one input layer, one hidden layer, and one highway network [105]. The method performs its training process as follows: encoding MLPs and FBM are utilized for integrating input data into a fused matrix; the average adjacency matrix is normalized following the technique proposed by Kipf and Welling [106] to prevent the problem of exploding/vanishing gradients; the method inputs the fused and normalized adjacency matrices to the GCN for estimating the hazard ratio; negative partial log-likelihood is calculated as the training loss and weights of the DNN are updated leveraging Adam optimization algorithm to minimize the loss.

MultimodalSurvivalPrediction employs a DNN architecture that consists of three modules: unsupervised learning for data representation, attention-based multimodal fusion, and survival prediction. The method uses MLP to encode each input data type into a distinct representation vector. The attention-based multimodal fusion module then leverages different perceptron layers to generate an attention vector for each data type. A unified representation vector is subsequently computed for the survival prediction task. The overall network is trained by concurrently minimizing this similarity loss as well as negative partial log-likelihood and cross-entropy loss associated with the survival prediction task. This dual-loss training strategy enhances the overall performance in survival prediction.

Overall, deep neural network approaches are efficient at handling large and complex datasets, and they are capable of capturing nonlinear covariate effects on the survival of cancer patients. However, these methodologies are computationally expensive and, in many cases, lack interpretability regarding the impact of specific covariates on survival. Overfitting can also be a significant issue, particularly when the training dataset is too small and does not contain enough samples to accurately represent all possible input values. Furthermore, many methods in this category, such as SALMON, GDP, MiNet, SurvivalNet, SAE, and CSAE, rely on the proportional hazards assumption of the CPH model, which presumes time-constant covariate effects on the hazard ratio.

Ensemble learning

There are four methods in this category: I-Boost [33], Priority-Lasso [107], blockForest [108], TCGA-omics-integration [109], and MDNNMD [32]. These methods use ensemble learning that combines multiple prediction models. Each of these models is trained on specific data types, on blocks of related covariates (e.g. genes and proteins in related biological pathways), on multiple subsamples, or bootstrapped samples of the training data. Each model considers different aspects of the same problem, and combining multiple models can potentially result in a more robust and accurate prediction model. Various ensemble learning approaches, such as boosting and random forests, are introduced and refined within this category.

I-Boost uses a CPH model to estimate the impact of multi-omics and clinical covariates on patient hazard ratios. This method employs a boosting approach to determine the coefficients of these covariates within the prediction model. For each iteration, the method first calculates the partial likelihood loss using the overall CPH model with its current coefficients. I-Boost then fits a CPH with elastic net regularization on each data type separately, calculating the loss values. The data type and corresponding CPH model yielding the greatest decrease in the loss function are then selected. Coefficients of this model are then used to update the coefficients in the overall CPH model.

Priority-Lasso sequentially ensembles multiple regression models built for each data type (linear regression, logistic regression, or CPH). Users are required to predefine a block structure for the input data, where each block can represent a data type or a set of features within a data type. Priority-Lasso consecutively fits a Lasso regression model to each data block, starting from the highest priority to the lowest one. Each subsequent model is used to refine the predictions of the previous ones. In the inference phase, the final prediction is obtained by summing the estimations of all fitted models given the corresponding blocks.

blockForest concatenates all input data types and treats features from each data type as distinct blocks. The method applies different parameters (such as sampling probability or weight) to each block of data, which affects the likelihood of a particular feature in a specific type being selected when training a subtree. This approach prevents bias toward data types with a high number of features and allows prioritizing input types based on their respective levels of predictive information. blockForest offers five variants of the random forest algorithm for training and prediction: VarProb, SplitWeights, BlockVarSel, RandomBlock, and BlockForest.

TCGA-omics-integration applies MAML [110], which optimizes model weights for adaptability across multiple tasks. The training process consists of a task-specific adaptation, and a meta-optimization. In the first step, a copy of the model's parameters is created for each task. SGD is then used to optimize these parameters on the task-specific support data, adapting the model for that task. Each adapted model is subsequently evaluated on the corresponding task-specific query data. In the second step, the losses obtained from applying the task-specific adapted models on the respective query data in the inner loop are combined into a meta-loss. Adam optimization algorithm is then used to compute the gradients of this meta-loss with respect to the initial model parameters and update these parameters to enhance the model's ability to adapt to multiple tasks simultaneously.

MDNNMD predicts binary survival outcomes by integrating predictions from multiple MLPs, each trained on a type of data. Each model consists of an input layer, four hidden layers, and an output layer. During training, batch normalization is applied to each hidden layer, and dropout is incorporated before the output layer. Each model generates a probability-based prediction of the vital status, which are then combined using a weighted linear aggregation function [111] to produce the final predicted survival outcome for the patient.

Overall, ensemble approaches can leverage the prediction power of complementary models. These approaches are not dependent on specific omics types and they provide the flexibility of adding new molecular types to the analysis. However, methods in this category (except for blockForest) follow the time-constant assumption of the covariate effects on the survival of patients, which might be violated in practice. There are also specific shortcomings of certain ensemble-learning-based approaches. I-Boost implements an iterative process of selecting data type for training a CPH model, which is usually computationally expensive. MDNNMD applies a simple aggregation strategy on predicted values obtained from its trained MLPs, which makes the method prone to overfitting.

Summary and practical guideline

Table 2 summarizes the key characteristics of the surveyed methods: main assumption, input, handling of missing data, output, integration strategy, modeling, validation data, and validation metric. Most methods follow either continuous-time modeling strategy or discrete-time strategy and formulate their models in one of three forms: parametric, semi-parametric, and nonparametric. MultimodalSurvivalPrediction is the only method that simultaneously trains two different models for predicting the hazard ratio (continuous-time + semi-parametric model) and probability of the vital status (discrete-time + parametric model) of patients. The availability of the methods (software link, documentation & user guides, programming language, publication year) are shown in Supplementary Table S1.

To further assist readers to choose a suitable method, we provide a general guideline as illustrated in Fig. 4. To begin with, the input of all surveyed methods includes multi-omics data and survival information of patients. If users do not have clinical variables or they want to investigate the impacts of only molecular features on the survival of patients, FGCNSurv might be a good option. Otherwise, they might select one of the remaining tools for their analysis based on the specific survival outcome that they want to predict (hazard ratio, vital status with probability, survival probability, survival time, and cumulative hazard).

Figure 4 also evaluates 20 methods utilizing six different metrics: (i) tutorial, (ii) documentation, (iii) case-study, (iv) installation, (v) user-friendliness, and (vi) accuracy. For each metric, a method is given a score from one (worst) to five (best). The overall score is equally weighted among the three criteria: (1) how accurate the method is (average of C-Index, td-C-Index, IBS, and D-Calibration [112]), (2) how well the method is documented (average of Tutorial, Documentation, and Case-Study), and (3) how reliable the implementation is (average of Installation and User-Friendliness). There are 12 methods with an overall score at least 3.5. These encompass Multimodal_NSCLC, GDP, SurvivalNet, CSAE, CustOmics, I-Boost, TCGA-omics-integration, OmiEmbed, IPF-LASSO, Priority-Lasso, blockForest, and MultimodalSurvivalPrediction. Among these, four are standalone packages (I-Boost, IPF-LASSO, blockForest, and PriorityLasso). The remaining eight methods are available as scripts on GitHub with a README file that provides the instruction for installation and execution. Details of TCGA data and analysis results can be found in Supplementary Note and Tables S2, S3, and S4.

Table 2. Summary of the survival prediction methods, including assumption, input (G: genomics, T: transcriptomics, N: noncoding RNA, E: epigenomics, P: proteomics, C: clinical), handling missing data (KNN: K-nearest neighbor imputation, median/mean: replacing missing values with median or mean, remove: delete records with missing values, zero: replace missing values with zero, NA: not addressed), integration strategy, output (ST: survival time, SP: survival probability, VP: vital status with probability, HR: hazard ratio, CH: cumulative hazard), and validation.

