

Parallel Adaptive Multi-Objective Evolutionary Learning of Discretized Bayesian Network Classifiers for Clinical Data

Damy M.F. Ha
d.m.f.ha@lumc.nl
Leiden University Medical Center
Leiden, The Netherlands

Thalea Schlender
t.schlender@lumc.nl
Leiden University Medical Center
Leiden, The Netherlands

Yvette M. van der Linden
Y.M.van_der_Linden@lumc.nl
Leiden University Medical Center
Leiden, The Netherlands

Peter A.N. Bosman
Peter.Bosman@cw.nl
Centrum Wiskunde & Informatica
Amsterdam, The Netherlands

Tanja Alderliesten
T.Alderliesten@lumc.nl
Leiden University Medical Center
Leiden, The Netherlands

Abstract

Bayesian Networks (BNs) are of interest from an explainable AI viewpoint, offering transparent probabilistic models for decision support. Baymex is a recently introduced multi-objective evolutionary algorithm for learning discretized BNs, enabling experts to trade-off different objectives of interest, such as likelihood, model complexity, and prior beliefs. While Baymex has been shown to outperform state-of-the-art BN learning approaches, Baymex still 1) requires a lot of computation time and 2) has only been evaluated on synthetic data. To improve scalability, we introduce a parallelization strategy as well as a mechanism that enables adaptively steering optimization toward networks that overfit less. We furthermore reconfigure Baymex to train a BN classifier through multi-objective optimization of cross-entropy loss and the BIC complexity term so as to evaluate its performance on real-world clinical classification tasks. Besides observing speedups up to over 54 times on a 16-core CPU, comparisons against clinically familiar baselines (decision trees, logistic regression, naive Bayes, and random forests) on two open-source (RADCUR and SUPPORT) and one in-house dataset, show that Baymex obtains statistically similar or better predictive performance while producing compact, clinically inspectable BNs. Importantly, Baymex finds multiple plausible BN classifiers that contain predictors consistent with established clinical factors.

CCS Concepts

• **Computing methodologies** → *Parallel computing methodologies*; **Bayesian network models**; *Randomized search*; Machine learning.

Keywords

Bayesian networks, evolutionary algorithms, multi-objective optimization, parallelization, adaptive steering, explainable AI

ACM Reference Format:

Damy M.F. Ha, Thalea Schlender, Yvette M. van der Linden, Peter A.N. Bosman, and Tanja Alderliesten. 2026. Parallel Adaptive Multi-Objective Evolutionary Learning of Discretized Bayesian Network Classifiers for

Clinical Data. In *Genetic and Evolutionary Computation Conference (GECCO Companion '26)*, July 13–17, 2026, San Jose, Costa Rica. ACM, New York, NY, USA, 16 pages. <https://doi.org/10.1145/3795101.3814711>

1 Introduction

Every day, clinicians integrate symptoms, comorbidities, imaging, and lab results to form probabilistic judgments about diagnosis, treatment response, and survival in order to guide their patients. Providing accurate data-driven decision support tools can help improve the consistency, transparency, and overall quality of these decisions. Although modern AI methods have demonstrated substantial promise across a wide range of clinical tasks, their adoption in practice remains limited, partly because most models are not sufficiently transparent [24].

Bayesian Networks (BNs) offer a trusted and tried alternative rooted in classic AI. BNs provide joint distributions over variables using a Directed Acyclic Graph (DAG) and conditional probability tables, allowing explicit modeling of conditional structure and enabling transparent probabilistic reasoning [23, 33]. In contrast to many black-box predictors, both the learned structure (the DAG) and the conditional probability tables can be inspected and challenged by domain experts. This makes BNs particularly interesting from an eXplainable AI (XAI) perspective [2, 3, 24].

Despite these benefits, learning even the structure (i.e., the DAG) of a BN from data remains computationally challenging [10, 21]. As data collection continues to grow in both the number of samples and features, performing (optimal) structure learning becomes progressively more intractable. Moreover, in many clinical applications, the data contains real-valued features, which are usually discretized in order to build a BN (e.g., [29, 31, 47]). However, the discretization procedure, including the number of bins used, shapes the BN structure that is recovered by a structure learning algorithm. Different binning choices modify the conditional probability tables in distinct ways, thereby influencing which dependencies get selected.

The recently introduced **Baymex** (Bayesian network learning via evolutionary intelligence for multi-objective explainable AI) algorithm [19] addresses part of this challenge by learning discretized BNs using a combination of state-of-the-art multi-objective Evolutionary Algorithms (EAs) [7, 8, 26]. This results in a search for solutions that best trade off objectives of interest, such as likelihood, model complexity, and agreement with expert beliefs. Accordingly, instead of providing a single "best" network, Baymex produces an



This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

GECCO Companion '26, San Jose, Costa Rica

© 2026 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-2488-6/2026/07

<https://doi.org/10.1145/3795101.3814711>

approximation of the Pareto set, containing multiple plausible BNs that trade-off the objectives of interest. Besides being able to glean additional insights from inspecting the optimized trade-offs, practitioners benefit from the option to choose the most appropriate model [19]. This offers greater flexibility and chances of uptake for real-world use compared to having to accept a single model returned by the vast majority of BN learning methods [2, 27, 38, 45]. While Baymex has been shown to outperform state-of-the-art BN learning approaches, Baymex still 1) requires a lot of computation time and 2) has only been evaluated on synthetic data. In this work, we address both limitations.

Firstly, EAs are naturally amenable to parallelization because they usually evaluate many candidate solutions per generation independently. Consequently, parallel evolutionary computation is supported by a substantial and well-established body of literature [9, 18, 20]. In this work, we parallelize Baymex and investigate a mechanism that enables adaptively steering optimization to focus as much as possible only on networks that do not overfit.

Secondly, we reconfigure Baymex to train a BN classifier through multi-objective optimization of cross-entropy loss and the BIC complexity term [37]. By doing so, we are able to tackle real-world classification tasks, finding BN classifiers that are optimized for trading off predictive performance and model complexity.

We evaluate the accelerated and fundamentally improved Baymex algorithm¹ on real-world clinical classification tasks. To this end, we include the SUPPORT dataset, comprising a large cohort originally designed for prognostic modeling of 180-day survival in seriously ill hospitalized adults [11, 22]. We also include the RADCUR dataset, comprising an open-access head-and-neck radiotherapy cohort for which we learn a classifier to predict whether patients achieve 2-year survival [44]. Finally, we revisit a study of patients with advanced cancer led by our radiotherapy department on survival in patients with spinal bone metastases and use Baymex to predict, via classification, a minimum 3-month survival [5]. We furthermore make comparisons with widely-used, clinically familiar baselines: logistic regression, decision trees, naive Bayes, and random forest.

Besides observing speedups up to over 54 times on a 16-core CPU, across datasets, we find that Baymex obtains statistically similar or better predictive performance than the explainable baselines and, in some cases, better than all baselines while producing transparently inspectable DAGs and conditional probability tables. Importantly, Baymex finds multiple plausible BN classifiers that contain predictors consistent with established clinical factors.

2 Background

2.1 Bayesian Networks

A Bayesian Network (BN) is a probabilistic graphical model that represents a factorized joint distribution over a set of random variables [33]. A BN is defined by a Directed Acyclic Graph (DAG) whose vertices represent random variables and whose directed edges (i.e., arcs) specify parent–child relationships used to factorize the joint distribution into local conditional probability distributions. Specifically, let $\mathbf{X} = (X_1, \dots, X_N)$ denote N random variables and let $G = (V, E)$ be a DAG whose vertices V correspond to the random

variables and edges E connect the vertices. A BN factorizes the joint probability distribution as follows: $p(\mathbf{X}) = \prod_{i=1}^N p(X_i \mid \text{pa}(X_i))$, where $\text{pa}(X_i)$ denotes all the variables corresponding to the parent vertices of the i -th vertex in the DAG.

