KI - Künstliche Intelligenz
https://doi.org/10.1007/s13218-023-00822-9

TECHNICAL CONTRIBUTION

Reconstructing Disease Histories in Huge Discrete State Spaces

Rudolf Schill[1,2] · Maren Klever[3] · Kevin Rupp[1,2] · Y. Linda Hu[1] · Andreas Lösch[1] · Peter Georg[4] · Simon Pfahler[4] · Stefan Vocht[1] · Stefan Hansch[1] · Tilo Wettig[4] · Lars Grasedyck[3] · Rainer Spang[1]

Received: 7 April 2023 / Accepted: 28 November 2023

Abstract

Many progressive diseases develop unnoticed and insidiously at the beginning. This leads to an observational gap, since the first data on the disease can only be obtained after diagnosis. Mutual Hazard Networks address this gap by reconstructing latent disease dynamics. They model the disease as a Markov chain on the space of all possible combinations of progression events. This space can be huge: Given a set of $n \geq 266$ events, its size exceeds the number of atoms in the universe. Mutual Hazard Networks combine time-to-event modeling with generalized probabilistic graphical models, regularization, and modern numerical tensor formats to enable efficient calculations in large state spaces using compressed data formats. Here we review Mutual Hazard Networks and put them in the context of machine learning theory. We describe how the Mutual Hazard assumption leads to a compact parameterization of the models and show how modern tensor formats allow for efficient computations in large state spaces. Finally, we show how Mutual Hazard Networks reconstruct the most likely history of glioblastomas.

Keywords Cancer genetics · Cancer progression model · Continuous-time Markov chains · Glioblastoma · Huge combinatorial state spaces · Low-rank tensor formats · Probabilistic graphical models · Proportional hazards · Reconstruction of latent processes

1 Disease Histories

Progressive diseases have long and complex histories. For example, cancer progresses over time as mutations accumulate in the genomes of cancer cells, immune cells invade the tumor, and cells leave the primary lesion and spread to other organs where they form metastases. In addition, clinical complications, the development of drug resistance, and in some cases death of the patient are events in the course of a disease. Every patient has their own disease history, including different progression events that may occur in different temporal orders. The onset of these stochastic processes is never observed. When a patient experiences symptoms and is diagnosed, many of the events have already occurred. To better understand the genesis of these diseases, we want to reconstruct the dynamics of progression-event accumulation. Additionally, to guide treatment decisions, we want to extrapolate the process to predict what will happen next in a patient's disease.

By our definition, event data are binary and consist of vectors that store all events which have occurred during the course of a patient's disease up to the time of observation, see Figs. 1a and 1c. In some cases, such data are available at several points in time, and we can therefore observe how the disease developed. However, more often, we have only one snapshot of the process. In cancer tissue is typically extracted only once, and we have to rely on this single observation of the disease to understand its entire course, past and future. Even more challenging, we do not know the time of onset of a tumor, and thus we do not know either to which point in time the observation corresponds.

Data show that progression events are typically not independent from one another [36]. In cancer, we observe certain

✉ Rainer Spang
Rainer.Spang@ur.de

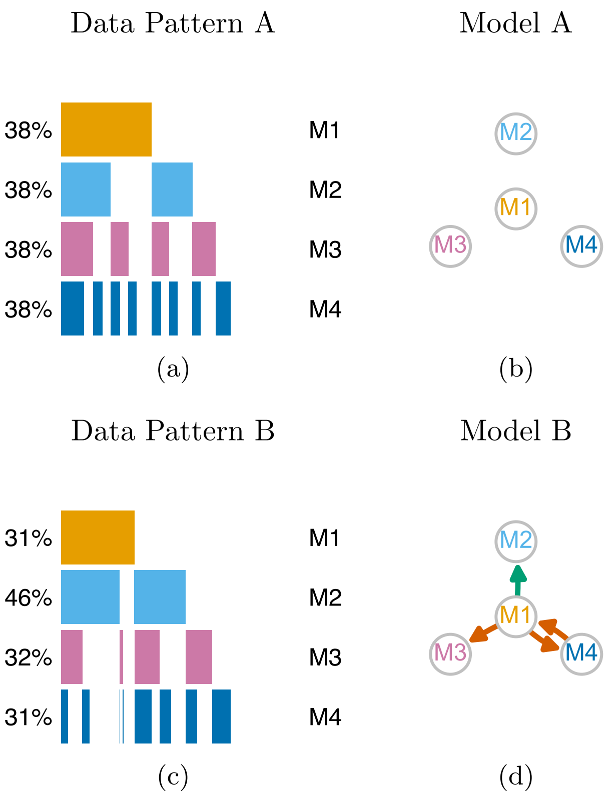
[1] Department of Informatics and Data Science, University of Regensburg, Regensburg 93053, Germany

[2] Department of Biosystems Science and Engineering, ETH Zürich, Basel 4058, Switzerland

[3] Institute for Geometry and Applied Mathematics, RWTH Aachen University, Aachen 52062, Germany

[4] Department of Physics, University of Regensburg, Regensburg 93040, Germany

Published online: 15 January 2024



	Model A	Model B
M1 → M2	1.7%	9.5%
M2 → M1	1.7%	3.0%
{M1, M2}	**3.4%**	**12.5%**

	Model A	Model B
M3 → M1→ M2	0.6%	2.4%
. . .	. . .	. . .
M2 → M1→ M3	0.6%	0.4%
{M1, M2, M3}	**3.9%**	**6.1%**

(e)

Fig. 1 Two Mutual Hazard Networks, Model A and B, describing four events M1-M4 (Figs. 1b and 1d, green arrows indicate promoting, orange arrows inhibiting dependencies) and their corresponding data patterns (Figs. 1a and 1c, columns are samples and rows indicate absence or presence of an event). In Model A, the events accumulate independently from one another, while Model B assumes certain dependencies among events. Figure 1e shows the probabilities with which orders or sets of events will be observed according to the two models

mutations predominantly in tumors that have also acquired a specific other mutation. Vice versa, certain pairs of mutations are hardly ever observed together, i.e., they display patterns of mutual exclusivity [32]. These dependencies can be modeled by assuming that the occurrence of one event changes the rate of another event. For example, in the case of mutually exclusive events, the event that occurs first makes the other less likely to occur. These dependencies are the key to reconstructing the course of the disease.

Figure 1 is an example of how such dependencies can be deduced from snapshot data. It shows two simulated data sets of four mutations recorded in a tumor cohort. Figure 1a was generated assuming independent mutations, while Fig. 1c was generated with certain dependencies between the mutations. Furthermore, the mutations have the same probability to manifest spontaneously, which we call their "base rate." We now explain two dependencies in Fig. 1.

In Fig. 1c, M4 occurs predominantly if M1 does not occur and vice versa. Model B in Fig. 1d explains these dependencies by assuming that the occurrence of one of the two makes the other occur with a lower rate. This is indicated by the two orange arrows between M1 and M4. Model A cannot explain this pattern.