Method	Assumption	Input	Missing data handling	Integration strategy	Modeling strategy	Output	Validation Data	Metric
Regularized linear regression								
IPF-LASSO [77]	time-constant covariate effects	G, T, C	NA	Early	continuous-time/discrete-time, parametric/semi-parametric	ST/VP/HR	TCGA, GEO	IBS
M2EFM [29]	time-constant covariate effects	T, E, C	KNN	Early	continuous-time, semi-parametric	HR	TCGA	C-Index, Log-Rank, Moderate Calibration
Multimodal_NSCLC [78]	time-constant covariate effects	T, N, E, C	median	Middle	continuous-time, semi-parametric	HR	TCGA	C-Index
Deep neural network								
SALMON [31]	time-constant covariate effects	G, T, N, C	NA	Early	continuous-time, semi-parametric	HR	TCGA	C-Index, Log-Rank
GDP [84]	time-constant covariate effects	G, T, P, C	mean	Early	continuous-time, semi-parametric	HR	TCGA	C-Index
MiNet [85]	time-constant covariate effects	G, T, E, C	KNN	Early	continuous-time, semi-parametric	HR	TCGA	C-Index, Log-Rank
SurvivalNet [86]	time-constant covariate effects	G, T, P, C	KNN	Early	continuous-time, semi-parametric	HR	TCGA	C-Index
TF-Loghazard Net [87]	time-varying covariate effects	G, T, E, C	NA	Middle	discrete-time, parametric	SP	TCGA	tdC-Index
TF-ESN [87]	time-varying covariate effects	G, T, E, C	NA	Middle	discrete-time, parametric	SP	TCGA	tdC-Index
SAE [88]	time-constant covariate effects	G, T, N, E, P, C	remove	Middle	continuous-time, semi-parametric	HR	TCGA	C-Index, Log-Rank
CSAE [88]	time-constant covariate effects	G, T, N, E, P, C	remove	Middle	continuous-time, semi-parametric	HR	TCGA	C-Index, Log-Rank
OmiEmbed [89]	time-varying covariate effects	T, N, E, C	mean	Middle	discrete-time, parametric	SP	TCGA, TARGET, GEO	C-Index, IBS
CustOmics [90]	time-constant covariate effects	G, T, E, C	remove	Mixed	continuous-time, semi-parametric	HR	TCGA	C-Index, Log-Rank, IBS
MultimodalSurvival-Prediction [91]	time-constant covariate effects	G, T, N, C	zero	Mixed	continuous-time, discrete-time, semi-parametric, parametric	HR+VP	TCGA	C-Index
FGCNSurv [92]	time-constant covariate effects	T, N	KNN	Middle	continuous-time, semi-parametric	HR	TCGA, PCAWG	C-Index, Log-Rank, AUC
Ensemble learning								
I-Boost [33]	time-constant covariate effects	G, T, N, P, C	NA	Early	continuous-time, semi-parametric	HR	TCGA	C-Index
Priority-Lasso [107]	time-constant covariate effects	G, T, C	remove	Early	continuous-time/discrete-time, parametric/semi-parametric	ST/VP/HR	GEO	C-Index, AUC, IBS, Moderate Calibration
blockForest [108]	covariate effects not considered	G, T, N, C	zero	Early	continuous-time/discrete-time, nonparametric	ST/VP/ CH+SP	TCGA	C-Index
TCGA-omics-integration [109]	time-constant covariate effects	T, P, C	KNN	Early	continuous-time, semi-parametric	HR	TCGA	C-Index, IBS
MDNNMD [32]	time-constant covariate effects	G, T, C	KNN	Late	discrete-time, parametric	VP	METABRIC	AUC, Log-Rank

	Tutorial	Documentation	Case-Study	Installation	User-Friendliness	Accuracy	Overall Score
FGCNSurv	3	1	3	4	4	2	2.78
M2EFM	5	5	5	2	4	2	3.33
Multimodal_NSCLC	5	4	2	3	3	4	3.56
SALMON	3	1	2	4	4	3	3.00
GDP	5	5	3	4	4	3	3.78
MiNet	3	1	2	2	2	3	2.33
SurvivalNet	5	5	2	5	4	2	3.50
SAE	5	5	3	4	4	2	3.44
CSAE	5	5	3	4	4	5	4.44
CustOmics	5	5	3	3	4	3	3.61
I-Boost	5	5	3	4	5	4	4.28
TCGA-omics-integration	1	5	3	3	4	4	3.50
MDNNMD	5	3	2	4	4	2	3.11
TF-LogHazard Net	1	3	2	1	1	3	2.00
TF-ESN	1	3	2	1	1	3	2.00
OmiEmbed	5	5	3	4	4	3	3.78
IPF-LASSO	5	5	2	5	3	4	4.00
Priority-Lasso	5	5	4	5	5	4	4.56
blockForest	5	5	3	5	5	5	4.78
MultimodalSurvival-Prediction	5	5	3	4	4	3	3.78

(**HR**: Hazard ratio, **VP**: Vital status with probability, **SP**: Survival probability, **ST**: Survival time, **CH**: Cumulative hazard)
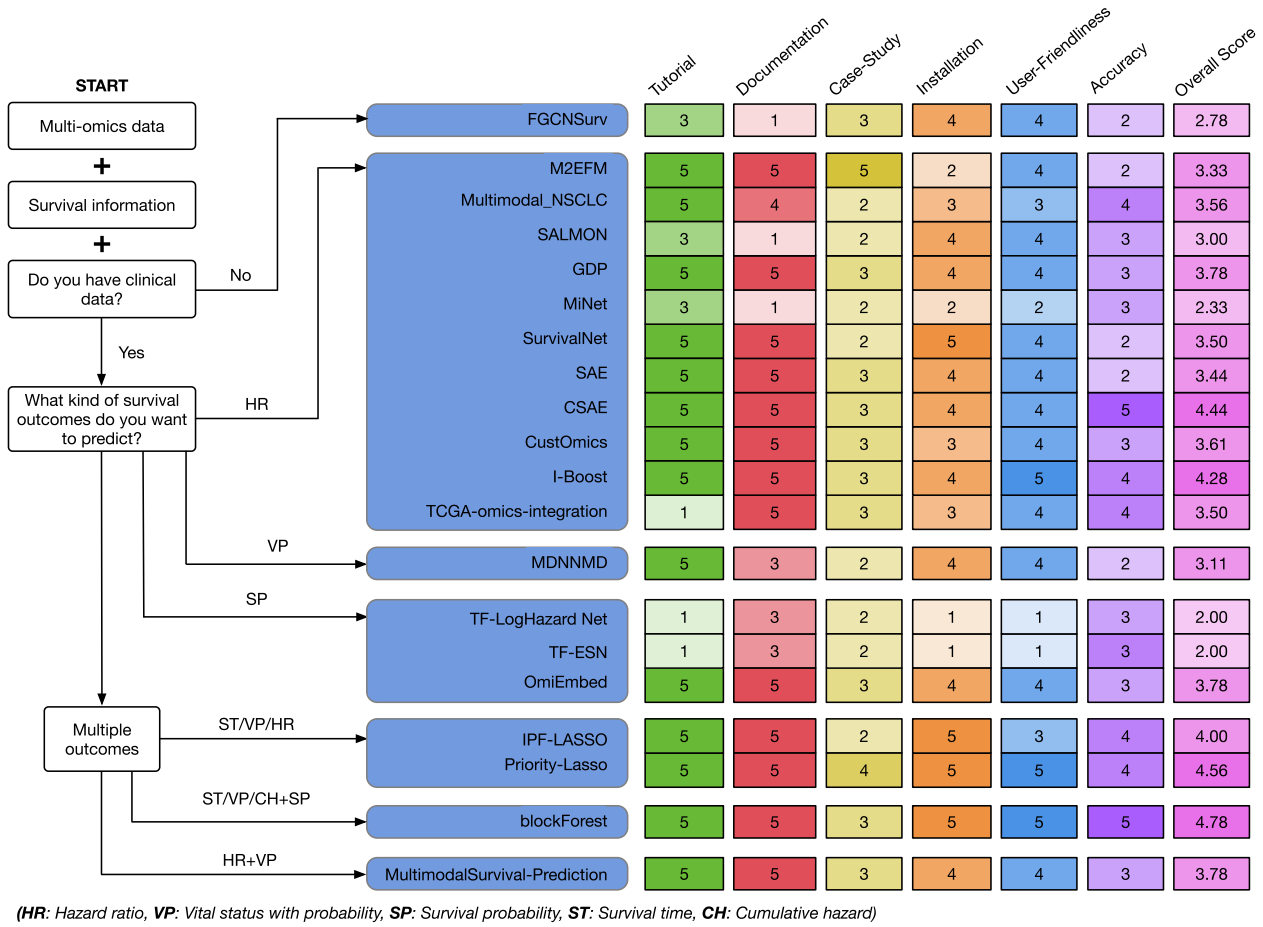
Figure 4. Method guideline and assessment scores (tutorial, documentation, case-study presentation, installation, user-friendliness, accuracy, and overall score).