2.2 Baymex: Learning Discretized BNs

To understand the proposed improvements that we introduce to Baymex in this paper, we will list core concepts of Baymex here, but we refer the interested reader to the literature for more details [19].

2.2.1 Mixed Discrete/Real-valued Decision Space. A central feature of Baymex is its capability to perform multi-objective optimization in mixed discrete and real-valued search spaces. Baymex models a BN's DAG structure using a discrete encoding scheme. For a network of N vertices, the space of potential edges is defined by $\ell_{\text{total}} = \frac{N(N-1)}{2}$ variables. Each potential edge between vertex i and j ($j > i$ and $i, j \leq N$) is modeled as a ternary decision variable $\in \{0, 1, 2\}$, representing no connection, a directed edge $i \rightarrow j$, or a directed edge $j \rightarrow i$, respectively. To ensure that every solution encodes an acyclic graph, Baymex implements a repair mechanism that identifies and repairs cyclic dependencies.

2.2.2 Discretization. To handle real-valued data, Baymex discretizes real-valued data during search using bins that are encoded into the solution. Specifically, for each real-valued feature j , a bin count B_j , i.e., the number of bins used to discretize the real-valued data, is encoded as a discrete decision variable between a minimum of 2 and a user-defined maximum $B_j^{\max}$. In Baymex, bin boundaries are defined as the midpoints between consecutive samples. To this end, the samples are sorted and duplicate values are deleted. An index in this sorted list then describes between which two consecutive samples the bin boundary is defined. As there are typically many indices, a real-valued decision variable is used to this end, which is rounded to the nearest integer for evaluation. Because depending on the number of bin boundaries to use, the best bin boundaries may be in very different locations, in Baymex $n - 1$ distinct indices are used for n bins for all $n \in \{2, 3, \dots, B_j^{\max}\}$:

$$\left[\underbrace{b_{21}^j}_{2 \text{ bins}}, \underbrace{b_{31}^j, b_{32}^j}_{3 \text{ bins}}, \underbrace{b_{41}^j, b_{42}^j, b_{43}^j}_{4 \text{ bins}}, \dots \right]$$

Here, b_{ki}^j is the i -th boundary used to separate variable j into k bins. The value of B_j determines which of the bin boundaries in the representation are active. The other bin boundaries remain in the solution, but are inactive, i.e., they are introns.

2.2.3 Variation via Gene-pool Optimal Mixing. Baymex combines core components drawn from different variants of the Gene-pool Optimal Mixing Evolutionary Algorithm (GOMEA), adapting them to the requirements of BN learning. For the specific goal of multi-objective optimization, Baymex draws upon a key component of multi-objective GOMEA [26]: clustering. Each generation, the population is clustered into k roughly equal groups in objective space, each addressing part of the approximation front. For each cluster, Baymex configures variation models separately, to tailor variation to the local structure of each region. Baymex makes use of three variation operators: GOM [26], GI-GOM [8], and RV-GOM [7], targeting BN structure, bin counts, and bin boundaries, respectively. Each of

¹The code for Baymex is available at: <https://github.com/damyha/baymex>.

these operators combine variation and selection by sampling new values for a subset of all variables in a solution and accepting the change only if it does not worsen the solution or if the solution can be added to the elitist archive. Otherwise, the changes are reverted. Each such subset is called a linkage set. How the linkage sets and the sampling procedure are configured, differs per type of variable, for which we kindly refer to the literature [7, 8, 19, 26].

2.2.4 Elitist Archive. To retain the non-dominated solutions found during the search, Baymex maintains an elitist archive. While the implementation supports capacity constraints (see [25]), in our experiments, the number of non-dominated solutions remains relatively small compared to the target archive size (10,000 solutions). As a result, the archive functions as a regular elitist archive.

2.2.5 Algorithmic Structure. Algorithm 1 summarizes the key aspects of a Baymex generation and explicitly shows the nested loops over clusters, linkage sets, and solutions. Additionally, it is highlighted where offspring are tested for entry into the elitist archive. For discrete variables, this is immediately after variation of a solution. For real-valued variables, this is after every solution in a cluster has undergone variation.

Algorithm 1: Pseudocode of Baymex’s optimization loop.

```

1 all_offspring_clusters = []
2 for each cluster
3   cluster_offspring = []
4   for each solution in cluster:
5     cluster_offspring.append(clone(solution))
6   for each linkage_set in cluster
7     if linkage_set.type == DISCRETE_GOMEA:
8       for each solution in cluster_offspring:
9         apply_GOM(solution, linkage_set) # Parallelizable
10        add_to_archive(solution)
11     elif linkage_set.type == DISCRETE_GI-GOMEA:
12       for each solution in cluster_offspring:
13         apply_GI-GOM(solution, linkage_set) # Parallelizable
14         add_to_archive(solution)
15     elif linkage_set.type == REAL-VALUED_GOMEA:
16       offspring_for_archive = []
17       for each solution in cluster_offspring:
18         apply_RV-GOM(solution, linkage_set) # Parallelizable
19         offspring_cluster.append(solution)
20       for each solution in offspring_for_archive:
21         add_to_archive(solution)
22   all_offspring_clusters.append(cluster_offspring)
23 replace_population_with_offspring(all_offspring_clusters)

```

2.2.6 Interleaved Multi-start Scheme. In Baymex, the need for manual tuning of the population size and number of clusters is avoided through the use of an Interleaved Multi-start Scheme (IMS). With IMS, several Baymex instances are managed, each with a distinct population size and number of clusters. The IMS uses a base population size of 30, and a base number of clusters of $k = m + 1$, where m is the number of optimization objectives. Every 8 generations of a Baymex instance, one generation of a Baymex instance is performed with twice the population size and one extra cluster. For more details of Baymex we kindly refer to [19].

2.3 Bayesian Network Classifier

BNs can be turned into probabilistic classifiers by designating one random variable as the class label Y and modeling the joint distribution $p(Y, X)$, from which predictions follow via the posterior $p(Y | X)$. BN classifiers have been studied for decades [16], spanning a spectrum from simple naive Bayes, which assumes feature independence given the class, to more flexible BN structures that can capture more complex dependencies [4, 46].

3 Methodology

3.1 Baymex Classifier

In this work, we use BN classification as a real-world test-bed for Baymex. Accordingly, we adapt Baymex from learning BNs to learning BN classifiers by minimizing the training cross-entropy loss. Conditional probabilities are estimated using Dirichlet smoothing with a uniform Dirichlet prior to handle samples unseen during training. We retain the core feature of multi-objective optimization in Baymex and use the BIC complexity term as a second minimization objective to trade-off accuracy and model complexity. For BNs, this is $\sum_{i=1}^N (B_i - 1) \prod_{X_j \in pa(X_i)} B_j \cdot \log\left(\frac{n}{2}\right)$, where n is the sample size. We keep the search space unchanged so that improvements obtained with Baymex for classification remain directly translatable to BN learning.

3.2 Parallelization

As can be seen in Algorithm 1, variation in Baymex proceeds by looping over clusters and, within each cluster, over linkage sets, applying variation to every solution. Consequently, parallelization of solution variation and evaluation can be done straightforwardly for the innermost loops (Lines 8, 12, and 17). However, updating the elitist archive remains a strictly sequential operation, necessitating the use of mutex semaphores (i.e., locks that ensure that only one parallel thread can access the elitist archive at a time).

Furthermore, the GI-GOM operator cannot be parallelized without making changes on an algorithmic level because in GI-GOM every solution exchanges genes with a random other solution sequentially. Therefore, concurrent threads can interfere by selecting the same partners, violating the operator’s sequential design. For parallelization, alternative formulations are required which pre-compute exchange partners. Although this does not diminish the speed-ups achievable through parallelization, it introduces minor algorithmic differences relative to the original GI-GOM. As demonstrated in the supplementary materials, these differences do not affect optimization performance compared to the original GI-GOMEA [8].