Next, we look at M1 and M2, whose relationship is not symmetric. In contrast to Data Pattern A, almost all samples in Data Pattern B that show M1 also have M2. On the other hand, only about half of the samples with M2 also show M1. Model B explains this by assuming that M1 increases the rate of M2, which is indicated by a green arrow, while M2 has no influence on the rate of M1.

In a model with such dependencies, different temporal orderings of events do not have equal likelihoods. Let us assume that a tumor has mutations M1 and M2. Since M1 makes future acquisition of M2 more likely, but not the other way around, the temporal order M1→M2 is more likely than M2→M1. Similarly, in a tumor with M1 and M3, the order M3→M1 is more likely than M1→M3, as M1 inhibits M3.

2 Related Work

The literature on disease progression models is long and has been excellently reviewed elsewhere [3, 13, 15, 24]. Here, we focus on recent contributions that paved the way for Mutual Hazard Networks. Beerenwinkel et al's Conjunctive Bayesian Networks [2] are Bayesian networks whose node variables are binary and represent the presence or absence of disease events. Events can only occur if all their parent events have already occurred. Mutual dependencies are not allowed. For mutually exclusive events, workarounds have been developed [11, 19]. Mutual dependencies were introduced by Hjelm et al's Network Aberration Models [26]. They model disease progression using Markov chains whose state spaces consist of all possible subsets of the events considered. Each event has an aberration intensity that can be increased - but not decreased - by other events. Also the probability that the sample is "discovered" is modeled, depending on the number of events accumulated. Johnston and Williams introduced HyperTraPS [28], a Markov chain Monte Carlo sampling algorithm that allows one to distinguish between different trajectories of event accumulation. In fact, this statistical platform can be seen as a sampling-based approach to learn the parameters of a Mutual Hazard Networks under certain additional assumptions on the nature of observation times. Finally, Gotovos et al describe a similar sampling-based algorithm that scales up parameter estimation in Mutual Hazard Networks in a way alternative to the low-rank tensor formats described here [20]. Finally, we mention the R package and web application EvAM-Tools, which allows one to train multiple state-of-the-art cancer progression models using a unified interface [14].

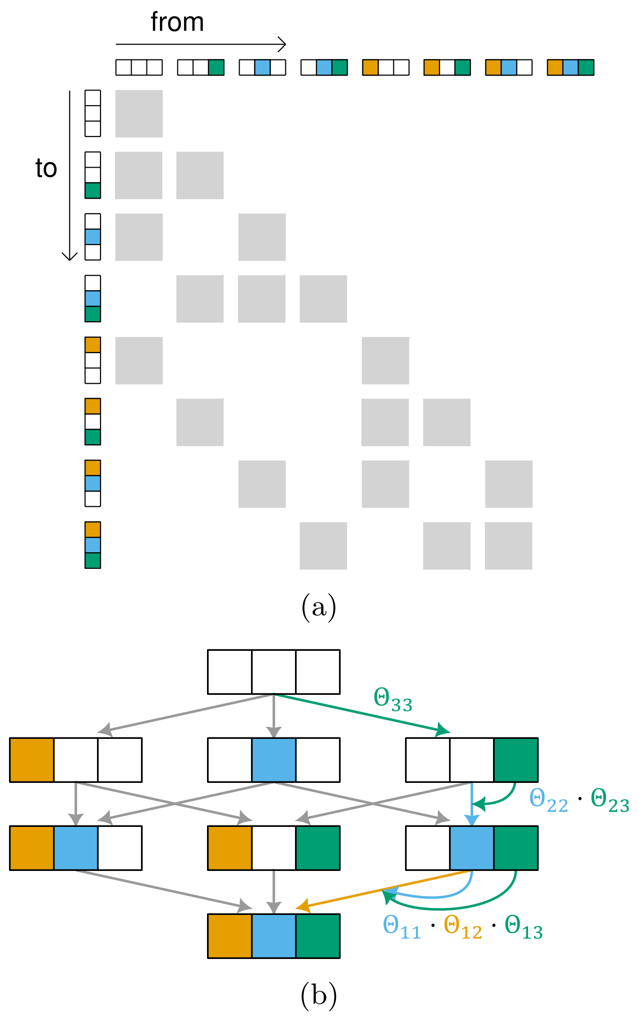


Fig. 2 Fig. 2a shows the sparse lower-triangular rate matrix Q_Θ of a Mutual Hazard Network with three events. Note that the states are ordered in lexicographical order. Figure 2b visualizes the corresponding Markov chain's states and some of the transition rates. Straight arrows indicate transitions that introduce a new event. They occur with the base rate of this event multiplied by the influence of other events that are already present, as indicated by curved arrows

3 Mutual Hazard Networks

Mutual Hazard Networks model disease progression with continuous-time Markov chains. They drastically reduce the number of free parameters using the Mutual Hazard assumption. Given cross-sectional data, optimal parameters can be found using maximum-likelihood estimation. This can be done for data with both known and unknown observation times [38, 39].

In the following, we describe Mutual Hazard Networks and their parameter inference.

3.1 Disease Progression Modeled by a Markov Chain

For a set of n binary events, we define a Markov chain X_t on the state space $S = \{0, 1\}^n$, representing all 2^n possible combinations of these events. The vectors in S represent observable states of the disease at some time point t and contain 1's for events that have occurred until time t and 0's for those events that have not. We assume that at time $t = 0$ no event has occurred yet and that events accumulate one at a time and irreversibly. In other words, if the ith entry of a state vector x switches from 0 to 1 (we denote the resulting vector by x_{+i}), all state vectors at later times hold a 1 in this entry.

The rate matrix Q of the Markov chain can be very large, as the state space S grows exponentially with the number of events. Note that by ordering S lexicographically, the rate matrix becomes lower triangular due to the irreversibility of events, as depicted in Fig. 2a.

3.2 Reducing the Number of Free Parameters in the Rate Matrix with the Proportional Hazard Assumption

The estimation of Q would be intractable for a large number of events n. [39] alleviate this problem by assuming additional structure of Q, introducing what we call the Mutual Hazard assumption. Their Mutual Hazard Networks are probabilistic graphical models that describe the Markov chain's transitions as Cox Proportional Hazard Models [9]. In concrete terms, the rate of transition from a state x to the state x_{+i} is parameterized as

$$q_{x \to x_{+i}} = \Theta_{ii} \prod_{x_j \neq 0} \Theta_{ij} \, .$$

This represents the accumulation rate of an event i as the product of the event's positive base rate Θ_{ii} and positive influence factors Θ_{ij} for each event j that has already occurred in x. The diagonal entries of the resulting matrix Θ represent the spontaneous rates at which an event occurs if no other events have yet occurred. Off-diagonal entries Θ_{ij} encode inhibiting (< 1) or promoting (> 1) modulations on the rates of events i from events j that occurred previously. The now much smaller parameter matrix Θ can be inferred using (regularized) maximum-likelihood estimation.