First, the tutorial metric demonstrates whether a detailed, comprehensible tutorial is provided for each surveyed tool. We assess the tutorial provided for each method according to all phases of a general analysis including data importing, data processing, model training, and survival prediction. Most of the reviewed methods have a high score in this metric.

Second, the documentation metric evaluates the quality of documentation for each method in terms of fundamental functions and parameters. Generally, methods with a software package (M2EFM, I-Boost, IPF-LASSO, Priority-Lasso, and blockForest) receive the highest score. GDP, SurvivalNet, SAE, CSAE, CustOmics also provide an exemplary and exhaustive documentation.

Third, the case-study metric assess each approach according to the validation technique presented in their paper. This metric is composed of two sub-criteria: (i) number of case studies using real datasets reported, and (ii) adaptation of external validation. A method earns three points if the corresponding paper presents at least three case studies involving high-quality, real datasets. Lower number of case studies reported leads to a deduction in this score. Moreover, if a method is validated using independent datasets, it is given two more points. M2EFM is the only tool that receives the maximum five points for the validation metric.

Fourth, the installation metric refers to how straightforward it is to install the software. We subtract points from the maximum score for a method if it either (i) does not provide a clear, adequate installation guidance (e.g. listing programming languages, dependency packages and their required versions as well as describing

how to set up programming environment, install dependencies, etc.), or (ii) requires manual installation of many dependencies, which can be challenging and time-consuming to users. Only a few methods (SurvivalNet, IPF-LASSO, Priority-Lasso, and block-Forest) receive the highest score for this metric.

Fifth, the user-friendliness metric refers to the ease and convenience of using each method from a practical standpoint. Specifically, this metric assesses how easy it is for users to perform analysis on example and new datasets utilizing each tool. Higher scores are given to methods that can be run using simple commands/functions. At the same time, a tool will be deducted some points for the fourth metric if it requires users to manually perform any step of the analysis pipeline other than data importing. I-Boost, PriorityLasso, and blockForest stand out as the most user-friendly methods.

Sixth, the accuracy score assesses how accurate each method is in predicting patient survival. For this purpose, we perform cross-validation of 17 TCGA datasets using four different metrics: Harrell's C-Index, td-C-Index, IBS, and D-Calibration (see Supplementary Note and Tables S2, S3, and S4). For each metric, we calculate the average score value across all datasets, and scale all values to a score between one (worst) and five (best). The final accuracy score is calculated as the average of the four scores. Overall, CSAE and blockForest have the highest accuracy with a score of 5. Multimodal_NSCLC, I-Boost, TCGA-omics-inegration, IPF-LASSO, and Priority-Lasso are also among the top-performing methods with an accuracy score of 4.

In our analysis, each type of multi-omics data has 5000 features, making the total number of features more than 10 times the number of samples in each dataset. All 20 methods of the three method categories (regularized linear regression, deep neural networks, and ensemble learning) are able to analyze the 17 datasets without crashing. In other words, all methods allow the number of covariates to be larger than the sample size. Regarding accuracy, each category has methods that achieve high accuracy. We would conclude that the performance of a method depends on the method details and implementation, rather than method category.

Outstanding challenges
Multi-omics integration
By incorporating different types of molecular data, researchers can better understand the complex progression of cancer, which typically involves the coordinated activities of multiple omics layers. For example, it has been reported that the integration of proteomics with genomics and transcriptomics allows for the identification of potential biomarkers that drive cancer progression after primary treatment for colon, rectal, and ovarian cancer [113, 114]. Utilizing prognostic molecular features, along with factors such as cell state, cell location, microenvironmental, and clinical information, has led to increased accuracy of survival prediction models, as reported in numerous studies [115–120]. Moreover, the use of multi-omics data boosts the reliability of survival prediction results. The combination of different molecular types allows researchers to increase the sample size of the study via taking a union of samples across omics types. Both of these factors can substantially enhance the statistical robustness and confidence level of the analysis results [121–123].

However, it is important to note that it is not always beneficial to add more omics types to the model. Integrating more omics types can add noise, redundancy, and inverse relationship among specific omics types to the model, all of which may undermine the model's performance [124, 125]. The authors of I-Boost and Multimodal_NSCLC validated their models using different combinations of multi-omics and clinical data across multiple datasets. Both studies reported that incorporating all data types did not derive the best result of their models. Therefore, the selection of omics types and integration approach should be made regarding important factors encompassing heterogeneity and interrelationships among different omics data. Intensive data analysis and expert knowledge are necessary to obtain the optimal subsets of omics types for the integrative analysis.

Moreover, the task of multi-omics integration is not straightforward and usually demands considerable effort. The heterogeneity nature of multi-omics data, inconsistency in data processing, and unknown interactions among omics layers contribute to prediction instability [126]. Therefore, it is fundamental to ensure the consistency among input omics data regarding applied assays and experiment designs, as well as data processing protocols. Besides that, in practice, the existence of missing data are common among omics datasets and this problem exacerbates when more omics types are added to the analysis [127]. To address these issues, the surveyed methods use various strategies in their integrative analysis, although many limitations remain, as discussed below.

First, prediction methods intersect samples among the different molecular and clinical data types during training process, i.e. the methods only keep patients that have all types of multi-omics data and clinical variables. Intersection can result in the loss of important observations, whereas imputing the entire representation of an omics type for certain patients could introduce bias into the analysis. Since multi-omics data are not always available for all patients, it would be beneficial to adopt transfer learning-based and multi-view matrix factorization-based imputation. These techniques can account for the interplay between the omics data containing missing values and other molecular types [127–129].

Second, current methods exhibit limitations in their data integration techniques. Half of the methods in this review perform early integration for their input data types. Many studies have demonstrated the superiority of other integration strategies over early integration, especially in case of great heterogeneity among data types [130–132]. Deep-learning-based architectures are capable of extracting meaningful latent variables out of multi-omics data. Researchers have proposed a mixed-integration deep-learning design, which comprises of different sub-networks for learning independent features from each omics and a central network for integrating these features [130, 133]. However, there is a lack of interpretability when using deep learning models, which is important to gain insights into the biology for prognosis and treatment.