Finally, Parallelization of GOM has no algorithmic implications, as each variation can be done fully independently.

3.3 Adaptive Steering

Early experiments with Baymex for learning BN classifiers resulted in large approximation fronts on the training data. However, when evaluated on validation data, many BNs became dominated, indicating substantial overfitting. Figure 1 illustrates this behavior on the SUPPORT dataset (see more in Section 4.2) for which we performed 30 independent runs on 30 different train-validation splits.

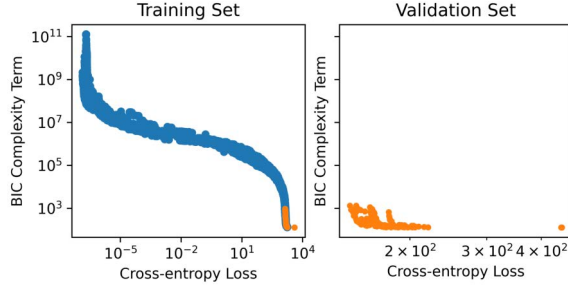


Figure 1: All approximation fronts (cross-entropy loss vs complexity) over 30 train-validation splits for the training set (left) and validation set (right). Corresponding solutions in the training and validation fronts are shown in orange.

Figure 1 (left) shows the approximation fronts obtained on the training folds, with training cross-entropy loss plotted against the BIC complexity term. The resulting BN classifiers cover a wide complexity range, going up to approximately 10^{11} . We then reevaluate the cross-entropy loss and BIC complexity term of these same BNs on the corresponding validation sets and retain only the non-dominated solutions. These validation fronts are shown on the right in Figure 1. The resulting set of BNs constitute a much more narrow range in terms of complexity, going up to only 10^3 .

Baymex appears to spend a substantial part of its computation budget exploring highly complex BN classifiers that perform well on the training set, but are costly to evaluate. Yet, many of these models become Pareto-dominated once validation cross-entropy is considered, indicating that they do not provide real accuracy-complexity benefits and are likely overfitting. This observation motivates the use of mechanisms that bias the search towards regions that are more likely to generalize well, e.g., [1, 17, 35].

In this work, we take inspiration from [1] and augment Baymex with a complexity cap that is adaptively guided by observations on the validation data set during optimization. In addition to having an elitist archive based on training-data objective values, we maintain a second elitist archive based on validation-data objective values. At the start of each generation, the current most complex BN is extracted from the validation elitist archive. The complexity of this solution is used as an adaptive reference threshold in the form of a hard constraint, denoted BIC_θ . Each solution x that gets evaluated in this generation, is assigned a constraint value that represents the excessive complexity of that solution, relative to this threshold: $C(x) = \max\{0, BIC(x) - f_c BIC_\theta\}$, where $BIC(\cdot)$ denotes the BIC complexity term and f_c determines the leeway we allow to discover BNs that are more complex, but have better accuracy. Solutions with BIC complexities smaller than $f_c BIC_\theta$ remain unconstrained, while solutions exceeding $f_c BIC_\theta$ incur increasing constraint values. We subsequently use constraint domination [14] to ensure that any solution that becomes infeasible during variation is rejected, and any solution that is already infeasible is only modified in ways that reduce its constraint violation. Empirical evaluation showed that a leeway factor of $f_c = 10$ provides sufficient flexibility for Baymex to discover more complex and accurate solutions, ensuring that the adaptive constraint does not overly restrict the search.

Since the constraint reference threshold is potentially changed each generation, solutions that were previously constrained may be no longer constrained in a new generation, or vice versa. For this

reason, after the constraint reference threshold has been set, we recompute the constraint values of all solutions in the population. Additionally, we empty the training elitist archive and reinsert those original solutions that satisfy the complexity constraint.

4 Experimental Setup

4.1 Training Pipeline

We compare Baymex against well-established baselines using a common outer train-test split. For the baselines (see Section 4.4.1), model selection is performed by standard 30-fold cross-validation on the training set to obtain many validation estimates of Baymex. Within each fold, candidate models are trained on the training subset, evaluated on the corresponding validation subset, and the model with the best validation score is selected. The final performance is then reported on the held-out test set.

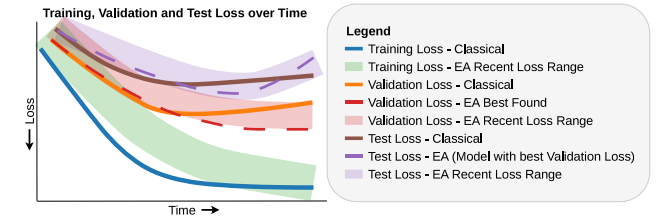


Figure 2: Example of validation-loss overfitting with an EA compared to a gradient-based method. EA search evaluates many more solutions, some with worse training loss but better validation loss, which increases the risk of overfitting when selecting the model with the best validation score.

Baymex can follow the standard model-selection procedure, but it can also use an alternative approach in which a refinement set is withheld from the training data to guide selection from the validation archive. This is motivated by the EA-based optimization underlying Baymex: unlike gradient-based methods, it is not restricted to a greedy improvement path in training loss. EA-based search explores a wider range of loss values, and while some solutions may degrade on the validation set, others may achieve validation scores that surpass the previous best (see Figure 2). Although this stronger exploratory capability improves optimization, it also increases the risk of overfitting when selecting solely based on validation performance. The refinement set therefore provides an additional selection step that mitigates this risk. Because overfitting is typically less severe on the validation set than on the training set, using a separate refinement set is expected to yield the best results when sufficient data is available. However, when data is scarce, splitting off a refinement set may itself induce overfitting, where models are selected by validation performance from the final elitist archive based on training loss, preferable.

4.2 Datasets

We evaluate Baymex and the baselines on three clinical datasets (described below) with a binary overall-survival endpoint, resulting in a binary classification task. When a dataset does not define a fixed-horizon target, we set a clinically meaningful prediction horizon and convert the recorded overall-survival time into a horizon-based label where 1 means survival, and 0 means the patient has deceased

before the prediction horizon. Since most datasets include censored individuals (i.e., the event has not yet occurred at analysis time), the true survival time is not always observed. We therefore consider each patient’s minimum survival time for the classification task, defined as the last available follow-up time (time of death if observed, censoring time otherwise).

SUPPORT. This is a cohort of hospitalized adults. The data contains 14 features including demographic variables, comorbidities, chronic comorbidities, physiological measurements, and laboratory values [11]. We use a publicly available preprocessed version of this data [22] and create a training/validation ($N_{\text{train}} = 7,985$) and held-out test ($N_{\text{test}} = 888$) partition that has no missing values. We applied semantic remapping (real-valued to ordinal or categorical) where clinically meaningful. The exact remapping is provided in our supplementary material.

RADCURE. This is an open-source head-and-neck cancer radiotherapy dataset comprising tabular features and CT images [44]. Here, we only use tabular features. We removed patients with missing values and collapsed rare discrete values based on methods in [28]. We used the provided time-to-event feature to predict two-year survival, as was done in the RADCURE challenge [44]. The selected features (16) and preprocessing steps (filtering and remapping) are fully specified in our supplementary material. After preprocessing, the cohort is reduced to a training/validation set of $N_{\text{train}} = 2,154$ samples and a held-out test set of $N_{\text{test}} = 364$ samples.

Spinal Metastases. This is a dataset from a study led by our clinical department on the survival of patients with advanced symptomatic spinal bone metastases who underwent radiotherapy and/or surgery [5]. The training/validation set contains $N_{\text{train}} = 1,043$ patients from a single-centre retrospective cohort of patients treated between January 2001 and December 2010. As an external validation set, 339 patients that had symptomatic spinal bone metastases are taken from the national Dutch bone metastases study [39]. The datasets contain 16 features and are preprocessed according to [5, 36] and a minimum three-month survival is predicted. Due to privacy regulations, this dataset is not publicly available.