3.3 Parameter Inference

Let D be a data set of binary vectors in S and p_D the corresponding empirical distribution on S. For a given Θ, the distribution of the Markov chain at time t is a vector of length 2^n given by

$$p_\Theta(t) = \exp(tQ_\Theta)p(0)\,, \tag{1}$$

where $p(0) = (1, 0, \dots, 0) \in [0, 1]^S$ denotes the distribution at $t = 0$, which is completely concentrated in the initial state $(0, \dots, 0) \in S$. This holds by construction, as every disease is assumed to start event-free.

To derive the likelihood of Θ and the likelihood's gradient given D, we distinguish between two scenarios: In the first scenario the observation times of the data points are unknown. In the case of cancer, for example, $t = 0$ corresponds to the onset of the cancer, which is unobservable in human data. Hence, even if we know the date of an observation, we still do not know how much time has passed between the onset of cancer and the observation, i.e., the Markov-chain time. In the second scenario, the observation time is known. For example, this is the case when a cancer is experimentally induced in a mouse by a researcher.

Unknown observation time. [39] assume that the unknown observation times are independent, exponentially distributed random variables with rate 1. Under this assumption, marginalizing over t in equation (1) yields

$$\begin{aligned} p_\Theta &= \int_0^\infty \exp(-t)\exp(tQ_\Theta)p(0)\,\mathrm{d}t \\ &= \underbrace{(I - Q_\Theta)}_{=:R_\Theta}{}^{-1}p(0)\,, \end{aligned}$$

and thus the log-likelihood of Θ given D is

$$S_D(\Theta) = \sum_{x\in D} \log\left(R_\Theta^{-1}p(0)\right)_x .$$

Its gradient is given by

$$\frac{\partial S_D}{\partial \Theta_{ij}} = \sum_{x\in D} \frac{1}{(p_\Theta)_x}\left(R_\Theta^{-1}\frac{\partial Q_\Theta}{\partial \Theta_{ij}}p_\Theta\right)_x .$$

Note that the computation of the log-likelihood and its derivative involve the application of the inverse of the $2^n \times 2^n$ matrix R_Θ to a vector, which is equivalent to solving a linear system of equations $R_\Theta p = q$. This can be done efficiently by taking advantage of the matrix' triangularity using either forward substitution or the Neumann series. The latter boils down to a finite sum of matrix–vector products, due to nilpotency of Q_Θ.

Known observation time. For every data point $x \in D$, let t_x be the time of observation. Following equation (1), the log-likelihood of Θ given D is

$$S_D(\Theta) = \sum_{x\in D} \log\left(\exp(t_x Q_\Theta)p(0)\right)_x .$$

The computation of both this log-likelihood and its gradient involve the matrix exponential of the $2^n \times 2^n$ matrix Q_Θ. Grassmann and Rupp et al give numerically stable algorithms approximating these with a series of matrix–vector products [22, 38].

Finally, in both cases log-likelihood maximization can be carried out using, for example, the L-BFGS(-B) algorithm, a quasi-Newton algorithm designed for optimization problems with many variables and limited memory usage [6].

Likelihood optimization can lead to parameter matrices Θ with many nontrivial off-diagonal entries different from 1. To avoid overfitting and at the same time reduce the complexity of the model, we enforce sparsity of the model using an L1-penalty. Our objective function thus becomes

$$S_D(\Theta) + \lambda \sum_{i\neq j} |\log(\Theta_{ij})|$$

for some $\lambda > 0$, which can be determined from cross-validation using S_D.

This allows us to visualize the Mutual Hazard Network as a graph with the events as nodes and the interactions, i.e., Θ entries, as edges between them, as in Figs. 1b, 1d, and 5.

4 Efficient Computation

Training a Mutual Hazard Network involves operations with the matrices R_Θ and Q_Θ, such as solving linear systems of equations $R_\Theta p = q$ or applying the matrix exponential $\exp\left(Q_\Theta\right)p$. Both matrices are huge. In fact, even for moderate n, they can be too large to store on any computer. For $n \geq 266$, the state space contains more elements than there are atoms in the observable universe. However, there are up to 800 genes known to be involved in cancer progression [1, 31, 41] whose mutations could be included as events in a comprehensive model.

The solution to this problem is the use of data formats that compress matrices and vectors but still allow for arithmetic computations.

As a first step, the proportional hazard assumption allows for a compact and computationally advantageous tensor representation of Q_Θ. It can be written as a short sum of tensor products of n small matrices,

$$Q_\Theta = \sum_{i=1}^{n} \bigotimes_{j=1}^{i-1} \begin{pmatrix} 1 & 0 \\ 0 & \Theta_{ij} \end{pmatrix} \otimes \begin{pmatrix} -\Theta_{ii} & 0 \\ \Theta_{ii} & 0 \end{pmatrix} \otimes \bigotimes_{j=i+1}^{n} \begin{pmatrix} 1 & 0 \\ 0 & \Theta_{ij} \end{pmatrix} . \quad (2)$$

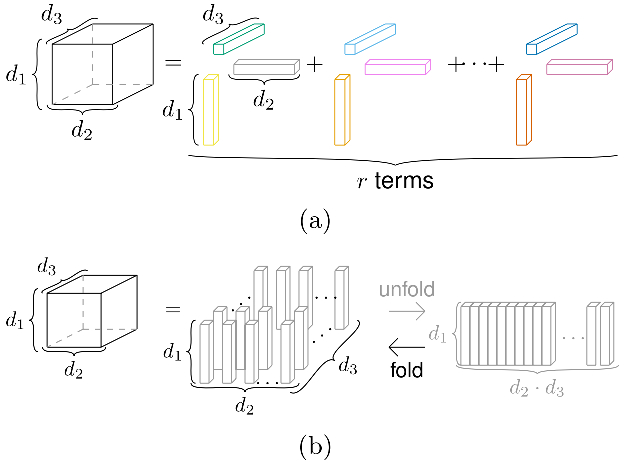
This tensor representation reduces the storage cost of Q_Θ from exponential to quadratic in n. In addition, it speeds up the matrix operations required to train a Mutual Hazard Network. For example, matrix–vector products can be reduced from $\mathcal{O}(2^{2n})$ to $\mathcal{O}(n2^{n-1})$ using the shuffle algorithm [5]. Although still intractable for large n, this can be a significant speed-up for moderate n. In fact, using this tensor representation, Mutual Hazard Networks of size $n = 25$ can be trained [39].

CP-format. To further reduce storage and computation costs, we want to represent also the operands of Q, namely the distribution vectors p over S, as a short sums of tensor products,

$$p = \sum_{i=1}^{r} \bigotimes_{j=1}^{n} p_i^{(j)} , \quad (3)$$

where the $p_i^{(j)}$ are vectors of dimension d_j. This encodes the operand p as a higher-order tensor of order n with dimensions d_1, ..., d_n. In Mutual Hazard Networks, the distribution p_Θ is a tensor of order n with constant dimensions $d_1 = \ldots = d_n = 2$. In the tensor literature, the representation in equation (3) is known as canonical polyadic (CP) format [7, 25], where the number r of terms is called the format's CP-rank (or simply rank). Figure 3a illustrates a CP-representation for an order-3 tensor with dimensions d_1, d_2, d_3 and CP-rank r. A core advantage of the CP-format is its low storage cost in $\mathcal{O}(dnr)$.