Modeling
In terms of modeling, there are numerous limitations that need to be addressed. First, prediction approaches neglect biological knowledge in their models. For example, DNA methylation is known to repress gene expression but current methods fail to take into account such interactions [134]. In addition, the relationship among genes and gene products residing within pathways can provide valuable insights for studies regarding the mechanism of cancer. Pathways such as PI3K/AKT/mTOR and Ras/MAPK have been reported to play critical roles in the development of tumors [135]. MiNet is the only method that attempts to incorporate gene set information into their DNN layers by connecting gene nodes with pathway nodes. The addition of biological knowledge to the data integration step, via novel techniques including hierarchical integration or network-based integration, can lead to improvement in performance of survival prediction models [136, 137].

Second, most of the methods favor semi-parametric modeling, i.e., most of them use CPH models, which estimate hazard ratios of cancer patients. It would be more beneficial if prognostic models can predict survival probability (or probability of other events) for a patient within a specified time period [138]. Currently, users need to perform additional steps of estimating patient survival using techniques such as Breslow estimator [75]. The use of parametric models can directly provide information about the survival of patients across time, which is one advantage over semi-parametric approaches in real-world applications. However, the efficacy of parametric models depends on the accuracy of their assumptions about survival time distribution, which can be challenging to ascertain. To address this issue, an ensemble strategy can be applied, in which various models with different formulations of survival time are fit to the same training data and the final predicted outcome is obtained by combining results from all trained models. This can help mitigate the reliance of the model on distribution assumptions to generate more robust predictions.

Third, the majority of the reviewed approaches follow the time-constant assumption of covariate effects on survival response variables. The violation of these assumptions may occur frequently in real-world scenarios, which leads to less accurate prediction [71, 80–83]. Also, methods that follow the time-constant covariate effect assumption would use the same function and

parameters for all time intervals, preventing them from incorporating time-dependent covariates such as age, weight, or smoking status. OmiEmbed, TF-Loghazard Net, and TF-ESN address this issue by a discrete-time prediction framework in which the time-varying covariate effects are formulated and time-dependent features can be added to the prediction model. However, generating time-varying omics-data is computationally expensive and heavily affected by many experimental factors. Future studies should investigate different nonparametric models in survival prediction, which do not rely on specific assumptions about the covariate effects.

Validation

Most methods use cross-validation for validation. Cross-validation might produce inflated accuracy, fostering an overoptimistic view about the efficacy of the proposed methods [139, 140]. It is essential to assess prediction methods using independent external data before applying them in a real-world setting. M2EFM and Priority-Lasso are the only two approaches that validated their models using independent datasets. Specifically, M2EFM trained their model on TCGA data (RNA-Seq and clinical variables) and tested on GSE39004 and GSE20685 (Affymetrix microarray and clinical variables). Priority-Lasso trained their model on AMLCG-1999 trial data [141] (Affymetrix microarray, gene mutation status, European LeukemiaNet genetic risk stratification score [142], and clinical variables). The model was subsequently validated on an independent dataset consisting of patients from AMLCG-2008 [143] and 40 patients from the AMLCG-1999 trial. These methods effectively simulate real-world scenarios where training and testing set were generated from independent sources using different assaying platforms. However, there exist major issues that become discernible to researchers solely upon the undertaking of external validation studies. For example, problems such as the missing of one or multiple omics types for new patients, disparities between the training data and external data (in terms of assaying platforms, underlying distribution, etc.) can undermine the prediction capability of the developed models.

In terms of metrics, most methods focus on discrimination power of their models but neglecting calibration aspects. This is indicated by the fact that 12 of the surveyed approaches only output hazard ratios, which are mainly designed to demonstrate the relative risk among patients. Poorly calibrated estimates often result in erroneous expectations with cancer patients and healthcare specialists [62]. In addition, while all of the reviewed methods use at least one discrimination metrics (among C-Index, AUC, log-Rank test), only a few involve calibration measures (moderate calibration) or overall performance metrics (e.g. IBS) in their validation. Moreover, the methods utilizing calibration measures do not provide codes for reproducing the results or explain in detail how calibration metrics/graphs are constructed. There are a variety of techniques for calibration measurements that have been proposed in previous studies even regarding one specific level of calibration [60, 138, 144]. Thus, users might have difficulties selecting suitable techniques to measure the calibration quality of their models.

Finally, it should be highlighted that the accuracy of all analysis methods is highly dependent on the quality of processed data. Most of the reviewed methods provide preprocessed data without including the data processing step in the source code/instruction files. Also, many approaches in this review utilize a mixture of external tools for the step of data processing. Exclusion of detailed guidance on the data processing procedure presents obvious impediments for users seeking to efficiently leverage the methods.

Conclusion

We review 20 methods for survival prediction using multi-omics data. We explain the basic building blocks of survival prediction approaches, discussing their model and validation strategies. Our main goals are to assist potential users, especially life science, biomedical, and clinical scientists, in choosing the most appropriate method for their analysis, as well as support computational scientists in developing novel methodologies that successfully address the current drawbacks.

We also discuss the outstanding challenges related to multi-omics integration, modeling, and validation. Outstanding issues in multi-omics integration include missing data types (i.e. not all patients have all types of data), inconsistent preprocessing, and suboptimal integration. Challenges in modeling include lack of integration of pathway topology and biological knowledge into the prediction models, and overreliance on semi-parametric models and strong assumptions of time-constant covariate effects. For validation, most approaches use cross-validation, which is prone to overfitting. These challenges need to be addressed in future development.

Key Points

- Accurate prediction of patient outcomes is pivotal in oncology research and treatment.
- Multi-omics integration can leverage the complementary information available in multiple types of data to improve the robustness of prediction models.
- This article provides a comprehensive review of 20 survival prediction approaches leveraging multi-omics data.
- This article discusses underlying assumptions, input and output, integration, modeling, and validation techniques of the survival prediction methods.
- This paper presents outstanding challenges in the field that remain unaddressed and need to be solved.

Supplementary data

Supplementary data is available at *Briefings in Bioinformatics* online.

Conflict of interest: None declared.

Funding

Data availability

The TCGA data used for method evaluation can be found at GDC data portal (https://portal.gdc.cancer.gov/). Source code for

analysis is available at https://github.com/tinnlab/Risk-Review-Benchmark.

References

1. Tate JG, Bamford S, Jubb HC. *et al.* COSMIC: The catalogue of somatic mutations in cancer. *Nucleic Acids Res* 2019;**47**:D941–7. https://doi.org/10.1093/nar/gky1015

2. Finotello F, Eduati F. Multi-omics profiling of the tumor microenvironment: Paving the way to precision immuno-oncology. *Front Oncol* 2018;**8**:430. https://doi.org/10.3389/fonc.2018.00430

3. Sammut S-J, Crispin-Ortuzar M, Chin S-F. *et al.* Multi-omic machine learning predictor of breast cancer therapy response. *Nature* 2022;**601**:623–9. https://doi.org/10.1038/s41586-021-04278-5

4. Grossman RL, Heath AP, Ferretti V. *et al.* Toward a shared vision for cancer genomic data. *N Engl J Med* 2016;**375**:1109–12. https://doi.org/10.1056/NEJMp1607591

5. Barretina J, Caponigro G, Stransky N. *et al.* The cancer cell line encyclopedia enables predictive modelling of anti-cancer drug sensitivity. *Nature* 2012;**483**:603–7. https://doi.org/10.1038/nature11003

6. Ghandi M, Huang FW, Jané-Valbuena J. *et al.* Next-generation characterization of the cancer cell line encyclopedia. *Nature* 2019;**569**:503–8.