4.3 Baymex Efficiency Enhancements

All runs to study Baymex enhancements were executed on an AMD EPYC Genoa 9654 with a wall-clock budget of 24 hours per fold. Here, we use the SUPPORT dataset, as it is our largest dataset and thus the case where search-efficiency enhancements are most needed. Because larger starting populations in IMS typically require more time to reach good solutions, whereas smaller populations offer less parallelization potential, we considered a range of minimum IMS population sizes: 30, 120, 480, and 1920. Three types of enhancements were studied.

4.3.1 Parallelization. We ran Baymex in two configurations: the original serial version on a single CPU core, and the parallelized version on 16 CPU cores.

4.3.2 End-Of-Generation (EOG) Elitist Archive Update. We study how updating the elitist archive at the end of each generation, rather than during variation, affects the search efficiency of Baymex.

4.3.3 Adaptive Steering. We study how adaptive steering influences the search efficiency of Baymex.

4.4 Clinical Data Classification

All runs of Baymex for clinical data classification were done with the parallel variant of Baymex with adaptive steering. The experiments on the SUPPORT and RADCURE datasets were performed on 16 AMD EPYC Genoa 9654 cores. Due to privacy restrictions, experiments on the spinal metastases dataset were performed on our in-house HPC system using 16 Intel E-2690 cores.

4.4.1 Baseline Classification Algorithms. We compare Baymex to four widely-used, well-established baselines.

- (1) Decision Tree (DT)
- (2) Random Forest (RF)
- (3) Binary Logistic Regression (LR)
- (4) Naive Bayes (NB)

We explicitly do not include models based on deep neural networks as they are not inherently interpretable. While the same can be said for RF, we include it because despite its limited intrinsic interpretability, it is often used as a standard baseline for tabular data, known for its strong and often-reported accuracy.

We use the same 30 training splits for all algorithms. For each baseline, we evaluate various hyperparameter settings (see supplementary material) and report the results obtained with the settings that achieved the best validation or refinement performance.

Decision Tree. A DT classifies data by repeatedly splitting data along branches based on features that best separate the target classes. A single DT can be highly interpretable when it remains small. Furthermore, a smaller tree can reduce overfitting. Therefore, for DT, we tune the maximum depth, minimum samples per split, and cost-complexity pruning strength (ccp_alpha).

Random Forest. An RF is an ensemble model that combines the predictions of multiple DTs, each independently trained on random subsets of samples and features. RF provides a strong non-linear baseline, at the cost of reduced interpretability relative to a single DT. We train an RF classifier on the cleaned dataset and tune the number of trees, maximum depth, number of features to select per tree, minimum number of samples per split, and pruning strength.

Binary Logistic Regression. Binary LR is a statistical method that models the probability of a two-category outcome by linking features to the log-odds of the outcome occurring. LR is considered interpretable because its coefficients correspond directly to log-odds changes. Some data preprocessing is however required. Real-valued data is standardized using z-normalization, categorical data is one-hot encoded, and ordinal data is left as is. We tune the solver, regularization type, regularization strength, and for the elastic-net solver the ℓ_1 mixing parameter.

Naive Bayes. NB is a probabilistic baseline that assumes conditional independence of features given the outcome. It can be viewed as a simplified analogue of BN classifiers, without additional learned structure. To preserve interpretability, we restrict NB to discrete features. For this, we either remove real-valued features or pre-discretize the variables using the technique of [15]. We also tune a

univariate feature-selection step (mutual information or ANOVA F -score ranking) and the Laplace smoothing strength.

4.4.2 Impression of Baymex’s Multi-Objective Front. To characterize the quality of the multi-objective approximation front obtained by Baymex, we aggregate all solutions over all folds and partition them into five complexity-based bins to get an idea of test performance in the groups of models of extreme low- and high-complexity, the knee region, and the intermediate regions between them. We construct the bins using both Equal-Width (EW) and Equal-Frequency (EF) discretization. For each bin, we report the distribution of test ROC-AUC, providing a clearer picture of how predictive performance is spread across the obtained front.

4.4.3 Statistical Testing. For each prediction task, we compare Baymex to the baseline algorithms described in Section 4.4.1 using the held-out test set. To do so, one needs to perform final model selection, however. Here, for simplicity, we choose the best model on the validation or refinement set. However, an expert may in practice choose to study the models and choose a different trade-off between complexity and validation (or refinement) accuracy. Importantly, the model with the best validation score may not be the model with the best test score. This could be another model that could be selected by an expert. For this reason we show test scores along the entire front, in bins, in Figure 8. For the best validation/refinement scoring model, we report the median, and 25%-75% quantiles, test ROC-AUC per algorithm and identify the best-performing algorithm. We then assess whether this best algorithm performs significantly better than each competing algorithm using a paired Wilcoxon signed-rank test with $\alpha = 0.05$. To correct for multiple testing, we apply a Bonferroni correction. In total, we conduct 34 tests, yielding a corrected significance level of $\alpha_{\text{corr}} = \alpha/34$.

5 Results

5.1 Baymex Efficiency Enhancements

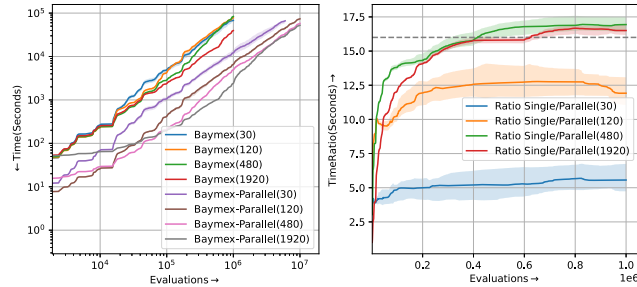


Figure 3: Evaluation throughput of serial and parallel Baymex on SUPPORT across minimum IMS population sizes. Left: wall-clock time at matched evaluation counts. Right: serial-to-parallel wall-clock ratios.

5.1.1 Parallelization. As expected, parallelization yields substantial speedups, increasing with larger IMS minimum population size, as can be seen in Figure 3. Interestingly, for IMS starting population sizes 480 and 1920, the time ratio exceeds the upper bound of achievable efficiency. The reason for this is that BNs with higher complexity take longer to evaluate. In parallel Baymex, BNs of varying complexity are generated at different stages of the search,

causing solutions to enter the elitist archive in a different order than in the single-core setting and resulting in different convergence behavior and solution types being evaluated.

5.1.2 End-Of-Generation (EOG) Elitist Archive Update. Frequent access to the mutex-protected elitist archive can become a bottleneck. Pushing all elitist archive operations to the end of a generation potentially alleviates this. As expected, reducing the frequency of elitist archive updates yields additional speedups, particularly for smaller minimum population sizes in IMS, as can be seen in Figure 4.

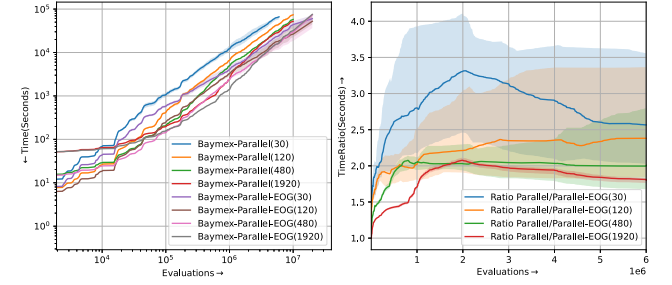


Figure 4: Impact of EOG elitist-archive updates in Baymex-Parallel on SUPPORT, evaluated across IMS minimum population sizes. Left: wall-clock time at matched evaluation counts. Right: wall-clock time ratio.