Limitations of the CP-format become evident when performing arithmetics within this format. With every operation, the rank can increase and with it storage and computational costs. For example, if we add two CP-tensors p and q with ranks r_p and r_q by appending the terms of q to those of p, the sum $p + q$ already has rank $r_p + r_q$. Similarly, applying a CP-operator Q with rank r_Q to a CP-tensor p with rank r_p results in a CP-tensor of rank $r_Q \cdot r_p$. The critical quantity for these tensor formats is no longer the order n, but the rank r. For this reason, these formats are called low-rank tensor formats.



(a)

(b)

Fig. 3 3a: Illustration for a CP-representation of a $d_1 \times d_2 \times d_3$ tensor p and CP-rank r, 3b: Unfolding of a $d_1 \times d_2 \times d_3$ tensor p with row dimension $\{1\}$ and column dimensions $\{2, 3\}$

Rank truncation. Low-rank tensor formats compress huge matrices and vectors efficiently. To keep the ranks low after performing arithmetic operations, we need an additional rank-truncation strategy, which approximates the tensor resulting from an arithmetic operation by another tensor of lower rank.

For tensors of order 2 (matrices) the singular value decomposition provides a best-rank r approximation by keeping only the singular vectors corresponding to the r largest singular values [16]. For higher-order tensors, the set of CP-tensors is not closed, and thus low-rank approximation within the CP-format is an ill-posed problem [40].

Using other low-rank tensor formats, truncation based on singular value decomposition can be generalized to higher-order tensors. In a nutshell, a higher-order tensor is unfolded into a matrix by selecting dimensions that define its rows while all others define its columns. The resulting matrices are called unfoldings. Figure 2b illustrates the isomorphism of unfolding and (re)folding an arbitrary tensor. Here a tensor p of order 3 with dimensions d_1, d_2, d_3 is unfolded into a matrix by selecting row dimension $\{1\}$ and column dimensions $\{2, 3\}$. Tree tensor formats take advantage of this idea.

Tensor trains. The low-rank tree tensor format we focus on is the tensor-train format [34], also known in physics as matrix product states [35, 46]. A tensor p of order n with dimensions d_1, ..., d_n is factorized into n smaller core-tensors $p^{(i)}$ of size $r_i \times d_i \times r_{i+1}$,

$$p_x = \sum_{j_1=1}^{r_1} \cdots \sum_{j_{n+1}=1}^{r_{n+1}} p^{(1)}_{j_1,x_1,j_2} \cdot p^{(2)}_{j_2,x_2,j_3} \cdots p^{(n)}_{j_n,x_n,j_{n+1}}$$

for all entries $x = (x_1, \ldots, x_n)$ with $r_1 = r_{n+1} = 1$. The tuple $(r_1, \ldots, r_{n+1})$ is called the tensor-train rank (or simply rank) of this factorization.

Every CP-tensor of CP-rank r can be represented in the tensor-train format with tensor-train rank bounded component-wise by r, while the reverse is not true in general. Figure 4a illustrates how a CP-tensor can be transformed into a tensor train.

Tensor trains have high compression rates, provided they have low rank components. Instead of storing a high-order tensor p with cost in $\mathcal{O}(d^n)$ only the cores $p^{(i)}$ are stored with cost in $\mathcal{O}(dnr^2)$, where $d_i \leq d$ and $r_i \leq r$.

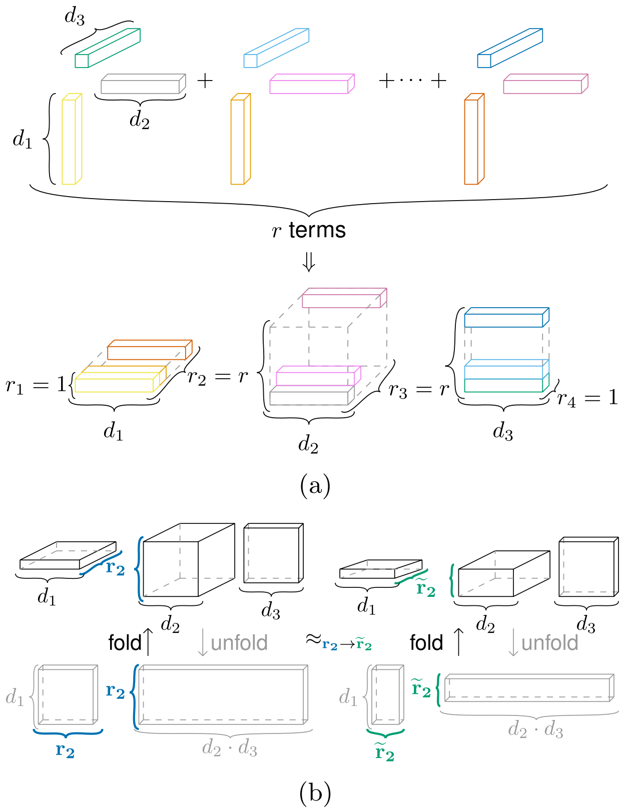


Fig. 4 4a: Transfer of a $d_1 \times d_2 \times d_3$ CP-tensor p with CP-rank r into a tensor-train format with rank (1, r, r, 1), 4b: Truncation of a tensor train p (black) by reducing $r_2 \to \widetilde{r}_2$ with corresponding low-rank factorized unfoldings given by the operation `reshape`$(p, d_1, d_2 \cdot d_3)$ (gray)

Table 1 Operations and their costs for tensors p, q and operators Q of order n with constant dimensions d in the tensor-train format with rank component-wise bounded by r [33]

Operation	Formula	Cost
Storage		$\mathcal{O}(ndr^2)$
Addition	$p+q$	$\mathcal{O}(ndr^3)$
Evaluation	p_x	$\mathcal{O}(ndr^2)$
Inner product	$\langle p, q\rangle$	$\mathcal{O}(ndr^3)$
Matrix–vector product	Q_p	$\mathcal{O}(nd^2r^4)$
Trunction	$\widetilde{p} \approx p$	$\mathcal{O}(ndr^3)$

Each rank component r_i, $i \le d$, corresponds to the matrix rank of an unfolding with row dimensions $\{1, \dots, i\}$. Using a rank-truncated singular value decomposition for the unfoldings in a hierarchical way gives a low-rank approximation in the tensor-train format [34]. Figure 4b illustrates a truncation step for an order $n = 3$ tensor p in the tensor-train format, where r_2 is truncated to $\widetilde{r}_2$. Tensor trains allow for efficient arithmetic operations. Table 1 lists operations together with their cost.

The performance of low-rank tensor methods greatly depends on the choice of unfoldings. In addition to tensor trains, several alternative formats are available, such as the hierarchical Tucker format [21, 23].

In summary, tensor formats combined with rank truncation can compress huge matrices and vectors. Moreover, arithmetic operations such as matrix–vector products to solve linear systems or the application of matrix exponentials can be carried out efficiently in these compressed formats. Even in situations where matrices such as Q_Θ and R_Θ have more entries than there are atoms in the observable universe, we can still perform approximate computations with them in compressed low-rank tensor formats. These formats have already been successfully used for higher-order Mutual Hazard Networks whose distributions could not be stored or computed using classical methods [18].