7. Nusinow DP, Szpyt J, Ghandi M. *et al.* Quantitative proteomics of the cancer cell line encyclopedia. *Cell* 2020;**180**:387–402.e16. https://doi.org/10.1016/j.cell.2019.12.023

8. de Anda-Jáuregui G, Hernández-Lemus E. Computational oncology in the multi-omics era: State of the art. *Front Oncol* 2020;**10**:423. https://doi.org/10.3389/fonc.2020.00423

9. Menyhárt O, Győrffy B. Multi-omics approaches in cancer research with applications in tumor subtyping, prognosis, and diagnosis. *Comput Struct Biotechnol J* 2021;**19**:949–60. https://doi.org/10.1016/j.csbj.2021.01.009

10. Karaman ED, Işik Z. Multi-omics data analysis identifies prognostic biomarkers across cancers. *Med Sci* 2023;**11**:44. https://doi.org/10.3390/medsci11030044

11. Granja JM, Klemm S, McGinnis LM. *et al.* Single-cell multiomic analysis identifies regulatory programs in mixed-phenotype acute leukemia. *Nat Biotechnol* 2019;**37**:1458–65. https://doi.org/10.1038/s41587-019-0332-7

12. Kan Z, Ding Y, Kim J. *et al.* Multi-omics profiling of younger Asian breast cancers reveals distinctive molecular signatures. *Nat Commun* 2018;**9**:1725. https://doi.org/10.1038/s41467-018-04129-4

13. Jiang L, Chao X, Bai Y. *et al.* Autosurv: Interpretable deep learning framework for cancer survival analysis incorporating clinical and multi-omics data. *npj Precis Oncol* 2024;**8**:4.

14. Chaudhary K, Poirion OB, Liangqun L. *et al.* Deep learning–based multi-omics integration robustly predicts survival in liver cancer. *Clin Cancer Res* 2018;**24**:1248–59. https://doi.org/10.1158/1078-0432.CCR-17-0853

15. Chen B, Wei W, Ma L. *et al.* Computational discovery of niclosamide ethanolamine, a repurposed drug candidate that reduces growth of hepatocellular carcinoma cells in vitro and in mice by inhibiting cell division cycle 37 signaling. *Gastroenterology* 2017;**152**:2022–36. https://doi.org/10.1053/j.gastro.2017.02.039

16. Asada K, Kobayashi K, Joutard S. *et al.* Uncovering prognosis-related genes and pathways by multi-omics analysis in lung cancer. *Biomolecules* 2020;**10**:524. https://doi.org/10.3390/biom10040524

17. Alejandro Balbin O, Prensner JR, Sahu A. *et al.* Reconstructing targetable pathways in lung cancer by integrating diverse omics data. *Nat Commun* 2013;**4**:1–13. https://doi.org/10.1038/ncomms3617

18. Bailey P, Chang DK, Nones K. *et al.* Genomic analyses identify molecular subtypes of pancreatic cancer. *Nature* 2016;**531**:47–52.

19. Zhou DC, Jayasinghe RG, Chen S. *et al.* Spatially restricted drivers and transitional cell populations cooperate with the microenvironment in untreated and chemo-resistant pancreatic cancer. *Nat Genet* 2022;**54**:1390–405.

20. Schulte-Sasse R, Budach S, Hnisz D. *et al.* Integration of multiomics data with graph convolutional networks to identify new cancer genes and their associated molecular mechanisms. *Nat Mach Intell* 2021;**3**:513–26. https://doi.org/10.1038/s42256-021-00325-y

21. Migliozzi S, Young Taek O, Hasanain M. *et al.* Integrative multi-omics networks identify PKCδ and DNA-PK as master kinases of glioblastoma subtypes and guide targeted cancer therapy. *Nature Cancer* 2023;**4**:181–202. https://doi.org/10.1038/s43018-022-00510-x

22. Charoentong P, Finotello F, Angelova M. *et al.* Pan-cancer immunogenomic analyses reveal genotype-immunophenotype relationships and predictors of response to checkpoint blockade. *Cell Rep* 2017;**18**:248–62. https://doi.org/10.1016/j.celrep.2016.12.019

23. Singh A, Shannon CP, Gautier B. *et al.* DIABLO: An integrative approach for identifying key molecular drivers from multi-omics assays. *Bioinformatics* 2019;**35**:3055–62. https://doi.org/10.1093/bioinformatics/bty1054

24. Liu K, Newbury PA, Glicksberg BS. *et al.* Evaluating cell lines as models for metastatic breast cancer through integrative analysis of genomic data. *Nat Commun* 2019;**10**:2138. https://doi.org/10.1038/s41467-019-10148-6

25. Zhang X, Li Y, Akinyemiju T. *et al.* Pathway-structured predictive model for cancer survival prediction: A two-stage approach. *Genetics* 2017;**205**:89–100. https://doi.org/10.1534/genetics.116.189191

26. López-García G, Jerez JM, Franco L. *et al.* Transfer learning with convolutional neural networks for cancer survival prediction using gene-expression data. *PloS One* 2020;**15**:e0230536. https://doi.org/10.1371/journal.pone.0230536

27. Kim SY, Jeong H-H, Kim J. *et al.* Robust pathway-based multi-omics data integration using directed random walks for survival prediction in multiple cancer studies. *Biol Direct* 2019;**14**:8. https://doi.org/10.1186/s13062-019-0239-8

28. Lai Y-H, Chen W-N, Hsu T-C. *et al.* Overall survival prediction of non-small cell lung cancer by integrating microarray and clinical data with deep learning. *Sci Rep* 2020;**10**:4679. https://doi.org/10.1038/s41598-020-61588-w

29. Thompson JA, Christensen BC, Marsit CJ. Methylation-to-expression feature models of breast cancer accurately predict overall survival, distant-recurrence free survival, and pathologic complete response in multiple cohorts. *Sci Rep* 2018;**8**:5190. https://doi.org/10.1038/s41598-018-23494-0

30. Hao J, Kim Y, Mallavarapu T. *et al.* Interpretable deep neural network for cancer survival analysis by integrating genomic and clinical data. *BMC Med Genomics* 2019;**12**:189. https://doi.org/10.1186/s12920-019-0624-2

31. Huang Z, Zhan X, Xiang S. *et al*. SALMON: Survival analysis learning with multi-omics neural networks on breast cancer. *Front Genet* 2019;**10**:166.

32. Sun D, Wang M, Li A. A multimodal deep neural network for human breast cancer prognosis prediction by integrating multi-dimensional data. *IEEE/ACM Trans Comput Biol Bioinform* 2018;**16**:841–50. https://doi.org/10.1109/TCBB.2018.2806438

33. Wong KY, Fan C, Tanioka M. *et al*. I-boost: An integrative boosting approach for predicting survival time with multiple genomics platforms. *Genome Biol* 2019;**20**:52.

34. Yuan Y, Van Allen EM, Omberg L. *et al*. Assessing the clinical utility of cancer genomic and proteomic data across tumor types. *Nat Biotechnol* 2014;**32**:644–52. https://doi.org/10.1038/nbt.2940

35. Shedden K, Taylor JMG, Enkemann SA. *et al*. Gene expression–based survival prediction in lung adenocarcinoma: A multi-site, blinded validation study. *Nat Med* 2008;**14**:822–7.

36. Van Wieringen WN, Kun D, Hampel R. *et al*. Survival prediction using gene expression data: A review and comparison. *Computational Statistics & Data Analysis* 2009;**53**:1590–603. https://doi.org/10.1016/j.csda.2008.05.021

37. Strijker M, Chen JW, Mungroop TH. *et al*. Systematic review of clinical prediction models for survival after surgery for resectable pancreatic cancer. *Journal of British Surgery* 2019;**106**:342–54.

38. Bakasa W, Viriri S. Pancreatic cancer survival prediction: A survey of the state-of-the-art. *Comput Math Methods Med* 2021;**1–17**:2021.