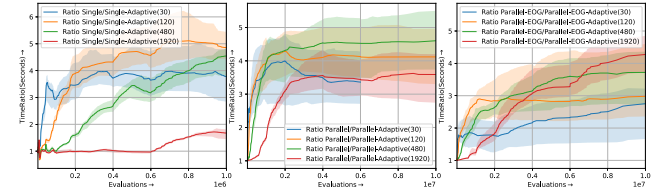


Figure 5: Effect of adaptive steering on the SUPPORT wall-clock time ratio for serial and parallel Baymex, with and without EOG updates, across minimum IMS population sizes.

5.1.3 Adaptive Steering. Figure 5 shows the time ratios of Baymex, Baymex-Parallel, and Baymex-Parallel-EOG relative to their adaptive variants. With the exception of Baymex-Adaptive(1920), adaptive steering yields additional speedups. This is because adaptive steering prevents the construction of overly complex networks, which are substantially more time-consuming to evaluate. Interestingly, Baymex-Adaptive(1920) shows no speedup before 600,000 evaluations and is even slower than the non-adaptive variant before 100,000. This stems from the warm-up required for the adaptive reference point, combined with the extra evaluations needed to re-evaluate constraint values for the entire population.

Overall, these results, together with those from the previous subsection on parallelization, show that to fully exploit N parallel computation cores, it is advantageous to start from an IMS population size that is (much) larger than the default 30 for serial Baymex.

5.1.4 Impact of Baymex Enhancements on Accuracy. Beyond speeding up evaluations, an important question is how the enhancements affect accuracy over time, as they also modify algorithmic behavior. Figure 6 shows the normalized cross-entropy loss of the best solution in the validation elitist archive over time for the different

Baymex variants. For each variant, we present the configuration with IMS minimum population size for which the median (taken over 30 runs) training accuracy of the best-in-front solution found by serial Baymex is matched in the shortest median time.

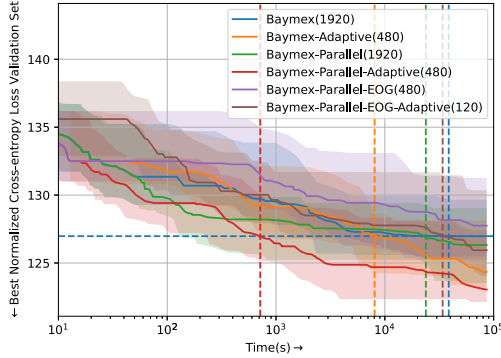


Figure 6: Convergence of best cross-validation value in the validation elitist archive for serial and parallel (EOG/adaptive) Baymex on SUPPORT. Solid lines: median. Shaded regions: 25th-75th quantiles. Dashed lines: time to reach original serial Baymex’s lowest loss.

The three best performing variants all use adaptive steering. Over time, the adaptive variants increasingly outperform their non-adaptive counterparts. Baymex-Parallel with adaptive steering attains the best median validation loss within 24 hours.

Baymex with adaptive steering reaches validation cross-entropy values comparable to those of serial Baymex approximately 1.6 times faster. Combined with parallel execution and EOG updates, the acceleration is up to $\approx 54\times$. To assess whether these efficiency gains also translate into better performance at different wall-clock budgets, we perform statistical tests at the end of the 24-hour budget as well as after 1 hour and after 5 minutes. Table 1 confirms that parallelized Baymex with adaptive steering achieves better cross-validation losses on all wall-budgets inspected. Furthermore, parallelism yields statistically significantly faster convergence, as the best result after 5 minutes is obtained by parallel Baymex. For larger datasets this difference is expected to only increase.

Baymex variant	5 Min	1 Hour	24 Hours
Baymex	131(129-135)	128(126-130)	127(125-130)
Baymex-Adaptive	131(129-134)	128(126-131)	124(123-128)
Baymex-Parallel	128 (126-130)	128(125-129)	126(124-129)
Baymex-Parallel-Adaptive	129 (126-132)	125 (123-129)	123 (122-127)
Baymex-Parallel-EOG	132(131-136)	130(127-133)	128(125-131)
Baymex-Parallel-EOG-Adaptive	130(129-133)	128(127-132)	126(124-130)

Table 1: The median validation cross-entropy loss and (25th-75th quantiles in brackets) at different wall-clock times. Values are rounded for readability. Bold entries indicate the statistically best performing methods.

5.2 Clinical Data Classification

For the final clinical data classification tasks, we used Baymex-Parallel with adaptive steering with an IMS minimum population size of 480 as the preferred variant of Baymex. Below, when we refer to Baymex, we mean this variant.

Algorithm	SUPPORT	RADCURE	Spinal
DT	0.64(0.62-0.65)	0.59(0.57-0.60)	0.74(0.73-0.74)
RF	0.67(0.67-0.68)	0.63(0.61-0.63)	0.76(0.74-0.77)
LR	0.65(0.65-0.66)*	0.61(0.60-0.61)	0.77(0.75-0.77)
NB	0.61(0.61-0.61)	0.64(0.63-0.64)	0.72(0.72-0.72)
Baymex (Refinement)	0.65(0.65-0.65)*	0.60(0.59-0.62)	0.77(0.74-0.78)
Baymex (Validation)	0.65(0.62-0.65)	0.62(0.61-0.64)	0.79(0.77-0.80)

Table 2: The median ROC-AUC of the SUPPORT, RADCURE, and spinal metastases test sets. Numbers between parenthesis are the 25th-75th quantiles. Bold entries indicate the statistically best performing methods. An asterisk (*) indicates the statistically best performing method on the SUPPORT dataset when RF is excluded.

5.2.1 General Baymex Observations. Results for all tested algorithms are summarized in Table 2. For Baymex, both the refinement approach and the validation approach are shown. As hypothesized, for larger datasets, the refinement approach is (slightly) preferable, whereas the converse is true for smaller datasets. In Figure 8, additional details are shown for all algorithms, with in particular extra results for Baymex showcasing the performance along the front via grouping models in separate complexity clusters. Here, only the most appropriate strategy is shown (i.e., refinement for SUPPORT, validation for RADCURE and spinal metastases).

5.2.2 SUPPORT. We ran Baymex for each split for 24 hours. The BN classifier with the best ROC-AUC Test score across 30 splits is shown in Figure 7. Baymex identified an arc between Race and Cancer, which is plausible given that race often correlates with cancer type, see, e.g., [40]. Importantly, we note that the BN captures statistical dependencies that improve predictive performance, not necessarily causal relationships.

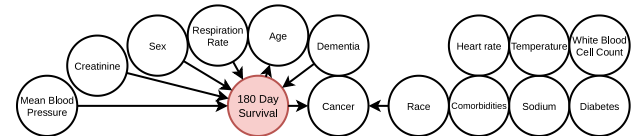


Figure 7: The best found BN classifier structure for SUPPORT (ROC-AUC Test=0.67). Only features connected to the event node are used by Baymex to predict the outcome.

Table 2 shows that RF statistically outperforms all competing methods, although the difference with LR and Baymex is extremely small. When excluding RF due to its limited interpretability, Baymex is statistically tied with LR.

RF shows a marginally higher ROC-AUC for SUPPORT. While Baymex has a powerful optimization engine, this result is not entirely unexpected: ensemble trees excel on heterogeneous tabular datasets and are less affected by structural uncertainty. Baymex, by contrast, must infer a probabilistic dependency structure, making it more sensitive to limited signal or noisy features. This can result in RF holding a slight edge despite overall comparable performance.