5 Tensor Formats and Probabilistic Graphical Models

Low-rank tensor formats have not been used frequently in machine learning. In contrast, probabilistic graphical models are well established in the field. For this reason, we want to bridge the gap between low-rank tensor formats and probabilistic graphical models with discrete random variables. Note that the graph of a Mutual Hazard Network cannot be directly equated with a probabilistic graphical model. However, the joint probability distributions for Mutual Hazard Networks can be approximately factorized in a similar way.

First, any joint probability distribution P of n discrete random variables X_1, ..., X_n over state spaces S_{X_1}, ..., S_{X_n} can be identified with a tensor p of order n,

$$p_{x_1,\dots,x_n} = P(X_1 = x_1, \dots, X_n = x_n) \tag{4}$$

for all states $x = (x_1, \dots, x_n)$. Thus p is non-negative, normalized and has dimensions $d_1 = |S_{X_1}|$, ..., $d_n = |S_{X_n}|$, where $|S_Y|$ denotes the cardinality of S_Y. Conversely, any non-negative, normalized tensor p of order n defines a joint probability distribution P over n discrete random variables.

Moreover, there is a connection between undirected discrete probabilistic graphical models and tensor formats [37]. A probabilistic graphical model for an undirected graph G over visible variables X_1, ..., X_n and hidden variables H_1, ..., H_m is a joint distribution P that factorizes into a set of clique potentials $\{\phi_C\}_C$,

$$P(X = x) = \sum_{h_1 \in S_{H_1}} \dots \sum_{h_m \in S_{H_m}} \prod_{C \text{ clique}} \phi_C(x_C, h_C) \tag{5}$$

for all states $x \in S_X$, where $x_C := \{x_i \mid X_i \in C\}$ and $h_C := \{h_j \mid H_j \in C\}$ [29]. Here, a clique C is a subset of variables that are all pairwise connected in G.

This factorization of the joint distribution P is directly related to the concept of conditional independence: Two random variables Y_1 and Y_2 are called conditionally independent given Z if $P(Y_1, Y_2|Z) = P(Y_1|Z) \cdot P(Y_2|Z)$ [29]. Thus, in a graphical model, two variables are conditionally independent given all other variables if and only if they are not directly connected by an edge. In the factorization (5) of P, two variables are conditionally independent given all others if and only if they never appear together in a clique potential.

Similarly to the tensor-train format, general tree-tensor formats can also be factorized into a set of core tensors $\{p^{(C)}\}_C$,

$$p_{x_1,\dots,x_n} = \sum_{k_1=1}^{r_1} \cdots \sum_{k_m=1}^{r_m} \prod_C p^{(C)}_{x_C,k_C} \tag{6}$$

for all $x = (x_1, \dots, x_n)$, where $(r_1, \dots, r_m)$ is the rank of p in the tree-tensor format. Note that in low-rank tensor formats the core tensors typically have a small order, e.g., order 3 for the tensor-train format, and thus the right-hand side of equation (6) typically reduces the storage complexity from exponential to linear in n. Assuming that all cores $p^{(C)}$ are non-negative, we observe the following relationship by comparing the factorizations: The core tensors $p^{(C)}$ can be seen as evaluations of the clique potentials ϕ_C, the rank of p corresponds to the cardinality of the hidden variables, i.e., $r_j = |S_{H_j}|$, and vice versa.

Based on this relationship, the low-rank approximation (assuming non-negative cores) can be understood as an approximation of a joint probability distribution by an undirected graphical model with small hidden variables. In other words, in a low-rank approximation of distributions with non-negative cores, we look for clique potentials with small maximal cliques and hidden variables with small state spaces whose model still describes the distribution as accurately as possible. Thus, in addition to its cost-effectiveness, non-negative low-rank tensor approximation of probability distributions provides an interesting aspect of understanding the model that warrants further investigation.

6 A History Book of Glioblastomas

Glioblastomas are the most common form of malignant primary brain tumor in adults, notorious for their aggressiveness and poor prognosis [27, 42]. Like for all cancers, genomic changes (the events that we will consider here) begin to accumulate long before the onset of symptoms and clinical presentation. At the time when they can be observed, the order and dynamics of their accumulation is thus obscured. [39] used Mutual Hazard Networks to reconstruct the genomic history of glioblastomas to better understand the dynamics of the disease.

The glioblastoma data set consists of 261 samples characterized by 486 genomic events (gene point mutations (M), gene amplifications (A), and gene deletions (D)) [4, 30]. To model on a subset of events which is both informative and sufficiently frequent in the data, the pre-selection strategy by Constantinescu et al was adopted. This resulted in a final set of 20 events (minimum event frequency 5.4%) [8].

On this data set Mutual Hazard Networks achieved a log-likelihood score of −7.97 in 5-fold cross-validation compared to −8.45 for an unconnected network. The latter assumes that all events occur independently of one another. This shows that the Mutual Hazard Network has in fact detected dependencies among events that generalizes to left-out samples.

The network in Fig. 5 models the dynamics of glioblastomas. A positive edge (green) from an event A to another event B indicates that, if A occurs, the rate for B increases. As a consequence, the average waiting time for event B is reduced once A has occurred, and more patients with A also acquire B before the time of observation. This is, for example, the case for IDH1 mutations that increase the rate of TP53 mutations. In fact, 71.4% of patients who show IDH1(M) also show TP53(M). This rate increase of TP53(M) given IDH1(M) is consistent with experimental observations: Watanabe et al [45] analyzed glioblastoma patients with multiple biopsies taken at different time points and found a strong tendency for these events to co-occur. For multiple cases in which they did co-occur, IDH1(M) preceded TP53(M), but never vice versa, suggesting both a temporal order and a dependency of TP53(M) occurrence on IDH1(M).

Analogously, a negative edge (orange) from A to B encodes that A reduces the rate of B. Given A, the expected waiting time for B is prolonged, and thus the probability that B occurs before the time of observation is reduced. The Mutual Hazard Network has identified pairs of events that mutually inhibit each other. For example, TP53(M) and MDM4(A) are connected by two inhibiting edges. Both events are frequent: 29.1% of the tumors have TP53(M) and 15.7% have MDM4(A). If we assume that the events occur independently of each other, we would expect that 4.6% have both mutations. However, only 2.7% of tumors have both, i.e., events occur less frequently in the same tumor than expected by chance. This data pattern is called mutual exclusivity and has been described frequently [17, 32, 36]. Often, mutually exclusive events are events that trigger similar changes in tumor cells, for example, they both block cell death. This can result in mutual exclusivity if cancer-cell fitness increases with the first event but would remain constant or even decrease with the second event, for example because their combined effect is redundant. In fact, TP53(M) and MDM4(A) both suppress programmed cell death. The TP53 mutation directly inactivates a promoter