39. Zhang Y, Wong G, Mann G. *et al*. Survbenchmark: Comprehensive benchmarking study of survival analysis methods using both omics data and clinical data. *GigaScience* 2022;**11**:giac071. https://doi.org/10.1093/gigascience/giac071

40. Bonato V, Baladandayuthapani V, Broom BM. *et al*. Bayesian ensemble methods for survival prediction in gene expression data. *Bioinformatics* 2011;**27**:359–67. https://doi.org/10.1093/bioinformatics/btq660

41. Shen S, Wang Y, Wang C. *et al*. SURVIV for survival analysis of mRNA isoform variation. *Nat Commun* 2016;**7**:11548.

42. Qiu YL, Zheng H, Devos A. *et al*. A meta-learning approach for genomic survival analysis. *Nat Commun* 2020;**11**:6350. https://doi.org/10.1038/s41467-020-20167-3

43. Ching T, Zhu X, Garmire LX. Cox-nnet: An artificial neural network method for prognosis prediction of high-throughput omics data. *PLoS Comput Biol* 2018;**14**:e1006076. https://doi.org/10.1371/journal.pcbi.1006076

44. Zhu X, Yao J, Huang J. Deep convolutional neural network for survival analysis with pathological images. In: Tian T (ed.), *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. Shenzhen: IEEE, 2016, 544–7.

45. Zhang Y, Lobo-Mueller EM, Karanicolas P. *et al*. Cnn-based survival model for pancreatic ductal adenocarcinoma in medical imaging. *BMC Med Imaging* 2020;**20**:1–8. https://doi.org/10.1186/s12880-020-0418-1

46. Sun L, Zhang S, Chen H. Brain tumor segmentation and survival prediction using multimodal mri scans with deep learning. *Front Neurosci* 2019;**13**:468066.

47. Hancock JT, Khoshgoftaar TM. Survey on categorical data for neural networks. *Journal of Big Data* 2020;**7**:28. https://doi.org/10.1186/s40537-020-00305-w

48. Dahouda MK, Joe I. A deep-learned embedding technique for categorical features encoding. *IEEE Access* 2021;**9**:114381–91. https://doi.org/10.1109/ACCESS.2021.3104357

49. Troyanskaya O, Cantor M, Sherlock G. *et al*. Missing value estimation methods for DNA microarrays. *Bioinformatics* 2001;**17**:520–5. https://doi.org/10.1093/bioinformatics/17.6.520

50. Leek JT, Evan Johnson W, Parker HS. *et al*. The sva package for removing batch effects and other unwanted variation in high-throughput experiments. *Bioinformatics* 2012;**28**:882–3. https://doi.org/10.1093/bioinformatics/bts034

51. Rappoport N, Shamir R. NEMO: Cancer subtyping by integration of partial multi-omic data. *Bioinformatics* 2019;**35**:3348–56. https://doi.org/10.1093/bioinformatics/btz058

52. Picard M, Scott-Boyer M-P, Bodein A. *et al*. Integration strategies of multi-omics data for machine learning analysis. *Comput Struct Biotechnol J* 2021;**19**:3735–46. https://doi.org/10.1016/j.csbj.2021.06.030

53. Shen R, Mo Q, Schultz N. *et al*. Integrative subtype discovery in glioblastoma using iCluster. *PLoS One* 2012;**7**:e35236. https://doi.org/10.1371/journal.pone.0035236

54. Hinton GE, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. *Science* 2006;**313**:504–7. https://doi.org/10.1126/science.1127647

55. Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. *Nature* 1986;**323**:533–6. https://doi.org/10.1038/323533a0

56. Antolini L, Boracchi P, Biganzoli E. A time-dependent discrimination index for survival data. *Stat Med* 2005;**24**:3927–44. https://doi.org/10.1002/sim.2427

57. Harrell FE, Califf RM, Pryor DB. *et al*. Evaluating the yield of medical tests. *JAMA* 1982;**247**:2543–6. https://doi.org/10.1001/jama.1982.03320430047030

58. Uno H, Cai T, Pencina MJ. *et al*. On the C-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data. *Stat Med* 2011;**30**:1105–17. https://doi.org/10.1002/sim.4154

59. Bewick V, Cheek L, Ball J. Statistics review 12: Survival analysis. *Crit Care* 2004;**8**:389–94. https://doi.org/10.1186/cc2955

60. McLernon DJ, Giardiello D, Van Calster B. *et al*. Steyerberg, topic groups 6, and 8 of the STRATOS initiative. Assessing performance and clinical usefulness in prediction models with survival outcomes: Practical guidance for Cox proportional hazards models. *Ann Intern Med* 2023;**176**:105–14. https://doi.org/10.7326/M22-0844

61. Van Calster B, Nieboer D, Vergouwe Y. *et al*. A calibration hierarchy for risk models was defined: From utopia to empirical data. *J Clin Epidemiol* 2016;**74**:167–76. https://doi.org/10.1016/j.jclinepi.2015.12.005

62. Van Calster B, McLernon DJ, Van Smeden M. *et al*. Topic group 'evaluating diagnostic tests, and prediction models' of the STRATOS initiative. Calibration: The achilles heel of predictive analytics. *BMC Med* 2019;**17**:230.

63. Kaplan EL, Meier P. Nonparametric estimation from incomplete observations. *J Am Stat Assoc* 1958;**53**:457–81. https://doi.org/10.1080/01621459.1958.10501452

64. Steyerberg EW, Vickers AJ, Cook NR. *et al*. Assessing the performance of prediction models: A framework for some traditional and novel measures. *Epidemiology* 2010;**21**:128–38. https://doi.org/10.1097/EDE.0b013e3181c30fb2

65. Park SY, Park JE, Kim H. *et al*. Review of statistical methods for evaluating the performance of survival or other time-to-event prediction models (from conventional to deep learning approaches). *Korean J Radiol* 2021;**22**:1697–707. https://doi.org/10.3348/kjr.2021.0223

66. Zhang Z, Reinikainen J, Adeleke KA. *et al*. Time-varying covariates and coefficients in Cox regression models. *Annals of*

Translational Medicine 2018;**6**:121–10. https://doi.org/10.21037/atm.2018.02.12

67. Cygu S, Seow H, Dushoff J. *et al.* Comparing machine learning approaches to incorporate time-varying covariates in predicting cancer survival time. *Sci Rep* 2023;**13**:1–10.

68. Kenneth, Tay J, Narasimhan B, Hastie T. Elastic net regularization paths for all generalized linear models. *J Stat Softw* 2023;**106**:1–31. https://doi.org/10.18637/jss.v106.i01

69. Nelson W. Theory and applications of hazard plotting for censored failure data. *Dent Tech* 1972;**14**:945–66. https://doi.org/10.1080/00401706.1972.10488991

70. Aalen O. Nonparametric inference for a family of counting processes. *The Annals of Statistics* 1978;**6**:701–26. https://doi.org/10.1214/aos/1176344247

71. Suresh K, Severn C, Ghosh D. Survival prediction models: An introduction to discrete-time modeling. *BMC Med Res Methodol* 2022;**22**:207. https://doi.org/10.1186/s12874-022-01679-6

72. Kleinbaum DG, Klein M. In: Gail M (ed.), *Survival Analysis: A Self-Learning Text*. New York: Springer, 2012.

73. Cox DR. Regression models and life-tables. *J R Stat Soc B Methodol* 1972;**34**:187–220.

74. Lin DY, Ying Z. Semiparametric analysis of the additive risk model. *Biometrika* 1994;**81**:61–71. https://doi.org/10.1093/biomet/81.1.61

75. Lin DY. On the Breslow estimator. *Lifetime Data Anal* 2007;**13**:471–80. https://doi.org/10.1007/s10985-007-9048-y

76. Turkson AJ, Ayiah-Mensah F, Nimoh V. Handling censoring and censored data in survival analysis: A standalone systematic literature review. *International Journal of Mathematics and Mathematical Sciences* 2021;**2021**:1–16.