5.2.3 RADCURE. We ran Baymex for each split for 24 hours. The BN classifier with the best ROC-AUC test score across 30 splits is shown in Figure 9. In addition to the direct predictors of 2-year survival (HPV, Smoking per Year, ECOG performance status, T-stage,

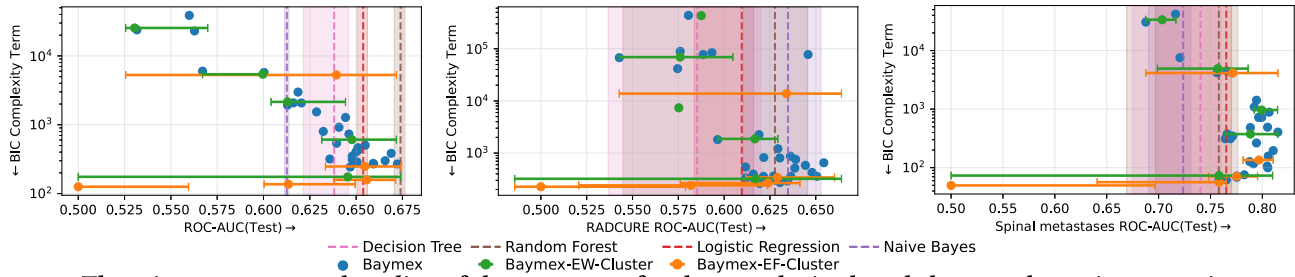


Figure 8: The min-max range and median of the test scores for the complexity-based clusters, shown in comparison to the models selected based on the best validation/refinement score. Left: SUPPORT, middle: RADCURE, right: Spinal metastases.

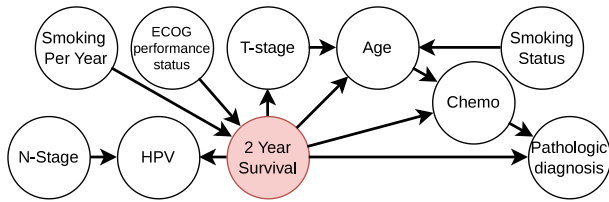


Figure 9: The best found BN classifier structure for RAD-CURE (ROC-AUC(Test)=0.68). Only features connected to the event node are used by Baymex to predict the outcome.

Age, Chemo, and Pathologic diagnosis), Baymex also identified several dependencies among clinical variables. These include arcs between N-stage and HPV, T-stage and Age, Smoking Status and Age, Age and Chemo, and Chemo and Pathologic diagnosis. Except for Pathologic diagnosis, all of these relationships are clinically plausible and consistent with known patterns in head-and-neck oncology [12, 32, 34, 42, 43].

Table 2 shows that, Baymex has the best performance among all tested algorithms, statistically tied with RF and NB. Moreover, in Figure 8, we see that multiple clusters contain solutions with even better test ROC-AUC scores.

5.2.4 Spinal Metastases. We ran Baymex for each split for 6 hours. The BN classifier with the best ROC-AUC Test score across 30 splits is shown in Figure 10. A comparison with literature reveals agreement with previously determined clinically relevant factors [5]. Specifically, Baymex identifies all four features reported as most important, namely: the Karnofsky score, the presence of visceral metastases and clinical profile variables.

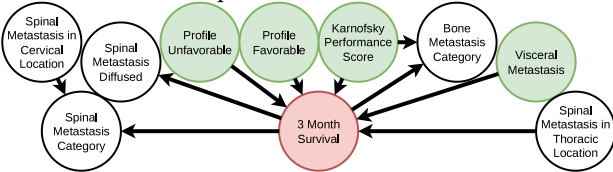


Figure 10: The best found BN classifier structure for spinal metastases (ROC-AUC Test=0.83). Features in green correspond to the four most important features identified in [5].

Table 2 shows that Baymex performs best among all algorithms, statistically tied with LR. Also, Figure 8 shows again multiple clusters containing solutions with even better test ROC-AUC scores.

6 Limitations and Future Work

While parallelization and adaptive steering proved effective in improving Baymex’s efficiency, additional gains may be achieved by

exploiting a key strength of its internal search engine, GOMEA: partial evaluations. In earlier work where BNs were optimized without a classification task, partial evaluations were not implemented because one of the optimized objectives (the Kullback–Leibler Divergence [13]) lacks the decomposability required for incremental score updates. In contrast, BN classification with cross-entropy loss and BIC complexity is decomposable into local score contributions for both objectives. However, preliminary experiments indicate that partial evaluations may introduce a new bottleneck: substantially increased memory consumption due to the need to retain discretized data and associated probability tables per solution. Addressing this challenge remains an important engineering task.

Beyond computational efficiency, this study demonstrates that Baymex can learn competitive BN classifiers from real-world clinical data, but it does not yet leverage Baymex’s multi-objective capabilities for human-centered model selection or knowledge discovery. A natural next step is to integrate domain experts into the evaluation loop to assess whether the learned models align with expert knowledge or meaningfully influence expert beliefs.

Finally, future work should explore how Baymex can support clinical practice more directly by enabling transparent model selection and facilitating joint knowledge discovery with clinicians.

7 Conclusion

In this work, we enhanced Baymex, an evolutionary multi-objective algorithm for Bayesian network structure learning and discretization, and adapted it for BN classification. We showed that parallelization and adaptive steering can substantially accelerate the search procedure. Across three real-world clinical prediction tasks, including our in-house spinal metastases case, Baymex achieved performance that is competitive with, and often superior to, established machine learning methods such as decision trees, logistic regression, naïve Bayes, and random forests. Importantly, Baymex also yields transparent models whose learned dependencies can provide clinically meaningful insights, supporting explainable AI while remaining state of the art for clinical classification.

Acknowledgments

This research is part of the research programme Open Competition Domain Science-KLEIN (project number OCENW.KLEIN.111), which is financed by the Dutch Research Council (NWO). Further, we thank SURF (www.surf.nl) for the support in using the National Supercomputer Snellius. T. Schlender was supported by the Gieskes-Strijbis Fonds (project "Uitlegbare Kunstmatige Intelligentie").