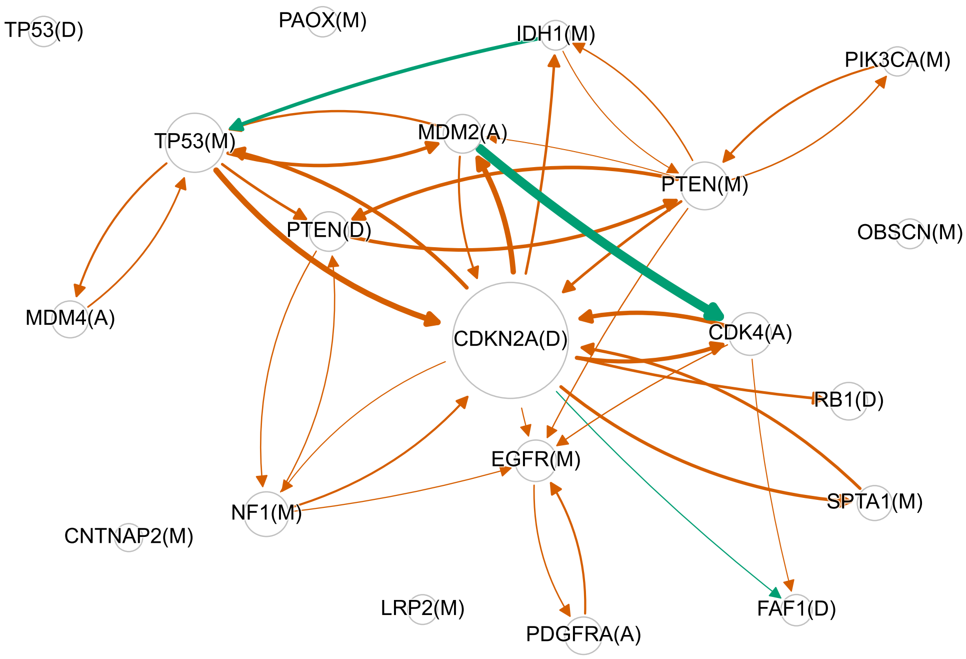


Fig. 5 A Mutual Hazard Network of genetic glioblastoma progression. The nodes are frequent mutations that accumulate in the genomes of glioblastoma cells and that the model was trained on. The size of the nodes scales with the base rate of the individual mutations. The edges represent the dependencies inferred by the model. Their widths scale with the absolute value of the logarithm of the corresponding entry in the parameter matrix Θ. Green edges encode promoting interactions, while orange edges encode inhibiting interactions

of cell death, namely TP53, and the MDM4 amplification over-activates an inhibitor of TP53 [12, 44].

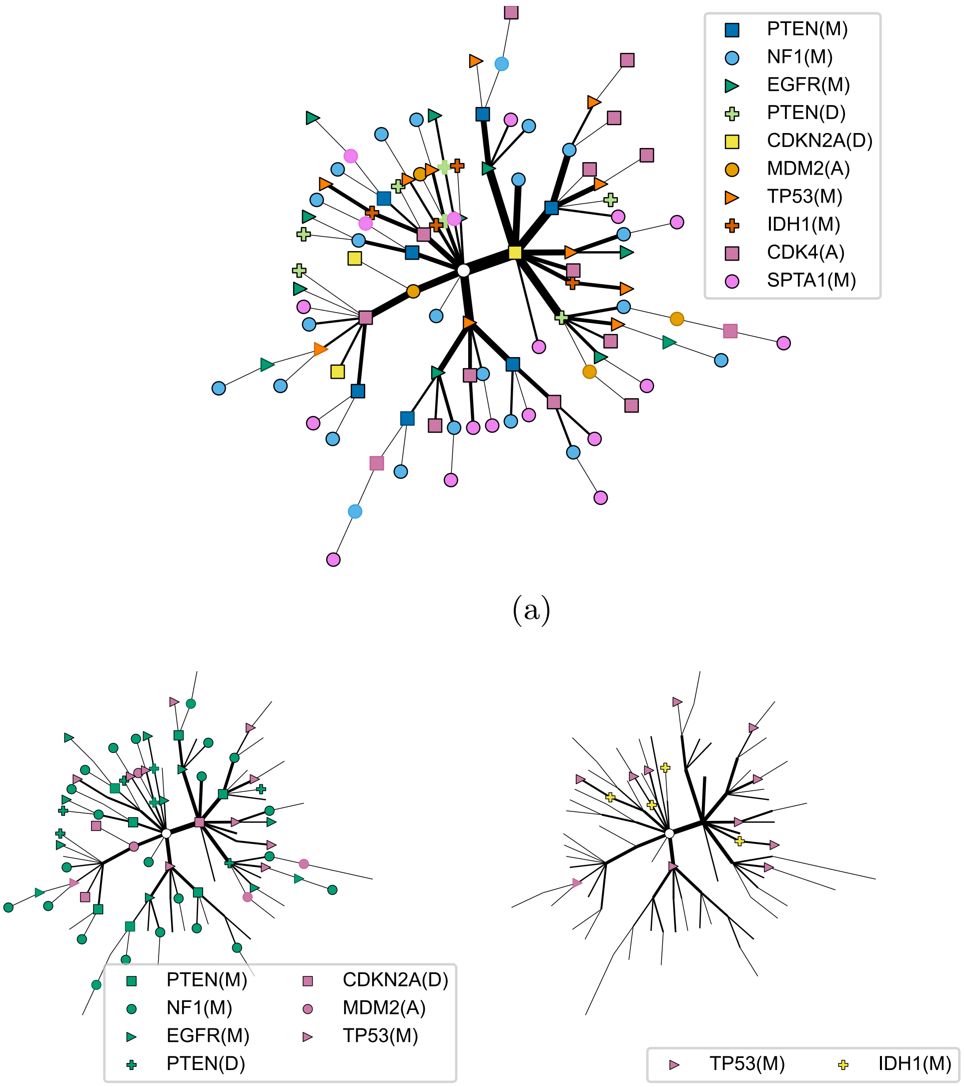
As mentioned above, the onset of cancer is never observed, and we do not know which events occur first. This constitutes one of the biggest scientific gaps in tumor biology. A trained Mutual Hazard Network can reconstruct in which way such a tumor history is most likely to have happened. For every tumor, we see sets of unordered events that occurred before the time of diagnosis without their temporal ordering. However, every temporal ordering of events corresponds to a Markov chain trajectory whose likelihood we can calculate [20], and thus we can reconstruct the history of a tumor by choosing the most likely trajectory.

Figure 6a shows a tree consisting of the reconstructed maximum-likelihood histories of 261 glioblastomas. The root of the tree corresponds to the starting point of all tumors, the state in which no event has occurred. The history of each tumor is encoded as a path from the root outwards to a black-contoured node, and the order of events along this path reconstructs the temporal ordering of its mutations. The width of a line encodes how many tumors share that part of their history. For easier visualization, the events shown in the tree are restricted to the 10 events that show the most interactions with other events.