77. Boulesteix A-L, De Bin R, Jiang X. *et al.* IPF-LASSO: Integrative-penalized regression with penalty factors for prediction based on multi-omics data. *Comput Math Methods Med* 2017;**2017**:1–14. https://doi.org/10.1155/2017/7691937

78. Ellen JG, Jacob E, Nikolaou N. *et al.* Autoencoder-based multimodal prediction of non-small cell lung cancer survival. *Sci Rep* 2023;**13**:15761. https://doi.org/10.1038/s41598-023-42365-x

79. Shabalin AA. Matrix eQTL: Ultra fast eQTL analysis via large matrix operations. *Bioinformatics* 2012;**28**:1353–8. https://doi.org/10.1093/bioinformatics/bts163

80. Prentice RL. Exponential survivals with censoring and explanatory variables. *Biometrika* 1973;**60**:279–88. https://doi.org/10.1093/biomet/60.2.279

81. Verweij PJM, van Houwelingen HC. Time-dependent effects of fixed covariates in cox regression. *Biometrics* 1995;**51**:1550–6. https://doi.org/10.2307/2533286

82. Scheike TH, Martinussen T. On estimation and tests of time-varying effects in the proportional hazards model. *Scand J Stat* 2004;**31**:51–62. https://doi.org/10.1111/j.1467-9469.2004.00372.x

83. Sauerbrei W, Royston P, Look M. A new proposal for multivariable modelling of time-varying effects in survival data based on fractional polynomial time-transformation. *Biom J* 2007;**49**:453–73. https://doi.org/10.1002/bimj.200610328

84. Xie G, Dong C, Kong Y. *et al.* Group lasso regularized deep learning for cancer prognosis from multi-omics and clinical features. *Genes* 2019;**10**:240.

85. Hao J, Masum M, Oh JH. *et al.* Gene-and pathway-based deep neural network for multi-omics data integration to predict cancer survival outcomes. In: Cai Z (ed.), *Bioinformatics Research and Applications: 15th International Symposium, ISBRA 2019, Barcelona, Spain, June 3–6, 2019, Proceedings* 15. Switzerland: Springer, 2019, 113–24.

86. Yousefi S, Amrollahi F, Amgad M. *et al.* Predicting clinical outcomes from large scale cancer genomic profiles with deep survival models. *Sci Rep* 2017;**7**:11707. https://doi.org/10.1038/s41598-017-11817-6

87. Zhang JZ, Wei X, Pingzhao H. Tightly integrated multiomics-based deep tensor survival model for time-to-event prediction. *Bioinformatics* 2022;**38**:3259–66. https://doi.org/10.1093/bioinformatics/btac286

88. da Costa Avelar PH, Laddach R, Karagiannis SN. *et al.* Multiomic data integration and feature selection for survival-based patient stratification via supervised concrete autoencoders. In: Nicosia G (ed.), *International Conference on Machine Learning, Optimization, and Data Science*. Switzerland: Springer, 2022, 47–61.

89. Zhang X, Xing Y, Sun K. *et al.* Omiembed: A unified multitask deep learning framework for multi-omics data. *Cancer* 2021;**13**:3047.

90. Benkirane H, Pradat Y, Michiels S. *et al.* Customics: A versatile deep-learning based strategy for multi-omics integration. *PLoS Comput Biol* 2023;**19**:e1010921.

91. Fan Z, Jiang Z, Liang H. *et al.* Pancancer survival prediction using a deep learning architecture with multimodal representation and integration. *Bioinformatics Advances* 2023;**3**:vbad006. https://doi.org/10.1093/bioadv/vbad006

92. Wen G, Li L. FGCNSurv: Dually fused graph convolutional network for multi-omics survival prediction. *Bioinformatics* 2023;**39**:btad472.

93. Zhang J, Huang K. Normalized imqcm: An algorithm for detecting weak quasi-cliques in weighted graph with applications in gene co-expression module discovery in cancers. *Cancer Informatics* 2014;**13**:CIN–S14021.

94. Golub GH, Reinsch C. Singular value decomposition and least squares solutions. In: Bauer FL (ed.), *Handbook for Automatic Computation: Volume II: Linear Algebra*. Berlin: Springer, 1971, 134–51.

95. Kingma DP, Ba J. Adam: A method for stochastic optimization. 3rd International Conference on Learning Representations (ICLR). 2015. https://arxiv.org/abs/1412.6980

96. Bottou L. Large-scale machine learning with stochastic gradient descent. In: Lechevallier Y (ed.), *Proceedings of COMPSTAT'2010: 19th International Conference on Computational Statistics*. Berlin: Springer, 2010, 177–86.

97. Lawrence MS, Stojanov P, Mermel CH. *et al.* Discovery and saturation analysis of cancer genes across 21 tumour types. *Nature* 2014;**505**:495–501.

98. Mermel CH, Schumacher SE, Hill B. *et al.* Gistic2. 0 facilitates sensitive and confident localization of the targets of focal somatic copy-number alteration in human cancers. *Genome Biol* 2011;**12**:1–14. https://doi.org/10.1186/gb-2011-12-4-r41

99. Andrew Futreal P, Coin L, Marshall M. *et al.* A census of human cancer genes. *Nat Rev Cancer* 2004;**4**:177–83. https://doi.org/10.1038/nrc1299

100. Snoek J, Larochelle H, Adams RP. Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems* 2012;**25**:1–9.

101. Bader BW, Kolda TG. Efficient matlab computations with sparse and factored tensors. *SIAM Journal on Scientific Computing* 2008;**30**:205–31.

102. Kvamme H, Borgan Ø. Continuous and discrete-time survival prediction with neural networks. *Lifetime Data Anal* 2021;**27**:710–36. https://doi.org/10.1007/s10985-021-09532-6

103. Maddison CJ, Mnih A, Teh YW. The concrete distribution: A continuous relaxation of discrete random variables.

5th International Conference on Learning Representations (ICLR) 2017. https://arxiv.org/abs/1611.00712

104. Li Y, Wang N, Liu J. *et al*. Factorized bilinear models for image recognition. In: Cucchiara R (ed.), *Proceedings of the IEEE International Conference on Computer Vision*. Piscataway, New Jersey: IEEE, 2017, 2079–87.

105. Srivastava RK, Greff K, Schmidhuber J. Training Very Deep Networks. In: Cortes C (ed.), *Advances in Neural Information Processing Systems 28*. Red Hook, New York: Curran Associates, 2016, 2377–85.

106. Thomas N. Kipf and max welling. Semi-supervised classification with graph convolutional networks. In: Bengio Y (ed.), *5th International Conference on Learning Representations (ICLR 2017)*. Red Hook, New York: Curran Associates, 2016, 2713–26.

107. Klau S, Jurinovic V, Hornung R. *et al*. Priority-lasso: A simple hierarchical approach to the prediction of clinical outcome using multi-omics data. *BMC Bioinformatics* 2018;**19**: 1–14.

108. Hornung R, Wright MN. Block forests: Random forests for blocks of clinical and omics covariate data. *BMC Bioinformatics* 2019;**20**:1–17.

109. Cho HJ, Shu M, Bekiranov S. *et al*. Interpretable meta-learning of multi-omics data for survival analysis and pathway enrichment. *Bioinformatics* 2023;**39**:btad113. https://doi.org/10.1093/bioinformatics/btad113

110. Finn C, Abbeel P, Levine S. Model-agnostic meta-learning for fast adaptation of deep networks. In: Precup D (ed.), *International Conference on Machine Learning*, pp. 1126–35. Proceedings of Machine Learning Research, 2017.

111. Zhou T, Kuscsik Z, Liu J-G. *et al*. Solving the apparent diversity-accuracy dilemma of recommender systems. *Proc Natl Acad Sci* 2010;**107**:4511–5.