References

- [1] Tanja Alderliesten, Peter A. N. Bosman, and Arjan Bel. 2015. Getting the most out of additional guidance information in deformable image registration by leveraging multi-objective optimization. In *Medical Imaging 2015: Image Processing*, Sébastien Ourselin and Martin A. Styner (Eds.), Vol. 9413. International Society for Optics and Photonics, SPIE, Orlando, Florida, United States, 94131R. doi:10.1117/12.2081438
- [2] João Miguel Alves, Daniel Matos, Tiago Martins, Diogo Cavaco, Pedro Carmo, Pedro Galvão, Francisco Moscoso Costa, Francisco Morgado, António Miguel Ferreira, Pedro Freitas, Cláudia Camila Dias, Pedro Pereira Rodrigues, and Pedro Adragão. 2025. Predicting Atrial Fibrillation Relapse Using Bayesian Networks: Explainable AI Approach. *JMIR Cardio* 9 (11 Feb 2025), e59380. doi:10.2196/59380
- [3] Alice Bernasconi, Alessio Zanga, Peter J. F. Lucas, Marco Scutari, Serena Di Cosimo, Maria Carmen De Santis, Eliana La Rocca, Paolo Baili, Ilaria Cavallo, Paolo Verderio, Chiara M. Ciniselli, Sara Pizzamiglio, Adriana Blanda, Paola Perego, Paola Vallerio, Fabio Stella, Annalisa Trama, and The Ada Working Group. 2024. From Real-World Data to Causally Interpretable Models: A Bayesian Network to Predict Cardiovascular Diseases in Adolescents and Young Adults with Breast Cancer. *Cancers* 16, 21, Article 3643 (2024), 14 pages. doi:10.3390/cancers16213643
- [4] Concha Bielza and Pedro Larrañaga. 2014. Discrete Bayesian Network Classifiers: A Survey. *ACM Comput. Surv.* 47, 1, Article 5 (July 2014), 43 pages. doi:10.1145/2576868
- [5] Laurens Bollen, Yvette M. van der Linden, Willem Pondaag, Marta Fiocco, Bas P.M. Pattinama, Corrie A.M. Marijnen, Rob G.H.H. Nelissen, Wilco C. Peul, and P.D. Sander Dijkstra. 2014. Prognostic factors associated with survival in patients with symptomatic spinal bone metastases: a retrospective cohort study of 1 043 patients. *Neuro-Oncology* 16, 7 (01 2014), 991–998. arXiv:https://academic.oup.com/neuro-oncology/article-pdf/16/7/991/6910747/not318.pdf doi:10.1093/neuonc/not318
- [6] Peter A. N. Bosman and Dirk Thierens. 2013. More Concise and Robust Linkage Learning by Filtering and Combining Linkage Hierarchies. In *Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation Conference (GECCO '13)*. ACM, New York, NY, USA, 359–366.
- [7] Anton Bouter, Ngoc Hoang Luong, Cees Witteveen, Tanja Alderliesten, and Peter A. N. Bosman. 2017. The Multi-Objective Real-Valued Gene-Pool Optimal Mixing Evolutionary Algorithm. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '17)*. Association for Computing Machinery, New York, NY, USA, 537–544. doi:10.1145/3071178.3071274
- [8] Anton Bouter, Dirk Thierens, and Peter A. N. Bosman. 2025. The Pitfalls and Potentials of Adding Gene-invariance to Optimal Mixing. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion (NH Malaga Hotel, Malaga, Spain) (GECCO '25 Companion)*. Association for Computing Machinery, New York, NY, USA, 1918–1926. doi:10.1145/3712255.3734293
- [9] Erick Cantú-Paz and David E. Goldberg. 2000. Efficient parallel genetic algorithms: theory and practice. *Computer Methods in Applied Mechanics and Engineering* 186, 2 (2000), 221–238. doi:10.1016/S0045-7825(99)00385-0
- [10] David Maxwell Chickering. 1996. Learning Bayesian Networks is NP-Complete. In *Learning from Data*, Doug Fisher and Hans-J. Lenz (Eds.). Lecture Notes in Statistics, Vol. 112. Springer, New York, NY, 121–130. doi:10.1007/978-1-4612-2404-4_12
- [11] Alfred F. Jr Connors, Neal V. Dawson, Norman A. Desbiens, et al. 1995. A Controlled Trial to Improve Care for Seriously Ill Hospitalized Patients: The Study to Understand Prognoses and Preferences for Outcomes and Risks of Treatments (SUPPORT). *JAMA* 274, 20 (Nov. 1995), 1591–1598. doi:10.1001/jama.1995.03530200027032
- [12] Geffer Thiago Batista Corrêa, Gabriela Alencar Bandeira, Bruna Gonçalves Cavalcanti, Francis Balduino Guimarães, Cristine Maria de Albuquerque Picado, Daniela Alves Cunha, Livia Morais Cabral, Thays Cesário Farias, Juliana Heránio da Silva, Francisco Luiz Dias, Diogo Arcoverde de Souza, Celso Gonçalves Ramos, Marise de Medeiros Macieira de Albuquerque, Rejane Pereira Neves de Melo, Márcia Figueiredo de Carvalho, Guilherme de Castro, Wanderley Marques Bernardo de Souza, Ricardo Fernando Arrais, and José Eduardo Gonçalves de Melo. 2012. Analysis of ECOG performance status in head and neck squamous cell carcinoma patients: association with sociodemographical and clinical factors, and overall survival. *Supportive Care in Cancer* 20, 10 (2012), 2679–2685. doi:10.1007/s00520-012-1386-y
- [13] Thomas M. Cover and Joy A. Thomas. 2006. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, USA.
- [14] Kalyanmoy Deb. 2000. An efficient constraint handling method for genetic algorithms. *Computer methods in applied mechanics and engineering* 186, 2-4 (2000), 311–338.
- [15] Usama M. Fayyad and Keki B. Irani. 1993. Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-93)*, Vol. 2. Morgan Kaufmann, 1022–1027. https://www.ijcai.org/Proceedings/93-2/Papers/022.pdf
- [16] Nir Friedman, Dan Geiger, and Moises Goldszmidt. 1997. Bayesian Network Classifiers. *Machine Learning* 29 (1997), 131–163. doi:10.1023/A:1007465528199
- [17] Sandra González-Gallardo, Rubén Saborido, Ana B. Ruiz, and Mariano Luque. 2021. Desirable Objective Ranges in Preference-Based Evolutionary Multiobjective Optimization. In *Applications of Evolutionary Computation*, Pedro A. Castillo and Juan Luis Jiménez Laredo (Eds.). Springer International Publishing, Cham, 227–241.
- [18] Arthur Guijt, Dirk Thierens, Tanja Alderliesten, and Peter A.N. Bosman. 2023. The Impact of Asynchrony on Parallel Model-Based EAs. In *Proceedings of the Genetic and Evolutionary Computation Conference (Lisbon, Portugal) (GECCO '23)*. Association for Computing Machinery, New York, NY, USA, 910–918. doi:10.1145/3583131.3590406
- [19] Dany Ha, Tanja Alderliesten, and Peter A.N. Bosman. 2026. *Baymex: A Novel Tool for Expert-Aware Discretized Bayesian Network Learning from Data*. SSRN Working Paper. SSRN. 33 pages. doi:10.2139/ssrn.6029523 Posted January 6, 2026. SSRN abstract ID 6029523.
- [20] Beichen Huang, Ran Cheng, Zhuozhao Li, Yaochu Jin, and Kay Chen Tan. 2025. EvoX: A Distributed GPU-Accelerated Framework for Scalable Evolutionary Computation. *IEEE Transactions on Evolutionary Computation* 29, 5 (2025), 1649–1662. doi:10.1109/TEVC.2024.3388550
- [21] Neville Kenneth Kitson, Anthony C. Constantinou, Zhigao Guo, Yang Liu, and Kiattikun Chobtham. 2023. A survey of Bayesian Network structure learning. *Artificial Intelligence Review* 56 (2023), 8721–8814. doi:10.1007/s10462-022-10351-w
- [22] William Knottenbelt, William McGough, Rebecca Wray, Woody Zhidong Zhang, Jiashuai Liu, Ines Prata Machado, Zeyu Gao, and Mireia Crispin-Ortuazar. 2025. CoxKAN: Kolmogorov-Arnold networks for interpretable, high-performance survival analysis. *Bioinformatics* 41, 8 (Aug. 2025), btaf413. doi:10.1093/bioinformatics/btaf413 PMID: 40685627. PMCID: PMC12341683. Published online 21 Jul 2025..
- [23] Daphne Koller and Nir Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press, Cambridge, MA. doi:10.5555/1795555
- [24] Evangelia Kyrimi, Kudakwashe Dube, Norman Fenton, Ali Fahmi, Mariana Raniere Neves, William Marsh, and Scott McLachlan. 2021. Bayesian networks in healthcare: What is preventing their adoption? *Artificial Intelligence in Medicine* 116 (2021), 102079. doi:10.1016/j.artmed.2021.102079 Epub 2021-04-22.
- [25] Hoang N. Luong and Peter A. N. Bosman. 2012. Elistit Archiving for Multi-Objective Evolutionary Algorithms: To Adapt or Not to Adapt. In *Parallel Problem Solving from Nature - PPSN XII*, Carlos A. Coello Coello, Vincenzo Cutello, Kalyanmoy Deb, Stephanie Forrest, Giuseppe Nicosia, and Mario Pavone (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 72–81.
- [26] Ngoc Hoang Luong, Han La Poutre, and Peter A.N. Bosman. 2018. Multi-objective Gene-pool Optimal Mixing Evolutionary Algorithm with the Interleaved Multi-start Scheme. *Swarm and Evolutionary Computation* 40 (2018), 238–254. doi:10.1016/j.swevo.2018.02.005
- [27] Eduardo Maekawa, Esben Jensen, Pepijn van de Ven, and Kim Mathiasen. 2024. Choosing the right treatment - combining clinicians' expert knowledge with data-driven predictions. *Frontiers in Psychiatry* 15 (Sept. 2024), 1422587. doi:10.3389/fpsy.2024.1422587
- [28] Mafalda Malafaia, Peter A. N. Bosman, Coen Rasch, and Tanja Alderliesten. 2025. Automated and Interpretable Survival Analysis from Multimodal Data. arXiv:2509.21600 [cs.AI] https://arxiv.org/abs/2509.21600
- [29] Pedro Mateus, Ruud G. P. M. van Stiphout, Sanne Mook, Vincenzo Valentini, Maria Anna Gambacorta, Quirina Monzani, Monique Intven, Arnaud Jochems, and Philippe Lambin. 2025. Predicting near-complete pathological response to (chemo)radiotherapy in patients with rectal cancer: A federated learning study. *Medical Physics* 52, 9 (2025), 6854–6865. doi:10.1002/mp.18034
- [30] M. M. Oken, R. H. Creech, D. C. Tormey, J. Horton, T. E. Davis, E. T. McFadden, and P. P. Carbone. 1982. Toxicity and response criteria of the Eastern Cooperative Oncology Group. *American Journal of Clinical Oncology* 5, 6 (1982), 649–655.
- [31] Biche Osong, Hajar Hasannejad, Henk van der Poel, Ben Vanneste, Joep van Roermund, Katja Aben, Johan Van Soest, Inge Van Oort, Laura Hochstenbach, Esther J. Bloemen-van Gorp, Andre Dekker, and Rianne R.R. Fijten. 2024. The value of PROMs for predicting erectile dysfunction in prostate cancer patients with Bayesian network. *Technical Innovations & Patient Support in Radiation Oncology* 31 (2024), 100234. doi:10.1016/j.tipsro.2024.100234
- [32] Jong Chul Park, Geun Hee Ye, Ross Merkin, and Thomas Roberts. 2025. Outcomes for recurrent or metastatic head and neck cancer by HPV status: a systematic review and meta-analysis. *The Oncologist* 30, 4 (04 2025), oyaf043. arXiv:https://academic.oup.com/oncolo/article-pdf/30/4/oyaf043/62909659/oyaf043.pdf doi:10.1093/oncolo/oyaf043
- [33] Judea Pearl. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, San Mateo, CA. xix+552 pages. doi:10.5555/534975 Other printings list ISBN 1558604790 / 9781558604797..
- [34] Cielito C. Reyes-Gibby, Karen O. Anderson, Kelly W. Merriman, Knox H. Todd, Sanjay S. Shete, and Ehab E. Hanna. 2014. Survival patterns in squamous cell carcinoma of the head and neck: Pain as an independent prognostic factor for