Mutations in glioblastomas can be broadly subdivided into two functional categories: Some of them are primarily known to enhance cell growth (EGFR(M), NF1(M), PTEN(M) and PTEN(D)), while others prevent cell death (CDKN2A(D), TP53(M), and MDM2(A)) [10]. Enhanced cell growth and inhibited cell death are both crucial to cancer progression.

Interestingly, the model uncovers a rigid temporal order of these two aberrations, which has been highlighted in Fig. 6b. There are three main branches initiated by CDKN2A(D), TP53(M), or MDM2(A), all of which are known to inhibit cell death. Most tumors show both cell-death-inhibiting and cell-growth-enhancing mutations, in which case the former almost always preceded the latter. There is only one rare context in which the order is reversed, namely, in 1.5% of glioblastomas the event PTEN(D) occurred before TP53(M) or MDM2(A) (roughly 11 o'clock on the graph in Fig. 6b). Furthermore, the analysis suggests a preferred order among multiple events involved in enhancing cell growth: PTEN(M) generally precedes NF1(M). Furthermore, returning to the example of IDH1(M) and TP53(M), the reconstructed tumor histories agree with the ordering proposed by Watanabe et al for all of the 10 cases where both events are present [45]. This can be seen in Fig. 6c.

In addition to reconstructing the past, Mutual Hazard Networks can also look into the future, which might help clinicians with treatment decisions. For example, promising results in treating glioblastomas have been shown for an anti-cancer compound called RG7112 in preclinical trials [43]. The therapeutic success of this compound depends on two genomic events, namely the presence of MDM2(A) and the absence of TP53(M) [43]. Let us assume that in the future an oncologist is treating a patient with MDM2(A), among other events. To decide whether or not to administer RG7112, it would be helpful to know whether TP53(M) is expected to occur soon. Moreover, assuming that the patient has MDM2(A) and CDKN2A(D), the model would infer a reduced TP53(M) rate and therefore a longer average waiting



(a)

(b)

(c)

Fig. 6 A reconstruction of the individual histories of 261 glioblastoma cases. For every case, the maximum-likelihood temporal ordering of its events is shown as reconstructed by the trained Mutual Hazard Network. The white central node represents the initial "healthy" state without events. Each trajectory from this state outwards, ending at a black-contoured node, shows the most likely order of events for at least one glioblastoma. Several cases can have a common history, which is indicated by the widths of the edges. The plot is restricted to the ten events with the largest sum of absolute interaction weights. Figure 6a shows all these ten events, while Fig. 6b shows only events primarily associated with either promotion of cell growth (green) or inhibition of cell death (magenta). Figure 6c shows only the two events TP53(M) and IDH1(M)

time, making the administration of RG7112 more attractive. In contrast, if the patient instead carried MDM2(A) and IDH1(M), TP53(M) would be expected to occur soon, and therefore the administration of RG7112 would be discouraged.

7 Summary

Mutual Hazard Networks turn snapshots of binary data into a dynamic model of stochastic progression over time. In cancer research, they can fill major gaps in the understanding of tumors by reconstructing their most likely history. Moreover, forecasting the future course of a tumor, could ideally guide treatment decisions. Initial results on glioblastomas are in line with our partial knowledge of this progression process and at the same time already generated new hypotheses.

Their efficient parameterization and ability to utilize modern tensor formats make them a valuable machine learning tool that could be applied to modeling any other suitable binary progression over time.

It still remains to further investigate properties of Mutual Hazard Networks, like their identifiability or the stability of the history reconstructions. Looking into the future, the model holds great potential for extension, for example by incorporating reversible events or non-binary events, just to name a few of them.

Funding Open Access funding enabled and organized by Projekt DEAL. This work was supported by the German Research Foundation, grants GR-3179/6-1 and TRR305.

Data availability The exact input data for MHN as described in Sect. 6 can be found on https://github.com/RudiSchill/MHN.