112. Haider H, Hoehn B, Davis S. *et al*. Effective ways to build and evaluate individual survival distributions. *Journal of Machine Learning Research* 2020;**21**:1–63.

113. Zhang B, Wang J, Wang X. *et al*. Proteogenomic characterization of human colon and rectal cancer. *Nature* 2014;**513**:382–7. https://doi.org/10.1038/nature13438

114. Zhang H, Liu T, Zhang Z. *et al*. Rodland, and CPTAC investigators. Integrated proteogenomic characterization of human high-grade serous ovarian cancer. *Cell* 2016;**166**:755–65. https://doi.org/10.1016/j.cell.2016.05.069

115. Pronina IV, Loginov VI, Burdennyy AM. *et al*. Dna methylation contributes to deregulation of 12 cancer-associated micrornas and breast cancer progression. *Gene* 2017;**604**:1–8. https://doi.org/10.1016/j.gene.2016.12.018

116. Gyoőrffy B, Bottai G, Fleischer T. *et al*. Aberrant dna methylation impacts gene expression and prognosis in breast cancer subtypes. *Int J Cancer* 2016;**138**:87–97. https://doi.org/10.1002/ijc.29684

117. O'Flanagan CH, Rossi EL, McDonell SB. *et al*. Metabolic reprogramming underlies metastatic potential in an obesity-responsive murine model of metastatic triple negative breast cancer. *NPJ Breast Cancer* 2017;**3**:1–11. https://doi.org/10.1038/s41523-017-0027-5

118. Eichner LJ, Perry M-C, Dufour CR. *et al*. Mir-378* mediates metabolic shift in breast cancer cells via the pgc-1β/errγ transcriptional pathway. *Cell Metab* 2010;**12**:352–61.

119. Nam AS, Chaligne R, Landau DA. Integrating genetic and non-genetic determinants of cancer evolution by single-cell multi-omics. *Nat Rev Genet* 2021;**22**:3–18. https://doi.org/10.1038/s41576-020-0265-5

120. Subramanian I, Verma S, Kumar S. *et al*. Multi-omics data integration, interpretation, and its application. *Bioinformatics and Biology Insights* 2020;**14**:117793221989905. https://doi.org/10.1177/1177932219899051

121. Ramasamy A, Mondry A, Holmes CC. *et al*. Key issues in conducting a meta-analysis of gene expression microarray datasets. *PLoS Med* 2008;**5**:e184. https://doi.org/10.1371/journal.pmed.0050184

122. Tseng GC, Ghosh D, Feingold E. Comprehensive literature review and statistical considerations for microarray meta-analysis. *Nucleic Acids Res* 2012;**40**:3785–99. https://doi.org/10.1093/nar/gkr1265

123. Canzler S, Schor J, Busch W. *et al*. Prospects and challenges of multi-omics data integration in toxicology. *Arch Toxicol* 2020;**94**: 371–88. https://doi.org/10.1007/s00204-020-02656-y

124. Rappoport N, Shamir R. Multi-omic and multi-view clustering algorithms: Review and cancer benchmark. *Nucleic Acids Res* 2018;**46**:10546–62.

125. Ma S, Ren J, Fenyö D. Breast cancer prognostics using multi-omics data. *AMIA Summits on Translational Science Proceedings* 2016;**2016**:52–9.

126. Maghsoudi Z, Nguyen H, Tavakkoli A. *et al*. A comprehensive survey of the approaches for pathway analysis using multi-omics data integration. *Brief Bioinform* 2022;**23**:bbac435. https://doi.org/10.1093/bib/bbac435

127. Kang M, Ko E, Mersha TB. A roadmap for multi-omics data integration using deep learning. *Brief Bioinform* 2022;**23**:bbab454. https://doi.org/10.1093/bib/bbab454

128. Zhou X, Chai H, Zhao H. *et al*. Imputing missing rna-sequencing data from dna methylation by using a transfer learning–based neural network. *GigaScience* 2020;**9**:giaa076.

129. Lin X, Boutros PC. Optimization and expansion of non-negative matrix factorization. *BMC Bioinformatics* 2020;**21**:7. https://doi.org/10.1186/s12859-019-3312-5

130. Sharifi-Noghabi H, Zolotareva O, Collins CC. *et al*. Moli: Multi-omics late integration with deep neural networks for drug response prediction. *Bioinformatics* 2019;**35**:i501–9. https://doi.org/10.1093/bioinformatics/btz318

131. El-Manzalawy Y, Hsieh T-Y, Shivakumar M. *et al*. Min-redundancy and max-relevance multi-view feature selection for predicting ovarian cancer survival using multi-omics data. *BMC Med Genomics* 2018;**11**:19–31. https://doi.org/10.1186/s12920-018-0388-0

132. Spicker JS, Brunak S, Frederiksen KS. *et al*. Integration of clinical chemistry, expression, and metabolite data leads to better toxicological class separation. *Toxicol Sci* 2008;**102**:444–54. https://doi.org/10.1093/toxsci/kfn001

133. Li Y, Fang-Xiang W, Ngom A. A review on machine learning principles for multi-view biological data integration. *Brief Bioinform* 2018;**19**:325–40.

134. Irvine RA, Lin IG, Hsieh C-L. Dna methylation has a local effect on transcription and histone acetylation. *Mol Cell Biol* 2002;**22**: 6689–96.

135. Yip HYK, Papa A. Signaling pathways in cancer: Therapeutic targets, combinatorial treatments, and new developments. *Cells* 2021;**10**:659. https://doi.org/10.3390/cells10030659

136. Cen W, Zhou F, Ren J. *et al*. A selective review of multi-level omics data integration using variable selection. *High-Throughput* 2019;**8**:4. https://doi.org/10.3390/ht8010004

137. Wörheide MA, Krumsiek J, Kastenmüller G. *et al*. Multi-omics integration in biomedical research–a metabolomics-centric review. *Anal Chim Acta* 2021;**1141**:144–62. https://doi.org/10.1016/j.aca.2020.10.038

138. Austin PC, Harrell Jr FE, van Klaveren D. Graphical calibration curves and the integrated calibration index (ICI) for survival models. *Stat Med* 2020;**39**:2714–42. https://doi.org/10.1002/sim.8570

139. Ioannidis JPA, Greenland S, Hlatky MA. *et al.* Increasing value and reducing waste in research design, conduct, and analysis. *The Lancet* 2014;**383**:166–75.

140. Vollmer S, Mateen BA, Bohner G. *et al.* Machine learning and artificial intelligence research for patient benefit: 20 critical questions on transparency, replicability, ethics, and effectiveness. *Br Med J* 2020;**368**:l6927.

141. Büchner T, Berdel WE, Schoch C. *et al.* Double induction containing either two courses or one course of high-dose cytarabine plus mitoxantrone and Postremission therapy by either autologous stem-cell transplantation or by prolonged maintenance for acute myeloid leukemia. *J Clin Oncol* 2006;**24**: 2480–9. https://doi.org/10.1200/JCO.2005.04.5013

142. Döhner H, Estey E, Grimwade D. *et al.* Diagnosis and management of AML in adults: 2017 ELN recommendations from an international expert panel. *Blood* 2017;**129**: 424–47.

143. Braess J, Kreuzer K-A, Spiekermann K. *et al.* High efficacy and significantly shortened neutropenia of dose-dense S-HAM As compared to standard double induction: First results of a prospective randomized trial (AML-CG 2008). *Blood* 2013;**122**:619.

144. Van Houwelingen HC. Validation, calibration, revision and combination of prognostic survival models. *Stat Med* 2000;**19**:3401–15. https://doi.org/10.1002/1097-0258(20001230)19:24\(⟨\)3401::AID-SIM554\(⟩\)3.0.CO;2-2