- survival. *The Journal of Pain* 15, 10 (2014), 1015–1022. doi:10.1016/j.jpain.2014.07.003
- [35] Ana Ruiz, Rub  n Saborido, and Mariano Luque. 2015. A preference-based evolutionary algorithm for multiobjective optimization: the weighting achievement scalarizing function genetic algorithm. *Journal of Global Optimization* 62, 1 (May 2015), 101–129. doi:10.1007/s10898-014-0214-y
- [36] Thalea Schlender, Catharina J. A. Romme, Yvette M. van der Linden, Luc R. C. W. van Lonkhuijzen, Peter A. N. Bosman, and Tanja Alderliesten. 2025. PISA: An AI Pipeline for Interpretable-by-design Survival Analysis Providing Multiple Complexity-Accuracy Trade-off Models. arXiv:2509.22673 [stat.AP] doi:10.48550/arXiv.2509.22673
- [37] Gideon Schwarz. 1978. Estimating the Dimension of a Model. *The Annals of Statistics* 6, 2 (March 1978), 461–464. doi:10.1214/aos/1176344136
- [38] Erica Secchettin, Salvatore Paiella, Danila Azzolina, Fabio Casciani, Roberto Salvia, Giuseppe Malleo, and Dario Gregori. 2025. Expert Judgment Supporting a Bayesian Network to Model the Survival of Pancreatic Cancer Patients. *Cancers (Basel)* 17, 2 (Jan. 2025), 301. doi:10.3390/cancers17020301
- [39] Elsbeth Steenland, Jan Leer, Hans van Houwelingen, Wendy J Post, Wilbert B van den Hout, Job Kievit, Hanneke de Haes, Hendrik Martijn, Bing Oei, Ernest Vonk, et al. 1999. The effect of a single fraction compared to multiple fractions on painful bone metastases: a global analysis of the Dutch Bone Metastasis Study. *Radiotherapy and Oncology* 52, 2 (1999), 101–109.
- [40] Claire E Thomas and Ulrike Peters. 2025. Genomic landscape of cancer in racially and ethnically diverse populations. *Nature Reviews Genetics* 26, 5 (2025), 336–349.
- [41] Union for International Cancer Control. 2025. *TNM Classification of Malignant Tumours* (9 ed.). Wiley-Blackwell.
- [42] Noam A. VanderWalde, Mary Fleming, Jared Weiss, and Bhishamjit S. Chera. 2013. Treatment of Older Patients With Head and Neck Cancer: A Review. *The Oncologist* 18, 5 (05 2013), 568–578. doi:10.1634/theoncologist.2012-0427
- [43] Jennifer R. Wang, Steven Habbous, Osvaldo Espin-Garcia, Duoduo Chen, Shao Hui Huang, Colleen Simpson, Wei Xu, Fei Fei Liu, Dale H. Brown, Ralph W. Gilbert, Patrick J. Gullane, Jonathan C. Irish, David P. Goldstein, and Geoffrey Liu. 2016. Comorbidity and performance status as independent prognostic factors in patients with head and neck squamous cell carcinoma. *Head & Neck* 38, 5 (2016), 736–742. doi:10.1002/hed.23947
- [44] Mattea L. Welch, Sejin Kim, Andrew J. Hope, Shao Hui Huang, Zhibin Lu, Joseph Marsilla, Michal Kazmierski, Katrina Rey-McIntyre, Tirth Patel, Brian O’Sullivan, John Waldron, Scott Bratman, Benjamin Haibe-Kains, Tony Tadic, Princess Margaret Head, and Neck Site Group. 2024. RADCURE: An open-source head and neck cancer CT dataset for clinical radiation therapy insights. *Medical Physics* 51, 4 (2024), 3101–3109. doi:10.1002/mp.16972 Epub 2024-02-16.
- [45] Yifan Yu, Shuaijie Zhang, Hongkai Li, and Fuzhong Xue. 2025. A bayesian network model for neurocognitive disorders digital screening in Chinese population: development and validation study. *BMC Psychiatry* 25, 1, Article 760 (2025), 14 pages. doi:10.1186/s12888-025-07189-1 Published 04 Aug 2025.
- [46] Fei Zheng and Geoffrey I. Webb. 2010. *Tree Augmented Naive Bayes*. Springer US, Boston, MA, 990–991. doi:10.1007/978-0-387-30164-8_850
- [47] Lu Zhong, Fan Yang, Shanshan Sun, Lijie Wang, Hong Yu, Xiushan Nie, Ailing Liu, Ning Xu, Lanfang Zhang, Mingjuan Zhang, Yue Qi, Huaijun Ji, Guiyuan Liu, Huan Zhao, Yanan Jiang, Jingyi Li, Chengcun Song, Xin Yu, Liu Yang, Jinchao Yu, Hu Feng, Fujun Yang, Xiaolei Guo, and Fuzhong Xue. 2024. Predicting lung cancer survival prognosis based on the conditional survival bayesian network. *BMC Medical Research Methodology* 24, 1 (2024), 16. doi:10.1186/s12874-023-02043-y