References

1. Bailey MH, Tokheim C, Porta-Pardo E et al (2018) Comprehensive Characterization of Cancer Driver Genes and Mutations. Cell 173(2):371-385.e18. https://doi.org/10.1016/j.cell.2018.02.060
2. Beerenwinkel N, Eriksson N, Sturmfels B (2006) Evolution on distributive lattices. J Theoretical Biol 242(2):409–420. https://doi.org/10.1016/j.jtbi.2006.03.013
3. Beerenwinkel N, Schwarz RF, Gerstung M et al (2014) Cancer Evolution: Mathematical Models and Computational Inference. System Biol 64(1):e1–e25. https://doi.org/10.1093/sysbio/syu081
4. Brennan CW, Verhaak RG, McKenna A et al (2013) The Somatic Genomic Landscape of Glioblastoma. Cell 155(2):462–477. https://doi.org/10.1016/j.cell.2013.09.034
5. Buis PE, Dyksen WR (1996) Efficient vector and parallel manipulation of tensor products. ACM Trans Mathe Software 22(1):18–23. https://doi.org/10.1145/225545.225548
6. Byrd RH, Lu P, Nocedal J et al (1995) A Limited Memory Algorithm for Bound Constrained Optimization. SIAM J Sci Comput 16(5):1190–1208. https://doi.org/10.1137/0916069
7. Carroll JD, Chang JJ (1970) Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young'' decomposition. Psychometrika 35(3):283–319. https://doi.org/10.1007/bf02310791
8. Constantinescu S, Szczurek E, Mohammadi P et al (2015) TiMEx: a waiting time model for mutually exclusive cancer alterations. Bioinformatics 32(7):968–975. https://doi.org/10.1093/bioinformatics/btv400
9. Cox DR (1972) Regression Models and Life-Tables. J Royal Stat Soc: Series B (Methodological) 34(2):187–202. https://doi.org/10.1111/j.2517-6161.1972.tb00899.x
10. Crespo I, Vital AL, Gonzalez-Tablas M et al (2015) Molecular and Genomic Alterations in Glioblastoma Multiforme. Am J Pathology 185(7):1820–1833. https://doi.org/10.1016/j.ajpath.2015.02.023
11. Cristea S, Kuipers J, Beerenwinkel N (2017) pathTiMEx: Joint Inference of Mutually Exclusive Cancer Pathways and Their Progression Dynamics. J Comput Biol 24(6):603–615. https://doi.org/10.1089/cmb.2016.0171
12. Danovi D, Meulmeester E, Pasini D et al (2004) Amplification of Mdmx (or Mdm4) Directly Contributes to Tumor Formation by Inhibiting p53 Tumor Suppressor Activity. Molecular Cellular Biol 24(13):5835–5843. https://doi.org/10.1128/MCB.24.13.5835-5843.2004
13. Diaz-Colunga J, Diaz-Uriarte R (2021) Conditional prediction of consecutive tumor evolution using cancer progression models: What genotype comes next? PLOS Comput Bio 17(12):1–23. https://doi.org/10.1371/journal.pcbi.1009055
14. Diaz-Uriarte R, Herrera-Nieto P (2022) EvAM-Tools: tools for evolutionary accumulation and cancer progression models. Bioinformatics 38(24):5457–5459. https://doi.org/10.1093/bioinformatics/btac710
15. Diaz-Uriarte R, Vasallo C (2019) Every which way? On predicting tumor evolution using cancer progression models. PLOS Comput Bio 15(8):1–29. https://doi.org/10.1371/journal.pcbi.1007246
16. Eckart C, Young G (1936) The approximation of one matrix by another of lower rank. Psychometrika 1(3):211–218. https://doi.org/10.1007/bf02288367
17. Gao Q, Cui Y, Shen Y et al (2019) Identifying Mutually Exclusive Gene Sets with Prognostic Value and Novel Potential Driver Genes in Patients with Glioblastoma. BioMed Res Inter 2019:1–7. https://doi.org/10.1155/2019/4860367
18. Georg P, Grasedyck L, Klever M, et al (2022) Low-rank tensor methods for Markov chains with applications to tumor progression models. Journal of Mathematical Biology 86(1). https://doi.org/10.1007/s00285-022-01846-9
19. Gerstung M, Eriksson N, Lin J, et al (2011) The Temporal Order of Genetic and Pathway Alterations in Tumorigenesis. PLoS ONE 6(11):e27,136. https://doi.org/10.1371/journal.pone.0027136, https://doi.org/10.1371/journal.pone.0027136
20. Gotovos A, Burkholz R, Quackenbush J, et al (2021) Scaling up Continuous-Time Markov Chains Helps Resolve Underspecification. https://doi.org/10.48550/arXiv.2107.02911
21. Grasedyck L (2010) Hierarchical Singular Value Decomposition of Tensors. SIAM J Matrix Analysis Appli 31(4):2029–2054. https://doi.org/10.1137/090764189
22. Grassmann W (1977) Transient solutions in markovian queueing systems. Comput Operations Res 4(1):47–53. https://doi.org/10.1016/0305-0548(77)90007-7
23. Hackbusch W, Kühn S (2009) A New Scheme for the Tensor Representation. J Four Analysis Appl 15(5):706–722. https://doi.org/10.1007/s00041-009-9094-9
24. Hainke K, Rahnenführer J, Fried R (2012) Cumulative disease progression models for cross-sectional data: A review and comparison. Biometrical J 54(5):617–640. https://doi.org/10.1002/bimj.201100186
25. Harshman R (1970) Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. UCLA Working Papers in Phonetics 16
26. Hjelm M, Höglund M, Lagergren J (2006) New Probabilistic Network Models and Algorithms for Oncogenesis. J Comput Bio 13(4):853–865. https://doi.org/10.1089/cmb.2006.13.853
27. Janjua TI, Rewatkar P, Ahmed-Cox A et al (2021) Frontiers in the treatment of glioblastoma: Past, present and emerging. Adv Drug Delivery Rev 171:108–138. https://doi.org/10.1016/j.addr.2021.01.012
28. Johnston IG, Williams BP (2016) Evolutionary Inference across Eukaryotes Identifies Specific Pressures Favoring Mitochondrial Gene Retention. Cell Syst 2(2):101–111. https://doi.org/10.1016/j.cels.2016.01.013
29. Koller D, Friedman N (2009) Probabilistic Graphical Models : Principles and Techniques. The MIT Press, Cambridge, Massachusetts, https://mitpress.mit.edu/9780262013192/probabilistic-graphical-models
30. Leiserson MDM, Blokh D, Sharan R et al (2013) Simultaneous Identification of Multiple Driver Pathways in Cancer. PLoS Comput Bio 9(5):e1003. https://doi.org/10.1371/journal.pcbi.1003054
31. Martinez-Jimenez F, Muinos F, Sentis I et al (2020) A compendium of mutational cancer driver genes. Nat Rev Cancer 20(10):555–572. https://doi.org/10.1038/s41568-020-0290-x
32. Mina M, Iyer A, Tavernari D et al (2020) Discovering functional evolutionary dependencies in human cancers. Nature Genet 52(11):1198–1207. https://doi.org/10.1038/s41588-020-0703-5
33. Oseledets IV (2011) Tensor-Train Decomposition. SIAM J Scientific. Comput 33:2295–2317. https://doi.org/10.1137/090752286

34. Oseledets IV, Tyrtyshnikov EE (2009) Breaking the Curse of Dimensionality, Or How to Use SVD in Many Dimensions. SIAM J Sci Comput 31(5):3744–3759. https://doi.org/10.1137/090748330
35. Östlund S, Rommer S (1995) Thermodynamic Limit of Density Matrix Renormalization. Phys Rev Let 75(19):3537–3540. https://doi.org/10.1103/physrevlett.75.3537
36. Park S, Lehner B (2015) Cancer type-dependent genetic interactions between cancer driver alterations indicate plasticity of epistasis across cell types. Molecular Systems Biology 11 (7): 824. https://doi.org/10.15252/msb.20156102
37. Robeva E, Seigal A (2018) Duality of graphical models and tensor networks. Informat Inference: A J IMA 8(2):273–288. https://doi.org/10.1093/imaiai/iay009
38. Rupp K, Schill R, Süskind J et al (2021). Differentiated uniformization: A new method for inferring Markov chains on combinatorial state spaces including stochastic epidemic models. https://doi.org/10.4850/arXiv.2112.10971
39. Schill R, Solbrig S, Wettig T et al (2019) Modelling cancer progression using Mutual Hazard Networks. Bioinformatics 36(1):241–249. https://doi.org/10.1093/bioinformatics/btz513
40. de Silva V, Lim LH (2008) Tensor Rank and the Ill-Posedness of the Best Low-Rank Approximation Problem. SIAM J Matrix Analysis Appl 30(3):1084–1127. https://doi.org/10.1137/06066518x
41. Sondka Z, Bamford S, Cole CG et al (2018) The COSMIC Cancer Gene Census: describing genetic dysfunction across all human cancers. Nature Rev Cancer 18(11):696–705. https://doi.org/10.1038/s41568-018-0060-1
42. Tan AC, Ashley DM, Lopez GY, et al (2020) Management of glioblastoma: State of the art and future directions. CA: A Cancer Journal for Clinicians 70 (4): 299–312. https://doi.org/10.3322/caac.21613
43. Verreault M, Schmitt C, Goldwirt L et al (2016) Preclinical Efficacy of the MDM2 Inhibitor RG7112 in MDM2-Amplified and TP53 Wild-type Glioblastomas. Clin Cancer Res 22(5):1185–1196. https://doi.org/10.1158/1078-0432.ccr-15-1015
44. Wasylishen AR, Lozano G (2016) Attenuating the p53 pathway in human cancers: Many means to the same end. Cold Spring Harbor Perspectives Med 6(8):a026. https://doi.org/10.1101/cshperspect.a026211
45. Watanabe T, Nobusawa S, Kleihues P et al (2009) IDH1 Mutations Are Early Events in the Development of Astrocytomas and Oligodendrogliomas. Am J Pathol 174(4):1149–1153. https://doi.org/10.2353/ajpath.2009.080958
46. White SR (1992) Density matrix formulation for quantum renormalization groups. Phys Rev Lett 69(19):2863–2866. https://doi.org/10.1103/physrevlett.69.2863